Data Warehouse

- Database designed for data analysis rather than business transaction processing.
- It contains historical data from operational systems.
- It separates the workload associated with the data analysis from that associated with processing business transactions.
The technological environment

- The database (relational)
- An Extraction - Transformation - Loading (ETL)
- An OnLine Analytical Processing (OLAP)
- Tools for analyzing or delivering data to users

Extraction - Transformation - Loading (ETL)

- ETL is short for extract, transform, load, three database functions that are combined into one tool to pull data out of one database and place it into another database.
- Extract is the process of reading data from a database.
- Transform is the process of converting the extracted data from its previous form into the form it needs to be in so that it can be placed into another database. Transformation occurs by using rules or lookup tables or by combining the data with other data.
- Load is the process of writing the data into the target database.
- ETL is used to migrate data from one database to another, to form data marts and data warehouses and also to convert databases from one format or type to another.
OLTP (On-line Transaction Processing)

- OLTP (On-line Transaction Processing) is characterized by a large number of short on-line transactions (INSERT, UPDATE, DELETE). The main emphasis for OLTP systems is put on very fast query processing, maintaining data integrity in multi-access environments and an effectiveness measured by number of transactions per second. In OLTP database there is detailed and current data, and schema used to store transactional databases is the entity model (usually 3NF).

OLAP (On-line Analytical Processing)

- OLAP (On-line Analytical Processing) is characterized by relatively low volume of transactions. Queries are often very complex and involve aggregations. For OLAP systems a response time is an effectiveness measure. OLAP applications are widely used by Data Mining techniques. In OLAP database there is aggregated, historical data, stored in multi-dimensional schemas (usually star schema).
OLAP Vs OLTP

We can divide IT systems into transactional (OLTP) and analytical (OLAP). In general we can assume that OLTP systems provide source data to data warehouses, whereas OLAP systems help to analyze it.

Architecture of a data warehouse
Warehouse Data Features

- Grouped by Subject
- Integrated
- Stables
- Historical data
- Variable granularity

Grouped by Subject

- Operational systems
  - Support business processes.
  - As a result, they contain the information required or produced by the process.
- Warehouses
  - The data warehouse structures (and data marts) group information by subject, regardless of the source operating system.
Integrated

- Operational systems
  - A single database is involved.
- Warehouses
  - Sources from different operational systems
  - External data sources.
  - Should integrate these different sources into a coherent form.
    - Name Conflicts
    - Conflicts of definition
    - Different units of measurement
    - Etc.

Stables

- Operational systems
  - Data is constantly being modified (additions, updates, withdrawals) at the discretion of operational transactions.
- Warehouses
  - The warehouse data are not intended for operational management.
  - Data are not updated to reflect operational transactions.
  - Data is never erased
Historical data

- **Operational systems**
  - Retains the current values of the situation.
  - Because of performance targets, historical data is removed (archived or purged).

- **Warehouses**
  - To support management analysis, and to find trends in behaviors and processes, historical values must be retained, not just current values.
  - As a result, the volume of data is increasing in a warehouse at a rate that may be greater than the operational system.

Variable Granularity

- **Granular computing (GrC)** is an emerging computing paradigm of information processing. It concerns the processing of complex information entities called information granules, which arise in the process of data abstraction and derivation of knowledge from information or data. Generally speaking, information granules are collections of entities that usually originate at the numeric level and are arranged together due to their similarity, functional or physical adjacency, indistinguishability, coherency, or the like.

- **Operational systems**
  - The information is kept at the finest level of detail.
  - When the need arises, the required data are totaled.

- **Warehouses**
  - Usually, the user starts by looking at the information at a summarized (aggregated) level.
  - When the need arises, a more detailed level of detail is explored.
  - It is therefore more efficient to retain aggregated information in the warehouse.
Synthesis of differences

<table>
<thead>
<tr>
<th></th>
<th>Operational system</th>
<th>Warehouse</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design</td>
<td>Normal shapes</td>
<td>Star pattern</td>
</tr>
<tr>
<td>Derived data</td>
<td>Few</td>
<td>Frequently</td>
</tr>
<tr>
<td>Historical data</td>
<td>Limited</td>
<td>Unlimited</td>
</tr>
<tr>
<td>Typical work</td>
<td>Known queries</td>
<td>Queries Ad Hoc</td>
</tr>
<tr>
<td>Typical request</td>
<td>Search for a concept</td>
<td>Runs millions of rows</td>
</tr>
<tr>
<td>Joins</td>
<td>Many</td>
<td>Limited</td>
</tr>
<tr>
<td>Index</td>
<td>Little</td>
<td>abundant</td>
</tr>
<tr>
<td>Inserting data</td>
<td>Live, by the user</td>
<td>In batch (ETL)</td>
</tr>
</tbody>
</table>

Cube

- A cube presents the aggregate data for each hierarchical level of each dimension of a diagram. The term “cube” is now devoted, but it does not refer to a structure purely three-dimensional, as defined by mathematics. The term represents the image **multi-dimensional vision**; some will even speak of a “hyper-cube” to imagery this aspect. There are several tool to explore and navigate through the cube data.
Star Schema

- The **star schema** is the simplest style of data mart schema and is the approach most widely used to develop data warehouses and dimensional data marts. The star schema consists of one or more fact tables referencing any number of dimension tables. The star schema is an important special case of the snowflake schema, and is more effective for handling simpler queries.

