TPC Express AI

TPCx-AI

Draft Specification

Version 0.6

March 2021

Transaction Processing Performance Council (TPC)

www.tpc.org
info@tpc.org

Legal Notice

The TPC reserves all right, title, and interest to this document and associated source code as provided under U.S. and international laws, including without limitation all patent and trademark rights therein.

Permission to copy without fee all or part of this document is granted provided that the TPC copyright notice, the title of the publication, and its date appear, and notice is given that copying is by permission of the Transaction Processing Performance Council. To copy otherwise requires specific permission.

No Warranty

TO THE MAXIMUM EXTENT PERMITTED BY APPLICABLE LAW, THE INFORMATION CONTAINED HEREIN IS PROVIDED "AS IS" AND WITH ALL FAULTS, AND THE AUTHORS AND DEVELOPERS OF THE WORK HEREBY DISCLAIM ALL OTHER WARRANTIES AND CONDITIONS, EITHER EXPRESS, IMPLIED OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, ANY (IF ANY) IMPLIED WARRANTIES, DUTIES OR CONDITIONS OF MERCHANTABILITY, OF FITNESS FOR A PARTICULAR PURPOSE, OF ACCURACY OR COMPLETENESS OF RESPONSES, OF RESULTS, OF WORKMANLIKE EFFORT, OF LACK OF VIRUSES, AND OF LACK OF NEGLIGENCE. ALSO, THERE IS NO WARRANTY OR CONDITION OF TITLE, QUIET ENJOYMENT, QUIET POSSESSION, AND CORRESPONDENCE TO DESCRIPTION OR NON-INFRINGEMENT WITH REGARD TO THE WORK.

IN NO EVENT WILL ANY AUTHOR OR DEVELOPER OF THE WORK BE LIABLE TO ANY OTHER PARTY FOR ANY DAMAGES, INCLUDING BUT NOT LIMITED TO THE COST OF PROCURING SUBSTITUTE GOODS OR SERVICES, LOST PROFITS, LOSS OF USE, LOSS OF DATA, OR ANY INCIDENTAL, CONSEQUENTIAL, DIRECT, INDIRECT, OR SPECIAL DAMAGES WHETHER UNDER CONTRACT, TORT, WARRANTY, OR OTHERWISE, ARISING IN ANY WAY OUT OF THIS OR ANY OTHER AGREEMENT RELATING TO THE WORK, WHETHER OR NOT SUCH AUTHOR OR DEVELOPER HAD ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

Trademarks

TPC Benchmark is a trademark of the Transaction Processing Performance Council.

Product names, logos, brands, and other trademarks featured or referred to within this Specification are the property of their respective trademark holders.

Document Revision History

Date	Version	**Description**
2/17/2021	0.5	Pre-Formal Review Version.
3/1/2021	0.6	Formal Review

TABLE OF CONTENTS

0 PREAMBLE

0.1 Introduction

Artificial intelligence (AI) has become a key transformational technology of our times. Advances in neural networks and other machine learning techniques have made it possible to use AI on a variety of use cases. From the public sector to aerospace, defense and academia, new and improved ways to use AI techniques are changing the way we harness data and analytics. This along with advances in compute, interconnect and memory technologies have made possible to solve complicated challenges that will ultimately benefit customers in production datacenter and cloud environments.

Abundant volumes of rich data from text, images, audio and video are the essential starting point for creating a benchmark that would represent the myriad of use cases and customers. Our attempt at the TPC is to begin capturing a subset of the customer scenarios with a continued focus on adding to the use cases as well as refreshing the benchmark with newer AI and machine learning techniques that solve challenges for relevant current production data science pipelines.

TPC Benchmark AI (TPCx-AI) is an AI benchmark that models several aspects of an AI or machine learning data science pipeline using a diverse dataset that includes images, text and audio. The benchmark defines and provides a means to evaluate the System Under Test (SUT) performance as a general-purpose data science system.

This benchmark illustrates AI or machine learning data science systems that:

- Generate and process large volumes of data.
- Train preprocessed data to produce realistic machine learning models
- Conduct accurate insights for real-world customer scenarios based on the generated models
- Can scale to large scale distributed configurations
- Allow for flexibility in configuration changes to meet the demands of the dynamic AI landscape.

The benchmark models real-life examples of companies and public-sector organizations that use a range of analytics techniques, both AI and more traditional machine learning approaches, as well as the potential application of these techniques in situations like those in which they have already been successfully deployed. In addition, the benchmark measures end to end time to insights for individual **use cases,** as well as throughput metrics to simulate multiuser environments for a given hardware, operating system, and data processing system configuration under a controlled, complex, multi-user AI or machine learning data science workload.

Comment: While separated from the main text for readability, comments and appendices are a part of the standard and their provisions must be enforced.

0.2 TPCx-AI Kit and Licensing

The TPCx-AI kit is available from the TPC website (see www.tpc.org/tpcx-ai/ for more information). Users must sign-up and agree to the TPCx-AI End User Licensing Agreement (EULA) to download the kit. All related work (such as collaterals, papers, derivatives) must acknowledge the TPC and include the TPCx-AI copyright. The TPCx-AI kit includes: TPCx-AI Specification document (this document), TPCx-AI Users Guide (README.md) documentation, scripts to set up the benchmark environment, code to execute the benchmark workload, Data Generator, use case related files, and Benchmark Driver.

0.3 General Implementation Guidelines

The purpose of TPC benchmarks is to provide relevant, objective performance data to industry users. To achieve that purpose, TPC benchmark specifications require benchmark tests be implemented with systems, products, technologies and pricing that:

a) Are generally available to users.
b) Are relevant to the market segment that the individual TPC benchmark models or represents (e.g., TPCx-AI models and represents complex, high data volume, decision support environments).
c) Would plausibly be implemented by a significant number of users in the market segment modeled or represented by the benchmark.

In keeping with these requirements, the TPCx-AI dataset must be implemented using commercially available software.

The use of new systems, products, technologies (hardware or software) and pricing is encouraged so long as they meet the requirements above. Specifically prohibited are benchmark systems, products, technologies, or pricing (hereafter referred to as "implementations") whose primary purpose is performance optimization of TPC benchmark results without any corresponding applicability to real-world applications and environments. In other words, all "benchmark special" implementations, which improve benchmark results but not real-world performance or pricing, are prohibited.

Several characteristics shall be evaluated to judge whether a particular implementation is a benchmark special. It is not required that each point below be met, but that the cumulative weight of the evidence be considered to identify an unacceptable implementation. Absolute certainty or certainty beyond a reasonable doubt is not required to make a judgment on this complex issue. The question that must be answered is: "Based on the available evidence, does the clear preponderance (the greater share or weight) of evidence indicate this implementation is a benchmark special?"

The following characteristics shall be used to judge whether an implementation is a benchmark special:

a) Is the implementation generally available, documented, and supported?

b) Does the implementation have significant restrictions on its use or applicability that limits its use beyond TPC benchmarks?

c) Is the implementation or part of the implementation poorly integrated into the larger product?

d) Does the implementation take special advantage of the limited nature of TPC benchmarks (e.g., model selection, accuracy requirements, query templates etc.) in a manner that would not be generally applicable to the environment the benchmark represents?

e) Is the use of the implementation discouraged by the vendor? (This includes failing to promote the implementation in a manner like other products and technologies.)

f) Does the implementation require uncommon sophistication on the part of the end-user, programmer, or system administrator?

g) Is the pricing unusual or non-customary for the vendor or unusual or non-customary compared to normal business practices? The following pricing practices are suspect:

- Availability of a discount to a small subset of possible customers.
- Discounts documented in an unusual or non-customary manner.
- Discounts that exceed 25% on small quantities and 50% on large quantities.
- Pricing featured as a close-out or one-time special.
- Unusual or non-customary restrictions on transferability of product, warranty, or maintenance on discounted items.

h) Is the implementation (including beta-release components) being purchased or used for applications in the market segment the benchmark represents? How many sites implemented it? How many end-users benefit from it? If the implementation is not currently being purchased or used, is there any evidence to indicate that it will be purchased or used by a significant number of end-user sites?

0.4 General Measurement Guidelines

TPCx-AI **Results** are expected to be accurate representations of system performance. Therefore, there are certain guidelines that are expected to be followed when measuring those **Results**. The approach or methodology to be used in the measurements are either explicitly described in the specification or implemented by the TPCx-AI Kit (Clause 5.1). When not described in the specification, the methodologies and approaches used must meet the following requirements:

- The approach is an accepted engineering practice or standard.
- The approach does not enhance the **Results**.
- The equipment used in measuring **Results** must conform to the requirements in….

- Fidelity and candor are maintained in reporting any anomalies in the **Results**, even if not specified in the benchmark requirements.

Comment: The use of new methodologies and approaches is encouraged so long as they meet the requirements above.

0.5 Definitions

- **Attestation Letter**

TPC-Certified Auditor's opinion regarding the compliance of a Result must be consigned in an **Attestation Letter** delivered directly to the Test Sponsor.

- **Availability Date**

The Availability Date is the System Availability Date defined in the TPC Pricing Specification.

B _____

- **Benchmark Special**

The Benchmark Special is defined as any aspect of the benchmark implementation with the primary purpose of the optimization of TPC Benchmark Results without any corresponding applicability to real-world applications and environments.

C _____

- **Commercially Available Product**

Commercially Available Product is defined in TPC Pricing Specification.

- **Compute Software**

Compute software runs on Compute Hardware providing required software capabilities to successfully execute the benchmark. Compute Software unless otherwise stated will be part of the System Under Test.

D _____

- **Data Redundancy**

The ability to have no permanent data loss after the permanent irrecoverable failure of any single Durable Medium containing tables, input data, output data, or metadata.

- **Data Generation**

The process of using PDGF to create the data in a format suitable for presentation to the load facility.

- **Data Node**

Data Nodes store data in a Hadoop cluster and is the name of the daemon that manages the data. File data is replicated on multiple Data Nodes for reliability and so that localized computation can be executed near the data[1].

- **Deep Learning**

[1] https://docs.cloudera.com/documentation/enterprise/6/6.3/topics/cm_mc_dn.html

Deep learning (also known as **deep structured learning**) is part of a broader family of machine learning methods based on artificial neural networks with representation learning[2].

E _____

- **Executive Summary**

Defined by the TPC Policies, an Executive Summary is a two to four-page summary of the Result.

F _____

- F-score

The **F-score** or **F-measure** is a measure of a test's accuracy. It is calculated from the precision and recall of the test, where the precision is the number of correctly identified positive results divided by the number of all positive results, including those not identified correctly, and the recall is the number of correctly identified positive results divided by the number of all samples that should have been identified as positive[3].

- **Full Disclosure Report (FDR)**

The Full Disclosure Report is a set of files that documents how a benchmark Result was implemented and executed in sufficient detail so that the Result can be reproduced given the appropriate hardware and software products.

- **Framework**

A Framework is a collection of software including API's, distributed computing engines, Machine learning, AI and libraries used to run TPCx-AI.

G _____

H _____

- **HDFS**

HDFS (Hadoop Distributed File System) is a file system that provides scalable and reliable data storage, and it was designed to span large clusters of commodity servers.

- **High Availability System**

Computing environments configured to provide nearly full-time availability are known as High Availability Systems. Such systems typically have redundant hardware and software that makes the system available despite failures. Well-designed high availability systems avoid having single points-of-failure. Any hardware or software component that can fail has a redundant component of the same type.

I _____

J _____

[2] https://en.wikipedia.org/wiki/Deep_learning

[3] https://en.wikipedia.org/wiki/F-score

- **JBOD**

JBOD (Just a Bunch of Disks) refers to a collection of hard disks that have not been configured to act as a redundant array of independent disks (RAID) array.

K _____

- **k-means**

k-means clustering is a method of vector quantization, originally from signal processing, that aims to partition n observations into k clusters in which each observation belongs to the cluster with the nearest mean (cluster centers or cluster centroid), serving as a prototype of the cluster.[4]

L _____

- **LCS (Licensed Compute Services)**

Licensed Compute Service (LCS) is defined in TPC Pricing Specification: Publicly offered processing, storage, network, and software services that are hosted on remote computer servers accessed via a Wide Area Network (e.g. the Internet). A Customer pays a license fee to the Licensed Compute Services vendor for the use of the processing, storage, network, and software services. The Licensed Compute Services are not located or installed on a Customer's premises.

M _____

- **Machine Learning**

- **Metastore/Metadata**

Descriptive information about the dataset including names and definitions of tables, indexes, and other schema objects. Various terms commonly used to refer collectively to the Metadata include Metastore, information schema, data dictionary, or system catalog. Metadata also includes additional information managed by the BDAS and stored in the dataset to define, manage and use other objects, e.g. users, connections, etc.

- **Master Node**

Master Node(s) provide a variety of storage and processing coordination services for a cluster. These are conceptually distinct from Data Nodes but sometimes share physical hardware. Where necessary for correct execution of the benchmark, data in the Master Node services is considered Metadata unless it is data for a co-located Data Node service in which case it is considered table data. Some Master Node services can be configured with sufficient instances as part of a High Availability System while others require other approaches to protecting against loss of service or data loss. Examples of Master Node services include Name Nodes, Checkpoint Nodes, Journal Nodes and Resource Manager

- **Metadata**

N _____

- **Name Node**

A Name Node is a particular class of Master Node service. *Name Nodes maintain the namespace tree for HDFS and a mapping of file blocks to Data Nodes where the data is stored. A simple HDFS cluster can have only one primary Name Node, supported by a secondary Name Node*[5]

[4] k-means clustering - Wikipedia

[5] https://docs.cloudera.com/documentation/enterprise/6/6.3/topics/cm_mc_nn.html

O _____

- **Operating System/OS**

The term **Operating System** refers to a commercially available program that, after being initially loaded into the computer by a boot program, manages all the other programs in a computer, or in a VM. The **Operating System** provides a software platform on top of which all other programs run. Without the **Operating System** and the core services that it provides no other programs can run and the computer would be non-functional. Other programs make use of the **Operating System** by making requests for services through a defined application program interface (API). All major computer platforms require an **Operating System**. The functions and services supplied by an **Operating System** include but are not limited to the following:

a) manages a dedicated set of processor and memory resources
b) maintains and manages a file system
c) loads applications into memory
d) ensures that the resources allocated to one application are not used by another application in an unauthorized manner

determines which applications should run in what order, and how much time should be allowed to run the application before giving another application a turn to use the systems resources

manages the sharing of internal memory among multiple applications

handles input and output to and from attached hardware devices such as hard disks, network interface cards, addon cards and other hardware devices.

Some examples of **Operating Systems** are listed below:

a) Windows
b) Unix (Solaris, AIX)
c) Linux (Red Hat, SUSE)
d) Mac OS

P _____

- **PDGF**
  The PDGF (Parallel Data Generator Framework) is part of TPCx-AI kit used to generate the **Test Dataset**.

- **Performance Metric**

The reported throughput as expressed in AI use cases per minute.

- **Performance Run**

The Performance Run is defined as the run following the validation test that consists of the 6 tests running in sequential order

- **Priced Configuration**

The Priced Configuration consists of components defined in the TPCx-AI Benchmark Standard including all hardware, software and maintenance.

- **Price/Performance Metric**

The Price/Performance Metric is the total price of the Priced Configuration divided by the TPCx-AI Performance Metric.

Q _____

R _____

- **Report**

The **Report** is an Adobe Acrobat PDF file in the **FDR**. The contents of the Report are defined in Clause 5.3

- **Reported**

The term Reported an item that is part of the FDR.

- **Result**

A performance test, documented by a Full Disclosure Report and Executive Summary submitted to the TPC, claiming to meet the requirements of the TPCx-AI Benchmark Standard.

S _____

- **Scoring**

The phase where the accuracy metric of the generated model is calculated for a given use case

- **Serving**

The phase where the generated model is used to predict results for a given use case

- **Software Version**

A Software Version uniquely identifies a software product, its release level, update level, and/or patch level. It is typically a string of alphanumeric characters that allows the software manufacturer to uniquely identify the software.

- **Substitution**

Substitution is the use of components in the Priced Configuration which are different than those used in the measured configuration.

- **Supporting Files**

Supporting Files refers to the contents of the Supporting Files folder in the FDR. The contents of this folder, consisting of various source files, scripts, and listing files, are defined in Clause 9.1.1 .

- **System Under Test (SUT)**

System Under Test (SUT**)** – is defined to be the sum of the components utilized in running a benchmark as specified in Clause 6.

T _____

- **Test Sponsor**

The **Test Sponsor** is the company officially submitting the Result with the FDR and will be charged the filing fee. Although multiple companies may sponsor a Result together, for the purposes of the TPC's processes the **Test Sponsor** must be a single company. A **Test Sponsor** need not be a TPC member. The **Test Sponsor** is responsible for maintaining the FDR with any necessary updates or corrections. The **Test Sponsor** is also the name used to identify the Result.

- **Test Dataset**

The Test Dataset is the dataset generated by **PDGF** for the defined scale factor used to execute the Load test, Power tests and Throughput test.

- **TPCx-AI approved Compute software**

Compute software and libraries that are not commercially available that cannot meet the pricing requirements that the TPCx-AI subcommittee has approved to be used as part of the benchmark kit configuration for a valid publication.

- **TPC-Certified Auditor (Auditor)**

The term **TPC-Certified Auditor** is used to indicate that the TPC has reviewed the qualification of the Auditor and has certified his/her ability to verify that benchmark **Results** are in compliance with a specification. (Additional details regarding the Auditor certification process and the audit process can be found in Section 9 of the TPC Policies document.)

- **Training**

The phase where the generated model is used to predict results for a given use case

U _____

- **Undo/Redo Log**

Undo/Redo Log: records all changes made in data files. The Undo/Redo Log makes it possible to replay all the actions executed by the BDAS. If something happens to one of the data files, a backed up data file can be restored and the Undo/Redo Log that was written since the backup can be played and applied which brings the data file to the state it had before it became unavailable. Not all BDAS environments utilize an Undo/Redo Log to accommodate recovery.

- **Use Case**

A Use Case defines a single problem solved by the AI and Machine learning Data Science Pipeline. It is Framework and syntax agnostic and can be implemented in many ways. In the TPCx-AI kit, all Use Cases implemented include data generation, data management, training, scoring and serving phases.

V _____

W _____

X _____

Y _____

Z _____

1 BUSINESS AND BENCHMARK MODEL

1.1 Overview

TPC Benchmark AI (TPCx-AI) contains benchmark components that can be used to assess a broad range of system topologies and implementation methodologies in a technical rigorous and directly comparable, vendor-neutral manner. The benchmark has been mapped to typical retail businesses & on-premise, cloud and **edge** environments. This clause outlines the business modeling assumptions that were adopted during the development of the benchmark, and their impact on the benchmarking environments.

