

DEVELOPING A FILTER KIT SYSTEM DATABASE: PROCEDURE AND
ANALYSIS

A Thesis

Presented to

the Faculty of the College of Science and Technology
Morehead State University

In Partial Fulfillment

of the Requirements for the Degree

Master of Science

by

He Shi

April 30, 2010

MSU
THESES
005.74
S555d

Accepted by the faculty of the College of Science and Technology, Morehead State University, in partial fulfillment of the requirements for the Master of Science degree.

Ahmad Zargai

Director of Thesis

Master's Committee: Ahmad Zargai, Chair

Yaqi You

C. Steven Hunt

May 7, 2010

Date

DEVELOPING A FILTER KIT SYSTEM DATABASE: PROCEDURE AND ANALYSIS

He Shi, M.S.
Morehead State University, 2010

Director of Thesis: Ahmad Zayyan

As business operations become more and more complex, the demand for change in Information Technology (IT) increases accordingly with associated risks that must be mitigated. Today's IT engineers are being asked to manage more information, and deliver that information to their users, with an ever increasing quality of service, in a timely manner. In order to help CLARK Material Handling Company meet the growing needs of engineers to access the server main database, this filter kit system database was proposed and designed to help users to access data from CLARK's server through the intranet.

This filter kit system database has two advantages. The first advantage is that the solution is easy to install and configure. The second advantage is that the solution is user-friendly. It is tightly integrated with Microsoft products, which means that employees can access its information in the Microsoft tools they use every day.

Rely on Windows Server 2008, SQL Server 2008, and Visual Studio 2008

provide a secure and trusted foundation, the filter kit system database offers advanced technology support for the latest platforms, improved management and Web tools, and access to relevant information throughout the intranet.

Accepted by:

Ahmad Zayari, Chair
Yagci Yoz
C. Steven Hunt

ACKNOWLEDGEMENT

The author wishes to express his gratefulness to all those individuals who facilitated their knowledge in completion of this thesis. Many people supported and guided during the course of completion of the thesis, their help is greatly appreciated.

Sincere appreciation is expressed to Dr. Ahmad Zargari, my Thesis Director, for providing technical guidance from the beginning of the research to its completion. Dr. Zargari was always available to make reviews and answer my queries. His constructive criticism and guidance during the thesis has been of great assistance.

Heartfelt gratitude is expressed to Dr. Yuqiu You. Her useful and effective assistance was always a positive help toward finishing this thesis. Without her, this thesis would not have been completed. Dr. You's continuous encouragement and support has been invaluable in completion of the thesis.

Honest thankfulness is expressed to Dr. Steve Hunt for his constructive comments and support during this thesis. He shared his academic and technical experiences from the beginning of the research to its completion. Even though, Dr. Hunt had very busy schedule, he was always available to make reviews and give suggestions.

Last but not least, I would also like to thank my wife and daughter for their constant support and encouragement. They have been a great moral support through the whole procedure of completion of this thesis.

TABLE OF CONTENTS

CHAPTER I: INTRODUCTION	1
Objectives	1
Significance of The Study	2
Definition of Terms.....	4
CHAPTER II: REVIEW OF LITERATURE	7
Review of Literature	7
Historical background	7
Related Research.....	23
CHAPTER III: METHODOLOGY	29
Restatement of Objectives	29
Research Instrument and Procedure.....	30
CHAPTER IV: CASE STUDY	35
Case Study	35
CHAPTER V: FINDINGS	43
Findings.....	43
Conclusion	44
Future Work.....	45
Reference	46
APPENDIX A	48
APPENDIX B	53

APPENDIX C	59
APPENDIX D	69
APPENDIX E	71

CHAPTER I

Introduction

As business operations become more and more complex, the demand for change in Information Technology (IT) increases accordingly with associated risks that must be mitigated. Today's IT engineers are being asked to manage more information, and deliver that information to their users, with an ever increasing quality of service, in a timely manner. In today's economic climate, IT is additionally tasked with reducing budgets and deriving greater value out of its existing investments. Before this filter kit system database is built, engineers only can find what they need by opening different Excel files and searching through them. The main purpose of this research was to provide a filter kit system database for engineers to access data from the server. This database can help engineers save time finding what they need and also help database administrators successfully deliver more information with higher quality of service, reduce the risk of change within IT, and make more efficient use of their IT budgets.

Objectives

The purpose of this study was to develop and analyze a filter kit system database for CLARK Material Handling Company. This study started with research of Database Management System (DBMS) and an introduction of several commonly

used DBMS. Because the server of CLARK Material Handling Company uses WINDOWS SERVER 2003 as the operation system, there are only two DBMS (SQL Server 2008 and MySQL version 5.1) that can be chosen. This study used SQL Server 2008 to generate the filter kit system database and compared it with the database made with MySQL version 5.1. Upon completion of the study, the following objectives were achieved:

Objective 1

Generate a filter kit system database for CLARK Material Handling Company and demonstrate several major design procedures.

Objective 2

Evaluate and analyze the filter kit system database. Demonstrate capability, advantages and disadvantages of this database.

Significance of the Study

In today's dynamic global economy, organizations are bombarded with the pressures of hyper-competitive markets. They can no longer take a "wait and see" approach. Rather, they need the ability to innovate faster and react quicker to stay one step ahead of the competition. In order to remain competitive, organizations must address three key challenges. First, they need to make it easy for users to access and

interact with all of the information sources and applications they need in order to be productive. Second, companies need to be able to enhance their portals and enterprise applications with rich Web 2.0 services to present a more comprehensive view across business processes. Finally, organizations need IT departments to deliver upgrades, customizations and new applications faster, as well as give users the ability to customize and personalize applications on their own in order to respond faster to the dynamic market.

In order to help CLARK Material Handling Company meet the growing needs of engineers to access the server main database, this filter kit system database is proposed and designed to help users access data from CLARK's server through the intranet. This database can be integrated into the main database. All the data in the database can be read and saved as a Microsoft Excel file. This database can allow the database administrator to easily manage and update because normal users only can visit the database through the Webpage different from the administrator's and has no permission to modify. Before the company can provide its employees with the kind of data-rich automated solutions they need, it has to face a problem that engineers only can find what they need by opening different Excel files again and again.

Definition of Terms

Database

A database system is an organized pool of logically-related data. Data is stored within the data structures of the database.

DBMS

A database management system (DBMS) is system software used to manage the organization, storage, access, security and integrity of data in a structured database.

The nature of database management systems has dramatically changed since the 1960 as the demand for data storage has increased and the technology to store data has plummeted.

RDBMS

A relational database management system (RDBMS) is a database management system (DBMS) that is based on the relational model as introduced by E. F. Codd. Most popular commercial and open source databases currently in use are based on the relational model.

SQL SERVER

Microsoft's SQL Server is a relational database management system that initially was targeted at small and medium sized businesses. Today, many large corporations

are using it for their enterprise information needs. The first version of SQL Server was launched in 1989. The current version of this DBMS is Microsoft SQL Server 2008. The SQL Server 2008 DBMS provides a secure, reliable, scalable, highly available relational database engine with improved performance and support for structured and unstructured (XML) data.

MYSQL

MySQL is a popular database management system that is used in some six million installations. It is widely used for web applications, especially in concert with PHP. The MySQL DBMS is owned and sponsored by the company MySQL AB. The company sells support and service contracts, as well as commercial licenses. The company makes MySQL available as free software and also under traditional commercial licensing arrangements for cases where the intended use is incompatible with use of the General Public License.

ORACLE

The Oracle Corporation launched its first commercial SQL relational database management systems in 1979. The DBMS runs on UNIX, Linux and Windows platforms and is widely used around the world. Oracle is the world's largest enterprise

software company and sells many applications system packages that run on its Oracle database.

DB2

DB2 is one of the families of relational database management system software products within IBM's broader Information Management Software line. Although there are different "editions" and "versions" of DB2, which run on devices ranging from handhelds to mainframes, most often DB2 refers to the DB2 Enterprise Server Edition, which runs on Unix (AIX), Windows, and Linux and z/OS servers.

CHAPTER II

Review of Literature

This chapter discusses basic concepts involved in a Database Management System. The chapter gives detailed information regarding related research, historical background and concepts of DBMS.

Historical background

A database can be defined as a “self-describing collection of integrated records” (Kroenke, 1995). This definition is one of the more succinct and thorough, but it is not necessarily standard, nor is it unique, although most distinct definitions have much commonality. However, a database is basically a way to capture, store, organize, and present data as information, typically for business, although many other uses for databases and database technologies have emerged since formally defined databases were created.

A formal database is comprised of several elements. First, the data itself is housed in data structures usually called files or tables. Second a piece (or pieces) of software called the DBMS (Database Management System) handles several critical tasks such as how the data is managed (input, processed, output, and stored), basic security, and replication/backup, among others. Finally another piece of software, the

user interface application handles how users interact with the database. Some database products have merged the DBMS with the user interface, or allow creation of a custom interface.

Research by Sam and Toby states that a Database Management System (DBMS) is a set of computer programs that controls the creation, maintenance, and the use of the database with computers as a platform for an organization and its end users. It allows organizations to place control of organization-wide database development in the hands of database administrators (DBAs) and other specialists. A DBMS is a system software package that helps the use of integrated collection of data records and files known as databases. It allows different user application programs to easily access the same database. DBMSs may use any of a variety of database models, such as the network model or relational model. In large systems, a DBMS allows users with other software to store and retrieve data in a structured way. Instead of having to write computer programs to extract information, a user can ask simple questions in a query language. Thus, many DBMS packages provide Fourth-generation programming language (4GLs) and other application development features. It helps to specify the logical organization for a database and access and use the information

within a database. It provides facilities for controlling data access, enforcing data integrity, managing concurrency controlled, and restoring database (Sam, 2006).

A common use of a database system is to track information about users, their name, login information, various addresses and phone numbers. In the navigational approach all of these data would be placed in a single record, and unused items would simply not be placed in the database. In the relational approach, the data would be normalized into a user table, an address table and a phone number table (for instance). Records would be created in these optional tables only if the address or phone numbers were actually provided (Codd, 2002).

DB2 DBMS

IBM has developed a complete family of relational database management system (RDBMS) software now called DB2 Universal Database™ (DB2 UDB) since 1970 (Jeff, 2003). In addition, IBM has built additional information management software with DB2 as its "engine" for purposes that include data warehousing, data analysis, data mining, media asset management, enterprise content management and information integration. DB2 UDB and the IBM DB2 information management portfolio represent one element of what IBM calls middleware, software that serves

as the glue uniting systems and software applications. DB2 is one of five IBM Software brands.

