TPC Benchmark Roadmap 2012

Raghunath Nambiar[1], Meikel Poess[2], Andrew Masland[3], H. Reza Taheri[4], Matthew Emmerton[5], Forrest Carman[6], and Michael Majdalany[7]

[1] Cisco Systems, Inc., 3800 Zanker Road, San Jose, CA 95134, USA
rnambiar@cisco.com
[2] Oracle Corporation, 500 Oracle Pkwy, Redwood Shores, CA 94065, USA
meikel.poess@oracle.com
[3] NEC Corporation of America, 14335 NE 24th Street, Bellevue, WA 98007, USA
andy.masland@necam.com
[4] VMware, Inc., 4301 Hillview Ave, Palo Alto CA 94304, USA
rtaheri@vmware.com
[5] IBM Canada, 8200 Warden Ave, Markham, ON L6G 1C7, Canada
memmerto@ca.ibm.com
[6] Owen Media, 3130 E. Madison St., Suite 206, Seattle, WA 98112, USA
forrestc@owenmedia.com
[7] LoBue & Majdalany Mgmt Group, 572B Ruger St. San Francisco, CA 94129, USA
majdalany@lm-mgmt.com

Abstract. The TPC has played, and continues to play, a crucial role in providing the computer industry with relevant standards for total system performance, price-performance and energy efficiency comparisons. Historically known for database-centric standards, the TPC is now developing standards for consolidation using virtualization technologies and multi-source data integration, and exploring new ideas such as Big Data and Big Data Analytics to keep pace with rapidly changing industry demands. This paper gives a high level overview of the current state of the TPC in terms of existing standards, standards under development and future outlook.

Keywords: Industry Standard Benchmarks, Transaction Processing Performance Council.

1 Introduction

In the 1980s, many companies practiced something known as "benchmarketing" – a practice in which organizations made performance claims based on internal benchmarks. The goal of running tailored benchmarks was simply to make one specific company's solution look far superior to that of the competition, with the objective of increasing sales. Companies created configurations specifically designed to maximize performance, called "benchmark specials," to force comparisons between non-comparable systems.

In response to this growing practice, a small group of individuals became determined to find a fair and neutral means to compare performance across database systems. Both

R. Nambiar and M. Poess (Eds.): TPCTC 2012, LNCS 7755, pp. 1–20, 2013.

influential academic database experts and well-known industry leaders contributed to this effort. Their important work eventually led to the creation of the TPC.

Founded in 1988, the Transaction Processing Performance Council (TPC) is a non-profit corporation dedicated to creating and maintaining benchmark standards, which measure database performance in a standardized, objective and verifiable manner. The TPC's goal is to create, manage and maintain a set of fair and comprehensive benchmarks that enable end-users and vendors to objectively evaluate system performance under well-defined, consistent and comparable workloads. As technology and end-customer solutions evolve, the TPC continuously reviews its benchmarks to ensure they reflect changing industry and marketplace requirements.

The TPC draws on its long history and experience to create meaningful benchmarks. The organization recently introduced the TPC-DS benchmark standard [4][6][7][12][17], which represents a modern, decision support workload. The TPC has also developed a TPC-Energy standard [1][15] designed to augment existing TPC benchmarks with energy metrics, so that end-users can understand the energy costs associated with a specific benchmark result. The organization is working on several new benchmarks for virtualized database environments.

Before the release of any new benchmark standard, the TPC creates a lengthy and detailed definition of the new benchmark. The resulting specifications are dense documents with stringent requirements; these very complete specifications help ensure that all published benchmark results are comparable. TPC members also constantly work to update and improve specifications to help them stay current and complete.

Unique to the TPC is the requirement that all published benchmarks be audited by an independent third party, which has been certified by the organization. This requirement ensures that published results adhere to all of the very specific benchmark requirements, and that results are accurate so any comparison across vendors or systems is, in fact, comparing "apples to apples."

The end result is that the TPC creates benchmark standards that reflect typical database workloads. The process of producing a benchmark is highly structured and audited so that valid assessments can be made across systems and vendors for any given benchmark. Reported results include performance, price/performance, and energy/performance, which help customers identify systems that deliver the highest level of performance, using the least amount of energy.

To date the TPC has approved a total of ten independent benchmark standards. Of these TPC-C [16], TPC-H [5], TPC-E [18][20] and TPC-DS are currently active standards. TPC-C and TPC-E are Online Transaction Processing (OLTP) benchmarks. Both benchmarks simulate a complete computing environment where a population of users executes transactions against a database. TPC-C is centered around the principal activities (transactions) of an order-entry environment, while TPC-E simulates the OLTP workload of a brokerage firm. TPC-H and TPC-DS are benchmarks that model several generally applicable aspects of a decision support system, including queries and continuous data maintenance. Both simulate the business model of a retail product supplier. TPC-DI, TPC-VMS and TPC-V are under development. The TPC-Pricing Specification and TPC-Energy Specification are common across all the benchmark standards. The timelines are shown in Figure 1.

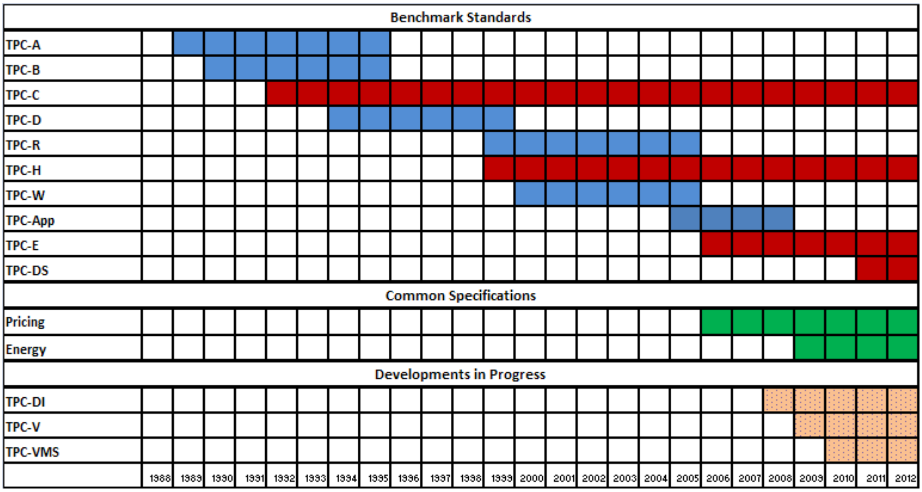

Fig. 1. TPC Benchmark Standards Timeline

The TPC continues to explore developments of new standards and enhancements to existing standards, and the TPC Technology Conference Series on Performance Evaluation and Benchmarking (TPCTC) initiative brings industry experts and researchers together to discuss novel ideas and methodologies in performance evaluation, measurement, and characterization [9][10][11].

