



TPC Benchmark™ C  
Full Disclosure Report

---

Oracle's Sun Fire X4800 M2 Server

*Using*

*Oracle Database 11g Release 2*

First Edition  
March 27, 2012

First Printing – March 27, 2012

Copyright © 2012 Oracle and/or its affiliates. All rights reserved.

Oracle, the Sponsor of this benchmark test, believes that the information in this document is accurate as of the publication date. The information in this document is subject to change without notice. The sponsor assumes no responsibility for any errors that may appear in this document.

The pricing information in this document is believed to accurately reflect the current prices as of the publication date. However, the Sponsor provides no warranty of the pricing information in this document.

The performance information in this document is for guidance only. System performance is highly dependent on many factors including system hardware, system and user software, and user application characteristics; relative system performance may vary significantly as a result of these and other factors. The Sponsor does not warrant or represent that a user can or will achieve similar performance. No warranty on system performance or price/performance is expressed or implied in this document.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. UNIX is a registered trademark licensed through X/Open Company, Ltd.

All rights reserved. This product and related documentation are protected by copyright and distributed under licenses restricting its use, copying, distribution, and decompilation. No part of this product or related documentation may be reproduced in any form by any means without prior written authorization of Sun and its licensors, if any.

RESTRICTED RIGHTS LEGEND: Use, duplication, or disclosure by the United States Government is subject to the restrictions set forth in DFARS 252.227-7013 (c)(1)(ii) and FAR 52.227-19, Rights in Technical Data and Computer Software (October 1988).

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

THIS PUBLICATION IS PROVIDED #AS IS# WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NON-INFRINGEMENT.

THIS PUBLICATION COULD INCLUDE TECHNICAL INACCURACIES OR TYPOGRAPHICAL ERRORS. CHANGES ARE PERIODICALLY ADDED TO THE INFORMATION HEREIN; THESE CHANGES WILL BE INCORPORATED IN NEW EDITIONS OF THE PUBLICATION. ORACLE AMERICA, INC. MAY MAKE IMPROVEMENTS AND/OR CHANGES IN THE PRODUCT(S) AND/OR THE PROGRAM(S) DESCRIBED IN THIS PUBLICATION AT ANY TIME.

## Abstract

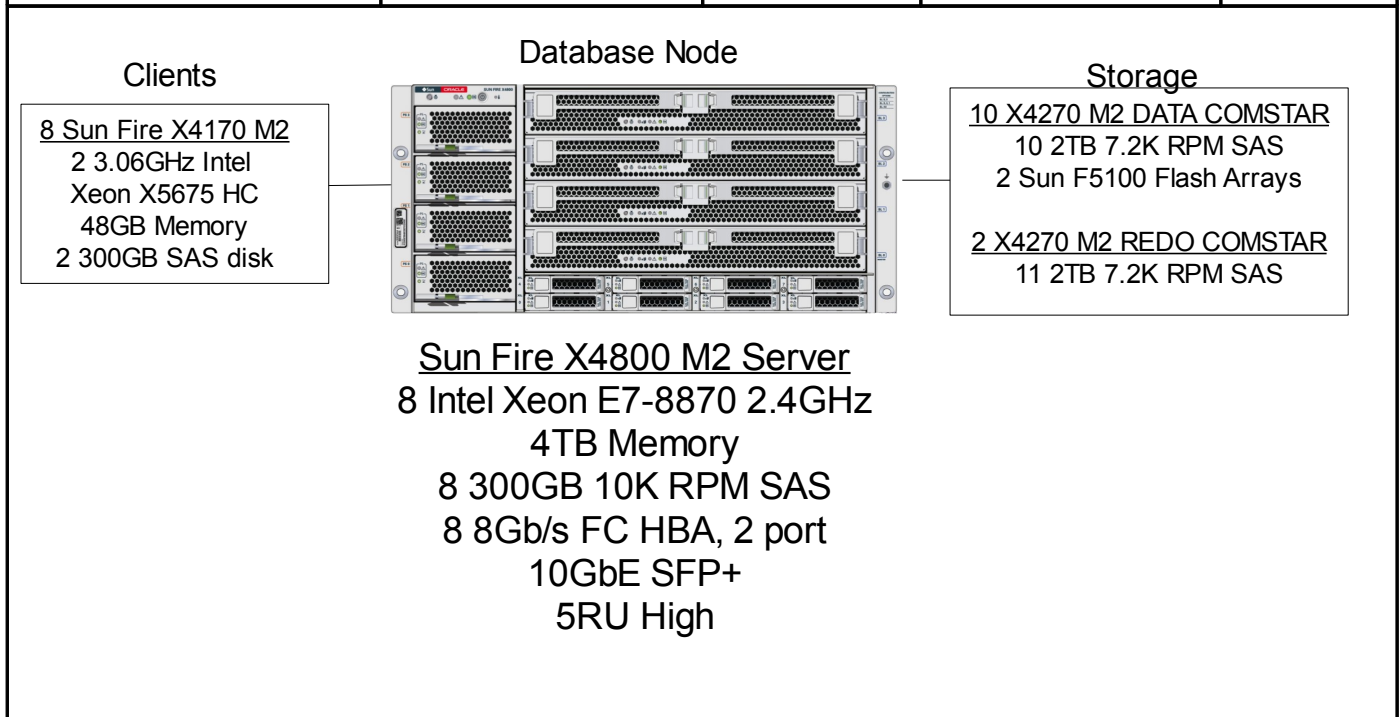
This report documents the methodology and results of the TPC Benchmark™ C test conducted on the following environment as measured by Oracle America, Inc. The benchmark configuration, environment and methodology used to produce and validate the test results, and the pricing model used to calculate the price/performance, were audited by Francois Raab of InfoSizing to verify compliance with the relevant TPC specifications.

<i>System</i>	<i>Processors</i>	<i>Database Environment</i>	<i>Operating System</i>
Sun Fire X4800 M2 Server	8 Intel Xeon E7-8870 2.4GHz	Oracle Database 11g Release 2	Oracle Linux with Unbreakable Enterprise Kernel Release 2

## TPC Benchmark C Metrics

<i>Total System Cost</i>	<i>TPC-C Throughput</i>	<i>Price / Performance</i>	<i>Availability Date</i>
Three year cost includes: <ul style="list-style-type: none"><li>• Hardware</li><li>• Software</li><li>• Maintenance</li></ul>	Maximum Qualified Throughput expressed as transactions per minute – C (tpmC)	Total System Cost / tpmC	Date for which all components, hardware and software are available for purchase
<b>\$4,483,729 USD</b>	<b>5,055,888</b>	<b>\$0.89 USD</b>	<b>June 26, 2012</b>

<b>ORACLE®</b>	<b>Sun Fire X4800 M2 Server</b>		TPC-C 5.11.0 TPC-Pricing 1.7.0	
			Report Date March 27, 2012	
Total System Cost	TPC-C Throughput	Price/Performance	Availability Date	
<b>\$4,483,729USD</b>	<b>5,055,888 tpmC</b>	<b>\$0.89USD/tpmC</b>	<b>June 26, 2012</b>	
Database Server Processors/Cores/Threads	Database Manager	Operating System	Other Software	Number of Users
Intel Xeon E7-8870 2.4GHz 8 / 80 / 160	Oracle Database 11g Release 2 Enterprise Edition	Oracle Linux with UEK Release 2	Tuxedo CFS-R Tier 1 Oracle iPlanet Web Server	4,000,000



System Component	Each Server Node		Each Client	
Processors/Cores/Threads and cache	8/80/160	Intel Xeon E7-8870 2.4GHz 30 MB Smart Cache	2/12/24	Intel Xeon X5675 12MB Smart Cache
Memory		4TB		48GB
Disk Controllers	8	8Gb/s FC HBA 2 Port	1	8 port Internal SAS
OS Disks (each system)	8	300GB 10K RPM SAS	2	300GB 10K RPM SAS
External Storage	1,600 122	24GB SSD Flash Modules 2TB 7.2K RPM SAS		
Total Storage		284.3TB		



# Sun Fire X4800 M2 Server

TPC-C 5.11.0  
TPC-Pricing 1.7.0

Description	Part Number	Price Source	Unit Price	Qty	Extended Price	Support
<b>Server Hardware</b>						
Sun Fire X4800 M2 Server: base chassis	7100176		6,702	1	6,702	
Power Jumper Cables, 2.5m x 1	SR-JUMP-1MC13		29	4	116	
300GB 10K rpm 2.5" SAS-2 HDD	RB-SS2CM-300G10K2		345	8	2,760	
1 Intel® Xeon® E7-8870 2.4GHz Proc w/ heat sink	7100128		7,222	8	57,776	
Two 32 GB DDR3-1066 DIMMs (factory installation)	7103610		4,634	64	296,576	
CPU Module w/ 32 DIMM slots, 2 USB ports	7100177		3,248	4	12,992	
6Gb/s SAS REM RAID HBA, ATO	SG-SAS6-R-REM-Z		1,079	1	1,079	
8Gb/s FC GbE HBA, EM, Qlogic, ATO	SG-PCIEFCGBE-Q8-N		2,089	6	12,534	
8Gb/s FC GbE HBA, EM, Qlogic	SG-XPICIEFCGBE-Q8-N		2,089	2	4,178	
Sun Dual 10GbE PCIe 2.0 FEM	4871A-Z-N		720	1	720	
Sun 10Gbps Dual Rate SFP+ SR	2129A		840	1	840	
Network Express Module	8508A		2,399	2	4,798	
			<b>Sub-Total</b>		<b>401,071</b>	
<b>Server Storage</b>						
Sun Fire X4270 M2 Server chassis	X4270M2-H1-AA	1s	3,618	12	43,416	
Oracle Solaris 11		1	0	12	0	
2TB 7.2K RPM SAS disk	RA-SS1CR-2T7K	1s	949	122	115,778	
Jumper Cable Kit: 1 x 1m C13	SR-JUMP-1MC13-N	1	29	64	1,856	
Intel® Xeon® X5675, 6-core 3.06GHz, w/ heat sink	4373A	1s	2,304	10	23,040	
Intel® Xeon® X5690, 6-core 3.47GHz, w/ heat sink	4375A	1s	2,706	2	5,412	
4GB (1 x 4GB) DDR3-1333 DIMM	4910A	1s	92	26	2,392	
1,200 W AC PSU (for factory installation)	5933A	1s	302	12	3,624	
StorageTek 8Gb/s FC PCI-e HBA dual port Qlogic	SG-PCIE2FC-QF8-Z	1	2,399	12	28,788	
Sun Storage 6 Gb/s SAS PCIe HBA: 8 Port	SGX-SAS6-EXT-Z	1	599	40	23,960	
Sun Storage 6Gb/s SAS PCIe HBA, 512 MB, RAID	SGX-SAS6-R-INT-Z	1	1,079	12	12,948	
Sun Storage F5100 Flash Array	TA-F5100-M2SA	1	19,995	20	399,900	
20 24GB SATA SLC Flash Modules	TA-24GBSTSF-20FM	1	33,750	80	2,700,000	
0.5M, Mini, shielded, SAS cable	XTA-0.5M-SAS	1	95	40	3,800	
15M LC to LC FC cable	X9734A-Z-N	1	105	24	2,520	
			<b>Sub-Total</b>		<b>3,367,434</b>	
<b>Server Software</b>						
Oracle Linux		1	0	1	0	
Oracle Linux Basic Support - 3 Years		1	3,597	1		3,597
Oracle 11g Enterprise Edition Per Processor for 3 years (for 40 processors)		1	23,750	40	950,000	
Partitioning, Per Processor for 3 years (for 40 processors)		1	5,750	40	230,000	
Incident Server Support for 3 years		1	2,300	3		6,900
			<b>Sub-Total</b>		<b>1,180,000</b>	<b>10,497</b>
<b>Client Hardware</b>						
Sun Fire X4170 M2	X4170M2-H1-AA	1	2,378	8	19,024	
300GB 10K RPM 2.5" SAS disk	RB-SS2CF-300G10K2	1	345	16	5,520	
Snap-In Slide Rail Rackmount kit	6325A-N	1	70	8	560	
Jumper Cable Kit: 1 x 1m C13	SR-JUMP-1MC13	1	29	8	232	
Intel® Xeon® X5675 3.06GHz, 6-core, w/ heat sink	4373A	1s	2,304	16	36,864	
Sun Storage 6Gb/s SAS PCIe HBA, Internal: 8 port	SG-SAS6-INT-Z	1	419	8	3,352	
4GB (1 x 4GB) DDR3-1333 DIMM	4910A	1s	92	96	8,832	
			<b>Sub-Total</b>		<b>74,384</b>	
<b>Client Software</b>						
Oracle Solaris 10 Pre-Install	5908A	1	0	8	0	
Oracle Solaris Development Tools Support	B59320	1	1,200	3		3,600
Oracle Fusion Middleware Web Tier for 3 years		1	2,500	48	120,000	
Oracle Premier Software Support		1	26,400	3		79,200
Tuxedo CFS-R Tier 1			1,800	8	14,400	
Oracle Premier Support for 3 years (Tuxedo CFS-R Tier 1)			3,168	3		9,504
			<b>Sub-Total</b>		<b>134,400</b>	<b>92,304</b>

## Other Hardware

Brocade 5300 w/ 80 8Gb/s SFPs	SGXSWBRO5300-8EB-N	1	139,995	1	139,995	
Rackmount kit for the Brocade 5300	SG-XSWBRO3X50-RK-N	1	250	1	250	
Jumper Cable Kit: 1 x 1m C13	SR-JUMP-1MC13	1	29	2	58	
<b>SUT Mgmt system and Service Processor Control Workstation</b>						
Sun Fire X4170 M2	X4170M2-H1-AA	1	2,378	1	2,378	
Oracle Solaris 10 Pre-Install	5894A-N	1	0	1	0	
US PC Peripheral Kit (Keyboard/Mouse)	X3701A-PC	1	50	1	50	
4GB (1x4GB) DDR3-1333	4910A	1	92	2	184	
Sun Storage 6Gb/s SAS PCIe HBA, Internal: 8 port	SG-SAS6-INT-Z	1	419	1	419	
300GB 10K RPM 2.5" SAS Disk	RB-SS2CF-300G10K2	1	345	1	345	
DVD+/-RW SATA-based drive ATO	8325A-N	1	134	1	134	
Jumper Cable Kit: 1 x 1m C13	SR-JUMP-1MC13	1	29	1	29	
Intel® Xeon® E5620 2.4GHz, 4-core, w/ heat sink	5924A	1	683	1	683	
Sun Rack II 42U	SR-1242E-N	1	2,849	2	5,698	
PDU 15kVA, Single Phase, LV	SR-15K-L630-N	1	1,200	4	4,800	
Jumper Cable Kit SunRack II	SR-JUMPKIT-N	1	198	2	396	
2M LC to LC FC Optical Cable	X9732A-Z-N	1	65	1	65	
Acer V173 DJb 17" LCD	2091793	2	102	3	306	
Brocade FastIronGS 648P switch	FGS648P	2	4,977	1	4,977	3,669
Brocade expansion module (10GbE)	FLS-1XG	2	850	1	850	
Netgear ProSafe Plus GS524E 24 port	JGS524E-100NAS	2	279	3	837	
Tripp Lite 25' Black Cat5e cable (+10%)	N002-025-BK	2	7	103	721	
			<b>Sub-Total</b>		<b>163,175</b>	
Oracle Premier Hardware Support	Q-PREM-SPRT-SYS	1	479,805	3		1,439,415
			<b>Total</b>		<b>5,320,464</b>	<b>1,545,885</b>
Total Oracle Software, Hardware and Maintenance Discount					(2,382,620)	
<b>3 Year Total Cost</b>					<b>\$4,483,729</b>	
					<b>tpmC</b>	<b>5,055,888</b>
					<b>\$/tpmC</b>	<b>\$0.89</b>

### Pricing Sources:

1. Oracle
2. CDW

### Notes:

1s – One or more components of the measured configuration have been substituted in the priced configuration. See the FDR for details.

### Audited by Francois Raab of InfoSizing, Inc.

Prices used in TPC Benchmarks reflect the actual prices a customer would pay for a one-time purchase of the stated components. Individually negotiated discounts are not permitted. Special prices based on assumptions about past or future purchases are not permitted. All discounts reflect standard pricing policies for the listed components. For complete details, see the pricing section of the TPC benchmark specifications. If you find that stated prices are not available according to these terms, please inform the TPC at [pricing@tpc.org](mailto:pricing@tpc.org). Thank you.



# Sun Fire X4800 M2 Server

TPC-C 5.11.0  
TPC-Pricing 1.7.0

Report Date  
March 27, 2012

## Numerical Quantities Summary:

MQTh, computed Maximum Qualified Throughput 5,055,888

<b>Response Time in Seconds</b>	<b>90<sup>th</sup> %</b>	<b>Avg.</b>	<b>Max.</b>
New-Order	0.22	0.166	2.531
Payment	0.21	0.163	2.671
Order-Status	0.21	0.163	2.446
Delivery (Interactive)	0.18	0.149	1.876
Delivery (Deferred)	0.02	0.010	1.001
Stock-Level	0.21	0.161	2.312
Menu	0.18	0.149	1.995

Response Time delay added for emulated components 0.1

## **Transaction Mix**, in percent of total transactions

New-Order	44.96%
Payment	43.01%
Order-Status	4.01%
Delivery	4.01%
Stock-Level	4.01%

## **Keying / Think Times** (in seconds)

	<b>Min.</b>		<b>Avg.</b>		<b>Max.</b>	
New-Order	18.001	0.001	18.007	12.028	18.938	120.851
Payment	3.001	0.001	3.007	12.028	3.934	120.734
Order-Status	2.001	0.001	2.007	10.034	2.919	100.664
Delivery	2.001	0.001	2.007	5.042	2.919	50.711
Stock-Level	2.001	0.001	2.007	5.042	2.916	50.957

## **Test Duration**

Ramp-up Time	35 minutes
Measurement Interval	134 minutes, 10 seconds
Number of Checkpoints	5
Checkpoint Interval	longest Interval: 1632 seconds
Number of transactions (all types) completed in Measurement Interval	1508749941

Benchmark Sponsor: Brad Carlile  
 Sr. Director, Strategic Applications Engineering (SAE)  
 Oracle Corporation  
 3295 NW 211th Terrace  
 Hillsboro OR 97124  
 March 26, 2012

I verified the TPC Benchmark™ C performance of the following Client Server configuration:

Platform: Sun Fire X4800 M2 Server  
 Operating system: Oracle Linux with UEK Release 2  
 Database Manager: Oracle Database 11g Release 2 Enterprise Edition  
 Transaction Manager: Tuxedo CFS-R Tier-1

The results were:

CPU's Speed	Memory	Disks	New-Order 90% RT	tpmC
<b>Server: Sun Fire X4800 M2 Server</b>				
8 x Intel Xeon E7-8870 (2.4 GHz)	4 TB (30 MB Smart Cache per processor)	8 x 300 GB 10K rpm SAS 1,600 x 24 GB SSD 122 x 2 TB 7.2K rpm SAS	0.22 Sec.	<b>5,055,888.02</b>
<b>Eight (8) Clients: Sun Fire X4170 M2 Server (each with)</b>				
2 x Intel Xeon X5675 HC (3.06 GHz)	48 GB (12 MB Smart Cache per processor)	2 x 300 GB 10K rpm SAS	n/a	n/a

In my opinion, these performance results were produced in compliance with the TPC requirements for the



benchmark.

The following verification items were given special attention:

- The transactions were correctly implemented
- The database records were the proper size
- The database was properly scaled and populated
- The ACID properties were met
- Input data was generated according to the specified percentages
- The transaction cycle times included the required keying and think times
- The reported response times were correctly measured
- At least 90% of all delivery transactions met the 80 Second completion time limit
- All 90% response times were under the specified maximums
- The measurement interval was representative of steady state conditions
- The reported measurement interval was 134 minutes, 10 seconds
- On-going incremental checkpoints were used during the measurement interval
- The 60 day storage requirement was correctly computed
- The system pricing was verified for major components and maintenance

Additional Audit Notes:

The measured configuration included:

- (10) Sun Fire X4275 Servers used as external storage controllers configured using COMSTAR that were substituted by (10) Sun Fire X4270 M2 Servers in the priced configuration
- (122) 2TB 7.2K rpm SATA disks that were substituted by (122) 2TB 7.2K rpm SAS disks in the priced configuration
- (8) X4170 M2 Clients each with:
  - (2) X5670 CPUs @ 2.4 GHz that were substituted by (2) X5675 CPUs @ 3.06 GHz
  - (2) 146 GB 10K rpm SAS disks that were substituted by (2) 300 GB 10K rpm SAS disks)

Based on the specifications of these items and on performance data collected during testing, it is my opinion that these substitutions have no significant effect on performance.

Respectfully Yours,



François Raab, President

# Table of Contents

TPC Benchmark C Metrics.....	3
Introduction.....	20
0 General Items.....	21
0.1 Application Code and Definition Statements.....	21
0.2 Sponsor.....	21
0.3 Parameter Settings.....	21
0.4 Configuration Diagrams.....	21
1 Clause 1: Logical Database Design Related Items.....	27
1.1 Table Definitions.....	27
1.2 Physical Organization of Database.....	27
1.3 Insert and Delete Operations.....	27
1.4 Partitioning.....	27
2 Clause 2: Transaction And Terminal Profiles Related Items .....	28
2.1 Random Number Generation.....	28
2.2 Input/Output Screen Layouts.....	28
2.3 Terminal Feature Verification.....	28
2.4 Presentation Manager or Intelligent Terminal.....	28
2.5 Percentage of Home and Remote Order-lines.....	28
2.6 Percentage of Rolled Back New-Orders.....	28
2.7 Items per New-Order.....	28
2.8 Percentage of Home and Remote Payments.....	28
2.9 Percentage of Access by Last Name.....	28
2.10 Percentage of Skipped Deliveries.....	29
2.11 Transaction Mix.....	29
2.12 Queuing Mechanism.....	29
3 Clause 3 Transaction and System Properties Related Items.....	30
3.1 Transaction System Properties (ACID).....	30
3.2 Atomicity.....	30
3.2.1 Completed Transaction.....	30
3.2.2 Aborted Transaction.....	30
3.2.3 Consistency.....	30
3.3 Isolation Tests.....	30
3.3.1 Isolation Test 1.....	31

3.3.2	Isolation Test 2.....	31
3.3.3	Isolation Test 3.....	31
3.3.4	Isolation Test 4.....	31
3.3.5	Isolation Test 5.....	32
3.3.6	Isolation Test 6.....	32
3.3.7	Isolation Test 7.....	32
3.3.8	Isolation Test 8.....	32
3.3.9	Isolation Test 9.....	33
3.4	Durability.....	33
3.4.1	Loss of Memory, Instantaneous Interruption, Power Loss, and Loss of Log.....	33
3.4.2	Loss of Durable Media Containing Database Tables.....	33
4	Clause 4: Scaling and Database Population Related Items.....	35
4.1	Initial Cardinality of Tables.....	35
4.2	Database Layout.....	35
4.3	Type of Database.....	35
4.4	Mapping of Database.....	35
4.5	60 Day Space Computation.....	36
5	Clause 5: Performance Metrics and Response Time Related Items.....	38
5.1	Measured tpmC.....	38
5.2	Response Times.....	38
5.3	Keying and Think Times.....	38
5.4	Response Time Frequency Distribution Curves .....	38
5.5	Think Time Frequency Distribution.....	41
5.6	Response Times versus Throughput.....	41
5.7	Throughput versus Elapsed Time.....	42
5.8	Steady State Determination.....	42
5.9	Work Performed During Steady State.....	42
5.10	Measurement Period Duration.....	42
5.11	Transaction Mix Regulation.....	43
5.12	Transaction Mix.....	43
5.13	Percentage of New-Order Transactions.....	43
5.14	Number of Order-lines per New-Order.....	43
5.15	Percentage of Remote Order-lines per New-Order.....	43
5.16	Percentage of Remote Payments.....	43
5.17	Percentage of Non-Primary access by C_LAST for Payment and Order-Status.....	43
5.18	Percentage of Skipped Delivery Transactions.....	43

5.19 Checkpoints.....	43
6 Clause 6: SUT, Driver and Communications Related Items.....	45
6.1 RTE Description.....	45
6.2 Lost Connections.....	45
6.3 Emulated Components.....	45
6.4 Configuration Diagrams.....	45
6.5 Network Configuration.....	45
6.6 Operator Intervention.....	46
7 Clause 7: Pricing Related Items.....	47
7.1 Hardware and Software Component Pricing.....	47
7.2 Total Three Year Cost.....	47
7.3 Availability.....	47
7.4 Hardware and Software Support.....	47
7.5 Statement of tpmC, Price/Performance.....	47
7.6 Country Specific Pricing.....	48
7.7 Orderability Date.....	48
8 Clause 8: Audit Related Items.....	49
8.1 Auditor's Report.....	49
Appendix A: Application Source.....	51
Client Source Code.....	51
tpccClient.c.....	51
tpccClient.h.....	51
tpccTux.c.....	51
tpccTux.h.....	53
tpccDiag.c.....	53
tpccDiag.h.....	53
tpccNsapi.c.....	53
tpccService.c.....	56
tpccService.h.....	71
tpccBool.h.....	71
tpccConst.h.....	71
tpccData.h.....	71
tpcc.h.....	71
Multi_svr.....	72
ora_errrpt.c.....	72
ora_err.h.....	73

tpcc_srv_del.c.....	73
tpcc_srv_init.c.....	79
tpcc_srv_util.c.....	79
ora_oci.h.....	80
tpccflags.h.....	82
tpcc.h.....	82
Uni_svr.....	83
ora_errrpt.c.....	83
tpcc_srv_init.c.....	84
tpcc_srv_newo.c.....	84
tpcc_srv_ords.c.....	89
tpcc_srv_paym.c.....	93
tpcc_srv_stock.c.....	98
tpcc_srv_util.c.....	99
Stored Procedures.....	100
paynz.sql.....	100
payz.sql.....	101
tkvcpdel.sql.....	101
tpvcpnew.sql.....	101
Appendix B: Database Build.....	105
analyze.sql.....	105
createdb.sql.....	105
createindex_icust1.sql.....	105
createindex_icust2.sql.....	105
createindex_idist.sql.....	106
createindex_iitem.sql.....	106
createindex_inord.sql.....	106
createindex_iordl.sql.....	106
createindex_iordr1.sql.....	106
createindex_iordr2.sql.....	106
createindex_istok.sql.....	106
createindex_iware.sql.....	106
createspacestats.sql.....	106
createstoredprocs.sql.....	107
createtable_cust.sql.....	107
createtable_dist.sql.....	107

createtable_hist.sql.....	107
createtable_item.sql.....	107
createtable_nord.sql.....	108
createtable_orcl.sql.....	108
createtable_orcl.sql.....	108
createtable_stok.sql.....	108
createtable_ware.sql.....	109
createts.sh.....	109
junk.....	109
loadcust.sh.....	111
loaddist.sh.....	115
loadhist.sh.....	115
loaditem.sh.....	116
loadnord.sh.....	116
loadordrordl.sh.....	120
loadstok.sh.....	126
loadware.sh.....	130
p_build2.ora.....	130
p_build.ora.....	130
p_create.ora.....	130
tkvcinin.sql.....	130
tpccload.c.....	136
tkvcinin.sql.....	152
RTE Parameters.....	158
config.....	158
Appendix C: Parameter Settings.....	159
Oracle Data-Base Parameters.....	159
p_run.ora.....	159
listener.ora.....	159
tnsnames.ora.....	159
Sun Fire X4800 M2 Parameters.....	159
version.....	160
cmdline.....	160
sysctl.conf.....	160
meminfo.....	160
cpuinfo.....	160

fdisk.....	161
scsi_id.out .....	284
30-tpcc-disks.rules.....	296
balance_eth_irqs.sh.....	331
balance_qla_irqs.sh.....	331
limits.conf.....	332
COMSTAR DATA Heads.....	332
Measured COMSTAR DATA head (X4275) .....	332
uname.....	332
prtdiag.....	332
prtconf.....	334
format.....	336
format for sample FMod.....	340
/etc/system.....	340
mpt.conf.....	341
sbadm.....	341
metastat .....	343
Measured and Priced COMSTAR DATA head (X4270 M2) .....	350
uname.....	350
prtdiag.....	350
prtconf.....	351
format.....	352
Format for sample FMod.....	358
/etc/system.....	358
mpt.conf.....	359
sbadm.....	359
metastat .....	361
COMSTAR REDO Heads.....	367
Measured Comstar REDO head (X4275).....	367
uname.....	367
prtdiag.....	367
prtconf.....	368
format.....	368
/etc/vfstab.....	368
/etc/system.....	369
mpt.conf.....	369

niwot.conf.....	370
sbadm .....	370
metastat.....	371
Priced and Measured COMSTAR REDO Head (X4270 M2).....	371
uname.....	371
prtdiag.....	371
prtconf.....	372
format.....	372
/etc/vfstab.....	372
/etc/system.....	373
mpt.conf.....	373
/etc/project.....	373
niwot.conf .....	373
sbadm.....	374
metastat .....	375
Clients.....	375
prtdiag.....	375
prtconf.....	375
format.....	376
/etc/release.....	376
/etc/vfstab.....	376
/etc/hosts.....	376
/etc/system.....	377
/etc/project.....	377
/etc/user_attr.....	377
metastat.....	377
ubbconfig.....	378
Brocade 5300 SAN Switch.....	380
Appendix D: Third Party Pricing.....	385



## *List of Figures*

Fig 1: Measured ConfigurationPriced Configuration.....	24
Figure 1: Database Mapping.....	35
Figure 2: 60 Day Space Calculations.....	36
Figure 3: New-Order Response Time Distribution.....	37
Figure 3: Payment Response Time Distribution.....	38
Figure 4: Order-Status Response Time Distribution.....	38
Figure 5: Delivery (Interactive) Response Time Distribution.....	39
Figure 6: Stock-Level Response Time Distribution.....	39
Figure 7: New-Order Think Time Distribution.....	40
Figure 8: New-Order Response Time versus Throughput.....	40
Figure 9: New-Order Throughput versus Time .....	41

## *List of Tables*

Table 1: X4800 M2 Server Configuration.....	20
Table 2: DATA COMSTAR Configuration.....	21
Table 3: REDO COMSTAR Configuration.....	21
Table 4: Client Configuration.....	22
Table 5: COMSTAR External Controller Substitution.....	23
Table 6: Transaction Input Percentages and Mix.....	28
Table 7: Table Cardinality.....	34
Table 8: RTE Parameter Input.....	44
Table 9: Statement of tpmC and Price/Performance.....	46
Table 10: Orderability Dates.....	47

## *Preface*

This report documents the compliance of the Oracle TPC Benchmark™ C testing on the Sun Fire X4800 M2 Server running Oracle Database 11g Release 2, executing the TPC Benchmark™ C Standard, Revision 5.11.0.

The TPC Benchmark™ C Full Disclosure Report is organized as follows:

- The main body of the document lists each item in Clause 8 of the TPC Benchmark™ C Standard and explains how each specification is satisfied.
- Appendix A contains the application source code that implements the Tuxedo CFS-R Tier 1 transaction server code and Oracle iPlanet Web Server plug-ins.
- Appendix B contains the code used to create and load the database.
- Appendix C contains the configuration information for the Oracle Linux with UEK Release 2, Oracle Solaris 11, Oracle iPlanet Web Server, Oracle Database 11g Release 2 and Tuxedo CFS-R Tier 1.

# ***Sun Fire X4800 M2 Server TPC Benchmark™ C Full Disclosure***

## ***Introduction***

The TPC Benchmark™ C Standard Specification requires test sponsors to publish, and make available to the public, a full disclosure report for the results to be considered compliant with the Standard.

This report is intended to satisfy the Standard's requirement for full disclosure. It documents the compliance of the benchmark tests required in the *TPC Benchmark™ C* results for the Sun Fire X4800 M2 Server running Oracle Database 11g Release 2.

In the *Standard Specification*, the main headings in Clause 8 are keyed to the other clauses. The headings in this report use the same sequence, so that they correspond to the titles or subjects referred to in Clause 8.

Each section in this report begins with the text of the corresponding item from Clause 8 of the *Standard Specification*, printed in italic type. The plain type text that follows explains how the tests comply with the TPC-C Benchmark

# 0 General Items

## 0.1 Application Code and Definition Statements

The application program (as defined in Clause 2.1.7) must be disclosed. This includes, but is not limited to, the code implementing the five transactions and the terminal input and output functions.

Appendix A contains the application source code that implements the transactions and forms modules.

## 0.2 Sponsor

A statement identifying the benchmark sponsor(s) and other participating companies must be provided.

This benchmark test was sponsored by Oracle America, Inc.

## 0.3 Parameter Settings

Settings must be provided for all customer-tunable parameters and options which have been changed from the defaults found in actual products, including but not limited to:

- Database tuning options
- Recovery/commit options
- Consistency/locking options
- Operating system and application configuration parameters

Appendix C contains all the required parameter settings for the X4800 M2 Server database node, the X4270M2 COMSTAR nodes, the Brocade 5300 SAN switches, the X4170M2 clients, all Oracle Linux with UEK Release 2 and Oracle Solaris 11 tunables, along with parameters for Oracle Database 11g Release 2, Oracle iPlanet Web Server and Tuxedo CFS-R Tier 1.

## 0.4 Configuration Diagrams

Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences.

### Database Nodes Description

The X4800 M2 Server is configured as described below:

	<i>Sun Fire X4800 M2 Server</i>
Processors	8 Intel Xeon E7-8870 2.4GHz
Processors / Cores / Threads	8 / 80 / 160
Memory	4 TB DDR3
Disks	8 300GB 10K RPM 2.5" SAS
Adapters	8 8Gb/s dual port PCI-E FC-AL
	1 internal PCI-E SAS
Networks	2 10GbE dual port PCI-E SFP+ Ethernet
	8 port 1GbE integrated Ethernet
Service Processor	Integrated

**Table 1: X4800 M2 Server Configuration**

### DATA COMSTAR Heads Description

The measured configuration consists of 9 Sun Fire X4275 Servers and 1 (priced) Sun Fire X4270M2 running Oracle Solaris 11 as Common Multi-protocol SCSI Target (COMSTAR) heads. The COMSTAR heads are configured and used as external storage controllers. The Sun Fire X4275 Server is no longer orderable and has been replaced by the Sun Fire X4270 M2 Server in the Priced Configuration. Each of the COMSTAR DATA heads are configured with the same hardware peripherals (except for the 2TB disks, SATA vs. SAS) and were loaded with the same distribution of database tables and indexes. The differences of the Sun Fire X4275 Server and the Sun Fire X4270 M2 Server are detailed below:

	<i>Sun Fire X4275 Server (Measured)</i>	<i>Sun Fire X4270 M2 Server (Priced)</i>
Processor	Intel Xeon E5540 2.53GHz QC	Intel Xeon X5675 3.06GHz HC
Processors / Cores / Threads	1 / 4 / 8	1 / 6 / 12
Memory	6GB	8GB
Disks	12 2TB SATA 3.5" 7.2K RPM	10 2TB SAS 3.5" 7.2K RPM
Adapters	1 8Gb/s dual port PCI-E FC-AL	
	4 6Gb/s external SAS PCI-E	
	1 6Gb/s internal SAS PCI-E	1 6Gb/s internal SAS/SATA RAID PCI-E
Network	4 port 1GbE integrated Ethernet	
Service Processor	Integrated	

**Table 2: DATA COMSTAR Configuration**

Each DATA COMSTAR head is configured with 2 Sun Storage F5100 Flash Arrays. Each F5100 array contains 80 24GB FMODS (Flash Modules), for a total of 1.92TB. The F5100 arrays are connected to the 8 port external SAS PCI-E adapters via 0.5m SAS cables. Each SAS port is connected to a FMOD expander port on the F5100 array for a total of 8 external SAS ports on the COMSTAR head to 8 FMOD expanders on the 2 F5100 arrays. Each FMOD is visible to the COMSTAR head as a separate device for a total of 160 devices per DATA COMSTAR head. Each of the 160 FMODs are partitioned into 4 equal sizes. Using Solaris Volume Manager (SVM), 4 160-way RAID-0 stripes are created. From these 4 stripes 289-290 soft partitions are created, sized for 1 datafile each. These are exported as LUNs over the 8Gb/s FC-AL to the Brocade 5300 SAN switch and to the X4800 M2 Server.

The COMSTAR heads of the DATA SAN have 10 2TB disks for Oracle Solaris 11 and to satisfy the 60 day space requirements. No database tables were stored on the 2TB disks for the measurement, they were used solely for backup of the initial database and for 60 day storage. Each DATA COMSTAR head is connected to the Brocade 5300 SAN switch via 2 8Gb/s fibre channel connections.

**REDO COMSTAR Heads Description (Logs)**

The measured configuration of the REDO consists of 1 X4275 Server and 1 (priced) X4270 M2 running Oracle Solaris 11 as Common Multi-protocol SCSI Target (COMSTAR) heads. The COMSTAR heads are configured and used as external storage controllers. Each of the REDO COMSTAR heads is configured with the same hardware peripherals, version of Oracle Solaris 11, and managed the same load for the Oracle redo (recovery) logs. Each of COMSTAR REDO heads are configured with the same hardware peripherals (except for the 2TB disks, SATA vs. SAS). The differences of the Sun Fire X4275 Server and the Sun Fire X4270 M2 Server are detailed below:

	<i>Sun Fire X4275 Server (Measured)</i>	<i>Sun Fire X4270 M2 Server (Priced)</i>
Processor	Intel Xeon E5540 2.53GHz QC	Intel Xeon X5690 3.47GHz HC
Processors / Cores / Threads	1 / 4 / 8	1 / 6 / 12
Memory	6GB	12GB
Disks	11 2TB SATA 3.5" 7.2K RPM	11 2TB SAS 3.5" 7.2K RPM
Adapters	1 8Gb/s dual port PCI-E FC-AL	
	1 6Gb/s internal SAS/SATA RAID PCI-E	
Network	4 port 1GbE integrated Ethernet	
Service Processor	Integrated	

**Table 3: REDO COMSTAR Configuration**

Each REDO COMSTAR is configured with 11 2TB SATA (or SAS) disks through the 6Gb/s Internal RAID adapter. One 2TB disk is dedicated for the OS and local filesystems. The remaining 10 2TB disks are configured as a 10-way striped Logical disk using the internal LSI HBA RAID controller. The write cache on the controller is enabled. Two disk slices are created on the Logical disk and exported as LUNs. These LUNs are exported to the Brocade 5300 SAN switch via 8Gb/s FC-AL. Mirroring and control of writes is handled through Oracle Database 11g Release 2. All redo log devices are directly connected to the X4800 M2 Server via 1 8Gb/s fibre channel connection.

**Client Configuration Description**

The TPC-C transactions are entered via a Remote Terminal Emulator which communicates with Oracle iPlanet Web Server running on 8 Sun Fire X4170 M2 clients. Each client is configured with:

	<i>Sun Fire X4170 M2 Server (Measured)</i>	<i>Sun Fire X4170 M2 Server (Priced)</i>
Processors	2 Intel Xeon X5670 2.93GHz HC	2 Intel Xeon X5675 3.06GHz HC
Processors/Cores/ Threads	2 / 12 / 24	
Memory	48 GB DDR3	
Disks	2 146GB 10K RPM 2.5" SAS	2 300GB 10K RPM 2.5" SAS
Adapters	1 6Gb/s internal PCI-E SAS	
Network	4 port 1GbE integrated Ethernet	
Service Processor	Integrated	

**Table 4: Client Configuration**

The client systems receive transactions via Oracle iPlanet Web Server which communicates with Tuxedo CFS-R Tier 1 with connections into the Oracle Database 11g Release 2.

There are two other networks configured on the system that do not participate in atomic transactions. The first is a service processor network which all systems are members of. This is used for powering on and off and system console access. The second network is the System Under Test (SUT) network between each of the X4800 M2 Server, X4270 M2s, X4170 M2s, SAN switch and 10GbE switch for system administration and other maintenance related activities.

**Substitutions**

Figure 1 shows the measured configuration and Figure 2 the full priced configuration. The DATA and REDO COMSTAR descriptions above also detail the differences for the priced and measured configurations. 9 of the 10 external storage controllers for the DATA storage environment will be substituted in the priced configuration, along with 1 of the 2 external storage controllers used in the REDO environment. The controllers being replaced are the Sun Fire X4275 Servers by the Sun Fire X4270 M2 Servers in the DATA and REDO environments. The substitution is done per Clause 2.3.4.5 of the Pricing specification using the following information:

		<i>DATA COMSTAR heads</i>			<i>REDO COMSTAR heads</i>			
	Qty	X4275	Qty	X4270 M2	Qty	X4275	Qty	X4270 M2
CPU	1	E5540	1	X5675	1	E5540	1	X5690
Type								
GHz		2.53		3.06		2.53		3.47
# Cores		4		6		4		6
# Threads		8		12		8		12
L1 cache	4	64KB	6	64KB	4	64KB	6	64KB
L2 cache	4	256KB	6	256KB	4	256KB	6	256KB
L3 cache		8MB		12MB		8MB		12MB
Memory								
Size	3	2GB	2	4GB	3	2GB	3	4GB
MHz		1066		1333		1066		1333
Type		DDR3		DDR3		DDR3		DDR3
<b>Adapters</b>								
Bus Type	1	6Gb/s SAS HBA, internal	1	6Gb/s SAS RAID, internal	1	6Gb/s SAS RAID internal	1	6Gb/s SAS RAID internal

	1	8Gb/s FC-AL	1	8Gb/s FC-AL	1	8Gb/s FC-AL	1	8Gb/s FC-AL
	4	6Gb/s SAS External	4	6Gb/s SAS External				

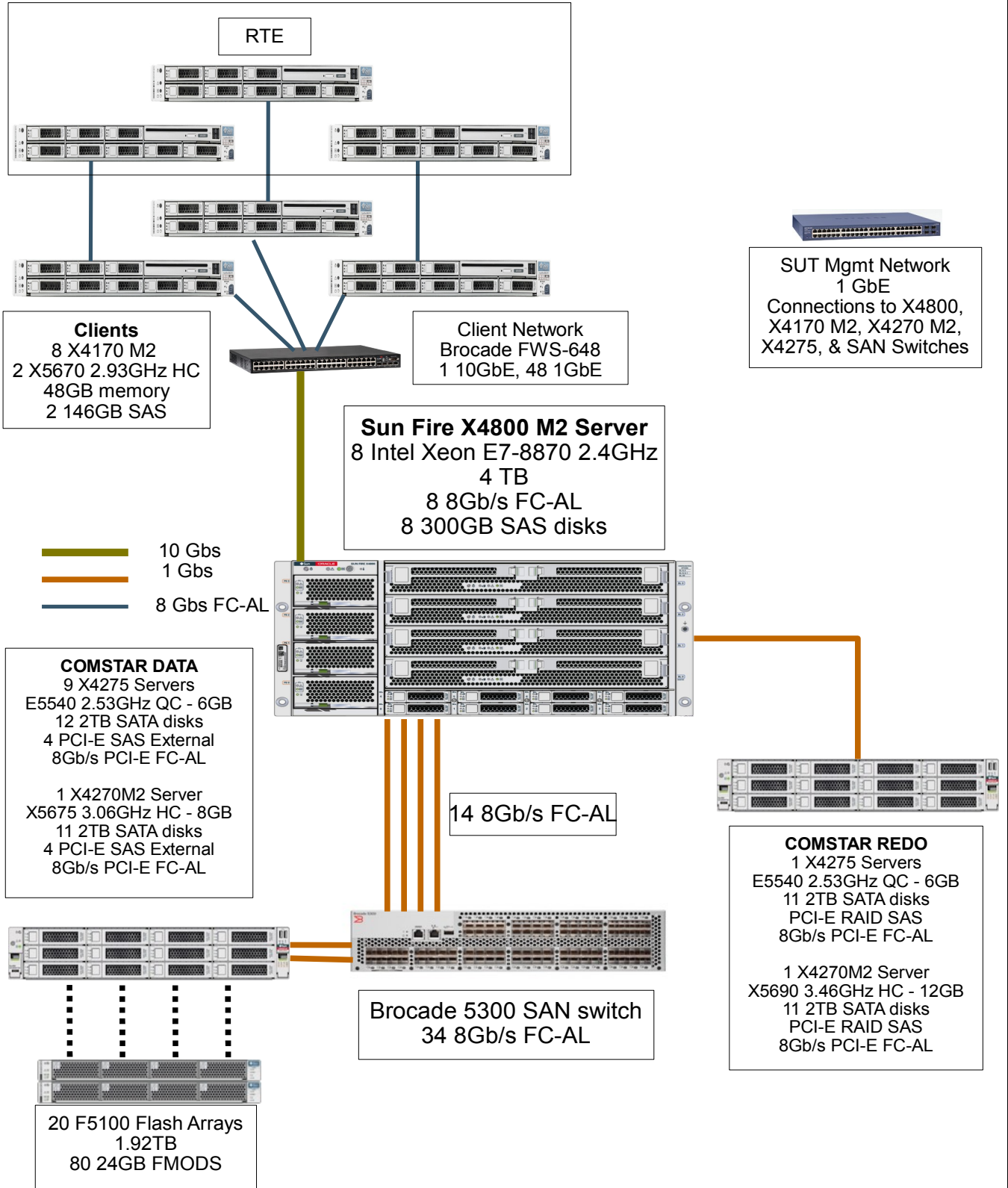
**Table 5: COMSTAR External Controller Substitution**

In addition, the processors for the X4170 M2 Servers used in the measured configuration as Clients are being substituted for in the priced configuration, along with the 146GB disk used for the operating system. The table below provides the details.

<b>Client Substitution</b>				
CPU		Intel Xeon X5670		Intel Xeon X5675
GHz		2.93		3.06
# Cores		6		6
# Threads		12		12
L1 Cache	6	64KB	6	64KB
L2 Cache	6	256KB	6	256KB
L3 Cache		12MB		12MB
Disks	3	146 GB SAS 10K RPM 2.5" SAS	2	300GB 10K RPM 2.5" SAS
Size	12	4GB	12	4GB
MHz		1333		1333
Type		DDR3		DDR3

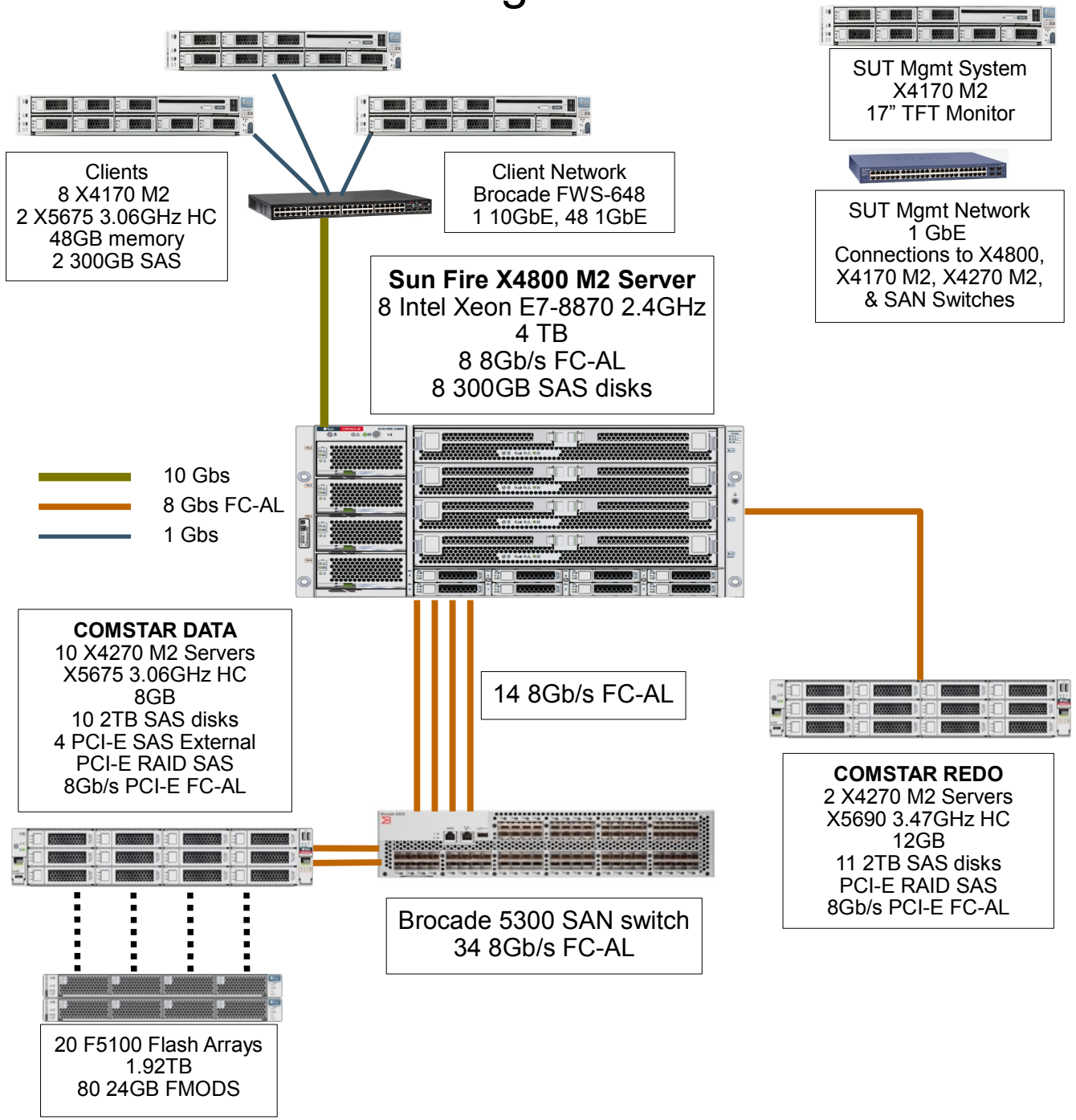


**Fig 1: Measured Configuration**



**Figure 2: Priced Configuration**

# Priced Configuration



— 10 Gbs  
— 8 Gbs FC-AL  
— 1 Gbs

**COMSTAR DATA**  
 10 X4270 M2 Servers  
 X5675 3.06GHz HC  
 8GB  
 10 2TB SAS disks  
 4 PCI-E SAS External  
 PCI-E RAID SAS  
 8Gb/s PCI-E FC-AL

14 8Gb/s FC-AL  
**Brocade 5300 SAN switch**  
 34 8Gb/s FC-AL

20 F5100 Flash Arrays  
 1.92TB  
 80 24GB FMODS

**SUT Mgmt System**  
 X4170 M2  
 17" TFT Monitor

**SUT Mgmt Network**  
 1 GbE  
 Connections to X4800,  
 X4170 M2, X4270 M2,  
 & SAN Switches

**COMSTAR REDO**  
 2 X4270 M2 Servers  
 X5690 3.47GHz HC  
 12GB  
 11 2TB SAS disks  
 PCI-E RAID SAS  
 8Gb/s PCI-E FC-AL

# 1 Clause 1: Logical Database Design Related Items

## 1.1 Table Definitions

*Listing must be provided for all table definition statements and all other statements used to set up the database.*

Appendix B describes the programs that define, create, and populate the Oracle database used for this testing.

## 1.2 Physical Organization of Database

*The physical organization of tables and indices, within the database, must be disclosed.*

Please see the section 0.4 Configuration Details for a detailed description of the SUT environment. Each DATA COMSTAR head has 2 F5100s for a total of 160 24GB Flash Modules (FMODS). Each FMOD is loaded with a VTOC to create four slices. Using SVM, a partition is created across all of the FMODs on a DATA head for each of the slices. Soft partitions are created on each of the four partitions also using SVM. The soft partitions are exported as raw devices through the Brocade 5300 switch to the X4800 M2 Server. These raw devices seen by the database server are then evenly “sliced” and the slices used by Oracle Database 11g Release 2 to create the tablespaces on. The full details of the scripts used to create the slices and partitions are in Appendix B.

## 1.3 Insert and Delete Operations

*It must be ascertained that insert and/or delete operations to any of the tables can occur concurrently with the TPC-C transaction mix. Furthermore, any restrictions in the SUT database implementation that precludes inserts beyond the limits defined in Clause 1.4.11 must be disclosed. This includes the maximum number of rows that can be inserted and the maximum key value for these new rows.*

All insert and delete functions were verified and fully operational during the entire benchmark.

## 1.4 Partitioning

*While there are a few restrictions placed upon horizontal or vertical partitioning of tables and rows in the TPC-C benchmark (see Clause 1.6), any such partitioning must be disclosed.*

The table History and iordr2 index, are partitioned horizontally. The details of the partitioning are disclosed in Appendix B for the database build environment.

## **2 Clause 2: Transaction And Terminal Profiles Related Items**

### **2.1 Random Number Generation**

*The method of verification for the random number generation must be described.*

The Random Number Generator used was SysVr4 nrand48() and erand48() UNIX calls.

### **2.2 Input/Output Screen Layouts**

*The actual layout of the terminal input/output screens must be disclosed.*

All screen layouts followed the specification exactly. The HTML source code used to generate the screens is available in Appendix A.

### **2.3 Terminal Feature Verification**

*The method used to verify that the emulated terminals provide all the features described in Clause 2.2.2.4 must be explained.*

The terminal attributes were verified by the auditor manually exercising each specification during the onsite audit portion of this benchmark.

### **2.4 Presentation Manager or Intelligent Terminal**

*Any usage of presentation managers or intelligent terminals must be explained.*

Application code running on the client machines implemented the TPC-C user interface. No presentation manager software or intelligent terminal features were used. The data is passed to the terminals using the HTML format, which can be displayed with any standard Web browser. The application code for the HTML display generation is listed in Appendix A.

### **2.5 Percentage of Home and Remote Order-lines**

*The percentage of home and remote order-lines in the New-Order transactions must be disclosed.*

The Transaction Input Percentages and Mix Table contains the percentage of home and remote order-lines for all of the New-Order transactions completed during the measurement interval.

### **2.6 Percentage of Rolled Back New-Orders**

*The percentage of New-Order transactions that were rolled back as a result of an unused item number must be disclosed.*

The Transaction Input Percentages and Mix table contains the percentage of New-Order transactions that were rolled back due to an illegal item being entered.

### **2.7 Items per New-Order**

*The number of items per orders entered by New-Order transactions must be disclosed.*

The Transaction Input Percentages and Mix table contains the average number of items ordered for each New-Order transaction.

### **2.8 Percentage of Home and Remote Payments**

*The percentage of home and remote Payment transactions must be disclosed.*

The Transaction Input Percentages and Mix table contains the percentage of home and remote warehouse transactions that occurred during the measurement interval for the Payment transactions.

### **2.9 Percentage of Access by Last Name**

*The percentage of Payment and Order-Status transactions that used non-primary key (C\_LAST) access to the database must be disclosed.*

The Transaction Input Percentages and Mix table contains the percentage of Payment and Order-Status transactions that were accessed by C\_LAST (non-primary key access) during the measurement interval.

## 2.10 Percentage of Skipped Deliveries

The percentage of Delivery transactions that were skipped as a result of an insufficient number of rows in the NEW-ORDER table must be disclosed.

The Transaction Input Percentages and Mix table contains the percentage of Delivery transactions which were “skipped” due to insufficient number of rows in the NEW-ORDER table.

## 2.11 Transaction Mix

The mix (i.e., percentages) of transaction types seen by the SUT must be disclosed.

The Transaction Input Percentages and Mix table contains the mix of each transaction type executed by the SUT.

New-Order	Sun Fire X4800 M2 Server
Percentage of Home order-lines	99.00%
Percentage of Remote order-lines	1.00%
Percentage of Rolled Back Transactions	1.00%
Avg. Number of Items per Transactions	10.00
Payment	
Percentage of Home Transactions	85.00%
Percentage of Remote Transactions	15.00%
Access by C_LAST (Non-primary key)	
Percentage of Payment Transactions	60.00%
Percentage of Order-Status Transactions	59.99%
Delivery	
Percentage of Deliveries skipped	0.00%
Transaction Mix	
New-Order	44.96%
Payment	43.01%
Order-Status	4.01%
Delivery	4.01%
Stock-Level	4.01%

**Table 6: Transaction Input Percentages and Mix**

## 2.12 Queuing Mechanism

The queuing mechanism used to defer the execution of the Delivery transaction must be disclosed.

Delivery transactions were submitted to servers using the same mechanism that other transactions used, Tuxedo API. The only difference was that tpcall() was used instead of tpcall() to call the server process asynchronously, i.e. control would return to the client thread immediately and the deferred delivery part would complete asynchronously in the server process.

## 3 Clause 3 Transaction and System Properties Related Items

### 3.1 Transaction System Properties (ACID)

*The results of the ACID tests must be disclosed along with a description of how the ACID requirements were met. This includes disclosing which case was followed for the execution of Isolation Test 7.*

The TPC Benchmark C Standard Specification defines a set of transaction processing system properties that a system under test (SUT) must support during the execution of the benchmark. Those properties are Atomicity, Consistency, Isolation, and Durability (ACID).

This section defines each of these properties, describes the steps taken to ensure that they were present during the test and describes a series of tests done to demonstrate compliance with the standard.

### 3.2 Atomicity

*The system under test must guarantee that transactions are atomic; the system will either perform all individual operations on the data, or will assure that no partially completed operations leave any effects on the data.*

#### 3.2.1 Completed Transaction

*Perform the Payment transaction for a randomly selected warehouse, district, and customer (by customer number as specified in Clause 2.5.1.2) and verify that the records in the CUSTOMER, DISTRICT, and WAREHOUSE tables have been changed appropriately.*

A row was randomly selected from the warehouse, district and customer tables and the balances noted. A payment transaction was started with the same warehouse, district and customer identifiers and a known amount. The payment transaction was committed and the rows were verified to contain correctly updated balances.

#### 3.2.2 Aborted Transaction

*Perform the Payment transaction for a randomly selected warehouse, district, and customer (by customer number as specified in Clause 2.5.1.2) and substitute a ROLLBACK of the transaction for the COMMIT of the transaction. Verify that the records in the CUSTOMER, DISTRICT, and WAREHOUSE tables have NOT been changed.*

A row was randomly selected from the warehouse, district and customer tables and the balances noted. A payment transaction was started with the same warehouse, district and customer identifiers and a known amount. The payment transaction was rolled back and the rows were verified to contain the original balances.

#### 3.2.3 Consistency

*Consistency is the property of the application that requires any execution of a database transaction to take the database from one consistent state to another, assuming that the database is initially in a consistent state.*

The benchmark specification requires explicit demonstration of the following four consistency conditions:

1. The sum of the district year-to-date (d\_ytd) balance for all districts within a warehouse is equal to the balance of the warehouse year-to-date (w\_ytd) for the district's warehouse;
2. For each district, the next order id (d\_next\_o\_id) minus one, is equal to the most recent order id (max(o\_id)) from the ORDER table for the associated district and warehouse. Also, the (d\_next\_o\_id) minus one is equal to the max(no\_o\_id) (most recent new-order) from the NEW-ORDER table;
3. For each district, the maximum order id minus minimum order id in the ORDER table plus one equals the number of rows in the NEW-ORDER table for that district;
4. For each district, the sum of the order line counts in the ORDER table equals the number of rows in the ORDER-LINE table for that district;

These consistency conditions were tested using a shell script to issue queries to the database. The results of the queries verified that the database was consistent for all four tests. The same four consistency tests were performed after the reported performance run during the audit.

### 3.3 Isolation Tests

The TPC Benchmark C Standard defines nine tests that are to be performed to demonstrate that required level of transaction isolation are met.

These tests are performed on the Sun Fire X4800 M2 Server configured for 400,000 warehouses. Each of the nine isolation tests were executed. All tests have been verified to demonstrate the desired transaction isolation level.

### 3.3.1 Isolation Test 1.

*This test demonstrates isolation for read-write conflicts of Order-Status and New-Order transactions when the New-Order transaction is committed.*

The test proceeds as follows:

1. An Order-Status transaction T0 was executed and committed for a randomly selected customer, and the order returned was noted.
2. A New-Order transaction T1 was started for the same customer used in T0. T1 was stopped prior to COMMIT.
3. An Order-Status transaction T2 was started for the same customer used in T1. T2 completed and was committed without being blocked by T1. T2 returned the same order that T0 had returned.
4. T1 was allowed to complete and was committed.
5. An Order-Status transaction T3 was started for the same customer used in T1. T3 returned the order inserted by T1.

### 3.3.2 Isolation Test 2

*This test demonstrates isolation for read-write conflicts of Order-Status and New-Order transactions when the New-Order transaction is rolled back.*

The test proceeds as follows:

1. An Order-Status transaction T0 was executed and committed for a randomly selected customer and the order returned was noted.
2. A New-Order transaction T1 with an invalid item number was started for the same customer used in T0. T1 was stopped immediately prior to ROLLBACK.
3. An Order-Status transaction T2 was started for the same customer used in T1. T2 completed and was committed without being blocked by T1. T2 returned the same order that T0 had returned.
4. T1 was allowed to ROLLBACK.
5. An Order-Status transaction T3 was started for the same customer used in T1. T3 returned the same order that T0 had returned.

### 3.3.3 Isolation Test 3

*This test demonstrates isolation for write-write conflicts of two New-Order transactions when both transactions are committed.*

The test proceeds as follows:

1. The D\_NEXT\_O\_ID of a randomly selected district was retrieved.
2. A New-Order transaction T1 was started for a randomly selected customer within the district used in step 1. T1 was stopped immediately prior to COMMIT.
3. Another New-Order transaction T2 was started for the same customer used in T1. T2 waited.
4. T1 was allowed to complete. T2 completed and was committed.
5. The order number returned by T1 was the same as the D\_NEXT\_O\_ID retrieved in step 1. The order number returned by T2 was one greater than the order number returned by T1.
6. The D\_NEXT\_O\_ID of the same district was retrieved again. It had been incremented by two (i.e. it was one greater than the order number returned by T2).

### 3.3.4 Isolation Test 4

*This test demonstrates isolation for write-write conflicts of two New-Order transactions when one transaction is rolled back.*

The test proceeds as follows:

1. The D\_NEXT\_O\_ID of a randomly selected district was retrieved.
2. A New-Order transaction T1, with an invalid item number, was started for a randomly selected customer within the district used in step 1. T1 was stopped immediately prior to ROLLBACK.
3. Another New-Order transaction T2 was started for the same customer used in T1. T2 waited.
4. T1 was allowed to roll back, and T2 completed and was committed.
5. The order number returned by T2 was the same as the D\_NEXT\_O\_ID retrieved in step 1.
6. The D\_NEXT\_O\_ID of the same district was retrieved again. It had been incremented by one (i.e. one greater than

the order number returned by T2).

### 3.3.5 Isolation Test 5

*This test demonstrates isolation for write-write conflicts of Payment and Delivery transactions when Delivery transaction is committed.*

The test proceeds as follows:

1. A query was executed to find out the customer who is to be updated by the next delivery transaction for a randomly selected warehouse and district.
2. The C\_BALANCE of the customer found in step 1 was retrieved.
3. A Delivery transaction T1 was started for the same warehouse used in step 1. T1 was stopped immediately prior to COMMIT.
4. A Payment transaction T2 was started for the same customer found in step 1. T2 waited.
5. T1 was allowed to complete. T2 completed and was committed.
6. The C\_BALANCE of the customer found in step 1 was retrieved again. The C\_BALANCE reflected the results of both T1 and T2.

### 3.3.6 Isolation Test 6

*This test demonstrates isolation for write-write conflicts of Payment and Delivery transactions when the Delivery transaction is rolled back.*

The test proceeds as follows:

1. A query was executed to find out the customer who is to be updated by the next delivery transaction for a randomly selected warehouse and district.
2. The C\_BALANCE of the customer found in step 1 was retrieved.
3. A Delivery transaction T1 was started for the same warehouse used in step 1. T1 was stopped immediately prior to ROLLBACK.
4. A Payment transaction T2 was started for the same customer found in step 1. T2 waited.
5. T1 was allowed to ROLLBACK. T2 completed and was committed. The C\_BALANCE of the customer found in step 1 was retrieved again. The C\_BALANCE reflected the results of only T2.

### 3.3.7 Isolation Test 7

*This test demonstrates repeatable reads for the New-Order transaction while an interactive transaction updates the prices of some items.*

The test proceeds as follows:

1. The I\_PRICE of two randomly selected items X and Y were retrieved.
2. A New-Order transaction T1 with a group of items including items X and Y was started. T1 was stopped immediately after retrieving the prices of all items. The prices of items X and Y retrieved matched those retrieved in step 1.
3. A transaction T2 was started to increase the price of items X and Y by 10%.
4. T2 did not stall and was committed.
5. T1 was resumed, and the prices of all items were retrieved again within T1. The prices of items X and Y matched those retrieved in step 1.
6. T1 was committed.
7. The prices of items X and Y were retrieved again. The values matched the values set by T2.

Execution followed *Case D* of *Clause 3.4.2.7*.

### 3.3.8 Isolation Test 8

*This test demonstrates isolation for phantom protection between New-Order and Order-Status transactions.*

The test proceeds as follows:

1. An Order-Status transaction T1 was started for a randomly selected customer.
2. T1 was stopped immediately after reading the order table for the selected customer to find the most recent order for that customer.
3. A New-Order transaction T2 was started for the same customer. T2 completed and was committed without being blocked by T1.



4. T1 was resumed and the ORDER table was read again to determine the most recent order for the same customer. The order found was the same as the one found in step 2.
5. T1 completed and was committed.

### 3.3.9 Isolation Test 9.

*This test demonstrates isolation for phantom protection between New-Order and Delivery transactions.*

The test proceeds as follows:

1. The NO\_D\_ID of all NEW\_ORDER rows for a randomly selected warehouse and district was changed to 11. The changes were committed.
2. A Delivery transaction T1 was started for the selected warehouse.
3. T1 was stopped immediately after reading the NEW\_ORDER table for the selected warehouse and district. No qualifying row was found.
4. A New-Order transaction T2 was started for the same warehouse and district. T2 completed and was committed without being blocked by T1.
5. T1 was resumed and the NEW\_ORDER table was read again. No qualifying row was found.
6. T1 completed and was committed.
7. The NO\_D\_ID of all NEW\_ORDER rows for the selected warehouse and district was restored to the original value. The changes were committed.

## 3.4 Durability

*The tested system must guarantee durability: the ability to preserve the effects of committed transactions and insure database consistency after recovery from any one of the failures listed in Clause 3.5.3.*

### 3.4.1 Loss of Memory, Instantaneous Interruption, Power Loss, and Loss of Log

This test was executed by following these steps:

1. The total number of orders was determined by the sum of D\_NEXT\_O\_ID from all rows in the district table; giving the beginning count.
2. The RTE was started with full user load.
3. The test was allowed to ramp up and ran in steady state for more than 5 minutes.
4. The test ran for a least one more minute.
5. The cable connecting one of the REDO COMSTAR heads to one of the database nodes is removed, thereby causing the loss of one-half of the mirrored log environment (loss of disk). Because the redo logs for the database node are mirrored across the REDO COMSTAR heads, the test continues to run without interruption.
6. The run continued for at least 5 minutes while maintaining steady state.
7. Power to the X4800 M2 Server was disconnected
8. The test was allowed to continue for at least one more minute.
9. The database was shutdown and restarted. Oracle automatic recovery was initiated and completed.
10. Step 1 was repeated, giving the ending count.
11. Consistency Test 3 was verified.
12. A sample from the success file was compared against the database.
13. The RTE report was used to determine the number of successful New-Order transactions. The difference between the counts in Steps 1 and 11 was compared with the RTE count to verify that no committed transactions were lost.

### 3.4.2 Loss of Durable Media Containing Database Tables

*This test can be executed on a small scaled database (at least 10%) under a reduced user load (at least 10%). The throughput is reaching at least 10% of the reported tpmC. When a reduced configuration is used, a target of 12% is recommended to avoid falling under the 10% threshold.*

This test was executed by following these steps:

1. The total number of orders was determined by the sum of D\_NEXT\_O\_ID from all rows in the district table; giving the beginning count.
2. The RTE was started with a reduced terminal load accounting for more than 10% of the reported throughput.
3. The test was allowed to ramp up and ran in steady state for more than 5 minutes.

4. A manual checkpoint was initiated and complete and the test was allowed to run for a least one more minute.
5. A cable between one DATA COMSTAR head and the Brocade 5300 SAN switch was disconnected. This causes the instantaneous loss of an exported LUN (disk) on the DATA head.
6. The test was allowed to run until a fault was detected and transactions start reporting errors.
7. The RTE was shutdown. The RTE report was generated.
8. The failed DATA COMSTAR head was restored.
9. The database tables were restored from backup, overwriting the data, except for the logs.
10. The database was started and the roll-forward recovery (Oracle instance recovery) was executed.
11. Step 1 was repeated, giving the ending count.
12. Consistency Test 3 was verified.
13. A sample from the success file was compared against the database.
14. The RTE report was used to determine the number of successful New-Order transactions. The difference between the counts in Steps 1 and 12 was compared with the RTE count to verify that no committed transactions were lost.

## 4 Clause 4: Scaling and Database Population Related Items

### 4.1 Initial Cardinality of Tables

The cardinality (e.g. number of rows) of each table, as it existed at the start of the benchmark run (see Clause 4.2), must be disclosed. If the database was over-scaled and inactive rows of the WAREHOUSE table were deleted (see Clause 4.2.2) the cardinality of the WAREHOUSE table as initially configured and the number of rows deleted must be disclosed.

This database was built with 400,000 warehouses. The following tables shows the initial cardinality of the tables after table population, and the cardinality prior to the measurement run.

<i>Table</i>	<i>Initial Row Count</i>	<i>Row Count Prior to Measured Run</i>
Warehouse	400,000	400,000
District	4,000,000	4,000,000
Customer	12,000,000,000	12,000,000,000
History	12,000,000,000	12,546,577,880
Orders	12,000,000,000	12,565,696,716
New order	3,600,000,000	3,656,069,076
Order line	120,003,410,560	125,660,385,630
Stock	40,000,000,000	40,000,000,000
Item	100,000	100,000

*Table 7: Table Cardinality*

### 4.2 Database Layout

The distribution of tables and logs across all media must be explicitly depicted for the tested and priced systems.

Refer to Section 0.4 Configuration Details for a description of the physical layout of the SUT. The Sun Fire X4800 M2 Server database logs are mirrored across the two REDO COMSTAR heads.

The database tables are distributed across all of the DATA COMSTAR heads. The mapping of the database tables to the DATA heads are described in Mapping of Database below. The physical setup of the DATA heads are described in detail in Section 0.4 DATA COMSTAR Heads Description.

### 4.3 Type of Database

A statement must be provided that describes:

1. The data model implemented by the DBMS used (e.g., relational, network hierarchical).
2. The database interface (e.g., embedded, call level) and access language (e.g., SQL, DL/I, COBOL read/write) used to implement the TPC-C transactions. If more than one interface/access language is used to implement TPC-C, each interface/access language must be described and a list of which interface/access language is used with which transaction type must be disclosed.

Oracle Database 11g Release 2 Enterprise Edition is a relational database management system. SQL stored procedures were invoked via the Oracle Call Interface (OCI). The application code appears in Appendix A.

### 4.4 Mapping of Database

The mapping of database partitions/replications must be explicitly described.

All of the database tables are partitioned across the DATA heads. Each DATA head has 160 24GB SSD devices. A VTOC is loaded onto each SSD which creates 4 separate partitions. Oracle Solaris Volume Manager creates a RAID 0 stripe across all 160 devices for each partition. Soft partitions are created within each of the 4 partitions. These soft partitions are exported to the SAN and are visible to the X4800 M2 Server. The following table shows the distribution of the soft

partitions for each DATA head to the database table.

<i>Data Node</i>	<i>cust</i>	<i>hist</i>	<i>icust1</i>	<i>icust2</i>	<i>iordr2</i>	<i>istok</i>	<i>misc</i>	<i>nord</i>	<i>ordr / ordl</i>	<i>stok</i>	<i>system</i>	<i>undo</i>
1	56	8	4	4	8	4	4	4	20	76	1	0
2	56	8	4	4	8	4	4	4	20	76	1	0
3	56	8	4	4	8	4	4	4	20	76	1	0
4	56	8	4	4	8	4	4	4	20	76	0	0
5	56	8	4	4	8	4	4	4	20	76	1	0
6	56	8	4	4	8	4	4	4	20	76	0	1
7	56	8	4	4	8	4	4	4	20	76	0	0
8	56	8	4	4	8	4	4	4	20	76	0	0
9	56	8	4	4	8	4	4	4	20	76	0	0
10	56	8	4	4	8	4	4	4	20	76	0	0
<b>Total</b>	<b>560</b>	<b>80</b>	<b>40</b>	<b>40</b>	<b>80</b>	<b>40</b>	<b>40</b>	<b>40</b>	<b>200</b>	<b>760</b>	<b>4</b>	<b>1</b>

**Figure 1: Database Mapping**

#### 4.5 60 Day Space Computation

Details of the 60 day space computations along with proof that the database is configured to sustain 8 hours of growth for the dynamic tables (Order, Order-Line, and History) must be disclosed (see Clause 4.2.3).

<b>Warehouses</b>		<b>400,000</b>			
<b>tpmC</b>		<b>5,055,888</b>			
		(MB)	(MB)	(MB)	(MB)
<b>Table</b>	<b>Rows</b>	<b>Table</b>	<b>Index</b>	<b>5% Growth</b>	<b>Total Space</b>
Warehouse	400,000	1,570	10	79	1,695
District	4,000,000	15,650	80	787	16,517
Item	100,000	20	10	2	32
Stock	40,000,000,000	11,174,551	786,750	598,065	12,559,366
Customer	12,000,000,000	7,822,290	840,000	433,114	9,095,404
New-Order	3,600,000,000	94,705	1,028	4,787	100,520
Orders	12,000,000,000	482,878	407,933	180,154	1,070,964
Order-Line	120,003,410,560	9,174,678	40,520	1,863,640	11,078,838
History	120,000,000,000	695,604	0	140,676	836,280
Overhead		11,296			11,296
Free Space	742,469				
Dynamic Space	12,986,082				
Static Space	21,784,793				
Daily Growth	2,626,247				
Daily Spread	0				
<b>60 Days Storage Requirements</b>					
60 Days (MB)	179,359,618				
60 Days (GB)	175,156				
<b>8 Hours Log Storage</b>					
<b>Log Requirements</b>					
Average log switch Interval (seconds)	1,621				
Log size with mirroring (GB)	1,436				

Log switch in 8 Hrs	18
8HR Log Req (GB)	25,505

<b>Storage</b>	<b>Measured</b>		<b>Priced</b>		
<b>Disk Type</b>	<b>Formatted Capacity(GB)</b>	<b># of Disks</b>	<b>Capacity (GB)</b>	<b># of Disks</b>	<b>Capacity (GB)</b>
F5100 1.92TB	22.88	1,600	36,608	1,600	36,608
2TB SAS	1,820	120	218,400	120	218,400
2TB SAS (Log RAID 10)	1,820	20	36,400	20	36,400
<b>Total Capacity</b>					<b>291,608</b>

*Figure 2: 60 Day Space Calculations*

## 5 Clause 5: Performance Metrics and Response Time Related Items

### 5.1 Measured tpmC

*Measured tpmC must be reported.*

The measured tpmC was 5,055,888

### 5.2 Response Times

*Ninetieth percentile, maximum and average response times must reported for all transaction types as well as for the menu response time.*

Please refer the Numerical Quantities Section of the Executive Summary of this report for the Response Time values.

### 5.3 Keying and Think Times

*The minimum, the average, and the maximum keying and think times must be reported for all transaction types.*

Please refer the Numerical Quantities Section of the Executive Summary of this report for the Keying and Think Time values.

### 5.4 Response Time Frequency Distribution Curves

*Response Time frequency distribution curves (see Clause 5.6.1) must be reported for each transaction type.*

*Figure 3: New-Order Response Time Distribution*

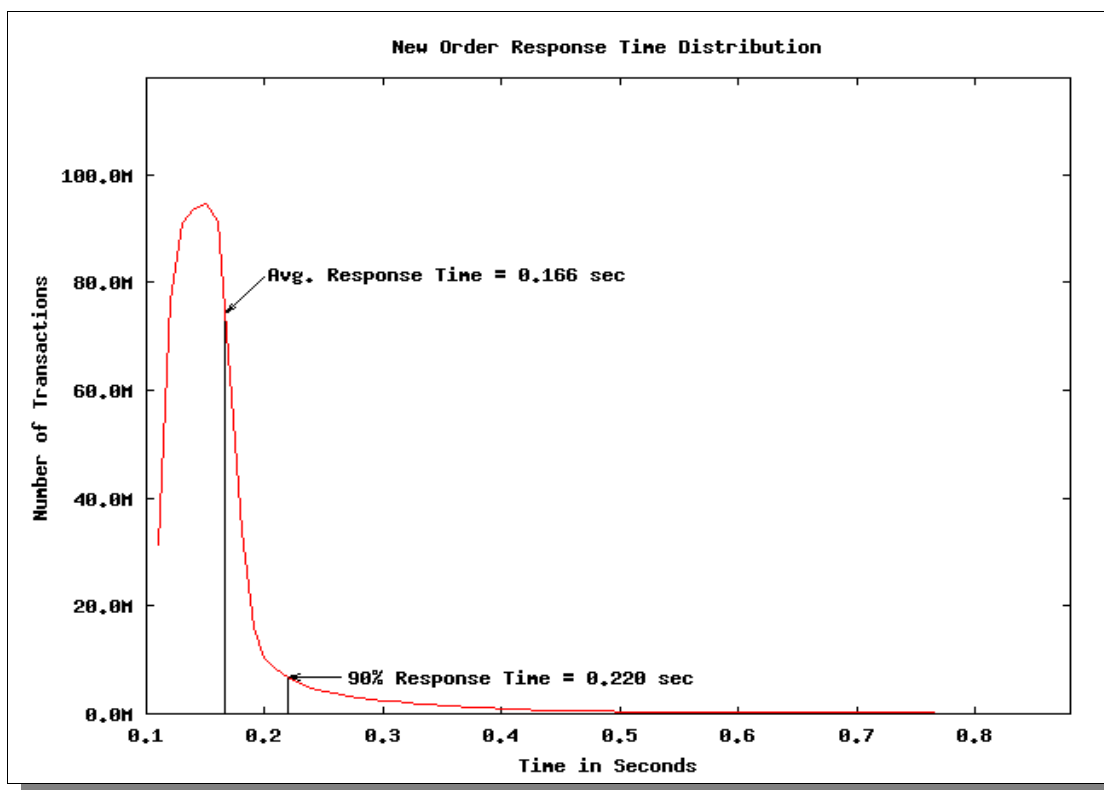


Figure 3: Payment Response Time Distribution

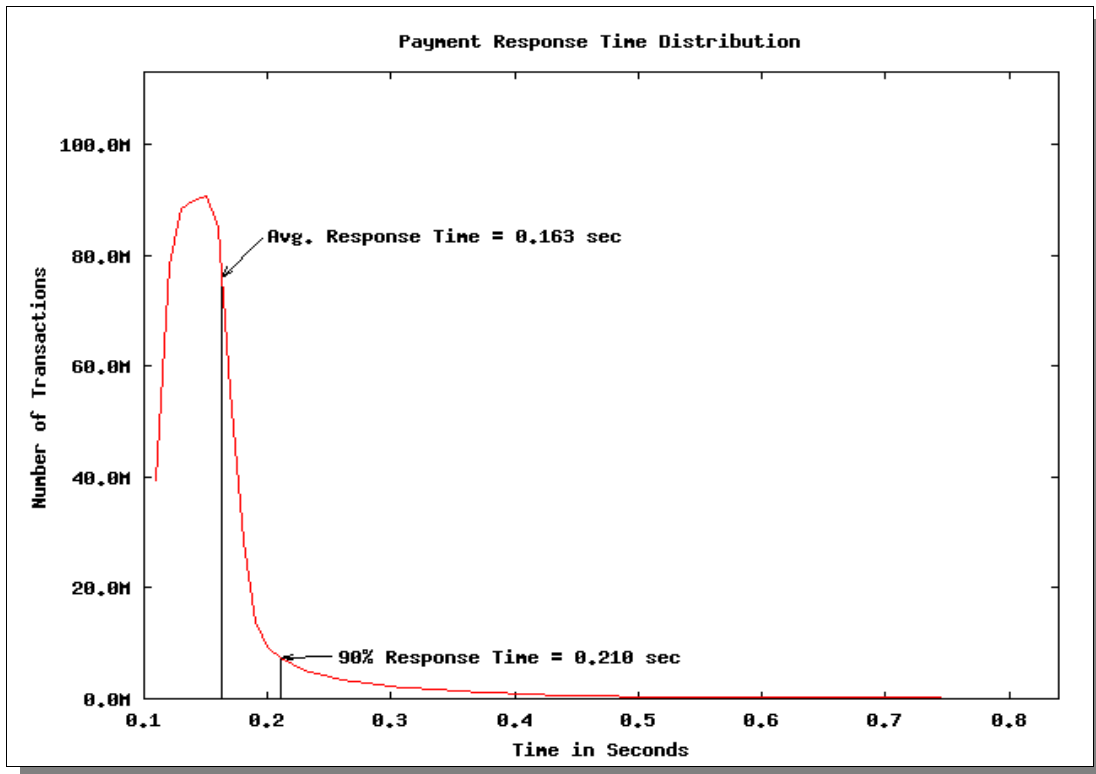


Figure 4: Order-Status Response Time Distribution

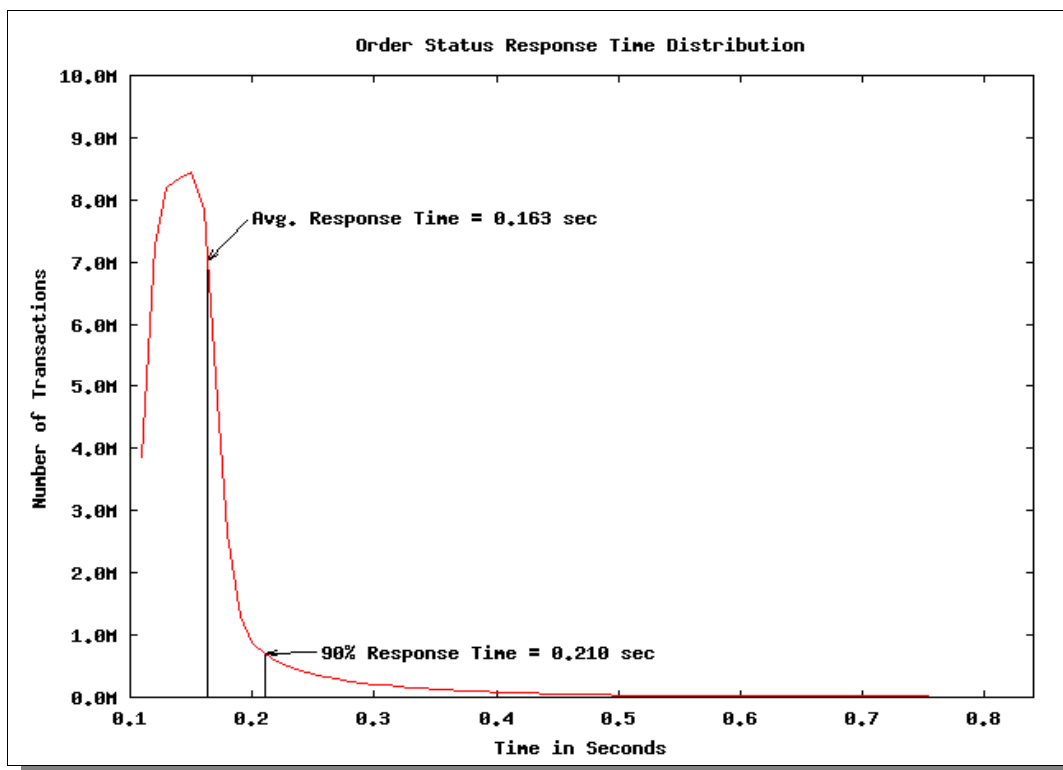


Figure 5: Delivery (Interactive) Response Time Distribution

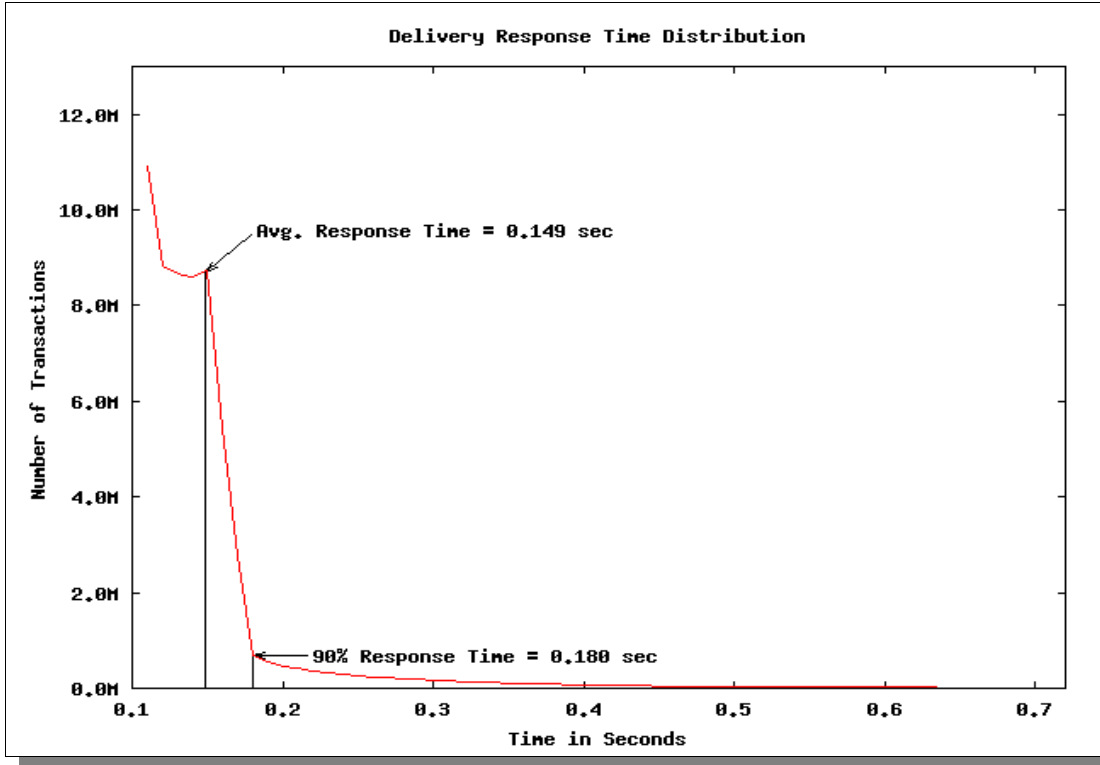
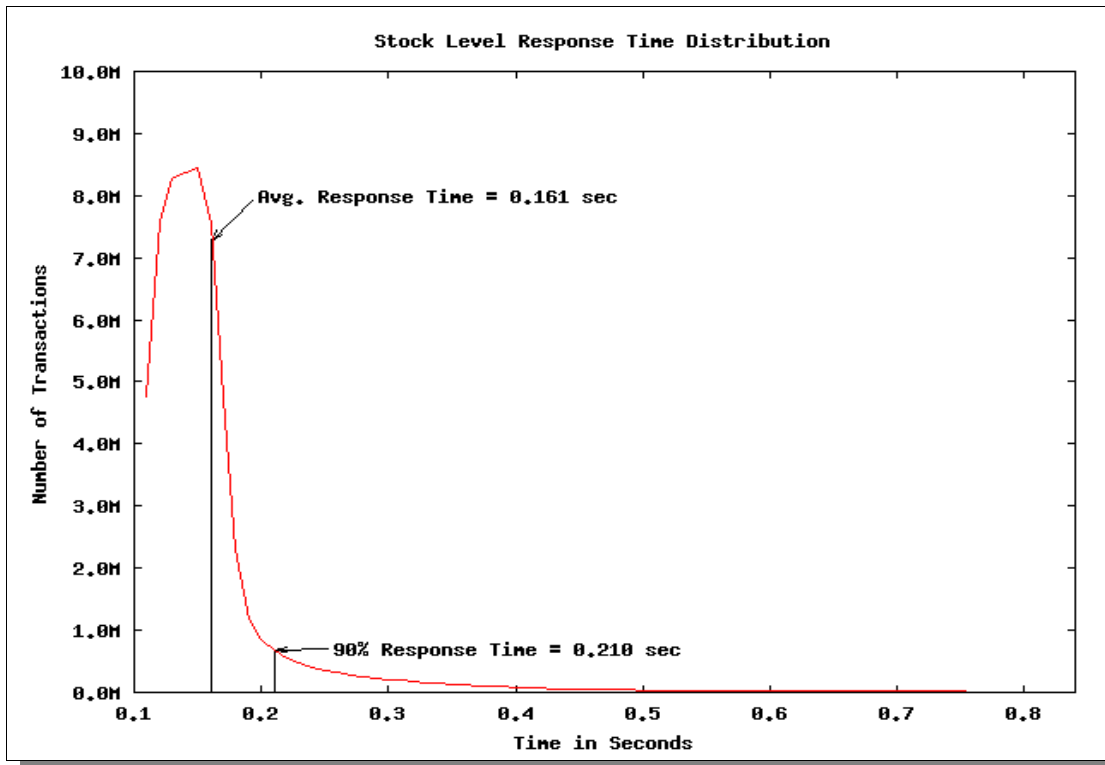


Figure 6: Stock-Level Response Time Distribution





## 5.5 Think Time Frequency Distribution

Think Time frequency distribution curves (see Clause 5.6.3) must be reported for the New-Order transaction.

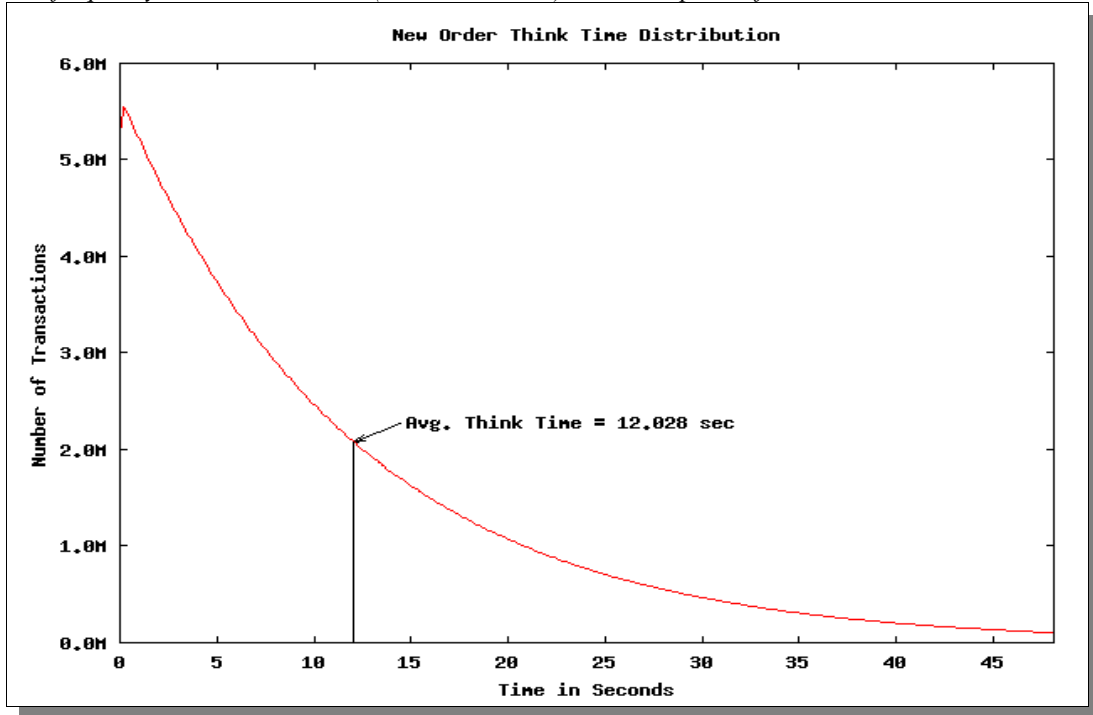


Figure 7: New-Order Think Time Distribution

## 5.6 Response Times versus Throughput

The performance curve for response times versus throughput (see Clause 5.6.2) must be reported for the New-Order transaction.

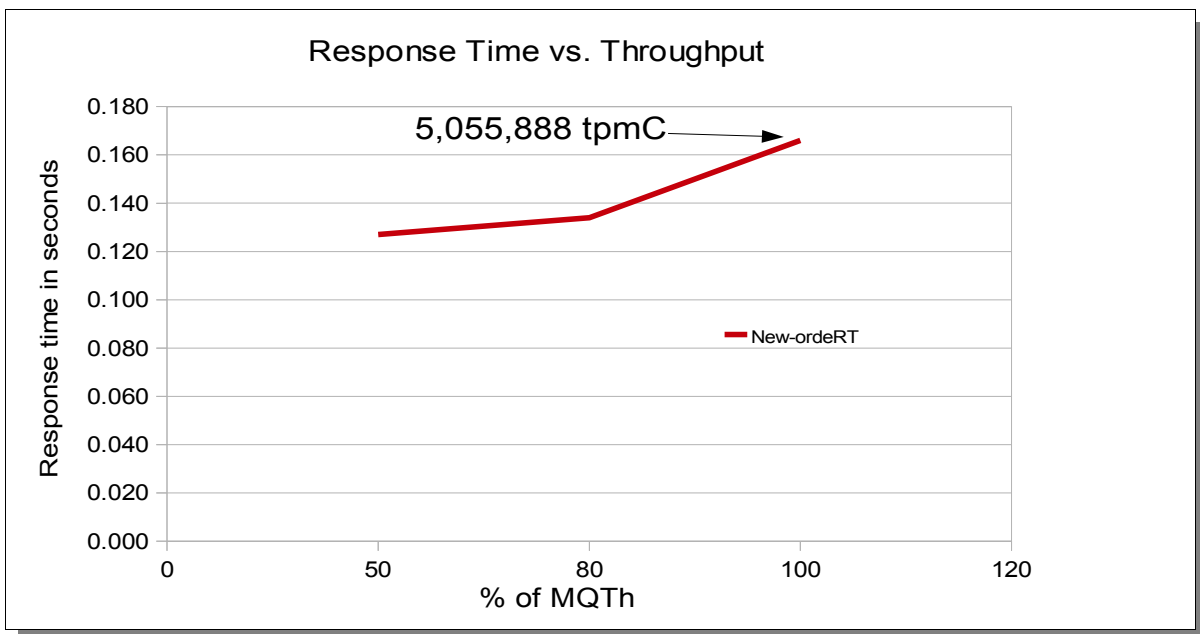


Figure 8: New-Order Response Time versus Throughput

## 5.7 Throughput versus Elapsed Time

A graph of throughput versus elapsed time (see Clause 5.6.4) must be reported for the New-Order transaction.

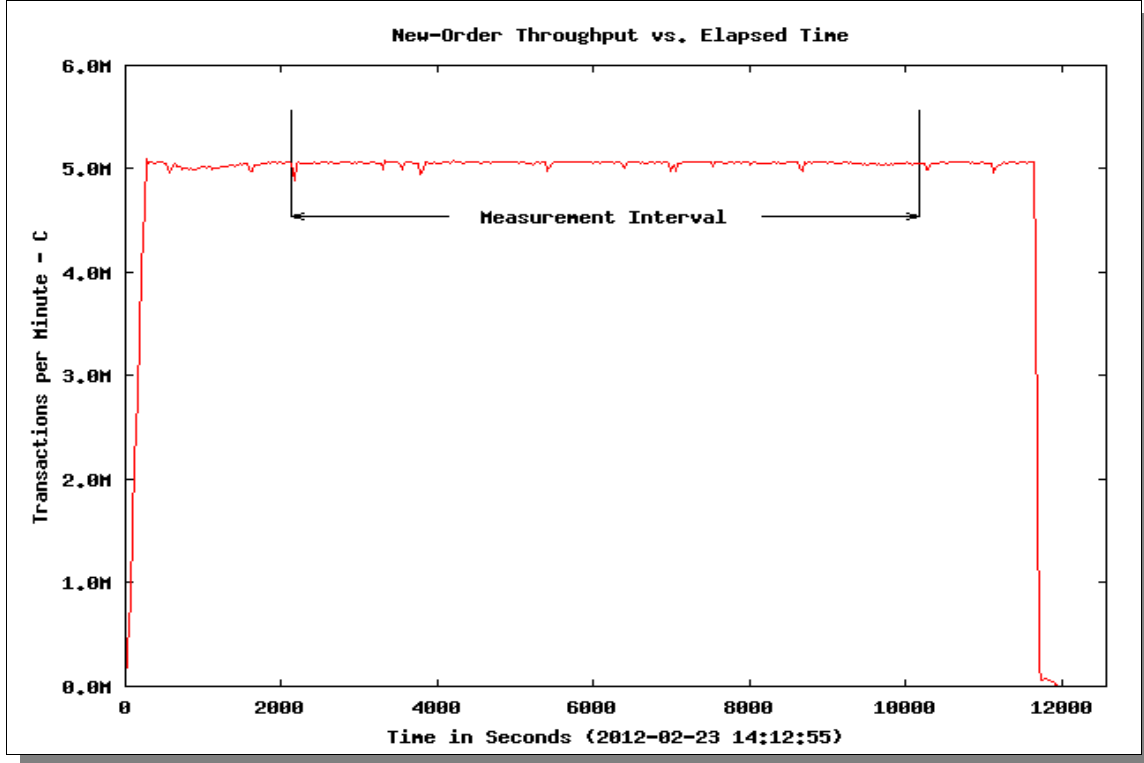


Figure 9: New-Order Throughput versus Time

## 5.8 Steady State Determination

The method used to determine that the SUT had reached a steady state prior to commencing the measurement interval (see Clause 5.5) must be described.

The throughput was verified by examining the throughput (tpmC) graph reported at 30 second intervals for the duration of the benchmark.

## 5.9 Work Performed During Steady State

A description of how the work normally performed during a sustained test (for example checkpointing, writing redo/undo log records, etc.), actually occurred during the measurement interval must be reported.

During the Test Run, emulated users submit TPC-C transactions according to the described mix, keying time and think times. The transactions are implemented in accordance with the requirements of the specification. An emulated user submits transaction input via HTTP and receives acknowledgment of the completed transaction. The response time is measured from the start of the transaction until the last byte is received by the RTE. Upon completion of a transaction, the RTE “thinks” for a randomly generated time period before selecting the next transaction. Upon selection of the next transaction to execute thru the Menu transaction, an emulated user delays for a period of time to simulate the rate of an individual inputting data. The transaction is submitted and the cycle continues until the Test Run completes.

During the execution of the transactions, Oracle maintains consistency of the database through the use of isolation properties that meet the requirements of the specification. Also, committed transactions are “logged” using Oracle's redo log functions. These logs ensure the system never loses any committed transactions. To ensure modified data pages are not left in memory for more than 30 minutes, Oracle implements an ongoing incremental checkpoint to flush modified pages to their respective tables on disk.

## 5.10 Measurement Period Duration

A statement of the duration of the measurement interval for the reported Maximum Qualified Throughput (tpmC) must be included.

The reported measured interval was 8,050 seconds (2 hours, 14 minutes, 10 seconds) long.

## 5.11 Transaction Mix Regulation

*The method of regulation of the transaction mix (e.g., card decks or weighted random distribution) must be described. If weighted distribution is used and the RTE adjusts the weights associated with each transaction type, the maximum adjustments to the weight from the initial value must be disclosed.*

A weighted distribution algorithm was used by the RTE to regulate the transaction mix. Weights for the various transactions were assigned before the measurement started and adjustments automatically made periodically during the run to maintain the desired mix.

## 5.12 Transaction Mix

*The percentage of the total mix for each transaction type must be disclosed.*

See Transaction Input Percentages and Mix table on page 29.

## 5.13 Percentage of New-Order Transactions

*The percentage of New-Order transactions rolled back as a result of invalid item number must be disclosed.*

See Transaction Input Percentages and Mix table on page 29.

## 5.14 Number of Order-lines per New-Order

*The average number of order-lines entered per New-Order transaction must be disclosed.*

See Transaction Input Percentages and Mix table on page 29.

## 5.15 Percentage of Remote Order-lines per New-Order

*The percentage of remote order-lines entered per New-Order transaction must be disclosed.*

See Transaction Input Percentages and Mix table on page 29.

## 5.16 Percentage of Remote Payments

*The percentage of remote payment transactions must be disclosed.*

See Transaction Input Percentages and Mix table on page 29.

## 5.17 Percentage of Non-Primary access by C\_LAST for Payment and Order-Status

*The percentage of customer selections by customer last name in the Payment and Order-Status transactions must be disclosed.*

See Transaction Input Percentages and Mix table on page 29.

## 5.18 Percentage of Skipped Delivery Transactions

*The percentage of Delivery transactions skipped due to there being fewer than necessary orders in the New-Order table must be disclosed.*

See Transaction Input Percentages and Mix table on page 29.

## 5.19 Checkpoints

*The number of checkpoints in the Measurement Interval, the time in seconds from the start of the Measurement Interval to the first checkpoint and the Checkpoint Interval must be disclosed.*

Oracle checkpoints ensure that modified data blocks are written to durable media. Oracle performs checkpoints in an incremental and continuous fashion in such a way that modified data blocks do not stay "dirty" in the shared memory longer than a specified duration.

Oracle controls checkpoints using one or combination of the following mechanisms:

- the redo log is filled to a specified capacity, causing a log switch
- the amount of data written to redo log reaches a value specified by `log_checkpoint_interval`
- the amount of the time since the last checkpoint reaches a value specified by `log_checkpoint_timeout`
- an explicit command
- maximum time to recover a crashed database instance, specified by `fast_mtrr_target`

The redo logs are sized to control the checkpoints. They are sized to guarantee that no data blocks are "dirty" for more than 1700 seconds in shared memory.

The following table contains information regarding checkpoints, both before and during the measurement interval.

<i>Checkpoints</i>				
Checkpoint #	Start Time	End Time	Duration	Comments
	14:17:31			Run starts
#0		14:19:28		
#1	14:22:09	14:46:39	0:24:30	
	14:48:25			Measurement starts
#2	14:49:21	15:13:37	0:24:16	
#3	15:16:19	15:40:31	0:24:12	
#4	15:43:13	16:07:25	0:24:12	
#5	16:10:07	16:34:29	0:24:22	
#6	16:37:11	17:01:31	0:24:20	
	17:02:35			Measurement ends
#7	17:04:14			
	17:26:16			Run Ends

The measurement interval begins at 14:48:25 and ends at 17:02:35. One checkpoint is completed before the measurement interval begins. There are five checkpoints completed within the measurement interval. The average checkpoint interval is 27 minutes, 04 seconds. A checkpoint during measurement interval lasts approximately 24 minutes and 22 seconds.

## 6 Clause 6: SUT, Driver and Communications Related Items

### 6.1 RTE Description

*If the RTE is commercially available, then its inputs must be specified. Otherwise, a description must be supplied of what inputs (e.g. scripts) to the RTE had been used.*

The RTE used was developed by Oracle and is proprietary. It consists of a master\_rte program which creates a thread for each RTE session and controls the run. After the run completes, a separate report generator program collects all the log files and generates the final statistics of a run.

Inputs to the RTE include the names of the RTE machines to run on, client machines to attach to, the database scale, the ramp-up, measurement and ramp-down times. The main inputs to the RTE are as follows:

<i>Input Type</i>	<i>Value</i>
Ramp-up Duration	2,100 seconds
Ramp-Down Duration	1421 second
Measurement Interval	8,050 seconds
Database Scale	400,000 warehouses
Total Users	4,000,000
Users/Driver	250,000
Number of RTEs	16
Client Ports	8080

**Table 8: RTE Parameter Input**

### 6.2 Lost Connections

*The number of terminal connections lost during the Measurement Interval must be disclosed (see Clause 6.6.2).*

The number of connections were counted when the RTE was first started and a count of the number of errors that occurred during the benchmark run were kept. There were no errors reported during the Measurement Interval, therefore no lost connections.

### 6.3 Emulated Components

*It must be demonstrated that the functionality and performance of the components being emulated in the Driver System are equivalent to that of the priced system. The results of the test described in Clause 6.6.3.4 must be disclosed.*

In the configuration, workstations are connected to the clients via HTTP in the same way as the emulated system. The driver system emulates the workstations by making a direct connection to the SUT for each terminal.

### 6.4 Configuration Diagrams

*A complete functional diagram of both the benchmark configuration and the configuration of the proposed (target) system must be disclosed. A detailed list of all software and hardware functionality being performed on the Driver System, and its interface to the SUT must be disclosed (see Clause 6.6.3.6).*

The Fig 1: Measured Configuration on page 25 details the configuration used during the measurement. The Figure 2: Priced Configuration on page 25 details the components that were priced. Refer to sections 6.1 and 6.3 above for a description of the emulated components of the Driver System.

### 6.5 Network Configuration

*The network configurations of both the tested services and the proposed (target) services which are being represented and a thorough explanation of exactly which parts of the proposed configuration are being replaced with the Driver System must be disclosed (see Clause 6.6.4).*

The measured configuration utilized 3 separate networks to administer, maintain, setup, and execute the TPC-C transactions. The first network provides administration and support services between the X4800 M2 Server, all of the COMSTAR heads, and the X4170M2 clients through a 1GbE network using 1 Netgear JGS548E 24 port switch. The

second is between the X4170 M2 clients and the X4800 M2 Server through the Brocade FastIron FWS648 switch. The third network provides system console and service processor control for all of the above machines over a 1GbE through the same Netgear switch as the administration network. The last network is a 1GbE ethernet between the X4170 M2 clients and the RTE. This RTE network provides the user and terminal emulation and therefore is not priced.

## **6.6 Operator Intervention**

*If the configuration requires operator intervention, the mechanism and the frequency of this intervention must be disclosed.*

The Sun Fire X4800 M2 Server configuration does not require any operator intervention to sustain the reported throughput for the 8 hour business day.

## 7 Clause 7: Pricing Related Items

### 7.1 Hardware and Software Component Pricing

A detailed list of hardware and software used in the priced system must be reported. Each separately orderable item must have vendor part number, description, release/revision level, and either general availability status or committed delivery date. If package-pricing is used, vendor part number of the package and a description uniquely identifying each of the components of the package must be disclosed. Pricing source(s) and effective date(s) of price(s) must also be reported.

A detailed price list is included as part of the Executive Summary included with this report. A third-party price quote from CDW is available in Appendix D.

### 7.2 Total Three Year Cost

The total 3-year price of the entire **Priced Configuration** must be reported, including: hardware, software, and maintenance charges. The justification of any **Discounts** applied must be disclosed in the price sheet. Sufficient detail of what items are being discounted and by how much they are being discounted must be provided so that the **Discount** amount used in the computation of the total system cost can be independently reproduced.

Details of the pricing for all components used in this measurement are included in the Executive Summary at the beginning of this document. Oracle's discounts are based on US list prices and for similar quantities and configurations. A total discount of 31.4% has been applied to all Oracle hardware, software, and services based on the total value and quantities of the components of the configuration, including full payment of all components and maintenance.

For assistance with any of these prices or their applicability to any customer's requirements please contact:

Mary Beth Pierantoni

mary.beth.pierantoni@oracle.com

### 7.3 Availability

The Committed delivery date for general availability (availability date) of products used in the price calculations must be reported. The Availability Date must be reported on the first page of the Executive Summary and with a precision of one day. When the priced system includes products with different availability dates, the reported availability date for the priced system must be the date at which all **Components** are committed to be **Generally Available**. Each **Component** used in the **Priced Configuration** is considered to be Available on the Availability Date unless an earlier date is specified.

All products will be available on June 26, 2012

### 7.4 Hardware and Software Support

Oracle's Premier Support for Systems consists of services in support of hardware systems, firmware, and software. For hardware systems this support is limited to Sun hardware only. For Oracle's Sun servers, this includes support for Oracle Linux with UEK Release 2 and Oracle Solaris 11. This support is for 24 hours a day, 7 days a week with a 4 hour response time.

Oracle Incident Server Support consists of software support for Oracle Database 11g Release 2, Partitioning and Tuxedo CFS-R Tier 1 with a 4 hour response time, 24 hours a day, 7 days a week.

### 7.5 Statement of tpmC, Price/Performance

A statement of the measured tpmC, as well as the respective calculations for 3-year pricing, price/performance (price/tpmC), and the availability date must be included.

<i>System</i>	<i>tpmC</i>	<i>3-year System Cost</i>	<i>\$/tpmC</i>	<i>Availability Date</i>
Sun Fire X4800 M2 Server with X4800 M2 Servers	5,055,888	\$4,483,729	\$0.89	June 26, 2012

Table 9: Statement of tpmC and Price/Performance

## 7.6 Country Specific Pricing

*Additional Clause 7 related items may be included in the Full Disclosure Report for each country specific priced configuration. Country specific pricing is subject to Clause 7.1.7*

The components for this configuration are priced using currency from the United States of America. All prices from Oracle are based upon US list prices.

## 7.7 Orderability Date

*For each of the components that are not orderable on the report date of the FDR, the following information must be included in the FDR:*

- *Name and part number of the item that is not orderable*
- *The date when the component can be ordered (on or before the Availability Date)*
- *The method to be used to order the component (at or below the quoted price) when that date arrives*
- *The method for verifying the price*

All components of this system under test are orderable as of the date of publishing. While all components are orderable immediately, the following components are not available until:

<b>Product</b>	<b>Availability Date</b>
Sun X4800 M2 Server	March 27, 2012
Oracle Database 11g Release 2 with Partitioning for Oracle Linux with uek2	June 26, 2012
Sun Storage F5100 Flash Array Firmware	June 26, 2012

**Table 10: Order-ability Dates**

The pricing for the components does not change. Customers who purchase the components at the current versions will be eligible to download and install upgrades to the levels used in this measurement by the availability date.



## **8 Clause 8: Audit Related Items**

### **8.1 Auditor's Report**

*The auditor's name, address, phone number, and a copy of the auditor's attestation letter indicating compliance must be included in the Full Disclosure Report*

The Auditor's Attestation Letter is included after the Executive Summary on page 4.



# Appendix A: Application Source

## Client Source Code

### tpccClient.c

```
/**
*** tpccClient.c
*** (c) 2000, 2001 Sun Microsystems, Inc. All rights reserved.
***
***/
```

```
#include <stdarg.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
```

```
#include <sys/types.h>
#include <sys/stat.h>
#include <errno.h>
#include <fcntl.h>
```

```
#include "tpccBool.h"
#include "tpcc.h"
```

```
BOOL bInitServer(WORKER *,int,int,BOOL);
BOOL bTuxInit(WORKER *pWorker);
```

```
BOOL bInitServer(WORKER *pWorker,int iTerminalId,int
iWorkerId,BOOL bProcess)
```

```
{
pWorker->iWarehouseId = 0;
pWorker->iDistrictId = 0;
pWorker->iWorkerId = iWorkerId;
pWorker->iTerminalId = iTerminalId;
pWorker->uScreenId = 0;
pWorker->iStatusId = 0;
pWorker->bRequestForm = TRUE;
pWorker->pTuxInData = NULL;
pWorker->pTuxOutData = NULL;
pWorker->ITuxDataLen = 0;
```

```
pWorker->pTpInf = NULL;
strcpy(pWorker->szErrorTxt,"");
strcpy(pWorker->szWork,"");
return(bTuxInit(pWorker));
}
```

### tpccClient.h

```
#include <malloc.h>
#include <stdarg.h>
#include <stdio.h>
#include <stdlib.h>
```

```
#include <string.h>
#include <signal.h>
#include <time.h>
#include <poll.h>
#include <thread.h>
#include <pthread.h>
#include <sys/timeb.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <errno.h>
#include <fcntl.h>
```

```
#include "tpccBool.h"
#include "tpcc.h"
```

```
#define BUFFER 0
#define PLUGIN_DIR "/usr/netscape/server4/plugins"
#define PLUGIN_URL "/tpcc"
#define PLUGIN_OBJ "tpccApp.so"
```

```
BOOL bInitServer(WORKER *,int,int,BOOL);
```

### tpccTux.c

```
/**
*** tpccTux.c
*** (c) 2006 Sun Microsystems, Inc. All rights reserved.
***
*** This file handles the entry points for calling BEA tuxedo
*** As perf Mar 4, 2009 pre-audit: changes for interactive compliance
***
***/
```

```
#include <stdio.h>
#include <errno.h>
#include "tpccBool.h"
#include "tpccTux.h"
#include "tpcc.h"
#include "nsapi.h"
#define TXERRCODE -100
static const char *newo_service = NEWORDER;
static const char *paym_service = PAYMENT;
static const char *ords_service = ORDERSTATUS;
static const char *del_service = DELIVERY;
static const char *stock_service = STOCKLEVEL;
```

```
BOOL bTuxIOInit(WORKER *pWorker)
```

```
{
    BOOL eStatus = FALSE;
    int ret;

    if((pWorker->pTpInf = (TPINIT *)tpalloc("TPINIT", NULL,
        TPINITNEED(0))) == NULL)
        /* if((pWorker->pTpInf = (TPINIT *)tpalloc("TPINIT", NULL,
            sizeof(TPINIT))) == NULL) */
        {
            if(tperrno == TPEOS)
                printf(pWorker->szErrorTxt, "tpalloc failed
```

```

in bTuxIOInit. TxErrno = %d Unix-Error = %d", tperrno, Uunixerr);
        }
        else
        {
            printf(pWorker->szErrorTxt, "tpalloc failed
in bTuxIOInit. TxErrno = %d", tperrno);
        }
        ereport(LOG_FAILURE, "%s Tperno = %d,
%s\n", pWorker->szErrorTxt, tperrno, tpsterror(tperrno));
        ret = tperrordetail(0);
        if(ret == -1) {
            ereport(LOG_FAILURE, "tperrordetail()
failed!\n");
            ereport(LOG_FAILURE, "Tperno = %d, %s\n",
tperrno, tpsterror(tperrno));
        }
        else if (ret != 0) {
            ereport(LOG_FAILURE, "errordetail:
%s\n",
            tpsterrordetail( ret,
0));
        }
        return(TRUE);
    }
    pWorker->pTpInf->flags = TPMULTICONTEXTS;
    pWorker->pTpInf->datalen = 0;

    if(tpinit(pWorker->pTpInf) == -1)
    {
        ereport(LOG_FAILURE, " bTuxIOInit: tpinit failed\n");
        ret = tperrordetail(0);
        if(ret == -1) {
            ereport(LOG_FAILURE, "tperrordetail()
failed!\n");
            ereport(LOG_FAILURE, "Tperno = %d,
%s\n", tperrno, tpsterror(tperrno));
        }
        else if (ret != 0) {
            ereport(LOG_FAILURE, "errordetail:
%s\n",
            tpsterrordetail( ret,
0));
        }
        if(tperrno == TPEOS)
        {
            printf(pWorker->szErrorTxt, "tpinit failed
in bTuxIOInit. TxErrno = %d Unix-Error = %d", tperrno, Uunixerr);
        }
        else
        {
            printf(pWorker->szErrorTxt, "tpinit failed
in bTuxIOInit. TxErrno = %d", tperrno);
        }
        tpfree((char *)pWorker->pTpInf);
        pWorker->pTpInf = NULL;
        ereport(LOG_FAILURE, " bTuxIOInit: tpinit failed returning -
```



```

        if(tperrno == TPEOS)
        {
            sprintf(pWorker->szErrorTxt,
"tpalloc failed in bTuxInit. TxErrno = %d Unix-Error = %d", tperrno,
Uunixerr);
        }
        else
        {
            sprintf(pWorker->szErrorTxt,
"tpalloc failed in bTuxInit. TxErrno = %d", tperrno);
        }
        tpfree((char *)pWorker->pTuxInData);
        pWorker->pTuxInData = NULL;
        pWorker->pTuxOutData = NULL;
        return(TRUE);
    }
}
return(FALSE);
}

BOOL bTuxTerm(WORKER *pWorker)
{
    if(pWorker->pTuxInData != NULL)
    {
        tpfree((char *)pWorker->pTuxInData);
    }
    if(pWorker->pTuxOutData != NULL)
    {
        tpfree((char *)pWorker->pTuxOutData);
    }
    pWorker->pTuxInData = NULL;
    pWorker->pTuxOutData = NULL;
    return(FALSE);
}

tpccTux.h
#ifndef __TPCCTUXH__
#define __TPCCTUXH__
#include <atmi.h> /* TUXEDO */
#include <Uunix.h> /* TUXEDO */
#include <userlog.h> /* TUXEDO */
#include "tpcc.h"
#include "tpccBool.h";
BOOL bTuxTerm(WORKER *);
BOOL bTuxInit(WORKER *);
BOOL bTuxIO(char *szService, WORKER *pTPCC, BOOL *pbTPRslt, int
*piTPRslt);
BOOL bTuxIOInit(WORKER *);
BOOL bTuxIOTerm(WORKER *);
BOOL bTuxTran(char *szService, WORKER *pWorker, BOOL *pbTPRslt,
int *piTPRslt);
#endif

tpccDiag.c
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include "tpccDiag.h"

#include "tpccBool.h"
#include "tpccConst.h"
#include "nsapi.h"
uint uDiagLevel;
char *pDiagHdr;
char *pErrHdr = {"*** TPCC Application Encounterd An Error ***"};
void vDiagInit(char * pDiagId)
{
    char *pszEnvData;
    if(DIAGNOSTICS)
    {
        if(pszEnvData = getenv("DIAGLEVEL")) == NULL)
        {
            uDiagLevel = DEFAULTDIAGLEVEL;
        }
        else
            uDiagLevel = abs(atoi(pszEnvData));
        if(uDiagLevel == 0)
        {
            uDiagLevel = DIAG_ERROR;
        }
        pDiagHdr = (char *) malloc(strlen(pDiagId) + 1);
        strcpy(pDiagHdr,pDiagId);
    }
    /*vDiagInit*/
}

void vDiagTerm(void)
{
    if(DIAGNOSTICS)
    {
        free(pDiagHdr);
    }
}
/*vDiagTerm*/

void vDiagWrite(char * pDiagBuffer, uint uSeverity)
{
    char *pDMsgs[3];
    uint uMsgCnt = 0;
    int iERslt = 0;
    if(DIAGNOSTICS)
    {
        if (uDiagLevel >= uSeverity)
        {
            if (uSeverity == DIAG_ERROR)
            {
                pDMsgs[0] = pDiagHdr;
                pDMsgs[1] = pErrHdr;
                pDMsgs[2] = pDiagBuffer;
                uMsgCnt = 3;
            }
            else
            {
                pDMsgs[0] = pDiagHdr;
                pDMsgs[1] = pDiagBuffer;
                uMsgCnt = 2;
            }
            if(uMsgCnt == 3)
            {
                ereport(LOG_FAILURE, "\n%:
%s\n%s",
                pDMsgs[0],pDMsgs[1],pDMsgs[2]);
            }
            else
            {
                ereport(LOG_FAILURE, "\n%:
%s",
                pDMsgs[0],pDMsgs[1]);
            }
        }
    }
}

tpccDiag.h
#ifndef __TPCCDIAGH__
#define __TPCCDIAGH__
#include <sys/types.h>
void vDiagInit(char * pDiagId);
void vDiagTerm(void);
void vDiagWrite(char * pDiagBuffer, uint uSeverity);
#endif

tpccNsapi.c
/**
*** tpccNsapi.c
*** (c) Sun Microsystems, Inc. All rights reserved.
***
*** This file handles the entry points for the initialization and service
*** routines of the TPC-C plugin for the iPlanet Web Server.
***
*** Modifications made from Review by Pallab Bhattachariya - March 29,
2001
*** DJC
*** Change to add NULL termination to the Request data read by the server.
*** - April 5 DJC
***
***/
#include <sys/types.h>
#include <sys/time.h>

#include "nsapi.h"
#include "tpccBool.h"
#include "tpcc.h"
#include "tpccClient.h"
#include "tpccTux.h"
#include "tpccDiag.h"
#include "tpccService.h"

/*
* string containers for the plugin paths
*/
char szPluginPath[128];

static int iMaxWorkers;
BOOL bMinTidSet = FALSE;

```

```

/* The following are to manage a protected index to the worker array */
static mutex_t  mtThrPvt;
static cond_t  ctIDCond;
static int  iThrPvtCtr = 0;
static int  *iFreeThrd;
static int  iFreeCount;

static thread_key_t  wKey;
static WORKER  **ppWorker;

/*
 * Start time of thread
 */
static unsigned int  uiStartSec;

void fKeyDestructor(void *vWorker)
{
    bTuxTerm((WORKER *)vWorker);
    free(vWorker);
}

/**
 ** GetMyHandle takes in the NSAPI Request struct, which contains the
 ** current
 ** (next) available thread ID in the iWS thread pool.
 **
 **/
#define THREAD_EXTRA 10
WORKER *GetMyHandle(Request *rq)
{
    int  iThrdId = -1;
    int  iIndex = -1;
    WORKER *pWorker = (WORKER *)NULL;
    char *pszReq = NULL;
    int  found = 0;
    int  attempts = 0;

    /* Get next free worker - Use a round robin - try to get equal use on all the
    connects */
    mutex_lock(&mtThrPvt);
    /* Get the data-struct index for this thread */
    /* iIndex = thr_self(); Not done anymore
    */

    while (iFreeCount == 0) {
        cond_wait(&ctIDCond,&mtThrPvt);
        if (iFreeThrd[iThrdId==iThrPvtCtr])
            goto done;
    }
    if (iThrPvtCtr == iMaxWorkers)
    {
        iThrPvtCtr = 0;
    }

    found = 0; attempts = 0;

    while (! found) {
        iThrdId = iThrPvtCtr++;
        if (!iFreeThrd[iThrdId])
        {
            if(iThrPvtCtr == iMaxWorkers)
            {
                attempts++;

                if (attempts == 2)
                {
                    ereport(LOG_FAILURE, "1 Thread %d from Pvt
                    IndexId %d > %d\n", iThrdId, iFreeCount, iMaxWorkers);
                    ereport(LOG_FAILURE, "1 too many attempts
                    -shouldn't happen - EXITING NOW\n");
                    mutex_unlock(&mtThrPvt);
                    return NULL;
                }
                iThrPvtCtr = 0;
            }
            continue;
        }
        break;
    }
    done:
    iFreeThrd[iThrdId]=0; /* This worker is not free anymore */

    iFreeCount--;
    mutex_unlock(&mtThrPvt);

    pWorker = ppWorker[iThrdId];
    pWorker->uiThrdId = iThrdId;

    return(pWorker);
}

WORKER *initHandle(unsigned int uiThrdId)
{
    WORKER *pWorker;
    if((pWorker = (WORKER *)malloc(sizeof(WORKER))) ==
    NULL)
    {
        ereport(LOG_FAILURE, "Malloc State");
        return NULL;
    }

    memset(pWorker, 0, sizeof(WORKER));
    pWorker->pTuxInData = NULL;
    pWorker->pTuxOutData = NULL;
    strcpy(pWorker->szErrorTxt,"");
    pWorker->iStatusId = STATUS_OK;
    pWorker->uiThrdId = uiThrdId;
    /*
     * Initialize the worker data structure.
     */
}

*/
if(bInitServer(pWorker,uiThrdId,uiThrdId,FALSE))
{
    ereport(LOG_FAILURE, pWorker->szErrorTxt);
    free(pWorker);
    return NULL;
}
/*
 * Initialize Tuxedo connection for this thread with tpinit()
 * call.
 */
if(bTuxIOInit(pWorker))
{
    ereport(LOG_FAILURE, pWorker->szErrorTxt);
    free(pWorker);
    return NULL;
}
return pWorker;
}

/*
 * GetTime() gets the duration passed since the uiStartSec. Called by
 * TpcService().
 */
static unsigned int  GetTime(void)
{
    struct timeval  tmNow;

    gettimeofday(&tmNow, (char *)NULL);
    return (((tmNow.tv_sec - uiStartSec) * 1000) + tmNow.tv_usec/1000);
}

/**
 ** TpcInit() is defined as the initialization routine in the server
 ** instance's obj.conf configuration file.
 **
 ** This function reads in the "max-workers" parameter from the TpcInit
 ** section of the obj.conf, which defines the total number of WORKER
 ** structures that the plugin will create and use for the duration of the
 ** server instance's existence.
 **
 ** The input and return variables are assigned according to the NSAPI
 ** specification.
 **
 **/

/* This is kind of a hack to buffer UP the number of worker threads in order
 * to work with the thread id's that get returned in get my handle.
 */

NSAPI_PUBLIC int  TpcInit(pblock *pb, Session *sn, Request *rq)
{
    struct timeval  tmNow;

    int  iLoop;

    mutex_init(&mtThrPvt, USYNC_THREAD, NULL);
}

```

```

cond_init(&ctIDCond, USYNC_THREAD, NULL);

    if(thr_keycreate(&wKey, fKeyDestructor) != 0)
    {
        ereport(LOG_FAILURE, "Global Key State Failed");
        return(REQ_ABORTED);
    }

    /*
    * Get current time
    */
    gettimeofday(&tmNow, (char *)NULL);
    uiStartSec = (tmNow.tv_sec * 1000) + tmNow.tv_usec/1000;

    /*
    * pblock_findval is an NSAPI call; this call determines max-
workers
    * in obj.conf
    * NOTE: make this value HIGHER than rqThrottle for worker
sizing.
    */
    iMaxWorkers = atoi(pblock_findval("max-workers", pb));

    ereport(LOG_INFORM, "TpccInit: Finished initializing %d
workers [ tpccApp.so.] with LIVE TUXEDO/DB backend\n", iMaxWorkers);

    sprintf(szPluginPath, "%s", PLUGIN_URL);

    return REQ_PROCEED;
}

static void freeWorker (int iThrdId)
{
    mutex_lock(&mtThrPvt);
    iFreeThrd[iThrdId]=1; /* free now */
    if(iFreeCount == 0)
    {
        iThrPvtCtr = iThrdId;
    }
    iFreeCount++;
    cond_signal(&ctIDCond);
    mutex_unlock(&mtThrPvt);
    return;
}

/**
** TpccService() is defined as the service routine in the server
** instance's obj.conf configuration file.
**
** The input and return variables are assigned according to the NSAPI
** specification.
**
**/
NSAPI_PUBLIC int TpccService(pblock *pb, Session *sn, Request *rq)
{
    int iTid;
    int iReqContentLen=0;
    int iLen=0;

    int return_value;
    char szReturnLength[8];
    /*following 2 moved to WORKER struct*/
    char *pszRequestData;
    char *pszReturnData;
    char *pszReqMethod;

    unsigned int uiStartReadTime;
    unsigned int uiEndReadTime;
    unsigned int uiStartWriteTime;
    unsigned int uiEndWriteTime;

    int iHtmlLen = 0;
    unsigned int uiStatus;
    WORKER *pWorker;

    /*
    * Gets the worker assigned to this iWS thread (defined in rq).
    */
    if(thr_getspecific(wKey, (void **)&pWorker))
    {
        ereport(LOG_FAILURE, "Invalid Key State");
        return REQ_ABORTED;
    }
    if(!pWorker)
    {
        if(pWorker = initHandle((unsigned int)thr_self()) ==
        NULL)
        {
            ereport(LOG_FAILURE, "Bad Handle
State");
            return REQ_ABORTED;
        }
        if(thr_setspecific(wKey, pWorker))
        {
            ereport(LOG_FAILURE, "Cannot Preserve
My Handle %d\n", thr_self());
            return (REQ_ABORTED);
        }
    }
    pszRequestData = pWorker->szRequestData;
    pszReturnData = pWorker->szReturnData;

    /*
    * Set the default protocol status to 200 (okay). May get changed
later
    * on error. (There is a second setting for this at the end of this
* function -- this one is likely premature and redundant.
* -- grog/20010507)
    */
    protocol_status(sn, rq, PROTOCOL_OK, NULL);

    /*
    * Force the return content's MIME type to text/html.
    */

    param_free(pblock_remove("content-type", rq->srvhdrs));
    pblock_nvinfosert("content-type", "text/html", rq->srvhdrs);

    /*
    * Determine the HTTP method used in the transaction. For our
* purposes, it should be either GET or POST.
    */
    pszReqMethod = pblock_findval("method", rq->reqpb);
    /*
    * If the method was GET, then there is no content data expected,
as
    * defined by the HTTP RFCs. Force the szRequestData to be
empty,
    * which will signal to uiServiceRequest that this is a login
* transaction.
    */
    if (!strncmp(pszReqMethod, "GET", 3))
    {
        strcpy(pszRequestData, "");
    }

    /*
    * If the method was POST, then we need to capture the query
string
    * send in the content field of the POST transaction.
    */
    else if (!strncmp(pszReqMethod, "POST", 4))
    {
        int iReadLength;
        int iLen;
        netbuf *nbuf;

        /*
        * Determine the length of the POST query. This is
defined
        * by the HTTP "Content-type" header value.
        */
        iLen = iReqContentLen =
atoi(pblock_findval("content-length",
rq->headers));

        /*
        * nbuf is a locally defined structure of the NSAPI
* type. The structure contains position, size, and other
* parameters relevant to the incoming data.
        */
        nbuf = sn->inbuf;

        /*
        * iReadLength is set to the length of the buffer (so
far).
        */
        iReadLength = nbuf->cursor - nbuf->pos;

        /*
        * Get any data sitting in the buffer.
        */
    }
}

```

<p>POST that should. buffer &gt;inbuf + nbuf-&gt;pos, read iLen, iReadLength, was the pointer iReadLength, iLen, 10); abort</p>	<pre> if (iReadLength) {     /*      * If iReadLength is longer than the actual      * query's defined length, we force it to be      * size to avoid reading more than we      */     if (iReadLength &gt; iReqContentLen)     {         iReadLength = iReqContentLen;     }      /*      * Copy in the buffer data and advance the      * pointer.      */     (void) memcpy(pszRequestData, nbuf-         &gt;inbuf + nbuf-&gt;pos,             iReadLength);     nbuf-&gt;pos += iReadLength; }  /*  * If we haven't read in all of the data, we will need to  * more data from the socket.  */ if (iReadLength &lt; iLen) {     /*      * We need to redefine the length to be read,      * to be the remainder. Hence we subtract      * the amount read so far, from iLen, which      * initially set to the full length of the query      * string.      */     /* Then read the remainder, starting from      * position offset by iReadLength.      */     iLen -= iReadLength;     iReadLength = net_read(sn-&gt;csd,         pszRequestData +         iReadLength, iLen, 10);      /*uiEndReadTime = GetTime();*/      /*      * If the query was not completely read, </pre>	<pre> /* servicing the HTTP request.  */ if (iReadLength != iLen) {     ereport(LOG_INFORM, "POST         freeWorker(pWorker-&gt;uiThrId);         return (REQ_ABORTED);     } } }  /* end POST  */  /*  * The request data is not terminated by a NULL always - do it just  * in case: DJC.  */ pszRequestData[iReqContentLen] = '\0';  /*  * Call uiServiceRequest(), which processes the HTTP request.  *  * This function takes the data in szRequestData (the query string,  * which is empty if it's GET-based login) and returns szReturnData,  * of length iHtmlLen. szReturnData contains the HTML source  * returned to the requestor.  */ uiStatus = uiServiceRequest(pWorker, pszRequestData,     pszReturnData, &amp;iHtmlLen);  /*  * Force the Content-length: header value to be the length of the  * HTML generated from uiServiceRequest()  */ param_free(pblock_remove("content-length", rq-&gt;srvhdrs)); pblock_nninsert("content-length", iHtmlLen, rq-&gt;srvhdrs);  /*  * NSAPI call to set the return status to successful (200).  */ protocol_status(sn, rq, PROTOCOL_OK, NULL);  /*  * Set the return value which will signal iWS to send the request.  */ return_value = protocol_start_response(sn,rq);  if (return_value == REQ_NOACTION) {     ereport(LOG_INFORM, "TpccService: NOACTION -         start response saw request method HEAD?\n");     return REQ_PROCEED; } uiStartWriteTime = GefTime();*/ </pre>	<pre> return_value = net_write(sn-&gt;csd, pszReturnData, iHtmlLen); /*     uiEndWriteTime = GetTime(); */  if (return_value == IO_ERROR) {     ereport(LOG_INFORM, "TpccService: REQ_EXIT -         start response returned IO_ERROR?\n");     return REQ_EXIT; }  return REQ_PROCEED; }  <b>tpccService.c</b> /**  *** tpccService.c  *** (c) 2000, 2001 Sun Microsystems, Inc. All rights reserved.  ***  *** This file handles the entry points for the initialization and service  *** routines of the TPC-C plugin for the iPlanet Web Server.  ***  *** For best readability, set tab size to 8 spaces.  *** As perf Mar 4, 2009 pre-audit: changes for interactive compliance  *** Mar 2009 DJC - upgrade Short--&gt;Int  ***/  #include &lt;stdio.h&gt; #include &lt;stdlib.h&gt; #include &lt;string.h&gt; #include &lt;ctype.h&gt; #include "tpccDiag.h" #include "tpccService.h" #include "tpccTux.h" #include "nsapi.h"  /*  * Each "screen" (or page) has a unique screen ID.  * Value is stored in pWorker-&gt;iScreenId.  */ #define SCREEN_NULL 0 #define SCREEN_LOGON 1 #define SCREEN_MENU 2 #define SCREEN_NEWORDER 3 #define SCREEN_PAYMENT 4 #define SCREEN_DELIVERY 5 #define SCREEN_ORDERSTATUS 6 #define SCREEN_STOCKLEVEL 7 #define SCREEN_EXIT 8 #define SCREEN_MAX 9  /*  * These values are set based on the value for the CMD= name tag.  * Returned from uiExtractCmd()  */ #define CMD_EXECUTE 1 #define CMD_NEWORDER_SCREEN 2 #define CMD_PAYMENT_SCREEN 3 </pre>
---	---	--	--



```

#define CMD_DELIVERY_SCREEN 4
#define CMD_ORDERSTATUS_SCREEN 5
#define CMD_STOCKLEVEL_SCREEN 6
#define CMD_EXIT 7
#define CMD_SUBMIT 8
#define CMD_MENU_SCREEN 9
#define CMD_MAX 10

/*
 * The following ALT_ constants are used to calculate empty values in name/
 * value pairs in a New Order POST submission.
 */
#define ALT_SIZEF 18 /* size of a valueless triplet Sxx*&Ixx*&Qxx*& */
#define ALT_SIZEEC 12 /* size of CMD=Execute */
#define ALT_SIZEEA 4 /* size of [S|I|Q]xx* */

/*
 * Maximum name tag size
 */
#define ALT_SIZEEW 100

/*
 * name tags for name/value pairs used in POST queries
 */
static const char *szItemList[] = {
    "I00*=", "I01*=", "I02*=", "I03*=", "I04*=",
    "I05*=", "I06*=", "I07*=", "I08*=", "I09*=",
    "I10*=", "I11*=", "I12*=", "I13*=", "I14*=", ""};
static const char *szSupplyList[] = {
    "S00*=", "S01*=", "S02*=", "S03*=", "S04*=",
    "S05*=", "S06*=", "S07*=", "S08*=", "S09*=",
    "S10*=", "S11*=", "S12*=", "S13*=", "S14*=", ""};
static const char *szQuantityList[] = {
    "Q00*=", "Q01*=", "Q02*=", "Q03*=", "Q04*=",
    "Q05*=", "Q06*=", "Q07*=", "Q08*=", "Q09*=",
    "Q10*=", "Q11*=", "Q12*=", "Q13*=", "Q14*=", ""};

extern char szPluginPath[];

/*
 * Valid values for the CMD name tag.
 */
static const char *szCmds[] =
{
    "Unknown",
    "Execute",
    "..NewOrder..",
    "..Payment..",
    "..Delivery..",
    "..Order-Status..",
    "..Stock-Level..",
    "..Exit..",
    "Submit",
    "Menu"
};

/*
 * Static HTML for the initial login screen.
 */
static const char *szLoginScreen =
    "<HTML>"
    "<HEAD><TITLE>Welcome To TPC-"
    "C</TITLE></HEAD><BODY>"
    "Please Identify your Warehouse and District for this
    session.<BR>"
    "<FORM ACTION=\"%s\" METHOD=\"POST\">"
    "<INPUT TYPE=\"hidden\" NAME=\"STID\" VALUE=\"%d\">"
    "<INPUT TYPE=\"hidden\" NAME=\"SCID\" VALUE=\"1\">"
    "<INPUT TYPE=\"hidden\" NAME=\"TRID\" VALUE=\"-2\">"
    "<INPUT TYPE=\"hidden\" NAME=\"WKID\" VALUE=\"0\">"
    "Warehouse ID <INPUT NAME=\"w_id\" SIZE=4<BR>"
    "District ID <INPUT NAME=\"d_id\" SIZE=2<<BR>"
    "<HR>"
    "<INPUT TYPE=\"submit\" NAME=\"CMD\"
    VALUE=\"Submit\">"
    "</FORM>";

/*
 * Static HTML segment for the option list on the menu screen.
 */
static const char *szMenuList =
    "<INPUT TYPE=\"submit\" NAME=\"CMD\"
    VALUE=\"..NewOrder..\">"
    "<INPUT TYPE=\"submit\" NAME=\"CMD\"
    VALUE=\"..Payment..\">"
    "<INPUT TYPE=\"submit\" NAME=\"CMD\"
    VALUE=\"..Delivery..\">"
    "<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Order-
    Status..\">"
    "<INPUT TYPE=\"submit\" NAME=\"CMD\" VALUE=\"..Stock-
    Level..\">"
    "<INPUT TYPE=\"submit\"
    NAME=\"CMD\"VALUE=\"..Exit..\">";

/*
 * HTML footer.
 */
static const char *szEndHtmlTag = "</BODY></HTML>\r\n\r\n";

/*
 * Name tags used for RBE (terminal) status and context identification.
 */
static const char *TRID = "TRID=";
static const char *WKID = "WKID=";
static const char *SCID = "SCID=";
static const char *STID = "STID=";

static const char *CMD = "CMD=";
static const char *ZIPPIC = "XXXXX-XXXXX";

/*
 * Eternal counter for terminal context identifier.
 */
static volatile int iTermCnt = 1;

/*
 * Function prototypes.
 */
BOOL bExecuteLogin(char *, char *, WORKER *, int *);
BOOL bExecuteForm(char *, char *, WORKER *, int *);
BOOL bExecuteNewOrder(char *, char *, WORKER *, int *);
BOOL bExecutePayment(char *, char *, WORKER *, int *);
BOOL bExecuteDelivery(char *, char *, WORKER *, int *);
BOOL bExecuteOrderStatus(char *, char *, WORKER *, int *);
BOOL bExecuteStockLevel(char *, char *, WORKER *, int *);
BOOL bExtractHidden(char *, unsigned int *, int *, int *);

unsigned int uiExtractCmd(char *, char *, unsigned int);

char * pszExtractLongKey(long *, char *, const char *, WORKER *);
char * pszExtractIntKey(int *, char *, const char *, WORKER *);
/* Don't use this anymore:
 * char * pszExtractShortKey(short *, char *, char *, WORKER *);
 */
char * pszExtractStringKey(char *, char *, const char *, WORKER *,
unsigned int);
char * pszExtractAmountKey(double *, char *, const char *, WORKER *);
char * pszExtractKeyValue(char *, const char *, char **, unsigned int);
char * pszExtractWDIDKey(int *, char *, char *, WORKER *);

int iLayoutLogin(char *, WORKER *);
int iLayoutMenu(char *, WORKER *);
int iLayoutNewOrder(char *, WORKER *);
int iLayoutPayment(char *, WORKER *);
int iLayoutDelivery(char *, WORKER *);
int iLayoutOrderStatus(char *, WORKER *);
int iLayoutStockLevel(char *, WORKER *);
int iLayoutFormHdr(char *, char *, WORKER *);
int iLayoutRespHdr(char *, char *, WORKER *);
int iLayoutHTMLString(char *, char *, unsigned int);
int iLayoutString(char *, const char *, char *);

void vStringCopy(char *, char *, int);
void vToUpper(char *);

BOOL bIsNumeric(char *);

/**
 ** uiServiceRequest() is the entry point for transaction processing. It is
 ** called from the TpccService() routine in tpccNsapi.c.
 **
 ** inputs:
 ** pWorker -- the WORKER struct assigned to the current thread.
 ** pszRecvMsg -- the POST query from the RBE
 **
 ** outputs:
 ** pszSendMsg -- HTML content to be returned to the RBE
 ** piSendLen -- length of HTML content
 ** uiStatus --
 **/
unsigned int
uiServiceRequest(WORKER * pWorker, char *pszRecvMsg, char

```

<pre> *pszSendMsg, int *piSendLen) {     int iWorkerId;     int iTerminalId;     unsigned int uiCmdId;     unsigned int uiStatus = SENDCLOSE;     pWorker-&gt;szErrorTxt[0] = 0;     pWorker-&gt;iStatusId = STATUS_OK;     pWorker-&gt;iWarehouseId = 0;     pWorker-&gt;iDistrictId = 0;      /*     * Check to make sure the Tuxedo interface is established and     * functional.     */     if(pWorker-&gt;pTuxInData == NULL)     {         if(bTuxInit(pWorker))         {             pWorker-&gt;iStatusId = ERR_TUX_INTERFACE;         }         *piSendLen = iLayoutLogin(pszSendMsg, pWorker);         goto ServiceXit;     }     /*     * Extract the hidden tags, and if successful (return value of 0),     * assign the relevant value to uiCmdId for the CMD tag     */     if(bExtractHidden(pszRecvMsg, &amp;pWorker-&gt;uScreenId, &amp;iWorkerId, &amp;iTerminalId))     {         if(pWorker-&gt;iWarehouseId != 0)         {             strcpy(pWorker-&gt;szErrorTxt, "Decode hidden fields error");         }         else         {             uiStatus = SEND;         }         *piSendLen = iLayoutLogin(pszSendMsg, pWorker);         goto ServiceXit;     }     uiCmdId = uiExtractCmd(pszRecvMsg, pWorker-&gt;szWork, sizeof(pWorker-&gt;szWork));      /*     *Check for multiple log in attempts.     */     if(pWorker-&gt;iWarehouseId != 0 &amp;&amp; uiCmdId == CMD_SUBMIT)     {         strcpy(pWorker-&gt;szErrorTxt, </pre>	<pre> ERRTXT_ALREADY_LOGGEDIN); pWorker-&gt;iStatusId = ERR_ALREADY_LOGGEDIN; *piSendLen = iLayoutMenu(pszSendMsg, pWorker); uiStatus = SEND; goto ServiceXit; } /* * CMD_SUBMIT is only used for logins. * If we've already logged in, determine this terminal's district and * warehouse ID and make certain they are valid. */ if(uiCmdId != CMD_SUBMIT) {     if(pszExtractWDIDKey(&amp;(pWorker-&gt;iWarehouseId), pszRecvMsg, "w_id=", pWorker) == NULL)     {         *piSendLen = iLayoutLogin(pszSendMsg, pWorker);         goto ServiceXit;     }     if(pszExtractWDIDKey(&amp;(pWorker-&gt;iDistrictId), pszRecvMsg, "d_id=", pWorker) == NULL)     {         *piSendLen = iLayoutLogin(pszSendMsg, pWorker);         goto ServiceXit;     }     if((pWorker-&gt;iWarehouseId == 0    pWorker-&gt;iDistrictId == 0) &amp;&amp; uiCmdId != CMD_SUBMIT)     {         strcpy(pWorker-&gt;szErrorTxt, "Must log in first!");         pWorker-&gt;iStatusId = ERR_WID_INVALID;         *piSendLen = iLayoutLogin(pszSendMsg, pWorker);         goto ServiceXit;     }     /*     * If the CMD value was something other than submit, we will be     expecting     * status, screen, and terminal ID name/value pairs     */     if(uiCmdId != CMD_SUBMIT)     {         if(iTerminalId == iWorkerId)         {             pWorker-&gt;iTerminalId = iTerminalId;             pWorker-&gt;iWorkerId = iWorkerId;         }         if(iTerminalId != pWorker-&gt;iTerminalId    iTerminalId != iWorkerId)         {             sprintf(pWorker-&gt;szErrorTxt, "%s: Received %ld, %ld (%ld)", ERRTXT_TRID, iTerminalId, </pre>	<pre> iWorkerId, pWorker);         goto ServiceXit;     } } /* * Now execute the command based on the uiCmdId. * * CMD_*_SCREEN cases return the data entry screen for the respective * transactions. * CMD_EXECUTE case processes data from executed transactions. * CMD_SUBMIT case logs in the user. * CMD_EXIT case logs out the user. */ switch (uiCmdId) { case CMD_MENU_SCREEN:     pWorker-&gt;iTerminalId = iTerminalId;     pWorker-&gt;iWorkerId = iWorkerId;     *piSendLen = iLayoutMenu(pszSendMsg, pWorker);     break; case CMD_NEWORDER_SCREEN:     pWorker-&gt;iTerminalId = iTerminalId;     pWorker-&gt;iWorkerId = iWorkerId;     *piSendLen = iLayoutNewOrder(pszSendMsg, pWorker);     break; case CMD_PAYMENT_SCREEN:     pWorker-&gt;iTerminalId = iTerminalId;     pWorker-&gt;iWorkerId = iWorkerId;     *piSendLen = iLayoutPayment(pszSendMsg, pWorker);     break; case CMD_DELIVERY_SCREEN:     pWorker-&gt;iTerminalId = iTerminalId;     pWorker-&gt;iWorkerId = iWorkerId;     *piSendLen = iLayoutDelivery(pszSendMsg, pWorker);     break; case CMD_ORDERSTATUS_SCREEN:     pWorker-&gt;iTerminalId = iTerminalId;     pWorker-&gt;iWorkerId = iWorkerId;     *piSendLen = iLayoutOrderStatus(pszSendMsg, pWorker);     break; case CMD_STOCKLEVEL_SCREEN:     pWorker-&gt;iTerminalId = iTerminalId;     pWorker-&gt;iWorkerId = iWorkerId;     *piSendLen = iLayoutStockLevel(pszSendMsg, pWorker);     break; case CMD_EXECUTE: </pre>
---	---	---

