Data Warehousing and OLAP Technology for Primary Industry

Taehan Kim[1)], Sang Chan Park[2)]

[1)]Department of Industrial Engineering, KAIST (taehan@kaist.ac.kr)
[2)]Department of Industrial Engineering, KAIST (sangpark@kaist.ac.kr)

Abstract

The concept of DW(data warehousing) and OLAP(on-line analytical processing) has been appealing to many practitioners and researchers, and the technology has been applied to many industries for several years. But there has not been any research about appropriate models of DW and OLAP methodology for primary industry, where many data sources exist and tremendous data of heterogeneous form occur almost everyday. In this paper, we propose a methodology of DW and OLAP for primary industry. Many data sources in primary industry will be introduced and the methodology that is composed of four stages will be explained in detail. The four stages are preprocess stage, data warehouse stage, OLAP stage, and documentation stage. Through these stages, the heterogeneous data originated from various data sources are integrated and summarized, and customized and distributed for each user finally. In the first stage, preprocess stage, our system contacts to various data sources and brings heterogeneous data from them. The data are stored, integrated, and summarized in the second stage, DW stage. In the third stage, OLAP stage, the summarized data are analyzed and the system generates many kinds of report. In final stage, documentation stage, the reports are converted to user-friendly form and distributed for each user. This paper also shows the application of our methodology to primary industry. Specifically, we applied our methodology and make a system for many users concerned with apple pomiculture. Quantitative and qualitative information about apple culture and fresh apple market is provided for the users concerned. This paper also describes the procedure for designing and developing the system. To make the system serve suitable information for many users, many data sources must be analyzed firstly when we design the system. We also have to recognize many requests from the future users, and define each screen and the data and information which will be shown in the screen that must be customized for each user. After determining each screen, we receive feedback from each user and refine the screen. This paper could be the first attempt to apply DW and OLAP technology for primary industry, and will provide useful knowledge for developing DW and OLAP system based on heterogeneous data sources and tremendous data.

1. Introduction and Research Motivations

Accurate and timely decision making in business management becomes more important than before, and so does the decision making about purchase on customer side. As information and communication technique advances rapidly these days, the amount of information that decision makers have to consider has been increased. Decision support system came into existence for these reasons, and it provides appropriate data and information for decision makers. But traditional decision support system is based on traditional database system and OLTP(on-line transaction processing), in which decision makers cannot look the data at various standpoint. Moreover, for the accurate decision making, users must aggregate much numeric data in multidimensional manner but OLTP can't supply those functions. OLAP appeared

for solving the problem and it allowed many users to view and aggregate quantitative data on the time that the users want. Until recently, many researches about DW and OLAP has been focused to processing and storing data, and it didn't cover how to summarize and analyze heterogeneous data, which are different to one another in data format, generation cycle, and so on. In this paper, we propose a methodology for integrating those data using DW and OLAP technology. We describe four stages, which are preprocess stage, DW stage, OLAP stage, and documentation stage. Under the predefined business rule and meta data, the heterogeneous data originated from different data sources will be processed, stored, summarized, analyzed, and distributed for users through the four stages. Detail explanation will be presented in Chapter 3. We also show how the proposed methodology can be applied to real business world in Chapter 4. Many heterogeneous data sources about apple cultivation and market will be presented and the data will be stored in our DW. After that, the data will be analyzed in OLAP system and will be distributed for user's purpose later. In Chapter 5, we conclude our paper and append future research direction about this paper.

2. DW and OLAP Overview

The concept of OLAP(On-line analytical processing) was introduced by Codd, Codd and Salley in their technical report in 1993 [1]. Until that time, OLTP had been widely used for business data processing and decision making, but it couldn't provide timely and accurate information for the users as described in previous chapter. OLAP can be defined as analysis-based decision-oriented information processing [2], and it draws necessary data from DW, which is a special database designed for analysis and decision making, and has aggregated data rather than daily transaction data. The comparison of OLTP with OLAP is presented in Table 1.

