Data is copied ... not moved

Data remains (còn) in our source systems and the copies are made and send into data warehouse

Rule that goven (chi phối) how we built our data warehouses and how whe organize and store our data

1. Data warehouse is an integrated environment

2. Data warehouse should be subject-oriented (định hướng theo chủ thể) (bất kể)
regardless of how many systems and which data come from which systems, we need to reorganize the data by subjects (chủ đề)

3. Time variant

Data warehouse contain historical data, not just current data

4. Non volatile (Tính e biến động)

Traditionally, we will periodically (định kỳ) load data into a data ware house Think of it as refreshing a data warehouse to keep it current and we do so in batches (nhóm , lô)

Batwoen the time of the last refresh and the next one the data warehouse stays as is , even if thousands, or perhap millions of transactions are occurring in our transactional systems . That is non-volatile

=> Data warehouse ramain stable is between refreshes

So we can the do things like stategic planning without the data chaing underneath us

We bring data to data ware house

=> We'll typically restructure and reorganize it to make it more useful for analysis

5. Why ?

To support data - driven (hướng dữ liệu) decision making

Reasons for you to built a Data Warehouse

1. Making data - driven decisions (quyết định theo hướng dữ liệu)

(trực giác)

Rather than having rely on solely on experience and intuition and even hunches (linh cảm)

2. One stop shopping (1 cửa hàng mua sắm)

In other words, the data that we need is all in a single location, rather than scattered (rải rác) among the transactional and operational applications where we get that data from

Making data driven decision

+ Past
+ Present
+ Future
+ Unknow

Compare Data Warehouse and Data lake

1. Data warehouse is often built on top of relational database

Some time data warehouse built on top of a multi dimentional database that's typically know as cube

2. Data lake built on top of some soft big data environment rather than a traditional relation data base

Diffrences

1. Volume

Big data and it usage in datalakes help us manage extremly large volume of data larger than we typically would include even the largest of data warehouses

Volume is one of the traditional three V's of big data

2. Velocity

(nhanh) (tiếp nhận)

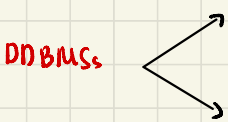
Big data also supports very rapid intake of new and change data much more rapidly than we typically do with traditional data warehousing

Variety (đa dạng)

Big data easily supports along with semi-stuctured data such as text messages, email, blogs and complex documents, as well as unstuctured data such as audio and video, ...

Compare data ware house to Data Virtualization

The root of data ware housing



Data virtualization (ảo hoá)

Data warehousing

Data virtualization

+ Can be thought of as a read-only distributed database
+ In-place data access. We access it from its original locations at the time we need to do so for reports and analytics
+ Many name over the years

Data virtualization use case

1. Simple transformations

If we have data that requires simple tranformations or perhaps even no transformations to use in BI and analytics

2. Small number of data source

3. Relaxed response time

Look at a simple end-to-end Data Warehousing Environment
(từ đầu → cuối)

1. A typical data warehousing environment

Data sources $\xrightarrow{\text{ETL}}$ Data warehouse

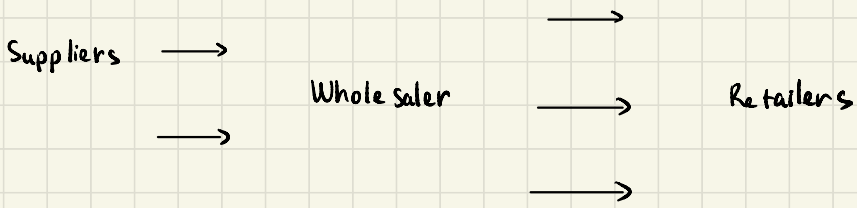
ETL

* Extract
* Tranform
* Load

2. Adding Complexity

Data source $\xrightarrow{\text{ETL}}$ Data Warehouse $\xrightarrow{\text{ETL}}$ Data marts

Ex:



