

**THE DEVELOPMENT OF MAHIDOL UNIVERSITY STUDENT
DATA MART SYSTEM (MU – SDM)**



**A RESEARCH PROJECT SUBMITTED IN PARTIAL
FULFILLMENT OF THE REQUIREMENTS FOR
THE DEGREE OF MASTER OF SCIENCE
(COMPUTER SCIENCE)
FACULTY OF GRADUATE STUDIES
MAHIDOL UNIVERSITY
2006**

COPYRIGHT OF MAHIDOL UNIVERSITY

Research Project
Entitled

**THE DEVELOPMENT OF MAHIDOL UNIVERSITY STUDENT
DATA MART SYSTEM (MU – SDM)**



Siriporn Rojanakosol.....

Miss Siriporn Rojanakosol
Candidate

Jarensri L. Mitranont.....

Assoc. Prof. Jarensri L. Mitranont, Ph.D.
Major-Advisor

Thanwadee Sunetnanta.....

Asst. Prof. Thanwadee Sunetnanta, Ph.D.
Co-Advisor

M.R. Jisnuson Svasti.....

Prof. Dr. M.R. Jisnuson Svasti, Ph.D.
Dean
Faculty of Graduate Studies

Supachai Tangwongsan.....

Assoc. Prof. Supachai Tangwongsan, Ph.D.
Chair
Master of Science Programme in
Computer Science
Faculty of Science

Research Project
Entitled

**THE DEVELOPMENT OF MAHIDOL UNIVERSITY STUDENT
DATA MART SYSTEM(MU – SDM)**

was submitted to the Faculty of Graduate Studies, Mahidol University
for the degree of Master of Science (Computer Science)

on
November 3, 2006



Siriporn Rojanakosol

Miss Siriporn Rojanakosol
Candidate

Yongyuth Permpoontanalarp

Asst. Prof. Yongyuth Permpoontanalarp, Ph.D.
Chair

Jarensri L. Mitranont

Assoc. Prof. Jarensri L. Mitranont, Ph.D.
Member

Thanwadee Sunetnanta

Asst. Prof. Thanwadee Sunetnanta, Ph.D.
Member

Jisnuson Svasti

Prof. Dr. M.R. Jisnuson Svasti, Ph.D.
Dean
Faculty of Graduate Studies
Mahidol University

Amaret Bhumiratana

Prof. Amaret Bhumiratana, Ph.D.
Dean
Faculty of Science
Mahidol University

ACKNOWLEDGEMENTS

The success of this research project is dedicated to the extensive support and assistance from my major advisor, Assoc. Prof. Jarernsri L. Mitrpanont, my co-advisors, Asst. Prof. Thanwadee Sunetnanta. I thank them deeply for their many suggestions, valuable feedback and guidance in this research project.

I would like to thank all the lecturers of the Department of Computer Science, Faculty of Science for their knowledge and education. I thank my best friends in the Computer Science Program and Mahidol University Computing Center for their kind support and encouragement.

Finally, I am grateful to my family for their love, encouragement and tender care. This research project is sincerely dedicated to my father, my mother and all the teachers who had taught me well throughout my childhood.

Siriporn Rojanakosol

THE DEVELOPMENT OF MAHIDOL UNIVERSITY STUDENT DATA MART SYSTEM(MU – SDM)

SIRIPORN ROJANAKOSOL 4537079 SCCS/M

M.Sc. (COMPUTER SCIENCE)

RESEARCH PROJECT ADVISOR : JARERNSRI L. MITR PANONT Ph.D.,
THANWADEE SUNETNANTA Ph.D.**ABSTRACT**

The Mahidol University Student Information System has been developed to support daily operation of academic departments. A large amount of data of the system has been established, but the university executives cannot fully utilize the detail and many structures of data for analysis, planning and making strategic decisions. The system cannot meet executive needs because the database is designed to support daily operations.

To solve this problem, an information technology called Data Mart was introduced to integrate, reconcile and summarize data from multiple operational systems into a common concept to make it available for decision support. In addition, the data mart refreshment process is designed to synchronize data mart with the underlying sources. Automatically propagating only changed data from operational sources into data mart is a better practice for data mart refreshment. It keeps historical information and is not time-consuming. The change monitoring is applied to detect changed data and reduce time and effort for data mart refreshment.

The MU-SDM prototype was developed to support data mart administrators in creating and maintaining a student data mart system and to provide a data analysis tool for university executives to utilize information. The tool is flexible, as well as easy to use and understand. It provides the necessary functions which allow university executives to view necessary academic information. Many essential cubes are designed to provide useful information and a user interface for presenting multidimensional information and graphs to users. Users can perform multidimensional operations, such as roll-up, drill-down, slice-dice and pivoting to view data from various perspectives.

The MU-SDM prototype was developed by using Microsoft SQL Server 2000 Analysis Services to create and maintain multidimensional data structures. The system has been tested on its functionalities and has been evaluated by using some interesting scenarios that usually happen in a university.

KEY WORDS : DATA WAREHOUSE / DATA MART / MULTIDIMENSION / OLAP / ETL

189 pp.

การพัฒนาระบบดาตามาร์ทนักศึกษามหาวิทยาลัยมหิดล
(THE DEVELOPMENT OF MAHIDOL UNIVERSITY STUDENT DATA
MART SYSTEM)

ศิริพร โรจนโกศล 4537079 SCCS/M

วท.ม. (วิทยาการคอมพิวเตอร์)

คณะกรรมการควบคุมโครงการวิจัย : เจริญศรี มิตรภานนท์ Ph.D., ชันวดี สุเนตพันธ์ Ph.D.

บทคัดย่อ

ระบบฐานข้อมูลนักศึกษา มหาวิทยาลัยมหิดลถูกพัฒนาขึ้นเพื่อสนับสนุนการปฏิบัติงานแก่ฝ่ายการศึกษาในมหาวิทยาลัยมหิดล มีข้อมูลจำนวนมากที่เกิดจากระบบ แต่ผู้บริหารไม่อาจใช้ประโยชน์จากข้อมูลเหล่านั้นได้เต็มที่ เนื่องจากการที่มีโครงสร้างหลากหลายไม่เหมาะกับการวิเคราะห์ วางแผน และการตัดสินใจเชิงกลยุทธ์ได้ เนื่องจากฐานข้อมูลถูกออกแบบให้เหมาะสมกับการปฏิบัติงานประจำวันเท่านั้น

เพื่อแก้ปัญหานี้ เทคโนโลยีดาตามาร์ทถูกนำมาใช้เพื่อรวบรวมข้อมูล ทำให้เหมาะสม และสรุปข้อมูลที่แตกต่างกันจากหลายระบบให้อยู่ในรูปแบบสารสนเทศ เพื่อสนับสนุนการตัดสินใจ นอกจากนี้แล้ว กระบวนการปรับปรุงดาตามาร์ทได้ถูกออกแบบขึ้น เพื่อให้ดาตามาร์ทสอดคล้องกับแหล่งข้อมูลซึ่งมีการเปลี่ยนแปลง การส่งต่อเฉพาะข้อมูลที่เปลี่ยนแปลงจากแหล่งข้อมูลไปยังดาตามาร์ทอย่างอัตโนมัติเป็นวิธีที่เหมาะสมสำหรับการปรับปรุงดาตามาร์ท เนื่องจากยังคงรักษาข้อมูลสารสนเทศจากอดีตในดาตามาร์ทไว้ และไม่ต้องใช้เวลามากในการประมวลผล เทคนิคในการตรวจจับข้อมูลที่มีการเปลี่ยนแปลงจึงถูกนำมาใช้เพื่อลดเวลาในการปรับปรุงดาตามาร์ท

MU-SDM เป็นระบบต้นแบบที่ถูกพัฒนาขึ้นเพื่อสนับสนุนผู้ดูแลดาตามาร์ทในการสร้างและบำรุงรักษาดาตามาร์ทนักศึกษา และมีเครื่องมือสำหรับการวิเคราะห์ให้แก่ผู้บริหารมหาวิทยาลัยในการใช้ประโยชน์ข้อมูล เครื่องมือนี้มีความยืดหยุ่นง่ายต่อการเข้าใจและใช้งาน แบบจำลองมีฟังก์ชันที่จำเป็นสำหรับผู้บริหารมหาวิทยาลัยในการดูข้อมูล มี Cubes ข้อมูลที่ให้สารสนเทศที่เป็นประโยชน์และนำเสนอข้อมูลในหลายมิติ พร้อมแสดงกราฟแก่ผู้ใช้งาน ผู้ใช้สามารถใช้ฟังก์ชันหลายมิติ ได้แก่ Roll-up, Drill-down, Slice-Dice และ Pivoting เพื่อดูข้อมูลในหลายมุมมอง

MU-SDM ถูกพัฒนาขึ้นโดยใช้ Microsoft SQL Server 2000 Analysis Services เป็นเครื่องมือในการสร้างและบำรุงรักษาโครงสร้างหลายมิติ ระบบได้ผ่านการทดสอบและประเมิน โดยการจำลองสถานการณ์ที่น่าสนใจที่มักเกิดขึ้นในมหาวิทยาลัย

189 หน้า.

CONTENTS

	Page
ACKNOWLEDGEMENTS	iii
ABSTRACT	iv
LIST OF TABLES	ix
LIST OF FIGURES	x
CHAPTER 1 INTRODUCTION	1
1.1 Problem Statement	2
1.2 Research Objectives	3
1.3 Scope of the Research	3
1.4 Chapter Organization	4
CHAPTER 2 LITERATURE REVIEW	5
2.1 Overview of Data Warehouse Concept	5
2.1.1 Data Warehouse Architecture	6
2.1.2 Dimensional Model	7
2.1.3 OLAP	9
2.1.4 Data Warehouse Processes	13
2.2 A Survey of Data warehouse Project	16
2.2.1 Existing University Data Warehouse Project	16
2.2.1.1 Stanford University (WHIPS Project)	16
2.2.1.2 The University of Washington Data Warehousing Program	18
2.2.1.3 Georgetown University	19
2.2.1.4 University of Pennsylvania Data Warehouse	21
2.2.1.5 University of Maryland Data Warehouse	21
2.2.1.6 Arizona State University Data Warehouse	22

CONTENTS (CONT.)

	Page
2.2.1.7 A Comparative Survey of Six Existing University Data Warehouse Projects	22
2.2.2 Data Warehouse Framework.	26
2.2.2.1 Paulraj's Framework	26
2.2.2.2 Mattison's Framework	28
2.2.2.3 Singh's Framework	29
2.2.2.4 Comparative Survey of Three Existing Data Warehouse Frameworks	29
2.3 Related Techniques	30
2.3.1 Change Monitoring	30
2.3.2 Trigger	32
2.3.3 Web-based Architecture	33
2.3.3.1 Web Programming	33
2.3.3.2 Data Access Technology	35
CHAPTER 3 DESIGN APPROACH OF THE MU STUDENT DATA MART SYSTEM	37
3.1 A Current System and Environment of the MU Student Information System	37
3.2 MU Student Data Mart System Approach	39
3.2.1 The Architecture of MU Student Data Mart	39
3.2.2 Design of Development Process of MU Student Data Mart	42
3.2.2.1 Extraction	44
3.2.2.2 Cleansing and Transformation	51
3.2.2.3 Loading	55
3.2.3 Design of Data Refreshment Process	56
3.2.4 Design of MU Student Data Cube	58
3.3 Structure of System Modules	58

CONTENTS (CONT.)

	Page
3.4 System Processes	59
3.4.1 ETL Process	60
3.4.2 Information Delivery	69
3.5 Database Design for MU Student Data Mart	70
3.5.1 Data Source of MU Student Data Mart	71
3.5.2 Data on MU Student Data Mart	76
3.5.3 Table Structure	81
CHAPTER 4 IMPLEMENTATION	91
4.1 System Environment	91
4.2 Implementation of MU-SDM	92
4.3 Cube Creation	98
4.4 Cubes of MU-SDM	99
CHAPTER 5 SYSTEM TESTING AND RESULTS	113
5.1 Two Main Functions of MU-SDM Prototype	113
5.1.1 ETL	113
5.1.2 Information Delivery	121
5.2 Testing Scenarios	160
CHAPTER 6 DISCUSSION AND CONCLUSION	180
6.1 Discussion and Conclusion	180
6.2 Difficulties and Lesson Learnt	183
6.3 Suggestions for Future Work	184
REFERENCES	185
BIOGRAPHY	189

LIST OF TABLES

		Page
Table 2.1	A Comparison of Components, Tools, Platform and Subject Area of University Data Warehouse Projects.	24
Table 3.1	Use Case Specifications of MU Student Data Mart	43
Table 3.2	Use Case Specification of Extraction	49
Table 3.3	Use Case Specification of Cleansing & Transformation	53
Table 3.4	Student Information System	72
Table 3.5	MU Student Data Mart	78
Table 4.1	Dimension Hierarchy of Student Cube	100
Table 4.2	Dimension Hierarchy of Student Graduation Cube	101
Table 4.3	Dimension Hierarchy of Registration Cube	103
Table 4.4	Dimension Hierarchy of Registration Subject Cube	105
Table 4.5	Dimension Hierarchy of Registration Cube	106
Table 4.6	Dimension Hierarchy of Program Allocation Cube	108
Table 4.7	Dimension Hierarchy of Grading Cube	110
Table 4.8	Dimension Hierarchy of GPA Cube	112

LIST OF FIGURES

	Page
Figure 2.1 Overview of Data Warehouse Infrastructure.	5
Figure 2.2 Data Warehouse Architecture.	7
Figure 2.3 Star Schema	8
Figure 2.4 Snowflake Schema	9
Figure 2.5 Cube of Sales	9
Figure 2.6 Three-Tier Architecture of ROLAP Model	10
Figure 2.7 Three-Tier Architecture of MOLAP Model	11
Figure 2.8 The Star Schema of Sales	11
Figure 2.9 Drill-Down and Roll-Up	12
Figure 2.10 Sales of Australia	12
Figure 2.11 Sales of Australia and Asia Region	13
Figure 2.12 Pivoting of Sales.	13
Figure 2.13 Data Warehouse Architecture	14
Figure 2.14 The WHIPS System Architecture for Warehouse Maintenance	18
Figure 2.15 The Architecture of UW Data Warehousing Program	19
Figure 2.16 The Architecture of Georgetown University Enterprise Data Warehouse	21
Figure 2.17 ASU's Data Warehouse Environment	22
Figure 2.18 Paulraj's Data Warehouse Components	26
Figure 2.19 Mattison's Data Warehouse Components	28
Figure 2.20 Singh's Data Warehouse Functions	29
Figure 2.21 Comparison of Three Data Warehouse Existing Framework	30
Figure 3.1 The Architecture of the Mahidol University Student Data Mart Development	40
Figure 3.2 ETL Process	41

LIST OF FIGURES (CONT.)

	Page
Figure 3.3 A High Level Use Case Diagram for MU Student Data Mart	42
Figure 3.4 Defer Extraction	46
Figure 3.5 Triggers and Aggregation Algorithm of Defer Extraction	47
Figure 3.6 Immediate Extraction	47
Figure 3.7 Triggers Algorithm of Immediate Extraction	48
Figure 3.8 A Detailed Use Case Diagram of Extraction	49
Figure 3.9 A Detailed Use Case Diagram of Cleansing & Transformation	53
Figure 3.10 Comparison of Three Existing Frameworks and MU Student Data Mart	56
Figure 3.11 A Structure Chart of Core Modules of MU Student Data Mart	59
Figure 3.12 A Context Diagram of MU Student Data Mart	59
Figure 3.13 A Data Flow Diagram Level 1 of MU Student Data Mart	60
Figure 3.14 A Data Flow Diagram Level 2 of ETL Process	61
Figure 3.15 A Structure Chart of Initial ETL Process	62
Figure 3.16 A Data Flow Diagram Level 3 of initial ETL Process	63
Figure 3.17 A Data Flow Diagram Level 4 of Transformation Process	65
Figure 3.18 A Data Flow Diagram Level 4 of Loading Process.	66
Figure 3.19 A Structure Chart of ETL for Refreshment Process	67
Figure 3.20 A Data Flow Diagram Level 4 of Extraction for Refreshment.	68
Figure 3.21 A Data Flow Diagram Level 5 of Change Extraction Process	69
Figure 3.22 A Structure Chart of Information Delivery	69
Figure 3.23 A Data Flow Diagram Level 2 of Information Delivery Process.	70
Figure 3.24 An ER Diagram of MU Student Information System	71
Figure 3.25 Conceptual Dimensional Schema for Student Data Mart	77
Figure 4.1 3-Tier Architecture MU-SDM	91
Figure 4.2 Student Cube's Schema	99
Figure 4.3 Student Cube Structure	100
Figure 4.4 Student Graduation Cube's Schema	101

LIST OF FIGURES (CONT.)

	Page
Figure 4.5 Student Graduation Cube Structure	102
Figure 4.6 Registration Cube's Schema	103
Figure 4.7 Registration Cube Structure	104
Figure 4.8 Registration Subject Cube's Schema	104
Figure 4.9 Registration Subject Cube Structure	105
Figure 4.10 Allocation Cube's Schema	106
Figure 4.11 Allocation Cube Structure	107
Figure 4.12 Program Allocation Cube's Schema	107
Figure 4.13 Program Allocation Cube Structure	109
Figure 4.14 Grading Cube's Schema	109
Figure 4.15 Grading Cube Structure	111
Figure 4.16 GPA Cube's Schema	111
Figure 4.17 GPA Cube Structure	112
Figure 5.1 Main Screen of ETL Process.	113
Figure 5.2 Tuition Calculation	114
Figure 5.3 Result of Tuition Calculation Process	115
Figure 5.4 Student Status	116
Figure 5.5 Student Status for Semester 3/2548	116
Figure 5.6 Edit Student Status	117
Figure 5.7 Update Student Status	117
Figure 5.8 Initial ETL	118
Figure 5.9 Executing Initial ETL	119
Figure 5.10 Refreshment of ETL	120
Figure 5.11 Executing Refreshment of ETL	120
Figure 5.12 Pivot Table of Student Information	122
Figure 5.13 Student Information Graph with Series in Row	122
Figure 5.14 Student Information Graph with Series in Column	123
Figure 5.15 Apply Filter to Student Information	124

LIST OF FIGURES (CONT.)

	Page
Figure 5.16 GPA with Filter on Gpalevel Dimension	124
Figure 5.17 Drill-Down Student Information in SC Faculty	125
Figure 5.18 Drill-down All Faculties to Display Student Information	126
Figure 5.19 Roll-Up Student Information to View Faculty Level	127
Figure 5.20 Roll-Up Student Information	127
Figure 5.21 Student Information with Faculty in Row	128
Figure 5.22 Student Information with Faculty in Column	128
Figure 5.23 Registration Cube with Payment Type Dimension in Row	129
Figure 5.24 Registration Cube with Payment Type Dimension in Column	129
Figure 5.25 GPA Cube with Program Dimension in Row	130
Figure 5.26 GPA Cube with Program Dimension in Column	130
Figure 5.27 Table Field List of Student Information Cube	131
Figure 5.28 Student Information Pivot Table	132
Figure 5.29 Student Information Graph with Series in Row	132
Figure 5.30 Student Information Graph with Series in Column	133
Figure 5.31 Student Information with Academic Year 2542	133
Figure 5.32 Table Field List of Student Graduation Cube	135
Figure 5.33 Student Graduation Pivot Table	135
Figure 5.34 Student Graduation Graph with Series in Row	136
Figure 5.35 Student Graduation Graph with Series in Column	136
Figure 5.36 Student Graduation Pivot Table with Percentage of Students	137
Figure 5.37 Table Field List of Registration Cube	140
Figure 5.38 Registration Pivot Table	141
Figure 5.39 Registration Graph with Series in Row	141
Figure 5.40 Registration Graph with Series in Column	141
Figure 5.41 Registration Pivot Table	142
Figure 5.42 Table Field List of Subject Cube	142
Figure 5.43 Subject Pivot Table	144

LIST OF FIGURES (CONT.)

	Page
Figure 5.44 Subject Graph with Series in Row	144
Figure 5.45 Subject Graph with Series in Column	144
Figure 5.46 Drill-Down Subject Pivot Table	145
Figure 5.47 Table Field List of Allocation Cube	146
Figure 5.48 Allocation Pivot Table	150
Figure 5.49 Allocation Graph with Series in Row	150
Figure 5.50 Allocation Graph with Series in Column	151
Figure 5.51 Allocation Pivot Table with Percentage of Amount	151
Figure 5.52 Table Field List of Allocation by Program	152
Figure 5.53 Allocation by Program Pivot Table	153
Figure 5.54 Allocation by Program Graph with Series in Row	154
Figure 5.55 Allocation by Program Graph with Series in Column	154
Figure 5.56 Table Field List of Grading Cube	155
Figure 5.57 Grading Pivot Table	156
Figure 5.58 Grading Graph with Series in Row	156
Figure 5.59 Grading Graph with Series in Column	157
Figure 5.60 Grading Pivot Table with Percentage of Students	157
Figure 5.61 Table Field List of GPA Cube	158
Figure 5.62 GPA Pivot Table	159
Figure 5.63 GPA Graph with Series in Row	159
Figure 5.64 GPA Graph with Series in Column	159
Figure 5.65 GPA Pivot Table with Percentage of Students	160
Figure 5.66 Student Information Pivot Table	161
Figure 5.67 Student Information Pivot Table at Program Level	162
Figure 5.68 Student Graduation Pivot Table	163
Figure 5.69 Drill-Down Student Graduation Pivot Table	163
Figure 5.70 Student Graduation Pivot Table with Percentage of Students	164
Figure 5.71 Filter on Semester Dimension	165

LIST OF FIGURES (CONT.)

	Page
Figure 5.72 Registration Pivot Table of 1/2548 Semester	165
Figure 5.73 Comparison of Registration Income of Years 2547 and 2548	166
Figure 5.74 Registration Subject Pivot Table	167
Figure 5.75 Registration Subject Pivot Table of Faculty of Science	168
Figure 5.76 Registration Subject Pivot Table of Computer Science Department	168
Figure 5.77 Registration Subject Pivot Table of SCCS155	169
Figure 5.78 Institute's Income from Registration of Academic Year 2547	170
Figure 5.79 Drill-Down Allocation Income to Lot No Level	171
Figure 5.80 Allocation by Program Pivot Table	172
Figure 5.81 Rotating Allocation by Program Pivot Table	173
Figure 5.82 Allocation by Program Pivot Table with Income Growth	173
Figure 5.83 An Alternative View of Allocation by Program Pivot Table	174
Figure 5.84 Grading Pivot Table of SCCS155	175
Figure 5.85 Grading Pivot Table of SCCS155 with Student (%)	175
Figure 5.86 GPA Pivot Table	176
Figure 5.87 GPA Pivot Table with Percentage of Students	176
Figure 5.88 GPA Pivot Table with GPA Greater Than 2.00	177
Figure 5.89 Alternative View of GPA Pivot Table	178
Figure 5.90 GPA Pivot Table	179

CHAPTER 1

INTRODUCTION

The Mahidol University (MU) Student Information System was established in 1980. The system consists of an admission system, a registration system, a grading system, and a curriculum system. It is developed to support daily processes of the university, such as maintaining student data, handling registration data, updating and controlling payment transactions, allocating registration income, handling grade transactions, and maintaining curriculum and subject data, etc. It provides an efficient operation for academic services. The system accumulates a large amount of data that grow rapidly and are stored in many structures and formats aimed at good performance of the university's business process. The system is designed to store data and facilitate academic operations and is useful for daily operations but it is not designed to provide information for analysis, forecasting and strategic planning that support decision making.

Nowadays, the competitive business needs an appropriate information system for running the daily operation and supporting decision makers to make faster and better decisions. Data warehousing is one of an information technologies that support managerial decision makers. It integrates, reconciles and summarizes a large amount and different format of data from multiple operational systems into a common concept to make it available for decision support. It brings the information more valuable for the business. The primary goal of the technology is to provide information readily accessible and easy to understand, which supports the strategic decision.

This research focuses on student information. We use a case study of Mahidol University to create the individual data warehouse called Mahidol University Student Data Mart (MU-SDM). The process of creating MU-SDM consists of integrating information from operational systems, reconciling and transforming it into a common format, and loading it into a data mart repository. The process of acquiring data for a data mart is called an ETL process (Extract, Transform and Loading). The data are

stored in multidimensional structures which support executives' analytical and navigational activities. In addition, the process also consists of the refreshment of the data mart when operational data are changed. The data mart refreshment is an important process that implies the efficient data collection and aggregation. It needs a mechanism for detecting changed data to incrementally refresh a data mart that can reduce time for processing. The change monitoring is added in ETL for refreshment. The next process is representing the information for the user to support his strategic decision. The executives can access the data mart via an intranet and perform analytical functions such as slice, dice, roll-up, drill-down and pivoting. The information is represented in a tabular and graphic format.

1.1 Problem Statement

Based on a study of the current system of Mahidol University student information and data mart technology, it has been found the following problem:

1. Lack of data warehouse to support university executives needs in academic area.

A large amount of data of Mahidol University student information system has been established but a decision maker cannot fully utilize the detail and many structures of data for decision making. These data are presented in multiple structures, formats and sources that are useful for daily operations, but they can partially be used for analysis, planning and making strategic decisions. The decision maker needs appropriate information technology to make it more useful for planning and making a decision. The heterogeneous data from multiple structures must be integrated and consolidated into a summarized and common format to make it useful for decision making.

2. Data of Mahidol University student information system are frequently changed, then the warehouse data must be synchronized with the underlying sources. Therefore, a data refreshment process must be designed to solve this problem.

After the data warehouse was built, as time advances, Mahidol University student information which is used as source of data warehouse are changed, so the warehouse data must be updated. This process is called data warehouse refreshment,

which is an important and complex process. To update the data warehouse by reloading it with current student information and propagating to the data warehouse with overriding previous warehouse state can cause a loss of historical information and is time-consuming. Periodic propagating by only changing data from operational sources to data warehouse is a better practice for data warehouse refreshment. It keeps historical information and is not time-consuming.

1.2 Research Objectives

This research project has the following objectives.

1. To study the current environment of Mahidol University Student Information System and data warehouse technology in order to find an approach for solving the problem.
2. To find an approach for designing and developing a specific data mart system to provide more information appropriate for decision-making
3. To apply an appropriate data warehouse technology to develop Mahidol University Student Data Mart.
4. To apply an appropriate change monitoring technique for data warehouse refreshment.

1.3 Scope of the Research

This research project focuses on the development of Mahidol University Student Data Mart System, which consists of the following steps of development.

1. Study the current environment of Mahidol University Student Information System and data warehouse issues, and survey existing university data warehouses to be used as a guideline for developing Mahidol University Student Data Mart.
2. Analyze and design a working system by proposing a framework for building and refreshing Mahidol University Student Data Mart.
3. Apply related techniques for building the data mart, design a change monitoring process to detect changed data and design change extraction for data mart refreshment.

4. Analyze and design database for storing Mahidol University Student Data Mart.
5. Develop a prototype of the analytical application to access Mahidol University Student Data Mart.

1.4 Chapter Organization

This research project is organized in 5 chapters as the following:

Chapter 1 : Introduction

This chapter introduces the overview, problem statement, objectives and scope of this research project.

Chapter 2 : Literature Review

The overview of data warehouses, and related research in data warehouses is described and used as a guideline for building MU Student Data Mart (MU-SDM).

Chapter 3 : Approach

This chapter describes related techniques and an application of techniques of MU-SDM.

Chapter 4 : System Analysis and Design

This chapter explains a design of the system process based on the principle of data flow diagram (DFD), conceptual schema, file structure, and input/output design.

Chapter 5 : Implementation Result

This chapter explains the MU-SDM functions and detail of testing scenarios.

Chapter 6 : Discussion and Conclusion

In conclusion, we discuss our work in this chapter besides giving suggestions for an improvement of the proposed system.

CHAPTER 2

LITERATURE REVIEW

In this chapter, the concept of data warehouse and existing data warehouse projects are introduced. In addition, the related techniques are reviewed.

2.1 Overview of Data Warehouse Concept

Data Warehouse is an information technology that integrates diverse, autonomous data into a common format for decision makers to make faster and better strategic decisions. The data warehouse contains integrated, historical, subject-oriented and summarized data that are transformed into a suitable format for analysis and decision making. For example, a sales data warehouse contains information on products sold, time of sales, locations of sale and sales persons.

Bill Inmon, the father of Data Warehousing, defined the data warehouse as follows: “A Data Warehouse is a subject-oriented, integrated, nonvolatile and time-variant collection of data in support of management’s decision.”[24]

A data mart is a small data warehouse. It is used to provide information for specific requirements of a company’s department or a geographical region. Examples are a data mart of sales department, finance department, and so on. A large organization may have regional branches in Europe, America and Asia, each of which has its own data mart.

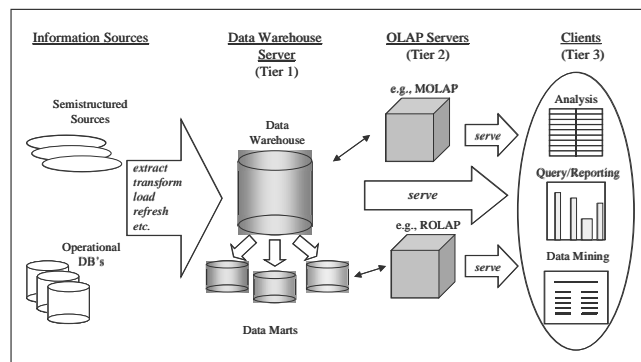


Figure 2.1 : Overview of Data Warehouse Infrastructure [25]

Figure 2.1 shows an overview of data warehouse infrastructure consisting of data sources that are extracted, transformed and loaded into a data warehouse. The data warehouse is periodically refreshed when the source data are changed. Besides a data warehouse, there are several departmental data marts or geographical data marts. The data warehouse and data mart are managed by the data warehouse servers storing data in multidimensional structure and serving it to presentation tools such as query/reporting tools, data mining tools, analysis tools in multidimensional views. Moreover, there is a metadata repository responsible for monitoring and administering the warehousing system[7].

2.1.1 Data Warehouse Architecture

Chamoni and Gluchowski [1], proposed three major data warehouse system architectures, namely, a virtual data warehouse, a single data warehouse and an individual data mart (Figure 2.2).

Virtual Data Warehouse : The concept is that of integrating data from various systems without storing or maintaining data in physical database. The end-user directly accesses operational data via the application.

Single Store Warehouse : A data warehouse stores operational data into a single or central database. This data warehouse may be generated from individual data marts. Data from various operational systems are integrated, consolidated, summarized and transformed into a consistent model to make it complete and available to a user.

Individual Data Mart : A data mart is a small data warehouse. A data mart may be created by a department to provide its own information, for example, a sales data mart, a production data mart, a finance data mart and so on. Another data mart comes from a regional office of companies. For example, a large organization may have regional offices in Europe, America and Asia, each of which has its own data mart.

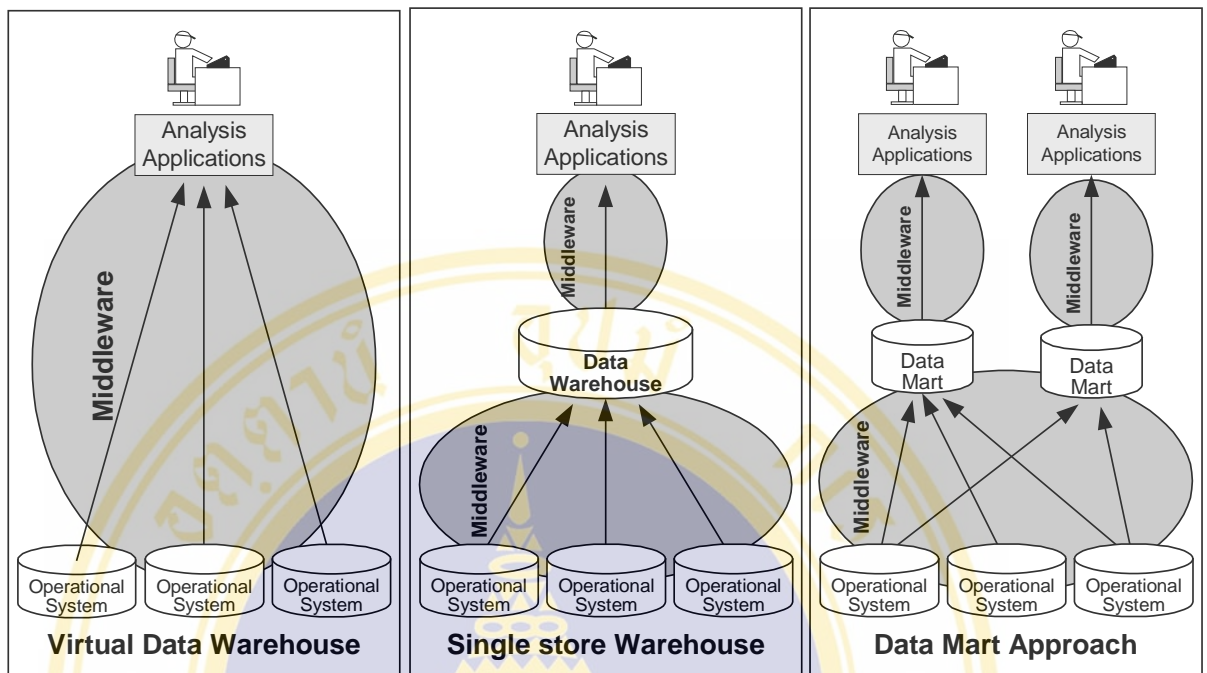


Figure 2.2 : Data Warehouse Architecture[1].

2.1.2 Dimensional model [8, 9]

A data warehouse is designed by using dimensional modeling which is composed of a central fact table and a set of dimension tables.

- Fact Table

A fact table contains the data that measures the business operations. The fact is used to analyze and summarize to gain a better understanding of the business. It contains numerical data (facts) that can be summarized to provide information about business operations. It also contains foreign keys that link with primary keys of related dimension tables which contain descriptive information of fact records.

- Dimension Table

Dimension tables are known as lookup or reference tables, composed of one or more hierarchies of attributes. Dimensional attributes are used to describe dimensional value in a fact table. They are often descriptive, textual data. The examples of dimension are customers, products, and time.

There are two basic conceptual schemas of Data Warehouse :

❑ Star Schema

A single fact table is the middle connected to a set of dimension tables with its foreign key and primary key of dimension tables. The relationship between a fact table and dimension tables is one-to-many relationship. The dimension tables are not normalized. The star schema provides an easy access for a user and supports simple and efficient queries because of fewer joins of a fact table with dimension tables for information retrieval. Figure 2.3 shows a simple star schema. It consists of the SaleFact fact table shown in the central diagram and three surrounding dimension tables of Product, Time and Store.

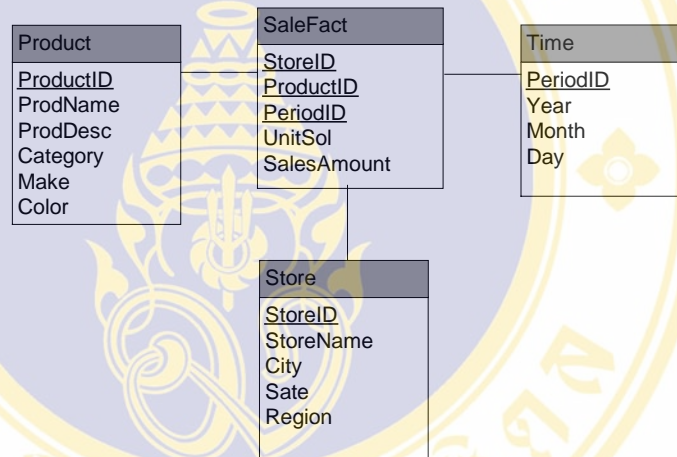


Figure 2.3 : Star Schema

❑ Snowflake Schema

The dimension tables are in a normalized form. The dimensional hierarchy is represented explicitly by one or more dimension tables divided into multiple tables with the sub-dimension tables joined with a to primary dimension table instead of the fact table. A snowflake schema increases the complexity and cost of queries because of more number of joins. This schema decreases storage space and redundancy but is easy to update and maintain the dimension tables. Figure 2.4 shows a snowflake schema. It consists of the SaleFact fact table shown in the central diagram and three surrounding groups of dimension tables of Product, Time and Store. The Product is split into three tables: Product, Model and Maker.

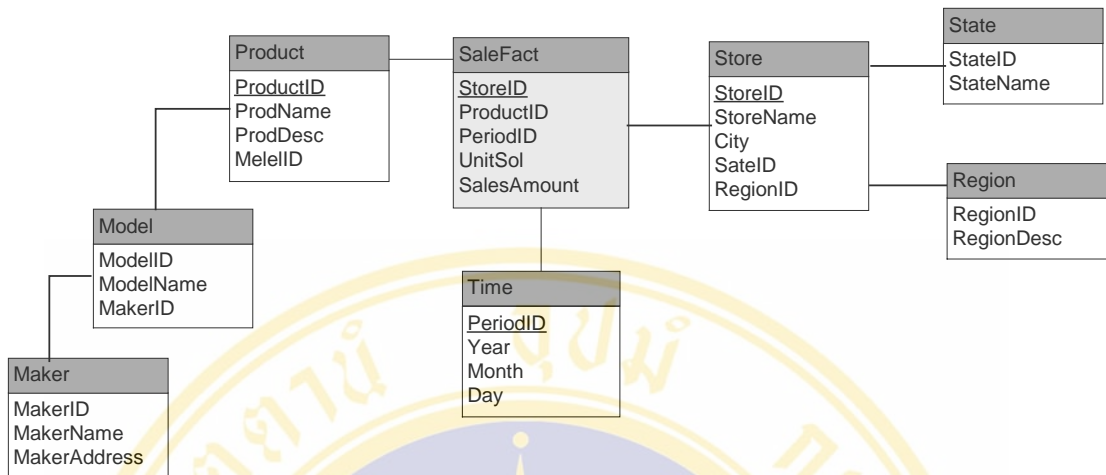


Figure 2.4 : Snowflake Schema

2.1.3 OLAP [4, 19]

OLAP stands for Online Analytical Processing, a technology that uses multidimensional data called cubes to provide a fast, consistent, interactive access to data warehouse data. A data cube is defined over a multidimensional space which consists of several dimensions, e.g., product, region, time, etc as shown in Figure 2.5. Data cube cell represents the numeric measure corresponding to the objects to be analyzed, e.g., sales, orders, inventory, etc.

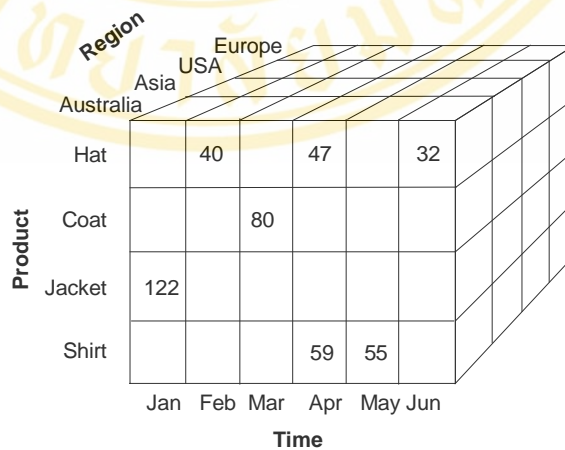


Figure 2.5 : Cube of Sales

OLAP Model

There are two basic OLAP models for building a data warehouse.

□ ROLAP

ROLAP stands for Relational Online Analytical Processing. ROLAP uses a standard relational DBMS and star schema to store warehouse data. This model presents data to the users in the form of business dimensions. This model can handle a large amount of data but the performance is slow. The architecture of a ROLAP model is three-tier architecture as shown in Figure 2.6. The users submit a complex query based on multidimensional view, and the ROLAP engine transforms the query into SQL to the relational database. Then, the engine converts the result from database into a multidimensional format before returning it to the users.

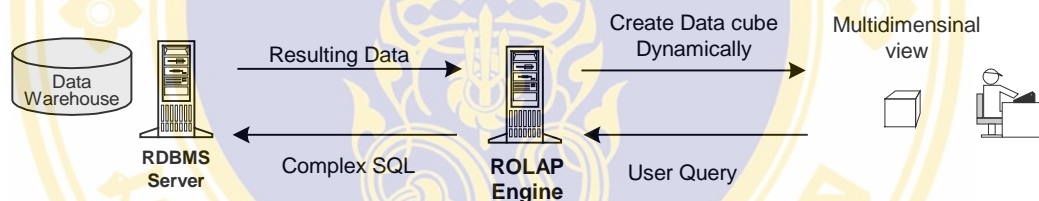


Figure 2.6 : Three-Tier Architecture of ROLAP Model

□ MOLAP

MOLAP stands for Multidimensional Online Analytical Processing. MOLAP stores data in a specialized multidimensional database. MOLAP provides good response time on OLAP queries since it processes data that is already stored in a multidimensional array in which all possible combinations of data are reflected. The three-tier architecture of a MOLAP model is illustrated in Figure 2.7.

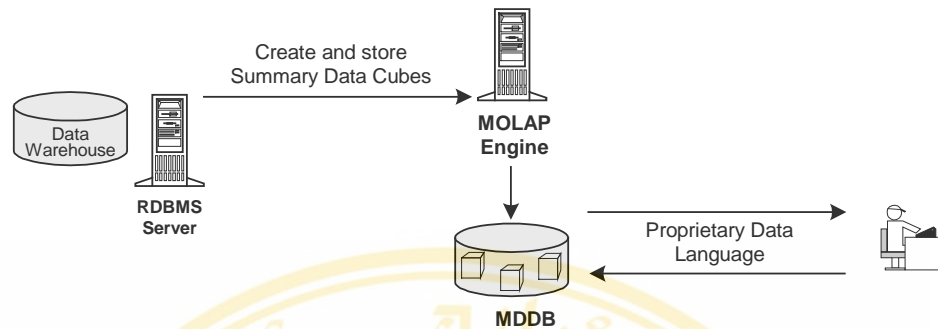


Figure 2.7 : Three-Tier Architecture of MOLAP Model

OLAP Operation

There are five basic operations of OLAP such as drill-down and roll-up, slice and dice, and pivoting. These operations are briefly described below.

- Drill-down and Roll-up

Figure 2.8 shows the star schema of sales with attributes of time dimension : date, month, quarter, and year. These attributes show a hierarchical sequence from date to year. A year includes quarters, a quarter includes months, and each month consists of dates. These attributes are called hierarchies of the time dimension.

Drill-down and Roll-up are basic hierarchical operations of OLAP when displaying data at multiple levels of aggregations. Drill-down refers to the operation of displaying data at a lower level of detail. And roll-up refers to the operation of displaying data at higher hierarchical levels of aggregation. Drill-down and roll-up are illustrated in Figure 2.9.

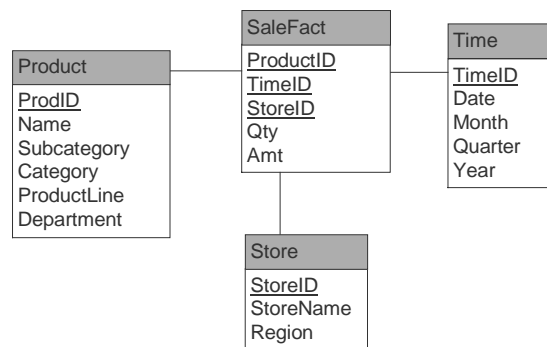


Figure 2.8 : The Star Schema of Sales

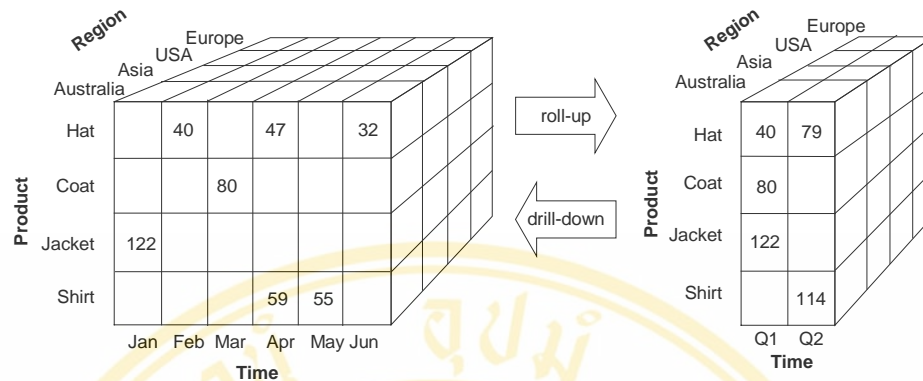


Figure 2.9 : Drill-down and Roll-up

❑ Slice and Dice

Slice and dice are OLAP operations that relate to the selection of cubes. A slice operation performs a selection on one dimension of a cube, and results in a subcube. For example, if we select Australia (Region dimension), we will reduce the dimensions of the cube, of which results is displayed in Figure 2.10.

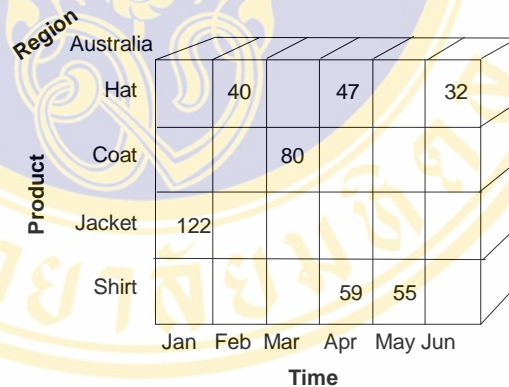


Figure 2.10 : Sales of Australia

A dice operation is used to partition or group one or more dimensions. This operation involves dividing a cube into small subcubes. For example, if we select Time dimension (Jan to Mar) and Region dimension (Australia, Asia) from the original cube, the subcube will be revealed in Figure 2.11.

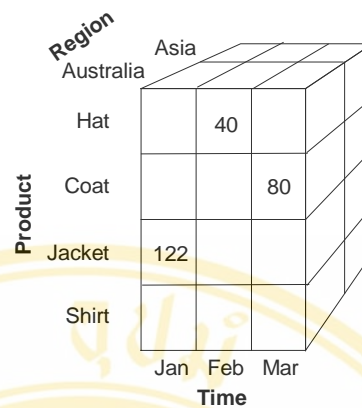


Figure 2.11 : Sales of Australia and Asia Region

□ Pivoting

Pivoting is an OLAP operation which rotates dimensions of report in order to provide an alternative presentation of data. For example, a report that has Time rows and Products columns is rotated into a report that has Product rows and Time columns as shown in Figure 2.12.

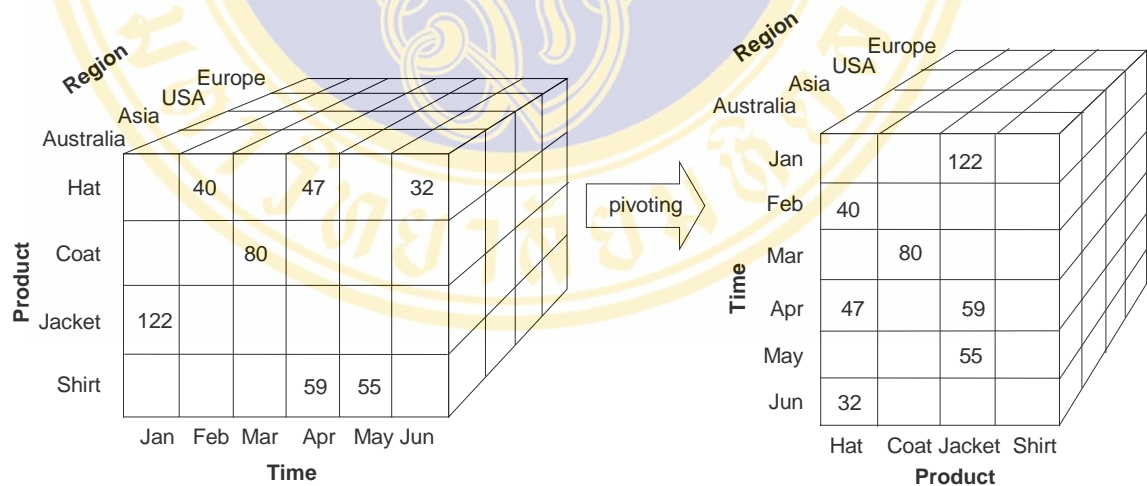


Figure 2.12 : Pivoting of Sales.

2.1.4 Data Warehouse Processes

Figure 2.13 shows a data warehouse architecture. It includes two major categories of data warehouse processes. The first one is a creating process which performs extracting operational data from diverse sources, transforming it into a consistent format of warehouse, and loading the transformed data into a data

warehouse repository. After the initial loading, the operations sources are changed and then the data warehouse must be refreshed periodically in order to keep the warehouse up-to-date. The process of creating a data warehouse is very time-consuming and complex. The other one is an analysis process, which involves reporting and visualizing in order to deliver information to a user. The four main processes of data warehousing are extraction, transformation, loading and information delivery, which are briefly described below.

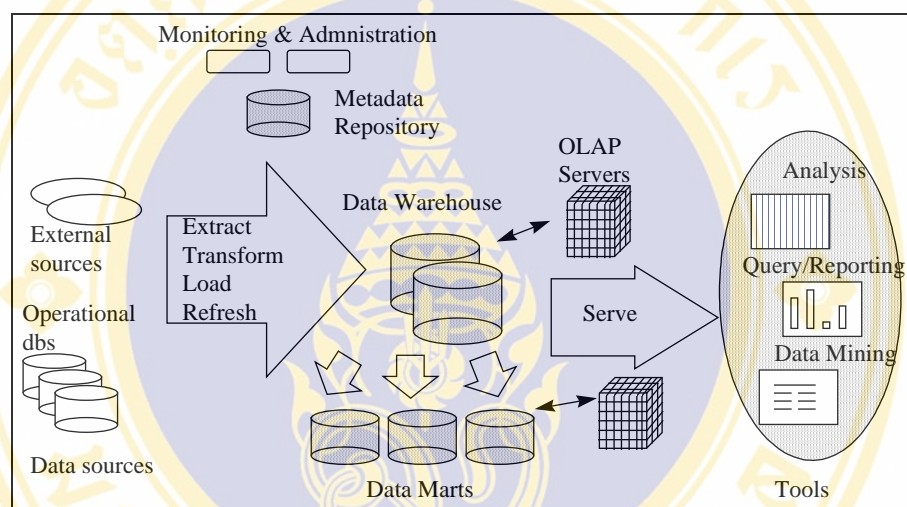


Figure 2.13 : Data Warehouse Architecture [7]

□ Extraction

This process deals with retrieving selected data from various sources and writing it into a data staging area. Each source may be different format or platform and so a warehouse developer must use an appropriate technique for each data source. The technique is used to extract data such as a database vendor provides the standard application programming interfaces (APIs) like OLE DB, ODBC, JDBC, etc. Some third party vendors provide middleware and proprietary gateways for extracting data from various database systems. In addition, for legacy systems, the extraction programs are written.

□ Transformation

Due to the difference among heterogeneous source data, the extracted data must be performed through several types of transformation in order to increase data quality and make it suitable for warehousing before moving it into the data warehouse. This process is responsible for data cleansing, data converting, data validation, data accuracy, as well as aggregating and summarizing data before moving it into target storage.

Cleansing data includes error detection, conflict resolution of data in the same source or different source, duplicate elimination when the data is integrated from multi-sources, and handling missing values.

Converting data includes:

- Combining multiple fields into one field
- Breaking down one field into separate fields.
- Mapping the same object from different sources which have a different presentation into a standard format.
- Converting the character set into a standard character set, such as converting the mainframe EBCDIC format into the ASCII format.
- Summarizing source data instead of loading all detail of data into the data warehouse.

□ Loading

It is the process of moving data into the data warehouse repository.

Paulraj Ponniah defined three types of loading described below [4]:

- Initial Loading – populating all the data warehouse tables for the very first time
- Incremental Loading – applying ongoing changes as necessary in a periodic manner
- Full Refreshment – completely erasing the contents of one or more tables and reloading with fresh data. Periodically rewriting the entire data warehouse or doing partial refreshment

to rewrite only specific tables. This type of loading is similar to the initial loading.

□ Information Delivery

It is the process of making an easy access to the information for users. This process provides an online query, interactive analysis sessions, and simplification of access by hiding internal complexities of data storage from users .

2.2 A Survey of Data Warehouse Projects

2.2.1 Existing University Data Warehouse Projects

We have studied some existing university data warehouse projects to be used as a guideline for developing our data warehouse. They are summarized as follows:

2.2.1.1 Stanford University (WHIPS Project) [11, 12, 13]

WHIPS stands for Warehousing Information Project at Stanford. It focuses on the integration components. Its architecture was designed to identify changed data at heterogeneous sources, transform and summarize it in accordance with warehouse specifications, and incrementally integrate them into the warehouse[12]. The WHIPS system architecture is shown in Figure 2.14. The system is composed of many modules that communicate with one another [13]. The modules are briefly described below.

- **Source and Monitors** : The monitor detects modifications on the source data and notify the integrator of them. The trigger-based monitors are implemented for cooperative sources, and snapshot monitors for flat file sources
- **Wrappers** : Wrappers are responsible for translating single source queries from the internal relational representation used in a view tree to queries in the native language of its sources.
- **Integrator** : The integrator role is to facilitate view maintenance by finding out which modification need to be

propagated to which view. The view tree automatically generates the rules that specify which view managers are interested in which modification when each view is defined. These rules are used by an integrator.

- **View Specifier** : Views are defined in subset of SQL, and the view specifier parses it into a view tree, adds relevant information from meta-data store, and sends the view tree to the integrator
- **Meta-data Store** : The meta-data store keeps catalog information about the sources and how to contact them, the relations stored at each source, the schema of each relation and also keeps track of all view definitions.
- **View Manager** : Each view manager is responsible for maintaining each view, using Strobe algorithms to maintain view consistency.
- **Query Processor** : The query processor is responsible for distributed query processing and using standard techniques to prune the queries. The query processor receives global queries from the view managers and poses the appropriate single-source queries to the source wrappers to answer them. It then passes the composite global query answers back to the view managers.
- **Warehouse and Warehouse Wrapper** : The warehouse wrapper receives view definitions and modifications in order to the view data in a canonical format, and translates them into the specific syntax of warehouse database.

