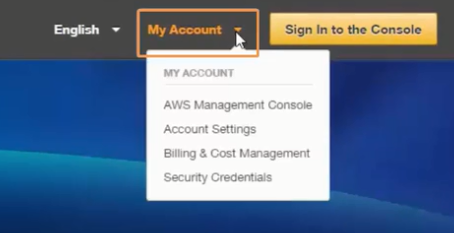
AWS EC2

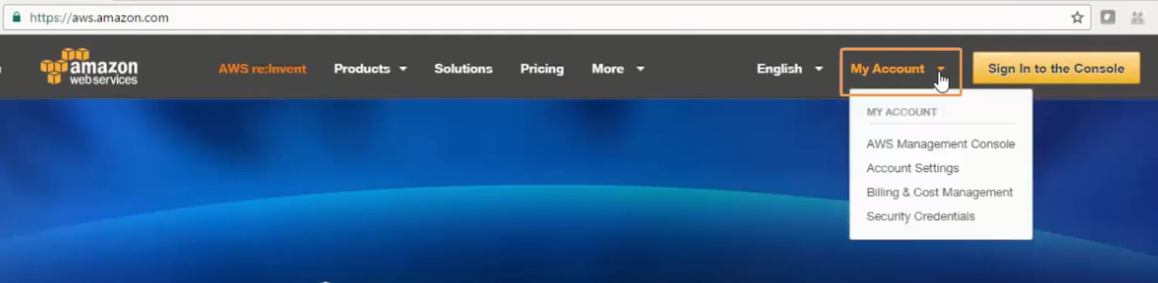


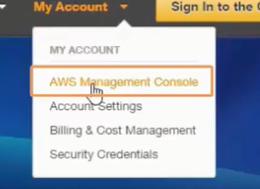
Create an account:

