Business intelligence
Systèmes décisionnels
Informatique décisionnelle

INTRODUCTION

The business environment (climate) is constantly changing, and it is becoming more and more complex. Organizations, private and public, are under pressures that force them to respond quickly to changes. Any business organization needs to continually monitor its business environment and its own performance, and then rapidly adjust its plans by closing the gap between the current performance of an organization and its desired performance, as expressed in its mission, objectives, and goals, and the strategy to achieve them. This includes monitoring the industry, the competitors, the suppliers, and the customers. Businesses use many techniques for understanding their environment and predicting the future for their own benefit and growth.

For years, managers considered decision making purely an art-a talent acquired over a long period through experience (i.e., learning by trial-and-error) and by using intuition. Data-based decisions are more effective than those based on feelings alone. Actions based on accurate data, information, knowledge, experimentation, and testing, using fresh insights, can more likely succeed and lead to sustained growth.

Managers usually make decisions by following a four-step process

1. Define the problem (i.e., a decision situation that may deal with some difficulty or with an opportunity).
2. Construct a model that describes the real-world problem.
3. Identify possible solutions to the modeled problem and evaluate the solutions.
4. Compare, choose, and recommend a potential solution to the problem.

To follow this process, one must make sure that sufficient alternative solutions are being considered, that the consequences of using these alternatives can be reasonably predicted, and that comparisons are done properly.

INFORMATION SYSTEMS SUPPORT FOR DECISION MAKING

Computer applications have moved from transaction processing and monitoring activities to problem analysis and solution applications, and much of the activity is done with Web-based technologies, in many cases accessed through mobile devices.

Analytics and BI tools such as data warehousing, data mining, online analytical processing (OLAF), dashboards, and the use of the Web for decision support are the cornerstones of today's modern management.

DEGREE OF STRUCTUREDNESS

decision-making processes fall along a continuum that ranges from highly structured (sometimes called **programmed**) to highly unstructured (i.e., **nonprogrammed**) decisions.

Structured processes are routine and typically repetitive problems for which standard solution methods exist.

Unstrnctured processes are fuzzy, complex problems for which there are no cut-and-dried solution methods.

An **unstructured problem** is one where the articulation of the problem or the solution approach may be unstructured in itself. In a **structured** problem, the procedures for obtaining the best (or at least a good enough) solution are known.

Semistructured problems fall between structured and unstructured problems, having some structured elements and some unstructured elements.

PHASES OF THE DECISION-MAKING PROCESS

Decision making is a process of choosing among two or more alternative courses of action for the purpose of attaining one or more goals.

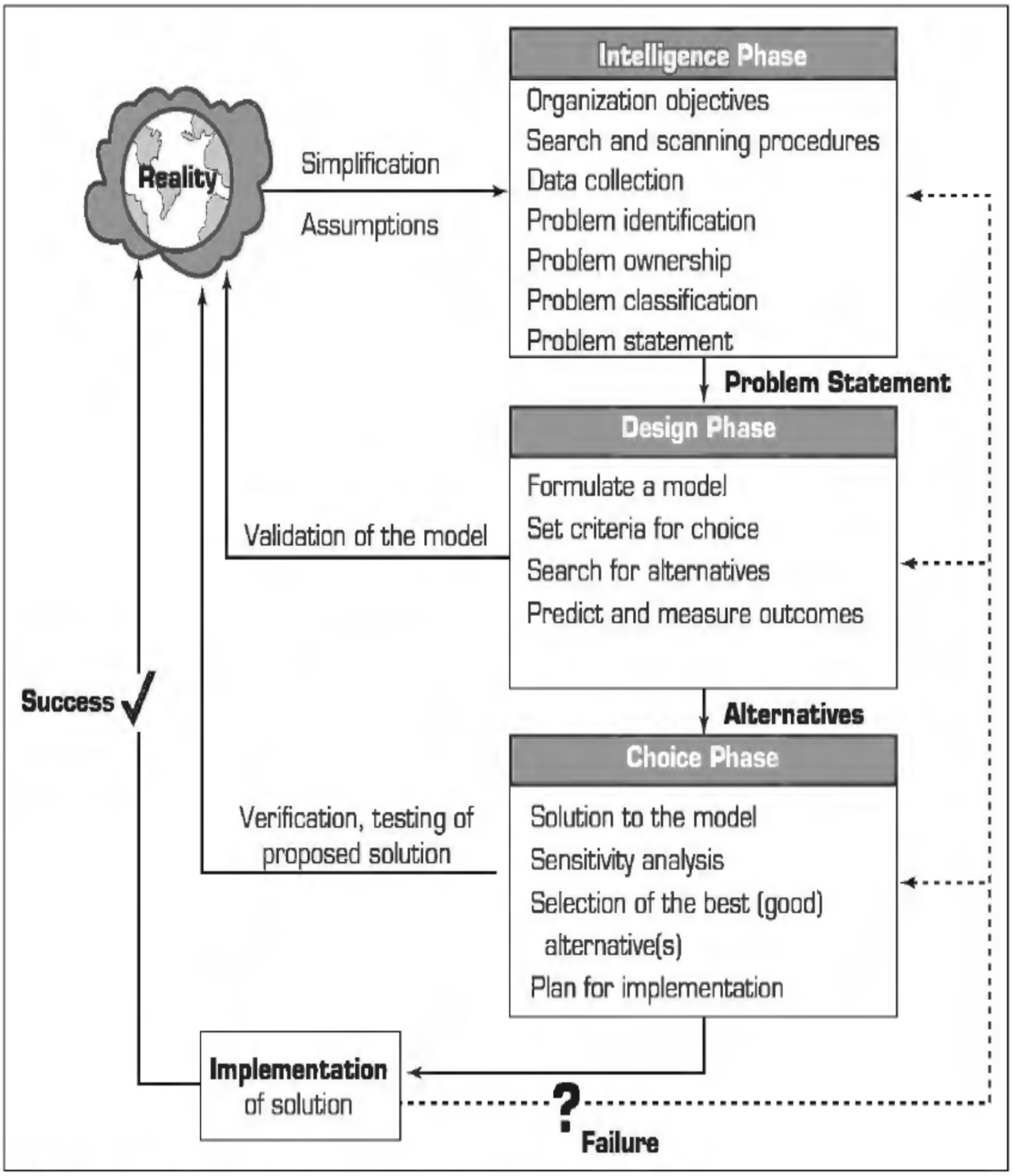


FIGURE 2.1 The Decision-Making/Modeling Process.

Business intelligence

Business intelligence (BI) is a technology-driven process for analyzing data and presenting actionable information to help corporate executives, business managers and other end users make more informed business decisions.

Business intelligence (BI) is a broad category of applications, technologies, and processes for gathering, storing, accessing, and analyzing data to help business users make better decisions.

BI encompasses a variety of tools, applications and methodologies that enable organizations to collect data from internal systems and external sources, prepare it for analysis, develop and run queries against the data, and create reports, dashboards and data visualizations to make the analytical results available to corporate decision makers as well as operational workers.

BI is any activity, tool, or process used to obtain the best information to support the process of making decisions.

Business intelligence is essentially timely, accurate, high-value, and actionable business insights, and the work processes and technologies used to obtain them.

The four most common components of a business intelligence system are ETL tools, data warehouses, OLAP techniques, and data-mining.

A Brief History of Bl

The term BI was coined by the Gartner Group in the mid-1990s. However, the concept is much older; it has its roots in the MIS reporting systems of the 1970s. During that period, reporting systems were static, two dimensional, and had no analytical capabilities. In the early 1980s, the concept of executive information systems (EIS) emerged. This concept expanded the computerized support to top-level managers and executives. Some of the capabilities introduced were dynamic multidimensional (ad hoc or on-demand) reporting, forecasting and prediction, trend analysis, drill-down to details, status access, and critical success factors. These features appeared in dozens of commercial products until the mid-1990s. Then the same capabilities and some new ones appeared under the name BI.

Today, a good BI-based enterprise information system contains all the information executives need. So, the original concept of EIS was transformed into BI. By 2005, BI systems started to include artificial intelligence capabilities as well as powerful analytical capabilities.

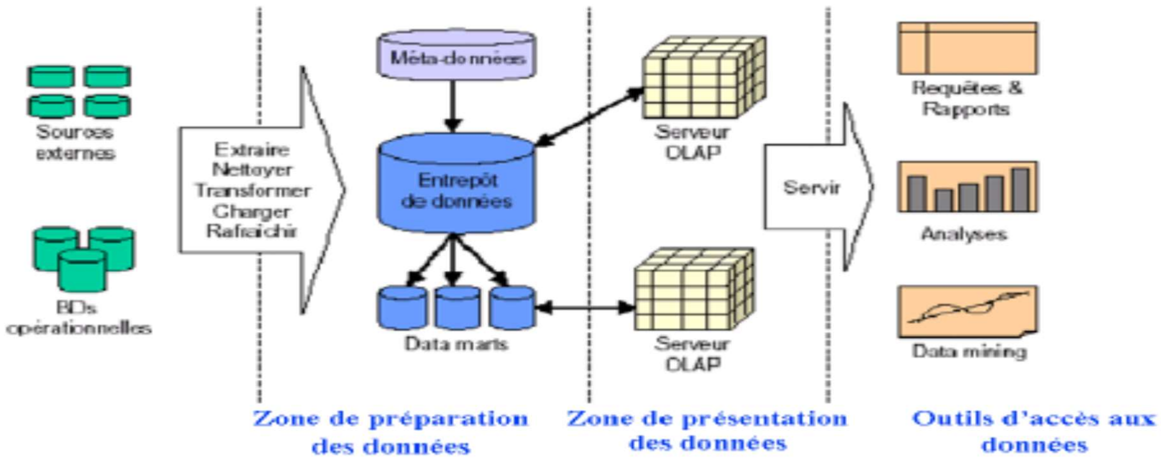
Enterprise resource planning (ERP) is business management software—usually a suite of integrated applications—that a company can use to store and manage data from every stage of business, including Product planning, cost and development, Manufacturing, Marketing and sales, Inventory management, Shipping and payment

Supply chain management (SCM) is the handling of the entire production flow of a good or service — starting from the raw components all the way to delivering the final product to the consumer. A company creates a network of suppliers ("links" in the chain) that move the product along from the suppliers of raw materials to those organizations that deal directly with users.