IBM markets different "editions" of DB2 with different licensing arrangements.

The cost of ownership therefore is based on the options that are needed. DB2

Editions include the following:

DB2Workgroup

DB2 Workgroup Unlimited

DB2 Enterprise Server Edition

DB2 UDB Data Warehouse Enterprise Edition (DWE)

The DB2 UDB Data Warehouse Enterprise Edition includes business intelligence capabilities such as ET, data mining and online analytics. DB2 has APIs for many languages including the following: .NET, CLI, Java, Python, Perl, PHP, C++, C, REXX, PL/1, Cobol, RPG, Fortran.

Oracle Database

According to the Oracle (2009), the Oracle Database is a relational database management system (RDBMS) produced and marketed by Oracle Corporation. An Oracle database is a collection of data treated as a unit. The purpose of a database is to store and retrieve related information. A database server is the key to solving the

problems of information management. In general, a server reliably manages a large amount of data in a multiuser environment so that many users can concurrently access the same data. All this is accomplished while delivering high performance. A database server also prevents unauthorized access and provides efficient solutions for failure recovery.

Oracle (2009) also claims, "Oracle Database is the first database designed for enterprise grid computing, the most flexible and cost effective way to manage information and applications. Enterprise grid computing creates large pools of industry-standard, modular storage and servers. With this architecture, each new system can be rapidly provisioned from the pool of components. There is no need for peak workloads, because capacity can be easily added or reallocated from the resource pools as needed."

Oracle includes several software mechanisms to fulfill the following important requirements of an information management system:

Data concurrency of a multiuser system must be maximized.

Data must be read and modified in a consistent fashion. The data that a user is viewing or changing is not changed (by other users) until the user is finished with the data.

High performance is required for maximum productivity from the many users of the database system.

Microsoft SQL Server 2008

SQL Server 2008, the most famous database management system, delivers on Microsoft's Data Platform vision by helping organizations to manage any data, any place, and any time. It also can store data from structured, semi-structured, and unstructured documents, such as images and rich media, directly within the database. SQL Server 2008 delivers a plenty set of integrated services that enable a user to do more with data such as query, search, synchronize, report, and analyze.

According to the Bteksoftware (2009), SQL Server 2008 provides the highest levels of security, reliability, and scalability for business-critical applications. To take advantage of new opportunities in today's fast-moving business world, companies need the ability to create and deploy data-driven solutions quickly. SQL Server 2008 reduces time and cost of management and development of applications.

SQL Server 2008 accesses data from across enterprise and provides control over data no matter where it's stored—from the largest servers within the data center to desktops to mobile devices. SQL Server 2008 provides a comprehensive platform that delivers intelligence where users want it. Information workers can access data directly

using the tools they use every day, such as the 2007 Microsoft Office system (Bteksoftware, 2009).

The current version of SQL Server, SQL Server 2008, (code-named "Katmai"), was released (RTM) on August 6, 2008 and aims to make data management self-tuning, self organizing, and self maintaining with the development of SQL Server Always On technologies, to provide near-zero downtime. SQL Server 2008 also includes support for structured and semi-structured data, including digital media formats for pictures, audio, video and other multimedia data. In current versions, such multimedia data can be stored as BLOBs (binary large objects), but they are generic bit streams. Intrinsic awareness of multimedia data will allow specialized functions to be performed on them. According to Paul Flessner, senior Vice President, Server Applications, Microsoft Corp., "SQL Server 2008 can be a data storage backend for different varieties of data: XML, email, time/calendar, file, document, spatial, etc as well as perform search, query, analyze, sharing, and synchronization across all data types" (Microsoft, 2009).

Other new data types include specialized date and time types and a spatial data type for location-dependent data. Better support for unstructured and semi-structured data is provided using the new FILESTREAM data type, which can be used to

reference any file stored on the file system. Structured data and metadata about the file are stored in SQL Server database, whereas the unstructured component is stored in the file system. Such files can be accessed via Win32 file handling APIs as well as via SQL Server using T-SQL; doing the latter accesses the file data as a BLOB. Backing up and restoring the database backs up or restores the referenced files as well. SQL Server 2008 also natively supports hierarchical data, and includes T-SQL constructs to directly deal with them, without using recursive queries (Microsoft, 2009).

Research by Informationweek (2007) supports The Full-Text Search functionality has been integrated with the database engine, which simplifies management and improves performance. Spatial data will be stored in two types. A "Flat Earth" (GEOMETRY or planar) data type represents geospatial data which has been projected from its native, spherical, coordinate system into a plane. A "Round Earth" data type (GEOGRAPHY) uses an ellipsoidal model in which the Earth is defined as a single continuous entity which does not suffer from the singularities such as the international dateline, poles, or map projection zone "edges". Approximately seventy methods are available to represent spatial operations for the Open Geospatial Consortium Simple Features for SQL, Version 1.1.

Fernando's (2008) study found the following: SQL Server includes better compression features, which also helps in improving scalability. It also includes Resource Governor that allows reserving resources for certain users or workflows. It also includes capabilities for transparent encryption of data as well as compression of backups. SQL Server 2008 supports the ADO.NET Entity Framework and the reporting tools, replication, and data definition will be built around the Entity Data Model.

According to (Fernando, 2008), there are some advantages of SQL Server 2008:

Resource Governor: SQL Server 2008 enables organizations to provide a consistent and predictable response to users with the new Resource Governor feature.

Resource Governor enables defining resource limits and priorities for different workloads, and helps ensure resources can't be unduly impacted by poorly constructed queries or other unusual workloads.

Backup Compression: With SQL Server 2008 Backup Compression, the compression is performed in memory before the data is transferred to disk.

Backups run significantly faster since fewer disks I/O are required. Backup compression reduces the storage required to keep backups online, reducing the overall cost of keeping disk-based backups.

Data Compression: SQL Server 2008 helps organizations store more data on hard drives, reducing loads on storage systems. That same feature also reduces the number of disk reads that databases must perform—which can degrade performance—by incorporating more data into system memory.

Filestream: Filestream is a new feature of Microsoft SQL Server 2008 that integrates the SQL Server database engine with an NTFS file system by storing binary large object (BLOB) data as files on the file system.

Change Data Capture: New for SQL Server 2008, Change Data Capture collects the complete content of changes, and places them in change tables. The feature maintains cross-table consistency and even works across schema changes. Change Data Capture can be used especially for the big tables that are used with reporting solution.

Enhanced Data Auditing: SQL Server 2008 provides comprehensive data auditing to help organizations monitor all events at the server and database levels, and scale-out auditing across the enterprise.

Transparent Data Encryption: SQL Server 2008 includes the transparent data encryption that enables organizations to encrypt data in an entire database without the need for application code changes.

Performance Data Collection: SQL Server 2008 provides Performance Studio, an integrated framework to collect, analyze, troubleshoot, and store SQL Server diagnostics information.

MySQL Enterprise Server 5.1

MySQL AB (2007) states, “Enterprise Server software is the most reliable, secure and up-to-date version of MySQL for cost-effectively delivering E-commerce, Online Transaction Processing (OLTP), and multi-terabyte Data Warehousing applications. It is a fully integrated transaction-safe, ACID compliant database with full commit, rollback, and crash recovery and row level locking capabilities.” According to Planetmysql (2006), MySQL delivers the ease of use, scalability, and performance that has made MySQL the world's most popular open source database.

The project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL is owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now a subsidiary of Sun Microsystems. As of 2009 Oracle Corporation began the process of acquiring Sun Microsystems (Charles, 2008).

Research by Robin and Arjen (2007) shows that MySQL is often used in free software projects that require a full-featured database management system, such as

WordPress, phpBB and other software built on the LAMP software stack. It is also used in very high-scale World Wide Web products including Wikipedia, Google and Facebook.

These are MySQL's features illustrated by MySQL AB (2009):

Scalability and Flexibility

The MySQL database server provides the ultimate in scalability, sporting the capacity to handle deeply embedded applications with a footprint of only 1MB to running massive data warehouses holding terabytes of information. Platform flexibility is a stalwart feature of MySQL with all flavors of Linux, UNIX, and Windows being supported. And, of course, the open source nature of MySQL allows complete customization for those wanting to add unique requirements to the database server.

High Performance

A unique storage-engine architecture allows database professionals to configure the MySQL database server specifically for particular applications, with the end result being amazing performance results. Whether the intended application is a high-speed transactional processing system or a high-volume web site that services a billion queries a day, MySQL can meet the most demanding

performance expectations of any system. With high-speed load utilities, distinctive memory caches, full text indexes, and other performance-enhancing mechanisms, MySQL offers all the right ammunition for today's critical business systems.

High Availability

Rock-solid reliability and constant availability are hallmarks of MySQL, with customers relying on MySQL to guarantee around-the-clock uptime. MySQL offers a variety of high-availability options from high-speed master/slave replication configurations, to specialized Cluster servers offering instant failover, to third party vendors offering unique high-availability solutions for the MySQL database server.

Robust Transactional Support

MySQL offers one of the most powerful transactional database engines on the market. Features include complete ACID (atomic, consistent, isolated, durable) transaction support, unlimited row-level locking, distributed transaction capability, and multi-version transaction support where readers never block writers and vice-versa. Full data integrity is also assured through server-enforced referential integrity, specialized transaction isolation levels, and instant deadlock detection.

Web and Data Warehouse Strengths

MySQL is the de-facto standard for high-traffic web sites because of its high-performance query engine, tremendously fast data inserts capability, and strong support for specialized web functions like fast full text searches. These same strengths also apply to data warehousing environments where MySQL scales up into the terabyte range for either single servers or scale-out architectures. Other features like main memory tables, B-tree and hash indexes, and compressed archive tables that reduce storage requirements by up to eighty-percent make MySQL a strong standout for both web and business intelligence applications.

Strong Data Protection

Because guarding the data assets of corporations is the number one job of database professionals, MySQL offers exceptional security features that ensure absolute data protection. In terms of database authentication, MySQL provides powerful mechanisms for ensuring only authorized users have entry to the database server, with the ability to block users down to the client machine level being possible. SSH and SSL support are also provided to ensure safe and secure connections. A granular object privilege framework is present so that users only see the data they should, and powerful data encryption and decryption functions

ensure that sensitive data is protected from unauthorized viewing. Finally, backup and recovery utilities provided through MySQL and third party software vendors allow for complete logical and physical backup as well as full and point-in-time recovery.

