OLAP REPORTING APPLICATION
USING
OFFICE WEB COMPONENTS

A Thesis
Presented to
The Graduate Faculty of The University of Akron

In Partial Fulfillment
of the Requirements for the degree
Master of Science

Swathi R Kasireddy
August, 2007
OLAP REPORTING APPLICATION

USING

OFFICE WEB COMPONENTS

Swathi R. Kasireddy

Thesis

Approved:                        Accepted:

Advisor                          Dean of the College
Dr. Chien-Chung Chan             Dr. Ronald F. Levant

Committee Member                 Dean of the Graduate School
Dr. Xuan-Hien Dang               Dr. George R. Newkome

Committee Member                 Date
Dr. Zhong-Hui Duan

Department Chair
Dr. Wolfgang Pelz
ABSTRACT

This thesis describes the design and implementation of OLAP Reporting Application using Microsoft® Office Web Components. Online Analytical Processing (OLAP) is used in wide variety of business applications to analyze the data and this helps in querying multidimensional data. Using Microsoft Office Web Components we can build a web based OLAP Reporting Applications. Microsoft Office Web Components are a collection of ActiveX controls. These are used for adding spreadsheet, chart, and data-manipulation functionality to Web pages and for viewing the added components on the Web. The Office Web Components are programmable, we can use them to build rich and interactive web-based solutions in many design environments, including Microsoft FrontPage, Microsoft Access data access pages, and Microsoft Visual Basic. We can also publish the Office Web Components directly from Microsoft Excel. And the user cannot display these components in Microsoft Word like any of the other environments. In this thesis I designed a web based application which allows the user to connect to an OLAP data source and run queries. And the user can copy the components from the web page into the Microsoft Word using Object Embedding Technique which serves the above purpose. Application developers can reuse this code instead of doing it from scratch which reduces the time of coding.
DEDICATION

Dedicated to my
Sister Mrs. Swapna Reddy Kandimalla
And to my
Brother Mr. Chandra Sekhar Reddy Kasireddy
Who helped me in every step of my life
And to all my
Other family members and friends
Who offered me love and support throughout the course of this thesis
And especially to
GOD
Who created effects and preserved all things through his almighty power.
ACKNOWLEDGEMENTS

I have worked with a great number of people whose contribution in various ways to the thesis. It is a pleasure to convey my thanks to them all in my humble acknowledgment. In the first place I would like to convey my gratitude to Dr. Chien-Chung Chan for his supervision, advice, and guidance from the very early stage of this research as well as giving me extraordinary experiences through out the work. I was extremely fortunate in having Dr. Chien-Chung Chan as my professor. I could never have started all of this without his prior teachings in Data Mining Classes and thus opened up unknown areas to me. Thank you. Above all and the most needed, he provided me constant encouragement and support in various ways. His guidance specially inspired and enriched my growth as a student want to be. I am thankful to him more than he knows.

I want to thank my thesis committee members, Dr. Xuan-Hien Dang and Dr Zhong-Hui Duan, for their guidance and encouragement. I gratefully acknowledge the faculty members of the Department of Computer Science for their advice, supervision, and crucial contribution through out my course work.

I would also thank my family for their thoughtful support, contribution which enabled me to complete this work, thank you. Finally, I would like to thank everybody who was important to the successful completion of this thesis. Last but not the least I thank God for enabling me to succeed in my life.
# TABLE OF CONTENTS

LIST OF TABLES........................................................................................................... x

LIST OF FIGURES......................................................................................................... xi

CHAPTER

I. INTRODUCTION......................................................................................................... 1

1.1 What is Online Analytical Processing (OLAP)?.............................................. 1

1.1.1 12 Rules of OLAP.......................................................................................... 2

1.1.2 OLAP functionality......................................................................................... 3

1.2 Data Mining......................................................................................................... 3

1.2.1 Major Elements of Data Mining................................................................. 4

1.2.2 How the system satisfies Codd’s twelve rules of OLAP......................... 4

1.3 Problem Statement and the solution............................................................... 5

1.4 Route map.......................................................................................................... 6

II. MICROSOFT SQL SERVER 2000 ANALYSIS SERVICES, OFFICE WEB
COMPONENTS AND DESCISION SUPPORT OBJECTS....................................... 8

2.1 Overview............................................................................................................ 8

2.2 OLAP Cube....................................................................................................... 8

2.2.1 Dimensions.................................................................................................. 9

2.2.2 Measures..................................................................................................... 9
2.2.3 OLAP Databases ................................................................. 10
2.2.4 Data Sources ................................................................. 10
2.2.5 Partitions ................................................................. 10
2.2.6 Cube roles ............................................................... 11
2.2.7 Cube Structure ............................................................ 12
2.3 Analysis Manager .......................................................... 13
  2.3.1 Key features of Analysis Services ................................. 14
  2.3.2 Creating a Cube ......................................................... 15
2.4 How to Start Analysis Manager ........................................... 18
2.5 Creating the Database ..................................................... 18
2.6 Creating a Data Source ................................................... 19
2.7 Selecting the measures to build the cube ............................. 21
2.8 Adding dimensions and levels to the cube .......................... 24
2.9 Setting the storage options and setting up the cube aggregations ....... 29
2.10 Preprocessing ............................................................. 31
2.11 Browsing a Cube .......................................................... 34
2.12 Microsoft® Office Web Components (OWC) ......................... 35
  2.12.1 Why Office Web Components .................................... 35
  2.12.2 What are Microsoft Office Web Components? ................. 36
  2.12.3 What are the versions of Office Web Components? ......... 37
  2.12.4 Where can we download OWC from? ............................ 37
  2.12.5 What are the types of Microsoft Office Web Components? .. 37
  2.12.6 System Requirements for Microsoft Office Web Components... 42
2.12.7 Creating an Interactive Web Pages using Office Web Components……………………………………………………… 42
2.12.8 What about creating an Interactive Web Page with Microsoft Word? ………………………………………………… 43
2.12.9 Solution to insert Office Web Components into a Microsoft Word Document ………………………………………………………... 43
2.12.10 Difference between excel pivot table and OWC pivot table…… 44
2.13 Decision Support Objects………………………………………………… 47
2.13.1 Decision Support Objects Architecture……………………………….47
2.13.2 Classes and Interfaces ...................................................................48
III. OLAP REPORTING APPLICATION DESIGN........................................... 49
3.1 Structure of the OLAP Reporting Application System……………….50
3.1.1 Client Server Architecture ..............................................................52
3.2 About Login Page and Login Failed Page…………………………. 55
3.3 About Main Page .............................................................................56
3.3.1 Displaying Date........................................................................... 57
3.3.2 Displaying Welcome Message..........................................................57
3.3.3 About the Drop down Controls on the Page............................ 57
3.3.4 About the Pivot Table Component and Pivot Chart Component….. 61
3.3.5 About the Buttons on the page....................................................... 62
3.4 Installations Required and Solving Runtime Errors......................... 64
3.5 Conclusions...................................................................................... 65
IV. RESULTS.............................................................................................. 66
4.1 Software Requirements ................................................................. 66
## LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1 Illustrates how the system satisfies Codd’s twelve rules of OLAP</td>
<td>4</td>
</tr>
<tr>
<td>2.1 Summary of cube process options</td>
<td>33</td>
</tr>
<tr>
<td>2.2 Summary of Microsoft Office Spreadsheet Component toolbar</td>
<td>38</td>
</tr>
<tr>
<td>control functionalities</td>
<td></td>
</tr>
<tr>
<td>2.3 Summary of Microsoft Office Pivot Table Component toolbar</td>
<td>40</td>
</tr>
<tr>
<td>control functionalities</td>
<td></td>
</tr>
<tr>
<td>2.4 Summary of Excel Pivot Table toolbar control functionalities</td>
<td>45</td>
</tr>
<tr>
<td>3.1 Summarizes the values of the connection string</td>
<td>63</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Screenshot of the Analysis Manager showing Databases, Data sources, Cubes, Partitions and Cube Roles</td>
<td>11</td>
</tr>
<tr>
<td>2.2</td>
<td>Star Schema</td>
<td>12</td>
</tr>
<tr>
<td>2.3</td>
<td>Snowflake Schema</td>
<td>13</td>
</tr>
<tr>
<td>2.4</td>
<td>Screenshot of the Analysis Manager</td>
<td>15</td>
</tr>
<tr>
<td>2.5</td>
<td>Screenshot of ODBC Data Source Administrator dialogue box</td>
<td>16</td>
</tr>
<tr>
<td>2.6</td>
<td>Screen shot of creating new data source dialogue box</td>
<td>17</td>
</tr>
<tr>
<td>2.7</td>
<td>Screen shot of ODBC Microsoft Access Setup wizard</td>
<td>17</td>
</tr>
<tr>
<td>2.8</td>
<td>Snapshot of selecting a database dialogue box</td>
<td>18</td>
</tr>
<tr>
<td>2.9</td>
<td>Snapshot of the database dialog box</td>
<td>19</td>
</tr>
<tr>
<td>2.10</td>
<td>Snapshot of the data link properties dialogue box</td>
<td>20</td>
</tr>
<tr>
<td>2.11</td>
<td>snapshot of the data Connection tab dialogue box</td>
<td>21</td>
</tr>
<tr>
<td>2.12</td>
<td>Snapshot of fact table selection dialog box of cube wizard</td>
<td>22</td>
</tr>
<tr>
<td>2.13</td>
<td>Snapshot selection of measures dialog box of cube wizard</td>
<td>23</td>
</tr>
<tr>
<td>2.14</td>
<td>Snapshot of dimension select or dimension create dialogue box</td>
<td>23</td>
</tr>
<tr>
<td>2.15</td>
<td>Snapshot of Create the dimension and defining hierarchy dialogue box</td>
<td>24</td>
</tr>
<tr>
<td>2.16</td>
<td>Snapshot of selecting dimension table dialogue box</td>
<td>25</td>
</tr>
<tr>
<td>2.17</td>
<td>Snapshot of selecting levels for dimensions dialogue box</td>
<td>26</td>
</tr>
<tr>
<td>2.18</td>
<td>Snapshot of selecting advanced options dialogue box</td>
<td>26</td>
</tr>
</tbody>
</table>
2.19 Snapshot of set changing property dialogue box............................27
2.20 Snapshot of specifying storage mode and member groups dialogue box.. 27
2.21 Snapshot of dimension wizard finish dialogue box........................28
2.22 Snapshot of selecting the type of data storage dialogue box.................29
2.23 Snapshot of set aggregation options dialogue box............................31
2.24 Snapshot of storage design wizard finish dialogue box......................31
2.25 Screenshot of the “Process a cube” dialog box..............................33
2.26 Screenshot of the Process dialog box........................................34
2.27 Screenshot of the Cube Browser.................................................35
2.28 Snapshot of Microsoft Office Spread sheet component toolbar..........38
2.29 Snapshot of Microsoft Office Pivot Table Component dialogue box.....39
2.30 Snapshot of Microsoft Office Pivot Table Component toolbar.............40
2.31 Snapshot of Microsoft Office Pivot Table Component........................44
2.32 Snapshot of Excel pivot table.....................................................45
2.33 Snapshot of Excel Pivot Table toolbar .........................................45
2.34 Snapshot of DSO Object Model Hierarchy......................................47
3.1 Illustrates about the Structure of the System.................................57
3.2 Use Case Diagram of the System................................................57
3.3 Illustrates the user interaction with the OLAP Reporting System..........54
3.4 Illustrates about the Connect method to connect to the Analysis Manager...55
4.1 Screenshot of the login.aspx page................................................67
4.2 Screenshot of the loginfailed.aspx page.........................................67
4.3 Screenshot of Help.aspx page.......................................................68
4.4 Screenshot of rkowc.aspx page............................................................... 68

