

# Data Warehouse use Geographic Information System for Ornamentals

Benjapuk Jongmuenwai, Parinya Chinjoho and Sangpetch Prachai, *Member, IAENG*

**Abstract**—The purpose of developments data warehouse uses databases on the data storage and historical static data of ornamental. The ornamentals in, this process includes extracting the new transaction data from the database, transforming, aggregating and then updating it into a data warehouse. Users usually expect interface responses to their queries. These fundamental requirements present a challenge to the data warehouse due to the high volume and update complexity of data warehouse refresh mechanism.

The geographic information system (GIS) is extensively used in various application domains, ranging from ornamental analysis and route planning for searching data with ornamental. Nowadays, GIS need to analyze their data with information into database are, represented not only as attribute data, but also in maps. Users are integrate, their map and website. We input, the address and the problem location appear. Therefore, we will present an overview of data warehouse and GIS which includes its components, the categories of its ornamental and changing dimension which will temporally, support a data warehouse. A geographic information system is widely developed by the availability of website data in real-time. It has statistical methods use data which is ornamentals variable and records more than 100. Finally, we discuss future directions in the field, including OLAP analysis over raster data.

**Index Terms**—Ornamentals, Data Warehouse (DW), Geographic Information System (GIS)

## I. INTRODUCTION

In a 2003 IT spending survey, 45 % of American companies that participated indicated that their 2003 IT purchasing budgets had increased compared with their budgets in 2002. Among the respondents database applications ranked top in areas of technology being built or had been implemented, with 42 % indicating a recent implementation [9]. The fast growth of databases enables companies to capture and store a great deal of business operation data and other business-relate data.

The data is stored in the databases is either historical or operational, are considered corporate resources and an asset that must be manage and used effectively. The demands are in both internal and external information which are growing rapidly.

Manuscript received January 8, 2016; revised January 20, 2016. This work was supported by grants from the research of Thailand and Nakhon Ratchasima Rajabhat University.

Benjapuk Jongmuenwai, Informatics Program, Nakhon Ratchasima Rajabhat University, 340 Suranarai road, Tumbon Naimaung, Maung District, Nakhon Ratchasima 30000, Thailand. Tel/Fax: 044-009-009/3240 (Email: benjapuk.j@nrru.ac.th).

Parinya Chinjoho, Informatics Program, Nakhon Ratchasima Rajabhat University, 340 Suranarai road, Tumbon Naimaung, Maung District, Nakhon Ratchasima 30000, Thailand. Tel/Fax: 044-009-009/3240 (Email: parinya.c@nrru.ac.th).

The rapidly growing are demand to analyze business information quickly led to an emergence in data warehousing [3]. Starting in the mid-to-late 1990s, data warehousing became one of the most important developments in the information systems field. It has been estimated that about 95 % of the Fortune 1000 companies either have a data warehouse in place or are planning to develop one [19]. Data warehousing are emerge from both business needs and technology advances. Since, the business environment has become more global, competitive, complex, and volatile, customer relationship management (CRM) and e-commerce initiatives are creating requirements for large, integrated data repositories and advanced analytical capabilities [8,16]. One of the reasons are data warehousing systems has been rapidly spreading over the last decade is due to their contribution to increasing the effectiveness and efficiency of decision-making processes within business and scientific domains [6]. The strategic are use of information from data warehousing can help to counter the negative effects of many of the challengers facing organizations. When data warehouse technologies are well position and properly implemented they can assist organizations by reduce business complexity, and discovering ways to leverage information for new sources of competitive advantage and business opportunities, and provide a high level of information readiness under conditions of uncertainty [12].

