This is the Pre-Published Version.

Application of Data Warehouse and Decision Support System in Construction Management

K.W. Chau[1], Ying Cao[2], M. Anson[1] and Jianping Zhang[2]

[1]Dept. of Civil & Structural Engineering, Hong Kong Polytechnic University, Hunghom, Kowloon, Hong Kong

[2]Dept. of Civil Engineering, Tsinghua University, Beijing 100084, China

Abstract

How to provide construction managers with information about and insight into the existing data so as to make decision more efficiently without interrupting the daily work of an On-Line Transaction Processing (OLTP) system is a problem during the construction management process. To solve this problem, the integration of a Data Warehouse and a Decision Support System (DSS) seems to be efficient. 'Data warehouse' technology is a new database branch, which has not been applied to construction management yet. Hence, it is worthwhile to experiment in this particular field, in order to gauge the full scope of its capability. First reviewed in this paper, are the concepts of the data warehouse, On-Line Analysis Processing (OLAP) and DSS. The method of creating a data warehouse is then shown, changing the data in the data warehouse into a multidimensional data cube and integrating the data warehouse with a DSS. Finally, an application example is given to illustrate the use of the Construction Management Decision Support System (CMDSS) developed in this study. Integration of a data warehouse and a DSS can enable the right data to be tracked down and provides the required information in a direct, rapid and meaningful way. Construction managers can view data from various perspectives with significantly reduced query time, thus making decisions faster and more comprehensive. The applications of a data warehousing integrated with a DSS in construction management practice are seen to have considerable potential.

Keywords: Decision-Support System, Project Management, Construction, Data Warehouse, On-Line Analysis Processing

1. Introduction

1) Using Decision Support System (DSS) in construction Management

At present, some transaction processing systems, which are updated continually throughout the day, are often employed to run the day-to-day business of a construction company (Sarda, 1999). For instance, if some materials are delivered into the warehouse, the OLTP will consistently make additions to the inventory. However, it is usually found in such systems that the construction process is a "temporary" and "specific" activity, which means the data of one project can seldom be used for another project. Is that the true situation? Probably not, because although construction products are 'unique', some similarities still exist between them and construction processes and management skills are typically common to all projects (Yang and Yau, 1996). How to analyze the successes and failures of finished projects and how to use the existing data to analyze patterns and trends for new projects are the problems we have to face.

During the project control phase, in order to take rectifying actions for any deviations in

the performance, project managers often need timely analysis reports to measure and monitor construction performance. They also need timely analysis reports to assist in making long-term decisions (Vanegas and Chinowsky, 1996). It is found that most of the reporting and analysis time was spent on collecting data from the various systems before the analysis can be made. Managers want and need more information, but analysts can provide only minimal information at a high cost within the desired time frames (Dyche, 2000). In order to provide information for predicting patterns and trends more convincingly and for analyzing a problem or situation more efficiently, an integrated decision support system designed for this particular purpose is needed.

An important role of a decision support system is to provide information for users to analyze situations and make decisions. Put in another way, a decision support system provides information for employees to make decisions and do their jobs more effectively (Poe et al., 1998). This decision-making can be of a long-term strategic nature, such as analyzing event patterns over several years to prevent or reduce the rate of occurrence of a particular event. Decision-making can also be short-term and tactical in nature, such as reviewing and changing the time schedule for a particular part of a project. Good systems provide the information needed so that employees are better equipped to make more informed decisions. Described in this paper is the development of a prototype decision support system, employing the new 'data warehouse' technology incorporating large quantity of analysis information needed for both long-term and short-term construction management decision-making.

2) **Using a data warehouse to support a DSS**

Being a new branch of the database community developed in recent years (Samtani et al., 1998), the 'data warehouse' is a read-only analytical database that is used as the foundation of a decision support system. The purpose of a data warehouse is to ensure the appropriate data is available to the appropriate end user at the appropriate time.

A data warehouse is a global repository that stores pre-processed queries on data, which reside in multiple, possibly heterogeneous, operational query base for making effective decisions (Poe et al., 1998). The contents of a data warehouse may be a replica of part of some source data or they may be the results of preprocessed queries or both. This method of data storage provides a powerful tool helping project organizations in making decisions. The architecture of a data warehousing system allows a number of alternative ways to integrate and query (such as previous or projected) information stored in it. Thus, a data warehouse coupled with OLAP enables project managers to creatively approach, analyze and understand project problems. The data warehouse system is used to provide solutions for construction problems since it transforms operational data into strategic decision-making information. The data warehouse stores summarized information instead of operational data. This summarized information is time-variant and provides effective answers to queries, such as "What are the supply patterns and trends of various construction materials?", "How is the material consumption this year different from its counterpart last year?", "How many accidents happened in the last ten years and how much did they cost?", "What is the percentage increase in the cost of human resources during the last five years?", "Did machine repair have any influence on construction progress? If so, what was the influence coefficient?" and so on. To extract this information from a distributed relational model, we would need to query multiple data sources and integrate the information at a particular point before presenting the answers to the user. In a data warehouse, such queries find their answers in a central place, thus reducing

the processing and management costs.

3) What's new in our system

As a matter of fact, decision support systems have been applied in construction management for several years. The early systems such as management information systems, report-oriented systems and so on are often born with flaws (Shumate, 2000). Firstly, they are not separated from transaction systems completely and the sharing of a database or data file slows down either transaction or analysis process. Secondly, because of the limitations of a relational database, users can only observe their data from flat views. Thirdly, these applications are all developed by computer specialists in information centers after lengthy data analysis, but sometimes not all the requirements of construction managers are embodied sufficiently. These problems could be solved in the Construction Management Decision Support System (CMDSS), developed in this study. The main characteristic of CMDSS is the separation of the analysis database from the operational database, which renders the decision support process much faster. Another advance is the use of OLAP, which changes the data in a relational database into multidimensional cubes that could be observed from all perspectives. In addition, visualization (use of graphic and presentation techniques) presents data from several kinds of views. In CMDSS, moreover, users could do a lot more on their own without computer experts preprogramming everything for them. Because of these advances, construction managers can make decisions more efficiently, which is the key objective of our system.