Comprehensive Application Development

One of the reasons MySQL is the world's most popular open source database is that it provides comprehensive support for every application development need. Within the database, support can be found for stored procedures, triggers, functions, views, cursors, ANSI-standard SQL, and more. For embedded applications, plug-in libraries are available to embed MySQL database support into nearly any application. MySQL also provides connectors and drivers (ODBC, JDBC, etc.) that allow all forms of applications to make use of MySQL as a preferred data management server. It doesn't matter if it's PHP, Perl, Java, Visual Basic, or .NET, MySQL offers application developers everything they need to be successful in building database-driven information systems.

Management Ease

MySQL offers exceptional quick-start capability with the average time from software download to installation completion being less than fifteen minutes. This

rule holds true whether the platform is Microsoft Windows, Linux, Macintosh, or UNIX. Once installed, self-management features like automatic space expansion, auto-restart, and dynamic configuration changes take much of the burden off already overworked database administrators. MySQL also provides a complete suite of graphical management and migration tools that allow a DBA to manage, troubleshoot, and control the operation of many MySQL servers from a single workstation. Many third party software vendor tools are also available for MySQL that handle tasks ranging from data design and ETL, to complete database administration, job management, and performance monitoring.

Open Source Freedom

Many corporations are hesitant to fully commit to open source software because they believe they can't get the type of support or professional service safety nets they currently rely on with proprietary software to ensure the overall success of their key applications. The questions of indemnification come up often as well. These worries can be put to rest with MySQL as complete around-the-clock support as well as indemnification is available through MySQL Enterprise. MySQL is not a typical open source project as all the software is owned and supported by MySQL AB, and because of this, a unique cost and support

model are available that provides a unique combination of open source freedom and trusted software with support.

Lowest Total Cost of Ownership

Accomplished through the use of the MySQL database server and scale-out architectures that utilize low-cost commodity hardware, corporations are finding that they can achieve amazing levels of scalability and performance, all at a cost that is far less than those offered by proprietary and scale-up software vendors.

Related Research

Infosys

A similar case was performed by Infosys Technologies (Infosys Technologies, 2008). Infosys Technologies Ltd. is an IT transformation leader for its clients around the world. With over 95,000 employees in more than 64 offices around the world, the company constantly searches for the best technology to help its customers meet their business goals. As Infosys began designing a solution to let automobile dealers share information on inventories and other resources, it wanted to offer a cloud-based solution that would be simple to deploy, easy to scale, and require zero maintenance from the customer. To create its hub-in-the-cloud solution it needed a cloud-based database, which it found with Microsoft SQL Data Services, part of the Microsoft

Services Platform. Infosys is using SQL Data Services to create a solution for auto dealers, which it can easily replicate to meet similar integration needs for healthcare, hospitality, insurance, and other industry verticals.

Infosys is using a three-tier architecture that includes (Infosys Technologies, 2008):

Web-based Interface: Dealer employees, wanting to check inventory at another dealership, or seeking OEM information, use a desktop browser to connect to the cloud-based Infosys solution.

Application Tier: Infosys provides the middle-tier code and business logic enabling integration of data between participating parties. This Infosys application tier provides connectivity between SDS in the cloud and the dealer's existing data systems. The Infosys Legacy Modernization solution, a part of the Catalytic IT solution suite is used with dealerships that have green-screen legacy systems. The Infosys application tier is hosted in the cloud.

Data Tier: Data from all parties—dealers, OEMs and other participants—is hosted on a dedicated instance of SDS in the cloud.

Benefits:

Infosys has found the platform it needs to create complete cloud-based solutions for its global customers. The company can create solutions that have low-cost

deployment and zero customer-side maintenance. The solutions can be easily replicated for other verticals, provide virtually unlimited scalability, and integrate easily with legacy systems.

Low cost Robust Infrastructure Ensures High Service Availability

Working with SDS and the rest of the Microsoft Services Platform has provided Infosys with the complete cloud infrastructure it needed to create its auto dealer solution. “We needed a robust cloud-based database solution to support hub-based integration,” Thethi notes. “That is exactly what SDS provides. The Services Platform also provides the infrastructure we need for deploying our application from the cloud, and for supporting authentication” (Thethi, 2009).

Infosys sees the platform as a great value proposition. “SQL Data Services and the rest of the Services Platform combine to form a cloud infrastructure that is very exciting,” says Thethi. “This opens the door for creating a new generation of cloud-based solutions”(Thethi, 2009).

Low-Cost Deployment and Zero-Maintenance

As Infosys searched for the best platform on which to build its cloud-based auto dealer integration solution, it sought a solution that would be inexpensive for its customers to deploy and maintain.

Easy Replication for Other Verticals

Auto dealerships are just one of a number of industries that Infosys sees benefiting from appliance-like cloud-based solutions using SDS as the data repository. Whether providing an integration point for hotels, clinics, or insurance agents, all of these verticals typically include geographically distributed operations that could benefit from easy to deploy, low-cost, and zero-maintenance solutions.

Thethi adds: “The need for a solution like this has always been there. With a cloud-based platform and leveraging Microsoft Services Platform we can give customers a scalable, zero-maintenance services and data hub. The cloud infrastructure is becoming real and the this appliance based approach for connecting the cloud with the enterprise makes it more relevant”

Newsgator

A research by Microsoft (2008) indicates, “NewsGator aggregates Really Simple Syndication (RSS) data feeds from across the Web to provide users with customized content delivery, enabling everyone to essentially create their own electronic newspaper. The company, which also provides Software as a Service to more than 50 media outlets including CNN and USA Today, stores some 2.5 billion RSS articles totaling about 4 terabytes on clustered databases running Microsoft SQL Server

database Software. NewsGator is upgrading its database infrastructure to SQL Server 2008 Enterprise Edition (64-bit) running on the Windows Server 2008 for 64-Bit Systems operating system to take advantage of a number of new features, including enhanced Database Mirroring for high availability, Backup Compression to reduce storage needs, and Resource Governor for allocating processing resources”.

Garanti Technology

Turkey’s Garanti Technology, which is owned by the Doğuş Group, a large Turkish conglomerate, was asked to create an enterprise resource planning (ERP) solution for Doğuş Automotive Group, an importer and distributor of passenger cars, light commercial vehicles, spare parts and accessories for Volkswagen, Audi, Porsche, and other manufacturers. Garanti Technology created an ERP solution linking Doğuş automotive dealers, part suppliers, and other key partners using the Microsoft Application Platform. The ERP solution, which includes a data warehouse, will soon be upgraded to Microsoft SQL Server 2008 database software to take advantage of new features that make it easier to manage data, better utilize resources, and enhance information security. The solution was developed using Microsoft Visual Studio 2008 (and earlier versions) and the Microsoft .NET Framework.

Benefits:

Developing and deploying its Turkuaz application using the Microsoft Application Platform provided Garanti Technology with a rapid development environment. Upgrading to SQL Server 2008 will provide additional benefits including easier database management, better resource utilization, and enhanced security.

CHAPTER III

Methodology

This chapter consists of methods and procedures used in conducting the present thesis. The following items are addressed: restatement of the problem, research instrument and procedure.

Restatement of Objectives

The purpose of study was to develop and analyze a filter kit system database for CLARK Material Handling Company. This study started with a research of Database Management System (DBMS) and an introduction of several commonly used DBMS. Because the server of CLARK Material Handling Company uses WINDOWS SERVER 2003 as operation system, there are only two DBMS (SQL Server 2008 and MySQL version 5.1) that can be chosen. This study used SQL Server 2008 to generate the filter kit system database and compares with the database made with MySQL version 5.1. Upon completion of the study, following objectives were achieved:

Objective 1

Generate a filter kit system database for CLARK Material Handling Company and demonstrate several major design procedures.

Objective 2

Evaluate and analyze the filter kit system database. Demonstrate capability, advantages and disadvantages of this database.

Research Instrument and Procedure

This part will demonstrate procedures about how to use SQL Server 2008 and Visual Studio 2008 to build a database and then design WebPages to connect with the database.

These are some main procedures for developing the filter kit system database:

- Download and install the SQL Server 2008
- Setup the administrator account and password
- Build a new database
- Get Excel files which contain all the filter kit's information including brand, model, engine, fuel, serial number, etc from the customer service department of CLARK Material Handling Company (see Appendix D).
- Import all the data from Excel files to the new database. Figure3.4 shows an example of new tables in filter kit system database.
- Integrate the new database to the main server database.

- Use Visual Studio 2008 to make an index page for both administrator and customers (Source codes see Appendix A and Appendix B). Figure3.1 is the data flow chart of the filter kit system database. It shows how users and administrator access through different index pages. Figure3.2 is the interface of user index page. There is a brief description in this page about how to use this database. Figure3.2 is the interface of administrator index page. Figure3.5 shows the edit window of Visual Studio 2008.
- Make relevant WebPages for brands including Clark, Doosan, Hyster, Mitsubishi, Nissan, and Toyota (Source codes see Appendix C).
- Connect all the WebPages with Database.
- Create a user manual (See Appendix E).