4.5 Snapshot of the message before connecting to the cube......................... 69

4.6 Snapshot of Field List............................................................................. 70

4.7 Results of the pivot table component and chart component with the Food Mart 2000 cube data and drill down operation.................................70
CHAPTER I

INTRODUCTION

Data is a key element in today’s Businesses. Data are any facts, numbers, or text that can be processed by a computer. The relationships among the data provide information. The information can be converted into knowledge. Organizations are accumulating vast and growing amounts of data in different formats and different databases. Businesses are facing a lot of problems in order to analyze their data. We have lots of data, but cannot make sense out of that data and turn it into useful information. And the solution for this is Data Mining which is also called knowledge discovery which allows organizations to take a wealth of data from disparate data sources and turn it into meaningful information. Online Analytical Processing (OLAP) and data mining are the essential tools for doing this [1, 2].

An OLAP and Data Mining tool helps managers and analysts in their process of decision making and giving them possibilities to discover hidden patterns in their data.

1.1 Online Analytical Processing (OLAP)

OLAP is a technology that allows users to examine business data and to get familiar with the information it contains. OLAP data model is multidimensional so it is
called a cube instead of a table. OLAP implies "digging through tons of data" to uncover patterns and relationships contained within the business activity and history. Data mining can be done with programs that analyze the data automatically. In order to better understand customer behavior and preferences, businesses are using data mining to pass through the huge amounts of information gathered via the Web [3].

1.1.1. Twelve Rules of OLAP:

E. F. Codd, father of the relational database, have produced the 12 rules for OLAP (on-line analytical processing) systems [5]. The twelve rules are as follows:

- Multi-Dimensional Conceptual View: Data should be presented to the user in a multi-dimensional pattern.
- Transparency: Users should not need to know that they are using an OLAP database.
- Accessibility: Tools should choose the best source of data to support a query.
- Consistent Reporting Performance: Performance should be the same regardless of the number of dimensions in use.
- Client-Server Architecture: Tools should be deployed in client server architecture.
- Generic Dimensionality: Dimensions are all equal; there should be no bias towards any one dimension.
- Dynamic Sparse Matrix Handling: Null values should be stored in an efficient way.
• Multi-User Support: Tools should support more than one user!

• Unrestricted Cross-Dimensional Operations: Aggregation rules should be applied consistently across all dimensions.

• Intuitive Data Manipulation: The user views of data should contain everything required without resorting to using menus or multiple trips across the user interface.

• Flexible Reporting: Users should be able to present data in any way they like.

• Unlimited Dimensions and Aggregation Levels: There should be no limit to the number of dimensions and levels in a model.

1.1.2. OLAP functionality

The functionalities of OLAP are calculations applied across dimensions, through hierarchies and/or across members. Drill-down to deeper levels, reach-through to underlying levels of data. OLAP storage unit is multidimensional so it is called a cube instead of a table. A multidimensional database that holds data more like a 3D spreadsheet rather than a relational database [6].

1.2 Data Mining

Data Mining is the process of finding meaningful relationships among data within very large databases. Data mining sometimes called knowledge discovery is the process of analyzing data from different perspectives and making it into useful information. [2, 28]
1.2.1. Major elements of Data Mining

Data mining consists of extracting, transforming, and loading data from different sources, storing and managing the data in a multidimensional database system, provide data access to business analysts and information technology professionals, analyze the data by application software, present the data in a useful format, such as a graph or table [2, 28].

1.2.2. How the system satisfies Codd’s twelve rules of OLAP

OLAP lets the user to create and manage multidimensional structures also called as cubes.

Table 1.1 Illustrates how the system satisfies Codd’s twelve rules of OLAP

<table>
<thead>
<tr>
<th>Rule</th>
<th>Satisfies or not</th>
<th>Explanation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-Dimensional Conceptual View</td>
<td>Yes</td>
<td>System can present the data in multidimensional pattern to the user</td>
</tr>
<tr>
<td>Transparency</td>
<td>Yes</td>
<td>Users should not need to know that they are using an OLAP database.</td>
</tr>
<tr>
<td>Accessibility</td>
<td>Yes</td>
<td>System chooses the best source of data from analysis manager to support the MDX query we write</td>
</tr>
<tr>
<td>Consistent Reporting Performance</td>
<td>Yes</td>
<td>System performance will be same regardless of the number of dimensions in use</td>
</tr>
<tr>
<td>Client-Server Architecture</td>
<td>Yes</td>
<td>System is deployed in client server architecture</td>
</tr>
<tr>
<td>Generic Dimensionality</td>
<td>Yes</td>
<td>Dimensions are equal when the system browse the cube data, there will be no bias towards any dimension</td>
</tr>
</tbody>
</table>
Table 1.1 Illustrates how the system satisfies Codd’s twelve rules of OLAP Continued

<table>
<thead>
<tr>
<th>Dynamic Sparse Matrix Handling</th>
<th>Yes</th>
<th>System does not deal with any null values but when sorting the values in ascending order through the button on the owc pivot table control, all the null values will be displayed first</th>
</tr>
</thead>
<tbody>
<tr>
<td>Multi-User Support</td>
<td>Yes</td>
<td>System allows multiple users to access the cube data</td>
</tr>
<tr>
<td>Unrestricted Cross-Dimensional Operations</td>
<td>Yes</td>
<td>System allows data roll ups through levels of a hierarchy</td>
</tr>
<tr>
<td>Intuitive Data Manipulation</td>
<td>Yes</td>
<td>The users views of data contain everything required and no need of multiple trips across menus or the user interface</td>
</tr>
<tr>
<td>Flexible Reporting</td>
<td>Yes</td>
<td>Users can present data in any way they like</td>
</tr>
<tr>
<td>Unlimited Dimensions and Aggregation Levels</td>
<td>Yes</td>
<td>There is no limit to the number of dimensions and levels in the system</td>
</tr>
</tbody>
</table>

1.3 Problem Statement and the solution

Microsoft® Analysis Services, installed with SQL server™ 2000, is the OLAP database engine and is able to build multidimensional cubes [3]. It also provides the application programs to browse the cube data. In order to use Analysis Services for browsing cube data users need to fully understand the cube structure, dimensions, measures and associated objects. This sometimes becomes critical for some users to understand the whole situation.
Analysis Services provides wizards and editors for developing OLAP cubes. It has been designed to be flexible for all users, but users have difficulty learning to use these features effectively. The solution for this is to design a front-end interface to meet the user’s requirements with the ability to evaluate data and to provide with good interface for avoiding the underlying complexities of the applications for the users. One of the aims of this thesis is to construct a client-application interface by using the Multi-dimensional Expressions (MDX), Microsoft® Office Web Components and Decision Support Objects (DSO) to query OLAP data to solve this problem. The Office Web Components are programmable; we can use them to build rich and interactive web-based solutions in many design environments, including Microsoft FrontPage, Microsoft Access data access pages, and Microsoft Visual Basic. We can also publish the Office Web Components directly from Microsoft Excel. And we cannot display these components in Microsoft Word and build interactive web-based solution like in the other environments. In this thesis we designed a web based application which allows the user to connect to an OLAP data source and run queries. And we can copy the components from the web page into the Microsoft Word using Object Embedding Technique which serves the above purpose.