TPCx-AI models the end to end AI and machine learning data science system of a retail business datacenter. The supporting schema contains vital business information, such as customer, order, financial and product data. The benchmark models the two most important components of any mature data science system:

- Data aggregation and data management, which converts the data to relevant **Test datasets** for processing
- Insights, which transforms the relevant data sets into accurate business intelligence

The benchmark abstracts the diversity of operations in data science pipelines, while retaining essential performance characteristics.

In a retail datacenter scenario, it is necessary to execute a diverse number of AI or machine learning data science use cases based on various departments that may or may not be related within the datacenter. This poses an additional challenge of trying to benchmark scenarios and business intelligence pipelines that would benefit the end user. As a results TPCx-AI comprises of many independent AI and machine learning use cases that facilitate in helping any retail business datacenter address and manage any business analysis environment.

We define a "**use case (s)**" as a targeted end to end pipeline of processes and technologies applied to a specific business challenge, with a measurable outcome. We recognize that use cases can be described at different levels of granularity. The use cases we chose for this benchmark correspond to descriptions of specific business challenges which industry experts acknowledged to the TPC as meaningful. For example, recommending the "next product to buy" or "detecting a fraudulent transaction" for e-commerce in the retail industry were considered examples of very important use cases. Higher level functions like "marketing" or "sales" were not sufficiently granular for the TPC to be considered use cases, even though they have relevance in the overall business development process.

While TPCx-AI does not aspire to be a model of how to build actual AI or machine learning data science pipelines, the benchmark has been granted a realistic context. It imitates the activity of retail businesses and datacenters with customer information, department stores, sales and financial data, product catalog and reviews, emails, datacenter logs as well as facial images and audio conversations.

The goal of selecting a retail business model is to assist the reader in relating intuitively to the components of the benchmark, without tracking that industry segment so tightly as to minimize the relevance of the benchmark. The TPCx-AI benchmark be used to characterize any industry that must transform operational and external data into business intelligence. The data represents a reasonable image of a business operation as they progress over time.

The TPCx-AI benchmark models the challenges of end to end artificial intelligence systems and pipelines where the power of machine learning and deep learning is used to: detect anomalies (fraud and failures), drive AI based logistics optimizations to reduce costs through real-time forecasts (classification, clustering, forecasting and prediction) and use **deep learning** AI techniques for customer service management and personalized marketing (face recognition and speech recognition).

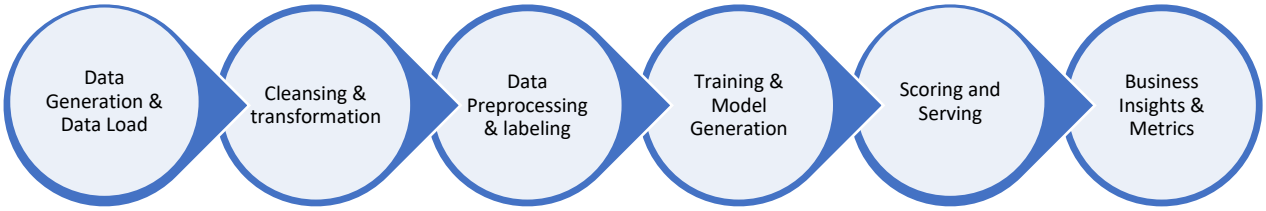


Figure 1-a Benchmark Use Case pipeline flow

1.2 Business model

TPCx-AI benchmark models the following use cases:

- Classifying categories and trip types using data from existing customer shopping trips
- Improve cross-selling by giving "next-product-to-buy" recommendations. Based on previously bought products recommend products that the customer might also be interested in. Those recommendation are found by comparing customers (by their products) and/ or products (by their customers)
- Suggest a price of an item based on its brand, product name, and description on an online marketplace.
- Find comments, reviews, or descriptions of items in a retail business with spam content. The problem to be solved is to identify those reviews that are spam
- Find customer segments based on their behavior. Clustering/segmentation of customers based on return behavior (return frequency, return/order ratio, …) and buying behavior (frequency of purchases, recency of purchases, …)
- Detect if a given financial transaction is fraudulent or not.
- Predict imminent hardware failure, given existing logs of hardware events.
- Forecast the weekly sales for each store department and each store of a retail chain with multiple stores based on a limited history of sales data.
- Accurately recognize customer facial images
- Accurately transcript audio conversations of customers to text

1.3 Data model

The TPCx-AI benchmark models a unified dataset schema of a retail datacenter with a full **AI** and **machine learning data science pipeline**. This environment allows a single as well as concurrent data science pipelines to be run at any given point in time. The data rests in persistent storage so is always available. The TPCx-AI dataset represents a system that contains structured, semi-structured as well as unstructured data which is typical of modern data systems. Some examples of structured data are Product, orders, customers, financial transactions tables. Customer images and audio conversations make up the unstructured dataset.

The Data generation phase of the benchmark scales each of the tables and unstructured datasets to simulate the growth of the datacenter data sizes. The hope of the benchmark is to mimic datasets of different company sizes that each would have the optimal data science pipelines (benchmark scaling or scale factor). Once generated, all the data is processed for subsequent stages of post-processing within the machine learning data science pipeline.

1.4 TPCx-AI Data Science Pipeline

1.4.1 Data Science Pipeline Assumptions

The **data management** stages, and data science pipeline modeled by the benchmark exhibit the following characteristics:

a) They address complex business problems with the intent to answer specific questions

b) The data is acquired from different sources and contains different formats
c) The data management stages like cleansing, exploration & preprocessing mimic modern commercial pipelines used in current production environments
d) They employ **training**, **serving** and **scoring** phases using production datasets available in the datacenter

To address the enormous range of use cases prevalent in today's modern data science pipelines, TPCx-AI utilizes a generalized data science pipeline model. This model allows the benchmark to capture important aspects of the pipeline where data scientists would train on preprocessed data, develop a production model, perform a scoring analysis to generate accuracy metrics and then use the built model in the serving phases with production quality data.

1.4.2 Data Generation

This phase of the benchmark models data acquisition wherein a built-in data generator (**PDGF**) provided by the TPC is used to generate 'relevant' data that will be used to answer specific business related questions the retail datacenter is trying to answer. The data is a combination of structured, semi-structured and unstructured data formats.

1.4.3 Data Loading

Data loading takes existing raw data generated by the TPC provided **PDGF** data generator and loads it into structured schema formats, or persistent storage (e.g. HDFS, or other file systems). Some data science pipelines also call this stage Data Wrangling.

1.4.4 Cleansing and Transformation

Ensuring data is cleaned or cleansed can offer significant business value. Numerous surveys have found that many large enterprises do not use data effectively due to the complexity of the data and redundancies thereof. Cleaning or Cleansing the data can significantly help achieve a long list of benefits including accurate insights, increased productivity, and efficient pipeline execution. Examples of cleansing are included but not limited to removal of null records and duplicate records from the **Test dataset**.

1.4.5 Data preprocessing and labeling

This phase involves deep study of the data, columns or patterns and inter-relationships between the entities and variables. This analysis also helps in eliminating irrelevant data and narrowing down on the key parameters needed to build an effective model. Examples of preprocessing include normalizing and standardizing the cleansed data and labeling for getting the **Test dataset** ready for building the model.

1.4.6 Training and Model Generation

Training usually results in the generation of a model. This phase involves many stages like choosing one or many machine learning or AI methods to choose from, generating models and re-iterating the processing by changing hyper parameters to build the final model. As in real world scenarios, the TPCx-AI kit models the training phase using one representative machine learning or AI model that will meet key accuracy criteria set forth by the benchmark. The benchmark assumes that this model will be deployed in production by the datacenter to be used for subsequent serving phases. Each unique use case will represent its own unique model training phase in the benchmark.

1.4.7 Scoring

This stage may also be known as the model validation stage. Scoring in the benchmark involves taking the generated model and validating its accuracy with a labeled dataset (new dataset generated by the **PDGF** data generator tool) not used for training so that the accuracy of the model can be determined. Each use case pipeline in the benchmark models its unique scoring phase. The benchmark also defines accuracy metrics and criteria that need to be met to have a successful **run**.

1.4.8 Model Deployment and Serving

Real world production model deployment and machine learning inference is represented by the TPCx-AI benchmark by deploying only one production model per use case. The TPCx-AI benchmark uses the model generated in the training phases to use conduct the serving phase for each of the respective use cases. For each use case, a new dataset is generated especially for the serving phase to represent production data that needs quality business insights.

1.4.9 Business insights and Metrics

TPCx-AI metrics demonstrate the value obtained using the end to end data science pipeline execution. The results from the benchmark model the powerful use of insights that can be used to grow the retail business, predict failure rates or anomalies, accurately conduct classification or marketing personalization.

2 LOGICAL DATASET DESIGN

2.1 Dataset Schema Overview

The TPCx-AI dataset comprises of the retail datacenter for an organization that contains 10 structured tables, 2 semi-structured table and 2 unstructured datasets. Figure 2-a shows the structured, semi-structured and unstructured datasets, the high-level definitions for each table and its relationship to other tables in the form of a relationship diagram.

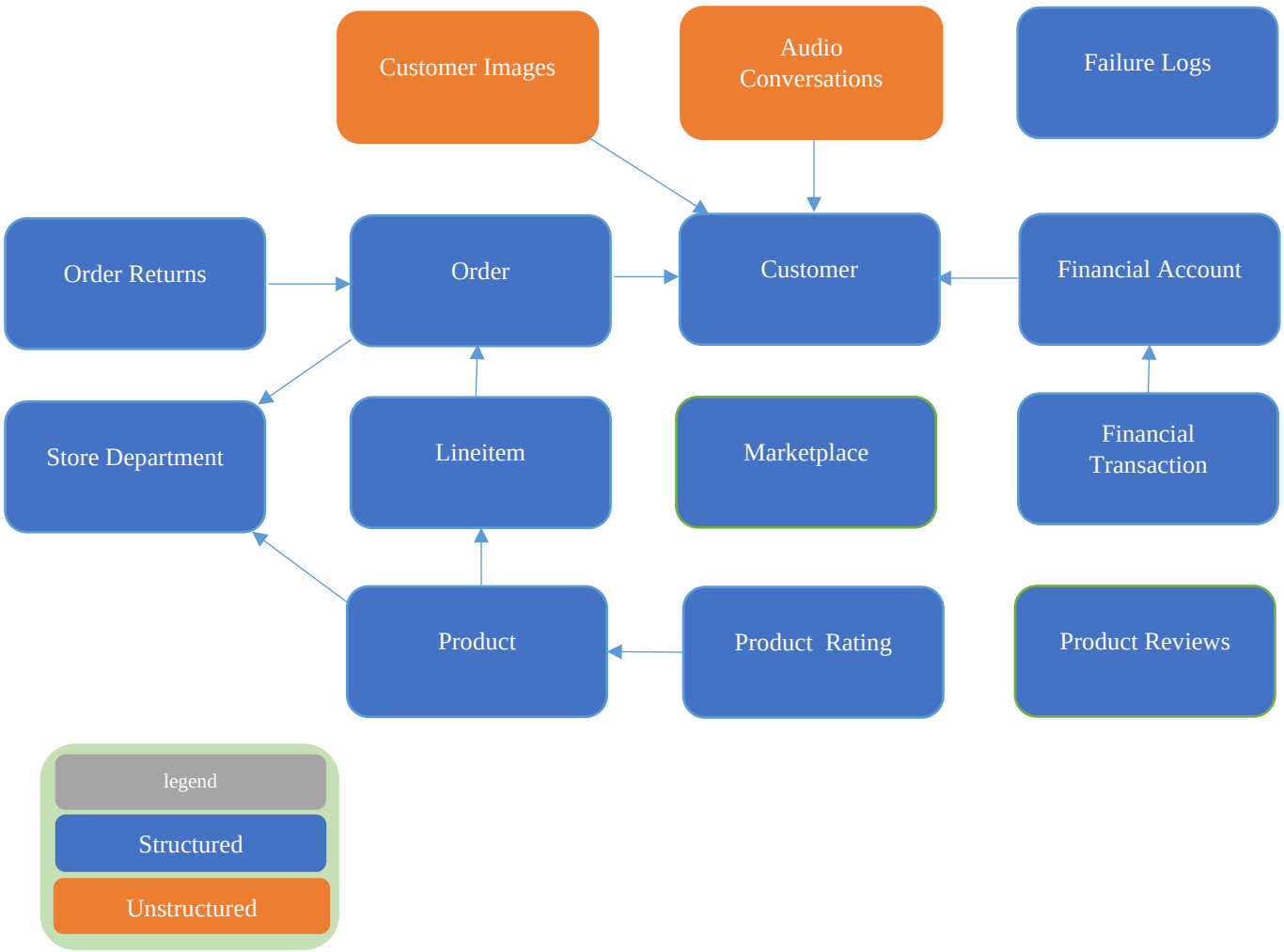


Figure 2-a – Dataset Schema

2.2 Dataset & Table Descriptions

2.2.1 Table description

2.2.1.1 Describes the high-level dataset or table

2.2.2 Column Definitions

2.2.2.1 Column Name

Each column is uniquely named. Columns that are part of the table's primary key are indicated in the column called Primary Key. If a table uses a composite primary key, then for convenience of reading the order of a given column in a table's primary key is listed in parentheses following the column name.

2.2.2.2 Datatype

Each column employs one of the following datatypes:

a) Identifier means that the column shall be able to hold any key value generated for that column.
b) INTEGER means that the column shall be able to exactly represent INTEGER values (i.e., values in increments of 1) in the range of at least (-2^{n-1}) to $(2^{n-1}-1)$, where n is 64.
c) Decimal(d, f) means that the column shall be able to represent decimal values up to and including d digits, of which f shall occur to the right of the decimal place; the values can be either represented exactly or interpreted to be in this range.
d) Char(N) means that the column shall be able to hold any string of characters of a fixed length of N.
e) Varchar(N) means that the column shall be able to hold any string of characters of a variable length with a maximum length of N. Columns defined as "varchar(N)" may optionally be implemented as "char(N)".
f) DATE means that the column shall be able to express any calendar day between January 1, 1900 and December 31, 2199.

The datatypes do not correspond to any specific SQL-standard datatype. The definitions are provided to highlight the properties that are required for a particular column.

2.2.2.3 Description

Description of each column as it relates to other columns within the dataset as well as the table.

2.2.3 Customer

Each row in this table represents a unique customer.

Column	Datatype	Description
c_customer_sk	INTEGER	Primary key for customer
c_customer_id	STRING	Customer ID
c_current_addr_sk	INTEGER	Foreign key to customer address
c_first_name	STRING	Customer first name
c_last_name	STRING	Customer last name
c_preferred_cust_flag	CHAR	
c_birth_day	INTEGER	Day of the month 1...31
c_birth_month	INTEGER	Month of year 1...12
c_birth_year	INTEGER	Year 1930...2002
c_birth_country	STRING	Customer birth country
c_login	STRING	Login name
c_email_address	STRING	Well-formed email address
c_cluster_id	INTEGER	Numeric label for UC01

Table 2-a Customer Table Column Definitions

2.2.4 Failure

Each row in this table represents the details of specific hardware failures within a datacenter

Column	Datatype	Description
Date	DATE	Date of the event (yyyy-MM-dd)
serial_number	STRING	Disk drive serial number
model	STRING	Disk drive model number
smart_5_raw	DOUBLE	Parameter value for smart_5
smart_10_raw	DOUBLE	Parameter value for smart_10
smart_184_raw	DOUBLE	Parameter value for smart_184
smart_187_raw	DOUBLE	Parameter value for smart_187
smart_188_raw	DOUBLE	Parameter value for smart_188
smart_197_raw	DOUBLE	Parameter value for smart_197
smart_198_raw	DOUBLE	Parameter value for smart_198

Table 2-b Failure Table Column Definitions

2.2.5 Financial Account

Each row in this table represents a financial account customer with respective transaction limits

Column	Datatype	Description
fa_customer_sk	INTEGER	Foreign key to customer

transaction limit	DOUBLE	Transaction limit set for the given customer

Table 2-c Financial Account Table Column Definitions

2.2.6 Financial Transactions

Each row in this table represents financial transaction details for a given time.

Column	Datatype	Description
amount	DOUBLE	Monetary amount that was sent
IBAN	STRING	Imitation of an IBAN
senderID	INTEGER	Foreign key to financial account
receiverID	INTEGER	Foreign key to financial account or random number
transactionID	INTEGER	Transaction ID
time	DATETIME	yyyy-MM-dd T hh:mm

Table 2-d Financial Transactions Table Column Definitions

2.2.7 Lineitem

Each row in this table represents a single Lineitem for a particular order

Column	Datatype	Description
li_order_id	INTEGER	Foreign key to order
li_product_id	INTEGER	Foreign key to product
quantity	INTEGER	Number of items
price	DOUBLE	Price paid per unit

Table 2-e Lineitem Table Column Definitions

2.2.8 Marketplace

Each row in this table represents the description of each marketplace.

Column	Datatype	Description
id	INTEGER	Record ID
description	STRING	Product description (may include branch and product name)

Table 2-f Marketplace Table Column Definitions

2.2.9 Order

Each row in this table represents a single Lineitem for details of an order

Column	Datatype	Description
o_order_id	INTEGER	Primary key
o_customer_sk	INTEGER	Foreign key to customer
weekday	STRING	Day of week (Monday, Tuesday, etc.)
date	DATE	yyyy-MM-dd
store	INTEGER	Foreign key to store_deptartment table

Table 2-g Order Table Column Definitions

2.2.10 Order_returns

Each row in this table represents a single line item for an item returned

Column	Datatype	Description
or_order_id	INTEGER	Foreign key to order
or_product_id	INTEGER	Foreign key to product
or_return_quantity	INTEGER	Number of units returned (at most the order quantity)

Table 2-h Order_returns Table Column Definitions

2.2.11 Product

Each row in this table represents a unique product id and its location

Column	Datatype	Description
p_product_id	INTEGER	Primary key
name	STRING	Arbitrary product name
department	STRING	Department name

Table 2-i Product Table Column Definitions

2.2.12 Product rating

Each row in this table represents a line item for the rating of a product

Column	Datatype	Description
userID	INTEGER	Foreign key to customer
productID	INTEGER	Foreign key to product

Table 2-j Product Table Column Definitions

2.2.13 Review

Each row in this table represents a single line item for product review.

