5-2013

A Comparison of Leading Database Storage Engines in Support of Online Analytical Processing in an Open Source Environment

Gabriel Tocci
East Tennessee State University

A Comparison of Leading Database Storage Engines in Support of Online Analytical Processing in an Open Source Environment

---

A thesis

presented to

the faculty of the Department of Computer and Information Science

East Tennessee State University

In partial fulfillment

of the requirements for the degree

Master of Science in Computer and Information Science

---

by

Gabriel Tocci

May 2013

Dr. Ronald Zucker, Chair

Dr. Don Bailes

Dr. Tony Pittarese

Keywords: Online Analytical Processing, Open Source, MyISAM, InnoDB

ABSTRACT

A Comparison of Leading Database Storage Engines in Support of

Online Analytical Processing in an Open Source Environment

by

Gabriel Tocci

Online Analytical Processing (OLAP) has become the de facto data analysis technology used in modern decision support systems. It has experienced tremendous growth, and is among the top priorities for enterprises. Open source systems have become an effective alternative to proprietary systems in terms of cost and function. The purpose of the study was to investigate the performance of two leading database storage engines in an open source OLAP environment.

Despite recent upgrades in performance features for the InnoDB database engine, the MyISAM database engine is shown to outperform the InnoDB database engine under a standard benchmark. This result was demonstrated in tests that included concurrent user sessions as well as asynchronous user sessions using data sets ranging from 6GB to 12GB. Although MyISAM outperformed InnoDB in all test performed, InnoDB provides ACID compliant transaction technologies are beneficial in a hybrid OLAP/OLTP system.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Online Analytical Processing (OLAP) systems enable business leaders and executives to analyze large amounts of data quickly and interactively. This ability provides insight into the business in a manner understandable to the user to support decision making. This work analyzed the performance of leading database storage engines in support of OLAP in an open source environment under a standard benchmark defined by the Transaction Processing Council [1].

OLAP has become the de facto data analysis technology used in modern decision support systems, commonly referred to as Business Intelligence (BI) or Executive Information Systems (EIS). Decision support systems have experienced tremendous growth and are among the top priorities for enterprises [2]. All principal DBMS (e.g. Oracle, Microsoft, and IBM) vendors now have offerings in data warehousing and OLAP technologies [3].

BI Systems were first proposed in 1958 by H.P. Luhn [4]. Although the specific technologies proposed to be used by Luhn are no longer relevant, many of the theoretical characteristics proposed in 1958 remain [4]. Effective decision support systems leverage a wealth of business data from numerous touch points and translate it into tangible and lucrative results [5]. BI is an information technology (IT) system, "that allows organizations to access, analyze, and share information across the organization … BI provides employees with information to make better decisions, and can be used in environments ranging from workgroups of 20 users to enterprise deployments exceeding 20,000." [6]. BI gives executives and key decision makers the ability to see business processes with a high level of transparency [6].

OLAP was developed to solve the significant technical challenges in efficiently analyzing large amounts of data [7]. This solution is facilitated through storage, retrieval, and analysis of

enterprise data in its natural, multidimensional perspective. This multidimensional data model is the structural difference between OLAP and its relational counterpart, Online Transactional Processing (OLTP) systems, that enables efficient analytical processing.

Open source software is developed under a license that is based on the idea of a free exchange of technical information. Sharing of technical specifications for software (source code) has proven to be an excellent development model because it grants every user the freedom to change the source code [8]. Although not a universal agreement, recent studies have been released that provide quantitative data proving open source software is a reasonable, and in some cases a superior, software solution [9, 10]. Open source BI is out of the innovation stage and has progressed from almost nothing a decade ago, into the mainstream with community and commercially supported projects [11].

This work investigates the performance of leading database storage engines developed using the open source model in an OLAP environment. The following steps are conducted:

- A brief analysis to confirm the storage engines are in compliance with open source requirements.
- A technical comparison of the storage engines, and identification of fundamental components that support OLAP.
- A performance benchmark comparison of the storage engines based on the Decision Support Benchmark (TPC-DS), as defined by the Transaction Processing Council.
- Two additional TPC-DS benchmark tests, executed on scaled data sets to determine if execution times scale in proportion to data set size.

Previous research exists showing strong development and maturity in open source BI user tools, but not the underlying engines [12]. Recent benchmarks have been performed that explore the leading open source database engines, and one, InnoDB, has undergone extensive updates in its most recent version [13, 14, 15]. Recent data on the performance of the leading open source database engines in support of OLAP has not been published with respect to a standard benchmark or the recent updates to InnoDB. The work presented here seeks to address this gap in the literature and determine the performance of the leading open source database storage engines in support of OLAP. This work is timely due to the strained global economic situation, the emergence of the TPC-DS as an industry standard OLAP benchmark, and the recent release of updates to the InnoDB storage engine [14, 16].

During a strained economic situation it is generally acceptable to simply maintain market share and profitability. With BI, organizations can actually improve market share and profitability [6]. Open source BI has become an effective alternative to proprietary systems [11]. Although the top reason for open source adoption is cost savings, reduced vendor dependence and ease of integration follow closely behind [11]. BI leverages existing Information Technology (IT) infrastructure, such as Enterprise Resource Planning (ERP) and Customer Relationship Management (CRM), by analyzing data previously stored in these existing systems without significant investment [6]. “Business enterprises prosper or fail according to the sophistication and speed of their information systems, and their ability to analyze and synthesize information using those systems.” [7].

The remainder of this work includes chapters organized as follows: Chapter Two provides background information and resources detailing the topics examined in this research. Chapter Three introduces the experimental methods used to compare of these engines, and

Chapter Four presents the results of these experiments. Chapter Five summarizes and discusses the data presented in Chapter Four. Chapter Six explains the significance these results conclude, as well as suggestions for further research avenues on this subject matter.

CHAPTER 2

BACKGROUND

This chapter includes background information and resources detailing the topics examined in this research. Topics include open source software development specifications, a survey of the leading open source database management system and its storage engine technologies, OLAP architectures, OLAP standards, OLAP benchmarking, and open source OLAP products.

2.1 The Open Source Software Development Model

Open source is a philosophy based on the idea of a free exchange of technical information. The idea dates back long before computers, and can be seen as far back as 1911 when Henry Ford won a challenge to the patent of George B. Sheldon [17]. Sheldon attempted to monopolize the automobile industry by patenting the gasoline engine. The ruling in this case created the Motor Vehicle Manufacturers Association, which freely shared technical information about patented automobile technology through licensing agreements [17]. Licensing agreements for openly shared technology is the mechanism that supports open source software.

Sharing of technical specifications for software, or source code, has proven to be an excellent development model because it grants every user the freedom to change the source code [8]. Many of the leading software products available today, such as Linux, Apache, PGP, Perl, and Python, have been developed using the open source development model.

In April 1998, the “Open Source Summit” brought together the leaders of many of the most important open source projects to discuss the benefits, problems, and raise awareness of open source software development. The development model was originally referred to as

freeware or source ware, but many developers of such products were not happy with that name. A result of the session was the establishment of the official name: open source [8].

The Open Source Initiative (OSI), a California based non-profit organization, is steward for the Open Source Definition (OSD). To bear the term "open source", a product must be in compliance with ten requirements of the OSD, listed in Table 1 [18].

Table 1. Open Source Definition Requirements [19]

1. Free Redistribution	The license shall not restrict any party from selling or giving away the software as a component of an aggregate software distribution containing programs from several different sources. The license shall not require a royalty or other fee for such sale.
2. Source Code	The program must include source code, and must allow distribution in source code as well as compiled form. Where some form of a product is not distributed with source code, there must be a well-publicized means of obtaining the source code for no more than a reasonable reproduction cost preferably, downloading via the Internet without charge. The source code must be the preferred form in which a programmer would modify the program. Deliberately obfuscated source code is not allowed. Intermediate forms such as the output of a preprocessor or translator are not allowed.
3. Derived Works	The license must allow modifications and derived works, and must allow them to be distributed under the same terms as the license of the original software.
4. Integrity of The Author's Source Code	The license may restrict source-code from being distributed in modified form only if the license allows the distribution of "patch files" with the source code for the purpose of modifying the program at build time. The license must explicitly permit distribution of software built from modified source code. The license may require derived works to carry a different name or version number from the original software.
5. No Discrimination Against Persons or Groups	The license must not discriminate against any person or group of persons.

(Table 1. continued)

6. No Discrimination Against Fields of Endeavor	The license must not restrict anyone from making use of the program in a specific field of endeavor. For example, it may not restrict the program from being used in a business, or from being used for genetic research.
7. Distribution of License	The rights attached to the program must apply to all to whom the program is redistributed without the need for execution of an additional license by those parties.
8. License Must Not Be Specific to a Product	The rights attached to the program must not depend on the program's being part of a particular software distribution. If the program is extracted from that distribution and used or distributed within the terms of the program's license, all parties to whom the program is redistributed should have the same rights as those that are granted in conjunction with the original software distribution.
9. License Must Not Restrict Other Software	The license must not place restrictions on other software that is distributed along with the licensed software. For example, the license must not insist that all other programs distributed on the same medium must be open-source software.
10. License Must Be Technology-Neutral	No provision of the license may be predicated on any individual technology or style of interface.

There are many different variations of open source software licenses, models, and governance structures [18]. “Each of these elements has profound implications for the type and size of the resulting community, the market penetration and distribution, the ability to recombine with other open source projects, the resistance to unintended or anticipated forks, and more.” [18]. Open source software is commonly differentiated by the type of license agreement to which it adheres.

According to OSI, the most commonly used open source licenses with a strong community are The Apache License 2.0, The New and Simplified BSD licenses, The GNU General Public License (GPL), The GNU Library or "Lesser" General Public License (LGPL),

The MIT license, The Mozilla Public License 1.1 (MPL), and The Common Development and Distribution License. The OSI has sixty-five open source licenses approved for use [20].

The primary driving factor attributed to the popularity and adoption of open source database software in business today is cost savings. As the CEO of RightNow, a large CRM provider, has observed:

"Using MySQL and other open source technologies, RightNow has built an enterprise-class CRM application hosting environment that supports over 3,000 deployments for some of the world's largest organizations…Our systems have facilitated over 1 billion customer interactions on behalf of our clients while maintaining reliability at or above 99.97 percent. Money spent on proprietary databases, when there is a viable open source alternative, is money wasted." [21].

2.2 Open Source Database Storage Engines

A database storage engine is the main software component of a database management system (DBMS). It facilitates the underlying create, read, update, and delete (CRUD) operations performed on the physical data [22].

2.2.1 Storage Engine History

The DBMS examined in this study, MySQL, is based on relational mathematics. This relational model for databases was proposed by E.F. Codd in 1970 [23]. His paper is considered a landmark because it was the first proposal of a disconnection between the logical organization of a database, or schema, from the physical data storage. This has been the standard DMBS architecture ever since, because it separates user operations from the changes in data representation caused by growth. Codd also introduced a normal form for managing the

collection of relationships, as well as operations on relations. This work shaped the modern DBMS, and is the technology used in the open source OLAP system examined in this study [23].

2.2.2 MySql

For over twelve years, MySQL has been the leading open source DBMS, and is the "M" in the popular open source LAMP server stack (Linux, Apache, MySQL, PHP/Perl/Python). Many large, successful companies rely on MySQL to manage their data-driven applications, including Google, Yahoo, NY Times, Cox Communications, The Associated Press, Symantec, Alcatel, Nokia, Nortel, Cisco, and Zappos [24]. MySQL is downloaded over 65,000 times daily and by 2008, MySQL was estimated to have a 50% market share of all database installations with over 16000 paying customers [25]. In terms of open source downloads, MySQL trails only the Mozilla Firefox Browser [24]. The leading news website, Weather.com, switched to MySQL from an unnamed proprietary database, and stated a, "30 percent increased capacity and 50 percent decreased cost" [26].

Several different storage engines are supported by MySQL Server. These various storage engines provide different capabilities to database administrators (DBA's) and software developers.

Since version 5.1, MySQL Server has featured a pluggable storage engine architecture. This architecture enables multiple storage engines to be enabled in a single database instance. This modular architecture enables a DBA to select a specialized storage engine for a particular application, such as transaction processing, data warehousing, Business Intelligence, or high availability, based on the needs of the system [27].

The MySQL server architecture separates software applications from the underlying storage engine via Connector APIs and service layers. Software developers interact with MySQL

through these Connector APIs and service layers as well. If changes to the requirements create the need for a different storage engine, changes to the software under development are not required [27].

Although MySQL offers database designers and administrators many choices when it comes to choosing a specialized storage engine, the MyISAM and InnoDB storage engines are the most widely used [15]. InnoDB became the default storage engine in MySQL version 5.5. In all previous versions, MyISAM was the default storage engine [14].

2.2.2.1 MyISAM. MyISAM is not a transaction-safe storage engine. It is used in situations that value high levels of query throughput over referential-integrity and muti-user concurrency. MyISAM has been identified as a good general engine for data marts and traditional data warehouses. Its primary advantage on query performance is its lack of referential-integrity constraints. Other advantageous features of MyISAM include full-text indexing and decreased database design effort. This simpler storage structure reduces the amount of required server resources on large queries [24, 28, 29]. The primary disadvantage of MyISAM on query performance is its dependence on table-level locking [28, 29].

2.2.2.2 InnoDB. The ACID model sets four requirements that a DBMS must achieve for compliance: atomicity, consistency, isolation and durability. InnoDB is a transaction safe, ACID compliant, MySQL storage engine, recommended for situations where query performance is not the only priority. InnoDB has the popular database transaction features of commit and rollback, as well as crash-recovery capabilities to protect data. InnoDB's primary advantage for query performance is non-locking reads. To reduce disk I/O for common queries, it stores user data in clustered indexes. MySQL claims that, "InnoDB's CPU efficiency is not matched by any other

disk-based relational database engine". The primary disadvantage of InnoDB on query performance is its referential-integrity constraints [28, 29].

In Version 5.5, MySQL made changes to the InnoDB Input Output (I/O) subsystem. These changes are designed to increase the I/O performance, and configurability [14].

In previous versions, InnoDB underutilized server capabilities by prefetching disk blocks and flushing dirty pages with only one background thread. Pages are the basic internal structure used to organize data in the database files. Dirty pages are modified, uncommitted pages, still in the buffer pool [30]. This version enables the utilization of multiple threads [31].

The number of background threads used for page I/O is exposed via system variables, and the default setting is four. Also, the number of I/O operations per second (IOPS) is now an exposed system variable. In previous versions, the IOPS setting was a compile-time parameter. The IOPS rate is a limit that prevents background I/O from exhausting server capability. A higher I/O rate enables the server to process a higher rate of page changes in the buffer pool. Many modern systems can exceed the previous default value, which would unnecessarily restrict I/O utilization [31]. These changes were made to increase system performance and configurability.

2.3 Online Analytical Processing

2.3.1 Analytical and Transactional Processing Incompatibilities

In 1993, E. Codd, S. Codd, and C. Salley, published a paper that describes the need for a new category of database processing called Online Analytical Processing (OLAP) [7]. OLAP has become the de facto data analysis technology used in modern decision support systems, commonly referred to as Business Intelligence (BI) or Executive Information Systems (EIS).

The name OLAP differentiates this new type of analytical database from its transactional counterpart, the Online Transaction Processing (OLTP) database. OLTP databases are the traditional, transaction based databases used to store day to day business data in organizations throughout the world. Transactional databases are structured for short, repetitive, isolated, atomic transactions. Transactions provide an accurate and powerful solution for creating, retrieving, updating, and deleting enterprise data [3]. OLTP provides users with up to date data at a high level of detail, where analytical applications provide summarized, historical, consolidated data, more appropriate for business analysts [5].

Key performance metrics for OLTP databases include a minimization of concurrency conflicts, consistency, recoverability, and transaction throughput. OLAP performance metrics key on query intensive workloads that are mostly ad-hoc and complex, accessing millions of records containing a lot of scans, joins, and aggregates [3].

Efficient OLAP systems must be implemented as entirely separate databases from OLTP systems because the physical design required for adequate performance of each is incompatible. To address the shortcomings of databases that existed in the late 1960's, such as being difficult to maintain, secure, and understand, OLTP databases were developed using relational mathematic theory [7]. Relational databases depend on the Entity-Relationship (ER) data model, which presents significant technical challenges in analyzing large amounts of data [32]. The ER data model reflects a strong emphasis on structure, which is excellent for transactional databases, but neglects architecture for analysis. OLTP databases fail to provide an effective solution for data analysis because it lacks the ability to consolidate, view, and analyze data from multiple perspectives, also referred to as dimensions [7].

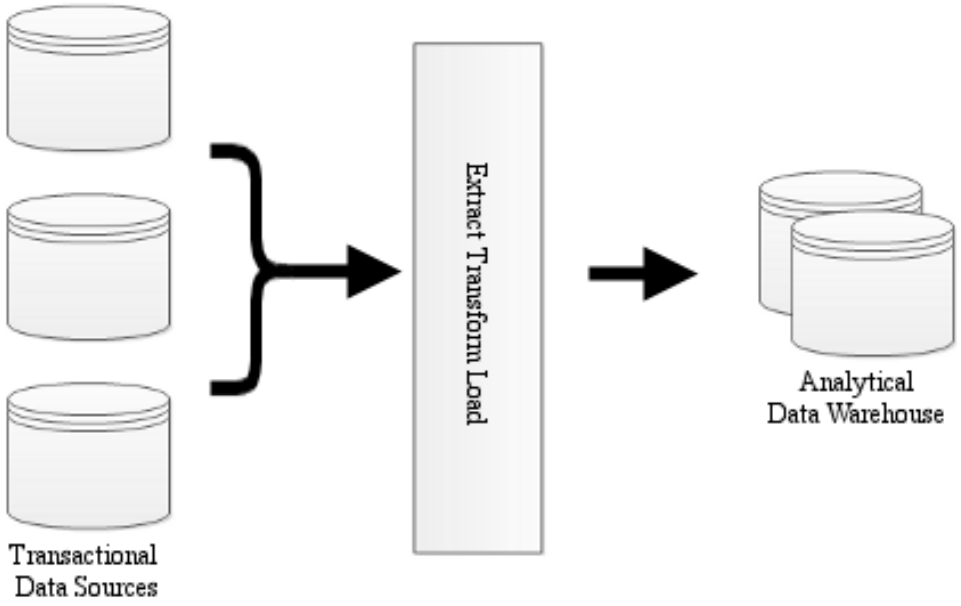


Figure 1. OLAP Server Architecture

2.3.2 Benefit to End Users

Not only has the amount of data stored by organizations experienced a dramatic increase, "The number of individuals within an organization who have a need to perform more sophisticated analysis is growing." [7]. OLAP databases provide many benefits to end users. Data in an OLAP database is arranged by business areas, and these business areas are uniquely defined by the data requirements of an individual company. Once the database administrator defines the business areas, end users have access to their own data organized in familiar fashion.

OLAP front end applications do not require end users, such as business analysts, operational managers, and executives, to learn the query languages. This empowers the end users to satisfy their own data requirements, and avoid reliance on Information Technology (IT) staff [33]. These applications also provide end users with the ability to manipulate data aggregations and formats to perfect their reports.

2.3.3 Multidimensional Data Model

Businesses naturally view themselves from multiple perspectives, such as time, location, and sales. OLAP facilitates this complex, multi-perspective analysis via complex data models, access methods, and implementation methods [3].

The objects being analyzed in this multi-perspective data model are referred to as measures, or facts. The perspectives for which the measure is viewed are referred to as dimensions. A dimensions scope is defined by its attributes, and the most commonly analyzed dimension is time, due to its significance in trend analysis [3]. The simultaneous analysis of measures along multiple dimensions is referred to as multidimensional data analysis [7]. Each unique set of dimensions and measures is referred to as a multidimensional cube.

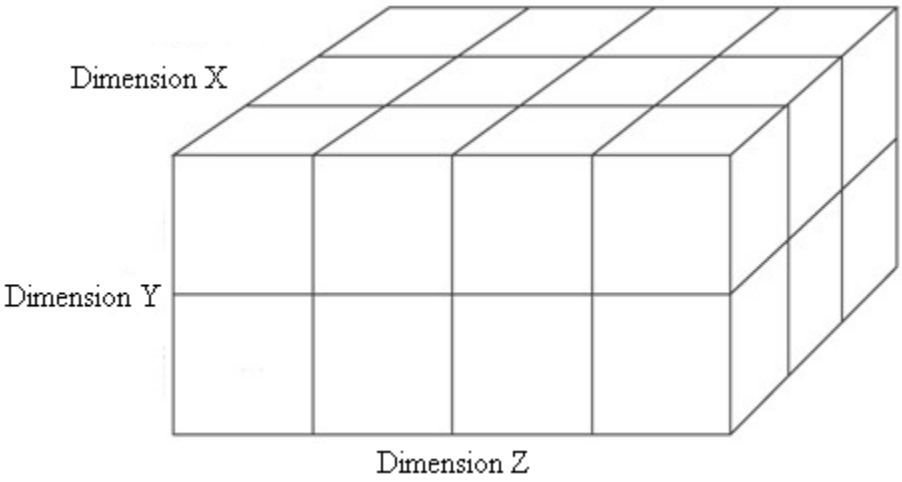


Figure 2. Multidimensional Data Structure

The historical nature of OLAP systems require summarized data for users to analyze. Data consolidation is the process of summarizing large amounts of data into single blocks of useful knowledge [7]. Hierarchies give OLAP engines a structure to effectively consolidate otherwise flat dimensions. A strong support for hierarchies is the principal conceptual feature that distinguishes the multidimensional and relational data models. The attributes of a dimension

related along hierarchical relationships, or consolidation paths, are shown in Figure 3 [3]. There are substantial differences in the hierarchy implementation methods among OLAP products [34].

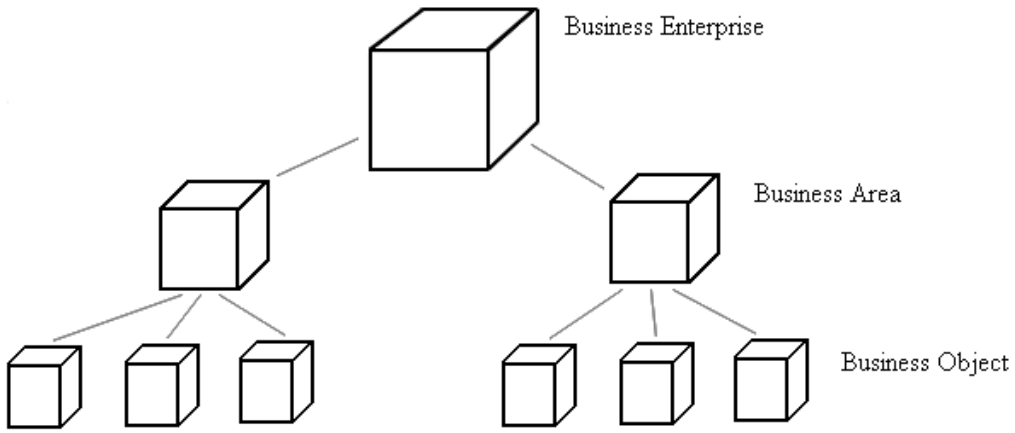


Figure 3. Consolidation Paths

2.3.4 OLAP Operations

Before multidimensional data analysis, data analysts were unable to efficiently change between data dimensions and aggregate levels of detail. Data analysts interactively navigate a cube using the OLAP operations rotate and slice and dice [35]. OLAP facilitates these operations efficiently using vector arithmetic [36].

Changing levels of consolidation is referred to as drilling down, or rolling up. A drill down is the downward traversal of the hierarchy from the most summarized level to the most detailed level [5]. A roll up, or aggregation, refers to the opposite, upward hierarchy traversal. Rotation, also referred to as pivoting, is changing the dimensional orientation of a measure [35]. A slice is a subset of a measure, where a specified value corresponds to an attribute of a dimension [35]. Slicing and dicing are the user initiated processes associated with navigating through slices using rotation, drill down and drill up [35].

OLAP operations frequently span multiple consolidation paths and dimensions [7]. These consolidations commonly involve complex statistical equations and computations such as

moving averages, percentage change between time periods, and inter-dimensional comparisons, such as sales and budget [5].

The complex processing involved with multidimensional data analysis and these consolidation operations is extremely cumbersome on database systems. Optimizing these processes was the primary motivation behind the development of OLAP [7].

2.3.5 OLAP Engine Architectures

2.3.5.1 ROLAP. Relational Online Analytical Processing (ROLAP) Engines use middleware between back end relational databases and front end OLAP client tools. The middleware generates indices, views, and multi-statement SQL for the relational database [5].

ROLAP uses a rich multidimensional metadata model called a star schema to implement OLAP on relational data stores. Star schemas implement a table for each multidimensional measure, referred to as a fact table. Dimensions are also implemented as tables with the columns representing its attributes. The rows of the fact table form foreign key relationships to the dimension tables, as illustrated in Figure 4. Breaking the normalization rules that that preserve the accuracy of relational schemas by star schemas is acceptable because OLAP databases are not required to be instantaneously accurate [7].

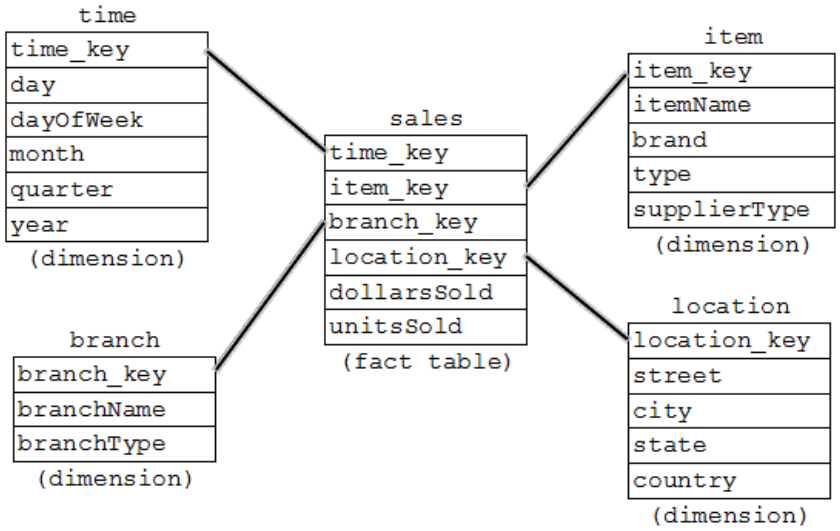


Figure 4. Star Schema

Although computing joins between fact tables and dimension tables are more efficient than arbitrary relations, intrinsic mismatches between the relational model and the multidimensional model can create performance bottlenecks [5]. As an effort to offset performance bottlenecks, ROLAP engines implement a multitude of optimization techniques.

Snowflake schemas are used to improve storage utilization via dimension table normalization [5]. Normalizing the dimension tables creates explicitly represented attribute hierarchies that eliminate redundancy [37]. Fact constellation schemas are snowflake schemas where multiple fact tables share dimension tables to improve storage utilization, as illustrated in Figure 5 [5]. Snowflake and fact constellation schemas increase query complexity and reduce query performance [37].

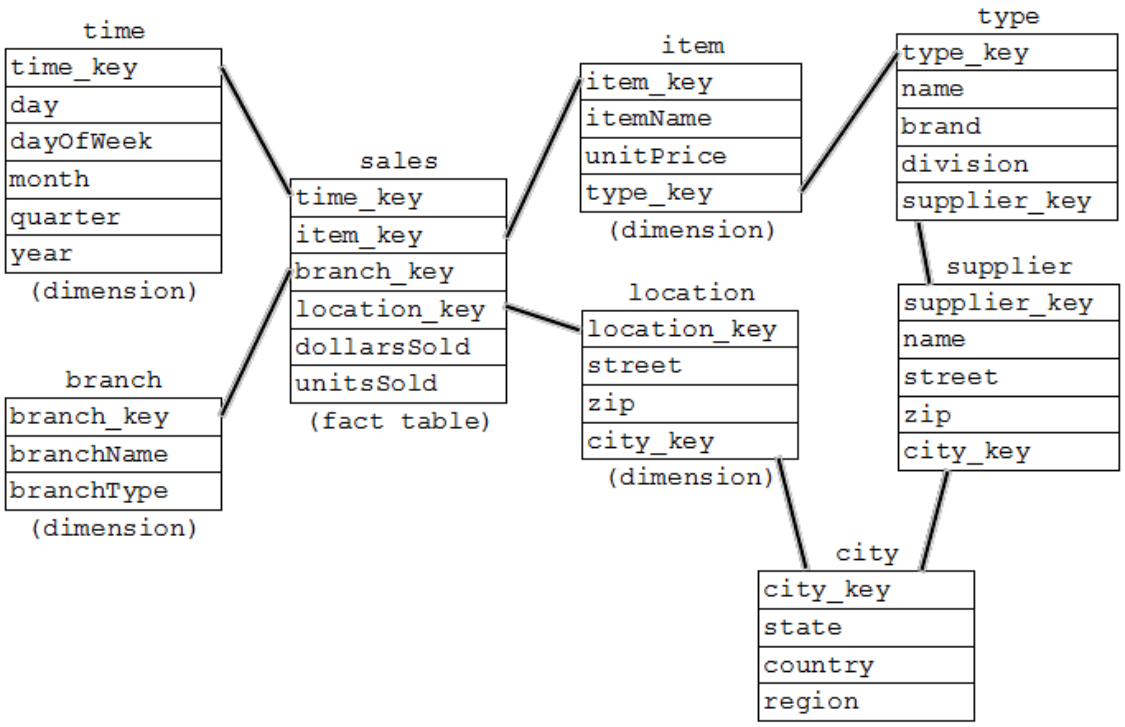


Figure 5. Snowflake Schema

If maximum storage utilization is not a requirement, join indices and materialized views can increase processing performance for complex queries. Join indices index key relationships between fact and dimension tables, and materialized views store commonly aggregated data. Join indices are effective for data selective queries, but ineffective for data intensive queries because they often require entire relations to be scanned sequentially. The difficulties in implementing join indices and materialized views are proper identification, effective use, and proper updating [5].

ROLAP vendors implement star transformations to improve star schema query performance. Rather than computing a Cartesian product of the dimension tables, this process combines bitmap indexes on individual fact table columns. This improves performance in cases with sparse data sets or a large number of dimensions [37].

2.3.5.2 MOLAP. Multidimensional Online Analytical Processing (MOLAP) Engines implement OLAP in its quintessential form, through direct mapping on a multidimensional storage layer [3]. The disadvantage of MOLAP is inefficient storage utilization for sparse data sets [5]. It is addressed through extensive data compression and two level storage representations [5]. Using two level data representations, a technique acquired from statistical databases, uses elements of a dense data set to index smaller sparse data sets [5].

2.3.5.3 HOLAP. Hybrid Online Analytical Processing (HOLAP) combines features of ROLAP and MOLAP. HOLAP engines divide complex queries into sub queries based on data set density. Queries spanning dense data sets are processed by MOLAP subsystems, and queries spanning sparse data sets are processed by ROLAP subsystems. The result sets are then combined and presented to the end user [5].

HOLAP systems store the majority of the data in relational databases to avoid problems attributed to sparsity. The multidimensional part of the system stores only the frequently accessed data. If the multidimensional data cannot answer the query, the system then transparently accesses the relational part of the system [38].

2.3.6 Data Homogenization. Data homogenization is the process of processing data from diverse physical data representations, including flat files, data feeds, and databases, into a logically consistent data structure and central location for analysis and reporting [7]. These diverse sources may contain data of varying quality, codes, and formats, which must be reconciled prior to data load [3]. This process is often referred to as Extract Transfer Load (ETL), and is a prerequisite to OLAP.

Approximate query processing, sometimes referred to as Approximate Query Answering (AQUA) is the process of answering queries using small, precompiled data samples, or synopsis to answer arbitrary aggregate queries [5, 39]. This part of data homogenization is intended to improve query processing performance.

Building an effective data warehouse schema can take years due to complex business modeling. To expedite this process, businesses often opt to create departmental data marts. Disjoint analysis and analysis synthesis are unfavorable facets of data mart implementation [5].

2.4 Open Source Online Analytical Processing Engines

The availability of production ready open source OLAP engines is limited, and only two were found. Mondrian is a ROLAP server, and Palo is a memory-based MOLAP server. The Mondrian OLAP engine has become the de facto OLAP engine for open source solutions [40]. It is not only included in the Pentaho BI suite, but also in other open source BI suites [12].

2.4.1 Mondrian

Mondrian began as an independent open source project in 2002 [12]. In 2005, Mondrian became a part of Pentaho's BI package. The most recent version of Mondrian is 3.3 (2012), is released under the Common Public License (CPL). Mondrian is a Java application, so it can run on platform with a Java Runtime Environment (JRE). It also uses Java Database Connectivity (JDBC), which can be integrated with most modern DBMSs. The Mondrian documentation consists of public facing web pages that total close to 200 pages of printed text. The Mondrian user forums are used actively [12].

The Mondrian project was involved in the standardization of olap4j, which is a common Java application programming interface (API) for OLAP servers. Olap4j is the preferred Mondrian API. Mondrian also supports XML for Analysis (XMLA) [41]. Mondrian processes

MDX queries, and cube schemas are specified in XML. Mondrian is scalable to large data sets, and is limited only by the underlying DBMS because it delegates aggregation to the DBMS [12].

2.4.2. Palo

Palo is a MOLAP server developed by Jedox AG [12]. It has a version released under the GPL, however this version has limited functionality. Documentation exists, but costs €29.50. Version 2.5 was released in early July 2008. Data sets are loaded into memory, thus data sets a limited to server memory allocation. Proprietary programming interfaces are required to communicate with Palo. There is also a free, but closed-source client add-in for Microsoft Excel. This Excel add-in is the primary client user interface [12].

2.4.3. Comparison

Due to the differences in architecture, standards compliance, and open source features, Mondrian was chosen to implement this experiment.

2.5 Industry Standards

2.5.1 Java Database Connectivity

The Java Database Connectivity (JDBC) framework gives Java application developers a common database access method that is platform agnostic. The classes of the JDBC API are open source and available from the Sun website (http://docs.oracle.com/javase/7/docs/api/).

JDBC builds on the existing Open Database Connectivity (ODBC) standard, increasing the abstraction level. ODBC is a standard that consolidated the commonality between DBMS's. JDBC-ODBC bridges exists to enable allow Java programs to connect to existing ODBC-enabled database software [42].

2.5.2 Multidimensional Expressions

Multidimensional Expressions (MDX) is the de facto query language for multidimensional data. It was released in 1998 by Microsoft as the language component of the OLE DB for OLAP [43].

```
SELECT [<axis_specification>
       [, <axis_specification>...]]
  FROM [<cube_specification>]
[WHERE [<slicer_specification>]]
```

Figure 6. MDX Query Syntax [44]

At first glance, MDX syntax in figure 6 appears similar to the Structured Query Language (SQL) traditionally used with relational databases. However, MDX is a completely new language with its own combinations of identifiers, expressions, operators, functions, comments, and keywords [44]. Translate MDX queries into traditional SQL queries would require synthesis of large SQL expressions for very simple MDX expressions [43].

