DATA WAREHOUSES AND DECISION SUPPORT SYSTEMS

(DW and DSS)

Chương 2
Các vấn đề cơ bản trong kho dữ liệu

Vietnam National University – HCMC
Ho Chi Minh City University of Technology
Faculty of Computer Science & Engineering


Mr. Bui Tien Duc, Meng


tienducut@gmail.com


0769690731

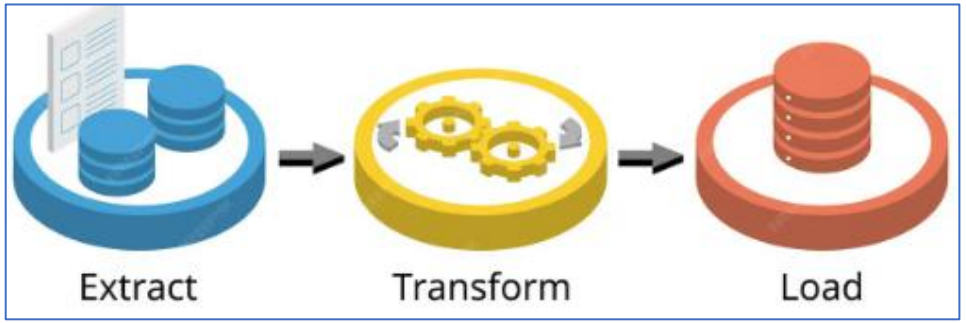
Introduction

1. Các khái niệm cơ bản về kho dữ liệu
2. Kiến trúc kho dữ liệu
3. Các đặc tính về dữ liệu của kho dữ liệu
4. Vấn đề về độ mịn dữ liệu
5. Vấn đề về chuyển đổi dữ liệu
6. Vấn đề về dữ liệu dẫn xuất
7. Vấn đề về siêu dữ liệu
8. Các vấn đề khác về kho dữ liệu

5. Vấn đề về chuyển đổi dữ liệu

5.1 Khái niệm **ETL** trong chuyển đổi dữ liệu

ETL (**E**xtract – **T**ransform – **L**oad) là quá trình chuẩn hoá và đưa dữ liệu từ hệ thống nguồn vào kho dữ liệu (Data Warehouse).



Trong đó:

Extract (Trích xuất): lấy dữ liệu từ các hệ thống nguồn (OLTP, file Excel, CSV, API,…).

Transform (Chuyển đổi): biến đổi dữ liệu về dạng thống nhất, phù hợp với cấu trúc của kho dữ liệu.

Load (Nạp): nạp dữ liệu đã xử lý vào các bảng fact và dimension trong Data Warehouse.

Chuyển đổi dữ liệu là giai đoạn trung tâm, đảm bảo dữ liệu từ nhiều nguồn không đồng nhất được tích hợp và trở thành dữ liệu có ý nghĩa, sẵn sàng phục vụ phân tích.

5. Vấn đề về chuyển đổi dữ liệu

ETL trong chuyển đổi dữ liệu

5. Vấn đề về chuyển đổi dữ liệu

5.2 Vai trò của chuyển đổi dữ liệu trong xây dựng kho dữ liệu

Kết nối dữ liệu từ nhiều nguồn: Các hệ thống vận hành khác nhau thường có cấu trúc và định dạng dữ liệu khác nhau. Chuyển đổi dữ liệu giúp hợp nhất chúng thành một chuẩn chung.

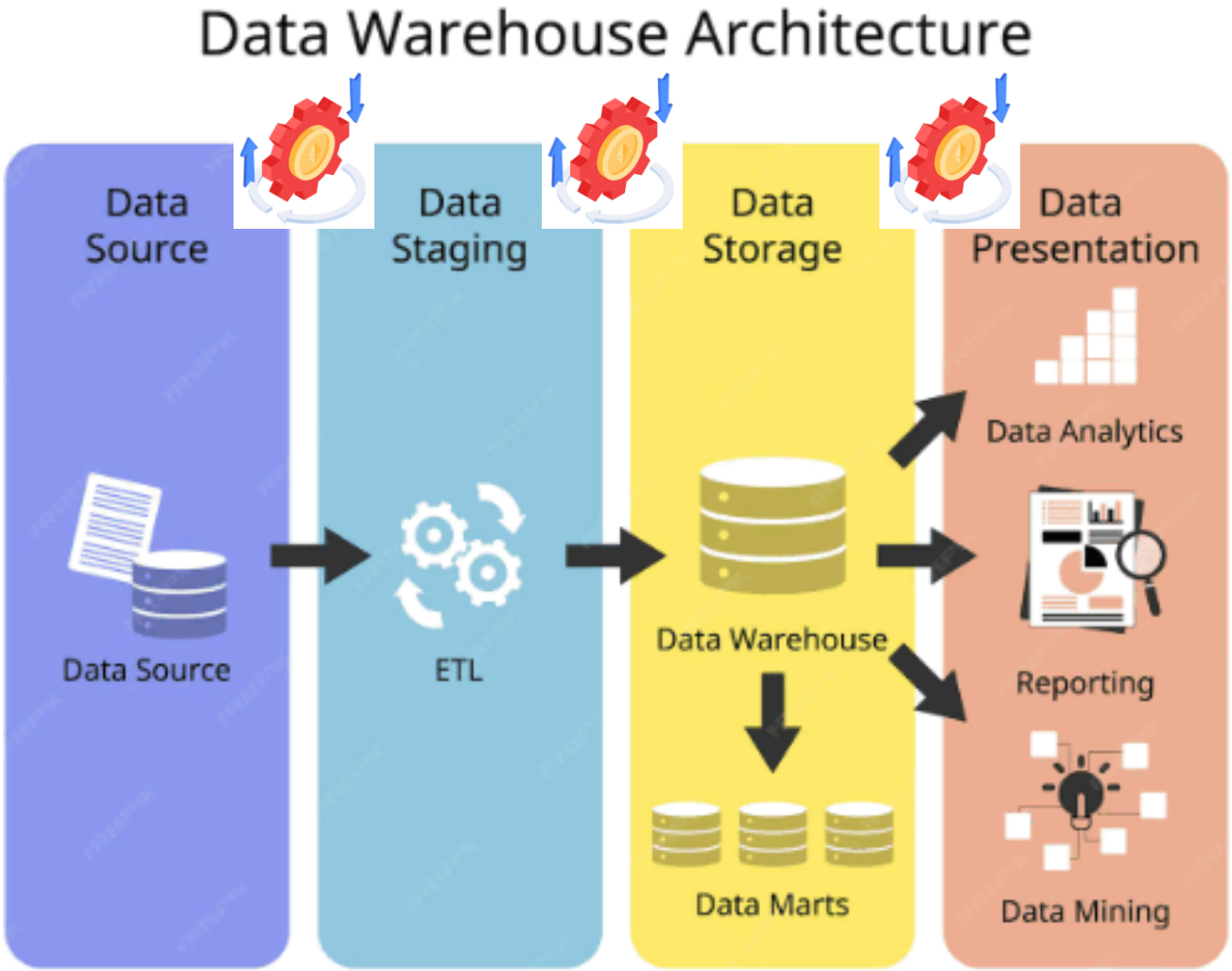
Làm giàu dữ liệu: Không chỉ chuẩn hoá, quá trình chuyển đổi còn có thể bổ sung thêm thuộc tính (derived attributes), tổng hợp dữ liệu (aggregation), hoặc phân loại dữ liệu (categorization).