A data warehouse (or on a smaller-scale data mart) is a repository of data create to support decision making. Data are extract from source systems, cleaned/scrubbed, transformed, and placed in data stores [7]. The data warehouse, which normally includes both transaction and non-transactional data, is typically use as the foundation of a decision support system (DSS) that aims to meet the requirement for a business user community [11]. A data warehouse has data suppliers who are responsible for delivering data to the ultimate end users either through structured query language (SQL) queries or custom-built decision-support applications, including decision support system (DSS) and executive information systems (EIS). The data warehouse are derive, meaning integrate, subject-oriented, time-variant, and non-volatile [10]. By integrated, it means that a data warehouse is construct by integrate multiple heterogeneous sources and must put data from disparate sources into a consistent format, by subject-oriented, it means that a data warehouse are organize around some major subjects rather than concentrating on day-to-day operations and business transactions, by time-variant, it means that data are store to provide information from a historical perspective and a focus is given on change over time, may be at various points in time, by non-volatile, it means that a data warehouse are always a physically separate database storing data transformed from the application data find in the operational environment.

The data store are input data warehouse remain static [8, 11]. As for the statistics perspective, its emphasis is on validity because this perspective focuses on the mathematical soundness of data warehouse.

The propose are to create virtual glades of flowers using Kinect gestures. The user gestures are read [14]. Since the arrival of computer storage technology and higher level programming languages [10] organizations, especially larger organizations, have put enormous amount of investment in their information system infrastructures.

Therefore, this paper proposes data warehouse use geographic information system for ornamentals, under concept of storage data ornamental more than 100 record. The data warehouse will be propose in section 2. Section 3 describes procedure for data warehousing. Section 4 presents data mapping for implementation system and geographic information system (GIS) on website. Section 5 presents concept geographic information system. Finally, the conclusions and future work in section 6.

## II. DATA WAREHOUSE

The data warehouse are new discipline much to do with the fact, that once one database (the transactional) are clear separate from the other (the historical/analytical), they are quite different kind in Table I.

TABLE I  
Differences between transactional databases and data warehouse

	Transactional database	Data warehouse
Purpose	Daily operations. Support for the software applications.	Information retrieval, reports, data analysis.
Data characteristics	Data about the organization inner working, changing data, internal data, incomplete data.	Historical data, internal and external data, descriptive data
Data models	Relational, object-relational, normalized	Multidimensional, snowflake, partially denormalized.
Users	Hundreds or thousands, applications, operators, administrators	Dozens, managers, executives, analysts (farmers and explorers)
Access	SQL, Read and Write	SQL and specific OLAP operators (slice & dice, roll, pivot), read-only

The table I. shows difference between transactional database and data warehouse are environmental the other. Thus, general user are use data warehouse for analysts data management over much in organization. Moreover, data warehouse help to for executives are reports from data models and multidimensional descriptive data.

### A. Exploitation: OLAP Operators and Tools

A data model are comprises a set of data structures and a set of operators over these structures. In the previous section we are see that the data structures or data scheme in the dimensional model. The facts with their measures, and the dimensions with their hierarchies and attributes for each level. We are describe that a single “operator” define by just choosing a measure from the fact and a level for each dimension, forming a data cube, and then, selecting the values

of one or more dimensions. The query tools are data warehouse under the multidimensional model usually are the graphical interface, which allow the user to select the data mart, to pick one or more measures for the facts (the aggregated projection), to choose the resolution in the dimension hierarchies and to express additional conditions.

With the multidimensional model and this hybrid data cube/table representation in mind it is easier to understand some additional operators that are more common in OLAP tools, called OLAP operators can be describes in 1-4:

- 1) Drill: de-aggregates the data following are paths of one or more dimensions.
- 2) Roll: aggregates the data following are paths of one or more dimensions.
- 3) Slice & dice: selects and projects the data into one of both sides of the report.
- 4) Pivot: changes one dimension from one side of the report to the other (rows by columns).

The design is data warehouse and OLAP tools are intend to avoid the recalculation of the aggregates each time one of the previous operators is apply. The idea are precalculate aggregations two several levels of the hierarchies and so making the traversal through dimension paths more efficient show in figure 1.