2.4.3 Extensible Markup Language for Analysis

The Extensible Markup Language for Analysis (XMLA) is the standard API for data interaction between OLAP Servers and Clients, and is illustrated in figure 7 [45]. The communication of data is implemented using the Hypertext Transfer Protocol (HTTP), Simple Object Access Protocol (SOAP), and Extensible Markup Language (XML) web standards. Using these web standards allow for development of hardware, platform, and location independent applications capable of implementing thin client architecture [45].

```
<Execute xmlns="urn:schemas-microsoft-com:xml-analysis"
SOAP-ENV:encodingStyle=
"http://schemas.xmlsoap.org/soap/encoding/">
<Command>
<Statement>
     select [Measures].members on Columns from Sales
</Statement>
<Command>
<Properties>
<PropertyList>
<DataSourceInfo>
Provider=Essbase;Data Source=local;
</DataSourceInfo>
<Catalog>Foodmart 2000</Catalog>
<Format>Multidimensional</Format>
<AxisFormat>ClusterFormat</AxisFormat>
</PropertyList>
</Properties>
</Execute>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

Figure 7. XMLA Execute Request [41]

2.6 Online Analytical Processing Performance Benchmarks

2.6.1 Analytical Processing Benchmark

The OLAP Council was an organization established to advocate the advancement of OLAP technology. “The mission of the OLAP Council is to educate the market about OLAP technology, provide common definitions, sponsor industry research and help position OLAP technology within a broader IT architecture”. The OLAP council also worked to establish standard OLAP terminology, interoperability guidelines, and the first analytical benchmark, the Analytical Processing Benchmark (APB-1) [35].

The APB-1 has been succeeded by the Decision Support Benchmark (TPC-DS) developed by the Transaction Processing Performance Council.

2.6.2 TPC-DS

For the last fifteen years, the TPC-D benchmark, and its successor TPC-H, have been used by industry and the research community to evaluate DSS performance. The Transaction Processing Performance Council recognized a paradigm shift in the industry and developed this TPC-DS benchmark. TPC-DS is now the industry standard DSS benchmark [16].

TPC-DS measures query throughput under a complex, controlled, multi-user workload for a given system under test (SUT), which includes server hardware, operating system, and database configuration. It is used by vendors to demonstrate system capabilities, by customers in purchasing software and servers, and by the research community for optimization development [46]. The database schema, data population, queries, and implementation rules are designed to broadly represent modern decision support systems and provide highly comparable, controlled, vendor-neutral, and repeatable tasks [46]. TPC-DS tests the upper boundaries of system performance by examining a large volume of data and answering real world business questions by executing queries of various complexities [47].

The schema, an aggregate of multiple star schemas, models the decision support functions of a typical multichannel retail product supplier contains essential business information, such as detailed customer, order, and product data in store, catalog, and internet channels [16]. The benchmark data generation utility uses real world data with common data skew where possible, such as seasonal sales and frequent names. A retail model helps readers relate the components of the benchmark intuitively [1].

The TPC-DS workload could be used to describe any retail supplier using BI to address complex business problems. The queries analyze and convert store, web and catalog sales channels operational facts into business intelligence using a variety of access patterns, query

phrasings, operators, and answer set constraints [16]. An intense query workload is necessary to preserve a realistic context [1].

2.7 Summary

This chapter included background information and resources detailing the topics examined in this thesis. Topics include open source software development specifications, a survey of the leading open source database management system and its storage engine technologies, OLAP architectures, OLAP standards, OLAP benchmarking, and open source OLAP products.

CHAPTER 3

EXPERIMENTAL METHODS

3.1 Motivation

As stated previously, this work investigated the performance of leading database storage engines in support of Online Analytical Processing (OLAP) in an open source environment. OLAP has become the de facto data analysis technology used in modern decision support systems, commonly referred to as Business Intelligence (BI) or Executive Information Systems (EIS). Decision support systems have experienced tremendous growth and are among the top priorities for enterprises [2]. All principal DBMS (e.g. Oracle, Microsoft, and IBM) vendors now have offerings in the data warehousing and OLAP technologies [3].

This work investigated the performance of leading database storage engines for OLAP developed using the open source model using the following steps:

- A brief analysis to confirm the storage engines are in compliance with open source requirements.
- A technical comparison of the storage engines, and identification of fundamental components that support OLAP.
- A complete performance benchmark comparison of the storage engines based on the Decision Support Benchmark (TPC-DS), as defined by the Transaction Processing Council.
- Two additional TPC-DS benchmark tests, executed on scaled data sets to determine if execution times scale in proportion to data set size.

Previous research has been performed that surveys open source BI tools [12]. Recent benchmarks have been performed that explore the leading open source database engines, and

one, InnoDB, has undergone extensive updates in its most recent version [13, 14, 15]. Recent data on the performance of the leading open source database engines in support of OLAP has not been published with respect to a standard benchmark, or the recent updates to InnoDB. The work described here has sought to address this gap in the literature and determine the performance of the leading open source database storage engines in support of OLAP. This work is timely due to the strained global economic situation, the emergence of the TPC-DS as an industry standard OLAP benchmark, and the recent release of updates to the InnoDB storage engine [14, 16].

During a strained economic situation it is generally acceptable to simply maintain market share and profitability, but with BI, organizations can actually increase these metrics [6]. Open source BI has become an effective alternative to proprietary systems in terms of cost and function [11]. The top reason for adoption is still cost savings, although reduced vendor dependence and ease of integration followed closely behind [11]. BI leverages existing Information Technology (IT) infrastructure, such as Enterprise Resource Planning (ERP) and Customer Relationship Management (CRM), by analyzing data previously stored in these existing systems without significant investment [6]. "Business enterprises prosper or fail according to the sophistication and speed of their information systems, and their ability to analyze and synthesize information using those systems." [7].

3.2 Open Source Compliance

The database engines in this study were examined to determine if they are in compliance with ten requirements of the OSD, as this research attempted to assist the adoption of open source database software only. Storage engines included in this research must be released with one of the sixty-five open source licenses approved for use by the OSI.

3.3 Feature Comparison

The storage engines in this study were examined to identify fundamental database components and technologies. The differences in these components and technologies have brought to bear declarative performance differences.

3.4 Online Analytical Processing Performance Benchmark

3.4.1 Benchmark Overview

TPC-DS measures query throughput under a complex, controlled, multi-user workload for a given system under test (SUT), which includes server hardware, operating system, and database configuration. It is used by vendors to demonstrate system capabilities, by customers in purchasing software and servers, and by the research community for optimization development [46]. The database schema, data population, queries, and implementation rules are designed to broadly represent modern decision support systems, and provide highly comparable, controlled, vendor-neutral, and repeatable tasks [46].

3.4.2 Data Model

The schema, an aggregate of multiple star schemas, models the decision support functions of a typical multichannel retail product supplier. It contains essential business information, such as detailed customer, order, and product data in store, catalog, and internet channels [16]. The benchmark data generation utility uses real world data with common data skew where possible, such as seasonal sales and frequent names. This retail model helps readers relate the components of the benchmark intuitively [1].

The TPC-DS star schemas consist of seven fact and seven dimension tables. The fact tables include a pair of fact tables, sales and returns, for each of the three sales channels. The remaining fact table models product inventory. Each fact table has a correlated cube definition.

The seven dimensions are item, date, store, customer, customer address, customer demographics, household demographics, and promotion. The seven dimension tables are used across the multiple cubes, and each contain a single column surrogate key, which is used to join the fact tables [1].

A detailed definition of the schemas can be found in the appendices. Appendix A contains the relational schema, appendix B contains the multidimensional schema, and appendix C contains the cube diagrams.

3.4.3 Database Population

The TPC-DS specification mandates the supplied data generator, dsgen, be used to generate data for population of the SUT database. dsgen source code is included as part of the electronically downloadable portion of the specification, and benchmark implementers are permitted to modify dsgen [1]. However, this research did not require dsgen modification.

The specification defines scale factor (SF), which determines the approximate raw size of the data produced by dsgen [1]. This research used an initial SF of 10GB, which creates 12GB of raw data, and is displayed in table 2. In order to realistically scale the benchmark, the fact tables were scaled to create raw data sets of 9GB and 6GB. Dimensions tables were not scaled [46].

Table 2. TPC-DS Raw Table Size

Table	Row Size (bytes)	Row Count (10GB SF)
call_center	305	24
catalog_page	139	12000
catalog_returns	166	1439749
catalog_sales	226	14401261
customer	132	500000
customer_address	110	250000
customer_demographics	42	1920800
date_dim	141	73049
household_demographics	21	7200

(Table 2. continued)

income_band	16	20
inventory	16	133110000
item	281	102000
promotions	124	500
reason	38	45
ship_mode	56	20
store	263	102
store_returns	134	2875432
store_sales	164	28800991
time_dim	59	86400
warehouse	117	10
web_page	96	200
web_returns	162	719217
web_sales	226	7197566
web_site	292	42

3.4.4 Data Analysis

The TPC-DS workload could be used to describe any retail supplier using BI to address complex business problems. The queries analyzed and converted the store, web and catalog sales channels operational facts into business intelligence using a variety of access patterns, query phrasings, operators, and answer set constraints [16]. An intense query workload was necessary to preserve a realistic context [1].

Although a modern DSS should support users with diverse needs such as reporting, ad-hoc, iterative OLAP, and data mining, this research is focused on OLAP performance. The OLAP queries implemented in this research were constructed to analyze large sets of business data, answer specific business questions, and determine meaningful trends. Due to the nature of changing business environments and diverse user needs, this research can only provide DBAs a limited degree of fore knowledge for performance planning [1].

3.4.5 Query Implementation

The query streams were implemented using a scenario-based user session. Each session contained a query sequence with each query leading to another. This model allows the benchmark to capture important aspects of the complex, iterative nature of OLAP queries [1]. Each user session contained an identical set of queries, implemented as a JMeter user thread.

Apache JMeter was used to drive all testing and data collection. JMeter is an open source Java client application designed to load test functional behavior and measure system performance under a concurrent load [48]. JMeter test plans detail the steps executed by JMeter. The complete TPC-DS JMeter test plan implemented for this research can be found in appendix E. Table 3 describes the queries and the results sets they retrieve.

Table 3. TPC-DS Queries

	MDX Query	Query Description
Query 1	Select [Measures].[Net Loss] on 0, [Item.Manufacturer].Members on 1, { [Web Page].[AAAAAAAAACAAAAAA], [Web Page].[AAAAAAAABAAAAAAA] } on 2 From [Web Returns]	Net loss, manufacturer, and web page from the web returns cube.
Query 2	Select [Measures].[Count] on 0, [Store.Location].[TN] on 1 From [Store Sales] Where { [Household Demographics].[1], [Household Demographics].[2], [Household Demographics].[3] }	Store locations in Tennessee from the store sales cube with buying potential less than $500.
Query 3	Select [Measures].[Profit] on 0, NonEmptyCrossJoin({ [Web Page].CurrentMember }, { [Item.Category].[5].Children, [Item.Category].[7].Children, [Item.Category].[10].Children }) on 1 From [Web] Where ([Date].[2000])	Profit for each item category, website member, and item category is music, home, or electronics from the web cube in 2000.

(Table 3. continued)

Query 4	Select [Measures].[Profit] on 0, NonEmptyCrossJoin({ [Web Page].CurrentMember }, { [Item.Category].[5].Children, [Item.Category].[8].Children, [Item.Category].[9].Children }) on 1 From [Web] Where ([Date].[1999])	Profit for each item category, website member, and item category is in music, sports, or books from the web cube in 1999.
Query 5	Select [Measures].[Total Net Loss] on 0, Filter({ [Call Center].Members }, [Call Center].[Manager].CurrentMember.Name = ‘Larry Mccray’ OR [Call Center].[Manager].CurrentMember.Name = ‘Mark Hightower’) on 1, { [Item.Category].[7].Children, [Item.Category].[9].Children} on 2 From [Catalog Returns] Where [Returned Date].[2002]	Total net loss from call centers managed by Larry McCray or Mark Hightower, and the item is in the home or books category from the catalog returns cube returned in 2002.
Query 6	With Member [Measures].[Item Color] as [Item.Item Info].CurrentMember.Properties(“Color”) Member [Measures].[Item Description] as [Item.Item Info].CurrentMember.Properties(“Description”) Select { [Measures].[Quantity], [Measures].[Item Color], [Measures].[Item Description] } on 0, { [Item.Item Info].[AAAAAAAAAAABAAAA], [Item.Item Info].[AAAAAAAAAAABDAAAA], [Item.Item Info].[AAAAAAAAAAADEAAAA], [Item.Item Info].[AAAAAAAAOENAAAAA] } on 1 From [Inventory]	Quantity, item color, description, and info for four items from the inventory cube.
Query 7	Select [Measures].[Net Loss] on 0, [Customer Demographics].[F].Children on 1, { [Date].[2002].[1].[1], [Date].[2002].[1].[2], [Date].[2002].[1].[3] } on 2 From [Web Returns]	Net loss, customer demographics, and date from the customer sales cube.

(Table 3. continued)

Query 8	Select [Measures].[Total Quantity] on 0, CrossJoin ({ [Item.Category].[7],[Item.Category].[10] }, [Promotion].[Email].Members) on 1 from [Catalog Sales]	Total quantity for the home and electronics categories from the catalog sales cube.
Query 9	Select [Measures].[Total Quantity] on 0, CrossJoin ({ [Item.Category].[3],[Item.Category].[6] }, [Promotion].[Email].Members) on 1 From [Catalog Sales]	Total quantity for the music and children categories from the catalog sales cube.

3.4.6. Performance Statistics

The primary TPC-DS performance statistic specified is Queries per Hour for Decision Support (QphDS). Two types of performance tests are specified by the TPC-DS, power tests (Tpt) and throughput tests (Ttt) [1].

Power Tests measure the performance of the SUT when processing a sequence of queries in a single stream fashion. These queries were executed in numerical order with only one query active at a time [1]. The power tests provide a statistic for comparison against concurrent session tests.

Throughput Tests measure the performance of the SUT when processing multiple concurrent user sessions. Each test is required to execute a minimum of 20 sessions, and queries were executed in random order. The throughput tests provide statistics (TTTn) for calculation in QphDS [1]. QphDS is calculated as follows in figure 8:

$$QphDS@SF = \frac{99 * 2 * 3600}{T_{tt1} + T_{tt2}}$$

Where:

- 99 is the number of queries per stream
- 2 is the number of query runs
- 3600 is the number of seconds in an hour
- TTT1 is the total elapsed time to complete the first throughput test
- TTT2 is the total elapsed time to complete the second throughput test
- SF is the scale factor used in the benchmark

Figure 8. Queries per Hour for Decision Support (QphDS) Formula

For each query, one atomic transaction was completed. The data reported for all benchmark tests includes the start time, finish time, and execution time interval. The interval time for each query executed must be individually reported, and rounded to the nearest millisecond. To avoid zero values, values less than five tenths of a millisecond are rounded up to one millisecond [1]. The minimum, 25th percentile, median, 75th percentile, and maximum times, along with standard deviation were also reported.

The QphDS calculation in this research varies from the TPC specification by omitting the data loading time. Data loading time is an ETL function outside the scope of this research. Details for steps to configure this TPC-DS benchmark implementation, including server configuration, are disclosed in Appendix D.

3.5 Server Configuration

The experiments were executed on a server with a 64-bit Intel Pentium G620 CPU (2.60 GHz) and 8GB of RAM. The server is operated by a 64-bit Debian installation, version 6.0 (Squeeze). The hard drive is a Western Digital WDC-1600JS (160GB) with a maximum external transfer speed of 3GB/sec, average seek time of 8.9ms, average rotational latency of 4.2ms,

spindle speed of 7200 RPM, and an 8MB Cache. MySQL version 5.1.61, Mondrian version 3.3, and dsgen version 1.3 were used in this research. The server architecture is illustrated in figure 9.

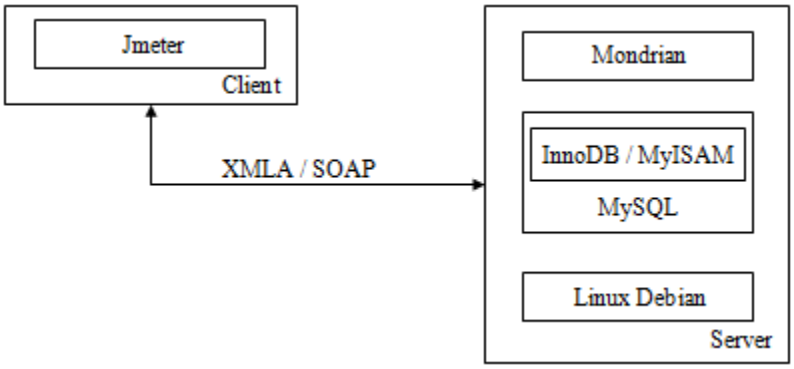


Figure 9. Server Architecture

CHAPTER 4

RESULTS

In this section, the representative results of this experiment are presented. A complete list of all experiment results can be found in Appendix F.

4.1 Feature Summary

The following feature summary presents general technical features and information for MyISAM and InnoDB. Identifying these features is important to help develop hypotheses on engine behavior, performance, and applicability.

Both MyISAM and InnoDB are released under the GPLv2. The storage limit of MyISAM is 256TB and 64TB for InnoDB. Unlike InnoDB, MyISAM is not an ACID compliant, transaction safe database. MyISAM employs table level locking, where InnoDB employs locking at the row level. Both support geospatial data types, but only MyISAM supports geospatial indexing. Both support B-Tree indexes, index caches, data compression, data encryption, replication, point in time recovery, query cache, and update statistics for the data dictionary. Neither support hash indexes or clustering. InnoDB does not support full-text search indexes; MyISAM does. InnoDB does support clustered indices, data caches, multi-version concurrency control, and foreign keys, where MyISAM does not. Table 4 provides a summary of the major features in these database engines.

Table 4. Feature Summary

Feature	MyISAM	InnoDB
License	GPL v2	GPL v2
Storage limits	256TB	64TB
Transactions	No	Yes
Locking granularity	Table	Row
Multi-Version Concurrency Control	No	Yes

(Table 4. continued)

Geospatial data type support	Yes	Yes
Geospatial indexing support	Yes	No
B-tree indexes	Yes	Yes
Hash indexes	No	No
Full-text search indexes	Yes	No
Clustered indexes	No	Yes
Data caches	No	Yes
Index caches	Yes	Yes
Compressed data	Yes	Yes
Encrypted data	Yes	Yes
Cluster database support	No	No
Replication support	Yes	Yes
Foreign key support	No	Yes
Backup / point-in-time recovery	Yes	Yes
Query cache support	Yes	Yes
Update statistics for data dictionary	Yes	Yes

4.2 TPC-DS Benchmark Results

The benchmark specified by the TSC-DS, and implemented in this study, captures important aspects of the complex and iterative nature of OLAP queries [1]. Each user session contained an identical set of queries, and was implemented as a JMeter user thread. Each test was executed once to prefill the MySQL query cache, with two subsequent runs recorded.

4.2.1 TPC-DS Power Test Statistics

Two TPC-DS Power Tests (tpt1 and tpt2) measured SUT performance when processing a sequence of ninety nine queries in a single stream. Queries were executed in numerical order with only one query active at a time [1]. Figures 10 – 18 present plots of the minimum, first quartile, median, third quartile, and maximum execution times for each power test query

executed. These power test execution time distributions provide comprehensive visual summary for comparison against the concurrent session benchmark.

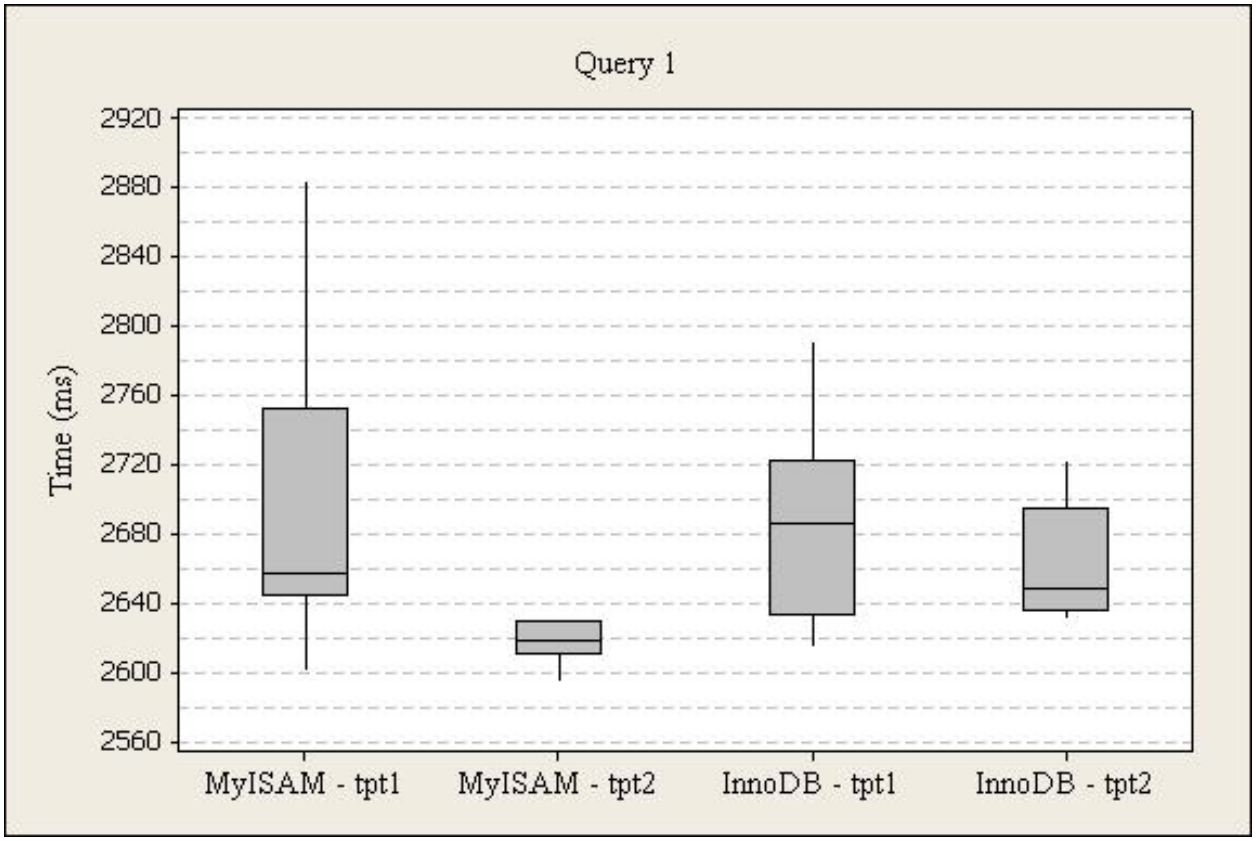


Figure 10. Power Test Query 1 Box Plot

The execution times for query one (net loss, manufacturer, and web page from the web returns cube), run one were higher for InnoDB (M=2961, SD=939) than MyISAM (M=2703, SD=90). For InnoDB, the times range from 2479ms to 5921ms, and 2633ms to 2722ms from the first to third quartile with a 2686ms median. For MyISAM, the times range from 2601ms to 2885ms, and 2644ms to 2753ms from the first to third quartile with a 2658ms median.

The execution times for query one, run two were higher for InnoDB (M=2929, SD=846) than MyISAM (M=2628, SD=35). For InnoDB, the times range from 2631ms to 5603ms, and 2636ms to 2695ms from the first to third quartile with a 2648ms median. For MyISAM, the

