DEVELOPING A DATA WAREHOUSE FOR A UNIVERSITY DECISION SUPPORT SYSTEM

A MASTER'S THESIS

in

Computer Engineering

Atılım University

by

GÜZİN TÜRKMEN

SEPTEMBER 2007

DEVELOPING A DATA WAREHOUSE FOR A UNIVERSITY DECISION SUPPORT SYSTEM

A THESIS SUBMITTED TO

THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES

OF

ATILIM UNIVERSITY

BY

GÜZİN TÜRKMEN

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

IN

THE DEPARTMENT OF COMPUTER ENGINEERING

SEPTEMBER 2007

Approval of the Graduate School of (Name of the Graduate School)

_____

(Prof. Dr. Selçuk Soyupak)

Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science/Arts.

_____

(Prof. Dr. İbrahim Akman)

Head of Department

This is to certify that we have read this thesis and that in our opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science/Arts.

_____	_____
(Prof. Dr. Ali Yazıcı)	(Asst. Prof. Dr. Nergiz Ercil Çağıltay)
Co-Supervisor	Supervisor

Examining Committee Members

Prof. Dr. İbrahim Akman	_____
Prof. Dr. Ali Yazıcı	_____
Assoc. Prof. Dr. Mohammad Rehan	_____
Asst. Prof. Dr. Nergiz Ercil Çağıltay	_____
Asst. Prof. Dr. Çiğdem Turhan	_____

ABSTRACT

DEVELOPING A DATA WAREHOUSE FOR A UNIVERSITY DECISION SUPPORT SYSTEM

Türkmen, Güzin

M.S., Computer Engineering Department

Supervisor: Asst. Prof. Dr. Nergiz Ercil Çağıltay

Co-Supervisor: Prof. Dr. Ali Yazıcı

September 2007, 85 pages

Data warehouse is an important contemporary issue for many organizations and is relatively a new field in the realm of information technology. As data warehousing is a new field, little 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 data warehousing. The purpose of this thesis is to provide a decision support system that will query the data taken from the existing Student Information System and generate reports as outputs in order to help administrative decision-making in the Atılım University Master Program.

Keywords: Decision Support Systems, Data warehouse, Education

ÖZ

ÜNİVERSİTE KARAR DESTEK SİSTEMİ İÇİN VERİ AMBARI TASARIMI

Türkmen, Güzin

Yüksek Lisans, Bilgisayar Mühendisliği Bölümü

Tez Yöneticisi: Asst. Prof. Dr. Nergiz Ercil Çağıltay

Ortak Tez Yöneticisi: Prof.Dr. Ali Yazıcı

Eylül 2007, 85 Sayfa

Veri ambarı bir çok organizasyon için çağdaş bir meseledir ve bilgi teknolojileri için yeni bir alandır. Özellikle eğitimsel amaçlı kullanımda, veri ambarları birçok fayda sağlar. Veri ambarları yeni bir alan olduğundan, akademik veri yapıları ve bu veriyi analiz etmedeki karmaşa ile ilgili çok az sayıda araştırma yapılmıştır. Eğitim kurumları başarıyı, ticari amaçlı organizasyonlardan çok farklı ölçer ve bu çevrede anlamlı olan analizler veri ambarlamada nadir problemler ortaya çıkarır. Bu tezin amacı, var olan Öğrenci Bilgi Sisteminden (MasterSIS) alınan veriyi sorgulayacak bir karar destek sistemi hazırlamak ve Atılım Üniversitesi Yüksek Lisans Programı'nda akademik karar vermeyi destekleyici rapor çıktıları almayı sağlamaktır.

Anahtar Kelimeler: Karar Destek Sistemleri, Veri Ambarı, Eğitim

ACKNOWLEDGEMENTS

I would like to thank my thesis committee for their time, interest, and input in this project. I would especially like to thank my supervisor Asst. Prof. Dr. Nergiz Ercil ÇAĞILTAY for her patience, encouragement, and wisdom and my co-supervisor Prof. Dr. Ali YAZICI for his support.

I should also express my appreciation to examination committee members Prof. Dr. İbrahim AKMAN, Assoc. Prof. Dr. Mohammad REHAN, Asst. Prof. Dr. Çiğdem TURHAN for their valuable suggestion and comments.

To my family, thank you all so much for your support. I couldn't have done this without you.

Finally, thank you to Information Technologies and Services department Assistant Manager Erhan Şahin at Atılım University for his support during the study.

TABLE OF CONTENTS

APPENDIX

LIST OF FIGURES

FIGURES

LIST OF TABLES

TABLE

LIST OF ABBREVIATIONS

DB	-	Database
DBMS	-	Database Management System
DSS	-	Decision Support System
DTS	-	Data Transformation Services
DW	-	Data Warehouse
ER	-	Entity Relationship
ERD	-	Entity Relationship Diagram
ETL	-	Extracting, Transforming, Loading
HTML	-	Hypertext Markup Language
MOLAP	-	Multidimensional OLAP
OLAP	-	Online Analytical Processing
OLTP	-	Online Transaction Processing
PHP	-	PHP Hypertext Preprocessor
RDBMS	-	Relational Database Management System
SQL	-	Structured Query Language

CHAPTER 1

INTRODUCTION

Universities accumulate large volumes of data which can be used to obtain information on the development and performance of academic activities. The information that can be obtained from academic databases will serve to look for answers to queries such as: a) Estimate the total number of academic staff in all programs in a range of years. b) Compute success ratios of students according to their faculties and departments. c) Find the number of academic personnel according to their titles in a specified year and program. d) Compute the total number of credits an instructor gives in a specified program and year. Automatic data mining techniques can be applied to obtain answers to such queries and to facilitate the development of strategies for improving the academic processes and the educational programs.

The aim of this study is to develop a Decision Support System (DSS) which targets to support the administrative task of planning the university's educational capacity in terms of the number of students that the offered courses can accommodate under the specified resource constraints. Goals also include, giving users access to data not included in the current information system (MasterSIS). This is done by creating a data warehouse by using some of University's current information system database. The Student Data Warehouse contains a subset of the student information which is stored in the University's Student Information System. The tables holds student personal information, academic history, courses and so on. A graphical query tool, Oracle Reports Developer Tool is distributed as part of the Data Warehouse. Oracle Reports Developer Tool enables users to perform

sophisticated queries on the warehouse, and formats the results as reports or html pages to be exported to other applications.

The focus of this thesis also includes the overall architecture and design of the system, the currently[1] used model for extraction, transformation and loading of data into data warehouse, and presenting data. The developed system's data warehouse is a central, logical site that stores many data models, supports central management's priorities and complements the university's administration needs. The system incorporates a broad scope of tasks, such as:

- Basic concepts are given about data warehousing architecture,
- Methods used for extracting data from legacy system, cleansing, transforming and preparing data for decision supports has been studied ,
- Accessing and analyzing data using tools has been accomplished,
- Data warehouse architecture for a University Information System is suggested and an example application is done in this study.

The main objective of this Master's thesis is to design a data warehouse for the Master Program at the Atılım University. This thesis arose from the need of a better source of information on which to base Graduate School of Natural and Applied Sciences' decision-making. Currently the student information system that will be defined as MasterSIS is used to obtain the required information. However, its support for management decision-making is limited.

Our objective is to restructure the data from MasterSIS into a data warehouse, which will make decision support easier. We also plan to design a user interface to be used specifically for decision support purposes especially for report analysis. The main users of the proposed system will be the Dean, Assistant Dean and decision makers from departments.

The thesis is organized as follows: Chapter 2 presents an overview of decision support system, enterprise systems, data warehouse concepts, and usage of data warehouse in education. Chapter 3 provides information about the research

[1] Atılım University Master Program Student Information database system (MasterSIS)

methodology of the thesis. Chapter 4 describes software development cycle of the study and a sample DSS model implemented for a university and lastly in Chapter 6 some conclusions are provided. Also some options for future work on the thesis are explored.

CHAPTER 2

LITERATURE REVIEW

In this chapter we reviewed the literature according to what a decision maker needs in order to make appropriate decisions. We also summarized the best way of holding data explaining different systems. Accordingly, this chapter covers the decision making and decision support systems, the enterprise systems and data warehouse.

2.1 Decision Making and Decision Support Systems

2.1.1 Decision Making

Decision making is an indispensable part of everyday life. We make hundreds of decisions each day. In order to make a good decision, we have to be informed about alternative options. These options can be in different forms such as numbers, graphics, and impressions. Turban classified decisions into three groups as structured, semi-structured and unstructured [1]. Structured decisions are repetitive and routine decisions. Unstructured decisions are non-routine decisions so that decision maker has to provide judgment. Semi-structured decisions include some characteristics of both structured and unstructured decisions. Decision maker has to provide judgment only for the parts that do not have an accepted procedure.

2.1.2 Managerial Decision Making

In an organization, the manager is the first and foremost decision maker. This is the same in higher education. President, Vice President, Dean and Department

Heads are the ones who make decisions. All managerial activities revolve around decision-making. However, the rapid change in the management environment result in the decision making process being more complicated today than in the past. The reasons for that can be summarized as follows:

- Making errors can be the cause of very big costs because of complexity in an organization
- It is difficult to access information in order to make decision
- There are lots of alternatives because of improved technology

Therefore decision maker has to make decision quickly especially in higher education [1].

2.1.3 Decision Support Systems

DSS exist to help people make decisions. DSS do not make decisions by themselves. They attempt to automate several tasks of the decision-making process of which the modeling is the core. To comprehend DSS, a person needs to understand the process of making decisions.

DSS uses data, provides easy user interface and can incorporate the decision maker's own insights. The tables in a decision-support database are heavily indexed, and the raw data is often preprocessed and organized to support the various types of queries to be used.

Purpose of Decision Support System

Holsapple and Whinston [2] stated that, the purpose of decision support systems is to improve the decision making ability of managers by allowing more or better decisions within the constraints of cognitive, time, and economic limits. More specifically, the purposes of a decision support system are:

- Supplementing the decision maker
- Allowing better intelligence, design, or choice

- Facilitating problem solving
- Providing aid for non structured decisions
- Managing knowledge

Supplementing the decision maker means that DSS should supplement one or more of a decision maker's abilities. DSS should help problem solving and make it more easy, smooth and fast.

Components of Decision Support Systems

A DSS is composed of four fundamental subsystems: Data management, model management, user interface and knowledge management subsystems.

- **Data Management System:** A DSS uses one or more data stores to provide relevant information to the decision support system. Some of them are maintained by the DSS itself and some are external data sources. Some database are primarily used and maintained by another information system with its own database management system and some DSS applications may have no separate DSS database. The data is entered into the DSS as needed.
- **Model Management System:** The model base gives decision makers access to a variety of models and assist them in decision making. It can include the model management system software that coordinates the use of models in a DSS.
- **Dialogue Subsystem or User Interface:** It allows users to interact with the DSS to obtain information. The user supplies information to the DSS and commands the DSS using this subsystem.

In addition, the user is considered as part of the system. The user interface is the hardware and software that facilitate communication and interaction between the user and the computer.

- **Knowledge Management Subsystem:** This is an optional subsystem and can support any of the other subsystems or act as an independent component. Also it provides knowledge for the solution of the specific problem [3].

Breiter and Light state that, [4-6] most theories of information management draw distinctions among data, information, and knowledge and there are three phases of process of transforming data into knowledge. It begins with raw data and ends with meaningful knowledge that is used to make decisions. They are the following:

Data exists in raw state. They do not have meaning itself, and therefore, can exist in any form. It changes according to person looking at it and turns into information.

- Information is data that is given meaning when connected to a context.
- Knowledge is the collection of information considered to be useful, and used to guide action.

According to Ackoff, [7] there are six steps in the process of data transformation into knowledge. In Figure 2.1, the process of transforming data into knowledge can be seen.

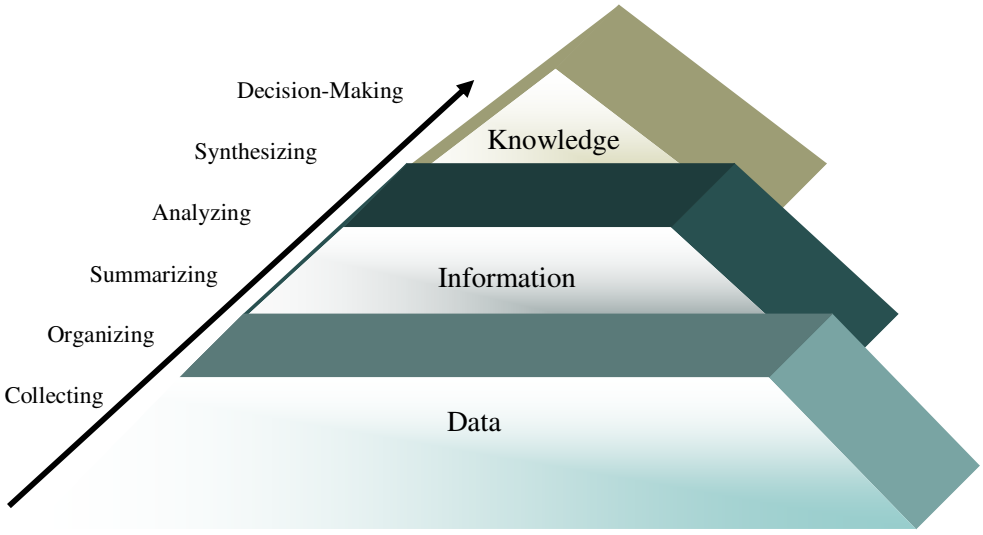

Figure 2.1 The process of transforming data into knowledge

The transformation process starts with collecting and organizing data, along with summarizing, analyzing, and synthesizing information to make decision. At the end of this process, the raw data becomes meaningful. Apart from these subsystems, according to Sauter, [8] there is a new subsystem referred to as message management subsystem, which is used to manage the data captured from electronic mail and

electronic discussion groups. A schematic view of a DSS and its components is shown in Figure 2.2 [7].

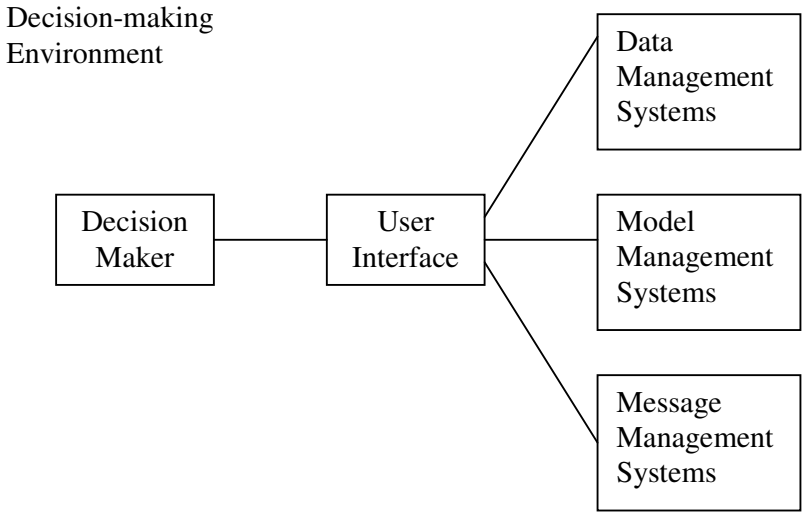

Figure 2.2 Components of DSS

The users are not changing data, the data is changed only by periodic, bulk updates made during off-hour, low-traffic times in the database. In this study, the DSS is designed to help administrators, faculty and staff with data and information for;

- Program review reports
- Campus and school planning projects
- Grants proposals
- Faculty and classified position requests
- Scheduling of classes and proposals to either expand or contract current class offerings.

2.2 Enterprise Systems

In education, financial systems, human resources management systems, student information systems, library systems, and other information systems provide the foundation of the higher education enterprise systems. The information that guides decision making in higher education derives from these information systems. Davenport [9] defined enterprise systems as packages of computer applications that

support many aspects of an organization's information needs. Different systems form an enterprise system in an organization. The information gathered from this system help a decision maker to make decisions. Figure 2.3 shows Atılım University's Enterprise system schema.

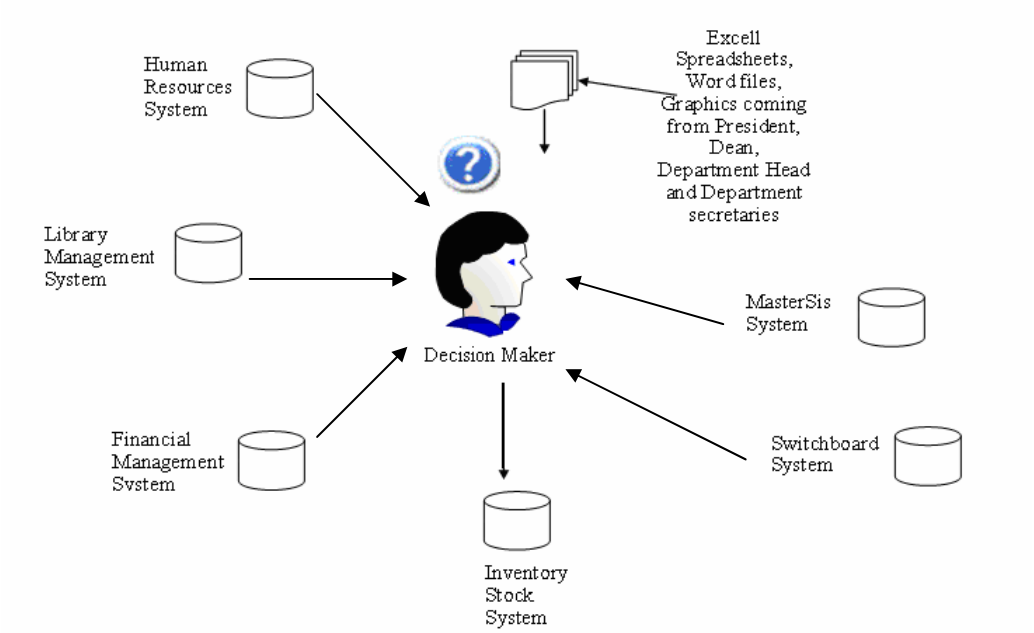

Figure 2.3 Atılım University Enterprise System Schema

2.2.1 Types of Enterprise Systems

There are many types of systems in a large Enterprise system [10].

- **Transaction System** - Users enter data into a transaction-oriented system and the results are visible immediately to all other users of the system. These are called Online Transaction Processing (OLTP) systems.
- **Reporting System** - Users need to view Enterprise Data textually and graphically based on user-defined criteria. These are; Online Analytical Processing (OLAP), Executive Information System (EIS), Decision Support System (DSS), Data Warehouse, Data Mart systems.
- **Batch System -** Batch systems process large amounts of data on a scheduled basis. They are part of Reporting Systems or Transaction Systems.

- **Real-time or Near-Real-Time System -** Real-time systems are usually used to control, monitor, or communicate with equipment that operates in real-time.

  Near-real-time subsystems are often a part of larger systems which include Reporting Systems or Transaction Systems.

2.2.2 Enterprise Systems and Decision Making

In enterprise systems, there are different interfaces, different data, and different data representations, duplicate and a lot of inconsistent information. The same data can be found in many different systems such as, an instructor information can exist in four different systems like library, student information system, finance and human resources systems. Therefore, combining the information and generating reports or making decision is very difficult in such enterprise systems [10].

In order to solve this complexity and help the decision-maker, integrated systems are offered as a solution. Figure 2.4 shows an example of an integrated system.

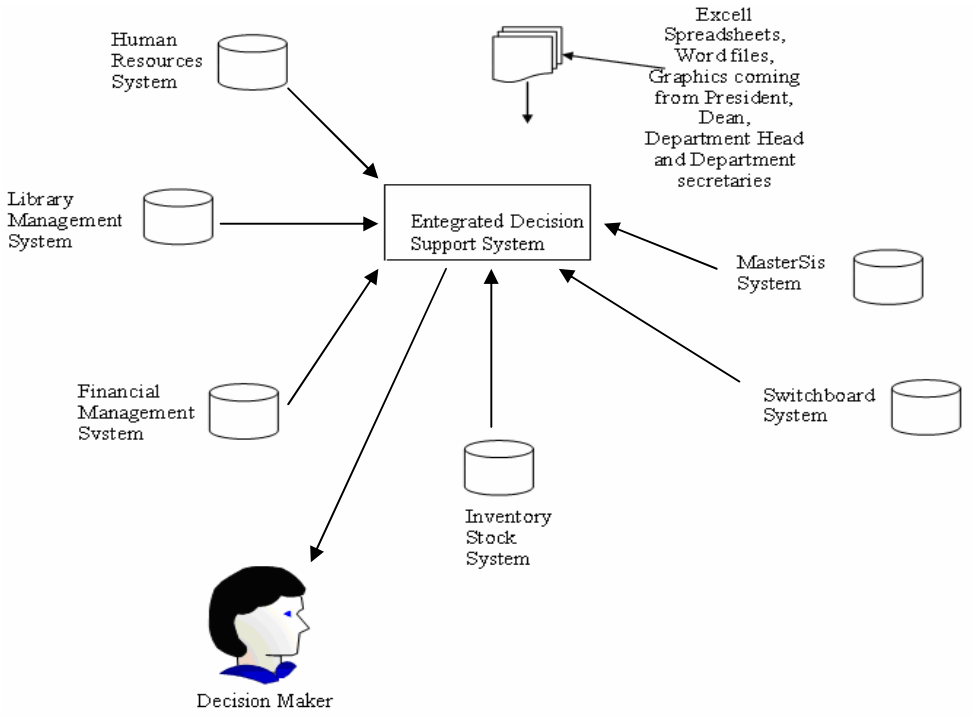

Figure 2.4 Integrated System

Integrated systems collect and combine information coming from different sources, provides integrated view to the user, a uniform user interface and supports data sharing among different systems [10].

2.3 Data Warehouse

In a large enterprise-wide database system, it is difficult to analyze the data since the data is stored in different formats and locations. To solve the integration problem of the enterprise systems, data warehouses are used as an integrated system. Educational institutions measures success very differently from business organizations therefore it is needed to develop a good data warehouse for academic institutions as well.

2.3.1 Definition of a Data Warehouse (DW)

Over the last few years, institutions 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. Today many organizations are trying to develop a DW to meet their business requirements.

Han and Kamber, define a DW as "a repository of information collected from multiple sources, stored under a unified scheme, and which usually resides at a single site" [11 p.1]. In educational terms, all past information available in electronic format about a school or district such as budget, payroll, student achievement and demographics are stored in one location.

