Advanced Databases

- Suchitra Patil

Data Lake

“If you think of a Data Mart as a store of bottled water, cleansed and packaged and structured for easy consumption, the Data Lake is a large body of water in a more natural state. The contents of the Data Lake stream in from a source to fill the lake, and various users of the lake can come to examine, dive in, or take samples.”

- James Dixon, founder and former CTO of Pentaho

What is a Data Lake?

- A data lake is a central location that handles a massive volume of **data in its native, raw format** and organizes **large volumes** of **highly diverse** data. Whether data is structured, unstructured, or semi-structured, it is loaded and stored as-is. Usually blobs or files

- Compared to a **hierarchical** data warehouse that saves data in files or folders, a data lake uses a **flat architecture** to store it

- A data lake is usually a single store of data including raw copies of source system data, sensor data, social data etc., and transformed data used for tasks such as reporting, visualization, advanced analytics and machine learning

What is a Data Lake?

Before Data Lake

- Relational Database
  - Simple, Reliable
  - Highly Structured Data
  - Not suitable for big data
- Rise of Internet
  - Large volume of data
  - Multiple Databases
  - Data Silos
- Data Warehouse
  - Collection of relational databases under a single umbrella
  - With time became available in the cloud
  - Integrated Data ready for analytical queries

Before Data Lake

- Data Warehouse downsides
  - Inability to store unstructured, raw data
  - Expensive, proprietary hardware and software
  - Difficulty scaling due to the tight coupling of storage and compute power

- Apache HADOOP™
  - a collection of open source software for big data analytics
  - capability to analyse raw data, structured, semi-structured and unstructured

Data Warehouse & Data Lake

Characteristics	Data Warehouse	Date 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)

Data Warehouse & Data Lake

Characteristics	Data Warehouse	Date Lake
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)

Data Warehouse & Data Lake

Characteristics	Data Warehouse	Date Lake
Analytics	Batch reporting, BI and visualizations	Machine Learning, Predictive analytics, data discovery and profiling
Users	Business analysts	Data scientists, Data developers, and Business analysts (using curated data)

Analysis Paradigms

Application Database	Data Warehouse	Date Lake
1. Relational Data Structuring 2. Ingest Data 3. Analyze	1. Report Data Structuring 2. Ingest Data 3. Analyze	1. Ingest Data 2. Analyze 3. Define Data Structure

Advantages of Developing a Data Lake

- Ability to collect all types of structured and unstructured data in a data lake
- More flexibility
- Ability to store raw data—you can refine it as your understanding and insight improves
- Unlimited ways to query the data
- Application of a variety of tools to gain insight into what the data means
- Ability to derive value from all types of data
- Elimination of data silos
- Democratized access to information via a unique, centralized view of data across the organization

Data Lake Architecture

- The architecture of a data lake refers to the features that are included within a data lake to make it easier to work with that data

- Even though data lakes are unstructured, it is still important to ensure that they offer the functionality and design features that your organization requires in order to easily interact with the data that they house

- A data lake should present three key characteristics
  - A single shared repository of data
  - Includes orchestration and job scheduling capabilities
  - Has a collection of workflows to execute