times range from 2549ms to 2706ms, and 2611ms to 2630ms from the first to third quartile with a 2618ms median.

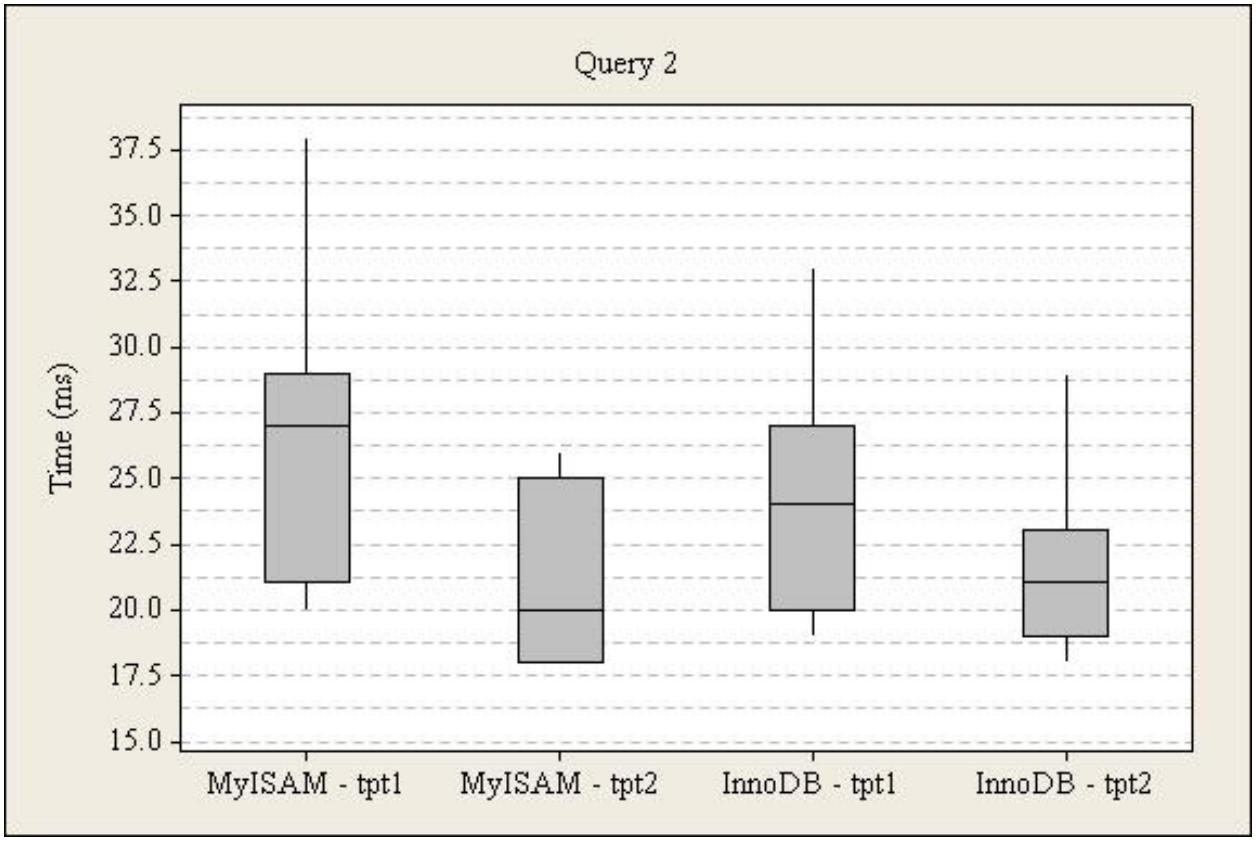


Figure 11. Power Test Query 2 Box Plot

The execution times for query two (store locations in Tennessee from the store sales cube with buying potential less than $500), run one were lower for InnoDB (M=25, SD=6) than MyISAM (M=25, SD=6). For InnoDB, the times range from 19ms to 41ms, and 20ms to 27ms from the first to third quartile with a 24ms median. For MyISAM, the times range from 20ms to 38ms, and 21ms to 29ms from the first to third quartile with a 27ms median.

The execution times for query two, run two were higher for InnoDB (M=21, SD=3) than MyISAM (M=20, SD=3). For InnoDB, the times range from 18ms to 29ms, and 19ms to 23ms from the first to third quartile with a 21ms median. For MyISAM, the times range from 18ms to 26ms, and 18ms to 25ms from the first to third quartile with a 20ms median.

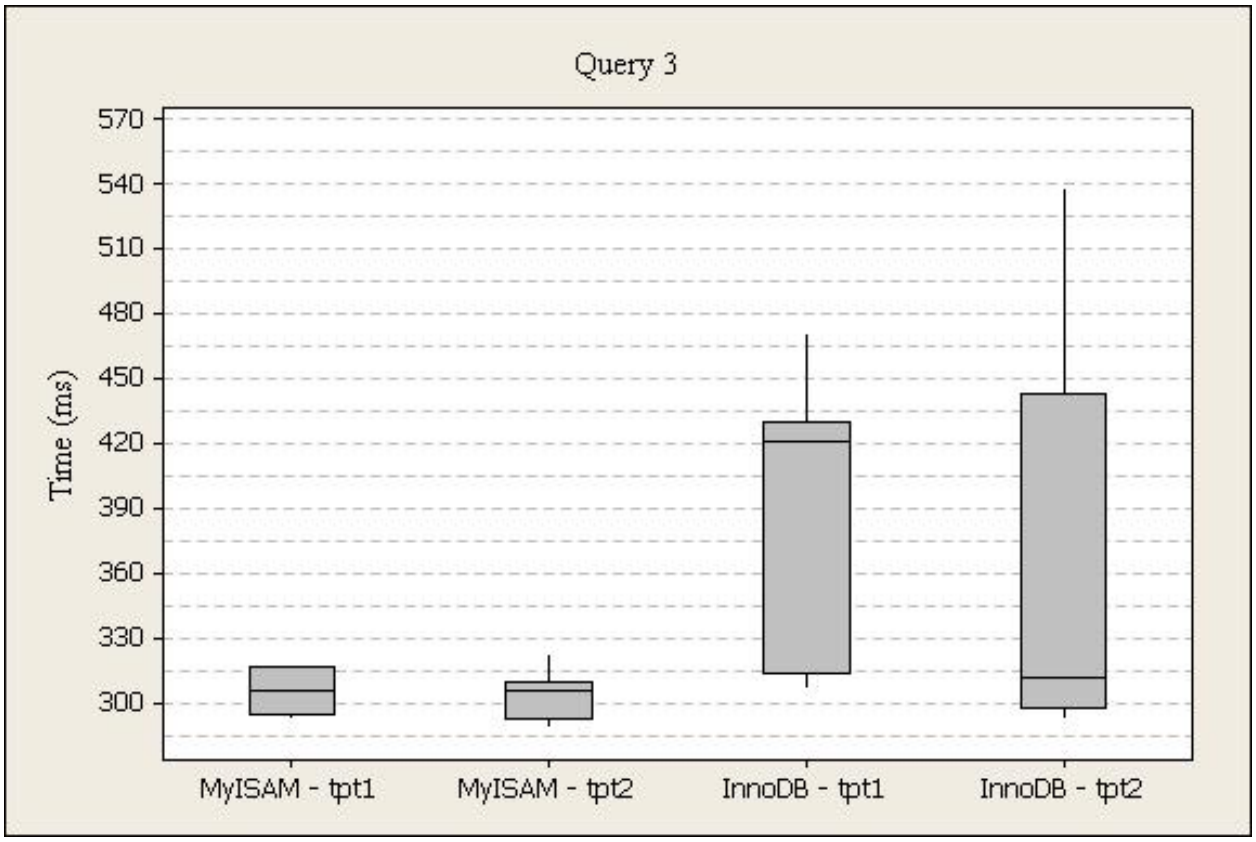


Figure 12. Power Test Query 3 Box Plot

The execution times for query three (profit for each item category, website member, and item category is music, home, or electronics from the web cube in 2000), run one were higher for InnoDB (M=389, SD=60) than MyISAM (M=330, SD=59). For InnoDB, the times range from 307ms to 472ms, and 314ms to 431ms from the first to third quartile with a 421ms median. For MyISAM, the times range from 293ms to 454ms, and 295ms to 317ms from the first to third quartile with a 306ms median.

The execution times for query three, run two were higher for InnoDB (M=353, SD=81) than MyISAM (M=302, SD=10). For InnoDB, the times range from 293ms to 539ms, and 298ms to 444ms from the first to third quartile with a 312ms median. For MyISAM, the times range from 289ms to 323ms, and 293ms to 310ms from the first to third quartile with a 306ms median.

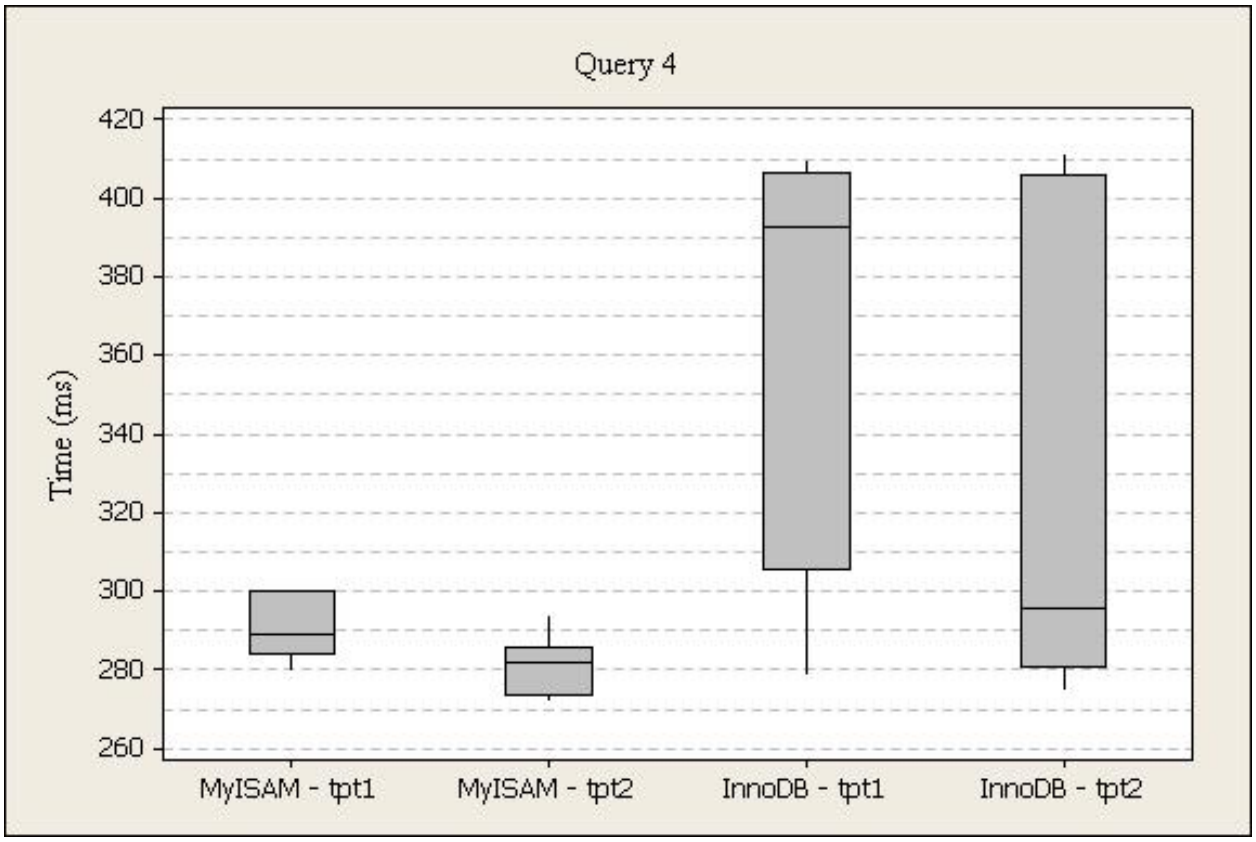


Figure 13. Power Test Query 4 Box Plot

The execution times for query four (profit for each item category, website member, and item category is in music, sports, or books from the web cube in 1999), run one were higher for InnoDB (M=362, SD=53) than MyISAM (M=313, SD=54). For InnoDB, the times range from 279ms to 410ms, and 306ms to 407ms from the first to third quartile with a 393ms median. For MyISAM, the times range from 280ms to 435ms, and 284ms to 300ms from the first to third quartile with a 289ms median.

The execution times for query four, run two were higher for InnoDB (M=345, SD=114) than MyISAM (M=281, SD=7). For InnoDB, the times range from 275ms to 674ms, and 281ms to 406ms from the first to third quartile with a 296ms median. For MyISAM, the times range from 272ms to 294ms, and 274ms to 286ms from the first to third quartile with a 282ms median.

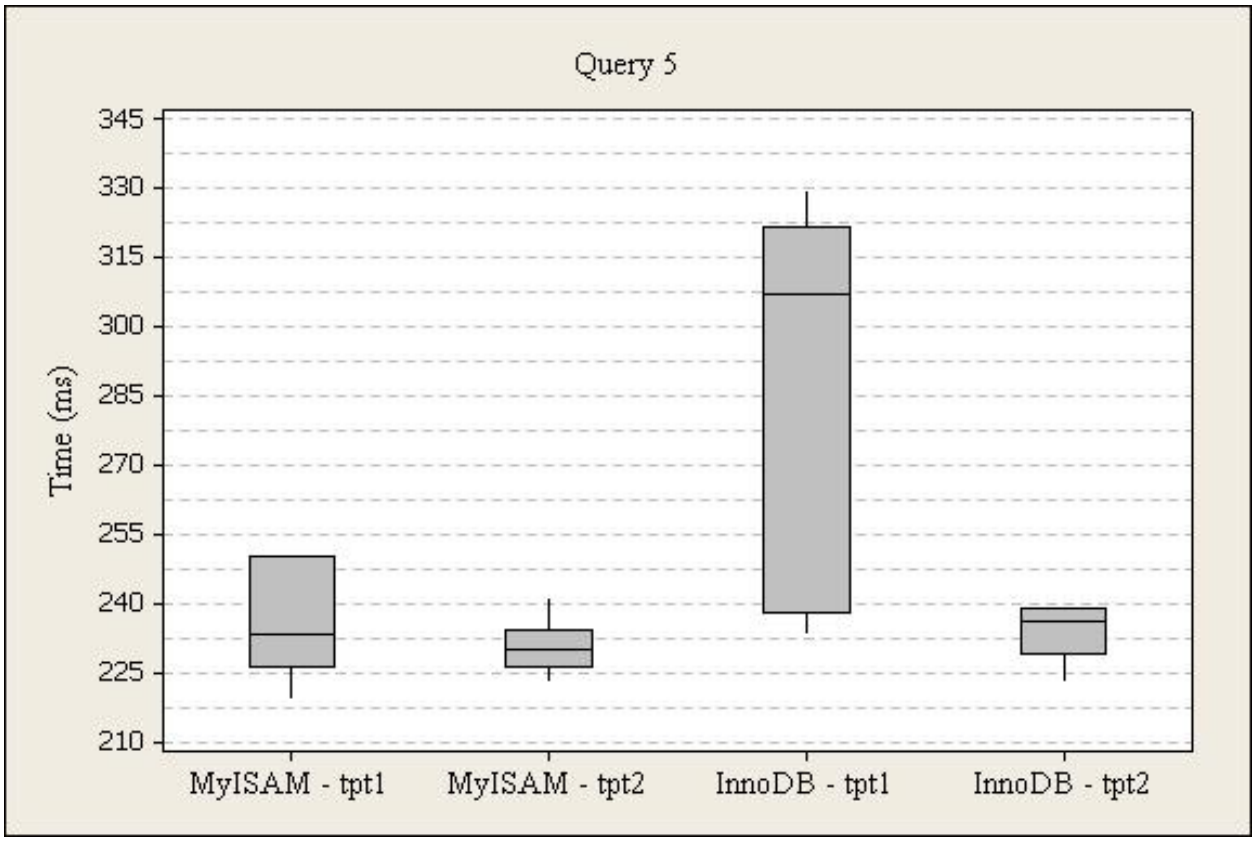


Figure 14. Power Test Query 5 Box Plot

The execution times for query five (total net loss from call centers managed by Larry McCray or Mark Hightower, and the item is in the home or books category from the catalog returns cube returned in 2002), run one were higher for InnoDB (M=288, SD=40) than MyISAM (M=251, SD=43). For InnoDB, the times range from 233ms to 330ms, and 238ms to 322ms from the first to third quartile with a 307ms median. For MyISAM, the times range from 219ms to 344ms, and 226ms to 250ms from the first to third quartile with a 233ms median.

The execution times for query five, run two were higher for InnoDB (M=276, SD=110) than MyISAM (M=230, SD=5). For InnoDB, the times range from 223ms to 614ms, and 229ms to 239ms from the first to third quartile with a 236ms median. For MyISAM, the times range from 223ms to 241ms, and 226ms to 234ms from the first to third quartile with a 230ms median.

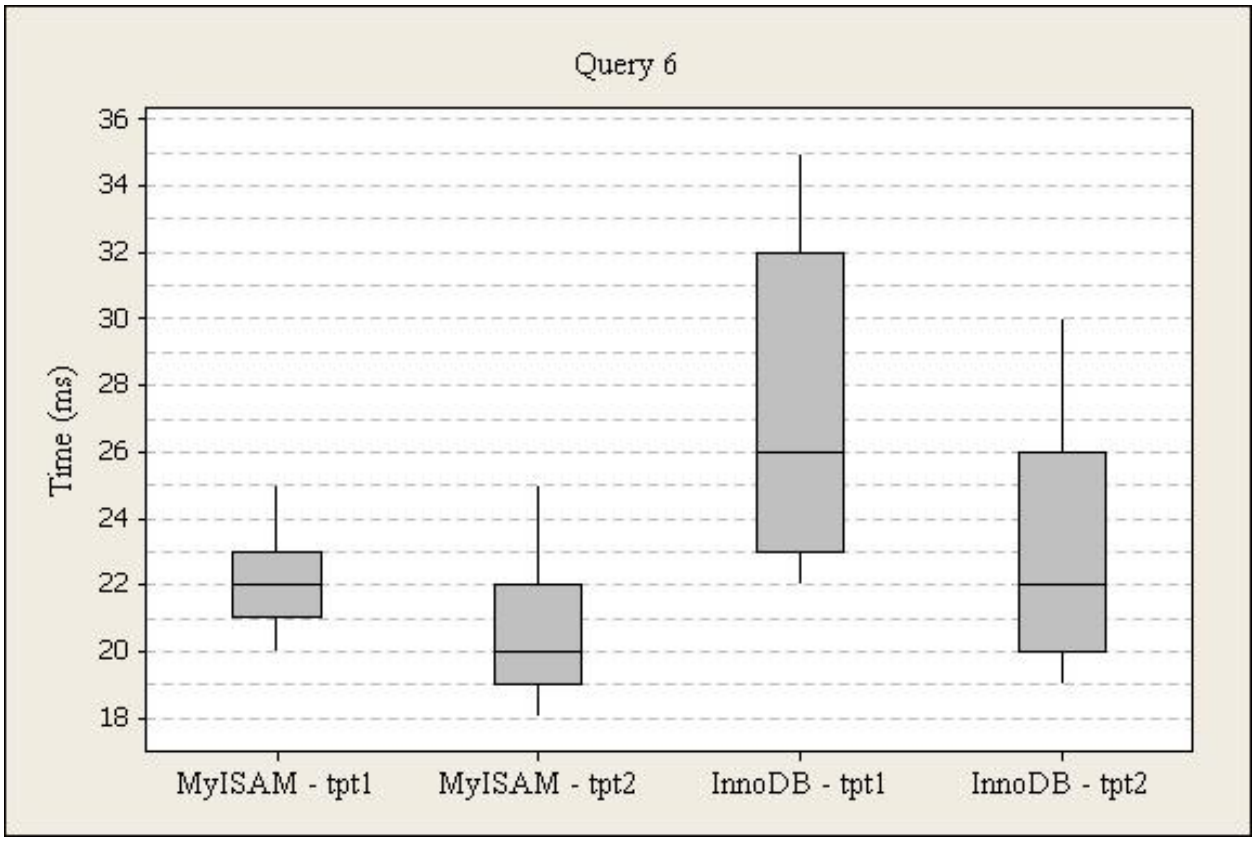


Figure 15. Power Test Query 6 Box Plot

The execution times for query six (quantity, item color, description, and info for four items from the inventory cube), run one were higher for InnoDB (M=27, SD=4) than MyISAM (M=22, SD=2). For InnoDB, the times range from 22ms to 35ms, and 23ms to 32ms from the first to third quartile with a 26ms median. For MyISAM, the times range from 21ms to 25ms, and 21ms to 23ms from the first to third quartile with a 22ms median.

The execution times for query six, run two were higher for InnoDB (M=23, SD=3) than MyISAM (M=20, SD=2). For InnoDB, the times range from 19ms to 30ms, and 20ms to 26ms from the first to third quartile with a 22ms median. For MyISAM, the times range from 18ms to 25ms, and 19ms to 22ms from the first to third quartile with a 20ms median.

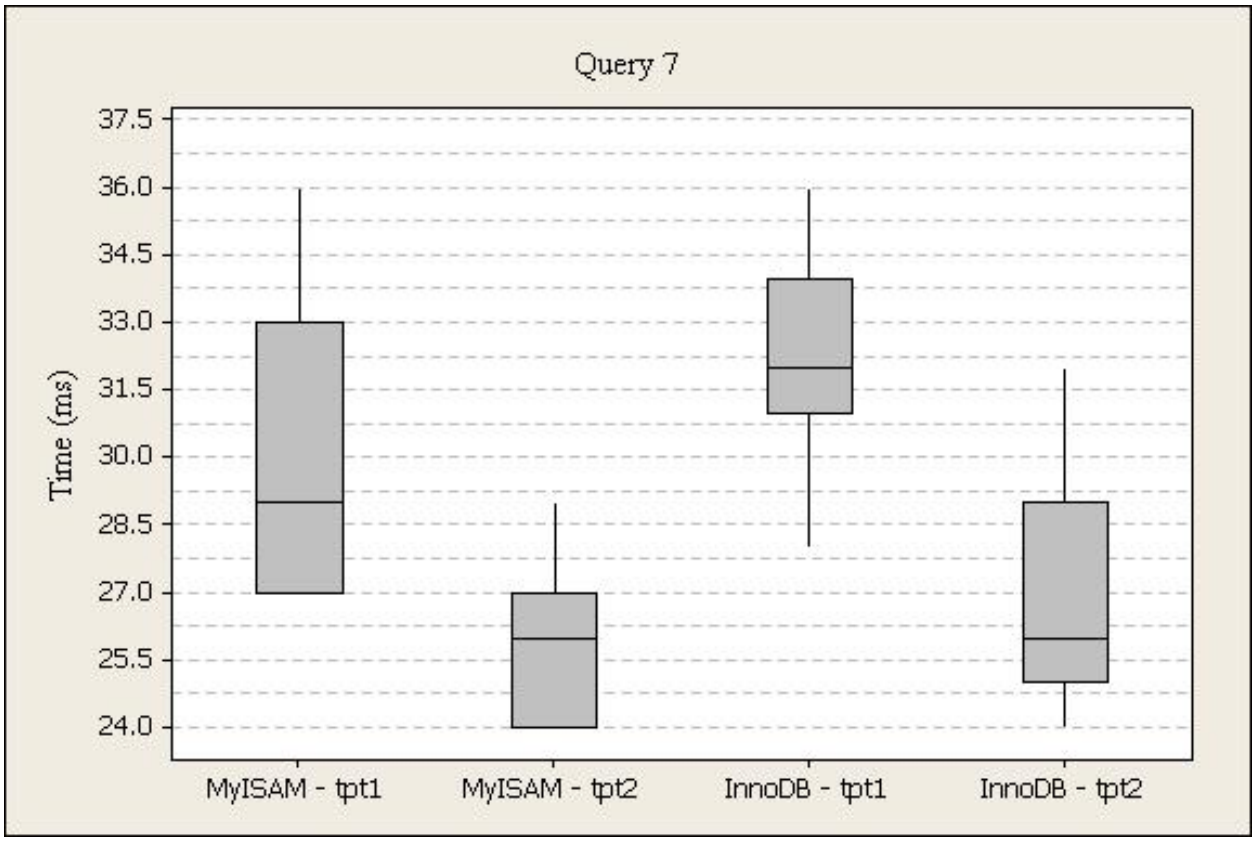


Figure 16. Power Test Query 7 Box Plot

The execution times for query seven (net loss, customer demographics, and date from the customer sales cube), run one were higher for InnoDB (M=32, SD=4) than MyISAM (M=30, SD=3). For InnoDB, the times range from 26ms to 41ms, and 31ms to 34ms from the first to third quartile with a 32ms median. For MyISAM, the times range from 27ms to 36ms, and 27ms to 33ms from the first to third quartile with a 29ms median.

The execution times for query seven, run two were lower for InnoDB (M=26, SD=3) than MyISAM (M=27, SD=6). For InnoDB, the times range from 24ms to 32ms, and 25ms to 29ms from the first to third quartile with a 26ms median. For MyISAM, the times range from 24ms to 47ms, and 24ms to 27ms from the first to third quartile with a 26ms median.

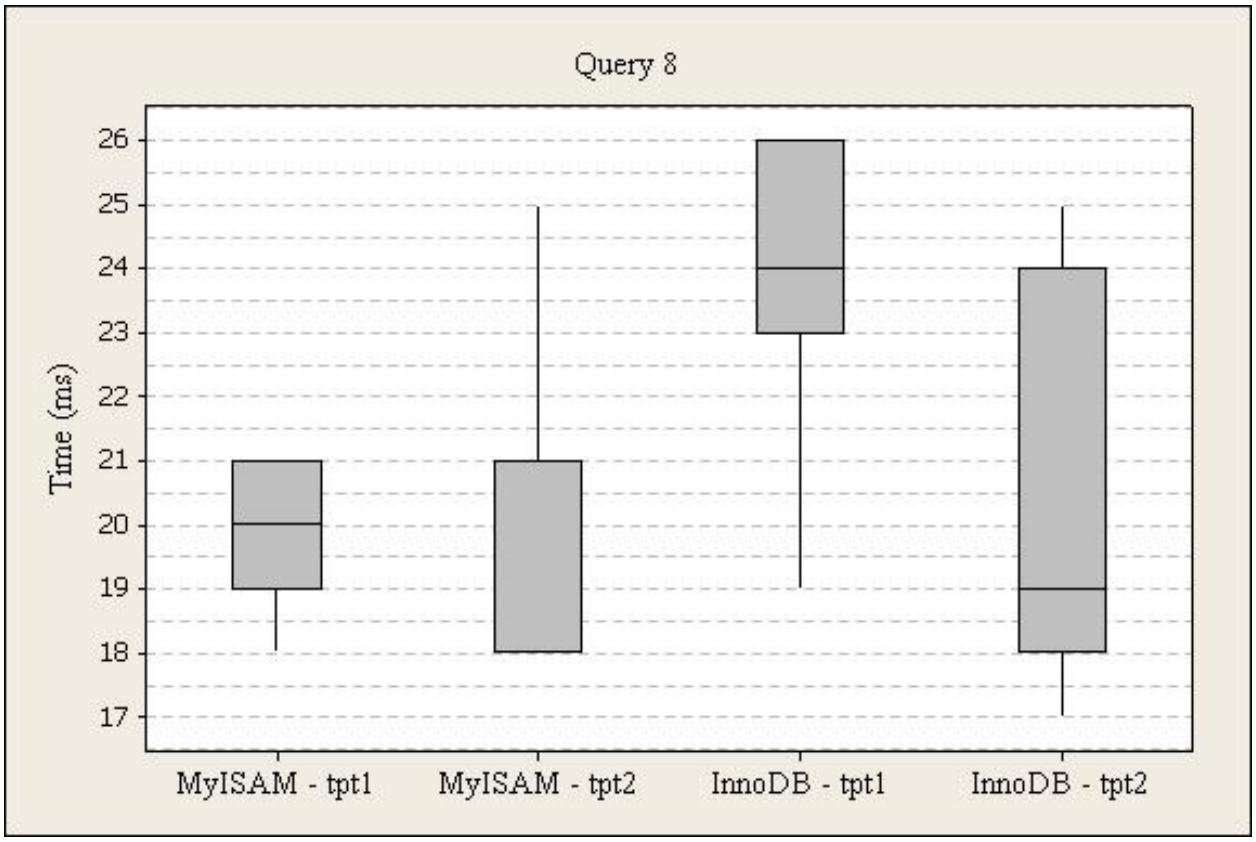


Figure 17. Power Test Query 8 Box Plot

The execution times for query eight (total quantity for the home and electronics categories from the catalog sales cube), run one were higher for InnoDB (M=24, SD=4) than MyISAM (M=21, SD=4). For InnoDB, the times range from 18ms to 32ms, and 23ms to 26ms from the first to third quartile with a 24ms median. For MyISAM, the times range from 18ms to 34ms, and 19ms to 21ms from the first to third quartile with a 20ms median.

The execution times for query eight, run two were higher for InnoDB (M=28, SD=29) than MyISAM (M=19, SD=3). For InnoDB, the times range from 17ms to 25ms, and 18ms to 24ms from the first to third quartile with a 19ms median. For MyISAM, the times range from 18ms to 26ms, and 18ms to 21ms from the first to third quartile with a 18ms median.

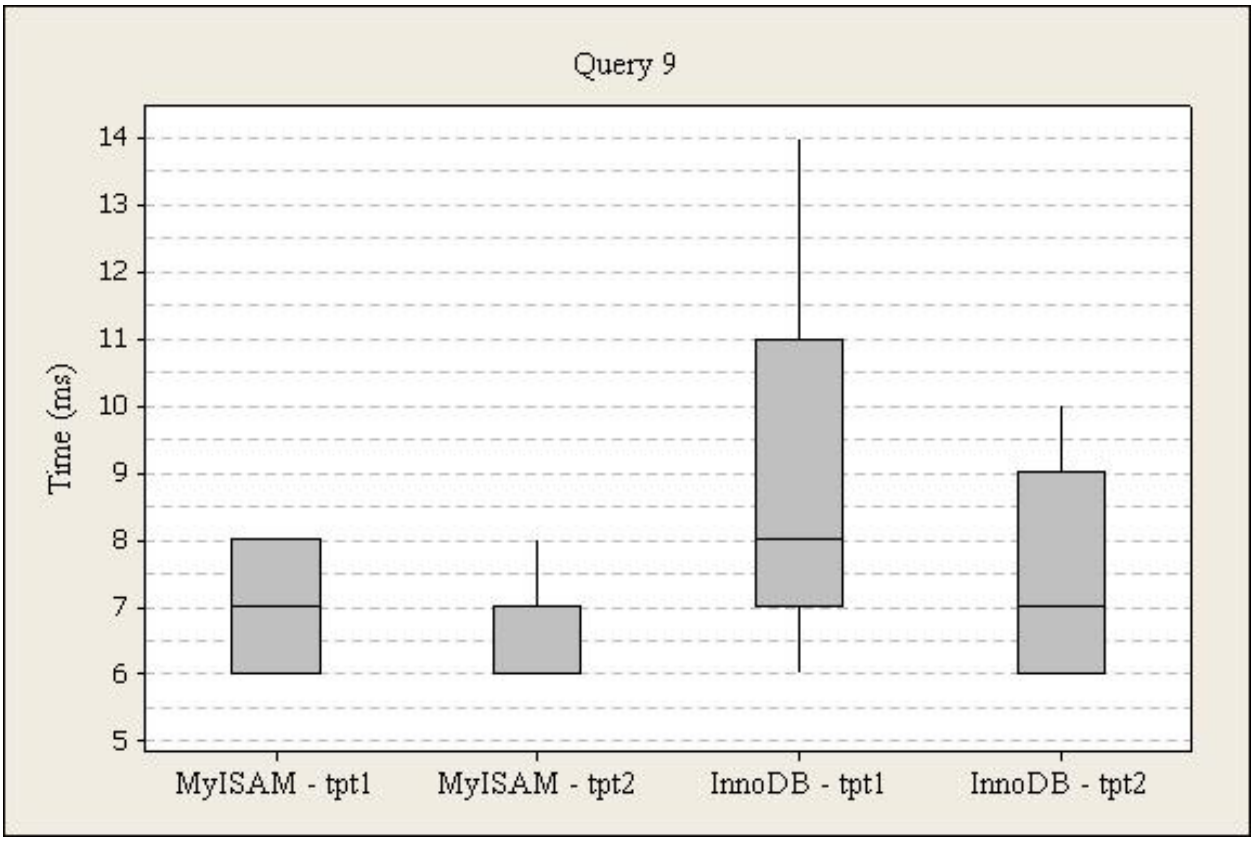


Figure 18. Power Test Query 9 Box Plot

The execution times for query nine (total quantity for the music and children categories from the catalog sales cube), run one were higher for InnoDB (M=9, SD=3) than MyISAM (M=7, SD=2). For InnoDB, the times range from 6ms to 14ms, and 7ms to 11ms from the first to third quartile with a 8ms median. For MyISAM, the times range from 6ms to 14ms, and 6ms to 8ms from the first to third quartile with a 7ms median.

The execution times for query nine, run two were higher for InnoDB (M=8, SD=5) than MyISAM (M=6, SD=1). For InnoDB, the times range from 6ms to 23ms, and 6ms to 9ms from the first to third quartile with a 7ms median. For MyISAM, the times range from 6ms to 8ms, and 6ms to 7ms from the first to third quartile with a 7ms median.

4.2.2 TPC-DS Throughput Test Statistics

Two TPC-DS Throughput Tests (ttt1 & ttt2) measured SUT performance when processing ninety nine queries via twenty concurrent user sessions. These queries were executed in random order [49]. Figures 19 – 27 present plots of the minimum, first quartile, median, third quartile, and maximum execution times for each throughput test executed used to calculate the TPC-DS performance metric, QphDS. These throughput test execution time distributions provide a comprehensive visual summary.

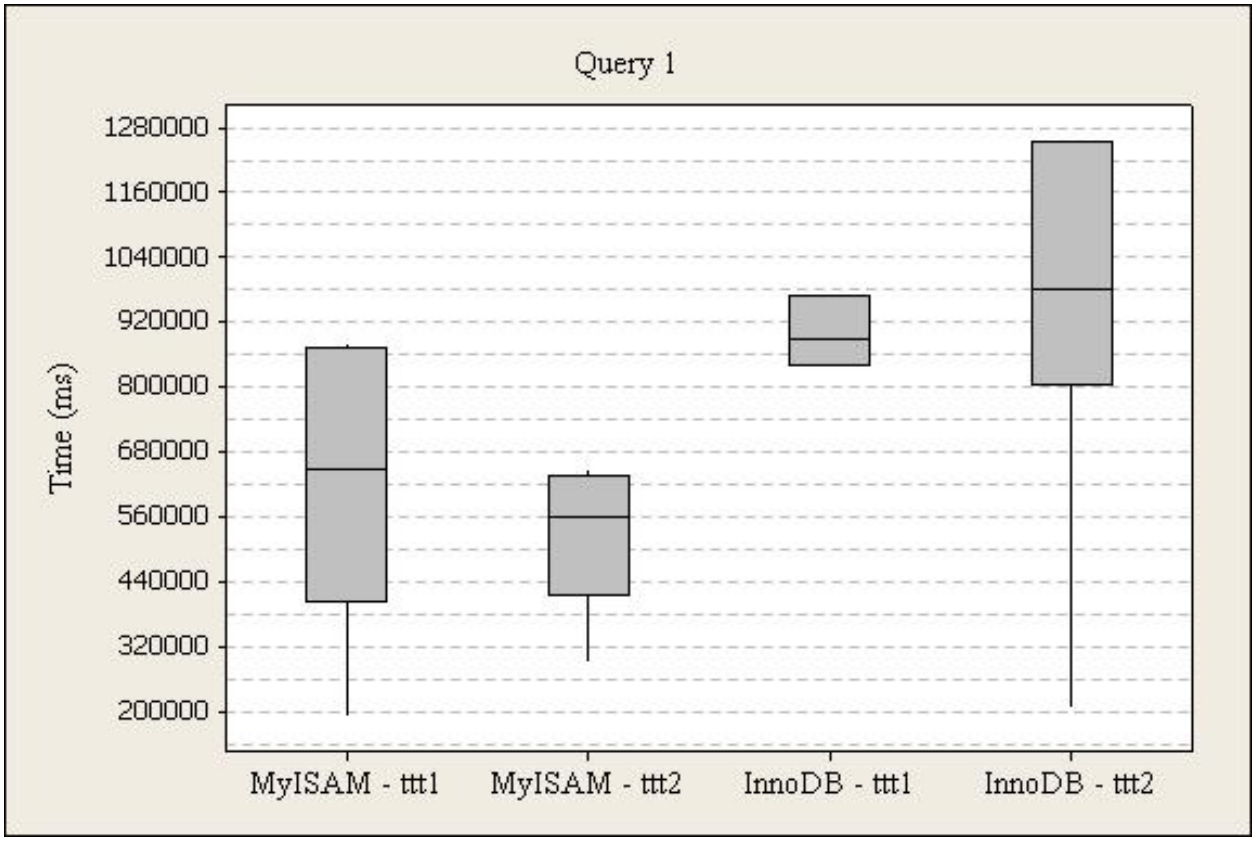


Figure 19. Throughput Test Query 1 Box Plot

The execution times for query one, run one were higher for InnoDB (M=826551, SD=197578) than MyISAM (M=600838, SD=236066). For InnoDB, the times range from 292444ms to 973888ms, and 841002ms to 968163ms from the first to third quartile with a

886710ms median. For MyISAM, the times range from 189528ms to 878470ms, and 401080ms to 876768ms from the first to third quartile with a 647326ms median.

The execution times for query one, run two were higher for InnoDB (M=939801, SD=293696) than MyISAM (M=515707, SD=127139). For InnoDB, the times range from 207036ms to 1257889ms, and 804597ms to 1252590ms from the first to third quartile with a 980922ms median. For MyISAM, the times range from 288268ms to 648286ms, and 416225ms to 636654ms from the first to third quartile with a 560387ms median.

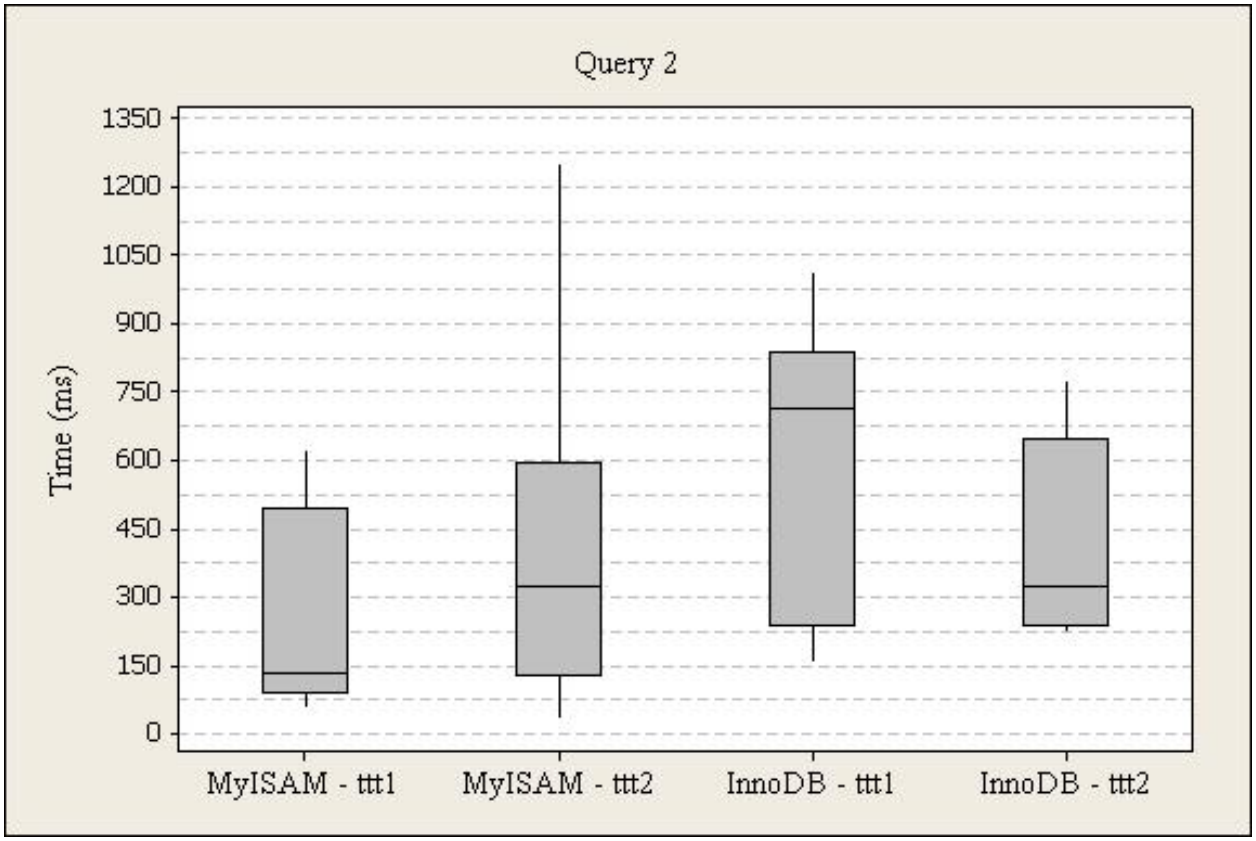


Figure 20. Throughput Test Query 2 Box Plot

The execution times for query two, run one were higher for InnoDB (M=635, SD=283) than MyISAM (M=253, SD=204). For InnoDB, the times range from 157ms to 1012ms, and 239ms to 835ms from the first to third quartile with a 711ms median. For MyISAM, the times

range from 55ms to 621ms, and 89ms to 493ms from the first to third quartile with a 130ms median.

The execution times for query two, run two were higher for InnoDB (M=709, SD=1068) than MyISAM (M=423, SD=336). For InnoDB, the times range from 224ms to 4044ms, and 236ms to 644ms from the first to third quartile with a 324ms median. For MyISAM, the times range from 34ms to 1252ms, and 126ms to 593ms from the first to third quartile with a 324ms median.

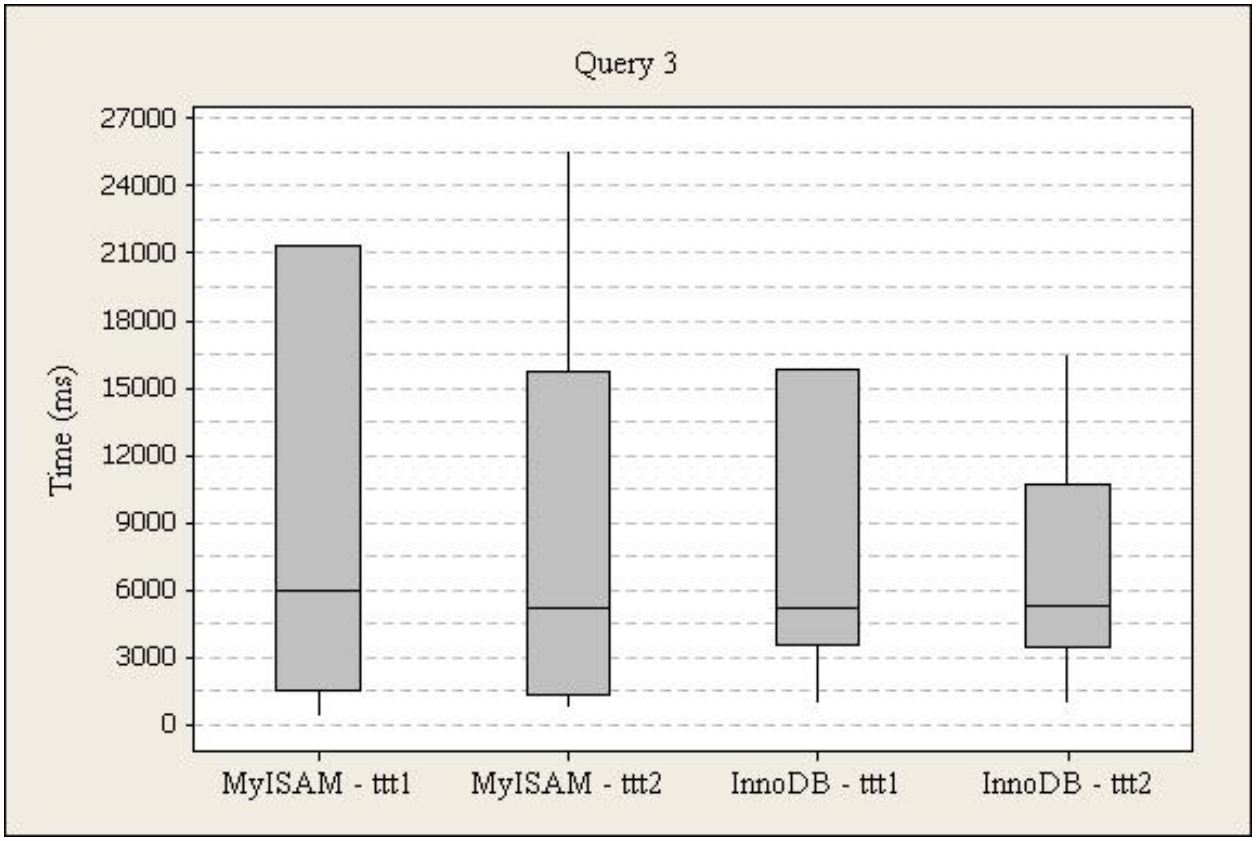


Figure 21. Throughput Test Query 3 Box Plot

The execution times for query three, run one were higher for InnoDB (M=42043, SD=80967) than MyISAM (M=28989, SD=48510). For InnoDB, the times range from 920ms to 258186ms, and 3568ms to 15774ms from the first to third quartile with a 5205ms median. For

MyISAM, the times range from 307ms to 149711ms, and 1501ms to 21302ms from the first to third quartile with a 5947ms median.

The execution times for query three, run two were higher for InnoDB (M=11019, SD=15108) than MyISAM (M=8020, SD=7514). For InnoDB, the times range from 882ms to 57011ms, and 3421ms to 10679ms from the first to third quartile with a 5254ms median. For MyISAM, the times range from 717ms to 25598ms, and 1277ms to 15663ms from the first to third quartile with a 5154ms median.

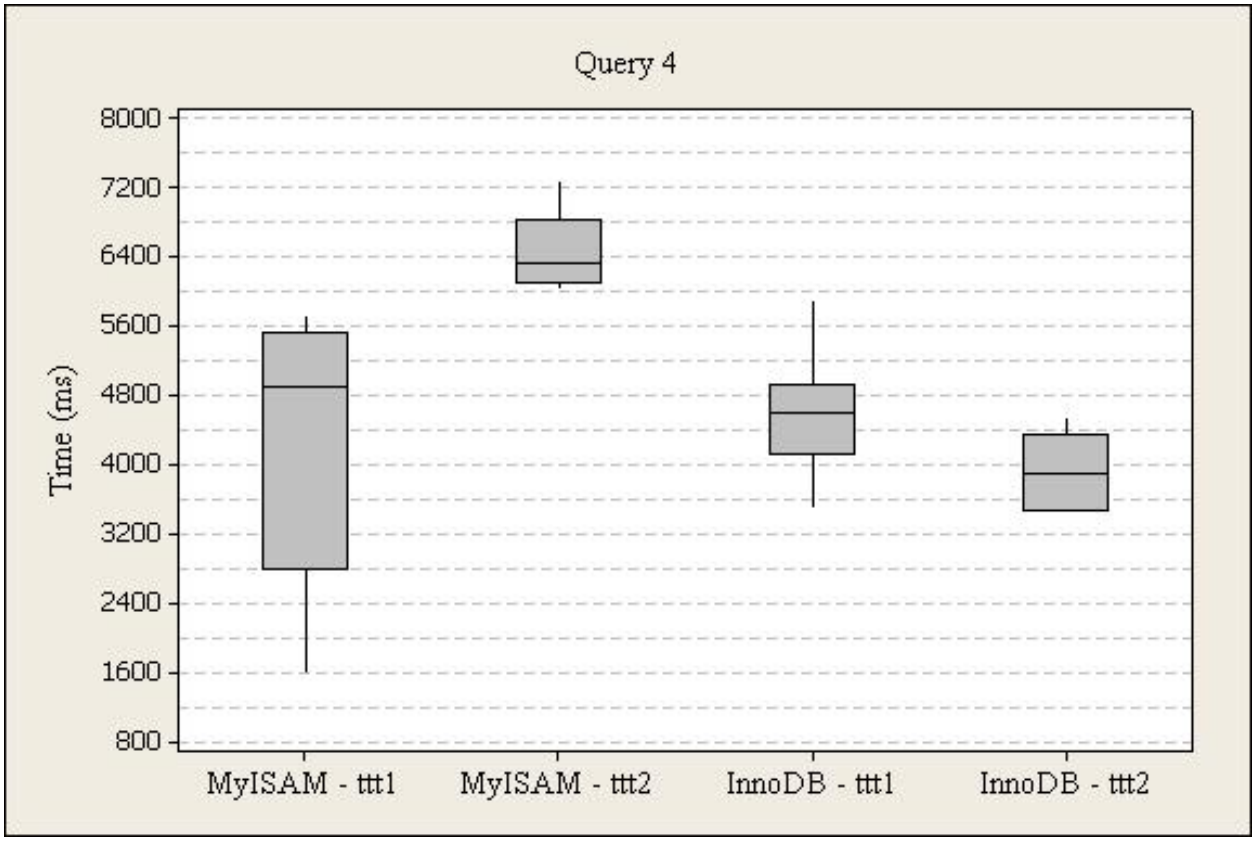


Figure 22. Throughput Test Query 4 Box Plot

The execution times for query four, run one were higher for InnoDB (M=4615, SD=619) than MyISAM (M=4277, SD=1446). For InnoDB, the times range from 3496ms to 5913ms, and 4130ms to 4935ms from the first to third quartile with a 4594ms median. For MyISAM, the

times range from 1582ms to 5733ms, and 2798ms to 5533ms from the first to third quartile with a 4895ms median.

The execution times for query four, run two were higher for InnoDB (M=3804, SD=748) than MyISAM (M=6256, SD=924). For InnoDB, the times range from 1774ms to 4552ms, and 3474ms to 4351ms from the first to third quartile with a 3908ms median. For MyISAM, the times range from 3581ms to 7297ms, and 6113ms to 6835ms from the first to third quartile with a 6330ms median.

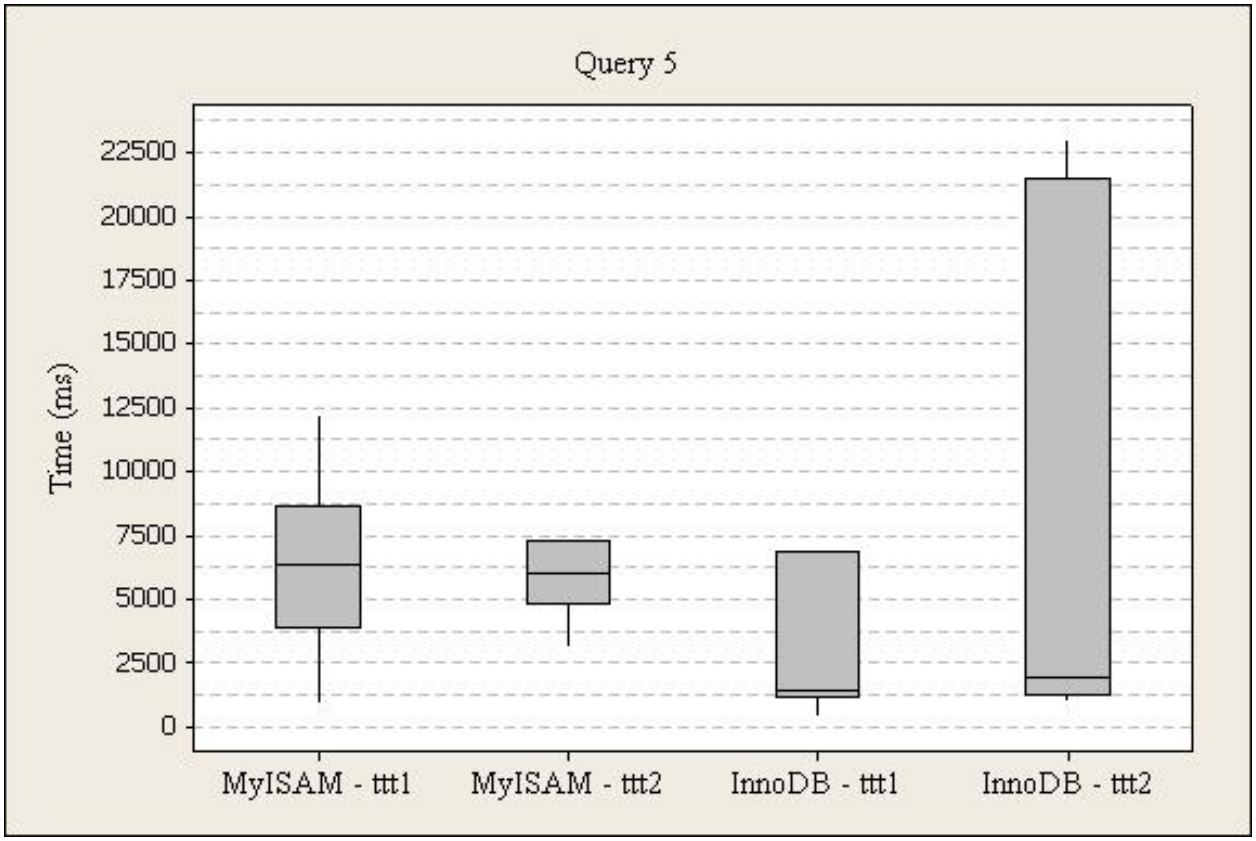


Figure 23. Throughput Test Query 5 Box Plot

The execution times for query five, run one were higher for InnoDB (M=14915, SD=27988) than MyISAM (M=7292, SD=5799). For InnoDB, the times range from 374ms to 91342ms, and 1101ms to 6815ms from the first to third quartile with a 1393ms median. For

MyISAM, the times range from 911ms to 22880ms, and 3846ms to 8667ms from the first to third quartile with a 6339ms median.

The execution times for query five, run two were higher for InnoDB (M=43365, SD=116414) than MyISAM (M=6896, SD=4131). For InnoDB, the times range from 982ms to 410681ms, and 1227ms to 21469ms from the first to third quartile with a 1946ms median. For MyISAM, the times range from 954ms to 16805ms, and 4840ms to 7274ms from the first to third quartile with a 5957ms median.

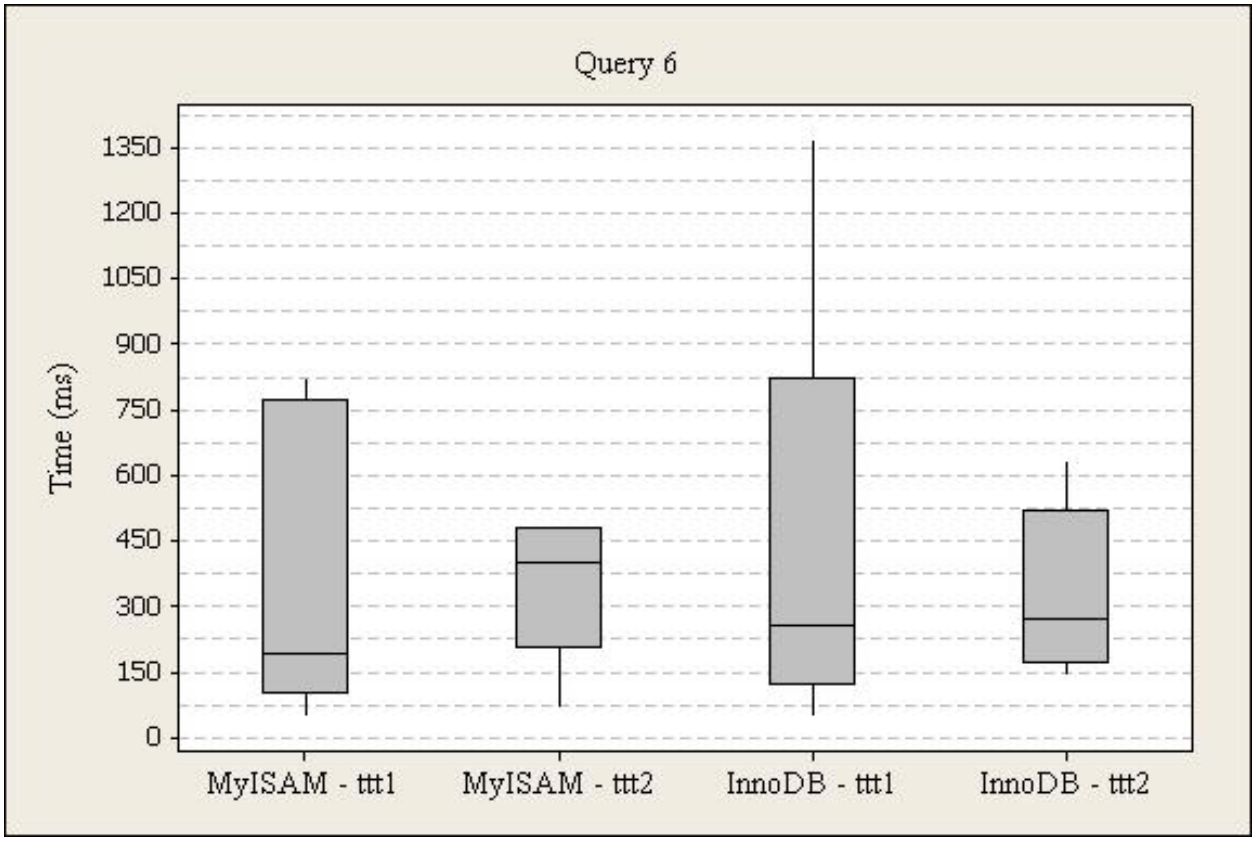


Figure 24. Throughput Test Query 6 Box Plot

The execution times for query six, run one were lower for InnoDB (M=474, SD=421) than MyISAM (M=474, SD=505). For InnoDB, the times range from 49ms to 1368ms, and 122ms to 821ms from the first to third quartile with a 254ms median. For MyISAM, the times

range from 47ms to 1821ms, and 101ms to 771ms from the first to third quartile with a 191ms median.

The execution times for query six, run two were lower for InnoDB (M=315, SD=168) than MyISAM (M=451, SD=307). For InnoDB, the times range from 142ms to 634ms, and 171ms to 518ms from the first to third quartile with a 273ms median. For MyISAM, the times range from 65ms to 1098ms, and 206ms to 480ms from the first to third quartile with a 398ms median.

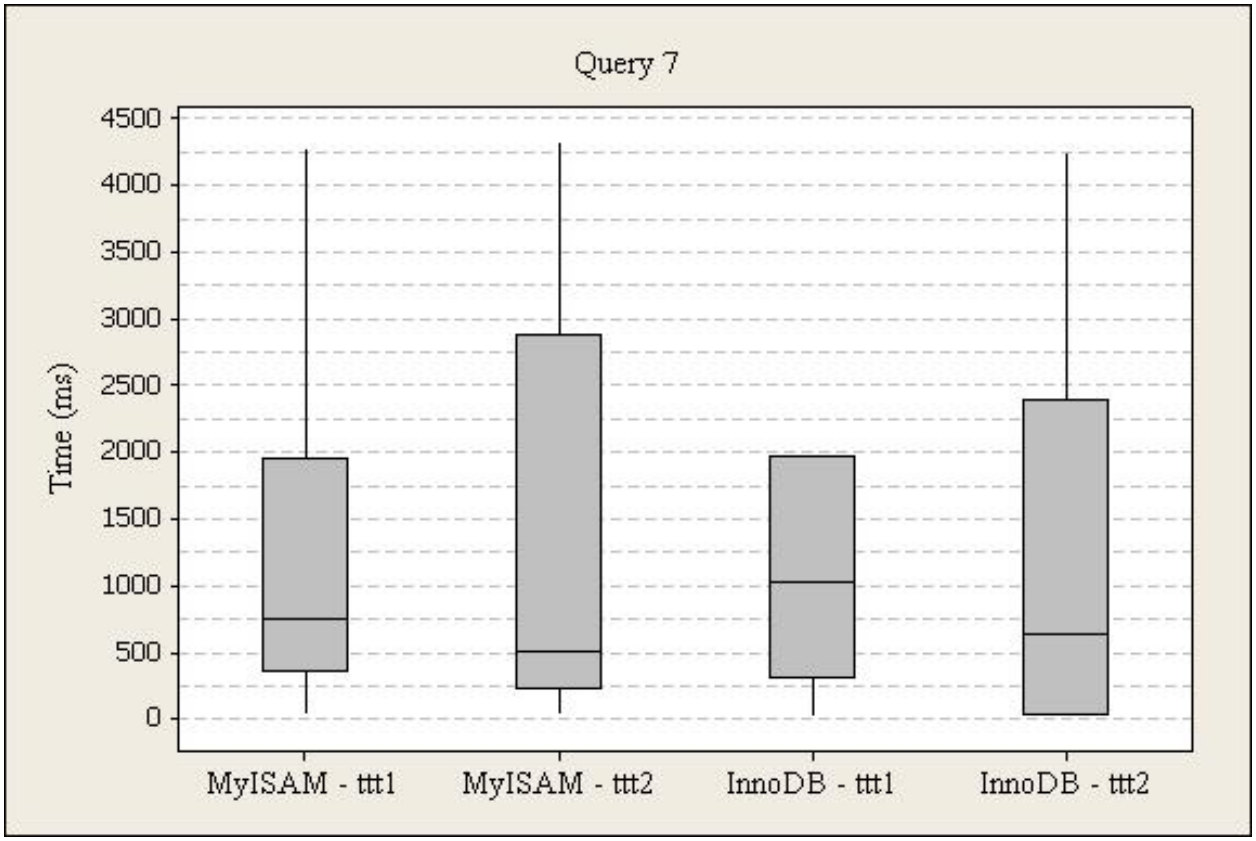


Figure 25. Throughput Test Query 7 Box Plot

The execution times for query seven, run one were lower for InnoDB (M=1995, SD=2515) than MyISAM (M=2267, SD=3808). For InnoDB, the times range from 15ms to 8304ms, and 302ms to 1973ms from the first to third quartile with a 1024ms median. For

MyISAM, the times range from 37ms to 13738ms, and 365ms to 1956ms from the first to third quartile with a 755ms median.

The execution times for query seven, run two were lower for InnoDB (M=1228, SD=1491) than MyISAM (M=1485, SD=1590). For InnoDB, the times range from 26ms to 4252ms, and 34ms to 2387ms from the first to third quartile with a 628ms median. For MyISAM, the times range from 31ms to 4334ms, and 233ms to 2885ms from the first to third quartile with a 512ms median.

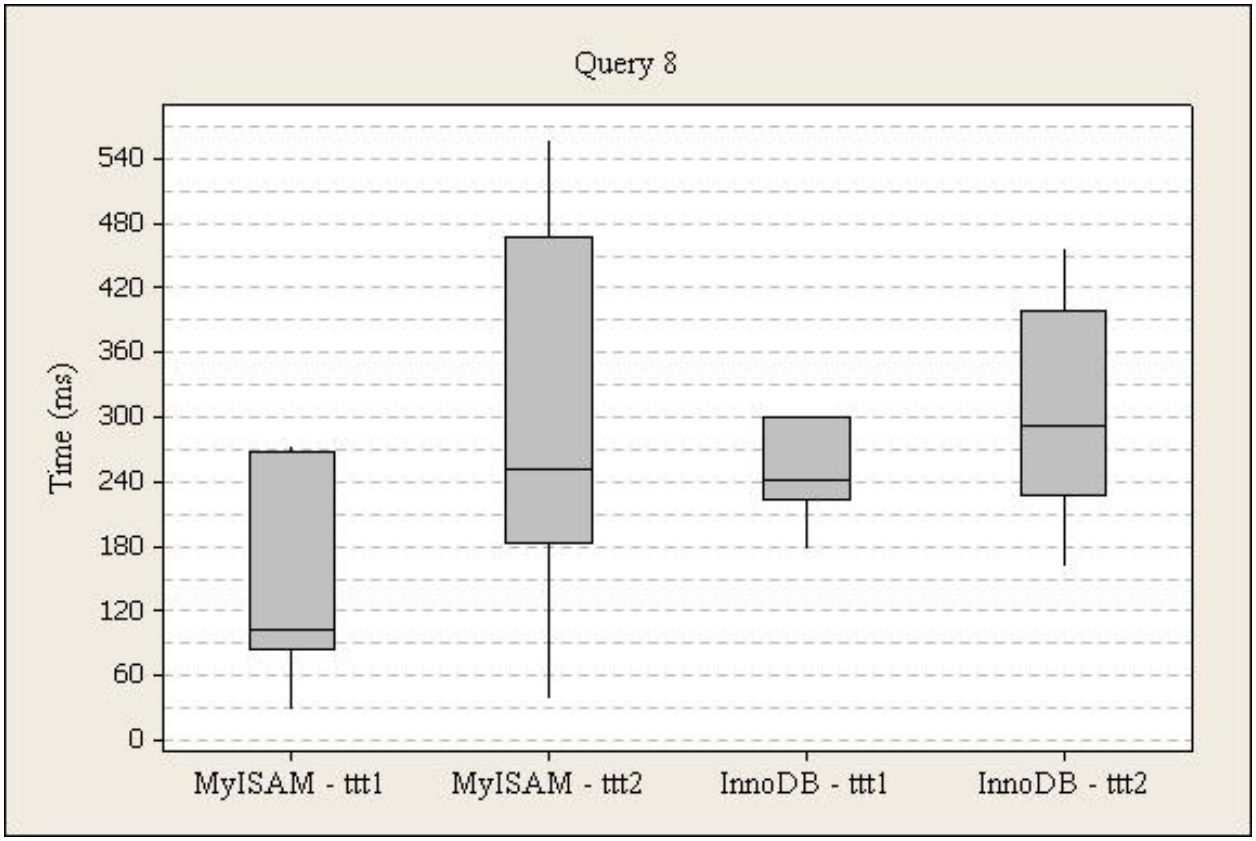


Figure 26. Throughput Test Query 8 Box Plot

The execution times for query eight, run one were higher for InnoDB (M=337, SD=215) than MyISAM (M=174, SD=174). For InnoDB, the times range from 176ms to 882ms, and 224ms to 299ms from the first to third quartile with a 241ms median. For MyISAM, the times

range from 28ms to 674ms, and 85ms to 368ms from the first to third quartile with a 103ms median.

The execution times for query eight, run two were higher for InnoDB (M=345, SD=183) than MyISAM (M=297, SD=157). For InnoDB, the times range from 161ms to 886ms, and 227ms to 399ms from the first to third quartile with a 292ms median. For MyISAM, the times range from 38ms to 558ms, and 182ms to 466ms from the first to third quartile with a 252ms median.

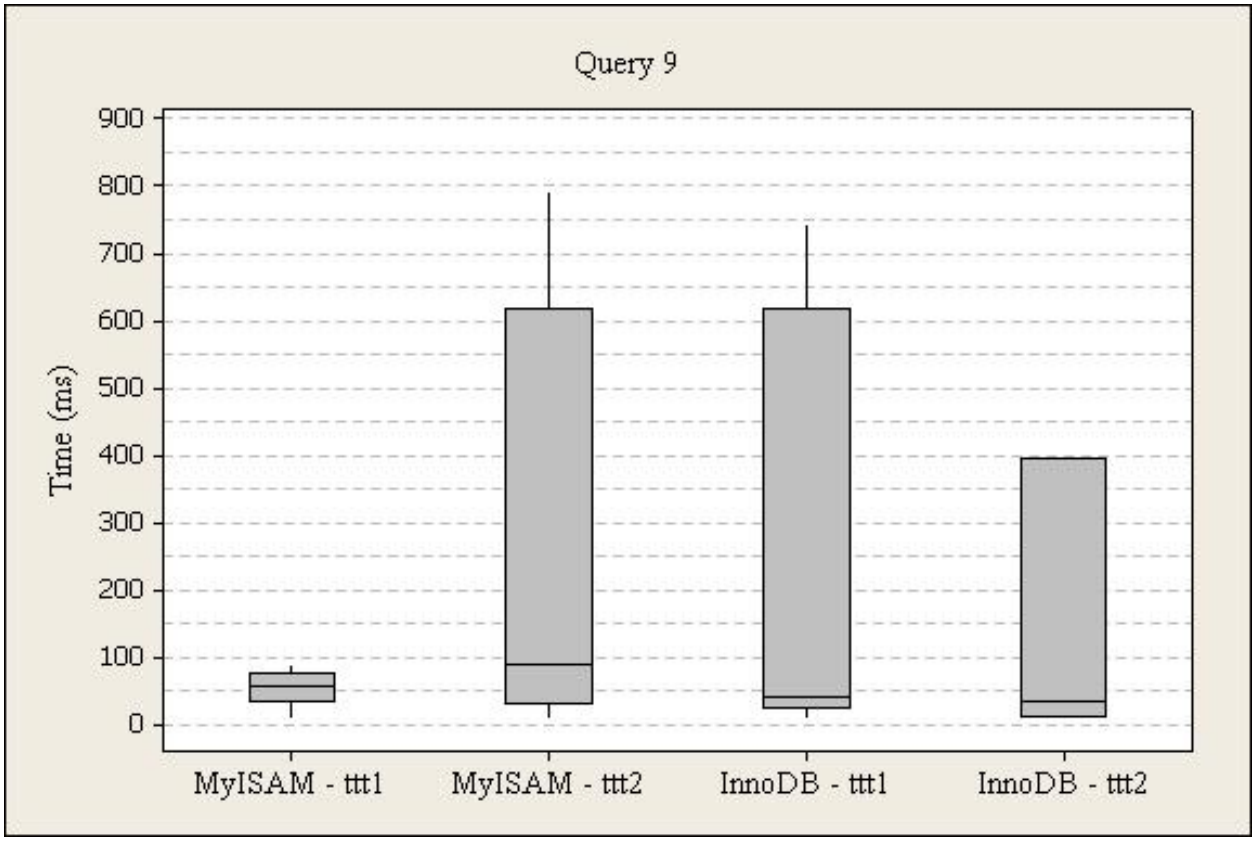


Figure 27. Throughput Test Query 9 Box Plot

The execution times for query nine, run one were higher for InnoDB (M=9443, SD=29168) than MyISAM (M=114, SD=206). For InnoDB, the times range from 7ms to 101676ms, and 25ms to 620ms from the first to third quartile with a 39ms median. For

MyISAM, the times range from 7ms to 760ms, and 33ms to 77ms from the first to third quartile with a 58ms median.

The execution times for query nine, run two were higher for InnoDB (M=406, SD=705) than MyISAM (M=274, SD=295). For InnoDB, the times range from 8ms to 2004ms, and 13ms to 396ms from the first to third quartile with a 35ms median. For MyISAM, the times range from 9ms to 794ms, and 31ms to 619ms from the first to third quartile with a 89ms median.

4.3 Scaled Data Size Summary

In addition to the TPC-DS benchmark, two additional throughput tests were performed to show how the execution times scale in proportion to data set size. Fact table data was scaled to 9GB and 6GB, and compared to the existing 12GB first run throughput test statistics. The 12GB was first scaled to 9GB, a throughout test was executed to prefill the MySQL query cache, and a subsequent run was recorded. The 9GB was then scaled to 6GB, a throughout test was executed to prefill the MySQL query cache, and a subsequent run was recorded. Only queries related to the fact tables were analyzed. Figures 28 – 33 present plots of the minimum, first quartile, median, third quartile, and maximum execution times for each test query executed on the scaled data sets.

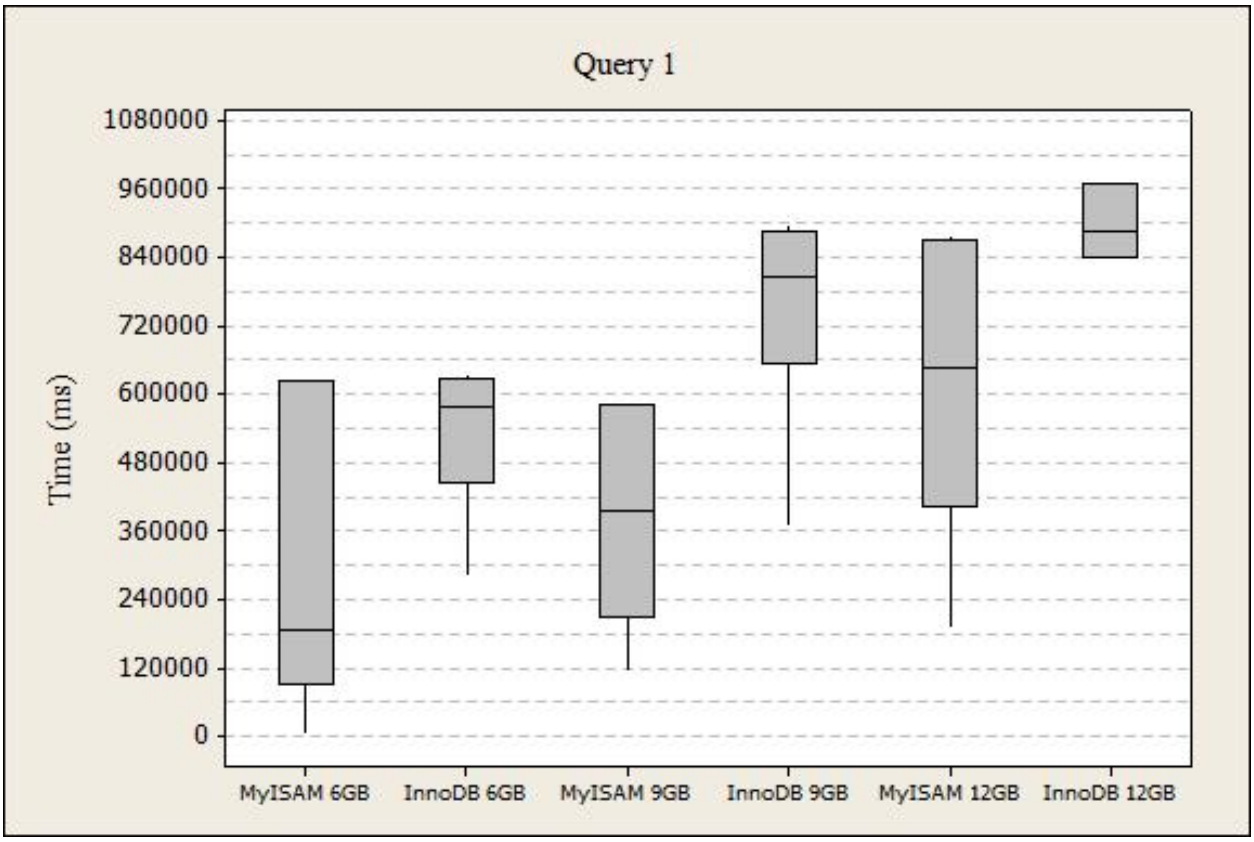


Figure 28. Throughput Test Query 1 Box Plot

The query one execution times using the 6GB test were higher for InnoDB (M=531905, SD=118641) than MyISAM (M=300656, SD=260313). For InnoDB, the times range from 282068ms to 634754ms, and 444617ms to 628323ms from the first to third quartile with a 577662ms median. For MyISAM, the times range from 4099ms to 624978ms, and 90733ms to 623039ms from the first to third quartile with a 186067ms median.

The query one execution times using the 9GB test were higher for InnoDB (M=715227, SD=253571) than MyISAM (M=396987, SD=182500). For InnoDB, the times range from 106915ms to 896923ms, and 654921ms to 887022ms from the first to third quartile with a 807185ms median. For MyISAM, the times range from 114307ms to 582845ms, and 207363ms to 580093ms from the first to third quartile with a 394558ms median.

The query one execution times using the 12GB test were higher for InnoDB (M=826551, SD=207221) than MyISAM (M=600838, SD=247588). For InnoDB, the times range from 292444ms to 973888ms, and 814002ms to 968163ms from the first to third quartile with a 886710ms median. For MyISAM, the times range from 189528ms to 582845ms, and 401080ms to 870768ms from the first to third quartile with a 647326ms median.

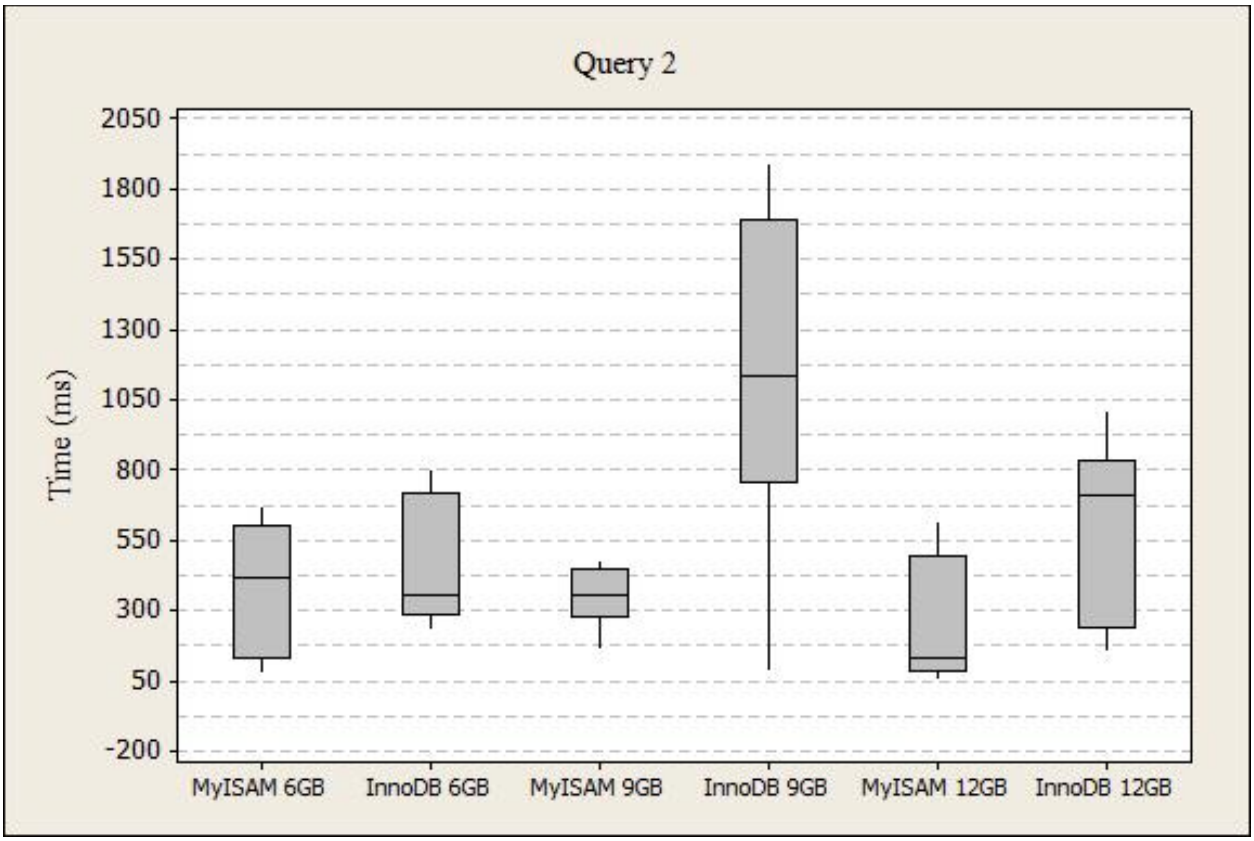


Figure 29. Throughput Test Query 2 Box Plot

The query two execution times using the 6GB test were higher for InnoDB (M=457, SD=212) than MyISAM (M=388, SD=217). For InnoDB, the times range from 233ms to 807ms, and 291ms to 722ms from the first to third quartile with a 359ms median. For MyISAM, the times range from 81ms to 676ms, and 136ms to 603ms from the first to third quartile with a 419ms median.

The query two execution times using the 9GB test were higher for InnoDB (M=1070, SD=612) than MyISAM (M=449, SD=352). For InnoDB, the times range from 86ms to 1895ms, and 759ms to 1695ms from the first to third quartile with a 1139ms median. For MyISAM, the times range from 168ms to 1475ms, and 281ms to 451ms from the first to third quartile with a 357ms median.

The query two execution times using the 12GB test were higher for InnoDB (M=636, SD=297) than MyISAM (M=253, SD=214). For InnoDB, the times range from 157ms to 1012ms, and 239ms to 835ms from the first to third quartile with a 711ms median. For MyISAM, the times range from 55ms to 621ms, and 89ms to 493ms from the first to third quartile with a 130ms median.

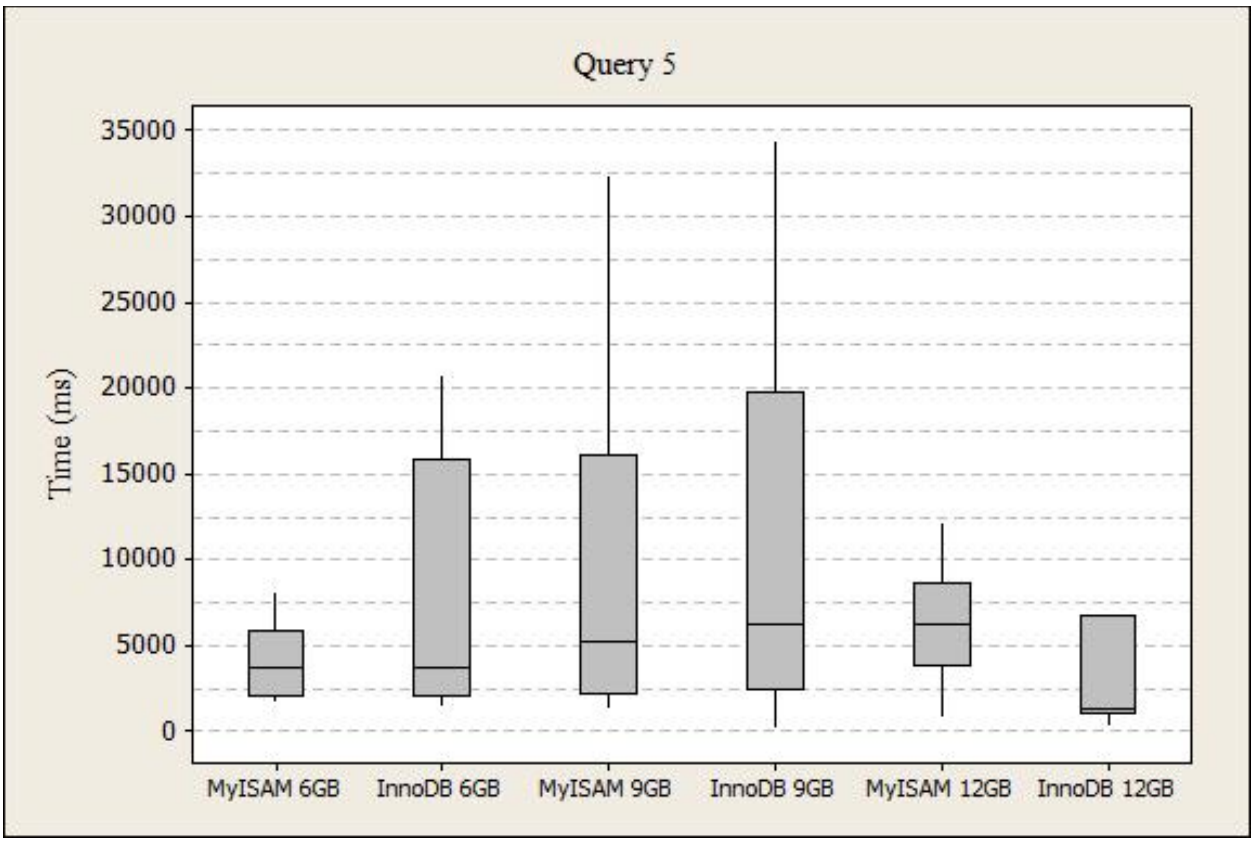


Figure 30. Throughput Test Query 5 Box Plot

The query five execution times using the 6GB test were higher for InnoDB (M=8512, SD=7822) than MyISAM (M=3898, SD=2036). For InnoDB, the times range from 1848ms to 20730ms, and 2138ms to 15816ms from the first to third quartile with a 3748ms median. For MyISAM, the times range from 1693ms to 8140ms, and 2080ms to 5929ms from the first to third quartile with a 3746ms median.

The query five execution times using the 9GB test were higher for InnoDB (M=29549, SD=67973) than MyISAM (M=9046, SD=9490). For InnoDB, the times range from 251ms to 232323ms, and 2478ms to 19784ms from the first to third quartile with a 6241ms median. For MyISAM, the times range from 1339ms to 32375ms, and 2185ms to 16095ms from the first to third quartile with a 5126ms median.

The query five execution times using the 12GB test were higher for InnoDB (M=14916, SD=29354) than MyISAM (M=7293, SD=6082). For InnoDB, the times range from 374ms to 91342ms, and 1101ms to 6815ms from the first to third quartile with a 1383ms median. For MyISAM, the times range from 911ms to 22880ms, and 3846ms to 8667ms from the first to third quartile with a 6339ms median.

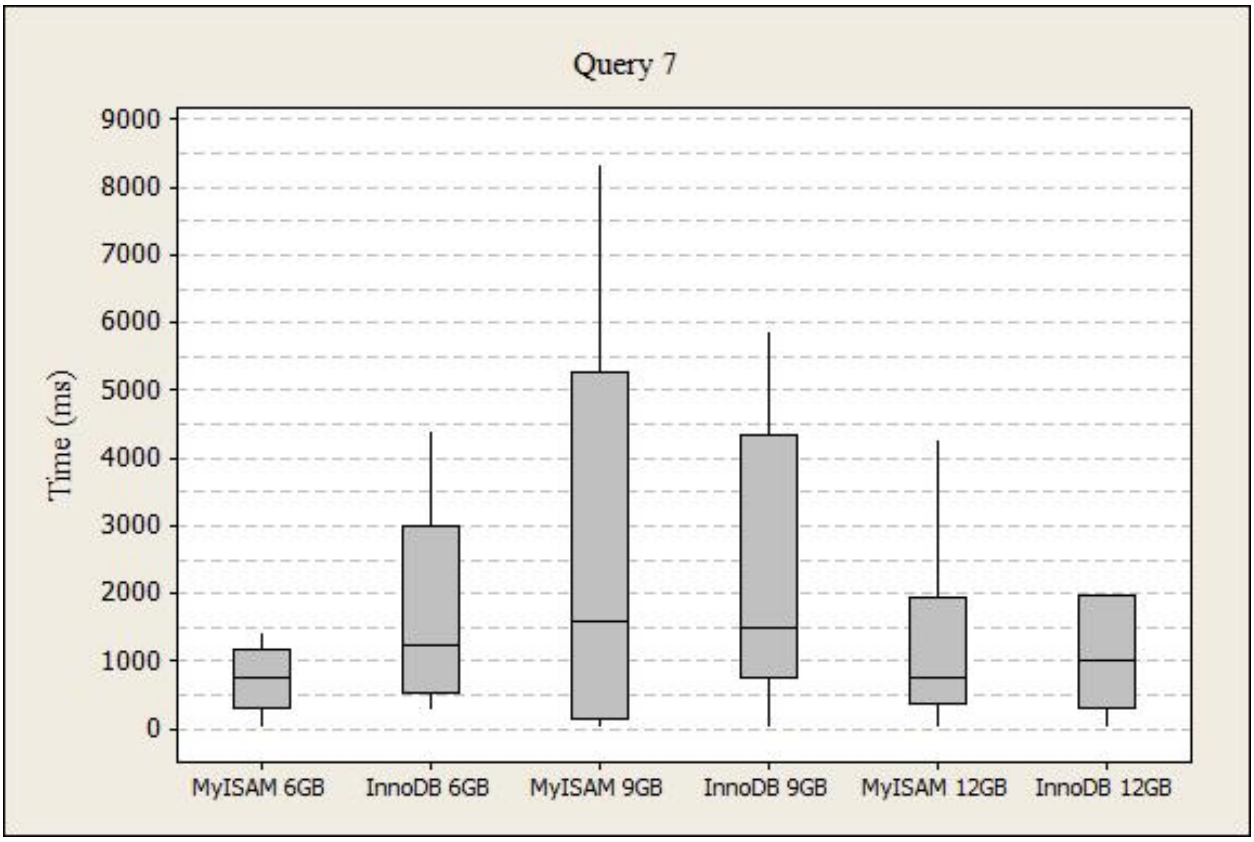


Figure 31. Throughput Test Query 7 Box Plot

The query seven execution times using the 6GB test were higher for InnoDB (M=1760, SD=1539) than MyISAM (M=875, SD=715). For InnoDB, the times range from 275ms to 4413ms, and 524ms to 2898ms from the first to third quartile with a 1244ms median. For MyISAM, the times range from 43ms to 2559ms, and 317ms to 1191ms from the first to third quartile with a 766ms median.

The query seven execution times using the 9GB test were lower for InnoDB (M=1996, SD=2003) than MyISAM (M=2683, SD=3053). For InnoDB, the times range from 27ms to 5880ms, and 772ms to 4352ms from the first to third quartile with a 1490ms median. For MyISAM, the times range from 35ms to 8328ms, and 164ms to 5264ms from the first to third quartile with a 1064ms median.

The query seven execution times using the 12GB test were lower for InnoDB (M=1996, SD=2637) than MyISAM (M=2267, SD=3994). For InnoDB, the times range from 15ms to 8304ms, and 302ms to 1937ms from the first to third quartile with a 1024ms median. For MyISAM, the times range from 37ms to 13738ms, and 365ms to 1956ms from the first to third quartile with a 755ms median.

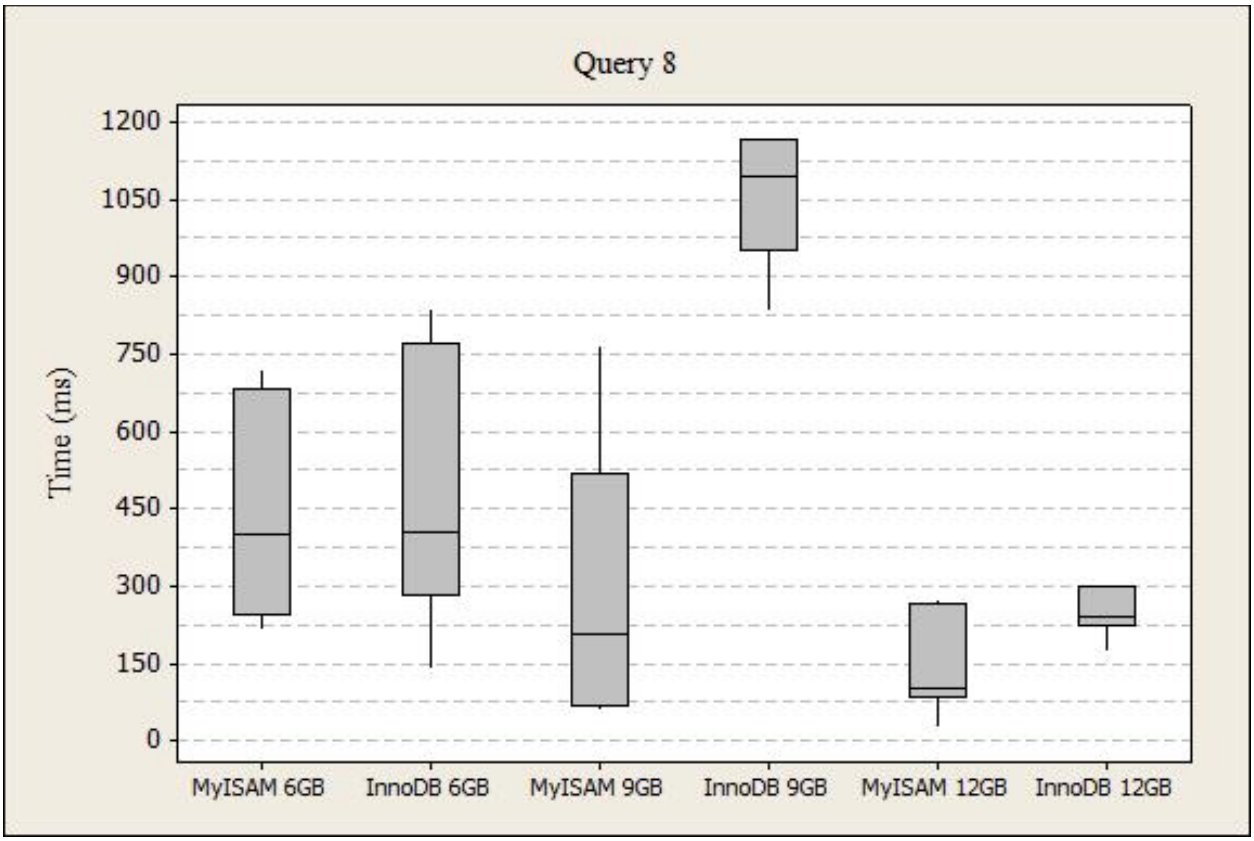


Figure 32. Throughput Test Query 8 Box Plot

The query eight execution times using the 6GB test were higher for InnoDB (M=480, SD=245) than MyISAM (M=431, SD=198). For InnoDB, the times range from 140ms to 837ms, and 284ms to 771ms from the first to third quartile with a 406ms median. For MyISAM, the times range from 217ms to 722ms, and 245ms to 682ms from the first to third quartile with a 404ms median.

The query eight execution times using the 9GB test were higher for InnoDB (M=1200, SD=423) than MyISAM (M=276, SD=248). For InnoDB, the times range from 434ms to 2218ms, and 952ms to 1165ms from the first to third quartile with a 1095ms median. For MyISAM, the times range from 62ms to 767ms, and 67ms to 517ms from the first to third quartile with a 209ms median.

The query eight execution times using the 12GB test were higher for InnoDB (M=338, SD=226) than MyISAM (M=175, SD=183). For InnoDB, the times range from 176ms to 882ms, and 224ms to 299ms from the first to third quartile with a 241ms median. For MyISAM, the times range from 28ms to 674ms, and 85ms to 268ms from the first to third quartile with a 103ms median.

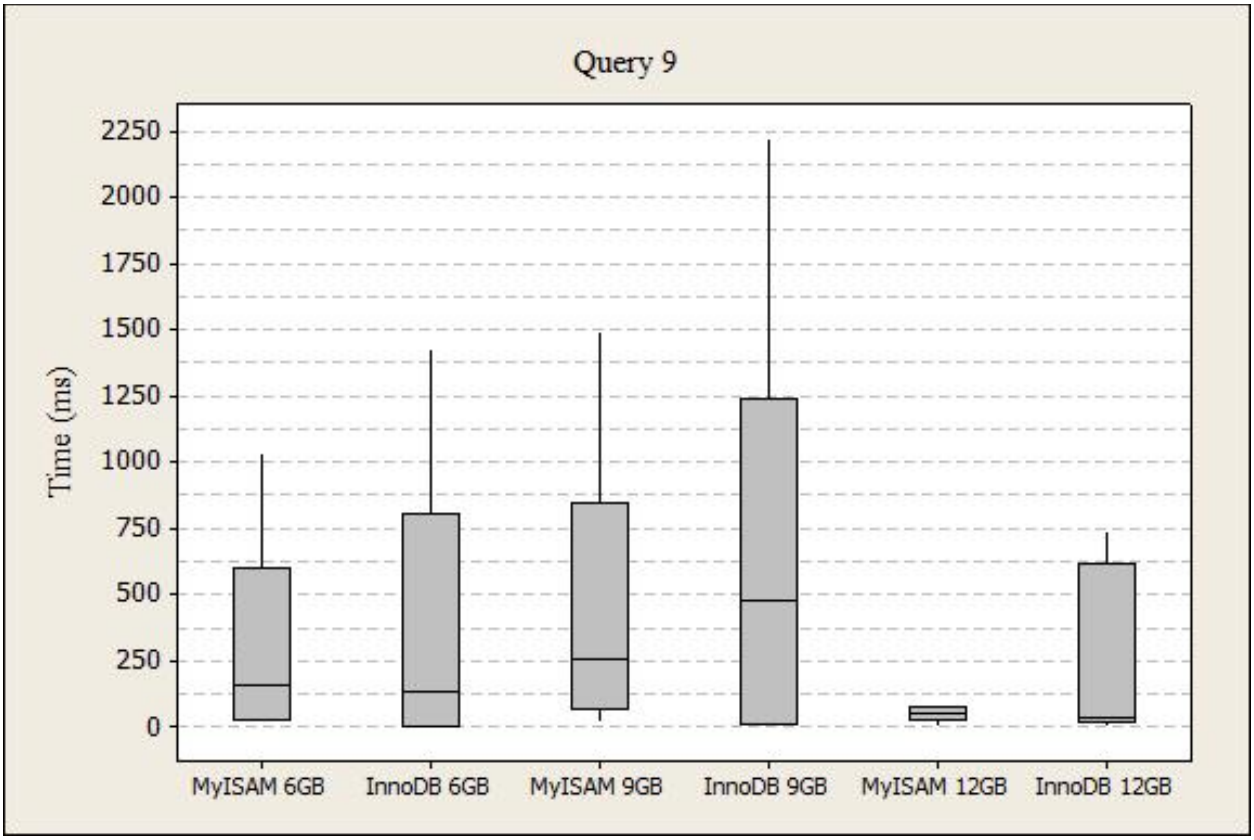


Figure 33. Throughput Test Query 9 Box Plot

The query nine execution times using the 6GB test were lower for InnoDB (M=372, SD=474) than MyISAM (M=498, SD=699). For InnoDB, the times range from 7ms to 1427ms, and 8ms to 807ms from the first to third quartile with a 140ms median. For MyISAM, the times range from 23ms to 2367ms, and 32ms to 605ms from the first to third quartile with a 160ms median.

