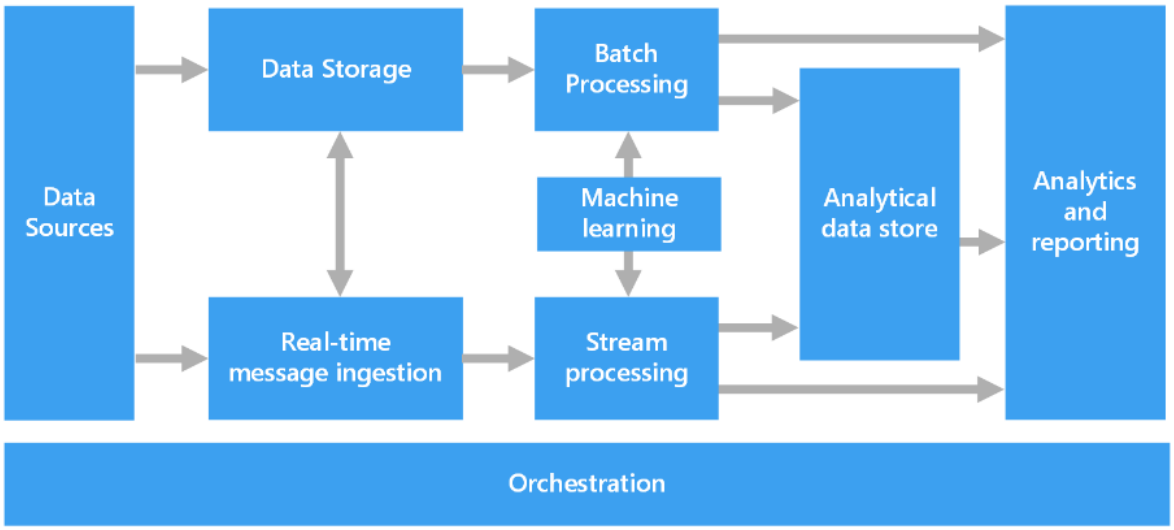
Big data architectures

A big data architecture is designed to handle the ingestion, processing, and analysis of data that is too large or complex for traditional database systems. The threshold at which organizations enter into the big data realm differs, depending on the capabilities of the users and their tools.

Components of a big data architecture



- **Data sources:** The obvious starting point of all big data solutions data sources may be static files produced by applications (web server log files), application data sources (relational databases), or real-time data sources (IoT devices).

- **Data storage:** Often referred to as a data lake, a distributed file store holds bulks of large files in different formats, which are subsequently used for batch processing operations.

- **Batch processing:** In order to make large datasets analysis-ready, batch processing carries out the filtering, aggregation, and preparation of the data files through long-running batch jobs.

- **Message ingestion:** This component of the big data architecture includes a way to capture and store messages from real-time sources for stream processing.

- **Stream processing:** Another preparatory step before data analytics, stream processing filters and aggregates the data after capturing real-time messages.

- **Analytical data store:** After preparing the data for analytics, most big data solutions serve the processed data in a structured format for further querying using analytical tools. The analytical data store that serves these queries can either be a Kimball-style relational data warehouse or a low-latency NoSQL technology.

- **Analysis and reporting:** One of the critical goals of most big data solutions, data analysis and reporting provides insights into the data. For this purpose, the big data

architecture may have a data modelling layer, support self-service BI, or even incorporate interactive data exploration.

- **Orchestration:** An orchestration technology can automate the workflows involved in repeated data processing operations, such as transforming the data source, moving data between sources and sinks, loading the processed data into an analytical data store, and final reporting.

Types of Big Data Architecture

Lambda Architecture

Lambda architecture is a hybrid approach that processes both batch (static) data and real-time processing data. It is employed for solving the problem of computing arbitrary functions. This deployment model approach aims to reduce latency and negligible errors while maintaining accuracy.

Kappa Architecture

An alternative to Lambda architecture is Kappa Architecture. The idea of this architecture is also to process both the real-time streaming and batch processing data. The Kappa architecture reduces the additional cost that comes up with the Lambda architecture by replacing the data sources medium with the message queuing.

Big Data Tools and Techniques

Big data tools are categorized into four buckets according to their utility:

- Massively Parallel Processing (MPP)
- No-SQL Databases
- Distributed Storage and Processing Tools
- Cloud Computing Tools

Benefits of Big Data Architecture

1. Parallel computing for high performance
To process large data sets quickly, big data architectures use parallel computing, in which multiprocessor servers perform numerous calculations at the same time.

2. Elastic scalability
Big Data architectures can be scaled horizontally, enabling the environment to be adjusted to the size of each workload.

3. Freedom of choice
The marketplace offers many solutions and platforms for use in Big Data architectures, such as Azure managed services, MongoDB Atlas, and Apache technologies.

4. Interoperability with related systems
You can create integrated platforms across different types of workloads, leveraging Big Data architecture components for IoT processing

Big Data Architecture Challenges

1. Security
Big data of the static variety is usually stored in a centralized data lake. Robust security is required

2. Complexity
A Big Data architecture typically contains many interlocking moving parts.

3. Evolving technologies
It can be daunting, as many Big Data technologies, practices, and standards are relatively new and still in a process of evolution.

4. Specialized skill sets
Big Data APIs built on mainstream languages are gradually coming into use.