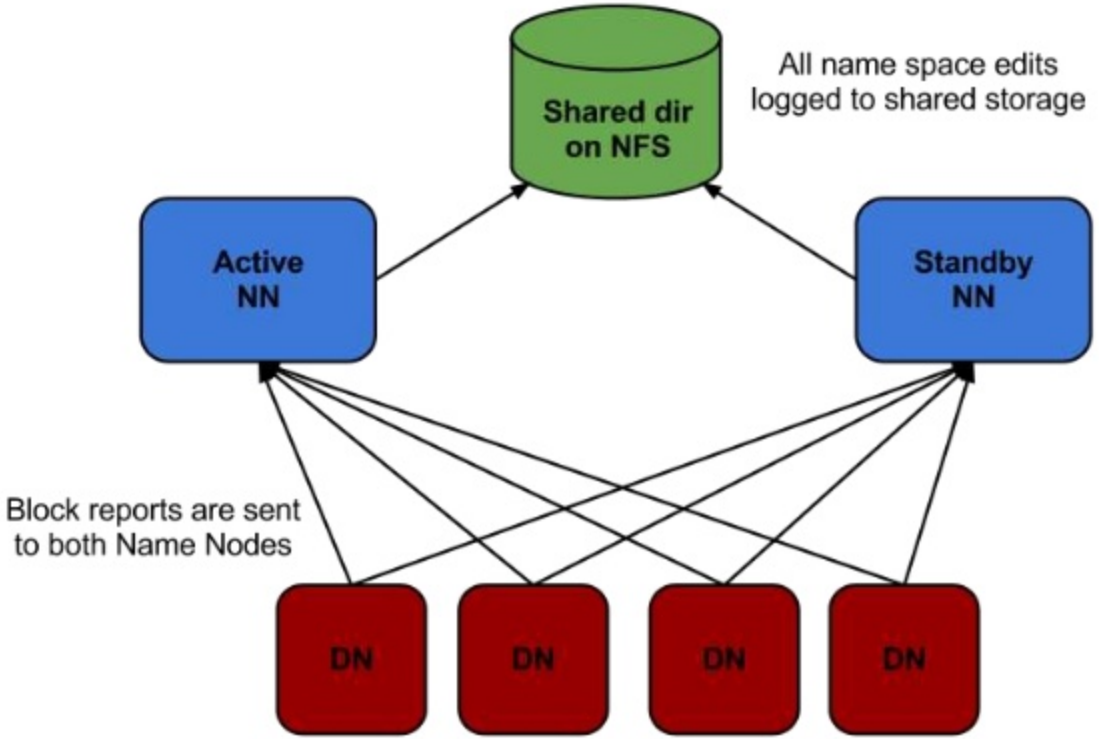
Introduction to Hadoop High Availability

Omid Vahdaty, DevOps Ninja

Prerequisites

- Be sure you know:
  - This is not an installation manual.
  - how to install an hadoop multi node cluster (single namenode + secondary)
  - 2 different ways to install Hadoop HA
    - HA with NFS
    - HA with QJM (Qurum Journal Manager)
  - **High availability Concepts**:
    - Having Secondary Namenode is NOT HA.
    - Active/standby = Active/Passive != Active/Active
    - Main Concern: Split Brain Scenario. Common solution: **STONITH** fencing method.

HA with NFS


Shared dir
on NFS
All name space edits
logged to shared storage
Active
NN
Standby
NN
Block reports are sent
to both Name Nodes
DN
DN
DN
DN

Hdfs-site.xml additions to HA (both methods)

```
<property>
  <name>dfs.nameservices</name>
  <value>mycluster</value>
</property>
<property>
  <name>dfs.ha.namenodes.mycluster</name>
  <value>nn1,nn2</value>
</property>
<property>
  <name>dfs.namenode.rpc-address.mycluster.nn1</name>
  <value>machine1.example.com:8020</value>
</property>
<property>
  <name>dfs.namenode.rpc-address.mycluster.nn2</name>
  <value>machine2.example.com:8020</value>
</property>
<property>
  <name>dfs.namenode.http-address.mycluster.nn1</name>
  <value>machine1.example.com:50070</value>
</property>
<property>
  <name>dfs.namenode.http-address.mycluster.nn2</name>
  <value>machine2.example.com:50070</value>
</property>
```