Table 1 OLTP vs. OLAP

	OLAP	OLTP
Purpose	Decision Support	Transaction Processing
Query	Complex, Ad hoc	Structured and Repetitive
Data Model	Multidimensional	Entity-Relationship
Information	Summarized Data	Detailed Records
Size	Megabytes to Gigabyes	Gigabytes to Terabytes

As you can see in the table, the two systems have different purposes and characteristics. OLAP with DW stores summarized and aggregated data whereas OLTP with traditional database deals with detailed transaction data. There is a conspicuous difference between two systems in the viewpoint of data model. OLAP adopts multidimensional data model, which allows users to view and aggregate a great deal of data multidimensionally. The suitable database schema for OLAP system is star schema or snowflake schema, where fact and dimension tables exist and user can specify necessary dimension and analyze data. Fig. 1 shows an example of star schema and snowflake schema. Dimensions with hierarchies are decomposed to save storage in snowflake schema whereas dimension tables remains as flat denormalized tables in star schema [3].

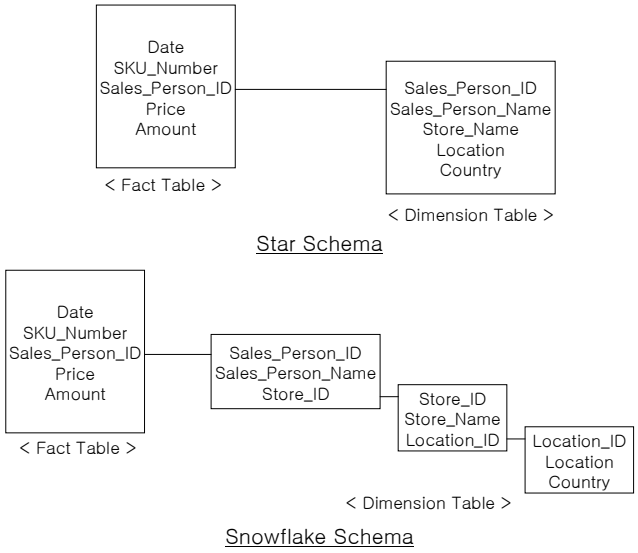

Fig. 1 Star Schema and Snowflake Schema

3. Information Integration Methodology

3.1 Integration Flow

Heterogeneous data from various kinds of data sources pass by four stages: preprocess stage, data warehouse stage, OLAP stage, and documentation stage. Through those stages, the data of different format and generation cycle are integrated and associated with one another, and converted to meaningful information for decision makers. Fig. 2 shows the overall structure of our methodology, where we can see the data flow with their association and integration. In this section, detail explanation will be given about the four stages.