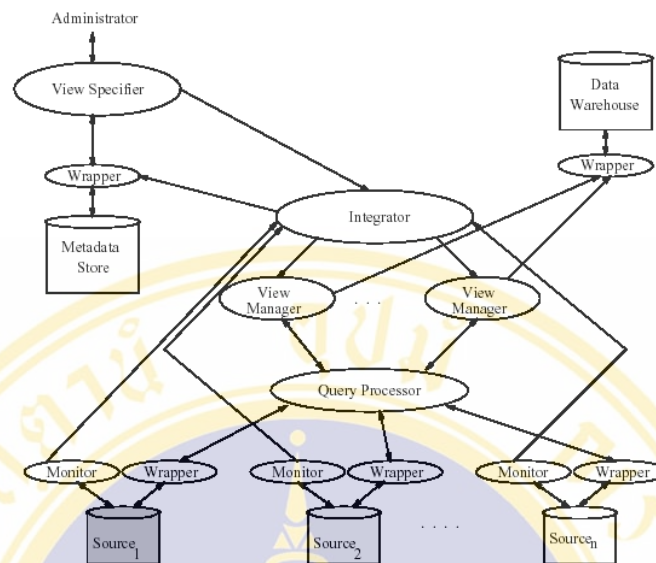


Figure 2.14 : The WHIPS System Architecture for Warehouse Maintenance [11]

The flow of message in the WHIPS system has three distinct operations. First, at startup, the modules identify themselves to one another. And whenever a new source becomes available, this action will occur. Second, the view is initialized and the system is prepared to maintain the view, whenever a view is defined. Third, each defined view is maintained in response to modifications that affect the view.

2.2.1.2 The University of Washington Data Warehousing Program [15]

The UW Data Warehousing Program (DWP) provides a central resource of integrated, historical and current data that the university community can access to meet informational needs in support of the UW academic, research and administrative missions. The DWP is developed based on a three-tier warehousing architecture

- **Tier 1: Persistent Staging Area** -- As data are extracted from source systems and databases, they are collected in the persistent staging area, where data organization & structures closely resemble their operational source systems.

- **Tier 2: Data Warehouses** -- From the persistent staging area, data warehouses are developed where the data are organized for distribution.
- **Tier 3: Data Marts** -- From the warehouses, data marts are produced where data are organized for access and grouped to address defined sets of business issues and questions.

In addition, the DWP is developing Operational Data Stores (ODS) to supplement the university's operational systems. These data stores are designed, developed, and operated to support applications.

The architecture of DWP is shown in Figure 2.15. The common data model (CDM) is the pathway by which data is moved from the persistent staging area to the data warehouse, the operational data store, and the exploration warehouse. The CDM defines data consistency across the common information architecture. As a model-based approach, it provides standards for subject area integration, and entity and attribute naming.

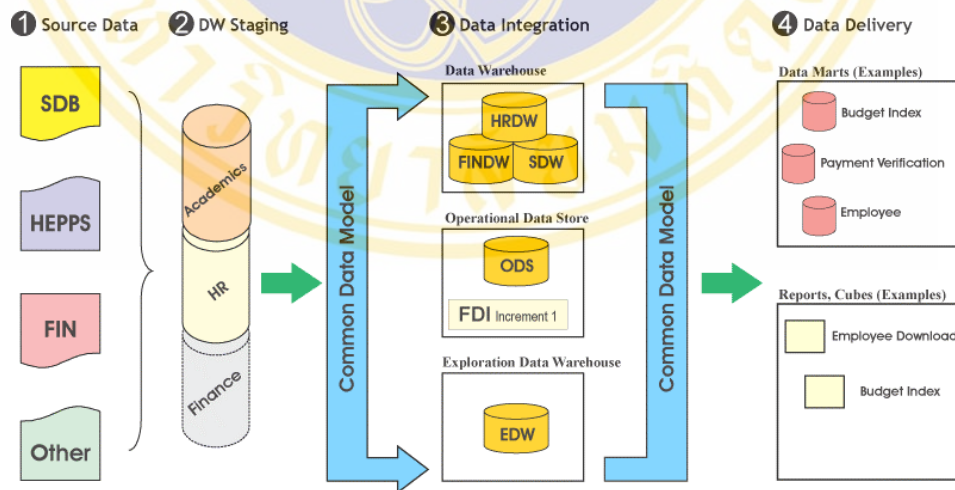


Figure 2.15 : The Architecture of UW Data Warehousing Program [15]

2.2.1.3 Georgetown University [14]

Georgetown University data warehouse architecture is shown in Figure 2.16. The architecture consists of three major environments listed below.

- **The Source Data Environment** contains various applications that provide financial management, student information and human resources data to the GU Data Warehouse.
- **The Data Warehouse Environment** includes the relational data warehouse and the multidimensional data marts and other tools for building and maintaining. Data extracted from diverse sources must be transformed in order to provide useful information. Transformations include data temporal and data type synchronization, data cleaning, data computation, and data aggregation. Transformed data must be loaded into the Relational Data Warehouse. The loading process can be both time-consuming and computationally intensive. Modeling the Data Warehouse includes the logical and physical structure of both the Relational Data Warehouse and Multi-Dimensional Data Mart(s) as well as the system configuration required to support functionality and performance requirements. Security processes protect warehouse data from unauthorized access. Note that since the Data Warehouse and Data Mart are read-only data bases, write "protection" is mainly confined to ensuring the fidelity of the load processes.
- **The User Access Environment** provides "front end" access to the Data Warehouse for the full range of end users. The report that is provided to users includes ad hoc query and reporting, scheduled query and reporting and multidimensional analysis.

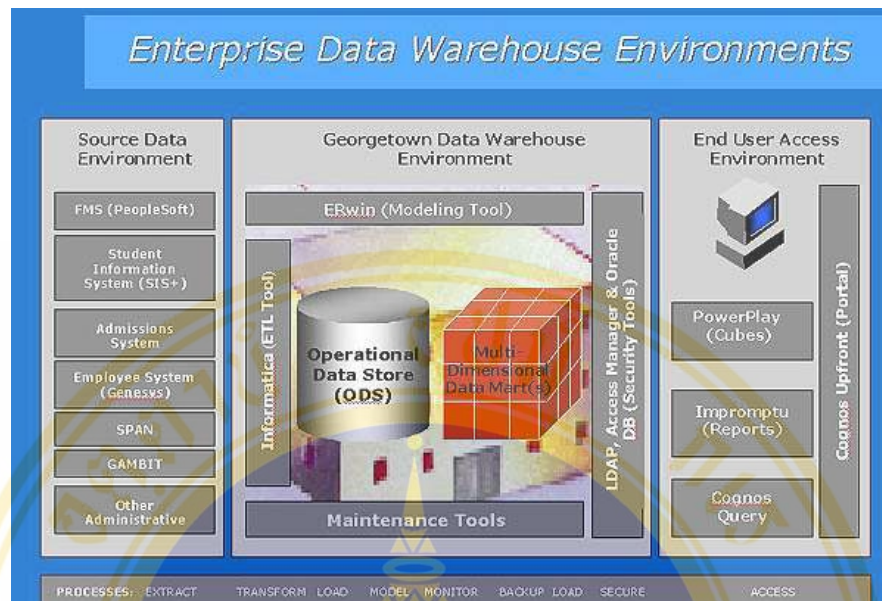


Figure 2.16 : The architecture of Georgetown University Enterprise Data Warehouse[14]

2.2.1.4 University of Pennsylvania Data Warehouse [16]

The University of Pennsylvania Data Warehouse makes data available to decision makers for query, analysis, and reporting. The Warehouse stores selected data from existing systems. The data is filtered, made consistent, and aggregated in various ways. The user access the warehouse without programming skills, and use desktop tools such as Business Objects support analysis and reporting. The Warehouse contains students and courses data, sponsored projects data, salary management data, general ledger data, space data and position inventory data.

2.2.1.5 University of Maryland Data Warehouse [17]

The University of Maryland Data Warehouse is a collection of integrated data from a transaction system, consisting of the Student Information System (SIS), the Academic Resources System (ARS), and the Financial Resources System (FRS). The data are identified, cleaned and restructured. In addition, derived data are added. This process is performed by an extraction program. The warehouse can be accessed by an authorized UM user for performing analysis and producing ad hoc queries and reports as desired without having to modify the transactional system.

2.2.1.6 Arizona State University Data Warehouse [18]

ASU data warehouse extracts data from the mainframe and loads it into UNIX server running an RDBMS. ASU warehouse server is a Sun 630 with 512 megabytes of memory and two processors, running the Sun Solaris 2.3 operating system. The RDBMS is sybase SQLServer release 10.x. The data warehouse contains four primary databases: student, finance, human resource, and course. These databases are updated weekly, biweekly, monthly, and yearly, depending upon the data. Users can view data with a query tool such as BrioQuery or MS Access and can export it to other tools. The environment of ASU data warehouse is shown in Figure 2.17.

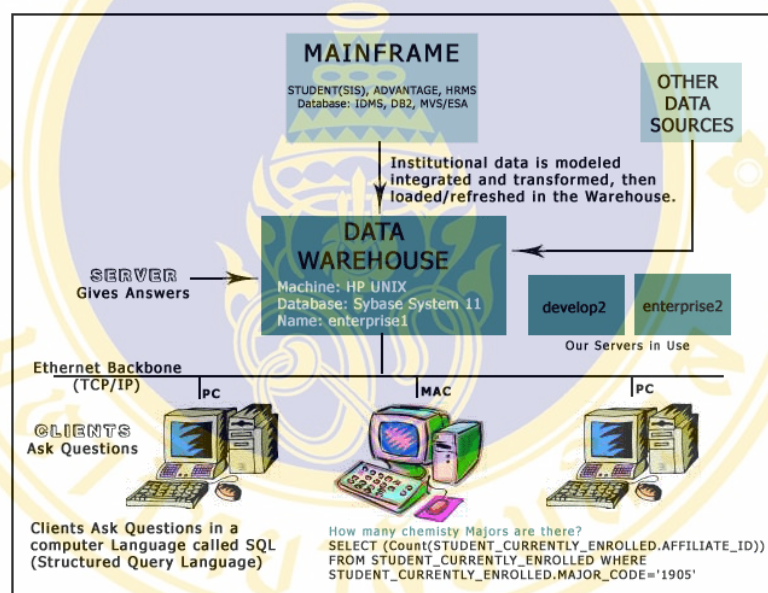


Figure 2.17 : ASU Data Warehouse Environment.[18]

2.2.1.7 A comparative Survey of Six Existing University Data Warehouse Projects

We have found that six universities' data warehouses have different environments as follows:

DBMS : Most universities use Oracle DBMS and the rest use Sybase and SQL Server 2000.

Tools : There are several commercial ETL (Extract, Transform and Loading), Data Modeling, Metadata and Query tools used to implement a data warehouse.

Schema : Most universities use Star schema to implement a data warehouse.

Server OS : The Operating System of the server is various, like Unix and Microsoft Windows but most universities prefer Unix.

Refresh Schedule : The schedule is determined by type of data.

Subject Area : The various subject areas are implemented such as HR, student, finance, payroll, etc. It varies for each university.

Table 2.1 shows a comparison of six universities' data warehouses.

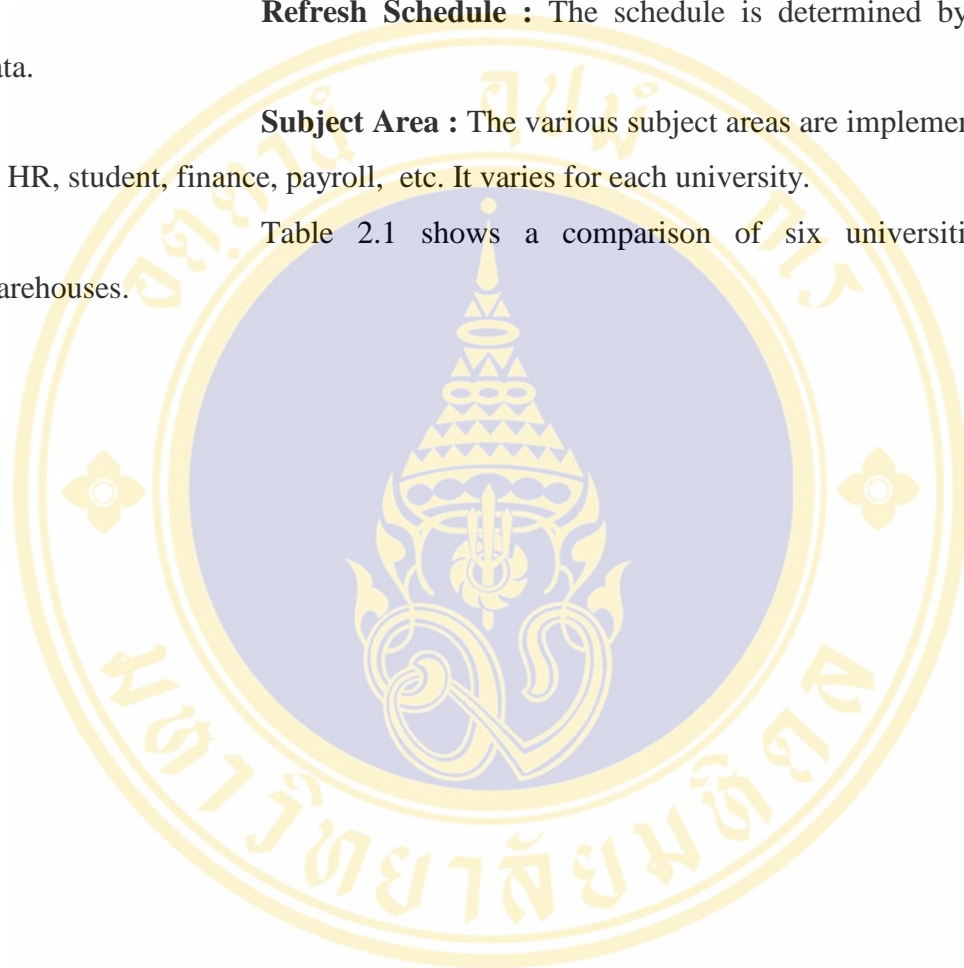


Table 2.1 : A Comparison of Components, Tools, Platform and Subject Area of University Data Warehouse Projects.

University	DBMS	Tools				Schema Type	Server OS	Refresh Schedule	Subject Area
		ETL	Data Modeling	MetaData	Query				
Stanford U (WHIPS Project)	Sybase	PowerCenter by Informatica	Visio	N/A	Brio and Business Objects	Star Schema	Sun Solaris	N/A	Accounting
U. of Washington	SQL Server 2000	SAS Enterprise ETL Server v9, MS DTS	Visible Analyst	Visible Analyst	Cognos Series 7 (Impromptu, PowerPlay, Upfront portal)	N/A	MS Windows 2000 Enterprise; Linux	N/A	HR, Academics, Finance, Payroll
Georgetown U.	Oracle.	Informatica.	Erwin.	Home grown Oracle tables	Cognos Suite, Series 7.	Star Schemas	Unix.	N/A	Student, Admissions, Financial, HR / Payroll, Facilities, Alumni Development, Technology. Future – Academic
U of Pennsylvania	Oracle 9.2.0.4	N/A	N/A	N/A	Business Objects	N/A	N/A	Varies for each data collection	Student Records, Financials

Table 2.1 : A Comparison of Components, Tools, Platform and Subject Area of University Data Warehouse Projects. (cont.)

University	DBMS	ETL			Tools			Schema Type	Server OS	Refresh Schedule	Subject Area
		Multiple Extract Methods	Data Modeling	MetaData	Query	Data Modeling	MetaData				
U. of Maryland	Oracle 8.1.7.4	Multiple Extract Methods	ERwin.	In-house application (Oracle, Java)	BrioQuery	RS/6000 SP running AIX 4.3.3	Denormalized Relational.	Determined by static/non-static nature of data, need, etc.	Budget, Financial Accounting, Personnel (HR), Payroll, Contracts & Grants, Financial Aid, Student Records and Admissions.		
Arizona State U.	Sybase.	Cobol or Sybase stored procedures	PowerDesigner	Homegrown definition tables accessed through BrioQuery remarks.	BrioQuery, MS-Access, Web development tools	Unix	Star Schema.	Determined by type of data	Financial, HR, Student Course.		

2.2.2 Data Warehouse Framework.

This section describes three data warehouse frameworks used as a guideline to design an MU Student Data Mart (MU-SDM) process. They are listed below.

- Paulraj Ponniah's Data Warehouse Framework [4]
- Rob Mattison's Data Warehouse Framework [2]
- Harry S. Singh's Data Warehouse Framework [3]

2.2.2.1 Paulraj's Framework

Paulraj [4] identified six major components of a data warehouse as six building blocks and grouped these architectural components into three major areas as shown in Figure 2.18. Each of the components has definite functions and provides specific services.

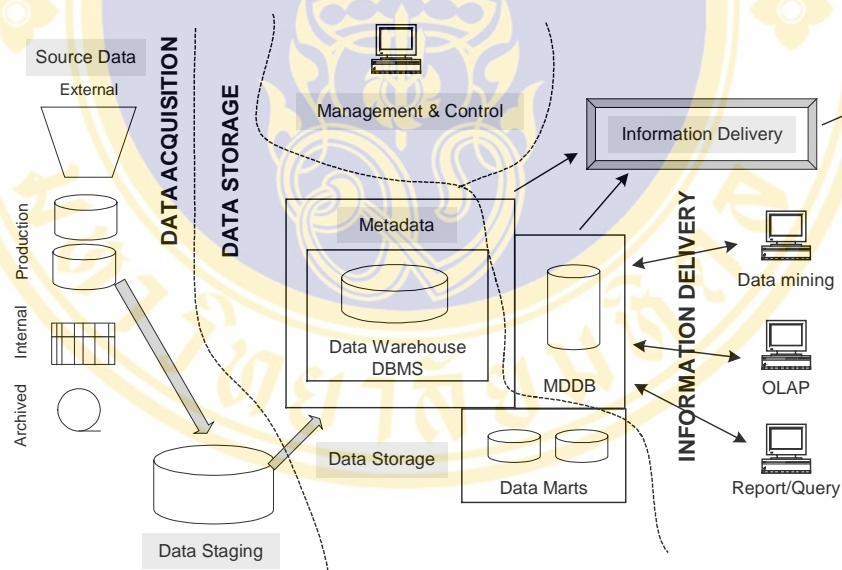


Figure 2.18 : Paulraj's Data Warehouse Components.[4]

The six major components of a data warehouse are:

- **Source Data** : The source data may be grouped in four categories such as data from an operational system, internal data, archived data and external data.
- **Data Staging** : Extracted data from separate sources are moved to an area with functions to clean, change, combine,

convert, deduplicate and prepare various source data into a suitable form for use in a data warehouse.

- **Data Storage:** A separate repository contains a large volume of historical data for analysis. The data is kept in a suitable structure for analysis and separates from storage of an operational system.
- **Information Delivery :** In various levels of a data warehouse users need different kinds of reports from it. In order to meet this requirement, the information delivery component includes different methods of delivery such as ad hoc reports, complex queries, MD analysis, statistical analysis, EIS and data mining.
- **Metadata :** It is data in the data warehouse which keeps the information about logical data structures, files and addresses, indexes and so on. This component serves as a directory of the contents of the data warehouse. Metadata is classified into three categories such as operational metadata, extraction and transformation metadata, and end-user metadata.
- **Management and Control Component :** This component interacts with the meta data component to perform a management and control function. It sits on top of all other components and coordinates the services and activities within a data warehouse.

The major areas of a data warehouse are

- **Data Acquisition :** This area covers a process of extracting source data, moving it to a staging area, and preparing data for being loaded into a data warehouse. This area includes two architectural components, source data and data staging. This area performs data extraction, transformation, and

loading, which is time-consuming, labor-intensive, and very important.

- **Data Storage** : This area covers a process of loading data from the staging area into the data warehouse. All functions of transforming and integrating the data are completed here.
- **Information Delivery** : This area covers a process of making information available, as well as easy to access, use and understand for a user. The information access is through online queries and interactive analysis sessions. Almost all modern data warehouses provide online analytical processing (OLAP).

2.2.2.2 Mattison's Framework

Rob Mattison [2] described the data warehouse framework. His framework comes from the idea that a data framework project is really made up of three major components (see Figure 2.19).

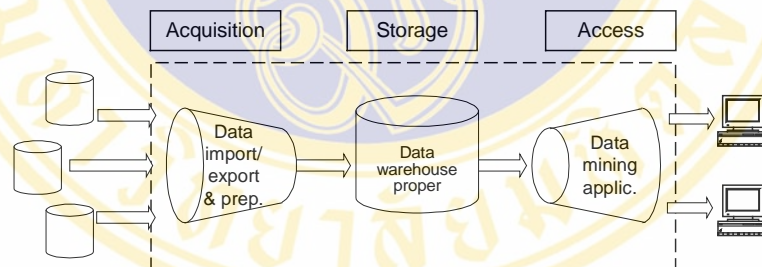


Figure 2.19 : Mattison's Data Warehouse Components.[2]

- **Acquisition Component** : is responsible for pulling the data from a source system, preparing it, loading it into the warehouse, and exporting it out again when the source is changed. This portion of the system includes all of the programs, applications, data-staging areas, and legacy systems interfaces for importing and exporting data.
- **Storage Component** : is the data warehouse itself, a large database that holds a huge amount of information from

various sources. The data is organized in an easily accessible format.

- **Access Component** : responsible for making a warehouse useful for decision making. This component includes all of the different data-mining applications.

2.2.2.3 Singh’s Framework

In 1997, Harry S. Singh [3] described functions of a data warehouse and the flow of data from an original source to the user as shown in Figure 2.20. The process is started with the data being accessed from a heterogeneous operating system. That data is transformed into a suitable format: including cleansing, reconciling, enhancement, summarizing and aggregating. And the data is delivered to data warehouse store based on a selected model. Finally, the valuable information is displayed to a user in the form of an analytical report, a multidimensional spreadsheet or others.

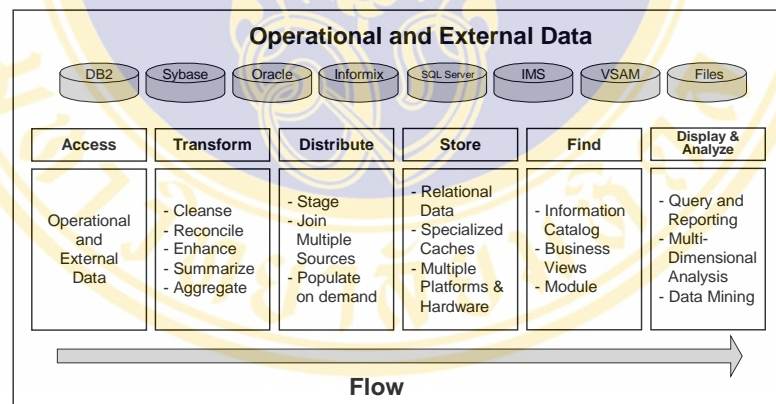


Figure 2.20 : Singh’s Data Warehouse Functions.[3]

2.2.2.4 Comparative Survey of Three Existing Data Warehouse Frameworks

Each framework has different data warehouse processes that are shown in Figure 2.21. Therefore, we have found that a major process of the three frameworks includes the process of preparing data which retrieves various data from operational sources and transforms it into suitable format of data warehouse. After that the data is stored in the data warehouse, which has specialized modeling concepts

and storage structures. The information is delivered to a user in an easily understandable and summarized form to be used for analysis and decision making.

Paulraj's Framework	Mattison's Framework	Singh's Framework
Data acquisition	Acquisition component	Access
- Extracting - Moving - Transformation - Cleansing - Integration	- Extraction - Cleansing - Formatting - Merging & Purging - Key processing	- Operational and External Data
Data storage	Storage Component	Transform
- Full refresh - Incremental load - Backup/Recovery - Security - Data Archival	- Metadata management - Data synchronization - Data transformation mapping	- Cleanse - Reconcile - Enhance - Summarize - Aggregate
Information Delivery	Access component	Distribute
- Query government & optimization - Content browse - Security control - Self-Service report generation	- Report generating - Visualizagion - Simulation	- Stage - Join Multiple Sources - Populate on demand
		Store
		- Relational Data - Specialized Caches - Multiple Platforms & Hardware
		Find
		- Information Catalog - Business Views - Models
		Display & Analyze
		- Query and Reporting - Multi-Dimensional Analysis - Data Mining

Figure 2.21 : Comparison of Three Existing Data Warehouse Frameworks

2.3 Related Techniques

Techniques used to develop MU Student Data Mart proposed in this research project consist of three techniques, namely: change monitoring, triggers, and web based.

2.3.1 Change Monitoring [6]

Data acquisition is a very time-consuming process of data warehouse. After initial loading, as time advances, the operational data is changed, then the data warehouse must be synchronized with the underlying sources. Changed data must be propagated into the data warehouse. The technique of change detection or monitoring must be applied in order to reduce time of a full refreshment process.

Change monitoring is an important process of detecting changed data in the underlying data source and propagating them into the data warehouse. The monitoring technique can be classified as follows:

- *Log-based monitoring* : It is the source system that maintains a log. The monitor inspects the log file and extracts the relevant modifications occurring since the last execution of the refreshment process. However, the log is not easily accessible and is hard to interpret. The example of this technique is WHIPS: using IBM's DataCapture, which reads the log for DB2 and generates a source change table.
- *Trigger-based monitoring* : It is for sources that support active mechanisms like triggers. When a source data is modified (the defined event occurs and condition is met), the trigger is fired and the change monitor is notified automatically to write relevant information into auxiliary tables. (the defined action is executed). The auxiliary tables are cleared after the data warehouse is refreshed. The example of a data warehouse project that implements triggers for change monitoring is WHIPS.
- *Replication-based monitoring* : Replication services of commercial DBMS like IBM's DataPropagator and Sysbase's Replication Server provide mechanisms for propagating updates on base table to replicas. A monitor can use this information to extract and deliver change to a data warehouse.
- *Application-assisted monitoring* : For non-DBMS, modification of an existing application or implementing new ones to detect change is the only choice. One choice is modifying an application program to timestamp changed data. The monitor periodically polls the source and selects data with timestamp greater than the one of the previous refreshment. Another choice is to adapt changing applications to maintain a log file after processing the operational updates.

- *Snapshot comparison* : A source performs snapshot monitoring by periodically reading the current source state (snapshot) and comparing it with the previous source state to compute differences of the two snapshots.

Trigger-based monitoring is used to detect change in a source system in this project. The commercial DBMS like MS SQL Server, which is used in this project, provides active mechanisms in the form of trigger. The detail of trigger is described in the next section.

2.3.2 Trigger [5]

A trigger is an event-driven special stored procedure (program) of the form :

On event If condition Do action

The trigger is stored in and managed by the DBMS. They run concurrently with the database and make the database active. The DBMS automatically fires the trigger in response to certain events on a particular table in a database. When the specified event occurs and the condition is checked, if the condition is satisfactory, the specified action is executed. The trigger is also known as event-condition-action rules (ECA rules). The operation of the trigger generates three pieces of work: **event detection**, **condition evaluation** and **action execution**, each of which has the following details.

□ **Event**

The database events are caused by database operations like retrieval or update of relations (Insert, Delete, Update), transactional events (e.g., commit, abort), or procedure calls.

□ **Condition**

A condition of a trigger is a predicate that should hold for an activated trigger to execute its action.

□ **Action**

An action is a set of database operations that are performed when the trigger event occurs. The actions can consist of INSERT, DELETE, and UPDATE operations.

2.3.3 Web-based Architecture [20]

Web architecture is a flexible and scalable implementation of multi-tier computing and uses a web browser (client) to retrieve information from a web server. The web server can interface with application servers which in turn interface with databases, enabling web applications to determine programmatically the information that is returned to the user. Web is not a network but an application that operates over networks using a communications protocol called HTTP (Hyper Text Transfer Protocol)

The HTTP is the standard for transferring used by World Wide Web. HTTP defines how messages are formatted and transmitted, and what actions web servers and web browsers should perform in response to various commands. To retrieve a web resource, the web browser specifies the detail of the required document in an HTTP Request message. The web server receives this message and, after processing it, completes the operation by returning either the document or an error in the HTTP Response message.

2.3.3.1 Web Programming

□ **PHP (PHP Hypertext Preprocessor)**

PHP is an open source server-side scripting language that is suited for web development and can be used to generate dynamic web content. PHP scripts are embedded within web pages along with HTML, similar to other web scripting languages such as ASP and JSP. Like ASP and JSP. PHP runs on the web server when a page is requested via HTTP, rather than on the Web browser or other client. PHP allows inserting instructions into web pages that the web server software will execute before sending those pages to a browser that requests them. PHP is a more flexible and stable development platform

- **ASP (Active Server Pages)**

ASP is a server-side scripting language that allows developers to add interactivity to their web sites. Active Server Pages allow developers to create dynamic, database-driven web pages on Microsoft IIS Web servers. ASP is developed by Microsoft to take text scripts in an HTML context and run them on the Web server to create dynamic and interactive pages. ASP is limited to web servers running IIS and Microsoft Personal Web Server. But Chili!Soft developed an ASP server for Unix and Linux.

- **JSP (Java Server Pages)[22]**

JSP is a server-side technology. Java Server pages are an extension to the Java servlet technology that was developed by Sun. JSP is a technology to dynamically generate web pages and simplify the process of developing web-based applications. JSP uses a combination of HTML, XML, and short Java scripts that, when compiled, make up servlets. JSP pages are similar in form and function to pages containing PHP or ASP. JSPs are not restricted to any specific platform or server.

- **CGI (Common Gateway Interface)**

CGI script is a small program, or script, that runs on the web server. It usually produces some sorts of dynamically-generated web page or some information pulled from a database. CGI is a mechanism allowing web servers to run external programs or scripts when requested by a web browser. CGI scripts may be written in any programming language. However, CGI scripts are usually written in Perl.

- **Cold Fusion [23]**

Cold Fusion is a Web application development tool that enables the rapid creation of interactive, dynamic, and information-rich Web sites. Cold Fusion does not require coding in traditional programming languages. Cold Fusion is used to create applications by extending standard HTML files with high level formatting functions, conditional operators, and database commands. These commands are instructions to the Cold Fusion processor and form the building blocks on which to build industrial strength applications.

2.3.3.2 Data Access Technology[21]

All database applications need technology for connecting between database and the program used to develop the system. The technology involves remote data sources with different data formats and storage mechanisms. The popular data access technology can be listed as follows:

- ❑ **OLE DB (Object Linking and Embedding for Database)**

OLE DB is a technology that will integrate data from heterogeneous sources such as spreadsheets, project plans, SQL, and legacy databases. OLE DB interfaces provide applications with uniform access to data stored in diverse information sources, or data stores.

- ❑ **ADO (ActiveX Data Objects)**

ADO is designed to be easy-to-use application-level interface to any OLE DB data provider. Its benefits are ease of use, high speed, low memory overhead, and language-independent. It also uses minimal network traffic and has few layers between the client application and the data source, all to provide lightweight, high-performance data access. ADO supports key features for building client/server and Web-based applications.

- ❑ **ODBC (Open DataBase Connectivity)**

ODBC provides a universal database connectivity API that enables applications to access data in a wide range of proprietary databases. Based on the X/Open SQL Access Group's Call Level Interface (CLI) specification, ODBC is an open, vendor-neutral way to uniformly access data stored in different formats and database engines.

- ❑ **RDO (Remote Data Object)**

RDO is specifically designed to access remote ODBC relational data sources, and makes it easier to use ODBC without complex application code. RDO is a primary means of accessing SQL Server, Oracle, or any relational database that is exposed with an ODBC driver.

- ❑ **DAO (Data Access Object)**

DAO provides data access to native Microsoft Jet engine databases (.mdb files), selected ISAM databases, and any ODBC data source. Historically, DAO is a popular solution when it comes to using Microsoft Access

(.mdb) and ISAM data sources such as Btrieve, FoxPro, Paradox, and dBase. DAO is tied to the Microsoft Jet engine because it uses the Microsoft Jet engine query and a result set.



CHAPTER 3

DESIGN APPROACH OF THE MU STUDENT DATA MART SYSTEM

This chapter discusses a system overview of Mahidol University (MU) Student Data Mart, describes related techniques which are applied for building a data warehouse and the analysis and detailed design of an MU Student Data Mart. It also presents the overall structure of system modules, system processes and the details of MU Student Data Mart database design

3.1 A Current System and Environment of the MU Student Information System

The Mahidol University (MU) Student Information System was established in 1980 and was expanded continuously to support new requirements and new situations. The system is developed by Mahidol University Computing Center (MUCC) to support daily processes of four divisions, i.e., Finance, Academic Affairs, Student Affairs, and Student Information and Processing Group. In addition, the system supports academic divisions of faculties that are distributed in four campuses, twenty three faculties and 14,000 students in more than one hundred and thirty undergraduate programs. It provides a convenient method of retrieving and updating information for the users. Such a system becomes a computerized system to support and improve the efficiency of services to students better than a manual system does.

The system consists of four main systems categorized by its function as listed below:

- Student Information: the system for maintaining student information and relevant profiles. Query and reporting are included as well. [34, 35]
- Registration System: the system for handling registration information. There are several sub-systems developed to complete the whole

registration process, such as Pre-Enrollment, E-payment, E-Classlist, Randomized Dormitory Accommodation, E-Loan, E-Cheque, Registration Income Allocation and so on.[26, 28, 29, 30, 31, 32, 33]

- Grading System: the system for processing grades information, GPA calculation, and query and reporting. In addition, the system includes a transcript and diploma generating system. [38, 39]
- Curriculum System: the system for maintaining university curriculum, subject and relevant data. [27, 36, 37]

The system is developed to conform the client/server architecture and the web-based environment depending on the nature of application and security concern to support students and registration staff that are distributed in university campuses. The system uses Microsoft SQL Server as a database management system. The system consists of more than sixty programs running on five databases to handle more than nine-gigabyte data.

At present, a large amount of data of the student information system has been established but the decision maker can only partially utilize the detail and many structures of data for decision making. When the decision maker needs summarized information of the system for planning, the reports are executed from large a amount of data, which takes a long time to process. The system lacks of information that supports a decision maker's needs.

The examples of information that are needed to support the decision making are listed below.

- Comparison of the number of admitted and graduate students in each batch.
- Percentage of graduated students compared with admitted students in a year.
- Number of students that enroll for each subject in each semester.
- Number of students and amount of payment in each day and semester.
- Percentage of income that faculty/institute gets from registration allocation.
- Income growth of faculty/institute derived from registration allocation in the past three years.

- Top five faculties/institutes that get maximum registration allocation income.
- Percentage of students that have GPA more than 3.00 or less than 2.00
- The distribution of students with each grade level.
- Etc.

These figures provide information that executive uses for developing and modifying the curriculum to be consistent with the new situation as well as the current problems and trend.

3.2 MU Student Data Mart System Approach

After studying several existing university data warehouse projects and three existing frameworks of data warehouse as described in section 2.2, we design the architecture of Mahidol University data warehouse with a data mart approach, and categorize the process of data warehouse into two groups as listed below.

- Building process of warehouse or ETL process – the back-end process, is responsible for preparing suitable data for a data mart. It is composed of many sub-processes to obtain data quality.
- Using process of warehouse or information delivery – this process provides tools for a decision maker to have an easy access to information for analysis, and presents it in various forms.

3.2.1 The Architecture of MU Student Data Mart

Figure 3.1 shows the architecture of an MU Student Data Mart. We use the current MU student database system as a case study to create Mahidol University Data Warehouse in this research project. This warehouse is designed with a data mart approach because the project focuses only on student information. The student system is composed of an Admission system, a Registration system, a Grading system and a Curriculum system. Our system applies a trigger technique to detect changed data in a source system in order to perform incremental refreshment of the data mart after an initial phase. And we implement the system using a client/server technique for building and maintaining the data mart, and using a web-based technique to deliver information to a user.

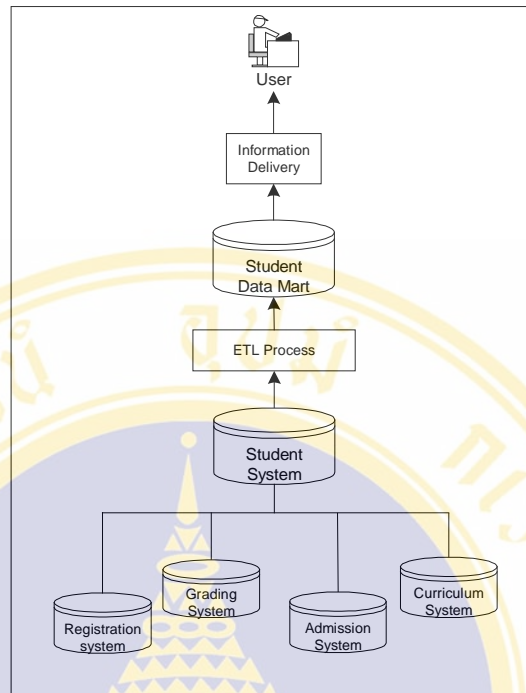


Figure 3.1 : The Architecture of the Mahidol University Student Data Mart Development

Based on Figure 3.1, we can describe the architecture in two major processes. The first one is a process of building and maintaining the data mart, which is called an ETL process. The other process is a using process called information delivery. The detail of each process is described below.

□ ETL Process

Data from a student system are used as the source data and populated into MU Student Data Mart through an ETL process. An ETL (Extract, Transform, Load) process is a data mart building and maintaining process. The process starts with the selected data from a student system being extracted and moved into a data staging area for preparing data before moving it to the data mart. Extracted data are detected and resolved error to increase data quality. Then, data are transformed into a data mart format through a transformation process, which includes convert, add attribute, split/merge attribute, create surrogate key, and summarize sub-processes. Not all data have to be performed in all sub-processes of transformation. It

depends on type of data. Finally, transformed data are loaded into a data mart repository to complete an ETL process.

This process is an iterative process, after an initial phase of building a data mart, as time advances, the source data are changed and then the data mart is updated periodically with changed data. In order to perform incremental refreshment of the data mart, the technique for detecting change is required. The change monitoring is used to detect change and after that the sub-processes of ETL process, as an initial phase are applied. The ETL process is shown in Figure 3.2 and the detail of each sub-process is described as follows:

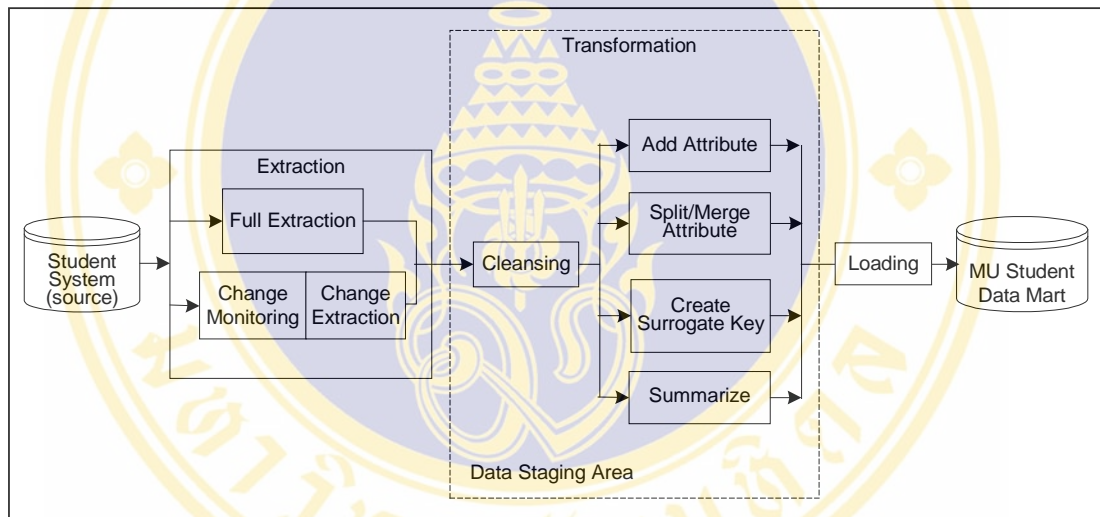


Figure 3.2 : ETL Process

□ Information Delivery

This process makes information available and easily accessible for a user. Many kinds of users need different aspects of queries and reports. For example, a predefined report, an ad hoc report, and complex data analysis, which performs a multidimensional operation: slicing, dicing, roll-up, drill-down and pivoting.

3.2.2 Design of Development Process of MU Student Data Mart

Based on MU Student Data Mart architecture and an ETL Process shown in Figures 3.1 and 3.2, we can explain by using a technique of Use Case Analysis in our research project. Figure 3.3 illustrates a high level use case diagram of the system.

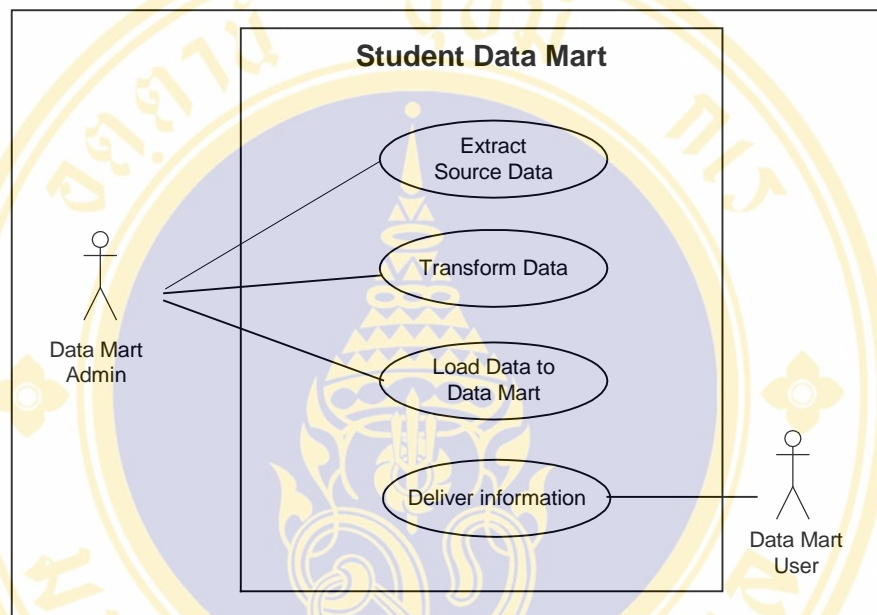


Figure 3.3 : A High Level Use Case Diagram for MU Student Data Mart

As seen in Figure 3.3, our system consists of four functions : extract source data, transform data, load data to a data mart, and use an analysis report. The first three functions are data mart building and maintaining functions. The actors of the system are Data Mart Administrators who perform the work of use case, and Data Mart Users who are receivers of an out case of the system. Table 3.1 provides the detailed specifications of the use case in Figure 3.3.

Table 3.1 Use Case Specifications of MU Student Data Mart

Use Case Name	Extract Source Data
Actors	Data Mart Admin
Entry Condition	Data Mart Admin needs selected source data in the initial phase and when source data is changed for preparing Data Mart according to the design.
Flow of Events	<ol style="list-style-type: none"> 1. Define selected data, trigger rule, schedule for incremental extract. 2. Retrieve data via an extraction module in the initial phase. 3. Deliver selected data to a data staging area. 4. Defined triggers are executed when source data is changed. 5. Retrieve changed data and propagate it into the data mart when a schedule is met.
Exit Condition	Data is propagated into a staging area.
Exceptions	-
Special Requirements	-

Use Case Name	Transform Data
Actors	Data Mart Admin.
Entry Condition	Extracted data is delivered to a staging area.
Flow of Events	<ol style="list-style-type: none"> 1. Define which data is dirty and define source and target data. 2. Clean extracted data. 3. Mapping source data to target data. 4. Convert data in a target format ready for updating data mart.
Exit Condition	Data is converted into a data mart format.
Exceptions	Dirty data that cannot be cleaned will be discarded.
Special Requirements	-

Table 3.1 Use Case Specifications of MU Student Data Mart (cont.)

Use Case Name	Load data to Data Mart
Actors	Data Mart Admin.
Entry Condition	Data is converted into a data mart format.
Flow of Events	<ol style="list-style-type: none"> 1. Ready data is propagated to the data mart. 2. Generate data cube of the data mart. 3. Analysis report is prepared for users.
Exit Condition	The data cube is ready for users
Exceptions	-
Special Requirements	Data mart has to be offline.

Use Case Name	Use Analysis Report
Actors	Data Mart User.
Entry Condition	Users request an analysis report.
Flow of Events	<ol style="list-style-type: none"> 1. Users specify conditions such as dimensions, fact that they need. 2. Retrieve data mart base on the specified conditions.
Exit Condition	Users obtain the result of data.
Exceptions	The function is allowed only if the data mart is available
Special Requirements	-

Based on the ETL Process and first three use cases shown in Figures 3.2 and 3.3, we can divide them into three processes of data mart building as the following :

- Extraction Process.
- Transformation Process
- Loading Process.

Details the of three processes are described below.

3.2.2.1 Extraction

Extraction is the first process of the ETL process used to obtain required data. This process deals with capturing selected data from a source system

and propagating it into a data staging area. In the initial stage, all selected data are performed. For data mart refreshment, only relevant changed data from the source system are captured. The process for detecting change in the source system is required before capturing it to the staging area is called change monitoring. Vavouras A. [6] classified a change monitoring technique for detecting change underlying source system into five techniques listed below:

- Log-based monitoring
- Trigger-based monitoring
- Replication-based monitoring
- Application-assisted monitoring
- Snapshot comparison.

A trigger technique is used to detect changed data in this project. Because the DBMS like SQL Server supports this technique, detecting a change with triggers does not need to modify an existing student information system and less affects current system. When a defined event occurs, and if the defined condition is valid, then the defined action will be executed by the trigger to capture changed data. The details of a change monitoring technique and triggers are described in chapter 2.

There are two types of change extractions: defer extraction and immediate extraction.

□ Defer Extraction

The source data are not immediately extracted when a change occurs in source data. The defined triggers insert reference keys of source data and type of change into a reference table. These triggers are simple and do not affect work load of an operational system. When a schedule of extraction is met or an extraction is requested, the aggregate process is executed to retrieve all relevant data according to reference keys in the reference table, summarize relevant data and deliver it to a data staging area to perform other processes of the ETL process. Finally, clear data of the reference table which has been used for a reference. (Figure 3.4).

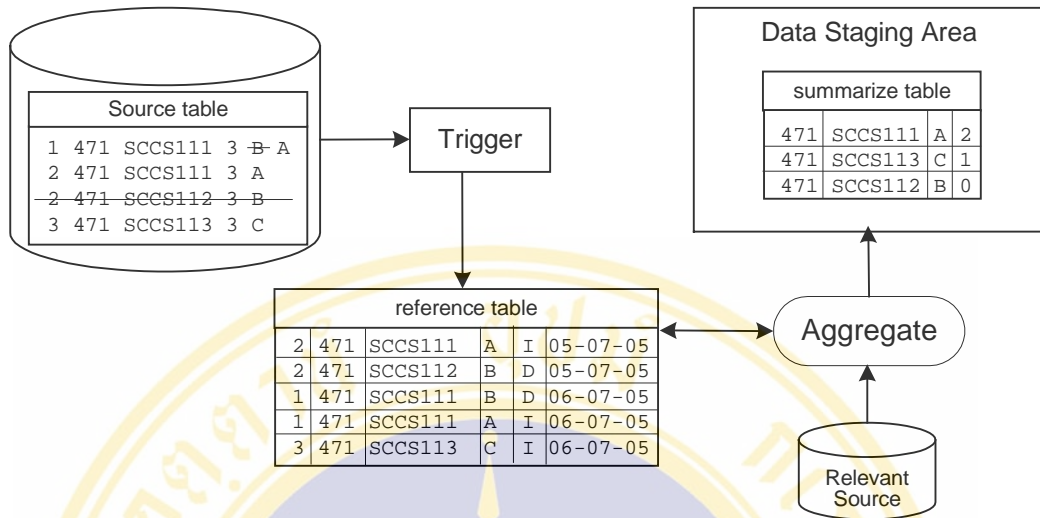


Figure 3.4 : Defer Extraction

For example, when a change occurs in a grade table, the triggers and an algorithm of aggregation are shown in Figure 3.5.

<p>Trigger when data is inserted into Grade table</p> <pre> IF Insert data to Grade table then BEGIN Add selected column from Inserted set and type of operation (I type) into a reference table . END IF </pre>
<p>Trigger when data is deleted from Grade table</p> <pre> IF Delete data from Grade table then BEGIN Add selected column from Deleted set and type of operation (D type) into a reference table. END IF </pre>
<p>Trigger when defined column of Grade table is updated</p> <pre> IF Grade column of Grade table is modified then BEGIN Add selected column from Deleted set and type of operation (D type) into a reference table. Add selected column from Inserted set and type of operation (I type) into a reference table. END IF </pre>

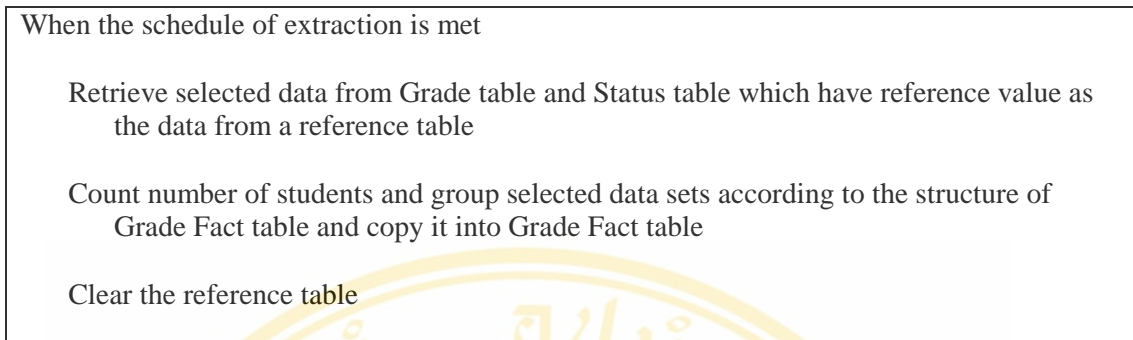


Figure 3.5 : Triggers and Aggregation Algorithm of Defer Extraction

□ Immediate Extraction

When a change occurs in source data, the defined triggers are executed to extract relevant source data and move it to a data staging area immediately. These triggers are complex as they perform many operations which are separated in defer extraction and may retrieve more than one table to complete all tasks at a time. The effect of this extraction may degrade performance of the operational system depending on the complexity of an action in a trigger. However, a trigger may not complete all tasks at a time. Immediate extraction is shown in Figure 3.6.

The examples of triggers of a grade table which perform immediate extraction, and the algorithm are shown in Figure 3.6 and Figure 3.7, respectively.

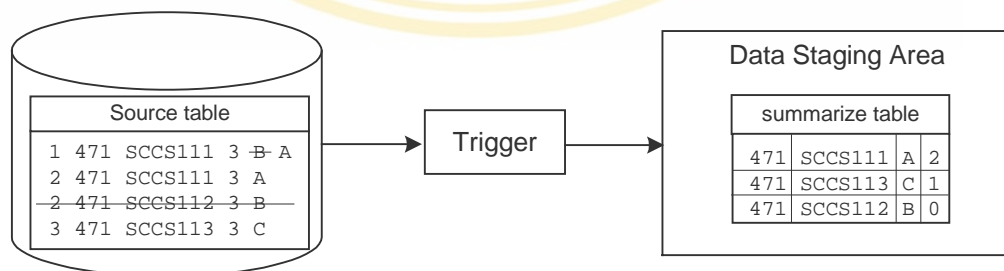


Figure 3.6 : Immediate Extraction

<p>Trigger when data is inserted into Grade table</p> <p>IF Insert data to Grade table then BEGIN Delete Grade Fact table which has reference keys as Inserted set's key.</p> <p>Retrieve selected data from Grade table and Status table which have reference value as the data from an inserted set Group selected data sets according to the structure of Grade Fact table and copy it into Grade Fact table</p> <p>END IF</p>
<p>Trigger when data is deleted from Grade table</p> <p>IF Delete data from Grade table then BEGIN Delete Grade Fact table which have reference key as Deleted set's key.</p> <p>Retrieve selected data from Grade table and Status table which have reference value as the data from a Deleted set Group selected data sets according to the structure of Grade Fact table and copy it into Grade Fact table</p> <p>END IF</p>
<p>Trigger when data is updated in Grade table</p> <p>IF Update data in Grade table then BEGIN Delete Grade Fact table which have reference key as Deleted set's key.</p> <p>Retrieve selected data from Grade table and Status table which have reference value as the data from a Deleted set Group selected data sets according to the structure of Grade Fact table and copy it into Grade Fact table</p> <p>Delete Grade Fact table which have reference key as Inserted set.</p> <p>Retrieve selected data from Grade table and Status table which have reference value as the data from inserted set Group selected data sets according to the structure of Grade Fact table and copy it into Grade Fact table</p> <p>END IF</p>

Figure 3.7 : Triggers Algorithm of Immediate Extraction

Two types of change extractions are used in this project depending on the kind of source table. The detailed use case diagram of the extraction process is depicted in Figure 3.8. The diagram contains three use cases : full extraction, change monitoring and change extraction. There are two ways to extract changed data: defer extraction and immediate extraction, but only one way is used to

extract changed data. Table 3.2 provides the detailed specifications of the use case in Figure 3.8.

