Session 5

Planning a Hadoop Cluster

Master Node Considerations

- ✓ Master processes tends to be RAM hungry but low on disk space consumption

- ✓ Serves critical functions without which server can’t live
  - Dual Power supplies
  - Bonded Network Interface cards
  - RAID 10 for NN storage

NameNode Considerations

- ✓ NN consumes 1 GB of RAM for every 1 million blocks
- ✓ NN disk requirements are modest since all metadata must fit in memory
- ✓ Reliability is paramount
- ✓ Dual channel or Triple channel memory controllers are recommended
- ✓ NN metadata should also be backed up to a SNN and a NFS
- ✓ SNN should have similar hardware as primary NN, so that it can replace the primary NN when needed
- ✓ In general, leave the NN alone

JobTracker Considerations

- ✓ Memory hungry b/c it keeps job counters and metadata in RAM for the last hundred jobs by default. Job details purged from memory no longer appear in Apache Hadoop’s JT web UI.

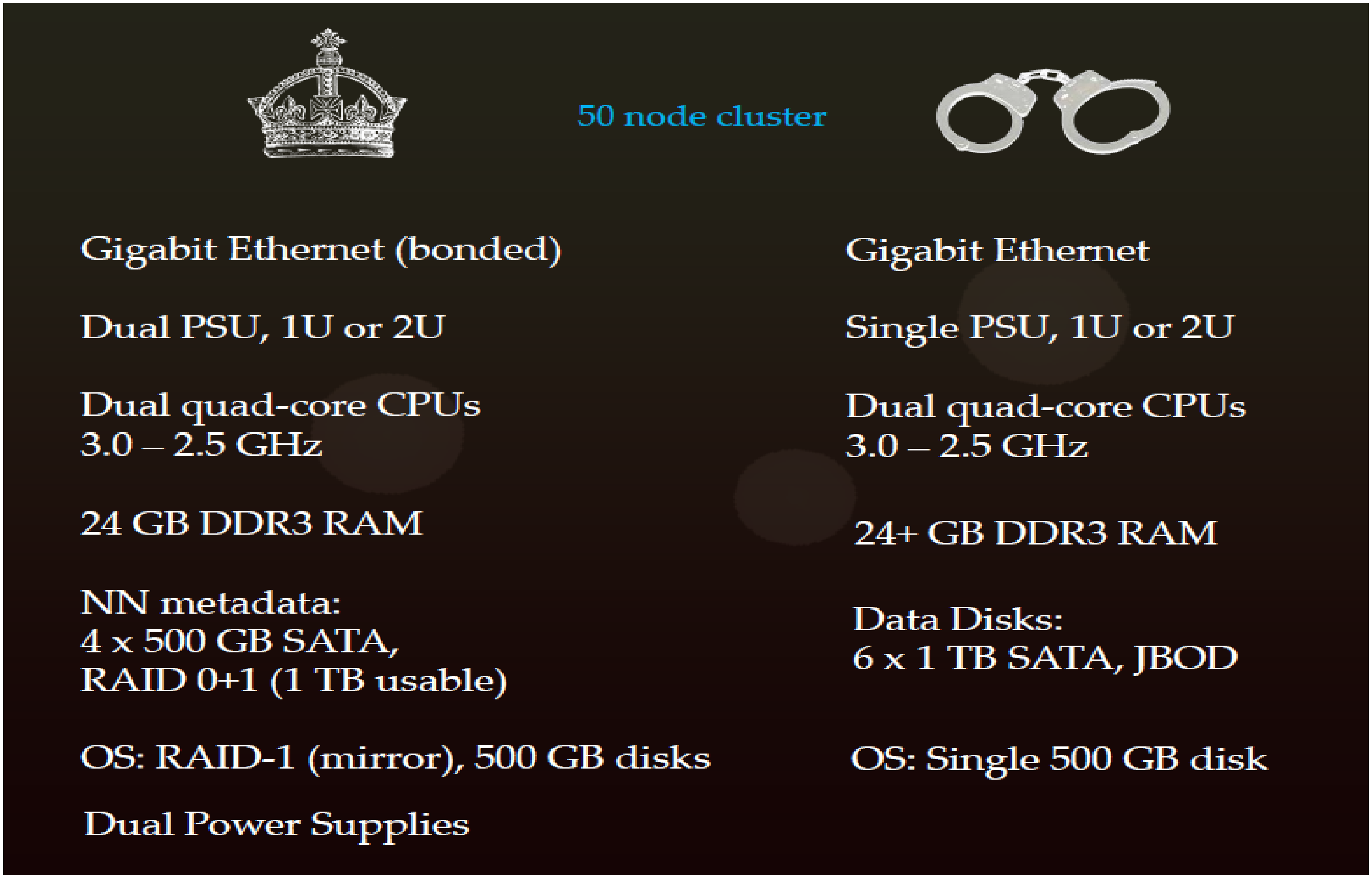
- ✓ Reliability is key, just like with the NN