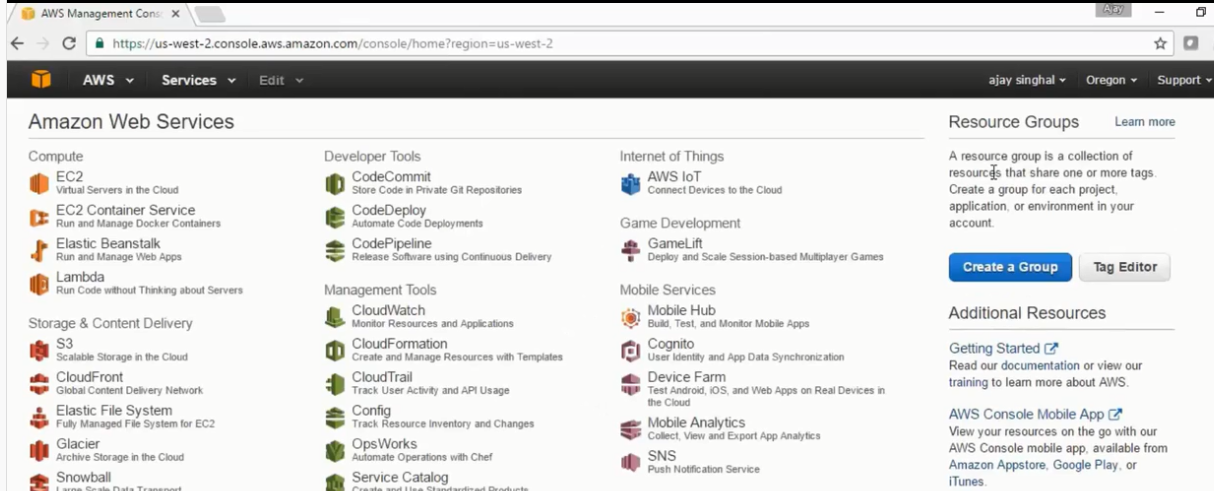




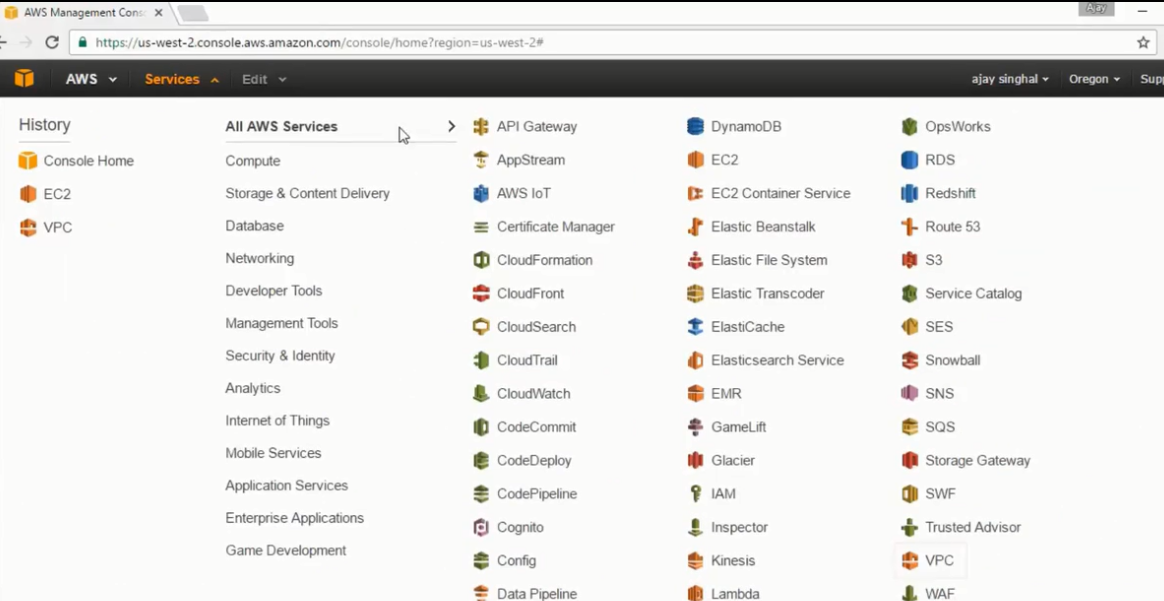


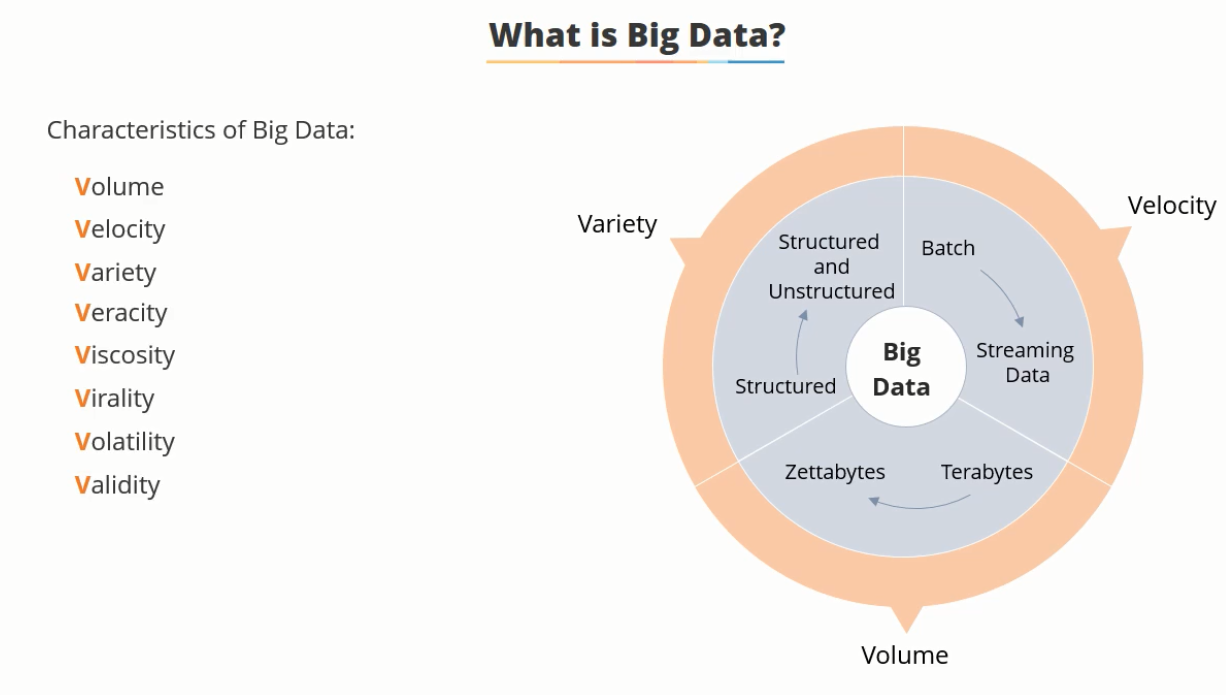




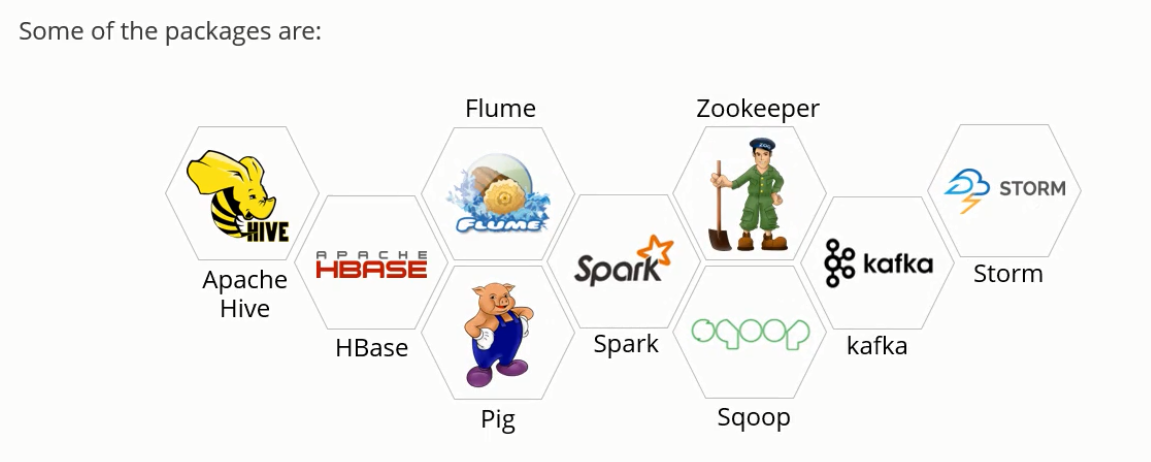


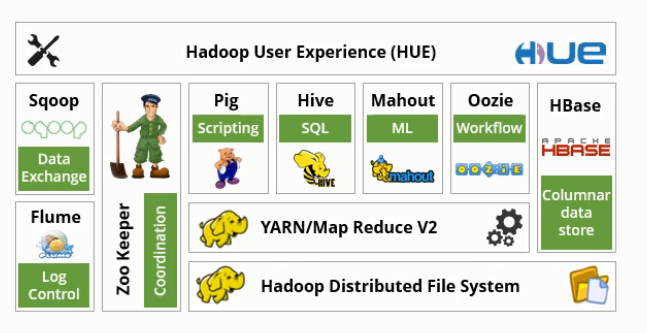
To create a VPC Click on Services:

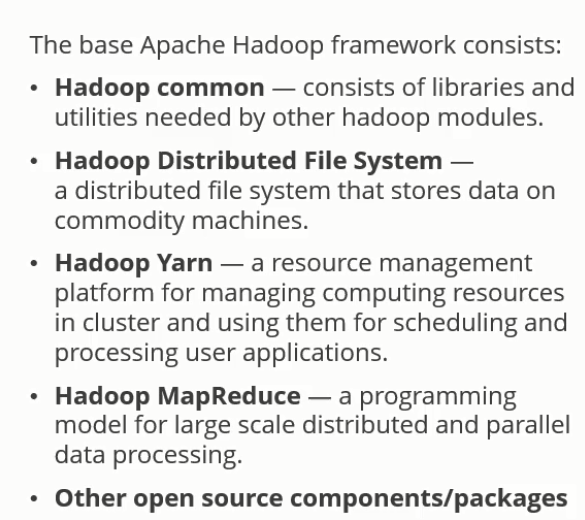




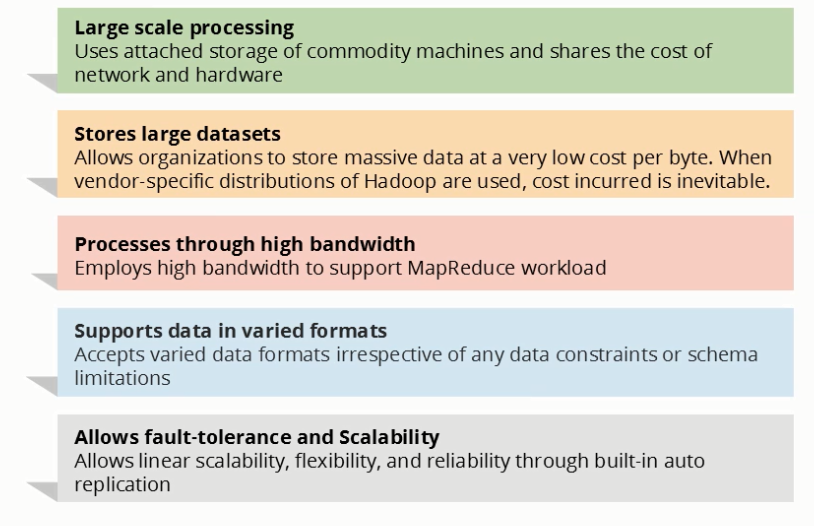








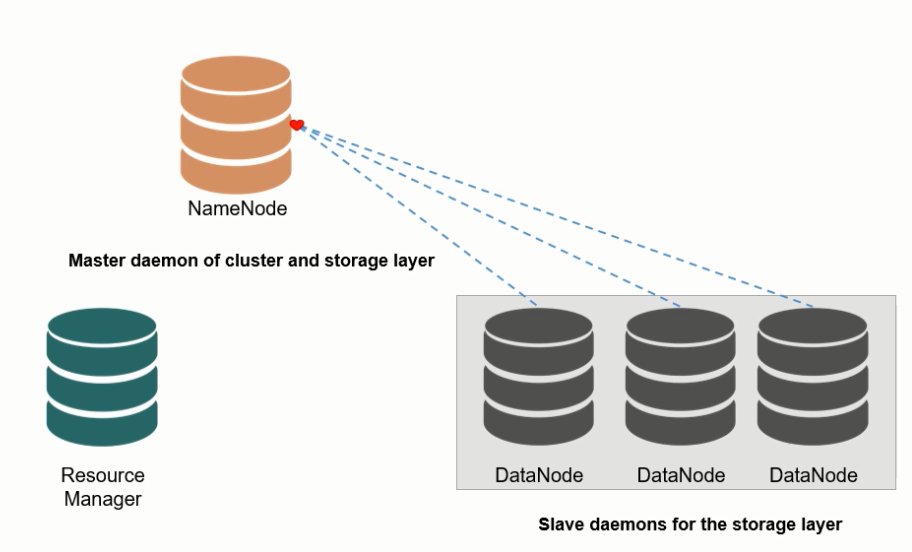
HDFS



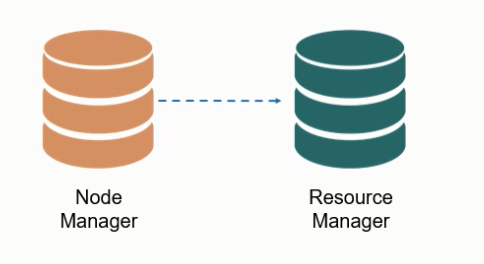


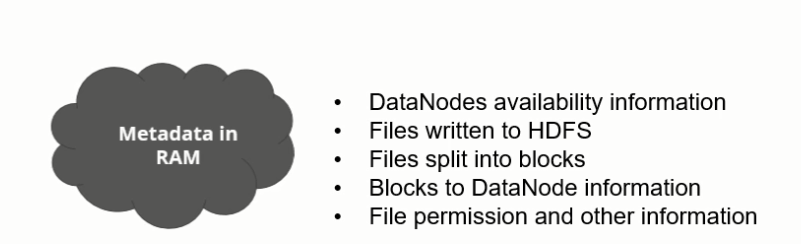
Resource Manager is the master of the processing layer

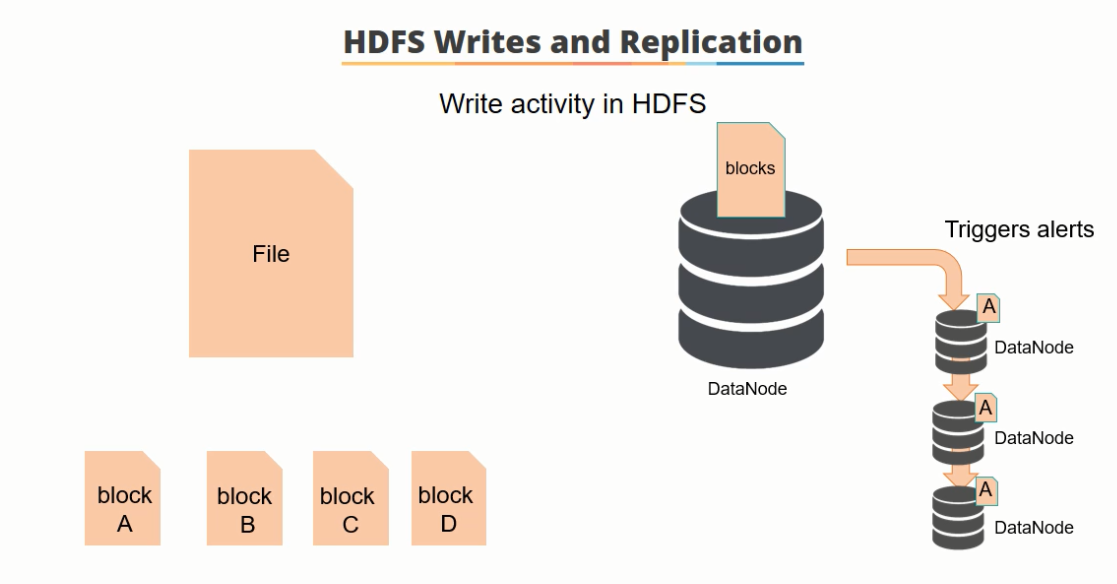
DataNodes send heart beat message to the name node which updates the metadata file continuously. API contacts NameNode. Data is broken into blocks and stored