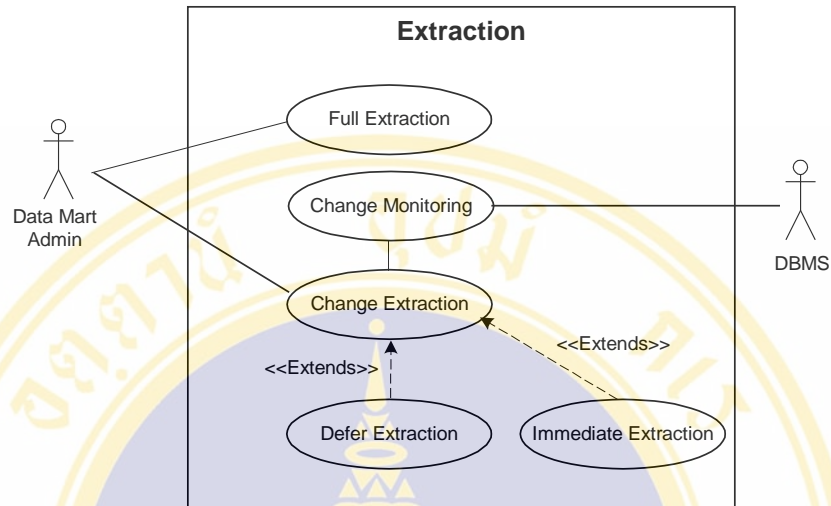


Figure 3.8 : A Detailed Use Case Diagram of Extraction

Table 3.2 Use Case Specification of Extraction

Use Case Name	Full Extraction
Actors	Data Mart Admin.
Entry Condition	Data Mart Admin needs selected source data in the initial phase for preparing data mart according to the design.
Flow of Events	<ol style="list-style-type: none"> 1. Define selected data. 2. Retrieve data via and extraction module. 3. Deliver selected data to a data staging area.
Exit Condition	Data is propagated into the staging area.
Exceptions	-
Special Requirements	-

Use Case Name	Change Monitoring
Actors	DBMS
Entry Condition	Source data is changed.
Flow of Events	<ol style="list-style-type: none"> 1. Define selected data, trigger rules. 2. Defined triggers are executed when source data are changed.

Table 3.2 Use Case Specification of Extraction (cont.)

Flow of Events	3. If the conditions are valid, then the defined actions will be executed to capture changed data.
Exit Condition	Data is sent to change extraction module.
Exceptions	-
Special Requirements	-

Use Case Name	Change Extraction
Actors	Data Mart Admin.
Entry Condition	Schedule of extraction.
Flow of Events	<ol style="list-style-type: none"> 1. Perform defer extraction or immediate extraction depending on type of table in order to retrieve data. 2. Propagate changed data into a staging area when the schedule is met.
Exit Condition	Data is propagated into a staging area.
Exceptions	-
Special Requirements	-

Use Case Name	Defer Extraction
Actors	Data Mart Admin.
Entry Condition	Schedule of extraction.
Flow of Events	<ol style="list-style-type: none"> 1. Insert changed data into a reference table. 2. When a schedule is met, retrieve relevant data according to reference data and perform other operations to complete the process. 3. Propagate data into a staging area.
Exit Condition	Data is propagated into a staging area.
Exceptions	-
Special Requirements	-

Table 3.2 Use Case Specification of Extraction (cont.)

Use Case Name	Immediate Extraction
Actors	Data Mart Admin.
Entry Condition	Source data is changed.
Flow of Events	1. Retrieve changed data via the triggers and perform other operations to complete the process. 2. Propagate data into a staging area.
Exit Condition	Data is propagated into a staging area.
Exceptions	-
Special Requirements	-

3.2.2.2 Cleansing and Transformation

Extracted data must be integrated from various tables. It may contain errors, inconsistent or incomplete data. It must be cleaned before being loaded into an MU Student Data Mart. The data must be transformed into a standard format and structure appropriate for begin stored in the data mart and the analytical purpose. A transformation process contains four tasks: add attribute, split/merge attribute, create surrogate key, and summarize. These tasks may be performed concurrently while extracting data, and not all kinds of data have to perform all tasks.

□ Cleansing

A data mart is used for decision making, and so the data in a data mart must be consistent and correct. However, integrating data from multiple tables may generate inconsistency or an error. This task deals with detecting major errors and removing them from the data to increase data quality. An error in data is defined and a cleansing rule is applied to resolve it depending on the type of error. For example, some missing data can be replaced with a default value.

□ Add Attribute

Some attributes are not contained in source data but they are necessary for a data mart. This task performs calculation to get a new attribute for an existing attribute.

- Split/Merge Attribute

A single attribute from source data may be split into more than one attribute in a data mart. For example, payment data is separated into three attributes of day, month and year.

In an opposite way, several attributes of source data are merged into a single attribute of a data mart.

- Create Surrogate key

A surrogate key is the primary key for a dimension table and differs from the keys in source data systems. The original key is contained in the dimension table but it is not used for a primary key. A surrogate key is used to maintain data warehouse information when dimensional data is changed. Automatically increasing integers make a good surrogate key.

- Summarize

Creating a summary of data and loading it into a data mart instead of loading low level of detail of all transactions is not suitable for analysis. Data is summarized at lowest granularity for analysis. For example, for a registration information, the number of students and summary of invoice amount are grouped by payment date and by type of payment

The detailed use case diagram of a cleansing & transformation process is depicted in Figure 3.9. The diagram contains five use cases : clean, add attribute, split/merge attribute, create a surrogate key and summarize. Table 3.3 provides the detailed specifications of the use case in Figure 3.9.

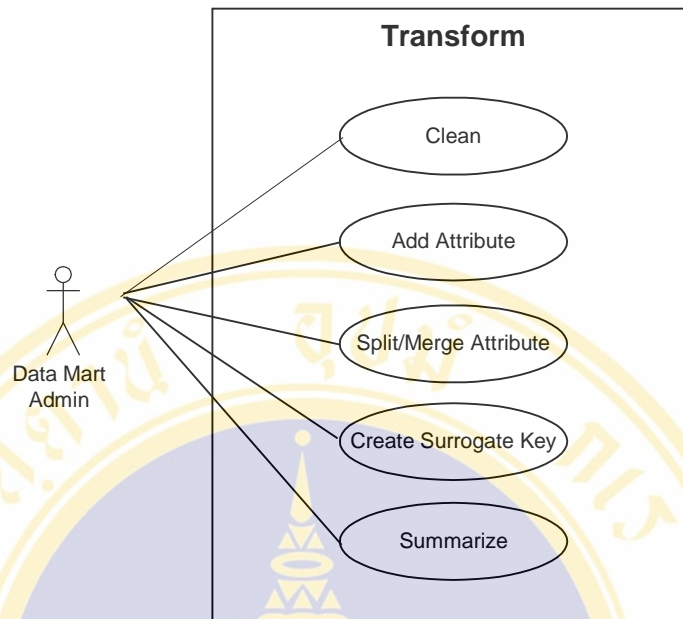


Figure 3.9 : A Detailed Use Case Diagram of Cleansing & Transformation

Table 3.3 Use Case Specification of Cleansing & Transformation

Use Case Name	Clean
Actors	Data Mart Admin.
Entry Condition	Extracted data is propagated into a staging area.
Flow of Events	<ol style="list-style-type: none"> 1. Define errors in data, cleansing rule. 2. Detect errors according to the definition. 3. Correct errors depending on types of errors.
Exit Condition	Data is clean.
Exceptions	Some data may be discarded depending on types of errors.
Special Requirements	-

Table 3.3 Use Case Specification of Cleansing & Transformation (cont.)

Use Case Name	Add Attribute
Actors	Data Mart Admin.
Entry Condition	Extracted data is passed to a transformation process or propagated into a staging area.
Flow of Events	<ol style="list-style-type: none"> 1. Define formulas to obtain needed data. 2. Execute the formulas 3. Store derived data.
Exit Condition	Derived data is stored.
Exceptions	-
Special Requirements	-

Use Case Name	Split/Merge Attribute
Actors	Data Mart Admin.
Entry Condition	Extracted data is passed to a transformation process or propagated into a staging area.
Flow of Events	<ol style="list-style-type: none"> 1. Define source attribute, target attributes mapping functions. 2. Execute the mapping functions to obtain target values.
Exit Condition	Target value is stored.
Exceptions	-
Special Requirements	-

Use Case Name	Create Surrogate Key
Actors	Data Mart Admin.
Entry Condition	Extracted data is passed to a transformation process or propagated into a staging area.
Flow of Events	<ol style="list-style-type: none"> 1. Define keys and primary keys of a dimension table. 2. Run the primary keys. 3. Store the surrogate keys
Exit Condition	Surrogate keys are stored.

Table 3.3 Use Case Specification of Cleansing & Transformation (cont.)

Exceptions	-
Special Requirements	-

Use Case Name	Summarize
Actors	Data Mart Admin.
Entry Condition	Extracted data is passed to a transformation process or propagated into a staging area.
Flow of Events	<ol style="list-style-type: none"> 1. Define aggregation functions, source data. 2. Process aggregation function. 3. Store the result.
Exit Condition	Summarized data is stored.
Exceptions	-
Special Requirements	-

3.2.2.3 Loading

After the data is transformed into a data mart format, the last process for building a data mart is loading it into a data mart. In the initial phase, transformed data in a staging area are propagated into the data mart. Propagating the large volume of data is a time-consuming process. After that, the data mart needs to be updated with changed data from an operational source. Incremental load or full refreshment as Paulraj Ponniah's [4] defined types of load is applied. For incremental load, if primary of a new transformed record matches the key of an existing record in the data mart, the record will be updated with the new record and the data which is not duplicated will append to the data mart. For full refreshment, the process is similar to that of initial load but the existing data in the data mart must be deleted before loading newly transformed data into the data mart.

Figure 3.10 shows a comparison of three existing frameworks (as described in section 2.2.2) and the designed process of an MU Student Data Mart.

Processes in Data Warehouse			
Paulraj's Framework	Mattison's Framework	Singh's Framework	MU Student Data Mart
Data acquisition	Acquisition component	Access	Building process (ETL Process)
- Extracting - Moving - Transformation - Cleansing - Integration	- Extraction - Cleansing - Formatting - Merging & Purging - Key processing	- Operational and External Data	- Change Monitoring - Extraction - Cleansing - Transformation - Integration - Summarization - Aggregation - Loading
Data storage	Storage Component	Transform	
- Full refresh - Incremental load - Backup/Recovery - Security - Data Archival	- Metadata management - Data synchronization - Data transformation mapping	- Cleanse - Reconcile - Enhance - Summarize - Aggregate	
Information Delivery	Access component	Distribute	Information Delivery
- Query government & optimization - Content browse - Security control - Self-Service report generation	- Report generating - Visualizagion - Simulation	- Stage - Join Multiple Sources - Populate on demand	- Analysis - Query - Presentation - Access Control
		Store	
		- Relational Data - Specialized Caches - Multiple Platforms & Hardware	
		Find	
		- Information Catalog - Business Views - Models	
		Display & Analyze	
		- Query and Reporting - Multi-Dimensional Analysis - Data Mining	

Figure 3.10 : Comparison of Three Existing Frameworks and MU Student Data Mart

3.2.3 Design of Data Refreshment Process

The data mart refreshment is a repetitive process which performs the ETL (Extraction, Transformation and Loading) task. When Mahidol University student information are changed, the contents of student data mart must be periodically refreshed to reflect the content of student information. Reloading all student information as the initial phase of building the data mart is a time-consuming task and loss historical information, then incremental refresh the data mart is required.

In order to incrementally refresh the data mart, change monitoring is added in the extraction process as shown in Figure 3.2. The goal of change monitoring is detecting relevant change in student information. This research project has applied a trigger technique for detecting changed data from a source in order to update the related data mart.

We design three types of triggers, namely: Insert, Update and Delete Triggers. These triggers are applied to interested data in student information system in order to detect change of data which come from university operations. These data consist of student data, invoice and registration data, allocation data, grade data and curriculum data. The details of data are described in section 3.5. We design three types of triggers in forty-five rules to detect change of fifteen tables of student information system.

When defined events of a source table such as insert, update and delete record occur (for example, a new student data is inserted into student table), then Insert Trigger of student table is executed and the conditions are verified. If the condition is valid, then the trigger automatically executes an action of inserting a record into a staging table, which is used for an extraction and transformation process.

Change monitoring is a part of extraction for refreshment in order to detect changed data. These changed data are extracted by change extraction the detail of which is described in section 3.2.2.1. The two types of change extractions namely: defer extraction and immediate extraction have different trigger rules. For defer extraction, when source table is changed, the trigger inserts reference key of source table and type of change into reference table. This table is used for the aggregate process in order to complete the extraction when schedule of extraction is met or an extraction is requested. For the immediate extraction, when a change occurs in the source table, the trigger captures data source and relevant data into a staging table to complete immediate extraction. In order not to degrade performance of student information system, we design an action of trigger as simple as possible but it can capture change of interested data and send this data to other processes.

Extracted data are transformed into a standard format and propagated into the data mart repository. These processes perform similar to an initial ETL for building the data mart. Moreover, the data cube is processed in order to read the dimension tables and the fact table, calculate aggregations and store the results in the cube.

3.2.4 Design of MU Student Data Cube

This research project uses star schema to design eight data cubes as listed below:

1. Student Cube shows the number of admitted and graduated students.
2. Student Graduation Cube shows the productivity of university with the number of graduated students.
3. Registration Cube shows the registration amount.
4. Registration Subject Cube shows the number of enrolled students and the tuition fee of subjects.
5. Allocation Cube shows allocated registration income.
6. Program Allocation Cube shows registration income that students for each program pay and the income is allocated to a faculty.
7. Grading Cube shows the number of students that get grade in each subject.
8. GPA Cube shows student productivity with their GPA.

These cubes carry information about university operations. Each cube consists of a single fact table and several dimension tables. The fact table contains foreign keys that link with primary keys of dimension tables and contains numerical data called measures. The measures are summarized to provide university operations. The dimension table is organized in hierarchical structure which is used for drill-down and roll-up operations. The detail of each cube is described in section 4.4.

3.3 Structure of System Modules

Figure 3.11 shows an overall structure of modules of an MU Student Data Mart system based on the work flow and use case analysis discussed in chapter 3. The system consists of two modules; namely, ETL and information delivery. A trigger technique is applied to detect changed data from an operational source in order to incrementally update the data mart, which is a sub-process of ETL. A web-based technique is applied to develop the system for delivery of information to a user in a web-based environment. The analysis and detailed design of these modules will be presented in the subsequent sections.

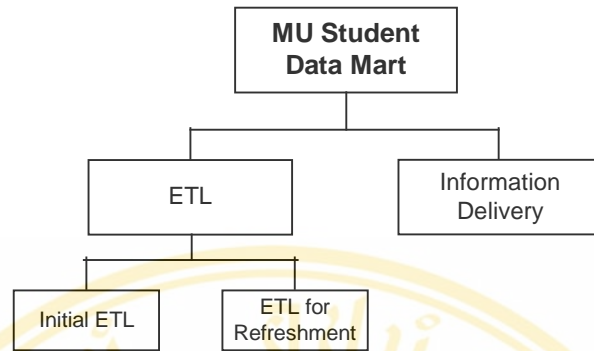


Figure 3.11 : A Structure Chart of Core Modules of MU Student Data Mart

3.4 System Processes

A context diagram of an MU Student Data Mart reveals a guideline to develop an MU Student Data Mart as shown in Figure 3.12.

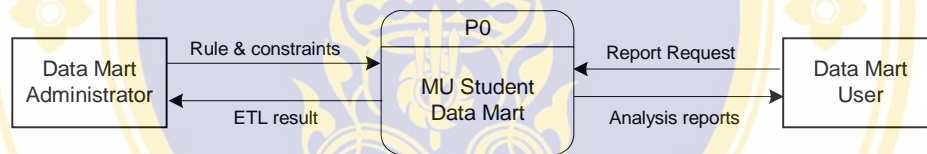


Figure 3.12 : A Context Diagram of MU Student Data Mart

As mentioned in the previous section, the process of the system developed in this research project can be divided into two major processes, namely, an ETL process and an information delivery process, of which details are described below.

P1 ETL process

This process is used to prepare information for a data mart by extracting selected data from an operational source and converting it into a suitable format of data mart before loading it into data mart storage.

P2 Information delivery process

It is a process used by data mart users. They retrieve analyzed information that is provided in several reports.

The two main processes of MU Student Data Mart can be elaborated in a data flow diagram level 1 as shown in Figure 3.13.

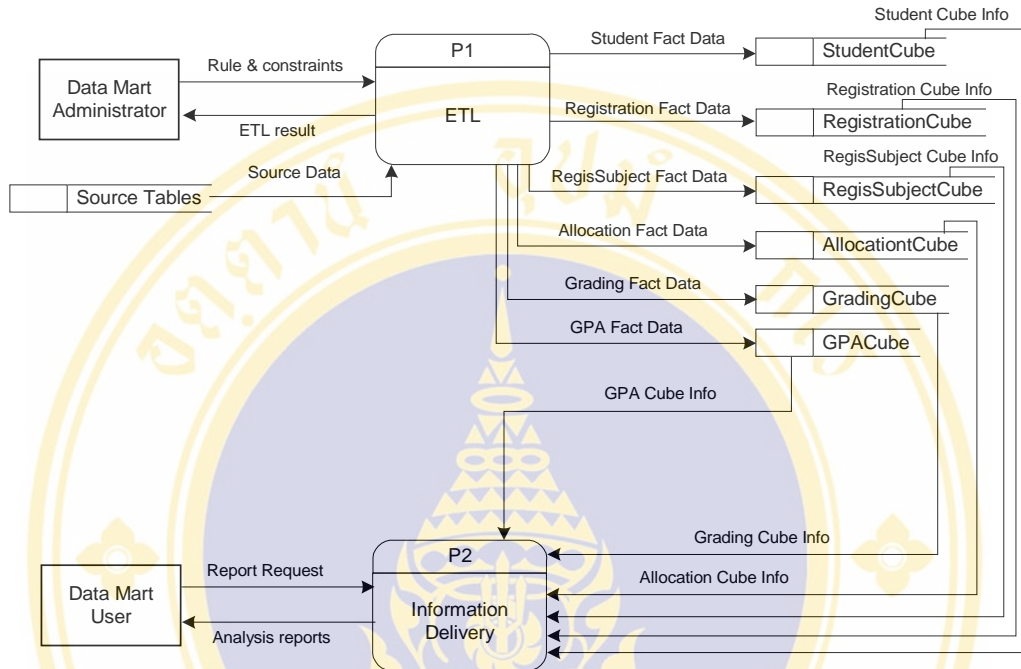


Figure 3.13 : A Data Flow Diagram Level 1 of MU Student Data Mart

3.4.1 ETL Process

An ETL process is a process for building and maintaining a data mart. The process starts from extracting selected data from an operational source, transforming selected data into an appropriate format of data mart and loading the information into a data mart repository, which is ready for analysis. As shown in Figure 3.11, an ETL process is derived in two sub-processes namely; initial ETL and ETL for refreshment. Details of each sub-process are described below.

P1.1 Initial ETL

This is used for building a data mart. The student information is extracted, transformed into a data mart format and loaded into data mart storage. These data are used to generate a data cube.

P1.2 ETL for Refreshment

This process is performed after the data mart is built. As time advances, the student information are changed then the data mart is updated. This process performs like as the initial ETL but extracts only relevant changed data to incrementally refresh the data mart.

Based on the two sub-processes of ETL, they can be written in a data flow diagram level 2 in Figure 3.14.

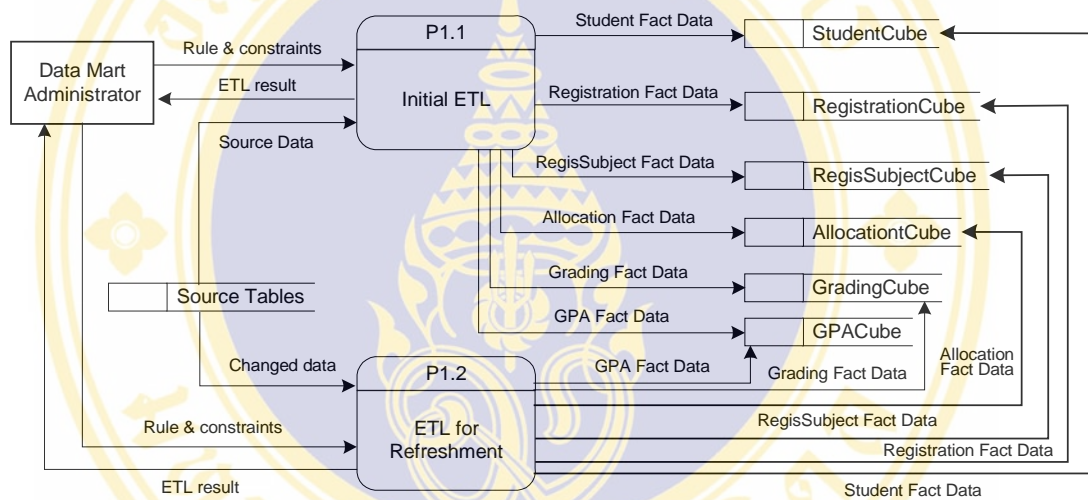


Figure 3.14 : A Data Flow Diagram Level 2 of ETL Process

3.4.1.1 Initial ETL

The initial ETL process consists of three main sub-processes; namely, extraction, transformation and loading, as shown in Figure 3.15.

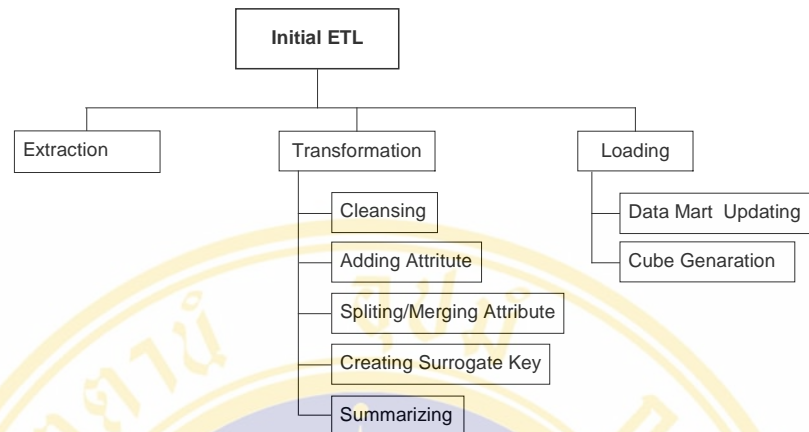


Figure 3.15 : A Structure Chart of Initial ETL Process

The Initial ETL process consists of three main sub-processes; namely, extraction, transformation and loading. Details of each sub-process are described below.

P1.1.1 Extraction

It is a process for retrieving selected data from an operational source. In the initial ETL process all needed data are retrieved, so we call full extraction. Selected data are sent to a next process and transformation.

P1.1.2 Transformation

It is a process for converting extracted data into a suitable format of data mart. Some kinds of data may need to be performed for several functions of transformation to complete the process.

P1.1.3 Loading

It is a process for populating suitable data to a data mart repository and generating a data cube which contains summarized data in the multidimensional structure.

Based on the three working functions of the initial ETL process, they can be written in a data flow diagram level 3 illustrated in Figure 3.16.

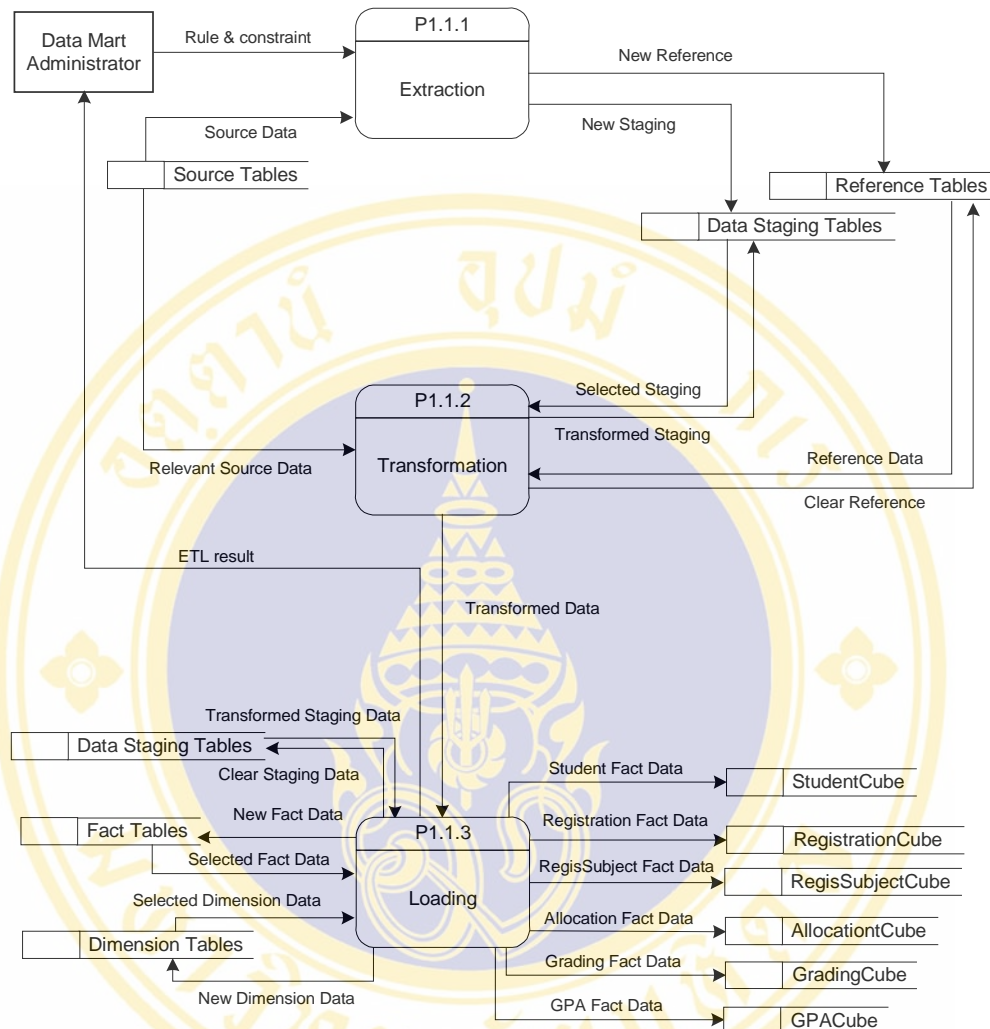


Figure 3.16 : A Data Flow Diagram Level 3 of Initial ETL Process

The transformation process consists of five sub-processes; namely, cleansing, adding attribute, splitting/merging attribute, creating a surrogate key and summarizing. Details of each sub-process are described below.

P1.1.2.1 Cleansing

The aggregated data from multiple tables may contain errors or incomplete data. This sub-process deals with detecting major errors and removing them from the data. For instance, missing data will be replaced by a default value and duplicate records are eliminated.

P1.1.2.2 Adding Attribute

This sub-process deals with calculating a new value for an attribute which is not contained in source data. For example, FacultyCode and Faculty column are added to Subject dimension by deriving from two left characters of SubjectCode and joined with Faculty table to get faculty name, consecutively.

P1.1.2.3 Splitting/Merging Attribute

This sub-process integrates two or more attributes into one attribute and separates a single attribute into more than one attribute. For example, SemesterID column, which contains academic year and semester, is separated into two columns of AcdYear and Semester.

P1.1.2.4 Creating A Surrogate Key

The key from a source cannot be used as a primary key of a data mart. This process finds out a new primary key of the data mart.

P1.1.2.5 Summarizing

It is a process of summarizing extracted data which have a low level of detail into summarized data and storing it in a data staging area.

Based on the five sub-processes of a transformation process, they can be written in a data flow diagram level 4 illustrated in Figure 3.17.

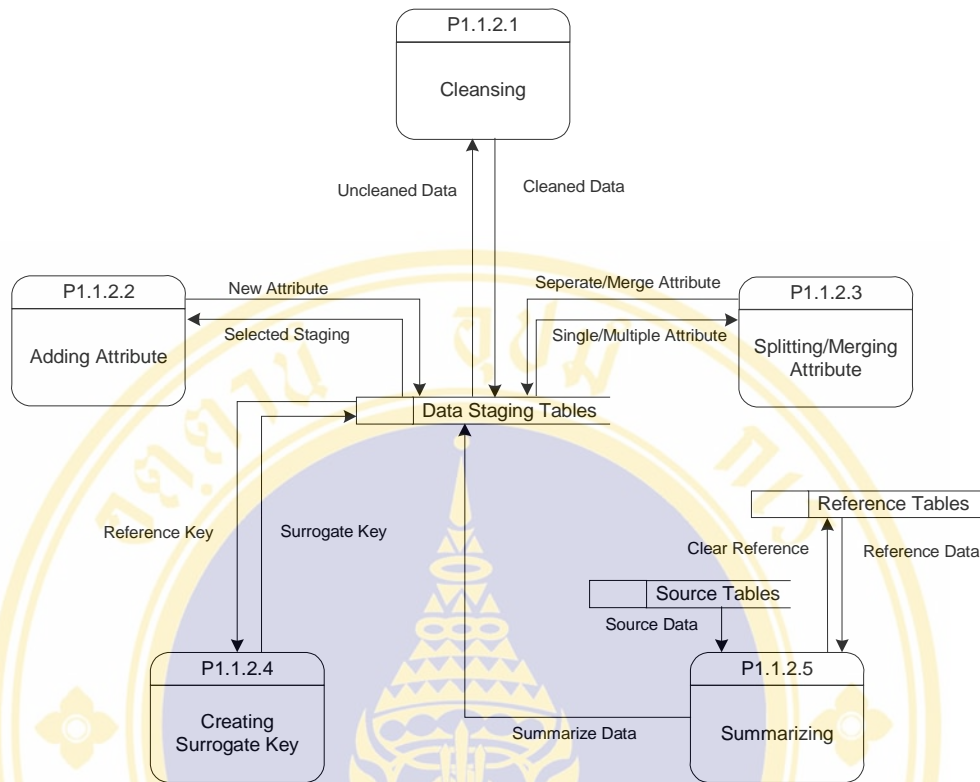


Figure 3.17 : A Data Flow Diagram Level 4 of Transformation Process

The loading process consists of two sub-processes; namely, data mart updating and cube generation. Details of each sub-process are described below.

P1.1.3.1 Data Mart Updating

It is a process for propagating staging data into data mart storage, which is used as a source of the cube and clear staging data in order to prepare storage for new changed data.

P1.1.3.2 Cube Generation

It is a process for making a cube which has multi-dimensional structure that lets users analyze data from many different perspectives. After defining a new cube to select a fact table as a source of the cube, identified measurement and dimension, the cube must be processed with the **Full process**

option. After its source data is changed, the cube must be processed again with **Incremental update** or **Refresh data** options.

Based on the two sub-processes of a loading process, they can be written in a data flow diagram level 4 illustrated in Figure 3.18.

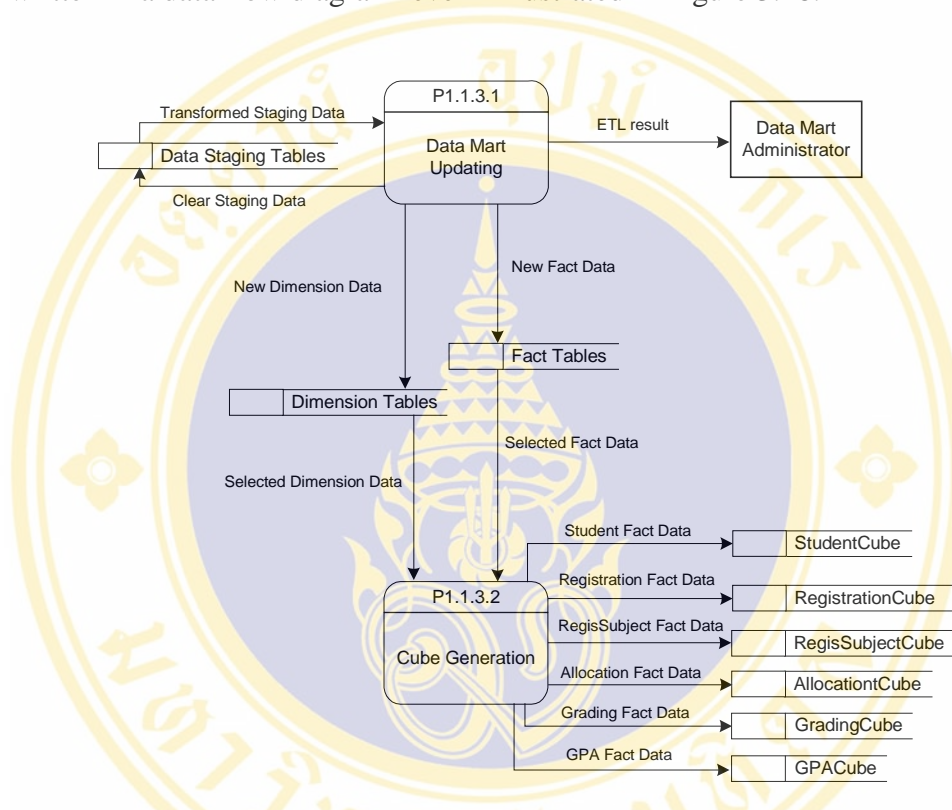


Figure 3.18 : A Data Flow Diagram Level 4 of Loading Process

3.4.1.2 ETL for Refreshment

This process consists of three main sub-processes; namely, extraction, transformation and loading, as shown in Figure 3.19.

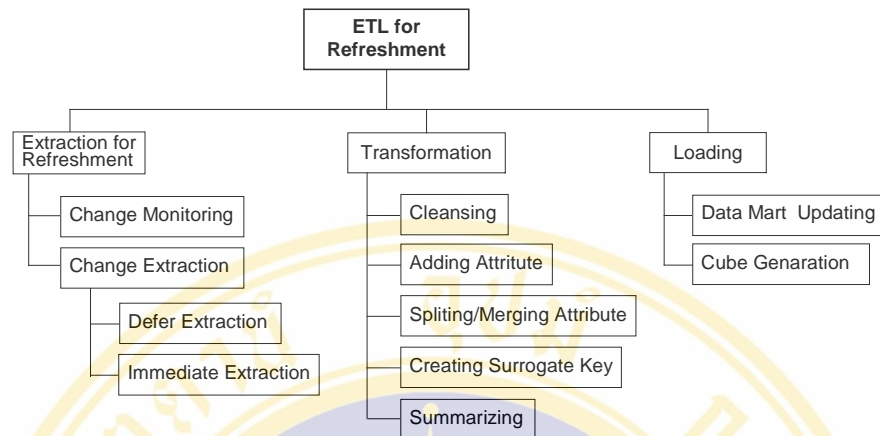


Figure 3.19 : A Structure Chart of ETL for Refreshment Process

P1.1.4 Extraction for Refreshment

This is a process for retrieving changed data and relevant data. It composes of change monitoring mechanism to detect changed data and retrieve it for sending to a next process.

The transformation of ETL for refreshment performs like the processes of initial ETL in order to convert relevant changed data into appropriate format of the data mart. These data are populated to the data mart repository and the data cubes are processed to incrementally update. It is performed by loading process.

The extraction for refreshment process consists of two sub-processes; namely, change monitoring and change extraction. Details of each sub-process are described below.

P1.2.1.1 Change Monitoring

This process is executed when changed data occur, then the defined triggers rules are verified. If the conditions are valid, then the defined actions will be executed to capture changed data.

P1.2.1.2 Change Extraction

It is a process for extracting changed data and relevant data from an operational source and populating it into reference tables or data staging tables depending on the kind of source table.

Based on the two sub-processes of the extraction for refreshment process, they can be written in a data flow diagram level 4 illustrated in Figure 3.20.

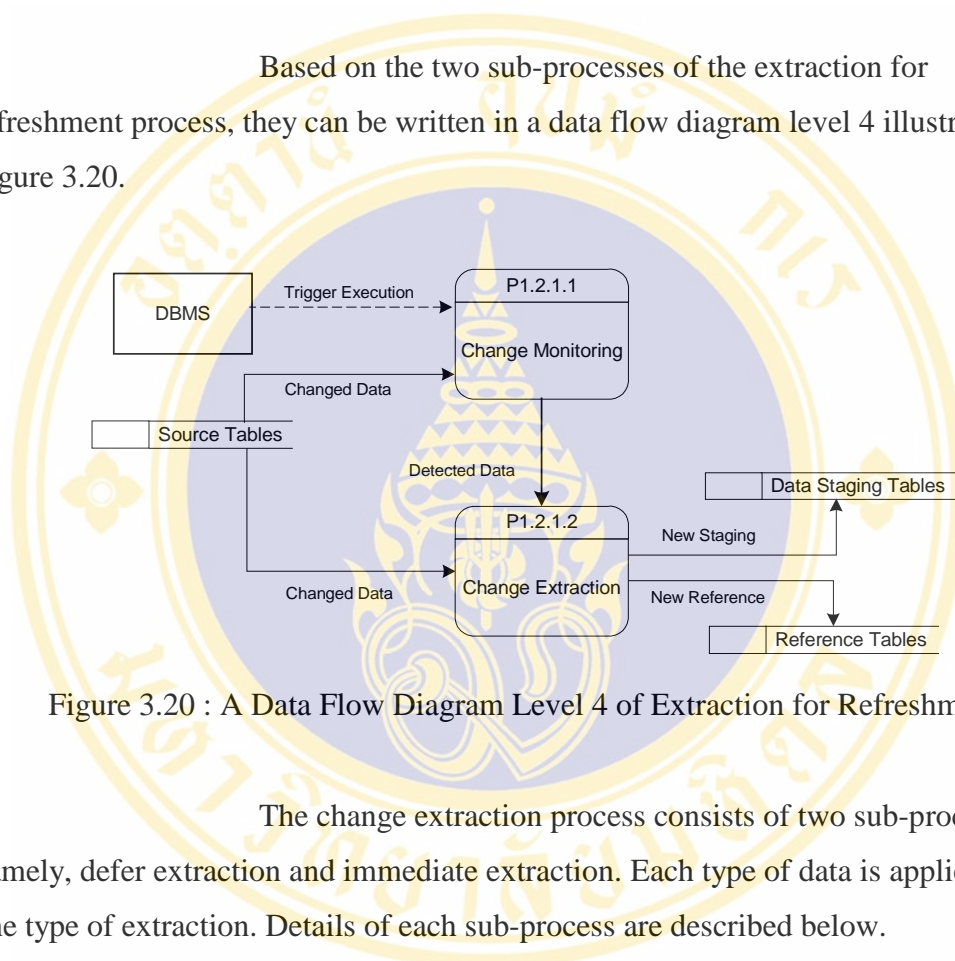


Figure 3.20 : A Data Flow Diagram Level 4 of Extraction for Refreshment

The change extraction process consists of two sub-processes; namely, defer extraction and immediate extraction. Each type of data is applied only one type of extraction. Details of each sub-process are described below.

P1.2.1.2.1 Defer Extraction

It is a process for extracting changed data from an operational source and populating it into reference tables. Until the schedule of extraction is met or extraction is required, reference tables are used to refer and extract all relevant data.

P1.2.1.2.2 Immediate Extraction

Immediate extraction is a process for extracting changed data from an operational source and immediately performing other functions in order to gain needed data.

Based on the two sub-processes of a change extraction process, they can be written in a data flow diagram level 5 illustrated in Figure 3.21.

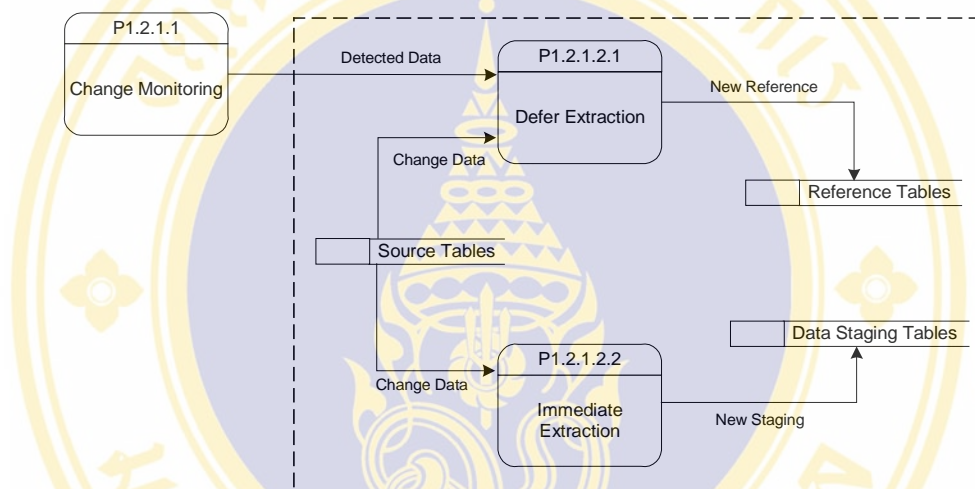


Figure 3.21 : A Data Flow Diagram Level 5 of Change Extraction Process

3.4.2 Information Delivery

An information delivery process is for generating an analytical report to users. This process uses a data cube to represent information in a multidimensional view. Users can use cube operations such as roll-up, drill-down, and slice and dice to view the information. A structure chart of Information Delivery is shown in Figure 3.22.

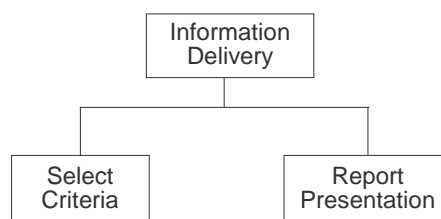


Figure 3.22 : A Structure Chart of Information Delivery

The information delivery process consists of two sub-processes; namely, select criteria and report presentation. Details of each sub-process are described below.

P2.1 Select Criteria

It is a process for defining dimensions, measurement and operation that will be presented in the report.

P2.2 Report Presentation

This process delivers information to users according to their requests. It retrieves the information from a cube and represents it in an analytical format which is easy to understand.

Based on the two working functions of the information delivery process, they can be written in a data flow diagram level 2 illustrated in Figure 3.23.

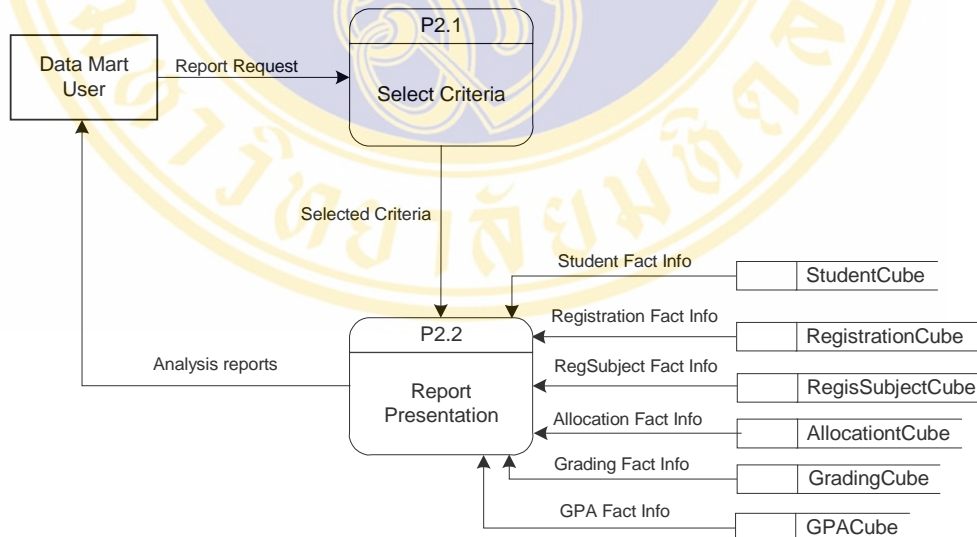


Figure 3.23 : A Data Flow Diagram Level 2 of Information Delivery Process

3.5 Database Design for MU Student Data Mart

After designing a process of MU Student Data Mart discussed earlier, this section presents database design of the system. The data in the system is divided into two parts as the following :

- Data source of MU Student Data Mart : Student Information system
- Data on MU Student Data Mart

3.5.1 Data Source of MU Student Data Mart

This section describes the data from Student Information System, which is used as the operational source of MU Student Data Mart. These data consist of student data, invoice and registration data, allocation data and grade data. Their relationship is displayed in Figure 3.24. The summarized table and its schemas are shown in this section.

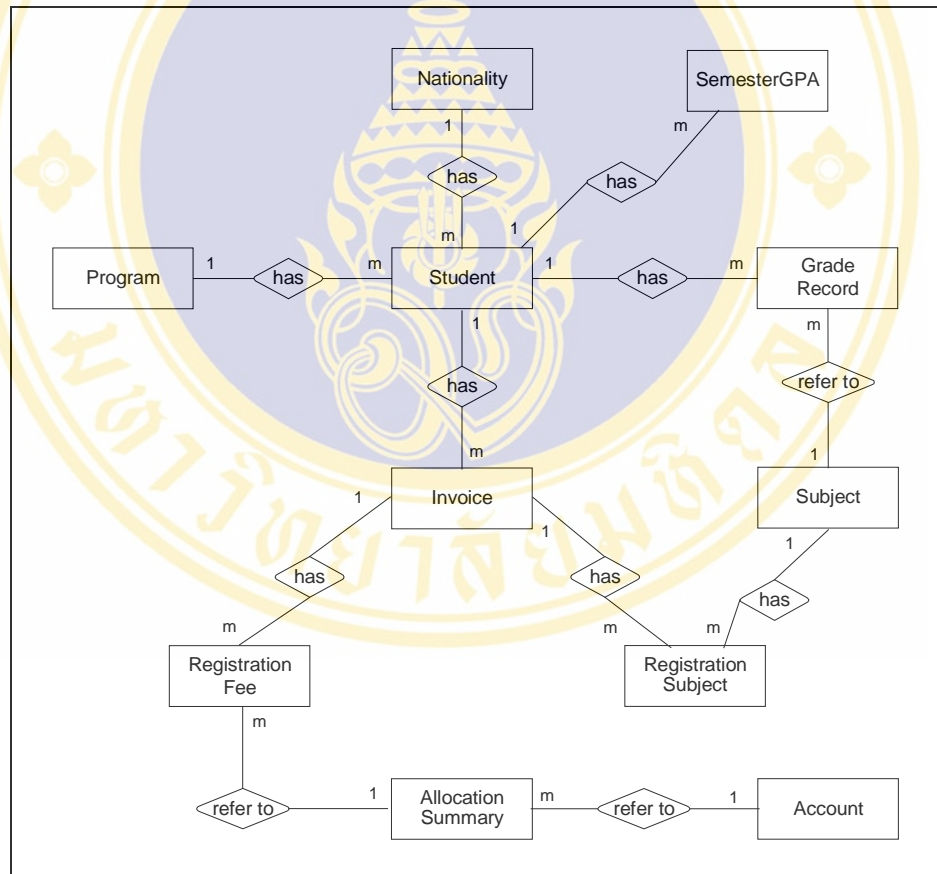


Figure 3.24 : An ER Diagram of MU Student Information System

3.5.1.1 Table of Student Information System

Figure 3.24 displays a conceptual view of a student information system while Table 3.4 summarizes data displaying a relationship, which is classified into three types of master files, transaction files and table files.

Table 3.4 Student Information System.

No	Name	Description	Type
1	Student	Records of student information.	Master
2	CurInvoice	Records of invoice for current semester	Transaction
3	Minvoice	Records of invoice for historical registration.	Transaction
4	CurFeeInvoice	Records of invoice with detail of fee type for current semester	Transaction
5	MfeeInvoice	Records of invoice with detail of fee type for historical registration.	Transaction
6	CurRegisSubject	Records of invoice with enrolled subject for current semester	Transaction
7	MregisSubject	Records of invoice with enrolled subject for historical registration.	Transaction
8	AllocationSummary	Records of registration income which are allocated to each account for each semester.	Transaction
9	ICTAllocationSummary	Records of registration income of ICT program which are allocated to each account for each semester.	Transaction
10	ProgramAllocation	Records of registration income which are allocated from program to each account for each semester.	Transaction
11	Grade	Records of grade of enrolled subjects of students.	Transaction
12	SemesterGPA	Records of GPA of a student in each semester.	Transaction
13	Gender	Records of gender.	Table file
14	Nationality	Records of nationality	Table file
15	Program	Records of academic program	Master
16	PaymentType	Records of payment type	Table file
17	Subject	Records of subject	Master
18	RegistrationType	Records of registration type for each subject.	Table file
19	GradeType	Records of grade that a students gets for each enrolled subject.	Table file
20	GPARange	Records of GPA range.	Table file
21	Account	Records of account that gets registration allocation	Table file

3.5.1.2 Database Schema of Student Information System

Based on Table 3.4, we can display the components of each table in various fields (the underlined attributes mean primary keys while italic attributes mean foreign keys). Detail of table structure will be given in section 3.5.3

Table Name : Student

<u>StudentID</u>	<i>TitleCode</i>	FirstName	LastName	ThaiFName
ThaiLName	Gender	BirthDate	<i>ProgramCode</i>	<i>MajorCode</i>
<i>GroupNum</i>	<i>NationalityCode</i>	AdmissionDate	StudentStatus	

Table Name : CurInvoice

<u>Semester</u>	<u>InvoiceNo</u>	<i>StudentID</i>	Class	RegisDate
InvoiceAmount	InvStatus	<i>MoneyType</i>	PaidStatus	PaidAmount
PaidDate	RegisSts	BillNo		

Table Name : Minvoice

<u>Semester</u>	<u>InvoiceNo</u>	<i>StudentID</i>	Class	RegisDate
InvoiceAmount	InvStatus	<i>MoneyType</i>	PaidStatus	PaidAmount
PaidDate	RegisSts	BillNo		

Table Name : CurFeeInvoice

<u>Semester</u>	<u>InvoiceNo</u>	<u>FeeTypeCode</u>	<i>StudentID</i>	FeeAmount
CancelSts				

Table Name : MfeeInvoice

<u>Semester</u>	<u>InvoiceNo</u>	<u>FeeTypeCode</u>	<i>StudentID</i>	FeeAmount
CancelSts				

Table Name : CurRegisSubject

<i>StudentId</i>	<u>Semester</u>	<u>SubjectCode</u>	<u>InvoiceNo</u>	Credit
RegisStatus	Grade	MoneyStatus	CancelSts	

Table Name : MRegisSubject

<u>StudentId</u>	<u>Semester</u>	<u>SubjectCode</u>	<u>InvoiceNo</u>	Credit
RegisStatus	Grade	MoneyStatus	CancelSts	

Table Name : AllocationSummary

<u>Semester</u>		<u>LotNo</u>		<u>Allocatedate</u>		<u>AccountCode</u>		<u>TotalAmount</u>	
A01	A02	A03	A04	B01	B02	B03	B04	B05	B06
B07	B08	B09	C11	C12	C31	C32	D02	D03	D04
D05	E01	M01	M02	M03	M04	M06	M07	M08	M09
M10	M11	M12	M13	M14	M16				

Table Name : ICTAllocationSummary

<u>Semester</u>		<u>LotNo</u>		<u>Allocatedate</u>		<u>AccountCode</u>		<u>TotalAmount</u>	
A01	A02	A03	A04	B01	B02	B03	B04	B05	B06
B08	B09	B10	B11	C11	C12	C31	C32	D01	D02
D03	D04	D05	E01	M15					

Table Name : ProgramAllocation

<u>Semester</u>	<u>LotNo</u>	<u>ProgramCode</u>	<u>MajorCode</u>	<u>GroupNum</u>	<u>Class</u>
<u>AccountCode</u>	TotalAmount				

Table Name : Grade

<u>StudentID</u>	<u>Semester</u>	<u>SubjectCode</u>	Credit	RegisStatus
Grade				

Table Name : SemesterGPA

<u>StudentID</u>	<u>Semester</u>	Class	<u>ProgramCode</u>	<u>MajorCode</u>
<u>GroupNum</u>	SubjectCount	SCreditRegis	SCreditEarn	SCreditCompute
SProduct	SGPA	CcreditRegis	CCreditEarn	CCreditCompute
CProduct	CGPA	LeaveSts		

Table Name : Gender

<u>Gender</u>	Description
---------------	-------------

Table Name : Nationality

NationalityCode	NationalityEname	NationalityTname	CountryName
-----------------	------------------	------------------	-------------

Table Name : PaymentType

<u>PaymentType</u>	Description
--------------------	-------------

Table Name : RegistrationType

<u>RegistrationType</u>	Description
-------------------------	-------------

Table Name : Program

<u>ProgramCode</u>	<u>MajorCode</u>	<u>GroupNum</u>	<i>FacultyCode</i>	DLevel
DegreeEName	DegreeTName	StudyYear	StudyMonth	SemOpen
SemClose	CancelSts	ISCED_Code		

Table Name : Subject

<u>SubjectCode</u>	SubjectName	SubjectTName	Credit	Lecture
Lab	Seminar	LectureHour	LabHour	SeminarHour
Week	SemesterOpen	SemBegin	SemClose	CancelSts

Table Name : GradeType

<u>Grade</u>	Description	Seq
--------------	-------------	-----

Table Name : GPARange

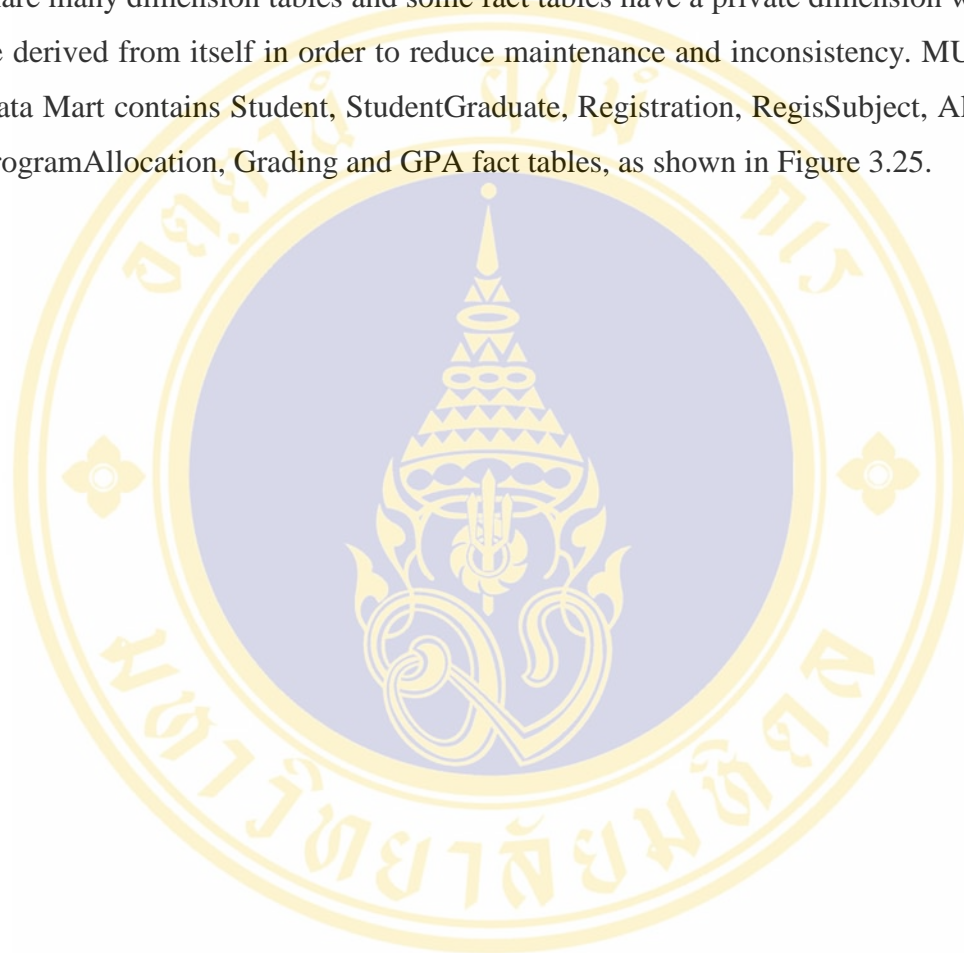
<u>GPAlevel</u>	LowerBound	UpperBound	Description
-----------------	------------	------------	-------------

Table Name : Account

<u>AccountCode</u>	AccountName	Institute	CancelSts	InstituteCode
--------------------	-------------	-----------	-----------	---------------

3.5.2 Data on MU Student Data Mart

After designing a process of MU Student Data Mart, the data model was designed as well. In order to optimize maintenance and performance, this project uses star and snowflake schema to present the data model. The multiple fact tables share many dimension tables and some fact tables have a private dimension which can be derived from itself in order to reduce maintenance and inconsistency. MU Student Data Mart contains Student, StudentGraduate, Registration, RegisSubject, Allocation, ProgramAllocation, Grading and GPA fact tables, as shown in Figure 3.25.