Column	Datatype	Description
ID	INTEGER	Primary key
text	STRING	Review text

Table 2-k Review Table Column Definitions

2.2.14 store_department

Each row in this table represents a unique store and department

Column	Datatype	Description
store	INTEGER	
department	STRING	Name of the department
periods	INTEGER	Number of weeks to forecast

Table 2-l Store_department Table Column Definitions

2.2.15 Customer_images_meta

Each row in this table represents the identity metadata associated with each customer

Column	Datatype	Description
Identity_serving	STRING	Unique ID for identity that does not leak name
sample	INTEGER	Unique ID for image
Img_filename	STRING	Relative path to the image file

Table 2-m Customer_images_meta Table Column Definitions

2.2.16 Customer images

The customer images dataset consists of varied sized .png files that contains a fixed set of identities of each customer.

2.2.17 Audio conversations

The customer audio conversations dataset consists of varied sized .wav files that contains conversation information of customers.

3 DATA SCALING & POPULATION

This clause defines the **Test dataset** population and how it scales.

3.1 Scaling Model

3.1.1 The TPCx-AI benchmark defines a set of discrete scaling points called "scale factors" based on the approximate size of the raw data produced by **PDGF**. The actual byte count may vary depending on individual hardware and software configurations.

3.1.2 The set of scale factors defined for TPCx-AI is:

1GB, 3GB, 10GB, 30GB, 100GB, 300GB, 1000GB, 3000GB, 10000GB, 30000GB, 100000GB

Comment: The TPC recognizes that additional benchmark development work is necessary to allow TPCx-AI to scale beyond a certain limit of Scale factor size.

Scale Factor	SF
1GB	1
3GB	3
10GB	10
30GB	30
100GB	100
300GB	300
1000GB	1000
3000GB	3000
10000GB	10000
30000GB	30000
100000GB	100000

Table 3-a Scale Factor and SF

3.1.3 Test sponsors may choose any scale factor from the defined series. No other scale factor may be used for a TPCx-AI result.

3.1.4 Results at the different scale factors are not comparable, due to the substantially different computational challenges found at different data volumes.

3.1.5 The row size information provided is an estimate and may vary from one benchmark submission to another depending on the precise data base implementation that is selected. It is provided solely to assist benchmark sponsors in the sizing of benchmark configurations.

3.2 Data Population

The data generator used is based on an extension of the Parallel Data Generation Framework (PDGF). PDGF is a parallel data generator that can produce large amounts of data for an arbitrary schema. The existing PDGF can be used to generate the entire TPCx-AI **Test dataset**. PDGF handles the volume well since it can scale the size of the data based on a scale factor. It also runs efficiently for large scale factors since it runs in parallel and can leverage large systems dedicated for the benchmark.

	1GB	3GB	10GB	30GB	100GB	300GB	1000GB	3000GB	10000GB
Number of Images	100	300	1,000	3,000	10,000	30,000	100,000	300,000	1,000,000
Number of Converstaions	100	300	1,000	3,000	10,000	30,000	100,000	300,000	1,000,000

Table 3-b Table of number of images and audio conversations

	Sample Row Count								
	1GB	3GB	10GB	30GB	100GB	300GB	1000GB	3000GB	10000GB
# Customer	70,711	145,773	357,071	823,104	2,037,768	4,482,884	9,994,749	19,640,519	39,239,776
# weeks	88	121	161	199	240	276	312	342	374
# Product	707	1,458	3,571	8,231	20,378	44,829	99,947	196,405	392,398
# Order	3,676,955	8,819,314	28,744,249	81,898,836	244,532,206	618,104,523	1,559,062,223	3,363,320,197	7,332,885,174
# LineItem	23,026,666	55,212,270	179,968,455	512,867,316	1,531,137,997	4,017,679,269	10,133,904,446	21,861,581,278	47,663,753,632
# Order Returns	1,331,621	3,200,152	10,430,034	29,736,421	88,789,335	200,883,963	506,695,222	1,093,079,064	23,838,187,682
# Financial Accounts	7,071	14,577	35,707	82,310	203,777	448,288	999,475	1,964,052	3,923,978
# Financial Transactions	7,353,840	17,638,170	57,488,270	163,796,900	489,064,800	1,236,209,006	3,118,124,445	6,726,640,393	14,665,770,348
# Samples	71	146	357	823	2,038	4,483	9,995	19,641	39,240
# Disks	707	1,458	3,571	8,231	20,378	44,829	99,947	196,405	392,398
# Failures	49,490	211,265	1,274,490	6,774,113	41,507,949	200,962,500	998,950,000	3,857,500,000	15,397,600,000
# Marketplace	70,711	145,773	357,071	823,103	2,037,768	4,482,884	9,994,749	19,640,519	39,239,776
# Identities	7	14	36	82	204	448	999	1,964	3,924
# Product Ratings	120,695	317,665	847,514	2,040,255	5,470,732	10,446,516	23,986,941	48,288,782	98,835,001
# Reviews	134,350	306,117	821,261	1,975,440	5,298,202	12,552,064	28,984,775	60,885,612	125,567,296

Table 3-c Table of Rows

4 USE CASE SUMMARY

4.1 Use case 1 - Customer Segmentation

The Customer segmentation (UC1) use case is designed to emulate the data science pipeline to find clusters of customers based on aggregate features where the customers are grouped based on their spending behavior. It involves creating subgroups of customers based on similar traits. The input in this use case consists of order and return transaction data from a retail business. The use case uses Tables Customer, Order, Lineitem and Order_returns

K-means clustering algorithm is used to derive the optimum number of clusters and understand the underlying customer segments based on the data provided. Clustering is an unsupervised machine learning technique, where there are no defined dependent and independent variables, i.e. the training samples are unlabeled. The patterns in the data is used to identify and group similar observations.

The output comprises of 4 distinct cluster centers.

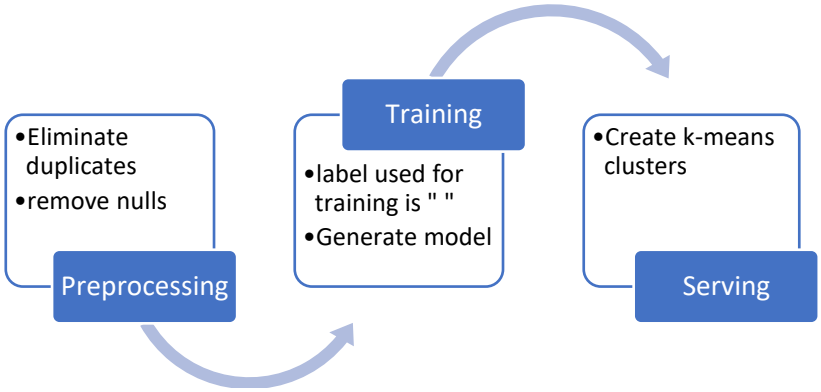


Figure 4-a Use case 1 *logical* data science pipeline

4.2 Use case 2 – Customer Conversation Transcription

The customer conversation transcription (UC2) use case is designed to emulate translating customer audio conversations to text. Usually and if not explicitly forbidden by the customer, customer calls are recorded. To index and classify the conversations it is necessary to transcribe them. Human transcription is costly; therefore an automatic system can be used to generate the transcriptions. This use case represents an automatic transcription of given customer calls. The input is a set of audio recordings of customer calls. Data acquisition consists of loading conversation data, transformation of data involves resampling audio to 16kHz, calculating Mel Frequency Cepstral Coefficients (MFCC) from raw audio, transforming labels to encoded sequences and then finally training a deep neural net model on the MFCC features.

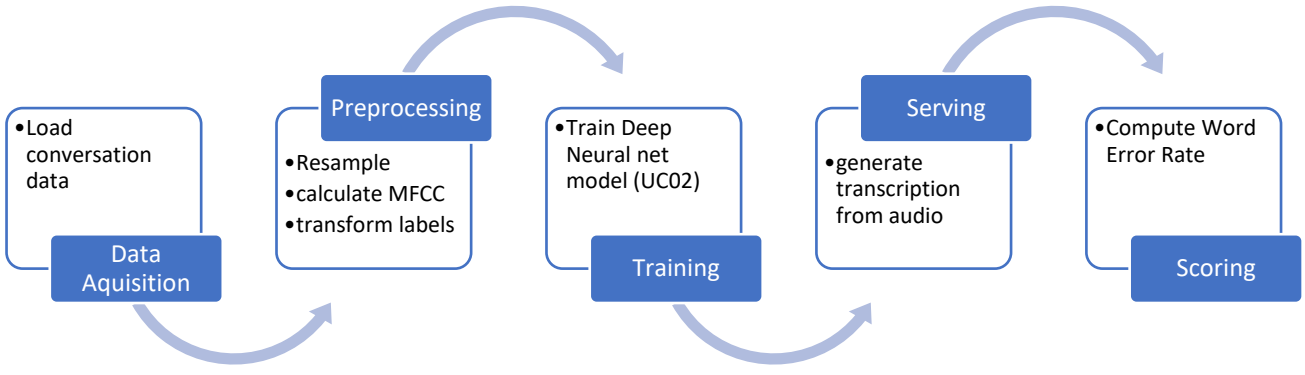


Figure 4-b Use case 2 logical data pipeline

4.3 Use Case 3 – Sales Forecasting

The sales forecasting (UC3) use case is designed to emulate weekly sales forecasting for up to 1 year for each department and each store given a limited history of sales data. The input tables are orders, product, Lineitem and store_department. Also included in the dataset are markdown events that might affect the sales of some departments. Each store of a retail chain with multiple stores has limited history of sales data. The retail chain has multiple stores, and each store has multiple departments, such as jewelry, toys, office supplies, kitchen accessories,

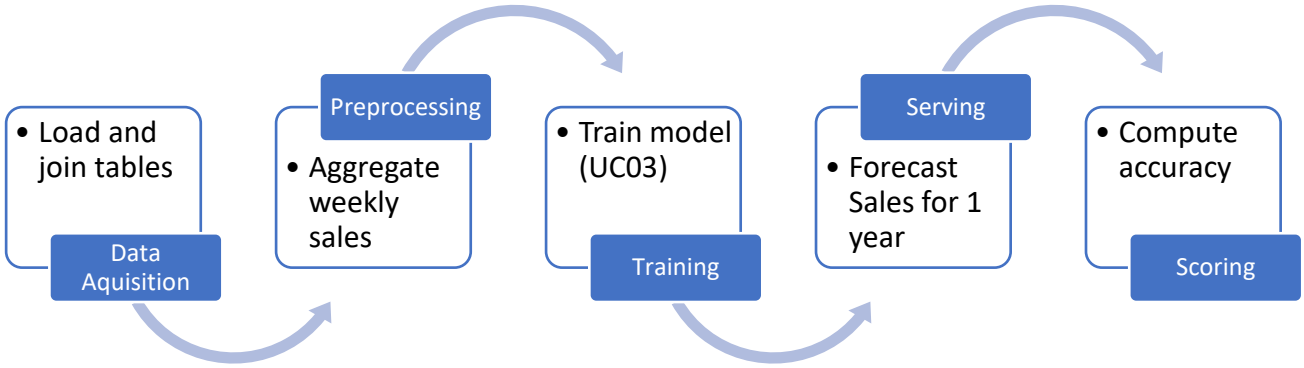


Figure 4-c Use case 3 data science pipeline

among others. The use case uses the tables Order, Lineitem, Product and Store_department

4.4 Use Case 4 – Spam Detection

The Spam Detection (UC4) use case is designed to emulate detection of spam content given a set of input comments, reviews, or descriptions in the context of a retail setting. has long been a common production use case. The data science pipeline tries to identify reviews that are spam. A Naïve Bayes model is trained and used to generate an accurate set of predictions. Naive Bayes methods are a set of supervised learning algorithms based on applying Bayes' theorem. In the "naive" assumption of conditional independence between every pair of features given the value of the class variable. The output is an array of predictions with reviews that are marked either as spam or truthful. Use case 6 uses the Reviews table.

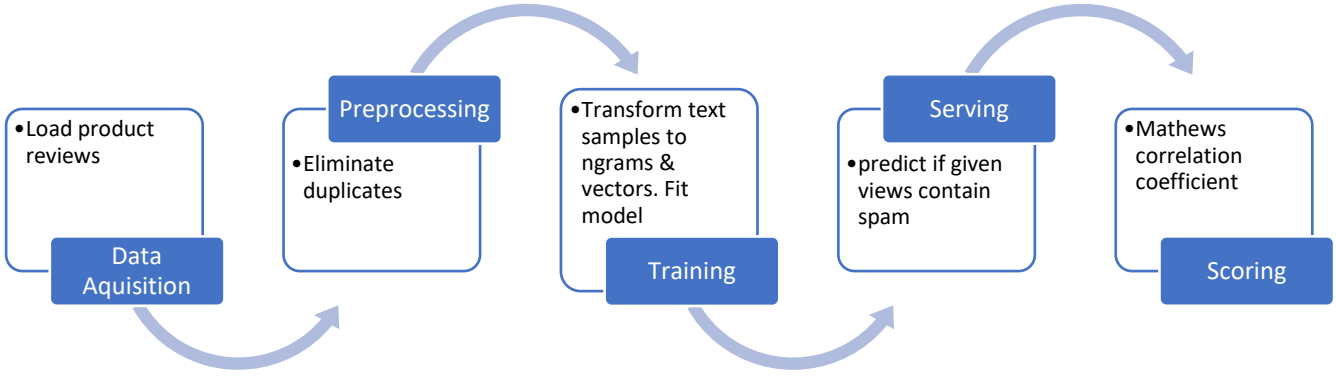


Figure 4-d Use Case 4 Data Science Pipeline

4.5 Use case 5 – Price Prediction

The price prediction (UC5) use case emulates the data science pipeline to suggest or predict a price of a retail item based on its brand, product name, and description on an online marketplace. All the input features in the dataset are user generated. The product description does not show any obvious structure and describes the product in detail. Use case 5 uses the Marketplace table as input.

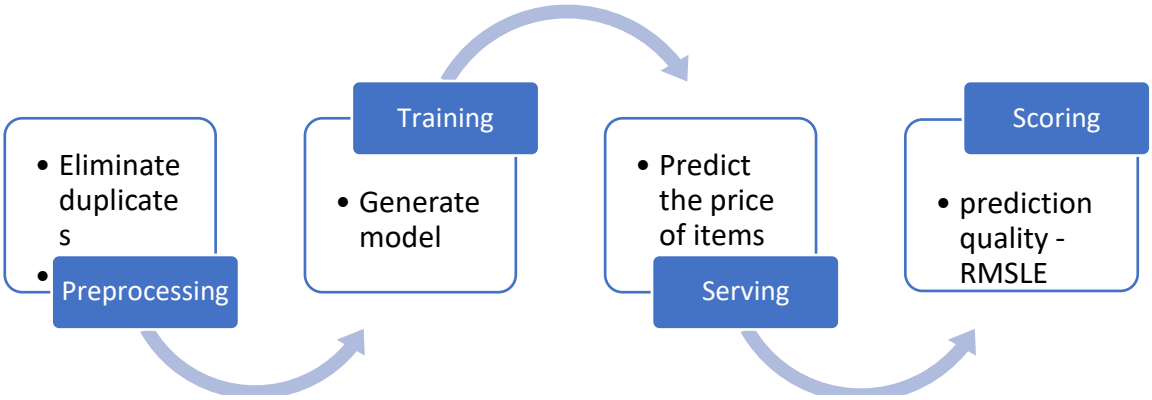


Figure 4-e Use case 5 Data Science Pipeline

4.6 Use Case 6 – Hardware failure

The hardware failure (UC6) use case is designed to emulate the data science pipeline that predicts and warns of impeding failures of hardware in advance. This is based on data and logs collected for every drive. The input is logs of hardware events. Data is pre-aggregated per day and the log records contain events, component IDs, date and failures. The model will predict if component failure is imminent for each set of, previously unseen, log entries. Use case 4 uses the Failures Table.

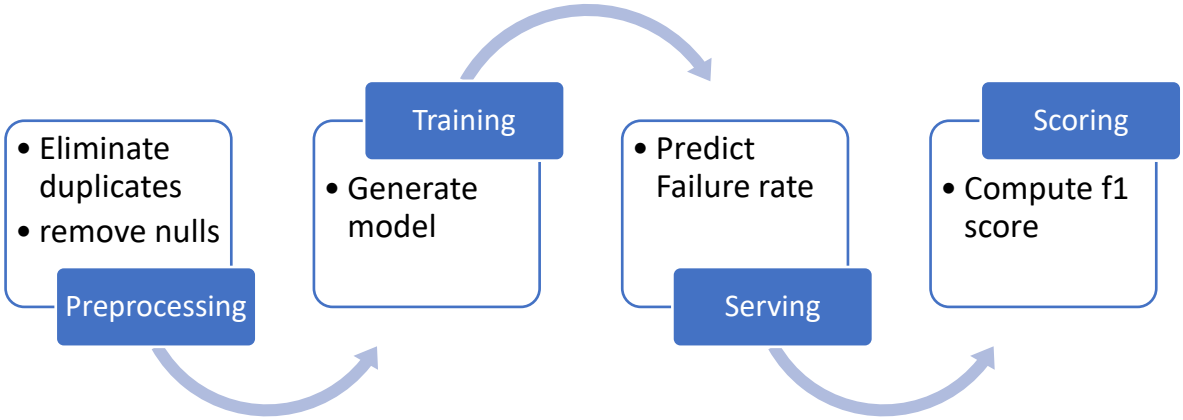


Figure 4-f Use case 6 Data Science Pipeline

4.7 Use case 7 – Product Rating

One of the biggest challenges faced by the retail industry is generating accurate marketing based on available history of information. In other words, how to accurately inform the customer of products that are available. The product rating (UC7) is designed to emulate a data science pipeline for a recommender system that one would typically find in an online retail environment. The starting point is the shopping history of the customers and more important their rating of historically purchased products. This rating history will be used to train a recommender system that, upon being trained, is able to recommend products that the customer might also be interested in. Use case 7 uses the Product_Rating table,

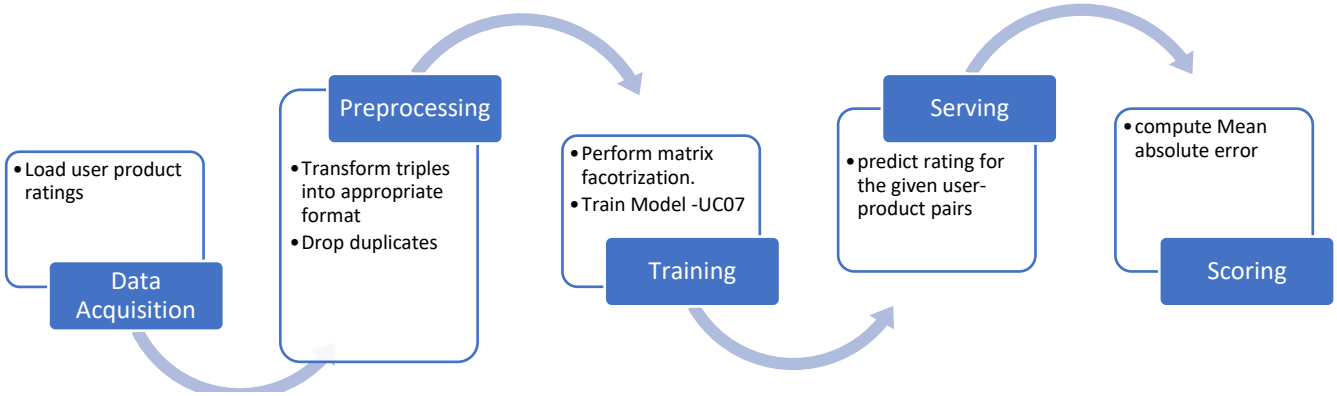


Figure 4-g Use case 7 Data Science Pipeline

4.8 Use case 8 – Classification of Trips

Customer shopping trips most often can be classified into different types of trips by the retail stores. The underlying assumption is that different trip types have different characteristics and patterns that make them unique. For instance, a weekly trip for grocery shopping would have different data associated with a holiday compared to a Christmas shopping trip. Use Case 8 (Classification of trips) is designed to emulate training a classifier using a pre-labeled data set, i.e. a data set already containing said trip types, of shopping transactions and then predict the future trip. This prediction could be used in many ways, from tailoring promotional material to customers, marketing activities, placing products in key departments based on the trip type and historical transactions.