1.4 Route Map

This thesis report covers the overall work on building ‘Olap Reporting Application Using Office Web Components’ for browsing the cube data. The thesis is organized as follows:
Chapter I provide an overview of Online Analytical Processing and Data Mining features, functionalities, rules and analysis techniques.

Chapter II gives a brief description about Microsoft SQL Server Analysis Services. The step-by-step processes for creation of an OLAP cube and browse the existing cube data. It also gives the brief idea about Microsoft® Office Web Components and Decision Support Objects (DSO).

Chapter III focus on the design and development of The OLAP Reporting Application using Office Web Components.

Chapter IV describes the implementations this application and results obtained with the implementation.

Chapter V presents a summary of the work that has been done in this thesis and also tells about the future work and the conclusion of this thesis.
CHAPTER II

MICROSOFT SQL SERVER 2000 ANALYSIS SERVICES, OFFICE WEB COMPONENTS AND DECISION SUPPORT OBJECTS

2.1. Overview

When we install Microsoft SQL server 2000, Analysis Services is installed with this and Analysis Manager which is shipped with Analysis Services is used for Analysis Server Administration [8, 29]. Microsoft SQL server 2000 Analysis Services supports OLAP which allows users to design, create, and manage multidimensional structures also called as cubes that contain data obtained from other data sources, such as relational databases [9].

2.2 OLAP Cube

The cube is the main OLAP structure which is used to view data. It is something similar to a table in the relational database system. The term cube refers to three dimensions in common, but here Analysis Services cube can have 128 dimensions [10]. A cube is a multidimensional structure that is defined by a set of dimensions and measures. Dimensions define the structure of the cube, while measures define the numerical values of interest to the user.
Each and every cube has a schema which is defined by a set of joined tables in the data warehouse from which the cube gets its source data. The middle table in the schema is the fact table, which is the source of the cube's measures. The other tables are dimension tables, which are the sources of the cube's dimensions. The minor alphanumeric values around the cube are the members of the dimensions. In MS SQL Server 2000 a cube can contain up to 128 dimensions, each with thousands and millions of members, and up to 1,024 measures. A cube with reserved number of dimensions and measures usually serves the requirements of users [26, 27].

Cubes right away follow the OLAP database in the object hierarchy. An OLAP database is a container for related cubes and the objects they share. We should create a database before we create a cube.

2.2.1. Dimensions

Dimensions are the structural attributes of the cube; a cube's dimensions are either private to the cube or shared with other cubes in the database. Private dimensions are created when the cube is created. Shared dimensions can be created before or during cube creation. The cube term generally implies three dimensions but here a cube can have up to 128 dimensions [26, 27].

2.2.2 Measures

In a cube, a measure is a set of values that are based on a column in the cube's fact table and are usually numeric. Measures are the central values of a cube. That is,
measures are the numeric data of main interest to users browsing a cube. The measures we select depend on the types of information users requests. The measures are created when the cube is created. A cube's measures are not shared with other cubes. A cube can have up to 1,024 measures [26, 27].

2.2.3 OLAP Databases

An OLAP database is a container for related cubes and the objects they share. These objects include data sources, shared dimensions, and database roles. If these objects are to be shared among multiple cubes, the objects and cubes must be within the same database. Databases are immediate objects to the Analysis server in the object hierarchy. Thus, after an Analysis server is installed, databases are the first objects to be created [26, 27].

2.2.4 Data Sources

After databases are created, data sources are usually the next objects that are to be created. A cube can have multiple data sources. They can be selected from the data sources in the database or created during cube creation. A cube's dimensions must have the same data source as the cube, but its partitions can have different data sources [26, 27].

2.2.5 Partitions

Partitions allow the source data of a cube to be distributed among multiple server computers. Each partition in a cube can have a different data source. These data sources
refer to relational databases on various computers. A single partition is automatically created for a cube when the cube is created [26, 27].

2.2.6 Cube roles

A Cube have cube roles on them, every cube must have at least one cube role in order to provide access to end users. Roles are used to control end users of having access to cube data or data mining models. Each cube role definition includes a list of Microsoft® Windows NT® 4.0 or Microsoft Windows® 2000 user accounts and groups and specifies the types and scope of access they are allowed. Cube roles can be created before or after cube creation. Cube roles are created after cube creation [26, 27].

Figure 2.1 illustrates the screenshot of Analysis Manager which shows Analysis Server, Databases, Data sources, Cubes, Partitions, Cube Roles.

![Figure 2.1 Screenshot of the Analysis Manager showing Databases, Data sources, Cubes, Partitions and Cube Roles.](image-url)
2.2.7 Cube Structure

A cube structure is defined by its measures and dimensions. They are derived from tables in the cube's data source. The set of tables from which a cube's measures and dimensions are derived is called the cube's schema. Every cube schema consists of a single fact table and one or more dimension tables. The cube's measures are derived from columns in the fact table. The cube's dimensions are derived from columns in the dimension tables. Cubes in the Analysis Services are built using one of two types of database schemas: the star schema and the snowflake schema [11]. Both schemas consist of a fact table and dimension tables. The Analysis Services aggregates data from these tables to build cubes. As shown in Figure 2.2, the star schema consists of a fact table and several dimension tables. Each dimension table corresponds to a column in the fact table. The data in the dimension tables are used to form the analytical queries in the fact table. Figure 2.3 shows the snowflake schema, in which several dimension tables are joined before being linked to the fact table. In simple words joining many star schemas we can form snowflake schema [26, 27]. The Example of star schema is shown in Appendix C and the example of snowflake schema is shown in Appendix D.

Figure 2.2 Star Schema
2.3 Analysis Manager

Microsoft® Analysis Services is shipped with SQL server™ 2000, and when we install Microsoft® SQL 2000 Analysis Services, Analysis Manager will also be installed as a tool for Analysis Server administration. To start Analysis Manager, we go to the Start button on the desktop, and then to the Programs group, from which we then need to point to the Microsoft® SQL Server™ 2000, then we can see the Analysis Services Group if we click Analysis Manager a console tree appears in the left pane of the management console, where we view the familiar hierarchical structure. Once inside Analysis Manager's console, we can see all the analysis servers established for our environment on the left-hand side. The analysis server name is automatically derived from the installation of Microsoft® SQL Server™ 2000, and is likely to be the name of the physical machine upon which each of us is working, provided defaults were accepted throughout the Microsoft® SQL Server™ 2000 installation [8].
2.3.1 Key features of Analysis Services [26, 27].

- Ease of Use: An extensive user interface with wizards
- Flexible Data Model: A flexible, robust data model for cube definition and storage. Supports various data and storage models like Multidimensional OLAP (MOLAP), Relational OLAP (ROLAP), Hybrid OLAP (HOLAP)
  - Multidimensional OLAP (MOLAP): The underlying data for a cube is stored along with aggregation data in a high-performance multidimensional structure. MOLAP storage provides excellent performance and data compression.
  - Relational OLAP (ROLAP): The underlying data for a cube is stored along with the aggregation data in a relational database. ROLAP storage enables you to take advantage of your investment in relational technology and enterprise data management tools.
  - Hybrid OLAP (HOLAP): The underlying data for a cube is stored in a relational database and the aggregation data is stored in a high-performance multidimensional structure. HOLAP storage offers the benefits of MOLAP for aggregations without necessitating duplication of the underlying detail data.
- Scalability: Scalable architecture that provides a variety of storage scenarios and an automated solution to the data explosion syndrome that plagues traditional OLAP technologies
- Integration: Integration of administration tools, security, data sources, and
client/server caching

- Widely Supported APIs and Open Architecture: Support for custom applications

Figure 2.4 illustrates the screenshot of the hierarchical, tree-view representation of the server and all its components in the left pane of the management console.

![Figure 2.4 Screenshot of the Analysis Manager](image)

2.3.2 Creating a Cube

Before we begin working with Analysis Manager, we must first set connections to the source of our data in the ODBC Data Source Administrator and we need to set up a
database structure. A database is a structure that holds cubes, roles, data sources, shared dimensions, and mining models together. Then you will need to connect to the data source that you set up earlier in the ODBC Data Source Administrator [26, 27].

1. Setting up the System Data Source Connection (DSN): The following are the steps to setup system data source connection (DSN).
   a. Click the Start button, point to Settings, click Control Panel, double-click Administrative Tools, and then double-click Data Sources (ODBC).

   ![Image of ODBC Data Source Administrator dialogue box]

   Figure 2.5 Screenshot of ODBC Data Source Administrator dialogue box

   b. Select Microsoft Access Driver (*.mdb), and then click Finish.
c. In the Data Source Name box, enter Swathi or some other name, and then under Database, click Select.
d. In the Select Database dialog box, browse to C:\Program Files\Microsoft Analysis Services\Samples, and then click FoodMart 2000.mdb. Click OK.

![Select Database](image)

Figure 2.8 Snapshot of selecting a database dialogue box

e. In the ODBC Microsoft Access Setup dialog box, click OK.

f. In the ODBC Data Source Administrator dialog box, click OK.

2.4 How to Start Analysis Manager

- Click the Start button, point to Programs, Microsoft SQL Server, and Analysis Services, and then click Analysis Manager.