The query nine execution times using the 9GB test were higher for InnoDB (M=634, SD=772) than MyISAM (M=574, SD=761). For InnoDB, the times range from 10ms to 2219s, and 15ms to 1238ms from the first to third quartile with a 476ms median. For MyISAM, the times range from 22ms to 2434ms, and 68ms to 849ms from the first to third quartile with a 255ms median.

The query nine execution times using the 12GB test were higher for InnoDB (M=9444, SD=3051) than MyISAM (M=114, SD=216). For InnoDB, the times range from 7ms to 101676ms, and 25ms to 620ms from the first to third quartile with a 39ms median. For MyISAM, the times range from 7ms to 760ms, and 33ms to 77ms from the first to third quartile with a 58ms median.

CHAPTER 5

ANALYSIS

5.1 Qualitative Analysis

Both MyISAM and InnoDB are released under the GPLv2 open source license. This license was released by the Free Software Foundation in June of 1991, and is in compliance with ten requirements of the OSD, listed in Table 1 [18, 50].

InnoDB is default storage engine in MySQL version 5.5. In all previous versions, MyISAM was the default storage engine [14]. In Version 5.5, MySQL also made changes to the InnoDB Input Output (I/O) subsystem designed to increase the I/O performance, and configurability [14]. In previous versions, InnoDB underutilized server capabilities by prefetching disk blocks and flushing dirty pages with only one background thread. Pages are the basic internal structure used to organize data in the database files. Dirty pages are modified pages that are uncommitted and remain in the buffer pool [30]. This version enables the utilization of multiple threads [31].

InnoDB is ACID compliant, with referential-integrity constraints, commit, rollback, and crash-recovery capabilities to protect data. The primary advantage for query performance with MyISAM is its lack of referential-integrity constraints. MyISAM is a much simpler storage structure, which reduces the amount of required server resources on large queries. MyISAM also includes full-text indexing. InnoDB's primary advantage for query performance is non-locking reads. The primary disadvantage on query performance for MyISAM is its dependence on table level locking. The primary disadvantage for InnoDB is its referential-integrity constraints [24, 28, 29]. These features are displayed in table 5.

Table 5. Feature Differences

Feature	MyISAM	InnoDB
Storage limits	256TB	64TB
Transactions	No	Yes
Locking granularity	Table	Row
Multi-Version Concurrency Control	No	Yes
Geospatial indexing support	Yes	No
Full-text search indexes	Yes	No
Clustered indexes	No	Yes
Data caches	No	Yes
Foreign key support	No	Yes

5.2 Quantitative Analysis

The two types of performance tests are specified by the TPC-DS are power tests (Tpt) and throughput tests (Ttt). The primary TPC-DS performance statistic specified is Queries per Hour for Decision Support (QphDS) [1].

5.2.1 Power Test Performance Comparison

Power Tests measured the performance of the SUT when processing a sequence of queries in a single stream fashion. These queries were executed in numerical order, with only one query active at a time. The power tests provide a statistic for comparison against concurrent session tests [1].

The data in Figure 34 show that MyISAM outperforms InnoDB by twelve percent when one query is active at a time. Percent change is calculated by dividing difference in execution time by the total execution time. The total execution time for MyISAM test 1 was 40760ms and 38930ms for test 2. The total execution time for InnoDB test 1 was 45320ms and 44152ms for test 2. The total execution time for both MyISAM powers tests was 79690ms. The total execution time for both InnoDB power tests was 89445ms. This yields the equation 1 - (89445 /

79690) = 0.122. This equation describes the difference in total power test execution times between MyISAM and InnoDB.

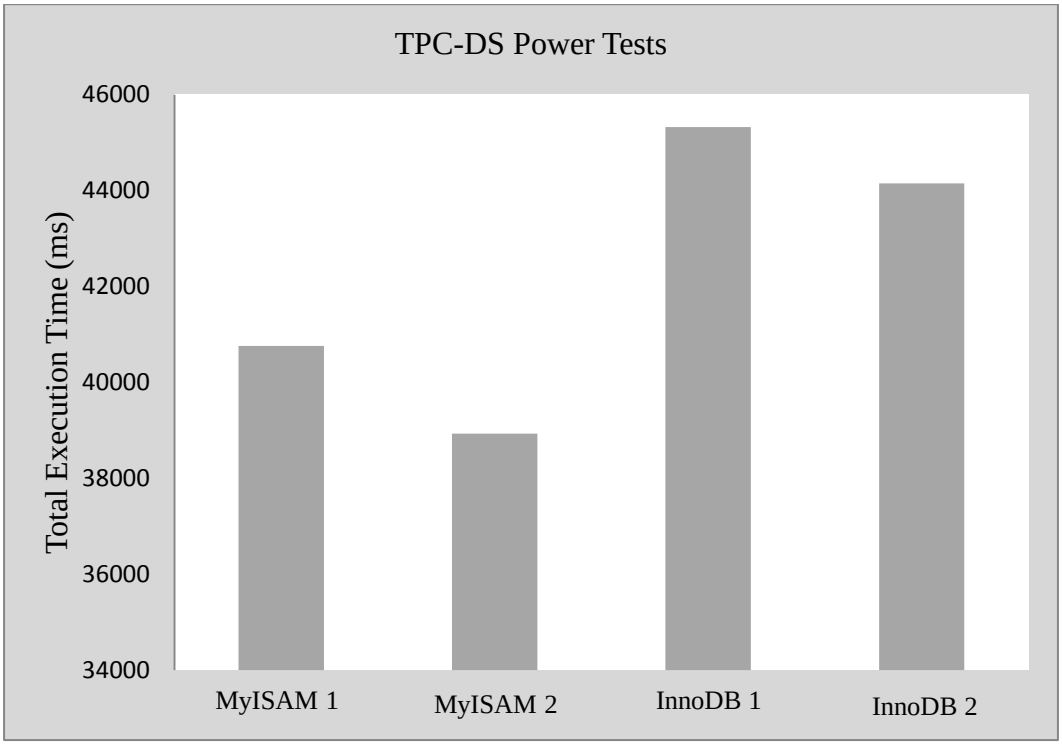


Figure 34. Power Test Performance Comparison

5.2.2 Throughput Test Performance Comparison

Throughput Tests measured the performance of the SUT when processing multiple concurrent user sessions. These queries were executed in random order, with multiple sessions active at a time. The throughput tests provide statistics (TTTn) for calculation in QphDS. Each test executed 20 sessions [1].

The data in Figure 35 show that MyISAM outperforms InnoDB by sixty-one percent when query streams have multiple concurrent queries. This figure was calculated by taking the sum of total execution times, and dividing. The total execution time for MyISAM test 1 was 7091504ms and 5937900ms for test 2. The total execution time for InnoDB test 1 was 9911152ms and 11010935ms for test 2. The total execution time for both MyISAM throughput

tests was 13029404ms. The total execution time for both InnoDB throughput tests was 20922087ms. This yields the equation 1 – (20922087 / 13029404) = 0.606. This equation describes the difference in total throughput test execution times between MyISAM and InnoDB.

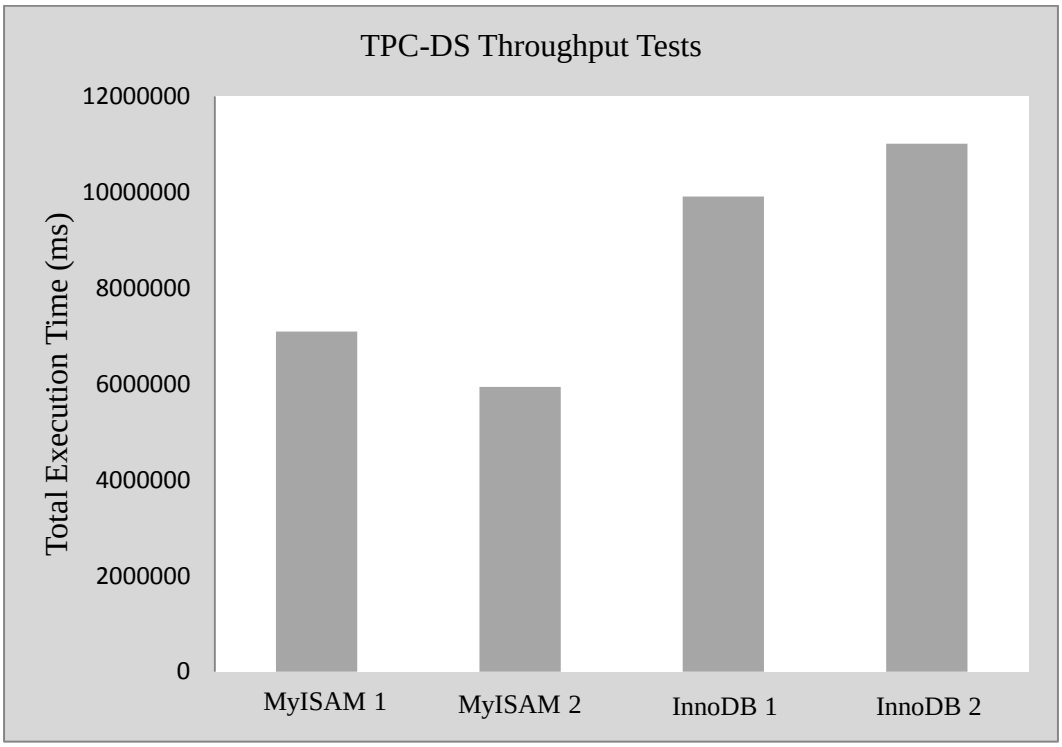


Figure 35. Throughput Test Performance Comparison

5.2.3 QphDS Comparison

The throughput tests provide statistics (TTTn) for calculation in QphDS. The timing interval for each query has been rounded to the nearest millisecond. To avoid zero values, values less than five tenths of a millisecond were rounded up to one millisecond. The scale factor determines the approximate raw size of the data produced by the TPC-DS dsgen. This research initially used a 10GB scale factor [1].

The data in Figure 37 show that MyISAM outperforms InnoDB in terms of QphDS by sixty-one percent. MyISAM has a calculated QphDS of 54.70. InnoDB has a calculated QphDS of 34.07. These figures were calculated by taking the product of the number of queries per

stream, the number of query runs, the number of seconds in an hour, and the scale factor, divided by the sum of the total execution time for the throughput tests and the product of the number of sessions and 0.01, as illustrated in figure 36. The number of streams is 20, and the scale factor is 10. For MyISAM, this yields the equation (99 * 2 * 3600) / (7092 + 5938) = 54.70. For InnoDB, this yields the equation (99 * 2 * 3600) / (9911 + 11011) = 34.07.

$$QphDS@SF = \frac{99 * 2 * 3600}{T_{tt1} + T_{tt2}}$$

Where:

- 99 is the number of queries per stream
- 2 is the number of query runs
- 3600 is the number of seconds in an hour
- T_{TT1} is the total elapsed time to complete the first throughput test
- T_{TT2} is the total elapsed time to complete the second throughput test
- SF is the scale factor used in the benchmark

Figure 36. Queries per Hour for Decision Support (QphDS) Formula

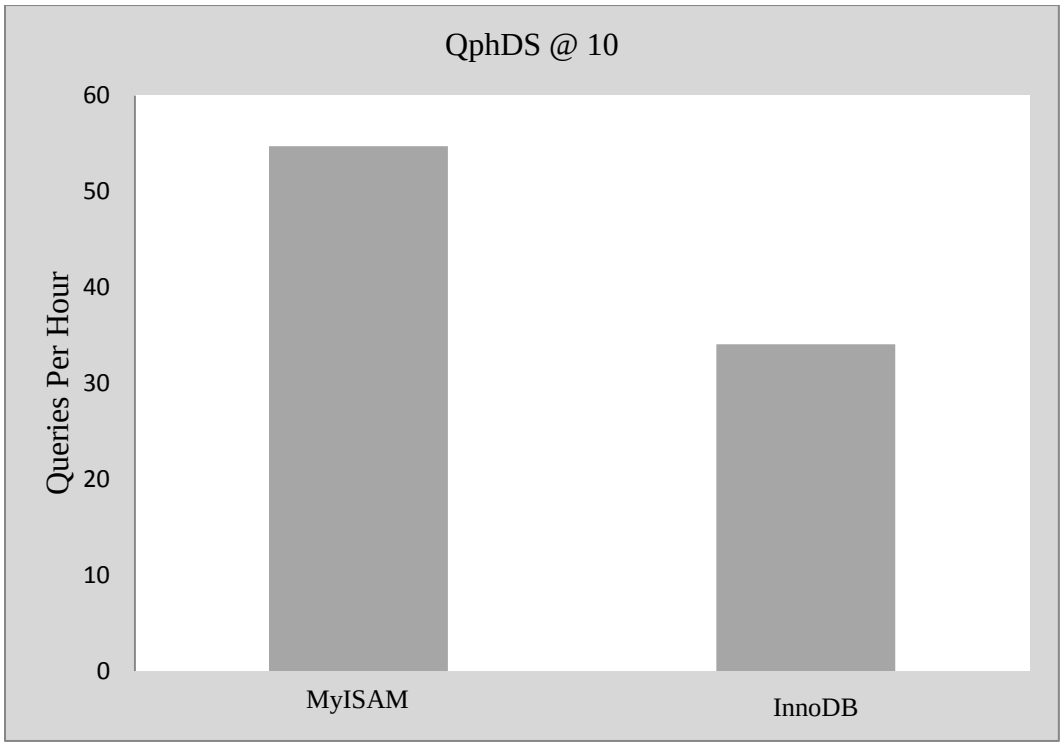


Figure 37. Queries per Hour for Decision Support (QphDS) Comparison

5.2.4 Scaled Data Set Test Performance Comparison

In addition to the TPC-DS benchmark, two additional throughput tests were performed to show how the execution times scale in proportion to data set size. The 12GB fact table data was first scaled to 9GB, a throughout test was executed to prefill the MySQL query cache, and a subsequent run was recorded. The 9GB fact table data was then scaled to 6GB, a throughout test was executed to prefill the MySQL query cache, and a subsequent run was recorded. These report measurements were compared to the existing 12GB data set throughput test first run. Only queries related to the fact tables were analyzed.

The data in Figure 38 show that MyISAM outperforms InnoDB by twenty four percent using the 6GB data set, thirty nine percent at 9GB, and forty five percent at 12GB. Percent change is calculated by dividing difference in execution time by the execution time. The mean percent change is reported in Figure 38. This equation illustrates the performance increase from MyISAM to InnoDB.

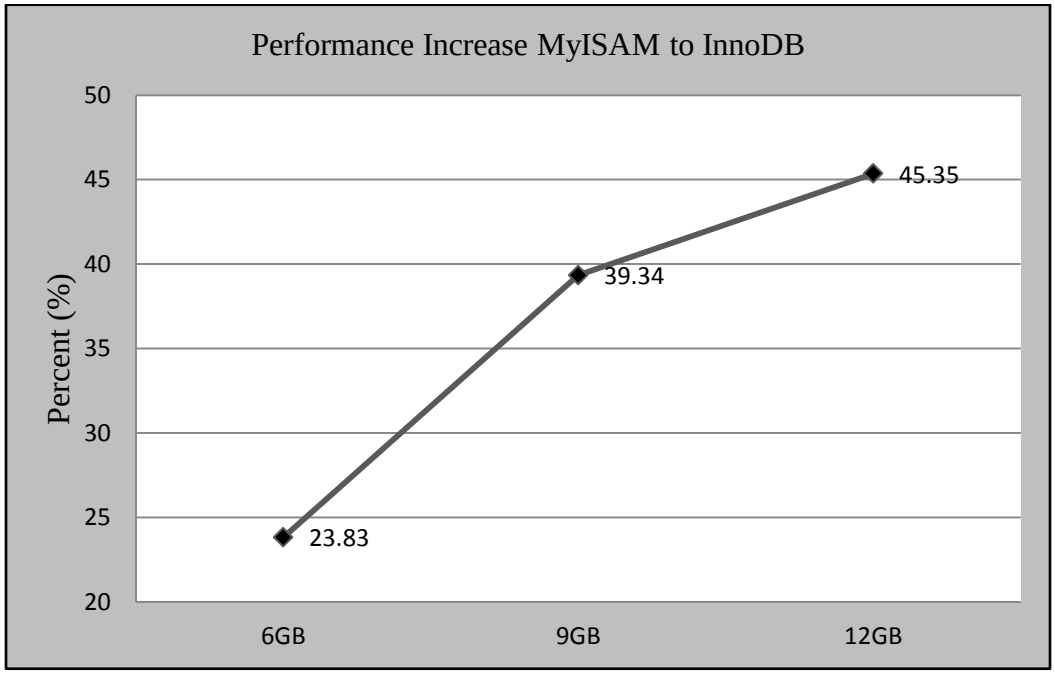


Figure 38. Percent Performance Increase MyISAM to InnoDB

Using the 6GB data set, the mean query one execution time is 300655ms for MyISAM and 531905ms for InnoDB. This yields the equation (531905 - 300655) / 531905 = 43.48 percent. For query two, the mean execution time is 387ms for MyISAM and 456ms for InnoDB. This yields the equation (456 - 387) / 456 = 15.13 percent. For query five, the mean execution time is 3897ms for MyISAM and 8511ms for InnoDB. This yields the equation (8511 - 3897) / 8511 = 54.21 percent. For query seven, the mean execution time is 875ms for MyISAM and 1759ms for InnoDB. This yields the equation (1759 - 875) / 1759 = 50.26 percent. For query eight, the mean execution time is 430ms for MyISAM and 497ms for InnoDB. This yields the equation (497 - 430) / 497 = 13.48 percent. For query nine, the mean execution time is 497ms for MyISAM and 372ms for InnoDB. This yields the equation (372 - 497) / 372 = -33.60 percent. The mean percent change for the 6GB data set is 23.83 percent.

Using the 9GB data set, the mean query one execution time is 396987ms for MyISAM and 715227ms for InnoDB. This yields the equation (715227 - 396987) / 715227 = 44.49 percent. For query two, the mean execution time is 449ms for MyISAM and 1070ms for InnoDB. This yields the equation (1070 - 449) / 1070 = 58.04 percent. For query five, the mean execution time is 9045ms for MyISAM and 29549ms for InnoDB. This yields the equation (29549 - 9045) / 29549 = 69.39 percent. For query seven, the mean execution time is 2683ms for MyISAM and 2193ms for InnoDB. This yields the equation (2193 - 2683) / 2193 = -22.34 percent. For query eight, the mean execution time is 276ms for MyISAM and 1199ms for InnoDB. This yields the equation (1199 - 276) / 1199 = 76.98 percent. For query nine, the mean execution time is 574ms for MyISAM and 634ms for InnoDB. This yields the equation (634 - 574) / 634 = 9.46 percent. The mean percent change for the 9GB data set is 39.34 percent.

Using the 12GB data set, the mean query one execution time is 600838ms for MyISAM and 826551ms for InnoDB. This yields the equation (826551 - 600838) / 826551 = 27.31 percent. For query two, the mean execution time is 253ms for MyISAM and 635ms for InnoDB. This yields the equation (635 - 253) / 635 = 60.16 percent. For query five, the mean execution time is 7291ms for MyISAM and 15915ms for InnoDB. This yields the equation (15915 - 7291) / 15915 = 51.11 percent. For query seven, the mean execution time is 2267ms for MyISAM and 1995ms for InnoDB. This yields the equation (1995 - 2267) / 1995 = -13.63 percent. For query eight, the mean execution time is 174ms for MyISAM and 337ms for InnoDB. This yields the equation (337 - 174) / 337 = 48.37 percent. For query nine, the mean execution time is 114ms for MyISAM and 9443ms for InnoDB. This yields the equation (9443 - 114) / 9443 = 98.79 percent. The mean percent change for the 12GB data set is 45.35 percent.

CHAPTER 6

CONCLUSIONS

6.1 Final Conclusions

Despite the recent upgrades in performance features for the InnoDB database engine, the MyISAM database engine has been shown to outperform the InnoDB database engine when performing the TPC-DS OLAP benchmark. This association was found in tests that include concurrent user sessions, as per Figure 34, as well as asynchronous user sessions, as per Figure 35. It has also been shown that MyISAM increasingly outperforms InnoDB as the data set size increases from 6GB to 12GB; see Figure 38.

Although MyISAM outperformed InnoDB in all tests performed, InnoDB provides ACID compliant transaction technologies that would be beneficial in a hybrid OLAP/OLTP system. Using the methods described in this research, DBA's designing an open source OLAP database system should carefully examine all available engines, and benchmark only in a clone of their production environment. Variation to schema, user sessions, data set, or system memory can also affect the outcome of the TPC-DS.

Thus, it has been observed that MyISAM opts for a query speed over integrity tradeoff that helps it to outperform InnoDB on OLAP queries.

6.2 Future Work

More research is warranted to examine the relationships between open source OLAP engines. Although it was observed that MyISAM outperformed InnoDB in all tests administered, the proportion of performance benefit showed a sub-linear trend as the data set size increased; see Figure 38. Future research should be performed on data sets larger than 12GB to establish increased confidence in these conclusions.

Future work in this area could include comparison against other storage engines, comparison of hybrid OLAP/OLTP systems, and benchmarks in a parallel processing environment. MySQL alone has ten open source database engines available. Hybrid systems that combine multiple database engines exist, and should be evaluated as viable solutions. With the advancements and cost reductions in parallel processing, it would be interesting to compare InnoDB and MySQL on a system with multiple processors. In addition, it would be interesting to perform this benchmark on a server with enough memory to store the entire database.

Finally, DBA's looking to implement an open source database OLAP system should perform the TPC-DS benchmark on an instance of their existing schema and dataset. This will produce accurate results with a high level of installation specific precision.

WORKS CITED

[1] Transaction Processing Performance Council, "TPC Benchmark DS 1.0," 2007.

[2] Y. Matias, "Trends in High Performance Analytics," in *ACM SIGMOD International Conference on Management of Data*, Chicago, IL, 2006.

[3] S. Chaudhuri and D. Umeshwar, "An Overview of Data Warehousing Technology and OLAP Technology," *SIGMOD Record*, vol. 26, no. 1, pp. 65-74, 1997.

[4] H. Luhn, "A Business Intelligence System," vol. 2, no. 4, pp. 314-319, October 1958.

[5] S. Chaudhuri, D. Umeshwar and V. Ganti, "Database Technology for Decision Support Systems," *IEEE Computer*, vol. 34, no. 12, pp. 48-55, 2001.

[6] M. Ritacco, "Business Intelligence - Now More Than Ever," 2001. [Online]. Available: www.clearpeaks.com/pdf/bi_nowmorethanever.pdf. [Accessed 6 February 2010].

[7] E. Codd, S. Codd and C. Salley, "Providing OLAP to User-Analysts: An IT Mandate," Hyperion Solutions, Sunnyvale, CA, 1993.

[8] G. V. Rossum, "Open Source Summit Trip Report," 10 April 1998. [Online]. Available: http://linuxgazette.net/issue28/rossum.html. [Accessed 26 January 2010].

[9] J. Paulson, G. D. C. Ltd., G. Succi and A. Eberlein, "An empirical study of open-source and closed-source software products," *IEEE Transactions on Software Engineering*, vol. 30, no. 4, pp. 246-256, 2004.

[10] D. A. Wheeler, "Why Open Source Software? Look at the Numbers!," 16 April 2007. [Online]. Available: http://www.dwheeler.com/oss_fs_why.html. [Accessed 5 August 2010].

[11] M. Madsen, "Open Source Solutions: Managing, Analyzing and Delivering Business

Information," BeyeNETWORK, Boulder, CO, 2009.

[12] C. Thomsen and T. B. Pedersen, "A Survey of Open Source Tools for Business Intelligence," in *Proceedings of the 7th international conference on Data Warehousing and Knowledge Discovery*, Berlin, 2005.

[13] Oracle, "MySQL 5.5 Reference Manual :: Storage Engine Performance," 2011. [Online]. Available: http://dev.mysql.com/doc/refman/5.5/en/performance-schema.html. [Accessed 11 April 2012].

[14] Oracle, "InnoDB as the Default MySQL Storage Engine," 2012. [Online]. Available: http://dev.mysql.com/doc/refman/5.5/en/innodb-default-se.html. [Accessed 14 April 2012].

[15] R. D. Schneider, MySQL Database Design and Tuning, MySQL Press, 2005.

[16] R. O. Nambiar and M. Poess, "The Making of TPC-DS," *Proceedings of the 32nd international conference on Very large data bases*, pp. 1049-1058, 12-15 September 2006.

[17] J. J. Fink, The Car Culture, Cambridge: MIT Press, 1977.

[18] M. Cote, J. Governor and S. O'Grady, "Redmonk," 19 April 2007. [Online]. Available: http://redmonk.com/public/goingopensource/Exploring_Open_Source-Options.pdf. [Accessed 26 January 2010].

[19] The Open Source Initiative, "The Open Source Definition," [Online]. Available: http://www.opensource.org/docs/osd. [Accessed 26 January 2010].

[20] The Open Source Initiative, "Open Source Licenses by Category," 2010. [Online]. Available: http://www.opensource.org/licenses/category. [Accessed 26 January 2010].

[21] S. Curry, *MySQL AB Completes Record Quarter & Year; 2005 Is a Year of Growth & Technical Achievement for World's Most Popular Open Source Database*, Cupertino,

California, 2006.

[22] Webopedia, "Database Engine," 2012. [Online]. Available: http://www.webopedia.com/TERM/D/database_engine.html. [Accessed 14 April 2012].

[23] E. F. Codd, "A Relational Model for Large Shared Data Banks," *Communications of the ACM*, vol. 13, no. 6, pp. 377-387, 1970.

[24] MySQL AB, "Enterprise Data Warehousing with MySQL," 2007. [Online]. Available: http://www.scribd.com/doc/3003152/Enterprise-Data-Warehousing-with-MySQL. [Accessed 14 April 2012].

[25] Oracle, "MySQL :: Market Share," 2012. [Online]. Available: http://www.mysql.com/why-mysql/marketshare/. [Accessed 26 December 2012].

[26] D. Dunn, "Open Source Secrets," 01 August 2007. [Online]. Available: http://www.cioinsight.com/c/a/Technology/Open-Source-Secrets/. [Accessed 7 April 2012].

[27] Oracle, "MySQL 5.5 Reference Manual :: 14 Storage Engines," 2012. [Online]. Available: http://dev.mysql.com/doc/refman/5.5/en/storage-engines.html. [Accessed 14 April 2012].

[28] M. Peters, "MySQL Storage Engines," 1 January 2008. [Online]. Available: http://www.softwareprojects.com/resources/programming/t-mysql-storage-engines-1470.html. [Accessed 8 April 2012].

[29] Y. Yang, "MySQL Engines: InnoDB vs. MyISAM – A Comparison of Pros and Cons," 2 September 2009. [Online]. Available: http://www.kavoir.com/2009/09/mysql-engines-innodb-vs-myisam-a-comparison-of-pros-and-cons.html. [Accessed 8 April 2012].

[30] Oracle, "Controlling the Flushing Rate of Dirty Pages," 2012. [Online]. Available: http://dev.mysql.com/doc/innodb/1.1/en/innodb-performance-adaptive_flushing.html.

[Accessed 28 April 2012].

[31] Oracle, "InnoDB Subsystem Changes," 2012. [Online]. Available: http://dev.mysql.com/doc/refman/5.5/en/innodb-io-changes.html. [Accessed 28 April 2012].

[32] E. Codd, "Extending the Database Relational Model to Capture More Meaning," *ACM Trans. Database Syst.*, vol. 4, no. 4, pp. 397 - 434, 1979.

[33] J. A. Palinski, Oracle SQL and PL/SQL Handbook, Boston, MA: Pearson Educaiton, 2003, pp. 323-336.

[34] E. Thomsen, OLAP Solutions, New York: John Wiley & Sons, Inc., 2002.

[35] The OLAP Council, "OLAP and OLAP Server Definitions," 1997. [Online]. Available: http://www.olapcouncil.org/research/glossaryly.htm. [Accessed 12 August 2010].

[36] InfoLibrarian, "InfoLibrarian Metadata White Paper," 2005. [Online]. Available: http://www.infolibcorp.com/Metadata White Paper.pdf. [Accessed 12 August 2010].

[37] Oracle, "Schemas," 1999. [Online]. Available: http://www.uriit.ru/japan/Our_Resources/Doc_Oracle8i/server.816/a76994/schemas.htm. [Accessed 13 August 2010].

[38] M. Golfarelli and S. Rizzi, Data Warehouse Design: Modern Principles and Methodologies, New York, NY: McGraw-Hill, 2009.

[39] Bell Laboratories, "AQUA Project Home Page," 7 August 2001. [Online]. Available: http://www.bell-labs.com/project/aqua/. [Accessed 17 August 2010].

[40] M. Golfarelli, "Open Source BI Platforms: A Functional and Architectural Comparison," in *Data Warehousing and Knowledge Discovery*, Berlin, 2009.

[41] Microsoft Corporation, "XML for Analysis Specification," 24 April 2001. [Online].

Available: http://msdn.microsoft.com/en-us/library/ms977626.aspx. [Accessed 27 August 2010].

[42] K. Henry, "Objective viewpoint: JDBC - Java database connectivity," *Crossroads,* p. 3, 2001.

[43] W. Pearson, "MDX at First Glance," 6 November 2002. [Online]. Available: http://www.databasejournal.com/features/mssql/article.php/10894_1495511_2/MDX-at-First-Glance-Introduction-to-SQL-Server-MDX-Essentials.htm. [Accessed 23 August 2010].

[44] Microsoft Corporation, "MDX Syntax Elements," 2010. [Online]. Available: http://msdn.microsoft.com/en-us/library/ms146020.aspx. [Accessed 23 August 2010].

[45] Simba Technologies, "What is XMLA?," 2010. [Online]. Available: http://www.xmlforanalysis.com/index.htm. [Accessed 23 August 2010].

[46] M. Poess, R. O. Nambiar and D. Walrath, "Why You Should Run TPC-DS:A Workload Analysis," *Proceedings of the 33rd international conference on Very large data bases,* pp. 1138-1149, 2007.

[47] Transaction Processing Performance Council, "TPC-DS," 2012. [Online]. Available: http://www.tpc.org/tpcds/default.asp. [Accessed 24 March 2012].

[48] Apache Software Foundation, "Apache JMeter User Manual," 2012. [Online]. Available: http://jmeter.apache.org/usermanual/intro.html. [Accessed 24 March 2012].

[49] Apache Software Foundation, "Apache JMeter Functions," DocJar, 2012. [Online]. Available: http://www.docjar.com/html/api/org/apache/jmeter/functions/Random.java.html. [Accessed 22 October 2012].

[50] Free Software Foundation, "GNU General Public License, version 2," 10 June 2012. [Online]. Available: http://www.gnu.org/licenses/gpl-2.0.html. [Accessed 15 August 2012].

[51] Webster, "Intelligence," in *Websters New Collegiate Dictionary*, Springfield, MA, G. & G. Merriam Co.,, 1958.

[52] The Open Source Initiative, "About Us," [Online]. Available: http://www.opensource.org/about. [Accessed 26 January 2010].

[53] N. Pendse, "What is OLAP?," 3 March 2008. [Online]. Available: http://www.bi-verdict.com/fileadmin/FreeAnalyses/fasmi.htm. [Accessed 26 February 2010].

[54] G. A. Moore, Living on the Fault Line, New York, NY: HarperColling Publishers, 2000.

[55] IDC, ""Vicious Circle" Is Impacting the U.S. Economy and IT Spending," 30 April 2001.

[56] The OLAP Council, "About Us," 1997. [Online]. Available: http://www.olapcouncil.org/about/aboutly.htm. [Accessed 19 August 2010].

[57] The OLAP Council, "OLAP Council APB-1 Benchmark Specification Release II," OLAP Council, 1998.

[58] R. M. Lerner, "MySQL Deserves a Double Take," *Linux journal*, p. 18, April 2007.

[59] StrateBI, "Open Source B.I. comparative," 2010.

[60] Oracle, "Overview of MySQL Storage Engine Architecture," 2012. [Online]. Available: http://dev.mysql.com/doc/refman/5.1/en/pluggable-storage-overview.html. [Accessed 12 April 2012].

APPENDIX A

Physical Representation of TPC-DS Relational Schema