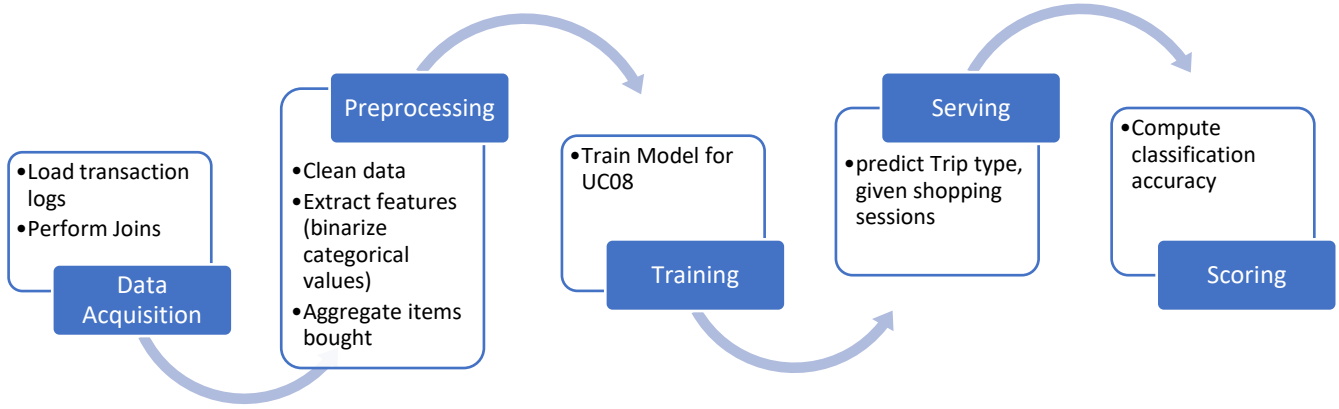


Figure 4-h Use case 8 Data Science Pipeline

4.9 Use case 9 – Facial recognition

The Facial recognition use case (UC9) is designed to emulate an end to end facial recognition data science pipeline. This use case identifies frequent customers or other persons of interest by training a classifier that recognizes previously seen faces. The data acquisition includes loading metadata as well as importing images using paths from the metadata. The images contain customer faces. The name of the customers is encoded and images aligned based on facial features. The image resolution is changed to fine-tune a pre-trained embedding. A logistic regression model is

trained on the embedding and the customer name, an embedding is created for aligned faces. Finally, the logistic regression model is used to recognize the face and identify the name for the image.

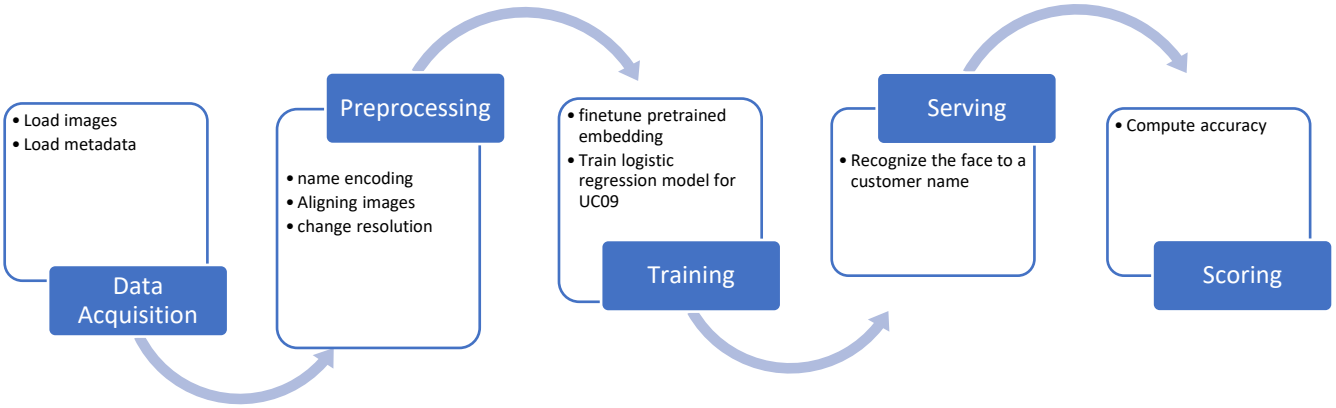


Figure 4-i Use case 9 Data Science Pipeline

4.10 Use case 10 – Fraud Detection

The Fraud detection use case (UC1) is designed to emulate whether a financial transaction is a normal transaction or a fraudulent transaction. The input in this use case consists of financial transactions in a retail business. The problem to be solved is to identify transactions that are fraudulent. Logistic regression is a linear model for classification that is trained for this use case. The output is an array of predictions (predicted class label per sample).

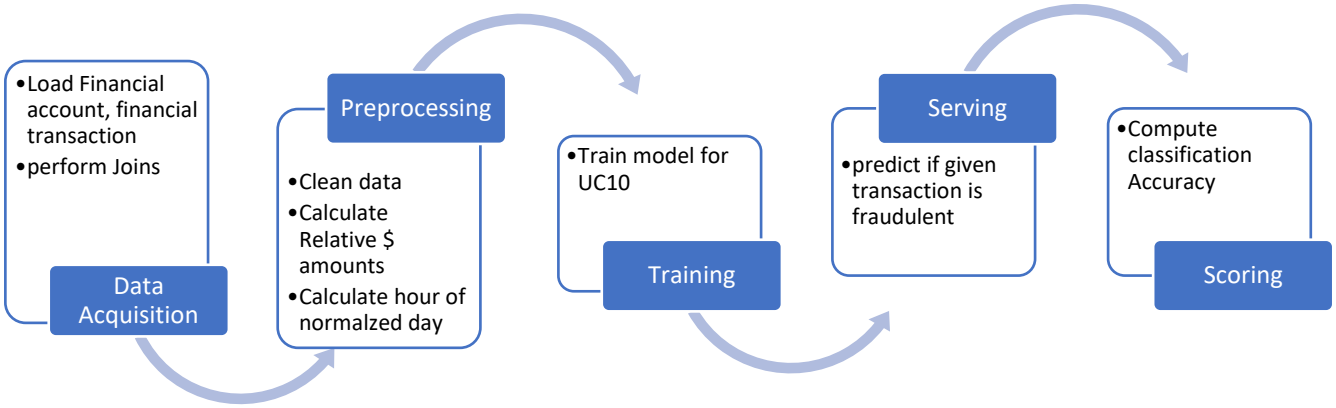


Figure 4-j Use case 10 Data Science Pipeline

5 BENCHMARK KIT CONTENT AND DEVELOPMENT

5.1 Benchmark Kit

This clause defines TPCx-AI Kit contents, its workload execution process, allowed modification by the test sponsor, and contents of the run report.

5.1.1 Kit Contents

The TPCx-AI kit contains the following:

a) TPCx-AI Specification document.
b) TPCx-AI Users Guide (README.md) documentation.
c) Configuration files to adapt important parameters to the SUT.
d) Scripts which control the benchmarking execution.
e) A driver that implements the high-level run logic, time measurement and result computation
f) A set of scripts which are called by the driver to perform benchmark and Use Case operations.
g) A set of scripts to automate result verification, checks on result cardinality and report generation.
h) A set of scripts to build parts of the kit that are not prebuilt for the user.
i) A set of scripts and notes to provide guidance and reference for the setup of virtual execution environments.

5.2 Kit Usage for a compliant result

5.2.1 To submit a compliant TPCx-AI **Result**, the **Test Sponsor** is required to use the TPCx-AI kit as outlined in the TPCx-AI Users Guide (README.md) with the following two exceptions:

- The setting of Kit Parameter files specified in Clause 5.4.1.
- Test Sponsor Kit Modifications explicitly allowed by Clause 5.6.

5.2.2 If there is a conflict between the TPCx-AI Specification and the TPCx-AI kit, the TPCx-AI kit implementation prevails.

5.3 Kit Run report

The output of the TPCx-AI kit is called the run report which includes the following:

- Version number of the TPCx-AI kit.
- The input parameters used to run the benchmark.
- The start, end and total elapsed times of the 6 tests (Clause 7.3) of each Benchmark Run (i.e. Loading, Power training, Power serving I, Power serving II, Throughput Serving, and Scoring).
- The computed TPCx-AI Secondary Metrics (Clause 7.5) for the Benchmark Run.
- The final performance metric for the benchmark run.
- Whether the run is considered valid (i.e. all phases were executed successfully, and all quality metric threshold conditions, as specified in clause 7.3.10.3, were met)

5.4 Kit Parameter settings

5.4.1 The following files and control the kit parameters that may be set by the **Test Sponsor**.

a) Generic Benchmark parameters defined in Appendix B
b) **Only the use case specific** parameters defined in Appendix B can be changed. All other parameters cannot be changed.

Comment: Global parameters are engine specific. The **Test Sponsor** can set their own parameters and must disclose as part of **FDR**. For example, when using the Apache Spark execution engine, the Test Sponsor can configure generic parameters, such as the number of tasks used to run one or more use cases, the distribution of those tasks across the nodes of the cluster, the amount of compute resources allocated for each task, compression algorithms, and serializer interfaces, among others.

5.5 Benchmark Kit

5.5.1 Prior to donating a TPCx-AI draft kit or making code modifications to an existing TPCx-AI release kit, a CLA must be in place and provided by the sponsor.

5.5.2 The draft kit and compute software must be reviewed by the TPCx-AI subcommittee during the review process where the subcommittee decides the validity of the software stack that can be used for publications.

5.5.3 **Comment:** If the draft kit contains commercially available libraries that perform the same function as in clause 6.3.1, the libraries will need to meet the pricing criteria as listed in the pricing specification. The test sponsor will be responsible for support or can hand over support to a third party.

5.6 Benchmark Kit Modifications

5.6.1 For kit changes or modifications other than those allowed by Clause 5.4 any TPC Member, company or individual may bring forward proposed kit changes to the TPCx-AI Benchmark Subcommittee. There are two methods of bringing forward these proposed kit changes:

a) Direct Method – A TPC Member, company, or individual may propose kit changes directly to the TPCx-AI Subcommittee.

b) Indirect Method – If the TPC Member, company, or individual wishes to remain anonymous then a **TPC Certified Auditor** can be used as an intermediary to interact with the TPCx-AI Subcommittee.

5.6.2 Regardless of which method is used, the individual that will be interacting with the TPCx-AI Subcommittee becomes the Change Sponsor.

5.6.3 Test Sponsor modifications to the provided scripts and configuration files in the TPCx-AI kit to facilitate system, platform and **Framework** differences are allowed without TPC approval. The allowed Test Sponsor Modifications are as follows:

a) Script changes necessary for the kit scripts to execute on a particular Operating System, container platforms or compute service as long as the changes do not alter the execution logic of the script.

b) **Use case** specific optimization **Framework** parameters can be set for the configuration parameters Appendix B only

5.6.4 General guidelines used by the subcommittee to review Kit changes include but are not limited to:

a) perform the same function as in clause 6.3

b) The same input data must be used

c) To the extent that the new framework accepts similar parameters to existing frameworks (number of iterations, number of clusters, regularization parameters), the values for these parameters should be similar to those used for existing frameworks. If there is a need for the parameters to be different there must be sufficient technical justification provided.

d) The new **compute software** should be initialized using techniques that are comparable to the existing **compute software** (e.g. for clustering, a new **Framework** should use the same random initialization approach).

5.6.5 General guidelines used by the subcommittee to review & incorporate new or different versions of the **TPCx-AI list of Compute software** that are unsupported and not validated or approved by the TPC:

5.6.5.1 New or Different versions of the **compute software** libraries and frameworks are intended to be used as part of the Kit (driver) can be used as long as it does not improve the performance of the SUT.

5.6.5.2 The new or different version should provide the same functionality as the previous approved **compute software** versions that are part of the existing benchmark kit

Comment: An example could be a schedular that works with python to distribute execution of the data pipeline on multiple nodes.

5.6.6 The new **compute software** should be capable of reporting the same accuracy/evaluation metrics (f1 score, precision, etc.) as existing ML **Frameworks** and these metrics must demonstrate a level of accuracy for the new framework that is at least as good (within margin of error) as the accuracy of the earlier **compute software** used in the comparison.

5.6.7 No modifications are allowed to the Bash scripts, Java code or the Python code provided in the TPCx-AI kit.

5.6.8 No JAR file optimizers are allowed to be used.

5.6.9 Any kit modifications not specified in Clause 5.4 must be brought forward to the Subcommittee as specified in Clause 5.6

5.6.10 Classification of Major, Minor and Third Tier Kit Modifications

It is necessary to ensure that the kit remains in sync with fast changing industry and technology landscape. The guidelines below illustrate the current structure of the Kit and help the Subcommittee to decide in a timely manner when evaluating a change proposal. These guidelines will help the Subcommittee do its due diligence and use its discretion to classify and process the change proposals. Modifications to the kit are divided into three types that follow the Revision classifications defined in the TPC Policies. They are:

a) Major Kit Modifications
b) Minor Kit Modifications
c) Third Tier Kit Modifications

5.6.11 Major Kit Modifications:

Major Kit Modifications result in a significant change to the **Use Cases** or intent of the TPCx-AI Benchmark as to make **Results** from the new version non-comparable with the **Results** of the current TPCx-AI version.

These are a few examples of Major Kit Modifications:

a) Additions, deletions, and modifications to a **Use Case**
b) changes to the Primary **Benchmark Metric**
c) changes which may alter the reference result set
d) changes made to run rules and Benchmark execution process
e) Maximum number of iterations
f) Learning rate

5.6.12 Minor Kit Modifications:

Minor Kit Modifications do not significantly alter the reference result set, the primary benchmark metrics, or the **Use Case**. Results are still comparable to the prior version. A few examples of Minor Kit changes:

a) Addition of a new **Framework** support (e.g. Pytorch)
b) bug fixes throughout the entire kit
c) optimizations to the Framework specific code
d) feature additions to Benchmark Driver
e) modifications to configuration parameter files
f) reference result set changes due to bug fixes
g) Framework feature support (or enhancements)
h) updates to independent library files
i) changes to the Data generator to support features and bugfixes that don't significantly change the characteristics of the datasets that are generated

5.6.13 Adding support for a new Framework or data science pipeline solution

The use case implementations included in TPCx-AI use machine learning software to solve problems. Machine learning software applications and techniques often involve random number generation that might result in slight variations in their final answer. Therefore, doing an exact comparison with a known fixed answer set is not practical and some other criteria must be applied to determine whether modifications are yielding **Results** that should be considered comparable. There are two general categories of changes that could impact the machine learning pipeline in the **TPCx-AI** kit:

a) Changes to the version/implementation of the SUT's machine learning library (for example a new version of the Spark MLLib library) without any changes to kit itself. The concern in this case is that a new version of the machine learning library could make a different tradeoff in accuracy vs performance compared to earlier

versions. The following criteria will be applied to evaluate whether results using a new library version should be comparable to previous **Results**:

- Results using the new library version must be generated without any changes to code or parameters in the kit (in particular there can be no changes to the input data, the parameters to the algorithm (e.g. number of clusters for **k-means**, algorithm initialization parameters including seeds for any random initialization, regularization parameters for classification algorithms, etc.).

b) Introduction of new machine learning **Frameworks** (not just new versions of the previously supported framework) may require actual changes in the kit code or parameters. This case is more subjective, but the general guidelines for considering results from a new ML **Framework** to be comparable are:

- The same input data must be used
- To the extent that the new framework accepts similar parameters to existing frameworks (number of iterations, number of clusters, regularization parameters), the values for these parameters should be similar to those used for existing frameworks. If there is a need for the parameters to be different there must be sufficient technical justification provided.
- The new **Framework** should be initialized using techniques that are comparable to the existing **Framework** (e.g. for clustering the new **Framework** should use the same random initialization approach).
- The new **Framework** should be capable of reporting the same accuracy/evaluation metrics (sum of squared distance, precision, AUC, etc.) as existing ML **Frameworks** and these metrics must demonstrate a level of accuracy for the new framework that is at least as good (within margin of error) as the accuracy of the earlier **Framework** used in the comparison.

5.6.14 Third Tier Kit Modifications:

Third Tier Kit Modifications are those changes that clarify some confusing or ambiguous area of the kit, but do not alter the benchmark kit code or the **Use Cases**. Results are still comparable to the prior version. These are a few examples of Third Tier changes:

a) changes in documentation

5.6.15 Simple Review of Kit Modifications

5.6.15.1 For Third Tier (Clause 5.5.9) or Minor kit (Clause 5.5.8) modifications, the Change Sponsor shall present the proposed changes to the Subcommittee. The Subcommittee through its normal course of business will review the proposed changes, make the appropriate kit changes and bring forward the changes to the Council as a new revision of the TPCx-AI Benchmark.