- The star schema gets its name from the physical model's resemblance to a star shape with a fact table at its center and the dimension tables surrounding it representing the star's points.

- **Star schemas are denormalized**, meaning the normal rules of normalization applied to transactional relational databases are relaxed during star schema design and implementation. The benefits of star schema denormalization are:
  - Simpler queries
  - Simplified business reporting logic
  - Query performance gains
  - Fast aggregations
Dimensions

- Contains business hierarchies and categories
- Provides the basis for business analysis
- Hierarchies allow you to zoom in (drill-down)
  - Annual, quarterly, monthly sales, etc.
- The categories characterize certain aspects of the data
  - Holidays, client types, etc.

Dimension (dimension table)

- Dimension tables usually have a relatively small number of records compared to fact tables, but each record may have a very large number of attributes to describe the fact data. Dimensions can define a wide variety of characteristics, but some of the most common attributes defined by dimension tables include:
  - **Time dimension** tables describe time at the lowest level of time granularity for which events are recorded in the star schema
  - **Geography dimension** tables describe location data, such as country, state, or city
  - **Product dimension** tables describe products
  - **Employee dimension** tables describe employees, such as sales people
  - Range dimension tables describe ranges of time, dollar values, or other measurable quantities to simplify reporting
  - Dimension tables are generally assigned a surrogate primary key, usually a single-column integer data type, mapped to the combination of dimension attributes that form the natural key.
Dimension (dimension table)

- The table is wide, which means many columns.
- Attributes are textual, few numeric values.
- Some attributes are not necessarily related to others.
- The concepts are denormalized to present a short path.
- Linked attributes can represent hierarchies.
- All proportions contain a smaller number of rows.

Measure

- Contain measurable items (metrics) about business areas.
  - Amounts (sales, costs), quantities, percentages, etc.
- They are targeted to the needs of business analysis.
Measure (Fact table)

- Fact tables record **measurements or metrics for a specific event**. Fact tables generally consist of numeric values, and foreign keys to dimensional data where descriptive information is kept. Fact tables are designed to a low level of uniform detail (referred to as "granularity" or "grain"), meaning facts can record events at a very atomic level. This can result in the accumulation of a large number of records in a fact table over time. Fact tables are defined as one of three types:
  - Transaction fact tables record facts about a specific event (e.g., sales events)
  - Snapshot fact tables record facts at a given point in time (e.g., account details at month end)
  - Accumulating snapshot tables record aggregate facts at a given point in time (e.g., total month-to-date sales for a product)
- Fact tables are generally assigned a surrogate key to ensure each row can be uniquely identified. This key is a simple primary key.

Measure (Fact table)

- Represents a certain level of detail
- The table is deep, not wide
- Contains metrics
  - completely additive
  - semi-additive
- Contains "holes"
  - May contain "degenerate" dimensions
The key to a dimension (problems)

- The key must represent the identity of the lowest granularity level of the dimension.
  - Dimension product -> product code
  - Customer dimension -> customer code
- Problems
  - If the product code is significant?
  - If the primary keys in the operating system are reused?

The key of a dimension (solution)

- Dimensions should never use the primary keys of operational systems.
- Solution
  - Using artificial keys (surrogate keys)
- Consequences
  - Table of correspondence between the primary key of the operational system and the artificial key of the warehouse.
  - Often one keeps the primary key in the dimension as an attribute.
The Key to a Measure

- Each dimension has a relation of 1 to many with the measure.
- The measurement table thus comprises each of the primary keys of the dimensions as a remote key.
- Solution
  - The primary key of the measurement is a complex key composed of all the remote keys of the dimensions.

Star Schema

```
<table>
<thead>
<tr>
<th>PRODUIT</th>
<th>TIME</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>ID</td>
</tr>
<tr>
<td>Code</td>
<td>Date</td>
</tr>
<tr>
<td>Name</td>
<td>Month</td>
</tr>
<tr>
<td>Model</td>
<td>Trimestre</td>
</tr>
<tr>
<td>Brand</td>
<td>Year</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>COMMANDE</th>
<th>PRODUIT</th>
<th>TIME</th>
<th>CUSTOMER</th>
<th>SELLER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID Product</td>
<td>ID</td>
<td>ID Time</td>
<td>ID Customer</td>
<td>ID Nom</td>
</tr>
<tr>
<td>ID Sales</td>
<td>$</td>
<td>$ Price</td>
<td>$ Profit</td>
<td>Region</td>
</tr>
<tr>
<td>Quantity</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CUSTOMER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID Code</td>
</tr>
<tr>
<td>Name</td>
</tr>
<tr>
<td>Address</td>
</tr>
<tr>
<td>ZipCode</td>
</tr>
<tr>
<td>City</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>SELLER</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID Nom</td>
</tr>
<tr>
<td>Territory</td>
</tr>
<tr>
<td>Region</td>
</tr>
</tbody>
</table>
```
Advantages of the star schema

- Easy to understand for the user
- Easier than a standardized model
- The user often makes his queries himself
- Easy navigation since there are fewer links
- Optimizes performance