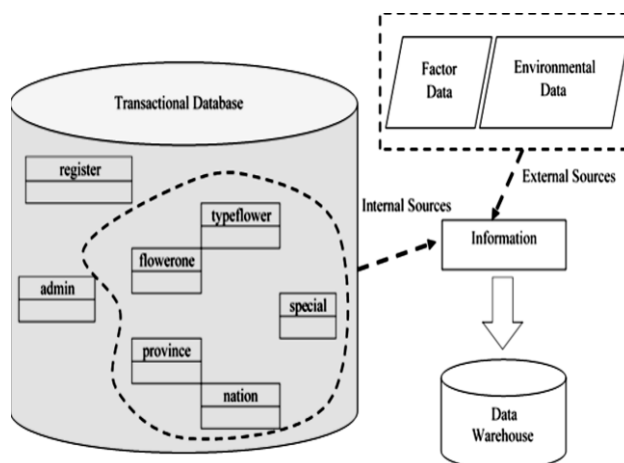


Fig. 1. Show architecture for Data warehousing.

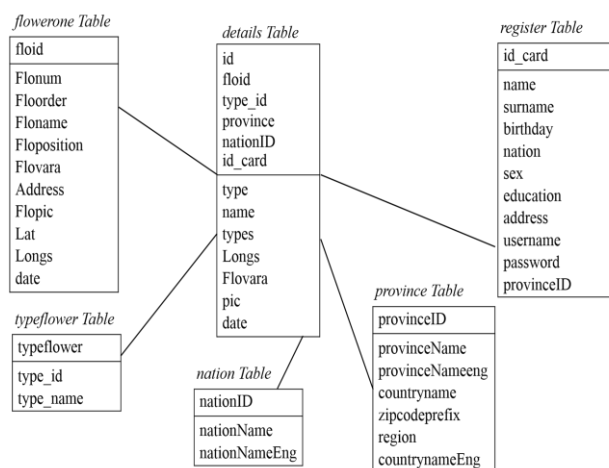


Fig. 2. Simple Star Schema for DW.

Figure 1. We are show architecture for data warehousing. The active data warehouse is provide an integrate information repository to drive strategic and tactical decision support within an organization including of internal source and external source. The managing are evolution of a DW architecture exchange in transactional database. The basic issues are should be solve include: (1) DW are model and metadata capable of representing and storing the history of evolution that concerns not only data but also data structures. (2) DW are simple star schema of representing and storing the dataset in database. In the example, the ornamental are store data and store table several of six table show in figure 2.

### B. Ornamental Dataset

The ornamental dataset uses in this provide by a major in Thailand. It's contains more than 100 records and more than 100 variables. The data are collect from January 2014 to December 2015. Each record has a class label to indicate its group ornamental status: either 'the north', 'the southern', 'the western', 'the north-eastern', 'the eastern' indicate data dictionary. Among these more than 100 records. For the purpose of ornamental classification, the 100 derive variables are choose to compute the model since they provide more precise information about group ornamental show figure 2.

In this section we illustrate how queries can be decompose into a number of smaller sub-queries that will act on fragments independently, with significant speedup. This subject is discussion in several works, which include our illustration is base on the query processing approach follow by the data warehouse architecture [4]. While, we also discuss other works on the next section such as querying metadata, version query, querying data warehouse in the other.

Figure 2 illustrates the basic architecture of DW for the share data warehouse, which can run the query is transform into high-level actions language by a query planner. We are describe basic query process functionality. For simplicity, start with the simplest possible example. Consider a single very large relation some 6 relation.

The number 3 step is specific actions are associate with the data warehouse that include:

- 1) The ADD\_COLUMN event: it was add to data structure in a table. For example a table flowerone, details, typeflower, register, nation and province. In this case, the column is also added to a corresponding table in DW, according to a mapping that relates data structure between DW store and metadata dictionary.
- 2) The DROP\_COLUMN event: it signalize that a column is drop from a data structure. The column retains it old values and new values are set to null. New\_table causes that a new version of the table is create without the drop column.
- 3) The ALTER\_COLUMN event: it is chang to a data structure in a table into multiple structures  $T_1, T_2, \dots, T_n$ , each of which stores the subset of rows from T. It modifies a new column or row is added to the table. This data merges the content of  $T_1, T_2, \dots, T_n$  before further processing.