```
create table dbgen_version
(
    dv_version                varchar(16)                    ,
    dv_create_date            date                           ,
    dv_create_time            time                           ,
    dv_cmdline_args           varchar(200)
);

create table customer_address
(
    ca_address_sk             integer               not null,
    ca_address_id             char(16)              not null,
    ca_street_number          char(10)                       ,
    ca_street_name            varchar(60)                    ,
    ca_street_type            char(15)                       ,
    ca_suite_number           char(10)                       ,
    ca_city                   varchar(60)                    ,
    ca_county                 varchar(30)                    ,
    ca_state                  char(2)                        ,
    ca_zip                    char(10)                       ,
    ca_country                varchar(20)                    ,
    ca_gmt_offset             decimal(5,2)                   ,
    ca_location_type          char(20)                       ,
    primary key (ca_address_sk)
);

create table customer_demographics
(
    cd_demo_sk                integer               not null,
    cd_gender                 char(1)                        ,
    cd_marital_status         char(1)                        ,
    cd_education_status       char(20)                       ,
    cd_purchase_estimate      integer                        ,
    cd_credit_rating          char(10)                       ,
    cd_dep_count              integer                        ,
    cd_dep_employed_count     integer                        ,
    cd_dep_college_count      integer                        ,
    primary key (cd_demo_sk)
);

create table date_dim
(
    d_date_sk                 integer               not null,
    d_date_id                 char(16)              not null,
    d_date                    date                           ,
    d_month_seq               integer                        ,
    d_week_seq                integer                        ,
    d_quarter_seq             integer                        ,
    d_year                    integer                        ,
    d_dow                     integer                        ,
    d_moy                     integer                        ,
    d_dom                     integer                        ,
```