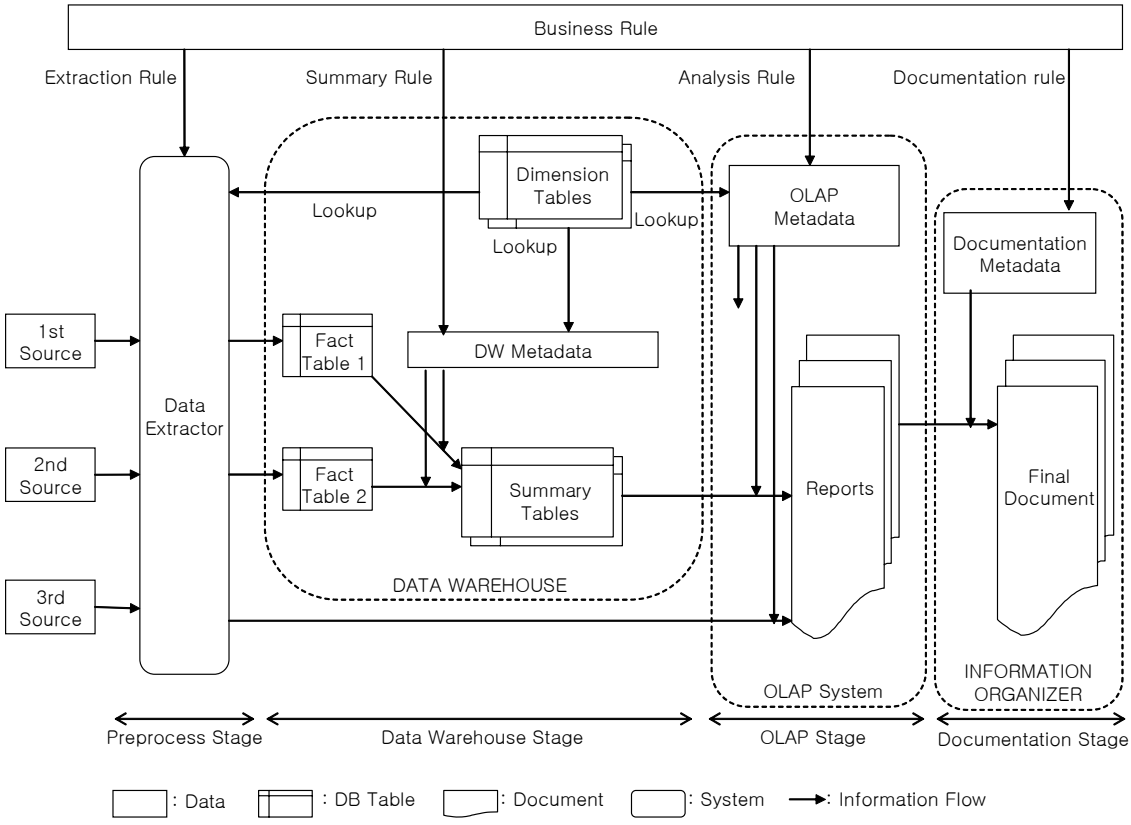

Fig. 2 Information Flow and Conversion

(1) Preprocess Stage

On the left side of Fig. 2, we can find three data sources. Only three sources were presented in the figure for the convenience of explanation, but in real business world, there can be much more data sources. When the data were originated from different organizations, the values have varying granularities, which means there are varying degrees of precision among the values [4]. In addition, the generation cycle of each data varies a lot, so the system has a difficulty to utilize the data timely. For arranging the variety of precision and cycle, data extractor has the extraction rule that was defined by business rule and extracts data from many data sources. To send appropriate data from data sources to data warehouse, it looks up dimension tables of data warehouse and fetches only the data of which dimensions are specified in dimension tables. It also filters out the data of which values are absurdly high or small. The extraction rule also defines extraction schedule and data format, so the data of different format and generation cycle are unified after passing this stage.

(2) DW Stage

In data warehouse stage, the data from data sources are stored and summarized in fact tables and summary tables. At first, the raw data from data sources are loaded in fact tables, then they are summarized several times and stored in summary tables finally. During the summation, the values are summarized and aggregated multidimensionally as the users want. For example, a summary table may contain time-series sales data regardless of customers, and another may reserve total sales amount by customers regardless of sales dates. For the multidimensional aggregation, we commonly use SQL(structured query language) as DW metadata, where how to handle and calculate values in each table is defined. Dimension tables play important role in other stages as well as DW stage. They constitute star or snowflake schema with fact and summary tables in DW, and give useful information to data extractor in preprocess stage and metadata in DW and OLAP stages.

(3) OLAP Stage

OLAP system extracts useful information and analyzes it. The system takes full advantage of the multidimensional characteristics of DW and generates various kinds of reports that the users want to see. The OLAP system includes OLAP metadata, which contains analysis rule given by business rule. There can be many summary tables in DW and many attributes for dimension or fact in each table. The metadata defines which fields to utilize and how to analyze them. When the metadata analyzes data and generates reports, it looks up dimension tables. OLAP system can also contact to data sources directly without data extractor, and fetch data in the data sources. It sometimes generates reports by using the data in DW and raw data source at the same time.