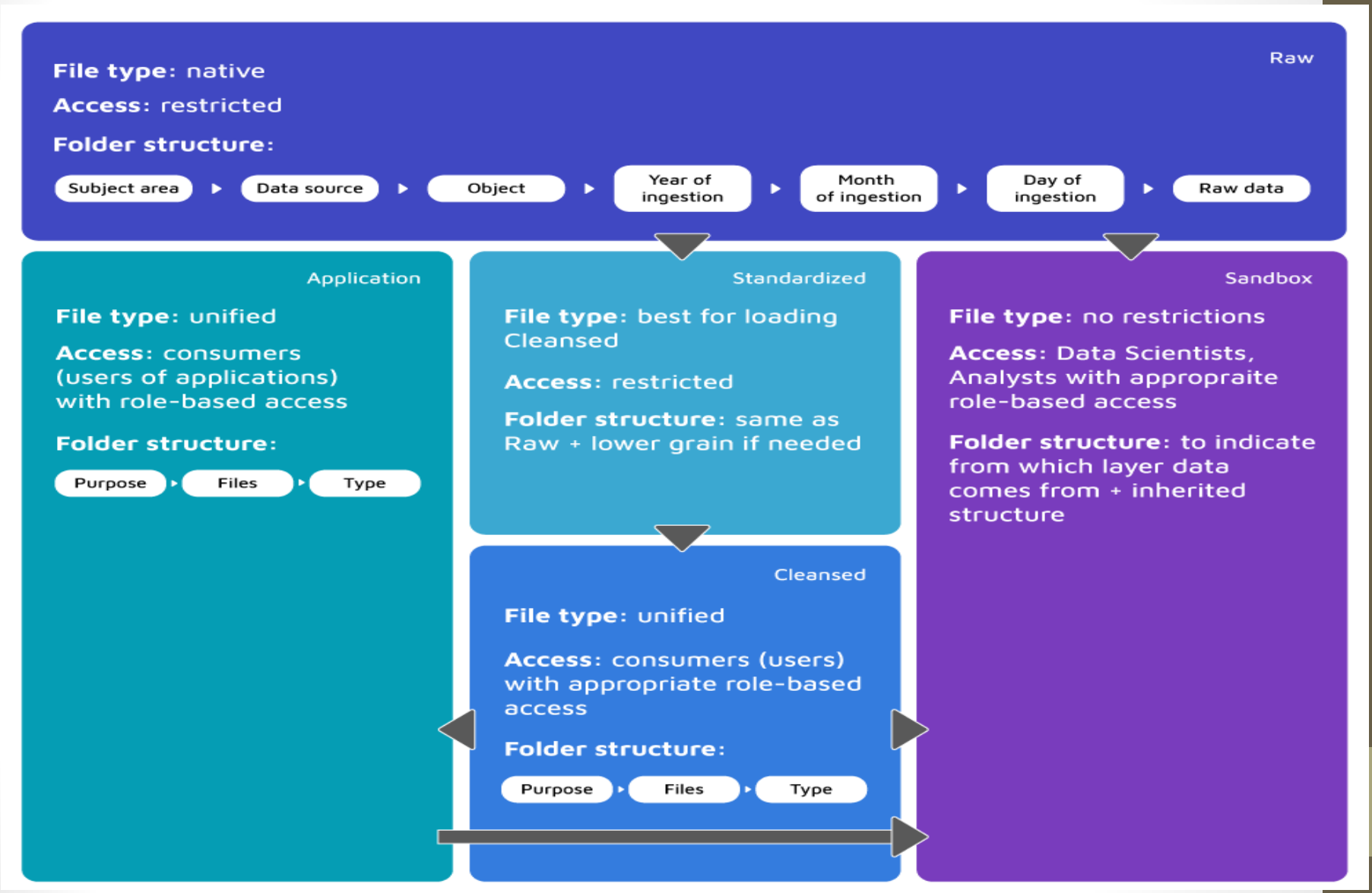
Data Lake Architecture

- Various types of sources for data
  - Operational data
    - ✓ sales, finances, inventory etc.
  - Auto-generated data
    - ✓ IoT devices, logs etc.
  - Human-generated data
    - ✓ social media posts, emails, web content etc. either coming from inside, or from outside the organization.
- All data stored in native format

Data Lake Architecture

- Though Data Lake is single repository, for flexibility it can be divided in following layers
  - Raw
  - Standardized
  - Cleansed
  - Application
  - Sandbox

Data Lake Layers



Data Lake Architecture : Layers

- Raw data layer
  - Also called the Ingestion Layer/Landing Area
  - Main objective is to ingest data into Raw as quickly and as efficiently as possible
  - **Data remains in native format, no transformations** are applied
  - No overriding allowed : **Duplicates and versions needs to be handled**
  - Raw needs to be arranged in the folders
  - Data here is **not ready to be used**, it requires a lot of knowledge in terms of appropriate and relevant consumption

Data Lake Architecture : Layers

- Standardized data layer
  - may be considered as optional in most implementations
  - Main **objective is to improve performance in data transfer** from Raw to Curated
  - Both daily transformations and on-demand loads are included
  - Format is chosen which fits best for cleansing
  - Structure same as Raw but may be partitioned to lower grain if needed

Data Lake Architecture : Layers

- Cleansed data layer
  - Also called Curated Layer/Conformed Layer
  - **Data is transformed into consumable data sets and it may be stored in files or tables**
  - Purpose and structure of data is already known which is cleansed and transformed
  - Denormalization and consolidation of different objects is common
  - Most complex part of the whole Data Lake solution
  - Usually, end users are granted access only to this layer

Data Lake Architecture : Layers

- Application data layer
  - Also called Trusted Layer/Secure Layer/Production Layer
  - Sourced from Cleansed data Layer and **enforced with any needed business logic**
  - **Anything specific to the application**; like row level security, surrogate keys shared among applications, machine learning models calculated on Data Lake etc.; will need this layer
  - The structure of the data remains the same, as in Cleansed

