Introduction to data warehousing

A Database Management System (DBMS) stores data in the form of tables, uses ER model and the goal is ACID properties. For example, a DBMS of college has tables for students, faculty, etc.

A Data Warehouse is separate from DBMS, it stores a huge amount of data, which is typically collected from multiple heterogeneous sources like files, DBMS, etc.

Data warehousing components

The major components of a data warehouse are as follows –

1) Data Sources – Data sources define an electronic repository of records that includes data of interest for administration use or analytics. The mainframe of databases (e.g. IBM DB2, ISAM, Adabas, Teradata, etc.), client-server databases (e.g. Teradata, IBM DB2, Oracle database, Informix, Microsoft SQL Server, etc.), PC databases (e.g. Microsoft Access, Alpha Five), spreadsheets (e.g. Microsoft Excel) and any other electronic storage of data.

2) Data Warehouse – The data warehouse is normally a relational database. It should be organized to hold data in a structure that best supports not only query and documenting but also advanced analysis techniques, such as data mining.

3) Reporting – The data in the data warehouse must be available to the organization's staff if the data warehouse is to be useful. There is a huge number of software applications that execute this function, or reporting can be custom-developed. Reporting tools includes are as follows:

    1. Business intelligence tools – These are software applications that clarify the process of development and production of business documents based on data warehouse information.

    2. Executive information systems (known more widely as Dashboard (business) – These are software applications that are used to display complex business metrics and information graphically to allow rapid understanding.

    3. Data Mining – Data mining tools are software that enables users to implement detailed numerical and statistical calculations on detailed data warehouse data to detect trends, identify design and analyze data.

Building a Data Warehouse –

Some steps that are needed for building any data warehouse are as following below:

1) To extract the data (transnational) from different data sources:
For building a data warehouse, a data is extracted from various data sources and that data is stored in central storage area. For extraction of the data Microsoft has come up with an excellent tool. When you purchase Microsoft SQL Server, then this tool will be available at free of cost.

2) To transform the transnational data:
   There are various DBMS where many of the companies stores their data. Some of them are: MS Access, MS SQL Server, Oracle, Sybase etc. Also, these companies save the data in spreadsheets, flat files, mail systems etc. Relating a data from all these sources is done while building a data warehouse.
3) To load the data (transformed) into the dimensional database:
   After building a dimensional model, the data is loaded in the dimensional database. This process combines the several columns together or it may split one field into the several columns. There are two stages at which transformation of the data can be performed and they are: while loading the data into the dimensional model or while data extraction from their origins.
4) To purchase a front-end reporting tool:
   There are top notch analytical tools are available in the market. These tools are provided by the several major vendors. A cost-effective tool and Data Analyzer is released by the Microsoft on its own.

Database System	Data Warehouse
It supports operational processes.	It supports analysis and performance reporting.
Capture and maintain the data.	Explore the data.
Current data.	Multiple years of history.
Data is balanced within the scope of this one system.	Data must be integrated and balanced from multiple system.
Data is updated when transaction occurs.	Data is updated on scheduled processes.
Data verification occurs when entry is done.	Data verification occurs after the fact.
100 MB to GB.	100 GB to TB.

Database System	Data Warehouse
ER based.	Star/Snowflake.
Application oriented.	Subject oriented.
Primitive and highly detailed.	Summarized and consolidated.
Flat relational.	Multidimensional.

Data Warehouse Architecture

Data Warehouse Architecture is complex as it's an information system that contains historical and commutative data from multiple sources. There are 3 approaches for constructing Data Warehouse layers: Single Tier, two tier and three tier. This 3-tier architecture of Data Warehouse is explained as below.

1) **Single-tier architecture: -**

The objective of a single layer is to minimize the amount of data stored. This goal is to remove data redundancy. This architecture is not frequently used in practice.

2) **Two-tier architecture: -**

Two-layer architecture is one of the Data Warehouse layers which separates physically available sources and data warehouse. This architecture is not expandable and also not supporting a large number of end-users. It also has connectivity problems because of network limitations.

3) **Three-Tier Data Warehouse Architecture**

This is the most widely used Architecture of Data Warehouse.

It consists of the Top, Middle and Bottom Tier.