<pre> pWorker-&gt;iTerminalId = iTerminalId; pWorker-&gt;iWorkerId = iWorkerId; bExecuteForm(pszRecvMsg, pszSendMsg, pWorker, piSendLen); break; case CMD_SUBMIT: bExecuteLogin(pszRecvMsg, pszSendMsg, pWorker, piSendLen); break; case CMD_EXIT: pWorker-&gt;iWarehouseId = 0; pWorker-&gt;iDistrictId = 0; pWorker-&gt;iTerminalId = -2; pWorker-&gt;iWorkerId = 0; strcpy(pWorker-&gt;szErrorTxt, "Logged Off"); *piSendLen = iLayoutLogin(pszSendMsg, pWorker); goto ServiceXit; default: strcpy(pWorker-&gt;szErrorTxt, ERRTXT_CMD_UNKNOWN); pWorker-&gt;iStatusId = ERR_CMD_UNKNOWN; if(pWorker-&gt;iWarehouseId == 0) { *piSendLen = iLayoutLogin(pszSendMsg, pWorker); goto ServiceXit; } else { pWorker-&gt;iTerminalId = iTerminalId; pWorker-&gt;iWorkerId = iWorkerId; *piSendLen = iLayoutMenu(pszSendMsg, pWorker); } break; } uiStatus = SEND; return(uiStatus); ServiceXit: /* Goto label for unsuccessful transactions */ return(uiStatus); } /** ** bExecuteLogin() processes POST submissions for user logins. It is called ** from uiServiceRequest(). **/ BOOL bExecuteLogin(char *pszInData, char *pszOutData, WORKER * pWorker, int *piSendLen) { int iWarehouseId; int iDistrictId; char *pszPtr; /* * Extract the warehouse ID and make sure it is not less than 1. </pre>	<pre> */ if((pszPtr = pszExtractIntKey(&amp;iWarehouseId, pszInData, "w_id=", pWorker)) == NULL) { *piSendLen = iLayoutLogin(pszOutData, pWorker); return(TRUE); } if(iWarehouseId &lt; 1) { sprintf(pWorker-&gt;szErrorTxt, "Warehouse Id (%d) Invalid", iWarehouseId); pWorker-&gt;iStatusId = ERR_WID_INVALID; *piSendLen = iLayoutLogin(pszOutData, pWorker); return(TRUE); } /* * Extract the district ID and make sure it is within the minimum * maximum bounds. */ if((pszPtr = pszExtractIntKey(&amp;iDistrictId, pszPtr, "d_id=", pWorker)) == NULL) { *piSendLen = iLayoutLogin(pszOutData, pWorker); return(TRUE); } if(iDistrictId &lt; 1) { sprintf(pWorker-&gt;szErrorTxt, "District Id (%d) Invalid", iDistrictId); pWorker-&gt;iStatusId = ERR_DID_INVALID; *piSendLen = iLayoutLogin(pszOutData, pWorker); return(TRUE); } pWorker-&gt;iWarehouseId = abs(iWarehouseId); pWorker-&gt;iDistrictId = abs(iDistrictId); /* * Assign a new terminal and worker (the latter of which is unnecessary) * if we see a value less than 1, indicating a terminal context has not * yet been defined or that the slave has defaulted back to the login * screen. This value is set by a global increment counter, iTermCnt. */ pWorker-&gt;iTerminalId = iTermCnt++; pWorker-&gt;iWorkerId = pWorker-&gt;iTerminalId; *piSendLen = iLayoutMenu(pszOutData, pWorker); return(FALSE); } </pre>	<pre> } /** ** bExecuteForm() processes the form data from POST submissions for the ** five TPC-C transaction types. It is called from uiServiceRequest(). **/ BOOL bExecuteForm(char *pszInData, char *pszOutData, WORKER * pWorker, int *piSendLen) { /* * SCREEN_* is extracted from the SCID= name tag in the POST query from * the RBE, and determines which form processing function should be * executed. * If the screen ID is unrecognized, return an error. */ switch (pWorker-&gt;uScreenId) { case SCREEN_NEWORDER: return(bExecuteNewOrder(pszInData, pszOutData, piSendLen)); case SCREEN_PAYMENT: return(bExecutePayment(pszInData, pszOutData, piSendLen)); case SCREEN_DELIVERY: return(bExecuteDelivery(pszInData, pszOutData, piSendLen)); case SCREEN_ORDERSTATUS: return(bExecuteOrderStatus(pszInData, pszOutData, piSendLen)); case SCREEN_STOCKLEVEL: return(bExecuteStockLevel(pszInData, pszOutData, piSendLen)); default: sprintf(pWorker-&gt;szErrorTxt, "%s (%ld)", ERRTXT_SCREEN_UNKNOWN, pWorker-&gt;uScreenId); pWorker-&gt;iStatusId = ERR_SCREEN_UNKNOWN; *piSendLen = iLayoutMenu(pszOutData, pWorker); break; } return(TRUE); } /** ** bExecuteNewOrder() processes the form data from POST submissions specific ** to the NewOrder transaction. It is called from bExecuteForm(). **/ BOOL </pre>
--	---	--

```

bExecuteNewOrder(char *pszInData, char *pszOutData, WORKER *
pWorker,
int *piSendLen)
{
    NEWO_INF *pNewInf;
    int iLen = 0;
    char szKey[20];
    char szCredit[14];
    char *pszPtr;
    char *pszPtr1;
    unsigned int uiLoop;
    BOOL bDone = FALSE;
    BOOL bTxReturn;
    BOOL bTxResult;
    int iTxResult;
    BOOL supply_is_null, item_is_null, quantity_is_null;
    int empty_slot;
    /*
    * Set pTuxInData such that it conforms to the New Order
    (NEWO_INF)
    * data structure.
    */
    pWorker->ITuxDataLen = sizeof(NEWO_INF);
    pNewInf = (NEWO_INF *)pWorker->pTuxInData;
    pNewInf->w_id = pWorker->iWarehouseId;
    pNewInf->status[0] = 0;

    /*
    * Extract the district ID and check that it is within bounds.
    */
    if(pszPtr = pszExtractIntKey(&pNewInf->d_id, pszInData,
"DID*=",
pWorker)) == NULL)
    {
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }

    /* don't allow less than 1 */
    if(pNewInf->d_id <= 0)
    {
        sprintf(pWorker->szErrorTxt, "DId Invalid - (%d)",
            pNewInf->d_id);
        pWorker->iStatusId = ERR_DID_INVALID;
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }

    /*
    * Extract customer ID and check its validity.
    */
    if(pszPtr = pszExtractIntKey(&pNewInf->c_id, pszPtr, "CID*=",
pWorker)) == NULL)
    {
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }
}

if(pNewInf->c_id <= 0)
{
    sprintf(pWorker->szErrorTxt,
        "CId Invalid - %ld", pNewInf->c_id);
    pWorker->iStatusId = ERR_CID_INVALID;
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}

/*
* Find first value-less name/value pair. As all name tags used by
the
* RBE / plugin environment end with *, "*=&" is a string that
will
* indicate the end of the name tag, the separating equal sign, and
* the ampersand leading to the next name/value pair. All
subsequent
* name tags should have empty values.
*/
pNewInf->o_ol_cnt = 0;
#if 0
pszPtr1 = strstr(pszPtr, "*=&");
if(pszPtr1)
{
    /*
    * Back up ALT_SIZEA (4) characters to beginning of
valueless
    * name tag strings.
    */
    pszPtr1 -= ALT_SIZEA;

    /* Takes the chunk of the POST data starting from the
first
    * blank field to the very end of the data; then backs up
to
    * right before CMD=Execute a name/value pair; then
checks to
    * make sure it is divisible by 3 times the length of the
* triplet of valueless named variables,
* e.g., S10*=&I10*=&Q10*=&
*
* If the resulting remainder is non-zero (i.e., there are
* extra characters where there should be none), an
error
    * occurs.
    */
    /* allow this */
    if((strlen(pszPtr1) - ALT_SIZEA) % ALT_SIZEA)
    {
        sprintf(pWorker->szErrorTxt,
            "Embedded Empty Order Lines Or
Mandatory Fields Blank");
        pWorker->iStatusId =
ERR_MANDATORY_FIELD;
    }
    *piSendLen = iLayoutMenu(pszOutData,
pWorker);
    return(TRUE);
}
#endif

/*
* Now loop through the name/value pairs for the order lines, and
* retrieve the entered warehouse, item, and quantity values.
* This will also check for blank fields and invalid values in the
* form.
*/
empty_slot = -1;
for(uiLoop = 0; uiLoop < MAX_OL; uiLoop++)
{
    supply_is_null = item_is_null = quantity_is_null = FALSE;
    if(pszPtr =
        pszExtractIntKey(&pNewInf->o_ol[uiLoop].ol_supply_w_id,
        pszPtr, szSupplyList[uiLoop], pWorker)) == NULL)
    {
        supply_is_null = TRUE;
    }
    if(pszPtr = pszExtractIntKey(&pNewInf->o_ol[uiLoop].ol_i_id,
        pszPtr, szItemList[uiLoop], pWorker)) == NULL)
    {
        item_is_null = TRUE;
    }
    if(pszPtr = pszExtractIntKey(
        &pNewInf->o_ol[uiLoop].ol_quantity,
        pszPtr, szQuantityList[uiLoop], pWorker)) == NULL)
    {
        quantity_is_null = TRUE;
    }
    /* Ok to have a completely blank line */
    if ( pNewInf->o_ol[uiLoop].ol_supply_w_id == 0 && pNewInf-
>o_ol[uiLoop].ol_i_id == 0 &&
        pNewInf->o_ol[uiLoop].ol_quantity == 0)
    {
        if (empty_slot == -1){
            empty_slot = uiLoop;
        }
        continue;
    }
    /* check for partially filled in line */
    if(pNewInf->o_ol[uiLoop].ol_i_id == 0)
    {
        if(pNewInf->o_ol[uiLoop].ol_supply_w_id != 0)
        {
            sprintf(pWorker->szErrorTxt,
                "Order Line %ld WId Supplied with No Item",
                uiLoop);
            pWorker->iStatusId = ERR_OL_INVALID;
            *piSendLen = iLayoutMenu(pszOutData, pWorker);
        }
    }
}
}

```

```

return(TRUE);
}
if(pNewInf->o_ol[uiLoop].ol_quantity != 0)
{
    sprintf(pWorker->szErrorTxt,
        "Order Line %ld Qty Supplied with No Item",
        uiLoop);
    pWorker->iStatusId = ERR_OL_INVALID;
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}
if(pNewInf->o_ol[uiLoop].ol_i_id != 0)
{
    if(pNewInf->o_ol[uiLoop].ol_supply_w_id == 0)
    {
        sprintf(pWorker->szErrorTxt,
            "Order Line %ld No S-WID",
            uiLoop);
        pWorker->iStatusId = ERR_OL_INVALID;
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }
    if(pNewInf->o_ol[uiLoop].ol_quantity == 0)
    {
        sprintf(pWorker->szErrorTxt,
            "Order Line %ld No Qunatity",
            uiLoop);
        pWorker->iStatusId = ERR_OL_INVALID;
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }
}
if(empty_slot != -1) {
    pNewInf->o_ol[empty_slot] = pNewInf->o_ol[uiLoop];
    /*after copy over data - zero out source area, just in case */
    pNewInf->o_ol[uiLoop].ol_supply_w_id=0;
    pNewInf->o_ol[uiLoop].ol_i_id=0;
    pNewInf->o_ol[uiLoop].ol_quantity=0;
    empty_slot++;
}
    pNewInf->o_ol_cnt++;
} /* end for() */
/*
 * Checks for minimum order line requirement (TPC-C v3.5 is
five).
 */
/* PASS THRU -
if(pNewInf->o_ol_cnt < MIN_OL)
{
    sprintf(pWorker->szErrorTxt, "Too Few Order Lines
%d",
        pNewInf->o_ol_cnt);
    pWorker->iStatusId = ERR_OL_COUNT;
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}
}

*/
        /*
        * Perform the Tuxedo transaction.
        */
        bTxReturn = bTuxTran(NEWORDER, pWorker, &bTxResult,
        &iTxResult);
        pNewInf = (NEWO_INF *)pWorker->pTuxOutData; /*
        Redundant? OutData */
        if(bTxReturn)
        {
            ereport(LOG_WARN, "%d] bExecuteNewOrder: Service Returned
            Error(%ld) : %s \n",
                pWorker->uiThrId, iTxResult, pNewInf->status );
            pWorker->iStatusId = ERR_TUX_INTERFACE;
            *piSendLen = iLayoutMenu(pszOutData, pWorker);
            return(TRUE);
        }
        /*
        * Check to make sure Tuxedo call was successful.
        */
        if(bTxResult && (iTxResult < SVC_NOERROR))
        {
            ereport(LOG_WARN, "%d] bExecuteNewOrder: Service Returned
            Error(%ld) : %s \n",
                pWorker->uiThrId, iTxResult, pNewInf->status );
            sprintf(pWorker->szErrorTxt,
                "New Order Service Returned Error(%ld) :
                %s",
                    iTxResult, pNewInf->status);
            pWorker->iStatusId = ERR_SERVICE_RSLT;
            *piSendLen = iLayoutMenu(pszOutData, pWorker);
            return(TRUE);
        }
        if(iTxResult == SVC_BADITEMID) {
            ereport(LOG_WARN, "%d] bExecuteNewOrder:
            Tuxedo Service : %s \n", pNewInf->status);
            pWorker->iStatusId = INVALID_ITEM_ID;
        }
        /*
        * Now package up the return data from the query (new order line
        * items, customer information, item prices) from the database into Item Name"
        * an HTML page.
        */
        /*
        * HTTP header and HTML title
        */
        iLen = iLayoutRespHdr(pszOutData, "TPC-C New Order",
        pWorker);
        /*
        * district and warehouse IDs
        */
        iLen+=sprintf(pszOutData + iLen,

                "<PRE>                               New Order<BR>"
                "Warehouse: %4.4d District: %2.2d",
                pNewInf->w_id, pNewInf->d_id);

                /*
                * transaction date/time stamp
                */
                if(!bTxResult)
                {
                    iLen+=sprintf(pszOutData + iLen,
                        "Date: %s <BR>", pNewInf->o_entry_d);
                }
                else
                {
                    iLen+=sprintf(pszOutData + iLen, "Date:<BR>");
                }

                /*
                *customer name, customer's credit
                */
                iLayoutHTMLString(pWorker->szWork, pNewInf->c_last,
                NAME_LEN);
                iLayoutHTMLString(szCredit, pNewInf->c_credit, 2);
                iLen+=sprintf(pszOutData + iLen,
                    "Customer: %4.4d Name: %s Credit: %s ",
                    pNewInf->c_id, pWorker->szWork, szCredit);

                if(!bTxResult)
                {
                    /*
                    * discount, and tax information for this transaction
                    */
                    iLen+=sprintf(pszOutData + iLen,
                        "%5.2f <BR>", pNewInf-
                        >c_discount);

                    iLen+=sprintf(pszOutData + iLen,
                        "Order Number: %8.8d Number of Lines:
                        %2.2d"
                        " W_tax: %5.2f D_tax: %5.2f",
                        pNewInf->o_id, pNewInf->o_ol_cnt,
                        pNewInf->w_tax,
                        pNewInf->d_tax);
                    iLen+=sprintf(pszOutData + iLen, " Supp_W Item_Id
                    " Qty Stock B/G Price Amount<BR>");

                    /*
                    * individual order lines
                    */
                    for(uiLoop = 0; uiLoop < (unsigned int)pNewInf-
                    >o_ol_cnt;
                    uiLoop++)
                    {
                        iLayoutHTMLString(pWorker->szWork,
                            pNewInf-

```





```

50);
iLayoutHTMLString(szWork3, szWork2,
if(!iLoop)
iLen+=sprintf(pszOutData + "OCD*=",
"Cust-Data:
else
iLen+=sprintf(pszOutData +
"%s<BR>", szWork3);
iLen,
"%s<BR>",
szWork3);
}
else
iLen+=sprintf(pszOutData + iLen,
"Cust-Data: <BR> <BR> <BR> <BR>");
iLen+=sprintf(pszOutData + iLen,
"<PRE><HR><BR>%s</FORM>%s", szMenuList,
szEndHtmlTag);
/*
* End of HTML generation for Payment.
*/
*piSendLen = iLen;
return(FALSE);
}
/**
** bExecuteDelivery() processes the form data from POST submissions
specific
** to the Delivery transaction. It is called from bExecuteForm().
**/
BOOL
bExecuteDelivery(char *pszInData, char *pszOutData, WORKER * pWorker,
int *piSendLen)
{
REQ_DEL *pReqDel;
DEL_INF *pDelInf;
DEL_INF DelInf;
int iLen = 0;
BOOL bTxReturn;
BOOL bTxResult;
int iTxResult;
struct timeval tv;

pDelInf = &DelInf;

/*
* Set pTuxInData such that it conforms to the Delivery
(DEL_INF)
* data structure.
*/
pWorker->ITuxDataLen = sizeof(REQ_DEL);
pReqDel = (REQ_DEL *)pWorker->pTuxInData;
pReqDel->w_id = pWorker->iWarehouseId;

szWork2[0] = 0;
/*
* Extract the carrier ID and check that it is within bounds.
*/
if(pszExtractIntKey(&pReqDel->o_carrier_id, pszInData,
"OCD*=",
pWorker) == NULL)
*piSendLen = iLayoutMenu(pszOutData, pWorker);
return(TRUE);
}
/* Pass thru
if(pReqDel->o_carrier_id < MIN_CARRIER
|| pReqDel->o_carrier_id > MAX_CARRIER)
{
sprintf(pWorker->szErrorTxt,
"Carrier Id Out of Range(%ld,%ld) - %ld",
MIN_CARRIER, MAX_CARRIER,
pReqDel->o_carrier_id);
pWorker->iStatusId = ERR_CARRIER_INVALID;
*piSendLen = iLayoutMenu(pszOutData, pWorker);
return(TRUE);
}
if(pReqDel->o_carrier_id < 1)
{
sprintf(pWorker->szErrorTxt, "Carrier Id Invalid");
pWorker->iStatusId = ERR_CARRIER_INVALID;
*piSendLen = iLayoutMenu(pszOutData, pWorker);
return(TRUE);
}
}
/*
* Make a note of the time the delivery request was received. This
* is necessary as the Delivery transaction is asynchronous, and
* its time of execution is needed at the time the report is created.
*/
(void)gettimeofday(&tv, NULL); /* convert to msec */
pReqDel->qtime = (tv.tv_sec * 1000) + (tv.tv_usec / 1000);

/*
* Perform the Tuxedo transaction.
*/
bTxReturn = bTuxTran(DELIVERY, pWorker, &bTxResult,
&iTxResult);
if(bTxReturn)
{
pWorker->iStatusId = ERR_TUX_INTERFACE;
*piSendLen = iLayoutMenu(pszOutData, pWorker);
return(TRUE);
}
pDelInf->o_carrier_id = pReqDel->o_carrier_id;
pDelInf->w_id = pReqDel->w_id;

/*
* Now package up the return data from the database in HTML
format.
*/
strcpy(pDelInf->status, "Delivery has been queued.");

iLen = iLayoutRespHdr(pszOutData, "TPC-C Delivery",
pWorker);
iLen+=sprintf(pszOutData + iLen,
"<PRE>
Delivery<BR>"
"Warehouse: %4.4d<BR><BR>"
"Carrier Number: %2.2d<BR><BR>"
"Execution Status: %25.25s<BR>",
pDelInf->w_id, pDelInf->o_carrier_id, pDelInf-
>status);
iLen+=sprintf(pszOutData + iLen,
"</PRE><HR><BR>%s</FORM>%s", szMenuList,
szEndHtmlTag);
/*
* End of HTML generation for Delivery.
*/
*piSendLen = iLen;
return(FALSE);
}
/**
** bExecuteOrderStatus() processes the form data from POST submissions
** specific to the OrderStatus transaction. It is called from bExecuteForm().
**/
BOOL
bExecuteOrderStatus(char *pszInData, char *pszOutData, WORKER *
pWorker, int *piSendLen)
{
ORD_INF *pOrdInf;
int iLen = 0;
int iLoop;
char *pszPtr;
char szWork2[50];
char szWork3[50];
BOOL bTxReturn;
BOOL bTxResult;
int iTxResult;

/*
* Set pTuxInData such that it conforms to the Order Status
(ORD_INF)
* data structure.
*/
pWorker->ITuxDataLen = sizeof(ORD_INF);
pOrdInf = (ORD_INF *)pWorker->pTuxInData;
pOrdInf->c_id = 0;
pOrdInf->c_last[0] = 0;
pOrdInf->w_id = pWorker->iWarehouseId;

/*
* Extract the district ID and check that it is within bounds.
*/
if((pszPtr = pszExtractIntKey(&pOrdInf->d_id, pszInData,
"DID*=",
pWorker)) == NULL)
{
*piSendLen = iLayoutMenu(pszOutData, pWorker);
return(TRUE);
}
}

```



```

}
if(pOrdInf->d_id < 1)
{
    sprintf(pWorker->szErrorTxt,
        "DId Invalid - (%d)",
        pOrdInf->d_id);
    pWorker->iStatusId = ERR_DID_INVALID;
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}
/*
 * Extract the customer ID and check its validity.
 */
if((pszPtr = pszExtractIntKey(&pOrdInf->c_id, pszPtr, "CID*=",
    pWorker)) == NULL)
{
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}
if((pszPtr = pszExtractStringKey(pOrdInf->c_last, pszPtr,
    "CLT*=",
    pWorker, NAME_LEN)) == NULL)
{
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}
if(pOrdInf->c_id == 0 && pOrdInf->c_last[0] == 0)
{
    strepy(pWorker->szErrorTxt,
        "Error - Customer Id and Name Empty");
    pWorker->iStatusId = ERR_IDANDNAME_EMPTY;
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}
if(pOrdInf->c_id != 0 && pOrdInf->c_last[0] != 0)
{
    strepy(pWorker->szErrorTxt,
        "Error - Specify Customer Id or Name, not Both");
    pWorker->iStatusId =
ERR_IDANDNAME_ENTERED;
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}
/*
 * Perform the Tuxedo transaction.
 */
bTxReturn = bTuxTran(ORDERSTATUS, pWorker, &bTxResult,
&iTxResult);
/*
 * Check to make sure Tuxedo call was successful.
 */
pOrdInf = (ORD_INF *)pWorker->pTuxOutData; /* ??redundant
OutData */
if(bTxReturn)
{
    pWorker->iStatusId = ERR_TUX_INTERFACE;
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}
if(bTxResult)
{
    sprintf(pWorker->szErrorTxt,
        "Order Status Service Returned
        Error(%ld) :",
        iTxResult);
    pWorker->iStatusId = ERR_SERVICE_RSLT;
    *piSendLen = iLayoutMenu(pszOutData, pWorker);
    return(TRUE);
}
/*
 * Now package up the return data from the database in HTML
 format.
 */
iLen = iLayoutRespHdr(pszOutData, "TPC-C Order-Status",
    pWorker);
iLen+=sprintf(pszOutData + iLen,
    "<PRE>
    Order-Status<BR>"
    "Warehouse: %4.4d District: %2.2d<BR>",
    pOrdInf->w_id, pOrdInf->d_id);
iLayoutHTMLString(pWorker->szWork, pOrdInf->c_first,
    NAME_LEN);
iLayoutHTMLString(szWork2, pOrdInf->c_middle, 2);
iLayoutHTMLString(szWork3, pOrdInf->c_last, NAME_LEN);
iLen+=sprintf(pszOutData + iLen,
    "Customer: %4.4d Name: %s %s %s<BR>"
    "Cust-Balance: $%9.2f<BR><BR>",
    pOrdInf->c_id, pWorker->szWork, szWork2, szWork3,
    pOrdInf->c_balance);
iLen+=sprintf(pszOutData + iLen,
    "Order-Number: %8.8d Entry-Date: %s Carrier-Number:
    %2.2d<BR>"
    "Supply-W Item-Id Qty Amount Delivery-Date<BR>",
    pOrdInf->o_id, pOrdInf->o_entry_d,
    pOrdInf->o_carrier_id);
/*
 * generate the order line information for this order status
 * transaction
 */
for(iLoop = 0; iLoop < pOrdInf->o_ol_cnt; iLoop++)
{
    iLen+=sprintf(pszOutData + iLen,
        " %4.4d %6.6d %2.2d $%7.2f
        pOrdInf->s_ol[iLoop].ol_supply_w_id,
        pOrdInf->s_ol[iLoop].ol_i_id,
        pOrdInf->s_ol[iLoop].ol_quantity,
        pOrdInf->s_ol[iLoop].ol_amount,
        pOrdInf->s_ol[iLoop].ol_delivery_d);
}
iLen+=sprintf(pszOutData + iLen,
    "<BR></PRE><HR><BR>%s</FORM>%s",
    szMenuList, szEndHtmlTag);
/*
 * End of HTML generation for Order Status.
 */
*piSendLen = iLen;
return(FALSE);
}
/**
 ** bExecuteStockLevel() processes the form data from POST submissions
 specific
 ** to the Stock Level transaction. It is called from bExecuteForm().
 **/
BOOL
bExecuteStockLevel(char *pszInData, char *pszOutData, WORKER *
    pWorker, int *piSendLen)
{
    STOCK_INF *pStockInf;
    int iLen = 0;
    BOOL bTxReturn;
    BOOL bTxResult;
    int iTxResult;
    /*
     * Set pTuxInData such that it conforms to the Stock Level
     (STOCK_INF)
     * data structure.
     */
    pWorker->ITuxDataLen = sizeof(STOCK_INF);
    pStockInf = (STOCK_INF *)pWorker->pTuxInData;
    pStockInf->w_id = pWorker->iWarehouseId;
    pStockInf->d_id = pWorker->iDistrictId;
    pStockInf->low_stock = 0;
    /*
     * Extract the threshold level and make sure it is valid.
     */
    if(pszExtractIntKey(&pStockInf->thresh_hold, pszInData,
    "THR*=",
    pWorker) == NULL)
    {
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }
    /*
     * if(pStockInf->thresh_hold < MIN_THRESHOLD
     || pStockInf->thresh_hold > MAX_THRESHOLD)
     {
        sprintf(pWorker->szErrorTxt,
            "Threshold Out of Range(%ld,%ld) - %ld",
            MIN_THRESHOLD,
            MAX_THRESHOLD, pStockInf->thresh_hold);
        pWorker->iStatusId = ERR_THRESHOLD_RANGE;
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }
}

```

```

*/
    if(pStockInf->thresh_hold < 1)
    {
        sprintf(pWorker->szErrorTxt, "Threshold Out of
Range");
        pWorker->iStatusId = ERR_THRESHOLD_RANGE;
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }
    /*
    * Perform the Tuxedo transaction.
    */
    bTxReturn = bTuxTran(STOCKLEVEL, pWorker, &bTxResult,
&iTxResult);
    pStockInf = (STOCK_INF *)pWorker->pTuxOutData; /* ??
redundant OutData */
    if(bTxReturn)
    {
        pWorker->iStatusId = ERR_TUX_INTERFACE;
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }
    if(bTxResult)
    {
        sprintf(pWorker->szErrorTxt,
Error("%ld) :",
            iTxResult);
        pWorker->iStatusId = ERR_SERVICE_RSLT;
        *piSendLen = iLayoutMenu(pszOutData, pWorker);
        return(TRUE);
    }
    /*
    * Now package up the return data from the database in HTML
format.
    */
    iLen = iLayoutRespHdr(pszOutData, "TPC-C Stock Level",
pWorker);
    iLen+=sprintf(pszOutData + iLen,
        "<PRE>          Stock-Level<BR>"
        "Warehouse: %4.4d  District: %2.2d<BR><BR>"
        "Stock Level Threshold: %2.2d<BR><BR>"
        "Low Stock: %3.3ld</PRE><BR><BR>"
        "%s</FORM>%s",
        pWorker->iWarehouseId, pWorker->iDistrictId,
        pStockInf->thresh_hold, pStockInf->low_stock,
        szMenuList, szEndHtmlTag);
    *piSendLen = iLen;
    /*
    * End of HTML generation for Stock Level.
    */
    return(FALSE);
}

/**
** bExtractHidden() finds and extracts the hidden values for TRID (terminal
** ID), WKID (worker ID), and SCID (screen ID). These are submitted as
*/
    ** name/value pairs via an emulated user's POST command. These tags are
    ** referred to as "hidden" because they use the <INPUT TYPE="HIDDEN">
HTML
    ** markup, which is used to send fixed values automatically with POST
    ** commands.
    **
    ** Called by uiServiceRequest().
    **/
    BOOL
    bExtractHidden(char *pszMessage, unsigned int *uScreenId, int *iWorkerId,
    int *iTerminalId)
    {
        char *pszPtr;
        BOOL bStatus = TRUE;

        /*
        * Extract TRID.
        */
        pszPtr = strstr(pszMessage, TRID);
        if(pszPtr == NULL)
            goto xit;
        pszPtr += 5;
        *iTerminalId = atoi(pszPtr);

        /*
        * Extract WKID.
        */
        pszPtr = strstr(pszMessage, WKID);
        if(pszPtr == NULL)
            goto xit;
        pszPtr += 5;
        *iWorkerId = atoi(pszPtr);

        /*
        * Extract SCID.
        */
        pszPtr = strstr(pszMessage, SCID);
        if(pszPtr == NULL)
            goto xit;
        pszPtr += 5;
        *uScreenId = abs(atoi(pszPtr));
        bStatus = FALSE;
    }
    return(bStatus);
}

/**
** uiExtractCmd() finds and extracts the value associated with the CMD tag.
**
** Called by uiServiceRequest().
**/
    unsigned int
    uiExtractCmd(char *pszMessage, char *pWork, unsigned int uiLen)
    {
        unsigned int u;
        char *pszPtr;
        char *pUpd;

        /*
        * Check for CMD key.
        */
        if(!(pszPtr = strstr(pszMessage, CMD)))
            return(0);
        pszPtr += 4;
        pUpd = pWork;
        while(*pszPtr && *pszPtr != '&' && *pszPtr != '\r' && *pszPtr !=
            '\n')
            *pUpd++ = *pszPtr++;
        *pUpd = 0;

        /*
        * Convert command name into command index.
        */
        for(u = 0; u < CMD_MAX; u++)
        {
            if(!strcmp(szCmds[u], pWork))
                return(u);
        }

        /*
        * Command string not found.
        */
        return(0);
    }

    /**
    ** pszExtractLongKey() finds and extracts the value associated with the
    ** variable pszKeyVal and stores it in a long variable.
    **/
    char *
    pszExtractLongKey(long *lpResult, char *pszHTML, const char *pszKeyVal,
    WORKER * pWorker)
    {
        char *pszRet;
        char *pszWork;
        if(pszRet = pszExtractKeyValue(pszHTML, pszKeyVal,
&pszWork,
            ALT_SIZEW)) == NULL)
        {
            sprintf(pWorker->szErrorTxt,
                "Error - Missing %s Key", pszKeyVal);
            pWorker->iStatusId = ERR_MISSING_KEY;
            return(NULL);
        }
        if(*pszWork != 0)
        {
            if(bIsNumeric(pszWork))
                sprintf(pWorker->szErrorTxt,
                    "Error - %s Value Not Numeric",
                    pszKeyVal);
            pWorker->iStatusId =
                ERR_NOT_NUMERIC;
            return(NULL);
        }
    }
}

```

```

        *ipResult = atoi(pszWork);
        return(pszRet);
    }

/**
** pszExtractIntKey() finds and extracts the value associated with the
** variable pszKeyVal and stores it in a int variable.
**
** This function is called from various bExtract<transaction>() functions.
**/
char *
pszExtractIntKey(int *ipResult, char *pszHTML, const char *pszKeyVal,
WORKER * pWorker)
{
    char *pszRet;
    char *pszWork;
    if((pszRet = pszExtractKeyValue(pszHTML, pszKeyVal,
&pszWork,
        ALT_SIZEW)) == NULL)
    {
        sprintf(pWorker->szErrorTxt,
            "Error - Missing %s Key", pszKeyVal);
        pWorker->iStatusId = ERR_MISSING_KEY;
        return(NULL);
    }
    if(*pszWork != 0)
    {
        if(bIsNumeric(pszWork))
        {
            sprintf(pWorker->szErrorTxt,
                "Error - %s Value Not Numeric",
                pszKeyVal);
            pWorker->iStatusId =
                ERR_NOT_NUMERIC;
            return(NULL);
        }
        *spResult = (short)atoi(pszWork);
        return(pszRet);
    }
    /**
    ** pszExtractWDIDKey() finds and extracts the value of the emulated user's
    ** Warehouse ID.
    **
    ** This function is called from uiServiceRequest().
    **/
    char *
    pszExtractWDIDKey(int *spResult, char *pszHTML, char *pszKeyVal,
WORKER *pWorker)
    {
        char *pszRet;
        char *pszWork;
        char *pszPtr;
        int uiMax = ALT_SIZEW;
        pszRet = pszHTML;
        if(!(pszWork = strstr(pszHTML, pszKeyVal)))
            return(NULL);
        pszWork+=(ALT_SIZEA + 1);
        uiMax--;
        pszPtr = pszWork;
        while(*pszWork && *pszWork != '&' && uiMax)
        {
            *pszPtr++ = *pszWork++;
            uiMax--;
        }
        *pszPtr = 0;
        if(pWorker->szWork[0] != 0)
        {
            if(bIsNumeric(pWorker->szWork))
            {
                sprintf(pWorker->szErrorTxt,
                    "Error - %s Value Not Numeric",
                    pszKeyVal);
                pWorker->iStatusId =
                    ERR_NOT_NUMERIC;
                return(NULL);
            }
        }
        *ipResult = atoi(pszWork);
        return(pszRet);
    }

/**
** pszExtractShortKey() finds and extracts the value associated with the
** variable pszKeyVal and stores it in a short variable.
**
** This function is called from various bExtract<transaction>() functions.
**/
char *
pszExtractShortKey(short *spResult, char *pszHTML, const char
*pszKeyVal, WORKER *pWorker)
{
    char *pszRet;
    char *pszWork;
    if((pszRet = pszExtractKeyValue(pszHTML, pszKeyVal,
&pszWork,
        ALT_SIZEW)) == NULL)
    {
        sprintf(pWorker->szErrorTxt,
            "Error - Missing %s Key", pszKeyVal);
        pWorker->iStatusId = ERR_MISSING_KEY;
        return(NULL);
    }
    if(*pszWork != 0)
    {
        if(bIsNumeric(pWorker->szWork))
        {
            sprintf(pWorker->szErrorTxt,
                "Error - %s Value Not Numeric",
                pszKeyVal);
            pWorker->iStatusId =
                ERR_NOT_NUMERIC;
            return(NULL);
        }
    }
    *spResult = atoi(pWorker->szWork);
    return(pszRet);
}

/**
** pszExtractStringKey() finds and extracts the value associated with the
** variable pszKeyVal and stores it in a string.
**
** This function is called from various bExtract<transaction>() functions.
**/
char *
pszExtractStringKey(char *szResult, char *pszHTML, const char
*pszKeyVal,
WORKER * pWorker, unsigned int uiMax)
{
    unsigned int uiLen;
    char *pszRet;
    char *pszWork;
    if((pszRet = pszExtractKeyValue(pszHTML, pszKeyVal,
&pszWork,
        ALT_SIZEW)) == NULL)
    {
        sprintf(pWorker->szErrorTxt,
            "Error - Missing %s Key", pszKeyVal);
        pWorker->iStatusId = ERR_MISSING_KEY;
        return(NULL);
    }
    uiLen = strlen(pszWork);
    if(uiLen > uiMax)
    {
        sprintf(pWorker->szErrorTxt,
            "Error - %s Key Input (%ld) Too Long
            (%ld)",
            pszKeyVal, uiLen, uiMax);
        pWorker->iStatusId = ERR_INPUT_TOO_LONG;
        return(NULL);
    }
    vToUpper(pszWork);
    strcpy(szResult, pszWork);
    return(pszRet);
}

/**
** pszExtractAmountKey() finds and extracts a dollar amount value
associated
** with the tag defined in variable pszKeyVal and stores it in a double.
**
** This function is called from various bExtract<transaction>() functions.
**/
char *
pszExtractAmountKey(double *dpResult, char *pszHTML, const char
*pszKeyVal,
WORKER *pWorker)
{
    char *pszPtr;
    char *pszRet;
    char *pszWork;

```

```

BOOL      bInvalid = FALSE;
if((pszRet = pszExtractKeyValue(pszHTML, pszKeyVal,
&pszWork,
    ALT_SIZEW)) == NULL)
{
    sprintf(pWorker->szErrorTxt,
        "Error - Missing %s Key", pszKeyVal);
    pWorker->iStatusId = ERR_MISSING_KEY;
    return(pszRet);
}
pszPtr = pszWork;
while(*pszPtr)
{
    if(*pszPtr == '.')
    {
        pszPtr++;
        if(!*pszPtr)
            break;
        if(*pszPtr < '0' || *pszPtr > '9')
        {
            bInvalid = TRUE;
            break;
        }
        pszPtr++;
        if(!*pszPtr)
            break;
        if(*pszPtr < '0' || *pszPtr > '9')
        {
            bInvalid = TRUE;
            break;
        }
        pszPtr++;
        if(*pszPtr)
        {
            bInvalid = TRUE;
            break;
        }
        break;
    }
    else if(*pszPtr < '0' || *pszPtr > '9')
    {
        bInvalid = TRUE;
        break;
    }
    pszPtr++;
}
if(!bInvalid)
    *dpResult = atof(pszWork);
else
{
    sprintf(pWorker->szErrorTxt,
        "Error - Invalid Amount iLayout (%s)", pszWork);
    pWorker->iStatusId = ERR_AMOUNT_BADFORM;
    pszRet = NULL;
}
return(pszRet);
}
/**
** pszExtractKeyValue() finds and extracts the raw value associated with the
** variable pszKeyVal and stores it in a string.
** This function is called from pszExtract<type>Key() functions which take
** the return values and convert them to their respective types.
**/
char *
pszExtractKeyValue(char *pszHTML, const char *pszKeyVal, char
**pszValue,
                    unsigned int uiMax)
{
    char    *pszPtr;
    char    *pszValue1;
    if(!(pszValue1 = strstr(pszHTML, pszKeyVal)))
    {
        *pszValue = NULL;
        return(NULL);
    }
    pszValue1+=(ALT_SIZEA + 1);
    *pszValue = pszValue1;
    uiMax--;
    if(!(pszPtr = strchr(pszValue1, '&')))
    {
        pszPtr = pszValue1;
        while(*pszPtr && *pszPtr != '&' && uiMax)
        {
            pszPtr++;
            uiMax--;
        }
        *pszPtr = 0;
        pszPtr++;
        return(pszPtr);
    }
}
/**
** iLayoutLogin() produces the HTML code segment for the TPC-C Login
Screen.
** It is called by uiServiceRequest().
**/
int
iLayoutLogin(char *pszOutData, WORKER *pWorker)
{
    int iLen = 0;
    char    szTemp[512];
    sprintf(szTemp, szLoginScreen, szPluginPath, pWorker-
>iStatusId);
    iLen = sprintf(pszOutData, "%s<BR>%s<BR>%s", szTemp,
                    pWorker->szErrorTxt, szEndHtmlTag);
    return(iLen);
}
/**
** iLayoutMenu() produces the HTML code segment for the TPC-C Main
Menu.
** It is called by uiServiceRequest().
**/
int
iLayoutMenu(char *pszOutData, WORKER * pWorker)
{
    int iLen = 0;
    iLen = sprintf(pszOutData,
        "<HTML><HEAD><TITLE>TPC-C
MainMenu</TITLE></HEAD><BODY>"
        "Select Desired Transaction.<BR><HR>"
        "<FORM ACTION=\"%s\" METHOD=\"%POST\">",
        szPluginPath);
    iLen+=sprintf(pszOutData + iLen,
        "<INPUT TYPE=\"%hidden\" NAME=\"%STID\"
VALUE=\"%d\">"
        "<INPUT TYPE=\"%hidden\" NAME=\"%SCID\"
VALUE=\"%d\">"
        "<INPUT TYPE=\"%hidden\" NAME=\"%TRID\"
VALUE=\"%d\">"
        "<INPUT TYPE=\"%hidden\" NAME=\"%WKID\"
VALUE=\"%d\">"
        "<INPUT TYPE=\"%hidden\" NAME=\"%w_id\"
VALUE=\"%d\">"
        "<INPUT TYPE=\"%hidden\" NAME=\"%d_id\"
VALUE=\"%d\">",
        pWorker->iStatusId, SCREEN_MENU,
        pWorker->iTerminalId, pWorker->iWorkerId,
        pWorker->iWarehouseId, pWorker->iDistrictId);
    iLen+=sprintf(pszOutData + iLen, "%s</FORM><BR>%s<BR>"
    %s",
        szMenuList, pWorker->szErrorTxt, szEndHtmlTag);
    return(iLen);
}
/**
** iLayoutNewOrder() produces the HTML code for the New Order entry
screen
** It is called by uiServiceRequest().
**/
int
iLayoutNewOrder(char *pszOutData, WORKER * pWorker)
{
    int iLen = 0;
    pWorker->uScreenId = SCREEN_NEWORDER;
    iLen = iLayoutFormHdr(pszOutData, "TPC-C New Order",
pWorker);
    iLen+=sprintf(pszOutData + iLen,
        "<PRE>
New Order<BR>"
        "Warehouse: %4.4d%s%s",
        pWorker->iWarehouseId,
        " District: <INPUT NAME=\"%DID*\" SIZE=2>Date:<BR>"
        "Customer: <INPUT NAME=\"%CID*\" SIZE=4> Name:Credit:"
        " %Disc:<BR>"
        "Order Number:      Number of Lines:   W_tax:."
        " D_tax:<BR><BR>"
        " Supp_W Item_Id Item Name          Qty Stock"
        " B/G Price Amount<BR>"
        "<INPUT NAME=\"%S00*\" SIZE=4> <INPUT NAME=\"%I00*")
}

```

```

SIZE=6>"          <INPUT NAME="Q00*" SIZE=2><BR>"          int
" <INPUT NAME="S01*" SIZE=4> <INPUT NAME="I01*" SIZE=4> iLayoutPayment(char *pszOutData, WORKER * pWorker)
SIZE=6>"          <INPUT NAME="Q01*" SIZE=2><BR>"          {
" <INPUT NAME="S02*" SIZE=4> <INPUT NAME="I02*" SIZE=4>          int iLen = 0;
          pWorker->uScreenId = SCREEN_PAYMENT;
SIZE=6>"          <INPUT NAME="Q02*" SIZE=2><BR>"          iLen = iLayoutFormHdr(pszOutData, "TPC-C Payment",
" <INPUT NAME="S03*" SIZE=4> <INPUT NAME="I03*" SIZE=4>          pWorker);
          iLen+=sprintf(pszOutData + iLen,
SIZE=6>"          <INPUT NAME="Q03*" SIZE=2><BR>"          "Date:<BR><BR>"
" <INPUT NAME="S04*" SIZE=4> <INPUT NAME="I04*" SIZE=4>          "Warehouse: %4.4d%%s%",
          pWorker->iWarehouseId,
SIZE=6>"          <INPUT NAME="Q04*" SIZE=2><BR>"          "District: <INPUT NAME="DID*" SIZE=2><BR>"
" <INPUT NAME="S05*" SIZE=4> <INPUT NAME="I05*" SIZE=4>          "SIZE=2><BR><BR><BR><BR><BR>"
          "Customer: <INPUT NAME="CID*" SIZE=4>"
SIZE=6>"          <INPUT NAME="Q05*" SIZE=2><BR>"          "Cust-Warehouse: <INPUT NAME="CWI*"
" <INPUT NAME="S06*" SIZE=4> <INPUT NAME="I06*" SIZE=4>          "Cust-District: <INPUT NAME="CDI*"
          SIZE=1><BR>"
SIZE=6>"          <INPUT NAME="Q06*" SIZE=2><BR>"          "Name:          <INPUT NAME="CLT*"
" <INPUT NAME="S07*" SIZE=4> <INPUT NAME="I07*" SIZE=4>          SIZE=16> "
          "Since:<BR>"
SIZE=6>"          <INPUT NAME="Q07*" SIZE=2><BR>"          "
          Credit:<BR>"
" <INPUT NAME="S08*" SIZE=4> <INPUT NAME="I08*" SIZE=4>          "
          Disc:<BR>"
          Phone:<BR>"
SIZE=6>"          <INPUT NAME="Q08*" SIZE=2><BR>"          "<BR>"
" <INPUT NAME="S09*" SIZE=4> <INPUT NAME="I09*" SIZE=4>          "Amount Paid:          $<INPUT NAME="HAM*"
          SIZE=7> "
SIZE=6>"          <INPUT NAME="Q09*" SIZE=2><BR>"          "New Cust Balance:<BR>"
" <INPUT NAME="S10*" SIZE=4> <INPUT NAME="I10*" SIZE=4>          "Credit Limit:<BR><BR>Cust-Data:
          <BR><BR><BR><BR><PRE><HR>"
SIZE=6>"          <INPUT NAME="Q10*" SIZE=2><BR>"          "
          <INPUT TYPE="submit"NAME="CMD"
" <INPUT NAME="S11*" SIZE=4> <INPUT NAME="I11*" SIZE=4>          VALUE="Execute">"
          <INPUT TYPE="submit"NAME="CMD"
SIZE=6>"          <INPUT NAME="Q11*" SIZE=2><BR>"          "
          <INPUT TYPE="submit"NAME="CMD"
" <INPUT NAME="S12*" SIZE=4> <INPUT NAME="I12*" SIZE=4>          VALUE="Menu">"
          "</FORM>", szEndHtmlTag);
SIZE=6>"          <INPUT NAME="Q12*" SIZE=2><BR>"          return(iLen);
" <INPUT NAME="S13*" SIZE=4> <INPUT NAME="I13*" SIZE=4>          }
          /**
          ** iLayoutDelivery() produces the HTML code for the Delivery entry screen
          **
          ** It is called by uiServiceRequest().
          **/
          int
          iLayoutDelivery(char *pszOutData, WORKER * pWorker)
          {
          int iLen = 0;
          pWorker->uScreenId = SCREEN_DELIVERY;
          iLen = iLayoutFormHdr(pszOutData, "TPC-C Delivery",
          pWorker);
          iLen+=sprintf(pszOutData + iLen,
          <PRE>          Delivery<BR>"
          "Warehouse: %4.4d%%s%",
          pWorker->iWarehouseId,
          <BR><BR>"
          "Carrier Number: <INPUT NAME="OCD*"
          SIZE=2><BR><BR>"
          "Execution Status:<BR><PRE><HR>"
          "Total:<BR><HR>"
          <INPUT TYPE="submit"NAME="CMD"
          VALUE="Execute">"
          <INPUT TYPE="submit"NAME="CMD"
          VALUE="Menu">"
          "</FORM>", szEndHtmlTag);
          return(iLen);
          }
          /**
          ** iLayoutPayment() produces the HTML code for the Payment entry screen
          **
          ** It is called by uiServiceRequest().
          **/
          int
          iLayoutPayment(char *pszOutData, WORKER * pWorker)
          {
          int iLen = 0;
          pWorker->uScreenId = SCREEN_PAYMENT;
          iLen = iLayoutFormHdr(pszOutData, "TPC-C Payment",
          pWorker);
          iLen+=sprintf(pszOutData + iLen,
          <PRE>          Payment<BR>"
          "Date:<BR><BR>"
          "Warehouse: %4.4d%%s%",
          pWorker->iWarehouseId,
          "District: <INPUT NAME="DID*" SIZE=2><BR>"
          "SIZE=2><BR><BR><BR><BR><BR>"
          "Customer: <INPUT NAME="CID*" SIZE=4>"
          "Cust-Warehouse: <INPUT NAME="CWI*"
          "Cust-District: <INPUT NAME="CDI*"
          SIZE=1><BR>"
          "Name:          <INPUT NAME="CLT*"
          SIZE=16> "
          "Since:<BR>"
          "
          Credit:<BR>"
          Disc:<BR>"
          Phone:<BR>"
          "<BR>"
          "Amount Paid:          $<INPUT NAME="HAM*"
          SIZE=7> "
          "New Cust Balance:<BR>"
          "Credit Limit:<BR><BR>Cust-Data:
          <BR><BR><BR><BR><PRE><HR>"
          "
          <INPUT TYPE="submit"NAME="CMD"
          VALUE="Execute">"
          <INPUT TYPE="submit"NAME="CMD"
          VALUE="Menu">"
          "</FORM>", szEndHtmlTag);
          return(iLen);
          }
          /**
          ** iLayoutDelivery() produces the HTML code for the Delivery entry screen
          **
          ** It is called by uiServiceRequest().
          **/
          int
          iLayoutDelivery(char *pszOutData, WORKER * pWorker)
          {
          int iLen = 0;
          pWorker->uScreenId = SCREEN_DELIVERY;
          iLen = iLayoutFormHdr(pszOutData, "TPC-C Delivery",
          pWorker);
          iLen+=sprintf(pszOutData + iLen,
          <PRE>          Delivery<BR>"
          "Warehouse: %4.4d%%s%",
          pWorker->iWarehouseId,
          <BR><BR>"
          "Carrier Number: <INPUT NAME="OCD*"
          SIZE=2><BR><BR>"
          "Execution Status:<BR><PRE><HR>"
          "Total:<BR><HR>"
          <INPUT TYPE="submit"NAME="CMD"
          VALUE="Execute">"
          <INPUT TYPE="submit"NAME="CMD"
          VALUE="Menu">"
          "</FORM>", szEndHtmlTag);
          return(iLen);
          }
          /**
          ** iLayoutStockLevel() produces the HTML code for the Stock Level entry
          ** screen
          **
          ** It is called by uiServiceRequest().
          **/
          int
          iLayoutStockLevel(char *pszOutData, WORKER * pWorker)
          {
          int iLen = 0;
          pWorker->uScreenId = SCREEN_STOCKLEVEL;
          iLen = iLayoutFormHdr(pszOutData, "TPC-C Stock Level",
          pWorker);
          iLen+=sprintf(pszOutData + iLen,

```

```

" <PRE>                Stock-Level<BR>"
"Warehouse: %4.4d District: %2.2d%$s",
pWorker->iWarehouseId, pWorker->iDistrictId,
" <BR><BR>"
"Stock Level Threshold: <INPUT
NAME="THR*"SIZE=2><BR><BR>"
"low stock:  <BR><HR>"
"<INPUT TYPE="submit"NAME="CMD"
VALUE="Execute">"
"<INPUT TYPE="submit"NAME="CMD"
VALUE="Menu">"
" </FORM>", szEndHtmlTag);
return(iLen);
}

/**
** iLayoutFormHdr() generates a string of pre-formatted HTML including
hidden
** tags containing warehouse, district, screen, and terminal information
** for a new form to be filled in by an emulated user.
**
** This function is called by the iLayout<transaction>() functions, i.e.,
** it is used to create entry forms for all of the transaction screens.
**/
int
iLayoutFormHdr(char *pszOutData, char *pszHeader, WORKER * pWorker) }
{
    int iLen = 0;
    iLen = sprintf(pszOutData,
        "<HTML><HEAD><TITLE>%s</TITLE></HEAD>"
        "<FORM ACTION="%"s" METHOD="POST">"
        "<INPUT TYPE="hidden" NAME="PI*"
VALUE="\">"
        "<INPUT TYPE="hidden" NAME="STID"
VALUE="0\">"
        "<INPUT TYPE="hidden" NAME="SCID"
VALUE="%"d\">"
        "<INPUT TYPE="hidden" NAME="TRID"
VALUE="%"d\">"
        "<INPUT TYPE="hidden" NAME="WKID"
VALUE="%"d\">"
        "<INPUT TYPE="hidden" NAME="w_id"
VALUE="%"d\">"
        "<INPUT TYPE="hidden" NAME="d_id"
VALUE="%"d\">",
        pszHeader, szPluginPath, pWorker->uScreenId,
        pWorker->iTerminalId, pWorker->iWorkerId,
        pWorker->iWarehouseId, pWorker->iDistrictId);
return(iLen);
}

/**
** iLayoutRespHdr() generates a string of pre-formatted HTML including
hidden
** tags containing warehouse, district, screen, and terminal information
** for a response to a submitted form from an emulated user.
**
** This function is called by the bExecute<transaction>() functions.
*/
int
iLayoutRespHdr(char *pszOutData, char *pszHeader, WORKER * pWorker)
{
    int iLen = 0;
    iLen = sprintf(pszOutData,
        "<HTML><HEAD><TITLE>%s</TITLE></HEAD>"
        "<FORM ACTION="%"s" METHOD="POST">"
        "<INPUT TYPE="hidden" NAME="w_id"
VALUE="%"d\">"
        "<INPUT TYPE="hidden" NAME="d_id"
VALUE="%"d\">"
        "<INPUT TYPE="hidden" NAME="STID"
VALUE="%"d\">"
        "<INPUT TYPE="hidden" NAME="SCID"
VALUE="%"d\">"
        "<INPUT TYPE="hidden" NAME="TRID"
VALUE="%"d\">"
        "<INPUT TYPE="hidden" NAME="WKID"
VALUE="%"d\">",
        pszHeader, szPluginPath, pWorker->uScreenId,
        pWorker->iTerminalId, pWorker->iWorkerId,
        pWorker->iWarehouseId, pWorker->iDistrictId);
return(iLen);
}

while(uiLen--)
{
    *pszOutData++ = ' ';
    iLen++;
}
*pszOutData = 0;
return(iLen);
}

int iLen = 0;
while(*pszPicture)
{
    if(*pszPicture == 'X')
    {
        if(*pszInData)
            *pszOutData++ = *pszInData+
        else
            *pszOutData++ = ' ';
    } else
        *pszOutData++ = *pszPicture;
    pszPicture++;
    iLen++;
}
*pszOutData = 0;
return(iLen);
}

void vStringCopy(char *pszDest, char *pszSrc, int iBytes)
{
    strncpy(pszDest, pszSrc, iBytes);
}

```

```

        pszDest[iBytes] = '\0';
        return;
    }
    /**
    ** bIsNumeric checks to make sure a string contains a numeric value.
    **
    ** Called from pszExtractLongKey(), et al.
    **/
    BOOL bIsNumeric(char *pszNum)
    {
        if(*pszNum == 0)
            return(TRUE);
        if (pszNum[0] == '-') {
            pszNum++;
        }
        while(*pszNum && isdigit(*pszNum))
            pszNum++;
        return(*pszNum);
    }

    /**
    ** vToUpper -- simple function to convert string to all upper-case letters.
    **
    ** Called from pszExtractStringKey() to make all key parsing consistent.
    **/
    void vToUpper(char *szArg)
    {
        char *szPtr = szArg;
        for(; *szPtr; szPtr++)
        {
            *szPtr = toupper(*szPtr);
        }
    }

tpccService.h
#ifndef __TPCCSERVICEH__
#define __TPCCSERVICEH__
#include "tpcc.h"
uint uiServiceRequest(WORKER *, char *, char *, int *);
#endif

tpccBool.h
#ifndef __TPCCBOOLH__
#define __TPCCBOOLH__
#ifdef TRUE
#undef TRUE
#endif
#ifdef FALSE
#undef FALSE
#endif
typedef enum
    {
        FALSE = 0,
        TRUE = 1
    } BOOL;
#endif

tpccConst.h
#ifndef __TPCCCONSTH__
#define __TPCCCONSTH__
/* ServiceRequest return codes*/
#define SEND 1
#define SENDCLOSE 2

#define NEWORDER "NEWO"
#define PAYMENT "PAYM"
#define ORDERSTATUS "ORDS"
#define STOCKLEVEL "STOCK"
#define DELIVERY "DEL"

/* TPCC Service return codes*/
#define SVC_BADITEMID 1
#define SVC_NOERROR 0
#define SVCERR_DEADLOCK -1
#define SVCERR_NOCUSTOMER -2
#define SVCERR_NOORDERS -3
#define SVCERR_DBLIB -4

/* Min/Max/Len Data Definitions*/
#define MIN_DID 1
#define MAX_DID 10
#define MIN_OL 5
#define MAX_OL 15
#define MIN_QUANTITY 1
#define MAX_QUANTITY 10
#define MIN_ITEM_ID 1
#define MAX_ITEM_ID 100000
#define MIN_CUST_ID 1
#define MAX_CUST_ID 3000
#define MIN_CARRIER 1
#define MAX_CARRIER 10
#define MIN_THRESHOLD 10
#define MAX_THRESHOLD 20
#define MAX_MSG_SZ 4096 /*Should Be Equal To Net Buf Size*/
#define MAX_DIAG_SZ 2000
#define STATUS_LEN 200
#define NAME_LEN 16
#define ADDR_LEN 20
#define ZIP_LEN 10
#define STATE_LEN 2

/* pTPCC->iStatusId codes*/
#define INVALID_ITEM_ID 1
#define STATUS_OK 0
#define ERR_CMD_UNKNOWN -10
#define ERRTXT_CMD_UNKNOWN "Unrecognized Command"
#define ERR_ALREADY_LOGGEDIN -11
#define ERRTXT_ALREADY_LOGGEDIN "Already Logged In"
#define ERR_TRID -12
#define ERRTXT_TRID "Terminal Id or Worker Id Error"
#define ERR_SCREEN_UNKNOWN -13
#define ERRTXT_SCREEN_UNKNOWN "Unrecognized Screen Id"
#define ERR_WID_INVALID -14
#define ERR_DID_INVALID -15
#define ERR_MISSING_KEY -16
#define ERR_NOT_NUMERIC -17

#define ERR_THRESHOLD_RANGE -18
#define ERR_EMBEDDED_EMPTY_OL -19
#define ERR_QUANTITY_INVALID -20
#define ERR_OL_INVALID -21
#define ERR_OL_COUNT -22
#define ERR_TUX_INTERFACE -23
#define ERR_SERVICE_RSLT -24
#define ERR_INPUT_TOO_LONG -25
#define ERR_IDANDNAME_EMPTY -26
#define ERR_IDANDNAME_ENTERED -27
#define ERR_AMOUNT_BADFORM -28
#define ERR_AMOUNT_INVALID -29
#define ERR_CARRIER_INVALID -30
#define ERR_CID_INVALID -31
#define ERR_MANDATORY_FIELD -32

/* Severity level of diagnostic report */
#define DIAG_FORCE 1
#define DIAG_ERROR 2
#define DIAG_STATE 3
#define DIAG_INFO 4

/* Environment variable defaults */
#define DEFAULTDIAGLEVEL DIAG_INFO
#define DEFAULTTEVENTLOG 0
#define DIAGNOSTICS TRUE
#endif

tpccData.h
#ifndef __TPCCH__
#define __TPCCH__
#include <atmi.h> /* TUXEDO */
#include <Uunix.h> /* TUXEDO */
#include <userlog.h> /* TUXEDO */

typedef struct
    {
        int iWarehouseId; /* TPCC WareHouse Id*/
        int iDistrictId; /* TPCC District Id*/
        int iWorkerId; /* TPCC Worker Thread Id*/
        int iTerminalId; /* TPCC Terminal Id*/
        uint uScreenId; /* TPCC Screen Id*/
        int iStatusId; /* TPCC Status Id*/
        BOOL bRequestForm; /* TPCC Form Requested*/
        char szErrorTxt[200]; /* Error Text*/
        char szWork[100]; /* Thread Work Area*/
        char *pTuxInData; /* Tux Buffer Area*/
        char *pTuxOutData; /* Tux Buffer Area*/
        long lTuxDataLen; /* Tux Buffer Len*/
        TPINIT *pTpInf; /* TUXEDO MULTICONTEXT*/
    } WORKER;

#endif

tpcc.h
/**
*** tpcc.h
*** (c) 2006 Sun Microsystems, Inc. All rights reserved.

```

```

***
*** For iWS/Tuxedo client definitions. Must be in synch
*** with ../vendor/oracle/multi-svrs
***
***

```

```

#ifndef __TPCCH__
#define __TPCCH__
#include <atmi.h> /* TUXEDO */
#include <Unix.h> /* TUXEDO */
#include <userlog.h> /* TUXEDO */
#include "tpccBool.h"
#include "tpccConst.h"

```

```

typedef struct items_inf
{
    int ol_supply_w_id;
    int ol_i_id;
    char ol_i_name[25];
    int ol_quantity;
    int ol_stock;
    char ol_brand_generic[2];
    double ol_i_price;
    double ol_amount;
} ITEMS_INF;

```

```

typedef struct newo_inf
{
    int w_id;
    int d_id;
    int c_id;
    int o_id;
    int o_ol_cnt;
    double c_discount;
    double w_tax;
    double d_tax;
    char o_entry_d[20];
    char c_credit[3];
    char c_last[17];
    ITEMS_INF o_ol[MAX_OL];
    char status[25];
    double total_amount;
} NEWO_INF;

```

```

typedef struct pay_inf
{
    int w_id;
    int d_id;
    int c_id;
    int c_w_id;
    int c_d_id;
    double h_amount;
    double c_credit_lim;
    double c_balance;
    double c_discount;
    char h_date[20];
    char w_street_1[21];
    char w_street_2[21];
    char w_city[21];
    char w_state[3];
    char w_zip[11];
}

```

```

char d_street_1[21];
char d_street_2[21];
char d_city[21];
char d_state[3];
char d_zip[11];
char c_first[17];
char c_middle[3];
char c_last[17];
char c_street_1[21];
char c_street_2[21];
char c_city[21];
char c_state[3];
char c_zip[11];
char c_phone[17];
char c_since[11];
char c_credit[3];

```

```

char c_data[201];
} PAY_INF;
typedef struct ord_itm_inf
{
    int ol_supply_w_id;
    int ol_i_id;
    int ol_quantity;
    double ol_amount;
    char ol_delivery_d[11];
}

```

```

} ORD_ITM_INF;
typedef struct ord_inf
{
    int o_ol_cnt;
    int w_id;
    int d_id;
    int c_id;
    int o_id;
    int o_carrier_id;
    double c_balance;
    char c_first[17];
    char c_middle[3];
    char c_last[17];
    char o_entry_d[20];
    ORD_ITM_INF s_ol[MAX_OL];
}

```

```

} ORD_INF;
typedef struct del_inf
{
    int w_id;
    int o_carrier_id;
    char status[26];
} DEL_INF;
/* Structure used to queue delivery transaction */
typedef struct req_del
{
    int w_id;
    int o_carrier_id;
    time_t qtime; /* Time transaction was queued */
} REQ_DEL;
typedef struct stock_inf
{
    int w_id;
    int d_id;
}

```

```

int thresh_hold;
int low_stock;
} STOCK_INF;

```

```

typedef struct
{
    unsigned int uiThrId; /* thread ID */
    int iWarehouseId; /* TPCC WareHouse Id */
    int iDistrictId; /* TPCC District Id */
    int iWorkerId; /* TPCC Worker Thread Id */
    int iTerminalId; /* TPCC Terminal Id */
    uint uScreenId; /* TPCC Screen Id */
    int iStatusId; /* TPCC Status Id */
    BOOL bRequestForm; /* TPCC Form Requested */
    char szErrorTxt[200]; /* Error Text */
    char szWork[100]; /* Thread Work Area */
    char szRequestData[16384];
    char szReturnData[24576]; /* was 16384 */
    char *pTuxInData; /* Tux Buffer Area */
    char *pTuxOutData; /* Tux Buffer Area */
    long lTuxDataLen; /* Tux Buffer Len */
    TPINIT *pTpInf; /* TUXEDO MULTICONTEXT */
    TPCONTEXT_T tpxContext; /* Tuxedo Context */
} WORKER;

```

```

WORKER *pGlobalWorkerArray;

```

```

#endif

```

## Multi\_svr

### ora\_errrpt.c

```

/*
 * Copyright (c) 1995 by Sun Microsystems, Inc.
 */

#pragma ident "@(#)ora_errrpt.c 1.1      95/09/14  SMI"

/*
 * these functions actually belong in ~dbbench/generic/c/msggh_log.c. We put
 * them
 * here because they have database specific statements.
 */

#include "ora_err.h"
#include "ora_oci.h"

errrpt(lda, cur, sqlvar)
ldadef *lda;
csrdef *cur;
text *sqlvar;
{
    text msg[2048];
    /* if (cur->rc) { */
        oerhms(lda, (sb2) cur->rc, msg, 2048);
        userlog("%s sql_variable %s\n", msg, sqlvar);
    }
}

```



```

if (cur->rc == DEADLOCK || (cur->rc ==
SNAPSHOT_TOO_OLD))
    return(RECOVER);
else
    return(IRRECERR);
/*      */
}
}

/* vmm313 void ocierror(fname, lineno, errhp, status) */
int ocierror(fname, lineno, errhp, status)
char *fname;
int lineno;
OCIError *errhp;
sword status;
{
text errbuf[512];
ub4 buflen;
sb4 errcode;
sb4 lstat;
ub4 recno=2;

switch (status) {
case OCI_SUCCESS:
    break;
case OCI_SUCCESS_WITH_INFO:
    (void) userlog("Module %s Line %d\n", fname, lineno);
    (void) userlog("Error - OCI_SUCCESS_WITH_INFO\n");
    break;
case OCI_NEED_DATA:
    (void) userlog("Module %s Line %d\n", fname, lineno);
    (void) userlog("Error - OCI_NEED_DATA\n");
    return (IRRECERR);
case OCI_NO_DATA:
    /*
    (void) userlog("Module %s Line %d\n", fname, lineno);
    (void) userlog("Error - OCI_NO_DATA\n");
    */
    return IRRECERR; /* for 8.1.4 */
break;
case OCI_ERROR:
    lstat = OCIErrorGet (errhp, (ub4) 1,
        (text *) NULL, &errcode, errbuf,
        (ub4) sizeof(errbuf), OCI_HTYPE_ERR);
    if (errcode == NOT_SERIALIZABLE) return (errcode);
    while (lstat != OCI_NO_DATA)
    {
        (void) userlog("Module %s Line %d\n", fname, lineno);
        (void) userlog("Error - %s\n", errbuf);
        lstat = OCIErrorGet (errhp, recno++, (text *) NULL, &errcode,
errbuf,
            (ub4) sizeof(errbuf), OCI_HTYPE_ERROR);
    }
    return (errcode);
break;
case OCI_INVALID_HANDLE:
    (void) userlog("Module %s Line %d\n", fname, lineno);
    (void) userlog("Error - OCI_INVALID_HANDLE\n");
break;
}

break;
case OCI_STILL_EXECUTING:
    (void) userlog("Module %s Line %d\n", fname, lineno);
    (void) userlog("Error - OCI_STILL_EXECUTE\n");
    return (IRRECERR);
case OCI_CONTINUE:
    (void) userlog("Module %s Line %d\n", fname, lineno);
    (void) userlog("Error - OCI_CONTINUE\n");
    return (IRRECERR);
default:
    (void) userlog("Module %s Line %d\n", fname, lineno);
    (void) userlog("Error - \n");
    return (IRRECERR);
}
return RECOVER;
}

ora_err.h
/*
* Copyright (c) 1994 by Sun Microsystems, Inc.
*/

#ifdef ORA_ERR_H
#define ORA_ERR_H

#pragma ident "@(#)ora_err.h 1.4 95/09/14 SMI"
/*
* this kludge is required because Oracle does not provide
* symbolic constants in a header file
*/

#define EDEADLOK 60
#define SQLNOTFOUND 1403
#define COLUMN_NULL -1405
#define EDUPLICATE -1
#define RECOVER  -10
#define IRRECERR -20
#define NOERR 111
#define DEL_ERROR -666
#define DEL_DATE_LEN 7
#define SQL_BUF_SIZE 8192

#endif ORA_ERR_H

tpcc_srv_del.c
/*
* Copyright (c) 1994 by Sun Microsystems, Inc.
*/
#pragma ident "@(#)tpcso_srv_del.pc 1.5 94/12/07 SMI"

/
=====+
| Copyright (c) 1996 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
+=====

| FILENAME
| pldel.c
| DESCRIPTION
| OCI version of DELIVERY transaction in TPC-C benchmark.
+=====*/
#ifdef ORA_TPCC
#define ORA_TPCC
#include "tpcc.h"
#endif

#include <stdlib.h>
#include <unistd.h>
#include <sys/signal.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>
#include <sys/utsname.h>
#include <errno.h>
#include <stdio.h>
/* #include "ora_err.h" */

/* Tuxedo */
#include "atmi.h"
#include "userlog.h"

#define MOVETO(element, struct_name) element = struct_name -> element
#define MOVEBACK(element, struct_name) struct_name -> element =
element

#include "tpccflags.h"

#ifdef IS0
#define IS0T0 "SELECT substr(value,1,5) FROM v$parameter \
WHERE name = 'instance_number'"
#endif

#define SQLTXT "BEGIN inittpc.init_del; END;"

#define SQLTXT1 "DELETE FROM nord WHERE no_d_id = :d_id \
AND no_w_id = :w_id and rownum <= 1 \
RETURNING no_o_id into :o_id"

#define SQLTXT3 "UPDATE ordr SET o_carrier_id = :carrier_id \
WHERE o_id = :o_id and o_d_id = :d_id and o_w_id = :w_id \
returning o_c_id into :o_c_id"

#define SQLTXT4 "UPDATE ordl \
SET ol_delivery_d = :cr_date \
WHERE ol_w_id = :w_id AND ol_d_id = :d_id AND ol_o_id = :o_id \
RETURNING sum(ol_amount) into :ol_amount"

#define SQLTXT6 "UPDATE cust SET c_balance = c_balance + :amt, \
c_delivery_cnt = c_delivery_cnt + 1 WHERE c_w_id = :w_id AND \
c_d_id = :d_id AND c_id = :c_id"

```

```

#define NDISTS 10
#define ROWIDLEN 20

struct delctx {
    sb2 del_o_id_ind[NDISTS];
    sb2 d_id_ind[NDISTS];
    sb2 c_id_ind[NDISTS];
    sb2 del_date_ind[NDISTS];
    sb2 carrier_id_ind[NDISTS];
    sb2 amt_ind[NDISTS];
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    sb2 inum_ind;
#endif

    ub4 del_o_id_len[NDISTS];
    ub4 c_id_len[NDISTS];
    int oid_ctx;
    int cid_ctx;
    OCIBind *olamt_bp;

    ub2 w_id_len[NDISTS];
    ub2 d_id_len[NDISTS];
    ub2 del_date_len[NDISTS];
    ub2 carrier_id_len[NDISTS];
    ub2 amt_len[NDISTS];
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    ub2 inum_len;
#endif

    ub2 del_o_id_rcode[NDISTS];
    ub2 cons_rcode[NDISTS];
    ub2 w_id_rcode[NDISTS];
    ub2 d_id_rcode[NDISTS];
    ub2 c_id_rcode[NDISTS];
    ub2 del_date_rcode[NDISTS];
    ub2 carrier_id_rcode[NDISTS];
    ub2 amt_rcode[NDISTS];
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    ub2 inum_rcode;
#endif
    int del_o_id[NDISTS];
    int del_d_id[NDISTS];
    int cons[NDISTS];
    int w_id[NDISTS];
    int d_id[NDISTS];
    int c_id[NDISTS];
    int carrier_id[NDISTS];
    int amt[NDISTS];
    ub4 del_o_id_rcnt;
    int retry;
    int retry_serial;
    int retry_snapshot;
    OCIRowid *no_rowid_ptr[NDISTS];
    OCIRowid *o_rowid_ptr[NDISTS];
    OCIDate del_date[NDISTS];
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
}

char inum[10];
#endif
OCISmt *curd0;
OCISmt *curd1;
OCISmt *curd2;
OCISmt *curd3;
OCISmt *curd4;
OCISmt *curd5;
OCISmt *curd6;
OCISmt *curdtest;

OCIBind *w_id_bp;
OCIBind *w_id_bp3;
OCIBind *w_id_bp4;
OCIBind *w_id_bp5;
OCIBind *w_id_bp6;
OCIBind *d_id_bp;
OCIBind *d_id_bp3;
OCIBind *d_id_bp4;
OCIBind *d_id_bp6;
OCIBind *o_id_bp;
OCIBind *cr_date_bp;
OCIBind *c_id_bp;
OCIBind *c_id_bp3;
OCIBind *no_rowid_bp;
OCIBind *carrier_id_bp;
OCIBind *o_rowid_bp;
OCIBind *del_o_id_bp;
OCIBind *del_o_id_bp3;
OCIBind *amt_bp;
OCIBind *bstr1_bp[10];
OCIBind *bstr2_bp[10];
OCIBind *retry_bp;
OCIBind *retry_serial_bp;
OCIBind *retry_snapshot_bp;
OCIDefine *inum_dp;
OCIDefine *d_id_dp;
OCIDefine *del_o_id_dp;
OCIDefine *no_rowid_dp;
OCIDefine *c_id_dp;
OCIDefine *o_rowid_dp;
OCIDefine *cons_dp;
OCIDefine *amt_dp;

int norow;
};

typedef struct delctx delctx;
struct pldelctx {

    ub2 del_d_id_len[NDISTS];
    ub2 del_o_id_len[NDISTS];

    ub2 w_id_len;
    ub2 d_id_len[NDISTS];
    ub2 o_c_id_len[NDISTS];
    ub2 sums_len[NDISTS];

    ub2 carrier_id_len;
    ub2 ordcnt_len;
    ub2 del_date_len;
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    ub2 inum_len;
#endif

    int del_o_id[NDISTS];
    int del_d_id[NDISTS];
    int o_c_id[NDISTS];
    int sums[NDISTS];
    OCIDate del_date;
    int carrier_id;
    int ordcnt;

    ub4 del_o_id_rcnt;
    ub4 del_d_id_rcnt;
    ub4 o_c_id_rcnt;
    ub4 sums_rcnt;

    int retry;
    int retry_serial;
    int retry_snapshot;
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    char inum[10];
#endif
    OCISmt *curp1;
    OCISmt *curp2;
    OCIBind *w_id_bp;
    OCIBind *d_id_bp;
    OCIBind *o_id_bp;
    OCIBind *o_c_id_bp;
    OCIBind *ordcnt_bp;
    OCIBind *sums_bp;
    OCIBind *del_date_bp;
    OCIBind *carrier_id_bp;
    OCIBind *retry_bp;
    OCIBind *retry_serial_bp;
    OCIBind *retry_snapshot_bp;

    int norow;

};
typedef struct pldelctx pldelctx;

static pldelctx *pldctx;

static delctx *dctx;

#ifdef DMLRETDEL
struct amtctx {
    int ol_amt[NITEMS];
    sb2 ol_amt_ind[NITEMS];
    ub4 ol_amt_len[NITEMS];
    ub2 ol_amt_rcode[NITEMS];
    int ol_cnt;
};

```

```

typedef struct amtctx amtctx;
amtctx *actx;

#endif

/* Global variables for delivery transaction */

static int w_id;
static int o_carrier_id;

/*static struct msgh_req message;*/ /* Transaction message */
int my_qid, my_id;
char my_name[] = "Del";

static int tx_count = 0; /* Transaction counter */
static FILE *delfile;
OCIDate cr_date;
int del_o_id[10];
int retries;
int retries_serial;
int retries_recoverr;
int retries_snapshot;

struct msgh_req message;

static char outbuf[2048]; /* Buffer for results file */

#ifdef DMLRETDEL
sb4 no_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
            dvoid **bufpp, ub4 *alenp, ub1 *piecep,
            dvoid **indpp)
{
    *bufpp = (dvoid*)0;
    *alenp = 0;
    *indpp = (dvoid*)0;
    *piecep = OCI_ONE_PIECE;
    return (OCI_CONTINUE);
}

sb4 TPC_oid_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
                dvoid **bufpp, ub4 *alenp, ub1 *piecep,
                dvoid **indpp, ub2 **rcodepp)
{
    *bufpp = &dctx->del_o_id[iter];
    *indpp = &dctx->del_o_id_ind[iter];
    dctx->del_o_id_len[iter]=sizeof(dctx->del_o_id[0]);
    *alenp = &dctx->del_o_id_len[iter];
    *rcodepp = &dctx->del_o_id_rcode[iter];
    *piecep = OCI_ONE_PIECE;
    return (OCI_CONTINUE);
}

sb4 cid_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
             dvoid **bufpp, ub4 *alenp, ub1 *piecep,
             dvoid **indpp, ub2 **rcodepp)
{
    *bufpp = &dctx->c_id[iter];
    *indpp = &dctx->c_id_ind[iter];
    dctx->c_id_len[iter]=sizeof(dctx->c_id[0]);
    *alenp = &dctx->c_id_len[iter];
    *rcodepp = &dctx->c_id_rcode[iter];
    *piecep = OCI_ONE_PIECE;
    return (OCI_CONTINUE);
}

sb4 amt_data(dvoid *ctxp, OCIBind *bp, ub4 iter, ub4 index,
             dvoid **bufpp, ub4 *alenp, ub1 *piecep,
             dvoid **indpp, ub2 **rcodepp)
{
    amtctx *actx;
    actx =(amtctx*)ctxp;
    *bufpp = &actx->ol_amt[index];
    *indpp = &actx->ol_amt_ind[index];
    actx->ol_amt_len[index]=sizeof(actx->ol_amt[0]);
    *alenp = &actx->ol_amt_len[index];
    *rcodepp = &actx->ol_amt_rcode[index];
    *piecep = OCI_ONE_PIECE;
    return (OCI_CONTINUE);
}

#endif

int get_del_tx_cnt()
{
    return tx_count;
}

static int proc_no ;

int plsqliflag = 1;

int init_del_tx()
{
    /******
    * BEGIN BLOCK OF COMMON CODE
    ******
    */

    text stmbuf[SQL_BUF_SIZE];

    /* For all servers - common routine to open/init session etc. */
    TPCinit();

    if (plsqliflag)
    {
        pldctx = (pldelctx *) malloc (sizeof(pldelctx));
        DISCARD memset(pldctx,(char)0,(ub4)sizeof(pldelctx));
        /* Initialize */
        DISCARD OCIHandleAlloc(tpcenv, (dvoid**)&pldctx->curp1,
        OCI_HTYPE_STMT, 0,
        (dvoid**)0);
    }
}

DISCARD sprintf ((char *) stmbuf, SQLTXT);
DISCARD OCIStmtPrepare(pldctx->curp1, errhp, stmbuf,
                    (ub4) strlen((char *)stmbuf),
                    OCI_NTV_SYNTAX, OCI_DEFAULT);
DISCARD OCIERROR(errhp,
                OCIStmtExecute(tpcenv,pldctx->curp1,errhp,1,0,NULLP(OCISnapshot),
                NULLP(OCISnapshot), OCI_DEFAULT));

DISCARD OCIHandleAlloc(tpcenv,(dvoid**) &pldctx->curp2,
OCI_HTYPE_STMT,
0, (dvoid**)0);
sqlfile("tkvcpe1.sql",stmbuf);
DISCARD OCIStmtPrepare(pldctx->curp2, errhp, stmbuf,
                    (ub4)strlen((char *)stmbuf), OCI_NTV_SYNTAX,
OCI_DEFAULT);
OCIBNDPL(pldctx->curp2, pldctx->w_id_bp , errhp,":w_id",
        ADR(w_id), SIZ(int), SQLT_INT,&pldctx->w_id_len);
OCIBNDPL(pldctx->curp2, pldctx->ordcnt_bp , errhp,":ordcnt",
        ADR(pldctx->ordcnt), SIZ(int), SQLT_INT,&pldctx->ordcnt_len);
OCIBNDPL(pldctx->curp2, pldctx->del_date_bp,errhp,":now",
        ADR(pldctx->del_date), SIZ(OCIDate), SQLT_ODT,&pldctx->del_date_len);
OCIBNDPL(pldctx->curp2, pldctx->carrier_id_bp , errhp,
        ":carrier_id", ADR(o_carrier_id), SIZ(int),
        SQLT_INT, &pldctx->carrier_id_len);

OCIBNDPLA(pldctx->curp2, pldctx->d_id_bp, errhp,":d_id",
        pldctx->del_d_id, SIZ(int),SQLT_INT, pldctx->del_d_id_len,
        NDISTS, &pldctx->del_d_id_rcnt);
OCIBNDPLA(pldctx->curp2, pldctx->o_id_bp, errhp,":order_id",
        pldctx->del_o_id,SIZ(int),SQLT_INT, pldctx->del_o_id_len,NDISTS,
        &pldctx->del_o_id_rcnt);
OCIBNDPLA(pldctx->curp2, pldctx->sums_bp, errhp,"sums",
        pldctx->sums,SIZ(int),SQLT_INT, pldctx->sums_len,NDISTS,
        &pldctx->sums_rcnt);
OCIBNDPLA(pldctx->curp2, pldctx->o_c_id_bp, errhp,":o_c_id",
        pldctx->o_c_id,SIZ(int),SQLT_INT, pldctx->o_c_id_len,NDISTS,
        &pldctx->o_c_id_rcnt);
OCIBND(pldctx->curp2, pldctx->retry_bp , errhp,":retry",
        ADR(pldctx->retry), SIZ(int),SQLT_INT);
}
else
{
    dctx = (delctx *) malloc (sizeof(delctx));
    memset(dctx,(char)0,sizeof(delctx));
    dctx->norow = 0;
    actx = (amtctx *) malloc (sizeof(amtctx));
    memset(actx,(char)0,sizeof(amtctx));

#ifdef ISO || defined(ISO5) || defined(ISO6) || defined(ISO8)
OCIHandleAlloc(tpcenv, (dvoid **)&dctx->curd0, OCI_HTYPE_STMT,
0,
(dvoid**)0);
sprintf ((char *) stmbuf, SQLTXT0);
#endif
}
}

```

```

OCIStmtPrepare(dctx->curd0, errhp, stmbuf, strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);
/* open fourth cursor */

OCIDFNRA(dctx->curd0, dctx->inum_dp, errhp, 1, dctx->inum, SIZ(dctx-
>inum),
SQL_STR, &(dctx->inum_ind), &(dctx->inum_len), &(dctx-
>inum_rcode));
#endif
/* If PLSQDEL and ISO? are both defined, then they both try to use
curd0! This could cause a problem. Will try to fix later - VMM 12/30/97 */
/* bind variables */

OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd1), OCI_HTYPE_STMT,
0,
(dvoid**)0);
DISCARD sprintf((char *) stmbuf, "%s", SQLTXT1);
DISCARD OCIStmtPrepare(dctx->curd1, errhp, stmbuf,
strlen((char *)stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT);

OCIBND(dctx->curd1, dctx->w_id_bp, errhp, ":w_id", dctx->w_id, SIZ(int),
SQL_INT);
OCIBNDRA(dctx->curd1, dctx->d_id_bp, errhp, ":d_id", dctx-
>d_id, SIZ(int),
SQL_INT, NULL, NULL, NULL);

OCIBNDRAD(dctx->curd1, dctx->del_o_id_bp, errhp, ":o_id",
SIZ(int), SQL_INT, NULL,
&dctx->oid_ctx, no_data, TPC_oid_data);

/* open third cursor */

DISCARD OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd3),
OCI_HTYPE_STMT,
0, (dvoid**)0);
DISCARD sprintf((char *) stmbuf, SQLTXT3);
DISCARD OCIStmtPrepare(dctx->curd3, errhp, stmbuf, strlen((char
*)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);

/* bind variables */

OCIBNDRA(dctx->curd3, dctx->carrier_id_bp, errhp, ":carrier_id",
dctx->carrier_id, SIZ(dctx->carrier_id[0]), SQL_INT,
dctx->carrier_id_ind, dctx->carrier_id_len, dctx->carrier_id_rcode);

OCIBNDRA(dctx->curd3, dctx->w_id_bp3, errhp, ":w_id", dctx-
>w_id, SIZ(int),
SQL_INT, NULL, NULL, NULL);
OCIBNDRA(dctx->curd3, dctx->d_id_bp3, errhp, ":d_id", dctx-
>d_id, SIZ(int),
SQL_INT, NULL, NULL, NULL);
OCIBNDRA(dctx->curd3, dctx->del_o_id_bp3, errhp, ":o_id", dctx-
>del_o_id,
SIZ(int), SQL_INT, NULL, NULL, NULL);
OCIBNDRAD(dctx->curd3, dctx->c_id_bp3, errhp, ":o_c_id", SIZ(int),
SQL_INT, NULL, &dctx->cid_ctx, no_data, cid_data);

OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd4), OCI_HTYPE_STMT,
0,
(dvoid**)0);
DISCARD sprintf((char *) stmbuf, "%s", SQLTXT1);
DISCARD OCIStmtPrepare(dctx->curd4, errhp, stmbuf, strlen((char
*)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);

OCIBND(dctx->curd4, dctx->w_id_bp4, errhp, ":w_id", dctx->w_id,
SIZ(int), SQL_INT);
OCIBND(dctx->curd4, dctx->d_id_bp4, errhp, ":d_id", dctx->d_id,
SIZ(int), SQL_INT);
OCIBND(dctx->curd4, dctx->o_id_bp, errhp, ":o_id", dctx->del_o_id,
SIZ(int), SQL_INT);
OCIBND(dctx->curd4, dctx->cr_date_bp, errhp, ":cr_date", dctx->del_date,
SIZ(OCIDate), SQL_ODT);
OCIBNDRAD(dctx->curd4, dctx->olamt_bp, errhp, ":ol_amount",
SIZ(int), SQL_INT, NULL, actx, no_data, amt_data);

/* open sixth cursor */

DISCARD OCIHandleAlloc(tpcenv, (dvoid **)&(dctx->curd6),
OCI_HTYPE_STMT,
0, (dvoid**)0);
DISCARD sprintf((char *) stmbuf, SQLTXT6);
DISCARD OCIStmtPrepare(dctx->curd6, errhp, stmbuf, strlen((char
*)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);

/* bind variables */

OCIBND(dctx->curd6, dctx->amt_bp, errhp, ":amt", dctx->amt, SIZ(int),
SQL_INT);
OCIBND(dctx->curd6, dctx->w_id_bp6, errhp, ":w_id", dctx-
>w_id, SIZ(int),
SQL_INT);
OCIBND(dctx->curd6, dctx->d_id_bp6, errhp, ":d_id", dctx->d_id, SIZ(int),
SQL_INT);
OCIBND(dctx->curd6, dctx->c_id_bp, errhp, ":c_id", dctx->c_id, SIZ(int),
SQL_INT);
}
/*****
* END BLOCK OF COMMON CODE
*****/

/*proc_stat_msg("init_del_tx()n");
proc_stat(); */

return(0);
}

void shiftdata(int from)
{
int i;
for (i=from; i<NDISTS-1; i++)
{
dctx->del_o_id_ind[i] = dctx->del_o_id_ind[i+1];
dctx->del_o_id[i] = dctx->del_o_id[i+1];
dctx->w_id[i] = dctx->w_id[i+1];
dctx->d_id[i] = dctx->d_id[i+1];
dctx->carrier_id[i] = dctx->carrier_id[i+1];
}
}

/* Structure used to queue delivery transaction */
struct req_struct {
int w_id;
int o_carrier_id;
time_t qtime; /* Time transaction was queued */
};

int delivery_tx(TPSCVINFO *rqst)
{ /*dt */
int i, j, v;
int invalid;
int tmp_id;
int rpc, rcount, errcode, execstatus;
int count;
time_t etime;
struct timeval tv;

/* float tmp_amt; changed form float to int */
int tmp_amt;
int del_o_id[10];
ub4 attr_size;

int len;
int retries=0, err = 0;

struct req_struct *delp;
delp = (struct req_struct *) (rqst->data);

/*****
* BEGIN BLOCK OF COMMON CODE
*****/

/*int rpc, rcount, errcode, execstatus;*/
MOVETO(w_id, delp);
MOVETO(o_carrier_id, delp);

/*
vgetdate(cr_date); */

OCIERROR(errhp, OCIDateSysDate(errhp, &cr_date));

tx_count++;

sprintf(outbuf, "Starting transaction %d queued at %ld\n",

```

```

tx_count, delp->qtime);

#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
int hasno;
int reread;
char sdate[30];

OCISstmtExecute(tpcsvc,dctx->curd0,errhp,1,0,0,0,OCI_DEFAULT);
sysdate (sdate);
printf ("Delivery started at %s on node %s\n", sdate, dctx->inum);
#endif
if (plsqflflag)
{
    pldctx->w_id_len = sizeof(int);
    pldctx->carrier_id_len = sizeof(int);
    for (i = 0; i < NDISTS; i++)
    {
        pldctx->del_o_id_len[i] = sizeof(int);
        del_o_id[i] = 0;
    }
    pldctx->del_date_len = DEL_DATE_LEN;
    DISCARD memcpy(&pldctx->del_date,&cr_date,sizeof(OCIDate));

    pldctx->retry=0;
    pldctx->retry_serial=0;
    pldctx->retry_snapshot=0;

    DISCARD OCIERROR(errhp,
        OCISstmtExecute(tpcsvc,pldctx->curp2,errhp,1,0,NULLP(CONST
OCISnapshot),
        NULLP(OCISnapshot),OCI_DEFAULT));
    for (i = 0; i < NDISTS; i++)
    {
        del_o_id[i] = 0;
    }
    for (i = 0; i < pldctx->del_o_id_rcnt; i++)
        del_o_id[pldctx->del_d_id[i] - 1] = pldctx->del_o_id[i];
}
else
{
retry:
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    reread = 1;
#endif
#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
iso:
#endif
    invalid = 0;

    /* initialization for array operations */
}

for (i = 0; i < NDISTS; i++)
{
    /*F*/
    dctx->del_o_id_ind[i] = TRUE;
    dctx->d_id_ind[i] = TRUE;
    dctx->c_id_ind[i] = TRUE;
    dctx->del_date_ind[i] = TRUE;
    dctx->carrier_id_ind[i] = TRUE;
    dctx->amt_ind[i] = TRUE;

    dctx->del_o_id_len[i] = SIZ(dctx->del_o_id[0]);
    dctx->w_id_len[i] = SIZ(dctx->w_id[0]);
    dctx->d_id_len[i] = SIZ(dctx->d_id[0]);
    dctx->c_id_len[i] = SIZ(dctx->c_id[0]);
    dctx->del_date_len[i] = DEL_DATE_LEN;
    dctx->carrier_id_len[i] = SIZ(dctx->carrier_id[0]);
    dctx->amt_len[i] = SIZ(dctx->amt[0]);

    dctx->w_id[i] = w_id;
    dctx->d_id[i] = i+1;
    dctx->carrier_id[i] = o_carrier_id;
    memcpy(&dctx->del_date[i],&cr_date,sizeof(OCIDate));
}

memset(actx,(char)0,sizeof(amtctx));

/* array select from new_order and orders tables */

execstatus=OCISstmtExecute(tpcsvc,dctx->curd1,errhp,NDISTS,0,
    NULLP(CONST
OCISnapshot),NULLP(OCISnapshot),OCI_DEFAULT);
if((execstatus != OCI_SUCCESS) && (execstatus != OCI_NO_DATA))
{
    DISCARD OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE)
    {
        retries++;
        retries_serial++;
        goto retry;
    }
}
else if (errcode == RECOVERR)
{
    retries++;
    retries_recoverr++;
    goto retry;
}
else if (errcode == SNAPSHOT_TOO_OLD)
{
    retries++;
    retries_snapshot++;
    goto retry;
}
else
{
    return -1;
}

DISCARD OCIAttrGet(dctx-
>curd3,OCI_HTYPE_STMT,&rcount,NULLP(ub4),
    OCI_ATTR_ROW_COUNT,errhp);

if (rcount != rpc)
{
#ifdef TUX
    userlog ("Error in TPC-C server %d: %d rows selected, %d ords
updated\n",
        proc_no, rpc, rcount);
}
}
}

/* mark districts with no new order */
DISCARD OCIAttrGet(dctx-
>curd1,OCI_HTYPE_STMT,&rcount,NULLP(ub4),
    OCI_ATTR_ROW_COUNT,errhp);
rpc = rcount;
if (rcount != NDISTS )
{
    int j = 0;
    for (i=0;i < NDISTS; i++)
    {
        if (dctx->del_o_id_ind[j] == 0) /* there is data here */
            j++;
        else
            shiftdata(j);
    }
}

execstatus=OCISstmtExecute(tpcsvc,dctx->curd3,errhp,rpc,0,
    NULLP(CONST
OCISnapshot),NULLP(OCISnapshot),OCI_DEFAULT);
if(execstatus != OCI_SUCCESS)
{
    DISCARD OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE)
    {
        retries++;
        retries_serial++;
        goto retry;
    }
}
else if (errcode == RECOVERR)
{
    retries++;
    retries_recoverr++;
    goto retry;
}
else if (errcode == SNAPSHOT_TOO_OLD)
{
    retries++;
    retries_snapshot++;
    goto retry;
}
else
{
    return -1;
}

DISCARD OCIAttrGet(dctx-
>curd3,OCI_HTYPE_STMT,&rcount,NULLP(ub4),
    OCI_ATTR_ROW_COUNT,errhp);

if (rcount != rpc)
{
#ifdef TUX
    userlog ("Error in TPC-C server %d: %d rows selected, %d ords
updated\n",
        proc_no, rpc, rcount);
}
}
}

```

```

#else
    DISCARD fprintf(stderr,
                    "Error in TPC-C server %d: %d rows selected, %d ords updated\n",
                    proc_no, rpc, rcount);
#endif

/* array update of order_line table */
execstatus=OCISStmtExecute(tpcsvc,dctx->curd4,errhp,rpc,0,
    NULLP(CONST
OCISnapshot,NULLP(OCISnapshot),OCI_DEFAULT);
if(execstatus != OCI_SUCCESS)
{
    DISCARD OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE)
    {
        retries++;
        retries_serial++;
        goto retry;
    }
    else if (errcode == RECOVERR)
    {
        retries++;
        retries_recoverr++;
        goto retry;
    }
    else if (errcode == SNAPSHOT_TOO_OLD)
    {
        retries++;
        retries_snapshot++;
        goto retry;
    }
    else
    {
        return -1;
    }
}
DISCARD OCIAttrGet(dctx-
>curd4,OCI_HTYPE_STMT,&rcount,NULLP(ub4),
    OCI_ATTR_ROW_COUNT,errhp);
/* transfer amounts */
for (i=0;i<rpc;i++)
{
    dctx->amt[i]=0;
    if (actx->ol_amt_rcode[i] == 0)
    {
        dctx->amt[i] = actx->ol_amt[i];
    }
}
#endif
printf("d_id:amount\n");
for (i = 0; i < rpc; i++)
    printf("%d:%.2f ", dctx->d_id[i], (float)dctx->amt[i]/100);
printf("\n");
#endif

/* array update of customer table */
#if defined(ISO5) || defined(ISO6)
execstatus=OCISStmtExecute(tpcsvc,dctx->curd6,errhp,rpc,0,0,0,
    OCI_DEFAULT);
#else
execstatus=OCISStmtExecute(tpcsvc,dctx->curd6,errhp,rpc,0,
    NULLP(CONST OCISnapshot),NULLP(OCISnapshot),
    OCI_COMMIT_ON_SUCCESS | OCI_DEFAULT);
#endif
if(execstatus != OCI_SUCCESS)
{
    OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
    errcode = OCIERROR(errhp,execstatus);
    if(errcode == NOT_SERIALIZABLE)
    {
        retries++;
        retries_serial++;
        goto retry;
    }
    else if (errcode == RECOVERR)
    {
        retries++;
        retries_recoverr++;
        goto retry;
    }
    else if (errcode == SNAPSHOT_TOO_OLD)
    {
        retries++;
        retries_snapshot++;
        goto retry;
    }
    else
    {
        return -1;
    }
}
DISCARD OCIAttrGet(dctx-
>curd6,OCI_HTYPE_STMT,&rcount,NULLP(ub4),
    OCI_ATTR_ROW_COUNT,errhp);
if (rcount != rpc) {
#ifdef TUX
    userlog ("Error in TPC-C server %d: %d rows selected, %d cust
    updated\n",
            proc_no, rpc, rcount);
#else
    DISCARD fprintf(stderr,
                    "Error in TPC-C server %d: %d rows selected, %d cust updated\n",
                    proc_no, rpc, rcount);
#endif
    DISCARD OCITransRollback(tpcsvc, errhp, OCI_DEFAULT);
    return (-1);
}

/* return o_id's in district id order */

for (i = 0; i < NDISTS; i++)
    del_o_id[i] = 0;
for (i = 0; i < rpc; i++)
    del_o_id[dctx->d_id[i] - 1] = dctx->del_o_id[i];
}

for (i = 0; i < 10; i++) {
    if (del_o_id[i] == 0) {
        /* No order found for this district */
        sprintf(outbuf+strlen(outbuf),
            "Delivery for District %d skipped\n", i+1);
    }
    else {
        sprintf(outbuf+strlen(outbuf),
            "Delivered order %d for district %d, warehouse %d, carrier
            %d\n",
            del_o_id[i], i+1, w_id, o_carrier_id);
    }
}
(void)gettimeofday(&tv, NULL); /* convert to msec */
etime = (tv.tv_sec * 1000) + (tv.tv_usec / 1000);
sprintf(outbuf+strlen(outbuf), "Transaction completed at %ld Retries %d
%d %d %d\n",
    etime,retries, retries_serial,retries_recoverr, retries_snapshot);
fwrite(outbuf, strlen(outbuf), 1, delfile);
fflush(delfile);

/*****
* END BLOCK OF COMMON CODE
*****/
return(0);
}

void cleanup(int code)
{
    if (dctx)
        free (dctx);
}

#if defined(ISO) || defined(ISO5) || defined(ISO6) || defined(ISO8)
    OCIHandleFree((dvoid *)dctx->curd0,OCI_HTYPE_STMT);
#endif
OCIHandleFree((dvoid *)dctx->curd1,OCI_HTYPE_STMT);
OCIHandleFree((dvoid *)dctx->curd2,OCI_HTYPE_STMT);
OCIHandleFree((dvoid *)dctx->curd3,OCI_HTYPE_STMT);
OCIHandleFree((dvoid *)dctx->curd4,OCI_HTYPE_STMT);
OCIHandleFree((dvoid *)dctx->curd5,OCI_HTYPE_STMT);
OCIHandleFree((dvoid *)dctx->curd6,OCI_HTYPE_STMT);

/* log off */
OCIHandleFree((dvoid *)tpcusr, OCI_HTYPE_SESSION);
OCIHandleFree((dvoid *)tpcsvc, OCI_HTYPE_SVCCTX);
OCIHandleFree((dvoid *)errhp, OCI_HTYPE_ERROR);
OCIHandleFree((dvoid *)tpcsrv, OCI_HTYPE_SERVER);
OCIHandleFree((dvoid *)tpcenv, OCI_HTYPE_ENV);

```

```

exit(code);
}

/* Tuxedo */
int tpsvrinit(int argc, char **argv)
{
    char *p;
    char filename[200];
    int proc_no, count;
    struct utsnme name;
    if ((p = getenv("CLNT_HOST")) == (char *)NULL) {
        userlog("CLNT_HOST environment variable not set - use same
name as for an rsh i.e. X64_0\n");
        exit(1);
    }

    proc_no = atoi(argv[optind]); /* Needs argument which is the proc_no
*/

    /* Get hostname of our machine and create results file */
    /* uname( &name); */
    /* strcpy(filename, p); */

    /* sprintf(filename, "/tmp/%s.del%d", name.nodename, proc_no); */
    sprintf(filename, "/tmp/%s.del%d", p, proc_no);

    delfile = fopen(filename, "w");
    if (delfile == NULL) {
        userlog("Cannot create file %s\n", filename);
    }
    return(init_del_tx()); /* Prepare transaction */
}

void tpsvrdone()
{
    fclose(delfile); /* Close results file */
}

int DEL(TPSVCINFO *rqst)
{
    if (delivery_tx(rqst))
        tpreturn(TPFAIL, 0, rqst->data, sizeof(struct req_struct), 0);
    else
        tpreturn(TPSUCCESS, 0, rqst->data, sizeof(struct req_struct), 0);
}

tpcc_srv_init.c
/*
 * Copyright (c) 1994 by Sun Microsystems, Inc.
 */

#pragma ident "@(#)tpcso_srv_stock.c 1.6 95/04/12 SMI"

/
=====+
| Copyright (c) 1994 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
+=====+
/* common to all 5 */
OCIInitialize(OCI_DEFAULT|OCI_OBJECT,(dvoid *)0,0,0,0);
OCIEnvInit(&tpcenv, OCI_DEFAULT, 0, (dvoid **)0);
OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcsrv,
OCI_HTYPE_SERVER, 0, (dvoid **)0);
OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&errhp,
OCI_HTYPE_ERROR, 0, (dvoid **)0);
OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcscv,
OCI_HTYPE_SVCCTX, 0, (dvoid **)0);
OCIServerAttach(tpcsrv, errhp, (text *)0,0,OCI_DEFAULT);
OCIAttrSet((dvoid *)tpcscv, OCI_HTYPE_SVCCTX, (dvoid *)tpcsrv,
(ub4)0,OCI_ATTR_SERVER, errhp);
OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcusr,
OCI_HTYPE_SESSION, 0, (dvoid **)0);
OCIAttrSet((dvoid *)tpcusr, OCI_HTYPE_SESSION, (dvoid *)uid,
(ub4)strlen(uid),OCI_ATTR_USERNAME, errhp);
OCIAttrSet((dvoid *)tpcusr, OCI_HTYPE_SESSION, (dvoid *)pwd,
(ub4)strlen(pwd),
OCI_ATTR_PASSWORD, errhp);
fprintf(stderr, "%d] Now call SessionBegin UID=%s PWD=%s\n", getpid(),
uid, pwd);

OCIERROR(errhp, OCISessionBegin(tpcscv, errhp, tpcusr,
OCI_CRED_RDBMS, OCI_DEFAULT));

OCIAttrSet(tpcscv, OCI_HTYPE_SVCCTX, tpcusr, 0,
OCI_ATTR_SESSION, errhp);

/* run all transaction in serializable mode */

OCIHandleAlloc(tpcenv, (dvoid **)&curi, OCI_HTYPE_STMT, 0,
(dvoid **)0);
sprintf((char *) stmbuf, SQLTXT);
OCIStmtPrepare(curi, errhp, stmbuf, strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);
OCIERROR(errhp,OCIStmtExecute(tpcscv, curi,
errhp,1,0,0,0,OCI_DEFAULT));
OCIHandleFree(curi, OCI_HTYPE_STMT);
/* end common ---- */

return(0);
}

tpcc_srv_util.c
/*
 * Copyright (c) 1995 by Sun Microsystems, Inc.
 */

#pragma ident "@(#)tpcso_srv_util.c 1.17 97/01/02 SMI"

/
=====+
| Copyright (c) 1995 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
+=====+
/* Common utility functions used by all tpcso_srv* programs */

#include <stdio.h>
#include <stdlib.h>

```

```

#include <sys/types.h>
#include <sys/file.h>

#include "ora_oci.h"
#include "ora_err.h"

FILE *vopen(fnam,mode)
char *fnam;
char *mode;
{
FILE *fd;

#ifdef DEBUG
fprintf(stderr, "tkvuopen() fnam: %s, mode: %s\n", fnam, mode);
#endif

fd = fopen((char *)fnam,(char *)mode);
if (!fd){
fprintf(stderr, "fopen on %s failed %d\n",fnam,fd);
exit(-1);
}
return(fd);
}

/*
int sqlfile(fnam,linebuf)
char *fnam;
text *linebuf;
*/
int sqlfile (char *fnam, text *linebuf)
{
FILE *fd;
int nulpt = 0;

#ifdef DEBUG
fprintf(stderr, "sqlfile() fnam: %s, linebuf: %s\n", fnam, linebuf);
#endif

fd = vopen(fnam,"r");
while (fgets((char *)linebuf+nulpt, SQL_BUF_SIZE,fd))
{
nulpt = strlen((char *)linebuf);
}
return(nulpt);
}

void vgetdate (unsigned char *oradt)
{
struct tm *loctime;
time_t int_time;

struct ORADATE {
unsigned char century;
unsigned char year;
unsigned char month;
unsigned char day;
unsigned char hour;
unsigned char minute;
unsigned char second;
} Date;

int day,month,year;

memcpy(&Date,oradt,7);

year = (Date.century-100)*100 + Date.year-100;
month = Date.month;
day = Date.day;
sprintf(outdate,"%02d-%02d-%04d",day,month,year);

return;
}

void cvtdmy (unsigned char *oradt, char *outdate)
{
struct ORADATE {
unsigned char century;
unsigned char year;
unsigned char month;
unsigned char day;
unsigned char hour;
unsigned char minute;
unsigned char second;
} Date;

int day,month,year;

memcpy(&Date,oradt,7);

year = (Date.century-100)*100 + Date.year-100;
month = Date.month;
day = Date.day;
hour = Date.hour - 1;
min = Date.minute - 1;
sec = Date.second - 1;

sprintf(outdate,"%02d-%02d-%04d %02d:%02d:%02d",
day,month,year,hour,min,sec);

return;
}

ora_oci.h
#pragma ident "@(#)oci.h 1.1 95/09/14 SMI"

/
*=====+
| Copyright (c) 1994 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
+=====+
|FILENAME
| tpccpl.h

```



DESCRIPTION	ocierror(_FILE_,_LINE_,(errp), \	/* ocierror(_FILE_,_LINE_,(errp),
Header file for TPC-C transactions in PL/SQL.	OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_HTYPE_BIND,0,	OCIBindArrayOfStruct((bndp),(errp),(progv),sizeof((indp)[0]),\
=====	(dvoid**0)); \	sizeof((alen)[0]),sizeof((arcode)[0])); */
=====*/	ocierror(_FILE_,_LINE_, (errp), \	
#ifndef TPCCPL_H	OCIBindByName((stmp), &(bndp), (errp), \	#define OCIDEFINE(stmp,dfnp,errp,pos,progv,progv1,ftype)\
#define TPCCPL_H	strlen((sqlvar)), \	OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(progv),(progv1),(ftype),\
	(text*)(sqlvar), \	0,0,0,OCI_DEFAULT)
	(progv), (progv1), \	
#include <stdio.h>	(ftype),0,0,0,0,OCI_DEFAULT));	#define OCIDEF(stmp,dfnp,errp,pos,progv,progv1,ftype) \
#include <ctype.h>		OCIHandleAlloc((stmp),(dvoid*)&(dfnp),OCI_HTYPE_DEFINE,0,\
#include <string.h>	#define	(dvoid**0);\
	OCIBNDRAD(stmp,bndp,errp,sqlvar,progv1,ftype,indp,ctxp,cbf_nodata,cbf_	OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(progv),(progv1),\
#include <oratypes.h>	data) \	(ftype),NULL,NULL,NULL,OCI_DEFAULT); \
#include <oci.h>	ocierror(_FILE_,_LINE_,(errp), \	
****	OCIBindByName((stmp),&(bndp),(errp),(text*)(sqlvar), \	#define OCIDFNRA(stmp,dfnp,errp,pos,progv,progv1,ftype,indp,alen,arcode)
#if __STDC__	strlen((sqlvar)),0,(progv1),(ftype), \	\
#include <ociapr.h>	indp,0,0,0,0,OCI_DATA_AT_EXEC)); \	OCIHandleAlloc((tpcenv),(dvoid*)&(dfnp),OCI_HTYPE_DEFINE,0,\
#else	ocierror("yufei",_LINE_,(errp), \	(dvoid**0);\
#include <ocikpr.h>	OCIBindDynamic((bndp),(errp),(ctxp),(cbf_nodata),(ctxp),	OCIDefineByPos((stmp),&(dfnp),(errp),(pos),(progv),\
#endif	(cbf_data));	(progv1),(ftype),(indp),
****/		(alen),\
		(arcode),OCI_DEFAULT); \
typedef struct cda_def csrdef;	#define	
typedef struct cda_def ldatdef;	OCIBNDRA(stmp,bndp,errp,sqlvar,progv,progv1,ftype,indp,alen,arcode) \	#define OBNDRV(lda,cursor,sqlvar,progv,progv1,ftype)\
	ocierror(_FILE_,_LINE_,(errp), \	if (obndrv((cursor),(text*)(sqlvar),NA,(ub1*)(progv),(progv1),(ftype),NA,\
	OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_HTYPE_BIND,0,	(sb2)*0, (text*)0, NA, NA))\
#ifndef DISCARD	(dvoid**0)); \	{errrpt(lda,cursor,sqlvar);return(-1);}\
#define DISCARD (void)	ocierror(_FILE_,_LINE_,(errp), \	else\
#endif	OCIBindByName((stmp),&(bndp),(errp),(text *)	DISCARD 0
	(sqlvar),strlen((sqlvar)), \	
#ifndef sword	(progv),(progv1),(ftype),	
#define sword int	(indp),(alen),(arcode),0,0,OCI_DEFAULT));	#define OBNDRA(lda,cursor,sqlvar,progv,progv1,ftype,indp,alen,arcode)\
#endif	/* ocierror(_FILE_,_LINE_,(errp), \	if (obndra((cursor),(text*)(sqlvar),NA,(ub1*)(progv),(progv1),(ftype),NA,\
	OCIBindArrayOfStruct((bndp),(errp),(progv1),sizeof((indp)[0]),\	(indp),(alen),(arcode),(ub4)0,(ub4)*0,(text*)0,NA,NA))\
#define VER7 2	sizeof((alen)[0]),sizeof((arcode)[0])); */	{errrpt(lda,cursor,sqlvar);return(-1);}\
		else\
		DISCARD 0
#define NA -1 /* ANSI SQL NULL */	#define	
#define NLT 1 /* length for string null terminator */	OCIBNDR(stmp,bndp,errp,sqlvar,progv,progv1,ftype,indp,alen,arcode) \	#define
#define DEADLOCK 60 /* ORA-00060: deadlock */	ocierror(_FILE_,_LINE_,(errp), \	OBNDRAA(lda,cursor,sqlvar,progv,progv1,ftype,indp,alen,arcode,ms,cs)\
#define NO_DATA_FOUND 1403 /* ORA-01403: no data found */	OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_HTYPE_BIND,0,	if (obndraa((cursor),(text*)(sqlvar),NA,(ub1*)(progv),(progv1),(ftype),NA,\
#define NOT_SERIALIZABLE 8177 /* ORA-08177: transaction not	(dvoid**0)); \	(indp),(alen),(arcode),(ub4)(ms),(ub4*)(cs),(text*)0,NA,NA))\
serializable */	ocierror(_FILE_,_LINE_,(errp), \	{errrpt(lda,cursor,sqlvar);return(-1);}\
#define SNAPSHOT_TOO_OLD 1555	OCIBindByName((stmp),&(bndp),(errp),(text *)	else\
	(sqlvar),strlen((sqlvar)), \	DISCARD 0
	(progv),(progv1),(ftype),	
#ifndef NULLP	(indp),(alen),(arcode),0,0,OCI_DEFAULT));	
#define NULLP (void *)NULL		#define
#endif /* NULLP */		ODEFIN(lda,cursor,pos,buf,buf1,ftype,indp,fmt,fmtl,fmtt,rln,rcode)\
	#define	if (odefin((cursor),(pos),(ub1*)(buf),(buf1),(ftype),(scale),(indp),\
#define ADR(object) ((ub1 *)&(object))	OCIBNDRAA(stmp,bndp,errp,sqlvar,progv,progv1,ftype,indp,alen,arcode,ms,	(text*)(fmt),(fmtl),(fmtt),(rln),(rcode))\
#define SIZ(object) ((sword)sizeof(object))	cu) \	{errrpt(lda,cursor,(text *)ftype);return(-1);}\
	ocierror(_FILE_,_LINE_, (errp), \	else\
typedef char date[24+NLT];	OCIHandleAlloc((stmp),(dvoid*)&(bndp),OCI_HTYPE_BIND,0,	DISCARD 0
typedef char varchar2;	(dvoid**0)); \	
	ocierror(_FILE_,_LINE_,(errp),\	
#define OCIERROR(errp,function)\	OCIBindByName((stmp),&(bndp),(errp),(text *)	#define OEXFET(lda,cursor,nrows,rc,exact)\
ocierror(_FILE_,_LINE_,(errp),(function));	(sqlvar),strlen((sqlvar)), \	if (oexfet((cursor),(nrows),(cancel),(exact))\
	(progv),(progv1),(ftype),(indp),(alen),(arcode),(ms),	{if ((cursor)->rc == 1403) DISCARD 0; \
#define OCIBND(stmp, bndp, errp, sqlvar, progv, progv1, ftype)	(cu),OCI_DEFAULT));	else if (errrpt(lda,cursor,(text *)"OEXFET")=="RECOVER") \

```

{orol(lda);return(RECOVERR);} \
else {orol(lda);return(-1);} \
else\
DISCARD 0

#define OOPEN(lda,cursor)\
if (oopen((cursor),(lda),(text*)0,NA,NA,(text*)0,NA))\
{errrpt(lda,cursor,(text *)"OOPEN");return(-1);} \
else\
DISCARD 0

#define OPARSE(lda,cursor,sqlstm,sql,defflg,lngflg)\
if (oparse((cursor),(sqlstm),(sb4)(sql),(defflg),(ub4)(lngflg)))\
{errrpt(lda,cursor,sqlstm);return(-1);} \
else\
DISCARD 0

#define OFEN(lda,cursor,nrows)\
if (ofen((cursor),(nrows)))\
{if (errrpt(lda,cursor,(text *)"OFEN")==RECOVERR) \
{orol(lda);return(RECOVERR);} \
else {orol(lda);return(-1);} \
else\
DISCARD 0

#define OEXEC(lda,cursor)\
if (oexec((cursor)))\
{if (errrpt(lda,cursor,(text *)"OEXEC")==RECOVERR) \
{orol(lda);return(RECOVERR);} \
else {orol(lda);return(-1);} \
else\
DISCARD 0

#define OCOM(lda,cursor)\
if (ocom((lda)) \
{errrpt(lda,cursor,(text *)"OCOM");orol(lda);return(-1);} \
else\
DISCARD 0

#define OEXN(lda,cursor,itors,rowoff)\
if (oexn((cursor),(itors),(rowoff)) \
{if (errrpt(lda,cursor,(text *)"OEXN")==RECOVERR) \
{orol(lda);return(RECOVERR);} \
else {orol(lda);return(-1);} \
else\
DISCARD 0

#endif

/* additions done for 814 -shishir */
#define OCI_ATTR_SRVRCTXT OCI_ATTR_SERVER
#define OCI_ATTR_USERCTXT OCI_ATTR_SESSION
#define OCI_ATTR_ROW CNT OCI_ATTR_ROW_COUNT
#define OCI_HTYPE_ERR OCI_HTYPE_ERROR
#define OCI_HTYPE_STM OCI_HTYPE_STMT

tpccflags.h
#define PLSQLNO
#define DMLRETDEL

tpcc.h
/* * $Header: tpcc.h 7030100.1 95/07/19 15:10:55 plai Generic $ Copyr (c)
1993 Oracle */
/*
=====+ | Copyright (c) 1995 Oracle Corp, Redwood Shores, CA | |
OPEN SYSTEMS PERFORMANCE GROUP | | All Rights Reserved |
=====+ | FILENAME | tpcc.h | DESCRIPTION | Include file for
TPC-C benchmark programs.
=====+
=====*/ #ifndef TPCC_H #define TPCC_H #ifndef FALSE #
define FALSE 0 #endif #ifndef TRUE # define TRUE 1 #endif #include
#include #include #ifndef boolean #define boolean int #endif
#include #include #include /* #ifndef __STDC__ #include "ociapr.h" #else
#include "ocikpr.h" #endif */ typedef struct cda_def csrdef; typedef struct
cda_def ldadef; /* TPC-C transaction functions */ #ifndef TUXEDO extern
int TPCinit (); extern int TPCnew (); extern int TPCpay (); extern int TPCord
(); extern int TPCdel (); extern int TPCsto (); extern void TPCexit (); extern
int TPCdumpinit (); extern void TPCdumpnew (); extern void TPCdumppay
(); extern void TPCdumpord (); extern void TPCdumpdel (); extern void
TPCdumpsto (); extern void TPCdumpexit (); /* extern void userlog(char*
fmt, ...); */ #endif /* Error codes */ #define RECOVERR -10 #define
IRRECERR -20 #define NOERR 111 #define DEL_ERROR -666 #define
DEL_DATE_LEN 77 #define NDISTS 10 #define NITEMS 15 #define
SQL_BUF_SIZE 8192 #define FULLDATE "dd-mon-yy.hh24:mi:ss" #define
SHORTDATE "dd-mm-yyyy" #define DELRT 80.0 #ifndef TUX extern int
tkvcninit (); extern int tkvcpinit (); extern int tkvcoint (); extern int
tkvcdinit (); extern int tkvcsinit (); extern int tkvcn (); extern int tkvcv
(); extern int tkvcv (); extern int tkvcs (); extern void tkvcndone ();
extern void tkvcvdone (); extern void tkvcodone (); extern void tkvcddone ();
extern void tkvcsdone (); extern int tkvcss (); /* for alter session to get
memory size and trace */ extern boolean multitrans; extern int ord_init;
#endif extern void errrpt (); extern int ocierror(char *fname, int
lineno,OCIError *errhp, sword status); extern int sqlfile(char *fname, text
*linebuf); extern FILE *lfp; extern FILE *fopen (); extern int proc_no; extern
int doid[]; extern int execstatus; extern int errcode; #ifndef TUX extern
OCIEnv *tpcenv; extern OCIServer *tpcsrv; extern OCIError *errhp; extern
OCISvcCtx *tpcsvc; extern OCISession *tpcsur; extern OCISmt *curntest;
#endif /* The bind and define handles for each transaction are included in
their respective header files. */ extern OCIEnv *tpcenv; extern OCIServer
*tpcsrv; extern OCIError *errhp; extern OCISvcCtx *tpcsvc; extern
OCISession *tpcsur; extern OCISmt *curi; extern char *uid; extern char
*pwd; /* for stock-level transaction */ extern int w_id; extern int d_id; extern
int c_id; extern int threshold; extern int low_stock; /* for delivery transaction
*/ extern int del_o_id[10]; extern int carrier_id; extern double qtime; extern
int retries; extern int retries_serial; extern int retries_recover; extern int
retries_snapshot; /* for order-status transaction */ extern int bylastname;
extern char c_last[17]; extern char c_first[17]; extern char c_middle[3];
extern double c_balance; extern int o_id; extern text o_entry_d[20]; extern int
o_carrier_id; extern int o_ol_cnt; extern int ol_supply_w_id[15]; extern int
ol_i_id[15]; extern int ol_quantity[15]; extern int ol_amount[15]; ub4
ol_del_len[15]; extern text ol_delivery_d[15][11]; /* xnie - begin */ extern
OCIRowid *o_rowid; /* xnie - end */ /* for payment transaction */ extern int
c_w_id; extern int c_d_id; extern int h_amount; extern char w_street_1[21];
extern char w_street_2[21]; extern char w_city[21]; extern char w_state[3];
extern char w_zip[10]; extern char d_street_1[21]; extern char d_street_2[21];
extern char d_city[21]; extern char d_state[3]; extern char d_zip[10]; extern
char c_street_1[21]; extern char c_street_2[21]; extern char c_city[21]; extern
char c_state[3]; extern char c_zip[10]; extern char c_phone[17]; extern text
c_since_d[11]; extern char c_credit[3]; extern int c_credit_lim; extern float
c_discount; extern char c_data[201]; extern text h_data[20]; /* for new order
transaction */ #ifndef TUX extern int nol_i_id[15]; extern int
nol_supply_w_id[15]; extern int nol_quantity[15]; extern int
nol_quantit0[15]; extern int nol_quantit9[15]; extern int nol_ytdqty[15];
extern int nol_amount[15]; extern int o_all_local; extern float w_tax; extern
float d_tax; extern float total_amount; extern char i_name[15][25]; extern int
i_name_strlen[15]; extern ub2 i_name_strlen_len[15]; extern ub2
i_name_strlen_rcode[15]; extern ub4 i_name_strlen_csize; extern int
s_quantity[15]; #if 0 extern char brand_gen[15]; #endif extern ub2
brand_gen_len[15]; extern ub2 brand_gen_rcode[15]; extern ub4
brand_gen_csize; extern int i_price[15]; extern char brand_generic[15][1];
extern int status; extern int tracelevel; /* Miscellaneous */ extern OCIDate
cr_date; extern OCIDate c_since; extern OCIDate o_entry_d_base; extern
OCIDate ol_d_base[15]; #endif #ifndef DISCARD # define DISCARD (void)
#endif #ifndef sword # define sword int #endif #define VER7 2 #define NA
-1 /* ANSI SQL NULL */ #define NLT 1 /* length for string null terminator
*/ #define DEADLOCK 60 /* ORA-00060: deadlock */ #define
NO_DATA_FOUND 1403 /* ORA-01403: no data found */ #define
NOT_SERIALIZABLE 8177 /* ORA-08177: transaction not serializable */
#define SNAPSHOT_TOO_OLD 1555 /* ORA-01555: snapshot too old */
#define NULLP # define NULLP(x) (x *)NULL #endif /* NULLP */ #define
ADR(object) ((ub1 *)&(object)) #define SIZ(object) ((sword)sizeof(object))
typedef char date[24+NLT]; typedef char varchar2; #define min(x,y) (((x) <
(y)) ? (x) : (y)) #define OCIERROR(errp,function)\
ocierror(__FILE__, __LINE__,(errp),(function)); #define OCIBND(stmp,
bndp, errp, sqlvar, progvl, ftype)\
ocierror(__FILE__, __LINE__,(errp), \
OCIHandleAlloc((stmp),(dvoid*)&(bndp)),OCI_HTYPE_BIND,0,
(dvoid**0)); \
ocierror(__FILE__, __LINE__,(errp), \
OCIBindByName((stmp), &(bndp), (errp), \
(text *)sqlvar), strlen((sqlvar)),
(progvl), (progvl), (ftype),0,0,0,0,OCI_DEFAULT)); /* bind arrays for sql */
#define
OCIBNDRA(stmp,bndp,errp,sqlvar,progvl,ftype,indp,alen,arcode) \
DISCARD ocierror(__FILE__, __LINE__,(errp), \
OCIHandleAlloc((stmp),
(dvoid*)&(bndp)),OCI_HTYPE_BIND,0,(dvoid**0)); \
DISCARD
ocierror(__FILE__, __LINE__,(errp), \
OCIBindByName((stmp),&(bndp),
(errp),(text *)sqlvar),strlen((sqlvar)),
(progvl),(progvl),(ftype),(indp),(alen),
(arcode),0,0,OCI_DEFAULT)); /* use with callback data */ #define
OCIBNDRAD(stmp,bndp,errp,sqlvar,progvl,ftype,indp,ctxp,\
cbf_nodata,cbf_data) \
DISCARD ocierror(__FILE__, __LINE__,(errp), \
OCIHandleAlloc((stmp),(dvoid*)&(bndp)),OCI_HTYPE_BIND,0,
(dvoid**0)); \
DISCARD
ocierror(__FILE__, __LINE__,(errp), \
OCIBindByName((stmp),&(bndp),(errp),(text *)sqlvar), \
strlen((sqlvar)),0,
(progvl),(ftype), \
indp,0,0,0,0,OCI_DATA_AT_EXEC)); \
DISCARD
ocierror(__FILE__, __LINE__,(errp), \
OCIBindDynamic((stmp),(errp),
(ctxp),(cbf_nodata),(ctxp),(cbf_data)); /* bind in/out for psql without
indicator and rcode */ #define
OCIBNDPL(stmp,bndp,errp,sqlvar,progvl,ftype,alen) \
DISCARD
ocierror(__FILE__, __LINE__,(errp), \
OCIHandleAlloc((stmp),
(dvoid*)&(bndp)),OCI_HTYPE_BIND,0,(dvoid**0)); \
DISCARD
ocierror(__FILE__, __LINE__,(errp), \
OCIBindByName((stmp),&(bndp),
(errp),(CONST text *)sqlvar), \
(sb4)strlen((CONST char *)sqlvar),
(dvoid*)(progvl),(progvl),(ftype), \
NULLP(dvoid),(alen), \
NULLP(ub2),

```

```
0,NULLP(ub4),OCI_DEFAULT)); /* bind in values for plsql with indicator
and rcode */ #define
OCIBNDR(stmp,bndp,errp,sqlvar,progvl,ftype,indp,alen,arcodes) \
DISCARD ocierror( FILE __LINE__,(errp), \ OCCHandleAlloc(stmp),
(dvoid**)&(bndp),OCI_HTYPE_BIND,0,(dvoid**0)); \ DISCARD
ocierror( FILE __LINE__,(errp), \ OCIBindByName(stmp,&(bndp),
(errp),(text*)(sqlvar),strlen((sqlvar)), \ (progvl),(progvl),(ftype),(indp),(alen),
(arcodes),0,0, \ OCI_DEFAULT)); /* bind in/out for plsql arrays without
indicator and rcode */ #define
OCIBNDPLA(stmp,bndp,errp,sqlvar,progvl,ftype,alen,ms,cu) \
DISCARD ocierror( FILE __LINE__,(errp), \ OCCHandleAlloc(stmp),
(dvoid**)&(bndp),OCI_HTYPE_BIND,0,(dvoid**0)); \ DISCARD
ocierror( FILE __LINE__,(errp), \ OCIBindByName(stmp,&(bndp),
(errp),(CONST text*)(sqlvar), \ (sb4)strlen((CONST char*)(sqlvar)),(void
*)(progvl), \ (progvl),(ftype),NULL,(alen),NULL,(ms),(cu),OCI_DEFAULT));
/* bind in/out values for plsql with indicator and rcode */ #define
OCIBNDRAA(stmp,bndp,errp,sqlvar,progvl,ftype,indp,alen,arcodes, \
ms,cu) \ ocierror( FILE __LINE__,(errp), \ OCCHandleAlloc(stmp),
(dvoid**)&(bndp),OCI_HTYPE_BIND,0,(dvoid**0)); \
ocierror( FILE __LINE__,(errp), \ OCIBindByName(stmp,&(bndp),
(errp),(text*)(sqlvar),strlen((sqlvar)), \ (progvl),(progvl),(ftype),(indp),(alen),
(arcodes),(ms),(cu),OCI_DEFAULT)); #define
OCIDEFINE(stmp,dfnp,errp,pos,progvl,ftype)
OCIDefineByPos(stmp,&(dfnp),(errp),(pos),(progvl),(ftype), \
0,0,0,OCI_DEFAULT); #define
OCIDEF(stmp,dfnp,errp,pos,progvl,ftype) \ OCCHandleAlloc(stmp),
(dvoid**)&(dfnp),OCI_HTYPE_DEFINE,0, \ (dvoid**0));
OCIDefineByPos(stmp,&(dfnp),(errp),(pos),(progvl),(progvl), \
(ftype),NULL,NULL,NULL,OCI_DEFAULT); \ #define
OCIDFNRA(stmp,dfnp,errp,pos,progvl,ftype,indp,alen,arcodes) \
OCCHandleAlloc(stmp),(dvoid**)&(dfnp),OCI_HTYPE_DEFINE,0, \
(dvoid**0); \ OCIDefineByPos(stmp,&(dfnp),(errp),(pos),(progvl), \
(progvl),(ftype),(indp),(alen), \ (arcodes),OCI_DEFAULT); #define
OCIDFNNDYN(stmp,dfnp,errp,pos,progvl,ftype,indp,ctxp,cbf_data) \
ocierror( FILE __LINE__,(errp), \ OCCHandleAlloc(stmp),
(dvoid**)&(dfnp),OCI_HTYPE_DEFINE,0, \ (dvoid**0)); \
ocierror( FILE __LINE__,(errp), \ OCIDefineByPos(stmp,&(dfnp),
(errp),(pos),(progvl), \ (progvl),(ftype), \ (indp),NULL,NULL,
OCI_DYNAMIC_FETCH)); \ ocierror( FILE __LINE__,(errp), \
OCIDefineDynamic((dfnp),(errp),(ctxp),(cbf_data)); /* New order */ struct
newinstruct { int w_id; int d_id; int c_id; int ol_i_id[15]; int
ol_supply_w_id[15]; int ol_quantity[15]; }; struct newoutstruct { int terror;
int o_id; int o_ol_cnt; char c_last[17]; char c_credit[3]; float c_discount; float
w_tax; float d_tax; char o_entry_d[20]; float total_amount; char i_name[15]
[25]; int s_quantity[15]; char brand_generic[15]; float i_price[15]; float
ol_amount[15]; char status[26]; int retry; }; struct newstruct { struct
newinstruct newin; struct newoutstruct newout; }; /* Payment */ struct
payinstruct { int w_id; int d_id; int c_w_id; int c_d_id; int c_id; int
bylastame; int h_amount; char c_last[17]; }; struct payoutstruct { int terror;
char w_street_1[21]; char w_street_2[21]; char w_city[21]; char w_state[3];
char w_zip[10]; char d_street_1[21]; char d_street_2[21]; char d_city[21];
char d_state[3]; char d_zip[10]; int c_id; char c_first[17]; char c_middle[3];
char c_last[17]; char c_street_1[21]; char c_street_2[21]; char c_city[21];
char c_state[3]; char c_zip[10]; char c_phone[17]; char c_since[11]; char
c_credit[3]; double c_credit_lim; float c_discount; double c_balance; char
c_data[201]; char h_date[20]; int retry; }; struct paystruct { struct payinstruct
payin; struct payoutstruct payout; }; /* Order status */ struct ordinstruct { int
w_id; int d_id; int c_id; int bylastname; char c_last[17]; }; struct ordoutstruct
```

```
{ int terror; int c_id; char c_last[17]; char c_first[17]; char c_middle[3];
double c_balance; int o_id; char o_entry_d[20]; int o_carrier_id; int o_ol_cnt;
int ol_supply_w_id[15]; int ol_i_id[15]; int ol_quantity[15]; float
ol_amount[15]; char ol_delivery_d[15][11]; int retry; }; struct ordstruct
{ struct ordinstruct ordin; struct ordoutstruct ordout; }; /* Delivery */ struct
delinstruct { int w_id; int o_carrier_id; double qtime; int in_timing_int; int
plsqliflag; }; struct deloutstruct { int terror; int retry; }; struct delstruct { struct
delinstruct delin; struct deloutstruct delout; }; /* Stock level */ struct
/* vmm313 void ocierror(fname, lineno, errhp, status) */
return(RECOVER);
else
return(IRRECERR);
} /*
}
struct stoostruct { int terror; int ocierror(fname, lineno, errhp, status)
char *fname;
int lineno;
OCIError *errhp;
sword status;
text errbuf[512];
ub4 buflen;
sb4 errcode;
sb4 lstat;
ub4 recno=2;

switch (status) {
case OCI_SUCCESS:
break;
case OCI_SUCCESS_WITH_INFO:
(void) userlog("Module %s Line %d\n", fname, lineno);
(void) userlog("Error - OCI_SUCCESS_WITH_INFO\n");
break;
case OCI_NEED_DATA:
(void) userlog("Module %s Line %d\n", fname, lineno);
(void) userlog("Error - OCI_NEED_DATA\n");
return (IRRECERR);
case OCI_NO_DATA:
(void) userlog("Module %s Line %d\n", fname, lineno);
(void) userlog("Error - OCI_NO_DATA\n");
return IRRECERR; /* for 8.1.4 */
break;
case OCI_ERROR:
lstat = OCIErrorGet (errhp, (ub4) 1,
(text *) NULL, &errcode, errbuf,
(ub4) sizeof(errbuf), OCI_HTYPE_ERR);
if (errcode == NOT_SERIALIZABLE) return (errcode);
if (errcode == SNAPSHOT_TOO_OLD) return (errcode);
while (lstat != OCI_NO_DATA)
{
(void) userlog("Module %s Line %d\n", fname, lineno);
(void) userlog("Error - %s\n", errbuf);
lstat = OCIErrorGet (errhp, recno++, (text *) NULL, &errcode,
errbuf,
(ub4) sizeof(errbuf), OCI_HTYPE_ERROR);
}
return (errcode);
break;
case OCI_INVALID_HANDLE:
(void) userlog("Module %s Line %d\n", fname, lineno);
(void) userlog("Error - OCI_INVALID_HANDLE\n");
break;
```

## Uni\_svr

### ora\_errrpt.c

```
/*
* Copyright (c) 1995 by Sun Microsystems, Inc.
*/
#pragma ident "@(#)ora_errrpt.c 1.1 95/09/14 SMI"

/*
* these functions actually belong in ~dbbench/generic/c/msgsh_log.c. We put
them
* here because they have database specific statements.
*/
#include "ora_err.h"
#include "ora_oci.h"

errrpt(lda, cur, sqlvar)
ldafdef *lda;
csrdef *cur;
text *sqlvar;
{
text msg[2048];
/* if (cur->rc) { */
oerhms(lda, (sb2) cur->rc, msg, 2048);
userlog("%s sql_variable %s\n", msg, sqlvar);

if (cur->rc == DEADLOCK || (cur->rc ==
SNAPSHOT_TOO_OLD))
```

```

case OCI_STILL_EXECUTING:
(void) userlog("Module %s Line %d\n", fname, lineno);
(void) userlog("Error - OCI_STILL_EXECUTE\n");
return (IRRECERR);
case OCI_CONTINUE:
(void) userlog("Module %s Line %d\n", fname, lineno);
(void) userlog("Error - OCI_CONTINUE\n");
return (IRRECERR);
default:
(void) userlog("Module %s Line %d\n", fname, lineno);
(void) userlog("Error - \n");
return (IRRECERR);
}
return RECOVERR;
}

tpcc_srv_init.c
/*
 * Copyright (c) 1994 by Sun Microsystems, Inc.
 */

#pragma ident "@(#)tpcso_srv_stock.c 1.6 95/04/12 SMI"

/
=====+
| Copyright (c) 1994 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
+=====+
| FILENAME |
| renamed to tpcc_srv_init.c |
| DESCRIPTION |
| OCI for opening connect/session in TPC-C benchmark. |
+=====+
/* #include "ora_oci.h" */
#ifndef ORA_TPCC
#define ORA_TPCC
#include "tpcc.h"
#endif

#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>
/* #include "./ora_err.h" */
/* Tuxedo */
#include "atmi.h"
#include "userlog.h"

OCIEnv *tpcenv;
OCIServer *tpcsrv;

OCIErr *errhp;
OCISvcCtx *tpscvc;
OCISession *tpcusr;
OCISmt *curi;

char *uid = "tpcc";
char *pwd = "tpcc";

#define SQLTXT "alter session set isolation_level = serializable"

/*
 * Initialize the environment, err-handle, attach, open session,
 * alter session to serializable. Common for all 5 TX.
 */

int status, execstatus, errcode;

/* external routines */
int init_stock_tx();
int init_ords_tx();
int init_paym_tx();
int init_newo_tx();

int stocklevel_tx( TPSVCINFO *rqst );
int ordstat_tx(TPSVCINFO *rqst);
int payment_tx(TPSVCINFO *rqst);
int neworder_tx(TPSVCINFO *rqst);

int TPCinit()
{
int i;
text stmbuf[16*1024];

/* common to all 5 */
OCIInitialize(OCI_DEFAULT|OCI_OBJECT,(dvoid *)0,0,0,0);
OCIEnvInit(&tpcenv, OCI_DEFAULT, 0, (dvoid **)0);
OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcsrv,
OCI_HTYPE_SERVER, 0, (dvoid **)0);
OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&errhp,
OCI_HTYPE_ERROR, 0, (dvoid **)0);
OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpscvc,
OCI_HTYPE_SVCCTX, 0, (dvoid **)0);
OCIServerAttach(tpcsrv, errhp, (text *)0,0,OCI_DEFAULT);
OCIAttrSet((dvoid *)tpscvc, OCI_HTYPE_SVCCTX, (dvoid *)tpcsrv,
(ub4)0,OCI_ATTR_SERVER, errhp);
OCIHandleAlloc((dvoid *)tpcenv, (dvoid **)&tpcusr,
OCI_HTYPE_SESSION, 0, (dvoid **)0);
OCIAttrSet((dvoid *)tpcusr, OCI_HTYPE_SESSION, (dvoid *)uid,
(ub4)strlen(uid),OCI_ATTR_USERNAME, errhp);
OCIAttrSet((dvoid *)tpcusr, OCI_HTYPE_SESSION, (dvoid *)pwd,
(ub4)strlen(pwd),
OCI_ATTR_PASSWORD, errhp);
fprintf(stderr, "%d] Now call SessionBegin UID=%s PWD=%s\n", getpid(),
uid, pwd);

OCIERROR(errhp, OCISessionBegin(tpscvc, errhp, tpcusr,
OCI_CRED_RDBMS, OCI_DEFAULT));

OCIAttrSet(tpscvc, OCI_HTYPE_SVCCTX, tpcusr, 0,
OCI_ATTR_SESSION, errhp);

/* run all transaction in serializable mode */

OCIHandleAlloc(tpcenv, (dvoid **)&curi, OCI_HTYPE_STMT, 0,
(dvoid **)0);
sprintf((char *) stmbuf, SQLTXT);
OCIStmtPrepare(curi, errhp, stmbuf, strlen((char *)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT);
OCIERROR(errhp,OCIStmtExecute(tpscvc, curi,
errhp,1,0,0,0,OCI_DEFAULT));
OCIHandleFree(curi, OCI_HTYPE_STMT);
/* end common ---- */

return(0);
}

/* Start of Tuxedo code */
int tpsvrit(int argc, char **argv)
{
int retcode;

char *two_task = getenv("TWO_TASK");
if (two_task == NULL) {
two_task = "(null)";
}
userlog("TWO_TASK=%s\n", two_task);

/* For all servers - common routine to open/init session etc. */
TPCinit();
/* Successful return is 0 */
if (retcode==init_newo_tx())
return (retcode);
if (retcode==init_paym_tx())
return (retcode);
if (retcode==init_ords_tx())
return (retcode);
if (retcode==init_stock_tx())
return (retcode);

return(0);
}

tpcc_srv_newo.c
/*
 * Copyright (c) 1995 by Sun Microsystems, Inc.
 */

#pragma ident "@(#)tpcso_srv_newo.c 1.14 97/01/02 SMI"

/
=====+
| Copyright (c) 1996 Oracle Corp, Redwood Shores, CA |

```

```

| OPEN SYSTEMS PERFORMANCE GROUP | int ol_quantity;
| All Rights Reserved | int s_quantity;
+=====+ char brand[2];
| FILENAME | double i_price;
| plnew.c | double ol_amount;
| DESCRIPTION | };
| OCI version (using PL/SQL stored procedure) of /* List of fields in neworder */
| NEW ORDER transaction in TPC-C benchmark.
| *** As perf Mar 4, 2009 pre-audit: changes for interactive compliance
+=====+
/*
#ifndef ORA_TPCC
# define ORA_TPCC
# include "tpcc.h"
#endif

#include <signal.h>
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>

/* TAKE THESE OUT IF NOT DEBUGGING */
#include "ora_oci.h"
#include "./ora_err.h"

/* Tuxedo includes */
#include "atmi.h"
#include "userlog.h"

static int tx_count = 0;

void vgetdate (unsigned char *oradt);
int sqlfile (char *fnam, text *linebuf);
void cvtdmyhms (unsigned char *oradt, char *outdate);

#define MOVETO(element, struct_name) element = struct_name->element
#define MOVEBACK(element, struct_name) struct_name->element =
element
#define MOVEBACK(element, cnt, struct_name) strncpy(struct_name-
>element, element, cnt)

/* Lists of items on an order */
/* These structures should match the struct definitions for no_struct
* defined in tpcc_client.h exactly.
* Any change to those, should be reflected here
*/

struct items_inf {
int ol_supply_w_id;
int ol_i_id;
char i_name[25];

int ol_quantity;
int s_quantity;
char brand[2];
double i_price;
double ol_amount;
};

struct newo_inf {
int w_id;
int d_id;
int c_id;
int o_id;
int o_ol_cnt;
double c_discount;
double w_tax;
double d_tax;
char o_entry_d[20];
char c_credit[3];
char c_last[17];
struct items_inf n_items[15];
char status_msg[25];
double total;
};

int indx[15];
void swap(struct newo_inf *str, int i, int j);
void q_sort(int *arr, struct newo_inf *str, int left, int right);

/*struct msg_h_req message; */
char blank_msg[25] = " ";

int my_qid, my_id;
char my_name[] = "Newo";

/******
* BEGIN BLOCK OF COMMON CODE
*****
/* struct newo_inf */
/* for new order transaction */
static int w_id;
static int d_id;
static int c_id;
static int o_id;
static int o_ol_cnt;
static float c_discount;
static text o_entry_d[20];
/* char o_entry_d[20]; */
static char c_credit[3];
static char c_last[17];
static char status_msg[25];

static double total;

static int nol_i_id[15];

static int nol_supply_w_id[15];
static int nol_quantity[15];
static int nol_quant[10][15];
static int nol_quant[9][15];
static int nol_ytdqty[15];
static int nol_amount[15];
static int o_all_local;
static float w_tax;
static float d_tax;
static float total_amount;
static char i_name[15][25];
static int s_quantity[15];
/* char brand_gen[15]; */
static int i_price[15];
static char brand_gen[15];
static char brand_generic[15][1];
static int tracelevel = 0;

static OCIDate cr_date;
static OCIDate c_since;
static OCIDate o_entry_d_base;
static OCIDate ol_d_base[15];
static dvoid *xmem;
static int retries;

#define SQLTXT2 "BEGIN inittpc.init_no(:idx larr); END;";

#define NITEMS 15
#define ROWIDLEN 20
#define OCIROWLEN 20

struct newctx {

ub2 nol_i_id_len[NITEMS];
ub2 nol_supply_w_id_len[NITEMS];
ub2 nol_quantity_len[NITEMS];
ub2 nol_amount_len[NITEMS];
ub2 s_quantity_len[NITEMS];
ub2 i_name_len[NITEMS];
ub2 i_price_len[NITEMS];
ub2 s_dist_info_len[NITEMS];
ub2 ol_o_id_len[NITEMS];
ub2 ol_number_len[NITEMS];
ub2 s_remote_len[NITEMS];
ub2 s_quant_len[NITEMS];
ub2 ol_dist_info_len[NITEMS];
ub2 s_bg_len[NITEMS];

int ol_o_id[NITEMS];
int ol_number[NITEMS];

int s_remote[NITEMS];
char s_dist_info[NITEMS][25];
OCISmt *curn1;
OCIBind *ol_i_id_bp;
OCIBind *ol_supply_w_id_bp;
OCIBind *i_price_bp;