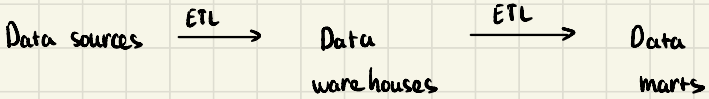
Built a centralized Data warehouse

1. Is a single data warehouse environment
2. With centralize data ware house you have a single database. Everything from all of your sources will feed into that single database
3. Support one-stop shopping

Historical challenges

1. Technology: face with large data volume
2. Work processes :
3. Organizational and human factors

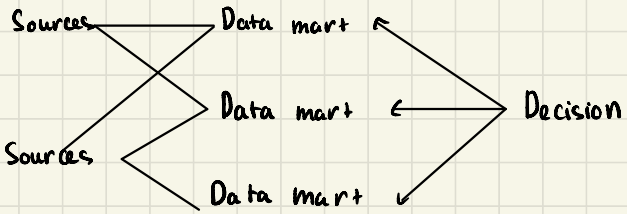
Compare a data ware house to a data mart



Dependent data marts

1. No data warehouse no data marts because their can be supplied with data

Independent data marts



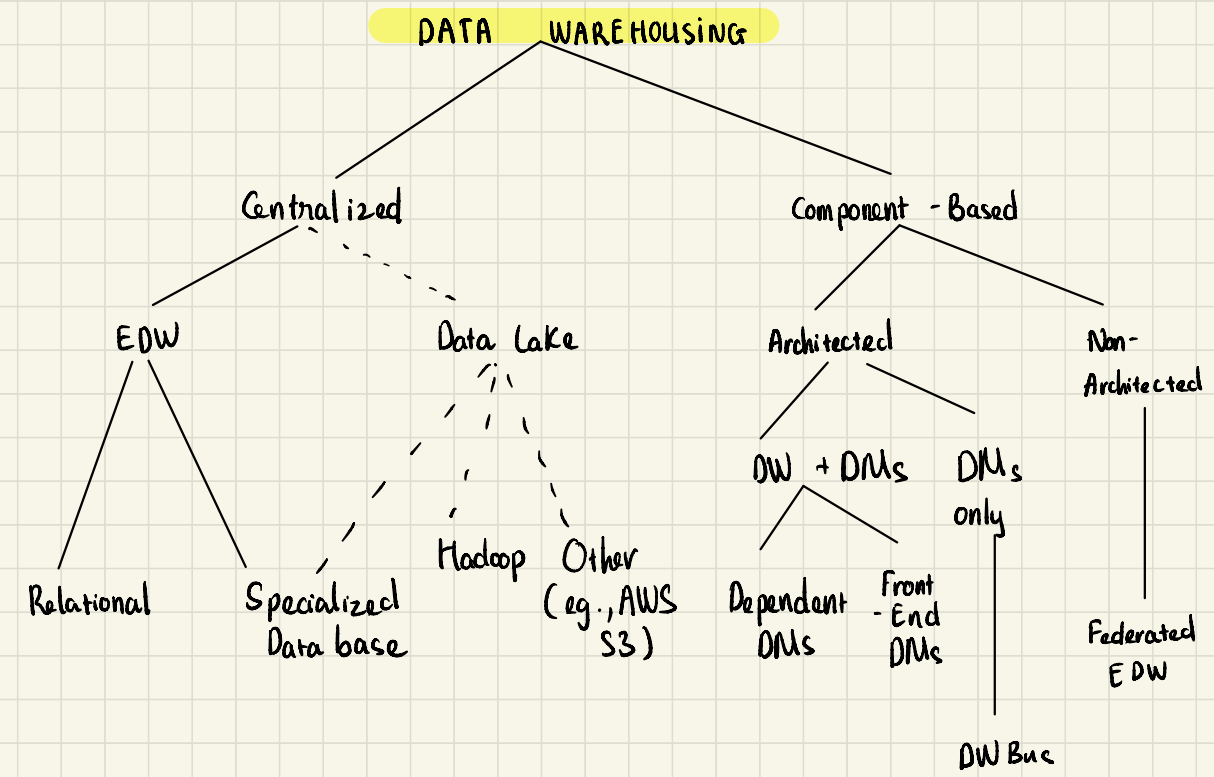
1. Independent data mart doesn't need a data warehouse
2. Each independent data mart draws data directly from one or more source application