The remainder of this paper is divided into five sections. The first section focuses on the benefits of TPC-E - the modern OLTP benchmark, the second section gives a historical perspective of benchmarks in the decision support space, the third section provides a high level overview of benchmark developments in the virtualization area, the fourth section summarizes the development of a new data integration benchmark, followed by a summary of the TPC Technology Conference Series on Performance Evaluation and Benchmarking.

2 OLTP Benchmarks

The TPC has developed and maintained a number of OLTP (Online Transaction Processing) benchmark standards over the course of its history. TPC-A (1989-1995) and TPC-B (1990-1995) were early attempts to create meaningful OLTP benchmarks but were quickly found lacking in various areas. TPC-C (1992-present) was the first comprehensive OLTP benchmark standard, which was designed around the order-entry model. The TPC-C benchmark is often referred to as TPC's flagship benchmark, with over 750 publications across a wide range of hardware and software platforms representing the evolution of transaction processing systems [16]. TPC-E (2007-present) was designed around a stock-trading model, and sought to be more representative of modern OLTP environments, and to address the high costs associated with constructing and operating large TPC-C benchmark environments [16][20].

2.1 A Comparison of TPC-C and TPC-E

The typical enterprise computing environment has changed considerably during the 15 years between the first releases of TPC-C and TPC-E. As a result there are some considerable differences in these two workloads. Tables 1 and Table 2 outline the business transactions found in each workload, along with some properties of each transaction, which will be expanded upon below.

Table 1. Business Transactions of TPC-C

Transaction	Mix	Access	ANSI Isolation	Type	Notes
New-Order	45%	Read-Write	Serializable	Core	Primary Metric
Payment	43%	Read-Write	Repeatable Read	Core	
Delivery	4%	Read-Write	Repeatable Read	Core	
Order-Status	4%	Read-Only	Repeatable Read	Lookup	
Stock-Level	4%	Read-Only	Read Committed	Lookup	

Table 2. Business Transactions of TPC-E

Transaction	Mix	Access	ANSI Isolation	Type	Notes
Trade-Order	10.1%	Read-Write	Repeatable Read	Core	
Market-Feed	1%	Read-Write	Repeatable Read	Core	Dependency on Trade-Order
Trade-Result	10%	Read-Write	Serializable	Core	Dependency on Trade-Order and Market-Feed Primary Metric
Broker-Volume	4.9%	Read-Only	Read Committed	Reporting	
Customer-Position	13%	Read-Only	Read Committed	Lookup	
Security-Detail	14%	Read-Only	Read Committed	Lookup	
Market-Watch	18%	Read-Only	Read Committed	Reporting	
Trade-Lookup	8%	Read-Only	Read Committed	Lookup	
Trade-Update	2%	Read-Write	Repeatable Read	Update	
Trade-Status	19%	Read-Only	Read Committed	Lookup	

The primary metric of both TPC-C and TPC-E is the transaction rate of a specific business transaction that is relevant to the workload. What is notable is that the primary metric of TPC-E (the Trade-Result transaction) has functional dependencies on other transactions (Trade-Order and Market-Feed), which makes attaining a higher performance score more challenging.

The workload composition by and large determines the complexity of the workload and the effort required to optimize the workload as a whole. In TPC-C, 92% of the workload is read-write and covers the core business transactions of an order-entry workload, with the remaining 8% covering supplemental lookup transactions. However, in TPC-E 21.1% of the workload is read-write and covers the core business transactions of a stock-trading workload. The remaining 78.9% is distributed among supplemental lookup (54%), reporting (22.9%) and update (2%) transactions. While the core business transactions still play an essential part in the workload, more attention must be given to the read-only operations on the system.

The data domains which a workload is designed around are a byproduct of the target audience of the workload. In TPC-C, the data is based around a single domain – the warehouse. In TPC-E, the data is arranged around two domains – customers and securities. By having data arranged around multiple domains, it becomes more difficult to partition data in order to take advantage of data locality.

The transaction isolation used when implementing the business transactions of the workload determine the amount of concurrency that the workload can exhibit. A workload with a larger number of heavily-isolated business transactions will exhibit less concurrency and lower performance. In TPC-C, 45% of the business transactions are at the highest isolation level, and 96% of the transactions are at the two highest isolation levels. In TPC-E, 10% of the business transactions are at the highest isolation level, and 23.1% of the transactions are at the two highest isolation levels. In both cases, the business transactions at the highest isolation level are also the primary metrics of the workloads. In addition, TPC-E has a much smaller proportion of business transactions at the highest isolation levels, which allows for greater concurrency.

The durability of a database and the redundancy of the storage subsystem supporting the database play an important role in the performance of the environment and relevance of the environment to a typical customer. In TPC-C, there are no specific durability or redundancy requirements, aside from the requirement that the database must be durable. Typically, this is validated by restoring from a backup and doing a roll-forward recovery through the transaction logs. In TPC-E, there are specific durability requirements (Business Recovery) and redundancy requirements (Data Accessibility). In the case of durability, the task of recovering from a system failure is measured and reported; it behooves test sponsors to implement hardware and software solutions that provide quick recovery while ensuring durability. In the case of redundancy, it is required that all storage devices are protected by some level of redundancy. This ensures that critical data is always protected and accessible, even in the face of component failures.