```

```

OCIBind *i_name_bp;
OCIBind *s_bg_bp;
ub4 nol_i_count;
ub4 nol_s_count;
ub4 nol_q_count;
ub4 nol_item_count;
ub4 nol_name_count;
ub4 nol_qty_count;
ub4 nol_bg_count;
ub4 nol_am_count;
ub4 s_remote_count;
OCISmt *curn2;
OCIBind *ol_quantity_bp;
OCIBind *s_remote_bp;
OCIBind *s_quantity_bp;
OCIBind *w_id_bp;
OCIBind *d_id_bp;
OCIBind *c_id_bp;
OCIBind *o_all_local_bp;
OCIBind *o_all_cnt_bp;
OCIBind *w_tax_bp;
OCIBind *d_tax_bp;
OCIBind *o_id_bp;
OCIBind *c_discount_bp;
OCIBind *c_credit_bp;
OCIBind *c_last_bp;
OCIBind *retries_bp;
OCIBind *cr_date_bp;
OCIBind *ol_o_id_bp;
OCIBind *ol_amount_bp;

sb2 w_id_len;
ub2 d_id_len;
ub2 c_id_len;
ub2 o_all_local_len;
ub2 o_ol_cnt_len;
ub2 w_tax_len;
ub2 d_tax_len;
ub2 o_id_len;
ub2 c_discount_len;
ub2 c_credit_len;
ub2 c_last_len;
ub2 retries_len;
ub2 cr_date_len;
};

typedef struct newctx newctx;

static newctx *nctx;

/*
 * Initialize the neworder transaction
 */

int
init_newo_tx()
{
    int execstatus, errcode;
    char filename[200];
    int proc_no;
    int i;
    text stmbuf[16*1024];
    nctx = (newctx *) malloc (sizeof(newctx));
    DISCARD memset(nctx,(char)0,sizeof(newctx));
    nctx->w_id_len = sizeof(w_id);
    nctx->d_id_len = sizeof(d_id);
    nctx->c_id_len = sizeof(c_id);
    nctx->o_all_local_len = sizeof(o_all_local);
    nctx->o_ol_cnt_len = sizeof(o_ol_cnt);
    nctx->w_tax_len = 0;
    nctx->d_tax_len = 0;
    nctx->o_id_len = sizeof(o_id);
    nctx->c_discount_len = 0;
    nctx->c_credit_len = 0;
    nctx->c_last_len = 0;
    nctx->retries_len = sizeof(retries);
    nctx->cr_date_len = sizeof(cr_date);

    /* open first cursor */
    DISCARD OCIERROR(errhp,OCIHandleAlloc(tpcenv,(dvoid **)&(&nctx-
    >curn1),
        OCI_HTYPE_STMT, 0, (dvoid**)0));
    sqlfile("tkvcnew.sql",stmbuf);
    DISCARD OCIERROR(errhp,OCISmtPrepare(nctx->curn1, errhp, stmbuf,
        strlen((char *)stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT));

    /* bind variables */

    OCIBNDPL(nctx->curn1, nctx->w_id_bp, errhp,
        ":w_id",ADR(w_id),SIZ(w_id),
        SQLT_INT, &nctx->w_id_len);
    OCIBNDPL(nctx->curn1, nctx->d_id_bp, errhp,
        ":d_id",ADR(d_id),SIZ(d_id),
        SQLT_INT, &nctx->d_id_len);
    OCIBNDPL(nctx->curn1, nctx->c_id_bp, errhp,
        ":c_id",ADR(c_id),SIZ(c_id),
        SQLT_INT, &nctx->c_id_len);
    OCIBNDPL(nctx->curn1, nctx->o_all_local_bp, errhp, ":o_all_local",
        ADR(o_all_local), SIZ(o_all_local),SQLT_INT, &nctx-
    >o_all_local_len);
    OCIBNDPL(nctx->curn1, nctx->o_all_cnt_bp, errhp,
        ":o_ol_cnt",ADR(o_ol_cnt),
        SIZ(o_ol_cnt),SQLT_INT, &nctx->o_ol_cnt_len);
    OCIBNDPL(nctx->curn1, nctx->w_tax_bp, errhp,
        ":w_tax",ADR(w_tax),SIZ(w_tax),
        SQLT_FLT, &nctx->w_tax_len);
    OCIBNDPL(nctx->curn1, nctx->d_tax_bp, errhp,
        ":d_tax",ADR(d_tax),SIZ(d_tax),
        SQLT_FLT, &nctx->d_tax_len);
    OCIBNDPL(nctx->curn1, nctx->o_id_bp, errhp,
        ":o_id",ADR(o_id),SIZ(o_id),
        SQLT_INT, &nctx->o_id_len);
    OCIBNDPL(nctx->curn1, nctx->c_discount_bp, errhp, ":c_discount",
        ADR(c_discount), SIZ(c_discount),SQLT_FLT, &nctx-
    >c_discount_len);

    OCIBNDPL(nctx->curn1, nctx->c_credit_bp, errhp, ":c_credit",c_credit,
        SIZ(c_credit),SQLT_CHR, &nctx->c_credit_len);
    OCIBNDPL(nctx->curn1, nctx->c_last_bp, errhp,
        ":c_last",c_last,SIZ(c_last),
        SQLT_STR, &nctx->c_last_len);
    OCIBNDPL(nctx->curn1, nctx->retries_bp, errhp, ":retry",ADR(retries),
        SIZ(retries),SQLT_INT, &nctx->retries_len);
    OCIBNDPL(nctx->curn1, nctx->cr_date_bp, errhp, ":cr_date",&cr_date,
        SIZ(OCIDate), SQLT_ODT, &nctx->cr_date_len);

    OCIBNDPLA(nctx->curn1, nctx->ol_i_id_bp,errhp,":ol_i_id",nol_i_id,
        SIZ(int), SQLT_INT, nctx->nol_i_id_len,NITEMS,&nctx-
    >nol_i_count);
    OCIBNDPLA(nctx->curn1, nctx->ol_supply_w_id_bp, errhp,
        ":ol_supply_w_id",
        nol_supply_w_id,SIZ(int),SQLT_INT, nctx->nol_supply_w_id_len,
        NITEMS, &nctx->nol_s_count);
    OCIBNDPLA(nctx->curn1, nctx->ol_quantity_bp,errhp,":ol_quantity",
        nol_quantity, SIZ(int),SQLT_INT,nctx->nol_quantity_len,
        NITEMS,&nctx->nol_q_count);
    OCIBNDPLA(nctx->curn1, nctx-
    >i_price_bp,errhp,":i_price",i_price,SIZ(int),
        SQLT_INT, nctx->i_price_len, NITEMS, &nctx->nol_item_count);
    OCIBNDPLA(nctx->curn1, nctx->i_name_bp,errhp,":i_name",i_name,
        SIZ(i_name[0]),SQLT_STR, nctx->i_name_len,NITEMS,
        &nctx->nol_name_count);
    OCIBNDPLA(nctx->curn1, nctx-
    >s_quantity_bp,errhp,":s_quantity",s_quantity,
        SIZ(int), SQLT_INT,nctx->s_quant_len,NITEMS,&nctx-
    >nol_qty_count);
    OCIBNDPLA(nctx->curn1, nctx-
    >s_bg_bp,errhp,":brand_generic",brand_generic,
        SIZ(char), SQLT_CHR,nctx->s_bg_len,NITEMS,&nctx-
    >nol_bg_count);
    OCIBNDPLA(nctx->curn1, nctx-
    >ol_amount_bp,errhp,":ol_amount",nol_amount,
        SIZ(int),SQLT_INT, nctx->nol_amount_len,NITEMS,&nctx-
    >nol_am_count);
    OCIBNDPLA(nctx->curn1, nctx->s_remote_bp,errhp,":s_remote",nctx-
    >s_remote,
        SIZ(int),SQLT_INT, nctx->s_remote_len,NITEMS,&nctx-
    >s_remote_count);

    /* open second cursor */
    DISCARD OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&(&nctx-
    >curn2),
        OCI_HTYPE_STMT, 0, (dvoid**)0);
    DISCARD sprintf((char *) stmbuf, SQLTXT2);
    DISCARD OCIERROR(errhp,OCISmtPrepare(nctx->curn2, errhp, stmbuf,
        strlen((char *)stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT));

    /* execute second cursor to init newinit package */
    {
        int idx1arr[NITEMS];
        OCIBind *idx1arr_bp;
        ub2 idx1arr_len[NITEMS];
        ub2 idx1arr_rcode[NITEMS];
    }
}

```

```

sb2 idx1arr_ind[NITEMS];
ub4 idx1arr_count;
ub2 idx;

for (idx = 0; idx < NITEMS; idx++) {
    idx1arr[idx] = idx + 1;
    idx1arr_ind[idx] = TRUE;
    idx1arr_len[idx] = sizeof(int);
}
idx1arr_count = NITEMS;
o_ol_cnt = NITEMS;

/* Bind array */
OCIBNDPLA(nctx->curn2, idx1arr_bp, errhp, "idx1arr", idx1arr,
    SIZ(int), SOLT_INT, idx1arr_len, NITEMS, &idx1arr_count);

execstatus = OCISmtExecute(tpscvc, nctx->curn2, errhp, 1, 0,
    NULLP(CONST
OCISnapshot), NULLP(OCISnapshot), OCI_DEFAULT);
if (execstatus != OCI_SUCCESS) {
    OCITransRollback(tpscvc, errhp, OCI_DEFAULT);
    errcode = OCIERROR(errhp, execstatus);
    userlog("%d] newo_tx ERROR %s:%d - execstatus=%d, errcode=
%d\n", getpid(), __FILE__, __LINE__, execstatus, errcode);
    return -1;
}
return(0);
}

/*****
* END BLOCK OF COMMON CODE
*****/

int get_newo_tx_cnt()
{
    return tx_count;
}

/*
* This function executes the neworder transaction
*/

#if ACID
#include <sys/types.h>
#include <time.h>
time_t curtime, *timep = &curtime;
#endif

int neworder_tx(TPSCVCINFO *rqst)
{
    int i;

    int status, execstatus, errcode;
    int rcount;
    ub4 datelen;
    struct newo_inf *neworder_p;

#if ACID
    int reread;
    char sdate[30];
    time(timep);
    userlog("ACID NEWORDER started at %s\n", ctime(timep));
#endif

    neworder_p = (struct newo_inf *) (rqst->data);

    /* initialize the bind variables */
    MOVETO(w_id, neworder_p);
    MOVETO(d_id, neworder_p);
    MOVETO(c_id, neworder_p);
    MOVETO(o_ol_cnt, neworder_p);

    for (i = 0; i < 15; i++) {
        nol_i_id[i] = neworder_p->n_items[i].ol_i_id;
        nol_supply_w_id[i] = neworder_p->n_items[i].ol_supply_w_id;
        nol_quantity[i] = neworder_p->n_items[i].ol_quantity;
    }

    retries = 0;

    for (i = 0; i < NITEMS; i++) indx[i] = i;
    q_sort(nol_i_id, neworder_p, 0, o_ol_cnt-1);

    tx_count++;

    strcpy(neworder_p->status_mesg, blank_mesg);
    /* vgetdate((unsigned char *)cr_date); */
    OCIERROR(errhp, OCIDateSysDate(errhp, &cr_date));
    datelen = sizeof(o_entry_d);
    OCIERROR(errhp,
        OCIDateToText(errhp, &cr_date, (text*)FULLDATE, SIZ(FULLDATE),
        (text*)0, 0,
        &datelen, o_entry_d));

    retry:

    status = 0; /* number of invalid items */

    /* get number of order lines, and check if all are local */

    o_ol_cnt = NITEMS;
    o_all_local = 1;
    for (i = 0; i < NITEMS; i++) {
        if (nol_i_id[i] == 0) {
            o_ol_cnt = i;
            break;
        }
        if (nol_supply_w_id[i] != w_id) {
            nctx->s_remote[i] = 1;
        }
    }

    o_all_local = 0;
}
else
    nctx->s_remote[i] = 0;

nctx->w_id_len = sizeof(w_id);
nctx->d_id_len = sizeof(d_id);
nctx->c_id_len = sizeof(c_id);
nctx->o_all_local_len = sizeof(o_all_local);
nctx->o_ol_cnt_len = sizeof(o_ol_cnt);
nctx->w_tax_len = 0;
nctx->d_tax_len = 0;
nctx->o_id_len = sizeof(o_id);
nctx->c_discount_len = 0;
nctx->c_credit_len = 0;
nctx->c_last_len = 0;
nctx->retries_len = sizeof(retries);
nctx->cr_date_len = sizeof(cr_date);
/* this is the row count */
rcount = o_ol_cnt;
nctx->nol_i_count = o_ol_cnt;
nctx->nol_q_count = o_ol_cnt;
nctx->nol_s_count = o_ol_cnt;
nctx->s_remote_count = o_ol_cnt;

nctx->nol_qty_count = 0;
nctx->nol_bg_count = 0;
nctx->nol_item_count = 0;
nctx->nol_name_count = 0;
nctx->nol_am_count = 0;

/* initialization for array operations */
for (i = 0; i < o_ol_cnt; i++) {
    nctx->ol_number[i] = i + 1;
    nctx->nol_i_id_len[i] = sizeof(int);
    nctx->nol_supply_w_id_len[i] = sizeof(int);
    nctx->nol_quantity_len[i] = sizeof(int);
    nctx->nol_amount_len[i] = sizeof(int);
    nctx->ol_o_id_len[i] = sizeof(int);
    nctx->ol_number_len[i] = sizeof(int);
    nctx->ol_dist_info_len[i] = nctx->s_dist_info_len[i];
    nctx->s_remote_len[i] = sizeof(int);
    nctx->s_quant_len[i] = sizeof(int);
    nctx->i_name_len[i] = 0;
    nctx->s_bg_len[i] = 0;
}
for (i = o_ol_cnt; i < NITEMS; i++) {
    nctx->nol_i_id_len[i] = 0;
    nctx->nol_supply_w_id_len[i] = 0;
    nctx->nol_quantity_len[i] = 0;
    nctx->nol_amount_len[i] = 0;
    nctx->ol_o_id_len[i] = 0;
    nctx->ol_number_len[i] = 0;
    nctx->ol_dist_info_len[i] = 0;
    nctx->s_remote_len[i] = 0;
    nctx->s_quant_len[i] = 0;
}

```

```

nctx->i_name_len[i]=0;
nctx->s_bg_len[i] = 0;
}

execstatus = OCISmtExecute(tpcsvc,nctx->curn1,errhp,1,0,0,0,
OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);

if(execstatus != OCI_SUCCESS) {

OCITransRollback(tpcsvc,errhp,OCI_DEFAULT);
errcode = OCIERROR(errhp,execstatus);

if(errcode == NOT_SERIALIZABLE) {
retries++;
userlog("%d] newo_tx ERROR - NOT SERIALIZE doing retry\n",
getpid());
goto retry;
} else if (errcode == RECOVERR) {
userlog("%d] newo_tx ERROR - RECOVERR doing retry\n", getpid());
retries++;
goto retry;
} else if (errcode == SNAPSHOT_TOO_OLD) {
userlog("%d] newo_tx ERROR - SNAPSHOT_TOO_OLD doing
retry\n", getpid());
retries++;
goto retry;
} else {
userlog("%d] newo_tx ERROR %s:%d - execstatus=%d, errcode=
%d\n", getpid(), __FILE__, __LINE__, execstatus, errcode);
sprintf(neworder_p->status_mesg, "Unknown Error");
return -1;
}
}

/* did the txn succeed ? */

if (rcount != o_ol_cnt) {
MOVEBACK(c_credit, 2, neworder_p);
MOVEBACK(c_last, 16, neworder_p);
MOVEBACK(o_id, neworder_p);
// This is an expected case. This is the neworder rollback transaction
// userlog("%d] newo_tx ERROR - rcount %d != o_ol_cnt %d\n",
getpid(), rcount, o_ol_cnt);
sprintf(neworder_p->status_mesg, "Item number is not valid");
return -1;
}

total = 0.0;
for (i = 0; i < o_ol_cnt; i++) {
strcpy(neworder_p->n_items[i].i_name, i_name[i]);
neworder_p->n_items[i].s_quantity = s_quantity[i];
brand_gen[i] = brand_generic[i][0];
neworder_p->n_items[i].brand[0] = brand_gen[i];
neworder_p->n_items[i].brand[1] = '\0';
neworder_p->n_items[i].i_price = ((double)i_price[i]) / 100;

neworder_p->n_items[i].ol_amount = (double)(nol_amount[i])/
100;
total = total + nol_amount[i];
}
total *= ((double)(10000 - c_discount)/10000) *
(1.0 + ((double)(d_tax)/10000) + ((double)(w_tax)/10000));
total = total/100;

/* fill in date for o_entry_d from time in beginning of txn
cvt2myhms((unsigned char *)cr_date, o_entry_d);
*/

MOVEBACK(o_id, neworder_p);
neworder_p->c_discount = ((double)c_discount) / 100;
neworder_p->w_tax = ((double)w_tax) / 100;
neworder_p->d_tax = ((double)d_tax) / 100;
strcpy (neworder_p->o_entry_d, (char*)o_entry_d, 20);
/* MOVEBACK(o_entry_d, 20, neworder_p); */
MOVEBACK(c_credit, 2, neworder_p);
MOVEBACK(c_last, 16, neworder_p);
MOVEBACK(total, neworder_p);

q_sort(indx, neworder_p, 0,o_ol_cnt-1);

/*****
* END BLOCK OF COMMON CODE
*****/

#if ACID
time(timep);
userlog("ACID NEWORDER w_id=%d, d_id=%d, c_id=%d, o_id=%d,
total=%f\n",
w_id, d_id, c_id, o_id, total);
userlog("ACID NEWORDER completed at %s\n", ctime(timep));
#endif
return(0);
}

/* the arrays are initialized based on a successful select from */
/* stock/item. We need to shift the values in the orderline array */
/* one position up to compensate when we have an invalid item */

void
cleanup_newo(code)
int code;
{
int i;

if (nctx)
free (nctx);

OCIHandleFree((dvoid *)nctx->curn1,OCI_HTYPE_STMT);
OCIHandleFree((dvoid *)nctx->curn2,OCI_HTYPE_STMT);
exit(code);
}

void q_sort(int *arr,struct newo_inf *str,int left, int right)
{
int i;

if(left >= right)
return;
for(i=left+1;i<=right;i++)
if(arr[i] < arr[left])
swap(str,left,i);
q_sort(arr,str,left+1,right);
}

void swap(struct newo_inf *str, int i, int j)
{
int temp;
double dtemp;
char tmpstr[25];
char tmpch[2];

temp = indx[i];
indx[i] = indx[j];
indx[j] = temp;

temp = nol_i_id[i];
nol_i_id[i] = nol_i_id[j];
nol_i_id[j] = temp;
temp = nol_supply_w_id[i];
nol_supply_w_id[i] = nol_supply_w_id[j];
nol_supply_w_id[j] = temp;

temp = nol_quantity[i];
nol_quantity[i] = nol_quantity[j];
nol_quantity[j] = temp;

strcpy(tmpstr,str->n_items[i].i_name, 25);
strcpy(str->n_items[i].i_name,str->n_items[j].i_name, 25);
strcpy(str->n_items[j].i_name,tmpstr, 25);

temp = str->n_items[i].s_quantity;
str->n_items[i].s_quantity = str->n_items[j].s_quantity;
str->n_items[j].s_quantity = temp;

/*
tmpch = str->n_items[i].brand;
str->n_items[i].brand= str->n_items[j].brand;
str->n_items[j].brand= tmpch;
*/

strcpy( tmpch ,str->n_items[i].brand, 2);
strcpy(str->n_items[i].brand ,str->n_items[j].brand, 2);
strcpy(str->n_items[j].brand ,tmpch, 2);

dtemp = str->n_items[i].i_price;
str->n_items[i].i_price = str->n_items[j].i_price;
str->n_items[j].i_price = dtemp;
}

```



```

dtemp = str->n_items[i].ol_amount;
str->n_items[i].ol_amount = str->n_items[j].ol_amount;
str->n_items[j].ol_amount = dtemp;
}
int NEWO( TPSVCINFO *rqst)
{
  if (neworder_tx(rqst)) {
    tpreturn(TPFAIL, 0, rqst->data, sizeof(struct newo_inf), 0);
  }
  else {
    tpreturn(TPSUCCESS, 0, rqst->data, sizeof(struct newo_inf), 0);
  }
}

tpcc_srv_orcls.c
/*
 * Copyright (c) 1995 by Sun Microsystems, Inc.
 */
#pragma ident "@(#)tpcsrv_orcls.c 1.17 97/01/02 SMI"
/
=====+
| Copyright (c) 1995 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
=====+
FILENAME
| plord.c
DESCRIPTION
| OCI version (using PL/SQL stored procedure) of
| ORDER STATUS transaction in TPC-C benchmark.
/* Copyright (c) 2002, Oracle Corporation. All rights reserved. */
/*
NAME
tkvcordq.c - OCI version using queues of ORDER STATUS
transaction in TPC-C benchmark.

DESCRIPTION
<short description of facility this file declares/defines>

EXPORT FUNCTION(S)

INTERNAL FUNCTION(S)
<other external functions defined - one-line descriptions>

STATIC FUNCTION(S)
<static functions defined - one-line descriptions>

NOTES
<other useful comments, qualifications, etc.>

```

```

MODIFIED (MM/DD/YY)
xnjie 06/25/02 - queue open cluster join.
heri 05/07/02 - Fix error in cursor.
heri 02/01/02 - Cleanup, remove indicator values and return codes.
lwwang 07/25/01 - Merged lwwang_tpccitrc
lwwang 07/23/01 - fix include
lwwang 07/23/01 - Creation
*** As per Mar 4, 2009 pre-audit: changes for interactive compliance;
added 'static'
=====+
=====*/
#ifndef ORA_TPCC
#define ORA_TPCC
#include "tpcc.h"
#endif

#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>

/* Tuxedo */
#include "atmi.h"
#include "userlog.h"

struct ord_itm_inf {
  int ol_supply_w_id;
  int ol_i_id;
  int ol_quantity;
  double ol_amount;
  char ol_delivery_d[11];
};

struct ord_inf {
  int o_ol_cnt;
  int w_id;
  int d_id;
  int c_id;
  int o_id;
  int o_carrier_id;
  double c_balance;
  char c_first[17];
  char c_middle[3];
  char c_last[17];
  text o_entry_d[20];
  struct ord_itm_inf o_items[15];
};

#define SQLTX1 "alter session set isolation_level = serializable"

#define MOVETO(element, struct_name) element = struct_name->element
#define MOVEBACK(element, struct_name) struct_name->element =
element

```

```

#define MOVEBACK(element, cnt, struct_name) strncpy(struct_name->
element, element, cnt)

/* List of fields in ordstat */
/* This structure should be EXACTLY identical to the one declared in client.h
*/
/* Lists of items on an order */

static int tx_count = 0; /* Transaction counter */

#ifndef ACID
#include <sys/types.h>
#include <time.h>
time_t curtime;
time_t *timep = &curtime;
#endif

/*****
 * BEGIN BLOCK OF COMMON CODE
 *****/
-----
STATIC FUNCTION DECLARATIONS
-----*/

#define SQLCUR0 "SELECT rowid FROM cust \
WHERE c_d_id = :d_id AND c_w_id = :w_id AND c_last =
:c_last \
ORDER BY c_last, c_d_id, c_w_id, c_first"

#define SQLCUR1 "SELECT /*+ USE_NL(cust) INDEX_DESC(ordr iordr2)
*/ \
c_id, c_balance, c_first, c_middle, c_last, \
o_id, o_entry_d, o_carrier_id, o_ol_cnt, ordr.rowid \
FROM cust, ordr \
WHERE cust.rowid = :cust_rowid \
AND o_d_id = c_d_id AND o_w_id = c_w_id AND o_c_id =
c_id \
ORDER BY o_c_id DESC, o_d_id DESC, o_w_id DESC, o_id
DESC"

#define SQLCUR2 "SELECT /*+ USE_NL(cust) INDEX_DESC(ordr
iordr2) */ \
c_balance, c_first, c_middle, c_last, \
o_id, o_entry_d, o_carrier_id, o_ol_cnt, ordr.rowid \
FROM cust, ordr \
WHERE c_id = :c_id AND c_d_id = :d_id AND c_w_id = :w_id \
AND o_d_id = c_d_id AND o_w_id = c_w_id AND o_c_id =
c_id \
ORDER BY o_c_id DESC, o_d_id DESC, o_w_id DESC, o_id
DESC"

#define SQLCUR3 "SELECT /*+ INDEX(ordl) */ \
ol_i_id, ol_supply_w_id, ol_quantity, ol_amount, ol_delivery_d \
FROM ordl \

```

```

WHERE ol_o_id = :o_id AND ol_d_id = :d_id AND ol_w_id =
:w_id"
#endif OLDER_VERSION
#define SQLCUR3 "SELECT /*+ ORDERED USE_NL(ordl)
CLUSTER(ordl) */\
    ol_i_id, ol_supply_w_id, ol_quantity, ol_amount, ol_delivery_d\
FROM ordl, ordl\
WHERE ordl.rowid = :ordl_rowid\
AND o_id = ol_o_id AND ol_d_id = o_d_id AND ol_w_id =
o_w_id"
#endif
#define SQLCUR4 "SELECT count(c_last) FROM cust\
WHERE c_d_id = :d_id AND c_w_id = :w_id AND c_last =
:c_last "

struct ordctx {
    ub2 c_rowid_len[100];
    ub2 ol_supply_w_id_len[NITEMS];
    ub2 ol_i_id_len[NITEMS];
    ub2 ol_quantity_len[NITEMS];
    ub2 ol_amount_len[NITEMS];
    ub2 ol_delivery_d_len[NITEMS];
    ub2 ol_w_id_len;
    ub2 ol_d_id_len;
    ub2 ol_o_id_len;

    ub4 ol_supply_w_id_csize;
    ub4 ol_i_id_csize;
    ub4 ol_quantity_csize;
    ub4 ol_amount_csize;
    ub4 ol_delivery_d_csize;
    ub4 ol_w_id_csize;
    ub4 ol_d_id_csize;
    ub4 ol_o_id_csize;

    OCISmt *curo0;
    OCISmt *curo1;
    OCISmt *curo2;
    OCISmt *curo3;
    OCISmt *curo4;
    OCIBind *c_id_bp;
    OCIBind *w_id_bp[4];
    OCIBind *d_id_bp[4];
    OCIBind *c_last_bp[2];
    OCIBind *o_id_bp;
    OCIBind *c_rowid_bp;
    /* OCIBind *o_rowid_bp; */
    OCIDefine *c_rowid_dp;
    OCIDefine *c_last_dp[2];
    OCIDefine *c_id_dp;
    OCIDefine *c_first_dp[2];
    OCIDefine *c_middle_dp[2];
    OCIDefine *c_balance_dp[2];

    OCIDefine *o_rowid_dp[2];
    OCIDefine *o_id_dp[2];
    OCIDefine *o_entry_d_dp[2];
    OCIDefine *o_cr_id_dp[2];
    OCIDefine *o_ol_cnt_dp[2];
    OCIDefine *ol_d_d_dp;
    OCIDefine *ol_i_id_dp;
    OCIDefine *ol_supply_w_id_dp;
    OCIDefine *ol_quantity_dp;
    OCIDefine *ol_amount_dp;
    OCIDefine *ol_d_base_dp;
    OCIDefine *c_count_dp;
    OCIRowid *c_rowid_ptr[100];
    OCIRowid *c_rowid_cust;
    OCIRowid *o_rowid;

    int cs;
    int cust_idx;
    int norow;
    int rcount;
    int somerows;
};

typedef struct ordctx ordctx;

struct defctx
{
    boolean reexec;
    ub4 count;
};

typedef struct defctx defctx;

static ordctx *octx;

static defctx cbctx;

static OCIDate o_entry_d_base;
static OCIDate ol_d_base[15];
/*
unsigned char o_entry_d_base[7];
unsigned char ol_d_base[15][7];
*/
/* struct ord_inf elements */
static int w_id;
static int d_id;
static int c_id;
static int o_id;
static int o_carrier_id;
static int o_ol_cnt;
static double c_balance;
static char c_first[17];
static char c_middle[3];
static char c_last[17];
/* char o_entry_d[20]; */
static text o_entry_d[20];
static ub4 datelen;

static int ol_supply_w_id[15];

static int ol_i_id[15];
static int ol_quantity[15];
static int ol_amount[15];
static ub4 ol_del_len[15];
static text ol_delivery_d[15][11];

static int retries;
static int retries_serial;
static int retries_recoverr;
static int retries_snapshot;

/*****
* END BLOCK OF COMMON CODE
*****/

int get_ords_tx_cnt()
{
    return tx_count;
}

/*
* Function: init ordstat transaction
* Prepare the ordstat transaction
*/

int init_ords_tx()
{
    /*****
    * BEGIN BLOCK OF COMMON CODE
    *****/

    int i;
    text stmbuf[SQL_BUF_SIZE];

    OCISmt *curi;

    octx = (ordctx *) malloc (sizeof(ordctx));
    memset(octx, (char)0, sizeof(ordctx));
    octx->cs = 1;
    octx->norow = 0;
    octx->somerows = 10;
    /* get the rowid handles */
    OCIERROR(errhp, OCIDescriptorAlloc((dvoid *)tpcenv, (dvoid **)&octx->o_rowid,
        (ub4)OCI_DTYPE_ROWID, (size_t) 0, (dvoid **)0));
    for(i=0; i<100; i++) {
        DISCARD OCIERROR(errhp, OCIDescriptorAlloc(tpcenv,
            (dvoid **)&octx->c_rowid_ptr[i], OCI_DTYPE_ROWID, 0,
            (dvoid **)0));
    }

    DISCARD OCIERROR(errhp,
        OCIHandleAlloc(tpcenv, (dvoid **)&octx->curo0, OCI_HTYPE_STMT, 0,
            (dvoid **)0));
    DISCARD OCIERROR(errhp,

```

```

OCIHandleAlloc(tpcenv,(dvoid**)&octx->curo1,OCI_HTYPE_STMT,0,
(dvoid**0));
DISCARD OCIERROR(errhp,
OCIHandleAlloc(tpcenv,(dvoid**)&octx->curo2,OCI_HTYPE_STMT,0,
(dvoid**0));
DISCARD OCIERROR(errhp,
OCIHandleAlloc(tpcenv,(dvoid**)&octx->curo3,OCI_HTYPE_STMT,0,
(dvoid**0));
DISCARD OCIERROR(errhp,
OCIHandleAlloc(tpcenv,(dvoid**)&octx->curo4,OCI_HTYPE_STMT,0,
(dvoid**0));

/* c_id = 0, use find customer by lastname. Get an array or rowid's back */
DISCARD sprintf((char *) stmbuf, SQLCUR0);
DISCARD OCIERROR(errhp,
OCIStmtPrepare(octx->curo0,errhp,stmbuf,(ub4)strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
DISCARD OCIERROR(errhp,
OCIAttrSet(octx->curo0,OCI_HTYPE_STMT,&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));
/* get order/customer info back based on rowid */
DISCARD sprintf((char *) stmbuf, SQLCUR1);
DISCARD OCIERROR(errhp,
OCIStmtPrepare(octx->curo1,errhp,stmbuf,(ub4)strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
DISCARD OCIERROR(errhp,
OCIAttrSet(octx->curo1,OCI_HTYPE_STMT,&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));

/* c_id == 0, use lastname to find customer */
DISCARD sprintf((char *) stmbuf, SQLCUR2);
DISCARD OCIERROR(errhp,
OCIStmtPrepare(octx->curo2,errhp,stmbuf,(ub4)strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
DISCARD OCIERROR(errhp,
OCIAttrSet(octx->curo2,OCI_HTYPE_STMT,&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));

DISCARD sprintf((char *) stmbuf, SQLCUR3);
DISCARD OCIERROR(errhp,
OCIStmtPrepare(octx->curo3,errhp,stmbuf,(ub4)strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
DISCARD OCIERROR(errhp,
OCIAttrSet(octx->curo3,OCI_HTYPE_STMT,&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));

DISCARD sprintf((char *) stmbuf, SQLCUR4);
DISCARD OCIERROR(errhp,
OCIStmtPrepare(octx->curo4,errhp,stmbuf,(ub4)strlen((char *)stmbuf),
OCI_NTV_SYNTAX,OCI_DEFAULT));
DISCARD OCIERROR(errhp,
OCIAttrSet(octx->curo4,OCI_HTYPE_STMT,&octx->norow,0,
OCI_ATTR_PREFETCH_ROWS,errhp));

for (i = 0; i < NITEMS; i++) {

octx->ol_supply_w_id_len[i] = sizeof(int);
octx->ol_i_id_len[i] = sizeof(int);

octx->ol_quantity_len[i] = sizeof(int);
octx->ol_amount_len[i] = sizeof(int);
octx->ol_delivery_d_len[i] = sizeof(ol_d_base[0]);
}

octx->ol_supply_w_id_csize = NITEMS;
octx->ol_i_id_csize = NITEMS;
octx->ol_quantity_csize = NITEMS;
octx->ol_amount_csize = NITEMS;
octx->ol_delivery_d_csize = NITEMS;
octx->ol_w_id_csize = NITEMS;
octx->ol_o_id_csize = NITEMS;
octx->ol_d_id_csize = NITEMS;
octx->ol_w_id_len = sizeof(int);
octx->ol_d_id_len = sizeof(int);
octx->ol_o_id_len = sizeof(int);

/* bind variables */

/* c_id (customer id) is not known */
OCIBND(octx->curo0,octx->w_id_bp[0],errhp,":w_id",ADR(w_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo0,octx->d_id_bp[0],errhp,":d_id",ADR(d_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo0,octx->c_last_bp[0],errhp,":c_last",c_last,
SIZ(c_last),SQLT_STR);
OCIDFNRA(octx->curo0,octx->c_rowid_dp,errhp,1,octx->c_rowid_ptr,
SIZ(OCIRowid*),SQLT_RDD,NULL,octx->c_rowid_len,NULL);

OCIBND(octx->curo1,octx->c_rowid_bp,errhp,":cust_rowid", &octx-
>c_rowid_cust,
sizeof(octx->c_rowid_ptr[0]),SQLT_RDD);
OCIDF(octx->curo1,octx-
>c_id_dp,errhp,1,ADR(c_id),SIZ(int),SQLT_INT);
OCIDF(octx->curo1,octx->c_balance_dp[0],errhp,2,ADR(c_balance),
SIZ(double),SQLT_FLT);
OCIDF(octx->curo1,octx->c_first_dp[0],errhp,3,c_first,SIZ(c_first)-1,
SQLT_CHR);
OCIDF(octx->curo1,octx->c_middle_dp[0],errhp,4,c_middle,
SIZ(c_middle)-1,SQLT_AFC);
OCIDF(octx->curo1,octx->c_last_dp[0],errhp,5,c_last,SIZ(c_last)-1,
SQLT_CHR);
OCIDF(octx->curo1,octx-
>o_id_dp[0],errhp,6,ADR(o_id),SIZ(int),SQLT_INT);
OCIDF(octx->curo1,octx->o_entry_d_dp[0],errhp,7,
&o_entry_d_base,SIZ(OCIDate),SQLT_ODT);
OCIDF(octx->curo1,octx->o_cr_id_dp[0],errhp,8,ADR(o_carrier_id),
SIZ(int),SQLT_INT);
OCIDF(octx->curo1,octx->o_ol_cnt_dp[0],errhp,9,ADR(o_ol_cnt),
SIZ(int),SQLT_INT);
OCIDF(octx->curo1,octx->o_rowid_dp[0],errhp,10,ADR(octx->o_rowid),SIZ(int),
SIZ(OCIRowid*),SQLT_RDD);

/* Bind for third cursor , no-zero customer id */
OCIBND(octx->curo2,octx->w_id_bp[1],errhp,":w_id",ADR(w_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo2,octx->d_id_bp[1],errhp,":d_id",ADR(d_id),
SIZ(int),SQLT_INT);

OCIBND(octx->curo2,octx->c_id_bp,errhp,":c_id",ADR(c_id),
SIZ(int),SQLT_INT);
OCIDF(octx->curo2,octx->c_balance_dp[1],errhp,1,ADR(c_balance),
SIZ(double),SQLT_FLT);
OCIDF(octx->curo2,octx->c_first_dp[1],errhp,2,c_first,SIZ(c_first)-1,
SQLT_CHR);
OCIDF(octx->curo2,octx->c_middle_dp[1],errhp,3,c_middle,
SIZ(c_middle)-1,SQLT_AFC);
OCIDF(octx->curo2,octx->c_last_dp[1],errhp,4,c_last,SIZ(c_last)-1,
SQLT_CHR);
OCIDF(octx->curo2,octx-
>o_id_dp[1],errhp,5,ADR(o_id),SIZ(int),SQLT_INT);
OCIDF(octx->curo2,octx->o_entry_d_dp[1],errhp,6,&o_entry_d_base,
SIZ(OCIDate),SQLT_ODT);
OCIDF(octx->curo2,octx->o_cr_id_dp[1],errhp,7,ADR(o_carrier_id),
SIZ(int),SQLT_INT);
OCIDF(octx->curo2,octx->o_ol_cnt_dp[1],errhp,8,ADR(o_ol_cnt),
SIZ(int),SQLT_INT);
OCIDF(octx->curo2,octx->o_rowid_dp[1],errhp,9,ADR(octx->o_rowid),
SIZ(OCIRowid*),SQLT_RDD);

/* Bind for last cursor */

OCIBND(octx->curo3,octx->w_id_bp[2],errhp,":w_id",ADR(w_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo3,octx->d_id_bp[2],errhp,":d_id",ADR(d_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo3,octx->o_id_bp,errhp,":o_id",ADR(o_id),
SIZ(int),SQLT_INT);

/*
OCIBND(octx->curo3,octx->c_id_bp,errhp,":c_id",ADR(c_id),
SIZ(int),SQLT_INT);
*/

/*
OCIBND(octx->curo3,octx->o_rowid_bp,errhp,":ordr_rowid",
&octx->o_rowid,SIZ(OCIRowid*),SQLT_RDD);
*/

OCIDFNRA(octx->curo3,octx->ol_i_id_dp,errhp,1,
ol_i_id,SIZ(int),SQLT_INT,
NULL,octx->ol_i_id_len,NULL);
OCIDFNRA(octx->curo3,octx->ol_supply_w_id_dp,errhp,2,
ol_supply_w_id,
SIZ(int),SQLT_INT,NULL,
octx->ol_supply_w_id_len,NULL);
OCIDFNRA(octx->curo3,octx->ol_quantity_dp,errhp,3,
ol_quantity,SIZ(int),
SQLT_INT,NULL,octx->ol_quantity_len,NULL);
OCIDFNRA(octx->curo3,octx->ol_amount_dp,errhp,4,ol_amount,
SQLT_INT,NULL,octx->ol_amount_len,NULL);
OCIDFNRA(octx->curo3,octx-
>ol_d_base_dp,errhp,5,ol_d_base,SIZ(OCIDate),
SQLT_ODT,NULL,octx->ol_delivery_d_len,NULL);

OCIBND(octx->curo4,octx->w_id_bp[3],errhp,":w_id",ADR(w_id),
SIZ(int),SQLT_INT);
OCIBND(octx->curo4,octx->d_id_bp[3],errhp,":d_id",ADR(d_id),

```

```

    SIZ(int),SQLT_INT);
OCIBND(octx->curo4,octx->c_last_bp[1],errhp,":c_last",c_last,
    SIZ(c_last), SQLT_STR);
OCIDEF(octx->curo4,octx->c_count_dp,errhp,1,ADR(octx-
>rcount),SIZ(int),
    SQLT_INT);

return (0);
}

/*****
* END BLOCK OF COMMON CODE
*****/

int ordstat_tx(TPSVCINFO *rqst)
{
    int i;
    int execstatus, rcount,errcode;
    struct ord_inf *ordstat_p;
    ordstat_p = (struct ord_inf *) (rqst->data);

    tx_count++;

#if ACID
    time(timep);
    userlog("ACID ORDSTAT Transaction begun at %s\n",
ctime(timep));
#endif

/*****
* BEGIN BLOCK OF COMMON CODE
*****/
    retries = 0;
    retries_serial = 0;
    retries_recoverr = 0;
    retries_snapshot = 0;

    for (i = 0; i < NITEMS; i++) {
        octx->ol_supply_w_id_len[i] = sizeof(int);
        octx->ol_i_id_len[i] = sizeof(int);
        octx->ol_quantity_len[i] = sizeof(int);
        octx->ol_amount_len[i] = sizeof(int);
        octx->ol_delivery_d_len[i] = sizeof(OCIDate);
    }
    octx->ol_supply_w_id_csize = NITEMS;
    octx->ol_i_id_csize = NITEMS;
    octx->ol_quantity_csize = NITEMS;
    octx->ol_amount_csize = NITEMS;
    octx->ol_delivery_d_csize = NITEMS;

retry:
    MOVETO(w_id, ordstat_p);
    MOVETO(d_id, ordstat_p);
    MOVETO(c_id, ordstat_p);

    if(c_id == 0) // by last name
        strcpy(c_last, ordstat_p->c_last);
        cbctx.reexec = FALSE;
        execstatus=OCISmtExecute(tpsvc,octx->curo0,errhp,100,0,
            NULLP(CONST
OCISnapshot),NULLP(OCISnapshot),OCI_DEFAULT);
        /* will get OCI_NO_DATA if <100 found */
        if ((execstatus != OCI_NO_DATA) && (execstatus != OCI_SUCCESS))
        {
            errcode=OCIERROR(errhp, execstatus);
            if((errcode == NOT_SERIALIZABLE) || (errcode == RECOVERR) ||
(errcode == SNAPSHOT_TOO_OLD))
            {
                DISCARD OCITransCommit(tpsvc,errhp,OCI_DEFAULT);
                retries++;
                goto retry;
            } else {
                userlog("%d] ordstat_tx ERROR %s:%d - EXITING -1, execstatus=%d,
errcode=%d\n", getpid(), __FILE__, __LINE__, execstatus, errcode);
                return -1;
            }
        }
        if (execstatus == OCI_NO_DATA) /* there are no more rows */
        {
            /* get rowcount, find middle one */
            DISCARD OCIAttrGet(octx-
>curo0,OCI_HTYPE_STMT,&rcount,NULL,
                OCI_ATTR_ROW_COUNT,errhp);
            if (rcount < 1)
            {
                userlog("ORDERSTATUS rcount=%d, w_id=%d, d_id=%d, c_id=%d,
c_last=%s, in_w_id=%d, in_d_id=%d, in_c_id=%d, in_c_last=%s.\n",rcount,
w_id, d_id, c_id, c_last,
                ordstat_p->w_id, ordstat_p->d_id, ordstat_p->c_id,
                ordstat_p->c_last);
                return (-1);
            }
            octx->cust_idx=(rcount)/2 ;
        }
        else
        {
            /* count the number of rows */
            execstatus=OCISmtExecute(tpsvc,octx->curo4,errhp,1,0,
                NULLP(CONST
OCISnapshot),NULLP(OCISnapshot),OCI_DEFAULT);
            if ((execstatus != OCI_NO_DATA) && (execstatus != OCI_SUCCESS))
            {
                errcode=OCIERROR(errhp, execstatus);
                if((errcode == NOT_SERIALIZABLE) || (errcode == RECOVERR) ||
                {
                    DISCARD OCITransCommit(tpsvc,errhp,OCI_DEFAULT);
                    retries++;
                    goto retry;
                } else {
                    userlog("%d] ordstat_tx ERROR %s:%d - EXITING -1,
execstatus=%d, errcode=%d\n", getpid(), __FILE__, __LINE__, execstatus,
errcode);
                    return -1;
                }
            }
            octx->cust_idx=0 ;
        }
        octx->c_rowid_cust = octx->c_rowid_ptr[octx->cust_idx];
        execstatus=OCISmtExecute(tpsvc,octx->curo1,errhp,1,0,
            NULLP(CONST
OCISnapshot),NULLP(OCISnapshot),OCI_DEFAULT);
        if (execstatus != OCI_SUCCESS)
        {
            errcode=OCIERROR(errhp,execstatus);
            DISCARD OCITransCommit(tpsvc,errhp,OCI_DEFAULT);
            if((errcode == NOT_SERIALIZABLE) || (errcode == RECOVERR)
                || (errcode == SNAPSHOT_TOO_OLD))
            {
                retries++;
                goto retry;
            } else {
                userlog("%d] ordstat_tx ERROR %s:%d - EXITING
-1, execstatus=%d, errcode=%d\n", getpid(), __FILE__, __LINE__,
execstatus, errcode);
                return -1;
            }
        }
        if (octx->rcount+1 < 2*10 ) {
            octx->cust_idx=(octx->rcount+1)/2 ;
        }
        else
        {
            cbctx.reexec = TRUE;
            cbctx.count = (octx->rcount+1)/2 ;
            execstatus=OCISmtExecute(tpsvc,octx->curo0,errhp,cbctx.count,
                0,NULLP(CONST OCISnapshot),
                NULLP(OCISnapshot),OCI_DEFAULT);
            /* will get OCI_NO_DATA if <100 found */
            if (cbctx.count > 0)
            {
                userlog ("did not get all rows ");
                return (-1);
            }

            if ((execstatus != OCI_NO_DATA) && (execstatus !=
OCI_SUCCESS))
            {
                errcode=OCIERROR(errhp, execstatus);
                if((errcode == NOT_SERIALIZABLE) || (errcode == RECOVERR))
                {
                    DISCARD OCITransCommit(tpsvc,errhp,OCI_DEFAULT);
                    retries++;
                    goto retry;
                } else {
                    userlog("%d] ordstat_tx ERROR %s:%d - EXITING
-1, execstatus=%d, errcode=%d\n", getpid(), __FILE__, __LINE__,
execstatus, errcode);
                    return -1;
                }
            }
            octx->cust_idx=0 ;
        }
        octx->c_rowid_cust = octx->c_rowid_ptr[octx->cust_idx];
        execstatus=OCISmtExecute(tpsvc,octx->curo1,errhp,1,0,
            NULLP(CONST
OCISnapshot),NULLP(OCISnapshot),OCI_DEFAULT);
        if (execstatus != OCI_SUCCESS)
        {
            errcode=OCIERROR(errhp,execstatus);
            DISCARD OCITransCommit(tpsvc,errhp,OCI_DEFAULT);
            if((errcode == NOT_SERIALIZABLE) || (errcode == RECOVERR)
                || (errcode == SNAPSHOT_TOO_OLD))
            {
                retries++;
                goto retry;
            } else {
                userlog("%d] ordstat_tx ERROR %s:%d - EXITING
-1, execstatus=%d, errcode=%d\n", getpid(), __FILE__, __LINE__,
execstatus, errcode);
                return -1;
            }
        }
    }
}

```

```

return -1;
}
}
} else
{
    c_last[1] = '\0';
    execstatus=OCIStmtExecute(tpscvc,octx->curo2,errhp,1,0,
        NULLP(CONST OCISnapshot),NULLP(OCISnapshot),
        OCI_DEFAULT);
    if (execstatus != OCI_SUCCESS)
    {
        errcode=OCIERROR(errhp,execstatus);
        DISCARD OCITransCommit(tpscvc,errhp,OCI_DEFAULT);
        if((errcode == NOT_SERIALIZABLE) || (errcode == RECOVERR)
            || (errcode == SNAPSHOT_TOO_OLD))
        {
            retries++;
            goto retry;
        }
        else
        {
            userlog("%d] ordstat_tx ERROR %s:%d - EXITING -1,
execstatus=%d, errcode=%d\n", getpid(), __FILE__, __LINE__, execstatus,
errcode);
            return -1;
        }
    }
    octx->ol_w_id_len = sizeof(int);
    octx->ol_d_id_len = sizeof(int);
    octx->ol_o_id_len = sizeof(int);

    execstatus = OCIStmtExecute(tpscvc,octx->curo3,errhp,o_ol_cnt,0,
        NULLP(CONST OCISnapshot),NULLP(OCISnapshot),
        OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
    if (execstatus != OCI_SUCCESS)
    {
        errcode=OCIERROR(errhp,execstatus);
        DISCARD OCITransCommit(tpscvc,errhp,OCI_DEFAULT);
        if((errcode == NOT_SERIALIZABLE) || (errcode == RECOVERR)
            || (errcode == SNAPSHOT_TOO_OLD))
        {
            retries++;
            goto retry;
        }
        else
        {
            userlog("%d] ordstat_tx ERROR %s:%d - EXITING -1,
execstatus=%d, errcode=%d\n", getpid(), __FILE__, __LINE__, execstatus,
errcode);
            return -1;
        }
    }

    /* clean up and convert the delivery dates put out debug return data*/
    for (i = 0; i < o_ol_cnt; i++)

```

```

{
    ol_del_len[i]=sizeof(ol_delivery_d[i]);
    DISCARD OCIERROR(errhp,OCIDateToText(errhp,&ol_d_base[i],
        (const text*)SHORTDATE,(ub1)strlen(SHORTDATE),(text*)0,0,
        &ol_del_len[i], ordstat_p->o_items[i].ol_delivery_d));
    /*
        cvtdmy(ol_d_base[i],ol_delivery_d[i]);
    cvtdmyhms(o_entry_d_base, ordstat_p->o_entry_d);
    */
    ordstat_p->o_items[i].ol_supply_w_id = ol_supply_w_id[i];
    ordstat_p->o_items[i].ol_i_id = ol_i_id[i];
    ordstat_p->o_items[i].ol_quantity = ol_quantity[i];
    ordstat_p->o_items[i].ol_amount = (double)(ol_amount[i])/ 100;
}

    datelen = sizeof(ordstat_p->o_entry_d);
    OCIERROR(errhp,
        OCIDateToText(errhp,&o_entry_d_base,
        (text*)FULLDATE,SIZ(FULLDATE),(text*)0,0,
        &datelen, ordstat_p->o_entry_d));
    /******
    * END BLOCK OF COMMON CODE
    *****/

    #if ACID
        time(timep);
        userlog("ACID ORDSTAT for w_id = %d, d_id = %d, c_id = %d,
o_id = %d\n",
            w_id, d_id, c_id, o_id);
        userlog("ACID ORDSTAT Transaction completed at %s\n",
            ctime(timep));
    #endif

    MOVEBACK(o_id, ordstat_p);
    MOVEBACK(o_carrier_id, ordstat_p);
    MOVEBACK(o_ol_cnt, ordstat_p);
    /* MOVEBACK(c_balance, ordstat_p); */
    ordstat_p->c_balance = c_balance / 100; /* convert to dollars &
cents */
    MOVECBACK(c_first, 16, ordstat_p);
    MOVECBACK(c_middle, 2, ordstat_p);
    MOVECBACK(c_last, 16, ordstat_p);
    /* MOVECBACK(o_entry_d, 19, ordstat_p); already
done */
    /* for search by clastname
    */
    MOVEBACK(c_id, ordstat_p);
    return(0);
}

void cleanup_ords(int code)

```

```

{
    if (octx)
        free(octx);

    /* log off */
    exit(code);
}

int ORDS(TPSVCINFO *rqst)
{
    if (ordstat_tx(rqst) )
        tpreturn(TPFAIL, 0, rqst->data, sizeof(struct ord_inf), 0);
    else
        tpreturn(TPSUCCESS, 0, rqst->data, sizeof(struct ord_inf), 0);
}

tpcc_srv_paym.c
/*
 * Copyright (c) 1995 by Sun Microsystems, Inc.
 */

#pragma ident "@(#)tpcso_srv_paym.c 1.17 97/01/02 SMI"

/
*=====+
| Copyright (c) 1995 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
+=====+
| FILENAME
| plpay.c
| DESCRIPTION
| OCI version (using PL/SQL stored procedure) of
| PAYMENT transaction in TPC-C benchmark.
| *** As perf Mar 4, 2009 pre-audit: changes for interactive compliance
+=====+
/*
#define ORA_TPCC
#define ORA_TPCC
#include "tpcc.h"
#endif

#include <stdlib.h>
#include <unistd.h>
#include <signal.h>
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>

/* Tuxedo */
#include "atmi.h"
#include "userlog.h"

```

```

static int tx_count = 0;
char d_city[21];
char d_state[3];
char d_zip[11];
char c_first[17];
char c_middle[3];
char c_last[17];
char c_street_1[21];
char c_street_2[21];
char c_city[21];
char c_state[3];
char c_zip[11];
char c_phone[17];
char c_since[11];
char c_credit[3];
char c_data[201];

static char c_first[17];
static char c_middle[3];
static char c_last[17];
static char c_street_1[21];
static char c_street_2[21];
static char c_city[21];
static char c_state[3];
static char c_zip[10];
static char c_phone[17];
static text c_since_d[11];
static char c_credit[3];
static int retries;

#define MOVETO(element, struct_name) \
        element = struct_name ->

#define MOVEBACK(element, struct_name) \
        struct_name -> element =

#define MOVECTO(element, cnt, struct_name) {
        int i;
        char c_data[201];
        strncpy(element, struct_name ->
element, cnt); \
        element[cnt] = '\0';
        #if ACID
        #include <sys/types.h>
        #include <time.h>
        static time_t curtime;
        static time_t *timep = &curtime;
        #endif
        for(i=0; i<=cnt; i++)
        {
            if(isspace(element[i]))
            {
                #if ACID
                #include <sys/types.h>
                #include <time.h>
                static time_t curtime;
                static time_t *timep = &curtime;
                #endif
                element[i] = '\0';
                break;
            }
        }
        #define MOVECBACK(element, cnt, struct_name) \
        strncpy(struct_name -> element,
element, cnt)
        struct pay_inf {
            int w_id;
            int d_id;
            int c_id;
            int c_w_id;
            int c_d_id;
            double h_amount;
            double c_credit_lim;
            double c_balance;
            double c_discount;
            char h_date[20];
            char w_street_1[21];
            char w_street_2[21];
            char w_city[21];
            char w_state[3];
            char w_zip[11];
            char d_street_1[21];
            char d_street_2[21];
            static int w_id;
            static int d_id;
            static int c_id, bylastname;
            static int c_w_id;
            static int c_d_id;
            static int h_amount;
            static int c_credit_lim;
            static double c_balance;
            static float c_discount;
            /*char h_date[20]; */
            static text h_date[20];
            static char w_street_1[21];
            static char w_street_2[21];
            static char w_city[21];
            static char w_state[3];
            static char w_zip[10];
            static char d_street_1[21];
            static char d_street_2[21];
            static char d_city[21];
            static char d_state[3];
            static char d_zip[10];
            static int w_id;
            static int d_id;
            static int c_id;
            static int c_w_id;
            static int c_d_id;
            static int h_amount;
            static int c_credit_lim;
            static double c_balance;
            static float c_discount;
            /*char h_date[20]; */
            static text h_date[20];
            static char w_street_1[21];
            static char w_street_2[21];
            static char w_city[21];
            static char w_state[3];
            static char w_zip[10];
            static char d_street_1[21];
            static char d_street_2[21];
            static char d_city[21];
            static char d_state[3];
            static char d_zip[10];
            #define SQLTXT_INIT "BEGIN inittpc.init_pay; END;"
            struct payctx {
                OCISmt *curpi;
                OCISmt *curp0;
                OCISmt *curp1;
                OCIBind *w_id_bp[2];
                ub2 w_id_len;
                OCIBind *d_id_bp[2];
                ub2 d_id_len;
                OCIBind *c_w_id_bp[2];
                ub2 c_w_id_len;
                OCIBind *c_d_id_bp[2];
                ub2 c_d_id_len;
                OCIBind *c_id_bp[2];
                ub2 c_id_len;
                OCIBind *h_amount_bp[2];
                ub2 h_amount_len;
                OCIBind *c_last_bp[2];
                ub2 c_last_len;
                OCIBind *w_street_1_bp[2];
                ub2 w_street_1_len;
                OCIBind *w_street_2_bp[2];
                ub2 w_street_2_len;
                OCIBind *w_city_bp[2];
                ub2 w_city_len;
                OCIBind *w_state_bp[2];
                ub2 w_state_len;
                OCIBind *w_zip_bp[2];
                ub2 w_zip_len;
                OCIBind *d_street_1_bp[2];
            }
        }
    }

```

```

ub2 d_street_1_len;
OCIBind *d_street_2_bp[2];
ub2 d_street_2_len;
OCIBind *d_city_bp[2];
ub2 d_city_len;
OCIBind *d_state_bp[2];
ub2 d_state_len;
OCIBind *d_zip_bp[2];
ub2 d_zip_len;
OCIBind *c_first_bp[2];
ub2 c_first_len;
OCIBind *c_middle_bp[2];
ub2 c_middle_len;
OCIBind *c_street_1_bp[2];
ub2 c_street_1_len;
OCIBind *c_street_2_bp[2];
ub2 c_street_2_len;
OCIBind *c_city_bp[2];
ub2 c_city_len;
OCIBind *c_state_bp[2];
ub2 c_state_len;
OCIBind *c_zip_bp[2];
ub2 c_zip_len;
OCIBind *c_phone_bp[2];
ub2 c_phone_len;
OCIBind *c_since_bp[2];
ub2 c_since_len;
OCIBind *c_credit_bp[2];
ub2 c_credit_len;
OCIBind *c_credit_lim_bp[2];
ub2 c_credit_lim_len;
OCIBind *c_discount_bp[2];
ub2 c_discount_len;
OCIBind *c_balance_bp[2];
ub2 c_balance_len;
OCIBind *c_data_bp[2];
ub2 c_data_len;
OCIBind *h_date_bp[2];
ub2 h_date_len;

OCIBind *retries_bp[2];
ub2 retries_len;

OCIBind *cr_date_bp[2];
ub2 cr_date_len;

OCIBind *byln_bp[2];
ub2 byln_len;
};

typedef struct payctx payctx;

static payctx *pctx;

/*****
 * END BLOCK OF COMMON CODE
 *****/

int get_paym_tx_cnt()
{
    return tx_count;
}

/*
 * Function: init payment transaction
 * Prepare the payment transaction
 */

/*****
 * BEGIN BLOCK OF COMMON CODE
 *****/

int init_paym_tx()
{
    text stmbuf[SQL_BUF_SIZE];
    pctx = (payctx *)malloc(sizeof(payctx));
    memset(pctx,(char)0,sizeof(payctx));

    /* cursor for init */
    DISCARD OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&(&(pctx->curpi)),
    OCI_HTYPE_STMT,0,(dvoid**)0));

    DISCARD OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&(&(pctx->curp0)),
    OCI_HTYPE_STMT,0,(dvoid**)0));
    DISCARD OCIERROR(errhp,OCIHandleAlloc(tpcenv, (dvoid **)&(&(pctx->curp1)),
    OCI_HTYPE_STMT,0,(dvoid**)0));

    /* build the init statement and execute it */
    sprintf((char*)stmbuf,SQLTXT_INIT);
    DISCARD OCIERROR(errhp,OCIStmtPrepare(pctx->curpi, errhp, stmbuf,
    strlen((char *)stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT));
    DISCARD OCIERROR(errhp, OCIStmtExecute(tpcenv,pctx->curpi,errhp,1,0,
    NULLP(CONST
    OCI_Snapshot),NULLP(OCI_Snapshot),OCI_DEFAULT));

    /* customer id != 0, go by last name */

    sqlfile("paynz.sql",stmbuf);
    DISCARD OCIERROR(errhp,OCIStmtPrepare(pctx->curp0, errhp, stmbuf,
    strlen((char *)stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT));

    /* customer id == 0, go by last name */

    sqlfile("payz.sql",stmbuf); /* sqlfile opens $O/bench/.../blocks/... */
    DISCARD OCIERROR(errhp,OCIStmtPrepare(pctx->curp1, errhp, stmbuf,
    strlen((char *)stmbuf), OCI_NTV_SYNTAX, OCI_DEFAULT));

    pctx->w_id_len = SIZ(w_id);
    pctx->d_id_len = SIZ(d_id);
    pctx->c_w_id_len = SIZ(c_w_id);
    pctx->c_d_id_len = SIZ(c_d_id);
    pctx->c_id_len = 0;
    pctx->h_amount_len = SIZ(h_amount);
    pctx->c_last_len = 0;
    pctx->w_street_1_len = 0;
    pctx->w_street_2_len = 0;
    pctx->w_city_len = 0;
    pctx->w_state_len = 0;
    pctx->w_zip_len = 0;
    pctx->d_street_1_len = 0;
    pctx->d_street_2_len = 0;
    pctx->d_city_len = 0;
    pctx->d_state_len = 0;
    pctx->d_zip_len = 0;
    pctx->c_first_len = 0;
    pctx->c_middle_len = 0;
    pctx->c_street_1_len = 0;
    pctx->c_street_2_len = 0;
    pctx->c_city_len = 0;
    pctx->c_state_len = 0;
    pctx->c_zip_len = 0;
    pctx->c_phone_len = 0;
    pctx->c_since_len = 0;
    pctx->c_credit_len = 0;
    pctx->c_credit_lim_len = 0;
    pctx->c_discount_len = 0;
    pctx->c_balance_len = sizeof(double);
    pctx->c_data_len = 0;
    pctx->h_date_len = 0;
    pctx->retries_len = SIZ(retries);
    pctx->cr_date_len = 7;

```

```

/* bind variables */

OCIBNDPL(pctx->curp0, pctx->w_id_bp[0],
errhp,":w_id",ADR(w_id),SIZ(int),
  SQLT_INT, NULL);
OCIBNDPL(pctx->curp0, pctx->d_id_bp[0],
errhp,":d_id",ADR(d_id),SIZ(int),
  SQLT_INT, NULL);
OCIBND(pctx->curp0, pctx->c_w_id_bp[0],
errhp,":c_w_id",ADR(c_w_id),SIZ(int),
  SQLT_INT);
OCIBND(pctx->curp0, pctx->c_d_id_bp[0],
errhp,":c_d_id",ADR(c_d_id),SIZ(int),
  SQLT_INT);
OCIBND(pctx->curp0, pctx->c_id_bp[0],
errhp,":c_id",ADR(c_id),SIZ(int),
  SQLT_INT);
OCIBNDPL(pctx->curp0, pctx->h_amount_bp[0],
errhp,":h_amount",ADR(h_amount),
  SIZ(int),SQLT_INT, &pctx->h_amount_len);
OCIBNDPL(pctx->curp0, pctx->c_last_bp[0],
errhp,":c_last",c_last,SIZ(c_last),
  SQLT_STR, &pctx->c_last_len);
OCIBNDPL(pctx->curp0, pctx->w_street_1_bp[0],
errhp,":w_street_1",w_street_1,
  SIZ(w_street_1),SQLT_STR, &pctx->w_street_1_len);
OCIBNDPL(pctx->curp0, pctx->w_street_2_bp[0],
errhp,":w_street_2",w_street_2,
  SIZ(w_street_2),SQLT_STR, &pctx->w_street_2_len);
OCIBNDPL(pctx->curp0, pctx->w_city_bp[0],
errhp,":w_city",w_city,SIZ(w_city),
  SQLT_STR, &pctx->w_city_len);
OCIBNDPL(pctx->curp0, pctx->w_state_bp[0], errhp,":w_state",w_state,
  SIZ(w_state), SQLT_STR, &pctx->w_state_len);
OCIBNDPL(pctx->curp0, pctx->w_zip_bp[0],
errhp,":w_zip",w_zip,SIZ(w_zip),
  SQLT_STR, &pctx->w_zip_len);
OCIBNDPL(pctx->curp0, pctx->d_street_1_bp[0],
errhp,":d_street_1",d_street_1,
  SIZ(d_street_1),SQLT_STR, &pctx->d_street_1_len);
OCIBNDPL(pctx->curp0, pctx->d_street_2_bp[0],
errhp,":d_street_2",d_street_2,
  SIZ(d_street_2),SQLT_STR, &pctx->d_street_2_len);
OCIBNDPL(pctx->curp0, pctx->d_city_bp[0],
errhp,":d_city",d_city,SIZ(d_city),
  SQLT_STR, &pctx->d_city_len);
OCIBNDPL(pctx->curp1, pctx->d_state_bp[1], errhp,":d_state",d_state,
  SIZ(d_state), SQLT_STR, &pctx->d_state_len);
OCIBNDPL(pctx->curp1, pctx->d_zip_bp[1],
errhp,":d_zip",d_zip,SIZ(d_zip),
  SQLT_STR, &pctx->d_zip_len);
OCIBNDPL(pctx->curp1, pctx->c_first_bp[1], errhp,":c_first",c_first,
  SIZ(c_first), SQLT_STR, &pctx->c_first_len);
OCIBNDPL(pctx->curp1, pctx->c_middle_bp[1],
errhp,":c_middle",c_middle,2,
  SQLT_AFC, &pctx->c_middle_len);

OCIBNDPL(pctx->curp1, pctx->c_street_1_bp[1],
errhp,":c_street_1",c_street_1,
  SIZ(c_street_1),SQLT_STR, &pctx->c_street_1_len);
OCIBNDPL(pctx->curp1, pctx->c_street_2_bp[1],
errhp,":c_street_2",c_street_2,
  SIZ(c_street_2),SQLT_STR, &pctx->c_street_2_len);
OCIBNDPL(pctx->curp1, pctx->c_city_bp[1], errhp,":c_city",c_city,
  SIZ(c_city),SQLT_STR, &pctx->c_city_len);
OCIBNDPL(pctx->curp1, pctx->c_state_bp[1], errhp,":c_state",c_state,
  SIZ(c_state), SQLT_STR, &pctx->c_state_len);
OCIBNDPL(pctx->curp1, pctx->c_zip_bp[1],
errhp,":c_zip",c_zip,SIZ(c_zip),
  SQLT_STR, &pctx->c_zip_len);
OCIBNDPL(pctx->curp1, pctx->c_phone_bp[1],
errhp,":c_phone",c_phone,
  SQLT_STR, &pctx->c_phone_len);

OCIBNDPL(pctx->curp0, pctx->c_street_1_bp[0],
errhp,":c_street_1",c_street_1,
  SIZ(c_street_1),SQLT_STR, &pctx->c_street_1_len);
OCIBNDPL(pctx->curp0, pctx->c_street_2_bp[0],
errhp,":c_street_2",c_street_2,
  SIZ(c_street_2),SQLT_STR, &pctx->c_street_2_len);
OCIBNDPL(pctx->curp0, pctx->c_city_bp[0],
errhp,":c_city",c_city,SIZ(c_city),
  SQLT_STR, &pctx->c_city_len);
OCIBNDPL(pctx->curp0, pctx->c_state_bp[0], errhp,":c_state",c_state,
  SIZ(c_state), SQLT_STR, &pctx->c_state_len);
OCIBNDPL(pctx->curp0, pctx->c_zip_bp[0],
errhp,":c_zip",c_zip,SIZ(c_zip),
  SQLT_STR, &pctx->c_zip_len);
OCIBNDPL(pctx->curp0, pctx->c_phone_bp[0],
errhp,":c_phone",c_phone,
  SIZ(c_phone), SQLT_STR, &pctx->c_phone_len);
OCIBNDPL(pctx->curp0, pctx->c_since_bp[0], errhp,":c_since",&c_since,
  SIZ(OCIDate), SQLT_ODT, &pctx->c_since_len);
OCIBNDPL(pctx->curp0, pctx->c_credit_bp[0], errhp,":c_credit",c_credit,
  SIZ(c_credit),SQLT_CHR, &pctx->c_credit_len);
OCIBNDPL(pctx->curp0, pctx->c_credit_lim_bp[0], errhp,":c_credit_lim",
  ADR(c_credit_lim),SIZ(int), SQLT_INT, &pctx->c_credit_lim_len);
OCIBNDPL(pctx->curp0, pctx->c_discount_bp[0], errhp,":c_discount",
  ADR(c_discount),SIZ(c_discount), SQLT_FLT, &pctx-
>c_discount_len);
OCIBNDPL(pctx->curp0, pctx->c_balance_bp[0], errhp,":c_balance",
  ADR(c_balance), SIZ(double),SQLT_FLT, &pctx->c_balance_len);
OCIBNDPL(pctx->curp0, pctx->c_data_bp[0],
errhp,":c_data",c_data,SIZ(c_data),
  SQLT_STR, &pctx->c_data_len);

/*
OCIBNDR(pctx->curp0, pctx->h_date_bp,
errhp,":h_date",h_date,SIZ(h_date),
  SQLT_STR, &pctx->h_date_ind, &pctx->h_date_len, &pctx-
>h_date_rc);
*/
OCIBNDPL(pctx->curp0, pctx->retries_bp[0], errhp,":retry",ADR(retries),
  SIZ(int), SQLT_INT, &pctx->retries_len);
OCIBNDPL(pctx->curp0, pctx->cr_date_bp[0],
errhp,":cr_date",ADR(cr_date),
  SIZ(OCIDate),SQLT_ODT, &pctx->cr_date_len);

/* ---- Binds for the second cursor */

OCIBNDPL(pctx->curp1, pctx->w_id_bp[1],
errhp,":w_id",ADR(w_id),SIZ(int),
  SQLT_INT, &pctx->w_id_len);
OCIBNDPL(pctx->curp1, pctx->d_id_bp[1],
errhp,":d_id",ADR(d_id),SIZ(int),
  SQLT_INT, &pctx->d_id_len);
OCIBND(pctx->curp1, pctx->c_w_id_bp[1],
errhp,":c_w_id",ADR(c_w_id),SIZ(int),
  SQLT_INT);
OCIBND(pctx->curp1, pctx->c_d_id_bp[1],
errhp,":c_d_id",ADR(c_d_id),SIZ(int),
  SQLT_INT);
OCIBNDPL(pctx->curp1, pctx->c_street_1_bp[1],
errhp,":c_street_1",c_street_1,
  SIZ(c_street_1),SQLT_STR, &pctx->c_street_1_len);
OCIBNDPL(pctx->curp1, pctx->c_street_2_bp[1],
errhp,":c_street_2",c_street_2,
  SIZ(c_street_2),SQLT_STR, &pctx->c_street_2_len);
OCIBNDPL(pctx->curp1, pctx->c_city_bp[1], errhp,":c_city",c_city,
  SIZ(c_city),SQLT_STR, &pctx->c_city_len);
OCIBNDPL(pctx->curp1, pctx->c_state_bp[1], errhp,":c_state",c_state,
  SIZ(c_state), SQLT_STR, &pctx->c_state_len);
OCIBNDPL(pctx->curp1, pctx->c_zip_bp[1],
errhp,":c_zip",c_zip,SIZ(c_zip),
  SQLT_STR, &pctx->c_zip_len);
OCIBNDPL(pctx->curp1, pctx->c_phone_bp[1],
errhp,":c_phone",c_phone,
  SQLT_STR, &pctx->c_phone_len);

```



```

    SIZ(c_phone), SQLT_STR, &pctx->c_phone_len);
OCIBNDPL(pctx->curp1, pctx->c_since_bp[1], errhp,":c_since",&c_since,
    SIZ(OCIDate), SQLT_ODT, &pctx->c_since_len);
OCIBNDPL(pctx->curp1, pctx->c_credit_bp[1], errhp,":c_credit",c_credit,
    SIZ(c_credit),SQLT_CHR, &pctx->c_credit_len);
OCIBNDPL(pctx->curp1, pctx->c_credit_lim_bp[1], errhp,":c_credit_lim",
    ADR(c_credit_lim),SIZ(int), SSQLT_INT, &pctx->c_credit_lim_len);
OCIBNDPL(pctx->curp1, pctx->c_discount_bp[1], errhp,":c_discount",
    ADR(c_discount),SIZ(c_discount), SSQLT_FLT, &pctx->
>c_discount_len);
OCIBNDPL(pctx->curp1, pctx->c_balance_bp[1], errhp,":c_balance",
    ADR(c_balance), SIZ(double),SQLT_FLT, &pctx->c_balance_len);
OCIBNDPL(pctx->curp1, pctx->c_data_bp[1],
errhp,":c_data",c_data,SIZ(c_data),
    SSQLT_STR, &pctx->c_data_len);
/*
OCIBNDR(pctx->curp1, pctx->h_date_bp1,
errhp,":h_date",h_date,SIZ(h_date),
    SSQLT_STR, &pctx->h_date_ind, &pctx->h_date_len, &pctx->
>h_date_rc);
*/
OCIBNDPL(pctx->curp1, pctx->retries_bp[1], errhp,":retry",ADR(retries),
    SIZ(int), SSQLT_INT, &pctx->retries_len);
OCIBNDPL(pctx->curp1, pctx->cr_date_bp[1],
errhp,":cr_date",ADR(cr_date),
    SIZ(OCIDate),SQLT_ODT, &pctx->cr_date_len);

return (0);
}

int execstatus,errcode;

/*****
* END BLOCK OF COMMON CODE
*****/

int payment_tx(TPSVCINFO *rqst)
{
ub4 hlen;
ub4 sincelen;

    struct pay_inf *payment_p;
    payment_p = (struct pay_inf *) (rqst->data);
    MOVETO(w_id, payment_p);
    MOVETO(d_id, payment_p);
    MOVETO(c_id, payment_p);
    MOVETO(c_w_id, payment_p);
    MOVETO(c_d_id, payment_p);

    h_amount = (int)(payment_p->h_amount * 100);
    strcpy(c_last, payment_p->c_last);
    tx_count++;

#if ACID
    time(timep);
    userlog("ACID PAYMENT Transaction Begun at %s\n",
ctime(timep));

    if (c_id == 0) {
        bylastname = 1;
        execstatus=OCISmtExecute(tpsvc,pctx->curp1,errhp,1,0,
            NULLP(CONST OCISnapshot),NULLP(OCISnapshot),
            OCI_DEFAULT|OCI_COMMIT_ON_SUCCESS);if(execstatus !=
OCI_SUCCESS) {
            OCITransRollback(tpsvc,errhp,OCI_DEFAULT);
            errcode = OCIERROR(errhp,execstatus);
            if(errcode == NOT_SERIALIZABLE) {
                retries++;
                goto retry;
            } else if (errcode == RECOVER) {
                retries++;
                goto retry;
            } else if (errcode == SNAPSHOT_TOO_OLD) {
                retries++;
                goto retry;
            } else {
                userlog("%d] payment_tx: OCIERROR UNRECOVERABLE %d %d
w_id:%d d_id:%d c_id:%d \n", getpid(), execstatus, errcode, w_id, d_id, c_id
);
                return -1;
            }
        }
        /*
        cvtdmyhms(cr_date,h_date);
        */
        hlen=SIZ(h_date);
        OCIERROR(errhp,OCIDateToText(errhp,&cr_date,
            (text*)FULLDATE,strlen(FULLDATE),(text*)0,0,&hlen,h_date));

        /*
        cvtdmy(c_since,c_since_d);
        */
        sincelen=SIZ(c_since_d);
        OCIERROR(errhp,OCIDateToText(errhp,&c_since,
            (text*)SHORTDATE,strlen(SHORTDATE),
            (text*)0,0,&sincelen,c_since_d));

        /*****
        * END BLOCK OF COMMON CODE
        *****/

        MOVEBACK(c_id, payment_p);
        payment_p->c_credit_lim = ((double)c_credit_lim) / 100;
        payment_p->c_discount = ((double)c_discount) / 100;
        payment_p->c_balance = c_balance / 100; /* convert to dollars &
cents */
        #if ACID
            time(timep);
            userlog("w_id %d, d_id %d, c_id %d, h_amount = %d, c_balance
= %f\n",
                w_id, d_id, c_id, h_amount, c_balance);
            userlog("ACID PAYMENT Transaction completed at %s\n",
ctime(timep));
        #endif
        strcpy(payment_p->c_since, (char *)c_since_d);
        MOVEBACK(h_date, 20, payment_p);
        MOVEBACK(w_street_1, 21, payment_p);
        MOVEBACK(w_street_2, 21, payment_p);
        MOVEBACK(w_city, 21, payment_p);
        MOVEBACK(w_state, 3, payment_p);
        MOVEBACK(w_zip, 11, payment_p);
        MOVEBACK(d_street_1, 21, payment_p);
        MOVEBACK(d_street_2, 21, payment_p);
        MOVEBACK(d_city, 21, payment_p);
        MOVEBACK(d_state, 3, payment_p);
        MOVEBACK(d_zip, 11, payment_p);
        MOVEBACK(c_first, 17, payment_p);
        MOVEBACK(c_middle, 3, payment_p);
        MOVEBACK(c_last, 17, payment_p);
        MOVEBACK(c_street_1, 21, payment_p);

```

```

        MOVECBACK(c_street_2, 21, payment_p);
        MOVECBACK(c_city, 21, payment_p);
        MOVECBACK(c_state, 3, payment_p);
        MOVECBACK(c_zip, 11, payment_p);
        MOVECBACK(c_phone, 17, payment_p);
        MOVECBACK(c_credit, 3, payment_p);
        strncpy(payment_p->c_data, c_data, 201);

        return(0);
    }

int PAYM(TPSVCINFO *rqst)
{
    if (payment_tx(rqst))
        tpreturn(TPFAIL, 0, rqst->data, sizeof(struct pay_inf), 0);
    else
        tpreturn(TPSUCCESS, 0, rqst->data, sizeof(struct pay_inf), 0);
}

tpcc_srv_stock.c
/*
 * Copyright (c) 1994 by Sun Microsystems, Inc.
 */
#pragma ident "@(#)tpcso_srv_stock.c 1.6 95/04/12 SMI"

/
=====
| Copyright (c) 1994 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
=====
| FILENAME
| plsto.c
| DESCRIPTION
| OCI version of STOCK LEVEL transaction in TPC-C benchmark.
=====
/*
*/
#include "ora_oci.h"
#ifdef ORA_TPCC
#define ORA_TPCC
#include "tpcc.h"
#endif

#include <stdlib.h>
#include <signal.h>
#include <stdio.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/msg.h>

/* Tuxedo */
#include "atmi.h"
#include "userlog.h"

static int tx_count = 0;

#define MOVETO(element, struct_name) element = struct_name -> element
#define MOVEBACK(element, struct_name) struct_name -> element = element

/* List of fields in stock */
/* This structure should be EXACTLY identical to the one declared in client.h */
/*
*/
/* List of fields in stock-level */

struct stock_inf {
    int w_id;
    int d_id;
    int threshold;
    int low_stock;
};

#ifdef PLSQLSTO
#define SQLTXT "BEGIN stocklevel.getstocklevel (:w_id, :d_id, :threshold, \
:low_stock); END;"
#else
#define SQLTXT "SELECT /*+ nocache (stok) noparallel (stok) */ count
(DISTINCT s_i_id) \
FROM ordl, stok, dist \
WHERE d_id = :d_id AND d_w_id = :w_id AND \
d_id = ol_d_id AND d_w_id = ol_w_id AND \
ol_i_id = s_i_id AND ol_w_id = s_w_id AND \
s_quantity < :threshold AND \
ol_o_id BETWEEN (d_next_o_id - 20) AND (d_next_o_id - 1) \
order by ol_o_id desc"
/* query using functional index */
/*
*/
#define SQLTXT "SELECT count (DISTINCT s_i_id) \
FROM ordl, stok, dist \
WHERE d_id = :d_id AND d_w_id = :w_id AND \
d_id = ol_d_id AND d_w_id = ol_w_id AND \
ol_o_id BETWEEN (d_next_o_id - 20) AND (d_next_o_id - 1)
AND \
decode(SIGN(s_quantity -21) , -1, s_w_id*100000 + s_i_id,
NULL) \
= ol_w_id*100000 + ol_i_id AND \
s_quantity < :threshold;"
*/
#endif

struct stoctx {
    OCISmt *curs;
    OCIBind *w_id_bp;
    OCIBind *d_id_bp;
    OCIBind *threshold_bp;
#ifdef PLSQLSTO
    OCIBind *low_stock_bp;
#else
    OCIDefine *low_stock_bp;
#endif
};

typedef struct stoctx stoctx;
static stoctx *sctx;

static int w_id;
static int d_id;
static int threshold;
static int low_stock;
static int retries;
static int retries_serial;
static int retries_recover;
static int retries_snapshot;
/*
*/
/* Initialize transaction */
/*
*/
int get_stock_tx_cnt()
{
    return tx_count;
}

int init_stock_tx()
{
}

/*****
* BEGIN BLOCK OF COMMON CODE
*****/

text stmbuf[SQL_BUF_SIZE];
#ifdef MULTI_SVR
/* For all servers - common routine to open/init session etc. */
TPCinit();
#endif
sctx = (stoctx *)malloc(sizeof(stoctx));
memset(sctx, (char)0, sizeof(stoctx));
sctx->norow = 0;

OCIERROR(errhp,
OCIHandleAlloc(tpcenv, (dvoid**)&sctx->curs, OCI_HTYPE_STMT, 0,
(dvoid**)0);
sprintf((char *) stmbuf, SQLTXT);
OCIERROR(errhp, OCISmtPrepare(sctx->curs, errhp, stmbuf, strlen((char
*)stmbuf),
OCI_NTV_SYNTAX, OCI_DEFAULT));
#endif
/* bind variables */
OCIBND(sctx->curs, sctx->w_id_bp, errhp, ":w_id", ADR(w_id), sizeof(int),
SQLT_INT);
OCIBND(sctx->curs, sctx->d_id_bp, errhp, ":d_id", ADR(d_id), sizeof(int),

```

```

SQLT_INT);
OCIBND(sctx->curs,sctx->threshold_bp,errhp,":threshold",
ADR(threshold),
sizeof(int),SQLT_INT);
#ifdef PLSQLSTO
OCIBND(sctx->curs,sctx->low_stock_bp,errhp,":low_stock",
ADR(low_stock),
sizeof(int),SQLT_INT);
#else
OCIDEFINE(sctx->curs,sctx->low_stock_bp,errhp,1,ADR(low_stock),
sizeof(int),SQLT_INT);
#endif

return(0);

/*****
* END BLOCK OF COMMON CODE
*****/
}

/*
* Function: do stocklevel transaction
* Input is the stocklevel structure. Output is low_stock field
*/
int stocklevel_tx( TPSVCINFO *rqst )
{
int err, execstatus,errcode;
struct stock_inf *stocklevel_p;
stocklevel_p = (struct stock_inf *) (rqst->data);

MOVETO(w_id, stocklevel_p);
MOVETO(d_id, stocklevel_p);
MOVETO(threshold, stocklevel_p);
/*****
* BEGIN BLOCK OF COMMON CODE
*****/

tx_count++;
retry:
execstatus= OCISmtExecute(tpscvc,sctx->curs,errhp,1,0,0,0,
OCI_COMMIT_ON_SUCCESS | OCI_DEFAULT);
if (execstatus != OCI_SUCCESS)
{
errcode=OCIERROR(errhp,execstatus);
OCITransCommit(tpscvc,errhp,OCI_DEFAULT);
if((errcode == NOT_SERIALIZABLE) || (errcode == RECOVERR)
|| (errcode == SNAPSHOT_TOO_OLD))
{
retries++;
if (errcode == NOT_SERIALIZABLE)
retries_serial++;
else if (errcode == RECOVERR)
retries_recoverr++;
else
retries_snapshot++;
goto retry;
} else {
userlog("%d] stocklevel_tx ERROR %s:%d - EXITING -1, execstatus=
%d, errcode=%d, w_id=%d, d_id=%d, threshold=%d\n", getpid(),
__FILE__, __LINE__, execstatus, errcode,
stocklevel_p->w_id, stocklevel_p->d_id, stocklevel_p-
>threshold);
return -1;
}
}

/*****
* END BLOCK OF COMMON CODE
*****/

MOVEBACK(low_stock, stocklevel_p);
return(0);
}

void cleanup_stock( int code)
{
/* log off */
if (sctx) free (sctx);

exit(code);
}

int STOCK( TPSVCINFO *rqst)
{
if(stocklevel_tx(rqst)) {
tpreturn(TPFAIL, 0, rqst->data, sizeof(struct
stock_inf), 0);
} else {
tpreturn(TPSUCCESS, 0, rqst->data, sizeof(struct
stock_inf), 0);
}
}

tpcc_srv_util.c
/*
* Copyright (c) 1995 by Sun Microsystems, Inc.
*/
#pragma ident "@(#)tpcso_srv_util.c 1.17 97/01/02 SMI"

/
*=====
| Copyright (c) 1995 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
+=====
/* Common utility functions used by all tpcso_srv* programs */

#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>

#include <sys/file.h>
#include "ora_oci.h"
#include "ora_err.h"

FILE *vopen(fnam,mode)
char *fnam;
char *mode;
{
FILE *fd;

#ifdef DEBUG
fprintf(stderr, "tkvopen() fnam: %s, mode: %s\n", fnam, mode);
#endif

fd = fopen((char *)fnam,(char *)mode);
if (!fd){
fprintf(stderr," fopen on %s failed %d\n",fnam,fd);
exit(-1);
}
return(fd);
}

/*
int sqlfile(fnam,linebuf)
char *fnam;
text *linebuf;
*/
int sqlfile (char *fnam, text *linebuf)
{
FILE *fd;
int nulpt = 0;

#ifdef DEBUG
fprintf(stderr, "sqlfile() fnam: %s, linebuf: %s\n", fnam, linebuf);
#endif

fd = vopen(fnam,"r");
while (fgets((char *)linebuf+nulpt, SQL_BUF_SIZE,fd))
{
nulpt = strlen((char *)linebuf);
}
return(nulpt);
}

void vgetdate (unsigned char *oradt)
{
struct tm *loctime;
time_t int_time;

struct ORADATE {
unsigned char century;
unsigned char year;
unsigned char month;
unsigned char day;
unsigned char hour;
unsigned char minute;
}

```

```

unsigned char    second;
} Date;
int century;
int cnvrtOK;

/* assume convert is successful */
cnvrtOK = 1;

/* get the current date and time as an integer */
time( &int_time);

/* Convert the current date and time into local time */
loctime = localtime( &int_time);

century = (1900+loctime->tm_year) / 100;

Date.century = (unsigned char)(century + 100);
if (Date.century < 119 || Date.century > 120) cnvrtOK = 0;
Date.year = (unsigned char)(loctime->tm_year+100);
if (Date.year < 100 || Date.year > 199) cnvrtOK = 0;
Date.month = (unsigned char)(loctime->tm_mon + 1);
if (Date.month < 1 || Date.month > 12) cnvrtOK = 0;
Date.day = (unsigned char)loctime->tm_mday;
if (Date.day < 1 || Date.day > 31) cnvrtOK = 0;
Date.hour = (unsigned char)(loctime->tm_hour + 1);
if (Date.hour < 1 || Date.hour > 24) cnvrtOK = 0;
Date.minute= (unsigned char)(loctime->tm_min + 1);
if (Date.minute < 1 || Date.minute > 60) cnvrtOK = 0;
Date.second= (unsigned char)(loctime->tm_sec + 1);
if (Date.second < 1 || Date.second > 60) cnvrtOK = 0;

if (cnvrtOK)
  memcpy(oradt,&Date,7);
else
  *oradt = '\0';

return;
}

```

```

void cvtdmy (unsigned char *oradt, char *outdate)
{

```

```

  struct ORADATE {
    unsigned char  century;
    unsigned char  year;
    unsigned char  month;
    unsigned char  day;
    unsigned char  hour;
    unsigned char  minute;
    unsigned char  second;
  } Date;

```

```

  int day,month,year;

```

```

  memcpy(&Date,oradt,7);

  year = (Date.century-100)*100 + Date.year-100;
  month = Date.month;
  day = Date.day;
  sprintf(outdate,"%02d-%02d-%4d",day,month,year);

  return;
}

```

```

void cvtdmyhms (unsigned char *oradt, char *outdate)
{

```

```

  struct ORADATE {
    unsigned char  century;
    unsigned char  year;
    unsigned char  month;
    unsigned char  day;
    unsigned char  hour;
    unsigned char  minute;
    unsigned char  second;
  } Date;

```

```

  int day,month,year;
  int hour,min,sec;

```

```

  memcpy(&Date,oradt,7);

```

```

  year = (Date.century-100)*100 + Date.year-100;
  month = Date.month;
  day = Date.day;
  hour = Date.hour - 1;
  min = Date.minute - 1;
  sec = Date.second - 1;

```

```

  sprintf(outdate,"%02d-%02d-%4d %02d:%02d:%02d",
    day,month,year,hour,min,sec);

```

```

  return;
}

```

## Stored Procedures

### paynz.sql

```

DECLARE /* paynz */
  not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
  deadlock EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
  snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
BEGIN
  LOOP BEGIN
    UPDATE ware
      SET w_ytd = w_ytd + :h_amount
      WHERE w_id = :w_id

```

```

RETURNING w_name, w_street_1, w_street_2, w_city, w_state, w_zip
  INTO inittpc.ware_name, :w_street_1, :w_street_2, :w_city,
    :w_state, :w_zip;

```

```

UPDATE cust

```

```

  SET c_balance = c_balance - :h_amount,
    c_ytd_payment = c_ytd_payment + :h_amount,
    c_payment_cnt = c_payment_cnt+1
  WHERE c_id = :c_id AND c_d_id = :c_d_id AND
    c_w_id = :c_w_id

```

```

RETURNING rowid, c_first, c_middle, c_last, c_street_1,
  c_street_2, c_city, c_state, c_zip, c_phone,
  c_since, c_credit, c_credit_lim,
  c_discount, c_balance
  INTO inittpc.cust_rowid, :c_first, :c_middle, :c_last, :c_street_1,
    :c_street_2, :c_city, :c_state, :c_zip, :c_phone,
    :c_since, :c_credit, :c_credit_lim,
    :c_discount, :c_balance;
IF SQL%NOTFOUND THEN
  raise NO_DATA_FOUND;
END IF;

```

```

IF :c_credit = 'BC' THEN

```

```

  UPDATE cust

```

```

    SET c_data = substr ((to_char (:c_id) || ' ' ||
      to_char (:c_d_id) || ' ' ||
      to_char (:c_w_id) || ' ' ||
      to_char (:d_id) || ' ' ||
      to_char (:w_id) || ' ' ||
      to_char (:h_amount/100, '9999.99') || ' ')
      || c_data, 1, 500)

```

```

  WHERE rowid = inittpc.cust_rowid
RETURNING substr(c_data,1, 200)
  INTO :c_data;

```

```

END IF;

```

```

UPDATE dist

```

```

  SET d_ytd = d_ytd + :h_amount
  WHERE d_id = :d_id
  AND d_w_id = :w_id

```

```

RETURNING d_name, d_street_1, d_street_2, d_city, d_state, d_zip
  INTO inittpc.dist_name, :d_street_1, :d_street_2, :d_city, :d_state,
    :d_zip;
IF SQL%NOTFOUND THEN
  raise NO_DATA_FOUND;
END IF;

```

```

INSERT INTO hist (h_c_id, h_c_d_id, h_c_w_id, h_d_id, h_w_id,
  h_amount, h_date, h_data)

```

```

VALUES

```

```

  (:c_id, :c_d_id, :c_w_id, :d_id, :w_id, :h_amount,
  :cr_date, inittpc.ware_name || ' ' || inittpc.dist_name);
EXIT;

```

```

EXCEPTION

```

```

WHEN not_serializable OR deadlock OR snapshot_too_old THEN
  ROLLBACK;
  :retry := :retry + 1;
END;

END LOOP;
END;

payz.sql
DECLARE /* payz */
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
BEGIN
LOOP BEGIN
UPDATE ware
SET w_ytd = w_ytd+h_amount
WHERE w_id = :w_id
RETURNING w_name,
w_street_1, w_street_2, w_city, w_state, w_zip
INTO inittpc.ware_name,
:w_street_1, :w_street_2, :w_city, :w_state, :w_zip;

SELECT rowid
BULK COLLECT INTO inittpc.row_id
FROM cust
WHERE c_d_id = :c_d_id AND c_w_id = :c_w_id AND c_last = :c_last
ORDER BY c_last, c_d_id, c_w_id, c_first;

inittpc.c_num := sql%rowcount;
inittpc.cust_rowid := inittpc.row_id((inittpc.c_num+1) / 2);

UPDATE cust
SET c_balance = c_balance - :h_amount,
c_ytd_payment = c_ytd_payment+ :h_amount,
c_payment_cnt = c_payment_cnt+1
WHERE rowid = inittpc.cust_rowid
RETURNING
c_id, c_first, c_middle, c_last, c_street_1, c_street_2,
c_city, c_state, c_zip, c_phone,
c_since, c_credit, c_credit_lim,
c_discount, c_balance
INTO :c_id, :c_first, :c_middle, :c_last,
:c_street_1, :c_street_2, :c_city, :c_state,
:c_zip, :c_phone, :c_since, :c_credit,
:c_credit_lim, :c_discount, :c_balance;

:c_data := '';
IF :c_credit = 'BC' THEN
UPDATE cust
SET c_data = substr ((to_char (:c_id) || '' ||
to_char (:c_d_id) || '' ||
to_char (:c_w_id) || '' ||
to_char (:d_id) || '' ||
to_char (:w_id) || '' ||
to_char (:h_amount/100, '9999.99') || '' )
|| c_data, 1, 500)
WHERE rowid = inittpc.cust_rowid
RETURNING substr(c_data,1, 200)
INTO :c_data;

END IF;

UPDATE dist
SET d_ytd = d_ytd+h_amount
WHERE d_id = :d_id
AND d_w_id = :w_id
RETURNING d_name, d_street_1, d_street_2, d_city,
d_state, d_zip
INTO inittpc.dist_name, :d_street_1, :d_street_2, :d_city,
:d_state, :d_zip;

IF SQL%NOTFOUND
THEN
raise NO_DATA_FOUND;
END IF;

INSERT INTO hist (h_c_id, h_c_d_id, h_c_w_id, h_d_id, h_w_id,
h_amount, h_date, h_data)
VALUES (:c_id, :c_d_id, :c_w_id, :d_id, :w_id, :h_amount,
:cr_date, inittpc.ware_name || ' ' || inittpc.dist_name);

EXIT;

EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old THEN
ROLLBACK;
:retry := :retry + 1;
END;

END LOOP;
END;

tkvcpdel.sql
declare
TYPE numarray IS TABLE OF NUMBER INDEX BY
BINARY_INTEGER;
TYPE numlist is varray (10) of number;
dist numarray;
amt numarray ;
cnt pls_integer;

not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable, -8177);
deadlock EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock, -60);
snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old, -1555);

BEGIN
LOOP BEGIN
FORALL d IN 1..10
DELETE FROM nord N
WHERE no_d_id = inittpc.dist(d)
AND no_w_id = :w_id
AND no_o_id = (select min (no_o_id)
from nord
where no_d_id = N.no_d_id
and no_w_id = N.no_w_id)
RETURNING no_d_id, no_o_id BULK COLLECT INTO :d_id,
:order_id;

:ordcnt := SQL%ROWCOUNT;

FORALL o in 1.. :ordcnt
UPDATE ordr SET o_carrier_id = :carrier_id
WHERE o_id = :order_id (o)
AND o_d_id = :d_id(o)
AND o_w_id = :w_id
RETURNING o_c_id BULK COLLECT INTO :o_c_id;

FORALL o in 1.. :ordcnt
UPDATE ordl SET ol_delivery_d = :now
WHERE ol_w_id = :w_id
AND ol_d_id = :d_id(o)
AND ol_o_id = :order_id(o)
RETURNING sum(ol_amount) BULK COLLECT INTO :sums;

FORALL c IN 1.. :ordcnt
UPDATE cust
SET c_balance = c_balance + :sums(c),
c_delivery_cnt = c_delivery_cnt + 1
WHERE c_w_id = :w_id
AND c_d_id = :d_id(c)
AND c_id = :o_c_id(c);
COMMIT;
EXIT;
EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old
THEN
ROLLBACK;
:retry := :retry + 1;
END;

END LOOP; -- for retry
END;

tpvcpnew.sql
-- New Order Anonymous block

DECLARE
idx PLS_INTEGER;
dummy_local PLS_INTEGER;
cache_ol_cnt PLS_INTEGER;
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old EXCEPTION;

```

```

PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);

PROCEDURE u1 IS
BEGIN
  FORALL idx IN 1 .. cache_ol_cnt
    UPDATE stock_item
      SET s_order_cnt = s_order_cnt + 1,
          s_ytd = s_ytd + :ol_quantity(idx),
          s_remote_cnt = s_remote_cnt + :s_remote(idx),
          s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
                          THEN s_quantity +91
                          ELSE s_quantity
                        END) - :ol_quantity(idx)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_01,
              i_price*:ol_quantity(idx),
              CASE WHEN i_data NOT LIKE '%ORIGINAL%'
                THEN 'G'
                ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
                        THEN 'G'
                        ELSE 'B'
                      END)
              END
    BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
                      :ol_amount,:brand_generic;
END u1;

PROCEDURE u2 IS
BEGIN
  FORALL idx IN 1 .. cache_ol_cnt
    UPDATE stock_item
      SET s_order_cnt = s_order_cnt + 1,
          s_ytd = s_ytd + :ol_quantity(idx),
          s_remote_cnt = s_remote_cnt + :s_remote(idx),
          s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
                          THEN s_quantity +91
                          ELSE s_quantity
                        END) - :ol_quantity(idx)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_02,
              i_price*:ol_quantity(idx),
              CASE WHEN i_data NOT LIKE '%ORIGINAL%'
                THEN 'G'
                ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
                        THEN 'G'
                        ELSE 'B'
                      END)
              END
    BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
                      :ol_amount,:brand_generic;
END u2;

PROCEDURE u3 IS
BEGIN
  FORALL idx IN 1 .. cache_ol_cnt
    UPDATE stock_item
      SET s_order_cnt = s_order_cnt + 1,
          s_ytd = s_ytd + :ol_quantity(idx),
          s_remote_cnt = s_remote_cnt + :s_remote(idx),
          s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
                          THEN s_quantity +91
                          ELSE s_quantity
                        END) - :ol_quantity(idx)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_03,
              i_price*:ol_quantity(idx),
              CASE WHEN i_data NOT LIKE '%ORIGINAL%'
                THEN 'G'
                ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
                        THEN 'G'
                        ELSE 'B'
                      END)
              END
    BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
                      :ol_amount,:brand_generic;
END u3;

PROCEDURE u4 IS
BEGIN
  FORALL idx IN 1 .. cache_ol_cnt
    UPDATE stock_item
      SET s_order_cnt = s_order_cnt + 1,
          s_ytd = s_ytd + :ol_quantity(idx),
          s_remote_cnt = s_remote_cnt + :s_remote(idx),
          s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
                          THEN s_quantity +91
                          ELSE s_quantity
                        END) - :ol_quantity(idx)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_04,
              i_price*:ol_quantity(idx),
              CASE WHEN i_data NOT LIKE '%ORIGINAL%'
                THEN 'G'
                ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
                        THEN 'G'
                        ELSE 'B'
                      END)
              END
    BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
                      :ol_amount,:brand_generic;
END u4;

PROCEDURE u5 IS
BEGIN
  FORALL idx IN 1 .. cache_ol_cnt
    UPDATE stock_item
      SET s_order_cnt = s_order_cnt + 1,
          s_ytd = s_ytd + :ol_quantity(idx),
          s_remote_cnt = s_remote_cnt + :s_remote(idx),
          s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
                          THEN s_quantity +91
                          ELSE s_quantity
                        END) - :ol_quantity(idx)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_05,
              i_price*:ol_quantity(idx),
              CASE WHEN i_data NOT LIKE '%ORIGINAL%'
                THEN 'G'
                ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
                        THEN 'G'
                        ELSE 'B'
                      END)
              END
    BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
                      :ol_amount,:brand_generic;
END u5;

PROCEDURE u6 IS
BEGIN
  FORALL idx IN 1 .. cache_ol_cnt
    UPDATE stock_item
      SET s_order_cnt = s_order_cnt + 1,
          s_ytd = s_ytd + :ol_quantity(idx),
          s_remote_cnt = s_remote_cnt + :s_remote(idx),
          s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
                          THEN s_quantity +91
                          ELSE s_quantity
                        END) - :ol_quantity(idx)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_06,
              i_price*:ol_quantity(idx),
              CASE WHEN i_data NOT LIKE '%ORIGINAL%'
                THEN 'G'
                ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
                        THEN 'G'
                        ELSE 'B'
                      END)
              END
    BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
                      :ol_amount,:brand_generic;
END u6;

PROCEDURE u7 IS
BEGIN
  FORALL idx IN 1 .. cache_ol_cnt
    UPDATE stock_item
      SET s_order_cnt = s_order_cnt + 1,
          s_ytd = s_ytd + :ol_quantity(idx),
          s_remote_cnt = s_remote_cnt + :s_remote(idx),
          s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
                          THEN s_quantity +91
                          ELSE s_quantity
                        END) - :ol_quantity(idx)
    WHERE i_id = :ol_i_id(idx)
    AND s_w_id = :ol_supply_w_id(idx)
    RETURNING i_price, i_name, s_quantity, s_dist_07,
              i_price*:ol_quantity(idx),
              CASE WHEN i_data NOT LIKE '%ORIGINAL%'
                THEN 'G'
                ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
                        THEN 'G'
                        ELSE 'B'
                      END)
              END
    BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
                      :ol_amount,:brand_generic;
END u7;

```

```

THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
:ol_amount,:brand_generic;
END u7;

PROCEDURE u8 IS
BEGIN
FORALL idx IN 1 .. cache_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + :ol_quantity(idx),
s_remote_cnt = s_remote_cnt + :s_remote(idx),
s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - :ol_quantity(idx)
WHERE i_id = :ol_i_id(idx)
AND s_w_id = :ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_08,
i_price* :ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
:ol_amount,:brand_generic;
END u8;

PROCEDURE u9 IS
BEGIN
FORALL idx IN 1 .. cache_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + :ol_quantity(idx),
s_remote_cnt = s_remote_cnt + :s_remote(idx),
s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - :ol_quantity(idx)
WHERE i_id = :ol_i_id(idx)
AND s_w_id = :ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_09,
i_price* :ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
END
BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
:ol_amount,:brand_generic;
END u9;

PROCEDURE u10 IS
BEGIN
FORALL idx IN 1 .. cache_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + :ol_quantity(idx),
s_remote_cnt = s_remote_cnt + :s_remote(idx),
s_quantity = (CASE WHEN s_quantity < :ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - :ol_quantity(idx)
WHERE i_id = :ol_i_id(idx)
AND s_w_id = :ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_10,
i_price* :ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO :i_price, :i_name, :s_quantity, inittpc.s_dist,
:ol_amount,:brand_generic;
END u10;

PROCEDURE fix_items IS
rows_lost PLS_INTEGER;
max_index PLS_INTEGER;
temp_index PLS_INTEGER;
BEGIN
idx := 1;
rows_lost := 0;
max_index := dummy_local;

WHILE (max_index != cache_ol_cnt) LOOP

WHILE (idx <= sql%rowcount AND
sql%bulk_rowcount(idx + rows_lost) = 1)
LOOP
idx := idx + 1;
END LOOP;

temp_index := max_index;
WHILE (temp_index >= idx + rows_lost) LOOP
:ol_amount(temp_index + 1) := :ol_amount(temp_index);
:i_price(temp_index + 1) := :i_price(temp_index);
:i_name(temp_index + 1) := :i_name(temp_index);
:s_quantity(temp_index + 1) := :s_quantity(temp_index);
inittpc.s_dist(temp_index + 1) := inittpc.s_dist(temp_index);
:brand_generic(temp_index + 1) := :brand_generic(temp_index);
temp_index := temp_index - 1;
END LOOP;

END fix_items;

IF (idx + rows_lost <= cache_ol_cnt) THEN
:i_price(idx + rows_lost) := 0;
:i_name(idx + rows_lost) := 'NO ITEM';
:s_quantity(idx + rows_lost) := 0;
inittpc.s_dist(idx + rows_lost) := NULL;
:brand_generic(idx + rows_lost) := '';
:ol_amount(idx + rows_lost) := 0;
rows_lost := rows_lost + 1;
max_index := max_index + 1;
END IF;

END LOOP;
END fix_items;

BEGIN
LOOP BEGIN
cache_ol_cnt := :o_ol_cnt;

UPDATE dist SET d_next_o_id = d_next_o_id + 1
WHERE d_id = :d_id AND d_w_id = :w_id
RETURNING d_tax, d_next_o_id-1
INTO :d_tax, :o_id;

SELECT c_discount, c_last, c_credit
INTO :c_discount, :c_last, :c_credit
FROM cust
WHERE c_id = :c_id AND c_d_id = :d_id AND c_w_id = :w_id;

SELECT w_tax
INTO :w_tax
FROM ware
WHERE w_id = :w_id;

INSERT INTO nord (no_o_id, no_d_id, no_w_id)
VALUES (:o_id, :d_id, :w_id);

INSERT INTO ordr (o_id, o_d_id, o_w_id, o_c_id, o_entry_d,
o_carrier_id, o_ol_cnt, o_all_local)
VALUES (:o_id, :d_id, :w_id, :c_id,
:cr_date, 11, :o_ol_cnt, :o_all_local);

dummy_local := :d_id;

IF (dummy_local < 6) THEN
IF (dummy_local < 3) THEN
IF (dummy_local = 1) THEN
u1;
ELSE
u2;
END IF;
ELSE
IF (dummy_local = 3) THEN
u3;
ELSIF (dummy_local = 4) then
u4;
ELSE

```

```

    u5;
  END IF;
END IF;
ELSE
  IF (dummy_local < 8) THEN
    IF (dummy_local = 6) THEN
      u6;
    ELSE
      u7;
    END IF;
  ELSE
    IF (dummy_local = 8) THEN
      u8;
    ELSIF (dummy_local = 9) then
      u9;
    ELSE
      u10;
    END IF;
  END IF;
END IF;

dummy_local := sql%rowcount;

IF (dummy_local != cache_ol_cnt ) THEN fix_items; END IF;

FORALL idx IN 1..dummy_local
  INSERT INTO ordl
    (ol_o_id,ol_d_id,ol_w_id,ol_number,ol_delivery_d,ol_i_id,
     ol_supply_w_id,ol_quantity,ol_amount,ol_dist_info)
  VALUES (:o_id,:d_id,:w_id, inittpc.idx1arr(idx), inittpc.nulldate,
          :ol_i_id(idx), :ol_supply_w_id(idx),
          :ol_quantity(idx), :ol_amount(idx), inittpc.s_dist(idx));

IF (dummy_local != :o_ol_cnt) THEN
:o_ol_cnt := dummy_local;
  ROLLBACK;
END IF;

EXIT;

EXCEPTION
  WHEN not_serializable OR deadlock OR snapshot_too_old THEN
    ROLLBACK;
    :retry := :retry + 1;
  END;
END LOOP;
END;
```



## Appendix B: Database Build

```
analyze.sql
spool analyze.log;
set echo on;

connect tpcc/tpcc

execute dbms_stats.GATHER_TABLE_STATS (OWNNAME=>'TPCC', -
    TABNAME=>'STOK', -
    PARTNAME=>NULL, -
    ESTIMATE_PERCENT=>1, -
    BLOCK_SAMPLE=>TRUE, -
    METHOD_OPT=>'FOR ALL COLUMNS SIZE 1', -
    DEGREE=>160, -
    GRANULARITY=>'DEFAULT', -
    CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS (OWNNAME=>'TPCC', -
    TABNAME=>'CUST', -
    PARTNAME=>NULL, -
    ESTIMATE_PERCENT=>1, -
    BLOCK_SAMPLE=>TRUE, -
    METHOD_OPT=>'FOR ALL COLUMNS SIZE 1', -
    DEGREE=>160, -
    GRANULARITY=>'DEFAULT', -
    CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS (OWNNAME=>'TPCC', -
    TABNAME=>'ORDR', -
    PARTNAME=>NULL, -
    ESTIMATE_PERCENT=>1, -
    BLOCK_SAMPLE=>TRUE, -
    METHOD_OPT=>'FOR ALL COLUMNS SIZE 1', -
    DEGREE=>160, -
    GRANULARITY=>'DEFAULT', -
    CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS (OWNNAME=>'TPCC', -
    TABNAME=>'ORDL', -
    PARTNAME=>NULL, -
    ESTIMATE_PERCENT=>1, -
    BLOCK_SAMPLE=>TRUE, -
    METHOD_OPT=>'FOR ALL COLUMNS SIZE 1', -
    DEGREE=>160, -
    GRANULARITY=>'DEFAULT', -
    CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS (OWNNAME=>'TPCC', -
    TABNAME=>'NORD', -
    PARTNAME=>NULL, -
    ESTIMATE_PERCENT=>1, -
    BLOCK_SAMPLE=>TRUE, -
    METHOD_OPT=>'FOR ALL COLUMNS SIZE 1', -
    DEGREE=>160, -
    GRANULARITY=>'DEFAULT', -
    CASCADE=>TRUE);

GRANULARITY=>'DEFAULT', -
CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS(OWNNAME=>'TPCC', -
    TABNAME=>'HIST', -
    PARTNAME=>NULL, -
    ESTIMATE_PERCENT=>1, -
    BLOCK_SAMPLE=>TRUE, -
    METHOD_OPT=>'FOR ALL COLUMNS SIZE 1', -
    DEGREE=>160, -
    GRANULARITY=>'DEFAULT', -
    CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS(OWNNAME=>'TPCC', -
    TABNAME=>'DIST', -
    PARTNAME=>NULL, -
    ESTIMATE_PERCENT=>1, -
    BLOCK_SAMPLE=>TRUE, -
    METHOD_OPT=>'FOR ALL COLUMNS SIZE 1', -
    DEGREE=>10, -
    GRANULARITY=>'DEFAULT', -
    CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS(OWNNAME=>'TPCC', -
    TABNAME=>'ITEM', -
    PARTNAME=>NULL, -
    ESTIMATE_PERCENT=>10, -
    BLOCK_SAMPLE=>TRUE, -
    METHOD_OPT=>'FOR ALL COLUMNS SIZE 1', -
    DEGREE=>1, -
    GRANULARITY=>'DEFAULT', -
    CASCADE=>TRUE);

execute dbms_stats.GATHER_TABLE_STATS(OWNNAME=>'TPCC', -
    TABNAME=>'WARE', -
    PARTNAME=>NULL, -
    ESTIMATE_PERCENT=>10, -
    BLOCK_SAMPLE=>TRUE, -
    METHOD_OPT=>'FOR ALL COLUMNS SIZE 1', -
    DEGREE=>10, -
    GRANULARITY=>'DEFAULT', -
    CASCADE=>TRUE);

set echo off;
spool off;
exit sql.sqlcode;

createdb.sql
/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatedb.sh Wed Jan 18
09:52:42 PST 2012 */
spool createdb.log
set echo on

shutdown abort

startup pfile=p_create.ora nomount
create database tpcc
controlfile reuse
maxinstances 1
datafile
'/export/home/oracle/tpcc_disks/system_1' size 2000M reuse
logfile
group 1 ('/export/home/oracle/tpcc_disks/log_0_0',
'/export/home/oracle/tpcc_disks/log_3_0') size 735000M reuse,
group 2 ('/export/home/oracle/tpcc_disks/log_0_3',
'/export/home/oracle/tpcc_disks/log_3_3') size 735000M reuse
sysaux datafile '/export/home/oracle/tpcc_disks/tpccaux' size 1200M reuse ;

create undo tablespace undo_1 datafile
'/export/home/oracle/tpcc_disks/roll1' size 8096M reuse blocksize 8K;

set echo off
exit sql.sqlcode

createindex_icust1.sql
/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:52:58 PST 2012 */
set timing on
set sqlblanklines on
spool createindex_icust1.log ;
set echo on ;
drop index icust1 ;
create unique index icust1 on cust ( c_w_id
, c_d_id
, c_id )
pctfree 1 intrans 3
storage ( buffer_pool default )
parallel 320
compute statistics
tablespace icust1_0 ;
set echo off
spool off
exit sql.sqlcode;

createindex_icust2.sql
/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:52:58 PST 2012 */
set timing on
set sqlblanklines on
spool createindex_icust2.log ;
set echo on ;
drop index icust2 ;
create unique index icust2 on cust ( c_last
, c_w_id
, c_d_id
```

```

,c_first
,c_id )
pctfree 1 initrans 3
storage ( buffer_pool default )
parallel 320
compute statistics
tablespace icust2_0 ;
set echo off
spool off
exit sql.sqlcode;

createindex_idist.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:52:58 PST 2012 */
set timing on
set sqlblanklines on
spool createindex_idist.log ;
set echo on ;
drop index idist ;
create unique index idist on dist ( d_w_id
,d_id )
pctfree 5 initrans 3
storage ( buffer_pool default )
parallel 1
compute statistics
tablespace misc_0 ;
set echo off
spool off
exit sql.sqlcode;

createindex_iitem.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:52:59 PST 2012 */
set timing on
set sqlblanklines on
spool createindex_iitem.log ;
set echo on ;
drop index iitem ;
create unique index iitem on item ( i_id )
pctfree 5 initrans 4
storage ( buffer_pool default )

compute statistics
tablespace misc_0 ;
set echo off
spool off
exit sql.sqlcode;

createindex_inord.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:53:00 PST 2012 */
set timing on
exit 0;

createindex_iordl.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:53:00 PST 2012 */
set timing on
exit 0;

createindex_iordr1.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:52:59 PST 2012 */
set timing on
exit 0;

createindex_iordr2.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:52:59 PST 2012 */
set timing on
set sqlblanklines on
spool createindex_iordr2.log ;
set echo on ;
drop index iordr2 ;
create unique index iordr2 on ordr ( o_w_id
,o_d_id
,o_c_id
,o_id )
global partition by range (o_w_id) (
partition iordr2_0 values less than ( 25001 ) tablespace iordr2_0
,partition iordr2_1 values less than ( 50001 ) tablespace iordr2_0
,partition iordr2_2 values less than ( 75001 ) tablespace iordr2_0
,partition iordr2_3 values less than ( 100001 ) tablespace iordr2_0
,partition iordr2_4 values less than ( 125001 ) tablespace iordr2_0
,partition iordr2_5 values less than ( 150001 ) tablespace iordr2_0
,partition iordr2_6 values less than ( 175001 ) tablespace iordr2_0
,partition iordr2_7 values less than ( 200001 ) tablespace iordr2_0
,partition iordr2_8 values less than ( 225001 ) tablespace iordr2_0
,partition iordr2_9 values less than ( 250001 ) tablespace iordr2_0
,partition iordr2_10 values less than ( 275001 ) tablespace iordr2_0
,partition iordr2_11 values less than ( 300001 ) tablespace iordr2_0
,partition iordr2_12 values less than ( 325001 ) tablespace iordr2_0
,partition iordr2_13 values less than ( 350001 ) tablespace iordr2_0
,partition iordr2_14 values less than ( 375001 ) tablespace iordr2_0
,partition iordr2_15 values less than ( maxvalue ) tablespace iordr2_0
)
parallel 320
pctfree 25 initrans 4

storage ( buffer_pool default )
compute statistics
tablespace iordr2_0 ;
set echo off
spool off
exit sql.sqlcode;

createindex_istok.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:52:59 PST 2012 */
set timing on
set sqlblanklines on
spool createindex_istok.log ;
set echo on ;
drop index istok ;
create unique index istok on stok ( s_i_id
,s_w_id )
pctfree 1 initrans 3
storage ( buffer_pool default )
parallel 320
compute statistics
tablespace istok_0 ;
set echo off
spool off
exit sql.sqlcode;

createindex_iware.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreateindex.sh Wed Jan 18
09:52:57 PST 2012 */
set timing on
set sqlblanklines on
spool createindex_iware.log ;
set echo on ;
drop index iware ;
create unique index iware on ware ( w_id )
pctfree 1 initrans 3
storage ( buffer_pool default )
parallel 1
compute statistics
tablespace misc_0 ;
set echo off
spool off
exit sql.sqlcode;

createspacestats.sql

@space_init
@space_get 5040000.0 400000
@space_rpt
spool off
exit sql.sqlcode;

```

**createtabledistsql**

```

spool createtabledistsql.log
@tkvcin.sql
spool off
exit sql.sqlcode;

```

**createtabledistsql**

```

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatetabledist.sh Wed Jan 18
09:52:42 PST 2012 */
set timing on
set sqlblanklines on
spool createtabledist.log
set echo on
drop cluster custcluster including tables ;

```

```

create cluster custcluster (
  c_id number
, c_d_id number
, c_w_id number
)
single table
hashkeys 12000000000
hash is (( c_id * ( 400000 * 10 ) + c_w_id * 10 + c_d_id )
size 630
pctfree 0 initrans 3
storage ( buffer_pool recycle ) parallel ( degree 160 )
tablespace cust_0;

```

```

create table cust (
  c_id number
, c_d_id number
, c_w_id number
, c_discount number
, c_credit char(2)
, c_last varchar2(16)
, c_first varchar2(16)
, c_credit_lim number
, c_balance number
, c_ytd_payment number
, c_payment_cnt number
, c_delivery_cnt number
, c_street_1 varchar2(20)
, c_street_2 varchar2(20)
, c_city varchar2(20)
, c_state char(2)
, c_zip char(9)
, c_phone char(16)
, c_since date
, c_middle char(2)
, c_data varchar2(500)
)
cluster custcluster (

```

```

  c_id
, c_d_id
, c_w_id
);
set echo off
spool off
exit sql.sqlcode;

```

**createtabledistsql**

```

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatetabledist.sh Wed Jan 18
09:52:43 PST 2012 */
set timing on
set sqlblanklines on
spool createtabledist.log
set echo on
drop cluster distcluster including tables ;

```

```

create cluster distcluster (
  d_id number
, d_w_id number
)
single table
hashkeys 4000000
hash is (((d_w_id * 10) + d_id)
size 3496
initrans 4
storage ( buffer_pool default )
tablespace misc_0;

```

```

create table dist (
  d_id number
, d_w_id number
, d_ytd number
, d_next_o_id number
, d_tax number
, d_name varchar2(10)
, d_street_1 varchar2(20)
, d_street_2 varchar2(20)
, d_city varchar2(20)
, d_state char(2)
, d_zip char(9)
)
cluster distcluster (
  d_id
, d_w_id
);
set echo off
spool off
exit sql.sqlcode;

```

**createtabledistsql**

```

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatetabledist.sh Wed Jan 18

```

```

09:52:44 PST 2012 */
set timing on
set sqlblanklines on
spool createtabledist.log
set echo on
drop table hist ;

```

```

create table hist (
  h_c_id number
, h_d_id number
, h_c_w_id number
, h_d_id number
, h_w_id number
, h_date date
, h_amount number
, h_data varchar2(24)
)
partition by range( h_w_id ) (
  partition hist_0 values less than ( 50001 ) tablespace hist_0
, partition hist_1 values less than ( 100001 ) tablespace hist_0
, partition hist_2 values less than ( 150001 ) tablespace hist_0
, partition hist_3 values less than ( 200001 ) tablespace hist_0
, partition hist_4 values less than ( 250001 ) tablespace hist_0
, partition hist_5 values less than ( 300001 ) tablespace hist_0
, partition hist_6 values less than ( 350001 ) tablespace hist_0
, partition hist_7 values less than ( maxvalue ) tablespace hist_0
)
pctfree 5 initrans 4
storage ( buffer_pool recycle )
;
set echo off
spool off
exit sql.sqlcode;

```

**createtabledistsql**

```

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatetabledist.sh Wed Jan 18
09:52:45 PST 2012 */
set timing on
set sqlblanklines on
spool createtabledist.log
set echo on
drop cluster itemcluster including tables ;

```

```

create cluster itemcluster (
  i_id number(6,0)
)
single table
hashkeys 100000
hash is ( i_id )
size 120
pctfree 0 initrans 3
storage ( buffer_pool keep )
tablespace misc_0;

```

```

create table item (

```

```

i_id number(6,0)
,i_name varchar2(24)
,i_price number
,i_data varchar2(50)
,i_im_id number
)
cluster itemcluster (
i_id
);
set echo off
spool off
exit sql.sqlcode;

createtable_nord.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatetable.sh Wed Jan 18
09:52:47 PST 2012 */
set timing on
set sqlblanklines on
spool createtable_nord.log
set echo on
drop cluster nordcluster_queue including tables ;

create cluster nordcluster_queue (
no_w_id number
,no_d_id number
,no_o_id number SORT
)

hashkeys 4000000
hash is ( (no_w_id - 1) * 10 + no_d_id - 1 )
size 190
tablespace nord_0;

create table nord (
no_w_id number
,no_d_id number
,no_o_id number sort
,constraint nord_uk primary key ( no_w_id
,no_d_id
,no_o_id )
)
cluster nordcluster_queue (
no_w_id
,no_d_id
,no_o_id
);
set echo off
spool off
exit sql.sqlcode;

createtable_ordl.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatetable.sh Wed Jan 18
09:52:46 PST 2012 */
set timing on
set sqlblanklines on
spool createtable_ordl.log
set echo on
create table ordl (
ol_w_id number
,ol_d_id number
,ol_o_id number sort
,ol_number number sort
,ol_i_id number
,ol_delivery_d date
,ol_amount number
,ol_supply_w_id number
,ol_quantity number
,ol_dist_info char(24)
,constraint ordl_uk primary key (ol_w_id,ol_d_id,ol_o_id,ol_number )
)
CLUSTER ordcluster_queue(ol_w_id,ol_d_id,ol_o_id,ol_number) ;
set echo off
spool off
exit sql.sqlcode;

createtable_ordr.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatetable.sh Wed Jan 18
09:52:46 PST 2012 */
set timing on
set sqlblanklines on
spool createtable_ordr.log
set echo on
drop cluster ordcluster_queue including tables ;

create cluster ordcluster_queue (
o_w_id number
,o_d_id number
,o_id number SORT
,o_number number SORT
)

hashkeys 4000000
hash is ( (o_w_id - 1) * 10 + o_d_id - 1 )
size 1490
tablespace ordr_0;

create table ordr (
o_id number sort
,o_w_id number
,o_d_id number
,o_c_id number
,o_carrier_id number
,o_ol_cnt number
,o_all_local number
,o_entry_d date
,constraint ordr_uk primary key ( o_w_id
,o_d_id
,o_id )
)
cluster ordcluster_queue (
o_w_id
,o_d_id
,o_id
);
set echo off
spool off
exit sql.sqlcode;

createtable_stok.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatetable.sh Wed Jan 18
09:52:45 PST 2012 */
set timing on
set sqlblanklines on
spool createtable_stok.log
set echo on
drop cluster stokcluster including tables ;

create cluster stokcluster (
s_i_id number
,s_w_id number
)
single table
hashkeys 40000000000
hash is ( (s_i_id * 400000 + s_w_id) )
size 270
pctfree 0 initrans 2 maxtrans 2
storage ( buffer_pool keep ) parallel ( degree 160 )
tablespace stok_0;

create table stok (
s_i_id number
,s_w_id number
,s_quantity number
,s_ytd number
,s_order_cnt number
,s_remote_cnt number
,s_data varchar2(50)
,s_dist_01 char(24)
,s_dist_02 char(24)
,s_dist_03 char(24)
,s_dist_04 char(24)
,s_dist_05 char(24)
,s_dist_06 char(24)
,s_dist_07 char(24)
,s_dist_08 char(24)
,s_dist_09 char(24)
,s_dist_10 char(24)
)
cluster stokcluster (
s_i_id
,s_w_id
);
set echo off

```

```

spool off
exit sql.sqlcode;

if expr $? != 0 > /dev/null; then
echo Creating tablespace for cust failed. Exiting.
exit 0
fi

createtable_ware.sql

/* created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatetable.sh Wed Jan 18
09:52:42 PST 2012 */
set timing on
set sqlblanklines on
spool createtable_ware.log
set echo on
drop cluster warecluster including tables ;

create cluster warecluster (
w_id number
)
single table
hashkeys 400000
hash is ( (w_id - 1) )
size 3496
initrans 2
storage ( buffer_pool default )
tablespace misc_0;

create table ware (
w_id number
,w_ytd number
,w_tax number
,w_name varchar2(10)
,w_street_1 varchar2(20)
,w_street_2 varchar2(20)
,w_city varchar2(20)
,w_state char(2)
,w_zip char(9)
)
cluster warecluster (
w_id
);
set echo off
spool off
exit sql.sqlcode;

createts.sh

#created automatically by
/export/home/oracle/tpcckit.linux/scripts/buildcreatets.sh Wed Jan 18
09:52:37 PST 2012

$tpcc_createts misc 40 1 500M 10240K unix 0 0 160 auto t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for ware failed. Exiting.
exit 0
fi

$tpcc_createts cust 560 1 15000M 2558294K unix 0 40 160 auto t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for cust failed. Exiting.
exit 0
fi

if expr $? != 0 > /dev/null; then
echo Creating tablespace for hist failed. Exiting.
exit 0
fi

$tpcc_createts hist 80 1 11000M 90923K unix 0 600 160 auto t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for hist failed. Exiting.
exit 0
fi

$tpcc_createts stok 760 1 15700M 3215155K unix 0 680 160 auto t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for stok failed. Exiting.
exit 0
fi

$tpcc_createts nord 40 1 3600M 105296K unix 0 1640 160 auto t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for nord failed. Exiting.
exit 0
fi

$tpcc_createts icust1 40 1 7400M 51198K unix 0 1680 160 16K t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for icust1 failed. Exiting.
exit 0
fi

$tpcc_createts icust2 40 1 15500M 51198K unix 0 1720 160 16K t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for icust2 failed. Exiting.
exit 0
fi

$tpcc_createts istok 40 1 21000M 51198K unix 0 1760 160 16K t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for istok failed. Exiting.
exit 0
fi

$tpcc_createts iordr2 80 1 7100M 102399K unix 0 1800 160 16K t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for iordr2 failed. Exiting.
exit 0
fi

$tpcc_createts temp 160 1 14000M 204800K unix 1 1880 160 auto t
if expr $? != 0 > /dev/null; then
echo Creating tablespace for temp failed. Exiting.
exit 0
fi

junk

SQL*Plus: Release 11.2.0.2.0 Production on Fri Feb 3 11:40:58 2012
Copyright (c) 1982, 2010, Oracle. All rights reserved.

Connected to:
Oracle Database 11g Enterprise Edition Release 11.2.0.2.0 - 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing
options

SQL> drop table tpcc_data;
Table dropped.

SQL> drop table tpcc_space;
Table dropped.

SQL> drop table tpcc_totSPACE;
Table dropped.

SQL> create table tpcc_data (
2 segment varchar2(18),
3 type varchar2(15),
4 blocks number,
5 block_size number,
6 five_pct number,
7 daily_grow number,
8 total number
9 );
Table created.

SQL> create table tpcc_space (
2 segment varchar2(18),
3 blocks number,
4 block_size number,
5 required number,
6 static number,
7 dynamic number,
8 oversize number
9 );
Table created.

SQL> create table tpcc_totSPACE (
2 tpm number,
3 nware number,
4 static number,
5 dynamic number,
6 oversize number,
7 daily_grow number,

```

8	daily_spre	number,	SQL>	insert into tpcc_data	8	) sp\$,	
9	space60	number	2	select 'SYS_AUX', 'SYS', sum(blocks), t.block_size, 0, 0,	9	(select distinct tablespace_name, segment_name name	
10	);		sum(blocks)	3	from dba_data_files f, dba_tablespaces t	10	from dba_extents
Table created.			4	where f.tablespace_name = 'SYS_AUX' and t.tablespace_name =	11	where owner = 'TPCC'	
SQL>	create unique index itpcc_data on tpcc_data (segment);		f.tablespace_name	5	group by t.block_size;	12	and (segment_type = 'CLUSTER' or segment_type = 'TABLE'
Index created.			5	group by t.block_size;	13	or segment_type = 'TABLE PARTITION' or segment_type =	
SQL>	create unique index itpcc_space on tpcc_space (segment);		1 row created.		14	or segment_type = 'INDEX PARTITION')	
Index created.			SQL>	insert into tpcc_data	15	and tablespace_name <> 'SYSTEM'	
SQL>	set echo off;		SQL>	2	select 'ROLL_SEG', 'SYS', sum(blocks), t.block_size, 0, 0,	16	) ex\$
SQL>	delete from tpcc_data;		sum(blocks)	3	from dba_data_files f, dba_tablespaces t	17	where sp\$.tablespace_name = ex\$.tablespace_name
0 rows deleted.			4	where f.tablespace_name like '%UNDO_TS%' and	18	group by ex\$.name, sp\$.block_size;	
SQL>	delete from tpcc_space;		f.tablespace_name = t.tablespace_name	5	group by f.tablespace_name, t.block_size;	17 rows created.	
0 rows deleted.			5	group by f.tablespace_name, t.block_size;			
SQL>	delete from tpcc_totospace;		0 rows created.			SQL>	
0 rows deleted.			SQL>	insert into tpcc_data		SQL>	
SQL>	delete from tpcc_totospace;		2	select 'DB_STAT', 'SYS', sum(blocks), t.block_size, 0, 0,		insert into tpcc_space	
0 rows deleted.			sum(blocks)	3	from dba_data_files f, dba_tablespaces t	2	select substr(f.tablespace_name,1,18), sum(blocks), t.block_size,
SQL>	insert into tpcc_data		4	where f.tablespace_name like '%SP_0%' and f.tablespace_name		3	0, 0, 0, 0
SQL>	2		= t.tablespace_name	5	group by f.tablespace_name, t.block_size;	4	from dba_data_files f, dba_tablespaces t
2	select substr(segment_name,1,18), substr(segment_type,1,15),		5	group by f.tablespace_name, t.block_size;		5	where (f.tablespace_name = 'SYSTEM' or f.tablespace_name =
3	sum(blocks), t.block_size,		1 row created.			6	'SYS_AUX')
4	round(sum(blocks) * 0.05), 0,		SQL>	update tpcc_data		2	and f.tablespace_name = t.tablespace_name
5	sum(blocks) + round(sum(blocks) * 0.05)		2	set five_pct = 0,		3	group by f.tablespace_name, t.block_size;
6	from dba_extents e, dba_tablespaces t		3	daily_grow = round(blocks * &&1 / 62.5 / &&2),		4	2 rows created.
7	where owner = 'TPCC' AND ( segment_type = 'INDEX' OR		4	total = blocks + round(blocks * &&1 / 62.5 / &&2)		5	
8	segment_type = 'INDEX PARTITION' OR segment_type =		5	where segment = 'HIST' OR segment =		6	
'CLUSTER'	OR segment_type = 'TABLE' OR segment_type = 'TABLE		'ORDRCLUSTER_QUEUE' OR	6	segment = 'IORDL';	SQL>	insert into tpcc_space
PARTITION')	AND e.tablespace_name <> 'SYSTEM' AND		6	segment = 'IORDL';		2	select 'DB_STAT', sum(blocks), t.block_size, 0, 0, 0, 0
e.tablespace_name <> 'SP_0'	AND e.tablespace_name = t.tablespace_name		old 3:	daily_grow = round(blocks * &&1 / 62.5 / &&2),		3	from dba_data_files f, dba_tablespaces t
11	AND e.tablespace_name = t.tablespace_name		new 3:	daily_grow = round(blocks * 5040000.0 / 62.5 / 400000),		4	where f.tablespace_name = 'SP_0' and f.tablespace_name =
12	group by segment_name, segment_type, t.block_size;		old 4:	total = blocks + round(blocks * &&1 / 62.5 / &&2)		5	t.tablespace_name
17 rows created.			new 4:	total = blocks + round(blocks * 5040000.0 / 62.5 / 400000)		5	group by f.tablespace_name, t.block_size;
SQL>			2 rows updated.			SQL>	
SQL>	insert into tpcc_data		SQL>	insert into tpcc_space		SQL>	update tpcc_space
2	select 'SYSTEM', 'SYS', sum(blocks), t.block_size, 0, 0,		2	select substr(ex\$.name,1,18), sum(sp\$.sz_blocks), sp\$.block_size,		2	set required =
sum(blocks)	3		0, 0, 0, 0	3	from	3	(
3	from dba_data_files f, dba_tablespaces t		4	(select f.tablespace_name , sum(blocks) sz_blocks, t.block_size		4	select sum(total)
4	where f.tablespace_name = 'SYSTEM' and t.tablespace_name =		block_size	5	from dba_data_files f, dba_tablespaces t	5	from tpcc_data
f.tablespace_name	5		5	from dba_data_files f, dba_tablespaces t		6	where tpcc_data.segment = tpcc_space.segment
5	group by t.block_size;		6	where f.tablespace_name <> 'SYSTEM' and f.tablespace_name		7	)
1 row created.			= t.tablespace_name	7	group by f.tablespace_name, t.block_size	8	where segment in
SQL>			7	group by f.tablespace_name, t.block_size		9	(
						10	select segment from tpcc_data

```

11 );
20 rows updated.
SQL>
SQL> update tpcc_space
2 set static =
3 (
4 select sum(total)
5 from tpcc_data
6 where tpcc_data.segment = tpcc_space.segment
7 )
8 where segment in
9 (
10 select segment from tpcc_data
11 );
20 rows updated.
SQL>
SQL> update tpcc_space
2 set static = 0,
3 dynamic =
4 (
5 select sum(blocks)
6 from tpcc_data
7 where tpcc_data.segment = tpcc_space.segment
8 )
9 where segment in ('HIST', 'ORDRCLUSTER_QUEUE',
'ORDL');
2 rows updated.
SQL>
SQL> update tpcc_space
2 set oversize = blocks - required;
20 rows updated.
SQL>
SQL> insert into tpcc_totSPACE
2 select &&1, &&2, sum(static * block_size)/1024, sum(dynamic
* block_size)/1024, sum(oversize * block_size)/1024, 0, 0, 0
3 from tpcc_space;
old 2: select &&1, &&2, sum(static * block_size)/1024, sum(dynamic *
block_size)/1024, sum(oversize * block_size)/1024, 0, 0, 0
new 2: select 5040000.0, 400000, sum(static * block_size)/1024,
sum(dynamic * block_size)/1024, sum(oversize * block_size)/1024, 0, 0, 0
1 row created.
SQL>
SQL> update tpcc_totSPACE
2 set daily_grow =
3 (
4 select sum(daily_grow * block_size)/1024
5 from tpcc_data

```

```

6 );
1 row updated.
SQL> update tpcc_totSPACE
2 set space60 = static + 60 * daily_grow;
1 row updated.
SQL> set echo off;
Disconnected from Oracle Database 11g Enterprise Edition Release 11.2.0.2.0
- 64bit Production
With the Partitioning, OLAP, Data Mining and Real Application Testing
options

```

**loadcust.sh**

```

#created automatically by
/export/home/oracle/tpcckit.linux/scripts/evenload.sh Wed Jan 18 09:52:53
PST 2012
rm -f loadcust*.log
cd $tpcc_bench
allprocs=
$tpcc_load -M 400000 -C -1 1 -m 9 >> loadcust0.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 10 -m 18 >> loadcust1.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 19 -m 27 >> loadcust2.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 28 -m 36 >> loadcust3.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 37 -m 45 >> loadcust4.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 46 -m 54 >> loadcust5.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 55 -m 63 >> loadcust6.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 64 -m 72 >> loadcust7.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 73 -m 81 >> loadcust8.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 82 -m 90 >> loadcust9.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 91 -m 99 >> loadcust10.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 100 -m 108 >> loadcust11.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 109 -m 117 >> loadcust12.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 118 -m 126 >> loadcust13.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 127 -m 135 >> loadcust14.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 136 -m 144 >> loadcust15.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 145 -m 153 >> loadcust16.log 2>&1 &
allprocs="$allprocs $!}"

```

```

$tpcc_load -M 400000 -C -1 154 -m 162 >> loadcust17.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 163 -m 171 >> loadcust18.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 172 -m 180 >> loadcust19.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 181 -m 189 >> loadcust20.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 190 -m 198 >> loadcust21.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 199 -m 207 >> loadcust22.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 208 -m 216 >> loadcust23.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 217 -m 225 >> loadcust24.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 226 -m 234 >> loadcust25.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 235 -m 243 >> loadcust26.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 244 -m 252 >> loadcust27.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 253 -m 261 >> loadcust28.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 262 -m 270 >> loadcust29.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 271 -m 279 >> loadcust30.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 280 -m 288 >> loadcust31.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 289 -m 297 >> loadcust32.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 298 -m 306 >> loadcust33.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 307 -m 315 >> loadcust34.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 316 -m 324 >> loadcust35.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 325 -m 333 >> loadcust36.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 334 -m 342 >> loadcust37.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 343 -m 351 >> loadcust38.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 352 -m 360 >> loadcust39.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 361 -m 369 >> loadcust40.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 370 -m 378 >> loadcust41.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 379 -m 387 >> loadcust42.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 388 -m 396 >> loadcust43.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 397 -m 405 >> loadcust44.log 2>&1 &
allprocs="$allprocs $!}"
$tpcc_load -M 400000 -C -1 406 -m 414 >> loadcust45.log 2>&1 &
allprocs="$allprocs $!}"

```









```

Stpcc_load -M 400000 -C -l 2871 -m 2880 >> loadcust307.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2881 -m 2890 >> loadcust308.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2891 -m 2900 >> loadcust309.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2901 -m 2910 >> loadcust310.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2911 -m 2920 >> loadcust311.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2921 -m 2930 >> loadcust312.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2931 -m 2940 >> loadcust313.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2941 -m 2950 >> loadcust314.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2951 -m 2960 >> loadcust315.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2961 -m 2970 >> loadcust316.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2971 -m 2980 >> loadcust317.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2981 -m 2990 >> loadcust318.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -C -l 2991 -m 3000 >> loadcust319.log 2>&1 &
allprocs="$Sallprocs ${!}"
error=0
for curproc in $Sallprocs; do
  wait $curproc
  error=`expr $? + $error`
done
exit `expr $error != 0`

loaddist.sh

cd $Stpcc_bench
Stpcc_load -M $Stpcc_scale -d > loaddist.log 2>&1

loadhist.sh

#created automatically by
/export/home/oracle/tpeckit.linux/scripts/evenload.sh Wed Jan 18 09:52:48
PST 2012
rm -f loadhist*.log
cd $Stpcc_bench
allprocs=
Stpcc_load -M 400000 -h -b 1 -e 3125 >> loadhist0.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 3126 -e 6250 >> loadhist1.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 6251 -e 9375 >> loadhist2.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 9376 -e 12500 >> loadhist3.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 12501 -e 15625 >> loadhist4.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 15626 -e 18750 >> loadhist5.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 18751 -e 21875 >> loadhist6.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 21876 -e 25000 >> loadhist7.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 25001 -e 28125 >> loadhist8.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 28126 -e 31250 >> loadhist9.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 31251 -e 34375 >> loadhist10.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 34376 -e 37500 >> loadhist11.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 37501 -e 40625 >> loadhist12.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 40626 -e 43750 >> loadhist13.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 43751 -e 46875 >> loadhist14.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 46876 -e 50000 >> loadhist15.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 50001 -e 53125 >> loadhist16.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 53126 -e 56250 >> loadhist17.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 56251 -e 59375 >> loadhist18.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 59376 -e 62500 >> loadhist19.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 62501 -e 65625 >> loadhist20.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 65626 -e 68750 >> loadhist21.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 68751 -e 71875 >> loadhist22.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 71876 -e 75000 >> loadhist23.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 75001 -e 78125 >> loadhist24.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 78126 -e 81250 >> loadhist25.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 81251 -e 84375 >> loadhist26.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 84376 -e 87500 >> loadhist27.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 87501 -e 90625 >> loadhist28.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 90626 -e 93750 >> loadhist29.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 93751 -e 96875 >> loadhist30.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 96876 -e 100000 >> loadhist31.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 100001 -e 103125 >> loadhist32.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 103126 -e 106250 >> loadhist33.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 106251 -e 109375 >> loadhist34.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 109376 -e 112500 >> loadhist35.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 112501 -e 115625 >> loadhist36.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 115626 -e 118750 >> loadhist37.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 118751 -e 121875 >> loadhist38.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 121876 -e 125000 >> loadhist39.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 125001 -e 128125 >> loadhist40.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 128126 -e 131250 >> loadhist41.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 131251 -e 134375 >> loadhist42.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 134376 -e 137500 >> loadhist43.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 137501 -e 140625 >> loadhist44.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 140626 -e 143750 >> loadhist45.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 143751 -e 146875 >> loadhist46.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 146876 -e 150000 >> loadhist47.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 150001 -e 153125 >> loadhist48.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 153126 -e 156250 >> loadhist49.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 156251 -e 159375 >> loadhist50.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 159376 -e 162500 >> loadhist51.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 162501 -e 165625 >> loadhist52.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 165626 -e 168750 >> loadhist53.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 168751 -e 171875 >> loadhist54.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 171876 -e 175000 >> loadhist55.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 175001 -e 178125 >> loadhist56.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 178126 -e 181250 >> loadhist57.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 181251 -e 184375 >> loadhist58.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 184376 -e 187500 >> loadhist59.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 187501 -e 190625 >> loadhist60.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 190626 -e 193750 >> loadhist61.log 2>&1 &
allprocs="$Sallprocs ${!}"
Stpcc_load -M 400000 -h -b 193751 -e 196875 >> loadhist62.log 2>&1 &
allprocs="$Sallprocs ${!}"

```

```

$tpcc_load -M 400000 -h -b 196876 -e 200000 >> loadhist63.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 200001 -e 203125 >> loadhist64.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 203126 -e 206250 >> loadhist65.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 206251 -e 209375 >> loadhist66.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 209376 -e 212500 >> loadhist67.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 212501 -e 215625 >> loadhist68.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 215626 -e 218750 >> loadhist69.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 218751 -e 221875 >> loadhist70.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 221876 -e 225000 >> loadhist71.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 225001 -e 228125 >> loadhist72.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 228126 -e 231250 >> loadhist73.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 231251 -e 234375 >> loadhist74.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 234376 -e 237500 >> loadhist75.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 237501 -e 240625 >> loadhist76.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 240626 -e 243750 >> loadhist77.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 243751 -e 246875 >> loadhist78.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 246876 -e 250000 >> loadhist79.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 250001 -e 253125 >> loadhist80.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 253126 -e 256250 >> loadhist81.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 256251 -e 259375 >> loadhist82.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 259376 -e 262500 >> loadhist83.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 262501 -e 265625 >> loadhist84.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 265626 -e 268750 >> loadhist85.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 268751 -e 271875 >> loadhist86.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 271876 -e 275000 >> loadhist87.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 275001 -e 278125 >> loadhist88.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 278126 -e 281250 >> loadhist89.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 281251 -e 284375 >> loadhist90.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 284376 -e 287500 >> loadhist91.log 2>&1 &
allprocs="$Sallprocs ${!}"

$tpcc_load -M 400000 -h -b 287501 -e 290625 >> loadhist92.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 290626 -e 293750 >> loadhist93.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 293751 -e 296875 >> loadhist94.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 296876 -e 300000 >> loadhist95.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 300001 -e 303125 >> loadhist96.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 303126 -e 306250 >> loadhist97.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 306251 -e 309375 >> loadhist98.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 309376 -e 312500 >> loadhist99.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 312501 -e 315625 >> loadhist100.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 315626 -e 318750 >> loadhist101.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 318751 -e 321875 >> loadhist102.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 321876 -e 325000 >> loadhist103.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 325001 -e 328125 >> loadhist104.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 328126 -e 331250 >> loadhist105.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 331251 -e 334375 >> loadhist106.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 334376 -e 337500 >> loadhist107.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 337501 -e 340625 >> loadhist108.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 340626 -e 343750 >> loadhist109.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 343751 -e 346875 >> loadhist110.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 346876 -e 350000 >> loadhist111.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 350001 -e 353125 >> loadhist112.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 353126 -e 356250 >> loadhist113.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 356251 -e 359375 >> loadhist114.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 359376 -e 362500 >> loadhist115.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 362501 -e 365625 >> loadhist116.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 365626 -e 368750 >> loadhist117.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 368751 -e 371875 >> loadhist118.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 371876 -e 375000 >> loadhist119.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 375001 -e 378125 >> loadhist120.log 2>&1 &
allprocs="$Sallprocs ${!}"

$tpcc_load -M 400000 -h -b 378126 -e 381250 >> loadhist121.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 381251 -e 384375 >> loadhist122.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 384376 -e 387500 >> loadhist123.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 387501 -e 390625 >> loadhist124.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 390626 -e 393750 >> loadhist125.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 393751 -e 396875 >> loadhist126.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -h -b 396876 -e 400000 >> loadhist127.log 2>&1 &
allprocs="$Sallprocs ${!}"
error=0
for curproc in $Sallprocs; do
    wait $curproc
    error=`expr $? + $error`
done
exit `expr $error != 0`

