UDC 004.65

Kudrenko S.A., Ph.D.
Pushkin Yu.A.
National Aviation University, Kyiv

MODEL AND SCHEME FOR ORGANIZING DATA WAREHOUSES

In traditional architecture, there are three general data warehouse models: virtual warehouse, data mart, and enterprise data warehouse:

A virtual datastore is a collection of separate databases that can be shared so that a user can efficiently access all of the data as if it were stored in a single datastore;

The data mart model is used for reporting and analyzing specific business lines. In this warehouse model, aggregated data from a number of source systems related to a specific business area, such as sales or finance;

The enterprise data warehouse model assumes storage of aggregated data that covers the entire organization. This model views the data warehouse as the heart of the enterprise information system with integrated data from all business units.

Star and snowflake schemas are two ways to structure your data warehouse.

A star schema has a centralized data store that is stored in a fact table. The schema splits the fact table into a series of denormalized dimension tables. The fact table contains the aggregated data that will be used for reporting, and the dimension table describes the stored data.

Denormalized projects are less complex because the data is grouped. The fact table uses only one link to attach to each dimension table. The simpler star schema design makes it much easier to write complex queries. A snowflake schema is different in that it uses normalized data. Normalization means organizing data efficiently so that all data dependencies are defined and each table contains a minimum of redundancy. In this way, the individual dimension tables are forked into separate dimension tables.

The snowflake scheme uses less disk space and better preserves data integrity. The main drawback is the complexity of the queries required to access the data – each query must go through multiple table joins to get the corresponding data.

There are two different ways to load data into the warehouse: ETL and ELT.

ETL (Extract, Transform, Load) first retrieves data from a pool of data sources. The data is stored in a temporary staging database. Transformation operations are then performed to structure and transform the data into an appropriate form for the target data warehouse system. The structured data is then loaded into the warehouse and ready for analysis.

In the case of ELT (Extract, Load, Transform), data is loaded immediately after being extracted from the original data pools. There is no staging database, which means that the data is immediately loaded into a single centralized repository.

The data is transformed in a data warehouse system for use with business intelligence and analytics tools. The structure of an organization's data warehouse also depends on its current situation and needs.

The basic structure allows end users of the warehouse to directly access, report, and analyze summary data from the source systems. This structure is useful for cases where data sources come from the same types of database systems.

Staging area storage is the next logical step in an organization with heterogeneous data sources with many different types and formats of data. The staging area converts the data into a generalized, structured format that is easier to query using analysis and reporting tools.

One type of middleware is adding data marts to a data warehouse. Data marts store summary data for a specific industry, making this data readily available for specific forms of analysis.

For example, adding data marts can enable financial analysts to more easily query detailed sales data and predict customer behavior. Data marts facilitate analysis by tailoring data specifically to meet the needs of the end user.

REFERENCES

1. Buyya R., Broberg J., Goscinski A., Cloud Computing. Principles and Paradigms, John Wiley & Sons, Inc., New Jersey. – 637 p.

2. Focus Group on Cloud Computing Technical Report, 2012, Part 1: Introduction to the cloud ecosystem: definitions, taxonomies, use cases and high-level requirements, ver.1.0. – pp.62.