Dependent	Independent
Sourced from data warehouse	Sources directly from applications and system
(Mostly) uniform data across marts thống nhất	Little or no uniformity across marts ≠ thống nhất
Architecturally straightforward	"Spaghetti" architecture

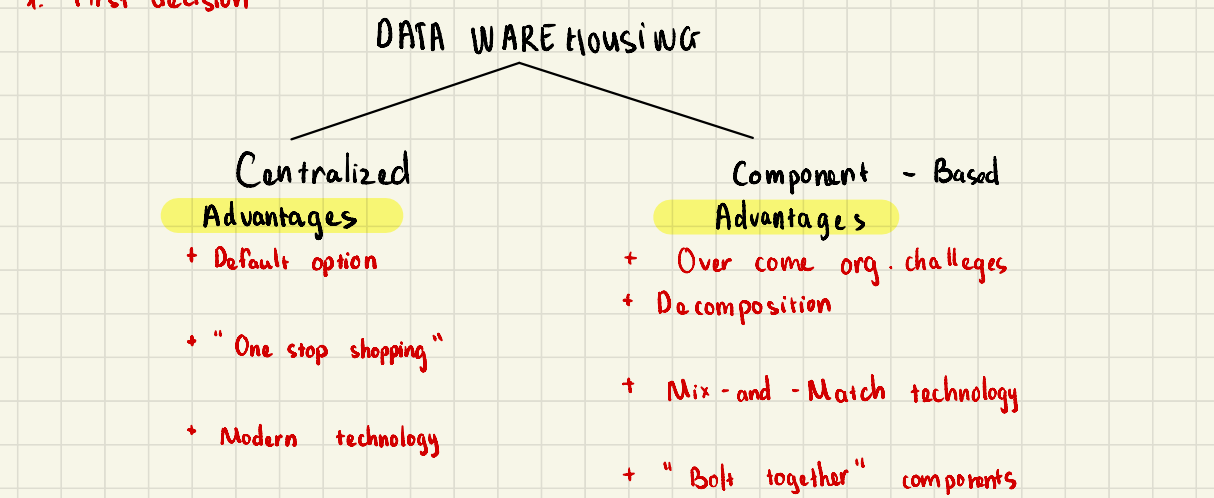
Data warehouse	Independent data mart
Many sources	One or more source
ETL from sources	ETL from sources
Probably larges data volumes	Possibly large data volumes
Dimensionally organized data	Dimensionally organized data

Your Data Warehousing Architectural Options

Many architectural options



1. First decision



Disadvantages

+ High cross-org cooperation
(Yêu cầu hợp tác tổ chức cheó cao)

+ High data governance
(Yêu cầu cao các quy tắc)

+ Ripple effects
(Hiệu ứng luồn sóng,
khi có 1 thay đổi nhỏ diễn
ra thì sẽ ảnh hưởng tổng
thể)

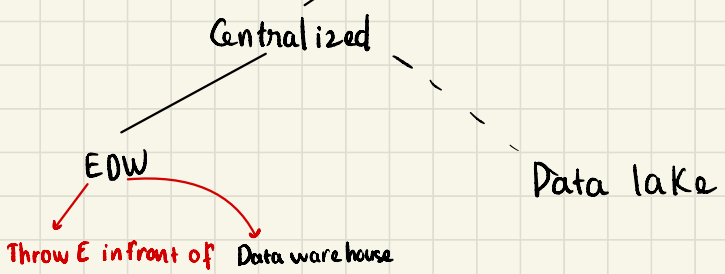
Disadvantages

+ Often inconsistent data

+ Difficult to cross - integrate
(tích hợp cheó)

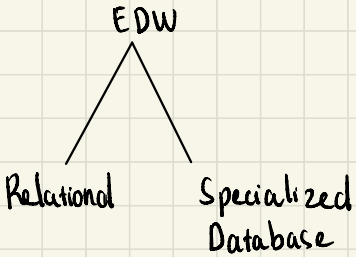
2. Emphasis on "enterprise"