(4) Documentation Stage

In documentation stage, the information organizer also has documentation metadata, where how to user-friendly represent the reports generated in the previous stage. The documentation rule in documentation metadata was given by business rule as well, and it gathers the analysis results from OLAP system and reorganizes them for end-user to view in easy way. As the OLAP system uses a few summary tables for generating a report, the information organizer uses a few reports for making a screen. Not only it collects the information that the users want to see, it also supplies the comprehensible interface for the users to access to the analysis results and view the data cubes (summary tables) multidimensionally.

3.2 Remarks about Data Conversion

The raw data from data sources are converted in preprocess stage and data warehouse stage. The stages can change the format of the data (e.g., from 20020315 to March 15, 2002), and can consider the value of dimensions or facts, and exclude the records of which values are out of our consideration (e.g., excluding the sales record of which amount is negative). There can be two kinds of conversion(or cleansing) in the proposed methodology. One is low-level or operational cleansing, and the other is high-level or strategic cleansing. The former is performed in preprocess stage, whereas the latter is performed in data warehouse stage. The operational cleansing is carried out to eliminate unnecessary data. The cleansing can be applied to the value of dimension or fact. For example, consider the situation where some sales data of pear are mixed in the raw data sources when the system is originally designed for analyzing the sales of apple. In that situation, we must eliminate the data about pear sales, so we have to examine the dimension value of sales record and determine whether the product is apple or not. The value of facts can also be examined in preprocess stage and eliminated beforehand. For example, if the trade amount represented in the unit of box is 3.5, the data is absolutely wrong, so we must exclude it. Like these, we apply operational cleansing usually when the original data have some errors. On the other hand, the strategic cleansing is applied when the data itself has no problem but we decided to exclude the data that is out of our consideration. Likewise in the operational cleansing, the strategic cleansing can be applied to dimension or fact. Consider the raw data source contains the records of apple sales, and the apples are sold to an individual customer or a corporation (e.g., an apple processor). If our information service strategy is to describe sales trend to individual customers, we have to examine the customer field and exclude the sales data to corporation. The cleansing can also be applied to facts fields. If we decided to contain the sales data of only a reasonable price in data warehouse, we inspect the price value of each sales record and can exclude the data of which price is too high or low. Thus, strategic cleansing is applied when the data has no defect but it is needless on the purpose of decision making or can distort the analysis result.

4. Application for Apple Distribution

4.1 Apple Distribution Overview

Apple is cultivated in the whole area of Korea, and the apple producers sell the apples in various kinds of unit at many wholesale markets. Usually the apple is packed up in a box but the number of apple is different to one another as the size of each apple. Actually, the weight of each box is various, so the price of each box varies a lot according to the packaging unit. In the wholesale market, the produced apple is put up for auction, and the wholesale dealers bid a price. So we can say the market participants at auction market are producers, wholesale dealers, and market provider. The market provider, which is called wholesale corporation, mediates each auction, and provides the information about producers, dealers, and the qualities of produced apples. At the perspective of apple trade information, the auctions are carried out at the supervision of corporation, and the data of each auction is recorded by corporation. There are many corporations in a wholesale market, and apple producers from various locations carry the produced apple to the market. The reason why the producers don't prefer only the nearest wholesale market is that the price levels are different according to the market. In the remainder of this chapter, an application of our methodology to apple distribution is presented. A data warehouse for the data about apple distribution is proposed, and the data conversion procedure from data sources to the screen for end-user is described. The proposed application can be used for decision support of market participants and government policy.

