



HDFS

Agenda

- HDFS federation
- High Availability
- FailOver



HDFS federation

- The name node stores all file – block mappings in memory. Thus when working with large number of small files, we can run out of namespace volume. To introduce horizontal scaling for namespace, HDFS federation is introduced.
- In HDFS federation multiple namenodes manages its own namespace volume. This namespace volume has its metadata and block pool(blocks for files in its namespace).
- Block pool is not partitioned. Means all blocks in a datanode do not have a particular set of files. Hence **each data node registers with each namenode**

High Availability

- If a namenode goes down, whole mapreduce processes will fail . Starting a new namenode from previously saved state may consume a lot of time. To ensure High Availability, a pair of namenodes in active-standby configuration is used.

Few changes for this are :

- I. Datanodes must send block reports to both datanodes.
 - II. The name node must use highly-available shared storage to share the edit log. When standby node starts, using edit log it can synchronise its state with failed name node and continue.
 - III. Client must be handled to deal with name node failure
- Journal nodes: When any namespace modification is performed by the Active node, it durably logs a record of the modification to a majority of these JNs. The Standby node is constantly watching them for changes to the edit log. As the Standby Node sees the edits, it applies them to its own namespace.

Automatic Failover

- Automatic failover adds two new components to an HDFS deployment: a ZooKeeper quorum, and the ZKFailoverController process

Thank you!

Contact:

EduPristine

702, Raaj Chambers, Old Nagardas Road, Andheri (E), Mumbai-400 069. INDIA

www.edupristine.com

Ph. +91 22 3215 6191