2. System design

1) Design data warehouse for decision support system

Existing data models used to design traditional OLTP systems may not be appropriate for modeling complex queries under a data warehouse environment. The transactions in OLTP systems are made up of simple, pre-defined queries. For example, if we want to know the latest arrival date of all materials in all warehouses, we can use a Structural Query Language (SQL) query such as "*SELECT flngMaterialID, MAX(fdtmArriveDate) AS Date, flngMaterialType, flngDepotID FROM tb_Stock GROUP BY flngMaterialID, flngMaterialType, flngDepotID HAVING (flngMaterialType = 2)*". In the data warehouse environments, queries tend to use connections between tables and have a longer computation time, such as "*Select Material.MaterialName, Material.MaterialKind, Inventory.Quantity from Material, Inventory, Material INNER JOIN Inventory ON Material.MaterialID = Inventory.MaterialID*". The above query provides the name and type of materials as well as the quantity in the inventory. Since the information on materials and inventory are put in different tables, a connection between table 'Material' and table 'Inventory' is required. This kind of processing environment warrants a multidimensional data model, a new perspective on data modeling.

The conceptual multidimensional data model can be physically realized in two ways, (1) by using a trusted relational database approach (star schema/ snowflake schema) or (2) by making use of a specialized multidimensional database. The 'star' schema is adopted here mainly because of its clarity, convenience and rapid indexing ability (Corey and Abbey, 1999). The other methods are not so suitable here since they involve more or less a much more complicated transformation, which does not appear to be justified in our situation.

In concise term, a star schema can be defined as a specific type of database design used to support analytical processing, which includes a specific set of denormalized tables. A star

schema contains two types of tables: fact tables and dimension tables. Fact tables contain the quantitative or factual data about a construction management entity. Dimension tables are smaller and hold descriptive data that reflect the dimensions of an entity. SQL queries then use predefined and user defined links between the fact and dimension tables within the star schema, with constraints on the data to return required information.

A typical material inventory model with sample dimensions and properties is shown in Table 1 for CMDSS developed in this project.

The core part of any star schema is the fact table, which is shown as Table 2.

Now the whole star schema model could be created. There is one material inventory fact table and five dimension tables shown in Figure 1. These dimension tables are connected with the fact table by foreign keys (FK), which can keep all the views coherent.

Besides the inventory star schema, the example given above, several other star schemas are designed in our system, including material issuing, material balance, material use, machine cost, machine use, machine repair, human resource use, salary, progress, non-compliance, event, etc. Each star schema includes a fact table and several dimension tables, just like the Inventory star schema.

In addition to the data model design, several other steps should be taken before the data warehouse can be completed. These steps are as follows:

- Data is extracted from the source systems, databases and files
- Data from the source systems is integrated
- Data is loaded into the data warehouse
- Data is transformed into the format that can be used by the front-end tool

The process of a data warehouse design is shown in Figure 2.

In CMDSS, the "Import and Export Data" tool is used to integrate data from distributed OLTP databases, files, etc.

With a view to transforming the fact table and dimension tables in the star schema designed above into a multidimensional cube that can be further explored by the front-end tools, such as Visual Basic, MS Access, MS Excel, the OLAP tool is applied here. Microsoft OLAP Services is based on and tightly linked to relational databases. However, it is a real multidimensional information system, where all information is modeled in terms of OLAP structures, not relational structures (Shumate, 2000). The OLAP structures are a valuable feature because many important analyses are difficult or impossible to phrase in SQL using tabular structures. For example, one characteristic of most OLAP applications is the need to provide fast access to aggregated source data. Precalculating all possible aggregations can lead to a tremendous increase in the storage requirements for the database, while calculating all aggregations on each occasion makes for a slow query response time. The approach taken by OLAP Services is to precalculate some of the possible aggregate data values, and to leave any remaining aggregation and all other calculations to be completed at query time. Microsoft OLAP Services provides a relatively well-optimized solution (Peterson and Pinkelman, 1999).

In this case, cubes, dimensions, measures, hierarchies, levels, and cells constitute the basic OLAP structures. These, taken together, define the logical structure of an OLAP database. A data cube is a structure for housing multidimensional data. Everyone using a data warehouse will use cubes when analyzing the data. Measures are the data that we wish to analyze, while dimensions define the organization of these measures (Thomsen et al., 1999). Our data warehouse may contain an inventory table that has fields for location, time, material, supplier,

storekeeper, price, quantity and total amount. If so, we will generally analyze price, quantity and total amount by warehouse, time, material, supplier and storekeeper. In this case, price, quantity and total amount will be our measures, and warehouse, time, material, supplier and storekeeper will each be a dimension. The elements of a dimension are called members. The path to organize members in a cube is called hierarchy. For instance, the time dimension may be organized in two hierarchies: natural time hierarchy and fiscal time hierarchy. In the former one, time may be organized in year, quarter, month and date. While in the latter one, time may be organized in fiscal month, fiscal week. A level refers to a group of related members which share a common meaning. For example, a level construct named 'month' may contain all of the month-level members in a time dimension. Each unique intersection composed of one member from every dimension in the cube is called a cell (Thomsen et al., 1999). For instance, the intersection of July (member of Time) and Hong Kong (member of Geography) constructs one cell, and the value of the cell could be either measure, such as price, quantity or total amount.