Worker Hardware Considerations

- If 1 TB of new day ingest per day is expected…
- With R=3, we now have 3 TB
- MR temporary data will need about 25% of the disk space
- So, 3.5 – 4TB of disk space required every day

- Each task will consume 2 – 4 GB of memory
- A machine with 48 GB of RAM will support between 10 – 20 tasks.

Sample Hardware Configuration



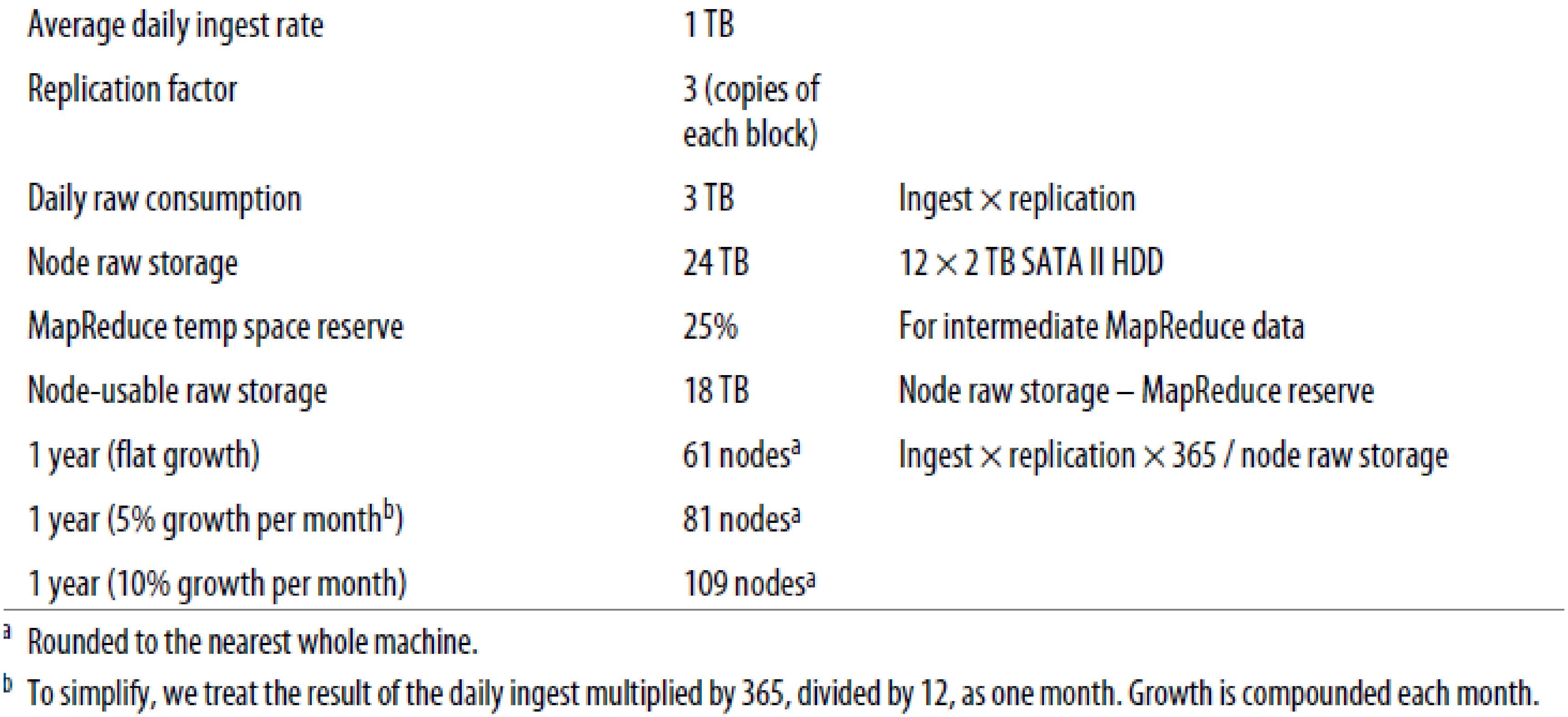
50 node cluster

Gigabit Ethernet (bonded)	Gigabit Ethernet
Dual PSU, 1U or 2U	Single PSU, 1U or 2U
Dual quad-core CPUs 3.0 – 2.5 GHz	Dual quad-core CPUs 3.0 – 2.5 GHz
24 GB DDR3 RAM	24+ GB DDR3 RAM
NN metadata: 4 x 500 GB SATA, RAID 0+1 (1 TB usable)	Data Disks: 6 x 1 TB SATA, JBOD
OS: RAID-1 (mirror), 500 GB disks	OS: Single 500 GB disk
Dual Power Supplies	

Cluster Sizing

- ✓ Can be optimized for a specific use case or for diverse workloads
- ✓ Most people prefer dense machines with lots of disks (twelve 2 TB disks per node)
- ✓ Most common scenario is to size based on the original data load and ingest rate (remember the replication factor, OS overhead and MR temp space 25%)
- ✓ It is recommended to run a test MR job on sample subset of data to help predict job completion times on a larger cluster

