The Benefits of Using a Relational Database Management System

Marymount University

Honor Pledge: I acknowledge that the Capstone Project is an independent study project to be completed individually. On my honor, I have not received aid on my Capstone Project other than what was provided by my faculty mentor and any persons explicitly cited in my work. I further acknowledge that if I have given any aid to another student in this course, the instructor of this course was made aware of my contributions.

Table of Contents

Project Objective

The objective of this paper is to provide an in-depth review of the current body of literature and research that focuses on the numerous and significant benefits of using a relational database management system, RDBMS, as opposed to spreadsheets for data storage, access, and manipulation.

The nature and presentation of my project has changed in both design details and audience after analysis of the feedback provided to me by Professor Narock, Department of Information Technology, Marymount University. After consideration of the points he made regarding the original format, which would have been an Application (hands-on) Project, I determined that my interest in this subject and direction I wanted to pursue- were more suited to a Research Project.

In the topic submission, the intent was to create a crisp, straightforward protocol that database vendors such as Oracle could follow when selling the benefits of converting to a relational database from a spreadsheet system. The audience would theoretically be a Human Resource, HR, employee who would be well versed in the deficiencies and drawbacks of using static spreadsheets to store and manipulate large amounts of data.

Based on feedback from Dr. Narock, he considered the subject of researching the benefits of a client changing to a relational database system to solve business data needs is an "interesting and relevant subject." However, the ability to test this hypothesis and present a side by side technical comparison might prove difficult. According to Dr. Narock, determining the client's ability to understand the protocol would also be problematic since my proposal for an application project did not focus on an actual specific individual or group, but was based on theoretically persuading a potential client to convert to a system such as Oracle's RDBMS.

This project did not receive a first round peer review; therefore, the only feedback being incorporated thus far is from the faculty advisor. When the second peer review is completed, any relevant feedback will be included in this section.

Client

Following the Real-Life-Real-Client-Solutions prompt for investigating a business client who could suggest a typical problem that their business might encounter, and also agree to function as the Topic Expert for the ensuing project, I determined a suitable subject. Since my Fall 2015 internship took me to Redwood City, California, I took advantage of not only my worksite's proximity to Oracle, but also the fact that my supervisor has worked extensively with Oracle product. She introduced me to Dr. John Soltani, Senior Director of IT at Oracle Education, and together we arrived at my topic and research direction.

My research will investigate the history and of spreadsheets and the evolution of relational database systems. The thesis being explored is that a RDBMS is dynamic by nature, as opposed to the static nature of spreadsheets, and able to organize data into tables of rows and columns which can connect to the rows in other tables, thus data is related. (Kruger, 2004) To augment this research I will design and implement a database solution that supports my thesis.

Faculty Advisor

Dr. Tom Narock received his Bachelor of Science from the University of Maryland, his Master of Science from Johns Hopkins University, and his Doctorate in Information Technology from the University of Maryland, Baltimore County. He joined the Marymount Faculty as an assistant professor in 2014 and teaches several courses in Information Technology, including data science, Web development, and mobile app development.

He was awarded a grant from the National Science Foundation last fall and works with other academic institutions and nonprofits to research new methods that assist computers in understanding how data are related.

I chose Dr. Narock to be my faculty advisor because he taught two of my IT classes at Marymount. I am confident he has the knowledge and background to provide feedback and advice on this subject.

Project Plan

Phase One- Articulation of Research Project: Discuss with the client problematic topics suitable to develop an independent study research project. (By Sept. 10)

Phase Two- Preparation of Topic Submission: I will search on-line sources, scholarly journals, and company history in order to better understand my subject and begin to formulate my hypothesis. (By Sept. 18)

Phase Three- Analysis of the Problem: Upon understanding of the initial project requirements and consideration of faculty advisor feedback, I will articulate a thesis that will be supported by research and development of high-level design that will validate my point of view. (by Sept. 28)

Phase Four- Implementation: The project will require installing the Oracle database and its tools. Oracle SQL and PLSQL will be used to create the Oracle database and import the data from the Spreadsheet. PLSQL programs will be created in order to create the user interface and the required database manipulations. (By Oct. 4 and ongoing)

Phase Five- Preparation of Report Rough Draft: An eight page report will be created documenting all phases of the project thus far. It will include more detailed research into the background of spreadsheets and databases. (by Oct. 22)

Phase Six- Preparation of Final Report: After faculty feedback and peer review I will product the final eight to ten page report. Any adjustments or recommendations from reviewers' comments will be considered. (By Nov. 6)

Phase Seven- Presentation: To assist with my presentation, I will create a power point presentation or video tutorial of my project from inception to completion, dependent upon faculty and peer feedback. (By Dec. 4)

Phase Eight-Reflections- I will write a one to two page reflection of my project and the extent to which my research did or did not support my thesis. What could have been done differently or better will be considered. (By Dec 11)

Resources

The resources needed includes access to internet IT sites, on-line internet IT blogs, and Oracle Technology Network site, technical articles, scholarly books, and other on-line resources. Additionally, it was necessary to install Oracle Database Software from the Oracle website. Access to all of the above was free.

