Information and Knowledge Management
ISSN 2224-5758 (Paper) ISSN 2224-896X (Online)
Vol.6, No.1, 2016
www.iiste.org

Modelling and Simulation of a Decision Support System Prototype Built on an Improved Data Warehousing Architecture for the School of Postgraduate, MAUTECH, Yola – Nigeria

Ibrahim Inuwa[1] Dr. N. D. Oye[2 *]

1. Department of Information Technology Modibbo Adama University of Technology Yola, Adamawa State - Nigeria.
2. Department of Computer Science Modibbo Adama University of Technology Yola, Adamawa State - Nigeria.

Abstract

A Data Warehouse (DW) is constructed with the goal of storing and providing all the relevant information that is generated along the heterogeneous databases of an organization. The development and management of precise and up-to-date information concerning academic staff, department, faculty, student's academic record etc. is critically important in the management of a university. This study has become necessary because, data warehousing is a new field, a small number of investigations has been done regarding the features of academic data analysis and report. At present, data warehousing is among the best solution for gathering and maintaining data for decision making. Therefore, the aim of this paper is to develop a DW prototype model for the School of Postgraduate Studies' (SPGS) programmes of Modibbo Adama University of Technology (MAUTEC), Yola. The objective of the study is to model and simulate a decision support system that is capable of querying the prototype DW database model to generate reports as output in order to help administrative decision making of the SPGS MAUTEC, Yola. The study has provided relevant literatures in relation to the subject matter. In the methodology, a secondary, field and case study research were conducted. The software engineering development methodology considered was the "Realistic Waterfall Model". The findings of this paper provide a DW prototype database model using a dimensional modeling technique and the graphic user interface tool for reports and analysis. The researchers have demonstrated their understanding on the subject matter and as a matter of fact, possible future work has been suggested from where we stopped.

Keywords - Data Warehouse, Modeling, Simulation, Prototype and Decision Support System

1. Introduction

The formation and management of accurate and up-to-date information regarding academic staff, department, faculty, student's academic record etc. is critically important in the management of a university. University frequently encompasses a lot of subsystems vital for its core processes and operations. Examples of such subsystems include the student registration system, the payroll system, the accounting system, the course management system, the staff management system and many others. In essence, all these systems are connected to many underlying distributed databases that are employed for every day transactions and processes. However, Youssef, (2012) argued that universities hardly employ systems for handling data analysis, forecasting, prediction, and decision making.

Data Warehouse (DW) is a system that retrieves and consolidates data periodically from the source systems into a dimensional or normalized data store. It usually keeps years of history and is queried for DSS or other analytical activities. It is typically updated in batches, not every time a transaction happens in the source system. DW is a foundation for influential data analysis, it supports professional decision making by encouraging administrators and other users to examine data and carry out analysis in a better way. Because the data has been integrated from different sources and sent into the repository, it can facilitate measurement of the effect of various combinations of factors in the University education system (Rainardi, 2008). DW is an essential current issue for many establishments and is relatively a new field in the realm of information technology. As data warehousing is a new field, a small number of research has been done regarding the characteristics of academic data and the complexity of analyzing such data. Educational institutions measure success very differently from business-oriented organizations and the analyses that are meaningful in such environments pose unique problems in DW. In general, a DW is constructed with the goal of storing and providing all the relevant information that is generated along the different databases of an organization (Alejandro and Adriana, 2000). Nowadays, almost every enterprise uses a database to store its vital data and information. For instance, dynamic websites, accounting information systems, payroll systems, stock management systems all rely on internal databases as a container to store and manage their data (Pant and Hsu, 1995).

Despite the availability of more and more powerful computers and communication networks, very few studies have been conducted regarding the characteristics of academic data and the complexity of analyzing such data. A number of administrators hardly get their hands on critical information that already exists in the

University. One of these administrators of the University is the Dean of the Postgraduate School. The School of Postgraduate Studies (SPGS) every day creates data about students, courses, programmes, staff etc., these are operational data, locked up in a myriad of manual and computer systems and are exceedingly difficult to analyze such data and come up with a timely information for decision making. It usually takes a lot of time to get to the information, takes too many people to pull the data together as such, the aim of this paper is to develop a DW Prototype Model for the SPGS programmes of MAUTECH, Yola. The research objective, derived from this aim is to model and simulate a Decision Support System (DSS) that is capable of querying the DW Prototype model to generate reports and analysis to help administrative decision making of the SPGS MAUTECH Yola.

2. Literature Review

Concepts of DSS and DW

This section focuses on providing the essential facts on the relevant literatures that connects to the study. Looking through several definitions we can find that Watson, Rainer and Chang (1991) defined the DSS as an extensible system, capable of ad-hoc analysis and decision modeling, focused on future planning and used at unplanned and irregular time stamps. Also Carlson and Sprague cited by Muntean (2004) defined DSS as being interactive systems that help decision makers use data and models in resolving unstructured and semi-structured economic problems. Turban (1998) defines a DSS as an interactive, flexible and adaptable system, exclusively designed to offer support in solving unstructured or semi-structured managerial problems, aiming to improve the decisional process. A DW is a system that retrieves and consolidates data periodically from the source systems into a dimensional or normalized data store. It usually keeps years of history and is queried for decision support or other analytical activities. It is typically updated in batches, not every time a transaction happens in the source system (Vincent, 2008). Over the last few years, organizations have increasingly turned to data warehousing to improve information flow and decision support. A DW can be a valuable asset in providing easy access to data for analysis and reporting. Unfortunately, building and maintaining an effective DW has several challenges (Güzin, 2007).

Online Transaction Processing (OLTP) systems and Online Analytical Processing (OLAP) systems

Operational systems are referred to as OLTP systems and are designed for real-time data entry and editing. One account, one student record, one inventory item, or one order can be updated or deleted at any time (Güzin, 2007). OLTP systems are useful for addressing the operational data needs of an organization and designed for day-to-day operations like payroll and accounting systems. They are not suitable for supporting decision-support queries. The data is updated continuously. These systems contain vast amount of data which can be difficult to access. However, to get valuable information from this data is crucial for the decision-maker. OLTP systems do not hold the data constantly. Hence the formalization of DW and Data Marts occurs. There are several benefits of OLTP systems. Today's online transaction processing increasingly requires support for transactions that span a network and may include more than one company. For this reason, new OLTP software uses client/server processing and brokering software that allows transactions to run on different computer platforms in a network (Arun and Atish, 2005).

Analytical systems, such as DWs are referred to as OLAP systems, OLAP are designed to help an organization with decision-making. A DW is updated periodically and its solutions offer query capabilities, trend analysis, and reporting (Güzin, 2007). OLAP is a category of software technology that provides fast, consistent and interactive access to information. OLAP is designed for planning, forecasting, managing - decision making and creating business structures and combine them in such a way as to allow users to quickly answer business questions (Surajit et al., 2001). Han and Kamber (2001) argue that an OLTP system is customer-oriented as opposed to a DW that is market-oriented. It is a bit difficult to combine data warehousing (OLAP) and OLTP capabilities in one system. Operational systems and analytical systems that include DWs, they differ in using data. OLTP system's main purpose is to capture information about economic activities of an organization. One might argue that the purpose of OLTP system is to get data into computers, whereas the purpose of DW is to get data or information out of computers (Temitope and Raufu, 2011).

DW Prototyping

Prototyping is the process of building a working replica of a system. The prototype is the equivalent of a mock-up in the hardware world. It may be used with the waterfall in a fashion similar to the Boehm Spiral or it may completely replace it. DeGrace and Stahl (1990) Says: "You get some sort of requirements list. Sometimes it is quite informal. If it is for your customer, the requirements could arrive in some sort of memo. Next, you transform the requirements into a working model by changing or operating your (prototype) to include them. With a 4GL (Fourth Generation Language), you transform the requirements into language and macro commands. With libraries, you write a driver, the top-level program, and select and insert calls to the library functions that represent the requirements. Then, you integrate them by writing code to handle input, output, and error processing functions, operator messages, and connections between functions (pp. 116, 117, 127)".