Tối ưu cho phân tích: Dữ liệu sau khi chuyển đổi thường được thiết kế theo mô hình Star Schema hoặc Snowflake Schema, giúp dễ dàng cho OLAP và Data Mining.

Đảm bảo khả năng mở rộng: Chuyển đổi dữ liệu tốt sẽ giúp kho dữ liệu dễ dàng tích hợp thêm các nguồn dữ liệu mới trong tương lai.

5. Vấn đề về chuyển đổi dữ liệu

5.2 Vai trò của chuyển đổi dữ liệu trong xây dựng kho dữ liệu



5. Vấn đề về chuyển đổi dữ liệu

5.3 Mối quan hệ giữa chuyển đổi dữ liệu và chất lượng dữ liệu

Tính chính xác (**Accuracy**): Quá trình chuyển đổi phát hiện và xử lý lỗi (ví dụ: giá trị NULL, sai kiểu dữ liệu).

Tính nhất quán (**Consistency**): Chuẩn hoá định dạng và mã hoá dữ liệu (ví dụ: chuẩn hoá ngày tháng, đơn vị đo lường, mã quốc gia).

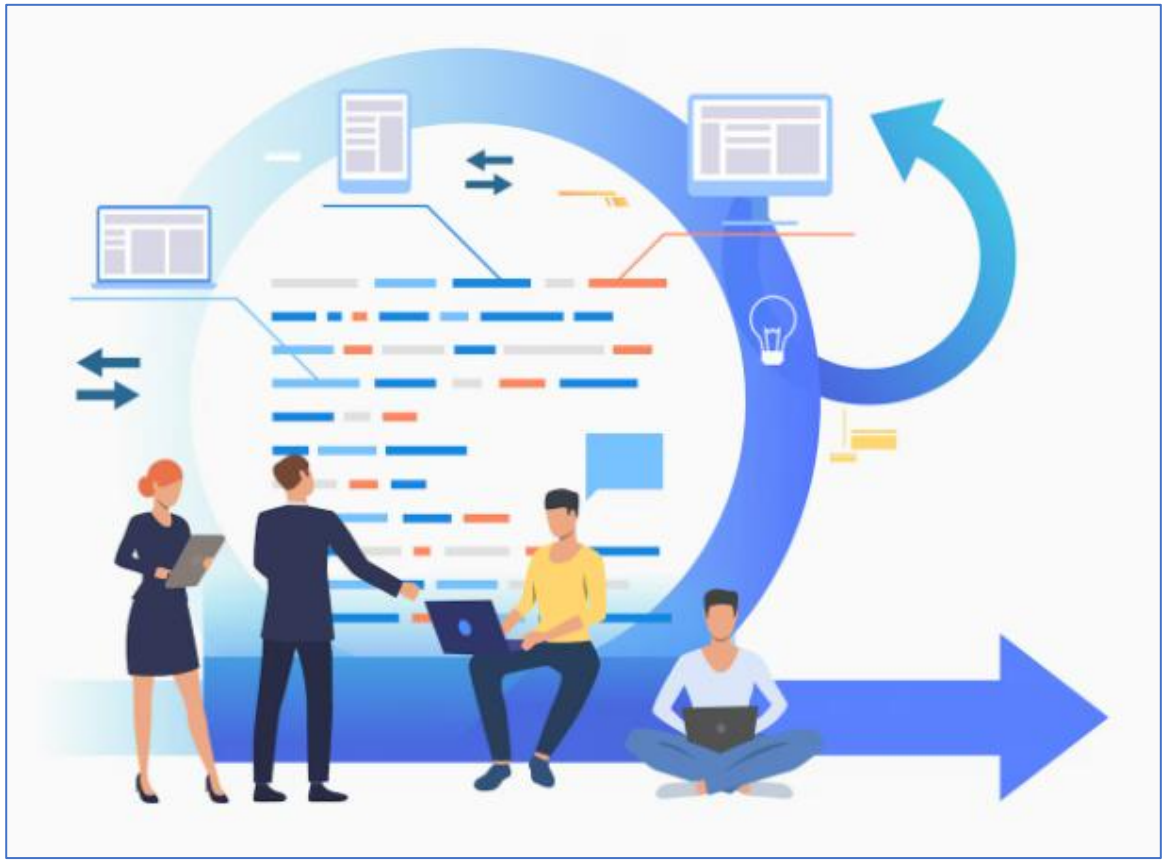
Tính đầy đủ (**Completeness**): Bổ sung dữ liệu thiếu (imputation), hoặc gắn nhãn rõ ràng các giá trị không xác định.

Tính kịp thời (**Timeliness**): Quá trình chuyển đổi được thiết kế để đảm bảo dữ liệu được cập nhật định kỳ (batch) hoặc theo thời gian thực (real-time).

Tính tin cậy (**Reliability**): Dữ liệu sau khi chuyển đổi đảm bảo phản ánh đúng thực tế, tạo niềm tin cho người dùng trong việc khai thác DSS (Decision Support Systems).

5. Vấn đề về chuyển đổi dữ liệu

Chuyển đổi dữ liệu trong ETL là bước **"cầu nối"** giữa dữ liệu thô (raw data) và dữ liệu phân tích (analytic-ready data). Đây là yếu tố quyết định đến **giá trị sử dụng và chất lượng** của kho dữ liệu, đồng thời là nền tảng để các công cụ OLAP, Data Mining và DSS hoạt động hiệu quả.



5. Vấn đề về chuyển đổi dữ liệu

5.4 Các loại chuyển đổi dữ liệu cơ bản

5.4.1 Chuyển đổi cú pháp (Syntactic Transformation)

Khái niệm: thay đổi cấu trúc và định dạng dữ liệu mà không thay đổi ý nghĩa.

Ví dụ:

Ngày tháng: 2025-09-25 → 25/09/2025.

Số điện thoại: +84981234567 → 0981234567.

Chuyển kiểu dữ liệu: từ VARCHAR sang DATE.

Ý nghĩa: Giúp dữ liệu thống nhất về định dạng, thuận lợi cho việc lưu trữ và xử lý.

5. Vấn đề về chuyển đổi dữ liệu

5.4 Các loại chuyển đổi dữ liệu cơ bản

5.4.2 Chuyển đổi ngữ nghĩa (Semantic Transformation)

Khái niệm: chuẩn hóa ý nghĩa và giá trị của dữ liệu.

Ví dụ:

Tiền tệ: 100 USD → 2,400,000 VND (theo tỷ giá).

Giới tính: “Nam/Nữ” ↔ “M/F” ↔ 0/1.

Mã quốc gia: “Vietnam”, “VN”, “VNM” → thống nhất thành “VN”.

Ý nghĩa: Loại bỏ mâu thuẫn ngữ nghĩa giữa các nguồn dữ liệu, nâng cao tính nhất quán.

5. Vấn đề về chuyển đổi dữ liệu

5.4 Các loại chuyển đổi dữ liệu cơ bản

5.4.3 Tích hợp dữ liệu (Data Integration)

Khái niệm: kết hợp dữ liệu từ nhiều hệ thống nguồn khác nhau thành một kho dữ liệu thống nhất.

Thách thức:

Trùng lặp dữ liệu (duplication).

Không đồng nhất khóa chính (ví dụ: khách hàng ID=123 ở CRM và KH00123 ở ERP).