```

#### loaditem.sh

```

cd $tpcc_bench
$tpcc_load -M $tpcc_scale -i > loaditem.log 2>&1

```

#### loadnord.sh

```

#created automatically by
/export/home/oracle/tpcckit.linux/scripts/evenload.sh Wed Jan 18 09:52:48
PST 2012
rm -f loadnord*.log
cd $tpcc_bench
allprocs=
$tpcc_load -M 400000 -n -b 1 -e 1250 >> loadnord0.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 1251 -e 2500 >> loadnord1.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 2501 -e 3750 >> loadnord2.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 3751 -e 5000 >> loadnord3.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 5001 -e 6250 >> loadnord4.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 6251 -e 7500 >> loadnord5.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 7501 -e 8750 >> loadnord6.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 8751 -e 10000 >> loadnord7.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 10001 -e 11250 >> loadnord8.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 11251 -e 12500 >> loadnord9.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -n -b 12501 -e 13750 >> loadnord10.log 2>&1 &
allprocs="$Sallprocs ${!}"

```







```

Stpcc_load -M 400000 -n -b 340001 -e 341250 >> loadnord272.log 2>&1 & Stpcc_load -M 400000 -n -b 376251 -e 377500 >> loadnord301.log 2>&1 & 2500 >> loadordrordl1.log 2>&1 &
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 341251 -e 342500 >> loadnord273.log 2>&1 & Stpcc_load -M 400000 -n -b 377501 -e 378750 >> loadnord302.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy2.dat -b 2501 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 342501 -e 343750 >> loadnord274.log 2>&1 & Stpcc_load -M 400000 -n -b 378751 -e 380000 >> loadnord303.log 2>&1 & allprocs="$Sallprocs $!}"
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 343751 -e 345000 >> loadnord275.log 2>&1 & Stpcc_load -M 400000 -n -b 380001 -e 381250 >> loadnord304.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy3.dat -b 3751 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 345001 -e 346250 >> loadnord276.log 2>&1 & Stpcc_load -M 400000 -n -b 381251 -e 382500 >> loadnord305.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy4.dat -b 5001 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 346251 -e 347500 >> loadnord277.log 2>&1 & Stpcc_load -M 400000 -n -b 382501 -e 383750 >> loadnord306.log 2>&1 & allprocs="$Sallprocs $!}"
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 347501 -e 348750 >> loadnord278.log 2>&1 & Stpcc_load -M 400000 -n -b 383751 -e 385000 >> loadnord307.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy5.dat -b 6251 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 348751 -e 350000 >> loadnord279.log 2>&1 & Stpcc_load -M 400000 -n -b 385001 -e 386250 >> loadnord308.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy6.dat -b 7501 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 350001 -e 351250 >> loadnord280.log 2>&1 & Stpcc_load -M 400000 -n -b 386251 -e 387500 >> loadnord309.log 2>&1 & allprocs="$Sallprocs $!}"
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 351251 -e 352500 >> loadnord281.log 2>&1 & Stpcc_load -M 400000 -n -b 387501 -e 388750 >> loadnord310.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy7.dat -b 8751 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 352501 -e 353750 >> loadnord282.log 2>&1 & Stpcc_load -M 400000 -n -b 388751 -e 390000 >> loadnord311.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy8.dat -b 10001 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 353751 -e 355000 >> loadnord283.log 2>&1 & Stpcc_load -M 400000 -n -b 390001 -e 391250 >> loadnord312.log 2>&1 & allprocs="$Sallprocs $!}"
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 355001 -e 356250 >> loadnord284.log 2>&1 & Stpcc_load -M 400000 -n -b 391251 -e 392500 >> loadnord313.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy9.dat -b 11251 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 356251 -e 357500 >> loadnord285.log 2>&1 & Stpcc_load -M 400000 -n -b 392501 -e 393750 >> loadnord314.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy10.dat -b 12501 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 357501 -e 358750 >> loadnord286.log 2>&1 & Stpcc_load -M 400000 -n -b 393751 -e 395000 >> loadnord315.log 2>&1 & allprocs="$Sallprocs $!}"
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 358751 -e 360000 >> loadnord287.log 2>&1 & Stpcc_load -M 400000 -n -b 395001 -e 396250 >> loadnord316.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy11.dat -b 13751 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 360001 -e 361250 >> loadnord288.log 2>&1 & Stpcc_load -M 400000 -n -b 396251 -e 397500 >> loadnord317.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy12.dat -b 15001 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 361251 -e 362500 >> loadnord289.log 2>&1 & Stpcc_load -M 400000 -n -b 397501 -e 398750 >> loadnord318.log 2>&1 & allprocs="$Sallprocs $!}"
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 362501 -e 363750 >> loadnord290.log 2>&1 & Stpcc_load -M 400000 -n -b 398751 -e 400000 >> loadnord319.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy13.dat -b 16251 -e
allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}" allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 363751 -e 365000 >> loadnord291.log 2>&1 & error=0 Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy14.dat -b 17501 -e
allprocs="$Sallprocs $!}" for curproc in $allprocs; do wait $curproc
Stpcc_load -M 400000 -n -b 365001 -e 366250 >> loadnord292.log 2>&1 & error=`expr $? + $error`
allprocs="$Sallprocs $!}" done
Stpcc_load -M 400000 -n -b 366251 -e 367500 >> loadnord293.log 2>&1 & exit `expr $error != 0`
allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 367501 -e 368750 >> loadnord294.log 2>&1 &
allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 368751 -e 370000 >> loadnord295.log 2>&1 & loadordrordl.sh
allprocs="$Sallprocs $!}"
Stpcc_load -M 400000 -n -b 370001 -e 371250 >> loadnord296.log 2>&1 & #created automatically by
allprocs="$Sallprocs $!}" /export/home/oracle/tpcckit.linux/scripts/evenload.sh Wed Jan 18 09:52:51
Stpcc_load -M 400000 -n -b 371251 -e 372500 >> loadnord297.log 2>&1 & PST 2012
allprocs="$Sallprocs $!}" rm -f loadordrordl*.log
Stpcc_load -M 400000 -n -b 372501 -e 373750 >> loadnord298.log 2>&1 & cd $tpcc_bench
allprocs="$Sallprocs $!}" allprocs=
Stpcc_load -M 400000 -n -b 373751 -e 375000 >> loadnord299.log 2>&1 & Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy0.dat -b 1 -e 1250
allprocs="$Sallprocs $!}" >> loadordrordl0.log 2>&1 &
Stpcc_load -M 400000 -n -b 375001 -e 376250 >> loadnord300.log 2>&1 & allprocs="$Sallprocs $!}"
allprocs="$Sallprocs $!}" Stpcc_load -M 400000 -o ${tpcc_disks_location}dummy1.dat -b 1251 -e

```













```

$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy308.dat -b 385001 -e $tpcc_load -M 400000 -S -j 937 -k 1248 >> loadstok3.log 2>&1 &
386250 >> loadordrordl308.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 1249 -k 1560 >> loadstok4.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy309.dat -b 386251 -e allprocs="$Sallprocs ${!}"
387500 >> loadordrordl309.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy310.dat -b 387501 -e $tpcc_load -M 400000 -S -j 1561 -k 1872 >> loadstok5.log 2>&1 &
388750 >> loadordrordl310.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 1873 -k 2184 >> loadstok6.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy311.dat -b 388751 -e allprocs="$Sallprocs ${!}"
390000 >> loadordrordl311.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 2185 -k 2496 >> loadstok7.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy312.dat -b 390001 -e $tpcc_load -M 400000 -S -j 2497 -k 2808 >> loadstok8.log 2>&1 &
391250 >> loadordrordl312.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 2809 -k 3120 >> loadstok9.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy313.dat -b 391251 -e allprocs="$Sallprocs ${!}"
392500 >> loadordrordl313.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 3121 -k 3432 >> loadstok10.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy314.dat -b 392501 -e $tpcc_load -M 400000 -S -j 3433 -k 3744 >> loadstok11.log 2>&1 &
393750 >> loadordrordl314.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 3745 -k 4056 >> loadstok12.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy315.dat -b 393751 -e allprocs="$Sallprocs ${!}"
395000 >> loadordrordl315.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 4057 -k 4368 >> loadstok13.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy316.dat -b 395001 -e $tpcc_load -M 400000 -S -j 4369 -k 4680 >> loadstok14.log 2>&1 &
396250 >> loadordrordl316.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 4681 -k 4992 >> loadstok15.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy317.dat -b 396251 -e allprocs="$Sallprocs ${!}"
397500 >> loadordrordl317.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 5305 -k 5616 >> loadstok17.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy318.dat -b 397501 -e $tpcc_load -M 400000 -S -j 5617 -k 5928 >> loadstok18.log 2>&1 &
398750 >> loadordrordl318.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 5929 -k 6240 >> loadstok19.log 2>&1 &
$tpcc_load -M 400000 -o ${tpcc_disks_location}dummy319.dat -b 398751 -e allprocs="$Sallprocs ${!}"
400000 >> loadordrordl319.log 2>&1 &
allprocs="$Sallprocs ${!}"

for curproc in $Sallprocs; do
  wait $curproc
  error=`expr $? + $error`
done
exit `expr $error != 0`

loadstok.sh

#created automatically by
/export/home/oracle/tpcckit.linux/scripts/evenload.sh Wed Jan 18 09:52:55
PST 2012
rm -f loadstok*.log
cd $tpcc_bench
allprocs=
$tpcc_load -M 400000 -S -j 1 -k 312 >> loadstok0.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 313 -k 624 >> loadstok1.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 625 -k 936 >> loadstok2.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 937 -k 1248 >> loadstok3.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 1249 -k 1560 >> loadstok4.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 1561 -k 1872 >> loadstok5.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 1873 -k 2184 >> loadstok6.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 2185 -k 2496 >> loadstok7.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 2497 -k 2808 >> loadstok8.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 2809 -k 3120 >> loadstok9.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 3121 -k 3432 >> loadstok10.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 3433 -k 3744 >> loadstok11.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 3745 -k 4056 >> loadstok12.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 4057 -k 4368 >> loadstok13.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 4369 -k 4680 >> loadstok14.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 4681 -k 4992 >> loadstok15.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 4993 -k 5304 >> loadstok16.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 5305 -k 5616 >> loadstok17.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 5617 -k 5928 >> loadstok18.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 5929 -k 6240 >> loadstok19.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 6241 -k 6552 >> loadstok20.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 6553 -k 6864 >> loadstok21.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 6865 -k 7176 >> loadstok22.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 7177 -k 7488 >> loadstok23.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 7489 -k 7800 >> loadstok24.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 7801 -k 8112 >> loadstok25.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 8113 -k 8424 >> loadstok26.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 8425 -k 8736 >> loadstok27.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 8737 -k 9048 >> loadstok28.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 9049 -k 9360 >> loadstok29.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 9361 -k 9672 >> loadstok30.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 9673 -k 9984 >> loadstok31.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 9985 -k 10296 >> loadstok32.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 10297 -k 10608 >> loadstok33.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 10609 -k 10920 >> loadstok34.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 10921 -k 11232 >> loadstok35.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 11233 -k 11544 >> loadstok36.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 11545 -k 11856 >> loadstok37.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 11857 -k 12168 >> loadstok38.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 12169 -k 12480 >> loadstok39.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 12481 -k 12792 >> loadstok40.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 12793 -k 13104 >> loadstok41.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 13105 -k 13416 >> loadstok42.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 13417 -k 13728 >> loadstok43.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 13729 -k 14040 >> loadstok44.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 14041 -k 14352 >> loadstok45.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 14353 -k 14664 >> loadstok46.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 14665 -k 14976 >> loadstok47.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 14977 -k 15288 >> loadstok48.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 15289 -k 15600 >> loadstok49.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 15601 -k 15912 >> loadstok50.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 15913 -k 16224 >> loadstok51.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 16225 -k 16536 >> loadstok52.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 16537 -k 16848 >> loadstok53.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 16849 -k 17160 >> loadstok54.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 17161 -k 17472 >> loadstok55.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 17473 -k 17784 >> loadstok56.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 17785 -k 18096 >> loadstok57.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 18097 -k 18408 >> loadstok58.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 18409 -k 18720 >> loadstok59.log 2>&1 &
allprocs="$Sallprocs ${!}"
$tpcc_load -M 400000 -S -j 18721 -k 19032 >> loadstok60.log 2>&1 &
allprocs="$Sallprocs ${!}"

```









```
done
exit `expr $error != 0`
```

### loadware.sh

```
cd $tpcc_bench
$tpcc_load -M $tpcc_scale -w > loadware.log 2>&1
```

### p\_build2.ora

```
compatible = 11.2.0.1
control_files = (/export/home/oracle/tpcc_disks/control_001,
/export/home/oracle/tpcc_disks/control_002)
db_16k_cache_size = 41333M
db_8k_cache_size = 15500M
db_block_size = 4096
db_cache_size = 41333M
db_files = 2066
db_name = tpcc
log_buffer = 10485760
plsql_optimize_level=2
shared_pool_size = 7750M
statistics_level = basic
dml_locks = 500
transactions = 2000
undo_management = auto
undo_retention = 2
processes = 2000
recovery_parallelism = 40
sessions = 2000
parallel_max_servers = 100
```

### p\_build.ora

```
compatible = 11.2.0.1
control_files = (/export/home/oracle/tpcc_disks/control_001,
/export/home/oracle/tpcc_disks/control_002)
db_16k_cache_size = 41333M
db_8k_cache_size = 15500M
db_block_size = 4096
db_cache_size = 41333M
db_files = 2066
db_name = tpcc
log_buffer = 104857600
plsql_optimize_level=2
shared_pool_size = 77500M
statistics_level = basic
dml_locks = 500
transactions = 2000
undo_management = auto
undo_retention = 2
processes = 2000
recovery_parallelism = 40
sessions = 2000
parallel_max_servers = 400
```

```
UNDO_TABLESPACE = undo_1
db_block_checksum = TRUE
db_block_checking = TRUE
```

### p\_create.ora

```
compatible = 11.2.0.1
control_files = (/export/home/oracle/tpcc_disks/control_001,
/export/home/oracle/tpcc_disks/control_002)
db_16k_cache_size = 41333M
db_8k_cache_size = 15500M
db_block_size = 4096
db_cache_size = 41333M
db_files = 2066
db_name = tpcc
log_buffer = 1048576
plsql_optimize_level=2
shared_pool_size = 7750M
statistics_level = basic
undo_management = manual
db_block_checksum = TRUE
db_block_checking = TRUE
```

### tkvcinin.sql

```
Rem
Rem $Header: tk_perf/benchmark_kits/tpcc-new/scripts/sql/tkvcinin.sql
Rem /main/2 2008/12/15 05:58:45 avliet Exp $
Rem
Rem tkvcinin.sql
Rem
Rem Copyright (c) 2001, 2008, Oracle and/or its affiliates.
Rem All rights reserved.
Rem
Rem NAME
Rem tkvcinin.sql - <one-line expansion of the name>
Rem
Rem DESCRIPTION
Rem <short description of component this file declares/defines>
Rem
Rem NOTES
Rem <other useful comments, qualifications, etc.>
Rem
Rem MODIFIED (MM/DD/YY)
Rem heri 05/03/02 - Short table names.
Rem lwang 07/24/01 - remove SET
Rem lwang 05/22/01 - Merged lwang_createdb
Rem lwang 05/21/01 - Created
Rem
```

```
-- The initnew package for storing variables used in the
-- New Order anonymous block
```

```
CREATE OR REPLACE PACKAGE inittpcc
AS
```

```
TYPE intarray IS TABLE OF INTEGER index by binary_integer;
TYPE distarray IS TABLE OF VARCHAR(24) index by binary_integer;
TYPE rowidarray IS TABLE OF ROWID INDEX BY BINARY_INTEGER;
nulldate DATE;
s_dist distarray;
idxlarr intarray;
s_remote intarray;
dist intarray;
row_id rowidarray;
cust_rowid rowid;
dist_name varchar2(11);
ware_name varchar2(11);
c_num pls_integer;
PROCEDURE init_no(idxarr intarray);
PROCEDURE init_del;
PROCEDURE init_pay;
END inittpcc;
/
show errors;
```

```
CREATE OR REPLACE PACKAGE BODY inittpcc AS
PROCEDURE init_no(idxarr intarray)
IS
BEGIN
```

```
-- initialize null date
nulldate := TO_DATE('09-15-1811', 'MM-DD-YYYY');

-- we found a savings of ~500 instructions when
-- initializing the cr_date array on both the client
-- and the server, instead of initializing on
-- the client and passing it to the server. this cannot be done
-- as we require the current system date
-- cr_date := SYSDATE;
```

```
-- initialize idxlarr on the client and store it here
-- as a package variable
idxlarr := idxarr;
END init_no;
```

```
PROCEDURE init_del
IS
BEGIN
FOR i IN 1 .. 10 LOOP
dist(i) := i;
END LOOP;
END init_del;
```

```
PROCEDURE init_pay IS
BEGIN
NULL;
END init_pay;
```

```
END inittpcc;
/
show errors
```

```
CREATE OR REPLACE PACKAGE tpcc
AS
```

```

TYPE intarray IS TABLE OF INTEGER index by binary_integer;
TYPE distarray IS TABLE OF VARCHAR(24) index by binary_integer;
TYPE rowidarray IS TABLE OF ROWID INDEX BY BINARY_INTEGER; );
TYPE chararray IS TABLE OF VARCHAR(1) index by binary_integer;
TYPE numarray IS TABLE OF NUMBER index by binary_integer;
TYPE datarray IS TABLE OF DATE INDEX BY BINARY_INTEGER;
nulldate DATE;
s_dist distarray;
idxlarr intarray;
s_remote intarray;
dist intarray;
row_id rowidarray;
cust_rowid rowid;
dist_name varchar2(11);
ware_name varchar2(11);
c_num pls_integer;

PROCEDURE neworder (
par_w_id BINARY_INTEGER,
par_d_id BINARY_INTEGER,
par_c_id BINARY_INTEGER,
par_o_all_local BINARY_INTEGER,
par_o_ol_cnt IN OUT BINARY_INTEGER,
par_w_tax IN OUT BINARY_INTEGER,
par_d_tax IN OUT BINARY_INTEGER,
par_o_id IN OUT BINARY_INTEGER,
par_c_discount IN OUT BINARY_INTEGER,
par_c_credit IN OUT varchar2,
par_c_last IN OUT varchar2,
par_retry IN OUT BINARY_INTEGER,
par_cr_date DATE,
par_ol_i_id intarray,
par_ol_supply_w_id intarray,
par_i_price IN OUT numarray,
par_i_name IN OUT distarray,
par_s_quantity IN OUT intarray,
par_brand_generic IN OUT chararray,
par_ol_amount IN OUT intarray,
par_s_remote intarray,
par_ol_quantity intarray
);

PROCEDURE orderstatus (
ware_id INTEGER,
dist_id INTEGER,
cust_id IN OUT INTEGER,
bylastname INTEGER,
cust_last IN OUT VARCHAR2,
cust_first OUT VARCHAR2,
cust_middle OUT VARCHAR2,
cust_balance OUT NUMBER,
ord_id IN OUT INTEGER,
ord_entry_d OUT VARCHAR2,
ord_carrier_id OUT INTEGER,
ord_ol_cnt OUT INTEGER,
oline_supply_w_id IN OUT intarray,
oline_i_id IN OUT intarray,
oline_quantity IN OUT intarray,
oline_amount IN OUT numarray,
oline_delivery_d OUT datarray
);

PROCEDURE delivery (
ware_id IN INTEGER,
dist_id IN OUT intarray,
order_id OUT intarray,
ordcnt OUT INTEGER,
sums OUT intarray,
del_date IN DATE,
carrier_id IN INTEGER,
order_c_id OUT intarray,
retry IN OUT INTEGER
);

PROCEDURE payment (
ware_id INTEGER,
dist_id INTEGER,
cust_w_id INTEGER,
cust_d_id INTEGER,
cust_id IN OUT INTEGER,
bylastname INTEGER,
hist_amount NUMBER,
cust_last IN OUT VARCHAR2,
ware_street_1 OUT VARCHAR2,
ware_street_2 OUT VARCHAR2,
ware_city OUT VARCHAR2,
ware_state OUT VARCHAR2,
ware_zip OUT VARCHAR2,
dist_street_1 OUT VARCHAR2,
dist_street_2 OUT VARCHAR2,
dist_city OUT VARCHAR2,
dist_state OUT VARCHAR2,
dist_zip OUT VARCHAR2,
cust_first OUT VARCHAR2,
cust_middle OUT VARCHAR2,
cust_street_1 OUT VARCHAR2,
cust_street_2 OUT VARCHAR2,
cust_city OUT VARCHAR2,
cust_state OUT VARCHAR2,
cust_zip OUT VARCHAR2,
cust_phone OUT VARCHAR2,
cust_since OUT DATE,
cust_credit IN OUT VARCHAR2,
cust_credit_lim OUT NUMBER,
cust_discount OUT NUMBER,
cust_balance IN OUT NUMBER,
cust_data OUT VARCHAR2,
cr_date IN DATE,
retry IN OUT INTEGER
);

PROCEDURE stocklevel (
ware_id INTEGER,
dist_id INTEGER,
threshold INTEGER,
low_stock OUT INTEGER
);

);
END tpcc;
/
show errors;

CREATE OR REPLACE PACKAGE BODY tpcc AS
rows_lost BINARY_INTEGER;
max_index BINARY_INTEGER;
temp_index BINARY_INTEGER;
idx BINARY_INTEGER;
dummy_local BINARY_INTEGER;
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);

PROCEDURE neworder (
par_w_id BINARY_INTEGER,
par_d_id BINARY_INTEGER,
par_c_id BINARY_INTEGER,
par_o_all_local BINARY_INTEGER,
par_o_ol_cnt IN OUT BINARY_INTEGER,
par_w_tax IN OUT BINARY_INTEGER,
par_d_tax IN OUT BINARY_INTEGER,
par_o_id IN OUT BINARY_INTEGER,
par_c_discount IN OUT BINARY_INTEGER,
par_c_credit IN OUT varchar2,
par_c_last IN OUT varchar2,
par_retry IN OUT BINARY_INTEGER,
par_cr_date DATE,
par_ol_i_id intarray,
par_ol_supply_w_id intarray,
par_i_price IN OUT numarray,
par_i_name IN OUT distarray,
par_s_quantity IN OUT intarray,
par_brand_generic IN OUT chararray,
par_ol_amount IN OUT intarray,
par_s_remote intarray,
par_ol_quantity intarray
)
IS
BEGIN
LOOP BEGIN
UPDATE dist SET d_next_o_id = d_next_o_id + 1
WHERE d_id = par_d_id AND d_w_id = par_w_id
RETURNING d_tax, d_next_o_id-1
INTO par_d_tax, par_o_id;

SELECT c_discount, c_credit, c_last
INTO par_c_discount, par_c_credit, par_c_last
FROM cust
WHERE c_id = par_c_id AND c_d_id = par_d_id AND c_w_id =
par_w_id;

```



```

END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_06,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END)
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpcc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u6
WHEN 7 THEN
-- ++++++ u7
BEGIN

FORALL idx IN 1 .. par_o_ol_cnt
UPDATE /*+ VECTOR_READ */ stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_07,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END)
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpcc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u7
WHEN 8 THEN
-- ++++++ u8
BEGIN

FORALL idx IN 1 .. par_o_ol_cnt
UPDATE /*+ VECTOR_READ */ stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_08,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END)
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpcc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u8
WHEN 9 THEN
-- ++++++ u9
BEGIN

FORALL idx IN 1 .. par_o_ol_cnt
UPDATE /*+ VECTOR_READ */ stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_09,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END)
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpcc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u9
WHEN 10 THEN
-- ++++++ u10
BEGIN

FORALL idx IN 1 .. par_o_ol_cnt
UPDATE /*+ VECTOR_READ */ stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_10,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END)
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpcc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u10
----- u1 thru u10 --- END ---
ELSE
EXIT;
END CASE;

-- cache the no of rows processed
dummy_local := sql%rowcount;

-- fix the rows if necessary
IF (dummy_local != par_o_ol_cnt ) THEN
-- used to be PROCEDURE fix_items IS
BEGIN
-- gotta shift price, name, s_quantity, brand_generic, s_dist, ol_amount
idx := 1;
-- found 0 bad rows
rows_lost := 0;
-- so many rows in out array to begin with
max_index := sql%rowcount;

WHILE (max_index != par_o_ol_cnt) LOOP

-- find item where item ids dont match
WHILE (idx <= sql%rowcount AND
sql%bulk_rowcount(idx + rows_lost) = 1)
LOOP
idx := idx + 1;
END LOOP;

-- shift the items please
temp_index := max_index;
WHILE (temp_index >= idx + rows_lost) LOOP
par_i_price(temp_index + 1) := par_i_price(temp_index);
par_i_name(temp_index + 1) := par_i_name(temp_index);
par_s_quantity(temp_index + 1) :=
par_s_quantity(temp_index);
par_ol_amount(temp_index + 1) := par_ol_amount(temp_index);
inittpcc.s_dist(temp_index + 1) := inittpcc.s_dist(temp_index);
par_brand_generic(temp_index + 1) :=
par_brand_generic(temp_index);
temp_index := temp_index - 1;

```

```

END LOOP;
-- values for the non-existent items if not at end
IF (idx + rows_lost <= par_o_ol_cnt) THEN
    par_i_price(idx + rows_lost) := 0;
    par_i_name(idx + rows_lost) := 'NO ITEM';
    par_ol_amount(idx + rows_lost) := 0;
    par_s_quantity(idx + rows_lost) := 0;
    inittpc.s_dist(idx + rows_lost) := NULL;
    par_brand_generic(idx + rows_lost) := '';
-- one more bad row
rows_lost := rows_lost + 1;
max_index := max_index + 1;
END IF;

END LOOP;
END ;
-- end of procedure fix_items;

END IF;

FORALL idx IN 1..par_o_ol_cnt
-- doesnt hurt if we insert entries for invalid item too
INSERT INTO ordl
VALUES (par_w_id, par_d_id, par_o_id, inittpc.idx1arr(idx),
par_ol_i_id(idx),
inittpc.nulldate, par_ol_amount(idx), par_ol_supply_w_id(idx),
par_ol_quantity(idx), inittpc.s_dist(idx));

--If there are no errors, then just return without COMMITING
--The COMMIT is done on the driver side by OCI
-- If there are errors, then rollback and set o_ol_cnt to the processed value
-- note that this is an extra bind ### till we manage to get errors handled
-- properly
IF (dummy_local != par_o_ol_cnt) THEN
    par_o_ol_cnt := dummy_local;
    ROLLBACK;
END IF;

EXIT;

EXCEPTION
    WHEN not_serializable OR deadlock OR snapshot_too_old THEN
        ROLLBACK;
        par_retry := par_retry + 1;
    END;
END LOOP;
END neworder ;

PROCEDURE orderstatus (
    ware_id      INTEGER,
    dist_id      INTEGER,
    cust_id      IN OUT INTEGER,
    bylastname   INTEGER,
    cust_last    IN OUT VARCHAR2,
    cust_first   OUT VARCHAR2,
    cust_middle  OUT VARCHAR2,
    cust_balance OUT NUMBER,
    ord_id       IN OUT INTEGER,
    ord_entry_d  OUT VARCHAR2,
    ord_carrier_id OUT INTEGER,
    ord_ol_cnt   OUT INTEGER,
    oline_supply_w_id IN OUT intarray,
    oline_i_id   IN OUT intarray,
    oline_quantity IN OUT intarray,
    oline_amount IN OUT numarray,
    oline_delivery_d OUT datarray
)
IS
    cust_rowid      ROWID;
    ol               BINARY_INTEGER;
    c_num           BINARY_INTEGER;
    row_id          rowidarray;
    not_serializable EXCEPTION;
    PRAGMA EXCEPTION_INIT(not_serializable,-8177);
    deadlock        EXCEPTION;
    PRAGMA EXCEPTION_INIT(deadlock,-60);
    snapshot_too_old EXCEPTION;
    PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
    CURSOR o_cur IS
        SELECT ol_i_id, ol_supply_w_id, ol_quantity, ol_amount,
            nvl(ol_delivery_d,to_date('15-09-1911','DD-MM-YYYY'))
        FROM ordl
        WHERE ol_d_id = dist_id AND ol_w_id = ware_id AND ol_o_id =
    ord_id;
    CURSOR c_cur IS
        SELECT rowid
        FROM cust
        WHERE c_d_id = dist_id AND c_w_id = ware_id AND c_last =
    cust_last
        ORDER BY c_w_id, c_d_id, c_last, c_first;
    BEGIN
        LOOP BEGIN
            IF bylastname != 0 THEN
                c_num := 0;
                FOR c_id_rec IN c_cur LOOP
                    c_num := c_num + 1;
                    row_id(c_num) := c_id_rec.rowid;
                END LOOP;
                cust_rowid := row_id((c_num + 1) / 2);

                SELECT c_id, c_balance, c_first, c_middle, c_last
                INTO cust_id, cust_balance, cust_first, cust_middle, cust_last
                FROM cust
                WHERE rowid = cust_rowid;
            ELSE
                SELECT c_balance, c_first, c_middle, c_last
                INTO cust_balance, cust_first, cust_middle, cust_last
                FROM cust
                WHERE c_id = cust_id AND c_d_id = dist_id AND c_w_id =
    ware_id;
            END IF;
        -- AVLIET added the rownum=1 clause to select only one ORDER
        -- according to TPC-C-spec (2.6.2.2) largest order_id must be selected
        SELECT o_id,
            to_char(o_entry_d, 'DD-MM-YYYY.HH24:MI:SS'),
            nvl(o_carrier_id,0), o_ol_cnt
        INTO ord_id,
            ord_entry_d,
            ord_carrier_id, ord_ol_cnt
        FROM ordr
        WHERE o_d_id = dist_id AND o_w_id = ware_id AND o_c_id =
    cust_id
            AND rownum = 1
            ORDER BY o_w_id, o_d_id, o_c_id, o_id DESC;

        ol := 0;
        FOR o_cur_rec IN o_cur LOOP
            ol := ol + 1;
            oline_i_id(ol) := o_cur_rec.ol_i_id;
            oline_supply_w_id(ol) := o_cur_rec.ol_supply_w_id;
            oline_quantity(ol) := o_cur_rec.ol_quantity;
            oline_amount(ol) := o_cur_rec.ol_amount;
            oline_delivery_d(ol) := o_cur_rec.del_date;
        END LOOP;

        COMMIT;
        EXIT;

        EXCEPTION
            WHEN not_serializable OR deadlock OR snapshot_too_old THEN
                ROLLBACK;
            END;
        END LOOP;

    END orderstatus;

PROCEDURE delivery (
    ware_id      IN      INTEGER,
    dist_id      IN OUT  intarray,
    order_id     OUT    intarray,
    ordent       OUT    INTEGER,
    sums         OUT    intarray,
    del_date     IN      DATE,
    carrier_id   IN      INTEGER,
    order_c_id   OUT    intarray,
    retry        IN OUT  INTEGER
) IS
    BEGIN
        LOOP BEGIN
            FORALL d IN 1..10
                DELETE /* index_asc (nord inord) */ FROM nord N
                WHERE no_d_id = inittpc.dist(d)
                AND no_w_id = ware_id

```

```

AND no_o_id = (select min (no_o_id)
from nord
where no_d_id = N.no_d_id
and no_w_id = N.no_w_id)
RETURNING no_d_id, no_o_id BULK COLLECT INTO dist_id,
order_id;

ordcnt := SQL%ROWCOUNT;

FORALL o in 1.. ordcnt
UPDATE ordr SET o_carrier_id = carrier_id
WHERE o_id = order_id(o)
AND o_d_id = dist_id(o)
AND o_w_id = ware_id
RETURNING o_c_id BULK COLLECT INTO order_c_id;

FORALL o in 1.. ordcnt
UPDATE ordl SET ol_delivery_d = del_date
WHERE ol_w_id = ware_id
AND ol_d_id = dist_id(o)
AND ol_o_id = order_id(o)
RETURNING sum(ol_amount) BULK COLLECT INTO sums;

FORALL c IN 1.. ordcnt
UPDATE cust
SET c_balance = c_balance + sums(c),
c_delivery_cnt = c_delivery_cnt + 1
WHERE c_w_id = ware_id
AND c_d_id = dist_id(c)
AND c_id = order_c_id(c);

COMMIT;
EXIT;
EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old
THEN
ROLLBACK;
retry := retry + 1;
END;

END LOOP; -- for retry
END delivery;

PROCEDURE payment (
ware_id INTEGER,
dist_id INTEGER,
cust_w_id INTEGER,
cust_d_id INTEGER,
cust_id IN OUT INTEGER,
bylastname INTEGER,
hist_amount NUMBER,
cust_last IN OUT VARCHAR2,
ware_street_1 OUT VARCHAR2,
ware_street_2 OUT VARCHAR2,
ware_city OUT VARCHAR2,
ware_state OUT VARCHAR2,
ware_zip OUT VARCHAR2,
dist_street_1 OUT VARCHAR2,

```

```

dist_street_2 OUT VARCHAR2,
dist_city OUT VARCHAR2,
dist_state OUT VARCHAR2,
dist_zip OUT VARCHAR2,
cust_first OUT VARCHAR2,
cust_middle OUT VARCHAR2,
cust_street_1 OUT VARCHAR2,
cust_street_2 OUT VARCHAR2,
cust_city OUT VARCHAR2,
cust_state OUT VARCHAR2,
cust_zip OUT VARCHAR2,
cust_phone OUT VARCHAR2,
cust_since OUT DATE,
cust_credit IN OUT VARCHAR2,
cust_credit_lim OUT NUMBER,
cust_discount OUT NUMBER,
cust_balance IN OUT NUMBER,
cust_data OUT VARCHAR2,
cr_date IN DATE,
retry IN OUT INTEGER
)
IS
TYPE rowidarray IS TABLE OF ROWID INDEX BY
BINARY_INTEGER;
cust_rowid ROWID;
dist_name VARCHAR2(11);
ware_name VARCHAR2(11);
c_num BINARY_INTEGER;
row_id rowidarray;
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
CURSOR c_cur IS
SELECT rowid
FROM cust
WHERE c_d_id = cust_d_id AND c_w_id = cust_w_id AND c_last =
cust_last
ORDER BY c_w_id, c_d_id, c_last, c_first;
BEGIN
LOOP BEGIN

IF bylastname != 0 THEN
c_num := 0;
FOR c_id_rec IN c_cur LOOP
c_num := c_num + 1;
row_id(c_num) := c_id_rec.rowid;
END LOOP;
cust_rowid := row_id ((c_num + 1) / 2);

UPDATE cust
SET c_balance = c_balance - hist_amount,
c_ytd_payment = c_ytd_payment + hist_amount,
c_payment_cnt = c_payment_cnt + 1
WHERE rowid = cust_rowid

```

```

RETURNING c_id, c_first, c_middle, c_last, c_street_1, c_street_2,
c_city, c_state, c_zip, c_phone,
c_since, c_credit, c_credit_lim,
c_discount, c_balance
INTO cust_id, cust_first, cust_middle, cust_last, cust_street_1,
cust_street_2, cust_city, cust_state, cust_zip, cust_phone,
cust_since, cust_credit, cust_credit_lim, cust_discount,
cust_balance;

ELSE

UPDATE cust
SET c_balance = c_balance - hist_amount,
c_ytd_payment = c_ytd_payment + hist_amount,
c_payment_cnt = c_payment_cnt + 1
WHERE c_id = cust_id AND c_d_id = cust_d_id AND
c_w_id = cust_w_id
RETURNING rowid, c_first, c_middle, c_last, c_street_1, c_street_2,
c_city, c_state, c_zip, c_phone,
c_since, c_credit, c_credit_lim,
c_discount, c_balance
INTO cust_rowid, cust_first, cust_middle, cust_last,
cust_street_1, cust_street_2, cust_city, cust_state,
cust_zip, cust_phone, cust_since, cust_credit,
cust_credit_lim, cust_discount, cust_balance;
END IF;

IF cust_credit = 'BC' THEN

UPDATE cust
SET c_data = substr ((to_char (cust_id) || ' ' ||
to_char (cust_d_id) || ' ' ||
to_char (cust_w_id) || ' ' ||
to_char (dist_id) || ' ' ||
to_char (ware_id) || ' ' ||
to_char (hist_amount, '9999.99') || ' |')
|| c_data, 1, 500)
WHERE rowid = cust_rowid
RETURNING substr (c_data, 1, 200)
INTO cust_data;
ELSE
cust_data := '';
END IF;

UPDATE dist
SET d_ytd = d_ytd + hist_amount
WHERE d_id = dist_id
AND d_w_id = ware_id
RETURNING d_name, d_street_1, d_street_2, d_city, d_state, d_zip
INTO dist_name, dist_street_1, dist_street_2, dist_city,
dist_state, dist_zip;

UPDATE ware
SET w_ytd = w_ytd + hist_amount
WHERE w_id = ware_id
RETURNING w_name, w_street_1, w_street_2, w_city, w_state, w_zip
INTO ware_name, ware_street_1, ware_street_2, ware_city,

```

```

ware_state, ware_zip;
/
show errors

INSERT INTO hist
(h_c_id, h_c_d_id, h_c_w_id, h_d_id, h_w_id, h_date,
 h_amount, h_data)
VALUES
(cust_id, cust_d_id, cust_w_id, dist_id, ware_id, cr_date,
 hist_amount, ware_name || ' ' || dist_name);

COMMIT;
EXIT;

EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old THEN
ROLLBACK;
retry := retry + 1;
END;

END LOOP;
END payment;

PROCEDURE stocklevel (
ware_id INTEGER,
dist_id INTEGER,
threshold INTEGER,
low_stock OUT INTEGER
)
IS
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
BEGIN

LOOP BEGIN

SELECT count (DISTINCT s_i_id)
INTO low_stock
FROM ordl, stok, dist
WHERE d_id = dist_id AND d_w_id = ware_id AND
d_id = ol_d_id AND d_w_id = ol_w_id AND
ol_i_id = s_i_id AND ol_w_id = s_w_id AND
s_quantity < threshold AND
ol_o_id BETWEEN (d_next_o_id - 20) AND (d_next_o_id - 1);

COMMIT;
EXIT;

EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old THEN
ROLLBACK;
END;
END LOOP;
END stocklevel;

END tpcc;

/
tpccload.c
#ifdef RCSID
static char *RCSid =
"$Header: tk_perf/benchmark_kits/tpcc-
new/benchrun/source/server/tpccload.c/main/1 2008/12/15 05:58:52 avliet
Exp $ Copyr (c) 1993 Oracle";
#endif /* RCSID */

/
=====+
| Copyright (c) 1994 Oracle Corp, Redwood Shores, CA |
| OPEN SYSTEMS PERFORMANCE GROUP |
| All Rights Reserved |
+=====+
| FILENAME |
| tpccload.c |
| DESCRIPTION |
| Load or generate TPC-C database tables. |
| Usage: tpccload -M <# of wares> [options] |
| options: -A load all tables |
| -w load ware table |
| -d load dist table |
| -c load cust table (cluster around c_w_id) |
| -C load cust table (cluster around c_id) |
| -i load item table |
| -s load stok table (cluster around s_w_id) |
| -S load stok table (cluster around s_i_id) |
| -h load hist table |
| -n load new-order table |
| -o <oline file> load order and order-line table |
| -b <ware#> beginning ware number |
| -e <ware#> ending ware number |
| -j <item#> beginning item number (with -S) |
| -k <item#> ending item number (with -S) |
| -l <cid#> beginning cid number (with -C) |
| -m <cid#> ending cid number (with -C) |
| -g generate rows to standard output |
+=====+
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <time.h>
#include <sys/types.h>
#include "tpcc.h"

#ifdef ORA_NT
#undef boolean
#include <process.h>
#include "dpbcore.h"

#define gettime dpbtimef
#define getcpu dpbcpu
#define lrand48() ((long)rand() <<15 | rand())
#ifdef _STDC_
#define PROTO(args) args
#else
#define PROTO(args) ()
#endif
#endif

#define DISTARR 10 /* dist insert array size */
#define CUSTARR 100 /* cust insert array size */
#define STOCARR 100 /* stok insert array size */
#define ITEMARR 100 /* item insert array size */
#define HISTARR 100 /* hist insert array size */
#define ORDEARR 100 /* order insert array size */
#define NEWOARR 100 /* new order insert array size */

#define DISTFAC 10 /* max. dist id */
#define CUSTFAC 3000 /* max. cust id */
#define STOCFAC 100000 /* max. stok id */
#define ITEMFAC 100000 /* max. item id */
#define HISTFAC 30000 /* history / warehouse */
#define ORDEFAC 3000 /* order / district */
#define NEWOFAC 900 /* new order / district */

#define C 0 /* constant in non-uniform dist. eqt. */
#define CNUM1 1 /* first constant in non-uniform dist. eqt. */
#define CNUM2 2 /* second constant in non-uniform dist. eqt. */
#define CNUM3 3 /* third constant in non-uniform dist. eqt. */

#define SEED 2 /* seed for random functions */

#define NOT_SERIALIZABLE 8177 /* ORA-08177: transaction not
serializable */
#define SNAPSHOT_TOO_OLD 1555 /* ORA-01555: snapshot too old */
#define RECOVERR -10
#define IRRECERR -20

#define SQLTXTW "INSERT INTO ware (w_id, w_ytd, w_tax, w_name,
w_street_1, w_street_2, w_city, w_state, w_zip) VALUES (:w_id,
30000000, :w_tax, :w_name, :w_street_1, \
:w_street_2, :w_city, :w_state, :w_zip)"

#define SQLXTD "INSERT INTO dist (d_id, d_w_id, d_ytd, d_tax,
d_next_o_id, d_name, d_street_1, d_street_2, d_city, d_state, d_zip)
VALUES (:d_id, :d_w_id, 30000000, :d_tax, \
3001, :d_name, :d_street_1, :d_street_2, :d_city, :d_state, :d_zip)"

#define SQLXTCQUERY "select /*+ HASH ( cust )*/ count(*) from cust
where c_w_id = :s_c_w_id and c_d_id = :s_c_d_id and c_id = :s_c_id"

#define SQLTXTC "INSERT INTO cust (C_ID, C_D_ID, C_W_ID,
C_FIRST, C_MIDDLE, C_LAST, C_STREET_1, C_STREET_2, C_CITY,

```



```

C_STATE, C_ZIP, C_PHONE, C_SINCE, C_CREDIT, C_CREDIT_LIM,
C_DISCOUNT, C_BALANCE, C_YTD_PAYMENT, C_PAYMENT_CNT,
C_DELIVERY_CNT, C_DATA) VALUES (:c_id, :c_d_id, :c_w_id, \
:c_first, 'OE', :c_last, :c_street_1, :c_street_2, :c_city, :c_state, \
:c_zip, :c_phone, SYSDATE, :c_credit, 5000000, :c_discount, -1000, 1000,
1, \
0, :c_data)"
#define SQLTXTH "INSERT INTO hist (h_c_id, h_c_d_id, h_c_w_id,
h_d_id, h_w_id, h_date, h_amount, h_data) VALUES (:h_c_id, :h_c_d_id,
:h_c_w_id, \
:h_d_id, :h_w_id, SYSDATE, 1000, :h_data)"
#define SQLTXTSQUERY "select /*+ HASH ( stok )*/ count(*) from stok
where s_w_id = :s_s_w_id and s_i_id = :s_s_i_id"
#define SQLTXTS "INSERT INTO stok (s_i_id, s_w_id,
s_quantity, s_dist_01, s_dist_02, s_dist_03, s_dist_04, s_dist_05, s_dist_06,
s_dist_07, s_dist_08, s_dist_09, s_dist_10, s_ytd, s_order_cnt, s_remote_cnt,
s_data) \
VALUES (:s_i_id, :s_w_id, :s_quantity, \
:s_dist_01, :s_dist_02, :s_dist_03, :s_dist_04, :s_dist_05, :s_dist_06, \
:s_dist_07, :s_dist_08, :s_dist_09, :s_dist_10, 0, 0, 0, :s_data)" \
#define SQLTXTI "INSERT INTO item
(I_ID, I_IM_ID, I_NAME, I_PRICE, I_DATA) VALUES (:i_id, :i_im_id,
:i_name, :i_price, \
:i_data)"
#define SQLTXTO1 "INSERT INTO odr (O_ID,
O_D_ID, O_W_ID, O_C_ID, O_ENTRY_D, O_CARRIER_ID, O_OL_CNT, O_
ALL_LOCAL) \
VALUES (:o_id, :o_d_id, :o_w_id, :o_c_id, \
SYSDATE, :o_carrier_id, :o_ol_cnt, 1)"
#define SQLTXTO2 "INSERT INTO odr (O_ID,
O_D_ID, O_W_ID, O_C_ID, O_ENTRY_D, O_CARRIER_ID, O_OL_CNT, O_
ALL_LOCAL) \
VALUES (:o_id, :o_d_id, :o_w_id, :o_c_id, \
SYSDATE, 11, :o_ol_cnt, 1)"
#define SQLXTOL1 "INSERT INTO ordl (OL_O_ID, OL_D_ID,
OL_W_ID, OL_NUMBER, OL_DELIVERY_D, OL_I_ID,
OL_SUPPLY_W_ID, OL_QUANTITY, OL_AMOUNT, OL_DIST_INFO) \
VALUES (:ol_o_id, :ol_d_id, \
:ol_w_id, :ol_number, SYSDATE, :ol_i_id, :ol_supply_w_id, 5, 0, \
:ol_dist_info)"
#define SQLXTOL2 "INSERT INTO ordl (OL_O_ID, OL_D_ID,
OL_W_ID, OL_NUMBER, OL_DELIVERY_D, OL_I_ID,
OL_SUPPLY_W_ID, OL_QUANTITY, OL_AMOUNT, OL_DIST_INFO) \
VALUES (:ol_o_id, :ol_d_id, \
:ol_w_id, :ol_number, to_date('01-Jan-1811'), :ol_i_id, :ol_supply_w_id,
5, :ol_amount, \
:ol_dist_info)"
#define SQLXTXNO "INSERT INTO nord (no_o_id, no_d_id, no_w_id)
VALUES (:no_o_id, :no_d_id, :no_w_id)"
#define SQLXTXENHA "alter session set \'_enable_hash_overflow\'=true"
#define SQLXTXDIHA "alter session set \'_enable_hash_overflow\'=false"
static char *lastname[] = {
"BAR",
"OUGHT",
"ABLE",
"PRI",
"PRES",
"ESE",
"ANTI",
"CALLY",
"ATION",
"EING"
};
char num9[10];
char num16[17];
char str2[3];
char str24[15][25];
int randperm3000[3000];
void initperm();
void randstr();
void randdatastr();
void randnum();
void randlastname (char*, int);
int NURand();
void sysdate();
OCISvcCtx *tpcsvc;
OCIServer *tpcsrv;
OCIError *errhp;
OCIEnv *tpcenv;
OCISession *tpcsusr;
OCISmt *curw;
OCISmt *curd;
OCISmt *curc;
OCISmt *curcs;
OCISmt *curh;
OCISmt *curs;
OCISmt *cursr;
OCISmt *curi;
OCISmt *curo1;
OCISmt *curo2;
OCISmt *curo11;
OCISmt *curo12;
OCISmt *curno;
OCIBind *w_id_bp = (OCIBind *) 0;
OCIBind *w_name_bp = (OCIBind *) 0;
OCIBind *w_street1_bp = (OCIBind *) 0;
OCIBind *w_street2_bp = (OCIBind *) 0;
OCIBind *w_city_bp = (OCIBind *) 0;
OCIBind *w_state_bp = (OCIBind *) 0;
OCIBind *w_zip_bp = (OCIBind *) 0;
OCIBind *w_tax_bp = (OCIBind *) 0;
OCIBind *d_id_bp = (OCIBind *) 0;
OCIBind *d_w_id_bp = (OCIBind *) 0;
OCIBind *d_name_bp = (OCIBind *) 0;
OCIBind *d_street1_bp = (OCIBind *) 0;
OCIBind *d_street2_bp = (OCIBind *) 0;
OCIBind *d_city_bp = (OCIBind *) 0;
OCIBind *d_state_bp = (OCIBind *) 0;
OCIBind *d_zip_bp = (OCIBind *) 0;
OCIBind *d_tax_bp = (OCIBind *) 0;
OCIDefine *s_c_ret_bp = (OCIDefine *) 0;
OCIBind *s_c_id_bp = (OCIBind *) 0;
OCIBind *s_c_d_id_bp = (OCIBind *) 0;
OCIBind *s_c_w_id_bp = (OCIBind *) 0;
OCIBind *c_id_bp = (OCIBind *) 0;
OCIBind *c_d_id_bp = (OCIBind *) 0;
OCIBind *c_w_id_bp = (OCIBind *) 0;
OCIBind *c_first_bp = (OCIBind *) 0;
OCIBind *c_last_bp = (OCIBind *) 0;
OCIBind *c_street1_bp = (OCIBind *) 0;
OCIBind *c_street2_bp = (OCIBind *) 0;
OCIBind *c_city_bp = (OCIBind *) 0;
OCIBind *c_state_bp = (OCIBind *) 0;
OCIBind *c_zip_bp = (OCIBind *) 0;
OCIBind *c_phone_bp = (OCIBind *) 0;
OCIBind *c_discount_bp = (OCIBind *) 0;
OCIBind *c_credit_bp = (OCIBind *) 0;
OCIBind *c_data_bp = (OCIBind *) 0;
OCIBind *i_id_bp = (OCIBind *) 0;
OCIBind *i_im_id_bp = (OCIBind *) 0;
OCIBind *i_name_bp = (OCIBind *) 0;
OCIBind *i_price_bp = (OCIBind *) 0;
OCIBind *i_data_bp = (OCIBind *) 0;
OCIDefine *s_s_ret_bp = (OCIDefine *) 0;
OCIBind *s_s_i_id_bp = (OCIBind *) 0;
OCIBind *s_s_w_id_bp = (OCIBind *) 0;
OCIBind *s_i_id_bp = (OCIBind *) 0;
OCIBind *s_w_id_bp = (OCIBind *) 0;
OCIBind *s_quantity_bp = (OCIBind *) 0;
OCIBind *s_dist_01_bp = (OCIBind *) 0;
OCIBind *s_dist_02_bp = (OCIBind *) 0;
OCIBind *s_dist_03_bp = (OCIBind *) 0;
OCIBind *s_dist_04_bp = (OCIBind *) 0;
OCIBind *s_dist_05_bp = (OCIBind *) 0;
OCIBind *s_dist_06_bp = (OCIBind *) 0;
OCIBind *s_dist_07_bp = (OCIBind *) 0;
OCIBind *s_dist_08_bp = (OCIBind *) 0;
OCIBind *s_dist_09_bp = (OCIBind *) 0;
OCIBind *s_dist_10_bp = (OCIBind *) 0;
OCIBind *s_data_bp = (OCIBind *) 0;
OCIBind *h_c_id_bp = (OCIBind *) 0;

```



```

int h_w_id[100];
int h_d_id[100];
int h_c_id[100];
char h_data[100][25];

int o_id[100];
int o_d_id[100];
int o_w_id[100];
int o_c_id[100];
int o_carrier_id[100];
int o_ol_cnt[100];

int ol_o_id[1500];
int ol_d_id[1500];
int ol_w_id[1500];
int ol_number[1500];
int ol_i_id[1500];
int ol_supply_w_id[1500];
int ol_amount[1500];
char ol_dist_info[1500][24];
int o_cnt;
int ol_cnt;

ub2 ol_o_id_len[1500];
ub2 ol_d_id_len[1500];
ub2 ol_w_id_len[1500];
ub2 ol_number_len[1500];
ub2 ol_i_id_len[1500];
ub2 ol_supply_w_id_len[1500];
ub2 ol_dist_info_len[1500];
ub2 ol_amount_len[1500];

ub4 ol_o_id_clen;
ub4 ol_d_id_clen;
ub4 ol_w_id_clen;
ub4 ol_number_clen;
ub4 ol_i_id_clen;
ub4 ol_supply_w_id_clen;
ub4 ol_dist_info_clen;
ub4 ol_amount_clen;

ub2 o_id_len[100];
ub2 o_d_id_len[100];
ub2 o_w_id_len[100];
ub2 o_c_id_len[100];
ub2 o_carrier_id_len[100];
ub2 o_ol_cnt_len[100];

ub4 o_id_clen;
ub4 o_d_id_clen;
ub4 o_w_id_clen;
ub4 o_c_id_clen;
ub4 o_carrier_id_clen;
ub4 o_ol_cnt_clen;

text stmbuf[16*1024];

int no_o_id[100];

```

```

int no_d_id[100];
int no_w_id[100];

char sdate[30];

#ifdef ORA_NT
clock_t begin_time, end_time;
clock_t begin_cpu, end_cpu;

char *arg_ptr, **end_args;
#else
double begin_time, end_time;
double begin_cpu, end_cpu;
double gettime(), getcpu();

extern int getopt();
extern char *optarg;
extern int optind, opterr;
int opt;
#endif

char *argstr="M:AwdcCisShno:b:e:j:k:l:m:g";
int do_A=0;
int do_w=0;
int do_d=0;
int do_i=0;
int do_c=0;
int do_C=0;
int do_s=0;
int do_S=0;
int do_h=0;
int do_o=0;
int do_n=0;
int gen=0;
int bware=1;
int eware=0;
int bitem=1;
int eitem=0;
int bcid=1;
int ecid=0;

FILE *olfp=NULL;
char olfname[100];
char *basename;
int status;
#ifdef ORA_NT
char fname[100];
FILE *logfile;
#endif /* ORA_NT */

/*-----+
| Parse command line -- look for scale factor.
|
+-----*/

if (argc == 1) {
myusage ();
}

```

```

#ifdef ORA_NT
end_args = argv + argc;
for (++argv; argv < end_args; )
{
arg_ptr = *argv++;

if (*arg_ptr != '-')
{
myusage ();
} else
{
switch (arg_ptr[1]) {
case '?': myusage ();
break;
case 'M': scale = atoi (*argv++);
break;
case 'A': do_A = 1;
break;
case 'w': do_w = 1;
break;
case 'd': do_d = 1;
break;
case 'c': do_c = 1;
break;
case 'C': do_C = 1;
break;
case 'i': do_i = 1;
break;
case 's': do_s = 1;
break;
case 'S': do_S = 1;
break;
case 'h': do_h = 1;
break;
case 'n': do_n = 1;
break;
case 'o': do_o = 1;
strcpy (olfname, *argv++);
break;
case 'b': bware = atoi (*argv++);
break;
case 'e': eware = atoi (*argv++);
break;
case 'j': bitem = atoi (*argv++);
break;
case 'k': eitem = atoi (*argv++);
break;
case 'l': bcid = atoi (*argv++);
break;
case 'm': ecid = atoi (*argv++);
break;
case 'g': gen = 1;
strcpy (fname, *argv++);
break;
case 'l': logfile=fopen(*argv++, "w");
break;
default: fprintf (stderr, "THIS SHOULD NEVER HAPPEN!!!\n");
}
}
}

```



```

OCIAttrSet((dvoid *)tpcusr, OCI_HTYPE_SESSION, (dvoid *)pwd,
(ub4)strlen(pwd),
OCI_ATTR_PASSWORD, errhp);
OCIERROR(errhp, OCISessionBegin(tpcsvc, errhp, tpcusr,
OCI_CRED_RDBMS, OCI_DEFAULT));

OCIAttrSet(tpcsvc, OCI_HTYPE_SVCCTX, tpcusr, 0,
OCI_ATTR_SESSION, errhp);

fprintf(stderr, "\nConnected to Oracle userid '%s/%s'\n", uid, pwd);

/* open cursors and parse statement */
if (do_A || do_w) {
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **)&curw),
OCI_HTYPE_STMT, 0, (dvoid**)0);
OCIERROR(errhp, OCIStmtPrepare(curw, errhp, (text *)SQLXTWTW,
strlen((char *)SQLXTWTW), (ub4) OCI_NTV_SYNTAX, (ub4)
OCI_DEFAULT));
}

if (do_A || do_d) {
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **)&curd),
OCI_HTYPE_STMT, 0, (dvoid**)0);
OCIERROR(errhp, OCIStmtPrepare(curd, errhp, (text *)SQLXTXD,
strlen((char *)SQLXTXD), (ub4) OCI_NTV_SYNTAX, (ub4)
OCI_DEFAULT));
}

if (do_A || do_c || do_C) {
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **)&curc),
OCI_HTYPE_STMT, 0, (dvoid**)0);
OCIERROR(errhp, OCIStmtPrepare(curc, errhp, (text *)SQLXTXC,
strlen((char *)SQLXTXC), (ub4) OCI_NTV_SYNTAX, (ub4)
OCI_DEFAULT));
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **)&curcs),
OCI_HTYPE_STMT, 0, (dvoid**)0);
OCIERROR(errhp, OCIStmtPrepare(curcs, errhp, (text
*)SQLXTXCQUERY,
strlen((char *)SQLXTXCQUERY), (ub4) OCI_NTV_SYNTAX,
(ub4) OCI_DEFAULT));
}

if (do_A || do_h) {
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **)&curh),
OCI_HTYPE_STMT, 0, (dvoid**)0);
OCIERROR(errhp, OCIStmtPrepare(curh, errhp, (text *)SQLXTXH,
strlen((char *)SQLXTXH), (ub4) OCI_NTV_SYNTAX, (ub4)
OCI_DEFAULT));
}

if (do_A || do_s || do_S) {
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **)&curcs),
OCI_HTYPE_STMT, 0, (dvoid**)0);
OCIERROR(errhp, OCIStmtPrepare(curcs, errhp, (text *)SQLXTS,
strlen((char *)SQLXTS), (ub4) OCI_NTV_SYNTAX, (ub4)
OCI_DEFAULT));
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **)&curss),
OCI_HTYPE_STMT, 0, (dvoid**)0);
OCIERROR(errhp, OCIStmtPrepare(curss, errhp, (text *)SQLXTSQUERY,
strlen((char *)SQLXTSQUERY), (ub4) OCI_NTV_SYNTAX,
(ub4) OCI_DEFAULT));
}

OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **)&curw1),
OCI_HTYPE_STMT, 0, (dvoid**)0);
DISCARD strcpy(fname, basename);
DISCARD strcat(fname, ".");
DISCARD strcat(fname, "benchrun/blocks/load_ordordl.sql");
stat = sqlfile(fname, stmbuf);
if (!stat)
{
fprintf(stderr, "unable to open %s\n", fname);
quit();
exit(1);
}
OCIERROR(errhp, OCIStmtPrepare(curo1, errhp, stmbuf,
strlen((char *)stmbuf), (ub4) OCI_NTV_SYNTAX, (ub4)
OCI_DEFAULT));
}

if (do_A || do_n) {
OCIERROR(errhp, OCIHandleAlloc(tpcenv, (dvoid **)&curno),
OCI_HTYPE_STMT, 0, (dvoid**)0);
OCIERROR(errhp, OCIStmtPrepare(curno, errhp, (text *)SQLXTNO,
strlen((char *)SQLXTNO), (ub4) OCI_NTV_SYNTAX, (ub4)
OCI_DEFAULT));
}

/* bind variables */

/* warehouse */

if (do_A || do_w) {
OCIERROR(errhp, OCIBindByName(curw, &w_id_bp, errhp, (text *)
(":w_id"), strlen(":w_id"),
(ub1 *)&w_id, sizeof(w_id), SQLT_INT, (dvoid *) 0, (ub2
*)0, (ub2 *)0,
(ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
OCIERROR(errhp, OCIBindByName(curw, &w_name_bp, errhp, (text
*)":w_name", strlen(":w_name"),
(ub1 *)w_name, 11, SQLT_STR, (dvoid *) 0, (ub2 *)0, (ub2
*)0,
(ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
}

OCIERROR(errhp, OCIBindByName(curw, &w_street1_bp, errhp,
OCIERROR(errhp, OCIBindByName(curw, &w_street2_bp, errhp,
OCIERROR(errhp, OCIBindByName(curw, &w_city_bp, errhp, (text
*)":w_city",
strlen(":w_city"), (ub1 *)w_city, 21, SQLT_STR,
(dvoid *) 0, (ub2 *)0, (ub2 *)0,
(ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
OCIERROR(errhp, OCIBindByName(curw, &w_state_bp, errhp, (text
*)":w_state",
strlen(":w_state"), (ub1 *)w_state, 2, SQLT_CHR,
(dvoid *) 0, (ub2 *)0, (ub2 *)0,
(ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
OCIERROR(errhp, OCIBindByName(curw, &w_zip_bp, errhp, (text
*)":w_zip",
strlen(":w_zip"), (ub1 *)w_zip, 9, SQLT_CHR,
(dvoid *) 0, (ub2 *)0, (ub2 *)0,
(ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
OCIERROR(errhp, OCIBindByName(curw, &w_tax_bp, errhp, (text
*)":w_tax",
strlen(":w_tax"), (ub1 *) &w_tax, sizeof(w_tax),
SQLT_FLT,
(dvoid *) 0, (ub2 *)0, (ub2 *)0,
(ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
}

/* district */

if (do_A || do_d) {
OCIERROR(errhp, OCIBindByName(curd, &d_id_bp, errhp, (text
*)":d_id",
strlen(":d_id"), (ub1 *)d_id, sizeof(int), SQLT_INT,
(dvoid *) 0, (ub2 *)0, (ub2 *)0,
(ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
OCIERROR(errhp, OCIBindByName(curd, &d_w_id_bp, errhp, (text
*)":d_w_id",
strlen(":d_w_id"), (ub1 *)d_w_id, sizeof(int),
SQLT_INT,
(dvoid *) 0, (ub2 *)0, (ub2 *)0,
(ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
OCIERROR(errhp, OCIBindByName(curd, &d_name_bp, errhp, (text
*)":d_name",
strlen(":d_name"), (ub1 *)d_name, sizeof(int), SQLT_STR,
(dvoid *) 0, (ub2 *)0, (ub2 *)0,
(ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
}

```

```

*)":d_name",
        strlen(":d_name"), (ub1 *)d_name, 11, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curd, &d_street1_bp, errhp, (text
*)":d_street_1",
        strlen(":d_street_1"), (ub1 *)d_street_1, 21,
        SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curd, &d_street2_bp, errhp, (text
*)":d_street_2",
        strlen(":d_street_2"), (ub1 *)d_street_2, 21,
        SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curd, &d_city_bp, errhp, (text
*)":d_city",
        strlen(":d_city"), (ub1 *)d_city, 21, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curd, &d_state_bp, errhp, (text
*)":d_state",
        strlen(":d_state"), (ub1 *)d_state, 2, SQLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curd, &d_zip_bp, errhp, (text
*)":d_zip",
        strlen(":d_zip"), (ub1 *)d_zip, 9, SQLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curd, &d_tax_bp, errhp, (text
*)":d_tax",
        strlen(":d_tax"), (ub1 *)d_tax, sizeof(float), SQLT_FLT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
    }
    /* customer */
    if (do_A || do_c || do_C) {
        OCIERROR(errhp, OCIBindByName(curcs, &s_c_id_bp, errhp, (text
*)":s_c_id",
        strlen(":s_c_id"), (ub1 *)&s_c_id, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curcs, &s_c_w_id_bp, errhp,
(text *)":s_c_w_id",
        strlen(":s_c_w_id"), (ub1 *)&s_c_w_id, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curcs, &s_c_d_id_bp, errhp, (text
*)":s_c_d_id",
        strlen(":s_c_d_id"), (ub1 *)&s_c_d_id, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curcs, &s_c_ret_bp, errhp, 1, &s_c_count, sizeof(int), SQ
        LT_INT,
        0, 0, 0, OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_id_bp, errhp, (text
*)":c_id",
        strlen(":c_id"), (ub1 *)c_id, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_d_id_bp, errhp, (text
*)":c_d_id",
        strlen(":c_d_id"), (ub1 *)c_d_id, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_w_id_bp, errhp, (text
*)":c_w_id",
        strlen(":c_w_id"), (ub1 *)c_w_id, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_first_bp, errhp, (text
*)":c_first",
        strlen(":c_first"), (ub1 *)c_first, 17, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_last_bp, errhp, (text
*)":c_last",
        strlen(":c_last"), (ub1 *)c_last, 17, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_street1_bp, errhp, (text
*)":c_street_1",
        strlen(":c_street_1"), (ub1 *)c_street_1, 21, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_street2_bp, errhp, (text
*)":c_street_2",
        strlen(":c_street_2"), (ub1 *)c_street_2, 21, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_city_bp, errhp, (text
*)":c_city",
        strlen(":c_city"), (ub1 *)c_city, 21, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_state_bp, errhp, (text
*)":c_state",
        strlen(":c_state"), (ub1 *)c_state, 2, SQLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_zip_bp, errhp, (text
*)":c_zip",
        strlen(":c_zip"), (ub1 *)c_zip, 9, SQLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_phone_bp, errhp, (text
*)":c_phone",
        strlen(":c_phone"), (ub1 *)c_phone, 16, SQLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_credit_bp, errhp, (text
*)":c_credit",
        strlen(":c_credit"), (ub1 *)c_credit, 2, SQLT_CHR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_discount_bp, errhp,
        (text *)":c_discount",
        strlen(":c_discount"), (ub1 *)c_discount, sizeof(float),
        SQLT_FLT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curc, &c_data_bp, errhp, (text
*)":c_data",
        strlen(":c_data"), (ub1 *)c_data, 501, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
    }
    /* item */
    if (do_A || do_i) {
        OCIERROR(errhp, OCIBindByName(curi, &i_id_bp, errhp, (text
*)":i_id",
        strlen(":i_id"), (ub1 *)i_id, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curi, &i_im_id_bp, errhp, (text
*)":i_im_id",
        strlen(":i_im_id"), (ub1 *)i_im_id, sizeof(int), SQLT_INT,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
        OCIERROR(errhp, OCIBindByName(curi, &i_name_bp, errhp, (text
*)":i_name",
        strlen(":i_name"), (ub1 *)i_name, 25, SQLT_STR,
        (dvoid *) 0, (ub2 *)0, (ub2 *)0,
        (ub4) 0, (ub4 *) 0, (ub4) OCI_DEFAULT));
    }
}

```



```

o_d_id_len[i] = sizeof(int);
o_w_id_len[i] = sizeof(int);
o_c_id_len[i] = sizeof(int);
o_carrier_id_len[i] = sizeof(int);
o_ol_cnt_len[i] = sizeof(int);
}

OCIERROR(errhp, OCIBindByName(curo1, &ol_o_id_bp, errhp, (text *)":ol_o_id",
    strlen(":ol_o_id"), (ub1 *)ol_o_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_o_id_len, (ub2 *)0,
    (ub4) 15*ORDEARR, (ub4 *)&ol_o_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_d_id_bp, errhp, (text *)":ol_d_id",
    strlen(":ol_d_id"), (ub1 *)ol_d_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_d_id_len, (ub2 *)0,
    (ub4) 15*ORDEARR, (ub4 *)&ol_d_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_w_id_bp, errhp, (text *)":ol_w_id",
    strlen(":ol_w_id"), (ub1 *)ol_w_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_w_id_len, (ub2 *)0,
    (ub4) 15*ORDEARR, (ub4 *)&ol_w_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_number_bp, errhp,
(text *)":ol_number",
    strlen(":ol_number"), (ub1 *)ol_number, sizeof(int),
SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_number_len, (ub2 *)0,
    (ub4) 15*ORDEARR, (ub4 *)&ol_number_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_i_id_bp, errhp, (text *)":ol_i_id",
    strlen(":ol_i_id"), (ub1 *)ol_i_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_i_id_len, (ub2 *)0,
    (ub4) 15*ORDEARR, (ub4 *)&ol_i_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_supply_w_id_bp,
errhp, (text *)":ol_supply_w_id",
    strlen(":ol_supply_w_id"), (ub1 *)ol_supply_w_id, sizeof(int),
SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_supply_w_id_len, (ub2 *)0,
    (ub4) 15*ORDEARR, (ub4 *)&ol_supply_w_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_dist_info_bp, errhp,
(text *)":ol_dist_info",
    strlen(":ol_dist_info"), (ub1 *)ol_dist_info, 24, SQLT_CHR,
    (dvoid *) 0, (ub2 *)ol_dist_info_len, (ub2 *)0,
    (ub4) 15*ORDEARR, (ub4 *)&ol_dist_info_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_amount_bp, errhp,
(text *)":ol_amount",
    strlen(":ol_amount"), (ub1 *)ol_amount, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_amount_len, (ub2 *)0,
    (ub4) 15*ORDEARR, (ub4 *)&ol_amount_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_o_id_bp, errhp, (text *)":ol_o_id",
    strlen(":ol_o_id"), (ub1 *)ol_o_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_o_id_len, (ub2 *)0,
    (ub4) ORDEARR, (ub4 *)&ol_o_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_d_id_bp, errhp, (text *)":ol_d_id",
    strlen(":ol_d_id"), (ub1 *)ol_d_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_d_id_len, (ub2 *)0,
    (ub4) ORDEARR, (ub4 *)&ol_d_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &ol_w_id_bp, errhp, (text *)":ol_w_id",
    strlen(":ol_w_id"), (ub1 *)ol_w_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)ol_w_id_len, (ub2 *)0,
    (ub4) ORDEARR, (ub4 *)&ol_w_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &o_c_id_bp, errhp, (text *)":o_c_id",
    strlen(":o_c_id"), (ub1 *)o_c_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)o_c_id_len, (ub2 *)0,
    (ub4) ORDEARR, (ub4 *)&o_c_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &o_carrier_id_bp, errhp,
(text *)":o_carrier_id",
    strlen(":o_carrier_id"), (ub1 *)o_carrier_id, sizeof(int),
SQLT_INT,
    (dvoid *) 0, (ub2 *)o_carrier_id_len, (ub2 *)0,
    (ub4) ORDEARR, (ub4 *)&o_carrier_id_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &o_ol_cnt_bp, errhp, (text *)":o_ol_cnt",
    strlen(":o_ol_cnt"), (ub1 *)o_ol_cnt, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)o_ol_cnt_len, (ub2 *)0,
    (ub4) ORDEARR, (ub4 *)&o_ol_cnt_clen, (ub4)
OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &o_cnt_bp, errhp, (text *)":order_rows",
    strlen(":order_rows"), (ub1 *)&o_cnt, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *)0, (ub4) OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curo1, &o_olcnt_bp, errhp, (text *)":ordl_rows",
    strlen(":ordl_rows"), (ub1 *)&ol_cnt, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *)0, (ub4) OCI_DEFAULT));

strlen(":ordl_rows"), (ub1 *)&ol_cnt, sizeof(int), SQLT_INT,
(dvoid *) 0, (ub2 *)0, (ub2 *)0,
(ub4) 0, (ub4 *)0, (ub4) OCI_DEFAULT));

/* new order */

if (do_A || do_n) {
OCIERROR(errhp, OCIBindByName(curno, &no_o_id_bp, errhp, (text *)":no_o_id",
    strlen(":no_o_id"), (ub1 *)no_o_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *)0, (ub4) OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curno, &no_d_id_bp, errhp, (text *)":no_d_id",
    strlen(":no_d_id"), (ub1 *)no_d_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *)0, (ub4) OCI_DEFAULT));

OCIERROR(errhp, OCIBindByName(curno, &no_w_id_bp, errhp, (text *)":no_w_id",
    strlen(":no_w_id"), (ub1 *)no_w_id, sizeof(int), SQLT_INT,
    (dvoid *) 0, (ub2 *)0, (ub2 *)0,
    (ub4) 0, (ub4 *)0, (ub4) OCI_DEFAULT));
}

/*-----+
| Initialize random number generator
|
+-----*/

srand (SEED);
#ifdef ORA_NT
srand48 (SEED);
#endif
initperm ();

/*-----+
| Load the WAREHOUSE table.
|
+-----*/

if (do_A || do_w) {
nrows = (long)eware - (long)bware + 1;

fprintf (stderr, "Loading/generating warehouse: w%d - w%d (%ld
rows)\n",
    bware, eware, nrows);

begin_time = gettimeofday ();
begin_cpu = getcpu ();

for (loop = bware; loop <= eware; loop++) {
w_tax = (float) ((lrand48 () % 2001) * 0.0001);
}
}

```



```

randstr (w_name, 6, 10);
randstr (w_street_1, 10, 20);
randstr (w_street_2, 10, 20);
randstr (w_city, 10, 20);
randstr (str2, 2, 2);
randnum (num9, 9);
num9[4] = num9[5] = num9[6] = num9[7] = num9[8] = '1';

if (gen) {
  printf ("%d 30000000 %6.4f %s %s %s %s %s %s\n", loop, w_tax,
    w_name, w_street_1, w_street_2, w_city, str2, num9);
  fflush (stdout);
}
else {
  w_id = loop;
  strncpy (w_state, str2, 2);
  strncpy (w_zip, num9, 9);

  status = OCISStmtExecute(tpcsvc, curw, errhp, (ub4) 1, (ub4) 0,
    (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
    (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
  if (status != OCI_SUCCESS) {
    fprintf (stderr, "Error at ware %d\n", loop);
    OCIERROR(errhp, status);
    quit ();
  }
  exit (1);
}
}
}

end_time = gettimeofday ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %ld rows loaded/generated in %10.2f sec. (%10.2f
cpu)\n\n", nrows, end_time - begin_time, end_cpu - begin_cpu);
}

/*-----+
| Load the DISTRICT table.
|
+-----*/

if (do_A || do_d) {
  nrows = ((long)eware - (long)bware + 1) * DISTFAC;

  fprintf (stderr, "Loading/generating district: w%d - w%d (%ld rows)\n",
    bware, eware, nrows);

  begin_time = gettimeofday ();
  begin_cpu = getcpu ();

  dwid = bware - 1;

  for (row = 0; row < nrows; ) {
    dwid++;

    for (i = 0; i < DISTARR; i++, row++) {
      d_tax[i] = (float) ((rand48 () % 2001) * 0.0001);
      randstr (d_name[i], 6, 10);

```

```

      randstr (d_street_1[i], 10, 20);
      randstr (d_street_2[i], 10, 20);
      randstr (d_city[i], 10, 20);
      randstr (str2, 2, 2);
      randnum (num9, 9);
      num9[4] = num9[5] = num9[6] = num9[7] = num9[8] = '1';

      if (gen) {
        printf ("%d %d 30000000 %6.4f 3001 %s %s %s %s %s %s\n",
          i + 1, dwid, d_tax[i], d_name[i], d_street_1[i],
            d_street_2[i], d_city[i], str2, num9 );
      }
      else {
        d_id[i] = i + 1;
        d_w_id[i] = dwid;
        strncpy (d_state[i], str2, 2);
        strncpy (d_zip[i], num9, 9);
      }
    }
  }

  if (gen) {
    fflush (stdout);
  }
  else {
    status = OCISStmtExecute(tpcsvc, curd, errhp, (ub4) DISTARR, (ub4)
      0,
      (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
      (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
    if (status != OCI_SUCCESS) {
      fprintf (stderr, "Aborted at ware %d, dist 1\n", dwid);
      OCIERROR(errhp, status);
      quit ();
      exit (1);
    }
  }
}

end_time = gettimeofday ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %ld rows loaded/generated in %10.2f sec. (%10.2f
cpu)\n\n", nrows, end_time - begin_time, end_cpu - begin_cpu);
}

/*-----+
| Load the CUSTOMER table.
|
+-----*/

if (do_A || do_c) {
  nrows = ((long)eware - (long)bware + 1) * CUSTFAC * DISTFAC;

  fprintf (stderr, "Loading/generating customer: w%d - w%d (%ld rows)\n",
    bware, eware, nrows);

  if (getenv("tpcc_hash_overflow")) {
    fprintf(stderr, "Hash overflow is enabled\n");

```

```

    OCIHandleAlloc(tpcenv, (dvoid **)&curi, OCI_HTYPE_STMT, 0,
      (dvoid**)0);
    sprintf ((char *) stmbuf, SQLTXTENHA);
    OCISStmtPrepare(cur, errhp, stmbuf, strlen((char *)stmbuf),
      OCI_NTV_SYNTAX, OCI_DEFAULT);
    OCIERROR(errhp, OCISStmtExecute(tpcsvc, curi,
      errhp, 1, 0, 0, OCI_DEFAULT));
    OCIHandleFree(cur, OCI_HTYPE_STMT);
    fprintf (stderr, "Customer loaded for horizontal partitioning\n");
  }
  else
  {
    fprintf (stderr, "Customer not loaded for horizontal partitioning\n");
  }
  begin_time = gettimeofday ();
  begin_cpu = getcpu ();

  s_c_id = 1;
  s_c_d_id = 1;
  s_c_w_id = bware;

  while (s_c_w_id <= eware) {
    status = OCISStmtExecute(tpcsvc, curcs, errhp, (ub4) 1, (ub4) 0,
      (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
      (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
    if (status != OCI_SUCCESS) {
      OCIERROR(errhp, status);
      quit ();
      exit (1);
    }

    if (s_c_count == 0) {
      s_c_w_id--;
      break;
    }
    else s_c_w_id++;
  }

  if (s_c_w_id < bware ) s_c_w_id = bware;
  else {
    if (s_c_w_id > eware ) s_c_w_id = eware;
    while (s_c_d_id <= DISTFAC) {
      status = OCISStmtExecute(tpcsvc, curcs, errhp, (ub4) 1, (ub4) 0,
        (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
        (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
      if (status != OCI_SUCCESS) {
        fprintf (stderr, "Select failed\n");
        OCIERROR(errhp, status);
        quit ();
        exit (1);
      }
    }
    if (s_c_count == 0) {
      s_c_d_id--;
      break;
    }
    else s_c_d_id++;
  }
  if (s_c_d_id > DISTFAC) s_c_d_id = DISTFAC;

```

```

while (s_c_id <= CUSTFAC) {
    status = OCISmtExecute(tpcsvc, curcs, errhp, (ub4) 1, (ub4) 0,
        (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
        (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
    if (status != OCI_SUCCESS) {
        OCIERROR(errhp, status);
        quit ();
        exit (1);
    }
    if (s_c_count == 0) break;
    else s_c_id++;
}
if (s_c_id > CUSTFAC) {
    if (s_c_d_id == DISTFAC) {
        s_c_d_id = 1;
        s_c_w_id++;
    } else {
        s_c_d_id++;
    }
    s_c_id = 1;
}

fprintf (stderr, "start at wid: %d, did: %d, cid: %d\n ", s_c_w_id,
s_c_d_id, s_c_id);
cid = s_c_id - 1;
cdid = s_c_d_id;
cwid = s_c_w_id;
nrows = ((long)eware - (long)s_c_w_id + 1) * DISTFAC * CUSTFAC -
((long)s_c_d_id - 1) * CUSTFAC - (long)s_c_id + 1;
fprintf (stderr, "remaining rows: %ld\n ", nrows);
loopcount = 0;

for (row = 0; row < nrows; ) {
    for (i = 0; i < CUSTARR && row < nrows; i++, row++) {
        cid++;
        if (cid > CUSTFAC) { /* cycle cust id */
            cid = 1; /* cheap mod */
            cdid++; /* shift dist cycle */
            if (cdid > DISTFAC) {
                cdid = 1;
                cwid++; /* shift ware cycle */
            }
        }
        c_id[i] = cid;
        c_d_id[i] = cdid;
        c_w_id[i] = cwid;
        if (cid <= 1000)
            randlastname (c_last[i], cid - 1);
        else
            randlastname (c_last[i], NURand (255, 0, 999, CNUM1));
        c_credit[i][1] = 'C';
        if (lrand48 () % 10)
            c_credit[i][0] = 'G';
        else
            c_credit[i][0] = 'B';
        c_discount[i] = (float)(lrand48 () % 5001) * 0.0001;

        randstr (c_first[i], 8, 16);
        randstr (c_street_1[i], 10, 20);
        randstr (c_street_2[i], 10, 20);
        randstr (c_city[i], 10, 20);
        randstr (str2, 2, 2);
        randnum (num9, 9);
        num9[4] = num9[5] = num9[6] = num9[7] = num9[8] = '1';
        randnum (num16, 16);
        randstr (c_data[i], 300, 500);

        if (gen) {
            printf ("%d %d %d %s OE %s %s %s %s %s %s %s %s %cC
5000000 %6.4f-1000 1000 1 0 %s\n",
                cid, cdid, cwid, c_first[i], c_last[i],
                c_street_1[i], c_street_2[i], c_city[i], str2, num9,
                num16, sdate, c_credit[i][0], c_discount[i], c_data[i]);
        }
        else {
            strncpy (c_state[i], str2, 2);
            strncpy (c_zip[i], num9, 9);
            strncpy (c_phone[i], num16, 16);
        }
    }
}

if (gen) {
    fflush (stdout);
}
else {
    status = OCISmtExecute(tpcsvc, curc, errhp, (ub4) i, (ub4) 0,
        (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
        (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);

    if (status != OCI_SUCCESS) {
        fprintf (stderr, "Aborted at w_id %d, d_id %d, c_id
%d\n",
                c_w_id[0], c_d_id[0], c_id[0]);
        OCIERROR(errhp, status);
        quit ();
        exit (1);
    }
}

if ((++loopcount) % 50)
    fprintf (stderr, ".");
else
    fprintf (stderr, "%ld rows committed\n ", row);
}

end_time = gettimeofday ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %ld rows loaded/generated in %10.2f sec. (%10.2f
cpu)\n\n", nrows < 0 ? 0 : nrows, end_time - begin_time, end_cpu -
begin_cpu);
if (getenv("tpcc_hash_overflow")) {
    fprintf(stderr, "Hash overflow is disabled\n");
    OCIHandleAlloc(tpcenv, (dvoid **)&curi, OCI_HTYPE_STMT, 0,
(dvoid**));
}
sprintf ((char *) stmbuf, SQLTXTDIHA);

OCISmtPrepare(curi, errhp, stmbuf, strlen((char *)stmbuf),
    OCI_NTV_SYNTAX, OCI_DEFAULT);
OCIERROR(errhp, OCISmtExecute(tpcsvc, curi,
errhp, 1, 0, 0, OCI_DEFAULT));
OCIHandleFree(curi, OCI_HTYPE_STMT);
}
}

/*-----+
| Load the CUSTOMER table (cluster around c_id)
+-----*/

if (do_C) {
    srand (bcid);
#ifdef ORA_NT
    srand48 (bcid);
#endif

    nrows = ((long)ecid - (long)bcid + 1) * ((long)eware - (long)bware + 1) *
DISTFAC;

    fprintf (stderr, "Loading/generating customer: c%d - c%d, w%d - w%d
(%ld rows)\n ", bcid, ecid, bware, eware, nrows);

    if (getenv("tpcc_hash_overflow")) {
        fprintf(stderr, "Hash overflow is enabled\n");
        OCIHandleAlloc(tpcenv, (dvoid **)&curi, OCI_HTYPE_STMT, 0,
(dvoid**));
        sprintf ((char *) stmbuf, SQLTXTENHA);
        OCISmtPrepare(curi, errhp, stmbuf, strlen((char *)stmbuf),
            OCI_NTV_SYNTAX, OCI_DEFAULT);
        OCIERROR(errhp, OCISmtExecute(tpcsvc, curi,
errhp, 1, 0, 0, OCI_DEFAULT));
        OCIHandleFree(curi, OCI_HTYPE_STMT);
        fprintf (stderr, "Customer loaded for horizontal partitioning\n");
    }
    else
    {
        fprintf (stderr, "Customer not loaded for horizontal partitioning\n");
    }
    begin_time = gettimeofday ();
    begin_cpu = getcpu ();

    s_c_id = bcid;
    s_c_d_id = 1;
    s_c_w_id = bware;

    while (s_c_id <= ecid) {
        status = OCISmtExecute(tpcsvc, curcs, errhp, (ub4) 1, (ub4) 0,
            (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
            (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            OCIERROR(errhp, status);
            quit ();
            exit (1);
        }
    }
}
}

```

```

if (s_c_count == 0) {
    s_c_id--;
    break;
}
else s_c_id++;
}
}

if (s_c_id < bcid) s_c_id = bcid;
else {
    if (s_c_id > ecid) s_c_id = ecid;
    while (s_c_w_id <= eware) {
        status = OCISStmtExecute(tpcsvc, curcs, errhp, (ub4) 1, (ub4) 0,
            (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
            (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            fprintf(stderr, "Select failed\n");
            OCIERROR(errhp, status);
            quit ();
            exit (1);
        }
    }
    if (s_c_count == 0) {
        s_c_w_id--;
        break;
    }
    else s_c_w_id++;
}
}

if (s_c_w_id > eware) s_c_w_id = eware;
else if (s_c_w_id < bware) s_c_w_id = bware;

while (s_c_d_id <= DISTFAC) {
    status = OCISStmtExecute(tpcsvc, curcs, errhp, (ub4) 1, (ub4) 0,
        (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
        (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
    if (status != OCI_SUCCESS) {
        OCIERROR(errhp, status);
        quit ();
        exit (1);
    }
    if (s_c_count == 0) break;
    else s_c_d_id++;
}
}

if (s_c_d_id > DISTFAC) {
    s_c_d_id=1;
    if (s_c_w_id==eware) {
        s_c_w_id=bware;
        s_c_id++;
    }
    else s_c_w_id++;
}
}

fprintf(stderr, "start at cid: %d, wid: %d, did: %d\n", s_c_id, s_c_w_id,
s_c_d_id);
cid = s_c_id;
cdid = s_c_d_id-1;
cwid = s_c_w_id;

nrows = ((long)ecid - (long)s_c_id + 1) * ((long)eware - (long)bware +
1) * DISTFAC - ((long)s_c_w_id - 1) * DISTFAC - (long)s_c_d_id + 1;
fprintf(stderr, "remaining rows: %ld\n", nrows);
loopcount = 0;

for (row = 0; row < nrows; ) {
    for (i = 0; i < CUSTARR && row < nrows; i++, row++) {
        cdid++;
        if (cdid > DISTFAC) { /* cycle dist id */
            cdid = 1; /* cheap mod */
            cwid++; /* shift dist cycle */
            if (cwid > eware) {
                cwid = bware; /* shift ware cycle */
                cid++;
            }
        }
        c_id[i] = cid;
        c_d_id[i] = cdid;
        c_w_id[i] = cwid;
        if (cid <= 1000)
            randlastname (c_last[i], cid - 1);
        else
            randlastname (c_last[i], NURand (255, 0, 999, CNUM1));
        c_credit[i][1] = 'C';
        if (lrand48 () % 10)
            c_credit[i][0] = 'G';
        else
            c_credit[i][0] = 'B';
        c_discount[i] = (float)((lrand48 () % 5001) * 0.0001);
        randstr (c_first[i], 8, 16);
        randstr (c_street_1[i], 10, 20);
        randstr (c_street_2[i], 10, 20);
        randstr (c_city[i], 10, 20);
        randstr (str2, 2, 2);
        randnum (num9, 9);
        num9[4] = num9[5] = num9[6] = num9[7] = num9[8] = '1';
        randnum (num16, 16);
        randstr (c_data[i], 300, 500);

        if (gen) {
            printf ("%d %d %d %s OE %s %s %s %s %s %s %s %s %cC
5000000 %6.4f -1000 1000 1 0 %s\n",
                cid, cdid, cwid, c_first[i], c_last[i],
                c_street_1[i], c_street_2[i], c_city[i], str2, num9,
                num16, sdate, c_credit[i][0], c_discount[i], c_data[i]);
        }
        else {
            strncpy (c_state[i], str2, 2);
            strncpy (c_zip[i], num9, 9);
            strncpy (c_phone[i], num16, 16);
        }
    }
}

if (gen) {
    fflush (stdout);
}
else {
    status = OCISStmtExecute(tpcsvc, curc, errhp, (ub4) i, (ub4) 0,
        (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
        (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
    if (status != OCI_SUCCESS) {
        fprintf(stderr, "Aborted at w_id %d, d_id %d, c_id
%d\n",
            c_w_id[0], c_d_id[0], c_id[0]);
        OCIERROR(errhp, status);
        quit ();
        exit (1);
    }
}

if ((++loopcount) % 50)
    fprintf(stderr, ".");
else
    fprintf(stderr, "%ld rows committed\n", row);
}

end_time = gettime ();
end_cpu = getcpu ();
fprintf(stderr, "Done. %ld rows loaded/generated in %10.2f sec. (%10.2f
cpu)\n\n", nrows < 0 ? 0 : nrows, end_time - begin_time, end_cpu -
begin_cpu);
if (getenv("tpcc_hash_overflow")) {
    fprintf(stderr, "Hash overflow is disabled\n");
    OCIHandleAlloc(tpcenv, (dvoid **)&curi, OCI_HTYPE_STMT, 0,
(dvoid**)0);
    sprintf ((char *) stmbuf, SQLTXTDIHA);
    OCISStmtPrepare(curi, errhp, stmbuf, strlen((char *)stmbuf),
        OCI_NTV_SYNTAX, OCI_DEFAULT);
    OCIERROR(errhp, OCISStmtExecute(tpcsvc, curi,
errhp, 1, 0, 0, 0, OCI_DEFAULT));
    OCIHandleFree(curi, OCI_HTYPE_STMT);
}
}

/*-----+
| Load the ITEM table.
+-----*/

if (do_A || do_i) {
    nrows = ITEMFAC;

    fprintf(stderr, "Loading/generating item: (%ld rows)\n", nrows);

    begin_time = gettime ();
    begin_cpu = getcpu ();

    loopcount = 0;

    for (row = 0; row < nrows; ) {
        for (i = 0; i < ITEMARR; i++, row++) {
            i_im_id[i] = (lrand48 () % 10000) + 1;
            i_price[i] = ((lrand48 () % 9901) + 100);
            randstr (i_name[i], 14, 24);

```

```

randdatastr (i_data[i], 26, 50);

if (gen) {
    printf ("%d %d %s %d %s\n", row + 1, i_im_id[i], i_name[i],
            i_price[i], i_data[i]);
}
else {
    i_id[i] = row + 1;
}
}

if (gen) {
    fflush (stdout);
}
else {
    status = OCISstmtExecute(tpcsvc, curi, errhp, (ub4) ITEMARR, (ub4)
0,
        (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
        (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
if (status != OCI_SUCCESS) {
    fprintf (stderr, "Aborted at i_id %d\n", i_id[0]);
    OCIERROR(errhp, status);
    quit ();
    exit (1);
}
}

if ((++loopcount) % 50)
    fprintf (stderr, ".");
else
    fprintf (stderr, "%ld rows committed\n ", row);
}

end_time = gettimeofday ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %ld rows loaded/generated in %10.2f sec. (%10.2f
cpu)\n\n", nrows, end_time - begin_time, end_cpu - begin_cpu);
}

/*-----
| Load the STOCK table.
|
+-----*/

if (do_A || do_s) {

    nrows = ((long)eware - (long)bware + 1) * STOCFAC;

    fprintf (stderr, "Loading/generating stock: w%d - w%d (%ld rows)\n ",
            bware, eware, nrows);

    begin_time = gettimeofday ();
    begin_cpu = getcpu ();

    s_s_i_id = 1;
    s_s_w_id = bware;

while (s_s_w_id <= eware) {
    status = OCISstmtExecute(tpcsvc, curss, errhp, (ub4) 1, (ub4) 0,
        (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
        (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
    if (status != OCI_SUCCESS) {
        OCIERROR(errhp, status);
        quit ();
        exit (1);
    }
}
if (s_s_count == 0) {
    s_s_w_id--;
    break;
}
else s_s_w_id++;

if (s_s_w_id < bware) s_s_w_id = bware;
else {
    if (s_s_w_id > eware) s_s_w_id = eware;
    while (s_s_i_id <= STOCFAC) {
        status = OCISstmtExecute(tpcsvc, curss, errhp, (ub4) 1, (ub4) 0,
            (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
            (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            OCIERROR(errhp, status);
            quit ();
            exit (1);
        }
    }
    if (s_s_count == 0) {
        break;
    }
    else s_s_i_id++;
}
if (s_s_i_id > STOCFAC) {
    s_s_i_id = 1;
    s_s_w_id++;
}

fprintf(stderr, "start at s_i_id: %d, s_w_id: %d\n ", s_s_i_id, s_s_w_id);

    sid = s_s_i_id - 1;
    swid = s_s_w_id;
    nrows = ((long)s_s_w_id + 1) * STOCFAC -
    ((long)s_s_i_id - 1);
    fprintf (stderr, "remaining rows: %ld\n ", nrows);
    loopcount = 0;

    for (row = 0; row < nrows; ) {
        /* added row < nrows condition on next line - alex.ni */
        for (i = 0; (i < STOCARR) && (row < nrows); i++, row++) {
            if (++sid > STOCFAC) { /* cheap mod */
                sid = 1;
                swid++;
            }
            s_quantity[i] = (lrand48 () % 91) + 10;
            randstr (s_dist_01[i], 24, 24);
            randstr (s_dist_02[i], 24, 24);
            randstr (s_dist_03[i], 24, 24);
            randstr (s_dist_04[i], 24, 24);
            randstr (s_dist_05[i], 24, 24);
            randstr (s_dist_06[i], 24, 24);
            randstr (s_dist_07[i], 24, 24);
            randstr (s_dist_08[i], 24, 24);
            randstr (s_dist_09[i], 24, 24);
            randstr (s_dist_10[i], 24, 24);
            randdatastr (s_data[i], 26, 50);

if (gen) {
    printf ("%d %d %d %s %s %s %s %s %s %s %s %s %s %s %s %s %s %s %s %s 0 0 0
%s\n",
            sid, swid, s_quantity[i], s_dist_01[i], s_dist_02[i],
            s_dist_03[i], s_dist_04[i], s_dist_05[i], s_dist_06[i],
            s_dist_07[i], s_dist_08[i], s_dist_09[i], s_dist_10[i],
            s_data[i]);
}
else {
    s_i_id[i] = sid;
    s_w_id[i] = swid;
}

if (gen) {
    fflush (stdout);
}
else {
    /* Changed to STOCKARR to i - alex.ni */
    status = OCISstmtExecute(tpcsvc, curs, errhp, (ub4) i, (ub4) 0,
        (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
        (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
    if (status != OCI_SUCCESS) {
        fprintf (stderr, "Aborted at w_id %d, s_i_id %d\n",
            s_w_id[0], s_i_id[0]);
        OCIERROR(errhp, status);
        quit ();
        exit (1);
    }
}

if ((++loopcount) % 50)
    fprintf (stderr, ".");
else
    fprintf (stderr, "%ld rows committed\n ", row);
}

    end_time = gettimeofday ();
    end_cpu = getcpu ();
    fprintf (stderr, "Done. %ld rows loaded/generated in %10.2f sec. (%10.2f
cpu)\n\n", nrows < 0 ? 0 : nrows, end_time - begin_time, end_cpu -
begin_cpu);
}

/*-----
| Load the STOCK table (cluster around s_i_id).
+-----*/

```

```

if (do_S) {
    nrows = ((long)eitem - (long)bitem + 1) * ((long)eware - (long)bware + 1);

    fprintf(stderr, "Loading/generating stock: i%d - i%d, w%d - w%d (%ld rows)\n ", bitem, eitem, bware, eware, nrows);

    begin_time = gettimeofday ();
    begin_cpu = getcpu ();

    s_s_i_id = bitem;
    s_s_w_id = bware;

    while (s_s_i_id <= eitem) {
        status = OCIStmtExecute(tpscvc, curss, errhp, (ub4) 1, (ub4) 0,
            (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
            (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            OCIERROR(errhp, status);
            quit ();
            exit (1);
        }
        if (s_s_count == 0) {
            s_s_i_id--;
            break;
        }
        else s_s_i_id++;
    }

    if (s_s_i_id < bitem) s_s_i_id = bitem;
    else {
        if (s_s_i_id > eitem) s_s_i_id = eitem;
        while (s_s_w_id <= eware) {
            status = OCIStmtExecute(tpscvc, curss, errhp, (ub4) 1, (ub4) 0,
                (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
                (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
            if (status != OCI_SUCCESS) {
                OCIERROR(errhp, status);
                quit ();
                exit (1);
            }
            if (s_s_count == 0) {
                break;
            }
            else s_s_w_id++;
        }
        if (s_s_w_id > eware) {
            s_s_w_id = bware;
            s_s_i_id++;
        }
    }

    fprintf(stderr, "start at s_i_id: %d, s_w_id: %d\n ", s_s_i_id, s_s_w_id);

    sid = s_s_i_id;
    swid = s_s_w_id - 1;
    nrows = ((long)eitem - (long)s_s_i_id + 1) * ((long)eware - (long)bware
+ 1) - ((long)s_s_w_id - (long)bware);
    fprintf(stderr, "remaining rows: %ld\n ", nrows);
    loopcount = 0;

    for (row = 0; row < nrows; ) {
        for (i = 0; i < STOCARR && row < nrows; i++, row++) {
            if (++swid > eware) { /* cheap mod */
                swid = bware;
                sid++;
            }
            s_quantity[i] = (lrand48 () % 91) + 10;
            randstr (s_dist_01[i], 24, 24);
            randstr (s_dist_02[i], 24, 24);
            randstr (s_dist_03[i], 24, 24);
            randstr (s_dist_04[i], 24, 24);
            randstr (s_dist_05[i], 24, 24);
            randstr (s_dist_06[i], 24, 24);
            randstr (s_dist_07[i], 24, 24);
            randstr (s_dist_08[i], 24, 24);
            randstr (s_dist_09[i], 24, 24);
            randstr (s_dist_10[i], 24, 24);
            randdatastr (s_data[i], 26, 50);

            if (gen) {
                printf ("%d %d %d %s %s %s %s %s %s %s %s %s %s 0 0 0
%s\n",
                    sid, swid, s_quantity[i], s_dist_01[i], s_dist_02[i],
                    s_dist_03[i], s_dist_04[i], s_dist_05[i], s_dist_06[i],
                    s_dist_07[i], s_dist_08[i], s_dist_09[i], s_dist_10[i],
                    s_data[i]);
            }
            else {
                s_i_id[i] = sid;
                s_w_id[i] = swid;
            }
        }
        if (gen) {
            fflush (stdout);
        }
        else {
            status = OCIStmtExecute(tpscvc, curs, errhp, (ub4) i, (ub4) 0,
                (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
                (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
            if (status != OCI_SUCCESS) {
                fprintf (stderr, "Aborted at w_id %d, s_i_id %d\n", s_w_id[0],
                    s_i_id[0]);
                OCIERROR(errhp, status);
                quit ();
                exit (1);
            }
        }
    }

    if ((++loopcount) % 50)
        fprintf (stderr, ".");
    else
        fprintf (stderr, " %ld rows committed\n ", row);
}

end_time = gettimeofday ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %ld rows loaded/generated in %10.2f sec. (%10.2f
cpu)\n\n", nrows < 0 ? 0 : nrows, end_time - begin_time, end_cpu -
begin_cpu);

/*-----+
| Load the HISTORY table.
|
+-----*/

if (do_A || do_h) {
    nrows = ((long)eware - (long)bware + 1) * HISTFAC;

    fprintf (stderr, "Loading/generating history: w%d - w%d (%ld rows)\n ",
        bware, eware, nrows);

    begin_time = gettimeofday ();
    begin_cpu = getcpu ();

    cid = 0;
    cdid = 1;
    cwid = bware;
    loopcount = 0;

    for (row = 0; row < nrows; ) {
        for (i = 0; i < HISTARR; i++, row++) {
            cid++;
            if (cid > CUSTFAC) { /* cycle cust id */
                cid = 1; /* cheap mod */
                cdid++; /* shift district cycle */
                if (cdid > DISTFAC) {
                    cdid = 1;
                    cwid++; /* shift warehouse cycle */
                }
            }
            h_c_id[i] = cid;
            h_d_id[i] = cdid;
            h_w_id[i] = cwid;
            randstr (h_data[i], 12, 24);
            if (gen) {
                printf ("%d %d %d %d %d %s 1000 %s\n", cid, cdid, cwid, cdid,
                    cwid, sdate, h_data[i]);
            }
        }
    }

    if (gen) {
        fflush (stdout);
    }
    else {
        status = OCIStmtExecute(tpscvc, curh, errhp, (ub4) HISTARR, (ub4)
            0,
            (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
            (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS);
        if (status != OCI_SUCCESS) {
            fprintf (stderr, "Aborted at w_id %d, d_id %d, c_id

```

```

%d\n",
    h_w_id[0], h_d_id[0], h_c_id[0]);
OCIERROR(errhp, status);
quit ();
exit (1);
}
}

if ((++loopcount) % 50)
    fprintf (stderr, ".");
else
    fprintf (stderr, "%ld rows committed\n ", row);
}

end_time = gettimeofday ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %ld rows loaded/generated in %10.2f sec. (%10.2f
cpu)\n\n", nrows, end_time - begin_time, end_cpu - begin_cpu);
}

/*-----+
| Load the ORDERS and ORDER-LINE table.
|
+-----*/

if (do_A || do_o) {

    int batch_olcnt;

    nrows = ((long)eware - (long)bware + 1) * ORDEFAC * DISTFAC;

    fprintf (stderr, "Loading/generating orders and order-line: w%d - w%d
(%ld ord, ~%ld ord)\n ", bware, eware, nrows, nrows * 10);

    begin_time = gettimeofday ();
    begin_cpu = getcpu ();

    cid = 0;
    cdid = 1;
    cwid = bware;
    loopcount = 0;

    for (row = 0; row < nrows; ) {

        batch_olcnt = 0;

        for (i = 0; i < ORDEARR; i++, row++) {
            cid++;
            if (cid > ORDEFAC) { /* cycle cust id */
                cid = 1; /* cheap mod */
                cdid++; /* shift district cycle */
                if (cdid > DISTFAC) {
                    cdid = 1;
                    cwid++; /* shift warehouse cycle */
                }
            }
            o_carrier_id[i] = lrand48 () % 10 + 1;
            o_ol_cnt[i] = olcnt = lrand48 () % 11 + 5;

            if (gen) {
                if (cid < 2101) {
                    printf ("%d %d %d %d %s %d %d l\n", cid, cdid, cwid,
                        randperm3000[cid - 1], sdate, o_carrier_id[i],
                        o_ol_cnt[i]);
                }
                else {
                    /* set carrierid to 11 instead of null */
                    printf ("%d %d %d %d %s 11 %d l\n", cid, cdid, cwid,
                        randperm3000[cid - 1], sdate, o_ol_cnt[i]);
                }
            }
            else {
                o_id[i] = cid;
                o_d_id[i] = cdid;
                o_w_id[i] = cwid;
                o_c_id[i] = randperm3000[cid - 1];
                if (cid >= 2101) {
                    o_carrier_id[i] = 11;
                }
            }

            for (j = 0; j < o_ol_cnt[i]; j++, batch_olcnt++) {
                ol_i_id[batch_olcnt] = sid = lrand48 () % 100000 + 1;
                if (cid < 2101)
                    ol_amount[batch_olcnt] = 0;
                else
                    ol_amount[batch_olcnt] = (lrand48 () % 999999 + 1) ;
                randstr (str24[j], 24, 24);

                if (gen) {
                    if (cid < 2101) {
                        fprintf (olfp, "%d %d %d %d %s %d %d 5 %ld %s\n", cid,
                            cdid, cwid, j + 1, sdate, ol_i_id[batch_olcnt], cwid,
                            ol_amount[batch_olcnt], str24[j]);
                    }
                    else {
                        /* Insert a default date instead of null date */
                        fprintf (olfp, "%d %d %d %d 01-Jan-1811 %d %d 5 %ld %s\n",
                            cid,
                            cdid, cwid, j + 1, ol_i_id[batch_olcnt], cwid,
                            ol_amount[batch_olcnt], str24[j]);
                    }
                }
                else {
                    ol_o_id[batch_olcnt] = cid;
                    ol_d_id[batch_olcnt] = cdid;
                    ol_w_id[batch_olcnt] = cwid;
                    ol_number[batch_olcnt] = j + 1;
                    ol_supply_w_id[batch_olcnt] = cwid;
                    strncpy (ol_dist_info[batch_olcnt], str24[j], 24);
                }
            }
            if (gen) {
                fflush (olfp);
            }
        }
        o_cnt = ORDEARR;
        ol_cnt = batch_olcnt;

        for (j = 0; j < batch_olcnt; j++) {
            ol_o_id_len[j] = sizeof(int);
            ol_d_id_len[j] = sizeof(int);
            ol_w_id_len[j] = sizeof(int);
            ol_number_len[j] = sizeof(int);
            ol_i_id_len[j] = sizeof(int);
            ol_supply_w_id_len[j] = sizeof(int);
            ol_dist_info_len[j] = 24;
            ol_amount_len[j] = sizeof(int);
        }
        for (j = batch_olcnt; j < 15*ORDEARR; j++) {
            ol_o_id_len[j] = 0;
            ol_d_id_len[j] = 0;
            ol_w_id_len[j] = 0;
            ol_number_len[j] = 0;
            ol_i_id_len[j] = 0;
            ol_supply_w_id_len[j] = 0;
            ol_dist_info_len[j] = 0;
            ol_amount_len[j] = 0;
        }

        o_id_clen = ORDEARR;
        o_d_id_clen = ORDEARR;
        o_w_id_clen = ORDEARR;
        o_c_id_clen = ORDEARR;
        o_carrier_id_clen = ORDEARR;
        o_ol_cnt_clen = ORDEARR;

        ol_o_id_clen = batch_olcnt;
        ol_d_id_clen = batch_olcnt;
        ol_w_id_clen = batch_olcnt;
        ol_number_clen = batch_olcnt;
        ol_i_id_clen = batch_olcnt;
        ol_supply_w_id_clen = batch_olcnt;
        ol_dist_info_clen = batch_olcnt;
        ol_amount_clen = batch_olcnt;

        OCIERROR(errhp, OCIStmtExecute(tpcsvc, cur1, errhp, (ub4) 1,
            (ub4) 0,
                (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
                (ub4) OCI_DEFAULT | OCI_COMMIT_ON_SUCCESS));

        if ((++loopcount) % 50) {
            fprintf (stderr, ".");
        } else {
            fprintf (stderr, "%ld orders committed\n ", row);
        }
    }

    end_time = gettimeofday ();
    end_cpu = getcpu ();
    fprintf (stderr, "Done. %ld orders loaded/generated in %10.2f sec.
(%10.2f cpu)\n\n", nrows, end_time - begin_time, end_cpu - begin_cpu);
}
}

```

```

/*-----+
| Load the NEW-ORDER table.
|
+-----*/
if (do_A || do_n) {
    nrows = ((long)eware - (long)bware + 1) * NEWOFAC * DISTFAC;

    fprintf(stderr, "Loading/generating new-order: w%d - w%d (%ld rows)\n",
        bware, eware, nrows);

    begin_time = gettimeofday ();
    begin_cpu = getcpu ();

    cid = 0;
    cdid = 1;
    cwid = bware;
    loopcount = 0;

    for (row = 0; row < nrows; ) {
        for (i = 0; i < NEWOARR; i++, row++) {
            cid++;
            if (cid > NEWOFAC) {
                cid = 1;
                cdid++;
                if (cdid > DISTFAC) {
                    cdid = 1;
                    cwid++;
                }
            }

            if (gen) {
                printf ("%d %d %d\n", cid + 2100, cdid, cwid);
            }
            else {
                no_o_id[i] = cid + 2100;
                no_d_id[i] = cdid;
                no_w_id[i] = cwid;
            }
        }

        if (gen) {
            fflush (stdout);
        }
        else {
            status = OCIStmtExecute(tpcsvc, curno, errhp, (ub4) NEWOARR,
                (CONST OCISnapshot*) 0, (OCISnapshot*) 0,
                (ub4) OCI_DEFAULT |
OCI_COMMIT_ON_SUCCESS);
            if (status != OCI_SUCCESS) {
                fprintf (stderr, "Aborted at w_id %d, d_id %d, o_id
%d\n", cwid, cdid, cid + 2100);
                OCIERROR(errhp, status);
                quit ();
                exit (1);
            }
        }
    }

    if ((++loopcount) % 45)
        fprintf (stderr, ".");
    else
        fprintf (stderr, "%ld rows committed\n ", row);
}

end_time = gettimeofday ();
end_cpu = getcpu ();
fprintf (stderr, "Done. %ld rows loaded/generated in %10.2f sec. (%10.2f
cpu)\n\n", nrows, end_time - begin_time, end_cpu - begin_cpu);

/*-----+
| clean up and exit.
|
+-----*/

if (olfp)
    fclose (olfp);
if (!gen)
    quit ();
exit (0);

void initperm ()
{
    int i;
    int pos;
    int temp;

    /* init randperm3000 */

    for (i = 0; i < 3000; i++)
        randperm3000[i] = i + 1;
    for (i = 3000; i > 0; i--) {
        pos = lrand48 () % i;
        temp = randperm3000[i - 1];
        randperm3000[i - 1] = randperm3000[pos];
        randperm3000[pos] = temp;
    }
}

void randstr (str, x, y)
char *str;
int x;
int y;
{
    int i, j;
    int len;

    len = (lrand48 () % (y - x + 1)) + x;
    for (i = 0; i < len; i++) {
        j = lrand48 () % 62;
        if (j < 26)
            str[i] = (char) (j + 'a');
        else if (j < 52)
            str[i] = (char) (j - 26 + 'A');
        else
            str[i] = (char) (j - 52 + '0');
    }
    str[len] = '\0';

    if ((lrand48 () % 10) == 0) {
        pos = (lrand48 () % (len - 8));
        str[pos] = 'O';
        str[pos + 1] = 'R';
        str[pos + 2] = 'T';
        str[pos + 3] = 'G';
        str[pos + 4] = 'I';
        str[pos + 5] = 'N';
        str[pos + 6] = 'A';
        str[pos + 7] = 'L';
    }
}

void randnum (str, len)
char *str;
int len;
{
    int i;

    for (i = 0; i < len; i++)
        str[i] = (char) (lrand48 () % 10 + '0');
    str[len] = '\0';
}

void randlastname (str, id)
char *str;
int id;
{
    else if (j < 52)
        str[i] = (char) (j - 26 + 'A');
    else
        str[i] = (char) (j - 52 + '0');
}

void randdatastr (str, x, y)
int x;
int y;
{
    int i, j;
    int len;
    int pos;

    len = (lrand48 () % (y - x + 1)) + x;
    for (i = 0; i < len; i++) {
        j = lrand48 () % 62;
        if (j < 26)
            str[i] = (char) (j + 'a');
        else if (j < 52)
            str[i] = (char) (j - 26 + 'A');
        else
            str[i] = (char) (j - 52 + '0');
    }
    str[len] = '\0';

    if ((lrand48 () % 10) == 0) {
        pos = (lrand48 () % (len - 8));
        str[pos] = 'O';
        str[pos + 1] = 'R';
        str[pos + 2] = 'T';
        str[pos + 3] = 'G';
        str[pos + 4] = 'I';
        str[pos + 5] = 'N';
        str[pos + 6] = 'A';
        str[pos + 7] = 'L';
    }
}

void randnum (str, len)
char *str;
int len;
{
    int i;

    for (i = 0; i < len; i++)
        str[i] = (char) (lrand48 () % 10 + '0');
    str[len] = '\0';
}

void randlastname (str, id)
char *str;
int id;
{

```

```

id = id % 1000;
strcpy (str, lastname[id / 100]);
strcat (str, lastname[(id / 10) % 10]);
strcat (str, lastname[id % 10]);
}

int NURand (A, x, y, cnum)
int A, x, y, cnum;
{
  int a, b;

  a = lrand48 () % (A + 1);
  b = (lrand48 () % (y - x + 1)) + x;
  return (((a | b) + cnum) % (y - x + 1)) + x;
}

void sysdate (sdate)
char *sdate;
{
  time_t tp;
  struct tm *tmptr;

  time (&tp);
  tmptr = localtime (&tp);
  strftime (sdate, 29, "%d-%b-%Y", tmptr);
}

int ocierror(fname, lineno, errhp, status)
char *fname;
int lineno;
OCIError *errhp;
sword status;
{
  text errbuf[512];
  sb4 errcode;
  sb4 lstat;
  ub4 recno=2;

  switch (status) {
  case OCI_SUCCESS:
    break;
  case OCI_SUCCESS_WITH_INFO:
    fprintf(stderr,"Module %s Line %d\n", fname, lineno);
    fprintf(stderr,"Error - OCI_SUCCESS_WITH_INFO\n");
    lstat = OCIErrorGet (errhp, recno++, (text *) NULL, &errcode, errbuf,
      (ub4) sizeof(errbuf), OCI_HTYPE_ERROR);
    fprintf(stderr,"Error - %s\n", errbuf);
    break;
  case OCI_NEED_DATA:
    fprintf(stderr,"Module %s Line %d\n", fname, lineno);
    fprintf(stderr,"Error - OCI_NEED_DATA\n");
    return (IRRECERR);
  case OCI_NO_DATA:
    fprintf(stderr,"Module %s Line %d\n", fname, lineno);
    fprintf(stderr,"Error - OCI_NO_DATA\n");
    return (IRRECERR);
  case OCI_ERROR:
    lstat = OCIErrorGet (errhp, (ub4) 1,
      (text *) NULL, &errcode, errbuf,
      (ub4) sizeof(errbuf),
      OCI_HTYPE_ERROR);
    if (errcode == NOT_SERIALIZABLE) return (errcode);
    if (errcode == SNAPSHOT_TOO_OLD) return (errcode);
    while (lstat != OCI_NO_DATA)
    {
      fprintf(stderr,"Module %s Line %d\n", fname, lineno);
      fprintf(stderr,"Error - %s\n", errbuf);
      lstat = OCIErrorGet (errhp, recno++, (text *) NULL, &errcode, errbuf,
        (ub4) sizeof(errbuf), OCI_HTYPE_ERROR);
    }
    return (errcode);
  case OCI_INVALID_HANDLE:
    fprintf(stderr,"Module %s Line %d\n", fname, lineno);
    fprintf(stderr,"Error - OCI_INVALID_HANDLE\n");
    exit(-1);
  case OCI_STILL_EXECUTING:
    fprintf(stderr,"Module %s Line %d\n", fname, lineno);
    fprintf(stderr,"Error - OCI_STILL_EXECUTE\n");
    return (IRRECERR);
  case OCI_CONTINUE:
    fprintf(stderr,"Module %s Line %d\n", fname, lineno);
    fprintf(stderr,"Error - OCI_CONTINUE\n");
    return (IRRECERR);
  default:
    fprintf(stderr,"Module %s Line %d\n", fname, lineno);
    fprintf(stderr,"Status - %s\n", status);
    return (IRRECERR);
  }
  return (RECOVERR);
}

tkvcinin.sql

Rem
Rem $Header: tk_perf/benchmark_kits/tpcc-new/scripts/sql/tkvcinin.sql
Rem/main/2 2008/12/15 05:58:45 avliet Exp $
Rem
Rem tkvcinin.sql
Rem
Rem Copyright (c) 2001, 2008, Oracle and/or its affiliates.
Rem All rights reserved.
Rem
Rem NAME
Rem tkvcinin.sql - <one-line expansion of the name>
Rem
Rem DESCRIPTION
Rem <short description of component this file declares/defines>
Rem
Rem NOTES
Rem <other useful comments, qualifications, etc.>
Rem
Rem MODIFIED (MM/DD/YY)
Rem heri 05/03/02 - Short table names.

CREATE OR REPLACE PACKAGE inittpcc
AS
  TYPE intarray IS TABLE OF INTEGER index by binary_integer;
  TYPE distarray IS TABLE OF VARCHAR(24) index by binary_integer;
  TYPE rowidarray IS TABLE OF ROWID INDEX BY BINARY_INTEGER;
  nulldate DATE;
  s_dist distarray;
  idx1arr intarray;
  s_remote intarray;
  dist intarray;
  row_id rowidarray;
  cust_rowid rowid;
  dist_name varchar2(11);
  ware_name varchar2(11);
  c_num pls_integer;
  PROCEDURE init_no(idxarr intarray);
  PROCEDURE init_del;
  PROCEDURE init_pay;
END inittpcc;
/
show errors;

CREATE OR REPLACE PACKAGE BODY inittpcc AS
PROCEDURE init_no (idxarr intarray)
IS
BEGIN
  -- initialize null date
  nulldate := TO_DATE('09-15-1811', 'MM-DD-YYYY');

  -- we found a savings of ~500 instructions when
  -- initializing the cr_date array on both the client
  -- and the server, instead of initializing on
  -- the client and passing it to the server. this cannot be done
  -- as we require the current system date
  -- cr_date := SYSDATE;

  -- initialize idx1arr on the client and store it here
  -- as a package variable
  idx1arr := idxarr;
END init_no;

PROCEDURE init_del
IS
BEGIN
  FOR i IN 1 .. 10 LOOP
    dist(i) := i;
  END LOOP;
END init_del;

PROCEDURE init_pay IS

```



```

BEGIN
  NULL;
END init_pay;

END inittppcc;
/
show errors

CREATE OR REPLACE PACKAGE tpcc
AS
TYPE intarray IS TABLE OF INTEGER index by binary_integer;
TYPE distarray IS TABLE OF VARCHAR(24) index by binary_integer;
TYPE rowidarray IS TABLE OF ROWID INDEX BY BINARY_INTEGER;
TYPE chararray IS TABLE OF VARCHAR(1) index by binary_integer;
TYPE numarray IS TABLE OF NUMBER index by binary_integer;
TYPE datarray IS TABLE OF DATE INDEX BY BINARY_INTEGER;
nulldate DATE;
s_dist          distarray;
idx1arr         intarray;
s_remote        intarray;
dist            intarray;
row_id          rowidarray;
cust_rowid      rowid;
dist_name       varchar2(11);
ware_name       varchar2(11);
c_num           pls_integer;

PROCEDURE neworder (
  par_w_id BINARY_INTEGER,
  par_d_id BINARY_INTEGER,
  par_c_id BINARY_INTEGER,
  par_o_all_local BINARY_INTEGER,
  par_o_ol_cnt IN OUT BINARY_INTEGER,
  par_w_tax IN OUT BINARY_INTEGER,
  par_d_tax IN OUT BINARY_INTEGER,
  par_o_id IN OUT BINARY_INTEGER,
  par_c_discount IN OUT BINARY_INTEGER,
  par_c_credit IN OUT varchar2,
  par_c_last IN OUT varchar2,
  par_retry IN OUT BINARY_INTEGER,
  par_cr_date DATE,
  par_ol_i_id intarray,
  par_ol_supply_w_id intarray,
  par_i_price IN OUT numarray,
  par_i_name IN OUT distarray,
  par_s_quantity IN OUT intarray,
  par_brand_generic IN OUT chararray,
  par_ol_amount IN OUT intarray,
  par_s_remote intarray,
  par_ol_quantity intarray
);

PROCEDURE orderstatus (
  ware_id          INTEGER,
  dist_id          INTEGER,
  cust_id          IN OUT INTEGER,
  bylastname       INTEGER,
  cust_last        IN OUT VARCHAR2,
  cust_first       OUT VARCHAR2,
  cust_middle      OUT VARCHAR2,
  cust_balance     OUT NUMBER,
  ord_id           IN OUT INTEGER,
  ord_entry_d      OUT VARCHAR2,
  ord_carrier_id   OUT INTEGER,
  ord_ol_cnt       OUT INTEGER,
  oline_supply_w_id IN OUT intarray,
  oline_i_id       IN OUT intarray,
  oline_quantity   IN OUT intarray,
  oline_amount     IN OUT numarray,
  oline_delivery_d OUT datarray

  PROCEDURE delivery (
    ware_id          IN          INTEGER,
    dist_id          IN OUT    intarray,
    order_id         OUT      intarray,
    ordcnt           OUT      INTEGER,
    sums             OUT      intarray,
    del_date         IN          DATE,
    carrier_id       IN          INTEGER,
    order_c_id       OUT      intarray,
    retry            IN OUT    INTEGER
  );

  PROCEDURE payment (
    ware_id          INTEGER,
    dist_id          INTEGER,
    cust_w_id        INTEGER,
    cust_d_id        INTEGER,
    cust_id          IN OUT INTEGER,
    bylastname       INTEGER,
    hist_amount      NUMBER,
    cust_last        IN OUT VARCHAR2,
    ware_street_1    OUT VARCHAR2,
    ware_street_2    OUT VARCHAR2,
    ware_city        OUT VARCHAR2,
    ware_state       OUT VARCHAR2,
    ware_zip         OUT VARCHAR2,
    dist_street_1    OUT VARCHAR2,
    dist_street_2    OUT VARCHAR2,
    dist_city        OUT VARCHAR2,
    dist_state       OUT VARCHAR2,
    dist_zip         OUT VARCHAR2,
    cust_first       OUT VARCHAR2,
    cust_middle      OUT VARCHAR2,
    cust_street_1    OUT VARCHAR2,
    cust_street_2    OUT VARCHAR2,
    cust_city        OUT VARCHAR2,
    cust_state       OUT VARCHAR2,
    cust_zip         OUT VARCHAR2,
    cust_phone       OUT VARCHAR2,
    cust_since       OUT DATE,
    cust_credit      IN OUT VARCHAR2,
    cust_credit_lim  OUT NUMBER,
    cust_discount    OUT NUMBER,
    cust_balance     IN OUT NUMBER,
    cust_data        OUT VARCHAR2,
    cr_date          IN DATE,
    retry            IN OUT INTEGER
  );

  PROCEDURE stocklevel (
    ware_id          INTEGER,
    dist_id          INTEGER,
    threshold        INTEGER,
    low_stock        OUT INTEGER
  );
END tpcc;
/
show errors;

CREATE OR REPLACE PACKAGE BODY tpcc AS
rows_lost        BINARY_INTEGER;
max_index        BINARY_INTEGER;
temp_index       BINARY_INTEGER;

idx              BINARY_INTEGER;
dummy_local      BINARY_INTEGER;
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock         EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);

PROCEDURE neworder (
  par_w_id BINARY_INTEGER,
  par_d_id BINARY_INTEGER,
  par_c_id BINARY_INTEGER,
  par_o_all_local BINARY_INTEGER,
  par_o_ol_cnt IN OUT BINARY_INTEGER,
  par_w_tax IN OUT BINARY_INTEGER,
  par_d_tax IN OUT BINARY_INTEGER,
  par_o_id IN OUT BINARY_INTEGER,
  par_c_discount IN OUT BINARY_INTEGER,
  par_c_credit IN OUT varchar2,
  par_c_last IN OUT varchar2,
  par_retry IN OUT BINARY_INTEGER,
  par_cr_date DATE,
  par_ol_i_id intarray,
  par_ol_supply_w_id intarray,
  par_i_price IN OUT numarray,
  par_i_name IN OUT distarray,
  par_s_quantity IN OUT intarray,
  par_brand_generic IN OUT chararray,
  par_ol_amount IN OUT intarray,
  par_s_remote intarray,
  par_ol_quantity intarray
)
IS
BEGIN
  LOOP BEGIN
    UPDATE dist SET d_next_o_id = d_next_o_id + 1
      WHERE d_id = par_d_id AND d_w_id = par_w_id

```

```

RETURNING d_tax, d_next_o_id-1
INTO par_d_tax, par_o_id;

SELECT c_discount, c_credit, c_last
INTO par_c_discount, par_c_credit, par_c_last
FROM cust
WHERE c_id = par_c_id AND c_d_id = par_d_id AND c_w_id =
par_w_id;

SELECT w_tax
INTO par_w_tax
FROM ware
WHERE w_id = par_w_id;

INSERT INTO nord
VALUES (par_w_id, par_d_id, par_o_id);
INSERT INTO ordr
VALUES (par_o_id, par_w_id, par_d_id, par_c_id, 11,
par_o_ol_cnt, par_o_all_local, par_cr_date);

dummy_local := par_d_id;

CASE dummy_local
----- u1 thru u10 --- BEGIN ---
WHEN 1 THEN
-- ++++++ u1
BEGIN
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_01,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u1
WHEN 2 THEN
-- ++++++ u2
BEGIN
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_02,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u2
WHEN 3 THEN
-- ++++++ u3
BEGIN
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_03,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u3
WHEN 4 THEN
-- ++++++ u4
BEGIN
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_04,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u4
WHEN 5 THEN
-- ++++++ u5
BEGIN
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_05,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpc.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u5
END;
-- ++++++ u5

```

```

WHEN 6 THEN
-- ++++++ u6
BEGIN
--
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_06,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpec.s_dist,
par_ol_amount,par_brand_generic;
END;
-- ++++++ u6
WHEN 7 THEN
-- ++++++ u7
BEGIN
--
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_07,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpec.s_dist,
par_ol_amount,par_brand_generic;
END;
END;
-- ++++++ u7
WHEN 8 THEN
-- ++++++ u8
BEGIN
--
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_08,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpec.s_dist,
par_ol_amount,par_brand_generic;
END;
END;
-- ++++++ u8
WHEN 9 THEN
-- ++++++ u9
BEGIN
--
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_09,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpec.s_dist,
par_ol_amount,par_brand_generic;
END;
END;
-- ++++++ u9
WHEN 10 THEN
-- ++++++ u10
BEGIN
--
--
FORALL idx IN 1 .. par_o_ol_cnt
UPDATE stock_item
SET s_order_cnt = s_order_cnt + 1,
s_ytd = s_ytd + par_ol_quantity(idx),
s_remote_cnt = s_remote_cnt + par_s_remote(idx),
s_quantity = (CASE WHEN s_quantity < par_ol_quantity (idx) + 10
THEN s_quantity +91
ELSE s_quantity
END) - par_ol_quantity(idx)
WHERE i_id = par_ol_i_id(idx)
AND s_w_id = par_ol_supply_w_id(idx)
RETURNING i_price, i_name, s_quantity, s_dist_10,
i_price*par_ol_quantity(idx),
CASE WHEN i_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE (CASE WHEN s_data NOT LIKE '%ORIGINAL%'
THEN 'G'
ELSE 'B'
END)
END
BULK COLLECT INTO par_i_price, par_i_name, par_s_quantity,
inittpec.s_dist,
par_ol_amount,par_brand_generic;
END;
END;
-- ++++++ u10
----- u1 thru u10 --- END ---
ELSE
EXIT;
END CASE;
-- cache the no of rows processed
dummy_local := sql%rowcount;
-- fix the rows if necessary
IF (dummy_local != par_o_ol_cnt ) THEN
-- used to be PROCEDURE fix_items IS
BEGIN
-- gotta shift price, name, s_quantity, brand_generic, s_dist, ol_amount
idx := 1;
-- found 0 bad rows
rows_lost := 0;
-- so many rows in out array to begin with
max_index := sql%rowcount;
WHILE (max_index != par_o_ol_cnt) LOOP
-- find item where item ids dont match
WHILE (idx <= sql%rowcount AND

```

```

        sql%bulk_rowcount(idx + rows_lost) = 1)
LOOP
    idx := idx + 1;
END LOOP;
-- shift the items please
temp_index := max_index;
WHILE (temp_index >= idx + rows_lost) LOOP
    par_i_price(temp_index + 1) := par_i_price(temp_index);
    par_i_name(temp_index + 1) := par_i_name(temp_index);
    par_s_quantity(temp_index + 1) :=
par_s_quantity(temp_index);
    par_ol_amount(temp_index + 1) := par_ol_amount(temp_index);
    inittpc.s_dist(temp_index + 1) := inittpc.s_dist(temp_index);
    par_brand_generic(temp_index + 1) :=
par_brand_generic(temp_index);
    temp_index := temp_index - 1;
END LOOP;
-- values for the non-existent items if not at end
IF (idx + rows_lost <= par_o_ol_cnt) THEN
    par_i_price(idx + rows_lost) := 0;
    par_i_name(idx + rows_lost) := 'NO ITEM';
    par_ol_amount(idx + rows_lost) := 0;
    par_s_quantity(idx + rows_lost) := 0;
    inittpc.s_dist(idx + rows_lost) := NULL;
    par_brand_generic(idx + rows_lost) := '';
-- one more bad row
rows_lost := rows_lost + 1;
max_index := max_index + 1;
END IF;
END LOOP;
END ;
-- end of procedure fix_items;
END IF;
FORALL idx IN 1..par_o_ol_cnt
-- doesnt hurt if we insert entries for invalid item too
INSERT INTO ordl
VALUES (par_w_id, par_d_id, par_o_id, inittpc.idx1arr(idx),
par_ol_i_id(idx),
inittpc.nulldate, par_ol_amount(idx), par_ol_supply_w_id(idx),
par_ol_quantity(idx), inittpc.s_dist(idx));
--If there are no errors, then just return without COMMITING
--The COMMIT is done on the driver side by OCI
-- If there are errors, then rollback and set o_ol_cnt to the processed value
-- note that this is an extra bind ### till we manage to get errors handled
-- properly
IF (dummy_local != par_o_ol_cnt) THEN
    par_o_ol_cnt := dummy_local;
    ROLLBACK;
END IF;
EXIT;
EXCEPTION
    WHEN not_serializable OR deadlock OR snapshot_too_old THEN
        ROLLBACK;
        par_retry := par_retry + 1;
    END;
END LOOP;
END neworder ;
PROCEDURE orderstatus (
    ware_id          INTEGER,
    dist_id          INTEGER,
    cust_id          IN OUT INTEGER,
    bylastname       INTEGER,
    cust_last       IN OUT VARCHAR2,
    cust_first      OUT VARCHAR2,
    cust_middle     OUT VARCHAR2,
    cust_balance    OUT NUMBER,
    ord_id          IN OUT INTEGER,
    ord_entry_d     OUT VARCHAR2,
    ord_carrier_id  OUT INTEGER,
    ord_ol_cnt      OUT INTEGER,
    oline_supply_w_id IN OUT intarray,
    oline_i_id     IN OUT intarray,
    oline_quantity  IN OUT intarray,
    oline_amount   IN OUT numarray,
    oline_delivery_d OUT datarray
)
IS
    cust_rowid      ROWID;
    ol              BINARY_INTEGER;
    c_num           BINARY_INTEGER;
    row_id         rowidarray;
    not_serializable EXCEPTION;
    PRAGMA EXCEPTION_INIT(not_serializable,-8177);
    deadlock       EXCEPTION;
    PRAGMA EXCEPTION_INIT(deadlock,-60);
    snapshot_too_old EXCEPTION;
    PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
    CURSOR o_cur IS
        SELECT ol_i_id, ol_supply_w_id, ol_quantity, ol_amount,
            nvl(ol_delivery_d,to_date('15-09-1911','DD-MM-YYYY'))
    del_date
        FROM ordl
        WHERE ol_d_id = dist_id AND ol_w_id = ware_id AND ol_o_id =
ord_id;
    CURSOR c_cur IS
        SELECT rowid
        FROM cust
        WHERE c_d_id = dist_id AND c_w_id = ware_id AND c_last =
cust_last
        ORDER BY c_w_id, c_d_id, c_last, c_first;
    BEGIN
        LOOP BEGIN
            IF bylastname != 0 THEN
                c_num := 0;
                FOR c_id_rec IN c_cur LOOP
                    c_num := c_num + 1;
                    row_id(c_num) := c_id_rec.rowid;
                END LOOP;
                cust_rowid := row_id((c_num + 1) / 2);
                SELECT c_id, c_balance, c_first, c_middle, c_last
                INTO cust_id, cust_balance, cust_first, cust_middle, cust_last
                FROM cust
                WHERE rowid = cust_rowid;
            ELSE
                SELECT c_balance, c_first, c_middle, c_last
                INTO cust_balance, cust_first, cust_middle, cust_last
                FROM cust
                WHERE c_id = cust_id AND c_d_id = dist_id AND c_w_id =
ware_id;
            END IF;
            -- AVLIET added the rownum=1 clause to select only one ORDER
            -- according to TPCC-spec (2.6.2.2) largest order_id must be selected
            SELECT o_id,
                to_char(o_entry_d, 'DD-MM-YYYY.HH24:MI:SS'),
                nvl(o_carrier_id,0), o_ol_cnt
            INTO ord_id,
                ord_entry_d,
                ord_carrier_id, ord_ol_cnt
            FROM ordr
            WHERE o_d_id = dist_id AND o_w_id = ware_id AND o_c_id =
cust_id
                AND rownum = 1
            ORDER BY o_w_id, o_d_id, o_c_id, o_id DESC;
            ol := 0;
            FOR o_cur_rec IN o_cur LOOP
                ol := ol + 1;
                oline_i_id(ol) := o_cur_rec.ol_i_id;
                oline_supply_w_id(ol) := o_cur_rec.ol_supply_w_id;
                oline_quantity(ol) := o_cur_rec.ol_quantity;
                oline_amount(ol) := o_cur_rec.ol_amount;
                oline_delivery_d(ol) := o_cur_rec.del_date;
            END LOOP;
            COMMIT;
            EXIT;
            EXCEPTION
                WHEN not_serializable OR deadlock OR snapshot_too_old THEN
                    ROLLBACK;
                END;
            END LOOP;
        END orderstatus;

```

```

PROCEDURE delivery (
  ware_id      IN      INTEGER,
  dist_id      IN OUT intarray,
  order_id     OUT    intarray,
  ordcnt       OUT    INTEGER,
  sums         OUT    intarray,
  del_date     IN      DATE,
  carrier_id   IN      INTEGER,
  order_c_id   OUT    intarray,
  retry        IN OUT  INTEGER
) IS
BEGIN
LOOP BEGIN
FORALL d IN 1..10
DELETE /* index_asc (nord inord) */ FROM nord N
WHERE no_d_id = inittpcc.dist(d)
AND no_w_id = ware_id
AND no_o_id = (select min (no_o_id)
from nord
where no_d_id = N.no_d_id
and no_w_id = N.no_w_id)
RETURNING no_d_id, no_o_id BULK COLLECT INTO dist_id,
order_id;

ordcnt := SQL%ROWCOUNT;

FORALL o in 1.. ordcnt
UPDATE ordr SET o_carrier_id = carrier_id
WHERE o_id = order_id(o)
AND o_d_id = dist_id(o)
AND o_w_id = ware_id
RETURNING o_c_id BULK COLLECT INTO order_c_id;

FORALL o in 1.. ordcnt
UPDATE ordl SET ol_delivery_d = del_date
WHERE ol_w_id = ware_id
AND ol_d_id = dist_id(o)
AND ol_o_id = order_id(o)
RETURNING sum(ol_amount) BULK COLLECT INTO sums;

FORALL c IN 1.. ordcnt
UPDATE cust
SET c_balance = c_balance + sums(c),
c_delivery_cnt = c_delivery_cnt + 1
WHERE c_w_id = ware_id
AND c_d_id = dist_id(c)
AND c_id = order_c_id(c);

COMMIT;
EXIT;
EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old
THEN
ROLLBACK;
retry := retry + 1;
END;

END LOOP; -- for retry
END delivery;

PROCEDURE payment (
  ware_id      INTEGER,
  dist_id      INTEGER,
  cust_w_id    INTEGER,
  cust_d_id    INTEGER,
  cust_id      IN OUT INTEGER,
  bylastname   INTEGER,
  hist_amount  NUMBER,
  cust_last    IN OUT VARCHAR2,
  ware_street_1 OUT VARCHAR2,
  ware_street_2 OUT VARCHAR2,
  ware_city    OUT VARCHAR2,
  ware_state   OUT VARCHAR2,
  ware_zip     OUT VARCHAR2,
  dist_street_1 OUT VARCHAR2,
  dist_street_2 OUT VARCHAR2,
  dist_city    OUT VARCHAR2,
  dist_state   OUT VARCHAR2,
  dist_zip     OUT VARCHAR2,
  cust_first   OUT VARCHAR2,
  cust_middle  OUT VARCHAR2,
  cust_street_1 OUT VARCHAR2,
  cust_street_2 OUT VARCHAR2,
  cust_city    OUT VARCHAR2,
  cust_state   OUT VARCHAR2,
  cust_zip     OUT VARCHAR2,
  cust_phone   OUT VARCHAR2,
  cust_since   OUT DATE,
  cust_credit  IN OUT VARCHAR2,
  cust_credit_lim OUT NUMBER,
  cust_discount OUT NUMBER,
  cust_balance IN OUT NUMBER,
  cust_data    OUT VARCHAR2,
  cr_date     IN DATE,
  retry       IN OUT INTEGER
)
IS
TYPE rowidarray IS TABLE OF ROWID INDEX BY
BINARY_INTEGER;
cust_rowid      ROWID;
dist_name       VARCHAR2(11);
ware_name       VARCHAR2(11);
c_num           BINARY_INTEGER;
row_id          rowidarray;
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock        EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
CURSOR c_cur IS
SELECT rowid
FROM cust
WHERE c_d_id = cust_d_id AND c_w_id = cust_w_id AND c_last =
cust_last
ORDER BY c_w_id, c_d_id, c_last, c_first;
BEGIN
LOOP BEGIN
IF bylastname != 0 THEN
c_num := 0;
FOR c_id_rec IN c_cur LOOP
c_num := c_num + 1;
row_id(c_num) := c_id_rec.rowid;
END LOOP;
cust_rowid := row_id ((c_num + 1) / 2);

UPDATE cust
SET c_balance = c_balance - hist_amount,
c_ytd_payment = c_ytd_payment + hist_amount,
c_payment_cnt = c_payment_cnt + 1
WHERE rowid = cust_rowid
RETURNING c_id, c_first, c_middle, c_last, c_street_1, c_street_2,
c_city, c_state, c_zip, c_phone,
c_since, c_credit, c_credit_lim,
c_discount, c_balance
INTO cust_id, cust_first, cust_middle, cust_last, cust_street_1,
cust_street_2, cust_city, cust_state, cust_zip, cust_phone,
cust_since, cust_credit, cust_credit_lim, cust_discount,
cust_balance;

ELSE

UPDATE cust
SET c_balance = c_balance - hist_amount,
c_ytd_payment = c_ytd_payment + hist_amount,
c_payment_cnt = c_payment_cnt + 1
WHERE c_id = cust_id AND c_d_id = cust_d_id AND
c_w_id = cust_w_id
RETURNING rowid, c_first, c_middle, c_last, c_street_1, c_street_2,
c_city, c_state, c_zip, c_phone,
c_since, c_credit, c_credit_lim,
c_discount, c_balance
INTO cust_rowid, cust_first, cust_middle, cust_last,
cust_street_1, cust_street_2, cust_city, cust_state,
cust_zip, cust_phone, cust_since, cust_credit,
cust_credit_lim, cust_discount, cust_balance;

END IF;

IF cust_credit = 'BC' THEN
UPDATE cust
SET c_data = substr ((to_char (cust_id) || '' ||
to_char (cust_d_id) || '' ||
to_char (cust_w_id) || '' ||
to_char (dist_id) || '' ||
to_char (ware_id) || '' ||
to_char (hist_amount, '9999.99') || '' ||
|| c_data, 1, 500)
WHERE rowid = cust_rowid
RETURNING substr (c_data, 1, 200)
INTO cust_data;
ELSE

```

```

cust_data := '';
END IF;

UPDATE dist
SET d_ytd = d_ytd + hist_amount
WHERE d_id = dist_id
AND d_w_id = ware_id
RETURNING d_name, d_street_1, d_street_2, d_city, d_state, d_zip
INTO dist_name, dist_street_1, dist_street_2, dist_city,
dist_state, dist_zip;

UPDATE ware
SET w_ytd = w_ytd + hist_amount
WHERE w_id = ware_id
RETURNING w_name, w_street_1, w_street_2, w_city, w_state, w_zip
INTO ware_name, ware_street_1, ware_street_2, ware_city,
ware_state, ware_zip;

INSERT INTO hist
(h_c_id, h_c_d_id, h_c_w_id, h_d_id, h_w_id, h_date,
h_amount, h_data)
VALUES
(cust_id, cust_d_id, cust_w_id, dist_id, ware_id, cr_date,
hist_amount, ware_name || ' ' || dist_name);

COMMIT;
EXIT;

EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old THEN
ROLLBACK;
retry := retry + 1;
END;

END LOOP;
END payment;

PROCEDURE stocklevel (
ware_id INTEGER,
dist_id INTEGER,
threshold INTEGER,
low_stock OUT INTEGER
)
IS
not_serializable EXCEPTION;
PRAGMA EXCEPTION_INIT(not_serializable,-8177);
deadlock EXCEPTION;
PRAGMA EXCEPTION_INIT(deadlock,-60);
snapshot_too_old EXCEPTION;
PRAGMA EXCEPTION_INIT(snapshot_too_old,-1555);
BEGIN

LOOP BEGIN

SELECT count (DISTINCT s_i_id)
INTO low_stock

```

```

FROM ordl, stok, dist
WHERE d_id = dist_id AND d_w_id = ware_id AND
d_id = ol_d_id AND d_w_id = ol_w_id AND
ol_i_id = s_i_id AND ol_w_id = s_w_id AND
s_quantity < threshold AND
ol_o_id BETWEEN (d_next_o_id - 20) AND (d_next_o_id - 1);
COMMIT;
EXIT;

EXCEPTION
WHEN not_serializable OR deadlock OR snapshot_too_old THEN
ROLLBACK;
END;
END LOOP;
END stocklevel;

END tpcc;
/
show errors

```

---

### *RTE Parameters*

---

#### **config**

```

setDriverParams :think_neworder => 12.04
setDriverParams :think_payment => 12.04
setDriverParams :think_orderstatus => 10.04
setDriverParams :think_delivery => 5.04
setDriverParams :think_stocklevel => 5.04

set :mix_payment => 43.01
set :mix_orderstatus => 4.01
set :mix_delivery => 4.01
set :mix_stocklevel => 4.01

set :warehouses => 380000
set :ramptime => 30*60
set :runtime => 3*60*60

```

## Appendix C: Parameter Settings

### Oracle Data-Base Parameters

#### p\_run.ora

```
job_queue_processes = 0
aq_tm_processes = 0
audit_trail = FALSE
compatible = 11.2.0.0.1
control_files = (/export/home/oracle/tpcc_disks/control_001,
/export/home/oracle/tpcc_disks/control_002)
db_block_size = 4096
```

```
db_16k_cache_size = 990G
db_8k_cache_size = 35G
db_cache_size = 100G
db_keep_cache_size = 2240G
db_recycle_cache_size = 490G
shared_pool_size = 80G
db_cache_advice = off
```

```
timed_statistics = FALSE
db_block_checksum = FALSE
db_block_checking = FALSE
db_files = 2100
db_name = tpcc
disk_asynch_io = TRUE
filesystemio_options = setall
db_writer_processes = 32
local_listener = all_listeners
log_buffer = 268435456
log_checkpoint_timeout = 0
#log_checkpoint_timeout = 1740
log_checkpoint_timeout = 0
log_checkpoints_to_alert = TRUE
java_pool_size = 0
plsql_optimize_level = 2
parallel_max_servers = 0
pga_aggregate_target = 0
fast_start_mttr_target = 0
statistics_level = basic
result_cache_max_size = 0
dml_locks = 2000
transactions = 1500
undo_management = auto
undo_retention = 1
undo_tablespace = undo_1
processes = 2000
recovery_parallelism = 40
sessions = 4960
```

#### listener.ora

```
tpcc_lsn=
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP)(HOST=x4800-002)(PORT=1521))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn1 =
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.100.100)
(PORT=7521))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn2 =
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.100.100)
(PORT=7522))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn3 =
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.100.100)
(PORT=7523))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn4 =
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.100.100)
(PORT=7524))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn5 =
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.101.100)
(PORT=7521))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn6 =
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.101.100)
(PORT=7522))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn7 =
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.101.100)
(PORT=7523))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn8 =
(ADDRESS_LIST=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.101.100)
(PORT=7524))
(CONNECT_DATA=(SID=tpcc)))
```

#### tnsnames.ora

```
all_listeners =
(DESCRIPTION =
(ADDRESS=(PROTOCOL=TCP)(Host=192.168.100.100)(PORT=7521))
(ADDRESS=(PROTOCOL=TCP)(Host=192.168.100.100)(PORT=7522))
(ADDRESS=(PROTOCOL=TCP)(Host=192.168.100.100)(PORT=7523))
(ADDRESS=(PROTOCOL=TCP)(Host=192.168.100.100)(PORT=7524))
(ADDRESS=(PROTOCOL=TCP)(Host=192.168.101.100)(PORT=7521))
(ADDRESS=(PROTOCOL=TCP)(Host=192.168.101.100)(PORT=7522))
(ADDRESS=(PROTOCOL=TCP)(Host=192.168.101.100)(PORT=7523))
(ADDRESS=(PROTOCOL=TCP)(Host=192.168.101.100)(PORT=7524)))
```

```
)
tpcc_lsn =
(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(HOST=x4800-002)(PORT=1521))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn1 =
(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.100.100)
(PORT=7521))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn2 =
(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.100.100)
(PORT=7522))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn3 =
(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.100.100)
(PORT=7523))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn4 =
(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.100.100)
(PORT=7524))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn5 =
(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.101.100)
(PORT=7521))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn6 =
(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.101.100)
(PORT=7522))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn7 =
(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.101.100)
(PORT=7523))
(CONNECT_DATA=(SID=tpcc)))
```

```
tpcc_lsn8 =
(DESCRIPTION=
(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.101.100)
(PORT=7524))
(CONNECT_DATA=(SID=tpcc)))
```