Demarest (2013) was explicit when it say that planning the developing and deployment of a standard DW should be taken as an IT project, hence what made IT project fail also applies when developing DW prototype, thus the need for Project Planning and following the system development life cycle. There is the need for careful planning, requirements specification, design, prototyping and implementation. According to Where Scape (2013) the model is a cycle rather than a serialized timeline. Based on Where Scape's experience, the cycle repeats every 12 to 18 months and consists of five major phases. Although specific vocabularies vary from organization to organization, the data warehousing industry is in agreement that the DW lifecycle (DWLC) model as can be seen in Figure 1.

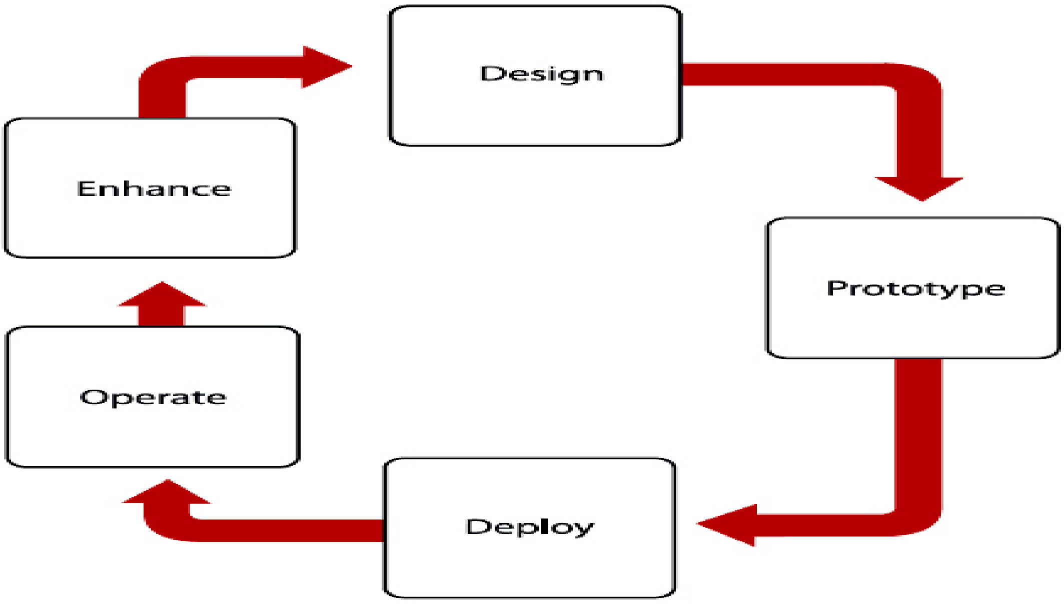

Figure 1: The DW Lifecycle (DWLC) Model (WhereScape, 2013).

The emphasis on prototyping in successful data warehousing projects springs from a generally recognized but little discussed principle shared by experienced data warehousing practitioners, formal requirements in a data warehousing project, even when formally stated by business users, are generally a waste of everyone's time. The reason for this apparently extreme statement is actually quite straightforward, and obvious to anyone who's participated in the early stages of a data warehousing project gone wrong.

Usage of DW in Education

To stay competitive in today's rapidly changing environment, an organization needs an effective database management system. Business enterprises can benefit from data warehousing. The concept of collecting data into separate, multidimensional repositories to handle complex decision making-activities can be used for educational purposes as well. Educational institutions need this capability to ensure quality data management for strategic decision-making. A review of the literature reveals that data warehousing is becoming an increasingly popular way to store and retrieve data. Many educational institutions are creating a DW to provide integrated administrative information for planning and reporting purposes. Inmon stated that; data warehousing is becoming "a solid business strategy" for institutional research. He also declares that, it helps higher education enterprises understand who their customer base is, what they do, and what types of courses and services do they offer affect their learning and satisfaction (Inmon, 1992). Nowadays, many higher education institutions are starting to see the value of the integrated, standardized, clean and easy access to data for better decision making. With the usage of data warehousing, analysis, management decision making and other reports can be done in a simpler way (Güzin, 2007).

DW Software Engineering Development Methodologies

A software development process, also known as a software development life cycle (SDLC), is a structure imposed on the development of a software product. It is often considered as a subset of system development life cycle. There are several models for such processes, each describing approaches to a variety of activities that take place during the process. Software Engineering (SE) is the application of a systematic, disciplined, quantifiable approach to the development, operation, and maintenance of software, and the study of these approaches; that is, the application of engineering to software because it integrates significant mathematics, computer science and practices whose origins are in engineering (Shikha, and Dinesh, 2012). Any software development process is divided into several logical stages that allow a software development company to organize its work efficiently in order to build a software product of the required functionality within a specific time frame and budget (Rajendra, and Dani, 2012).

Velicanu and Bucharest (2007) argued that one of the following software development methods can be used in developing a DW:

i. Waterfall approach, which requires a structured and systematic analysis at each step, before going forward.
ii. Spiral (iterative) approach, which allows fast generation of more and more developed functional systems.

They further argued that, the most adequate method for developing DW is the iterative method. In this approach, much iteration is carried out so that new versions of the system are created. The business matters are approached one after the other. The method provides a scalable architecture and answers the informational demands of the whole organization. It also allows an efficient management of the users' requirements and reduces the possible risks. The waterfall methodology is also known as the sequential methodology. The iterative methodology is also known as the spiral or incremental methodology. The waterfall methodology is one of the oldest methodologies (it has been around for about 30 years) and is widely used by many organizations to build their systems, including their DW systems. The iterative methodology is very different from waterfall. It is not as widely used as waterfall, but it gives good benefits if used correctly. Some people prefer waterfall and some people prefer iterative, we compare the two and describe their advantages and disadvantages, along with some personal experience in implementing them in data warehousing projects (Vincent, 2008). The available literature on DW design mainly focuses on traditional, linear approaches such as the waterfall approach, and it appears to be only loosely related to the sophisticated design methodologies that have been emerging in the software engineering community. Though some works about agile data warehousing have appeared (Hughes, 2008), there are also evidences that applying an agile approach tout court to DW design has several risks, such as that of inappropriately narrowing the DW scope (Beyer and Richardson, 2010). Shikha and Dinesh (2012) opines that software development teams, taking into account its goals and the scale of a particular project, and have a number of well-established software development models to choose from.

In order to analyze the potential advantages arising from the application of contemporary Software Engineering Methodologies to a DW project, Matteo *et al.,* (2011) has proposed the 4WD methodology, a design methodology that combines the main principles emerging from these methodologies to the uniqueness of DW projects. The principles underlying 4WD are risk-based iteration, evolutionary and incremental prototyping, user involvement, component reuse, formal and light documentation, and automated schema transformation.

Therefore, even though there are number of models software Development Company adopts the best-suited model, which facilitates the software development process and boosts the productivity of its team members.

Theoretical Framework

The basis of the study is a framework discussed concerning the design concepts and approach to DW (Ibrahim and Oye, 2015; Ibrahim, and Garba, 2015). Theatrical framework (Top-Down model and Bottom-Up model), the two-design processes are referred to as Top-down process, as described by Inmon, (1993), and Bottom-up process as described by Kimball and Ross (2002).The design of the DW depends on the approaches of the two fathers of DW developers. Bill Inmon begins with an Extraction, Transformation, and Loading (ETL) process working from legacy and/or external data sources. Extraction transformation, process data from these sources and output it to a centralized Data Staging Area. Following this, data and metadata are loaded into the Enterprise DW and the centralized metadata repository. Once these are constituted, Data Marts are created from summarized data DW and metadata. In the top-down model, integration between the DW and the Data Marts is automatic as long as the discipline of constituting Data Marts as subsets of the DW is maintained.

The central idea in Bottom-up process is to construct the DW incrementally over time from independently developed Data Marts. The process begins with ETL for one or more Data Marts. No common data staging area is required. There is generally a separate area for each Data Mart. There may not even be standardization on the ETL tool. The Model was introduced by Ralph Kimball. For the purpose of this study, Ralph Kimball's approach is adopted, which is the Kimball's development lifecycle, this states with one Data Mart (e.g. Department of IT ; Ibrahim and Oye, 2015) later on further Data Mart are added e.g. Department Computer science and so on. Data flows from sources into Data Marts, then into the DW. Due to the time constraint, it is easier to complete a process for a subset of the school's departments based on the Data Mart and link it up to the prototype DW gradually.