Applied Knowledge

In order to perform this research, I applied prior knowledge gained from several of my IT courses at Marymount University. Such courses include: Computer Concepts, Software Engineering, and Java Programming. Technical writing skills from English classes were also helpful.

Completion of this project requires a sufficient understanding of relational database systems, including database modeling, basic understanding of SQL command language, and basic understanding of PL/SQL programming language.

My Topic Expert provided background materials and directions as to how to prepare for this research. Assistance was provided by lectures on the nature of spreadsheets and relational database systems and the advantages of the database design.

Work to be Performed by Others

My client mentor effectively explained and guided me through understanding of the complex and technical concepts and facilitated a richer learning experience for me. I expect such a collaborative effort will continue through to project completion.

Risk Factors

If the potential new client already has an existing database system, additional expenses would be minimal. Otherwise, the client would be required to purchase adequate Oracle database software and licensing. If existing IT personnel cannot sufficiently train users for the new system, then additional expertise would have to be employed.

Project Details

Overview

Research of internet technical documents, media sources such as newspaper and magazine articles, scholarly books and business journals, as well as peer reviewed academic papers, provide ample evidence to support the benefits of relational database management systems, or (RDBMS). A review and presentation of such material, presented in a research paper, will convince potential users of relational database products that such a move would be worth the investment, solve existing problems posed by using spreadsheets, and increase efficiency and productivity. Ideally, a salesperson should be able to refer to the research, facts, and declarations in such a document for assistance in any presentation to a client such as a Human Resource employee at a growing company.

With static spreadsheets, if HR wanted to access usage to include multiple users in multiple locations, there would be serious concerns over the integrity of the data. The problems and limitations of spreadsheets will be qualified and to some extent quantified so that it can be contrasted to a RDBMS . This comparison will be proven and supported by research in such a way that a potential new database customer will understand the benefits of the relational database.

History of Spreadsheet Problems

A key marker in the history of spreadsheets was the 1984 launch of Microsoft Excel, the first spreadsheet to use a graphical interface with pull down menus and a point and click capability. (Power,2004) Spreadsheets do not offer the ability to structure and label data as fully as a database and do not offer the ability to query the database. While using spreadsheets is ideal for creating one time analysis, it can become problematic as the data grows over time; new rows and columns get added, formulas are modified, and data and formulas are not consistently updated. These mistakes could lead to “bad results and decisions.” (Spector, 2013)

There is also a problem when dealing with large spreadsheets due to errors in copying and pasting. If all of the cells are not properly selected, there will be missing information. When dealing with large spreadsheets, such omissions can be hard to spot and consequently a business may be unknowingly working with incorrect data. (Spector, 2013)

Lack of auditing and revision control can make it difficult to determine who changed what and when. Lack of revision control further increases risk errors due to the inability to track and test changes made to a document. This lack of security and control can cause regulatory compliance problems and increase vulnerability to fraud. (Spector, 2013)

Further, large spreadsheets can become disorganized and difficult to use if new cells are not properly identified with a range name. Failure to do so makes sorting and searching difficult and time consuming. Similarly, sorting and searching can become messy if range names are not deleted when all of its cells are deleted. This is an important step in maintaining accurate documentation. (Olshan, 2013)

According to an article by IT expert Jeremy Olshan, published in 2013 in Marketwatch, 88% of spreadsheets have errors. He writes that “while number crunching once reserved for accountants and statisticians are now much easier", thanks to programs like Microsoft Excel, this "best-selling spreadsheet software has also contributed to the proliferation of bad math."

Another analysis of multiple studies dealing with spreadsheet errors was conducted by Ray Panko, Professor of IT at the University of Hawaii and an authority on "bad spreadsheet practices", yielding even more dire statistics. According to Panko, close to 90% of spreadsheet documents contain errors, and when large spreadsheets have "thousands of formulas, there will be dozens of undetected errors." The arduous tasks of maintaining spreadsheets is tedious and vulnerable to human error related to fatigue, or simply not paying attention.(Olshan, 2013)

History of the Relational Database: Proven Solutions

The problems and limitations of static spreadsheets were resolved by the invention of the relational database model created by E.F. Codd in 1970 while he worked at IBM. Before this groundbreaking work, databases were hierarchical and data was arranged in a way that limited data manipulation by the user. Codd’s model addressed the problem posed by the static nature of spreadsheets that made it difficult for an employee to change the data due to the way it was organized. Building on the development of Codd’s database model, a small start-up company named Oracle brought the first relational database product, also called Oracle, to the IT market.