### Sun Fire X4800 M2 Parameters

**version**

Linux version 2.6.39-100.0.5.el6uek (root@ca-test111.us.oracle.com) (gcc version 4.4.5 20110214 (Red Hat 4.4.5-6) (GCC) ) #1 SMP Mon Dec 12 12:33:34 PST 2011

**cmdline**

kernel /vmlinuz-2.6.39-100.0.5.el6uek ro root=UUID=51db5970-3835-495d-9034-08f4426e5cdb rd\_NO\_LUKS rd\_NO\_LVM rd\_NO\_MD rd\_NO\_DM LANG=en\_US.UTF-8 SYSFONT=latarcyrheb-sun16 KEYBOARDTYPE=pc KEYTABLE=us nosoftlockup console=tty0 loglevel=3 console=ttyS0,9600 earlyprintk=ttyS0,9600 rcutree.rcu\_cpu\_stall\_suppress=1 elevator=noop idle=mwait selinux=0 nohz=off cgroup\_disable=memory

**sysctl.conf**

```
# Kernel sysctl configuration file for Red Hat Linux
#
# For binary values, 0 is disabled, 1 is enabled. See sysctl(8) and
# sysctl.conf(5) for more details.

# Controls IP packet forwarding
net.ipv4.ip_forward = 0

# Controls source route verification
net.ipv4.conf.default.rp_filter = 1

#oracle wants this:
net.ipv4.ip_local_port_range = 9000 65500
# Do not accept source routing
net.ipv4.conf.default.accept_source_route = 0

# Controls the System Request debugging functionality of the kernel
kernel.sysrq = 0

# Controls whether core dumps will append the PID to the core filename.
# Useful for debugging multi-threaded applications.
kernel.core_uses_pid = 1

# Controls the use of TCP syncookies
net.ipv4.tcp_syncookies = 1

# Disable netfilter on bridges.
net.bridge.bridge-nf-call-ip6tables = 0
net.bridge.bridge-nf-call-iptables = 0
net.bridge.bridge-nf-call-arptables = 0

kernel.msgmnb = 65536
kernel.msgmax = 65536
# semaphores: semmsl, semmns, semopm, semmni
kernel.sem = 250 32000 100 128
fs.file-max = 6815744
fs.aio-max-nr = 1048576
net.core.rmem_default = 262144
net.core.rmem_max = 4194304
net.core.wmem_default = 262144
net.core.wmem_max = 1048586
```

fs.inotify.max\_user\_watches = 100000

kernel.shmall=4398046511104  
kernel.shmmax=1099511627776  
vm.nr\_hugepages=2024640

kernel.sched\_rt\_runtime\_us=-1  
vm.zone\_reclaim\_mode=1  
vm.numa\_zonelist\_order=none  
kernel.sched\_latency\_ns=1000000000  
kernel.sched\_min\_granularity\_ns=5000000

**meminfo**

```
MemTotal: 4236142204 kB
MemFree: 81399968 kB
Buffers: 104924 kB
Cached: 1104252 kB
SwapCached: 0 kB
Active: 798140 kB
Inactive: 229008 kB
Active(anon): 420456 kB
Inactive(anon): 25324 kB
Active(file): 377684 kB
Inactive(file): 203684 kB
Unevictable: 0 kB
Mlocked: 0 kB
SwapTotal: 35656928 kB
SwapFree: 35656928 kB
Dirty: 20 kB
Writeback: 0 kB
AnonPages: 391804 kB
Mapped: 23768 kB
Shmem: 54212 kB
Slab: 644500 kB
SReclaimable: 161828 kB
SUnreclaim: 482672 kB
KernelStack: 12152 kB
PageTables: 14788 kB
NFS_Unstable: 0 kB
Bounce: 0 kB
WritebackTmp: 0 kB
CommitLimit: 80496668 kB
Committed_AS: 2095408 kB
VmallocTotal: 34359738367 kB
VmallocUsed: 6785512 kB
VmallocChunk: 30594921156 kB
HardwareCorrupted: 0 kB
AnonHugePages: 47104 kB
HugePages_Total: 2024640
HugePages_Free: 1881282
HugePages_Rsvd: 1775441
HugePages_Surp: 0
Hugepagesize: 2048 kB
DirectMap4k: 7488 kB
DirectMap2M: 2080768 kB
```

DirectMap1G: 4292870144 kB

**cpuinfo**

```
processor : 0
vendor_id : GenuineIntel
cpu family : 6
model : 47
model name : Intel(R) Xeon(R) CPU E7- 8870 @ 2.40GHz
stepping : 2
cpu MHz : 2401.000
cache size : 30720 KB
physical id : 0
siblings : 20
core id : 0
cpu cores : 10
apicid : 0
initial apicid : 0
fpu : yes
fpu_exception : yes
cpuid level: 11
wp : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge
mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall
nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl
xtopology nonstop_tsc aperfmperf pni pclmulqdq dtes64 monitor ds_cpl vmx
smx est tm2 sse3 cx16 xtpr pdcm dca sse4_1 sse4_2 popcnt aes lahf_lm ida
arat epb dts tpr_shadow vnmi flexpriority ept vpid
bogomips : 4787.69
clflush size : 64
cache_alignment : 64
address sizes : 44 bits physical, 48 bits virtual
power management:
```

**[ 1-158 identical]**

```
processor : 159
vendor_id : GenuineIntel
cpu family : 6
model : 47
model name : Intel(R) Xeon(R) CPU E7- 8870 @ 2.40GHz
stepping : 2
cpu MHz : 2401.000
cache size : 30720 KB
physical id : 7
siblings : 20
core id : 9
cpu cores : 10
apicid : 243
initial apicid : 243
fpu : yes
fpu_exception : yes
cpuid level: 11
wp : yes
flags : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge
mca cmov pat pse36 clflush dts acpi mmx fxsr sse sse2 ss ht tm pbe syscall
nx pdpe1gb rdtscp lm constant_tsc arch_perfmon pebs bts rep_good nopl
```



xtopology nonstop\_tsc aperfmperf pni pelmulq dtes64 monitor ds\_cpl vmx  
 smx est tm2 sse3 cx16 xtrp pdcm dea sse4\_1 sse4\_2 popcnt aes lahf\_lm ida  
 arat epb dts tpr\_shadow vnmi flexpriority ept vpid  
 bogomips : 4787.85  
 clflush size : 64  
 cache\_alignment : 64  
 address sizes : 44 bits physical, 48 bits virtual  
 power management:

**fdisk**

Disk /dev/sdc: 299.4 GB, 299439751168 bytes  
 255 heads, 63 sectors/track, 36404 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x0000ceb6b

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Disk /dev/sda: 299.4 GB, 299439751168 bytes  
 255 heads, 63 sectors/track, 36404 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x0009fd21

Device	Boot	Start	End	Blocks	Id	System
/dev/sda1	*	1	131	1048576	83	Linux
Partition 1 does not end on cylinder boundary.						
/dev/sda2		131	392	2097152	82	Linux swap / Solaris
Partition 2 does not end on cylinder boundary.						
/dev/sda3		392	36405	289274880	83	Linux

Disk /dev/sde: 299.4 GB, 299439751168 bytes  
 255 heads, 63 sectors/track, 36404 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00052793

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Disk /dev/sdd: 299.4 GB, 299439751168 bytes  
 255 heads, 63 sectors/track, 36404 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x000a4d3c

Device	Boot	Start	End	Blocks	Id	System
/dev/sdd1		1	17755	142617006	83	Linux
/dev/sdd2		17756	21933	33559785	82	Linux swap / Solaris
/dev/sdd3		21934	34988	104864287+	83	Linux

Disk /dev/sdf: 299.4 GB, 299439751168 bytes  
 255 heads, 63 sectors/track, 36404 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00029bca

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Disk /dev/sdg: 299.4 GB, 299439751168 bytes  
 255 heads, 63 sectors/track, 36404 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x0008944a

Device	Boot	Start	End	Blocks	Id	System
/dev/sdg1	*	2	36404	292407097+	bf	Solaris

Disk /dev/sdh: 299.4 GB, 299439751168 bytes  
 255 heads, 63 sectors/track, 36404 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00003569

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Disk /dev/sdb: 299.4 GB, 299439751168 bytes  
 255 heads, 63 sectors/track, 36404 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x26f2964d

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	36404	292415098+	83	Linux

Disk /dev/sdi: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdi1		1	1913	15364031+	ee	GPT

Disk /dev/sdn: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdn1		1	1913	15364031+	ee	GPT

Disk /dev/sdt: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdt1		1	1913	15364031+	ee	GPT

Disk /dev/sdw: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdw1		1	1913	15364031+	ee	GPT

Disk /dev/sdl: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdl1		1	1913	15364031+	ee	GPT

Disk /dev/sdj: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdj1		1	1913	15364031+	ee	GPT

Disk /dev/sdo: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdo1		1	1913	15364031+	ee	GPT

Disk /dev/sdm: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdm1		1	1913	15364031+	ee	GPT

Disk /dev/sdy: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdy1	1	1913	15364031+	ee	GPT	

Disk /dev/sdaa: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaa1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sds: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sds1	1	1913	15364031+	ee	GPT	

Disk /dev/sdv: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdv1	1	1913	15364031+	ee	GPT	

Disk /dev/sdac: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdac1	1	1403	11268031+	ee	GPT	

Disk /dev/sdx: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdx1	1	1913	15364031+	ee	GPT	

Disk /dev/sdab: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdab1	1	1913	15364031+	ee	GPT	

Disk /dev/sdz: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdz1	1	1913	15364031+	ee	GPT	

Disk /dev/sdq: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdq1	1	1913	15364031+	ee	GPT	

Disk /dev/sdk: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdk1	1	1913	15364031+	ee	GPT	

Disk /dev/sdp: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdp1	1	1913	15364031+	ee	GPT	

Disk /dev/sdae: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdae1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdr: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdr1	1	1913	15364031+	ee	GPT	

Disk /dev/sdad: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdad1	1	1913	15364031+	ee	GPT	

Disk /dev/sdu: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdu1	1	1913	15364031+	ee	GPT	

Disk /dev/sdaf: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaf1	1	1977	15876031+	ee	GPT	

Disk /dev/sdal: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
 /dev/sdall 1 944 7581631+ ee GPT  
 Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdar: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdar1	1	2678	21508031+	ee	GPT	

Disk /dev/sdax: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdax1	1	7522	60420031+	ee	GPT	

Disk /dev/sdbg: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbg1	1	2002	16080831+	ee	GPT	

Disk /dev/sdah: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdah1	1	2678	21508031+	ee	GPT	

Disk /dev/sdbt: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbt1	1	1913	15364031+	ee	GPT	

Disk /dev/sdbr: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbr1	1	2002	16080831+	ee	GPT	

Disk /dev/sdai: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdai1	1	1977	15876031+	ee	GPT	

Disk /dev/sdby: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdby1	1	1913	15364031+	ee	GPT	

Disk /dev/sdbv: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbv1	1	2002	16080831+	ee	GPT	

Disk /dev/sdbz: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbz1	1	1913	15364031+	ee	GPT	

Disk /dev/sdca: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

Device	Boot	Start	End	Blocks	Id	System
/dev/sdca1	1	1913	15364031+	ee	GPT	

Disk /dev/sdbm: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbm1	1	2002	16080831+	ee	GPT	

Disk /dev/sdce: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdce1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdbu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbu1	1	2002	16080831+	ee	GPT	

Disk /dev/sdbw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbw1	1	2002	16080831+	ee	GPT	

Disk /dev/sdcb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcb1	1	1913	15364031+	ee	GPT	

Disk /dev/sdag: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdag1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbe: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbe1		1	2002	16080831+	ee	GPT

Disk /dev/sdbx: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbx1		1	1913	15364031+	ee	GPT

Disk /dev/sdcd: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcd1		1	1913	15364031+	ee	GPT

Disk /dev/sdbc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbc1		1	2002	16080831+	ee	GPT

Disk /dev/sdau: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdau1		1	2002	16080831+	ee	GPT

Disk /dev/sdak: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdak1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdba: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdba1		1	2002	16080831+	ee	GPT

Disk /dev/sdcf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcf1		1	1913	15364031+	ee	GPT

Disk /dev/sdao: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdao1		1	7522	60420031+	ee	GPT

Disk /dev/sdaj: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaj1		1	2678	21508031+	ee	GPT

Disk /dev/sdat: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdat1		1	2002	16080831+	ee	GPT

Disk /dev/sdas: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdas1		1	7522	60420031+	ee	GPT

Disk /dev/sdaw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaw1		1	2002	16080831+	ee	GPT

Disk /dev/sday: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sday1		1	2002	16080831+	ee	GPT

Disk /dev/sdam: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdam1		1	7522	60420031+	ee	GPT

Disk /dev/sdav: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System



Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbk1		1	2002	16080831+	ee	GPT

Disk /dev/sdcl: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcl1		1	1913	15364031+	ee	GPT

Disk /dev/sdch: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdch1		1	1913	15364031+	ee	GPT

Disk /dev/sdci: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdci1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdcj: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcj1		1	1913	15364031+	ee	GPT

Disk /dev/sdcl: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcl1		1	7522	60420031+	ee	GPT

Disk /dev/sdco: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdco1		1	1913	15364031+	ee	GPT

Disk /dev/sdcn: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcn1		1	1403	11268031+	ee	GPT

Disk /dev/sdcq: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcq1		1	7522	60420031+	ee	GPT

Disk /dev/sdct: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdct1		1	1977	15876031+	ee	GPT

Disk /dev/sdcs: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcs1		1	1913	15364031+	ee	GPT

Disk /dev/sdcy: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcy1		1	2002	16080831+	ee	GPT

Disk /dev/sdcx: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcx1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdcu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcu1		1	2002	16080831+	ee	GPT

Disk /dev/sdda: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdda1		1	2002	16080831+	ee	GPT

Disk /dev/sdde: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdde1		1	2678	21508031+	ee	GPT

Disk /dev/sddb: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System



/dev/sddx1 1 2002 16080831+ ee GPT

Disk /dev/sddy: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sddy1	1	1977	15876031+	ee	GPT	

Disk /dev/sddc: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sddc1	1	2002	16080831+	ee	GPT	

Disk /dev/sdcr: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcr1	1	2678	21508031+	ee	GPT	

Disk /dev/sddg: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sddg1	1	2002	16080831+	ee	GPT	

Disk /dev/sdcw: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcw1	1	2002	16080831+	ee	GPT	

Disk /dev/sdcm: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcm1	1	1913	15364031+	ee	GPT	

Disk /dev/sdcv: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcv1	1	7522	60420031+	ee	GPT	

Disk /dev/sddj: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sddj1	1	2002	16080831+	ee	GPT	

Disk /dev/sdeb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdeb1	1	2002	16080831+	ee	GPT	

Disk /dev/sdcz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcz1	1	2002	16080831+	ee	GPT	

Disk /dev/sdck: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdck1	1	1913	15364031+	ee	GPT	

Disk /dev/sddz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sddz1	1	1913	15364031+	ee	GPT	

Disk /dev/sdec: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdec1	1	2678	21508031+	ee	GPT	

Disk /dev/sdcp: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcp1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sddm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sddm1	1	2002	16080831+	ee	GPT	

Disk /dev/sdea: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdea1	1	2002	16080831+	ee	GPT	

Disk /dev/sded: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------



/dev/sded1 1 1913 15364031+ ee GPT

Disk /dev/sdee: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdee1	1	7522	60420031+	ee	GPT	

Disk /dev/sdef: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdef1	1	2002	16080831+	ee	GPT	

Disk /dev/sdeh: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdeh1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdeg: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdeg1	1	2002	16080831+	ee	GPT	

Disk /dev/sdek: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdek1	1	2678	21508031+	ee	GPT	

Disk /dev/sdej: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdej1	1	2002	16080831+	ee	GPT	

Disk /dev/sdei: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdei1	1	2002	16080831+	ee	GPT	

Disk /dev/sdel: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdel1	1	2002	16080831+	ee	GPT	

Disk /dev/sdeu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdeu1	1	1913	15364031+	ee	GPT	

Disk /dev/sdeq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdeq1	1	1913	15364031+	ee	GPT	

Disk /dev/sder: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sder1	1	2002	16080831+	ee	GPT	

Disk /dev/sdew: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdew1	1	1913	15364031+	ee	GPT	

Disk /dev/sdep: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdep1	1	2002	16080831+	ee	GPT	

Disk /dev/sdez: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdez1	1	1913	15364031+	ee	GPT	

Disk /dev/sden: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sden1	1	7522	60420031+	ee	GPT	

Disk /dev/sdeo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdeo1	1	2002	16080831+	ee	GPT	

Disk /dev/sdfc: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

/dev/sdfe1 1 1913 15364031+ ee GPT

Disk /dev/sdes: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdes1	1	2002	16080831+	ee	GPT	

Disk /dev/sdfg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfg1	1	1913	15364031+	ee	GPT	

Disk /dev/sdem: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdem1	1	2002	16080831+	ee	GPT	

Disk /dev/sdfa: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfa1	1	2002	16080831+	ee	GPT	

Disk /dev/sdfe: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfe1	1	1913	15364031+	ee	GPT	

Disk /dev/sdex: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdex1	1	2002	16080831+	ee	GPT	

Disk /dev/sdfd: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfd1	1	1913	15364031+	ee	GPT	

Disk /dev/sdet: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdet1	1	2002	16080831+	ee	GPT	

Disk /dev/sdey: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdey1	1	1913	15364031+	ee	GPT	

Disk /dev/sdff: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdff1	1	2002	16080831+	ee	GPT	

Disk /dev/sdfb: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfb1	1	1913	15364031+	ee	GPT	

Disk /dev/sdev: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdev1	1	2002	16080831+	ee	GPT	

Disk /dev/sdfi: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfi1	1	1913	15364031+	ee	GPT	

Disk /dev/sdfh: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfh1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdfj: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfj1	1	1913	15364031+	ee	GPT	

Disk /dev/sdfk: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfk1	1	1913	15364031+	ee	GPT	

Disk /dev/sdfn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

/dev/sdfn1 1 1913 15364031+ ee GPT

Disk /dev/sdfo: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfo1	1	1913	15364031+	ee	GPT	

Disk /dev/sdff: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdff1	1	1977	15876031+	ee	GPT	

Disk /dev/sdfs: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfs1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdfp: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfp1	1	1977	15876031+	ee	GPT	

Disk /dev/sdfm: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfm1	1	1403	11268031+	ee	GPT	

Disk /dev/sdfv: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfv1	1	460	3690431+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdfr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfr1	1	1913	15364031+	ee	GPT	

Disk /dev/sdfy: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfy1	1	2678	21508031+	ee	GPT	

Disk /dev/sdfz: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfz1	1	7522	60420031+	ee	GPT	

Disk /dev/sdgc: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgc1	1	7522	60420031+	ee	GPT	

Disk /dev/sdfw: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfw1	1	2678	21508031+	ee	GPT	

Disk /dev/sdge: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdge1	1	2002	16080831+	ee	GPT	

Disk /dev/sdgg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgg1	1	1913	15364031+	ee	GPT	

Disk /dev/sdgb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgb1	1	2002	16080831+	ee	GPT	

Disk /dev/sdgi: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgi1	1	2002	16080831+	ee	GPT	

Disk /dev/sdgd: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgd1	1	7522	60420031+	ee	GPT	

Disk /dev/sdfq: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfq1		1	7522	60420031+	ee	GPT

Disk /dev/sdgi: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgi1		1	2002	16080831+	ee	GPT

Disk /dev/sdga: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdga1		1	2002	16080831+	ee	GPT

Disk /dev/sdf1: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdf1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdgy: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgy1		1	1913	15364031+	ee	GPT

Disk /dev/sdfu: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfu1		1	7522	60420031+	ee	GPT

Disk /dev/sdgu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgu1		1	1913	15364031+	ee	GPT

Disk /dev/sdhu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhu1		1	1913	15364031+	ee	GPT

Disk /dev/sdhf: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhf1		1	1913	15364031+	ee	GPT

Disk /dev/sdhg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhg1		1	1913	15364031+	ee	GPT

Disk /dev/sdgm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgm1		1	2002	16080831+	ee	GPT

Disk /dev/sdhi: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhi1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdhn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhn1		1	1913	15364031+	ee	GPT

Disk /dev/sdfx: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdfx1		1	2002	16080831+	ee	GPT

Disk /dev/sdgi: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgi1		1	2002	16080831+	ee	GPT

Disk /dev/sdgl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgl1		1	2002	16080831+	ee	GPT

Disk /dev/sdhr: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhr1		1	7522	60420031+	ee	GPT



I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhu1		1	1913	15364031+	ee	GPT

Disk /dev/sdgv: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgv1		1	2002	16080831+	ee	GPT

Disk /dev/sdhd: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhd1		1	1913	15364031+	ee	GPT

Disk /dev/sdhh: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhh1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdho: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdho1		1	1913	15364031+	ee	GPT

Disk /dev/sdgz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdgz1		1	2002	16080831+	ee	GPT

Disk /dev/sdhj: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhj1		1	1913	15364031+	ee	GPT

Disk /dev/sdhl: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhl1		1	2678	21508031+	ee	GPT

Disk /dev/sdhe: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhe1		1	2002	16080831+	ee	GPT

Disk /dev/sdhp: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhp1		1	7522	60420031+	ee	GPT

Disk /dev/sdhv: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhv1		1	2002	16080831+	ee	GPT

Disk /dev/sdhs: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhs1		1	1913	15364031+	ee	GPT

Disk /dev/sdhk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhk1		1	2002	16080831+	ee	GPT

Disk /dev/sdhy: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhy1		1	1913	15364031+	ee	GPT

Disk /dev/sdhx: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhx1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdhz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdhz1		1	2002	16080831+	ee	GPT

Disk /dev/sdia: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdia1		1	2002	16080831+	ee	GPT

Disk /dev/sdic: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdic1		1	1977	15876031+	ee	GPT

Disk /dev/sdig: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdig1		1	2002	16080831+	ee	GPT

Disk /dev/sdih: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdih1		1	2678	21508031+	ee	GPT

Disk /dev/sdif: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdif1		1	1913	15364031+	ee	GPT

Disk /dev/sdik: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdik1		1	2002	16080831+	ee	GPT

Disk /dev/sdim: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdim1		1	1913	15364031+	ee	GPT

Device	Boot	Start	End	Blocks	Id	System
/dev/sdim1		1	1913	15364031+	ee	GPT

Disk /dev/sdiq: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdiq1		1	1403	11268031+	ee	GPT

Disk /dev/sdip: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdip1		1	2002	16080831+	ee	GPT

Disk /dev/sdio: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdio1		1	7522	60420031+	ee	GPT

Disk /dev/sdiu: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdiu1		1	1977	15876031+	ee	GPT

Disk /dev/sdis: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdis1		1	2002	16080831+	ee	GPT

Disk /dev/sdiy: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdiy1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdid: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdid1		1	2002	16080831+	ee	GPT

Disk /dev/sdij: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdij1		1	2002	16080831+	ee	GPT

Disk /dev/sdib: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdib1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdin: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdin1		1	2002	16080831+	ee	GPT

Disk /dev/sdit: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000





Disk /dev/sdje: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdje1	1	2002	16080831+	ee	GPT	

Disk /dev/sdjo: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjo1	1	2002	16080831+	ee	GPT	

Disk /dev/sdjj: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjj1	1	1913	15364031+	ee	GPT	

Disk /dev/sdjj: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjj1	1	1913	15364031+	ee	GPT	

Disk /dev/sdjm: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjm1	1	2002	16080831+	ee	GPT	

Disk /dev/sdjs: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjs1	1	1913	15364031+	ee	GPT	

/dev/sdjs1 1 1913 15364031+ ee GPT

Disk /dev/sdjn: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjn1	1	1913	15364031+	ee	GPT	

Disk /dev/sdjp: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjp1	1	1913	15364031+	ee	GPT	

Disk /dev/sdjq: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjq1	1	1913	15364031+	ee	GPT	

Disk /dev/sdjr: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjr1	1	2002	16080831+	ee	GPT	

Disk /dev/sdju: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdju1	1	1913	15364031+	ee	GPT	

Disk /dev/sdjt: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
 /dev/sdjt1 1 944 7581631+ ee GPT  
 Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdjk: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjk1	1	2678	21508031+	ee	GPT	

Disk /dev/sdjl: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjl1	1	1913	15364031+	ee	GPT	

Disk /dev/sdjm: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjm1	1	1913	15364031+	ee	GPT	

Disk /dev/sdjn: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjn1	1	1913	15364031+	ee	GPT	

Disk /dev/sdkd: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkd1	1	1913	15364031+	ee	GPT	

Disk /dev/sdkb: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkb1		1	1913	15364031+	ee	GPT

Disk /dev/sdkf: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkf1		1	1913	15364031+	ee	GPT

Disk /dev/sdjj: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdjj1		1	7522	60420031+	ee	GPT

Disk /dev/sdka: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdka1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdkh: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkh1		1	1913	15364031+	ee	GPT

Disk /dev/sdki: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdki1		1	1913	15364031+	ee	GPT

Disk /dev/sdke: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdke1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdkj: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkj1		1	7522	60420031+	ee	GPT

Disk /dev/sdkk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkk1		1	2002	16080831+	ee	GPT

Disk /dev/sdko: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdko1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdkg: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkg1		1	2002	16080831+	ee	GPT

Disk /dev/sdkl: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkl1		1	1913	15364031+	ee	GPT

Disk /dev/sdkn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkn1		1	2002	16080831+	ee	GPT

Disk /dev/sdkc: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdkc1		1	2002	16080831+	ee	GPT

Disk /dev/sdku: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdku1		1	2678	21508031+	ee	GPT

Disk /dev/sdks: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdks1		1	1913	15364031+	ee	GPT

Disk /dev/sdkp: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders



Device	Boot	Start	End	Blocks	Id	System
/dev/sdlk1		1	2002	16080831+	ee	GPT

Disk /dev/sdll: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdll1		1	1977	15876031+	ee	GPT

Disk /dev/sdln: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdln1		1	2002	16080831+	ee	GPT

Disk /dev/sdlf: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlf1		1	2002	16080831+	ee	GPT

Disk /dev/sdlc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlc1		1	2002	16080831+	ee	GPT

Disk /dev/sdlm: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlm1		1	2002	16080831+	ee	GPT

Disk /dev/sdlo: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlo1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdli: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdli1		1	2002	16080831+	ee	GPT

Disk /dev/sdlp: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlp1		1	1913	15364031+	ee	GPT

Disk /dev/sdlq: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlq1		1	2002	16080831+	ee	GPT

Disk /dev/sdlr: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlr1		1	2002	16080831+	ee	GPT

Disk /dev/sdls: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

Device	Boot	Start	End	Blocks	Id	System
/dev/sdls1		1	1913	15364031+	ee	GPT

Disk /dev/sdlv: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlv1		1	1913	15364031+	ee	GPT

Disk /dev/sdlw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlw1		1	2002	16080831+	ee	GPT

Disk /dev/sdlu: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlu1		1	1913	15364031+	ee	GPT

Disk /dev/sdly: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdly1		1	2002	16080831+	ee	GPT

Disk /dev/sdlx: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlx1		1	1913	15364031+	ee	GPT

Disk /dev/sdlt: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdl1	1	7522	60420031+	ee	GPT	

Disk /dev/sdlz: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdlz1	1	1913	15364031+	ee	GPT	

Disk /dev/sdma: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdma1	1	2002	16080831+	ee	GPT	

Disk /dev/sdmb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmb1	1	1913	15364031+	ee	GPT	

Disk /dev/sdmd: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmd1	1	1913	15364031+	ee	GPT	

Disk /dev/sdmc: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmc1	1	1913	15364031+	ee	GPT	

Disk /dev/sdme: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdme1	1	2002	16080831+	ee	GPT	

Disk /dev/sdmf: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmf1	1	2002	16080831+	ee	GPT	

Disk /dev/sdmg: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmg1	1	1913	15364031+	ee	GPT	

Disk /dev/sdml: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdml1	1	1913	15364031+	ee	GPT	

Disk /dev/sdmm: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmm1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdmi: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmi1	1	2002	16080831+	ee	GPT	

Disk /dev/sdms: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdms1	1	2678	21508031+	ee	GPT	

Disk /dev/sdmh: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmh1	1	1913	15364031+	ee	GPT	

Disk /dev/sdmp: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmp1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdmt: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmt1	1	1913	15364031+	ee	GPT	

Disk /dev/sdmu: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdmu1	1	906	7274431+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:



Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnc1		1	7522	60420031+	ee	GPT

Disk /dev/sdnc: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnl1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdnn: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnn1		1	2678	21508031+	ee	GPT

Disk /dev/sdnc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnc1		1	2002	16080831+	ee	GPT

Disk /dev/sdnj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnj1		1	2002	16080831+	ee	GPT

Disk /dev/sdnn: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnl1		1	1403	11268031+	ee	GPT

Disk /dev/sdnl: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnl1		1	2002	16080831+	ee	GPT

Disk /dev/sdng: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdng1		1	2002	16080831+	ee	GPT

Disk /dev/sdnp: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnp1		1	2002	16080831+	ee	GPT

Disk /dev/sdno: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdno1		1	2002	16080831+	ee	GPT

Disk /dev/sdnq: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnq1		1	2002	16080831+	ee	GPT

Disk /dev/sdnc: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnc1		1	1977	15876031+	ee	GPT

Disk /dev/sdnc: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnc1		1	7522	60420031+	ee	GPT

Disk /dev/sdnc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnc1		1	2002	16080831+	ee	GPT

Disk /dev/sdnc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnc1		1	2002	16080831+	ee	GPT

Disk /dev/sdnc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnc1		1	2002	16080831+	ee	GPT

Disk /dev/sdnc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdnc1		1	2002	16080831+	ee	GPT

Disk /dev/sdnc: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders





Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdop1		1	2002	16080831+	ee	GPT

Disk /dev/sdos: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdos1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdoq: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdoq1		1	1913	15364031+	ee	GPT

Disk /dev/sdoj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdoj1		1	2002	16080831+	ee	GPT

Disk /dev/sdot: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdot1		1	1913	15364031+	ee	GPT

Disk /dev/sdor: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdor1		1	1913	15364031+	ee	GPT

Disk /dev/sdov: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdov1		1	2002	16080831+	ee	GPT

Disk /dev/sdow: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdow1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdou: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdou1		1	1913	15364031+	ee	GPT

Disk /dev/sdpc: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpc1		1	1913	15364031+	ee	GPT

Disk /dev/sdoy: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdoy1		1	2002	16080831+	ee	GPT

Disk /dev/sdpa: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpa1		1	7522	60420031+	ee	GPT

Disk /dev/sdox: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdox1		1	1913	15364031+	ee	GPT

Disk /dev/sdpd: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpd1		1	7522	60420031+	ee	GPT

Disk /dev/sdpb: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpb1		1	1977	15876031+	ee	GPT

Disk /dev/sdpi: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpi1		1	1913	15364031+	ee	GPT

Disk /dev/sdph: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

/dev/sdph1 1 2002 16080831+ ee GPT

Disk /dev/sdoz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdoz1	1	1913	15364031+	ee	GPT	

Disk /dev/sdpm: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpm1	1	1913	15364031+	ee	GPT	

Disk /dev/sdpg: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpg1	1	2002	16080831+	ee	GPT	

Disk /dev/sdph: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdph1	1	7522	60420031+	ee	GPT	

Disk /dev/sdpl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpl1	1	2002	16080831+	ee	GPT	

Disk /dev/sdpe: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpe1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdpk: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpk1	1	2678	21508031+	ee	GPT	

Disk /dev/sdpf: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpf1	1	2678	21508031+	ee	GPT	

Disk /dev/sdpo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpo1	1	2002	16080831+	ee	GPT	

Disk /dev/sdpn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpn1	1	2002	16080831+	ee	GPT	

Disk /dev/sdpq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpq1	1	2002	16080831+	ee	GPT	

Disk /dev/sdpp: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpp1	1	1913	15364031+	ee	GPT	

Disk /dev/sdpv: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpv1	1	944	7581631+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdps: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdps1	1	2002	16080831+	ee	GPT	

Disk /dev/sdpw: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpw1	1	2002	16080831+	ee	GPT	

Disk /dev/sdpy: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdpy1	1	2678	21508031+	ee	GPT	

Disk /dev/sdpr: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes





Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdqx1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdrh: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrh1		1	2002	16080831+	ee	GPT

Disk /dev/sdri: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdri1		1	2002	16080831+	ee	GPT

Disk /dev/sdra: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdra1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdrj: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrj1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdrk: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrk1		1	7522	60420031+	ee	GPT

Disk /dev/sdrg: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrg1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdrl: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrl1		1	1913	15364031+	ee	GPT

Disk /dev/sdrn: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrn1		1	2678	21508031+	ee	GPT

Disk /dev/sdrs: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrs1		1	7522	60420031+	ee	GPT

Disk /dev/sdrp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrp1		1	2002	16080831+	ee	GPT

Disk /dev/sdrv: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrv1		1	1977	15876031+	ee	GPT

Disk /dev/sdrw: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdrw1		1	2002	16080831+	ee	GPT

Disk /dev/sdru: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdru1		1	1913	15364031+	ee	GPT

Disk /dev/sdsa: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsa1		1	2678	21508031+	ee	GPT

Disk /dev/sdsb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsb1		1	2002	16080831+	ee	GPT



I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsl1		1	1913	15364031+	ee	GPT

Disk /dev/sdsm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsm1		1	2002	16080831+	ee	GPT

Disk /dev/sdsn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsn1		1	1913	15364031+	ee	GPT

Disk /dev/sdso: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdso1		1	2002	16080831+	ee	GPT

Disk /dev/sdsq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsq1		1	1913	15364031+	ee	GPT

Disk /dev/sdss: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdss1		1	1913	15364031+	ee	GPT

Disk /dev/sdsp: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsp1		1	1913	15364031+	ee	GPT

Disk /dev/sdsr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsr1		1	1913	15364031+	ee	GPT

Disk /dev/sdsu: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsu1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdst: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdst1		1	2002	16080831+	ee	GPT

Disk /dev/sdsv: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsv1		1	1913	15364031+	ee	GPT

Disk /dev/sdsw: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsw1		1	1913	15364031+	ee	GPT

Disk /dev/sdsx: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsx1		1	1913	15364031+	ee	GPT

Disk /dev/sdsz: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsz1		1	2678	21508031+	ee	GPT

Disk /dev/sdsy: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdsy1		1	2002	16080831+	ee	GPT

Disk /dev/sdtb: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtb1		1	1913	15364031+	ee	GPT

Disk /dev/sdte: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdte1		1	2002	16080831+	ee	GPT

Disk /dev/sdta: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdta1		1	1403	11268031+	ee	GPT

Disk /dev/sdtd: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtd1		1	1913	15364031+	ee	GPT

Disk /dev/sdtf: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtf1		1	2002	16080831+	ee	GPT

Disk /dev/sdtg: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtg1		1	7522	60420031+	ee	GPT

Disk /dev/sdth: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdth1		1	1913	15364031+	ee	GPT

Disk /dev/sdte: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdte1		1	1977	15876031+	ee	GPT

Disk /dev/sdtl: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtl1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdti: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdti1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdtj: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtj1		1	1913	15364031+	ee	GPT

Disk /dev/sdtk: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0xfedd0850

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtk1		1	1913	15364031+	ee	GPT

Disk /dev/sdto: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdto1		1	2002	16080831+	ee	GPT

Disk /dev/sdtq: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtql		1	2678	21508031+	ee	GPT

Disk /dev/sdtn: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtn1		1	7522	60420031+	ee	GPT

Disk /dev/sdtp: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtp1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdts: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdts1		1	7522	60420031+	ee	GPT

Disk /dev/sdtm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtm1		1	2002	16080831+	ee	GPT

Disk /dev/sdtt: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000



Device	Boot	Start	End	Blocks	Id	System
/dev/sdtt1	1	2002	16080831+	ee	GPT	

Disk /dev/sdtr: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtr1	1	2002	16080831+	ee	GPT	

Disk /dev/sdtw: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtw1	1	1913	15364031+	ee	GPT	

Disk /dev/sdtu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtu1	1	2002	16080831+	ee	GPT	

Disk /dev/sdtz: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtz1	1	1913	15364031+	ee	GPT	

Disk /dev/sdtx: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdtx1	1	1913	15364031+	ee	GPT	

Disk /dev/sdttv: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdttv1	1	906	7274431+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdty: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdty1	1	2002	16080831+	ee	GPT	

Disk /dev/sduc: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduc1	1	1913	15364031+	ee	GPT	

Disk /dev/sdua: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdua1	1	2002	16080831+	ee	GPT	

Disk /dev/sdud: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdud1	1	2002	16080831+	ee	GPT	

Disk /dev/sdub: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

Device	Boot	Start	End	Blocks	Id	System
/dev/sdub1	1	7522	60420031+	ee	GPT	

Disk /dev/sduf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduf1	1	1913	15364031+	ee	GPT	

Disk /dev/sdug: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdug1	1	2002	16080831+	ee	GPT	

Disk /dev/sdue: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdue1	1	2002	16080831+	ee	GPT	

Disk /dev/sduj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduj1	1	2002	16080831+	ee	GPT	

Disk /dev/sduh: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduh1	1	1913	15364031+	ee	GPT	

Disk /dev/sdui: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdui1		1	7522	60420031+	ee	GPT

Disk /dev/sduk: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduk1		1	1913	15364031+	ee	GPT

Disk /dev/sdun: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdun1		1	1403	11268031+	ee	GPT

Disk /dev/sdum: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdum1		1	2002	16080831+	ee	GPT

Disk /dev/sduo: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduo1		1	1977	15876031+	ee	GPT

Disk /dev/sdup: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdup1		1	2002	16080831+	ee	GPT

Disk /dev/sduq: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduq1		1	2002	16080831+	ee	GPT

Disk /dev/sdul: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdull1		1	2002	16080831+	ee	GPT

Disk /dev/sdur: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdur1		1	2678	21508031+	ee	GPT

Disk /dev/sdut: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdut1		1	1913	15364031+	ee	GPT

Disk /dev/sdus: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdus1		1	2002	16080831+	ee	GPT

Disk /dev/sduu: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduu1		1	1977	15876031+	ee	GPT

Disk /dev/sduv: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduv1		1	2002	16080831+	ee	GPT

Disk /dev/sdux: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdux1		1	2002	16080831+	ee	GPT

Disk /dev/sduw: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduw1		1	7522	60420031+	ee	GPT

Disk /dev/sduz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduz1		1	2002	16080831+	ee	GPT

Disk /dev/sduy: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sduy1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdva: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdva1 1 1913 15364031+ ee GPT

Disk /dev/sdvc: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvc1 1 7522 60420031+ ee GPT

Disk /dev/sdvd: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvd1 1 2002 16080831+ ee GPT

Disk /dev/sdvb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvb1 1 2002 16080831+ ee GPT

Disk /dev/sdve: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdve1 1 2002 16080831+ ee GPT

Disk /dev/sdvf: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvf1 1 1913 15364031+ ee GPT

Disk /dev/sdvo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvo1 1 2002 16080831+ ee GPT

Disk /dev/sdvg: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvg1 1 2002 16080831+ ee GPT

Disk /dev/sdvi: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvi1 1 2002 16080831+ ee GPT

Disk /dev/sdvj: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvj1 1 1913 15364031+ ee GPT

Disk /dev/sdvk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvk1 1 2002 16080831+ ee GPT

Disk /dev/sdvl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvl1 1 2002 16080831+ ee GPT

Disk /dev/sdvm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvm1 1 2002 16080831+ ee GPT

Disk /dev/sdvn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvn1 1 2002 16080831+ ee GPT

Disk /dev/sdvh: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvh1 1 2002 16080831+ ee GPT

Disk /dev/sdvp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvp1 1 2002 16080831+ ee GPT

Disk /dev/sdvr: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdvr1 1 944 7581631+ ee GPT  
Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdvq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000



Disk /dev/sdwl: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwl		1	1913	15364031+	ee	GPT

Disk /dev/sdwj: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwj		1	1913	15364031+	ee	GPT

Disk /dev/sdwl: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwl		1	2002	16080831+	ee	GPT

Disk /dev/sdwm: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwm		1	1977	15876031+	ee	GPT

Disk /dev/sdwn: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwn		1	1403	11268031+	ee	GPT

Disk /dev/sdvw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdvw		1	2002	16080831+	ee	GPT

/dev/sdww: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdww		1	1977	15876031+	ee	GPT

Disk /dev/sdwp: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwp		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdwo: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwo		1	2002	16080831+	ee	GPT

Disk /dev/sdww: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdww		1	2002	16080831+	ee	GPT

Disk /dev/sdws: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdws		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdwt: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwt		1	2002	16080831+	ee	GPT

Disk /dev/sdww: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdww		1	7522	60420031+	ee	GPT

Disk /dev/sdwy: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwy		1	2002	16080831+	ee	GPT

Disk /dev/sdwr: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwr		1	7522	60420031+	ee	GPT

Disk /dev/sdwx: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwx		1	2002	16080831+	ee	GPT

Disk /dev/sdxa: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxa1		1	2002	16080831+	ee	GPT

Disk /dev/sdwz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdwz1		1	2002	16080831+	ee	GPT

Disk /dev/sdxb: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxb1		1	2002	16080831+	ee	GPT

Disk /dev/sdxc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxc1		1	2002	16080831+	ee	GPT

Disk /dev/sdxg: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxg1		1	1913	15364031+	ee	GPT

Disk /dev/sdxd: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxd1		1	2002	16080831+	ee	GPT

Disk /dev/sdxf: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxfl		1	2002	16080831+	ee	GPT

Disk /dev/sdxi: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxi1		1	1913	15364031+	ee	GPT

Disk /dev/sdxk: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxk1		1	1403	11268031+	ee	GPT

Disk /dev/sdxj: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxj1		1	1913	15364031+	ee	GPT

Disk /dev/sdxl: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxl1		1	1977	15876031+	ee	GPT

Disk /dev/sdxe: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxe1		1	2002	16080831+	ee	GPT

Disk /dev/sdxh: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxh1		1	1913	15364031+	ee	GPT

Disk /dev/sdxm: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxm1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdxn: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxn1		1	7522	60420031+	ee	GPT

Disk /dev/sdxo: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxo1		1	2002	16080831+	ee	GPT

Disk /dev/sdxt: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxt1		1	2002	16080831+	ee	GPT

Disk /dev/sdxu: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxu1		1	1913	15364031+	ee	GPT

Disk /dev/sdxx: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxx1		1	1913	15364031+	ee	GPT

Disk /dev/sdyb: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdyb1		1	7522	60420031+	ee	GPT

Disk /dev/sdxv: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxv1		1	1913	15364031+	ee	GPT

Disk /dev/sdyc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdyc1		1	2002	16080831+	ee	GPT

Disk /dev/sdxy: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxy1		1	1403	11268031+	ee	GPT

Disk /dev/sdxs: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxs1		1	2002	16080831+	ee	GPT

Disk /dev/sdye: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdye1		1	2002	16080831+	ee	GPT

Disk /dev/sdxw: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxw1		1	1913	15364031+	ee	GPT

Disk /dev/sdya: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdya1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdxz: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxz1		1	1977	15876031+	ee	GPT

Disk /dev/sdyd: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

Device	Boot	Start	End	Blocks	Id	System
/dev/sdyd1		1	2002	16080831+	ee	GPT

Disk /dev/sdyf: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdyf1		1	2002	16080831+	ee	GPT

Disk /dev/sdxp: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxp1		1	2002	16080831+	ee	GPT

Disk /dev/sdxr: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxr1		1	2002	16080831+	ee	GPT

Disk /dev/sdxq: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdxq1		1	2002	16080831+	ee	GPT

Disk /dev/sdyg: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdyg1		1	2002	16080831+	ee	GPT

Disk /dev/sdyk: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyk1 1 1913 15364031+ ee GPT

Disk /dev/sdym: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdym1 1 1403 11268031+ ee GPT

Disk /dev/sdyi: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyil 1 1913 15364031+ ee GPT

Disk /dev/sdyo: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyo1 1 460 3690431+ ee GPT  
Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdyh: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyh1 1 2002 16080831+ ee GPT

Disk /dev/sdyj: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyj1 1 1913 15364031+ ee GPT

Disk /dev/sdyl: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyl1 1 1913 15364031+ ee GPT

Disk /dev/sdyn: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyn1 1 1977 15876031+ ee GPT

Disk /dev/sdyp: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyp1 1 7522 60420031+ ee GPT

Disk /dev/sdyq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyq1 1 2002 16080831+ ee GPT

Disk /dev/sdyr: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyr1 1 2002 16080831+ ee GPT

Disk /dev/sdys: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System

/dev/sdys1 1 2002 16080831+ ee GPT

Disk /dev/sdyw: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyw1 1 1913 15364031+ ee GPT

Disk /dev/sdyy: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyy1 1 1913 15364031+ ee GPT

Disk /dev/sdzb: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdzb1 1 1977 15876031+ ee GPT

Disk /dev/sdza: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdza1 1 1403 11268031+ ee GPT

Disk /dev/sdyz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdyz1 1 1913 15364031+ ee GPT

Disk /dev/sdyx: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000



Device	Boot	Start	End	Blocks	Id	System
/dev/sdyx1		1	1913	15364031+	ee	GPT

Disk /dev/sdyt: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdyt1		1	2002	16080831+	ee	GPT

Disk /dev/sdyv: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdyv1		1	2002	16080831+	ee	GPT

Disk /dev/sdzc: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdze1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdyu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdyu1		1	2002	16080831+	ee	GPT

Disk /dev/sdzl: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzl1		1	1913	15364031+	ee	GPT

Disk /dev/sdzd: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzd1		1	7522	60420031+	ee	GPT

Disk /dev/sdzh: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzh1		1	2002	16080831+	ee	GPT

Disk /dev/sdzm: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzm1		1	1913	15364031+	ee	GPT

Disk /dev/sdzi: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzi1		1	2002	16080831+	ee	GPT

Disk /dev/sdzj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzj1		1	2002	16080831+	ee	GPT

Disk /dev/sdzn: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzn1		1	1913	15364031+	ee	GPT

Disk /dev/sdzk: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzk1		1	1913	15364031+	ee	GPT

Disk /dev/sdzo: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzo1		1	1403	11268031+	ee	GPT

Disk /dev/sdze: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdze1		1	2002	16080831+	ee	GPT

Disk /dev/sdzf: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzf1		1	2002	16080831+	ee	GPT

Disk /dev/sdzp: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdzp1		1	1977	15876031+	ee	GPT

Disk /dev/sdzg: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdzg1 1 2002 16080831+ ee GPT

Disk /dev/sdq: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdq1 1 460 3690431+ ee GPT  
Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdzt: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdzt1 1 2002 16080831+ ee GPT

Disk /dev/sdzu: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdzu1 1 2002 16080831+ ee GPT

Disk /dev/sdzz: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdzz1 1 7522 60420031+ ee GPT

Disk /dev/sdzz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdzz1 1 1913 15364031+ ee GPT

Disk /dev/sdzy: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdzy1 1 1913 15364031+ ee GPT

Disk /dev/sdaad: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaad1 1 1977 15876031+ ee GPT

Disk /dev/sdaab: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaab1 1 1913 15364031+ ee GPT

Disk /dev/sdxx: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdxx1 1 2002 16080831+ ee GPT

Disk /dev/sdaaa: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaaa1 1 1913 15364031+ ee GPT

Disk /dev/sdzw: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System

/dev/sdzw1 1 2002 16080831+ ee GPT

Disk /dev/sdaae: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaae1 1 460 3690431+ ee GPT  
Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdzs: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdzs1 1 2002 16080831+ ee GPT

Disk /dev/sdzv: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdzv1 1 2002 16080831+ ee GPT

Disk /dev/sdaac: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaac1 1 1403 11268031+ ee GPT

Disk /dev/sdaaf: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaaf1 1 7522 60420031+ ee GPT

Disk /dev/sdaah: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaah1		1	2002	16080831+	ee	GPT

Disk /dev/sdaam: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaam1		1	1913	15364031+	ee	GPT

Disk /dev/sdaap: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaap1		1	1913	15364031+	ee	GPT

Disk /dev/sdaao: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaao1		1	1913	15364031+	ee	GPT

Disk /dev/sdaag: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaag1		1	2002	16080831+	ee	GPT

Disk /dev/sdaan: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaan1		1	1913	15364031+	ee	GPT

Disk /dev/sdaai: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaai1		1	2002	16080831+	ee	GPT

Disk /dev/sdaaj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaaj1		1	2002	16080831+	ee	GPT

Disk /dev/sdaak: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaak1		1	2002	16080831+	ee	GPT

Disk /dev/sdaal: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaal1		1	2002	16080831+	ee	GPT

Disk /dev/sdaaq: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaaq1		1	1403	11268031+	ee	GPT

Disk /dev/sdaas: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaas1		1	1913	15364031+	ee	GPT

/dev/sdaas1 1 460 3690431+ ee GPT  
 Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdaar: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaar1		1	1977	15876031+	ee	GPT

Disk /dev/sdaat: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaat1		1	7522	60420031+	ee	GPT

Disk /dev/sdaau: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaau1		1	2002	16080831+	ee	GPT

Disk /dev/sdaav: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaav1		1	2002	16080831+	ee	GPT

Disk /dev/sdaax: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaax1		1	2002	16080831+	ee	GPT

Disk /dev/sdaaw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaaw1		1	2002	16080831+	ee	GPT

Disk /dev/sdaay: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaay1		1	2002	16080831+	ee	GPT

Disk /dev/sdaba: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaba1		1	1913	15364031+	ee	GPT

Disk /dev/sdaaz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaaz1		1	2002	16080831+	ee	GPT

Disk /dev/sdabe: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabe1		1	1403	11268031+	ee	GPT

Disk /dev/sdabb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabb1		1	1913	15364031+	ee	GPT

Disk /dev/sdabd: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabd1		1	1913	15364031+	ee	GPT

Disk /dev/sdabc: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabc1		1	1913	15364031+	ee	GPT

Disk /dev/sdabg: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabg1		1	7522	60420031+	ee	GPT

Disk /dev/sdabf: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabf1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdabh: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabh1		1	7522	60420031+	ee	GPT

Disk /dev/sdabi: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabi1		1	2002	16080831+	ee	GPT

Disk /dev/sdabj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabj1		1	2002	16080831+	ee	GPT

Disk /dev/sdabo: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabo1		1	1913	15364031+	ee	GPT

Disk /dev/sdabl: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabl1		1	2002	16080831+	ee	GPT

Disk /dev/sdabm: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabm1		1	2002	16080831+	ee	GPT

Disk /dev/sdabk: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabk1		1	2002	16080831+	ee	GPT

Disk /dev/sdabn: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabn1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdabq: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabq1		1	1913	15364031+	ee	GPT

Disk /dev/sdabr: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabr1		1	1913	15364031+	ee	GPT

Disk /dev/sdabp: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabp1		1	1913	15364031+	ee	GPT

Disk /dev/sdabs: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabs1		1	1403	11268031+	ee	GPT

Disk /dev/sdabt: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabt1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdabu: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabu1		1	7522	60420031+	ee	GPT

Disk /dev/sdabv: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabv1		1	7522	60420031+	ee	GPT

Disk /dev/sdabw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabw1		1	2002	16080831+	ee	GPT

Disk /dev/sdabx: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabx1		1	2002	16080831+	ee	GPT

Disk /dev/sdacc: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacc1		1	1913	15364031+	ee	GPT

Disk /dev/sdaby: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaby1		1	2002	16080831+	ee	GPT

Disk /dev/sdacd: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacd1		1	1913	15364031+	ee	GPT

Disk /dev/sdabz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdabz1		1	2002	16080831+	ee	GPT

Disk /dev/sdaca: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaca1		1	2002	16080831+	ee	GPT

Disk /dev/sdacf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacf1		1	1913	15364031+	ee	GPT

Disk /dev/sdace: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

/dev/sdace1 1 1913 15364031+ ee GPT

Disk /dev/sdach: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdach1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):

phys=(0, 0, 1) logical=(0, 0, 2)

Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdacb: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacb1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):

phys=(0, 0, 1) logical=(0, 0, 2)

Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdacg: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacg1		1	1403	11268031+	ee	GPT

Disk /dev/sdadj: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadj1		1	7522	60420031+	ee	GPT

Disk /dev/sdaci: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

/dev/sdaci1 1 7522 60420031+ ee GPT

Disk /dev/sdacl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacl1		1	2002	16080831+	ee	GPT

Disk /dev/sdack: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdack1		1	2002	16080831+	ee	GPT

Disk /dev/sdacq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacq1		1	1913	15364031+	ee	GPT

Disk /dev/sdacr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacr1		1	1913	15364031+	ee	GPT

Disk /dev/sdact: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdact1		1	1913	15364031+	ee	GPT

Disk /dev/sdacm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacm1		1	2002	16080831+	ee	GPT

Disk /dev/sdacs: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacs1		1	1913	15364031+	ee	GPT

Disk /dev/sdacl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacl1		1	2002	16080831+	ee	GPT

Disk /dev/sdacu: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacu1		1	1403	11268031+	ee	GPT

Disk /dev/sdacc: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdacc1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):

phys=(0, 0, 1) logical=(0, 0, 2)

Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdaco: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaco1		1	2002	16080831+	ee	GPT



Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadl1		1	7522	60420031+	ee	GPT

Disk /dev/sdadn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadn1		1	2002	16080831+	ee	GPT

Disk /dev/sdadm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadm1		1	2002	16080831+	ee	GPT

Disk /dev/sdadq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadq1		1	2002	16080831+	ee	GPT

Disk /dev/sdadr: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadrl		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdads: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdads1		1	1913	15364031+	ee	GPT

Disk /dev/sdadt: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadt1		1	1913	15364031+	ee	GPT

Disk /dev/sdado: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdado1		1	2002	16080831+	ee	GPT

Disk /dev/sdadp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadp1		1	2002	16080831+	ee	GPT

Disk /dev/sdadu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadu1		1	1913	15364031+	ee	GPT

Disk /dev/sdadv: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadv1		1	1913	15364031+	ee	GPT

Disk /dev/sdadw: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadw1		1	1403	11268031+	ee	GPT

Disk /dev/sdady: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdady1		1	7522	60420031+	ee	GPT

Disk /dev/sdaef: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaef1		1	1913	15364031+	ee	GPT

Disk /dev/sdadx: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdadx1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdaee: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaee1		1	2002	16080831+	ee	GPT

Disk /dev/sdadz: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------



/dev/sdadz1 1 7522 60420031+ ee GPT

Disk /dev/sdaec: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaec1	1	2002	16080831+	ee	GPT	

Disk /dev/sdaeh: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaeh1	1	1913	15364031+	ee	GPT	

Disk /dev/sdaeb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaeb1	1	2002	16080831+	ee	GPT	

Disk /dev/sdaei: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaei1	1	1913	15364031+	ee	GPT	

Disk /dev/sdaea: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaea1	1	2002	16080831+	ee	GPT	

Disk /dev/sdaej: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaej1	1	1913	15364031+	ee	GPT	

Disk /dev/sdaed: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaed1	1	2002	16080831+	ee	GPT	

Disk /dev/sdaeg: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaeg1	1	65	516031+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdaek: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaek1	1	1913	15364031+	ee	GPT	

Disk /dev/sdael: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdael1	1	1403	11268031+	ee	GPT	

Disk /dev/sdaen: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaen1	1	1913	15364031+	ee	GPT	

Disk /dev/sdaem: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaem1	1	906	7274431+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdaer: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaer1	1	1913	15364031+	ee	GPT	

Disk /dev/sdaep: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaep1	1	1913	15364031+	ee	GPT	

Disk /dev/sdaet: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaet1	1	1403	11268031+	ee	GPT	

Disk /dev/sdaes: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaes1	1	2002	16080831+	ee	GPT	

Disk /dev/sdaeo: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdae0l		1	7522	60420031+	ee	GPT

Disk /dev/sdaq: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaq1		1	7522	60420031+	ee	GPT

Disk /dev/sdae: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdae1		1	2002	16080831+	ee	GPT

Disk /dev/sdaex: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaex1		1	7522	60420031+	ee	GPT

Disk /dev/sdaev: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdae1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdaew: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaew1		1	2002	16080831+	ee	GPT

Disk /dev/sdafb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdafb1		1	2002	16080831+	ee	GPT

Disk /dev/sdaey: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaey1		1	2002	16080831+	ee	GPT

Disk /dev/sdafa: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdafa1		1	2002	16080831+	ee	GPT

Disk /dev/sdafa: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaf1		1	1913	15364031+	ee	GPT

Disk /dev/sdafc: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdafc1		1	2002	16080831+	ee	GPT

Disk /dev/sdaez: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdae1		1	7522	60420031+	ee	GPT

Disk /dev/sdaff: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaff1		1	2002	16080831+	ee	GPT

Disk /dev/sdafd: 1262 MB, 1262419968 bytes  
255 heads, 63 sectors/track, 153 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdafd1		1	154	1232831+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(153, 122, 33)

Disk /dev/sdafg: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdafg1		1	2002	16080831+	ee	GPT

Disk /dev/sdafh: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdafh1		1	1913	15364031+	ee	GPT

Disk /dev/sdafl: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdafl1		1	1913	15364031+	ee	GPT

```

Disk /dev/sdafk: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafk1 1 1913 15364031+ ee GPT

Disk /dev/sdafn: 11.5 GB, 11538464768 bytes
255 heads, 63 sectors/track, 1402 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafn1 1 1403 11268031+ ee GPT

Disk /dev/sdafr: 61.9 GB, 61870112768 bytes
255 heads, 63 sectors/track, 7521 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafr1 1 7522 60420031+ ee GPT

Disk /dev/sdafj: 528 MB, 528416768 bytes
255 heads, 63 sectors/track, 64 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafj1 1 65 516031+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdafm: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafm1 1 1913 15364031+ ee GPT

Disk /dev/sdafo: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafo1 1 1913 15364031+ ee GPT

Disk /dev/sdafq: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafq1 1 1913 15364031+ ee GPT

I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafq1 1 1913 15364031+ ee GPT

Disk /dev/sdafv: 11.5 GB, 11538464768 bytes
255 heads, 63 sectors/track, 1402 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafv1 1 1403 11268031+ ee GPT

Disk /dev/sdafx: 7449 MB, 7449018368 bytes
255 heads, 63 sectors/track, 905 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafx1 1 7522 60420031+ ee GPT

Disk /dev/sdafp: 7449 MB, 7449018368 bytes
255 heads, 63 sectors/track, 905 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafp1 1 906 7274431+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdafz: 61.9 GB, 61870112768 bytes
255 heads, 63 sectors/track, 7521 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafz1 1 7522 60420031+ ee GPT

Disk /dev/sdafy: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System
/dev/sdafy1 1 2002 16080831+ ee GPT

Disk /dev/sdafw: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders

```

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdafw1		1	2002	16080831+	ee	GPT

Disk /dev/sdagb: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagb1		1	7522	60420031+	ee	GPT

Disk /dev/sdage: 55 MB, 55508992 bytes  
 255 heads, 63 sectors/track, 6 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdage1		1	7	54207+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(6, 190, 56)

Disk /dev/sdagd: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagd1		1	2002	16080831+	ee	GPT

Disk /dev/sdaga: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaga1		1	2002	16080831+	ee	GPT

Disk /dev/sdagc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagc1		1	2002	16080831+	ee	GPT

Disk /dev/sdagg: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaggl		1	2002	16080831+	ee	GPT

Disk /dev/sdagf: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagfl		1	2002	16080831+	ee	GPT

Disk /dev/sdagh: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaghl		1	2002	16080831+	ee	GPT

Disk /dev/sdagi: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagil		1	2002	16080831+	ee	GPT

Disk /dev/sdagk: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagkl		1	1913	15364031+	ee	GPT

Disk /dev/sdagl: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagll		1	1913	15364031+	ee	GPT

Disk /dev/sdagj: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagjl		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdagm: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagml		1	1913	15364031+	ee	GPT

Disk /dev/sdago: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagol		1	1913	15364031+	ee	GPT

Disk /dev/sdagq: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdagql		1	1913	15364031+	ee	GPT

Disk /dev/sdagp: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

```

/dev/sdagpl      1      1913  15364031+ ee GPT
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdagn1      1      1913  15364031+ ee GPT

Disk /dev/sdagn: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdagn1      1      1913  15364031+ ee GPT

Disk /dev/sdagr: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdagr1      1      1913  15364031+ ee GPT

Disk /dev/sdags: 11.5 GB, 11538464768 bytes
255 heads, 63 sectors/track, 1402 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdags1      1      1403  11268031+ ee GPT

Disk /dev/sdagx: 61.9 GB, 61870112768 bytes
255 heads, 63 sectors/track, 7521 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdagx1      1      7522  60420031+ ee GPT

Disk /dev/sdagv: 7449 MB, 7449018368 bytes
255 heads, 63 sectors/track, 905 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdagv1      1      906  7274431+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdagz: 61.9 GB, 61870112768 bytes
255 heads, 63 sectors/track, 7521 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdagz1      1      7522  60420031+ ee GPT

Disk /dev/sdaha: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdaha1      1      1913  15364031+ ee GPT

Disk /dev/sdahb: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdahb1      1      2002  16080831+ ee GPT

Disk /dev/sdahc: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdahc1      1      2002  16080831+ ee GPT

Disk /dev/sdaha: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdaha1      1      2002  16080831+ ee GPT

Disk /dev/sdahf: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdahf1      1      2002  16080831+ ee GPT

Disk /dev/sdahg: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdahg1      1      1913  15364031+ ee GPT

Disk /dev/sdahh: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdahh1      1      1913  15364031+ ee GPT

Disk /dev/sdahi: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

Device Boot      Start    End  Blocks Id System
/dev/sdahi1      1      1913  15364031+ ee GPT

```

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahg1		1	2002	16080831+	ee	GPT

Disk /dev/sdahj: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahj1		1	2002	16080831+	ee	GPT

Disk /dev/sdahh: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahh1		1	2002	16080831+	ee	GPT

Disk /dev/sdahd: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahd1		1	2002	16080831+	ee	GPT

Disk /dev/sdahi: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahi1		1	2002	16080831+	ee	GPT

Disk /dev/sdahk: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahk1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):

phys=(0, 0, 1) logical=(0, 0, 2)

Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdahm: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahm1		1	1913	15364031+	ee	GPT

Disk /dev/sdahn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahn1		1	1913	15364031+	ee	GPT

Disk /dev/sdaho: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaho1		1	1913	15364031+	ee	GPT

Disk /dev/sdahp: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahp1		1	1403	11268031+	ee	GPT

Disk /dev/sdahq: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahq1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):

phys=(0, 0, 1) logical=(0, 0, 2)

Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdahs: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahs1		1	7522	60420031+	ee	GPT

Disk /dev/sdahr: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahr1		1	7522	60420031+	ee	GPT

Disk /dev/sdahv: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahv1		1	2002	16080831+	ee	GPT

Disk /dev/sdaht: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaht1		1	2002	16080831+	ee	GPT

Disk /dev/sdahx: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahx1		1	2002	16080831+	ee	GPT

Disk /dev/sdahz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahz1		1	1913	15364031+	ee	GPT

Disk /dev/sdahw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahw1		1	2002	16080831+	ee	GPT

Disk /dev/sdahu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahu1		1	2002	16080831+	ee	GPT

Disk /dev/sdahy: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahy1		1	1913	15364031+	ee	GPT

Disk /dev/sdaia: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaia1		1	1913	15364031+	ee	GPT

Disk /dev/sdaib: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaib1		1	1913	15364031+	ee	GPT

Disk /dev/sdaic: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaic1		1	1403	11268031+	ee	GPT

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaic1		1	1403	11268031+	ee	GPT

Disk /dev/sdaie: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaie1		1	7522	60420031+	ee	GPT

Disk /dev/sdaid: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaid1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdaig: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaig1		1	1913	15364031+	ee	GPT

Disk /dev/sdaih: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaih1		1	2002	16080831+	ee	GPT

Disk /dev/sdaii: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaii1		1	1913	15364031+	ee	GPT

Disk /dev/sdaik: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaik1		1	1913	15364031+	ee	GPT

Disk /dev/sdaif: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaif1		1	7522	60420031+	ee	GPT

Disk /dev/sdaij: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaij1		1	2002	16080831+	ee	GPT

Disk /dev/sdaim: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaim1		1	1913	15364031+	ee	GPT

Disk /dev/sdail: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdail1		1	2002	16080831+	ee	GPT

Disk /dev/sdaio: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaio1		1	1403	11268031+	ee	GPT

Disk /dev/sdaip: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaip1		1	2002	16080831+	ee	GPT

Disk /dev/sdain: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdain1		1	2002	16080831+	ee	GPT

Disk /dev/sdaiq: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaiq1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdais: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdais1		1	7522	60420031+	ee	GPT

Disk /dev/sdair: 2150 MB, 2150563840 bytes  
 255 heads, 63 sectors/track, 261 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdair1		1	262	2100159+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(261, 116, 47)

Disk /dev/sdait: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdait1		1	7522	60420031+	ee	GPT

Disk /dev/sdaiu: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaiu1		1	1913	15364031+	ee	GPT

Disk /dev/sdaiv: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaiv1		1	2002	16080831+	ee	GPT

Disk /dev/sdaiw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaiw1		1	2002	16080831+	ee	GPT

Disk /dev/sdaix: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaix1		1	1913	15364031+	ee	GPT

Disk /dev/sdaiz: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaiz1		1	1913	15364031+	ee	GPT

Disk /dev/sdaiy: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaiy1		1	2002	16080831+	ee	GPT

Disk /dev/sdajb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajb1		1	1913	15364031+	ee	GPT

Disk /dev/sdaja: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaja1		1	2002	16080831+	ee	GPT

Disk /dev/sdaje: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaje1		1	1913	15364031+	ee	GPT

Disk /dev/sdajf: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajf1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdajg: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders



Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajl	1	1913	15364031+	ee	GPT	

Disk /dev/sdajh: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajhl	1	7522	60420031+	ee	GPT	

Disk /dev/sdajc: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajcl	1	2002	16080831+	ee	GPT	

Disk /dev/sdajd: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajdl	1	1403	11268031+	ee	GPT	

Disk /dev/sdajj: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajjl	1	7522	60420031+	ee	GPT	

Disk /dev/sdajl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajll	1	2002	16080831+	ee	GPT	

Disk /dev/sdajk: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajkl	1	1913	15364031+	ee	GPT	

Disk /dev/sdaji: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajil	1	1913	15364031+	ee	GPT	

Disk /dev/sdajm: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajml	1	1403	11268031+	ee	GPT	

Disk /dev/sdajo: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajol	1	906	7274431+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdajn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajnl	1	2002	16080831+	ee	GPT	

Disk /dev/sdajp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajpl	1	2002	16080831+	ee	GPT	

Disk /dev/sdajq: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajql	1	7522	60420031+	ee	GPT	

Disk /dev/sdajr: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajrl	1	2002	16080831+	ee	GPT	

Disk /dev/sdajv: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajvl	1	2002	16080831+	ee	GPT	

Disk /dev/sdajt: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajt1	1	2002	16080831+	ee	GPT	

Disk /dev/sdajx: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajxl	1	1913	15364031+	ee	GPT	

Disk /dev/sdajs: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajs1		1	7522	60420031+	ee	GPT

Disk /dev/sdaju: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaju1		1	2002	16080831+	ee	GPT

Disk /dev/sdajy: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajy1		1	2002	16080831+	ee	GPT

Disk /dev/sdajw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajw1		1	2002	16080831+	ee	GPT

Disk /dev/sdaka: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaka1		1	1913	15364031+	ee	GPT

Disk /dev/sdakb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakb1		1	1913	15364031+	ee	GPT

Disk /dev/sdajz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdajz1		1	2002	16080831+	ee	GPT

Disk /dev/sdakc: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakc1		1	1913	15364031+	ee	GPT

Disk /dev/sdakd: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakd1		1	1913	15364031+	ee	GPT

Disk /dev/sdakf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakf1		1	1913	15364031+	ee	GPT

Disk /dev/sdakh: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakh1		1	1913	15364031+	ee	GPT

Disk /dev/sdakg: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakg1		1	906	7274431+	ee	GPT

/dev/sdakg1 1 906 7274431+ ee GPT  
 Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdake: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdake1		1	1403	11268031+	ee	GPT

Disk /dev/sdaki: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaki1		1	7522	60420031+	ee	GPT

Disk /dev/sdakj: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakj1		1	1913	15364031+	ee	GPT

Disk /dev/sdako: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdako1		1	2002	16080831+	ee	GPT

Disk /dev/sdakp: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakp1		1	7522	60420031+	ee	GPT

Disk /dev/sdakq: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakk1		1	7522	60420031+	ee	GPT

Disk /dev/sdakn: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakan1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdakm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakml		1	2002	16080831+	ee	GPT

Disk /dev/sdakl: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakll		1	1403	11268031+	ee	GPT

Disk /dev/sdakq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakql		1	2002	16080831+	ee	GPT

Disk /dev/sdaks: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaks1		1	2002	16080831+	ee	GPT

Disk /dev/sdakr: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakrl		1	7522	60420031+	ee	GPT

Disk /dev/sdakt: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakt1		1	2002	16080831+	ee	GPT

Disk /dev/sdakv: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakov1		1	2002	16080831+	ee	GPT

Disk /dev/sdakw: 2101 MB, 2101280768 bytes  
255 heads, 63 sectors/track, 255 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakwl		1	256	2052031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(255, 118, 55)

Disk /dev/sdakx: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakxl		1	2002	16080831+	ee	GPT

Disk /dev/sdakz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdakz1		1	1913	15364031+	ee	GPT

Disk /dev/sdalc: 209 MB, 209649664 bytes  
255 heads, 63 sectors/track, 25 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalc1		1	26	204735+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(25, 124, 35)

Disk /dev/sdaku: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaku1		1	2002	16080831+	ee	GPT

Disk /dev/sdald: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdald1		1	1913	15364031+	ee	GPT

Disk /dev/sdala: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdala1		1	2002	16080831+	ee	GPT

Disk /dev/sdalb: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalb1		1	1913	15364031+	ee	GPT

Disk /dev/sdaky: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaky1		1	2002	16080831+	ee	GPT

Disk /dev/sdale: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdale1		1	1913	15364031+	ee	GPT

Disk /dev/sdali: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdali1		1	1913	15364031+	ee	GPT

Disk /dev/sdalf: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalf1		1	1403	11268031+	ee	GPT

Disk /dev/sdalg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalg1		1	1913	15364031+	ee	GPT

Disk /dev/sdalh: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdahl1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdalk: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalk1		1	1913	15364031+	ee	GPT

Disk /dev/sdalm: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalm1		1	1913	15364031+	ee	GPT

Disk /dev/sdalj: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalj1		1	7522	60420031+	ee	GPT

Disk /dev/sdaln: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaln1		1	2002	16080831+	ee	GPT

Disk /dev/sdall: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdall1		1	7522	60420031+	ee	GPT

Disk /dev/sdalo: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalo1		1	1403	11268031+	ee	GPT

Disk /dev/sdalaq: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalaq1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdalp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalp1		1	2002	16080831+	ee	GPT

Disk /dev/sdalar: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalar1		1	2002	16080831+	ee	GPT

Disk /dev/sdals: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdals1		1	1913	15364031+	ee	GPT

Disk /dev/sdalt: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalt1		1	2002	16080831+	ee	GPT

Disk /dev/sdama: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdama1		1	2002	16080831+	ee	GPT

Disk /dev/sdaly: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaly1		1	7522	60420031+	ee	GPT

Disk /dev/sdalw: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalw1		1	1913	15364031+	ee	GPT

Disk /dev/sdalz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalz1		1	1913	15364031+	ee	GPT

Disk /dev/sdalu: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalu1		1	7522	60420031+	ee	GPT

Disk /dev/sdalc: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalc1		1	1913	15364031+	ee	GPT

Disk /dev/sdalu: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalu1		1	2002	16080831+	ee	GPT

Disk /dev/sdalc: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdalc1		1	1913	15364031+	ee	GPT

Disk /dev/sdamb: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamb1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdamd: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamd1		1	2002	16080831+	ee	GPT

Disk /dev/sdamf: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamf1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdami: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdami1		1	7522	60420031+	ee	GPT

Disk /dev/sdame: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdame1		1	2002	16080831+	ee	GPT

Disk /dev/sdamj: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamj1		1	1913	15364031+	ee	GPT

Disk /dev/sdamg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamg1		1	1913	15364031+	ee	GPT

Disk /dev/sdaml: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaml1		1	7522	60420031+	ee	GPT

Disk /dev/sdamh: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamh1		1	2002	16080831+	ee	GPT

Disk /dev/sdamk: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamk1		1	2002	16080831+	ee	GPT

Disk /dev/sdamm: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaml1		1	1913	15364031+	ee	GPT

Disk /dev/sdamn: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamn1		1	2002	16080831+	ee	GPT

Disk /dev/sdamo: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamo1		1	1403	11268031+	ee	GPT

Disk /dev/sdamp: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamp1		1	2002	16080831+	ee	GPT

Disk /dev/sdamr: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamr1		1	2002	16080831+	ee	GPT

Disk /dev/sdamq: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamq1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdams: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdams1		1	7522	60420031+	ee	GPT

Disk /dev/sdamt: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamt1		1	1913	15364031+	ee	GPT

Disk /dev/sdamu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamu1		1	2002	16080831+	ee	GPT

Disk /dev/sdamv: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamv1		1	7522	60420031+	ee	GPT

Disk /dev/sdamx: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamx1		1	2002	16080831+	ee	GPT

Disk /dev/sdamw: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamw1		1	1913	15364031+	ee	GPT

Disk /dev/sdamy: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamy1		1	2002	16080831+	ee	GPT

Disk /dev/sdanb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdanb1		1	1913	15364031+	ee	GPT

Disk /dev/sdana: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdanal		1	1913	15364031+	ee	GPT

Disk /dev/sdamz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdamz1		1	1913	15364031+	ee	GPT

Disk /dev/sdanc: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdanc1		1	2002	16080831+	ee	GPT

Disk /dev/sdanf: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdanf1		1	1913	15364031+	ee	GPT

Disk /dev/sdand: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdand1		1	1403	11268031+	ee	GPT

Disk /dev/sdane: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdane1		1	2002	16080831+	ee	GPT

Disk /dev/sdanh: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdanh1		1	1913	15364031+	ee	GPT

Disk /dev/sdang: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdang1		1	2002	16080831+	ee	GPT

Disk /dev/sdanj: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdanj1		1	2002	16080831+	ee	GPT

Disk /dev/sdank: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdank1		1	1913	15364031+	ee	GPT

Disk /dev/sdani: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdani1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdanm: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdanm1		1	7522	60420031+	ee	GPT

Disk /dev/sdanl: 8493 MB, 8493400064 bytes  
255 heads, 63 sectors/track, 1032 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdanl1		1	1033	8294335+	ee	GPT

Disk /dev/sdann: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdann1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdano: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdano1		1	7522	60420031+	ee	GPT

Disk /dev/sdanp: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdanp1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdanq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------





Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaok1		1	1913	15364031+	ee	GPT

Disk /dev/sdaoh: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaoh1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdaon: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaon1		1	2002	16080831+	ee	GPT

Disk /dev/sdaoo: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaoo1		1	1913	15364031+	ee	GPT

Disk /dev/sdaom: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaom1		1	1913	15364031+	ee	GPT

Disk /dev/sdaoq: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaoql		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdaoj: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaoj1		1	7522	60420031+	ee	GPT

Disk /dev/sdaop: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaop1		1	2002	16080831+	ee	GPT

Disk /dev/sdaor: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaor1		1	2002	16080831+	ee	GPT

Disk /dev/sdaol: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaol1		1	7522	60420031+	ee	GPT

Disk /dev/sdaov: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaov1		1	7522	60420031+	ee	GPT

Disk /dev/sdaos: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaos1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdaox: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaox1		1	1913	15364031+	ee	GPT

Disk /dev/sdaot: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaot1		1	2002	16080831+	ee	GPT

Disk /dev/sdaou: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaou1		1	2002	16080831+	ee	GPT

Disk /dev/sdaow: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaow1		1	7522	60420031+	ee	GPT

Disk /dev/sdaoy: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaoy1		1	2002	16080831+	ee	GPT

Disk /dev/sdaoz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaoz1		1	2002	16080831+	ee	GPT

Disk /dev/sdapa: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapa1		1	1913	15364031+	ee	GPT

Disk /dev/sdape: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapc1		1	1913	15364031+	ee	GPT

Disk /dev/sdapb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapb1		1	2002	16080831+	ee	GPT

Disk /dev/sdapg: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapg1		1	2002	16080831+	ee	GPT

Disk /dev/sdape: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdape1		1	2002	16080831+	ee	GPT

Disk /dev/sdapf: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapf1		1	1913	15364031+	ee	GPT

Disk /dev/sdapg: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapg1		1	1403	11268031+	ee	GPT

Disk /dev/sdaph: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaph1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdapi: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapi1		1	7522	60420031+	ee	GPT

Disk /dev/sdapj: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapj1		1	7522	60420031+	ee	GPT

Disk /dev/sdapk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapk1		1	2002	16080831+	ee	GPT

Disk /dev/sdapp: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapp1		1	1913	15364031+	ee	GPT

Disk /dev/sdaps: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaps1		1	1913	15364031+	ee	GPT

Disk /dev/sdapo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapo1		1	2002	16080831+	ee	GPT

Disk /dev/sdapq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapq1		1	1913	15364031+	ee	GPT

Disk /dev/sdapl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdap1l		1	2002	16080831+	ee	GPT

Disk /dev/sdapm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapml		1	2002	16080831+	ee	GPT

Disk /dev/sdapn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapnl		1	2002	16080831+	ee	GPT

Disk /dev/sdapr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaprl		1	1913	15364031+	ee	GPT

Disk /dev/sdapv: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapvl		1	7522	60420031+	ee	GPT

Disk /dev/sdapy: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapy1		1	2002	16080831+	ee	GPT

Disk /dev/sdaqb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqbl		1	2002	16080831+	ee	GPT

Disk /dev/sdapt: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapt1		1	1403	11268031+	ee	GPT

Disk /dev/sdaqe: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqel		1	1913	15364031+	ee	GPT

Disk /dev/sdapz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapzl		1	2002	16080831+	ee	GPT

Disk /dev/sdapu: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapul		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)

Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdapx: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapx1		1	2002	16080831+	ee	GPT

Disk /dev/sdaqa: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqal		1	2002	16080831+	ee	GPT

Disk /dev/sdaqc: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqcl		1	1913	15364031+	ee	GPT

Disk /dev/sdapw: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdapwl		1	7522	60420031+	ee	GPT

Disk /dev/sdaqd: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqdl		1	1913	15364031+	ee	GPT

Disk /dev/sdaqf: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqfl		1	1913	15364031+	ee	GPT

Disk /dev/sdaqg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaq1		1	1913	15364031+	ee	GPT

Disk /dev/sdaqm: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqm1		1	7522	60420031+	ee	GPT

Disk /dev/sdaqj: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqj1		1	1913	15364031+	ee	GPT

Disk /dev/sdaq: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaq1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdaqn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqn1		1	1913	15364031+	ee	GPT

Disk /dev/sdaqh: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqh1		1	1403	11268031+	ee	GPT

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqh1		1	1403	11268031+	ee	GPT

Disk /dev/sdaq1: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaq11		1	1913	15364031+	ee	GPT

Disk /dev/sdaqk: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqk1		1	7522	60420031+	ee	GPT

Disk /dev/sdaqo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqo1		1	2002	16080831+	ee	GPT

Disk /dev/sdaqp: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqp1		1	1403	11268031+	ee	GPT

Disk /dev/sdaqr: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqr1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdaqu: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaq1		1	2002	16080831+	ee	GPT

Disk /dev/sdarh: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarh1		1	1403	11268031+	ee	GPT

Disk /dev/sdaqw: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqw1		1	2002	16080831+	ee	GPT

Disk /dev/sdaqt: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaq1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdarg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarg1		1	1913	15364031+	ee	GPT

Disk /dev/sdaqx: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqx1		1	2002	16080831+	ee	GPT

Disk /dev/sdard: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdard1		1	1913	15364031+	ee	GPT

Disk /dev/sdare: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdare1		1	1913	15364031+	ee	GPT

Disk /dev/sdaqy: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqy1		1	2002	16080831+	ee	GPT

Disk /dev/sdarb: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarb1		1	2002	16080831+	ee	GPT

Disk /dev/sdarc: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarc1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdarf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarf1		1	1913	15364031+	ee	GPT

Disk /dev/sdaqv: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqv1		1	7522	60420031+	ee	GPT

Disk /dev/sdaqz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqz1		1	2002	16080831+	ee	GPT

Disk /dev/sdaqq: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqq1		1	2002	16080831+	ee	GPT

Disk /dev/sdari: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdari1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdara: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdara1		1	2002	16080831+	ee	GPT

Disk /dev/sdarj: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarj1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdaqz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaqz1		1	2002	16080831+	ee	GPT

Disk /dev/sdark: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdark1		1	7522	60420031+	ee	GPT

Disk /dev/sdarm: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarm1		1	2002	16080831+	ee	GPT

Disk /dev/sdarl: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarl1		1	2002	16080831+	ee	GPT

/dev/sdar1 1 2002 16080831+ ee GPT

Disk /dev/sdam: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdam1		1	2002	16080831+	ee	GPT

Disk /dev/sdaro: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaro1		1	2002	16080831+	ee	GPT

Disk /dev/sdarp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarp1		1	2002	16080831+	ee	GPT

Disk /dev/sdars: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdars1		1	1913	15364031+	ee	GPT

Disk /dev/sdart: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdart1		1	1913	15364031+	ee	GPT

Disk /dev/sdarv: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarv1		1	1403	11268031+	ee	GPT

Disk /dev/sdaru: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaru1		1	1913	15364031+	ee	GPT

Disk /dev/sdarq: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarq1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdarr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarr1		1	1913	15364031+	ee	GPT

Disk /dev/sdarw: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarw1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdarx: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdarx1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdary: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdary1		1	7522	60420031+	ee	GPT

Disk /dev/sdasc: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasc1		1	2002	16080831+	ee	GPT

Disk /dev/sdasb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasb1		1	2002	16080831+	ee	GPT

Disk /dev/sdaz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaz1		1	2002	16080831+	ee	GPT

Disk /dev/sdasa: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasa1		1	2002	16080831+	ee	GPT

Disk /dev/sdasd: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasd1		1	2002	16080831+	ee	GPT

Disk /dev/sdasf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasf1		1	1913	15364031+	ee	GPT

Disk /dev/sdash: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdash1		1	1913	15364031+	ee	GPT

Disk /dev/sdasg: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasg1		1	1913	15364031+	ee	GPT

Disk /dev/sdask: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdask1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdasi: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasi1		1	1913	15364031+	ee	GPT

Disk /dev/sdasj: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasj1		1	1403	11268031+	ee	GPT

Disk /dev/sdase: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdase1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdasl: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasl1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdasm: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasm1		1	7522	60420031+	ee	GPT

Disk /dev/sdasn: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasn1		1	2002	16080831+	ee	GPT

Disk /dev/sdaso: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaso1		1	2002	16080831+	ee	GPT

Disk /dev/sdasq: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasq1		1	1913	15364031+	ee	GPT

Disk /dev/sdass: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdass1		1	1913	15364031+	ee	GPT

Disk /dev/sdasr: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasr1		1	2002	16080831+	ee	GPT

Disk /dev/sdast: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdast1		1	2002	16080831+	ee	GPT

Disk /dev/sdasy: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasyl		1	1913	15364031+	ee	GPT

Disk /dev/sdasw: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaswl		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdasu: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasul		1	1913	15364031+	ee	GPT

Disk /dev/sdasx: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasxl		1	1977	15876031+	ee	GPT

Disk /dev/sdasp: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaspl		1	2002	16080831+	ee	GPT

Disk /dev/sdasv: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdasvl		1	1913	15364031+	ee	GPT

Disk /dev/sdatb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdatbl		1	1913	15364031+	ee	GPT

Disk /dev/sdasz: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaszl		1	2678	21508031+	ee	GPT

Disk /dev/sdata: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdata1		1	7522	60420031+	ee	GPT

Disk /dev/sdatc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdatcl		1	2002	16080831+	ee	GPT

Disk /dev/sdatg: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdatgl		1	2002	16080831+	ee	GPT

Disk /dev/sdath: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdath1		1	1403	11268031+	ee	GPT

Disk /dev/sdate: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdate1		1	2002	16080831+	ee	GPT

Disk /dev/sdatd: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdatd1		1	1913	15364031+	ee	GPT

Disk /dev/sdatf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdatf1		1	1913	15364031+	ee	GPT

Disk /dev/sdati: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdati1		1	2002	16080831+	ee	GPT

Disk /dev/sdatj: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdatj1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:



```

phys=(1023, 254, 63) logical=(905, 159, 22)
Disk /dev/sdatk: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatk1    1    2002  16080831+ ee GPT

Disk /dev/sdatm: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatm1    1    2002  16080831+ ee GPT

Disk /dev/sdatl: 3779 MB, 3779002368 bytes
255 heads, 63 sectors/track, 459 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatl1    1    460  3690431+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdato: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdato1    1    2002  16080831+ ee GPT

Disk /dev/sdatn: 61.9 GB, 61870112768 bytes
255 heads, 63 sectors/track, 7521 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatn1    1    7522  60420031+ ee GPT

Disk /dev/sdatr: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatr1    1    1913  15364031+ ee GPT

Disk /dev/sdatq: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatq1    1    2002  16080831+ ee GPT

Disk /dev/sdatp: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatp1    1    2002  16080831+ ee GPT

Disk /dev/sdati: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdati1    1    460  3690431+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdatj: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatj1    1    2002  16080831+ ee GPT

Disk /dev/sdatk: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatk1    1    1913  15364031+ ee GPT

Disk /dev/sdatl: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatl1    1    1913  15364031+ ee GPT

Disk /dev/sdatm: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatm1    1    1913  15364031+ ee GPT

Disk /dev/sdatn: 528 MB, 528416768 bytes
255 heads, 63 sectors/track, 64 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatn1    1    65  516031+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdatz: 16.3 GB, 16257056768 bytes
255 heads, 63 sectors/track, 1976 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatz1    1    1977  15876031+ ee GPT

Disk /dev/sdatw: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdatw1    1    1913  15364031+ ee GPT

Disk /dev/sdaaa: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

  Device Boot  Start  End  Blocks Id System
/dev/sdaaa1    1    2002  16080831+ ee GPT

```

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaul		1	1913	15364031+	ee	GPT

Disk /dev/sdaub: 22.0 GB, 22024224768 bytes

255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaub1		1	2678	21508031+	ee	GPT

Disk /dev/sdauc: 15.7 GB, 15732768768 bytes

255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauc1		1	1913	15364031+	ee	GPT

Disk /dev/sdaue: 15.7 GB, 15732768768 bytes

255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaue1		1	1913	15364031+	ee	GPT

Disk /dev/sdauf: 16.5 GB, 16466771968 bytes

255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauf1		1	2002	16080831+	ee	GPT

Disk /dev/sdaud: 61.9 GB, 61870112768 bytes

255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaud1		1	7522	60420031+	ee	GPT

Disk /dev/sdauh: 16.5 GB, 16466771968 bytes

255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauh1		1	2002	16080831+	ee	GPT

Disk /dev/sdaug: 11.5 GB, 11538464768 bytes

255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaug1		1	1403	11268031+	ee	GPT

Disk /dev/sdaui: 7449 MB, 7449018368 bytes

255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaui1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):

phys=(0, 0, 1) logical=(0, 0, 2)

Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdauj: 16.5 GB, 16466771968 bytes

255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauj1		1	2002	16080831+	ee	GPT

Disk /dev/sdauk: 3779 MB, 3779002368 bytes

255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauk1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):

phys=(0, 0, 1) logical=(0, 0, 2)

Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdaul: 16.5 GB, 16466771968 bytes

255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaul1		1	2002	16080831+	ee	GPT

Disk /dev/sdaum: 61.9 GB, 61870112768 bytes

255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaum1		1	7522	60420031+	ee	GPT

Disk /dev/sdaun: 16.5 GB, 16466771968 bytes

255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaun1		1	2002	16080831+	ee	GPT

Disk /dev/sdaur: 16.5 GB, 16466771968 bytes

255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaur1		1	2002	16080831+	ee	GPT

Disk /dev/sdauo: 16.5 GB, 16466771968 bytes

255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauo1		1	2002	16080831+	ee	GPT

Disk /dev/sdauq: 16.5 GB, 16466771968 bytes

255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauq1		1	2002	16080831+	ee	GPT

Disk /dev/sdaup: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaup1		1	2002	16080831+	ee	GPT

Disk /dev/sdaut: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaut1		1	2002	16080831+	ee	GPT

Disk /dev/sdauu: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauu1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdauw: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauw1		1	1913	15364031+	ee	GPT

Disk /dev/sdaus: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaus1		1	2002	16080831+	ee	GPT

Disk /dev/sdauv: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauv1		1	1913	15364031+	ee	GPT

Disk /dev/sdauy: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauy1		1	1913	15364031+	ee	GPT

Disk /dev/sdaux: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaux1		1	1913	15364031+	ee	GPT

Disk /dev/sdauz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdauz1		1	1913	15364031+	ee	GPT

Disk /dev/sdava: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdava1		1	1913	15364031+	ee	GPT

Disk /dev/sdavic: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdavic1		1	1913	15364031+	ee	GPT

Disk /dev/sdavid: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdavid1		1	2678	21508031+	ee	GPT

Disk /dev/sdavi: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdavi1		1	7522	60420031+	ee	GPT

Disk /dev/sdavi: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdavi1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdavn: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdavn1		1	1977	15876031+	ee	GPT

Disk /dev/sdave: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdave1		1	1913	15364031+	ee	GPT

Disk /dev/sdavi: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavl1 1 2002 16080831+ ee GPT

Disk /dev/sdavg: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavg1 1 1403 11268031+ ee GPT

Disk /dev/sdavn: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 460 3690431+ ee GPT  
Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdavn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 2002 16080831+ ee GPT

Disk /dev/sdavl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavl1 1 2002 16080831+ ee GPT

Disk /dev/sdavn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 2002 16080831+ ee GPT

Disk /dev/sdavn: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 7522 60420031+ ee GPT

Disk /dev/sdavo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavo1 1 2002 16080831+ ee GPT

Disk /dev/sdavn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 2002 16080831+ ee GPT

Disk /dev/sdavn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 2002 16080831+ ee GPT

Disk /dev/sdavn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 2002 16080831+ ee GPT

Disk /dev/sdavn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System

/dev/sdavn1 1 2002 16080831+ ee GPT

Disk /dev/sdavn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 1913 15364031+ ee GPT

Disk /dev/sdavn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 1913 15364031+ ee GPT

Disk /dev/sdavn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 2002 16080831+ ee GPT

Disk /dev/sdavn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 2002 16080831+ ee GPT

Disk /dev/sdavn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdavn1 1 1913 15364031+ ee GPT

Disk /dev/sdavn: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawal		1	1977	15876031+	ee	GPT

Disk /dev/sdawb: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawb1		1	1913	15364031+	ee	GPT

Disk /dev/sdavz: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdavz1		1	1913	15364031+	ee	GPT

Disk /dev/sdavy: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdavy1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdawd: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawd1		1	7522	60420031+	ee	GPT

Disk /dev/sdawc: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawc1		1	2678	21508031+	ee	GPT

Disk /dev/sdawe: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawe1		1	1913	15364031+	ee	GPT

Disk /dev/sdawg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawg1		1	1913	15364031+	ee	GPT

Disk /dev/sdawf: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawf1		1	2002	16080831+	ee	GPT

Disk /dev/sdawi: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawi1		1	1913	15364031+	ee	GPT

Disk /dev/sdawj: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawj1		1	2002	16080831+	ee	GPT

Disk /dev/sdawah: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawn1		1	2002	16080831+	ee	GPT

Disk /dev/sdawk: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawk1		1	1403	11268031+	ee	GPT

Disk /dev/sdawn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawn1		1	2002	16080831+	ee	GPT

Disk /dev/sdawnm: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawm1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdowo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdowo1		1	2002	16080831+	ee	GPT

Disk /dev/sdawp: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawp1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:

phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdawl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaw1		1	2002	16080831+	ee	GPT

Disk /dev/sdawq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawq1		1	1913	15364031+	ee	GPT

Disk /dev/sdaws: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaws1		1	1913	15364031+	ee	GPT

Disk /dev/sdawt: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawt1		1	2002	16080831+	ee	GPT

Disk /dev/sdawr: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawr1		1	7522	60420031+	ee	GPT

Disk /dev/sdawu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawu1		1	1913	15364031+	ee	GPT

Disk /dev/sdawy: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawy1		1	1913	15364031+	ee	GPT

Disk /dev/sdaww: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaww1		1	1913	15364031+	ee	GPT

Disk /dev/sdawx: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawx1		1	2002	16080831+	ee	GPT

Disk /dev/sdaxd: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaxd1		1	1913	15364031+	ee	GPT

Disk /dev/sdawv: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawv1		1	2002	16080831+	ee	GPT

Disk /dev/sdawz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdawz1		1	2002	16080831+	ee	GPT

Disk /dev/sdaxa: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaxa1		1	1403	11268031+	ee	GPT

Disk /dev/sdaxc: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaxc1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdaxb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaxb1		1	2002	16080831+	ee	GPT

Disk /dev/sdaxe: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaxe1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdaxf: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxfl 1 460 3690431+ ee GPT  
Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdaxh: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxhl 1 1913 15364031+ ee GPT

Disk /dev/sdaxj: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxjl 1 1913 15364031+ ee GPT

Disk /dev/sdaxg: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxgl 1 7522 60420031+ ee GPT

Disk /dev/sdaxi: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxil 1 2002 16080831+ ee GPT

Disk /dev/sdaxl: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System

/dev/sdaxll 1 1977 15876031+ ee GPT

Disk /dev/sdaxo: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxol 1 2678 21508031+ ee GPT

Disk /dev/sdaxk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxkl 1 2002 16080831+ ee GPT

Disk /dev/sdaxm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxml 1 2002 16080831+ ee GPT

Disk /dev/sdaxn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxnl 1 1913 15364031+ ee GPT

Disk /dev/sdaxq: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxql 1 7522 60420031+ ee GPT

Disk /dev/sdaxp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxpl 1 2002 16080831+ ee GPT

Disk /dev/sdaxs: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxsl 1 2002 16080831+ ee GPT

Disk /dev/sdaxr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxrl 1 1913 15364031+ ee GPT

Disk /dev/sdaxt: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxtl 1 2002 16080831+ ee GPT

Disk /dev/sdaxu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxul 1 1913 15364031+ ee GPT

Disk /dev/sdaxv: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
/dev/sdaxvl 1 2002 16080831+ ee GPT

Disk /dev/sdaxw: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	/dev/sdayc1 1 2002 16080831+ ee GPT	/dev/sdayh1 1 2002 16080831+ ee GPT
Device Boot Start End Blocks Id System /dev/sdaxw1 1 65 516031+ ee GPT Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(64, 61, 61)	Disk /dev/sdayb: 7449 MB, 7449018368 bytes 255 heads, 63 sectors/track, 905 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sdayg: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdaxx: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System /dev/sdayb1 1 906 7274431+ ee GPT Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(905, 159, 22)	Device Boot Start End Blocks Id System /dev/sdayg1 1 1913 15364031+ ee GPT
Device Boot Start End Blocks Id System /dev/sdaxx1 1 1913 15364031+ ee GPT	Disk /dev/sdayd: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sdayj: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdaxz: 11.5 GB, 11538464768 bytes 255 heads, 63 sectors/track, 1402 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System /dev/sdayd1 1 2002 16080831+ ee GPT	Device Boot Start End Blocks Id System /dev/sdayj1 1 2002 16080831+ ee GPT
Device Boot Start End Blocks Id System /dev/sdaxz1 1 1403 11268031+ ee GPT	Disk /dev/sdaye: 3779 MB, 3779002368 bytes 255 heads, 63 sectors/track, 459 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sdayi: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdaxy: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System /dev/sdaye1 1 460 3690431+ ee GPT Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(459, 111, 36)	Device Boot Start End Blocks Id System /dev/sdayi1 1 1913 15364031+ ee GPT
Device Boot Start End Blocks Id System /dev/sdaxy1 1 2002 16080831+ ee GPT	Disk /dev/sdayf: 61.9 GB, 61870112768 bytes 255 heads, 63 sectors/track, 7521 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sdayk: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdaya: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System /dev/sdayf1 1 7522 60420031+ ee GPT	Device Boot Start End Blocks Id System /dev/sdayk1 1 1913 15364031+ ee GPT
Device Boot Start End Blocks Id System /dev/sdaya1 1 2002 16080831+ ee GPT	Disk /dev/sdayh: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sdaym: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdayc: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System	Device Boot Start End Blocks Id System /dev/sdaym1 1 1913 15364031+ ee GPT
Device Boot Start End Blocks Id System	Device Boot Start End Blocks Id System	Disk /dev/sdayn: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000



Device	Boot	Start	End	Blocks	Id	System
/dev/sdayn1		1	2002	16080831+	ee	GPT

Disk /dev/sdayr: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayr1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdayl: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayl1		1	2002	16080831+	ee	GPT

Disk /dev/sdayp: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayp1		1	2002	16080831+	ee	GPT

Disk /dev/sdayq: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayq1		1	2678	21508031+	ee	GPT

Disk /dev/sdayo: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayo1		1	1977	15876031+	ee	GPT

Disk /dev/sdays: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdays1		1	7522	60420031+	ee	GPT

Disk /dev/sdayt: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayt1		1	2002	16080831+	ee	GPT

Disk /dev/sdayv: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayv1		1	2002	16080831+	ee	GPT

Disk /dev/sdayu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayu1		1	2002	16080831+	ee	GPT

Disk /dev/sdayw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayw1		1	2002	16080831+	ee	GPT

Disk /dev/sdayx: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayx1		1	2002	16080831+	ee	GPT

Disk /dev/sdaza: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdaza1		1	1913	15364031+	ee	GPT

Disk /dev/sdayz: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayz1		1	1913	15364031+	ee	GPT

Disk /dev/sdazb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdazb1		1	1913	15364031+	ee	GPT

Disk /dev/sdayy: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdayy1		1	2002	16080831+	ee	GPT

Disk /dev/sdazd: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdazd1		1	1977	15876031+	ee	GPT

Disk /dev/sdazc: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000



Disk /dev/sdazu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdazu1		1	2002	16080831+	ee	GPT

Disk /dev/sdazx: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdazx1		1	2002	16080831+	ee	GPT

Disk /dev/sdazw: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdazw1		1	1913	15364031+	ee	GPT

Disk /dev/sdazy: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdazy1		1	1913	15364031+	ee	GPT

Disk /dev/sdazz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdazz1		1	2002	16080831+	ee	GPT

Disk /dev/sdbaa: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbaa1		1	1913	15364031+	ee	GPT

Disk /dev/sdbab: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbab1		1	2002	16080831+	ee	GPT

Disk /dev/sdbad: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbad1		1	1403	11268031+	ee	GPT

Disk /dev/sdbac: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbac1		1	1913	15364031+	ee	GPT

Disk /dev/sdbaf: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbaf1		1	7522	60420031+	ee	GPT

Disk /dev/sdbah: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbah1		1	7522	60420031+	ee	GPT

Disk /dev/sdbae: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbae1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbai: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbai1		1	1913	15364031+	ee	GPT

Disk /dev/sdbaj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbaj1		1	2002	16080831+	ee	GPT

Disk /dev/sdbag: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbag1		1	1913	15364031+	ee	GPT

Disk /dev/sdbak: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbak1		1	1913	15364031+	ee	GPT

Disk /dev/sdbam: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbam1		1	1913	15364031+	ee	GPT

Disk /dev/sdbal: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbal1		1	2002	16080831+	ee	GPT

Disk /dev/sdban: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdban1		1	2002	16080831+	ee	GPT

Disk /dev/sdbap: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbap1		1	2002	16080831+	ee	GPT

Disk /dev/sdbao: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbao1		1	1977	15876031+	ee	GPT

Disk /dev/sdba: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdba1		1	2678	21508031+	ee	GPT

Disk /dev/sdbas: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbas1		1	2002	16080831+	ee	GPT

/dev/sdbas1 1 7522 60420031+ ee GPT

Disk /dev/sdbat: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
 /dev/sdbat1 1 65 516031+ ee GPT  
 Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbaw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbaw1		1	2002	16080831+	ee	GPT

Disk /dev/sdbau: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbau1		1	2002	16080831+	ee	GPT

Disk /dev/sdbar: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbar1		1	2002	16080831+	ee	GPT

Disk /dev/sdbav: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbav1		1	2002	16080831+	ee	GPT

Disk /dev/sdbax: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbax1		1	1913	15364031+	ee	GPT

Disk /dev/sdbay: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbay1		1	2002	16080831+	ee	GPT

Disk /dev/sdbaz: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbaz1		1	1913	15364031+	ee	GPT

Disk /dev/sdbbb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbb1		1	1913	15364031+	ee	GPT

Disk /dev/sdbba: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbba1		1	2002	16080831+	ee	GPT

Disk /dev/sdbbc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbc1		1	2002	16080831+	ee	GPT

Disk /dev/sdbbd: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbd1		1	1913	15364031+	ee	GPT

Disk /dev/sdbbe: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbe1		1	1403	11268031+	ee	GPT

Disk /dev/sdbbg: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbg1		1	7522	60420031+	ee	GPT

Disk /dev/sdbbf: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbf1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbbi: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbi1		1	2002	16080831+	ee	GPT

Disk /dev/sdbbh: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbh1		1	7522	60420031+	ee	GPT

Disk /dev/sdbbj: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbj1		1	2002	16080831+	ee	GPT

Disk /dev/sdbbk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbk1		1	2002	16080831+	ee	GPT

Disk /dev/sdbbl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbl1		1	2002	16080831+	ee	GPT

Disk /dev/sdbbm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbm1		1	2002	16080831+	ee	GPT

Disk /dev/sdbbo: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbo1		1	1913	15364031+	ee	GPT

Disk /dev/sdbbn: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbn1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbbp: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbp1		1	1913	15364031+	ee	GPT

Disk /dev/sdbbq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbq1		1	1913	15364031+	ee	GPT

Disk /dev/sdbbr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbr1		1	1913	15364031+	ee	GPT

Disk /dev/sdbbs: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbs1		1	1403	11268031+	ee	GPT

Disk /dev/sdbbt: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device Boot Start End Blocks Id System  
 /dev/sdbbt1 1 906 7274431+ ee GPT  
 Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbbv: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbv1		1	7522	60420031+	ee	GPT

Disk /dev/sdbbu: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbu1		1	7522	60420031+	ee	GPT

Disk /dev/sdbbw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbw1		1	2002	16080831+	ee	GPT

Disk /dev/sdbbz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbz1		1	2002	16080831+	ee	GPT

Disk /dev/sdbbx: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbbx1		1	2002	16080831+	ee	GPT

Disk /dev/sdbby: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbby1		1	2002	16080831+	ee	GPT

Disk /dev/sdbca: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbca1		1	2002	16080831+	ee	GPT

Disk /dev/sdbcb: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcb1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbcc: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcc1		1	1913	15364031+	ee	GPT

Disk /dev/sdbcd: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcd1		1	1913	15364031+	ee	GPT

Disk /dev/sdbcf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcf1		1	1913	15364031+	ee	GPT

Disk /dev/sdbce: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbce1		1	1913	15364031+	ee	GPT

Disk /dev/sdbcg: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcg1		1	1403	11268031+	ee	GPT

Disk /dev/sdbch: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbch1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbci: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbci1		1	7522	60420031+	ee	GPT

Disk /dev/sdbcj: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

/dev/sdbcj1 1 7522 60420031+ ee GPT

Disk /dev/sdbck: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbck1		1	2002	16080831+	ee	GPT

Disk /dev/sdbcn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcn1		1	2002	16080831+	ee	GPT

Disk /dev/sdbcm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcm1		1	2002	16080831+	ee	GPT

Disk /dev/sdbco: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbco1		1	2002	16080831+	ee	GPT

Disk /dev/sdbcl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcl1		1	2002	16080831+	ee	GPT

Disk /dev/sdbcp: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcp1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbcq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcq1		1	1913	15364031+	ee	GPT

Disk /dev/sdbct: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbct1		1	1913	15364031+	ee	GPT

Disk /dev/sdbcs: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcs1		1	1913	15364031+	ee	GPT

Disk /dev/sdbcr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcr1		1	1913	15364031+	ee	GPT

Disk /dev/sdbcu: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcu1		1	1403	11268031+	ee	GPT

Disk /dev/sdbcv: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcv1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbcw: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcw1		1	7522	60420031+	ee	GPT

Disk /dev/sdbcy: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcy1		1	2002	16080831+	ee	GPT

Disk /dev/sdbcz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcz1		1	2002	16080831+	ee	GPT

Disk /dev/sdbcx: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbcx1		1	7522	60420031+	ee	GPT

Disk /dev/sdbdb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbdb1		1	2002	16080831+	ee	GPT

Disk /dev/sdbdf: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbdf1		1	1913	15364031+	ee	GPT

Disk /dev/sdbdg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbdg1		1	1913	15364031+	ee	GPT

Disk /dev/sdbdd: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbdd1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbde: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbde1		1	1913	15364031+	ee	GPT

Disk /dev/sdbdi: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbdi1		1	1403	11268031+	ee	GPT

Disk /dev/sbdba: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdba1		1	2002	16080831+	ee	GPT

Disk /dev/sdbdc: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbdc1		1	2002	16080831+	ee	GPT

Disk /dev/sbdbh: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdbh1		1	1913	15364031+	ee	GPT

Disk /dev/sbdbn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdbn1		1	2002	16080831+	ee	GPT

Disk /dev/sbdbm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdbm1		1	2002	16080831+	ee	GPT

Disk /dev/sbdbj: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdbj1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sbdbk: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdbk1		1	7522	60420031+	ee	GPT

Disk /dev/sbdbo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdbo1		1	2002	16080831+	ee	GPT

Disk /dev/sbdbl: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdbl1		1	7522	60420031+	ee	GPT

Disk /dev/sbdbp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdbp1		1	2002	16080831+	ee	GPT

Disk /dev/sbdbt: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbdbt1		1	1913	15364031+	ee	GPT



```

Disk /dev/sdbds: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbds1          1    1913  15364031+ ee GPT

Disk /dev/sdbdr: 528 MB, 528416768 bytes
255 heads, 63 sectors/track, 64 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbdr1          1      65    516031+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
 phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbdq: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbdq1          1    2002  16080831+ ee GPT

Disk /dev/sdbdu: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sbdbu1          1    1913  15364031+ ee GPT

Disk /dev/sdbdv: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbdv1          1    1913  15364031+ ee GPT

Disk /dev/sdbdw: 11.5 GB, 11538464768 bytes
255 heads, 63 sectors/track, 1402 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes

    Device Boot      Start   End  Blocks  Id System

I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbdw1          1    1403  11268031+ ee GPT

Disk /dev/sdbdy: 61.9 GB, 61870112768 bytes
255 heads, 63 sectors/track, 7521 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sbdby1          1    7522  60420031+ ee GPT

Disk /dev/sdbdx: 7449 MB, 7449018368 bytes
255 heads, 63 sectors/track, 905 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbdx1          1     906   7274431+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
 phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbdz: 61.9 GB, 61870112768 bytes
255 heads, 63 sectors/track, 7521 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbdz1          1    7522  60420031+ ee GPT

Disk /dev/sdbdb: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbdb1          1    2002  16080831+ ee GPT

Disk /dev/sdbea: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbea1          1    2002  16080831+ ee GPT

Disk /dev/sdbec: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbec1          1    2002  16080831+ ee GPT

Disk /dev/sdbed: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbed1          1    2002  16080831+ ee GPT

Disk /dev/sdbef: 528 MB, 528416768 bytes
255 heads, 63 sectors/track, 64 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbef1          1      65    516031+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
 phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbee: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbee1          1    2002  16080831+ ee GPT

Disk /dev/sdbeg: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start   End  Blocks  Id System
/dev/sdbeg1          1    1913  15364031+ ee GPT

Disk /dev/sdbeh: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders

```

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbeh1		1	1913	15364031+	ee	GPT

Disk /dev/sdbei: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbei1		1	1913	15364031+	ee	GPT

Disk /dev/sdbej: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbei1		1	1913	15364031+	ee	GPT

Disk /dev/sdbek: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbek1		1	1403	11268031+	ee	GPT

Disk /dev/sdbem: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbem1		1	7522	60420031+	ee	GPT

Disk /dev/sdben: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdben1		1	7522	60420031+	ee	GPT

Disk /dev/sdbel: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbel1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbeo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbeo1		1	2002	16080831+	ee	GPT

Disk /dev/sdbep: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbep1		1	2002	16080831+	ee	GPT

Disk /dev/sdbeu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbeu1		1	1913	15364031+	ee	GPT

Disk /dev/sdbeq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbeq1		1	2002	16080831+	ee	GPT

Disk /dev/sdbew: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbew1		1	1913	15364031+	ee	GPT

Disk /dev/sdber: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdber1		1	2002	16080831+	ee	GPT

Disk /dev/sdbev: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbev1		1	1913	15364031+	ee	GPT

Disk /dev/sdbet: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbet1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbex: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbex1		1	1913	15364031+	ee	GPT

Disk /dev/sdbes: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------



phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdbfp: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfp1		1	7522	60420031+	ee	GPT

Disk /dev/sdbfo: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfo1		1	7522	60420031+	ee	GPT

Disk /dev/sdbfs: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfs1		1	2002	16080831+	ee	GPT

Disk /dev/sdbfr: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfr1		1	2002	16080831+	ee	GPT

Disk /dev/sdbfv: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfv1		1	1913	15364031+	ee	GPT

Disk /dev/sdbfq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfq1		1	2002	16080831+	ee	GPT

Disk /dev/sdbft: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbft1		1	2002	16080831+	ee	GPT

Disk /dev/sdbfu: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfu1		1	2002	16080831+	ee	GPT

Disk /dev/sdbfw: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfw1		1	1913	15364031+	ee	GPT

Disk /dev/sdbfy: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfy1		1	1913	15364031+	ee	GPT

Disk /dev/sdbfx: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfx1		1	1913	15364031+	ee	GPT

Disk /dev/sdbfz: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbfz1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdbga: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbga1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbgc: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgc1		1	7522	60420031+	ee	GPT

Disk /dev/sdbgb: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgb1		1	7522	60420031+	ee	GPT

Disk /dev/sdbge: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbge1		1	2002	16080831+	ee	GPT

Disk /dev/sdbgf: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgf1		1	2002	16080831+	ee	GPT

Disk /dev/sdbgg: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgg1		1	2002	16080831+	ee	GPT

Disk /dev/sdbgd: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgd1		1	2002	16080831+	ee	GPT

Disk /dev/sdbgi: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgi1		1	1913	15364031+	ee	GPT

Disk /dev/sdbgh: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgh1		1	2002	16080831+	ee	GPT

Disk /dev/sdbgj: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgj1		1	1913	15364031+	ee	GPT

Disk /dev/sdbgk: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgk1		1	1913	15364031+	ee	GPT

Disk /dev/sdbgl: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgl1		1	1913	15364031+	ee	GPT

Disk /dev/sdbgm: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgm1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdbgn: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgn1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbgr: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgr1		1	2002	16080831+	ee	GPT

Disk /dev/sdbgs: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgs1		1	2002	16080831+	ee	GPT

Disk /dev/sdbgt: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgt1		1	2002	16080831+	ee	GPT

Disk /dev/sdbgq: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgq1		1	2002	16080831+	ee	GPT

Disk /dev/sdbgp: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgp1		1	7522	60420031+	ee	GPT

Disk /dev/sdbgo: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbgo1		1	7522	60420031+	ee	GPT

Disk /dev/sdbgu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System

<p>/dev/sdbgu1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbgw: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbgw1</td> <td>1</td> <td>1913</td> <td>15364031+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbgx: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbgx1</td> <td>1</td> <td>1913</td> <td>15364031+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbgv: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbgv1</td> <td>1</td> <td>1913</td> <td>15364031+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbgy: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbgy1</td> <td>1</td> <td>1913</td> <td>15364031+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbgz: 7763 MB, 7763591168 bytes 255 heads, 63 sectors/track, 943 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbgz1</td> <td>1</td> <td>944</td> <td>7581631+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(943, 221, 46)</p> <p>Disk /dev/sdbhb: 61.9 GB, 61870112768 bytes 255 heads, 63 sectors/track, 7521 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p>	Device	Boot	Start	End	Blocks	Id	System	/dev/sdbgw1	1	1913	15364031+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbgx1	1	1913	15364031+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbgv1	1	1913	15364031+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbgy1	1	1913	15364031+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbgz1	1	944	7581631+	ee	GPT		<p>Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhb1</td> <td>1</td> <td>7522</td> <td>60420031+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbhc: 61.9 GB, 61870112768 bytes 255 heads, 63 sectors/track, 7521 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhc1</td> <td>1</td> <td>7522</td> <td>60420031+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbhd: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhd1</td> <td>1</td> <td>2002</td> <td>16080831+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbha: 7449 MB, 7449018368 bytes 255 heads, 63 sectors/track, 905 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbha1</td> <td>1</td> <td>906</td> <td>7274431+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(905, 159, 22)</p> <p>Disk /dev/sdbhe: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhe1</td> <td>1</td> <td>2002</td> <td>16080831+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbhi: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p>	Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhb1	1	7522	60420031+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhc1	1	7522	60420031+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhd1	1	2002	16080831+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbha1	1	906	7274431+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhe1	1	2002	16080831+	ee	GPT		<table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhi1</td> <td>1</td> <td>1913</td> <td>15364031+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbhj: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhj1</td> <td>1</td> <td>1913</td> <td>15364031+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbhf: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhf1</td> <td>1</td> <td>2002</td> <td>16080831+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbhg: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhg1</td> <td>1</td> <td>2002</td> <td>16080831+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbhh: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhh1</td> <td>1</td> <td>2002</td> <td>16080831+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbhk: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <table border="1"> <thead> <tr> <th>Device</th> <th>Boot</th> <th>Start</th> <th>End</th> <th>Blocks</th> <th>Id</th> <th>System</th> </tr> </thead> <tbody> <tr> <td>/dev/sdbhk1</td> <td>1</td> <td>1913</td> <td>15364031+</td> <td>ee</td> <td>GPT</td> <td></td> </tr> </tbody> </table> <p>Disk /dev/sdbhl: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes Disk identifier: 0x00000000</p>	Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhi1	1	1913	15364031+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhj1	1	1913	15364031+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhf1	1	2002	16080831+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhg1	1	2002	16080831+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhh1	1	2002	16080831+	ee	GPT		Device	Boot	Start	End	Blocks	Id	System	/dev/sdbhk1	1	1913	15364031+	ee	GPT	
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbgw1	1	1913	15364031+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbgx1	1	1913	15364031+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbgv1	1	1913	15364031+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbgy1	1	1913	15364031+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbgz1	1	944	7581631+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhb1	1	7522	60420031+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhc1	1	7522	60420031+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhd1	1	2002	16080831+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbha1	1	906	7274431+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhe1	1	2002	16080831+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhi1	1	1913	15364031+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhj1	1	1913	15364031+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhf1	1	2002	16080831+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhg1	1	2002	16080831+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhh1	1	2002	16080831+	ee	GPT																																																																																																																																																																																																																													
Device	Boot	Start	End	Blocks	Id	System																																																																																																																																																																																																																												
/dev/sdbhk1	1	1913	15364031+	ee	GPT																																																																																																																																																																																																																													

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhl1		1	1913	15364031+	ee	GPT

Disk /dev/sdbhn: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhn1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbhm: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhm1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdbho: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbho1		1	7522	60420031+	ee	GPT

Disk /dev/sdbhp: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhp1		1	7522	60420031+	ee	GPT

Disk /dev/sdbhq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhq1		1	2002	16080831+	ee	GPT

Disk /dev/sdbhr: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhr1		1	2002	16080831+	ee	GPT

Disk /dev/sdbhs: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhs1		1	2002	16080831+	ee	GPT

Disk /dev/sdbhv: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhv1		1	1913	15364031+	ee	GPT

Disk /dev/sdbhu: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhu1		1	2002	16080831+	ee	GPT

Disk /dev/sdbhw: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhw1		1	1913	15364031+	ee	GPT

Disk /dev/sdbhx: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhx1		1	1913	15364031+	ee	GPT

Disk /dev/sdbht: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbht1		1	2002	16080831+	ee	GPT

Disk /dev/sdbhy: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbhy1		1	1913	15364031+	ee	GPT

Disk /dev/sdbib: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbib1		1	7522	60420031+	ee	GPT

Disk /dev/sdbia: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbia1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbid: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

<p>Device Boot Start End Blocks Id System /dev/sdbid1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbhz: 7763 MB, 7763591168 bytes 255 heads, 63 sectors/track, 943 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbhz1 1 944 7581631+ ee GPT Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(943, 221, 46)</p> <p>Disk /dev/sdbic: 61.9 GB, 61870112768 bytes 255 heads, 63 sectors/track, 7521 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbic1 1 7522 60420031+ ee GPT</p> <p>Disk /dev/sdbie: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbie1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbif: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbif1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbig: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbig1 1 2002 16080831+ ee GPT</p>	<p>Disk /dev/sdbii: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbii1 1 1913 15364031+ ee GPT</p> <p>Disk /dev/sdbij: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbij1 1 1913 15364031+ ee GPT</p> <p>Disk /dev/sdbih: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbih1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbik: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbik1 1 1913 15364031+ ee GPT</p> <p>Disk /dev/sdbim: 11.5 GB, 11538464768 bytes 255 heads, 63 sectors/track, 1402 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbim1 1 1403 11268031+ ee GPT</p> <p>Disk /dev/sdbil: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbil1 1 1913 15364031+ ee GPT</p>	<p>/dev/sdbil 1 1913 15364031+ ee GPT</p> <p>Disk /dev/sdbin: 7449 MB, 7449018368 bytes 255 heads, 63 sectors/track, 905 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbin1 1 906 7274431+ ee GPT Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(905, 159, 22)</p> <p>Disk /dev/sdbio: 3779 MB, 3779002368 bytes 255 heads, 63 sectors/track, 459 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbio1 1 460 3690431+ ee GPT Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(459, 111, 36)</p> <p>Disk /dev/sdbip: 61.9 GB, 61870112768 bytes 255 heads, 63 sectors/track, 7521 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbip1 1 7522 60420031+ ee GPT</p> <p>Disk /dev/sdbiq: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbiq1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbir: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbir1 1 2002 16080831+ ee GPT</p>
--	---	---



<p>/dev/sdbir1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbis: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbis1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbit: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbit1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbiu: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbiu1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbiv: 528 MB, 528416768 bytes 255 heads, 63 sectors/track, 64 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbiv1 1 65 516031+ ee GPT</p> <p>Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(64, 61, 61)</p> <p>Disk /dev/sdbiw: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbiw1 1 1913 15364031+ ee GPT</p> <p>Disk /dev/sdbix: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders</p>	<p>Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbix1 1 1913 15364031+ ee GPT</p> <p>Disk /dev/sdbiy: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbiy1 1 1913 15364031+ ee GPT</p> <p>Disk /dev/sdbiz: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbiz1 1 1913 15364031+ ee GPT</p> <p>Disk /dev/sdbja: 11.5 GB, 11538464768 bytes 255 heads, 63 sectors/track, 1402 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbja1 1 1403 11268031+ ee GPT</p> <p>Disk /dev/sdbjb: 7449 MB, 7449018368 bytes 255 heads, 63 sectors/track, 905 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbjb1 1 906 7274431+ ee GPT</p> <p>Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(905, 159, 22)</p> <p>Disk /dev/sdbjd: 61.9 GB, 61870112768 bytes 255 heads, 63 sectors/track, 7521 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p>	<p>Device Boot Start End Blocks Id System /dev/sdbjd1 1 7522 60420031+ ee GPT</p> <p>Disk /dev/sdbjc: 3779 MB, 3779002368 bytes 255 heads, 63 sectors/track, 459 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbjc1 1 460 3690431+ ee GPT</p> <p>Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(459, 111, 36)</p> <p>Disk /dev/sdbje: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbje1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbjk: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbjk1 1 1913 15364031+ ee GPT</p> <p>Disk /dev/sdbjf: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbjf1 1 2002 16080831+ ee GPT</p> <p>Disk /dev/sdbjh: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000</p> <p>Device Boot Start End Blocks Id System /dev/sdbjh1 1 2002 16080831+ ee GPT</p>
--	---	---

Disk /dev/sdbjg: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjg1		1	2002	16080831+	ee	GPT

Disk /dev/sdbjj: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjj1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbji: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbji1		1	2002	16080831+	ee	GPT

Disk /dev/sdbjm: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjm1		1	1913	15364031+	ee	GPT

Disk /dev/sdbjl: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjl1		1	1913	15364031+	ee	GPT

Disk /dev/sdbjo: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjo1		1	1403	11268031+	ee	GPT

Disk /dev/sdbjq: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjq1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdbjp: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjp1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbjr: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjr1		1	7522	60420031+	ee	GPT

Disk /dev/sdbjn: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjn1		1	1913	15364031+	ee	GPT

Disk /dev/sdbjs: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjs1		1	2002	16080831+	ee	GPT

Disk /dev/sdbjt: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjt1		1	2002	16080831+	ee	GPT

Disk /dev/sdbju: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbju1		1	2002	16080831+	ee	GPT

Disk /dev/sdbjz: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjz1		1	1913	15364031+	ee	GPT

Disk /dev/sdbjw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjw1		1	2002	16080831+	ee	GPT

Disk /dev/sdbka: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbka1		1	1913	15364031+	ee	GPT

Disk /dev/sdbjy: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjy1		1	1913	15364031+	ee	GPT

Disk /dev/sdbjv: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjv1		1	2002	16080831+	ee	GPT

Disk /dev/sdbjx: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbjx1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbkb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkb1		1	1913	15364031+	ee	GPT

Disk /dev/sdbkd: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkd1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbkc: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkc1		1	1403	11268031+	ee	GPT

Disk /dev/sdbkh: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkh1		1	2002	16080831+	ee	GPT

Disk /dev/sdbke: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbke1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdbkg: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkg1		1	2002	16080831+	ee	GPT

Disk /dev/sdbkf: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkf1		1	7522	60420031+	ee	GPT

Disk /dev/sdbki: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbki1		1	2002	16080831+	ee	GPT

Disk /dev/sdbkl: 528 MB, 528416768 bytes  
 255 heads, 63 sectors/track, 64 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkl1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbko: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbko1		1	1913	15364031+	ee	GPT

Disk /dev/sdbkm: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkm1		1	1913	15364031+	ee	GPT

Disk /dev/sdbkq: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkq1		1	1403	11268031+	ee	GPT

Disk /dev/sdbkj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkj1		1	2002	16080831+	ee	GPT

Disk /dev/sdbkn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkn1		1	1913	15364031+	ee	GPT

Disk /dev/sdbkp: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkp1		1	1913	15364031+	ee	GPT

Disk /dev/sdbkk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkk1		1	2002	16080831+	ee	GPT

Disk /dev/sdbkr: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkr1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbks: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbks1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdbku: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbku1		1	2002	16080831+	ee	GPT

Disk /dev/sdbkt: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkt1		1	7522	60420031+	ee	GPT

Disk /dev/sdbkz: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkz1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdblb: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdblb1		1	1913	15364031+	ee	GPT

Disk /dev/sdblx: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdblx1		1	2002	16080831+	ee	GPT

Disk /dev/sdblc: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdblc1		1	1913	15364031+	ee	GPT

Disk /dev/sdbla: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbla1		1	1913	15364031+	ee	GPT

Disk /dev/sdbkw: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkw1		1	2002	16080831+	ee	GPT

Disk /dev/sdbkv: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbkv1		1	2002	16080831+	ee	GPT

Disk /dev/sdbld: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbld1		1	1913	15364031+	ee	GPT

Disk /dev/sdbky: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbky1		1	2002	16080831+	ee	GPT

Disk /dev/sdblc: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	1403	11268031+	ee	GPT

Disk /dev/sdb1: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	2002	16080831+	ee	GPT

Disk /dev/sdb1: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdb1: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdb1: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	7522	60420031+	ee	GPT

Disk /dev/sdb1: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	2002	16080831+	ee	GPT

Disk /dev/sdb1: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	2002	16080831+	ee	GPT

Disk /dev/sdb1: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	1913	15364031+	ee	GPT

Disk /dev/sdb1: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	2002	16080831+	ee	GPT

Disk /dev/sdb1: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	2002	16080831+	ee	GPT

Disk /dev/sdb1: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	1913	15364031+	ee	GPT

Disk /dev/sdb1: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdb1: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdb1: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	1913	15364031+	ee	GPT

Disk /dev/sdb1: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	1403	11268031+	ee	GPT

Disk /dev/sdb1: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdb1		1	7522	60420031+	ee	GPT

```

Disk /dev/sdblt: 7449 MB, 7449018368 bytes
255 heads, 63 sectors/track, 905 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdblt1          1         906     7274431+  ee  GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
 phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbllq: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbllq1        1        1913     15364031+ ee  GPT

Disk /dev/sdbll: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbll1         1         2002     16080831+ ee  GPT

Disk /dev/sdbly: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbly1         1         2002     16080831+ ee  GPT

Disk /dev/sdblx: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdblx1         1         2002     16080831+ ee  GPT

Disk /dev/sdbma: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbma1         1         2002     16080831+ ee  GPT

I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbma1         1         2002     16080831+ ee  GPT

Disk /dev/sdbmc: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbmc1         1        1913     15364031+ ee  GPT

Disk /dev/sdblz: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdblz1         1         2002     16080831+ ee  GPT

Disk /dev/sdbmb: 528 MB, 528416768 bytes
255 heads, 63 sectors/track, 64 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbmb1         1          65       516031+  ee  GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
 phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbmd: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbmd1         1        1913     15364031+ ee  GPT

Disk /dev/sdbme: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbme1         1        1913     15364031+ ee  GPT

Disk /dev/sdbmf: 15.7 GB, 15732768768 bytes
255 heads, 63 sectors/track, 1912 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbmf1         1        1913     15364031+ ee  GPT

Disk /dev/sdbmg: 11.5 GB, 11538464768 bytes
255 heads, 63 sectors/track, 1402 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbmg1         1        1403     11268031+ ee  GPT

Disk /dev/sdbmh: 7449 MB, 7449018368 bytes
255 heads, 63 sectors/track, 905 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbmh1         1         906     7274431+  ee  GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
 phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbmp: 528 MB, 528416768 bytes
255 heads, 63 sectors/track, 64 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbmp1         1          65       516031+  ee  GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
 phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
 phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbmm: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

    Device Boot      Start         End      Blocks   Id  System
/dev/sdbmm1         1        2002     16080831+ ee  GPT

```

/dev/sdbmm1 1 2002 16080831+ ee GPT

Disk /dev/sdbmo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmo1		1	2002	16080831+	ee	GPT

Disk /dev/sdbmk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmk1		1	2002	16080831+	ee	GPT

Disk /dev/sdbmn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmn1		1	2002	16080831+	ee	GPT

Disk /dev/sdbml: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbml1		1	2002	16080831+	ee	GPT

Disk /dev/sdbmj: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmj1		1	7522	60420031+	ee	GPT

Disk /dev/sdbmi: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmi1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdbmq: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmq1		1	1785	14337983+	ee	GPT

Disk /dev/sdbmr: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmr1		1	1785	14337983+	ee	GPT

Disk /dev/sdbms: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbms1		1	1785	14337983+	ee	GPT

Disk /dev/sdbmt: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmt1		1	1785	14337983+	ee	GPT

Disk /dev/sdbmv: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmv1		1	1785	14337983+	ee	GPT

Disk /dev/sdbmw: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmw1		1	1785	14337983+	ee	GPT

Disk /dev/sdbmu: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmu1		1	1785	14337983+	ee	GPT

Disk /dev/sdbmx: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbmx1		1	1785	14337983+	ee	GPT

Disk /dev/sdbna: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbna1		1	1785	14337983+	ee	GPT

Disk /dev/sdbnz: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbnz1		1	1785	14337983+	ee	GPT

Disk /dev/sdbnb: 14.7 GB, 14682095616 bytes  
255 heads, 63 sectors/track, 1784 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------























I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtb1		1	1913	15364031+	ee	GPT

Disk /dev/sdbtc: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtc1		1	1913	15364031+	ee	GPT

Disk /dev/sdbtd: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtd1		1	1913	15364031+	ee	GPT

Disk /dev/sdbte: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbte1		1	1977	15876031+	ee	GPT

Disk /dev/sdbtf: 22.0 GB, 22024224768 bytes  
255 heads, 63 sectors/track, 2677 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtf1		1	2678	21508031+	ee	GPT

Disk /dev/sdbtg: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtg1		1	7522	60420031+	ee	GPT

Disk /dev/sdbth: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbth1		1	2002	16080831+	ee	GPT

Disk /dev/sdbti: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbti1		1	2002	16080831+	ee	GPT

Disk /dev/sdbtj: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtj1		1	2002	16080831+	ee	GPT

Disk /dev/sdbtk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtk1		1	2002	16080831+	ee	GPT

Disk /dev/sdbtl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtl1		1	2002	16080831+	ee	GPT

Disk /dev/sdbtm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtm1		1	2002	16080831+	ee	GPT

Disk /dev/sdbtn: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtn1		1	1913	15364031+	ee	GPT

Disk /dev/sdbto: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbto1		1	1913	15364031+	ee	GPT

Disk /dev/sdbtp: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtp1		1	1913	15364031+	ee	GPT

Disk /dev/sdbtq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtq1		1	1913	15364031+	ee	GPT

Disk /dev/sdbtr: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbtr1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdbts: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	/dev/sdbtx1 1 2002 16080831+ ee GPT	Device Boot Start End Blocks Id System /dev/sdbud1 1 1913 15364031+ ee GPT
Device Boot Start End Blocks Id System /dev/sdbts1 1 906 7274431+ ee GPT Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(905, 159, 22)	Disk /dev/sdbty: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sdbue: 7763 MB, 7763591168 bytes 255 heads, 63 sectors/track, 943 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdbtt: 61.9 GB, 61870112768 bytes 255 heads, 63 sectors/track, 7521 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System /dev/sdbtyl 1 2002 16080831+ ee GPT	Device Boot Start End Blocks Id System /dev/sdbue1 1 944 7581631+ ee GPT Partition 1 has different physical/logical beginnings (non-Linux?): phys=(0, 0, 1) logical=(0, 0, 2) Partition 1 has different physical/logical endings: phys=(1023, 254, 63) logical=(943, 221, 46)
Device Boot Start End Blocks Id System /dev/sdbtt1 1 7522 60420031+ ee GPT	Disk /dev/sdbtz: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sdbuf: 22.0 GB, 22024224768 bytes 255 heads, 63 sectors/track, 2677 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdbtu: 61.9 GB, 61870112768 bytes 255 heads, 63 sectors/track, 7521 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System /dev/sdbtz1 1 2002 16080831+ ee GPT	Device Boot Start End Blocks Id System /dev/sdbuf1 1 2678 21508031+ ee GPT
Device Boot Start End Blocks Id System /dev/sdbtu1 1 7522 60420031+ ee GPT	Disk /dev/sdbua: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sdbug: 61.9 GB, 61870112768 bytes 255 heads, 63 sectors/track, 7521 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdbtv: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System /dev/sdbua1 1 1913 15364031+ ee GPT	Device Boot Start End Blocks Id System /dev/sdbug1 1 7522 60420031+ ee GPT
Device Boot Start End Blocks Id System /dev/sdbtv1 1 2002 16080831+ ee GPT	Disk /dev/sdbub: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sbuh: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdbtw: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System /dev/sdbub1 1 1913 15364031+ ee GPT	Device Boot Start End Blocks Id System /dev/sbuh1 1 2002 16080831+ ee GPT
Device Boot Start End Blocks Id System /dev/sdbtw1 1 2002 16080831+ ee GPT	Disk /dev/sdbuc: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sbui: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Disk /dev/sdbtx: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Device Boot Start End Blocks Id System /dev/sdbuc1 1 1913 15364031+ ee GPT	Device Boot Start End Blocks Id System /dev/sbui1 1 2002 16080831+ ee GPT
Device Boot Start End Blocks Id System	Disk /dev/sdbud: 15.7 GB, 15732768768 bytes 255 heads, 63 sectors/track, 1912 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000	Disk /dev/sbuj: 16.5 GB, 16466771968 bytes 255 heads, 63 sectors/track, 2001 cylinders Units = cylinders of 16065 * 512 = 8225280 bytes Sector size (logical/physical): 512 bytes / 512 bytes I/O size (minimum/optimal): 512 bytes / 512 bytes Disk identifier: 0x00000000
Device Boot Start End Blocks Id System	Device Boot Start End Blocks Id System /dev/sbuj1 1 2002 16080831+ ee GPT	

Disk /dev/sdbuj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbuj1		1	2002	16080831+	ee	GPT

Disk /dev/sdbuk: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbuk1		1	2002	16080831+	ee	GPT

Disk /dev/sdbul: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbul1		1	2002	16080831+	ee	GPT

Disk /dev/sdbum: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbum1		1	2002	16080831+	ee	GPT

Disk /dev/sdbun: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbun1		1	1913	15364031+	ee	GPT

Disk /dev/sdbuo: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbuo1		1	1913	15364031+	ee	GPT

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbuo1		1	1913	15364031+	ee	GPT

Disk /dev/sdbup: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbup1		1	1913	15364031+	ee	GPT

Disk /dev/sdbuq: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbuq1		1	1913	15364031+	ee	GPT

Disk /dev/sdbur: 7763 MB, 7763591168 bytes  
 255 heads, 63 sectors/track, 943 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbur1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdbus: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbus1		1	2678	21508031+	ee	GPT

Disk /dev/sdbut: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sbut1		1	7522	60420031+	ee	GPT

Disk /dev/sdbuu: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbuu1		1	2002	16080831+	ee	GPT

Disk /dev/sdbuv: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbuv1		1	2002	16080831+	ee	GPT

Disk /dev/sdbuw: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbuw1		1	2002	16080831+	ee	GPT

Disk /dev/sdbux: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbux1		1	2002	16080831+	ee	GPT

Disk /dev/sdbuy: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbuy1		1	2002	16080831+	ee	GPT

Disk /dev/sdbuz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbuz1		1	2002	16080831+	ee	GPT

Disk /dev/sdbva: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbva1		1	1913	15364031+	ee	GPT

Disk /dev/sdbvb: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvb1		1	1913	15364031+	ee	GPT

Disk /dev/sdbvc: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvc1		1	1913	15364031+	ee	GPT

Disk /dev/sdbvd: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvd1		1	1913	15364031+	ee	GPT

Disk /dev/sdbve: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbve1		1	1403	11268031+	ee	GPT

Disk /dev/sdbvf: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvf1		1	1977	15876031+	ee	GPT

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvf1		1	1977	15876031+	ee	GPT

Disk /dev/sdbvg: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvg1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdbvh: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvh1		1	7522	60420031+	ee	GPT

Disk /dev/sdbvi: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvi1		1	2002	16080831+	ee	GPT

Disk /dev/sdbvj: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvj1		1	2002	16080831+	ee	GPT

Disk /dev/sdbvk: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvk1		1	2002	16080831+	ee	GPT

Disk /dev/sdbvl: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvl1		1	2002	16080831+	ee	GPT

Disk /dev/sdbvm: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvm1		1	2002	16080831+	ee	GPT

Disk /dev/sdbvn: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvn1		1	2002	16080831+	ee	GPT

Disk /dev/sdbvo: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvo1		1	1913	15364031+	ee	GPT

Disk /dev/sdbvp: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvp1		1	1913	15364031+	ee	GPT

Disk /dev/sdbvq: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvq1		1	1913	15364031+	ee	GPT

Disk /dev/sdbvr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvr1		1	1913	15364031+	ee	GPT

Disk /dev/sdbvs: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvs1		1	1403	11268031+	ee	GPT

Disk /dev/sdbvt: 16.3 GB, 16257056768 bytes  
255 heads, 63 sectors/track, 1976 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvt1		1	1977	15876031+	ee	GPT

Disk /dev/sdbvu: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvu1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdbvv: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvv1		1	7522	60420031+	ee	GPT

Disk /dev/sdbvw: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvw1		1	2002	16080831+	ee	GPT

Disk /dev/sdbvx: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvx1		1	2002	16080831+	ee	GPT

Disk /dev/sdbvy: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvy1		1	2002	16080831+	ee	GPT

Disk /dev/sdbvz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbvz1		1	2002	16080831+	ee	GPT

Disk /dev/sdbwa: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwa1		1	2002	16080831+	ee	GPT

Disk /dev/sdbwb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwb1		1	2002	16080831+	ee	GPT

Disk /dev/sdbwc: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwc1		1	1913	15364031+	ee	GPT

Disk /dev/sdbwd: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwd1		1	1913	15364031+	ee	GPT

Disk /dev/sdbwe: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwe1		1	1913	15364031+	ee	GPT

Disk /dev/sdbwf: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwf1		1	1913	15364031+	ee	GPT

Disk /dev/sdbwg: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwg1		1	1403	11268031+	ee	GPT

Disk /dev/sdbwh: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwh1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):

phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbwi: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwi1	1	7522	60420031+	ee	GPT	

Disk /dev/sdbwj: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwj1	1	7522	60420031+	ee	GPT	

Disk /dev/sdbwk: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwk1	1	2002	16080831+	ee	GPT	

Disk /dev/sdbwl: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwl1	1	2002	16080831+	ee	GPT	

Disk /dev/sdbwm: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwm1	1	2002	16080831+	ee	GPT	

Disk /dev/sdbwn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwn1	1	2002	16080831+	ee	GPT	

Disk /dev/sdbwo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwo1	1	2002	16080831+	ee	GPT	

Disk /dev/sdbwp: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwp1	1	65	516031+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbwq: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwq1	1	1913	15364031+	ee	GPT	

Disk /dev/sdbwr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwr1	1	1913	15364031+	ee	GPT	

Disk /dev/sdbws: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbws1	1	1913	15364031+	ee	GPT	

Disk /dev/sdbwt: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwt1	1	1913	15364031+	ee	GPT	

Disk /dev/sdbwu: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwu1	1	1403	11268031+	ee	GPT	

Disk /dev/sdbwv: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwv1	1	906	7274431+	ee	GPT	

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbww: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbww1	1	7522	60420031+	ee	GPT	

Disk /dev/sdbwx: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwx1	1	7522	60420031+	ee	GPT	

Disk /dev/sdbwy: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwy1		1	2002	16080831+	ee	GPT

Disk /dev/sdbwz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbwz1		1	2002	16080831+	ee	GPT

Disk /dev/sdbxa: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxa1		1	2002	16080831+	ee	GPT

Disk /dev/sdbxb: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxb1		1	2002	16080831+	ee	GPT

Disk /dev/sdbxc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxc1		1	2002	16080831+	ee	GPT

Disk /dev/sdbxd: 55 MB, 55508992 bytes  
 255 heads, 63 sectors/track, 6 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxd1		1	7	54207+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):

phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(6, 190, 56)

Disk /dev/sdbxe: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxe1		1	1913	15364031+	ee	GPT

Disk /dev/sdbxf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxf1		1	1913	15364031+	ee	GPT

Disk /dev/sdbxg: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxg1		1	1913	15364031+	ee	GPT

Disk /dev/sdbxh: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxh1		1	1913	15364031+	ee	GPT

Disk /dev/sdbxi: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxi1		1	1403	11268031+	ee	GPT

Disk /dev/sdbxj: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxj1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbxk: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxk1		1	7522	60420031+	ee	GPT

Disk /dev/sdbxl: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxl1		1	7522	60420031+	ee	GPT

Disk /dev/sdbxm: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxm1		1	2002	16080831+	ee	GPT

Disk /dev/sdbxn: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxn1		1	2002	16080831+	ee	GPT

Disk /dev/sdbxo: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

/dev/sdbxo1 1 2002 16080831+ ee GPT

Disk /dev/sdbxp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxp1		1	2002	16080831+	ee	GPT

Disk /dev/sdbxq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxq1		1	2002	16080831+	ee	GPT

Disk /dev/sdbxr: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxr1		1	1913	15364031+	ee	GPT

Disk /dev/sdbxs: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxs1		1	1913	15364031+	ee	GPT

Disk /dev/sdbxt: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxt1		1	1913	15364031+	ee	GPT

Disk /dev/sdbxu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxu1		1	1913	15364031+	ee	GPT

Disk /dev/sdbxv: 7763 MB, 7763591168 bytes  
255 heads, 63 sectors/track, 943 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxv1		1	944	7581631+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(943, 221, 46)

Disk /dev/sdbxw: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxw1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbxx: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxx1		1	7522	60420031+	ee	GPT

Disk /dev/sdbxy: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxy1		1	7522	60420031+	ee	GPT

Disk /dev/sdbxz: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbxz1		1	2002	16080831+	ee	GPT

Disk /dev/sdbya: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbya1		1	2002	16080831+	ee	GPT

Disk /dev/sdbyb: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyb1		1	2002	16080831+	ee	GPT

Disk /dev/sdbyc: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyc1		1	2002	16080831+	ee	GPT

Disk /dev/sdbyd: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyd1		1	2002	16080831+	ee	GPT

Disk /dev/sdbye: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbye1		1	1913	15364031+	ee	GPT

Disk /dev/sdbyf: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes



I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyfl		1	1913	15364031+	ee	GPT

Disk /dev/sdbyg: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyg1		1	1913	15364031+	ee	GPT

Disk /dev/sdbyh: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyh1		1	1913	15364031+	ee	GPT

Disk /dev/sdbyi: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyi1		1	1403	11268031+	ee	GPT

Disk /dev/sdbyj: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyj1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbyk: 3779 MB, 3779002368 bytes  
255 heads, 63 sectors/track, 459 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

/dev/sdbyk1 1 460 3690431+ ee GPT  
Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdbyl: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyl1		1	7522	60420031+	ee	GPT

Disk /dev/sdbym: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbym1		1	2002	16080831+	ee	GPT

Disk /dev/sdbyn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyn1		1	2002	16080831+	ee	GPT

Disk /dev/sdbyo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyo1		1	2002	16080831+	ee	GPT

Disk /dev/sdbyp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyp1		1	2002	16080831+	ee	GPT

Disk /dev/sdbyq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyq1		1	2002	16080831+	ee	GPT

Disk /dev/sdbyr: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyr1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbys: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbys1		1	1913	15364031+	ee	GPT

Disk /dev/sdbyt: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyt1		1	1913	15364031+	ee	GPT

Disk /dev/sdbyu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyu1		1	1913	15364031+	ee	GPT

Disk /dev/sdbyv: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyv1		1	1913	15364031+	ee	GPT

Disk /dev/sdbyw: 16.3 GB, 16257056768 bytes  
 255 heads, 63 sectors/track, 1976 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyw1		1	1977	15876031+	ee	GPT

Disk /dev/sdbyx: 22.0 GB, 22024224768 bytes  
 255 heads, 63 sectors/track, 2677 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyx1		1	2678	21508031+	ee	GPT

Disk /dev/sdbyy: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyy1		1	7522	60420031+	ee	GPT

Disk /dev/sdbyz: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbyz1		1	2002	16080831+	ee	GPT

Disk /dev/sdbza: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbza1		1	2002	16080831+	ee	GPT

Disk /dev/sdbzb: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzb1		1	2002	16080831+	ee	GPT

Disk /dev/sdbzc: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzc1		1	2002	16080831+	ee	GPT

Disk /dev/sdbzd: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzd1		1	2002	16080831+	ee	GPT

Disk /dev/sdbze: 16.5 GB, 16466771968 bytes  
 255 heads, 63 sectors/track, 2001 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbze1		1	2002	16080831+	ee	GPT

Disk /dev/sdbzf: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzf1		1	1913	15364031+	ee	GPT

Disk /dev/sdbzg: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzg1		1	1913	15364031+	ee	GPT

Disk /dev/sdbzh: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzh1		1	1913	15364031+	ee	GPT

Disk /dev/sdbzi: 15.7 GB, 15732768768 bytes  
 255 heads, 63 sectors/track, 1912 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzi1		1	1913	15364031+	ee	GPT

Disk /dev/sdbzj: 11.5 GB, 11538464768 bytes  
 255 heads, 63 sectors/track, 1402 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzj1		1	1403	11268031+	ee	GPT