Microsoft OLAP Services is strongly based on relational databases. All dimension levels and cube measures need to correspond to columns of tables, views, or queries. They can be in many different tables or all in one table, so long as dimension tables and fact tables can be joined in a single query. OLAP Services uses a highly declarative linkage among dimension, cube structures and Relational Database Management System (RDBMS) tables. Once the links are created, OLAP Services will form all queries on the linked tables and manipulate all query results.

Many cubes about material, machine, human source, progress, quality and event are created in our system on the basis of the star schemas designed before. New multidimensional cubes can be added at any moment by the users as the need arises. An 'Inventory" cube created in our system is shown in Figure 3.

The major operations that could be done on OLAP cubes are Selection, Roll-up, Drill-Down and Slice, through which we can view data from all perspectives and all levels (Vassiliadis and Skiadopoulos, 2000).

2) **Design of DSS for construction management**

The following fundamental questions should be first understood prior to the design of a DSS. How are reports and analyses shared between users? How are structured navigation paths or command buttons created? Is there a query controller to limit the allowable elapsed time run for a query or to limit the total number of rows that can be returned? Is there an ability to run a query during off-peak hours to save costs? … These questions are quite important for deciding the aim and the direction of a DSS, which also means the success of it.

The purpose of the DSS is to enable analysts to extract information quickly and easily. The data being analyzed are often historical in nature: daily, weekly, and yearly results. For this reason, the System Development Life Cycle (SDLC) for DSS is quite different from an On-Line Transaction Processing (OLTP) system. The SDLC of a common OLTP system will usually go through several phases: planning, analysis, design, development, testing, and implementation. The focus of a DSS is data, not construction processing and their associated functionality. This lack of domain functionality implies a much faster development life cycle.

Under the DSS application, a front-end tool is employed to create predefined reports to accommodate the need for different levels of users to have pre-built reports to begin their analysis (Poe et al., 1998). Hence, the general data access processes are visualization of the data warehouse, formulation of the request, processing the request, and presentation of the

results. After the data warehouse design and OLAP transformation, two steps are left to create a DSS application. The first is to design the front-end interface. The second is to generate codes to access and navigate metadata to obtain information on the data in the warehouse, and link it together with the front-end interface.

A major consideration in designing the DSS interface is the level of users. Different type of DSS interface should be designed for different users. Generally, we may face two distinctly different levels of users:

- The experienced users who develop ad hoc queries using parameters. These users were also trained on the data model.
- The novice or casual users, who are most comfortable in a point and click environment, where icons are employed instead of queries.

For the above reason, two different interfaces were designed for our CMDSS. The first is designed for the experienced users. They can select parameters in the parameter dialog box and view the data more flexibly. A typical parameter dialog box is shown in Figure 4.

The second interface is designed for the novice who can just click and view the predefined information directly. The selection interface is shown in Figure 5.

Moreover, data returned from a request in this DSS can be displayed in a wide variety of ways. Here data reports and some graphical views are explored to view the data and assist users in gathering insights from the data. Once a data report is presented, a wide range of capabilities may allow modification of the display, including: changing the axis of the report (Rows and columns), changing the sort order of the results, adding subtotals and grand totals at appropriate breaks in the reports, formatting of fonts, styles, sizes, and colors. Graphical display of information allows easy detection of trends and anomalies. A wide variety of graph types are available here, including line, pie, bar, area and so on. With a graphical display, users may be able to change graph type, axis labels, colors and titles.

The next significant step following the design of the interface was to generate codes integrating the data warehouse and the interface. In the novice interface, once a user presses a button, a Precoded SQL statement will be passed directly to the Multidimensional Database Management System (MDBMS) and the appropriate answer set will be displayed on the interface by the prediction function. But in the interface for experienced users, once a user has formulated a request, our system can translate the end user's request to generate the appropriate SQL statements, get the result set from the MDBMS and display it in a variety of views. Because of its good connections with databases, Visual Basic 6.0 was adopted here to develop our CMDSS, using also ActiveX Data Objectives Multidimensional (ADOMD) as a vehicle for communication. The following code is a sample used to display the value of a cell in a grid.

```
For intColum = 0 To Cellset.Axes(0).Positions.Count –1
    For intRow = 0 To Cellset.Axes(1).Positions.Count –1
        Grid.Column = intColumn + intFixedColumns
        Grid.Row = intRow + intFixedRows
        Grid.Value = Cellset (intColumn, intRow).FormattedValue
    Next intRow
Next intColumn
```

Other advanced capabilities that many construction organizations require to support the full decision-making process may also include: exception reporting (alerts are messages that appear when user defined conditions are not met or when there is a problem), drill-down (users

have the freedom to "drive off" the existing report and retrieve information that may lie along, above, or below the current level of detail. Drill-down can be done from a report or a graph), data surfacing (we can keep the report layout constant but change the constraints. For example, a material-consumption trend report for concrete in Hong Kong, changed to a material-consumption trend for steel in Beijing), ranking (review information that is ranked on one or more columns), and automation (mechanisms are in place to schedule recurring analysis at a specified time).

In brief, there are a lot of differences between DSS development and common OLTP system development, which are embodied on the objectives, databases, SDLC, capabilities and development tools.

3. **Application**

CMDSS was used in the Hong Kong Polytechnic University Student Dormitory construction project. The results enable the prototype properties to be validated and demonstrate the capabilities of the system in real application case. As an illustrated example, the procedure in using CMDSS to facilitate the inventory decision is shown in the following paragraphs in a step by step manner. The procedure for other decision-making is similar.