Sample Cluster Sizing

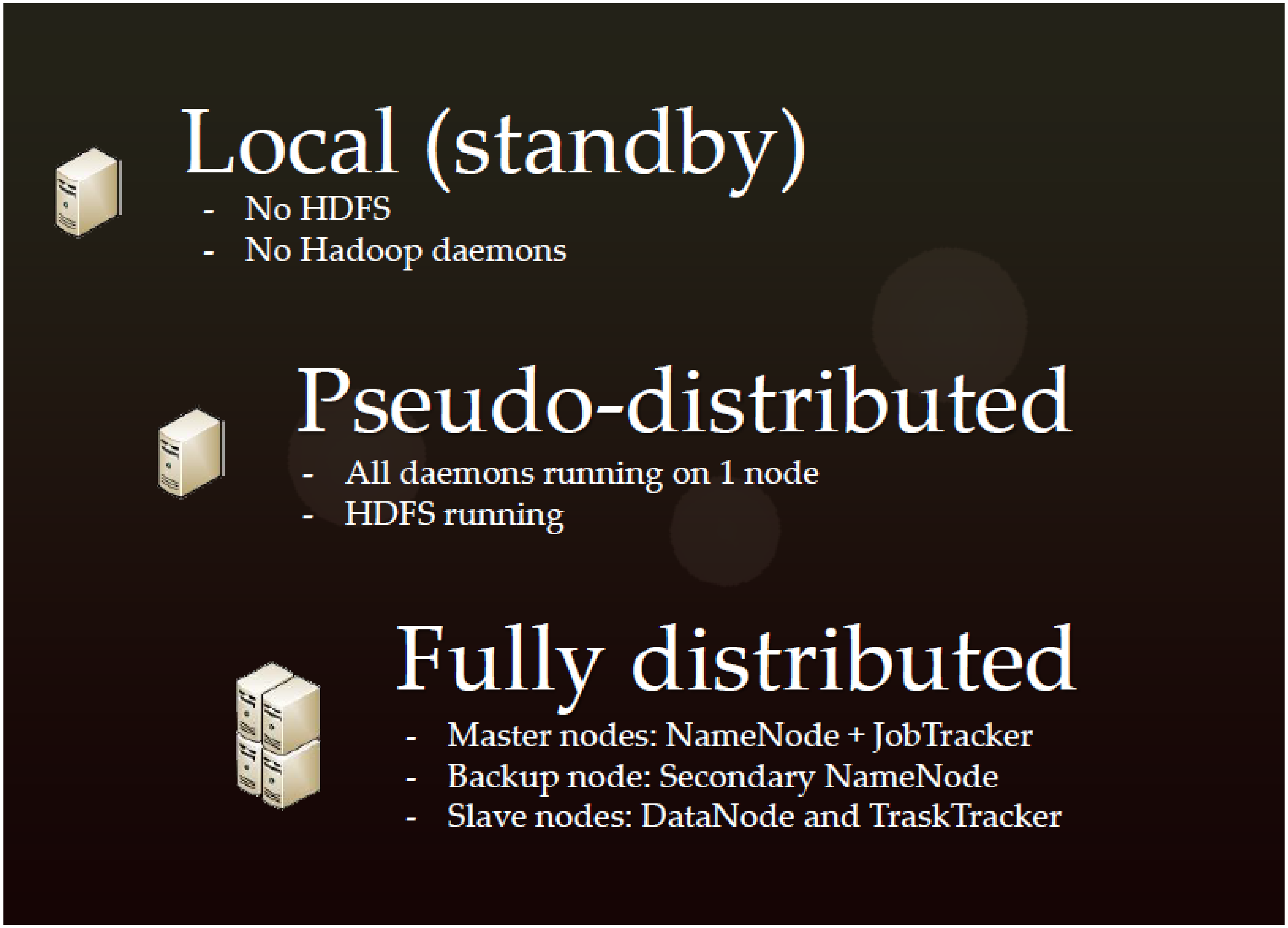


Average daily ingest rate	1 TB	
Replication factor	3 (copies of each block)	
Daily raw consumption	3 TB	Ingest × replication
Node raw storage	24 TB	12 × 2 TB SATA II HDD
MapReduce temp space reserve	25%	For intermediate MapReduce data
Node-usable raw storage	18 TB	Node raw storage – MapReduce reserve
1 year (flat growth)	61 nodes[a]	Ingest × replication × 365 / node raw storage
1 year (5% growth per month[b])	81 nodes[a]	
1 year (10% growth per month)	109 nodes[a]	

[a] Rounded to the nearest whole machine.

[b] To simplify, we treat the result of the daily ingest multiplied by 365, divided by 12, as one month. Growth is compounded each month.

Deployment Modes



Local (standby)

- No HDFS
- No Hadoop daemons

Pseudo-distributed

- All daemons running on 1 node
- HDFS running

Fully distributed

- Master nodes: NameNode + JobTracker
- Backup node: Secondary NameNode
- Slave nodes: DataNode and TraskTracker

Operating Systems

- **RedHat:** aimed at commercial and enterprise-customers
- **CentOS:** mirrors RedHat, slow to adopt new features
- **Fedora:** more of a playground for new features, RH sponsor
- **Debian:** not backed by commercial entity, governed by project rules
- **Windows:** ?

Java

- All machines should run the same version of Java + patch level.
- Use a 64-bit architecture and JDK

- See the Hadoop community list of tested JVMs here: http://wiki.apache.org/hadoop/HadoopJavaVersions

Hadoop Directories

Hadoop home: Hadoop software installed here, typically not in user’s home dir. Is usually in /usr/local, /opt or /usr. Can be made read only when properly configured.

Datanode data directories: stores HDFS blocks. DN assumes each dir is a separate physical spindle and it round robins blocks between disks. This disks are also often used for the tasktracker MR local spill data.

NN directories: Used by NN to store FS metadata. NN assumes each dir is a separate disk and replicates all writes to each device synchronously to ensure availability. Generally no more than 100 GB is needed. One of these dirs is usually a NFS mount.

