The Data warehouse Toolkit – Guide to Dimensional modelling

Data Capture and Data Analysis

Information is a very important asset and can be used either for operation record keeping or for analytical decision making. Some common attributes of an operational system are they are optimized to process transactional data very efficiently and that they usually don’t keep historical information. On the other hand, the Business intelligence systems are used to measure and evaluate performance. History is normally preserved.

What is the goal of a data warehouse?

To provide a scalable, efficient and easy way to access information

The way the data are structured should be understandable and intuitive to business users and not only to developers. In addition, the users should be able to use tools to access the data without large wait times – querying of the database should be fast

In addition, they data should be clean, consistent and credible – data quality is a very important aspect

There should be a single version of truth of the data

A data warehouse can help decision making – it’s a decision support system

But to be successful – the business users should actively use it and the management of a business or organization should perceive it as strategic.

Dimension modelling can provide data that are easy to understand and can improve query efficiency

ETL -The extract transform and Load system

Database systems, a practical approach

*Business intelligence (BI) is an umbrella term that refers to the processes for collecting and analyzing data, the technologies used in these processes, and the information obtained from these processes with the purpose of facilitating corporate decision making*

A single integrated view of the data is presented to the user

**Data Integration (the data warehouse etl tooklit)**

Significant Data Integration sometimes can happen before the data are imported in a data warehouse, for example when a business has an ERP. But even in those cases, it is likely that other systems and data sources exist outside of the ERP system

A data warehouse is not a project, it consists of many projects. Each data mart is a separate project with its own timeline - each data mart contains conformed dimensions so that each integrates into a single cohesive unit, the enterprise data warehouse. A better term is that a data warehouse is a process.

ETL

ETL stands for Extract, Transform and Load and it’s a term widely used in data warehousing. The ETL system is the backbone of a data warehouse as it is responsible for the extraction of the data from external sources, the cleaning of the data and the loading of the data into the data warehouse. The design and implementation of an ETL system is a complex project divided into many subtasks. There are many different methodologies, tools and technologies for ETL development and implementation. ETL design is a significant part of the Business Intelligence lifecycle (Moss and Atre Shaku, 2003). Before implementing any data flows, The ETL Team should take into consideration the business requirements and plan the ETL solution accordingly.

The Extract Step: The Extract Step of the ETL process should be planned based on the business needs. The business requirements set by end users define the data sources or specific entities and attributes of an operational system that need to be considered for integration into the data warehouse. Data Sources or other database objects that are not useful for analysis should not be considered.

The Transform Step: Part of the Transform step is also driven by the business needs. Specific Business rules are applied at this step. The data are cleaned, conformed and ready to be imported in the data warehouse.

The Load Step: The end users want to have easy access to information and they should be able to understand the underlying data model. The Business Intelligence Team should chose a data model that is simple, scalable and efficient and the ETL Team needs then to design the ETL processes to load the data efficiently in the data structure that has been implemented based on the chosen data model.

An additional business requirement that affects all of the steps above is the data latency requirement (Kimball Ralph, 2004). The frequency the data warehouse needs to be updated with fresh data is one of the most important aspects to consider by the design of an ETL solution.

To conclude, it is obvious that the ETL design and implementation is driven by the business requirements

The business requirements set by end users define the data sources or specific entities and attributes of an operational system that need to be considered for integration into the data warehouse. The end users also define the business rules that should be applied in the transformation stage.

(Kimball Ralph, 2004)

Capture changed data

Use audit fields / ensure that these fields are dependable and don’t have null values

The logic is to to compare the last modified date and

Time of each record to the maximum date and time that existed during the

Previous load and take all those that are greater.

Database Log Scraping or Sniffing /need to check this, it is used for real-time ETL. Drawback: The Log can be truncated and all transactions will be lost (p.107)

Avoid Time Extracts (take all rows that we inserted or modified getdate()-1) because if the ETL fails overnight then we may lose records

Process of Elimination (compare source tables and target table row by row to identify changes –not efficient)

Incremental load

Real Time Streaming ETL systems

From a technical architecture perspective, it has the potential to change the big-bang approach needed during the  
nightly batch ETL load windows to a continuous ETL-like flow throughout the day.

First generation of real/time ETL / the ODS, operational data store

Second generation: 2 fact tables, one real-time and one static p.427 – not sure what is the benefit of this

Point to Point vs Hub and Spoke solutions /Important! (Application integration)

Micro batch ETL / like conventional ETL but higher frequency

For micro batch, we have the following methods for identifying changed records:

* Timestamps/audit fields. These fields should have index to improve performance but the index increases operational overhead on inserts and updates
* ETL Log tables. A trigger is created in the OLTP database and populates a etl log table with the unique identifier of the rows that have changed or have updated. The etl process is joining then the etl log table with the source table and extracts the rows. The overhead on the OLTP system is reduced as the trigger driven Inserts are not heavy.
* (DBMS) log scrapers. Log scrappers can find the SQL statements of Inserts and Updates in the log files of the database and apply directly the changes to the target tables
* Network Sniffers

**Batch ETL** : simplest approach for delivering near real-time data warehousing reporting (one direction, no need to import back to source system)

**EIA: Enterprise Application Integration**

set of adapter and broker components

that move business transactions, in the form of messages, across the various

systems in the integration network

Publish&Subscribe Technology

**Capture, Transfrom, Flow**

The application layer of the transactional applications is bypassed. Instead, direct database-to-database exchanges are executed. Transactions, both new facts and dimension changes,  
can be moved directly from the operational systems to the data warehouse  
staging tables with low latency, typically a few second

**Enterprise Information Integration**

Real/time Reporting,transformations on the fly, no data warehouse

To read: https://tdwi.org/articles/2006/10/23/enterprise-information-integration-a-technology-for-providing-integrated-views.aspx