DATA WAREHOUSING

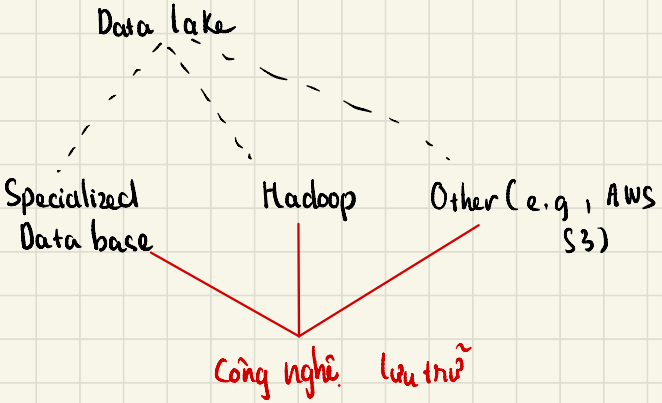


=> We can think of an enterprise data warehouse (Kho dữ liệu doanh nghiệp)

as the default approach when we're building centralized environment

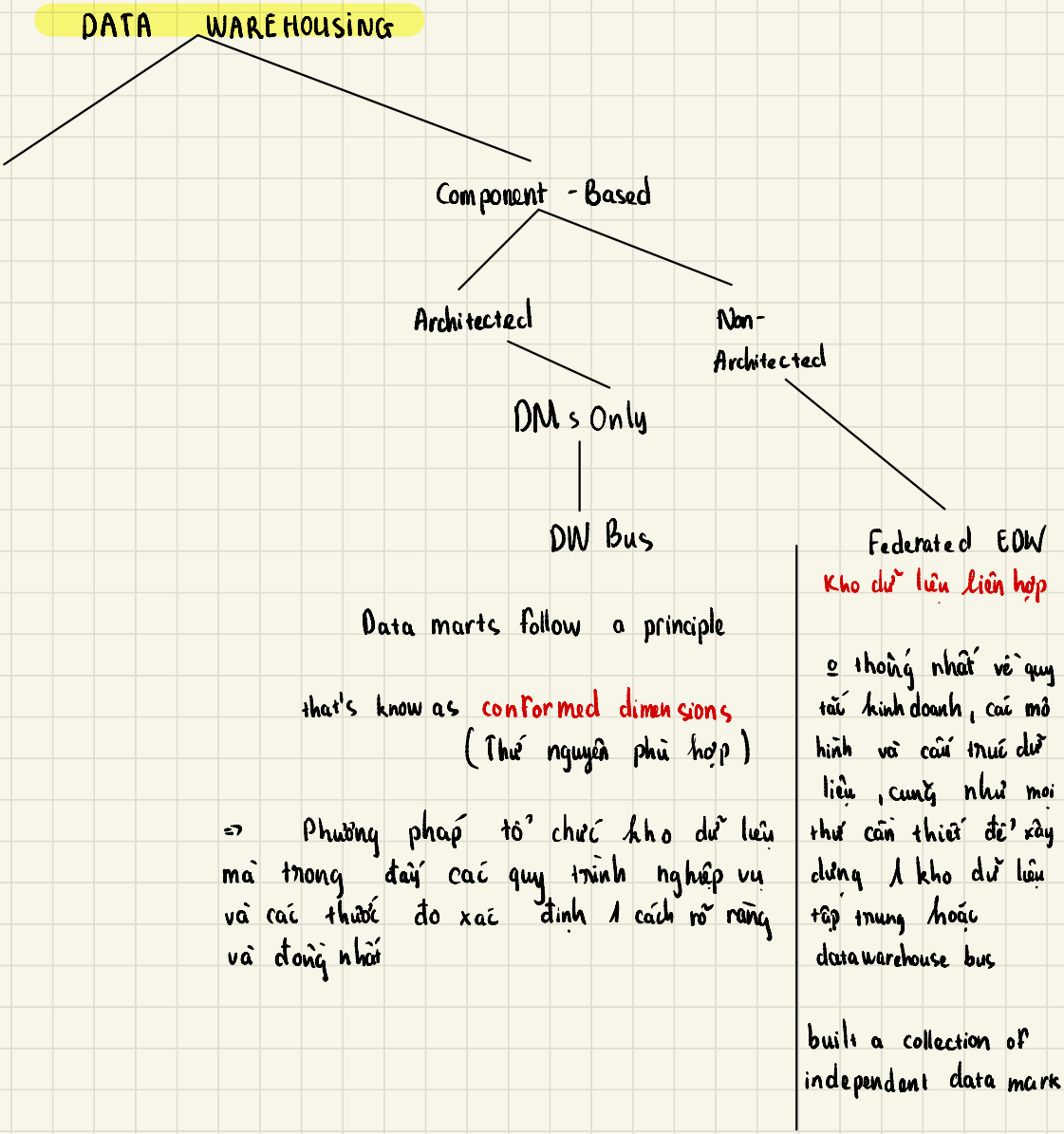


Specialized large scale database and other things that are typically know as Warehousing applicances (thiết bị lưu trữ dữ liệu)



DATA WAREHOUSING

- Component - Based (Hướng thành phần)
  - Architected (có cấu trúc)
    - DW + DMs (Data warehouse + Data Marts)
      - Dependent DMs
        - Data sources --ETL--> Data warehouse --ETL--> Data Marts
        - Variation of Dependent DMs call (biến thể)
          - Coporate information factory or CIF (nhà máy thông tin)
      - Front-end DMs
        - Data Marts ⟶ Data Warehouse
    - DMs Only (Data Marts only)
  - Non-Architected (ọ có cấu trúc)


DATA WAREHOUSING
Component - Based
Architected
Non-Architected
DMs Only
DW Bus
Federated EDW
Kho dữ liệu liên hợp
Data marts follow a principle
that's know as conformed dimensions
(Thứ nguyên phù hợp)
=> Phương pháp tổ chức kho dữ liệu mà trong đấy các quy trình nghiệp vụ và các thước đo xác định 1 cách rõ ràng và đồng nhất