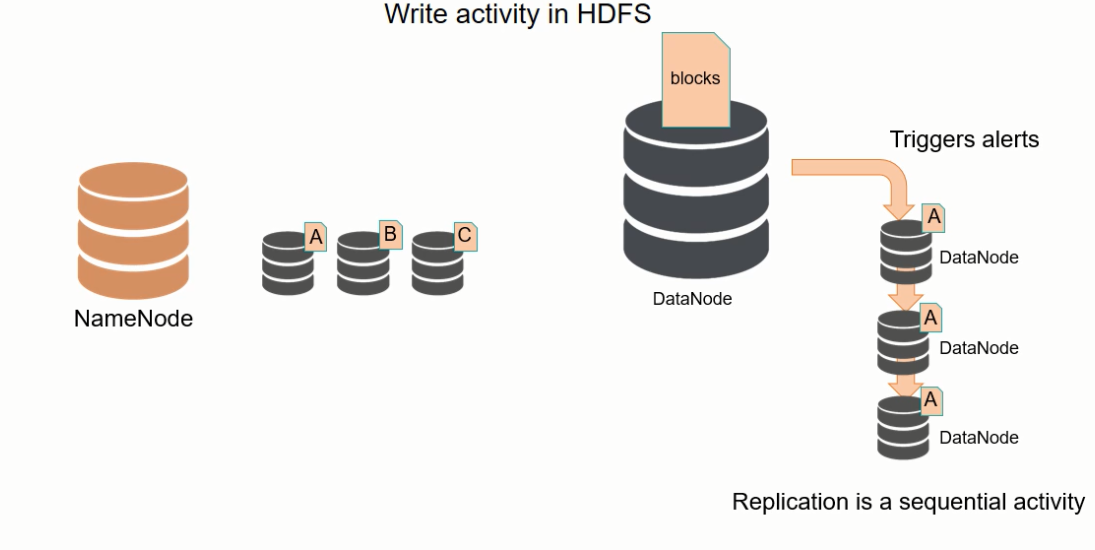


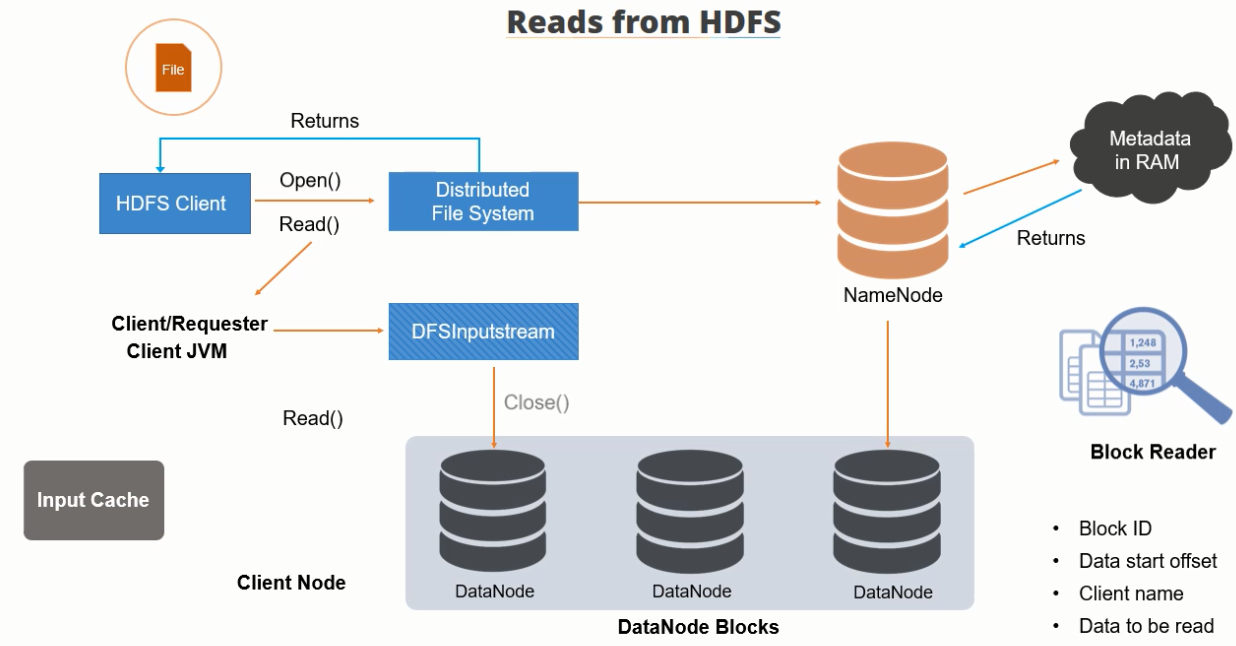
Node managers send details to the resource manager which keeps the consolidated details for the cluster