All of these differences stem from a desire to reflect changes in real-world OLTP workloads, which are based around complex and highly-integrated business processes. While it is still true that these workloads are built around a set of core OLTP business transactions, the successful implementation of the entire workload relies on a variety of factors, including the interaction of multiple business transactions with each other, co-existence with many other light-weight lookup and reporting transactions, the need to operate on different domains of data, increased concurrency due to reduced isolation levels, and the requirement for redundant storage to minimize the impact of outages.

2.2 The Relevance of TPC-C and TPC-E Today

With the advent of any new standard, it is always tempting to deprecate and discontinue older standards. However, there are benefits in keeping the standards active as they appeal to different audiences. TPC-C has a long history. It is a well-understood workload, is simple to implement and execute, and the order-entry model is easy to conceptualize. This makes it a great choice for simple performance measurement and analysis purposes, by both test sponsors and academia.

TPC-E, on the other hand, is relatively new. It is a complex workload, which is more difficult to implement. The larger number of transactions, their explicit and implicit dependencies, the use of multiple data domains and increased concurrency rates make it more challenging to understand how everything operates – but customers face these problems every day with their production OLTP workloads. This makes TPC-E a very useful engineering tool, as test sponsors can better understand how their hardware and software behaves in such an environment, and use that knowledge to improve the adaptability, scalability and performance of their products to better serve their customers.

2.3 Why a Stock Trading Workload?

When most people think of OLTP, they think of retail or financial applications, as these are quite central to our everyday lives. This is part of the reason why the TPC-C order-entry model has become entrenched as the standard for OLTP benchmarking.

When the TPC released TPC-E designed around a stock-trading model, there was some initial confusion. While stock-trading does combine many aspects of retail and financial OLTP workloads, it is very much an outlier. While the TPC has always maintained that the TPC-E workload is quite relevant to today's OLTP environments, a demonstration will make this much clearer.

Table 3. Transformation of TPC-E into an Order-Entry Workload

	TPC-E (Stock-Trading)	**Transformed (Order-Entry)**
Business Model Description	A brokerage firm, where the brokerage accepts trades from customers which are then fulfilled by the market.	A web-based retailer, where the retailer accepts orders from customers which are then fulfilled directly by suppliers.
Business Model Components	Broker Market	Retailer Supplier
Schema	Trade Security Exchange	Order Item DistributionCenter
Transactions	Trade-Order Market-Feed Trade-Result Broker-Volume Customer-Position Market-Watch Security-Detail Trade-Lookup Trade-Update Trade-Status	Order-Entry Supplier-Feed Order-Completion Retailer-Volume Customer-Status Supplier-Watch Item-Detail Order-Lookup Order-Update Order-Status

For ease of comprehension, it is not very hard to transform TPC-E from a stock-trading model into an order-entry model. While this is not a perfect transformation, it is sufficient to understand how the complex TPC-E stock-trading workload adequately represents a complex order-entry workload that is relevant to today's real-world applications. Under this transformation, the new workload has:

- multiple data domains – customers and items
- core OLTP queries
- customers who purchase items from a retailer (Order-Entry)
- retailers who pass on those orders to a supplier in batches (Supplier-Feed)
- suppliers who fulfill the orders (Order-Completion)
- reporting queries (Retailer-Volume, Supplier-Watch)
- lookup queries (Order-Lookup, Order-Status, Customer-Position, Item-Detail)
- update queries (Order-Update)

This transformation demonstrates that the core OLTP transactions are present, the reporting and lookup transactions serve meaningful purposes and the data domains are relevant to the order-entry model. Hence, the choice of the stock-trading model is quite valid for an OLTP workload, as it mirrors the complexity of today's order-entry workloads.

In summary, the TPC-E workload presents a very relevant, complex, OLTP workload that brings the challenges of customer environments into the engineering departments of test sponsors. TPC-E will continue to push the boundaries of effective optimization and performance tunings, ultimately for the benefit of customers. However, this does not diminish the position of TPC-C in this area, which is still a useful workload for academic and engineering analysis.

3 Decision Support Benchmarks

For the last decade, the research community and the industry have used TPC-D and its successor TPC-H to evaluate the performance of decision support (DSS) technology. Recognizing the paradigm shifts that happened in the industry over the last fifteen years, the TPC has developed a new decision support benchmark, TPC-DS, which was released in February 2012. The ideas and tools of TPC-DS stem from an early papers in SIGMOD [4], VLDB [6] and WOSP [2]. From an ease of benchmarking perspective it is similar to TPC-D and TPC-H. However, it adjusts for new technology and new approaches the industry has embarked upon over the fifteen years.

3.1 History of Decision Support Benchmarks

The roots of TPC-H date back to April 1994 when the TPC's first decision support benchmark, TPC-D, was released. For the technology available at that time, TPC-D imposed many challenges on both hardware and DBMS systems. Although the development of aggregate/summary structures, originally spurred by TPC-D (e.g. join indices, summary tables, materialized views, etc.) benefitted the industry, they effectively broke the benchmark because the decrease of query elapsed times resulted in an over proportional increase in the main performance metric. As a consequence the TPC spun off two modified versions of TPC-D: TPC-H and TPC-R. The main difference between TPC-H and TPC-R was that TPC-R allowed the use of aggregate/summary structures, where TPC-H prohibited their use. As a result, TPC-H posed a more challenging workload that was more customer-relevant and garnered the support of the industry.

3.2 Overview of the TPC-H Workload

TPC-H implements an ad-hoc decision support benchmark. The ad-hoc nature of the benchmark is intended to simulate a real-life scenario where database administrators (DBAs) do not know which queries will be executed against the database system; hence, knowledge about its queries and data may not be used to optimize the DBMS system. It uses a 3rd Normal Form (3NF) schema consisting of eight tables, which can be populated with up to 100 terabytes (TB) of raw data with mostly uniform distributions. It contains 22 complex and long running queries combined with 2 data maintenance functions (insert and delete). Six of the eight tables grow linearly with the scale factor.