According to Inmon, "a DW is a **subject-oriented, integrated, time-variant,** and **non-volatile** collection of data in support of management's decision making process" [12 p.1].

Inmon and Hackathorn stated that [16 p.285]:

“A DW is a separate store of data extracted from one or more production databases to produce an authoritative source for decision support. It can be considered a decision support system. Reporting data is another important aspect of data warehousing because the main output from DW systems are either queries with minimal formatting or formal reports.”

We can summarize some features of DW as follows [13-16];

Subject-oriented: DWs are designed to help one analyze data. Usually, this type of data is not suitable for decision-makers to use. Decision-makers need subject-oriented data. For example, to learn more about university students success ratios, one can build a warehouse that concentrates on grades. Using this warehouse, one can answer questions like "How many Computer Engineering students passed from “Compe523” course last fall?" This ability to define a DW by subject matter, grades in this case makes the DW subject oriented.

Integrated: Integration is related to subject orientation. DW must put data from disparate sources (like excel sheets, relational database) into a consistent format. They must resolve such problems as naming conflicts and inconsistencies among units of measure. When they achieve this, they are said to be integrated.

Nonvolatile: Nonvolatile means that, once entered into the warehouse, data should not change. There is no update of data as in an Online Transaction Processing (OLTP) database, and does not require transaction processing, recovery and concurrency control mechanisms. DW requires only two operations in data accessing: loading and access of data. This is logical because the purpose of a warehouse is to enable one to analyze what has occurred. That is why it does not require transaction processing, recovery and control mechanisms.

Time-Variant: In order to discover trends in business, analysts need large amounts of data. This is in contrast with OLTP systems, where performance requirements demand that historical data be moved to an archive. A DW's focus on change over

time is what is meant by the term time variant. Operational database contains current value data. The key structure of operational data may or may not contain some element of time.

2.3.2 Transaction Processing versus Decision Support Databases

There are two main types of database systems: Operational systems and analytical systems that include DWs. They differ in using data. Operational systems are referred to as online transaction processing systems and designed for real-time data entry and editing. One account, one student record, one inventory item, or one order can be updated or deleted in any time. Transactions are generally predefined and require the database to provide fast access. Analytical systems, such as DWs, are designed to help an organization with decision-making. A DW is updated periodically and its solutions offer query capabilities, trend analysis, and reporting.

2.3.2.1 Online Analytical Processing (OLAP)

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. “How many blue sweaters were sold via mail-order in Ankara so far this year?” or in a school database “How many students pass from Compe523 lesson this year?” are kinds of questions that OLAP is very good at answering.

There are several advantages of OLAP systems. Its simple design is easy to understand. Queries return consistent results. Data can be separated and combined by means of every possible measure in the business and users have access to business descriptions. In terms of the technology, an OLAP database can be implemented on top of an existing relational database (called Relational OLAP, ROLAP) or it can be implemented via a specialized data store (called MOLAP). In ROLAP, the data request is translated into SQL and the relational database is queried for the answer. In MOLAP, the specialized data store is preloaded with the answers to all possible queries so that any request for data can be returned quickly [17].

2.3.2.2 Online Transaction Processing (OLTP)

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 [18]. 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. It reduces performance cost of redundant data, supports high transaction rates and captures primary business production information. Besides its advantages, there are also some problems. These can be itemized as follows; [18]

- Only "pre-canned" reports could be generated on a scheduled basis
- ad-hoc real-time querying was virtually impossible
- Virtually no usage of historical data
- With a normalized structure, we have to query from multiple tables to get the detailed information.

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.

2.3.3 Data Mart

Data warehousing is a large, expensive solution and generally used for big organizations. However a data mart is similar to a DW according to structure and purpose but it is limited to a departmental or specific subject need. Thomas Connolly and Carolyn Begg [14] defined a data mart as a subset of the data in a DW and it

summarized the data relating to a department or a specific function. A schema showing the data marts and DW is shown in Figure 2.5.

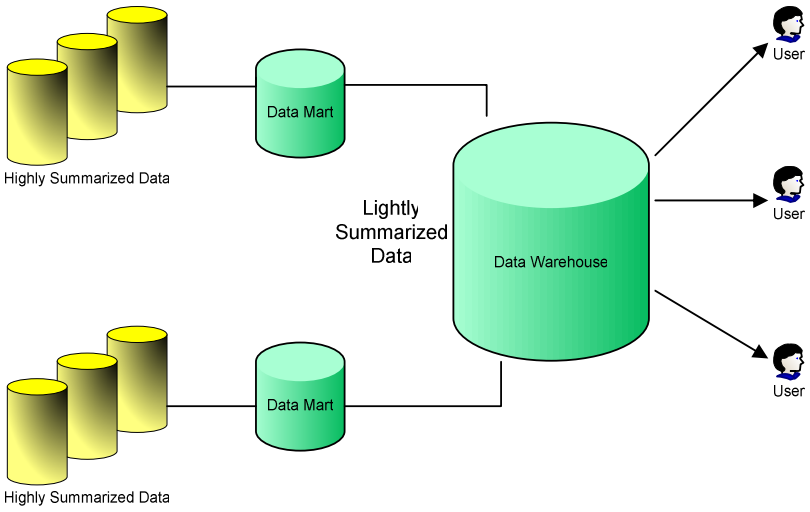

Figure 2.5 Data Marts and DW

Data marts contain less data than DWs as they serve to a specific purpose. The followings are advantages of using a data mart rather than a DW [15].

- Data marts contain less data, therefore query response times are shorter compared to DW.
- Data integration and transformation process of data takes shorter time.
- Data marts are more simplified than DWs, so building takes less time.
- Besides the advantages, there are some disadvantages of using a data mart.
- Size of a data mart is smaller than a DW as it answers small organization needs. If the size will be increased, its performance will decrease.

2.3.4 Structure of the DW

Data contained in a DW contains five types of data: data currency, existing data, data summarization (lightly and highly summarized data), and Metadata. The following Figure 2.6 shows the structure of the DW [15].

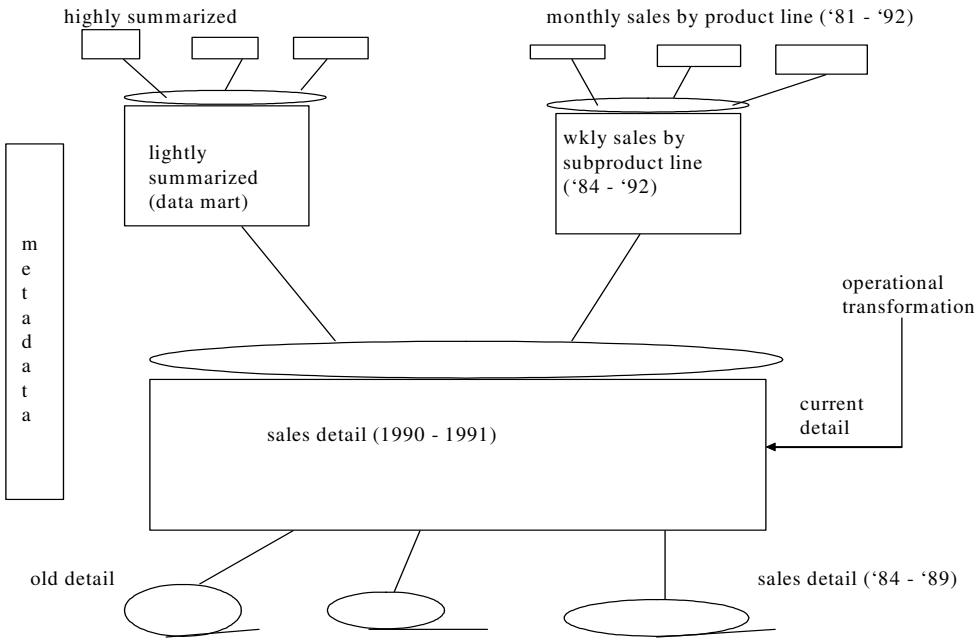

Figure 2.6 Structure of the DW

Metadata

Tannenbaum [19] defined metadata as; the detailed description of the instance data; the format and characteristics of populated instance data; instances and values depend on the role of the metadata recipient. Examples of metadata include filenames, data element definitions, data element names, lengths, and program names and so on.

An important component of the DW environment is metadata. Metadata or data about data is a critical component in a warehouse environment. Metadata has many uses. Users need to know what data is available, how it is organized and what it means. It allows the end user to navigate through the possibilities. In other words, when a user approaches a DW where there is no metadata, the user does not know where to begin the analysis. Metadata typically contains [14]:

- Structure of data as known to the programmer and to the DSS analyst
- Source data
- Transformation of data
- Data model

- DW
- History of extracts

Data Currency

Current detailed data is at the heart of the DW and reflects the most recent happenings. This data is stored according to a data model. Older detailed data accumulates over time as fresh data enters the application. Generally, in most DSS, data obtained from detailed records on transactions are cleaned and then loaded [20].

Data Summarization

Data is lightly summarized and stored in the DW. The data might be summarized over different units of time and different attributes. Data could also be found in a highly summarized form for quick retrieval. The highly summarized data could be physically stored within or outside the DW but is logically considered a part of the DW [21].

2.3.5 DW Modeling Techniques

DW modeling is the process of building a model for the data in order to store in the DW. This chapter provides a basic understanding of data modeling, specifically for the purpose of implementing a DW. There are two data modeling techniques that are relevant in a data warehousing environment are Entity Relationship (ER) modeling and dimensional modeling.

ER Modeling and Dimensional Fact Modeling

ER modeling produces a data model of the specific area of interest, using two basic concepts: entities and the relationships between those entities. Detailed ER models also contain attributes, which can be properties of either the entities or the relationships. The ER model is an abstraction tool because it can be used to understand and simplify the ambiguous data relationships in the business world and complex systems environments.

Dimensional modeling uses three basic concepts: measures, facts, and dimensions. Dimensional modeling is powerful in representing the requirements of the business user in the context of database tables. Measures are numeric values that are can be added and calculated. ER modeling uses the following concepts: entities, attributes and the relationships between entities. The ER model can be used to understand and simplify the ambiguous data relationships in the business world and complex systems environments. An entity is represented on the diagram as a rectangular box. The name of the entity appears in the top section of the rectangle, such as Instructor and Department in the below Figure 2.7. Attributes are listed in the lower part of the rectangle that represents the entity which they belong to. A relationship is a line between the boxes that represents the entities that it links. The relationship may be identified in different ways like; one-to-one, many-to many, and one to many.

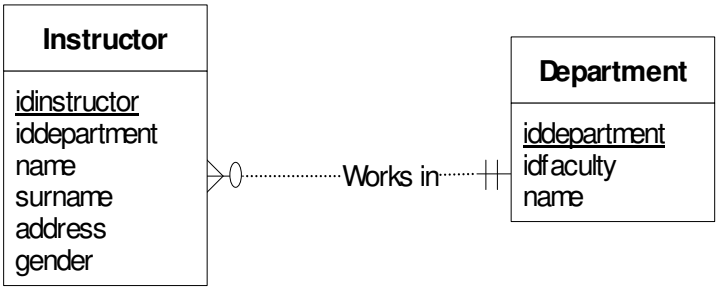

Figure 2.7 Entity Relationship Diagrams

In some respects, dimensional modeling is simpler, more expressive, and easier to understand than ER modeling. Dimensional modeling uses three basic concepts: measures, facts, and dimensions. Dimensional modeling is powerful in representing the requirements of the business user in the context of database tables [22]. The dimensional model is a logical design technique that seeks to make data available to the user in an intuitive framework that is intended to facilitate querying [23]. Dimensional model is composed of fact tables and dimensional tables where fact tables are normalized tables that represent the process being tracked. It is a logical design technique, used in Data Warehousing databases, which is an alternative to Entity-Relationship (ER) modeling [24]. Figure 2.8 represents the dimension tables and fact table with primary keys and foreign keys.

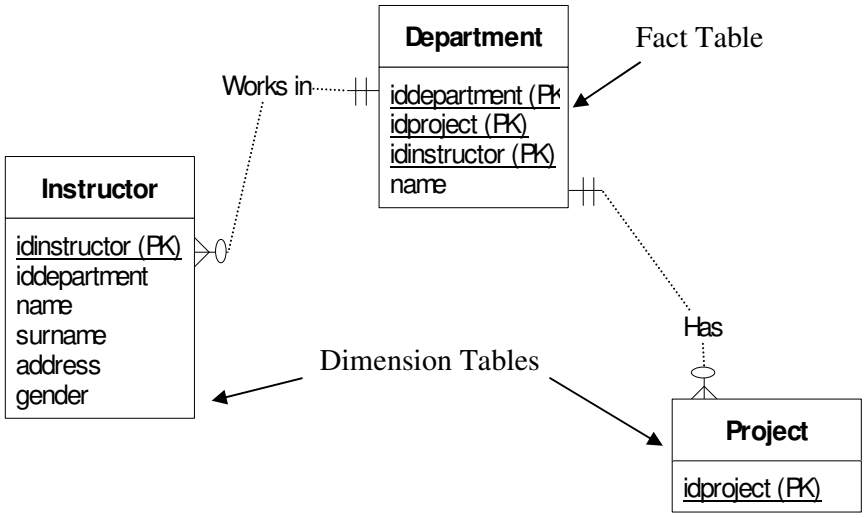

Figure 2.8 Dimension Table and Fact Table with Primary and Foreign Keys

Each dimension table is assigned a primary key that are not related to one another. The primary key is unique key for the dimension table, which is replicated in a fact table where it is referred to as a foreign key. A fact table is a table that contains the data (factual history) like Student Id, Course Id, Semester Id, etc.

Connolly and Begg derived the way a DW or a data mart structure in dimensional modeling into several ways [14]; Flat schema, Terraced Schema, Star Schema, Fact Constellation Schema, Galaxy Schema, Snowflake Schema, Star Cluster Schema, and Starflake Schema. However there are two basic models that widely used in dimensional modeling: Star model and Snowflake model.

Star Schema

The schema associated with a dimensional model is referred to as a star schema. It is widely used because of its simplicity. It is called a star schema as its shape resembles a star. A star schema consists of a central fact table joined with several dimension tables. The fact table is large and has all the transaction data and numbers, the dimension tables are smaller and have descriptive information. In Figure 2.9, you can see fact and dimension tables. Instr_Credit table is the fact table and Term, Instructor, Department and course tables are dimension tables.

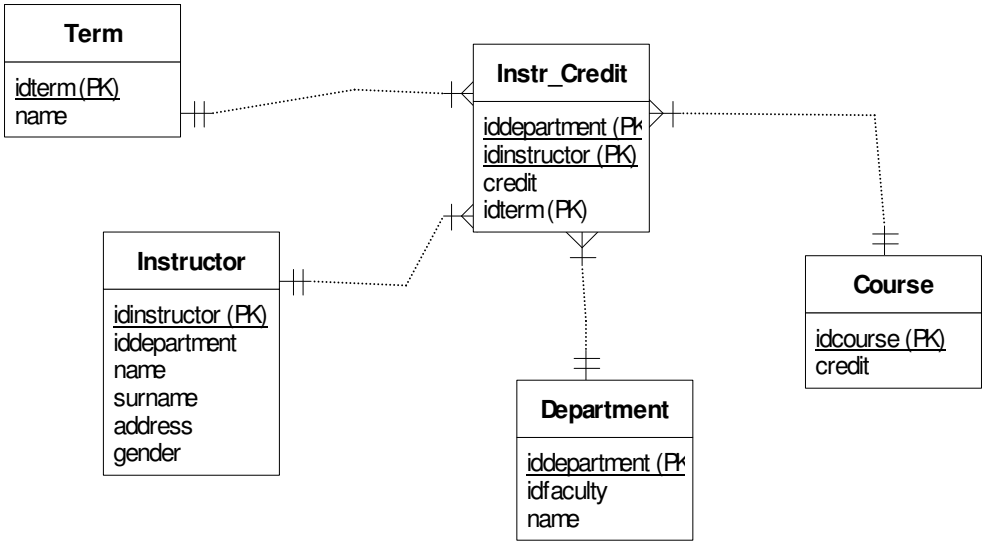

Figure 2.9 Dimension and Fact tables in Star schema

Snowflake Schema

Snowflake schema model is derived from the star schema and, as can be seen, looks like a snowflake.The snowflake model is the result of decomposing one or more of the dimensions, which generally have hierarchies between themselves. Many-to-one relationships among members within a dimension table can be defined as a separate dimension table, forming a hierarchy. For example in the Figure 2.10, a starflake schema is given.

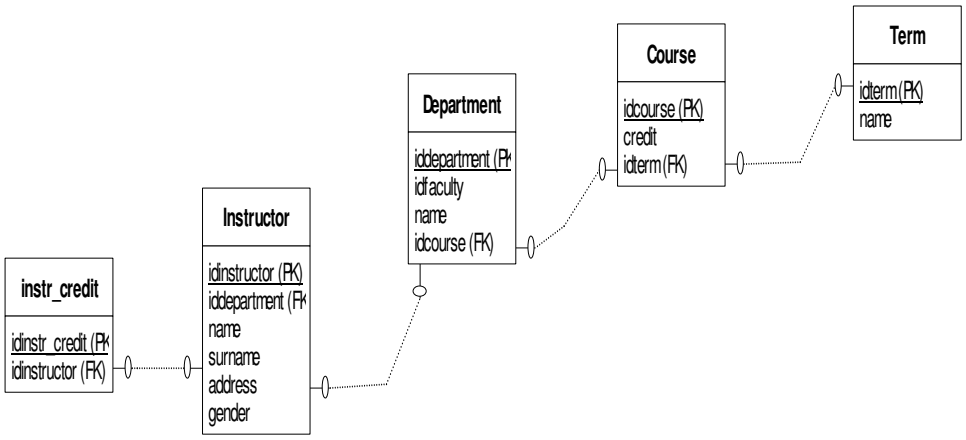

Figure 2.10 An Example of a Snowflake Schema

2.3.5.1 Benefits of Data Warehousing

There are several benefits of data warehousing. The most important ones are listed below.

- DW improves access to administrative information for decision makers.
- It can get data quickly and easily perform analysis. One can work with better information, make decisions based on data. DW increases productivity of corporate decision-makers.
- Data extraction from its original data sources into the central area resolves the performance problem, which arises from performing complex analyses on operational data.
- Data in the warehouse is stored in specialized form, called a multidimensional database. This form makes data querying efficient and fast.
- A huge amount of data is usually collected in the DW. Compared with relational databases that are still very popular today, data in the warehouse does not need to be in normalized form. In fact, it is usually de-normalized to support faster data retrieval.

2.3.5.2 DW Modeling Tools

Building a DW from independent data sources is a difficult process. This process involves extracting, converting, cleaning, integration and transformation of the data. In order to do these operations, an ETL (extract, transform, and load) tool is required. The key steps that need to be undertaken to transform raw operational data to a form that can be stored in a DW for analysis are: [25]

- ***Extraction*** The goal of the data extraction step is to bring data from different sources into a database before modification.
- ***Converting*** the data into a format that is suitable to the DW.
- ***Cleaning*** of the data. Data entry errors and differences in schema formation can cause for example student dimension table to have several corresponding entries for a single student.

- ***Integration*** of the different datasets to suit the data model of the DW.
- ***Transformation*** of the data through summarization and creation of new attributes. It is a set of rules and scripts that typically handles the transformation of data from an input schema to the destination schema.

2.3.5.3 Data Analysis and Assessment Tools

When these steps have been completed the data is ready for analysis. There are several options to carry out these steps. The following products allow schools and districts to gather, analyze, and make the most effectiveness from their data [26].

Quella Business Intelligence Solution is an easy-to-use tool covering the Business Intelligence Process from Data Extraction to Analysis. As a total solution, Quella extracts transactional data from heterogeneous data sources, consolidates the extracted data and provides necessary business information to decision makers through List Reports, Charts and OLAP-Reports.

Cognos allows users to create individual education plans. Its web-based system lets multiple people access the forms simultaneously.

Oracle Warehouse Builder (OWB) is Oracle's comprehensive tool for ETL, fully integrated relational and dimensional modeling, data quality, data auditing, and full life cycle management of data and metadata.

SQL Manager Lite for Oracle Simplify and automate database development process, design, explore and maintain existing databases, build compound SQL query statements, manage database user rights and manipulate data in different ways.

Apple's PowerSchool is a web-based SIS that delivers real time information across the Internet. This means you can easily access your data anywhere, anytime via your web browser using your current Internet connection. PowerSchool stores all data in a single centralized relational database, and allows reporting.

Cancery's Student Management System is designed as a web-based Student Information System built on Microsoft .NET. This release gives districts and schools an open, centralized system for managing student information. System requires lower hardware, software, infrastructure, and personnel costs than decentralized solutions. It allows user to visualize data through graphs and charts and generate basic reports.

ERStudio a model-driven data architecture and database design solution, helps companies discover, document, and reuse data assets. With round-trip database support, data architects have the power to analyze existing data sources as well as design and implement high quality databases. The highly-readable visual format enhances communication across functions.

Sunguars's Performance PLUS provides data extraction, aggregation, and analytic tools.

Octopus is a simple Java-based Extraction, Transform, and Loading (ETL) tool. It may connect to any JDBC data sources and perform transformations defined in an XML file. A load job-generator is provided to generate Octopus load job skeletons from an existing database. Many different types of databases can be mixed (MSSQL, Oracle, DB2, QED, JDBC-ODBC with Excel and Access, MySQL, CSV-files, XML-files,...) Three special JDBC drivers come with Octopus to support JDBC access to CSV-files (CSV-JDBC), MS-SQL (FreeTDS) and XML. Octopus supports Ant and JUnit to create a database / tables and extract/load data during a build or test process.

BrioQuery is a sophisticated ad hoc query and reporting application for client/server database environments. BrioQuery provides non-technical users with easy access to the valuable data available in the university's Sybase data extracts.

2.4 Data Warehouse 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 [12].

Data warehousing collects and organizes data from multiple sources so that it can then be easily analyzed, extracted and used. Because of this, the DW is a core component for enabling decision making for educational institutions. The use of a DW presents several potential advantages for educational institutions, including timely access and to better evaluate the data. Institutions can easily and quickly identify troublesome trends in its enrollment and evaluate why they are occurring. They can determine strengths and weaknesses within themselves. The use of a DW allows an educational institution to use this information in making appropriate decisions, and that is the desired end goal of the data warehousing process [27]. By using a DW, educational institutions could overcome several problems. Not only do they set operational data free from complex retrieval, but also they benefit from many other issues, concerning the management. For example, consider these few questions, which could be answered easily, using a DW in a higher education institution:

- Total number of academic staff in all programs according to their titles in a range of years
- Total number of students in faculty of engineering according to their programs in a range of years
- An estimate of the total number of students that will take the course.
- Letter grades of a specific course according to different years
- Success ratios of faculties and departments in a range of years

- The number of female/male students registered in the university according to program and year.
- Total number of students in a faculty or program and total number of academic staff according to their titles in a range of years
- Total number of part-time and full-time instructor list according to their titles.
- Total number of credits an academic staff gives in a range of years.