Giải pháp:

Áp dụng Entity Resolution để nhận diện cùng một thực thể.

Sử dụng Surrogate Key trong bảng Dimension để gắn mã định danh duy nhất.

Ý nghĩa: Đảm bảo kho dữ liệu không bị dư thừa và có toàn cảnh (single version of truth).

5. Vấn đề về chuyển đổi dữ liệu

5.4 Các loại chuyển đổi dữ liệu cơ bản

5.4.4 Tổng hợp và rút gọn dữ liệu (Aggregation & Summarization)

Khái niệm: biến đổi dữ liệu chi tiết (transactional data) thành dữ liệu tổng hợp (summary data).

Ví dụ:

Dữ liệu giao dịch bán lẻ (mỗi hóa đơn, mỗi sản phẩm) → Doanh thu theo ngày/tháng/quý/năm.

Dữ liệu điểm thi sinh viên (theo từng môn) → GPA theo học kỳ/năm học.

Ý nghĩa:

Hỗ trợ các mức granularity khác nhau cho OLAP (drill-down, roll-up).

Giảm dung lượng lưu trữ, tăng hiệu năng phân tích.

5. Vấn đề về chuyển đổi dữ liệu

5.5 Thách thức trong chuyển đổi dữ liệu

Khối lượng dữ liệu rất lớn → cần tối ưu hiệu năng.

Dữ liệu từ nhiều nguồn không đồng nhất (định dạng, chuẩn mã, đơn vị đo).

Dữ liệu thường xuyên thay đổi, cập nhật.

Rủi ro sai sót nếu quy trình ETL không được kiểm soát chặt chẽ.

5. Vấn đề về chuyển đổi dữ liệu

5.6 Công cụ và phương pháp hỗ trợ

ETL Tools: Informatica, Talend, **Pentaho**, Microsoft SSIS.

Ngôn ngữ lập trình: SQL, Python (pandas), Apache Spark.

Kiến trúc hiện đại: Data Lakehouse, Snowflake, BigQuery.

Nhận xét:

Chuyển đổi dữ liệu là trái tim của ETL và đóng vai trò then chốt trong việc xây dựng kho dữ liệu.

Quyết định trực tiếp đến chất lượng, độ tin cậy và tính hữu dụng của dữ liệu.

Là nền tảng để triển khai OLAP, Data Mining, DSS một cách chính xác và hiệu quả.

5. Vấn đề về chuyển đổi dữ liệu

5.7 Minh họa chuyển đổi dữ liệu cơ bản

Ví dụ minh họa

Thao tác đã thực hiện chuyển đổi:
Chuyển đổi cú pháp: chuẩn hoá ngày tháng, số điện thoại.
Chuyển đổi ngữ nghĩa: đổi "Nam" → "M", chuyển USD → VND.
Tích hợp: gán Surrogate Key (CustKey).
Tổng hợp: có thể thêm cột doanh thu theo tháng/năm.

Trước chuyển đổi:

CustomerID	Gender	Phone	Date	Amount	Currency
001	Nam	+84981234567	2025-09-25	100	USD
KH002	F	0982222333	25/09/2025	2000000	VND

Sau chuyển đổi:

CustKey	Gender	Phone	Date	Amount_VND
1001	M	0981234567	25/09/2025	2,400,000
1002	F	0982222333	25/09/2025	2,000,000

5. Vấn đề về chuyển đổi dữ liệu

5.8 Các thách thức chính

5.8.1 Độ phức tạp cao – nhiều nguồn dữ liệu không đồng nhất

Nguyên nhân:

Mỗi hệ thống nguồn có định dạng, mã hóa, cấu trúc dữ liệu riêng (CRM, ERP, file Excel, API,...).

Ví dụ: hệ thống bán hàng dùng “KH001”, hệ thống chăm sóc khách hàng dùng “CUST-1”.

Hậu quả: khó tích hợp và đối chiếu dữ liệu giữa các nguồn.

Giải pháp:

Xây dựng chuẩn chung (chuẩn mã khách hàng, chuẩn ngày tháng).

Sử dụng Surrogate Key để gắn định danh duy nhất cho thực thể.

5. Vấn đề về chuyển đổi dữ liệu

5.8 Các thách thức chính

5.8.2 Khối lượng dữ liệu lớn – cần hiệu năng và tối ưu

Nguyên nhân:

Dữ liệu phát sinh liên tục từ giao dịch, cảm biến, log website.

Ví dụ: một siêu thị có hàng triệu hóa đơn mỗi ngày.

Hậu quả:

Thời gian xử lý ETL lâu.

Hệ thống dễ quá tải.

Giải pháp:

Tối ưu truy vấn SQL, chỉ lấy trường cần thiết.

Dùng xử lý song song (parallel processing), Apache Spark, hoặc công nghệ cloud (Snowflake, BigQuery).

5. Vấn đề về chuyển đổi dữ liệu

5.8 Các thách thức chính

5.8.3 Độ tin cậy – tránh sai sót khi chuyển đổi

Nguyên nhân:

- Sai kiểu dữ liệu (string ↔ int).
- Thiếu dữ liệu (NULL).
- Trùng lặp bản ghi.

Ví dụ: “Giới tính” có giá trị “Nam/Nữ”, “M/F”, “1/0” → nếu không chuẩn hóa sẽ gây sai kết quả phân tích.

Giải pháp:

- Áp dụng quy trình Data Quality: kiểm tra, làm sạch, chuẩn hóa.
- Thêm bước Validation sau khi chuyển đổi.

5. Vấn đề về chuyển đổi dữ liệu

5.8 Các thách thức chính

5.8.4 Thay đổi liên tục – hệ thống nguồn cập nhật thường xuyên

Nguyên nhân:

- Nghiệp vụ thay đổi (thêm thuộc tính mới, đổi mã sản phẩm).
- Hệ thống nguồn được nâng cấp hoặc thay đổi cấu trúc.

Hậu quả:

- Dữ liệu trong kho nhanh chóng lỗi thời.
- Mất đồng bộ giữa OLTP và Data Warehouse.

Giải pháp:

- Thiết kế ETL linh hoạt, dễ mở rộng.
- Cập nhật định kỳ (batch) hoặc real-time streaming (Kafka, CDC – Change Data Capture).

5. Vấn đề về chuyển đổi dữ liệu

5.8 Các thách thức chính

5.8.5 Thay đổi liên tục – hệ thống nguồn cập nhật thường xuyên

Nguyên nhân:

Nghiệp vụ thay đổi (thêm thuộc tính mới, đổi mã sản phẩm).

Hệ thống nguồn được nâng cấp hoặc thay đổi cấu trúc.

Hậu quả:

Dữ liệu trong kho nhanh chóng lỗi thời.

Mất đồng bộ giữa OLTP và Data Warehouse.

Giải pháp:

Thiết kế ETL linh hoạt, dễ mở rộng.

Cập nhật định kỳ (batch) hoặc real-time streaming (Kafka, CDC – Change Data Capture).

5. Vấn đề về chuyển đổi dữ liệu

Vai trò của Chuyển đổi dữ liệu

- Chuyển đổi dữ liệu là “trái tim” của quá trình ETL trong kho dữ liệu.
- Quyết định trực tiếp đến:
  - Chất lượng dữ liệu
  - Độ tin cậy
  - Khả năng sử dụng của Data Warehouse
- Là bước nền tảng cho các kỹ thuật phân tích nâng cao:
  - OLAP
  - Data Mining
  - DSS

6. Vấn đề về dữ liệu dẫn xuất

Dẫn nhập

Dữ liệu dẫn xuất (Derived Data): dữ liệu được tính toán hoặc tổng hợp từ dữ liệu gốc.