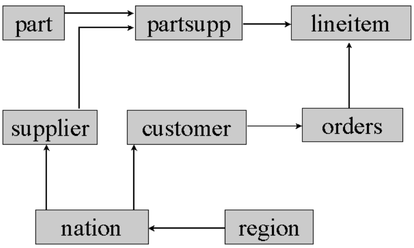

Fig. 2. TPC-H ER-Diagram

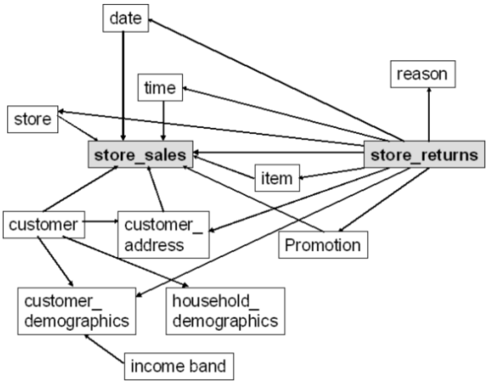

Fig. 3. TPC-DS ER-Diagram

The differences between today's decision support systems and the TPC-H benchmark specification are manifold. The TPC-H schema, although sufficiently complex to test the early systems, is not representative of all of today's more complex DSS implementations, where schemas are typically composed of a larger number of tables and columns. Furthermore, the industry's choice of schema implementation appears to have shifted from pure 3NF schemas to variations of the star schema, such as snowflake schemas.

The purity of TPC-H's 3NF schema and the low number of tables and columns may not fully reveal the differences in indexing techniques and query optimizers. Because the main tables scale linearly with the database size (scale factor), the cardinalities of some tables reach unrealistic proportions at large scale factors.

For instance, at scale factor 100,000 the database models a retailer selling 20 billion distinct parts to 15 billion customers at a transaction rate of 150 billion per year.

The database population, consisting of mostly un-skewed and synthetic data, imposes little challenges on statistic collection and optimal plan generation by the query optimizer.

The TPC-H data maintenance functions (rf1, rf2) merely constrain a potential excessive use of indices rather than testing the DBMS' capability of performing realistic data maintenance operations, common during Extraction Transformation Load (ETL) processes, also known as Data Integration (DI) processes. The data maintenance functions insert and delete orders randomly rather than ordered by time. The inserted data is assumed to be clean so that no data transformations are necessary. Data is loaded and deleted from 2 out of 8 tables.

There are relatively few distinct queries in TPC-H, and because they are known before benchmark execution, engineers can tune optimizers and execution paths to artificially increase performance of the system under test. Also, actual data warehouses are not subject to the TPC-H benchmark constraints and will define indices on non-date and non-key columns as well as contain summary tables.

3.3 Overview of the TPC-DS Workload

While TPC-DS [17] may be applied to any industry that must transform operational and external data into business intelligence, the workload has been granted a realistic context. It models the decision support tasks of a typical retail product supplier. 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. TPC-DS takes the marvels of TPC-H and TPC-R and fuses them into a modern DSS benchmark. Its main focus areas include

i) Realistic benchmark context;

ii) Multiple snowflake schemas (also known as a Snowstorm schema) with shared dimensions;

iii) 24 tables with an average of 18 columns;

iv) Realistic table content and scaling

v) Representative skewed database content;

vi) Realistic workload;

vii) 99 distinct SQL 99 queries with random substitutions;

viii) Ad-hoc, reporting, iterative and extraction queries;

viiii) Continuous ETL (data integration process) and

x) Easy to understand, yet meaningful and un-breakable metric.

3.4 Benchmark Schema and Data Population

The schema, an aggregate of multiple star schemas, contains essential business information such as detailed customer, order, and product data for the classic sales channels: store, catalog and Internet. Wherever possible, real world data are used to populate each table with common data skews such as seasonal sales and frequent names. In order to realistically scale the benchmark from small to large datasets, fact tables scale linearly while dimensions scale sub-linearly [7][12].

The design of the data set is motivated by the need to challenge the statistic gathering algorithms used for deciding the optimal query plan and the data placement algorithms, such as clustering and partitioning. TPC-DS uses a hybrid approach of data domain and data scaling. While pure synthetic data generators have great advantages, TPC-DS uses a mixture of both synthetic and real world based data domains. Synthetic data sets are well understood, easy to define and implement. However, following the TPC's paradigm to create benchmarks that businesses can relate to, a hybrid approach to data set design scores many advantages over both pure synthetic and pure real world data. This approach allows both realistically skewed data distributions yet still a predictable workload.

Compared to previous TPC decision support benchmarks, TPC-DS uses much wider tables (up to 39 columns), with domains ranging from integer, float (with various precisions), char, varchar (of various lengths) and date. Combined with a large number of tables (total of 25 tables and 429 columns) the schema gives both the opportunity to develop realistic and challenging queries as well as the opportunity for innovative data placement algorithms and other schema optimizations, such as complex auxiliary data structures. The number of times columns are referenced in the dataset varies between 0 and 189.

Of those columns accessed, the largest numbers of columns are referenced between 5 and 49 times. The large column set and diverse query set of TPC-DS also protects its metric from unrealistic tuning and artificial inflations of the metric, a problem which rapidly destroyed the usefulness of TPC-D in the late 1990s. That, combined with the complex data maintenance functions and load time participating in the primary performance metric, creates the need for fast and efficient algorithms to create and maintain auxiliary data structures and the invention of new algorithms.

The introduction of NULL values into any column except the primary keys opens yet another dimension of challenges for the query optimizer compared to prior TPC decision support benchmarks. The percent of NULL values in each non-primary key column varies from 4 to 100 percent based on the column. Most columns have 4 percent NULL values. The important rec_end_date columns have 50 percent NULL values. Some columns were unused (total of 236) or intentionally left entirely NULL for future enhancements of the query set.