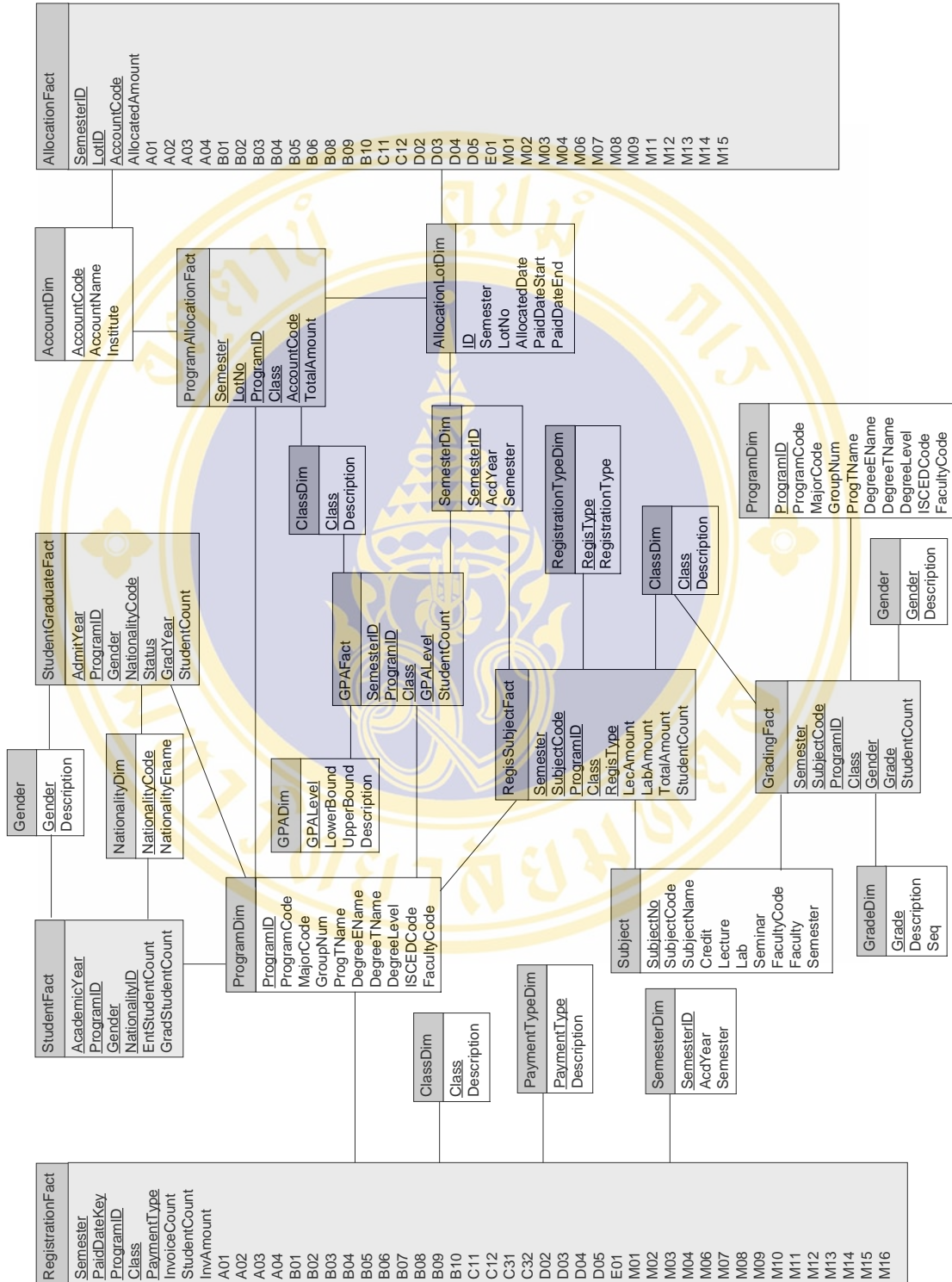


Figure 3.25 : Conceptual Dimensional Schema for Student Data Mart

Figure 3.25. shows an MU Student Data Mart data model, which can be summarized in section 3.5.2.1, and the database schema are displayed in section 3.5.2.2.

3.5.2.1 Table of MU Student Data Mart

Table 3.5 summarizes tables of an MU Student Data Mart. These tables are classified into two types of fact table and dimension table.

Table 3.5 MU Student Data Mart.

No	Name	Description	Type
1	StudentFact	Records of summarized student.	Fact
2	StudentGraduateFact	Records of summarized graduate student.	Fact
3	RegistrationFact	Records of summarized registration information.	Fact
4	RegisSubjectFact	Records of summarized enrolled subject.	Fact
5	AllocationFact	Records of registration income which is allocated to each account.	Fact
6	ProgramAllocationFact	Records of registration income which is allocated from a program to each account.	Fact
7	GradingFact	Records of summarized Grade in each semester.	Fact
8	GPAFact	Records of summarized GPA information.	Fact
9	GenderDim	Records of gender.	Dimension
10	NationalityDim	Records of nationality	Dimension
11	ProgramDim	Records of student programs	Dimension
12	DateDim	Records of date	Dimension
13	PaymentTypeDim	Records of payment types	Dimension
14	SubjectDim	Records of subjects	Dimension
15	RegistrationTypeDim	Records of registration type for each subject.	Dimension
16	GradeTypeDim	Records of grade type	Dimension
17	GPADim	Records of GPA range.	Dimension
18	AccountDim	Records of account that gets registration allocation	Dimension
19	AllocationLotDim	Records of allocation lot	Dimension
20	SemesterDim	Records of semester	Dimension

3.5.2.2 Database Schema of MU Student Data Mart

Based on Table 3.5, we can display the components of each table in various fields (the underlined attributes mean primary keys while italic attributes mean foreign keys). Detail of table structure will be given in section 3.5.3

Table Name : StudentFact

<u>AcademicYear</u>	<u>ProgramID</u>	<u>Gender</u>	<u>NationalityID</u>	EntStudentCount
GradStudentCount				

Table Name : StudentGraduateFact

AdmitYear	ProgramID	Gender	NationalityCode	Status	GradYear
StudentCount					

Table Name : RegistrationFact

<u>Semester</u>	<u>PaidDateKey</u>			<u>ProgramID</u>		<u>Class</u>		<u>PaymentType</u>	
InvoiceCount	StudentCount			InvAmount		A01	A02	A03	A04
B01	B02	B03	B04	B05	B06	B07	B08	B09	B10
C11	C12	C31	C32	D02	D03	D04	D05	E01	M01
M02	M03	M04	M06	M07	M08	M09	M10	M11	M12
M13	M14	M15	M16						

Table Name : RegisSubjectFact

<u>Semester</u>	<u>SubjectCode</u>	<u>ProgramID</u>	<u>Class</u>	<u>RegisType</u>
LecAmount	LabAmount	TotalAmount	StudentCount	

Table Name : AllocationFact

<u>Semester</u>	<u>LotID</u>		<u>AccountCode</u>			AllocatedAmount		A01	A02
A03	A04	B01	B02	B03	B04	B05	B06	B08	B09
B10	C11	C12	C31	C32	D02	D03	D04	D05	E01
M01	M02	M03	M04	M06	M07	M08	M09	M10	M11
M12	M13	M14	M15	M16					

Table Name : ProgramAllocationFact

Semester	LotNo	ProgramID	Class	AccountCode	TotalAmount
----------	-------	-----------	-------	-------------	-------------

Table Name : GradingFact

<u>Semester</u>	<u>SubjectCode</u>	<u>ProgramID</u>	<u>Class</u>	<u>Gender</u>
<u>Grade</u>	StudentCount			

Table Name : GPAFact

<u>Semester</u>	<u>ProgramID</u>	<u>Class</u>	<u>GPAlevel</u>	StudentCount
-----------------	------------------	--------------	-----------------	--------------

Table Name : Gender

<u>Gender</u>	Description
---------------	-------------

Table Name : NationalityDim

<u>NationalityID</u>	NationalityEname	NationalityTname
----------------------	------------------	------------------

Table Name : ProgramDim

<u>ProgramID</u>	ProgramCode	MajorCode	GroupNum	ProgTName
DegreeEName	DegreeTName	DegreeLevel	ICSEDCode	FacultyCode

Table Name : ClassDim

<u>Class</u>	Description
--------------	-------------

Table Name : PaymentTypeDim

<u>PaymentType</u>	Description
--------------------	-------------

Table Name : Subject

<u>SubjectNo</u>	SubjectCode	SubjectName	Credit	Lecture	Lab	Seminar
FacultyCode	Faculty	Semester				

Table Name : RegistrationTypeDim

<u>RegistrationType</u>	Description	Seq
-------------------------	-------------	-----

Table Name : GradeTypeDim

<u>Grade</u>	Description	Seq
--------------	-------------	-----

Table Name : GPADim

<u>GPAlevel</u>	LowerBound	UpperBound	Description
-----------------	------------	------------	-------------

Table Name : AccountDim

<u>AccountCode</u>	AccountName	Institute
--------------------	-------------	-----------

Table Name : AllocationLotDim

<u>LotID</u>	Semester	LotNo	AllocatedDate	PaidDateStart
PaidDateEnd				

Table Name : SemesterDim

SemesterID	AcadYear	Semester
------------	----------	----------

3.5.3 Table Structure

Based on the design of an MU Student Data Mart data model used to create data cubes for an analytical report, we can display details of data file structure that is used in this prototype as the following:

- **Table Structure of Student Data Mart**

File Name : StudentFact

Description : Records of summarized student.

File Type : Fact

No.	Field Name	Description	Type	Size	Key
1	AcademicYear	Academic year	Varchar	4	
2	ProgramID	Identification number of program	Smallint	2	
3	Gender	Gender	Varchar	1	
4	NationalityID	Identification number of nationality	Varchar	3	
5	EntStudentCount	Number of admission students	Int	4	
6	GradStudentCount	Number of graduate students	Int	4	

File Name : StudentGraduateFact

Description : Records of summarized student graduation.

File Type : Fact

No.	Field Name	Description	Type	Size	Key
1	AcademicYear	Academic year	Varchar	4	
2	ProgramID	Identification number of program	Smallint	2	
3	Gender	Gender	Varchar	1	
4	NationalityCode	Identification code of nationality	Varchar	3	
5	Status	Student status	Varchar	8	
6	GradYear	Graduate or resign year	Varchar	4	
7	StudentCount	Number of graduate students	Int	4	

File Name : RegistrationFact

Description : Records of summarized registration information.

File Type : Fact

No.	Field Name	Description	Type	Size	Key
1	Semester	Semester	Varchar	3	
2	PaidDateKey	Paid date	Smalldatetime	4	
3	ProgramID	Identification number of program	Smallint	2	
4	Class	Class of student	Varchar	1	
5	PaymentType	Payment type	Varchar	1	
6	InvoiceCount	Number of invoices	Smallint	2	
7	StudentCount	Number of students	Smallint	2	
8	InvAmount	Amount of money of invoice	Real	4	

No.	Field Name	Description	Type	Size	Key
9	A01	Amount of money of first-time admission fee	Real	4	
10	A02	Amount of money of damage deposit fee	Real	4	
11	A03	Amount of money of document fee	Real	4	
12	A04	Amount of money of student ID card fee	Real	4	
13	B01	Amount of money of university fee	Real	4	
14	B02	Amount of money of library fee	Real	4	
15	B03	Amount of money of health service fee	Real	4	
16	B04	Amount of money of student and sport activity fee	Real	4	
17	B05	Amount of money of art and recreation activity fee	Real	4	
18	B06	Amount of money of college fee	Real	4	
19	B07	Amount of money of other fee	Real	4	
20	B08	Amount of money of bank fee	Real	4	
21	B09	Amount of money of internet fee	Real	4	
22	B10	Amount of money of computing and database services	Real	4	
23	C11	Amount of money of tuition fee (lecture)	Real	4	
24	C12	Amount of money of tuition fee (laboratory)	Real	4	
25	C31	Amount of money of summer tuition fee (lecture) for summer	Real	4	
26	C32	Amount of money of summer tuition fee (laboratory)	Real	4	
27	D02	Amount of money of dormitory fee	Real	4	
28	D03	Amount of money of dormitory deposit fee	Real	4	
29	D04	Amount of money of late registration fee	Real	4	
30	D05	Amount of money of student status maintenance fee	Real	4	
31	E01	Amount of money of E01 fee	Real	4	
32	M01	Amount of money of M01 fee	Real	4	
33	M02	Amount of money of M02 fee	Real	4	
34	M03	Amount of money of M03 fee	Real	4	
35	M04	Amount of money of M04 fee	Real	4	
36	M06	Amount of money of M06 fee	Real	4	
37	M07	Amount of money of M07 fee	Real	4	

No.	Field Name	Description	Type	Size	Key
38	M08	Amount of money of M08 fee	Real	4	
39	M09	Amount of money of M09 fee	Real	4	
40	M10	Amount of money of M10 fee	Real	4	
41	M11	Amount of money of M11 fee	Real	4	
42	M12	Amount of money of M12 fee	Real	4	
43	M13	Amount of money of M13 fee	Real	4	
44	M14	Amount of money of M14 fee	Real	4	
45	M15	Amount of money of ICT fee	Real	4	
46	M16	Amount of money of M16 fee	Real	4	

File Name : RegisSubjectFact

Description : Records of summarized enrolled subject.

File Type : Fact

No.	Field Name	Description	Type	Size	Key
1	Semester	Semester	Varchar	3	
2	SubjectCode	Subject Code	Varchar	7	
3	ProgramID	Identification number of program	SmallInt	2	
4	Class	Class of student	Varchar	1	
5	RegisType	Registration type	Varchar	1	
6	LecAmount	Amount of tuition fee (lecture)	Real	4	
7	LabAmount	Amount of tuition fee (lab)	Real	4	
8	TotalAmount	Amount of tuition fee	Real	4	
9	StudentCount	Number of students	SmallInt	2	

File Name : AllocationFact

Description : Records of registration income which is allocated to each account.

File Type : Fact

No.	Field Name	Description	Type	Size	Key
1	Semester	Semester	Varchar	3	
2	LotID	Identification number of allocation lot	Smallint	2	
3	AccountCode	Account code	Varchar	50	
4	AllocatedAmount	Amount of allocated income	Real	4	
5	A01	Allocated income of first-time admission fee	Real	4	

No.	Field Name	Description	Type	Size	Key
6	A02	Allocated income of damage deposit fee	Real	4	
7	A03	Allocated income of document fee	Real	4	
8	A04	Allocated income of student ID card fee	Real	4	
9	B01	Allocated income of university fee	Real	4	
10	B02	Allocated income of library fee	Real	4	
11	B03	Allocated income of health service fee	Real	4	
12	B04	Allocated income of student and sport activity fee	Real	4	
13	B05	Allocated income of art and recreation activity fee	Real	4	
14	B06	Allocated income of college fee	Real	4	
15	B07	Allocated income of other fee	Real	4	
16	B08	Allocated income of bank fee	Real	4	
17	B09	Allocated income of internet fee	Real	4	
18	B10	Allocated income of computing and database services	Real	4	
19	C11	Allocated income of tuition fee (lecture)	Real	4	
20	C12	Allocated income of tuition fee (laboratory)	Real	4	
21	C31	Allocated income of summer tuition fee (lecture) for summer	Real	4	
22	C32	Allocated income of summer tuition fee (laboratory)	Real	4	
23	D02	Allocated income of dormitory fee	Real	4	
24	D03	Allocated income of dormitory deposit fee	Real	4	
25	D04	Allocated income of late registration fee	Real	4	
26	D05	Allocated income of student status maintenance fee	Real	4	
27	E01	Allocated income of E01 fee	Real	4	
28	M01	Allocated income of M01 fee	Real	4	
29	M02	Allocated income of M02 fee	Real	4	
30	M03	Allocated income of M03 fee	Real	4	
31	M04	Allocated income of M04 fee	Real	4	
32	M06	Allocated income of M06 fee	Real	4	
33	M07	Allocated income of M07 fee	Real	4	

No.	Field Name	Description	Type	Size	Key
34	M08	Allocated income of M08 fee	Real	4	
35	M09	Allocated income of M09 fee	Real	4	
36	M10	Allocated income of M10 fee	Real	4	
37	M11	Allocated income of M11 fee	Real	4	
38	M12	Allocated income of M12 fee	Real	4	
39	M13	Allocated income of M13 fee	Real	4	
40	M14	Allocated income of M14 fee	Real	4	
41	M15	Allocated income of ICT fee	Real	4	
42	M16	Allocated income of M16 fee	Real	4	

File Name : ProgramAllocationFact

Description : Records of registration income which is allocated from program to each account for each semester.

File Type : Fact

No.	Field Name	Description	Type	Size	Key
1	Semester	Semester	Varchar	3	
2	LotNo	Identification number of allocation lot	Tinyint	1	
3	ProgramID	Identification number of program	Smallint	2	
4	Class	Class of student	Varchar	1	
5	AccountCode	Account code	Varchar	11	
6	TotalAmount	Amount of allocated income	Money	8	

File Name : GradingFact

Description : Records of summarized Grade in each semester.

File Type : Fact

No.	Field Name	Description	Type	Size	Key
1	Semester	Semester	Varchar	3	
2	SubjectCode	Subject Code	Varchar	7	
3	ProgramID	Identification number of program	Smallint	2	
4	Class	Class of student	Varchar	1	
5	Gender	Gender	Varchar	1	
6	Grade	Grade type	Varchar	2	
7	StudentCount	Number of students	Smallint	2	

File Name : GPAFact

Description : Records of summarized GPA information.

File Type : Fact

No.	Field Name	Description	Type	Size	Key
1	Semester	Semester	Varchar	3	
2	ProgramID	Identification number of program	Smallint	2	
3	Class	Class of student	Varchar	1	
4	GPALevel	Level of GPA	Tinyint	1	
5	StudentCount	Number of students	Smallint	2	

File Name : Gender

Description : **Records of gender.**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	Gender	Gender	Varchar	1	PK
2	Description	Description	Varchar	20	PK

File Name : **NationalityDim**

Description : **Records of nationality**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	NationalityCode	Nationality Code	Varchar	3	PK
2	NationalityEname	English nationality	Varchar	30	
3	NationalityTname	Thai nationality	Varchar	30	

File Name : **ProgramDim**

Description : **Records of student programs**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	ProgramID	Program identification number	Smallint	2	PK
2	ProgramCode	Program code	Varchar	5	
3	MajorCode	Major Code	Varchar	4	
4	GroupNum	Group	Varchar	1	
5	ProgTName	Program name	Varchar	90	
6	DegreeEName	English degree name	Varchar	90	
7	DegreeTName	Thai degree name	Varchar	90	

No.	Field Name	Description	Type	Size	Key
8	DegreeLevel	Level of degree	Varchar	35	
9	ISCEDCode	ISCED Code	Varchar	3	
10	FacultyCode	Faculty Code	Varchar	2	

File Name : **ClassDim**

Description : **Records of student class**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	Class	Class of student	Varchar	1	PK
2	Description	Description of Class of student	Varchar	10	

File Name : **PaymentTypeDim**

Description : **Records of payment types**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	PaymentType	Payment code	Varchar	1	PK
2	Description	Description of payment code	Varchar	30	

File Name : **Subject**

Description : **Records of subjects**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	SubjectNo	Subject identification number	Int	8	PK
2	SubjectCode	Subject Code	Varchar	7	FK
3	SubjectName	Subject name	Varchar	50	
4	Credit	Credit	Tinyint	1	
5	Lecture	Lecture credit	Tinyint	1	
6	Lab	Lab credit	Tinyint	1	
7	Seminar	Seminar credit	Tinyint	1	
8	FacultyCode	Faculty Code	Varchar	2	
9	Faculty	Faculty Name	Varchar	55	
10	Semester	Semester	Varchar	3	

File Name : **RegistrationTypeDim**

Description : **Records of registration type for each subject.**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	RegistrationType	Registration code	Varchar	1	PK
2	Description	Description of payment code	Varchar	25	
3	Seq	Sequence of RegistrationType	Tinyint	1	

File Name : **GradeTypeDim**

Description : **Records of grade type.**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	Grade	Grade	Varchar	2	PK
2	Description	Description of grade	Varchar	25	
3	Seq	Sequence of Grade	Tinyint	1	

File Name : **GPADim**

Description : **Records of GPA range.**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	GPALevel	GPA Level	Tinyint	1	PK
2	LowerBound	Lower bound	Real	4	
3	UpperBound	Upper bound	Real	4	
4	Description	Description of GPALevel	Varchar	15	

File Name : **AccountDim**

Description : **Records of account that gets registration allocation**

File Type : **Dimension**

No.	Field Name	Description	Type	Size	Key
1	AccountCode	Account Code	Varchar	11	PK
2	AccountName	Account Name	Varchar	100	
3	Institute	Institute	Varchar	50	

File Name : AllocationLotDim

Description : Records of allocation lot

File Type : Dimension

No.	Field Name	Description	Type	Size	Key
1	LotID	Identification number	Smallint	2	PK
2	Semester	Semester	Varchar	3	
3	LotNo	Lot number	Tinyint	1	
4	AllocatedDate	Allocation date	Smalldatetime	4	
5	PaidDateStart	Paid date from	Smalldatetime	4	
6	PaidDateEnd	Paid date to	Smalldatetime	4	

File Name : SemesterDim

Description : Records of semester

File Type : Dimension

No.	Field Name	Description	Type	Size	Key
1	SemesterID	Identification number	Varchar	3	PK
2	AcdYear	Academic year	Smallint	2	
3	Semester	Semester	Varchar	6	

CHAPTER 4

IMPLEMENTATION

This chapter describes the implementation of MU Student Data Mart. It begins with a description of the system environment. Then, we show how the MU-SDM can be implemented.

4.1 System Environment

The MU-SDM has a 3-tier architecture comprising a client tier, an application logic tier and a data mart database tier. The client or presentation tier enables users to provide requests to the OLAP engine. The application logic tier comprises an OLAP engine that executes multidimensional reports from multiple end users. The database tier stores data and connects to the application logic tier. Figure 4.1 shows 3-tier architecture of MU-SDM.

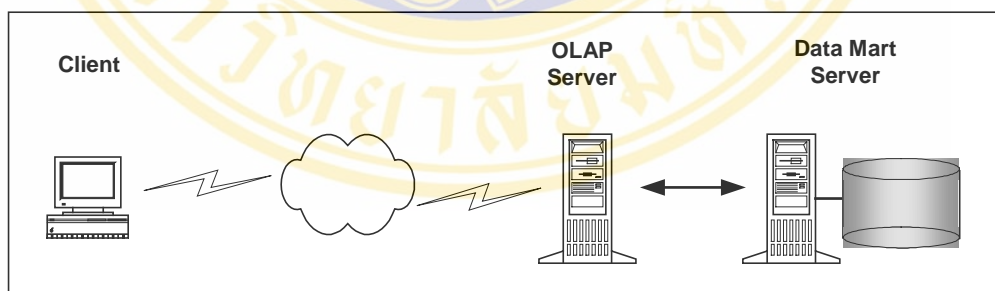


Figure 4.1 : 3-Tier Architecture of MU-SDM

Our system prototype is developed with the following environments.

Client Environment

- Desktop computer
 - Intel Pentium 4 Processor 3.00 GHz with 512 MB RAM.

- Software installed
 - Microsoft Windows XP.
 - Microsoft Internet Explorer 6.0
 - Microsoft SQL Server 2000
 - Microsoft SQL Server 2000 Analysis Services
 - Microsoft Visual Basic 6.0
 - Microsoft FrontPage 2002

Server Environment

- Desktop computer simulated as a data mart server.
 - Intel Pentium 4 Processor 3.00 GHz with 512 MB RAM.
- Software installed
 - Microsoft Windows XP
 - Microsoft SQL Server 2000
 - Microsoft SQL Server 2000 Analysis Services
 - Internet Information Services

4.2 Implementation of MU-SDM

Analyzing data with multidimensional information is useful for decision making. We implement the MU-SDM as an analytical data application in accordance with the design as described in a previous section. In this prototype, we use Microsoft SQL Server 2000 Analysis Services to create and maintain multidimensional data structure and use Microsoft Office component to develop user interface for presenting the multidimensional cube and graph. The system starts from an ETL process, which is a data acquisition process, to an information delivery process, which presents multidimensional data to users for analysis.

4.2.1 ETL Implementation

To implement ETL process, we develop three main modules of ETL described below:

- *Extraction Module* is responsible for retrieving selected data from an operational source. As described in the previous section, there are three types of extractions; namely, full extraction, defer extraction, and immediate extraction.
- *Transformation Module* is responsible for converting extracted data into a suitable format of data mart.
- *Loading Module* is responsible for updating data mart storage, which is used to create student cubes.

Extraction

For an initial phase, ETL is executed in order to extract all selected student information. The process starts with creating temporary tables, staging tables and data mart tables that are used to store all necessary data of the system. The next step is retrieving all selected data from an operational source before sending the data to a transformation and loading process.

As time advances and the student information is changed, in order to incrementally refresh the data mart, change monitoring is required. The extraction for refreshment is executed in order to extract changed data. The change monitoring detects changed data by using forty-five trigger rules in fifteen tables before applying defer extraction or immediate extraction to retrieve changed and relevant data and then sending it to a transformation and loading process.

The algorithm of defer extraction and immediate extraction are described in section 3.2.2.1. The defer extraction is proper with the complex table which change of it relate to other tables. And the immediate extraction is proper with the simple table which change of it do not affect the others. We classified the tables which implement two type of extraction as listed below:

- The table which is applied defer extraction are : CurInvoice, MInvoice, Grade, SemesterGPA, Subject.
- The table which is applied immediate extraction are : Student, AllocationSummary, ICTAllocationSummary, ProgramAllocation, Program, Account, Nationality, PaymentType, RegistrationType, GradeType.

The following is an algorithm of the extraction process written in pseudo code below:

1. If initial phase then
2. Create temporary tables.
3. Create staging tables.
4. Create data mart tables.
5. Retrieve all selected data from operational tables.
6. Insert the selected data into temporary tables.
7. Call Transformation
8. Call Loading
9. Else
10. /* Extraction for refreshment */
11. /* Change monitoring is performed by trigger */
12. If defined event occurs then
13. If valid condition then
14. If defer extraction then
15. Insert reference keys of changed data and type of change into reference table.
16. If extraction is requested then
17. Retrieve data and relevant record of reference keys
18. Call Transformation
19. End if
20. Else
21. /* immediate extraction */
22. Join changed record with relevant tables
23. Call Transformation
24. End if
25. end if
26. end if
27. end if

Transformation

This process retrieves extracted data and converts it into a suitable format of data mart. There are five tasks of transformation, but not all kinds of data have to be performed of all tasks. It depends on the type of data. The following is pseudo code showing an algorithm of the transformation process.

1. Joins a reference table with relevant tables
2. Do while not end of reference records
3. Find last change.
4. Insert last changed records into a temporary table.
5. Move next
6. Loop
7. If missing value then
8. Update with default value
9. End if
10. If dimension table need a surrogate key then
11. Find maximum number and add with one.
12. Use the number as a surrogate key of dimension.
13. End if
14. If need new column then
15. Apply an expression to derive a new column.
16. Update the records with the new value.
17. End if
18. If column needs splitting then
19. Apply an expression to split the column.
20. Update the records with the new split value.
21. End if
22. If columns need merging then
23. Apply an expression to merge the columns.
24. Update the records with the new merged value.
25. End if
26. Use an aggregation function to summarize data with grouping data.
27. Map an operational source column to a data mart column.

28. Insert summary data into staging tables.
29. Delete temporary tables and reference tables.
- 30.

Loading

Transformed data are loaded into data mart storage. In the initial phase, the dimensions and cubes have to be created from dimension tables and fact tables. The Cubes are stored in multidimensional structure that is optimized for rapid information retrieval. We use Microsoft SQL Server 200 Analysis Services to create cubes and dimensions. In order to complete a loading process, after building a cube, the cube has to be processed. There are three methods for processing a cube, such as Full Process, Incremental Update and Refresh data. Full process is performed when creating a cube, and its dimension or structure is changed. Incremental Update is performed when new data is added to a cube. Refresh data method is appropriate when the underlying data in the data mart has been changed but cube's structure is still the same. When this method is used, data in a cube is cleared and data from the source should be reloaded and recalculated.

The following pseudo code shows a loading process.

1. Retrieve a staging table.
2. Do while not end of record
3. If new record then
4. Insert new records to the data mart.
5. Else
6. Update existing records of data mart with a new value.
7. End if
8. Move next
9. Loop
10. Delete a staging table.

```
11. If initial load then
12.   /* Create a new dimension for initial load. */
13.   Select a table used as source of dimension.
14.   Define column level used for roll-up and drill-down operations.
15.   Define dimension name.
16.   If new dimension or structure is changed then
17.     Process the dimension with rebuilding the dimension structure option to
        complete creating dimension.
18.   Else
19.     Process the dimension with incremental update option.
20.   End if
21.   /* Create a new data cube for initial load. */
22.   Select a fact table from data source.
23.   Select the numeric columns that define cube measures.
24.   Select dimension as cube dimension.
25.   Add fact's column to define private dimension.
26.   Add an expression to use as calculated member.
27.   Save the cube.
28. End if
29. If a new cube or structure is changed then
30.   Process the cube with full process option.
31. else if data is change and structure is same then
32.   Process the cube with refresh data option.
33. else
34.   Process the cube with incremental update option
35. end if
```

4.2.2 Information Delivery Implementation

When a building process of data mart is completed, the cube is available for delivery to a user. We use Microsoft office components to develop user interface for presenting the multidimensional cube and chart to a user. A user can perform multidimensional operations such as roll-up, drill-drown, slice and dice, and pivoting. The process is described below:

1. Connect to data mart server.
2. Select database
3. Select data cube.
4. Define dimension columns as column and row headers of a pivot table
5. Define measures of the pivot table
6. Use the pivot table as a data source of chart which displays corresponding with the pivot table.

4.3 Cube Creation

This section shows steps of creating a cube, and we choose Program Allocation cube as an example. The step starts with an ETL process being executed in order to load data for the data mart to create a cube and process it, as described below.

1. Execute initial ETL process of MU-SDM

We select tables used as dimension tables and execute the process to extract operational data, transform and load it to a student data mart. The dimensions for Program Allocation cube are Account, Allocation lot, Class and Program. Then select Allocation by Program table and use it as a fact table, and execute the process to extract operational data, as well as transform and load it to the student data mart.

2. Create dimensions and Cube

Creating Program Allocation starts with creating dimensions such as Account, Allocation Lot, Class and Program by defining tables used as source, columns and its hierarchy levels. Then use these dimensions as dimensions of the cube, and select ProgramAllocationFact table to be used as fact of cube and set TotalAmount column as cube's measure. Furthermore, we define a calculated member by using a multidimensional expression. The detail of cube structure is described in section 4.4.

3. Execute refresh ETL process of MU-SDM

After Program Allocation cube is created and operational data is changed, the ETL process is executed again in order to load new data to the data mart. The refresh ETL is required. The step is similar to initial ETL but only changed data and its relevant are detected by change monitoring are extracted, transformed and loaded into the student data mart. In the last step, the related dimensions and Program Allocation cube are processed automatically to completely update the cube. This step involves reading the dimension tables and the fact table, calculating aggregations, and storing the results in the cube.

4.4 Cubes of MU-SDM

There are eight data cubes of MU-SDM. These cubes are composed of dimensions, measures and calculated members. Every cube schema consists of a single fact table and several dimension tables. The cube's measures are derived from columns in the fact table. The calculated members that are used as measures are derived from an expression. And the cube's dimensions are derived from columns in the dimension tables. The dimensions have hierarchical structure that is used for roll-up and drill-down operations. The structure of MU-SDM cubes is described below.

□ Student Cube

Figure 4.2 shows star schema of Student cube. The table StudentFact at the center is the fact table with the foreign keys ProgramID, Gender, and NationalityID to the corresponding dimension tables, which carry the information on the hierarchical structure of the respective dimensions.

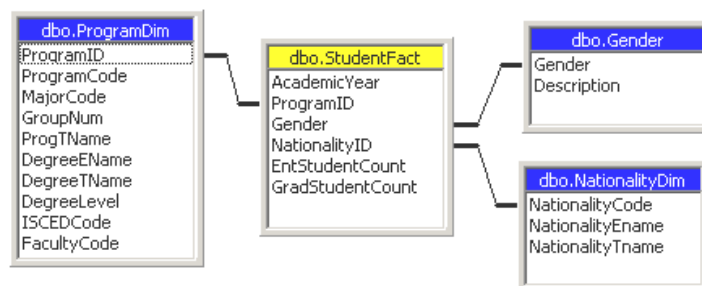


Figure 4.2 : Student Cube's Schema

A student cube has four dimensions; namely, Academic Year, Gender, Nationality and Program dimensions. The Academic Year dimension is organized in an Academic Year hierarchy. It is a private dimension which is derived from a StudentFact table. The gender dimension is organized in a Gender hierarchy. The Nationality is organized in Nationality Name hierarchy. And a Program dimension is organized in a Faculty Code and Program hierarchy. Table 4.1 shows dimension hierarchy.

Table 4.1 : Dimension Hierarchy of Student Cube.

Dimension	Level
Academic Year	▪ Academic Year
Gender	▪ Gender
Nationality	▪ Nationality Name
Program	▪ Faculty Code ▪▪ Program

The Student cube's measures show the number of admitted and graduated students. The Student cube structure is shown as tree structure in Figure 4.3.

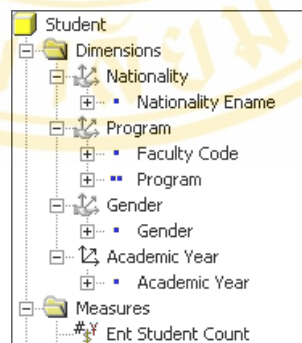


Figure 4.3 : Student Cube Structure

□ Student Graduation Cube

Figure 4.4 shows star schema of a Student Graduation cube. The table StudentGraduateFact at the center is the fact table with the foreign keys ProgramID,

Gender, and NationalityID to the corresponding dimension tables, which carry the information on the hierarchical structure of the respective dimensions.

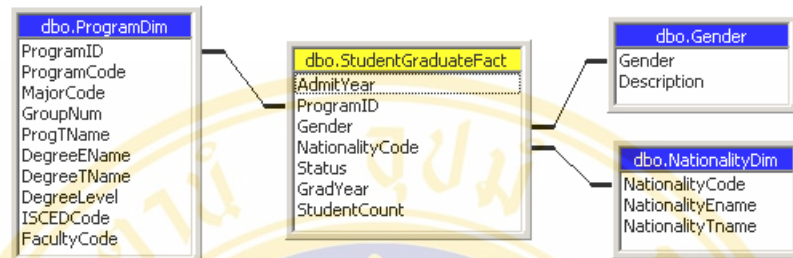


Figure 4.4 : Student Graduation Cube's Schema

A student Graduation cube has five dimensions; namely, Admit Year, Program, Gender, Nationality and Status dimensions. The Academic Year and Status dimension are private dimensions which are derived from the StudentGraduateFact table. The former is organized in an Academic Year hierarchy and the latter is organized in a Status and Grad Year hierarchy. The gender dimension is organized in a Gender hierarchy. The Nationality is organized in Nationality Name hierarchy. And a Program dimension is organized in a Faculty Code and Program hierarchy. Table 4.2 shows dimension hierarchy.

Table 4.2 : Dimension Hierarchy of Student Graduation Cube.

Dimension	Level
Admit Year	▪ Admit Year
Gender	▪ Gender
Nationality	▪ Nationality Name
Program	▪ Faculty Code ▪▪ Program
Status	▪ Status ▪▪ Grad Year

The Student Graduation cube's measures show the number of graduated and resign students, and calculated members that are formulated from measure and

dimensions show the percentage of students. The formula of calculated members is shown as follows:

$$\text{Amount}(\%) = \frac{[\text{Measures}].[\text{Student Count}]}{([\text{Measures}].[\text{Student Count}], \text{Ancestor}([\text{Gender}].\text{CurrentMember}, [\text{Gender}].[(\text{All})]), \text{Ancestor}([\text{Nationality}].\text{CurrentMember}, [\text{Nationality}].[(\text{All})]), \text{Ancestor}([\text{Program}].\text{CurrentMember}, [\text{Program}].[(\text{All})]), \text{Ancestor}([\text{Status}].\text{CurrentMember}, [\text{Status}].[(\text{All})])])}$$

The Student Graduation cube structure is shown as tree structure in Figure 4.5.

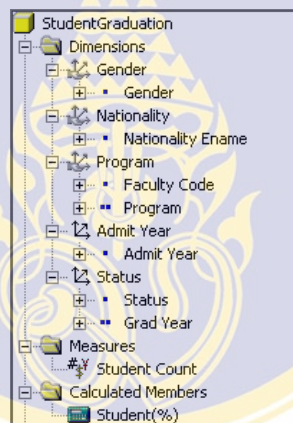


Figure 4.5 : Student Graduation Cube Structure.

□ Registration Cube

Figure 4.6 shows star schema of a Registration cube. The table RegistrationFact at the center is the fact table with the foreign keys Semester, PaidDateKey, ProgramID, Class, and PaymentType to the corresponding dimension tables, which carry the information on the hierarchical structure of the respective dimensions.

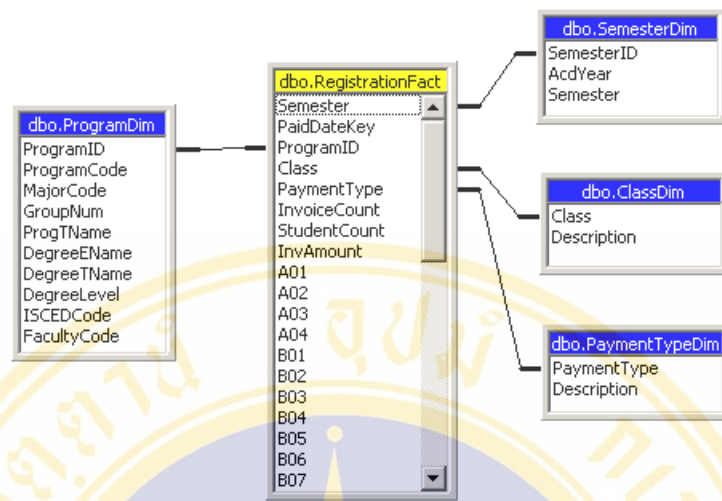


Figure 4.6 : Registration Cube’s Schema

A Registration cube has five dimensions; namely, Semester, Payment Date, Program, Class and Payment Type dimensions. The semester dimension is organized in AcdYear and Semester hierarchy. The Payment Date dimension is a private dimension which is derived from a RegistrationFact table. This dimension is organized in a Year, Month and Day hierarchy. The program dimension is organized in Faculty Code and Program hierarchy. The class dimension is organized in class hierarchy. And the payment type dimension is organized in payment type hierarchy. Table 4.3 shows dimension hierarchy.

Table 4.3 : Dimension Hierarchy of Registration Cube.

Dimension	Level
Semester	▪ Acd Year
	▪▪ Semester
Payment Date	▪ Year
	▪▪ Month
	▪▪▪ Day
Program	▪ Faculty Code
	▪▪ Program
Class	▪ Class
Payment Type	▪ Payment Type

The Registration cube’s measures show the number of Invoices and students, as well as the amount of invoice and registration fees. The Registration cube structure is shown as tree structure in Figure 4.7.

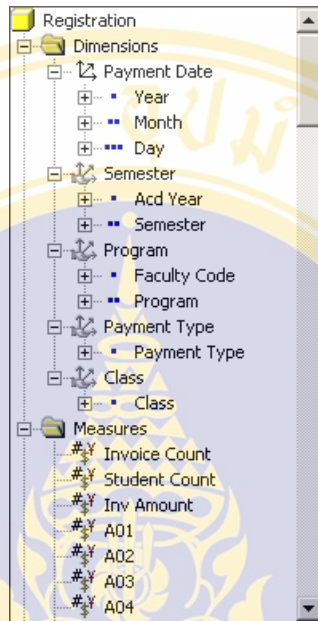


Figure 4.7 : Registration Cube Structure.

□ Registration Subject Cube

Figure 4.8 shows star schema of a Registration Subject cube. The table RegisSubjectFact at the center is the fact table with the foreign keys Semester, SubjectCode, ProgramID and RegisType to the corresponding dimension tables, which carry the information on the hierarchical structure of the respective dimensions.

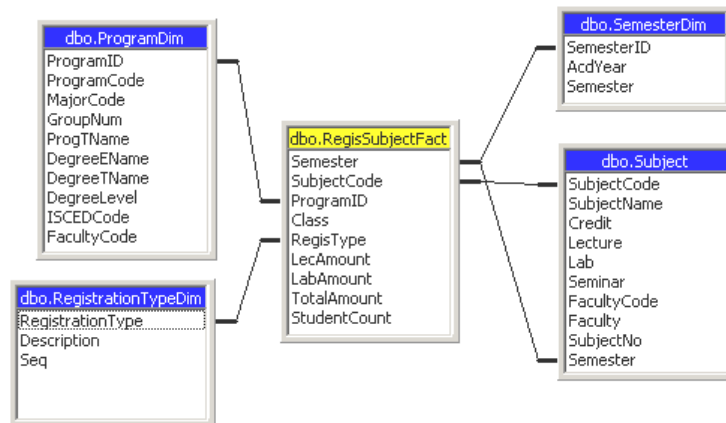


Figure 4.8 : Registration Subject Cube’s Schema

A Registration Subject cube has four dimensions; namely, Semester, Subject, Program and Registration Type dimensions. The semester is organized in Acd Year and Semester hierarchy. The Subject dimension is organized in a Faculty, Department and Subject hierarchy. The Program dimension is organized in Faculty Code and Program hierarchy. And the Registration Type dimension is organized in Registration type hierarchy. Table 4.4 shows dimension hierarchy.

Table 4.4 : Dimension Hierarchy of Registration Subject Cube.

Dimension	Level
Semester	<ul style="list-style-type: none"> ▪ Acd Year ▪▪ Semester
Subject	<ul style="list-style-type: none"> ▪ Faculty ▪▪ Department ▪▪▪ Subject
Program	<ul style="list-style-type: none"> ▪ Faculty Code ▪▪ Program
Registration Type	<ul style="list-style-type: none"> ▪ Registration Type

The Registration Subject cube’s measures show the number of students and amount of tuition fee. The Registration Subject cube structure is shown as tree structure in Figure 4.9.

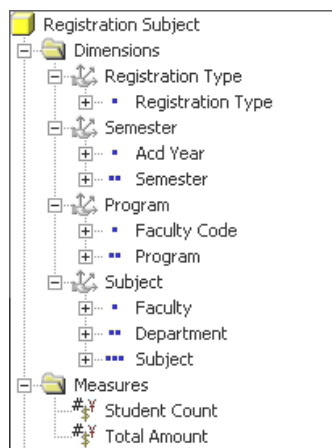


Figure 4.9 : Registration Subject Cube Structure.

❑ Allocation Cube

Figure 4.10 shows snowflake schema of an Allocation cube. The table AllocationFact at the center is the fact table with the foreign keys Semester, LotNo, and AccountCode to the corresponding dimension tables, which carry the information on the hierarchical structure of the respective dimensions.

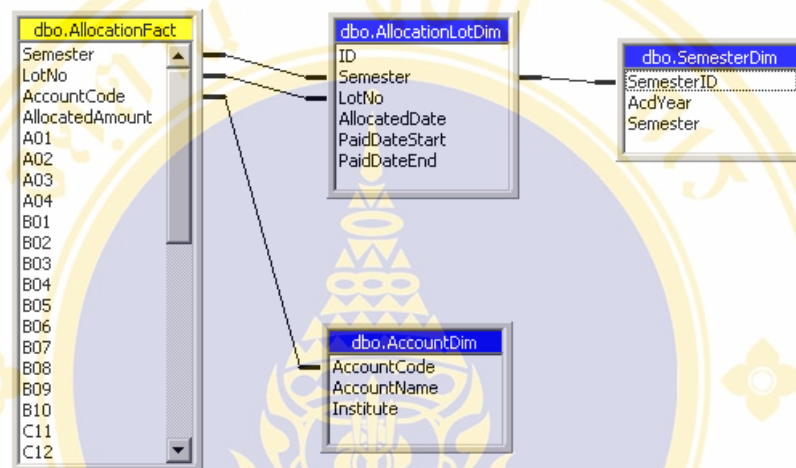


Figure 4.10 : Allocation Cube’s Schema

An Allocation cube has two dimensions; namely, Account and Allocation Lot dimensions. The Account dimension is organized in Institute and Account Name hierarchy. And the Allocation Lot dimension is organized in Acd Year, Semester and Lot No hierarchy. Table 4.5 shows dimension hierarchy.

Table 4.5 : Dimension Hierarchy of Registration Cube.

Dimension	Level
Account	▪ Institute
	▪▪ Account Name
Allocation Lot	▪ Acd Year
	▪▪ Semester
	▪▪▪ Lot No

The Allocation cube’s measures show the amount of allocated income and fees. Calculated members are formulated from measures and dimensions show the percentage of allocated income amount. The formula of calculated members is shown as follows:

$$\text{Amount(\%)} = \frac{[\text{Measures}].[\text{Amount}]}{([\text{Measures}].[\text{Amount}], \text{Ancestor}([\text{Account}].\text{CurrentMember}, [\text{Account}].[(\text{All})])])}$$

The Allocation cube structure is shown as tree structure in Figure 4.11.

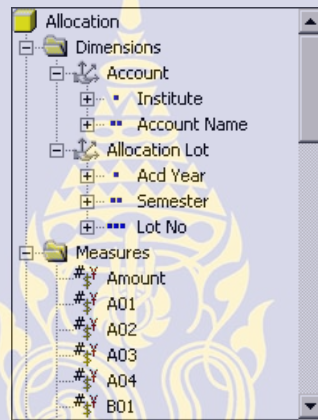


Figure 4.11 : Allocation Cube Structure.

□ Program Allocation Cube

Figure 4.12 shows schema of a Program Allocation cube. The table ProgramAllocationFact at the center is the fact table with the foreign keys Semester, LotNo, ProgramID, Class and AccountCode to the corresponding dimension tables, which carry the information on the hierarchical structure of the respective dimensions.

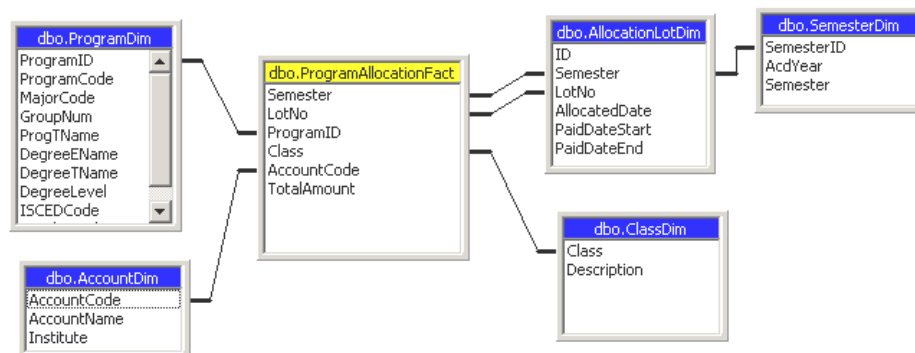


Figure 4.12 : Program Allocation Cube’s Schema

The Program Allocation cube has four dimensions; namely, Allocation Lot, Program, Class and Account dimensions. The Allocation Lot dimension with snowflake schema is organized in Acd Year, Semester and Lot No hierarchy. The Program dimension is organized in Faculty Code and Program hierarchy. The Class dimension is organized in Class hierarchy. And the Account dimension is organized in Institute and Account Name hierarchy. Table 4.6 shows dimension hierarchy.

Table 4.6 : Dimension Hierarchy of Program Allocation Cube.

Dimension	Level
Allocation Lot	<ul style="list-style-type: none"> ▪ Acd Year ▪▪ Semester ▪▪ Lot No
Program	<ul style="list-style-type: none"> ▪ Faculty Code ▪▪ Program
Class	<ul style="list-style-type: none"> ▪ Class
Account	<ul style="list-style-type: none"> ▪ Institute ▪▪ Account Name

The Program Allocation cube's measures show the amount of allocated income and calculated members that are formulated from measure and dimensions show the percentage of allocated income and income growth. The formula of calculated members is shown as follows:

$$\text{Amount(\%)} = \frac{[\text{Measures}].[Total Amount]}{([\text{Measures}].[Total Amount], \text{Ancestor}([\text{Account}].CurrentMember, [\text{Account}].[All]), \text{Ancestor}([\text{Program}].CurrentMember, [\text{Program}].[All]), \text{Ancestor}([\text{Class}].CurrentMember, [\text{Class}].[All])})}$$

$$\text{Income Growth} = [\text{Measures}].[Total Amount] - ([\text{Measures}].[Total Amount], [\text{Allocation Lot}].PrevMember)$$

The Program Allocation cube structure is shown as tree structure in Figure 4.13.

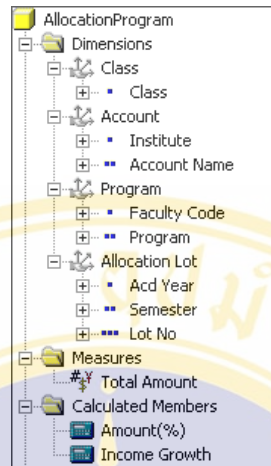


Figure 4.13 : Program Allocation Cube Structure.

□ Grading Cube

Figure 4.14 shows star schema of Grading cube. The table GradingFact at the center is the fact table with the foreign keys Semester, SubjectCode, ProgramID, Class and Grade to the corresponding dimension tables, which carry the information on the hierarchical structure of the respective dimensions.

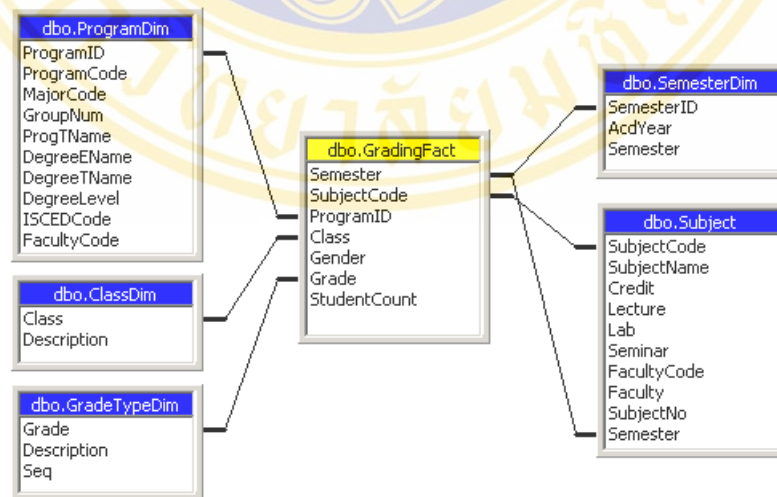


Figure 4.14 : Grading Cube’s Schema

The Grading cube has five dimensions; namely, Semester, Subject, Program, Class and Grade dimensions. The semester is organized in Acd Year and

Semester hierarchy. The Subject dimension is organized in a Faculty, Department and Subject hierarchy. The Program dimension is organized in Faculty Code and Program hierarchy. The Class dimension is organized in Class hierarchy. And the Grade dimension is organized in Grade hierarchy. Table 4.7 shows dimension hierarchy.

Table 4.7 : Dimension Hierarchy of Grading Cube.

Dimension	Level
Semester	<ul style="list-style-type: none"> ▪ Acd Year ▪▪ Semester
Subject	<ul style="list-style-type: none"> ▪ Faculty ▪▪▪ Department ▪▪ Subject
Program	<ul style="list-style-type: none"> ▪▪ Faculty Code ▪▪ Program
Class	<ul style="list-style-type: none"> ▪ Class
Grade	<ul style="list-style-type: none"> ▪ Grade

The Grading cube's measures show the number of students and calculated members that are formulated from measure and dimensions show the percentage of students. The formula of calculated member is shown as follows:

$$\text{Student}(\%) = \frac{[\text{Measures}].[Student\ Count]}{([\text{Measures}].[Student\ Count], \text{Ancestor}([\text{Program}].CurrentMember, [\text{Program}].[All]), \text{Ancestor}([\text{Class}].CurrentMember, [\text{Class}].[All]), \text{Ancestor}([\text{Grade}].CurrentMember, [\text{Grade}].[All]))}$$

The Grading cube structure is shown as tree structure in Figure 4.15.

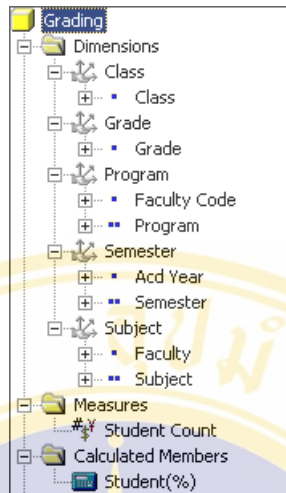


Figure 4.15 : Grading Cube Structure.