Hadoop Directories Contd…

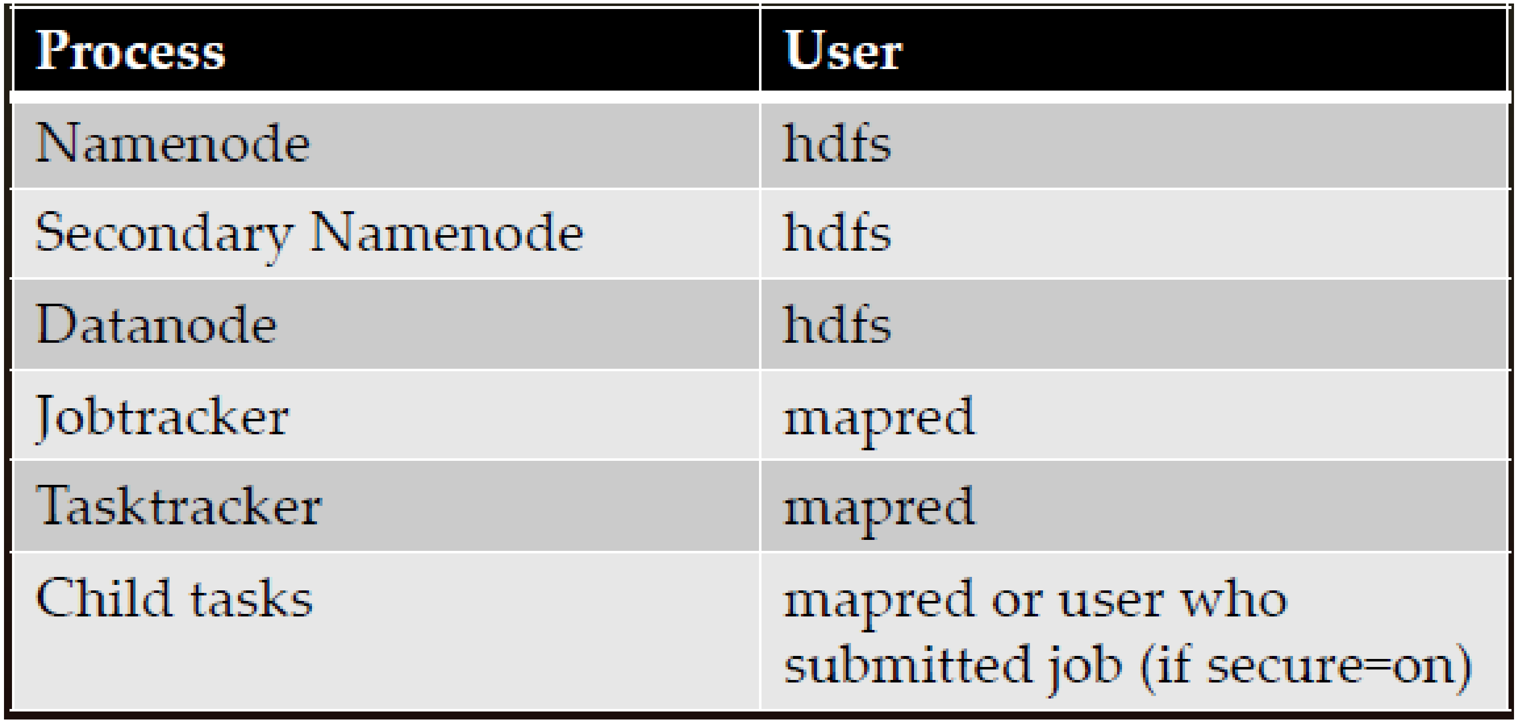
MR local directories: stores tasktracker temporary data. These dirs can store a moderate amount depending on job characteristics.

Hadoop log directories: common dir used by all daemons to store log data, job and task data.