3.5 Benchmark Workload

The benchmark abstracts the diversity of operations found in an information analysis application, while retaining essential performance characteristics. As it is necessary to execute a great number of queries and data transformations to completely manage any business analysis environment, TPC-DS defines 99 distinct SQL-99 queries –with Online Analytical Processing (OLAP) amendment –and 12 data maintenance operations covering typical DSS-like query types such as ad-hoc, reporting, iterative (drill down/up) and extraction queries and periodic refresh of the database.

Due to strict implementation rules it is possible to amalgamate ad-hoc and reporting queries into the same benchmark; it is possible to use sophisticated auxiliary data structures for reporting queries while prohibiting them for ad-hoc queries. Although the emphasis is on information analysis, the benchmark recognizes the need to periodically refresh the database (ETL). The database is not a one-time snapshot of a business operations database, nor is it a database where OLTP applications are running concurrently. The database must be able to support queries and data maintenance operations against all tables. Some TPC benchmarks (e.g., TPC-C and TPC-App) model the operational aspect of the business environment where transactions are executed on a real time basis; other benchmarks (e.g. TPC-H) address the simpler, more static model of decision support. The TPC-DS benchmark, however, models the challenges of business intelligence systems where operational data is used both to support sound business decisions in near real-time and to direct long-range planning and exploration. The TPC-DS operations address complex business problems using a variety of access patterns, query phrasings, operators and answer set constraints.

3.6 Metric

The TPC-DS workload consists of three distinct disciplines: Database Load, Power Run and Throughput Run. The power run executes 99 templates using random bind variables. Each throughput run executes multiple sessions each executing the same 99 query templates with different bind variables in permutated order, thereby simulating a workload of multiple concurrent users accessing the system.

- SF is the scale factor used for a benchmark
- S is the number of concurrent streams, i.e. the number of concurrent users
- Q is the total number of weighted queries: Q=3* S*99, with S being the number of streams executed in a throughput run
- TPT=TPower*S, where TPT is the total elapsed time to complete the Power Test
- TTT= TTT1+TTT2, where TTT1 is the total elapsed time of Throughput Test 1 and TTT2 is the total elapsed time of Throughput Test 2
- TLD is the load factor TLD=0.01*S*TLoad, and TLoad is the actual load time
- TPT, TTT and TLD quantities are in units of decimal hours with a resolution of at least 1/3600th of an hour (i.e., 1 second)

The Performance Metric reflects the effective query throughput per second. The numerator represents the total number of queries executed on the system "198 * S", where 198 is the 99 individual queries times two query runs and S is the number of concurrent simulated users. The denominator represents the total elapsed time as the sum of Query Run1, Data Maintenance Run, Query Run 2 and a fraction of the Load Time. Note that the elapsed time of the data maintenance run is the aggregate of S executions of all data maintenance functions. By dividing the total number of queries by the total elapsed time, this metric represents queries executed per time period. Using an arithmetic mean to compute the primary benchmark metric should cause DBMS developers to concentrate primarily on long-running queries first, and then progressively continue with shorter queries. This generally matches the normal business case, where customers spend most of their tuning resources on the slowest queries. For a complete specification, please refer to [17].

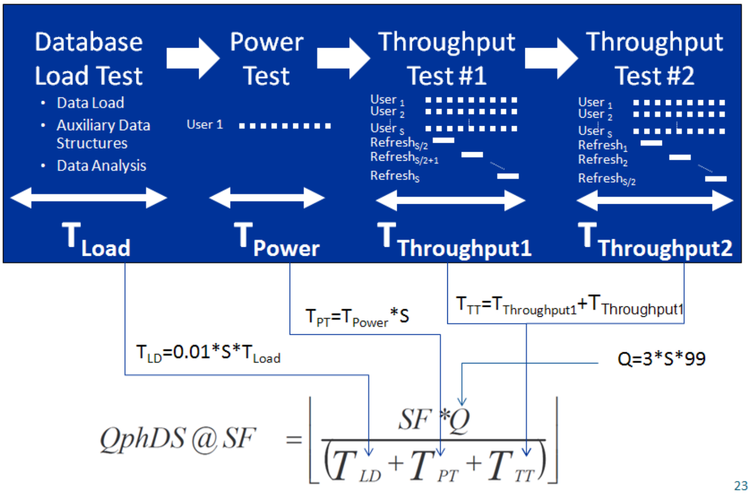

Fig. 4. TPC-DS Execution Order

4 Virtualization Benchmarks

Virtualization on x86 systems started out as a means of allowing multiple Linux and Windows operating systems to execute simultaneously on a single PC, but it has since become a foundation of enterprise data centers. It enables:

- Consolidation of multiple operating environments onto one server
- Migration of a VM to a new physical server while the applications on the VM continue to be in use, freeing the original server for maintenance operations
- Live migration of VMs between hosts allows for a rich set of load balancing and resource management features. Virtualization is the fundamental enabling technology behind cloud computing.
- High-Availability after a server failure by allowing its VMs to restart on a new server
- Fault-tolerance on generic servers without hardware fault-tolerance features

Databases are the last frontier to be conquered by virtualization. Only recently have virtualized servers been able to offer the level of throughput, predictable performance, scaling, storage and networking load, and reliability that databases demand. This in turn has led to customer demands for better metrics to compare virtualization technologies under database workloads. TPCTC 2009 and TPCTC 2010 papers outlined this need, and presented proposals for developing a benchmark for virtualized databases [14][8].

4.1 The Evolution of Two Virtualization Benchmark Endeavors

In response to this demand, the TPC formed a Development Subcommittee[1] in 2010 with the goal of developing a new benchmark called TPC-V. During the development phase of TPC-V, it became obvious that a second, simpler benchmark would be useful because:

- TPC-V is a complex benchmark which will take a few years to develop.
- The development of a new workload necessitates system and database vendors to develop new benchmark kits, for use during prototyping.
- It represents a complex, cloud-inspired workload but there is also demand for a simpler configuration of small numbers of databases virtualized on one server.