I. **Bottom Tier:** The database of the Datawarehouse servers as the bottom tier. It is usually a relational database system. Data is cleansed, transformed, and loaded into this layer using back-end tools.
II. **Middle Tier:** The middle tier in Data warehouse is an OLAP server which is implemented using either ROLAP or MOLAP model. For a user, this application tier presents an abstracted view of the database. This layer also acts as a middleware between the end-user and the database.

III. **Top-Tier:** The top tier is a front-end client layer. Top tier is the tools and API that you connect and get data out from the data warehouse. It could be Query tools, reporting tools, managed query tools, Analysis tools and Data mining tools.

Warehouse schema design

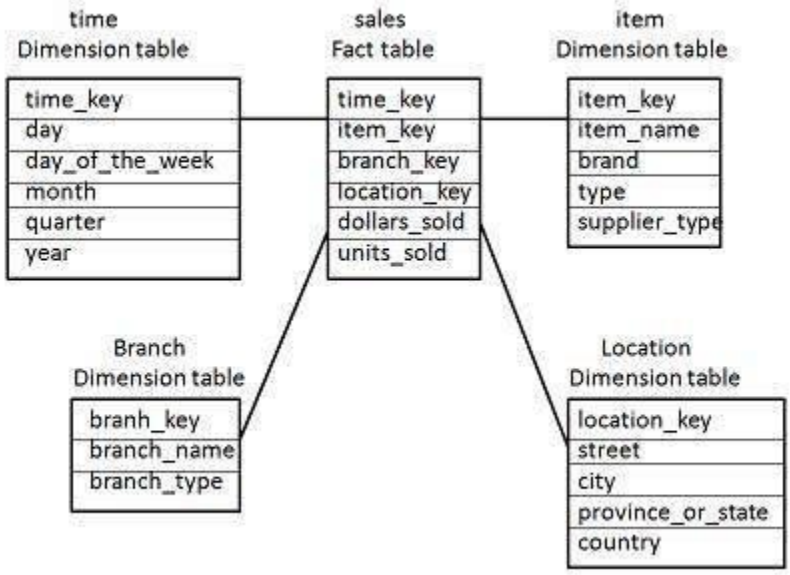
Schema is a logical description of the entire database. It includes the name and description of records of all record types including all associated data-items and aggregates. Much like a database, a data warehouse also requires to maintain a schema. A database uses relational model, while a data warehouse uses Star, Snowflake, and Fact Constellation schema. In this chapter, we will discuss the schemas used in a data warehouse.

1) Star Schema

Each dimension in a star schema is represented with only one-dimension table.

This dimension table contains the set of attributes.

The following diagram shows the sales data of a company with respect to the four dimensions, namely time, item, branch, and location.

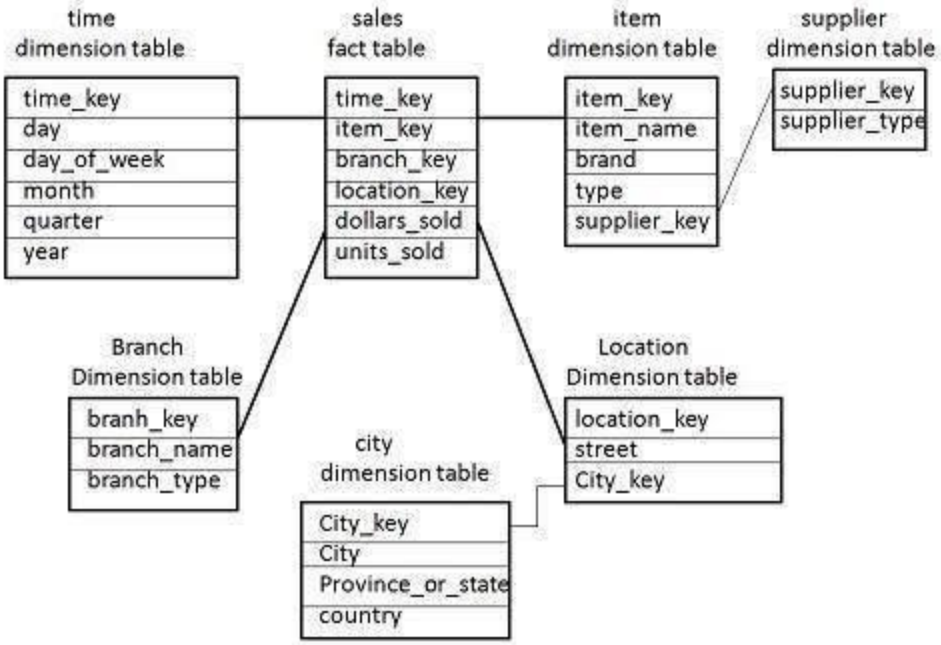


There is a fact table at the center. It contains the keys to each of four dimensions.

