
LUT
University
LAND
OF THE
CURIOUS

WHO AM I?

- Who: Jiri Musto, D.Sc.
- Position: Doctoral researcher
- Knowledge area:
  - Data and information quality, information systems, data mining, data analysis, relational databases, NoSQL databases,
  - Object-oriented programming, Android development, game development, web-development
- Courses I teach (2022-2023):
  - Bachelor level: Basics of database systems, database system management
  - Master's level: Data-intensive systems (distributed databases), business intelligence and data mining, business process modelling

CT30A3401 – DISTRIBUTED SYSTEMS

DISTRIBUTED DATABASE SYSTEMS

Lecture

Jiri Musto, D.Sc.

AGENDA

- General information
- Replication and sharding
- Query processing
- System architectures
- Final notes

CT30A3401 – DISTRIBUTED SYSTEMS

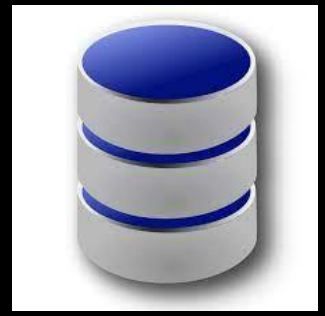
DISTRIBUTED DATABASE SYSTEMS: GENERAL INFORMATION

Lecture

Jiri Musto, D.Sc.

WHAT IS A DATABASE?

- In theory: Databases are anything that store interrelated data
  - Text files, excel sheets, sheets of paper, etc.



- In practice:
  - Databases illustrate the real world
  - Database is a logically coherent collection of data
  - The database structure and data are designed and implemented for a specific purpose

MOTIVATION FOR DATABASES

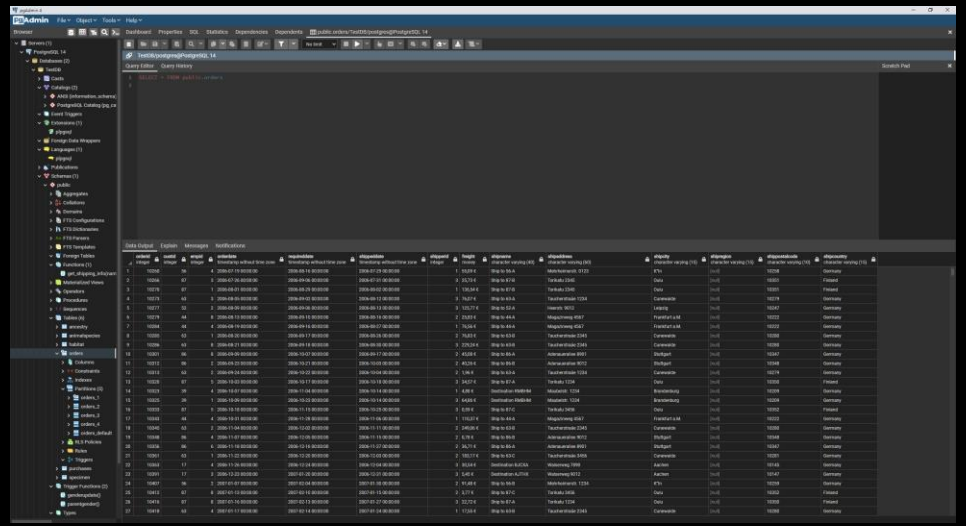
- Databases were created to resolve the issue of data storage
  - Storing data in an application limits access
  - Using a collection of without a database system requires repeating functionality for each application
  - Provides several advantages such as efficiency, portability, reliability, and security
- Databases store data in various structures and formats
- Data is stored in files (database files)
  - The file type and structure depends on the database management system
  - .xml, .csv, json, .db, .couch, .mdf/.ldf, etc.
- Databases are best used with multiple users and changing data

DATABASE TYPES

- Databases can stored on different locations:
  - Local device (client):
    - Often used to store cached / temporary data
    - Can be a database or flat file
    - Tied to the local device
  - Physical database (server):
    - Most common database
    - Accessed in local network or remotely
    - Access controlled with authentication
  - Cloud database:
    - Newest option for databases
    - Data accessed through the cloud platform
    - Access controlled with authentication

DATABASE MANAGEMENT SYSTEMS (DBMS)

- DBMS is the software that operates between the data and connecting applications
- DBMS gives a logical view of data to connecting applications / users and handles the physical operations to the data



- Used to centralize data management and security
- Often offers a graphical user interface to manage the data

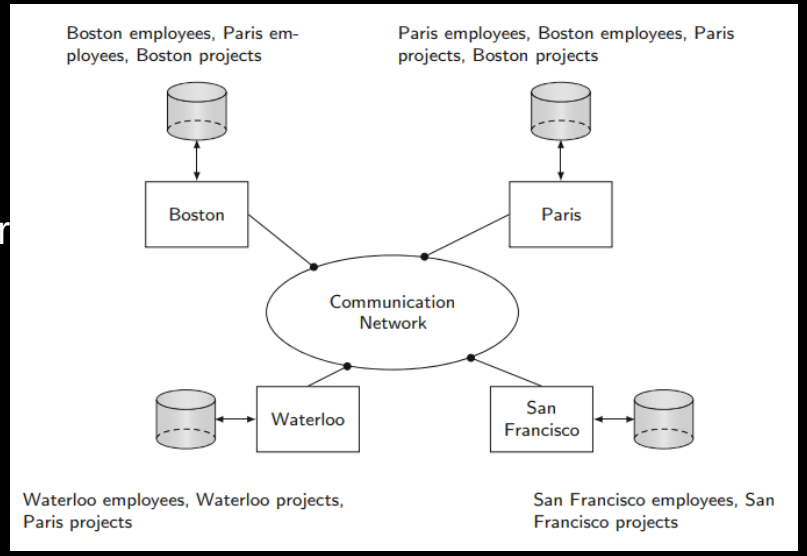
DATABASE SYSTEM

- Database system is the combination of
  - Database
  - Database management system
  - Software(s) that uses the aforementioned
    - Mediators, wrappers, user interfaces, etc.

- Database systems can be:
  - Centralized or distributed
    - Databases operate in one location or multiple locations (distributed over a network)
  - Single or multidatabase
    - Multidatabase system can have similar or different databases

EXAMPLES OF DIFFERENT DATABASE SYSTEMS

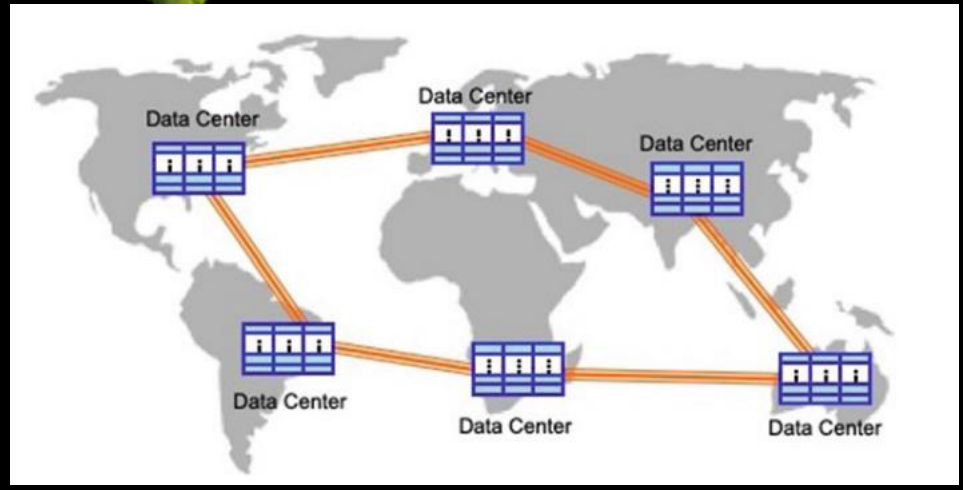
- Centralized system
  - Customer database for small company
- Distributed system
  - Customer database for different locations
  - Each location has their own customer database
  - Main office has a database containing all customers



- Peer-to-peer
  - Torrents, blockchain
- Data Stream System
  - Streaming services
- Data lake
  - Unstructured big data storage
- Data warehouse
  - Structured data lake

Source: Özsy, M. T., Valduriez, P., Principles of Distributed Database Systems, 978-3-030-26252-5

DISTRIBUTED DATABASES



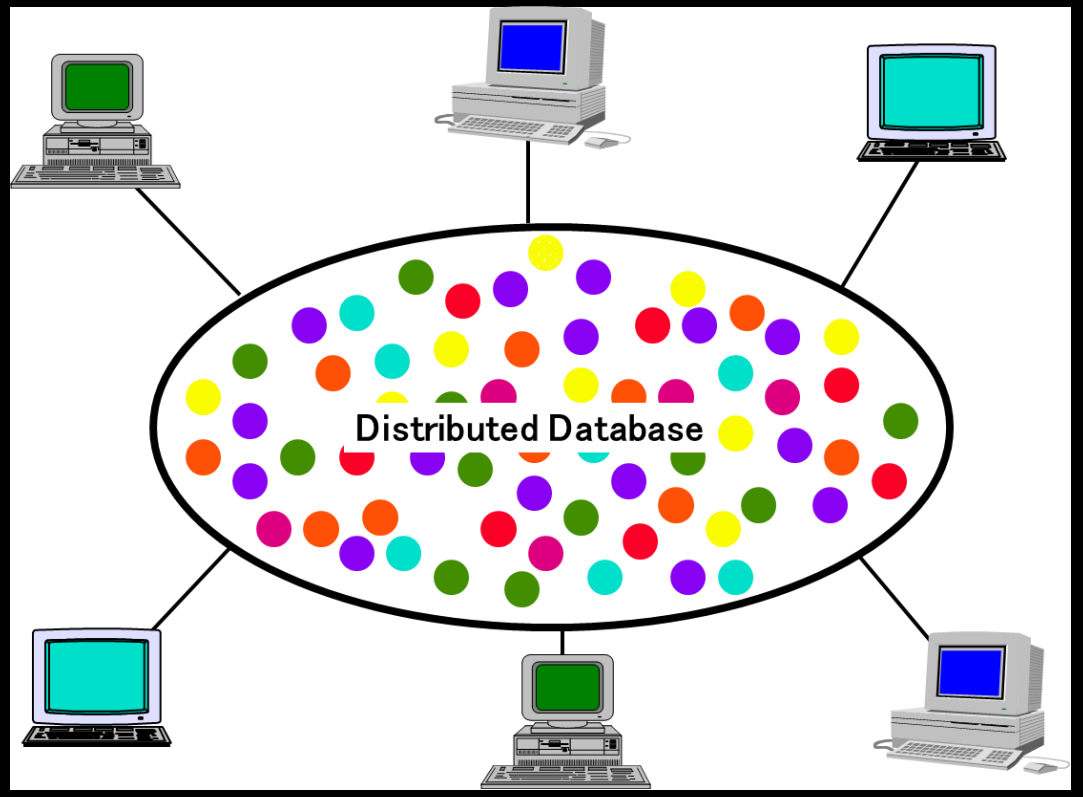
- Data is stored at multiple sites
- Databases are **logically integrated** but **physically distributed**
- Distributed database is a database, not a collection of files
- Distributed DBMS is not a remote file system