2.5 Creating the Database

A database is a structure that holds cubes, roles, data sources, shared dimensions, and mining models together. The databases are shown in Figure 2.1. In the Analysis Manager tree view, expand Analysis Servers. Click the name of the server here it is SQLSERVER,
and then connection with the Analysis server will be established. Right-click the server's name and then click New Database. Then a Dialog Box appears as shown in Figure 2.9. In the Database dialog box as, in the Database name box, enter the desired database name, and then click OK. In the Analysis Manager tree pane, expand the server, and then expand the database you just created [26, 27], then the database contains the following items as shown in Figure 2.1:

- Data Sources
- Cubes
- Shared Dimensions
- Mining Models
- Database Roles

![Database dialog box](image)

Figure 2.9 Snapshot of the database dialog box

2.6 Creating a Data Source

After creating the database we have to create a data source. The data source contains the information of the data used in the cube. Setting up a data source in Analysis Manager connects our database to the system data source name (DSN) that we
have set up in the ODBC Data Source Administrator. All our data will come from this source as we build your cube. In the Analysis Manager tree pane, right-click the Data Sources folder under the database we have created earlier, and then click New Data Source. In the Data Link Properties dialog box, click the Provider tab [26, 27], and then click Microsoft OLE DB Provider for ODBC Drivers as shown in Figure 2.10.

![Image of Data Link Properties dialogue box]

which we have created as shown in Figure 2.11. Click Figure 2.10 Snapshot of the data link properties dialogue box

Now Click the Connection tab, and then from the Use data source name list, click the
Test Connection to confirm that everything works fine.

![Data Link Properties](image)

Figure 2.11 snapshot of the data Connection tab dialogue box

Click OK to close the Data Link Properties dialog box.

2.7 Selecting the measures to build the cube

A cube is a multidimensional structure of data. Cubes are defined by a set of dimensions and measures. In the Analysis Manager tree pane, under the database we have created, right-click the Cubes folder, click to New Cube, and then click Wizard. Measures are the quantitative values in the database that we want to analyze. Measures are analyzed against the different dimension categories of a cube. In the Welcome step of the Cube Wizard, click Next. In the Select a fact table from a data source step, expand the
data source created [26, 27], and then we should click the table we want as shown in Figure 2.12.

Figure 2.12 Snapshot of fact table selection dialog box of cube wizard

Now we can view the data in the table we have selected by clicking Browse data. After you finish browsing data, close the Browse data window, and then click Next.

To define the measures for our cube, under Fact table numeric columns, double-click
a column to add it as a measure [26, 27]. Repeat this procedure for the other measures we want to add and then click Next as shown in Figure 2.13.

![Figure 2.13 Snapshot selection of measures dialog box of cube wizard](image)

After this the Cube Wizard asks the user to select dimensions or to create dimensions as shown in Figure 2.14.

![Figure 2.14 Snapshot of dimension select or dimension create dialogue box](image)
2.8 Adding dimensions and levels to the cube

In the above wizard press New Dimension then a Welcome step dialog box will appear, click Next. In the Choose how you want to create the dimension step, select Star Schema which is a single dimension table or what ever schema you want to select as shown in Figure 2.15, and then click Next [26, 27].

![Dimension Wizard](image)

Figure 2.15 Snapshot of Create the dimension and defining hierarchy dialogue box

In the Select the dimension table step, click the table you want. You can view the data contained in the table by clicking Browse Data. When you are finished viewing the table, click Next as shown in Figure 2.16.
Now we have to define the levels for our dimension. In the select the levels for your dimension step, select the columns you want as shown in Figure 2.17 and then click Next.
Figure 2.17 Snapshot of selecting levels for dimensions dialogue box

Now select all from the advanced options dialogue box as shown in Figure 2.18

Figure 2.18 Snapshot of selecting advanced options dialogue box

26
In the set changing property dialogue box select ‘Yes, the new dimension is changing dimension’ as shown in Figure 2.19 and click Next.

![Figure 2.19 Snapshot of set changing property dialogue box](image)

When you click next in the above step it will go to specify storage mode and members group dialogue box as shown in Figure 2.20, in that select store as multidimensional OLAP (MOLP) and click Next [26, 27].

![Figure 2.20 Snapshot of specifying storage mode and member groups dialogue box](image)
After clicking Next we will be presented with the finish dialogue box then click Finish to return to the Cube Wizard. We can designate whether this dimension will be shared or private using the Share this dimension with other cubes check box, which is located on the lower left corner of the screen [26, 27]. Leave the box selected as shown in Figure 2.21.

![Figure 2.21 Snapshot of dimension wizard finish dialogue box](image)

After clicking Finish we will return to the cube wizard and there you will be asked to enter the desired cube name and click Finish, the cube will be created under the database we listed.
2.9 Setting the storage options and setting up the cube aggregations [26, 27]

- The storage mode determines how the data is organized in the server. Supports various data and storage models like Multidimensional OLAP (MOLAP), Relational OLAP (ROLAP), Hybrid OLAP (HOLAP).

The Storage Design Wizard is used to select the option for the cube in the Analysis Manager. Select whatever option you want from the dialogue box as shown in Figure 2.22.

![Storage Design Wizard](image)

**Select the type of data storage**

- **MOLAP**
  This option stores both the data and the aggregations in multidimensional structures.

- **ROLAP**
  This option leaves your data in its existing relational structure and stores the aggregation tables in the relational database.

- **HOLAP**
  This option leaves your data in existing relational tables and stores the aggregations in a multidimensional structure.

For information about these options, click Help.

Figure 2.22 Snapshot of selecting the type of data storage dialogue box
Click Next after selecting the type of data storage, then we will be presented with set aggregation option wizard, click in the option you want then press Next as shown in Figure 2.23

Aggregations are precalculated summaries of data that improve query response time by having the answers ready before the questions are asked. For example, when a data warehouse fact table contains hundreds of thousands of rows, a query requesting the weekly sales totals for a particular product line can take a long time to answer if the fact table has to be scanned to compute the answer. However, the response can be almost immediate if the summarization data to answer this query has been precalculated. Precalculation of summary data is the foundation for the rapid response times of OLAP technology.

Aggregations are stored in the multidimensional structure in cells at coordinates specified by the dimensions [26, 27].

Precalculation of all possible aggregations in a cube results in the fastest possible response time for all queries. However, the storage and processing time required for the aggregations can be substantial. Storage requirements depend not only on the number of dimensions and measures, but also on the number of levels in the dimensions and the number of members of each level [26, 27].
2.10 Preprocessing

After clicking Next in the above step we will be presented with a Finish storage design wizard, in that select process now option as shown in figure 2.24.

Figure 2.23 Snapshot of set aggregation options dialogue box

Figure 2.24 Snapshot of storage design wizard finish dialogue box
Click Finish and the cube will be processed as shown in the Figure 2.25.

After making structural changes to a cube, you must process the cube before attempting to browse its data. We need to process our cube after completing any of the following:

- Building the cube and designing its storage options and aggregations
- Changing the cube's structure (measures, dimensions, and so on) and saving the changes to the cube
- Changing the structure of a shared dimension used in the cube

Also, if data in the cube's data warehouse has been added or changed, processing is recommended in order to ensure accurate results when browsing the cube.

When you process a cube, the aggregations designed for the cube are calculated and the cube is loaded with the calculated aggregations and data. Processing a cube involves reading the dimension tables to populate the levels with members from the actual data, reading the fact table, calculating specified aggregations, and storing the results in the cube. After a cube is processed, users can query it.

In the Analysis Manager, there are three options to process a cube. These options are summarized in Table 2.1, and these three options can be selected in the “Process a Cube” dialog box, as shown in Figure 2.25
Table 2.1 Summary of cube process options

<table>
<thead>
<tr>
<th>Options of Process</th>
<th>Circumstances</th>
</tr>
</thead>
<tbody>
<tr>
<td>Full process</td>
<td>Modifying the structure of the cube</td>
</tr>
<tr>
<td>Incremental update</td>
<td>Adding new data to the cube</td>
</tr>
<tr>
<td>Refresh data</td>
<td>Clear out and replacing a cube’s source data</td>
</tr>
</tbody>
</table>

Figure 2.25 Screenshot of the “Process a cube” dialog box
2.11 Browsing a Cube

Using Cube Browser, we can look at data in different ways: we can filter the amount of dimension data that is visible, we can drill down to see greater detail, and we can drill up to see less detail [26, 27].

- In the Analysis Manager tree pane, right-click the desired cube, and then click Browse Data.
- Then the cube Browser appears which displays the cube as a flat grid as shown in Figure 2.27

Figure 2.26 Screenshot of the Process dialog box
The cube browser allows the user to drill up and drill down into deeper levels of data. Browsing the cube using Analysis Manager is shown in Appendix B and browsing using OLAP Reporting Application is shown in Appendix A.

2.12 Microsoft® Office Web Components (OWC)

Microsoft Office Web Components are a collection of Component Object Model (COM) controls for publishing spreadsheets, charts, and databases to the Web.

2.12.1 Why Office Web Components?

Business people are turning to intranets and internet to share information with one another and with customers. In the past only highly technical Web masters used to
understand how to create and publish Web pages. Other users were limited to reading what Web masters published. This is changed with the introduction of some products such as the Microsoft® FrontPage® Web site creation and management tool which made it possible for the people to create and share documents on the Web.

Microsoft Office 2000 has taken a step forward. Actually the Web browsers can’t sort, filter, or recalculate totals on Web pages, these all can be possible by using Microsoft Office Web Components. Microsoft Access and Excel users share their documents on the corporate intranet and still preserve the interactivity that adds so much value to the information because of these office web components [12, 13].