o thống nhất về quy tắc kinh doanh, các mô hình và cấu trúc dữ liệu, cũng như mọi thứ cần thiết để xây dựng 1 kho dữ liệu tập trung hoặc data warehouse bus
built a collection of independent data marts

What is a cube ?

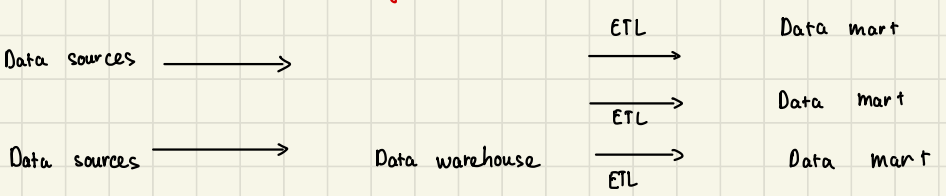
* Cube = Multi dimensional database (MDBMS)
* Not a relational data base (RDBMS)
* Specialized "dimensionally-aware" database

Today : best for smaller-scale DWs, DMs

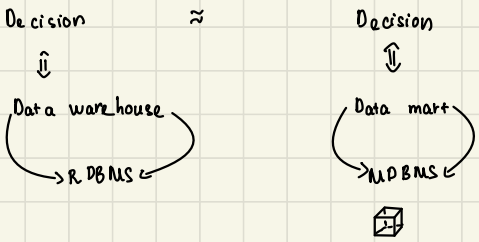
CUBE : advantages and disadvantages

1. Fast query respone time
2. "Modest" data volumes (khiêm tốn)
3. Less flexible data structures than RDBMS

Data warehouses and marts together



From user perspective ...



Including Operational Data Stores in Your Data Warehousing environment

What is the difference between a DW and an ODS?

What is ODS?

* Integrates data from multiple source
* Emphasis on current operational data
  ↓
  Nhấn mạnh
* Often real-time source ⟶ ODS data feeds
  ↘
  Trực tiếp truyền data hơn
  thay vì xếp hàng đợi datawarehouse refresh
* "Tell me what happening right now"
* Popular late 1990s / early 2000s

ODSs and warehouse s: Option 1