Technical Support for Thesis

One characteristic of an RDBMS is the independence of physical data storage from logical data structures. In Oracle database, a database schema is a collection of logical data structures, or schema objects. A database user owns a database schema, which has the same name as the user name.

Schema objects are user-created structures that directly refer to the data in the database. The database supports many types of schema objects, the most important of which are tables and indexes. (Taylor, 2013)

A Table describes an entity such as ‘employees’. A table is defined with a table name, such as employees, and set of columns. A table contains a set of rows. A column identifies an attribute of the entity described by the table, whereas a row identifies an instance of the entity. For example, attributes of the ‘employees’ entity correspond to columns for employee ID and last name, etc. A row in employees table identifies a specific employee. (Caffrey, 2011)

An Index is an optional data structure that you can create on one or more columns of a table. Indexes provide direct, fast access to rows, which can increase the performance of data retrieval. When processing a request, the database can use available indexes to locate the requested rows efficiently. Indexes are useful when applications often query a specific row or range of rows.

Indexes are logically and physically independent of the data. Thus, you can drop and create indexes with no effect on the tables or other indexes. All applications continue to function after you drop an index. This is a dramatic improvement over spreadsheets that will significantly improve quality and efficacy of database employee work. (Kruger, 2004)

In a database application, maintaining data integrity means ensuring that the data in the tables that the application manipulates conforms to the appropriate business rules. This is critical for a client concerned about safety of data. These rules are called Integrity Constraints. One example is a NOT NULL integrity constraint. This constraint forces the column to contain a value in every row in the table. Oracle ensures this by including ‘Constraints’, which dictate that

a business rule specifies a condition or relationship that must always be true or must always be false. For example, a business rule might be that no employee can have a salary over $100,000 or that every employee in the 'employees' table must belong to a department in the 'departments' table. Business rules vary from company to company, because each company defines its own policies about salaries, employee numbers, and inventory tracking, and so on. For the client, this aspect of relational database provides a critical advance and improvement over spreadsheets. The client can optionally specify rules for each column of a table. (Taylor, 2013)

Structured Query Language (SQL)

A general requirement for a RDBMS is to adhere to the accepted industry standards for a data access language. SQL is a set-based declarative language that provides an interface to an RDBMS such as Oracle Database. In contrast to procedural languages such as C, which describe how things should be done, SQL is nonprocedural and describes what should be done. Users specify the result that they want (for example, the names of current employees), not how to derive it. (Ruel,2014) All operations on the data in an Oracle database are performed using SQL statements. SQL statements enable you to perform the following tasks:

- Query data
- Insert, update, and delete rows in a database table
- Create, replace, alter, and drop objects in the database
- Control access to the database and its objects
- Guarantee database consistency and integrity

Oracle PLSQL

PL/SQL is a procedural extension to Oracle SQL. PL/SQL is integrated with Oracle Database, enabling you to use all of the Oracle Database SQL statements, functions, and data types. You can use PL/SQL to control the flow of a SQL program, use variables, and write error-handling procedures. (Feuerstein, 2013)

Database Transactions

Transactions are one of the features that set Oracle database apart from a file system. A RDBMS must be able to group SQL statements so that they are either all committed, which means they are applied to the database, or all rolled back, meaning they are undone. A transaction is a logical, atomic unit of work that contains one or more SQL statements. (Ruel, 2014)

An illustration of the need for database transactions is a funds transfer from a savings account to a checking account. Such a transfer consists of three separate operations; decrease the savings account, increase the checking account, and record the transaction in the transaction journal. Oracle guarantees that all three operations succeed or fail as a unit. For example, if a hardware failure prevents a statement in the transaction from executing, then the other statements must be rolled back. The basic principle of a transaction is "all or nothing": an atomic operation succeeds or fails as a whole. (Dorsey, 2014)

Data Concurrency

A requirement of a multi-user RDBMS is the control of data concurrency, which is the simultaneous access of the same data by multiple users. Without concurrency controls, users could change data improperly, compromising data integrity. For example, one user could update a row while a different user simultaneously updates it. If multiple users access the same data, then one way of managing concurrency is to make users wait. However, the goal of a DBMS is to reduce wait time so it is either nonexistent or negligible.(Dorsey, 2014) All SQL statements that modify data must proceed with as little interference as possible.(Kruger, 2004) Destructive interactions, which are interactions that incorrectly update data or alter underlying data structures, must be avoided.