2.12.2 What are Microsoft Office Web Components?

• Microsoft Office Web Components are a collection of Component Object Model (COM) controls for publishing spreadsheets, charts, and databases to the Web. They are also used to view these items when published [14].

• If we have Microsoft Office FrontPage 2003, Microsoft Office Access 2003, and Microsoft Office Excel 2003 installed, Office Web Components allows us to publish interactive data as part of a Web page. Office web components are used with Microsoft Internet Explorer version 5.01 or later to view a published control (spreadsheet, chart, or database) on a Web page and to view data access pages [14].

• When we use Internet Explorer to browse a Web page that contains an Office Web Component, we can interact with the page right in our browser—we can
sort, filter, enter values for formula calculations, expand and collapse details, pivot, and so on. The COM controls provide the interactivity [12, 13].

- By installing the Office Web Components tool, users can view published components and data access pages on the Web without having to install Office 2003 [14].

2.12.3 What are the versions of Office Web Components?

- The Microsoft Office Web Components are a set of controls released by Microsoft. The latest version, O.W.C. version 11, of the components was released with Microsoft Office 2003. Version 10 was released with Microsoft Office XP and Version 9 was released with Microsoft Office 2000.

- In this work am using Microsoft Office Web Components Version 10.

2.12.4 Where can we download OWC from?

- We can download Office XP Web Component Tool pack from any Microsoft Download Center or from the reference link [15].

2.12.5 What are the types of Microsoft Office Web Components?

The Microsoft Office Web Components include a spreadsheet, a PivotTable dynamic view, a data source, and a chart [12].

- Spreadsheet

The spreadsheet component provides a recalculation engine, a full-function library, and a simple spreadsheet user interface in Web pages. Calculations can refer to spreadsheet cells or to any control on the page or URL via the Microsoft
Internet Explorer document object model. Office 2000 users create Web pages with spreadsheet components by saving Excel workbooks as Web pages and by selecting the option to publish the page interactively [12].

Spreadsheet component looks similar to the Excel spreadsheet, Figure 2.28 shows the snapshot of Spreadsheet component toolbar.

![Figure 2.28 Snapshot of Microsoft Office Spreadsheet component toolbar](image)

Table 2.2 Summary of Microsoft Office Spreadsheet Component toolbar control functionalities

<table>
<thead>
<tr>
<th>Buttons</th>
<th>Behavior in Spreadsheet Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Undo</td>
<td>Can undo the following operations: sorting, applying a filter, remove a filter, applying a number format, changing a cell's value, pasting, changing the font formatting (font style, size, bold, italic, underline, color), changing the cell formatting (fill, alignment), clearing a cell, and inserting a column or row.</td>
</tr>
<tr>
<td>Cut, Copy, Paste</td>
<td>Supports cutting, copying, or pasting HTML and text to the Clipboard, but does not support cutting, copying, or pasting BIFF, RTF, or other formats.</td>
</tr>
<tr>
<td>AutoSum</td>
<td>Functions the same as in Excel.</td>
</tr>
<tr>
<td>Sort Ascending, Sort Descending</td>
<td>Functions the same as in Excel. The shortcut menu command cascades to a Sort menu that lists field headings to sort by. There is no Sort dialog box as in Excel.</td>
</tr>
<tr>
<td>AutoFilter</td>
<td>Contains check boxes in a drop-down list so that multiple items can be selected.</td>
</tr>
<tr>
<td>Export To Excel</td>
<td>Opens Microsoft Excel and copies all the data from the Spreadsheet Component to a new, read-only worksheet that has a name in the format of OWCSheet####.htm where #### is a random number. There are 65,536 rows as in Excel, but there are 676 columns going from A to ZZ, instead of the 256 columns available in Excel. If the extra columns are used and the data is exported back to Excel, the data in the extra columns is lost.</td>
</tr>
<tr>
<td>Property Toolbox</td>
<td>Displays the Spreadsheet Component's Property Toolbox</td>
</tr>
<tr>
<td>Help</td>
<td>Displays Microsoft Spreadsheet Help.</td>
</tr>
</tbody>
</table>
• PivotTable

The PivotTable dynamic views component enables users to analyze information by sorting, grouping, filtering, outlining, and pivoting. The data can come from a spreadsheet range, from a relational database (such as Microsoft Access or Microsoft SQL Server™ database), or from any data source that supports multidimensional OLEDB (such as Microsoft Decision Support Server). When an Excel user saves a PivotTable or QueryTable dynamic view as an interactive Web page, the page contains a PivotTable component. Web pages with PivotTable components can also be designed directly in the Access Data Access Pages designer [12].

The pivot table component looks as shown in Figure 2.29

The PivotTable Component toolbar

![Microsoft Office PivotTable List](image)

Drop Filter Fields Here

Drop Column Fields Here

Drop Row Fields Here

Drop Totals or Detail Fields Here

Figure 2.29 Snapshot of Microsoft Office Pivot Table Component dialogue box

In the above figure the arrow shows the pivot table component toolbar which is shown briefly in Figure 2.30
The Table 2.3 outlines what the controls on the PivotTable Component toolbar do [18].

Table 2.3 Summary of Microsoft Office Pivot Table Component toolbar control functionalities

<table>
<thead>
<tr>
<th>Control</th>
<th>Control type</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>About</td>
<td>Button</td>
<td>Displays About Microsoft Office Web Components dialog box.</td>
</tr>
<tr>
<td>Copy</td>
<td>Button</td>
<td>Copies selected data to the Clipboard.</td>
</tr>
<tr>
<td>Sort Ascending, Sort Descending</td>
<td>Toggle</td>
<td>When turned on, sort’s column or row fields in ascending or descending order. When turned off, data reverts to the way it was saved. Unlike PivotTable reports in Excel, blanks sort to the top.</td>
</tr>
<tr>
<td>AutoFilter</td>
<td>Toggle</td>
<td>When turned on, hides or shows items according to whether their check boxes have been cleared or selected in the Column or Row field’s drop-down list. When turned off, all items are displayed.</td>
</tr>
<tr>
<td>AutoCalc</td>
<td>Menu</td>
<td>Gives you a choice of summarizing fields by using the Sum, Count, Min, or Max functions. Can be disabled by the creator of the PivotTable list.</td>
</tr>
<tr>
<td>SubTotal</td>
<td>Toggle</td>
<td>Shows or hides subtotals for selected field.</td>
</tr>
<tr>
<td>Move to Row Area</td>
<td>Button</td>
<td>Moves selected column field to row area (selection becomes a row field).</td>
</tr>
<tr>
<td>Move to Column Area</td>
<td>Button</td>
<td>Moves selected row field to column area (selection becomes a column field).</td>
</tr>
<tr>
<td>Move to Filter Area</td>
<td>Button</td>
<td>Moves selected field to Filter field area (selection becomes a filter). Can be restricted by the creator of the PivotTable list.</td>
</tr>
<tr>
<td>Move to Detail</td>
<td>Button</td>
<td>Moves selected field to Detail (data) area. Fields cannot be moved if either of the following is true: the creator of the PivotTable list restricted access to data area or the PivotTable list is based on source data from an OLAP database.</td>
</tr>
<tr>
<td>Promote</td>
<td>Button</td>
<td>Moves selected field to the next outer level (away from the data area).</td>
</tr>
</tbody>
</table>
Table 2.3 Summary of Microsoft Office Pivot Table Component toolbar control functionalities continued.

<table>
<thead>
<tr>
<th>Demote</th>
<th>Button</th>
<th>Moves selected field to the next inner level (closer to the data area).</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expand/Collapse</td>
<td>Toggle</td>
<td>Expands or collapses the data detail for selected field, item of data, or cell in data area. Unlike in Excel, there is only one button to toggle between Expand and Collapse. In Excel, these are separate buttons.</td>
</tr>
<tr>
<td>Refresh</td>
<td>Button</td>
<td>Updates data from source list or database.</td>
</tr>
<tr>
<td>Export to Excel</td>
<td>Button</td>
<td>Starts or switches to Excel and copies data into a new workbook as a PivotTable report. Should be used only when further analysis is needed. To place data in an Excel spreadsheet, use Copy.</td>
</tr>
<tr>
<td>Property Toolbox</td>
<td>Button</td>
<td>Displays the PivotTable Component's Property Toolbox.</td>
</tr>
<tr>
<td>Field List</td>
<td>Button</td>
<td>Displays a hierarchical list of fields available from the PivotTable list's source data.</td>
</tr>
<tr>
<td>Help</td>
<td>Button</td>
<td>Displays Microsoft PivotTable List Help.</td>
</tr>
</tbody>
</table>

- Data Source

  The data source component is the reporting engine behind Data Access Pages and the PivotTable component. It manages communication with back-end database servers and determines which database records can be displayed on the page. It relies on Microsoft Active Data Objects (ADO) for plumbing, and like all the Office Web Components, it is fully programmable [12].

- Chart

  The chart component graphically displays information from the spreadsheet, from the PivotTable dynamic views, or from the data source component. Because it is bound directly to other controls on the page, it updates instantly in response
to user interactions with the other components. When an Excel user saves a workbook containing a chart as an interactive Web page, the page contains a chart component. Office Web Component charts can also be created and edited directly in the Access Data Access Pages designer [12].

The toolbar at the top of the chart provides a set of commands for working with the chart display and chart data. The toolbar is most useful when your chart is based on data from a PivotTable list or database table or query, and you want to sort, filter, or change the display of data in a chart [].