These reasons led to the formation of a second Development Subcommittee to develop a TPC-VMS (TPC Virtual Measurement Single System) benchmark. The TPC-VMS Specification leverages the existing TPC benchmarks, TPC-C, TPC-E, TPC-H and TPC-DS, by adding the methodology and requirements for running and reporting virtualization metrics. A major driving force behind TPC-VMS was defining the specification in such a way as to make it possible for benchmark sponsors to run an existing TPC benchmark on a virtual server without the need for modification to the existing benchmarking kit for that benchmark. Hence, it is expected that the TPC-VMS specification will be completed quickly, and will lead to a large number of publications using existing benchmarking kits.

TPC-VMS and TPC-V fulfill the demands of two different market segments. TPC-VMS emulates a simple consolidation scenario of three identical databases running the same workload on the same OS, DBMS, etc. TPC-V emulates a complex cloud computing environment with varying numbers of VMs, two different workloads (with OLTP and DSS properties), dynamic increases and decreases of the load presented to each VM, etc.

4.2 TPC-VMS Benchmark

The goals for TPC-VMS are to measure TPC benchmarks in a virtualized environment as follows:

[1] A TPC Development Subcommittee is the working forum within the TPC for the development of a Specification.

- Provide the virtualization measurements that a typical customer of the particular systems benchmarked would consume.
- Provide virtualization metrics that are comparable between systems under test for a particular TPC Benchmark Standard.
- Provide for repeatable and documented measurements.
- Leverage existing TPC Benchmark Standards without requiring any implementation changes.

The TPC Benchmark Standard Database Servers are consolidated onto the Consolidated Database Server as depicted by Figure 5. As shown the Database Server's Operating Systems and DBMSs are consolidated onto the Consolidated Database Server each in a separate Virtual Machine.

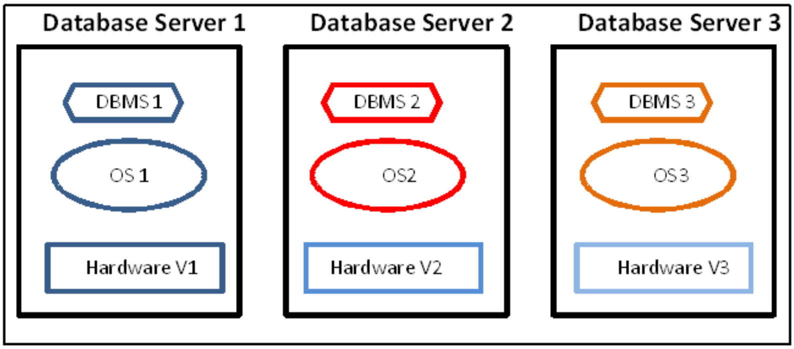

Fig. 5. TPC-VMS Consolidation Configuration

Aggregating the results of three different benchmark runs into a single metric can be daunting, given the potential for gaming the results by boosting the performance of one VM at the expense of another. TPC-VMS avoids the aggregation problem by defining the metric as the lowest metric reported by any of the three VMs.

Early prototyping results with the TPC-E and TPC-H benchmarks have shown that TPC-VMS will meet its goals with a quick development schedule. The benchmark specification is nearing completion, and TPC-VMS is expected to be approved in late 2012.

4.3 TPC-VMC Benchmark Proposal

The TPC has formed a Working Group to investigate a TPC-VMC (TPC Virtual Measurement Complex Systems) benchmark. The idea is to extend TPC-VMS into a more complex benchmark. The Working Group is considering the following functions:

- Elasticity: the load presented to the VMs will vary with time
- Live migrations: Due to increase in load, VMs will migrate from one server to a second, idle server

- Deployment: this important property of cloud computing datacenters will be emulated by the creation of a VM from a template, and subsequent deployment and booting up of the VM before it starts running transactions

TPC-VMC is still in the definition stage. Since it can be run using existing TPC benchmarking kits, it is a good alternative to TPC-V should TPC-V development be delayed due to its dependence on a new benchmarking kit.

4.4 TPC-V Benchmark

The TPC-V Development Subcommittee chose to base TPC-V upon the existing TPC-E [18] benchmark to speed the development process. Using the TPC-E transactions as a base, the working group has defined 3 VMs that together form a ***Set*** for the TPC-V benchmark. The functionality of the Tier B component of the TPC-E System Under Test (SUT) has been divided into two separate VMs. One VM handles the Trade-Lookup and Trade-Update transactions, simulating the high storage I/O load of a decision support environment. The second VM services all other transactions, which have a CPU-heavy profile and represent an OLTP environment.

Tier A in TPC-V functions similarly to a TPC-E Tier A with one major difference: based on the transaction type, it routes the transaction to one of the two Tier B VMs. In Figure 6, notations TL, TU, etc. under the VMs are the 2-letter abbreviations of TPC E transactions.

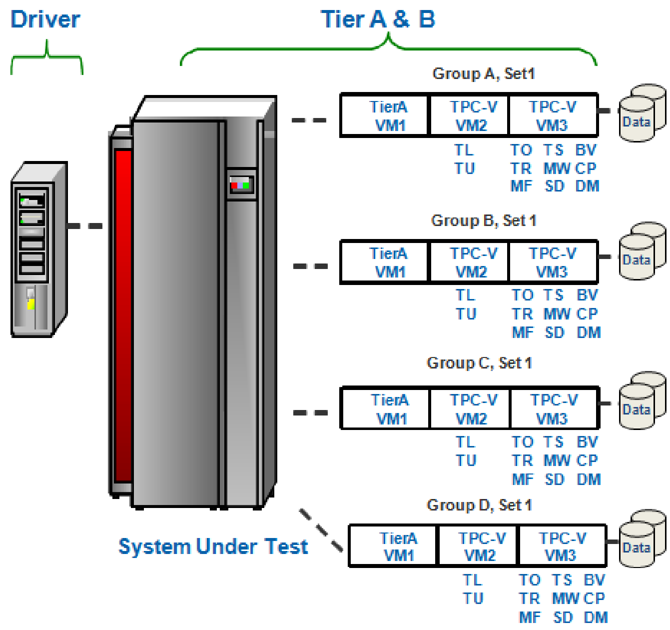

Fig. 6. Sample Components of Test Configuration

The Subcommittee has devised a Set architecture whereby both the number of Sets, and the load placed on each Set grow as the performance of the system increases. The advantage here is that the benchmark will emulate the behavior of real-world servers: more powerful servers host more VMs, but also VMs that handle more load.