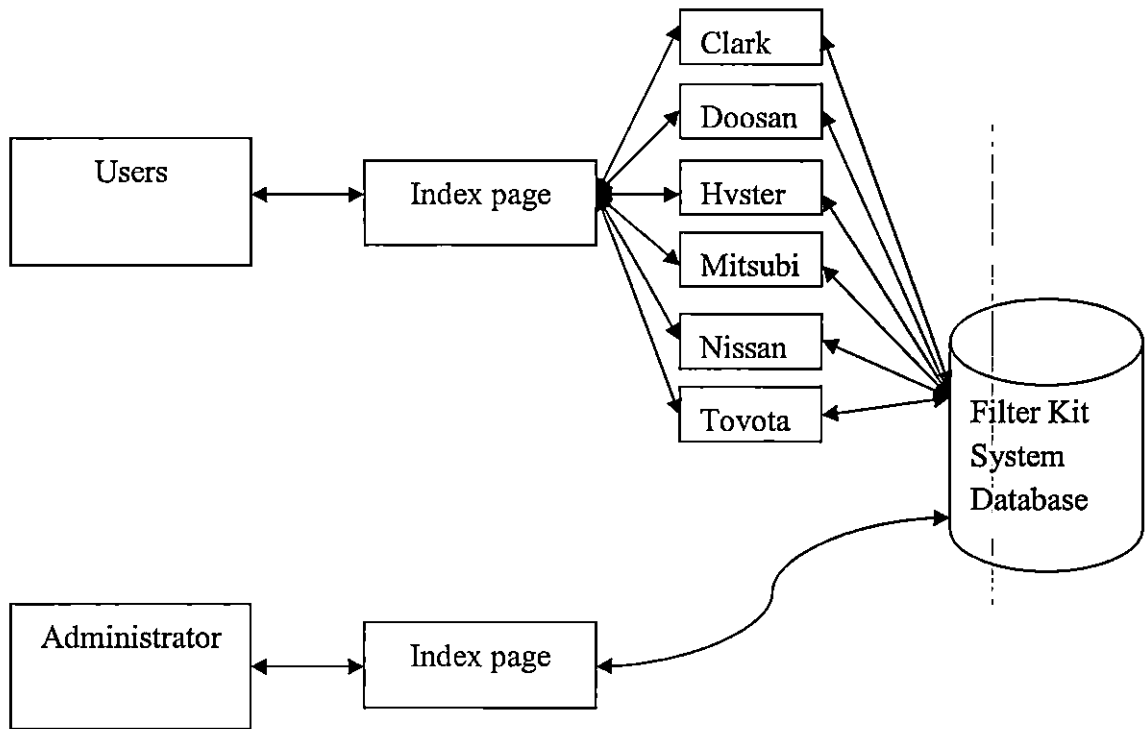


Figure3.1: Data flow chart for Filter Kit System Database

CLARK
THE FORKLIFT

CLARK Online Intranet Services

Selecting Your Filter Kit B. O. M

User Manual

Please make selections from items where "Please Select" displays. Information in lower selection boxes depend on their upper selection boxes. When only one item or "all" appears in the drop down menu, click on it and move to the next item. Once all items have been selected the appropriate kit, bill of material will appear. Please note that kit BOM reference numbers are provided for your convenience only. Complete kits are not presently stocked and sold by Clark.

Brand :

Model :

Engine :

Fuel :

Figure3.2: User Index page for Filter Kit System Database

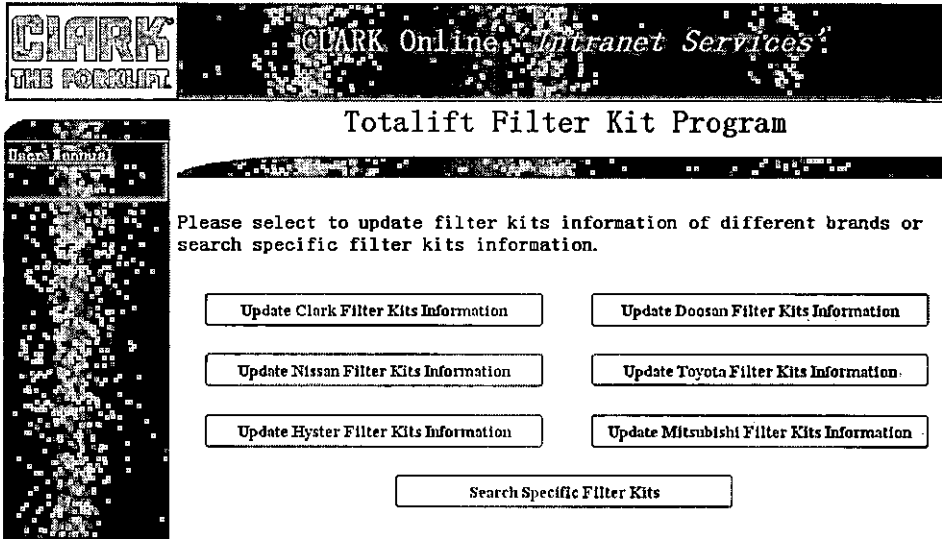


Figure3.3: Administrator Index page for Filter Kit System Database

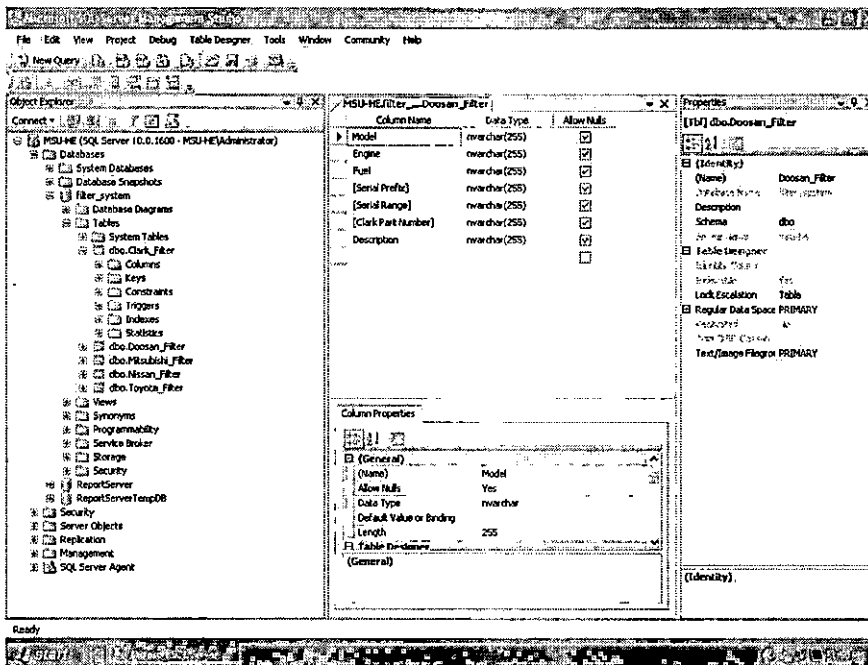


Figure3.4: The table of Clark in filter kit system database

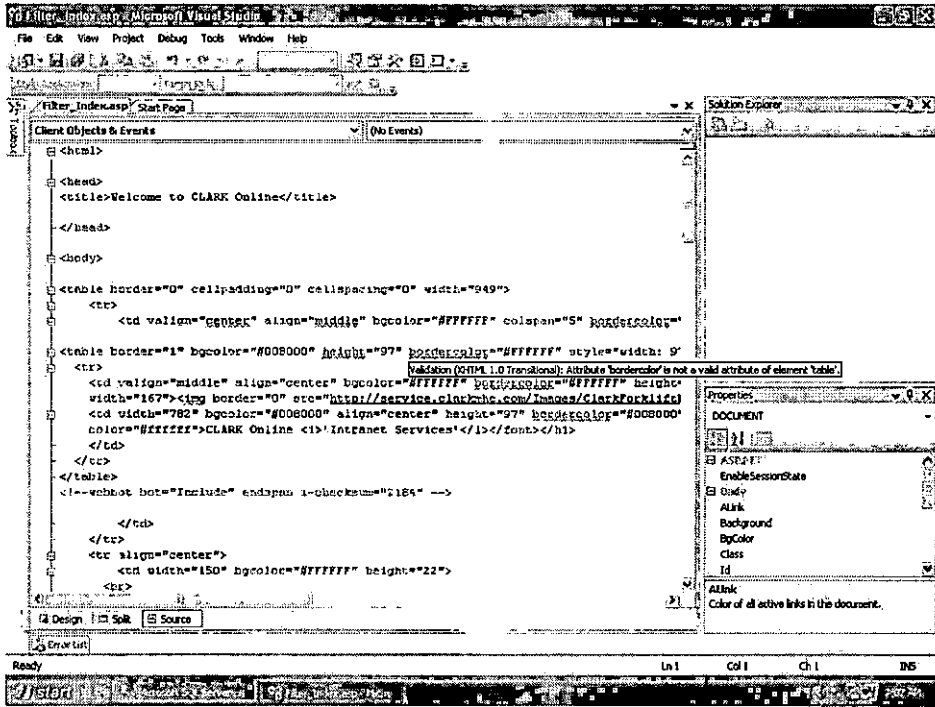


Figure 3.5: Edit window for Administrator Index page

CHAPTER IV

Case Study

Comparison between Filter kit system database made by Microsoft SQL Server and My SQL

The platform support, hardware requirements, software requirements, price, limits, performance, and security of the filter kit system database will depend on which kind of DBMS is used to build this filter kit system database. To compare the filter kit system database made by Microsoft SQL Server and My SQL, emphasis will be placed on a comparison between these two kinds of DBMS.

Platform support comparison

Microsoft SQL Server 2008 only works on Windows-based platforms, including: Windows XP, Windows Server 2003, Windows Vista, Windows Server 2008, Windows 7, and Windows Server 2008 R2.

In comparison with Microsoft SQL Server 2008, MySQL works on many different system platforms, including AIX, BSDi, FreeBSD, HP-UX, i5/OS, Linux, Mac OS X, NetBSD, Novell NetWare, OpenBSD, OpenSolaris, eComStation, OS/2 Warp, QNX, IRIX, Solaris, Symbian, SunOS, SCO OpenServer, SCO UnixWare, Sanos, Tru64 and Microsoft Windows. A port of MySQL to OpenVMS also exists.

Hardware requirements

Table 4.1: Minimum hardware requirements to install and run SQL Server 2008

Hardware	Requirements
Processor	Pentium III 1.0 GHz or higher
Memory	512 MB RAM (minimum for Desktop Engine), 2.0 GB RAM or more recommended
Hard disk space	280 MB (Database Engine and data files, Replication, and Full-Text Search) 90 MB (Analyze Services and data files), 120 MB (Reporting Services and Report Manager), 120 MB (Integration Services), Client Components: 850 MB SQL Server Books Online and SQL Server Compact Books Online: 240 MB
Display	SQL Server 2008 graphical tools require VGA or higher resolution: at least 1,024x768 pixel resolution.
Drive	A CD or DVD drive, as appropriate, is required for installation from disc.
Other Devices	Pointing device: A Microsoft mouse or compatible pointing device is required.

MySQL version 5.1 is not as powerful as SQL Server 2008 and uses less hardware resources. To install MySQL version 5.1, one should have near 1.0 G RAM and near 600 Mb hard disk space.