While most SQL operations are ready in a way that may allow nodes to process data with exchange requirements, the join operator may occur in considerable data exchange

overheads if rows to be join are not co-locate or equi-partition. Thus, we are the join must match rows from two data set with the same value for a specific attribute.

For instance, the following SQL query is search and categorial for ornamental that include :

1) Attribute of a given name and a type or an equivalent attribute are results submission in T1.

T1. Algorithm searching results :

```

Algorithm T1 :
A = flower, B = typeflower and R = type_id, flonum
SQL = "SELECT * from A r, B p where r.R = p.R";
If (R > 0)
{
    SQL := " and r.R = " . $R;
Return;
}
if(R > 0)
{
    SQL := " and R = " . $R;
Return;
}
If ($flovare != ")
{
    SQL := " and flovare = " . $flovare . """;
Return;
}
If ($txt_n != ")
{
    SQL := " and floname like '%" . $txt_n . "%"";
Return;
}
    
```

2) Query submission are including of: categorial structure of ornaments, we bring to group original, type theoretic constructions, such as the ornamental algebra and the algebraic ornament in categorial terms and uncover the build blocks out of which they are carve out. We are interpret the mathematical properties of ornaments into type theory such as the pullback of ornaments to discover meaningful software artefacts.

Being at the interface between type theory and category theory, this paper targets both communities. To the type theorist, we offer a more semantic account of ornaments and use website story map thus gained to introduce new type theoretic constructions. We present a type theory, i.e. programming language that offers an interesting for categorial ideas, story map. Our approach can be summarize as categorial structure programming. The ornaments are an instance of that graphic between a categorial concept and an analysis, type theory presentation, the universe of ornaments. To help bridge the gap between type theory and category theory, we have to provide the type theory with examples of the categorial ornamental with the computational intuition behind the type theoretic objects.

### C. Categorial Map

In this section, we are definitions and results from category theory that will be use GIS. Therefore, we are well on the details. However, to help readers not familiar with these tools for analyze, we shall give many examples, thus provide an ornamental for these concepts.

1) Locally cartesian-close category.

There are locally cartesian-closed categories introduce. To give a categorial model of (extensional) dependent type theory. The key idea of that presentation is use of adjunctions

to model types [20].

2) Definition (Locally cartesian-closed category): A locally cartesian-closed category are that a pullback complete and such that, data each base change function. The index map are remains change real time, because of but, following data ornamental and analysis, we can be rename it follow index function. We obtain the following definition:

$$S: J \rightarrow SET$$

$$P: S_j \rightarrow SET$$

Note that, to group for cluster, we are count quantify type variables, such as  $j$  in the definition of SET.  $S_j$  definition of P. The data of S, P, is call a container and is denoted  $J \rightarrow SET, S_j \rightarrow SET$ . The class can containers detail index of ornamental for category in figure 3.

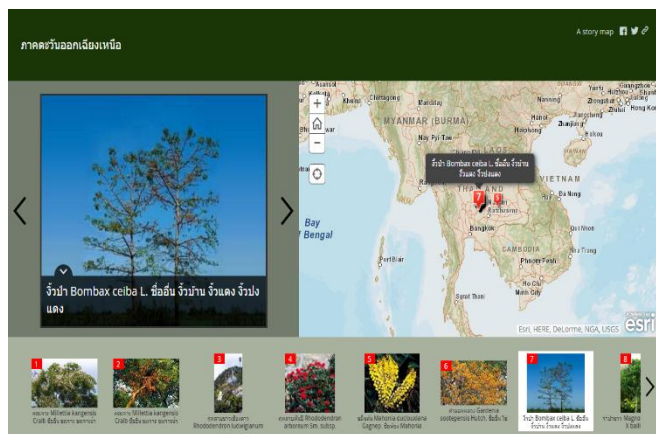


Fig. 3 DW for categorical map.