Note that these questions above are not very easy to answer just by using a relational database. The main reason behind this problem is a simple fact namely, "relational databases are good at retrieving small number of records quickly, but they do not retrieve a large number of records and summarize them on the fly" [28].

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.

Breiter and Light stated that [29]; school information systems constitute a clear sub-group of management information systems that are used in educational organizations. In school, distinct information systems support different types of decisions: administrative information systems, learning management systems and assessment information systems. Figure 2.11 illustrates the school information systems topology.

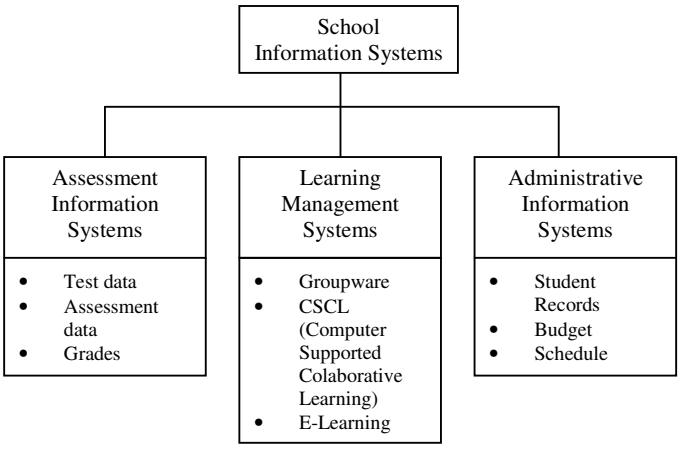

Figure 2.11 Topology of school information systems

Administrative information systems consist of financial, human resources, student grades, course, student fees, research, student enrollment, faculty workload and administrative areas. Learning Management System (LMS) supports learning process. LMS is software that automates the administration of training events.

According to Remes [30], all LMS's manage the log-in and registration of users, manage course catalogs, record data from learners, and provide reports to management. Assessment information systems get data about the student online test performance. In an effort to contribute to the growth of the higher education Data Warehousing industry and to see the different usages of data warehousing, different higher education institutions are examined from literature. From the literature survey, we have understood that, higher educational institutions are using DW solutions for administrative and assessment information purposes only. Research about learning management systems were not encountered. The following sections summarize these studies.

2.4.1 Administrative DW Solutions

In order to allow university decision-makers to evaluate various programs and academic departments, Baylor University started using a DW and decision support system [31]. The main goal was to bring data together for administrative process that let administrators and decision makers answer questions and make better decisions. Other goals included giving university decision makers the ability to create ad-hoc queries, generate reports, and perform analyses without having to request work from the university's IT department. To do this, they used SAS System as a database and SAS MDDB Report viewer as a Query tool. They developed an internet-based system. They have reported that, by giving user direct access to information the organization saved the time of their staff and helped to make quick and effective decisions.

Manitoba University [32] has a number of legacy systems that are used to support the day to day operations of the university. The disadvantages were numerous to the university in relying on operational systems for data in support of management decision making. Accessing different systems to get different kinds of

information and then trying to rationalize them is both difficult and time consuming. They thought that data warehousing would increase their ability to respond and their productivity. In this university, the clients using the system from outside could not make their own analyses because the hierarchical storage structure of the data did not allow for ad-hoc access. In order to extract the data, high level programming was needed, and there was no easy way to get information into desktop tools. These situations made the development of a warehouse a priority for the university. Their data model defines subject areas, such as ***Students, Employees, Courses, Space,*** and ***Alumni***. They designed the DW on a Sybase Server.

MIT'S DW has been in existence for about 5 years and they have Personnel, Financial, Purchasing, Telephone, Student, and Award information in the warehouse [33]. The main purpose of the warehouse was a reporting and data distribution environment for departments, labs and centers at the MIT. Besides, their DW is acting as a hub, to facilitate the exchange of information between systems. They are using BrioQuery tool as an end user query and reporting tool. Developers designed the DW in a way that users can easily understand and use. They have organized the information in star schemas and they gave direct SQL access to the DW in order to give flexibility to the users.

2.4.2 Assessment DW Solutions

Ingham reports that [34], Polytechnic University, using an integrated student information system as a main source of data developed a DW because of the difficulty of retrieving information in such enterprise system. They developed their DW for assessment process in order to get student test scores and alumni data as a report. They stored the institutional data in a relational database format and used ten year data purged from existing system. The system serves to the department heads and administrators.

2.4.3 Data Warehousing Studies in Turkey

Under the direction of the researches done, since data warehousing is a new field, no study related with data warehousing has been encountered in higher

education in Turkey except a study done in Eastern Meditterranean University by Deniz and Ersan [35]. They designed a software package called PADSS for academic purposes in order to analyze student performance data for academic decision making.

In Turkey, data warehousing studies are generally done for government institutions. Researches show that, DW projects are in the development stage in recent years. For example, a DW system is developed for the Republic of Turkey Prime Ministry Under secretariat of Customs in order to help the decision maker in decision process.

Rationale for this study was the lack of high-quality data available for decision making purposes to administrators in the Atılım University. University administrators who use poor quality data to assist them in making their decisions aimed to be bedeviled by making incorrect judgments. For example, an academic vice-president using one-year old student credit hour production data for faculty needs projection, may not hire the correct number of faculty for each department in the coming year. A vice-president of student affairs using incorrect admissions data may not have enough housing available for his or her new students. Finally, a financial vice-president working with redundant budget data may be forced to change multiple data elements each time a budget change occurs.

To solve these problems in this study, we have developed a DW based on Master of Science program of the university. This study describes how the DW is modeled and developed.

CHAPTER 3

RESEARCH METHODOLOGY

This is a single case study done at one of the higher education institutions (Atılım University) of Turkey. At this university, there is an information system called MasterSIS which provides services for current needs of the master program of the university. However, there is no provided solution for the decision making processes yet. Accordingly, this study aims to develop a decision support tool based on data warehousing approach.

This study examined some of the data from Computer Engineering, Manufacturing Engineering, Mechatronics Engineering, Software Engineering and Civil Engineering master programs in the Graduate School of Natural and Applied Sciences of the Atılım University. The university is currently using an information system which is developed by the Atılım University Information Technologies and Services Department. The possible users of the system to be developed are the administrative and academic personnel. The reports are generated according to 240 numbers of students, 65 academic staff, and 204 courses in the system. PostgreSQL DBMS is used to store the data. The system tables are created using dimensional modeling as it was mentioned in Section 2.3.5. The following steps have been followed in this case study.

In the first phase, in order to better understand the problem at the Atılım University, interviews have been conducted with six people who have been involved in the decision making process of the university. The interviews are done with three administrative and three academic staff. The interview guide is given in the Appendix A.

Secondly, the review of the existing literature relevant to the study was completed. Areas of concentration included; decision support systems, enterprise systems, DW development techniques, design methodologies and use of DW in higher education. The goal of this literature review was to ensure that the most recent contributions to the relevant literature have been used in the preparation of this study. The dimensional model of the proposed system is designed, the SQL queries are written, the data is extracted, transformed and loaded into the system, reports are created and a user interface is developed.

Thirdly, after the development phase is completed, feedback from the users about the use of the developed system is collected by means of interviews. In this stage, interviews with three administrative and three academic staff are conducted. The main purpose of these interviews is to take user responses and their scale and satisfaction about the developed system. The interview guide is given in the Appendix B.

CHAPTER 4

LIFE CYCLE OF THE STUDY

The development of a DW for decision support goes through several phases that may be called decision support life cycle. These phases are given below.

4.1 Software Development Methodology

The Spiral Modeling approach is used in this study. Steps of the development methodology are summarized in Figure 4.1. The focus is primarily on data modeling requirements. Spiral development method has been evolving based on experience with various refinements of the waterfall model as applied to software projects.

The Spiral model reduces risk early by breaking down the project into smaller iterations. This causes part of the testing and design to occur earlier in the project, which in turn allows us more time to fix problems as they are discovered. This increases the probability that either the project will succeed or we will realize early that we cannot complete the project successfully within the allocated time and budget.

The starting point is to perform specific business data analysis process to identifying the DW project by a group of end user through solution integration all steps will be performed. Figure 4.1 shows Spiral Development Method cycle.

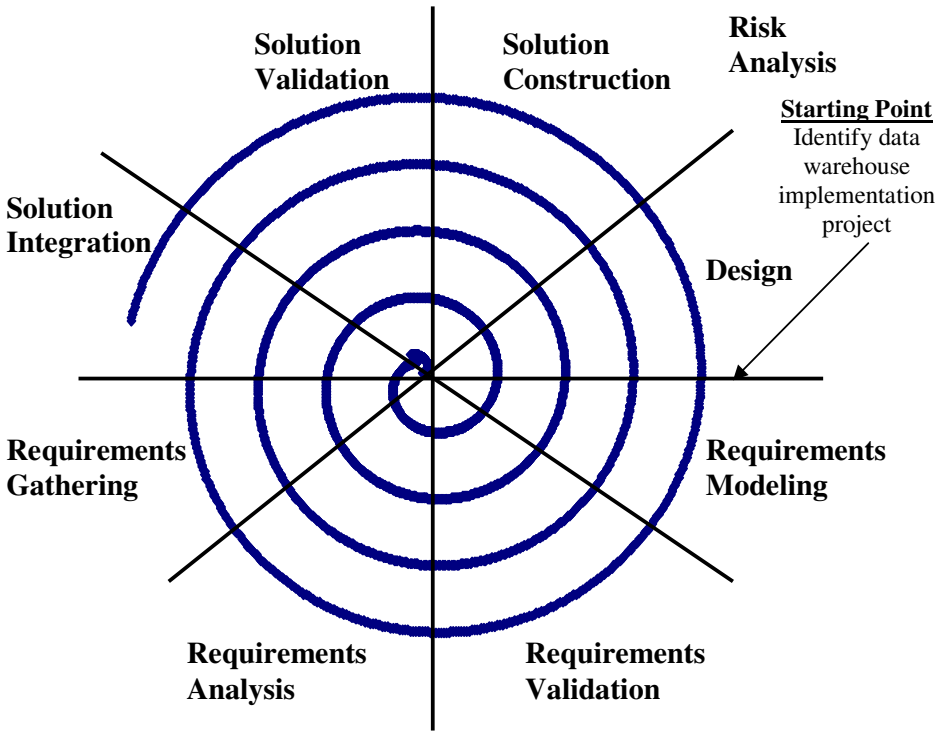

Figure 4.1 Spiral Development Method

4.1.1 Requirement Gathering

This phase of the lifecycle is concerned with the understanding of the business needs and data requirements of the users of the system. During requirements gathering, end-user requirements are collected and documented. Requirements gathering is often incorporated in some way into studies of the business process and information analysis activities in which end users are involved. Requirements gathering therefore are very much oriented toward understanding the problem domain for which the modeling will be done. Usually end-user requirements at this stage are documented rather informally or at least they are not represented in detailed schemas. According to Kimball [24], business users and their requirements impact almost every decision made throughout the implementation of a DW.

Gathering data requirements phase includes:

- Conducting user interviews
- Documenting results

4.1.2 Requirement Analysis

During requirement analysis, informal end-user requirements are further investigated, and initial dimensional models are produced showing facts, measures, dimension keys, and dimension hierarchies. Dimension hierarchies can include parallel hierarchical paths. Models produced during requirements analysis must be kept simple, because these initial dimensional models must be discussed with end users.

4.1.3 Requirements Validation

Initial dimensional models are used in this process of validating the end-user requirements and for assessing the scope and impact of the development project. These activities are schematically summarized in Figure 4.2 [36].

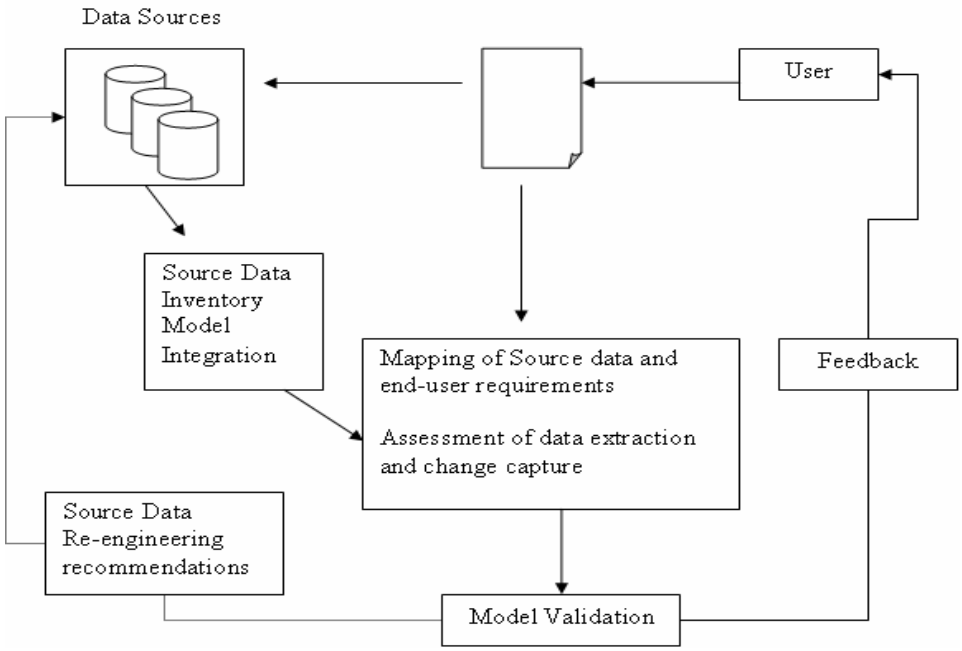

Figure 4.2 Requirements Validation

4.1.4 Requirements Modeling

In this phase of the cycle, validated user requirements are modeled. These models are further developed into detailed dimensional models, showing all elements of the model and their properties. Detailed dimensional models can further be extended and optimized. Many techniques in this area should be thought of as advanced modeling techniques.

4.1.5 Risk Analysis

The purpose of the risk analysis phase is to assess both technical and management risks. This step is not performed in this study.

4.1.6 Design, Construction, Validation, and Integration

Design and construction activities have to be performed after the requirements are modeled. This step will possibly extend and change the models produced in the previous stages. After that, validation of the proposed system has to be performed with the users of the system. According to user satisfaction the integration part is done. If any changes are needed they are added to the system.

With the direction of the spiral research methodology technique, the steps of the study are completed. In the following section, the research methodology is described.

4.2 Implementing a Student DW

4.2.1 Requirement Gathering

Requirement gathering process is done as the first stage as it was mentioned in Section 4.1.1. In order to accomplish a successful warehouse study, we must take into account user requirements from the people who are responsible for student data. We must discover the information required to be used for better decision-making. For that purpose, we carried out several interviews in order to determine what kind of

data must be available in our DW and how this data should be organized and manipulated to achieve our goal.

In particular we conducted interviews with three key people in administrative unit and three academic staff. The administrative staff in Student Affairs department provided us with documentation and sample reports from the current system. Upon examining the current system, we found that it suffers from the following problems in Section 4.2.2.

4.2.2 Requirement Analysis

After the requirement gathering phase is done, the problems are determined and classified according to the interview results. Accordingly the problems of the current system can be summarized as follows.

Problem 1

The current system does not give the number of students taking each course over a number of terms. In order to make resource allocation like how many sections need to be offered in a term and how many academic personnel are needed to offer the courses, it is important for the administrators to get this information.

Problem 2

Knowing the success ratios of departments according to years is important information for an administrator. It is not possible to reach this information in the currently used system. Especially, Student Affair Department gives several reports every year to the president of the university and to the supreme authority for the regulation of higher education; Council of Higher Education (YÖK), like “Activity Report” and “Inspection Report”. Currently, all of these reports’ outputs are calculated by human effort.

Problem 3

To see the changes in the number of registered students for a specified academic program and the changes in the number of academic staff in the same years

are important for an administrator in order to make resource planning. The current system is incapable of giving this information.

Problem 4

In the currently used system, it is not possible to see the academic staff's course loads and the total number of changes in academic staff according to their titles for a range of years.

Under the lights of these problems, a DSS was developed to answer a wide range of questions about Students, Master Program's Academic Personnel, and Courses. This is not a comprehensive list but represents the kinds of questions that can be answered using the developed DSS. The list can be expanded according to decision-makers needs.

- What is the total number of academic staff in The Graduate School of Natural and Applied Sciences?
- What is the total number of academic staff in The Graduate School of Natural and Applied Sciences group by titles?
- What is the total number of academic staff in all programs according to their titles in a range of years?
- What is the total number of students in The Graduate School of Natural and Applied Sciences in a range of years?
- What is the total number of students in The Graduate School of Natural and Applied Sciences according to their programs in a range of years?
- What is the total number of credits an academic staff gives in a range of years?
- What is the total number of part-time or full-time academic staff according to their titles?
- What is the number of female/male students according to program and year?
- How many students successfully complete courses?
- How many students complete a course with an A or B grade?
- How many students successfully complete Compe103 course?
- How many students fail courses?

- What are the success ratios of faculties and departments in a range of years?

In this step, the initial dimensional model that shows fact tables, dimension tables, measures, and dimension keys of the developed DSS is described. Table 4.1 shows this in detail.

Tablo 4.1 Tables of the system

Fact Tables	Dimension Tables	Primary Keys of the Fact Tables
DWPROGTITLE	donemders, donem, birim, program, yariyil unvan, hoca	iddonemders, idhoca, idunvan, iddonem, idprogram, idyariyil
DWINSTRCREDIT	donem, ders, yariyil, hoca, program, birim, donemders	iddonem, idders, idyariyil, idhoca, idprogram, idbirim, iddonemders
DWFAILEDSTUDENTS	donem, yariyil, dersnotu, derskayidi, donemtip, birim, program, donemders	iddonem, idyariyil, iddersnotu, idderskayidi, iddonemtip, idbirim, idprogram, iddonemders
DWGENDERBYPROGRAM	yariyil, donem, ogrenci, bolumogrenci, program	idyariyil, iddonem, idogrenci, idprogram, idbolumogrenci,

4.2.3 Validation

The main activities that are performed as part of the requirement validation are:

- The initial models are analyzed with the end users.
- Data sources are identified. In this study, there is only one available data source which is MasterSIS database.
- To conclude the requirement validation phase, we performed the sizing of our model. For initial loading, the database size estimation is as follows according the data taken from the university between years 2001-2005:
    1. donemders: There are 328 opened courses.
    2. donem: There are 13 terms.
    3. birim: There are 344 records in sections.
    4. program: There are 5 different master programs.

5. yariyil: There are 5 semesters from 2001 to 2006.
6. unvan: There are 6 different titles.
7. hoca: There are 65 instructors.
8. ders: There are 224 courses.
9. dersnotu: There are 1021 student marks.
10. derskayidi: There are 1021 course registers.
11. donemtip: There are 3 terms. Fall, Spring, and Summer
12. ogrenci: There are 258 students.
13. bolumogrenci: There are currently 258 records and the number increase every term.

Finally, we need approximately (Total Fact Table Size Estimation) 450 MB of space initially in the server. This number will increase at the end of every term because of the loading of the new term data.

4.2.4 Requirements Modeling

In this phase of the development cycle, user requirements are modeled. Database is planned to be stored on a Oracle 10g RDBMS. RDBMS is selected rather than a multidimensional database due to the scalability of RDBMS being better than multidimensional database. RDBMS uses SQL, which is the standard language for query and analysis. Multidimensional databases rely on a proprietary API for data access and analysis. In addition to these, RDBMS have a variety of features such as triggers, stored procedures that enhance the usefulness of the system.

In this study, the star schema is used to model the DW. **Microsoft Office Visio 2003** and **ER Assistant** tools are used while creating the schemas. This schema is chosen as it is much more understandable by the end users. Designs of the dimension tables are done as shown in Figure 4.3.

Figure 4.3 Dimension Tables

4.2.5 Design of the DW

After creating all dimensions in the design, the star schema declaration in detail is done in this phase. The figures are created using **ER Assistant** tool and the cardinalities of the relations are done using **Information Engineering Style**. As mentioned in Chapter 2, the center of the star schema contains a ***fact table***. Dimensional tables surround each fact table. Every dimensional table has its own primary key and connects to fact tables with these keys. Fact tables store primary keys of the dimension tables. In Figure 4.4 a model of the DW designed for Atılım University can be seen.