□ GPA Cube

Figure 4.16 shows star schema of a GPA cube. The table GPAFact at the center is the fact table with the foreign keys Semester, ProgramID, Class and GPALevel to the corresponding dimension tables, which carry the information on the hierarchical structure of the respective dimensions.

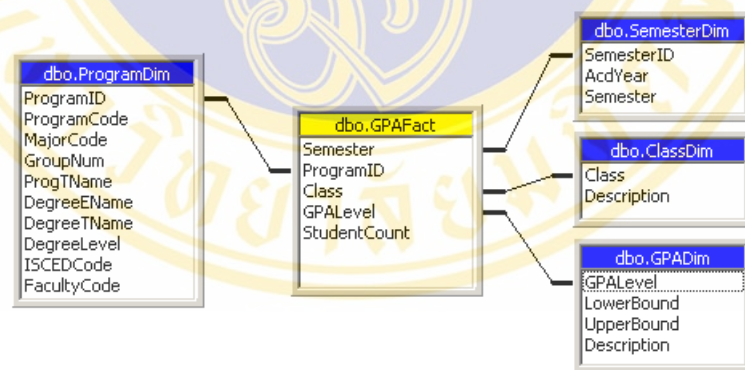


Figure 4.16 : GPA Cube's Schema

The GPA cube has four dimensions; namely, Semester, Program, Class and GPALevel dimensions. The semester is organized in Acd Year and Semester hierarchy. The Program dimension is organized in Faculty Code and Program hierarchy. The Class dimension is organized in Class hierarchy. And the GPALevel dimension is organized in GPALevel hierarchy. Table 4.8 shows dimension hierarchy.

Table 4.8 : Dimension Hierarchy of GPA Cube.

Dimension	Level
Semester	<ul style="list-style-type: none"> ▪ Acd Year ▪▪ Semester
Program	<ul style="list-style-type: none"> ▪ Faculty Code ▪▪ Program
Class	<ul style="list-style-type: none"> ▪ Class
GPAlevel	<ul style="list-style-type: none"> ▪ GPAlevel

The GPA cube’s measures show the number of students and calculated members that are formulated from measure and dimensions show the percentage of students. The formula of calculated member is shown as follows:

$$\text{Student(\%)} = \frac{[\text{Measures}].[\text{Student Count}]}{([\text{Measures}].[\text{Student Count}], \text{Ancestor}([\text{Program}].\text{CurrentMember}, [\text{Program}].[(\text{All})]), \text{Ancestor}([\text{GPAlevel}].\text{CurrentMember}, [\text{GPAlevel}].[(\text{All})]), \text{Ancestor}([\text{Class}].\text{CurrentMember}, [\text{Class}].[(\text{All})])])}$$

The GPA cube structure is shown as tree structure in Figure 4.17.

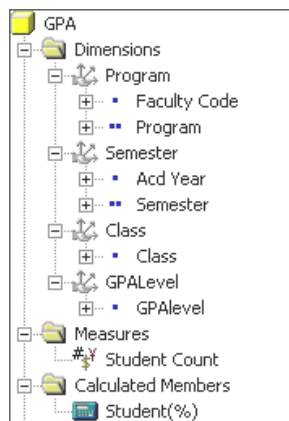


Figure 4.17 : GPA Cube Structure.

CHAPTER 5

SYSTEM TESTING AND RESULTS

This chapter shows testing of the MU-SDM prototype. In the first section, we describe main function of MU-SDM. The following part is the detail of testing derived from testing scenarios.

5.1 Two Main Functions of MU-SDM Prototype

As proposed in chapter 4, MU-SDM has two major functions: namely, ETL and information delivery. In this section, we explain them in detail, and demonstrate the functionality of each process.

5.1.1 ETL

In this part, we show how to prepare information for the data mart. This process is operated by a data mart administrator. The main screen of ETL process is shown in Figure 5.1. There are three processes: preparation, initial ETL and refreshment of ETL.

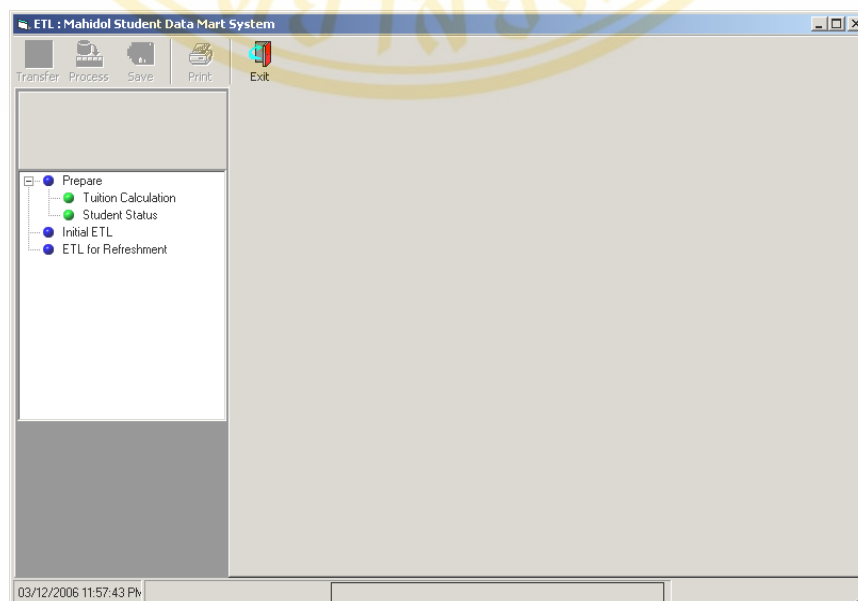


Figure 5.1 : Main Screen of ETL Process.

5.1.1.1 Preparation

The first process of the system is to prepare some data that can not be completed from an operational system but it is important for MU-SDM. There are two sub-processes listed below.

□ Tuition Calculation

Tuition for each subject in a semester provides information that a university executive knows about income from an operation of a faculty. This important information is needed for MU-SDM but an operational system provides tuition information as a flag that has to use some rules to interpret into an exact amount. Applying the rules when extracting this information into a data mart is time-consuming and there are some exceptional rules that cause incorrect calculation. Moreover, modifying the operational system to provide tuition amount when a student enrolls a subject affects a related system. Tuition calculation is shown in Figure 5.2.

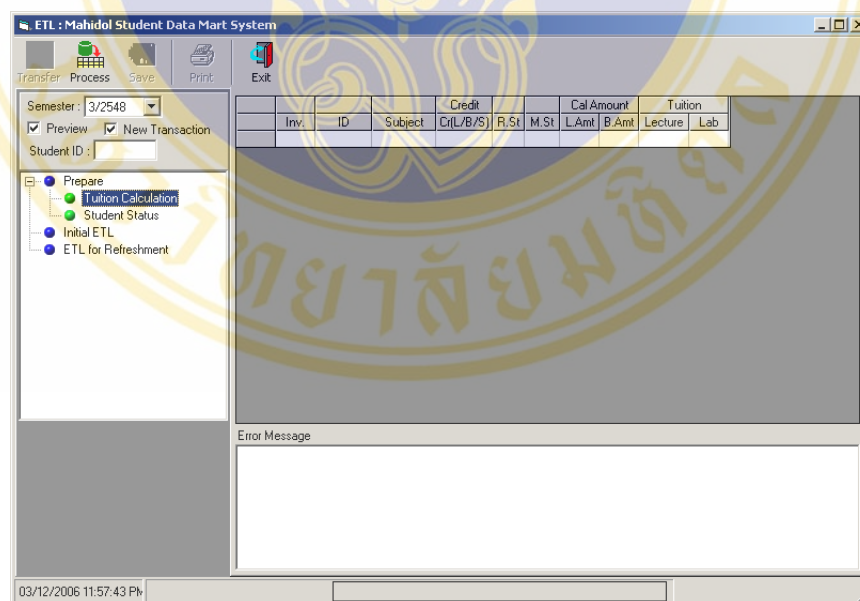


Figure 5.2 : Tuition Calculation

Figure 5.2 shows a tuition calculation process. Users can define criteria for this process as listed below:

- Semester : to define semester that wants to process.
- Preview check-box : to view tuition before updating or unchecking in order to update automatically
- New Transaction check-box : check on the check-box to calculate only a transaction that has not been calculated for tuition.
- Student ID : to define an enrolled transaction of a student that wants to process.

After defining criteria and running the tuition calculation process, the results will be shown in Figure 5.3

Invt	ID	Subject	Credit	R St	M St	Cal Amount		Tuition	
			Cr(L/B/S)			L.Amt	B.Amt	Lecture	Lab
2713	4771194	PHST488	3(0/3/0)	N	N	0	3,600		
					Total	600	4,800		
2714	4771195	PHID487	2(1/1/0)	N	N	600	1,200		
2714	4771195	PHST488	3(0/3/0)	N	N	0	3,600		
					Total	600	4,800		
2715	4771196	PHID487	2(1/1/0)	N	N	600	1,200		
2715	4771196	PHST488	3(0/3/0)	N	N	0	3,600		
					Total	600	4,800		
3002	4771197	PHID487	2(1/1/0)	N	N	600	1,200		
3002	4771197	PHST488	3(0/3/0)	N	N	0	3,600		
					Total	600	4,800		
3001	4771198	PHID487	2(1/1/0)	N	N	600	1,200		
3001	4771198	PHST488	3(0/3/0)	N	N	0	3,600		
					Total	600	4,800		
2716	4771199	PHID487	2(1/1/0)	N	N	600	1,200		
2716	4771199	PHST488	3(0/3/0)	N	N	0	3,600		
					Total	600	4,800	600	4,800

Figure 5.3 : Result of Tuition Calculation Process.

□ Student Status

Student status consists of many types of data, such as program and class of students, which are changed every semester. Normally, these data are generated by an operational system. Moreover, when the system needs some data, it will be generated by MU-SDM directly. The MU-SDM requires student status data to serve the process of fact table generation.

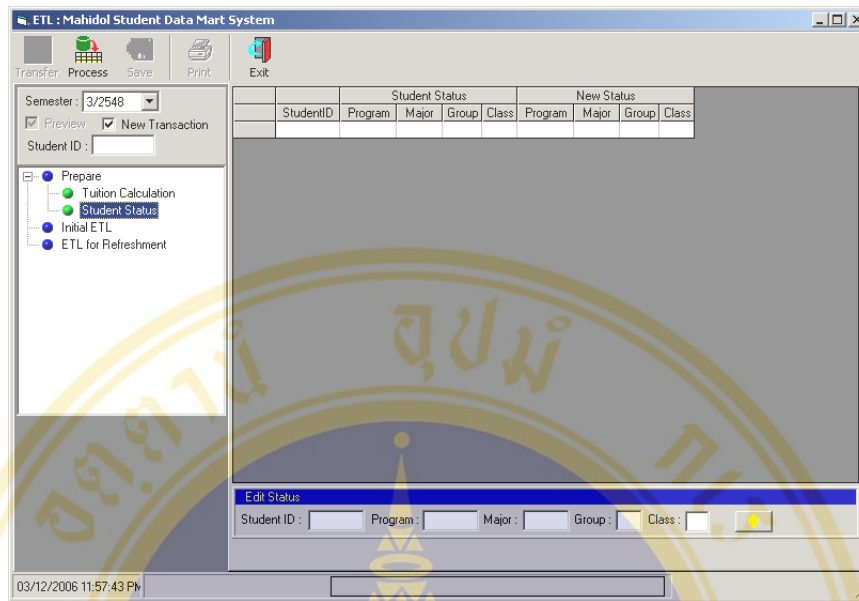


Figure 5.4 : Student Status.

Figure 5.4 shows a screen of Student Status process. A user can define criteria for this processing the same way as the Tuition Calculation process. After defining the criteria and starting the Student Status process, the result will be shown in Figure 5.5.

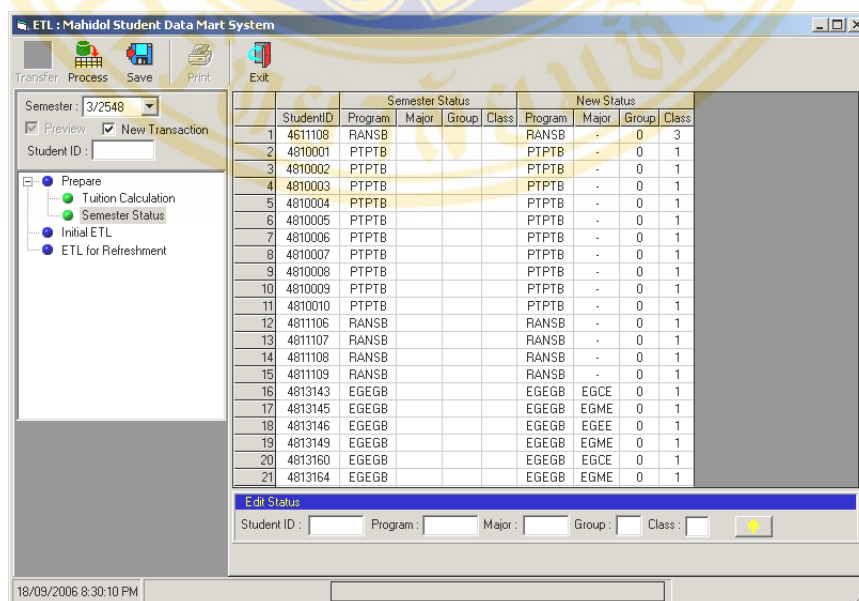



Figure 5.5 : Student Status for Semester 3/2548.

Users can change student status information by double-clicking student row, and student status will appear at the bottom of the screen as shown in Figure 5.6. Users can enter new status and click  button to change student status in grid. And click save button to update student status. After finishing the process, the system will be shown in Figure 5.7.

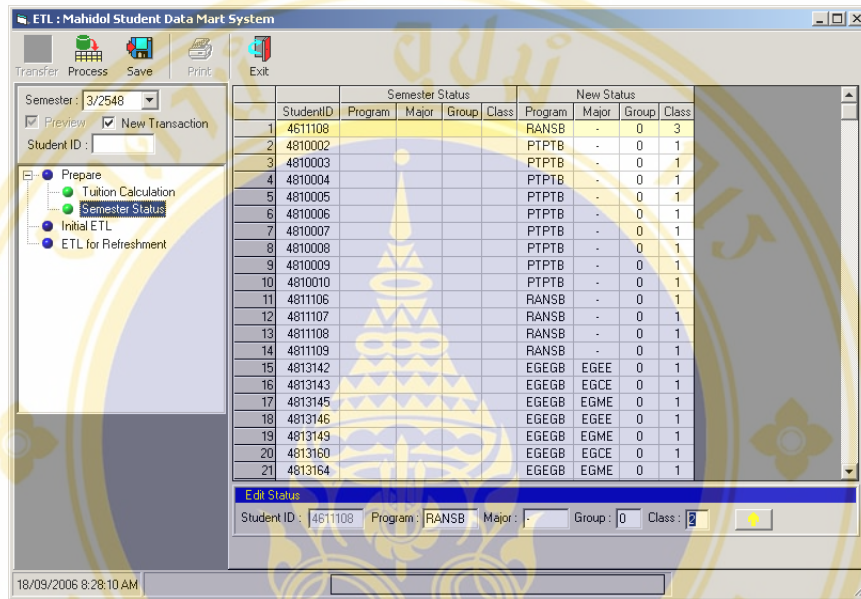


Figure 5.6 : Edit Student Status.

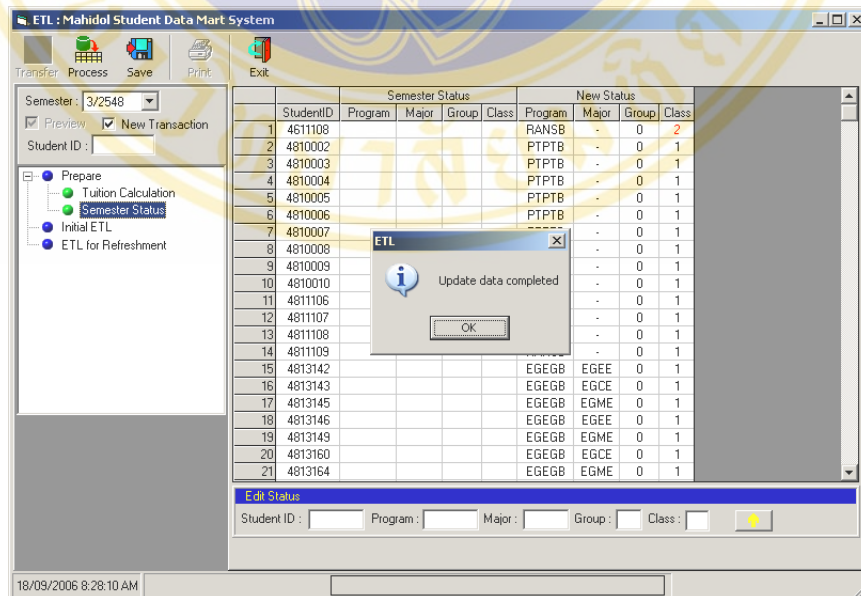


Figure 5.7 : Update Student Status.

5.1.1.2 Initial ETL

This process is a data acquisition process, which is responsible for extracting operational data, transforming it into an appropriate format of the student data mart, and loading data into source data of MU-SDM data cube. This process is useful for a data mart administrator, who does not have to perform several complex processes of data acquisition before building a data cube.

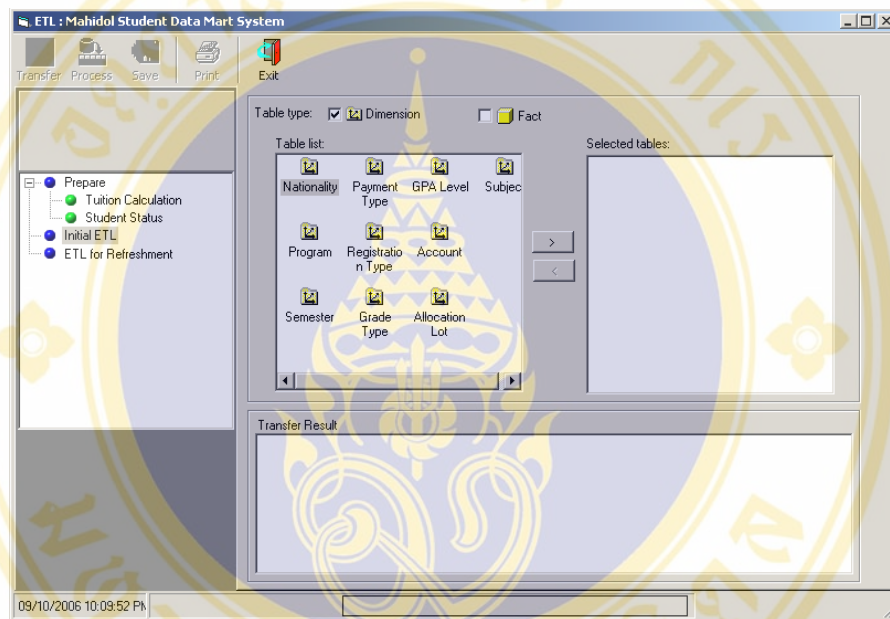


Figure 5.8 : Initial ETL

Figure 5.8 shows Initial ETL. There are two types of tables, namely: dimension and fact tables. Users can define the type of table that wants to process by checking on table type check-box. The table icon will be shown in a table list, and they may uncheck the table type to remove the table from the table list. Select a table in the table list and click right arrow to move it to selected tables. Click a process button to execute an initial ETL process. The system will show the status of the process in transfer result box, as shown in Figure 5.9.

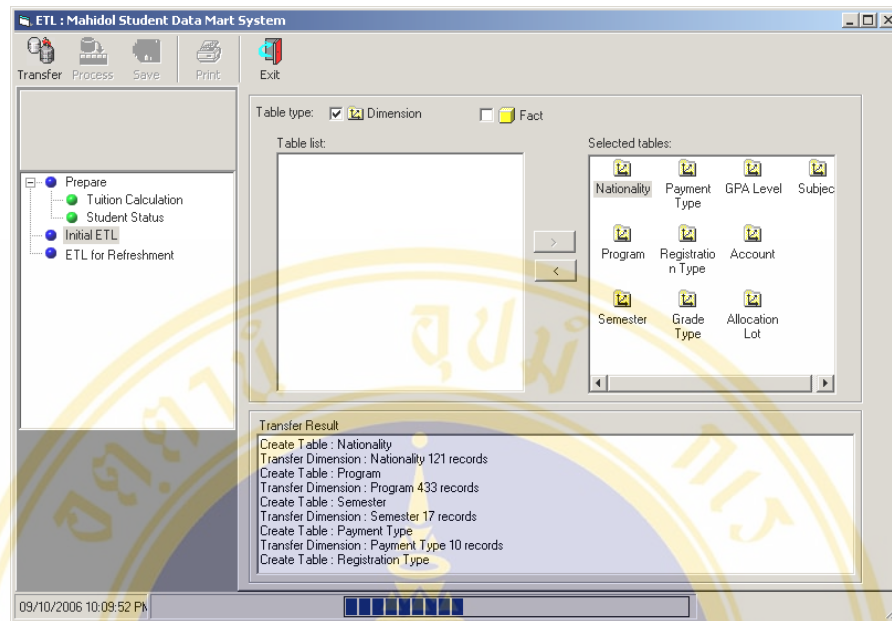


Figure 5.9 : Executing Initial ETL

5.1.1.3 ETL for Refreshment

After initial ETL and time advances, operational data that is integrated into the data mart is changed, and so the data mart content must be consistent with the underlying sources. Data mart refreshment must be performed. Changed data and its related data from the operational system are extracted, transformed and loaded into the source of data cube. The last step is processing dimensions and cubes in order to completely refresh the student data mart.

We use a trigger technique to detect changed data in a student information system in order to perform refreshment of ETL process. Triggers are executed when data modification (Insert, Update and Delete) on specified table on which the trigger is defined occurs. We classify three types of triggers on a table, namely: Insert, Update and Delete triggers. These triggers perform action of inserting records into a specified table which is used for an extraction and transformation process. All interesting changed data of the student information system can be captured.

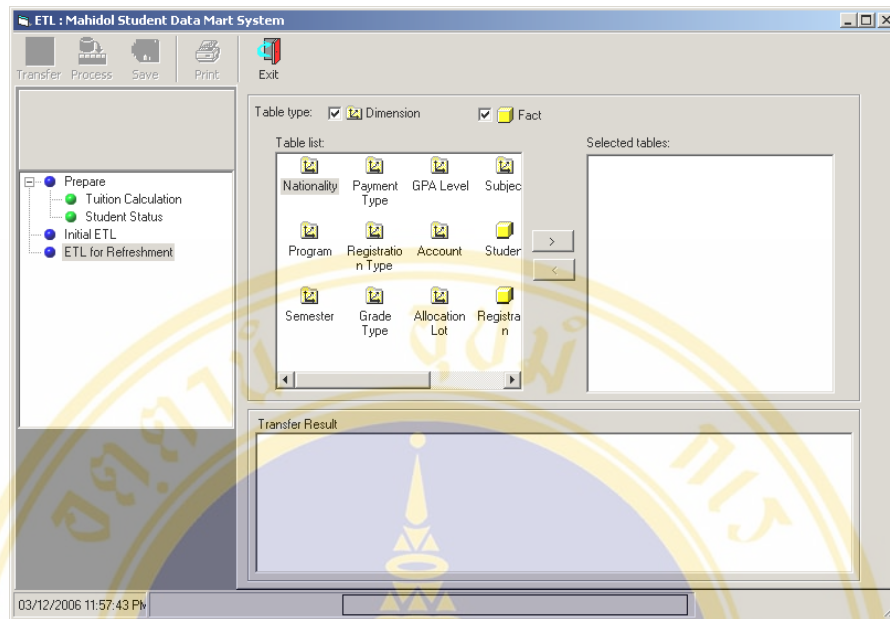


Figure 5.10 : ETL for Refreshment

Figure 5.10 shows Refreshment of ETL. The type of table and a table selecting method are performed as Initial ETL after selecting table and clicking a process button, as shown in Figure 5.11.

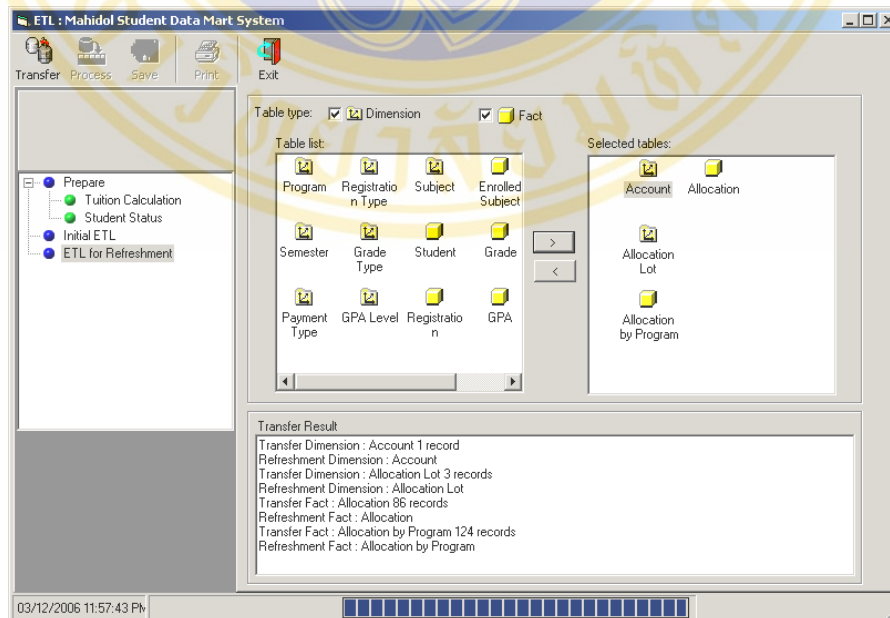


Figure 5.11 : Executing Refreshment of ETL

5.1.2 Information Delivery

Information delivery is a process that provides multidimensional analysis to users. Multidimensional analysis is a data analysis process that provides a flexible and easy-to-use function and useful information for a decision maker. MU-SDM provides multidimensional analysis that the decision makers can define any type of grouping data to analyze the organization's operation. They can easily use drag and drop operation to rotate, roll-up, drill-down the dimension and define filter of dimension that they would like to see. MU-SDM provides two styles of view that are useful for decision makers; namely, a pivot table style (spreadsheet-like) and a graphic style.

5.1.2.1 Multidimensional Analysis Component

MU-SDM represents multidimensional information with a pivot table, as shown in Figure 5.12, multidimensional view of student information. The pivot table is composed of five areas listed below.

- ❑ Filter Area

This area provides a drop-down interface to filter the desired data. Users can define multi-fields and multi-values as needed.

- ❑ Column Area

Area for displaying dimension fields in a column.

- ❑ Row Area

Area for displaying dimension fields in a row.

- ❑ Data/Measure Area

Area for displaying measures in a grid table.

- ❑ Table Field List

A list of all dimensions and measures of a data cube is shown in this area. The Table Field List is composed of two types of fields: namely, Measure and Dimension. These Dimensions are displayed as hierarchy.

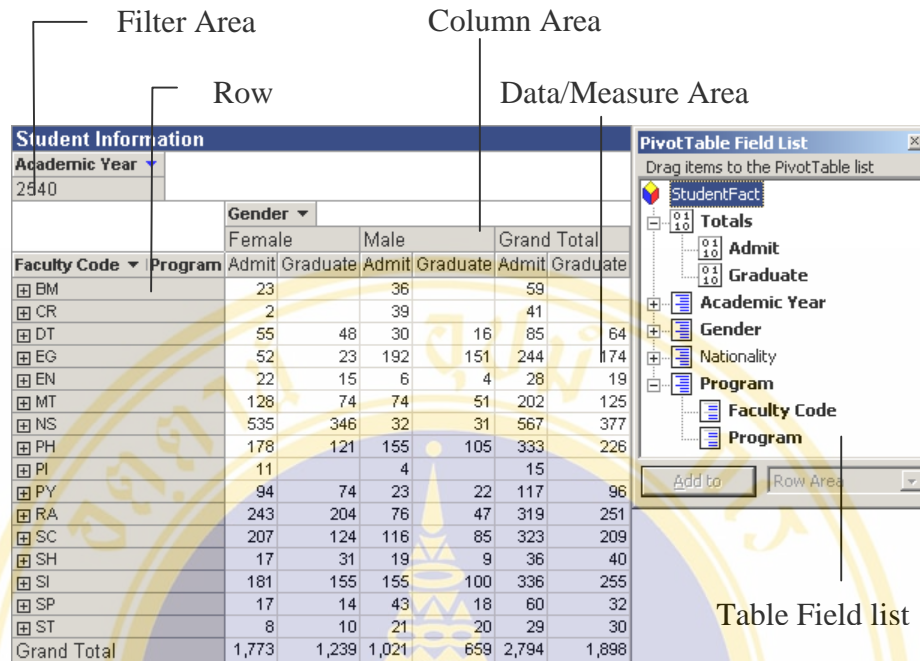


Figure 5.12 : Pivot Table of Student Information.

Additionally, the MU-SDM provides graphic information to users. The graph displays as a dynamic graph that changes according to the pivot table, as shown in Figure 5.13, student information graph with series in row and Figure 5.14, student information graph with series in column.

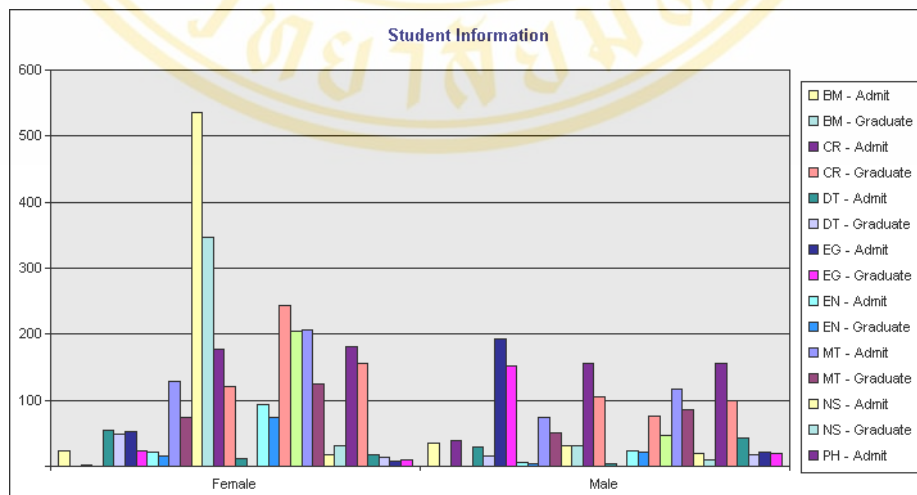


Figure 5.13 : Student Information Graph with Series in Row.

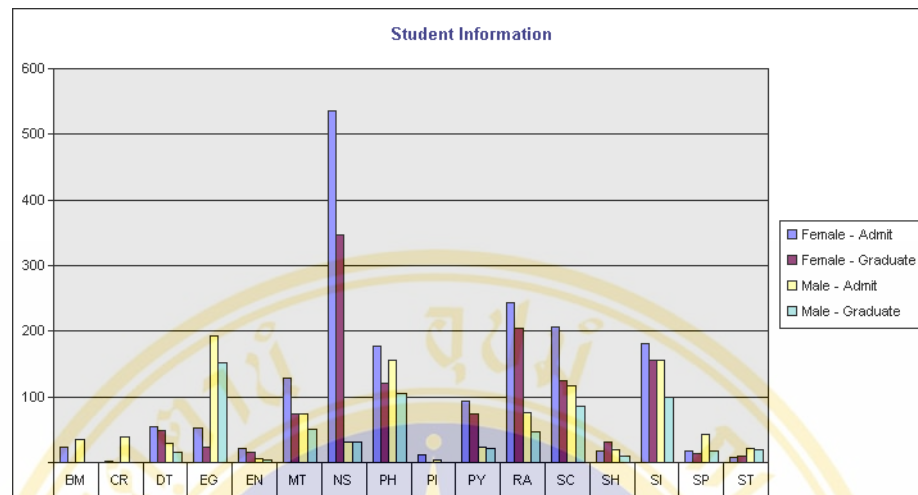


Figure 5.14 : Student Information Graph with Series in Column

5.1.2.2 Multidimensional Analysis Function

MU-SDM provides basic functions to implement multidimensional analysis, as listed below.

- Add field.

Users can add a needed dimension field or measure to any area of the pivot table by dragging field from Table Field List and dropping it in any area. Another way to add field into any area of the Pivot Table is by selecting the field in Table Field List, defining the area from a drop-down button at the bottom of Table Field List, and clicking Add to button.

- Remove field.

An easy way to remove unneeded field is by selecting field heading and dragging it outside the pivot table area. When left mouse is released, the field will be removed. Another way is right-clicking in field heading of the Pivot Table, and the context menu will appear, and then select Remove Field from the context menu

- Filter

Users can define filter by dragging a needed dimension from the Table Field List to the Filter Area. For example, they may define filter on Academic Year dimension, click drop-down button of dimension and check all

dimension values or a needed dimension value that they want to see. The pivot table will be changed according to the filter, as shown in Figure 5.15.

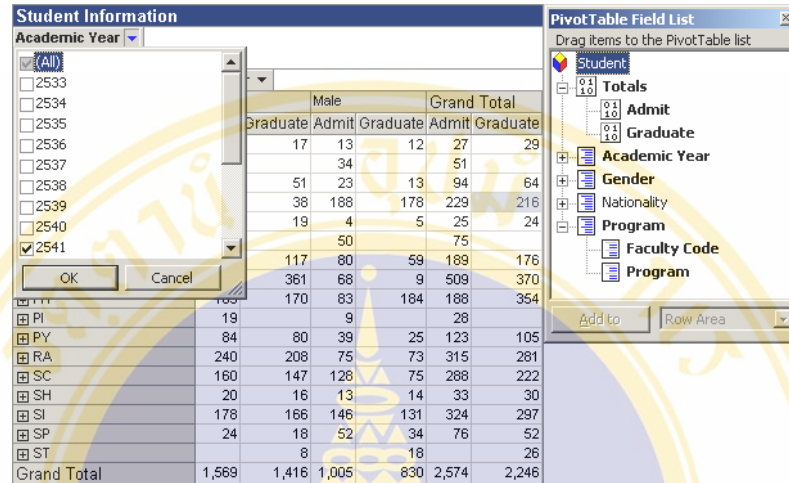


Figure 5.15 : Apply Filter to Student Information.

Additionally, users can specify the filter of the dimension on Row or Column Area by checking a needed dimension value. For example, a GPA cube with apply filter on column, as shown in Figure 5.16.

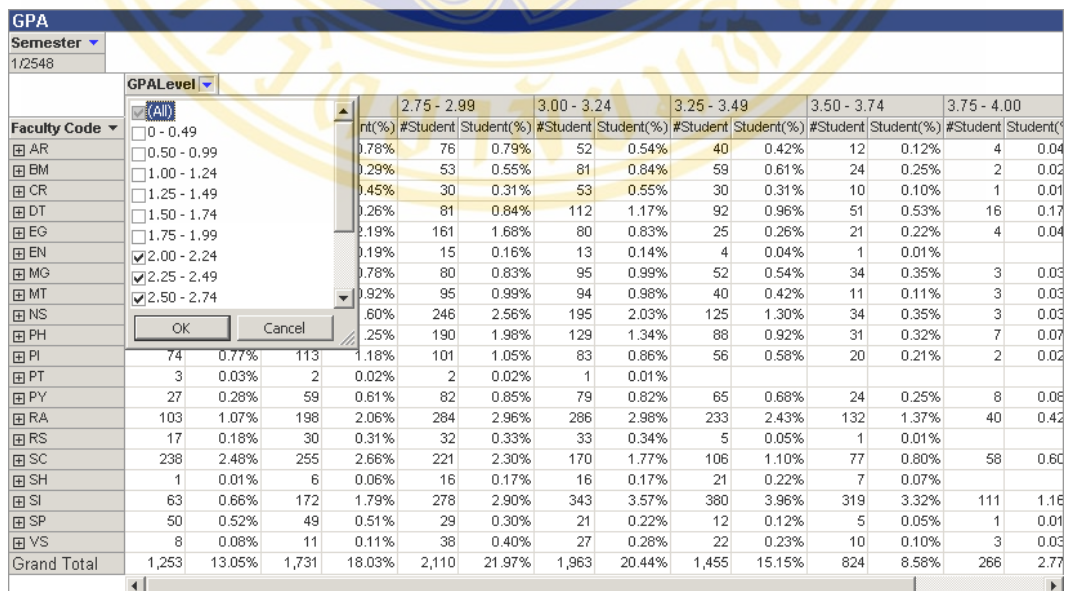


Figure 5.16 : GPA with Filter on GPA Level Dimension.

□ Drill-down

Drill-down is an operation that allows users to view data in a lower level of aggregation. Users can click plus(+) sign of each dimension value to move down to detail of an interesting dimension. Figure 5.17 shows a detail of SC faculty (Faculty of Science), which is composed of several programs and shows the number of admitted and graduated students in 2541 Academic Year, grouped by gender.

Student Information		AcademicYear 2541					
		Gender				Grand Total	
		Female		Male			
Faculty Code	Program	Admit	Graduate	Admit	Graduate	Admit	Graduate
BM		14	17	13	12	27	29
CR		17		34		51	
DT		71	51	23	13	94	64
EG		41	38	188	178	229	216
EN		21	19	4	5	25	24
MS		25		50		75	
MT		109	117	80	59	189	176
NS		441	361	68	9	509	370
PH		105	170	83	184	188	354
PI		19		9		28	
PY		84	80	39	25	123	105
RA		240	208	75	73	315	281
SC	วิทยาศาสตร์	14		15		29	
	วิทยาศาสตร์ (คณิตศาสตร์)	19	33	8	3	27	36
	วิทยาศาสตร์ (คอมพิวเตอร์)	12	15	51	37	63	52
	วิทยาศาสตร์ (เคมี)	47	24	20	10	67	34
	วิทยาศาสตร์ (ชีววิทยา)	25	21	6	7	31	28
	วิทยาศาสตร์ (เทคโนโลยีชีวภาพ)	34	37	19	12	53	49
	วิทยาศาสตร์ (ฟิสิกส์)	4	8	4	2	8	10
	วิทยาศาสตร์ (ฟิสิกส์)	5	9	5	4	10	13
	Total	160	147	128	75	288	222
SH		20	16	13	14	33	30
SI		178	166	146	131	324	297
SP		24	18	52	34	76	52
ST				8		18	26
Grand Total		1,569	1,416	1,005	830	2,574	2,246

Figure 5.17 : Drill-down Student Information in SC Faculty.

An easy way to drill-down all faculties is by right-clicking on the Faculty Code heading and clicking Expand Items from the context menu. All faculties will be expanded and show programs under the faculty, as shown in Figure 5.18

Student Information							
AcademicYear							
2540							
Faculty Code	Program	Gender				Grand Total	
		Female		Male		Admit	Graduate
		Admit	Graduate	Admit	Graduate	Admit	Graduate
<input type="checkbox"/> BM	แพทยศาสตร์	23		35		58	
	วิทยาลัยการแพทย์					1	
	Total	23		36		59	
<input type="checkbox"/> CR	ศิลปศาสตร์ (ศาสศึกษา)	2		39		41	
	Total	2		39		41	
<input type="checkbox"/> DT	ทันตแพทยศาสตร์	55	48	30	16	85	64
	Total	55	48	30	16	85	64
<input type="checkbox"/> EG	วิศวกรรมศาสตร์	4		19		23	
	วิศวกรรมศาสตร์ (คอมพิวเตอร์)	4		32		36	
	วิศวกรรมศาสตร์ (เคมี)	11	5	18	18	29	23
	วิศวกรรมศาสตร์ (เครื่องกล)	5	1	35	30	40	31
	วิศวกรรมศาสตร์ (ไฟฟ้า)	15	8	24	55	39	63
	วิศวกรรมศาสตร์ (โยธา)	4	4	37	31	41	35
	วิศวกรรมศาสตร์ (อุตสาหกรรม)	9	5	27	17	36	22
	Total	52	23	192	151	244	174
<input type="checkbox"/> EN	วิทยาศาสตร์ (วิทยาศาสตร์และเทคโนโลยี)	22	15	6	4	28	19
	Total	22	15	6	4	28	19
<input type="checkbox"/> MT	วิทยาศาสตร์ (เทคนิคการแพทย์)	72	41	39	18	111	59
	วิทยาศาสตร์ (รังสีเทคนิค 2 ปี)	13	10	14	20	27	30
	วิทยาศาสตร์ (รังสีเทคนิค)	43	23	21	13	64	36
	Total	128	74	74	51	202	125
<input type="checkbox"/> NS	พยาบาลศาสตร์	373	202	2		375	202
	พยาบาลศาสตร์ (ต่อเนื่อง 1 ปี)	62	56	21	22	83	78
	พยาบาลศาสตร์ (ต่อเนื่อง 2 ปี)	54	58	1	3	55	61
	พยาบาลศาสตร์ (ต่อเนื่อง 3 ปีครึ่ง)	46	30	8	6	54	36
	Total	535	346	32	31	567	377
<input type="checkbox"/> PH	วิทยาศาสตร์ (โภชนาการ) 2 ปี			9	4		13
	วิทยาศาสตร์ (สาธารณสุข-โภชนาการ)	9	1	7		16	1
	วิทยาศาสตร์ (สาธารณสุขศาสตร์)	16		6		22	
	วิทยาศาสตร์ (สาธารณสุข-สุขภาพ)	3	2	2	1	5	3
	วิทยาศาสตร์ (สาธารณสุข-อนามัยชุมชน)	5		5	2	10	2
	วิทยาศาสตร์ (สาธารณสุข-อนามัยสิ่งแวดล้อม)	41	40	9	10	50	50
	วิทยาศาสตร์ (สาธารณสุข-อาชีวอนามัยและความปลอดภัย)	30	30	17	22	47	52
	วิทยาศาสตร์ (สาธารณสุข-อาหารและโภชนาการ)	7	8	7	1	14	9
	สาธารณสุขศาสตร์	67	29	102	57	169	86
	สาธารณสุขศาสตร์ (2 ปี)		2		8		10
	Total	178	121	155	105	333	226
<input type="checkbox"/> PI	แพทยศาสตร์	11		4		15	
	Total	11		4		15	
<input type="checkbox"/> PY	เภสัชศาสตร์	94	74	23	22	117	96
	Total	94	74	23	22	117	96
<input type="checkbox"/> RA	พยาบาลศาสตร์	160	144			160	144
	แพทยศาสตร์	83	60	76	47	159	107
	Total	243	204	76	47	319	251
<input type="checkbox"/> SC	วิทยาศาสตร์	25		13		38	
	วิทยาศาสตร์ (คณิตศาสตร์)	27	26	8	7	35	33
	วิทยาศาสตร์ (คอมพิวเตอร์)	17	18	46	41	63	59
	วิทยาศาสตร์ (เคมี)	48	28	12	7	60	35
	วิทยาศาสตร์ (ชีววิทยา)	47	9	8	6	55	15
	วิทยาศาสตร์ (เทคโนโลยีชีวภาพ)	31	32	22	18	53	50
	วิทยาศาสตร์ (พฤกษศาสตร์)	5	8			5	8
	วิทยาศาสตร์ (ฟิสิกส์)	6	3	7	6	13	9
	วิทยาลัยการแพทย์	1				1	
	Total	207	124	116	85	323	209
<input type="checkbox"/> SH	วิทยาศาสตร์ (เวชระเบียน)	17	31	19	9	36	40
	Total	17	31	19	9	36	40
<input type="checkbox"/> SI	แพทยศาสตร์	104	83	126	84	230	167
	วิทยาศาสตร์ (กายภาพบำบัด)	57	44	12	7	69	51
	วิทยาศาสตร์ (เวชนิศาสตร์)	3	10	9	2	12	12
	วิทยาศาสตร์ (เวชศาสตร์การธนาคารเลือด)	17	18	8	7	25	25
	Total	181	155	155	100	336	255
<input type="checkbox"/> SP	วิทยาศาสตร์การกีฬา (2 ปี)	2	4	6	8	8	12
	วิทยาศาสตร์การกีฬา (4 ปี)	15	10	37	10	52	20
	Total	17	14	43	18	60	32
<input type="checkbox"/> ST	วิทยาศาสตร์บัณฑิต (เทคโนโลยีการเพาะขยายพันธุ์สัตว์)	8	10	21	20	29	30
	Total	8	10	21	20	29	30
Grand Total		1,773	1,239	1,021	659	2,794	1,898

Figure 5.18 : Drill-down All Faculties to Display Student Information

□ Roll-up

Roll-up is the reverse operation of drill-down. Users can view data in a higher level of aggregation by clicking minus(-) sign of each dimension

value. As shown in Figure 5.18, if users click minus sign of SC faculty, the data in the program level will be hidden. Or right-click on the Faculty Code heading and select Collapse Items from the context menu, all programs will be hidden, as shown in Figure 5.19.

Student Information						
Academic Year ▼						
2541						
Gender ▼						
		Female		Male		Grand Total
Faculty Code ▼ Program	Admit	Graduate	Admit	Graduate	Admit	Graduate
BM	14	17	13	12	27	29
CR	17		34		51	
DT	71	51	23	13	94	64
EG	41	38	188	178	229	216
EN	21	19	4	5	25	24
MS	25		50		75	
MT	109	117	80	59	189	176
NS	441	361	68	9	509	370
PH	105	170	83	184	188	354
PI	19		9		28	
PY	84	80	39	25	123	105
RA	240	208	75	73	315	281
SC	160	147	128	75	288	222
SH	20	16	13	14	33	30
SI	178	166	146	131	324	297
SP	24	18	52	34	76	52
ST		8		18		26
Grand Total	1,569	1,416	1,005	830	2,574	2,246

Figure 5.19 : Roll-up Student Information to View Faculty Level.

Another way to view data in a higher level is by removing some dimensions, and the student information will be displayed, as shown in Figure 5.20.

Student Information						
Academic Year ▼						
2541						
Gender ▼						
		Female		Male		Grand Total
Admit	Graduate	Admit	Graduate	Admit	Graduate	
Here	1,569	1,416	1,005	830	2,574	2,246
Drop Row Fields						

Figure 5.20 : Roll-up Student Information.

□ Pivoting

Pivoting is an OLAP operation that is flexible and provides multi-views of information to users. Users can rotate or change perspective of presentation to view data in an alternative view. Users can change a dimension in

different views by dragging and dropping the dimension in row or column. For example, Figure 5.21 shows Student Information with Faculty dimension in row and Gender dimension in column, and an alternative view of Student Information is displaying a Faculty dimension in column and a Gender dimension in row, as shown in Figure 5.22.

Student Information						
Academic Year ▼						
2541						
Gender ▼						
		Female		Male		Grand Total
Faculty Code ▼	Admit	Graduate	Admit	Graduate	Admit	Graduate
BM	14	17	13	12	27	29
CR	17		34		51	
DT	71	51	23	13	94	64
EG	41	38	188	178	229	216
EN	21	19	4	5	25	24
MS	25		50		75	
MT	109	117	80	59	189	176
NS	441	361	68	9	509	370
PH	105	170	83	184	188	354
PI	19		9		28	
PY	84	80	39	25	123	105
RA	240	208	75	73	315	281
SC	160	147	128	75	288	222
SH	20	16	13	14	33	30
SI	178	166	146	131	324	297
SP	24		18	52	34	76
ST			8		18	26
Grand Total	1,569	1,416	1,005	830	2,574	2,246

Figure 5.21 : Student Information with Faculty in Row.

Student Information																			
Academic Year ▼																			
2541																			
Faculty Code ▼																			
		BM		CR		DT		EG		EN		MS		MT		NS		PH	
Gender ▼	Admit	Graduate	Admit	Graduate	Admit	Graduate	Admit	Graduate	Admit	Graduate	Admit	Graduate	Admit	Graduate	Admit	Graduate	Admit	Graduate	
Female	14	17	17		71	51	41	38	21	19	25	109	117	441	361	105	11		
Male	13	12	34		23	13	188	178	4	5	50	80	59	68	9	83	1		
Grand Total	27	29	51		94	64	229	216	25	24	75	189	176	509	370	188	3		

Figure 5.22 : Student Information with Faculty in Column.

A pivoting registration cube with a Payment Type dimension in row and a Payment Date dimension in column is shown in Figure 5.23. And a pivoting registration cube to show a Payment Type dimension in column and a Payment Date dimension in row is shown in Figure 5.24.

Registration					
Semester		Year			
1/2548		2004		2005	2006
		December	Total		Grand Total
Payment Type	Amount	Amount	Amount	Amount	Amount
E-Payment				84,036,355.00	142,015.00
เงินสด				8,745,455.00	8,745,455.00
เช็คของขงวิทย				15,100.00	15,100.00
เช็คธนาคาร	4,999,600.00	4,999,600.00		575,250.00	575,250.00
เช็คส่วนตัว				48,651,865.00	1,658,095.00
ตัวแลกเงินธนาคาร	1,171,600.00	1,171,600.00		55,200.00	55,200.00
Grand Total	6,171,200.00	6,171,200.00		142,471,825.00	1,800,110.00

Figure 5.23 : Registration Cube with Payment Type Dimension in Row

Registration									
Semester		Payment Type							
1/2548		E-Payment	เงินสด	เงินสด	เช็คของขงวิทย	เช็คธนาคาร	เช็คส่วนตัว	ตัวแลกเงินธนาคาร	Grand Total
Year	Month	Amount	Amount	Amount	Amount	Amount	Amount	Amount	Amount
2004	December					4,999,600.00		1,171,600.00	6,171,200.00
	Total					4,999,600.00		1,171,600.00	6,171,200.00
2005	January					25,950.00			25,950.00
	February					1,600,800.00		55,200.00	1,656,000.00
	March	1,929,140.00			575,250.00				2,504,390.00
	April	14,783,240.00	29,795.00			59,860.00			14,872,895.00
	May	20,103,735.00	230,525.00			6,031,315.00	55,200.00	337,400.00	26,758,175.00
	June	41,079,135.00	6,994,495.00	15,100.00		34,454,390.00			82,543,120.00
	July	2,484,940.00	1,427,785.00			266,955.00			4,179,680.00
	August	1,175,210.00	31,195.00			5,168,575.00			6,374,980.00
	September	1,960,235.00	22,095.00			1,030,575.00			3,012,905.00
	October	403,995.00	9,565.00			615.00			414,175.00
	November	103,780.00				12,830.00			116,610.00
	December	12,945.00							12,945.00
Total		84,036,355.00	8,745,455.00	15,100.00	575,250.00	48,651,865.00	55,200.00	392,600.00	142,471,825.00
2006	January	14,385.00				44,815.00			59,200.00
	February	35,940.00				2,530.00			38,470.00
	March	89,860.00				1,590,135.00			1,679,995.00
	April	1,830.00				20,615.00			22,445.00
	Total	142,015.00				1,658,095.00			1,800,110.00
Grand Total		84,178,370.00	8,745,455.00	15,100.00	575,250.00	55,309,560.00	55,200.00	1,564,200.00	150,443,135.00

Figure 5.24 : Registration Cube with Payment Type Dimension in Column.

A pivoting GPA cube with a Program dimension in row and a GPA Level dimension in column is shown in Figure 5.25. And a pivoting GPA cube to show a Program dimension in column and a GPA Level dimension in row is shown in Figure 5.26.

GPA									
Semester 1/2547									
GPA Level									
	2.00 - 2.24	2.25 - 2.49	2.50 - 2.74	2.75 - 2.99	3.00 - 3.24	3.25 - 3.49	3.50 - 3.74	3.75 - 4.00	Grand Total
Faculty Code Program	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student
AR	8	22	20	20	21	15	9	6	146
BM		7	18	28	45	65	42	13	222
CR	2	5	10	23	33	61	40	27	215
DT	1		20	38	87	134	99	68	458
EG	96	189	264	215	139	68	24	8	1,058
EN	12	14	21	25	17	7	5		110
MG	9	32	46	56	93	95	60	30	432
MT	16	39	102	116	112	106	61	22	585
NS	6	20	78	182	222	190	102	20	829
PH	18	50	85	117	160	141	79	42	712
PI	10	37	70	103	104	80	63	14	488
PY	6	17	44	88	99	106	75	24	472
RA	7	35	109	222	250	281	218	118	1,276
RS		7	25	26	26	11	3		102
SC	104	159	275	270	242	184	101	77	1,615
SH				3	13	28	10	9	64
SI	3	18	86	172	297	354	351	274	1,668
SP	23	57	49	50	30	22	14	9	280
VS	3	1	16	17	49	32	26	11	163
Grand Total	324	709	1,338	1,771	2,039	1,980	1,382	772	10,895

Figure 5.25 : GPA Cube with Program Dimension in Row.

GPA										
Semester 1/2547										
Faculty Code Program										
	PI	PY	RA	RS	SC	SH	SI	SP	VS	Grand Total
GPA Level	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student
0 - 0.49	1	2	3	1	54		2	3	3	107
0.50 - 0.99		1			13		1	6		23
1.00 - 1.24		1	1		11			8	1	32
1.25 - 1.49	2	1	1		19		1	5	1	44
1.50 - 1.74	2	2	1	3	51			3		123
1.75 - 1.99	10	6	7		104		3	23	3	324
2.00 - 2.24	37	17	35	7	159		18	57	1	709
2.25 - 2.49	70	44	109	25	275		86	49	16	1,338
2.50 - 2.74	103	88	222	26	270	3	172	50	17	1,771
2.75 - 2.99	104	99	250	26	242	13	297	30	49	2,039
3.00 - 3.24	80	106	281	11	184	28	354	22	32	1,980
3.25 - 3.49	63	75	218	3	101	10	351	14	26	1,382
3.50 - 3.74	14	24	118		77	9	274	9	11	772
3.75 - 4.00	2	6	30		55	1	109	1	3	251
Grand Total	488	472	1,276	102	1,615	64	1,668	280	163	10,895

Figure 5.26 : GPA Cube with Program Dimension in Column.

Pivoting helps users to set a dimension in an appropriate view. Which dimensions in row and which dimensions in column depend on the nature of dimensions.