Source: Özsy, M. T., Valduriez, P., Principles of Distributed Database Systems, 978-3-030-26252-5

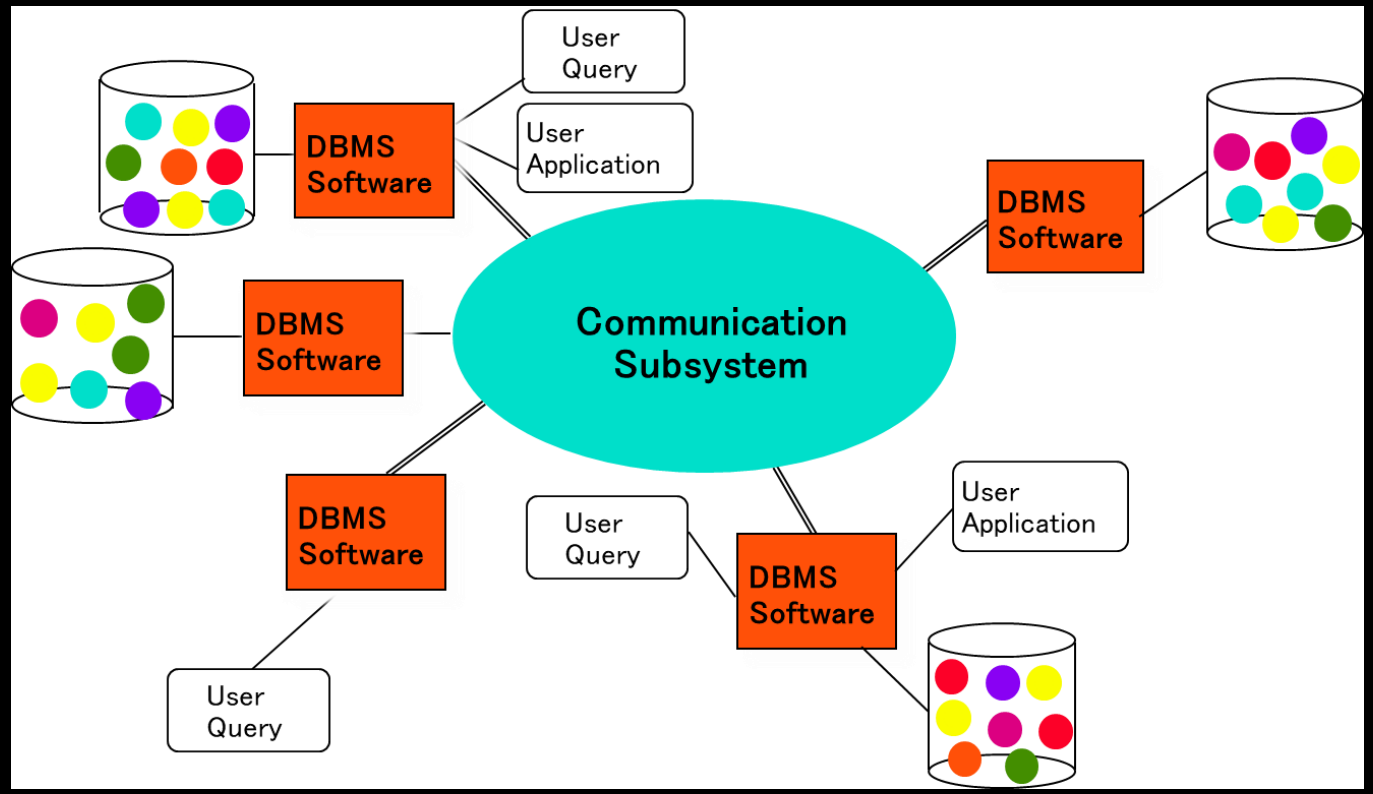
DISTRIBUTED DATABASE SYSTEM PROMISES

- Transparent management
  - Users do not have to know how the low-level system behaves
- Improved reliability and availability
  - Replicated data, failure protocols, concurrency transparency
- Improved performance
  - Parallel execution
- Easier expansion / scalability
  - Add more servers vs. Upgrade servers

DISTRIBUTED DATABASE SYSTEM – USER VIEW



DISTRIBUTED DATABASE SYSTEM – REALITY



DISTRIBUTED DATABASE SYSTEM ISSUES

- Design
  - How to distribute
  - Replication (centralized / distributed, eager / lazy)
- Query processing
  - Optimization
  - User transactions to data manipulation
- Concurrency control
  - Deadlocks, synchronization
- Reliability
  - Durability, how to make the system resilient to failures

CT30A3401 – DISTRIBUTED SYSTEMS

DISTRIBUTED DATABASE SYSTEMS: REPLICATION, SHARDING, QUERY PROCESSING

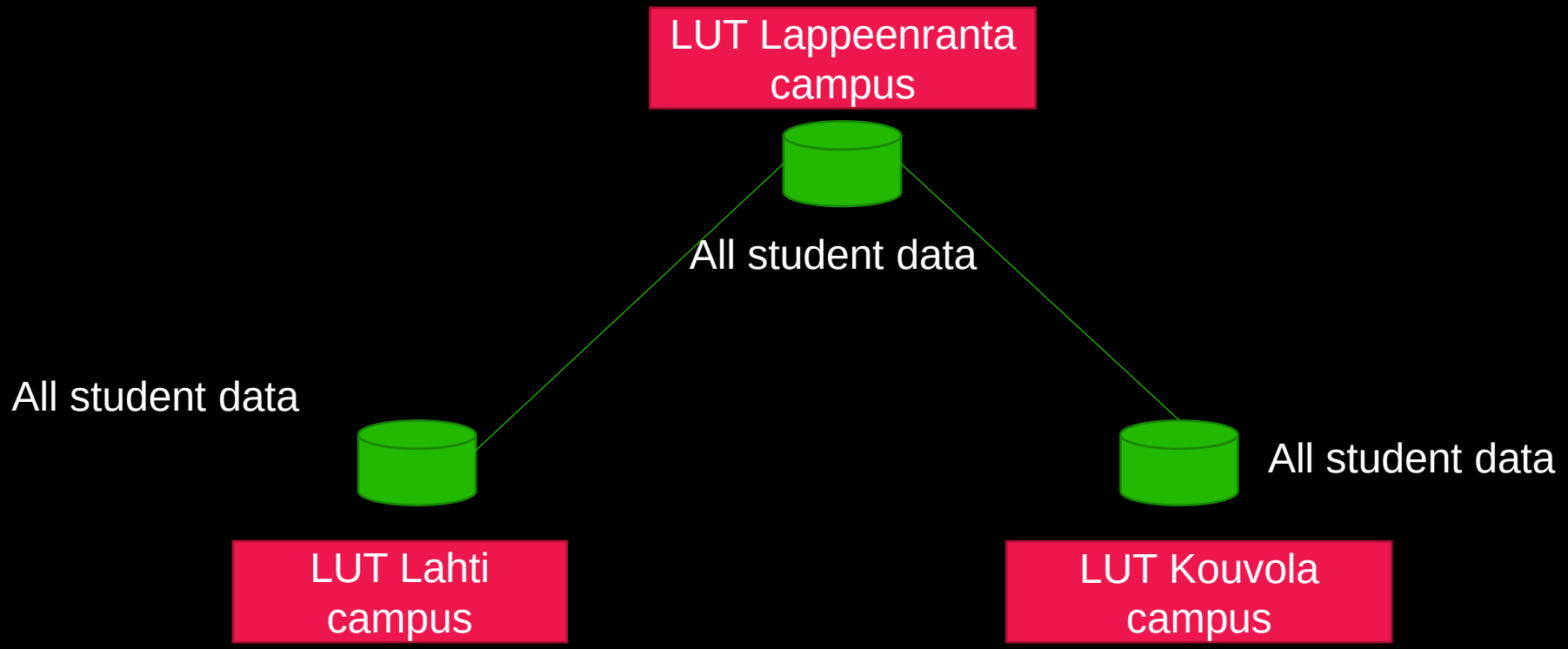
Lecture

Jiri Musto, D.Sc.