3. Methodology

Conducting any type of research should be governed by a well-defined research methodology based on the scientific principles (Eldabi, 2002). This section is consisting of the approaches that the researchers used to carry out the study. There are three major phases in this section, which include analysis, design and the DW prototype development. The following methods were used during the study:

i. Secondary Research: Due to the time constraint, it allows us to move close to the aim of this study by examining the existing data collated by the SPGS MAUTECH.
ii. Field based Research: To better comprehend the focus of the study we did a little of field research in the form of questions through the interview. Interviews has been conducted with some selected people who have been involved in the decision making process of the university. The interview is with some administrative and academic staff of the SPGS MAUTECH, Yola.
iii. Case study: The SPGS MAUTECH, Yola is considered as our case study to evaluate the objective of the research.

System Analysis and Research Methods

This phase involves outlining the functions that the DW prototype can achieve, and an ideal working environment in which the DW prototype is delivered. Also in this stage, the decision maker's requirements are to understand the workings of users in relations to the school's requirement and how they want to use the solution. What data are they currently making use of and what would they like to do with such data? The analysis stage involved a detailed learning of the existing system, leading to requirements of a new system. It also involved a detailed study of various operations performed by the system and their relationships. The requirements are gathered through a chain of interviews with the different stakeholders. Answers from these users generate the requirements needed for further design of the DW prototype. During the analysis, we studied the happenings of the SPGS and we selected three departments (Information Technology (IT), Mathematics (MA) and the Postgraduate school (PG)) to design their respective Data Marts. We decided to consider the SPGS as a department in this study. Within the scope of this study, we looked at developing Data Marts for these departments, and the integration of these Data Marts formed the SPGS DW prototype. Also at this phase, we had an interview session with some decision makers in the departments; they also agreed to share with us more about what they do, how they do it and in what ways the study can be of support to them.

Functional requirements

Requirements are divided into functional requirements and non-functional requirements. Functional requirements are related with detailed functions, tasks or actions the system must support. The functional requirements are as follows:

i. The users should be able to access data from different SPGS's application in one single location.
ii. The users should be able to access data from different SPGS's application in a setup which is easy to operate.
iii. The system should be able to display figures and charts.
iv. The system should be able to update figures and charts.
v. The users should be able to download displayed figures and chats.
vi. The users should be able to print downloaded figures and charts.
vii. The system should be able to render reports in different formats which are useful to the users.
viii. The users should be able to access the system from anywhere at any point in time (24/7).

Nonfunctional requirements

i. The system should be based on windows authentication so that user does not require to log on to the application many times. Single sign-on.
ii. The front end application should be web enabled and no installation is required on users system.
iii. The front end application should be integrated into the existing portal like Share-Point and intranet.
iv. User level permission is required in order to protect the integrity of the data and restrict user's accessibility to data.
v. The system should perform very well at all times and should be easy to recover after system down time.
vi. The system should be able to keep up to-date information at all times.

User requirements

i. Ability to use data from DW Prototype database for data mining activities.
ii. Ability to perform complex queries to produce reports and analysis with little effort.
iii. Ability to download view and print information from the DW prototype and use it for further analysis.
iv. System reliability at all times.

System requirements

For the purpose of this study, the system requirements are divided into Hardware and Software. The Hardware system requirements are:

i. Central Processing Unit (CPU): Intel (R) CORE i5-2430M CPU @ 2.4GHz.
ii. Installed memory (RAM): 4.00 GB.
iii. Monitor: Super VGA (AGP 32MB).
iv. Peripherals: Keyboard, Mouse, USB and etc.

The Software system requirements are:

i. System type: 64-bit Operating System (Windows 7 Ultimate).
ii. Database engine: Microsoft SQL Server 2005 and 2012.
iii. ETL tools: SQL Server Integration Service (SSIS).
iv. Visual Studio 2010 Professional
v. Programming Language: ASP .NET, XML, CSS and C#.
vi. Web browser: Firefox and Internet Explorer.

System Design

This phase is focused to translating the systems requirements into a set of specifications through physical data models or Data Marts for the DW prototype. As mentioned in our literature review, the two fathers of DW are Inmon and Kimball (see section 2). They have different approaches to the design of the DW. For the purpose of this study, we have adopted the Kimball's design approach. The data from each department is treated as a Data Mart and we designed from start to finish a complete Data Marts for the selected departments. One of the main aims of the DW is to extract data from different OLTP or flat file sources and combine them in a single source for easy access and make best use of the data. The two processes of DW are data load and access. The design of the database started with the principle and theories of database design and the rule that support the departmental needs.

The Software Development Methodology Adopted

In a DW development, the DW life cycle and the Software engineering development methodologies can contribute meaningfully because it creates well established approach and technique for the development processes. DW development is an iterative process as such we have used the "Realistic Waterfall Model" for the DW prototype development. The Realistic Waterfall model is a risk oriented iterative enhancement, and recognizes that requirements are not always available and clear when the system is first implemented. Since designing and building a DW is an iterative process, the realistic waterfall model is one of the development methodologies of choice. This is to ensure that any business requirements not clear at the beginning of the analysis stage can be often revisited. The Figure 2 shows a realistic waterfall model in the spiral model fashion of the DW life cycle.

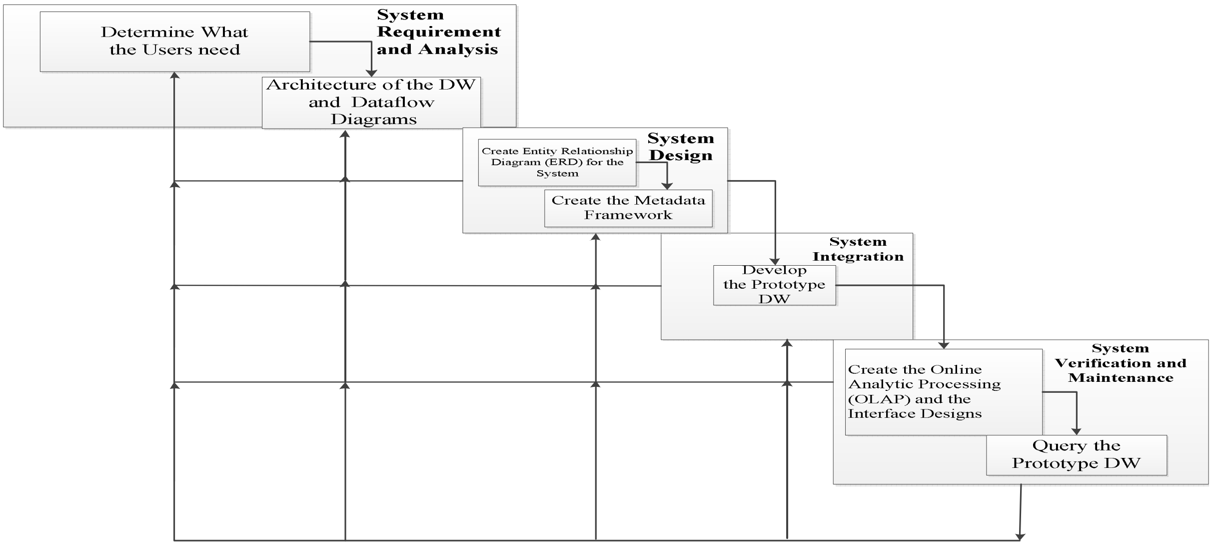

Figure 2: A Realistic Waterfall Model in the Spiral Model of the DW Life Cycle.

By removing the dotted arrows from Figure 2 that means we are going to have the traditional Waterfall Model deprived of any iteration. The traditional Waterfall Model is frequently analyzed by software engineers because it cannot accommodate changes in the process and the requirements analysis is frozen at some point in the development process, another challenge is the bottlenecks in the testing period (previously discussed in section 2). The testing period is the riskiest time in a waterfall methodology. But to be fair, there are ways to mitigate these risks. For instance, we can build a prototype (which is not far away from one of the objectives of this study) where you put all the components together and run them as a whole architecture to detect potential issues early. Furthermore, by building a prototype, the users are able to see the user interface early. The researchers started with Data Mart design by identifying the measures of the fact tables. The measures are the basis for feedback information that the decision makers require. For the purpose of this study, we adopted the star schema for the prototype DW database design.