Software requirements

Table 4.2: Minimum software requirements to install and run SQL Server 2008:

Component	Requirement
Framework	<p>SQL Server Setup installs the following software components required by the product:</p> <ul style="list-style-type: none"> • .NET Framework 3.5 SP1 • SQL Server Native Client • SQL Server Setup support files
Software	<p>SQL Server Setup requires Microsoft Windows Installer 4.5 or a later version</p> <p>After installing required components, SQL Server Setup will verify that the computer where SQL Server 2008 will be installed also meets all the other requirements for a successful installation.</p>
Network Software	<p>Network software requirements for the 64-bit versions of SQL Server 2008 are the same as the requirements for the 32-bit versions.</p> <p>Supported operating systems have built-in network software. Stand-alone named and default instances support the following network protocols:</p> <ul style="list-style-type: none"> • Shared memory • Named Pipes • TCP/IP • VIA
Internet Software	<p>Microsoft Internet Explorer 6 SP1 or a later version is required for all installations of SQL Server 2008. Internet Explorer 6 SP1 or a</p>

	later version is required for Microsoft Management Console (MMC), SQL Server Management Studio, Business Intelligence Development Studio, the Report Designer component of Reporting Services, and HTML Help.
--	---

Table 4.3: MySQL version 5.1 requires the following software:

Platform	Operating System Version
Windows-based	Windows 95/98/NT/2008/XP/2003
Sun Solaris	Solaris 8 (SPARC)
FreeBSD	FreeBSD 4.x (x86)
Mac OS X	Mac OS X v10.2
HP-UX	HP-UX 10.20 (RISC 1.0), HP-UX 11.11 (PA-RISC 1.1 and 2.0), HP-UX 11.11 (PA-RISC 2.0, 64-bit only)
AIX-Based	AIX 5.1 (RS6000), AIX 4.3.2 (RS6000), AIX 4.3.3 (RS6000)
QNX	QNX 6.2.1 (x86)
SGI Irix	SGI Irix 6.5
Dec OSF	Dec OSF 5.1 (Alpha)

Price comparison

Microsoft SQL Server 2008 licensing provides the option to purchase SQL Server 2008 under a server operating system license with incremental Client Access Licenses (CALs), or a per-processor license model. The table below provides guidance on estimated retail pricing for U.S. and Canadian editions of SQL Server 2008:

Table 4.4: Price for U.S. and Canadian editions of SQL Server 2008

Editions	Processor Pricing	Server Plus CAL Pricing
Enterprise Edition A comprehensive data management and business intelligence platform providing enterprise class scalability, high availability, and security for running business-critical applications	Retail \$24,999 Example* \$23,911	Retail \$13,969 with 25 CALs Example* \$8,487 \$162 per additional CAL
Standard Edition A complete data management and business intelligence platform providing best-in-class ease of use and manageability for running departmental applications	Retail \$5,999 Example* \$5,737	Retail \$1,849 with 5 CALs Example* \$885 server, \$162 per additional CAL
Workgroup Edition A reliable data management and reporting platform providing secure, remote synchronization, and management capabilities for running branch applications	Retail \$3,899 Example* \$3,700	Retail \$739 with 5 CALs Example* \$730 per server \$146 per additional CAL

*This is representative pricing for a company purchasing a small number of Processor or Server licenses through Microsoft Volume Licensing.

The MySQL version 5.1 is Dual Licensed. Users can choose to use the MySQL software as an Open Source/Free Software product under the terms of the GNU General Public License or can purchase a standard commercial license from MySQL AB.

The MySQL v5.1 Server commercial license is per database server (single installed MySQL binary). The difference between Basic MySQL version and other versions is that Basic has less abilities and can't get any problem resolution support.

The price comparisons below were based on the MySQL Licensing Prices from MySQL AB.

Table 4.5: MySQL Licensing Price

MySQL Enterprise	Basic	Silver	Gold	Platinum
	USD 599 /Server/Year	USD 1999 /Server/Year	USD 2999 /Server/Year	USD 4999 /Server/Year

SQL Server 2008 and MySQL v5.1 limits

Table 4.6: SQL Server 2008 and MySQL version 5.1 limits:

Feature	SQL Server 2008	MySQL v5.1
column name length	128	64
index name length	128	64
table name length	128	64
max indexes per table	250	32
index length	900	1024
max index column length	900	255
columns per index	16	16
max char() size	8000	1048543
max varchar() size	8000	1048543
max blob size	2147483647	1048543

max number of columns in GROUP BY	Limited only by number of bytes (8060)	64	
max number of columns in ORDER BY	Limited only by number of bytes (8060)	64	
tables per SELECT statement	256	31	
max columns per table	1024	2599	
max table row length	8036	65534	
longest SQL statement	16777216	1048574	
constant string size in SELECT	16777207	1048565	

Security comparison

Security is becoming increasingly important as more networks are connected together. Organizations' assets must be protected, particularly its databases, which contain company's valuable information. Security is one of the critical features of a database engine.

SQL Server 2008 provides rich security features to protect data and network resources. It is much easier to install securely, since all but the most essential features are either not installed by default or disabled if they are installed. SQL Server provides plenty of tools to configure the server, particularly for SQL Server Surface Area Configuration. Its authentication features are stronger because SQL Server more closely integrates with Windows authentication and protects against weak or ancient passwords. Granting and controlling what a user can do when authenticated is far more flexible with granular permissions, SQL Server Agent proxies, and execution

context. Even metadata is more secure, since the system metadata views return information only about objects that the user has permission to use in some way. At the database level, encryption provides a final layer of defense while the separation of users and schemas makes managing users easier.

These guiding principles are evident throughout SQL Server 2008, which provides all the tools to secure databases:

Secure by design: Software requires a secure design as a foundation for repelling attackers and protecting data.

Secure by default: System administrators should not have to work to make a fresh installation secure; it should be that way by default.

Secure in deployment: Software should help to keep itself updated with the latest security patches and assist in maintenance.

Communications: Communicate best practices and evolving threat information so that administrators can proactively protect their systems.

MySQL version 5.1 also provides several kinds of commands and tools to make sure the safety of database. But most of the security introduction part tells us how to minimize the number of attack chances of the systems by correctly setting parameters.

CHAPTER V

Findings

After the comparison with SQL Server 2008 and MySQL version 5.1, it can be found that the filter kit system database made with SQL Server 2008 is better than MySQL version 5.1. Both of these two DBMS can be used to build stable and efficient system. The stability and effectiveness of applications and databases depend rather on the experience of the database developers and database administrator than on the database's provider. The filter kit system database made by SQL Server 2008 has some advantages in comparison with MySQL version 5.1:

- SQL Server 2008 is generally accepted as easier to install, use and manage.
- The filter kit system database based on SQL Server 2008 includes compression, which reduces disk space from 30% to 70%, thus reducing storage and backup costs – MySQL does not have this capability.
- The filter kit system database based on SQL Server 2008 can best support Windows-based platforms which are used in CLARK Material Handling Company's server.
- SQL Server 2008 provides capabilities for solutions that require the highest availability and security, and need capabilities like mirroring to simplify failover,

transparent data encryption to secure all data with no application changes, external hardware key management for advanced security, and online CPU changes with no downtime. MySQL has none of these capabilities.

- The filter kit system database based on SQL Server 2008 has the best compatibility with Microsoft Office which is widely used in organizations and companies. Design and deliver reports of existing data, mine millions of rows to find trends, and move data from applications like SAP or excel sheets to a data warehouse with no development – all this is integrated out of the box with SQL Server. MySQL does not have any Business Intelligence capabilities

Conclusion

This thesis has two goals in effort. The first goal is that the solution be easy to install and configure. The second goal is that the solution be user-friendly. It is tightly integrated with Microsoft products, which means that employees can access its information in the Microsoft tools they use every day.

Rely on Windows Server 2003, SQL Server 2008, and Visual Studio 2008 provide a secure and trusted foundation, the filter kit system database offers advanced technology support for the latest platforms, improved management and Web tools, and access to relevant information throughout the intranet.

Future Work

According to Amu (2010), “Databases are centralized collections of data for use by business applications. They typically store critical and sensitive information including customer account data. Databases can exist on mainframes, networks, and standalone PCs. Because they can be repositories of the financial institution’s most critical information, databases pose unique risks. Failure to adequately manage and secure databases can lead to unintentional or unauthorized modification, destruction, or disclosure of sensitive information. Unauthorized disclosure of confidential information can result in reputation, legal, and operational risk to the institution and possible financial loss.” Database security is the last line of defense. So this study can be further extended to greater focus on the protection of private data from both internal and external attacks than IT pros have traditionally given it.

References

- Anonymous.(2008). *Infosys Technologies*. From:
http://www.microsoft.com/casestudies/Case_Study_Detail.aspx?CaseStudyID=4000002880
- Bteksoftware. (2009). *SQL2008_ProductOverview* From:
www.bteksoftware.com/docs/SQL2008_ProductOverview.docx
- Benjamin Amu. (2010). *Risks and security issues in database management*. From:
<http://www.technologytimesng.com/2010/03/15/risks-and-security-issues-in-database-management/>
- Charles, Babcock. (2008). *Sun Locks Up MySQL, Looks To Future Web Development*. InformationWeek. From:
<http://www.informationweek.com/news/showArticle.jhtml?articleID=206900327>.
- Codd, E.F.(2002).*A Relational Model of Data for Large Shared Data Banks*. Communications of the ACM 13 (6): 377–387.
- Fernando, Azpeitia Lopez. (2008). *SQL Server 2008 Full-Text Search: Internals and Enhancements*. From <http://msdn.microsoft.com/en-us/library/cc721269.aspx>.
- Informationweek. (2007). *Microsoft Gives Peek At Next Version Of SQL Server*. From
<http://www.informationweek.com/software/showArticle.jhtml?articleID=199500164&subSection=Development>.
- Jeff Jones. (2003). *The Big Picture: IBM DB2 Information Management Software and DB2 Universal Database*, From:
<http://www.ibm.com/developerworks/data/library/techarticle/0301jones/0301jones.html>
- Kroenke, D. (1995). *Database Processing: Fundamentals, Design, and Implementation*. Prentice Hall.
- Microsoft. (2009). *Database Engine Enhancements*. From

<http://msdn.microsoft.com/en-us/library/ms170910.aspx>.

Microsoft. (2008). *RSS Aggregator NewsGator Manages 2.5 Billion Articles with SQL Server 2008*, From:

http://www.microsoft.com/casestudies/Case_Study_Detail.aspx?CaseStudyID=4000002409

MySQL AB. (2007). *What is MySQL?, MySQL 5.0 Reference Manual*. From:
<http://dev.mysql.com/doc/refman/5.0/en/what-is-mysql.html>.