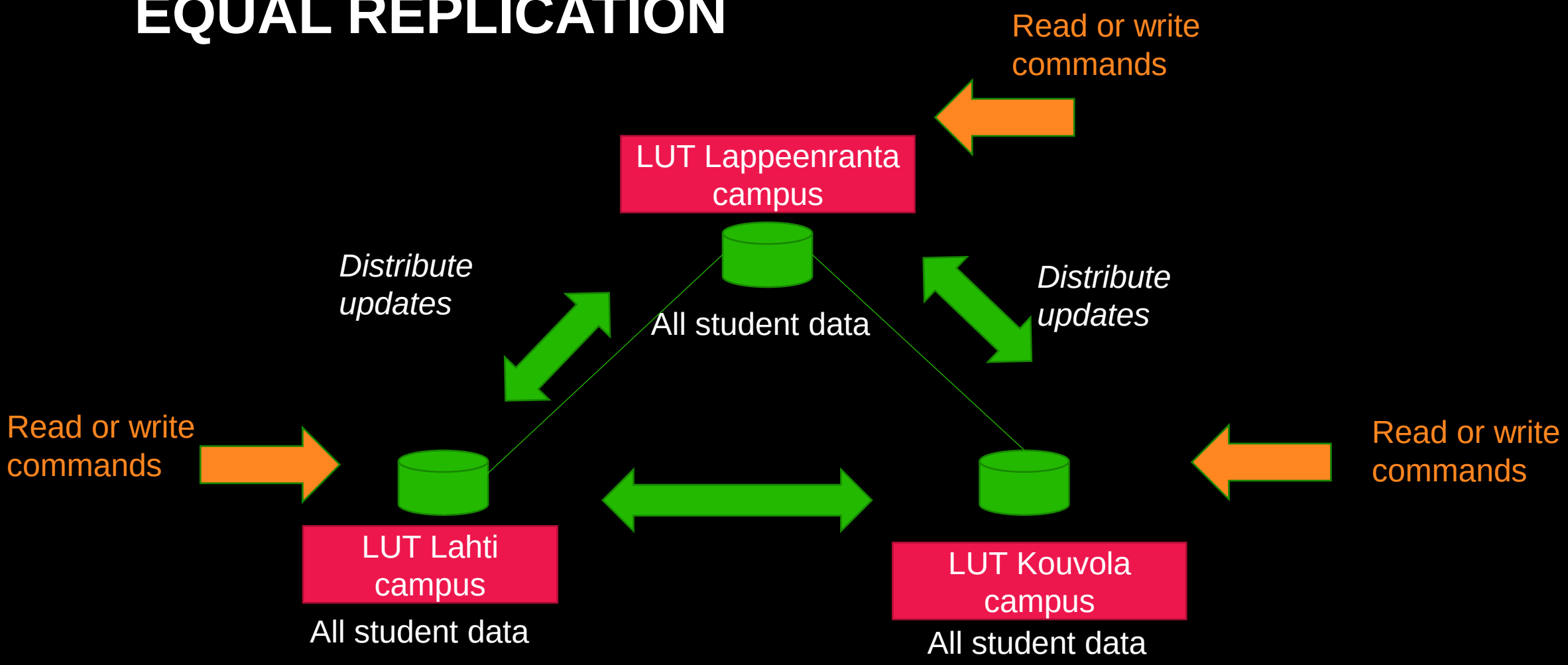
TERMINOLOGY

1. Replication
   - Same data is duplicated to multiple databases
2. Fragmentation / partitioning
   - Both terms are used, fragmentation also has another meaning in database terminology
   - In distributed databases, the terms refer to splitting tables into smaller pieces and storing them separately
3. Sharding
   - Similar to partitioning with the difference that sharding explicitly implies that the data is stored in different locations