4. Results and Implementation

System Prototype Development and Validation

The system prototype development is the actual application of the analysis and design that has been carried out. In this phase of the study, we have designed the DW (Fact and dimension tables) prototype, the ETL (Extract, Transform and Load) and the front end application for the purpose of this study. Validation process involves the confirmation by examination that an information system (DW prototype) has been realized appropriately and it is in conformity with the User's needs and intended use.

The researchers have designed the actual physical Data Mart Models of all the independent departmental Data Marts on an SQL Server Database Management System 2005. The actual database tables can be seen in the Figure 3.

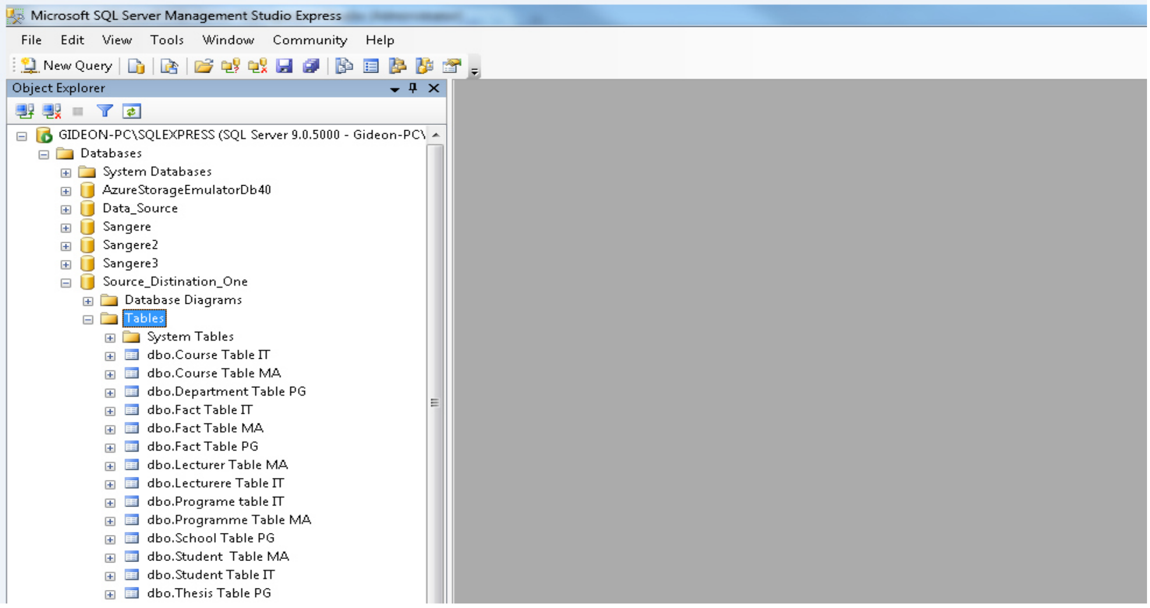

Figure 3: Physical Data Mart Tables on an SQL Database.

The main column is a unique key in the dimension tables and it is the primary key, and the same primary keys of the dimension tables are serving as foreign keys on the Fact tables, this is so because the researcher needs to enforce a referential integrity. The primary keys on the dimensional tables are usually the surrogate keys, they are unique and not null; and they uniquely classify the records on the dimension tables. The surrogate keys are used because the data to each of the dimensional table is from different source and there is need to have a unique key to identify the record and that is why the referential integrity is very essential in realm of database. The referential integrity allows you to establish relationships (parent-child relationship) in database amongst tables, with the purpose of ensuring that every row in the child table has a corresponding parent entry in the parent table. The DW is enhanced for data recovery and it is very essential that users are able to run their reports and analysis as quickly as possible. In the data storage, it is good to have a database structure and the correct index. Indexes are the indicators to the record stored in a database. In the concept of the DW, indexes are very essential and they help in the loading and data retrieval of the DW. Indexing can significantly improve the query and loading performance of data warehousing. The researcher has fact dimensional tables, in dimensional modeling and they need different indexing and primary keys. The researcher also used this surrogate key column the grouped index of the dimension table. The reason for doing this is because in the DW, the dimension tables are joined to the Fact table on the surrogate key column. As this will allow for a good query performance, for the reason that in database a grouped index usually regulates the order of how the rows are physically stored on.

Now, coming to the Fact tables, which contains the Keys to the Dimension tables so that it can uniquely identify the record in a table. The physical ERD design of the Data Marts Models is presented in the Figure 4, Figure 5 and Figure 6. With one to many entity relationships and referential integrity was enforced.

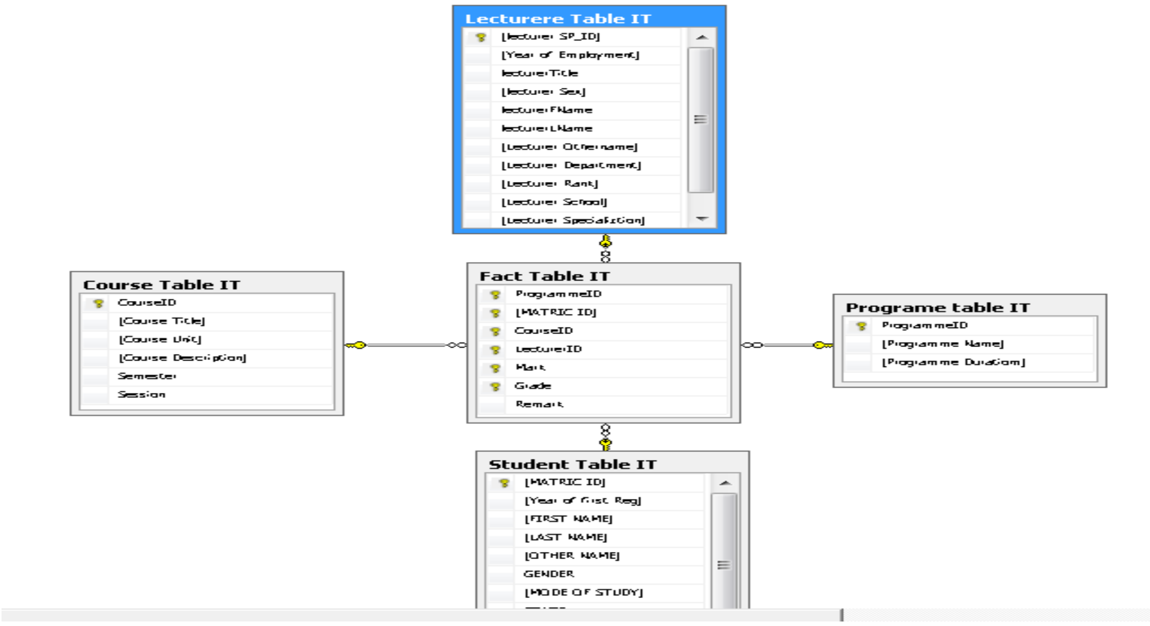

Figure 4: Data Mart Physical Model for the IT Department by a Star Schema.