Customer relationship management (CRM) is a set of integrated, data-driven software solutions that help manage, track, and store information related to your company's current and potential customers. By keeping this information in a centralized system, business teams have access to the insights they need, the moment they need them.

Three-Tier Data Warehouse Architecture

Data Warehouses usually have a three-level (tier) architecture that includes:



Traditional data warehouse architecture employs a three-tier structure composed of the following tiers.

- Data Tier: The Data Tier mainly consists of the Data Sources, ETL Tool, and Data Warehouse.
- Middle tier (Application Tier): The middle tier houses an OLAP server, which transforms the data into a structure better suited for analysis and complex querying. The OLAP server can work in two ways: either as an extended

relational database management system that maps the operations on multidimensional data to standard relational operations (Relational OLAP), or using a multidimensional OLAP model that directly implements the multidimensional data and operations.

- Presentation Tier: Presentation tier is the client layer. This tier holds the tools used for high-level data analysis, querying, reporting and mining.

Accessing Data Warehouses

Analysis is the last level common to all data warehouse architecture types where end users query data warehouses: *reports, Data visualization, OLAP, and dashboards*. End users often use the information stored to a data warehouse as a starting point for additional business intelligence applications, such as what-if analyses and data mining.

1. Reporting and Querying

- This approach is oriented to those users who need to have regular access to the information in an almost static way.
- A report is defined by a query and a layout. A layout can look like a table or a chart (diagrams, histograms, pies, and so on).
- The most common features in modern querying and reporting tools are
    - SQL-free query building
    - Drag-and-drop creation of queries and reports
    - Basic data-analysis features (such as creating a calculation on a report)
    - Graphical features that allow users to easily build charts and graphs
    - Easy publication of information — e-mail, Internet, intranet, wherever
    - colleagues might need to see it

- Reporting tools includes: SAP Business Objects Web Intelligence (WebI), Crystal Reports, SAP Lumira, Dashboard Designer, IBM Cognos, Microsoft Power BI, Tableau.

2. OLAP

- OLAP might be the main way to exploit information in a data warehouse.
- It gives end users the opportunity to analyze and explore data interactively on the basis of the multidimensional model.
- The most common operators provided by most OLAP clients are roll-up, drill-down, slice-and-dice, pivot, drill-across, and drill-through

- OLAP attempts to analyze complex data in real time on a database that is constantly updated with transactional data. The OLAP optimizes the searching of huge data files by means of automatic generation of SQL queries (Olszak & Ziemba, 2006).
- OLAP offers techniques for data analysis and drilling data and the tools are mainly used for interactive report generations.
- OLAP is an improvement to earlier single dimensional analysis tools that allowed managers to analyze data from only one perspective at a time. By providing managers with a multi-dimensional tool, OLAP enables managers to analyze data from multiple perspectives and explore it in order to discover hidden information (Matei, 2010).
- OLAP tools includes: IBM Cognos, Micro Strategy, Palo OLAP Server, Apache Kylin, icCube, Pentaho BI, Mondrian, Excel.

3. Data visualization and Dashboards

- **Data visualization** is the representation of information and data using charts, graphs, maps, and other visual tools. These visual displays of information communicate complex data relationships and data-driven insights in a way that is easy to understand.

  Interactive data visualization is the use of tools and processes to produce a visual representation of data which can be explored and analyzed directly within the visualization itself. This interaction can help uncover insights which lead to better, data-driven decisions.

- **Dashboards**: A business dashboard is an information management tool that visually tracks, analyzes and displays key performance indicators (KPI), metrics and key data points to monitor the health of a business, department or specific process.
  - Dashboards are often used to track company performance in real-time, enabling companies to immediately see possible issues or opportunities. They may be used to monitor a number of variables, like as sales, revenue, customer happiness, and staff performance.
  - Data is visualized on a dashboard as tables, line charts, bar charts and gauges so that users can track the health of their business against benchmarks and goals.
  - Data visualization tools includes:Tableau, Microsoft Power BI,Qlik Sense,Looker,Demo,IBM Cognos,

Key Performance Indicators

- KPIs specifically help determine a company's strategic, financial, and operational achievements, especially compared to those of other businesses within the same sector.

- Key performance indicators (KPIs) measure a company's success vs. a set of targets, objectives, or industry peers.
- KPIs can be financial, including net profit (or the bottom line, net income), revenues minus certain expenses, or the current ratio (liquidity and cash availability).
- Most KPIs fall into four different categorie Strategic, operational or Functional(on specific departments or functions)

- Examples of KPI and Merics
    - Financial Metrics and KPIs : Profitability ratios (i.e., net profit margin), Solvency ratios (i.e., total debt-to-total-assets ratio), Turnover ratios (i.e., inventory turnover)
  - Customer Experience Metrics and KPI :Number of new ticket requests, Number of resolved tickets , Average resolution time, Average response time , Customer satisfaction rating
  - Process Performance Metrics and KPI: Total cycle time , Error rate, Quality rate
  - Marketing KPIs: Website traffic ,Social media traffic (This KPI tracks the views, follows, likes, retweets, shares, engagement, and other measurable interactions between customers and the company's social media profiles.

4. Data Mining

- Data mining is a process of discovering patterns in large data sets that go beyond simple analysis.
- Data Mining is a combination of discovering techniques + prediction techniques whereas, OLAP – Provides with a very good view of what is happening, but cannot predict what will happen in the future or why it is happening.
- Data mining techniques include: Clustering, Classification, time series analysis, market basket analysis, regression, etc.
- Data mining tools includes SAS Enterprise Miner, Oracle Data Miner, IBM SPSS Modeler, Tibco Data Science. Apache Mahout, Weka, …etc.

Data Warehouse

In simple terms, a data warehouse (DW) is a pool of data produced to support decision making; it is also a repository of current and historical data of potential interest to managers throughout the organization. Data are usually structured to be available in a form ready for analytical processing activities (i.e., online analytical processing [OLAP], data mining, querying, reporting, and other decision support applications. A data warehouse is a subject-oriented, integrated, time-variant, nonvolatile collection of data in support of management's decision-making process.

- Subject oriented: A data warehouse is organized around the key subjects (or High level entities) of the enterprise. Major subjects may include customers, patients, students and products. Operational databases hinge on many different enterprise-specific applications.

- Integrated: The data housed in the data warehouse is defined using consistent naming conventions, formats, encoding structures, and related characteristics.
- Time-variant: Data in the data warehouse contains a time dimension so that it may be used as a historical record of the business.
- Non-volatile: Data in the data warehouse is loaded and refreshed from operational systems, but cannot be updated by end-users."

Data warehouse features

- A data warehouse is a collection of relevant business data that is organized and validated (Cody et al. 2002) so that it can be analyzed to support business decision making.
- Data warehouses are populated with data that has been extracted from distributed databases, often heterogeneous and, in some cases, external to the organization which is using it.
- Data warehouses are subject-oriented databases that are integrated into an information system.
- They are time relevant, meaning that they are snapshots of a point of time within the information system and they are not updatable so as to maintain the integrity of the historical point in which the snapshot of data is taken.
- Data warehouses are offline, meaning that they reside on a different system than that of the data of which they are storing a snapshot.
- The data warehouse is considered the core component of a business intelligence system (Negash, 2004). This collection of data is used to support the management decision-making process (Hevner & March, 2005).
- Hevner and March (2005) conclude that the key role of a data warehouse is to provide an understanding of business problems, opportunities, and performance based on compelling business intelligence facilitating decision making.

- The primary goal of data warehouse systems should always be to properly define a process to transform data into information.
- A Data warehouse is typically used to connect and analyze business data from heterogeneous sources. The data warehouse is the core of the BI system which is built for data analysis and reporting.
- The decision support database (Data Warehouse) is maintained separately from the organization's operational database.
- Data Warehouse (system) is an information system which provides users with current and historical decision support information which is difficult to access or present in the traditional operational data store.
- Data Warehousing (DW) is the process for collecting and managing data from varied sources to provide meaningful business insights.

DataMart [7]

A data mart is a subset of a data warehouse oriented to a specific business line. Data marts are small in size and are more flexible compared to a Data warehouse.

- Data Mart allows faster access of Data due to reduction in volume of data.
- Data mart are simpler and less costly to implement when compared to corporate Data warehouse.
- Data is partitioned and allows very granular access control privileges.
- Data can be segmented and stored on different hardware/software platforms.
- Data Mart cannot provide company-wide data analysis as their data set is limited.

Types of Data Mart

There are three main types of data mart:

- Independent: Independent data marts are created by drawing data directly from operational, external or both sources.
- Dependent: Dependent data mart is created from data warehouse.
- Hybrid: This type of data marts can take data from data warehouses or operational systems.

Benefits of a Separating data warehouse from transactional databases

- Separating processing and analysis of data warehouse from transactional databases, improves the performance of both systems.
- Decision support requires integration and consolidation (aggregation, summarization) of data from different sources.
- Bringing greater quality, consistency and accuracy to the data handled by a company, resulting in better decision making by its management team.
- Decision support requires historical data which operational DBs do not typically maintain.

