# 12.1

**● Explain the need of Flume**

Flume lets Hadoop users ingest high-volume streaming data into HDFS for storage. Specifically, Flume allows users to:

1. Stream data:

Ingest streaming data from multiple sources into Hadoop for storage and analysis

2. Insulate systems:

Buffer storage platform from transient spikes, when the rate of incoming data exceeds the rate at which data can be written to the destination

3. Guarantee data delivery:

Flume NG uses channel-based transactions to guarantee reliable message delivery. When a message moves from one agent to another, two transactions are started, one on the agent that delivers the event and the other on the agent that receives the event. This ensures guaranteed delivery semantics

4. Scale horizontally:

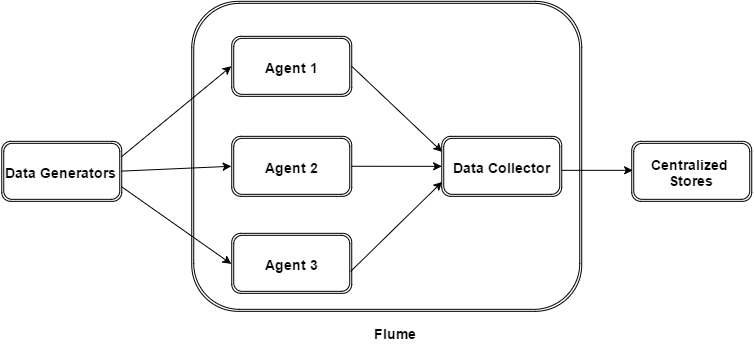
To ingest new data streams and additional volume as needed

Enterprises use Flume’s powerful streaming capabilities to land data from high-throughput streams in the Hadoop Distributed File System (HDFS). Typical sources of these streams are application logs, sensor and machine data, geo-location data and social media. These different types of data can be landed in Hadoop for future analysis using interactive queries in Apache Hive. Or they can feed business dashboards served ongoing data by Apache HBase.

In one specific example, Flume is used to log manufacturing operations. When one run of product comes off the line, it generates a log file about that run. Even if this occurs hundreds or thousands of times per day, the large volume log file data can stream through Flume into a tool for same-day analysis with Apache Storm or months or years of production runs can be stored in HDFS and analyzed by a quality assurance engineer using Apache Hive.

**● Explain the working of Flume and its components in brief.**

The following illustration depicts the basic architecture of Flume. As shown in the illustration, **data generators** (such as Facebook, Twitter) generate data which gets collected by individual Flume **agents** running on them. Thereafter, a **data collector** (which is also an agent) collects the data from the agents which is aggregated and pushed into a centralized store such as HDFS or HBase.



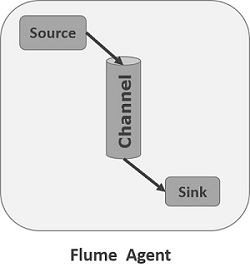
Flume Event

An **event** is the basic unit of the data transported inside **Flume**. It contains a payload of byte array that is to be transported from the source to the destination accompanied by optional headers. A typical Flume event would have the following structure −



Flume Agent

An **agent** is an independent daemon process (JVM) in Flume. It receives the data (events) from clients or other agents and forwards it to its next destination (sink or agent). Flume may have more than one agent. Following diagram represents a **Flume Agent**



As shown in the diagram a Flume Agent contains three main components namely, **source**, **channel**, and **sink**.

Source

A **source** is the component of an Agent which receives data from the data generators and transfers it to one or more channels in the form of Flume events.

Apache Flume supports several types of sources and each source receives events from a specified data generator.

**Example** − Avro source, Thrift source, twitter 1% source etc.

Channel

A **channel** is a transient store which receives the events from the source and buffers them till they are consumed by sinks. It acts as a bridge between the sources and the sinks.

These channels are fully transactional and they can work with any number of sources and sinks.

**Example** − JDBC channel, File system channel, Memory channel, etc.

Sink

A **sink** stores the data into centralized stores like HBase and HDFS. It consumes the data (events) from the channels and delivers it to the destination. The destination of the sink might be another agent or the central stores.

**Example** − HDFS sink

**Note** − A flume agent can have multiple sources, sinks and channels. We have listed all the supported sources, sinks, channels in the Flume configuration chapter of this tutorial.

Additional Components of Flume Agent

What we have discussed above are the primitive components of the agent. In addition to this, we have a few more components that play a vital role in transferring the events from the data generator to the centralized stores.

Interceptors

Interceptors are used to alter/inspect flume events which are transferred between source and channel.

Channel Selectors

These are used to determine which channel is to be opted to transfer the data in case of multiple channels. There are two types of channel selectors −

* **Default channel selectors** − these are also known as replicating channel selectors they replicates all the events in each channel.
* **Multiplexing channel selectors** − these decides the channel to send an event based on the address in the header of that event.

Sink Processors

These are used to invoke a particular sink from the selected group of sinks. These are used to create failover paths for your sinks or load balance events across multiple sinks from a channel.