- Choosing the interface: Two kinds of interface are available. One interface is for novices and the other is for experienced users. Here we choose a self-defined interface, which is designed for experienced users, with the tool bar shown in Figure 6.
- Choosing display views: There are several views available to display the query result, including line chart, pie chart, bar chart, area chart and report file. Users can choose any of them. A line chart is selected in Figure 6.
- Input parameters: Some parameters are entered before the query result is viewed. Here we choose inventory as an observing cube, and choose the Time dimension (Quarter level) as the X Axis, the Material dimension (Material Kind level) as the Y Axis and the Amount as measure. Users can also attach some conditions to the query. Conditions can be selected in the combo-box and added by 'Add' button or cleared by 'Reset' button. In this example, we want to determine the total amount of the materials stored in the warehouse whose storekeepers are female with materials from suppliers in Beijing. This condition is displayed in the condition box as "[SupplierGeography].[City].[Beijing], [Gender].[Gender].[F]". The parameter selection dialog box is shown in Figure 7.
- Result display: After entering the parameters, results will be displayed on the interface. An Inventory cube, which is observed in several views in the DSS, is shown in Figure 8. In the line chart, X-axis is Time Dimension and Y-axis is Total Cumulative Amount together with contributions from different material types. We can observe the Total Cumulative Amount of all the material types in four quarters of a year. The legend on the right of the chart displays the colors representing different material types. In the grid shown below the chart, the values of the amount of each material type at each quarter of the year are given. The same information can be observed in four different types of charts: Line chart, Pie chart, Bar chart and Area chart.

The above results illustrate that construction managers can determine the inventory trend of the material and the amount of each material type easier from the information shown by the above charts and the grids in Figures 6 to 8. It facilitates significantly the managers to formulate an appropriate inventory decision or warehouse storage strategy.

4. **Conclusions**

In this paper, the development of a prototype construction management decision support system, acronym CMDSS, employing the integration of the 'data warehouse' technology with an OLAP is delineated. It has been illustrated, through a real case application in Hong Kong, that CMDSS is advanced at least in the following aspects: CMDSS enables insight to be gained into the factors having impacts on construction management activities that will help managers in making decisions to improve management performance. CMDSS is interactive. Users can interact with the computer so that the users can constantly refine the view of data to pursue various ways of thought. CMDSS provides extremely fast response to queries. CMDSS is multidimensional. Users can view figures from multiple perspectives and can also choose different view angles. In Short, CMDSS is able to assist project managers by providing accurate and timely information for construction decision-making. The integration of a data warehouse and a DSS appears to be a promising way to solve decision-making problems during construction management process.

References:

Corey M. and Abbey M., SQL Server 7 Data Warehousing, McGraw-Hill, 1999.

Dyche J., e-Data Turning Data into Information with Data Warehousing, Addison-Wesley, 2000

Peterson T. and Pinkelman J., Microsoft OLAP Unleashed, SAMS, 1999

Poe V., Klauer P. and Brobst S., Building A Data warehouse for Decision Support, Prentice Hall, 1998

Samtani S., Mohania M., Kumar V., and Kambayashi Y. Recent Advances and Research Problems in Data Warehousing. Advances in Database Technologies-ER'98 Workshops on Data Warehousing and Data Mining, Mobile Data Access, and Collaborative Work Support and Spation-temporal Data Management Singapore, November 1998 Proceedings, 81-92.

Sarda N.L. Temporal Issues in Data Warehouse Systems. Database Applicaitions in Non-Traditional Environments'99, The Proceedings of the 1999 International Symposium on Database Application in Non-traditional Environments (DANTE'99), 27-34.

Shumate J., A Practical Guide to Microsoft OLAP Server, Addison-Wesley, 2000

Thomsen E., Spofford G. and Chase D., Microsoft OLAP Solutions, Wiley Computer Publishing, 1999

Vanegas J. and Chinowsky P., Computing in civil Engineering, American Society of Civil Engineers, 1996.

Vassiliadis P. and Skiadopoulos S. Modelling and Optimisation Issues for multidimensional Databases. Advanced Information System Engineering, 12th International Conference, CaiSE 2000, Stockholm, Sweden, June 2000 Proceedings, 482-497.

Yang Jyh-Bin and Yau Nie-Jia. Application of Case-Based Reasoning in Construction Engineering and Management. Computing in Civil Engineering, Proceedings of the Third Congress held in conjunction with A/E/C Systems'96, 663-669.

Table 1 Dimensions and Properties

Time Dimension	Material Dimension	Warehouse Dimension	Supplier Dimension	StoreKeeper Dimension
Time_key	Material_key	Warehouse_key	Name	Name
Date	Name	Name	Type	Birthday
Year	Type	Position	Street Address	Joining Date
Month	Spec	Structure Type	City	Gender
Day	Unit	Content Type	Province	
Quarter		Management Fee	Country	
Day of week			Postal Code	
			Telephone	
			Fax	
			Email	