Khác biệt

Dữ liệu gốc (Raw Data): dữ liệu ban đầu, chưa xử lý.

Dữ liệu dẫn xuất (Derived Data): kết quả xử lý, tính toán từ dữ liệu gốc.

Vai trò

Giúp phân tích nhanh, trực quan.

Hỗ trợ ra quyết định chính xác, kịp thời.

6. Vấn đề về dữ liệu dẫn xuất

6.1 Các dạng dữ liệu dẫn xuất

Dữ liệu tính toán lại (Computed Data)

Sinh ra từ phép tính: tổng, trung bình, tỷ lệ, tỷ trọng…

Ví dụ: Doanh thu bình quân/khách hàng, phần trăm tăng trưởng doanh số.

Dữ liệu tổng hợp (Aggregated Data)

Dữ liệu chi tiết được gom nhóm, tóm lược.

Ví dụ: Doanh thu ngày → doanh thu tháng → doanh thu quý.

6. Vấn đề về dữ liệu dẫn xuất

6.1 Các dạng dữ liệu dẫn xuất

Dữ liệu phân loại (Categorized Data)

Phân nhóm hoặc phân cụm dữ liệu.

Ví dụ: Khách hàng hạng A/B/C theo giá trị mua hàng; xếp loại học lực sinh viên.

Dữ liệu chuẩn hóa (Standardized Data)

Chuyển đổi về thang đo, đơn vị, chuẩn chung.

Ví dụ: USD → VND; nhiệt độ °F → °C; mã quốc gia “Vietnam/VN/VNM” → “VN”.

6. Vấn đề về dữ liệu dẫn xuất

6.2 Ý nghĩa của dữ liệu dẫn xuất

Nâng cao hiệu quả phân tích: Dữ liệu dẫn xuất giúp rút ngắn thời gian truy vấn, giảm độ phức tạp khi xử lý khối lượng dữ liệu lớn.

Hỗ trợ ra quyết định nhanh chóng: Các chỉ số được tính toán sẵn (doanh thu, lợi nhuận, tăng trưởng, GPA…) giúp nhà quản trị có ngay thông tin để quyết định.

Giảm tải cho hệ thống: Hạn chế việc lặp lại các phép tính phức tạp nhiều lần, tránh tiêu tốn tài nguyên.

Cung cấp thông tin trực quan: Các chỉ số KPI, thước đo hiệu suất được trình bày rõ ràng, dễ hiểu, phục vụ phân tích và báo cáo quản trị.

6. Vấn đề về dữ liệu dẫn xuất

6.3 Thách thức trong xử lý dữ liệu dẫn xuất

Độ chính xác: nếu công thức sai → kết quả sai.

Đồng bộ: dữ liệu dẫn xuất cần cập nhật khi dữ liệu gốc thay đổi.

Dung lượng: dữ liệu dẫn xuất lưu trữ nhiều mức tổng hợp khác nhau → tăng chi phí.

Trùng lặp logic: nhiều nguồn tạo cùng một chỉ số với cách tính khác nhau.

6. Vấn đề về dữ liệu dẫn xuất

6.4 Công cụ và phương pháp tạo dữ liệu dẫn xuất

ETL Tools: tính toán trước khi load.

OLAP Cubes: sinh dữ liệu tổng hợp theo nhiều chiều.

SQL/Python: tạo các trường dẫn xuất (calculated fields).

Dashboard/BI Tools: tính toán động (on-the-fly).

Ý nghĩa của dữ liệu dẫn xuất

Dữ liệu dẫn xuất là một phần tất yếu của kho dữ liệu.

Cần cân bằng giữa tính sẵn sàng cho phân tích và chi phí lưu trữ – duy trì.

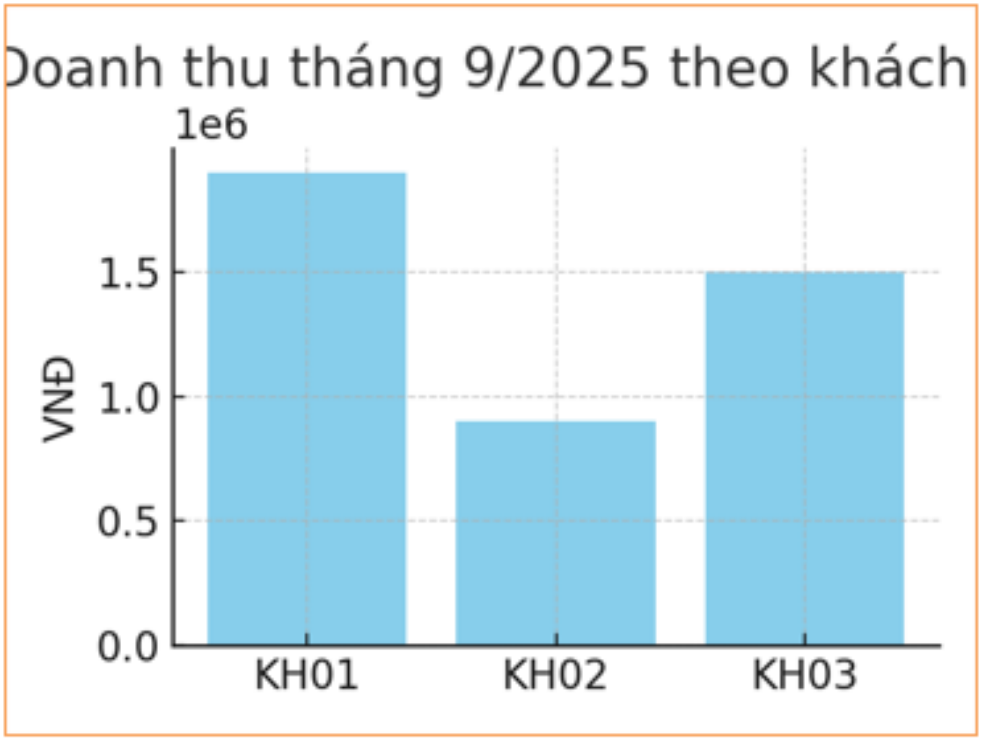
Quản lý tốt dữ liệu dẫn xuất giúp nâng cao hiệu quả DSS và BI.

6. Vấn đề về dữ liệu dẫn xuất

6.5 Ví dụ minh họa: Dữ liệu dẫn xuất (Retail)

Bảng dữ liệu gốc (Raw Data) – giao dịch bán hàng theo ngày; với Bảng dữ liệu dẫn xuất (Derived Data)

Ngày	Mã KH	Doanh thu (VNĐ)	Chi phí (VNĐ)
01/09/2025	KH01	1,200,000	800,000
01/09/2025	KH02	900,000	500,000
02/09/2025	KH01	700,000	400,000
02/09/2025	KH03	1,500,000	1,000,000



7. Vấn đề về siêu dữ liệu

7.1 Khái niệm siêu dữ liệu (Metadata)

Định nghĩa:

Metadata là “dữ liệu về dữ liệu”, tức là thông tin mô tả về cấu trúc, nội dung, nguồn gốc, và cách sử dụng dữ liệu trong hệ thống.