5.1.2.3 MU-SDM Multidimensional Analysis

MU-SDM provides eight cubes of multidimensional analysis, as listed below.

- Student Information Cube

This cube shows the number of students that is admitted and graduated in each academic year. The cube is composed of two measures and four dimensions listed below, and is shown as hierarchy in Figure 5.27.

- **Admit** is a measure that shows the number of admitted students
- **Graduate** is a measure that shows the number of graduated students
- **AcademicYear** is a dimension that shows an academic year
- **Gender** is a dimension that shows gender of students
- **Nationality** is a dimension that shows nationality of students
- **Program** is a dimension that shows a program of students. There are two levels of the dimension, namely: Faculty Code and Program. Faculty Code represents a faculty of students and Program field represents programs under the faculty.

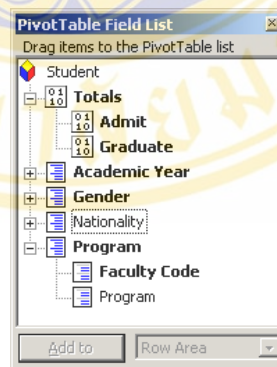


Figure 5.27 : Table Field List of Student Information Cube.

Student Information is shown as a pivot table in Figure 5.28. Users can perform a multidimensional analysis function to view appropriate information as needed. Graphic views of Student Information are shown in Figure 5.29, Student Information graph with series in row, and Figure 5.30, Student Information graph with series in column.

Student Information								
Drop Filter Fields Here								
		Gender						
		Female		Male		Grand Total		
Academic Year	Faculty Code	Program	Admit	Graduate	Admit	Graduate	Admit	Graduate
2533			698	4	434		1,132	4
2534			959	3	523	22	1,482	25
2535			1,160	204	608	161	1,768	365
2536			1,326	694	758	351	2,084	1,045
2537			1,555	813	825	353	2,380	1,166
2538			1,696	1,064	933	493	2,629	1,557
2539			1,816	1,197	984	651	2,800	1,848
2540			1,773	1,239	1,021	659	2,794	1,898
2541			1,569	1,416	1,005	830	2,574	2,246
2542			1,576	1,462	1,079	793	2,655	2,255
2543			1,621	1,479	1,084	801	2,705	2,280
2544			1,517	1,401	897	801	2,414	2,202
2545			1,612	1,275	986	767	2,600	2,042
2546			1,912	1,410	1,245	790	3,157	2,200
2547			2,401	1,298	1,356	817	3,757	2,115
2548			2,592	1,347	1,418	801	4,010	2,148
Grand Total			25,783	16,306	15,158	9,090	40,941	25,396

Figure 5.28 : Student Information Pivot Table.

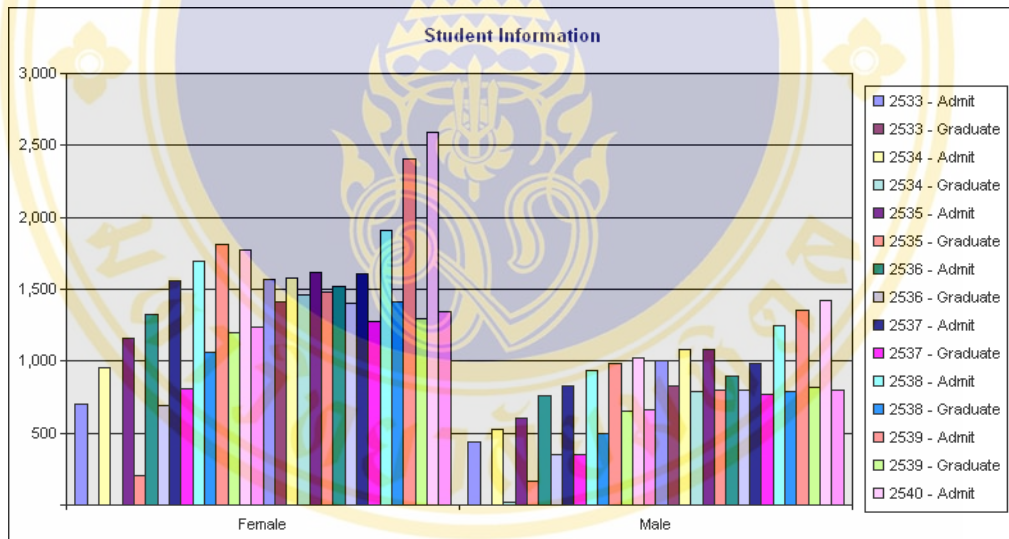


Figure 5.29 : Student Information Graph with Series in Row

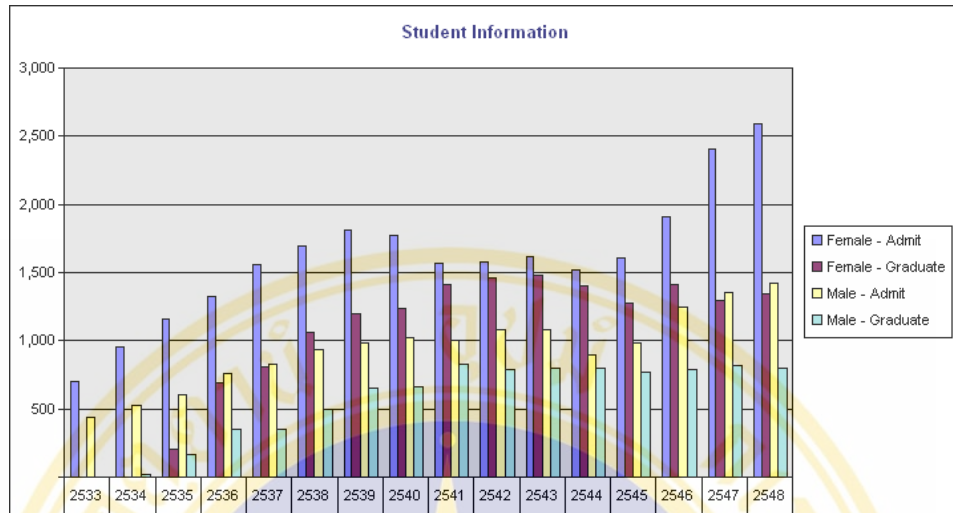


Figure 5.30 : Student Information Graph with Series in Column

In Figure 5.28, users can drill-down to academic year 2542 in order to view the program level by moving Academic Year field to Filter Field Area and checking on academic year 2542. Expand faculty to view all programs, and the pivot table will be shown in Figure 5.31.

Student Information		Gender		Grand Total		
Academic Year						
2542						
Faculty Code	Program	Female Admit	Female Graduate	Male Admit	Male Graduate	Grand Total
PI	แพทยศาสตร์	45	37	82		82
	Total	45	37	82		82
PY	เภสัชศาสตร์	83	74	40	23	123
	เภสัชศาสตร์ (พิเศษ)	9	3			12
	Total	92	74	43	23	135
RA	พยาบาลศาสตร์	141	140	6		147
	แพทยศาสตร์	79	82	68	53	147
	วิทยาศาสตร์การแพทย์	2	5	1	7	1
	Total	222	222	79	54	301
SC	วิทยาศาสตร์	22		14		36
	วิทยาศาสตร์ (คณิตศาสตร์)	33	23	20	6	53
	วิทยาศาสตร์ (คอมพิวเตอร์)	17	9	39	38	56
	วิทยาศาสตร์ (เคมี)	41	25	19	9	60
	วิทยาศาสตร์ (ชีววิทยา)	44	32	19	5	63
	วิทยาศาสตร์ (เทคโนโลยีชีวภาพ)	42	30	8	19	50
	วิทยาศาสตร์ (พฤกษศาสตร์)	17	6	15	1	32
	วิทยาศาสตร์ (ฟิสิกส์)	8	2	14	3	22
	Total	224	127	148	81	372
SH	วิทยาศาสตร์ (เวชระเบียน)	17	12	8	11	25
	Total	17	12	8	11	25
SI	แพทยศาสตร์	106	117	118	111	224
	วิทยาศาสตร์ (กายภาพบำบัด)	35	38	19	11	54
	วิทยาศาสตร์ (เวชทันตกรรม)	6	8	6	4	12
	วิทยาศาสตร์ (เวชศาสตร์การธนาคารเลือด)	16	19	10	6	26
	วิทยาศาสตร์การแพทย์	2		3	4	5
	Total	165	182	156	136	318
SP	วิทยาศาสตร์การกีฬา (2 ปี)	6	3	5	9	11
	วิทยาศาสตร์การกีฬา (4 ปี)	21	16	32	28	53
	Total	27	19	37	37	64
VS	สัตวแพทยศาสตร์	24		8		32
	Total	24		8		32
Grand Total		1,576	1,462	1,079	795	2,655

Figure 5.31 : Student Information with Academic Year 2542.

□ Student Graduation Cube

The cube is composed of one measure and five dimensions listed below, and is shown in Table Field List as hierarchy in Figure 5.32.

- **#Student** is a measure that shows the number of students.
- **Student(%)** is a measure that shows percentage of students.
- **Admit Year** is a dimension that shows academic year that a student was admitted.
- **Gender** is a dimension that shows gender of students
- **Nationality** is a dimension that shows nationality of students
- **Program** is a dimension that shows a program of students. There are two levels of the dimension, namely: Faculty Code and Program. Faculty Code represents a faculty of students and Program field represents a program under the faculty.
- **Status** is a dimension that shows academic status of students. There are two fields of the dimension that are represented as hierarchical level namely: Status field and Grad Year field. Status field is represented in two values, Grad for graduate student status and Non-Grad for non-graduate student status. Grad Year field represents an academic year that students graduate or resign.

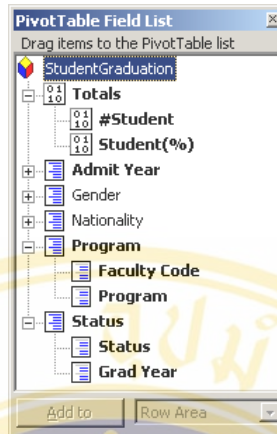


Figure 5.32 : Table Field List of Student Graduation Cube.

A Student Graduation cube is shown as a pivot table in Figure 5.33. Users can use a multidimensional analysis function to view appropriate information as needed. Graphic views of Student Graduation are shown in Figure 5.34, Student Graduation graph with series in row, and Figure 5.35, Student Graduation graph with series in column.

Student Graduation										
Admit Year										
2542										
Status										
Graduated										
Grad Year										
2543 2544 2545 2546 2547 2548 Total										
Faculty Code										
Program										
#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student
						29	1	30	1	31
			10	11	6	1	28	21		49
					66	2	68	6		74
			129	32	14	4	179	43		222
			12				12	6		18
		1	6	12	3		22	33		55
	29		136				165	42		207
	71		275	9			355	43		398
		96	126	4			226	43		269
					72	5	77	5		82
			10	92	1		103	32		135
			137	4	140	3	284	17		301
	44	4	260	4			312	60		372
	22						22	3		25
	37	1	49	1	213	7	308	13		321
	9	1	42	5	1		58	6		64
					28		28	4		32
Grand Total	212	103	1,192	174	573	23	2,277	378		2,655

Figure 5.33 : Student Graduation Pivot Table.

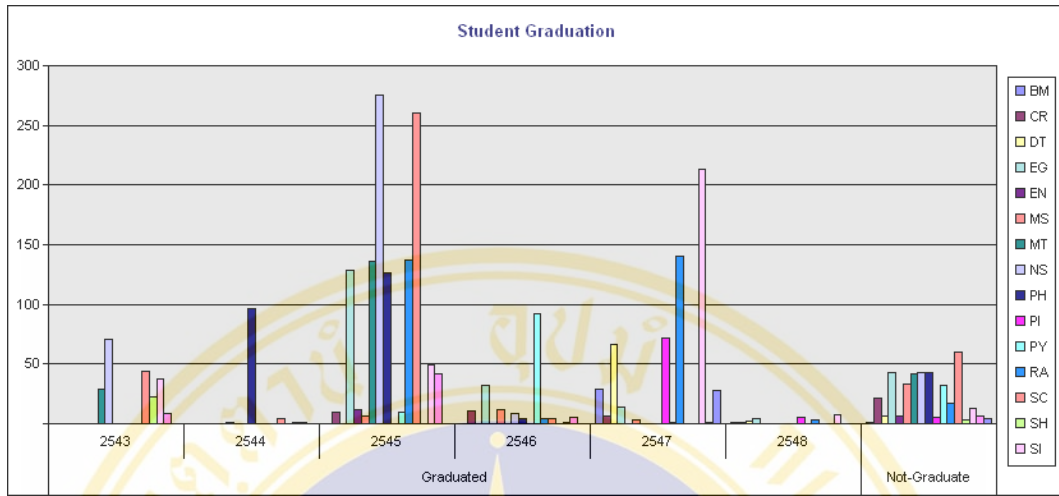


Figure 5.34 : Student Graduation Graph with Series in Row

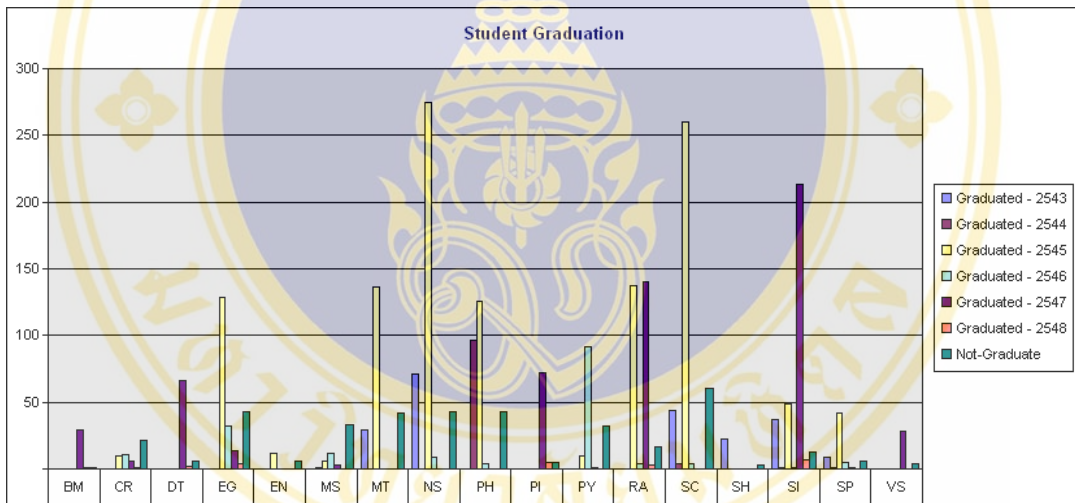


Figure 5.35 : Student Graduation Graph with Series in Column

Figure 5.36 shows Student Graduation Pivot Table that represents the percentage of graduated and not-graduate students.

Student Graduation										
Admit Year										
2542										
Faculty Code	Program	Status						Total	Not-Graduate	Grand Total
		Graduated								
		2543	2544	2545	2546	2547	2548			
		Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	
BM						1.09%	0.04%	1.13%	0.04%	1.17%
CR				0.38%	0.41%	0.23%	0.04%	1.05%	0.79%	1.85%
DT						2.49%	0.08%	2.56%	0.23%	2.79%
EG				4.86%	1.21%	0.53%	0.15%	6.74%	1.62%	8.36%
EN				0.45%				0.45%	0.23%	0.68%
MS			0.04%	0.23%	0.45%	0.11%		0.83%	1.24%	2.07%
MT		1.09%		5.12%				6.21%	1.58%	7.80%
NS		2.67%		10.36%	0.34%			13.37%	1.62%	14.99%
PH			3.62%	4.75%	0.15%			8.51%	1.62%	10.13%
PI						2.71%	0.19%	2.90%	0.19%	3.09%
PY				0.38%	3.47%	0.04%		3.88%	1.21%	5.08%
RA				5.16%	0.15%	5.27%	0.11%	10.70%	0.64%	11.34%
SC		1.66%	0.15%	9.79%	0.15%			11.75%	2.26%	14.01%
SH		0.83%						0.83%	0.11%	0.94%
SI		1.39%	0.04%	1.85%	0.04%	8.02%	0.26%	11.60%	0.49%	12.09%
SP		0.34%	0.04%	1.58%	0.19%	0.04%		2.18%	0.23%	2.41%
VS						1.05%		1.05%	0.15%	1.21%
Grand Total		7.98%	3.88%	44.90%	6.55%	21.58%	0.87%	85.76%	14.24%	100.00%

Figure 5.36 : Student Graduation Pivot Table with Percentage of Students.

Registration Cube

The cube is composed of forty measures and five dimensions listed below and shown in Table Field List as hierarchy in Figure 5.37.

- **Amount** is a measure that shows amount of invoices.
- **#Invoice** is a measure that shows number of invoices.
- **#Student** is a measure that shows number of students.
- **A01** is a measure that shows amount of admission fee
- **A02** is a measure that shows amount of damage deposit
- **A03** is a measure that shows amount of document fee (New student)
- **A04** is a measure that shows amount of student identification card fee
- **B01** is a measure that shows amount of university fee
- **B02** is a measure that shows amount of library fee
- **B03** is a measure that shows amount of health insurance fee
- **B04** is a measure that shows amount of sport facilities fee

- **B05** is a measure that shows amount of art and recreation fee
- **B06** is a measure that shows amount of college fee
- **B07** is a measure that shows amount of university activity fee
- **B08** is a measure that shows amount of Bank fee
- **B09** is a measure that shows amount of Internet Fee
- **B10** is a measure that shows amount of computing and database services
- **B11** is a measure that shows amount of add/drop fee
- **C11** is a measure that shows amount of tuition fee (lecture)
- **C12** is a measure that shows amount of tuition fee (laboratory)
- **C31** is a measure that shows amount of tuition fee (lecture) for summer
- **C32** is a measure that shows amount of tuition fee (laboratory) for summer
- **D01** is a measure that shows amount of damage
- **D02** is a measure that shows amount of dormitory fee.
- **D03** is a measure that shows amount of dormitory deposit fee.
- **D04** is a measure that shows amount of Late Registration Fee
- **D05** is a measure that shows amount of student status maintenance fee
- **D06** is a measure that shows amount of other fee
- **E01** is a measure that shows amount of document fee
- **M01** is a measure that shows amount of major fee for nursing science
- **M02** is a measure that shows amount of major fee for medical records

- **M03** is a measure that shows amount of major fee for computer science
- **M04** is a measure that shows amount of major fee for engineering
- **M06** is a measure that shows amount of major fee for transfusion medicine
- **M07** is a measure that shows amount of major fee for radiological technology
- **M08** is a measure that shows amount of major fee for dental surgery
- **M09** is a measure that shows amount of major fee for pharmacy
- **M10** is a measure that shows amount of major fee for nursing science
- **M11** is a measure that shows amount of major fee for dental technology
- **M12** is a measure that shows amount of major fee for veterinary medicine
- **M13** is a measure that shows amount of major fee for public health
- **M14** is a measure that shows amount of major fee for physiotherapy
- **M15** is a measure that shows amount of major fee for ICT
- **M16** is a measure that shows amount of major fee for applied Thai traditional medicine
- **Class** is a dimension that shows class of students.
- **Payment Date** is a time dimension that is composed of three fields, namely, Year, Month and Day.
- **Payment Type** is a dimension is represent method of payment.

- **Program** is a dimension that shows program of students. There are two levels of the dimension, namely: Faculty Code and Program. Faculty Code represents a faculty of students and Program field represents a program under the faculty
- **Semester** is a dimension that students make a registration. There are two fields of dimension that are represented as a hierarchical level, namely: Acd Year field and Semester field. Acd Year represents an academic year of registration, and Semester field represents a semester under an academic year.

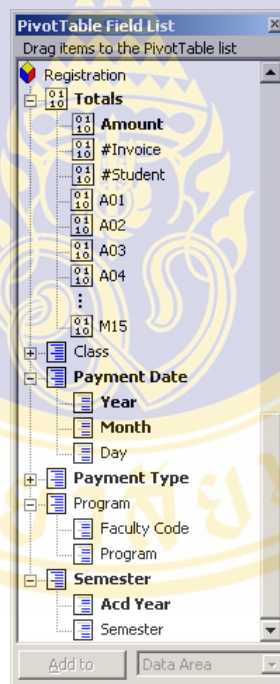


Figure 5.37 : Table Field List of Registration Cube.

A Registration cube is shown as a pivot table in Figure 5.38. Users can use a multidimensional analysis function to view appropriate information as needed. Graphic views of the Registration cube are shown in Figure 5.39, Registration graph with series in row, and Figure 5.40, Registration graph with series in column.

Registration										
Semester 1/2548										
		Payment Type								
		E-Payment	เงินสด	เช็คของขงขั้ว	เช็คธนาคาร	เช็คส่วนตัว	ตัวแลกเงินธนาคาร	Grand Total		
Year	Month	Amount	Amount	Amount	Amount	Amount	Amount	Amount	Amount	
2004	December					4,999,600.00		1,171,600.00	6,171,200.00	
	Total					4,999,600.00		1,171,600.00	6,171,200.00	
2005	January					25,950.00			25,950.00	
	February					1,600,800.00		55,200.00	1,656,000.00	
	March	1,929,140.00		575,250.00					2,504,390.00	
	April	14,783,240.00	29,795.00			59,860.00			14,872,895.00	
	May	20,103,735.00	230,525.00			6,031,315.00	55,200.00	337,400.00	26,758,175.00	
	June	41,079,135.00	6,994,495.00	15,100.00		34,454,390.00			82,543,120.00	
	July	2,484,940.00	1,427,785.00			266,955.00			4,179,680.00	
	August	1,175,210.00	31,195.00			5,168,575.00			6,374,980.00	
	September	1,960,235.00	22,095.00			1,030,575.00			3,012,905.00	
	October	403,995.00	9,565.00			615.00			414,175.00	
	November	103,780.00				12,830.00			116,610.00	
	December	12,945.00							12,945.00	
	Total	84,036,355.00	8,745,455.00	15,100.00	575,250.00	48,651,865.00	55,200.00	392,600.00	142,471,825.00	
2006	January	14,385.00				44,815.00			59,200.00	
	February	35,940.00				2,530.00			38,470.00	
	March	89,860.00				1,590,135.00			1,679,995.00	
	April	1,830.00				20,615.00			22,445.00	
	Total	142,015.00				1,658,095.00			1,800,110.00	
Grand Total		84,178,370.00	8,745,455.00	15,100.00	575,250.00	55,309,960.00	55,200.00	1,564,200.00	150,443,135.00	

Figure 5.38 : Registration Pivot Table

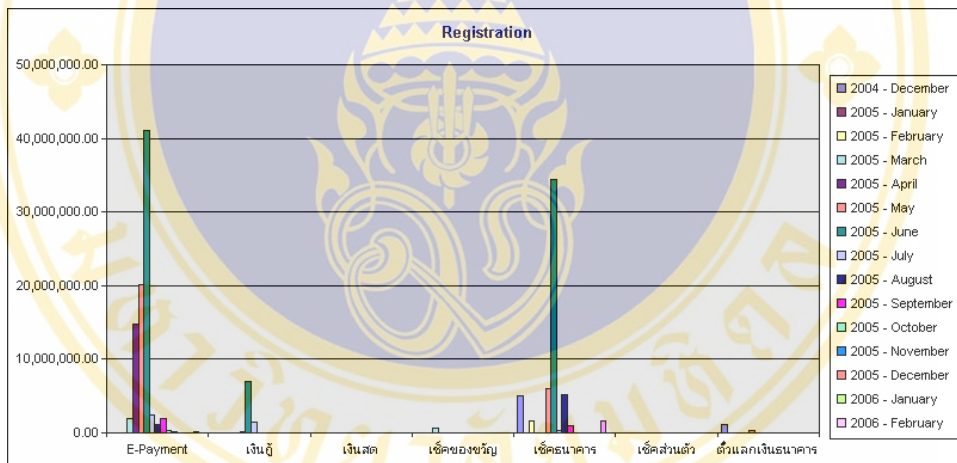


Figure 5.39 : Registration Graph with Series in Row

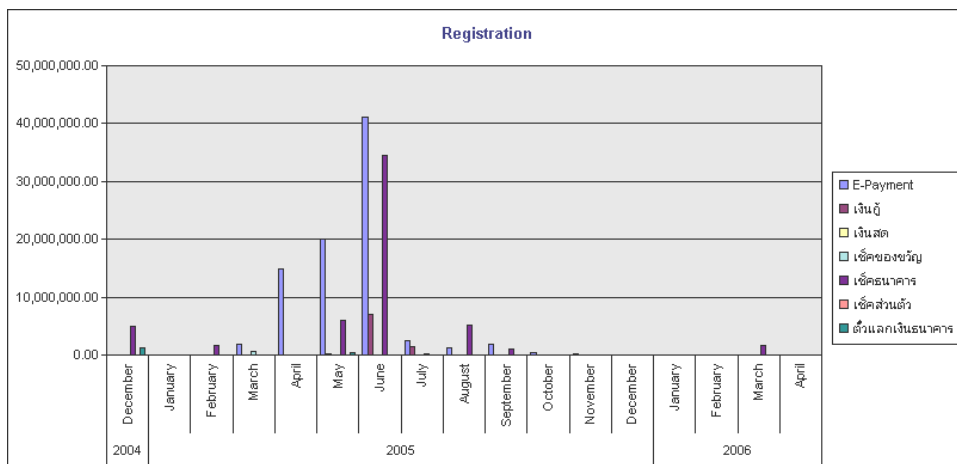


Figure 5.40 : Registration Graph with Series in Column

In Figure 5.38, a Registration Pivot Table shows the amount of registration income in semester 1/2548 grouped by year and month of payment and payment type. Users can add another interesting measure that represents the number of students by dragging #Student field from Table Field List and dropping into the data area, and the pivot table will be shown in Figure 5.41.

Registration											
Semester		Payment Type									
1/2548		E-Payment		เงินสด		เช็คของขงวิทยาลัย		เช็คธนาคาร			
Year	Month	Amount	#Student	Amount	#Student	Amount	#Student	Amount	#Student	Amount	#Student
2004	December									4,999,600.00	103
	Total									4,999,600.00	103
2005	January									25,950.00	3
	February									1,600,800.00	29
	March	1,929,140.00	171					575,250.00	65		
	April	14,783,240.00	1,096	29,795.00	3					59,860.00	4
	May	20,103,735.00	2,019	230,525.00	25					6,031,315.00	192
	June	41,079,135.00	3,449	6,994,495.00	863	15,100.00	2			34,454,390.00	3,180
	July	2,484,940.00	556	1,427,785.00	149					266,955.00	38
	August	1,175,210.00	404	31,195.00	3					5,168,575.00	284
	September	1,960,235.00	1,669	22,095.00	3					1,030,575.00	68
	October	403,995.00	204	9,565.00	1					615.00	1
	November	103,780.00	52							12,830.00	3
	December	12,945.00	3								
	Total	84,036,355.00	9,623	8,745,455.00	1,047	15,100.00	2	575,250.00	65	48,651,865.00	3,802
2006	January	14,385.00	9							44,815.00	2
	February	35,940.00	6							2,530.00	2
	March	89,860.00	4							1,590,135.00	58
	April	1,830.00	2							20,615.00	1
	Total	142,015.00	21							1,658,095.00	63
Grand Total		84,178,370.00	9,644	8,745,455.00	1,047	15,100.00	2	575,250.00	65	55,309,560.00	3,966

Figure 5.41 : Registration Pivot Table

□ Subject Cube

A Subject cube represents the number of enrolled students and the amount of tuition fee of subjects. This cube is composed of two measures and four dimensions listed below and shown as hierarchy in Figure 5.42.

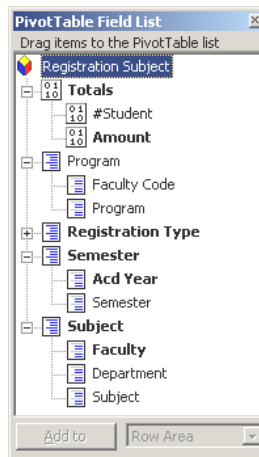


Figure 5.42 : Table Field List of Subject Cube.

- **#Student** is a measure that shows number of students.
- **Amount** is a measure that shows tuition fee that students pay for a subject.
- **Program** is a dimension that shows a program of students. There are two levels of the dimension, namely: Faculty Code and Program. Faculty Code represents a faculty of students and Program field represents a program under the faculty.
- **Registration Type** is a dimension. There are four types of registrations such as normal, add, drop and withdraw.
- **Semester** is a dimension that students enroll in a subject. There are two fields of dimension that are represented as a hierarchical level, namely: Acd Year field and Semester field. Acd Year field represents an academic year of registration and Semester field represents a semester under an academic year.
- **Subject** is a dimension. There are three levels of the dimension, namely: Faculty, Department and Subject fields. Faculty field represents a faculty, Department field represents a department that operates the subject, and Subject field represents the code and name of a subject under the faculty.

Figure 5.43 shows a pivot table of Subject. It shows tuition fee of an enrolled subject of students that the university gets in academic year 2548, grouped by faculty. Users can use a multidimensional analysis function to view appropriate information as needed. Graphic views of the Subject cube are shown in Figure 5.44, Subject graph with series in row, and Figure 5.45, Subject graph with series in column.

Subject		Registration Type				
Semester		Normal	Add	Drop	Withdraw	Grand Total
2548		Amount	Amount	Amount	Amount	Amount
Faculty	Subject					
	BANGKOK METROPOLITAN MEDICAL COLLEGE	26,400.00	0.00	0.00	0.00	26,400.00
	COLLEGE OF MANAGEMENT	3,079,800.00	195,000.00	31,800.00	3,000.00	3,309,600.00
	COLLEGE OF MUSIC	709,200.00	226,200.00			935,400.00
	COLLEGE OF RELIGIOUS STUDIES	1,567,200.00	10,800.00	4,200.00	7,200.00	1,589,400.00
	COLLEGE OF SPORTS SCIENCE AND TECHNOLOGY	1,632,000.00	1,200.00	600.00		1,633,800.00
	FACULTY OF ARTS	3,269,200.00	16,800.00	934,000.00	18,000.00	4,238,000.00
	FACULTY OF DENTISTRY	3,267,400.00	0.00	2,000.00	800.00	3,270,200.00
	FACULTY OF ENGINEERING	6,416,600.00	922,200.00	44,400.00	164,000.00	7,547,200.00
	FACULTY OF ENVIRONMENT AND RESOURCE STUDIES	487,600.00	31,200.00			518,800.00
	FACULTY OF MEDICAL TECHNOLOGY	2,548,000.00	20,800.00	5,600.00		2,574,400.00
	FACULTY OF MEDICINE RAMATHIBODI HOSPITAL	8,540,700.00	692,400.00	95,600.00	8,200.00	9,336,900.00
	FACULTY OF MEDICINE SIRIRAJ HOSPITAL	13,772,950.00	187,400.00	3,600.00	6,200.00	13,970,150.00
	FACULTY OF NURSING	3,575,400.00	0.00	600.00		3,576,000.00
	FACULTY OF PHARMACY	4,910,600.00	1,200.00		6,600.00	4,918,400.00
	FACULTY OF PHYSICAL THERAPY & APPLIED MOVEMENT SCIEN		18,200.00			18,200.00
	FACULTY OF PUBLIC HEALTH	6,644,600.00	1,200.00	7,600.00		6,653,400.00
	FACULTY OF SCIENCE	51,228,600.00	637,200.00	1,380,000.00	761,800.00	54,007,600.00
	FACULTY OF SOCIAL SCIENCES AND HUMANTIES	5,471,000.00	992,600.00	86,800.00	400.00	6,550,800.00
	FACULTY OF TROPICAL MEDICINE	46,800.00				46,800.00
	FACULTY OF VETERINARY SCIENCE	1,291,400.00	4,200.00		4,600.00	1,300,200.00
	PRABOROMARAUCHANOK INSTITUTE	41,400.00	0.00	0.00		41,400.00
	RATCHASUDA COLLEGE	776,200.00	3,400.00		3,600.00	783,200.00
	Grand Total	119,303,050.00	3,962,000.00	2,596,800.00	984,400.00	126,846,250.00

Figure 5.43 : Subject Pivot Table

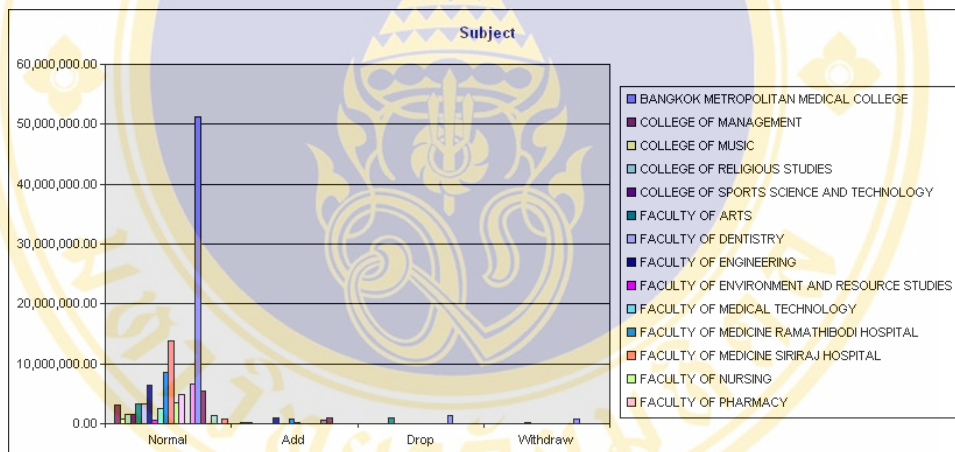


Figure 5.44 : Subject Graph with Series in Row.

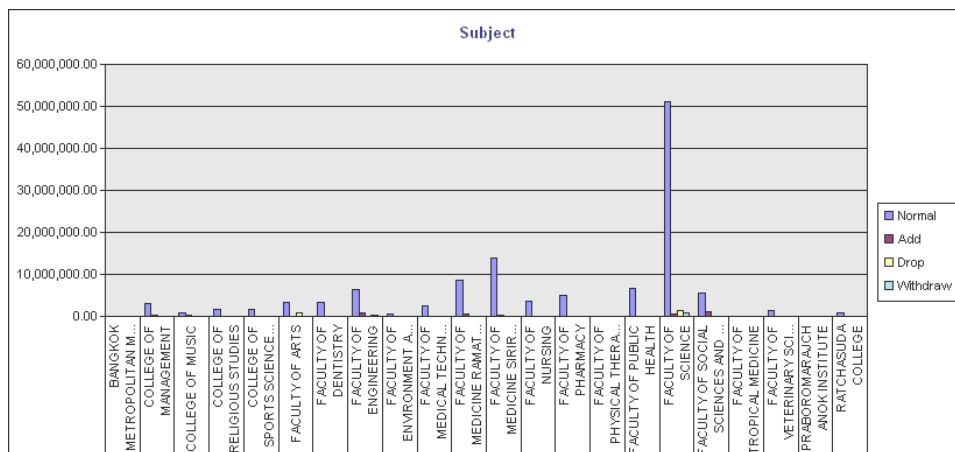


Figure 5.45 : Subject Graph with Series in Column.

Based on Figure 5.43, users can drill-down to the subject level to view a detail under the faculty, for example, expand Faculty of Science. Additionally, users can add #Student measure from Table Filed List to the data area in order to view the number of students. The Subject Pivot Table is shown in Figure 5.46.

Subject		Registration Type						v/with
		Normal		Add		Drop		
Faculty	Subject	Amount	#Student	Amount	#Student	Amount	#Student	Amc
BANGKOK METROPOLITAN MEDICAL COLLEGE		26,400.00	1,449	0.00		4	0.00	2
COLLEGE OF MANAGEMENT		3,079,800.00	5,082	195,000.00		367	31,800.00	52
COLLEGE OF MUSIC		709,200.00	1,182	226,200.00		572		
COLLEGE OF RELIGIOUS STUDIES		1,567,200.00	2,582	10,800.00		20	4,200.00	7
COLLEGE OF SPORTS SCIENCE AND TECHNOLOGY		1,632,000.00	3,279	1,200.00		2	600.00	1
FACULTY OF ARTS		3,269,200.00	4,767	7,200.00		1,361	934,000.00	1,299
FACULTY OF DENTISTRY		3,267,400.00	9,627	0.00		1	2,000.00	5
FACULTY OF ENGINEERING		6,416,600.00	10,546	922,200.00		1,404	44,400.00	76
FACULTY OF ENVIRONMENT AND RESOURCE STUDIES		487,600.00	821	31,200.00		78		
FACULTY OF MEDICAL TECHNOLOGY		2,548,000.00	5,645	20,800.00		77	5,600.00	14
FACULTY OF MEDICINE RAMATHIBODI HOSPITAL		8,497,300.00	9,393	682,200.00		891	95,600.00	152
FACULTY OF MEDICINE SIRIRAJ HOSPITAL		13,772,950.00	16,497	187,400.00		271	3,600.00	7
FACULTY OF NURSING		3,315,400.00	6,141	0.00		1	600.00	1
FACULTY OF PHARMACY		4,910,600.00	5,726	1,200.00		2		
FACULTY OF PHYSICAL THERAPY & APPLIED MOVEMENT SCIENCE				18,200.00		91		
FACULTY OF PUBLIC HEALTH		6,565,400.00	6,813	1,200.00		8	7,600.00	9
FACULTY OF SCIENCE	SCAG211 SC&TECHNO OF ECONOMIC CROP PRO	16,800.00	21					
	SCAG221 SC&TECHNO OF ECONO ANIMAL PRO	21,000.00	21	1,000.00		1		
	SCAG312 INDUSTRIAL CROP SC FOR ENERGY	18,400.00	23				800.00	1
	SCAG313 PHYSIOLOGY IN CROP PRODUCTION	18,400.00	23				800.00	1
	SCAG314 PLANT PROPAGATION & SEED SC	18,400.00	23					
	SCAG331 SOIL SCIENCE	18,400.00	23				800.00	1
	SCAG401 PRIN OF INTEGRAT PLANT PEST.	17,600.00	23	0.00		1		
	SCAG403 PRIN OF PLANT PATHOLOGY & PRO	13,600.00	17					
	SCAG411 PLANT BREEDING	18,400.00	23					
	SCAG412 ORNAMENTAL PLANTS & LANDSCAP	800.00	1	16,000.00		20		
	SCAG413 MUSHROOM PRODUCTION	12,800.00	16				800.00	1
	SCAG414 PLANT TISSUE CULTURE FOR AGRIC	6,400.00	8				8,000.00	10
	SCAG416 PLANT HYDROPONIC TECHNOLOGY	8,000.00	10	27,200.00		34	800.00	1
	SCAG452 ALCO FERMENT&DISTILL TECH FOR	13,600.00	17	800.00		2		
	SCAG457 POSTHARVEST SCIENCE & TECHNOLO	18,400.00	23	4,000.00		6		
	SCAG462 TRAINING	9,600.00	24					
	SCAG476 BIOTECHNOLOGY IN AGRICULTURE	17,600.00	22	0.00		1		
	SCAG487 SEMINAR	9,200.00	23	0.00		1		

Figure 5.46 : Drill-down Subject Pivot Table

Allocation Cube

An Allocation cube represents the amount of allocated registration income that a faculty/institute gets for a semester. After students had paid for their registration, this income was allocated to related institutes according to the allocation rule. The allocation is performed several times per semester depending on the amount of registration income or a registration period. Allocation information is useful since it makes the executive know how much a faculty/institute gets income from registration.

This cube is composed of thirty-nine measures and two dimensions listed below and shown as hierarchy in Figure 5.47.

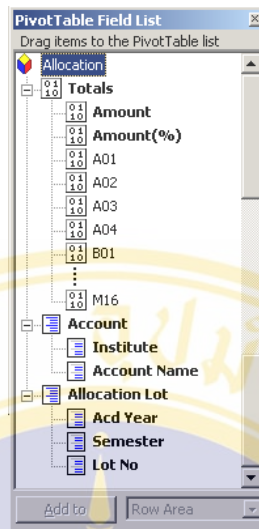


Figure 5.47 : Table Field List of Allocation Cube.

- **Amount** is a measure that shows allocated income.
- **Amount(%)** is a measure that shows the percentage of allocated income.
- **A01** is a measure that shows allocated amount of admission fee.
- **A02** is a measure that shows allocated amount of damage deposit.
- **A03** is a measure that shows allocated amount of document fee (New student).
- **A04** is a measure that shows allocated amount of student identification card fee.
- **B01** is a measure that shows allocated amount of university fee.
- **B02** is a measure that shows allocated amount of library fee.
- **B03** is a measure that shows allocated amount of health insurance fee.
- **B04** is a measure that shows allocated amount of sport facilities fee.

- **B05** is a measure that shows allocated amount of art and recreation fee.
- **B06** is a measure that shows allocated amount of college fee.
- **B07** is a measure that shows allocated amount of university activity fee.
- **B08** is a measure that shows allocated amount of Bank fee.
- **B09** is a measure that shows allocated amount of Internet fee.
- **B10** is a measure that shows allocated amount of computing and database services fee.
- **B11** is a measure that shows allocated amount of add/drop fee.
- **C11** is a measure that shows allocated amount of tuition fee (lecture).
- **C12** is a measure that shows allocated amount of tuition fee (laboratory).
- **C31** is a measure that shows allocated amount of tuition fee (lecture) for summer.
- **C32** is a measure that shows allocated amount of tuition fee (laboratory) for summer.
- **D01** is a measure that shows allocated amount of damage.
- **D02** is a measure that shows allocated amount of dormitory fee.
- **D03** is a measure that shows allocated amount of dormitory deposit fee.
- **D04** is a measure that shows allocated amount of Late Registration Fee.
- **D05** is a measure that shows allocated amount of student status maintenance fee.

- **D06** is a measure that shows allocated amount of other fee.
- **E01** is a measure that shows allocated amount of document fee.
- **M01** is a measure that shows allocated amount of major fee for a nursing science program.
- **M02** is a measure that shows allocated amount of major fee for a medical record program.
- **M03** is a measure that shows allocated amount of major fee for a computer science program.
- **M04** is a measure that shows allocated amount of major fee for an engineering program.
- **M06** is a measure that shows allocated amount of major fee for a transfusion medicine program.
- **M07** is a measure that shows allocated amount of major fee for a radiological technology program.
- **M08** is a measure that shows allocated amount of major fee for a dental a surgery program.
- **M09** is a measure that shows allocated amount of major fee for a pharmacy program.
- **M10** is a measure that shows amount of major fee for a nursing science program.
- **M11** is a measure that shows amount of major fee for a dental technology program.
- **M12** is a measure that shows amount of major fee for a veterinary medicine program.
- **M13** is a measure that shows allocated amount of major fee for a public health program.
- **M14** is a measure that shows allocated amount of major fee for a physiotherapy
- **M15** is a measure that shows allocated amount of major fee for an ICT program.

- **M16** is a measure that shows allocated amount of major fee for an applied Thai traditional medicine program.
- **Account** is a dimension that is composed of two fields, namely: Institute field and Account Name field. Institute field is a faculty or institute that gets allocation income. Account Name field is an account of a faculty or institute.
- **Allocation Lot** is a dimension that is composed of three fields, represented as a hierarchical level, namely: Acd Year, Semester and Lot No. Acd Year field represents an academic year. Semester field represents a registration semester. Lot No field represents Lot of allocation in a semester.

Figure 5.48 shows a pivot table of Allocation. It shows an allocation amount in the academic year 2547 that is expanded in semester level and grouped by institute. Users can use multidimensional analysis functions to view appropriate information as needed. Graphic views of Allocation are shown in Figure 5.49, Allocation graph with series in row, and Figure 5.50, Allocation graph with series in column.

Allocation					
Drop Filter Fields Here					
Acad Year ▾ Semester					Grand Total
2547					
1/2547			2/2547		
Institute ▾ Account Name	Amount	Amount	Amount	Total Amount	Amount
กองกิจการนักศึกษา	9,254,300.00	2,553,530.00		11,807,830.00	11,807,830.00
คณะกายภาพบำบัดและวิทยาศาสตร์การน	767,465.00	931,870.00		1,699,335.00	1,699,335.00
คณะทันตแพทยศาสตร์	3,079,260.00	3,418,675.00	600.00	6,498,535.00	6,498,535.00
คณะเทคนิคการแพทย์	1,352,355.00	1,283,360.00	199,170.00	2,834,885.00	2,834,885.00
คณะพยาบาลศาสตร์	1,319,372.50	1,356,702.50	235,800.00	2,911,875.00	2,911,875.00
คณะแพทยศาสตร์ โรงพยาบาลรามธิบดี	2,757,265.00	3,183,165.00	162,840.00	6,103,270.00	6,103,270.00
คณะแพทยศาสตร์ศิริราชพยาบาล	7,196,620.00	1,948,795.00	155,070.00	9,300,485.00	9,300,485.00
คณะเภสัชศาสตร์	3,369,875.00	4,015,825.00	480.00	7,386,180.00	7,386,180.00
คณะวิทยาศาสตร์	15,013,175.00	12,642,575.00	625,200.00	28,280,950.00	28,280,950.00
คณะวิศวกรรมศาสตร์	6,320,145.00	7,259,540.00	98,520.00	13,678,205.00	13,678,205.00
คณะเวชศาสตร์เขตร้อน		21,600.00		21,600.00	21,600.00
คณะศิลปศาสตร์	332,150.00	400,150.00	10,800.00	743,100.00	743,100.00
คณะสังคมศาสตร์และมนุษยศาสตร์	1,757,860.00	1,689,345.00		3,447,205.00	3,447,205.00
คณะสัตวแพทยศาสตร์	1,262,305.00	1,128,550.00	23,520.00	2,414,375.00	2,414,375.00
คณะสาธารณสุขศาสตร์	3,840,865.00	3,195,575.00	314,160.00	7,350,600.00	7,350,600.00
คณะสิ่งแวดล้อมและทรัพยากรศาสตร์	203,835.00	213,075.00	15,840.00	432,750.00	432,750.00
โครงการจัดตั้งม.มหิดลกาญจนบุรี	944,885.00	522,105.00		1,466,990.00	1,466,990.00
โครงการจัดตั้งม.มหิดลนครสวรรค์	24,395.00	23,800.00		48,195.00	48,195.00
ธนาคาร	78,830.00	104,880.00	25,080.00	208,790.00	208,790.00
มหาวิทยาลัย	41,751,477.50	29,057,912.50	1,435,760.00	72,245,150.00	72,245,150.00
วิทยาลัยการจัดการ	798,480.00	843,840.00	136,440.00	1,778,760.00	1,778,760.00
วิทยาลัยรังสรรค์ศิลป	95,040.00	347,760.00		442,800.00	442,800.00
วิทยาลัยราชสุดา	261,210.00	227,525.00		488,735.00	488,735.00
วิทยาลัยวิทยาศาสตร์และเทคโนโลยีการค	645,120.00	600,010.00		1,245,130.00	1,245,130.00
วิทยาลัยศาสนศึกษา	565,325.00	521,325.00	155,880.00	1,242,530.00	1,242,530.00
ศูนย์สาขา	4,483,100.00	5,627,350.00		10,110,450.00	10,110,450.00
สำนักคอมพิวเตอร์	3,561,210.00	3,069,300.00		6,630,510.00	6,630,510.00
สำนักงานคนแคบต์	944,175.00	781,275.00		1,725,450.00	1,725,450.00
สำนักหอสมุด	2,895,025.00	2,496,875.00		5,391,900.00	5,391,900.00
Grand Total	130,995,560.00	104,749,660.00	4,202,660.00	239,947,880.00	239,947,880.00

Figure 5.48 : Allocation Pivot Table

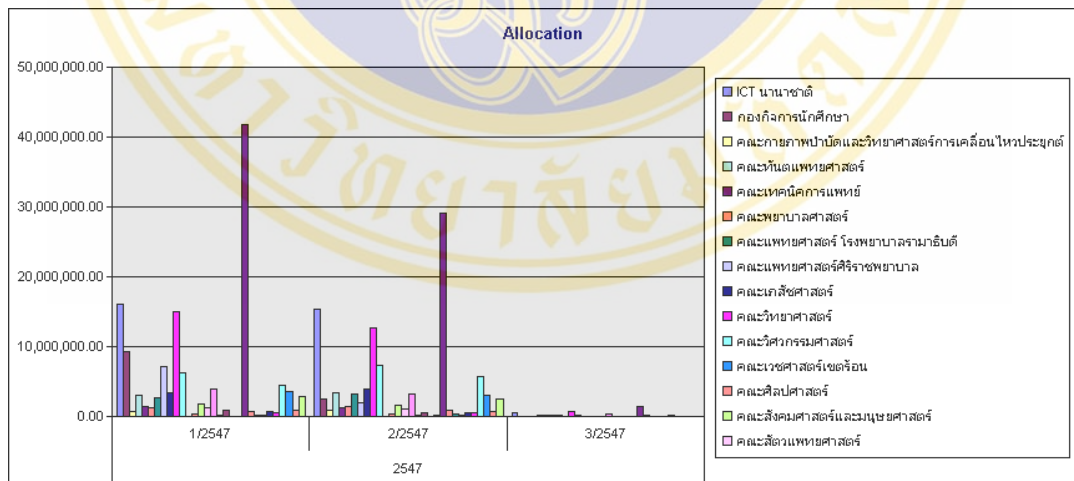


Figure 5.49 : Allocation Graph with Series in Row

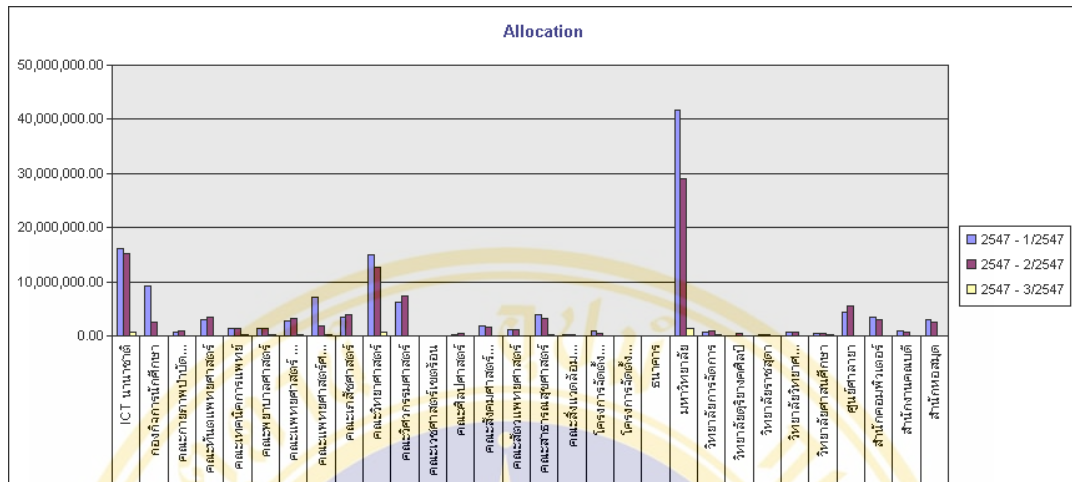


Figure 5.50 : Allocation Graph with Series in Column

Figure 5.51 shows a pivot table of Allocation. It shows an allocation amount and percentage of the amount that an institute gets in a semester of the academic year 2547.

Allocation								
Drop Filter Fields Here								
	Acad Year ▾ Semester							
	1/2547		2/2547		3/2547		Total	
Institute	Amount	Amount(%)	Amount	Amount(%)	Amount	Amount(%)	Amount	Amount(%)
มหาวิทยาลัย	41,751,477.50	31.87%	29,057,912.50	27.74%	1,435,760.00	34.16%	72,245,150.00	30.11%
คณะแพทยศาสตร์ศิริราชพยาบาล	7,196,620.00	5.49%	1,948,795.00	1.86%	155,070.00	3.69%	9,300,485.00	3.88%
คณะแพทยศาสตร์ โรงพยาบาลรามธิบดี	2,757,265.00	2.10%	3,183,165.00	3.04%	162,840.00	3.87%	6,103,270.00	2.54%
คณะวิทยาศาสตร์	15,013,175.00	11.46%	12,642,575.00	12.07%	625,200.00	14.88%	28,280,950.00	11.79%
คณะทันตแพทยศาสตร์	3,079,260.00	2.35%	3,418,675.00	3.26%	600.00	0.01%	6,498,535.00	2.71%
กองกิจการนักศึกษา	9,254,300.00	7.06%	2,553,530.00	2.44%			11,807,830.00	4.92%
สำนักหอสมุด	2,895,025.00	2.21%	2,496,875.00	2.38%			5,391,900.00	2.25%
คณะศิลปศาสตร์	332,150.00	0.25%	400,150.00	0.38%	10,800.00	0.26%	743,100.00	0.31%
สำนักงานคณบดี	944,175.00	0.72%	781,275.00	0.75%			1,725,450.00	0.72%
ศูนย์ศาลาया	4,483,100.00	3.42%	5,827,350.00	5.37%			10,110,450.00	4.21%
วิทยาลัยศาสนศึกษา	565,325.00	0.43%	521,325.00	0.50%	155,880.00	3.71%	1,242,530.00	0.52%
ธนาคาร	78,830.00	0.06%	104,880.00	0.10%	25,080.00	0.60%	208,790.00	0.09%
คณะกายภาพบำบัดและวิทยาศาสตร์การเคลื่อนไหว	767,465.00	0.59%	931,870.00	0.89%			1,699,335.00	0.71%
คณะเทคนิคการแพทย์	1,352,355.00	1.03%	1,283,360.00	1.23%	199,170.00	4.74%	2,834,885.00	1.18%
คณะพยาบาลศาสตร์	1,319,372.50	1.01%	1,356,702.50	1.30%	235,800.00	5.61%	2,911,875.00	1.21%
คณะเภสัชศาสตร์	3,369,875.00	2.57%	4,015,825.00	3.83%	480.00	0.01%	7,386,180.00	3.08%
คณะวิศวกรรมศาสตร์	6,320,145.00	4.82%	7,259,540.00	6.93%	98,520.00	2.34%	13,678,205.00	5.70%
คณะเวชศาสตร์เขตร้อน			21,600.00	0.02%			21,600.00	0.01%
คณะสังคมศาสตร์และมนุษยศาสตร์	1,757,860.00	1.34%	1,689,345.00	1.61%			3,447,205.00	1.44%
คณะสัตวแพทยศาสตร์	1,262,305.00	0.96%	1,128,550.00	1.08%	23,520.00	0.56%	2,414,375.00	1.01%
คณะสาธารณสุขศาสตร์	3,840,865.00	2.93%	3,195,575.00	3.05%	314,160.00	7.48%	7,350,600.00	3.06%
คณะสิ่งแวดล้อมและทรัพยากรศาสตร์	203,835.00	0.16%	213,075.00	0.20%	15,840.00	0.38%	432,750.00	0.18%
โครงการจัดตั้งม.มหิดลกาญจนบุรี	944,885.00	0.72%	522,105.00	0.50%			1,466,990.00	0.61%
โครงการจัดตั้งม.มหิดลนครสวรรค์	24,395.00	0.02%	23,800.00	0.02%			48,195.00	0.02%
วิทยาลัยการจัดการ	798,480.00	0.61%	843,840.00	0.81%	136,440.00	3.25%	1,778,760.00	0.74%
วิทยาลัยธุรกิจบัณฑิตย์	95,040.00	0.07%	347,760.00	0.33%			442,800.00	0.18%
วิทยาลัยราชสุดา	261,210.00	0.20%	227,525.00	0.22%			488,735.00	0.20%
วิทยาลัยวิทยาศาสตร์และเทคโนโลยีการกีฬา	645,120.00	0.49%	600,010.00	0.57%			1,245,130.00	0.52%
Grand Total	130,995,560.00	100.00%	104,749,660.00	100.00%	4,202,660.00	100.00%	239,947,880.00	100.00%

Figure 5.51 : Allocation Pivot Table with Percentage of Amount

□ Allocation by Program Cube

An Allocation by Program cube represents registration income that students of each program pay, and the income is allocated to a related faculty/institute. This cube provides an alternative view of registration allocation that makes the executive know which is a source of income program and which faculty/institute gets allocated income. This cube is composed of three measures and four dimensions listed below and shown as hierarchy in Figure 5.52.