5.6.15.2 If the proposed changes are significant, the Subcommittee may require that the Change Sponsor follow the Formal Review Process defined in Clause 5.6.18 and Clause 5.6.19.

5.6.16 Formal Review of Kit Modifications

For Major (Clause 5.6.11) kit Modifications, at the request to the Subcommittee or if the Change Sponsor so desires, the Change Sponsor shall adhere to the following Formal Review Process.

5.6.17 Formal Proposal of Kit Modifications

5.6.17.1 Step 1: The Change Sponsor must submit to the chair of the TPCx-AI Subcommittee the following information:

a) The proposed code changes or new **Framework** code or model
b) The reason for proposing the changes
c) Result set from the proposed changes
d) Complete source code access if the proposed change prototype is available

Comment: To facilitate decision making process change sponsor may provide hardware and software required to validate and review the proposed changes.

5.6.17.2 Step 2: The chair of the TPCx-AI Subcommittee will add a discussion of the proposed changes to the agenda of the next Subcommittee meeting that can be attended by the Change Sponsor.

5.6.17.3 Step 3: The Change Sponsor will present the proposed changes to the TPCx-AI Subcommittee.

5.6.17.4 Step 4: The TPCx-AI Subcommittee will vote on one of three courses of action for the proposed changes.

a) Reject the proposed changes.

b) Review the proposed changes as a Minor Kit Modification.
c) Review the proposed changes as a Major Kit Modification.

5.6.17.5 If the proposed changes are rejected, no further action is necessary. Otherwise, the proposed changes immediately enter a Proposed Change Review period.

5.6.18 Formal Review of Proposed Major Kit Modifications – Approximately six to twelve Week review period.

If the proposed changes were voted to be a Major Kit Modification, then the Subcommittee chair will select at least three members of the Subcommittee to act as primary reviewers of the proposed changes. The Subcommittee chair will also determine the length of the review period and communicate the due date to the primary reviewers and to the Subcommittee. The primary reviewers' job is to examine and test the proposed changes. The primary reviewers are to give their recommendation to the Subcommittee no later than the due date set by the Subcommittee chair which is approximately six to twelve weeks.

5.6.19 Formal Review of Proposed Minor Kit Modification – Six-week review period

If the proposed changes were voted to be a Minor Kit Modification, then the Subcommittee chair will select at least two members of the committee to act as primary reviewers of the proposed changes. The primary reviewers' job is to examine and test the proposed changes. The primary reviewers are to give their recommendation to the committee no more than six weeks later.

5.6.20 Formal Review by Subcommittee

Once the review period ends and the primary reviewers have given their recommendations, the subcommittee will vote on whether to accept the proposed changes into the TPCx-AI benchmark kit.

If the changes are accepted, then the changes will be added to the kit.

5.7 Kit Validation

Before any kit can be submitted for approval as a new revision of the TPCx-AI Benchmark Standard, all changes must pass the self-validation tests in the kit.

6 SYSTEM UNDER TEST (SUT)

6.1 Logical Breakdown of System Under Test

6.1.1 The tested and **reported** configuration is composed of the hardware and software components that are employed in the TPCx-AI benchmark test whose cost and performance are described by the benchmark metrics.

6.1.2 System Under Test consists of:

6.1.2.1 Hardware components that can be bare-metal, virtual machines or virtual instances.

6.1.2.2 **Compute Software**: Compute software runs on Compute Hardware providing required software capabilities to successfully execute the benchmark.

6.1.2.3 Data Storage Software: Data Storage software runs on Data Storage hardware providing required software to create, store, and access input, output, intermediate, and temp data during the benchmark execution.

6.1.2.4 Compute Hardware: Compute hardware provides multi-device compute capable hardware to execute the benchmark.

6.1.2.5 Data Storage Hardware: Data Storage hardware provides data storage capability using various kinds of persistent storage medium.

6.1.2.6 Network Hardware and Software: Network Hardware and software provides capability to connect hardware and software in the **SUT** to communicate and perform data transfer over the network.

6.1.2.7 Devices in addition to listed above used in the **SUT**, for example compute devices and/or data storage devices, for e.g. FPGA, Accelerator appliance, Accelerator cards, compression add-on cards, encryption add-on cards etc. and their supporting software stack, device driver software, plug-in software.

6.1.2.8 Any hardware and software devices of all networks required to connect and support the **SUT** systems

6.1.2.9 Device running benchmark driver hardware and software resides on a separate system facilitating the execution of the benchmark, without interfering and influencing the **SUT**. This device is not part of the **SUT** and contains necessary SW and configuration to interact with the **SUT** and can be in form of Desktop, Notebook, or a Server.

6.1.2.10 Possibly Licensed Compute Services**.**

6.1.2.11 TPCx-AI kit provides fully integrated benchmark and driver software to run on **SUT**.

6.1.2.12 The figures below show example models of the target SUT.

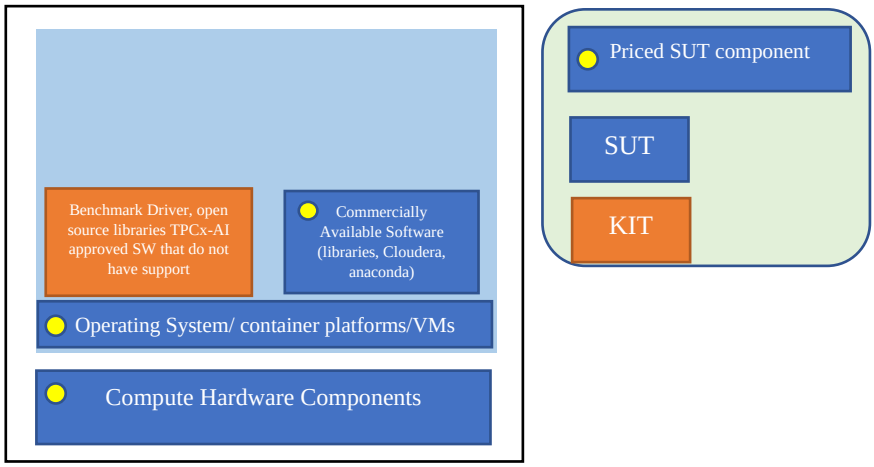


Figure 6-a Single SUT Configuration

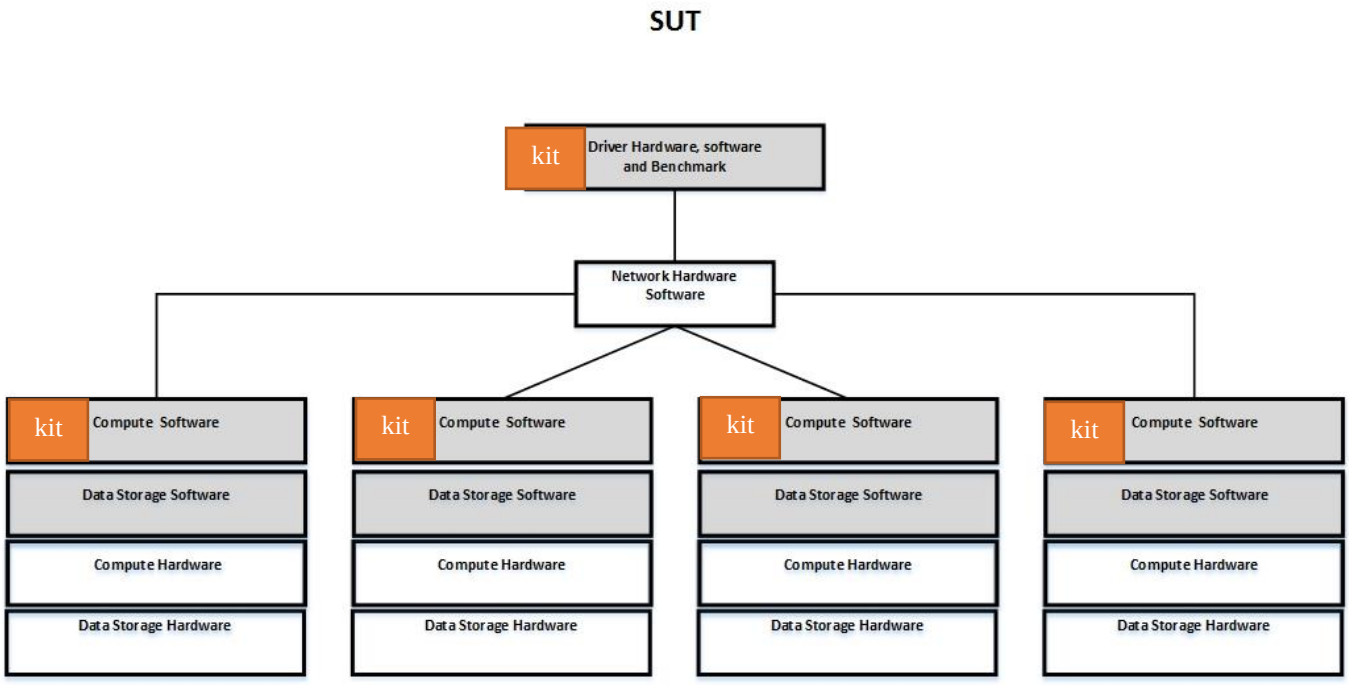


Figure 6-b Clustered SUT Configuration

6.2 Commercially Available Products

Except for the TPCx-AI benchmark driver software and **TPCx-AI approved list of compute libraries** needed to run the benchmark, all **SUT** components must be **Commercially Available Products**. The source code of any non-commercially available products used to implement the **SUT** (including but not limited to scripts used to install, configure and tune the **SUT**) must be disclosed.

6.3 Benchmark driver & compute software and libraries

The following constitute the software that is part of the Benchmark Kit & **compute software that is part of the SUT**.

6.3.1 Benchmark kit or driver software needed to run the benchmark as shown in the figures above will be not be priced.

6.3.2 All software and libraries and frameworks used as part of the **compute software** (clause 6.1.2.2) are part of the SUT.

6.3.3 The TPCx-AI subcommittee will maintain a list of **TPCx-AI approved list of compute software (including version)** as allowable software to be used for the purposes of benchmark publications where the pricing specification may not apply. They may be considered as part of the TPCx-AI benchmark driver software and not the SUT.

6.3.4 The **TPCx-AI approved list of compute software** libraries listed by the TPCx-AI subcommittee will have been validated for a specific version of the release kit and the subcommittee shall vote in new items on that list (clause 5.6.5)

6.4 Data Redundancy Requirement

The following clauses describe required **Data Redundancy** characteristics of the **SUT**. The failures described are not induced during the benchmark Execution.

6.4.1 Durable Medium: A durable medium that is either:

a) An inherently non-volatile medium (e.g., magnetic disk, magnetic tape, optical disk, solid state disk, Phase Change Memory, or technology, similar to Phase Change Memory. etc.)or

b) A volatile medium with its own self-contained power supply that will retain and permit the transfer of data, before any data is lost, to an inherently non-volatile medium after the failure of external power.

6.4.2 The System Under Test must be configured to satisfy the requirements for **Data Redundancy** described in this clause. **Data Redundancy** is demonstrated by the **SUT** being able to maintain operations with full data access during and after the permanent irrecoverable failure of any single storage Medium containing tables, input, output, or **Metadata** (including **Master Node/Name Node Metadata** [where present]).

Comment: A configured and priced Uninterruptible Power Supply (UPS) is not considered external power.

Comment: DRAM can be considered a durable storage medium if it can preserve data long enough to satisfy the requirement (b) above. For example, if memory is accompanied by an Uninterruptible Power Supply, and the contents of memory can be transferred to an inherently non-volatile medium during the failure, then the memory is considered durable. Note that no distinction is made between main memory and memory performing similar permanent or temporary data storage in other parts of the system (e.g., disk controller caches).

6.4.3 Data Redundancy Reporting Requirements

6.4.3.1 The test sponsor must guarantee that the test system will not lose data due to a permanent irrecoverable failure of any single durable medium. Use case execution is not permitted to fail and no data can be lost in the presence of a single durable medium failure. This requirement applies to all Durable Medium containing TPCx-AI data (e.g., **Test dataset** input data, serving and scoring datasets including table data, **Metadata** (includes **Master Node/Name Node Metadata** [where present], **Undo/Redo Log** data [where present], and "temporary data" [where present]). This requirement also applies to any benchmark use case results(output data) stored on the SUT.

6.4.3.2 For **HDFS** file systems providing redundancy via 3-way replication, erasure coding, etc.:

Provide a report showing data resiliency. For example, with **HDFS** this can be done by running "hdfs fsck –blocks". "hdfs dfs -du -s –h" and "hdfs ec -getPolicy -path /". When TPCxAI_Benchmarkrun.sh is run, this report will automatically be generated at the end of each benchmark run.

- For 3-way replication, the "default replication factor" should be at least 3 and "under-replicated blocks" should be zero.
- For erasure coding, the auditor must verify that the codec, node_count, and parity settings results in redundancy at least equivalent to 3-way replication.

Comment: Typically this will be accomplished by verifying that "under-erasure-coded block groups"=0, num_parity blocks ≥3 and node_count ≥ (num_data_blocks + num_parity_blocks) but the auditor may need to perform an additional, implementation specific review.

6.4.3.3 For distributed file systems other than **HDFS,** like traditional JBOD, redundancy must be proved by the **Test sponsor**. They must provide a description of the data redundancy approach describing both hardware and software used to achieve the data redundancy and explain why it is at least equivalent to the data redundancy provided by traditional local-**JBOD** storage and **HDFS** replication factor of 3.

Comment: If stored in a distributed filesystem, **Test Dataset** Input Data, Metastore Data and Output Data must be set to at least an equivalent of replication factor three for HDFS on JBOD. Non HDFS distributed file systems must demonstrate data redundancy equivalent to using replication factor three in HDFS .

6.4.3.4 For SUT components NOT using a distributed file system such as **HDFS,** but running on a single node SUT that provide redundancy via a **High Availability System:**

a) The test sponsor needs to provide a report that explains the configuration in sufficient detail to satisfy the auditor/PPB that outlines the use of distinct durable mediums for the individual service instances in the **HAS**.

b) While encouraged, there is no requirement for triple redundancy for this class of data.

Comment: A single durable medium failure could take down a service instance in the **HAS** but continued execution would be guaranteed by the existence of a secondary service instance using a distinct durable medium.

Comment: For consistency with the distributed file system model, no explicit test is necessarily required.

c) The solution must guarantee uninterrupted access to the data on durable medium when a single Durable Media failure occurs.

d) The test sponsor must provide a report from a system tool detailing the media redundancy hardware/software configuration to the satisfaction of the auditor/PPB (e.g., a report showing that RAID-5 or RAID-10 is used).

Comment: Roll-forward recovery from an archive dataset copy (e.g., a copy taken prior to the run) using **Undo/Redo Log** data is not acceptable as the recovery mechanism in the case of durable medium failure. Note that "checkpoints", "control points", "consistency points", etc. of the dataset taken during a run are not considered to be archives.

Comment: For consistency with the distributed file system model, no explicit test is necessarily required.

Comment: Use Case execution may not fail due to a permanent irrecoverable failure of any single durable medium containing TPCx-AI data. However, medium failures are not allowed during benchmark runs to be considered valid (e.g., to avoid the possibility of higher performance when 3-way replication degrades into 2-way replication on medum failure).

Comment: At the heart of this requirement is handling the failure of any single durable media for consistency with other TPC benchmarks. For distributed file systems, while **HDFS** 2-way replication would have satisfied the spirit of this requirement, **HDFS** deployments using 3-way replication were the norm (both for redundancy and for performance). Consequently, while the requirement mandates handling only a single point of failure, solutions using distributed file systems additionally require equivalence to 3-way replication.

7 EXECUTION RULES AND PERFORMANCE METRICS

7.1 Benchmark Execution

7.1.1 A Benchmark Execution is defined as a Validation test (Clause 7.2) followed by the **Benchmark Run** that consists of 6 tests.

7.1.2 The Test sponsor runs the following scripts, in the order as they are presented:

a) TPCxAI_Validation.sh

b) TPCxAI_Benchmarkrun.sh

7.1.3 No part of the **SUT** may be restarted during the Benchmark Execution. If there is a non-recoverable error reported by any of the applications, operating system, or hardware in any of the tests (Clause 7.1.1) or between validation and benchmark runs, the run is considered invalid. If a recoverable error is detected in any of the tests and is automatically dealt with or corrected by the applications, operating system, or hardware, then the run is considered valid provided the run meets all other requirements. However, manual intervention by the **Test Sponsor** is not allowed. If a recoverable error requires manual intervention to deal with or correct, then the run is considered invalid.

7.2 Validation Test

7.2.1 The Validation test performs data generation, data load, power training test, power serving test, and scoring test with scale factor 1 to perform a result validation against the reference result set in the kit . Validation test ensures that the setup used by the Test Sponsor to produce the publication are comparable to the reference result set generated.

7.2.2 The validation result set for SF1 is the reference result used to validate the **SUT** for result correctness.

7.2.3 The intent of result validation is to validate all the use cases against SF1 and compare it against the reference result set packaged with the benchmark kit. This is exercised against the **SUT** before the Benchmark Run as part of **SUT** Validation Test.

7.2.4 Populate the **SUT** with SF1 dataset and schema information.

7.2.5 Execute the use cases using same standard parameters as will be used by the benchmark run. Verify the report generated by the driver by comparing the output to the reference result set.

7.2.6 Result Validation Process is defined in TPCx-AI_Validation.sh script and the generated report shall state that the output of scoring is similar of better than the reference accuracy metrics for all use cases.

7.2.7 The steps are provided below:

- *ENGINE_VALIDATION_DATAGENERATION* : This phase as defined in TPCx-AI_Validation.sh generates a dataset at a fixed scale factor of 1 (SF1).
- *ENGINE_VALIDATION_LOAD_TEST*: During this phase, as defined in TPCx-AI_Validation.sh the data generated (Clause 7.3.6.1) will be loaded into the final location from where they will be eventually accessed to execute each one of subsequent benchmark tests (Clause 7.3.1).
- *ENGINE_VALIDATION_POWER_TRAINING_TEST*: During this phase as defined in TPCx-AI_Validation.sh, all use cases' training phase will be run in sequence and the results are stored in persistent storage.
- *ENGINE_VALIDATION_POWER_SERVING_TEST*: During this phase as defined in TPCx-AI_Validation.sh, all use cases' serving phase will be run in sequence and the results are stored in persistent storage.
- *ENGINE_VALIDATION_RESULT_VALIDATION*: During this automated phase as defined in TPCx-AI_Validation.sh, the benchmark driver compares the accuracy results from all **use cases** against a known reference results packaged with the kit.