Disk /dev/sdbzk: 7449 MB, 7449018368 bytes  
 255 heads, 63 sectors/track, 905 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzk1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbzl: 3779 MB, 3779002368 bytes  
 255 heads, 63 sectors/track, 459 cylinders  
 Units = cylinders of 16065 \* 512 = 8225280 bytes  
 Sector size (logical/physical): 512 bytes / 512 bytes  
 I/O size (minimum/optimal): 512 bytes / 512 bytes  
 Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzl1		1	460	3690431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
 phys=(0, 0, 1) logical=(0, 0, 2)  
 Partition 1 has different physical/logical endings:  
 phys=(1023, 254, 63) logical=(459, 111, 36)

Disk /dev/sdbzm: 61.9 GB, 61870112768 bytes  
 255 heads, 63 sectors/track, 7521 cylinders

Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzm1		1	7522	60420031+	ee	GPT

Disk /dev/sdbzn: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzn1		1	2002	16080831+	ee	GPT

Disk /dev/sdbzo: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzo1		1	2002	16080831+	ee	GPT

Disk /dev/sdbzp: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzp1		1	2002	16080831+	ee	GPT

Disk /dev/sdbzq: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzq1		1	2002	16080831+	ee	GPT

Disk /dev/sdbzr: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzr1		1	2002	16080831+	ee	GPT

Disk /dev/sdbzs: 528 MB, 528416768 bytes  
255 heads, 63 sectors/track, 64 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzs1		1	65	516031+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(64, 61, 61)

Disk /dev/sdbzt: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzt1		1	1913	15364031+	ee	GPT

Disk /dev/sdbzu: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzu1		1	1913	15364031+	ee	GPT

Disk /dev/sdbzv: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzv1		1	1913	15364031+	ee	GPT

Disk /dev/sdbzw: 15.7 GB, 15732768768 bytes  
255 heads, 63 sectors/track, 1912 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzw1		1	1913	15364031+	ee	GPT

Disk /dev/sdbzx: 11.5 GB, 11538464768 bytes  
255 heads, 63 sectors/track, 1402 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes

I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzx1		1	1403	11268031+	ee	GPT

Disk /dev/sdbzy: 7449 MB, 7449018368 bytes  
255 heads, 63 sectors/track, 905 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzy1		1	906	7274431+	ee	GPT

Partition 1 has different physical/logical beginnings (non-Linux?):  
phys=(0, 0, 1) logical=(0, 0, 2)  
Partition 1 has different physical/logical endings:  
phys=(1023, 254, 63) logical=(905, 159, 22)

Disk /dev/sdbzz: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdbzz1		1	7522	60420031+	ee	GPT

Disk /dev/sdcaa: 61.9 GB, 61870112768 bytes  
255 heads, 63 sectors/track, 7521 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcaa1		1	7522	60420031+	ee	GPT

Disk /dev/sdcab: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
/dev/sdcab1		1	2002	16080831+	ee	GPT

Disk /dev/sdcac: 16.5 GB, 16466771968 bytes  
255 heads, 63 sectors/track, 2001 cylinders  
Units = cylinders of 16065 \* 512 = 8225280 bytes  
Sector size (logical/physical): 512 bytes / 512 bytes  
I/O size (minimum/optimal): 512 bytes / 512 bytes  
Disk identifier: 0x00000000

Device	Boot	Start	End	Blocks	Id	System
--------	------	-------	-----	--------	----	--------

```

/dev/sdcac1      1      2002  16080831+ ee GPT
Disk /dev/sdcad: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

   Device Boot      Start   End  Blocks  Id System
/dev/sdcad1        1      2002  16080831+ ee GPT

Disk /dev/sdcae: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

   Device Boot      Start   End  Blocks  Id System
/dev/sdcae1        1      2002  16080831+ ee GPT

Disk /dev/sdcaf: 16.5 GB, 16466771968 bytes
255 heads, 63 sectors/track, 2001 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

   Device Boot      Start   End  Blocks  Id System
/dev/sdcaf1        1      2002  16080831+ ee GPT

Disk /dev/sdcag: 528 MB, 528416768 bytes
255 heads, 63 sectors/track, 64 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes
Sector size (logical/physical): 512 bytes / 512 bytes
I/O size (minimum/optimal): 512 bytes / 512 bytes
Disk identifier: 0x00000000

   Device Boot      Start   End  Blocks  Id System
/dev/sdcag1        1        65   516031+ ee GPT
Partition 1 has different physical/logical beginnings (non-Linux?):
 phys=(0, 0, 1) logical=(0, 0, 2)
Partition 1 has different physical/logical endings:
 phys=(1023, 254, 63) logical=(64, 61, 61)

scsi_id.out
scsi_id -g -d for every /dev/*

/dev/sda 3600605b0016298e0ff0000460454aeb8
/dev/sda1 3600605b0016298e0ff0000460454aeb8
/dev/sda2 3600605b0016298e0ff0000460454aeb8
/dev/sda3 3600605b0016298e0ff0000460454aeb8
/dev/sdaa 3600144f054c0f0000000000a28d218
/dev/sdaaa 3600144f019554e0000000000d04d267
/dev/sdaab 3600144f0e18d8e0000000000d01d258
/dev/sdaac 3600144f0c4e70e0000000000d05d216
/dev/sdach 3600144f0e18d8e0000000000d01d161
/dev/sdaci 3600144f0c4e70e0000000000d05d230
/dev/sdaj 3600144f0e18d8e0000000000d01d175
/dev/sdack 3600144f0ed35ce0000000000d07d127
/dev/sdacl 3600144f0c4e70e0000000000d05d141
/dev/sdacm 3600144f0e18d8e0000000000d01d189
/dev/sdacn 3600144f0ed35ce0000000000d07d141
/dev/sdaco 3600144f0e18d8e0000000000d01d203
/dev/sdacp 3600144f0c4e70e0000000000d05d258
/dev/sdacq 3600144f0ed35ce0000000000d07d155
/dev/sdacr 3600144f0c4e70e0000000000d05d272
/dev/sdacs 3600144f0e18d8e0000000000d01d217
/dev/sdact 3600144f0c4e70e0000000000d05d286
/dev/sdacu 3600144f0e18d8e0000000000d01d231
/dev/sdacr 3600144f0ed35ce000000000d07d169
/dev/sdacw 3600144f0e18d8e0000000000d01d245
/dev/sdacr 3600144f0ed35ce000000000d07d183
/dev/sdacy 3600144f0e18d8e0000000000d01d259
/dev/sdacz 3600144f0ed35ce000000000d07d197
/dev/sdad 3600144f0054c0f0000000000a28d221
/dev/sdada 3600144f0e18d8e0000000000d01d273
/dev/sdadb 3600144f0ed35ce0000000000d07d211
/dev/sdadc 3600144f0e18d8e0000000000d01d287
/dev/sdadd 3600144f09c8d8e000000000d02d104
/dev/sdade 3600144f0ed35ce000000000d07d225
/dev/sdadf 3600144f0ed35ce000000000d07d239
/dev/sdadg 3600144f09c8d8e000000000d02d118
/dev/sdadh 3600144f0ed35ce000000000d07d253
/dev/sdadi 3600144f09c8d8e000000000d02d132
/dev/sdadj 3600144f0ed35ce000000000d07d267
/dev/sdadk 3600144f09c8d8e000000000d01d146
/dev/sdadl 3600144f0ed35ce000000000d07d281
/dev/sdadm 3600144f09c8d8e000000000d02d160
/dev/sdadn 3600144f09c8d8e000000000d02d174
/dev/sdado 3600144f078f08e000000000d08d113
/dev/sdadp 3600144f09c8d8e000000000d02d188
/dev/sdadq 3600144f09c8d8e000000000d02d202
/dev/sdadr 3600144f078f08e000000000d08d127
/dev/sdads 3600144f09c8d8e000000000d02d216
/dev/sdadt 3600144f078f08e000000000d08d141
/dev/sdadu 3600144f09c8d8e000000000d02d230
/dev/sdadv 3600144f078f08e000000000d08d155
/dev/sdadw 3600144f09c8d8e000000000d02d244
/dev/sdadx 3600144f078f08e000000000d08d169
/dev/sdady 3600144f09c8d8e000000000d02d258
/dev/sdadz 3600144f078f08e000000000d08d183
/dev/sdae 3600144f0054c0f0000000000a28d222
/dev/sdaea 3600144f09c8d8e000000000d02d272
/dev/sdaeb 3600144f078f08e000000000d08d197
/dev/sdaec 3600144f09c8d8e000000000d02d286
/dev/sdaed 3600144f078f08e000000000d08d211
/dev/sdaee 3600144f078f08e000000000d08d225
/dev/sdaef 3600144f0ed35ce000000000d07d104
/dev/sdaeg 3600144f078f08e000000000d08d239
/dev/sdae 3600144f0ed35ce000000000d07d118
/dev/sdaei 3600144f078f08e000000000d08d253
/dev/sdaej 3600144f0ed35ce000000000d07d132

```

/dev/sdaek 3600144f078f08e0000000000d08d267  
/dev/sdael 3600144f078f08e0000000000d08d281  
/dev/sdaem 3600144f0ed35ce0000000000d07d146  
/dev/sdaen 3600144f0ed35ce0000000000d07d160  
/dev/sdaeo 3600144f08129ce0000000000d09d113  
/dev/sdaep 3600144f0ed35ce0000000000d07d174  
/dev/sdaeq 3600144f0ed35ce0000000000d07d188  
/dev/sdaer 3600144f08129ce0000000000d09d127  
/dev/sdaes 3600144f0ed35ce0000000000d07d202  
/dev/sdaet 3600144f0ed35ce0000000000d07d216  
/dev/sdaeu 3600144f08129ce0000000000d09d141  
/dev/sdaev 3600144f0ed35ce0000000000d07d230  
/dev/sdaew 3600144f08129ce0000000000d09d155  
/dev/sdaex 3600144f0ed35ce0000000000d07d244  
/dev/sdaey 3600144f08129ce0000000000d09d169  
/dev/sdaez 3600144f0ed35ce0000000000d07d258  
/dev/sdaf 3600144f0054c0f0000000000a28d223  
/dev/sdafa 3600144f08129ce0000000000d09d183  
/dev/sdafb 3600144f0ed35ce0000000000d07d272  
/dev/sdafc 3600144f08129ce0000000000d09d197  
/dev/sdafd 3600144f0ed35ce0000000000d07d286  
/dev/sdafe 3600144f08129ce0000000000d09d211  
/dev/sdaf f 3600144f08129ce0000000000d09d225  
/dev/sdafg 3600144f08129ce0000000000d06d104  
/dev/sdafh 3600144f08129ce0000000000d09d239  
/dev/sdafi 3600144f08452ce0000000000d06d118  
/dev/sdafj 3600144f08129ce0000000000d09d253  
/dev/sdafk 3600144f08129ce0000000000d09d267  
/dev/sdafl 3600144f08452ce0000000000d06d132  
/dev/sdafm 3600144f08129ce0000000000d09d281  
/dev/sdafn 3600144f08452ce0000000000d06d146  
/dev/sdaf o 3600144f08452ce0000000000d06d160  
/dev/sdafp 3600144f059274e000000000d10d113  
/dev/sdafq 3600144f08452ce0000000000d06d174  
/dev/sdaf r 3600144f08452ce0000000000d06d188  
/dev/sdaf s 3600144f059274e000000000d10d127  
/dev/sdaf t 3600144f08452ce0000000000d06d202  
/dev/sdaf u 3600144f059274e000000000d10d141  
/dev/sdaf v 3600144f059274e000000000d10d155  
/dev/sdaf w 3600144f08452ce0000000000d06d216  
/dev/sdaf x 3600144f059274e000000000d10d169  
/dev/sdaf y 3600144f08452ce0000000000d06d230  
/dev/sdaf z 3600144f059274e000000000d10d183  
/dev/sdag 3600144f0054c0f0000000000a28d224  
/dev/sdaga 3600144f08452ce0000000000d06d244  
/dev/sdagb 3600144f059274e000000000d10d197  
/dev/sdagc 3600144f08452ce0000000000d06d258  
/dev/sdagd 3600144f059274e000000000d10d211  
/dev/sdage 3600144f08452ce0000000000d06d272  
/dev/sdagf 3600144f059274e000000000d10d225  
/dev/sdag g 3600144f08452ce0000000000d06d286  
/dev/sdag h 3600144f059274e000000000d10d239  
/dev/sdagi 3600144f059274e000000000d10d253  
/dev/sdag j 3600144f059274e000000000d10d267  
/dev/sdag k 3600144f059274e000000000d10d281  
/dev/sdag l 3600144f07d498e000000000d47d104  
/dev/sdag m 3600144f08452ce0000000000d06d113  
/dev/sdagn 3600144f07d498e000000000d47d118

/dev/sdago 3600144f08452ce0000000000d06d127  
/dev/sdagp 3600144f07d498e000000000d47d132  
/dev/sdagq 3600144f08452ce0000000000d06d141  
/dev/sdagr 3600144f07d498e000000000d47d146  
/dev/sdags 3600144f08452ce0000000000d06d155  
/dev/sdagt 3600144f08452ce0000000000d06d169  
/dev/sdagu 3600144f07d498e000000000d47d160  
/dev/sdagv 3600144f08452ce0000000000d06d183  
/dev/sdagw 3600144f07d498e000000000d47d174  
/dev/sdagx 3600144f08452ce0000000000d06d197  
/dev/sdagy 3600144f07d498e000000000d47d188  
/dev/sdagz 3600144f08452ce0000000000d06d211  
/dev/sdaha 3600144f0054c0f000000000a28d225  
/dev/sdahb 3600144f07d498e000000000d47d202  
/dev/sdahc 3600144f08452ce0000000000d06d225  
/dev/sdahd 3600144f07d498e000000000d47d216  
/dev/sdah e 3600144f07d498e000000000d47d230  
/dev/sdahf 3600144f08452ce0000000000d06d253  
/dev/sdahg 3600144f07d498e000000000d47d244  
/dev/sdahh 3600144f08452ce0000000000d06d267  
/dev/sdahi 3600144f07d498e000000000d47d258  
/dev/sdah j 3600144f08452ce0000000000d06d281  
/dev/sdahk 3600144f07d498e000000000d47d272  
/dev/sdah l 3600144f07d498e000000000d47d286  
/dev/sdahm 3600144f08129ce000000000d09d104  
/dev/sdah n 3600144f08129ce000000000d09d118  
/dev/sdaho 3600144f08129ce000000000d09d132  
/dev/sdahp 3600144f08129ce000000000d09d146  
/dev/sdahq 3600144f08129ce000000000d09d160  
/dev/sdahr 3600144f08129ce000000000d09d174  
/dev/sdahs 3600144f08129ce000000000d09d188  
/dev/sdah t 3600144f08129ce000000000d09d202  
/dev/sdah u 3600144f08129ce000000000d09d216  
/dev/sdah v 3600144f08129ce000000000d09d230  
/dev/sdah w 3600144f08129ce000000000d09d244  
/dev/sdah x 3600144f08129ce000000000d09d258  
/dev/sdah y 3600144f08129ce000000000d09d272  
/dev/sdah z 3600144f08129ce000000000d09d286  
/dev/sdai 3600144f0054c0f000000000a28d226  
/dev/sdaia 3600144f059274e000000000d10d104  
/dev/sdaib 3600144f059274e000000000d10d118  
/dev/sdaic 3600144f059274e000000000d10d132  
/dev/sdaid 3600144f059274e000000000d10d146  
/dev/sdaie 3600144f059274e000000000d10d160  
/dev/sdaif 3600144f059274e000000000d10d174  
/dev/sdaig 3600144f059274e000000000d10d188  
/dev/sdaih 3600144f059274e000000000d10d202  
/dev/sdaii 3600144f059274e000000000d10d216  
/dev/sdaij 3600144f059274e000000000d10d230  
/dev/sdaik 3600144f059274e000000000d10d244  
/dev/sdail 3600144f059274e000000000d10d258  
/dev/sdaim 3600144f059274e000000000d10d272  
/dev/sdain 3600144f059274e000000000d10d286  
/dev/sdaio 3600144f019554e000000000d04d110  
/dev/sdaip 3600144f019554e000000000d04d124  
/dev/sdaiq 3600144f019554e000000000d04d138  
/dev/sdair 3600144f019554e000000000d04d152

/dev/sdais 3600144f019554e000000000d04d166  
/dev/sdait 3600144f019554e000000000d04d180  
/dev/sdaiu 3600144f019554e000000000d04d194  
/dev/sdaiv 3600144f019554e000000000d04d208  
/dev/sdaiw 3600144f019554e000000000d04d222  
/dev/sdaix 3600144f019554e000000000d04d236  
/dev/sdaiy 3600144f019554e000000000d04d250  
/dev/sdaiz 3600144f019554e000000000d04d264  
/dev/sdaj 3600144f0054c0f000000000a28d227  
/dev/sdaja 3600144f019554e000000000d04d278  
/dev/sdajb 3600144f0e18d8e000000000d01d110  
/dev/sdajc 3600144f0e18d8e000000000d01d124  
/dev/sdajd 3600144f0e18d8e000000000d01d138  
/dev/sdaje 3600144f0e18d8e000000000d01d152  
/dev/sdajf 3600144f0e18d8e000000000d01d166  
/dev/sdajg 3600144f0e18d8e000000000d01d180  
/dev/sdajh 3600144f0e18d8e000000000d01d194  
/dev/sdaji 3600144f0e18d8e000000000d01d208  
/dev/sdajj 3600144f0e18d8e000000000d01d222  
/dev/sdajk 3600144f0e18d8e000000000d01d236  
/dev/sdajl 3600144f0e18d8e000000000d01d250  
/dev/sdajm 3600144f0e18d8e000000000d01d264  
/dev/sdajn 3600144f0e18d8e000000000d01d278  
/dev/sdajo 3600144f07d498e000000000d47d110  
/dev/sdajp 3600144f07d498e000000000d47d124  
/dev/sdajq 3600144f07d498e000000000d47d138  
/dev/sdajr 3600144f07d498e000000000d47d152  
/dev/sdajs 3600144f07d498e000000000d47d166  
/dev/sdajt 3600144f07d498e000000000d47d180  
/dev/sdaju 3600144f07d498e000000000d47d194  
/dev/sdajv 3600144f07d498e000000000d47d208  
/dev/sdajw 3600144f07d498e000000000d47d222  
/dev/sdajx 3600144f07d498e000000000d47d236  
/dev/sdajy 3600144f07d498e000000000d47d250  
/dev/sdajz 3600144f07d498e000000000d47d264  
/dev/sdak 3600144f0054c0f000000000a28d228  
/dev/sdakb 3600144f0c4e70e000000000d05d110  
/dev/sdakc 3600144f0c4e70e000000000d05d124  
/dev/sdakd 3600144f0c4e70e000000000d05d138  
/dev/sdake 3600144f0c4e70e000000000d05d152  
/dev/sdakf 3600144f0c4e70e000000000d05d166  
/dev/sdakg 3600144f0c4e70e000000000d05d180  
/dev/sdakh 3600144f0c4e70e000000000d05d194  
/dev/sdaki 3600144f0c4e70e000000000d05d208  
/dev/sdakj 3600144f0c4e70e000000000d05d222  
/dev/sdakk 3600144f0c4e70e000000000d05d236  
/dev/sdakl 3600144f0c4e70e000000000d05d250  
/dev/sdakm 3600144f0c4e70e000000000d05d264  
/dev/sdakn 3600144f0c4e70e000000000d05d278  
/dev/sdako 3600144f09c8d8e000000000d02d110  
/dev/sdakp 3600144f09c8d8e000000000d02d124  
/dev/sdakq 3600144f09c8d8e000000000d02d138  
/dev/sdakr 3600144f09c8d8e000000000d02d152  
/dev/sdaks 3600144f09c8d8e000000000d02d166  
/dev/sdakt 3600144f09c8d8e000000000d02d180  
/dev/sdaku 3600144f09c8d8e000000000d02d194  
/dev/sdakov 3600144f09c8d8e000000000d02d208

/dev/sdakw 3600144f09c8d8e0000000000d02d222  
/dev/sdakx 3600144f09c8d8e0000000000d02d236  
/dev/sdaky 3600144f09c8d8e0000000000d02d250  
/dev/sdakz 3600144f07d498e0000000000d47d111  
/dev/sdal 3600144f0054c0f0000000000a28d229  
/dev/sdala 3600144f09c8d8e0000000000d02d264  
/dev/sdalb 3600144f09c8d8e0000000000d02d278  
/dev/sdalc 3600144f07d498e0000000000d47d125  
/dev/sdald 3600144f07d498e0000000000d47d139  
/dev/sdale 3600144f07d498e0000000000d47d153  
/dev/sdalf 3600144f07d498e0000000000d47d167  
/dev/sdalg 3600144f07d498e0000000000d47d181  
/dev/sdalh 3600144f07d498e0000000000d47d195  
/dev/sdali 3600144f07d498e0000000000d47d209  
/dev/sdalj 3600144f07d498e0000000000d47d223  
/dev/sdalk 3600144f07d498e0000000000d47d237  
/dev/sdall 3600144f078f08e0000000000d08d110  
/dev/sdalm 3600144f07d498e0000000000d47d251  
/dev/sdaln 3600144f07d498e0000000000d47d265  
/dev/sdalo 3600144f078f08e0000000000d08d124  
/dev/sdalp 3600144f07d498e0000000000d47d279  
/dev/sdalq 3600144f078f08e0000000000d08d138  
/dev/sdalar 3600144f078f08e0000000000d08d152  
/dev/sdals 3600144f078f08e0000000000d08d166  
/dev/sdalt 3600144f019554e0000000000d04d111  
/dev/sdalu 3600144f078f08e0000000000d08d180  
/dev/sdalv 3600144f019554e0000000000d04d125  
/dev/sdalw 3600144f078f08e0000000000d08d194  
/dev/sdalx 3600144f019554e0000000000d04d139  
/dev/sdaly 3600144f078f08e0000000000d08d208  
/dev/sdalz 3600144f019554e0000000000d04d153  
/dev/sdam 3600144f0054c0f0000000000a28d230  
/dev/sdama 3600144f078f08e0000000000d08d222  
/dev/sdamb 3600144f019554e0000000000d04d167  
/dev/sdamc 3600144f078f08e0000000000d08d236  
/dev/sdamd 3600144f019554e0000000000d04d181  
/dev/sdame 3600144f078f08e0000000000d08d250  
/dev/sdamf 3600144f019554e0000000000d04d195  
/dev/sdamg 3600144f078f08e0000000000d08d264  
/dev/sdamh 3600144f019554e0000000000d04d209  
/dev/sdami 3600144f078f08e0000000000d08d278  
/dev/sdamj 3600144f019554e0000000000d04d223  
/dev/sdamk 3600144f019554e0000000000d04d237  
/dev/sdaml 3600144f019554e0000000000d04d251  
/dev/sdamm 3600144f019554e0000000000d04d265  
/dev/sdamn 3600144f019554e0000000000d04d279  
/dev/sdamo 3600144f059274e0000000000d10d110  
/dev/sdamp 3600144f0e18d8e0000000000d01d111  
/dev/sdamq 3600144f059274e0000000000d10d124  
/dev/sdamr 3600144f0e18d8e0000000000d01d125  
/dev/sdams 3600144f059274e0000000000d10d138  
/dev/sdamt 3600144f0e18d8e0000000000d01d139  
/dev/sdamu 3600144f059274e0000000000d10d152  
/dev/sdamv 3600144f0e18d8e0000000000d01d153  
/dev/sdamw 3600144f059274e0000000000d10d166  
/dev/sdamx 3600144f0e18d8e0000000000d01d167  
/dev/sdamy 3600144f059274e0000000000d10d180  
/dev/sdamz 3600144f0e18d8e0000000000d01d181

/dev/sdan 3600144f0054c0f0000000000a28d231  
/dev/sdana 3600144f059274e0000000000d10d194  
/dev/sdanb 3600144f0e18d8e0000000000d01d195  
/dev/sdanc 3600144f059274e0000000000d10d208  
/dev/sdand 3600144f0e18d8e0000000000d01d209  
/dev/sdane 3600144f059274e0000000000d10d222  
/dev/sdanf 3600144f0e18d8e0000000000d01d223  
/dev/sdang 3600144f059274e0000000000d10d236  
/dev/sdanh 3600144f0e18d8e0000000000d01d237  
/dev/sdani 3600144f059274e0000000000d10d250  
/dev/sdanj 3600144f0e18d8e0000000000d01d251  
/dev/sdank 3600144f059274e0000000000d10d264  
/dev/sdanl 3600144f0e18d8e0000000000d01d265  
/dev/sdann 3600144f059274e0000000000d10d278  
/dev/sdano 3600144f0c4e70e0000000000d05d111  
/dev/sdanp 3600144f0c4e70e0000000000d05d125  
/dev/sdanq 3600144f0c4e70e0000000000d05d139  
/dev/sdanr 3600144f0ed35ce0000000000d07d110  
/dev/sdans 3600144f0c4e70e0000000000d05d153  
/dev/sdant 3600144f0c4e70e0000000000d05d167  
/dev/sdanu 3600144f0ed35ce0000000000d07d124  
/dev/sdanv 3600144f0c4e70e0000000000d05d181  
/dev/sdanw 3600144f0ed35ce0000000000d07d138  
/dev/sdanx 3600144f0ed35ce0000000000d07d152  
/dev/sdany 3600144f0c4e70e0000000000d05d195  
/dev/sdanz 3600144f0c4e70e0000000000d05d209  
/dev/sdao 3600144f0054c0f0000000000a28d232  
/dev/sdaoa 3600144f0ed35ce0000000000d07d166  
/dev/sdaob 3600144f0c4e70e0000000000d05d223  
/dev/sdaoc 3600144f0ed35ce0000000000d07d180  
/dev/sdaod 3600144f0c4e70e0000000000d05d237  
/dev/sdaoe 3600144f0ed35ce0000000000d07d194  
/dev/sdaof 3600144f0c4e70e0000000000d05d251  
/dev/sdaog 3600144f0ed35ce0000000000d07d208  
/dev/sdaoh 3600144f0c4e70e0000000000d05d265  
/dev/sdaoi 3600144f0ed35ce0000000000d07d222  
/dev/sdaoj 3600144f0c4e70e0000000000d05d279  
/dev/sdaok 3600144f0ed35ce0000000000d07d236  
/dev/sdaol 3600144f0ed35ce0000000000d07d250  
/dev/sdaom 3600144f0ed35ce0000000000d07d264  
/dev/sdaon 3600144f08452ce0000000000d06d111  
/dev/sdaoo 3600144f0ed35ce0000000000d07d278  
/dev/sdaop 3600144f0ed35ce0000000000d07d112  
/dev/sdaoq 3600144f08452ce0000000000d06d125  
/dev/sdaor 3600144f08129ce0000000000d09d110  
/dev/sdaos 3600144f0ed35ce0000000000d07d126  
/dev/sdaot 3600144f08452ce0000000000d06d139  
/dev/sdaou 3600144f08129ce0000000000d09d124  
/dev/sdaov 3600144f08452ce0000000000d06d153  
/dev/sdaow 3600144f0ed35ce0000000000d07d140  
/dev/sdaox 3600144f08129ce0000000000d09d138  
/dev/sdaoy 3600144f08452ce0000000000d06d167  
/dev/sdaoz 3600144f0ed35ce0000000000d07d154  
/dev/sdap 3600144f0054c0f0000000000a28d233  
/dev/sdapa 3600144f08129ce0000000000d09d152  
/dev/sdapb 3600144f08452ce0000000000d06d181  
/dev/sdapc 3600144f0ed35ce0000000000d07d168

/dev/sdapd 3600144f08452ce0000000000d06d195  
/dev/sdape 3600144f08129ce0000000000d09d166  
/dev/sdapf 3600144f0ed35ce0000000000d07d182  
/dev/sdapg 3600144f08129ce0000000000d09d180  
/dev/sdaph 3600144f08452ce0000000000d06d209  
/dev/sdapi 3600144f0ed35ce0000000000d07d196  
/dev/sdapij 3600144f08129ce0000000000d09d194  
/dev/sdapk 3600144f08452ce0000000000d06d223  
/dev/sdapl 3600144f0ed35ce0000000000d07d210  
/dev/sdapl 3600144f08129ce0000000000d09d208  
/dev/sdapl 3600144f08452ce0000000000d06d237  
/dev/sdapo 3600144f0ed35ce0000000000d07d224  
/dev/sdapp 3600144f08129ce0000000000d09d222  
/dev/sdapp 3600144f08452ce0000000000d06d251  
/dev/sdapr 3600144f0ed35ce0000000000d07d238  
/dev/sdaps 3600144f08129ce0000000000d09d236  
/dev/sdapt 3600144f08452ce0000000000d06d265  
/dev/sdapu 3600144f0ed35ce0000000000d07d252  
/dev/sdapv 3600144f08129ce0000000000d09d250  
/dev/sdapw 3600144f08452ce0000000000d06d279  
/dev/sdapx 3600144f0ed35ce0000000000d07d266  
/dev/sdapy 3600144f08129ce0000000000d09d264  
/dev/sdapz 3600144f09c8d8e0000000000d02d111  
/dev/sdaq 3600144f0054c0f0000000000a28d234  
/dev/sdaqa 3600144f0ed35ce0000000000d07d280  
/dev/sdaqb 3600144f08129ce0000000000d09d278  
/dev/sdaqc 3600144f09c8d8e0000000000d02d125  
/dev/sdaqd 3600144f08452ce0000000000d06d110  
/dev/sdaqe 3600144f08452ce0000000000d06d124  
/dev/sdaqf 3600144f09c8d8e0000000000d02d139  
/dev/sdaqg 3600144f0e18d8e0000000000d01d112  
/dev/sdaqh 3600144f08452ce0000000000d06d138  
/dev/sdaqj 3600144f09c8d8e0000000000d02d153  
/dev/sdaqk 3600144f09c8d8e0000000000d02d167  
/dev/sdaql 3600144f0e18d8e0000000000d01d140  
/dev/sdaqm 3600144f08452ce0000000000d06d152  
/dev/sdaqn 3600144f0e18d8e0000000000d01d154  
/dev/sdaqo 3600144f09c8d8e0000000000d02d181  
/dev/sdaqp 3600144f08452ce0000000000d06d166  
/dev/sdaqq 3600144f0e18d8e0000000000d01d168  
/dev/sdaqr 3600144f08452ce0000000000d06d180  
/dev/sdaqs 3600144f09c8d8e0000000000d02d195  
/dev/sdaq 3600144f0e18d8e0000000000d01d182  
/dev/sdaq 3600144f08452ce0000000000d06d194  
/dev/sdaqv 3600144f09c8d8e0000000000d02d209  
/dev/sdaqw 3600144f0e18d8e0000000000d01d196  
/dev/sdaqx 3600144f08452ce0000000000d06d208  
/dev/sdaqy 3600144f09c8d8e0000000000d02d223  
/dev/sdaqz 3600144f0e18d8e0000000000d01d210  
/dev/sdar 3600144f0054c0f0000000000a28d235  
/dev/sdara 3600144f08452ce0000000000d06d222  
/dev/sdarb 3600144f09c8d8e0000000000d02d237  
/dev/sdarc 3600144f0e18d8e0000000000d01d224  
/dev/sdard 3600144f08452ce0000000000d06d236  
/dev/sdare 3600144f09c8d8e0000000000d02d251  
/dev/sdarf 3600144f0e18d8e0000000000d01d238  
/dev/sdarg 3600144f08452ce0000000000d06d250

/dev/sdarh 3600144f09c8d8e0000000000d02d265  
/dev/sdari 3600144f0e18d8e0000000000d01d252  
/dev/sdarj 3600144f08452ce0000000000d06d264  
/dev/sdark 3600144f09c8d8e0000000000d02d279  
/dev/sdarl 3600144f0e18d8e0000000000d01d266  
/dev/sdarm 3600144f08452ce0000000000d06d278  
/dev/sdarn 3600144f0e18d8e0000000000d01d280  
/dev/sdaro 3600144f019554e0000000000d04d112  
/dev/sdarp 3600144f0ed35ce0000000000d07d111  
/dev/sdarq 3600144f019554e0000000000d04d126  
/dev/sdarr 3600144f0ed35ce0000000000d07d125  
/dev/sdars 3600144f019554e0000000000d04d140  
/dev/sdart 3600144f0ed35ce0000000000d07d139  
/dev/sdaru 3600144f019554e0000000000d04d154  
/dev/sdarv 3600144f0ed35ce0000000000d07d153  
/dev/sdarw 3600144f019554e0000000000d04d168  
/dev/sdarx 3600144f0ed35ce0000000000d07d167  
/dev/sdary 3600144f019554e0000000000d04d182  
/dev/sdarz 3600144f0ed35ce0000000000d07d181  
/dev/sdas 3600144f0054c0f0000000000a28d236  
/dev/sdasa 3600144f019554e0000000000d04d196  
/dev/sdasb 3600144f0ed35ce0000000000d07d195  
/dev/sdasc 3600144f019554e0000000000d04d210  
/dev/sdasd 3600144f0ed35ce0000000000d07d209  
/dev/sdase 3600144f019554e0000000000d04d224  
/dev/sdasf 3600144f0ed35ce0000000000d07d223  
/dev/sdasg 3600144f019554e0000000000d04d238  
/dev/sdash 3600144f0ed35ce0000000000d07d237  
/dev/sdasi 3600144f019554e0000000000d04d252  
/dev/sdasj 3600144f0ed35ce0000000000d07d251  
/dev/sdasl 3600144f019554e0000000000d04d266  
/dev/sdasl 3600144f0ed35ce0000000000d07d265  
/dev/sdasm 3600144f019554e0000000000d04d280  
/dev/sdasn 3600144f0ed35ce0000000000d07d279  
/dev/sdaso 3600144f08129ce0000000000d09d112  
/dev/sdasp 3600144f078f08e0000000000d08d111  
/dev/sdasq 3600144f08129ce0000000000d09d126  
/dev/sdasr 3600144f078f08e0000000000d08d125  
/dev/sdass 3600144f08129ce0000000000d09d140  
/dev/sdast 3600144f08129ce0000000000d09d154  
/dev/sdasu 3600144f078f08e0000000000d08d139  
/dev/sdasv 3600144f08129ce0000000000d09d168  
/dev/sdasw 3600144f078f08e0000000000d08d153  
/dev/sdasx 3600144f08129ce0000000000d09d182  
/dev/sdasz 3600144f078f08e0000000000d08d167  
/dev/sdat 3600144f0054c0f0000000000a28d237  
/dev/sdata 3600144f078f08e0000000000d08d181  
/dev/sdatb 3600144f08129ce0000000000d09d210  
/dev/sdatc 3600144f078f08e0000000000d08d195  
/dev/sdatd 3600144f08129ce0000000000d09d224  
/dev/sdate 3600144f078f08e0000000000d08d209  
/dev/sdatf 3600144f08129ce0000000000d09d238  
/dev/sdatg 3600144f078f08e0000000000d08d223  
/dev/sdath 3600144f078f08e0000000000d08d237  
/dev/sdati 3600144f08129ce0000000000d09d252  
/dev/sdatj 3600144f078f08e0000000000d08d251  
/dev/sdatk 3600144f08129ce0000000000d09d266

/dev/sdatl 3600144f078f08e0000000000d08d265  
/dev/sdatm 3600144f08129ce0000000000d09d280  
/dev/sdatn 3600144f078f08e0000000000d08d279  
/dev/sdato 3600144f059274e0000000000d10d111  
/dev/sdatp 3600144f07d498e0000000000d47d112  
/dev/sdatq 3600144f059274e0000000000d10d125  
/dev/sdatr 3600144f07d498e0000000000d47d126  
/dev/sdats 3600144f059274e0000000000d10d139  
/dev/sdatt 3600144f07d498e0000000000d47d140  
/dev/sdatu 3600144f059274e0000000000d10d153  
/dev/sdatv 3600144f07d498e0000000000d47d154  
/dev/sdatw 3600144f059274e0000000000d10d167  
/dev/sdatx 3600144f07d498e0000000000d47d168  
/dev/sdaty 3600144f059274e0000000000d10d181  
/dev/sdatz 3600144f07d498e0000000000d47d182  
/dev/sdau 3600144f0054c0f0000000000a28d238  
/dev/sdaua 3600144f059274e0000000000d10d195  
/dev/sdaub 3600144f07d498e0000000000d47d196  
/dev/sdauc 3600144f059274e0000000000d10d209  
/dev/sdaud 3600144f07d498e0000000000d47d210  
/dev/sdaue 3600144f059274e0000000000d10d223  
/dev/sdauf 3600144f07d498e0000000000d47d224  
/dev/sdaug 3600144f059274e0000000000d10d237  
/dev/sdauh 3600144f07d498e0000000000d47d238  
/dev/sdauh 3600144f059274e0000000000d10d251  
/dev/sdauj 3600144f07d498e0000000000d47d252  
/dev/sdauk 3600144f059274e0000000000d10d265  
/dev/sdaul 3600144f07d498e0000000000d47d266  
/dev/sdaum 3600144f059274e0000000000d10d279  
/dev/sdaun 3600144f07d498e0000000000d47d280  
/dev/sdaou 3600144f0c4e70e0000000000d05d112  
/dev/sdaup 3600144f08129ce0000000000d09d111  
/dev/sdauq 3600144f0c4e70e0000000000d05d126  
/dev/sdaur 3600144f08129ce0000000000d09d125  
/dev/sdaus 3600144f0c4e70e0000000000d05d140  
/dev/sdaut 3600144f08129ce0000000000d09d139  
/dev/sdauu 3600144f0c4e70e0000000000d05d154  
/dev/sdauv 3600144f08129ce0000000000d09d153  
/dev/sdauw 3600144f0c4e70e0000000000d05d168  
/dev/sdaux 3600144f08129ce0000000000d09d167  
/dev/sdauy 3600144f0c4e70e0000000000d05d182  
/dev/sdauz 3600144f08129ce0000000000d09d181  
/dev/sdava 3600144f0054c0f0000000000a28d239  
/dev/sdava 3600144f08129ce0000000000d09d195  
/dev/sdavb 3600144f0c4e70e0000000000d05d196  
/dev/sdavc 3600144f08129ce0000000000d09d209  
/dev/sdavd 3600144f0c4e70e0000000000d05d210  
/dev/sdave 3600144f08129ce0000000000d09d223  
/dev/sdavf 3600144f08129ce0000000000d09d237  
/dev/sdavg 3600144f0c4e70e0000000000d05d224  
/dev/sdavh 3600144f08129ce0000000000d09d251  
/dev/sdavi 3600144f0c4e70e0000000000d05d238  
/dev/sdavj 3600144f08129ce0000000000d09d265  
/dev/sdavk 3600144f08129ce0000000000d09d279  
/dev/sdavl 3600144f0c4e70e0000000000d05d252  
/dev/sdavm 3600144f0c4e70e0000000000d05d266  
/dev/sdavn 3600144f0c4e70e0000000000d05d280  
/dev/sdavo 3600144f09c8d8e0000000000d02d112

/dev/sdavp 3600144f09c8d8e0000000000d02d126  
/dev/sdavaq 3600144f09c8d8e0000000000d02d140  
/dev/sdavr 3600144f09c8d8e0000000000d02d154  
/dev/sdavs 3600144f09c8d8e0000000000d02d168  
/dev/sdavr 3600144f09c8d8e0000000000d02d182  
/dev/sdavu 3600144f09c8d8e0000000000d02d196  
/dev/sdavu 3600144f09c8d8e0000000000d02d210  
/dev/sdavr 3600144f09c8d8e0000000000d02d224  
/dev/sdavax 3600144f09c8d8e0000000000d02d238  
/dev/sdavy 3600144f09c8d8e0000000000d02d252  
/dev/sdavy 3600144f09c8d8e0000000000d02d266  
/dev/sdaw 3600144f0054c0f0000000000a28d240  
/dev/sdawa 3600144f09c8d8e0000000000d02d280  
/dev/sdawa 3600144f078f08e0000000000d08d112  
/dev/sdawc 3600144f078f08e0000000000d08d126  
/dev/sdawd 3600144f078f08e0000000000d08d140  
/dev/sdawe 3600144f078f08e0000000000d08d154  
/dev/sdawf 3600144f078f08e0000000000d08d168  
/dev/sdawg 3600144f078f08e0000000000d08d182  
/dev/sdawn 3600144f078f08e0000000000d08d196  
/dev/sdawn 3600144f078f08e0000000000d08d210  
/dev/sdawn 3600144f078f08e0000000000d08d224  
/dev/sdawj 3600144f078f08e0000000000d08d238  
/dev/sdawk 3600144f078f08e0000000000d08d252  
/dev/sdawm 3600144f078f08e0000000000d08d266  
/dev/sdawn 3600144f078f08e0000000000d08d280  
/dev/sdawn 3600144f059274e0000000000d10d112  
/dev/sdawn 3600144f059274e0000000000d10d126  
/dev/sdawn 3600144f059274e0000000000d10d140  
/dev/sdawn 3600144f059274e0000000000d10d154  
/dev/sdawn 3600144f059274e0000000000d10d168  
/dev/sdawn 3600144f059274e0000000000d10d182  
/dev/sdawn 3600144f059274e0000000000d10d196  
/dev/sdawn 3600144f059274e0000000000d10d210  
/dev/sdawn 3600144f059274e0000000000d10d224  
/dev/sdawn 3600144f059274e0000000000d10d238  
/dev/sdawn 3600144f059274e0000000000d10d252  
/dev/sdawn 3600144f059274e0000000000d10d266  
/dev/sdax 3600144f0054c0f0000000000a28d241  
/dev/sdaxa 3600144f059274e0000000000d10d280  
/dev/sdaxb 3600144f08452ce0000000000d06d112  
/dev/sdaxe 3600144f08452ce0000000000d06d126  
/dev/sdaxd 3600144f08452ce0000000000d06d140  
/dev/sdaxe 3600144f08452ce0000000000d06d154  
/dev/sdaxf 3600144f08452ce0000000000d06d168  
/dev/sdaxg 3600144f08452ce0000000000d06d182  
/dev/sdaxh 3600144f08452ce0000000000d06d196  
/dev/sdaxi 3600144f08452ce0000000000d06d210  
/dev/sdaxj 3600144f08452ce0000000000d06d224  
/dev/sdaxk 3600144f08452ce0000000000d06d238  
/dev/sdaxl 3600144f08452ce0000000000d06d252  
/dev/sdaxm 3600144f08452ce0000000000d06d266  
/dev/sdaxn 3600144f08452ce0000000000d06d280  
/dev/sdaxo 3600144f09c8d8e0000000000d02d102  
/dev/sdaxp 3600144f09c8d8e0000000000d02d116  
/dev/sdaxq 3600144f09c8d8e0000000000d02d130  
/dev/sdaxr 3600144f09c8d8e0000000000d02d144  
/dev/sdaxs 3600144f09c8d8e0000000000d02d158

/dev/sdaxt 3600144f09c8d8e0000000000d02d172  
/dev/sdaxu 3600144f09c8d8e0000000000d02d186  
/dev/sdaxv 3600144f09c8d8e0000000000d02d200  
/dev/sdaxw 3600144f09c8d8e0000000000d02d214  
/dev/sdaxx 3600144f09c8d8e0000000000d02d228  
/dev/sdaxy 3600144f09c8d8e0000000000d02d242  
/dev/sdaxz 3600144f09c8d8e0000000000d02d256  
/dev/sday 3600144f0054c0f0000000000a28d242  
/dev/sdaya 3600144f09c8d8e0000000000d02d270  
/dev/sdayb 3600144f09c8d8e0000000000d02d284  
/dev/sdayc 3600144f0e18d8e0000000000d01d102  
/dev/sdayd 3600144f0e18d8e0000000000d01d116  
/dev/sdaye 3600144f0e18d8e0000000000d01d130  
/dev/sdayf 3600144f0e18d8e0000000000d01d144  
/dev/sdayg 3600144f0e18d8e0000000000d01d158  
/dev/sdayh 3600144f0e18d8e0000000000d01d172  
/dev/sdayi 3600144f0e18d8e0000000000d01d186  
/dev/sdayj 3600144f0e18d8e0000000000d01d200  
/dev/sdayk 3600144f0e18d8e0000000000d01d214  
/dev/sdayl 3600144f0e18d8e0000000000d01d228  
/dev/sdaym 3600144f0e18d8e0000000000d01d242  
/dev/sdayn 3600144f0e18d8e0000000000d01d256  
/dev/sdayo 3600144f0e18d8e0000000000d01d270  
/dev/sdayp 3600144f0e18d8e0000000000d01d284  
/dev/sdayq 3600144f07d498e000000000d47d102  
/dev/sdayr 3600144f07d498e000000000d47d116  
/dev/sdays 3600144f07d498e000000000d47d130  
/dev/sdayt 3600144f07d498e000000000d47d144  
/dev/sdayu 3600144f07d498e000000000d47d158  
/dev/sdayv 3600144f07d498e000000000d47d172  
/dev/sdayw 3600144f07d498e000000000d47d186  
/dev/sdayx 3600144f07d498e000000000d47d200  
/dev/sdayy 3600144f07d498e000000000d47d214  
/dev/sdayz 3600144f07d498e000000000d47d228  
/dev/sdaz 3600144f0054c0f0000000000a28d243  
/dev/sdaza 3600144f07d498e000000000d47d242  
/dev/sdazb 3600144f07d498e000000000d47d256  
/dev/sdazc 3600144f07d498e000000000d47d270  
/dev/sdazd 3600144f07d498e000000000d47d284  
/dev/sdaze 3600144f0c4e70e000000000d05d102  
/dev/sdazf 3600144f0c4e70e000000000d05d116  
/dev/sdazg 3600144f0c4e70e000000000d05d130  
/dev/sdazh 3600144f0c4e70e000000000d05d144  
/dev/sdazi 3600144f0c4e70e000000000d05d158  
/dev/sdazj 3600144f0c4e70e000000000d05d172  
/dev/sdazk 3600144f0c4e70e000000000d05d186  
/dev/sdazl 3600144f0c4e70e000000000d05d200  
/dev/sdazm 3600144f0c4e70e000000000d05d214  
/dev/sdazn 3600144f0c4e70e000000000d05d228  
/dev/sdazo 3600144f0c4e70e000000000d05d242  
/dev/sdazp 3600144f0e18d8e000000000d01d106  
/dev/sdazq 3600144f0c4e70e000000000d05d256  
/dev/sdazr 3600144f0e18d8e000000000d01d120  
/dev/sdazs 3600144f0c4e70e000000000d05d270  
/dev/sdazt 3600144f0c4e70e000000000d05d284  
/dev/sdazu 3600144f0e18d8e000000000d01d134  
/dev/sdazv 3600144f0e18d8e000000000d01d148  
/dev/sdazw 3600144f0e18d8e000000000d01d162

/dev/sdazx 3600144f0e18d8e000000000d01d176  
/dev/sdazy 3600144f0e18d8e000000000d01d190  
/dev/sdazz 3600144f0e18d8e000000000d01d204  
/dev/sdb 3600605b001629a901385850e511b3336  
/dev/sdb1 3600605b001629a901385850e511b3336  
/dev/sdba 3600144f0054c0f000000000a28d244  
/dev/sdbaa 3600144f019554e000000000d04d102  
/dev/sdbab 3600144f0e18d8e000000000d01d218  
/dev/sdbac 3600144f0e18d8e000000000d01d232  
/dev/sdbad 3600144f019554e000000000d04d116  
/dev/sdbae 3600144f0e18d8e000000000d01d246  
/dev/sdbaf 3600144f019554e000000000d04d130  
/dev/sdbag 3600144f0e18d8e000000000d01d260  
/dev/sdbah 3600144f019554e000000000d04d144  
/dev/sdbai 3600144f0e18d8e000000000d01d274  
/dev/sdbaj 3600144f0e18d8e000000000d01d288  
/dev/sdbak 3600144f019554e000000000d04d158  
/dev/sdbal 3600144f019554e000000000d04d172  
/dev/sdbam 3600144f019554e000000000d04d186  
/dev/sdban 3600144f09c8d8e000000000d02d106  
/dev/sdbao 3600144f019554e000000000d04d200  
/dev/sdbap 3600144f09c8d8e000000000d02d120  
/dev/sdbaq 3600144f019554e000000000d04d214  
/dev/sdbar 3600144f09c8d8e000000000d02d134  
/dev/sdbas 3600144f019554e000000000d04d228  
/dev/sdbat 3600144f09c8d8e000000000d02d148  
/dev/sdbau 3600144f019554e000000000d04d242  
/dev/sdbav 3600144f09c8d8e000000000d02d162  
/dev/sdbaw 3600144f019554e000000000d04d256  
/dev/sdbax 3600144f09c8d8e000000000d02d176  
/dev/sdbay 3600144f019554e000000000d04d270  
/dev/sdbaz 3600144f09c8d8e000000000d02d190  
/dev/sdbb 3600144f0054c0f000000000a28d245  
/dev/sdbba 3600144f019554e000000000d04d284  
/dev/sdbbb 3600144f09c8d8e000000000d02d204  
/dev/sdbbc 3600144f09c8d8e000000000d02d218  
/dev/sdbbd 3600144f09c8d8e000000000d02d232  
/dev/sdbbe 3600144f0ed35ce00000000d07d102  
/dev/sdbbf 3600144f09c8d8e000000000d02d246  
/dev/sdbbg 3600144f09c8d8e000000000d02d260  
/dev/sdbbh 3600144f0ed35ce000000000d07d116  
/dev/sdbbi 3600144f09c8d8e000000000d02d274  
/dev/sdbbj 3600144f0ed35ce000000000d07d130  
/dev/sdbbk 3600144f09c8d8e000000000d02d288  
/dev/sdbbl 3600144f0ed35ce000000000d07d144  
/dev/sdbbm 3600144f0ed35ce000000000d07d158  
/dev/sdbbn 3600144f0ed35ce000000000d07d172  
/dev/sdbbo 3600144f07d498e000000000d47d106  
/dev/sdbbp 3600144f0ed35ce000000000d07d186  
/dev/sdbbq 3600144f07d498e000000000d47d120  
/dev/sdbbr 3600144f0ed35ce000000000d07d200  
/dev/sdbbs 3600144f07d498e000000000d47d134  
/dev/sdbbt 3600144f0ed35ce000000000d07d214  
/dev/sdbbu 3600144f07d498e000000000d47d148  
/dev/sdbbv 3600144f0ed35ce000000000d07d228  
/dev/sdbbw 3600144f07d498e000000000d47d162  
/dev/sdbbx 3600144f0ed35ce000000000d07d242  
/dev/sdbby 3600144f07d498e000000000d47d176

/dev/sdbbz 3600144f0ed35ce000000000d07d256  
/dev/sdbc 3600144f0054c0f000000000a28d246  
/dev/sdbca 3600144f07d498e000000000d47d190  
/dev/sdbc 3600144f0ed35ce000000000d07d270  
/dev/sdbcc 3600144f07d498e000000000d47d204  
/dev/sdbcd 3600144f0ed35ce000000000d07d284  
/dev/sdbce 3600144f07d498e000000000d47d218  
/dev/sdbcf 3600144f07d498e000000000d47d232  
/dev/sdbcg 3600144f07d498e000000000d47d246  
/dev/sdbch 3600144f078f08e000000000d08d102  
/dev/sdbci 3600144f07d498e000000000d47d260  
/dev/sdbcj 3600144f078f08e000000000d08d116  
/dev/sdbck 3600144f07d498e000000000d47d274  
/dev/sdbcl 3600144f078f08e000000000d08d130  
/dev/sdbcm 3600144f07d498e000000000d47d288  
/dev/sdbcn 3600144f078f08e000000000d08d144  
/dev/sdbco 3600144f078f08e000000000d08d158  
/dev/sdbcp 3600144f078f08e000000000d08d172  
/dev/sdbcq 3600144f0c4e70e000000000d05d106  
/dev/sdbcr 3600144f078f08e000000000d08d186  
/dev/sdbcs 3600144f078f08e000000000d08d200  
/dev/sdbct 3600144f0c4e70e000000000d05d120  
/dev/sdbcu 3600144f078f08e000000000d08d214  
/dev/sdbcv 3600144f0c4e70e000000000d05d134  
/dev/sdbcw 3600144f078f08e000000000d08d228  
/dev/sdbcx 3600144f0c4e70e000000000d05d148  
/dev/sdbcy 3600144f078f08e000000000d08d242  
/dev/sdbcz 3600144f0c4e70e000000000d05d162  
/dev/sdbd 3600144f0054c0f000000000a28d247  
/dev/sbdba 3600144f078f08e000000000d08d256  
/dev/sbdbb 3600144f0c4e70e000000000d05d176  
/dev/sdbdc 3600144f078f08e000000000d08d270  
/dev/sdbdd 3600144f0c4e70e000000000d05d190  
/dev/sdbde 3600144f078f08e000000000d08d284  
/dev/sdbdf 3600144f0c4e70e000000000d05d204  
/dev/sdbdg 3600144f059274e000000000d10d102  
/dev/sdbdh 3600144f0c4e70e000000000d05d218  
/dev/sdbdi 3600144f0c4e70e000000000d05d232  
/dev/sdbdj 3600144f059274e000000000d10d116  
/dev/sdbdk 3600144f0c4e70e000000000d05d246  
/dev/sdbdl 3600144f059274e000000000d10d130  
/dev/sdbdm 3600144f0c4e70e000000000d05d260  
/dev/sdbdn 3600144f059274e000000000d10d144  
/dev/sdbdo 3600144f0c4e70e000000000d05d274  
/dev/sdbdp 3600144f059274e000000000d10d158  
/dev/sdbdq 3600144f0c4e70e000000000d05d288  
/dev/sbdbr 3600144f059274e000000000d10d172  
/dev/sbdb 3600144f059274e000000000d10d186  
/dev/sdbdt 3600144f019554e000000000d04d106  
/dev/sbdbu 3600144f059274e000000000d10d200  
/dev/sdbdv 3600144f019554e000000000d04d120  
/dev/sdbdw 3600144f059274e000000000d10d214  
/dev/sdbdx 3600144f019554e000000000d04d134  
/dev/sbdby 3600144f059274e000000000d10d228  
/dev/sbdbz 3600144f019554e000000000d04d148  
/dev/sdbe 3600144f0054c0f000000000a28d248  
/dev/sdbea 3600144f059274e000000000d10d242  
/dev/sdbeb 3600144f019554e000000000d04d162



/dev/sdbec 3600144f059274e0000000000d10d256  
/dev/sdbed 3600144f019554e0000000000d04d176  
/dev/sdbee 3600144f059274e0000000000d10d270  
/dev/sdbef 3600144f019554e0000000000d04d190  
/dev/sdbeg 3600144f059274e0000000000d10d284  
/dev/sdbeh 3600144f019554e0000000000d04d204  
/dev/sdbei 3600144f019554e0000000000d04d218  
/dev/sdbej 3600144f019554e0000000000d04d232  
/dev/sdbek 3600144f019554e0000000000d04d246  
/dev/sdbel 3600144f08129ce0000000000d09d102  
/dev/sdbem 3600144f019554e0000000000d04d260  
/dev/sdben 3600144f08129ce0000000000d09d116  
/dev/sdbeo 3600144f019554e0000000000d04d274  
/dev/sdbep 3600144f08129ce0000000000d09d130  
/dev/sdbeq 3600144f019554e0000000000d04d288  
/dev/sdber 3600144f08129ce0000000000d09d144  
/dev/sdbes 3600144f0ed35ce0000000000d07d106  
/dev/sdbet 3600144f08129ce0000000000d09d158  
/dev/sdbeu 3600144f0ed35ce0000000000d07d120  
/dev/sdbev 3600144f08129ce0000000000d09d172  
/dev/sdbew 3600144f0ed35ce0000000000d07d134  
/dev/sdbex 3600144f08129ce0000000000d09d186  
/dev/sdbey 3600144f0ed35ce0000000000d07d148  
/dev/sdbez 3600144f08129ce0000000000d09d200  
/dev/sdbf 3600144f0054c0f0000000000a28d249  
/dev/sdbfa 3600144f0ed35ce0000000000d07d162  
/dev/sdbfb 3600144f08129ce0000000000d09d214  
/dev/sdbfc 3600144f0ed35ce0000000000d07d176  
/dev/sdbfd 3600144f08129ce0000000000d09d228  
/dev/sdbfe 3600144f0ed35ce0000000000d07d190  
/dev/sdbff 3600144f08129ce0000000000d09d242  
/dev/sdbfg 3600144f0ed35ce0000000000d07d204  
/dev/sdbfh 3600144f08129ce0000000000d09d256  
/dev/sdbfi 3600144f0ed35ce0000000000d07d218  
/dev/sdbfj 3600144f08129ce0000000000d09d270  
/dev/sdbfk 3600144f0ed35ce0000000000d07d232  
/dev/sdbfl 3600144f08129ce0000000000d09d284  
/dev/sdbfm 3600144f0ed35ce0000000000d07d246  
/dev/sdbfn 3600144f0ed35ce0000000000d07d260  
/dev/sdbfo 3600144f0ed35ce0000000000d07d274  
/dev/sdbfp 3600144f08452ce0000000000d06d102  
/dev/sdbfq 3600144f0ed35ce0000000000d07d288  
/dev/sdbfr 3600144f08452ce0000000000d06d116  
/dev/sdbfs 3600144f08452ce0000000000d06d130  
/dev/sdbft 3600144f078f08e0000000000d08d106  
/dev/sdbfu 3600144f08452ce0000000000d06d144  
/dev/sdbfv 3600144f078f08e0000000000d08d120  
/dev/sdbfw 3600144f08452ce0000000000d06d158  
/dev/sdbfx 3600144f078f08e0000000000d08d134  
/dev/sdbfy 3600144f08452ce0000000000d06d172  
/dev/sdbfz 3600144f078f08e0000000000d08d148  
/dev/sdbg 3600144f0054c0f0000000000a28d250  
/dev/sdbga 3600144f08452ce0000000000d06d186  
/dev/sdbgb 3600144f078f08e0000000000d08d162  
/dev/sdbgc 3600144f08452ce0000000000d06d200  
/dev/sdbgd 3600144f078f08e0000000000d08d176  
/dev/sdbge 3600144f08452ce0000000000d06d214  
/dev/sdbgf 3600144f078f08e0000000000d08d190

/dev/sdbgg 3600144f08452ce0000000000d06d228  
/dev/sdbgh 3600144f019554e0000000000d08d204  
/dev/sdbgi 3600144f08452ce0000000000d06d242  
/dev/sdbgj 3600144f078f08e0000000000d08d218  
/dev/sdbgk 3600144f08452ce0000000000d06d256  
/dev/sdbgl 3600144f078f08e0000000000d08d232  
/dev/sdbgm 3600144f08452ce0000000000d06d270  
/dev/sdbgn 3600144f078f08e0000000000d08d246  
/dev/sdbgo 3600144f08452ce0000000000d06d284  
/dev/sdbgp 3600144f078f08e0000000000d08d260  
/dev/sdbgq 3600144f078f08e0000000000d08d274  
/dev/sdbgr 3600144f078f08e0000000000d08d288  
/dev/sdbgs 3600144f059274e0000000000d10d106  
/dev/sdbgt 3600144f059274e0000000000d10d120  
/dev/sdbgu 3600144f059274e0000000000d10d134  
/dev/sdbgv 3600144f059274e0000000000d10d148  
/dev/sdbgw 3600144f059274e0000000000d10d162  
/dev/sdbgx 3600144f059274e0000000000d10d176  
/dev/sdbgy 3600144f059274e0000000000d10d190  
/dev/sdbgz 3600144f059274e0000000000d10d204  
/dev/sdbh 3600144f0054c0f0000000000a28d251  
/dev/sdbha 3600144f059274e0000000000d10d218  
/dev/sdbhb 3600144f059274e0000000000d10d232  
/dev/sdbhc 3600144f059274e0000000000d10d246  
/dev/sdbhd 3600144f059274e0000000000d10d260  
/dev/sdbhe 3600144f059274e0000000000d10d274  
/dev/sdbhf 3600144f059274e0000000000d10d288  
/dev/sdbhg 3600144f08129ce0000000000d09d106  
/dev/sdbhh 3600144f08129ce0000000000d09d120  
/dev/sdbhi 3600144f08129ce0000000000d09d134  
/dev/sdbhj 3600144f08129ce0000000000d09d148  
/dev/sdbhk 3600144f08129ce0000000000d09d162  
/dev/sdbhl 3600144f08129ce0000000000d09d176  
/dev/sdbhm 3600144f08129ce0000000000d09d190  
/dev/sdbhn 3600144f08129ce0000000000d09d204  
/dev/sdbho 3600144f08129ce0000000000d09d218  
/dev/sdbhp 3600144f08129ce0000000000d09d232  
/dev/sdbhq 3600144f08129ce0000000000d09d246  
/dev/sdbhr 3600144f08129ce0000000000d09d260  
/dev/sdbhs 3600144f08129ce0000000000d09d274  
/dev/sdbht 3600144f08129ce0000000000d09d288  
/dev/sdbhu 3600144f08452ce0000000000d06d106  
/dev/sdbhv 3600144f08452ce0000000000d06d120  
/dev/sdbhw 3600144f08452ce0000000000d06d134  
/dev/sdbhx 3600144f078f08e0000000000d06d148  
/dev/sdbhy 3600144f08452ce0000000000d06d162  
/dev/sdbhz 3600144f08452ce0000000000d06d176  
/dev/sdbi 3600144f0054c0f0000000000a28d252  
/dev/sdbia 3600144f08452ce0000000000d06d190  
/dev/sdbib 3600144f08452ce0000000000d06d204  
/dev/sdbic 3600144f08452ce0000000000d06d218  
/dev/sdbid 3600144f08452ce0000000000d06d232  
/dev/sdbie 3600144f08452ce0000000000d06d246  
/dev/sdbif 3600144f08452ce0000000000d06d260  
/dev/sdbig 3600144f08452ce0000000000d06d274  
/dev/sdbih 3600144f08452ce0000000000d06d288  
/dev/sdbii 3600144f0e18d8e0000000000d01d109  
/dev/sdbij 3600144f0e18d8e0000000000d01d123

/dev/sdbik 3600144f0e18d8e0000000000d01d137  
/dev/sdbil 3600144f0e18d8e0000000000d01d151  
/dev/sdbim 3600144f0e18d8e0000000000d01d165  
/dev/sdbin 3600144f0e18d8e0000000000d01d179  
/dev/sdbio 3600144f0e18d8e0000000000d01d193  
/dev/sdbip 3600144f0e18d8e0000000000d01d207  
/dev/sdbiq 3600144f0e18d8e0000000000d01d221  
/dev/sdbir 3600144f0e18d8e0000000000d01d235  
/dev/sdbis 3600144f0e18d8e0000000000d01d249  
/dev/sdbit 3600144f0e18d8e0000000000d01d263  
/dev/sdbiu 3600144f0e18d8e0000000000d01d277  
/dev/sdbiv 3600144f078f08e0000000000d08d109  
/dev/sdbiw 3600144f078f08e0000000000d08d123  
/dev/sdbix 3600144f078f08e0000000000d08d137  
/dev/sdbiy 3600144f09c8d8e0000000000d08d103  
/dev/sdbiz 3600144f078f08e0000000000d08d151  
/dev/sdbj 3600144f0054c0f0000000000a28d253  
/dev/sdbja 3600144f078f08e0000000000d08d165  
/dev/sdbjb 3600144f09c8d8e0000000000d08d117  
/dev/sdbjc 3600144f078f08e0000000000d08d179  
/dev/sdbjd 3600144f09c8d8e0000000000d02d131  
/dev/sdbje 3600144f078f08e0000000000d08d193  
/dev/sdbjf 3600144f09c8d8e0000000000d02d145  
/dev/sdbjg 3600144f078f08e0000000000d08d207  
/dev/sdbjh 3600144f09c8d8e0000000000d02d159  
/dev/sdbji 3600144f078f08e0000000000d08d221  
/dev/sdbjj 3600144f09c8d8e0000000000d02d173  
/dev/sdbjk 3600144f078f08e0000000000d08d235  
/dev/sdbjl 3600144f09c8d8e0000000000d02d187  
/dev/sdbjm 3600144f078f08e0000000000d08d249  
/dev/sdbjn 3600144f09c8d8e0000000000d02d201  
/dev/sdbjo 3600144f078f08e0000000000d08d263  
/dev/sdbjp 3600144f078f08e0000000000d08d277  
/dev/sdbjq 3600144f09c8d8e0000000000d02d215  
/dev/sdbjr 3600144f09c8d8e0000000000d02d229  
/dev/sdbjs 3600144f09c8d8e0000000000d02d243  
/dev/sdbjt 3600144f019554e0000000000d04d109  
/dev/sdbju 3600144f09c8d8e0000000000d02d257  
/dev/sdbjv 3600144f09c8d8e0000000000d02d271  
/dev/sdbjw 3600144f019554e0000000000d04d123  
/dev/sdbjx 3600144f09c8d8e0000000000d02d285  
/dev/sdbjy 3600144f019554e0000000000d04d137  
/dev/sdbjz 3600144f019554e0000000000d04d151  
/dev/sdbk 3600144f0054c0f0000000000a28d254  
/dev/sdbka 3600144f019554e0000000000d04d165  
/dev/sdbkb 3600144f019554e0000000000d04d179  
/dev/sdbkc 3600144f019554e0000000000d04d103  
/dev/sdbkd 3600144f019554e0000000000d04d193  
/dev/sdbke 3600144f019554e0000000000d04d117  
/dev/sdbkf 3600144f019554e0000000000d04d207  
/dev/sdbkg 3600144f019554e0000000000d04d131  
/dev/sdbkh 3600144f019554e0000000000d04d221  
/dev/sdbki 3600144f019554e0000000000d04d145  
/dev/sdbkj 3600144f019554e0000000000d04d235  
/dev/sdbkk 3600144f019554e0000000000d04d159  
/dev/sdbkl 3600144f019554e0000000000d04d249  
/dev/sdbkm 3600144f019554e0000000000d04d173  
/dev/sdbkn 3600144f019554e0000000000d04d263

/dev/sdbko 3600144f019554e0000000000d04d187  
/dev/sdbkp 3600144f019554e0000000000d04d277  
/dev/sdbkq 3600144f019554e0000000000d04d201  
/dev/sdbkr 3600144f019554e0000000000d04d215  
/dev/sdbks 3600144f09c8d8e0000000000d02d109  
/dev/sdbkt 3600144f019554e0000000000d04d229  
/dev/sdbku 3600144f019554e0000000000d04d243  
/dev/sdbkv 3600144f09c8d8e0000000000d02d123  
/dev/sdbkw 3600144f019554e0000000000d04d257  
/dev/sdbkx 3600144f019554e0000000000d04d271  
/dev/sdbky 3600144f09c8d8e0000000000d02d137  
/dev/sdbkz 3600144f019554e0000000000d04d285  
/dev/sdbll 3600144f0054c0f0000000000a28d255  
/dev/sdbla 3600144f09c8d8e0000000000d02d151  
/dev/sdblb 3600144f09c8d8e0000000000d02d165  
/dev/sdblc 3600144f09c8d8e0000000000d02d179  
/dev/sdbld 3600144f0c4e70e0000000000d05d103  
/dev/sdble 3600144f09c8d8e0000000000d02d193  
/dev/sdblf 3600144f09c8d8e0000000000d02d207  
/dev/sdblg 3600144f0c4e70e0000000000d05d117  
/dev/sdblh 3600144f09c8d8e0000000000d02d221  
/dev/sdbli 3600144f0c4e70e0000000000d05d131  
/dev/sdblj 3600144f09c8d8e0000000000d02d235  
/dev/sdblk 3600144f0c4e70e0000000000d05d145  
/dev/sdbll 3600144f09c8d8e0000000000d02d249  
/dev/sdblm 3600144f0c4e70e0000000000d05d159  
/dev/sdbln 3600144f09c8d8e0000000000d02d263  
/dev/sdblo 3600144f0c4e70e0000000000d05d173  
/dev/sdblp 3600144f09c8d8e0000000000d02d277  
/dev/sdblq 3600144f0c4e70e0000000000d05d187  
/dev/sdblr 3600144f0c4e70e0000000000d05d201  
/dev/sdbls 3600144f0c4e70e0000000000d05d215  
/dev/sdblt 3600144f0c4e70e0000000000d05d229  
/dev/sdblu 3600144f0c4e70e0000000000d05d243  
/dev/sdblv 3600144f0c4e70e0000000000d05d257  
/dev/sdblw 3600144f0c4e70e0000000000d05d271  
/dev/sdblx 3600144f07d498e0000000000d47d109  
/dev/sdbly 3600144f0c4e70e0000000000d05d285  
/dev/sdblz 3600144f07d498e0000000000d47d123  
/dev/sdbm 3600144f0054c0f0000000000a28d256  
/dev/sdbma 3600144f07d498e0000000000d47d137  
/dev/sdbmb 3600144f07d498e0000000000d47d103  
/dev/sdbmc 3600144f07d498e0000000000d47d151  
/dev/sdbmd 3600144f07d498e0000000000d47d117  
/dev/sdbme 3600144f07d498e0000000000d47d165  
/dev/sdbmf 3600144f07d498e0000000000d47d131  
/dev/sdbmg 3600144f07d498e0000000000d47d179  
/dev/sdbmh 3600144f07d498e0000000000d47d145  
/dev/sdbmi 3600144f07d498e0000000000d47d193  
/dev/sdbmj 3600144f07d498e0000000000d47d159  
/dev/sdbmk 3600144f07d498e0000000000d47d207  
/dev/sdbml 3600144f07d498e0000000000d47d173  
/dev/sdbmm 3600144f07d498e0000000000d47d221  
/dev/sdbmn 3600144f07d498e0000000000d47d187  
/dev/sdbmo 3600144f07d498e0000000000d47d235  
/dev/sdbmp 3600144f07d498e0000000000d47d201  
/dev/sdbmq 3600144f07d498e0000000000d47d249  
/dev/sdbmr 3600144f07d498e0000000000d47d215

/dev/sdbms 3600144f07d498e0000000000d47d263  
/dev/sdbmt 3600144f07d498e0000000000d47d229  
/dev/sdbmu 3600144f07d498e0000000000d47d277  
/dev/sdbmv 3600144f07d498e0000000000d47d243  
/dev/sdbmw 3600144f07d498e0000000000d47d257  
/dev/sdbmx 3600144f0ed35ce0000000000d7d109  
/dev/sdbmy 3600144f07d498e0000000000d47d271  
/dev/sdbmz 3600144f0ed35ce0000000000d7d123  
/dev/sdbn 3600144f0054c0f0000000000a28d257  
/dev/sdbna 3600144f07d498e0000000000d47d285  
/dev/sdbnb 3600144f0ed35ce0000000000d7d137  
/dev/sdbnc 3600144f0ed35ce0000000000d7d151  
/dev/sdbnd 3600144f0ed35ce0000000000d7d103  
/dev/sdbne 3600144f0ed35ce0000000000d7d165  
/dev/sdbnf 3600144f0ed35ce0000000000d7d117  
/dev/sdbng 3600144f0ed35ce0000000000d7d179  
/dev/sdbnh 3600144f0ed35ce0000000000d7d131  
/dev/sdbni 3600144f0ed35ce0000000000d7d193  
/dev/sdbnj 3600144f0ed35ce0000000000d7d145  
/dev/sdbnk 3600144f0ed35ce0000000000d7d207  
/dev/sdbnl 3600144f0ed35ce0000000000d7d159  
/dev/sdbnm 3600144f0ed35ce0000000000d7d221  
/dev/sdbnn 3600144f0ed35ce0000000000d7d173  
/dev/sdbno 3600144f0ed35ce0000000000d7d235  
/dev/sdbnp 3600144f0ed35ce0000000000d7d187  
/dev/sdbnq 3600144f0ed35ce0000000000d7d249  
/dev/sdbnr 3600144f0ed35ce0000000000d7d201  
/dev/sdbns 3600144f0ed35ce0000000000d7d263  
/dev/sdbnt 3600144f0ed35ce0000000000d7d215  
/dev/sdbnu 3600144f0ed35ce0000000000d7d277  
/dev/sdbnv 3600144f0ed35ce0000000000d7d229  
/dev/sdbnw 3600144f0ed35ce0000000000d7d243  
/dev/sdbnx 3600144f0ed35ce0000000000d7d257  
/dev/sdbny 3600144f0ed35ce0000000000d7d271  
/dev/sdbnz 3600144f08452ce0000000000d06d109  
/dev/sdbo 3600144f0054c0f0000000000a28d258  
/dev/sdboa 3600144f0ed35ce0000000000d7d285  
/dev/sdbob 3600144f08452ce0000000000d06d123  
/dev/sdboc 3600144f08452ce0000000000d06d137  
/dev/sdbod 3600144f08452ce0000000000d06d151  
/dev/sdboe 3600144f078f08e0000000000d08d103  
/dev/sdbof 3600144f08452ce0000000000d06d165  
/dev/sdbog 3600144f078f08e0000000000d08d117  
/dev/sdboh 3600144f08452ce0000000000d08d179  
/dev/sdboi 3600144f078f08e0000000000d08d131  
/dev/sdboj 3600144f08452ce0000000000d06d193  
/dev/sdbok 3600144f078f08e0000000000d08d145  
/dev/sdbol 3600144f08452ce0000000000d06d207  
/dev/sdbom 3600144f078f08e0000000000d08d159  
/dev/sbon 3600144f08452ce0000000000d06d221  
/dev/sboo 3600144f078f08e0000000000d08d173  
/dev/sbop 3600144f08452ce0000000000d06d235  
/dev/sboq 3600144f078f08e0000000000d08d187  
/dev/sbor 3600144f08452ce0000000000d06d249  
/dev/sbos 3600144f078f08e0000000000d08d201  
/dev/sbot 3600144f08452ce0000000000d06d263  
/dev/sbou 3600144f078f08e0000000000d08d215  
/dev/sbov 3600144f08452ce0000000000d06d277

/dev/sbow 3600144f078f08e0000000000d08d229  
/dev/sbox 3600144f078f08e0000000000d08d243  
/dev/sboy 3600144f0c4e70e0000000000d05d109  
/dev/sboz 3600144f078f08e0000000000d08d257  
/dev/sbpb 3600144f0054c0f0000000000a28d259  
/dev/sbpa 3600144f078f08e0000000000d08d271  
/dev/sbpb 3600144f0c4e70e0000000000d05d123  
/dev/sbpc 3600144f078f08e0000000000d08d285  
/dev/sbpd 3600144f0c4e70e0000000000d05d137  
/dev/sbpe 3600144f0c4e70e0000000000d05d151  
/dev/sbpf 3600144f08129ce0000000000d09d103  
/dev/sbpg 3600144f0c4e70e0000000000d05d165  
/dev/sbph 3600144f0c4e70e0000000000d05d179  
/dev/sbpi 3600144f08129ce0000000000d09d117  
/dev/sbjp 3600144f0c4e70e0000000000d05d193  
/dev/sbjk 3600144f0c4e70e0000000000d05d207  
/dev/sbjl 3600144f08129ce0000000000d09d131  
/dev/sbjm 3600144f0c4e70e0000000000d05d221  
/dev/sbjn 3600144f08129ce0000000000d09d145  
/dev/sbjp 3600144f0c4e70e0000000000d05d235  
/dev/sbjp 3600144f08129ce0000000000d09d159  
/dev/sbjq 3600144f0c4e70e0000000000d05d249  
/dev/sbjr 3600144f08129ce0000000000d09d173  
/dev/sbjt 3600144f0c4e70e0000000000d05d263  
/dev/sbjt 3600144f08129ce0000000000d09d187  
/dev/sbjp 3600144f0c4e70e0000000000d05d277  
/dev/sbjv 3600144f08129ce0000000000d09d201  
/dev/sbjw 3600144f059274e0000000000d10d109  
/dev/sbjx 3600144f08129ce0000000000d09d215  
/dev/sbjy 3600144f08129ce0000000000d09d229  
/dev/sbjz 3600144f059274e0000000000d10d123  
/dev/sbdq 3600144f0054c0f0000000000a28d260  
/dev/sbdqa 3600144f08129ce0000000000d09d243  
/dev/sbdqb 3600144f059274e0000000000d10d137  
/dev/sbdqc 3600144f08129ce0000000000d09d257  
/dev/sbdqd 3600144f059274e0000000000d10d151  
/dev/sbdqe 3600144f08129ce0000000000d09d271  
/dev/sbdqf 3600144f059274e0000000000d10d165  
/dev/sbdqg 3600144f08129ce0000000000d09d285  
/dev/sbdqh 3600144f059274e0000000000d10d179  
/dev/sbdqi 3600144f059274e0000000000d10d193  
/dev/sbdqj 3600144f059274e0000000000d10d103  
/dev/sbdqk 3600144f059274e0000000000d10d207  
/dev/sbdql 3600144f059274e0000000000d10d117  
/dev/sbdqm 3600144f059274e0000000000d10d221  
/dev/sbdqn 3600144f059274e0000000000d10d131  
/dev/sbdqo 3600144f059274e0000000000d10d235  
/dev/sbdqp 3600144f059274e0000000000d10d145  
/dev/sbdqr 3600144f059274e0000000000d10d249  
/dev/sbdq 3600144f059274e0000000000d10d159  
/dev/sbdqs 3600144f059274e0000000000d10d263  
/dev/sbdqt 3600144f059274e0000000000d10d173  
/dev/sbdqu 3600144f059274e0000000000d10d277  
/dev/sbdqv 3600144f059274e0000000000d10d187  
/dev/sbdqw 3600144f08129ce0000000000d09d109  
/dev/sbdqx 3600144f059274e0000000000d10d201  
/dev/sbdqy 3600144f08129ce0000000000d09d123  
/dev/sbdqz 3600144f059274e0000000000d10d215

/dev/sdbr 3600144f0054c0f0000000000a28d261  
/dev/sdbra 3600144f08129ce0000000000d09d137  
/dev/sdbrb 3600144f059274e0000000000d10d229  
/dev/sdbrc 3600144f08129ce0000000000d09d151  
/dev/sdbrd 3600144f059274e0000000000d10d243  
/dev/sdbre 3600144f08129ce0000000000d09d165  
/dev/sdbrf 3600144f059274e0000000000d10d257  
/dev/sdbrg 3600144f08129ce0000000000d09d179  
/dev/sdbrh 3600144f059274e0000000000d10d271  
/dev/sdbri 3600144f08129ce0000000000d09d193  
/dev/sdbrij 3600144f059274e0000000000d10d285  
/dev/sdbrk 3600144f08129ce0000000000d09d207  
/dev/sdbrl 3600144f08452ce0000000000d06d103  
/dev/sdbrm 3600144f08129ce0000000000d09d221  
/dev/sdbrn 3600144f08452ce0000000000d06d117  
/dev/sdbro 3600144f08129ce0000000000d09d235  
/dev/sdbrp 3600144f08452ce0000000000d06d131  
/dev/sdbrq 3600144f08129ce0000000000d09d249  
/dev/sdbrr 3600144f08452ce0000000000d06d145  
/dev/sdbrs 3600144f08129ce0000000000d09d263  
/dev/sdbrt 3600144f08452ce0000000000d06d159  
/dev/sdbru 3600144f08129ce0000000000d09d277  
/dev/sdbrv 3600144f08452ce0000000000d06d173  
/dev/sdbrw 3600144f08452ce0000000000d06d187  
/dev/sdbrx 3600144f08452ce0000000000d06d201  
/dev/sdbry 3600144f08452ce0000000000d06d215  
/dev/sdbrz 3600144f08452ce0000000000d06d229  
/dev/sdbs 3600144f0054c0f0000000000a28d262  
/dev/sdbsa 3600144f08452ce0000000000d06d243  
/dev/sdbsb 3600144f08452ce0000000000d06d257  
/dev/sdbsc 3600144f08452ce0000000000d06d271  
/dev/sdbsd 3600144f08452ce0000000000d06d285  
/dev/sdbse 3600144f09c8d8e000000000d02d105  
/dev/sdbsf 3600144f09c8d8e000000000d02d119  
/dev/sdbsg 3600144f09c8d8e000000000d02d133  
/dev/sdbsh 3600144f09c8d8e000000000d02d147  
/dev/sdbsi 3600144f09c8d8e000000000d02d161  
/dev/sdbsj 3600144f09c8d8e000000000d02d175  
/dev/sdbsk 3600144f09c8d8e000000000d02d189  
/dev/sdbsl 3600144f09c8d8e000000000d02d203  
/dev/sdbsm 3600144f09c8d8e000000000d02d217  
/dev/sdbsn 3600144f09c8d8e000000000d02d231  
/dev/sdbso 3600144f09c8d8e000000000d02d245  
/dev/sdbsp 3600144f09c8d8e000000000d02d259  
/dev/sdbsq 3600144f09c8d8e000000000d02d273  
/dev/sbsr 3600144f09c8d8e000000000d02d287  
/dev/sbss 3600144f019554e000000000d04d105  
/dev/sdbst 3600144f019554e000000000d04d119  
/dev/sdbsu 3600144f019554e000000000d04d133  
/dev/sdbsv 3600144f019554e000000000d04d147  
/dev/sdbsw 3600144f019554e000000000d04d161  
/dev/sdbsx 3600144f019554e000000000d04d175  
/dev/sdbsy 3600144f019554e000000000d04d189  
/dev/sdbsz 3600144f019554e000000000d04d203  
/dev/sdbt 3600144f0054c0f0000000000a28d263  
/dev/sdbta 3600144f019554e000000000d04d217  
/dev/sdbtb 3600144f019554e000000000d04d231  
/dev/sdbtc 3600144f019554e000000000d04d245

/dev/sdbtd 3600144f019554e000000000d04d259  
/dev/sdbte 3600144f019554e000000000d04d273  
/dev/sdbtf 3600144f019554e000000000d04d287  
/dev/sdbtg 3600144f0c4e70e000000000d05d105  
/dev/sdbth 3600144f0c4e70e000000000d05d119  
/dev/sdbti 3600144f0c4e70e000000000d05d133  
/dev/sdbtj 3600144f0c4e70e000000000d05d147  
/dev/sdbtk 3600144f0c4e70e000000000d05d161  
/dev/sdbtl 3600144f0c4e70e000000000d05d175  
/dev/sdbtm 3600144f0c4e70e000000000d05d189  
/dev/sdbtn 3600144f0c4e70e000000000d05d203  
/dev/sdbto 3600144f0c4e70e000000000d05d217  
/dev/sdbtp 3600144f0c4e70e000000000d05d231  
/dev/sdbtq 3600144f0c4e70e000000000d05d245  
/dev/sdbtr 3600144f0c4e70e000000000d05d259  
/dev/sdbts 3600144f0c4e70e000000000d05d273  
/dev/sdbtt 3600144f0c4e70e000000000d05d287  
/dev/sdbtu 3600144f07d498e00000000d47d105  
/dev/sdbtv 3600144f07d498e00000000d47d119  
/dev/sdbtw 3600144f07d498e00000000d47d133  
/dev/sdbtx 3600144f07d498e00000000d47d147  
/dev/sdbty 3600144f07d498e00000000d47d161  
/dev/sdbtz 3600144f07d498e00000000d47d175  
/dev/sdbu 3600144f0054c0f000000000d28d264  
/dev/sdbua 3600144f07d498e00000000d47d189  
/dev/sdbub 3600144f07d498e00000000d47d203  
/dev/sdbuc 3600144f07d498e00000000d47d217  
/dev/sdbud 3600144f07d498e00000000d47d231  
/dev/sdbue 3600144f07d498e00000000d47d245  
/dev/sdbuf 3600144f07d498e00000000d47d259  
/dev/sdbug 3600144f07d498e00000000d47d273  
/dev/sbuh 3600144f07d498e00000000d47d287  
/dev/sbui 3600144f0ed35ce00000000d07d105  
/dev/sbuj 3600144f0ed35ce00000000d07d119  
/dev/sbuk 3600144f0ed35ce00000000d07d133  
/dev/sbul 3600144f0ed35ce00000000d07d147  
/dev/sbum 3600144f0ed35ce00000000d07d161  
/dev/sbun 3600144f0ed35ce00000000d07d175  
/dev/sbuo 3600144f0ed35ce00000000d07d189  
/dev/sbup 3600144f0ed35ce00000000d07d203  
/dev/sbuq 3600144f0ed35ce00000000d07d217  
/dev/sbur 3600144f0ed35ce00000000d07d231  
/dev/sbus 3600144f0ed35ce00000000d07d245  
/dev/sbut 3600144f0ed35ce00000000d07d259  
/dev/sbuu 3600144f0ed35ce00000000d07d273  
/dev/sbuv 3600144f0ed35ce00000000d07d287  
/dev/sbuw 3600144f078f08e00000000d08d105  
/dev/sbux 3600144f078f08e00000000d08d119  
/dev/sbuy 3600144f078f08e00000000d08d133  
/dev/sbuz 3600144f078f08e00000000d08d147  
/dev/sbvb 3600144f0054c0f000000000a28d265  
/dev/sbva 3600144f078f08e00000000d08d161  
/dev/sbvb 3600144f078f08e00000000d08d175  
/dev/sbvc 3600144f078f08e00000000d08d189  
/dev/sbvd 3600144f078f08e00000000d08d203  
/dev/sbve 3600144f078f08e00000000d08d217  
/dev/sbvf 3600144f078f08e00000000d08d231  
/dev/sbvg 3600144f078f08e00000000d08d245

/dev/sdbvh 3600144f078f08e000000000d08d259  
/dev/sdbvi 3600144f078f08e000000000d08d273  
/dev/sbvj 3600144f078f08e000000000d08d287  
/dev/sbvk 3600144f08129ce000000000d09d105  
/dev/sbvl 3600144f08129ce000000000d09d119  
/dev/sbvm 3600144f08129ce000000000d09d133  
/dev/sbvn 3600144f08129ce000000000d09d147  
/dev/sbvo 3600144f08129ce000000000d09d161  
/dev/sbvp 3600144f08129ce000000000d09d175  
/dev/sbvq 3600144f08129ce000000000d09d189  
/dev/sbvr 3600144f08129ce000000000d09d203  
/dev/sbvs 3600144f08129ce000000000d09d217  
/dev/sbvt 3600144f08129ce000000000d09d231  
/dev/sbvu 3600144f08129ce000000000d09d245  
/dev/sbvv 3600144f08129ce000000000d09d259  
/dev/sbvw 3600144f08129ce000000000d09d273  
/dev/sbvx 3600144f08129ce000000000d09d287  
/dev/sbvy 3600144f059274e000000000d10d105  
/dev/sbvz 3600144f059274e000000000d10d119  
/dev/sbw 3600144f0054c0f000000000a28d266  
/dev/sbwa 3600144f059274e000000000d10d133  
/dev/sbwb 3600144f059274e000000000d10d147  
/dev/sbwc 3600144f059274e000000000d10d161  
/dev/sbwd 3600144f059274e000000000d10d175  
/dev/sbwe 3600144f059274e000000000d10d189  
/dev/sbwf 3600144f059274e000000000d10d203  
/dev/sbwg 3600144f059274e000000000d10d217  
/dev/sbwh 3600144f059274e000000000d10d231  
/dev/sbwi 3600144f059274e000000000d10d245  
/dev/sbwj 3600144f059274e000000000d10d259  
/dev/sbwk 3600144f059274e000000000d10d273  
/dev/sbwl 3600144f059274e000000000d10d287  
/dev/sbwm 3600144f08452ce00000000d06d105  
/dev/sbwn 3600144f08452ce00000000d06d119  
/dev/sbwo 3600144f08452ce00000000d06d133  
/dev/sbwp 3600144f08452ce00000000d06d147  
/dev/sbwq 3600144f08452ce00000000d06d161  
/dev/sbwr 3600144f08452ce00000000d06d175  
/dev/sbws 3600144f08452ce00000000d06d189  
/dev/sbwt 3600144f08452ce00000000d06d203  
/dev/sbwu 3600144f08452ce00000000d06d217  
/dev/sbvw 3600144f08452ce00000000d06d231  
/dev/sbww 3600144f08452ce00000000d06d245  
/dev/sbwx 3600144f08452ce00000000d06d259  
/dev/sbwy 3600144f08452ce00000000d06d273  
/dev/sbwz 3600144f08452ce00000000d06d287  
/dev/sbxb 3600144f0054c0f000000000a28d267  
/dev/sbxa 3600144f0054c0f000000000a01d200  
/dev/sbxb 3600144f0054c0f000000000a01d201  
/dev/sbxc 3600144f0054c0f000000000a01d202  
/dev/sbxd 3600144f0054c0f000000000a01d203  
/dev/sbxe 3600144f0054c0f000000000a01d204  
/dev/sbxf 3600144f0054c0f000000000a01d205  
/dev/sbxg 3600144f0054c0f000000000a01d206  
/dev/sbxh 3600144f0054c0f000000000a01d207  
/dev/sbxi 3600144f0054c0f000000000a01d208  
/dev/sbxj 3600144f0054c0f000000000a01d209  
/dev/sbxk 3600144f0054c0f000000000a01d210

/dev/sdbx1 3600144f0054c0f00000000000a01d211  
/dev/sdbxm 3600144f0054c0f00000000000a01d212  
/dev/sdbxn 3600144f0054c0f00000000000a01d213  
/dev/sdbxo 3600144f0054c0f00000000000a01d214  
/dev/sdbxp 3600144f0054c0f00000000000a01d215  
/dev/sdbxq 3600144f0054c0f00000000000a01d216  
/dev/sdbxr 3600144f0054c0f00000000000a01d217  
/dev/sdbxs 3600144f0054c0f00000000000a01d218  
/dev/sdbxt 3600144f0054c0f00000000000a01d219  
/dev/sdbxu 3600144f0054c0f00000000000a01d220  
/dev/sdbxv 3600144f0054c0f00000000000a01d221  
/dev/sdbxw 3600144f0054c0f00000000000a01d222  
/dev/sdbxx 3600144f0054c0f00000000000a01d223  
/dev/sdbxy 3600144f0054c0f00000000000a01d224  
/dev/sdbxz 3600144f0054c0f00000000000a01d225  
/dev/sdby 3600144f0054c0f0000000000a28d268  
/dev/sdbya 3600144f0054c0f00000000000a01d226  
/dev/sdbyb 3600144f0054c0f00000000000a01d227  
/dev/sdbyc 3600144f0054c0f00000000000a01d228  
/dev/sdbyd 3600144f0054c0f00000000000a01d229  
/dev/sdbye 3600144f0054c0f00000000000a01d230  
/dev/sdbyf 3600144f0054c0f00000000000a01d231  
/dev/sdbyg 3600144f0054c0f00000000000a01d232  
/dev/sdbyh 3600144f0054c0f00000000000a01d233  
/dev/sdbyi 3600144f0054c0f00000000000a01d234  
/dev/sdbyj 3600144f0054c0f00000000000a01d235  
/dev/sdbyk 3600144f0054c0f00000000000a01d236  
/dev/sdbyl 3600144f0054c0f00000000000a01d237  
/dev/sdbym 3600144f0054c0f00000000000a01d238  
/dev/sdbyn 3600144f0054c0f00000000000a01d239  
/dev/sdbyo 3600144f0054c0f00000000000a01d240  
/dev/sdbyp 3600144f0054c0f00000000000a01d241  
/dev/sdbyq 3600144f0054c0f00000000000a01d242  
/dev/sdbyr 3600144f0054c0f00000000000a01d243  
/dev/sdbys 3600144f0054c0f00000000000a01d244  
/dev/sdbyt 3600144f0054c0f00000000000a01d245  
/dev/sdbyu 3600144f0054c0f00000000000a01d246  
/dev/sdbyv 3600144f0054c0f00000000000a01d247  
/dev/sdbyw 3600144f0054c0f00000000000a01d248  
/dev/sdbyx 3600144f0054c0f00000000000a01d249  
/dev/sdbyy 3600144f0054c0f00000000000a01d250  
/dev/sdbyz 3600144f0054c0f00000000000a01d251  
/dev/sdbz 3600144f0054c0f0000000000a28d269  
/dev/sdbza 3600144f0054c0f00000000000a01d252  
/dev/sdbzb 3600144f0054c0f00000000000a01d253  
/dev/sdbzc 3600144f0054c0f00000000000a01d254  
/dev/sdbzd 3600144f0054c0f00000000000a01d255  
/dev/sdbze 3600144f0054c0f00000000000a01d256  
/dev/sdbzf 3600144f0054c0f00000000000a01d257  
/dev/sdbzg 3600144f0054c0f00000000000a01d258  
/dev/sdbzh 3600144f0054c0f00000000000a01d259  
/dev/sdbzi 3600144f0054c0f00000000000a01d260  
/dev/sdbzj 3600144f0054c0f00000000000a01d261  
/dev/sdbzk 3600144f0054c0f00000000000a01d262  
/dev/sdbzl 3600144f0054c0f00000000000a01d263  
/dev/sdbzm 3600144f0054c0f00000000000a01d264  
/dev/sdbzn 3600144f0054c0f00000000000a01d265  
/dev/sdbzo 3600144f0054c0f00000000000a01d266

/dev/sdbzp 3600144f0054c0f00000000000a01d267  
/dev/sdbzq 3600144f0054c0f00000000000a01d268  
/dev/sdbzr 3600144f0054c0f00000000000a01d269  
/dev/sdbzs 3600144f0054c0f00000000000a01d270  
/dev/sdbzt 3600144f0054c0f00000000000a01d271  
/dev/sdbzu 3600144f0054c0f00000000000a01d272  
/dev/sdbzv 3600144f0054c0f00000000000a01d273  
/dev/sdbzw 3600144f0054c0f00000000000a01d274  
/dev/sdbzx 3600144f0054c0f00000000000a01d275  
/dev/sdbzy 3600144f0054c0f00000000000a01d276  
/dev/sdbzz 3600144f0054c0f00000000000a01d277  
/dev/sdc 3600605b0016298e0ff0000460454bafa  
/dev/sdca 3600144f0054c0f0000000000a28d270  
/dev/sdcaa 3600144f0054c0f0000000000a01d278  
/dev/sdcab 3600144f0054c0f0000000000a01d279  
/dev/sdcac 3600144f0a3770e0000000000a01d300  
/dev/sdcad 3600144f0a3770e0000000000a01d303  
/dev/sdcae 3600144f0dbed0d0000000000a01d700  
/dev/sdcael 3600144f0dbed0d0000000000a01d701  
/dev/sdcalf 3600144f0dbed0d0000000000a01d701  
/dev/sdcbl 3600144f0054c0f0000000000a28d271  
/dev/sdcb 3600144f0054c0f0000000000a28d271  
/dev/sdcb 3600144f0054c0f0000000000a28d272  
/dev/sdcd 3600144f0054c0f0000000000a28d273  
/dev/sdce 3600144f0054c0f0000000000a28d274  
/dev/sdcf 3600144f0054c0f0000000000a28d275  
/dev/sdceg 3600144f0054c0f0000000000a28d276  
/dev/sdch 3600144f0054c0f0000000000a28d277  
/dev/sdci 3600144f0054c0f0000000000a28d278  
/dev/sdcj 3600144f0054c0f0000000000a28d279  
/dev/sdck 3600144f0a3770e0000000000a03d300  
/dev/sdcl 3600144f0a3770e0000000000a03d303  
/dev/sdcm 3600144f019554e00000000000d04d114  
/dev/sdcn 3600144f019554e00000000000d04d128  
/dev/sdco 3600144f019554e00000000000d04d142  
/dev/sdcp 3600144f019554e00000000000d04d156  
/dev/sdcq 3600144f019554e00000000000d04d170  
/dev/sdcr 3600144f019554e00000000000d04d184  
/dev/sdcs 3600144f019554e00000000000d04d198  
/dev/sdct 3600144f019554e00000000000d04d212  
/dev/sdca 3600144f019554e00000000000d04d226  
/dev/sdca 3600144f019554e00000000000d04d240  
/dev/sdcw 3600144f019554e00000000000d04d254  
/dev/sdcx 3600144f019554e00000000000d04d268  
/dev/sdcy 3600144f019554e00000000000d04d282  
/dev/sdcz 3600144f09c8d8e0000000000d02d114  
/dev/sdd 3600605b0016298e0ff0000460454c04b  
/dev/sdd1 3600605b0016298e0ff0000460454c04b  
/dev/sdd2 3600605b0016298e0ff0000460454c04b  
/dev/sdd3 3600605b0016298e0ff0000460454c04b  
/dev/sdda 3600144f09c8d8e0000000000d02d128  
/dev/sddb 3600144f09c8d8e0000000000d02d142  
/dev/sddc 3600144f09c8d8e0000000000d02d156  
/dev/sddd 3600144f09c8d8e0000000000d02d170  
/dev/sdde 3600144f09c8d8e0000000000d02d184  
/dev/sddf 3600144f09c8d8e0000000000d02d198  
/dev/sddg 3600144f09c8d8e0000000000d02d212  
/dev/sddh 3600144f09c8d8e0000000000d02d226

/dev/sddi 3600144f09c8d8e0000000000d02d240  
/dev/sddj 3600144f09c8d8e0000000000d02d254  
/dev/sddk 3600144f09c8d8e0000000000d02d268  
/dev/sddl 3600144f09c8d8e0000000000d02d282  
/dev/sddm 3600144f0e18d8e0000000000d01d114  
/dev/sddn 3600144f0e18d8e0000000000d01d128  
/dev/sddo 3600144f0e18d8e0000000000d01d142  
/dev/sddp 3600144f0e18d8e0000000000d01d156  
/dev/sddq 3600144f0e18d8e0000000000d01d170  
/dev/sddr 3600144f0e18d8e0000000000d01d184  
/dev/sdds 3600144f0e18d8e0000000000d01d198  
/dev/sddt 3600144f0e18d8e0000000000d01d212  
/dev/sddu 3600144f0e18d8e0000000000d01d226  
/dev/sddv 3600144f0e18d8e0000000000d01d240  
/dev/sddw 3600144f0e18d8e0000000000d01d254  
/dev/sddx 3600144f0e18d8e0000000000d01d268  
/dev/sddy 3600144f0e18d8e0000000000d01d282  
/dev/sddz 3600144f08452ce0000000000d06d114  
/dev/sde 3600605b0016298e0ff0000460454c06a  
/dev/sdea 3600144f08452ce0000000000d06d128  
/dev/sdeb 3600144f08452ce0000000000d06d142  
/dev/sdec 3600144f08452ce0000000000d06d156  
/dev/sded 3600144f08452ce0000000000d06d170  
/dev/sdee 3600144f08452ce0000000000d06d184  
/dev/sdef 3600144f08452ce0000000000d06d198  
/dev/sdeg 3600144f08452ce0000000000d06d212  
/dev/sdeh 3600144f08452ce0000000000d06d226  
/dev/sdei 3600144f08452ce0000000000d06d240  
/dev/sdej 3600144f08452ce0000000000d06d254  
/dev/sdek 3600144f08452ce0000000000d06d268  
/dev/sdel 3600144f08452ce0000000000d06d282  
/dev/sdem 3600144f078f08e0000000000d08d114  
/dev/sden 3600144f078f08e0000000000d08d128  
/dev/sdeo 3600144f078f08e0000000000d08d142  
/dev/sdep 3600144f078f08e0000000000d08d156  
/dev/sdeq 3600144f078f08e0000000000d08d170  
/dev/sder 3600144f078f08e0000000000d08d184  
/dev/sdes 3600144f078f08e0000000000d08d198  
/dev/sdet 3600144f078f08e0000000000d08d212  
/dev/sdeu 3600144f078f08e0000000000d08d226  
/dev/sdev 3600144f078f08e0000000000d08d240  
/dev/sdew 3600144f078f08e0000000000d08d254  
/dev/sdex 3600144f078f08e0000000000d08d268  
/dev/sdey 3600144f078f08e0000000000d08d282  
/dev/sdez 3600144f0c4e70e0000000000d05d114  
/dev/sdf 3600605b0016298e0ff0000460454cc17  
/dev/sdfa 3600144f0c4e70e0000000000d05d128  
/dev/sdfb 3600144f0c4e70e0000000000d05d142  
/dev/sdfc 3600144f0c4e70e0000000000d05d156  
/dev/sdfd 3600144f0c4e70e0000000000d05d170  
/dev/sdfe 3600144f0c4e70e0000000000d05d184  
/dev/sdff 3600144f0c4e70e0000000000d05d198  
/dev/sdfg 3600144f0c4e70e0000000000d05d212  
/dev/sdfh 3600144f0c4e70e0000000000d05d226  
/dev/sdfi 3600144f0c4e70e0000000000d05d240  
/dev/sdfj 3600144f0c4e70e0000000000d05d254  
/dev/sdfk 3600144f0c4e70e0000000000d05d268  
/dev/sdfl 3600144f0c4e70e0000000000d05d282

```

/dev/sdfm 3600144f07d498e0000000000d47d114
/dev/sdfn 3600144f07d498e0000000000d47d128
/dev/sdfo 3600144f07d498e0000000000d47d142
/dev/sdfr 3600144f07d498e0000000000d47d156
/dev/sdfq 3600144f07d498e0000000000d47d170
/dev/sdfr 3600144f07d498e0000000000d47d184
/dev/sdfs 3600144f07d498e0000000000d47d198
/dev/sdft 3600144f07d498e0000000000d47d212
/dev/sdfu 3600144f07d498e0000000000d47d226
/dev/sdfv 3600144f07d498e0000000000d47d240
/dev/sdfw 3600144f07d498e0000000000d47d254
/dev/sdfx 3600144f07d498e0000000000d47d268
/dev/sdfy 3600144f07d498e0000000000d47d282
/dev/sdfz 3600144f0ed35ce0000000000d07d114
/dev/sdgg 36000605b0016298e0ff0000460454d193
/dev/sdgl 36000605b0016298e0ff0000460454d193
/dev/sdgm 36000605b0016298e0ff0000460454d193
/dev/sdgn 36000605b0016298e0ff0000460454d193
/dev/sdgo 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193
/dev/sdgp 36000605b0016298e0ff0000460454d193

```

```

/dev/sdhk 3600144f059274e0000000000d10d268
/dev/sdhl 3600144f059274e0000000000d10d282
/dev/sdhm 3600144f0e18d8e0000000000d01d101
/dev/sdhn 3600144f0e18d8e0000000000d01d115
/dev/sdho 3600144f0e18d8e0000000000d01d129
/dev/sdhp 3600144f0e18d8e0000000000d01d143
/dev/sdhq 3600144f0e18d8e0000000000d01d157
/dev/sdhr 3600144f0e18d8e0000000000d01d171
/dev/sdhs 3600144f0e18d8e0000000000d01d185
/dev/sdht 3600144f0e18d8e0000000000d01d199
/dev/sdhu 3600144f0e18d8e0000000000d01d213
/dev/sdhv 3600144f0e18d8e0000000000d01d227
/dev/sdhw 3600144f0e18d8e0000000000d01d241
/dev/sdhx 3600144f0e18d8e0000000000d01d255
/dev/sdhy 3600144f0e18d8e0000000000d01d269
/dev/sdhz 3600144f0e18d8e0000000000d01d283
/dev/sdi 3600144f0054c0f0000000000a28d200
/dev/sdia 3600144f09c8d8e0000000000d02d101
/dev/sdib 3600144f09c8d8e0000000000d02d115
/dev/sdic 3600144f09c8d8e0000000000d02d129
/dev/sdid 3600144f09c8d8e0000000000d02d143
/dev/sdie 3600144f09c8d8e0000000000d02d157
/dev/sdif 3600144f09c8d8e0000000000d02d171
/dev/sdig 3600144f09c8d8e0000000000d02d185
/dev/sdih 3600144f09c8d8e0000000000d02d199
/dev/sdii 3600144f09c8d8e0000000000d02d213
/dev/sdij 3600144f09c8d8e0000000000d02d227
/dev/sdik 3600144f09c8d8e0000000000d02d241
/dev/sdil 3600144f09c8d8e0000000000d02d255
/dev/sdim 3600144f09c8d8e0000000000d02d269
/dev/sdin 3600144f09c8d8e0000000000d02d283
/dev/sdio 3600144f019554e0000000000d04d101
/dev/sdip 3600144f019554e0000000000d04d115
/dev/sdiq 3600144f019554e0000000000d04d129
/dev/sdir 3600144f019554e0000000000d04d143
/dev/sdis 3600144f019554e0000000000d04d157
/dev/sdit 3600144f019554e0000000000d04d171
/dev/sdiu 3600144f019554e0000000000d04d185
/dev/sdiv 3600144f019554e0000000000d04d199
/dev/sdiw 3600144f019554e0000000000d04d213
/dev/sdix 3600144f019554e0000000000d04d227
/dev/sdiy 3600144f019554e0000000000d04d241
/dev/sdiz 3600144f019554e0000000000d04d255
/dev/sdj 3600144f0054c0f0000000000a28d201
/dev/sdja 3600144f078f08e0000000000d08d269
/dev/sdjb 3600144f019554e0000000000d04d283
/dev/sdjc 3600144f0c4e70e0000000000d05d101
/dev/sdjd 3600144f0c4e70e0000000000d05d115
/dev/sdje 3600144f0c4e70e0000000000d05d129
/dev/sdjf 3600144f0c4e70e0000000000d05d143
/dev/sdjg 3600144f0c4e70e0000000000d05d157
/dev/sdjh 3600144f0c4e70e0000000000d05d171
/dev/sdji 3600144f0c4e70e0000000000d05d185
/dev/sdjj 3600144f0c4e70e0000000000d05d199
/dev/sdjk 3600144f0c4e70e0000000000d05d213
/dev/sdjl 3600144f0c4e70e0000000000d05d227
/dev/sdjm 3600144f0c4e70e0000000000d05d241
/dev/sdjn 3600144f0c4e70e0000000000d05d255

```

```

/dev/sdjo 3600144f0c4e70e0000000000d05d269
/dev/sdjp 3600144f0c4e70e0000000000d05d283
/dev/sdjq 3600144f07d498e0000000000d47d101
/dev/sdjr 3600144f07d498e0000000000d47d115
/dev/sdjs 3600144f07d498e0000000000d47d129
/dev/sdjt 3600144f07d498e0000000000d47d143
/dev/sdju 3600144f07d498e0000000000d47d157
/dev/sdjk 3600144f07d498e0000000000d47d171
/dev/sdjk 3600144f07d498e0000000000d47d185
/dev/sdjk 3600144f07d498e0000000000d47d199
/dev/sdjk 3600144f07d498e0000000000d47d213
/dev/sdjk 3600144f07d498e0000000000d47d227
/dev/sdjk 3600144f0054c0f0000000000a28d202
/dev/sdka 3600144f07d498e0000000000d47d241
/dev/sdkb 3600144f07d498e0000000000d47d255
/dev/sdkc 3600144f07d498e0000000000d47d269
/dev/sdkd 3600144f07d498e0000000000d47d283
/dev/sdke 3600144f0ed35ce0000000000d07d101
/dev/sdkf 3600144f0ed35ce0000000000d07d115
/dev/sdkg 3600144f0ed35ce0000000000d07d129
/dev/sdkg 3600144f0ed35ce0000000000d07d143
/dev/sdkg 3600144f0ed35ce0000000000d07d157
/dev/sdkj 3600144f0ed35ce0000000000d07d171
/dev/sdkk 3600144f0ed35ce0000000000d07d185
/dev/sdkl 3600144f0ed35ce0000000000d07d199
/dev/sdkm 3600144f0ed35ce0000000000d07d213
/dev/sdkn 3600144f0ed35ce0000000000d07d227
/dev/sdko 3600144f0ed35ce0000000000d07d241
/dev/sdkp 3600144f0ed35ce0000000000d07d255
/dev/sdkq 3600144f0ed35ce0000000000d07d269
/dev/sdkr 3600144f0ed35ce0000000000d07d283
/dev/sdks 3600144f078f08e0000000000d08d101
/dev/sdkt 3600144f078f08e0000000000d08d115
/dev/sdku 3600144f078f08e0000000000d08d129
/dev/sdkv 3600144f078f08e0000000000d08d143
/dev/sdkw 3600144f078f08e0000000000d08d157
/dev/sdkx 3600144f078f08e0000000000d08d171
/dev/sdky 3600144f078f08e0000000000d08d185
/dev/sdkz 3600144f078f08e0000000000d08d199
/dev/sdl 3600144f0054c0f0000000000a28d203
/dev/sdla 3600144f078f08e0000000000d08d213
/dev/sdlb 3600144f078f08e0000000000d08d227
/dev/sdlc 3600144f078f08e0000000000d08d241
/dev/sdlc 3600144f078f08e0000000000d08d255
/dev/sdle 3600144f078f08e0000000000d08d269
/dev/sdlf 3600144f078f08e0000000000d08d283
/dev/sdlg 3600144f08129ce0000000000d09d101
/dev/sdlh 3600144f08129ce0000000000d09d115
/dev/sdli 3600144f08129ce0000000000d09d129
/dev/sdlj 3600144f08129ce0000000000d09d143
/dev/sdlk 3600144f08129ce0000000000d09d157
/dev/sdll 3600144f08129ce0000000000d09d171
/dev/sdlm 3600144f08129ce0000000000d09d185
/dev/sdln 3600144f08129ce0000000000d09d199
/dev/sdlo 3600144f08129ce0000000000d09d213
/dev/sdlp 3600144f08129ce0000000000d09d227
/dev/sdlq 3600144f08129ce0000000000d09d241
/dev/sdlr 3600144f08129ce0000000000d09d255

```

/dev/sdls 3600144f08129ce0000000000d09d269 /dev/sdnw 3600144f019554e0000000000d04d275 /dev/sdq 3600144f0054c0f0000000000a28d208  
/dev/sdlt 3600144f08129ce0000000000d09d283 /dev/sdnox 3600144f019554e0000000000d04d289 /dev/sdqa 3600144f0c4e70e0000000000d05d247  
/dev/sdlu 3600144f059274e0000000000d10d101 /dev/sdny 3600144f09c8d8e000000000d02d107 /dev/sdqb 3600144f0c4e70e0000000000d05d164  
/dev/sdlv 3600144f059274e0000000000d10d115 /dev/sdnz 3600144f09c8d8e000000000d02d121 /dev/sdqc 3600144f0c4e70e0000000000d05d261  
/dev/sdlw 3600144f059274e0000000000d10d129 /dev/sdo 3600144f0054c0f0000000000a28d206 /dev/sdqd 3600144f0c4e70e0000000000d05d178  
/dev/sdlx 3600144f059274e0000000000d10d143 /dev/sdoa 3600144f09c8d8e000000000d02d135 /dev/sdqe 3600144f0c4e70e0000000000d05d275  
/dev/sdly 3600144f059274e0000000000d10d157 /dev/sdob 3600144f09c8d8e000000000d02d149 /dev/sdqf 3600144f0c4e70e0000000000d05d192  
/dev/sdlz 3600144f059274e0000000000d10d171 /dev/sdoc 3600144f09c8d8e000000000d02d163 /dev/sdqg 3600144f0c4e70e0000000000d05d289  
/dev/sdm 3600144f0054c0f0000000000a28d204 /dev/sdod 3600144f09c8d8e000000000d02d177 /dev/sdqh 3600144f0c4e70e0000000000d05d206  
/dev/sdma 3600144f059274e0000000000d10d185 /dev/sdoe 3600144f09c8d8e000000000d02d191 /dev/sdqi 3600144f0c4e70e0000000000d05d220  
/dev/sdmb 3600144f059274e0000000000d10d199 /dev/sdof 3600144f09c8d8e000000000d02d205 /dev/sdqj 3600144f0c4e70e0000000000d05d234  
/dev/sdmc 3600144f059274e0000000000d10d213 /dev/sdog 3600144f09c8d8e000000000d02d219 /dev/sdqk 3600144f0c4e70e0000000000d05d248  
/dev/sdmd 3600144f059274e0000000000d10d227 /dev/sdoh 3600144f09c8d8e000000000d02d233 /dev/sdql 3600144f0c4e70e0000000000d05d262  
/dev/sdme 3600144f059274e0000000000d10d241 /dev/sdoi 3600144f09c8d8e000000000d02d247 /dev/sdqm 3600144f0c4e70e0000000000d05d276  
/dev/sdmf 3600144f059274e0000000000d10d255 /dev/sdoj 3600144f09c8d8e000000000d02d261 /dev/sdqn 3600144f059274e0000000000d10d107  
/dev/sdmg 3600144f059274e0000000000d10d269 /dev/sdok 3600144f09c8d8e000000000d02d275 /dev/sdqq 3600144f059274e0000000000d10d121  
/dev/sdmh 3600144f059274e0000000000d10d283 /dev/sdol 3600144f09c8d8e000000000d02d289 /dev/sdqp 3600144f019554e0000000000d04d108  
/dev/sdmi 3600144f08452ce0000000000d06d101 /dev/sdom 3600144f078f08e0000000000d08d107 /dev/sdqr 3600144f059274e0000000000d10d135  
/dev/sdmj 3600144f08452ce0000000000d06d115 /dev/sdon 3600144f078f08e0000000000d08d121 /dev/sdqs 3600144f059274e0000000000d10d149  
/dev/sdmk 3600144f08452ce0000000000d06d129 /dev/sdoo 3600144f078f08e0000000000d08d135 /dev/sdqt 3600144f019554e0000000000d04d136  
/dev/sdml 3600144f08452ce0000000000d06d143 /dev/sdop 3600144f078f08e0000000000d08d149 /dev/sdqu 3600144f059274e0000000000d10d163  
/dev/sdmm 3600144f08452ce0000000000d06d157 /dev/sdoq 3600144f078f08e0000000000d08d163 /dev/sdqv 3600144f059274e0000000000d10d177  
/dev/sdmn 3600144f08452ce0000000000d06d171 /dev/sdor 3600144f078f08e0000000000d08d177 /dev/sdqw 3600144f019554e0000000000d04d150  
/dev/sdmo 3600144f08452ce0000000000d06d185 /dev/sdos 3600144f078f08e0000000000d08d191 /dev/sdqx 3600144f059274e0000000000d10d191  
/dev/sdmp 3600144f08452ce0000000000d06d199 /dev/sdot 3600144f078f08e0000000000d08d205 /dev/sdqy 3600144f019554e0000000000d04d164  
/dev/sdmq 3600144f08452ce0000000000d06d213 /dev/sdou 3600144f078f08e0000000000d08d219 /dev/sdqz 3600144f059274e0000000000d10d205  
/dev/sdmr 3600144f08452ce0000000000d06d227 /dev/sdov 3600144f078f08e0000000000d08d233 /dev/sdr 3600144f0054c0f0000000000a28d209  
/dev/sdms 3600144f08452ce0000000000d06d241 /dev/sdow 3600144f078f08e0000000000d08d247 /dev/sdra 3600144f019554e0000000000d04d178  
/dev/sdmt 3600144f08452ce0000000000d06d255 /dev/sdox 3600144f078f08e0000000000d08d261 /dev/sdrb 3600144f059274e0000000000d10d219  
/dev/sdmu 3600144f08452ce0000000000d06d269 /dev/sdoy 3600144f078f08e0000000000d08d275 /dev/sdrc 3600144f019554e0000000000d04d192  
/dev/sdmv 3600144f08452ce0000000000d06d283 /dev/sdoz 3600144f0ed35ce0000000000d07d107 /dev/sdrd 3600144f059274e0000000000d10d233  
/dev/sdmw 3600144f0e18d8e0000000000d01d107 /dev/sdp 3600144f0054c0f0000000000a28d207 /dev/sdre 3600144f019554e0000000000d04d206  
/dev/sdmx 3600144f0e18d8e0000000000d01d121 /dev/sdpa 3600144f0ed35ce0000000000d07d121 /dev/sdrf 3600144f059274e0000000000d10d247  
/dev/sdmy 3600144f0e18d8e0000000000d01d135 /dev/sdpb 3600144f0ed35ce0000000000d07d135 /dev/sdrg 3600144f019554e0000000000d04d220  
/dev/sdmz 3600144f0e18d8e0000000000d01d149 /dev/sdpc 3600144f0ed35ce0000000000d07d149 /dev/sdrh 3600144f059274e0000000000d10d261  
/dev/sdn 3600144f0054c0f0000000000a28d205 /dev/sdpd 3600144f0ed35ce0000000000d07d163 /dev/sdri 3600144f019554e0000000000d04d234  
/dev/sdna 3600144f0e18d8e0000000000d01d163 /dev/sdpe 3600144f0ed35ce0000000000d07d177 /dev/sdrl 3600144f059274e0000000000d10d275  
/dev/sdnb 3600144f0e18d8e0000000000d01d177 /dev/sdpf 3600144f0ed35ce0000000000d07d191 /dev/sdrk 3600144f019554e0000000000d04d248  
/dev/sdnc 3600144f0e18d8e0000000000d01d191 /dev/sdpg 3600144f0ed35ce0000000000d07d205 /dev/sdrl 3600144f019554e0000000000d04d262  
/dev/sdnd 3600144f0e18d8e0000000000d01d205 /dev/sdph 3600144f0ed35ce0000000000d07d219 /dev/sdrm 3600144f07d498e0000000000d47d107  
/dev/sdne 3600144f0e18d8e0000000000d01d219 /dev/sdpl 3600144f0ed35ce0000000000d07d233 /dev/sdrn 3600144f019554e0000000000d04d276  
/dev/sdnf 3600144f0e18d8e0000000000d01d233 /dev/sdpl 3600144f0ed35ce0000000000d07d275 /dev/sdro 3600144f07d498e0000000000d47d121  
/dev/sdng 3600144f0e18d8e0000000000d01d247 /dev/sdpm 3600144f0c4e70e0000000000d05d119 /dev/sdrp 3600144f07d498e0000000000d47d135  
/dev/sdnh 3600144f0e18d8e0000000000d01d261 /dev/sdpo 3600144f0c4e70e0000000000d05d121 /dev/sdrq 3600144f07d498e0000000000d47d149  
/dev/sdni 3600144f0e18d8e0000000000d01d275 /dev/sdpp 3600144f0c4e70e0000000000d05d149 /dev/sdrr 3600144f078f08e0000000000d08d108  
/dev/sdnj 3600144f0e18d8e0000000000d01d289 /dev/sdpq 3600144f0c4e70e0000000000d05d163 /dev/sdrs 3600144f07d498e0000000000d47d163  
/dev/sdnk 3600144f019554e0000000000d04d107 /dev/sdpr 3600144f0c4e70e0000000000d05d177 /dev/sdrt 3600144f078f08e0000000000d08d122  
/dev/sdnl 3600144f019554e0000000000d04d121 /dev/sdps 3600144f0c4e70e0000000000d05d191 /dev/sdrv 3600144f078f08e0000000000d08d136  
/dev/sdnm 3600144f019554e0000000000d04d135 /dev/sdpt 3600144f0c4e70e0000000000d05d108 /dev/sdrw 3600144f07d498e0000000000d47d191  
/dev/sdnn 3600144f019554e0000000000d04d149 /dev/sdpu 3600144f0c4e70e0000000000d05d205 /dev/sdry 3600144f07d498e0000000000d47d205  
/dev/sdno 3600144f019554e0000000000d04d163 /dev/sdpv 3600144f0c4e70e0000000000d05d122 /dev/sdrz 3600144f078f08e0000000000d08d164  
/dev/sdnp 3600144f019554e0000000000d04d177 /dev/sdpw 3600144f0c4e70e0000000000d05d219 /dev/sds 3600144f0054c0f0000000000a28d210  
/dev/sdnq 3600144f019554e0000000000d04d191 /dev/sdpx 3600144f0c4e70e0000000000d05d136 /dev/sdsa 3600144f07d498e0000000000d47d219  
/dev/sdnr 3600144f019554e0000000000d04d205 /dev/sdpy 3600144f0c4e70e0000000000d05d233 /dev/sdsb 3600144f078f08e0000000000d08d178  
/dev/sdns 3600144f019554e0000000000d04d219 /dev/sdpz 3600144f0c4e70e0000000000d05d150 /dev/sdsc 3600144f07d498e0000000000d47d233  
/dev/sdnt 3600144f019554e0000000000d04d233  
/dev/sdnu 3600144f019554e0000000000d04d247  
/dev/sdnv 3600144f019554e0000000000d04d261

/dev/sdsd 3600144f078f08e0000000000d08d192  
/dev/sdse 3600144f07d498e0000000000d47d247  
/dev/sdsf 3600144f078f08e0000000000d08d206  
/dev/sdsg 3600144f07d498e0000000000d47d261  
/dev/sdsh 3600144f078f08e0000000000d08d220  
/dev/sdsi 3600144f07d498e0000000000d47d275  
/dev/sdsj 3600144f078f08e0000000000d08d234  
/dev/sdsk 3600144f07d498e0000000000d47d289  
/dev/sdsl 3600144f078f08e0000000000d08d248  
/dev/sdsm 3600144f078f08e0000000000d08d262  
/dev/sdsn 3600144f08129ce0000000000d09d107  
/dev/sdso 3600144f078f08e0000000000d08d276  
/dev/sdsp 3600144f08129ce0000000000d09d121  
/dev/sdsq 3600144f0e18d8e0000000000d01d108  
/dev/sdsr 3600144f08129ce0000000000d09d135  
/dev/sdss 3600144f0e18d8e0000000000d01d122  
/dev/sdst 3600144f08129ce0000000000d09d149  
/dev/sdsu 3600144f0e18d8e0000000000d01d136  
/dev/sdsv 3600144f08129ce0000000000d09d163  
/dev/sdsw 3600144f0e18d8e0000000000d01d150  
/dev/sdsx 3600144f08129ce0000000000d09d177  
/dev/sdsv 3600144f0e18d8e0000000000d01d164  
/dev/sdsz 3600144f0e18d8e0000000000d01d178  
/dev/sdt 3600144f0054c0f0000000000a28d211  
/dev/sdta 3600144f08129ce0000000000d09d191  
/dev/sdtb 3600144f0e18d8e0000000000d01d192  
/dev/sdte 3600144f08129ce0000000000d09d205  
/dev/sdtd 3600144f0e18d8e0000000000d01d206  
/dev/sdte 3600144f08129ce0000000000d09d219  
/dev/sdtf 3600144f0e18d8e0000000000d01d220  
/dev/sdtg 3600144f08129ce0000000000d09d233  
/dev/sdth 3600144f0e18d8e0000000000d01d234  
/dev/sdti 3600144f08129ce0000000000d09d247  
/dev/sdtj 3600144f0e18d8e0000000000d01d248  
/dev/sdtk 3600144f08129ce0000000000d09d261  
/dev/sdtl 3600144f0e18d8e0000000000d01d262  
/dev/sdtm 3600144f08129ce0000000000d09d275  
/dev/sdtn 3600144f0e18d8e0000000000d01d276  
/dev/sdto 3600144f08452ce0000000000d06d107  
/dev/sdtp 3600144f08129ce0000000000d09d108  
/dev/sdtq 3600144f08452ce0000000000d06d121  
/dev/sdtr 3600144f08129ce0000000000d09d122  
/dev/sdts 3600144f08129ce0000000000d09d136  
/dev/sdtt 3600144f08452ce0000000000d06d135  
/dev/sdtu 3600144f08129ce0000000000d09d150  
/dev/sdtt 3600144f08452ce0000000000d06d149  
/dev/sdtw 3600144f08129ce0000000000d09d164  
/dev/sdtx 3600144f08452ce0000000000d06d163  
/dev/sdty 3600144f08129ce0000000000d09d178  
/dev/sdtz 3600144f08452ce0000000000d06d177  
/dev/sdu 3600144f0054c0f0000000000a28d212  
/dev/sdua 3600144f08129ce0000000000d09d192  
/dev/sdub 3600144f08129ce0000000000d06d206  
/dev/sduc 3600144f08452ce0000000000d06d191  
/dev/sdud 3600144f08129ce0000000000d09d220  
/dev/sdue 3600144f08452ce0000000000d06d205  
/dev/sduf 3600144f08129ce0000000000d09d234  
/dev/sdug 3600144f08452ce0000000000d06d219

/dev/sduh 3600144f08129ce0000000000d09d248  
/dev/sdii 3600144f08452ce0000000000d06d233  
/dev/sduj 3600144f08129ce0000000000d09d262  
/dev/sduk 3600144f08452ce0000000000d06d247  
/dev/sdul 3600144f08129ce0000000000d09d276  
/dev/sdum 3600144f08452ce0000000000d06d261  
/dev/sdun 3600144f07d498e0000000000d47d108  
/dev/sduo 3600144f08452ce0000000000d06d275  
/dev/sdup 3600144f08452ce0000000000d06d289  
/dev/sduq 3600144f07d498e0000000000d47d122  
/dev/sdur 3600144f07d498e0000000000d47d136  
/dev/sdus 3600144f07d498e0000000000d47d150  
/dev/sdut 3600144f07d498e0000000000d47d164  
/dev/sduu 3600144f07d498e0000000000d47d178  
/dev/sduv 3600144f07d498e0000000000d47d192  
/dev/sduw 3600144f07d498e0000000000d47d206  
/dev/sdud 3600144f07d498e0000000000d47d220  
/dev/sduy 3600144f07d498e0000000000d47d234  
/dev/sduz 3600144f07d498e0000000000d47d248  
/dev/sdv 3600144f0054c0f0000000000a28d213  
/dev/sdva 3600144f07d498e0000000000d47d262  
/dev/sdvb 3600144f07d498e0000000000d47d276  
/dev/sdvc 3600144f09c8d8e0000000000d02d108  
/dev/sdvd 3600144f09c8d8e0000000000d02d122  
/dev/sdve 3600144f09c8d8e0000000000d02d136  
/dev/sdvv 3600144f09c8d8e0000000000d02d150  
/dev/sdvg 3600144f09c8d8e0000000000d02d164  
/dev/sdvh 3600144f09c8d8e0000000000d02d178  
/dev/sdvi 3600144f09c8d8e0000000000d02d192  
/dev/sdvj 3600144f09c8d8e0000000000d02d206  
/dev/sdvk 3600144f09c8d8e0000000000d02d220  
/dev/sdvl 3600144f09c8d8e0000000000d02d234  
/dev/sdvm 3600144f09c8d8e0000000000d02d248  
/dev/sdvn 3600144f09c8d8e0000000000d02d262  
/dev/sdvo 3600144f09c8d8e0000000000d02d276  
/dev/sdvp 3600144f059274e0000000000d10d108  
/dev/sdvq 3600144f059274e0000000000d10d122  
/dev/sdvr 3600144f059274e0000000000d10d136  
/dev/sdvs 3600144f059274e0000000000d10d150  
/dev/sdvt 3600144f059274e0000000000d10d164  
/dev/sdvv 3600144f059274e0000000000d10d178  
/dev/sdvw 3600144f059274e0000000000d10d192  
/dev/sdvw 3600144f059274e0000000000d10d206  
/dev/sdvv 3600144f059274e0000000000d10d220  
/dev/sdvy 3600144f059274e0000000000d10d234  
/dev/sdvz 3600144f059274e0000000000d10d248  
/dev/sdw 3600144f0054c0f0000000000a28d214  
/dev/sdwa 3600144f059274e0000000000d10d262  
/dev/sdwb 3600144f059274e0000000000d10d276  
/dev/sdwc 3600144f08452ce0000000000d06d108  
/dev/sdwd 3600144f08452ce0000000000d06d122  
/dev/sdwe 3600144f08452ce0000000000d06d136  
/dev/sdww 3600144f08452ce0000000000d06d150  
/dev/sdwx 3600144f08452ce0000000000d06d164  
/dev/sdwh 3600144f08452ce0000000000d06d178  
/dev/sdwi 3600144f08452ce0000000000d06d192  
/dev/sdwj 3600144f08452ce0000000000d06d206  
/dev/sdww 3600144f08452ce0000000000d06d220

/dev/sdwl 3600144f08452ce0000000000d06d234  
/dev/sdwm 3600144f08452ce0000000000d06d248  
/dev/sdwn 3600144f08452ce0000000000d06d262  
/dev/sdwo 3600144f08452ce0000000000d06d276  
/dev/sdwp 3600144f0ed35ce0000000000d07d108  
/dev/sdwq 3600144f0ed35ce0000000000d07d122  
/dev/sdwr 3600144f0ed35ce0000000000d07d136  
/dev/sdws 3600144f0ed35ce0000000000d07d150  
/dev/sdwt 3600144f0ed35ce0000000000d07d164  
/dev/sdww 3600144f0ed35ce0000000000d07d178  
/dev/sdwx 3600144f0ed35ce0000000000d07d192  
/dev/sdww 3600144f0ed35ce0000000000d07d206  
/dev/sdwx 3600144f0ed35ce0000000000d07d220  
/dev/sdwy 3600144f0ed35ce0000000000d07d234  
/dev/sdwz 3600144f0ed35ce0000000000d07d248  
/dev/sdx 3600144f0054c0f0000000000a28d215  
/dev/sdxa 3600144f0ed35ce0000000000d07d262  
/dev/sdxb 3600144f0ed35ce0000000000d07d276  
/dev/sdxc 3600144f0e18d8e0000000000d01d103  
/dev/sdxd 3600144f0e18d8e0000000000d01d117  
/dev/sdxe 3600144f0e18d8e0000000000d01d131  
/dev/sdxf 3600144f0e18d8e0000000000d01d145  
/dev/sdxg 3600144f0e18d8e0000000000d01d159  
/dev/sdxh 3600144f0e18d8e0000000000d01d173  
/dev/sdxi 3600144f0e18d8e0000000000d01d187  
/dev/sdxj 3600144f0e18d8e0000000000d01d201  
/dev/sdxk 3600144f0e18d8e0000000000d01d215  
/dev/sdxl 3600144f0e18d8e0000000000d01d229  
/dev/sdxm 3600144f0e18d8e0000000000d01d243  
/dev/sdxn 3600144f0e18d8e0000000000d01d257  
/dev/sdxo 3600144f0e18d8e0000000000d01d271  
/dev/sdxx 3600144f0e18d8e0000000000d01d285  
/dev/sdxx 3600144f0e18d8e0000000000d01d113  
/dev/sdxx 3600144f0e18d8e0000000000d01d127  
/dev/sdxx 3600144f0e18d8e0000000000d01d141  
/dev/sdxt 3600144f0e18d8e0000000000d01d155  
/dev/sdxu 3600144f0e18d8e0000000000d01d169  
/dev/sdxv 3600144f0e18d8e0000000000d01d183  
/dev/sdxw 3600144f0e18d8e0000000000d01d197  
/dev/sdxx 3600144f0e18d8e0000000000d01d211  
/dev/sdxy 3600144f0e18d8e0000000000d01d225  
/dev/sdxz 3600144f0e18d8e0000000000d01d239  
/dev/sdy 3600144f0054c0f0000000000a28d216  
/dev/sdya 3600144f0e18d8e0000000000d01d253  
/dev/sdyb 3600144f0e18d8e0000000000d01d267  
/dev/sdyc 3600144f0e18d8e0000000000d01d281  
/dev/sdyd 3600144f09c8d8e0000000000d02d113  
/dev/sdye 3600144f09c8d8e0000000000d02d127  
/dev/sdyf 3600144f09c8d8e0000000000d02d141  
/dev/sdyg 3600144f09c8d8e0000000000d02d155  
/dev/sdyh 3600144f019554e0000000000d04d104  
/dev/sdyi 3600144f09c8d8e0000000000d02d169  
/dev/sdyj 3600144f019554e0000000000d04d118  
/dev/sdyk 3600144f09c8d8e0000000000d02d183  
/dev/sdyl 3600144f019554e0000000000d04d132  
/dev/sdym 3600144f09c8d8e0000000000d02d197  
/dev/sdyn 3600144f019554e0000000000d04d146  
/dev/sdyo 3600144f09c8d8e0000000000d02d211

```
/dev/sdyp 3600144f019554e0000000000d04d160 d01d105",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_40" BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d125",NAME=
/dev/sdyq 3600144f09c8d8e0000000000d02d225 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d106",NAME=
"tpcc-disk-
/dev/sdyr 3600144f019554e0000000000d04d174 "tpcc-disk-
d01d125",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_240"
/dev/sdys 3600144f09c8d8e0000000000d02d239 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d126",NAME=
/dev/sdyt 3600144f019554e0000000000d04d188 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d107",NAME=
"tpcc-disk-
/dev/sdyu 3600144f09c8d8e0000000000d02d253 "tpcc-disk-
d01d126",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_250"
/dev/sdyv 3600144f019554e0000000000d04d202 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d127",NAME=
/dev/sdyw 3600144f09c8d8e0000000000d02d267 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d108",NAME=
"tpcc-disk-
/dev/sdyx 3600144f019554e0000000000d04d216 "tpcc-disk-
d01d127",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_260"
/dev/sdyz 3600144f09c8d8e0000000000d02d281 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d128",NAME=
/dev/sdyz 3600144f019554e0000000000d04d230 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d109",NAME=
"tpcc-disk-
/dev/sdz 3600144f0054c0f0000000000a28d217 "tpcc-disk-
d01d128",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_270"
/dev/sdza 3600144f019554e0000000000d04d244 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d129",NAME=
/dev/sdzb 3600144f019554e0000000000d04d258 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d110",NAME=
"tpcc-disk-
/dev/sdzc 3600144f019554e0000000000d04d113 "tpcc-disk-
d01d129",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_280"
/dev/sdzd 3600144f019554e0000000000d04d272 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d130",NAME=
/dev/sdze 3600144f019554e0000000000d04d127 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d111",NAME=
"tpcc-disk-
/dev/sdzf 3600144f019554e0000000000d04d286 "tpcc-disk-
d01d130",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_290"
/dev/sdzg 3600144f019554e0000000000d04d141 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d131",NAME=
/dev/sdzh 3600144f0e18d8e0000000000d01d104 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d112",NAME=
"tpcc-disk-
/dev/sdzi 3600144f019554e0000000000d04d155 "tpcc-disk-
d01d131",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_300"
/dev/sdzj 3600144f0e18d8e0000000000d01d118 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d132",NAME=
/dev/sdzk 3600144f019554e0000000000d04d169 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d113",NAME=
"tpcc-disk-
/dev/sd zl 3600144f0e18d8e0000000000d01d132 "tpcc-disk-
d01d132",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_310"
/dev/sdzm 3600144f019554e0000000000d04d183 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d133",NAME=
/dev/sdzn 3600144f0e18d8e0000000000d01d146 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d114",NAME=
"tpcc-disk-
/dev/sdzo 3600144f019554e0000000000d04d197 "tpcc-disk-
d01d133",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_320"
/dev/sd zp 3600144f0e18d8e0000000000d01d160 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d134",NAME=
/dev/sd zq 3600144f019554e0000000000d04d211 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d115",NAME=
"tpcc-disk-
/dev/sd zr 3600144f0e18d8e0000000000d01d174 "tpcc-disk-
d01d134",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_330"
/dev/sd zs 3600144f019554e0000000000d04d225 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d135",NAME=
/dev/sdzt 3600144f0e18d8e0000000000d01d188 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d116",NAME=
"tpcc-disk-
/dev/sd zu 3600144f0e18d8e0000000000d01d202 "tpcc-disk-
d01d135",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_340"
/dev/sd zv 3600144f0e18d8e0000000000d01d216 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d136",NAME=
/dev/sd zw 3600144f019554e0000000000d04d239 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d117",NAME=
"tpcc-disk-
/dev/sd zx 3600144f0e18d8e0000000000d01d230 "tpcc-disk-
d01d136",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_350"
/dev/sd zy 3600144f019554e0000000000d04d253 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d137",NAME=
/dev/sd zz 3600144f0e18d8e0000000000d01d244 BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d118",NAME=
"tpcc-disk-
d01d137",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_360"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d138",NAME=
"tpcc-disk-
d01d138",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_370"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d139",NAME=
"tpcc-disk-
d01d139",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_380"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d140",NAME=
"tpcc-disk-
d01d140",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_390"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d141",NAME=
"tpcc-disk-
d01d141",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_400"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d142",NAME=
"tpcc-disk-
d01d142",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_410"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d143",NAME=
"tpcc-disk-
d01d143",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_420"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d144",NAME=
```

**30-tpcc-disks.rules**

```
KERNEL=="sd*1",BUS=="scsi",PROGRAM="/sbin/scsi_id -g -d
$Stempnode"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d101",NAME=
"tpcc-disk-
d01d101",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_0"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d102",NAME=
"tpcc-disk-
d01d102",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_10"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d103",NAME=
"tpcc-disk-
d01d103",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_20"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d104",NAME=
"tpcc-disk-
d01d104",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_30"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d105",NAME=
"tpcc-disk-
d01d105",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_40"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d106",NAME=
"tpcc-disk-
d01d106",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_50"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d107",NAME=
"tpcc-disk-
d01d107",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_60"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d108",NAME=
"tpcc-disk-
d01d108",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_70"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d109",NAME=
"tpcc-disk-
d01d109",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_80"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d110",NAME=
"tpcc-disk-
d01d110",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_90"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d111",NAME=
"tpcc-disk-
d01d111",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_100"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d112",NAME=
"tpcc-disk-
d01d112",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_110"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d113",NAME=
"tpcc-disk-
d01d113",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_120"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d114",NAME=
"tpcc-disk-
d01d114",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_130"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d115",NAME=
"tpcc-disk-
d01d115",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_140"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d116",NAME=
"tpcc-disk-
d01d116",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_150"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d117",NAME=
"tpcc-disk-
d01d117",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_160"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d118",NAME=
"tpcc-disk-
d01d118",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_170"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d119",NAME=
"tpcc-disk-
d01d119",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_180"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d120",NAME=
"tpcc-disk-
d01d120",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_190"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d121",NAME=
"tpcc-disk-
d01d121",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_200"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d122",NAME=
"tpcc-disk-
d01d122",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_210"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d123",NAME=
"tpcc-disk-
d01d123",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_220"
BUS=="scsi",RESULT=="3600144f0e18d8e0000000000d01d124",NAME=
"tpcc-disk-
d01d124",GROUP:="dba",OWNER:="oracle",SYMLINK+="cust_0_230"
```















































































```

"tpcc-disk-          d47d274",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_652"
d47d255",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_462"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d275",NAME=
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d256",NAME="tpcc-disk-          balance_eth_irqs.sh
"tpcc-disk-          d47d275",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_662"
d47d256",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_472"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d276",NAME=#!/bin/ksh
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d257",NAME="tpcc-disk-
"tpcc-disk-          d47d276",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_672"   set -A CPULIST 0 10 40 50 60 70 80 90 100 110 120 130 140 150
d47d257",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_482"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d277",NAME=
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d258",NAME="tpcc-disk-          ETHIRQ='cd /proc/irq; ls */eth0-TxRx* | grep TxRx | awk 'BEGIN {FS="/"}
"tpcc-disk-          d47d277",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_682"   {print $1}'
d47d258",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_492"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d278",NAME=
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d259",NAME="tpcc-disk-          id=0
"tpcc-disk-          d47d278",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_692"   for irq in $ETHIRQ; do
d47d259",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_502"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d279",NAME=   cpu=${CPULIST[$id]}
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d260",NAME="tpcc-disk-          echo "echo $cpu > /proc/irq/$irq/smp_affinity_list"
"tpcc-disk-          d47d279",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_702"   echo $cpu > /proc/irq/$irq/smp_affinity_list
d47d260",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_512"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d280",NAME=   ((id=id+1))
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d261",NAME="tpcc-disk-          if [ $id -gt 13 ]; then
"tpcc-disk-          d47d280",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_712"   id=0
d47d261",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_522"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d281",NAME=   fi
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d262",NAME="tpcc-disk-          done
"tpcc-disk-          d47d281",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_722"
d47d262",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_532"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d282",NAME=
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d263",NAME="tpcc-disk-          balance_qla_irqs.sh
"tpcc-disk-          d47d282",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_732"
d47d263",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_542"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d283",NAME=#!/bin/bash
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d264",NAME="tpcc-disk-          service irqbalance stop
"tpcc-disk-          d47d283",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_742"
d47d264",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_552"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d284",NAME=last_node=-1
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d265",NAME="tpcc-disk-          declare -i count=0
"tpcc-disk-          d47d284",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_752"   declare -i cpu cpu1 cpu2 cpu3 cpu4
d47d265",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_562"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d285",NAME=
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d266",NAME="tpcc-disk-          for dir in /sys/bus/pci/drivers/qla2xxx/0000* ; do
"tpcc-disk-          d47d285",GROUP:="dba",OWNER:="oracle",SYMLINK+="misc_0_2"   node=`cat $dir/numa_node`
d47d266",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_572"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d286",NAME=   irqs=`cat $dir/msi_irqs`
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d267",NAME="tpcc-disk-          if [ "`echo $irqs | wc -w`" != "2" ]; then
"tpcc-disk-          d47d286",GROUP:="dba",OWNER:="oracle",SYMLINK+="misc_0_12"   echo >&2 "script expects 2 interrupts per device"
d47d267",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_582"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d287",NAME=   exit 1
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d268",NAME="tpcc-disk-          fi
"tpcc-disk-          d47d287",GROUP:="dba",OWNER:="oracle",SYMLINK+="misc_0_22"   first_cpu=`sed 's/.*!/' < $dir/local_cpulist`
d47d268",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_592"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d288",NAME=
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d269",NAME="tpcc-disk-          echo $node $irqs $first_cpu $dir
"tpcc-disk-          d47d288",GROUP:="dba",OWNER:="oracle",SYMLINK+="misc_0_32"   done | sort | while read node irq1 irq2 cpu1 dir ; do
d47d269",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_602"   BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d289",NAME=   cpu2=$cpu1+10
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d270",NAME="tpcc-disk-          cpu3=$cpu1+80
"tpcc-disk-          d47d289",GROUP:="dba",OWNER:="oracle",SYMLINK+="control_002"   cpu4=$cpu1+90
d47d270",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_612"   BUS=="scsi",RESULT=="360080e50002402e60000058f4f39237d",NAME=   if [ "$node" != "$last_node" ]; then
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d271",NAME="tpcc-disk-          count=1
"tpcc-disk-          2540_10",GROUP:="dba",OWNER:="oracle",SYMLINK+="log2540_10"   cpu=$cpu1
d47d271",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_622"   BUS=="scsi",RESULT=="360080e50002401a00000045d4f3923e9",NAME=   else
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d272",NAME="tpcc-disk-          count=$count+1
"tpcc-disk-          2540_11",GROUP:="dba",OWNER:="oracle",SYMLINK+="log2540_11"   case $count in
d47d272",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_632"   BUS=="scsi",RESULT=="360080e50001c39160000039e4f392421",NAME=   2)          cpu=$cpu2;;
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d273",NAME="tpcc-disk-          2540_20",GROUP:="dba",OWNER:="oracle",SYMLINK+="log2540_20"   3)          cpu=$cpu3;;
"tpcc-disk-          d47d273",GROUP:="dba",OWNER:="oracle",SYMLINK+="stok_0_642"   BUS=="scsi",RESULT=="360080e5000233a4c0000039d4f39246f",NAME=   4)          cpu=$cpu4;;
BUS=="scsi",RESULT=="3600144f07d498e0000000000d47d274",NAME="tpcc-disk-
"tpcc-disk-          2540_21",GROUP:="dba",OWNER:="oracle",SYMLINK+="log2540_21"