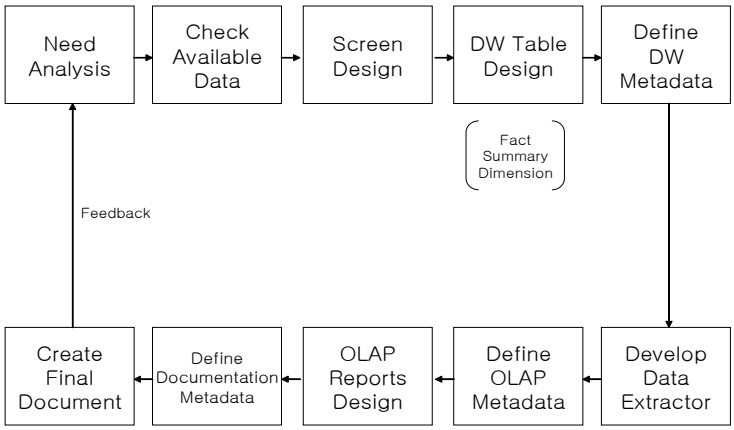

Fig. 3 DSS Development Procedure

4.2 Development Procedure

Fig. 3 shows the overall procedure of the development of the decision support system for the apple market participants concerned. The procedure consists of ten steps, and the detail will be explained in this sub-section.

(1) Need Analysis

The first step is to recognize the need of the end-users. There can be three kinds of users about apple distribution, and each user wants to obtain different information. The developer and system designer have to know which information they want to see mainly. They also have to know what information to be served in consecutive order for decision making.

(2) Check Available Data

Sometimes the request by the user cannot be satisfied because of the lack of information in raw data sources. When we develop the system, we have to examine raw data sources and have to know what kind of data can be obtained from the sources. After that, we decide which information to incorporate in the system and which request not to satisfy.

(3) Screen Design

After verifying the data availability, we design the screens that the end-user will see. In a screen, the necessary information must be presented. The user interface such as input box, combo box, or button must be provided user-friendly. Not only the design in one screen is important, but the whole screen design such as the sequence of all screens is also important because the decision maker refers the screens, so the sequence of screens has to be in accord with the sequence of information considered by decision maker.

(4) DW Table Design

We used relational database for data warehouse, and made star schema for fact, summary, and dimension tables. After defining each screen, we can know which dimensions and facts will be used in each screen. So, when we design the fact and summary table, we have to define each field properly in each table so as to lessen the database operation

time. Dimension tables must also be designed necessarily and sufficiently for each screen and decision making.

(5) Define DW Metadata

DW metadata is expressed in SQL. In a SQL, source tables and target table is described. Which field to read in source table, how to integrate the values, and which field to write at in target table are also defined in the metadata. An example of SQL for calculating the moving average for the last 15 days follows. Self join is used for obtaining moving average.

```
insert into TARGET_TABLE
(
        YYYYMMDD,
        CLASS_CODE,
        CORP_NAME,
        PRODUCT_UNIT,
        UNIT_CODE,
        B15DAY_PRICE,           --Moving Average Price
        B15DAY_PRICE_CPR,       --Today's Price to Moving Average Price
        B15DAY_AMT,             --Moving Average Trade Amount
        B15DAY_AMT_CPR          --Today's Trade Amount to Moving Average Trade Amount
)
select
        a.YYYYMMDD,
        a.CLASS_CODE,
        a.CORP_NAME,
        a.PRODUCT_UNIT,
        a.UNIT_CODE,
        sum(b.AVG_PRICE * b.TRADE_AMT) / sum(b.TRADE_AMT) B15DAY_PRICE,
        (avg(a.AVG_PRICE) - B15DAY_PRICE) * 100 / B15DAY_PRICE B15DAY_PRICE_CPR,
        sum(b.TRADE_AMT) / 15 B15DAY_AMT,
        (avg(a.TRADE_AMT) - B15DAY_AMT) * 100 / B15DAY_AMT B15DAY_AMT_CPR
from SOURCE_TABLE a, SOURCE_TABLE b
where
        a.YYYYMMDD between convert(CHAR(8), dateadd(DD, +1, b.YYYYMMDD), 112)
                       and convert(CHAR(8), dateadd(DD, +15, b.YYYYMMDD), 112)
        and a.YYYYMMDD <= convert(CHAR(8), today(*), 112)
        and a.YYYYMMDD >= convert(CHAR(8), dateadd(dd, -60, today(*)), 112)
        and a.CLASS_CODE = b.CLASS_CODE
        and a.CORP_NAME = b.CORP_NAME
        and a.PRODUCT_UNIT = b.PRODUCT_UNIT
        and a.UNIT_CODE = b.UNIT_CODE
group by
        a.YYYYMMDD,
        a.CLASS_CODE,
        a.CORP_NAME,
        a.PRODUCT_UNIT,
        a.UNIT_CODE;
commit;
```