Table 2 Fact Table

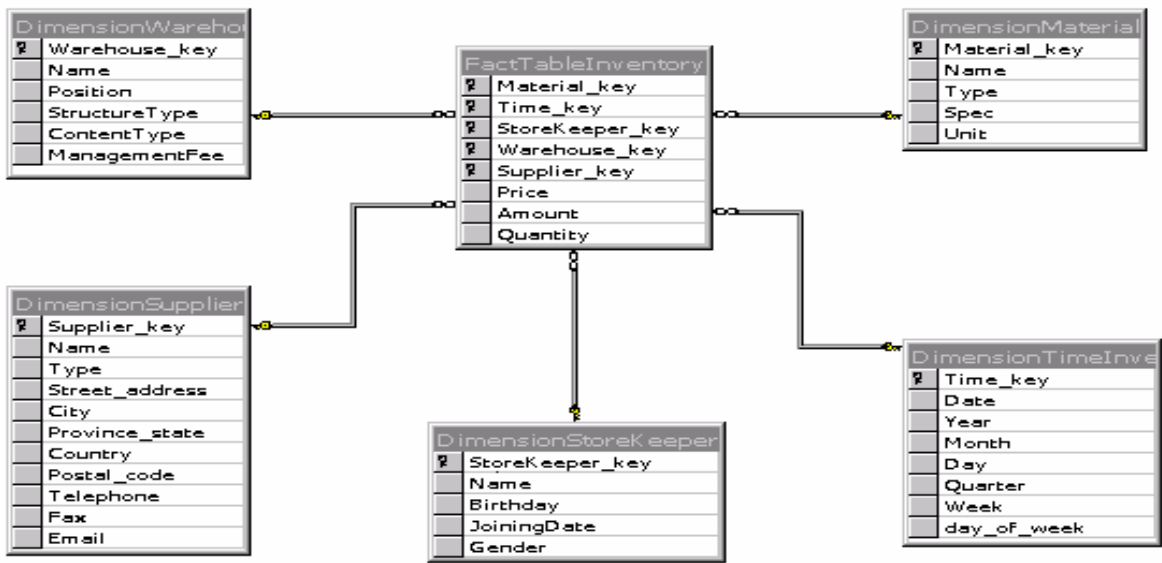| Foreign Key	Measures
Material_key	Price
Time_key	Quantity
StoreKeeper_Key	Total Amount
Warehouse_key	
Supplier_key	

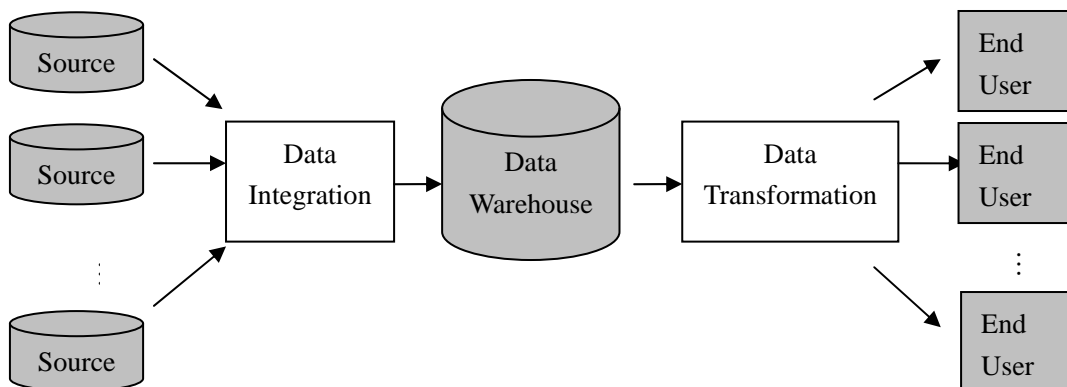Figure 1: Material Inventory Star Schema

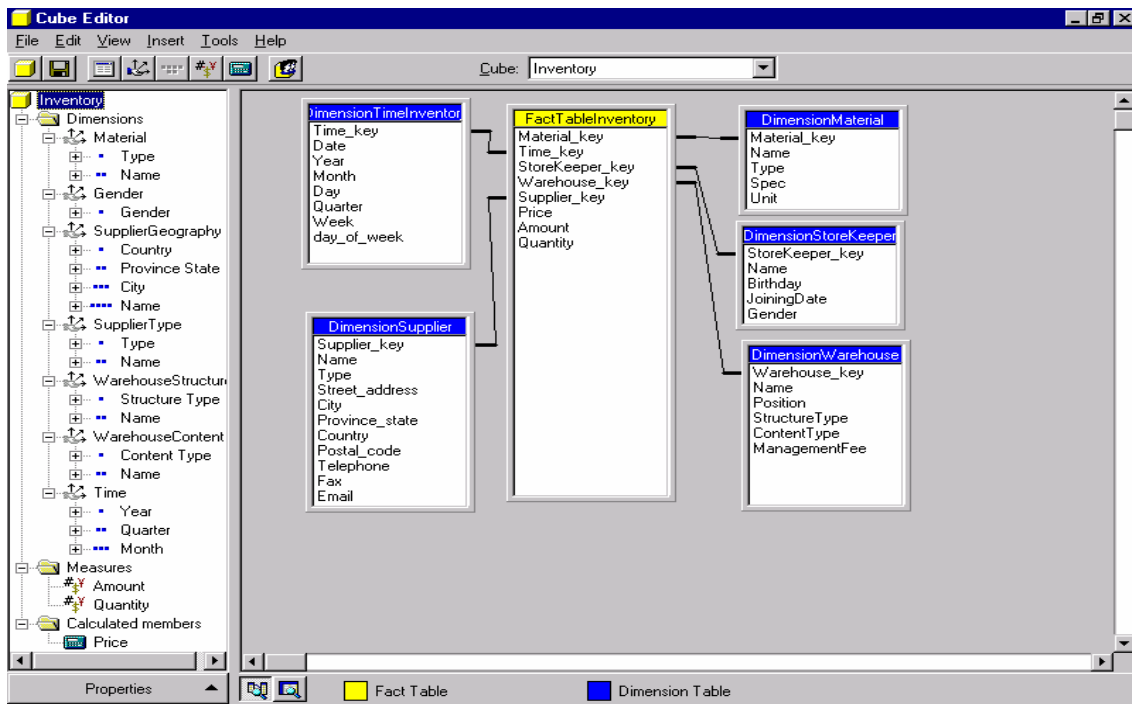Figure 2: Process of Data Warehouse Design