Hadoop PID directory: dir used by all daemons to store pid files. Very small.

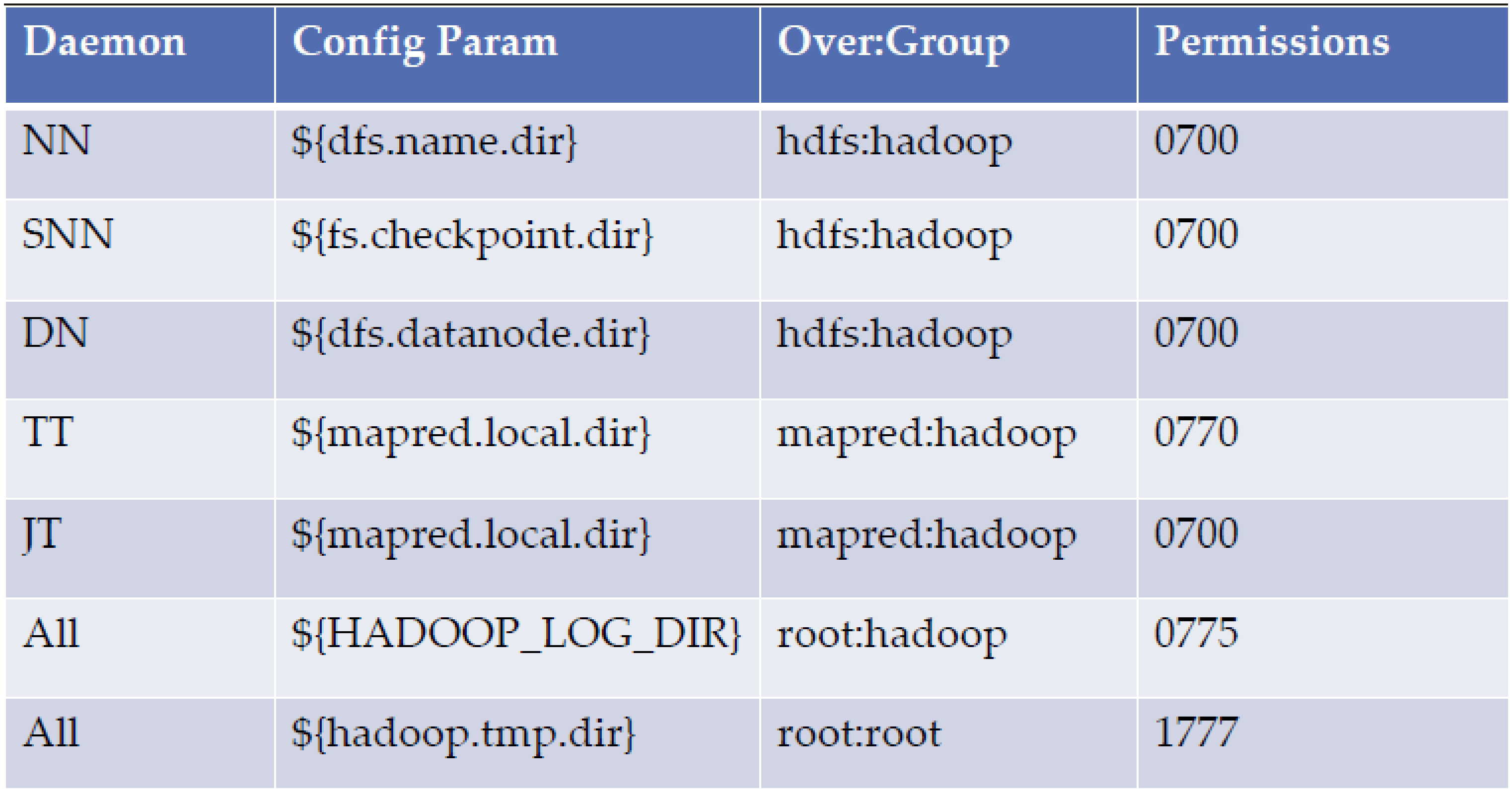
Hadoop temp directory: for small, short lived files. Mostly used on the machines from which MR jobs are submitted and contains a copy of the JAR file that gets sent to the JT. It is set to /tmp/hadoop-${username} by default.