### III. PROCEDURE FOR DATA WAREHOUSING

If you want to submit your file with one column for storage, please do the following:

Input: The data set  $A=A_1, A_2, A_3, \dots, A_n$ , boundary  $b$

#### A. Querying Metadata

With the support of the DW a user can explicitly query metadata for the purpose of analyzing the change history of either the whole DW versions. The functionality of the language allows to execute two types of querying, namely:

- 1) A query searching for DW versions that include an indicated schema object or a dimension instance, and
- 2) A query retrieving the evolution history of an indicated schema object or a dimension instance. A query of the first type will be called a version query and a query of the second type will be called an object evolution query.

#### B. Version Query

In particular, a version query allows to search for DW versions that include:

- 1) An attribute of a given name and a type or an equivalent attribute (in the case of changing attribute names in DW versions) in a fact or a level table.
- 2) A table (either fact or level) of a given name or an equivalent table (in the case of changing table names in DW versions).
- 3) A table (either fact or level) that has a given exact or partial structure.
- 4) A dimension that has a given exact or partial structure.

- 5) A dimension instance that has a given exact or partial structure.

#### C. Querying Data Warehouses

In the area of querying temporal or multiversion data warehouses, to the best of our knowledge, only three contributions exist [15, 17, 18]. It allows to query either a DW version valid at a certain point in time or to query the latest DW version. It can provide consistent query results under changes to the structure of dimensions (adding a level) and dimension instances (adding a level instance, reclassifying a level instance). Additionally, OLAP allows to explicitly query metadata on DW changes. The support for handling missing fact or level attributes or domain changes between schema versions was not addressed in the publications.

The selection of DW versions is done based on a time interval provided by a user, Since queried DW versions may differ with respect to the structure of their dimension instances, in order to allow query result comparison, data coming from queried versions must be converted into the structure of a selected base version. Moreover, this a query language syntax in query processing the tables that difference with respect to their structures.

### IV. DATA MAPPING

Eder, J. et al. [1, 2]. The explore are possibilities impose by data transformations that require one-to-many mappings, i.e., transformations that produce several output tuples for each input tuple. This kind of operations is typically encountered in scenarios. Since relation algebra is not equipped with an operator that performs this kind of input-output mapping. Carreira et al. extend it by proposing a new operator called data mapper and explore its semantics and properties. In this discussion, we mainly focus on the work of a long version of a previous work [5]. The authors define the data mapper operator as a computable function mapping the space of values of an input schema to the space of values of an output schema. A second rule directs how a selection condition that uses attributes of the output of a data mapper can be translated to the attributes that generate them and thus be pushed through the mapper. A third rule deals with how projection can help avoid unnecessary computations of mappers that will be subsequently projected-out later. Finally, report some first results on their experimentation with implementing the data mappers in a real RDBMS. A more detail description of alternative implementations is given concern unions, recursive queries, table functions, stored procedures and pivoting operations as candidates for the implementation of the data mapper operator. The first four alternatives where use for experimentation in two different DBMSs and table functions appear to provide the highest throughput for data mappers [13].

### V. GEOGRAPHIC INFORMATION SYSTEM (GIS)

The ornamental data model are introduces in GIS. This map are design to be use as a basemap by GIS professionals and as a reference map by anyone. The map includes administrative boundaries, cities, water features, physiographic features, parks, landmarks, highways, roads, railways, and airports overlay on land cover and shade relief imagery for add context. The map provides coverage for the

world down to a scale of ~1:72k. Coverage is provide down to ~1:4k for the following areas: Australia and New Zealand, India, Europe, Canada, Mexico, the continental United States and Hawaii; South America and Central America; Africa, and most of the Middle East. Coverage down to ~1:1k and ~1:2k is available in select urban areas. This basemap is compile from a variety of best available sources from several data provider, including the U.S. Geological Survey (USGS), U.S. Environmental Protection Agency (EPA), U.S. National Park Service (NPS), Food and Agriculture Organization of the United Nations (FAO), Department of Natural Resources Canada (NRCAN), GeoBase, Agriculture and Agri-Food Canada, DeLorme, HERE, Esri, OpenStreetMap contributors, and the GIS User Community.