```

```

*)
echo "more devices than
expected on node $node"
count=1
cpu=$cpu1;;
esac
fi
last_node=$node
echo "#$dir"
echo "echo $cpu > /proc/irq/$irq1/smp_affinity_list"
echo "echo $cpu > /proc/irq/$irq2/smp_affinity_list"
echo
echo $cpu > /proc/irq/$irq1/smp_affinity_list
echo $cpu > /proc/irq/$irq2/smp_affinity_list
done

limits.conf
# /etc/security/limits.conf
#
#Each line describes a limit for a user in the form:
#
#<domain> <type> <item> <value>
#
#Where:
#<domain> can be:
# - an user name
# - a group name, with @group syntax
# - the wildcard *, for default entry
# - the wildcard %, can be also used with %group syntax,
# for maxlogin limit
#
#<type> can have the two values:
# - "soft" for enforcing the soft limits
# - "hard" for enforcing hard limits
#
#<item> can be one of the following:
# - core - limits the core file size (KB)
# - data - max data size (KB)
# - fsize - maximum filesize (KB)
# - memlock - max locked-in-memory address space (KB)
# - nofile - max number of open files
# - rss - max resident set size (KB)
# - stack - max stack size (KB)
# - cpu - max CPU time (MIN)
# - nproc - max number of processes
# - as - address space limit (KB)
# - maxlogins - max number of logins for this user
# - maxsyslogins - max number of logins on the system
# - priority - the priority to run user process with
# - locks - max number of file locks the user can hold
# - sigpending - max number of pending signals
# - msgqueue - max memory used by POSIX message queues (bytes)
# - nice - max nice priority allowed to raise to values: [-20, 19]
# - rtprio - max realtime priority
#

#<domain> <type> <item> <value>
#
#@student soft core 0
#@faculty soft nproc 20
#@faculty hard nproc 50
#ftp hard nproc 0
#@student - maxlogins 4
oracle soft nofile 4096
oracle hard nofile 65536
oracle soft nproc 10240
oracle hard nproc 16384

# G5 TPC settings.
#oracle soft stack 10240

@dba soft memlock unlimited
@dba hard memlock unlimited
@dba soft stack unlimited
@dba hard stack unlimited

# End of file

==== On-Board Devices =====
Zoar 2x GbE.
Zoar 2x GbE.
Zoar 2x GbE.
Zoar 2x GbE.

==== Upgradeable Slots =====
ID Status Type Description
-----
0 in use PCI Express PCIE0
1 in use PCI Express PCIE1
2 in use PCI Express PCIE2
3 in use PCI Express PCIE3
4 in use PCI Express PCIE4
5 in use PCI Express PCIE5
.....
System Configuration: Oracle Corporation i86pc
Memory size: 6136 Megabytes
System Peripherals (Software Nodes):
i86pc
scsi_vhci, instance #0
pci, instance #0
pci108e,4845 (driver not attached)
pci8086,3408, instance #0
pci108e,0, instance #0
pci108e,0, instance #1
pci8086,3409, instance #1
pci108e,0, instance #2
pci108e,0, instance #3
pci8086,340a, instance #2
pci1077,171, instance #0
pci1077,171, instance #1
pci8086,340c, instance #3
pci1000,3150, instance #0
sd, instance #0
sd, instance #2
sd, instance #1
sd, instance #3
sd, instance #4

#<domain> <type> <item> <value>
#
#@student hard nproc 20
#@faculty soft nproc 20
#@faculty hard nproc 50
#ftp hard nproc 0
#@student - maxlogins 4
oracle soft nofile 4096
oracle hard nofile 65536
oracle soft nproc 10240
oracle hard nproc 16384

# G5 TPC settings.
#oracle soft stack 10240

@dba soft memlock unlimited
@dba hard memlock unlimited
@dba soft stack unlimited
@dba hard stack unlimited

# End of file

==== Processor Sockets =====
Version Location Tag
-----
Intel(R) Xeon(R) CPU E5540 @ 2.53GHz CPU 1

==== Memory Device Sockets =====
Type Status Set Device Locator Bank Locator
-----
Unknown in use 0 D2 BANK2
Unknown empty 0 D1 BANK1
Unknown empty 0 D0 BANK0
Unknown in use 0 D5 BANK5

```

sd, instance #5  
sd, instance #6  
sd, instance #7  
sd, instance #8  
sd, instance #9  
sd, instance #10  
sd, instance #11  
smp, instance #0  
ses, instance #12  
pci8086,340e, instance #4  
pci111d,806e, instance #6  
pci111d,806e, instance #7  
pci1000,30b0, instance #0  
iport, instance #13  
smp, instance #5  
disk, instance #144  
disk, instance #97  
disk, instance #121  
disk, instance #124  
disk, instance #130  
disk, instance #133  
disk, instance #138  
disk, instance #142  
disk, instance #99  
disk, instance #103  
disk, instance #29  
disk, instance #106  
disk, instance #109  
disk, instance #34  
disk, instance #114  
disk, instance #117  
disk, instance #148  
disk, instance #151  
disk, instance #156  
disk, instance #31  
enclosure, instance #4  
iport, instance #9  
smp, instance #2  
disk, instance #80  
disk, instance #37  
disk, instance #85  
disk, instance #40  
disk, instance #89  
disk, instance #45  
disk, instance #94  
disk, instance #12  
disk, instance #13  
disk, instance #47  
disk, instance #62  
disk, instance #65  
disk, instance #68  
disk, instance #72  
disk, instance #74  
disk, instance #78  
disk, instance #56  
disk, instance #14  
disk, instance #51  
disk, instance #58

enclosure, instance #1  
iport, instance #4  
pci111d,806e, instance #8  
pci1000,30b0, instance #1  
iport, instance #11  
smp, instance #3  
disk, instance #93  
disk, instance #41  
disk, instance #73  
disk, instance #76  
disk, instance #81  
disk, instance #84  
disk, instance #88  
disk, instance #91  
disk, instance #54  
disk, instance #17  
disk, instance #59  
disk, instance #64  
disk, instance #98  
disk, instance #101  
disk, instance #104  
disk, instance #52  
disk, instance #16  
disk, instance #15  
disk, instance #44  
disk, instance #49  
enclosure, instance #2  
iport, instance #12  
smp, instance #6  
disk, instance #158  
disk, instance #108  
disk, instance #161  
disk, instance #111  
disk, instance #165  
disk, instance #116  
disk, instance #167  
disk, instance #33  
disk, instance #36  
disk, instance #134  
disk, instance #136  
disk, instance #139  
disk, instance #147  
disk, instance #149  
disk, instance #152  
disk, instance #131  
disk, instance #126  
disk, instance #39  
disk, instance #122  
disk, instance #120  
enclosure, instance #5  
iport, instance #7  
pci8086,3410, instance #5  
pci111d,806e, instance #9  
pci111d,806e, instance #10  
pci1000,30b0, instance #2  
iport, instance #10  
smp, instance #4  
disk, instance #18

disk, instance #19  
disk, instance #21  
disk, instance #23  
disk, instance #25  
disk, instance #26  
disk, instance #27  
disk, instance #61  
disk, instance #32  
disk, instance #67  
disk, instance #70  
disk, instance #75  
disk, instance #77  
disk, instance #82  
disk, instance #87  
disk, instance #90  
disk, instance #96  
disk, instance #20  
disk, instance #22  
disk, instance #24  
enclosure, instance #3  
iport, instance #14  
smp, instance #7  
disk, instance #160  
disk, instance #112  
disk, instance #162  
disk, instance #115  
disk, instance #164  
disk, instance #42  
disk, instance #169  
disk, instance #46  
disk, instance #123  
disk, instance #125  
disk, instance #48  
disk, instance #118  
disk, instance #128  
disk, instance #137  
disk, instance #140  
disk, instance #143  
disk, instance #146  
disk, instance #150  
disk, instance #154  
disk, instance #157  
enclosure, instance #6  
iport, instance #6  
pci111d,806e, instance #11  
pci1000,30b0, instance #3  
iport, instance #8  
smp, instance #1  
disk, instance #100  
disk, instance #71  
disk, instance #79  
disk, instance #83  
disk, instance #86  
disk, instance #92  
disk, instance #95  
disk, instance #102  
disk, instance #105  
disk, instance #107



disk, instance #93	disk, instance #61	disk, instance #60
disk, instance #41	disk, instance #32	disk, instance #63
disk, instance #73	disk, instance #67	disk, instance #69
disk, instance #76	disk, instance #70	disk, instance #66
disk, instance #81	disk, instance #75	enclosure, instance #0
disk, instance #84	disk, instance #77	iport, instance #15
disk, instance #88	disk, instance #82	smp, instance #8
disk, instance #91	disk, instance #87	disk, instance #166
disk, instance #54	disk, instance #90	disk, instance #110
disk, instance #17	disk, instance #96	disk, instance #168
disk, instance #59	disk, instance #20	disk, instance #113
disk, instance #64	disk, instance #22	disk, instance #170
disk, instance #98	disk, instance #24	disk, instance #38
disk, instance #101	enclosure, instance #3	disk, instance #171
disk, instance #104	iport, instance #14	disk, instance #43
disk, instance #52	smp, instance #7	disk, instance #119
disk, instance #16	disk, instance #160	disk, instance #141
disk, instance #15	disk, instance #112	disk, instance #145
disk, instance #44	disk, instance #162	disk, instance #153
disk, instance #49	disk, instance #115	disk, instance #155
enclosure, instance #2	disk, instance #164	disk, instance #159
iport, instance #12	disk, instance #42	disk, instance #163
smp, instance #6	disk, instance #169	disk, instance #127
disk, instance #158	disk, instance #46	disk, instance #50
disk, instance #108	disk, instance #123	disk, instance #129
disk, instance #161	disk, instance #125	disk, instance #132
disk, instance #111	disk, instance #48	disk, instance #135
disk, instance #165	disk, instance #118	enclosure, instance #7
disk, instance #116	disk, instance #128	iport, instance #5
disk, instance #167	disk, instance #137	pci8086,342d (driver not attached)
disk, instance #33	disk, instance #140	pci8086,342e (driver not attached)
disk, instance #36	disk, instance #143	pci8086,3422 (driver not attached)
disk, instance #134	disk, instance #146	pci8086,3423, instance #0
disk, instance #136	disk, instance #150	pci8086,3438 (driver not attached)
disk, instance #139	disk, instance #154	pci108e,4845 (driver not attached)
disk, instance #147	disk, instance #157	pci108e,4845 (driver not attached)
disk, instance #149	enclosure, instance #6	pci108e,4845 (driver not attached)
disk, instance #152	iport, instance #6	pci108e,4845 (driver not attached)
disk, instance #131	pci111d,806e, instance #11	pci108e,4845 (driver not attached)
disk, instance #126	pci1000,30b0, instance #3	pci108e,4845 (driver not attached)
disk, instance #39	iport, instance #8	pci108e,4845 (driver not attached)
disk, instance #122	smp, instance #1	pci108e,4845 (driver not attached)
disk, instance #120	disk, instance #100	pci108e,4845 (driver not attached)
enclosure, instance #5	disk, instance #71	pci108e,4845 (driver not attached)
iport, instance #7	disk, instance #79	pci108e,4845 (driver not attached)
pci8086,3410, instance #5	disk, instance #83	pci108e,4845 (driver not attached)
pci111d,806e, instance #9	disk, instance #86	pci108e,4845 (driver not attached)
pci111d,806e, instance #10	disk, instance #92	pci108e,4845 (driver not attached)
pci1000,30b0, instance #2	disk, instance #95	pci108e,4845 (driver not attached)
iport, instance #10	disk, instance #102	pci108e,4845 (driver not attached)
smp, instance #4	disk, instance #105	pci8086,244e, instance #0
disk, instance #18	disk, instance #107	display, instance #0
disk, instance #19	disk, instance #35	isa, instance #0
disk, instance #21	disk, instance #53	motherboard (driver not attached)
disk, instance #23	disk, instance #55	asy, instance #0
disk, instance #25	disk, instance #57	motherboard (driver not attached)
disk, instance #26	disk, instance #28	pit_beep, instance #0
disk, instance #27	disk, instance #30	pci108e,4845, instance #0











```

/pci@0,0/pci8086,3410@9/pci111d,806e@0/pci111d,806e@4/pci1000, 2 backup wu 0 - 23433 22.88GB *
30b0@0/iport@f/disk@w5080020000558240,0 (23434/0/0) 47992832
171. c21t5080020000558252d0 <ATA-MARVELLS88SA02-D129 cyl 3 unassigned wu 0
23434 alt 2 hd 16 sec 128> (0/0/0) 0
/pci@0,0/pci8086,3410@9/pci111d,806e@0/pci111d,806e@4/pci1000, 4 unassigned wm 50 - 5886 5.70GB * root device and root filesystem configuration:
30b0@0/iport@f/disk@w5080020000558252,0 (5837/0/0) 11954176 *
Specify disk (enter its number): 5 unassigned wm 5887 - 11723 5.70GB * The following may be used to override the defaults provided by
the boot program:
format for sample FMod (5837/0/0) 11954176 *
12. c14t508002000055734Ad0 <ATA-MARVELLS88SA02-D129 cyl (5837/0/0) 11954176 * rootfs: Set the filesystem type of the root.
23434 alt 2 hd 16 sec 128> 8 boot wu 0 - 0 1.00MB * rootdev: Set the root device. This should be a fully
the expanded physical pathname. The default is
30b0@0/iport@f0/disk@w508002000055734a,0 (1/0/0) 2048 * the physical pathname of the device where the
boot program resides. The physical pathname is
Specify disk (enter its number): 12 (0/0/0) 0 * highly platform and configuration
selecting c14t508002000055734Ad0 dependent.
[disk formatted]
selecting c14t508002000055734Ad0
[disk formatted]
/dev/dsk/c14t508002000055734Ad0s4 is part of SVM volume stripe:d0.
Please see metaclear(1M).
/dev/dsk/c14t508002000055734Ad0s5 is part of SVM volume stripe:d1.
Please see metaclear(1M).
/dev/dsk/c14t508002000055734Ad0s6 is part of SVM volume stripe:d2.
Please see metaclear(1M).
/dev/dsk/c14t508002000055734Ad0s7 is part of SVM volume stripe:d3.
Please see metaclear(1M).

/etc/system
.....
Oracle Solaris 11 Express snv_151a X86
Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
Assembled 04 November 2010
.....
*ident "%Z%%M%" %I% %E% SMI" /* SVR4 1.5 */
*
* CDDL HEADER START
*
* The contents of this file are subject to the terms of the
* Common Development and Distribution License, Version 1.0 only
* (the "License"). You may not use this file except in compliance
* with the License.
*
* You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
* or http://www.opensolaris.org/os/licensing.
* See the License for the specific language governing permissions
* and limitations under the License.
*
* When distributing Covered Code, include this CDDL HEADER in each
* file and include the License file at usr/src/OPENSOLARIS.LICENSE.
* If applicable, add the following below this CDDL HEADER, with the
* fields enclosed by brackets "[]" replaced with your own identifying
* information: Portions Copyright [yyyy] [name of copyright owner]
*
* CDDL HEADER END
*
*
* SYSTEM SPECIFICATION FILE
*
* moddir:
*
* Set the search path for modules. This has a format similar to the
* csh path variable. If the module isn't found in the first directory
* it tries the second and so on. The default is /kernel /usr/kernel
*
* Example:
*
* set:
*
FORMAT MENU:
disk - select a disk
type - select (define) a disk type
partition - select (define) a partition table
current - describe the current disk
format - format and analyze the disk
fdisk - run the fdisk program
repair - repair a defective sector
label - write label to the disk
analyze - surface analysis
defect - defect list management
backup - search for backup labels
verify - read and display labels
save - save new disk/partition definitions
inquiry - show vendor, product and revision
volname - set 8-character volume name
!<cmd> - execute <cmd>, then return
quit
partition> p
Current partition table (original):
Total disk cylinders available: 23434 + 2 (reserved cylinders)

Part Tag Flag Cylinders Size
Blocks
0 unassigned wu 0 0
(0/0/0) 0
1 swap wu 1 - 49 49.00MB
(49/0/0) 100352

```

```

* Set an integer variable in the kernel or a module to a new value. # takes no effect /dev/md/rdsk/d120
* This facility should be used with caution. See system(4). # 600144f0e18d8e0000000000d01d121 15732768768
* # disable-sata-mpxio="no"; /dev/md/rdsk/d121
* Examples: ..... 600144f0e18d8e0000000000d01d122 15732768768
* system:0:::: /dev/md/rdsk/d122
* To set variables in 'unix': user.root:1:::: 600144f0e18d8e0000000000d01d123 15732768768
* noproject:2:::: /dev/md/rdsk/d123
* set nautopush=32 default:3:::: 600144f0e18d8e0000000000d01d124 15732768768
* set maxusers=40 group.staff:10:::: /dev/md/rdsk/d124
* ..... 600144f0e18d8e0000000000d01d125 15732768768
* To set a variable named 'debug' in the module named 'test_module' /dev/md/rdsk/d125
* 600144f0e18d8e0000000000d01d126 15732768768
* set test_module:debug = 0x13 /dev/md/rdsk/d126
* sbadm 600144f0e18d8e0000000000d01d127 15732768768
* Found 190 LU(s) /dev/md/rdsk/d127
* /dev/md/rdsk/d128
* 600144f0e18d8e0000000000d01d128 15732768768
* /dev/md/rdsk/d128
* ..... 600144f0e18d8e0000000000d01d129 15732768768
* /dev/md/rdsk/d129
* mpt.conf 600144f0e18d8e0000000000d01d101 15732768768
* # /dev/md/rdsk/d101 600144f0e18d8e0000000000d01d130 15732768768
* # Copyright 2008 Sun Microsystems, Inc. All rights reserved. /dev/md/rdsk/d102 600144f0e18d8e0000000000d01d102 15732768768
* # Use is subject to license terms. /dev/md/rdsk/d103 600144f0e18d8e0000000000d01d103 15732768768
* # /dev/md/rdsk/d104 600144f0e18d8e0000000000d01d104 15732768768
* # The mpt driver, as a pHCI driver, must specify the vHCI class it /dev/md/rdsk/d105 600144f0e18d8e0000000000d01d105 15732768768
* # belongs to( SCSI_vhci). /dev/md/rdsk/d106 600144f0e18d8e0000000000d01d106 15732768768
* # ddi-vhci-class="scsi_vhci"; /dev/md/rdsk/d107 600144f0e18d8e0000000000d01d107 15732768768
* # # I/O multipathing feature (MPxIO) can be enabled or disabled using /dev/md/rdsk/d108 600144f0e18d8e0000000000d01d108 15732768768
* # mpxio-disable property. Setting mpxio-disable="no" will activate /dev/md/rdsk/d109 600144f0e18d8e0000000000d01d109 15732768768
* # I/O multipathing; setting mpxio-disable="yes" disables the feature. /dev/md/rdsk/d110 600144f0e18d8e0000000000d01d110 15732768768
* # # Global mpxio-disable property: /dev/md/rdsk/d111 600144f0e18d8e0000000000d01d111 15732768768
* # # To globally enable MPxIO on all mpt controllers set: /dev/md/rdsk/d112 600144f0e18d8e0000000000d01d112 15732768768
* # mpxio-disable="no"; /dev/md/rdsk/d113 600144f0e18d8e0000000000d01d113 15732768768
* # # To globally disable MPxIO on all mpt controllers set: /dev/md/rdsk/d114 600144f0e18d8e0000000000d01d114 15732768768
* # mpxio-disable="yes"; /dev/md/rdsk/d115 600144f0e18d8e0000000000d01d115 15732768768
* # # You can also enable or disable MPxIO on a per HBA basis. /dev/md/rdsk/d116 600144f0e18d8e0000000000d01d116 15732768768
* # Per HBA settings override the global setting for the specified HBAs. /dev/md/rdsk/d117 600144f0e18d8e0000000000d01d117 15732768768
* # To disable MPxIO on a controller whose parent is /pci@7c0/pci@0/pci@9 /dev/md/rdsk/d118 600144f0e18d8e0000000000d01d118 15732768768
* # and the unit-address is "0" set: /dev/md/rdsk/d119 600144f0e18d8e0000000000d01d119 15732768768
* # name="mpt" parent="/pci@7c0/pci@0/pci@9" unit-address="0" mpxio- /dev/md/rdsk/d120 600144f0e18d8e0000000000d01d120 15732768768
* # disable="yes"; /dev/md/rdsk/d121 600144f0e18d8e0000000000d01d121 15732768768
* # mpxio-disable="yes"; /dev/md/rdsk/d122 600144f0e18d8e0000000000d01d122 15732768768
* # /dev/md/rdsk/d123 600144f0e18d8e0000000000d01d123 15732768768
* # # SATA mpxio supported /dev/md/rdsk/d124 600144f0e18d8e0000000000d01d124 15732768768
* # # To disable SATA mpxio, set /dev/md/rdsk/d125 600144f0e18d8e0000000000d01d125 15732768768
* # disable-sata-mpxio="yes"; /dev/md/rdsk/d126 600144f0e18d8e0000000000d01d126 15732768768
* # When mpxio-disable="yes" is set, the disable-sata-mpxio property /dev/md/rdsk/d127 600144f0e18d8e0000000000d01d127 15732768768

```



/dev/md/rdisk/d236	/dev/md/rdisk/d265	/dev/rdisk/c20t50800200005582C1d0s5
600144f0e18d8e0000000000d01d237 16466771968	600144f0e18d8e0000000000d01d266 16466771968	/dev/rdisk/c14t508002000055734Bd0s5
/dev/md/rdisk/d237	/dev/md/rdisk/d266	/dev/rdisk/c16t50800200005573C2d0s5
600144f0e18d8e0000000000d01d238 16466771968	600144f0e18d8e0000000000d01d267 16466771968	/dev/rdisk/c18t508002000055828Cd0s5
/dev/md/rdisk/d238	/dev/md/rdisk/d267	/dev/rdisk/c20t50800200005582C2d0s5
600144f0e18d8e0000000000d01d239 16466771968	600144f0e18d8e0000000000d01d268 16466771968	/dev/rdisk/c14t508002000055734Cd0s5
/dev/md/rdisk/d239	/dev/md/rdisk/d268	/dev/rdisk/c16t50800200005573C3d0s5
600144f0e18d8e0000000000d01d240 16466771968	600144f0e18d8e0000000000d01d269 16466771968	/dev/rdisk/c18t508002000055828Dd0s5
/dev/md/rdisk/d240	/dev/md/rdisk/d269	/dev/rdisk/c20t50800200005582C3d0s5
600144f0e18d8e0000000000d01d241 16466771968	600144f0e18d8e0000000000d01d270 16466771968	/dev/rdisk/c14t508002000055734Dd0s5
/dev/md/rdisk/d241	/dev/md/rdisk/d270	/dev/rdisk/c16t50800200005573C4d0s5
600144f0e18d8e0000000000d01d242 16466771968	600144f0e18d8e0000000000d01d271 16466771968	/dev/rdisk/c18t508002000055828Ed0s5
/dev/md/rdisk/d242	/dev/md/rdisk/d271	/dev/rdisk/c20t50800200005582C4d0s5
600144f0e18d8e0000000000d01d243 16466771968	600144f0e18d8e0000000000d01d272 16466771968	/dev/rdisk/c14t508002000055734Ed0s5
/dev/md/rdisk/d243	/dev/md/rdisk/d272	/dev/rdisk/c16t50800200005573C5d0s5
600144f0e18d8e0000000000d01d244 16466771968	600144f0e18d8e0000000000d01d273 16466771968	/dev/rdisk/c18t508002000055828Fd0s5
/dev/md/rdisk/d244	/dev/md/rdisk/d273	/dev/rdisk/c20t50800200005582C5d0s5
600144f0e18d8e0000000000d01d245 16466771968	600144f0e18d8e0000000000d01d274 16466771968	/dev/rdisk/c14t508002000055734Fd0s5
/dev/md/rdisk/d245	/dev/md/rdisk/d274	/dev/rdisk/c16t50800200005573C6d0s5
600144f0e18d8e0000000000d01d246 16466771968	600144f0e18d8e0000000000d01d275 16466771968	/dev/rdisk/c18t5080020000558280d0s5
/dev/md/rdisk/d246	/dev/md/rdisk/d275	/dev/rdisk/c20t50800200005582C6d0s5
600144f0e18d8e0000000000d01d247 16466771968	600144f0e18d8e0000000000d01d276 16466771968	/dev/rdisk/c14t5080020000557350d0s5
/dev/md/rdisk/d247	/dev/md/rdisk/d276	/dev/rdisk/c16t50800200005573C7d0s5
600144f0e18d8e0000000000d01d248 16466771968	600144f0e18d8e0000000000d01d277 16466771968	/dev/rdisk/c18t5080020000558281d0s5
/dev/md/rdisk/d248	/dev/md/rdisk/d277	/dev/rdisk/c20t50800200005582C7d0s5
600144f0e18d8e0000000000d01d249 16466771968	600144f0e18d8e0000000000d01d278 16466771968	/dev/rdisk/c14t5080020000557351d0s5
/dev/md/rdisk/d249	/dev/md/rdisk/d278	/dev/rdisk/c16t50800200005573C8d0s5
600144f0e18d8e0000000000d01d250 16466771968	600144f0e18d8e0000000000d01d279 16466771968	/dev/rdisk/c18t5080020000558282d0s5
/dev/md/rdisk/d250	/dev/md/rdisk/d279	/dev/rdisk/c20t50800200005582C8d0s5
600144f0e18d8e0000000000d01d251 16466771968	600144f0e18d8e0000000000d01d280 16466771968	/dev/rdisk/c14t5080020000557352d0s5
/dev/md/rdisk/d251	/dev/md/rdisk/d280	/dev/rdisk/c16t50800200005573C9d0s5
600144f0e18d8e0000000000d01d252 16466771968	600144f0e18d8e0000000000d01d281 16466771968	/dev/rdisk/c18t5080020000558283d0s5
/dev/md/rdisk/d252	/dev/md/rdisk/d281	/dev/rdisk/c20t50800200005582C9d0s5
600144f0e18d8e0000000000d01d253 16466771968	600144f0e18d8e0000000000d01d282 16466771968	/dev/rdisk/c14t5080020000557353d0s5
/dev/md/rdisk/d253	/dev/md/rdisk/d282	/dev/rdisk/c16t50800200005573CAd0s5
600144f0e18d8e0000000000d01d254 16466771968	600144f0e18d8e0000000000d01d283 16466771968	/dev/rdisk/c18t5080020000558285d0s5
/dev/md/rdisk/d254	/dev/md/rdisk/d283	/dev/rdisk/c20t50800200005582CA0s5
600144f0e18d8e0000000000d01d255 16466771968	600144f0e18d8e0000000000d01d284 16466771968	/dev/rdisk/c14t5080020000557347d0s5
/dev/md/rdisk/d255	/dev/md/rdisk/d284	/dev/rdisk/c16t50800200005573CBd0s5
600144f0e18d8e0000000000d01d256 16466771968	600144f0e18d8e0000000000d01d285 528416768	/dev/rdisk/c18t5080020000558286d0s5
/dev/md/rdisk/d256	600144f0e18d8e0000000000d01d286 528416768	/dev/md/rdisk/d286/dev/rdisk/c20t50800200005582CBd0s5
600144f0e18d8e0000000000d01d257 16466771968	600144f0e18d8e0000000000d01d287 528416768	/dev/md/rdisk/d287/dev/rdisk/c14t5080020000557340d0s5
/dev/md/rdisk/d257	600144f0e18d8e0000000000d01d288 528416768	/dev/md/rdisk/d288/dev/rdisk/c16t50800200005573CCd0s5
600144f0e18d8e0000000000d01d258 16466771968	600144f0e18d8e0000000000d01d289 1262419968	/dev/rdisk/c18t5080020000558287d0s5
/dev/md/rdisk/d258	/dev/md/rdisk/d289	/dev/rdisk/c20t50800200005582CCd0s5
600144f0e18d8e0000000000d01d259 16466771968	600144f0e18d8e0000000000d01d290 209649664	/dev/md/rdisk/d290/dev/rdisk/c14t5080020000557341d0s5
/dev/md/rdisk/d259		/dev/rdisk/c16t50800200005573CDd0s5
600144f0e18d8e0000000000d01d260 16466771968		/dev/rdisk/c18t5080020000558288d0s5
/dev/md/rdisk/d260		/dev/rdisk/c20t50800200005582CDd0s5
600144f0e18d8e0000000000d01d261 16466771968	<b>metastat</b>	/dev/rdisk/c14t5080020000557342d0s5
/dev/md/rdisk/d261		/dev/rdisk/c16t50800200005573CEd0s5
600144f0e18d8e0000000000d01d262 16466771968	d290 -p /dev/md/rdisk/d1 -o 1817997312 -b 409600	/dev/rdisk/c18t5080020000558289d0s5
/dev/md/rdisk/d262	d1 l 160 /dev/rdisk/c16t50800200005573C0d0s5	/dev/rdisk/c20t50800200005582CEd0s5
600144f0e18d8e0000000000d01d263 16466771968	/dev/rdisk/c18t508002000055828Ad0s5	/dev/rdisk/c14t5080020000557343d0s5
/dev/md/rdisk/d263	/dev/rdisk/c20t50800200005582C0d0s5	/dev/rdisk/c16t50800200005573CFd0s5
600144f0e18d8e0000000000d01d264 16466771968	/dev/rdisk/c14t508002000055734Ad0s5	/dev/rdisk/c18t5080020000558290d0s5
/dev/md/rdisk/d264	/dev/rdisk/c16t50800200005573C1d0s5	/dev/rdisk/c20t50800200005582CFd0s5
600144f0e18d8e0000000000d01d265 16466771968	/dev/rdisk/c18t508002000055828Bd0s5	/dev/rdisk/c14t5080020000557344d0s5

```

/dev/rdisk/c16t50800200005573D0d0s5
/dev/rdisk/c18t5080020000558291d0s5
/dev/rdisk/c20t50800200005582D0d0s5
/dev/rdisk/c14t5080020000557345d0s5
/dev/rdisk/c16t50800200005573D1d0s5
/dev/rdisk/c18t5080020000558292d0s5
/dev/rdisk/c20t50800200005582D1d0s5
/dev/rdisk/c14t5080020000557346d0s5
/dev/rdisk/c16t50800200005573D2d0s5
/dev/rdisk/c18t5080020000558293d0s5
/dev/rdisk/c20t50800200005582D2d0s5
/dev/rdisk/c14t5080020000557349d0s5
/dev/rdisk/c16t50800200005573D3d0s5
/dev/rdisk/c18t5080020000558284d0s5
/dev/rdisk/c20t50800200005582D3d0s5
/dev/rdisk/c14t5080020000557348d0s5
/dev/rdisk/c17t508002000055730Ad0s5
/dev/rdisk/c19t508002000055820Ad0s5
/dev/rdisk/c21t508002000055824Ad0s5
/dev/rdisk/c15t508002000055738Ad0s5
/dev/rdisk/c17t508002000055730Bd0s5
/dev/rdisk/c19t508002000055820Bd0s5
/dev/rdisk/c21t508002000055824Bd0s5
/dev/rdisk/c15t508002000055738Bd0s5
/dev/rdisk/c17t508002000055730Cd0s5
/dev/rdisk/c19t508002000055820Cd0s5
/dev/rdisk/c21t508002000055824Cd0s5
/dev/rdisk/c15t508002000055738Cd0s5
/dev/rdisk/c17t508002000055730Dd0s5
/dev/rdisk/c19t508002000055820Dd0s5
/dev/rdisk/c21t508002000055824Dd0s5
/dev/rdisk/c15t508002000055738Dd0s5
/dev/rdisk/c17t508002000055730Ed0s5
/dev/rdisk/c19t508002000055820Ed0s5
/dev/rdisk/c21t508002000055824Ed0s5
/dev/rdisk/c15t508002000055738Ed0s5
/dev/rdisk/c17t508002000055730Fd0s5
/dev/rdisk/c19t508002000055820Fd0s5
/dev/rdisk/c21t508002000055824Fd0s5
/dev/rdisk/c15t508002000055738Fd0s5
/dev/rdisk/c17t5080020000557300d0s5
/dev/rdisk/c19t5080020000558210d0s5
/dev/rdisk/c21t5080020000558245d0s5
/dev/rdisk/c15t5080020000557393d0s5
/dev/rdisk/c17t5080020000557306d0s5
/dev/rdisk/c19t5080020000558211d0s5
/dev/rdisk/c21t5080020000558246d0s5
/dev/rdisk/c15t5080020000557380d0s5
/dev/rdisk/c17t5080020000557307d0s5
/dev/rdisk/c19t5080020000558212d0s5
/dev/rdisk/c21t5080020000558247d0s5
/dev/rdisk/c15t5080020000557388d0s5
/dev/rdisk/c17t5080020000557308d0s5
/dev/rdisk/c19t5080020000558213d0s5
/dev/rdisk/c21t5080020000558248d0s5
/dev/rdisk/c15t5080020000557387d0s5
/dev/rdisk/c17t5080020000557309d0s5
/dev/rdisk/c19t5080020000558209d0s5
/dev/rdisk/c21t5080020000558249d0s5
/dev/rdisk/c15t5080020000557386d0s5
/dev/rdisk/c17t5080020000557311d0s5
/dev/rdisk/c19t5080020000558208d0s5
/dev/rdisk/c21t5080020000558250d0s5
/dev/rdisk/c15t5080020000557312d0s5
/dev/rdisk/c17t5080020000557320d0s5
/dev/rdisk/c19t5080020000558207d0s5
/dev/rdisk/c21t5080020000558251d0s5
/dev/rdisk/c15t5080020000557384d0s5
/dev/rdisk/c17t5080020000557313d0s5
/dev/rdisk/c19t5080020000558206d0s5
/dev/rdisk/c21t5080020000558253d0s5
/dev/rdisk/c15t5080020000557383d0s5
/dev/rdisk/c17t5080020000557305d0s5
/dev/rdisk/c19t5080020000558205d0s5
/dev/rdisk/c21t5080020000558244d0s5
/dev/rdisk/c15t5080020000557382d0s5
/dev/rdisk/c17t5080020000557304d0s5
/dev/rdisk/c19t5080020000558204d0s5
/dev/rdisk/c21t5080020000558243d0s5
/dev/rdisk/c15t5080020000557385d0s5
/dev/rdisk/c17t5080020000557303d0s5
/dev/rdisk/c19t5080020000558203d0s5
/dev/rdisk/c21t5080020000558242d0s5
/dev/rdisk/c15t5080020000557390d0s5
/dev/rdisk/c17t5080020000557310d0s5
/dev/rdisk/c19t5080020000558202d0s5
/dev/rdisk/c21t5080020000558241d0s5
/dev/rdisk/c15t5080020000557391d0s5
/dev/rdisk/c17t5080020000557301d0s5
/dev/rdisk/c19t5080020000558201d0s5
/dev/rdisk/c21t5080020000558240d0s5
/dev/rdisk/c15t5080020000557392d0s5
/dev/rdisk/c17t5080020000557302d0s5
/dev/rdisk/c19t5080020000558200d0s5
/dev/rdisk/c21t5080020000558252d0s5
/dev/rdisk/c15t5080020000557381d0s5 -i 256b
d3285 -p /dev/md/rdisk/d8001 -o 2743988224 -b 1032192
d8001 -m /dev/md/rdisk/d8012 /dev/md/rdisk/d8013 0
d8012 1 1 /dev/rdisk/c7t4d0s6
d8013 1 1 /dev/rdisk/c7t5d0s6
d3286 -p /dev/md/rdisk/d8002 -o 2921345024 -b 1032192
d8002 -m /dev/md/rdisk/d8014 /dev/md/rdisk/d8015 0
d8014 1 1 /dev/rdisk/c7t6d0s6
d8015 1 1 /dev/rdisk/c7t7d0s6
d3287 -p /dev/md/rdisk/d8003 -o 2907828224 -b 1032192
d8003 -m /dev/md/rdisk/d8016 /dev/md/rdisk/d8017 0
d8016 1 1 /dev/rdisk/c7t8d0s6
d8017 1 1 /dev/rdisk/c7t9d0s6
d3288 -p /dev/md/rdisk/d8004 -o 2848436224 -b 1032192
d8004 -m /dev/md/rdisk/d8018 /dev/md/rdisk/d8019 0
d8018 1 1 /dev/rdisk/c7t10d0s6
d8019 1 1 /dev/rdisk/c7t11d0s6
d3289 -p /dev/md/rdisk/d8004 -o 2845969408 -b 2465792
d3209 -p /dev/md/rdisk/d8003 -o 2875665408 -b 32161792
d3210 -p /dev/md/rdisk/d8002 -o 2889182208 -b 32161792
d3212 -p /dev/md/rdisk/d8000 -o 3091310592 -b 32161792
d8000 -m /dev/md/rdisk/d8010 /dev/md/rdisk/d8011 0
d8010 1 1 /dev/rdisk/c7t1d0s6
d8011 1 1 /dev/rdisk/c7t3d0s6
d3213 -p /dev/md/rdisk/d8000 -o 3059147776 -b 32161792
d3214 -p /dev/md/rdisk/d8001 -o 2679662592 -b 32161792
d3215 -p /dev/md/rdisk/d8002 -o 2857019392 -b 32161792
d3216 -p /dev/md/rdisk/d8003 -o 2843502592 -b 32161792
d289 -p /dev/md/rdisk/d0 -o 1817997312 -b 2465792
d0 1 160 /dev/rdisk/c14t508002000055734Ad0s4
/dev/rdisk/c16t50800200005573C0d0s4
/dev/rdisk/c18t508002000055828Ad0s4
/dev/rdisk/c20t50800200005582C0d0s4
/dev/rdisk/c14t508002000055734Bd0s4
/dev/rdisk/c16t50800200005573C1d0s4
/dev/rdisk/c18t508002000055828Bd0s4
/dev/rdisk/c20t50800200005582C1d0s4
/dev/rdisk/c14t508002000055734Cd0s4
/dev/rdisk/c16t50800200005573C2d0s4
/dev/rdisk/c18t508002000055828Cd0s4
/dev/rdisk/c20t50800200005582C2d0s4
/dev/rdisk/c14t508002000055734Dd0s4
/dev/rdisk/c16t50800200005573C3d0s4
/dev/rdisk/c18t508002000055828Dd0s4
/dev/rdisk/c20t50800200005582C3d0s4
/dev/rdisk/c14t508002000055734Ed0s4
/dev/rdisk/c16t50800200005573C4d0s4
/dev/rdisk/c18t508002000055828Ed0s4
/dev/rdisk/c20t50800200005582C4d0s4
/dev/rdisk/c14t508002000055734Fd0s4
/dev/rdisk/c16t50800200005573C5d0s4
/dev/rdisk/c18t508002000055828Fd0s4
/dev/rdisk/c20t50800200005582C5d0s4
/dev/rdisk/c14t5080020000557350d0s4
/dev/rdisk/c16t50800200005573C6d0s4
/dev/rdisk/c18t5080020000558280d0s4
/dev/rdisk/c20t50800200005582C6d0s4
/dev/rdisk/c14t5080020000557351d0s4
/dev/rdisk/c16t50800200005573C7d0s4
/dev/rdisk/c18t5080020000558281d0s4
/dev/rdisk/c20t50800200005582C7d0s4
/dev/rdisk/c14t5080020000557352d0s4
/dev/rdisk/c16t50800200005573C8d0s4
/dev/rdisk/c18t5080020000558282d0s4
/dev/rdisk/c20t50800200005582C8d0s4
/dev/rdisk/c14t5080020000557353d0s4
/dev/rdisk/c16t50800200005573C9d0s4
/dev/rdisk/c18t5080020000558283d0s4
/dev/rdisk/c20t50800200005582C9d0s4
/dev/rdisk/c14t508002000055734d0s4
/dev/rdisk/c16t50800200005573CAd0s4
/dev/rdisk/c18t5080020000558285d0s4
/dev/rdisk/c20t50800200005582CAd0s4
/dev/rdisk/c14t5080020000557340d0s4
/dev/rdisk/c16t50800200005573CBd0s4
/dev/rdisk/c18t5080020000558286d0s4
/dev/rdisk/c20t50800200005582CBd0s4
/dev/rdisk/c14t5080020000557341d0s4
/dev/rdisk/c16t50800200005573CCd0s4

```







/dev/rdskc/c20t50800200005582D2d0s6	/dev/rdskc/c17t5080020000557312d0s6	d3244 -p /dev/md/rdsk/d8001 -o 2486685696 -b 32161792
/dev/rdskc/c14t5080020000557349d0s6	/dev/rdskc/c19t5080020000558206d0s6	d3245 -p /dev/md/rdsk/d8002 -o 2664042496 -b 32161792
/dev/rdskc/c16t50800200005573D2d0s6	/dev/rdskc/c21t5080020000558253d0s6	d3246 -p /dev/md/rdsk/d8003 -o 2650525696 -b 32161792
/dev/rdskc/c18t5080020000558284d0s6	/dev/rdskc/c15t5080020000557383d0s6	d3247 -p /dev/md/rdsk/d8004 -o 2620829696 -b 32161792
/dev/rdskc/c20t50800200005582D3d0s6	/dev/rdskc/c17t5080020000557313d0s6	d3248 -p /dev/md/rdsk/d8004 -o 2588666880 -b 32161792
/dev/rdskc/c14t5080020000557348d0s6	/dev/rdskc/c19t5080020000558205d0s6	d3249 -p /dev/md/rdsk/d8003 -o 2618362880 -b 32161792
/dev/rdskc/c16t50800200005573D3d0s6	/dev/rdskc/c21t5080020000558244d0s6	d3250 -p /dev/md/rdsk/d8002 -o 2631879680 -b 32161792
/dev/rdskc/c19t508002000055820Ad0s6	/dev/rdskc/c15t5080020000557382d0s6	d3251 -p /dev/md/rdsk/d8001 -o 2454522880 -b 32161792
/dev/rdskc/c21t508002000055824Ad0s6	/dev/rdskc/c17t5080020000557305d0s6	d3252 -p /dev/md/rdsk/d8000 -o 2834008064 -b 32161792
/dev/rdskc/c15t508002000055738Ad0s6	/dev/rdskc/c19t5080020000558204d0s6	d3253 -p /dev/md/rdsk/d8000 -o 2801845248 -b 32161792
/dev/rdskc/c17t508002000055730Ad0s6	/dev/rdskc/c21t5080020000558243d0s6	d3254 -p /dev/md/rdsk/d8001 -o 2422360064 -b 32161792
/dev/rdskc/c19t508002000055820Bd0s6	/dev/rdskc/c15t5080020000557385d0s6	d3255 -p /dev/md/rdsk/d8002 -o 2599716864 -b 32161792
/dev/rdskc/c21t508002000055824Bd0s6	/dev/rdskc/c17t5080020000557304d0s6	d3256 -p /dev/md/rdsk/d8003 -o 2586200064 -b 32161792
/dev/rdskc/c15t508002000055738Bd0s6	/dev/rdskc/c19t5080020000558203d0s6	d3257 -p /dev/md/rdsk/d8004 -o 2556504064 -b 32161792
/dev/rdskc/c17t508002000055730Bd0s6	/dev/rdskc/c21t5080020000558242d0s6	d3258 -p /dev/md/rdsk/d8004 -o 2524341248 -b 32161792
/dev/rdskc/c19t508002000055820Cd0s6	/dev/rdskc/c15t5080020000557390d0s6	d3259 -p /dev/md/rdsk/d8003 -o 2554037248 -b 32161792
/dev/rdskc/c21t508002000055824Cd0s6	/dev/rdskc/c17t5080020000557303d0s6	d3260 -p /dev/md/rdsk/d8002 -o 2567554048 -b 32161792
/dev/rdskc/c15t508002000055738Cd0s6	/dev/rdskc/c19t5080020000558202d0s6	d3261 -p /dev/md/rdsk/d8001 -o 2390197248 -b 32161792
/dev/rdskc/c17t508002000055730Cd0s6	/dev/rdskc/c21t5080020000558241d0s6	d3262 -p /dev/md/rdsk/d8000 -o 2769682432 -b 32161792
/dev/rdskc/c19t508002000055820Dd0s6	/dev/rdskc/c15t5080020000557391d0s6	d3263 -p /dev/md/rdsk/d8000 -o 2737519616 -b 32161792
/dev/rdskc/c21t508002000055824Dd0s6	/dev/rdskc/c17t5080020000557310d0s6	d3264 -p /dev/md/rdsk/d8001 -o 2358034432 -b 32161792
/dev/rdskc/c15t508002000055738Dd0s6	/dev/rdskc/c19t5080020000558201d0s6	d3265 -p /dev/md/rdsk/d8002 -o 2535391232 -b 32161792
/dev/rdskc/c17t508002000055730Dd0s6	/dev/rdskc/c21t5080020000558240d0s6	d3266 -p /dev/md/rdsk/d8003 -o 2521874432 -b 32161792
/dev/rdskc/c19t508002000055820Ed0s6	/dev/rdskc/c15t5080020000557392d0s6	d3267 -p /dev/md/rdsk/d8004 -o 2492178432 -b 32161792
/dev/rdskc/c21t508002000055824Ed0s6	/dev/rdskc/c17t5080020000557301d0s6	d3268 -p /dev/md/rdsk/d8004 -o 2460015616 -b 32161792
/dev/rdskc/c15t508002000055738Ed0s6	/dev/rdskc/c19t5080020000558200d0s6	d3269 -p /dev/md/rdsk/d8003 -o 2489711616 -b 32161792
/dev/rdskc/c17t508002000055730Ed0s6	/dev/rdskc/c21t5080020000558252d0s6	d3270 -p /dev/md/rdsk/d8002 -o 2503228416 -b 32161792
/dev/rdskc/c19t508002000055820Fd0s6	/dev/rdskc/c15t5080020000557381d0s6	d3271 -p /dev/md/rdsk/d8001 -o 2325871616 -b 32161792
/dev/rdskc/c21t508002000055824Fd0s6	/dev/rdskc/c17t5080020000557302d0s6 -i 256b	d3272 -p /dev/md/rdsk/d8000 -o 2705356800 -b 32161792
/dev/rdskc/c15t508002000055738Fd0s6	d286 -p /dev/md/rdsk/d1 -o 1816964864 -b 1032192	d3273 -p /dev/md/rdsk/d8000 -o 2673193984 -b 32161792
/dev/rdskc/c17t508002000055730Fd0s6	d285 -p /dev/md/rdsk/d0 -o 1816964864 -b 1032192	d3274 -p /dev/md/rdsk/d8001 -o 2293708800 -b 32161792
/dev/rdskc/c19t5080020000558210d0s6	d284 -p /dev/md/rdsk/d3 -o 1784802816 -b 32161792	d3275 -p /dev/md/rdsk/d8002 -o 2471065600 -b 32161792
/dev/rdskc/c21t5080020000558245d0s6	d283 -p /dev/md/rdsk/d2 -o 1784802816 -b 32161792	d3276 -p /dev/md/rdsk/d8003 -o 2457548800 -b 32161792
/dev/rdskc/c15t5080020000557393d0s6	d3219 -p /dev/md/rdsk/d8003 -o 2811339776 -b 32161792	d3277 -p /dev/md/rdsk/d8004 -o 2427852800 -b 32161792
/dev/rdskc/c17t5080020000557300d0s6	d3220 -p /dev/md/rdsk/d8002 -o 2824856576 -b 32161792	d3278 -p /dev/md/rdsk/d8004 -o 2395689984 -b 32161792
/dev/rdskc/c19t5080020000558211d0s6	d3221 -p /dev/md/rdsk/d8001 -o 2647499776 -b 32161792	d3279 -p /dev/md/rdsk/d8003 -o 2425385984 -b 32161792
/dev/rdskc/c21t5080020000558246d0s6	d3222 -p /dev/md/rdsk/d8000 -o 3026984960 -b 32161792	d3280 -p /dev/md/rdsk/d8002 -o 2438902784 -b 32161792
/dev/rdskc/c15t5080020000557380d0s6	d3223 -p /dev/md/rdsk/d8000 -o 2994822144 -b 32161792	d3281 -p /dev/md/rdsk/d8001 -o 2261545984 -b 32161792
/dev/rdskc/c17t5080020000557306d0s6	d3224 -p /dev/md/rdsk/d8001 -o 2615336960 -b 32161792	d3282 -p /dev/md/rdsk/d8000 -o 2641031168 -b 32161792
/dev/rdskc/c19t5080020000558212d0s6	d3225 -p /dev/md/rdsk/d8002 -o 2792693760 -b 32161792	d3283 -p /dev/md/rdsk/d8000 -o 2608868352 -b 32161792
/dev/rdskc/c21t5080020000558247d0s6	d3226 -p /dev/md/rdsk/d8003 -o 2779176960 -b 32161792	d3284 -p /dev/md/rdsk/d8001 -o 2229383168 -b 32161792
/dev/rdskc/c15t5080020000557388d0s6	d3227 -p /dev/md/rdsk/d8004 -o 2749480960 -b 32161792	d3200 -p /dev/md/rdsk/d8002 -o 2318061568 -b 120840192
/dev/rdskc/c17t5080020000557307d0s6	d3228 -p /dev/md/rdsk/d8004 -o 2717318144 -b 32161792	d3201 -p /dev/md/rdsk/d8003 -o 2304544768 -b 120840192
/dev/rdskc/c19t5080020000558213d0s6	d3229 -p /dev/md/rdsk/d8003 -o 2747014144 -b 32161792	d3202 -p /dev/md/rdsk/d8004 -o 2274848768 -b 120840192
/dev/rdskc/c21t5080020000558248d0s6	d3230 -p /dev/md/rdsk/d8002 -o 2760530944 -b 32161792	d3203 -p /dev/md/rdsk/d8004 -o 2154007552 -b 120840192
/dev/rdskc/c15t5080020000557387d0s6	d3231 -p /dev/md/rdsk/d8001 -o 2583174144 -b 32161792	d3204 -p /dev/md/rdsk/d8003 -o 2183703552 -b 120840192
/dev/rdskc/c17t5080020000557308d0s6	d3232 -p /dev/md/rdsk/d8000 -o 2962659328 -b 32161792	d3205 -p /dev/md/rdsk/d8002 -o 2197220352 -b 120840192
/dev/rdskc/c19t5080020000558209d0s6	d3233 -p /dev/md/rdsk/d8000 -o 2930496512 -b 32161792	d3206 -p /dev/md/rdsk/d8001 -o 2108541952 -b 120840192
/dev/rdskc/c21t5080020000558249d0s6	d3234 -p /dev/md/rdsk/d8001 -o 2551011328 -b 32161792	d3207 -p /dev/md/rdsk/d8000 -o 2488027136 -b 120840192
/dev/rdskc/c15t5080020000557386d0s6	d3235 -p /dev/md/rdsk/d8002 -o 2728368128 -b 32161792	d3208 -p /dev/md/rdsk/d8000 -o 2367185920 -b 120840192
/dev/rdskc/c17t5080020000557309d0s6	d3236 -p /dev/md/rdsk/d8003 -o 2714851328 -b 32161792	d3185 -p /dev/md/rdsk/d8001 -o 2101159936 -b 7380992
/dev/rdskc/c19t5080020000558208d0s6	d3237 -p /dev/md/rdsk/d8004 -o 2685155328 -b 32161792	d3186 -p /dev/md/rdsk/d8002 -o 2189838336 -b 7380992
/dev/rdskc/c21t5080020000558250d0s6	d3238 -p /dev/md/rdsk/d8004 -o 2652992512 -b 32161792	d3187 -p /dev/md/rdsk/d8003 -o 2176321536 -b 7380992
/dev/rdskc/c15t5080020000557389d0s6	d3239 -p /dev/md/rdsk/d8003 -o 2682688512 -b 32161792	d3188 -p /dev/md/rdsk/d8004 -o 2146625536 -b 7380992
/dev/rdskc/c17t5080020000557311d0s6	d3240 -p /dev/md/rdsk/d8002 -o 2696205312 -b 32161792	d3173 -p /dev/md/rdsk/d8004 -o 2132075520 -b 14548992
/dev/rdskc/c19t5080020000558207d0s6	d3241 -p /dev/md/rdsk/d8001 -o 2518848512 -b 32161792	d3174 -p /dev/md/rdsk/d8003 -o 2161771520 -b 14548992
/dev/rdskc/c21t5080020000558251d0s6	d3242 -p /dev/md/rdsk/d8000 -o 2898333696 -b 32161792	d3175 -p /dev/md/rdsk/d8002 -o 2175288320 -b 14548992
/dev/rdskc/c15t5080020000557384d0s6	d3243 -p /dev/md/rdsk/d8000 -o 2866170880 -b 32161792	d3176 -p /dev/md/rdsk/d8001 -o 2086609920 -b 14548992

d3177 -p /dev/md/rdsk/d8000 -o 2352635904 -b 14548992  
d3178 -p /dev/md/rdsk/d8000 -o 2338085888 -b 14548992  
d3179 -p /dev/md/rdsk/d8001 -o 2072059904 -b 14548992  
d3180 -p /dev/md/rdsk/d8002 -o 2160738304 -b 14548992  
d3165 -p /dev/md/rdsk/d8003 -o 2146607104 -b 15163392  
d3166 -p /dev/md/rdsk/d8004 -o 2116911104 -b 15163392  
d3167 -p /dev/md/rdsk/d8004 -o 2101746688 -b 15163392  
d3168 -p /dev/md/rdsk/d8003 -o 2131442688 -b 15163392  
d3157 -p /dev/md/rdsk/d8002 -o 2138201088 -b 22536192  
d3158 -p /dev/md/rdsk/d8001 -o 2049522688 -b 22536192  
d3159 -p /dev/md/rdsk/d8000 -o 2315548672 -b 22536192  
d3160 -p /dev/md/rdsk/d8000 -o 2293011456 -b 22536192  
d3161 -p /dev/md/rdsk/d8001 -o 2026985472 -b 22536192  
d3162 -p /dev/md/rdsk/d8002 -o 2115663872 -b 22536192  
d3163 -p /dev/md/rdsk/d8003 -o 2108905472 -b 22536192  
d3164 -p /dev/md/rdsk/d8004 -o 2079209472 -b 22536192  
d3101 -p /dev/md/rdsk/d8004 -o 2048480256 -b 30728192  
d3102 -p /dev/md/rdsk/d8003 -o 2078176256 -b 30728192  
d3103 -p /dev/md/rdsk/d8002 -o 2084934656 -b 30728192  
d3104 -p /dev/md/rdsk/d8001 -o 1996256256 -b 30728192  
d3105 -p /dev/md/rdsk/d8000 -o 2262282240 -b 30728192  
d3106 -p /dev/md/rdsk/d8000 -o 2231553024 -b 30728192  
d3107 -p /dev/md/rdsk/d8001 -o 1965527040 -b 30728192  
d3108 -p /dev/md/rdsk/d8002 -o 2054205440 -b 30728192  
d3109 -p /dev/md/rdsk/d8003 -o 2047447040 -b 30728192  
d3110 -p /dev/md/rdsk/d8004 -o 2017751040 -b 30728192  
d3111 -p /dev/md/rdsk/d8004 -o 1987021824 -b 30728192  
d3112 -p /dev/md/rdsk/d8003 -o 2016717824 -b 30728192  
d3113 -p /dev/md/rdsk/d8002 -o 2023476224 -b 30728192  
d3114 -p /dev/md/rdsk/d8001 -o 1934797824 -b 30728192  
d3115 -p /dev/md/rdsk/d8000 -o 2200823808 -b 30728192  
d3116 -p /dev/md/rdsk/d8000 -o 2170094592 -b 30728192  
d3117 -p /dev/md/rdsk/d8001 -o 1904068608 -b 30728192  
d3118 -p /dev/md/rdsk/d8002 -o 1992747008 -b 30728192  
d3119 -p /dev/md/rdsk/d8003 -o 1985988608 -b 30728192  
d3120 -p /dev/md/rdsk/d8004 -o 1956292608 -b 30728192  
d3121 -p /dev/md/rdsk/d8004 -o 1925563392 -b 30728192  
d3122 -p /dev/md/rdsk/d8003 -o 1955259392 -b 30728192  
d3123 -p /dev/md/rdsk/d8002 -o 1962017792 -b 30728192  
d3124 -p /dev/md/rdsk/d8001 -o 1873339392 -b 30728192  
d3125 -p /dev/md/rdsk/d8000 -o 2139365376 -b 30728192  
d3126 -p /dev/md/rdsk/d8000 -o 2108636160 -b 30728192  
d3127 -p /dev/md/rdsk/d8001 -o 1842610176 -b 30728192  
d3128 -p /dev/md/rdsk/d8002 -o 1931288576 -b 30728192  
d3129 -p /dev/md/rdsk/d8003 -o 1924530176 -b 30728192  
d3130 -p /dev/md/rdsk/d8004 -o 1894834176 -b 30728192  
d3131 -p /dev/md/rdsk/d8004 -o 1864104960 -b 30728192  
d3132 -p /dev/md/rdsk/d8003 -o 1893800960 -b 30728192  
d3133 -p /dev/md/rdsk/d8002 -o 1900559360 -b 30728192  
d3134 -p /dev/md/rdsk/d8001 -o 1811880960 -b 30728192  
d3135 -p /dev/md/rdsk/d8000 -o 2077906944 -b 30728192  
d3136 -p /dev/md/rdsk/d8000 -o 2047177728 -b 30728192  
d3137 -p /dev/md/rdsk/d8001 -o 1781151744 -b 30728192  
d3138 -p /dev/md/rdsk/d8002 -o 1869830144 -b 30728192  
d3139 -p /dev/md/rdsk/d8003 -o 1863071744 -b 30728192  
d3140 -p /dev/md/rdsk/d8004 -o 1833375744 -b 30728192  
d3141 -p /dev/md/rdsk/d8004 -o 1802646528 -b 30728192  
d3142 -p /dev/md/rdsk/d8003 -o 1832342528 -b 30728192  
d3143 -p /dev/md/rdsk/d8002 -o 1839100928 -b 30728192  
d3144 -p /dev/md/rdsk/d8001 -o 1750422528 -b 30728192  
d3145 -p /dev/md/rdsk/d8000 -o 2016448512 -b 30728192  
d3146 -p /dev/md/rdsk/d8000 -o 1985719296 -b 30728192  
d3147 -p /dev/md/rdsk/d8001 -o 1719693312 -b 30728192  
d3148 -p /dev/md/rdsk/d8002 -o 1808371712 -b 30728192  
d3149 -p /dev/md/rdsk/d8003 -o 1801613312 -b 30728192  
d3150 -p /dev/md/rdsk/d8004 -o 1771917312 -b 30728192  
d3151 -p /dev/md/rdsk/d8004 -o 1741188096 -b 30728192  
d3152 -p /dev/md/rdsk/d8003 -o 1770884096 -b 30728192  
d3153 -p /dev/md/rdsk/d8002 -o 1777642496 -b 30728192  
d3154 -p /dev/md/rdsk/d8001 -o 1688964096 -b 30728192  
d3155 -p /dev/md/rdsk/d8000 -o 1954990080 -b 30728192  
d3156 -p /dev/md/rdsk/d8000 -o 1924260864 -b 30728192  
d3169 -p /dev/md/rdsk/d8001 -o 1657210880 -b 31752192  
d3170 -p /dev/md/rdsk/d8002 -o 1745889280 -b 31752192  
d3171 -p /dev/md/rdsk/d8003 -o 1739130880 -b 31752192  
d3172 -p /dev/md/rdsk/d8004 -o 1709434880 -b 31752192  
d3181 -p /dev/md/rdsk/d8004 -o 1666417664 -b 43016192  
d3182 -p /dev/md/rdsk/d8003 -o 1696113664 -b 43016192  
d3183 -p /dev/md/rdsk/d8002 -o 1702872064 -b 43016192  
d3184 -p /dev/md/rdsk/d8001 -o 1614193664 -b 43016192  
d3189 -p /dev/md/rdsk/d8000 -o 1803419648 -b 120840192  
d3190 -p /dev/md/rdsk/d8000 -o 1682578432 -b 120840192  
d282 -p /dev/md/rdsk/d1 -o 1784802816 -b 32161792  
d281 -p /dev/md/rdsk/d0 -o 1784802816 -b 32161792  
d280 -p /dev/md/rdsk/d3 -o 1752640768 -b 32161792  
d279 -p /dev/md/rdsk/d2 -o 1752640768 -b 32161792  
d278 -p /dev/md/rdsk/d1 -o 1752640768 -b 32161792  
d277 -p /dev/md/rdsk/d0 -o 1752640768 -b 32161792  
d276 -p /dev/md/rdsk/d3 -o 1720478720 -b 32161792  
d275 -p /dev/md/rdsk/d2 -o 1720478720 -b 32161792  
d274 -p /dev/md/rdsk/d1 -o 1720478720 -b 32161792  
d273 -p /dev/md/rdsk/d0 -o 1720478720 -b 32161792  
d272 -p /dev/md/rdsk/d3 -o 1688316672 -b 32161792  
d271 -p /dev/md/rdsk/d2 -o 1688316672 -b 32161792  
d270 -p /dev/md/rdsk/d1 -o 1688316672 -b 32161792  
d269 -p /dev/md/rdsk/d0 -o 1688316672 -b 32161792  
d268 -p /dev/md/rdsk/d3 -o 1656154624 -b 32161792  
d267 -p /dev/md/rdsk/d2 -o 1656154624 -b 32161792  
d266 -p /dev/md/rdsk/d1 -o 1656154624 -b 32161792  
d265 -p /dev/md/rdsk/d0 -o 1656154624 -b 32161792  
d264 -p /dev/md/rdsk/d3 -o 1623992576 -b 32161792  
d263 -p /dev/md/rdsk/d2 -o 1623992576 -b 32161792  
d262 -p /dev/md/rdsk/d1 -o 1623992576 -b 32161792  
d261 -p /dev/md/rdsk/d0 -o 1623992576 -b 32161792  
d260 -p /dev/md/rdsk/d3 -o 1591830528 -b 32161792  
d259 -p /dev/md/rdsk/d2 -o 1591830528 -b 32161792  
d258 -p /dev/md/rdsk/d1 -o 1591830528 -b 32161792  
d257 -p /dev/md/rdsk/d0 -o 1591830528 -b 32161792  
d256 -p /dev/md/rdsk/d3 -o 1559668480 -b 32161792  
d255 -p /dev/md/rdsk/d2 -o 1559668480 -b 32161792  
d254 -p /dev/md/rdsk/d1 -o 1559668480 -b 32161792  
d253 -p /dev/md/rdsk/d0 -o 1559668480 -b 32161792  
d252 -p /dev/md/rdsk/d3 -o 1527506432 -b 32161792  
d251 -p /dev/md/rdsk/d2 -o 1527506432 -b 32161792  
d250 -p /dev/md/rdsk/d1 -o 1527506432 -b 32161792  
d249 -p /dev/md/rdsk/d0 -o 1527506432 -b 32161792  
d248 -p /dev/md/rdsk/d3 -o 1495344384 -b 32161792  
d247 -p /dev/md/rdsk/d2 -o 1495344384 -b 32161792  
d246 -p /dev/md/rdsk/d1 -o 1495344384 -b 32161792  
d245 -p /dev/md/rdsk/d0 -o 1495344384 -b 32161792  
d244 -p /dev/md/rdsk/d3 -o 1463182336 -b 32161792  
d243 -p /dev/md/rdsk/d2 -o 1463182336 -b 32161792  
d242 -p /dev/md/rdsk/d1 -o 1463182336 -b 32161792  
d241 -p /dev/md/rdsk/d0 -o 1463182336 -b 32161792  
d240 -p /dev/md/rdsk/d3 -o 1431020288 -b 32161792  
d239 -p /dev/md/rdsk/d2 -o 1431020288 -b 32161792  
d238 -p /dev/md/rdsk/d1 -o 1431020288 -b 32161792  
d237 -p /dev/md/rdsk/d0 -o 1431020288 -b 32161792  
d236 -p /dev/md/rdsk/d3 -o 1398858240 -b 32161792  
d235 -p /dev/md/rdsk/d2 -o 1398858240 -b 32161792  
d234 -p /dev/md/rdsk/d1 -o 1398858240 -b 32161792  
d233 -p /dev/md/rdsk/d0 -o 1398858240 -b 32161792  
d232 -p /dev/md/rdsk/d3 -o 1366696192 -b 32161792  
d3191 -p /dev/md/rdsk/d8001 -o 1493352448 -b 120840192  
d3192 -p /dev/md/rdsk/d8002 -o 1582030848 -b 120840192  
d231 -p /dev/md/rdsk/d2 -o 1366696192 -b 32161792  
d230 -p /dev/md/rdsk/d1 -o 1366696192 -b 32161792  
d229 -p /dev/md/rdsk/d0 -o 1366696192 -b 32161792  
d228 -p /dev/md/rdsk/d3 -o 1334534144 -b 32161792  
d227 -p /dev/md/rdsk/d2 -o 1334534144 -b 32161792  
d226 -p /dev/md/rdsk/d1 -o 1334534144 -b 32161792  
d3193 -p /dev/md/rdsk/d8003 -o 1575272448 -b 120840192  
d3194 -p /dev/md/rdsk/d8004 -o 1545576448 -b 120840192  
d3195 -p /dev/md/rdsk/d8004 -o 1424735232 -b 120840192  
d225 -p /dev/md/rdsk/d0 -o 1334534144 -b 32161792  
d224 -p /dev/md/rdsk/d3 -o 1302372096 -b 32161792  
d223 -p /dev/md/rdsk/d2 -o 1302372096 -b 32161792  
d222 -p /dev/md/rdsk/d1 -o 1302372096 -b 32161792  
d221 -p /dev/md/rdsk/d0 -o 1302372096 -b 32161792  
d3196 -p /dev/md/rdsk/d8003 -o 1454431232 -b 120840192  
d3197 -p /dev/md/rdsk/d8002 -o 1461189632 -b 120840192  
d3198 -p /dev/md/rdsk/d8001 -o 1372511232 -b 120840192  
d3199 -p /dev/md/rdsk/d8000 -o 1561737216 -b 120840192  
d2285 -p /dev/md/rdsk/d8001 -o 1371478016 -b 1032192  
d2286 -p /dev/md/rdsk/d8002 -o 1460156416 -b 1032192  
d2287 -p /dev/md/rdsk/d8003 -o 1453398016 -b 1032192  
d220 -p /dev/md/rdsk/d3 -o 1270210048 -b 32161792  
d219 -p /dev/md/rdsk/d2 -o 1270210048 -b 32161792  
d218 -p /dev/md/rdsk/d1 -o 1270210048 -b 32161792  
d217 -p /dev/md/rdsk/d0 -o 1270210048 -b 32161792  
d216 -p /dev/md/rdsk/d3 -o 1238048000 -b 32161792  
d215 -p /dev/md/rdsk/d2 -o 1238048000 -b 32161792  
d214 -p /dev/md/rdsk/d1 -o 1238048000 -b 32161792  
d213 -p /dev/md/rdsk/d0 -o 1238048000 -b 32161792  
d212 -p /dev/md/rdsk/d3 -o 1205885952 -b 32161792  
d211 -p /dev/md/rdsk/d2 -o 1205885952 -b 32161792  
d210 -p /dev/md/rdsk/d1 -o 1205885952 -b 32161792  
d209 -p /dev/md/rdsk/d0 -o 1205885952 -b 32161792  
d208 -p /dev/md/rdsk/d3 -o 1085045504 -b 120840192  
d207 -p /dev/md/rdsk/d2 -o 1085045504 -b 120840192  
d206 -p /dev/md/rdsk/d1 -o 1085045504 -b 120840192  
d205 -p /dev/md/rdsk/d0 -o 1085045504 -b 120840192  
d204 -p /dev/md/rdsk/d3 -o 964205056 -b 120840192  
d2288 -p /dev/md/rdsk/d8004 -o 1423702016 -b 1032192

d2289 -p /dev/md/rdsk/d8004 -o 1421235200 -b 2465792  
d2209 -p /dev/md/rdsk/d8003 -o 1421235200 -b 32161792  
d2210 -p /dev/md/rdsk/d8002 -o 1427993600 -b 32161792  
d2211 -p /dev/md/rdsk/d8001 -o 1339315200 -b 32161792  
d2212 -p /dev/md/rdsk/d8000 -o 1529574400 -b 32161792  
d203 -p /dev/md/rdsk/d2 -o 964205056 -b 120840192  
d202 -p /dev/md/rdsk/d1 -o 964205056 -b 120840192  
d2213 -p /dev/md/rdsk/d8000 -o 1497411584 -b 32161792  
d2214 -p /dev/md/rdsk/d8001 -o 1307152384 -b 32161792  
d2215 -p /dev/md/rdsk/d8002 -o 1395830784 -b 32161792  
d2216 -p /dev/md/rdsk/d8003 -o 1389072384 -b 32161792  
d2217 -p /dev/md/rdsk/d8004 -o 1389072384 -b 32161792  
d2218 -p /dev/md/rdsk/d8004 -o 1356909568 -b 32161792  
d2219 -p /dev/md/rdsk/d8003 -o 1356909568 -b 32161792  
d2220 -p /dev/md/rdsk/d8002 -o 1363667968 -b 32161792  
d2221 -p /dev/md/rdsk/d8001 -o 1274989568 -b 32161792  
d2222 -p /dev/md/rdsk/d8000 -o 1465248768 -b 32161792  
d2223 -p /dev/md/rdsk/d8000 -o 1433085952 -b 32161792  
d2224 -p /dev/md/rdsk/d8001 -o 1242826752 -b 32161792  
d2225 -p /dev/md/rdsk/d8002 -o 1331505152 -b 32161792  
d2226 -p /dev/md/rdsk/d8003 -o 1324746752 -b 32161792  
d2227 -p /dev/md/rdsk/d8004 -o 1324746752 -b 32161792  
d2228 -p /dev/md/rdsk/d8004 -o 1292583936 -b 32161792  
d2229 -p /dev/md/rdsk/d8003 -o 1292583936 -b 32161792  
d2230 -p /dev/md/rdsk/d8002 -o 1299342336 -b 32161792  
d2231 -p /dev/md/rdsk/d8001 -o 1210663936 -b 32161792  
d2232 -p /dev/md/rdsk/d8000 -o 1400923136 -b 32161792  
d2233 -p /dev/md/rdsk/d8000 -o 1368760320 -b 32161792  
d2234 -p /dev/md/rdsk/d8001 -o 1178501120 -b 32161792  
d2235 -p /dev/md/rdsk/d8002 -o 1267179520 -b 32161792  
d2236 -p /dev/md/rdsk/d8003 -o 1260421120 -b 32161792  
d2237 -p /dev/md/rdsk/d8004 -o 1260421120 -b 32161792  
d2238 -p /dev/md/rdsk/d8004 -o 1228258304 -b 32161792  
d2239 -p /dev/md/rdsk/d8003 -o 1228258304 -b 32161792  
d2240 -p /dev/md/rdsk/d8002 -o 1235016704 -b 32161792  
d2241 -p /dev/md/rdsk/d8001 -o 1146338304 -b 32161792  
d2242 -p /dev/md/rdsk/d8000 -o 1336597504 -b 32161792  
d2243 -p /dev/md/rdsk/d8000 -o 1304434688 -b 32161792  
d2244 -p /dev/md/rdsk/d8001 -o 1114175488 -b 32161792  
d2245 -p /dev/md/rdsk/d8002 -o 1202853888 -b 32161792  
d2246 -p /dev/md/rdsk/d8003 -o 1196095488 -b 32161792  
d2247 -p /dev/md/rdsk/d8004 -o 1196095488 -b 32161792  
d2248 -p /dev/md/rdsk/d8004 -o 1163932672 -b 32161792  
d2249 -p /dev/md/rdsk/d8003 -o 1163932672 -b 32161792  
d2250 -p /dev/md/rdsk/d8002 -o 1170691072 -b 32161792  
d2251 -p /dev/md/rdsk/d8001 -o 1082012672 -b 32161792  
d2252 -p /dev/md/rdsk/d8000 -o 1272271872 -b 32161792  
d2253 -p /dev/md/rdsk/d8000 -o 1240109056 -b 32161792  
d2254 -p /dev/md/rdsk/d8001 -o 1049849856 -b 32161792  
d2255 -p /dev/md/rdsk/d8002 -o 1138528256 -b 32161792  
d2256 -p /dev/md/rdsk/d8003 -o 1131769856 -b 32161792  
d2257 -p /dev/md/rdsk/d8004 -o 1131769856 -b 32161792  
d2258 -p /dev/md/rdsk/d8004 -o 1099607040 -b 32161792  
d2259 -p /dev/md/rdsk/d8003 -o 1099607040 -b 32161792  
d2260 -p /dev/md/rdsk/d8002 -o 1106365440 -b 32161792  
d2261 -p /dev/md/rdsk/d8001 -o 1017687040 -b 32161792  
d2262 -p /dev/md/rdsk/d8000 -o 1207946240 -b 32161792  
d2263 -p /dev/md/rdsk/d8000 -o 1175783424 -b 32161792

d2264 -p /dev/md/rdsk/d8001 -o 985524224 -b 32161792  
d2265 -p /dev/md/rdsk/d8002 -o 1074202624 -b 32161792  
d2266 -p /dev/md/rdsk/d8003 -o 1067444224 -b 32161792  
d2267 -p /dev/md/rdsk/d8004 -o 1067444224 -b 32161792  
d2268 -p /dev/md/rdsk/d8004 -o 1035281408 -b 32161792  
d2269 -p /dev/md/rdsk/d8003 -o 1035281408 -b 32161792  
d2270 -p /dev/md/rdsk/d8002 -o 1042039808 -b 32161792  
d2271 -p /dev/md/rdsk/d8001 -o 953361408 -b 32161792  
d2272 -p /dev/md/rdsk/d8000 -o 1143620608 -b 32161792  
d2273 -p /dev/md/rdsk/d8000 -o 1111457792 -b 32161792  
d2274 -p /dev/md/rdsk/d8001 -o 921198592 -b 32161792  
d2275 -p /dev/md/rdsk/d8002 -o 1009876992 -b 32161792  
d2276 -p /dev/md/rdsk/d8003 -o 1003118592 -b 32161792  
d2277 -p /dev/md/rdsk/d8004 -o 1003118592 -b 32161792  
d2278 -p /dev/md/rdsk/d8004 -o 970955776 -b 32161792  
d2279 -p /dev/md/rdsk/d8003 -o 970955776 -b 32161792  
d2280 -p /dev/md/rdsk/d8002 -o 977714176 -b 32161792  
d2281 -p /dev/md/rdsk/d8001 -o 889035776 -b 32161792  
d2282 -p /dev/md/rdsk/d8000 -o 1079294976 -b 32161792  
d2283 -p /dev/md/rdsk/d8000 -o 1047132160 -b 32161792  
d2284 -p /dev/md/rdsk/d8001 -o 856872960 -b 32161792  
d2200 -p /dev/md/rdsk/d8002 -o 856872960 -b 120840192  
d2201 -p /dev/md/rdsk/d8003 -o 850114560 -b 120840192  
d2202 -p /dev/md/rdsk/d8004 -o 850114560 -b 120840192  
d2203 -p /dev/md/rdsk/d8004 -o 729273344 -b 120840192  
d2204 -p /dev/md/rdsk/d8003 -o 729273344 -b 120840192  
d2205 -p /dev/md/rdsk/d8002 -o 736031744 -b 120840192  
d2206 -p /dev/md/rdsk/d8001 -o 736031744 -b 120840192  
d2207 -p /dev/md/rdsk/d8000 -o 926290944 -b 120840192  
d2208 -p /dev/md/rdsk/d8000 -o 805449728 -b 120840192  
d2185 -p /dev/md/rdsk/d8001 -o 728649728 -b 7380992  
d2186 -p /dev/md/rdsk/d8002 -o 728649728 -b 7380992  
d2187 -p /dev/md/rdsk/d8003 -o 721891328 -b 7380992  
d2188 -p /dev/md/rdsk/d8004 -o 721891328 -b 7380992  
d2173 -p /dev/md/rdsk/d8004 -o 707341312 -b 14548992  
d2174 -p /dev/md/rdsk/d8003 -o 707341312 -b 14548992  
d2175 -p /dev/md/rdsk/d8002 -o 714099712 -b 14548992  
d2176 -p /dev/md/rdsk/d8001 -o 714099712 -b 14548992  
d2177 -p /dev/md/rdsk/d8000 -o 790899712 -b 14548992  
d2178 -p /dev/md/rdsk/d8000 -o 776349696 -b 14548992  
d2179 -p /dev/md/rdsk/d8001 -o 699549696 -b 14548992  
d2180 -p /dev/md/rdsk/d8002 -o 699549696 -b 14548992  
d2165 -p /dev/md/rdsk/d8003 -o 692176896 -b 15163392  
d2166 -p /dev/md/rdsk/d8004 -o 692176896 -b 15163392  
d2167 -p /dev/md/rdsk/d8004 -o 677012480 -b 15163392  
d2168 -p /dev/md/rdsk/d8003 -o 677012480 -b 15163392  
d2157 -p /dev/md/rdsk/d8002 -o 677012480 -b 22536192  
d2158 -p /dev/md/rdsk/d8001 -o 677012480 -b 22536192  
d2159 -p /dev/md/rdsk/d8000 -o 753812480 -b 22536192  
d2160 -p /dev/md/rdsk/d8000 -o 731275264 -b 22536192  
d2161 -p /dev/md/rdsk/d8001 -o 654475264 -b 22536192  
d2162 -p /dev/md/rdsk/d8002 -o 654475264 -b 22536192  
d2163 -p /dev/md/rdsk/d8003 -o 654475264 -b 22536192  
d2164 -p /dev/md/rdsk/d8004 -o 654475264 -b 22536192  
d2101 -p /dev/md/rdsk/d8004 -o 623746048 -b 30728192  
d2102 -p /dev/md/rdsk/d8003 -o 623746048 -b 30728192  
d2103 -p /dev/md/rdsk/d8002 -o 623746048 -b 30728192  
d2104 -p /dev/md/rdsk/d8001 -o 623746048 -b 30728192

d2105 -p /dev/md/rdsk/d8000 -o 700546048 -b 30728192  
d2106 -p /dev/md/rdsk/d8000 -o 669816832 -b 30728192  
d2107 -p /dev/md/rdsk/d8001 -o 593016832 -b 30728192  
d2108 -p /dev/md/rdsk/d8002 -o 593016832 -b 30728192  
d2109 -p /dev/md/rdsk/d8003 -o 593016832 -b 30728192  
d2110 -p /dev/md/rdsk/d8004 -o 593016832 -b 30728192  
d2111 -p /dev/md/rdsk/d8004 -o 562287616 -b 30728192  
d2112 -p /dev/md/rdsk/d8003 -o 562287616 -b 30728192  
d2113 -p /dev/md/rdsk/d8002 -o 562287616 -b 30728192  
d2114 -p /dev/md/rdsk/d8001 -o 562287616 -b 30728192  
d2115 -p /dev/md/rdsk/d8000 -o 639087616 -b 30728192  
d2116 -p /dev/md/rdsk/d8000 -o 608358400 -b 30728192  
d2117 -p /dev/md/rdsk/d8001 -o 531558400 -b 30728192  
d2118 -p /dev/md/rdsk/d8002 -o 531558400 -b 30728192  
d2119 -p /dev/md/rdsk/d8003 -o 531558400 -b 30728192  
d2120 -p /dev/md/rdsk/d8004 -o 531558400 -b 30728192  
d2121 -p /dev/md/rdsk/d8004 -o 500829184 -b 30728192  
d2122 -p /dev/md/rdsk/d8003 -o 500829184 -b 30728192  
d2123 -p /dev/md/rdsk/d8002 -o 500829184 -b 30728192  
d2124 -p /dev/md/rdsk/d8001 -o 500829184 -b 30728192  
d2125 -p /dev/md/rdsk/d8000 -o 577629184 -b 30728192  
d2126 -p /dev/md/rdsk/d8000 -o 546899968 -b 30728192  
d2127 -p /dev/md/rdsk/d8001 -o 470099968 -b 30728192  
d2128 -p /dev/md/rdsk/d8002 -o 470099968 -b 30728192  
d2129 -p /dev/md/rdsk/d8003 -o 470099968 -b 30728192  
d2130 -p /dev/md/rdsk/d8004 -o 470099968 -b 30728192  
d2131 -p /dev/md/rdsk/d8004 -o 439370752 -b 30728192  
d2132 -p /dev/md/rdsk/d8003 -o 439370752 -b 30728192  
d2133 -p /dev/md/rdsk/d8002 -o 439370752 -b 30728192  
d201 -p /dev/md/rdsk/d0 -o 964205056 -b 120840192  
d200 -p /dev/md/rdsk/d3 -o 843364608 -b 120840192  
d199 -p /dev/md/rdsk/d2 -o 843364608 -b 120840192  
d198 -p /dev/md/rdsk/d1 -o 843364608 -b 120840192  
d197 -p /dev/md/rdsk/d0 -o 843364608 -b 120840192  
d196 -p /dev/md/rdsk/d3 -o 722524160 -b 120840192  
d2134 -p /dev/md/rdsk/d8001 -o 439370752 -b 30728192  
d2135 -p /dev/md/rdsk/d8000 -o 516170752 -b 30728192  
d2136 -p /dev/md/rdsk/d8000 -o 485441536 -b 30728192  
d2137 -p /dev/md/rdsk/d8001 -o 408641536 -b 30728192  
d2138 -p /dev/md/rdsk/d8002 -o 408641536 -b 30728192  
d2139 -p /dev/md/rdsk/d8003 -o 408641536 -b 30728192  
d195 -p /dev/md/rdsk/d2 -o 722524160 -b 120840192  
d194 -p /dev/md/rdsk/d1 -o 722524160 -b 120840192  
d193 -p /dev/md/rdsk/d0 -o 722524160 -b 120840192  
d192 -p /dev/md/rdsk/d3 -o 601683712 -b 120840192  
d191 -p /dev/md/rdsk/d2 -o 601683712 -b 120840192  
d190 -p /dev/md/rdsk/d1 -o 601683712 -b 120840192  
d189 -p /dev/md/rdsk/d0 -o 601683712 -b 120840192  
d188 -p /dev/md/rdsk/d3 -o 594302464 -b 7380992  
d187 -p /dev/md/rdsk/d2 -o 594302464 -b 7380992  
d186 -p /dev/md/rdsk/d1 -o 594302464 -b 7380992  
d185 -p /dev/md/rdsk/d0 -o 594302464 -b 7380992  
d184 -p /dev/md/rdsk/d3 -o 551286016 -b 43016192  
d183 -p /dev/md/rdsk/d2 -o 551286016 -b 43016192  
d182 -p /dev/md/rdsk/d1 -o 551286016 -b 43016192  
d181 -p /dev/md/rdsk/d0 -o 551286016 -b 43016192  
d180 -p /dev/md/rdsk/d3 -o 536736768 -b 14548992  
d179 -p /dev/md/rdsk/d2 -o 536736768 -b 14548992

```

d178 -p /dev/md/rdsd/d1 -o 536736768 -b 14548992
d177 -p /dev/md/rdsd/d0 -o 536736768 -b 14548992
d176 -p /dev/md/rdsd/d3 -o 522187520 -b 14548992
d175 -p /dev/md/rdsd/d2 -o 522187520 -b 14548992
d174 -p /dev/md/rdsd/d1 -o 522187520 -b 14548992
d173 -p /dev/md/rdsd/d0 -o 522187520 -b 14548992
d172 -p /dev/md/rdsd/d3 -o 490435072 -b 31752192
d171 -p /dev/md/rdsd/d2 -o 490435072 -b 31752192
d170 -p /dev/md/rdsd/d1 -o 490435072 -b 31752192
d169 -p /dev/md/rdsd/d0 -o 490435072 -b 31752192
d168 -p /dev/md/rdsd/d3 -o 475271424 -b 15163392
d167 -p /dev/md/rdsd/d2 -o 475271424 -b 15163392
d166 -p /dev/md/rdsd/d1 -o 475271424 -b 15163392
d165 -p /dev/md/rdsd/d0 -o 475271424 -b 15163392
d164 -p /dev/md/rdsd/d3 -o 452734976 -b 22536192
d163 -p /dev/md/rdsd/d2 -o 452734976 -b 22536192
d162 -p /dev/md/rdsd/d1 -o 452734976 -b 22536192
d161 -p /dev/md/rdsd/d0 -o 452734976 -b 22536192
d160 -p /dev/md/rdsd/d3 -o 430198528 -b 22536192
d159 -p /dev/md/rdsd/d2 -o 430198528 -b 22536192
d158 -p /dev/md/rdsd/d1 -o 430198528 -b 22536192
d157 -p /dev/md/rdsd/d0 -o 430198528 -b 22536192
d156 -p /dev/md/rdsd/d3 -o 399470080 -b 30728192
d155 -p /dev/md/rdsd/d2 -o 399470080 -b 30728192
d154 -p /dev/md/rdsd/d1 -o 399470080 -b 30728192
d153 -p /dev/md/rdsd/d0 -o 399470080 -b 30728192
d152 -p /dev/md/rdsd/d3 -o 368741632 -b 30728192
d151 -p /dev/md/rdsd/d2 -o 368741632 -b 30728192
d150 -p /dev/md/rdsd/d1 -o 368741632 -b 30728192
d149 -p /dev/md/rdsd/d0 -o 368741632 -b 30728192
d148 -p /dev/md/rdsd/d3 -o 338013184 -b 30728192
d147 -p /dev/md/rdsd/d2 -o 338013184 -b 30728192
d146 -p /dev/md/rdsd/d1 -o 338013184 -b 30728192
d2140 -p /dev/md/rdsd/d8004 -o 408641536 -b 30728192
d2141 -p /dev/md/rdsd/d8004 -o 377912320 -b 30728192
d2142 -p /dev/md/rdsd/d8003 -o 377912320 -b 30728192
d2143 -p /dev/md/rdsd/d8002 -o 377912320 -b 30728192
d2144 -p /dev/md/rdsd/d8001 -o 377912320 -b 30728192
d145 -p /dev/md/rdsd/d0 -o 338013184 -b 30728192
d144 -p /dev/md/rdsd/d3 -o 307284736 -b 30728192
d143 -p /dev/md/rdsd/d2 -o 307284736 -b 30728192
d2145 -p /dev/md/rdsd/d8000 -o 454712320 -b 30728192
d2146 -p /dev/md/rdsd/d8000 -o 423983104 -b 30728192
d2147 -p /dev/md/rdsd/d8001 -o 347183104 -b 30728192
d2148 -p /dev/md/rdsd/d8002 -o 347183104 -b 30728192
d2149 -p /dev/md/rdsd/d8003 -o 347183104 -b 30728192
d142 -p /dev/md/rdsd/d1 -o 307284736 -b 30728192
d141 -p /dev/md/rdsd/d0 -o 307284736 -b 30728192
d140 -p /dev/md/rdsd/d3 -o 276556288 -b 30728192
d139 -p /dev/md/rdsd/d2 -o 276556288 -b 30728192
d138 -p /dev/md/rdsd/d1 -o 276556288 -b 30728192
d137 -p /dev/md/rdsd/d0 -o 276556288 -b 30728192
d136 -p /dev/md/rdsd/d3 -o 245827840 -b 30728192
d135 -p /dev/md/rdsd/d2 -o 245827840 -b 30728192
d134 -p /dev/md/rdsd/d1 -o 245827840 -b 30728192
d133 -p /dev/md/rdsd/d0 -o 245827840 -b 30728192
d132 -p /dev/md/rdsd/d3 -o 215099392 -b 30728192
d2150 -p /dev/md/rdsd/d8004 -o 347183104 -b 30728192

```

```

d2151 -p /dev/md/rdsd/d8004 -o 316453888 -b 30728192
d2152 -p /dev/md/rdsd/d8003 -o 316453888 -b 30728192
d131 -p /dev/md/rdsd/d2 -o 215099392 -b 30728192
d130 -p /dev/md/rdsd/d1 -o 215099392 -b 30728192
d129 -p /dev/md/rdsd/d0 -o 215099392 -b 30728192
d128 -p /dev/md/rdsd/d3 -o 184370944 -b 30728192
d127 -p /dev/md/rdsd/d2 -o 184370944 -b 30728192
d2153 -p /dev/md/rdsd/d8002 -o 316453888 -b 30728192
d2154 -p /dev/md/rdsd/d8001 -o 316453888 -b 30728192
d2155 -p /dev/md/rdsd/d8000 -o 393253888 -b 30728192
d2156 -p /dev/md/rdsd/d8000 -o 362524672 -b 30728192
d2169 -p /dev/md/rdsd/d8001 -o 284700672 -b 31752192
d2170 -p /dev/md/rdsd/d8002 -o 284700672 -b 31752192
d2171 -p /dev/md/rdsd/d8003 -o 284700672 -b 31752192
d2172 -p /dev/md/rdsd/d8004 -o 284700672 -b 31752192
d2181 -p /dev/md/rdsd/d8004 -o 241683456 -b 43016192
d2182 -p /dev/md/rdsd/d8003 -o 241683456 -b 43016192
d2183 -p /dev/md/rdsd/d8002 -o 241683456 -b 43016192
d2184 -p /dev/md/rdsd/d8001 -o 241683456 -b 43016192
d2189 -p /dev/md/rdsd/d8000 -o 241683456 -b 120840192
d2190 -p /dev/md/rdsd/d8000 -o 120842240 -b 120840192
d2191 -p /dev/md/rdsd/d8001 -o 120842240 -b 120840192
d2192 -p /dev/md/rdsd/d8002 -o 120842240 -b 120840192
d2193 -p /dev/md/rdsd/d8003 -o 120842240 -b 120840192
d2194 -p /dev/md/rdsd/d8004 -o 120842240 -b 120840192
d2195 -p /dev/md/rdsd/d8004 -o 1024 -b 120840192
d2196 -p /dev/md/rdsd/d8003 -o 1024 -b 120840192
d2197 -p /dev/md/rdsd/d8002 -o 1024 -b 120840192
d2198 -p /dev/md/rdsd/d8001 -o 1024 -b 120840192
d2199 -p /dev/md/rdsd/d8000 -o 1024 -b 120840192
d126 -p /dev/md/rdsd/d1 -o 184370944 -b 30728192
d125 -p /dev/md/rdsd/d0 -o 184370944 -b 30728192
d124 -p /dev/md/rdsd/d3 -o 153642496 -b 30728192
d123 -p /dev/md/rdsd/d2 -o 153642496 -b 30728192
d122 -p /dev/md/rdsd/d1 -o 153642496 -b 30728192
d121 -p /dev/md/rdsd/d0 -o 153642496 -b 30728192
d120 -p /dev/md/rdsd/d3 -o 122914048 -b 30728192
d119 -p /dev/md/rdsd/d2 -o 122914048 -b 30728192
d118 -p /dev/md/rdsd/d1 -o 122914048 -b 30728192
d117 -p /dev/md/rdsd/d0 -o 122914048 -b 30728192
d116 -p /dev/md/rdsd/d3 -o 92185600 -b 30728192
d115 -p /dev/md/rdsd/d2 -o 92185600 -b 30728192
d114 -p /dev/md/rdsd/d1 -o 92185600 -b 30728192
d113 -p /dev/md/rdsd/d0 -o 92185600 -b 30728192
d112 -p /dev/md/rdsd/d3 -o 61457152 -b 30728192
d111 -p /dev/md/rdsd/d2 -o 61457152 -b 30728192
d110 -p /dev/md/rdsd/d1 -o 61457152 -b 30728192
d109 -p /dev/md/rdsd/d0 -o 61457152 -b 30728192
d108 -p /dev/md/rdsd/d3 -o 30728704 -b 30728192
d107 -p /dev/md/rdsd/d2 -o 30728704 -b 30728192
d106 -p /dev/md/rdsd/d1 -o 30728704 -b 30728192
d105 -p /dev/md/rdsd/d0 -o 30728704 -b 30728192
d104 -p /dev/md/rdsd/d3 -o 256 -b 30728192
d103 -p /dev/md/rdsd/d2 -o 256 -b 30728192
d102 -p /dev/md/rdsd/d1 -o 256 -b 30728192
d101 -p /dev/md/rdsd/d0 -o 256 -b 30728192

```

**Measured and Priced COMSTAR DATA head (X4270 M2)**

**uname**

SunOS x4270-047 5.11 11.0 i86pc i386 i86pc

**prtdiag**

System Configuration: Oracle Corporation SUN FIRE X4270 M2 SERVER  
 BIOS Configuration: American Megatrends Inc. 08100102 10/11/2011  
 BMC Configuration: IPMI 1.5 (KCS: Keyboard Controller Style)

==== Processor Sockets =====

Version	Location	Tag
Intel(R) Xeon(R) CPU	X5675	@ 3.07GHz CPU 0

==== Memory Device Sockets =====

Type	Status	Set	Device	Locator	Bank	Locator
Unknown	empty	0	D2		/SYS/MB/P0	
Unknown	empty	0	D1		/SYS/MB/P0	
Unknown	empty	0	D0		/SYS/MB/P0	
Unknown	empty	0	D5		/SYS/MB/P0	
Unknown	empty	0	D4		/SYS/MB/P0	
Unknown	empty	0	D3		/SYS/MB/P0	
Unknown	in use	0	D8		/SYS/MB/P0	
Unknown	empty	0	D7		/SYS/MB/P0	
Unknown	empty	0	D6		/SYS/MB/P0	
Unknown	empty	0	D2		/SYS/MB/P1	
Unknown	empty	0	D1		/SYS/MB/P1	
Unknown	empty	0	D0		/SYS/MB/P1	
Unknown	empty	0	D5		/SYS/MB/P1	
Unknown	empty	0	D4		/SYS/MB/P1	
Unknown	empty	0	D3		/SYS/MB/P1	
Unknown	empty	0	D8		/SYS/MB/P1	
Unknown	empty	0	D7		/SYS/MB/P1	
Unknown	empty	0	D6		/SYS/MB/P1	
FLASH	in use	0				

==== On-Board Devices =====

Intel 82576 Ethernet Controller.  
 Intel 82576 Ethernet Controller.  
 Intel 82576 Ethernet Controller.  
 Intel 82576 Ethernet Controller.  
 ICH10R SATA Controller.

==== Upgradeable Slots =====

ID	Status	Type	Description
1	in use	PCI Express	PCIE0
2	in use	PCI Express	PCIE1

3 in use PCI Express PCIE2  
 4 in use PCI Express PCIE3  
 5 in use PCI Express PCIE4  
 6 in use PCI Express PCIE5

**prtconf**

System Configuration: Oracle Corporation i86pc  
 Memory size: 8184 Megabytes  
 System Peripherals (Software Nodes):

i86pc  
 scsi\_vhci, instance #0  
 pci, instance #0  
 pci108e,484c (driver not attached)  
 pci8086,3408, instance #0  
 pci108e,484c, instance #0  
 pci108e,484c, instance #1  
 pci8086,3409, instance #1  
 pci108e,484c, instance #2  
 pci108e,484c, instance #3  
 pci8086,340a, instance #2  
 pci1077,171, instance #0  
 pci1077,171, instance #1  
 pci8086,340c, instance #3  
 pci1000,9263, instance #0  
 sd, instance #0  
 sd, instance #1  
 sd, instance #2  
 sd, instance #3  
 sd, instance #4  
 sd, instance #5  
 sd, instance #6  
 sd, instance #7  
 sd, instance #8  
 sd, instance #9  
 sd, instance #10  
 pci8086,340e, instance #4  
 pci111d,806e, instance #6  
 pci111d,806e, instance #7  
 pci1000,30b0, instance #0  
 iport, instance #4  
 iport, instance #9  
 smp, instance #13  
 disk, instance #220  
 disk, instance #224  
 disk, instance #226  
 enclosure, instance #13  
 disk, instance #217  
 disk, instance #218  
 disk, instance #219  
 disk, instance #225  
 disk, instance #221  
 disk, instance #227  
 disk, instance #228  
 disk, instance #229  
 disk, instance #230

disk, instance #231  
 disk, instance #232  
 disk, instance #234  
 disk, instance #235  
 disk, instance #222  
 disk, instance #236  
 disk, instance #237  
 disk, instance #238  
 iport, instance #8  
 smp, instance #0  
 disk, instance #18  
 disk, instance #19  
 disk, instance #22  
 enclosure, instance #0  
 disk, instance #12  
 disk, instance #14  
 disk, instance #16  
 disk, instance #20  
 disk, instance #21  
 disk, instance #23  
 disk, instance #24  
 disk, instance #25  
 disk, instance #26  
 disk, instance #27  
 disk, instance #28  
 disk, instance #29  
 disk, instance #30  
 disk, instance #11  
 disk, instance #13  
 disk, instance #15  
 disk, instance #17  
 pci111d,806e, instance #8  
 pci1000,30b0, instance #1  
 iport, instance #5  
 iport, instance #12  
 smp, instance #4  
 disk, instance #122  
 disk, instance #91  
 disk, instance #124  
 disk, instance #92  
 disk, instance #126  
 enclosure, instance #4  
 disk, instance #94  
 disk, instance #96  
 disk, instance #102  
 disk, instance #106  
 disk, instance #98  
 disk, instance #100  
 disk, instance #104  
 disk, instance #108  
 disk, instance #110  
 disk, instance #112  
 disk, instance #114  
 disk, instance #116  
 disk, instance #118  
 disk, instance #120  
 disk, instance #128  
 iport, instance #10

smp, instance #11  
 enclosure, instance #11  
 disk, instance #244  
 disk, instance #214  
 disk, instance #248  
 disk, instance #240  
 disk, instance #242  
 disk, instance #216  
 disk, instance #215  
 disk, instance #246  
 disk, instance #250  
 disk, instance #252  
 disk, instance #254  
 disk, instance #256  
 disk, instance #258  
 disk, instance #260  
 disk, instance #262  
 disk, instance #264  
 disk, instance #266  
 disk, instance #268  
 disk, instance #270  
 disk, instance #272  
 pci8086,3410, instance #5  
 pci111d,806e, instance #9  
 pci111d,806e, instance #10  
 pci1000,30b0, instance #2  
 iport, instance #6  
 iport, instance #13  
 smp, instance #12  
 enclosure, instance #12  
 disk, instance #241  
 disk, instance #243  
 disk, instance #245  
 disk, instance #223  
 disk, instance #233  
 disk, instance #239  
 disk, instance #247  
 disk, instance #249  
 disk, instance #251  
 disk, instance #253  
 disk, instance #255  
 disk, instance #257  
 disk, instance #259  
 disk, instance #261  
 disk, instance #263  
 disk, instance #265  
 disk, instance #267  
 disk, instance #269  
 disk, instance #271  
 disk, instance #273  
 iport, instance #11  
 smp, instance #3  
 enclosure, instance #3  
 disk, instance #78  
 disk, instance #79  
 disk, instance #82  
 disk, instance #72  
 disk, instance #74

disk, instance #76	disk, instance #285	used-resources (driver not attached)
disk, instance #80	disk, instance #286	iscsi, instance #0
disk, instance #81	disk, instance #287	fcioe, instance #0
disk, instance #83	disk, instance #288	options, instance #0
disk, instance #84	disk, instance #289	pseudo, instance #0
disk, instance #85	disk, instance #290	xsvc, instance #0
disk, instance #86	pci8086,342d (driver not attached)	vga_arbiter, instance #0
disk, instance #87	pci8086,342e (driver not attached)	intel-iommu, instance #0
disk, instance #88	pci8086,3422 (driver not attached)	
disk, instance #89	pci8086,3423, instance #0	<b>format</b>
disk, instance #90	pci8086,3438 (driver not attached)	AVAILABLE DISK SELECTIONS:
disk, instance #71	pci108e,484c (driver not attached)	0. c5t5080020000557D00d0 <ATA-MARVELLSD88SA02-D129 cyl
disk, instance #73	pci108e,484c (driver not attached)	23434 alt 2 hd 16 sec 128>
disk, instance #75	pci108e,484c (driver not attached)	/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@4/pci1000,
disk, instance #77	pci108e,484c (driver not attached)	30b0@0/iport@f/disk@w5080020000557d00,0
pci111d,806e, instance #11	pci108e,484c (driver not attached)	/dev/chassis/SUN-Storage-
pci1000,30b0, instance #3	pci108e,484c (driver not attached)	F5100.5080020000557d00/EXP0/FMod00/disk
iport, instance #7	pci108e,484c (driver not attached)	1. c5t5080020000557D01d0 <ATA-MARVELLSD88SA02-D129 cyl
iport, instance #15	pci108e,484c, instance #0	23434 alt 2 hd 16 sec 128>
smp, instance #7	pci108e,484c, instance #1	/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@4/pci1000,
disk, instance #137	pci108e,484c, instance #2	30b0@0/iport@f/disk@w5080020000557d01,0
disk, instance #139	device, instance #0	/dev/chassis/SUN-Storage-
disk, instance #142	keyboard, instance #2	F5100.5080020000557d00/EXP0/FMod01/disk
enclosure, instance #6	mouse, instance #3	2. c5t5080020000557D02d0 <ATA-MARVELLSD88SA02-D129 cyl
disk, instance #133	pci108e,484c, instance #0	23434 alt 2 hd 16 sec 128>
disk, instance #135	hub, instance #0	/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@4/pci1000,
disk, instance #147	communications, instance #0	30b0@0/iport@f/disk@w5080020000557d02,0
disk, instance #140	pci108e,484c, instance #3	/dev/chassis/SUN-Storage-
disk, instance #141	pci108e,484c, instance #4	F5100.5080020000557d00/EXP0/FMod02/disk
disk, instance #143	pci108e,484c, instance #5	3. c5t5080020000557D03d0 <ATA-MARVELLSD88SA02-D129 cyl
disk, instance #144	pci108e,484c, instance #1	23434 alt 2 hd 16 sec 128>
disk, instance #145	hub, instance #1	/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@4/pci1000,
disk, instance #146	pci8086,244e, instance #0	30b0@0/iport@f/disk@w5080020000557d03,0
disk, instance #148	display, instance #0	/dev/chassis/SUN-Storage-
disk, instance #149	isa, instance #0	F5100.5080020000557d00/EXP0/FMod03/disk
disk, instance #150	motherboard (driver not attached)	4. c5t5080020000557D04d0 <ATA-MARVELLSD88SA02-D129 cyl
disk, instance #151	asy, instance #0	23434 alt 2 hd 16 sec 128>
disk, instance #132	motherboard (driver not attached)	/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@4/pci1000,
disk, instance #134	pit_beep, instance #0	30b0@0/iport@f/disk@w5080020000557d04,0
disk, instance #136	pci108e,484c, instance #0	/dev/chassis/SUN-Storage-
disk, instance #138	pci108e,484c (driver not attached)	F5100.5080020000557d00/EXP0/FMod04/disk
iport, instance #14	ioapics (driver not attached)	5. c5t5080020000557D05d0 <ATA-MARVELLSD88SA02-D129 cyl
smp, instance #10	ioapic, instance #0 (driver not attached)	23434 alt 2 hd 16 sec 128>
enclosure, instance #10	fw, instance #0	/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@4/pci1000,
disk, instance #276	cpu, instance #0	30b0@0/iport@f/disk@w5080020000557d05,0
disk, instance #211	cpu, instance #1	/dev/chassis/SUN-Storage-
disk, instance #278	cpu, instance #2	F5100.5080020000557d00/EXP0/FMod05/disk
disk, instance #274	cpu, instance #3	6. c5t5080020000557D06d0 <ATA-MARVELLSD88SA02-D129 cyl
disk, instance #275	cpu, instance #4	23434 alt 2 hd 16 sec 128>
disk, instance #284	cpu, instance #5	/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@4/pci1000,
disk, instance #212	cpu, instance #6	30b0@0/iport@f/disk@w5080020000557d06,0
disk, instance #277	cpu, instance #7	/dev/chassis/SUN-Storage-
disk, instance #213	cpu, instance #8	F5100.5080020000557d00/EXP0/FMod06/disk
disk, instance #279	cpu, instance #9	7. c5t5080020000557D07d0 <ATA-MARVELLSD88SA02-D129 cyl
disk, instance #280	cpu, instance #10	23434 alt 2 hd 16 sec 128>
disk, instance #281	cpu, instance #11	/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@4/pci1000,
disk, instance #282	sb, instance #1	
disk, instance #283		













```

30b0@0/iptort@f0/disk@w508002000050f08c,0 /pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@6,0 *
/dev/chassis/SUN-Storage- 167. c3t7d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> *
F5100.508002000050f080/EXP2/FMod12/disk /pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@7,0 *
153. c1t508002000050f08Dd0 <ATA-MARVELLS88SA02-D129 cyl 168. c3t8d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> the
23434 alt 2 hd 16 sec 128> /pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@8,0 *
/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, 169. c3t9d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> boot
30b0@0/iptort@f0/disk@w508002000050f08d,0 /pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@9,0 *
/dev/chassis/SUN-Storage- 170. c3t10d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> *
F5100.508002000050f080/EXP2/FMod13/disk /pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@a,0 dependent.
154. c1t508002000050f08Ed0 <ATA-MARVELLS88SA02-D129 cyl Specify disk (enter its number): *
23434 alt 2 hd 16 sec 128> /pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, *
/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, *
30b0@0/iptort@f0/disk@w508002000050f08e,0 Format for sample FMod *
/dev/chassis/SUN-Storage- partition> p *
F5100.508002000050f080/EXP2/FMod14/disk Current partition table (original): *
155. c1t508002000050f08Fd0 <ATA-MARVELLS88SA02-D129 cyl Total disk cylinders available: 23434 + 2 (reserved cylinders) *
23434 alt 2 hd 16 sec 128> /pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, *
/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, Part Tag Flag Cylinders Size Blocks *
30b0@0/iptort@f0/disk@w508002000050f08f,0 /dev/chassis/SUN-Storage- 0 unassigned wu 0 0 (0/0/0) 0 *
F5100.508002000050f080/EXP2/FMod15/disk 1 swap wu 1 - 49 49.00MB (49/0/0) 100352 *
156. c1t508002000050f090d0 <ATA-MARVELLS88SA02-D129 cyl 2 backup wu 0 - 23433 22.88GB (23434/0/0) 47992832 *
23434 alt 2 hd 16 sec 128> /pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, 3 unassigned wu 0 0 (0/0/0) 0 *
/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, 4 unassigned wm 50 - 5886 5.70GB (5837/0/0) 11954176 *
30b0@0/iptort@f0/disk@w508002000050f090,0 /dev/chassis/SUN-Storage- 5 unassigned wm 5887 - 11723 5.70GB (5837/0/0) 11954176 *
/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, 6 usr wm 11724 - 17560 5.70GB (5837/0/0) 11954176 *
F5100.508002000050f080/EXP2/FMod16/disk 7 unassigned wm 17561 - 23397 5.70GB (5837/0/0) 11954176 *
157. c1t508002000050f091d0 <ATA-MARVELLS88SA02-D129 cyl 8 boot wu 0 - 0 1.00MB (1/0/0) 2048 *
23434 alt 2 hd 16 sec 128> /pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, 9 unassigned wu 0 0 (0/0/0) 0 *
/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, partition> *
30b0@0/iptort@f0/disk@w508002000050f091,0 /dev/chassis/SUN-Storage- *
F5100.508002000050f080/EXP2/FMod17/disk /etc/system *
158. c1t508002000050f092d0 <ATA-MARVELLS88SA02-D129 cyl *
23434 alt 2 hd 16 sec 128> /pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, *
/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, *ident "%Z%%M% %I% %E% SMI" /* SVR4 1.5 */ *
30b0@0/iptort@f0/disk@w508002000050f092,0 /dev/chassis/SUN-Storage- *
F5100.508002000050f080/EXP2/FMod18/disk * SYSTEM SPECIFICATION FILE *
159. c1t508002000050f093d0 <ATA-MARVELLS88SA02-D129 cyl *
23434 alt 2 hd 16 sec 128> /pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, * moddir: *
/pci@0,0/pci8086,340e@7/pci111d,806e@0/pci111d,806e@2/pci1000, *
30b0@0/iptort@f0/disk@w508002000050f093,0 /dev/chassis/SUN-Storage- *
F5100.508002000050f080/EXP2/FMod19/disk *
160. c3t0d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> boot *
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@0,0 *
161. c3t1d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> *
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@1,0 *
162. c3t2d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> *
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@2,0 *
163. c3t3d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> *
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@3,0 *
164. c3t4d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> *
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@4,0 *
165. c3t5d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> *
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@5,0 *
166. c3t6d0 <LSI-MR9261-8i-2.90 cyl 60754 alt 2 hd 255 sec 252> *

```

```

*
* rootdev: Set the root device. This should be a fully
* expanded physical pathname. The default is
the
* physical pathname of the device where the
boot
* program resides. The physical pathname is
* highly platform and configuration
dependent.
*
* Example:
* rootfs:ufs
* rootdev:/sbus@1,f800000/esp@0,800000/sd@3,0:a
*
* (Swap device configuration should be specified in /etc/vfstab.)
*
* exclude:
*
* Modules appearing in the moddir path which are NOT to be
loaded,
* even if referenced. Note that 'exclude' accepts either a module
name,
* or a filename which includes the directory.
*
* Examples:
* exclude: win
* exclude: sys/shmsys
*
* forceload:
*
* Cause these modules to be loaded at boot time, (just before
mounting
* the root filesystem) rather than at first reference. Note that
* forceload expects a filename which includes the directory. Also
* note that loading a module does not necessarily imply that it will
* be installed.
*
* Example:
* forceload: drv/foo
*
* set:
*
* Set an integer variable in the kernel or a module to a new value.
* This facility should be used with caution. See system(4).
*
* Examples:
*
* To set variables in 'unix':
*
* set nautopt=32
* set maxusers=40
*

```

	GUID	DATA SIZE	SOURCE	
* To set a variable named 'debug' in the module named 'test_module'				/dev/md/rdsk/d128
* set test_module:debug = 0x13	600144f07d498e0000000000d47d101	15732768768		600144f07d498e0000000000d47d129 15732768768
				/dev/md/rdsk/d129
	600144f07d498e0000000000d47d101	15732768768		600144f07d498e0000000000d47d130 15732768768
/etc/release				/dev/md/rdsk/d130
Oracle Solaris 11 11/11 X86	600144f07d498e0000000000d47d102	15732768768		600144f07d498e0000000000d47d131 15732768768
Copyright (c) 1983, 2011, Oracle and/or its affiliates. All rights reserved.				/dev/md/rdsk/d131
Assembled 18 October 2011	600144f07d498e0000000000d47d103	15732768768		600144f07d498e0000000000d47d132 15732768768
				/dev/md/rdsk/d132
	600144f07d498e0000000000d47d104	15732768768		600144f07d498e0000000000d47d133 15732768768
				/dev/md/rdsk/d133
<b>mpt.conf</b>	600144f07d498e0000000000d47d105	15732768768		600144f07d498e0000000000d47d134 15732768768
#				/dev/md/rdsk/d134
# Copyright (c) 2011, Oracle and/or its affiliates. All rights reserved.	600144f07d498e0000000000d47d106	15732768768		600144f07d498e0000000000d47d135 15732768768
#				/dev/md/rdsk/d135
#	600144f07d498e0000000000d47d107	15732768768		600144f07d498e0000000000d47d136 15732768768
#				/dev/md/rdsk/d136
# The mpt driver, as a pHCI driver, must specify the vHCI class it	600144f07d498e0000000000d47d108	15732768768		600144f07d498e0000000000d47d137 15732768768
# belongs to(scsi_vhci).				/dev/md/rdsk/d137
#	600144f07d498e0000000000d47d109	15732768768		600144f07d498e0000000000d47d138 15732768768
ddi-vhci-class="scsi_vhci";				/dev/md/rdsk/d138
#	600144f07d498e0000000000d47d110	15732768768		600144f07d498e0000000000d47d139 15732768768
# I/O multipathing feature (MPxIO) can be enabled or disabled using				/dev/md/rdsk/d139
# mpzio-disable property. Setting mpzio-disable="no" will activate	600144f07d498e0000000000d47d111	15732768768		600144f07d498e0000000000d47d140 15732768768
# I/O multipathing; setting mpzio-disable="yes" disables the feature.				/dev/md/rdsk/d140
#	600144f07d498e0000000000d47d112	15732768768		600144f07d498e0000000000d47d141 15732768768
#				/dev/md/rdsk/d141
# Global mpzio-disable property:	600144f07d498e0000000000d47d113	15732768768		600144f07d498e0000000000d47d142 15732768768
#				/dev/md/rdsk/d142
# To globally enable MPxIO on all mpt controllers set:	600144f07d498e0000000000d47d114	15732768768		600144f07d498e0000000000d47d143 15732768768
# mpzio-disable="no";				/dev/md/rdsk/d143
#	600144f07d498e0000000000d47d115	15732768768		600144f07d498e0000000000d47d144 15732768768
# To globally disable MPxIO on all mpt controllers set:				/dev/md/rdsk/d144
# mpzio-disable="yes";	600144f07d498e0000000000d47d116	15732768768		600144f07d498e0000000000d47d145 15732768768
#				/dev/md/rdsk/d145
# You can also enable or disable MPxIO on a per HBA basis.	600144f07d498e0000000000d47d117	15732768768		600144f07d498e0000000000d47d146 15732768768
# Per HBA settings override the global setting for the specified HBAs.				/dev/md/rdsk/d146
# To disable MPxIO on a controller whose parent is /pci@7c0/pci@0/pci@9	600144f07d498e0000000000d47d118	15732768768		600144f07d498e0000000000d47d147 15732768768
# and the unit-address is "0" set:				/dev/md/rdsk/d147
# name="mpt" parent="/pci@7c0/pci@0/pci@9" unit-address="0" mpzio-	600144f07d498e0000000000d47d119	15732768768		600144f07d498e0000000000d47d148 15732768768
disable="yes";				/dev/md/rdsk/d148
#	600144f07d498e0000000000d47d120	15732768768		600144f07d498e0000000000d47d149 15732768768
mpzio-disable="yes";				/dev/md/rdsk/d149
#	600144f07d498e0000000000d47d121	15732768768		600144f07d498e0000000000d47d150 15732768768
# SATA mpzio supported				/dev/md/rdsk/d150
#	600144f07d498e0000000000d47d122	15732768768		600144f07d498e0000000000d47d151 15732768768
# To disable SATA mpzio, set				/dev/md/rdsk/d151
# disable-sata-mpzio="yes";	600144f07d498e0000000000d47d123	15732768768		600144f07d498e0000000000d47d152 15732768768
# When mpzio-disable="yes" is set, the disable-sata-mpzio property				/dev/md/rdsk/d152
# takes no effect	600144f07d498e0000000000d47d124	15732768768		600144f07d498e0000000000d47d153 15732768768
#				/dev/md/rdsk/d153
disable-sata-mpzio="no";	600144f07d498e0000000000d47d125	15732768768		600144f07d498e0000000000d47d154 15732768768
				/dev/md/rdsk/d154
	600144f07d498e0000000000d47d126	15732768768		600144f07d498e0000000000d47d155 15732768768
<b>sbadm</b>				/dev/md/rdsk/d155
	600144f07d498e0000000000d47d127	15732768768		600144f07d498e0000000000d47d156 15732768768
				/dev/md/rdsk/d156
Found 189 LU(s)	600144f07d498e0000000000d47d128	15732768768		600144f07d498e0000000000d47d157 11538464768





/dev/md/rdisk/d244	/dev/md/rdisk/d273	/dev/rdisk/c8t508002000050F046d0s4
600144f07d498e0000000000d47d245 16466771968	600144f07d498e0000000000d47d274 16466771968	/dev/rdisk/c5t5080020000557D07d0s4
/dev/md/rdisk/d245	/dev/md/rdisk/d274	/dev/rdisk/c4t5080020000557D87d0s4 /dev/rdisk/c14t508002000050F007d0s4
600144f07d498e0000000000d47d246 16466771968	600144f07d498e0000000000d47d275 16466771968	/dev/rdisk/c8t508002000050F047d0s4
/dev/md/rdisk/d246	/dev/md/rdisk/d275	/dev/rdisk/c5t5080020000557D08d0s4
600144f07d498e0000000000d47d247 16466771968	600144f07d498e0000000000d47d276 16466771968	/dev/rdisk/c4t5080020000557D88d0s4 /dev/rdisk/c14t508002000050F008d0s4
/dev/md/rdisk/d247	/dev/md/rdisk/d276	/dev/rdisk/c8t508002000050F048d0s4
600144f07d498e0000000000d47d248 16466771968	600144f07d498e0000000000d47d277 16466771968	/dev/rdisk/c5t5080020000557D09d0s4
/dev/md/rdisk/d248	/dev/md/rdisk/d277	/dev/rdisk/c4t5080020000557D89d0s4 /dev/rdisk/c14t508002000050F009d0s4
600144f07d498e0000000000d47d249 16466771968	600144f07d498e0000000000d47d278 16466771968	/dev/rdisk/c8t508002000050F049d0s4
/dev/md/rdisk/d249	/dev/md/rdisk/d278	/dev/rdisk/c5t5080020000557D0Ad0s4
600144f07d498e0000000000d47d250 16466771968	600144f07d498e0000000000d47d279 16466771968	/dev/rdisk/c4t5080020000557D8Ad0s4
/dev/md/rdisk/d250	/dev/md/rdisk/d279	/dev/rdisk/c14t508002000050F00Ad0s4
600144f07d498e0000000000d47d251 16466771968	600144f07d498e0000000000d47d280 16466771968	/dev/rdisk/c8t508002000050F04Ad0s4 /dev/rdisk/c5t5080020000557D0Bd0s4
/dev/md/rdisk/d251	/dev/md/rdisk/d280	/dev/rdisk/c4t5080020000557D8Bd0s4
600144f07d498e0000000000d47d252 16466771968	600144f07d498e0000000000d47d281 16466771968	/dev/rdisk/c14t508002000050F00Bd0s4
/dev/md/rdisk/d252	/dev/md/rdisk/d281	/dev/rdisk/c8t508002000050F04Bd0s4
600144f07d498e0000000000d47d253 16466771968	600144f07d498e0000000000d47d282 16466771968	/dev/rdisk/c5t5080020000557D0Cd0s4 /dev/rdisk/c4t5080020000557D8Cd0s4
/dev/md/rdisk/d253	/dev/md/rdisk/d282	/dev/rdisk/c14t508002000050F00Cd0s4
600144f07d498e0000000000d47d254 16466771968	600144f07d498e0000000000d47d283 16466771968	/dev/rdisk/c8t508002000050F04Cd0s4 /dev/rdisk/c5t5080020000557D0Dd0s4
/dev/md/rdisk/d254	/dev/md/rdisk/d283	/dev/rdisk/c4t5080020000557D8Dd0s4
600144f07d498e0000000000d47d255 16466771968	600144f07d498e0000000000d47d284 16466771968	/dev/rdisk/c14t508002000050F00Dd0s4
/dev/md/rdisk/d255	/dev/md/rdisk/d284	/dev/rdisk/c8t508002000050F04Dd0s4
600144f07d498e0000000000d47d256 16466771968	600144f07d498e0000000000d47d285 528416768	/dev/rdisk/c5t5080020000557D0Ed0s4 /dev/rdisk/c4t5080020000557D8Ed0s4
/dev/md/rdisk/d256	/dev/md/rdisk/d285	/dev/rdisk/c14t508002000050F00Ed0s4 /dev/rdisk/c8t508002000050F04Ed0s4
600144f07d498e0000000000d47d257 16466771968	600144f07d498e0000000000d47d286 528416768	/dev/rdisk/c5t5080020000557D0Fd0s4
/dev/md/rdisk/d257	/dev/md/rdisk/d286	/dev/rdisk/c4t5080020000557D8Fd0s4
600144f07d498e0000000000d47d258 16466771968	600144f07d498e0000000000d47d287 528416768	/dev/rdisk/c14t508002000050F00Fd0s4 /dev/rdisk/c8t508002000050F04Fd0s4
/dev/md/rdisk/d258	/dev/md/rdisk/d287	/dev/rdisk/c5t5080020000557D10d0s4
600144f07d498e0000000000d47d259 16466771968	600144f07d498e0000000000d47d288 528416768	/dev/rdisk/c4t5080020000557D90d0s4 /dev/rdisk/c14t508002000050F010d0s4
/dev/md/rdisk/d259	/dev/md/rdisk/d288	/dev/rdisk/c8t508002000050F050d0s4
600144f07d498e0000000000d47d260 16466771968	600144f07d498e0000000000d47d289 55508992	/dev/rdisk/c5t5080020000557D11d0s4
/dev/md/rdisk/d260	/dev/md/rdisk/d289	/dev/rdisk/c4t5080020000557D91d0s4 /dev/rdisk/c14t508002000050F011d0s4
600144f07d498e0000000000d47d261 16466771968		/dev/rdisk/c8t508002000050F051d0s4
/dev/md/rdisk/d261	<b>metastat</b>	/dev/rdisk/c5t5080020000557D12d0s4
600144f07d498e0000000000d47d262 16466771968	d289 -p /dev/md/rdisk/d0 -o 1817997312 -b 108544	/dev/rdisk/c4t5080020000557D92d0s4 /dev/rdisk/c14t508002000050F012d0s4
/dev/md/rdisk/d262	d0 1 160 /dev/rdisk/c5t5080020000557D00d0s4	/dev/rdisk/c8t508002000050F052d0s4
600144f07d498e0000000000d47d263 16466771968	/dev/rdisk/c4t5080020000557D80d0s4 /dev/rdisk/c14t508002000050F000d0s4	/dev/rdisk/c5t5080020000557D13d0s4
/dev/md/rdisk/d263	/dev/rdisk/c8t508002000050F040d0s4	/dev/rdisk/c4t5080020000557D93d0s4 /dev/rdisk/c14t508002000050F013d0s4
600144f07d498e0000000000d47d264 16466771968	/dev/rdisk/c5t5080020000557D01d0s4	/dev/rdisk/c8t508002000050F053d0s4
/dev/md/rdisk/d264	/dev/rdisk/c4t5080020000557D81d0s4 /dev/rdisk/c14t508002000050F001d0s4	/dev/rdisk/c9t5080020000557D40d0s4
600144f07d498e0000000000d47d265 16466771968	/dev/rdisk/c8t508002000050F041d0s4	/dev/rdisk/c2t5080020000557DC0d0s4
/dev/md/rdisk/d265	/dev/rdisk/c5t5080020000557D02d0s4	/dev/rdisk/c1t508002000050F0C0d0s4 /dev/rdisk/c11t508002000050F080d0s4
600144f07d498e0000000000d47d266 16466771968	/dev/rdisk/c4t5080020000557D82d0s4 /dev/rdisk/c14t508002000050F002d0s4	/dev/rdisk/c9t5080020000557D41d0s4
/dev/md/rdisk/d266	/dev/rdisk/c8t508002000050F042d0s4	/dev/rdisk/c2t5080020000557DC1d0s4
600144f07d498e0000000000d47d267 16466771968	/dev/rdisk/c5t5080020000557D03d0s4	/dev/rdisk/c1t508002000050F0C1d0s4 /dev/rdisk/c11t508002000050F081d0s4
/dev/md/rdisk/d267	/dev/rdisk/c4t5080020000557D83d0s4 /dev/rdisk/c14t508002000050F003d0s4	/dev/rdisk/c9t5080020000557D42d0s4
600144f07d498e0000000000d47d268 16466771968	/dev/rdisk/c8t508002000050F043d0s4	/dev/rdisk/c2t5080020000557DC2d0s4
/dev/md/rdisk/d268	/dev/rdisk/c5t5080020000557D04d0s4	/dev/rdisk/c11t508002000050F082d0s4
600144f07d498e0000000000d47d269 16466771968	/dev/rdisk/c4t5080020000557D84d0s4 /dev/rdisk/c14t508002000050F004d0s4	/dev/rdisk/c9t5080020000557D43d0s4
/dev/md/rdisk/d269	/dev/rdisk/c8t508002000050F044d0s4	/dev/rdisk/c2t5080020000557DC3d0s4
600144f07d498e0000000000d47d270 16466771968	/dev/rdisk/c5t5080020000557D05d0s4	/dev/rdisk/c1t508002000050F0C3d0s4 /dev/rdisk/c11t508002000050F083d0s4
/dev/md/rdisk/d270	/dev/rdisk/c4t5080020000557D85d0s4 /dev/rdisk/c14t508002000050F005d0s4	/dev/rdisk/c9t5080020000557D44d0s4
600144f07d498e0000000000d47d271 16466771968	/dev/rdisk/c8t508002000050F045d0s4	/dev/rdisk/c2t5080020000557DC4d0s4
/dev/md/rdisk/d271	/dev/rdisk/c5t5080020000557D06d0s4	/dev/rdisk/c1t508002000050F0C4d0s4 /dev/rdisk/c11t508002000050F084d0s4
600144f07d498e0000000000d47d272 16466771968	/dev/rdisk/c4t5080020000557D86d0s4 /dev/rdisk/c14t508002000050F006d0s4	/dev/rdisk/c9t5080020000557D45d0s4
/dev/md/rdisk/d272		/dev/rdisk/c2t5080020000557DC5d0s4
600144f07d498e0000000000d47d273 16466771968		/dev/rdisk/c1t508002000050F0C5d0s4 /dev/rdisk/c11t508002000050F085d0s4

/dev/rds/c9t5080020000557D46d0s4 /dev/rds/c8t508002000050F045d0s7 /dev/rds/c1t508002000050F0C4d0s7 /dev/rds/c11t508002000050F085d0s7  
/dev/rds/c2t5080020000557DC6d0s4 /dev/rds/c5t5080020000557D05d0s7 /dev/rds/c9t5080020000557D45d0s7  
/dev/rds/c1t508002000050F0C6d0s4 /dev/rds/c11t508002000050F086d0s4 /dev/rds/c4t5080020000557D85d0s7 /dev/rds/c14t508002000050F005d0s7 /dev/rds/c2t5080020000557DC5d0s7  
/dev/rds/c9t5080020000557D47d0s4 /dev/rds/c8t508002000050F046d0s7 /dev/rds/c1t508002000050F0C5d0s7 /dev/rds/c11t508002000050F086d0s7  
/dev/rds/c2t5080020000557DC7d0s4 /dev/rds/c5t5080020000557D06d0s7 /dev/rds/c9t5080020000557D46d0s7  
/dev/rds/c1t508002000050F0C7d0s4 /dev/rds/c11t508002000050F087d0s4 /dev/rds/c4t5080020000557D86d0s7 /dev/rds/c14t508002000050F006d0s7 /dev/rds/c2t5080020000557DC6d0s7  
/dev/rds/c9t5080020000557D48d0s4 /dev/rds/c8t508002000050F047d0s7 /dev/rds/c1t508002000050F0C6d0s7 /dev/rds/c11t508002000050F087d0s7  
/dev/rds/c2t5080020000557DC8d0s4 /dev/rds/c5t5080020000557D07d0s7 /dev/rds/c9t5080020000557D47d0s7  
/dev/rds/c1t508002000050F0C8d0s4 /dev/rds/c11t508002000050F088d0s4 /dev/rds/c4t5080020000557D87d0s7 /dev/rds/c14t508002000050F007d0s7 /dev/rds/c2t5080020000557DC7d0s7  
/dev/rds/c9t5080020000557D49d0s4 /dev/rds/c8t508002000050F048d0s7 /dev/rds/c1t508002000050F0C7d0s7 /dev/rds/c11t508002000050F088d0s7  
/dev/rds/c2t5080020000557DC9d0s4 /dev/rds/c5t5080020000557D08d0s7 /dev/rds/c9t5080020000557D48d0s7  
/dev/rds/c1t508002000050F0C9d0s4 /dev/rds/c11t508002000050F089d0s4 /dev/rds/c4t5080020000557D88d0s7 /dev/rds/c14t508002000050F008d0s7 /dev/rds/c2t5080020000557DC8d0s7  
/dev/rds/c9t5080020000557D4A0s4 /dev/rds/c8t508002000050F049d0s7 /dev/rds/c1t508002000050F0C8d0s7 /dev/rds/c11t508002000050F089d0s7  
/dev/rds/c2t5080020000557DCA0s4 /dev/rds/c5t5080020000557D09d0s7 /dev/rds/c9t5080020000557D49d0s7  
/dev/rds/c1t508002000050F0CA0s4 /dev/rds/c4t5080020000557D89d0s7 /dev/rds/c14t508002000050F009d0s7 /dev/rds/c2t5080020000557DC9d0s7  
/dev/rds/c11t508002000050F08A0s4 /dev/rds/c8t508002000050F04Ad0s7 /dev/rds/c5t5080020000557D0Ad0s7 /dev/rds/c1t508002000050F0C9d0s7  
/dev/rds/c9t5080020000557D4Bd0s4 /dev/rds/c4t5080020000557D8Ad0s7 /dev/rds/c11t508002000050F08Ad0s7  
/dev/rds/c2t5080020000557DCBd0s4 /dev/rds/c14t508002000050F00Ad0s7 /dev/rds/c9t5080020000557D4Ad0s7  
/dev/rds/c1t508002000050F0CBd0s4 /dev/rds/c8t508002000050F04Bd0s7 /dev/rds/c1t508002000050F0CAd0s7  
/dev/rds/c11t508002000050F08Bd0s4 /dev/rds/c5t5080020000557D0Bd0s7 /dev/rds/c4t5080020000557D8Bd0s7 /dev/rds/c1t508002000050F0CAd0s7  
/dev/rds/c9t5080020000557D4Cd0s4 /dev/rds/c14t508002000050F00Bd0s7 /dev/rds/c11t508002000050F08Bd0s7  
/dev/rds/c2t5080020000557DCCd0s4 /dev/rds/c8t508002000050F04Cd0s7 /dev/rds/c9t5080020000557D4Bd0s7  
/dev/rds/c1t508002000050F0CCd0s4 /dev/rds/c5t5080020000557D0Cd0s7 /dev/rds/c4t5080020000557D8Cd0s7 /dev/rds/c2t5080020000557DCBd0s7  
/dev/rds/c9t5080020000557D4Dd0s4 /dev/rds/c14t508002000050F00Cd0s7 /dev/rds/c1t508002000050F0CBd0s7  
/dev/rds/c2t5080020000557DCDd0s4 /dev/rds/c8t508002000050F04Dd0s7 /dev/rds/c5t5080020000557D0Dd0s7 /dev/rds/c11t508002000050F08Cd0s7  
/dev/rds/c1t508002000050F0CDd0s4 /dev/rds/c14t508002000050F00Dd0s7 /dev/rds/c9t5080020000557D4Cd0s7  
/dev/rds/c9t5080020000557D4Ed0s4 /dev/rds/c2t5080020000557DCEd0s4 /dev/rds/c5t5080020000557D0Ed0s7 /dev/rds/c4t5080020000557D8Ed0s7 /dev/rds/c11t508002000050F08Dd0s7  
/dev/rds/c1t508002000050F0CEDd0s4 /dev/rds/c14t508002000050F00Ed0s7 /dev/rds/c8t508002000050F04Fd0s7 /dev/rds/c9t5080020000557D4Dd0s7  
/dev/rds/c11t508002000050F08Ed0s4 /dev/rds/c5t5080020000557D0Fd0s7 /dev/rds/c2t5080020000557DCDd0s7  
/dev/rds/c9t5080020000557D4Fd0s4 /dev/rds/c2t5080020000557DCFd0s4 /dev/rds/c4t5080020000557D8Fd0s7 /dev/rds/c1t508002000050F0CFd0s7  
/dev/rds/c1t508002000050F0CFd0s4 /dev/rds/c11t508002000050F08Fd0s4 /dev/rds/c14t508002000050F00Fd0s7 /dev/rds/c8t508002000050F050d0s7 /dev/rds/c11t508002000050F08Ed0s7  
/dev/rds/c9t5080020000557D50d0s4 /dev/rds/c2t5080020000557DD0d0s4 /dev/rds/c5t5080020000557D10d0s7 /dev/rds/c9t5080020000557D4Ed0s7 /dev/rds/c2t5080020000557DCEd0s7  
/dev/rds/c1t508002000050F0D0d0s4 /dev/rds/c11t508002000050F090d0s4 /dev/rds/c4t5080020000557D90d0s7 /dev/rds/c14t508002000050F010d0s7 /dev/rds/c1t508002000050F0CEd0s7  
/dev/rds/c9t5080020000557D51d0s4 /dev/rds/c2t5080020000557DD1d0s4 /dev/rds/c8t508002000050F051d0s7 /dev/rds/c9t5080020000557D4Fd0s7 /dev/rds/c1t508002000050F08Fd0s7  
/dev/rds/c1t508002000050F0D1d0s4 /dev/rds/c11t508002000050F091d0s4 /dev/rds/c5t5080020000557D11d0s7 /dev/rds/c2t5080020000557DCFd0s7 /dev/rds/c1t508002000050F0CFd0s7  
/dev/rds/c9t5080020000557D52d0s4 /dev/rds/c2t5080020000557DD2d0s4 /dev/rds/c4t5080020000557D91d0s7 /dev/rds/c14t508002000050F011d0s7 /dev/rds/c11t508002000050F090d0s7 /dev/rds/c9t5080020000557D50d0s7  
/dev/rds/c1t508002000050F0D2d0s4 /dev/rds/c11t508002000050F092d0s4 /dev/rds/c8t508002000050F052d0s7 /dev/rds/c2t5080020000557DD0d0s7 /dev/rds/c1t508002000050F0D0d0s7  
/dev/rds/c9t5080020000557D53d0s4 /dev/rds/c2t5080020000557DD3d0s4 /dev/rds/c5t5080020000557D12d0s7 /dev/rds/c11t508002000050F091d0s7 /dev/rds/c9t5080020000557D51d0s7  
/dev/rds/c1t508002000050F0D3d0s4 /dev/rds/c11t508002000050F093d0s4 /dev/rds/c4t5080020000557D92d0s7 /dev/rds/c14t508002000050F012d0s7 /dev/rds/c2t5080020000557DD3d0s7 /dev/rds/c1t508002000050F0D3d0s7  
-i 256b /dev/rds/c8t508002000050F053d0s7 /dev/rds/c11t508002000050F092d0s7 /dev/rds/c9t5080020000557D52d0s7  
d288 -p /dev/md/rds/d3 -o 1816964864 -b 1032192 /dev/rds/c5t5080020000557D13d0s7 /dev/rds/c2t5080020000557DD2d0s7 /dev/rds/c1t508002000050F0D2d0s7  
d3 1 160 /dev/rds/c8t508002000050F040d0s7 /dev/rds/c4t5080020000557D93d0s7 /dev/rds/c14t508002000050F013d0s7 /dev/rds/c11t508002000050F093d0s7 /dev/rds/c9t5080020000557D53d0s7  
/dev/rds/c5t5080020000557D00d0s7 /dev/rds/c11t508002000050F080d0s7 /dev/rds/c9t5080020000557D40d0s7 /dev/rds/c2t5080020000557DD3d0s7 /dev/rds/c1t508002000050F0D3d0s7  
/dev/rds/c4t5080020000557D80d0s7 /dev/rds/c14t508002000050F000d0s7 /dev/rds/c2t5080020000557DC0d0s7 /dev/rds/c11t508002000050F081d0s7 d287 -p /dev/md/rds/d2 -o 1816964864 -b 1032192  
/dev/rds/c8t508002000050F041d0s7 /dev/rds/c1t508002000050F0C0d0s7 /dev/rds/c9t5080020000557D41d0s7 d2 1 160 /dev/rds/c14t508002000050F000d0s6  
/dev/rds/c5t5080020000557D01d0s7 /dev/rds/c9t5080020000557D41d0s7 /dev/rds/c2t5080020000557DC1d0s7 /dev/rds/c1t508002000050F040d0s6  
/dev/rds/c4t5080020000557D81d0s7 /dev/rds/c14t508002000050F001d0s7 /dev/rds/c1t508002000050F0C1d0s7 /dev/rds/c11t508002000050F082d0s7 /dev/rds/c5t5080020000557D00d0s6  
/dev/rds/c8t508002000050F042d0s7 /dev/rds/c1t508002000050F0C1d0s7 /dev/rds/c11t508002000050F082d0s7 /dev/rds/c5t5080020000557D00d0s6 /dev/rds/c4t5080020000557D80d0s6 /dev/rds/c14t508002000050F001d0s6  
/dev/rds/c5t5080020000557D02d0s7 /dev/rds/c9t5080020000557D42d0s7 /dev/rds/c2t5080020000557DC2d0s7 /dev/rds/c8t508002000050F041d0s6  
/dev/rds/c4t5080020000557D82d0s7 /dev/rds/c14t508002000050F002d0s7 /dev/rds/c1t508002000050F0C2d0s7 /dev/rds/c11t508002000050F083d0s7 /dev/rds/c5t5080020000557D01d0s6  
/dev/rds/c8t508002000050F043d0s7 /dev/rds/c1t508002000050F0C2d0s7 /dev/rds/c9t5080020000557D81d0s6 /dev/rds/c4t5080020000557D81d0s6 /dev/rds/c14t508002000050F002d0s6  
/dev/rds/c5t5080020000557D03d0s7 /dev/rds/c9t5080020000557D43d0s7 /dev/rds/c2t5080020000557DC3d0s7 /dev/rds/c8t508002000050F042d0s6  
/dev/rds/c4t5080020000557D83d0s7 /dev/rds/c14t508002000050F003d0s7 /dev/rds/c1t508002000050F0C3d0s7 /dev/rds/c11t508002000050F084d0s7 /dev/rds/c5t5080020000557D02d0s6  
/dev/rds/c8t508002000050F044d0s7 /dev/rds/c1t508002000050F0C3d0s7 /dev/rds/c9t5080020000557D44d0s7 /dev/rds/c4t5080020000557D82d0s6 /dev/rds/c14t508002000050F003d0s6  
/dev/rds/c5t5080020000557D04d0s7 /dev/rds/c9t5080020000557D44d0s7 /dev/rds/c2t5080020000557DC4d0s7 /dev/rds/c8t508002000050F043d0s6  
/dev/rds/c4t5080020000557D84d0s7 /dev/rds/c14t508002000050F004d0s7 /dev/rds/c1t508002000050F0C4d0s7