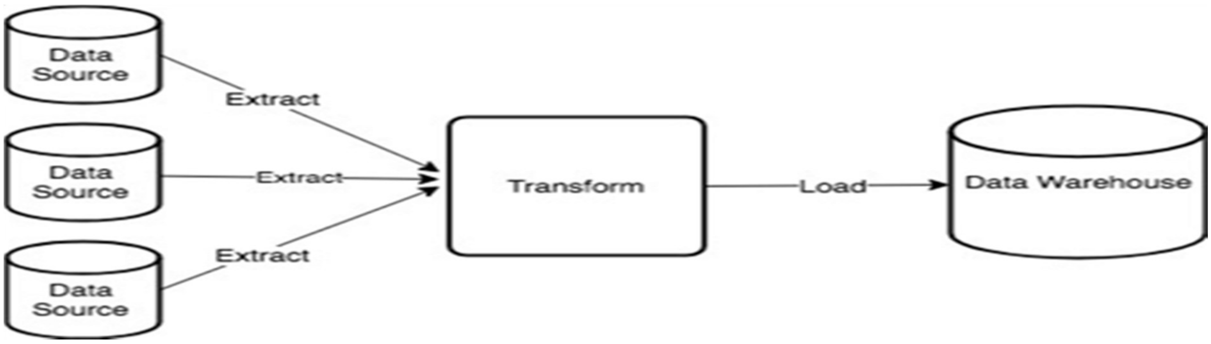
ETL (Extract, Transform, and Load) Process [5]

ETL solutions are divided into three distinct stages

1. The *extraction* stage: This stage involves obtaining access to data originating from different, often heterogeneous sources. These sources are often distributed across multiple platforms and can be part of a customer's information system (Schink, 2009).

2. The *transformation* stage: This stage transforms the extracted data and is considered the most complex stage of the ETL process. The transformation stage converts the data into the same schema of the data warehouse to which it is to be loaded. The transformation phase is usually performed by means of traditional programming languages, script languages or the SQL language (Olszak & Ziemba, 2006).

3. The *load* stage: The load stage *pushes* the transformed data and loads the data warehouses with data that are aggregated and filtered (Olszak & Ziemba, 2007).



- ETL (Extraction, Transformation and Loading) is a data integration method where data from one or more source systems is first read (i.e. extracted), then made to go through some changes (data cleaning and transformation into a proper storage format/structure) and then the changed data is written to a target system (i.e. loaded) into the final target database such as an operational data store, a data mart, Data Lake or a data warehouse.

- Since the data extraction takes time, it is common to execute the three phases in pipeline. While the data is being extracted, another transformation process executes while processing the data already received and prepares it for loading while the data loading begins without waiting for the completion of the previous phases.
- ETL takes place once when a data warehouse is populated for the first time, then incremental extraction, used to update data warehouses regularly.
- The requirement of a business intelligence system to be able to extract data in different formats from disperse sources, transform them into like formats, and then load them into the appropriate data warehouse has traditionally made the ETL process the most expensive aspect of a business intelligence system (Hevner & March, 2005).

- In some cases a business intelligence system may have a dedicated but separate data warehouse that acts as a staging area where the ETL process can do low level analysis and transformation in this data warehouse prior to loading it into the enterprise data warehouses (Castellanos et al., 2009).

A typical ETL workflow will [6]

- Periodically connect to one or more operational data sources such as an ERP, CRM (Customer Relationship Management), SCM (supply chain management), Corporate Emails, Reference Websites, Wikis and Forums or Log Files. This is usually a nightly process.
- Extract batches of rows from one or more source system tables based on some criteria. For example, it will read all rows since the last batch of data was loaded.
- Copy the extracted data to a staging area. This can be another RDBMS of the same type or a blob storage.
- Perform transformation on the staged data. These transformations:
    - Are done in-memory or in temporary tables on disk.
    - Can be iterative in nature (for example, aggregating batch of rows per customer transaction).
    - May involve transactions, so any in-process error may abort the whole operation.
    - Usually write the process outputs to log files for debugging.
    - Can involve operations like data cleansing, filtering, converting data types, formatting, enriching, applying lookups and calculations, masking, removing duplicates, sorting and aggregating to name just a few.
- Connect to the target data warehouse and copy the processed data to one or more tables (for example, aggregated summary to measure tables and dimension data to dimension tables).

ETL transformation types

- Cleaning: Mapping NULL to 0 or "Male" to "M" and "Female" to "F," date format consistency, etc.
- Deduplication: Identifying and removing duplicate records
- Format revision: Character set conversion, unit of measurement conversion, date/time conversion, etc.
- Key restructuring: Establishing key relationships across tables Advanced transformations:
- Derivation Applying business rules to your data that derive new calculated values from existing data – for example, creating a revenue metric that subtracts taxes
- Filtering: Selecting only certain rows and/or columns

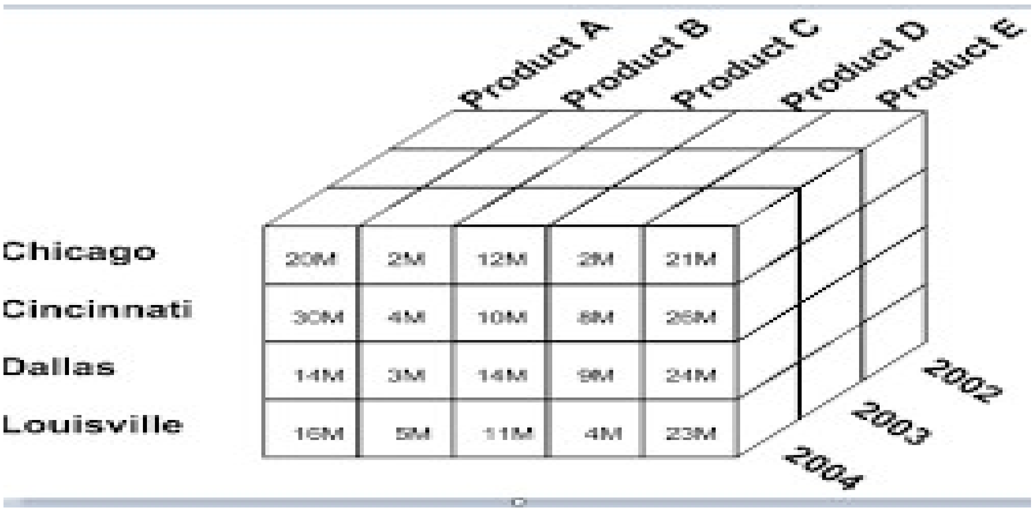
- Joining: Linking data from multiple sources – for example, adding ad spend data across multiple platforms, such as Google Adwords and Facebook Ads
- Splitting: Splitting a single column into multiple columns
- Data validation: Simple or complex data validation – for example, if the first three columns in a row are empty then reject the row from processing
- Summarization: Values are summarized to obtain total figures which are calculated and stored at multiple levels as business metrics – for example, adding up all purchases a customer has made to build a customer lifetime value (CLV) metric
- Aggregation: Data elements are aggregated from multiple data sources and databases
- Integration: Give each unique data element one standard name with one standard definition. Data integration reconciles different data names and values for the same data element.

The multidimensional data model [10]

The multidimensional is an integral part of On-Line Analytical Processing, or OLAP. Because OLAP is on-line, it must provide answers quickly; analysts pose iterative queries during interactive sessions, not in batch jobs that run overnight. And because OLAP is also analytic, the queries are complex. The multidimensional data model is designed to solve complex queries in real time.

The multidimensional data model is important because it enforces simplicity. As Ralph Kimball states in his landmark book, The Data Warehouse Toolkit:

The multidimensional data model is composed of logical cubes, measures, dimensions, hierarchies, levels, and attributes.



Facts [10]

Facts are the measurements/metrics or facts from your business process. For a Sales business process, a measurement would be quarterly sales number.
Measures are organized by dimensions, which typically include a Time dimension.

A critical decision in defining a measure is the lowest level of detail (sometimes called the grain). Users may never view this base level (granularity) data, but it determines the types of analysis that can be performed.
For example, market analysts (unlike order entry personnel) do not need to know that Beth Miller in Ann Arbor, Michigan, placed an order for a size 10 blue polka-dot dress on July 6, 2002, at 2:34 p.m. But they might want to find out which color of dress was most popular in the summer of 2002 in the Midwestern United States.

The base level determines whether analysts can get an answer to this question. For this particular question, Time could be rolled up into months, Customer could be rolled up into regions, and Product could be rolled up into items (such as dresses) with an attribute of color. However, this level of aggregate data could not answer the question: At what time of day are women most likely to place an order? An important decision is the extent to which the data has been pre-aggregated before being loaded into a data warehouse.

Dimensions [7]

Dimensions provides the context surrounding a business process event. In simple terms, they give who, what, where of a fact. In the Sales business process, for the fact quarterly sales number, dimensions would be

- Who – Customer Names
- Where – Location
- What – Product Name

Attributes [7]

The Attributes are the various characteristics of the dimension in multidimensional data modeling.

In the Location dimension, the attributes can be State, Country, Zipcode etc. Attributes are used to search, filter, or classify facts.

Dimension Hierarchies and Levels [10]

A hierarchy is a way to organize data at different levels of aggregation. In viewing data, analysts use dimension hierarchies to recognize trends at one level, drill down to lower levels to identify reasons for these trends, and roll up to higher levels to see what affect these trends have on a larger sector of the business.

Each level represents a position in the hierarchy. Each level above the base (or most detailed) level contains aggregate values for the levels below it. The members at different levels have a one-to-many parent-child relation.
Example: Year-quarter-month-week-day

Types of Schema [10]

The relational implementation of the multidimensional data model is typically a star schema, or a snowflake schema. A star schema is a convention for organizing the data into dimension tables and fact tables,

Dimension Tables

A star schema stores all of the information about a dimension in a single table. Each level of a hierarchy is represented by a column in the dimension table. Attributes are stored in columns of the dimension tables.