MySQL AB. (2009). *MySQL Enterprise Features*. From:
<http://www.mysql.com/products/enterprise>

Oracle. (2009). *Introduction to the Oracle Database*, From:

http://download.oracle.com/docs/cd/B13789_01/server.101/b10743/intro.htm

Planetmysql.(2006). *MySQL Max Build Policy*. From:

<http://www.planetmysql.org/kaj/?p=58>.

Robin Schumacher, Arjen Lentz. (2007). *Dispelling the Myths*. From:

<http://dev.mysql.com/tech-resources/articles/dispelling-the-myths.html>.

Sam Lightstone, Toby J. Teorey, Tom Nadeau. (2006). *Physical database design*, Elsevier.

Thethi. (2009). *Infosys Creates Cloud-Based Solution for Auto Dealers using SQL Data Services*. From:

<http://www.infosys.com/research/publications/Documents/cloud-computing.pdf>

APPENDIX A

Source code of the Administrator Index Website


```
<html>

<head>
<title>Welcome to CLARK Online</title>

</head>

<body>

<table border="0" cellpadding="0" cellspacing="0" width="949">
  <tr>
    <td valign="center" align="middle" bgcolor="#FFFFFF" colspan="5" bordercolor="#FFFFFF"
width="949"><!--webbot bot="Include" U-Include="include/header.asp" TAG="BODY" startspan -->

<table border="1" bgcolor="#008000" height="97" bordercolor="#FFFFFF" style="width: 977px">
  <tr>
    <td valign="middle" align="center" bgcolor="#FFFFFF" bordercolor="#FFFFFF" height="97"
width="167"></td>
    <td width="782" bgcolor="#008000" align="center" height="97" bordercolor="#008000"><h1><font
color="#ffffff">CLARK Online <i>'Intranet Services'</i></font></h1>
    </td>
  </tr>
</table>
<!--webbot bot="Include" endspan i-checksum="2184" -->

    </td>
  </tr>
  <tr align="center">
    <td width="150" bgcolor="#FFFFFF" height="22">
    <br>
    
    </td>
    <td width="15" height="22">
    </td>
```

```

        <td colspan="3" height="22" style="width: 797px">
<strong><span style="font-size: 24pt">Totalift Filter Kit Filter kit system database</span></strong></td>
        </tr>
<tr>
        <td width="170" bgcolor="#008000" valign="top" align="left" bordercolor="#008000"
style="height: 414px">

<table border="1" width="170" bgcolor="#008000">
<tr>
        <td width="160" bgcolor="#008000" valign="top" align="left" bordercolor="#008000">

<font color="#FFFFFF"><a href="Totalift Filter Kits User Manual.pdf" style="color: #FFFFFF"
target="_blank"><strong>
        User Mannual</strong></a><br />
<br />
</font>
<font color="#FFFFFF"><strong>
        <br />
</strong></font>
                </td>
        </tr>
</table>
        <p>&nbsp;</p>
</td>
        <td width="15" style="height: 414px">&nbsp;<br />
</td>
        <td colspan="3" style="width: 797px; height: 414px;" valign="top"><strong><span
style="font-size: 24pt"></span></strong>
        <p>
                <b><font size="+3">
                        </font></b>&nbsp;</p><form name="Form1" style="font-weight: bold; font-size: 20px">
                Please select to update filter kits information of different brands or
search specific
                filter kits information.<br />

```

```

<br />

<table style="width:
800px">
  <tr>
    <td style="width: 400px; height: 60px" align="center">
      <button onclick="window.open('Clark Filters.xls')" style="font-weight: bold; height:
35px; font-size: 13pt; width: 350px; font-family: 'Times New Roman';">
        Update Clark Filter Kits Information</button></td>
    <td colspan="2" style="width: 400px; height: 60px" align="center">
      <button onclick="window.open('Doosan Filters.xls')" style="font-weight: bold;
width: 350px; height: 35px; font-size: 13pt; font-family: 'Times New Roman';">
        Update Doosan Filter Kits Information</button></td>
    </tr>
    <tr>
      <td style="width: 400px; height: 60px;" align="center">
        <button onclick="window.open('Nissan Filters.xls')" style="width: 350px;
font-weight: bold; height: 35px; font-size: 13pt; font-family: 'Times New Roman';">
          Update Nissan Filter Kits Information</button></td>
      <td colspan="2" style="width: 400px; height: 60px;" align="center">
        <button onclick="window.open('Toyota Filters.xls')" style="width: 350px;
font-weight: bold; height: 34px; font-size: 13pt; font-family: 'Times New Roman';">
          Update Toyota Filter Kits Information</button></td>
    </tr>
    <tr>
      <td align="center" style="width: 400px; height: 60px">
        <button onclick="window.open('Hyster Filters.xls')" style="width: 350px;
font-weight: bold; height: 35px; font-size: 13pt; font-family: 'Times New Roman';">
          Update Hyster Filter Kits Information</button></td>
      <td align="center" colspan="2" style="width: 400px; height: 60px"><button
onclick="window.open('Mitsubishi Filters.xls')" style="width: 350px; font-weight: bold; height: 35px;
font-size: 13pt; font-family: 'Times New Roman';">
        Update Mitsubishi Filter Kits Information</button></td>
    </tr>
  </tr>
  <tr>
    <td align="center"

```

```

colspan="3" style="height: 60px">
    <button onclick="window.location='FilterInfo.asp'" style="font-weight: bold; width:
350px; height: 35px; font-size: 13pt; font-family: 'Times New Roman';">
        Search Specific Filter Kits</button></td>
</tr>
</table>
<!--webbot bot="Include" endspan i-checksum="61767" -->

    <p></td>
</tr>
</table>
<!--webbot bot="Include" U-Include="include/footer.asp" TAG="BODY" startspan -->

<table border="0" width="949" cellspacing="0" cellpadding="0">
<tr>
    <td width="949"><p align="center"><font color="#0000ff"
        face="Verdana,Verdana,Geneva,Arial,Helvetica,sans-serif" size="1"><i>Last Updated: <script
        language="JavaScript">document.write(document.lastModified);
</script> 01/02/2001 15:03:08</i></font><br>
        <font color="#000000" face="Verdana,Verdana,Geneva,Arial,Helvetica,sans-serif"
        size="1">Designed,
            Filter kit system databasemed &amp; Maintained by the Clark Web Filter kit system databaseming
            Team<br>
            For technical problems, send email to the <a
            href="mailto:webmaster@clarkmhc.com">Webmaster</a><br>
            ?1998, 1999, 2000, 2001 CLARK Material Handling Company - All Rights Reserved</font></td>
</tr>
</table>
<!--webbot bot="Include" endspan i-checksum="1610" -->
</body>
</html>

```

APPENDIX B

Source code of the User Index Website

```
<html>

<head>
<title>Welcome to CLARK Online</title>
</head>

<body>

<table border="0" cellpadding="0" cellspacing="0" width="949">
  <tr>
    <td valign="center" align="middle" bgcolor="#FFFFFF" colspan="5" bordercolor="#FFFFFF"
width="949"><!--webbot bot="Include" U-Include="include/header.asp" TAG="BODY" startspan -->

<table border="1" bgcolor="#008000" height="97" bordercolor="#FFFFFF" style="width: 949px">
  <tr>
    <td valign="middle" align="center" bgcolor="#FFFFFF" bordercolor="#FFFFFF" height="97"
width="167"></td>
    <td width="782" bgcolor="#008000" align="center" height="97" bordercolor="#008000"><h1><font
color="#ffffff">CLARK Online <i>'Intranet Services'</i></font></h1>
    </td>
  </tr>
</table>
<!--webbot bot="Include" endspan i-checksum="2184" -->

    </td>
  </tr>
  <tr align="center">
    <td width="150" bgcolor="#FFFFFF" height="22">
    <br>
    
    </td>
    <td width="15" height="22">
    </td>
    <td colspan="3" height="22" style="width: 797px">
```

```

<strong><span style="font-size: 24pt">Selecting Your Filter Kit B.O.M</span></strong></td>
  </tr>
<tr>
  <td width="170" bgcolor="#008000" valign="top" align="left" bordercolor="#008000" ;
style="height: 414px">
  <!--webbot bot="Include" U-Include="include/left.asp" TAG="BODY" startspan -->

<table border="1" width="170" bgcolor="#008000">
  <tr>
    <td width="160" bgcolor="#008000" valign="top" align="left" bordercolor="#008000">

<font color="#FFFFFF"><a href="Totalift Filter Kits User Manual.pdf" style="color: #FFFFFF"
target="_blank"><strong>
  User Mannual</strong></a><br />
  <br />
</font>
<font color="#FFFFFF"><strong>
  <br />
</strong></font>
      </td>
    </tr>
  </table>
  <!--webbot bot="Include" endspan i-checksum="43390" -->
  <p>&nbsp;</p>
</td>
  <td width="15" style="height: 414px">&nbsp;<br />
  </td>
  <td colspan="3" style="width: 797px; height: 414px;" valign="top"><strong><span
style="font-size: 24pt"></span></strong>
  <p>
    <b><font size="+3">
      </font></b>&nbsp;</p>
  <form name="Form1" style="font-weight: bold; font-size: 20px">
    Please make selections from items where "Please Select" displays.

```

Information in

one item

item. Once

appear.

lower selection boxes depend on their upper selection boxes. When only

or "all" appears in the drop down menu, click on it and move to the next

all items have been selected the appropriate kit, bill of material will

convenience only.

Please note that kit BOM reference numbers are provided for your

Complete kits are

not presently stocked and sold by Clark.