For more information on this map, including our terms of use, visit us online at url and show in Figure 4. (<http://www.arcgis.com/home/webmap/viewer.html>)

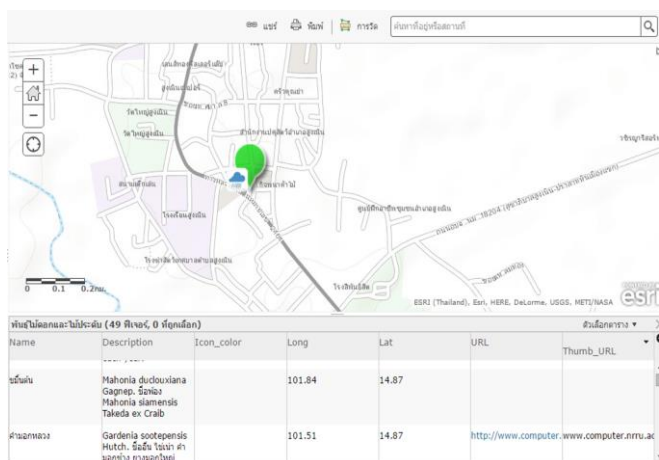


Fig. 4. DW ornamental screen showing present of GIS.

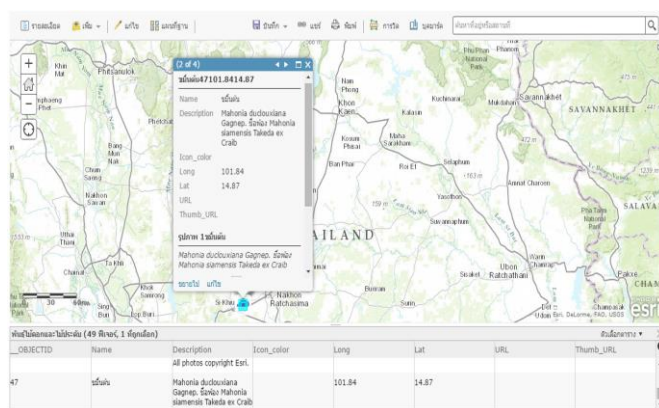


Fig. 5. DW ornamental screen showing details of GIS.

The model defines a GIS dimension as composed of a set of graphs, each one describing a set geometries in a thematic layer. A GIS dimension is, as usual in databases, compose of a schema and instances. Table II. Shows the instance of a GIS dimension: the bottom level of each hierarchy, denoted the Algebraic part, contains the infinite points in a layar, and could be described by means of linear algebraic equalities and in-equalities [5]. Below this part there is the Geometric part, which stores the identifiers of the geometric elements of the GIS, and is use to solve the geometric part of a query. Each point in the Algebraic part may correspond to one or more elements in the Geometric part (e.g., if more than one

polylines intersect with each other). Thus, at the GIS dimension instance level. We are define rollup relations (denoted  $r_L^{geo1 \rightarrow geo2}$ ). For instance,  $r_{L_{city}}^{Point \rightarrow Pg}(x, y, pg_1)$  says that, in a layer  $L_{city}$  a point (x,y) corresponds to a polygon identify by  $pg_1$  in the Geometric part.