7.2.8 The elapsed time for Validation Test is not included as part of Benchmark Metric calculation.

7.2.9 The elapsed time for Validation Test is not counted as part of Benchmark Execution.

7.2.10 For all other scale factors, used in the Benchmark Run, the benchmark driver at the end of the benchmark performs output validation checking for the presence of output data from power test and throughput test in order to qualify successful benchmark execution.

7.2.11 Output data for Validation test to be verified:

7.2.11.1 The training phase of each use case has generated a model file successfully

7.2.11.2 The model file for each use case is used to conduct the serving test for that use case successfully

7.2.11.3 The model file for each use case is also used to conduct the scoring tests for that use case.

7.2.11.4 The scoring results for each use case should meet or beat the reference result set provided in the Kit.

Use Case	Quality metric name
UC1	K means clusters
UC2	Word error rate
UC3	Forecast accuracy
UC4	Matthews correlation coefficient
UC5	Prediction quality (RMSLE)
UC6	F1 score
UC7	Mean Absolute Error
UC8	Classification accuracy
UC9	Face recognition accuracy
UC10	Classification accuracy

Figure 7-a Validation Test Accuracy metrics

7.2.11.5 Software library versions for all libraries used must be listed.

7.2.12 Software library versions for TPCx-AI approved compute libraries are defined below and must be used in the result validation.

Library	Distribution	Version
Horovod	Apache 2.0	0.19
MLCXGBoost	Apache 2.0	1.1
Tensorflow	Apache 2.0	2.1.0
Scallop	BSD 3	3.1
joblib	BSD 3 clause License	1.0
pandas	BSD 3 clause License	1.2
sklearn	BSD 3 clause License	0.24
Opencv-Python	MIT	4.2
typing	Python Software Foundation license	3.7
pyyaml	MIT	5.0
numpy	BSD 3 clause License	1.20
Statsmodels.tsa.holtwinters	BSD 3 clause License	0.12
tqdm	MIT	4.0
surprise	BSD 3 clause License	1.1
scipy	BSD 3 clause License	1.6
xgboost	Apache	1.3
imblearn	MIT	0.7

Figure 7-b TPCx-AI approved compute software list

7.3 Benchmark Run

7.3.1 All TPCx-AI tests are initiated by the TPCx-AI master scripts which can be executed from any of the nodes in the **SUT**. The tests are listed below:

a) Load Test
b) Power Training Test

c) Power Serving Test I
d) Power Serving Test II
e) Throughput Test
f) Scoring

7.3.2 A valid run consists of 6 separate tests run sequentially. These tests may not overlap in their execution times. For example, the start of Power Serving Test I may not begin until Power Training Test is complete, the start of Power Serving Test II may not begin until Power Serving Test I is complete, etc.

7.3.3 The **Test Sponsor** sets the Benchmark Driver Parameters used during the tests are set per Appendix B.

7.3.4 The elapsed time for each test in Clause 7.3.1 must be reported. However, the elapsed time of the Scoring test will not be considered for the computation of the final performance metric. Instead, the results of the Scoring test are used to determine whether the **Performance Run** was a successful one based on whether the thresholds of the quality metrics for each use case were met or not.

7.3.5 Parameters *BENCHMARK_START* and *BENCHMARK_STOP* in the Run Report determine the overall elapsed time for the Performance Run.

7.3.6 Load Test

7.3.6.1 The process of copying the input dataset files to the final location from where they will be eventually accessed to execute each one of the subsequent benchmark phases (Clause 7.3.1) is known as the Load Test. The Load Test consists of the following phases:

a) Data Generation
b) Optional Data relocation
c) Optional Data preparation

7.3.6.2 The **Load Test** must not include the execution of any of the queries in the **Power Test** or **Throughput Test** or any similar query.

7.3.6.3 Data Generation

7.3.6.3.1 The process of using **PDGF** to create the data in a format suitable for presentation to the load facility. **PDGF** generates different types of data, including images, audio, and text-based flat files. PDGF generates data to local file systems in the SUT from where they can either:

a) be read directly by the workloads to start execution of the **Use Cases**, or
b) be copied to a distributed file system from where eventually will be read as part of the **Use Cases** execution.

7.3.6.3.2 If after **PDGF** generates data in the local file systems of the SUT, the **Use Cases** are run directly against the data in this location, then the data generation is considered to contribute to the load time. However, if after **PDGF** generates the data in the local file systems of the SUT**, another process is needed to move the data to a final location** from where eventually they will be read as part of the **Use Cases** execution, then the generation time is not included in the load time.

7.3.6.4 Optional Data Relocation: Copy to final Dataset Location.

7.3.6.4.1 If the location of the **PDGF** output is different from the final Dataset Location, the data must be copied into the final Dataset Location. This phase is timed and contributes to the load time. Note that this copy may be done as part of the optional format conversion in the Data Preparation phase, in which case the time is captured as part of the Data Preparation timing. If multiple data copies occur between the **PDGF** generation and the placement of the data in the final Dataset Location, only the final copy into the Dataset location is included in the load time. For example, if **PDGF** generates data initially to a location external to the **SUT**, the flat files are subsequently copied to a staging area on the **SUT**, and then the data is copied again from the staging area into the Dataset Location as part of the Data Preparation format conversion, only the final copy is included in the load time.

7.3.6.5 Optional Data preparation

7.3.6.5.1 Example of an optional data preparation phase includes all additional work, beyond the Generation and Relocation steps, required to prepare the data for the use case execution. This includes but is not limited to the following steps:

a) Creation of **Metadata**.
b) Computing statistics for the dataset.
c) Conversion of the data into an alternative or optimized format. An example would be conversion from the row-oriented format in the flat files to a compressed and/or columnar format. Note this is an optional step – if the flat files have been placed in the final Dataset Location by earlier load steps, then it's permissible to run data preprocessing directly against the flat files in their original format in the Dataset Location.

7.3.6.5.2 Any format conversion or creation of auxiliary data structures must meet the following requirements:

a) it must not lose information from the original **Test Dataset**.

b) it cannot make use of any knowledge of the **workloads** in the benchmark.

7.3.6.5.3 For example, the conversion can't remove columns that aren't referenced by the benchmark **use case execution**, and creation of materialized views that pre-compute some or all the results needed for preprocessing, training, serving or scoring is not allowed.

7.3.6.5.4 All work done during Data Preparation is timed and included in the load time.

7.3.6.5.5 Any additional work required to prepare the data for execution (e.g. format conversion) is timed and included in the load time.

7.3.7 Power Training Test

7.3.7.1 Power training test determines the maximum speed the **SUT** can process the Training of all 10 use cases. The Training pipelines of all use cases **must run sequentially first.** The result of the training test should be the generation of a training model file at the completion of the training stage of each of the use cases.

7.3.7.2 The Power training test is timed and included as part of the overall metric

7.3.8 Power Serving Test I and Power Serving Test II

7.3.8.1 Power Serving tests determine the maximum speed the **SUT** can process the Serving stages of all 10 use cases.

7.3.8.2 The Serving Tests follow the Power training Test after the model files for all the use cases have been created.

7.3.8.3 There are 2 Power Serving tests run sequentially:

a) Power Serving 1 Test

b) Power Serving 2 Test

7.3.8.4 There shall be no change in configurations or tuning between the two serving tests.

7.3.9 Throughput Test

7.3.9.1 The Throughput Test runs all use case serving phases concurrently. Each stream runs all **use cases** serving pipelines according to the placement order mentioned in Clause 7.3.9.2. The Default streams for throughput test is set to 2, the number of concurrent streams is configurable with no maximum limit.

7.3.9.2 Query placement in the serving throughput test is performed using the automatic shuffling of the streams. Query placement for 100 streams are shown in Appendix D.

7.3.10 Scoring Test

7.3.10.1 The Scoring Test generates a small dataset with ground truth labels

7.3.10.2 During this scoring test, a separate serving phase is executed in sequence for all 10 use cases against a newly generated data (excluding the truth labels) and the resulting labels from separate serving phase are compared with the ground truth labels to determine the accuracy metric or error incurred by the use case.

7.3.10.3 The scoring test is not part of the timed components of the benchmark. However, this step is crucial to determine whether the quality threshold defined for each use case is met.

Use Case	Quality metric name
UC1	K means clusters
UC2	Word error rate
UC3	Forecast accuracy
UC4	Matthews correlation coefficient
UC5	Prediction quality (RMSLE)
UC6	F1 score
UC7	Mean Absolute Error
UC8	Classification accuracy
UC9	Face recognition accuracy
UC10	Classification accuracy

Figure 7-c Benchmark Run Accuracy Metrics

7.3.11 The **Reported Performance Metric** is the TPCx-AI **Performance Metric** for the **Benchmark Run**. There must not be any interruption during the tests, and all tests should be run without intervention.

7.4 Configuration and Tuning

7.4.1 The **SUT** cannot be reconfigured, changed, or re-tuned by the **Test Sponsor** during or between any of the tests described in Clause 7.3.1.

7.4.2 Any manual tunings to the **SUT** must be performed before the beginning of the benchmark execution described in Clause 7.1.2, and must be fully disclosed.

7.4.3 Automated changes and tuning performed on the **SUT** between any of the tests are allowed.

7.4.4 Any changes to default tunings or parameters of the applications, **Operating Systems**, or hardware of the **SUT** must be disclosed.

7.4.5 Any changes deemed with the characteristics of Benchmark Special in Clause 0.3 are not allowed.

7.5 Metrics

7.5.1 TPCx-AI defines three primary metrics:

a) A Performance Metric, AIUCpm@SF, reflecting the TPCx-AI use case throughput (see Clause7.5.6)
b) A Price-Performance metric, $/AIUCpm@SF (see Clause 7.5.77.5.6)
c) System availability date (see Clause 7.5.8).

7.5.2 Secondary metrics are additional metrics defined below are provided as part of the **Report :**

a) Computed Load Metric T_LD as defined in Clause 7.5.4
b) Computed Power Training Test Metric T_{PTT}
c) Computed Power Serving Test Metric T_{PST}
d) Computed Throughput Test Metric T_{TT}
e) Elapsed time for each Use Case in load test and all the Power and throughput tests.
f) Accuracy metrics for each use case
g) When TPC_Energy option is chosen for reporting, the TPCx-AI energy metric reports the power per performance and is expressed as Watts/AIUCpm@SF. (see TPC-Energy specification for additional requirements).

7.5.3 Each secondary metric shall be referenced in conjunction with the scale factor at which it was achieved. For example, Load Time references shall take the form of Load Time @ SF, or "Load Time = 10 hours @ 300GB".

7.5.4 The **Performance Metric** of the TPCx-AI benchmark, AIUCpm@SF, is computed by combining metric components representing the load, power, and throughput tests:

7.5.4.1 T_{LD} is the load factor computed as:

$$\mathbf{T_{LD}} = \mathbf{0.3} * \boldsymbol{T_{Load}}$$

7.5.4.1.1 Where T_{Load} is the elapsed time of the Load Test (Clause 7.3.6) in seconds and 0.3 is a multiplication factor to adjust the contribution of Load test in the performance metric.

7.5.4.2 T_{PTT} is the geometric mean of the elapsed time UT in seconds of each of the Use case Training times as measured during the Power Training Test (Clause 7.3.7), multiplied by the number of **Use cases** in the benchmark:

$$\mathrm{T_{PTT}} = 10 * \sqrt[10]{\prod_{i=1}^{i=10} \mathrm{UT}(i)}$$

7.5.4.2.1 Where UT(*i*) is the elapsed time in seconds of the **Use Case** *i* during the Power Training Test and 10 is the number of **Use cases** in the Benchmark.

7.5.4.3 The Performance Run contains 2 power serving tests (Clause 7.3.8).

7.5.4.4 T_{PST}, T_{PST1} & T_{PST2} are derived as follows:

7.5.4.5 T_{PST1} is the geometric mean of the elapsed time US in seconds of each of the **Use case** Serving times as measured during the Serving Power Test I (Clause a)7.3.8), multiplied by the number of **Use cases** in the benchmark:

$$T_{PST1} = 10 * \sqrt[10]{\prod_{i=1}^{i=10} US1(i)}$$

7.5.4.5.1 Where US1(*i*) is the elapsed time in seconds of the **Use Case** *i* during the Serving Power Test I and 10 is the number of **Use cases** in the Benchmark.

7.5.4.6 T_{PST2} is the geometric mean of the elapsed time US in seconds of each of the **Use case** Serving times as measured during the Serving Power Test II (Clause 7.3.8), multiplied by the number of **Use cases** in the benchmark:

$$T_{PST2} = 10 * \sqrt[10]{\prod_{i=1}^{i=10} US2(i)}$$

7.5.4.6.1 Where US2(*i*) is the elapsed time in seconds of the **Use Case** *i* during the Serving Power Test II and 10 is the number of **Use cases** in the Benchmark.

7.5.4.7 T_{PST} is the lower of the two Serving Power tests T_{PST1} & T_{PST2}

7.5.4.8 T_{TT} is the throughput test metric computed as the total elapsed time of the throughput test divided by the number of streams as measured during the Throughput Test (Clause 7.3.9).

7.5.4.8.1 T_{Tput} is the elapsed time of all streams from the Throughput Test.

- *n* is the number of streams in the Throughput Test (Clause 7.3.9).

$$T_{TT} = \frac{1}{n} T_{Tput}$$

7.5.5 The Performance Metric (AIUCpm@SF)

The primary performance metric of the benchmark is AIUCph@SF, the effective query throughput of the benchmarked configuration, defined as:

$$AIUCpm@SF = \frac{SF*N*60}{\sqrt[4]{T_{PTT}*T_{PST}*T_{TT}*T_{LD}}}$$

Where:

- SF is defined in Clause 3, and is based on the scale factor used in the benchmark
- N is the total number of use cases executed in a Run
- T_{PTT} where T_{PTT} is the total elapsed time to complete the Training Power Test, as defined in Clause 7.3.7,
- T_{PST} is the total elapsed time to complete the serving power test. This time is the lower time of the two Power serving test times (T_{PST1} & T_{PST2})
- T_{TT} is the total elapsed time of Throughput Test as defined in Clause 7.3.9.
- T_{LD} is the load factor computed as $T_{LD}=0.3*T_{Load}$, and T_{Load} is the time to finish the load, as defined in Clause 7.3.6.
- T_{PTT}, T_{PST}, T_{TT} and T_{LD} quantities are in units of decimal hours with a resolution of at least $1/3600^{th}$ of an hour (i.e., 1 second)

7.5.6 The Price Performance Metric $/AIUCpm@SF
The price-performance metric for the benchmark is defined as:

$$\$ / AIUCpm@SF = \frac{P}{AIUCpm@SF}$$

Where:

- P is the price of the Priced System as defined in Clause 8.2
- AIUCpm@SF is the reported performance metric as defined in Clause 7.5.5.

7.5.7 If a benchmark configuration is priced in a currency other than US dollars, the units of the price-performance metrics may be adjusted to employ the appropriate currency.

7.5.8 The System Availability Date, as defined in the TPC Pricing Specification must be disclosed in any references to either the performance or price-performance metric of the benchmark.

7.5.9 Fair Metric Comparison

7.5.9.1.1 Results at the different scale factors are not comparable, due to the substantially different computational challenges found at different data volumes. Similarly, the system price/performance may not scale down linearly with a decrease in dataset size due to configuration changes required by changes in dataset size.

7.5.10 If results measured against different dataset sizes (i.e., with different scale factors) appear in a printed or electronic communication, then each reference to a result or metric must clearly indicate the dataset size against which it was obtained. In particular, all textual references to TPCx-AI metrics (performance or price/performance) appearing must be expressed in the form that includes the size of the test dataset as an integral part of the metric's name, i.e. including the "@size" suffix. This applies to metrics quoted in text or tables as well as those used to annotate charts or graphs. If metrics are presented in graphical form, then the test dataset size on which metric is based must be immediately discernible either by appropriate axis labeling or data point labeling.

7.5.11 In addition, the results must be accompanied by a disclaimer stating:

a) "The TPC believes that comparisons of TPCx-AI results measured against different dataset sizes are misleading and discourages such comparisons".
b) Any TPCx-AI result is comparable to other TPCx-AI results regardless of the number of query streams used during the test (as long as the scale factors chosen for their respective test datasets were the same).

7.5.12 Required Reporting Components
To be compliant with the TPCx-AI standard and the TPC's fair use policies, all public references to TPCx-AI results for a given configuration must include the following components:

- The size of the test data expressed separately or as part of the metric's names (e.g., AIUCph@10GB).
- The TPCx-AI Performance Metric, AIUCpm@Size.
- The TPCx-AI Price/Performance metric, $/AIUCpm@Size.
- The **Availability Date** of the complete configuration (see TPC Pricing Specification located on the TPC website (http://www.tpc.org).

7.5.13 The following is an example of compliant reporting of TPCx-AI results:
Example 1: At 10GB the RALF/3000 Server has a TPCx-AI use case per hour metric of 3010 when run against a 10GB dataset yielding a TPCX-AI Price/Performance of $1,202 per use case-per-hour and will be available 26-Jan-21.

8 PRICING

8.1 Introduction

This section defines the components, functional requirements of what is priced, and what Substitutions are allowed. How Substitutions are performed is defined in TPC Pricing Specification. Rules for pricing the Priced Configuration and associated software and maintenance are included in the TPC Pricing Specification located at www.tpc.org.

8.1.1 Pricing Methodology

8.1.1.1 A 1-Year Pricing Methodology (as defined in the TPC Pricing Specification) must be used to calculate the price and the price/performance result of the TPCx-AI benchmark.

8.1.1.2 The Pricing Model 1 – Default Pricing Model (as defined in the TPC Pricing Specification) is the only pricing model allowed in a TPCx-AI result.

8.2 Priced Configuration

8.2.1 The system to be priced must include the hardware and software components present in the System Under Test (SUT), a communication interface that can support user interface devices, additional operational components configured on the test system, and maintenance on all of the above

8.2.2 Calculation of the priced configuration consists of:

a) Price of the SUT as tested and defined in Clause 6.1

8.2.3 Price of all software licenses for software used in the SUT

8.2.4 Price of a communication interface capable of supporting the required number of user interface devices

8.2.5 Price of additional products (software or hardware) required for customary operation, administration and maintenance of the SUT for a period of 1 year

8.2.6 Price of all products required to create, execute, administer, and maintain the executables or necessary to create and populate the test environment

Comment: Note that Clause 7.3.6.4 explicitly permits data generation to be external to the **SUT** in certain situations. In these situations, the products required for such external to **SUT** data generation would not be priced if the auditor is satisfied that the solution meets the requirements of Clause 7.3.6.4

8.3 Specifically excluded from the priced configuration calculation are:

a) end-user communication devices and related cables, connectors, and switches.

