Data Centered Architecture

Intro:

Data-centered architecture is a software architecture style that focuses on data and how it is accessed and manipulated by different components. The main purpose of this style is to achieve integrity and consistency of data.

Working:

Data-centered architecture consists of a central data store or repository and a collection of data accessors or agents that operate on the data store. The data store is the only means of communication among the data accessors. There are two types of data-centered architecture repository and blackboard.

Data Sources: Data-centered architecture starts with identifying the various sources of data within an organization. These sources can include databases, legacy systems, external APIs, data streams, files, and more.

2. Data Integration: Once the data sources are identified, the architecture focuses on integrating the data from these disparate sources. Data integration involves techniques such as data extraction, transformation, and loading (ETL), data replication, data virtualization, or the use of data integration platforms. The goal is to bring the data together in a unified and coherent manner.

3. Data Storage: Data-centered architecture provides a structured approach to data storage. It encompasses decisions regarding the storage technologies, such as relational databases, NoSQL databases, data warehouses, data lakes, or a combination of these. The choice depends on factors like data volume, data structure, performance requirements, and scalability needs.

4. Data Management: Data management involves activities like data modeling, data governance, data quality management, and metadata management. Data modeling establishes the structure and relationships between different data elements. Data governance ensures that data is properly managed, protected, and compliant with regulations. Data quality management focuses on maintaining data accuracy, consistency, and completeness. Metadata management deals with capturing and organizing metadata, which provides information about the data attributes, sources, and context.

5. Data Access: Data-centered architecture enables efficient and controlled access to the stored data. This can be achieved through the use of data APIs, query languages (such as SQL), or custom data access layers. The architecture should support different access patterns, such as read-intensive or write-intensive operations, real-time queries, batch processing, or analytics.

6. Data Processing and Analytics: Data-centered architecture facilitates data processing and analytics capabilities. It enables performing operations like data transformation, aggregation, filtering, and enrichment. This can be achieved through batch processing frameworks (e.g., Hadoop, Spark) or real-time streaming frameworks (e.g., Apache Kafka, Apache Flink). Analytics tools and technologies can be integrated to extract insights, perform data mining, machine learning, or generate reports and dashboards.

7. Data Security and Privacy: Data-centered architecture incorporates robust security measures to protect data from unauthorized access, breaches, or data loss. It includes encryption, access controls, authentication mechanisms, and monitoring systems. Additionally, privacy considerations are addressed to ensure compliance with data protection regulations and best practices.

8. Scalability and Performance: The architecture is designed to handle large volumes of data and support scalability. This can be achieved through horizontal scaling (adding more resources or nodes) or vertical scaling (upgrading existing resources). The architecture also focuses on optimizing data processing, storage, and retrieval performance to meet performance requirements.

9. Data Lifecycle Management: Data-centered architecture considers the entire data lifecycle, including data ingestion, storage, processing, archiving, and disposal. It includes strategies for data retention, data archival, backup and recovery, data versioning, and data purging based on business and regulatory requirements.

10. Integration with Applications: Finally, data-centered architecture enables seamless integration of data with various applications, systems, or services. It supports data sharing and interoperability across different applications, ensuring consistent and accurate data across the organization.

- Data is the central focus of the architecture

- Three key layers: storage, processing, and presentation

- Storage layer captures and persists data

- Processing layer transforms and analyzes data

- Presentation layer provides access and visualization

- In a Data-Centered Architecture, data is treated as a first-class citizen, and all components revolve around it.

- The architecture consists of three key layers: the storage layer, the processing layer, and the presentation layer.

- The storage layer is responsible for capturing and persisting data from various sources, such as databases, data lakes, and streaming platforms.

- The processing layer performs transformations, aggregations, and analytics on the data, using technologies like distributed computing, big data frameworks, and machine learning.

- The presentation layer provides interfaces and tools for users to access and visualize the data, enabling them to derive insights and make informed decisions.

Example:

One example of a software product that uses data-centered architecture is \*\*Salesforce CRM\*\*³.

- Salesforce CRM is a cloud-based platform that helps businesses manage their customer relationships, sales, marketing, analytics, and more³.

- Salesforce CRM uses a central database to store all the customer data and a set of applications to access and manipulate the data³.

- Salesforce CRM also allows users to create custom applications using its platform as a service (PaaS) feature called Force.com³.

Apache Kafka as an example of Data-Centered Architecture

- Distributed streaming platform

- Handles high volumes of data streams in real-time

- Acts as a central data hub

- Enables efficient data processing and integration