Finally, there is the OLAP part for storing nonspatial data. This part contains the conventional OLAP structures, as define geometric part are associated to the OLAP part a function, (denoted  $\alpha_{L,D}^{dim Level \rightarrow geo}$ ). For instance,  $\alpha_{L,Icon\_color}^{Name \rightarrow na}$  associates information about a name in the OLAP part (Icon\_color) in a dimension Lat, Long to identifier of a polyline ( $n_a$ ) in a layer denoted ( $L_{icon\_color}$ ), which represents detail in the GIS.

Figure 5. Shows detail of ornamental for GIS dimension instance for Lat and Long in the dimension schema from table II. We can see that an instance of a GIS dimension in the website.

Finally, while all GIS software components are open source: the database, postgres, and its GIS extension GIS url (<http://www.arcgis.com/home/webmap/viewer.html>)

TABLE II.  
Category Instances

Category	Instance
OBJECTID	{ {_OBJECTID}, (OBJECTID <sub>1</sub> ), (OBJECTID <sub>2</sub> ), (OBJECTID <sub>n</sub> ) }
Name	{ {Name}, (Name <sub>1</sub> ), (Name <sub>2</sub> ), (Name <sub>n</sub> ) }
Description	{ {Description}, (Description <sub>1</sub> ), (Description <sub>2</sub> ), (Description <sub>n</sub> ) }
Icon_color	{ {Icon_color}, (Icon_color <sub>1</sub> ), (Icon_color <sub>2</sub> ), (Icon_color <sub>n</sub> ) }
Long	{ {Long}, (Long <sub>1</sub> ), (Long <sub>2</sub> ), (Long <sub>n</sub> ) }
Lat	{ {Lat}, (Lat <sub>1</sub> ), (Lat <sub>2</sub> ), (Lat <sub>n</sub> ) }
URL	{ {URL}, (URL <sub>1</sub> ), (URL <sub>2</sub> ), (URL <sub>n</sub> ) }
Thumb_URL	{ {Thumb_URL}, (Thumb_URL <sub>1</sub> ), (Thumb_URL <sub>2</sub> ), (Thumb_URL <sub>n</sub> ) }

Example 1. Consider a fact table containing category instance in table II.

The GIS data model is basically compose of category instances, and the table of items. For example, our ornamental application above can include eight category schemas, OBJECTID, Name, Description, Icon\_color, Long, Lat, URL and Thumb\_URL. Each category schema is compose of a set attributes that describe it. An element in a category is denote a category occur, and the set of all occur in all category in an application is a category instance. A set of category instances for our running example is show in Figure 4. Where, for example, the category name is one occurrences. A value of the attribute name represents the geometric extension of the corresponding category occur (e.g., in the first tuple, Long and Lat can be point (101.84, 14.87)). Adding a OBJECTID interval to a category occur, produces an item. All other attributes are store elsewhere.

## VI. CONCLUSION

Data warehouse is an effective tool that enables an organization or the other to manage data, and to transform them into strategic decision and show information. The data that organizations store are considered ornamental. Data controllers cannot do anything until they input into the



data warehouse to Fig 4. DW ornamental screen showing present of GIS. Fig. 5. DW ornamental screen showing details of GIS corporate information into databases. Therefore, internal and web technology will become the interface for users, to allow users to open web browsers in their data warehouse worldwide on public and private networks, and eliminate the need to replicate data across diverse geographic locations. Thus strong data warehouse management sponsorship and an effective administration team may become a crucial factor to provide an organization with the information service.

Today's data warehouse is limited to storage of structure data in the form of records, field, and database. Unstructured data, such as multimedia maps, graphs, pictures, sound, and video files are demand increase in organizations. How to manage the storage and retrieval of unstructured data and how to search for specify data item set a real challenge for data warehouse administration and management. Alternative storage, especially the near-line storage, which is one of the two forms of alternative storage is considered to be one of the best for the future solution for managing the storage in data warehouse. This recent growth in website use and advances in e-business application push the data warehouse from the back office, where it is accessed by only a few businesses employees, to the front lines of the organization, where all users can use it. Additionally, building distribute warehouses, which are normally call data mart, will be a technical advance in data warehouses including an increasing ability to process, automatic information delivery, greater support of object extensions, very large database support, and user friendly web enable analysis applications. These are capabilities that should make data warehouses of the future more powerful and easy to use, which will further increase the importance of data warehouse technology for education or business.