<table border="0">

<tr align="left">

<td style="width: 93px; height: 35px">Brand : </td>

<td style="height: 35px"><select

onchange="window.open(this.options[this.selectedIndex].value,'_top')"

style="width: 280px">

<option selected="selected" value="">Please Select</option>

<option value="clark_filter.asp">Clark</option>

<option value="doosan_filter.asp">Doosan</option>

<option value="hyster_filter.asp">Hyster</option>

<option value="Mitsubishi_filter.asp">Mitsubishi</option>

<option value="nissan_filter.asp">Nissan</option>

<option value="toyota_filter.asp">Toyota</option>

</select>

</td></tr>

<tr align="left">

<td style="width: 93px; height: 35px">Model : </td>

<td style="height: 35px"><select style="width: 280px" id="Select1" onclick="return

Select1_onclick()">

<option selected="selected">Please Select Brand First</option>

</select>

</td></tr>


```

        <tr align="left">
            <td style="width: 93px; height: 35px"><b>Engine : </b></td>
            <td style="height: 35px"><select style="width: 280px">
<option selected="selected">Please Select Brand First</option>
            </select>
            </td></tr>

        <tr align="left">
            <td style="width: 93px; height: 35px"><b>Fuel : </b></td>
            <td style="height: 35px"><select style="width: 280px">
<option selected="selected">Please Select Brand First</option>
            </select>
            </td></tr>
    </table>
<!--webbot bot="Include" endspan i-checksum="61767" -->

    <p></td>
</tr>
</table>
<!--webbot bot="Include" U-include="include/footer.asp" TAG="BODY" startspan -->

<table border="0" width="949" cellspacing="0" cellpadding="0">
    <tr>
        <td width="949"><p align="center"><font color="#0000ff"
        face="Verdana,Verdana,Geneva,Arial,Helvetica,sans-serif" size="1"><i>Last Updated: <script
        language="JavaScript">document.write(document.lastModified);
</script> 01/02/2001 15:03:08</i></font><br>
        <font color="#000000" face="Verdana,Verdana,Geneva,Arial,Helvetica,sans-serif"
        size="1">Designed,
        Programmed & Maintained by the Clark Web Programming Team<br>
        For technical problems, send email to the <a
        href="mailto:webmaster@clarkmhc.com">Webmaster</a><br>
        ?1998, 1999, 2000, 2001 CLARK Material Handling Company - All Rights Reserved</font></td>
    </tr>
</table>

```

```
<!--webbot bot="Include" endspan i-checksum="1610" -->  
  </body>  
</html>
```

APPENDIX C

Source code of the Product Websites

```

<html>

<head>
<title>Welcome to CLARK Online</title>
    <%
    dim strSQL
    set con = Server.CreateObject("ADODB.Connection")
    DSN="Driver={SQL
Server};server=sql.clarkmhc.com;uid=usedtruck;pwd=clark;database=Filters"
    con.Open(DSN)
    if request.QueryString("modelname")="" then
    que = "drop table Clark_Filter;" & _
"SELECT * INTO Clark_Filter FROM
OPENDATASOURCE('Microsoft.Jet.OLEDB.4.0','Data Source=C:\inetpub\wwwroot\Clark
Filters.xls;Extended Properties=Excel 8.0')...[Clark_Filter_Kits_Priced$];" & _
"delete from Clark_Filter where Brand IS NULL;"
    con.Execute(que)
    end if
    %>
</head>

<body>

<table border="0" cellpadding="0" cellspacing="0" width="949">
    <tr>
        <td valign="center" align="middle" bgcolor="#FFFFFF" colspan="5" bordercolor="#FFFFFF"
width="949"><!--webbot bot="Include" U-Include="include/header.asp" TAG="BODY" startspan -->

<table border="1" bgcolor="#008000" height="97" bordercolor="#FFFFFF" style="width: 1023px">
    <tr>
        <td valign="middle" align="center" bgcolor="#FFFFFF" bordercolor="#FFFFFF" height="97"
width="167"></td>
        <td width="782" bgcolor="#008000" align="center" height="97" bordercolor="#008000"><h1><font
color="#ffffff">CLARK Online <i>'Intranet Services'</i></font></h1>
        </td>

```

```

</tr>
</table>
<!--webbot bot="Include" endspan i-checksum="2184" -->

</td>
</tr>
<tr align="center">
<td width="150" bgcolor="#FFFFFF" height="22">
<br>

</td>
<td width="15" height="22">
</td>
<td colspan="3" height="22" style="width: 797px">
<strong><span style="font-size: 24pt">Selecting Clark Filter Kit B.O.M</span></strong></td>
</tr>
<tr>
<td width="170" bgcolor="#008000" valign="top" align="left" bordercolor="#008000"
style="height: 414px">
<!--webbot bot="Include" U-Include="include/left.asp" TAG="BODY" startspan -->

<table border="1" width="170" bgcolor="#008000">
<tr>
<td width="160" bgcolor="#008000" valign="top" align="left" bordercolor="#008000">

<font color="#FFFFFF"><a href="Filter_Index.asp" style="color: #FFFFFF"><strong>
Totalift Kits Filter kit system database</strong></a><br />
<br />
</font>
<font color="#FFFFFF"><a href="Totalift Filter Kits User Manual.pdf" style="color: #FFFFFF"
target="_blank"><strong>
User Mannual</strong></a><br />
<br />
</font>
<font color="#FFFFFF"><strong>

```

```

<br />
</strong></font>
      </td>
    </tr>
  </table>
  <!--webbot bot="Include" endspan i-checksum="43390" -->
  <p>&nbsp;</p>
</td>
<td width="15" style="height: 414px">&nbsp;</td>
<td colspan="3" style="width: 797px; height: 414px; font-weight: bold; font-size: 15pt;"
valign="top"><strong><span style="font-size: 24pt"></span></strong>
  <p>
    <b><font size="+3">
      </font></b>&nbsp;</p>

```

Please make selections from items where "Please Select" displays. Information in lower selection boxes depend on their upper selection boxes. When only one item or 换 appears in the drop down menu, click on it and move to the next item. Once all items have been selected the appropriate kit, bill of material will appear.

Please note that kit BOM reference numbers are provided for your convenience only.

Complete kits are not presently stocked and sold by Clark.

<!--webbot bot="Include" endspan i-checksum="61767" -->

```

<table style="width: 850px; height: 57px" border="0">
  <tr>
    <td style="width: 111px; height: 35px">
      <strong>Brand : &nbsp; &nbsp; &nbsp; </strong></td>
    <td colspan="3" style="height: 35px">
      <select onchange="window.open(this.options[this.selectedIndex].value,'_top')"
        style="width: 310px">
        <option value="">Please Select</option>
        <option value="clark_filter.asp" selected>Clark</option>
        <option value="doosan_filter.asp">Doosan</option>
        <option value="Hyster_filter.asp">Hyster</option>
        <option value="Mitsubishi_filter.asp">Mitsubishi</option>
        <option value="nissan_filter.asp">Nissan</option>
        <option value="toyota_filter.asp">Toyota</option>
      </select>
    </td>
    <td colspan="2" rowspan="5">
      <form id="result">
        <table style="width:400">
          <%
            if request.QueryString("filtername")<>"" then
              resultstring="select * from Clark_Filter where Model="" +
request.QueryString("modelname")+"" and Engine="" + request.QueryString("enginename")+"" and
Fuel="" + request.QueryString("fuelname")+"" and [Serial Number]=""+
request.QueryString("serialname")+"" and Filter="" + request.QueryString("filtername")+""
              set result = con.Execute(resultstring)
              response.write("<tr style=" & "background-color:Green" & " align=center" & "><td
style=width:250px><b style=" & "color:White" & ">Description</b></td><td style=width:150px><b style="
& "color:White" & ">Clart Part Number</b></td></tr>")
              count = 0
              do while not result.EOF
                des = result("Description")
                num = result("Clark Part Number")
                count=count+1
                i = count mod 2
                if i = 1 then

```

```

response.write("<tr style=" & "background-color:#E3EAEB;" & " align=center" & "><td>" &
des & "</td><td>" & num & "</td></tr>")
else
response.write("<tr style=" & "background-color:White;" & " align=center" & "><td>" & des
& "</td><td>" & num & "</td></tr>")
end if
result.moveToNext()
loop
end if
%>

</table>
</form>
</td>
</tr>
<tr>
<td style="width: 111px; height: 35px;">
<strong>Model :</strong></td>
<td colspan="3" style="height: 35px">
<select style="width: 310px"
onchange="window.open(this.options[this.selectedIndex].value,'_top')">
<%
set result1 = con.Execute("select distinct Model from Clark_Filter order by Model")
model_name=request.QueryString("modelName")
if model_name="" then
response.write("<option selected>Please Select</option>")
end if
do while not result1.EOF
results1 = result1("Model")
if results1 <> model_name then
response.write("<option value='clark_filter.asp?modelName=" & results1 & "'>"&
results1 & "</option>")
else
response.write("<option selected>" & model_name & "</option>")
end if
result1.moveToNext()

```



```

loop
%>
        </select>
    </td>
</tr>
<tr>
    <td style="height: 35px; width: 111px;">
        <strong>Engine :</strong></td>
    <td style="width: 135px; height: 35px;">
        <select style="width: 148px;"
onchange="window.open(this.options[this.selectedIndex].value,'_top')">
    <%
        enginestr="select distinct Engine from Clark_Filter where Model=" +
request.QueryString("modelName")+""
        set result2 = con.Execute(enginestr)
        if request.QueryString("enginename")="" then
            response.write("<option selected>Please Select</option>")
        end if
        do while not result2.EOF
            results2 = result2("Engine")
            if results2 <> request.QueryString("enginename") then
                response.write("<option value='clark_filter.asp?modelName=" &
request.QueryString("modelName") & "&enginename=" & results2 & ">" & results2 & "</option>")
            else
                response.write("<option selected>" & request.QueryString("enginename") & "</option>")
            end if
            result2.moveNext()
        loop
    %>
        </select>
    </td>
    <td style="height: 35px; width: 39px;">
        <strong>Fuel :</strong></td>
    <td style="height: 35px; width: 51px;">
        <select style="width: 103px;"
onchange="window.open(this.options[this.selectedIndex].value,'_top')">

```

```

<%
    fuelstr="select distinct Fuel from Clark_Filter where Model=" +
request.QueryString("modelname")+"" and Engine=" + request.QueryString("enginename")+""
    set result3 = con.Execute(fuelstr)
    if request.QueryString("fuelname")="" then
        response.write("<option selected>Please Select</option>")
    end if
    do while not result3.EOF
        results3 = result3("Fuel")
        if results3 <> request.QueryString("fuelname") then
            response.write("<option value='clark_filter.asp?modelname=" &
request.QueryString("modelname") & "&enginename=" & request.QueryString("enginename") &
"&fuelname=" & results3 & ">" & results3 & "</option>")
        else
            response.write("<option selected>" & request.QueryString("fuelname") & "</option>")
        end if
        result3.moveNext()
    loop
%>
        </select>
    </td>
</tr>
<tr>
    <td style="width: 111px; height: 35px;">
        <strong>Serial Number :</strong></td>
    <td colspan="3" style="height: 35px">
        <select style="width: 310px;"
onchange="window.open(this.options[this.selectedIndex].value,'_top')">
    <%
        serialstr="select distinct [Serial Number] from Clark_Filter where Model=" +
request.QueryString("modelname")+"" and Engine=" + request.QueryString("enginename")+"" and
Fuel=" + request.QueryString("fuelname")+""
        set result4 = con.Execute(serialstr)
        if request.QueryString("serialname")="" then
            response.write("<option selected>Please Select</option>")
        end if