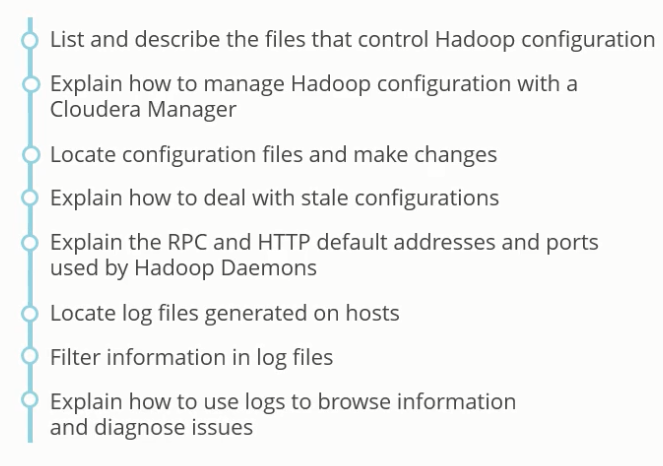


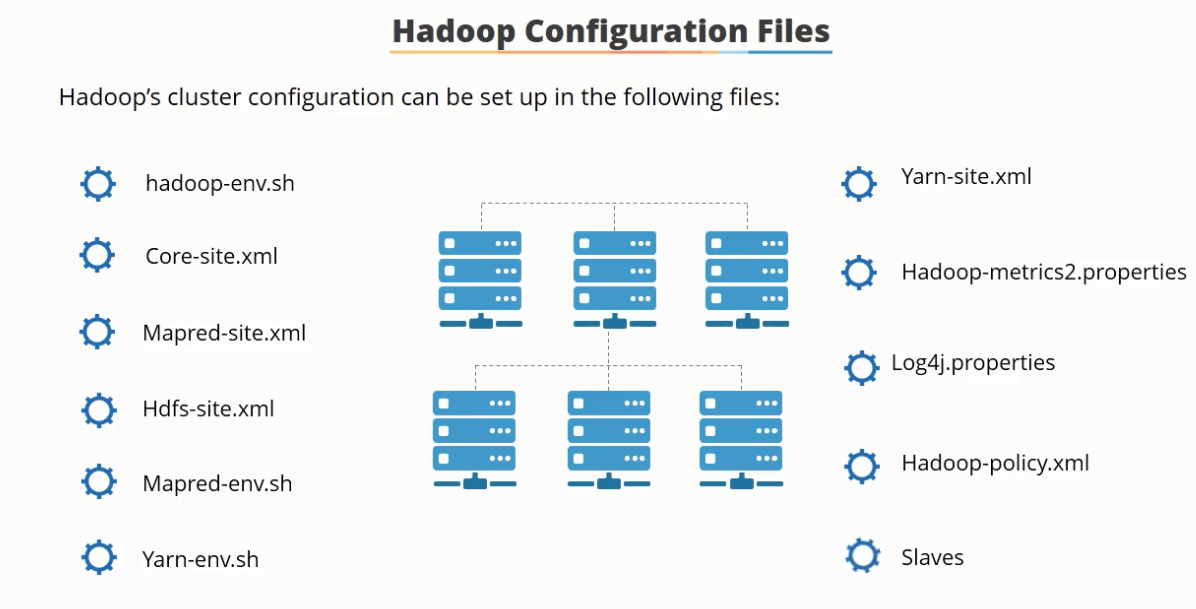


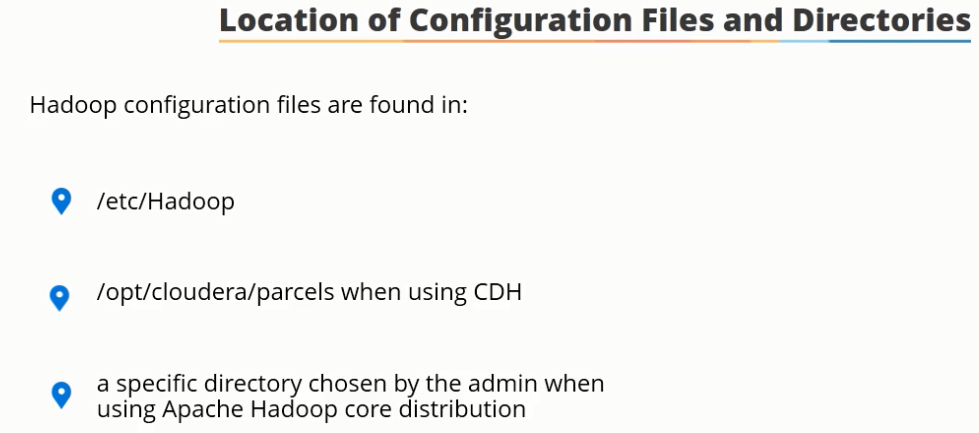


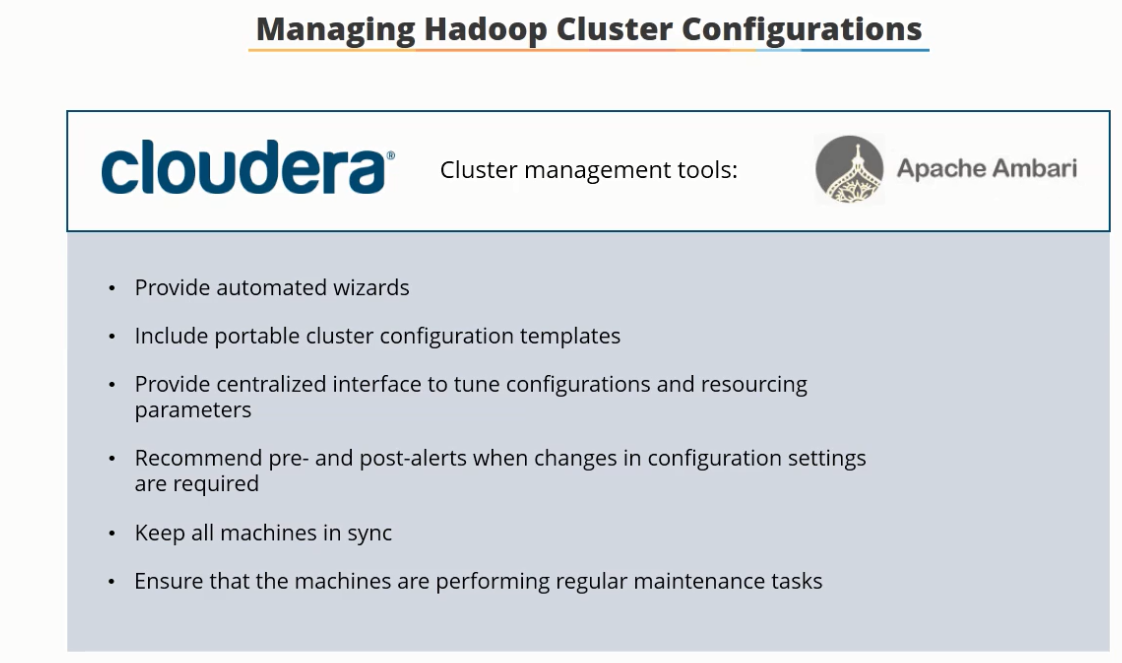








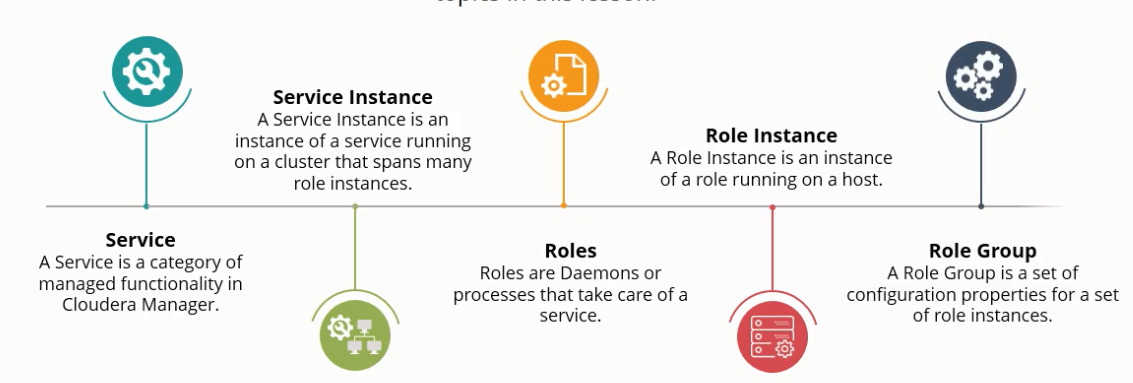


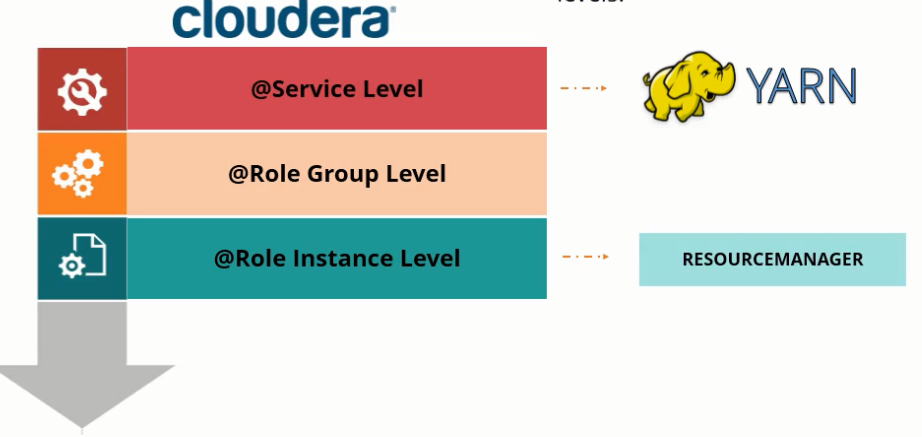


Management tools help in propagating changes across the cluster. They provide automated tools to quickly deploy a cluster. Ensure consistency across environments with portable cluster configuration templates.