Figure 4.5 represents the first dimensional model of the project. The ***DWPROGTITLE Fact table*** is surrounded by dimensions tables, ***donem, unvan, hoca, section, donemders, donem, program,*** and ***yariyil.***

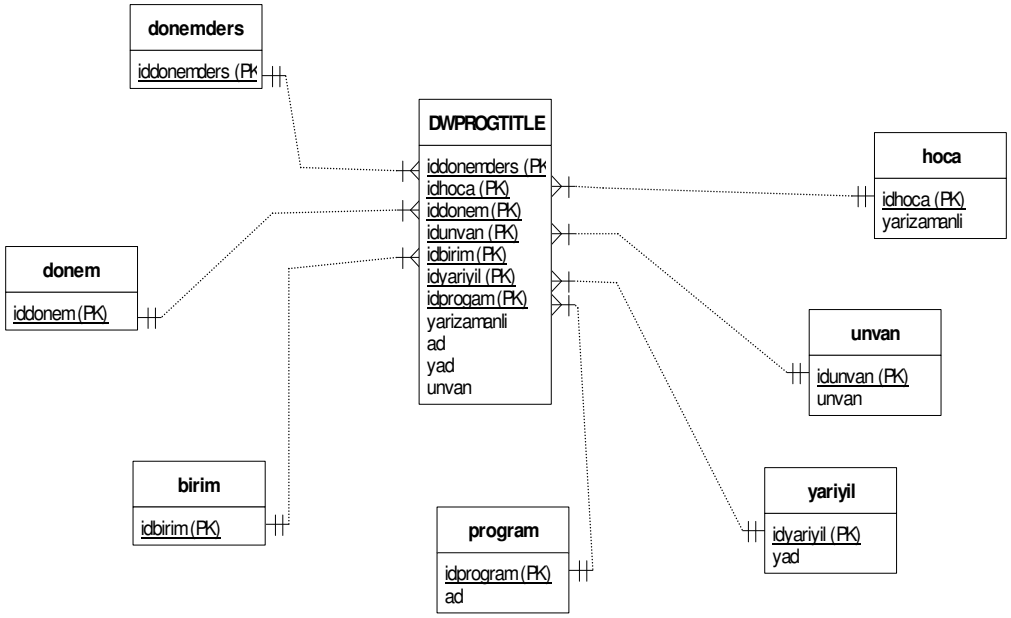

Figure 4.5 Dimensional Model for Instructors

Figure 4.6 represents the second dimensional model of the project. The ***DWINSTRCREDIT Fact table*** is surrounded by dimensions tables, ***donem, ders, yariyil, program, birim,*** and ***donemders.***

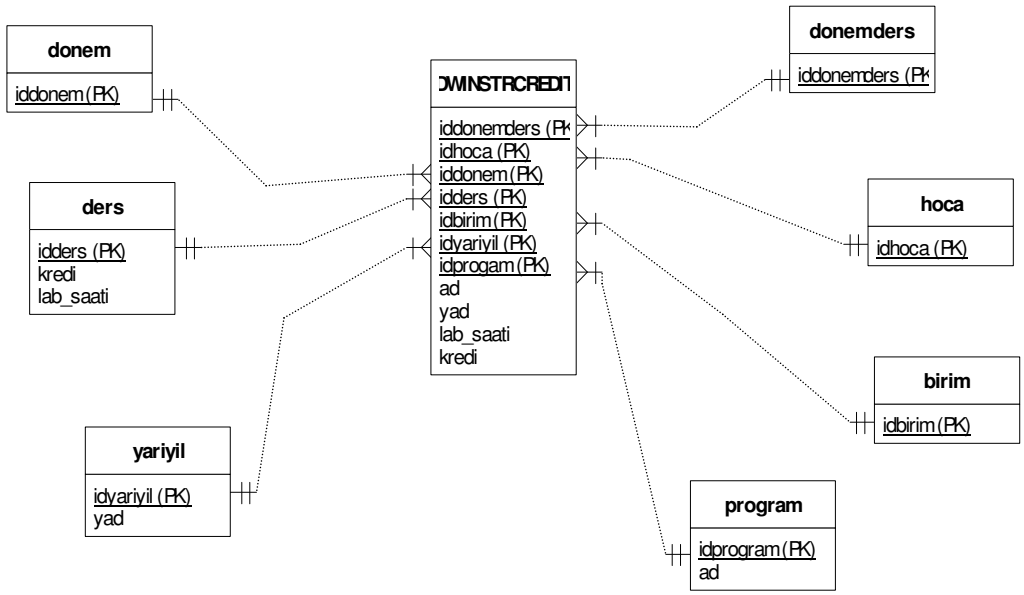

Figure 4.6 Dimensional Model for Courses

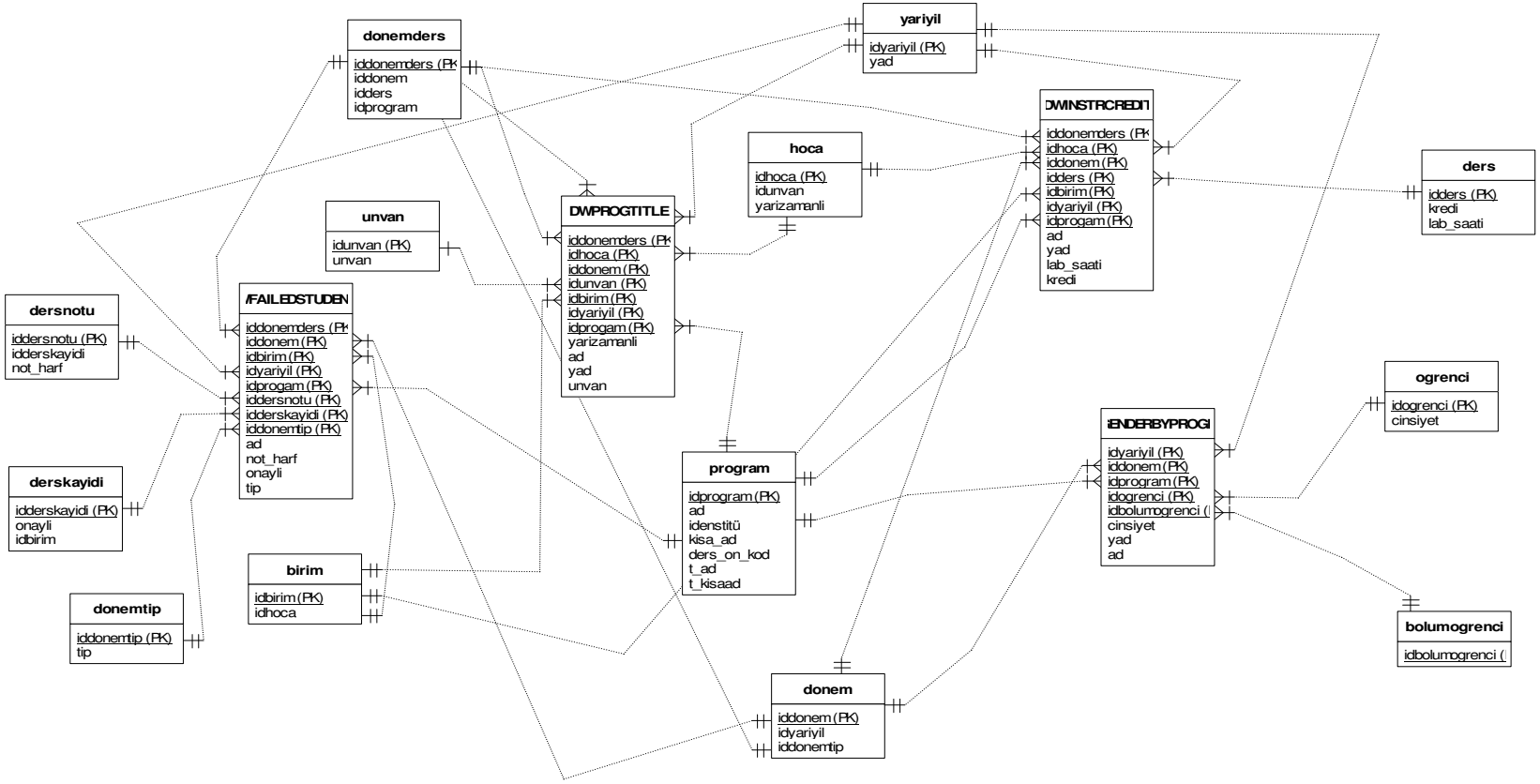

Figure 4.4 Model of the DW by Star Schema

Figure 4.7 represents the third dimensional model of the project. The ***DWFAILEDSTUDENTS Fact table*** is surrounded by dimensions tables, ***donem, yariyil, dersnotu, derskayidi, donemtip, program, birim,*** and ***donemders.***

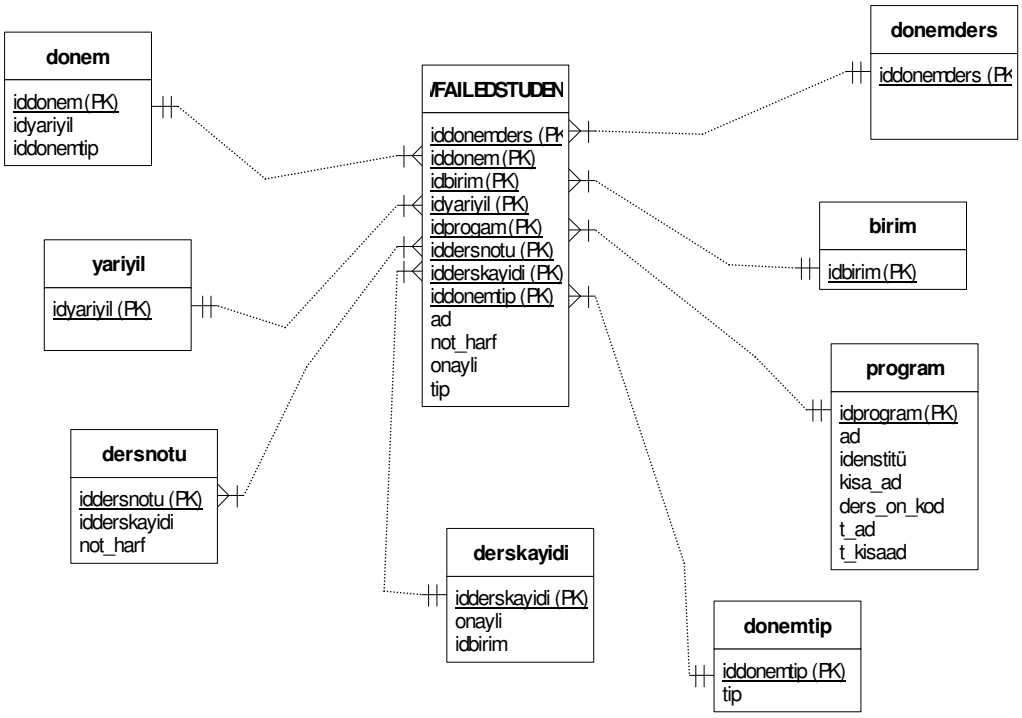

Figure 4.7 Dimensional Model for Students

Figure 4.8 represents the forth dimensional model of the project. The ***DWGENDERBYPROGRAM Fact table*** is surrounded by dimensions tables, ***yariyil, donem,, bolumogrenci, ogrenci,*** and ***program.***

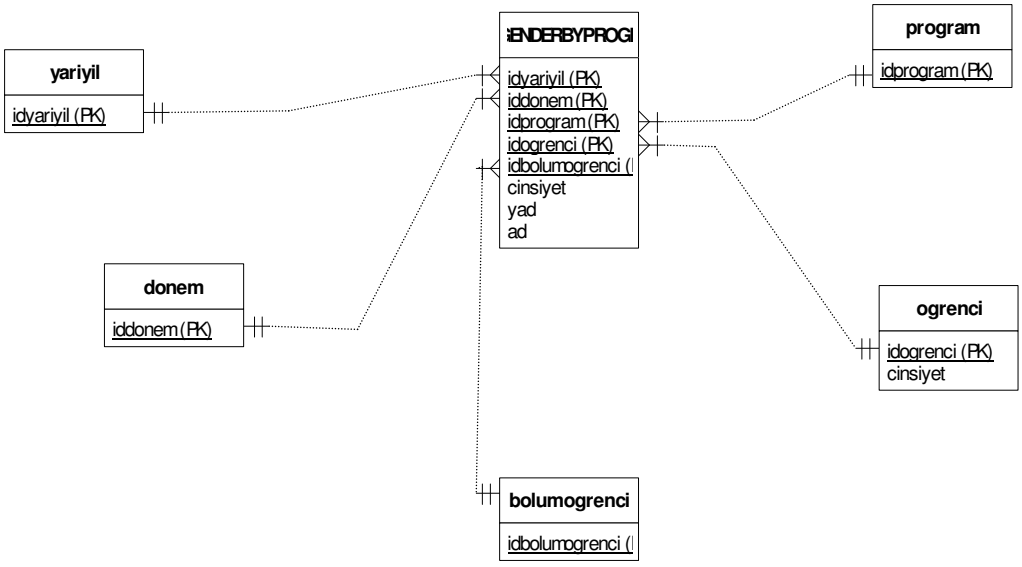

Figure 4.8 Dimensional Model for Genders

4.2.6 Converting PostgreSQL tables into Oracle 10g

Oracle 10g is used in the development of the DW. As the source system data was in PostgreSQL DBMS and as PostgreSQL is an easy to install and administer tool, it is not used by operators of large services because there is no way to build a truly massive PostgreSQL installation or there is no one that can tolerate hardware failures, a convert operation has to be done. The source system tables with their data have to be transformed into Oracle database. This is done by using Oracle PostgreSQL Import, Export and Convert Software 7.0 tool. Figure 4.9 shows an example. A schema called System is created and under this schema, source system tables are converted.

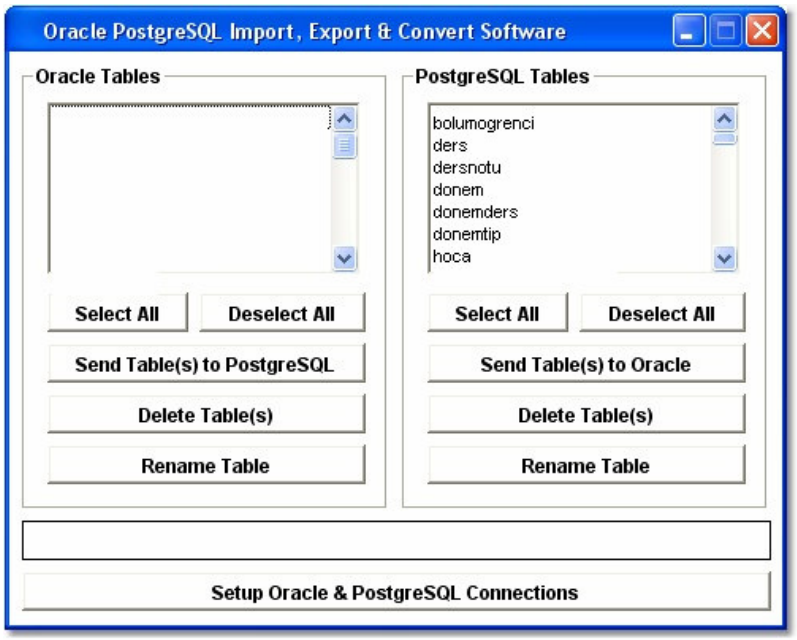

Figure 4.9 Converting operation from PostgreSQL to Oracle

Some of the fields of the tables holding date could not be sent with this tool because of the incompatibility in the data types. SQL insert into command as shown below, are used for this operation.

INSERT INTO ayrilmanedeni (idayrilmanedeni, neden, ayrilmatipi, ayrilmatarihi) VALUES (11, '', 0, to_timestamp('2005-09-20','YYYY-MM-DD'));

4.2.7 Creating the DW Database

After the convert operation of the source system tables, the DW tables are created. This operation is done using **ORACLE Enterprise Manager Console** in **ORACLE Warehouse Builder**. A tool called **SQL Manager Lite for Oracle** is used for easy access to Oracle database. In Figure 4.10 a snapshot of a screen from this tool can be seen.

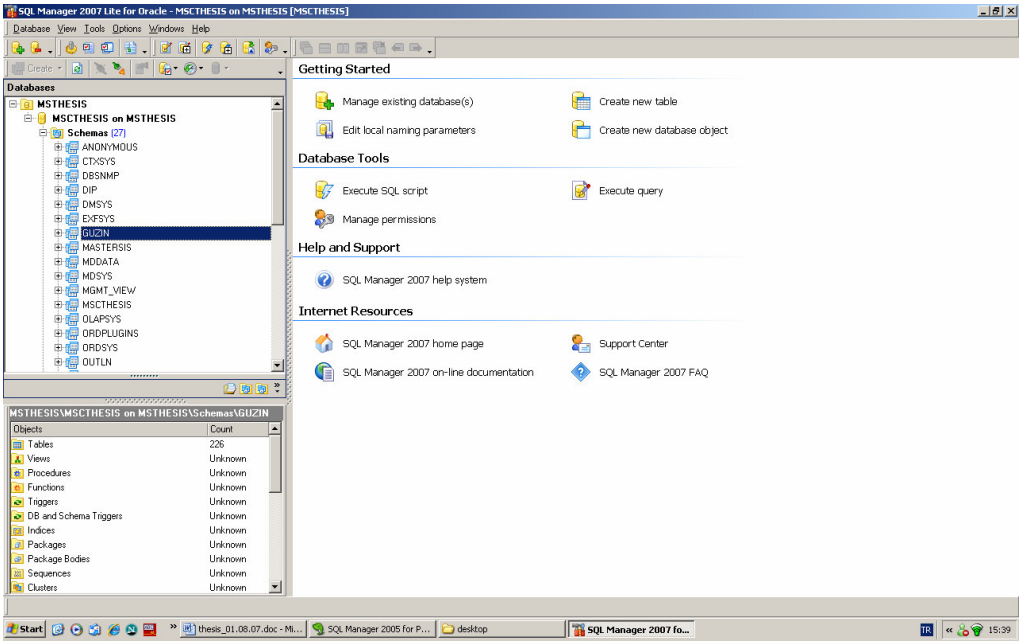

Figure 4.10 A snapshot of a screen from SQL Manager 2007 Lite for Oracle

A new schema is created and the tables are created under this schema using Warehouse Builder. Physical table descriptions are in Appendix D. After the creation of the DW tables, the ETL process that is mentioned in Chapter 2 is performed.

4.2.8 Data Transformation and Migration

Organizations need to be centralized data to improve decision-making process but data can be stored in different sources and different formats. In this study, the source data is only the MasterSIS database. The data flow diagram of the system is shown in Figure 4.11.

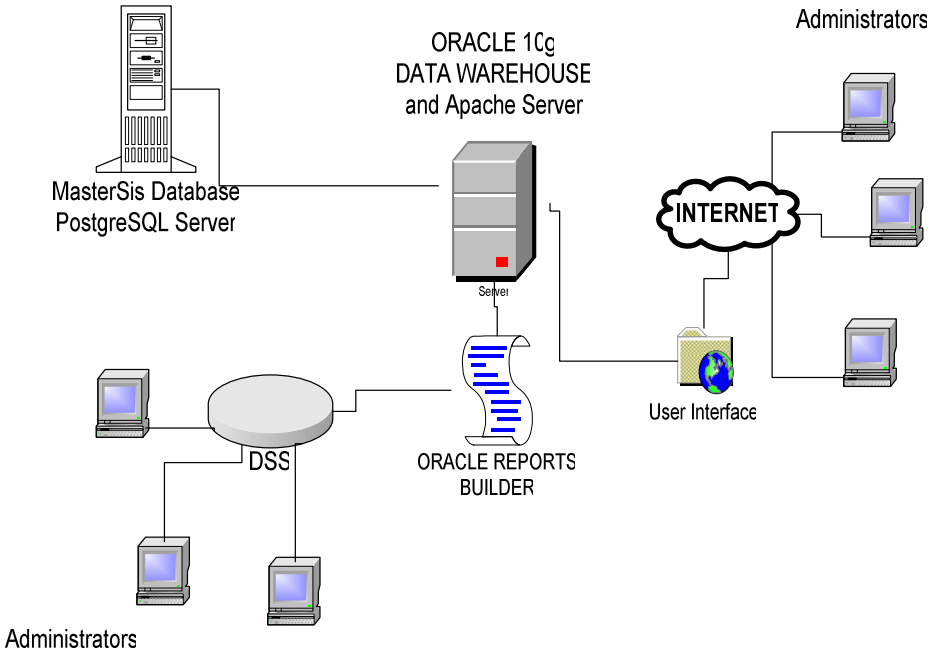

Figure 4.11 Atılım University DW Schema

In DW process, the data transformation and migration is the most important step as data in the DW should be complete, correct and up-to-date. In this study, extraction, transforming, and loading operation of the data from source system to the developed DW is achieved by using PHP language. PHP Hypertext Preprocessor is a programming language that allows web developers to create dynamic content that interacts with databases. This process can be done at the end of each term on a web-based user interface. The PHP codes that obtain this operation can be seen in the APPENDIX E. In Figure 4.12, the user interface created by using **Microsoft FrontPage 2003** and **PHP** language can be seen. User can select the term and by clicking the button, the newly term data taken from the source system is inserted into the DW tables.

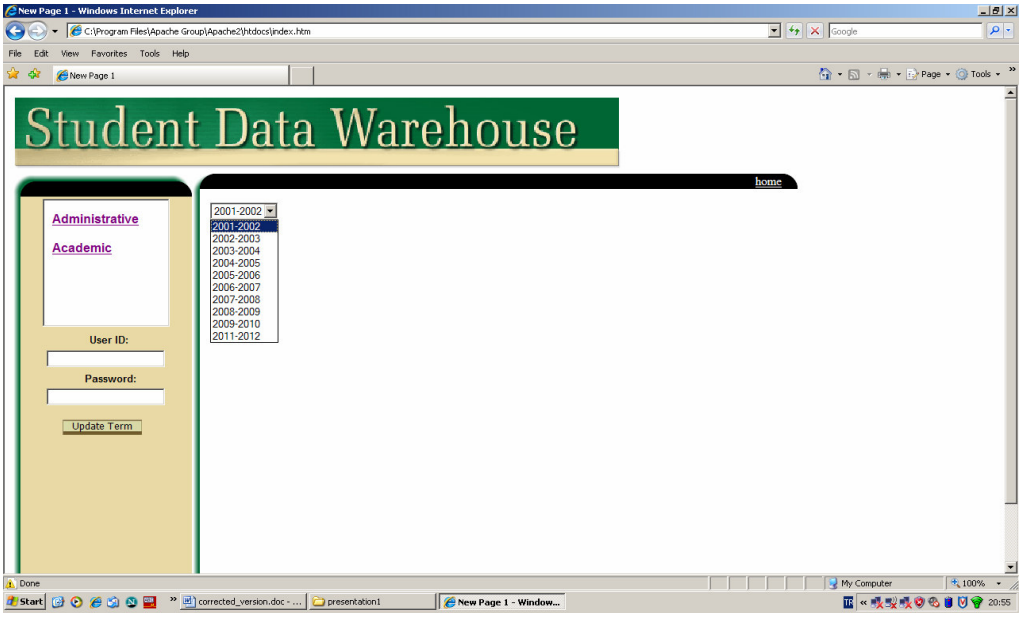

Figure 4.12 User Interface to update term

After the ETL process, the design and development of some reporting applications used for decision support purposes is done. **Oracle Reports Builder** tool is used to implement these applications. Oracle Reports Builder is a product of ORACLE. It is under **Oracle Developer Suite** and it is free software. Reports Builder is an easy to use analysis tool for high-performance end user access information.

4.2.9 Sample Reports

In this section some sample reports are presented that were produced using the DW described in Chapter 2. The purpose of these reports was to show the utility of the DW approach. Although these reports were based on the user requirements, it was not intended to design a full set of management reports. This set of sample reports can be used as a basis for generating a more complete set at a future date. The reports are listed according to academic, student and course reports.