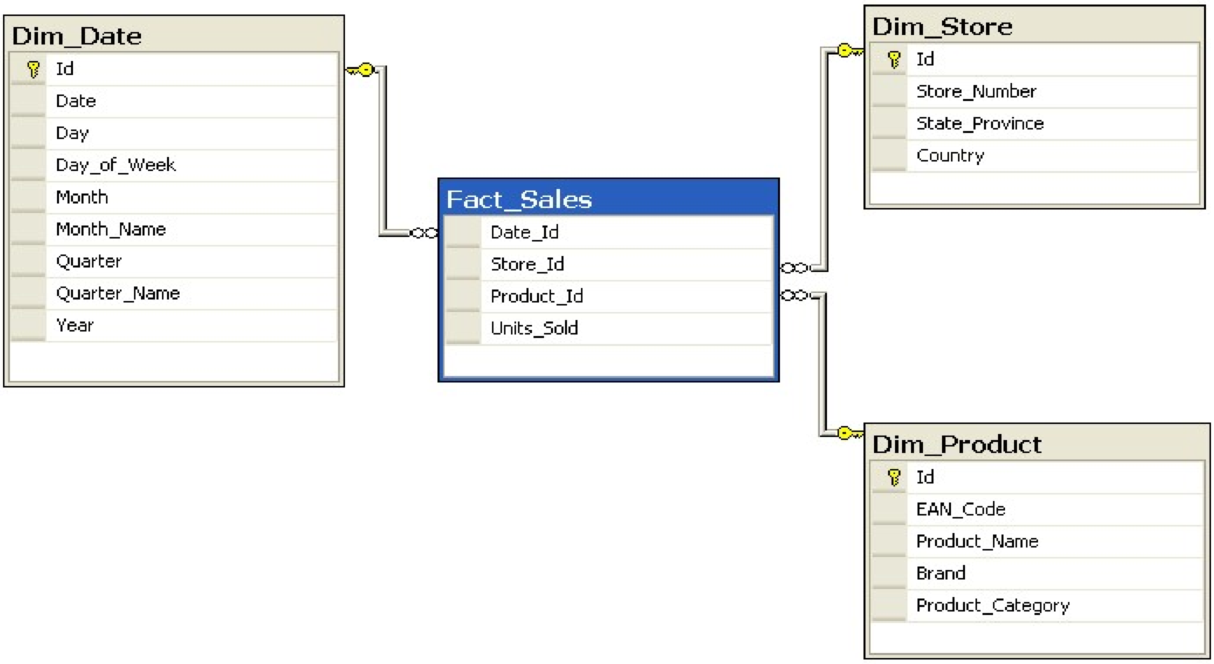
A snowflake schema normalizes the dimension members by storing each level in a separate table.

Fact Tables

Measures are stored in fact tables. Fact tables contain a composite primary key, which is composed of several foreign keys (one for each dimension table) and a column for each measure that uses these dimensions.
A fact table that does not contain any measure is a fact-less fact table. This table will only contain keys from different dimension tables. This is often used to resolve a many-to-many cardinality issue

Star schema

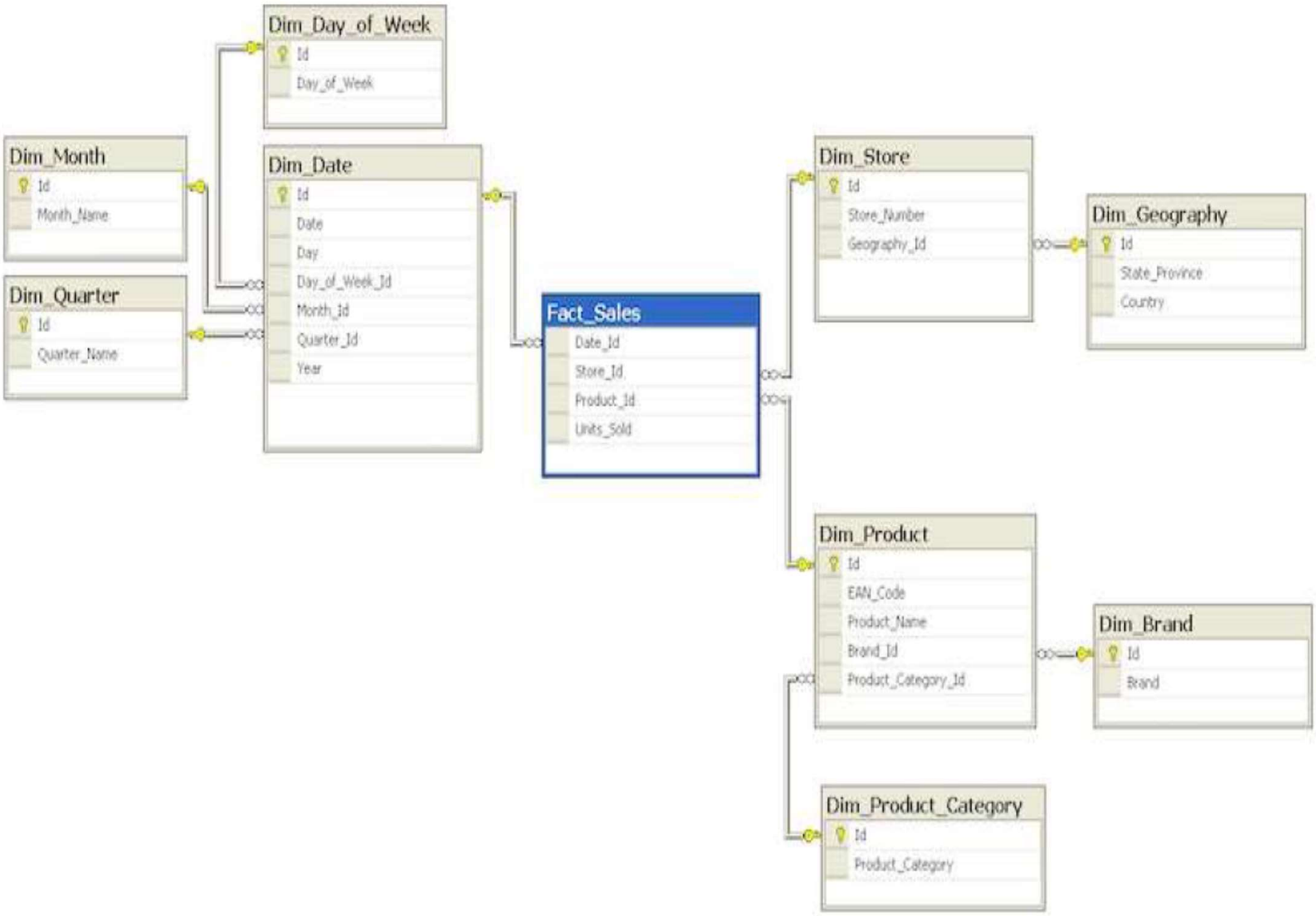
The following diagram illustrates what a simple star schema looks like:



Source [11]

Snow flake schema

The following diagram illustrates what a simple snow flake schema looks like:



Source [11]

Types of Keys:

Primary Key –

Dimension table's every row is identified by a unique value which is generally known as primary key of the table.

Surrogate Key –

These are the keys which are generated by the system and generally does not have any built in meaning. It is UNIQUE since it is consecutively created number for each record being embedded in the table.

It is MEANINGLESS since it doesn't convey any business significance in regards to the record it is connected to in any table. It is SEQUENTIAL since it is doled out in successive request as and when new records are made in the table, beginning with one and going up to the most elevated number that is required.

Foreign Key –

In the fact table the primary key of other dimension table is act as the foreign key.

Alternate key –

It is also a unique value of the table and generally knows as secondary key of the table. Usually it is the business key.

Composite key –

It is the key which consist of two or more attribute.

E/R Model & multidimensional Model characteristics

- E/R modeling is a design technique in which we store the data in highly normalized form inside a relational database.
- A table is in third normal form (3NF) if each non-key column is independent of other non-key columns, and is dependent only on the key.
- The E/R model basically focuses on three things, entities, attributes, and relationships.
- The objective of normalization is to minimize redundancy by not having the same data stored in multiple tables. As a result, normalization can minimize any integrity issues because SQL updates then only need to be applied to a single table. However, queries, particularly those involving very large tables, that include a join of the data stored in multiple normalized tables may require
- Additional effort and programming to achieve acceptable performance.
- Data warehousing applications are mostly read only, and therefore typically can benefit from denormalization. Denormalization is a technique that involves duplicating data in one or more tables to minimize or eliminate time consuming joins.
- A dimensional model is also commonly called a star schema.
- The dimensional model consists of two types of tables having different characteristics. They are: Fact table and Dimension table.

OLAP

Up until now, our focus has been on the technology foundations of a BI solution. First, we identify the important data in your company; that is, the transactional information that resides on ERP, CRM and other operational systems.
Next we gather the data together into a single place, and in a single format; that's the job of the ETL processes, the data warehouse and related components.
Finally, you provide access to that mountain of data in the form of querying and reporting tools.

An OLAP application is software designed to allow users to navigate, retrieve, and present business data. Rather than taking data from a relational system, writing complex queries to retrieve it, then manually inserting it into a report to analyze, OLAP tools cut out the middle steps by actually storing the data in a report-ready format.

With a little drag-and-drop action, we are looking at the data in a completely new way. There is no need to work through confusing query logic or write long SQL strings.

Multidimensional Analysis

An OLAP data-and-reporting environment is different from a traditional database environment. In an OLAP system, users work with data in dimensional form rather than relational form.
In Table 5-1, Although the data is (of course) displayed in two dimensions (horizontal and vertical), in OLAP terms this is considered one dimensional data. In other words, we're looking at sales data from one particular perspective (or dimension): in this case, by region.
Table 5-4 combine two of the dimensions into a table.