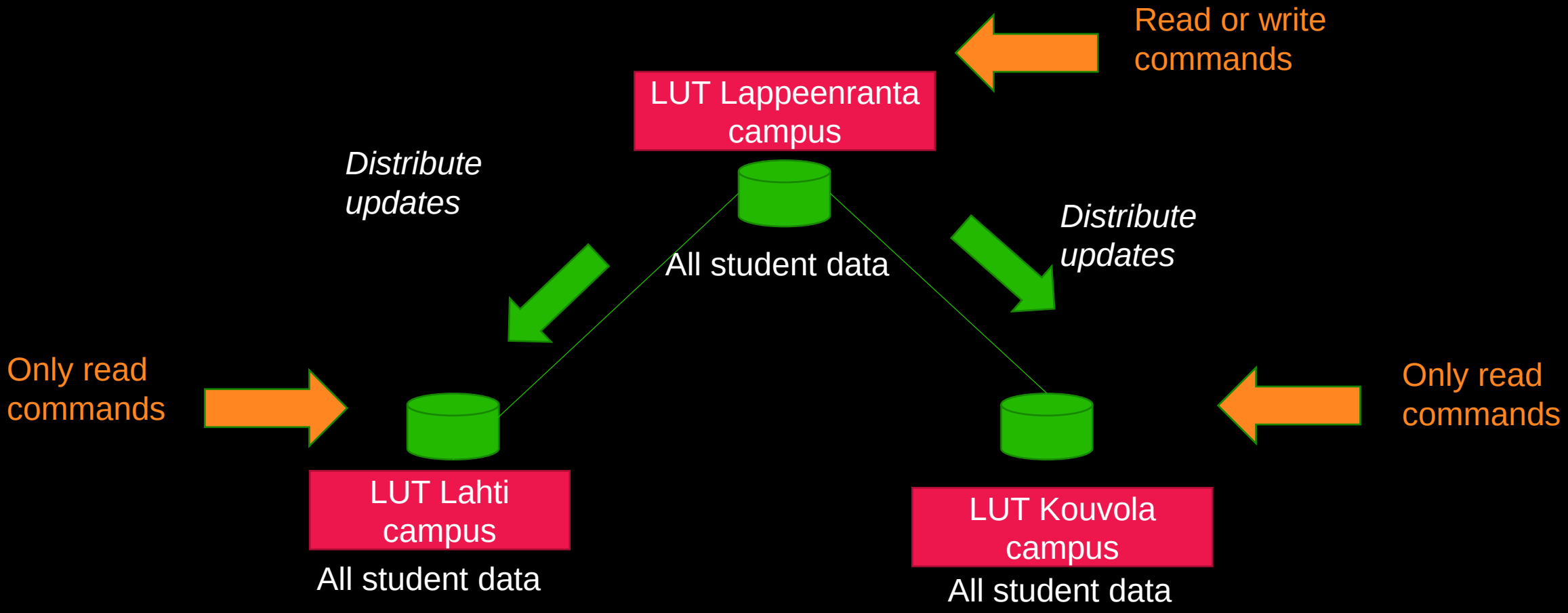
REPLICATION



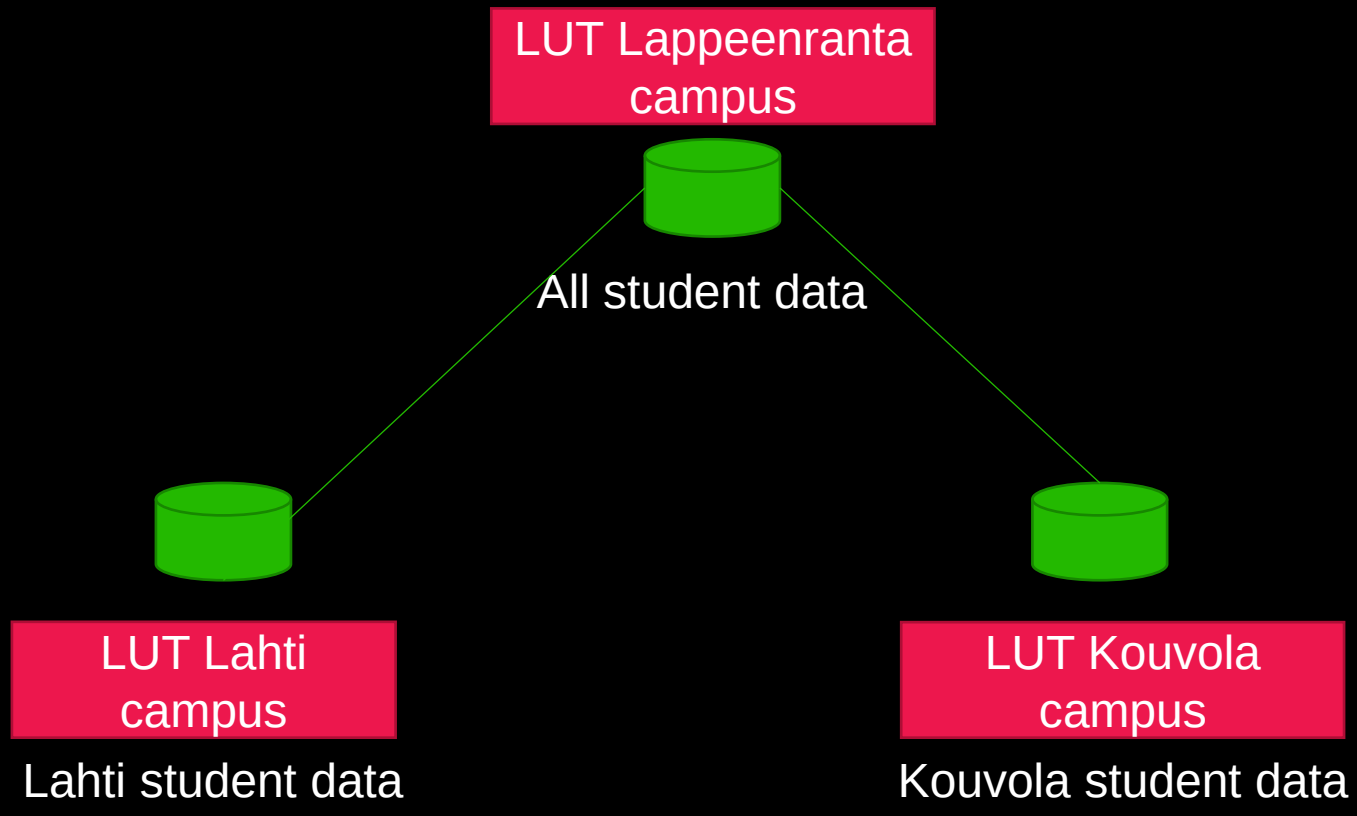
EQUAL REPLICATION



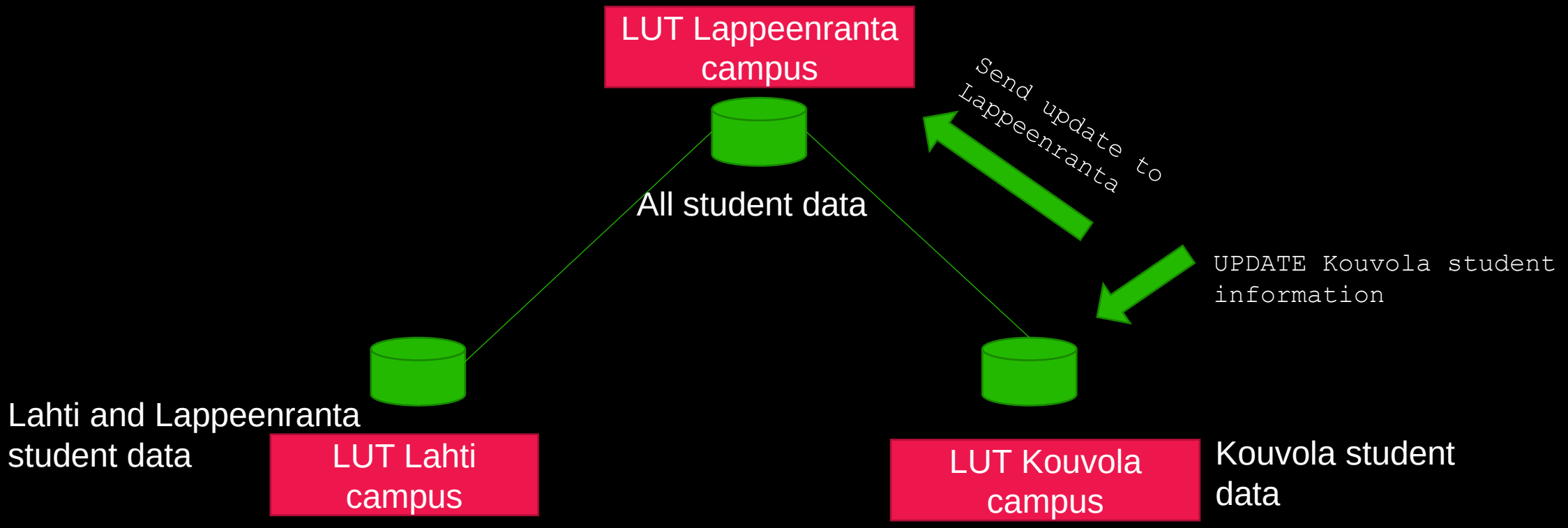
PRIMARY-SECONDARY REPLICATION



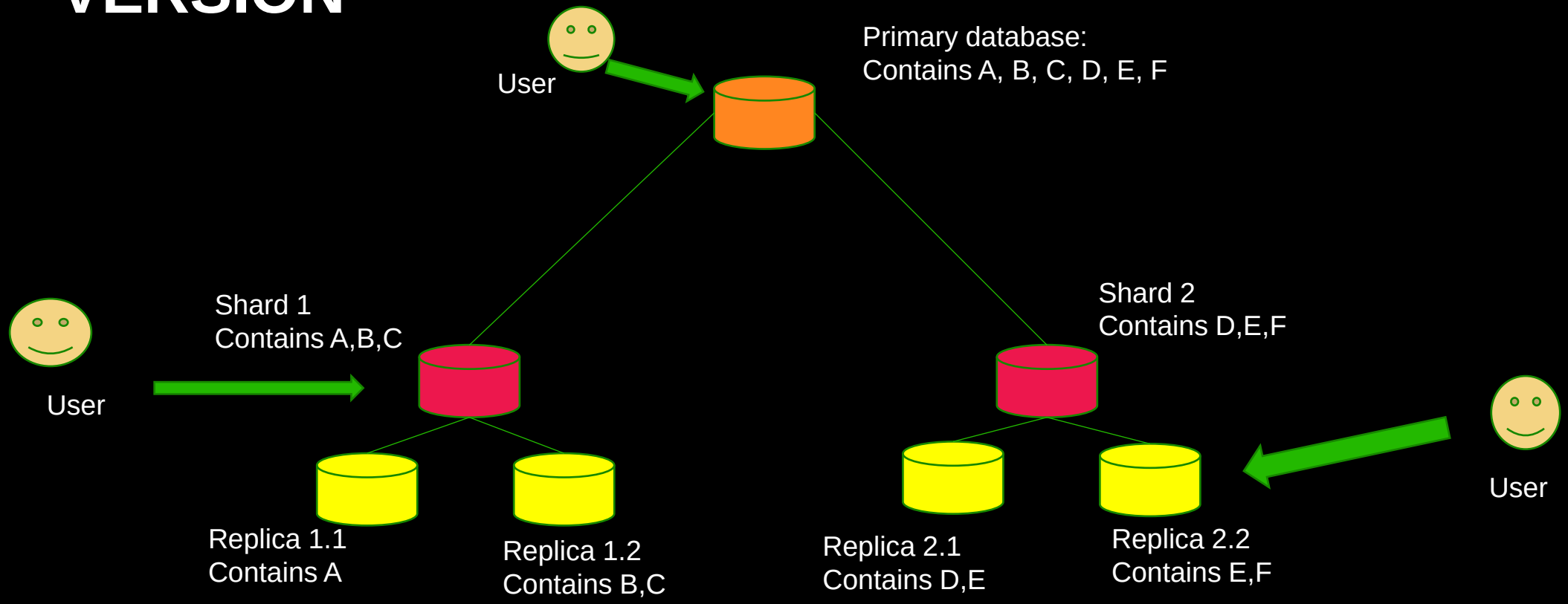
SHARDING / PARTITIONING



DATA SHARDING AND REPLICATION

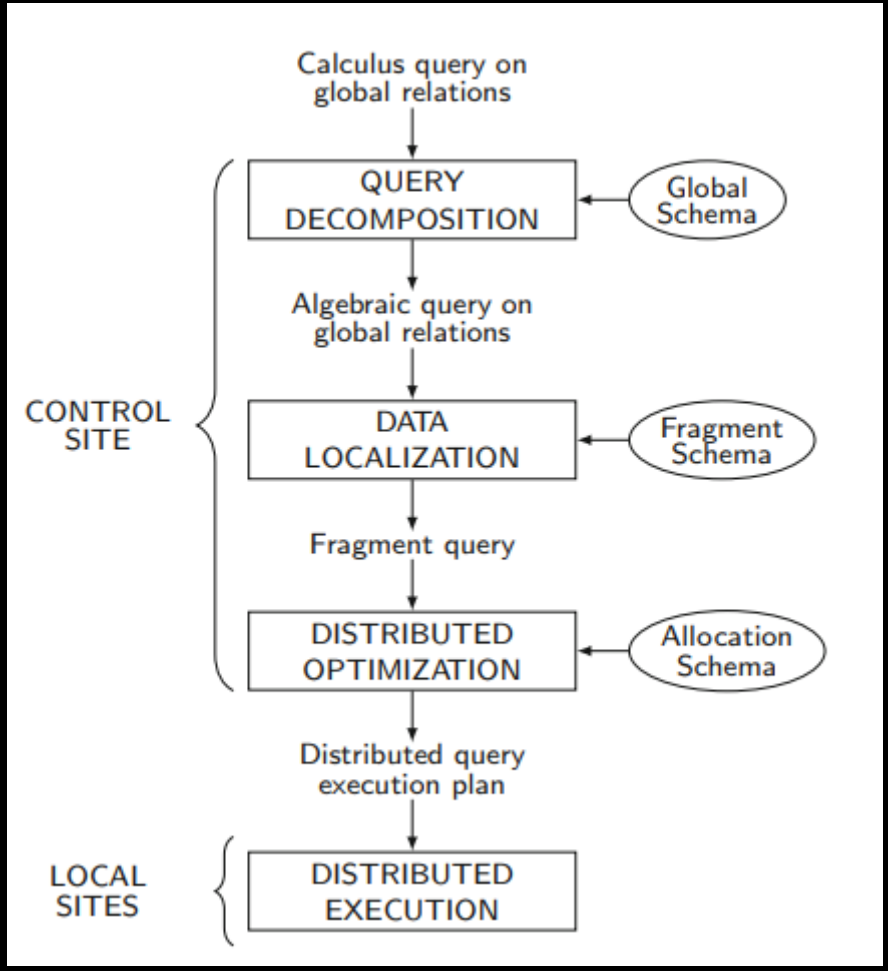


DATA SHARDING AND REPLICATION GENERIC VERSION



QUERY PROCESSING

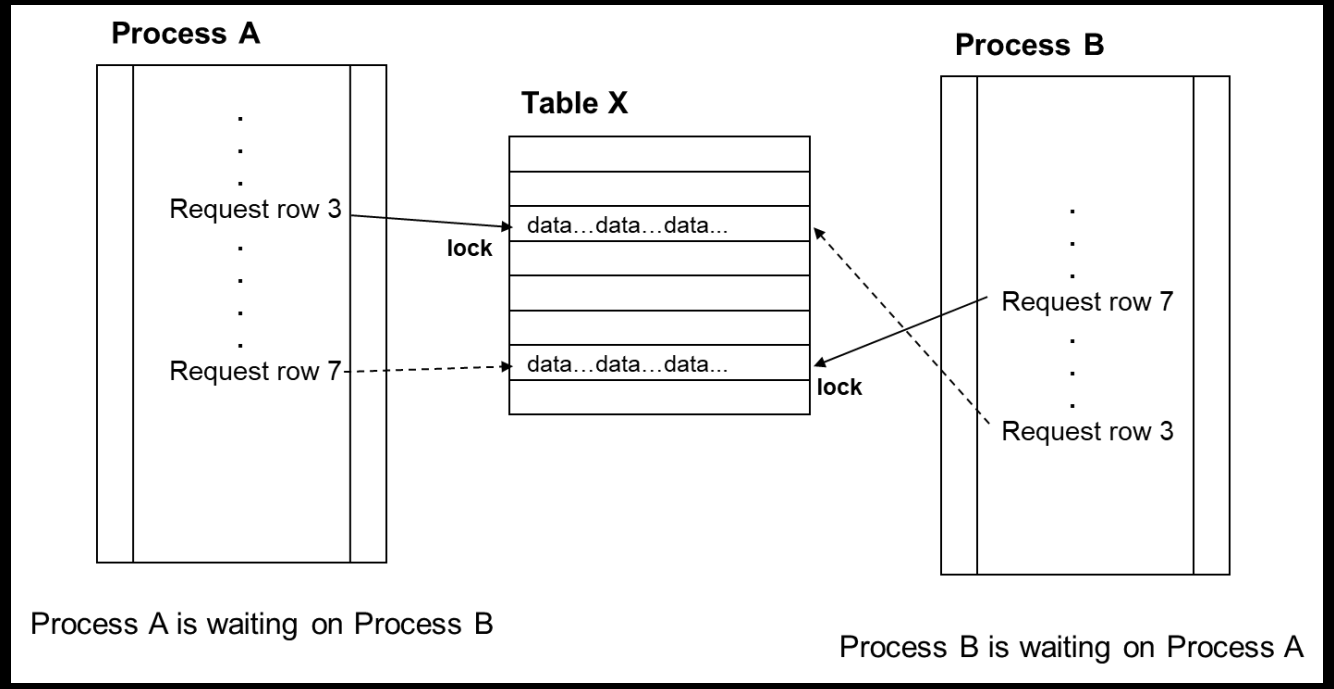
- Queries are distributed amongst the relevant databases
- Communication between databases is necessary and there are different communication protocols
- One point always acts as the ***control site*** that makes sure all involved databases are ready and the query can be processed
- Depending on the communication, there are different failure/termination protocols as well



Source: Özsy, M. T., Valduriez, P., Principles of Distributed Database Systems, 978-3-030-26252-5

DEADLOCKS AND TIMEOUTS

- Concurrent users inevitably lead to users accessing the same data
- If another user reaches the data first
  - The other users are put on hold
  - Timeout may occur
- If users require the same data at the same time
  - Deadlock occurs
- In distributed databases, amount of deadlocks increase because of delays between the databases