(6) Develop Data Extractor

Ardent DataStage is used for data extractor. In development of data extractor, we consider the characteristics of various data sources, data generation cycle of each source, and operational data cleansing rule explained in Chapter 3. Fig. 4 shows an example of data extractor design. In the figure, the data source named "Data_Source" is accessed via ODBC, and the data from the source pass through the transformer named "Transformation". Four dimension tables are looked up, and the data of which dimension values don't exist in the dimension tables are rejected. Approved data is contained in a text file named "Accepted_Data".

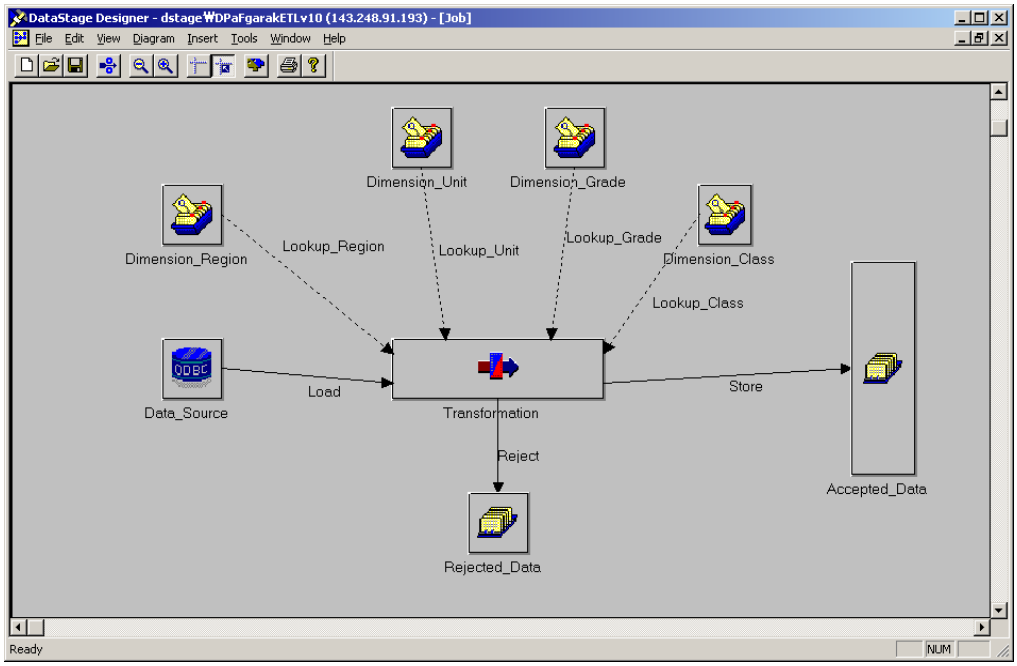

Fig. 4 Data Extractor Design

(7) Define OLAP Metadata

For OLAP Metadata definition, we used BusinessObjects Designer. The OLAP metadata defines the necessary tables in DW for analysis. An example of metadata definition is shown at Fig. 5. Note that new variables must be defined in OLAP stage because only the values in DW are not sufficient for decision support. In the figure, a new variable called "MAX-MIN" is defined, and it can be obtained by the expression "APPLE_CORP.MAX_PRICE – APPLE_CORP.MIN_PRICE", which means the difference between maximum price and minimum price.