```

```

do while not result4.EOF
results4 = result4("Serial Number")
if results4 <> request.QueryString("serialname") then
response.write("<option value='clark_filter.asp?modelname=" &
request.QueryString("modelname") & "&enginename=" & request.QueryString("enginename") &
"&fuelname=" & request.QueryString("fuelname") & "&serialname=" & results4 & "'>" & results4 &
"</option>")
else
response.write("<option selected>" & request.QueryString("serialname") & "</option>")
end if
result4.MoveNext()
loop
%>
</select>
</td>
</tr>
<tr>
<td style="width: 111px; height: 35px;">
<strong>Filter :</strong></td>
<td colspan="3" style="height: 35px">
<select style="width: 310px;"
onchange="window.open(this.options[this.selectedIndex].value,'_top')">
<%
filterstr="select distinct Filter from Clark_Filter where Model="" +
request.QueryString("modelname")+"" and Engine="" + request.QueryString("enginename")+"" and
Fuel="" + request.QueryString("fuelname")+"" and [Serial Number]=""+
request.QueryString("serialname")+""
set result5 = con.Execute(filterstr)
if request.QueryString("filtername")="" then
response.write("<option selected>Please Select</option>")
end if
do while not result5.EOF
results5 = result5("Filter")
if results5 <> request.QueryString("filtername") then
response.write("<option value='clark_filter.asp?modelname=" &
request.QueryString("modelname") & "&enginename=" & request.QueryString("enginename") &

```

```

"&fuelname=" & request.QueryString("fuelname") & "&serialname=" & request.QueryString("serialname")
& "&filtername=" & results5 & ">" & results5 & "</option>")
    else
        response.write("<option selected>" & request.QueryString("filtername") & "</option>")
    end if
    result5.MoveNext()
loop
%>
        </select>
        </td>
    </tr>
</table>
</td>
</tr>
</table>
<!--webbot bot="Include" U-Include="include/footer.asp" TAG="BODY" startspan -->

<table border="0" width="949" cellspacing="0" cellpadding="0">
<tr>
<td width="949"><p align="center"><font color="#0000ff"
face="Verdana,Verdana,Geneva,Arial,Helvetica,sans-serif" size="1"><i>Last Updated: <script
language="JavaScript">document.write(document.lastModified);
</script> 01/02/2001 15:03:08</i></font><br>
<font color="#000000" face="Verdana,Verdana,Geneva,Arial,Helvetica,sans-serif"
size="1">Designed,
    Filter kit system databasemed & Maintained by the Clark Web Filter kit system databaseming
Team<br>
    For technical problems, send email to the <a
href="mailto:webmaster@clarkmhc.com">Webmaster</a><br>
    ?1998, 1999, 2000, 2001 CLARK Material Handling Company - All Rights Reserved</font></td>
</tr>
</table>
<!--webbot bot="Include" endspan i-checksum="1610" -->
</body>
</html>

```

APPENDIX D

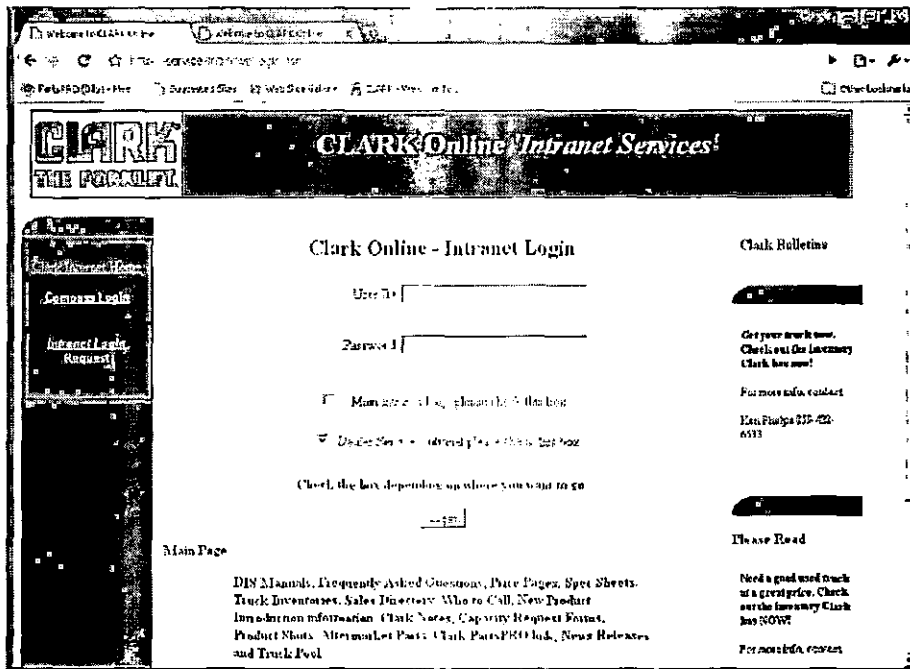
Excel Data for Filter Kit Filter kit system database

Brand	Model	Engine	Fuel	Serial Number	Filter	Clark Part Number	Description
CLARK	F235	Cont. Y112	GAS	ALL	ALL	1807225	BOM REFERENCE NO.
CLARK	F235	Cont. Y112	GAS	ALL	ALL	990936	FILTER OIL
CLARK	F235	Cont. Y112	GAS	ALL	ALL	668788	FILTER AIR
CLARK	F235	Cont. Y112	GAS	ALL	ALL	1804337	FILTER FUEL
CLARK	F235	Cont. Y112	GAS	ALL	ALL	1810620	FILTER HYD
CLARK	F235	Cont. Y112	GAS	ALL	ALL	1779011	FILTER SUMP CAP
CLARK	F235	Cont. Y112	GAS	ALL	ALL	990937	FILTER TRANS
CLARK	F235	Cont. Y112	LPG	2202 thru 3677	ALL	1807226	BOM REFERENCE NO.
CLARK	F235	Cont. Y112	LPG	2202 thru 3677	ALL	990936	FILTER OIL
CLARK	F235	Cont. Y112	LPG	2202 thru 3677	ALL	668788	FILTER AIR
CLARK	F235	Cont. Y112	LPG	2202 thru 3677	ALL	853023	FILTER FUEL (bulkhead element)
CLARK	F235	Cont. Y112	LPG	2202 thru 3677	ALL	1810620	FILTER HYD
CLARK	F235	Cont. Y112	LPG	2202 thru 3677	ALL	1779011	FILTER SUMP CAP
CLARK	F235	Cont. Y112	LPG	2202 thru 3677	ALL	990937	FILTER TRANS
CLARK	F235	Cont. Y112	LPG	3855 thru 3935	ALL	1807227	BOM REFERENCE NO.
CLARK	F235	Cont. Y112	LPG	3855 thru 3935	ALL	990936	FILTER OIL
CLARK	F235	Cont. Y112	LPG	3855 thru 3935	ALL	668788	FILTER AIR
CLARK	F235	Cont. Y112	LPG	3855 thru 3935	ALL	996840	FILTER FUEL (bulkhead element)
CLARK	F235	Cont. Y112	LPG	3855 thru 3935	ALL	1810620	FILTER HYD
CLARK	F235	Cont. Y112	LPG	3855 thru 3935	ALL	1779011	FILTER SUMP CAP
CLARK	F235	Cont. Y112	LPG	3855 thru 3935	ALL	990937	FILTER TRANS
CLARK	F235	Cont. Y112	LPG	4155 and Above	ALL	1807228	BOM REFERENCE NO.
CLARK	F235	Cont. Y112	LPG	4155 and Above	ALL	990936	FILTER OIL
CLARK	F235	Cont. Y112	LPG	4155 and Above	ALL	668788	FILTER AIR
CLARK	F235	Cont. Y112	LPG	4155 and Above	ALL	905053	FILTER FUEL (VFF30 element)
CLARK	F235	Cont. Y112	LPG	4155 and Above	ALL	1810620	FILTER HYD
CLARK	F235	Cont. Y112	LPG	4155 and Above	ALL	1779011	FILTER SUMP CAP
CLARK	F235	Cont. Y112	LPG	4155 and Above	ALL	990937	FILTER TRANS
CLARK	F235	Cont. Y112	GAS	ALL	ALL	1807229	BOM REFERENCE NO.
CLARK	F235	Cont. Y112	GAS	ALL	ALL	990936	FILTER OIL
CLARK	F235	Cont. Y112	GAS	ALL	ALL	668788	FILTER AIR
CLARK	F235	Cont. Y112	GAS	ALL	ALL	1804337	FILTER FUEL
CLARK	F235	Cont. Y112	GAS	ALL	ALL	1810620	FILTER HYD
CLARK	F235	Cont. Y112	GAS	ALL	ALL	1779011	FILTER SUMP CAP
CLARK	F235	Cont. Y112	LPG	ALL	ALL	1807230	BOM REFERENCE NO.
CLARK	F235	Cont. Y112	LPG	ALL	ALL	990936	FILTER OIL
CLARK	F235	Cont. Y112	LPG	ALL	ALL	668788	FILTER AIR
CLARK	F235	Cont. Y112	LPG	ALL	ALL	853023	FILTER FUEL (bulkhead element)
CLARK	F235	Cont. Y112	LPG	ALL	ALL	1810620	FILTER HYD
CLARK	F235	Cont. Y112	LPG	ALL	ALL	1779011	FILTER SUMP CAP
CLARK	F355	Wauk. D155G	GAS	2504	ALL	1807231	BOM REFERENCE NO.
CLARK	F355	Wauk. D155G	GAS	2504	ALL	854571	FILTER OIL
CLARK	F355	Wauk. D155G	GAS	2504	ALL	668788	FILTER AIR
CLARK	F355	Wauk. D155G	GAS	2504	ALL	1804337	FILTER FUEL
CLARK	F355	Wauk. D155G	GAS	2504	ALL	1810437	FILTER HYD
CLARK	F355	Wauk. D155G	GAS	2504	ALL	1779011	FILTER SUMP CAP
CLARK	F355	Wauk. D155G	GAS	2504	ALL	1765756	FILTER STEERING

APPENDIX E

User Manual for Filter Kit Filter kit system database

1. To access the new PM Kit lookup you must first log on to the CLARK intranet at <http://service.clarkmhc.com>



2. Click on Main Intranet Page