The fact table also contains the attributes, namely dollars sold and units sold.

2) Snowflake Schema

I. Some dimension tables in the Snowflake schema are normalized.
II. The normalization splits up the data into additional tables.
III. Unlike Star schema, the dimensions table in a snowflake schema are normalized. For example, the item dimension table in star schema is normalized and split into two-dimension tables, namely item and supplier table.



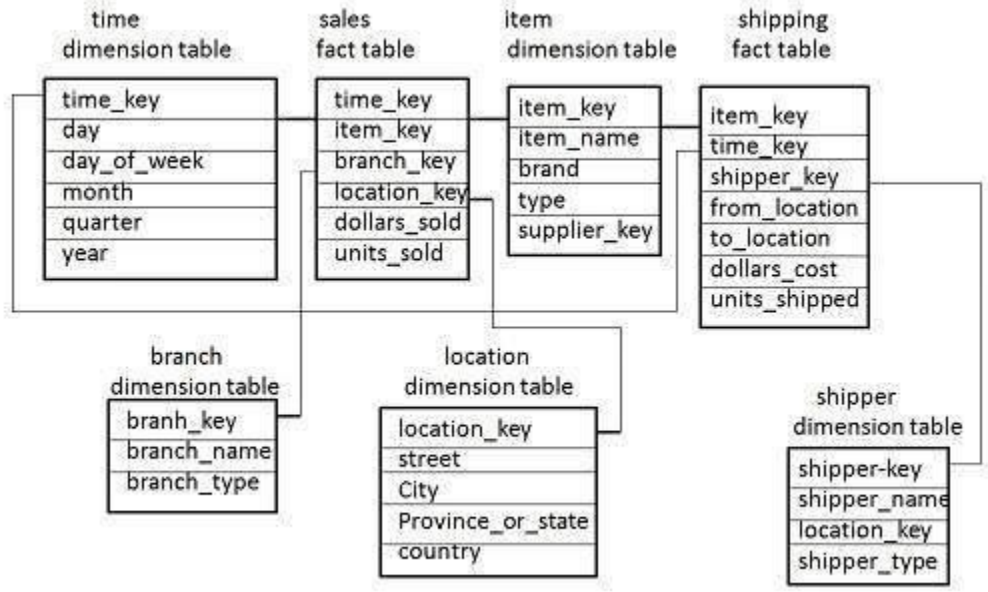
Now the item dimension table contains the attributes item key, item name, type, brand, and supplier-key.

The supplier key is linked to the supplier dimension table. The supplier dimension table contains the attributes supplier key and supplier type.

3) Fact Constellation Schema

A fact constellation has multiple fact tables. It is also known as galaxy schema.

The following diagram shows two fact tables, namely sales and shipping.



The sales fact table is same as that in the star schema.

The shipping fact table has the five dimensions, namely item key, time key, shipper key, from location, to location.

The shipping fact table also contains two measures, namely dollars sold and units sold.

It is also possible to share dimension tables between fact tables. For example, time, item, and location dimension tables are shared between the sales and shipping fact table.

Data Warehouse Design

Datawarehouse is lake of data from multiple sources and integrated for online business analytical processing (OLAP), it needs to done all requirement within the business stages, thus data warehouse design is complex, lengthy and with error process. Thus, as per business data warehouse function's is change over a time which result change in the requirement of system, Therefore, data warehouse and OLAP systems are dynamic, and process of design data ware house is continuous.

Data warehouse design has different way from view materialization in industries. Data warehouse seems as database system with special needs such as answering management related queries. Target of this design is to how to record from multiple data source should be extracted, transformed, and loaded (ETL) to organize in database of Datawarehouse.

There are two approaches

1. "Top-down" approach
2. "Bottom-up" approach

"Bill Inmon is the father of data warehousing".