Table 5-1 A One-dimensional Table

Annual Sales Data	
Region	*Amount*
Northeast	$45,091
Southeast	$73,792
Central	$88,122
West	$63,054
TOTAL:	$270,059

Table 5-4 Two Dimensions of Data Combined

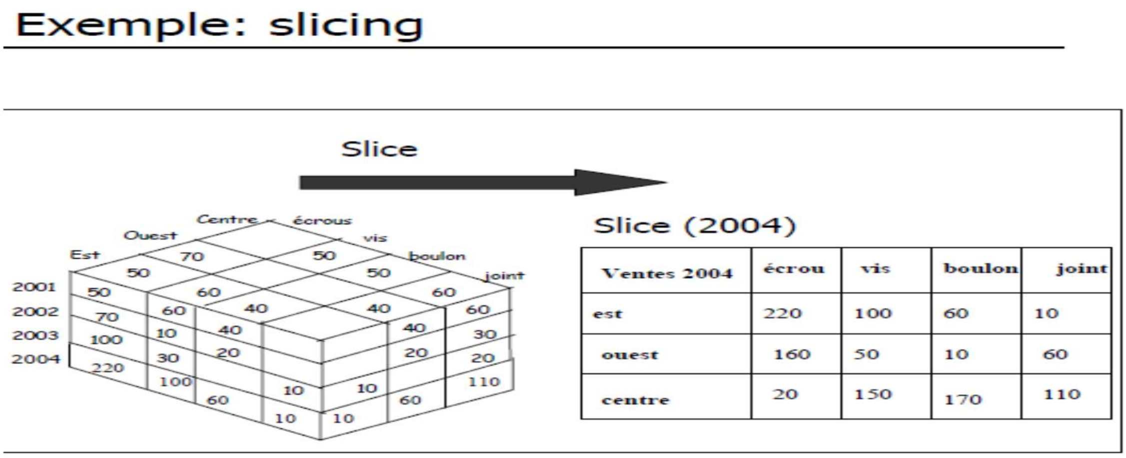
Annual Sales Data			
Product			*Totals*
Quarter	*Gizmo*	*Widget*	
Q1	$23,199	$32,688	$55,887
Q2	$24,798	$62,861	$87,659
Q3	$14,555	$9,043	$23,598
Q4	$26,145	$76,770	$102,915
TOTALS	$88,697	$181,362	$270,059

To store multidimensional data, we conceptualize it as a cube, where each of the axes represents a different dimension of the same information.

OLAP OPERATIONS

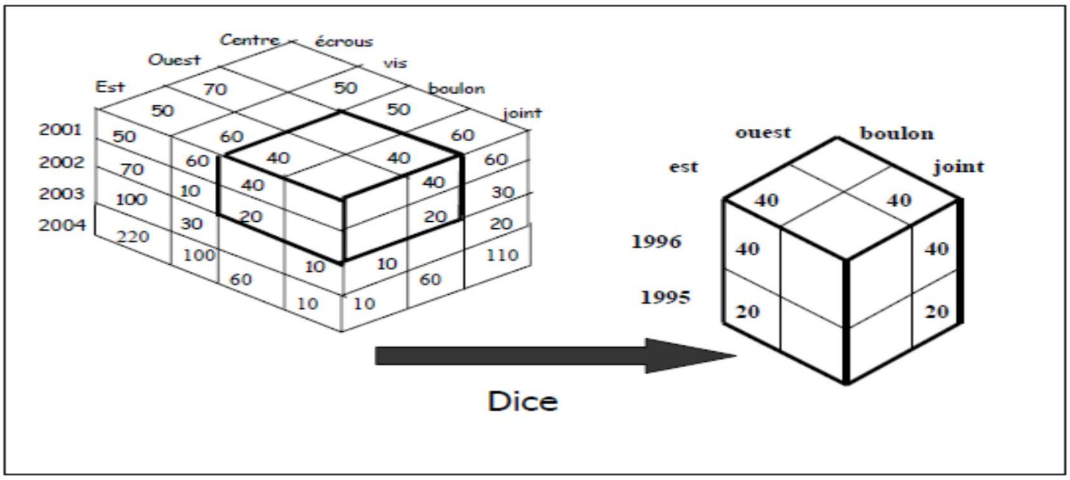
The main operational structure in OLAP is based on a concept called cube. A cube in OLAP provides a user-friendly environment for interactive data analysis. A number of OLAP data cube operations exist to materialize different views of data, allowing interactive querying and analysis of the data.

- The slice operation selects one particular dimension from a given cube and provides a new sub-cube.
- Dice selects two or more dimensions from a given cube and provides a new sub-cube. Consider the following diagram that shows the dice operation.
- Roll-up performs aggregation on a data cube in any of the following ways −
  - *By climbing up a concept hierarchy for a dimension*
  - *By dimension reduction.*
- Drill-down is the reverse operation of roll-up. It is performed by either
  - *By stepping down a concept hierarchy for a dimension*
  - *By introducing a new dimension.*
- Drill Through capability allows users to view relational transactions that make up a multidimensional point in an OLAP Cube. The Drill-Through therefore brings to light data in the Cube constituted from the Data Source, which often is withing an RDBMS.
- Pivot: A pivot is a means of changing the dimensional orientation of a report or ad hoc query-page display.



Ventes 2004	écrou	vis	boulon	joint
est	220	100	60	10
ouest	160	50	10	60
centre	20	150	170	110

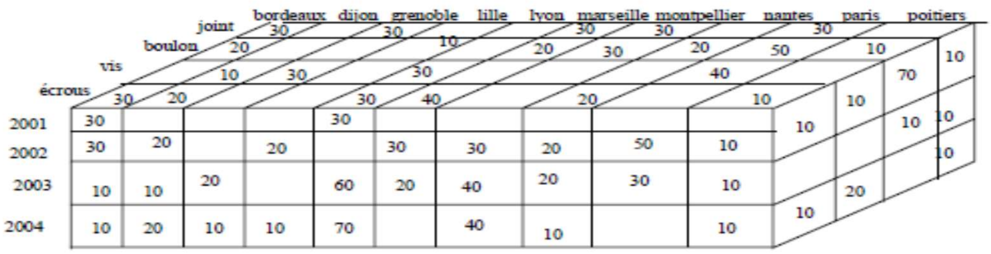
Exemple: Dicing



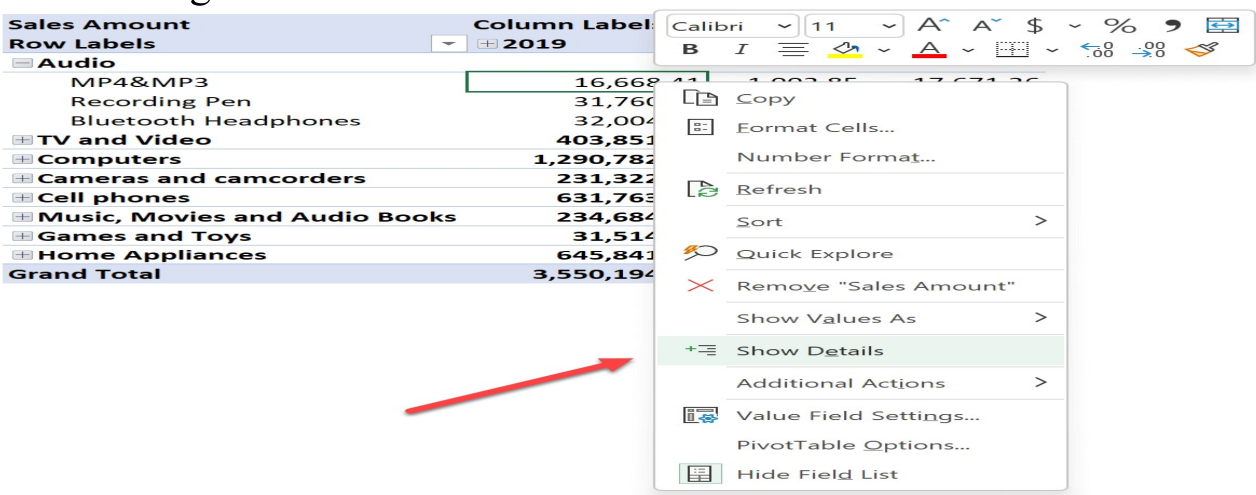
Drill-Down exemple

Drill-down ~ opération inverse de Roll-up

Drill-down du niveau des régions au niveau villes



Drill Through



Data returned for Sales Amount, Audio - MP4&MP3, 2019 (First 1000 rows).

Sales[Order Number]	Sales[Line Number]	Sales[Quantity]	Sales[Unit Price]	Sales[Net Price]	Sales[Unit Cost]	Sales[Currency Code]	Sales[Exchange Rate]
344903	0	1	21.57	21.57	11	EUR	0.8834
328901	0	6	12.99	12.99	6.62	EUR	0.8774
335203	1	4	255	255	84.49	EUR	0.8846
329607	2	8	77.68	77.68	35.72	EUR	0.873
330201	0	2	109.95	98.955	50.56	CAD	1.3265
353000	2	2	59.99	59.3901	30.58	CAD	1.3282
330606	4	3	59.99	52.7912	30.58	CAD	1.3273
362606	3	1	59.99	59.99	30.58	AUD	1.4648
340806	0	10	21.57	21.57	11	AUD	1.4183
343002	0	1	232	232	106.69	GBP	0.7909
334200	2	6	299.23	299.23	99.14	GBP	0.7705
335802	1	8	95.95	84.436	48.92	CAD	1.3402
345200	2	5	95.95	83.4765	48.92	EUR	0.8877
365103	1	2	232	215.76	106.69	EUR	0.8937
357705	0	1	199.9	173.913	91.93	EUR	0.8998
329605	0	4	255	219.3	84.49	USD	1
365209	0	3	134	125.96	61.62	USD	1
345800	1	5	109.95	100.0545	50.56	USD	1
340800	1	3	109.95	97.8555	50.56	USD	1
329003	2	2	21.57	18.7659	11	USD	1
364216	2	2	12.99	11.3013	6.62	USD	1
365100	3	5	299.23	299.23	99.14	USD	1
349400	5	10	12.99	11.4312	6.62	USD	1
335307	3	5	255	249.9	84.49	USD	1
330608	2	10	109.95	109.95	50.56	USD	1
364400	2	6	109.95	109.95	50.56	USD	1
333705	2	2	12.99	12.99	6.62	USD	1
329906	1	2	232	229.68	106.69	USD	1
361901	1	4	255	224.4	84.49	USD	1
358100	0	7	299.23	299.23	99.14	USD	1
347004	0	5	21.57	19.8444	11	USD	1
336203	0	2	21.57	19.6287	11	USD	1
363807	0	2	59.99	53.3911	30.58	USD	1

OLAP implementations [12]

Vendors offer a variety of OLAP products that can be grouped into three categories: multidimensional OLAP (MOLAP), relational OLAP (ROLAP), and hybrid OLAP (HOLAP). Here is a breakdown of the differences between them.