Academic Staff Reports

Figure 4.13 represents the first sample report. This report gives detailed information about the increase in the number of academic staff according to years in The Graduate School of Natural and Applied Sciences concerning five programs. In Figure 4.14, the total number is given according to programs and in Figure 4.15; the total number of academic staff can be seen according to their titles.

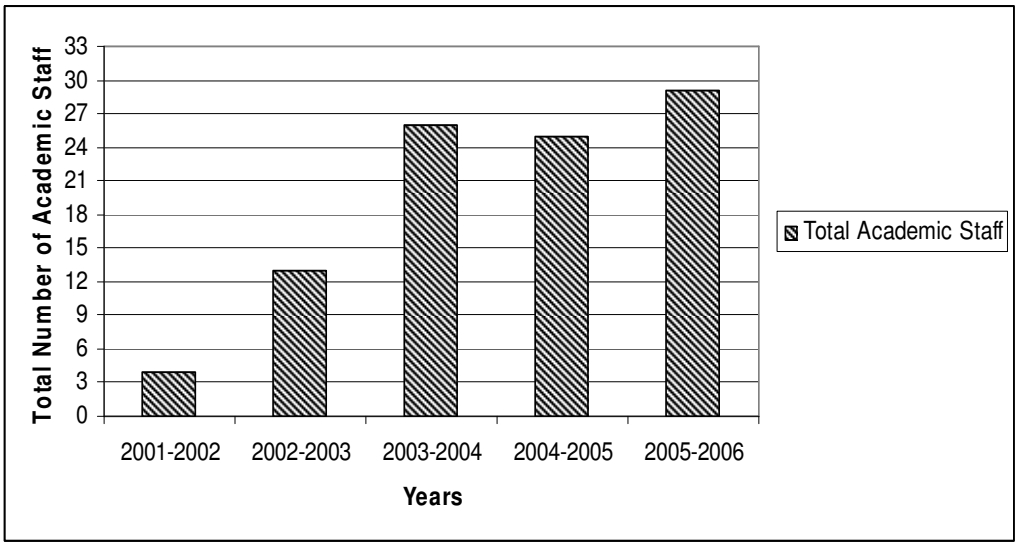

Figure 4.13 Total number of Academic staff in The Graduate School of Natural and Applied Sciences

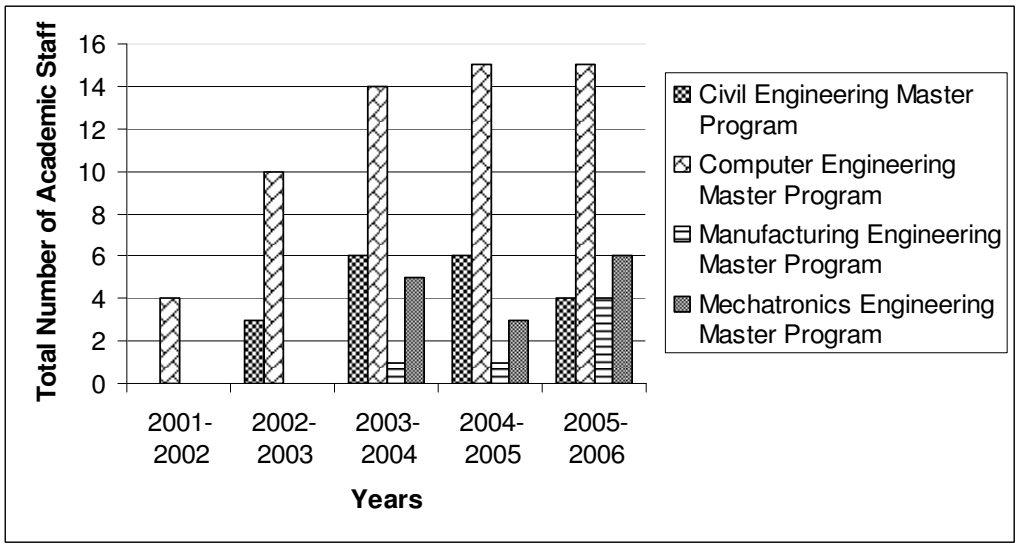

Figure 4.14 Total number of Academic Staff in The Graduate School of Natural and Applied Sciences According to Programs in a range of years

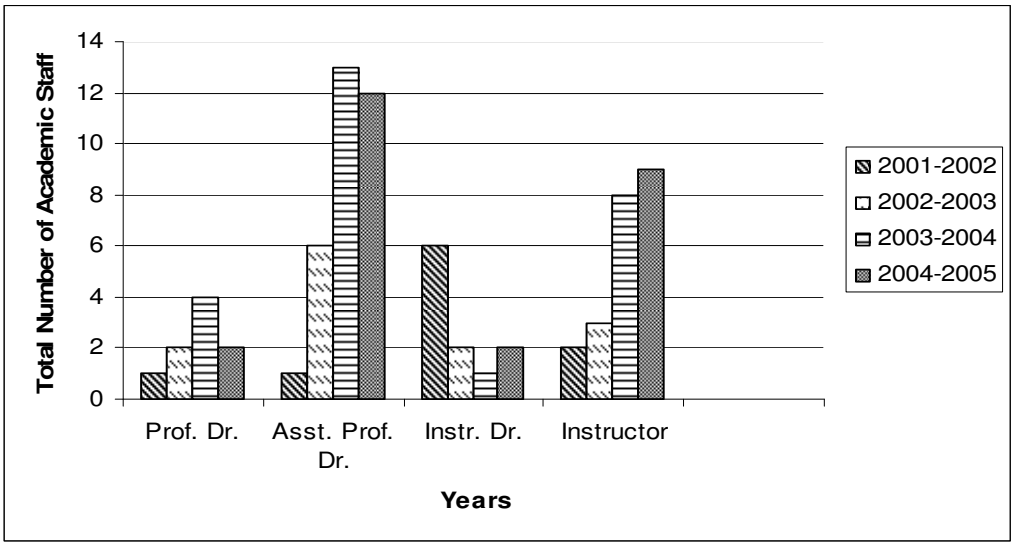

Figure 4.15 Total Number of Academic Staff in The Graduate School of Natural and Applied Sciences According to Titles in a Range of Years

Figure 4.16 shows the number of part time academic staff in each academic program for different terms. The same report can also be generated for the full-time academic staff.

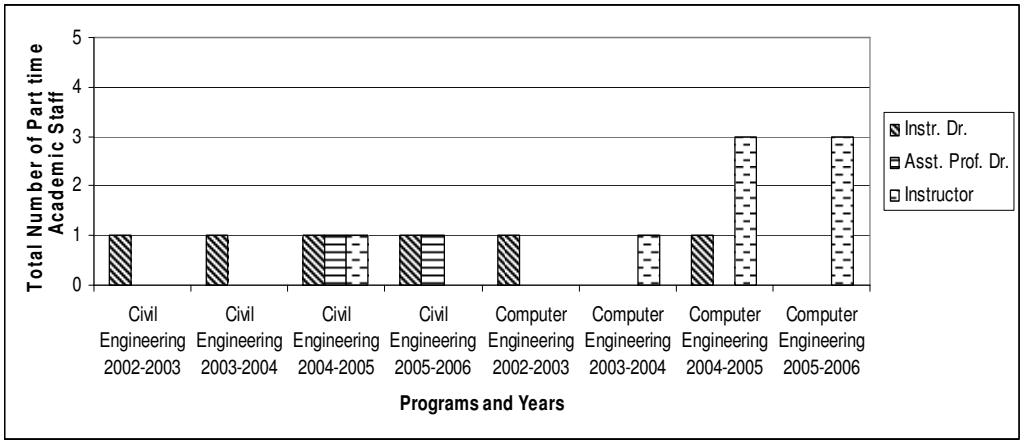

Figure 4.16 Total Number of Part time Academic Staff in The Graduate School of Natural and Applied Sciences According to Their Programs and Titles in a Range of Years

Figure 4.17 shows the course loads totals on academic staffs in different programs and different years. The graphs can also be viewed in report form. In Figure 4.18, the course loads in report form can be examined with instructor Id's. Decision-maker can get different solutions by combining these two reports.

Figure 4.17 shows the total number of credits an academic staff gives in different programs and different years.

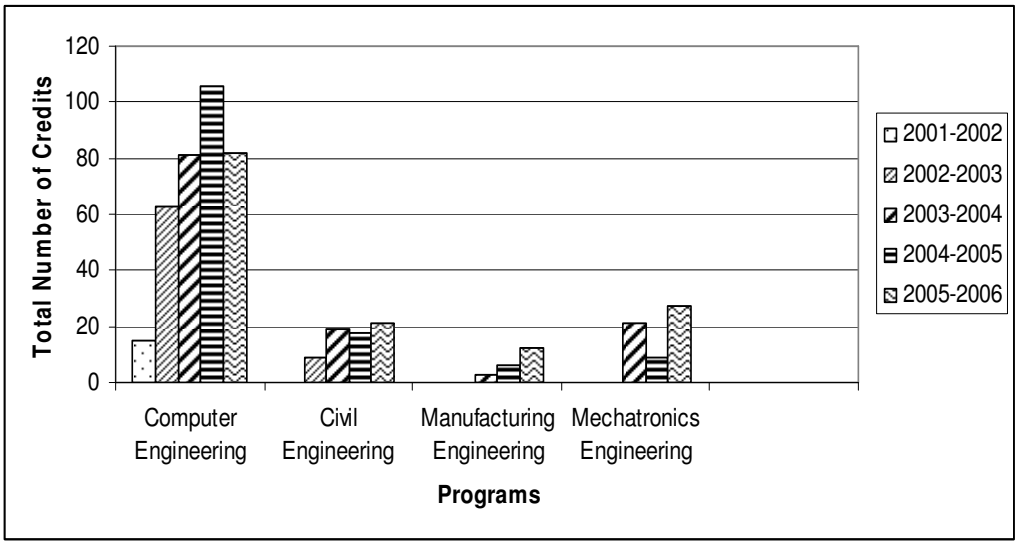

Figure 4.17 Total numbers of credits an academic staff in The Graduate School of Natural and Applied Sciences gives in a range of years

This graph can be viewed in a report format in detail with instructor id's and credits as shown in Figure 4.18. The report is sorted by Instructor Id.

Total Number of Credits an Academic Staff Gives in a Range of Years

Instructor Id	Term	Total Number Of Credit	Program
35	2001-2002	3	Computer Engineering - Master's Program
	2002-2003	9	Computer Engineering - Master's Program
	2003-2004	12	Computer Engineering - Master's Program
	2004-2005	6	Computer Engineering - Master's Program
	2005-2006	6	Computer Engineering - Master's Program
36	2001-2002	3	Computer Engineering - Master's Program
	2002-2003	6	Computer Engineering - Master's Program
	2003-2004	6	Computer Engineering - Master's Program
	2004-2005	9	Computer Engineering - Master's Program
	2005-2006	9	Computer Engineering - Master's Program
37	2001-2002	6	Computer Engineering - Master's Program
	2002-2003	0	Computer Engineering - Master's Program
	2002-2003	15	Computer Engineering - Master's Program
	2003-2004	0	Computer Engineering - Master's Program
	2003-2004	12	Computer Engineering - Master's Program
	2004-2005	0	Computer Engineering - Master's Program
	2004-2005	12	Computer Engineering - Master's Program
	2005-2006	0	Computer Engineering - Master's Program

Figure 4.18 Total numbers of credits an academic staff in The Graduate School of Natural and Applied Sciences gives in a range of years with Instructor Id's

Student Reports

The user can generate graphics on student data as well as academic staff data. Figure 4.19 shows the Master Programs' population with respect to years in between 2001-2005. In Figure 4.20, registration to programs according to years and departments can be seen. These graphics gives opinion to the administration about programs having high demands.

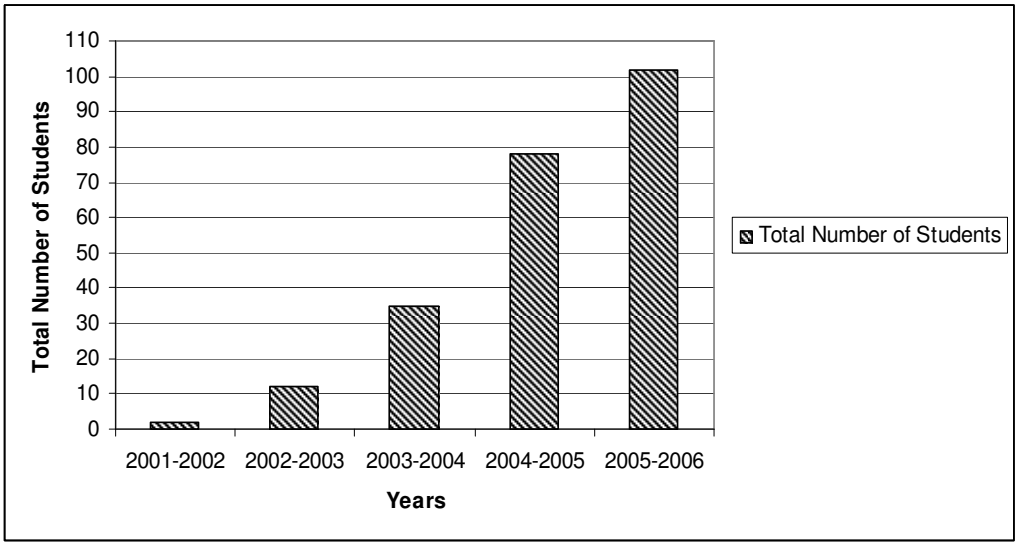

Figure 4.19 Total numbers of students in The Graduate School of Natural and Applied Sciences in a range of years

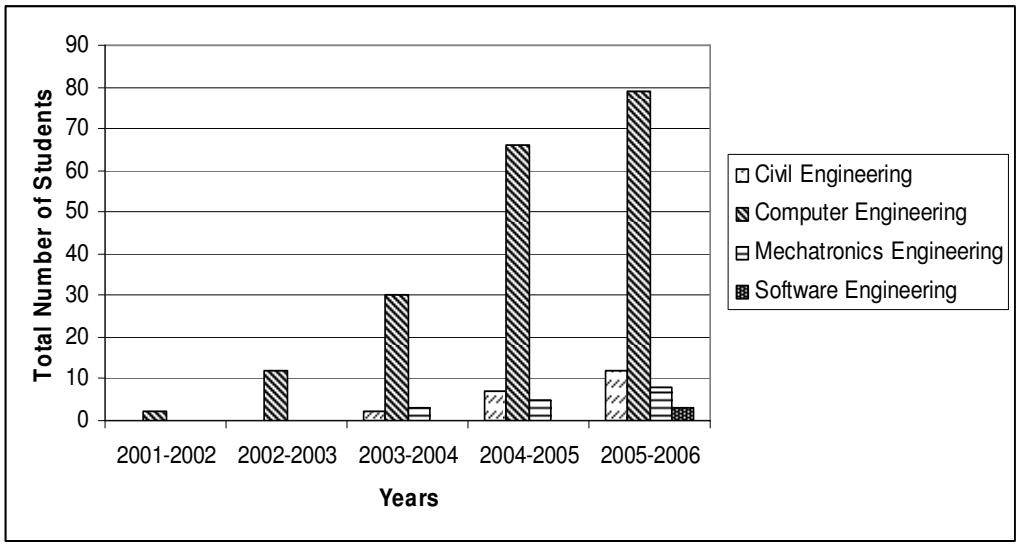

Figure 4.20 Total numbers of students in The Graduate School of Natural and Applied Sciences according to their programs in a range of years

Figures 4.21 and Figure 4.22 provides us with the number of male and female students in each academic program for given terms. This report is useful in showing demographic trends, which needed to be reported to external organizations. Also these graphs show the trend of enrollment according to gender which can be used to help in recruitment and retention activities.

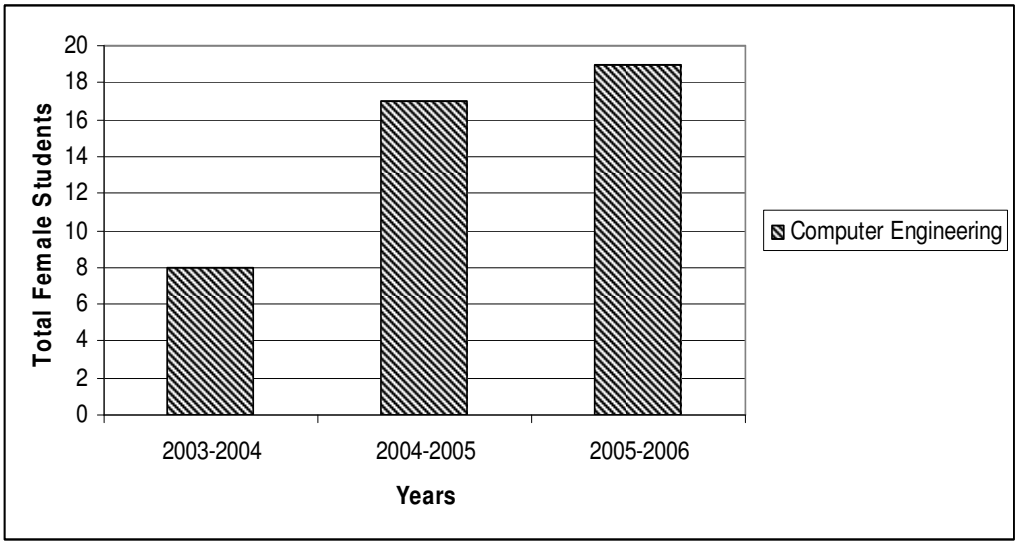

Figure 4.21 Total numbers of female students in The Graduate School of Natural and Applied Sciences according to their programs in a range of years

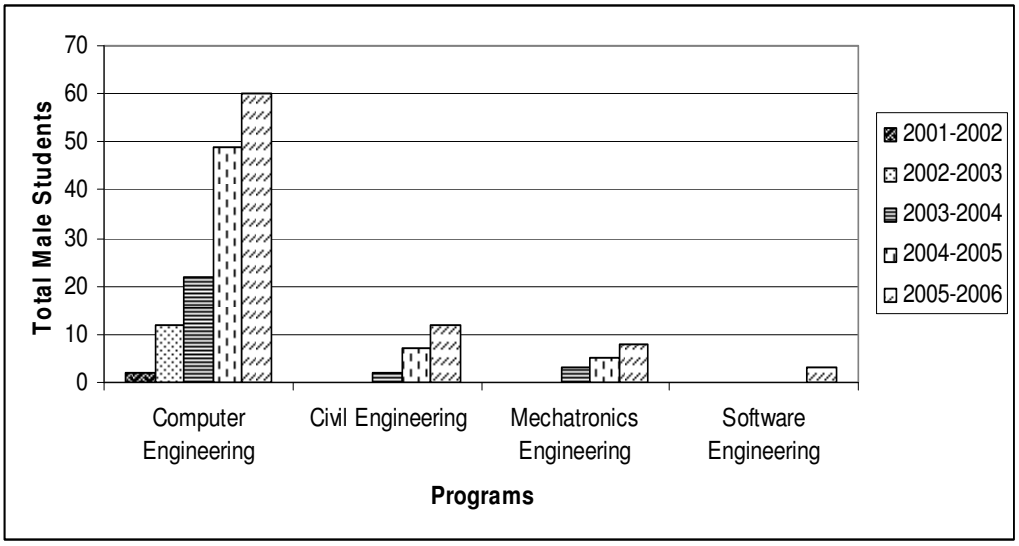

Figure 4.22 Total numbers of male students in The Graduate School of Natural and Applied Sciences according to their programs in a range of years

The academic successes of students are generally measured in terms of the grade point average (GPA) and the cumulative grade point averages (CGPA). Security obligations of the University did not let us use the grades and GPAs, that is why, the successes in courses are shown with letter grades. Figure 4.23 shows the letter grade distribution of a course in Computer Engineering Master Program in years between 2003-2004.

Figure 4.23 Distribution of Letter Grades in a course in Computer Engineering Master Program in 2003-2004

Figure 4.24 represents the total number of students that have passed from courses in Computer Engineering Master Program in 2005-2006 and in Figure 4.25; the number of failed students from the same courses in the same year can be seen. These reports help the administrator in order to make comments about the courses.

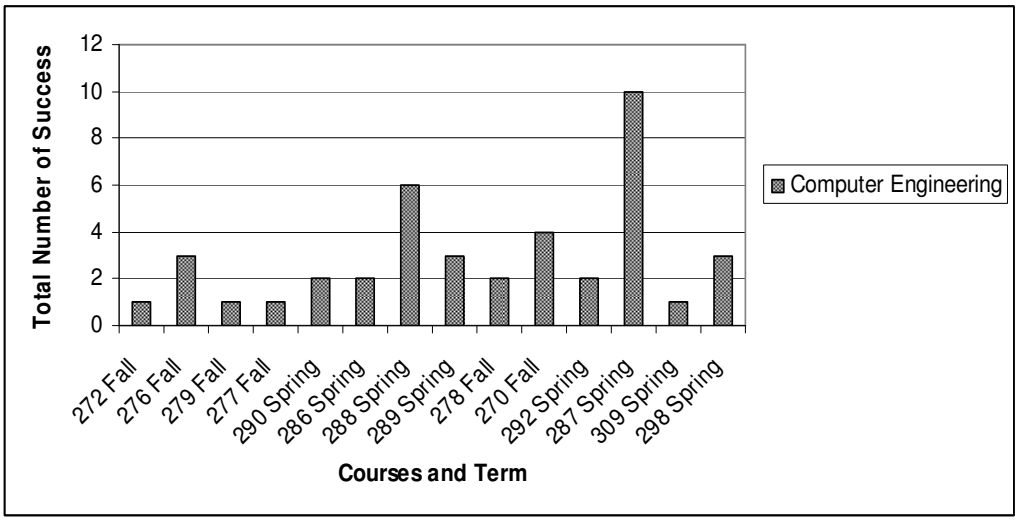

Figure 4.24 Total number of students pass from all courses in Computer Engineering in term 2005-2006

Total Number of Failed Students from All Courses in Computer Engineering in Term 2005-2006

Program	Donemders	Idyariyil	Tip	Total Fails
Computer Engineering - Master's Program	284	1	Fall	1
	287	1	Spring	1
	272	1	Fall	2
	286	1	Spring	2
	292	1	Spring	1
	294	1	Spring	1
	298	1	Spring	2
	294	1	Spring	1
	291	1	Spring	2
	278	1	Fall	1
	Total:			**14**
Total:				**14**

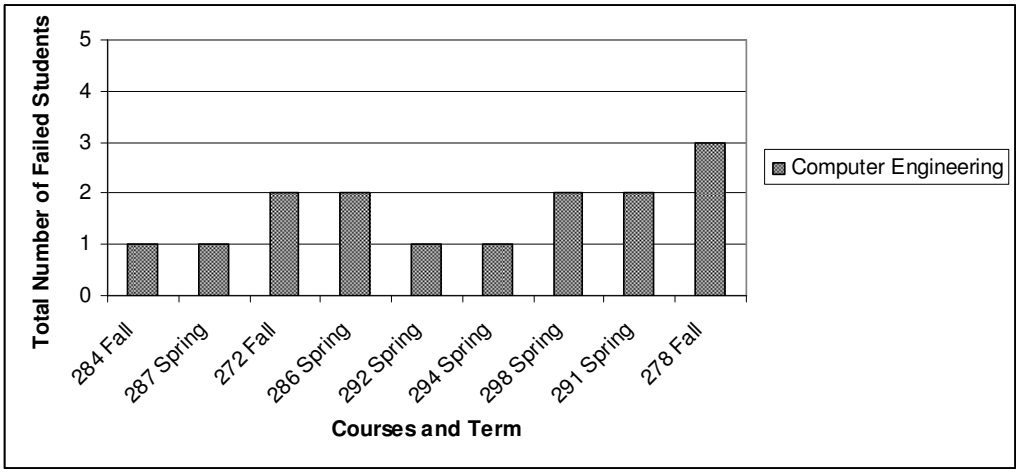

Figure 4.25 Total number of failed students from all courses in Computer Engineering in term 2005-2006

4.2.10 User Interface

The reports that were generated before using Oracle Reports Builder tool can be viewed by the user in an internet-based environment. Internet Explorer 6.0 or later can be used in order to view the reports. The site is done using Microsoft Office FrontPage tool as it was mentioned in Section 4.2.8.