/dev/rdisk/c2t5080020000557DC2d0s5 d279 -p /dev/md/rdisk/d2 -o 1752640768 -b 32161792  
/dev/rdisk/c1t508002000050FC2d0s5 /dev/rdisk/c11t508002000050F082d0s5 d278 -p /dev/md/rdisk/d1 -o 1752640768 -b 32161792  
/dev/rdisk/c9t5080020000557D42d0s5 d277 -p /dev/md/rdisk/d0 -o 1752640768 -b 32161792  
/dev/rdisk/c2t5080020000557DC3d0s5 d276 -p /dev/md/rdisk/d3 -o 1720478720 -b 32161792  
/dev/rdisk/c1t508002000050F0C3d0s5 /dev/rdisk/c11t508002000050F083d0s5 d275 -p /dev/md/rdisk/d2 -o 1720478720 -b 32161792  
/dev/rdisk/c9t5080020000557D43d0s5 d274 -p /dev/md/rdisk/d1 -o 1720478720 -b 32161792  
/dev/rdisk/c2t5080020000557DC4d0s5 d273 -p /dev/md/rdisk/d0 -o 1720478720 -b 32161792  
/dev/rdisk/c1t508002000050F0C4d0s5 /dev/rdisk/c11t508002000050F084d0s5 d272 -p /dev/md/rdisk/d3 -o 1688316672 -b 32161792  
/dev/rdisk/c9t5080020000557D44d0s5 d271 -p /dev/md/rdisk/d2 -o 1688316672 -b 32161792  
/dev/rdisk/c2t5080020000557DC5d0s5 d270 -p /dev/md/rdisk/d1 -o 1688316672 -b 32161792  
/dev/rdisk/c1t508002000050F0C5d0s5 /dev/rdisk/c11t508002000050F085d0s5 d269 -p /dev/md/rdisk/d0 -o 1688316672 -b 32161792  
/dev/rdisk/c9t5080020000557D45d0s5 d268 -p /dev/md/rdisk/d3 -o 1656154624 -b 32161792  
/dev/rdisk/c2t5080020000557DC6d0s5 d267 -p /dev/md/rdisk/d2 -o 1656154624 -b 32161792  
/dev/rdisk/c1t508002000050F0C6d0s5 /dev/rdisk/c11t508002000050F086d0s5 d266 -p /dev/md/rdisk/d1 -o 1656154624 -b 32161792  
/dev/rdisk/c9t5080020000557D46d0s5 d265 -p /dev/md/rdisk/d0 -o 1656154624 -b 32161792  
/dev/rdisk/c2t5080020000557DC7d0s5 d264 -p /dev/md/rdisk/d3 -o 1623992576 -b 32161792  
/dev/rdisk/c1t508002000050F0C7d0s5 /dev/rdisk/c11t508002000050F087d0s5 d263 -p /dev/md/rdisk/d2 -o 1623992576 -b 32161792  
/dev/rdisk/c9t5080020000557D47d0s5 d262 -p /dev/md/rdisk/d1 -o 1623992576 -b 32161792  
/dev/rdisk/c2t5080020000557DC8d0s5 d261 -p /dev/md/rdisk/d0 -o 1623992576 -b 32161792  
/dev/rdisk/c1t508002000050F0C8d0s5 /dev/rdisk/c11t508002000050F088d0s5 d260 -p /dev/md/rdisk/d3 -o 1591830528 -b 32161792  
/dev/rdisk/c9t5080020000557D48d0s5 d259 -p /dev/md/rdisk/d2 -o 1591830528 -b 32161792  
/dev/rdisk/c2t5080020000557DC9d0s5 d258 -p /dev/md/rdisk/d1 -o 1591830528 -b 32161792  
/dev/rdisk/c1t508002000050F0C9d0s5 /dev/rdisk/c11t508002000050F089d0s5 d257 -p /dev/md/rdisk/d0 -o 1591830528 -b 32161792  
/dev/rdisk/c9t5080020000557D49d0s5 /dev/rdisk/c2t5080020000557DCAd0s5 d256 -p /dev/md/rdisk/d3 -o 1559668480 -b 32161792  
/dev/rdisk/c1t508002000050F0CAd0s5 d255 -p /dev/md/rdisk/d2 -o 1559668480 -b 32161792  
/dev/rdisk/c11t508002000050F08Ad0s5 d254 -p /dev/md/rdisk/d1 -o 1559668480 -b 32161792  
/dev/rdisk/c9t5080020000557D4Ad0s5 d253 -p /dev/md/rdisk/d0 -o 1559668480 -b 32161792  
/dev/rdisk/c2t5080020000557DCBd0s5 d252 -p /dev/md/rdisk/d3 -o 1527506432 -b 32161792  
/dev/rdisk/c1t508002000050F0CBd0s5 d251 -p /dev/md/rdisk/d2 -o 1527506432 -b 32161792  
/dev/rdisk/c11t508002000050F08Bd0s5 d250 -p /dev/md/rdisk/d1 -o 1527506432 -b 32161792  
/dev/rdisk/c9t5080020000557D4Bd0s5 d249 -p /dev/md/rdisk/d0 -o 1527506432 -b 32161792  
/dev/rdisk/c2t5080020000557DCCd0s5 d248 -p /dev/md/rdisk/d3 -o 1495344384 -b 32161792  
/dev/rdisk/c1t508002000050F0CCd0s5 d247 -p /dev/md/rdisk/d2 -o 1495344384 -b 32161792  
/dev/rdisk/c11t508002000050F08Cd0s5 d246 -p /dev/md/rdisk/d1 -o 1495344384 -b 32161792  
/dev/rdisk/c9t5080020000557D4Cd0s5 d245 -p /dev/md/rdisk/d0 -o 1495344384 -b 32161792  
/dev/rdisk/c2t5080020000557DCDd0s5 d244 -p /dev/md/rdisk/d3 -o 1463182336 -b 32161792  
/dev/rdisk/c1t508002000050F0CDd0s5 d243 -p /dev/md/rdisk/d2 -o 1463182336 -b 32161792  
/dev/rdisk/c11t508002000050F08Dd0s5 d242 -p /dev/md/rdisk/d1 -o 1463182336 -b 32161792  
/dev/rdisk/c9t5080020000557D4Dd0s5 d241 -p /dev/md/rdisk/d0 -o 1463182336 -b 32161792  
/dev/rdisk/c2t5080020000557DCEd0s5 /dev/rdisk/c1t508002000050F0CEd0s5 d240 -p /dev/md/rdisk/d3 -o 1431020288 -b 32161792  
/dev/rdisk/c11t508002000050F08Ed0s5 d239 -p /dev/md/rdisk/d2 -o 1431020288 -b 32161792  
/dev/rdisk/c9t5080020000557D4Ed0s5 /dev/rdisk/c2t5080020000557DCFd0s5 d238 -p /dev/md/rdisk/d1 -o 1431020288 -b 32161792  
/dev/rdisk/c1t508002000050F0CFd0s5 /dev/rdisk/c11t508002000050F08Fd0s5 d237 -p /dev/md/rdisk/d0 -o 1431020288 -b 32161792  
/dev/rdisk/c9t5080020000557D4Fd0s5 /dev/rdisk/c2t5080020000557DD0d0s5 d236 -p /dev/md/rdisk/d3 -o 1398858240 -b 32161792  
/dev/rdisk/c1t508002000050F0D0d0s5 /dev/rdisk/c11t508002000050F09d0s5 d235 -p /dev/md/rdisk/d2 -o 1398858240 -b 32161792  
/dev/rdisk/c9t5080020000557D50d0s5 /dev/rdisk/c2t5080020000557DD1d0s5 d234 -p /dev/md/rdisk/d1 -o 1398858240 -b 32161792  
/dev/rdisk/c1t508002000050F0D1d0s5 /dev/rdisk/c11t508002000050F091d0s5 d233 -p /dev/md/rdisk/d0 -o 1398858240 -b 32161792  
/dev/rdisk/c9t5080020000557D51d0s5 /dev/rdisk/c2t5080020000557DD2d0s5 d232 -p /dev/md/rdisk/d3 -o 1366696192 -b 32161792  
/dev/rdisk/c1t508002000050F0D2d0s5 /dev/rdisk/c11t508002000050F092d0s5 d231 -p /dev/md/rdisk/d2 -o 1366696192 -b 32161792  
/dev/rdisk/c9t5080020000557D52d0s5 /dev/rdisk/c2t5080020000557DD3d0s5 d230 -p /dev/md/rdisk/d1 -o 1366696192 -b 32161792  
/dev/rdisk/c1t508002000050F0D3d0s5 /dev/rdisk/c11t508002000050F093d0s5 d229 -p /dev/md/rdisk/d0 -o 1366696192 -b 32161792  
/dev/rdisk/c9t5080020000557D53d0s5 -i 256b d228 -p /dev/md/rdisk/d3 -o 1334534144 -b 32161792  
d285 -p /dev/md/rdisk/d0 -o 1816964864 -b 1032192 d227 -p /dev/md/rdisk/d2 -o 1334534144 -b 32161792  
d284 -p /dev/md/rdisk/d3 -o 1784802816 -b 32161792 d226 -p /dev/md/rdisk/d1 -o 1334534144 -b 32161792  
d283 -p /dev/md/rdisk/d2 -o 1784802816 -b 32161792 d225 -p /dev/md/rdisk/d0 -o 1334534144 -b 32161792  
d282 -p /dev/md/rdisk/d1 -o 1784802816 -b 32161792 d224 -p /dev/md/rdisk/d3 -o 1302372096 -b 32161792  
d281 -p /dev/md/rdisk/d0 -o 1784802816 -b 32161792 d223 -p /dev/md/rdisk/d2 -o 1302372096 -b 32161792  
d280 -p /dev/md/rdisk/d3 -o 1752640768 -b 32161792 d222 -p /dev/md/rdisk/d1 -o 1302372096 -b 32161792  
d221 -p /dev/md/rdisk/d0 -o 1302372096 -b 32161792  
d220 -p /dev/md/rdisk/d3 -o 1270210048 -b 32161792  
d219 -p /dev/md/rdisk/d2 -o 1270210048 -b 32161792  
d218 -p /dev/md/rdisk/d1 -o 1270210048 -b 32161792  
d217 -p /dev/md/rdisk/d0 -o 1270210048 -b 32161792  
d216 -p /dev/md/rdisk/d3 -o 1238048000 -b 32161792  
d215 -p /dev/md/rdisk/d2 -o 1238048000 -b 32161792  
d214 -p /dev/md/rdisk/d1 -o 1238048000 -b 32161792  
d213 -p /dev/md/rdisk/d0 -o 1238048000 -b 32161792  
d212 -p /dev/md/rdisk/d3 -o 1205885952 -b 32161792  
d211 -p /dev/md/rdisk/d2 -o 1205885952 -b 32161792  
d210 -p /dev/md/rdisk/d1 -o 1205885952 -b 32161792  
d209 -p /dev/md/rdisk/d0 -o 1205885952 -b 32161792  
d208 -p /dev/md/rdisk/d3 -o 1085045504 -b 120840192  
d207 -p /dev/md/rdisk/d2 -o 1085045504 -b 120840192  
d206 -p /dev/md/rdisk/d1 -o 1085045504 -b 120840192  
d205 -p /dev/md/rdisk/d0 -o 1085045504 -b 120840192  
d204 -p /dev/md/rdisk/d3 -o 964205056 -b 120840192  
d203 -p /dev/md/rdisk/d2 -o 964205056 -b 120840192  
d202 -p /dev/md/rdisk/d1 -o 964205056 -b 120840192  
d201 -p /dev/md/rdisk/d0 -o 964205056 -b 120840192  
d200 -p /dev/md/rdisk/d3 -o 843364608 -b 120840192  
d199 -p /dev/md/rdisk/d2 -o 843364608 -b 120840192  
d198 -p /dev/md/rdisk/d1 -o 843364608 -b 120840192  
d197 -p /dev/md/rdisk/d0 -o 843364608 -b 120840192  
d196 -p /dev/md/rdisk/d3 -o 722524160 -b 120840192  
d195 -p /dev/md/rdisk/d2 -o 722524160 -b 120840192  
d194 -p /dev/md/rdisk/d1 -o 722524160 -b 120840192  
d193 -p /dev/md/rdisk/d0 -o 722524160 -b 120840192  
d192 -p /dev/md/rdisk/d3 -o 601683712 -b 120840192  
d191 -p /dev/md/rdisk/d2 -o 601683712 -b 120840192  
d190 -p /dev/md/rdisk/d1 -o 601683712 -b 120840192  
d189 -p /dev/md/rdisk/d0 -o 601683712 -b 120840192  
d188 -p /dev/md/rdisk/d3 -o 594302464 -b 7380992  
d187 -p /dev/md/rdisk/d2 -o 594302464 -b 7380992  
d186 -p /dev/md/rdisk/d1 -o 594302464 -b 7380992  
d185 -p /dev/md/rdisk/d0 -o 594302464 -b 7380992  
d184 -p /dev/md/rdisk/d3 -o 551286016 -b 43016192  
d183 -p /dev/md/rdisk/d2 -o 551286016 -b 43016192  
d182 -p /dev/md/rdisk/d1 -o 551286016 -b 43016192  
d181 -p /dev/md/rdisk/d0 -o 551286016 -b 43016192  
d180 -p /dev/md/rdisk/d3 -o 536736768 -b 14548992  
d179 -p /dev/md/rdisk/d2 -o 536736768 -b 14548992  
d178 -p /dev/md/rdisk/d1 -o 536736768 -b 14548992  
d177 -p /dev/md/rdisk/d0 -o 536736768 -b 14548992  
d176 -p /dev/md/rdisk/d3 -o 522187520 -b 14548992  
d175 -p /dev/md/rdisk/d2 -o 522187520 -b 14548992  
d174 -p /dev/md/rdisk/d1 -o 522187520 -b 14548992  
d173 -p /dev/md/rdisk/d0 -o 522187520 -b 14548992  
d172 -p /dev/md/rdisk/d3 -o 490435072 -b 31752192  
d171 -p /dev/md/rdisk/d2 -o 490435072 -b 31752192  
d170 -p /dev/md/rdisk/d1 -o 490435072 -b 31752192  
d169 -p /dev/md/rdisk/d0 -o 490435072 -b 31752192  
d168 -p /dev/md/rdisk/d3 -o 475271424 -b 15163392  
d167 -p /dev/md/rdisk/d2 -o 475271424 -b 15163392  
d166 -p /dev/md/rdisk/d1 -o 475271424 -b 15163392  
d165 -p /dev/md/rdisk/d0 -o 475271424 -b 15163392  
d164 -p /dev/md/rdisk/d3 -o 452734976 -b 22536192

d163 -p /dev/md/rdsk/d2 -o 452734976 -b 22536192  
d162 -p /dev/md/rdsk/d1 -o 452734976 -b 22536192  
d161 -p /dev/md/rdsk/d0 -o 452734976 -b 22536192  
d160 -p /dev/md/rdsk/d3 -o 430198528 -b 22536192  
d159 -p /dev/md/rdsk/d2 -o 430198528 -b 22536192  
d158 -p /dev/md/rdsk/d1 -o 430198528 -b 22536192  
d157 -p /dev/md/rdsk/d0 -o 430198528 -b 22536192  
d156 -p /dev/md/rdsk/d3 -o 399470080 -b 30728192  
d155 -p /dev/md/rdsk/d2 -o 399470080 -b 30728192  
d154 -p /dev/md/rdsk/d1 -o 399470080 -b 30728192  
d153 -p /dev/md/rdsk/d0 -o 399470080 -b 30728192  
d152 -p /dev/md/rdsk/d3 -o 368741632 -b 30728192  
d151 -p /dev/md/rdsk/d2 -o 368741632 -b 30728192  
d150 -p /dev/md/rdsk/d1 -o 368741632 -b 30728192  
d149 -p /dev/md/rdsk/d0 -o 368741632 -b 30728192  
d148 -p /dev/md/rdsk/d3 -o 338013184 -b 30728192  
d147 -p /dev/md/rdsk/d2 -o 338013184 -b 30728192  
d146 -p /dev/md/rdsk/d1 -o 338013184 -b 30728192  
d145 -p /dev/md/rdsk/d0 -o 338013184 -b 30728192  
d144 -p /dev/md/rdsk/d3 -o 307284736 -b 30728192  
d143 -p /dev/md/rdsk/d2 -o 307284736 -b 30728192  
d142 -p /dev/md/rdsk/d1 -o 307284736 -b 30728192  
d141 -p /dev/md/rdsk/d0 -o 307284736 -b 30728192  
d140 -p /dev/md/rdsk/d3 -o 276556288 -b 30728192  
d139 -p /dev/md/rdsk/d2 -o 276556288 -b 30728192  
d138 -p /dev/md/rdsk/d1 -o 276556288 -b 30728192  
d137 -p /dev/md/rdsk/d0 -o 276556288 -b 30728192  
d136 -p /dev/md/rdsk/d3 -o 245827840 -b 30728192  
d135 -p /dev/md/rdsk/d2 -o 245827840 -b 30728192  
d134 -p /dev/md/rdsk/d1 -o 245827840 -b 30728192  
d133 -p /dev/md/rdsk/d0 -o 245827840 -b 30728192  
d132 -p /dev/md/rdsk/d3 -o 215099392 -b 30728192  
d131 -p /dev/md/rdsk/d2 -o 215099392 -b 30728192  
d130 -p /dev/md/rdsk/d1 -o 215099392 -b 30728192  
d129 -p /dev/md/rdsk/d0 -o 215099392 -b 30728192  
d128 -p /dev/md/rdsk/d3 -o 184370944 -b 30728192  
d127 -p /dev/md/rdsk/d2 -o 184370944 -b 30728192  
d126 -p /dev/md/rdsk/d1 -o 184370944 -b 30728192  
d125 -p /dev/md/rdsk/d0 -o 184370944 -b 30728192  
d124 -p /dev/md/rdsk/d3 -o 153642496 -b 30728192  
d123 -p /dev/md/rdsk/d2 -o 153642496 -b 30728192  
d122 -p /dev/md/rdsk/d1 -o 153642496 -b 30728192  
d121 -p /dev/md/rdsk/d0 -o 153642496 -b 30728192  
d120 -p /dev/md/rdsk/d3 -o 122914048 -b 30728192  
d119 -p /dev/md/rdsk/d2 -o 122914048 -b 30728192  
d118 -p /dev/md/rdsk/d1 -o 122914048 -b 30728192  
d117 -p /dev/md/rdsk/d0 -o 122914048 -b 30728192  
d116 -p /dev/md/rdsk/d3 -o 92185600 -b 30728192  
d115 -p /dev/md/rdsk/d2 -o 92185600 -b 30728192  
d114 -p /dev/md/rdsk/d1 -o 92185600 -b 30728192  
d113 -p /dev/md/rdsk/d0 -o 92185600 -b 30728192  
d112 -p /dev/md/rdsk/d3 -o 61457152 -b 30728192  
d111 -p /dev/md/rdsk/d2 -o 61457152 -b 30728192  
d110 -p /dev/md/rdsk/d1 -o 61457152 -b 30728192  
d109 -p /dev/md/rdsk/d0 -o 61457152 -b 30728192  
d108 -p /dev/md/rdsk/d3 -o 30728704 -b 30728192  
d107 -p /dev/md/rdsk/d2 -o 30728704 -b 30728192  
d106 -p /dev/md/rdsk/d1 -o 30728704 -b 30728192  
d105 -p /dev/md/rdsk/d0 -o 30728704 -b 30728192  
d104 -p /dev/md/rdsk/d3 -o 256 -b 30728192  
d103 -p /dev/md/rdsk/d2 -o 256 -b 30728192  
d102 -p /dev/md/rdsk/d1 -o 256 -b 30728192  
d101 -p /dev/md/rdsk/d0 -o 256 -b 30728192  
d3289 -p /dev/md/rdsk/d8001 -o 2743064576 -b 108544  
d8001 -m /dev/md/rdsk/d8012 /dev/md/rdsk/d8013 0  
d8012 1 1 /dev/rdsk/c3t3d0s6  
d8013 1 1 /dev/rdsk/c3t4d0s6  
d3285 -p /dev/md/rdsk/d8002 -o 2921345024 -b 1032192  
d8002 -m /dev/md/rdsk/d8014 /dev/md/rdsk/d8015 0  
d8014 1 1 /dev/rdsk/c3t5d0s6  
d8015 1 1 /dev/rdsk/c3t6d0s6  
d3286 -p /dev/md/rdsk/d8003 -o 2907828224 -b 1032192  
d8003 -m /dev/md/rdsk/d8016 /dev/md/rdsk/d8017 0  
d8016 1 1 /dev/rdsk/c3t7d0s6  
d8017 1 1 /dev/rdsk/c3t8d0s6  
d3287 -p /dev/md/rdsk/d8004 -o 2845569024 -b 1032192  
d8004 -m /dev/md/rdsk/d8018 /dev/md/rdsk/d8019 0  
d8018 1 1 /dev/rdsk/c3t9d0s6  
d8019 1 1 /dev/rdsk/c3t10d0s6  
d3288 -p /dev/md/rdsk/d8004 -o 2844535808 -b 1032192  
d3209 -p /dev/md/rdsk/d8003 -o 2875665408 -b 32161792  
d3210 -p /dev/md/rdsk/d8002 -o 2889182208 -b 32161792  
d3211 -p /dev/md/rdsk/d8001 -o 2710901760 -b 32161792  
d3212 -p /dev/md/rdsk/d8000 -o 3091310592 -b 32161792  
d8000 -m /dev/md/rdsk/d8010 /dev/md/rdsk/d8011 0  
d8010 1 1 /dev/rdsk/c3t1d0s6  
d8011 1 1 /dev/rdsk/c3t2d0s6  
d3213 -p /dev/md/rdsk/d8000 -o 3059147776 -b 32161792  
d3214 -p /dev/md/rdsk/d8001 -o 2678738944 -b 32161792  
d3215 -p /dev/md/rdsk/d8002 -o 2857019392 -b 32161792  
d3216 -p /dev/md/rdsk/d8003 -o 2843502592 -b 32161792  
d3217 -p /dev/md/rdsk/d8004 -o 2812372992 -b 32161792  
d3218 -p /dev/md/rdsk/d8000 -o 2780210176 -b 32161792  
d3219 -p /dev/md/rdsk/d8003 -o 2811339776 -b 32161792  
d3220 -p /dev/md/rdsk/d8002 -o 2824856576 -b 32161792  
d3221 -p /dev/md/rdsk/d8001 -o 2646576128 -b 32161792  
d3222 -p /dev/md/rdsk/d8000 -o 3026984960 -b 32161792  
d3223 -p /dev/md/rdsk/d8000 -o 2994822144 -b 32161792  
d3224 -p /dev/md/rdsk/d8001 -o 2614413312 -b 32161792  
d3225 -p /dev/md/rdsk/d8002 -o 2792693760 -b 32161792  
d3226 -p /dev/md/rdsk/d8003 -o 2779176960 -b 32161792  
d3227 -p /dev/md/rdsk/d8004 -o 2748047360 -b 32161792  
d3228 -p /dev/md/rdsk/d8004 -o 2715884544 -b 32161792  
d3229 -p /dev/md/rdsk/d8003 -o 2747014144 -b 32161792  
d3230 -p /dev/md/rdsk/d8002 -o 2760530944 -b 32161792  
d3231 -p /dev/md/rdsk/d8001 -o 2582250496 -b 32161792  
d3232 -p /dev/md/rdsk/d8000 -o 2962659328 -b 32161792  
d3233 -p /dev/md/rdsk/d8000 -o 2930496512 -b 32161792  
d3234 -p /dev/md/rdsk/d8001 -o 2550087680 -b 32161792  
d3235 -p /dev/md/rdsk/d8002 -o 2728368128 -b 32161792  
d3236 -p /dev/md/rdsk/d8003 -o 2714851328 -b 32161792  
d3237 -p /dev/md/rdsk/d8004 -o 2683721728 -b 32161792  
d3238 -p /dev/md/rdsk/d8004 -o 2651558912 -b 32161792  
d3239 -p /dev/md/rdsk/d8003 -o 2682688512 -b 32161792  
d3240 -p /dev/md/rdsk/d8002 -o 2696205312 -b 32161792  
d3241 -p /dev/md/rdsk/d8001 -o 2517924864 -b 32161792  
d3242 -p /dev/md/rdsk/d8000 -o 2898333696 -b 32161792  
d3243 -p /dev/md/rdsk/d8000 -o 2866170880 -b 32161792  
d3244 -p /dev/md/rdsk/d8001 -o 2485762048 -b 32161792  
d3245 -p /dev/md/rdsk/d8002 -o 2664042496 -b 32161792  
d3246 -p /dev/md/rdsk/d8003 -o 2650525696 -b 32161792  
d3247 -p /dev/md/rdsk/d8004 -o 2619396096 -b 32161792  
d3248 -p /dev/md/rdsk/d8004 -o 2587233280 -b 32161792  
d3249 -p /dev/md/rdsk/d8003 -o 2618362880 -b 32161792  
d3250 -p /dev/md/rdsk/d8002 -o 2631879680 -b 32161792  
d3251 -p /dev/md/rdsk/d8001 -o 2453599232 -b 32161792  
d3252 -p /dev/md/rdsk/d8000 -o 2834008064 -b 32161792  
d3253 -p /dev/md/rdsk/d8000 -o 2801845248 -b 32161792  
d3254 -p /dev/md/rdsk/d8001 -o 2421436416 -b 32161792  
d3255 -p /dev/md/rdsk/d8002 -o 2599716864 -b 32161792  
d3256 -p /dev/md/rdsk/d8003 -o 2586200064 -b 32161792  
d3257 -p /dev/md/rdsk/d8004 -o 2555070464 -b 32161792  
d3258 -p /dev/md/rdsk/d8004 -o 2522907648 -b 32161792  
d3259 -p /dev/md/rdsk/d8003 -o 2554037248 -b 32161792  
d3260 -p /dev/md/rdsk/d8002 -o 2567554048 -b 32161792  
d3261 -p /dev/md/rdsk/d8001 -o 2389273600 -b 32161792  
d3262 -p /dev/md/rdsk/d8000 -o 2769682432 -b 32161792  
d3263 -p /dev/md/rdsk/d8000 -o 2737519616 -b 32161792  
d3264 -p /dev/md/rdsk/d8001 -o 2357110784 -b 32161792  
d3265 -p /dev/md/rdsk/d8002 -o 2535391232 -b 32161792  
d3266 -p /dev/md/rdsk/d8003 -o 2521874432 -b 32161792  
d3267 -p /dev/md/rdsk/d8004 -o 2490744832 -b 32161792  
d3268 -p /dev/md/rdsk/d8004 -o 2458582016 -b 32161792  
d3269 -p /dev/md/rdsk/d8003 -o 2489711616 -b 32161792  
d3270 -p /dev/md/rdsk/d8002 -o 2503228416 -b 32161792  
d3271 -p /dev/md/rdsk/d8001 -o 2324947968 -b 32161792  
d3272 -p /dev/md/rdsk/d8000 -o 2705356800 -b 32161792  
d3273 -p /dev/md/rdsk/d8000 -o 2673193984 -b 32161792  
d3274 -p /dev/md/rdsk/d8001 -o 2292785152 -b 32161792  
d3275 -p /dev/md/rdsk/d8002 -o 2471065600 -b 32161792  
d3276 -p /dev/md/rdsk/d8003 -o 2457548800 -b 32161792  
d3277 -p /dev/md/rdsk/d8004 -o 2426419200 -b 32161792  
d3278 -p /dev/md/rdsk/d8004 -o 2394256384 -b 32161792  
d3279 -p /dev/md/rdsk/d8003 -o 2425385984 -b 32161792  
d3280 -p /dev/md/rdsk/d8002 -o 2438902784 -b 32161792  
d3281 -p /dev/md/rdsk/d8001 -o 2260622336 -b 32161792  
d3282 -p /dev/md/rdsk/d8000 -o 2461031168 -b 32161792  
d3283 -p /dev/md/rdsk/d8000 -o 2608868352 -b 32161792  
d3284 -p /dev/md/rdsk/d8001 -o 2228459520 -b 32161792  
d3200 -p /dev/md/rdsk/d8002 -o 2318061568 -b 120840192  
d3201 -p /dev/md/rdsk/d8003 -o 2304544768 -b 120840192  
d3202 -p /dev/md/rdsk/d8004 -o 2273415168 -b 120840192  
d3203 -p /dev/md/rdsk/d8004 -o 2152573952 -b 120840192  
d3204 -p /dev/md/rdsk/d8003 -o 2183703552 -b 120840192  
d3205 -p /dev/md/rdsk/d8002 -o 2197220352 -b 120840192  
d3206 -p /dev/md/rdsk/d8001 -o 2107618304 -b 120840192  
d3207 -p /dev/md/rdsk/d8000 -o 2488027136 -b 120840192  
d3208 -p /dev/md/rdsk/d8000 -o 2367185920 -b 120840192  
d3185 -p /dev/md/rdsk/d8001 -o 2100236288 -b 7380992  
d3186 -p /dev/md/rdsk/d8002 -o 2189838336 -b 7380992  
d3187 -p /dev/md/rdsk/d8003 -o 2176321536 -b 7380992  
d3188 -p /dev/md/rdsk/d8004 -o 2145191936 -b 7380992  
d3173 -p /dev/md/rdsk/d8004 -o 2130641920 -b 14548992  
d3174 -p /dev/md/rdsk/d8003 -o 2161771520 -b 14548992

d3175 -p /dev/md/rdsk/d8002 -o 2175288320 -b 14548992  
d3176 -p /dev/md/rdsk/d8001 -o 2085686272 -b 14548992  
d3177 -p /dev/md/rdsk/d8000 -o 2352635904 -b 14548992  
d3178 -p /dev/md/rdsk/d8000 -o 2338085888 -b 14548992  
d3179 -p /dev/md/rdsk/d8001 -o 2071136256 -b 14548992  
d3180 -p /dev/md/rdsk/d8002 -o 2160738304 -b 14548992  
d3165 -p /dev/md/rdsk/d8003 -o 2146607104 -b 15163392  
d3166 -p /dev/md/rdsk/d8004 -o 2115477504 -b 15163392  
d3167 -p /dev/md/rdsk/d8004 -o 2100313088 -b 15163392  
d3168 -p /dev/md/rdsk/d8003 -o 2131442688 -b 15163392  
d3157 -p /dev/md/rdsk/d8002 -o 2138201088 -b 22536192  
d3158 -p /dev/md/rdsk/d8001 -o 2048599040 -b 22536192  
d3159 -p /dev/md/rdsk/d8000 -o 2315548672 -b 22536192  
d3160 -p /dev/md/rdsk/d8000 -o 2293011456 -b 22536192  
d3161 -p /dev/md/rdsk/d8001 -o 2026061824 -b 22536192  
d3162 -p /dev/md/rdsk/d8002 -o 2115663872 -b 22536192  
d3163 -p /dev/md/rdsk/d8003 -o 2108905472 -b 22536192  
d3164 -p /dev/md/rdsk/d8004 -o 2077775872 -b 22536192  
d3101 -p /dev/md/rdsk/d8004 -o 2047046656 -b 30728192  
d3102 -p /dev/md/rdsk/d8003 -o 2078176256 -b 30728192  
d3103 -p /dev/md/rdsk/d8002 -o 2084934656 -b 30728192  
d3104 -p /dev/md/rdsk/d8001 -o 1995332608 -b 30728192  
d3105 -p /dev/md/rdsk/d8000 -o 2262282240 -b 30728192  
d3106 -p /dev/md/rdsk/d8000 -o 2231553024 -b 30728192  
d3107 -p /dev/md/rdsk/d8001 -o 1964603392 -b 30728192  
d3108 -p /dev/md/rdsk/d8002 -o 2054205440 -b 30728192  
d3109 -p /dev/md/rdsk/d8003 -o 2047447040 -b 30728192  
d3110 -p /dev/md/rdsk/d8004 -o 2016317440 -b 30728192  
d3111 -p /dev/md/rdsk/d8004 -o 1985588224 -b 30728192  
d3112 -p /dev/md/rdsk/d8003 -o 2016717824 -b 30728192  
d3113 -p /dev/md/rdsk/d8002 -o 2023476224 -b 30728192  
d3114 -p /dev/md/rdsk/d8001 -o 1933874176 -b 30728192  
d3115 -p /dev/md/rdsk/d8000 -o 2200823808 -b 30728192  
d3116 -p /dev/md/rdsk/d8000 -o 2170094592 -b 30728192  
d3117 -p /dev/md/rdsk/d8001 -o 1903144960 -b 30728192  
d3118 -p /dev/md/rdsk/d8002 -o 1992747008 -b 30728192  
d3119 -p /dev/md/rdsk/d8003 -o 1985988608 -b 30728192  
d3120 -p /dev/md/rdsk/d8004 -o 1954859008 -b 30728192  
d3121 -p /dev/md/rdsk/d8004 -o 1924129792 -b 30728192  
d3122 -p /dev/md/rdsk/d8003 -o 1955259392 -b 30728192  
d3123 -p /dev/md/rdsk/d8002 -o 1962017792 -b 30728192  
d3124 -p /dev/md/rdsk/d8001 -o 1872415744 -b 30728192  
d3125 -p /dev/md/rdsk/d8000 -o 2139365376 -b 30728192  
d3126 -p /dev/md/rdsk/d8000 -o 2108636160 -b 30728192  
d3127 -p /dev/md/rdsk/d8001 -o 1841686528 -b 30728192  
d3128 -p /dev/md/rdsk/d8002 -o 1931288576 -b 30728192  
d3129 -p /dev/md/rdsk/d8003 -o 1924530176 -b 30728192  
d3130 -p /dev/md/rdsk/d8004 -o 1893400576 -b 30728192  
d3131 -p /dev/md/rdsk/d8004 -o 1862671360 -b 30728192  
d3132 -p /dev/md/rdsk/d8003 -o 1893800960 -b 30728192  
d3133 -p /dev/md/rdsk/d8002 -o 1900559360 -b 30728192  
d3134 -p /dev/md/rdsk/d8001 -o 1810957312 -b 30728192  
d3135 -p /dev/md/rdsk/d8000 -o 2077906944 -b 30728192  
d3136 -p /dev/md/rdsk/d8000 -o 2047177248 -b 30728192  
d3137 -p /dev/md/rdsk/d8001 -o 1780228096 -b 30728192  
d3138 -p /dev/md/rdsk/d8002 -o 1869830144 -b 30728192  
d3139 -p /dev/md/rdsk/d8003 -o 1863071744 -b 30728192  
d3140 -p /dev/md/rdsk/d8004 -o 1831942144 -b 30728192  
d3141 -p /dev/md/rdsk/d8004 -o 1801212928 -b 30728192  
d3142 -p /dev/md/rdsk/d8003 -o 1832342528 -b 30728192  
d3143 -p /dev/md/rdsk/d8002 -o 1839100928 -b 30728192  
d3144 -p /dev/md/rdsk/d8001 -o 1749498880 -b 30728192  
d3145 -p /dev/md/rdsk/d8000 -o 2016448512 -b 30728192  
d3146 -p /dev/md/rdsk/d8000 -o 1985719296 -b 30728192  
d3147 -p /dev/md/rdsk/d8001 -o 1718769664 -b 30728192  
d3148 -p /dev/md/rdsk/d8002 -o 1808371712 -b 30728192  
d3149 -p /dev/md/rdsk/d8003 -o 1801613312 -b 30728192  
d3150 -p /dev/md/rdsk/d8004 -o 1770483712 -b 30728192  
d3151 -p /dev/md/rdsk/d8004 -o 1739754496 -b 30728192  
d3152 -p /dev/md/rdsk/d8003 -o 1770884096 -b 30728192  
d3153 -p /dev/md/rdsk/d8002 -o 1777642496 -b 30728192  
d3154 -p /dev/md/rdsk/d8001 -o 1688040448 -b 30728192  
d3155 -p /dev/md/rdsk/d8000 -o 1954990080 -b 30728192  
d3156 -p /dev/md/rdsk/d8000 -o 1924260864 -b 30728192  
d3169 -p /dev/md/rdsk/d8001 -o 1656287232 -b 31752192  
d3170 -p /dev/md/rdsk/d8002 -o 1745889280 -b 31752192  
d3171 -p /dev/md/rdsk/d8003 -o 1739130880 -b 31752192  
d3172 -p /dev/md/rdsk/d8004 -o 1708001280 -b 31752192  
d3181 -p /dev/md/rdsk/d8004 -o 1664984064 -b 43016192  
d3182 -p /dev/md/rdsk/d8003 -o 1696113664 -b 43016192  
d3183 -p /dev/md/rdsk/d8002 -o 1702872064 -b 43016192  
d3184 -p /dev/md/rdsk/d8001 -o 1613270016 -b 43016192  
d3189 -p /dev/md/rdsk/d8000 -o 1803419648 -b 120840192  
d3190 -p /dev/md/rdsk/d8000 -o 1682578432 -b 120840192  
d3191 -p /dev/md/rdsk/d8001 -o 1492428800 -b 120840192  
d3192 -p /dev/md/rdsk/d8002 -o 1582308848 -b 120840192  
d3193 -p /dev/md/rdsk/d8003 -o 1575272448 -b 120840192  
d3194 -p /dev/md/rdsk/d8004 -o 1544142848 -b 120840192  
d3195 -p /dev/md/rdsk/d8004 -o 1423301632 -b 120840192  
d3196 -p /dev/md/rdsk/d8003 -o 1454431232 -b 120840192  
d3197 -p /dev/md/rdsk/d8002 -o 1461189632 -b 120840192  
d3198 -p /dev/md/rdsk/d8001 -o 1371587584 -b 120840192  
d3199 -p /dev/md/rdsk/d8000 -o 1561737216 -b 120840192  
d2289 -p /dev/md/rdsk/d8001 -o 1371478016 -b 108544  
d2285 -p /dev/md/rdsk/d8002 -o 1460156416 -b 1032192  
d2286 -p /dev/md/rdsk/d8003 -o 1453398016 -b 1032192  
d2287 -p /dev/md/rdsk/d8004 -o 1422268416 -b 1032192  
d2288 -p /dev/md/rdsk/d8004 -o 1421235200 -b 1032192  
d2209 -p /dev/md/rdsk/d8003 -o 1421235200 -b 32161792  
d2210 -p /dev/md/rdsk/d8002 -o 1427993600 -b 32161792  
d2211 -p /dev/md/rdsk/d8001 -o 1339315200 -b 32161792  
d2212 -p /dev/md/rdsk/d8000 -o 1529574400 -b 32161792  
d2213 -p /dev/md/rdsk/d8000 -o 1497411584 -b 32161792  
d2214 -p /dev/md/rdsk/d8001 -o 1307152384 -b 32161792  
d2215 -p /dev/md/rdsk/d8002 -o 1395830784 -b 32161792  
d2216 -p /dev/md/rdsk/d8003 -o 1389072384 -b 32161792  
d2217 -p /dev/md/rdsk/d8004 -o 1389072384 -b 32161792  
d2218 -p /dev/md/rdsk/d8004 -o 1356909568 -b 32161792  
d2219 -p /dev/md/rdsk/d8003 -o 1356909568 -b 32161792  
d2220 -p /dev/md/rdsk/d8002 -o 1363667968 -b 32161792  
d2221 -p /dev/md/rdsk/d8001 -o 1274989568 -b 32161792  
d2222 -p /dev/md/rdsk/d8000 -o 1465248768 -b 32161792  
d2223 -p /dev/md/rdsk/d8000 -o 1433085952 -b 32161792  
d2224 -p /dev/md/rdsk/d8001 -o 1242826752 -b 32161792  
d2225 -p /dev/md/rdsk/d8002 -o 1331505152 -b 32161792  
d2226 -p /dev/md/rdsk/d8003 -o 1324746752 -b 32161792  
d2227 -p /dev/md/rdsk/d8004 -o 1324746752 -b 32161792  
d2228 -p /dev/md/rdsk/d8004 -o 1292583936 -b 32161792  
d2229 -p /dev/md/rdsk/d8003 -o 1292583936 -b 32161792  
d2230 -p /dev/md/rdsk/d8002 -o 1299342336 -b 32161792  
d2231 -p /dev/md/rdsk/d8001 -o 1210663936 -b 32161792  
d2232 -p /dev/md/rdsk/d8000 -o 1400923136 -b 32161792  
d2233 -p /dev/md/rdsk/d8000 -o 1368760320 -b 32161792  
d2234 -p /dev/md/rdsk/d8001 -o 1178501120 -b 32161792  
d2235 -p /dev/md/rdsk/d8002 -o 1267179520 -b 32161792  
d2236 -p /dev/md/rdsk/d8003 -o 1260421120 -b 32161792  
d2237 -p /dev/md/rdsk/d8004 -o 1260421120 -b 32161792  
d2238 -p /dev/md/rdsk/d8004 -o 1228258304 -b 32161792  
d2239 -p /dev/md/rdsk/d8003 -o 1228258304 -b 32161792  
d2240 -p /dev/md/rdsk/d8002 -o 1235016704 -b 32161792  
d2241 -p /dev/md/rdsk/d8001 -o 1146338304 -b 32161792  
d2242 -p /dev/md/rdsk/d8000 -o 1336597504 -b 32161792  
d2243 -p /dev/md/rdsk/d8000 -o 1304434688 -b 32161792  
d2244 -p /dev/md/rdsk/d8001 -o 1114175488 -b 32161792  
d2245 -p /dev/md/rdsk/d8002 -o 1202853888 -b 32161792  
d2246 -p /dev/md/rdsk/d8003 -o 1196095488 -b 32161792  
d2247 -p /dev/md/rdsk/d8004 -o 1196095488 -b 32161792  
d2248 -p /dev/md/rdsk/d8004 -o 1163932672 -b 32161792  
d2249 -p /dev/md/rdsk/d8003 -o 1163932672 -b 32161792  
d2250 -p /dev/md/rdsk/d8002 -o 1170691072 -b 32161792  
d2251 -p /dev/md/rdsk/d8001 -o 1082012672 -b 32161792  
d2252 -p /dev/md/rdsk/d8000 -o 1272271872 -b 32161792  
d2253 -p /dev/md/rdsk/d8000 -o 1240109056 -b 32161792  
d2254 -p /dev/md/rdsk/d8001 -o 1049849856 -b 32161792  
d2255 -p /dev/md/rdsk/d8002 -o 1138528256 -b 32161792  
d2256 -p /dev/md/rdsk/d8003 -o 1131769856 -b 32161792  
d2257 -p /dev/md/rdsk/d8004 -o 1131769856 -b 32161792  
d2258 -p /dev/md/rdsk/d8004 -o 1099607040 -b 32161792  
d2259 -p /dev/md/rdsk/d8003 -o 1099607040 -b 32161792  
d2260 -p /dev/md/rdsk/d8002 -o 1106365440 -b 32161792  
d2261 -p /dev/md/rdsk/d8001 -o 1017687040 -b 32161792  
d2262 -p /dev/md/rdsk/d8000 -o 1207946240 -b 32161792  
d2263 -p /dev/md/rdsk/d8000 -o 1157583424 -b 32161792  
d2264 -p /dev/md/rdsk/d8001 -o 985524224 -b 32161792  
d2265 -p /dev/md/rdsk/d8002 -o 1074202624 -b 32161792  
d2266 -p /dev/md/rdsk/d8003 -o 1067444224 -b 32161792  
d2267 -p /dev/md/rdsk/d8004 -o 1067444224 -b 32161792  
d2268 -p /dev/md/rdsk/d8004 -o 1035281408 -b 32161792  
d2269 -p /dev/md/rdsk/d8003 -o 1035281408 -b 32161792  
d2270 -p /dev/md/rdsk/d8002 -o 1042039808 -b 32161792  
d2271 -p /dev/md/rdsk/d8001 -o 953361408 -b 32161792  
d2272 -p /dev/md/rdsk/d8000 -o 1143620608 -b 32161792  
d2273 -p /dev/md/rdsk/d8000 -o 1111457792 -b 32161792  
d2274 -p /dev/md/rdsk/d8001 -o 921198592 -b 32161792  
d2275 -p /dev/md/rdsk/d8002 -o 1009876992 -b 32161792  
d2276 -p /dev/md/rdsk/d8003 -o 1003118592 -b 32161792  
d2277 -p /dev/md/rdsk/d8004 -o 1003118592 -b 32161792  
d2278 -p /dev/md/rdsk/d8004 -o 970955776 -b 32161792  
d2279 -p /dev/md/rdsk/d8003 -o 970955776 -b 32161792  
d2280 -p /dev/md/rdsk/d8002 -o 977714176 -b 32161792  
d2281 -p /dev/md/rdsk/d8001 -o 889035776 -b 32161792  
d2282 -p /dev/md/rdsk/d8000 -o 1079294976 -b 32161792  
d2283 -p /dev/md/rdsk/d8000 -o 1047132160 -b 32161792  
d2284 -p /dev/md/rdsk/d8001 -o 856872960 -b 32161792

```

d2200 -p /dev/md/rdsk/d8002 -o 856872960 -b 120840192
d2201 -p /dev/md/rdsk/d8003 -o 850114560 -b 120840192
d2202 -p /dev/md/rdsk/d8004 -o 850114560 -b 120840192
d2203 -p /dev/md/rdsk/d8004 -o 729273344 -b 120840192
d2204 -p /dev/md/rdsk/d8003 -o 729273344 -b 120840192
d2205 -p /dev/md/rdsk/d8002 -o 736031744 -b 120840192
d2206 -p /dev/md/rdsk/d8001 -o 736031744 -b 120840192
d2207 -p /dev/md/rdsk/d8000 -o 926290944 -b 120840192
d2208 -p /dev/md/rdsk/d8000 -o 805449728 -b 120840192
d2185 -p /dev/md/rdsk/d8001 -o 728649728 -b 7380992
d2186 -p /dev/md/rdsk/d8002 -o 728649728 -b 7380992
d2187 -p /dev/md/rdsk/d8003 -o 721891328 -b 7380992
d2188 -p /dev/md/rdsk/d8004 -o 721891328 -b 7380992
d2173 -p /dev/md/rdsk/d8004 -o 707341312 -b 14548992
d2174 -p /dev/md/rdsk/d8003 -o 707341312 -b 14548992
d2175 -p /dev/md/rdsk/d8002 -o 714099712 -b 14548992
d2176 -p /dev/md/rdsk/d8001 -o 714099712 -b 14548992
d2177 -p /dev/md/rdsk/d8000 -o 790899712 -b 14548992
d2178 -p /dev/md/rdsk/d8000 -o 776349696 -b 14548992
d2179 -p /dev/md/rdsk/d8001 -o 699549696 -b 14548992
d2180 -p /dev/md/rdsk/d8002 -o 699549696 -b 14548992
d2165 -p /dev/md/rdsk/d8003 -o 692176896 -b 15163392
d2166 -p /dev/md/rdsk/d8004 -o 692176896 -b 15163392
d2167 -p /dev/md/rdsk/d8004 -o 677012480 -b 15163392
d2168 -p /dev/md/rdsk/d8003 -o 677012480 -b 15163392
d2157 -p /dev/md/rdsk/d8002 -o 677012480 -b 22536192
d2158 -p /dev/md/rdsk/d8001 -o 677012480 -b 22536192
d2159 -p /dev/md/rdsk/d8000 -o 753812480 -b 22536192
d2160 -p /dev/md/rdsk/d8000 -o 731275264 -b 22536192
d2161 -p /dev/md/rdsk/d8001 -o 654475264 -b 22536192
d2162 -p /dev/md/rdsk/d8002 -o 654475264 -b 22536192
d2163 -p /dev/md/rdsk/d8003 -o 654475264 -b 22536192
d2164 -p /dev/md/rdsk/d8004 -o 654475264 -b 22536192
d2101 -p /dev/md/rdsk/d8004 -o 623746048 -b 30728192
d2102 -p /dev/md/rdsk/d8003 -o 623746048 -b 30728192
d2103 -p /dev/md/rdsk/d8002 -o 623746048 -b 30728192
d2104 -p /dev/md/rdsk/d8001 -o 623746048 -b 30728192
d2105 -p /dev/md/rdsk/d8000 -o 700546048 -b 30728192
d2106 -p /dev/md/rdsk/d8000 -o 669816832 -b 30728192
d2107 -p /dev/md/rdsk/d8001 -o 593016832 -b 30728192
d2108 -p /dev/md/rdsk/d8002 -o 593016832 -b 30728192
d2109 -p /dev/md/rdsk/d8003 -o 593016832 -b 30728192
d2110 -p /dev/md/rdsk/d8004 -o 593016832 -b 30728192
d2111 -p /dev/md/rdsk/d8004 -o 562287616 -b 30728192
d2112 -p /dev/md/rdsk/d8003 -o 562287616 -b 30728192
d2113 -p /dev/md/rdsk/d8002 -o 562287616 -b 30728192
d2114 -p /dev/md/rdsk/d8001 -o 562287616 -b 30728192
d2115 -p /dev/md/rdsk/d8000 -o 639087616 -b 30728192
d2116 -p /dev/md/rdsk/d8000 -o 608358400 -b 30728192
d2117 -p /dev/md/rdsk/d8001 -o 531558400 -b 30728192
d2118 -p /dev/md/rdsk/d8002 -o 531558400 -b 30728192
d2119 -p /dev/md/rdsk/d8003 -o 531558400 -b 30728192
d2120 -p /dev/md/rdsk/d8004 -o 531558400 -b 30728192
d2121 -p /dev/md/rdsk/d8004 -o 500829184 -b 30728192
d2122 -p /dev/md/rdsk/d8003 -o 500829184 -b 30728192
d2123 -p /dev/md/rdsk/d8002 -o 500829184 -b 30728192
d2124 -p /dev/md/rdsk/d8001 -o 500829184 -b 30728192
d2125 -p /dev/md/rdsk/d8000 -o 577629184 -b 30728192

```

```

d2126 -p /dev/md/rdsk/d8000 -o 546899968 -b 30728192
d2127 -p /dev/md/rdsk/d8001 -o 470099968 -b 30728192
d2128 -p /dev/md/rdsk/d8002 -o 470099968 -b 30728192
d2129 -p /dev/md/rdsk/d8003 -o 470099968 -b 30728192
d2130 -p /dev/md/rdsk/d8004 -o 470099968 -b 30728192
d2131 -p /dev/md/rdsk/d8004 -o 439370752 -b 30728192
d2132 -p /dev/md/rdsk/d8003 -o 439370752 -b 30728192
d2133 -p /dev/md/rdsk/d8002 -o 439370752 -b 30728192
d2134 -p /dev/md/rdsk/d8001 -o 439370752 -b 30728192
d2135 -p /dev/md/rdsk/d8000 -o 516170752 -b 30728192
d2136 -p /dev/md/rdsk/d8000 -o 485441536 -b 30728192
d2137 -p /dev/md/rdsk/d8001 -o 408641536 -b 30728192
d2138 -p /dev/md/rdsk/d8002 -o 408641536 -b 30728192
d2139 -p /dev/md/rdsk/d8003 -o 408641536 -b 30728192
d2140 -p /dev/md/rdsk/d8004 -o 408641536 -b 30728192
d2141 -p /dev/md/rdsk/d8004 -o 377912320 -b 30728192
d2142 -p /dev/md/rdsk/d8003 -o 377912320 -b 30728192
d2143 -p /dev/md/rdsk/d8002 -o 377912320 -b 30728192
d2144 -p /dev/md/rdsk/d8001 -o 377912320 -b 30728192
d2145 -p /dev/md/rdsk/d8000 -o 454712320 -b 30728192
d2146 -p /dev/md/rdsk/d8000 -o 423983104 -b 30728192
d2147 -p /dev/md/rdsk/d8001 -o 347183104 -b 30728192
d2148 -p /dev/md/rdsk/d8002 -o 347183104 -b 30728192
d2149 -p /dev/md/rdsk/d8003 -o 347183104 -b 30728192
d2150 -p /dev/md/rdsk/d8004 -o 347183104 -b 30728192
d2151 -p /dev/md/rdsk/d8004 -o 316453888 -b 30728192
d2152 -p /dev/md/rdsk/d8003 -o 316453888 -b 30728192
d2153 -p /dev/md/rdsk/d8002 -o 316453888 -b 30728192
d2154 -p /dev/md/rdsk/d8001 -o 316453888 -b 30728192
d2155 -p /dev/md/rdsk/d8000 -o 393253888 -b 30728192
d2156 -p /dev/md/rdsk/d8000 -o 362524672 -b 30728192
d2169 -p /dev/md/rdsk/d8001 -o 284700672 -b 31752192
d2170 -p /dev/md/rdsk/d8002 -o 284700672 -b 31752192
d2171 -p /dev/md/rdsk/d8003 -o 284700672 -b 31752192
d2172 -p /dev/md/rdsk/d8004 -o 284700672 -b 31752192
d2181 -p /dev/md/rdsk/d8004 -o 241683456 -b 43016192
d2182 -p /dev/md/rdsk/d8003 -o 241683456 -b 43016192
d2183 -p /dev/md/rdsk/d8002 -o 241683456 -b 43016192
d2184 -p /dev/md/rdsk/d8001 -o 241683456 -b 43016192
d2189 -p /dev/md/rdsk/d8000 -o 241683456 -b 120840192
d2190 -p /dev/md/rdsk/d8000 -o 120842240 -b 120840192
d2191 -p /dev/md/rdsk/d8001 -o 120842240 -b 120840192
d2192 -p /dev/md/rdsk/d8002 -o 120842240 -b 120840192
d2193 -p /dev/md/rdsk/d8003 -o 120842240 -b 120840192
d2194 -p /dev/md/rdsk/d8004 -o 120842240 -b 120840192
d2195 -p /dev/md/rdsk/d8004 -o 1024 -b 120840192
d2196 -p /dev/md/rdsk/d8003 -o 1024 -b 120840192
d2197 -p /dev/md/rdsk/d8002 -o 1024 -b 120840192
d2198 -p /dev/md/rdsk/d8001 -o 1024 -b 120840192
d2199 -p /dev/md/rdsk/d8000 -o 1024 -b 120840192

```

### COMSTAR REDO Heads

Measured Comstar REDO head (X4275)

#### uname

SunOS dcredo1 5.11 snv\_151a i86pc i386 i86pc

#### prtdiag

```

.....
System Configuration: SUN MICROSYSTEMS SUN FIRE X4275 SERVER
BIOS Configuration: American Megatrends Inc. 07060223 03/04/2010
BMC Configuration: IPMI 1.5 (KCS: Keyboard Controller Style)

```

#### ==== Processor Sockets =====

Version	Location Tag
Intel(R) Xeon(R) CPU	E5540 @ 2.53GHz CPU 1

#### ==== Memory Device Sockets =====

Type	Status	Set	Device	Locator	Bank	Locator
Unknown	in use	0	D2		BANK2	
Unknown	empty	0	D1		BANK1	
Unknown	empty	0	D0		BANK0	
Unknown	in use	0	D5		BANK5	
Unknown	empty	0	D4		BANK4	
Unknown	empty	0	D3		BANK3	
Unknown	in use	0	D8		BANK8	
Unknown	empty	0	D7		BANK7	
Unknown	empty	0	D6		BANK6	
Unknown	empty	0	D2		BANK2	
Unknown	empty	0	D1		BANK1	
Unknown	empty	0	D0		BANK0	
Unknown	empty	0	D5		BANK5	
Unknown	empty	0	D4		BANK4	
Unknown	empty	0	D3		BANK3	
Unknown	empty	0	D8		BANK8	
Unknown	empty	0	D7		BANK7	
Unknown	empty	0	D6		BANK6	
FLASH	in use	0				

#### ==== On-Board Devices =====

Zoar 2x GbE.  
Zoar 2x GbE.  
Zoar 2x GbE.  
Zoar 2x GbE.

#### ==== Upgradeable Slots =====

ID	Status	Type	Description
0	in use	PCI Express	PCIE0
1	available	PCI Express	PCIE1
2	available	PCI Express	PCIE2
3	in use	PCI Express	PCIE3
4	available	PCI Express	PCIE4
5	available	PCI Express	PCIE5

prtconf

System Configuration: Oracle Corporation i86pc
Memory size: 6136 Megabytes
System Peripherals (Software Nodes):

i86pc

scsi\_vhci, instance #0
pci, instance #0
pci108e,4845 (driver not attached)
pci8086,3408, instance #0
pci108e,0, instance #0
pci108e,0, instance #1 (driver not attached)
pci8086,3409, instance #1
pci108e,0, instance #2 (driver not attached)
pci108e,0, instance #3 (driver not attached)
pci8086,340a, instance #2
pci1077,171, instance #0
pci1077,171, instance #1
pci8086,340c, instance #3
pci1000,9263, instance #0
sd, instance #0
sd, instance #2
pci8086,340e, instance #4
pci111d,806e, instance #6
pci111d,806e, instance #7
pci111d,806e, instance #8
pci8086,3410, instance #5
pci111d,806e, instance #9
pci111d,806e, instance #10
pci111d,806e, instance #11
pci8086,342d (driver not attached)
pci8086,342e (driver not attached)
pci8086,3422 (driver not attached)
pci8086,3423, instance #0
pci8086,3438 (driver not attached)
pci108e,4845 (driver not attached)
pci108e,4845 (driver not attached)
pci108e,4845 (driver not attached)
pci108e,4845 (driver not attached)
pci108e,4845 (driver not attached)
pci108e,4845 (driver not attached)
pci108e,4845 (driver not attached)
pci108e,4845 (driver not attached)
pci108e,4845, instance #0
pci108e,4845, instance #1
pci108e,4845, instance #2
device, instance #0
keyboard, instance #0
mouse, instance #1
pci108e,4845, instance #0
hub, instance #0
pci108e,4845, instance #3
pci108e,4845, instance #4
pci108e,4845, instance #5
pci108e,4845, instance #1

storage, instance #0
disk, instance #1
pci8086,244e, instance #0
display, instance #0
isa, instance #0
motherboard (driver not attached)
asy, instance #0
motherboard (driver not attached)
pit\_bEEP, instance #0
pci108e,4845, instance #0
pci108e,4845 (driver not attached)
ioapics (driver not attached)
ioapic, instance #0 (driver not attached)
fw, instance #0
cpu, instance #0
cpu, instance #1
cpu, instance #2
cpu, instance #3
cpu, instance #4
cpu, instance #5
cpu, instance #6
cpu, instance #7
sb, instance #1
used-resources (driver not attached)
iscsi, instance #0
options, instance #0
pseudo, instance #0
agpgart, instance #0 (driver not attached)
xsvc, instance #0

format

Searching for disks..done

AVAILABLE DISK SELECTIONS:

- 0. c7t0d0 <LSI cyl 60754 alt 2 hd 255 sec 252>
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@0,0
1. c7t1d0 <LSI-MR9261-8i-2.50-18.18TB>
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@1,0
2. c8t0d0 <TUSB625-USB20 DISK DRIV- cyl 1946 alt 2 hd 255 sec 63>
/pci@0,0/pci108e,4845@1d,7/storage@1/disk@0,0

Specify disk (enter its number): 1

selecting c7t1d0

[disk formatted]

/dev/dsk/c7t1d0s4 is part of SVM volume stripe:d120. Please see metaclear(1M).

/dev/dsk/c7t1d0s5 is part of SVM volume stripe:d130. Please see metaclear(1M).

/dev/dsk/c7t1d0s6 contains an SVM mdb. Please see metadb(1M).

FORMAT MENU:

- disk - select a disk
type - select (define) a disk type
partition - select (define) a partition table

- current - describe the current disk
format - format and analyze the disk
fdisk - run the fdisk program
repair - repair a defective sector
label - write label to the disk
analyze - surface analysis
defect - defect list management
backup - search for backup labels
verify - read and display labels
inquiry - show vendor, product and revision
volname - set 8-character volume name
!<cmd> - execute <cmd>, then return

quit
format> p

PARTITION MENU:

- 0 - change `0' partition
1 - change `1' partition
2 - change `2' partition
3 - change `3' partition
4 - change `4' partition
5 - change `5' partition
6 - change `6' partition
expand - expand label to use whole disk
select - select a predefined table
modify - modify a predefined partition table
name - name the current table
print - display the current table
label - write partition map and label to the disk
!<cmd> - execute <cmd>, then return

quit
partition> p

Current partition table (original):

Total disk sectors available: 39042932669 + 16384 (reserved sectors)

Table with 6 columns: Part, Tag, Flag, First Sector, Size, Last Sector. It lists disk partitions 0 through 8, including reserved space.

/etc/vfstab

Table with 7 columns: #to mount, #to fsck, device, mount point, FS type, fsck pass, mount at boot, mount options. It lists various system mounts like /devices, /proc, ctf, objfs, and sharefs.



```

fd - /dev/fd fd - no -
swap - /tmp tmpfs - yes -

/dev/zvol/dsk/rpool/swap - - swap - no -

/etc/system
-----
Oracle Solaris 11 Express snv_151a X86
Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
Assembled 04 November 2010
-----
*ident "%Z%%M% %I% %E% SMI" /* SVR4 1.5 */
*
* CDDL HEADER START
*
* The contents of this file are subject to the terms of the
* Common Development and Distribution License, Version 1.0 only
* (the "License"). You may not use this file except in compliance
* with the License.
*
* You can obtain a copy of the license at usr/src/OPENSOLARIS.LICENSE
* or http://www.opensolaris.org/os/licensing.
* See the License for the specific language governing permissions
* and limitations under the License.
*
* When distributing Covered Code, include this CDDL HEADER in each
* file and include the License file at usr/src/OPENSOLARIS.LICENSE.
* If applicable, add the following below this CDDL HEADER, with the
* fields enclosed by brackets "[]" replaced with your own identifying
* information: Portions Copyright [yyyy] [name of copyright owner]
*
* CDDL HEADER END
*
* SYSTEM SPECIFICATION FILE
*
* moddir:
*
* Set the search path for modules. This has a format similar to the
* csh path variable. If the module isn't found in the first directory
* it tries the second and so on. The default is /kernel /usr/kernel
*
* Example:
*
* moddir: /kernel /usr/kernel /other/modules
*
* root device and root filesystem configuration:
*
* The following may be used to override the defaults provided by
* the boot program:
*
* rootfs: Set the filesystem type of the root.
*
* rootdev: Set the root device. This should be a fully
* expanded physical pathname. The default is
the
*
* physical pathname of the device where the
boot
*
* program resides. The physical pathname is
highly platform and configuration
dependent.
*
* Example:
*
* rootfs:ufs
*
* rootdev:/sbus@1,f8000000/esp@0,8000000/sd@3,0:a
*
* (Swap device configuration should be specified in /etc/vfstab.)
*
* exclude:
*
* Modules appearing in the moddir path which are NOT to be
loaded,
*
* even if referenced. Note that 'exclude' accepts either a module
name,
*
* or a filename which includes the directory.
*
* Examples:
*
* exclude: win
*
* exclude: sys/shmsys
*
* forceload:
*
* Cause these modules to be loaded at boot time, (just before
mounting
*
* the root filesystem) rather than at first reference. Note that
forceload expects a filename which includes the directory. Also
note that loading a module does not necessarily imply that it will
be installed.
*
* Example:
*
* forceload: drv/foo
*
* set:
*
* Set an integer variable in the kernel or a module to a new value.
This facility should be used with caution. See system(4).
*
* Examples:
*
* To set variables in 'unix':
*
* set nautopush=32
*
* set maxusers=40
*
* To set a variable named 'debug' in the module named 'test_module'
set test_module:debug = 0x13
set autoup=300
* As per Paul's recommend:
set maxphys=262144
-----
mpt.conf
-----
#
# Copyright 2008 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
#
# The mpt driver, as a pHCI driver, must specify the vHCI class it
# belongs to(scsi_vhci).
#
# ddi-vhci-class="scsi_vhci";
#
# I/O multipathing feature (MPxIO) can be enabled or disabled using
# mpzio-disable property. Setting mpzio-disable="no" will activate
# I/O multipathing; setting mpzio-disable="yes" disables the feature.
#
# Global mpzio-disable property:
#
# To globally enable MPxIO on all mpt controllers set:
# mpzio-disable="no";
#
# To globally disable MPxIO on all mpt controllers set:
# mpzio-disable="yes";
#
# You can also enable or disable MPxIO on a per HBA basis.
# Per HBA settings override the global setting for the specified HBAs.
# To disable MPxIO on a controller whose parent is /pci@7c0/pci@0/pci@9
# and the unit-address is "0" set:
# name="mpt" parent="/pci@7c0/pci@0/pci@9" unit-address="0" mpzio-
disable="yes";
#
mpzio-disable="yes";
#
# SATA mpzio supported
#
# To disable SATA mpzio, set
# disable-sata-mpzio="yes";
# When mpzio-disable="yes" is set, the disable-sata-mpzio property
# takes no effect
#
disable-sata-mpzio="no";
-----
system:0:::
user.root:1:::
noproject:2:::
default:3:::
group.staff:10:::
-----
Fri Dec 2 10:33:36 PST 2011
end sysostune

```

<b>niwot.conf</b>	600144f0054c0f0000000000a01d210	13045268480	600144f0054c0f0000000000a01d239	13045268480
	/dev/md/rdsk/d210		/dev/md/rdsk/d239	
	600144f0054c0f0000000000a01d211	13045268480	600144f0054c0f0000000000a01d240	13045268480
	/dev/md/rdsk/d211		/dev/md/rdsk/d240	
	600144f0054c0f0000000000a01d212	13045268480	600144f0054c0f0000000000a01d241	13045268480
	/dev/md/rdsk/d212		/dev/md/rdsk/d241	
	600144f0054c0f0000000000a01d213	13045268480	600144f0054c0f0000000000a01d242	13045268480
	/dev/md/rdsk/d213		/dev/md/rdsk/d242	
	600144f0054c0f0000000000a01d214	13045268480	600144f0054c0f0000000000a01d243	13045268480
	/dev/md/rdsk/d214		/dev/md/rdsk/d243	
	600144f0054c0f0000000000a01d215	13045268480	600144f0054c0f0000000000a01d244	13045268480
	/dev/md/rdsk/d215		/dev/md/rdsk/d244	
	600144f0054c0f0000000000a01d216	13045268480	600144f0054c0f0000000000a01d245	13045268480
	/dev/md/rdsk/d216		/dev/md/rdsk/d245	
	600144f0054c0f0000000000a01d217	13045268480	600144f0054c0f0000000000a01d246	13045268480
	/dev/md/rdsk/d217		/dev/md/rdsk/d246	
	600144f0054c0f0000000000a01d218	13045268480	600144f0054c0f0000000000a01d247	13045268480
	/dev/md/rdsk/d218		/dev/md/rdsk/d247	
	600144f0054c0f0000000000a01d219	13045268480	600144f0054c0f0000000000a01d248	13045268480
	/dev/md/rdsk/d219		/dev/md/rdsk/d248	
600144f0054c0f0000000000a01d220	13045268480	600144f0054c0f0000000000a01d249	13045268480	
/dev/md/rdsk/d220		/dev/md/rdsk/d249		
600144f0054c0f0000000000a01d221	13045268480	600144f0054c0f0000000000a01d250	13045268480	
/dev/md/rdsk/d221		/dev/md/rdsk/d250		
600144f0054c0f0000000000a01d222	13045268480	600144f0054c0f0000000000a01d251	13045268480	
/dev/md/rdsk/d222		/dev/md/rdsk/d251		
600144f0054c0f0000000000a01d223	13045268480	600144f0054c0f0000000000a01d252	13045268480	
/dev/md/rdsk/d223		/dev/md/rdsk/d252		
600144f0054c0f0000000000a01d224	13045268480	600144f0054c0f0000000000a01d253	13045268480	
/dev/md/rdsk/d224		/dev/md/rdsk/d253		
600144f0054c0f0000000000a01d225	13045268480	600144f0054c0f0000000000a01d254	13045268480	
/dev/md/rdsk/d225		/dev/md/rdsk/d254		
600144f0054c0f0000000000a01d226	13045268480	600144f0054c0f0000000000a01d255	13045268480	
/dev/md/rdsk/d226		/dev/md/rdsk/d255		
600144f0054c0f0000000000a01d227	13045268480	600144f0054c0f0000000000a01d256	13045268480	
/dev/md/rdsk/d227		/dev/md/rdsk/d256		
600144f0054c0f0000000000a01d228	13045268480	600144f0054c0f0000000000a01d257	13045268480	
/dev/md/rdsk/d228		/dev/md/rdsk/d257		
600144f0054c0f0000000000a01d229	13045268480	600144f0054c0f0000000000a01d258	13045268480	
/dev/md/rdsk/d229		/dev/md/rdsk/d258		
600144f0054c0f0000000000a01d230	13045268480	600144f0054c0f0000000000a01d259	13045268480	
/dev/md/rdsk/d230		/dev/md/rdsk/d259		
600144f0054c0f0000000000a01d231	13045268480	600144f0054c0f0000000000a01d260	13045268480	
/dev/md/rdsk/d231		/dev/md/rdsk/d260		
600144f0054c0f0000000000a01d232	13045268480	600144f0054c0f0000000000a01d261	13045268480	
/dev/md/rdsk/d232		/dev/md/rdsk/d261		
600144f0054c0f0000000000a01d233	13045268480	600144f0054c0f0000000000a01d262	13045268480	
/dev/md/rdsk/d233		/dev/md/rdsk/d262		
600144f0054c0f0000000000a01d234	13045268480	600144f0054c0f0000000000a01d263	13045268480	
/dev/md/rdsk/d234		/dev/md/rdsk/d263		
600144f0054c0f0000000000a01d235	13045268480	600144f0054c0f0000000000a01d264	13045268480	
/dev/md/rdsk/d235		/dev/md/rdsk/d264		
600144f0054c0f0000000000a01d236	13045268480	600144f0054c0f0000000000a01d265	13045268480	
/dev/md/rdsk/d236		/dev/md/rdsk/d265		
600144f0054c0f0000000000a01d237	13045268480	600144f0054c0f0000000000a01d266	13045268480	
/dev/md/rdsk/d237		/dev/md/rdsk/d266		
600144f0054c0f0000000000a01d238	13045268480	600144f0054c0f0000000000a01d267	13045268480	
/dev/md/rdsk/d238		/dev/md/rdsk/d267		

Adapter 0 -- Virtual Drive Information:  
Virtual Disk: 1 (Target Id: 1)  
Name:  
RAID Level: Primary-0, Secondary-0, RAID Level Qualifier-0  
Size:18.180 TB  
State: Optimal  
Stripe Size: 256 KB  
Number Of Drives:10  
Span Depth:1  
Default Cache Policy: WriteBack, ReadAheadNone, Cached, Write Cache  
OK if Bad BBU  
Current Cache Policy: WriteBack, ReadAheadNone, Cached, Write Cache  
OK if Bad BBU  
Access Policy: Read/Write  
Disk Cache Policy: Disabled  
Encryption Type: None

Exit Code: 0x00  
Fri Dec 2 10:33:28 PST 2011  
List all Software Volumes and COMSTAR LUNS  
.....  
This lists the attached storage (format) for DATA COMSTAR unit ...  
Fri Dec 2 10:33:28 PST 2011  
.....

**sbadm**

Found 84 LU(s)

GUID	DATA SIZE	SOURCE
600144f0054c0f0000000000a01d200	13045268480	/dev/md/rdsk/d200
600144f0054c0f0000000000a01d201	13045268480	/dev/md/rdsk/d201
600144f0054c0f0000000000a01d202	13045268480	/dev/md/rdsk/d202
600144f0054c0f0000000000a01d203	13045268480	/dev/md/rdsk/d203
600144f0054c0f0000000000a01d204	13045268480	/dev/md/rdsk/d204
600144f0054c0f0000000000a01d205	13045268480	/dev/md/rdsk/d205
600144f0054c0f0000000000a01d206	13045268480	/dev/md/rdsk/d206
600144f0054c0f0000000000a01d207	13045268480	/dev/md/rdsk/d207
600144f0054c0f0000000000a01d208	13045268480	/dev/md/rdsk/d208
600144f0054c0f0000000000a01d209	13045268480	/dev/md/rdsk/d209

```

600144f0054c0f00000000000a01d268 13045268480
/dev/md/rdsk/d268
600144f0054c0f00000000000a01d269 13045268480
/dev/md/rdsk/d269
600144f0054c0f00000000000a01d270 13045268480
/dev/md/rdsk/d270
600144f0054c0f00000000000a01d271 13045268480
/dev/md/rdsk/d271
600144f0054c0f00000000000a01d272 13045268480
/dev/md/rdsk/d272
600144f0054c0f00000000000a01d273 13045268480
/dev/md/rdsk/d273
600144f0054c0f00000000000a01d274 13045268480
/dev/md/rdsk/d274
600144f0054c0f00000000000a01d275 13045268480
/dev/md/rdsk/d275
600144f0054c0f00000000000a01d276 13045268480
/dev/md/rdsk/d276
600144f0054c0f00000000000a01d277 13045268480
/dev/md/rdsk/d277
600144f0054c0f00000000000a01d278 13045268480
/dev/md/rdsk/d278
600144f0054c0f00000000000a01d279 13045268480
/dev/md/rdsk/d279
600144f0a3770e00000000000a01d300 751619211264
/dev/rdsk/c7t1d0s0
600144f0a3770e00000000000a01d303 751619211264
/dev/rdsk/c7t1d0s1
600144f0a3770e00000000000a01d400 107374116864
/dev/rdsk/c7t1d0s2
600144f0a3770e00000000000a01d403 107374116864
/dev/rdsk/c7t1d0s3
.....
metastat

d7503 -p /dev/md/rdsk/d130 -o 5957666816 -b 1468006000
d130 1 1 /dev/rdsk/c7t1d0s5
d7502 -p /dev/md/rdsk/d130 -o 4489660416 -b 1468006000
d7501 -p /dev/md/rdsk/d130 -o 3021654016 -b 1468006000
d7500 -p /dev/md/rdsk/d130 -o 1553647616 -b 1468006000
d7403 -p /dev/md/rdsk/d120 -o 6442435584 -b 1468006000
d120 1 1 /dev/rdsk/c7t1d0s4
d7402 -p /dev/md/rdsk/d120 -o 4974429184 -b 1468006000
d7401 -p /dev/md/rdsk/d120 -o 3506422784 -b 1468006000
d7400 -p /dev/md/rdsk/d120 -o 2038416384 -b 1468006000
d279 -p /dev/md/rdsk/d120 -o 2012936192 -b 25479168
d278 -p /dev/md/rdsk/d120 -o 1987456000 -b 25479168
d277 -p /dev/md/rdsk/d120 -o 1961975808 -b 25479168
d276 -p /dev/md/rdsk/d120 -o 1936495616 -b 25479168
d275 -p /dev/md/rdsk/d120 -o 1911015424 -b 25479168
d274 -p /dev/md/rdsk/d120 -o 1885535232 -b 25479168
d273 -p /dev/md/rdsk/d120 -o 1860055040 -b 25479168
d272 -p /dev/md/rdsk/d120 -o 1834574848 -b 25479168
d271 -p /dev/md/rdsk/d120 -o 1809094656 -b 25479168
d270 -p /dev/md/rdsk/d120 -o 1783614464 -b 25479168
d269 -p /dev/md/rdsk/d120 -o 1758134272 -b 25479168

```

```

d268 -p /dev/md/rdsk/d120 -o 1732654080 -b 25479168
d267 -p /dev/md/rdsk/d120 -o 1707173888 -b 25479168
d266 -p /dev/md/rdsk/d120 -o 1681693696 -b 25479168
d265 -p /dev/md/rdsk/d120 -o 1656213504 -b 25479168
d264 -p /dev/md/rdsk/d120 -o 1630733312 -b 25479168
d263 -p /dev/md/rdsk/d120 -o 1605253120 -b 25479168
d262 -p /dev/md/rdsk/d120 -o 1579772928 -b 25479168
d261 -p /dev/md/rdsk/d120 -o 1554292736 -b 25479168
d260 -p /dev/md/rdsk/d120 -o 1528812544 -b 25479168
d259 -p /dev/md/rdsk/d120 -o 1503332352 -b 25479168
d258 -p /dev/md/rdsk/d120 -o 1477852160 -b 25479168
d257 -p /dev/md/rdsk/d120 -o 1452371968 -b 25479168
d256 -p /dev/md/rdsk/d120 -o 1426891776 -b 25479168
d255 -p /dev/md/rdsk/d120 -o 1401411584 -b 25479168
d254 -p /dev/md/rdsk/d120 -o 1375931392 -b 25479168
d253 -p /dev/md/rdsk/d120 -o 1350451200 -b 25479168
d252 -p /dev/md/rdsk/d120 -o 1324971008 -b 25479168
d251 -p /dev/md/rdsk/d120 -o 1299490816 -b 25479168
d250 -p /dev/md/rdsk/d120 -o 1274010624 -b 25479168
d249 -p /dev/md/rdsk/d120 -o 1248530432 -b 25479168
d248 -p /dev/md/rdsk/d120 -o 1223050240 -b 25479168
d247 -p /dev/md/rdsk/d120 -o 1197570048 -b 25479168
d246 -p /dev/md/rdsk/d120 -o 1172089856 -b 25479168
d245 -p /dev/md/rdsk/d120 -o 1146609664 -b 25479168
d244 -p /dev/md/rdsk/d120 -o 1121129472 -b 25479168
d243 -p /dev/md/rdsk/d120 -o 1095649280 -b 25479168
d242 -p /dev/md/rdsk/d120 -o 1070169088 -b 25479168
d241 -p /dev/md/rdsk/d120 -o 1044688896 -b 25479168
d240 -p /dev/md/rdsk/d120 -o 1019208704 -b 25479168
d239 -p /dev/md/rdsk/d120 -o 993728512 -b 25479168
d238 -p /dev/md/rdsk/d120 -o 968248320 -b 25479168
d237 -p /dev/md/rdsk/d120 -o 942768128 -b 25479168
d236 -p /dev/md/rdsk/d120 -o 917287936 -b 25479168
d235 -p /dev/md/rdsk/d120 -o 891807744 -b 25479168
d234 -p /dev/md/rdsk/d120 -o 866327552 -b 25479168
d233 -p /dev/md/rdsk/d120 -o 840847360 -b 25479168
d232 -p /dev/md/rdsk/d120 -o 815367168 -b 25479168
d231 -p /dev/md/rdsk/d120 -o 789886976 -b 25479168
d230 -p /dev/md/rdsk/d120 -o 764406784 -b 25479168
d229 -p /dev/md/rdsk/d120 -o 738926592 -b 25479168
d228 -p /dev/md/rdsk/d120 -o 713446400 -b 25479168
d227 -p /dev/md/rdsk/d120 -o 687966208 -b 25479168
d226 -p /dev/md/rdsk/d120 -o 662486016 -b 25479168
d225 -p /dev/md/rdsk/d120 -o 637005824 -b 25479168
d224 -p /dev/md/rdsk/d120 -o 611525632 -b 25479168
d223 -p /dev/md/rdsk/d120 -o 586045440 -b 25479168
d222 -p /dev/md/rdsk/d120 -o 560565248 -b 25479168
d221 -p /dev/md/rdsk/d120 -o 535085056 -b 25479168
d220 -p /dev/md/rdsk/d120 -o 509604864 -b 25479168
d219 -p /dev/md/rdsk/d120 -o 484124672 -b 25479168
d218 -p /dev/md/rdsk/d120 -o 458644480 -b 25479168
d217 -p /dev/md/rdsk/d120 -o 433164288 -b 25479168
d216 -p /dev/md/rdsk/d120 -o 407684096 -b 25479168
d215 -p /dev/md/rdsk/d120 -o 382203904 -b 25479168
d214 -p /dev/md/rdsk/d120 -o 356723712 -b 25479168
d213 -p /dev/md/rdsk/d120 -o 331243520 -b 25479168
d212 -p /dev/md/rdsk/d120 -o 305763328 -b 25479168
d211 -p /dev/md/rdsk/d120 -o 280283136 -b 25479168

```

```

d210 -p /dev/md/rdsk/d120 -o 254802944 -b 25479168
d209 -p /dev/md/rdsk/d120 -o 229322752 -b 25479168
d208 -p /dev/md/rdsk/d120 -o 203842560 -b 25479168
d207 -p /dev/md/rdsk/d120 -o 178362368 -b 25479168
d206 -p /dev/md/rdsk/d120 -o 152882176 -b 25479168
d205 -p /dev/md/rdsk/d120 -o 127401984 -b 25479168
d204 -p /dev/md/rdsk/d120 -o 101921792 -b 25479168
d203 -p /dev/md/rdsk/d120 -o 76441600 -b 25479168
d202 -p /dev/md/rdsk/d120 -o 50961408 -b 25479168
d201 -p /dev/md/rdsk/d120 -o 25481216 -b 25479168
d200 -p /dev/md/rdsk/d120 -o 1024 -b 25479168
d7301 -p /dev/md/rdsk/d130 -o 209715200 -b 209714000
d7300 -p /dev/md/rdsk/d130 -o 1024 -b 209714000
d7302 -p /dev/md/rdsk/d130 -o 1343933440 -b 209714000
Fri Dec 2 10:33:28 PST 2011
end syvols_luns

```

**Priced and Measured COMSTAR REDO Head (X4270 M2)**

**uname**

SunOS x4270-028 5.11 11.0 i86pc i386 i86pc

**prtdiag**

System Configuration: Oracle Corporation SUN FIRE X4270 M2 SERVER  
 BIOS Configuration: American Megatrends Inc. 08080102 05/23/2011  
 BMC Configuration: IPMI 1.5 (KCS: Keyboard Controller Style)

==== Processor Sockets =====

Version	Location Tag
Intel(R) Xeon(R) CPU	X5690 @ 3.47GHz CPU 0

==== Memory Device Sockets =====

Type	Status	Set	Device	Locator	Bank	Locator
Unknown	in use	0	D2		/SYS/MB/	P0
Unknown	empty	0	D1		/SYS/MB/	P0
Unknown	empty	0	D0		/SYS/MB/	P0
Unknown	in use	0	D5		/SYS/MB/	P0
Unknown	empty	0	D4		/SYS/MB/	P0
Unknown	empty	0	D3		/SYS/MB/	P0
Unknown	in use	0	D8		/SYS/MB/	P0
Unknown	empty	0	D7		/SYS/MB/	P0
Unknown	empty	0	D6		/SYS/MB/	P0
Unknown	empty	0	D2		/SYS/MB/	P0
Unknown	empty	0	D1		/SYS/MB/	P0
Unknown	empty	0	D0		/SYS/MB/	P0
Unknown	empty	0	D5		/SYS/MB/	P0
Unknown	empty	0	D4		/SYS/MB/	P0
Unknown	empty	0	D3		/SYS/MB/	P0

```
Unknown empty 0 D8 /SYS/MB/P0
Unknown empty 0 D7 /SYS/MB/P0
Unknown empty 0 D6 /SYS/MB/P0
FLASH in use 0
```

==== On-Board Devices =====

```
Intel 82576 Ethernet Controller.
Intel 82576 Ethernet Controller.
Intel 82576 Ethernet Controller.
Intel 82576 Ethernet Controller.
ICH10R SATA Controller.
```

==== Upgradeable Slots =====

```
ID Status Type Description
```

```
1 in use PCI Express PCIE0
2 available PCI Express PCIE1
3 available PCI Express PCIE2
4 in use PCI Express PCIE3
5 available PCI Express PCIE4
6 available PCI Express PCIE5
```

**prtconf**

```
System Configuration: Oracle Corporation i86pc
Memory size: 12280 Megabytes
System Peripherals (Software Nodes):
```

i86pc

```
scsi_vhci, instance #0
pci, instance #0
pci108e,484c (driver not attached)
pci8086,3408, instance #0
pci108e,484c, instance #0
pci108e,484c, instance #1
pci8086,3409, instance #1
pci108e,484c, instance #2
pci108e,484c, instance #3
pci8086,340a, instance #2
pci1077,171, instance #0
pci1077,171, instance #1
pci8086,340c, instance #3
pci1000,9263, instance #0
sd, instance #0
sd, instance #1
pci8086,340e, instance #4
pci111d,806e, instance #6
pci111d,806e, instance #7
pci111d,806e, instance #8
pci8086,3410, instance #5
pci111d,806e, instance #9
pci111d,806e, instance #10
pci111d,806e, instance #11
pci8086,342d (driver not attached)
pci8086,342e (driver not attached)
pci8086,3422 (driver not attached)
```

```
pci8086,3423, instance #0
pci8086,3438 (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c, instance #0
pci108e,484c, instance #1
pci108e,484c, instance #2
device, instance #0
keyboard, instance #2
mouse, instance #3
pci108e,484c, instance #0
hub, instance #0
communications, instance #0
pci108e,484c, instance #3
pci108e,484c, instance #4
keyboard, instance #0
pci108e,484c, instance #5
mouse, instance #1
pci108e,484c, instance #1
hub, instance #1
pci8086,244e, instance #0
display, instance #0
isa, instance #0
motherboard (driver not attached)
asy, instance #0
motherboard (driver not attached)
pit_beep, instance #0
pci108e,484c, instance #0
pci108e,484c (driver not attached)
ioapics (driver not attached)
ioapic, instance #0 (driver not attached)
fw, instance #0
cpu, instance #0
cpu, instance #1
cpu, instance #2
cpu, instance #3
cpu, instance #4
cpu, instance #5
cpu, instance #6
cpu, instance #7
cpu, instance #8
cpu, instance #9
cpu, instance #10
cpu, instance #11
sb, instance #1
used-resources (driver not attached)
iscsi, instance #0
fcoe, instance #0
options, instance #0
pseudo, instance #0
xsvc, instance #0
vga_arbiter, instance #0
```

**format**

AVAILABLE DISK SELECTIONS:

```
0. c3t0d0 <LSI-MR9261-8i-2.12 cyl 60754 alt 2 hd 255 sec 252>
rootdisk
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@0,0
1. c3t1d0 <LSI-MR9261-8i-2.50-18.18TB>
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@1,0
Specify disk (enter its number):
Searching for disks...done
```

AVAILABLE DISK SELECTIONS:

```
0. c3t0d0 <LSI-MR9261-8i-2.12 cyl 60754 alt 2 hd 255 sec 252>
rootdisk
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@0,0
1. c3t1d0 <LSI-MR9261-8i-2.50-18.18TB>
/pci@0,0/pci8086,340c@5/pci1000,9263@0/sd@1,0
Specify disk (enter its number): p
```

PARTITION MENU:

```
0 - change `0' partition
1 - change `1' partition
2 - change `2' partition
3 - change `3' partition
4 - change `4' partition
5 - change `5' partition
6 - change `6' partition
select - select a predefined table
modify - modify a predefined partition table
name - name the current table
print - display the current table
label - write partition map and label to the disk
!<cmd> - execute <cmd>, then return
quit
```

partition> p

Current partition table (original):

Total disk sectors available: 39042932669 + 16384 (reserved sectors)

Part	Tag	Flag	First Sector	Size	Last Sector
0	usr	wm	34	800.00GB	1677721633
1	usr	wm	1677721634	800.00GB	3355443233
2	usr	wm	3355443234	100.00GB	3565158433
3	usr	wm	3565158434	100.00GB	3774873633
4	usr	wm	3774873634	8.00TB	20954742817
5	usr	wm	20954742818	8.00TB	38134612001
6	usr	wm	38134612002	100.00MB	38134816801
8	reserved	wm	39042932703	8.00MB	39042949086

**/etc/vfstab**

```
#device device mount FS fsck mount mount
#to mount to fsck point type pass at boot options
```

```

#
/devices - /devices devfs - no -
/proc - /proc proc - no -
ctfs - /system/contract ctfs - no -
objfs - /system/object objfs - no -
sharefs - /etc/dfs/sharetab sharefs - no -
fd - /dev/fd fd - no -
swap - /tmp tmpfs - yes -

/dev/zvol/dsk/rpool/swap - - swap - no -

/etc/system
*ident "%Z%%M% %I% %E% SMI" /* SVR4 1.5 */
*
* SYSTEM SPECIFICATION FILE
*
* moddir:
*
* Set the search path for modules. This has a format similar to the
* csh path variable. If the module isn't found in the first directory
* it tries the second and so on. The default is /kernel /usr/kernel
*
* Example:
*
* moddir: /kernel /usr/kernel /other/modules
*
* root device and root filesystem configuration:
*
* The following may be used to override the defaults provided by
* the boot program:
*
* rootfs: Set the filesystem type of the root.
*
* rootdev: Set the root device. This should be a fully
* expanded physical pathname. The default is
*
* the physical pathname of the device where the
* boot program resides. The physical pathname is
* highly platform and configuration
* dependent.
*
* Example:
*
* rootfs:ufs
* rootdev:/sbus@1,f8000000/esp@0,800000/sd@3,0:a
*
* (Swap device configuration should be specified in /etc/vfstab.)
*
* exclude:
*
* Modules appearing in the moddir path which are NOT to be
* loaded,
*
* even if referenced. Note that `exclude' accepts either a module
name,
*
* or a filename which includes the directory.
*
* Examples:
*
* exclude: win
* exclude: sys/shmsys
*
* forceload:
*
* Cause these modules to be loaded at boot time, (just before
mounting
*
* the root filesystem) rather than at first reference. Note that
* forceload expects a filename which includes the directory. Also
* note that loading a module does not necessarily imply that it will
* be installed.
*
* Example:
*
* forceload: drv/foo
*
* set:
*
* Set an integer variable in the kernel or a module to a new value.
* This facility should be used with caution. See system(4).
*
* Examples:
*
* To set variables in 'unix':
*
* set nautopush=32
* set maxusers=40
*
* To set a variable named 'debug' in the module named 'test_module'
*
* set test_module:debug = 0x13
*
* set autoup=300
* set tune_t_fsflushr=5
*
* set maxphys=262144
*
* set apix_enable=0
* set intel_iommu:immu_enable=0
*
* mpt.conf
*
* #
* # Copyright (c) 2011, Oracle and/or its affiliates. All rights reserved.
* #
* #
* # The mpt driver, as a pHCI driver, must specify the vHCI class it
* # belongs to( SCSI_vhci).
* #
ddi-vhci-class="scsi_vhci";
#
# I/O multipathing feature (MPxIO) can be enabled or disabled using
# mpzio-disable property. Setting mpzio-disable="no" will activate
# I/O multipathing; setting mpzio-disable="yes" disables the feature.
#
# Global mpzio-disable property:
#
# To globally enable MPxIO on all mpt controllers set:
# mpzio-disable="no";
#
# To globally disable MPxIO on all mpt controllers set:
# mpzio-disable="yes";
#
# You can also enable or disable MPxIO on a per HBA basis.
# Per HBA settings override the global setting for the specified HBAs.
# To disable MPxIO on a controller whose parent is /pci@7c0/pci@0/pci@9
# and the unit-address is "0" set:
# name="mpt" parent="/pci@7c0/pci@0/pci@9" unit-address="0" mpzio-
disable="yes";
#
mpzio-disable="yes";
#
# SATA mpzio supported
#
# To disable SATA mpzio, set
# disable-sata-mpzio="yes";
# When mpzio-disable="yes" is set, the disable-sata-mpzio property
# takes no effect
#
disable-sata-mpzio="no";

/etc/project
system:0:::
user.root:1:::
nopproject:2:::
default:3:::
group.staff:10:::

niwot.conf
Virtual Drive: 1 (Target Id: 1)
Name :
RAID Level : Primary-0, Secondary-0, RAID Level Qualifier-0
Size : 18.180 TB
Parity Size : 0
State : Optimal
Strip Size : 256 KB
Number Of Drives : 10
Span Depth : 1
Default Cache Policy: WriteBack, ReadAheadNone, Cached, Write Cache

```

OK if Bad BBU	600144f0054c0f0000000000a28d221	14682095616	600144f0054c0f0000000000a28d250	14682095616
Current Cache Policy: WriteBack, ReadAheadNone, Cached, Write Cache	/dev/md/rdsk/d221		/dev/md/rdsk/d250	
OK if Bad BBU	600144f0054c0f0000000000a28d222	14682095616	600144f0054c0f0000000000a28d251	14682095616
Access Policy : Read/Write	/dev/md/rdsk/d222		/dev/md/rdsk/d251	
Disk Cache Policy : Disabled	600144f0054c0f0000000000a28d223	14682095616	600144f0054c0f0000000000a28d252	14682095616
Encryption Type : None	/dev/md/rdsk/d223		/dev/md/rdsk/d252	
<b>sbadm</b>	600144f0054c0f0000000000a28d224	14682095616	600144f0054c0f0000000000a28d253	14682095616
sbadm list-lu output:	/dev/md/rdsk/d224		/dev/md/rdsk/d253	
	600144f0054c0f0000000000a28d225	14682095616	600144f0054c0f0000000000a28d254	14682095616
	/dev/md/rdsk/d225		/dev/md/rdsk/d254	
	600144f0054c0f0000000000a28d226	14682095616	600144f0054c0f0000000000a28d255	14682095616
	/dev/md/rdsk/d226		/dev/md/rdsk/d255	
Found 82 LU(s)	600144f0054c0f0000000000a28d227	14682095616	600144f0054c0f0000000000a28d256	14682095616
	/dev/md/rdsk/d227		/dev/md/rdsk/d256	
	600144f0054c0f0000000000a28d228	14682095616	600144f0054c0f0000000000a28d257	14682095616
	/dev/md/rdsk/d228		/dev/md/rdsk/d257	
	600144f0054c0f0000000000a28d229	14682095616	600144f0054c0f0000000000a28d258	14682095616
	/dev/md/rdsk/d229		/dev/md/rdsk/d258	
	600144f0054c0f0000000000a28d230	14682095616	600144f0054c0f0000000000a28d259	14682095616
	/dev/md/rdsk/d230		/dev/md/rdsk/d259	
	600144f0054c0f0000000000a28d231	14682095616	600144f0054c0f0000000000a28d260	14682095616
	/dev/md/rdsk/d231		/dev/md/rdsk/d260	
	600144f0054c0f0000000000a28d232	14682095616	600144f0054c0f0000000000a28d261	14682095616
	/dev/md/rdsk/d232		/dev/md/rdsk/d261	
	600144f0054c0f0000000000a28d233	14682095616	600144f0054c0f0000000000a28d262	14682095616
	/dev/md/rdsk/d233		/dev/md/rdsk/d262	
	600144f0054c0f0000000000a28d234	14682095616	600144f0054c0f0000000000a28d263	14682095616
	/dev/md/rdsk/d234		/dev/md/rdsk/d263	
	600144f0054c0f0000000000a28d235	14682095616	600144f0054c0f0000000000a28d264	14682095616
	/dev/md/rdsk/d235		/dev/md/rdsk/d264	
	600144f0054c0f0000000000a28d236	14682095616	600144f0054c0f0000000000a28d265	14682095616
	/dev/md/rdsk/d236		/dev/md/rdsk/d265	
	600144f0054c0f0000000000a28d237	14682095616	600144f0054c0f0000000000a28d266	14682095616
	/dev/md/rdsk/d237		/dev/md/rdsk/d266	
	600144f0054c0f0000000000a28d238	14682095616	600144f0054c0f0000000000a28d267	14682095616
	/dev/md/rdsk/d238		/dev/md/rdsk/d267	
	600144f0054c0f0000000000a28d239	14682095616	600144f0054c0f0000000000a28d268	14682095616
	/dev/md/rdsk/d239		/dev/md/rdsk/d268	
	600144f0054c0f0000000000a28d240	14682095616	600144f0054c0f0000000000a28d269	14682095616
	/dev/md/rdsk/d240		/dev/md/rdsk/d269	
	600144f0054c0f0000000000a28d241	14682095616	600144f0054c0f0000000000a28d270	14682095616
	/dev/md/rdsk/d241		/dev/md/rdsk/d270	
	600144f0054c0f0000000000a28d242	14682095616	600144f0054c0f0000000000a28d271	14682095616
	/dev/md/rdsk/d242		/dev/md/rdsk/d271	
	600144f0054c0f0000000000a28d243	14682095616	600144f0054c0f0000000000a28d272	14682095616
	/dev/md/rdsk/d243		/dev/md/rdsk/d272	
	600144f0054c0f0000000000a28d244	14682095616	600144f0054c0f0000000000a28d273	14682095616
	/dev/md/rdsk/d244		/dev/md/rdsk/d273	
	600144f0054c0f0000000000a28d245	14682095616	600144f0054c0f0000000000a28d274	14682095616
	/dev/md/rdsk/d245		/dev/md/rdsk/d274	
	600144f0054c0f0000000000a28d246	14682095616	600144f0054c0f0000000000a28d275	14682095616
	/dev/md/rdsk/d246		/dev/md/rdsk/d275	
	600144f0054c0f0000000000a28d247	14682095616	600144f0054c0f0000000000a28d276	14682095616
	/dev/md/rdsk/d247		/dev/md/rdsk/d276	
	600144f0054c0f0000000000a28d248	14682095616	600144f0054c0f0000000000a28d277	14682095616
	/dev/md/rdsk/d248		/dev/md/rdsk/d277	
	600144f0054c0f0000000000a28d249	14682095616	600144f0054c0f0000000000a28d278	14682095616
	/dev/md/rdsk/d249		/dev/md/rdsk/d278	

600144f0054c0f00000000000a28d279 14682095616  
/dev/md/rdsd/d279  
600144f0a3770e0000000000a03d300 858993393664  
/dev/rdsd/c3t1d0s0  
600144f0a3770e0000000000a03d303 858993393664  
/dev/rdsd/c3t1d0s1

**metastat**

d278 -p /dev/md/rdsd/d120 -o 8855956480 -b 28676096  
d120 l 1 /dev/rdsd/c3t1d0s4  
d277 -p /dev/md/rdsd/d120 -o 8827279360 -b 28676096  
d276 -p /dev/md/rdsd/d120 -o 87988602240 -b 28676096  
d275 -p /dev/md/rdsd/d120 -o 8769925120 -b 28676096  
d274 -p /dev/md/rdsd/d120 -o 8741248000 -b 28676096  
d273 -p /dev/md/rdsd/d120 -o 8712570880 -b 28676096  
d272 -p /dev/md/rdsd/d120 -o 8683893760 -b 28676096  
d271 -p /dev/md/rdsd/d120 -o 8655216640 -b 28676096  
d270 -p /dev/md/rdsd/d120 -o 8626539520 -b 28676096  
d269 -p /dev/md/rdsd/d120 -o 8597862400 -b 28676096  
d279 -p /dev/md/rdsd/d120 -o 8884633600 -b 28676096  
d268 -p /dev/md/rdsd/d120 -o 8569185280 -b 28676096  
d267 -p /dev/md/rdsd/d120 -o 8540508160 -b 28676096  
d266 -p /dev/md/rdsd/d120 -o 8511831040 -b 28676096  
d265 -p /dev/md/rdsd/d120 -o 8483153920 -b 28676096  
d264 -p /dev/md/rdsd/d120 -o 8454476800 -b 28676096  
d262 -p /dev/md/rdsd/d120 -o 8397122560 -b 28676096  
d261 -p /dev/md/rdsd/d120 -o 8368445440 -b 28676096  
d260 -p /dev/md/rdsd/d120 -o 8339768320 -b 28676096  
d259 -p /dev/md/rdsd/d120 -o 8311091200 -b 28676096  
d258 -p /dev/md/rdsd/d120 -o 8282414080 -b 28676096  
d257 -p /dev/md/rdsd/d120 -o 8253736960 -b 28676096  
d256 -p /dev/md/rdsd/d120 -o 8225059840 -b 28676096  
d255 -p /dev/md/rdsd/d120 -o 8196382720 -b 28676096  
d254 -p /dev/md/rdsd/d120 -o 8167705600 -b 28676096  
d263 -p /dev/md/rdsd/d120 -o 8425799680 -b 28676096  
d253 -p /dev/md/rdsd/d120 -o 8139028480 -b 28676096  
d252 -p /dev/md/rdsd/d120 -o 8110351360 -b 28676096  
d251 -p /dev/md/rdsd/d120 -o 8081674240 -b 28676096  
d250 -p /dev/md/rdsd/d120 -o 8052997120 -b 28676096  
d249 -p /dev/md/rdsd/d120 -o 8024320000 -b 28676096  
d248 -p /dev/md/rdsd/d120 -o 7995642880 -b 28676096  
d246 -p /dev/md/rdsd/d120 -o 7938288640 -b 28676096  
d245 -p /dev/md/rdsd/d120 -o 7909611520 -b 28676096  
d244 -p /dev/md/rdsd/d120 -o 7880934400 -b 28676096  
d243 -p /dev/md/rdsd/d120 -o 7852257280 -b 28676096  
d242 -p /dev/md/rdsd/d120 -o 7823580160 -b 28676096  
d241 -p /dev/md/rdsd/d120 -o 7794903040 -b 28676096  
d240 -p /dev/md/rdsd/d120 -o 7766225920 -b 28676096  
d239 -p /dev/md/rdsd/d120 -o 7737548800 -b 28676096  
d238 -p /dev/md/rdsd/d120 -o 7708871680 -b 28676096  
d247 -p /dev/md/rdsd/d120 -o 7966965760 -b 28676096  
d237 -p /dev/md/rdsd/d120 -o 7680194560 -b 28676096  
d236 -p /dev/md/rdsd/d120 -o 7651517440 -b 28676096  
d235 -p /dev/md/rdsd/d120 -o 7622840320 -b 28676096  
d234 -p /dev/md/rdsd/d120 -o 7594163200 -b 28676096

d233 -p /dev/md/rdsd/d120 -o 7565486080 -b 28676096  
d232 -p /dev/md/rdsd/d120 -o 7536808960 -b 28676096  
d230 -p /dev/md/rdsd/d120 -o 7479454720 -b 28676096  
d229 -p /dev/md/rdsd/d120 -o 7450777600 -b 28676096  
d228 -p /dev/md/rdsd/d120 -o 7422100480 -b 28676096  
d227 -p /dev/md/rdsd/d120 -o 7393423360 -b 28676096  
d226 -p /dev/md/rdsd/d120 -o 7364746240 -b 28676096  
d225 -p /dev/md/rdsd/d120 -o 7336069120 -b 28676096  
d224 -p /dev/md/rdsd/d120 -o 7307392000 -b 28676096  
d223 -p /dev/md/rdsd/d120 -o 7278714880 -b 28676096  
d222 -p /dev/md/rdsd/d120 -o 7250037760 -b 28676096  
d231 -p /dev/md/rdsd/d120 -o 7508131840 -b 28676096  
d221 -p /dev/md/rdsd/d120 -o 7221360640 -b 28676096  
d220 -p /dev/md/rdsd/d120 -o 7192683520 -b 28676096  
d219 -p /dev/md/rdsd/d120 -o 7164006400 -b 28676096  
d218 -p /dev/md/rdsd/d120 -o 7135329280 -b 28676096  
d217 -p /dev/md/rdsd/d120 -o 7106652160 -b 28676096  
d216 -p /dev/md/rdsd/d120 -o 7077975040 -b 28676096  
d214 -p /dev/md/rdsd/d120 -o 7020620800 -b 28676096  
d213 -p /dev/md/rdsd/d120 -o 6991943680 -b 28676096  
d212 -p /dev/md/rdsd/d120 -o 6963266560 -b 28676096  
d211 -p /dev/md/rdsd/d120 -o 6934589440 -b 28676096  
d210 -p /dev/md/rdsd/d120 -o 6905912320 -b 28676096  
d209 -p /dev/md/rdsd/d120 -o 6877235200 -b 28676096  
d208 -p /dev/md/rdsd/d120 -o 6848558080 -b 28676096  
d207 -p /dev/md/rdsd/d120 -o 6819880960 -b 28676096  
d206 -p /dev/md/rdsd/d120 -o 6791203840 -b 28676096  
d215 -p /dev/md/rdsd/d120 -o 7049297920 -b 28676096  
d205 -p /dev/md/rdsd/d120 -o 6762526720 -b 28676096  
d204 -p /dev/md/rdsd/d120 -o 6733849600 -b 28676096  
d203 -p /dev/md/rdsd/d120 -o 6705172480 -b 28676096  
d202 -p /dev/md/rdsd/d120 -o 6676495360 -b 28676096  
d201 -p /dev/md/rdsd/d120 -o 6647818240 -b 28676096  
d200 -p /dev/md/rdsd/d120 -o 6619141120 -b 28676096  
d7502 -p /dev/md/rdsd/d130 -o 3728999424 -b 1654784000  
d130 l 1 /dev/rdsd/c3t1d0s5  
d7501 -p /dev/md/rdsd/d130 -o 2074214400 -b 1654784000  
d7500 -p /dev/md/rdsd/d130 -o 419429376 -b 1654784000  
d7403 -p /dev/md/rdsd/d120 -o 4964356096 -b 1654784000  
d7402 -p /dev/md/rdsd/d120 -o 3309571072 -b 1654784000  
d7401 -p /dev/md/rdsd/d120 -o 1654786048 -b 1654784000  
d7400 -p /dev/md/rdsd/d120 -o 1024 -b 1654784000  
d7301 -p /dev/md/rdsd/d130 -o 209715200 -b 209714000  
d7300 -p /dev/md/rdsd/d130 -o 1024 -b 209714000  
d7503 -p /dev/md/rdsd/d130 -o 5383784448 -b 1654784000

**Clients**

**prtdiag**

System Configuration: SUN MICROSYSTEMS SUN FIRE X4170 M2  
SERVER

BIOS Configuration: American Megatrends Inc. 08030103 04/28/2010  
BMC Configuration: IPMI 1.5 (KCS: Keyboard Controller Style)

==== Processor Sockets =====

Version	Location Tag
Intel(R) Xeon(R) CPU	X5670 @ 2.93GHz CPU 0
Intel(R) Xeon(R) CPU	X5670 @ 2.93GHz CPU 1

==== Memory Device Sockets =====

Type	Status	Set	Device	Locator	Bank	Locator
Unknown in use	0	D2			BANK2	
Unknown in use	0	D1			BANK1	
Unknown empty	0	D0			BANK0	
Unknown in use	0	D5			BANK5	
Unknown in use	0	D4			BANK4	
Unknown empty	0	D3			BANK3	
Unknown in use	0	D8			BANK8	
Unknown in use	0	D7			BANK7	
Unknown empty	0	D6			BANK6	
Unknown in use	0	D2			BANK2	
Unknown in use	0	D1			BANK1	
Unknown empty	0	D0			BANK0	
Unknown in use	0	D5			BANK5	
Unknown in use	0	D4			BANK4	
Unknown empty	0	D3			BANK3	
Unknown in use	0	D8			BANK8	
Unknown in use	0	D7			BANK7	
Unknown empty	0	D6			BANK6	
Unknown in use	0	D2			BANK2	
Unknown in use	0	D1			BANK1	
Unknown empty	0	D0			BANK0	
Unknown in use	0	D5			BANK5	
Unknown in use	0	D4			BANK4	
Unknown empty	0	D3			BANK3	
Unknown in use	0	D8			BANK8	
Unknown in use	0	D7			BANK7	
Unknown empty	0	D6			BANK6	
FLASH	in use	0				

==== On-Board Devices =====

Intel 82576 Ethernet Controller.  
Intel 82576 Ethernet Controller.  
Intel 82576 Ethernet Controller.  
Intel 82576 Ethernet Controller.

==== Upgradeable Slots =====

ID	Status	Type	Description
0	in use	PCI Express	PCIE0
1	in use	PCI Express	PCIE1
2	available	PCI Express	PCIE2

**prtconf**

System Configuration: Sun Microsystems i86pc  
Memory size: 49144 Megabytes  
System Peripherals (Software Nodes):

i86pc  
scsi\_vhci, instance #0  
isa, instance #0  
motherboard (driver not attached)  
asy, instance #0

```

motherboard (driver not attached)
pci, instance #0
pci108e,484c (driver not attached)
pci8086,3408, instance #0
pci108e,484c, instance #0
pci108e,484c, instance #1
pci8086,3409, instance #1
pci108e,484c, instance #2
pci108e,484c, instance #3
pci8086,340a, instance #2
pci1000,9263, instance #0
sd, instance #0
sd, instance #1
pci8086,340e (driver not attached)
pci8086,3410 (driver not attached)
pci8086,342d (driver not attached)
pci8086,342e (driver not attached)
pci8086,3422 (driver not attached)
pci8086,3423, instance #0
pci8086,3438 (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c (driver not attached)
pci108e,484c, instance #0
pci108e,484c, instance #1
pci108e,484c, instance #2
device, instance #0
keyboard, instance #0
mouse, instance #1
pci108e,484c, instance #0
hub, instance #1
pci108e,484c, instance #3
pci108e,484c, instance #4
pci108e,484c, instance #5
pci108e,484c, instance #1
hub, instance #0
pci8086,244e, instance #0
display, instance #0
pci108e,484c (driver not attached)
pci108e,484c, instance #0
pci108e,484c (driver not attached)
ioapics (driver not attached)
ioapic, instance #0 (driver not attached)
iscsi, instance #0
pseudo, instance #0
agpgart, instance #0 (driver not attached)
options, instance #0
objmgr, instance #0
xsvc, instance #0 (driver not attached)
used-resources (driver not attached)
cpus, instance #0
cpu, instance #0
cpu, instance #1

cpu, instance #2
cpu, instance #3
cpu, instance #4
cpu, instance #5
cpu, instance #6
cpu, instance #7
cpu, instance #8
cpu, instance #9
cpu, instance #10
cpu, instance #11

format
Searching for disks...done

AVAILABLE DISK SELECTIONS:
0. c0t0d0 <DEFAULT cyl 17747 alt 2 hd 255 sec 63>
   /pci@0,0/pci8086,340a@3/pci1000,9263@0/sd@0,0
1. c0t1d0 <DEFAULT cyl 17747 alt 2 hd 255 sec 63>
   /pci@0,0/pci8086,340a@3/pci1000,9263@0/sd@1,0
Specify disk (enter its number):

/etc/release
Oracle Solaris 10 9/10 s10x_u9wos_14a X86
Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
Assembled 11 August 2010

/etc/vfstab
#device          device          mount          FS
#to mount        to fsck         mount          point          type          pass
#
fd               -              /dev/fd        fd              -              no             -
/proc            -              /proc          proc            -              no             -
/dev/md/dsk/d1   -              -              -              swap          -              no
/dev/md/dsk/d0   /dev/md/dsk/d0 /dev/md/rdisk/d0 /              ufs           1
/devices         -              /devices      devfs          -              no             -
sharefs         -              /etc/dfs/sharetab sharefs        -              no
ctfs             -              /system/contract ctfs          -              no
objfs           -              /system/object  objfs         -              no
swap            -              /tmp          tmpfs         -              yes            -

/etc/hosts
#
# Internet host table
#
::1             localhost
127.0.0.1      localhost
10.137.233.1   dccl1 dccl1 x4170-017 loghost
#
#####
##### #

```



```

# Private Point to Point (p2p) connections between ClientX nodes and
# rteX nodes. Because these are POINT TO POINT, and further, PRIVATE to name,
# the three node Client/RTE tuples of three, this ONE section good for ALL!!
#
#The following is GENERIC FOR ALL CLIENT DRIVER TUPLES!
192.168.200.1 drvrte1 # p2p, client driver 1, clientX:igb2
192.168.200.2 rte1drv # p2p, client driver 1, rteX-1:igb2
192.168.201.1 drvrte2 # p2p, client driver 2, clientX:igb3
192.168.201.2 rte2drv # p2p, client driver 2, rteX-2:igb3
#
# endit: client2driver (or RTE) p2p's

/etc/system
*ident "@(#)system 1.18 97/06/27 SMI" /* SVR4 1.5 */
*
* SYSTEM SPECIFICATION FILE
*
* moddir:
*
* Set the search path for modules. This has a format similar to the
* csh path variable. If the module isn't found in the first directory
* it tries the second and so on. The default is /kernel /usr/kernel
*
* Example:
* moddir: /kernel /usr/kernel /other/modules
*
* root device and root filesystem configuration:
*
* The following may be used to override the defaults provided by
* the boot program:
*
* rootfs: Set the filesystem type of the root.
*
* rootdev: Set the root device. This should be a fully
* expanded physical pathname. The default is
*
* physical pathname of the device where the
* program resides. The physical pathname is
* highly platform and configuration
* dependent.
*
* Example:
* rootfs:ufs
* rootdev:/sbus@1,f8000000/esp@0,8000000/sd@3,0:a
*
* (Swap device configuration should be specified in /etc/vfstab.)
*
* exclude:
*
* Modules appearing in the moddir path which are NOT to be
* loaded,

```

```

* even if referenced. Note that 'exclude' accepts either a module
* or a filename which includes the directory.
*
* Examples:
* exclude: win
* exclude: sys/shmsys
*
* * forceload:
*
* Cause these modules to be loaded at boot time, (just before
* mounting
* the root filesystem) rather than at first reference. Note that
* forceload expects a filename which includes the directory. Also
* note that loading a module does not necessarily imply that it will
* be installed.
*
* Example:
* forceload: drv/foo
*
* * set:
*
* Set an integer variable in the kernel or a module to a new value.
* This facility should be used with caution. See system(4).
*
* Examples:
*
* To set variables in 'unix':
*
* set nautopush=32
* set maxusers=40
*
* To set a variable named 'debug' in the module named 'test_module'
*
* set test_module:debug = 0x13
*
* * Begin MDD root info (do not edit)
* rootdev:/pseudo/md@0:0,0,blk
* * End MDD root info (do not edit)
* *TPCC-Setup-Pallab-Do-not-play with it
* set rlim_fd_max=2000000
* set rlim_fd_cur=2000000
* set sq_max_size=0
* *set ddi_msix_alloc_limit=4
* *set pcpusmp:apic_multi_msi_max=4
* *set pcpusmp:apic_msix_max=4
* *set pcpusmp:apic_intr_policy=1
*
* *set ip:ip_queue_bind=0
* *set ip:ip_queue_fanout=1
* *set ip:ip_soft_rings_cnt=4
* *set ip_queue_soft_ring=1
* *set ip:tcp_queue_wput=1

```

```

set shmsys:shminfo_shmmax=0xffffffff
set shmsys:shminfo_shmmin=1
set shmsys:shminfo_shmmni=800
set shmsys:shminfo_shmseg=800
*set shmsys:shminfo_shmmni=500
*set shmsys:shminfo_shmseg=600
*
set semsys:seminfo_semmni=24576
set semsys:seminfo_semmns=1500
set semsys:seminfo_semmns=18432
set semsys:seminfo_semmnu=12000
set semsys:seminfo_semopm=500
set semsys:seminfo_semvmx=65536
*
set msgsys:msginfo_msgmni=5000
set msgsys:msginfo_msgmap=200000
set msgsys:msginfo_msgmnb=800000
set msgsys:msginfo_msgmax=32768
set msgsys:msginfo_msgtql=5000

/etc/project
system:0::::
user.root:1::::
noproject:2::::
default:3::::
group.staff:10::::
group.dba:100:DBA Project::dba:process.max-sem-
nsems=(priv,1024,deny);project.max-sem-ids=(priv,1024,
deny);project.max-shm-memory=(priv,644245094400,deny);project.max-
msg-ids=(priv,8192,deny)

/etc/user_attr
#
# Copyright 2007 Sun Microsystems, Inc. All rights reserved.
# Use is subject to license terms.
#
# /etc/user_attr
#
# execution attributes for profiles. see user_attr(4)
#
#ident "@(#)user_attr 1.1 07/01/31 SMI"
#
#
adm::::profiles=Log Management
lp::::profiles=Printer Management
postgres::::type=role;profiles=Postgres Administration,All
root::::auths=solaris.*;solaris.grant;profiles=Web Console
Management,All;lock_after_retries=no;min_label=admin_low;clearance=admin
_high
tpc::::type=normal;profiles=Primary Administrator
dbbench::::type=normal;profiles=Primary Administrator

metastat
d0 -m d4 d5 1
d4 1 1 c0t0d0s0
d5 1 1 c0t1d0s0
d1 -m d2 d3 1
d2 1 1 c0t0d0s1

```

```

d3 1 1 c0t1d0s1
ubbcfg
*RESOURCES
IPCKEY 40001
MASTER tpccli
PERM 0666
MAXACCESSERS 4096
MAXSERVERS 4096
MAXSERVICES 4096
MAXBUFTYPE 8192
MAXBUFSTYPE 8192
NOTIFY IGNORE
MODEL SHM
LDBAL Y
SCANUNIT 10
SANITYSCAN 10
BLOCKTIME 10
BBLQUERY 10
OPTIONS NO_XA,NO_AA

*MACHINES
"dccl1" LMID="tpcccli"
TUXCONFIG="/export/home/tpc/tuxedo/tuxconfig"
ROOTDIR="/opt/tuxedo"
APPDIR="/export/home/tpc/tuxedo"
ULOGPFX="/export/home/tpc/tuxedo/logs/ULOGHOST"
TUXDIR="/opt/tuxedo"
SPINCOUNT=10000

*GROUPS
"DEL1" LMID="tpcccli" GRPNO=1 ENVFILE="env.two_task1"
"DEL2" LMID="tpcccli" GRPNO=2 ENVFILE="env.two_task2"
"DEL3" LMID="tpcccli" GRPNO=3 ENVFILE="env.two_task3"
"DEL4" LMID="tpcccli" GRPNO=4 ENVFILE="env.two_task4"
"group1" LMID="tpcccli" GRPNO=5 ENVFILE="env.two_task1"
"group2" LMID="tpcccli" GRPNO=6 ENVFILE="env.two_task2"
"group3" LMID="tpcccli" GRPNO=7 ENVFILE="env.two_task3"
"group4" LMID="tpcccli" GRPNO=8 ENVFILE="env.two_task4"

*SERVERS
tpccsrvdel SRVGRP="DEL1" SRVID=1 RQADDR=delq1 REPLYQ=N
CLOPT="-A -- 1"
tpccsrvdel SRVGRP="DEL2" SRVID=2 RQADDR=delq2 REPLYQ=N
CLOPT="-A -- 2"
tpccsrvdel SRVGRP="DEL3" SRVID=3 RQADDR=delq3 REPLYQ=N
CLOPT="-A -- 3"
tpccsrvdel SRVGRP="DEL4" SRVID=4 RQADDR=delq4 REPLYQ=N
CLOPT="-A -- 4"
tpccsrvdel SRVGRP="DEL1" SRVID=5 RQADDR=delq5 REPLYQ=N
CLOPT="-A -- 5"
tpccsrvdel SRVGRP="DEL2" SRVID=6 RQADDR=delq6 REPLYQ=N
CLOPT="-A -- 6"
#tpccsrvdel SRVGRP="DEL3" SRVID=7 RQADDR=delq7 REPLYQ=N
CLOPT="-A -- 7"
#tpccsrvdel SRVGRP="DEL4" SRVID=8 RQADDR=delq8 REPLYQ=N
CLOPT="-A -- 8"

#tpccsrvdel SRVGRP="DEL1" SRVID=9 RQADDR=delq9 REPLYQ=N MIN=1 MAX=1
CLOPT="-A -- 9"
tpccsrvora SRVGRP="group3" SRVID=103 RQADDR=svcq3 REPLYQ=Y
#tpccsrvdel SRVGRP="DEL2" SRVID=10 RQADDR=delq10 REPLYQ=N MIN=1 MAX=1
CLOPT="-A -- 10"
tpccsrvora SRVGRP="group4" SRVID=104 RQADDR=svcq4 REPLYQ=Y
#tpccsrvdel SRVGRP="DEL3" SRVID=11 RQADDR=delq11 REPLYQ=N MIN=1 MAX=1
CLOPT="-A -- 11"
tpccsrvora SRVGRP="group1" SRVID=105 RQADDR=svcq5 REPLYQ=Y
#tpccsrvdel SRVGRP="DEL4" SRVID=12 RQADDR=delq12 REPLYQ=N MIN=1 MAX=1
CLOPT="-A -- 12"
tpccsrvora SRVGRP="group2" SRVID=106 RQADDR=svcq6 REPLYQ=Y
#tpccsrvdel SRVGRP="DEL1" SRVID=13 RQADDR=delq13 REPLYQ=N MIN=1 MAX=1
CLOPT="-A -- 13"
tpccsrvora SRVGRP="group3" SRVID=107 RQADDR=svcq7 REPLYQ=Y
#tpccsrvdel SRVGRP="DEL2" SRVID=14 RQADDR=delq14 REPLYQ=N MIN=1 MAX=1
CLOPT="-A -- 14"
tpccsrvora SRVGRP="group4" SRVID=108 RQADDR=svcq8 REPLYQ=Y
#tpccsrvdel SRVGRP="DEL1" SRVID=15 RQADDR=delq15 REPLYQ=N MIN=1 MAX=1
CLOPT="-A -- 15"
tpccsrvora SRVGRP="group1" SRVID=109 RQADDR=svcq9 REPLYQ=Y
#tpccsrvdel SRVGRP="DEL2" SRVID=16 RQADDR=delq16 REPLYQ=N MIN=1 MAX=1
CLOPT="-A -- 16"
tpccsrvora SRVGRP="group2" SRVID=110 RQADDR=svcq10
#tpccsrvdel SRVGRP="DEL3" SRVID=17 RQADDR=delq17 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 17"
tpccsrvora SRVGRP="group3" SRVID=111 RQADDR=svcq11
#tpccsrvdel SRVGRP="DEL4" SRVID=18 RQADDR=delq18 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 18"
tpccsrvora SRVGRP="group4" SRVID=112 RQADDR=svcq12
#tpccsrvdel SRVGRP="DEL1" SRVID=19 RQADDR=delq19 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 19"
tpccsrvora SRVGRP="group1" SRVID=113 RQADDR=svcq13
#tpccsrvdel SRVGRP="DEL2" SRVID=20 RQADDR=delq20 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 20"
tpccsrvora SRVGRP="group2" SRVID=114 RQADDR=svcq14
#tpccsrvdel SRVGRP="DEL3" SRVID=21 RQADDR=delq21 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 21"
tpccsrvora SRVGRP="group3" SRVID=115 RQADDR=svcq15
#tpccsrvdel SRVGRP="DEL4" SRVID=22 RQADDR=delq22 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 22"
tpccsrvora SRVGRP="group4" SRVID=116 RQADDR=svcq16
#tpccsrvdel SRVGRP="DEL1" SRVID=23 RQADDR=delq23 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 23"
tpccsrvora SRVGRP="group1" SRVID=117 RQADDR=svcq17
#tpccsrvdel SRVGRP="DEL2" SRVID=24 RQADDR=delq24 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 24"
tpccsrvora SRVGRP="group2" SRVID=118 RQADDR=svcq18
#tpccsrvdel SRVGRP="DEL3" SRVID=25 RQADDR=delq25 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 25"
tpccsrvora SRVGRP="group3" SRVID=119 RQADDR=svcq19
#tpccsrvdel SRVGRP="DEL4" SRVID=26 RQADDR=delq26 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 26"
tpccsrvora SRVGRP="group4" SRVID=120 RQADDR=svcq20
#tpccsrvdel SRVGRP="DEL1" SRVID=27 RQADDR=delq27 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 27"
tpccsrvora SRVGRP="group1" SRVID=121 RQADDR=svcq21
#tpccsrvdel SRVGRP="DEL2" SRVID=28 RQADDR=delq28 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 28"
tpccsrvora SRVGRP="group2" SRVID=122 RQADDR=svcq22
#tpccsrvdel SRVGRP="DEL3" SRVID=29 RQADDR=delq29 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 29"
tpccsrvora SRVGRP="group3" SRVID=123 RQADDR=svcq23
#tpccsrvdel SRVGRP="DEL4" SRVID=30 RQADDR=delq30 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 30"
tpccsrvora SRVGRP="group4" SRVID=124 RQADDR=svcq24
#tpccsrvdel SRVGRP="DEL1" SRVID=31 RQADDR=delq31 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 31"
tpccsrvora SRVGRP="group1" SRVID=125 RQADDR=svcq25
#tpccsrvdel SRVGRP="DEL2" SRVID=32 RQADDR=delq32 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 32"
tpccsrvora SRVGRP="group2" SRVID=126 RQADDR=svcq26
#tpccsrvdel SRVGRP="DEL3" SRVID=33 RQADDR=delq33 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 33"
tpccsrvora SRVGRP="group3" SRVID=127 RQADDR=svcq27
#tpccsrvdel SRVGRP="DEL4" SRVID=34 RQADDR=delq34 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 34"
tpccsrvora SRVGRP="group4" SRVID=128 RQADDR=svcq28
#tpccsrvdel SRVGRP="DEL1" SRVID=35 RQADDR=delq35 REPLYQ=N REPLYQ=Y MIN=1 MAX=1
CLOPT="-A -- 35"
tpccsrvora SRVGRP="group1" SRVID=129 RQADDR=svcq29
REPLYQ=Y MIN=1 MAX=1
tpccsrvora SRVGRP="group2" SRVID=130 RQADDR=svcq30
REPLYQ=Y MIN=1 MAX=1
tpccsrvora SRVGRP="group3" SRVID=131 RQADDR=svcq31

```



```

REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=219 RQADDR=svcq119
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=220 RQADDR=svcq120
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=221 RQADDR=svcq121
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=222 RQADDR=svcq122
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=223 RQADDR=svcq123
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=224 RQADDR=svcq124
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=225 RQADDR=svcq125
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=226 RQADDR=svcq126
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=227 RQADDR=svcq127
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=228 RQADDR=svcq128
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=229 RQADDR=svcq129
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=230 RQADDR=svcq130
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=231 RQADDR=svcq131
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=232 RQADDR=svcq132
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=233 RQADDR=svcq133
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=234 RQADDR=svcq134
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=235 RQADDR=svcq135
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=236 RQADDR=svcq136
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=237 RQADDR=svcq137
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=238 RQADDR=svcq138
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=239 RQADDR=svcq139
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=240 RQADDR=svcq140
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=241 RQADDR=svcq141
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=242 RQADDR=svcq142
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=243 RQADDR=svcq143
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=244 RQADDR=svcq144
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=245 RQADDR=svcq145
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=246 RQADDR=svcq146
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=247 RQADDR=svcq147

```

```

REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=248 RQADDR=svcq148
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=249 RQADDR=svcq149
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=250 RQADDR=svcq150
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=251 RQADDR=svcq151
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=252 RQADDR=svcq152
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=253 RQADDR=svcq153
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=254 RQADDR=svcq154
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=255 RQADDR=svcq155
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=256 RQADDR=svcq156
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=257 RQADDR=svcq157
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=258 RQADDR=svcq158
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=259 RQADDR=svcq159
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=260 RQADDR=svcq160
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=261 RQADDR=svcq161
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=262 RQADDR=svcq162
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=263 RQADDR=svcq163
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=264 RQADDR=svcq164
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=265 RQADDR=svcq165
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=266 RQADDR=svcq166
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=267 RQADDR=svcq167
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=268 RQADDR=svcq168
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=269 RQADDR=svcq169
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=270 RQADDR=svcq170
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=271 RQADDR=svcq171
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=272 RQADDR=svcq172
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=273 RQADDR=svcq173
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=274 RQADDR=svcq174
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=275 RQADDR=svcq175
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=276 RQADDR=svcq176

```

```

REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=277 RQADDR=svcq177
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=278 RQADDR=svcq178
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=279 RQADDR=svcq179
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=280 RQADDR=svcq180
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=281 RQADDR=svcq181
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=282 RQADDR=svcq182
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group3" SRVID=283 RQADDR=svcq183
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group4" SRVID=284 RQADDR=svcq184
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group1" SRVID=285 RQADDR=svcq185
REPLYQ=Y MIN=1 MAX=1
#tpcc_srv_ora SRVGRP="group2" SRVID=286 RQADDR=svcq186
REPLYQ=Y MIN=1 MAX=1

```

```

*SERVICES
DEL
NEWO
ORDS
PAYM
STOCK

```

## Brocade 5300 SAN Switch

Switch Information Report for b5300-2

List of Switches

Switch ID	Worldwide Name	Enet IP Addr	FC IP Addr	Name
3:	fff03	10:00:00:05:1e:9a:8e:be	10.196.0.101	0.0.0.0 >"b5300-2"

Current Switch Information

```

Ethernet IP Address: 10.196.0.101
Ethernet Subnetmask: 255.255.0.0
Fibre Channel IP Address: 0.0.0.0
Fibre Channel Subnetmask: 0.0.0.0
Gateway Address: 10.196.0.1
Ethernet IPv6 Addresses:

Kernel: 2.6.14.2
Fabric OS: v6.2.0c
Made on: Mon Feb 23 19:32:16 2009
Flash: Thu May 7 13:14:02 2009
BootProm: 1.0.14

```

List of Inter-Switch Links

Local Domain ID: 3

Local Port	Domain	Remote Port	State								
-----				36	36	id	N8	No_Light		FC4s: FCP	
List of Ports				37	37	id	N8	No_Light		PortSymb: [6] "qlt1,0"	
switchName:	b5300-2			38	38	id	N8	No_Light		NodeSymb: [7] "dcdata1"	
switchType:	64.3			39	39	id	N8	No_Light		Fabric Port Name: 20:00:00:05:1e:9a:8e:be	
switchState:	Online			40	40	id	N8	No_Light		Permanent Port Name: 21:00:00:24:ff:20:7d:b5	
switchMode:	Native			41	41	id	N8	No_Light		Port Index: 0	
switchRole:	Principal			42	42	id	N8	No_Light		Share Area: No	
switchDomain:	3			43	43	id	N8	No_Light		Device Shared in Other AD: No	
switchId: fffc03				44	44	id	N8	No_Light		Redirect: No	
switchWwn: 10:00:00:05:1e:9a:8e:be				45	45	id	N8	No_Light		N 030100; 3;21:00:00:24:ff:20:7d:b4;20:00:00:24:ff:20:7d:b4; na	
zoning: ON (dc120f_initial_ewt)				46	46	id	N8	No_Light		FC4s: FCP	
switchBeacon: OFF				47	47	id	N8	No_Light		PortSymb: [6] "qlt0,0"	
FC Router: OFF				48	48	id	N8	No_Light		NodeSymb: [7] "dcdata1"	
Allow XISL Use: OFF				49	49	id	N8	No_Light		Fabric Port Name: 20:01:00:05:1e:9a:8e:be	
LS Attributes: [FID: 128, Base Switch: No, Default Switch: Yes]				50	50	id	N8	No_Light		Permanent Port Name: 21:00:00:24:ff:20:7d:b4	
Area Port Media Speed State Proto				51	51	id	N8	No_Light		Port Index: 1	
0 0 id N8 Online F-Port 21:00:00:24:ff:20:7d:b5				52	52	id	N8	No_Light		Share Area: No	
1 1 id N8 Online F-Port 21:00:00:24:ff:20:7d:b4				53	53	id	N8	No_Light		Device Shared in Other AD: No	
2 2 id N8 Online F-Port 21:00:00:1b:32:92:78:19				54	54	id	N8	No_Light		Redirect: No	
3 3 id N8 Online F-Port 21:01:00:1b:32:b2:78:19				55	55	id	N8	No_Light		N 030200; 3;21:00:00:1b:32:92:78:19;20:00:00:1b:32:92:78:19; na	
4 4 id N8 Online F-Port 21:00:00:24:ff:30:1c:4e				56	56	id	N8	No_Light		FC4s: FCP	
5 5 id N8 Online F-Port 21:00:00:24:ff:30:1c:4f				57	57	id	N8	No_Light		PortSymb: [6] "qlt0,0"	
6 6 id N8 Online F-Port 21:00:00:24:ff:2f:d1:fc				58	58	id	N8	No_Light		NodeSymb: [7] "dcdata2"	
7 7 id N8 Online F-Port 21:00:00:24:ff:2f:d1:fd				59	59	id	N8	No_Light		Fabric Port Name: 20:02:00:05:1e:9a:8e:be	
8 8 id N8 Online F-Port 21:00:00:1b:32:92:8f:19				60	60	id	N8	No_Light		Permanent Port Name: 21:00:00:1b:32:92:78:19	
9 9 id N8 Online F-Port 21:01:00:1b:32:b2:8f:19				61	61	id	N8	No_Light		Port Index: 2	
10 10 id N8 No_Light				62	62	id	N8	No_Light		Share Area: No	
11 11 id N8 No_Light				63	63	id	N8	No_Light		Device Shared in Other AD: No	
12 12 id N8 Online F-Port 21:00:00:1b:32:92:c2:19				64	64	id	N8	Online	F-Port 21:00:00:24:ff:32:e3:db	Redirect: No	
13 13 id N8 Online F-Port 21:01:00:1b:32:b2:c2:19				65	65	id	N8	Online	F-Port 21:00:00:24:ff:32:e3:da	N 030300; 3;21:01:00:1b:32:b2:78:19;20:01:00:1b:32:b2:78:19; na	
14 14 id N8 Online F-Port 21:00:00:1b:32:92:22:18				66	66	id	N8	Online	F-Port 21:00:00:24:ff:32:e3:d5	FC4s: FCP	
15 15 id N8 Online F-Port 21:01:00:1b:32:b2:22:18				67	67	id	N8	Online	F-Port 21:00:00:24:ff:32:e3:d4	PortSymb: [6] "qlt1,0"	
16 16 id N8 Online F-Port 21:00:00:1b:32:92:07:1a				68	68	id	N8	Online	F-Port 21:00:00:24:ff:32:e3:e1	NodeSymb: [7] "dcdata2"	
17 17 id N8 Online F-Port 21:01:00:1b:32:b2:07:1a				69	69	id	N8	Online	F-Port 21:00:00:24:ff:32:e3:e0	Fabric Port Name: 20:03:00:05:1e:9a:8e:be	
18 18 id N8 Online F-Port 21:00:00:24:ff:20:74:46				70	70	id	N8	Online	F-Port 21:00:00:24:ff:32:e3:d7	Permanent Port Name: 21:01:00:1b:32:b2:78:19	
19 19 id N8 Online F-Port 21:00:00:24:ff:20:74:47				71	71	id	N8	Online	F-Port 21:00:00:24:ff:31:5a:1b	Port Index: 3	
20 20 id N8 Online F-Port 21:01:00:1b:32:b2:14:19				72	72	id	N8	Online	F-Port 21:00:00:24:ff:31:5a:1b	Share Area: No	
21 21 id N8 Online F-Port 21:00:00:1b:32:92:14:19				73	73	id	N8	Online	F-Port 21:00:00:24:ff:31:5a:1a	Device Shared in Other AD: No	
22 22 id N8 No_Light				74	74	id	N8	Online	F-Port 21:00:00:24:ff:31:5a:1d	Redirect: No	
23 23 id N8 No_Light				75	75	id	N8	Online	F-Port 21:00:00:24:ff:31:5a:1c	N 030400; 3;21:00:00:24:ff:30:1c:4e;20:00:00:24:ff:30:1c:4e; na	
24 24 id N8 No_Light				76	76	id	N8	Online	F-Port 21:00:00:24:ff:31:5a:23	FC4s: FCP	
25 25 id N8 No_Light				77	77	id	N8	Online	F-Port 21:00:00:24:ff:31:5a:22	PortSymb: [6] "qlt0,0"	
26 26 id N8 No_Light				78	78	id	N8	No_Light		NodeSymb: [7] "dcdata3"	
27 27 id N8 No_Light				79	79	id	N8	No_Light		Fabric Port Name: 20:04:00:05:1e:9a:8e:be	
28 28 id N8 No_Light				Name Server							Permanent Port Name: 21:00:00:24:ff:30:1c:4e
29 29 id N8 No_Light				{						Port Index: 4	
30 30 id N8 No_Light				030000 030100 030200 030300 030400 030500 030600 030700						Share Area: No	
31 31 id N8 No_Light				030800 030900 030c00 030d00 030e00 030f00 031000 031100						Device Shared in Other AD: No	
32 32 id N8 No_Light				031200 031300 031400 031500 034000 034100 034200 034300						Redirect: No	
33 33 id N8 No_Light				034400 034500 034600 034700 034800 034900 034a00 034b00						N 030500; 3;21:00:00:24:ff:30:1c:4f;20:00:00:24:ff:30:1c:4f; na	
34 34 id N8 No_Light				034c00 034d00						FC4s: FCP	
35 35 id N8 No_Light				34 Nx_Ports in the Fabric }						PortSymb: [6] "qlt1,0"	
				{						NodeSymb: [7] "dcdata3"	
				Type Pid COS PortName NodeName TTL(sec)						Fabric Port Name: 20:05:00:05:1e:9a:8e:be	
				N 030000; 3;21:00:00:24:ff:20:7d:b5;20:00:00:24:ff:20:7d:b5; na						Permanent Port Name: 21:00:00:24:ff:30:1c:4f	
										Port Index: 5	
										Share Area: No	
										Device Shared in Other AD: No	

Redirect: No  
 N 030600; 3;21:00:00:24:ff:2f:d1:fc;20:00:00:24:ff:2f:d1:fc; na  
 FC4s: FCP  
 PortSymb: [6] "qlt0,0"  
 NodeSymb: [7] "dcdata4"  
 Fabric Port Name: 20:06:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:24:ff:2f:d1:fc  
 Port Index: 6  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 030700; 3;21:00:00:24:ff:2f:d1:fd;20:00:00:24:ff:2f:d1:fd; na  
 FC4s: FCP  
 PortSymb: [6] "qlt1,0"  
 NodeSymb: [7] "dcdata4"  
 Fabric Port Name: 20:07:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:24:ff:2f:d1:fd  
 Port Index: 7  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 030800; 3;21:00:00:1b:32:92:8f:19;20:00:00:1b:32:92:8f:19; na  
 FC4s: FCP  
 PortSymb: [6] "qlt0,0"  
 NodeSymb: [7] "dcdata5"  
 Fabric Port Name: 20:08:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:1b:32:92:8f:19  
 Port Index: 8  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 030900; 3;21:01:00:1b:32:b2:8f:19;20:01:00:1b:32:b2:8f:19; na  
 FC4s: FCP  
 PortSymb: [6] "qlt1,0"  
 NodeSymb: [7] "dcdata5"  
 Fabric Port Name: 20:09:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:01:00:1b:32:b2:8f:19  
 Port Index: 9  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 030c00; 3;21:00:00:1b:32:92:c2:19;20:00:00:1b:32:92:c2:19; na  
 FC4s: FCP  
 PortSymb: [6] "qlt0,0"  
 NodeSymb: [7] "dcdata7"  
 Fabric Port Name: 20:0c:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:1b:32:92:c2:19  
 Port Index: 12  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 030d00; 3;21:01:00:1b:32:b2:c2:19;20:01:00:1b:32:b2:c2:19; na  
 FC4s: FCP  
 PortSymb: [6] "qlt1,0"  
 NodeSymb: [7] "dcdata7"  
 Fabric Port Name: 20:0d:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:01:00:1b:32:b2:c2:19  
 Port Index: 13

Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 030e00; 3;21:00:00:1b:32:92:22:18;20:00:00:1b:32:92:22:18; na  
 FC4s: FCP  
 PortSymb: [6] "qlt0,0"  
 NodeSymb: [7] "dcdata8"  
 Fabric Port Name: 20:0e:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:1b:32:92:22:18  
 Port Index: 14  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 030f00; 3;21:01:00:1b:32:b2:22:18;20:01:00:1b:32:b2:22:18; na  
 FC4s: FCP  
 PortSymb: [6] "qlt1,0"  
 NodeSymb: [7] "dcdata8"  
 Fabric Port Name: 20:0f:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:01:00:1b:32:b2:22:18  
 Port Index: 15  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 031000; 3;21:00:00:1b:32:92:07:1a;20:00:00:1b:32:92:07:1a; na  
 FC4s: FCP  
 PortSymb: [6] "qlt0,0"  
 NodeSymb: [7] "dcdata9"  
 Fabric Port Name: 20:10:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:1b:32:92:07:1a  
 Port Index: 16  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 031100; 3;21:01:00:1b:32:b2:07:1a;20:01:00:1b:32:b2:07:1a; na  
 FC4s: FCP  
 PortSymb: [6] "qlt1,0"  
 NodeSymb: [7] "dcdata9"  
 Fabric Port Name: 20:11:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:01:00:1b:32:b2:07:1a  
 Port Index: 17  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 031200; 3;21:00:00:24:ff:20:74:46;20:00:00:24:ff:20:74:46; na  
 FC4s: FCP  
 PortSymb: [6] "qlt0,0"  
 NodeSymb: [8] "dcdata10"  
 Fabric Port Name: 20:12:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:24:ff:20:74:46  
 Port Index: 18  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 031300; 3;21:00:00:24:ff:20:74:47;20:00:00:24:ff:20:74:47; na  
 FC4s: FCP  
 PortSymb: [6] "qlt1,0"  
 NodeSymb: [8] "dcdata10"  
 Fabric Port Name: 20:13:00:05:1e:9a:8e:be

Permanent Port Name: 21:00:00:24:ff:20:74:47  
 Port Index: 19  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 031400; 3;21:01:00:1b:32:b2:14:19;20:01:00:1b:32:b2:14:19; na  
 FC4s: FCP  
 PortSymb: [6] "qlt1,0"  
 NodeSymb: [7] "dcdata6"  
 Fabric Port Name: 20:14:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:01:00:1b:32:b2:14:19  
 Port Index: 20  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 031500; 3;21:00:00:1b:32:92:14:19;20:00:00:1b:32:92:14:19; na  
 FC4s: FCP  
 PortSymb: [6] "qlt0,0"  
 NodeSymb: [7] "dcdata6"  
 Fabric Port Name: 20:15:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:1b:32:92:14:19  
 Port Index: 21  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 034000; 3;21:00:00:24:ff:32:e3:db;20:00:00:24:ff:32:e3:db; na  
 FC4s: IPFC FCP  
 Fabric Port Name: 20:40:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:24:ff:32:e3:db  
 Port Index: 64  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 034100; 3;21:00:00:24:ff:32:e3:da;20:00:00:24:ff:32:e3:da; na  
 FC4s: IPFC FCP  
 Fabric Port Name: 20:41:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:24:ff:32:e3:da  
 Port Index: 65  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 034200; 3;21:00:00:24:ff:32:e3:d5;20:00:00:24:ff:32:e3:d5; na  
 FC4s: IPFC FCP  
 Fabric Port Name: 20:42:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:24:ff:32:e3:d5  
 Port Index: 66  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 034300; 3;21:00:00:24:ff:32:e3:d4;20:00:00:24:ff:32:e3:d4; na  
 FC4s: IPFC FCP  
 Fabric Port Name: 20:43:00:05:1e:9a:8e:be  
 Permanent Port Name: 21:00:00:24:ff:32:e3:d4  
 Port Index: 67  
 Share Area: No  
 Device Shared in Other AD: No  
 Redirect: No  
 N 034400; 3;21:00:00:24:ff:32:e3:e1;20:00:00:24:ff:32:e3:e1; na

FC4s: IPFC FCP	Permanent Port Name: 21:00:00:24:ff:31:5a:1c	3,17	
Fabric Port Name: 20:44:00:05:1e:9a:8e:be	Port Index: 75	3,18	
Permanent Port Name: 21:00:00:24:ff:32:e3:e1	Share Area: No	3,19	
Port Index: 68	Device Shared in Other AD: No	3,20	
Share Area: No	Redirect: No	3,21	
Device Shared in Other AD: No	N 034c00; 3;21:00:00:24:ff:31:5a:23;20:00:00:24:ff:31:5a:23; na	3,22	
Redirect: No	FC4s: IPFC FCP	3,23	
N 034500; 3;21:00:00:24:ff:32:e3:e0;20:00:00:24:ff:32:e3:e0; na	Fabric Port Name: 20:4c:00:05:1e:9a:8e:be	3,64	
FC4s: IPFC FCP	Permanent Port Name: 21:00:00:24:ff:31:5a:23	3,65	
Fabric Port Name: 20:45:00:05:1e:9a:8e:be	Port Index: 76	3,66	
Permanent Port Name: 21:00:00:24:ff:32:e3:e0	Share Area: No	3,67	
Port Index: 69	Device Shared in Other AD: No	3,68	
Share Area: No	Redirect: No	3,69	
Device Shared in Other AD: No	N 034d00; 3;21:00:00:24:ff:31:5a:22;20:00:00:24:ff:31:5a:22; na	3,70	
Redirect: No	FC4s: IPFC FCP	3,71	
N 034600; 3;21:00:00:24:ff:32:e3:d7;20:00:00:24:ff:32:e3:d7; na	Fabric Port Name: 20:4d:00:05:1e:9a:8e:be	3,72	
FC4s: IPFC FCP	Permanent Port Name: 21:00:00:24:ff:31:5a:22	3,73	
Fabric Port Name: 20:46:00:05:1e:9a:8e:be	Port Index: 77	3,74	
Permanent Port Name: 21:00:00:24:ff:32:e3:d7	Share Area: No	3,75	
Port Index: 70	Device Shared in Other AD: No	3,76	
Share Area: No	Redirect: No	3,77	
Device Shared in Other AD: No	The Local Name Server has 34 entries }	zone: REDO_Nodes	3,40
Redirect: No	Zoning Information		3,41
N 034700; 3;21:00:00:24:ff:32:e3:d6;20:00:00:24:ff:32:e3:d6; na	Defined configuration:		3,42
FC4s: IPFC FCP	cfg: dc120f_initial_ewt		3,43
Fabric Port Name: 20:47:00:05:1e:9a:8e:be	zone: Data_Nodes; REDO_Nodes		3,78
Permanent Port Name: 21:00:00:24:ff:32:e3:d6	zone: Data_Nodes	3,0; 3,1; 3,2; 3,3; 3,4; 3,5; 3,6; 3,7; 3,8; 3,9; 3,10; 3,11;	3,79
Port Index: 71		3,12; 3,13; 3,14; 3,15; 3,16; 3,17; 3,18; 3,19; 3,20;	
Share Area: No			SFP Serial ID Information
Device Shared in Other AD: No	3,21;	3,22; 3,23; 3,64; 3,65; 3,66; 3,67; 3,68; 3,69; 3,70;	Port 0: id (sw) Vendor: BROCADE
Redirect: No			Speed: 200,400,800_MB/s
N 034800; 3;21:00:00:24:ff:31:5a:1b;20:00:00:24:ff:31:5a:1b; na			Serial No: UAF109100000BFH
FC4s: IPFC FCP	3,71;	3,72; 3,73; 3,74; 3,75; 3,76; 3,77	Port 1: id (sw) Vendor: BROCADE
Fabric Port Name: 20:48:00:05:1e:9a:8e:be			Speed: 200,400,800_MB/s
Permanent Port Name: 21:00:00:24:ff:31:5a:1b	zone: REDO_Nodes	3,40; 3,41; 3,42; 3,43; 3,78; 3,79	Serial No: UAF109100000BER
Port Index: 72			Port 2: id (sw) Vendor: BROCADE
Share Area: No			Speed: 200,400,800_MB/s
Device Shared in Other AD: No			Serial No: UAF109100000BEV
Redirect: No	Effective configuration:		Port 3: id (sw) Vendor: BROCADE
N 034900; 3;21:00:00:24:ff:31:5a:1a;20:00:00:24:ff:31:5a:1a; na	cfg: dc120f_initial_ewt		Speed: 200,400,800_MB/s
FC4s: IPFC FCP	zone: Data_Nodes		Serial No: UAF109100000BFN
Fabric Port Name: 20:49:00:05:1e:9a:8e:be		3,0	Port 4: id (sw) Vendor: BROCADE
Permanent Port Name: 21:00:00:24:ff:31:5a:1a		3,1	Speed: 200,400,800_MB/s
Port Index: 73		3,2	Port 5: id (sw) Vendor: BROCADE
Share Area: No		3,3	Speed: 200,400,800_MB/s
Device Shared in Other AD: No		3,4	Serial No: UAF109100000BFC
Redirect: No		3,5	Port 6: id (sw) Vendor: BROCADE
N 034a00; 3;21:00:00:24:ff:31:5a:1d;20:00:00:24:ff:31:5a:1d; na		3,6	Speed: 200,400,800_MB/s
FC4s: IPFC FCP		3,7	Serial No: UAF109100000BEU
Fabric Port Name: 20:4a:00:05:1e:9a:8e:be		3,8	Port 7: id (sw) Vendor: BROCADE
Permanent Port Name: 21:00:00:24:ff:31:5a:1d		3,9	Speed: 200,400,800_MB/s
Port Index: 74		3,10	Serial No: UAF1091100004HV
Share Area: No		3,11	Port 8: id (sw) Vendor: BROCADE
Device Shared in Other AD: No		3,12	Speed: 200,400,800_MB/s
Redirect: No		3,13	Port 9: id (sw) Vendor: BROCADE
N 034b00; 3;21:00:00:24:ff:31:5a:1c;20:00:00:24:ff:31:5a:1c; na		3,14	Speed: 200,400,800_MB/s
FC4s: IPFC FCP		3,15	Serial No: UAF1091100002FU
Fabric Port Name: 20:4b:00:05:1e:9a:8e:be		3,16	Port 10: id (sw) Vendor: BROCADE
			Speed: 200,400,800_MB/s
			Serial No: UAF109100000A3T
			Port 11: id (sw) Vendor: BROCADE
			Speed: 200,400,800_MB/s
			Serial No: UAF10911000058P
			Port 12: id (sw) Vendor: BROCADE
			Speed: 200,400,800_MB/s
			Serial No: UAF109100000AN3










## Appendix D: Third Party Pricing



800.800.4239

## Shopping Cart

Item	Quantity	Availability	Unit Price	Item Total
 <p><b>Brocade expansion module</b>  MFG Part#: FLS-1XG  CDW Part#: 1285843  UNSPSC: 43201404  <b>Pricing Option Applied:</b> Extranet Price</p>	1	<u>1-3 days</u>	\$849.99	\$849.99
 <p><b>Tripp Lite 25' Black Cat5e or Cat5 RJ45 Molded 350mhz UTP Patch Cable</b>  MFG Part#: N002-025-BK  CDW Part#: 324521  UNSPSC: 26121604  <b>Pricing Option Applied:</b> Extranet Price</p>	103	<u>In Stock</u>	\$6.99	\$719.97
 <p><b>NETGEAR ProSafe Plus JGS524E 24-port Gigabit Ethernet Switch - switch - 24</b>  MFG Part#: JGS524E-100NAS  CDW Part#: 2352401  UNSPSC: 43222612  <b>Pricing Option Applied:</b> Extranet Price</p>	3	<u>In Stock</u>	\$278.99	\$836.97
 <p><b>Brocade FastIronGS 648P</b>  MFG Part#: FGS648P  CDW Part#: 978924  UNSPSC: 43222612  <b>Pricing Option Applied:</b> Extranet Price</p>	1	<u>1-3 days</u>	\$4,976.99	\$4,976.99
 <p><b>Brocade Essential Direct Support 4-Hour Parts - extended service agreement</b>  MFG Part#: FGS48-SVL-R4P-1  CDW Part#: 2019294  UNSPSC: 81111812  <b>Pricing Option Applied:</b> Extranet Price</p>	3	<u>Call</u>	\$1,222.99	\$3,668.97


**Acer V173 DJb 17" LCD**

MFG Part#: ET.BV3RP.D03

CDW Part#: 2091793

UNSPSC: 43211902

Pricing Option Applied: Extranet Price

3

In Stock

\$101.99

\$305.97

**Subtotal: \$11,358.86**

Tax and shipping will be calculated in checkout.

[Lease Option](#) (\$354.40 /month)

**Items Related to Products in Your Cart**

 CDW 7' CAT5e or CAT5 RJ45 Patch  
Cable Black

 Only  
**\$2.99**

 Tripp Lite 50' Black Cat5e or Cat5  
RJ45 Molded 350Mhz UTP Patch  
Cable

 Only  
**\$11.99**

 CDW 14' CAT5e or CAT5 RJ45 Patch  
Cable Black

 Only  
**\$4.99**

 Kingston DataTraveler I G3 - USB  
flash drive - 4 GB

 Only  
**\$6.99**
**Customers Who Bought Products in Your Cart Also Bought...**

 CDW 3' CAT5e or CAT5 RJ45 Patch  
Cable Blue

 Only  
**\$1.99**

 Belkin RJ45 CAT5e Round Modular  
Plug, 100 Pack

 Only  
**\$37.99**

 ViewSonic VA2231wm-LED - LED  
monitor - 22"

 Only  
**\$142.99**

 NETGEAR FS108 8-Port Fast Ethernet  
Switch

 Only  
**\$47.99**

This page was printed on 3/23/2012 12:48:30 PM.