Top-down Design Approach

1. By Bill Inmon
   First the cube or data warehouse is designed Then the relational database (data mart) is constructed using star schema The cube is deployed
3. ETL (Extraction, Transformation, Loading)
4. Data is extracted from different data sources
5. Data warehouse is built with ETL
6. summarization performed on Aggregation and the data.
7. Then ETL performed to load data into data marts

Advantages of top-down design

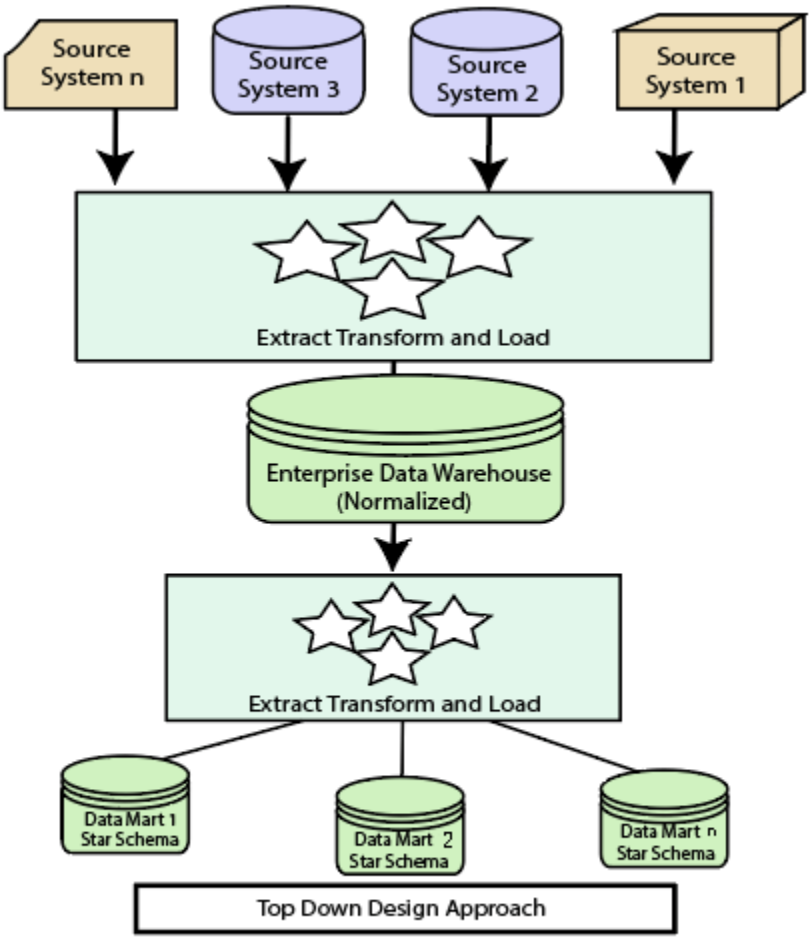
Data Marts are loaded from the data warehouses.

Developing new data mart from the data warehouse is very easy.

Disadvantages of top-down design

This technique is inflexible to changing departmental needs.

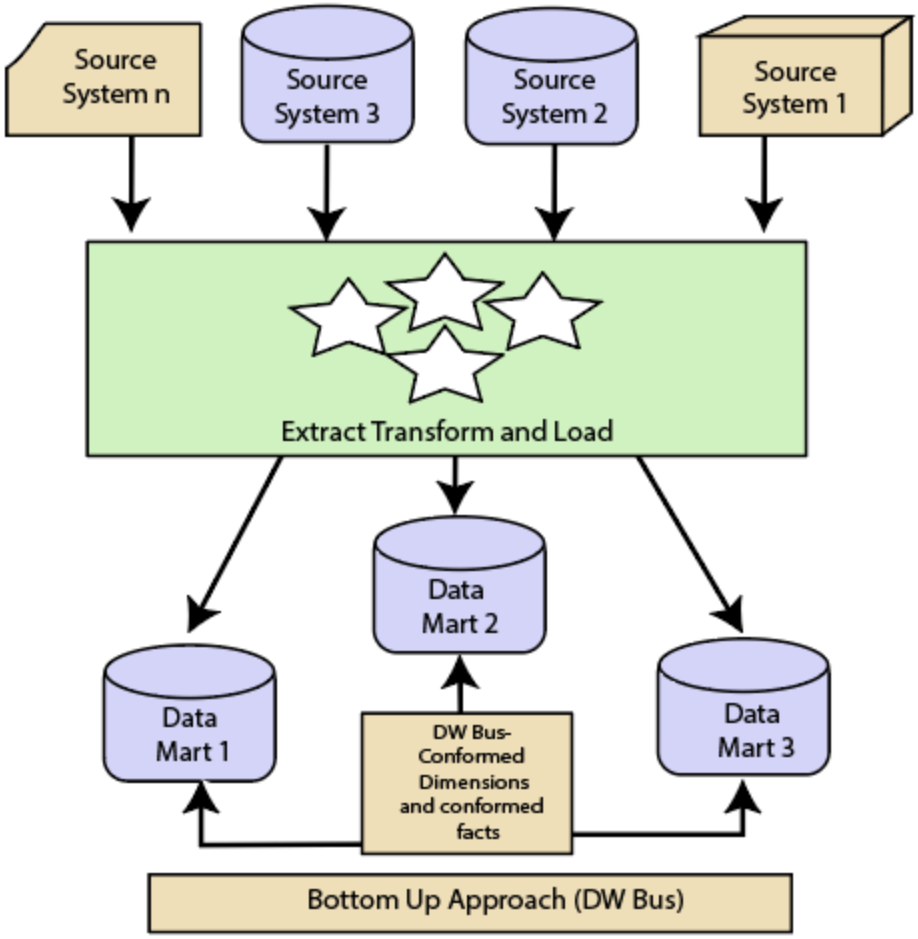
The cost of implementing the project is high.