```
    d_qoy                     integer                       ,
    d_fy_year                 integer                       ,
    d_fy_quarter_seq          integer                       ,
    d_fy_week_seq             integer                       ,
    d_day_name                char(9)                       ,
    d_quarter_name            char(6)                       ,
    d_holiday                 char(1)                       ,
    d_weekend                 char(1)                       ,
    d_following_holiday       char(1)                       ,
    d_first_dom               integer                       ,
    d_last_dom                integer                       ,
    d_same_day_ly             integer                       ,
    d_same_day_lq             integer                       ,
    d_current_day             char(1)                       ,
    d_current_week            char(1)                       ,
    d_current_month           char(1)                       ,
    d_current_quarter         char(1)                       ,
    d_current_year            char(1)                       ,
    primary key (d_date_sk)
);

create table warehouse
(
    w_warehouse_sk            integer               not null,
    w_warehouse_id            char(16)              not null,
    w_warehouse_name          varchar(20)                   ,
    w_warehouse_sq_ft         integer                       ,
    w_street_number           char(10)                      ,
    w_street_name             varchar(60)                   ,
    w_street_type             char(15)                      ,
    w_suite_number            char(10)                      ,
    w_city                    varchar(60)                   ,
    w_county                  varchar(30)                   ,
    w_state                   char(2)                       ,
    w_zip                     char(10)                      ,
    w_country                 varchar(20)                   ,
    w_gmt_offset              decimal(5,2)                  ,
    primary key (w_warehouse_sk)
);

create table ship_mode
(
    sm_ship_mode_sk           integer               not null,
    sm_ship_mode_id           char(16)              not null,
    sm_type                   char(30)                      ,
    sm_code                   char(10)                      ,
    sm_carrier                char(20)                      ,
    sm_contract               char(20)                      ,
    primary key (sm_ship_mode_sk)
);

create table time_dim
(
    t_time_sk                 integer               not null,
    t_time_id                 char(16)              not null,
    t_time                    integer                       ,
    t_hour                    integer                       ,
```

```
    t_minute                    integer                         ,
    t_second                    integer                         ,
    t_am_pm                     char(2)                         ,
    t_shift                     char(20)                        ,
    t_sub_shift                 char(20)                        ,
    t_meal_time                 char(20)                        ,
    primary key (t_time_sk)
);

create table reason
(
    r_reason_sk                 integer                 not null,
    r_reason_id                 char(16)                not null,
    r_reason_desc               char(100)                       ,
    primary key (r_reason_sk)
);

create table income_band
(
    ib_income_band_sk           integer                 not null,
    ib_lower_bound              integer                         ,
    ib_upper_bound              integer                         ,
    primary key (ib_income_band_sk)
);

create table item
(
    i_item_sk                   integer                 not null,
    i_item_id                   char(16)                not null,
    i_rec_start_date            date                            ,
    i_rec_end_date              date                            ,
    i_item_desc                 varchar(200)                    ,
    i_current_price             decimal(7,2)                    ,
    i_wholesale_cost            decimal(7,2)                    ,
    i_brand_id                  integer                         ,
    i_brand                     char(50)                        ,
    i_class_id                  integer                         ,
    i_class                     char(50)                        ,
    i_category_id               integer                         ,
    i_category                  char(50)                        ,
    i_manufact_id               integer                         ,
    i_manufact                  char(50)                        ,
    i_size                      char(20)                        ,
    i_formulation               char(20)                        ,
    i_color                     char(20)                        ,
    i_units                     char(10)                        ,
    i_container                 char(10)                        ,
    i_manager_id                integer                         ,
    i_product_name              char(50)                        ,
    primary key (i_item_sk)
);

create table store
(
    s_store_sk                  integer                 not null,
    s_store_id                  char(16)                not null,
    s_rec_start_date            date                            ,
```

```
    s_rec_end_date            date                          ,
    s_closed_date_sk          integer                       ,
    s_store_name              varchar(50)                   ,
    s_number_employees        integer                       ,
    s_floor_space             integer                       ,
    s_hours                   char(20)                      ,
    s_manager                 varchar(40)                   ,
    s_market_id               integer                       ,
    s_geography_class         varchar(100)                  ,
    s_market_desc             varchar(100)                  ,
    s_market_manager          varchar(40)                   ,
    s_division_id             integer                       ,
    s_division_name           varchar(50)                   ,
    s_company_id              integer                       ,
    s_company_name            varchar(50)                   ,
    s_street_number           varchar(10)                   ,
    s_street_name             varchar(60)                   ,
    s_street_type             char(15)                      ,
    s_suite_number            char(10)                      ,
    s_city                    varchar(60)                   ,
    s_county                  varchar(30)                   ,
    s_state                   char(2)                       ,
    s_zip                     char(10)                      ,
    s_country                 varchar(20)                   ,
    s_gmt_offset              decimal(5,2)                  ,
    s_tax_precentage          decimal(5,2)                  ,
    primary key (s_store_sk)
);

create table call_center
(
    cc_call_center_sk         integer               not null,
    cc_call_center_id         char(16)              not null,
    cc_rec_start_date         date                          ,
    cc_rec_end_date           date                          ,
    cc_closed_date_sk         integer                       ,
    cc_open_date_sk           integer                       ,
    cc_name                   varchar(50)                   ,
    cc_class                  varchar(50)                   ,
    cc_employees              integer                       ,
    cc_sq_ft                  integer                       ,
    cc_hours                  char(20)                      ,
    cc_manager                varchar(40)                   ,
    cc_mkt_id                 integer                       ,
    cc_mkt_class              char(50)                      ,
    cc_mkt_desc               varchar(100)                  ,
    cc_market_manager         varchar(40)                   ,
    cc_division               integer                       ,
    cc_division_name          varchar(50)                   ,
    cc_company                integer                       ,
    cc_company_name           char(50)                      ,
    cc_street_number          char(10)                      ,
    cc_street_name            varchar(60)                   ,
    cc_street_type            char(15)                      ,
    cc_suite_number           char(10)                      ,
    cc_city                   varchar(60)                   ,
    cc_county                 varchar(30)                   ,
```

```
    cc_state                    char(2)                        ,
    cc_zip                      char(10)                       ,
    cc_country                  varchar(20)                    ,
    cc_gmt_offset               decimal(5,2)                   ,
    cc_tax_percentage           decimal(5,2)                   ,
    primary key (cc_call_center_sk)
);

create table customer
(
    c_customer_sk               integer                 not null,
    c_customer_id               char(16)                not null,
    c_current_cdemo_sk          integer                        ,
    c_current_hdemo_sk          integer                        ,
    c_current_addr_sk           integer                        ,
    c_first_shipto_date_sk      integer                        ,
    c_first_sales_date_sk       integer                        ,
    c_salutation                char(10)                       ,
    c_first_name                char(20)                       ,
    c_last_name                 char(30)                       ,
    c_preferred_cust_flag       char(1)                        ,
    c_birth_day                 integer                        ,
    c_birth_month               integer                        ,
    c_birth_year                integer                        ,
    c_birth_country             varchar(20)                    ,
    c_login                     char(13)                       ,
    c_email_address             char(50)                       ,
    c_last_review_date          char(10)                       ,
    primary key (c_customer_sk)
);

create table web_site
(
    web_site_sk                 integer                 not null,
    web_site_id                 char(16)                not null,
    web_rec_start_date          date                           ,
    web_rec_end_date            date                           ,
    web_name                    varchar(50)                    ,
    web_open_date_sk            integer                        ,
    web_close_date_sk           integer                        ,
    web_class                   varchar(50)                    ,
    web_manager                 varchar(40)                    ,
    web_mkt_id                  integer                        ,
    web_mkt_class               varchar(50)                    ,
    web_mkt_desc                varchar(100)                   ,
    web_market_manager          varchar(40)                    ,
    web_company_id              integer                        ,
    web_company_name            char(50)                       ,
    web_street_number           char(10)                       ,
    web_street_name             varchar(60)                    ,
    web_street_type             char(15)                       ,
    web_suite_number            char(10)                       ,
    web_city                    varchar(60)                    ,
    web_county                  varchar(30)                    ,
    web_state                   char(2)                        ,
    web_zip                     char(10)                       ,
    web_country                 varchar(20)                    ,
```

```
    web_gmt_offset            decimal(5,2)                  ,
    web_tax_percentage        decimal(5,2)                  ,
    primary key (web_site_sk)
);

create table store_returns
(
    sr_returned_date_sk       integer                       ,
    sr_return_time_sk         integer                       ,
    sr_item_sk                integer               not null,
    sr_customer_sk            integer                       ,
    sr_cdemo_sk               integer                       ,
    sr_hdemo_sk               integer                       ,
    sr_addr_sk                integer                       ,
    sr_store_sk               integer                       ,
    sr_reason_sk              integer                       ,
    sr_ticket_number          integer               not null,
    sr_return_quantity        integer                       ,
    sr_return_amt             decimal(7,2)                  ,
    sr_return_tax             decimal(7,2)                  ,
    sr_return_amt_inc_tax     decimal(7,2)                  ,
    sr_fee                    decimal(7,2)                  ,
    sr_return_ship_cost       decimal(7,2)                  ,
    sr_refunded_cash          decimal(7,2)                  ,
    sr_reversed_charge        decimal(7,2)                  ,
    sr_store_credit           decimal(7,2)                  ,
    sr_net_loss               decimal(7,2)                  ,
    primary key (sr_item_sk, sr_ticket_number)
);

create table household_demographics
(
    hd_demo_sk                integer               not null,
    hd_income_band_sk         integer                       ,
    hd_buy_potential          char(15)                      ,
    hd_dep_count              integer                       ,
    hd_vehicle_count          integer                       ,
    primary key (hd_demo_sk)
);

create table web_page
(
    wp_web_page_sk            integer               not null,
    wp_web_page_id            char(16)              not null,
    wp_rec_start_date         date                          ,
    wp_rec_end_date           date                          ,
    wp_creation_date_sk       integer                       ,
    wp_access_date_sk         integer                       ,
    wp_autogen_flag           char(1)                       ,
    wp_customer_sk            integer                       ,
    wp_url                    varchar(100)                  ,
    wp_type                   char(50)                      ,
    wp_char_count             integer                       ,
    wp_link_count             integer                       ,
    wp_image_count            integer                       ,
    wp_max_ad_count           integer                       ,
    primary key (wp_web_page_sk)
```

```
);

create table promotion
(
    p_promo_sk                 integer                 not null,
    p_promo_id                 char(16)                not null,
    p_start_date_sk            integer                         ,
    p_end_date_sk              integer                         ,
    p_item_sk                  integer                         ,
    p_cost                     decimal(15,2)                   ,
    p_response_target          integer                         ,
    p_promo_name               char(50)                        ,
    p_channel_dmail            char(1)                         ,
    p_channel_email            char(1)                         ,
    p_channel_catalog          char(1)                         ,
    p_channel_tv               char(1)                         ,
    p_channel_radio            char(1)                         ,
    p_channel_press            char(1)                         ,
    p_channel_event            char(1)                         ,
    p_channel_demo             char(1)                         ,
    p_channel_details          varchar(100)                    ,
    p_purpose                  char(15)                        ,
    p_discount_active          char(1)                         ,
    primary key (p_promo_sk)
);

create table catalog_page
(
    cp_catalog_page_sk         integer                 not null,
    cp_catalog_page_id         char(16)                not null,
    cp_start_date_sk           integer                         ,
    cp_end_date_sk             integer                         ,
    cp_department              varchar(50)                     ,
    cp_catalog_number          integer                         ,
    cp_catalog_page_number     integer                         ,
    cp_description             varchar(100)                    ,
    cp_type                    varchar(100)                    ,
    primary key (cp_catalog_page_sk)
);

create table inventory
(
    inv_date_sk                integer                 not null,
    inv_item_sk                integer                 not null,
    inv_warehouse_sk           integer                 not null,
    inv_quantity_on_hand       integer                         ,
    primary key (inv_date_sk, inv_item_sk, inv_warehouse_sk)
);

create table catalog_returns
(
    cr_returned_date_sk        integer                         ,
    cr_returned_time_sk        integer                         ,
    cr_item_sk                 integer                 not null,
    cr_refunded_customer_sk    integer                         ,
    cr_refunded_cdemo_sk       integer                         ,
    cr_refunded_hdemo_sk       integer                         ,
```

```
    cr_refunded_addr_sk       integer                       ,
    cr_returning_customer_sk  integer                       ,
    cr_returning_cdemo_sk     integer                       ,
    cr_returning_hdemo_sk     integer                       ,
    cr_returning_addr_sk      integer                       ,
    cr_call_center_sk         integer                       ,
    cr_catalog_page_sk        integer                       ,
    cr_ship_mode_sk           integer                       ,
    cr_warehouse_sk           integer                       ,
    cr_reason_sk              integer                       ,
    cr_order_number           integer               not null,
    cr_return_quantity        integer                       ,
    cr_return_amount          decimal(7,2)                  ,
    cr_return_tax             decimal(7,2)                  ,
    cr_return_amt_inc_tax     decimal(7,2)                  ,
    cr_fee                    decimal(7,2)                  ,
    cr_return_ship_cost       decimal(7,2)                  ,
    cr_refunded_cash          decimal(7,2)                  ,
    cr_reversed_charge        decimal(7,2)                  ,
    cr_store_credit           decimal(7,2)                  ,
    cr_net_loss               decimal(7,2)                  ,
    primary key (cr_item_sk, cr_order_number)
);

create table web_returns
(
    wr_returned_date_sk       integer                       ,
    wr_returned_time_sk       integer                       ,
    wr_item_sk                integer               not null,
    wr_refunded_customer_sk   integer                       ,
    wr_refunded_cdemo_sk      integer                       ,
    wr_refunded_hdemo_sk      integer                       ,
    wr_refunded_addr_sk       integer                       ,
    wr_returning_customer_sk  integer                       ,
    wr_returning_cdemo_sk     integer                       ,
    wr_returning_hdemo_sk     integer                       ,
    wr_returning_addr_sk      integer                       ,
    wr_web_page_sk            integer                       ,
    wr_reason_sk              integer                       ,
    wr_order_number           integer               not null,
    wr_return_quantity        integer                       ,
    wr_return_amt             decimal(7,2)                  ,
    wr_return_tax             decimal(7,2)                  ,
    wr_return_amt_inc_tax     decimal(7,2)                  ,
    wr_fee                    decimal(7,2)                  ,
    wr_return_ship_cost       decimal(7,2)                  ,
    wr_refunded_cash          decimal(7,2)                  ,
    wr_reversed_charge        decimal(7,2)                  ,
    wr_account_credit         decimal(7,2)                  ,
    wr_net_loss               decimal(7,2)                  ,
    primary key (wr_item_sk, wr_order_number)
);

create table web_sales
(
    ws_sold_date_sk           integer                       ,
    ws_sold_time_sk           integer                       ,
```

```
    ws_ship_date_sk           integer                       ,
    ws_item_sk                integer               not null,
    ws_bill_customer_sk       integer                       ,
    ws_bill_cdemo_sk          integer                       ,
    ws_bill_hdemo_sk          integer                       ,
    ws_bill_addr_sk           integer                       ,
    ws_ship_customer_sk       integer                       ,
    ws_ship_cdemo_sk          integer                       ,
    ws_ship_hdemo_sk          integer                       ,
    ws_ship_addr_sk           integer                       ,
    ws_web_page_sk            integer                       ,
    ws_web_site_sk            integer                       ,
    ws_ship_mode_sk           integer                       ,
    ws_warehouse_sk           integer                       ,
    ws_promo_sk               integer                       ,
    ws_order_number           integer               not null,
    ws_quantity               integer                       ,
    ws_wholesale_cost         decimal(7,2)                  ,
    ws_list_price             decimal(7,2)                  ,
    ws_sales_price            decimal(7,2)                  ,
    ws_ext_discount_amt       decimal(7,2)                  ,
    ws_ext_sales_price        decimal(7,2)                  ,
    ws_ext_wholesale_cost     decimal(7,2)                  ,
    ws_ext_list_price         decimal(7,2)                  ,
    ws_ext_tax                decimal(7,2)                  ,
    ws_coupon_amt             decimal(7,2)                  ,
    ws_ext_ship_cost          decimal(7,2)                  ,
    ws_net_paid               decimal(7,2)                  ,
    ws_net_paid_inc_tax       decimal(7,2)                  ,
    ws_net_paid_inc_ship      decimal(7,2)                  ,
    ws_net_paid_inc_ship_tax  decimal(7,2)                  ,
    ws_net_profit             decimal(7,2)                  ,
    primary key (ws_item_sk, ws_order_number)
);

create table catalog_sales
(
    cs_sold_date_sk           integer                       ,
    cs_sold_time_sk           integer                       ,
    cs_ship_date_sk           integer                       ,
    cs_bill_customer_sk       integer                       ,
    cs_bill_cdemo_sk          integer                       ,
    cs_bill_hdemo_sk          integer                       ,
    cs_bill_addr_sk           integer                       ,
    cs_ship_customer_sk       integer                       ,
    cs_ship_cdemo_sk          integer                       ,
    cs_ship_hdemo_sk          integer                       ,
    cs_ship_addr_sk           integer                       ,
    cs_call_center_sk         integer                       ,
    cs_catalog_page_sk        integer                       ,
    cs_ship_mode_sk           integer                       ,
    cs_warehouse_sk           integer                       ,
    cs_item_sk                integer               not null,
    cs_promo_sk               integer                       ,
    cs_order_number           integer               not null,
    cs_quantity               integer                       ,
    cs_wholesale_cost         decimal(7,2)                  ,
```

```
    cs_list_price             decimal(7,2)                  ,
    cs_sales_price            decimal(7,2)                  ,
    cs_ext_discount_amt       decimal(7,2)                  ,
    cs_ext_sales_price        decimal(7,2)                  ,
    cs_ext_wholesale_cost     decimal(7,2)                  ,
    cs_ext_list_price         decimal(7,2)                  ,
    cs_ext_tax                decimal(7,2)                  ,
    cs_coupon_amt             decimal(7,2)                  ,
    cs_ext_ship_cost          decimal(7,2)                  ,
    cs_net_paid               decimal(7,2)                  ,
    cs_net_paid_inc_tax       decimal(7,2)                  ,
    cs_net_paid_inc_ship      decimal(7,2)                  ,
    cs_net_paid_inc_ship_tax  decimal(7,2)                  ,
    cs_net_profit             decimal(7,2)                  ,
    primary key (cs_item_sk, cs_order_number)
);

create table store_sales
(
    ss_sold_date_sk           integer                       ,
    ss_sold_time_sk           integer                       ,
    ss_item_sk                integer               not null,
    ss_customer_sk            integer                       ,
    ss_cdemo_sk               integer                       ,
    ss_hdemo_sk               integer                       ,
    ss_addr_sk                integer                       ,
    ss_store_sk               integer                       ,
    ss_promo_sk               integer                       ,
    ss_ticket_number          integer               not null,
    ss_quantity               integer                       ,
    ss_wholesale_cost         decimal(7,2)                  ,
    ss_list_price             decimal(7,2)                  ,
    ss_sales_price            decimal(7,2)                  ,
    ss_ext_discount_amt       decimal(7,2)                  ,
    ss_ext_sales_price        decimal(7,2)                  ,
    ss_ext_wholesale_cost     decimal(7,2)                  ,
    ss_ext_list_price         decimal(7,2)                  ,
    ss_ext_tax                decimal(7,2)                  ,
    ss_coupon_amt             decimal(7,2)                  ,
    ss_net_paid               decimal(7,2)                  ,
    ss_net_paid_inc_tax       decimal(7,2)                  ,
    ss_net_profit             decimal(7,2)                  ,
    primary key (ss_item_sk, ss_ticket_number)
);

delete from call_center;
delete from catalog_page;
delete from catalog_returns;
delete from catalog_sales;
delete from customer;
delete from customer_address;
delete from customer_demographics;
delete from date_dim;
delete from dbgen_version;
delete from household_demographics;
delete from income_band;
delete from inventory;
```

```
delete from item;
delete from promotion;
delete from reason;
delete from ship_mode;
delete from store;
delete from store_returns;
delete from store_sales;
delete from time_dim;
delete from warehouse;
delete from web_page;
delete from web_returns;
delete from web_sales;
delete from web_site;

load data infile '/mnt/external/data/call_center.dat'
into table call_center fields terminated by '|'
(cc_call_center_sk,cc_call_center_id,cc_rec_start_date,cc_rec_end_date,cc_clo
sed_date_sk,cc_open_date_sk,cc_name,cc_class,cc_employees,cc_sq_ft,cc_hours,c
c_manager,cc_mkt_id,cc_mkt_class,cc_mkt_desc,cc_market_manager,cc_division,cc
_division_name,cc_company,cc_company_name,cc_street_number,cc_street_name,cc_
street_type,cc_suite_number,cc_city,cc_county,cc_state,cc_zip,cc_country,cc_g
mt_offset,cc_tax_percentage);

load data infile '/mnt/external/data/catalog_page.dat'
into table catalog_page fields terminated by '|'
(cp_catalog_page_sk,cp_catalog_page_id,cp_start_date_sk,cp_end_date_sk,cp_dep
artment,cp_catalog_number,cp_catalog_page_number,cp_description,cp_type);

load data infile '/mnt/external/data/catalog_returns.dat'
into table catalog_returns fields terminated by '|'
(cr_returned_date_sk,cr_returned_time_sk,cr_item_sk,cr_refunded_customer_sk,c
r_refunded_cdemo_sk,cr_refunded_hdemo_sk,cr_refunded_addr_sk,cr_returning_cus
tomer_sk,cr_returning_cdemo_sk,cr_returning_hdemo_sk,cr_returning_addr_sk,cr_
call_center_sk,cr_catalog_page_sk,cr_ship_mode_sk,cr_warehouse_sk,cr_reason_s
k,cr_order_number,cr_return_quantity,cr_return_amount,cr_return_tax,cr_return
_amt_inc_tax,cr_fee,cr_return_ship_cost,cr_refunded_cash,cr_reversed_charge,c
r_store_credit,cr_net_loss);

load data infile '/mnt/external/data/catalog_sales.dat'
into table catalog_sales fields terminated by '|'
(cs_sold_date_sk,cs_sold_time_sk,cs_ship_date_sk,cs_bill_customer_sk,cs_bill_
cdemo_sk,cs_bill_hdemo_sk,cs_bill_addr_sk,cs_ship_customer_sk,cs_ship_cdemo_s
k,cs_ship_hdemo_sk,cs_ship_addr_sk,cs_call_center_sk,cs_catalog_page_sk,cs_sh
ip_mode_sk,cs_warehouse_sk,cs_item_sk,cs_promo_sk,cs_order_number,cs_quantity
,cs_wholesale_cost,cs_list_price,cs_sales_price,cs_ext_discount_amt,cs_ext_sa
les_price,cs_ext_wholesale_cost,cs_ext_list_price,cs_ext_tax,cs_coupon_amt,cs
_ext_ship_cost,cs_net_paid,cs_net_paid_inc_tax,cs_net_paid_inc_ship,cs_net_pa
id_inc_ship_tax,cs_net_profit);

load data infile '/mnt/external/data/customer.dat'
into table customer fields terminated by '|'
c_customer_sk,c_customer_id,c_current_cdemo_sk,c_current_hdemo_sk,c_current_a
ddr_sk,c_first_shipto_date_sk,c_first_sales_date_sk,c_salutation,c_first_name
,c_last_name,c_preferred_cust_flag,c_birth_day,c_birth_month,c_birth_year,c_b
irth_country,c_login,c_email_address,c_last_review_date);

load data infile '/mnt/external/data/customer_address.dat'
```

```
into table customer_address fields terminated by '|'
(ca_address_sk,ca_address_id,ca_street_number,ca_street_name,ca_street_type,c
a_suite_number,ca_city,ca_county,ca_state,ca_zip,ca_country,ca_gmt_offset,ca_
location_type);

load data infile '/mnt/external/data/customer_demographics.dat'
into table customer_demographics fields terminated by '|'
(cd_demo_sk,cd_gender,cd_marital_status,cd_education_status,cd_purchase_estim
ate,cd_credit_rating,cd_dep_count,cd_dep_employed_count,cd_dep_college_count)
;

load data infile '/mnt/external/data/date_dim.dat'
into table date_dim fields terminated by '|'
(d_date_sk,d_date_id,d_date,d_month_seq,d_week_seq,d_quarter_seq,d_year,d_dow
,d_moy,d_dom,d_qoy,d_fy_year,d_fy_quarter_seq,d_fy_week_seq,d_day_name,d_quar
ter_name,d_holiday,d_weekend,d_following_holiday,d_first_dom,d_last_dom,d_sam
e_day_ly,d_same_day_lq,d_current_day,d_current_week,d_current_month,d_current
_quarter,d_current_year);

load data infile '/mnt/external/data/dbgen_version.dat'
into table dbgen_version fields terminated by '|'
(dv_version,dv_create_date,dv_create_time,dv_cmdline_args);

load data infile '/mnt/external/data/household_demographics.dat'
into table household_demographics fields terminated by '|'
(hd_demo_sk,hd_income_band_sk,hd_buy_potential,hd_dep_count,hd_vehicle_count)
;

load data infile '/mnt/external/data/income_band.dat'
into table income_band fields terminated by '|'
(ib_income_band_sk,ib_lower_bound,ib_upper_bound);

load data infile '/mnt/external/data/inventory.dat'
into table inventory fields terminated by '|'
(inv_date_sk,inv_item_sk,inv_warehouse_sk,inv_quantity_on_hand);

load data infile '/mnt/external/data/item.dat'
into table item fields terminated by '|'
(i_item_sk,i_item_id,i_rec_start_date,i_rec_end_date,i_item_desc,i_current_pr
ice,i_wholesale_cost,i_brand_id,i_brand,i_class_id,i_class,i_category_id,i_ca
tegory,i_manufact_id,i_manufact,i_size,i_formulation,i_color,i_units,i_contai
ner,i_manager_id,i_product_name);

load data infile '/mnt/external/data/promotion.dat'
into table promotion fields terminated by '|'
(p_promo_sk,p_promo_id,p_start_date_sk,p_end_date_sk,p_item_sk,p_cost,p_respo
nse_target,p_promo_name,p_channel_dmail,p_channel_email,p_channel_catalog,p_c
hannel_tv,p_channel_radio,p_channel_press,p_channel_event,p_channel_demo,p_ch
annel_details,p_purpose,p_discount_active);

load data infile '/mnt/external/data/reason.dat'
into table reason fields terminated by '|'
(r_reason_sk,r_reason_id, r_reason_desc);

load data infile '/mnt/external/data/ship_mode.dat'
into table ship_mode fields terminated by '|'
(sm_ship_mode_sk,sm_ship_mode_id,sm_type,sm_code,sm_carrier,sm_contract);
```

```
load data infile '/mnt/external/data/store.dat'
into table store fields terminated by '|'
(s_store_sk,s_store_id,s_rec_start_date,s_rec_end_date,s_closed_date_sk,s_sto
re_name,s_number_employees,s_floor_space,s_hours,s_manager,s_market_id,s_geog
raphy_class,s_market_desc,s_market_manager,s_division_id,s_division_name,s_co
mpany_id,s_company_name,s_street_number,s_street_name,s_street_type,s_suite_n
umber,s_city,s_county,s_state,s_zip,s_country,s_gmt_offset,s_tax_precentage);

load data infile '/mnt/external/data/store_returns.dat'
into table store_returns fields terminated by '|'
(sr_returned_date_sk,sr_return_time_sk,sr_item_sk,sr_customer_sk,sr_cdemo_sk,
sr_hdemo_sk,sr_addr_sk,sr_store_sk,sr_reason_sk,sr_ticket_number,sr_return_qu
antity,sr_return_amt,sr_return_tax,sr_return_amt_inc_tax,sr_fee,sr_return_shi
p_cost,sr_refunded_cash,sr_reversed_charge,sr_store_credit,sr_net_loss);

load data infile '/mnt/external/data/store_sales.dat'
into table store_sales fields terminated by '|'
(ss_sold_date_sk,ss_sold_time_sk,ss_item_sk,ss_customer_sk,ss_cdemo_sk,ss_hde
mo_sk,ss_addr_sk,ss_store_sk,ss_promo_sk,ss_ticket_number,ss_quantity,ss_whol
esale_cost,ss_list_price,ss_sales_price,ss_ext_discount_amt,ss_ext_sales_pric
e,ss_ext_wholesale_cost,ss_ext_list_price,ss_ext_tax,ss_coupon_amt,ss_net_pai
d,ss_net_paid_inc_tax,ss_net_profit);

load data infile '/mnt/external/data/time_dim.dat'
into table time_dim fields terminated by '|'
(t_time_sk,t_time_id,t_time,t_hour,t_minute,t_second,t_am_pm,t_shift,t_sub_sh
ift,t_meal_time);

load data infile '/mnt/external/data/warehouse.dat'
into table warehouse fields terminated by '|'
(w_warehouse_sk,w_warehouse_id,w_warehouse_name,w_warehouse_sq_ft,w_street_nu
mber,w_street_name,w_street_type,w_suite_number,w_city,w_county,w_state,w_zip
,w_country,w_gmt_offset);

load data infile '/mnt/external/data/web_page.dat'
into table web_page fields terminated by '|'
(wp_web_page_sk,wp_web_page_id,wp_rec_start_date,wp_rec_end_date,wp_creation_
date_sk,wp_access_date_sk,wp_autogen_flag,wp_customer_sk,wp_url,wp_type,wp_ch
ar_count,wp_link_count,wp_image_count,wp_max_ad_count);

load data infile '/mnt/external/data/web_returns.dat'
into table web_returns fields terminated by '|'
(wr_returned_date_sk,wr_returned_time_sk,wr_item_sk,wr_refunded_customer_sk,w
r_refunded_cdemo_sk,wr_refunded_hdemo_sk,wr_refunded_addr_sk,wr_returning_cus
tomer_sk,wr_returning_cdemo_sk,wr_returning_hdemo_sk,wr_returning_addr_sk,wr_
web_page_sk,wr_reason_sk,wr_order_number,wr_return_quantity,wr_return_amt,wr_
return_tax,wr_return_amt_inc_tax,wr_fee,wr_return_ship_cost,wr_refunded_cash,
wr_reversed_charge,wr_account_credit,wr_net_loss);

load data infile '/mnt/external/data/web_sales.dat'
into table web_sales fields terminated by '|'
(ws_sold_date_sk,ws_sold_time_sk,ws_ship_date_sk,ws_item_sk,ws_bill_customer_
sk,ws_bill_cdemo_sk,ws_bill_hdemo_sk,ws_bill_addr_sk,ws_ship_customer_sk,ws_s
hip_cdemo_sk,ws_ship_hdemo_sk,ws_ship_addr_sk,ws_web_page_sk,ws_web_site_sk,w
s_ship_mode_sk,ws_warehouse_sk,ws_promo_sk,ws_order_number,ws_quantity,ws_who
lesale_cost,ws_list_price,ws_sales_price,ws_ext_discount_amt,ws_ext_sales_pri
```

```
ce,ws_ext_wholesale_cost,ws_ext_list_price,ws_ext_tax,ws_coupon_amt,ws_ext_sh
ip_cost,ws_net_paid,ws_net_paid_inc_tax,ws_net_paid_inc_ship,ws_net_paid_inc_
ship_tax,ws_net_profit);

load data infile '/mnt/external/data/web_site.dat'
into table web_site fields terminated by '|'
(web_site_sk,web_site_id,web_rec_start_date,web_rec_end_date,web_name,web_ope
n_date_sk,web_close_date_sk,web_class,web_manager,web_mkt_id,web_mkt_class,we
b_mkt_desc,web_market_manager,web_company_id,web_company_name,web_street_numb
er,web_street_name,web_street_type,web_suite_number,web_city,web_county,web_s
tate,web_zip,web_country,web_gmt_offset,web_tax_percentage);

alter table call_center add constraint cc_d1 foreign key  (cc_closed_date_sk)
references date_dim (d_date_sk);
alter table call_center add constraint cc_d2 foreign key  (cc_open_date_sk)
references date_dim (d_date_sk);
alter table catalog_page add constraint cp_d1 foreign key  (cp_end_date_sk)
references date_dim (d_date_sk);
alter table catalog_page add constraint cp_p foreign key  (cp_promo_id)
references promotion (p_promo_sk);
alter table catalog_page add constraint cp_d2 foreign key  (cp_start_date_sk)
references date_dim (d_date_sk);
alter table catalog_returns add constraint cr_cc foreign key
(cr_call_center_sk) references call_center (cc_call_center_sk);
alter table catalog_returns add constraint cr_cp foreign key
(cr_catalog_page_sk) references catalog_page (cp_catalog_page_sk);
alter table catalog_returns add constraint cr_i foreign key  (cr_item_sk)
references item (i_item_sk);
alter table catalog_returns add constraint cr_r foreign key  (cr_reason_sk)
references reason (r_reason_sk);
alter table catalog_returns add constraint cr_a1 foreign key
(cr_refunded_addr_sk) references customer_address (ca_address_sk);
alter table catalog_returns add constraint cr_cd1 foreign key
(cr_refunded_cdemo_sk) references customer_demographics (cd_demo_sk);
alter table catalog_returns add constraint cr_c1 foreign key
(cr_refunded_customer_sk) references customer (c_customer_sk);
alter table catalog_returns add constraint cr_hd1 foreign key
(cr_refunded_hdemo_sk) references household_demographics (hd_demo_sk);
alter table catalog_returns add constraint cr_d1 foreign key
(cr_returned_date_sk) references date_dim (d_date_sk);
alter table catalog_returns add constraint cr_i foreign key
(cr_returned_time_sk) references time_dim (t_time_sk);
alter table catalog_returns add constraint cr_a2 foreign key
(cr_returning_addr_sk) references customer_address (ca_address_sk);
alter table catalog_returns add constraint cr_cd2 foreign key
(cr_returning_cdemo_sk) references customer_demographics (cd_demo_sk);
alter table catalog_returns add constraint cr_c2 foreign key
(cr_returning_customer_sk) references customer (c_customer_sk);
alter table catalog_returns add constraint cr_hd2 foreign key
(cr_returning_hdemo_sk) references household_demographics (hd_demo_sk);
alter table catalog_returns add constraint cr_d2 foreign key
(cr_ship_date_sk) references date_dim (d_date_sk);
alter table catalog_returns add constraint cr_sm foreign key
(cr_ship_mode_sk) references ship_mode (sm_ship_mode_sk);
alter table catalog_returns add constraint cr_w2 foreign key
(cr_warehouse_sk) references warehouse (w_warehouse_sk);
```

```
alter table catalog_sales add constraint cs_b_a foreign key
(cs_bill_addr_sk) references customer_address (ca_address_sk);
alter table catalog_sales add constraint cs_b_cd foreign key
(cs_bill_cdemo_sk) references customer_demographics (cd_demo_sk);
alter table catalog_sales add constraint cs_b_c foreign key
(cs_bill_customer_sk) references customer (c_customer_sk);
alter table catalog_sales add constraint cs_b_hd foreign key
(cs_bill_hdemo_sk) references household_demographics (hd_demo_sk);
alter table catalog_sales add constraint cs_cc foreign key
(cs_call_center_sk) references call_center (cc_call_center_sk);
alter table catalog_sales add constraint cs_cp foreign key
(cs_catalog_page_sk) references catalog_page (cp_catalog_page_sk);
alter table catalog_sales add constraint cs_i foreign key  (cs_item_sk)
references item (i_item_sk);
alter table catalog_sales add constraint cs_p foreign key  (cs_promo_sk)
references promotion (p_promo_sk);
alter table catalog_sales add constraint cs_s_a foreign key
(cs_ship_addr_sk) references customer_address (ca_address_sk);
alter table catalog_sales add constraint cs_s_cd foreign key
(cs_ship_cdemo_sk) references customer_demographics (cd_demo_sk);
alter table catalog_sales add constraint cs_s_c foreign key
(cs_ship_customer_sk) references customer (c_customer_sk);
alter table catalog_sales add constraint cs_d1 foreign key  (cs_ship_date_sk)
references date_dim (d_date_sk);
alter table catalog_sales add constraint cs_s_hd foreign key
(cs_ship_hdemo_sk) references household_demographics (hd_demo_sk);
alter table catalog_sales add constraint cs_sm foreign key  (cs_ship_mode_sk)
references ship_mode (sm_ship_mode_sk);
alter table catalog_sales add constraint cs_d2 foreign key  (cs_sold_date_sk)
references date_dim (d_date_sk);
alter table catalog_sales add constraint cs_t foreign key  (cs_sold_time_sk)
references time_dim (t_time_sk);
alter table catalog_sales add constraint cs_w foreign key  (cs_warehouse_sk)
references warehouse (w_warehouse_sk);
alter table customer add constraint c_a foreign key  (c_current_addr_sk)
references customer_address (ca_address_sk);
alter table customer add constraint c_cd foreign key  (c_current_cdemo_sk)
references customer_demographics (cd_demo_sk);
alter table customer add constraint c_hd foreign key  (c_current_hdemo_sk)
references household_demographics (hd_demo_sk);
alter table customer add constraint c_fsd foreign key
(c_first_sales_date_sk) references date_dim (d_date_sk);
alter table customer add constraint c_fsd2 foreign key
(c_first_shipto_date_sk) references date_dim (d_date_sk);
alter table household_demographics add constraint hd_ib foreign key
(hd_income_band_sk) references income_band (ib_income_band_sk);
alter table inventory add constraint inv_d foreign key  (inv_date_sk)
references date_dim (d_date_sk);
alter table inventory add constraint inv_i foreign key  (inv_item_sk)
references item (i_item_sk);
alter table inventory add constraint inv_w foreign key  (inv_warehouse_sk)
references warehouse (w_warehouse_sk);
alter table promotion add constraint p_end_date foreign key  (p_end_date_sk)
references date_dim (d_date_sk);
alter table promotion add constraint p_i foreign key  (p_item_sk) references
item (i_item_sk);
```

```
alter table promotion add constraint p_start_date foreign key
(p_start_date_sk) references date_dim (d_date_sk);
alter table store add constraint s_close_date foreign key  (s_closed_date_sk)
references date_dim (d_date_sk);
alter table store_returns add constraint sr_a foreign key  (sr_addr_sk)
references customer_address (ca_address_sk);
alter table store_returns add constraint sr_cd foreign key  (sr_cdemo_sk)
references customer_demographics (cd_demo_sk);
alter table store_returns add constraint sr_c foreign key  (sr_customer_sk)
references customer (c_customer_sk);
alter table store_returns add constraint sr_hd foreign key  (sr_hdemo_sk)
references household_demographics (hd_demo_sk);
alter table store_returns add constraint sr_i foreign key  (sr_item_sk)
references item (i_item_sk);
alter table store_returns add constraint sr_r foreign key  (sr_reason_sk)
references reason (r_reason_sk);
alter table store_returns add constraint sr_ret_d foreign key
(sr_returned_date_sk) references date_dim (d_date_sk);
alter table store_returns add constraint sr_t foreign key
(sr_return_time_sk) references time_dim (t_time_sk);
alter table store_returns add constraint sr_s foreign key  (sr_store_sk)
references store (s_store_sk);
alter table store_sales add constraint ss_a foreign key  (ss_addr_sk)
references customer_address (ca_address_sk);
alter table store_sales add constraint ss_cd foreign key  (ss_cdemo_sk)
references customer_demographics (cd_demo_sk);
alter table store_sales add constraint ss_c foreign key  (ss_customer_sk)
references customer (c_customer_sk);
alter table store_sales add constraint ss_hd foreign key  (ss_hdemo_sk)
references household_demographics (hd_demo_sk);
alter table store_sales add constraint ss_i foreign key  (ss_item_sk)
references item (i_item_sk);
alter table store_sales add constraint ss_p foreign key  (ss_promo_sk)
references promotion (p_promo_sk);
alter table store_sales add constraint ss_d foreign key  (ss_sold_date_sk)
references date_dim (d_date_sk);
alter table store_sales add constraint ss_t foreign key  (ss_sold_time_sk)
references time_dim (t_time_sk);
alter table store_sales add constraint ss_s foreign key  (ss_store_sk)
references store (s_store_sk);
alter table web_page add constraint wp_ad foreign key  (wp_access_date_sk)
references date_dim (d_date_sk);
alter table web_page add constraint wp_cd foreign key  (wp_creation_date_sk)
references date_dim (d_date_sk);
alter table web_returns add constraint wr_i foreign key  (wr_item_sk)
references item (i_item_sk);
alter table web_returns add constraint wr_r foreign key  (wr_reason_sk)
references reason (r_reason_sk);
alter table web_returns add constraint wr_ref_a foreign key
(wr_refunded_addr_sk) references customer_address (ca_address_sk);
alter table web_returns add constraint wr_ref_cd foreign key
(wr_refunded_cdemo_sk) references customer_demographics (cd_demo_sk);
alter table web_returns add constraint wr_ref_c foreign key
(wr_refunded_customer_sk) references customer (c_customer_sk);
alter table web_returns add constraint wr_ref_hd foreign key
(wr_refunded_hdemo_sk) references household_demographics (hd_demo_sk);
```

```
alter table web_returns add constraint wr_ret_d foreign key
(wr_returned_date_sk) references date_dim (d_date_sk);
alter table web_returns add constraint wr_ret_t foreign key
(wr_returned_time_sk) references time_dim (t_time_sk);
alter table web_returns add constraint wr_ret_a foreign key
(wr_returning_addr_sk) references customer_address (ca_address_sk);
alter table web_returns add constraint wr_ret_cd foreign key
(wr_returning_cdemo_sk) references customer_demographics (cd_demo_sk);
alter table web_returns add constraint wr_ret_c foreign key
(wr_returning_customer_sk) references customer (c_customer_sk);
alter table web_returns add constraint wr_ret_cd foreign key
(wr_returning_hdemo_sk) references household_demographics (hd_demo_sk);
alter table web_returns add constraint wr_wp foreign key  (wr_web_page_sk)
references web_page (wp_web_page_sk);
alter table web_sales add constraint ws_b_a foreign key  (ws_bill_addr_sk)
references customer_address (ca_address_sk);
alter table web_sales add constraint ws_b_cd foreign key  (ws_bill_cdemo_sk)
references customer_demographics (cd_demo_sk);
alter table web_sales add constraint ws_b_c foreign key
(ws_bill_customer_sk) references customer (c_customer_sk);
alter table web_sales add constraint ws_b_cd foreign key  (ws_bill_hdemo_sk)
references household_demographics (hd_demo_sk);
alter table web_sales add constraint ws_i foreign key  (ws_item_sk)
references item (i_item_sk);
alter table web_sales add constraint ws_p foreign key  (ws_promo_sk)
references promotion (p_promo_sk);
alter table web_sales add constraint ws_s_a foreign key  (ws_ship_addr_sk)
references customer_address (ca_address_sk);
alter table web_sales add constraint ws_s_cd foreign key  (ws_ship_cdemo_sk)
references customer_demographics (cd_demo_sk);
alter table web_sales add constraint ws_s_c foreign key
(ws_ship_customer_sk) references customer (c_customer_sk);
alter table web_sales add constraint ws_s_d foreign key  (ws_ship_date_sk)
references date_dim (d_date_sk);
alter table web_sales add constraint ws_s_hd foreign key  (ws_ship_hdemo_sk)
references household_demographics (hd_demo_sk);
alter table web_sales add constraint ws_sm foreign key  (ws_ship_mode_sk)
references ship_mode (sm_ship_mode_sk);
alter table web_sales add constraint ws_d2 foreign key  (ws_sold_date_sk)
references date_dim (d_date_sk);
alter table web_sales add constraint ws_t foreign key  (ws_sold_time_sk)
references time_dim (t_time_sk);
alter table web_sales add constraint ws_w2 foreign key  (ws_warehouse_sk)
references warehouse (w_warehouse_sk);
alter table web_sales add constraint ws_wp foreign key  (ws_web_page_sk)
references web_page (wp_web_page_sk);
alter table web_sales add constraint ws_ws foreign key  (ws_web_site_sk)
references web_site (web_site_sk);
alter table web_site add constraint web_d1 foreign key  (web_close_date_sk)
references date_dim (d_date_sk);
alter table web_site add constraint web_d2 foreign key  (web_open_date_sk)
references date_dim (d_date_sk);
```

APPENDIX B

Physical Representation of TPC-DS OLAP Schema

```
<Schema name="TPCDS">
  <Dimension type="StandardDimension" highCardinality="false" name="Date">
    <Hierarchy name="Date" hasAll="true" primaryKey="d_date_sk">
      <Table name="date_dim">
      </Table>
      <Level name="Year" column="d_year" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Quarter" column="d_qoy" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Month" column="d_moy" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Date" column="d_date" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
    <Hierarchy name="Day" hasAll="true" primaryKey="d_date_sk">
      <Table name="date_dim">
      </Table>
      <Level name="DayName" column="d_day_name" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" highCardinality="false" name="Item">
    <Hierarchy name="Manufacturer" hasAll="true" primaryKey="i_item_sk">
      <Table name="item">
      </Table>
      <Level name="Manufacturer" table="item" column="i_manufact_id"
type="Integer" uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
        <Property name="Manufacturer Name" column="i_manufact" type="String">
        </Property>
      </Level>
      <Level name="Brand" table="item" column="i_brand_id" type="Integer"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        <Property name="Brand Name" column="i_brand" type="String">
        </Property>
      </Level>
    </Hierarchy>
    <Hierarchy name="Category" hasAll="true" primaryKey="i_item_sk">
      <Table name="item">
      </Table>
      <Level name="Category" table="item" column="i_category_id"
type="Integer" uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
        <Property name="Category Name" column="i_category" type="String">
        </Property>
      </Level>
      <Level name="Class" table="item" column="i_class_id" type="Integer"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        <Property name="Class Name" column="i_class" type="String">
```