**These tools cannot be used to manage new machines added to the cluster, especially if they have a different in the hardware specifications and are different from the machines already existing in the cluster. To manage these, tools such as CHEF, PUPPET, CFEngine, SALTStack, Ansible and so on are required**

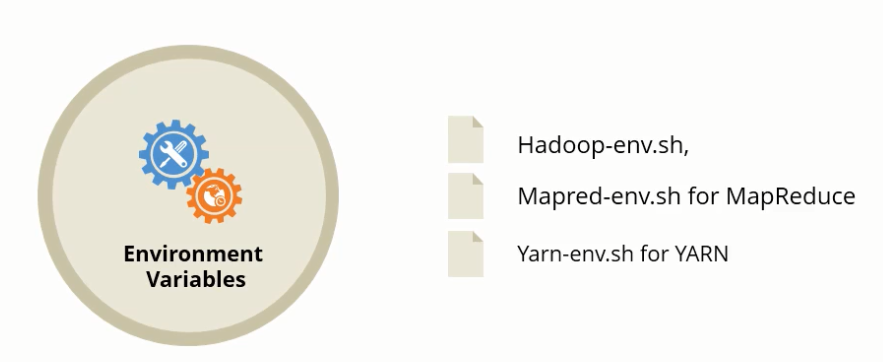
Puppet, Chef, Ansible and SaltStack present different paths to achieve a common goal of managing large-scale server infrastructure efficiently, with minimal input from developers and sysadmins. All four configuration management tools are designed to reduce the complexity of configuring distributed infrastructure resources, enabling speed, and ensuring reliability and compliance.

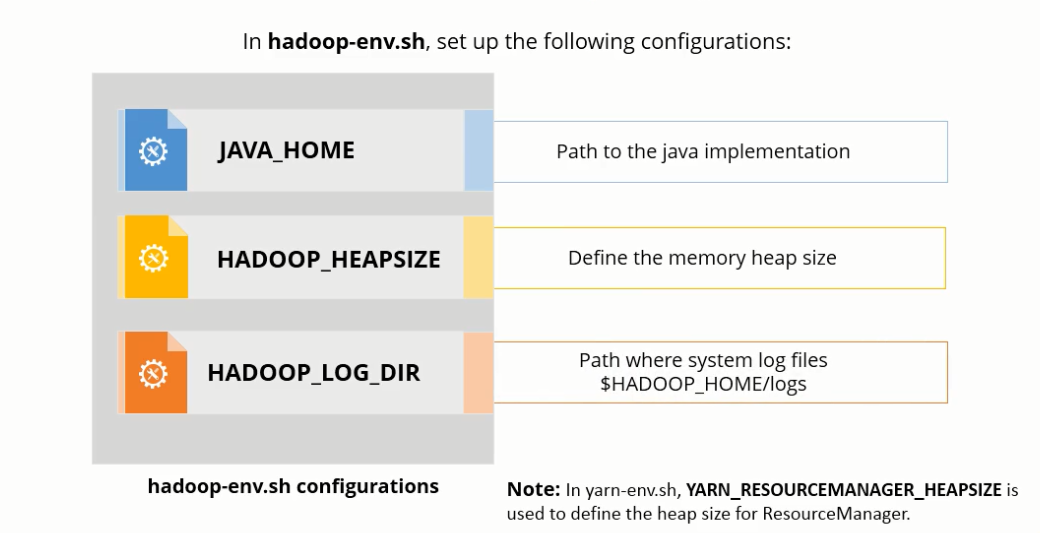




Configurations can be set at the @Service Level or @Role Instance Level. The latter overrides the former settings







Specifying the location of the Name Node. Value implies the port that the name node will listen to for a remote set up