Comment: end-user communication device here means driver node used to start, stop and orchestrate the benchmark, however devices used to connect to the end-user device with its cable is part of pricing.

b) equipment and tools used exclusively in the production of the **Full Disclosure Report**

8.4 Additional Operational Components

8.4.1 Additional products included on a customer installed configuration are also to be included in the priced configuration if explicitly required for the operation, administration, or maintenance of the priced configuration. Examples of such products are:

a) operator console
b) user interface terminal
c) CD drive
d) software, if required for initial load or maintenance updates
e) all cables used to connect components of the **SUT** except as noted in Clause 8.1.1.2

8.5 Allowable Substitutions

8.5.1 **Substitution** is defined as a deliberate act to replace components of the **Priced Configuration** by the **Test Sponsor** as a result of failing the availability requirements of the TPC Pricing Specification or when the part number for a component changes. This also requires compliance with the TPC Pricing Specification.

Comments: Corrections or "fixes" to components of the **Priced Configuration** are often required during the life of products. These changes are not considered **Substitutions** so long as the part number of the priced component does not change. Suppliers of hardware and software may update the components of the **Priced Configuration**, but these updates

must not negatively impact the reported **Performance Metric** or numerical quantities more than two percent. The following are not considered **Substitutions**:

a) Software patches to resolve a security vulnerability
b) Silicon revision to correct errors
c) New supplier of functionally equivalent components (for example memory chips, disk drives, etc.)

8.5.2 Some hardware components of the **Priced Configuration** may be substituted after the **Test Sponsor** has demonstrated to the **Auditor's** satisfaction that the substituting components do not negatively impact the reported **Performance Metric** or numerical quantities. All **Substitutions** must be **Reported** in the **FDR** and noted by the **Auditor.** The following hardware components may be substituted:

a) durable medium (for example disk drives) and cables
b) durable medium enclosure
c) network interface card
d) router
e) bridge
f) network switch
g) repeater

Comment: If any hardware component is substituted, then the result must be audited by an Auditor

9 FULL DISCLOSURE

9.1 Full Disclosure Report Requirements

9.1.1 A **Full Disclosure Report (FDR)** is required. This section specifies the requirements of the **FDR**.

The **FDR** is a zip file of a directory structure containing the following:

- A **Report** in Adobe Acrobat PDF format
- An **Executive Summary Statement** in Adobe Acrobat PDF format
- The **Supporting Files** consisting of any source files, configuration files, or scripts modified by the **Test Sponsor** and the output files generated by the TPCx-AI kit. Requirements for the **FDR** file directory structure are described below.

9.1.2 The **FDR** should be sufficient to allow an interested reader to evaluate and, if necessary, recreate an implementation of the TPCx-AI **Result** given the appropriate hardware and software products. If any sections in the **FDR** refer to another section of the **FDR**, the names of the referenced scripts/programs must be clearly labeled in each section. Unless explicitly stated otherwise, "disclosed" or "reported" refers to disclosing or reporting in the **FDR**.

Comment: Since the test environment may consist of a set of scripts and corresponding input files, it is important to disclose and clearly identify, by name, the scripts and input files in the **FDR**.

9.2 Format Guidelines

9.2.1 While established practice or practical limitations may cause a particular benchmark disclosure to differ from the examples provided in various small ways, every effort should be made to conform to the format guidelines. The intent is to make it as easy as possible for a reviewer to read, compare, and evaluate material in different benchmark disclosures.

9.2.2 All sections of the report, including appendices, must be printed using font sizes of a minimum of 8 points.

9.2.3 The Executive Summary must be included near the beginning of the Report.

9.2.4 The order and titles of sections in the Report and Supporting Files must correspond with the order and titles of sections from the TPCx-AI Specification (i.e., this document). The intent is to make it as easy as possible for readers to compare and contrast material in different Reports.

9.2.5 The directory structure of the **FDR** has three parts:

a) Executive Summary Statement - contains Executive Summary statement
b) Report - contains Report
c) Supporting Files - contains Supporting Files

9.3 General Items

9.3.1 The **FDR** must follow all reporting rules of the TPC Pricing Specification, located at www.tpc.org. For clarity and readability, the TPC Pricing Specification requirements may be repeated in the TPCx-AI Specification.

9.3.2 A statement identifying the benchmark **Test Sponsor** and other participating companies must be provided.

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

- Configuration parameters and options for server, storage, network and other hardware components used by the **SUT**.
- Configuration parameters and options for the **Operating System** and file system components used by the **SUT**.
- Configuration parameters and options for any other software components (e.g. compiler optimization options) used by the **SUT**.

Comment: In the event that some parameters and options are set multiple times, it must be easily discernible by an interested reader when the parameter or option was modified and what new value it received each time.

Comment: This requirement can be satisfied by providing a full list of all parameters and options, as long as all those that have been modified from their default values have been clearly identified and these parameters and options are only set once.

9.3.4 Explicit response to individual disclosure requirements specified in the body of earlier sections of this document must be provided.

9.3.5 Diagrams of both measured and priced configurations must be provided, accompanied by a description of the differences. This includes, but is not limited to:

- total number and type of nodes used
- total number and type of processors used/total number of cores used/total number of threads used (including sizes of L2 and L3 caches)
- size of allocated memory, and any specific mapping/partitioning of memory unique to the test.
- number and type of data storage units disk units, controllers, and if applicable, **LCS** volumes
- number of channels or bus connections to disk units, including their protocol type
- number of LAN (for example, Ethernet) connections and speed for switches and if applicable, other hardware components used in the test or are incorporated into the pricing structure
- type and the run-time execution location of software components

Clause 6 illustrates a measured benchmark configuration where each server uses Ethernet connections, an external driver, and two processors each with sixteen cores and two threads per core in the **SUT**. Note that this diagram does not depict or imply any optimal configuration for the TPCx-AI benchmark measurement.

Depending on the implementation of the **SUT**, the configuration diagram must include any key functional entities that were used during the execution of the benchmark. Examples of such entities include **Name Node**, Secondary **Name Node**, **Data Node**, Job/Task Tracker, Resource Manager/Node Manager, etc.

Figure 9-a Sample Configuration Diagram

- n x Server Rack in scale out configuration.
- n x My Server Model B, 4/32/64 My CPU Model Z (2.7 GHz, 20MB cache, 130W), 128GB, My RAID Controller with 1GB BBWC
- n x My Storage Array Model A with 8 X 1TB 10K SAS HDD
- n x My Switch Model X 10GbE
- n x Top of the Rack switch.
- **LCS** results can show LCS instance configuration instead of physical hardware equipment.

Comment: Detailed diagrams for system configurations and architectures can vary widely, and it is impossible to provide exact guidelines suitable for all implementations. The intent here is to describe the system components and connections in sufficient detail to allow independent reconstruction of the measurement environment. This example diagram shows homogeneous nodes. This does not preclude **Test Sponsors** from using heterogeneous nodes if the system diagram reflects the correct system configuration.

9.4 Software Components and Dataset Distribution

The distribution of software components, roles and dataset across all media must be explicitly described using a format similar to that shown in the following example for the tested and priced configuration.

Server	**Role(s)**	**Count**	**Virtual**	**Host Name(s)**	**HW/SW Configuration**	**Storage Setup**
Worker	Yarn NM/ Spark Worker	50	N	TPCx-AI[100-150]	• Vendor Server Model Name. • HW/SW Config (Processor Model, socket count, Frequency, Core count). • DRAM capacity. • Storage x HDD Model. • Network and BW link speed. • OS Model and version. • Framework SW Model and version. • Graphics Adapters or Accelerators • Details of Additional HW/SW if any.	**OS: Model x GB SSD, Intermediate/Shuffle/Temp Data: x Model x GB SSD, Distributed FS: x Model 12x SAS/SATA Hard drive/**
Distro Manger	Cloudera Manager	1	N	TPCx-AI-CM	• Vendor Server Model Name. • HW/SW Config (Processor Model, socket count, Frequency, Core count). • DRAM capacity. • Storage x HDD Model. • Network and BW link speed. • OS Model and version. • Framework SW Model and version. • Details of Additional HW/SW if any.	**OS: Model x GB SSD.**

Gateway SUT Driver	YARN/SPARK Gateway	1	N	TPCx-AI-Driver 1	• Vendor Server Model Name. • HW/SW Config (Processor Model, socket count, Frequency, Core count). • DRAM capacity. • Storage x HDD Model. • Network and BW link speed. • OS Model and version. • Framework SW Model and version. • Details of Additional HW/SW if any.	
Name Node/Resource Manager	YARN/NN/Zookeeper	1	N	TPCx-AI_NN1	• Vendor Server Model Name. • HW/SW Config (Processor Model, socket count, Frequency, Core count). • DRAM capacity. • Storage x HDD Model. • Network and BW link speed. • OS Model and version. • Framework SW Model and version. Details of Additional HW/SW if any.	

Table 9-a Example Layout Description

9.4.1 The distribution of various software components across the system must be explicitly described using a format similar to that shown in Table-3 in Clause 9.4 for both the tested and priced configuration.

Comment: Software components might vary from across different implementations.

9.4.2 The file system used in all nodes must be disclosed as well as any distributed file system used during the execution of the benchmark and its client API version. The report must clearly state in what file system the input dataset was generated and whether the data was moved to a different file system during the Load Test (Clause 7.3.6).

9.4.3 **All Frameworks** and tools used in the benchmark should be disclosed in the report (e.g. Python, HDFS, Spark, YARN, MPI, TensorFlow, Java).

9.4.4 If there were any additional vendor supported patches applied to the **SUT**, details of such patches should be disclosed.

9.5 Workload Related Items

9.5.1 Any script or text used to set for all hardware and software tunable parameters must be included in the **Report**.

9.5.2 Version number of the TPCx-AI kit must be Included in the **Report**.

9.5.3 The Run Report generated by the TPCx-AI benchmark kit must be included in the **Report**.

9.5.4 Elapsed times of all Use Cases during the power and throughput **tests must be** reported from the **Performance Run**, grouped respectively as Machine Learning (ML) or Deep Learning (DL); Structured, semi-structured or unstructured depending on the characteristics of the input data set.

9.5.5 **C**ompletion times for individual **Use Cases** run as part of the **Performance Run** should be included in the **Report**.

9.5.5.1 Output report from successful **SUT** Validation test must be included in the **Report** (Clause 7.2)

9.5.5.2 Global Benchmark **Parameter** files (Clause 5.4.1) must be included in the **Report**.

9.5.5.3 Use case specific configuration parameters (Clause 5.4.1) must be included in the **Report**.

9.6 SUT Related Items

9.6.1 Details of any Specialized Hardware/Software used in the **SUT** must be included in the report.

9.6.2 Relevant **Framework** configuration files from **SUT**, for the **Performance Run** must be included in the report e.g. Yarn-Site.xml, Hdfs-site.xml etc.

9.6.3 **General execution environment** information as well as any special environment configuration that is relevant to the benchmark must be included in the report in form of envinfo.log from a representative worker node from every role in the server.

9.6.4 The data storage ratio must be disclosed. It is computed by dividing the total physical data storage present in the **Priced Configuration** (expressed in TB) by the chosen Scale Factor as defined in Clause 3. Let r be the ratio. The **Reported** value for r must be rounded to the nearest 0.01. That is, reported value=round(r, 2). For example, a **SUT** configured with 96 disks of 1TB capacity for a 1TB Scale Factor has a data storage ratio of 96.

Comment: For the reporting of configured data storage capacity, terabyte (TB) is defined to be 10^12 bytes.

Comment: For consumption-based storage provisioning in **LCS**, the maximum storage provisioned during the entire benchmark test is considered to be the total physical data storage present.

9.6.5 The Scale Factor to memory ratio must be disclosed. It is computed by dividing the Scale Factor by the total physical memory present in the **Priced Configuration**. Let r be this ratio. The **Reported** ratio must be rounded to the nearest 0.01. That is, reported value=round(r,2). For example, a system configured with 1TB of physical memory for a 10TB Scale Factor has a memory ratio of 10.00.

Comment: For LCS, the maximum provisioned memory during the entire benchmark test is considered to be the total physical memory present.

9.7 Metrics and Scale Factors

9.7.1 The **Reported Performance Metric** (AIUCpm@SF for the Benchmark **Run**) must be disclosed in the **Report**.

9.7.2 The **Price/Performance Metric** ($/AIUCpm@SF) for the Benchmark **Performance Run** must be disclosed in the **Report**.

9.7.3 The Scale Factor used for the **Result** must be disclosed in the **Report.**

9.7.4 The number of streams in the throughput run used for the **Result** must be disclosed in the **Report.**

9.7.5 The total elapsed time for the execution of the Benchmark **Performance Run** must be disclosed in the **Report.**

9.7.6 The time for each of the six tests (Clause 7.3.1) must be disclosed for the Benchmark **Performance Run.**

9.7.7 The scoring metrics resulting from the Scoring test of the Benchmark Performance Run must be disclosed in the report.

9.8 Audit Related Items

If the benchmark is audited by an independent **Auditor**, the **Auditor's** agency name, address, phone number, and **Attestation Letter** with a brief audit summary report indicating compliance must be included in the **Report**. A statement should be included specifying whom to contact to obtain further information regarding the audit process.

9.8.1 Executive Summary Statement

The **Executive Summary** is a high-level overview of a TPCx-AI implementation. It should provide the salient characteristics of a benchmark execution (metrics, configuration, pricing, etc.) without the exhaustive detail found in the **FDR**. When the TPC-Energy optional reporting is selected by the **Test Sponsor**, the additional requirements and format of TPC-Energy related items in the **Executive Summary** are included in the TPC Energy Specification, located at www.tpc.org.

9.8.2 The **Executive Summary** has three components:

- Implementation Overview
- Pricing Spreadsheet
- Numerical Quantities

9.8.3 Page Layout

Each component of the **Executive Summary** should appear on a page by itself. Each page should use a standard header and format, including:

- 1/2-inch margins, top and bottom
- 3/4-inch left margin, 1/2-inch right margin
- 2 pt. frame around the body of the page. All interior lines should be 1 pt.

9.8.4 Implementation Overview

The implementation overview page contains five sets of data, each laid out across the page as a sequence of boxes using 1 pt. rule, with a title above the required quantity. Both titles and quantities should use a 9-12 pt. Times font unless otherwise noted.

9.8.4.1 The first section contains information about the sponsor and system identification.

Title	**Font**
Sponsor Name or Logo	16-20 pt. Bold (for Name)
System Identification	16-20 pt. Bold
Version Numbers for TPCx-AI, TPC-Pricing and TPC-Energy (if reported)	16-20 pt. Bold
Report Date	16-20t. Bold

Table 9-b Sponsor and System Identification

Comment: It is permissible to use or include company logos when identifying the sponsor.

Comment: The report date must be disclosed with a precision of 1 day. The precise format is left to the **Test Sponsor**.

9.8.4.2 The second section contains the Total System Cost, the TPCx-AI **Reported Performance Metric** and the **Price/Performance Metric**.

Title	Quantity	Precision	Font
Total System Cost	1 yr. Cost of ownership (Clause 8.1.1.2)	1	16-20 pt. Bold
Reported Performance Metric	AIUCpm (Clause 7.5.5)	0.01	16-20 pt. Bold
Price/Performance	$/ AIUCpm (Clause 7.5.6)	0.01	16-20 pt. Bold

Table 9-c Benchmark Results

9.8.4.2.1 Depending on the currency used for publication this $ sign must be changed to ISO currency symbol.

9.8.4.3 The third section contains detailed diagrams of the measured configuration (Clause 9.3.5) and the Software components distribution table (Clause 9.4)

Title	Quantity	Font
Framework /Engine Software	Product name and Product Version	9-12 pt. Times
Operating System	Product name, **Software Version** of OS, File System Type and Version	9-12 pt. Times
Other Software	Product name and **Software Version** of other software components (example Java)	9-12 pt. Times
System **Availability Date**	The **Availability Date** of the system, defined in the TPC Pricing Specification	9-12 pt. Times
SF (Scale Factor)	SF in as defined in Clause 3	16-20pt. Bold
Streams	Concurrent Streams used in Clause	16-20pt. Bold

Table 9-d System Configuration Information

Comment: The **Software Version** must uniquely identify the orderable software product referenced in the **Priced Configuration** (for example, RALF/2000 4.2.1).

9.8.4.4 The fourth section contains the storage and memory ratios, see (Clause 9.6.)

Title	Precision	Font

Physical Storage/Scale Factor	0.01	9-12 pt. Times
Scale Factor/Physical Memory	0.01	9-12 pt. Times
Main Data Redundancy Model	n/a	9-12 pt. Times

Table 9-e Storage and Memory Ratios

9.8.4.5 The fifth section contains the components, including:

- total number and type of nodes used/total number of processors used with their types and speeds in GHz/ total number of cores used/total number of threads used, see (Clause 9.3.5)
- main and cache memory sizes
- network speed and topology (e.g. Top of the Rack switch, in-rack switch)
- storage type, quantity and configuration.

9.8.5 Pricing Spreadsheet

9.8.5.1 The major categories in the Price Spreadsheet, as appropriate, are as follows:

- network(s)
- server(s) /node(s)
- storage
- software

9.8.5.2 Discounts (may optionally be included with above major category subtotal calculations).

9.8.6 Numerical Quantities Summary

9.8.6.1 The Numerical Quantities Summary page contains six sets of runtime data, presented in tabular form, detailing the Load Test, Power Training Test, Power Serving Test I, Power Serving Test II, Throughput Test, and Scoring Test for the **Benchmark Run**. Each set of data should be headed by its given title and clearly separated from the other tables.

9.8.6.2 The first section contains the Scale Factor, Number of streams, and **SUT** Validation test status for the reported benchmark execution **Result**.