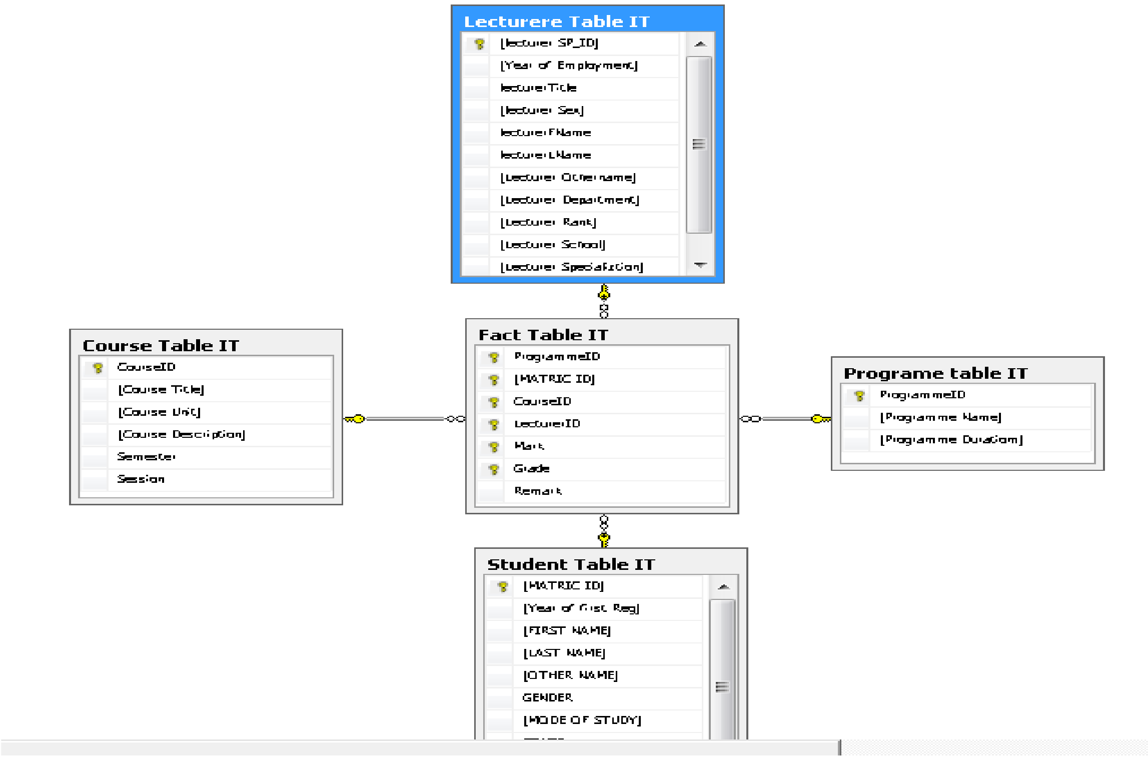

Figure 5: Data Mart Physical Model for the MA Department by a Star Schema.

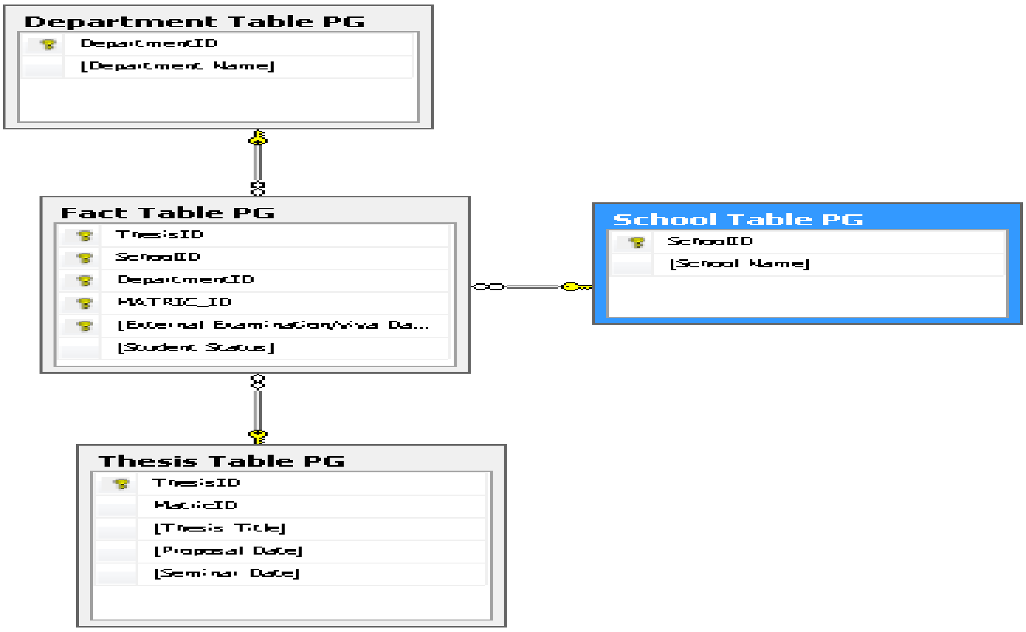

Figure 6: Data Mart Physical Model for the PG Department by a Star Schema.

Only one data source system was considered for the purpose of this study, imaginary data was generated by simulation using the SQL Server Database Management System 2005. The data we generated is for five years (five sessions: 2005/2006 – 2009/2010). The staging area is a database we created also on an SQL Server Database Management System 2005, to serve as a cache area of the ETL processes. The actual design of the staging database was carried out by executing many scripts on the SQL Server Database Management System 2005 as can be seen in the Figure 8.

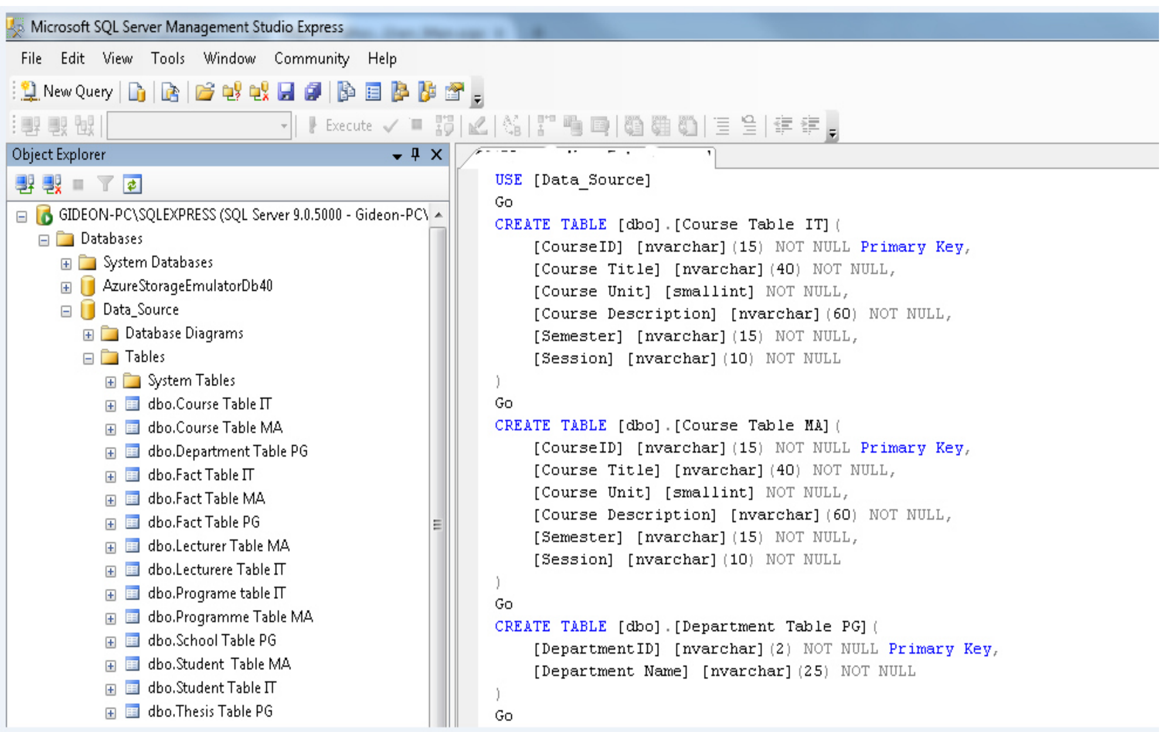

Figure 8: The SQL Server 2005 Showing Some Scripts for the Staging Database Tables.

The physical ETL design for the data load from the source system to the staging database was design using SSIS; the Figure 9 demonstrates some of the loading processes to the data staging area. Data from the source system was staged temporarily at the staging area before it was transferred into DW database properly. It became easier for us to create the staging database because we copied tables directly from the source system without a change to their attributes. The database structure of the staging area is comparable to that of the source system. From the design of the ETL process architecture, the researcher came up with the design of the physical ETL process as can be seen in the Figure 9. It shows some of the Dimension and Fact tables that were extracted from the source system into the staging database area. The tables were cleansed and refreshed at this point and ready for transfer into their respective departmental Data Marts. Having designed the Fact and Dimension tables and the extraction of data from the source system, then the researcher populated the Data Marts with the data that was extracted earlier from the staging databases. It is now from the Data Marts that another ETL was performed to transport the data into the DW prototype database.