CT30A3401 – DISTRIBUTED SYSTEMS

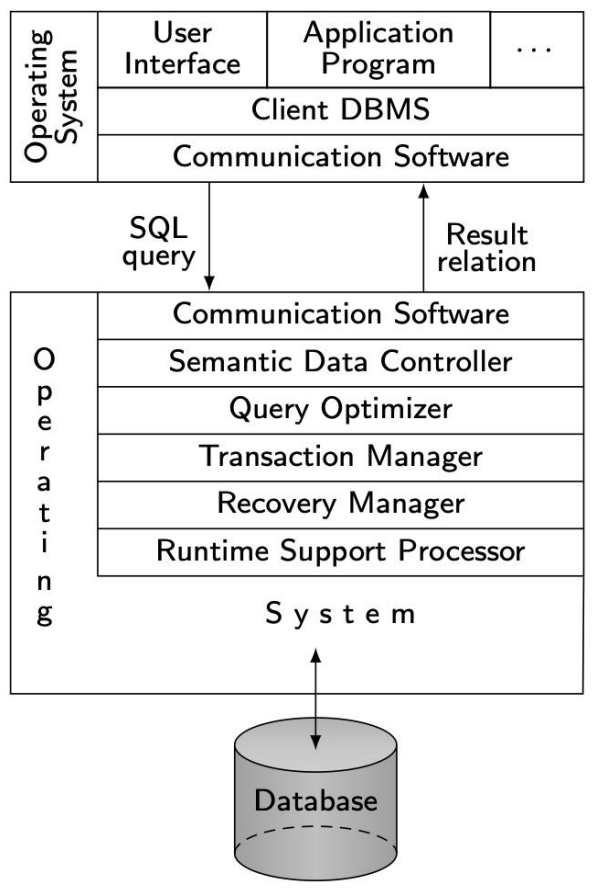
DISTRIBUTED DATABASE SYSTEMS: SYSTEM ARCHITECTURES

Lecture

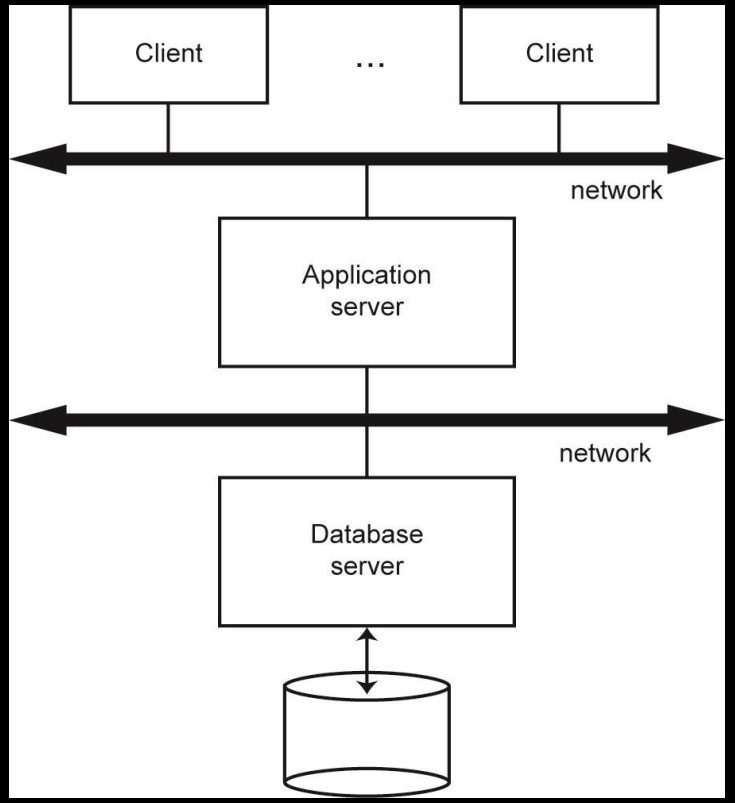
Jiri Musto, D.Sc.