#### REFERENCES

- [1] Eder, J., and Koncilia, C., *Changes of Dimension Data in Temporal Data Warehouses*. Proceedings International Conference on Data Warehousing and Knowledge Discovery, Munich, Germany, 2001, pp. 284-293.
- [2] Eder, J., Koncilia, C., and Morzy, T., *The COMET Metamodel For Temporal Data Warehouses*. Proceedings International Conference on Advanced Information Systems Engineering, Toronto, Canada, 2002, pp. 83-99.
- [3] Finnegan, Murphy, and O'Riordan., *Challenging the hierarchical perspective on information systems: Implications from external information analysis*. Journal of Information Technology, 1999, vol. 14(1), pp. 23-37.
- [4] Furtado Pedro., *A Survey of Parallel and Distributed Data Warehouses*. 2011, pp. 148-169. Universidade Coimbra, Portugal.
- [5] Gomez, Kuijpers, and Vaisman., *A State-of-the-Art in Spatio-Temporal Data Warehousing, OLAP and Mining*. 2011, pp. 200-236.
- [6] Golfarelli, M., Lechtenborger, J., Rizzi, S., and Gossen, G. *Schema versioning in data warehouses: Enabling cross-version querying via schema augmentation*. Data & Knowledge Engineering, 2006, vol. 59, pp. 435-459.
- [7] Gorla, N., *Features to consider in a data warehousing system*. Communications of the ACM, 2003, vol. 46(11), pp. 111-115.
- [8] Han, J., and Kamber, M. (2006). *Data mining: Concepts and techniques* (2<sup>nd</sup> ed.). San Francisco, CA: Morgan Kaufmann Publisher.
- [9] Information Technology Toolbox. (2004). "2003 ITtoolbox spending survey. Available: <http://data-warehouse.ittoolbox.com/research>.
- [10] Inmon, W.H., (2002). *Building the data warehouse* (3<sup>rd</sup> ed.). New York, NY: John Wiley & Sons Inc.
- [11] Khan, A. (2003). *Data warehousing 101: Concepts and implementation*. Electronic version, Online Bookstore: iUniverse.

- [12] Love, B., *Strategic DSS/data warehouse: A case study in failure*. Journal of Data Warehousing, 1996, vol. 1(1), pp. 36-40.
- [13] Moreno, F., Fileto, R., and Arango, F., *Season queries on a temporal multidimensional model for OLAP, Mathematical and Computer Modelling*, 2010, vol. 52(7-8), pp. 1103-1109.
- [14] Roiger, R.J., and Geatz, M. W. (2003). *Data mining: A tutorial-based primer*. New York: Addison Wesley.
- [15] Rizzi and Golfarelli., *Temporal Data Warehousing: Approaches and Techniques*. 2011 . pp. 1-18.
- [16] TOUMI Lyazid, MOUSSAOUI Abdelouhab, and UGUR Ahmet. *A linear programming approach for bitmap join indexes selection in data warehouse*. ScienceDirect, 2015, vol. 52, pp. 169-177.
- [17] Vaisman, A., and Mendelzon, A. (2001). *A Temporal Query Language for OLAP: Implementation and a Case Study*. Proceedings DBPL.
- [18] Vassiliadis Panos., *A Survey of Extract-Transform-Load Technology*. 2011, pp. 171-199.
- [19] Wixon, B.H., and Watson, H., *An empirical investigation of the factors affecting data warehousing success*. MIS Quarterly, 2001, vol. 25(1), pp. 17-41.
- [20] Wixon, B.H., and Watson, H., *Mathematical Proceedings of the Cambridge Phiosophical Society*. (1984). Cambridge Philosophical Society, vol. 95, pp. 33-48.