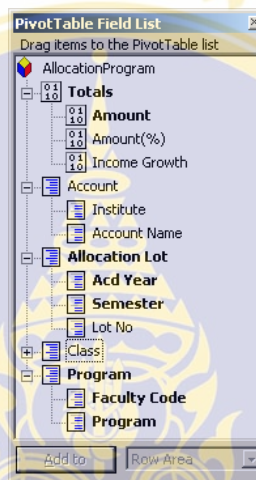


Figure 5.52 : Table Field List of Allocation by Program.

- **Amount** is a measure that shows allocated income.
- **Amount(%)** is a measure that shows percentage of allocated income.
- **Income Growth** is a measure that shows growth of allocated income.
- **Account** is a dimension that is composed of two fields, namely: Institute field and Account Name field. Institute field is a faculty or institute that gets allocated income. Account Name field is an account of a faculty or institute.
- **Allocation Lot** is a dimension that is composed of three fields, represented as a hierarchical level, namely: Acad Year, Semester and Lot No. Acad Year field represents an academic year. Semester field represents a

registration semester. Lot No field represents Lot of allocation in a semester.

- **Class** is a dimension that represents a class of students.
- **Program** is a dimension that shows a program of students. There are two levels of the dimension, namely: Faculty Code and Program. Faculty Code field represents a faculty of students and Program field represent a program under the faculty.

Figure 5.53 shows a pivot table of Allocation by Program. It shows registration income of programs of the faculty of science in the semester 1/2547 that is allocated to an institute and the faculty. Users can use multidimensional analysis functions to view appropriate information as needed. Graphic views of Allocation by Program are shown in Figure 5.54, Allocation by Program graph with series in row, and Figure 5.55 Allocation by Program graph with series in column.

Allocation Program						
Allocation Lot		Institute				
1/2547		มหาวิทยาลัย	คณะวิทยาศาสตร์	คณะสังคมศาสตร์และมนุษยศาสตร์	วิทยาลัยศรียางศิลป์	กองกิจการนักศึกษา
Faculty Code	Program	Amount	Amount	Amount	Amount	Amount
SC	วิทยาศาสตร์	1,178,630.00	854,635.00		95,040.00	278,140.00
	วิทยาศาสตร์ (การเกษตร)	272,505.00	193,560.00		19,800.00	66,740.00
	วิทยาศาสตร์ (คณิตศาสตร์)	316,360.00	311,785.00		13,320.00	108,100.00
	วิทยาศาสตร์ (คอมพิวเตอร์)	600,700.00	2,568,755.00			106,220.00
	วิทยาศาสตร์ (เคมี)	549,320.00	572,730.00		23,040.00	163,560.00
	วิทยาศาสตร์ (ชีววิทยา)	371,005.00	366,780.00		17,640.00	124,080.00
	วิทยาศาสตร์ (ชีววิทยาเชิงอนุรักษ์)	155,245.00	114,480.00		8,280.00	39,480.00
	วิทยาศาสตร์ (เทคโนโลยีการอาหาร)	209,990.00	144,360.00		16,320.00	52,640.00
	วิทยาศาสตร์ (เทคโนโลยีชีวภาพ)	443,070.00	447,565.00		18,000.00	141,940.00
	วิทยาศาสตร์ (ธรณีศาสตร์)	119,485.00	66,240.00		11,040.00	21,620.00
	วิทยาศาสตร์ (ฟิสิกส์)	41,390.00	40,005.00		1,800.00	14,100.00
	วิทยาศาสตร์ (ฟิสิกส์)	122,580.00	120,995.00		9,000.00	38,540.00
	Total	4,380,280.00	5,801,890.00		138,240.00	1,155,160.00
Grand Total		4,380,280.00	5,801,890.00		138,240.00	1,155,160.00

Figure 5.53 : Allocation by Program Pivot Table

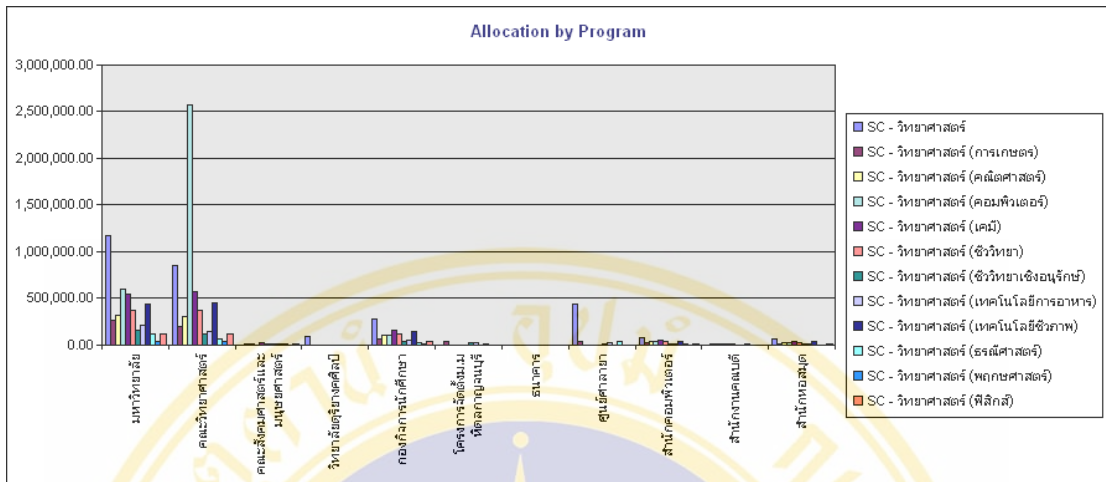


Figure 5.54 : Allocation by Program Graph with Series in Row

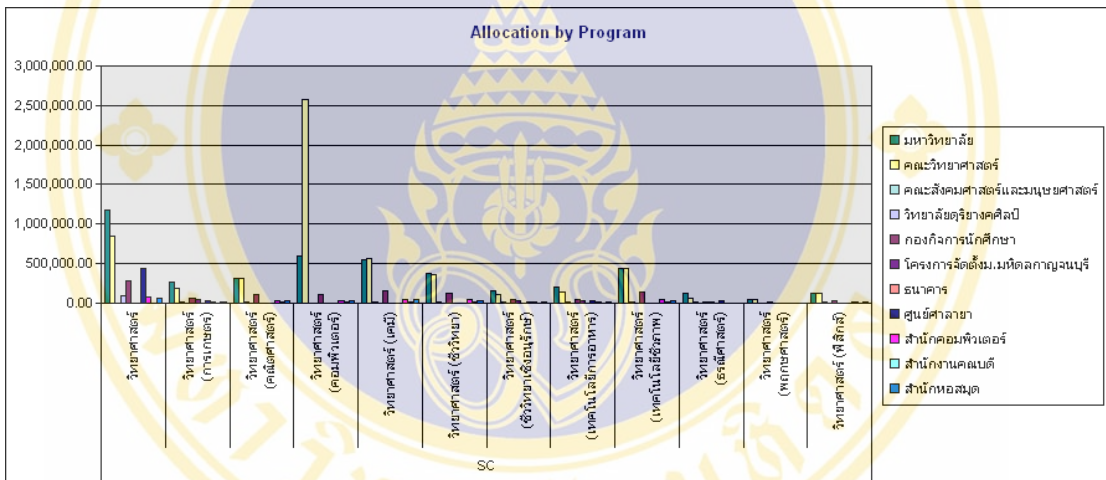


Figure 5.55 : Allocation by Program Graph with Series in Column

□ Grading Cube

A Grading cube shows the number of students that gets grade in each subject that they enroll. It is classified by program and class of students and semester. This cube also shows the percentage of students. It is composed of two measures and five dimensions listed below and shown as hierarchy in Figure 5.56.

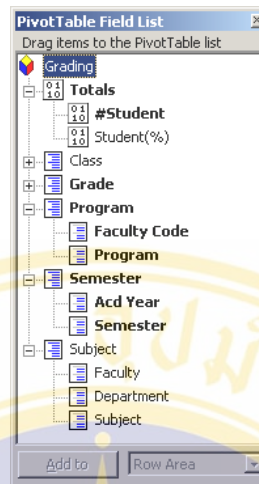


Figure 5.56 : Table Field List of Grading Cube.

- **#Student** is a measure that shows the number of enrolled students for a subject
- **Student(%)** is a measure that shows the percentage of enrolled students for a subject.
- **Class** is a dimension that represents a class of students.
- **Program** is a dimension that shows a program of students. There are two levels of the dimension, namely: Faculty Code and Program. Faculty Code field represents a faculty of students and Program field represents a program under the faculty.
- **Semester** is a dimension that students enroll a subject. There are two fields of dimension that are represented as a hierarchical level, namely: Acad Year field and Semester field. Acad Year field represents an academic year of registration and Semester field represents a semester under an academic year.
- **Subject** is a dimension. There are three levels of the dimension, namely: Faculty, Department and Subject fields. Faculty field represents a faculty. Department field represents a department that operates the subject

and Subject field represents the code and name of a subject under the faculty.

Figure 5.57 shows a pivot table of Grading. It shows the number of students that enroll in Subject SCCS155 Computer Applications in the academic year 2547, grouped by faculty and grade. Users can use multidimensional analysis functions to view appropriate information as needed. Graphic views of a Grading cube are shown in Figure 5.58, Grading graph with series in row, and Figure 5.59, Grading graph with series in column.

Grading													
Semester		Subject											
2547		SCCS155 COMPUTER APPLICATIONS											
		Grade											Grand Total
		A	B+	B	C+	C	D+	D	X	W	F	I	
Faculty Code	Program	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student
AR		1	21	49	43	16						16	146
CR				1	25	40	4	1	9	1			81
DT		25	26	16	6	3			2				78
EG		12	9	11	9								41
EN		1	1		2								4
MT		11	18	34	60	22	1				1		147
NS		3	43	110	57	9						1	223
PH		3	11	32	49	18	1						114
PI		9	20	25	34								88
PY		14	39	26	4								83
RA		5	10	23	99	11					3		151
SC		44	51	56	59	18						1	229
SI		7	10	27	44	3			19		1		111
SP				2	22	19	6	3					52
VS		13	12	2	1								28
Grand Total		148	271	414	514	159	12	4	30	1	22	1	1,576

Figure 5.57 : Grading Pivot Table

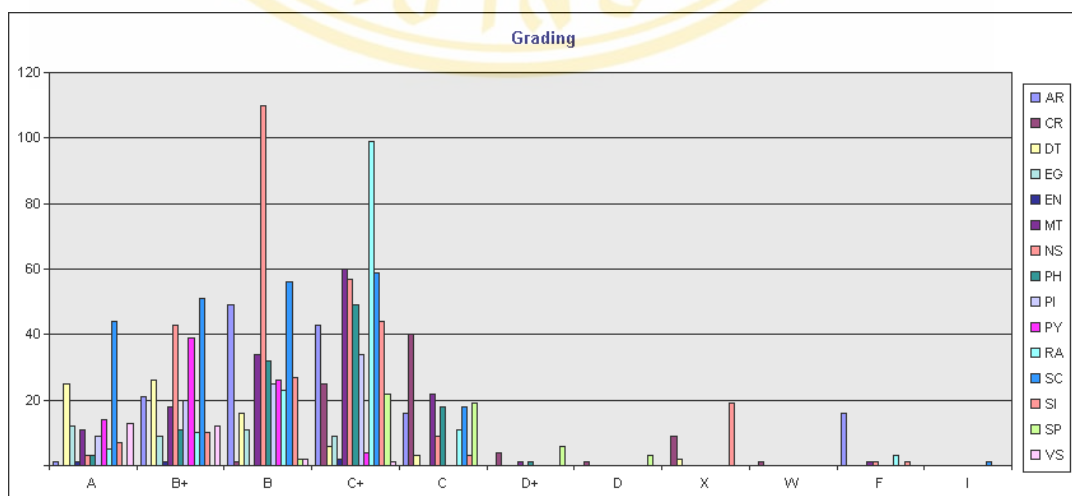


Figure 5.58 : Grading Graph with Series in Row

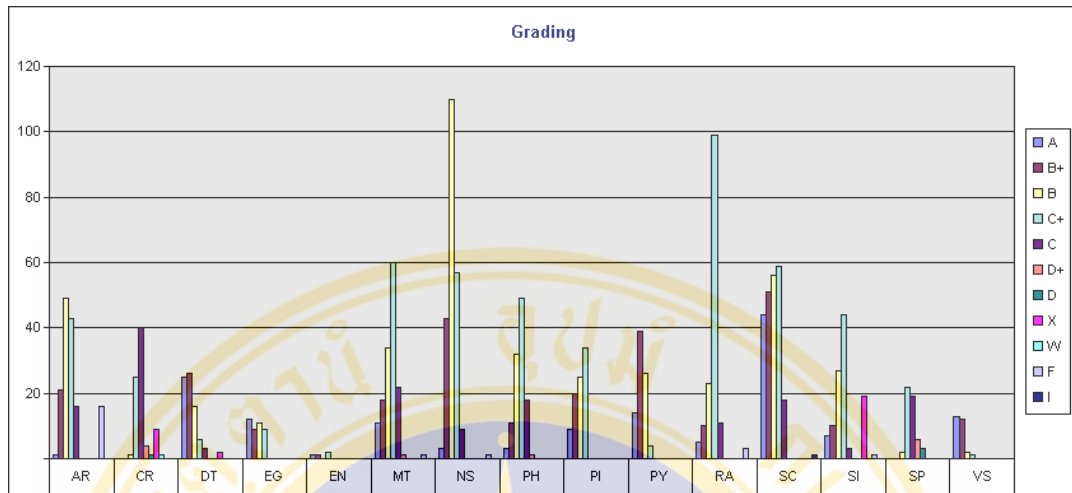


Figure 5.59 : Grading Graph with Series in Column

Figure 5.60 shows a pivot table of Grading. It shows the percentage of students that enroll in Subject SCCS155 Computer Applications in the academic year 2547, grouped by faculty and grade.

Grading		Semester 2547 Subject SCCS155 COMPUTER APPLICATIONS												Grand Total	
Faculty Code	Program	Grade											Student(%)		
		A	B+	B	C+	C	D+	D	X	W	F	I			
AR		0.06%	1.33%	3.11%	2.73%	1.02%							1.02%		9.26%
CR				0.06%	1.59%	2.54%	0.25%	0.06%	0.57%	0.06%					5.14%
DT		1.59%	1.65%	1.02%	0.38%	0.19%			0.13%						4.95%
EG		0.76%	0.57%	0.70%	0.57%										2.60%
EN		0.06%	0.06%		0.13%										0.25%
MT		0.70%	1.14%	2.16%	3.81%	1.40%	0.06%					0.06%			9.33%
NS		0.19%	2.73%	6.98%	3.62%	0.57%						0.06%			14.15%
PH		0.19%	0.70%	2.03%	3.11%	1.14%	0.06%								7.23%
PI		0.57%	1.27%	1.59%	2.16%										5.58%
PY		0.89%	2.47%	1.65%	0.25%										5.27%
RA		0.32%	0.63%	1.46%	6.28%	0.70%						0.19%			9.58%
SC		2.79%	3.24%	3.55%	3.74%	1.14%									14.53%
SI		0.44%	0.63%	1.71%	2.79%	0.19%			1.21%		0.06%				7.04%
SP				0.13%	1.40%	1.21%	0.38%	0.19%							3.30%
VS		0.82%	0.76%	0.13%	0.06%										1.78%
Grand Total		9.39%	17.20%	26.27%	32.61%	10.09%	0.76%	0.25%	1.90%	0.06%	1.40%	0.06%			100.00%

Figure 5.60 : Grading Pivot Table with Percentage of Students

□ GPA Cube

A GPA cube shows the number and percentage of students that gets a GPA range in a semester. It is classified by program and class of students. This cube provides academic efficiency of students. This cube is composed of two measures and four dimensions listed below and shown as hierarchy in Figure 5.61.

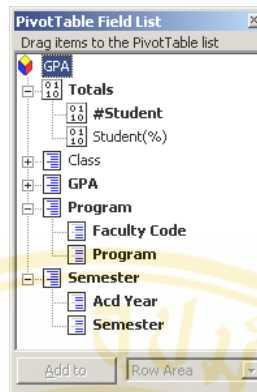


Figure 5.61 : Table Field List of GPA Cube.

- **#Student** is a measure that shows the number of students.
- **Student(%)** is a measure that shows the percentage of students for a subject.
- **Class** is a dimension that represents a class of students.
- **GPA** is a dimension that represents a GPA range.
- **Program** is a dimension that shows a program of students. There are two levels of the dimension, namely: Faculty Code and Program. Faculty Code field represents a faculty of students and Program field represents a program under the faculty.
- **Semester** is a dimension. There are two fields of dimension that are represented as a hierarchical level namely: Acd Year field and Semester field. Acd Year field represents an academic year and Semester field represents a semester under an academic year.

Figure 5.62 shows a pivot table of GPA. It shows the number of students that enrolls in an academic year 1/2547, grouped by faculty and GPA level. Users can use multidimensional analysis functions to view appropriate information as needed. Graphic views of a GPA cube are shown in Figure 5.63, GPA graph with series in row, and Figure 5.64, GPA graph with series in column.

GPA									
Semester 1/2547									
Faculty Code Program	GPAlevel								Grand Total
	2.00 - 2.24	2.25 - 2.49	2.50 - 2.74	2.75 - 2.99	3.00 - 3.24	3.25 - 3.49	3.50 - 3.74	3.75 - 4.00	
	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student
AR	8	22	20	20	21	15	9	6	146
BM	7	7	18	28	45	65	42	13	222
CR	2	5	10	23	33	61	40	27	215
DT	1		20	38	87	134	99	68	458
EG	96	189	263	216	139	68	24	8	1,058
EN	12	14	21	25	17	7	5		110
MG	9	32	46	56	93	95	60	30	432
MT	16	39	102	116	112	106	61	22	585
NS	6	20	78	182	222	190	102	20	829
PH	18	50	85	117	160	141	79	42	712
PI	10	37	70	103	104	80	63	14	488
PY	6	17	44	88	99	106	75	24	472
RA	7	35	109	222	250	281	218	118	1,276
RS	7	25	25	26	11	3			102
SC	103	160	275	270	242	184	101	77	1,615
SH				3	13	28	10	9	64
SI	3	18	86	172	297	354	351	274	1,688
SP	23	49	49	50	30	22	14	9	280
VS	3	1	16	17	49	32	26	11	163
Grand Total	323	710	1,337	1,772	2,039	1,980	1,382	772	10,895

Figure : 5.62 GPA Pivot Table

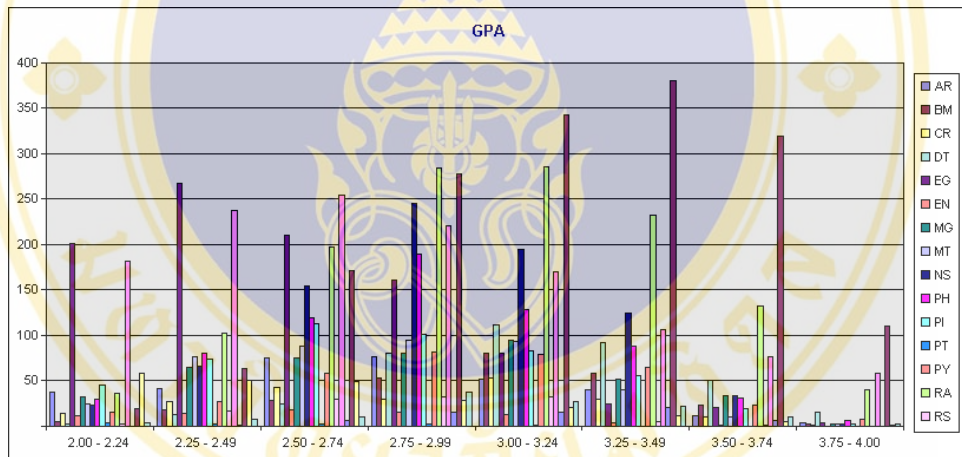


Figure 5.63 : GPA Graph with Series in Row

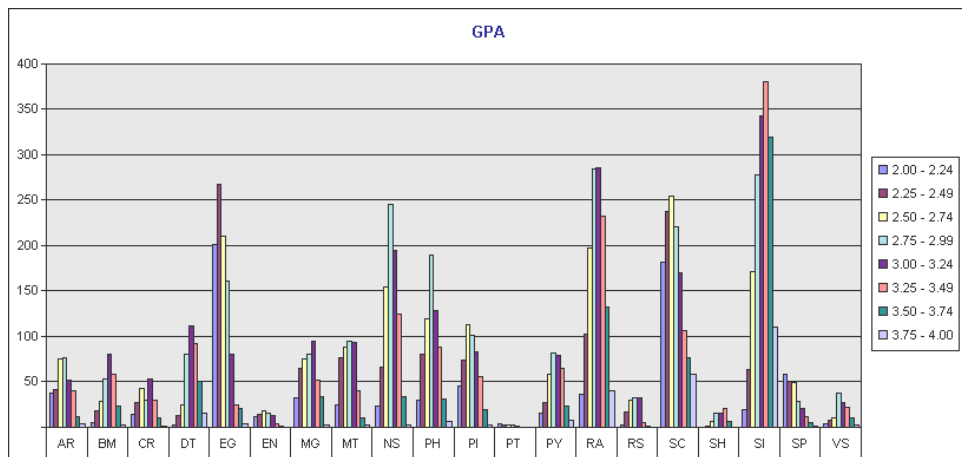


Figure 5.64 : GPA Graph with Series in Column

Figure 5.65 shows a pivot table of GPA. It shows the percentage of students that enrolls in the academic year 1/2547, grouped by faculty and GPA level.

GPA											
Semester		GPAlevel									Grand Total
1/2547		1.99	2.00 - 2.24	2.25 - 2.49	2.50 - 2.74	2.75 - 2.99	3.00 - 3.24	3.25 - 3.49	3.50 - 3.74	3.75 - 4.00	
Faculty Code	Program	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	
AR		0.07%	0.20%	0.18%	0.18%	0.19%	0.14%	0.08%	0.06%	1.34%	
BM			0.06%	0.17%	0.26%	0.41%	0.60%	0.39%	0.12%	2.04%	
CR		0.02%	0.05%	0.09%	0.21%	0.30%	0.56%	0.37%	0.25%	1.97%	
DT		0.01%		0.18%	0.35%	0.80%	1.23%	0.91%	0.62%	4.20%	
EG		0.88%	1.73%	2.41%	1.98%	1.28%	0.62%	0.22%	0.07%	9.71%	
EN		0.11%	0.13%	0.19%	0.23%	0.16%	0.06%	0.05%		1.01%	
MG		0.08%	0.29%	0.42%	0.51%	0.85%	0.87%	0.55%	0.28%	3.97%	
MT		0.15%	0.36%	0.94%	1.06%	1.03%	0.97%	0.56%	0.20%	5.37%	
NS		0.06%	0.18%	0.72%	1.67%	2.04%	1.74%	0.94%	0.18%	7.61%	
PH		0.17%	0.46%	0.78%	1.07%	1.47%	1.29%	0.73%	0.39%	6.54%	
PI		0.09%	0.34%	0.64%	0.95%	0.95%	0.73%	0.58%	0.13%	4.48%	
PY		0.06%	0.16%	0.40%	0.81%	0.91%	0.97%	0.69%	0.22%	4.33%	
RA		0.06%	0.32%	1.00%	2.04%	2.29%	2.58%	2.00%	1.08%	11.71%	
RS		0.06%	0.23%	0.23%	0.24%	0.24%	0.10%	0.03%		0.94%	
SC		0.95%	1.47%	2.52%	2.48%	2.22%	1.69%	0.93%	0.71%	14.82%	
SH					0.03%	0.12%	0.26%	0.09%	0.08%	0.59%	
SI		0.03%	0.17%	0.79%	1.58%	2.73%	3.25%	3.22%	2.51%	15.31%	
SP		0.21%	0.52%	0.45%	0.46%	0.28%	0.20%	0.13%	0.08%	2.57%	
YS		0.03%	0.01%	0.15%	0.16%	0.45%	0.29%	0.24%	0.10%	1.50%	
Grand Total		2.96%	6.52%	12.27%	16.26%	18.72%	18.17%	12.68%	7.09%	100.00%	

Figure : 5.65 GPA Pivot Table with Percentage of Students

5.2 Testing Scenarios

We use MU-SDM to provide information for university executives. We define test cases to evaluate the system using the following scenarios:

- Use MU-SDM to compare the number of students of the university.
- The executive would like to know the productivity of the university in the past six years.
- The executive would like to know registration income for the first semester of the academic year 2548.
- Use MU-SDM to approximate registration income.
- MU-SDM provides tuition fee information.
- MU-SDM provides income information of faculties and institutes from registration.
- The executive would like to know a source of allocated income of the faculty of science.
- The executive wants to know income growth of faculties and institutes.
- MU-SDM provides information of distribution of grade.

- MU-SDM provides information about the efficiency of students.

Details of scenarios are as the following:

Scenario 1 : Use MU-SDM to compare the number of students of the university.

Situation : The executive wants to know the number of admitted students and graduated students in each academic year.

Step

1. Select Student menu.
2. Add Academic Year dimension to Row Area.
3. Add Gender dimension to Column Area.
4. Add Admit and Graduate field under Totals measure to data area, and the Student Pivot Table will be displayed in Figure 5.66.

Student Information						
Drop Filter Fields Here						
Academic Year	Female		Male		Grand Total	
	Admit	Graduate	Admit	Graduate	Admit	Graduate
2533	698	4	435		1,133	4
2534	959	3	523	22	1,482	25
2535	1,160	204	608	161	1,768	365
2536	1,326	694	758	352	2,084	1,046
2537	1,555	813	825	353	2,380	1,166
2538	1,696	1,064	933	493	2,629	1,557
2539	1,816	1,197	984	651	2,800	1,848
2540	1,773	1,239	1,021	659	2,794	1,898
2541	1,569	1,416	1,005	830	2,574	2,246
2542	1,576	1,462	1,079	795	2,655	2,257
2543	1,621	1,480	1,084	801	2,705	2,281
2544	1,517	1,401	897	804	2,414	2,205
2545	1,608	1,275	992	768	2,600	2,043
2546	1,912	1,410	1,245	790	3,157	2,200
2547	2,401	1,298	1,356	817	3,757	2,115
2548	2,593	1,226	1,417	614	4,010	1,840
Grand Total	25,780	16,186	15,162	8,910	40,942	25,096

Figure 5.66 : Student Information Pivot Table.

Figure 5.66 shows the number of admitted students and graduated students in each academic year. Academic Year 2533 have 1,133 new students and there are 4 students graduating from a one-year program this year.

The executive can drill-down to program level through the following steps.

5. Add Program dimension to Row Area, after Academic Year field.
6. Expand a particular academic year to view the faculty and program level, and the Pivot Table will be shown in Figure 5.67.

Student Information						
Drop Filter Fields Here						
Gender						
Female						
Male						
Grand Total						
Academic Year	Faculty Code	Program	Admit	Graduate	Admit	Graduate
2533			698	4	435	1,133
2534			959	3	523	1,482
2535			1,160	204	608	1,768
2536			1,326	694	758	2,084
2537			1,555	813	825	2,380
2538			1,696	1,064	933	2,629
2539			1,816	1,197	984	2,800
2540			1,773	1,239	1,021	2,794
2541			1,569	1,416	1,005	2,574
2542			1,576	1,462	1,079	2,655
2543	BM		16	18	15	31
	CR		33	1	26	59
	DT		65	54	27	92
	EG	วิศวกรรมศาสตร์	2		21	23
		วิศวกรรมศาสตร์ (คอมพิวเตอร์)	7	4	30	37
		วิศวกรรมศาสตร์ (เคมี)	14	11	22	36
		วิศวกรรมศาสตร์ (เครื่องกล)	2	5	34	36
		วิศวกรรมศาสตร์ (ไฟฟ้า)	4	16	32	36
		วิศวกรรมศาสตร์ (โยธา)	4	4	27	31
		วิศวกรรมศาสตร์ (อุตสาหกรรม)	15	8	21	36
		Total	48	48	187	200
	EN		24	15	7	31
	MS		17		39	56
	MT		133	123	75	208
	NS		318	411	46	664
	PH		126	111	86	212
	PI		44		36	80
	PY		87	80	50	137
	RA		194	227	90	391
	RS		58		38	96
	SC		235	172	161	396
	SH		13	15	9	22
	SI		169	186	149	318
	SP		22	19	33	55
	YS		19		10	29
	Total		1,621	1,480	1,084	2,705
2544			1,517	1,401	897	2,414
2545			1,608	1,275	992	2,600
2546			1,912	1,410	1,245	3,157
2547			2,401	1,298	1,356	3,757
2548			2,593	1,226	1,417	4,010
Grand Total			25,780	16,186	15,162	40,942

Figure 5.67 : Student Information Pivot Table at Program Level.

Scenario 2 : The executive would like to know the productivity of the university in the past six years.

Situation : The executive wants to know the number of graduated students in each year that were admitted in 2542 in order to compare number of graduated and non-graduate students in each program. This information represents the productivity of the university.

Step

1. The executive selects Student Graduation menu. The Student Graduation Cube will be shown on the right screen.
2. Drag Admit Year field to Filter Area and check on admit year 2542.
3. Drag Program dimension to Row Area, and the faculty code and program field under program dimension will appear in Row Area.

4. Add Status and Grad Year field under Status dimension to Column Area.

Student Graduation Pivot Table will be shown in Figure 5.68.

Student Graduation				
Admit Year 2542				
Faculty Code	Program	Status		Grand Total
		Graduated	Not-Graduate	
		#Student	#Student	#Student
BM		30	1	31
CR		28	21	49
DT		68	6	74
EG		179	43	222
EN		12	6	18
MS		22	33	55
MT		165	42	207
NS		355	43	398
PH		226	43	269
PI		77	5	82
PY		103	32	135
RA		284	17	301
SC		312	60	372
SH		22	3	25
SI		308	13	321
SP		58	6	64
VS		28	4	32
Grand Total		2,277	378	2,655

Figure 5.68 Student Graduation Pivot Table.

5. The executive can drill-down to program level by clicking plus sign(+) of the faculty or view detail of graduate students by clicking plus sign of grad status, and the cube will be shown in Figure 5.69.

Student Graduation										
Admit Year 2542										
Faculty Code	Program	Status						Total	Not-Graduate	Grand Total
		Graduated								
		2543	2544	2545	2546	2547	2548			
		#Student	#Student	#Student	#Student	#Student	#Student	#Student	#Student	
BM						29	1	30	1	31
CR				10	11	6	1	28	21	49
DT						66	2	68	6	74
EG	วิศวกรรมศาสตร์								25	25
	วิศวกรรมศาสตร์ (คอมพิวเตอร์)			28	3	1		32	2	34
	วิศวกรรมศาสตร์ (เคมี)			21	6	1	2	30	3	33
	วิศวกรรมศาสตร์ (เครื่องกล)			35				35		35
	วิศวกรรมศาสตร์ (ไฟฟ้า)			35				35	2	37
	วิศวกรรมศาสตร์ (โยธา)			9	8	4	2	23	5	28
	วิศวกรรมศาสตร์ (อุตสาหกรรม)			1	15	8		24	6	30
	Total			129	32	14	4	179	43	222
EN				12				12	6	18
MS			1	6	12	3		22	33	55
MT		29		136				165	42	207
NS		71		275	9			355	43	398
PH			96	126	4			226	43	269
PI						72	5	77	5	82
PY				10	92	1		103	32	135
RA				137	4	140	3	284	17	301
SC	วิทยาศาสตร์								36	36
	วิทยาศาสตร์ (คณิตศาสตร์)			50	1			51	2	53
	วิทยาศาสตร์ (คอมพิวเตอร์)	44	4	1				49	7	56
	วิทยาศาสตร์ (เคมี)			60				60		60
	วิทยาศาสตร์ (ชีววิทยา)			59	2			61	2	63
	วิทยาศาสตร์ (เทคโนโลยีชีวภาพ)			49				49	1	50
	วิทยาศาสตร์ (เภสัชศาสตร์)			25				25	7	32
	วิทยาศาสตร์ (ฟิสิกส์)			16	1			17	5	22
	Total	44	4	260	4			312	60	372
SH		22						22	3	25
SI		37	1	49	1	213	7	308	13	321
SP		9	1	42	5	1		58	6	64
VS						28		28	4	32
Grand Total		212	103	1,192	174	573	23	2,277	378	2,655

Figure 5.69 : Drill-down Student Graduation Pivot Table.

6. Remove #Student field from data area and add Student(%) field to data area to view the percentage of graduated and not-graduate students. The Student Graduation Pivot Table will be shown in Figure 5.70

Student Graduation											
Admit Year		Status ▾ Grad Year							Not-Graduate	Grand Total	
2542		Graduated									
		2543	2544	2545	2546	2547	2548	Total			
Faculty Code ▾ Program		Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	Student(%)	#Student	#Student	
BM				0.38%	0.41%		1.09%	0.04%	1.13%	0.04%	1.17%
CR							0.23%	0.04%	1.05%	0.79%	1.85%
DT							2.49%	0.08%	2.56%	0.23%	2.79%
EG										0.94%	0.94%
วิศวกรรมศาสตร์										0.08%	1.26%
วิศวกรรมศาสตร์ (คอมพิวเตอร์)					1.05%	0.11%	0.04%		1.21%	0.08%	1.24%
วิศวกรรมศาสตร์ (เคมี)					0.79%	0.23%	0.04%	0.08%	1.13%	0.11%	1.24%
วิศวกรรมศาสตร์ (เครื่องกล)					1.32%				1.32%		1.32%
วิศวกรรมศาสตร์ (ไฟฟ้า)					1.32%				1.32%	0.08%	1.39%
วิศวกรรมศาสตร์ (โยธา)					0.34%	0.30%	0.15%	0.08%	0.87%	0.19%	1.05%
วิศวกรรมศาสตร์ (อุตสาหกรรม)					0.04%	0.56%	0.30%		0.90%	0.23%	1.13%
Total					4.86%	1.21%	0.53%	0.15%	6.74%	1.62%	8.36%
EN					0.45%				0.45%	0.23%	0.68%
MS			0.04%		0.23%	0.45%	0.11%		0.83%	1.24%	2.07%
MT		1.09%			5.12%				6.21%	1.58%	7.80%
NS		2.67%			10.36%	0.34%			13.37%	1.62%	14.99%
PH			3.62%		4.75%	0.15%			8.51%	1.62%	10.13%
PI							2.71%	0.19%	2.90%	0.19%	3.09%
PY					0.38%	3.47%	0.04%		3.88%	1.21%	5.08%
RA					5.16%	0.15%	5.27%	0.11%	10.70%	0.64%	11.34%
SC										1.36%	1.36%
วิทยาศาสตร์									1.92%	0.08%	2.00%
วิทยาศาสตร์ (คณิตศาสตร์)					1.88%	0.04%			1.92%	0.08%	2.00%
วิทยาศาสตร์ (คอมพิวเตอร์)		1.66%	0.15%		0.04%				1.85%	0.26%	2.11%
วิทยาศาสตร์ (เคมี)					2.26%				2.26%		2.26%
วิทยาศาสตร์ (ชีววิทยา)					2.22%	0.08%			2.30%	0.08%	2.37%
วิทยาศาสตร์ (เทคโนโลยีชีวภาพ)					1.85%				1.85%	0.04%	1.88%
วิทยาศาสตร์ (พฤกษศาสตร์)					0.94%				0.94%	0.26%	1.21%
วิทยาศาสตร์ (ฟิสิกส์)					0.60%	0.04%			0.64%	0.19%	0.83%
Total		1.66%	0.15%		9.79%	0.15%			11.75%	2.26%	14.01%
SH									0.83%	0.11%	0.94%
SI					1.39%	0.04%	8.02%	0.26%	11.60%	0.49%	12.09%
SP					0.34%	0.04%	1.58%	0.19%	2.18%	0.23%	2.41%
VS									1.05%	0.15%	1.21%
Grand Total		7.98%	3.88%	44.90%	6.55%	21.58%	0.87%	85.76%	14.24%	100.00%	

Figure 5.70 : Student Graduation Pivot Table with Percentage of Students.

Scenario 3 : The executive would like to know registration income for the first semester of the academic year 2548.

Situation : The executive wants to know how much registration income the university gets for the semester 1/2548. And he/she would like to know the type and period of payment.

Step

1. The executive selects Registration menu. The Registration Pivot Table will be shown on the right screen.
2. Drag Semester field from Table Field List to Filter Area.

3. Click drop-down button of Semester dimension to show all semesters under the dimension and check 1/2548 semester. Filter on Semester dimension will be displayed in Figure 5.71.

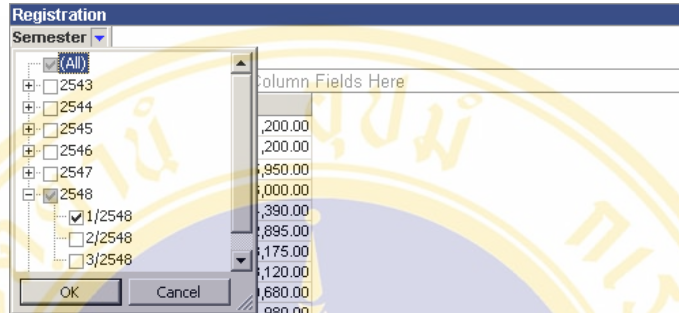


Figure 5.71 : Filter on Semester Dimension.

4. Drag Payment Date dimension from Table Field List to Row Area to display all levels of Payment Date (Year, Month, Day).
5. Drag Payment Type dimension from Table Field List to Column Area.
6. Expand Year dimension in order to drill-down to month dimension, and Registration Pivot Table will be shown in Figure 5.72. Additionally, he/she can drill-down to day level of payment date by expanding month dimension.

Registration									
Semester									
1/2548									
		Payment Type							
		E-Payment	เงินกู้	เงินสด	เช็คของบัญชี	เช็คธนาคาร	เช็คส่วนตัว	ตัวแลกเงินธนาคาร	Grand Total
Year	Month	Amount	Amount	Amount	Amount	Amount	Amount	Amount	Amount
2004	December					4,999,600.00		1,171,600.00	6,171,200.00
	Total					4,999,600.00		1,171,600.00	6,171,200.00
2005	January					25,950.00			25,950.00
	February					1,600,800.00		55,200.00	1,656,000.00
	March	1,929,140.00			575,250.00				2,504,390.00
	April	14,783,240.00	29,795.00			59,860.00			14,872,895.00
	May	20,103,735.00	230,525.00			6,031,315.00	55,200.00	337,400.00	26,758,175.00
	June	41,079,135.00	6,994,495.00	15,100.00		34,454,390.00			82,543,120.00
	July	2,484,940.00	1,427,785.00			266,955.00			4,179,680.00
	August	1,175,210.00	31,195.00			5,168,575.00			6,374,980.00
	September	1,960,235.00	22,095.00			1,030,575.00			3,012,905.00
	October	403,995.00	9,565.00			615.00			414,175.00
	November	103,780.00				12,830.00			116,610.00
	December	12,945.00							12,945.00
	Total	84,036,355.00	8,745,455.00	15,100.00	575,250.00	48,651,865.00	55,200.00	392,600.00	142,471,825.00
2006	January	14,385.00				44,815.00			59,200.00
	February	35,940.00				2,530.00			38,470.00
	March	89,860.00				1,590,135.00			1,679,995.00
	April	1,830.00				20,815.00			22,445.00
	Total	142,015.00				1,658,095.00			1,800,110.00
Grand Total		84,178,370.00	8,745,455.00	15,100.00	575,250.00	55,309,560.00	55,200.00	1,564,200.00	150,443,135.00

Figure 5.72 : Registration Pivot Table of 1/2548 Semester.

Scenario 4 : Use MU-SDM to approximate registration income.

Situation : The executive wants to know approximated registration income of the academic year 2548 from registration income of last year. Registration income of the academic year 2547 provides a trend of income for the next year including the distribution of income.

Step

1. Select Registration menu.
2. Drag Year and Month field under Payment Date dimension from Table Field List to Row Area
3. Drag Semester dimension from Table Field List to Column Area, Acd Year (Academic Year) and Semester field will appear.
4. Click drop-down of Acd Year field and check on 2547 and 2548 to display registration income only in the academic years 2547 and 2548.
5. Drill-down Acd Year field to show a detail of registration income in the semester level. The registration income will be shown in Figure 5.73

Registration									
Drop Filter Fields Here									
		Acad Year ▾ Semester							
		2547				2548			
		1/2547	2/2547	3/2547	Total	1/2548	2/2548	3/2548	Total
Year ▾	Month	Inv Amount	Inv Amount	Inv Amount	Inv Amount	Inv Amount	Inv Amount	Inv Amount	Inv Amount
2003	December	6,234,700.00			6,234,700.00				
2004	February	4,821,000.00			4,821,000.00				
	March	1,576,690.00			1,576,690.00				
	April	13,938,725.00			13,938,725.00				
	May	30,755,780.00			30,755,780.00				
	June	65,013,660.00			65,013,660.00				
	July	4,132,570.00	517,485.00		4,650,055.00				
	August	2,956,175.00	3,133,565.00		6,089,740.00				
	September	2,019,890.00	4,134,295.00		6,154,185.00				
	October	154,450.00	30,435,390.00		30,589,840.00				
	November	556,065.00	57,623,750.00		58,179,815.00				
	December	10,795.00	4,817,460.00		4,828,255.00	6,171,200.00			6,171,200.00
2005	January	47,325.00	1,408,540.00		1,455,865.00		25,950.00		25,950.00
	February	172,330.00	1,245,190.00	319,470.00	1,736,990.00	1,656,000.00			1,656,000.00
	March	69,405.00	1,065,230.00	1,129,345.00	2,263,980.00	2,504,390.00			2,504,390.00
	April	40,445.00	124,745.00	2,055,080.00	2,220,270.00	14,872,895.00			14,872,895.00
	May		33,250.00	407,475.00	440,725.00	26,758,175.00			26,758,175.00
	June	14,480.00	17,625.00	20,170.00	52,275.00	82,543,120.00			82,543,120.00
	July	24,030.00	3,460.00	207,325.00	234,815.00	4,179,680.00	243,305.00		4,422,985.00
	August	1,215.00	131,080.00	3,645.00	135,940.00	6,374,980.00	3,798,195.00		10,173,175.00
	September	20,015.00	53,210.00	31,205.00	104,430.00	3,012,905.00	3,728,325.00		6,741,230.00
	October		5,445.00	1,815.00	7,260.00	414,175.00	31,790,055.00		32,204,230.00
	November		52,775.00	14,360.00	67,135.00	116,610.00	70,683,485.00		70,800,095.00
	December	415.00	1,415.00	368,835.00	370,665.00	12,945.00	5,172,000.00		5,184,945.00
2006	January		2,430.00	32,375.00	34,805.00	59,200.00	1,654,735.00		1,713,935.00
	February		1,215.00		1,215.00	38,470.00	1,524,205.00	374,800.00	1,937,475.00
	March					1,650,435.00	462,165.00	182,470.00	2,295,070.00
Grand Total		132,560,160.00	104,807,555.00	4,591,100.00	241,958,815.00	150,391,130.00	119,056,470.00	557,270.00	270,004,870.00

Figure 5.73 : Comparison of Registration Income of Years 2547 and 2548.

The executive can approximate registration income of the 2548 academic year that will be over two hundred and seventy millions and the income will be completed in January 2007.

Scenario 5 : MU-SDM provides tuition fee information.

Situation : The executive wants to know how much income the university gets from its tuition fee in the 2547 academic year.

Step

1. Select Subject menu.
2. Drag Semester dimension from Table Field List to Filter Area and check on the academic year 2547.
3. Drag Faculty field under Subject dimension to Row Area, the Registration Subject Pivot Table will be displayed in Figure 5.74.

Subject	
Semester	Drop Column
2547	
Faculty	Amount
BANGKOK METROPOLITAN MEDICAL COLLEGE	0.00
COLLEGE OF MANAGEMENT	2,966,400.00
COLLEGE OF MUSIC	738,000.00
COLLEGE OF RELIGIOUS STUDIES	1,663,200.00
COLLEGE OF SPORTS SCIENCE AND TECHNOLOGY	1,648,800.00
FACULTY OF ARTS	965,400.00
FACULTY OF DENTISTRY	3,235,600.00
FACULTY OF ENGINEERING	6,657,000.00
FACULTY OF ENVIRONMENT AND RESOURCE STUDIES	560,600.00
FACULTY OF MEDICAL TECHNOLOGY	2,587,200.00
FACULTY OF MEDICINE RAMATHIBODI HOSPITAL	8,221,600.00
FACULTY OF MEDICINE SIRIRAJ HOSPITAL	12,790,200.00
FACULTY OF NURSING	3,641,200.00
FACULTY OF PHARMACY	4,580,500.00
FACULTY OF PUBLIC HEALTH	6,806,800.00
FACULTY OF SCIENCE	45,791,600.00
FACULTY OF SOCIAL SCIENCES AND HUMANITIES	5,653,200.00
FACULTY OF TROPICAL MEDICINE	36,000.00
FACULTY OF VETERINARY SCIENCE	1,391,800.00
PRABOROMARAJCHANOK INSTITUTE	42,600.00
RATCHASUDA COLLEGE	619,200.00
Grand Total	110,596,900.00

Figure 5.74 : Registration Subject Pivot Table

The executive can drill-down to detail of the Faculty of Science, which has highest tuition, to see tuition and the number of enrolled students in the department level.

4. Remove Faculty field from Row Area and add Department filed under Subject dimension to Row Area and check on the Faculty of Science only.

5. Add Registration Type dimension to Column Area
6. Add #Student field from Table Field List and drop after Amount field, the cube will be displayed in Figure 5.75.

Subject		Registration Type					Grand Total
Semester		Normal	Add	Drop	Withdraw	Grand Total	
2547		Amount	Amount	Amount	Amount	Amount	
Department							
AGRICULTURAL SCIENCE		200,600.00	26,200.00		3,200.00	230,000.00	
DEPT. OF ANATOMY		552,800.00	11,200.00	1,600.00	8,800.00	574,400.00	
DEPT. OF BIOCHEMISTRY		789,000.00	2,800.00	600.00		792,400.00	
DEPT. OF BIOLOGY		4,290,400.00	46,800.00	623,200.00	6,600.00	4,967,000.00	
DEPT. OF BIOTECHNOLOGY		963,800.00	30,800.00	16,200.00	2,000.00	1,012,800.00	
DEPT. OF CHEMISTRY		6,543,000.00	273,600.00	32,200.00	23,000.00	6,871,800.00	
DEPT. OF COMPUTER SCIENCE		6,780,000.00	456,600.00	9,000.00	95,400.00	7,341,000.00	
DEPT. OF FOREIGN LANGUAGES		5,224,000.00	1,222,400.00	220,400.00	4,800.00	6,671,600.00	
DEPT. OF MATHEMATICS		5,821,200.00	202,800.00	32,200.00	218,800.00	6,275,000.00	
DEPT. OF MICROBIOLOGY		522,400.00	0.00	3,200.00		525,600.00	
DEPT. OF PATHOBIOLOGY		83,200.00	88,600.00			171,800.00	
DEPT. OF PHARMACOLOGY		218,800.00	30,000.00		600.00	249,400.00	
DEPT. OF PHYSICS		5,947,400.00	209,400.00	39,600.00	108,800.00	6,305,200.00	
DEPT. OF PHYSIOLOGY		492,000.00				492,000.00	
INTERDISCIPLINARY COURSES		3,132,000.00	7,600.00	1,600.00	20,400.00	3,161,600.00	
PROGRAM : GEOSCIENCE		18,000.00				18,000.00	
PROGRAM : PLANT SCIENCE		94,600.00	30,600.00	6,000.00	2,000.00	133,200.00	
Grand Total		41,673,200.00	2,639,400.00	985,800.00	494,400.00	45,792,800.00	

Figure 5.75 : Registration Subject Pivot Table of Faculty of Science.

7. Drill-down to the subject level of the department of computer science by clicking plus sign(+) of DEPT. OF COMPUTER SCIENCE, and the pivot table will be shown in Figure 5.76.

Subject		Registration Type					Grand Total
Semester		Normal	Add	Drop	Withdraw	Grand Total	
2547		Amount	Amount	Amount	Amount	Amount	
Department	Subject						
AGRICULTURAL SCIENCE		200,600.00	26,200.00		3,200.00	230,000.00	
DEPT. OF ANATOMY		552,800.00	11,200.00	1,600.00	8,800.00	574,400.00	
DEPT. OF BIOCHEMISTRY		789,000.00	2,800.00	600.00		792,400.00	
DEPT. OF BIOLOGY		4,290,400.00	46,800.00	623,200.00	6,600.00	4,967,000.00	
DEPT. OF BIOTECHNOLOGY		963,800.00	30,800.00	16,200.00	2,000.00	1,012,800.00	
DEPT. OF CHEMISTRY		6,543,000.00	273,600.00	32,200.00	23,000.00	6,871,800.00	
DEPT. OF COMPUTER SCIENCE	SCCS155 COMPUTER APPLICATIONS	575,400.00	82,200.00	3,000.00	600.00	661,200.00	
	SCCS155 MICROCOMPUTER APPLICATIONS	288,000.00				288,000.00	
	SCCS200 FUNDAMENTALS OF PROGRAMMING	951,000.00	6,000.00		3,000.00	960,000.00	
	SCCS209 ADVANCED PROGRAMMING	966,000.00	99,000.00		27,000.00	1,092,000.00	
	SCCS211 INTRODUCT TO DIGITAL SYSTEMS	528,000.00	102,000.00		30,000.00	660,000.00	
	SCCS221 INTRODUCT TO COM ARCHITECTURE	324,000.00				324,000.00	
	SCCS301 TECHNICAL ENGLISH I	337,800.00	9,600.00		3,600.00	351,000.00	
	SCCS302 TECHNICAL ENGLISH II	33,600.00				33,600.00	
	SCCS315 INTRO TO COMPUTER SYSTEMS	34,800.00				34,800.00	
	SCCS316 INTRO TO COMPUTER ORGANIZATION	34,800.00				34,800.00	
	SCCS319 INTERNET TECHNOLOGIES & APPLIC	30,600.00				30,600.00	
	SCCS320 DISCRETE STRUCTURE	586,800.00	105,600.00		3,600.00	696,000.00	
	SCCS321 DATA STRUCT&ALGORITHM ANALY	358,800.00				358,800.00	
	SCCS323 COMPUTER DATA COMMUNICATION	309,000.00	9,000.00		3,000.00	321,000.00	
	SCCS341 OPERAT SYS & COMP ARCHITECT I	33,600.00			600.00	34,200.00	
	SCCS343 PRINCIPLES OF OPERAT SYSTEMS	288,000.00	9,000.00	3,000.00	15,000.00	315,000.00	
	SCCS363 INFORM SYSTEMS IN ORGANIZATION	303,000.00	9,000.00		9,000.00	321,000.00	
	SCCS375 ADV. MATH I FOR COMP. SC.	33,600.00				33,600.00	
	SCCS376 ADV. MATH II FOR COMP. SC.	28,800.00	2,400.00			31,200.00	
	SCCS377 ADVANCED MATH III FOR COMP. SC	30,600.00				30,600.00	
	SCCS411 DATABASE MANAGEMENT SYSTEMS	334,800.00	10,800.00	3,000.00		348,600.00	
	SCCS414 INFOR.STORAGE & RETRIEVAL	30,600.00				30,600.00	
	SCCS415 COMPUTER DATA COMMUNICATION	33,600.00				33,600.00	
	SCCS417 INFORMATION SYST.IN ORGANZTION.	34,800.00				34,800.00	
	SCCS418 INFOR.SYSTEM ANALY. & DESIGN	28,800.00	2,400.00			31,200.00	
	SCCS421 COMP.NETWORKS & DISTRI.PROCES.	28,800.00	2,400.00			31,200.00	
	SCCS422 LOCAL AREA NETWORKS	30,600.00				30,600.00	
	SCCS431 SOFTWARE DESIGN & DEVELOPMENT	28,800.00	2,400.00			31,200.00	

Figure 5.76 : Registration Subject Pivot Table of Computer Science Department.

The executive can drill-down to a detail of the subject that he/she is interested in, such as SCCS155 via the following steps.

8. Check on 1/2547 under Semester dimension at Filter Area.
9. Move Subject field to Filter Area and check on SCCS155.
10. Add Program dimension to Row Area, and the Faculty Code and Program field will be displayed on row header.
11. Expand all Faculty Codes to drill-down to the Program level under the faculty, the Registration Subject Pivot Table will be displayed in Figure 5.77.