```
        </Property>
      </Level>
    </Hierarchy>
    <Hierarchy name="Manager" hasAll="true" primaryKey="i_item_sk">
      <Table name="item">
      </Table>
      <Level name="Manager ID" table="item" column="i_manager_id"
type="Integer" uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
    <Hierarchy name="Item Info" hasAll="true" primaryKey="i_item_sk">
      <Table name="item">
      </Table>
      <Level name="Item ID" table="item" column="i_item_id" type="String"
uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
        <Property name="Description" column="i_item_desc" type="String">
        </Property>
        <Property name="Color" column="i_color" type="String">
        </Property>
        <Property name="Current Price" column="i_current_price"
type="Numeric">
        </Property>
      </Level>
      <Level name="Key" table="item" column="i_item_sk" type="String"
uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" highCardinality="false" name="Store">
    <Hierarchy name="Store" hasAll="true" primaryKey="s_store_sk">
      <Table name="store">
      </Table>
      <Level name="Store ID" table="store" column="s_store_id" type="String"
uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Store Name" table="store" column="s_store_name"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
    <Hierarchy name="Location" hasAll="true" primaryKey="s_store_sk">
      <Table name="store">
      </Table>
      <Level name="State" table="store" column="s_state" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Zip" table="store" column="s_zip" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="GMT Offset" table="store" column="s_gmt_offset"
type="Numeric" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" highCardinality="false"
name="Customer">
    <Hierarchy name="Customer" hasAll="true" primaryKey="c_customer_sk">
```

```
      <Table name="customer">
      </Table>
      <Level name="ID" table="customer" column="c_customer_id" type="String"
uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
    <Hierarchy name="Address" hasAll="true" primaryKey="c_current_addr_sk"
primaryKeyTable="customer">
      <Join leftKey="c_current_addr_sk" rightKey="ca_address_sk">
        <Table name="customer">
        </Table>
        <Table name="customer_address">
        </Table>
      </Join>
      <Level name="State" table="customer_address" column="ca_state"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="City" table="customer_address" column="ca_city"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Zip" table="customer_address" column="ca_zip"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Street Name" table="customer_address"
column="ca_street_name" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Steet Number" table="customer_address"
column="ca_street_number" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="GMT Offset" table="customer_address"
column="ca_gmt_offset" type="Numeric" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
    <Hierarchy name="Customer Demographics" hasAll="true"
primaryKey="c_customer_sk" primaryKeyTable="customer">
      <Join leftKey="c_current_cdemo_sk" rightKey="cd_demo_sk">
        <Table name="customer">
        </Table>
        <Table name="customer_demographics">
        </Table>
      </Join>
      <Level name="Gender" table="customer_demographics" column="cd_gender"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Marital Status" table="customer_demographics"
column="cd_marital_status" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Education Status" table="customer_demographics"
column="cd_education_status" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
```

```
    <Hierarchy name="Household Demographics" hasAll="true"
primaryKey="c_customer_sk" primaryKeyTable="customer">
      <Join leftKey="c_current_hdemo_sk" rightKey="hd_demo_sk">
        <Table name="customer">
        </Table>
        <Table name="household_demographics">
        </Table>
      </Join>
      <Level name="Buy Potential" table="household_demographics"
column="hd_buy_potential" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
    <Hierarchy name="First Sales Date" hasAll="true"
primaryKey="c_customer_sk" primaryKeyTable="customer">
      <Join leftKey="c_first_sales_date_sk" rightKey="d_date_sk">
        <Table name="customer">
        </Table>
        <Table name="date_dim">
        </Table>
      </Join>
      <Level name="Year" table="date_dim" column="d_year" type="Integer"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
    <Hierarchy name="First ShipTo Date" hasAll="true"
primaryKey="c_customer_sk" primaryKeyTable="customer">
      <Join leftKey="c_first_shipto_date_sk" rightKey="d_date_sk">
        <Table name="customer">
        </Table>
        <Table name="date_dim">
        </Table>
      </Join>
      <Level name="Year" table="date_dim" column="d_year" type="Integer"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" highCardinality="false" name="Customer
Demographics">
    <Hierarchy name="Demographics" hasAll="true" primaryKey="cd_demo_sk">
      <Table name="customer_demographics">
      </Table>
      <Level name="Gender" table="customer_demographics" column="cd_gender"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Marital Status" table="customer_demographics"
column="cd_marital_status" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Education Status" table="customer_demographics"
column="cd_education_status" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
```

```
  <Dimension type="StandardDimension" highCardinality="false" name="Household
Demographics">
    <Hierarchy name="Demographics" hasAll="true" primaryKey="hd_demo_sk">
      <Table name="household_demographics">
      </Table>
      <Level name="Dependents" table="household_demographics"
column="hd_dep_count" type="Integer" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Buy Potential" table="household_demographics"
column="hd_buy_potential" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" highCardinality="false"
name="Promotion">
    <Hierarchy name="Channel" hasAll="true" primaryKey="p_promo_sk">
      <Table name="promotion">
      </Table>
      <Level name="TV" table="promotion" column="p_channel_tv" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Event" table="promotion" column="p_channel_event"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Email" table="promotion" column="p_channel_email"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
    </Hierarchy>
  </Dimension>
  <Dimension type="StandardDimension" highCardinality="false" name="Customer
Address">
    <Hierarchy name="Address" hasAll="true" primaryKey="ca_address_sk">
      <Table name="customer_address">
      </Table>
      <Level name="State" table="customer_address" column="ca_state"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="City" table="customer_address" column="ca_city"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Zip" table="customer_address" column="ca_zip"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Street Name" table="customer_address"
column="ca_street_name" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="Street Number" table="customer_address"
column="ca_street_number" type="String" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
      <Level name="GMT Offset" table="customer_address"
column="ca_gmt_offset" type="Numeric" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
      </Level>
```

```
    </Hierarchy>
  </Dimension>
  <Cube name="Store Sales" cache="true" enabled="true">
    <Table name="store_sales">
    </Table>
    <DimensionUsage source="Item" name="Item" foreignKey="ss_item_sk"
highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Date" name="Date" foreignKey="ss_sold_date_sk"
highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Store" name="Store" foreignKey="ss_store_sk"
highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Customer" name="Customer"
foreignKey="ss_customer_sk" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Household Demographics" name="Household
Demographics" foreignKey="ss_hdemo_sk" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Customer Demographics" name="Customer
Demographics" foreignKey="ss_cdemo_sk" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Promotion" name="Promotion"
foreignKey="ss_promo_sk" highCardinality="false">
    </DimensionUsage>
    <Dimension type="StandardDimension" foreignKey="ss_sold_time_sk"
highCardinality="false" name="Time">
      <Hierarchy name="Time" hasAll="true" primaryKey="t_time_sk">
        <Table name="time_dim">
        </Table>
        <Level name="Hour" table="time_dim" column="t_hour" type="Integer"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="Minute" table="time_dim" column="t_minute"
type="Integer" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Measure name="Total Extended Price" column="ss_ext_sales_price"
aggregator="sum" visible="true">
    </Measure>
    <Measure name="Total Sales Price" column="ss_sales_price"
aggregator="sum" visible="true">
    </Measure>
    <Measure name="Average Quantity" column="ss_quantity" aggregator="avg"
visible="true">
    </Measure>
    <Measure name="Average List Price" column="ss_list_price"
aggregator="avg" visible="true">
    </Measure>
    <Measure name="Average Coupon Amount" column="ss_coupon_amt"
aggregator="avg" visible="true">
    </Measure>
    <Measure name="Average Sales Price" column="ss_sales_price"
aggregator="avg" visible="true">
```

```
    </Measure>
    <Measure name="Total Net Profit" column="ss_net_profit" aggregator="sum"
visible="true">
    </Measure>
    <Measure name="Total Wholesale Cost" column="ss_wholesale_cost"
aggregator="sum" visible="true">
    </Measure>
    <Measure name="Total Coupon Amount" column="ss_coupon_amt"
aggregator="sum" visible="true">
    </Measure>
    <Measure name="Total List Price" column="ss_list_price" aggregator="sum"
visible="true">
    </Measure>
    <Measure name="Count" column="ss_item_sk" aggregator="distinct-count"
visible="true">
    </Measure>
    <CalculatedMember name="Gross Margin" formula="[Measures].[Total Net
Profit] / [Measures].[Total Extended Price]" dimension="Measures"
visible="true">
    </CalculatedMember>
  </Cube>
  <Cube name="Web Sales" cache="true" enabled="true">
    <Table name="web_sales">
    </Table>
    <Dimension type="StandardDimension" foreignKey="ws_web_site_sk"
highCardinality="false" name="Web Site">
      <Hierarchy name="Web Site" hasAll="true" primaryKey="web_site_sk">
        <Table name="web_site">
        </Table>
        <Level name="Company" column="web_company_name" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="ID" table="web_site" column="web_site_id" type="String"
uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <DimensionUsage source="Date" name="Date" foreignKey="ws_sold_date_sk"
highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Item" name="Item" foreignKey="ws_item_sk"
highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Customer" name="Billed Customer"
foreignKey="ws_bill_customer_sk" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Customer Address" name="Delivery Customer"
foreignKey="ws_ship_addr_sk" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Promotion" name="Promotion"
foreignKey="ws_promo_sk" highCardinality="false">
    </DimensionUsage>
    <Measure name="Sales Price" column="ws_sales_price" aggregator="sum"
visible="true">
    </Measure>
    <Measure name="Order Count" column="ws_order_number"
aggregator="distinct-count" visible="true">
```

```
    </Measure>
    <Measure name="Shipping Cost" column="ws_ext_ship_cost" aggregator="sum"
visible="true">
    </Measure>
    <Measure name="Net Profit" column="ws_net_profit" aggregator="sum"
visible="true">
    </Measure>
    <Measure name="Net Paid" column="ws_net_paid" aggregator="sum"
visible="true">
    </Measure>
  </Cube>
  <Cube name="Store Returns" cache="true" enabled="true">
    <Table name="store_returns">
    </Table>
    <DimensionUsage source="Date" name="Date"
foreignKey="sr_returned_date_sk" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Item" name="Item" foreignKey="sr_item_sk"
highCardinality="false">
    </DimensionUsage>
    <Dimension type="StandardDimension" foreignKey="sr_item_sk"
highCardinality="false" name="Return">
      <Hierarchy name="Info" hasAll="true" primaryKey="sr_item_sk">
        <Table name="store_returns">
        </Table>
        <Level name="Customer Key" table="store_returns"
column="sr_customer_sk" type="Integer" uniqueMembers="false"
levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="Store Key" table="store_returns" column="sr_store_sk"
type="Integer" uniqueMembers="false" levelType="Regular"
hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <DimensionUsage source="Store" name="Store" foreignKey="sr_store_sk">
    </DimensionUsage>
    <DimensionUsage source="Customer" name="Customer"
foreignKey="sr_customer_sk">
    </DimensionUsage>
    <Measure name="Total Net Loss" column="sr_net_loss" aggregator="sum"
visible="true">
    </Measure>
    <Measure name="Total Return Amount" column="sr_return_amt"
aggregator="sum" visible="true">
    </Measure>
  </Cube>
  <Cube name="Catalog Sales" cache="true" enabled="true">
    <Table name="catalog_sales">
    </Table>
    <DimensionUsage source="Item" name="Item" foreignKey="cs_item_sk"
highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Promotion" name="Promotion"
foreignKey="cs_promo_sk" highCardinality="false">
    </DimensionUsage>
```

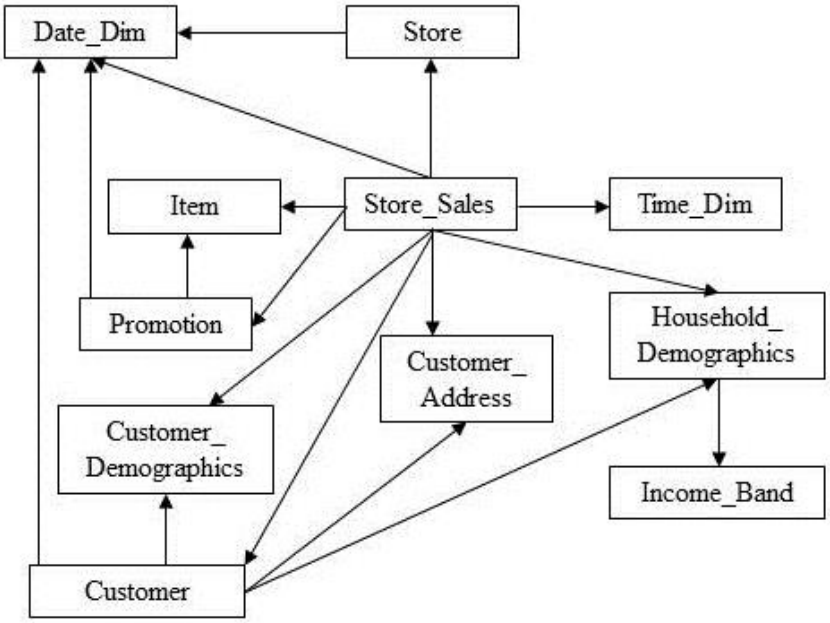
```
    <DimensionUsage source="Date" name="Sold Date"
foreignKey="cs_sold_date_sk" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Customer Address" name="Billing Address"
foreignKey="cs_bill_addr_sk" highCardinality="false">
    </DimensionUsage>
    <Dimension type="StandardDimension" foreignKey="cs_catalog_page_sk"
highCardinality="false" name="Catalog Page">
      <Hierarchy name="Catalog Page" hasAll="true"
primaryKey="cp_catalog_page_sk">
        <Table name="catalog_page">
        </Table>
        <Level name="ID" table="catalog_page" column="cp_catalog_page_id"
type="String" uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <Measure name="Total Net Profit" column="cs_net_profit" aggregator="sum"
visible="true">
    </Measure>
    <Measure name="Total Extended Price" column="cs_ext_sales_price"
aggregator="sum" visible="true">
    </Measure>
    <Measure name="Total Extended Discount" column="cs_ext_discount_amt"
aggregator="sum" visible="true">
    </Measure>
    <Measure name="Total Quantity" column="cs_quantity" aggregator="sum"
visible="true">
    </Measure>
    <Measure name="Total Wholesale Cost" column="cs_wholesale_cost"
aggregator="sum" visible="true">
    </Measure>
    <Measure name="Total Sales Price" column="cs_sales_price"
aggregator="sum" visible="true">
    </Measure>
  </Cube>
  <Cube name="Catalog Returns" cache="true" enabled="true">
    <Table name="catalog_returns">
    </Table>
    <Dimension type="StandardDimension" foreignKey="cr_catalog_page_sk"
highCardinality="false" name="Catalog Page">
      <Hierarchy name="Catalog Page" hasAll="true"
primaryKey="cp_catalog_page_sk">
        <Table name="catalog_page">
        </Table>
        <Level name="ID" column="cp_catalog_page_id" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <DimensionUsage source="Item" name="Item" foreignKey="cr_item_sk"
highCardinality="false">
    </DimensionUsage>
    <Dimension type="StandardDimension" foreignKey="cr_call_center_sk"
highCardinality="false" name="Call Center">
      <Hierarchy name="Call Center" hasAll="true"
primaryKey="cc_call_center_sk">
```

```
        <Table name="call_center">
        </Table>
        <Level name="ID" table="call_center" column="cc_call_center_id"
type="String" uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="Name" table="call_center" column="cc_name" type="String"
uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
        <Level name="Manager" table="call_center" column="cc_manager"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <DimensionUsage source="Date" name="Returned Date"
foreignKey="cr_returned_date_sk" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Customer" name="Returning Customer"
foreignKey="cr_returning_customer_sk" highCardinality="false">
    </DimensionUsage>
    <Measure name="Total Net Loss" column="cr_net_loss" aggregator="sum"
visible="true">
    </Measure>
  </Cube>
  <Cube name="Web Returns" cache="true" enabled="true">
    <Table name="web_returns">
    </Table>
    <Dimension type="StandardDimension" foreignKey="wr_web_page_sk"
highCardinality="false" name="Web Page">
      <Hierarchy name="Web Page" hasAll="true" primaryKey="wp_web_page_sk">
        <Table name="web_page">
        </Table>
        <Level name="ID" table="web_page" column="wp_web_page_id"
type="String" uniqueMembers="true" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <DimensionUsage source="Item" name="Item" foreignKey="wr_item_sk"
highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Customer Demographics" name="Customer
Demographics" foreignKey="wr_refunded_cdemo_sk" highCardinality="false">
    </DimensionUsage>
    <DimensionUsage source="Date" name="Date"
foreignKey="wr_returned_date_sk" highCardinality="false">
    </DimensionUsage>
    <Measure name="Net Loss" column="wr_net_loss" aggregator="sum"
visible="true">
    </Measure>
  </Cube>
  <Cube name="Inventory" cache="true" enabled="true">
    <Table name="inventory">
    </Table>
    <DimensionUsage source="Date" name="Date" foreignKey="inv_date_sk"
highCardinality="false">
    </DimensionUsage>
    <Dimension type="StandardDimension" foreignKey="inv_warehouse_sk"
highCardinality="false" name="Warehouse">
```

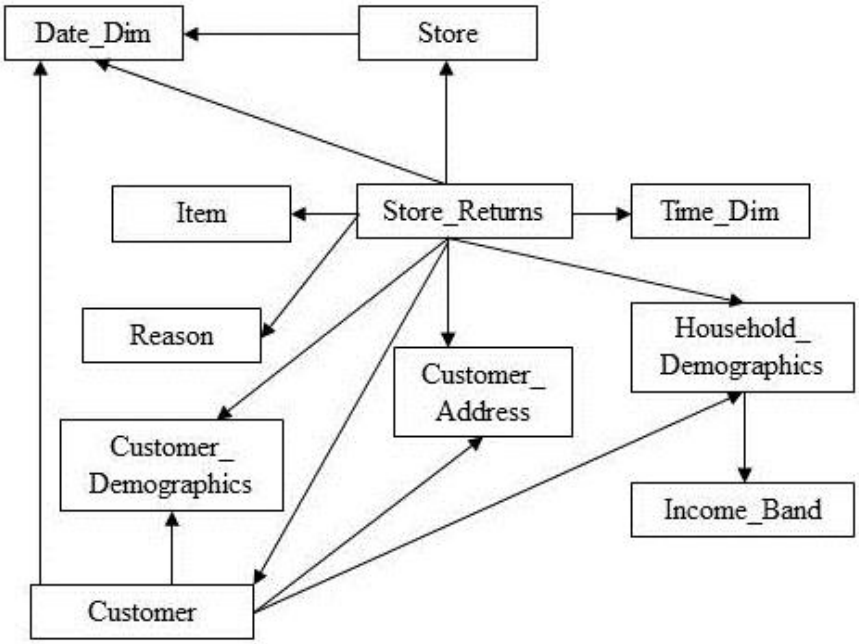
```
      <Hierarchy name="Warehouse" hasAll="true" primaryKey="w_warehouse_sk">
        <Table name="warehouse">
        </Table>
        <Level name="Name" table="warehouse" column="w_warehouse_name"
type="String" uniqueMembers="false" levelType="Regular" hideMemberIf="Never">
        </Level>
      </Hierarchy>
    </Dimension>
    <DimensionUsage source="Item" name="Item" foreignKey="inv_item_sk"
highCardinality="false">
    </DimensionUsage>
    <Measure name="Quantity" column="inv_quantity_on_hand"
aggregator="distinct-count" visible="true">
    </Measure>
  </Cube>
  <VirtualCube enabled="true" name="Web">
    <VirtualCubeDimension cubeName="Web Sales" highCardinality="false"
name="Date">
    </VirtualCubeDimension>
    <VirtualCubeDimension cubeName="Web Returns" highCardinality="false"
name="Web Page">
    </VirtualCubeDimension>
    <VirtualCubeDimension cubeName="Web Sales" highCardinality="false"
name="Item">
    </VirtualCubeDimension>
    <VirtualCubeMeasure cubeName="Web Sales" name="[Measures].[Net Profit]"
visible="true">
    </VirtualCubeMeasure>
    <VirtualCubeMeasure cubeName="Web Returns" name="[Measures].[Net Loss]"
visible="true">
    </VirtualCubeMeasure>
    <CalculatedMember name="Profit" formula="[Measures].[Net Profit] -
[Measures].[Net Loss]" dimension="Measures" visible="true">
    </CalculatedMember>
  </VirtualCube>
</Schema>
```

APPENDIX C

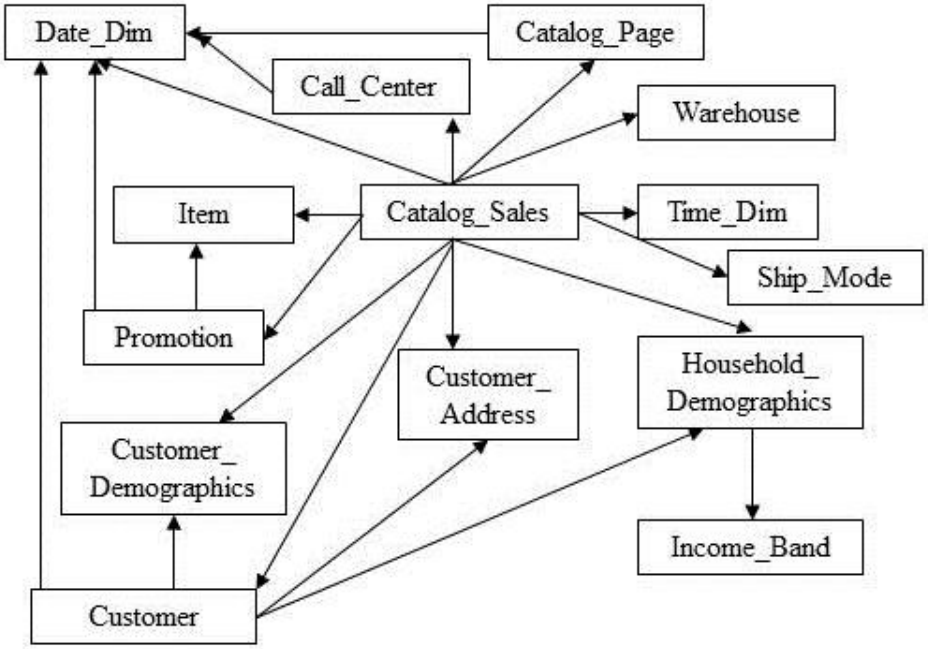
Cube Diagrams



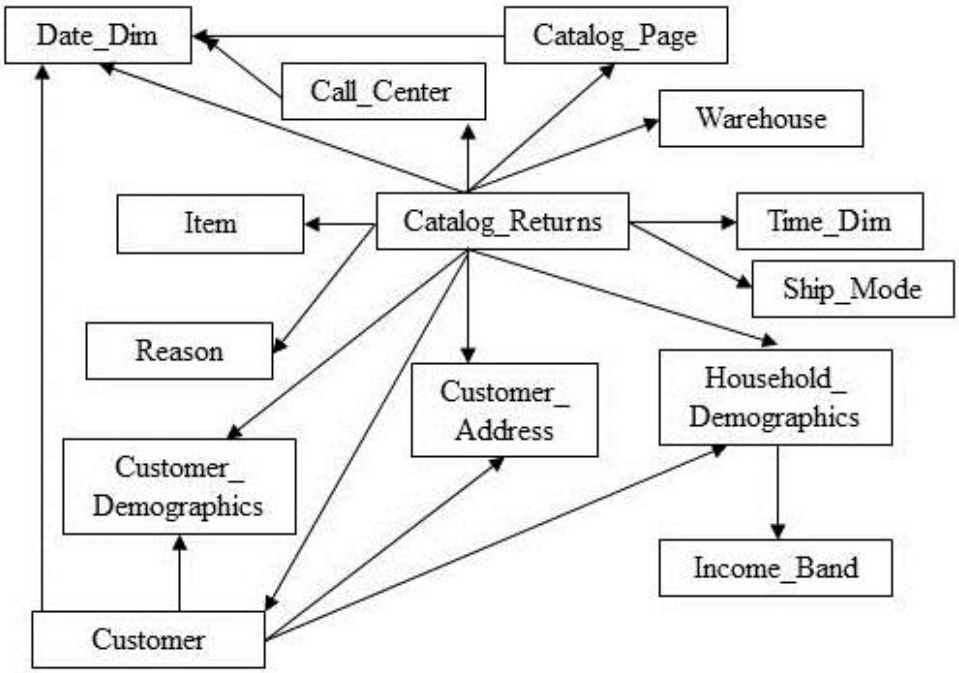
Store Sales



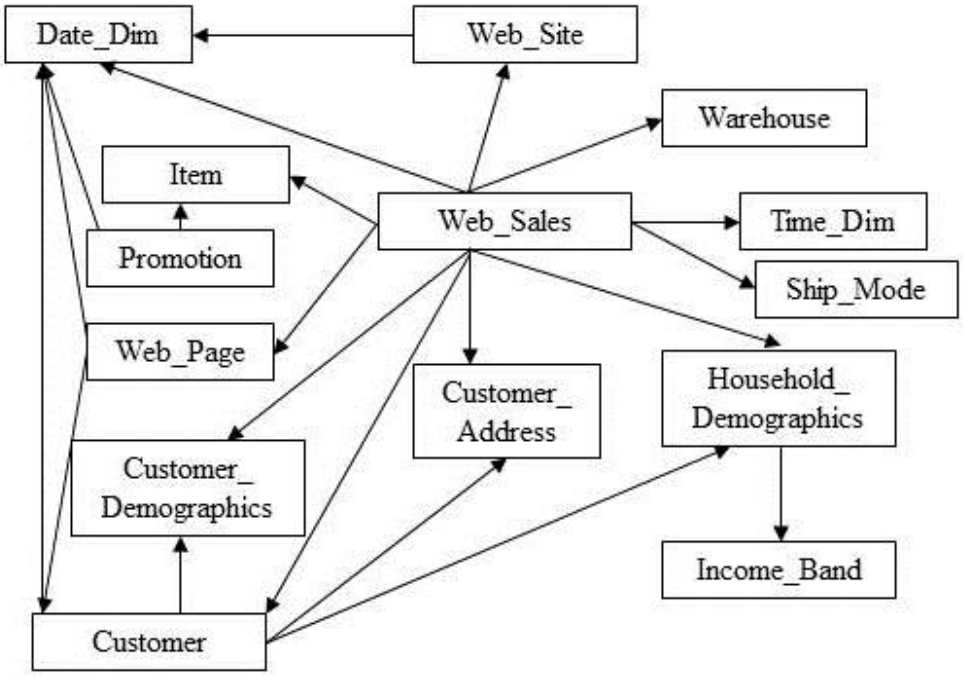
Store Returns



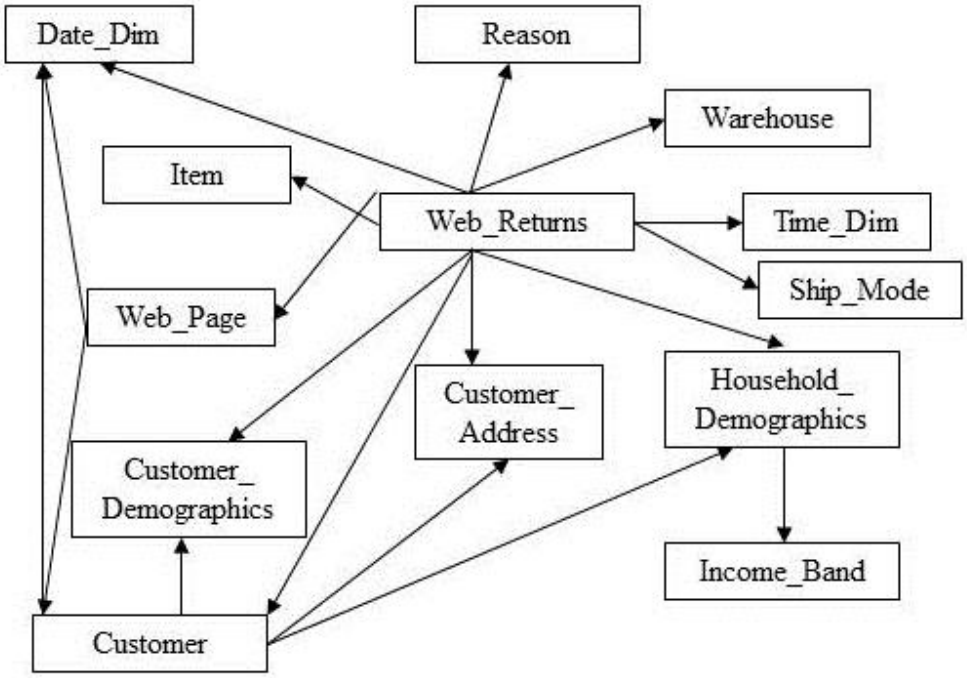
Catalog Sales



Catalog Returns



Web Sales



Web Returns

APPENDIX D

Server Configuration

```
## Install Debian 6 (Squeeze)

## APT
############  /etc/apt/sources.list  ############
deb http://ftp.us.debian.org/debian/ squeeze main non-free contrib
deb http://security.debian.org/ squeeze/updates main non-free contrib
############  /etc/apt/sources.list  ############
apt-get update

## SSH (optional)
apt-get install ssh
############  /etc/ssh/ssh_config  ########################
ForwardAgent yes
ForwardX11 yes
############  /etc/ssh/ssh_config  ########################
/etc/init.d/ssh reload
update-rc.d ssh defaults

## JAVA
apt-get install sun-java6-jdk
echo 'JAVA_HOME="/usr/lib/jvm/java-6-sun"' >> /etc/environment
echo 'JRE_HOME="/usr/lib/jvm/java-6-sun/jre"' >> /etc/environment
shutdown -r now

## Tomcat
apt-get install tomcat6 tomcat6-admin
/etc/init.d/tomcat6 stop
mkdir /opt/tomcat
cd /opt/tomcat
ln -s /etc/tomcat6/ conf
ln -s /usr/share/tomcat6/bin/ bin
ln -s /usr/share/tomcat6/lib/ lib
ln -s /var/lib/tomcat6/bin/ bin
ln -s /var/log/tomcat6/ logs

############  /opt/tomcat/conf/tomcat-users.xml  ############
<user username="toccig" password="tpass1" roles="admin,manager"/>
############  /opt/tomcat/conf/tomcat-users.xml  ############

netstat -pan | grep ":80\|:443"
/etc/init.d/tomcat6 restart
netstat -pan | grep ":80\|:443"

## MYSQL Install
apt-get install mysql-server-5.1

############  /etc/mysql/my.cnf  ############
character_set_server=utf8
collation_server=utf8_general_ci
############  /etc/mysql/my.cnf  ############
/etc/init.d/mysql restart
```

```
## TPC-DS Data Load
cd /opt
wget http://www.tpc.org/tpcds/tpcds_20080121.zip
mkdir tpc-ds
unzip tpcds_20080121.zip -d tpc-ds
rm tpcds_20080121.zip
cd tpc-ds
cp Makefile.suite Makefile
vi Makefile
######## /opt/Makefile ##########
OS = LINUX
######## /opt/Makefile ##########
make -f Makefile
mkdir data
./dsdgen -scale 100 -dir data
mysql -u root -p
mysql> create database tpcds;
mysql> use tpcds;
mysql> source tpcds.sql;
mysql> set session BULK_INSERT_BUFFER_SIZE=1073741824;
mysql> set session MYISAM_SORT_BUFFER_SIZE=1073741824;
mysql> set global KEY_BUFFER_SIZE=1073741824;
mysql> source tpcdsDataload.sql;
mysql> source tpcds_ri.sql;
mysql> exit;

## Mondrian Install
cd /opt/
wget sourceforge.net/projects/mondrian/files/mondrian/mondrian-
3.2.1.13885/mondrian-3.2.1.13885.zip
unzip mondrian-3.2.1.13885.zip
mv mondrian-3.2.1.13885 mondrian
rm mondrian-3.2.1.13885.zip

## Mysql Jdbc connector
cd /tmp
wget -O mysql-connector-java-5.1.18.tar.gz
http://www.mysql.com/get/Downloads/Connector-J/mysql-connector-java-
5.1.18.tar.gz/from/http://mysql.he.net/
tar -xvf mysql-connector-java-5.1.18.tar.gz
cp mysql-connector-java-5.1.18-bin.jar /opt/mondrian/lib/
rm -r mysql-connector*

## log4j
wget http://apache.pesat.net.id/logging/log4j/1.2.15/apache-log4j-
1.2.15.tar.gz
tzr -xvf apache-log4j-1.2.16.tar.gz
cp apache-log4j-1.2.16/log4j-1.2.16.jar /opt/mondrian/lib/
mv /opt/mondrian/lib/log4j-1.2.16.jar /opt/mondrian/lib/log4j.jar
rm -r log4j-1.2.16
rm apache-log4j-1.2.16.tar.gz

## eigenbase
wget http://www.java2s.com/Code/JarDownload/eigenbase-xom-1.3.0.11999.jar.zip
wget  http://www.java2s.com/Code/JarDownload/eigenbase/eigenbase-
resgen.jar.zip
wget http://www.java2s.com/Code/JarDownload/eigenbase-properties.jar.zip
```

```
unzip eigenbase-xom-1.3.0.11999.jar.zip
unzip eigenbase-resgen.jar.zip
unzip eigenbase-properties.jar.zip
cp eigenbase-xom-1.3.0.11999.jar /opt/mondrian/lib/
mv /opt/mondrian/lib/eigenbase-xom-1.3.0.11999.jar
/opt/mondrian/lib/eigenbase-xom.jar
cp eigenbase-resgen.jar /opt/mondrian/lib/
cp eigenbase-properties.jar /opt/mondrian/lib/
rm eig*

## Mondrian Configuration
mkdir /opt/tomcat/webapps/mondrian/
cd /opt/tomcat/webapps/mondrian/
unzip /opt/mondrian/lib/mondrian.war
cd /opt/tomcat6/webapps/mondrian/WEB-INF/

vi queries/tpcds.xml
###### /opt/tomcat6/webapps/mondrian/WEB-INF/queries/tpcds.xml #######
see appendix B
###### /opt/tomcat6/webapps/mondrian/WEB-INF/queries/tpcds.xml #######

vi datasources.xml
######## /opt/tomcat6/webapps/mondrian/WEB-INF/datasources.xml ##########
<DataSourceName>Provider=Mondrian;DataSource=tpcds;</DataSourceName>
<DataSourceDescription>Mondrian TPCDS Benchmark Data
Warehouse</DataSourceDescription>
<URL>http://localhost:8080/mondrian/xmla</URL>
<Catalog name="tpcds">
<DataSourceInfo>Provider=mondrian;Jdbc=jdbc:mysql://localhost/tpcds;JdbcUser=
root;JdbcPassword=rpass1;JdbcDrivers=com.mysql.jdbc.Driver;Catalog=/WEB-
INF/queries/tpcds.xml</DataSourceInfo>
######## /opt/tomcat6/webapps/mondrian/WEB-INF/datasources.xml ##########

vi mondrian.properties
####### /opt/tomcat/webapps/mondrian/WEB-INF/mondrian.properties #######
mondrian.test.connectString=
Provider=mondrian;Jdbc=jdbc:mysql://localhost/tpcds;JdbcUser=root;JdbcPasswor
d=rpass1;JdbcDrivers=com.mysql.jdbc.Driver;Catalog=/WEB-INF/queries/tpcds.xml
####### /opt/tomcat/webapps/mondrian/WEB-INF/mondrian.properties #######

cp /opt/tomcat/webapps/mondrian/WEB-INF/mondrian.properties
/opt/tomcat/webapps/mondrian/mondrian.properties

vi web.xml
############  /opt/tomcat/webapps/mondrian/WEB-INF/web.xml  ############
  <context-param>
    <param-name>connectString</param-name>
    <param-
value>mondrian.test.connectString=Provider=mondrian;Jdbc=jdbc:mysql://localho
st/tpcds?user=root&amp;password=rpass;JdbcDrivers=com.mysql.jdbc.Driver;Catal
og=/WEB-INF/queries/tpcds.xml;</param-value>
  </context-param>
############  /opt/tomcat/webapps/mondrian/WEB-INF/web.xml  ############

#test in browser
http://192.168.1.115:8080/mondrian
```

APPENDIX E

JMeter Test Plan


```
<?xml version="1.0" encoding="UTF-8"?>
<jmeterTestPlan version="1.2" properties="2.2">
  <hashTree>
    <TestPlan guiclass="TestPlanGui" testclass="TestPlan" testname="Test
Plan" enabled="true">
      <stringProp name="TestPlan.comments"></stringProp>
      <boolProp name="TestPlan.functional_mode">false</boolProp>
      <boolProp name="TestPlan.serialize_threadgroups">false</boolProp>
      <elementProp name="TestPlan.user_defined_variables"
elementType="Arguments" guiclass="ArgumentsPanel" testclass="Arguments"
testname="User Defined Variables" enabled="true">
        <collectionProp name="Arguments.arguments"/>
      </elementProp>
      <stringProp name="TestPlan.user_define_classpath">C:\Program Files
(x86)\JMeter\jakarta-jmeter-2.4\lib\mysql-connector-java-5.1.13-
bin.jar</stringProp>
    </TestPlan>
    <hashTree>
      <ThreadGroup guiclass="ThreadGroupGui" testclass="ThreadGroup"
testname="Thread Group" enabled="true">
        <stringProp name="ThreadGroup.on_sample_error">continue</stringProp>
        <elementProp name="ThreadGroup.main_controller"
elementType="LoopController" guiclass="LoopControlPanel"
testclass="LoopController" testname="Loop Controller" enabled="true">
          <boolProp name="LoopController.continue_forever">false</boolProp>
          <stringProp name="LoopController.loops">1</stringProp>
        </elementProp>
        <stringProp name="ThreadGroup.num_threads">1</stringProp>
        <stringProp name="ThreadGroup.ramp_time">1</stringProp>
        <longProp name="ThreadGroup.start_time">1289681623000</longProp>
        <longProp name="ThreadGroup.end_time">1289681623000</longProp>
        <boolProp name="ThreadGroup.scheduler">false</boolProp>
        <stringProp name="ThreadGroup.duration"></stringProp>
        <stringProp name="ThreadGroup.delay"></stringProp>
      </ThreadGroup>
      <hashTree>
        <ResultCollector guiclass="SummaryReport" testclass="ResultCollector"
testname="Summary Report" enabled="true">
          <boolProp name="ResultCollector.error_logging">false</boolProp>
          <objProp>
            <name>saveConfig</name>
            <value class="SampleSaveConfiguration">
              <time>true</time>
              <latency>true</latency>
              <timestamp>true</timestamp>
              <success>true</success>
              <label>true</label>
              <code>true</code>
              <message>true</message>
              <threadName>true</threadName>
              <dataType>true</dataType>
              <encoding>false</encoding>
```

```
              <assertions>true</assertions>
              <subresults>true</subresults>
              <responseData>false</responseData>
              <samplerData>false</samplerData>
              <xml>false</xml>
              <fieldNames>false</fieldNames>
              <responseHeaders>false</responseHeaders>
              <requestHeaders>false</requestHeaders>
              <responseDataOnError>false</responseDataOnError>

<saveAssertionResultsFailureMessage>false</saveAssertionResultsFailureMessage
>
              <assertionsResultsToSave>0</assertionsResultsToSave>
              <bytes>true</bytes>
            </value>
          </objProp>
          <stringProp name="filename"></stringProp>
        </ResultCollector>
        <hashTree/>
        <ResultCollector guiclass="ViewResultsFullVisualizer"
testclass="ResultCollector" testname="View Results Tree" enabled="true">
          <boolProp name="ResultCollector.error_logging">false</boolProp>
          <objProp>
            <name>saveConfig</name>
            <value class="SampleSaveConfiguration">
              <time>true</time>
              <latency>true</latency>
              <timestamp>true</timestamp>
              <success>true</success>
              <label>true</label>
              <code>true</code>
              <message>true</message>
              <threadName>true</threadName>
              <dataType>true</dataType>
              <encoding>false</encoding>
              <assertions>true</assertions>
              <subresults>true</subresults>
              <responseData>false</responseData>
              <samplerData>false</samplerData>
              <xml>true</xml>
              <fieldNames>false</fieldNames>
              <responseHeaders>false</responseHeaders>
              <requestHeaders>false</requestHeaders>
              <responseDataOnError>false</responseDataOnError>

<saveAssertionResultsFailureMessage>false</saveAssertionResultsFailureMessage
>
              <assertionsResultsToSave>0</assertionsResultsToSave>
              <bytes>true</bytes>
            </value>
          </objProp>
          <stringProp name="filename"></stringProp>
        </ResultCollector>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
```

```
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
Select [Measures].[Net Loss] on 0, [Item.Manufacturer].Members on 1, {[Web
Page].[AAAAAAAAACAAAAAA], [Web Page].[AAAAAAAABAAAAAAA]} on 2 from [Web
Returns]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
select [Measures].[Count] on 0, [Store.Location].[TN] on 1 from [Store Sales]
where { [Household Demographics].[1], [Household Demographics].[2],
[Household Demographics].[3] }
&lt;/Statement&gt;
```

```
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
Select [Measures].[Profit] on 0,  NonEmptyCrossJoin({[Web
Page].CurrentMember},{[Item.Category].[5].Children,
[Item.Category].[7].Children, [Item.Category].[10].Children}) on 1 from [Web]
where ([Date].[2000])
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
```

```
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
Select [Measures].[Total Net Loss] on 0, Filter({[Call Center].Members},
[Call Center].[Manager].CurrentMember.Name = &apos;Larry Mccray&apos; OR
[Call Center].[Manager].CurrentMember.Name = &apos;Mark Hightower&apos;) on
1, {[Item.Category].[7].Children, [Item.Category].[9].Children} on 2  from
[Catalog Returns] where [Returned Date].[2002]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
```

```
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
With Member [Measures].[Item Color] as &apos;[Item.Item
Info].CurrentMember.Properties(&quot;Color&quot;)&apos;
Member [Measures].[Item Description] as &apos;[Item.Item
Info].CurrentMember.Properties(&quot;Description&quot;)&apos;
Select {[Measures].[Quantity], [Measures].[Item Color], [Measures].[Item
Description]} on 0, {[Item.Item Info].[AAAAAAAAAAABAAAA], [Item.Item
Info].[AAAAAAAAAABDAAAA], [Item.Item Info].[AAAAAAAAAADEAAAA], [Item.Item
Info].[AAAAAAAAOENAAAAA]} on 1 from [Inventory]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
Select [Measures].[Total Sales Price] on 0, CrossJoin({[Sold
Date].[2000].Children, [Sold Date].[2001].Children}, [Promotion].Members) on
1 from [Catalog Sales] where [Item.Manager].[11]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
```

```
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Tabular&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
With Member [Measures].[Month Subtotal] as &apos;Sum([Date].[Month].Members,
[Measures].[Total Sales Price])&apos;
Select [Measures].[Month Subtotal] on 0, [Store].Members on 1 from [Store
Sales] where [Item.Manager].[6]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
```

```
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
select  {[Measures].[Total Extended Price]} on columns,
Order([Item.Manager].[97], [Measures].[Total Extended Price], DESC) on rows
from [Store Sales] where [Date].[2002].[4].[12]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
```

```
    &lt;Statement&gt;
Select [Measures].[Net Loss] on 0, [Customer Demographics].[F].Children on 1,
{[Date].[2002].[1].[1], [Date].[2002].[1].[2], [Date].[2002].[1].[3]} on 2
from [Web Returns]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
Select [Measures].[Total Quantity] on 0,
CrossJoin({[Item.Category].[7],[Item.Category].[10]},
[Promotion].[Email].Members) on 1 from [Catalog Sales]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Tabular&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
```

```
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
select [Measures].[Total Net Loss] on 0,
NonEmptyCrossJoin({[Item.Manufacturer].[995], [Item.Manufacturer].[117]},
[Store.Location].[TN]) on 1 from [Store Returns] where
[Customer.Address].[TN]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
        <SoapSampler guiclass="SoapSamplerGui" testclass="SoapSampler"
testname="SOAP/XML-RPC Request" enabled="true">
          <elementProp name="HTTPsampler.Arguments" elementType="Arguments">
            <collectionProp name="Arguments.arguments"/>
          </elementProp>
          <stringProp
name="SoapSampler.URL_DATA">http://192.168.1.115:8080/mondrian/xmla</stringPr
op>
```

```
          <stringProp name="HTTPSamper.xml_data">&lt;soapenv:Envelope
xmlns:soapenv=&quot;http://schemas.xmlsoap.org/soap/envelope/&quot;
xmlns:xsd=&quot;http://www.w3.org/2001/XMLSchema&quot;
xmlns:xsi=&quot;http://www.w3.org/2001/XMLSchema-instance&quot;&gt;
&lt;soapenv:Body&gt;
&lt;Execute xmlns=&quot;urn:schemas-microsoft-com:xml-analysis&quot;&gt;
  &lt;Command&gt;
    &lt;Statement&gt;
Select [Measures].[Order Count] on 0,
NonEmptyCrossJoin({[Item.Category].[1],[Item.Category].[3],
[Item.Category].[6]}, [Promotion].[Email].Members) on 1 from [Web Sales]
&lt;/Statement&gt;
  &lt;/Command&gt;
  &lt;Properties&gt;
    &lt;PropertyList&gt;
      &lt;Catalog&gt;tpcds&lt;/Catalog&gt;