### 2.12.6 System Requirements for Microsoft Office Web Components

The Office Web Components require Microsoft Internet Explorer Version 4.01 or greater running on Microsoft Windows® 95, Windows 98, or Windows NT 4.0® operating system. Hardware requirements are 16 megabytes of RAM and any Intel 486 or Pentium Processor, or any DEC Alpha Processor. To design a component-based page with Access, or browse a page created with Access, you must have Microsoft Internet Explorer 5.x or greater [12].

### 2.12.7 Creating an Interactive Web Pages using Office Web Components

1. Creating an interactive web pages with Microsoft Excel [12]

   - Users create interactive Web pages from Excel by selecting the Save As Web Page from the Excel File menu.
   - When published interactively, the .htm page that is generated contains OBJECT tags that refer to the Office Web Components.
• When a user browses the page with Microsoft Internet Explorer 4.0 or 5.0, the controls are instantiated in place, providing interactive regions inside the page.

2. Creating an Interactive Web Page with Microsoft Access [12]

• Interactive Web pages can also be designed directly in the Data Access Pages designer in Microsoft Access. Users can add Office Web Components to a Web page by dragging and dropping from the Toolbox.

• The Data Access Pages designer contains the Office 2000 Field List, which allows users to build up complex data pages from databases without first building complicated SQL statements.

3. Creating an Interactive Web Page with Microsoft FrontPage [12].

• In Microsoft FrontPage 2000 office web components can be inserted from the Insert Tab on the menu bar and click component and select the component we want.

2.12.8 What about creating an Interactive Web Page with Microsoft Word?

• In Microsoft word any document could be made a web page by selecting ‘Save as Web page’ option from File menu. But we cannot insert Office Web Components into a word document like it is done in other environments.

2.12.9 Solution to insert Office Web Components into a Microsoft Word Document.

• The solution to insert Office Web Components into a Microsoft Word Document is to create an interactive web page with any of the above three methods or using programming language like VB, C# or scripting languages
like VB Script or Java etc to create an interactive web page with Office Web Components and copy the Web components from the web page into clipboard and pasting it in the word document or pasting them as an image, there is a problem if we paste them as an image that is they will not be interactive. This type of pasting is called as Object Embedding and we can publish the document as the web page \[19\].

2.12.10 Difference between excel pivot table and OWC pivot table

As we know that we can export the OWC pivot table into Excel and the difference between the actual OWC pivot table and the pivot table exported to excel is OWC pivot table has all the functionalities as shown in Figure 2.30 and as described in Table 2.3, but the pivot table exported to excel will lacks some of those.

The OWC pivot table with data looks as shown in Figure 2.31 and the exported pivot table into excel looks as shown in Figure 2.32

![MS Excel Pivot Table](attachment:image.png)

Figure 2.31 Snapshot of Microsoft Office Pivot Table Component
Figure 2.32 Snapshot of Excel pivot table

The excel pivot table toolbar is shown in Figure 2.33

Figure 2.32 Snapshot of Excel pivot table toolbar

Table 2.4 Summary of Excel Pivot Table toolbar control functionalities.

<table>
<thead>
<tr>
<th>Control</th>
<th>Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pivot Table Drop-Down</td>
<td>This displays the full list of pivot table commands</td>
</tr>
<tr>
<td>Format Report</td>
<td>Displays a dialog box which allows the users to select the format they want, something like templates for the report</td>
</tr>
<tr>
<td>Pivot Chart</td>
<td>Starts the chart wizard, which guides the users through the steps for creating an embedded chart on a worksheet or modifying an existing chart</td>
</tr>
<tr>
<td>Hide Detail</td>
<td>In a pivot table or pivot chart report, it hides the displayed detail data. On an outlined worksheet, hides the detail rows or columns of a selected summary row or column. Hides the items in a group.</td>
</tr>
</tbody>
</table>
Table 2.4 Summary of Excel Pivot Table toolbar control functionalities continued

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Show Detail</td>
<td>In a pivot table report, it displays detail data if available for the selected cell or selected field/item. On an outlined worksheet, displays the hidden detail rows or columns of a selected summary row or column. Shows all the hidden items in the pivot table.</td>
</tr>
<tr>
<td>Include Hidden Items in Totals</td>
<td>Lets us to show hidden items in totals</td>
</tr>
<tr>
<td>Always Display Items</td>
<td>Always shows the field item buttons with drop-down arrows in the pivot table.</td>
</tr>
<tr>
<td>Refresh Detail</td>
<td>Updates the data in a pivot table or a pivot chart report if the source data has changed.</td>
</tr>
<tr>
<td>Field Settings</td>
<td>Displays a dialog box allowing users to change settings, such as subtotals, sort options, and layout options, for the selected field in the pivot table or pivot chart report</td>
</tr>
<tr>
<td>Show Field list</td>
<td>Displays or hides the pivot table or pivot chart field buttons on the pivot table toolbar. Also, in a pivot table report, displays or hides the layout diagram outlines and labels.</td>
</tr>
<tr>
<td>Group/Ungroup</td>
<td>Enables users to group or ungroup Microsoft Excel pivot table formats to the users liking.</td>
</tr>
<tr>
<td>Refresh Data</td>
<td>Refreshes the data</td>
</tr>
</tbody>
</table>

As mentioned earlier the Excel pivot table lacks some functionalities such as Sort Ascending, Sort Descending, Auto filter, Show Top/Bottom Items and Show as. The main difference between the OWC pivot table and Excel pivot table is additional fields can be added by dragging and dropping from the field list in both of them. In the OWC pivot table we can remove the fields we want by right clicking on the field name and clicking remove field but in Excel pivot table we can only add fields and we cannot delete any fields.
2.13 Decision Support Objects

Decision Support Objects (DSO) is a library of Component Object Model (COM) classes and interfaces that provide access to the Analysis server [20].

2.13.1 Decision Support Objects Architecture

DSO uses hierarchically arranged groups of objects to define basic elements of Analysis Services data storage, as implemented by the Analysis server. These basic elements are databases, data sources, dimensions, cubes, data mining models, and roles. DSO maintains these basic elements in a hierarchical structure where elements contain other elements in a tree, with the server object at the root of the tree. Other objects support this basic structure. The Figure 2.34 shows DSO object model hierarchy [20-22].

![Figure 2.34 Snapshot of DSO Object Model Hierarchy](image)

The DSO Server object contains a collection that defines databases accessed by the server. Each database can contain groups of objects that define cubes, linked cubes, or virtual cubes. A cube contains one or more partitions, which contain one or more aggregations. Linked cubes serve to provide local server access to a cube on another
server; the remote server publishes the cube, and the local server subscribes to it by creating a linked cube. A virtual cube is a special case of a cube, combining portions of the cubes it contains, similar to the way a relational database view combines portions of tables. A database can also contain one or more relational or OLAP data mining models, represented in DSO by the MiningModel object. Databases also can contain roles, used to manage security on the database and its associated cubes and data mining models [20].

The Decision Support Objects (DSO) library supplies a hierarchical object model for use with any development environment that can support Component Object Model (COM) objects and interfaces, such as Microsoft Visual C++®, Microsoft Visual Basic®, and Microsoft Visual Basic Scripting Edition [21].

2.13.2 Classes and interfaces

In the Figure 2.34 the two most important collections of objects namely databases and cubes are supplied with a collection called MDStores that provides the methods and properties we use to manipulate the objects. The DSO Database, Cube and Partition objects implement the MDStore class interface as well as their own class interface. The main class used in the project is ‘rkowc’, in this class the server object, database object and cube object are defined which uses the methods and properties provided by the interface MDStore.

This project uses Microsoft Visual Basic® to retrieve database names and cube names into drop downs using Decision Support Objects.
CHAPTER III

OLAP REPORTING APPLICATION DESIGN

Microsoft® Analysis Services is the OLAP database engine which is able to build multidimensional cubes [3]. It also provides the application programs to browse the cube data. In order to use Analysis Services for browsing cube data users need to fully understand the cube structure, dimensions, measures and associated objects. This sometimes becomes critical for some users to understand the whole scenario. The solution for this is to design a front-end interface to meet the user’s requirements with the ability to analyze data and to provide with flexible interface for avoiding the underlying complexities of the applications for the users.

One of the aims of this thesis is to construct a client-application interface by using the Multi-dimensional Expressions (MDX), Microsoft® Office Web Components and Decision Support Objects (DSO) to query OLAP data to solve this problem which satisfies the twelve rules of OLAP introduced by Codd.

The complex and powerful underlying features of Analysis Services can be accessed through the Decision Support Objects (DSO) library, which supports a robust COM object model providing complete control of all Meta data in Analysis Services [22].
Decision Support Objects (DSO) is a library of Component Object Model (COM) interfaces and classes that provide access to the Analysis server. These classes and interfaces, when used together, form an object model that corresponds to the internal structure of the objects managed by Microsoft® SQL Server™ 2000 Analysis Services and can be used to manage them programmatically [20].

This chapter will introduce the OLAP Reporting Application developed for viewing the cube in Microsoft SQL server 2000 environment.

3.1 Structure of the OLAP Reporting Application System

Here in this project we have a series of pages which when satisfies a condition goes to another page. The series is illustrated below.

Here in this project first we have a login page, which when logged on goes to the main page if the login succeeds or else redirects to the login failed page if the login fails. This Structure is illustrated in Figure 3.1.

