# APACHE NIFI

## What is Apache NIFI ?

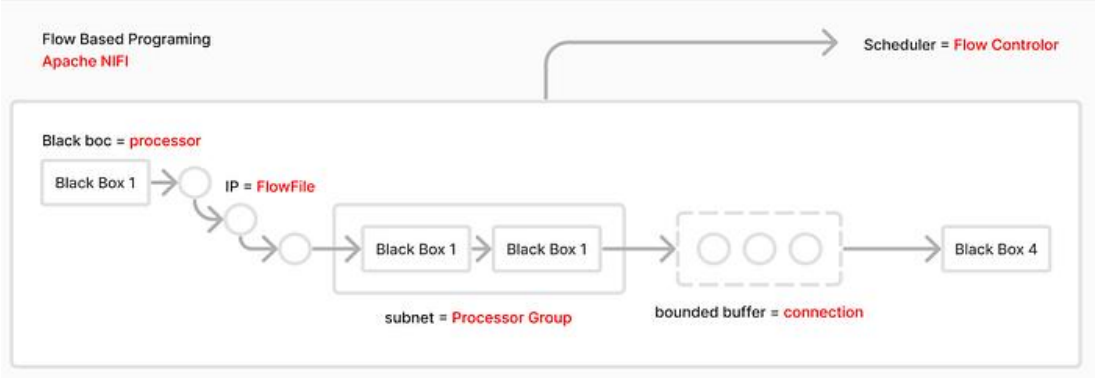
- Apache NiFi is a real-time data ingestion and distribution platform designed for ease of use and high reliability.
- At its core, NiFi:
  - Automates the flow of data between diverse systems
  - Transforms and cleans data on the fly
  - Supports many formats (JSON, CSV, Avro, geo-data, social feeds) and protocols (SFTP, HDFS, Kafka, HTTP)
  - Scales from a single node to a zero-leader clustered environment
  - Secure Data Flow with SSL, user authentication, and data encryption
- With a drag-and-drop web UI and a library of over 300 processors, you can visually build and monitor your data pipelines without writing a single line of code.

## Why Should We Use NIFI

- Remember the four Vs of Big Data — Volume, Velocity, Variety, and Veracity.
- NiFi tackles them all:
- Volume & Velocity: Parallel processing and configurable back-pressure let NiFi handle massive, high-speed data streams.
- Variety: Out-of-the-box processors support dozens of data formats and protocols, so you can connect almost anything.
- Veracity: Built-in data provenance and schema validation ensure data quality and auditability.
- Reliability: State management, checkpointing, and replay mechanisms mean no message is ever lost.

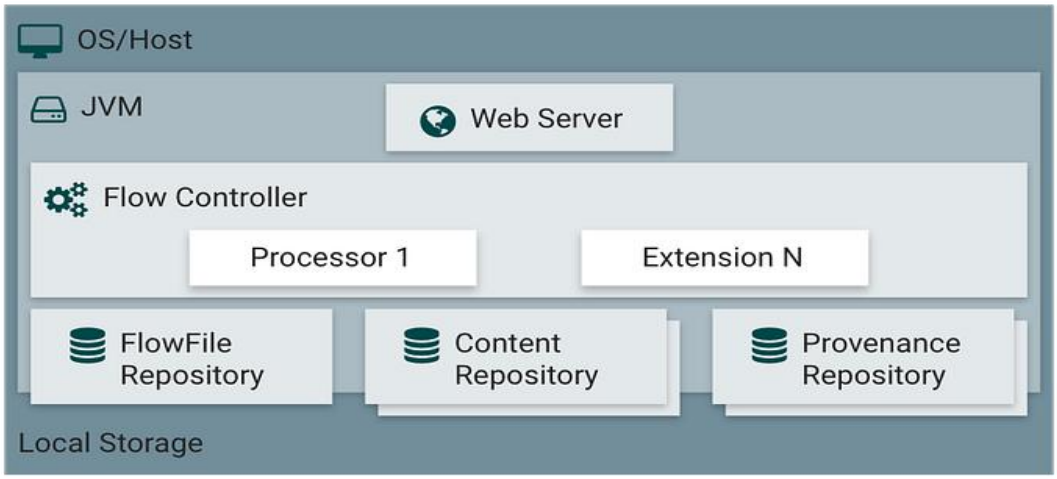
## Core Concepts

- NiFi's fundamental design concepts are closely aligned with the principles of Flow-Based Programming (FBP).
- NiFi is inspired by Flow-Based Programming (FBP), where applications are networks of reusable, black-box processes connected by data streams. Key NiFi components include:

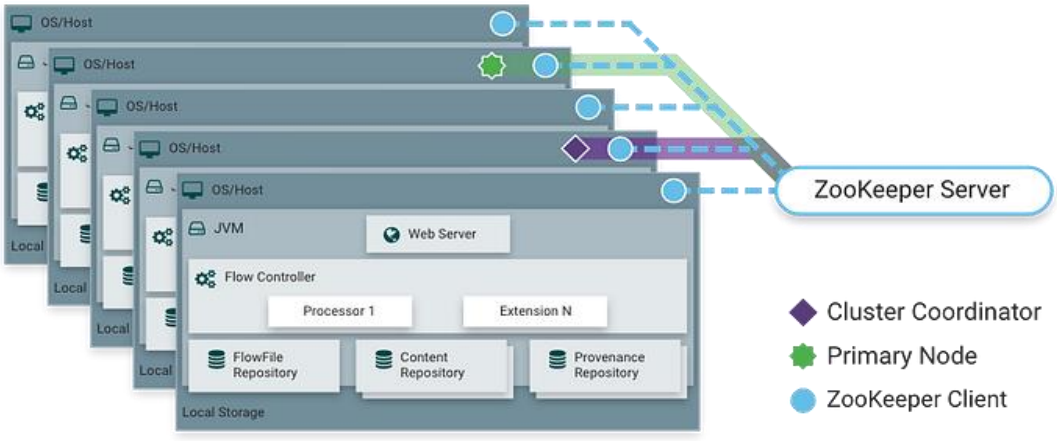


- **FlowFile**: A data packet consisting of content and metadata (attributes).
- **Processor:** The workhorse that ingests, transforms, routes, or writes data. Each processor has relationships for success, failure, and original FlowFiles.
- **Connection:** Buffers between processors that act like queues, smoothing out differences in processing speed.
- **Process Group**: A nested collection of processors and connections — think of it as a reusable sub-workflow.
- **Flow Controller:** The brain that schedules and manages processor threads.

## NiFi Architecture




- NiFi runs on the JVM and comprises:
- Web Server: Hosts the HTTP API and UI.
- Flow Controller: Allocates resources and orchestrates the dataflow.
- Extensions: Processors, controller services, and reporting tasks.
- Repositories:
- FlowFile Repository: Tracks the state of each FlowFile.
  Content Repository: Stores the actual data bytes.
  Provenance Repository: Logs every event (ingest, route, modify) for auditing.



- Starting with the NiFi 1.0 release, a Zero-Leader Clustering paradigm has been adopted.
- In this setup, each node in a NiFi cluster performs the same tasks on different sets of data.
- Apache ZooKeeper facilitates the election of a single node as the Cluster Coordinator, ensuring that failover is managed automatically.
- All nodes in the cluster send their heartbeat and status information to the Cluster Coordinator, who is responsible for managing the connection and disconnection of nodes.
- Additionally, each cluster includes one Primary Node, also elected by ZooKeeper.
- As a DataFlow manager, you can interact with the NiFi cluster via the user interface (UI) of any node. Any changes you make through the UI are replicated across all nodes in the cluster, providing multiple points of entry for managing the data flow.

## Apache NiFi- General Features

- The general features of Apache NiFi are as follows −
- Apache NiFi provides a web-based user interface, which provides seamless experience between design, control, feedback, and monitoring.
- It is highly configurable. This helps users with guaranteed delivery, low latency, high throughput, dynamic prioritization, back pressure and modify flows on runtime.
- It also provides data provenance module to track and monitor data from the start to the end of the flow.
- Developers can create their own custom processors and reporting tasks according to their needs.
- NiFi also provides support to secure protocols like SSL, HTTPS, SSH and other encryptions.
- It also supports user and role management and also can be configured with LDAP for authorization.

## Apache NiFi -Key Concepts

- The key concepts of Apache NiFi are as follows −
- Process Group − It is a group of NiFi flows, which helps a userto manage and keep flows in hierarchical manner.
- Flow − It is created connecting different processors to transfer and modify data if required from one data source or sources to another destination data sources.
- Processor − A processor is a java module responsible for either fetching data from sourcing system or storing it in destination system. Other processors are also used to add attributes or change content in flowfile.
- Flowfile − It is the basic usage of NiFi, which represents the single object of the data picked from source system in NiFi. NiFiprocessormakes changes to flowfile while it moves from the source processor to the destination. Different events like CREATE, CLONE, RECEIVE, etc. are performed on flowfile by different processors in a flow.

- Event − Events represent the change in flowfile while traversing through a NiFi Flow. These events are tracked in data provenance.
- Data provenance − It is a repository.It also has a UI, which enables users to check the information about a flowfile and helps in troubleshooting if any issues that arise during the processing of a flowfile.

## Apache NiFi Advantages

- Apache NiFi enables data fetching from remote machines by using SFTP and guarantees data lineage.
- Apache NiFi supports clustering, so it can work on multiple nodes with same flow processing different data, which increase the performance of data processing.
- It also provides security policies on user level, process group level and other modules too.
- Its UI can also run on HTTPS, which makes the interaction of users with NiFi secure.
- NiFi supports around 188 processors and a user can also create custom plugins to support a wide variety of data systems.

## Apache NiFi Disadvantages

- When node gets disconnected from NiFi cluster while a user is making any changes in it, then the flow.xml becomes invalid.A node cannot connect back to the cluster unless admin manually copies flow.xml from the connected node.
- Apache NiFi have state persistence issue in case of primary node switch, which sometimes makes processors not able to fetch data from sourcing systems.