CLIENT/SERVER ARCHITECTURE

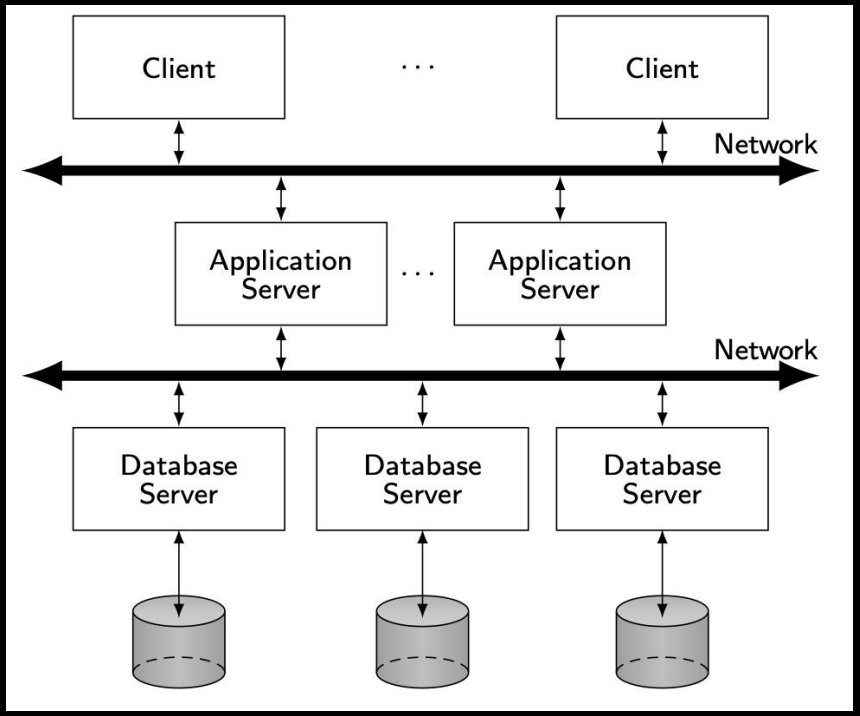
Client / server



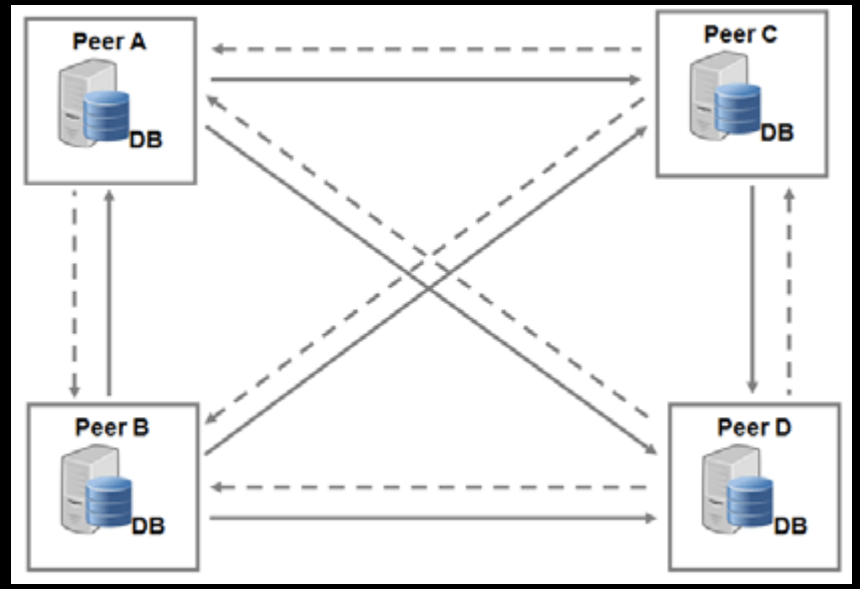
Database server



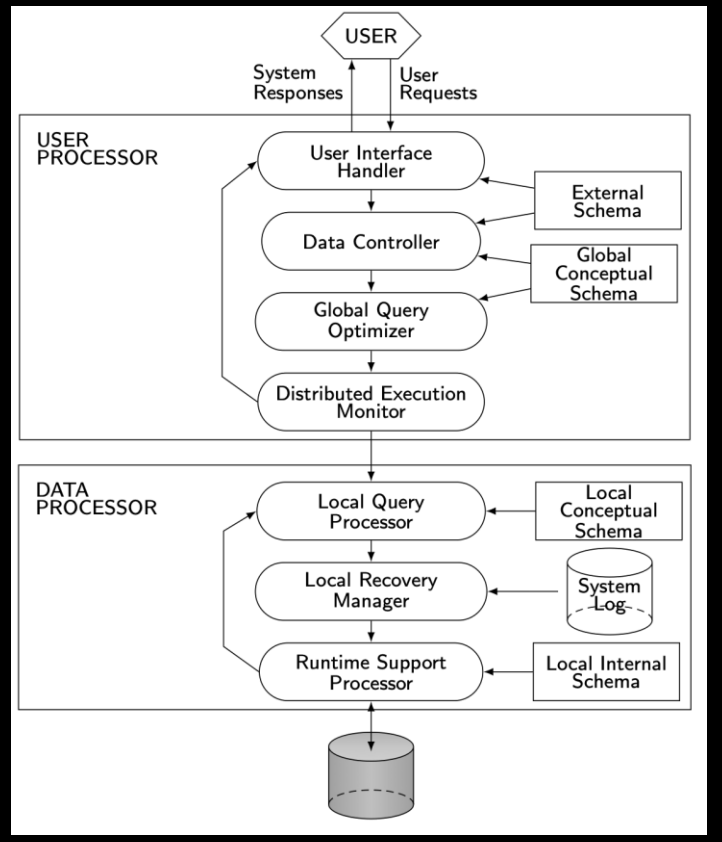
Distributed Database server



PEER-TO-PEER ARCHITECTURE

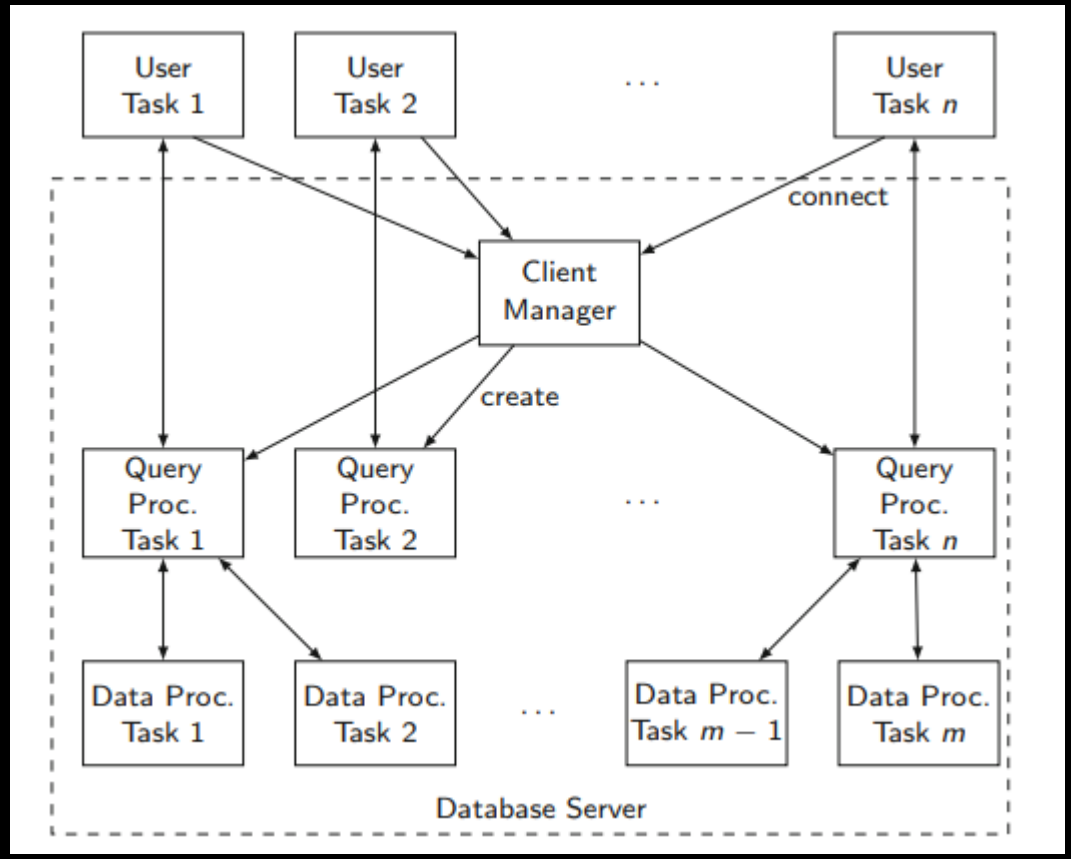


- There is a global schema of the existing data
- User query is distributed to peers
- Each peer has their own local schema, that is a part of the global schema



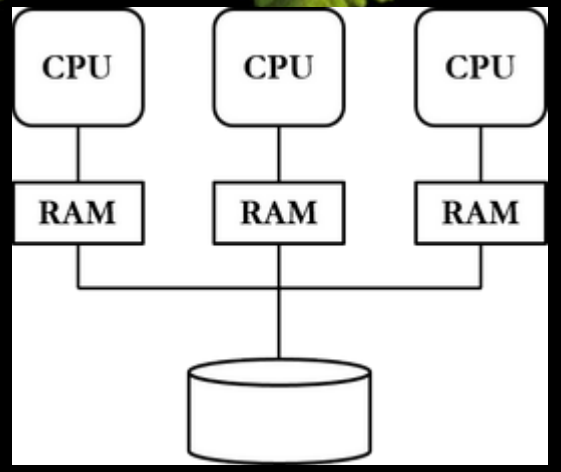
PARALLEL DATABASE SYSTEM

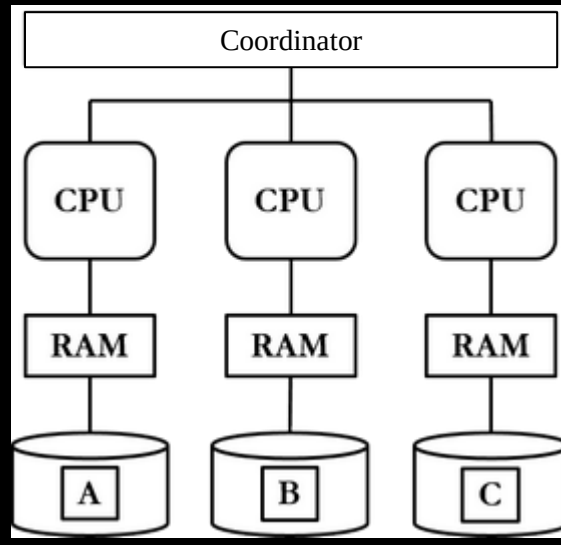
- A distributed database system on parallel computers
- Main purpose is to improve performance
  - I/O bottleneck
- Useful in:
  - Online transaction processing (OLTP)
  - Decision support systems (DSS)
  - Parallel query processing



PARALLEL ARCHITECTURE

- Shared disk
  - Each processor has their own instance of the same database
  - Requires a lock manager for global cache consistency
- Shared nothing
  - Each processor has their own database instance
  - Best cost/performance ratio
- Shared memory / Shared everything
  - Processors share the central memory
  - Not a real parallel architecture anymore, more of an extension on the other two (because each modern CPU is multicore)





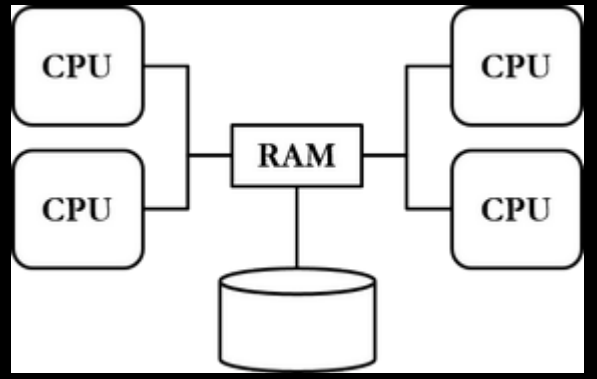
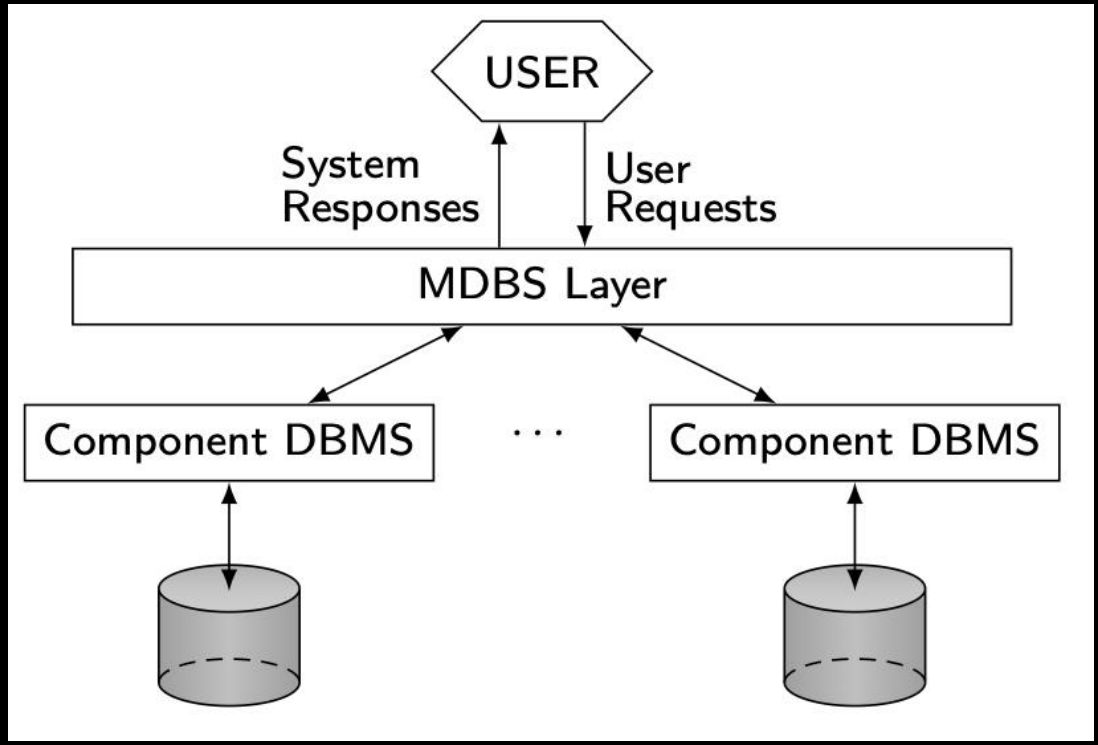


Image source: https://tim.jyu.fi/view/kurssit/tktl/itka204/kurssimoniste

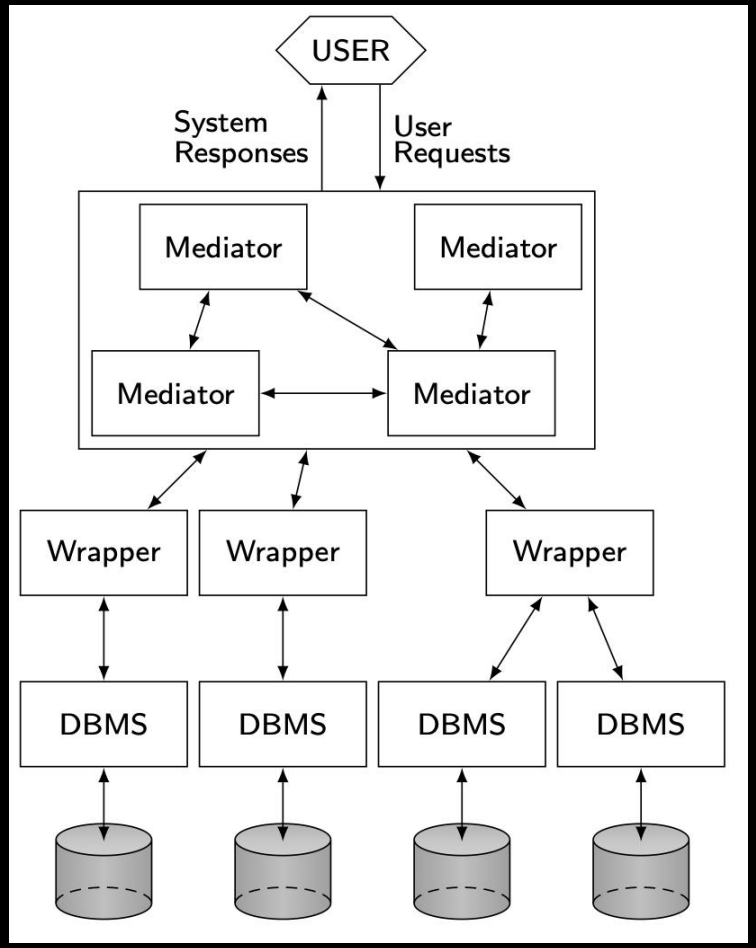
MULTIDATABASE SYSTEM

- DBMS have no idea of each others existence
- MDBS layer handles the communication between user and the databases
- If databases are heterogenous, wrappers and mediators are used