ROLAP

ROLAP stands for Relational Online Analytical Processing. ROLAP stores data in columns and rows (also known as relational tables) and retrieves the information on demand through user submitted queries. A ROLAP database can be accessed through complex SQL queries to calculate information. ROLAP can handle large data volumes, but the larger the data, the slower the processing times.

Because queries are made on-demand, ROLAP does not require the storage and pre-computation of information. However, the disadvantage of ROLAP implementations are the potential performance constraints and scalability limitations that result from large and inefficient join operations between large tables. Examples of popular ROLAP products include Metacube by Stanford Technology Group, Red Brick Warehouse by Red Brick Systems, and AXSYS Suite by Information Advantage.

MOLAP

MOLAP stands for Multidimensional Online Analytical Processing. MOLAP uses a multidimensional cube that accesses stored data through various combinations. Data is pre-computed, pre-summarized, and stored (a difference from ROLAP, where queries are served on-demand).

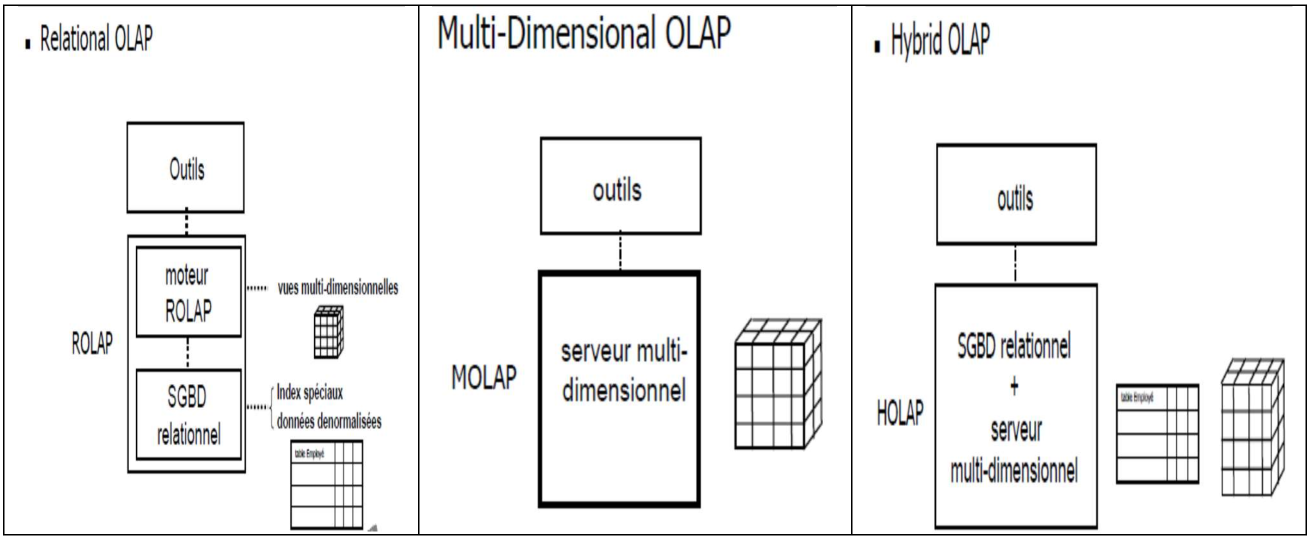
A multicube approach has proved successful in MOLAP products. In this approach, a series of dense, small, precalculated cubes make up a hypercube. Tools that incorporate MOLAP include Oracle Essbase, IBM Cognos, and Apache Kylin.

Its simple interface makes MOLAP easy to use, even for inexperienced users. Its speedy data retrieval makes it the best for “slicing and dicing” operations. One major disadvantage of MOLAP is that it is less scalable than ROLAP, as it can handle a limited amount of data.

HOLAP

HOLAP stands for Hybrid Online Analytical Processing. As the name suggests, the HOLAP storage mode connects attributes of both MOLAP and ROLAP. Since HOLAP involves storing part of your data in a ROLAP store and another part in a MOLAP store, developers get the benefits of both.

With this use of the two OLAPs, the data is stored in both multidimensional databases and relational databases. The decision to access one of the databases depends on which is most appropriate for the requested processing application or type. This setup allows much more flexibility for handling data. For theoretical processing, the data is stored in a multidimensional database. For heavy processing, the data is stored in a relational database.



OLAP SERVER PRODUCTS

Essbase, ssas, IBM Cognos, MicroStrategy Intelligence Server, Mondrian OLAP server,SAS OLAP Server, StarRocks, icCube, Jedox Kyvos ,Apache Kylin, Apache Pinot , Apache Druid, ClickHouse,

OLTP vs OLAP

Online transaction processing (OLTP) captures, stores, and processes data from transactions in real time.

Online analytical processing (OLAP) applies complex queries to large amounts of historical data, aggregated from OLTP databases and other sources, for data mining, analytics, and business intelligence projects

	OLTP	OLAP
Characteristics	Handles a large number of small transactions Simple standardized queries	Handles large volumes of data with complex queries
Purpose	Control and run essential business operations in real time. day-to-day business transactions	Plan, solve problems, support decisions, discover hidden insights
Operations	Based on INSERT, UPDATE, DELETE commands	Based on SELECT commands to aggregate data for reporting
Database design	Normalized databases for efficiency	Denormalized databases for analysis
Response time	Milliseconds	Seconds, minutes, or hours depending on the amount of data to process
Source	Transactions	Aggregated data from transactions
Data updates	Short, fast updates initiated by user	Data periodically refreshed with scheduled, long-running batch jobs
Space requirements	Generally small if historical data is archived	Generally large due to aggregating large datasets
Users	Customer-facing personnel, clerks, online shoppers	Knowledge workers such as data analysts, business analysts, and executives
Example	ATM center	Amazon analyzes purchases by its customers to come up with a personalized home page with products which likely interest to their customer.

Data Warehouse development Approaches [13]

Two competing approaches are employed. The first approach is that of Bill Inmon, who is often called "the father of data warehousing." Inmon supports a top-down development approach that adapts traditional relational database tools to the development needs of an enterprise-wide data warehouse, also known as the EDW approach.

The second approach is that of Ralph Kimball, who proposes a bottom-up approach that employs dimensional modeling, also known as the data mart approach. Kimball's data mart strategy is a "plan big, build small" approach.

Data Warehouse Architectures

Data Warehouse Architecture is the conceptualization of how the data warehouse is built. The scientific literature often distinguishes five types of architecture for data warehouse systems

1. Independent data marts architecture

Has its origins in DSS where the data was organized around the application rather than being treated as an organization wide resource.

While independent data marts may meet localized needs, they do not provide a single version of the truth for the entire organization. They typically have inconsistent data definitions and inconsistent dimensions and measures that make it difficult to run distributed queries across the marts. They are also costly and time consuming to maintain.

- In independent data marts architecture, different data marts are separately designed and built in a non-integrated fashion.
- This architecture can be initially adopted in the absence of a strong sponsorship toward an enterprise-wide warehousing project, or when the organizational divisions that make up the company are loosely coupled.
- It tends to be soon replaced by other architectures that better achieve data integration and cross-reporting.

2. The bus architecture, recommended by Ralph Kimball

This architecture is a viable alternative to the independent data marts where the individual marts are linked to each other via some kind of middleware. Because the data are linked among the individual marts, there is a better chance of maintaining data consistency across the enterprise (at least at the metadata level). Even though it allows for complex data queries across data marts, the performance of these types of analysis may not be at a satisfactory level.

- Similar to the preceding architecture, with adoption of conformed dimensions.

- One mart is created for a single business process and additional marts are developed using the conformed dimensions of the first mart
- Notice that there is no central data warehouse with normalized data; all of the data is stored in the marts in a dimensional data model.

3. The hub-and-spoke architecture

The building of an enterprise data warehouse starts with an enterprise-level analysis of data requirements (Inmon et al. 2001). Special attention is given to building a scalable infrastructure. Using this enterprise view of data, the architecture is developed in an iterative manner, subject area by subject area (e.g., marketing data, sales data). Data is stored in the warehouse in 3rd normal form. Dependent data marts are created that source data from the warehouse.

The dependent marts can either be physical (i.e., a separate server) or logical (i.e., a view within the warehouse). The dependent data marts may be developed for departmental, functional area, or special purposes (e.g., data mining/predicted analytics) and use a data model appropriate for its intended use (e.g., relational, multidimensional).

Because the dependent data marts obtain their data from the warehouse, a single version of the truth is maintained.

4. The centralized architecture Recommended by Bill Inmon

Can be seen as a particular implementation of the hub-and-spoke architecture without data marts.

5. The federated architecture

- The federated architecture is sometimes adopted in dynamic contexts where preexisting data warehouses/data marts are to be integrated to provide a single, cross-organization decision support environment (for instance, in the case of mergers and acquisitions).
- Data warehouse/data mart are either virtually or physically integrated.
- It often exists when acquisitions, mergers, and reorganizations have resulted in a complex decision support environment that would be costly and time-consuming to integrate.