The UML Use Case Diagram for the system structure is drawn and it is shown in Figure 3.2. Use Case diagram tells us about the overview of the usage requirements for the system. After redirecting to the main page we have three buttons namely Help, Connect and Contact Us. If we press Help button we will be redirected to the Help Page. If we press Connect button it takes the data from text box and drop down boxes on the main page and connects to the analysis manager and displays the output this functionality is shown in Figure 3.4. If we press Contact Us button it opens a mail editor from which we can send a mail to the author.
Figure 3.1 Illustrates about the Structure of the System.

Figure 3.2 Use Case Diagram of the System
3.1.1 Client Server Architecture

The architecture of this system is divided into two parts: the client architecture and the server architecture. The server architecture includes the engines and provides the functionality to access the analysis services and the client architecture includes the front end interface.

The connection to the analysis server is illustrated in the Figure 3.3. It shows how the OWC pivot table interacts with the OLAP Cube. Here in the system OWC pivot table component is linked to the OWC Chart Component, whenever the data in the OWC Pivot table is changed the chart changes accordingly.

1. Server Requirements

The server requirements for the OLAP Reporting Application using OWC are:

- Microsoft Office web components version 10 running on Microsoft Windows® 95, Windows 98, or Windows NT 4.0® operating system.
- 16 megabytes of RAM and any Intel 486 or Pentium Processor, or any DEC Alpha Processor.
- Internet Explorer Version 4.01 or greater.
- Microsoft Data Access Components (MDAC) 2.6 or higher
- IIS 5.0 or later
- Microsoft .NET Framework 1.1
- Connection to SQL Server 2000 SP3a and SP4 instance
- Appropriate user account credentials to access the analysis services.
2. Client Requirements

There are two kinds of client requirements in this project the first one is accessing the OLAP Reporting Application using OWC from client browser and the other is accessing the word document which is created from copying and pasting the office web components from client machines.

The client requirements for accessing the OLAP Reporting Application from client browser are:

- Internet Explorer Version 4.01 or greater.
- No need to install office web components on the client machine. When the user first run the application on the client, the office web components are installed by themselves if they are not present on the machine.
- This application is not compatible with Mozilla Firefox.

The client requirements for accessing the word document are:

- The office web components are compatible in Word 2003 or later versions of Microsoft Word.
- In order to access the word document from client machine, the client should be running an instance of the OLAP Reporting Application in order to connect to the data source.
- Internet Explorer Version 4.01 or greater.
Figure 3.3 Illustrates the user interaction with the OLAP Reporting System

First connection to the analysis services is established through the connection string. After the construction of the connection string, the application uses an object called DataMember to connect to the OLAP cube which is required by the user. After connecting to the OLAP cube, the application uses an object called CommandText to retrieve the OLAP cube data into the OWC pivot table and it links the chart space to the pivot table through an object called DataSource with which the OWC chart object gets its source data.
3.2 About Login Page and Login Failed Page

Here in this project we have a login page which can allow only one user with a single username and password. In future we can enhance this to allow many users and can include registration process in the page. For this purpose I used Java Script. The Login page has two text boxes one for user name and other for password. When we enter username and password and click the Login button then we are directed to the main page.
Here if we enter user name as “test” and password as “password” then we will be directed to main page which is rkowc.aspx and if we enter anything other than the above username and password we will be directed to the loginfailed.aspx page which says that our login failed and that page has a button which again redirects to the login page to login again.

3.3 About Main page (rkowc.aspx page)

The main page has time displayed on it and the time displayed is taken according to the system clock and it also displays the welcome message according to the time. It has a Microsoft Office Pivot Table Component and Microsoft Office Pivot Chart Component. It also has one text box which asks for the server name something like “localhost” or the ip address of the computer name where we have analysis server something like “xxx.xxx.xx.x” and two drop downs in which one displays the database names from the analysis services. If the user selects anything from the first drop down list then the second drop down list is populated with the cube names related to the selected database. This is done using Decision Support Objects (DSO). The page has three buttons one is help button which when pressed says about the controls on the web page and another is the contact me button which when pressed opens an Outlook file which will have email address specified to send a mail to the author of the program if they have any problems in understanding the code. The other button is the connect button which when pressed takes the server name from the text box and the database name and cube name from the drop downs and connects to the Analysis Services and gets the OLAP cube data and fills in the Microsoft Office Pivot Table Component and gives out the chart related to that values in the pivot chart component.
3.3.1 Displaying Date

The main page displays the time which is taken from the system clock. The code for this is written in VBScript. Method used to get the system time here is `time()`.

`Time()`

This method is used to display the system time.

3.3.2 Displaying Welcome Message

The main page also displays the welcome message according to the time some thing like “Good Morning”, “Good Afternoon” and so on. The code for this is written in Java Script. Method used to get the system time here is `now.getHours()`.

`getHours()`

This function takes the system time that is exact hours from the system and used to display that or anything else according to that.

3.3.3 About the Drop down Controls on the Page

We have two drop down list boxes on the page one for retrieving the database names and the other for retrieving the cube names from the analysis server.

To retrieve the database names and the cube names I used Decision Support Objects (DSO). As described earlier Decision Support Objects (DSO) is a library of Component Object Model (COM) interfaces and classes that provide access to the Analysis server.
DSO Hierarchy

DSO uses hierarchically arranged groups of objects to define basic elements of Analysis Services data storage, as implemented by the Analysis server. These basic elements are databases, data sources, dimensions, cubes, data mining models, and roles. DSO maintains these basic elements in a hierarchical structure where elements contain other elements in a tree, with the server object at the root of the tree. Figure 2.34 shows an overview of the DSO object model hierarchy [20-22].

DSO object model hierarchy consists of a server object that represents the OLAP Server service that we want to use, an MDStores collection that represents the objects within the server, a Dimensions collection for describing the cube, Levels collections within a dimension, and Measures collections for the facts about what's in the cube.

Here in the project first to retrieve database names and cube names into the drop down list boxes we need to first open connection to server using the Server object which allows us to access and control the OLAP server itself. This object is mainly used to establish a session with the OLAP Server through its Connect method. The only argument necessary is the name of the server. To do this first we need to define the server object, database object and cube object in the class and then open the connection in the page_load method and it is written in VB.NET like as shown below

Defining Server Objects

To define the server object, database object and cube object

Dim dsoServer As New DSO.Server

Dim dsoDB As DSO.Database
Dim dsoCube As DSO.MDStore

To open connection

    To open the connection with the server we have to include the following code

dsoServer = CreateObject("DSO.Server")

dsoServer.CloseServer()

dsoServer.Connect("SQLSERVER")

    Here SQLSERVER is the name of the SQL Server. Before opening the
connection to the SQLSERVER, dsoServer.CloseServer () is included because it ends the
session and releases all resources and objects used through that server first.

Adding Database names to drop down lists

    After opening the connection to the SQLSERVER we have to add database names
to the drop down list box like as shown below

' Step through the databases in the server object.

    For Each dsoDB In dsoServer.MDStores

        ' Print the name of the database into drop down box

            cboDbName.Items.Add(dsoDB.Name)

    Next

After the above method we have to get the cube names into the second drop down list
when some database is selected in the first drop down, to do this first we have to clear all
the items in the second drop down list as shown below
Clearing the Cubes List

    listcubes.Items.Clear() 'Clearing the cube list drop down

    If any database doesn't have any cubes then the cube drop down list box displays
    "none" else it displays the cube names in the drop down list box, then at last we have to
    close the server connection.

    Dim strDataBaseName As String
    strDataBaseName = cboDbName.SelectedValue
    If dsoServer.MDStores.Find(strDataBaseName) Then
        dsoDB = dsoServer.MDStores.Item(strDataBaseName)
        If dsoDB.MDStores.Count = 0 Then
            listcubes.Items.Add("None")
            ' Debug.Print "  Cube: None"
        Else
            'Adding cube names to a drop down list box
            For Each dsoCube In dsoDB.MDStores
                listcubes.Items.Add(dsoCube.Name)
            Next
        End If
    End If
    dsoServer.CloseServer()
End Sub
3.3.4 About the Pivot Table Component and Pivot Chart Component

The page has two components one is pivot table components and the other is pivot chart component. Depending on the development environment, several iterations of these controls might be available. The versions of these controls will depend on the versions of Microsoft Office installed on the local machine (i.e., Microsoft Office Chart 11.0, 10.0, 9.0). These version numbers correspond with the build numbers of Office [17].

To use the Pivot Table or Pivot Chart Web component, the controls must be added to the toolbox. The controls can be found as a COM Components object and be added using the Add\Remove Items menu from the toolbox context menu. Once the control has been added, it can be dragged on to the Web form or aspx page. This action places an <OBJECT> tag representing the Pivot Table control or pivot chart control into the HTML.

CLASSID

One interesting element about the Pivot Table and Chart object is the ability to change the version of the object depending on the CLASSID attribute. Changing the CLASSID to correspond to the version of the pivot control desired is an easy way of changing compatibility to accommodate clients [17].

The CLASSID for the pivot table is classid="clsid:0002E552-0000-0000-C000-000000000046" and for the pivot chart component is classid="clsid:0002E556-0000-0000-C000-000000000046".
3.3.5 About the Buttons on the page

Firstly we have three buttons on the page namely Help, Connect, Contact Me. These three have three functionalities as explained below.

Help Button

Help button when presses redirects the control to the help page which tells about how to use the controls on the Page.

Contact Me Button

Then Contact Me button which when pressed open an Outlook file which has the email address already specified of the author of the program which allows the user to send a mail to the author if they have any problem in understanding the project.

Connect Button

Lastly, the Connect button which when pressed takes the input from the server name text box and the selected values from the two drop down boxes namely database names and cube names and then it connects to the Analysis Server and retrieves the cube data and fills in the Pivot Table component and then according to the data in the pivot table component, Pivot Chart Component is also drawn.