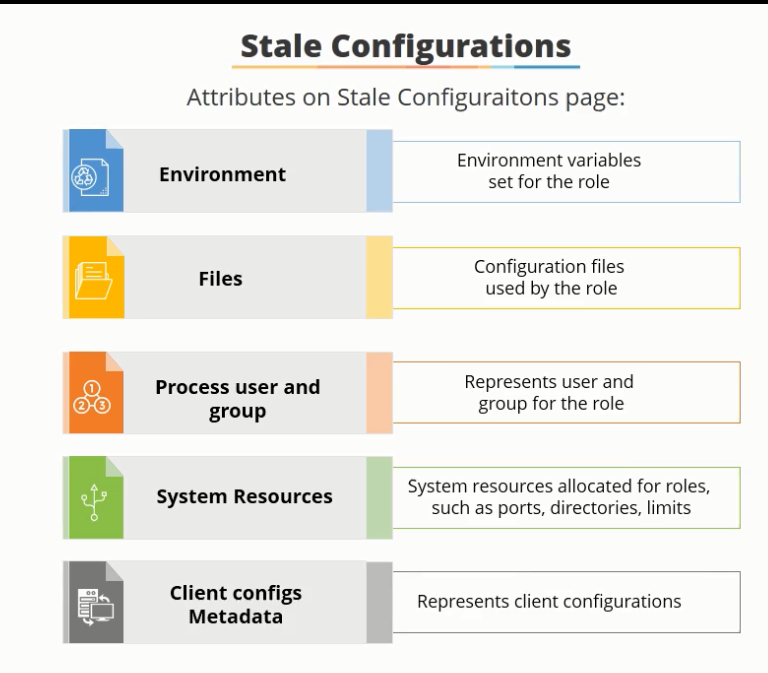


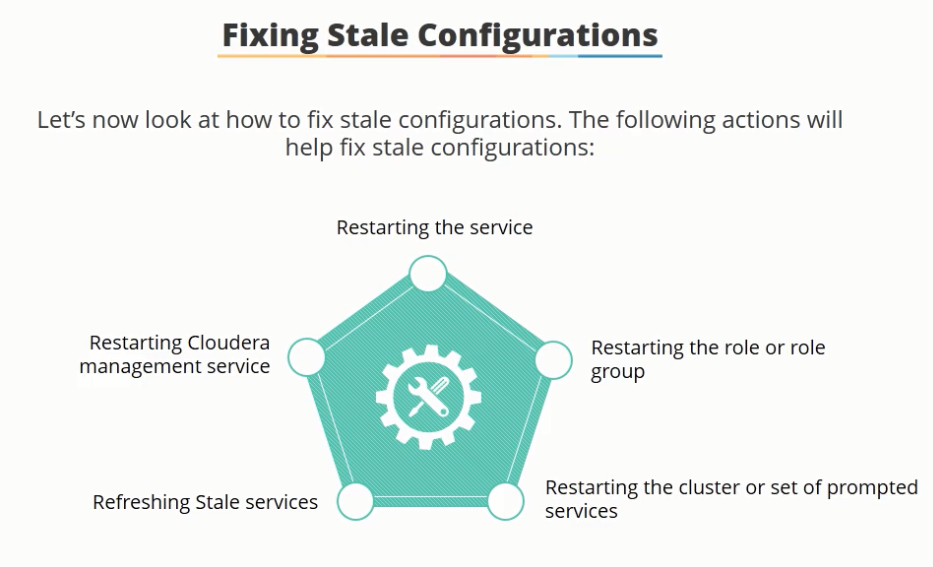




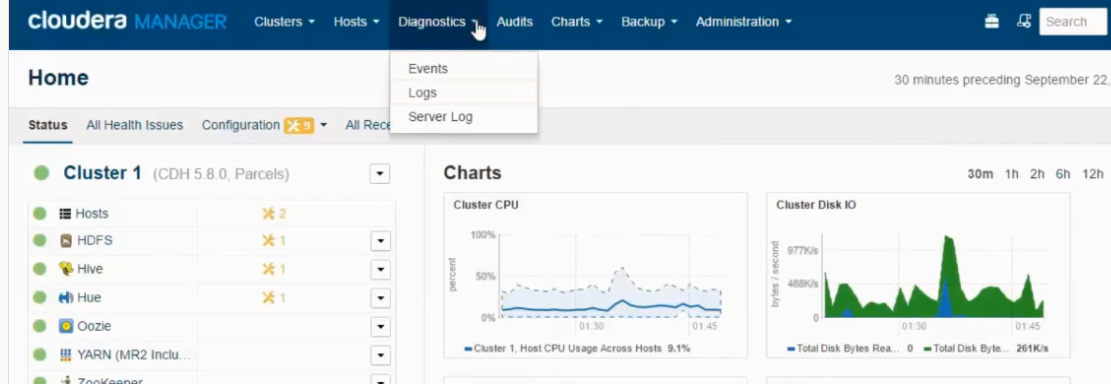
dfs.namenode.checkpont.dir => Secondary name node path

