Database

DATA AND INSIGHTS

Introduction

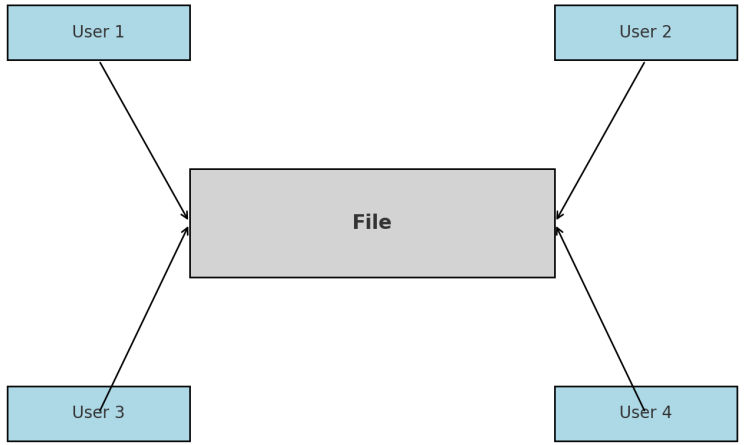
- Businesses generate vast amounts of data every day, from simple text messages and emails to complex files and transaction records.
- It contains crucial information about customers, sales, and operations.
- FreshDirect's Secret sauce: https://www.youtube.com/watch?v=sKvRhjWJkO4

Data Management Requirements

- Store large amounts of data in a single location (virtually from a single location, physically the data can be in multiple locations)
- Easily retrieve and manipulate data as needed
- Ensure data consistency and accuracy
- Implement security measures to protect sensitive data
- Analyze and report on data to inform business decisions

Filesystem is not Enough

- Concurrent access: Multiple users might try to access and update the same order file simultaneously, leading to conflicts and data corruption.
- Lack of transactional consistency: If an order is processed but the inventory update fails, the data becomes inconsistent, leading to incorrect inventory levels.
- Lack of durability: If the file system crashes or becomes corrupted, we risk losing all our transactional data.



Database Management System(DBMS)

- A Database Management System (DBMS) is a software solution that enables the efficient management and utilization of data.
- It serves as a bridge between users and databases, providing a platform for data storage, retrieval, and manipulation.
- The need for a DBMS arises from the complexities of managing large amounts of data, ensuring data integrity, and supporting multiple user access.

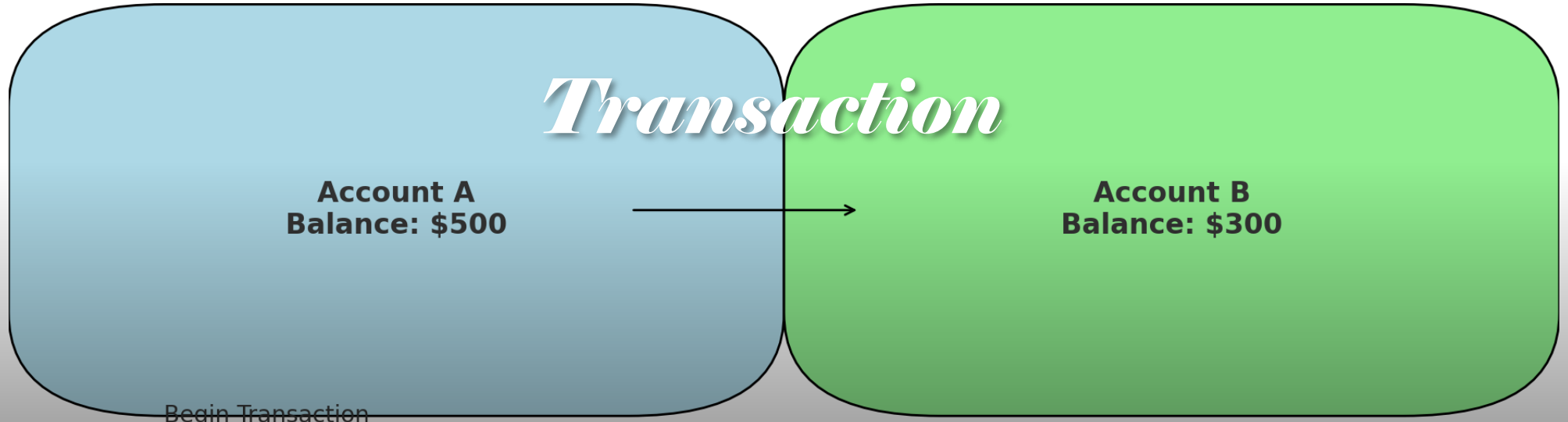
DBMS Functions

- Data Definition: Defining database structure, including schema and relationships.
- Data Storage: Storing data in a secure and efficient manner.
- Data Retrieval: Providing mechanisms for data retrieval and querying.
- Data Manipulation: Allowing data insertion, update, and deletion.
- Data Security: Ensuring authorized access and preventing data breaches.
- Data Integrity: Maintaining data consistency and accuracy.
- Performance Optimization: Ensuring efficient data processing and querying.
- Multi-User Access: Supporting concurrent access and managing user permissions.
- Data Backup and Recovery: Ensuring data safety and availability.