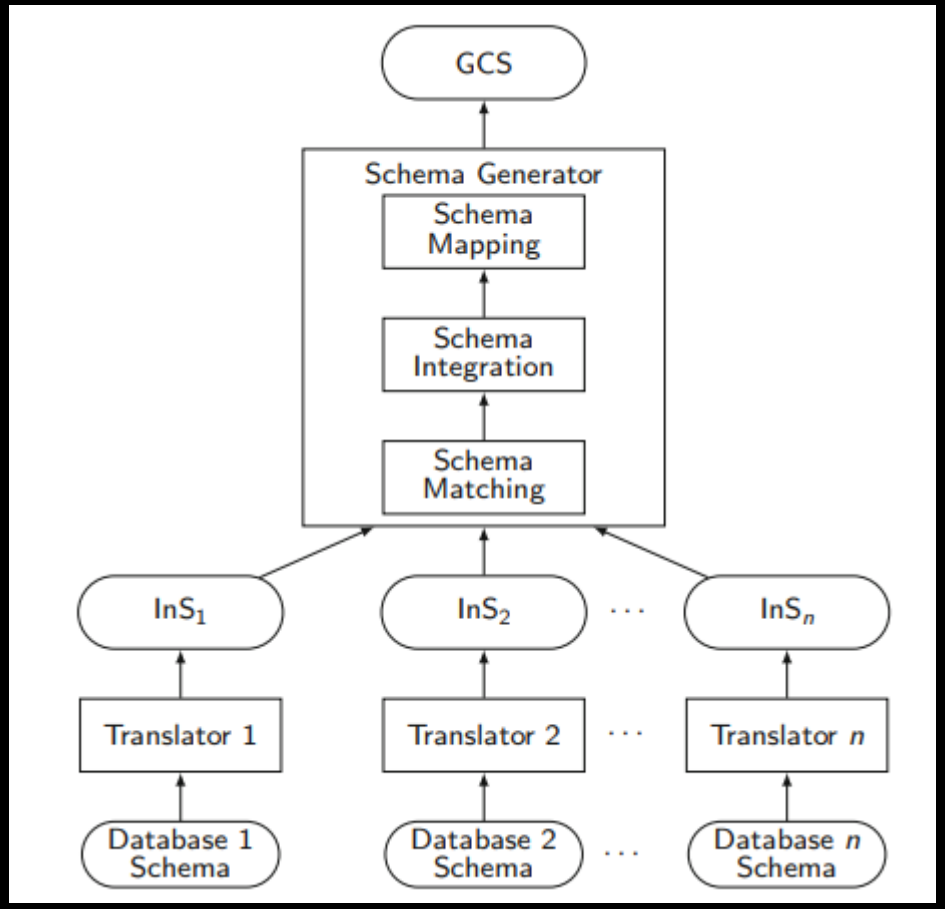
MEDIATOR/WRAPPER

- Mediators
  - Communicate and distribute user queries to the correct locations
  - Combine results from different DBMS to one package
- Wrappers
  - Rewrite the queries to match their local DBMS
  - Wrap results to match what client expects



DATABASE INTEGRATION

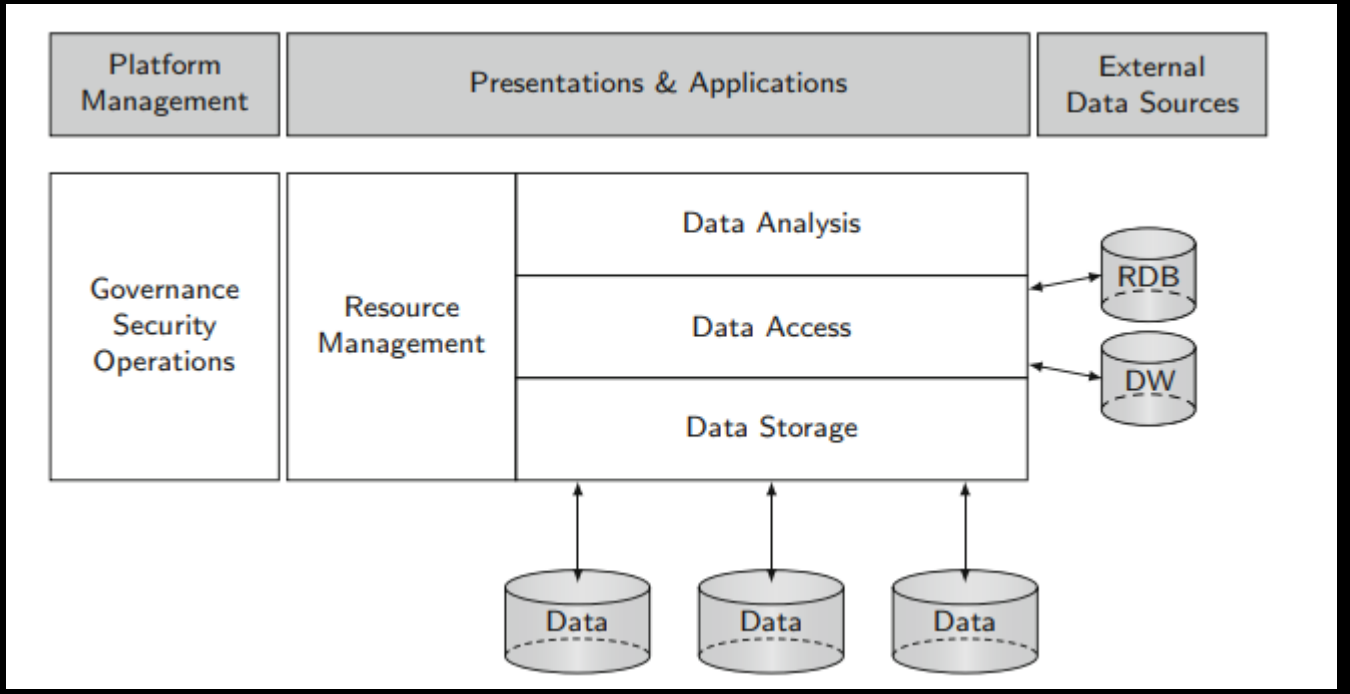
- Integrating different databases together requires that the database schemas can be matched together
- Two ways: Bottom-up or top-down
- Bottom-up
  - Each local database schema is translated to a intermediate schema which are then matched an combined into a global schema
- Top-down follows the normal distributed database design



Source: Özsy, M. T., Valduriez, P., Principles of Distributed Database Systems, 978-3-030-26252-5

DATA LAKES

- Data lakes are used in big data applications
- Similar to data warehouse but the data is stored in its “natural” format
  - Data parsing done during query
- Lacks consistency and quality

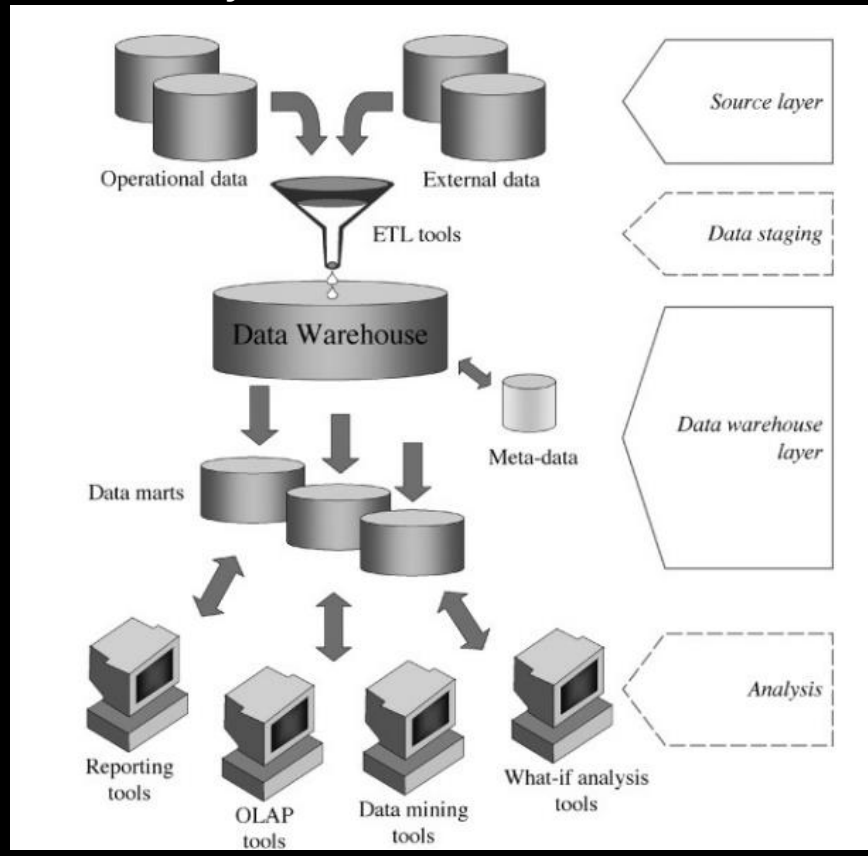


Source: Özsy, M. T., Valduriez, P., Principles of Distributed Database Systems, 978-3-030-26252-5

DATA WAREHOUSE

- An organized version of data lake
  - Data is curated with ETL tools (organized and structured based on warehouse design)
- OLAP (and OLTP) is the main use case for data warehouses
- Data warehousing could be defined as:
  "*A collection of methods, techniques, and tools to support knowledge workers and analysts to conduct data analyses that help performing decision making processes and improving information resources*"
  - Sounds quite similar to the information system definition, right?