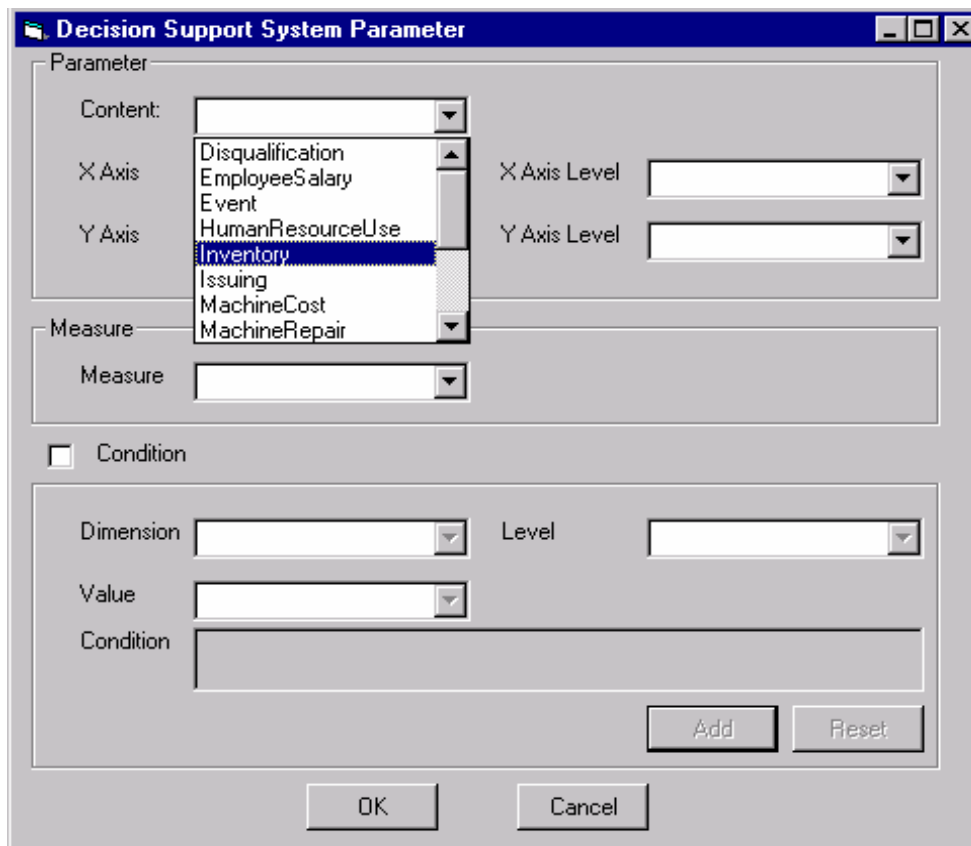Figure 3: Inventory Cube In Microsoft OLAP