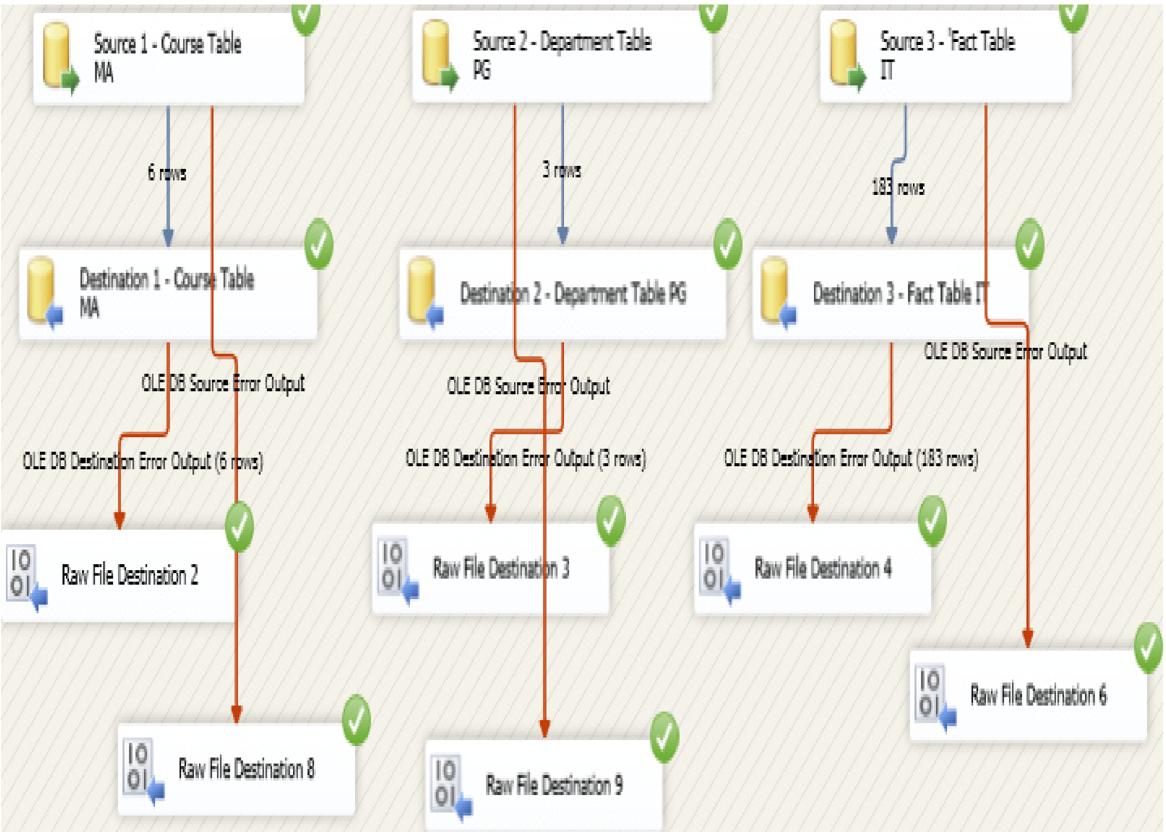

Figure 9: First Data Extraction from Source System to the Staging Area Database.

Creating and loading of the DW prototype database

Our DW database was created on an SQL Server 2012, and the data loading was also done through the SSIS package. Fact and Dimension tables were pushed into the DW Prototype database, the Figure 10 shows Physical database of the DW Prototype Model by Star Schema. The DW Prototype database is the one that integrates the Fact and Dimension tables of departments into the SQL Server 2012. It was the part of the SQL Server application that we used for the database repository. The DW Prototype database was populated with the correct data of good quality that we can make use of as the data repository. DSS is all about making the best use of the available data in order to make better decision about the organization.

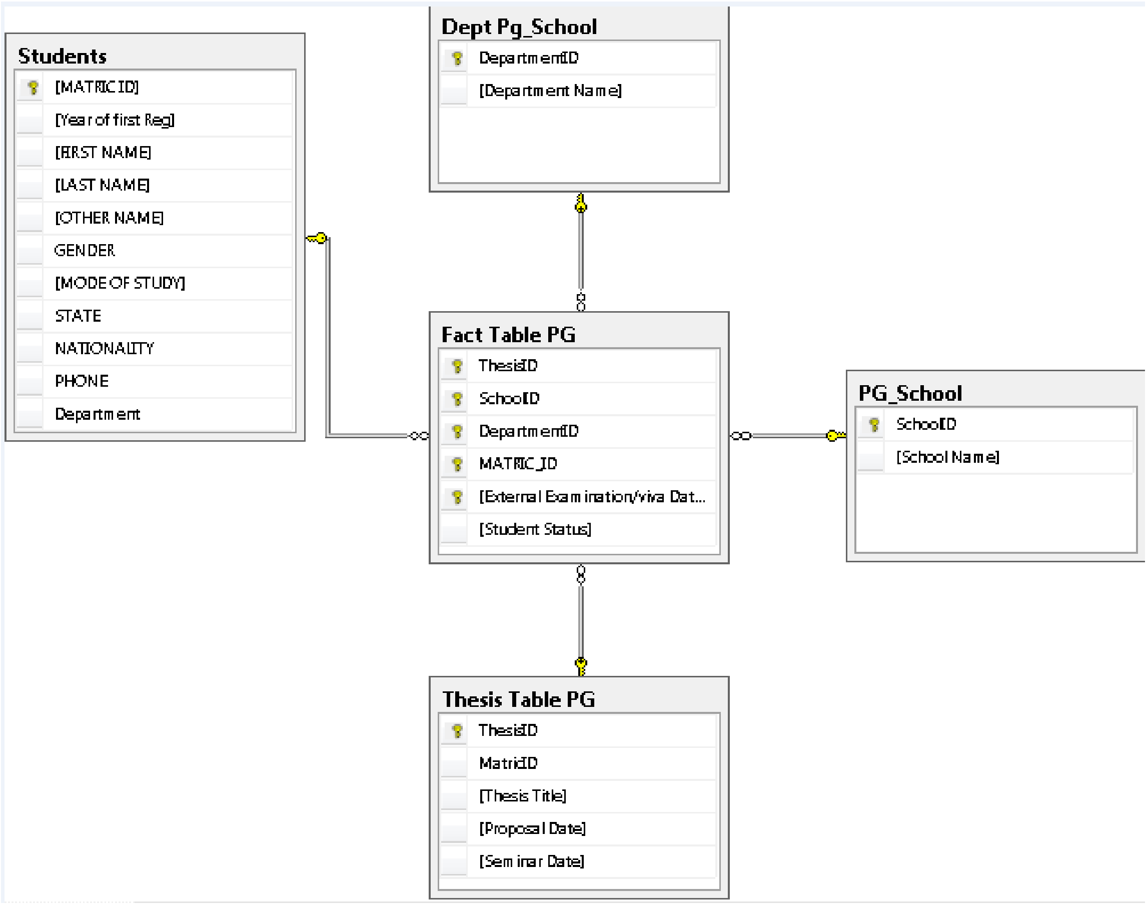

Figure 10: The DW Database Physical Model by a Star Schema.

System Verification, Sample Reports and Analysis

The best way to verifying the simulated data in the DW Prototype is to prepare queries on these data. In this study, certain sample reports were presented and the purpose of these reports is to demonstrate the usefulness of the DW approach to decision making. Even though these reports were based on the user requirements, it was not planned to design a full set of management reports and analysis. This set of sample reports can be used as a basis for generating more comprehensive sets by applying complex queries on the data. We have classified the reports that we generated from the DW Prototype Model based on Lecturer, Student and course.

The User Interface (Input and Output designs)

The Interface design comprises of both input and output designs. We have used the SQL Server reporting Services 2012 to build reports based on the data contained in DW prototype database. The SQL Server Reporting Services 2012 is a special Server that is used for reporting and analysis service, it is linked with the DW database Server for reporting and analysis activities. The system input was designed so as to receive data from the users and administrators of the DW system. Figure 11 shows the User interface for analysis and reporting services.