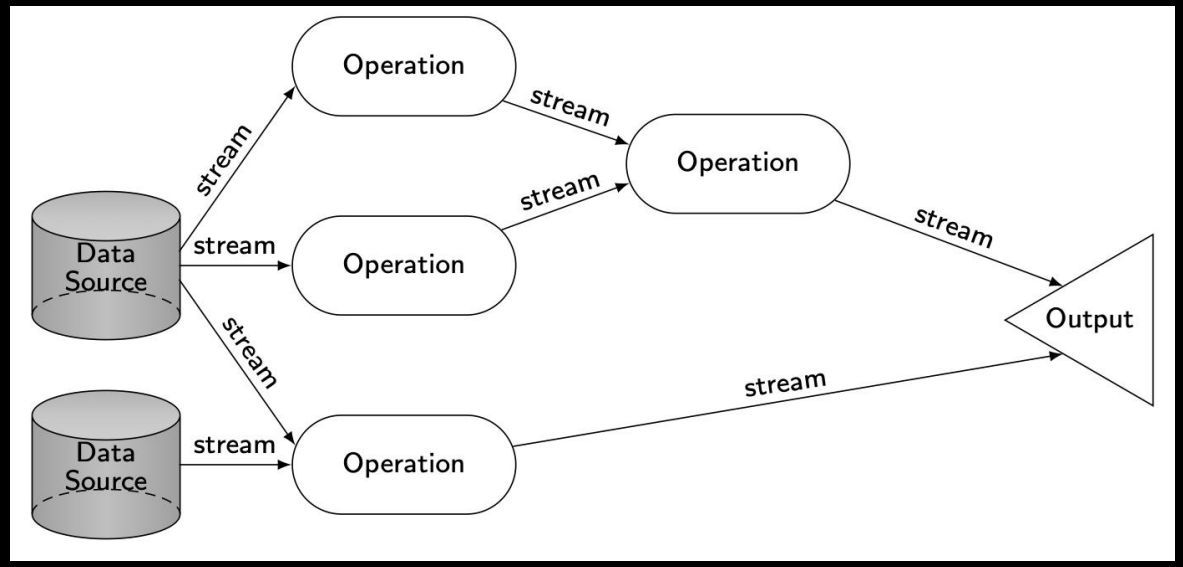
Two-layer architecture



Source: Golfarelli, M., Rizzi, S., Data Warehouse Design, Modern Principles and Methodologies, 978-0-07-161040-7

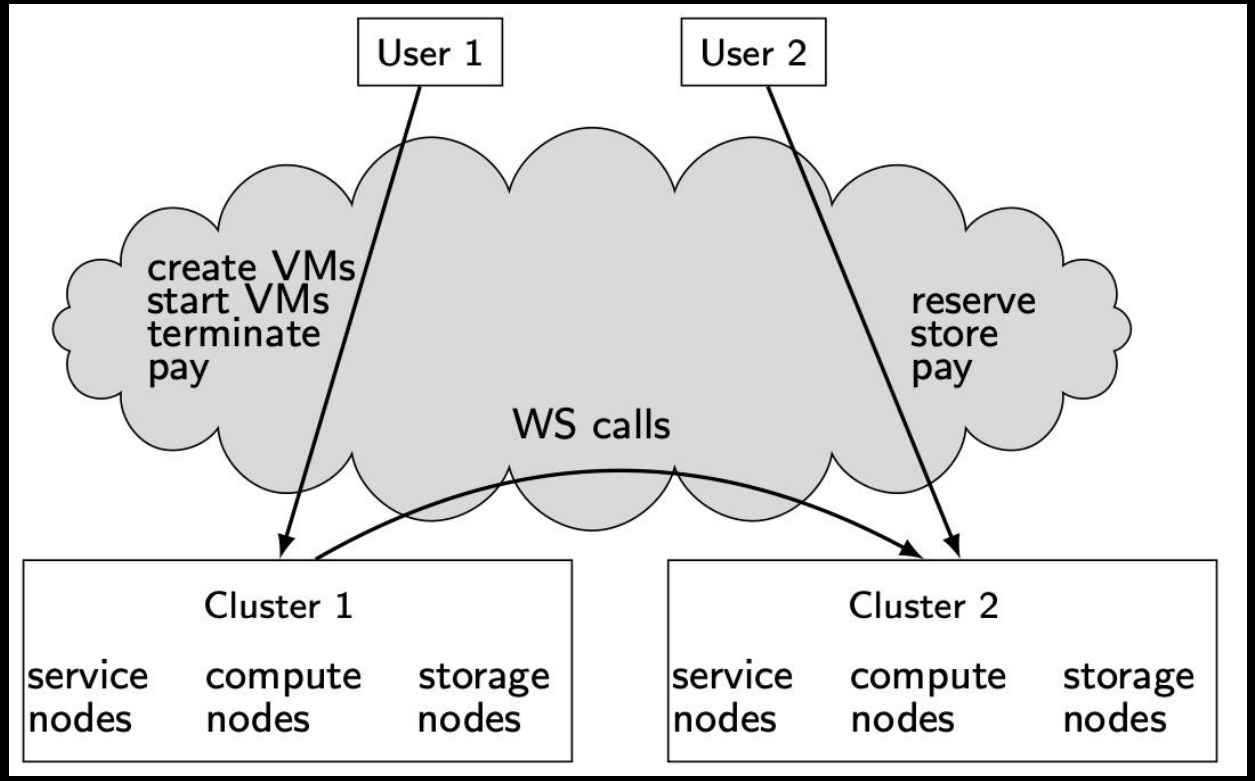
DATA STREAM SYSTEMS

- Queries
  - Persistent queries
  - Data pushed through query plan, not pulled
- Processing issues
  - Arrivals and expiration in windowed query
  - Load management (stream arrival rate > processing capability)
  - Out-of-order processing (arrival vs. creation order)



CLOUD COMPUTING

- Provided over the Internet
- Three main database models
  - Shared DBMS server
    - Tenants share servers but have their own databases
    - Good isolation, inefficient resource management
  - Shared database
    - Tenants share databases but have their own schema & tables
    - Good resource management and isolation, but large amount of overhead because of the number of tables
  - Shared tables
    - Tenants share tables and each row is tied to specific customer
    - Good for resource management, bad for security and performance



CT30A3401 – DISTRIBUTED SYSTEMS

DISTRIBUTED DATABASE SYSTEMS: FINAL NOTES

Lecture

Jiri Musto, D.Sc.

USE AND RELEVANCE IN MODERN WORLD

- Most companies with a larger userbase will have a distributed database system
- With a distributed system, often comes a distributed database system (but not always)
- Even if a company does not have a distributed database, they may decide to build one for their business purposes (business intelligence usage)

USE-CASES

- Streaming companies (Netflix, HBO, Disney+)
- Video game companies (Steam, EA, Ubisoft)
- Video games (online games, first-person shooters)
- E-commerce (eBay, Amazon)
- Universities (LUT)
- IT companies (TietoEvry)
- Factories (UPM, StoraEnso)

NETFLIX

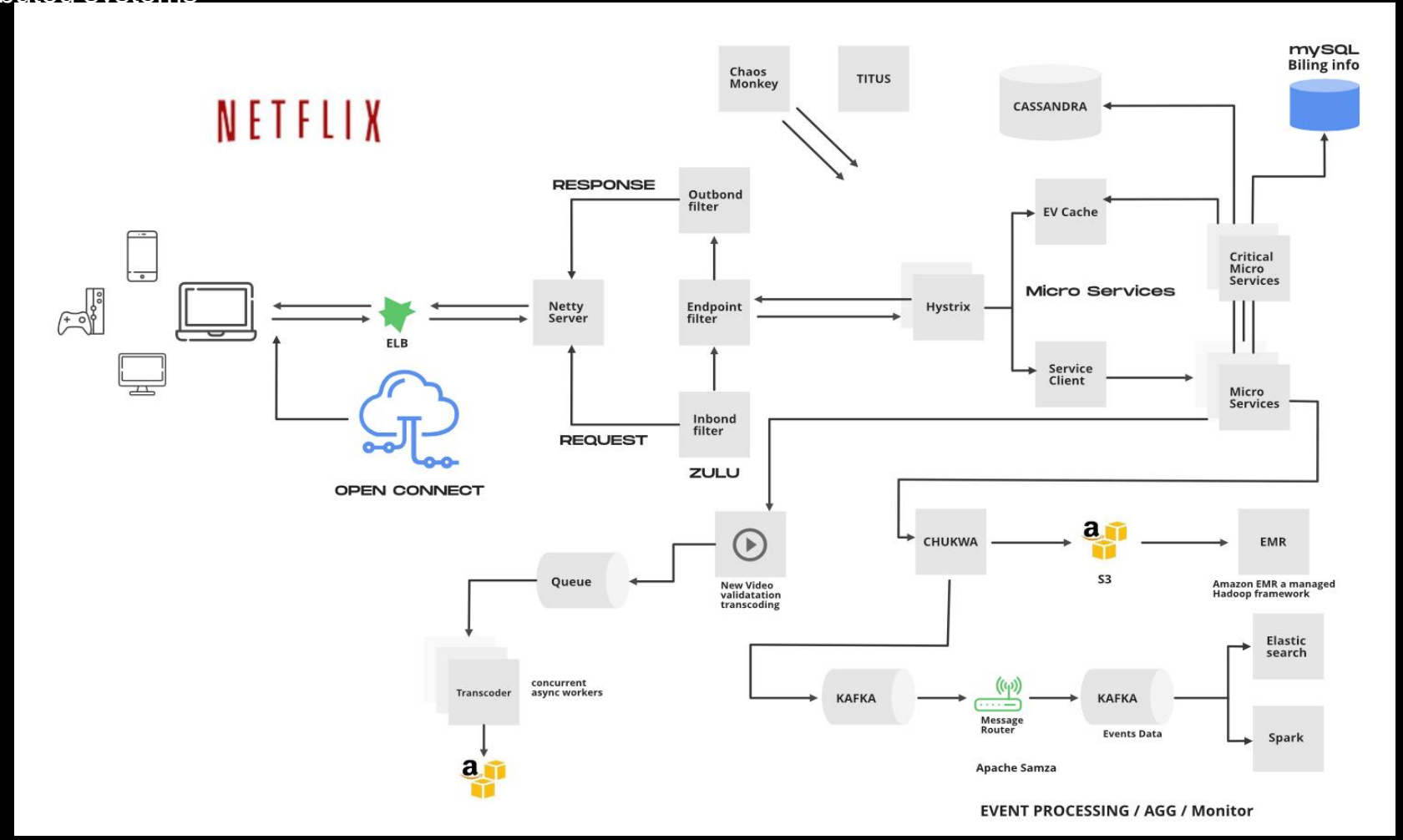


Image source: https://www.geeksforgeeks.org/system-design-netflix-a-complete-architecture/

SUMMARY

- Distributed database systems are more common than you may think
- Centralized database systems are mostly used in smaller companies with no outside customer access
  - Even then, there may be a distributed database system
- Often you may end with multiple database systems in one organization based on your needs

- Building a distributed database system is not so difficult…
- …Building a properly functioning distributed database system is

QUESTIONS
?


LUT
University