To connect to the cube data in the Analysis Server first we need to do the following:

a. Setting up the connection string and connect to the server
b. Displaying the results.

Setting up the Connection String and Connect to the server

In order to get the cube data the initial step is to set up the connection string
which connects to the Analysis Server. The connection string consists of the values which are summarized in the table 3.1

Table 3.1 Summarizes the values of the connection string

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Provider</td>
<td>Name of the OLE DB for OLAP used to connect to the OLAP engine. In the Analysis Services, this value is MSOLAP2.</td>
</tr>
<tr>
<td>Data Source</td>
<td>The location of the server or the ip address of the computer where the analysis server resides.</td>
</tr>
<tr>
<td>Initial Catalog</td>
<td>The name of OLAP database objects to be connected.</td>
</tr>
<tr>
<td>User ID</td>
<td>Username to use for connecting to the server.</td>
</tr>
<tr>
<td>Password</td>
<td>Password used for user to connect to the server.</td>
</tr>
</tbody>
</table>

About the Provider, Data Source, Initial Catalog

The provider in the connection string is the name of the OLE DB for OLAP provider which is used to connect to the Analysis Services and the value for that is MSOLAP.2. To connect to the local or a remote OLAP server database from SQL Server, we have to install the MSOLAP.2 OLE DB provider on the computer that is running SQL Server. The MSOLAP.2 OLE DB provider is installed when you install the OLAP client components from the SQL Server 8.0. The data source is the ip address or the host name of the server. The initial catalog is the database object in the specified server we want to connect to. After setting up the connection string we need to get the results which is done using the method Connect_Click () in the project. This method uses a Multi Dimensional Expressions (MDX) query to get the result into the pivot table component. After retrieving the cube data into the pivot table component we can add some more fields using the field list option on the pivot table component and we can also drill up and drill
Multidimensional Expressions (MDX)

The Multi Dimensional Expressions (MDX) are used to query the multi dimensional objects in the analysis Server. Here in this work the query returns the totals of all the members. This is done using a function called AddCalculatedMembers [25].

The syntax of MDX query used is

Select AddCalculatedMembers (Set_Expression) on columns from <cube name>

3.4 Installations Required and Solving Runtime Errors

While doing this we have I have faced some runtime errors which can be solved by installing some components. The errors which I got are as follows and their solution with which it can be resolved are also specified

a. Error1: "Unable to connect to the registry on the server (Server Name), or you are not a member of the OLAP Administrators group on this server."

Solution: This error is solved by installing Service Pack 3(a) on the server.

b. Error2: "Cannot connect to the Analysis server on computer '<sqlserver name>' The connection with the server is lost".

Solution: This error occurred after installing Service Pack 3(a), and this error is corrected first time in Service Pack 4, so installing SP4 on the server is the solution for this error.

c. Error3: “Cannot connect to the repository. Analysis server: SQLSERVER Error: Could not use "; file already in use”.
Solution: This error occurred after installing SP4 on the server, the solution for this error is that we need to set the authentication as windows for the project and disable anonymous access and we need to have OLAP Administrative privileges and System Administrative privileges and add the OLAP Administrators group in the security tab of C:/Program files/ Analysis Services/bin folder and give them Full control on that folder.

3.5 Conclusions

This OLAP Reporting Application using OWC provides a solution for creating cube browser. This work hides the underlying complexities of browsing the cube data through analysis services.
CHAPTER IV

RESULTS

The OLAP Reporting application using OWC in this thesis is tested with many cubes in the analysis services. The purpose of testing is to understand the implementation of browsing the cube data.

4.1 Software Requirements

The OLAP Reporting Application using OWC is implemented in the Microsoft SQL server 2000 environment using Visual Basic.Net (VB.Net) and some part of it is done suing VBScript and Java Script. An ASP.Net web-based application implemented is used to browse the OLAP cube data for the end-users.

4.2 Implementation of the OLAP Reporting Application using OWC

The ASP.NET web-based application, done using VB.net, VBScript and Java Script, is used to browse the cube data. This application’s user interface developed in this work, has a login page called login.aspx as shown in Figure 4.1, login failed page called loginfoailed.aspx as shown in Figure 4.2, help page called help.aspx as shown in Figure 4.3 and the main page into which the data is retrieved, called rkowc.aspx as shown in Figure 4.4.
Figure 4.1 Screenshot of the login.aspx page

Figure 4.2 Screenshot of the loginfailed.aspx page
Figure 4.3 Screenshot of Help.aspx page

Figure 4.4 Screenshot of rkowc.aspx page
In the rkowc.aspx page we need to give the server name and select the database name from the first drop down box and cube name to which we want to connect from the second drop down box and then click the Connect to cube button. Here we tested this using the Food Mart 2000 database and Sales cube. When we select database name and cube name click connect to cube button then a message will be displayed as shown in Figure 4.5

Figure 4.5 Snapshot of the message before connecting to the cube.

Click ‘OK’ in the dialogue box above then the cube data will be displayed in the pivot table component and the related chart will be displayed in the pivot chart component as shown in Figure 4.4.

4.3 Drill-down and Drill-up Functions

After displaying the cube data in the pivot table component, using the field list of the pivot table component we can add fields to the rows or columns of the pivot table. And then we can drill up and drill down accordingly using the ‘+’ and ‘-‘ symbols on the pivot table component. Drill up is viewing data in less detail and drill down is viewing data at a greater level of detail. The field list is shown in Figure 4.6 and the result of the
drill down operation is shown in Figure 4.7.

Figure 4.6 Snapshot of Field List

Figure 4.7 Results of the pivot table component and chart component with the Food Mart 2000 cube data and Drill down operation.
CHAPTER V

FUTURE WORK

This chapter summarizes the main contributions and future work of this thesis regarding the OLAP Reporting Application using OWC. This chapter also explains about some future works based on the current work.

5.1 Contributions

The main purpose of this thesis is to develop the OLAP Reporting Application Using OWC for application programmers to build user-friendly interface applications for OLAP solutions. This work is also done to hide the complexities in using the analysis services in the process of browsing the OLAP cubes.

The contributions are summarized as follows:

- Development of the OLAP Reporting Application using Office Web Components.
- Detailed analysis of the functionalities of Analysis Manager in the process of viewing the OLAP cubes using OWC.
- Development of the web-based interface application to browse the cubes data, which hides the complexity of browsing the cube data using Analysis Services.
• Development of the web-based interface application to browse the cube data which satisfies Multi-Dimensional Conceptual View rule through which data can be presented to the user in a multi-dimensional pattern.

• Development of the application which satisfies Transparency rule through which users should not need to know that they are using an OLAP database, they can just select the database names and cube names from the drop down lists.

• Development of the application which satisfies Accessibility rule through which the application chooses the best source of the data to support the MDX query.

• Development of the application which satisfies Consistent Reporting Performance rule through which the performance will be the same regardless of the number of dimensions in use.

• Development of the application which satisfies Client-Server Architecture rule.

• Development of the application which satisfies Generic Dimensionality rule which tells users that dimensions are all equal; there should be no bias towards any one dimension.

• Development of the application which satisfies Dynamic Sparse Matrix Handling rule through which null values are handled in an efficient way using sort ascending and sort descending methods.
• Development of the application which satisfies Multi-User Support rule through which the application support more than one user.

• Development of the application which satisfies Unrestricted Cross-Dimensional Operations rule which allows data roll ups through levels of a hierarchy.

• Development of the application which satisfies Intuitive Data Manipulation rule through which the user views of data contain everything required without resorting to using menus or multiple trips across the user interface.

• Development of the application which satisfies Flexible Reporting rule through which users can present data in any way they like.

• Development of the application which satisfies Unlimited Dimensions and Aggregation Levels rule which allows any number of dimensions and levels.

• Applying the case study of many cubes in the OLAP Reporting Application.

• Building an Interactive web page with Microsoft word using Object Embedding Technique.

5.2 Future Works

This research developed the OLAP Reporting Application using OWC for browsing the OLAP cubes. It also demonstrated their functionalities with many cubes. However, the application for viewing the data of mining model is not developed and implemented. The development of this application will be the future work. Here in this work we used Decision Support Objects (DSO), it is said that this feature will be removed in the next version of
Microsoft SQL Server and in future we can use Analysis Management Objects (AMO) instead of DSO. Decision Support Objects are included in SQL Server 2005 to support the unmanaged applications. AMO are included in SQL Server 2005. Using Decision Support Objects provides access to only the member of the OLAP Administrators group. And this is one of the limitations for this project, in future work we need do the same using some other technology which can allow the members of the non administrative group to access the data.
BIBLIOGRAPHY


APPENDICES
APPENDIX A

APPLICATION INTERFACE OF OLAP REPORTING APPLICATION USING OWC

Figure A.1 Screenshot of the OLAP Reporting Application interface for the end-users
## APPENDIX B

APPLICATION INTERFACE OF CUBE BROWSER OF ANALYSIS SERVICES

![Screenshot of the “Cube Browser” in the Analysis Services](image)

**Figure B.1** Screenshot of the “Cube Browser” in the Analysis Services
APPENDIX C

EXAMPLE OF STAR SCHEMA

Figure C.1 Example of Star Schema in Analysis Manager
APPENDIX D

EXAMPLE OF SNOWFLAKE SCHEMA

Figure D.1 Example of Snowflake Schema in Analysis Manager