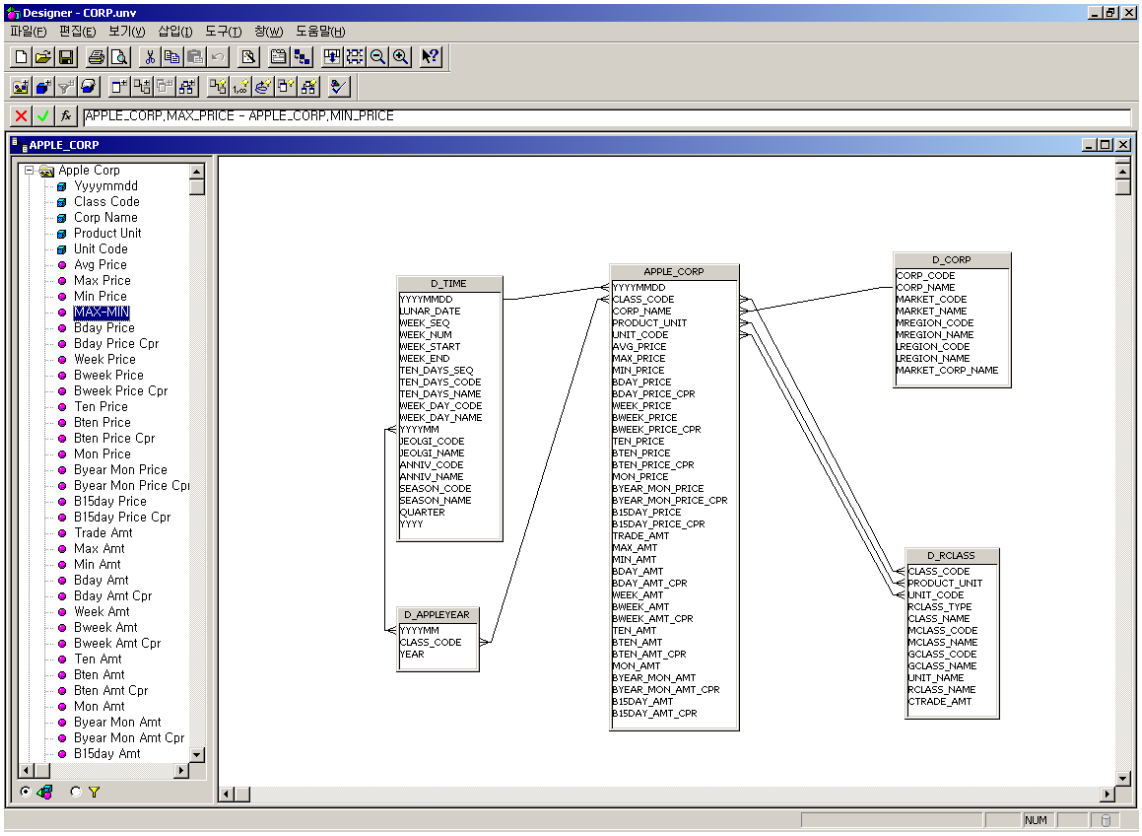

Fig. 5 OLAP Metadata Definition

(8) OLAP Reports Design

For the report generation, we used BusinessObjects. Based on the variables and metadata defined in the previous step, we defined each value in the report templates. Fig. 6 shows a sample screen shot for report design. In this step, we drew the variables from the left side to right side, and adjusted the expression format. In the report templates, we can append new fields that can be obtained by the mixture of the variables at the left side. The values seem to be fixed in the figure, but we can maintain the newest data by the daily refreshment of the report templates.