Ví dụ: trong bảng Fact_Sales, metadata sẽ cho biết:

SalesAmount: kiểu dữ liệu Decimal(10,2), ý nghĩa “Tổng số tiền khách hàng đã thanh toán, sau khi trừ giảm giá và cộng thuế”.

DateKey: khóa ngoại liên kết đến bảng Dim_Date.

Mục đích: Metadata đóng vai trò như bản đồ chỉ đường cho người dùng và nhà phân tích khi truy cập vào kho dữ liệu, giúp họ hiểu dữ liệu đến từ đâu, có ý nghĩa gì, và được xử lý như thế nào.

7. Vấn đề về siêu dữ liệu

7.2 Sự khác biệt vai trò siêu dữ liệu giữa hai môi trường

a) Trong môi trường vận hành (Operational Environment)

Metadata chủ yếu phục vụ kỹ thuật hệ thống, gồm:

- Cấu trúc bảng, tên cột, kiểu dữ liệu.
- Ràng buộc toàn vẹn, quan hệ giữa các bảng.
- Thông tin lưu trữ vật lý (file, chỉ mục, log).

Người dùng cuối (end-users) hầu như không quan tâm nhiều đến metadata.

Mục tiêu: đảm bảo hệ thống chạy ổn định, chính xác, đúng logic nghiệp vụ.

7. Vấn đề về siêu dữ liệu

7.2 Sự khác biệt vai trò siêu dữ liệu giữa hai môi trường

b) Trong môi trường kho dữ liệu (Data Warehouse Environment)

Metadata đóng vai trò trung tâm cho mọi hoạt động phân tích và DSS.

Bao gồm cả hai loại:

1. Metadata kỹ thuật (Technical Metadata):

Nguồn gốc dữ liệu (data lineage): dữ liệu đến từ hệ thống nào, trường nào.

Quá trình ETL: dữ liệu đã được biến đổi như thế nào.

Cấu trúc của schema (star/snowflake).

2. Metadata nghiệp vụ (Business Metadata):

Ý nghĩa kinh doanh của dữ liệu.

Định nghĩa chuẩn về các chỉ số (KPI, doanh thu, lợi nhuận, chi phí).

Quy tắc phân loại (ví dụ: khách hàng VIP là gì).

Người dùng cuối (DSS analyst, nhà quản trị, nhân viên kinh doanh) dựa vào metadata để hiểu và khai thác dữ liệu.

Mục tiêu: đảm bảo tính minh bạch, khả năng truy xuất, và sự tin cậy của thông tin trong DSS.

7. Vấn đề về siêu dữ liệu

So sánh môi trường triển khai:

Ở môi trường vận hành nghiệp vụ chức năng, metadata là “tài liệu kỹ thuật” dành cho DBA và developer.

Ở môi trường kho dữ liệu, metadata còn là “từ điển dữ liệu” dành cho nhà phân tích và quản trị, giúp họ hiểu và dùng dữ liệu trong ra quyết định.

Metadata trong kho dữ liệu không chỉ dừng ở mức mô tả kỹ thuật, mà còn là **cầu nối giữa dữ liệu và ngôn ngữ kinh doanh** – điều làm nên sự khác biệt cốt lõi so với metadata trong hệ thống vận hành.

7. Vấn đề về siêu dữ liệu

7.3. Vai trò và ý nghĩa của Metadata trong kho dữ liệu

7.3.1 Hỗ trợ người dùng phi kỹ thuật (Business Users, DSS Analyst)

Vấn đề: Nhiều nhà phân tích, nhà quản trị không quen với khái niệm bảng, khóa ngoại, chỉ mục…

Giải pháp: Metadata đóng vai trò như từ điển dữ liệu giúp họ hiểu ý nghĩa:

SalesAmount: “Tổng số tiền khách hàng chi trả sau giảm giá, bao gồm thuế”.

CustomerType: “Loại khách hàng (VIP, Regular, New) theo quy tắc phân loại doanh nghiệp”.

Ý nghĩa: Nhờ metadata, người dùng không cần biết cấu trúc kỹ thuật vẫn có thể truy cập, diễn giải và sử dụng dữ liệu chính xác.

7. Vấn đề về siêu dữ liệu

7.3.2 Lưu vết quá trình ETL và biến đổi dữ liệu (Data Lineage)

Metadata ghi lại toàn bộ hành trình của dữ liệu từ nguồn gốc → quá trình biến đổi → kho dữ liệu.

Ví dụ: Trường SalesAmount trong Fact_Sales được lấy từ OrderValue của hệ thống POS, sau khi chuyển đổi từ USD sang VND.

Ý nghĩa: Người phân tích có thể truy xuất ngược để kiểm chứng tính đúng đắn của dữ liệu, nâng cao tính minh bạch.

7. Vấn đề về siêu dữ liệu

7.3.3 Quản lý thay đổi theo thời gian (Historical Metadata)

Kho dữ liệu thường lưu trữ dữ liệu nhiều năm (5–10 năm). Trong thời gian này:

Cấu trúc bảng nguồn có thể thay đổi.

Định nghĩa nghiệp vụ có thể điều chỉnh (ví dụ: cách xác định khách hàng VIP).

Metadata giúp theo dõi phiên bản và sự tiến hóa của dữ liệu theo thời gian.

Ví dụ: Đảm bảo báo cáo năm 2015 và 2025 được hiểu đúng trong bối cảnh của từng thời kỳ.

7. Vấn đề về siêu dữ liệu

7. 3.4 Đảm bảo tính nhất quán và khả năng tra cứu

Metadata cung cấp định nghĩa thống nhất cho các chỉ số và khái niệm trong toàn tổ chức.

Ví dụ: “Doanh thu” được chuẩn hóa → tất cả phòng ban khi báo cáo sẽ hiểu giống nhau.

Ý nghĩa: Tránh trường hợp mỗi phòng ban tự định nghĩa khác nhau, gây mâu thuẫn số liệu.

7. Vấn đề về siêu dữ liệu

7. 3.5 Nâng cao khả năng quản trị và an toàn dữ liệu

Metadata lưu trữ thông tin về phân quyền truy cập, người dùng nào có thể xem dữ liệu nào.

Ngoài ra còn quản lý vòng đời dữ liệu (data lifecycle).

Ý nghĩa: Hỗ trợ kiểm soát bảo mật, tuân thủ quy định (ví dụ: GDPR, ISO 27001).

Kết luận :

Metadata không chỉ là “thông tin phụ” mà chính là xương sống của kho dữ liệu. Nó giúp:

Người dùng hiểu dữ liệu.

Hệ thống quản lý dữ liệu.

Doanh nghiệp tin cậy khi dùng dữ liệu để phân tích và ra quyết định.

7. Vấn đề về siêu dữ liệu

7.4. Thách thức trong quản lý siêu dữ liệu

7.4.1 Metadata phân tán và không đồng nhất

Thực tế:

Mỗi công cụ ETL (Informatica, Talend, SSIS), công cụ BI (Power BI, Tableau) hay hệ quản trị CSDL (Oracle, SQL Server, Snowflake) đều sinh ra metadata riêng.

Metadata này thường không tương thích và khó tích hợp với nhau.

Hậu quả:

Gây ra tình trạng metadata phân tán (metadata silos).

Người dùng khó tìm "phiên bản duy nhất" của dữ liệu (single source of truth).