A system output is the most important component of a working system because the interactivity of the system depends on its output. This is the main reason why the output of an information system determines the effectiveness and efficiency of the system. The system output present information to the users and administrators based on the required queries performed on the DW database. The researchers have presented sample reports and the motive of these reports is to demonstrate the usefulness of the DW approach. The reports were interactively produced by the use of the SQL Server Reporting tool based on user's requirements. The reports were scheduled based on Lecturer and student reports.

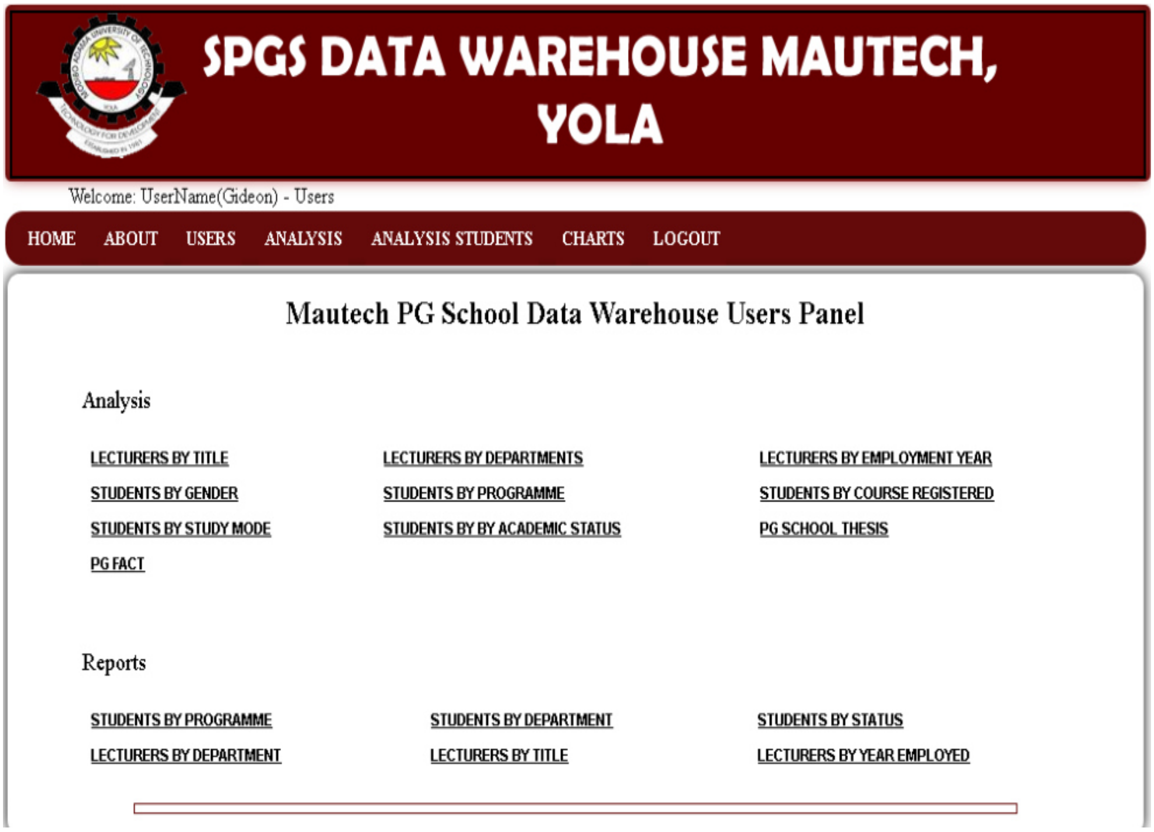

Figure 11: The User Analysis and Reports Interface.

Reports based on lecturers and students analysis

The Figure 12 gives changes in the number of Lectures according to their year of employment.

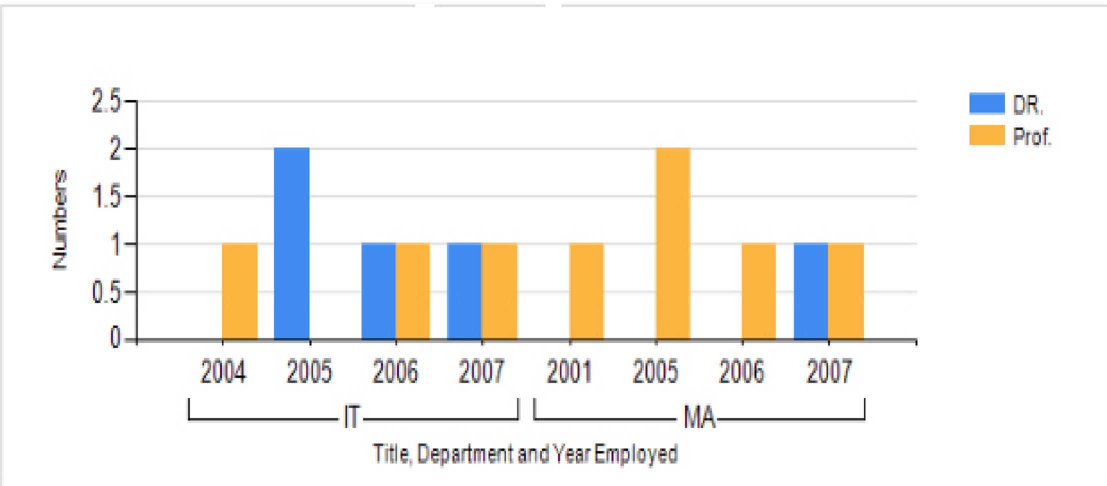

Figure 12: Lecturers by their Title, Department and Year of Employment

Figure 13 presents us with information based on the number of male and female (gender) students in each department from 2005 - 2010. The report is useful in presenting the demographic behavior of students in a long run. This graph provides information to the decision maker about the department and program having higher concentration of students in a long run. The decision maker can use the distribution to determine the trend and pattern of student enrolment by their choice of study mode in a very long run.

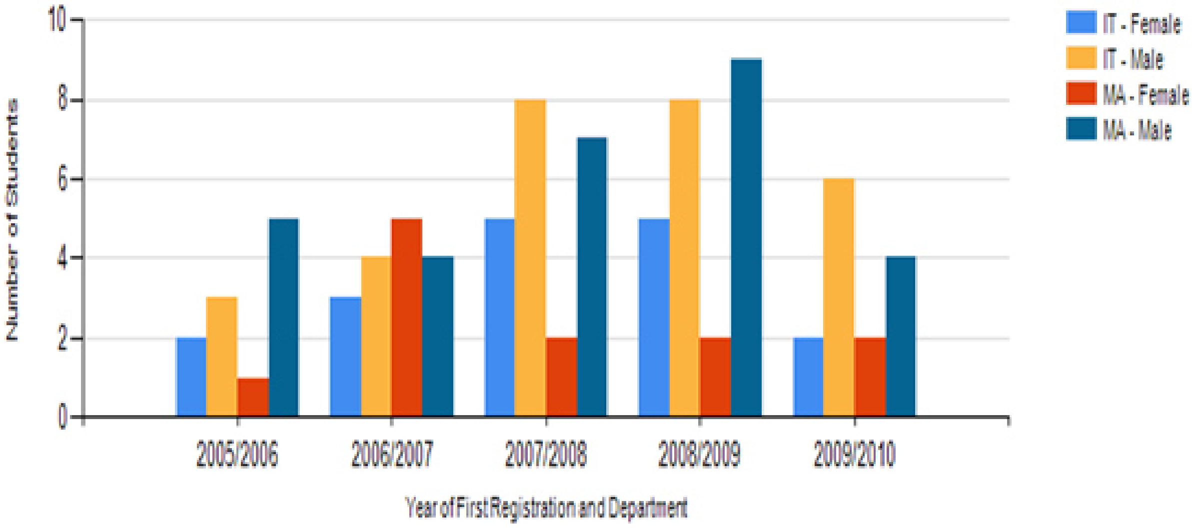

Figure 13: Student by their Gender and Year of First Registration.

5. Conclusion Discussion And Recommendations

The research has introduced the idea of the DW prototyping within the context of the University atmosphere, to better incorporate systems for simpler and improved data analysis, reporting and querying. The DW Prototype we built contains only the data that is required for the reporting and analysis for the purpose of this study and it can be updated periodically, such that all the data can be integrated from different source systems into the central DW. Periodic update on a daily basis is possible because contemporary technologies have narrowed the gap between updates. This can enable organizations to have a "real time" DW which can be analyzed using an OLAP Server (the replica of the DW itself).