&lt;DataSourceInfo&gt;Provider=Mondrian;DataSource=tpcds;&lt;/DataSourceInfo&
gt;
      &lt;Format&gt;Multidimensional&lt;/Format&gt;
      &lt;AxisFormat&gt;TupleFormat&lt;/AxisFormat&gt;
    &lt;/PropertyList&gt;
  &lt;/Properties&gt;
&lt;/Execute&gt;
        &lt;/soapenv:Body&gt;
  &lt;/soapenv:Envelope&gt;</stringProp>
          <stringProp name="SoapSampler.xml_data_file"></stringProp>
          <stringProp name="SoapSampler.SOAP_ACTION"></stringProp>
          <stringProp name="SoapSampler.SEND_SOAP_ACTION">true</stringProp>
          <boolProp name="HTTPSampler.use_keepalive">false</boolProp>
        </SoapSampler>
        <hashTree/>
      </hashTree>
    </hashTree>
  </hashTree>
</jmeterTestPlan>
```

APPENDIX F

TPC-DS Benchmark Results

MyISAM Power Test Run 1

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	42:55.2	User Session 1-1	Query 1	2753	Success	645
2	42:58.0	User Session 1-1	Query 2	27	Success	13277
3	42:58.0	User Session 1-1	Query 3	293	Success	509132
4	42:58.3	User Session 1-1	Query 4	296	Success	475202
5	42:58.6	User Session 1-1	Query 5	250	Success	376014
6	42:58.8	User Session 1-1	Query 6	21	Success	16047
7	42:58.8	User Session 1-1	Query 7	28	Success	17119
8	42:58.9	User Session 1-1	Query 8	19	Success	14580
9	42:58.9	User Session 1-1	Query 9	6	Success	729
10	42:58.9	User Session 1-1	Query 1	2647	Success	645
11	43:01.6	User Session 1-1	Query 2	21	Success	13277
12	43:01.6	User Session 1-1	Query 3	308	Success	509132
13	43:01.9	User Session 1-1	Query 4	284	Success	475202
14	43:02.2	User Session 1-1	Query 5	226	Success	376014
15	43:02.4	User Session 1-1	Query 6	22	Success	16047
16	43:02.4	User Session 1-1	Query 7	34	Success	17119
17	43:02.5	User Session 1-1	Query 8	21	Success	14580
18	43:02.5	User Session 1-1	Query 9	6	Success	729
19	43:02.5	User Session 1-1	Query 1	2644	Success	645
20	43:05.1	User Session 1-1	Query 2	21	Success	13277
21	43:05.2	User Session 1-1	Query 3	295	Success	509132
22	43:05.5	User Session 1-1	Query 4	289	Success	475202
23	43:05.7	User Session 1-1	Query 5	236	Success	376014
24	43:06.0	User Session 1-1	Query 6	23	Success	16047
25	43:06.0	User Session 1-1	Query 7	27	Success	17119
26	43:06.0	User Session 1-1	Query 8	20	Success	14580
27	43:06.1	User Session 1-1	Query 9	7	Success	729
28	43:06.1	User Session 1-1	Query 1	2656	Success	645
29	43:08.7	User Session 1-1	Query 2	20	Success	13277
30	43:08.7	User Session 1-1	Query 3	306	Success	509132
31	43:09.0	User Session 1-1	Query 4	300	Success	475202
32	43:09.4	User Session 1-1	Query 5	227	Success	376014
33	43:09.6	User Session 1-1	Query 6	22	Success	16047
34	43:09.6	User Session 1-1	Query 7	29	Success	17119

35	43:09.6	User Session 1-1	Query 8	20	Success	14580
36	43:09.7	User Session 1-1	Query 9	7	Success	729
37	43:09.7	User Session 1-1	Query 1	2885	Success	645
38	43:12.5	User Session 1-1	Query 2	33	Success	13277
39	43:12.6	User Session 1-1	Query 3	293	Success	509132
40	43:12.9	User Session 1-1	Query 4	293	Success	475202
41	43:13.2	User Session 1-1	Query 5	230	Success	376014
42	43:13.4	User Session 1-1	Query 6	22	Success	16047
43	43:13.4	User Session 1-1	Query 7	27	Success	17119
44	43:13.5	User Session 1-1	Query 8	20	Success	14580
45	43:13.5	User Session 1-1	Query 9	6	Success	729
46	43:13.5	User Session 1-1	Query 1	2723	Success	645
47	43:16.2	User Session 1-1	Query 2	20	Success	13277
48	43:16.2	User Session 1-1	Query 3	317	Success	509132
49	43:16.5	User Session 1-1	Query 4	283	Success	475202
50	43:16.8	User Session 1-1	Query 5	235	Success	376014
51	43:17.1	User Session 1-1	Query 6	21	Success	16047
52	43:17.1	User Session 1-1	Query 7	27	Success	17119
53	43:17.1	User Session 1-1	Query 8	19	Success	14580
54	43:17.1	User Session 1-1	Query 9	14	Success	729
55	43:17.1	User Session 1-1	Query 1	2693	Success	645
56	43:19.8	User Session 1-1	Query 2	27	Success	13277
57	43:19.9	User Session 1-1	Query 3	301	Success	509132
58	43:20.2	User Session 1-1	Query 4	280	Success	475202
59	43:20.5	User Session 1-1	Query 5	233	Success	376014
60	43:20.7	User Session 1-1	Query 6	20	Success	16047
61	43:20.7	User Session 1-1	Query 7	28	Success	17119
62	43:20.7	User Session 1-1	Query 8	18	Success	14580
63	43:20.8	User Session 1-1	Query 9	6	Success	729
64	43:20.8	User Session 1-1	Query 1	2657	Success	645
65	43:23.4	User Session 1-1	Query 2	29	Success	13277
66	43:23.5	User Session 1-1	Query 3	314	Success	509132
67	43:23.8	User Session 1-1	Query 4	284	Success	475202
68	43:24.1	User Session 1-1	Query 5	225	Success	376014
69	43:24.3	User Session 1-1	Query 6	21	Success	16047
70	43:24.3	User Session 1-1	Query 7	36	Success	17119
71	43:24.3	User Session 1-1	Query 8	18	Success	14580
72	43:24.4	User Session 1-1	Query 9	8	Success	729
73	43:24.4	User Session 1-1	Query 1	2617	Success	645
74	43:27.0	User Session 1-1	Query 2	38	Success	13277
75	43:27.0	User Session 1-1	Query 3	454	Success	509132

76	43:27.5	User Session 1-1	Query 4	422	Success	475202
77	43:27.9	User Session 1-1	Query 5	338	Success	376014
78	43:28.3	User Session 1-1	Query 6	28	Success	16047
79	43:28.3	User Session 1-1	Query 7	32	Success	17119
80	43:28.3	User Session 1-1	Query 8	25	Success	14580
81	43:28.3	User Session 1-1	Query 9	7	Success	729
82	43:28.4	User Session 1-1	Query 1	2858	Success	645
83	43:31.2	User Session 1-1	Query 2	27	Success	13277
84	43:31.2	User Session 1-1	Query 3	453	Success	509132
85	43:31.7	User Session 1-1	Query 4	435	Success	475202
86	43:32.1	User Session 1-1	Query 5	344	Success	376014
87	43:32.5	User Session 1-1	Query 6	25	Success	16047
88	43:32.5	User Session 1-1	Query 7	33	Success	17119
89	43:32.5	User Session 1-1	Query 8	34	Success	14580
90	43:32.6	User Session 1-1	Query 9	8	Success	729
91	43:32.6	User Session 1-1	Query 1	2601	Success	645
92	43:35.2	User Session 1-1	Query 2	21	Success	13277
93	43:35.2	User Session 1-1	Query 3	297	Success	509132
94	43:35.5	User Session 1-1	Query 4	287	Success	475202
95	43:35.8	User Session 1-1	Query 5	219	Success	376014
96	43:36.0	User Session 1-1	Query 6	20	Success	16047
97	43:36.0	User Session 1-1	Query 7	33	Success	17119
98	43:36.1	User Session 1-1	Query 8	20	Success	14580
99	43:36.1	User Session 1-1	Query 9	7	Success	729

MyISAM Power Test Run 2

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	44:46.8	User Session 1-1	Query 1	2595	Success	645
2	44:49.4	User Session 1-1	Query 2	20	Success	13277
3	44:49.5	User Session 1-1	Query 3	306	Success	509132
4	44:49.8	User Session 1-1	Query 4	283	Success	475202
5	44:50.0	User Session 1-1	Query 5	229	Success	376014
6	44:50.3	User Session 1-1	Query 6	20	Success	16047
7	44:50.3	User Session 1-1	Query 7	26	Success	17119
8	44:50.3	User Session 1-1	Query 8	18	Success	14580
9	44:50.3	User Session 1-1	Query 9	7	Success	729
10	44:50.4	User Session 1-1	Query 1	2623	Success	645
11	44:53.0	User Session 1-1	Query 2	26	Success	13277
12	44:53.0	User Session 1-1	Query 3	292	Success	509132
13	44:53.3	User Session 1-1	Query 4	283	Success	475202

14	44:53.6	User Session 1-1	Query 5	223	Success	376014
15	44:53.8	User Session 1-1	Query 6	25	Success	16047
16	44:53.8	User Session 1-1	Query 7	26	Success	17119
17	44:53.9	User Session 1-1	Query 8	18	Success	14580
18	44:53.9	User Session 1-1	Query 9	6	Success	729
19	44:53.9	User Session 1-1	Query 1	2630	Success	645
20	44:56.5	User Session 1-1	Query 2	18	Success	13277
21	44:56.5	User Session 1-1	Query 3	293	Success	509132
22	44:56.8	User Session 1-1	Query 4	272	Success	475202
23	44:57.1	User Session 1-1	Query 5	231	Success	376014
24	44:57.3	User Session 1-1	Query 6	22	Success	16047
25	44:57.4	User Session 1-1	Query 7	26	Success	17119
26	44:57.4	User Session 1-1	Query 8	25	Success	14580
27	44:57.4	User Session 1-1	Query 9	7	Success	729
28	44:57.4	User Session 1-1	Query 1	2692	Success	645
29	45:00.1	User Session 1-1	Query 2	18	Success	13277
30	45:00.1	User Session 1-1	Query 3	289	Success	509132
31	45:00.4	User Session 1-1	Query 4	274	Success	475202
32	45:00.7	User Session 1-1	Query 5	241	Success	376014
33	45:00.9	User Session 1-1	Query 6	19	Success	16047
34	45:01.0	User Session 1-1	Query 7	47	Success	17119
35	45:01.0	User Session 1-1	Query 8	20	Success	14580
36	45:01.0	User Session 1-1	Query 9	6	Success	729
37	45:01.1	User Session 1-1	Query 1	2618	Success	645
38	45:03.7	User Session 1-1	Query 2	19	Success	13277
39	45:03.7	User Session 1-1	Query 3	309	Success	509132
40	45:04.0	User Session 1-1	Query 4	278	Success	475202
41	45:04.3	User Session 1-1	Query 5	230	Success	376014
42	45:04.5	User Session 1-1	Query 6	18	Success	16047
43	45:04.5	User Session 1-1	Query 7	24	Success	17119
44	45:04.6	User Session 1-1	Query 8	19	Success	14580
45	45:04.6	User Session 1-1	Query 9	6	Success	729
46	45:04.6	User Session 1-1	Query 1	2706	Success	645
47	45:07.3	User Session 1-1	Query 2	25	Success	13277
48	45:07.3	User Session 1-1	Query 3	310	Success	509132
49	45:07.6	User Session 1-1	Query 4	282	Success	475202
50	45:07.9	User Session 1-1	Query 5	231	Success	376014
51	45:08.1	User Session 1-1	Query 6	21	Success	16047
52	45:08.2	User Session 1-1	Query 7	26	Success	17119
53	45:08.2	User Session 1-1	Query 8	26	Success	14580
54	45:08.2	User Session 1-1	Query 9	7	Success	729

55	45:08.2	User Session 1-1	Query 1	2612	Success	645
56	45:10.8	User Session 1-1	Query 2	26	Success	13277
57	45:10.9	User Session 1-1	Query 3	323	Success	509132
58	45:11.2	User Session 1-1	Query 4	273	Success	475202
59	45:11.5	User Session 1-1	Query 5	236	Success	376014
60	45:11.7	User Session 1-1	Query 6	22	Success	16047
61	45:11.7	User Session 1-1	Query 7	29	Success	17119
62	45:11.8	User Session 1-1	Query 8	18	Success	14580
63	45:11.8	User Session 1-1	Query 9	7	Success	729
64	45:11.8	User Session 1-1	Query 1	2611	Success	645
65	45:14.4	User Session 1-1	Query 2	18	Success	13277
66	45:14.4	User Session 1-1	Query 3	293	Success	509132
67	45:14.7	User Session 1-1	Query 4	294	Success	475202
68	45:15.0	User Session 1-1	Query 5	234	Success	376014
69	45:15.2	User Session 1-1	Query 6	24	Success	16047
70	45:15.3	User Session 1-1	Query 7	24	Success	17119
71	45:15.3	User Session 1-1	Query 8	21	Success	14580
72	45:15.3	User Session 1-1	Query 9	6	Success	729
73	45:15.3	User Session 1-1	Query 1	2594	Success	645
74	45:17.9	User Session 1-1	Query 2	19	Success	13277
75	45:17.9	User Session 1-1	Query 3	295	Success	509132
76	45:18.2	User Session 1-1	Query 4	279	Success	475202
77	45:18.5	User Session 1-1	Query 5	226	Success	376014
78	45:18.7	User Session 1-1	Query 6	20	Success	16047
79	45:18.8	User Session 1-1	Query 7	27	Success	17119
80	45:18.8	User Session 1-1	Query 8	18	Success	14580
81	45:18.8	User Session 1-1	Query 9	7	Success	729
82	45:18.8	User Session 1-1	Query 1	2612	Success	645
83	45:21.4	User Session 1-1	Query 2	20	Success	13277
84	45:21.5	User Session 1-1	Query 3	306	Success	509132
85	45:21.8	User Session 1-1	Query 4	286	Success	475202
86	45:22.1	User Session 1-1	Query 5	229	Success	376014
87	45:22.3	User Session 1-1	Query 6	20	Success	16047
88	45:22.3	User Session 1-1	Query 7	24	Success	17119
89	45:22.3	User Session 1-1	Query 8	18	Success	14580
90	45:22.4	User Session 1-1	Query 9	6	Success	729
91	45:22.4	User Session 1-1	Query 1	2622	Success	645
92	45:25.0	User Session 1-1	Query 2	21	Success	13277
93	45:25.0	User Session 1-1	Query 3	314	Success	509132
94	45:25.3	User Session 1-1	Query 4	289	Success	475202
95	45:25.6	User Session 1-1	Query 5	226	Success	376014

96	45:25.8	User Session 1-1	Query 6	19	Success	16047
97	45:25.8	User Session 1-1	Query 7	25	Success	17119
98	45:25.9	User Session 1-1	Query 8	18	Success	14580
99	45:25.9	User Session 1-1	Query 9	8	Success	729

InnoDB Power Test Run 1

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	14:51.6	User Session 1-1	Query 1	5921	Success	645
2	14:57.6	User Session 1-1	Query 2	41	Success	13277
3	14:57.6	User Session 1-1	Query 3	323	Success	509132
4	14:57.9	User Session 1-1	Query 4	306	Success	475202
5	14:58.2	User Session 1-1	Query 5	235	Success	376014
6	14:58.5	User Session 1-1	Query 6	23	Success	16047
7	14:58.5	User Session 1-1	Query 7	36	Success	17119
8	14:58.5	User Session 1-1	Query 8	32	Success	14580
9	14:58.6	User Session 1-1	Query 9	7	Success	729
10	14:58.6	User Session 1-1	Query 1	2479	Success	645
11	15:01.1	User Session 1-1	Query 2	27	Success	13277
12	15:01.1	User Session 1-1	Query 3	472	Success	509132
13	15:01.6	User Session 1-1	Query 4	398	Success	475202
14	15:02.0	User Session 1-1	Query 5	307	Success	376014
15	15:02.3	User Session 1-1	Query 6	35	Success	16047
16	15:02.3	User Session 1-1	Query 7	32	Success	17119
17	15:02.4	User Session 1-1	Query 8	23	Success	14580
18	15:02.4	User Session 1-1	Query 9	11	Success	729
19	15:02.4	User Session 1-1	Query 1	2686	Success	645
20	15:05.1	User Session 1-1	Query 2	26	Success	13277
21	15:05.1	User Session 1-1	Query 3	421	Success	509132
22	15:05.5	User Session 1-1	Query 4	407	Success	475202
23	15:05.9	User Session 1-1	Query 5	322	Success	376014
24	15:06.3	User Session 1-1	Query 6	32	Success	16047
25	15:06.3	User Session 1-1	Query 7	32	Success	17119
26	15:06.3	User Session 1-1	Query 8	26	Success	14580
27	15:06.4	User Session 1-1	Query 9	7	Success	729
28	15:06.4	User Session 1-1	Query 1	2633	Success	645
29	15:09.0	User Session 1-1	Query 2	25	Success	13277
30	15:09.0	User Session 1-1	Query 3	441	Success	509132
31	15:09.5	User Session 1-1	Query 4	384	Success	475202
32	15:09.9	User Session 1-1	Query 5	330	Success	376014
33	15:10.2	User Session 1-1	Query 6	25	Success	16047

34	15:10.2	User Session 1-1	Query 7	34	Success	17119
35	15:10.3	User Session 1-1	Query 8	24	Success	14580
36	15:10.3	User Session 1-1	Query 9	10	Success	729
37	15:10.3	User Session 1-1	Query 1	2791	Success	645
38	15:13.1	User Session 1-1	Query 2	24	Success	13277
39	15:13.1	User Session 1-1	Query 3	425	Success	509132
40	15:13.6	User Session 1-1	Query 4	409	Success	475202
41	15:14.0	User Session 1-1	Query 5	326	Success	376014
42	15:14.3	User Session 1-1	Query 6	28	Success	16047
43	15:14.3	User Session 1-1	Query 7	32	Success	17119
44	15:14.3	User Session 1-1	Query 8	24	Success	14580
45	15:14.4	User Session 1-1	Query 9	8	Success	729
46	15:14.4	User Session 1-1	Query 1	2692	Success	645
47	15:17.1	User Session 1-1	Query 2	20	Success	13277
48	15:17.1	User Session 1-1	Query 3	314	Success	509132
49	15:17.4	User Session 1-1	Query 4	312	Success	475202
50	15:17.7	User Session 1-1	Query 5	233	Success	376014
51	15:18.0	User Session 1-1	Query 6	23	Success	16047
52	15:18.0	User Session 1-1	Query 7	32	Success	17119
53	15:18.0	User Session 1-1	Query 8	26	Success	14580
54	15:18.1	User Session 1-1	Query 9	8	Success	729
55	15:18.1	User Session 1-1	Query 1	2716	Success	645
56	15:20.8	User Session 1-1	Query 2	19	Success	13277
57	15:20.8	User Session 1-1	Query 3	307	Success	509132
58	15:21.1	User Session 1-1	Query 4	280	Success	475202
59	15:21.4	User Session 1-1	Query 5	238	Success	376014
60	15:21.6	User Session 1-1	Query 6	26	Success	16047
61	15:21.7	User Session 1-1	Query 7	28	Success	17119
62	15:21.7	User Session 1-1	Query 8	19	Success	14580
63	15:21.7	User Session 1-1	Query 9	7	Success	729
64	15:21.7	User Session 1-1	Query 1	2645	Success	645
65	15:24.4	User Session 1-1	Query 2	24	Success	13277
66	15:24.4	User Session 1-1	Query 3	416	Success	509132
67	15:24.8	User Session 1-1	Query 4	410	Success	475202
68	15:25.2	User Session 1-1	Query 5	306	Success	376014
69	15:25.5	User Session 1-1	Query 6	28	Success	16047
70	15:25.6	User Session 1-1	Query 7	41	Success	17119
71	15:25.6	User Session 1-1	Query 8	23	Success	14580
72	15:25.6	User Session 1-1	Query 9	8	Success	729
73	15:25.7	User Session 1-1	Query 1	2678	Success	645
74	15:28.3	User Session 1-1	Query 2	33	Success	13277

75	15:28.4	User Session 1-1	Query 3	431	Success	509132
76	15:28.8	User Session 1-1	Query 4	405	Success	475202
77	15:29.2	User Session 1-1	Query 5	320	Success	376014
78	15:29.5	User Session 1-1	Query 6	32	Success	16047
79	15:29.6	User Session 1-1	Query 7	33	Success	17119
80	15:29.6	User Session 1-1	Query 8	25	Success	14580
81	15:29.6	User Session 1-1	Query 9	14	Success	729
82	15:29.6	User Session 1-1	Query 1	2614	Success	645
83	15:32.3	User Session 1-1	Query 2	23	Success	13277
84	15:32.3	User Session 1-1	Query 3	429	Success	509132
85	15:32.7	User Session 1-1	Query 4	393	Success	475202
86	15:33.1	User Session 1-1	Query 5	314	Success	376014
87	15:33.4	User Session 1-1	Query 6	26	Success	16047
88	15:33.5	User Session 1-1	Query 7	31	Success	17119
89	15:33.5	User Session 1-1	Query 8	25	Success	14580
90	15:33.5	User Session 1-1	Query 9	14	Success	729
91	15:33.5	User Session 1-1	Query 1	2722	Success	645
92	15:36.3	User Session 1-1	Query 2	19	Success	13277
93	15:36.3	User Session 1-1	Query 3	308	Success	509132
94	15:36.6	User Session 1-1	Query 4	279	Success	475202
95	15:36.9	User Session 1-1	Query 5	239	Success	376014
96	15:37.1	User Session 1-1	Query 6	22	Success	16047
97	15:37.2	User Session 1-1	Query 7	26	Success	17119
98	15:37.2	User Session 1-1	Query 8	18	Success	14580
99	15:37.2	User Session 1-1	Query 9	6	Success	729

InnoDB Power Test Run 2

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	13:44.5	User Session 1-1	Query 1	5603	Success	645
2	13:50.1	User Session 1-1	Query 2	22	Success	13277
3	13:50.1	User Session 1-1	Query 3	298	Success	509132
4	13:50.4	User Session 1-1	Query 4	289	Success	475202
5	13:50.7	User Session 1-1	Query 5	239	Success	376014
6	13:51.0	User Session 1-1	Query 6	22	Success	16047
7	13:51.0	User Session 1-1	Query 7	25	Success	17119
8	13:51.0	User Session 1-1	Query 8	20	Success	14580
9	13:51.0	User Session 1-1	Query 9	23	Success	729
10	13:51.0	User Session 1-1	Query 1	2695	Success	645
11	13:53.8	User Session 1-1	Query 2	23	Success	13277
12	13:53.8	User Session 1-1	Query 3	539	Success	509132

13	13:54.3	User Session 1-1	Query 4	674	Success	475202
14	13:55.0	User Session 1-1	Query 5	614	Success	376014
15	13:55.6	User Session 1-1	Query 6	25	Success	16047
16	13:55.6	User Session 1-1	Query 7	32	Success	17119
17	13:55.7	User Session 1-1	Query 8	120	Success	14580
18	13:55.8	User Session 1-1	Query 9	8	Success	729
19	13:55.8	User Session 1-1	Query 1	2689	Success	645
20	13:58.5	User Session 1-1	Query 2	21	Success	13277
21	13:58.5	User Session 1-1	Query 3	305	Success	509132
22	13:58.8	User Session 1-1	Query 4	296	Success	475202
23	13:59.1	User Session 1-1	Query 5	239	Success	376014
24	13:59.4	User Session 1-1	Query 6	30	Success	16047
25	13:59.4	User Session 1-1	Query 7	27	Success	17119
26	13:59.4	User Session 1-1	Query 8	18	Success	14580
27	13:59.4	User Session 1-1	Query 9	9	Success	729
28	13:59.5	User Session 1-1	Query 1	2722	Success	645
29	14:02.2	User Session 1-1	Query 2	19	Success	13277
30	14:02.2	User Session 1-1	Query 3	293	Success	509132
31	14:02.5	User Session 1-1	Query 4	281	Success	475202
32	14:02.8	User Session 1-1	Query 5	235	Success	376014
33	14:03.0	User Session 1-1	Query 6	19	Success	16047
34	14:03.0	User Session 1-1	Query 7	26	Success	17119
35	14:03.1	User Session 1-1	Query 8	19	Success	14580
36	14:03.1	User Session 1-1	Query 9	6	Success	729
37	14:03.1	User Session 1-1	Query 1	2631	Success	645
38	14:05.7	User Session 1-1	Query 2	19	Success	13277
39	14:05.7	User Session 1-1	Query 3	457	Success	509132
40	14:06.2	User Session 1-1	Query 4	412	Success	475202
41	14:06.6	User Session 1-1	Query 5	229	Success	376014
42	14:06.9	User Session 1-1	Query 6	24	Success	16047
43	14:06.9	User Session 1-1	Query 7	30	Success	17119
44	14:06.9	User Session 1-1	Query 8	18	Success	14580
45	14:06.9	User Session 1-1	Query 9	6	Success	729
46	14:06.9	User Session 1-1	Query 1	2631	Success	645
47	14:09.6	User Session 1-1	Query 2	23	Success	13277
48	14:09.6	User Session 1-1	Query 3	315	Success	509132
49	14:09.9	User Session 1-1	Query 4	281	Success	475202
50	14:10.2	User Session 1-1	Query 5	226	Success	376014
51	14:10.4	User Session 1-1	Query 6	26	Success	16047
52	14:10.5	User Session 1-1	Query 7	24	Success	17119
53	14:10.5	User Session 1-1	Query 8	18	Success	14580

54	14:10.5	User Session 1-1	Query 9	10	Success	729
55	14:10.5	User Session 1-1	Query 1	2645	Success	645
56	14:13.2	User Session 1-1	Query 2	20	Success	13277
57	14:13.2	User Session 1-1	Query 3	297	Success	509132
58	14:13.5	User Session 1-1	Query 4	288	Success	475202
59	14:13.8	User Session 1-1	Query 5	237	Success	376014
60	14:14.0	User Session 1-1	Query 6	21	Success	16047
61	14:14.0	User Session 1-1	Query 7	25	Success	17119
62	14:14.1	User Session 1-1	Query 8	17	Success	14580
63	14:14.1	User Session 1-1	Query 9	6	Success	729
64	14:14.1	User Session 1-1	Query 1	2648	Success	645
65	14:16.7	User Session 1-1	Query 2	18	Success	13277
66	14:16.8	User Session 1-1	Query 3	313	Success	509132
67	14:17.1	User Session 1-1	Query 4	300	Success	475202
68	14:17.4	User Session 1-1	Query 5	234	Success	376014
69	14:17.6	User Session 1-1	Query 6	20	Success	16047
70	14:17.6	User Session 1-1	Query 7	25	Success	17119
71	14:17.7	User Session 1-1	Query 8	19	Success	14580
72	14:17.7	User Session 1-1	Query 9	7	Success	729
73	14:17.7	User Session 1-1	Query 1	2686	Success	645
74	14:20.4	User Session 1-1	Query 2	23	Success	13277
75	14:20.4	User Session 1-1	Query 3	444	Success	509132
76	14:20.9	User Session 1-1	Query 4	406	Success	475202
77	14:21.3	User Session 1-1	Query 5	327	Success	376014
78	14:21.6	User Session 1-1	Query 6	27	Success	16047
79	14:21.6	User Session 1-1	Query 7	29	Success	17119
80	14:21.7	User Session 1-1	Query 8	25	Success	14580
81	14:21.7	User Session 1-1	Query 9	6	Success	729
82	14:21.7	User Session 1-1	Query 1	2643	Success	645
83	14:24.3	User Session 1-1	Query 2	29	Success	13277
84	14:24.4	User Session 1-1	Query 3	312	Success	509132
85	14:24.7	User Session 1-1	Query 4	296	Success	475202
86	14:25.0	User Session 1-1	Query 5	223	Success	376014
87	14:25.2	User Session 1-1	Query 6	21	Success	16047
88	14:25.2	User Session 1-1	Query 7	29	Success	17119
89	14:25.3	User Session 1-1	Query 8	24	Success	14580
90	14:25.3	User Session 1-1	Query 9	8	Success	729
91	14:25.3	User Session 1-1	Query 1	2636	Success	645
92	14:27.9	User Session 1-1	Query 2	20	Success	13277
93	14:28.0	User Session 1-1	Query 3	312	Success	509132
94	14:28.3	User Session 1-1	Query 4	275	Success	475202

95	14:28.5	User Session 1-1	Query 5	236	Success	376014
96	14:28.8	User Session 1-1	Query 6	20	Success	16047
97	14:28.8	User Session 1-1	Query 7	24	Success	17119
98	14:28.8	User Session 1-1	Query 8	20	Success	14580
99	14:28.9	User Session 1-1	Query 9	6	Success	729

MyISAM Throughput Test Run 1

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	58:19.5	User Session 18 18-1	Query 8	28	Success	14580
2	58:19.5	User Session 2 2-1	Query 2	55	Success	13277
3	58:19.5	User Session 1 1-1	Query 9	58	Success	729
4	58:19.5	User Session 21 21-1	Query 9	58	Success	729
5	58:19.5	User Session 13 13-1	Query 9	66	Success	729
6	58:19.5	User Session 15 15-1	Query 9	66	Success	729
7	58:19.5	User Session 4 4-1	Query 6	88	Success	16047
8	58:19.5	User Session 20 20-1	Query 8	93	Success	14580
9	58:19.5	User Session 8 8-1	Query 8	111	Success	14580
10	58:19.5	User Session 22 22-1	Query 2	114	Success	13277
11	58:19.5	User Session 12 12-1	Query 8	126	Success	14580
12	58:19.5	User Session 10 10-1	Query 6	130	Success	16047
13	58:19.5	User Session 16 16-1	Query 2	127	Success	13277
14	58:19.5	User Session 18 18-1	Query 6	101	Success	16047
15	58:19.6	User Session 4 4-1	Query 8	85	Success	14580
16	58:19.6	User Session 12 12-1	Query 2	62	Success	13277
17	58:19.6	User Session 10 10-1	Query 8	65	Success	14580
18	58:19.6	User Session 8 8-1	Query 2	89	Success	13277
19	58:19.6	User Session 22 22-1	Query 8	103	Success	14580
20	58:19.5	User Session 2 2-1	Query 4	1582	Success	475202
21	58:19.5	User Session 14 14-1	Query 4	1928	Success	475202
22	58:21.4	User Session 14 14-1	Query 8	268	Success	14580
23	58:19.6	User Session 20 20-1	Query 4	2798	Success	475202
24	58:21.7	User Session 14 14-1	Query 6	820	Success	16047
25	58:21.1	User Session 2 2-1	Query 6	1821	Success	16047
26	58:22.5	User Session 14 14-1	Query 2	493	Success	13277
27	58:22.4	User Session 20 20-1	Query 2	621	Success	13277
28	58:19.6	User Session 18 18-1	Query 4	3901	Success	475202
29	58:22.9	User Session 2 2-1	Query 8	674	Success	14580
30	58:23.0	User Session 20 20-1	Query 6	771	Success	16047
31	58:23.5	User Session 18 18-1	Query 2	537	Success	13277
32	58:19.7	User Session 12 12-1	Query 4	4584	Success	475202

33	58:19.6	User Session 16 16-1	Query 4	4895	Success	475202
34	58:19.5	User Session 11 11-1	Query 5	5226	Success	376014
35	58:19.7	User Session 10 10-1	Query 4	5043	Success	475202
36	58:24.3	User Session 12 12-1	Query 6	570	Success	16047
37	58:19.5	User Session 7 7-1	Query 5	5384	Success	376014
38	58:19.5	User Session 19 19-1	Query 3	5387	Success	509132
39	58:24.5	User Session 16 16-1	Query 6	520	Success	16047
40	58:24.7	User Session 10 10-1	Query 2	401	Success	13277
41	58:19.7	User Session 4 4-1	Query 4	5515	Success	475202
42	58:19.5	User Session 6 6-1	Query 4	5733	Success	475202
43	58:24.7	User Session 11 11-1	Query 7	503	Success	17119
44	58:19.7	User Session 8 8-1	Query 4	5539	Success	475202
45	58:24.9	User Session 7 7-1	Query 7	365	Success	17119
46	58:19.7	User Session 22 22-1	Query 4	5533	Success	475202
47	58:24.9	User Session 19 19-1	Query 7	370	Success	17119
48	58:25.0	User Session 16 16-1	Query 8	273	Success	14580
49	58:25.2	User Session 11 11-1	Query 9	89	Success	729
50	58:25.2	User Session 7 7-1	Query 9	77	Success	729
51	58:25.2	User Session 4 4-1	Query 2	154	Success	13277
52	58:25.2	User Session 6 6-1	Query 2	130	Success	13277
53	58:25.2	User Session 8 8-1	Query 6	164	Success	16047
54	58:25.3	User Session 22 22-1	Query 6	191	Success	16047
55	58:25.3	User Session 6 6-1	Query 8	93	Success	14580
56	58:25.4	User Session 6 6-1	Query 6	47	Success	16047
57	58:19.6	User Session 13 13-1	Query 3	5947	Success	509132
58	58:25.3	User Session 7 7-1	Query 3	563	Success	509132
59	58:19.5	User Session 17 17-1	Query 5	8667	Success	376014
60	58:28.2	User Session 17 17-1	Query 7	100	Success	17119
61	58:28.3	User Session 17 17-1	Query 9	35	Success	729
62	58:28.3	User Session 17 17-1	Query 3	1501	Success	509132
63	58:19.6	User Session 15 15-1	Query 1	195280	Success	623
64	58:25.3	User Session 11 11-1	Query 1	189528	Success	623
65	01:34.8	User Session 11 11-1	Query 3	10272	Success	509132
66	01:34.8	User Session 15 15-1	Query 3	10279	Success	509132
67	01:45.1	User Session 15 15-1	Query 7	1540	Success	17119
68	01:46.7	User Session 15 15-1	Query 5	12174	Success	376014
69	58:19.5	User Session 3 3-1	Query 1	401080	Success	645
70	05:00.6	User Session 3 3-1	Query 5	22880	Success	376014
71	05:23.4	User Session 3 3-1	Query 7	4287	Success	17119
72	58:19.5	User Session 5 5-1	Query 1	577956	Success	645
73	05:27.7	User Session 3 3-1	Query 3	149711	Success	645

74	07:57.4	User Session 3 3-1	Query 9	33	Success	729
75	07:57.4	User Session 5 5-1	Query 5	6339	Success	376014
76	08:03.8	User Session 5 5-1	Query 7	1956	Success	17119
77	08:05.7	User Session 5 5-1	Query 9	760	Success	729
78	08:06.5	User Session 5 5-1	Query 3	21302	Success	509132
79	58:29.8	User Session 17 17-1	Query 1	620031	Success	623
80	58:25.5	User Session 13 13-1	Query 1	647326	Success	645
81	09:12.8	User Session 13 13-1	Query 5	3846	Success	376014
82	09:16.7	User Session 13 13-1	Query 7	1291	Success	17119
83	58:19.6	User Session 21 21-1	Query 1	661312	Success	645
84	09:20.9	User Session 21 21-1	Query 3	4226	Success	509132
85	09:25.1	User Session 21 21-1	Query 5	6890	Success	376014
86	09:32.0	User Session 21 21-1	Query 7	755	Success	17119
87	58:19.6	User Session 1 1-1	Query 1	694601	Success	645
88	09:54.2	User Session 1 1-1	Query 5	6891	Success	376014
89	10:01.0	User Session 1 1-1	Query 7	13738	Success	17119
90	10:14.8	User Session 1 1-1	Query 3	109394	Success	509132
91	58:25.3	User Session 19 19-1	Query 1	870768	Success	645
92	12:56.0	User Session 19 19-1	Query 9	7	Success	729
93	12:56.0	User Session 19 19-1	Query 5	1014	Success	376014
94	58:19.5	User Session 9 9-1	Query 1	878470	Success	645
95	12:58.0	User Session 9 9-1	Query 9	7	Success	729
96	58:25.9	User Session 7 7-1	Query 1	872867	Success	645
97	12:58.0	User Session 9 9-1	Query 5	911	Success	376014
98	12:58.9	User Session 9 9-1	Query 7	37	Success	17119
99	12:58.9	User Session 9 9-1	Query 3	307	Success	509132

MyISAM Throughput Test Run 2

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	31:30.2	User Session 9 9-1	Query 9	105	Success	729
2	31:30.3	User Session 19 19-1	Query 9	89	Success	729
3	31:30.3	User Session 21 21-1	Query 9	75	Success	729
4	31:30.2	User Session 4 4-1	Query 8	182	Success	14580
5	31:30.3	User Session 18 18-1	Query 6	159	Success	16047
6	31:30.2	User Session 5 5-1	Query 7	199	Success	17119
7	31:30.3	User Session 16 16-1	Query 8	221	Success	14580
8	31:30.3	User Session 10 10-1	Query 8	239	Success	14580
9	31:30.5	User Session 5 5-1	Query 9	61	Success	729
10	31:30.3	User Session 12 12-1	Query 8	252	Success	14580
11	31:30.3	User Session 17 17-1	Query 7	282	Success	17119

12	31:30.2	User Session 6 6-1	Query 6	447	Success	16047
13	31:30.4	User Session 4 4-1	Query 6	273	Success	16047
14	31:30.5	User Session 10 10-1	Query 6	398	Success	16047
15	31:30.5	User Session 12 12-1	Query 6	381	Success	16047
16	31:30.5	User Session 16 16-1	Query 2	495	Success	13277
17	31:30.7	User Session 6 6-1	Query 8	466	Success	14580
18	31:30.4	User Session 18 18-1	Query 2	734	Success	13277
19	31:31.2	User Session 6 6-1	Query 2	34	Success	13277
20	31:30.9	User Session 12 12-1	Query 2	295	Success	13277
21	31:30.9	User Session 10 10-1	Query 2	324	Success	13277
22	31:30.4	User Session 9 9-1	Query 5	3060	Success	376014
23	31:30.3	User Session 20 20-1	Query 4	3581	Success	475202
24	31:33.9	User Session 20 20-1	Query 8	558	Success	14580
25	31:30.2	User Session 3 3-1	Query 5	4840	Success	376014
26	31:34.4	User Session 20 20-1	Query 2	1252	Success	13277
27	31:35.1	User Session 3 3-1	Query 9	619	Success	729
28	31:30.6	User Session 17 17-1	Query 5	5637	Success	376014
29	31:30.2	User Session 2 2-1	Query 4	6027	Success	475202
30	31:30.4	User Session 19 19-1	Query 5	5957	Success	376014
31	31:35.7	User Session 20 20-1	Query 6	1098	Success	16047
32	31:30.2	User Session 7 7-1	Query 5	6847	Success	376014
33	31:30.3	User Session 14 14-1	Query 4	6835	Success	475202
34	31:30.2	User Session 8 8-1	Query 4	6864	Success	475202
35	31:36.3	User Session 2 2-1	Query 6	983	Success	16047
36	31:30.5	User Session 5 5-1	Query 5	6757	Success	376014
37	31:31.2	User Session 10 10-1	Query 4	6113	Success	475202
38	31:31.2	User Session 12 12-1	Query 4	6230	Success	475202
39	31:31.0	User Session 16 16-1	Query 4	6459	Success	475202
40	31:31.2	User Session 6 6-1	Query 4	6264	Success	475202
41	31:31.2	User Session 18 18-1	Query 4	6330	Success	475202
42	31:30.7	User Session 4 4-1	Query 4	6823	Success	475202
43	31:30.3	User Session 13 13-1	Query 3	7282	Success	509132
44	31:37.1	User Session 14 14-1	Query 6	480	Success	16047
45	31:30.3	User Session 22 22-1	Query 4	7297	Success	475202
46	31:37.1	User Session 7 7-1	Query 7	494	Success	17119
47	31:37.1	User Session 8 8-1	Query 6	477	Success	16047
48	31:37.2	User Session 2 2-1	Query 8	347	Success	14580
49	31:37.6	User Session 7 7-1	Query 9	20	Success	729
50	31:37.6	User Session 8 8-1	Query 8	38	Success	14580
51	31:37.5	User Session 4 4-1	Query 2	112	Success	13277
52	31:37.5	User Session 16 16-1	Query 6	206	Success	16047

53	31:37.6	User Session 14 14-1	Query 2	126	Success	13277
54	31:37.6	User Session 22 22-1	Query 8	140	Success	14580
55	31:37.5	User Session 18 18-1	Query 8	265	Success	14580
56	31:37.6	User Session 2 2-1	Query 2	185	Success	13277
57	31:37.6	User Session 13 13-1	Query 7	233	Success	17119
58	31:37.7	User Session 22 22-1	Query 6	65	Success	16047
59	31:37.6	User Session 8 8-1	Query 2	593	Success	13277
60	31:37.7	User Session 14 14-1	Query 8	534	Success	14580
61	31:37.8	User Session 22 22-1	Query 2	503	Success	13277
62	31:37.3	User Session 5 5-1	Query 3	1277	Success	509132
63	31:37.8	User Session 13 13-1	Query 5	954	Success	376014
64	31:38.8	User Session 13 13-1	Query 9	31	Success	729
65	31:33.4	User Session 9 9-1	Query 1	293387	Success	623
66	31:38.6	User Session 5 5-1	Query 1	288268	Success	623
67	36:26.8	User Session 9 9-1	Query 3	15663	Success	509132
68	36:42.5	User Session 9 9-1	Query 7	4334	Success	17119
69	31:30.3	User Session 11 11-1	Query 1	416225	Success	645
70	38:26.5	User Session 11 11-1	Query 5	12391	Success	376014
71	38:38.9	User Session 11 11-1	Query 7	4329	Success	17119
72	31:30.2	User Session 1 1-1	Query 1	457949	Success	645
73	39:08.2	User Session 1 1-1	Query 9	594	Success	729
74	38:43.2	User Session 11 11-1	Query 3	25598	Success	509132
75	39:08.8	User Session 11 11-1	Query 9	794	Success	729
76	39:08.8	User Session 1 1-1	Query 3	15720	Success	509132
77	39:24.5	User Session 1 1-1	Query 5	16805	Success	376014
78	39:41.3	User Session 1 1-1	Query 7	2885	Success	17119
79	31:30.4	User Session 21 21-1	Query 1	542484	Success	645
80	40:32.9	User Session 21 21-1	Query 5	7274	Success	376014
81	40:40.1	User Session 21 21-1	Query 3	8305	Success	509132
82	40:48.4	User Session 21 21-1	Query 7	2206	Success	17119
83	31:36.2	User Session 17 17-1	Query 1	560387	Success	645
84	40:56.6	User Session 17 17-1	Query 9	9	Success	729
85	40:56.6	User Session 17 17-1	Query 3	3308	Success	509132
86	31:36.3	User Session 19 19-1	Query 1	587546	Success	645
87	41:23.9	User Session 19 19-1	Query 7	512	Success	17119
88	31:30.3	User Session 15 15-1	Query 1	595759	Success	645
89	41:26.0	User Session 15 15-1	Query 7	830	Success	17119
90	41:24.4	User Session 19 19-1	Query 3	5154	Success	509132
91	41:26.9	User Session 15 15-1	Query 3	4292	Success	509132
92	41:31.2	User Session 15 15-1	Query 5	5341	Success	376014
93	41:36.5	User Session 15 15-1	Query 9	623	Success	729

94	31:37.6	User Session 7 7-1	Query 1	636654	Success	645
95	42:14.3	User Session 7 7-1	Query 3	717	Success	509132
96	31:35.7	User Session 3 3-1	Query 1	648286	Success	645
97	31:38.8	User Session 13 13-1	Query 1	645832	Success	645
98	42:24.0	User Session 3 3-1	Query 3	904	Success	509132
99	42:25.0	User Session 3 3-1	Query 7	31	Success	17119

InnoDB Throughput Test Run 1

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	39:19.3	User Session 7 7-1	Query 9	25	Success	729
2	39:19.3	User Session 11 11-1	Query 9	35	Success	729
3	39:19.3	User Session 13 13-1	Query 9	36	Success	729
4	39:19.3	User Session 19 19-1	Query 9	39	Success	729
5	39:19.3	User Session 14 14-1	Query 8	176	Success	14580
6	39:19.3	User Session 16 16-1	Query 8	180	Success	14580
7	39:19.3	User Session 10 10-1	Query 8	231	Success	14580
8	39:19.3	User Session 12 12-1	Query 8	229	Success	14580
9	39:19.3	User Session 4 4-1	Query 8	241	Success	14580
10	39:19.3	User Session 8 8-1	Query 6	254	Success	16047
11	39:19.3	User Session 20 20-1	Query 6	245	Success	16047
12	39:19.3	User Session 22 22-1	Query 6	252	Success	16047
13	39:19.3	User Session 6 6-1	Query 6	273	Success	16047
14	39:19.3	User Session 5 5-1	Query 7	302	Success	17119
15	39:19.6	User Session 5 5-1	Query 9	69	Success	729
16	39:19.6	User Session 20 20-1	Query 8	287	Success	14580
17	39:19.6	User Session 6 6-1	Query 8	291	Success	14580
18	39:19.6	User Session 22 22-1	Query 8	299	Success	14580
19	39:19.5	User Session 14 14-1	Query 2	737	Success	13277
20	39:19.5	User Session 16 16-1	Query 2	735	Success	13277
21	39:19.8	User Session 20 20-1	Query 2	711	Success	13277
22	39:19.9	User Session 6 6-1	Query 2	696	Success	13277
23	39:19.3	User Session 7 7-1	Query 5	1269	Success	376014
24	39:19.3	User Session 9 9-1	Query 5	1393	Success	376014
25	39:20.2	User Session 14 14-1	Query 6	655	Success	16047
26	39:20.7	User Session 9 9-1	Query 7	788	Success	17119
27	39:21.5	User Session 9 9-1	Query 9	745	Success	729
28	39:19.3	User Session 2 2-1	Query 4	3496	Success	475202
29	39:19.3	User Session 18 18-1	Query 4	4089	Success	475202
30	39:22.8	User Session 2 2-1	Query 6	1076	Success	16047
31	39:19.9	User Session 22 22-1	Query 4	4529	Success	475202

32	39:19.5	User Session 12 12-1	Query 4	4876	Success	475202
33	39:19.6	User Session 8 8-1	Query 4	4900	Success	475202
34	39:19.5	User Session 10 10-1	Query 4	4935	Success	475202
35	39:19.3	User Session 13 13-1	Query 3	5205	Success	509132
36	39:20.6	User Session 6 6-1	Query 4	4130	Success	475202
37	39:20.6	User Session 20 20-1	Query 4	4157	Success	475202
38	39:23.9	User Session 2 2-1	Query 8	882	Success	14580
39	39:23.4	User Session 18 18-1	Query 6	1368	Success	16047
40	39:19.3	User Session 11 11-1	Query 3	5513	Success	509132
41	39:24.4	User Session 12 12-1	Query 6	821	Success	16047
42	39:24.4	User Session 22 22-1	Query 2	835	Success	13277
43	39:20.2	User Session 16 16-1	Query 4	5154	Success	475202
44	39:24.5	User Session 8 8-1	Query 2	957	Success	13277
45	39:24.8	User Session 18 18-1	Query 8	672	Success	14580
46	39:24.8	User Session 2 2-1	Query 2	699	Success	13277
47	39:19.6	User Session 4 4-1	Query 4	5913	Success	475202
48	39:20.6	User Session 7 7-1	Query 3	4876	Success	509132
49	39:20.9	User Session 14 14-1	Query 4	4594	Success	475202
50	39:25.2	User Session 12 12-1	Query 2	239	Success	13277
51	39:24.5	User Session 10 10-1	Query 2	1012	Success	13277
52	39:25.4	User Session 16 16-1	Query 6	122	Success	16047
53	39:25.5	User Session 10 10-1	Query 6	102	Success	16047
54	39:25.4	User Session 18 18-1	Query 2	157	Success	13277
55	39:25.4	User Session 8 8-1	Query 8	224	Success	14580
56	39:25.5	User Session 4 4-1	Query 2	217	Success	13277
57	39:25.7	User Session 4 4-1	Query 6	49	Success	16047
58	39:24.6	User Session 13 13-1	Query 5	1252	Success	376014
59	39:19.3	User Session 17 17-1	Query 5	91342	Success	376014
60	40:50.7	User Session 17 17-1	Query 7	5772	Success	17119
61	40:56.4	User Session 17 17-1	Query 9	604	Success	729
62	39:19.3	User Session 3 3-1	Query 1	292444	Success	623
63	44:11.8	User Session 3 3-1	Query 7	8304	Success	17119
64	44:20.1	User Session 3 3-1	Query 5	51260	Success	376014
65	45:11.3	User Session 3 3-1	Query 3	258186	Success	623
66	40:57.0	User Session 17 17-1	Query 1	596676	Success	623
67	49:29.5	User Session 3 3-1	Query 9	101676	Success	1298
68	39:25.5	User Session 7 7-1	Query 1	841327	Success	623
69	39:25.8	User Session 13 13-1	Query 1	841002	Success	623
70	39:19.4	User Session 19 19-1	Query 1	847454	Success	623
71	53:26.8	User Session 13 13-1	Query 7	1970	Success	17119
72	53:26.8	User Session 7 7-1	Query 7	1973	Success	17119

73	50:53.7	User Session 17 17-1	Query 3	156520	Success	509132
74	53:26.8	User Session 19 19-1	Query 3	3568	Success	509132
75	53:30.4	User Session 19 19-1	Query 5	1101	Success	376014
76	53:31.5	User Session 19 19-1	Query 7	704	Success	17119
77	39:19.7	User Session 5 5-1	Query 1	886710	Success	645
78	54:06.4	User Session 5 5-1	Query 3	15774	Success	509132
79	54:22.2	User Session 5 5-1	Query 5	6815	Success	376014
80	39:19.3	User Session 21 21-1	Query 1	933177	Success	645
81	54:52.5	User Session 21 21-1	Query 3	5548	Success	509132
82	39:19.3	User Session 15 15-1	Query 1	942093	Success	645
83	54:58.0	User Session 21 21-1	Query 5	4818	Success	376014
84	55:02.9	User Session 21 21-1	Query 7	1024	Success	17119
85	55:03.9	User Session 21 21-1	Query 9	23	Success	729
86	55:01.4	User Session 15 15-1	Query 3	4538	Success	509132
87	55:06.0	User Session 15 15-1	Query 5	4052	Success	376014
88	55:10.0	User Session 15 15-1	Query 7	1029	Success	17119
89	55:11.0	User Session 15 15-1	Query 9	620	Success	729
90	39:22.2	User Session 9 9-1	Query 1	968163	Success	645
91	55:30.4	User Session 9 9-1	Query 3	1835	Success	509132
92	39:19.3	User Session 1 1-1	Query 1	973888	Success	645
93	39:24.9	User Session 11 11-1	Query 1	969131	Success	645
94	55:33.2	User Session 1 1-1	Query 3	920	Success	509132
95	55:34.0	User Session 11 11-1	Query 5	374	Success	376014
96	55:34.4	User Session 11 11-1	Query 7	73	Success	17119
97	55:34.1	User Session 1 1-1	Query 5	398	Success	376014
98	55:34.5	User Session 1 1-1	Query 7	15	Success	17119
99	55:34.5	User Session 1 1-1	Query 9	7	Success	729

InnoDB Throughput Test Run 2

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	01:28.3	User Session 9 9-1	Query 9	22	Success	729
2	01:28.3	User Session 11 11-1	Query 7	34	Success	17119
3	01:28.3	User Session 21 21-1	Query 9	25	Success	729
4	01:28.3	User Session 17 17-1	Query 7	40	Success	17119
5	01:28.3	User Session 11 11-1	Query 9	13	Success	729
6	01:28.3	User Session 21 21-1	Query 7	26	Success	17119
7	01:28.3	User Session 17 17-1	Query 9	35	Success	729
8	01:28.3	User Session 4 4-1	Query 6	142	Success	16047
9	01:28.3	User Session 8 8-1	Query 6	143	Success	16047
10	01:28.3	User Session 6 6-1	Query 6	171	Success	16047

11	01:28.3	User Session 22 22-1	Query 8	291	Success	14580
12	01:28.4	User Session 4 4-1	Query 8	161	Success	14580
13	01:28.3	User Session 20 20-1	Query 8	292	Success	14580
14	01:28.3	User Session 18 18-1	Query 2	297	Success	13277
15	01:28.4	User Session 6 6-1	Query 2	233	Success	13277
16	01:28.6	User Session 22 22-1	Query 2	248	Success	13277
17	01:28.6	User Session 4 4-1	Query 2	324	Success	13277
18	01:28.7	User Session 6 6-1	Query 8	344	Success	14580
19	01:28.8	User Session 22 22-1	Query 6	518	Success	16047
20	01:28.3	User Session 5 5-1	Query 5	1097	Success	376014
21	01:28.3	User Session 7 7-1	Query 5	1227	Success	376014
22	01:29.4	User Session 5 5-1	Query 9	150	Success	729
23	01:28.3	User Session 15 15-1	Query 5	1475	Success	376014
24	01:29.5	User Session 7 7-1	Query 7	644	Success	17119
25	01:29.5	User Session 5 5-1	Query 7	628	Success	17119
26	01:28.3	User Session 14 14-1	Query 4	3479	Success	475202
27	01:28.6	User Session 20 20-1	Query 4	3474	Success	475202
28	01:28.3	User Session 2 2-1	Query 2	4044	Success	13277
29	01:28.3	User Session 16 16-1	Query 4	4289	Success	475202
30	01:28.3	User Session 12 12-1	Query 4	4351	Success	475202
31	01:31.8	User Session 14 14-1	Query 8	866	Success	14580
32	01:28.3	User Session 19 19-1	Query 3	4398	Success	509132
33	01:28.3	User Session 21 21-1	Query 3	4352	Success	509132
34	01:28.3	User Session 10 10-1	Query 4	4440	Success	475202
35	01:28.9	User Session 4 4-1	Query 4	3886	Success	475202
36	01:29.3	User Session 22 22-1	Query 4	3459	Success	475202
37	01:28.6	User Session 18 18-1	Query 4	4232	Success	475202
38	01:32.0	User Session 20 20-1	Query 2	777	Success	13277
39	01:29.0	User Session 6 6-1	Query 4	3908	Success	475202
40	01:32.3	User Session 2 2-1	Query 6	634	Success	16047
41	01:28.4	User Session 8 8-1	Query 4	4552	Success	475202
42	01:32.6	User Session 14 14-1	Query 2	396	Success	13277
43	01:32.6	User Session 16 16-1	Query 8	456	Success	14580
44	01:32.6	User Session 12 12-1	Query 8	399	Success	14580
45	01:32.8	User Session 18 18-1	Query 6	273	Success	16047
46	01:32.7	User Session 10 10-1	Query 2	376	Success	13277
47	01:32.8	User Session 20 20-1	Query 6	278	Success	16047
48	01:33.0	User Session 2 2-1	Query 8	220	Success	14580
49	01:30.1	User Session 5 5-1	Query 3	3114	Success	509132
50	01:33.0	User Session 8 8-1	Query 8	296	Success	14580
51	01:33.0	User Session 12 12-1	Query 6	230	Success	16047

52	01:33.0	User Session 16 16-1	Query 2	236	Success	13277
53	01:33.0	User Session 14 14-1	Query 6	292	Success	16047
54	01:33.1	User Session 18 18-1	Query 8	227	Success	14580
55	01:33.1	User Session 10 10-1	Query 8	243	Success	14580
56	01:33.3	User Session 16 16-1	Query 6	215	Success	16047
57	01:33.3	User Session 8 8-1	Query 2	224	Success	13277
58	01:28.3	User Session 3 3-1	Query 3	5254	Success	509132
59	01:33.5	User Session 3 3-1	Query 9	396	Success	729
60	01:33.3	User Session 12 12-1	Query 2	644	Success	13277
61	01:33.3	User Session 10 10-1	Query 6	569	Success	16047
62	01:32.7	User Session 19 19-1	Query 5	1370	Success	376014
63	01:33.2	User Session 2 2-1	Query 4	1774	Success	475202
64	01:33.9	User Session 3 3-1	Query 5	1946	Success	376014
65	01:35.9	User Session 3 3-1	Query 7	51	Success	17119
66	01:28.3	User Session 1 1-1	Query 1	207036	Success	623
67	04:55.3	User Session 1 1-1	Query 3	57011	Success	509132
68	01:34.1	User Session 19 19-1	Query 1	651370	Success	623
69	12:25.4	User Session 19 19-1	Query 7	3656	Success	17119
70	12:29.1	User Session 19 19-1	Query 9	49	Success	729
71	05:52.3	User Session 1 1-1	Query 5	410681	Success	376014
72	12:43.0	User Session 1 1-1	Query 9	1756	Success	729
73	12:44.8	User Session 1 1-1	Query 7	4252	Success	17119
74	01:28.3	User Session 13 13-1	Query 1	804597	Success	645
75	14:52.9	User Session 13 13-1	Query 7	2387	Success	17119
76	14:55.3	User Session 13 13-1	Query 3	16506	Success	509132
77	15:11.8	User Session 13 13-1	Query 5	21469	Success	376014
78	15:33.2	User Session 13 13-1	Query 9	2004	Success	729
79	01:28.3	User Session 11 11-1	Query 1	940028	Success	645
80	17:08.4	User Session 11 11-1	Query 3	10679	Success	509132
81	01:33.3	User Session 5 5-1	Query 1	966094	Success	623
82	17:19.0	User Session 11 11-1	Query 5	23007	Success	376014
83	01:28.4	User Session 17 17-1	Query 1	980922	Success	645
84	17:49.3	User Session 17 17-1	Query 5	5113	Success	376014
85	17:54.4	User Session 17 17-1	Query 3	7867	Success	509132
86	01:28.3	User Session 9 9-1	Query 1	997729	Success	645
87	18:06.0	User Session 9 9-1	Query 3	7728	Success	509132
88	18:13.8	User Session 9 9-1	Query 5	8657	Success	376014
89	18:22.4	User Session 9 9-1	Query 7	1762	Success	17119
90	01:30.1	User Session 7 7-1	Query 1	1026475	Success	645
91	18:36.6	User Session 7 7-1	Query 9	12	Success	729
92	18:36.6	User Session 7 7-1	Query 3	3421	Success	509132

93	01:32.7	User Session 21 21-1	Query 1	1253085	Success	645
94	22:25.8	User Session 21 21-1	Query 5	982	Success	376014
95	01:29.7	User Session 15 15-1	Query 1	1257889	Success	645
96	22:27.6	User Session 15 15-1	Query 9	8	Success	729
97	22:27.7	User Session 15 15-1	Query 7	31	Success	17119
98	01:35.9	User Session 3 3-1	Query 1	1252590	Success	645
99	22:27.7	User Session 15 15-1	Query 3	882	Success	509132

MyISAM Power Test Summary

	Run One					Run Two				
	Total	Mean	Min	Max	StDev	Total	Mean	Min	Max	StDev
Query 1	29734	2703	2601	2885	90	28915	2628	2594	2706	35
Query 2	284	25	20	38	6	230	20	18	26	3
Query 3	3631	330	293	454	59	3330	302	289	323	10
Query 4	3453	313	280	435	54	3093	281	272	294	7
Query 5	2763	251	219	344	43	2536	230	223	241	5
Query 6	245	22	20	28	2	230	20	18	25	2
Query 7	334	30	27	36	3	304	27	24	47	6
Query 8	234	21	18	34	4	219	19	18	26	3
Query 9	82	7	6	14	2	73	6	6	8	1

InnoDB Power Test Summary

	Run One					Run Two				
	Total	Mean	Min	Max	StDev	Total	Mean	Min	Max	StDev
Query 1	32577	2961	2479	5921	939	32229	2929	2631	5603	846
Query 2	281	25	19	41	6	237	21	18	29	3
Query 3	4287	389	307	472	60	3885	353	293	539	81
Query 4	3983	362	279	410	53	3798	345	275	674	114
Query 5	3170	288	233	330	40	3039	276	223	614	110
Query 6	300	27	22	35	4	255	23	19	30	4
Query 7	357	32	26	41	4	296	26	24	32	3
Query 8	265	24	18	32	4	318	28	17	120	29
Query 9	100	9	6	14	3	195	8	6	23	5

MyISAM Throughput Test Summary

	Run One					Run Two				
	Total	Mean	Min	Max	StDev	Total	Mean	Min	Max	StDev
Query 1	6609219	600838	189528	878470	236066	5672777	515707	288268	648286	127139
Query 2	2783	253	55	621	204	4653	423	34	1252	336
Query 3	318889	28989	307	149711	48510	88220	8020	717	25598	7514
Query 4	47051	4277	1582	5733	1446	68823	6256	3581	7297	924
Query 5	80222	7292	911	22880	5799	75863	6896	954	16805	4131
Query 6	5223	474	47	1821	504	4967	451	65	1098	307
Query 7	24942	2267	37	13738	3808	16335	1485	31	4334	1590
Query 8	1919	174	28	674	174	3242	294	38	558	157
Query 9	1256	114	7	760	206	3020	274	9	794	295

InnoDB Throughput Test Summary

	Run One					Run Two				
	Total	Mean	Min	Max	StDev	Total	Mean	Min	Max	StDev
Query 1	9092065	826551	292444	973888	197578	10337815	939801	207036	1257889	293696
Query 2	6995	635	157	1012	283	7799	709	224	4044	1068
Query 3	462483	42043	920	258186	80967	121212	11019	882	57011	15108
Query 4	50773	4615	3496	5913	619	41844	3804	1774	4552	748
Query 5	164074	14915	374	91342	27988	477024	43365	982	410681	116414
Query 6	5217	474	49	1368	421	3465	315	142	634	168
Query 7	21954	1995	15	8304	2515	13511	1228	26	4252	1491
Query 8	3712	337	176	882	215	3795	345	161	866	183
Query 9	103879	9443	7	101676	29168	4470	406	8	2004	705

APPENDIX G

TPC-DS Scaled Data Set Benchmark Results

MyISAM 6GB

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	31:09.9	User Session 3 3-1	Query 9	32	Success	729
2	31:09.9	User Session 13 13-1	Query 9	23	Success	729
3	31:09.9	User Session 14 14-1	Query 2	86	Success	13275
4	31:09.9	User Session 18 18-1	Query 2	81	Success	13275
5	31:09.9	User Session 2 2-1	Query 2	136	Success	13275
6	31:09.9	User Session 20 20-1	Query 6	219	Success	16047
7	31:10.0	User Session 18 18-1	Query 6	154	Success	16047
8	31:09.9	User Session 19 19-1	Query 7	239	Success	17116
9	31:09.9	User Session 8 8-1	Query 6	280	Success	16047
10	31:10.0	User Session 2 2-1	Query 6	153	Success	16047
11	31:09.9	User Session 10 10-1	Query 6	285	Success	16047
12	31:09.9	User Session 16 16-1	Query 6	277	Success	16047
13	31:09.9	User Session 4 4-1	Query 6	300	Success	16047
14	31:10.0	User Session 14 14-1	Query 6	228	Success	16047
15	31:10.1	User Session 19 19-1	Query 9	160	Success	729
16	31:10.1	User Session 18 18-1	Query 8	258	Success	14566
17	31:10.2	User Session 10 10-1	Query 8	223	Success	14566
18	31:10.2	User Session 14 14-1	Query 8	217	Success	14566
19	31:10.2	User Session 4 4-1	Query 8	245	Success	14566
20	31:10.1	User Session 20 20-1	Query 8	339	Success	14566
21	31:10.2	User Session 8 8-1	Query 2	306	Success	13275
22	31:10.4	User Session 4 4-1	Query 2	442	Success	13275
23	31:10.5	User Session 20 20-1	Query 2	526	Success	13275
24	31:10.5	User Session 8 8-1	Query 8	551	Success	14566
25	31:09.9	User Session 1 1-1	Query 3	2508	Success	508741
26	31:09.9	User Session 7 7-1	Query 3	3469	Success	508741
27	31:12.4	User Session 1 1-1	Query 7	967	Success	17116
28	31:09.9	User Session 15 15-1	Query 5	3746	Success	373357
29	31:09.9	User Session 17 17-1	Query 5	3880	Success	373357
30	31:09.9	User Session 5 5-1	Query 5	3907	Success	373357
31	31:09.9	User Session 13 13-1	Query 3	4323	Success	508741
32	31:09.9	User Session 3 3-1	Query 3	4359	Success	508741
33	31:09.9	User Session 11 11-1	Query 3	4396	Success	508741
34	31:13.8	User Session 17 17-1	Query 9	605	Success	729

35	31:09.9	User Session 22 22-1	Query 4	4485	Success	473578
36	31:13.4	User Session 1 1-1	Query 9	1037	Success	729
37	31:10.3	User Session 19 19-1	Query 1	4099	Success	645
38	31:10.2	User Session 2 2-1	Query 4	4346	Success	473578
39	31:09.9	User Session 6 6-1	Query 4	4712	Success	473578
40	31:10.2	User Session 16 16-1	Query 4	4514	Success	473578
41	31:09.9	User Session 12 12-1	Query 4	4841	Success	473578
42	31:13.6	User Session 15 15-1	Query 7	1154	Success	17116
43	31:13.4	User Session 7 7-1	Query 7	1439	Success	17116
44	31:10.4	User Session 10 10-1	Query 4	4464	Success	473578
45	31:14.2	User Session 13 13-1	Query 7	766	Success	17116
46	31:10.4	User Session 14 14-1	Query 4	4671	Success	473578
47	31:14.4	User Session 22 22-1	Query 8	722	Success	14566
48	31:14.8	User Session 15 15-1	Query 9	372	Success	729
49	31:14.5	User Session 2 2-1	Query 8	682	Success	14566
50	31:10.4	User Session 18 18-1	Query 4	4891	Success	473578
51	31:14.6	User Session 6 6-1	Query 8	690	Success	14566
52	31:10.9	User Session 4 4-1	Query 4	4460	Success	473578
53	31:14.7	User Session 16 16-1	Query 2	676	Success	13275
54	31:14.7	User Session 12 12-1	Query 2	640	Success	13275
55	31:11.0	User Session 8 8-1	Query 4	4401	Success	473578
56	31:11.0	User Session 20 20-1	Query 4	4446	Success	473578
57	31:14.9	User Session 10 10-1	Query 2	603	Success	13275
58	31:15.1	User Session 22 22-1	Query 6	505	Success	16047
59	31:15.3	User Session 6 6-1	Query 2	419	Success	13275
60	31:15.4	User Session 16 16-1	Query 8	403	Success	14566
61	31:15.4	User Session 12 12-1	Query 8	405	Success	14566
62	31:09.9	User Session 9 9-1	Query 5	5929	Success	373357
63	31:09.9	User Session 21 21-1	Query 5	5938	Success	373357
64	31:15.6	User Session 22 22-1	Query 2	350	Success	13275
65	31:15.7	User Session 6 6-1	Query 6	339	Success	16047
66	31:15.8	User Session 12 12-1	Query 6	334	Success	16047
67	31:15.8	User Session 9 9-1	Query 7	317	Success	17116
68	31:15.9	User Session 21 21-1	Query 7	350	Success	17116
69	31:16.1	User Session 9 9-1	Query 9	158	Success	729
70	31:14.3	User Session 3 3-1	Query 5	2080	Success	373357
71	31:14.3	User Session 11 11-1	Query 5	2103	Success	373357
72	31:14.4	User Session 19 19-1	Query 5	2017	Success	373357
73	31:14.8	User Session 7 7-1	Query 5	1693	Success	373357
74	31:16.5	User Session 7 7-1	Query 9	92	Success	729
75	31:15.2	User Session 15 15-1	Query 3	1571	Success	508741

76	31:16.2	User Session 21 21-1	Query 3	808	Success	508741
77	31:17.0	User Session 21 21-1	Query 9	603	Success	729
78	31:16.4	User Session 19 19-1	Query 3	1241	Success	508741
79	31:14.4	User Session 1 1-1	Query 1	51577	Success	645
80	32:06.0	User Session 1 1-1	Query 5	8140	Success	373357
81	31:14.4	User Session 17 17-1	Query 1	90733	Success	645
82	32:45.1	User Session 17 17-1	Query 7	2559	Success	17116
83	32:47.7	User Session 17 17-1	Query 3	8663	Success	508741
84	31:13.8	User Session 5 5-1	Query 1	107593	Success	645
85	33:01.4	User Session 5 5-1	Query 7	601	Success	17116
86	33:02.0	User Session 5 5-1	Query 9	24	Success	729
87	33:02.0	User Session 5 5-1	Query 3	4991	Success	508741
88	31:16.3	User Session 9 9-1	Query 1	180796	Success	645
89	31:15.0	User Session 13 13-1	Query 1	186067	Success	645
90	34:21.1	User Session 13 13-1	Query 5	3444	Success	373357
91	34:17.1	User Session 9 9-1	Query 3	7474	Success	508741
92	31:16.4	User Session 11 11-1	Query 1	204150	Success	645
93	34:40.5	User Session 11 11-1	Query 7	1191	Success	17116
94	34:41.7	User Session 11 11-1	Query 9	2367	Success	729
95	31:16.7	User Session 15 15-1	Query 1	609455	Success	645
96	31:16.4	User Session 3 3-1	Query 1	623039	Success	645
97	41:39.4	User Session 3 3-1	Query 7	43	Success	17116
98	31:16.6	User Session 7 7-1	Query 1	624970	Success	645
99	31:17.6	User Session 21 21-1	Query 1	624733	Success	645

MyISAM 9GB

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	57:36.7	User Session 3 3-1	Query 9	22	Success	729
2	57:36.7	User Session 1 1-1	Query 7	46	Success	17106
3	57:36.7	User Session 15 15-1	Query 9	64	Success	729
4	57:36.7	User Session 11 11-1	Query 9	69	Success	729
5	57:36.7	User Session 19 19-1	Query 9	68	Success	729
6	57:36.7	User Session 4 4-1	Query 6	141	Success	16047
7	57:36.7	User Session 6 6-1	Query 6	145	Success	16047
8	57:36.7	User Session 8 8-1	Query 6	159	Success	16047
9	57:36.7	User Session 7 7-1	Query 7	166	Success	17106
10	57:36.7	User Session 3 3-1	Query 7	164	Success	17106
11	57:36.7	User Session 2 2-1	Query 8	217	Success	14574
12	57:36.7	User Session 18 18-1	Query 8	209	Success	14574
13	57:36.7	User Session 22 22-1	Query 8	211	Success	14574

14	57:36.9	User Session 7 7-1	Query 9	130	Success	729
15	57:36.7	User Session 20 20-1	Query 2	345	Success	13275
16	57:36.7	User Session 16 16-1	Query 2	357	Success	13275
17	57:36.9	User Session 18 18-1	Query 2	168	Success	13275
18	57:36.7	User Session 14 14-1	Query 2	395	Success	13275
19	57:36.7	User Session 12 12-1	Query 2	417	Success	13275
20	57:36.8	User Session 6 6-1	Query 2	281	Success	13275
21	57:36.8	User Session 8 8-1	Query 2	270	Success	13275
22	57:36.8	User Session 4 4-1	Query 2	302	Success	13275
23	57:37.1	User Session 14 14-1	Query 8	63	Success	14574
24	57:37.1	User Session 6 6-1	Query 8	62	Success	14574
25	57:37.1	User Session 8 8-1	Query 8	67	Success	14574
26	57:37.0	User Session 20 20-1	Query 8	155	Success	14574
27	57:37.1	User Session 4 4-1	Query 8	127	Success	14574
28	57:37.2	User Session 20 20-1	Query 6	911	Success	16047
29	57:37.1	User Session 12 12-1	Query 6	1104	Success	16047
30	57:37.1	User Session 16 16-1	Query 6	1165	Success	16047
31	57:38.2	User Session 12 12-1	Query 8	767	Success	14574
32	57:36.8	User Session 11 11-1	Query 5	5216	Success	373905
33	57:37.0	User Session 7 7-1	Query 5	4993	Success	373905
34	57:36.7	User Session 21 21-1	Query 5	5559	Success	373905
35	57:36.9	User Session 22 22-1	Query 4	6056	Success	474956
36	57:36.7	User Session 9 9-1	Query 3	6295	Success	509170
37	57:37.3	User Session 4 4-1	Query 4	6000	Success	474956
38	57:37.2	User Session 14 14-1	Query 4	6119	Success	474956
39	57:37.2	User Session 8 8-1	Query 4	6106	Success	474956
40	57:36.8	User Session 19 19-1	Query 3	6740	Success	509170
41	57:37.2	User Session 6 6-1	Query 4	6596	Success	474956
42	57:42.3	User Session 21 21-1	Query 7	1604	Success	17106
43	57:36.9	User Session 2 2-1	Query 4	6990	Success	474956
44	57:37.1	User Session 18 18-1	Query 4	6959	Success	474956
45	57:43.3	User Session 14 14-1	Query 6	1250	Success	16047
46	57:36.7	User Session 1 1-1	Query 3	7919	Success	509170
47	57:43.9	User Session 21 21-1	Query 9	849	Success	729
48	57:39.0	User Session 12 12-1	Query 4	5978	Success	474956
49	57:38.2	User Session 16 16-1	Query 4	6756	Success	474956
50	57:44.6	User Session 1 1-1	Query 9	383	Success	729
51	57:38.1	User Session 20 20-1	Query 4	6925	Success	474956
52	57:36.7	User Session 10 10-1	Query 4	8365	Success	474956
53	57:43.9	User Session 2 2-1	Query 2	1475	Success	13275
54	57:44.0	User Session 18 18-1	Query 6	1319	Success	16047

55	57:43.0	User Session 22 22-1	Query 6	2392	Success	16047
56	57:43.0	User Session 9 9-1	Query 7	2383	Success	17106
57	57:45.0	User Session 16 16-1	Query 8	641	Success	14574
58	57:45.4	User Session 9 9-1	Query 9	255	Success	729
59	57:45.1	User Session 10 10-1	Query 6	585	Success	16047
60	57:45.4	User Session 22 22-1	Query 2	451	Success	13275
61	57:45.3	User Session 2 2-1	Query 6	470	Success	16047
62	57:45.6	User Session 10 10-1	Query 2	479	Success	13275
63	57:43.5	User Session 19 19-1	Query 5	3004	Success	373905
64	57:46.1	User Session 10 10-1	Query 8	517	Success	14574
65	57:42.0	User Session 11 11-1	Query 3	4707	Success	509170
66	57:42.0	User Session 7 7-1	Query 3	4750	Success	509170
67	57:45.0	User Session 1 1-1	Query 5	1892	Success	373905
68	57:46.7	User Session 11 11-1	Query 7	259	Success	17106
69	57:45.6	User Session 9 9-1	Query 5	1339	Success	373905
70	57:36.7	User Session 17 17-1	Query 1	114307	Success	623
71	59:31.0	User Session 17 17-1	Query 3	26768	Success	509170
72	57:36.8	User Session 15 15-1	Query 1	169124	Success	645
73	59:57.8	User Session 17 17-1	Query 5	32375	Success	373905
74	00:30.1	User Session 17 17-1	Query 7	4311	Success	17106
75	00:34.5	User Session 17 17-1	Query 9	551	Success	729
76	00:25.9	User Session 15 15-1	Query 3	22442	Success	509170
77	00:48.3	User Session 15 15-1	Query 7	6955	Success	17106
78	57:36.7	User Session 5 5-1	Query 1	207363	Success	645
79	00:55.3	User Session 15 15-1	Query 5	17354	Success	373905
80	01:04.0	User Session 5 5-1	Query 5	9491	Success	373905
81	01:13.5	User Session 5 5-1	Query 3	28412	Success	509170
82	01:42.0	User Session 5 5-1	Query 7	8328	Success	17106
83	01:50.3	User Session 5 5-1	Query 9	2434	Success	729
84	57:44.7	User Session 21 21-1	Query 1	306817	Success	623
85	57:36.7	User Session 13 13-1	Query 1	322127	Success	645
86	02:51.5	User Session 21 21-1	Query 3	13815	Success	509170
87	02:58.8	User Session 13 13-1	Query 3	20180	Success	509170
88	03:19.0	User Session 13 13-1	Query 5	16095	Success	373905
89	03:35.1	User Session 13 13-1	Query 7	5264	Success	17106
90	03:40.4	User Session 13 13-1	Query 9	1492	Success	729
91	57:46.7	User Session 7 7-1	Query 1	394558	Success	645
92	57:36.9	User Session 3 3-1	Query 1	556469	Success	645
93	57:46.9	User Session 1 1-1	Query 1	550676	Success	645
94	06:53.3	User Session 3 3-1	Query 3	4436	Success	509170
95	06:57.8	User Session 3 3-1	Query 5	2185	Success	373905

96	57:47.0	User Session 9 9-1	Query 1	580093	Success	645
97	57:46.5	User Session 19 19-1	Query 1	582480	Success	645
98	07:29.0	User Session 19 19-1	Query 7	35	Success	17106
99	57:47.0	User Session 11 11-1	Query 1	582845	Success	645

InnoDB 6GB

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	07:47.8	User Session 7 7-1	Query 9	76	Success	729
2	07:47.8	User Session 14 14-1	Query 8	140	Success	14566
3	07:47.9	User Session 20 20-1	Query 6	203	Success	16047
4	07:47.8	User Session 12 12-1	Query 2	251	Success	13275
5	07:47.8	User Session 2 2-1	Query 8	284	Success	14566
6	07:47.8	User Session 10 10-1	Query 6	271	Success	16047
7	07:47.8	User Session 13 13-1	Query 7	275	Success	17116
8	07:47.8	User Session 4 4-1	Query 2	291	Success	13275
9	07:47.8	User Session 18 18-1	Query 8	278	Success	14566
10	07:47.8	User Session 21 21-1	Query 7	320	Success	17116
11	07:47.8	User Session 8 8-1	Query 2	359	Success	13275
12	07:47.8	User Session 16 16-1	Query 2	345	Success	13275
13	07:48.1	User Session 13 13-1	Query 9	151	Success	729
14	07:48.2	User Session 21 21-1	Query 9	88	Success	729
15	07:48.1	User Session 10 10-1	Query 2	462	Success	13275
16	07:48.1	User Session 18 18-1	Query 6	429	Success	16047
17	07:48.1	User Session 2 2-1	Query 2	464	Success	13275
18	07:47.8	User Session 1 1-1	Query 3	809	Success	508741
19	07:48.0	User Session 14 14-1	Query 2	722	Success	13275
20	07:48.1	User Session 4 4-1	Query 6	604	Success	16047
21	07:48.2	User Session 16 16-1	Query 8	567	Success	14566
22	07:48.6	User Session 1 1-1	Query 9	140	Success	729
23	07:48.6	User Session 18 18-1	Query 2	321	Success	13275
24	07:48.7	User Session 14 14-1	Query 6	386	Success	16047
25	07:47.8	User Session 9 9-1	Query 5	2138	Success	373357
26	07:47.8	User Session 19 19-1	Query 5	3347	Success	373357
27	07:50.0	User Session 9 9-1	Query 7	1367	Success	17116
28	07:47.8	User Session 11 11-1	Query 5	3748	Success	373357
29	07:47.8	User Session 6 6-1	Query 4	3933	Success	473578
30	07:47.8	User Session 3 3-1	Query 3	4211	Success	508741
31	07:51.2	User Session 19 19-1	Query 7	958	Success	17116
32	07:48.1	User Session 20 20-1	Query 4	4583	Success	473578
33	07:52.2	User Session 19 19-1	Query 9	508	Success	729

34	07:47.9	User Session 22 22-1	Query 4	4884	Success	473578
35	07:51.7	User Session 6 6-1	Query 6	996	Success	16047
36	07:47.8	User Session 15 15-1	Query 3	5189	Success	508741
37	07:48.7	User Session 4 4-1	Query 4	4446	Success	473578
38	07:48.6	User Session 2 2-1	Query 4	4611	Success	473578
39	07:48.2	User Session 8 8-1	Query 4	5024	Success	473578
40	07:48.1	User Session 12 12-1	Query 4	5132	Success	473578
41	07:48.6	User Session 10 10-1	Query 4	4679	Success	473578
42	07:48.3	User Session 21 21-1	Query 3	5039	Success	508741
43	07:48.3	User Session 13 13-1	Query 3	5057	Success	508741
44	07:48.9	User Session 18 18-1	Query 4	4512	Success	473578
45	07:48.8	User Session 16 16-1	Query 4	4689	Success	473578
46	07:52.8	User Session 22 22-1	Query 6	712	Success	16047
47	07:52.7	User Session 20 20-1	Query 2	807	Success	13275
48	07:52.8	User Session 6 6-1	Query 8	719	Success	14566
49	07:49.1	User Session 14 14-1	Query 4	4411	Success	473578
50	07:53.0	User Session 15 15-1	Query 7	524	Success	17116
51	07:53.2	User Session 2 2-1	Query 6	391	Success	16047
52	07:53.2	User Session 12 12-1	Query 8	349	Success	14566
53	07:53.2	User Session 8 8-1	Query 6	364	Success	16047
54	07:53.2	User Session 4 4-1	Query 8	406	Success	14566
55	07:53.2	User Session 10 10-1	Query 8	344	Success	14566
56	07:53.5	User Session 22 22-1	Query 8	771	Success	14566
57	07:53.5	User Session 20 20-1	Query 8	777	Success	14566
58	07:53.5	User Session 6 6-1	Query 2	769	Success	13275
59	07:53.6	User Session 8 8-1	Query 8	837	Success	14566
60	07:53.5	User Session 16 16-1	Query 6	989	Success	16047
61	07:52.0	User Session 3 3-1	Query 5	2434	Success	373357
62	07:54.2	User Session 22 22-1	Query 2	233	Success	13275
63	07:53.6	User Session 12 12-1	Query 6	899	Success	16047
64	07:51.6	User Session 11 11-1	Query 3	3125	Success	508741
65	07:53.3	User Session 21 21-1	Query 5	1498	Success	373357
66	07:53.3	User Session 13 13-1	Query 5	1484	Success	373357
67	07:47.8	User Session 17 17-1	Query 1	282068	Success	623
68	12:29.9	User Session 17 17-1	Query 5	20730	Success	373357
69	12:50.6	User Session 17 17-1	Query 3	25348	Success	508741
70	13:16.0	User Session 17 17-1	Query 7	2989	Success	17116
71	13:19.0	User Session 17 17-1	Query 9	1427	Success	729
72	07:48.8	User Session 1 1-1	Query 1	370718	Success	623
73	13:59.5	User Session 1 1-1	Query 5	15861	Success	373357
74	14:15.3	User Session 1 1-1	Query 7	4413	Success	17116

75	07:47.9	User Session 7 7-1	Query 1	444617	Success	645
76	15:12.5	User Session 7 7-1	Query 5	20576	Success	373357
77	07:53.6	User Session 15 15-1	Query 1	506332	Success	623
78	16:19.9	User Session 15 15-1	Query 9	874	Success	729
79	15:33.1	User Session 7 7-1	Query 3	52506	Success	508741
80	16:25.6	User Session 7 7-1	Query 7	4362	Success	17116
81	16:20.8	User Session 15 15-1	Query 5	14512	Success	373357
82	07:54.7	User Session 11 11-1	Query 1	565027	Success	645
83	17:19.7	User Session 11 11-1	Query 7	2262	Success	17116
84	17:22.0	User Session 11 11-1	Query 9	807	Success	729
85	07:47.8	User Session 5 5-1	Query 1	577662	Success	645
86	17:25.5	User Session 5 5-1	Query 3	6789	Success	508741
87	17:32.3	User Session 5 5-1	Query 7	1244	Success	17116
88	17:33.5	User Session 5 5-1	Query 5	7301	Success	373357
89	17:40.8	User Session 5 5-1	Query 9	7	Success	729
90	07:54.8	User Session 21 21-1	Query 1	590112	Success	645
91	07:52.7	User Session 19 19-1	Query 1	620857	Success	645
92	18:13.5	User Session 19 19-1	Query 3	5769	Success	508741
93	07:54.5	User Session 3 3-1	Query 1	628323	Success	645
94	18:22.8	User Session 3 3-1	Query 7	641	Success	17116
95	18:23.4	User Session 3 3-1	Query 9	7	Success	729
96	07:54.8	User Session 13 13-1	Query 1	630486	Success	645
97	07:51.3	User Session 9 9-1	Query 1	634754	Success	645
98	18:26.1	User Session 9 9-1	Query 9	8	Success	729
99	18:26.1	User Session 9 9-1	Query 3	295	Success	508741

InnoDB 9GB

Test	Start Time	Session	Query	Execution Time	Result	Bytes
1	40:21.2	User Session 5 5-1	Query 9	11	Success	729
2	40:21.2	User Session 19 19-1	Query 9	15	Success	729
3	40:21.2	User Session 21 21-1	Query 9	35	Success	729
4	40:21.2	User Session 17 17-1	Query 7	53	Success	17106
5	40:21.1	User Session 6 6-1	Query 8	1145	Success	14574
6	40:21.2	User Session 8 8-1	Query 2	1139	Success	13275
7	40:21.2	User Session 22 22-1	Query 8	1089	Success	14574
8	40:22.3	User Session 22 22-1	Query 2	86	Success	13275
9	40:22.3	User Session 6 6-1	Query 2	132	Success	13275
10	40:21.1	User Session 4 4-1	Query 6	1650	Success	16047
11	40:22.4	User Session 6 6-1	Query 6	378	Success	16047
12	40:21.2	User Session 20 20-1	Query 6	1610	Success	16047

13	40:21.2	User Session 10 10-1	Query 6	1652	Success	16047
14	40:21.2	User Session 18 18-1	Query 6	1615	Success	16047
15	40:21.2	User Session 5 5-1	Query 5	2478	Success	373905
16	40:22.8	User Session 10 10-1	Query 8	952	Success	14574
17	40:22.8	User Session 18 18-1	Query 2	959	Success	13275
18	40:22.8	User Session 4 4-1	Query 8	983	Success	14574
19	40:22.8	User Session 20 20-1	Query 8	1097	Success	14574
20	40:23.8	User Session 10 10-1	Query 2	1895	Success	13275
21	40:23.8	User Session 18 18-1	Query 8	2218	Success	14574
22	40:21.2	User Session 11 11-1	Query 3	5109	Success	509170
23	40:23.6	User Session 5 5-1	Query 7	2713	Success	17106
24	40:21.2	User Session 16 16-1	Query 4	5176	Success	474956
25	40:26.4	User Session 16 16-1	Query 2	759	Success	13275
26	40:26.3	User Session 11 11-1	Query 7	896	Success	17106
27	40:21.1	User Session 2 2-1	Query 4	6251	Success	474956
28	40:21.2	User Session 13 13-1	Query 5	6241	Success	373905
29	40:27.2	User Session 11 11-1	Query 9	1470	Success	729
30	40:27.4	User Session 13 13-1	Query 9	1238	Success	729
31	40:21.1	User Session 1 1-1	Query 3	7682	Success	509170
32	40:22.3	User Session 8 8-1	Query 4	6550	Success	474956
33	40:21.2	User Session 12 12-1	Query 4	7767	Success	474956
34	40:28.8	User Session 1 1-1	Query 9	341	Success	729
35	40:27.4	User Session 2 2-1	Query 8	1782	Success	14574
36	40:21.1	User Session 7 7-1	Query 3	8064	Success	509170
37	40:22.4	User Session 22 22-1	Query 4	6919	Success	474956
38	40:28.9	User Session 8 8-1	Query 8	1095	Success	14574
39	40:28.9	User Session 12 12-1	Query 6	1010	Success	16047
40	40:27.1	User Session 16 16-1	Query 6	2814	Success	16047
41	40:29.3	User Session 22 22-1	Query 6	775	Success	16047
42	40:30.0	User Session 12 12-1	Query 8	834	Success	14574
43	40:30.0	User Session 16 16-1	Query 8	839	Success	14574
44	40:29.2	User Session 2 2-1	Query 6	1608	Success	16047
45	40:30.0	User Session 8 8-1	Query 6	844	Success	16047
46	40:30.8	User Session 2 2-1	Query 2	1318	Success	13275
47	40:30.8	User Session 12 12-1	Query 2	1808	Success	13275
48	40:23.8	User Session 4 4-1	Query 4	10016	Success	474956
49	40:23.9	User Session 20 20-1	Query 4	9895	Success	474956
50	40:33.8	User Session 4 4-1	Query 2	1203	Success	13275
51	40:26.0	User Session 18 18-1	Query 4	9066	Success	474956
52	40:33.8	User Session 20 20-1	Query 2	1695	Success	13275
53	40:21.2	User Session 14 14-1	Query 4	14330	Success	474956

54	40:22.8	User Session 6 6-1	Query 4	12729	Success	474956
55	40:35.5	User Session 14 14-1	Query 8	1165	Success	14574
56	40:25.7	User Session 10 10-1	Query 4	11072	Success	474956
57	40:36.7	User Session 14 14-1	Query 2	778	Success	13275
58	40:28.7	User Session 13 13-1	Query 3	8864	Success	509170
59	40:37.5	User Session 14 14-1	Query 6	483	Success	16047
60	40:37.5	User Session 13 13-1	Query 7	2403	Success	17106
61	40:29.2	User Session 1 1-1	Query 1	106915	Success	623
62	42:16.1	User Session 1 1-1	Query 5	34380	Success	373905
63	42:50.5	User Session 1 1-1	Query 7	5880	Success	17106
64	40:21.1	User Session 3 3-1	Query 1	368328	Success	623
65	46:29.5	User Session 3 3-1	Query 3	73009	Success	509170
66	47:42.5	User Session 3 3-1	Query 5	232323	Success	623
67	40:39.9	User Session 13 13-1	Query 1	654921	Success	623
68	51:34.8	User Session 3 3-1	Query 7	4352	Success	17106
69	51:39.2	User Session 3 3-1	Query 9	2219	Success	729
70	40:21.2	User Session 19 19-1	Query 1	754696	Success	645
71	52:55.9	User Session 19 19-1	Query 3	8368	Success	509170
72	53:04.3	User Session 19 19-1	Query 5	19784	Success	373905
73	40:21.2	User Session 15 15-1	Query 1	786823	Success	645
74	53:24.1	User Session 19 19-1	Query 7	4698	Success	17106
75	53:28.0	User Session 15 15-1	Query 5	7740	Success	373905
76	53:35.7	User Session 15 15-1	Query 7	847	Success	17106
77	53:36.6	User Session 15 15-1	Query 9	544	Success	729
78	53:37.1	User Session 15 15-1	Query 3	10816	Success	509170
79	40:21.3	User Session 17 17-1	Query 1	807185	Success	645
80	53:48.4	User Session 17 17-1	Query 9	476	Success	729
81	53:48.9	User Session 17 17-1	Query 5	10001	Success	373905
82	53:58.9	User Session 17 17-1	Query 3	17664	Success	509170
83	40:29.2	User Session 7 7-1	Query 1	846596	Success	645
84	54:35.8	User Session 7 7-1	Query 5	4702	Success	373905
85	54:40.5	User Session 7 7-1	Query 7	772	Success	17106
86	54:41.3	User Session 7 7-1	Query 9	10	Success	729
87	40:21.3	User Session 21 21-1	Query 1	865503	Success	645
88	54:46.8	User Session 21 21-1	Query 5	5597	Success	373905
89	54:52.3	User Session 21 21-1	Query 7	1490	Success	17106
90	54:53.8	User Session 21 21-1	Query 3	9266	Success	509170
91	40:28.7	User Session 11 11-1	Query 1	887022	Success	645
92	55:15.7	User Session 11 11-1	Query 5	1545	Success	373905
93	40:21.2	User Session 9 9-1	Query 1	896923	Success	645
94	55:18.1	User Session 9 9-1	Query 5	251	Success	373905

95	40:26.4	User Session 5 5-1	Query 1	892588	Success	645
96	55:18.3	User Session 9 9-1	Query 9	619	Success	729
97	55:19.0	User Session 5 5-1	Query 3	561	Success	509170
98	55:19.0	User Session 9 9-1	Query 3	562	Success	509170
99	55:19.5	User Session 9 9-1	Query 7	27	Success	17106

MyISAM Scaled Data Set Benchmark Summary

	6GB							
	Total	Min	Q1	Mean	Median	Q3	Max	SD
Query 1	3307212	4099	90733	300656	186067	623039	624970	260313
Query 2	4265	81	136	387.7	419	603	676	217.4
Query 5	42877	1693	2080	3898	3746	5929	8140	2036
Query 7	9626	43	317	875	766	1191	2559	715
Query 8	4735	217	245	430.5	404	682	722	198.2
Query 9	5473	23	32	498	160	605	2367	699

	9GB							
	Total	Min	Q1	Mean	Median	Q3	Max	SD
Query 1	4366859	114307	207363	396987	394558	580093	582845	182500
Query 2	4940	168	281	449	357	451	1475	352
Query 5	99503	1339	2185	9046	5126	16095	32375	9490
Query 7	29515	35	164	2683	1604	5264	8328	3053
Query 8	3036	62	67	276	209	517	767	248.2
Query 9	6317	22	68	574	255	849	2434	761

	12GB							
	Total	Min	Q1	Mean	Median	Q3	Max	SD
Query 1	6609219	189528	401080	600838	647326	870768	582845	247588
Query 2	2783	55	89	253	130	493	621	214
Query 5	80222	911	3846	7293	6339	8667	22880	6082
Query 7	42942	37	365	2267	755	1956	13738	3994
Query 8	1919	28	85	174.5	103	268	674	182.8
Query 9	1256	7	33	114.2	58	77	760	215.8

InnoDB Scaled Data Set Benchmark Summary

	6GB							
	Total	Min	Q1	Mean	Median	Q3	Max	SD
Query 1	5850956	282068	444617	531905	577662	628323	634754	118641
Query 2	5024	233	291	456.7	359	722	807	212.4
Query 5	93629	1848	2138	8512	3748	15816	20730	7822
Query 7	19355	275	524	1760	1244	2898	4413	1539
Query 8	5472	140	284	479.5	406	771	837	244.5
Query 9	4093	7	8	372	140	807	1427	474

	9GB							
	Total	Min	Q1	Mean	Median	Q3	Max	SD
Query 1	7867500	106915	654921	715227	807185	887022	896923	253571
Query 2	11772	86	759	1070	1139	1695	1895	612
Query 5	325042	251	2478	29549	6241	19784	232323	67973
Query 7	24131	27	772	1996	1490	4352	5880	2003
Query 8	13199	434	952	1200	1095	1165	2218	423
Query 9	6978	10	15	634	476	1238	2219	722

	12GB							
	Total	Min	Q1	Mean	Median	Q3	Max	SD
Query 1	9092065	292444	841002	826551	886710	968163	973888	207221
Query 2	6995	157	239	635.9	711	835	1012	296.6
Query 5	164074	374	1101	14916	1393	6815	91342	29354
Query 7	21954	15	302	1996	1024	1937	8304	2637
Query 8	3712	176	224	337.5	241	299	882	226
Query 9	103879	7	25	9444	39	620	101676	3051

VITA

GABRIEL TOCCI

Education:	Oliver Ames High School, North Easton, MA. Hesser College, Manchester, NH. Network Engineering, A.S., 2003 East Tennessee State University, Johnson City, TN. Computer and Information Science, B.S., 2008 East Tennessee State University, Johnson City, TN. Computer and Information Science, M.S., 2013
Professional Experience:	Service Technician, Qualxserv LLC, Tewksbury, MA. 2004-2007 Graduate Assistant, E.T.S.U., Johnson City, TN. 2008-2009 Programmer/Systems Analyst, E.T.S.U., Johnson City, TN. 2009-2013