9.8.6.3 The second section contains measurement results and metric from the **Performance Run**.

Performance Run	
Item Title	**Precision**
Overall Run Start Time	yyyy-mm-dd hh:mm:ss.sss
Overall Run End Time	yyyy-mm-dd hh:mm:ss.sss
Overall Run Total Elapsed Time	hh:mm:ss.sss
Start of Load Test	yyyy-mm-dd hh:mm:ss.sss
End of Load Test	yyyy-mm-dd hh:mm:ss.sss
Load Test Elapsed Time	hh:mm:ss.sss
Start of Power Training Test	yyyy-mm-dd hh:mm:ss.sss
End of Power Training Test	yyyy-mm-dd hh:mm:ss.sss
Power Training Test Elapsed Time	hh:mm:ss.sss
Start of Power Serving Test I	yyyy-mm-dd hh:mm:ss.sss

Performance Run	
Item Title	**Precision**
End of Power Serving Test I	yyyy-mm-dd hh:mm:ss.sss
Power Serving Test I Elapsed Time	hh:mm:ss.sss
Start of Power Serving Test II	yyyy-mm-dd hh:mm:ss.sss
End of Power Serving Test II	yyyy-mm-dd hh:mm:ss.sss
Power Serving Test II Elapsed Time	hh:mm:ss.sss
Start of Throughput Test	yyyy-mm-dd hh:mm:ss.sss
End of Throughput Test	yyyy-mm-dd hh:mm:ss.sss
Throughput Test Elapsed Time	hh:mm:ss.sss
Start of Scoring Test	yyyy-mm-dd hh:mm:ss.sss
End of Scoring Test	yyyy-mm-dd hh:mm:ss.sss
Scoring Test Elapsed Time	hh:mm:ss.sss
Performance Metric (AIUCpm@SF)	x,xxx.xx

Table 9-f Measurement Result for Performance Run

9.8.7 The Third section contains Individual loading, training and serving times and accuracy metrics for each use case.

Use Case	Load Time	Training Time	Serving Time1	Serving Time 2	Throughput Time	Accuracy Metric
UC01	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx
UC02	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx
UC03	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx
UC04	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx
UC05	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx
UC06	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx
UC07	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx
UC08	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx
UC09	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx
UC10	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	hh:mm:ss.sss	xx.xxxxx

Table 9-g Use case performance Metrics

9.8.8 TPCx-AI Run Report

The Run Report per Clause 5.3 from TPCx-AI must be included in the **Report** immediately after the **Executive Summary.**

9.9 Availability of the Full Disclosure Report

The **Full Disclosure Report** must be readily available to the public. The **Report** and Supporting Files must be made available when the **Result** is made public. In order to use the phrase "TPCx-AI Benchmark", the **Full Disclosure Report** must have been previously submitted electronically to the TPC using the procedure described in the TPC Policies and Guidelines document.

9.9.1 The official **Full Disclosure Report** must be available in English but may be translated to additional languages.

9.10 Revisions to the Full Disclosure Report

The requirements for a revision to an **FDR** are specified in the TPC Pricing Specification.

10 AUDIT

This clause defines the audit requirements for TPCx-AI. The auditor needs to ensure that the benchmark under audit complies with the TPCx-AI specification. Rules for auditing Pricing information are included in the TPC Pricing Specification located at www.tpc.org. When the TPC-Energy optional reporting is selected by the test sponsor, the rules for auditing of TPC-Energy related items are included in the TPC Energy Specification located at www.tpc.org.

10.1 General Rules

Before publication, a TPCx-AI **Result** must be certified to be compliant with the spirit and letter of the TPCx-AI Benchmark Standard by an Independent Certified TPC Auditor or a TPCx-AI Pre-Publication Board. The **Test Sponsor** can choose the certification be performed by either by a Certified TPC Auditor or a Pre-Publication Board.

10.2 Independent Audit

The term independent is defined as "the outcome of the benchmark carries no financial benefit to the auditing agency other than fees earned directly related to the audit." The independent audit of the benchmark is described in TPC Policies on www.tpc.org. In addition, the auditing agency cannot have supplied any performance consulting under contract for the benchmark.

In addition, the following conditions must be met:

a) The auditing agency cannot be financially related to the sponsor. For example, the auditing agency is financially related if it is a dependent division of the sponsor, the majority of its stock is owned by the sponsor, etc.

b) The auditing agency cannot be financially related to any one of the suppliers of the measured/priced configuration, e.g., the DBMS supplier, the disk supplier, etc.

0.1.1.1 The **Auditor**'s opinion regarding the compliance of a **Result** must be consigned in an **Attestation Letter** delivered directly to the **Test Sponsor**. To document that a **Result** has been audited, the **Attestation Letter** must be included in the **Report** and made readily available to the public. Upon request, and after approval from the **Test Sponsor**, a detailed audit report may be produced by the **Auditor**.

0.1.2 Pre-Publication Board

The term Pre-Publication Board as defined by the TPC Policies consists of one or more individuals that have been chosen by the TPCx-AI Benchmark Subcommittee to certify **Results** for publication. For TPCx-AI **Results** the Pre-Publication Board consists of 3 members from the TPCx-AI Benchmark Subcommittee. Each member serves a period of six months. The membership will be rotated through the TPCx-AI Benchmark Subcommittee membership. The submission is confidential to the Pre-Publication Board until the **Result** is published. The Pre-Publication Board must complete the review within 10 business days. If no issues are raised within the 10 business day period, the **Result** is considered certified for publication.

0.1.3 Results Based on Existing TPCx-AI Results

A **Test Sponsor** can demonstrate compliance of a new **Result** produced without running any performance test by referring to the certification of a **Result**, if the following conditions are all met:

- The referenced **Result** has already been published by the same or by another **Test Sponsor**.
- The new **Result** must have the same hardware and software architecture and configuration as the referenced **Result**. The only exceptions allowed are for elements not involved in the processing logic of the **SUT** (e.g., number of peripheral slots, power supply, cabinetry, fans, etc.)
- The **Test Sponsor** of the already published **Result** gives written approval for its use as referenced by the **Test Sponsor** of the new **Result**.
- The **Auditor** verifies that there are no significant functional differences between the priced components used for both **Results** (i.e., differences are limited to labeling, packaging and pricing.)
- The **Auditor** or Pre-Publication Board reviews the **FDR** of the new **Result** for compliance.

- If certification is performed by an independent **Auditor**, a new **Attestation Letter** must be included in the **Report** of the new **Result**.

  Comment: The intent of this clause is to allow publication of benchmarks for systems with different packaging and model numbers that are considered to be identical using the same **Performance Run**. For example, a rack mountable system and a freestanding system with identical electronics can use the same Performance Run for publication, with, appropriate changes in pricing.

  Comment: Although it should be apparent to a careful reader that the **FDR** for the two **Results** are based on the same set of performance tests, the **FDR** for the new **Result** is not required to explicitly state that it is based on the performance tests of another published **Result**.

  Comment: When more than one **Result** is published based on the same set of performance tests, only one of the **Results** from this group can occupy a numbered slot in each of the benchmark **Result** "Top Ten" lists published by the TPC. The **Test Sponsors** of this group of **Results** must all agree on which **Result** from the group will occupy the single slot. In case of disagreement among the **Test Sponsors**, the decision will default to the **Test Sponsor** of the earliest publication from the group.

10.3 Audit Checklist

A generic audit checklist is provided as part of this specification. The **generic** audit checklist specifies the requirements that must be checked to ensure a **Result** is compliant with the TPCx-AI Specification. Not only should the TPCx-AI requirements be checked for accuracy but the **Auditor** or Pre-Publication Board must ensure that the **FDR** accurately reflects the **Result**.

Comment: An independent **Auditor** must be used for those audit checklist items that refer to pricing or energy.

10.3.1 Verify that the TPCx-AI provided kit is used and verify its version

10.3.2 Verify that all 6 tests (Load, Power Training, Power Serving I, Power Serving II, Throughput, Scoring) (Clause 7.3) of the **Performance Run** completed with no error reported.

10.3.3 Verify Validation tests (Clause 7.2) is completed with no error reported.

10.3.4 Verify Benchmark Execution has been executed according to Clause 7.1

10.3.5 Verify the Validation test results reported for SF1 matches with reference result set provided with the TPCx-AI kit (Clause 7.2)

10.3.6 Verify that all scripts and source code to implement the benchmark has been included in the **Report**.

10.3.7 Verify Kit run report contains all information mentioned in Clause 5.4

10.3.8 Verify Test Sponsor Kit Modifications Clause 5.7 have been followed to ensure the parameter settings was performed as defined in the specification and required reports, files are provided as part of the FDR.

10.3.9 Verify Clause 5.7.6 and 5.7.7 is followed and no Java code files were modified and no JAR file optimizers were used.

10.3.10 Verify the test execution has produced the required output by checking the logfiles to see if all the **training phases** have created a model file and scoring phases complete with accuracy metrics in the **Report**.

10.3.11 Verify that all components of the **SUT** are commercially available as per the TPC Pricing Specification.

10.3.12 Verify that all components of the **SUT** are included in the pricing.

10.3.13 Verify no aspect of **SUT**, including the dataset size, tuning parameters were changed in the **Performance Run**.

10.3.14 Verify that the SF used for publication is valid according to Clause 3.

10.3.15 Verify that the metrics are **Reported** as per the requirements in Clause 7

10.3.16 Verify that the **SUT** Pricing **Report** is in compliance with the TPC Pricing specification.

Comment: The auditor should also review the **SUT** pricing details in Clause 6.2 to verify the **SUT** Pricing Report

10.3.17 Verify that the Energy report is in compliance with the TPC Energy specification (if reported).

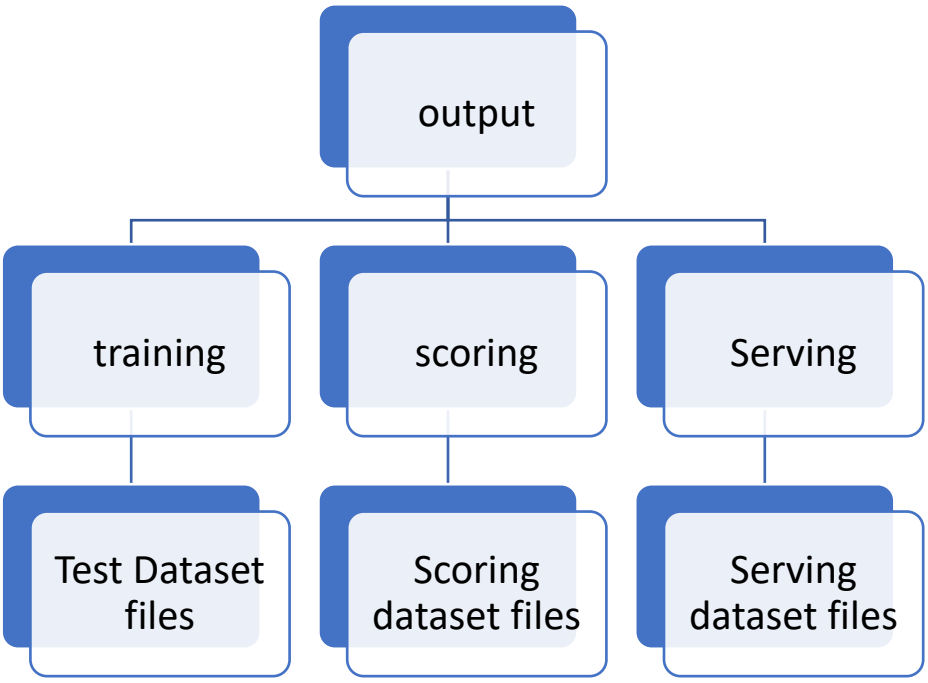
10.3.18 Verify that Full Disclosure Report and Executive Summary Reports are accurately reported and comply with the reporting requirements. This includes but not limited to.

d) metric calculation
e) system availability
f) the diagrams of both measured and priced configuration
g) system pricing
h) the numerical quantity summary
i) Parameter files required as part of FDR are provided.

APPENDIX A – PDGF USER GUIDE

PDGF is a TPC provided software package designed to facilitate the data generation of the TPCx-AI test dataset. This appendix provides information on how a Test Sponsor is to use the features and functionality of PDGF.

PDGF generates the Test Dataset in an output folder along with the dataset for serving and scoring.



With PDGF the **Test Sponsor** specifies the scale factor to be used and accordingly a unified dataset is generated where the size of the Test dataset will be very close to the scale factor specified.

APPENDIX B – BENCHMARK PARAMETERS

Scale factor (-sf): The -sf parameter determines the size of the synthetic data set that will be generated and that will be the input to the Use cases execution.

Configuration file (-c): The -c parameter is used by the test sponsor to specify the configuration file that will be used to run the benchmark.

APPENDIX C – USE CASE SPECIFIC PARAMETERS

Some use cases provide a set of settings that can be changed by the test sponsor prior to running the benchmark. Only the parameters listed below can be changed.

Use case 2

Epochs: Number of epochs for which the neural network will be trained.

Batch: Batch size used during the training and serving workload execution. Different batch sizes can be used for training and for serving.

Use case 5

Epochs: Number of epochs for which the neural network will be trained.

Batch: Batch size used during the training and serving workload execution. Different batch sizes can be used for training and for serving.

Use case 7

Num-blocks: The number of blocks used to parallelize computation (set to -1 to auto-configure)

Use case 8

num-workers: Controls how many parallel workers we want to have when training a XGBoost Classification Model.

Nthread: The number of threads used by each XGBoost worker. Spark requires that all of nthread * numWorkers cores should be available before the training runs.

Use case 9

Epochs_embedding: Number of epochs for which the neural network will be trained.

Batch: Batch size used during the training and serving workload execution. Different batch sizes can be used for training and for serving.

Parameters that are not allowed to be changed:

a) Maximum number of iterations
b) Learning rate

APPENDIX D - THROUGHPUT TEST STREAM PLACEMENT

For convenience the following table displays the order of serving use case templates for the first 100 streams. The order is the same for all scale factors.

Stream	Use cases									
1	3	5	10	6	1	7	4	8	9	2
2	1	4	5	10	3	2	9	6	7	8
3	9	5	2	6	4	10	1	7	8	3
4	4	1	3	9	5	2	10	6	7	8
5	9	8	5	4	10	1	7	3	6	2
6	5	8	9	4	1	3	10	7	2	6
7	1	8	5	2	10	9	6	7	3	4
8	3	8	7	9	4	6	1	2	10	5
9	3	5	1	6	9	2	8	7	10	4
10	6	5	3	9	10	2	4	1	8	7
11	1	5	6	4	10	8	3	7	2	9
12	4	6	2	9	8	1	10	7	5	3
13	9	1	10	5	8	7	3	4	2	6
14	10	5	4	6	7	2	9	3	1	8
15	10	3	8	5	9	4	7	1	2	6
16	10	7	5	1	6	9	8	4	3	2
17	1	8	2	5	9	10	3	7	6	4
18	4	10	6	9	3	1	2	5	7	8
19	3	5	7	6	1	4	10	2	9	8
20	9	10	1	8	3	7	6	2	4	5
21	9	2	8	7	5	10	6	4	3	1
22	9	2	7	10	5	1	3	4	6	8
23	5	7	6	4	8	3	9	10	1	2
24	6	9	2	3	7	5	1	4	8	10
25	5	10	1	3	6	4	7	2	9	8
26	5	8	3	9	4	1	10	7	6	2
27	1	6	9	7	10	4	3	2	5	8
28	3	8	7	4	5	2	1	10	9	6
29	4	7	5	9	2	10	8	1	6	3
30	7	9	5	10	1	6	8	3	2	4
31	9	1	8	4	6	10	3	7	5	2
32	4	5	9	10	2	1	3	6	8	7
33	4	1	5	10	3	8	6	2	7	9
34	2	9	1	3	5	7	4	6	10	8
35	2	6	4	10	7	5	1	3	9	8
36	10	7	1	4	5	3	9	6	2	8
37	5	4	1	3	7	9	8	2	10	6
38	4	2	5	7	8	3	10	1	9	6
39	5	9	7	1	3	4	6	10	2	8
40	1	8	7	6	4	5	2	3	10	9
41	10	5	4	8	3	7	9	2	1	6
42	5	10	6	2	1	7	8	9	4	3
43	4	7	2	6	9	3	10	8	1	5
44	5	10	4	1	8	3	7	9	2	6
45	7	5	9	6	8	2	3	4	1	10
46	3	1	6	10	8	9	4	7	5	2
47	1	4	10	2	3	5	7	8	6	9
48	7	5	1	9	8	4	6	3	10	2
49	10	8	6	1	4	2	5	3	9	7
50	8	7	5	3	6	2	10	4	9	1
51	7	3	6	9	4	8	2	5	10	1
52	6	8	5	3	10	9	2	4	7	1
53	2	5	9	10	3	8	6	4	1	7
54	6	2	3	5	10	4	7	9	8	1
55	8	4	5	1	6	7	3	9	10	2
56	8	2	7	4	5	9	6	3	1	10
57	4	10	8	5	3	7	1	6	2	9
58	2	3	1	4	6	9	5	7	8	10
59	10	4	7	1	2	3	5	8	6	9
60	10	4	3	7	5	1	6	9	8	2
61	4	6	3	7	9	5	8	10	1	2
62	5	8	9	6	10	1	7	3	4	2
63	3	9	10	7	8	2	1	5	6	4
64	8	7	3	4	2	1	5	10	6	9
65	7	5	8	4	1	3	10	6	2	9
66	1	2	6	8	7	10	3	5	4	9
67	7	9	4	5	8	2	10	3	6	1
68	5	2	4	7	9	1	8	3	10	6
69	6	8	5	9	3	4	7	2	10	1
70	3	10	7	5	1	2	9	8	6	4
71	6	10	1	4	2	5	9	7	8	3
72	1	4	3	5	7	10	9	6	8	2
73	6	2	1	3	5	7	4	10	8	9
74	2	5	3	7	10	1	9	4	6	8
75	3	8	7	10	5	1	4	9	6	2
76	4	9	8	5	1	6	10	3	2	7
77	3	4	2	9	1	6	10	7	8	5
78	9	6	8	5	4	2	7	10	3	1
79	9	3	4	2	7	1	5	8	10	6
80	8	2	3	1	7	9	6	4	10	5
81	10	5	6	2	3	8	1	9	7	4
82	6	7	2	9	8	5	10	4	3	1
83	2	9	7	5	8	6	3	10	4	1
84	5	4	10	3	6	8	7	2	9	1
85	6	5	3	9	2	10	4	8	1	7
86	3	7	9	5	2	8	10	6	1	4
87	2	6	9	8	5	7	4	1	10	3
88	8	7	1	4	10	2	3	5	6	9
89	4	9	7	8	6	10	1	2	3	5
90	7	5	1	9	8	2	10	6	3	4
91	10	1	5	7	8	6	4	3	2	9
92	8	3	4	5	9	6	7	10	1	2
93	8	5	7	10	1	2	4	6	9	3
94	9	7	2	8	6	10	5	3	1	4
95	1	10	7	3	9	4	8	2	5	6
96	8	2	1	5	10	7	4	3	9	6
97	6	2	10	7	5	4	8	9	1	3
98	2	4	3	9	5	6	1	10	7	8
99	3	5	1	7	2	4	8	9	6	10
100	9	1	6	8	10	4	7	3	5	2