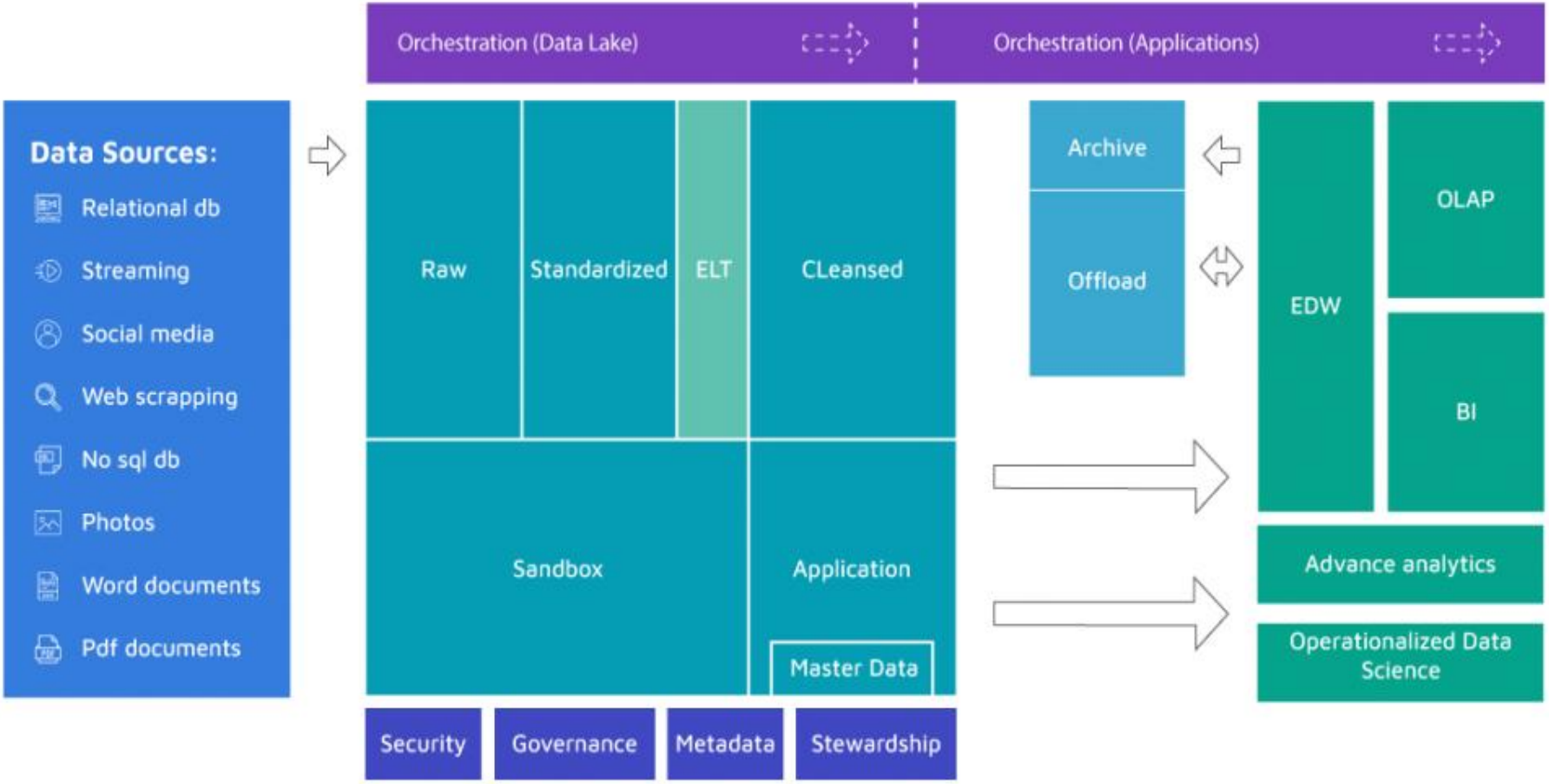
Data Lake Architecture : Layers

- Sandbox data layer
  - Another layer that might be **considered optional**
  - Meant for **advanced analysts' and data scientists**' work
  - Experiments can be carried out for searching patterns or correlations
  - For any source from internet enriching the data, Sandbox is the proper place

Data Lake Architecture

- Key Components
  - Security
  - Governance
  - Metadata
  - Stewardship
  - Master Data
  - Archive
  - Offload
  - Monitoring/Orchestration and ELT Processes

Data Lake Architecture



Data Lake Architecture : Components

- Security
  - Important aspect even though not exposed to broad audience, especially during the initial phase and architecting
  - Not like relational databases, with an artillery of security mechanisms
- Governance
  - Monitoring and logging (or lineage) operations will become crucial at some point for measuring performance and adjusting the Data Lake