Ví dụ: Một chỉ số "Lợi nhuận" được định nghĩa khác nhau trong công cụ kế toán và công cụ BI → kết quả báo cáo mâu thuẫn.

7. Vấn đề về siêu dữ liệu

7. 4.2 Khối lượng và độ phức tạp lớn

Thực tế:

Kho dữ liệu tích hợp từ nhiều hệ thống nguồn, kéo theo hàng trăm nghìn bảng, cột và quy tắc biến đổi.

Ngoài dữ liệu có cấu trúc, còn có dữ liệu phi cấu trúc (văn bản, hình ảnh, log).

Hậu quả:

Metadata trở nên đồ sộ và khó quản lý.

Việc tra cứu hoặc đồng bộ metadata chậm, kém hiệu quả.

Ví dụ: Metadata cho một data lake có thể lớn hơn nhiều so với dữ liệu gốc (do phải lưu thông tin mô tả, chỉ mục, phiên bản).

7. Vấn đề về siêu dữ liệu

7. 4.3 Thiếu chuẩn hóa và tích hợp

Thực tế:

Không phải tổ chức nào cũng áp dụng chuẩn metadata (như CWM – Common Warehouse Metamodel, hay ISO 11179).

Nhiều nơi lưu trữ metadata dưới dạng thủ công (Excel, Word) hoặc phụ thuộc công cụ riêng lẻ.

Hậu quả:

Metadata khó tái sử dụng khi thay đổi công nghệ.

Dẫn đến chi phí cao khi tích hợp hoặc di chuyển hệ thống.

Ví dụ: Một công ty dùng SSIS lưu metadata trong SQL Server, sau này muốn chuyển sang Snowflake thì gặp khó khăn vì metadata không thể chuyển trực tiếp.

7. Vấn đề về siêu dữ liệu

7. 4.4 Cập nhật và đồng bộ theo thời gian

Thực tế:

Cấu trúc dữ liệu nguồn thay đổi liên tục (thêm cột, đổi định nghĩa).

Nếu metadata không được cập nhật kịp, người dùng sẽ dựa vào thông tin lỗi thời.

Hậu quả:

Mất niềm tin vào dữ liệu.

DSS có thể đưa ra quyết định sai lệch.

Ví dụ: Nếu cột “Doanh thu” ban đầu bao gồm VAT nhưng sau này thay đổi chỉ còn giá trị chưa VAT, metadata không cập nhật thì báo cáo sẽ sai.

7. Vấn đề về siêu dữ liệu

7. 4.5 Quản lý vòng đời và quyền truy cập

Thực tế:

Metadata cũng có vòng đời: được tạo ra, thay đổi, và cuối cùng không còn phù hợp.

Ngoài ra, metadata chứa thông tin nhạy cảm (nguồn dữ liệu, quyền truy cập).

Hậu quả:

Nếu không quản trị tốt, metadata trở thành “rác dữ liệu” (data swamp).

Rủi ro bảo mật khi metadata tiết lộ thông tin nhạy cảm.

Ví dụ: Metadata ghi lại mapping giữa CSDL kế toán và kho dữ liệu → nếu lộ ra ngoài, hacker có thể hiểu toàn bộ cấu trúc dữ liệu tài chính.

7. Vấn đề về siêu dữ liệu

Quản lý metadata là một bài toán khó và phức tạp. Nếu không có chiến lược và công cụ phù hợp, metadata sẽ mất đi vai trò "kim chỉ nam" và trở thành gánh nặng cho kho dữ liệu.

7. Vấn đề về siêu dữ liệu

Ví dụ minh họa: Metadata trong kho dữ liệu bán lẻ

1. Metadata mô tả bảng Fact_Sales (Technical + Business Metadata)

Tên bảng: Fact_Sales

Nguồn dữ liệu: Hệ thống POS (Point-of-Sale).

Trường dữ liệu:

Ý nghĩa: Bảng **Fact_Sales** cho phép phân tích doanh thu theo thời gian, sản phẩm và khách hàng.

Tên trường	Kiểu dữ liệu	Khóa	Business meaning (Ý nghĩa nghiệp vụ)
DateKey	INT	FK	Khóa ngoại liên kết đến Dim_Date để xác định ngày bán hàng
ProductKey	INT	FK	Khóa ngoại liên kết đến Dim_Product để xác định sản phẩm
CustomerKey	INT	FK	Khóa ngoại liên kết đến Dim_Customer để xác định khách hàng
SalesAmount	DECIMAL(12,2)		Tổng số tiền khách hàng chi trả sau khi trừ giảm giá và cộng thuế

7. Vấn đề về siêu dữ liệu

Ví dụ minh họa: Metadata trong kho dữ liệu bán lẻ

2. Metadata lưu vết quá trình ETL (Data Lineage)

Nguồn gốc trường SalesAmount:

Lấy từ cột OrderValue của hệ thống OLTP (tblOrders).

Được quy đổi từ USD sang VND với tỷ giá cố định 1 USD = 24,000 VND.

Công thức **ETL**: SalesAmount = OrderValue * ExchangeRate

Mapping chi tiết:

Ý nghĩa: Metadata ETL giúp người phân tích biết chính xác SalesAmount trong kho dữ liệu được hình thành từ cột nào, đã qua những phép biến đổi gì.

Trường OLTP (Nguồn)	Quy tắc ETL (Transform)	Trường DW (Đích)
tblOrders.OrderID	Copy trực tiếp → Fact_Sales.SalesID	SalesID
tblOrders.OrderValue (USD)	Nhân với ExchangeRate = 24000	Fact_Sales.SalesAmount (VND)
tblOrders.OrderDate	Convert format YYYY-MM-DD → DateKey	Fact_Sales.DateKey

7. Vấn đề về siêu dữ liệu

Ví dụ minh họa: Metadata trong kho dữ liệu bán lẻ

3. Metadata quản lý & truy cập (Access Metadata)

Phân quyền:

Nhân viên bán hàng: chỉ xem dữ liệu theo khu vực quản lý.

Quản lý chi nhánh: xem tất cả khách hàng trong chi nhánh.

Ban giám đốc: xem toàn bộ dữ liệu công ty.

Ý nghĩa: Metadata về phân quyền đảm bảo dữ liệu được sử dụng đúng đối tượng, đúng mục đích, vừa hỗ trợ phân tích vừa đảm bảo bảo mật.

7. Vấn đề về siêu dữ liệu

NHận xét:

Metadata không chỉ mô tả cấu trúc bảng, mà còn cho biết **ý nghĩa kinh doanh, nguồn gốc, phép biến đổi ETL và quyền truy cập.**

Đây là cơ sở để **DSS analyst** và **nhà quản trị** hiểu và tin tưởng khi sử dụng dữ liệu để ra quyết định.

7. Vấn đề về siêu dữ liệu

Kết luận – Siêu dữ liệu trong Kho dữ liệu

Siêu dữ liệu (Metadata) = “Dữ liệu về dữ liệu”

Giữ vai trò hạ tầng cốt lõi trong kho dữ liệu.

Ý nghĩa

Đảm bảo chất lượng và tính minh bạch của dữ liệu.

Hỗ trợ khả năng phân tích và ra quyết định.

Tạo niềm tin cho người dùng khi khai thác DSS.

Yêu cầu

Cần có hệ thống quản trị metadata mạnh mẽ để đảm bảo hiệu quả lâu dài.