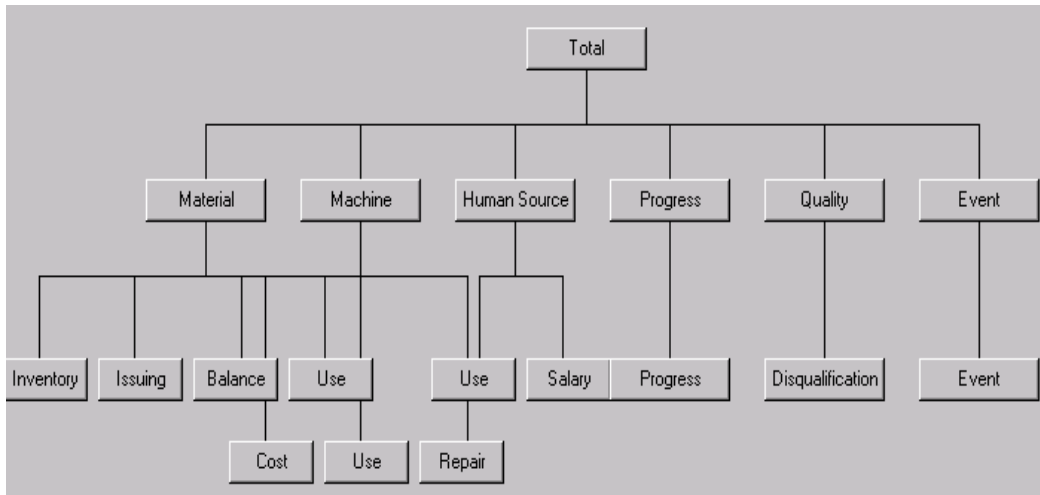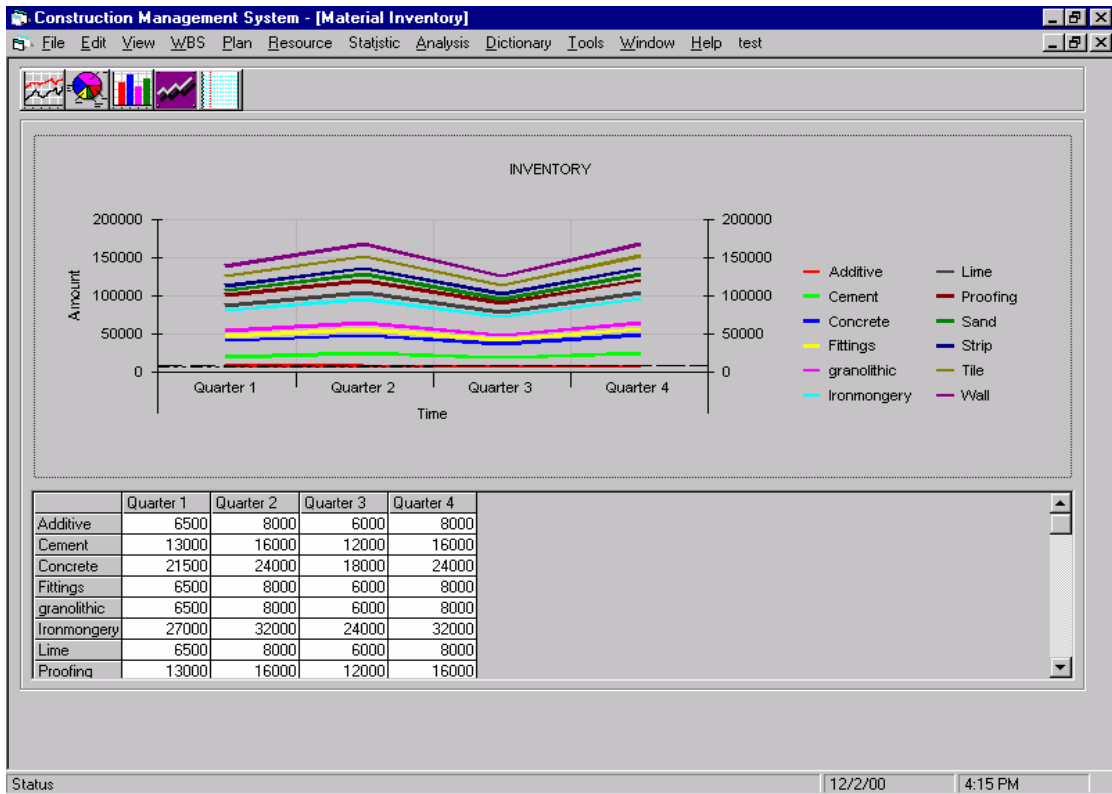Figure 4: Parameter Selection Dialog Box for DSS
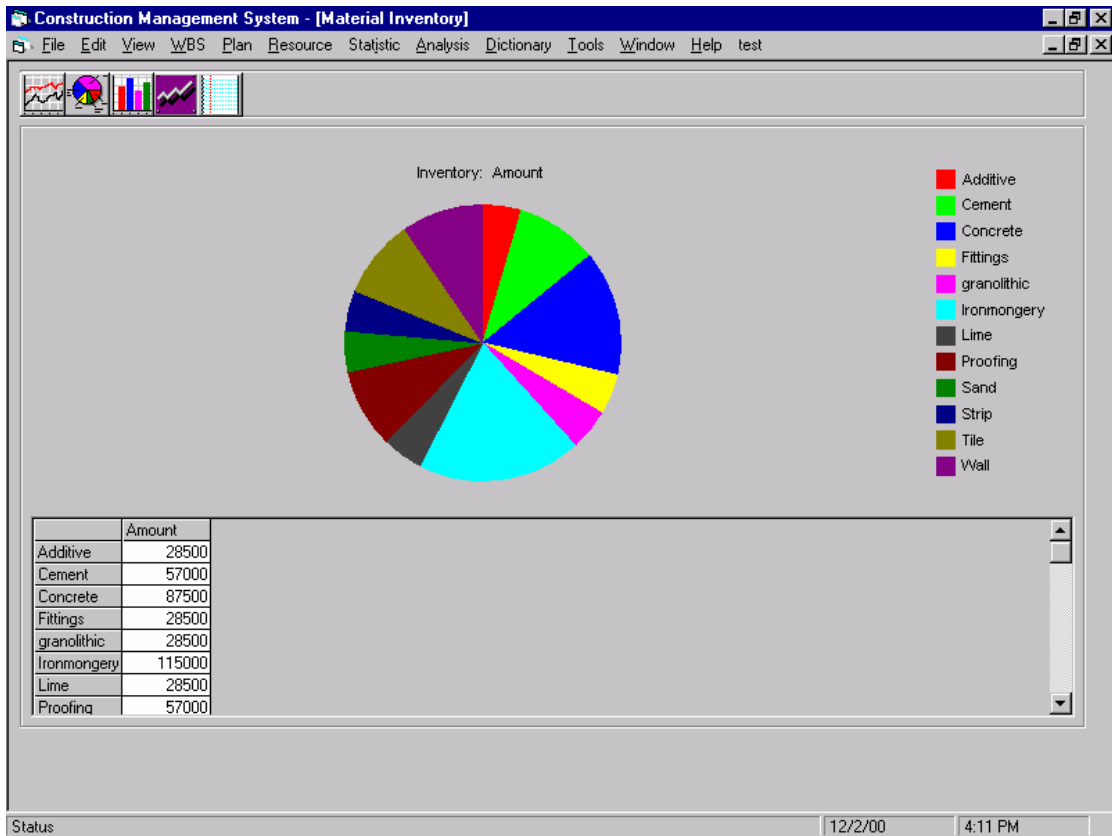
Figure 5: Predefined Information Selection Box