Another major feature of TPC-V is varying the load to the many VMs during the measurement interval to emulate the elasticity that is ubiquitous in cloud computing environments. The overall load will remain constant, but the portion directed to each Set will vary. An expected side effect is configuring the VMs with oversubscribed resources that are typical of virtualized servers. For example, a server with 64 physical CPUs might need to run with 24 VMs whose virtual CPUs total 150 in order to handle the elastic nature of the load. The chart below depicts how the load to four Sets might vary over the 2-hour Measurement Interval.

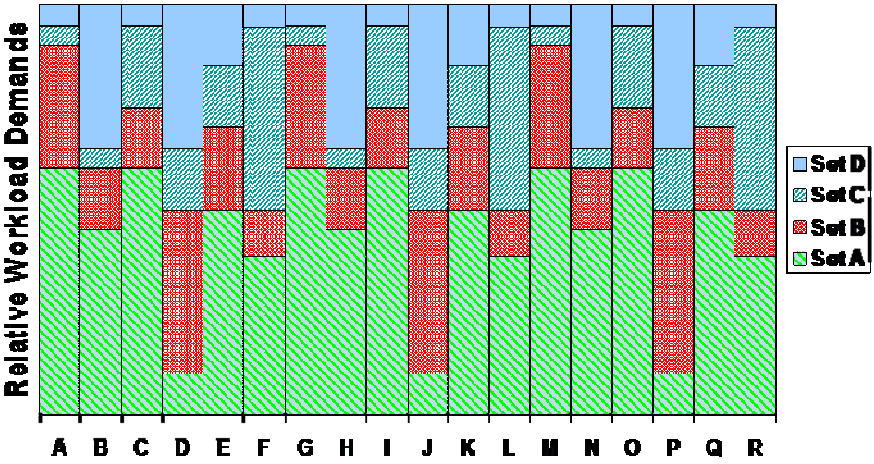

Fig. 7. TPC V elastic load variation

4.5 A Reference Kit for TPC-V Benchmark

The Development Subcommittee has taken on the task of developing a publicly-available, end-to-end reference kit for the benchmark. The reference kit is being developed in Java and C++ and will use the PostgreSQL open source database. This is the first TPC benchmark that will be available with an end-to-end kit, making it possible for anyone to download the kit and immediately run a very complex benchmark using an open source database. Furthermore, the availability of the kit with its driver, which will take care of apportioning the load among the VMs as well as varying the load from period to period, will mean the test sponsors are relieved from dealing with all the complex features of the benchmark. If a sponsor chooses to run the benchmark against a commercial DBMS, they can replace PostgreSQL with a commercial DBMS and publish results.

4.6 Status of TPC-V Benchmark

The TPC-V Development Subcommittee has focused much of its effort on developing the end-to-end reference kit, as it will be highly useful when prototyping the workload and evaluating changes. The kit is now mostly functional, as it is able to drive both the TPC-E and TPC-V workloads, and runs all but 2 of the transactions. Early prototyping results indicate that while PostgreSQL performance might not match that of a highly-tuned commercial database, it will be more than sufficient to evaluate a heavily-virtualized database benchmark.

5 Data Integration Benchmark

The TPC-DI benchmark originated from the TPC-ETL initiative, outlined in [3]. It is designed to be a performance test for systems that move and integrate data from various data sources, so called Data Integration (DI) systems (a.k.a. Extract, Transform and Load, or ETL systems). As these systems perform an intricate part in building data warehouse systems, they have been around for quite some time and are available from a number of vendors. However, until now there has been no standard to compare them in a fair and accurate way.

The benchmark workload transforms data extracted from an On-Line Transaction Processing (OTLP) system and loads it along with data from ancillary data sources (including tabular and hierarchical structures) into a data warehouse. The source and destination schemas, data transformations and implementation rules have been designed to be broadly representative of modern data integration requirements. No single benchmark can reflect the entire range of possible DI requirements. However, using data and operation models of a retail brokerage the TPC-DI benchmark exercises a breadth of system components associated with DI environments, which are characterized by:

- The manipulation and loading of large volumes of data
- A mixture of transformation types including error checking, surrogate key lookups, data type conversions, aggregation operations, data updates, etc.
- Historical loading and incremental updates of a destination Data Warehouse using the transformed data
- Consistency requirements ensuring that the integration process results in reliable and accurate data
- Multiple data sources having different formats
- Multiple data tables with varied data types, attributes and inter-table relationships

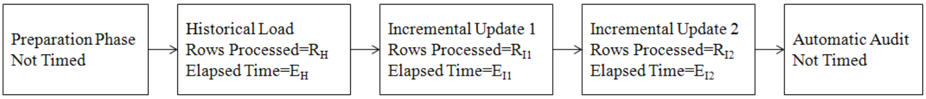

Fig. 8. TPC-DI Benchmark Phases

The Performance Metric reported by TPC-DI is a throughput measure, the number of source rows processed per second. Conceptually, it is calculated by dividing the total rows processed by the elapsed time of the run. Each benchmark run consists of the following phases, which are performed in the following sequence:

The primary performance metric is defined as: GeoMean(T_H, min(T_{I1} , T_{I2})) with:

- T_H being the historical load performance: $T_H = \frac{R_G}{E_H}$
- T_{Ii} $i \in \{1,2\}$ being the incremental load performance: $T_H = \frac{R_{Ii}}{\max(E_{Ii}, 1800)}$

TPC-DI is still under development and, therefore, the specification may change until its planned release in 2013.