8. Các vấn đề khác về kho dữ liệu

Các vấn đề thường gặp khác trong kho dữ liệu

8.1 Vấn đề về chất lượng dữ liệu (Data Quality)

Chất lượng dữ liệu (Data Quality) là mức độ dữ liệu phản ánh chính xác thực tế và đáp ứng đúng yêu cầu của người dùng cuối.

Trong kho dữ liệu, chất lượng dữ liệu là yếu tố sống còn vì mọi phân tích, báo cáo và ra quyết định đều dựa vào dữ liệu được nạp vào.

8. Các vấn đề khác về kho dữ liệu

Các vấn đề thường gặp về chất lượng dữ liệu

a) Dữ liệu thiếu (Missing Data)

Một số thuộc tính không có giá trị (NULL) → gây khó khăn khi tổng hợp hoặc tính toán.

Ví dụ: khách hàng không có ngày sinh → khó phân tích nhóm tuổi.

b) Dữ liệu trùng lặp (Duplicate Data)

Cùng một thực thể xuất hiện nhiều lần với ID hoặc biểu diễn khác nhau.

Ví dụ: “KH001 – Nguyễn Văn A” và “KH-0001 – Nguyen Van A” là cùng một khách hàng.

8. Các vấn đề khác về kho dữ liệu

c) Dữ liệu sai lệch hoặc không nhất quán (Inconsistent Data)

Các hệ thống nguồn định nghĩa khác nhau cho cùng một thuộc tính.

Ví dụ: doanh thu “Sales” trong hệ thống A bao gồm VAT, nhưng trong hệ thống B lại không.

d) Dữ liệu lỗi thời (Outdated Data)

Thông tin không được cập nhật kịp thời, dẫn đến quyết định dựa trên dữ liệu cũ.

Ví dụ: địa chỉ khách hàng thay đổi nhưng hệ thống chưa cập nhật.

e) Dữ liệu không chuẩn hóa (Non-standardized Data)

Định dạng không thống nhất: ngày tháng, mã quốc gia, đơn vị tiền tệ.

Ví dụ: “Việt Nam”, “VN”, “VNM” cùng chỉ một quốc gia.

8. Các vấn đề khác về kho dữ liệu

Tác động đến DSS

Mất độ tin cậy: Người dùng không tin vào dữ liệu → không sử dụng DSS.

Quyết định sai lầm: Báo cáo dựa trên dữ liệu sai → dẫn đến chiến lược sai.

Tăng chi phí: Nhiều thời gian bị lãng phí cho việc làm sạch dữ liệu thủ công.

8. Các vấn đề khác về kho dữ liệu

Giải pháp nâng cao chất lượng dữ liệu

Xây dựng quy trình làm sạch dữ liệu (Data Cleansing): phát hiện và sửa lỗi, loại bỏ bản ghi trùng.

Áp dụng quy tắc chuẩn hóa (Standardization): định nghĩa thống nhất cho các thuộc tính quan trọng.

Quản trị dữ liệu (Data Governance): ban hành chính sách quản lý dữ liệu trong toàn tổ chức.

Tích hợp công cụ chất lượng dữ liệu (Data Quality Tools): Informatica Data Quality, Talend, Trifacta.

Cập nhật định kỳ: thiết lập cơ chế giám sát để phát hiện dữ liệu lỗi thời.

8. Các vấn đề khác về kho dữ liệu

Ví dụ

Trước khi làm sạch

CustomerID	Name	BirthDate	Country	Sales
KH001	Nguyễn Văn A	NULL	VN	10,000
001KH	Nguyen Van A	12/05/1990	Vietnam	10,000
KH002	Trần Thị B	05/08/1985	VNM	8,000

Sau khi chuẩn hóa:

CustKey	Name	BirthDate	Country	Sales
1001	Nguyễn Văn A	12/05/1990	VN	10,000
1002	Trần Thị B	05/08/1985	VN	8,000

Kết quả: dữ liệu **nhất quán, đầy đủ, tin cậy** → đảm bảo phân tích DSS chính xác.

8. Các vấn đề khác về kho dữ liệu

8.2 Vấn đề về bảo mật và quyền truy cập (Security & Access Control)

Bảo mật dữ liệu (Data Security): đảm bảo dữ liệu trong kho dữ liệu không bị truy cập, thay đổi hoặc lộ lọt trái phép.

Quyền truy cập (Access Control): cơ chế xác định ai được phép truy cập dữ liệu nào, theo vai trò, chức năng hoặc nhu cầu công việc.

8. Các vấn đề khác về kho dữ liệu

Các vấn đề bảo mật trong kho dữ liệu

a) Dữ liệu nhạy cảm

Kho dữ liệu thường chứa thông tin quan trọng: tài chính, lương, thông tin khách hàng, giao dịch.

Nếu lộ ra ngoài → gây thiệt hại nghiêm trọng cho tổ chức.

Ví dụ: số thẻ tín dụng, địa chỉ email khách hàng.

b) Truy cập không phù hợp

Người dùng không được phân quyền chính xác có thể xem dữ liệu vượt quá thẩm quyền.

Ví dụ: nhân viên bán hàng chỉ nên xem dữ liệu khách hàng trong khu vực của mình, không phải toàn quốc.

c) Tấn công an ninh mạng

Hacker có thể khai thác lỗ hổng trong hệ thống ETL, metadata repository hoặc BI để xâm nhập dữ liệu.

d) Chia sẻ dữ liệu bên ngoài

Khi dữ liệu được chia sẻ với đối tác, nhà cung cấp hoặc qua cloud, rủi ro mất an toàn tăng cao.

8. Các vấn đề khác về kho dữ liệu

Mô hình phân quyền người dùng

Administrator (Admin):

Quản lý toàn bộ hệ thống, phân quyền, bảo trì.

Data Analyst / Business Analyst:

Có quyền truy cập dữ liệu chi tiết hoặc tổng hợp, nhưng không được thay đổi cấu trúc.

End-user (Người dùng cuối, nhà quản lý):

Chỉ truy cập dữ liệu tổng hợp (KPI, báo cáo BI, dashboard).

External Partner (Đối tác ngoài):

Chỉ được truy cập một phần dữ liệu công khai hoặc được chia sẻ có kiểm soát.

8. Các vấn đề khác về kho dữ liệu

Ví dụ minh họa

Trước khi áp dụng phân quyền:

Nhân viên bán hàng A có thể truy cập toàn bộ doanh thu của công ty → rò rỉ thông tin.

Sau khi áp dụng phân quyền RBAC:

Nhân viên A chỉ được xem dữ liệu khách hàng trong khu vực TP.HCM.

Quản lý khu vực có quyền xem toàn bộ dữ liệu của chi nhánh miền Nam.

Ban giám đốc mới được xem dữ liệu toàn quốc.

Kết quả: dữ liệu an toàn hơn, người dùng chỉ truy cập đúng phạm vi cần thiết.

Kết luận:

Vấn đề bảo mật và quyền truy cập trong kho dữ liệu là thiết yếu, vì dữ liệu là tài sản chiến lược của doanh nghiệp.

Nếu không có cơ chế bảo vệ chặt chẽ, dữ liệu nhạy cảm có thể bị lạm dụng hoặc rò rỉ, làm mất uy tín tổ chức và gây thiệt hại lớn.