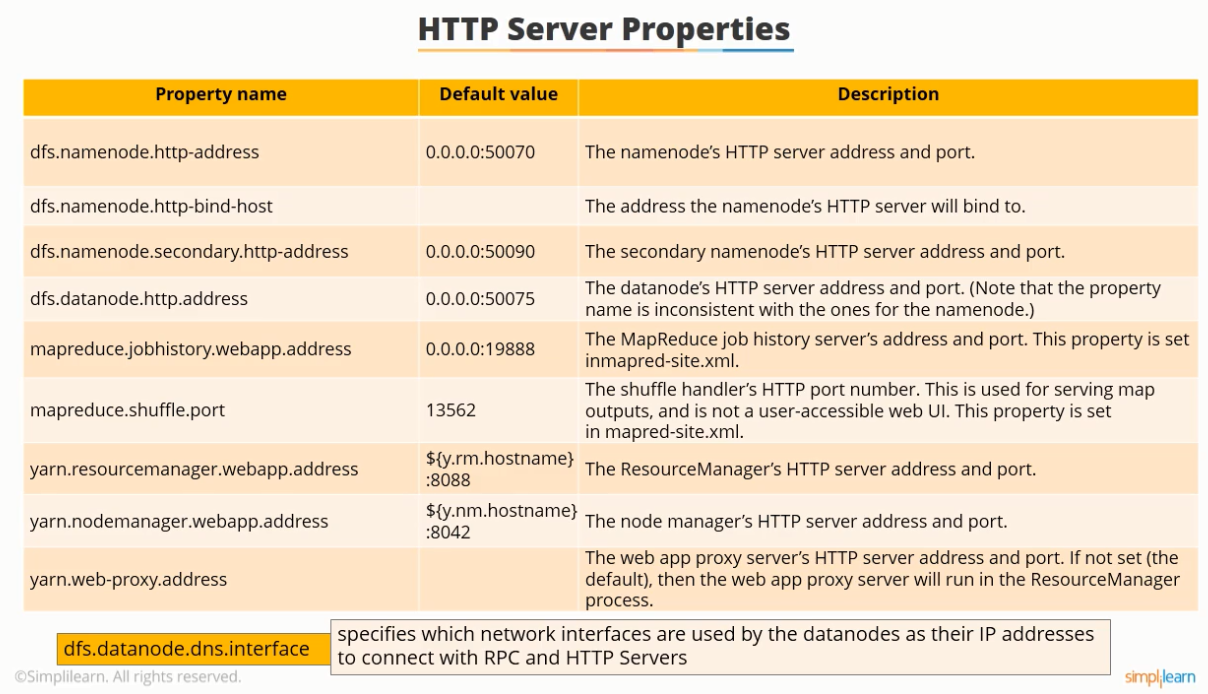


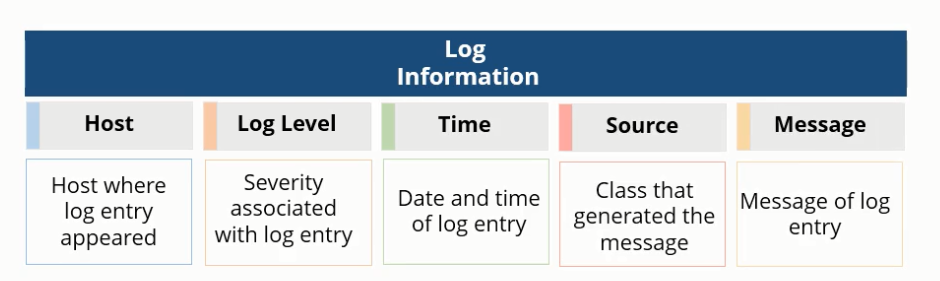


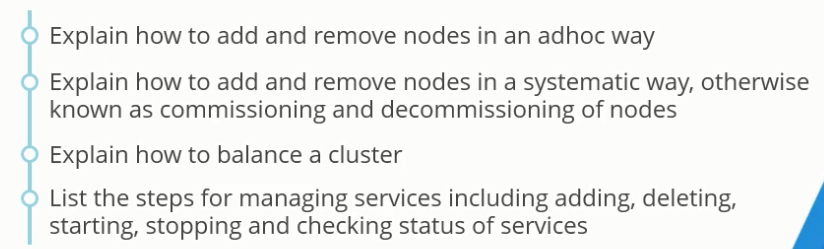


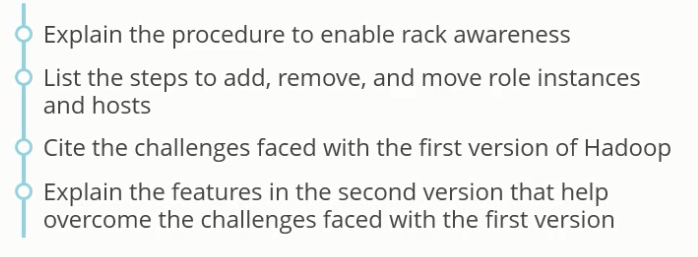
Cloudera Manager => Diagnostics => Logs contains all of the log files generated by the services





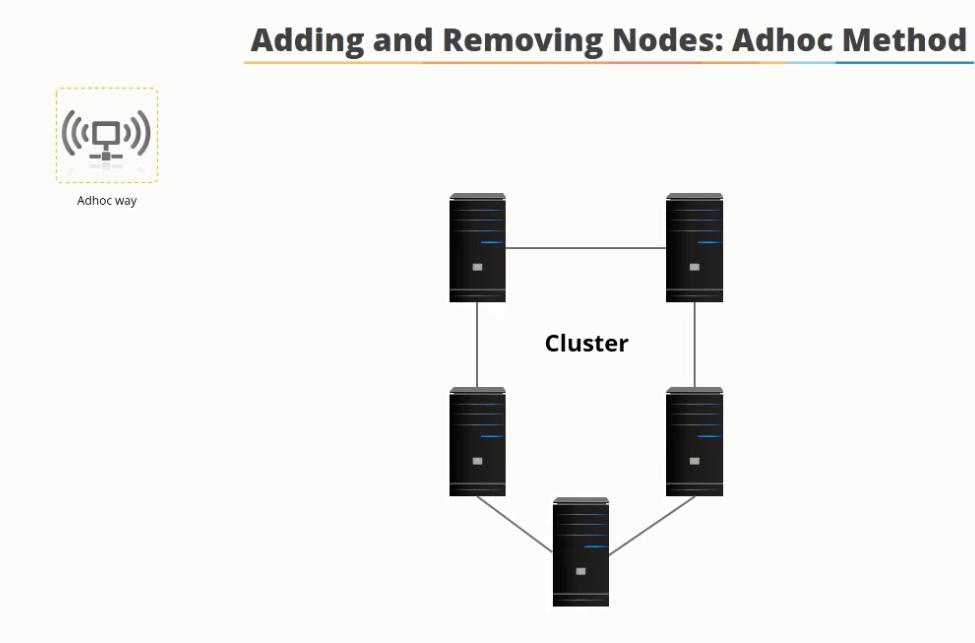




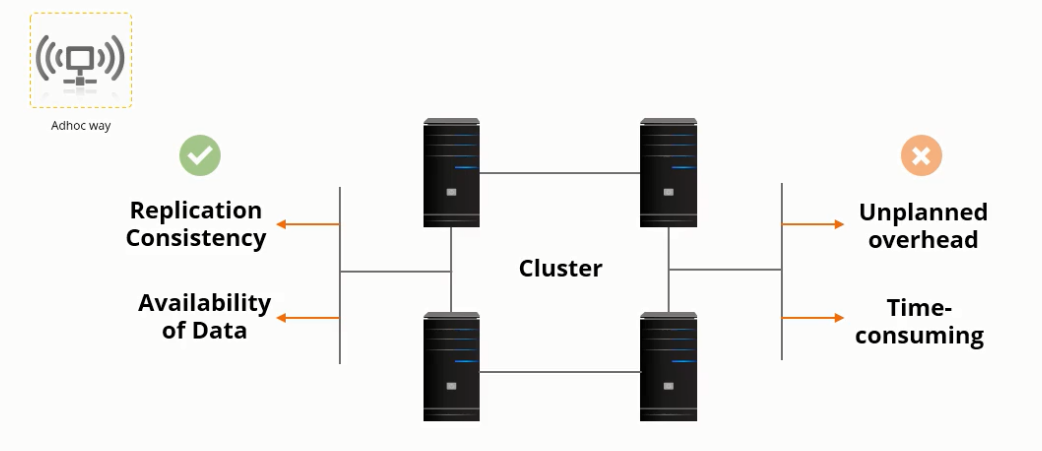


Adding or removing node from the cluster is possible either by the Ad-hoc way or systematic way.

To add a node to the cluster simply add a node and allow it to connect to the name node or simply remove the node from the cluster

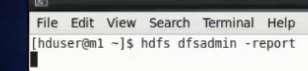


This incurs a lot of stress on the name node



In the systematic approach, the name node is notified about the addition prior to the insertion of the name node (which is also known as commissioning a node). Deleting the node is called decommissioning the node

Demo:

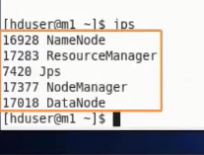
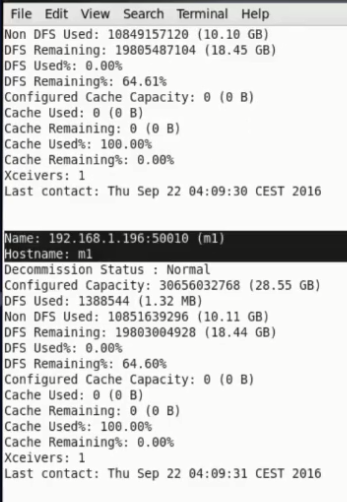


Shows you how many data nodes are running in the system, and their status.

To remove a machine in the ad-hoc method, just power it off.

Listing the machine with the jps command, only consists of a single now after powering it off

After a machine is deactivated, name node will still try to contact the machine. The name node will wait for approximately 10 minutes before declaring the machine dead.



The same command now shows the contacting information

