Data Access Proxy Topics

Hemendra Kathi, Derek Sadler, Reed Havens

What Is Data

Definition: Data in Computer Science refers to any piece of information that can be stored or processed by a computer system.

Examples of Data:

- Numbers
- Text
- Images
- Video
- And Much More

Data States:

- At Rest
- In Transit
- In Use

Data Categories:

- Structured Data
- Unstructured Data

Structured Data

Definition: Data that comes in a standardized format, has a structure following a data model, and is easy easy to access. This data is typically found in a database.

- Structured Data Accounts for roughly 20% of data world-wide

Characteristics:

- Information that is easy to query
- Is organized in a way that is easily understood
- Is stored in rows and columns that conforms to a database.
- Analyzing and processing the data is efficient.
- Similar entities can be grouped together

Common Structured Data Stores

Examples of Structured Data Includes:

- Relational Databases: Database that consists of rows with columns.
  - MySQL
  - PostgreSQL
  - SQLite
  - Oracle Database
- Spreadsheets: A sheet that contains information.
  - Microsoft Excel
  - Google Sheets
- XML and Json Files: Provides a standardized way to format data.
- Data Warehouse

Unstructured Data

Definition: Data that does not conform to any model and has no identifiable structure. The data has no organization to it and cannot be stored in any logical way.

- Unstructured Data accounts for the majority of data with it accounting for 80% of data

Characteristics:

- Lack of structure
- Does not follow any rules
- Comes in many forms
- Can come in high volume
- Difficult to store

Common Unstructured Data Stores

Examples of Unstructured data

- Email
- Text Files
- Social Media and Websites
- Mobile and communication data
- Media
- Scientific data
- Digital Surveillance Satellite Imagery

Where It can be found

- NoSQL databases
- Object storage
- File Stores
- Search Engines
- Content Management Systems
- Big Data Platforms

Databases I

Definition – organized collection of structured information

Components

- Data (rows and columns, etc.)
- Database management system (DBMS)
  - Interface between user and data which allows the user to interact with systems and data
  - E.g. SQL, Oracle Database, dBase, Microsoft Access
  - Performance monitoring, tuning, backup, recovery

Database vs spreadsheet

- Amount of data stored
- Access to the data
- Form of the data and possible manipulations

Databases II

Types of databases

- **Relational**
  - Rows and columns
- **Object oriented**
  - C++, Java
- **Distributed**
  - Multiple servers hosting different parts of DB
- **Warehouses**
  - Central repository made for fast analysis
- **NoSQL**
  - Non-relational DB
- **Graph**
  - Stores data in terms of relationships between entities
- **Cloud**
  - DBaaS

Databases III

Database challenges

- **Scaling** – can be difficult to manage increasingly large amounts of data; system may be unfit
- **Security** – data breaches are common; many vulnerable points
- **Demand –** high demand in short intervals can be difficult to support
- **Maintenance –** admins must constantly monitor performance and prevent problems
- **Access –** data is meant to be used but only by specific parties for specific purposes

Data Warehouses I

Definition – enterprise system used for the analysis and reporting of structured and semi-structured data from multiple sources

Uses

- Point of sale systems
- Real time decision making
- Predictive insights

Benefits

- Useful for ad hoc analysis, custom analysis
- Provides long-term picture of data for a company
- Consolidated data from many sources
- Specifically designed for analytics
- Adjust pricing
- Predict fraud

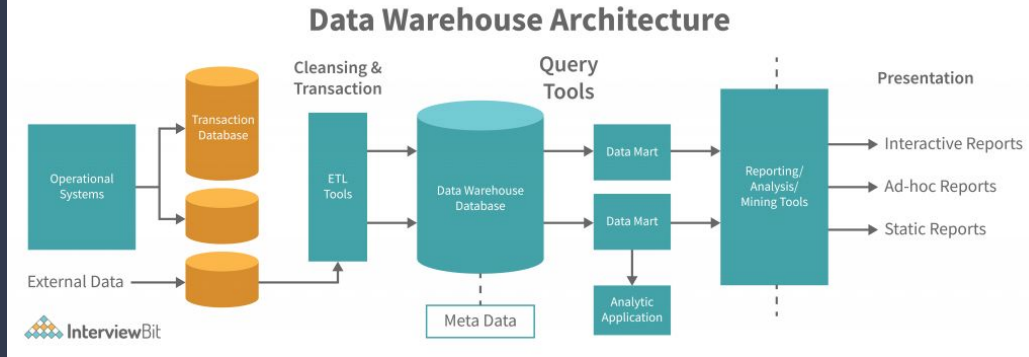
Data Warehouses II

Structure of Data Warehouse

- Top tier – front-end client, presentation of results through various tools
- Middle tier – analytical engine that access and uses the data
- Bottom tier – database server, where data is loaded and stored