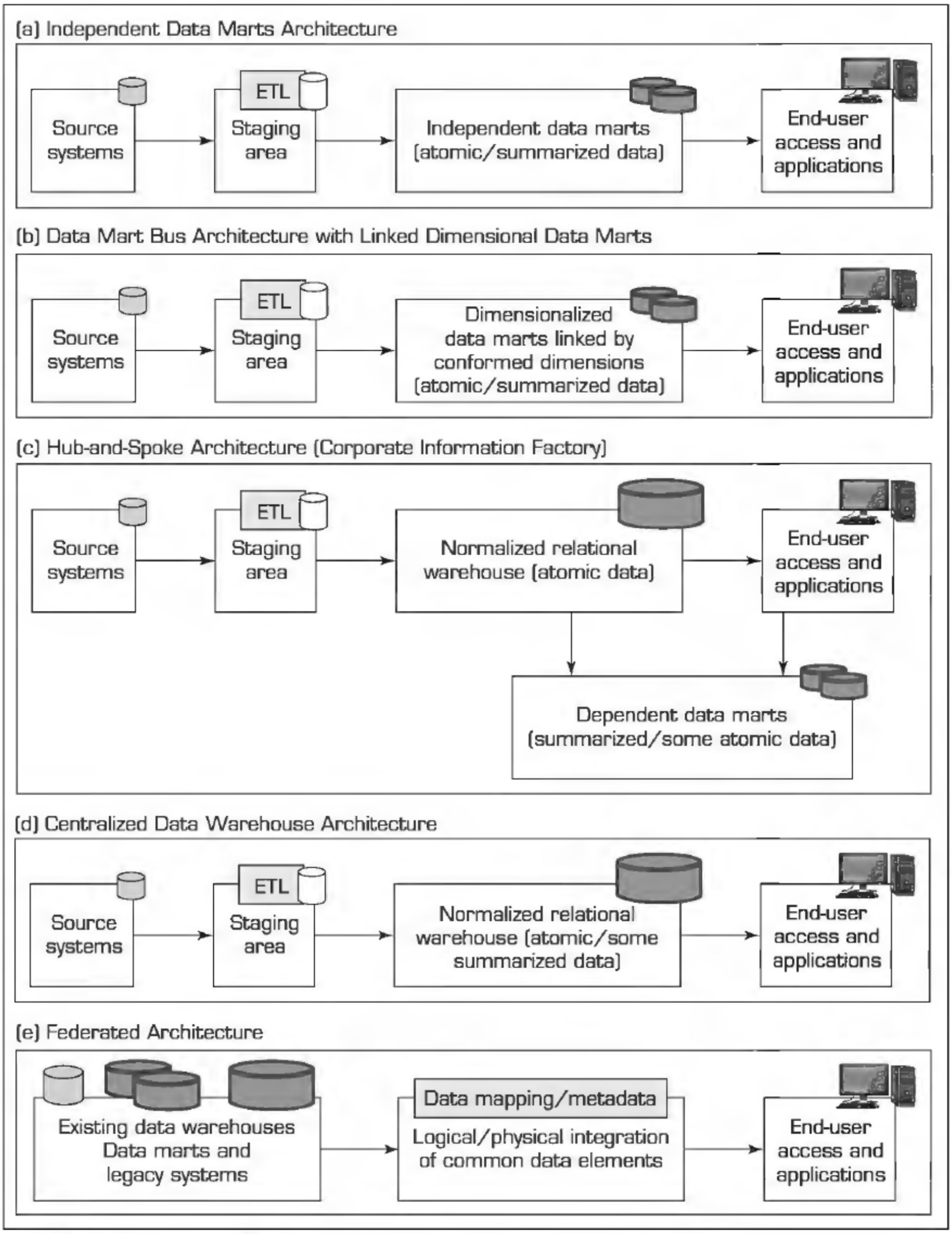
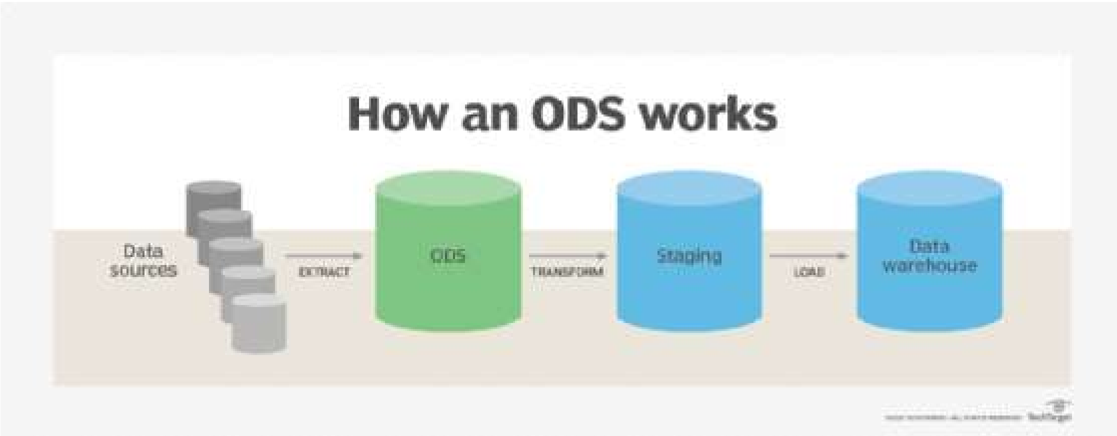


FIGURE 3.7 Alternative Data Warehouse Architectures. *Source:* Adapted from T. Ariyachandra and H. Watson, "Which Data Warehouse Architecture Is Most Successful?" *Business Intelligence Journal,* Vol. 11, No. 1, First Quarter, 2006, pp. 4–6.

Operational data store (ODS)

- An operational data store (ODS) is a type of database that's often used as an interim logical area for a data warehouse.

- An operational data store usually stores and processes data in real time. An ODS is connected to multiple data sources and pulls data(ETL) into a central location for activities such as operational reporting and real-time analysis.
- In some cases, data from an ODS is replicated and then ETL is used to transport the replicated data to a data warehouse.
- As operational data stores ingest data, new incoming data overwrites existing data.
- ODS tools contain up-to-date versions of business data integrated from data sources, which is useful for BI tasks such as managing logistics, tracking orders and monitoring customer activity.



Cloud data warehouses

Data warehouses are the foundation for business intelligence. They replicate data from multiple sources into a repository that can serve as a single source of truth for an organization's analytics. The repository is structured and optimized for analytics processing.

Today, cloud data warehouses replace some legacy on-premises systems, and they are a popular choice for many organizations adopting new systems.

Cloud data warehouses offer speed and scalability that legacy systems cannot. The cloud platform provides the ability to quickly scale to meet just about any processing demands. Administrators can scale processing and storage resources up or down with a few mouse clicks.

Snowflake, Amazon Redshift, Microsoft Azure Synapse Analytics, and Google BigQuery, are examples of cloud data warehouses.

Snowflake is a cloud-based data warehouse that runs on AWS, GCP, and Azure. Unlike the other data warehouses, Snowflake is the only solution that doesn't run on its own cloud.

Types of Facts

There are three types of facts:

1. Additive Facts

Additive facts can be used with any aggregation function like Sum(), Avg() etc. Example is Quantity, sales amount etc.

2. Semi Additive Facts

Semi-additive facts are those where only a few of aggregation function can be applied.
For example, Consider bank account details. You cannot apply the Sum() on the bank balance that does not give useful results but min() and max() function may return useful information.

3. Non-Additive Facts

You cannot use numeric aggregation functions such as Sum(), Avg() etc on Non-additive facts.
For example of non-additive fact is any kind of ratio or percentage. Non numeric facts can also be a non-additive facts.

Types of Dimensions

1. Slowly Changing Dimensions
2. Rapidly Changing Dimensions
3. Junk Dimensions
4. Inferred Dimensions
5. Conformed Dimensions
6. Degenerate Dimensions
7. Role Playing Dimensions
8. Shrunken Dimensions
9. Static Dimensions

1. Slowly Changing Dimensions

Attributes of a dimension that would undergo changes over time. It depends on the business requirement whether particular attribute history of changes should be preserved in the data warehouse. This is called a slowly changing attribute and a dimension containing such an attribute is called a slowly changing dimension.
There are 6 types of Slowly Changing Dimension. The more commonly used are:
Type 0 – Fixed Dimension. No changes allowed, dimension never changes.
Type 1 – No History. ...

Type 2 – Row Versioning. ...

Type 1:
The new record replaces the original record. No trace of the old record exists. For example, if we originally have the following table:

Customer Key	Name	State
1001	Williams	New York

After Williams moved from New York to Los Angeles, the new information replaces the new record, and we have the following table:

ustomer Key	Name	State
1001	Williams	Los Angeles

Type 2:
A new record is added into the customer dimension table. Thereby, the customer is treated essentially as two people.

In our example, recall we originally have the following table:

Customer Key	Name	State
1001	Williams	New York

After Williams moved from New York to Los Angeles, we add the new information as a new row into the table table (assuming the effective date of change is February 20, 2021):

In order to support type 2 changes, we need to add four columns to our table:

Surrogate Key – the original Customer Key will no longer be sufficient to identify the specific record we require, we therefore need to create a new ID that the fact records can join to specifically.

· Start Date – The date from which the specific historical version is active

· End Date – The date to which the specific historical version record is active

With these elements in place, our table will now look like:

New ID	Customer Key	Name	State	Start date	End date
1	1001	Williams	New York	20-FEB-2010	20-FEB-2021
2	1001	Williams	Los Angeles	20-FEB-2021	NULL

2. Rapidly Changing Dimensions

A dimension attribute that changes frequently is a rapidly changing attribute. If you don't need to track the changes, the rapidly changing attribute is no problem, but if you do need to track the changes, using a standard slowly changing dimension technique can result in a huge inflation of the size of the dimension. One solution is to move the attribute to its own dimension, with a separate foreign key in the fact table. This new dimension is called a rapidly changing dimension.