In Figure 4.26 the index page is shown. The user has two main hyperlinks; administrative and instructor. These links shows different reports.

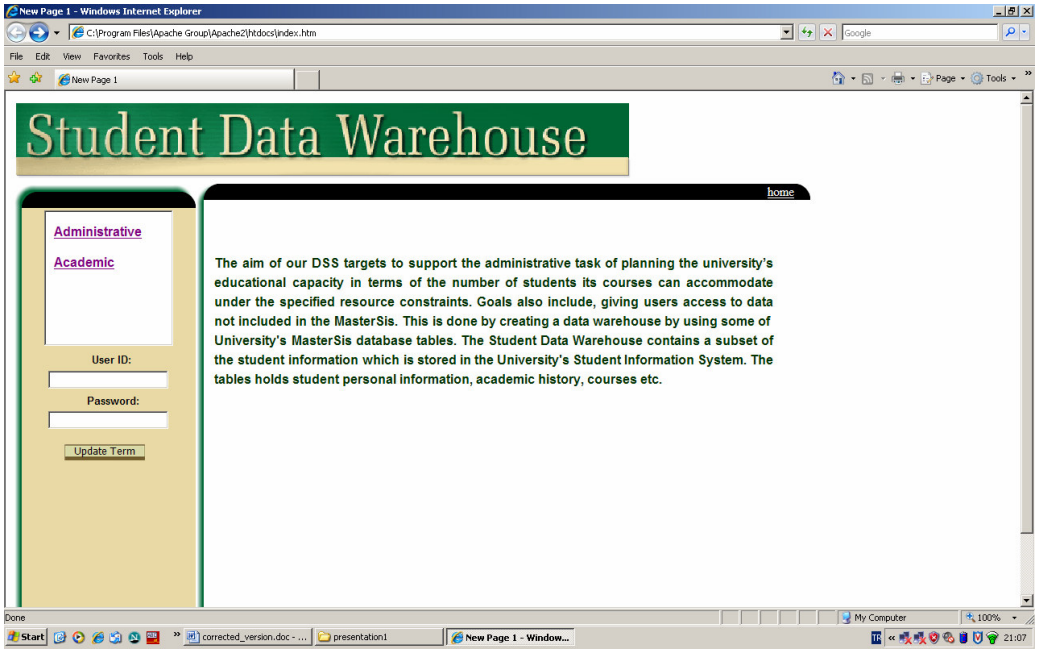

Figure 4.26 The Index Page of the Site

Figures 4.27 and 4.28 shows the reports when user clicks administrative or instructor link.

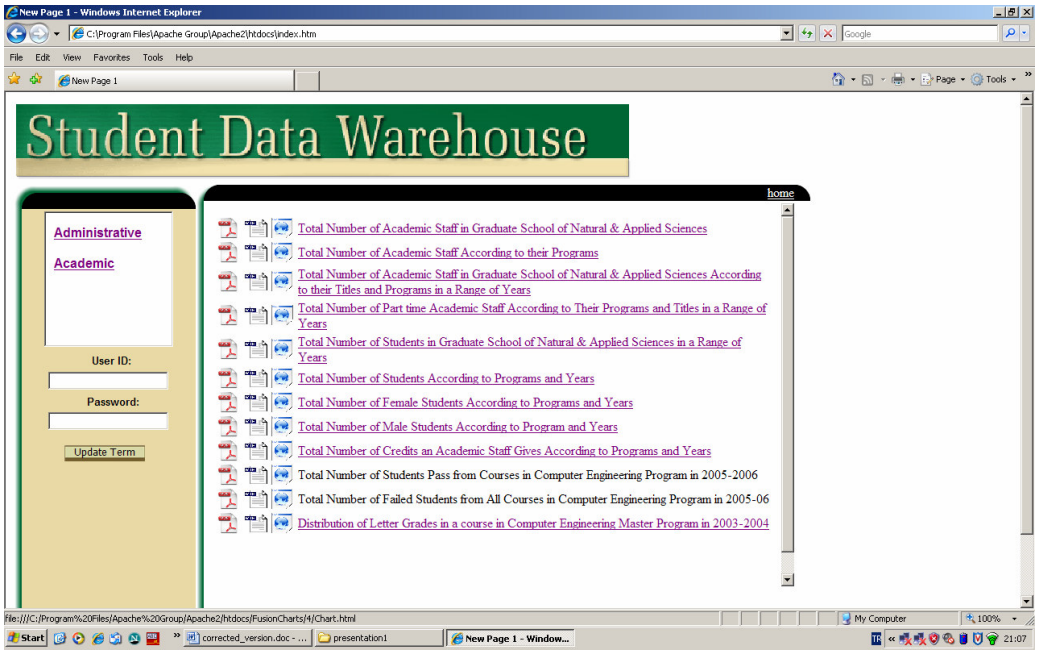

Figure 4.27 Administrative Hyperlink

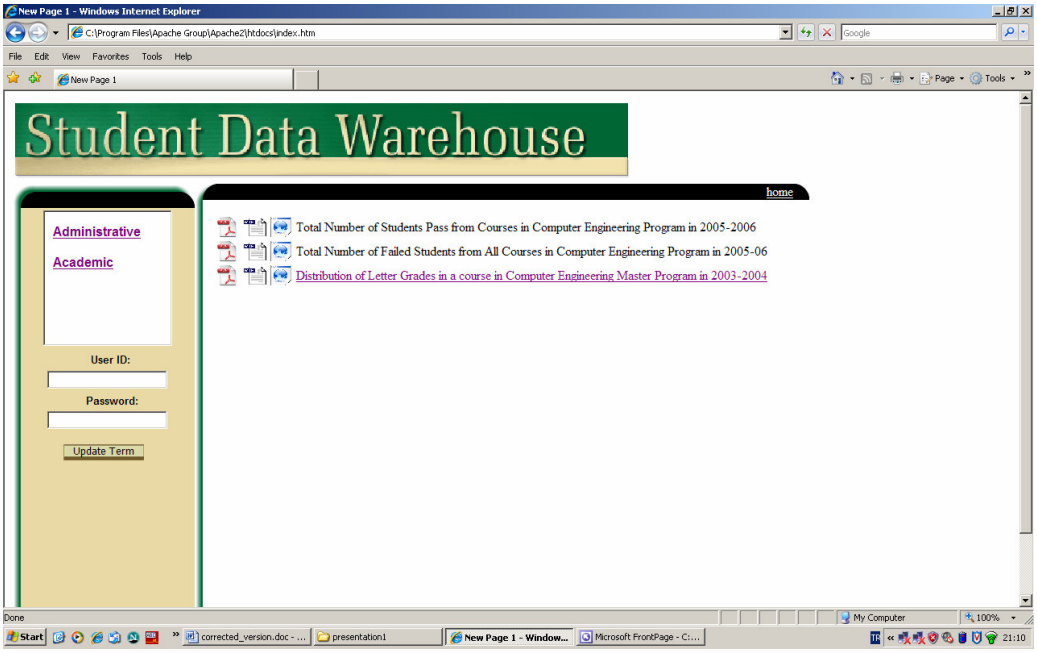

Figure 4.28 Instructor Hyperlink

The reports can be viewed in HTML, PDF or RTF formats. The below figure show the reports in different formats. If the users want to download a report, they can save the reports by clicking the report type icon. Figures 4.29-4.31 shows the reports screen shots.

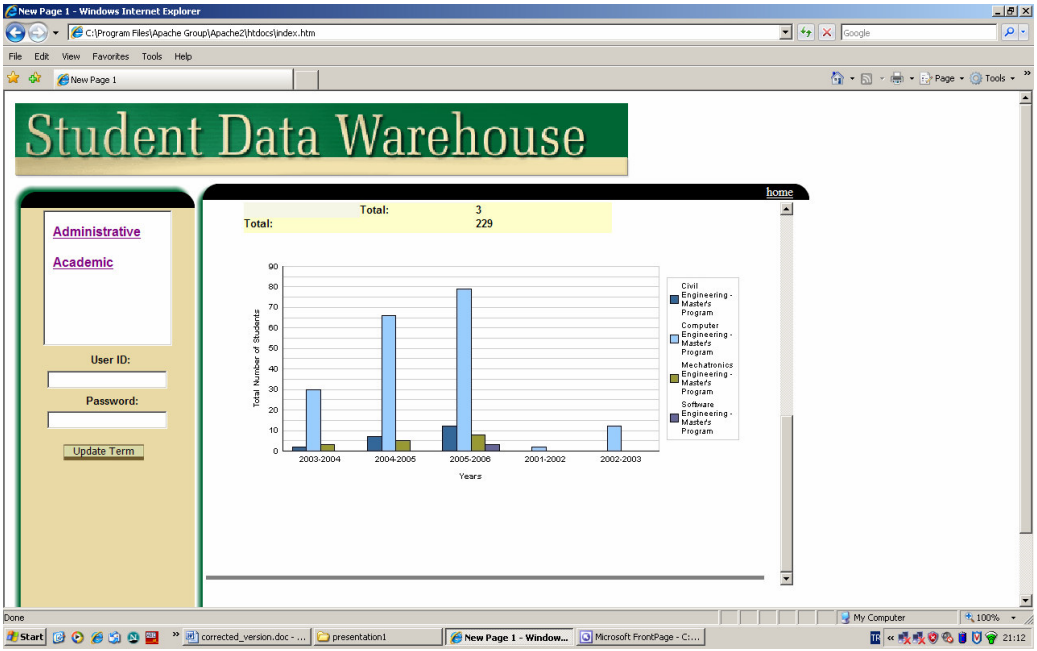

Figure 4.29 Report in HTML format

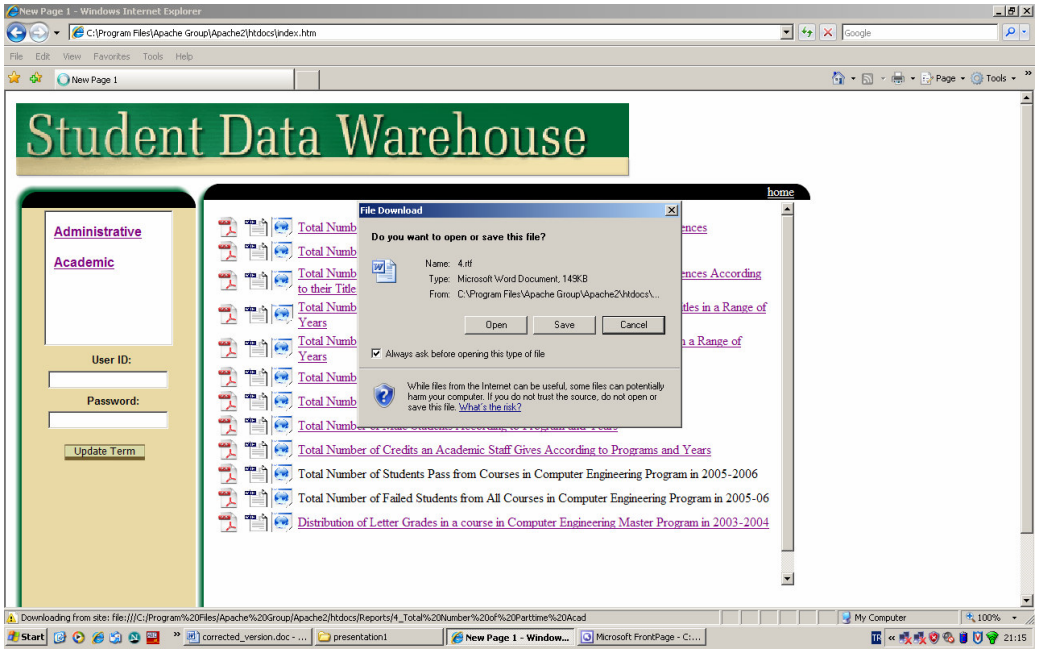

Figure 4.30 Report in RTF format

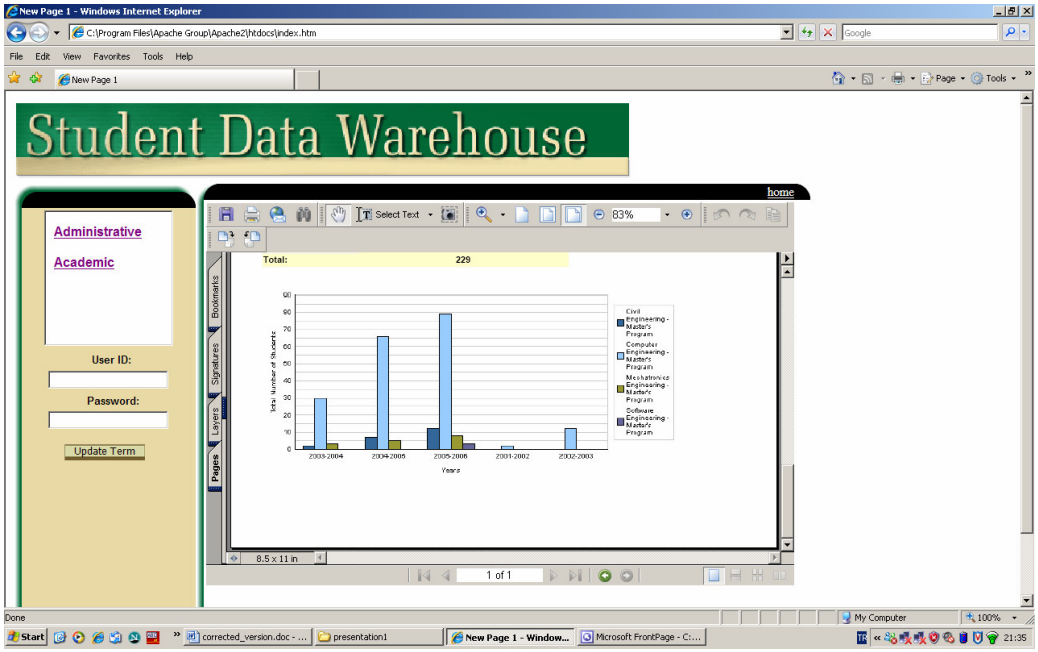

Figure 4.31 Report in PDF format

As it was mentioned in Section 4.2.8, database admin can load the new term data to the DW tables on an internet-based environment entering his username and password to the related fields. Figure 4.32 shows the screenshot of the update page.

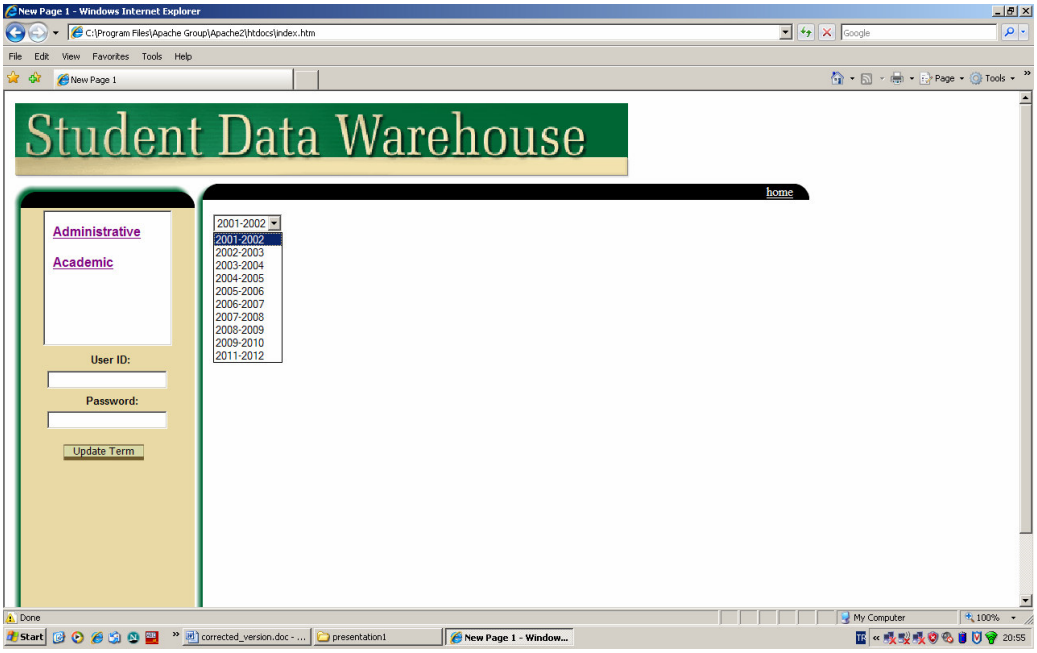

Figure 4.32 Update Screen

4.2.11 Requirement Gathering

The first step in the development methodology is the development of a set of requirements. This section describes the results of the second interview done with three people from administrative unit and three from academic unit of the Atilim University. These interviews are conducted in order to gather their further requirements and opinions on the developed system as well as their expectations for the future study of the developed system. The interview questions can be found in Appendix B.

During the interviews, the questions were asked about the use of the system and how this system can make their work easier. They found the system very useful for report building and analyzing. They said that, this system will save their time. One of the administrative staff said that;

"I am spending lots of time to create these kinds of reports. Most of the reports you have generated are very similar to mine and with this system I can create them in a few minutes. In order to find the academic staff total for a specific program, I have to connect MasterSIS and count the staff one by one and if it is needed for different

years, I have to do the same thing from hundreds of records. After that I carry the results to MS Excel in order to create a chart."

Another academic staff, who is holding a managerial position in the university, said that;

"We are giving such reports to the Board of Trustees and we need these kinds of graphics to see the changes in numbers either in students or in academic staff in order to make planning."

The results of the interview will guide us during the future work of this study.

4.2.12 Backup Operation

Backup operation is the most important step in a DW. This operation saves the data from disk failure, human error, software failure, fire, and so on. Database administrator has to get the backup of the system at the end of every term after the updating of the new term data. Backup operation can be done using SQL Manager tool. In Figure 4.33 backup screen can be seen. Database administrator can get the backup of the DW, from the tools menu → Extract Database option.

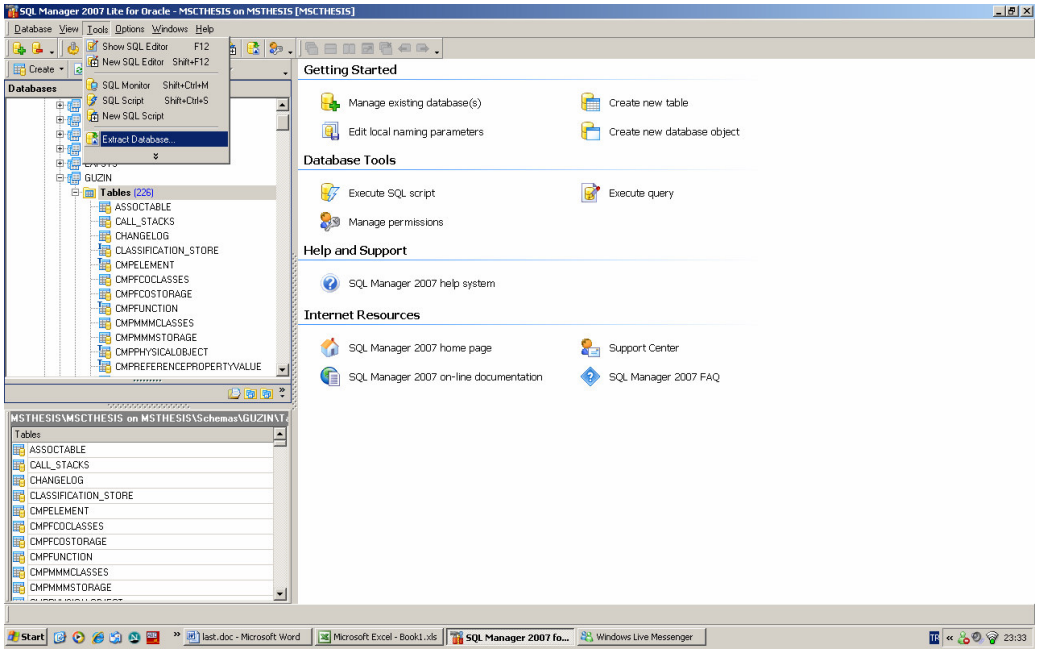

Figure 4.33 Backup Screen in SQL Manager Lite for Oracle 10g

Figure 4.34 shows the starting of a backup operation screen. Database administrator can save a copy of the DW into a file by clicking next button as shown below.