Two types of memory storage

- Frequent items in fast-access storage
- Non-frequent in cheaper, slower storage



Data Lake I

Definition – centralized repository for structured, unstructured data at any scale

Uses

- Allows for new types of analytics
- Relational data from line of business, non-relational data from mobile, IOT
- Storage is unstructured

Data may be raw, uncurated

Users

- Data scientists, data developers, business analysts

Data Lake II

Handles any amount of data in real time

Components

- Data ingestion
- Storage
- Security
- Analytics
- Governance

Data Lake vs Data Warehouse

Characteristics	Data Warehouse	Data Lake
Data	Relational from transactional systems, operational databases, and line of business applications	Non-relational and relational from IoT devices, web sites, mobile apps, social media, and corporate applications
Schema	Designed prior to the DW implementation (schema-on-write)	Written at the time of analysis (schema-on-read)
Price/Performance	Fastest query results using higher cost storage	Query results getting faster using low-cost storage
Data Quality	Highly curated data that serves as the central version of the truth	Any data that may or may not be curated (ie. raw data)
Users	Business analysts	Data scientists, Data developers, and Business analysts (using curated data)
Analytics	Batch reporting, BI and visualizations	Machine Learning, Predictive analytics, data discovery and profiling

Data Lineage I

Definition: The process of tracking the flow of data over time.

- Data lineage uncovers the life cycle of data
- It aims to show the complete data flow, from start to finish
- How the data was transformed, what changed, and why

It allows companies to:

- Track errors in data processes
- Implement process changes with lower risk
- Perform system migrations with confidence

Data Lineage II

Use Cases:

Data Modeling:Data lineage helps to accurately reflect changes over time through data model diagrams, highlighting new or outdated connections or tables.

Data Migration:provides a view of how this data has progressed through the organization, it assists teams in planning for these system migrations or upgrades, expediting the overall transition to the new storage environment.

Impact Analysis: Data lineage tools can provide visibility into the impact of specific business changes, such as any downstream reporting

Data Mesh I

Definition: A data mesh is a decentralized data architecture that organizes data by a specific business domain.

- Give individual teams control over datasets.
- Focus on data as a product
- Ensure data governance and security

Tries to solve:

- Lack of Ownership
- Lack of quality
- Organizational Scaling

Data Mesh II

Four Fundamental Principles of Data Mesh

Domain Ownership: Each domain is responsible for creating, managing, storing, and sharing the data it creates without relying on a central data team

Data as a Product: Instead of treating data as a by-product of business processes, it should be seen as the product itself.

Self Serve Data Platform: For domain teams to be fully autonomous and manage their data products end-to-end, self-serve data infrastructure must be in place.

Federated Governance: The domain data owners will follow a set of federal/global data governance rules, while retaining their autonomy.

Data Mesh III

Benefits:

- Greater autonomy and control over your data, leading to faster decision-making
- Product thinking gets embedded everywhere
- Easier data discovery and accessibility
- Greater scalability of data systems with autonomous data domains and teams
- Better data quality when the team creating data is in charge of managing it and extracting value from it
- Interoperability across data domains
- Better regulatory compliance and data security

Data Lake House

Definition:

A data lake house is an open data management architecture that combines the flexibility and scalability benefits of a data lake with the data structures and data management features of a data warehouse.

4 Elements of Data Lakehouse

1)*Atomicity* means that when processing transactions, either the entire transaction succeeds or none of it does. This helps prevent data loss or corruption in case of an interruption in a process.

2)*Consistency* makes sure that transactions take place in predictable, consistent ways. It ensures all data is valid according to predefined rules, maintaining integrity of the data.

Data Lake House II

Elements of Data Lakehouse(Contd.)

3) *Isolation* guarantees that no transaction can be affected by any other transaction in the system until it is completed. This makes it possible for multiple parties to read and write from the same system at the same time without them interfering with one another.

4) *Durability* ensures that changes made to the data in a system persist once a transaction is complete, even if there is a system failure. Any changes that result from a transaction are stored permanently.

Data Lake House III

Benefits:

- It eliminates simple extract, transfer, and load (ETL) jobs because query engines are connected directly to the data lake.
- It reduces data redundancy with a single tool used to process data, instead of managing data on multiple platforms with multiple tools.
- It enables direct connection to multiple BI and analytics tools.
- It makes data governance easier because sensitive data does not have to be moved from one data pool to another and can be managed from one point.

Thank you!