**12.1.1. Explain the need of Flume.**

Flume is a distributed, reliable, and available service for efficiently collecting, aggregating, and moving large amounts of log data. It has a simple and flexible architecture based on streaming data flows.

Flume is designed for high-volume ingestion into Hadoop of event-based data.

**Stream data:**

Flume ingests streaming data from multiple sources into Hadoop for storage and analysis.

**Insulate systems:**

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

**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.

**Scale horizontally:**

Flume can ingest new data streams and additional volume as needed.

 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 analysed by a quality assurance engineer using Apache Hive.

**12.1.2 Working of Flume:**

**COMPONENTS:**

Flume consists of

* **Event**
* **Source**
* **Sink**
* **Channel**
* **Agent**
* **Client**

**Event:**

Event is a singular unit of data (typically a single log entry) that is transported by Flume from point of origination to point of destination.

A typical Flume event would have a header and byte payload

**Source:**

Source is the entity through which data enters into Flume. Sources either actively poll for data or passively wait for data to be delivered to them.

Example: Avro source, Thrift source

**Sink:**

Sink is the entity that delivers the data to the destination. A variety of sinks allow data to be streamed to a range of destinations.

Example: HDFS sink

**Channel:**

Channel is the conduit between the Source and the Sink. Sources ingest events into the channel and the sinks drain the channel.

Example: File system channel, Memory channel etc.,

**Agent:**

Agent is any physical Java virtual machine running Flume. It is a collection of sources, sinks and channels.

**Client:**

Client is the entity that produces and transmits the Event to the Source operating within the Agent.

**WORKING:**

* A flow in Flume starts from the Client.
* The Client transmits the Event to a Source operating within the Agent.
* The Source receiving this Event then delivers it to one or more Channels.
* One or more Sinks operating within the same Agent drains these Channels.
* Channels decouple the ingestion rate from drain rate using the familiar producer-consumer model of data exchange.
* When spikes in client side activity cause data to be generated faster than can be handled by the provisioned destination capacity can handle, the Channel size increases. This allows sources to continue normal operation for the duration of the spike.
* The Sink of one Agent can be chained to the Source of another Agent. This chaining enables the creation of complex data flow topologies.