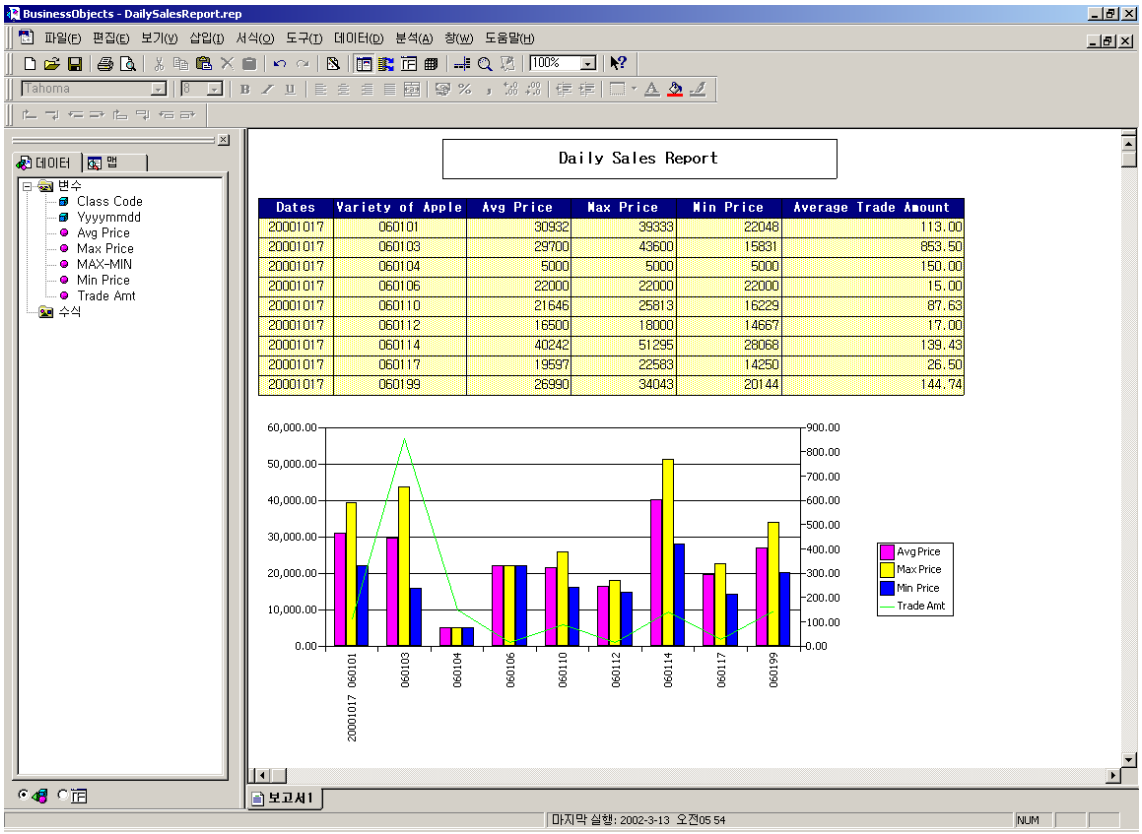

Fig. 6 OLAP Report Design

(9) Define Documentation Metadata

The generated reports in the previous step are difficult to see and utilize for decision making. In addition to that, end-users may want to see other information with a report in one screen. For an example, apple producer may want to see the daily price variation with the daily weather. To fulfill these requirements, we define documentation metadata, and the metadata daily updates the screens for end-user. HTML and web editor is a good tool for organizing information in the respect that all computers have web browser.

(10) Create Final Document and Feedback

After performing the previous nine steps, we make the data pass from the data sources through the four stages explained in Chapter 3. In each stage, we can schedule data update plans and make the newest data flow in the stage. We have to show end-users the final documents, receive feedback from them, and calibrate the system. To verify if each value is not absurd is also necessary for the system development.

5. Conclusion Remarks

In this paper, we proposed a data warehouse model for primary industry, especially for apple distribution. The data about primary industry varies in the format, generation cycle, and so on, and we are sure that this paper is the first attempt to establish data warehouse and OLAP technology for primary industry. Further researches needs to be carried out concerned with capturing user's request and reflecting it in the final document effectively. To develop a methodology of customizing the screen for each user is challenging as well.

References

[1] E. F. Codd, S. B. Codd, & C. T. Salley; Providing OLAP to User-Analysts: An IT Mandate, Technical report, E. F. Codd Associates, 1993

[2] Erik Thomsen; OLAP solutions: Building Multidimensional Information Systems, Wiley Computer Publishing, 1997

[3] Ralph Kimball; The Data Warehouse Toolkit, John wiley & Sons, 1996

[4] T. B. Pedersen, C. S. Jensen, & C. E. Dyreson; A Foundation for Capturing and Querying Complex Multidimensional Data, Information Systems, Vol. 26, pp. 383-423, 2001