hdfs-site.xml additions to HA with NFS method

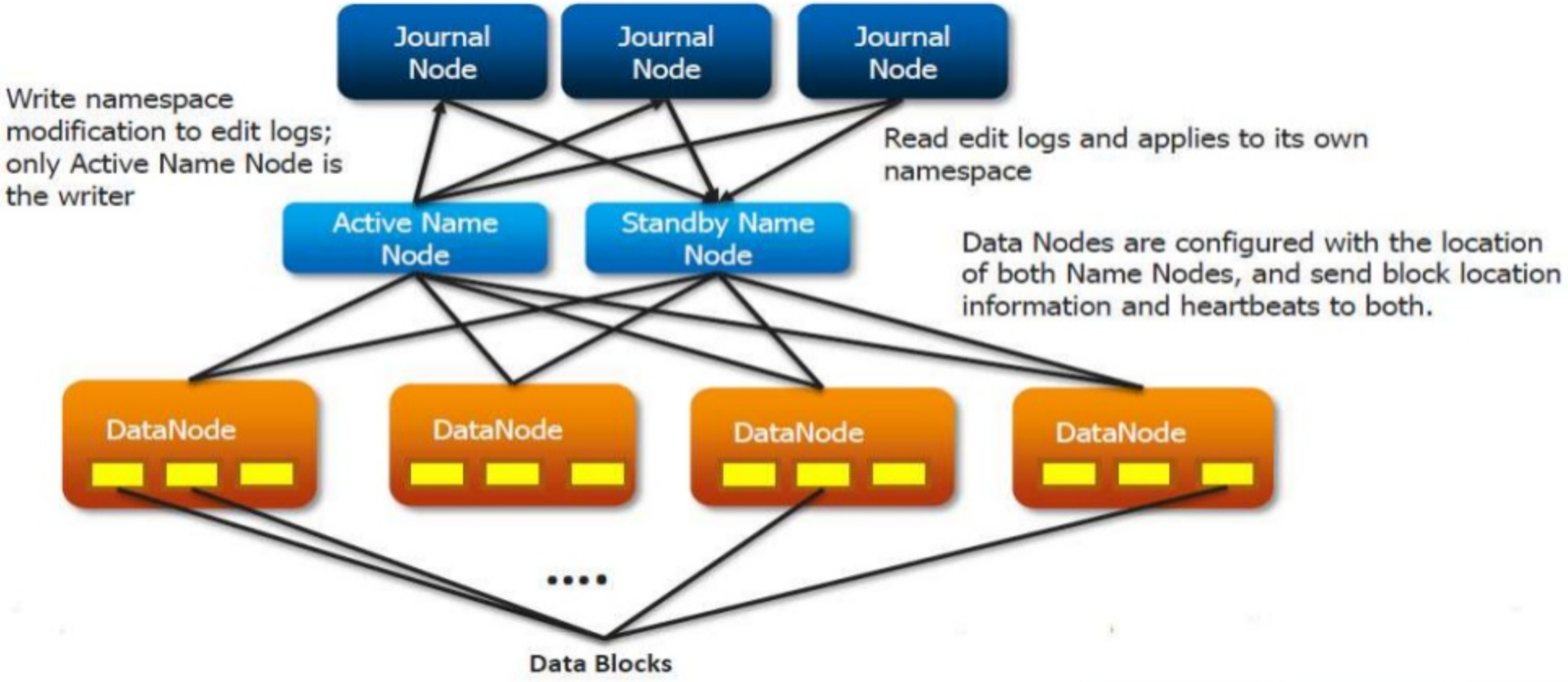
```
<property>
 <name>dfs.namenode.shared.edits.dir</name>
  <value>file:///mnt/namenode</value>
</property>
   <property>
     <name>dfs.ha.fencing.methods</name>
     <value>sshfence</value>
    </property>

    <property>
     <name>dfs.ha.fencing.ssh.private-key-files</name>
     <value>/home/hadoopuser/.ssh/id_rsa</value>
    </property>
```

Core-site.xml additions

```
<property>
  <name>fs.defaultFS</name>
  <value>hdfs://mycluster</value>
</property>
```

HA with Qurum Journal Manager



Hdfs-site.xml additions to HA with QJM

<property>
<name>dfs.namenode.shared.edits.dir</name>
<value>qjournal://node1.example.com:8485;node2.example.com:8485;node3.example.com:8485/mycluster</value>
</property>

Core-site.xml addition with HA QJM method

```
<property>
  <name>fs.defaultFS</name>
  <value>hdfs://mycluster</value>
</property>
<property>
  <name>dfs.journalnode.edits.dir</name>
  <value>/path/to/journal/node/local/data</value>
</property>
```

Before you start experimenting...

1. Be sure to understand the format NN flow.
2. Learn how to “upgrade” from one NN to 2 NN
3. Test your cluster is healthy (cluster up, does not mean no ERROR in the logs of each component)
4. Make sure you understand format section in the APACHE documentation of HA installation carefully.
5. Play with some Scenarios:
    a. Manual Failover: **hdfs haadmin -failover nn2 nn1**
    b. getServiceState: **hdfs haadmin -getServiceState nn1**

Be Careful of twilight zone

1. Cluster appears to be up(no errors), **but** both NN are in **Standby (default when starting cluster)**
2. Datanode appears to be up(no errors), **but** Datanode is out of sync with NN.

Format new cluster

- ***"hadoop-daemon.sh start journalnode"*** *(on all QJM machines)*
- ***"hdfs namenode -format"*** *(for fresh cluster)*
- ***"hdfs namenode -bootstrapStandby"*** on the unformatted NameNode (on existing cluster only)
- *"hdfs namenode -initializeSharedEdits"*, which will initialize the JournalNodes with the edits data from the local NameNode edits directories.
- **"hadoop-daemon.sh start namenode"**
- Read logs - JPS is not enough
- Check NN state (make sure you have at least one in active)
  - hdfs haadmin -failover nn2 nn1

Starting a cluster

- ***start-dfs.sh***
- Activate a NN: ***hdfs haadmin -failover nn2 nn1***
- If Quorum not achieved(*"**Got too many exceptions to achieve quorum size 2/3**"*):
  - ***Stop-dfs.sh***
  - *on all Journal nodes: "**hadoop-daemon.sh start journalnode** "*
  - ***Start-dfs.sh***
  - *Activate a NN: **"hdfs haadmin -failover nn2 nn1"***
  - *Confirm NN is active via**: "hdfs haadmin -getServiceState nn1**"*

A quick word about Failover flow

If the first NameNode is in the Active state, an attempt will be made to gracefully transition it to the Standby state. If this fails, the fencing methods (as configured by **dfs.ha.fencing.methods**) will be attempted in order until one succeeds. Only after this process will the second NameNode be transitioned to the Active state. If the fencing methods fail, the second NameNode is not transitioned to Active state and an error is returned.

If the first NameNode is in the Standby state, **the command will execute and say the failover is successful, but no failover occurs** and the two NameNodes will remain in their original states.

Corrupted your Testing cluster?

- Delete your namenode folder
- Delete your datanode folder
- Delete your journalnode folder
- Start again…
- Yes, you need to format the cluster again.