3. Junk Dimensions

A junk dimension is a single table with a combination of different and unrelated attributes to avoid having a large number of foreign keys in the fact table. Junk dimensions are often created to manage the foreign keys created by rapidly changing dimensions.

4. Inferred Dimensions

While loading fact records, a dimension record may not yet be ready. One solution is to generate a surrogate key with null for all the other attributes. This should technically be called an inferred member, but is often called an inferred dimension.

5. Conformed Dimensions

A dimension that is used in multiple locations is called a conformed dimension. A conformed dimension may be used with multiple fact tables in a single database, or across multiple data marts or data warehouses.

6. Degenerate Dimensions

A degenerate dimension is when the dimension attribute is stored as part of fact table, and not in a separate dimension table. These are essentially dimension keys for which there are no other attributes. In a data warehouse, these are often used as the result of a drill through query to analyze the source of an aggregated number in a report. You can use these values to trace back to transactions in the OLTP system.

7. Role Playing Dimensions

A role-playing dimension is one where the same dimension key — along with its associated attributes — can be joined to more than one foreign key in the fact table. For example, a fact table may include foreign keys for both ship date and delivery date. But the same date dimension attributes apply to each foreign key, so you can join the same dimension table to both foreign keys. Here the date dimension is taking multiple roles to map ship date as well as delivery date, and hence the name of role playing dimension.

8. Shrunken Dimensions

A shrunken dimension is a subset of another dimension. For example, the orders fact table may include a foreign key for product, but the target fact table may include a foreign key only for productcategory, which is in the product table, but much less granular. Creating a smaller dimension table, with productcategory as its primary key, is one way of dealing with this situation of heterogeneous grain. If the product dimension is snowflaked, there is probably already a separate table for productcategory, which can serve as the shrunken dimension.

9. Static Dimensions

Static dimensions are not extracted from the original data source, but are created within the context of the data warehouse. A static dimension can be loaded manually — for example with status codes — or it can be generated by a procedure, such as a date or time dimension.

Types of fact tables

Following are the three types of fact tables:

- Transaction fact table
- Snapshot fact table
- Accumulating snapshot fact table

Transaction fact table

A transaction fact table captures detailed information about individual business transactions or events. It records every occurrence at the most granular level, providing a comprehensive view of operational data.

Example

Consider a retail business with a transaction fact table capturing sales data. The table may include the following columns:

Transaction ID	Product ID	Customer ID	Quantity	Unit price	Total sales
1000	100	200	2	10	20
10001	101	201	4	15	60
10002	102	202	5	20	100

Snapshot fact table
A periodic snapshot fact table captures aggregated metrics at specific intervals or time periods, summarizing business activities. It provides a concise overview of performance and trends over time.

Example

Snapshot date	Total units	Average defects	Total sales
2023-06-19	300	0.1	1000
2023-06-20	400	0.05	1500
2023-06-21	450	0.2	2000

Consider a manufacturing company that maintains a periodic snapshot fact table to track monthly production data:

Accumulating snapshot fact table
An accumulating snapshot fact table tracks the incremental progress of a specific business process or workflow. It captures significant milestones and associated measures throughout the process.

Example
Consider an e-commerce company with an accumulating snapshot fact table monitoring the order fulfillment process:

Order ID	Order date	Shipping date	Delivery date	Total days
501	2023-06-01	2023-06-05	2023-06-08	7
502	2023-06-02	2023-06-06	2023-06-10	8
503	2023-06-03	2023-06-07	2023-06-13	10

In the above example, the table tracks the key stages of the order fulfillment process. Each row represents a unique order. The accumulating snapshot fact table allows easy analysis of the order fulfillment process by capturing the dates at each stage. It enables organizations to measure the time taken for each step, monitor the status of orders in real-time, identify bottlenecks, and optimize the overall order fulfillment cycle.

Factless Fact Table

A factless fact table is a fact table that does not have any measures. It is essentially an intersection of dimensions.

For example, the fact table of student attendance in classes would consist of 3 dimensions: the student dimension, the time dimension, and the class dimension. With this factless fact table one can easily answer the following questions:

How many students attended a particular class on a particular day?

How many classes on average does a student attend on a given day?

Real-time business intelligence [8] or Operational BI

The rise of big data and the growing demand for real-time data analysis have led to the development of the real-time data warehouse (RTDW), also known as active data warehousing (ADW), Real-time data warehousing, near- real-time data warehousing, zero-latency warehousing.

Active Data Warehousing is a data management and analytics approach that integrates real-time data processing and analytics capabilities with traditional data warehousing. It combines the benefits of both approaches to deliver timely and actionable insights to businesses.

In traditional data warehousing, data is typically stored and processed in batch mode, meaning that there is a delay between data being collected and insights being generated. Active Data Warehousing, on the other hand, enables businesses to process and analyze data in near real-time, allowing for faster decision-making and more timely insights.

While traditional BI presents historical data for manual analysis, RTBI **compares current** business events with **historical** patterns to **detect** problems or opportunities automatically. This automated analysis capability enables **corrective** actions to be initiated and/or business rules to be **adjusted** to optimize business processes.
RTBI has several types of deployment and operational architectures, including:

- Event-based data analytics that trigger the detection of specific data events
- Server-less data analytics used to directly extract data from the source, rather than the data warehouse or repository

Stream mining, **in-memory analytics**, and **in-database analytics**—have enabled **real-time analytics** for customer interactions, fraud detection, and situational intelligence. Business activity monitoring (BAM) utilizes real-time data and dashboards in order to monitor current operations.

Columnar database

- A columnar database is a type of database management system that stores data in columns rather than rows, optimizing query performance by enabling efficient data retrieval and analysis. It is important for enhanced analytics and reporting, offering faster query speeds and improved compression.
- Columnar storage enables better data compression due to the similarity of data within a column, and it allows for more efficient querying and aggregation, especially in analytical and reporting tasks.
- Columnar databases are particularly well-suited for handling large-scale data warehousing and big data analytics applications where operations often involve aggregating, summarizing, or searching across large datasets.
- Columnar databases store data vertically, optimizing for analytics and large-scale aggregations, while row-oriented databases store data horizontally, prioritizing transactional processing and quick access to entire records.
- some of the examples of columnar database: Apache Cassandra, Amazon redshift, Google BigQuery, Vertica, ClickHouse, Snowflake

Change data capture (CDC)

Change data capture (CDC) refers to the process of identifying and capturing changes made to data in a database and then delivering those changes in real-time to a downstream process or system.

Capturing every change from transactions in a source database and moving them to the target in real-time keeps the systems in sync and provides for reliable data replication.

Change Data Capture Methods

- Log-based CDC. This is the most efficient way to implement CDC. When a new transaction comes into a database, it gets logged into a log file with no impact on the source system. And you can pick up those changes and then move those changes from the log.
- Query-based CDC. Here you query the data in the source to pick up changes. This approach is more invasive to the source systems because you need something like a timestamp in the data itself.
- Trigger-based CDC. In this approach, you change the source application to trigger the write to a change table and then move it. This approach reduces database performance because it requires multiple writes each time a row is updated, inserted, or deleted.

Database vendors provide embedded CDC or change data capture technology. The newest generation of log-based CDC tools are fully integrated with most ETL tools and a wide variety of source and target systems such as CDC for Oracle and SQL Server CDC. They can also replicate data to targets such as Snowflake and Azure. This lets you leverage one tool for all of your real-time data integration and data warehousing needs.

Bibliographie

[1] B. Travica, Decision Making and Information Systems (Chapter 9.

[2] «https://www.simplekpi.com/Blog/KPI-Reports-Explained-A-complete-guide,» [En ligne].

[3] «https://clockify.me/blog/business/kpi/,» [En ligne].

[4] «https://en.wikipedia.org/wiki/Dashboard_(business),» [En ligne].

[5] «https://en.wikipedia.org/wiki/Extract,_transform,_load,» [En ligne].

[6] «https://www.mssqltips.com/sqlservertip/5937/etl-vs-elt-features-and-use-cases/,» [En ligne].

[7] «https://www.guru99.com/dimensional-model-data-warehouse.html,» [En ligne].

[8] «https://en.wikipedia.org/wiki/Real-time_business_intelligence,» [En ligne].

[9] «https://www.sqlshack.com/an-overview-of-etl-and-elt-architecture/,» [En ligne].

[10] «https://docs.oracle.com/cd/B13789_01/olap.101/b10333/multimodel.htm,» [En ligne].

[11] «https://www.xplenty.com/blog/snowflake-schemas-vs-star-schemas-what-are-they-and-how-are-they-different/,» [En ligne].

[12] «https://www.sisense.com/glossary/olap/#:~:text=HOLAP%20stands%20for%20Hybrid%20Online,» [En ligne].

[13] «https://panoply.io/data-warehouse-guide/data-mart-vs-data-warehouse/,» [En ligne].

[14] «http://www2.cs.uregina.ca/~dbd/cs831/notes/kdd/1_kdd.html,» [En ligne].

[15] «https://www.guru99.com/data-mart-tutorial.html,» [En ligne].

[16] «https://www.mssqltips.com/sqlservertip/5937/etl-vs-elt-features-and-use-cases/,» [En ligne].

[17] «https://www.xplenty.com/blog/snowflake-schemas-vs-star-schemas-what-are-they-and-how-are-they-different/,» [En ligne].

[18] «Wikipedia,» [En ligne].

[19] https://www.techtarget.com/searchbusinessanalytics/definition/big-data-analytics

[24] https://cloud.google.com/discover/what-is-a-data-lakehouse?authuser=0#:~:text=A%20data%20lakehouse%20is%20a%20modern%20data%20architecture%20that%20creates,organized%20sets%20of%20structured%20data