Manjitsing K. Valvi

Data Lake Architecture : Components

- Metadata
  - all the schemas, reload intervals, additional descriptions of the purpose of data, with descriptions on how it is meant to be used
- Stewardship
  - Depending the scale of need responsibility is delegated either to separate teams of owners(users)
- Master Data
  - An essential part of serving ready-to-use data
  - Either store it in Data Lake or reference it while executing ELT processes

Data Lake Architecture : Components

- Archive
  - Data Lakes are often used to keep some archive data that comes originally from DWH
  - might face some performance and storage related problems if there is relational DWH solution
- Offload
  - In case there is DWH solution, Offload might be used in order to offload some time/resource consuming ETL processes to your Data Lake, which might be cheaper and faster

Data Lake Architecture : Components

- Monitoring/Orchestration and ELT Processes
  - Tool is needed for orchestration of flow as data is being pushed from Raw to Sandbox and Application layers
  - Tools or some additional resources for the transformation

consider an example of a data lake implementation for a fictional e-commerce company called "E-CommerceX.“

E-CommerceX operates a large online marketplace where customers can purchase a wide range of products, including electronics, clothing, home goods, and more.
The company collects various types of data from its operations, including:
Transaction data: Information about purchases made by customers, including product details, prices, quantities, and payment methods.
Customer data: Details about customers, such as demographics, contact information, purchase history, and browsing behavior.
Website analytics: Data on website traffic, user interactions, clickstream behavior, page views, bounce rates, and conversion rates.

Product data: Information about product catalogs, including product names, descriptions, categories, and attributes.
Inventory data: Details about available inventory, stock levels, replenishment schedules, and supplier information.
Marketing data: Insights from marketing campaigns, including ad impressions, clicks, conversions, and ROI metrics.
Reviews and feedback: Customer reviews, ratings, feedback, and sentiment analysis from social media platforms or review websites.
To leverage this data effectively for business insights and decision-making, E-CommerceX decides to implement a data lake architecture using cloud-based storage and processing technologies.

Here's how the data lake is set up:

Storage: E-CommerceX leverages a cloud-based storage solution, such as Amazon S3 or Google Cloud Storage, to store its raw data in its native format. Each type of data (transaction, customer, website analytics, etc.) is stored in separate folders or buckets within the data lake.

Ingestion: Data pipelines are set up to ingest data from various sources into the data lake. For example, transaction data might be ingested from the company's e-commerce platform database using batch processing, while website analytics data might be streamed in real-time from web server logs using technologies like Apache Kafka.

Schema on Read: The raw data is stored as-is, without any transformation or schema enforcement at write time. Instead, schema enforcement and data processing occur dynamically at query time. This allows flexibility in data exploration and analysis without the need for predefined schemas.

Analytics: Data analysts, data scientists, and business users can access the data lake using a variety of tools and technologies. For SQL-based querying and analysis, tools like Amazon Athena or Google BigQuery can be used. For advanced analytics and machine learning, platforms like Apache Spark or TensorFlow can be integrated with the data lake.

Data Governance and Security: E-CommerceX implements robust data governance policies and access controls to ensure data privacy, security, and compliance with regulations such as general data protection regulation(GDPR). Role-based access control (RBAC) is used to manage permissions, and data encryption is employed to protect sensitive information.

Scalability and Cost Efficiency: The cloud-based data lake architecture offers scalability and cost efficiency, allowing E-CommerceX to store and analyze large volumes of data without upfront infrastructure investments. The pay-as-you-go pricing model ensures that the company only pays for the resources it consumes.

With the data lake in place, E-CommerceX can now perform a wide range of analytics and derive valuable insights from its data. For example:

1. Data analysts can analyze transaction data to identify trends in customer purchasing behavior and product popularity.
2. Data scientists can build recommendation models using customer data and product information to personalize recommendations for users.
3. Marketing teams can analyze website analytics data to optimize marketing campaigns and improve conversion rates.
4. Operations teams can use inventory data to optimize supply chain management and ensure adequate stock levels.
5. Overall, the data lake enables E-CommerceX to leverage its data assets more effectively, drive business growth, and deliver a better experience for its customers.

References

- https://www.dataversity.net/brief-history-data-lakes/#
- https://www.xplenty.com/blog/data-lake-architecture-guide/
- https://aws.amazon.com/big-data/datalakes-and-analytics/what-is-a-data-lake/
- https://lingarogroup.com/blog/data-lake-architecture/
- https://databricks.com/discover/data-lakes/history