Top Down Design Approach

Bottom-Up Design Approach

1. This approach is given by Ralph Kimball
2. Also called Kimball methodology
3. REVERESE of top-down
4. The databases/ data marts are designed first Star Schema
5. The cube is designed
6. Finally, the data is loaded into data warehouse
7. Data marts are directly loaded,
8. the data is collected from various data sources
9. Through ETL this data from data mart is loaded into data warehouse
10. Data from data marts is aggregated and summarized. Then loaded in data warehouse



Bottom Up Design Approach

Advantages of bottom-up design

Documents can be generated quickly.

The data warehouse can be extended to accommodate new business units.

It is just developing new data marts and then integrating with other data marts.

Disadvantages of bottom-up design

the locations of the data warehouse and the data marts are reversed in the bottom-up approach design.

Differentiate between Top-Down Design Approach and Bottom-Up Design Approach

Top-Down Design Approach	Bottom-Up Design Approach
Breaks the vast problem into smaller subproblems.	Solves the essential low-level problem and integrates them into a higher one.

Inherently architected- not a union of several data marts.	Inherently incremental; can schedule essential data marts first.
Single, central storage of information about the content.	Departmental information stored.
Centralized rules and control.	Departmental rules and control.
It includes redundant information.	Redundancy can be removed.
It may see quick results if implemented with repetitions.	Less risk of failure, favorable return on investment, and proof of techniques.

Extraction

- Extraction is the operation of extracting information from a source system for further use in a data warehouse environment. This is the first stage of the ETL process.
- Extraction process is often one of the most time-consuming tasks in the ETL.
- The source systems might be complicated and poorly documented, and thus determining which data needs to be extracted can be difficult.
- The data has to be extracted several times in a periodic manner to supply all changed data to the warehouse and keep it up-to-date.

Cleanup

The cleansing stage is crucial in a data warehouse technique because it is supposed to improve data quality. The primary data cleansing features found in ETL tools are rectification and homogenization. They use specific dictionaries to rectify typing mistakes and to recognize synonyms, as well as rule-based cleansing to enforce domain-specific rules and defines appropriate associations between values.

The following examples show the essential of data cleaning:

If an enterprise wishes to contact its users or its suppliers, a complete, accurate and up-to-date list of contact addresses, email addresses and telephone numbers must be available.

Transformation

Transformation is the core of the reconciliation phase. It converts records from its operational source format into a particular data warehouse format. If we implement a three-layer architecture, this phase outputs our reconciled data layer.

Cleansing and **Transformation** processes are often closely linked in ETL tools.

Selecting an ETL Tool

1) Selection of an appropriate ETL Tools is an important decision that has to be made in choosing the importance of an ODS or data warehousing application.
2) The ETL tools are required to provide coordinated access to multiple data sources so that relevant data may be extracted from them. An ETL tool would generally contains tools for data cleansing, re-organization, transformations, aggregation, calculation and automatic loading of information into the object database.
3) An ETL tool should provide a simple user interface that allows data cleansing and data transformation rules to be specified using a point-and-click approach.
4) When all mappings and transformations have been defined, the ETL tool should automatically generate the data extract/transformation/load programs, which typically run-in batch mode.