User View of DBMS

- Create (Data Insertion): This function allows users to add new data to the database, such as inserting a new customer record or adding a new product to an e-commerce platform.
- Read (Data Retrieval): This function enables users to retrieve existing data from the database, such as querying customer information or fetching product details.
- Update (Data Modification): This function allows users to modify existing data in the database, such as updating a customer's address or changing a product's price.
- Delete (Data Deletion): This function enables users to remove data from the database, such as deleting an obsolete product or removing a customer record.

Transaction



Begin Transaction

1. Check if Account A has $100
2. Debit $100 from Account A
3. Credit $100 to Account B
4. Commit Transaction

If any step fails:

Rollback Transaction

ACID

- **Atomicity**: A transaction is an atomic unit of processing; it either fully completes or fully fails.
- **Consistency**: A transaction must transition the database from one valid state to another, maintaining database rules. For example, if Account A has $500 and Account B has $300 before the transfer, the total amount of money in both accounts combined should still be $800 after the transfer.
- **Isolation**: Transactions are executed in isolation from one another. Intermediate states of a transaction are not visible to other transactions. Suppose two transactions are happening simultaneously: transferring $100 from Account A to Account B and transferring $50 from Account B to Account C. Isolation ensures that these transactions do not interfere with each other.
- **Durability**: Once a transaction has been committed, it will remain so, even in the event of a system failure.

Rollback

- Rollback is the process of undoing a transaction if it cannot be completed successfully.
- In our example, suppose the system crashes or encounters an error after deducting $100 from Account A but before adding it to Account B. The transaction system will initiate a rollback, which restores Account A's balance to its original amount of $500, ensuring that no partial changes are applied to the database.
- This maintains the database's integrity and consistency.

Data Models

- Relational: uses table-based structure. MySQL (open source), PostgreSQL (open source), Oracle, and SqlServer (Microsoft). Access is a relational DBMS for personal users.
- Document: Also known as NoSQL databases, uses self-describing documents. MongoDB, Couchbase, and RavenDB are popular document databases.
- Key-value: stores simple keys and values for high-performance data processing. Redis and Riak are two examples.

Relational DB

- High performance
- High reliability
- Data integrity (transactions)
- SQL

Relational Model

- Tables (relations)
- Rows (tuples)
- Columns
- Primary Key
- Foreign Key
- Data Integrity

SQL

- SQL (Structured Query Language) is a standard programming language designed for managing and manipulating data stored in relational database management systems.
- SQL is used to perform various operations such as creating, reading, updating, and deleting data in a database. These operations are commonly referred to as CRUD (Create, Read, Update, Delete) operations.

SQL Tutorial

https://www.w3schools.com/sql/

- SELECT
- UPDATE
- DELETE
- JOIN by foreign key

SQL Transaction

-- Step 1: Ensure Isolation

```
SET TRANSACTION ISOLATION LEVEL SERIALIZABLE;
```

-- Step 2: Begin Transaction, if any error happens before successful transaction commit, rollback all changes.

```
BEGIN TRANSACTION;
```

-- Step 3: Debit Operation (Part one of Atomicity)

```
UPDATE Accounts SET balance = balance - 100 WHERE account_id = 'Account_A';
```

-- Step 4: Credit Operation (Part two of Atomicity)

```
UPDATE Accounts SET balance = balance + 100 WHERE account_id = 'Account_B';
```

-- Step 5: Commit Transaction (Durability)