Figure 6: DSS interface selection and display views selection

Figure 7: Parameter Selection

(a)
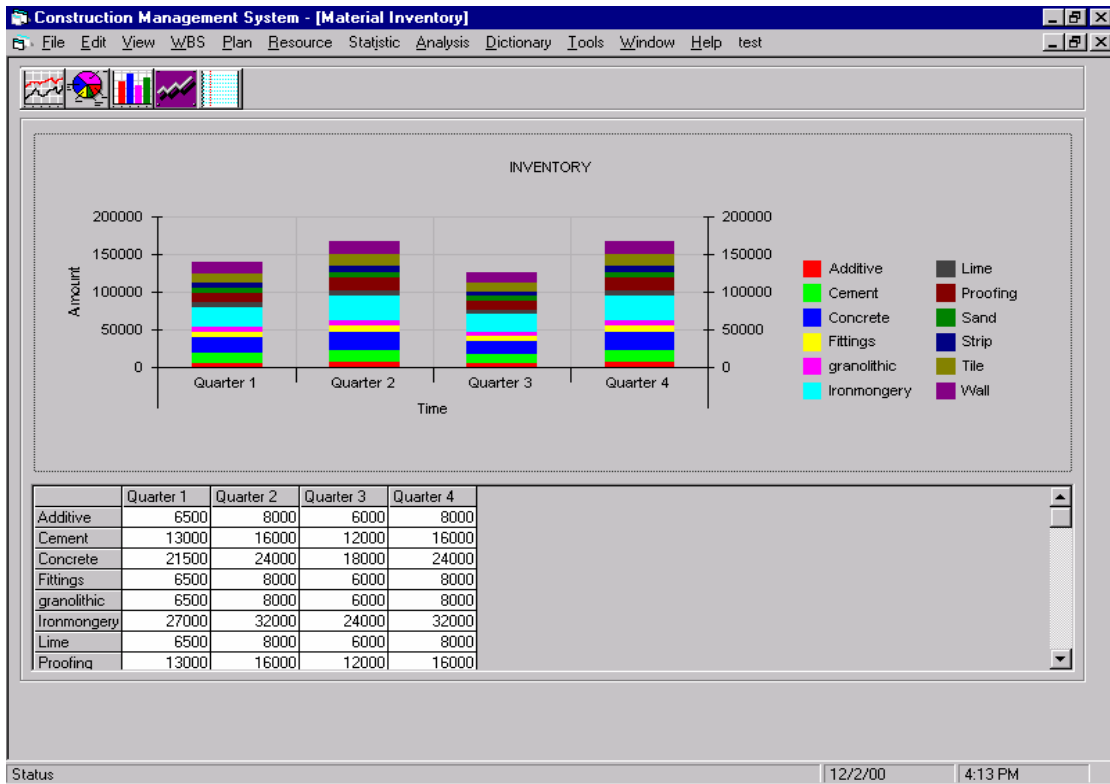

(b)

(c)

File   Edit   View   WBS   Plan   Resource   Statistic   Analysis   Dictionary   Tools   Window   Help   test

INVENTORY

	Quarter 1	Quarter 2	Quarter 3	Quarter 4
Additive	6500	8000	6000	8000
Cement	13000	16000	12000	16000
Concrete	21500	24000	18000	24000
Fittings	6500	8000	6000	8000
granolithic	6500	8000	6000	8000
Ironmongery	27000	32000	24000	32000
Lime	6500	8000	6000	8000
Proofing	13000	16000	12000	16000

Status   12/2/00   4:13 PM

(d)

Construction Management System - [Material Inventory]

File   Edit   View   WBS   Plan   Resource   Statistic   Analysis   Dictionary   Tools   Window   Help   test

INVENTORY

	Quarter 1	Quarter 2	Quarter 3	Quarter 4
Additive	6500	8000	6000	8000
Cement	13000	16000	12000	16000
Concrete	21500	24000	18000	24000
Fittings	6500	8000	6000	8000
granolithic	6500	8000	6000	8000
Ironmongery	27000	32000	24000	32000
Lime	6500	8000	6000	8000
Proofing	13000	16000	12000	16000

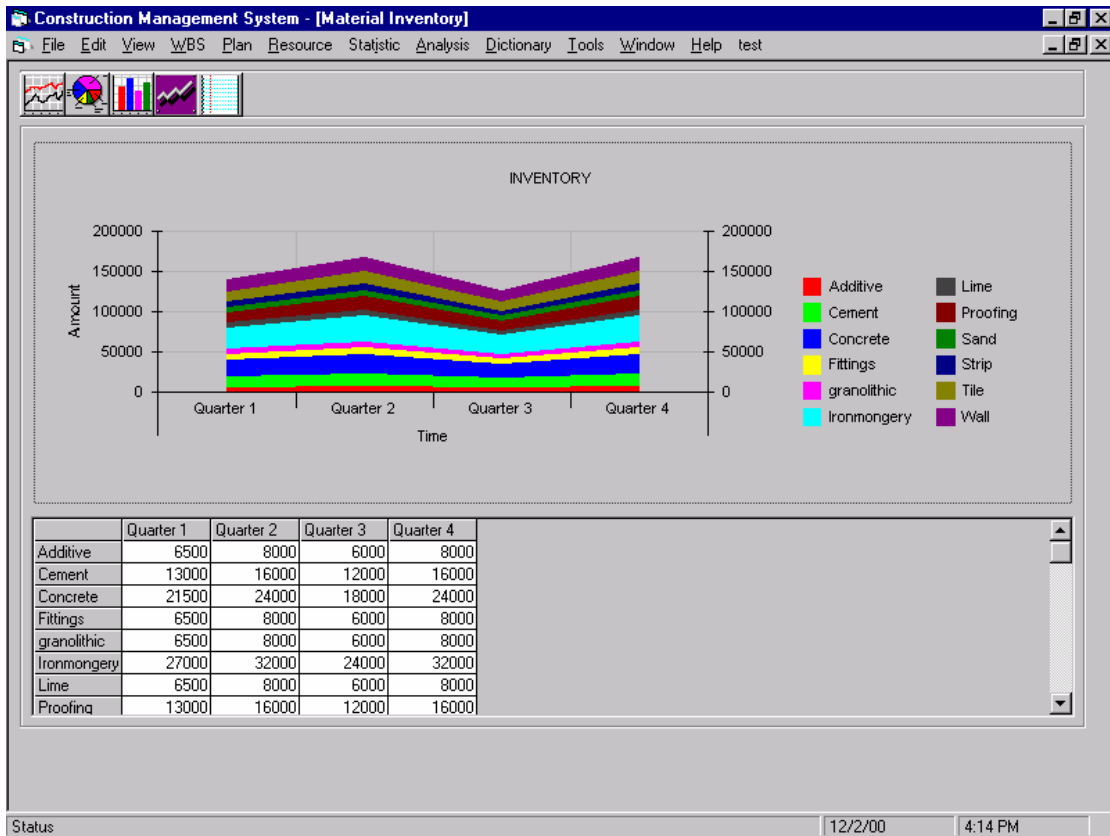Status   12/2/00   4:14 PM

Figure 8(a) to (d): Material Inventory observed from Various Views