Subject		Registration Type		Grand Total	
Semester	Subject	Normal		Amount	#Student
1/2547	SCCS155 COMPUTER APPLICATIONS				
Faculty Code	Program	Amount	#Student	Amount	#Student
AR	ศิลปศาสตร์ (ภาษาไทย)	48,600.00	81	48,600.00	81
	ศิลปศาสตร์ (ภาษาอังกฤษ)	39,000.00	65	39,000.00	65
	Total	87,600.00	146	87,600.00	146
NS	พยาบาลศาสตร์	133,800.00	223	133,800.00	223
	Total	133,800.00	223	133,800.00	223
PY	เกษตรศาสตร์	49,800.00	83	49,800.00	83
	Total	49,800.00	83	49,800.00	83
VS	สัตวแพทยศาสตร์	16,800.00	28	16,800.00	28
	Total	16,800.00	28	16,800.00	28
Grand Total		288,000.00	480	288,000.00	480

Figure 5.77 : Registration Subject Pivot Table of SCCS155.

Scenario 6 : MU-SDM provides income information of faculties and institutes from registration.

Situation : The executive wants to know registration income of the 2547 academic year that is allocated to faculties and institutes.

Step

1. Select Allocation menu.
2. Drag Semester field under Allocation Lot dimension to Column Area.
3. Click drop-down of Semester field and check on the 2547 academic year.
4. Drag Account dimension to Row Area, and the Institute and Account Name field will be displayed on Row Area.
5. Expand institute field to drill-down to Account Name. The Allocation Pivot Table will be displayed in Figure 5.78.

Allocation					
Drop Filter Fields Here					
		Semester			Grand Total
		1/2547	2/2547	3/2547	
Institute	Account Name	Amount	Amount	Amount	Amount
☐ กองกิจการนักศึกษา	กิจกรรมนักศึกษาบัณฑิต	1,812,520.00	1,587,080.00		3,399,600 ▲
	ค่าบริการสุขภาพนักศึกษา	6,320,100.00	1,200.00		6,321,300
	เงินบำรุงศิลปวัฒนธรรมและนันทนาการ	2,500.00	2,200.00		4,700
	เงินสนับสนุนกีฬา	1,119,180.00	963,050.00		2,082,230
	Total	9,254,300.00	2,553,530.00		11,807,830
☐ มหาวิทยาลัย	กองทุนเงินืมเพื่อการศึกษา	870,560.00	845,808.00	75,408.00	1,791,776
	กองทุนเงินสงเคราะห์นักศึกษา	1,053,350.00	200.00		1,053,550
	เงินประกันหอพักนักศึกษา	1,234,200.00			1,234,200
	เงินประกันหอพักนักศึกษา (กาญจนบุรี)	165,000.00			165,000
	เงินพัฒนาการเรียนการสอน	2,104,410.00	1,291,745.00		3,396,155
	รายได้มหาวิทยาลัย	35,928,662.50	26,920,159.50	1,360,352.00	64,209,174
	เอกสารการลงทะเบียน	395,295.00			395,295
	Total	41,751,477.50	29,057,912.50	1,435,760.00	72,245,150
☐ คณะกายภาพบำบัดและวิทยาศาสตร์การเคลื่อนไหวประยุกต์		767,465.00	931,870.00		1,699,335
☐ คณะทันตแพทยศาสตร์		3,079,260.00	3,418,675.00	600.00	6,498,535
☐ คณะเทคนิคการแพทย์		1,352,355.00	1,263,360.00	199,170.00	2,814,885
☐ คณะพยาบาลศาสตร์		1,319,372.50	1,356,702.50	235,800.00	2,911,875
☐ คณะแพทยศาสตร์ โรงพยาบาลรามธิบดี		2,757,265.00	3,183,165.00	162,840.00	6,103,270
☐ คณะแพทยศาสตร์ศิริราชพยาบาล		7,196,620.00	1,948,795.00	155,070.00	9,300,485
☐ คณะเภสัชศาสตร์		3,369,875.00	4,015,825.00	460.00	7,386,160
☐ คณะวิทยาศาสตร์		15,013,175.00	12,642,575.00	625,200.00	28,280,950
☐ คณะวิศวกรรมศาสตร์		6,320,145.00	7,259,540.00	98,520.00	13,678,205
☐ คณะเวชศาสตร์เขตร้อน			21,600.00		21,600
☐ คณะศิลปศาสตร์		332,150.00	400,150.00	10,800.00	743,100
☐ คณะสังคมศาสตร์และมนุษยศาสตร์		1,757,860.00	1,689,345.00		3,447,205
☐ คณะสัตวแพทยศาสตร์		1,262,305.00	1,128,550.00	23,520.00	2,414,375
☐ คณะสาธารณสุขศาสตร์		3,840,865.00	3,195,575.00	314,160.00	7,350,600
☐ คณะสิ่งแวดล้อมและทรัพยากรศาสตร์		203,835.00	213,075.00	15,840.00	432,750
☐ โครงการจัดตั้งม.มหิดลกาญจนบุรี		944,885.00	522,105.00		1,466,990
☐ โครงการจัดตั้งม.มหิดลนครสวรรค์		24,395.00	23,800.00		48,195
☐ ธนาคาร		78,830.00	104,880.00	25,080.00	208,790
☐ วิทยาลัยการจัดการ		798,480.00	843,840.00	136,440.00	1,778,760
☐ วิทยาลัยดุริยางคศิลป์		95,040.00	347,760.00		442,800
☐ วิทยาลัยราชสุดา		261,210.00	227,525.00		488,735
☐ วิทยาลัยวิทยาศาสตร์และเทคโนโลยีการกีฬา		645,120.00	600,010.00		1,245,130
☐ วิทยาลัยศาสนศึกษา		565,325.00	521,325.00	155,880.00	1,242,530
☐ ศูนย์ศาลาษา		4,483,100.00	5,627,350.00		10,110,450
☐ สำนักคอมพิวเตอร์		3,561,210.00	3,069,300.00		6,630,510
☐ สำนักงานเขตพื้นที่		944,175.00	781,275.00		1,725,450
☐ สำนักหอสมุด		2,895,025.00	2,496,875.00		5,391,900
Grand Total		130,995,560.00	104,749,660.00	4,202,660.00	239,947,880 ▼

Figure 5.78 : Institute’s Income from Registration of Academic Year 2547.

Additionally, the executive can drill-down to a detail of allocation in each semester via the following steps.

6. Add Lot No field to Column Area.
7. Expand Semester field to drill-down to Lot No Level, the Allocation Pivot Table will be displayed in Figure 5.79

Allocation								
Drop Filter Fields Here								
		Semester ▾ Lot No						
		1/2547						
		1	2	3	4	5	6	7
Institute ▾	Account Name	Amount	Amount	Amount	Amount	Amount	Amount	Amount
☑ กองกิจการนักศึกษา	กิจกรรมนักศึกษาบัณฑิต	224,290.00	305,380.00	1,135,410.00	65,040.00	21,280.00	3,500.00	
	ค่าบริการสุขภาพนักศึกษา	710,400.00	1,191,000.00	3,791,100.00	277,200.00	90,000.00	15,000.00	
	เงินบำรุงศิลปวัฒนธรรมและนันทนาการ							
	เงินสนับสนุนกีฬา	160,010.00	200,260.00	653,910.00	46,300.00	15,200.00	2,500.00	
	Total	1,094,700.00	1,696,640.00	5,580,420.00	388,540.00	126,480.00	21,000.00	
☑ มหาวิทยาลัย	กองทุนเงินอิมมูเพื่อการศึกษ		86,784.00	731,568.00	46,064.00	5,264.00	32.00	
	กองทุนเงินสงเคราะห์นักศึกษา	118,400.00	198,500.00	631,850.00	46,200.00	15,000.00	2,500.00	
	เงินประกันหอพักนักศึกษา		41,800.00	1,191,300.00	1,100.00			
	เงินพัฒนาการเรียนการสอน			165,000.00				
	รายได้มหาวิทยาลัย	5,288,215.00	6,173,276.00	19,129,519.50	2,256,076.00	745,801.00	135,058.00	
	เอกสารการลงทะเบียน	41,580.00	66,690.00	257,325.00	22,470.00	5,670.00	780.00	
	Total	5,448,195.00	6,780,620.00	23,994,322.50	2,373,800.00	771,735.00	138,370.00	1
☑ คณะกายภาพบำบัดและวิทยาศาสตร์การเคลื่อนไหวประยุกต์			286,245.00	472,030.00		8,595.00	595.00	
☑ คณะทันตแพทยศาสตร์		1,179,930.00	1,310,825.00	483,210.00	13,150.00	84,550.00		
☑ คณะเทคนิคการแพทย์			1,063,975.00	106,650.00	167,110.00	11,385.00		
☑ คณะพยาบาลศาสตร์			786,130.00	520,312.50	12,930.00			
☑ คณะแพทยศาสตร์ โรงพยาบาลรามาธิบดี		1,631,635.00	369,950.00	688,395.00	30,810.00	34,435.00		
☑ คณะแพทยศาสตร์ศิริราชพยาบาล		2,789,800.00	1,726,735.00	2,500,975.00	128,655.00	46,735.00	2,400.00	
☑ คณะเภสัชศาสตร์			681,750.00	2,546,910.00	129,140.00	8,075.00		
☑ คณะวิทยาศาสตร์		1,381,990.00	1,708,615.00	11,402,445.00	225,170.00	31,065.00		
☑ คณะวิศวกรรมศาสตร์			45,725.00	5,674,190.00	531,370.00	62,280.00		
☑ คณะเวชศาสตร์เขตร้อน								
☑ คณะศิลปศาสตร์		329,875.00	2,275.00					
☑ คณะสังคมศาสตร์และมนุษยศาสตร์			134,400.00	1,301,445.00	234,840.00	85,255.00	1,080.00	
☑ คณะสัตวแพทยศาสตร์			108,880.00	1,132,445.00	4,715.00	8,430.00		
☑ คณะสาธารณสุขศาสตร์			7,890.00	806,215.00	1,637,175.00	933,675.00	282,800.00	1
☑ คณะสิ่งแวดล้อมและทรัพยากรศาสตร์			2,635.00	198,440.00	2,520.00			
☑ โครงการจัดตั้งม.มหิดลกาญจนบุรี			31,285.00	910,810.00	2,195.00	595.00		
☑ โครงการจัดตั้งม.มหิดลนครสวรรค์				24,395.00				
☑ ธนาคาร		9,790.00	15,300.00	37,020.00	11,500.00	3,780.00	20.00	
☑ วิทยาลัยการจัดการ			68,400.00	689,400.00	35,640.00	5,040.00		
☑ วิทยาลัยดุริยางคศิลป์				95,040.00				
☑ วิทยาลัยราชสุดา					257,280.00	955.00	2,975.00	
☑ วิทยาลัยวิทยาศาสตร์และเทคโนโลยีการกีฬา			2,395.00	608,470.00	11,265.00	22,990.00		
☑ วิทยาลัยศาสนศึกษา		52,200.00	360.00	237,590.00	230,405.00	44,770.00		
☑ ศูนย์ศาลา		160,150.00	314,400.00	3,903,050.00	49,300.00	15,200.00	2,500.00	
☑ สำนักคอมพิวเตอร์		480,900.00	601,200.00	1,962,300.00	138,900.00	45,600.00	7,500.00	
☑ สำนักงานเคลมเบต		176,250.00	195,560.00	499,655.00	50,710.00	16,720.00	2,750.00	
☑ สำนักหอสมุด		400,350.00	500,900.00	1,635,025.00	115,750.00	38,000.00	6,250.00	
Grand Total		15,135,765.00	18,443,090.00	68,011,160.00	6,791,465.00	2,398,345.00	467,645.00	3

Figure 5.79 : Drill-down Allocation Income to Lot No Level.

Scenario 7 : The executive would like to know a source of allocated income of the faculty of science.

Situation : The executive wants to know from which department the allocated income of the faculty of science in the 2547 academic year was derived.

Step

1. Select Allocation by Program menu.
2. Drag Acd Year field under Allocation Lot dimension to Filter Area.
3. Add Institute field under Account dimension and Faculty Code under Program dimension to Row Area, respectively.
4. Add Amount field and Amount(%) field to data area.

5. Expand คณะวิทยาศาสตร์ Institute to display Faculty Code, the Allocation by Program Pivot Table will be displayed in Figure 5.80

Allocation Program			
Allocation Lot		Drop Column Fields Here	
2547			
Institute	Faculty Code	Amount	Amount(%)
กองกิจการนักศึกษา		13,811,580.00	5.76%
คณะกายภาพบำบัดและวิทยาศาสตร์การเคลื่อนไหวประยุกต์		1,699,335.00	0.71%
คณะทันตแพทยศาสตร์		6,498,535.00	2.71%
คณะเทคโนโลยีการแพทย์		2,834,885.00	1.18%
คณะพยาบาลศาสตร์		2,911,875.00	1.21%
คณะแพทยศาสตร์ โรงพยาบาลรามธิบดี		6,103,270.00	2.54%
คณะแพทยศาสตร์ศิริราชพยาบาล		9,300,485.00	3.88%
คณะเภสัชศาสตร์		7,386,180.00	3.08%
คณะวิทยาศาสตร์			
	AR	52,560.00	0.02%
	BM	654,820.00	0.27%
	CR	26,640.00	0.01%
	DT	672,300.00	0.28%
	EG	2,059,525.00	0.86%
	EN	251,470.00	0.10%
	MG	342,480.00	0.14%
	MT	1,696,935.00	0.71%
	NS	1,713,420.00	0.71%
	PH	1,136,965.00	0.47%
	PI	1,348,395.00	0.56%
	PY	949,140.00	0.40%
	RA	2,726,060.00	1.14%
	RS	18,720.00	0.01%
	SC	11,814,870.00	4.92%
	SI	2,131,565.00	0.89%
	SP	430,660.00	0.18%
	VS	254,425.00	0.11%
	Total	28,280,950.00	11.79%
คณะวิศวกรรมศาสตร์		13,678,205.00	5.70%
คณะเวชศาสตร์เขตร้อน		21,600.00	0.01%
คณะศิลปศาสตร์		743,100.00	0.31%
คณะสังคมศาสตร์และมนุษยศาสตร์		3,447,205.00	1.44%
คณะสัตวแพทยศาสตร์		2,414,375.00	1.01%
คณะสาธารณสุขศาสตร์		7,350,600.00	3.06%
คณะสิ่งแวดล้อมและทรัพยากรศาสตร์		432,750.00	0.18%
โครงการจัดตั้งม.มหิดลกาญจนบุรี		691,390.00	0.29%
โครงการจัดตั้งม.มหิดลนครสวรรค์		48,195.00	0.02%
ธนาคาร		208,790.00	0.09%
มหาวิทยาลัย		72,245,150.00	30.11%
วิทยาลัยการจัดการ		1,778,760.00	0.74%
วิทยาลัยดุริยางคศิลป์		442,800.00	0.18%
วิทยาลัยราชสุดา		488,735.00	0.20%
วิทยาลัยวิทยาศาสตร์และเทคโนโลยีการกีฬา		1,245,130.00	0.52%
วิทยาลัยศาสนศึกษา		1,242,530.00	0.52%
ศูนย์ศาลาษา		8,882,300.00	3.70%
สำนักคอมพิวเตอร์		6,630,510.00	2.76%
สำนักงานเขตพื้นที่		1,725,450.00	0.72%
สำนักหอสมุด		5,391,900.00	2.25%
Grand Total		239,947,880.00	100.00%

Figure 5.80 : Allocation by Program Pivot Table.

6. The executive can change a dimension presentation by rotating the pivot table, and move Institute field from Row Area to Column Area, as shown in Figure 5.81.

Allocation Program										
Allocation Lot 2547										
Institute										
<input type="checkbox"/> คณะเภสัชศาสตร์ <input type="checkbox"/> คณะวิทยาศาสตร์ <input type="checkbox"/> คณะวิศวกรรมศาสตร์ <input type="checkbox"/> คณะเวชศาสตร์เขตร้อน <input type="checkbox"/> คณะศิลปศาสตร์										
Faculty Code	Amount	Amount(%)	Amount	Amount(%)	Amount	Amount(%)	Amount	Amount(%)	Amount	Amount(%)
AR			52,560.00	0.02%					705,900.00	0.29%
BM			654,820.00	0.27%						
CR			26,640.00	0.01%					17,040.00	0.01%
DT			672,300.00	0.28%					360.00	0.00%
EG			2,059,525.00	0.86%	13,662,005.00	5.69%			1,440.00	0.00%
EN			251,470.00	0.10%						
MG			342,480.00	0.14%						
MT			1,696,935.00	0.71%						
NS			1,713,420.00	0.71%						
PH			1,136,965.00	0.47%						
PI			1,348,395.00	0.56%			21,600.00	0.01%	1,800.00	0.00%
PY	7,386,180.00	3.08%	949,140.00	0.40%						
RA			2,726,060.00	1.14%					5,400.00	0.00%
RS			18,720.00	0.01%						
SC			11,814,870.00	4.92%					11,160.00	0.00%
SH										
SI			2,131,565.00	0.89%	16,200.00	0.01%				
SP			430,660.00	0.18%						
YS			254,425.00	0.11%						
Grand Total	7,386,180.00	3.08%	28,280,950.00	11.79%	13,678,205.00	5.70%	21,600.00	0.01%	743,100.00	0.31%

Figure 5.81 : Rotating Allocation by Program Pivot Table.

Scenario 8 : The executive wants to know income growth of faculties and institutes.

Situation : The executive wants to know income growth that faculties and institutes get from registration allocation.

Step

1. Select Allocation by Program menu.
2. Add Acd Year and Semester fields under Allocation Lot dimension to Row Area.
3. Add Institute field under Account dimension to Column Area.
4. Add Amount and Income Growth fields to data area.
5. Expand the academic year 2546, the Allocation by Program Pivot Table will be displayed in Figure 5.82.

Allocation Program										
Drop Filter Fields Here										
Institute Account Name										
<input type="checkbox"/> โครงการจัดตั้งม.มหิดลนครสวรรค์ <input type="checkbox"/> ธนาคาร <input type="checkbox"/> มหาวิทยาลัย <input type="checkbox"/> วิทยาลัยการจัดการ										
Acad Year	Semester	Amount	Income Growth	Amount	Income Growth	Amount	Income Growth	Amount	Income Growth	
2546	1/2546			81,960.00	81,960.00	38,616,385.00	38,616,385.00	663,120.00	663,120.00	
	2/2546			85,550.00	3,590.00	26,811,325.00	-11,805,060.00	700,920.00	37,800.00	
	3/2546			29,080.00	-56,470.00	1,742,830.00	-25,068,495.00	136,440.00	-564,480.00	
	Total			196,590.00	196,590.00	67,170,540.00	67,170,540.00	1,500,480.00	1,500,480.00	
2547		48,195.00	48,195.00	208,790.00	12,200.00	72,245,150.00	5,074,610.00	1,778,760.00	278,280.00	
2548		95,200.00	47,005.00	220,390.00	11,600.00	78,446,185.00	6,201,035.00	1,985,400.00	206,640.00	
Grand Total		143,395.00	143,395.00	625,770.00	625,770.00	217,861,875.00	217,861,875.00	5,264,640.00	5,264,640.00	

Figure 5.82 : Allocation by Program Pivot Table with Income Growth.

6. Move Institute field from Column Area to Row Area, in front of Acd Year field.
7. Expand Institute field to view a detail in the academic year and semester level, the pivot table will be displayed in Figure 5.83. Income growth of each institute is a comparison of institute's income at a time with the previous time in the same time level.

Allocation Program			
Drop Filter Fields Here		Drop Column Fields Here	
Institute	Acad Year	Semester	
กองกิจการนักศึกษา	2546		12,966,930.00
	2547		13,811,580.00
	2548		15,541,890.00
คณะวิทยาศาสตร์	2546		28,588,900.00
	2547		28,280,950.00
	2548		28,357,355.00
คณะวิศวกรรมศาสตร์	2546		12,311,855.00
	2547		13,878,205.00
	2548		15,389,525.00
มหาวิทยาลัย	2546	1/2546	38,616,385.00
		2/2546	26,811,325.00
		3/2546	1,742,830.00
		Total	67,170,540.00
	2547		72,245,150.00
2548		78,446,185.00	

Figure 5.83 : An Alternative View of Allocation by Program Pivot Table.

Scenario 9 : MU-SDM provides information of the distribution of grades.

Situation : After seeing the registration subject information of SCCS155, the executive wants to know the distribution of grades that students got for the subject in 1/2547 semester.

Step

1. Select Grading menu.
2. Drag Semester field from Table Field List to Filter Area and check on 1/2547.
3. Add Subject dimension from Table Field List to Filter Area and check on SCCS155 COMPUTER APPLICATIONS.
4. Add Program dimension to Row Area, and the Faculty Code and Program fields will be displayed on row header.
5. Add Grade dimension from Table Field List to Column Area, the Grading Pivot Table will be displayed in Figure 5.84.

Faculty Code		Program	A	B+	B	C+	C	F	Grand Total
AR	ศิลปศาสตร์ (ภาษาไทย)		11	20	30	11	9	81	
	ศิลปศาสตร์ (ภาษาอังกฤษ)	1	10	29	13	5	7	65	
	Total	1	21	49	43	16	16	146	
NS	พยาบาลศาสตร์	3	43	110	57	9	1	223	
	Total	3	43	110	57	9	1	223	
PY	เภสัชศาสตร์	14	39	26	4			83	
	Total	14	39	26	4			83	
VS	สัตวแพทยศาสตร์	13	12	2	1			28	
	Total	13	12	2	1			28	
Grand Total			31	115	187	105	25	17	480

Figure 5.84 : Grading Pivot Table of SCCS155.

6. Add Student(%) field to data area to show the percentage of students, the Grading Pivot Table will be displayed in Figure 5.85.

Faculty Code	Program	A		B+		B		C+		C		F		Grand Total	
		#Student	Student(%)	#Student	Student(%)	#Student	Student(%)	#Student	Student(%)	#Student	Student(%)	#Student	Student(%)	#Student	Student(%)
AR	ศิลปศาสตร์ (ภาษาไทย)			11	2.29%	20	4.17%	30	6.25%	11	2.29%	9	1.88%	81	16.88%
	ศิลปศาสตร์ (ภาษาอังกฤษ)	1	0.21%	10	2.08%	29	6.04%	13	2.71%	5	1.04%	7	1.46%	65	13.54%
	Total	1	0.21%	21	4.38%	49	10.21%	43	8.96%	16	3.33%	16	3.33%	146	30.42%
NS	พยาบาลศาสตร์	3	0.63%	43	8.96%	110	22.92%	57	11.88%	9	1.88%	1	0.21%	223	46.46%
	Total	3	0.63%	43	8.96%	110	22.92%	57	11.88%	9	1.88%	1	0.21%	223	46.46%
PY	เภสัชศาสตร์	14	2.92%	39	8.13%	26	5.42%	4	0.83%					83	17.29%
	Total	14	2.92%	39	8.13%	26	5.42%	4	0.83%					83	17.29%
VS	สัตวแพทยศาสตร์	13	2.71%	12	2.50%	2	0.42%	1	0.21%					28	5.83%
	Total	13	2.71%	12	2.50%	2	0.42%	1	0.21%					28	5.83%
Grand Total		31	6.46%	115	23.96%	187	38.96%	105	21.88%	25	5.21%	17	3.54%	480	100.00%

Figure 5.85 : Grading Pivot Table of SCCS155 with Students (%).

Scenario 10 : MU-SDM provides information about the efficiency of students.

Situation : The executive wants to know the number of students that has good GPA for 1/2547 semester. This information represents the efficiency of students.

Step

1. Select GPA menu.
2. Add Semester field to Field Filter Area and check on 1/2547.
3. Add Faculty Code field to Row Area.
4. Add GPA Level field to Column Area.
5. Add Student field to data area, the GPA Pivot Table will be displayed in Figure 5.86.

GPA									
Semester 1/2547									
Faculty Code	GPAlevel								
	2.00 - 2.24	2.25 - 2.49	2.50 - 2.74	2.75 - 2.99	3.00 - 3.24	3.25 - 3.49	3.50 - 3.74	3.75 - 4.00	Grand Total
AR	3	22	20	20	21	15	9	6	146
BM	7	7	18	28	45	65	42	13	222
CR	2	5	10	23	33	61	40	27	215
DT	1		20	38	87	134	99	68	458
EG	5	189	263	216	139	68	24	8	1,058
EN	2	14	21	25	17	7	5		110
MG	3	32	46	56	93	95	60	30	432
MT	5	39	102	116	112	106	61	22	585
NS	5	20	78	182	222	190	102	20	829
PH	3	50	85	117	160	141	79	42	712
PI	0	37	70	103	104	80	63	14	488
PY	5	17	44	88	99	106	75	24	472
RA	7	35	109	222	250	281	218	118	1,276
RS		7	25	26	26	11	3		102
SC	3	180	275	270	242	184	101	77	1,615
SH				3	13	28	10	9	64
SI	3	18	86	172	297	354	351	274	1,668
SP	3	57	49	50	30	22	14	9	280
VS	3	1	16	17	49	32	26	11	163
Grand Total	3	710	1,337	1,772	2,039	1,980	1,382	772	10,895

Figure 5.86 : GPA Pivot Table.

- Remove #Student field from data area and add Student(%) to data area to view the percentage of students, the GPA Pivot Table will be shown in Figure 5.87.

GPA									
Semester 1/2547									
Faculty Code	GPAlevel								
	Student(%)	2.25 - 2.49	2.50 - 2.74	2.75 - 2.99	3.00 - 3.24	3.25 - 3.49	3.50 - 3.74	3.75 - 4.00	Student(%)
AR	%	0.20%	0.18%	0.18%	0.19%	0.14%	0.08%	0.06%	1.34%
BM	%	0.06%	0.17%	0.26%	0.41%	0.60%	0.39%	0.12%	2.04%
CR	%	0.05%	0.09%	0.21%	0.30%	0.56%	0.37%	0.25%	1.97%
DT	%		0.18%	0.35%	0.80%	1.23%	0.91%	0.62%	4.20%
EG	%	1.73%	2.41%	1.98%	1.28%	0.62%	0.22%	0.07%	9.71%
EN	%	0.13%	0.19%	0.23%	0.16%	0.06%	0.05%		1.01%
MG	%	0.29%	0.42%	0.51%	0.85%	0.87%	0.55%	0.28%	3.97%
MT	%	0.36%	0.94%	1.06%	1.03%	0.97%	0.56%	0.20%	5.37%
NS	%	0.18%	0.72%	1.67%	2.04%	1.74%	0.94%	0.18%	7.61%
PH	%	0.46%	0.78%	1.07%	1.47%	1.29%	0.73%	0.39%	6.54%
PI	%	0.34%	0.64%	0.95%	0.95%	0.73%	0.58%	0.13%	4.48%
PY	%	0.16%	0.40%	0.81%	0.91%	0.97%	0.69%	0.22%	4.33%
RA	%	0.32%	1.00%	2.04%	2.29%	2.58%	2.00%	1.08%	11.71%
RS	%	0.06%	0.23%	0.24%	0.24%	0.10%	0.03%		0.94%
SC	%	1.47%	2.52%	2.48%	2.22%	1.69%	0.93%	0.71%	14.82%
SH	%			0.03%	0.12%	0.26%	0.09%	0.08%	0.59%
SI	%	0.17%	0.79%	1.58%	2.73%	3.25%	3.22%	2.51%	15.31%
SP	%	0.52%	0.45%	0.46%	0.28%	0.20%	0.13%	0.08%	2.57%
VS	%	0.01%	0.15%	0.16%	0.45%	0.29%	0.24%	0.10%	1.50%
Grand Total	%	6.52%	12.27%	16.26%	18.72%	18.17%	12.68%	7.09%	100.00%

Figure 5.87 : GPA Pivot Table with Percentage of Students.

The executive can view a detail of students that get GPA greater than 2.00 through the following steps.

7. Set filter on GPA Level dimension by checking on GPA greater than 2.00.
8. Drag #Student field from Table Field List and drop in front of Student(%) field in data area. The cube will be displayed in Figure 5.88

GPA													
Semester		GPAlevel											
1/2547		2.75 - 2.99		3.00 - 3.24		3.25 - 3.49		3.50 - 3.74		3.75 - 4.00		Grand Total	
Faculty Code	Student(%)	#Student	Student(%)	#Student	Student(%)	#Student	Student(%)	#Student	Student(%)	#Student	Student(%)	#Student	Student(%)
SI	1.68%	297	2.90%	354	3.46%	351	3.43%	274	2.67%	109	1.06%	1661	16.22%
SC	2.64%	242	2.36%	184	1.80%	101	0.99%	77	0.75%	55	0.54%	1364	13.32%
RA	2.17%	250	2.44%	281	2.74%	218	2.13%	118	1.15%	30	0.29%	1263	12.33%
DT	0.37%	87	0.85%	134	1.31%	99	0.97%	68	0.66%	11	0.11%	457	4.46%
PH	1.14%	160	1.56%	141	1.38%	79	0.77%	42	0.41%	8	0.08%	682	6.66%
CR	0.22%	33	0.32%	61	0.60%	40	0.39%	27	0.26%	7	0.07%	206	2.01%
PY	0.86%	99	0.97%	106	1.03%	75	0.73%	24	0.23%	6	0.06%	459	4.48%
MG	0.55%	93	0.91%	95	0.93%	60	0.59%	30	0.29%	5	0.05%	417	4.07%
MT	1.13%	112	1.09%	106	1.03%	61	0.60%	22	0.21%	4	0.04%	562	5.49%
NS	1.78%	222	2.17%	190	1.85%	102	1.00%	20	0.20%	4	0.04%	818	7.99%
EG	2.11%	139	1.36%	68	0.66%	24	0.23%	8	0.08%	3	0.03%	910	8.88%
YS	0.17%	49	0.48%	32	0.31%	26	0.25%	11	0.11%	3	0.03%	155	1.51%
BM	0.27%	45	0.44%	65	0.63%	42	0.41%	13	0.13%	2	0.02%	220	2.15%
PI	1.01%	104	1.02%	80	0.78%	63	0.62%	14	0.14%	2	0.02%	473	4.62%
SH	0.03%	13	0.13%	26	0.27%	10	0.10%	9	0.09%	1	0.01%	64	0.62%
SP	0.49%	30	0.29%	22	0.21%	14	0.14%	9	0.09%	1	0.01%	232	2.26%
AR	0.20%	21	0.21%	15	0.15%	9	0.09%	6	0.06%			113	1.10%
EN	0.24%	17	0.17%	7	0.07%	5	0.05%					89	0.87%
RS	0.25%	26	0.25%	11	0.11%	3	0.03%					98	0.96%
Grand Total	17.30%	2039	19.91%	1980	19.33%	1382	13.49%	772	7.54%	251	2.45%	10243	100.00%

Figure 5.88 : GPA Pivot Table with GPA Greater Than 2.00

The executive can drill-down through GPA level to consider the Program level. For example, the executive wants to know which faculty and program whose students get GPA not less than 3.75.

9. Add Program dimension to Row Area.
10. Move GPA Level from Column Area to the front of Faculty Code field in order to display a more proper presentation of the pivot table.
11. Expand GPA 3.75-4.00 to view Program level, the GPA Pivot Table will be displayed in Figure 5.89.

GPA			Drop Column Fields Here	
Semester			#Student	Student(%)
1/2547				
GPAlevel	Faculty Code	Program	#Student	Student(%)
2.00 - 2.24			710	6.93%
2.25 - 2.49			1337	13.05%
2.50 - 2.74			1772	17.30%
2.75 - 2.99			2039	19.91%
3.00 - 3.24			1980	19.33%
3.25 - 3.49			1382	13.49%
3.50 - 3.74			772	7.54%
3.75 - 4.00	BM		2	0.02%
	CR		7	0.07%
	DT		11	0.11%
	EG	วิศวกรรมศาสตร์ (คอมพิวเตอร์)	2	0.02%
		วิศวกรรมศาสตร์	1	0.01%
		Total	3	0.03%
	MG		5	0.05%
	MT	วิทยาศาสตร์ (รังสีเทคนิค 2 ปี)	2	0.02%
		วิทยาศาสตร์ (เทคนิคการแพทย์)	1	0.01%
		วิทยาศาสตร์ (รังสีเทคนิค)	1	0.01%
		Total	4	0.04%
	NS		4	0.04%
	PH	สาธารณสุขศาสตร์ (พิเศษ 3 ปี)	6	0.06%
		วิทยาศาสตร์ (สาธารณสุขศาสตร์)	1	0.01%
		วิทยาศาสตร์ (สาธารณสุข-อาหารและโภชนาการ)	1	0.01%
		Total	8	0.08%
	PI		2	0.02%
	PY	เภสัชศาสตร์	4	0.04%
		เภสัชศาสตร์ (พิเศษ)	2	0.02%
		Total	6	0.06%
	RA		30	0.29%
	SC	วิทยาศาสตร์ (เทคโนโลยีสารสนเทศและการสื่อสาร)	15	0.15%
		วิทยาศาสตร์	13	0.13%
		วิทยาศาสตร์ (เทคโนโลยีชีวภาพ)	6	0.06%
		วิทยาศาสตร์ (เคมี)	5	0.05%
		วิทยาศาสตร์ (ชีววิทยา)	5	0.05%
		วิทยาศาสตร์ (คอมพิวเตอร์)	4	0.04%
		วิทยาศาสตร์ (ฟิสิกส์)	3	0.03%
		วิทยาศาสตร์ (คณิตศาสตร์)	2	0.02%
		วิทยาศาสตร์ (เทคโนโลยีการอาหาร)	1	0.01%
		วิทยาศาสตร์ (พฤกษศาสตร์)	1	0.01%
		Total	55	0.54%
	SH		1	0.01%
	SI	แพทยศาสตร์	96	0.94%
		การแพทย์แผนไทยประยุกต์บัณฑิต	6	0.06%
		วิทยาศาสตร์ (กายภาพบำบัด)	3	0.03%
		วิทยาศาสตร์ (เวชศาสตร์การธนาคารเลือด)	3	0.03%
		วิทยาศาสตร์ (เวชทันตกรรม)	1	0.01%
		Total	109	1.06%
	SP		1	0.01%
	VS		3	0.03%
	Total		251	2.45%
Grand Total			10243	100.00%

Figure 5.89 : Alternative View of GPA Pivot Table

To find out most students who get GPA not less than 3.75 study in which faculty, perform the following steps.

12. Set filter on GPA Level dimension by checking on GPA 3.75 – 4.00.
13. Remove Program field from Row Area
14. Click Collapse Items in the context menu that appears.
15. Right-click data area of #Student field

- Click sort descending on the context menu that appears. The GPA Pivot Table will be displayed faculty whose most student get GPA not less than 3.75 in the top row as shown in Figure 5.90.

GPA			
Semester 1/2547			
GPAlevel	Faculty Code	#Student	Student(%)
3.75 - 4.00	SI	109	43.43%
	SC	55	21.91%
	RA	30	11.95%
	DT	11	4.38%
	PH	8	3.19%
	CR	7	2.79%
	PY	6	2.39%
	MG	5	1.99%
	MT	4	1.59%
	NS	4	1.59%
	EG	3	1.20%
	VS	3	1.20%
	BM	2	0.80%
	PI	2	0.80%
	SH	1	0.40%
	SP	1	0.40%
Total		251	100.00%
Grand Total		251	100.00%

Figure 5.90 : GPA Pivot Table

CHAPTER 6

DISCUSSION AND CONCLUSION

After we have worked on the MU-SDM, in this chapter, we will make the discussion and conclusion of our research. In Addition, some interesting issues for future work are also suggested as a closing section.

6.1 Discussion and Conclusion

This research project has been developed to support the university executives' requirement for an information system that provides management information and analysis, supporting their decision in the education area. They need summary information from which they can analyze trends that affect the university. This information cannot be generated by an operational system because its database design is not optimized and needs time to process. The project uses a data mart technology to solve this problem. Moreover, in order to refresh the data mart, we use change monitoring for detecting changed data and extraction it in order to incrementally update the data mart.

The data mart is designed to support management information and an analytical purpose. It uses OLAP, a technology allowing decision makers to interactively access a wide variety of information to discover a trend and analyze critical factors. OLAP data is organized into multidimensional cubes. This structure gives a better performance for queries. The data warehouse provides a user-friendly interface that is flexible and offers multidimensional views. The process of data mart data acquisition is called ETL. This process is complex and time-consuming, and composed of several sub-processes such as extraction, and transforming and loading data from the operational source systems into the data mart, and creating the cubes. This process is more complex as time advances and operational source data are changed, making it necessary for the student cubes to be refreshed. Refreshing the data mart by repeating an ETL process with all operational data is a very time-

consuming task and may lead to a loss of history. The ETL for refreshment is required in order to incrementally update a data mart. The change monitoring is applied for detecting changed data. We design forty-five trigger rules, which are applied with fifteen tables as change monitoring mechanism. The data are extracted, transformed in data mart format and propagated to the data mart periodically through the course of change. Finally, the data cube is processed to load dimension and fact tables and stored in the cube.

Based on the idea presented in chapters 3 and 4, we have designed and developed the prototype of MU-SDM to support data mart administrators in creating and maintaining the student data mart, and have developed a data analysis tool for university executives to make a meaningful decision. The system covers the necessary functions to support university executives' decision-making in the academic area. The system has been tested on its functionalities and evaluated by using some interesting scenarios that usually happen in a university. We use Mahidol University student information as a case study to create a student data mart and use a trigger technique to detect changed data to incrementally update the data mart. We also create essential cubes that provide useful information for decision making and develop a user interface for presenting multidimensional information and graph to the user. Users can perform multidimensional operations, such as roll-up, drill-down, slice-dice and pivoting to view data in various views. However, some features such as interpretation and a structural change must be enhanced.

In order to show that the MU-SDM has more benefit than the report generation process of MU student information system. We use scenario 9: MU-SDM provides information of the distribution of grades in section 5.2 to compare Grading Cube with the grading report generation in some aspects.

- Grading Cube
 1. The data mart is designed to aggregate and summarize a large amount of data and transformed in appropriate format which is ready to be used for analysis. Then it provides rapid answers to complex analytical queries.

2. The MU-SDM structure is easy for user to navigate and understand. User can change multiple views and set criteria with drag and drop feature.
 3. MU-SDM keeps historical data which is used for analysis and decision.
- Grading report generation
 1. Data from student information system are not available for analysis. In order to generate this report which is a complex query, many processes are performed. The database is designed confirm to normalized model, then the queries have to join with many tables and use aggregation operation, and old transaction are separated from current database then data are loaded from multiple tables. It takes several days for preparing and generating the report, and process must be repeated when the report is requested.
 2. Multiple aspects of presentation are not available for users.
 3. Old transactions are removed from current database to reduce processing time for performing many operations such as read, insert, update and delete, and to utilize disk space, So historical data is not available.

In summary, the major benefits derived from the MU-SDM are:

1. MU-SDM provides a user-friendly interface that is easy to use. The ETL process performs necessary tasks to acquire data mart data. Then the data mart administrator finds it easy to create and maintain the data mart. Besides, the information delivery process is designed to provide flexible, easy-to-use functions and useful information for decision makers. They can use easily drag and drop operation to perform OLAP operations.
2. Various steps and complexity of creating and maintaining a data mart gain more convenience and a faster operation through an automated process of ETL, which helps solve the problems for data mart administrators. This

process reduces time and effort for acquiring data mart data and enables accuracy and consistency.

3. In order to reduce time and effort for data mart refreshment, the ETL for refreshment is designed by adding change monitoring task for detecting changed data. These data are processed by change extraction before transformed and propagated into the data mart.
4. MU-SDM provides analytical information to decision makers that can not be found in an operational system. It provides multidimensional views that are useful, flexible, as well as easy to use and understand for decision makers for their strategic planning and trend analysis.

6.2 Difficulties and Lesson Learnt

The difficulties of this research project are listed as follows:

1. ETL is a complex process for data acquisition. It consolidates, integrates, summarizes and converts operational data into data warehouse. The process is more complex when operational source are changed and data warehouse must be synchronized with.
2. The data warehouse schema which provide information support the university executives for strategic planning needs well analysis and design.

The lesson learnt from this research project are listed as follows:

1. The data warehouse concept consists of the architecture, data modeling, OLAP technology and data warehouse process.
2. Change monitoring which help to incrementally refresh the data warehouse, is an important process. It detects changed data by applying trigger technique. It reduces human effort and time consuming for the data warehouse refreshment.
3. Good understanding the nature of operational system is important for designing triggers that are used to detect changed data.
4. Microsoft SQL Server 2000 Analysis Services is an OLAP tool for organizing data from a data warehouse into multidimensional cube to answer complex analytical queries.

5. Office pivot table is a Microsoft Office component which is used for displaying data cube in an interactive table to the user.

6.3 Suggestions for Future Work

This prototype is designed and developed to support the basic function of a student data mart. However, in order to implement a full scale of data warehouse which provides more useful functions to university executives, some features should be improved and enhanced in the following areas.

- **Add other student data and analytical functions.** MU-SDM will be more useful to university executives if some additional student data, such as graduate student information, instructor work-load information and other types of interpretations, etc. are provided. Since the information is provided by different sources, the data mart should have a more complex process to aggregate it before being loaded into the student data mart.
- **Add other data marts.** In order to complete the data warehouse and more usefully support university executives, other subjects of a university may be added, such as personnel and financial information, to create other data marts and combine them to create a university data warehouse. Thus, all data marts are related and give more analytical information to university executives. Moreover, because these operational data are provided by multiple sources, the data warehouse can solve the problem by integrating data from different sources and structures.
- **Support structural change.** If the interesting structure of a data source or cube structure it changed, the data warehouse data and its structure can be maintained.
- **Data warehouse security.** The system classifies users according to the limitation of access, like a limited access user and an unlimited access user, or classifies the data by confidentiality as public, confidential and top secret. Therefore, an access control policy based on the class of user and data may be used to define which subject areas are allowed to which users.

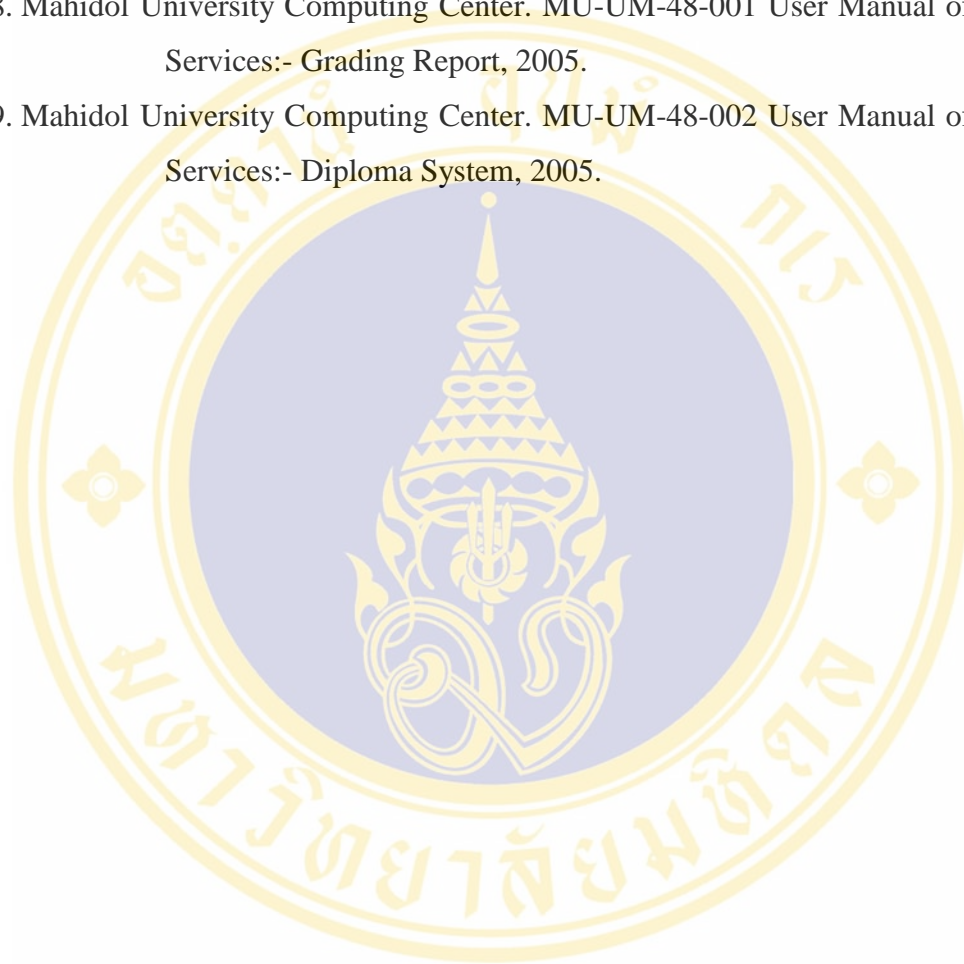
REFERENCES

1. P. Chamoni, P.Gluchowski. Analytische Informationssysteme: Data Warehouse, On-Line Analytical Processing, Data Mining., Berlin: Springer-Verlag; 1999.
2. Rob Mattison. Data Warehousing : Strategies, Technologies, and Techniques. New York: Willey Publ; 1998.
3. Harry S. Singh. Data Warehousing : concepts, technologies, implementations, and management. New Jersey: Prentice Hall PTR; 1997.
4. Paulraj Ponniah. Data Warehousing Fundamentals : A Comprehensive Guide for IT Professionals. Wiley-Interscience; 2001.
5. Lee, D., W. Mao, H. Chiu, W. W. Chu. Designing Triggers with Trigger-By-Example. Knowledge and Information Systems 2004; 7: 110 - 134.
6. Vavouras A. A Metadata-Driven Approach for Data Warehouse Refreshment [Dissertation]. University of Zurich; 2002.
7. S. Chaudhuri, U. Dayal. An Overview of Data Warehousing and OLAP Technology. ACM SIGMOD Record, 26:1, March 1997.
8. Creating and Using Data Warehouses Overview. Available from: <http://msdn.microsoft.com/library/default.asp>[Accessed January 10, 2004].
9. SQL Server 2000 Resort Kit. Available from: <http://www.microsoft.com/resources/documentation/sql/2000/all/reskit/en-us/default.mspx> [Accessed January 10, 2004].
10. Akanit Chairut, The Exploitation of Web-based and Active Rules Techniques for Data Preparation, Tracking and Verification for MU Annual Promotion [Master Thesis]. Bangkok: Department of Computer Science, Faculty of Science, Mahidol University; 2002.

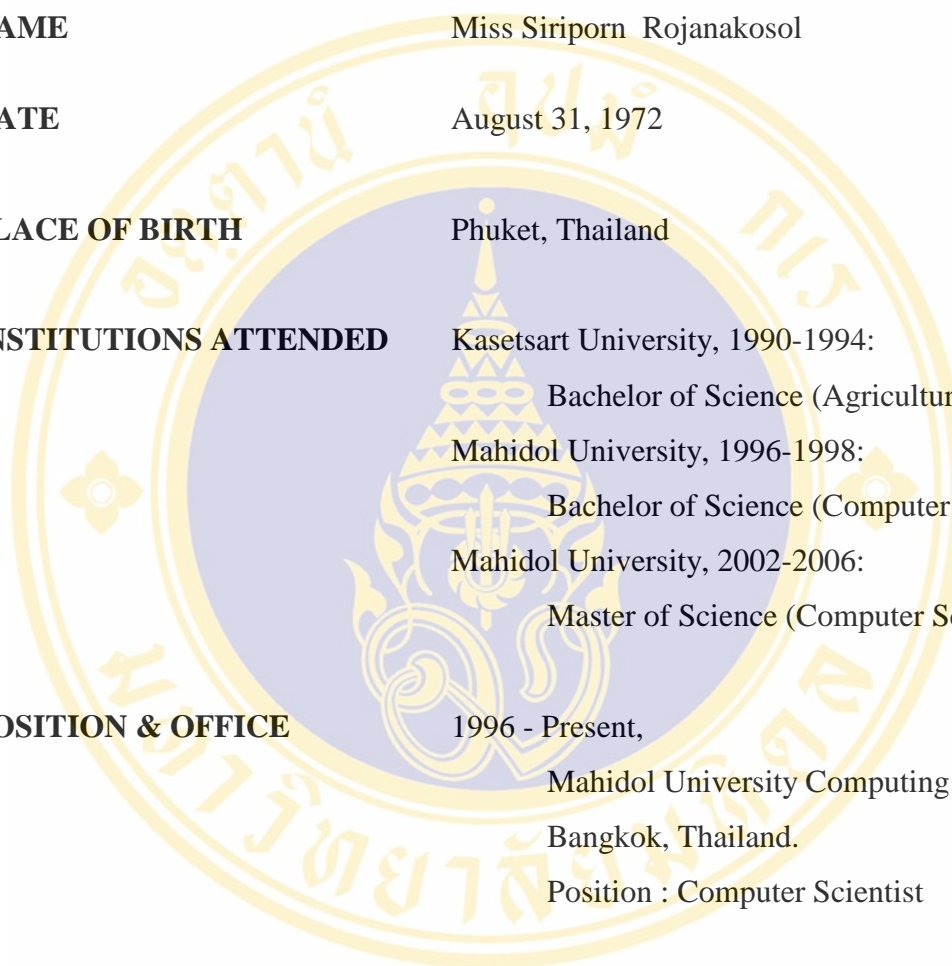
11. J. Hammer, H. Garcia-Molina, J. Widom, W. J. Labio, Y. Zhuge. The Stanford Data Warehousing Project. IEEE Data Engineering Bulletin, June 1995.
12. W. J. Labio, Y. Zhuge, J. L. Wiener, H. Gupta, H. Garcia-Molina, J. Widom. The WHIPS Prototype for Data Warehouse Creation and Maintenance. In Proceedings of the ACM SIGMOD Conference, Tuscon, Arizona, May, 1997. Short Overview.
13. J. L. Wiener, H. Gupta, W. J. Labio, Y. Zhuge, H. Garcia-Molina, J. Widom. "A System Prototype for Warehouse View Maintenance." In Proceedings of the ACM Workshop on Materialized Views: Techniques and Applications, Montreal, Canada, June 7, 1996
14. Georgetown University Enterprise Data Warehouse. Available from : <http://guweb.georgetown.edu/uis/ia/dw/> [Accessed January 20, 2004]
15. University of Washington Data Warehouse Program. Available form : <http://ucs.admin.washington.edu/dwp/> [Accessed January 20, 2004]
16. University of Pennsylvania Data Warehouse. Available From : <http://www.upenn.edu/computing/da/dw/> [Accessed June 6, 2005]
17. University of Maryland Data Warehouse. Available From : <http://www.oit.umd.edu/units/dataadmin/DataWarehouse/> [Accessed June 6, 2005]
18. Arizona State University Data Warehouse. Available From : http://www.asu.edu/data_admin/ [Accessed June 6, 2005]
19. Business Intelligence Tools. Available from : <http://www.businessintelligencetools.com/olap.html> [Accessed June 10, 2005]
20. T.Willis, P.ilton, M.Brown, M.Reynolds, M.Lane Thomas, C.Caison, J.hasan, T.Barnaby, P.Boutquin, R.Harrison, D.shlosberg, T.Waters, "Professional VB6 Web Programming", Wrox Press Ltd., 1999.
21. Charles Williams, "Professional Visual Basic 6 Databases", Wrox Press Ltd., 1999.

22. New to Java Programming Center. Available from :
<http://java.sun.com/developer/onlineTraining/new2java/programming/learn/unravelingjava.html>. [Accessed August 13, 2005]
23. The Cold Fusion Web Database Construction Kit. Available from :
<http://docs.rinet.ru/Cold/ch01/ch01.htm>. [Accessed August 13, 2005]
24. W.H. Inmon. Building the Data Warehouse. 2nd edition. John Wiley & Sons, Inc., New York 1996.
25. Data Warehouse Models and OLAP Operations. Available from :
<http://www.inf.unibz.it/~franconi/teaching/2002/cs636/CS636-olap.ppt>. [Accessed January 10, 2006]
26. Mahidol University Computing Center. Technical Manual of Mahidol University E-Payment System, 2000.
27. Mahidol University Computing Center. Technical Manual of MU Curriculum Management System, 2000.
28. Mahidol University Computing Center. Web-based Registration System and Other Service System, 2000.
29. Mahidol University Computing Center. MU-UM-44-001 User Manual of Student Services:- Student Pre-enrollment System on the WEB, 2001.
30. Mahidol University Computing Center. MU-UM-44-002 User Manual of Student Services:- E-Payment System, 2001.
31. Mahidol University Computing Center. MU-UM-44-003 User Manual of Services Portal for Academic Administration Support System:-E-Registration System for STAFF, 2001.
32. Mahidol University Computing Center. MU-UM-44-004 User Manual of Student Services:- Student Pre-enrollment Approval and Invoice Processing on the WEB, 2001.
33. Mahidol University Computing Center. MU-UM-44-005 User Manual of Student Services:- Add/Drop/Withdraw/Leave Enrollment, 2001.
34. Mahidol University Computing Center. MU-UM-45-008 User Manual of Student Services:- Student File Maintenance, 2002.
35. Mahidol University Computing Center. MU-UM-45-014 User Manual of Student Services:- MU Student Reports, 2002.

36. Mahidol University Computing Center. MU-UM-45-019 User Manual of MU Academic Information Portal:- Curriculum File Maintenance, 2002.
37. Mahidol University Computing Center. MU-UM-45-020 User Manual of MU Academic Information Portal:- Course File Maintenance, 2002.
38. Mahidol University Computing Center. MU-UM-48-001 User Manual of Student Services:- Grading Report, 2005.
39. Mahidol University Computing Center. MU-UM-48-002 User Manual of Student Services:- Diploma System, 2005.



BIOGRAPHY



NAME	Miss Siriporn Rojanakosol
DATE	August 31, 1972
PLACE OF BIRTH	Phuket, Thailand
INSTITUTIONS ATTENDED	Kasetsart University, 1990-1994: Bachelor of Science (Agriculture) Mahidol University, 1996-1998: Bachelor of Science (Computer Science) Mahidol University, 2002-2006: Master of Science (Computer Science)
POSITION & OFFICE	1996 - Present, Mahidol University Computing Center, Bangkok, Thailand. Position : Computer Scientist