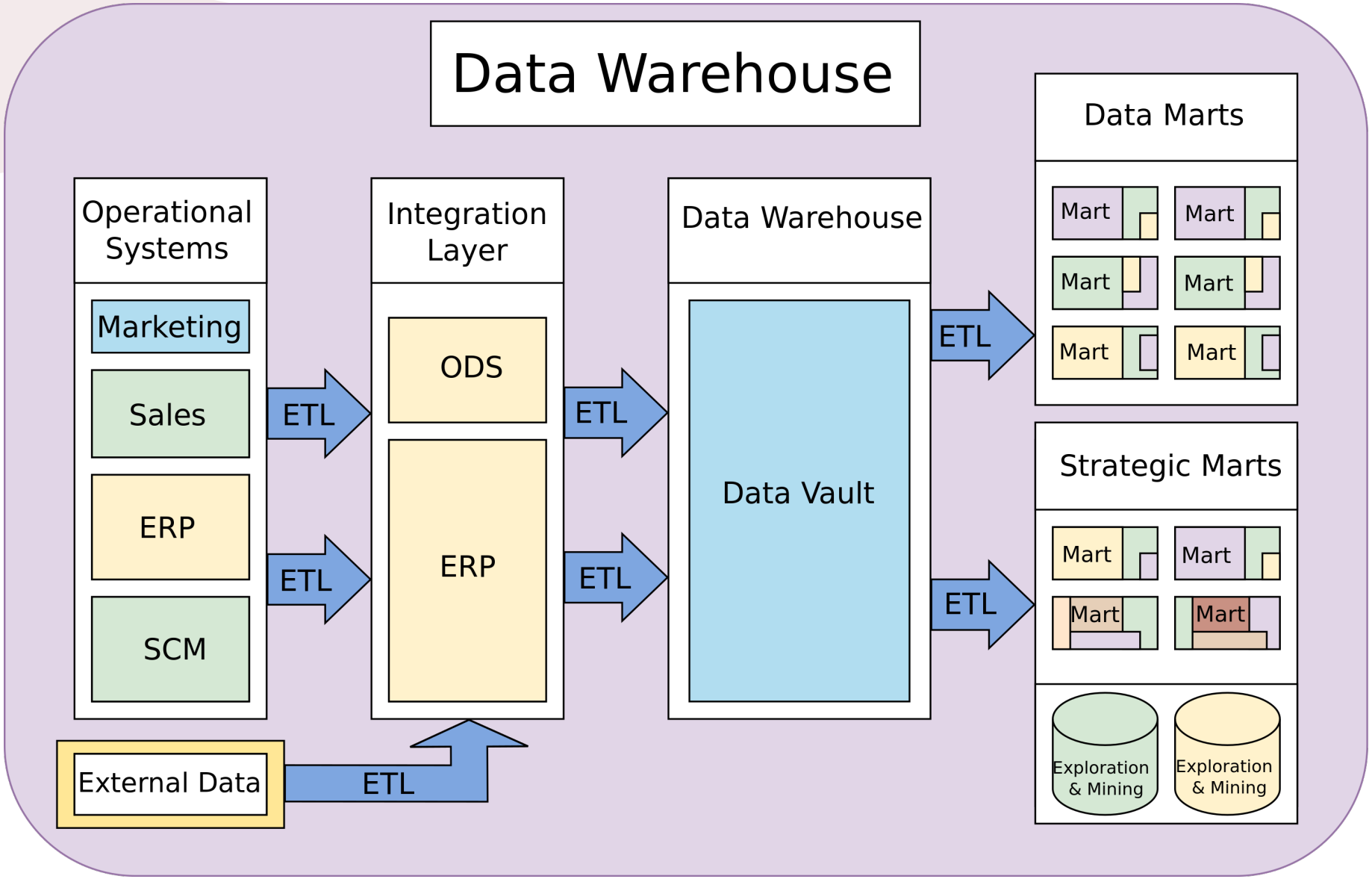
```
COMMIT TRANSACTION;
```

Data Warehouse

- A Data Warehouse is designed to support business intelligence (BI) activities, such as data analysis and reporting.
- A Data Warehouse typically consists of dimensions and facts. Dimensions provide context to the data, such as time, location, and product category, while facts represent the data being measured, such as sales amount and number of units sold.

Data Warehouse Benefits

- It enables improved decision-making by providing a comprehensive view of an organization's data.
- It increases efficiency by automating the process of consolidating and analyzing data.
- It stores and process aggregated data. Aggregated data refers to the summary of large datasets into smaller, more manageable chunks. This summary can be in the form of totals, averages, counts, or other calculations.


Data Warehouse
Operational Systems
Marketing
Sales
ERP
SCM
External Data
ETL
ETL
ETL
Integration Layer
ODS
ERP
ETL
ETL
Data Warehouse
Data Vault
ETL
ETL
Data Marts
Mart
Mart
Mart
Mart
Mart
Mart
Strategic Marts
Mart
Mart
Mart
Mart
Exploration & Mining
Exploration & Mining

Data Warehouse Components

- ETL (Extract, Transform, Load): ETL is a process used to extract data from multiple sources, transform the data into a standardized format, and load it into a target system such as a data warehouse.
- Data Vault: Data Vault is a data warehousing architecture that focuses on storing data in a raw, untransformed form.
- Data Marts: Data Marts are subsets of a data warehouse, designed to serve a specific business need or department. They are typically built on top of a Data Vault and contain transformed data, making it easier for business users to access and analyze the data

Big Data

- Big Data refers to the vast and complex sets of data that traditional data processing tools and techniques cannot manage due to their large size, variety, and speed of generation.
- It is defined by its four V's: Volume, Variety, Velocity, and Veracity.
- Big Data includes structured, semi-structured, and unstructured data, such as social media data, sensor data, and text data.

Cloud Database

- Scalability
- Reliability
- Reduced Management Effort

- Examples: Microsoft Azure, AWS, Google Cloud