● What is meant by FlumeNG ?

Flume NG is a refactoring of Flume. To solve certain known issues and limitations, Flume requires a refactoring of some core classes and systems.

The following known issues are specifically to be addressed

Code complexity. Flume has evolved over the last few years and has a fair amount of extraneous code. Core component lifecycle standardization and control code (e.g. anything that can be started or stopped, sources, sinks). (Static) Configuration access throughout the code base. Drastic simplification of common data paths (e.g. durability as an element of the source rather than a disconnected sink).

● Can Flume provides 100 % reliability to the data flow?

Important thing in flume is that all the transfers of the events are in transactional in nature. Hence the loss of the data is totally avoided. i.e every time the receiver gets the data it commits the received chunk and also sends the acknowledgement.

It is similar to bank transaction. Once Initiated and if the transaction is interrupted due to some reasons then the rollback of the action takes place and again the same transaction is initiated hence no data loss is caused during the data flow and hence the efficiency is 100%.

● Can Flume can distributes data to multiple destinations?

Yes flume can distribute the data to multiple destinations. As the flume agent can declare more than one source, channels and sink. Hence they can take inputs from many sources and buffer in multiple channels as well as give them to multiple sinks(Destinations) . For doing this on has to change the configuration of the flume and allocate the respective sinks.

e.g

# set channel for sinks

agent\_foo.sinks.hdfs-ca-sink.channel = mem-channel-ca

agent\_foo.sinks.hdfs-ny-sink.channel = mem-channel-ny

agent\_foo.sinks.null-sink.channel = mem-channel-other

● Explain about the different channel types in Flume. And which channel type is faster?

Different Channels in Flume are

1. Memory Channel - The Memory Channel is an in-memory channel that stores events written to it on the heap. For all practical purposes, the Memory Channel is an in-memory queue—the sources write to its tail and sinks read off its head. The Memory Channel supports very high throughput, as it holds all data in memory

* The events are stored in an in-memory queue with configurable max size. It’s ideal for flows that need higher throughput and are prepared to lose the staged data in the event of agent failures.
* The maximum number of events stored in the channel -100.
* The maximum number of events the channel will take from a source or give to a sink per transaction-100.
* Timeout in seconds for adding or removing an event-3sec
* Defines the percent of buffer between byteCapacity and the estimated total size of all events in the channel, to account for data in headers- 20.

1. File Channel

The File Channel is Flume’s persistent channel. It writes out all events to disk and thus does not lose data on process or machine shutdown or crash. The File Channel ensures that any events committed into the channel are removed from the channel only when a sink takes the events and commits the transaction, even if the machine or agent crashed and was restarted

* The maximum size of transaction supported by the channel-100.
* Amount of time (in millis) between checkpoints – 30000 mili secs.
* Maximum capacity of the channel- 1000000
* Amount of time (in sec) to wait for a put operation- 3.