The researcher focused on developing a DW and DSS prototype for a University decision making using MAUTECH, Yola as our case study. The study considered the ideologies of data warehousing in the course of this research and demonstrated how data can be incorporated from diverse heterogeneous source systems into a sole historical data storehouse (DW) that is capable of conveying a DSS to the University's administrators and other end users. The researcher have also been able to develop a data reporting and analysis tool (GUI) so that by clicking a button users can interact with the system to get a speedy and timely information needed for the University decision making.

The design of the DW Prototype was based on three departments (IT, MA and PG) within the SPGS; Data Marts was developed based on these departments and the integration of these Data Marts forms the SPGS prototype DW. The ideologies that the researcher had followed to develop this prototype system makes it scalable and as such, it can be adopted by institution of learning. Institutions of higher learning, particularly in Nigeria can begin to implement this project as part of their strategic decision making tool. The DW Prototype is a DSS that can be implemented by any University provided that the project must have a strong goal of supporting both University administrators and other intended users. It is practical that DSS should be part of the institutional decision making processes.

Developing a DW for a University DSS is very essential, particularly for academic resource planning and decision-making. DSS must not only support us to understand the past, but also it strive us to work towards new prospects.

Lastly, the researchers have concluded the study by briefly stating the contributions the study has made to the body of knowledge as follows:

Model and Simulate a Decision Support System (DSS) that is capable of querying the DW Prototype to generate reports and analysis as output in order to help administrative decision making of the SPGS MAUTECH Yola.

The DSS prototype we realized in this study could serve as a platform for designing, modeling and implementing data mining algorithms that will explore our simulated data to find the patterns and relationships that describe the data, and to predict the unknown or future values of the data. No wonder, Vincent (2008) believed that the key value of data mining is the ability to understand why some things happened in the past and the ability to predict what will happen in the future. To refer to predicting the future with regard to data mining, some people use the term forecasting, and some call it predictive analytics. On the other hand, when data mining is used to explain the current or past situation, it is called descriptive modeling, descriptive analytics, or knowledge discovery.

In this project, the scope was limited to only three programme of study from two different departments and few numbers of staff and students were considered. The scope of the study could be considered in future. In addition to these, building a more user friendly and stress free to use querying, reporting and analysis tools could be planned for a future study:

i. Full Implementation of the DW Prototype Model should be considered.
ii. This study did not cover the security policy aspect of the DW, while implementing this model security policy should be considered first.
iii. Feasibility study to determine the cost of implementing DW should also be considered.

DW systems are characterized by a long and expensive development process that hardly meets the ambitious requirements of today's market. We suggest that more research on the methodological problems related to DW design should be considered, with the aim of improving the development process from different points of view.

REFERENCES

Alejandro, G. and Adriana M. (2000). An Overview of Data Warehouse Design Approaches and Techniques. Instituto de Computación, Facultad de Ingenieria, Universidad de la República, Montevideo, Uruguay.
Arun, S. and Atish, P. S. (2005). A Comparison of Data Warehousing Methodologies. *Communications of the ACM*.48, 3.
Beyer, M., Richardson, J. (2010). Agile techniques augment but do not replace Decision Support System and Data Warehouse best practice. Tech. Rep. G00201031, Gartner Research.
Chang S. S., (1991). The Impact of Technology on Users and the Work place. *New Directions for Institutional Research*. 103. 3 – 21.
DeGrace, P. & Stahl, L. H. (1990). *Wicked Problems, Righteous Solutions: A Catalogue of Modern Software Engineering Paradigms*. Reprinted with permission of Prentice Hall, Englewood Cliffs, New Jersey, pp. 116, 117, and 127.
Demarest, M. (2013). Data Warehouse Prototyping: Reducing Risk, Securing Commitment and Improving Project Governance. Retrieved June 28, 2014, from: http// www.wherescape.com.
Eldabi, T. (2002). Quantitative and Qualitative Decision Making Methods in Simulation Modeling. *Journal of Management Decision*. 40, 64-73.
Güzin, T. (2007). Developing a Data Warehouse for a University Decision Support System. The Graduate School of Natural and Applied Sciences, Atilim University.
Han, J. and Kamber, M. (2001). Data Mining: Concepts and Techniques, San Francisco: *Morgan Kaufmann.*
Hughes, R. (2008). Agile Data Warehousing: Delivering world-class Decision Support System using Scrum and XP. *IUniverse.*
Ibrahim, I. and Garba, E. J. (2015a). An Improved Data Warehouse Architecture for SPGS MAUTECH, Yola – Nigeria. *West African Journal of Industrial and Academic Research.* 14, (1).
Ibrahim I. Oye, N. D. (2015b). Design of a Data Warehouse Model for a University Decision Support System. *Journal of Information and Knowledge Management*. 5, (12).
Inmon, W.H., (1993). *Building the Data Warehouse.* A Wiley QED publication. John Wiley and Sons, Inc. New York 123-133.
Inmon, W. H. (1992. a). *Building the Data Warehouse*. New York. John Wiley & Sons, Inc.
Kimball, R. & Ross, M. (2002). *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. New York. John Wiley & Sons, Inc.
Matteo, G., Stefano R. and Elisa, T. (2011). Modern Software Engineering Methodologies Meet DW Design: 4WD. *DEIS, University of Bologna. A. Cuzzocrea and U. Dayal (Eds.): DaWaK. 6862, 66–79.*
Muntean, M. (2004). *Initiation in OLAP Technology: Theory and Practice*, ASE Publishing House, Bucharest.
Pant, S. and Hsu, C.(1995). Strategic Information Systems Planning: A Review Information Resources Management Association International Conference, Atlanta.
Rainardi, V. (2008). *Building a Data Warehouse*. New York. Apress, Springer-Verlag New York, Inc.
Rajendra, G. S. and Dani, A.R., (2012). Comparative Study of Prototype Model for Software Engineering with System Development Life Cycle. *IOSR Journal of Engineering (IOSRJEN). 2, 7, 21-24.*
Shikha, M. and Dinesh, C. J. (2012). A Comparative Analysis of Different types of Models in Software Development Life Cycle. *International Journal of Advanced Research in Computer Science and Software Engineering*. 2, 5,285-290.
Surajit, C. Umeshwar, D. and Venkatesh G. (2001). Database Technology for Decision Support Systems. *Journal of IEEE Computer science*. 34, 12, 48-55.

Temitope, A. O. and Raufu, O. O. (2011). Design of Data Warehouse and DSS: A case study of a Retail Industry. School of Computing, Blekinge Institute of Technology, Karlskrona, Sweden.

Turban, E. (1995). *Decision support and expert systems: Management Support Systems.* New Jersey. Englewood Cliffs, Prentice Hall.

Velicanu, M. & Bucharest, G. M., (2007). Building a Data Warehouse step by step. *Journal of Informatica Economică, nr. 2, 42.*

Vincent, R. (2008). *Building a Data Warehouse with Examples in SQL Server.* New Jersey. Englewood Cliffs, Prentice Hall.

Where Scape, (2013). Data Warehouse Prototyping. Retrieved June 28, 2014, from: *http//* *www.wherescape.com.*

Youssef, B. (2012). A Data Warehouse Design for a Typical University Information System. *Journal of Computer Science & Research (JCSCR).* 1, 6, 12-17.

[1]**Ibrahim Inuwa** receives his Associate degree in Information Management Technology and Bachelor's Degree in Mathematics with Economics from the Federal University of Technology, Yola – Nigeria in 2004 and 2010 respectively. At the moment he is a Masters student (M. Tech. in IT Management) in the Department of Information Technology, Faculty of Management and Information Technology Modibbo Adama University of Technology, Yola – Nigeria. +2348065273810: ibrahiminuwa1985@gmail.com

[2]**Dr. Oye, N. D**. receives his M.Tech OR (Operations Research) degree from the Federal University of Technology Yola-Nigeria in 2002. He is currently a lecturer in the same University (for the past 15 years). He obtained his PhD in Information System at the Department of Information System, Faculty of Computer Science Univeristi Teknologi Malaysia in 2013. +2347037460352; oyenath@yahoo.co.uk.