Multi-Dimensional Data Model, Data Cubes, Stars, Snow Flakes, Fact Constellations

Multi-Dimensional Data Model

A multidimensional model views data in the form of a data-cube. A data cube enables data to be modeled and viewed in multiple dimensions. It is defined by dimensions and facts.

The dimensions are the perspectives or entities concerning which an organization keeps records

. For example, a shop may create a sales data warehouse to keep records of the store's sales for the dimension time, item, and location. These dimensions allow the save to keep track of things, for example, monthly sales of items and the locations at which the items were sold. Each dimension has a table related to it, called a dimensional table, which describes the dimension further. For example, a dimensional table for an item may contain the attributes item_name, brand, and type.

A multidimensional data model is organized around a central theme, for example, sales. This theme is represented by a fact table. Facts are numerical measures. The fact table contains the names of the facts or measures of the related dimensional tables.

Data Cubes

In computer programming contexts, a data cube (or data cube) is a multi-dimensional ("n-D") array of values. Typically, the term data cube is applied in contexts where these arrays are massively larger than the hosting computer's main memory; examples include multi-terabyte/petabyte data warehouses and time series of image data.

Stars

Star schema is the fundamental schema among the data mart schema and it is simplest. This schema is widely used to develop or build a data warehouse and dimensional data marts. It includes one or more fact tables indexing any number of dimensional tables. The star schema is a necessary cause of the snowflake schema. It is also efficient for handling basic queries.

It is said to be star as its physical model resembles to the star shape having a fact table at its center and the dimension tables at its peripheral representing the star's points.

Advantages of Star Schema:

Simpler Queries

Join logic of star schema is quite cinch in comparison to other join logic which are needed to fetch data from a transactional schema that is highly normalized.

Simplified Business Reporting Logic

In comparison to a transactional schema that is highly normalized, the star schema makes simpler common business reporting logic, such as as-of reporting and period-over-period.

Feeding Cubes

Star schema is widely used by all OLAP systems to design OLAP cubes efficiently. In fact, major OLAP systems deliver a ROLAP mode of operation which can use a star schema as a source without designing a cube structure.

Disadvantages of Star Schema:

- Data integrity is not enforced well since in a highly de-normalized schema state.
- Not flexible in terms if analytical needs as a normalized data model.
- Star schemas don't reinforce many-to-many relationships within business entities at least not frequently.

Snow Flakes

Snowflake Schema in data warehouse is a logical arrangement of tables in a multidimensional database such that the ER diagram resembles a snowflake shape. A Snowflake Schema is an extension of a Star Schema, and it adds additional dimensions. The dimension tables are normalized which splits data into additional tables.

Characteristics of Snowflake:

- The main benefit of the snowflake schema it uses smaller disk space.

- Easier to implement a dimension is added to the Schema
- Due to multiple tables query performance is reduced
- The primary challenge that you will face while using the snowflake Schema is that you need to perform more maintenance efforts because of the more lookup tables.

Fact Constellations

Fact constellation is a measure of online analytical processing, which is a collection of multiple fact tables sharing dimension tables, viewed as a collection of stars. It can be seen as an extension of the star schema.

A fact constellation schema has multiple fact tables. It is also known as galaxy schema. It is widely used schema and more complex than star schema and snowflake schema. It is possible to create fact constellation schema by splitting original star schema into more star schema. It has many fact tables and some common dimension table.

Advantage:

Provides a flexible schema.

Disadvantage:

It is much more complex and hence, hard to implement and maintain.

What is OLAP (Online Analytical Processing)?

- **OLAP** stands for **On-Line Analytical Processing**.
- OLAP is a technology that organize large business database and support complex analysis
- This data has been transformed from raw information to reflect the real dimensionality of the enterprise as understood by the clients.
- **OLAP** implement the multidimensional analysis of business information and support the capability for complex estimations, trend analysis, and sophisticated data modeling. It is rapidly enhancing the essential foundation for Intelligent Solutions containing Business Performance Management, Planning, Budgeting, Forecasting, Financial Documenting, Analysis, Simulation-Models, Knowledge Discovery, and Data Warehouses Reporting.
- OLAP enables end-clients to perform ad hoc analysis of record in multiple dimensions, providing the insight and understanding they require for better decision making