8. Các vấn đề khác về kho dữ liệu

8.3 Vấn đề về hiệu năng và tối ưu truy vấn (Performance Issues)

Hiệu năng trong Kho dữ liệu

Nguyên nhân gây ra vấn đề hiệu năng

Khối lượng dữ liệu rất lớn (hàng triệu → hàng tỷ bản ghi).

Câu truy vấn OLAP phức tạp (join nhiều bảng, tính toán tổng hợp).

Người dùng yêu cầu thời gian phản hồi nhanh cho báo cáo và dashboard.

Hệ quả nếu không tối ưu

Truy vấn chạy chậm → giảm trải nghiệm người dùng.

Khó khai thác DSS đúng thời điểm → ảnh hưởng ra quyết định.

8. Các vấn đề khác về kho dữ liệu

8.3 Vấn đề về hiệu năng và tối ưu truy vấn (Performance Issues)

Các kỹ thuật tối ưu truy vấn

1. Tối ưu chỉ mục (Indexing)

Tạo chỉ mục cho các cột thường dùng trong điều kiện WHERE, JOIN.

Giúp tăng tốc độ lọc và tìm kiếm.

2. Phân vùng dữ liệu (Partitioning)

Chia bảng lớn thành các partition nhỏ theo ngày, khu vực, hoặc loại sản phẩm.

Truy vấn chỉ cần đọc partition liên quan → giảm I/O.

3. Cân bằng giữa dữ liệu chi tiết & tổng hợp

Dữ liệu chi tiết: phục vụ phân tích sâu (drill-down).

Dữ liệu tổng hợp (summary tables, materialized views): phục vụ báo cáo nhanh.

Giải pháp: kết hợp cả hai mức dữ liệu để đáp ứng nhu cầu đa dạng.

8. Các vấn đề khác về kho dữ liệu

8.4 Vấn đề về tính mở rộng & duy trì (Scalability & Maintenance)

Nguyên nhân

Khối lượng dữ liệu tăng nhanh (theo năm, theo quý).

Yêu cầu thêm chiều phân tích mới (ví dụ: kênh bán hàng online).

Thách thức

Kho dữ liệu dễ bị “phình to”, truy vấn chậm.

Schema thay đổi (schema evolution): thêm cột, đổi định nghĩa KPI → ảnh hưởng báo cáo cũ.

Giải pháp

Thiết kế kiến trúc modular và mở rộng ngang (horizontal scaling).

Dùng công nghệ cloud DW (Snowflake, BigQuery, Redshift).

Có quy trình quản lý thay đổi schema → đảm bảo tương thích ngược.

8. Các vấn đề khác về kho dữ liệu

8.5 Vấn đề về tích hợp với hệ thống khác (Integration Issues)

Nguyên nhân

Kho dữ liệu phải kết nối nhiều nguồn: ERP, CRM, POS, IoT, Data Lake.

Ngoài ra cần liên kết với công cụ phân tích (OLAP, Data Mining, BI).

Thách thức

Dữ liệu không đồng bộ (real-time vs batch).

Xung đột định nghĩa chỉ số.

Tích hợp dữ liệu phi cấu trúc.

Giải pháp

Tích hợp với ODS (Operational Data Store) để làm vùng trung gian.

Sử dụng ETL/ELT pipelines hiện đại (Kafka, Spark, Airflow).

Áp dụng chuẩn hóa dữ liệu & metadata để giảm mâu thuẫn.

8. Các vấn đề khác về kho dữ liệu

8.6 Vấn đề về chi phí & quản trị (Cost & Governance)

Chi phí

- Đầu tư hạ tầng (server, storage, license phần mềm).
- Vận hành (ETL jobs, bảo trì, nhân sự chuyên môn).
- Cloud DW theo mô hình pay-as-you-go → dễ phát sinh chi phí lớn.

Quản trị dữ liệu (Data Governance)

- Xây dựng chính sách quản trị dữ liệu: định nghĩa chuẩn KPI, trách nhiệm sở hữu dữ liệu.
- Đảm bảo tuân thủ pháp lý (GDPR, HIPAA, ISO 27001).
- Quản lý vòng đời dữ liệu (data lifecycle): từ khi sinh ra → sử dụng → lưu trữ → hủy bỏ.

Ý nghĩa

- Kiểm soát chi phí hợp lý.
- Đảm bảo dữ liệu chính xác, an toàn, đáng tin cậy.

Tổng kết nội dung bài học

Tổng kết – Các vấn đề cơ bản trong Kho dữ liệu

Khái niệm & đặc trưng

Định nghĩa của Inmon: Subject-oriented, Integrated, Non-volatile, Time-variant.

Kho dữ liệu là nền tảng cho DSS & BI.

Kiến trúc kho dữ liệu

Star Schema, Snowflake Schema, Galaxy Schema.

Fact Table (số liệu định lượng) & Dimension Table (ngữ cảnh phân tích).

Đặc tính dữ liệu & chất lượng

Data Quality Dimensions: Accuracy, Completeness, Consistency, Validity, Timeliness, Uniqueness, Accessibility.

Khác biệt giữa đặc tính chất lượng dữ liệu và đặc trưng dữ liệu theo Inmon.

Tổng kết nội dung bài học

Tổng kết – Các vấn đề cơ bản trong Kho dữ liệu

Các vấn đề cốt lõi
- Độ mịn dữ liệu (Granularity).
- Chuyển đổi dữ liệu (Data Transformation – ETL).
- Dữ liệu dẫn xuất (Derived Data).
- Siêu dữ liệu (Metadata).

Các vấn đề khác
- Chất lượng dữ liệu.
- Bảo mật & quyền truy cập.
- Hiệu năng & tối ưu truy vấn.
- Tính mở rộng & duy trì.
- Tích hợp với hệ thống khác.
- Chi phí & quản trị dữ liệu.

Ý nghĩa: Kho dữ liệu là trung tâm của hệ DSS. Quản lý tốt các vấn đề trên giúp hệ thống ổn định, minh bạch, đáng tin cậy và bền vững.

Thảo luận 3 phút

Câu hỏi thảo luận nhóm

Độ mịn dữ liệu (Granularity):

Theo bạn, nên lưu dữ liệu bán hàng ở mức chi tiết giao dịch từng sản phẩm hay ở mức tổng hợp theo ngày? Ưu và nhược điểm của từng cách là gì?

Chuyển đổi dữ liệu (Data Transformation):

Hãy nêu một ví dụ trong thực tế (ngoài lớp học) mà dữ liệu cần phải chuyển đổi để đảm bảo tính nhất quán. Vì sao bước chuyển đổi này lại quan trọng?

Dữ liệu dẫn xuất (Derived Data):

Theo nhóm bạn, nên lưu trữ dữ liệu dẫn xuất (ví dụ: lợi nhuận, doanh thu bình quân) sẵn trong kho dữ liệu hay nên tính toán động khi cần phân tích? Tại sao?

Siêu dữ liệu (Metadata):

Hãy thảo luận: nếu kho dữ liệu không có metadata rõ ràng, điều gì sẽ xảy ra khi nhà quản trị hoặc nhà phân tích muốn sử dụng dữ liệu?

Các vấn đề khác:

Trong số các vấn đề: chất lượng dữ liệu, bảo mật, hiệu năng, mở rộng, tích hợp, chi phí, theo bạn vấn đề nào là thách thức lớn nhất khi triển khai kho dữ liệu ở doanh nghiệp Việt Nam? Vì sao?

THANK YOU FOR YOUR ATTENTION