Oracle Database uses locks to control concurrent access to data. A lock is a mechanism that prevents destructive interaction between transactions accessing a shared resource. Locks help ensure data integrity while allowing maximum concurrent access to data. (Ruel, 2014)

Data Consistency

In Oracle Database, each user must see a consistent view of the data, including visible changes made by a user's own transactions and committed transactions of other users. For example, the database must prevent dirty reads, which occur when one transaction sees uncommitted changes made by another concurrent transaction.

Oracle Database always enforces statement-level read consistency, which guarantees that the data that a single query returns is committed and consistent for a single point in time. Depending on the transaction isolation level, this point is the time at which the statement was opened or the time the transaction began. The Flashback Query feature enables you to specify this point in time explicitly. (Ruel,2014)

The database can also provide read consistency to all queries in a transaction, known as transaction-level read consistency. In this case, each statement in a transaction sees data from the same point in time, which is the time at which the transaction occurs.

Advantages of Trigger

Triggers are schema objects that are similar to subprograms but differ in the way they are invoked. Triggers are implicitly invoked by the database when a triggering event occurs**.** The correct use of triggers enables you to build and deploy applications that are more robust and use the database more effectively. You can use triggers to automatically generate derived column values, prevent invalid transactions, provide auditing and event logging, and record information about table access. (Dorsey,2014)

You can use triggers to enforce low-level business rules common for all client applications. For example, several applications may access the employees table. If a trigger on this table ensures the format of inserted data, then this business logic does not need to be reproduced in every client. There are many reasons to use triggers to customize a database management system. Triggers let you customize your database management system. For example, you can use triggers to automatically generate virtual column values and log events. One can also gather statistics on

table access and modify table data when DML statements are issued against views. Another use of triggers is to enforce referential integrity when child and parent tables are on different nodes of a distributed database. Users may also publish information about database events, user events, and SQL statements to subscribing applications. Triggers are also used to prevent DML operations on a table after regular business hours and prevent invalid transactions. By utilizing triggers, the client can enforce complex business or referential integrity rules that you cannot define with constraints. (Taylor, 2013)

Conclusion

The capacity and efficiency of relationship database, such as the Oracle RDBMS, is well-suited to a data problem such as the one examined in this paper, that being a typical Human Resource Department issue. The client can store and process very large amounts of data quickly, with several levels of security, and collaboratively. (Kruger,2004) Therefore, the ability of database systems to handle large amounts of data simultaneously is a tremendous advantage for a RDBMS such as Oracle's.

References

Caffrey, Melanie, (2011,Sept/Oct) Get Your Information In Order, *Oracle Magazine*, Retrieved from http://www.oracle.com/technicalnetwork/issue-archive/2013.

Caffrey,Melanie, (2012,Jan/Feb) Getting Answers With Select, *Oracle Magazine*, Retrieved from http://www.oracle.com/technicalnetwork/issue-archive/2013 .

Codd, E.F. (1970). "A Relational Model of Data for Large Shared Data Banks". *Communications of the ACM***13**(6): 377

Dorsey, Dr. Paul, and Rosenblum, Michael, (2014*). Oracle PL/SQL Performance Tuning Tips & Tecniques*. New York: McGraw-Hill Education

Feuerstein, Steven, (2013,Nov/Dec) SQL in PL/SQL Enhancements, *Oracle Magazine*, Retrieved from http://www.oracle.com/technicalnetwork/issue-archive/2013

Kruger, Birgitta, (2004 Oct/Dec) Why You Should Use A Relational Database Instead Of A Spreadsheet*, Cybernetics and Systems* vol. 35 Issue 7/8 pp683-696

Olshan, Jeremy, (2013) *Marketwatch*

Power, D.J., (2004, August) A Brief History Of Spreadsheets, *DSS Resources,* Retrieved from http://dssresources.com/history/sshistory.html

Ruel, C., & Wessler, M. (2014). *Oracle PL/SQL for dummies.* Hoboken, NJ: John Wiley & Sons.

Spector, Lincoln, Five Excel Nightmares*, PCWorld Magazine* Retrieved from www.pcworld.com/article/229504/five-...article

Taylor, A. (2013*). SQL for dummies 6th edition.* Hoboken, NJ: John Wiley.

Additional Research From the following Online Manual from Oracle.Com:

Introduction to Oracle Database, Oracle Database PL/SQL Language Reference, Oracle Database Development Guide, Concepts for Database Developers Guide