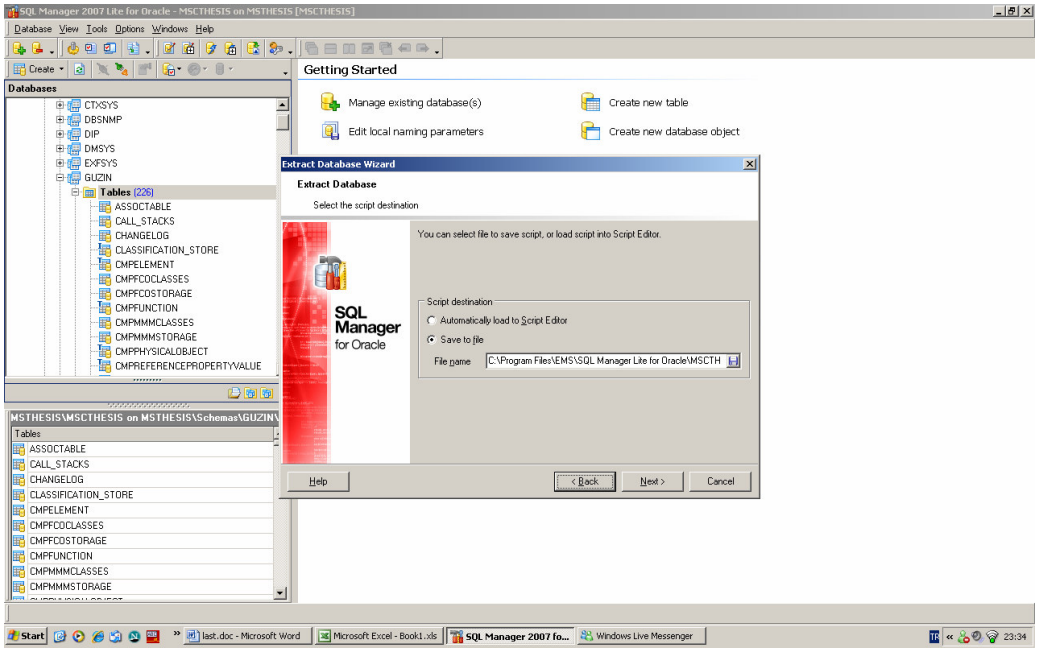

Figure 4.34 Starting Backup Operation

CHAPTER 5

DISCUSSIONS and CONCLUSIONS

In this Chapter we summarize the work accomplished and give some directions for future work. The main objective of this Master's thesis was to develop a DW for Atılım University Master Program in order to help decision makers. This objective has been accomplished. The university now has a better source of information on which to base decision-making. The major benefits of the proposed DSS can be listed as follows:

- By using the dimension modeling, a single dimension table of the DW can be used against multiple fact tables. These tables can be varied upon the user needs.
- Retrieving the historical data is quick so users will save time.
- The system can answer multiple questions. The queries written in SQL can be varied upon the user needs. Users can generate new reports and graphics using the pre-created queries and can write their own queries by only changing some special fields in SQL statements.
- Possible users of the system can access the data easily using Oracle Reports Builder tool installed their computers.
- Database admin can easily update the new term data into the DW tables on internet environment.

However, implementing a DW is a difficult task. It needs to be implemented carefully and step by step without making any mistake in terms of technology choice. In developing this DSS, a life cycle development methodology proposed by Kimball [24] has been used. Based on user requirements, the required data is extracted from

MasterSIS source system and the data is converted into Oracle database. This operation took longer time than planned because finding a suitable tool wasted our time. Lots of tools are tested like Quella Cognitus, Quella Creator, Cognos but a case tool for a complete DW design is not available and most of them are for commercial use. Oracle Warehouse Builder, a free Oracle tool is used for this purpose. Because of the incompatibility in the data types of two different DBMS, we faced another bottleneck. This is solved using SQL command. The reports developed in this Masters thesis are primarily used to show the utility of the DW approach. In the future, a wider set of reports should be designed according to the organizational requirements.

Besides the problems and advantages, the use of a DW presents several potential advantages for higher education institutions, including timely access to evaluate data. The use of a DW allows an educational institution to use information in making appropriate decisions, and that is the desired end goal of the data warehousing process.

6.1 Limitations of the study

There were some limitations in this study. One of them was the limitation of the data. Because of the security problems, the university whole data could not be taken. If the whole data has been given, the results of the reports would be more realistic.

The only source was the MasterSIS data in this study. Other sources of information systems can be added like undergraduate, finance, human resources, and library systems.

During this study no financial support is taken so, free tools are used and because of time limitation the system usage can not be monitored.

6.2 Future Study

For this thesis, the source data was limited in scope to the master student information contained in MasterSIS by taking into consideration the internal use of students' and academic staff's personal information. In the future, more student information can be added to the DW and the DW can be supported by some other systems. Besides these, developing a user friendly and more easy to use query generator tool is planned for a future study.

As a final word, it can be said that in education, building a DW is more than necessary especially for resource planning and decision-making.

APPENDIX A

Interview Guide 1

REQUIREMENTS ANALYSIS

- What type of analysis in routine do you perform?
- How do you get the data in order to do your analysis?
- Does the current system answer your needs?
- What kind of difficulties do you have with the current system?
- What steps do you follow in making these analyses?
- How much historical information is required?
- What capabilities do you think should be added to the system?

APPENDIX B

Interview Guide 2

REQUIREMENTS ANALYSIS

- What do you think about the system?
- Do you think this system can make your work easy?
- What else should be added to the system do you think?
- What sort of benefits can this system bring? Do you think this system can contribute to Atilim University's resource planning?
- Who do you think can use this system?

APPENDIX C

Table Data Description

Original Table **English Translation**

YARIYIL TABLE TERM TABLE

AD	IDYARIYIL
2001-2002	5
2002-2003	6
2003-2004	7
2004-2005	2
2005-2006	1

TERM	IDTERM
2001-2002	5
2002-2003	6
2003-2004	7
2004-2005	2
2005-2006	1

PROGRAM TABLE

AD	IDPROGRAM	PROGRAM	IDPROGRAM
Computer Engineering - Master's Program	19	Computer Engineering - Master's Program	19
Civil Engineering - Master's Program	20	Civil Engineering - Master's Program	20
Manufacturing Engineering – Master's Program	22	Manufacturing Engineering – Master's Program	22
Mechatronics Engineering - Master's Program	23	Mechatronics Engineering - Master's Program	23
Software Engineering - Master's Program	24	Software Engineering - Master's Program	24

TITLE TABLE

IDUNVAN	UNVAN
10	Prof.Dr
11	Assoc.Prof. Dr.
14	Asst. Prof. Dr.
26	Instructor
28	Instr. Dr
31	Assistant

IDTITLE	TITLE
10	Prof.Dr
11	Assoc.Prof. Dr.
14	Asst. Prof. Dr.
26	Instructor
28	Instr. Dr
31	Assistant

APPENDIX D

Physical Fact Table Descriptions

DWPROGTITLE FACT TABLE

```
CREATE TABLE GUZIN.DWPROGTITLE
(
  IDDONEMDERS NUMBER(10,0),
  IDHOCA NUMBER(10,0),
  IDDONEM NUMBER(10,0),
  IDUNVAN NUMBER(10,0),
  IDBIRIM NUMBER(10,0),
  IDYARIYIL NUMBER(10,0),
  IDPROGRAM NUMBER(10,0),
  PARTTIME NUMBER(1,0),
  AD VARCHAR2(50),
  YAD VARCHAR2(10),
  UNVAN VARCHAR2(25)
)
Begin
for rec in
( select SYSTEM.SECTION.idbirim,
SYSTEM.DONEM.iddonem,
SYSTEM.YARIYIL.idyariyil,
SYSTEM.DONEMDERS.iddonemders,
SYSTEM.HOCA.idhoca,
SYSTEM.UNVAN.idunvan,
SYSTEM.PROGRAM.idprogram,
SYSTEM.HOCA.parttime,
SYSTEM.PROGRAM.ad,
SYSTEM.YARIYIL.yad,
SYSTEM.UNVAN.unvan
from SYSTEM.program, SYSTEM.donem, SYSTEM.donemders, SYSTEM.hoca,
SYSTEM.section, SYSTEM.unvan, SYSTEM.yariyil
where SYSTEM.donem.iddonem = SYSTEM.donemders.iddonem
and SYSTEM.donemders.iddonemders = SYSTEM.section.iddonem_ders
and SYSTEM.hoca.idhoca = SYSTEM.section.idhoca
and SYSTEM.hoca.idunvan = SYSTEM.unvan.idunvan
```

```
and SYSTEM.program.idprogram = SYSTEM.donemders.idprogram
and SYSTEM.yariyil.idyariyil = SYSTEM.donem.idyariyil) loop

Insert into GUZIN.DWPROGTITLE (iddonemders, idhoca, iddonem, idunvan,
idbirim, idyariyil, idprogram, parttime, ad, yad, unvan)
values (rec.iddonemders, rec.idhoca, rec.iddonem, rec.idunvan, rec.idbirim,
rec.idyariyil, rec.idprogram, rec.parttime, rec.ad, rec.yad, rec.unvan);
end loop;
End;
```

DWINSTRCREDIT FACT TABLE

```
CREATE TABLE GUZIN.DWINSTRCREDIT
(
  IDDERS NUMBER(10,0),
  KREDI NUMBER(10,0),
  LAB_SAATI NUMBER(1,0),
  IDDONEM NUMBER(10,0),
  IDDONEMDERS NUMBER(10,0),
  IDHOCA NUMBER(10,0),
  IDBIRIM NUMBER(10,0),
  IDYARIYIL NUMBER(10,0),
  IDPROGRAM NUMBER(10,0),
  AD VARCHAR2(50),
  YAD VARCHAR2(10)
)
Begin
for rec in
( select SYSTEM.DERS.idders,
SYSTEM.SECTION.idbirim,
SYSTEM.DONEM.iddonem,
SYSTEM.YARIYIL.idyariyil,
SYSTEM.YARIYIL.yad,
SYSTEM.DONEMDERS.iddonemders,
SYSTEM.HOCA.idhoca,
SYSTEM.PROGRAM.idprogram,
SYSTEM.PROGRAM.ad,
SYSTEM.DERS.kredi,
SYSTEM.DERS.lab_saati

from SYSTEM.ders, SYSTEM.program, SYSTEM.donem, SYSTEM.donemders,
SYSTEM.hoca, SYSTEM.section, SYSTEM.yariyil
where SYSTEM.donem.iddonem = SYSTEM.donemders.iddonem
and SYSTEM.donemders.iddonemders = SYSTEM.section.iddonem_ders
and SYSTEM.hoca.idhoca = SYSTEM.section.idhoca
and SYSTEM.ders.idders = SYSTEM.donemders.idders
and SYSTEM.program.idprogram = SYSTEM.donemders.idprogram
and SYSTEM.yariyil.idyariyil = SYSTEM.donem.idyariyil) loop
```

```
Insert into GUZIN.DWINSTRCREDIT(iddonem, idders, iddonemders, idhoca,
idprogram, idbirim, idyariyil, kredi, lab_saati, ad, yad)
values (rec.iddonem, rec.idders, rec.iddonemders, rec.idhoca, rec.idprogram,
rec.idbirim, rec.idyariyil, rec.kredi, rec.lab_saati, rec.ad, rec.yad);
end loop;
End;
```

DWGENDERBYPROGRAM FACT TABLE

```
CREATE TABLE GUZIN.DWGENDERBYPROGRAM
(
  IDYARIYIL NUMBER(10,0),
  IDDONEM NUMBER(10,0),
  IDPROGRAM NUMBER(10,0),
  IDOGRENCI NUMBER(10,0),
  CINSIYET CHAR(1),
  YAD VARCHAR2(10),
  AD VARCHAR2(50)
)
Begin
for rec in
( select SYSTEM.OGRENCI.idogrenci,
SYSTEM.OGRENCI.cinsiyet,
SYSTEM.DONEM.iddonem,
SYSTEM.YARIYIL.idyariyil,
SYSTEM.YARIYIL.yad,
SYSTEM.PROGRAM.idprogram,
SYSTEM.PROGRAM.ad

from SYSTEM.program, SYSTEM.donem, SYSTEM.yariyil, SYSTEM.ogrenci,
SYSTEM.bolumogrenci, SYSTEM.ogrencidonemderskayit
where SYSTEM.ogrenci.idogrenci = SYSTEM.bolumogrenci.idogrenci
and SYSTEM.bolumogrenci.idbolumogrenci =
SYSTEM.ogrencidonemderskayit.idbolum_ogrenci
and SYSTEM.bolumogrenci.idprogram = SYSTEM.program.idprogram
and SYSTEM.ogrencidonemderskayit.iddonem = SYSTEM.donem.iddonem
and SYSTEM.donem.idyariyil = SYSTEM.yariyil.idyariyil) loop

Insert into GUZIN.DWGENDERBYPROGRAM(idogrenci, cinsiyet, iddonem,
idyariyil, idprogram, ad, yad)
values (rec.idogrenci, rec.cinsiyet, rec.iddonem, rec.idyariyil, rec.idprogram, rec.ad,
rec.yad);
end loop;
End;
```

DWFAILEDSTUDENTS FACT TABLE

```
CREATE TABLE GUZIN.DWFAILEDSTUDENTS
(
  IDYARIYIL NUMBER(10,0),
```

```
  IDDONEM NUMBER(10,0),
  IDDONEMDERS NUMBER(10,0),
  IDDERSNOTU NUMBER(10,0),
  IDDERSKAYIDI NUMBER(10,0),
  NOT_HARF VARCHAR2(2),
  AD VARCHAR2(100),
  IDPROGRAM NUMBER(10,0),
  IDDONEMTIP NUMBER(10,0),
  TIP VARCHAR2(15),
  IDBIRIM NUMBER(10,0),
  ONAYLI NUMBER(1,0)
)
Begin
for rec in
( select
SYSTEM.DERSNOTU.iddersnotu,
SYSTEM.DERSNOTU.idderskayidi,
SYSTEM.DERSNOTU.not_harf,
SYSTEM.DERSKAYIDI.idbirim,
SYSTEM.DERSKAYIDI.onayli,
SYSTEM.section.iddonem_ders,
SYSTEM.DONEMDERS.iddonemders,
SYSTEM.DONEMDERS.idprogram,
SYSTEM.DONEM.iddonem,
SYSTEM.DONEM.idyariyil,
SYSTEM.DONEM.iddonemtip,
SYSTEM.DONEMTIP.tip,
SYSTEM.PROGRAM.ad
from SYSTEM.DERSNOTU, SYSTEM.DERSKAYIDI, SYSTEM.section,
SYSTEM.DONEMDERS, SYSTEM.DONEM, SYSTEM.YARIYIL,
SYSTEM.DONEMTIP, SYSTEM.PROGRAM
where
SYSTEM.dersnotu.idderskayidi = SYSTEM.derskayidi.idderskayidi
and SYSTEM.DERSKAYIDI.idbirim = SYSTEM.section.idbirim
and SYSTEM.section.iddonem_ders = SYSTEM.DONEMDERS.iddonemders
and SYSTEM.DONEMDERS.iddonem = SYSTEM.DONEM.iddonem
and SYSTEM.DONEM.idyariyil = SYSTEM.YARIYIL.idyariyil
and SYSTEM.DONEM.iddonemtip = SYSTEM.DONEMTIP.iddonemtip
and SYSTEM.DONEMDERS.idprogram = SYSTEM.PROGRAM.idprogram
and SYSTEM.DERSKAYIDI.onayli = '1'
) loop
Insert into GUZIN.DWFAILEDSTUDENTS (iddonem, idyariyil, iddonemders,
iddersnotu,
idderskayidi, not_harf, ad, idprogram, iddonemtip, tip, idbirim)
values (rec.iddonem, rec.idyariyil, rec.iddonemders, rec.iddersnotu,
rec.idderskayidi, rec.not_harf, rec.ad, rec.idprogram, rec.iddonemtip,
rec.tip, rec.idbirim); end loop; End;
```