6 TPC Technology Conference Initiative

Over the past quarter-century, the Transaction Processing Performance Council (TPC) has developed several industry standards for performance benchmarking, which have been a significant driving force behind the development of faster, less expensive, and more energy efficient systems.

To keep pace with the rapidly changing information technology landscape, four years ago the TPC initiated the international conference series on Performance Evaluation and Benchmarking (TPCTC). The objective of this conference series is to bring industry experts and research community together in developing new standards and enhancing existing standards in performance evaluation and benchmarking.

The first TPC Technology Conference on Performance Evaluation and Benchmarking (TPCTC 2009) was held in conjunction with the 35th International Conference on Very Large Data Bases (VLDB 2009) in Lyon, France from August 24th to August 28th, 2009 [9]. The second conference (TPCTC 2010) was held in conjunction with the 36th International Conference on Very Large Data Bases (VLDB 2010) in Singapore from September 13th to September 17th, 2010 [10], while the third (TPCTC 2011) was held in conjunction with the 37th International Conference on Very Large Data Bases (VLDB 2011) in Seattle from August 29th to September 3rd, 2011 [11]. This conference series has been a tremendous success. The initiation of the development of benchmarks in virtualization and data integration has been a direct result.

The areas of focus of the fourth TPC Technology Conference on Performance Evaluation and Benchmarking (TPCTC 2012) include:

- Big Data analytics and infrastructure
- Database Appliances
- Cloud Computing
- In-memory databases
- Social media infrastructure
- Business intelligence
- Complex event processing
- Database optimizations
- Green computing
- Disaster tolerance and recovery
- Energy and space efficiency
- Hardware innovations
- Data Integration
- Hybrid workloads
- Virtualization

Acknowledgements. The authors thank the past and present members of the TPC for their contribution to the specifications and documents referenced in this paper.

TPC Benchmark^{TM} and TPC-C^{TM} are trademarks of the Transaction Processing Performance Council.

References

1. Young, E., Cao, P., Nikolaiev, M.: First TPC-Energy Benchmark: Lessons Learned in Practice. In: Nambiar, R., Poess, M. (eds.) TPCTC 2010. LNCS, vol. 6417, pp. 136–152. Springer, Heidelberg (2011)
2. Stephens, J.M., Poess, M.: MUDD: a multi-dimensional data generator. In: WOSP 2004, pp. 104–109 (2004)
3. Wyatt, L., Caufield, B., Pol, D.: Principles for an ETL Benchmark. In: Nambiar, R., Poess, M. (eds.) TPCTC 2009. LNCS, vol. 5895, pp. 183–198. Springer, Heidelberg (2009)
4. Poess, M., Smith, B., Kollár, L., Larson, P.-Å.: TPC-DS, taking decision support benchmarking to the next level. In: SIGMOD Conference 2002, pp. 582–587 (2002)
5. Poess, M., Floyd, C.: New TPC Benchmarks for Decision Support and We-Commerce. SIGMOD 2000 Record 29(4), 64–71 (2000)
6. Poess, M., Stephens, J.M.: Generating Thousand Benchmark Queries in Seconds. In: VLDB 2004, pp. 1045–1053 (2004)
7. Poess, M., Nambiar, R., Walrath, D.: Why You Should Run TPC-DS: A Workload Analysis. In: VLDB 2007, pp. 1138–1149 (2007)
8. Sethuraman, P., Taheri, H.R.: TPC-V: A Benchmark for Evaluating the Performance of Database Applications in Virtual Environments. In: Nambiar, R., Poess, M. (eds.) TPCTC 2010. LNCS, vol. 6417, pp. 121–135. Springer, Heidelberg (2011)
9. Nambiar, R., Poess, M. (eds.): Topics in Performance Evaluation, Measurement and Characterization. Springer (2012) ISBN 978-3-642-32626-4
10. Nambiar, R., Poess, M. (eds.): Performance Evaluation, Measurement and Characterization of Complex Systems. Springer (2011) ISBN 978-3-642-18205-1
11. Nambiar, R., Poess, M. (eds.): Performance Evaluation and Benchmarking. Springer (2009) ISBN 978-3-642-10423-7
12. Nambiar, R., Poess, M.: The Making of TPC-DS. In: VLDB 2006, pp. 1049–1058 (2006)
13. Nambiar, R., Poess, M.: Transaction Performance vs. Moore's Law: A Trend Analysis. In: Nambiar, R., Poess, M. (eds.) TPCTC 2010. LNCS, vol. 6417, pp. 110–120. Springer, Heidelberg (2011)

14. Bose, S., Mishra, P., Sethuraman, P., Taheri, R.: Benchmarking Database Performance in a Virtual Environment. In: Nambiar, R., Poess, M. (eds.) TPCTC 2009. LNCS, vol. 5895, pp. 167–182. Springer, Heidelberg (2009)
15. TPC Energy Specification, http://www.tpc.org/tpc_energy/spec/TPC-Energy_Specification_1.2.0.pdf
16. TPC: TPC Benchmark C Specification, http://www.tpc.org/tpcc/spec/tpcc_current.pdf
17. TPC: TPC Benchmark DS Specification, http://www.tpc.org/tpcds/spec/tpcds_1.1.0.pdf
18. TPC: TPC-Pricing Specification, http://www.tpc.org/tpce/spec/v1.12.0/TPCE-v1.12.0.pdf
19. TPC: TPC Benchmark H Specification, http://www.tpc.org/tpch/spec/tpch2.14.4.pdf
20. Hogan, T.: Overview of TPC Benchmark E: The Next Generation of OLTP Benchmarks. In: Nambiar, R., Poess, M. (eds.) TPCTC 2009. LNCS, vol. 5895, pp. 84–98. Springer, Heidelberg (2009)