APPENDIX E

Extract, Transform, and Loading with PHP

```
<?php
  echo 'Yil : '.$yillar.'<br>';
  $db="mscthesis";
  $c1 = oci_connect("SYSTEM", "guzin", $db);
  $c2 = oci_connect("SYSTEM", "guzin", $db);
  $dwparttimequery="select SYSTEM.SECTION.idbirim,
SYSTEM.DONEM.iddonem, SYSTEM.YARIYIL.idyariyil,
SYSTEM.DONEMDERS.iddonemders, SYSTEM.HOCA.idhoca,
SYSTEM.UNVAN.idunvan, SYSTEM.PROGRAM.idprogram,
SYSTEM.HOCA.parttime";
  $dwparttimequery=$dwparttimequery." from SYSTEM.program, SYSTEM.donem,
SYSTEM.donemders, SYSTEM.hoca, SYSTEM.section, SYSTEM.unvan,
SYSTEM.yariyil";
  $dwparttimequery=$dwparttimequery." where SYSTEM.donem.iddonem =
SYSTEM.donemders.iddonem and SYSTEM.donemders.iddonemders =
SYSTEM.section.iddonem_ders and SYSTEM.hoca.idhoca =
SYSTEM.section.idhoca and SYSTEM.hoca.idunvan = SYSTEM.unvan.idunvan
and SYSTEM.program.idprogram = SYSTEM.donemders.idprogram and
SYSTEM.yariyil.idyariyil = SYSTEM.donem.idyariyil and
SYSTEM.donem.idyariyil = '".$yillar."'";
  $dwparttime = oci_parse($c1, $dwparttimequery);
  oci_execute($dwparttime, OCI_DEFAULT);
  echo '<table>';
    echo '<tr>';
      echo '<td>';
        echo 'andnbsp;';
      echo '</td>';
      echo '<td>';
        echo 'idbirim';
      echo '</td>';
      echo '<td>';
        echo 'iddonem';
      echo '</td>';
      echo '<td>';
        echo 'idyariyil';
      echo '</td>';
```

```
echo '<td>';
      echo 'iddonemders';
    echo '</td>';
    echo '<td>';
      echo 'idhoca';
    echo '</td>';
    echo '<td>';
      echo 'idunvan';
    echo '</td>';
    echo '<td>';
      echo 'idprogram';
    echo '</td>';
    echo '<td>';
      echo 'parttime';
    echo '</td>';
  echo '</tr>';
  $i=0;
 while (oci_fetch($dwparttime)) {
  echo '<tr>';
    echo '<td>';
      echo $i;
    echo '</td>';
    echo '<td>';
      echo oci_result($dwparttime, "IDBIRIM");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwparttime, "IDDONEM");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwparttime, "IDYARIYIL");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwparttime, "IDDONEMDERS");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwparttime, "IDHOCA");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwparttime, "IDUNVAN");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwparttime, "IDPROGRAM");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwparttime, "PARTTIME");
    echo '</td>';
  echo '</tr>';
```

```
  $dwparttimeinsertquery[$i]="insert into GUZIN.DWPARTTIME (IDDONEM,
IDDONEMDERS, IDHOCA, PARTTIME, IDPROGRAM, IDBIRIM, IDUNVAN,
IDYARIYIL) values ('".oci_result($dwparttime, "IDDONEM")."',
'".oci_result($dwparttime, "IDDONEMDERS")."', '".oci_result($dwparttime,
"IDHOCA")."', '".oci_result($dwparttime, "PARTTIME")."',
'".oci_result($dwparttime, "IDPROGRAM")."', '".oci_result($dwparttime,
"IDBIRIM")."', '".oci_result($dwparttime, "IDUNVAN")."',
'".oci_result($dwparttime, "IDYARIYIL")."')";
    $dwparttimeinsert = oci_parse($c2, $dwparttimeinsertquery[$i]);
    oci_execute($dwparttimeinsert, OCI_DEFAULT);
    $i=$i+1;
  }
  echo '</table>';
  echo '<br>';

  $dwprogtitlequery="select SYSTEM.SECTION.idbirim,
SYSTEM.DONEM.iddonem, SYSTEM.YARIYIL.idyariyil,
SYSTEM.DONEMDERS.iddonemders, SYSTEM.HOCA.idhoca,
SYSTEM.UNVAN.idunvan, SYSTEM.PROGRAM.idprogram,
SYSTEM.HOCA.parttime, SYSTEM.PROGRAM.ad, SYSTEM.YARIYIL.yad,
SYSTEM.UNVAN.unvan ";
  $dwprogtitlequery=$dwprogtitlequery." from SYSTEM.program, SYSTEM.donem,
SYSTEM.donemders, SYSTEM.hoca, SYSTEM.section, SYSTEM.unvan,
SYSTEM.yariyil";
  $dwprogtitlequery=$dwprogtitlequery." where SYSTEM.donem.iddonem =
SYSTEM.donemders.iddonem and SYSTEM.donemders.iddonemders =
SYSTEM.section.iddonem_ders and SYSTEM.hoca.idhoca =
SYSTEM.section.idhoca and SYSTEM.hoca.idunvan = SYSTEM.unvan.idunvan
and SYSTEM.program.idprogram = SYSTEM.donemders.idprogram and
SYSTEM.yariyil.idyariyil = SYSTEM.donem.idyariyil and
SYSTEM.donem.idyariyil = '".$yillar."'";
  $dwprogtitle = oci_parse($c1, $dwprogtitlequery);
  oci_execute($dwprogtitle, OCI_DEFAULT);

   echo '<table>';
    echo '<tr>';
      echo '<td>';
        echo ' ';
      echo '</td>';
      echo '<td>';
        echo 'idbirim';
      echo '</td>';
      echo '<td>';
        echo 'iddonem';
      echo '</td>';
      echo '<td>';
        echo 'idyariyil';
      echo '</td>';
      echo '<td>';
        echo 'iddonemders';
```

```
    echo '</td>';
    echo '<td>';
      echo 'idhoca';
    echo '</td>';
    echo '<td>';
      echo 'idunvan';
    echo '</td>';
    echo '<td>';
      echo 'idprogram';
    echo '</td>';
    echo '<td>';
      echo 'parttime';
    echo '</td>';
    echo '<td>';
      echo 'ad';
    echo '</td>';
      echo '<td>';
      echo 'yad';
    echo '</td>';
      echo '<td>';
      echo 'unvan';
    echo '</td>';
  echo '</tr>';
  $i=0;
 while (oci_fetch($dwprogtitle)) {
  echo '<tr>';
    echo '<td>';
      echo $i;
    echo '</td>';
    echo '<td>';
      echo oci_result($dwprogtitle, "IDBIRIM");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwprogtitle, "IDDONEM");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwprogtitle, "IDYARIYIL");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwprogtitle, "IDDONEMDERS");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwprogtitle, "IDHOCA");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwprogtitle, "IDUNVAN");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwprogtitle, "IDPROGRAM");
    echo '</td>';
```

```
      echo '<td>';
        echo oci_result($dwprogtitle, "PARTTIME");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwprogtitle, "AD");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwprogtitle, "YAD");
      echo '</td>';

      echo '<td>';
        echo oci_result($dwprogtitle, "UNVAN");
      echo '</td>';
    echo '</tr>';
    $dwprogtitleinsertquery[$i]="insert into GUZIN.DWPROGTITLE (IDDONEM,
IDDONEMDERS, IDHOCA, PARTTIME, IDPROGRAM, IDBIRIM, IDUNVAN,
IDYARIYIL, AD, YAD, UNVAN) values ('".oci_result($dwprogtitle,
"IDDONEM")."', '".oci_result($dwprogtitle, "IDDONEMDERS")."',
'".oci_result($dwprogtitle, "IDHOCA")."', '".oci_result($dwprogtitle,
"PARTTIME")."', '".oci_result($dwprogtitle, "IDPROGRAM")."',
'".oci_result($dwprogtitle, "IDBIRIM")."', '".oci_result($dwprogtitle,
"IDUNVAN")."', '".oci_result($dwprogtitle, "IDYARIYIL")."', '";
    $degis=str_replace("'","''",oci_result($dwprogtitle, "AD"));
    $dwprogtitleinsertquery[$i]=$dwprogtitleinsertquery[$i].$degis."',
'".oci_result($dwprogtitle, "YAD")."', '".oci_result($dwprogtitle, "UNVAN")."')";
    $dwprogtitleinsert = oci_parse($c2, $dwprogtitleinsertquery[$i]);
    oci_execute($dwprogtitleinsert, OCI_DEFAULT);
    $i=$i+1;
  }
  echo '</table>';
  echo '<br>';
  for ($i=0;$i<count($dwprogtitleinsertquery);$i++)
  echo $dwprogtitleinsertquery[$i].'<br>';


  $dwinstrcreditquery="select SYSTEM.DERS.idders, SYSTEM.DERS.kredi,
SYSTEM.DERS.lab_saati, SYSTEM.DONEM.iddonem,
SYSTEM.DONEMDERS.iddonemders, SYSTEM.HOCA.idhoca,
SYSTEM.SECTION.idbirim, SYSTEM.YARIYIL.idyariyil,
SYSTEM.PROGRAM.idprogram, SYSTEM.PROGRAM.ad,
SYSTEM.YARIYIL.yad ";
  $dwinstrcreditquery=$dwinstrcreditquery." from SYSTEM.ders,
SYSTEM.program, SYSTEM.donem, SYSTEM.donemders, SYSTEM.hoca,
SYSTEM.section, SYSTEM.yariyil";
  $dwinstrcreditquery=$dwinstrcreditquery." where SYSTEM.donem.iddonem =
SYSTEM.donemders.iddonem and SYSTEM.donemders.iddonemders =
SYSTEM.section.iddonem_ders and SYSTEM.hoca.idhoca =
SYSTEM.section.idhoca and SYSTEM.ders.idders = SYSTEM.donemders.idders
and SYSTEM.program.idprogram = SYSTEM.donemders.idprogram and
```

```
SYSTEM.yariyil.idyariyil = SYSTEM.donem.idyariyil and
SYSTEM.donem.idyariyil= '".$yillar."'";
  $dwinstrcredit = oci_parse($c1, $dwinstrcreditquery);
  oci_execute($dwinstrcredit, OCI_DEFAULT);

   echo '<table>';
    echo '<tr>';
      echo '<td>';
        echo ' ';
      echo '</td>';
      echo '<td>';
        echo 'idders';
      echo '</td>';
      echo '<td>';
        echo 'kredi';
      echo '</td>';
      echo '<td>';
        echo 'lab_saati';
      echo '</td>';
      echo '<td>';
        echo 'iddonem';
      echo '</td>';
      echo '<td>';
        echo 'iddonemders';
      echo '</td>';
      echo '<td>';
        echo 'idhoca';
      echo '</td>';
      echo '<td>';
        echo 'idbirim';
      echo '</td>';
      echo '<td>';
        echo 'idyariyil';
      echo '</td>';
      echo '<td>';
        echo 'idprogram';
      echo '</td>';
        echo '<td>';
        echo 'ad';
      echo '</td>';
        echo '<td>';
        echo 'yad';
      echo '</td>';
    echo '</tr>';
    $i=0;
  while (oci_fetch($dwinstrcredit)) {
    echo '<tr>';
      echo '<td>';
        echo $i;
      echo '</td>';
```

```
      echo '<td>';
        echo oci_result($dwinstrcredit, "IDDERS");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwinstrcredit, "KREDI");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwinstrcredit, "LAB_SAATI");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwinstrcredit, "IDDONEM");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwinstrcredit, "IDDONEMDERS");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwinstrcredit, "IDHOCA");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwinstrcredit, "IDBIRIM");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwinstrcredit, "IDYARIYIL");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwinstrcredit, "IDPROGRAM");
      echo '</td>';
      echo '<td>';
        echo oci_result($dwinstrcredit, "AD");
      echo '</td>';

      echo '<td>';
        echo oci_result($dwinstrcredit, "YAD");
      echo '</td>';
    echo '</tr>';
    $dwinstrcreditinsertquery[$i]="insert into GUZIN.DWINSTRCREDIT (IDDERS,
KREDI, LAB_SAATI, IDDONEM, IDDONEMDERS, IDHOCA, IDBIRIM,
IDYARIYIL, IDPROGRAM, AD, YAD) values ('".oci_result($dwinstrcredit,
"IDDERS")."', '".oci_result($dwinstrcredit, "KREDI")."', '".oci_result($dwinstrcredit,
"LAB_SAATI")."', '".oci_result($dwinstrcredit, "IDDONEM")."',
'".oci_result($dwinstrcredit, "IDDONEMDERS")."', '".oci_result($dwinstrcredit,
"IDHOCA")."', '".oci_result($dwinstrcredit, "IDBIRIM")."',
'".oci_result($dwinstrcredit, "IDYARIYIL")."', '".oci_result($dwinstrcredit,
"IDPROGRAM")."', '";
    $degis=str_replace("'","''",oci_result($dwinstrcredit, "AD"));
    $dwinstrcreditinsertquery[$i]=$dwinstrcreditinsertquery[$i].$degis."',
'".oci_result($dwinstrcredit, "YAD")."')";
    $dwinstrcreditinsert = oci_parse($c2, $dwinstrcreditinsertquery[$i]);
    oci_execute($dwinstrcreditinsert, OCI_DEFAULT);
    $i=$i+1;
```

```
 }
 echo '</table>';
 echo '<br>';
 for ($i=0;$i<count($dwinstrcreditinsertquery);$i++)
 echo $dwinstrcreditinsertquery[$i].'<br>';


 $dwgenderbyprogramquery="select SYSTEM.OGRENCI.idogrenci,
SYSTEM.OGRENCI.cinsiyet, SYSTEM.DONEM.iddonem,
SYSTEM.YARIYIL.idyariyil, SYSTEM.YARIYIL.yad,
SYSTEM.PROGRAM.idprogram, SYSTEM.PROGRAM.ad ";
 $dwgenderbyprogramquery=$dwgenderbyprogramquery." from SYSTEM.program,
SYSTEM.donem, SYSTEM.yariyil, SYSTEM.ogrenci, SYSTEM.bolumogrenci,
SYSTEM.ogrencidonemderskayit ";
 $dwgenderbyprogramquery=$dwgenderbyprogramquery." where
SYSTEM.ogrenci.idogrenci = SYSTEM.bolumogrenci.idogrenci and
SYSTEM.bolumogrenci.idbolumogrenci =
SYSTEM.ogrencidonemderskayit.idbolum_ogrenci and
SYSTEM.bolumogrenci.idprogram = SYSTEM.program.idprogram and
SYSTEM.ogrencidonemderskayit.iddonem = SYSTEM.donem.iddonem and
SYSTEM.donem.idyariyil = SYSTEM.yariyil.idyariyil and
SYSTEM.yariyil.idyariyil = '".$yillar."'";
 $dwgenderbyprogram = oci_parse($c1, $dwgenderbyprogramquery);
 oci_execute($dwgenderbyprogram, OCI_DEFAULT);

  echo '<table>';
   echo '<tr>';
    echo '<td>';
     echo ' ';
    echo '</td>';
    echo '<td>';
     echo 'idyariyil';
    echo '</td>';
    echo '<td>';
     echo 'iddonem';
    echo '</td>';
    echo '<td>';
     echo 'idprogram';
    echo '</td>';
    echo '<td>';
     echo 'idogrenci';
    echo '</td>';
    echo '<td>';
     echo 'cinsiyet';
    echo '</td>';
    echo '<td>';
     echo 'yad';
    echo '</td>';
    echo '<td>';
     echo 'ad';
```

```
    echo '</td>';
    echo '</tr>';
   $i=0;
  while (oci_fetch($dwgenderbyprogram)) {
   echo '<tr>';
    echo '<td>';
     echo $i;
    echo '</td>';
    echo '<td>';
     echo oci_result($dwgenderbyprogram, "IDYARIYIL");
    echo '</td>';
    echo '<td>';
     echo oci_result($dwgenderbyprogram, "IDDONEM");
    echo '</td>';
    echo '<td>';
     echo oci_result($dwgenderbyprogram, "IDPROGRAM");
    echo '</td>';
    echo '<td>';
     echo oci_result($dwgenderbyprogram, "IDOGRENCI");
    echo '</td>';
    echo '<td>';
     echo oci_result($dwgenderbyprogram, "CINSIYET");
    echo '</td>';
    echo '<td>';
     echo oci_result($dwgenderbyprogram, "YAD");
    echo '</td>';
    echo '<td>';
     echo oci_result($dwgenderbyprogram, "AD");
    echo '</td>';
   echo '</tr>';
   $dwgenderbyprograminsertquery[$i]="insert into

GUZIN.DWGENDERBYPROGRAM (IDYARIYIL, IDDONEM, IDPROGRAM,
IDOGRENCI, CINSIYET, YAD, AD) values ('".oci_result($dwgenderbyprogram,
"IDYARIYIL")."', '".oci_result($dwgenderbyprogram, "IDDONEM")."',
'".oci_result($dwgenderbyprogram, "IDPROGRAM")."',
'".oci_result($dwgenderbyprogram, "IDOGRENCI")."',
'".oci_result($dwgenderbyprogram, "CINSIYET")."',
'".oci_result($dwgenderbyprogram, "YAD")."', '";
   $degis=str_replace("'","''",oci_result($dwgenderbyprogram, "AD"));

$dwgenderbyprograminsertquery[$i]=$dwgenderbyprograminsertquery[$i].$degis."')
";
   $dwgenderbyprograminsert = oci_parse($c2,
$dwgenderbyprograminsertquery[$i]);
   oci_execute($dwgenderbyprograminsert, OCI_DEFAULT);
   $i=$i+1;
  }
 echo '</table>';
 echo '<br>';
```

```
 for ($i=0;$i<count($dwgenderbyprograminsertquery);$i++)
 echo $dwgenderbyprograminsertquery[$i].'<br>';


 $dwfailedstudentsquery="select SYSTEM.DERSNOTU.iddersnotu,
SYSTEM.DERSNOTU.idderskayidi, SYSTEM.DERSNOTU.not_harf,
SYSTEM.DERSKAYIDI.idbirim, SYSTEM.DERSKAYIDI.onayli,
SYSTEM.section.iddonem_ders, SYSTEM.DONEMDERS.iddonemders,
SYSTEM.DONEMDERS.idprogram, SYSTEM.DONEM.iddonem,
SYSTEM.DONEM.idyariyil, SYSTEM.DONEM.iddonemtip,
SYSTEM.DONEMTIP.tip, SYSTEM.PROGRAM.ad ";
 $dwfailedstudentsquery=$dwfailedstudentsquery." from SYSTEM.dersnotu,
SYSTEM.derskayidi, SYSTEM.section, SYSTEM.donemders, SYSTEM.donem,
SYSTEM.yariyil, SYSTEM.donemtip, SYSTEM.program ";
 $dwfailedstudentsquery=$dwfailedstudentsquery." where
SYSTEM.dersnotu.idderskayidi = SYSTEM.derskayidi.idderskayidi and
SYSTEM.DERSKAYIDI.idbirim = SYSTEM.section.idbirim and
SYSTEM.section.iddonem_ders = SYSTEM.DONEMDERS.iddonemders and
SYSTEM.DONEMDERS.iddonem = SYSTEM.DONEM.iddonem and
SYSTEM.DONEM.idyariyil = SYSTEM.YARIYIL.idyariyil and
SYSTEM.DONEM.iddonemtip = SYSTEM.DONEMTIP.iddonemtip and
SYSTEM.DONEMDERS.idprogram = SYSTEM.PROGRAM.idprogram and
SYSTEM.DERSKAYIDI.onayli = '1' and SYSTEM.YARIYIL.IDYARIYIL =
'".$yillar."'";
 $dwfailedstudents = oci_parse($c1, $dwfailedstudentsquery);
 oci_execute($dwfailedstudents, OCI_DEFAULT);

  echo '<table>';
   echo '<tr>';
     echo '<td>';
       echo ' ';
     echo '</td>';
     echo '<td>';
       echo 'idyariyil';
     echo '</td>';
     echo '<td>';
       echo 'iddonem';
     echo '</td>';
     echo '<td>';
       echo 'iddonemders';
     echo '</td>';
     echo '<td>';
       echo 'iddersnotu';
     echo '</td>';
     echo '<td>';
       echo 'idderskayidi';
     echo '</td>';
     echo '<td>';
       echo 'not_harf';
     echo '</td>';
```

```
    echo '<td>';
      echo 'ad';
    echo '</td>';
    echo '<td>';
      echo 'idprogram';
    echo '</td>';
    echo '<td>';
      echo 'iddonemtip';
    echo '</td>';
      echo '<td>';
      echo 'tip';
    echo '</td>';
      echo '<td>';
      echo 'idbirim';
    echo '</td>';
      echo '<td>';
      echo 'onayli';
    echo '</td>';
      echo '</tr>';
  $i=0;
 while (oci_fetch($dwfailedstudents)) {
  echo '<tr>';
    echo '<td>';
      echo $i;
    echo '</td>';
    echo '<td>';
      echo oci_result($dwfailedstudents, "IDYARIYIL");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwfailedstudents, "IDDONEM");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwfailedstudents, "IDDONEMDERS");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwfailedstudents, "IDDERSNOTU");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwfailedstudents, "IDDERSKAYIDI");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwfailedstudents, "NOT_HARF");
    echo '</td>';
    echo '<td>';
      echo oci_result($dwfailedstudents, "AD");
    echo '</td>';
         echo '<td>';
      echo oci_result($dwfailedstudents, "IDPROGRAM");
    echo '</td>';
         echo '<td>';
```

```
      echo oci_result($dwfailedstudents, "IDDONEMTIP");
     echo '</td>';
          echo '<td>';
      echo oci_result($dwfailedstudents, "TIP");
     echo '</td>';
          echo '<td>';
      echo oci_result($dwfailedstudents, "IDBIRIM");
     echo '</td>';
          echo '<td>';
      echo oci_result($dwfailedstudents, "ONAYLI");
     echo '</td>';
    echo '</tr>';
  $dwfailedstudentsinsertquery[$i]="insert into GUZIN.DWFAILEDSTUDENTS
(IDYARIYIL, IDDONEM, IDDONEMDERS, IDDERSNOTU, IDDERSKAYIDI,
NOT_HARF, AD, IDPROGRAM, IDDONEMTIP, TIP, IDBIRIM, ONAYLI)
values ('".oci_result($dwfailedstudents, "IDYARIYIL")."',
'".oci_result($dwfailedstudents, "IDDONEM")."', '".oci_result($dwfailedstudents,
"IDDONEMDERS")."', '".oci_result($dwfailedstudents, "IDDERSNOTU")."',
'".oci_result($dwfailedstudents, "IDDERSKAYIDI")."',
'".oci_result($dwfailedstudents, "NOT_HARF")."', '";
  $degis=str_replace("'","''",oci_result($dwfailedstudents, "AD"));
  $dwfailedstudentsinsertquery[$i]=$dwfailedstudentsinsertquery[$i].$degis."',
'".oci_result($dwfailedstudents, "IDPROGRAM")."', '".oci_result($dwfailedstudents,
"IDDONEMTIP")."', '".oci_result($dwfailedstudents, "TIP")."',
'".oci_result($dwfailedstudents, "IDBIRIM")."', '".oci_result($dwfailedstudents,
"ONAYLI")."')";
  $dwfailedstudentsinsert = oci_parse($c2, $dwfailedstudentsinsertquery[$i]);
   oci_execute($dwfailedstudentsinsert, OCI_DEFAULT);
   $i=$i+1;
 }
 echo '</table>';
 echo '<br>';
 for ($i=0;$i<count($dwfailedstudentsinsertquery);$i++)
 echo $dwfailedstudentsinsertquery[$i].'<br>';

 oci_close($c1);
 oci_commit($c2);
 oci_close($c2);

?>
```

REFERENCES

[1] Turban, E., *Decision support and expert systems: management support systems,* Englewood Cliffs, N.J., Prentice Hall, 1995

[2] Holsapple, C.W and Whinston A.B., *Decision support systems*, New York: West publishing Company, 1996

[3] Mallach, E.G., *Understanding Decision Support Systems and Expert Systems*, Irwin. pp. 428-435, 1994

[4] Breiter, A., & Light, D., "Data for Improvement: Factors for designing effective information systems to support decision-making in schools", *Educational Technology and Society*, vol. 9, no. 3, pp. 206-217, 2006

[5] Breiter, A., and Light, D., "Decision Support Systems in Schools – from Data Collection to Decision Making", *America's Conference on Information Systems* (AMCIS), New York, NY, 2004a

[6] Breiter, A., and Light, D., "Teachers in the "intelligent" school – From data collection to decision-making", Paper presented at the *ED-MEDIA*, Lugano, Switzerland, 2004b

[7] Ackoff, R. L., "From Data to Wisdom", *Journal of Applied Systems Analysis*, vol. 16, pp. 3-9, 1989

[8] Sauter, V. L., *Decision support systems: an applied managerial approach,* New York, John Wiley, 1997

[9] Thomas H. Davenport, *Working Knowledge: How Organizations Manage What They Know*, Harvard Business School Press, 1997

[10] Prof. Dr. Aidong Zhang, Lecture Notes on Data Mining http://www.cse.buffalo.edu/pub/WWW/faculty/azhang/, Last attempt 20th, August, 2007

[11] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers, 2001

[12] Inmon, W. H., *What is a Data Warehouse?*, Sunnyvale,CA: PRISM Solutions, Inc. Tech Topic 1, no. 1, p. 1.,1992

[13] Romm M., *Introduction to Data Warehousing*, San Diego SQL User Group

[14] Thomas Conolly and Caroly Begg., *Database Systems*, 3th Edition, Addison-Wesley, 2002

[15] W.H.Inmon, *Building the Data Warehouse,* 3rd Edition, John Wiley, Chap.2, p. 36, 2002

[16] W. H. Inmon and R. D. Hackathorn, *Using the Data Warehouse*, John Viley and Sons, Chap 11, 1994

[17] Surajit Chaudhuri, Umeshwar Dayal, Venkatesh Ganti, "Database Technology for Decision Support Systems", *IEEE Computer*, vol: 34, no: 12, p. 48-55, December 2001

[18] Arun Sen and Atish P. Sinha, "A Comparison of Data Warehousing Methodologies", *Communications of the ACM*, vol. 48. no.3, March 2005

[19] Tannenbaum, Adrienne, *Metadata solutions: using metamodels, repositories, XML, and enterprise portals to generate information on demand,* Addison-Wesley, Boston, Mass, p. 461, 2002

[20] Paul Gray and Hugh J. Watson, "Present and Future Directions in Data Warehousing", *The DATA BASE for Advances in Information Systems-Summer* vol.29, no.3, 1998

[21] Ananth Srinivasan, David Sundaram and Joseph Davis, *Implementing Decision Support Systems*, McGraw-Hill, 2000

[22] Chuck Ballard, Dirk Herreman, Don Schau, Rhonda Bell,Eunsaeng Kim, Ann Valencic, "Data Modeling Techniques for Data Warehousing", *International Technical Support Organization,* http://www.redbooks.ibm.com

[23] Kimball, Ralph, *The Data Warehouse Toolkit,* New York: John Wiley and Sons, 1996

[24] Kimball, Ralph, Reeves, Laura, Ross, Margy, and Thornthwaite Warren, *The Data Warehouse Lifecycle Toolkit,* New York, John Wiley and Sons, Inc. 1998

[25] H.Galhardas, "Declarative Data Cleaning Model, Language and Algorithms", *Proc. VLDB Conf.*, Morgan Kaufmann, San Francisco, pp.371-380, 2001

[26] *Data Analysis and Assessment Tools*, Technology and Learning, 10536728, vol.25, Issue 11, Jun 2005

[27] Microsoft, "*Data Driven Decision Making in Higher Education, Improving decision making across the campus*", February 2004

[28] David Friend, "An introduction to OLAP: An explanation of multidimensional database terminology and technology", *OLAP Forum*, Orlando, Florida, February 1995

[29] Andreas Breiter, Daniel Light, "Data for School Improvement: Factors for designing effective information systems to support decision-making in schools", *Educational Technology and Society*, vol.9, no.3, p. 206-217, 2006

[30] R. Remes, "Learning Management System", *WDS'05 Proceedings of Contributed Papers*, Part I, pp. 207–212, 2005

[31] Gerri Furlow, "The Case for Building a Data Warehouse", *IT Professional*, vol. 3, no. 4, pp. 31-34, Jul/Aug, 2001

[32] Thelma G.Lussier and Beverley Doern, "Improving Decision Making Support by Building a Data Warehouse", *5th Annual Conference of The Canadian Institutional Research and Planning Association*, Calgary, Alberta, October 6-8, 1996

[33] Scott Thorne, *The Data Warehouse: Keeping it Simple*, Educause Quarterly, no.3, 2000

[34] Joanne Ingham, "Data Warehousing: A Tool for the Outcomes Assessment Process", *IEEE Transactions on Education*, vol.43, no.2, May 2000

[35] Dervis Z. Deniz and İbrahim Ersan, "An Academic Decision-Support System Based on Academic Performance Evaluation for Student and Program Assessment", *Int. J. Engng Ed.* vol.18, no.2, pp.236-244, Printed in Great Britain, 2002

[36] Levent Bayoğlu, *Management Information System and Data Warehousing*, MSc Thesis, June 2000