Hadoop Daemon Users



Process	User
Namenode	hdfs
Secondary Namenode	hdfs
Datanode	hdfs
Jobtracker	mapred
Tasktracker	mapred
Child tasks	mapred or user who submitted job (if secure=on)

Hadoop Dirs + Permisiions



Daemon	Config Param	Over:Group	Permissions
NN	${dfs.name.dir}	hdfs:hadoop	0700
SNN	${fs.checkpoint.dir}	hdfs:hadoop	0700
DN	${dfs.datanode.dir}	hdfs:hadoop	0700
TT	${mapred.local.dir}	mapred:hadoop	0770
JT	${mapred.local.dir}	mapred:hadoop	0700
All	${HADOOP_LOG_DIR}	root:hadoop	0775
All	${hadoop.tmp.dir}	root:root	1777

Kernel Tuning

- The following two parameters should be configured in: /etc/sysctl.conf so they survive a reboot vm.

- **swappiness:**
  - -ontrols kernel's tendency to swap app data from memory to disk
  - Valid range is 0 – 100, higher values mean kernel will swap more
  - Swapping can cause daemons to timeout, especially bad for HBase
  - Recommended setting is 0 to turn it off; system default is between 60-80

- **vm.overcommit_memory** (if using Hadoop streaming):
  - 0: check if enough memory is available and allow allocation or deny
  - 1: permit memory allocation in excess of physical RAM + swap
  - 2: always return success to an app's request for memory
  - Recommended setting is 1

Hadoop Configuration

- **Admin Controlled**
  - ✓ Cluster
  - ✓ Daemon

- **Developer Controlled**
  - ✓ Job
  - ✓ Individual Operation(like replication level in fs command)

Configuration Files - 1

hadoop-env.sh: bourne shell fragment sourced by hadoop scripts, specifies environment variables used by hadoop JDK, pid file and log file directories

core-site.xml: specifies parameters relevant to all hadoop daemons and clients

hdfs-site.xml: specifies parameters used by HDFS daemons and clients

mapred-site.xml: specifies parameters uesd by MR daemons and clients

Configuration Files - 2

log4j.properties: contains all log config info

masters: optional, newline seperated list of machines that run the SNN. Used by the start-*.sh helper scripts only

slaves: optional, newline separated list of machine names that run the DN/TT daemons. Used by the start-*.sh helper scripts only

dfs.include: optional, newline separated list of machine names permitted to connect to the NN

dfs.exclude: optional, newline separated list of machines not permitted to connect to the NN

Configuration Files - 3

fair-scheduler.xml: optional, used to specify resource pools and settings for the Fair Scheduler plugin for MR

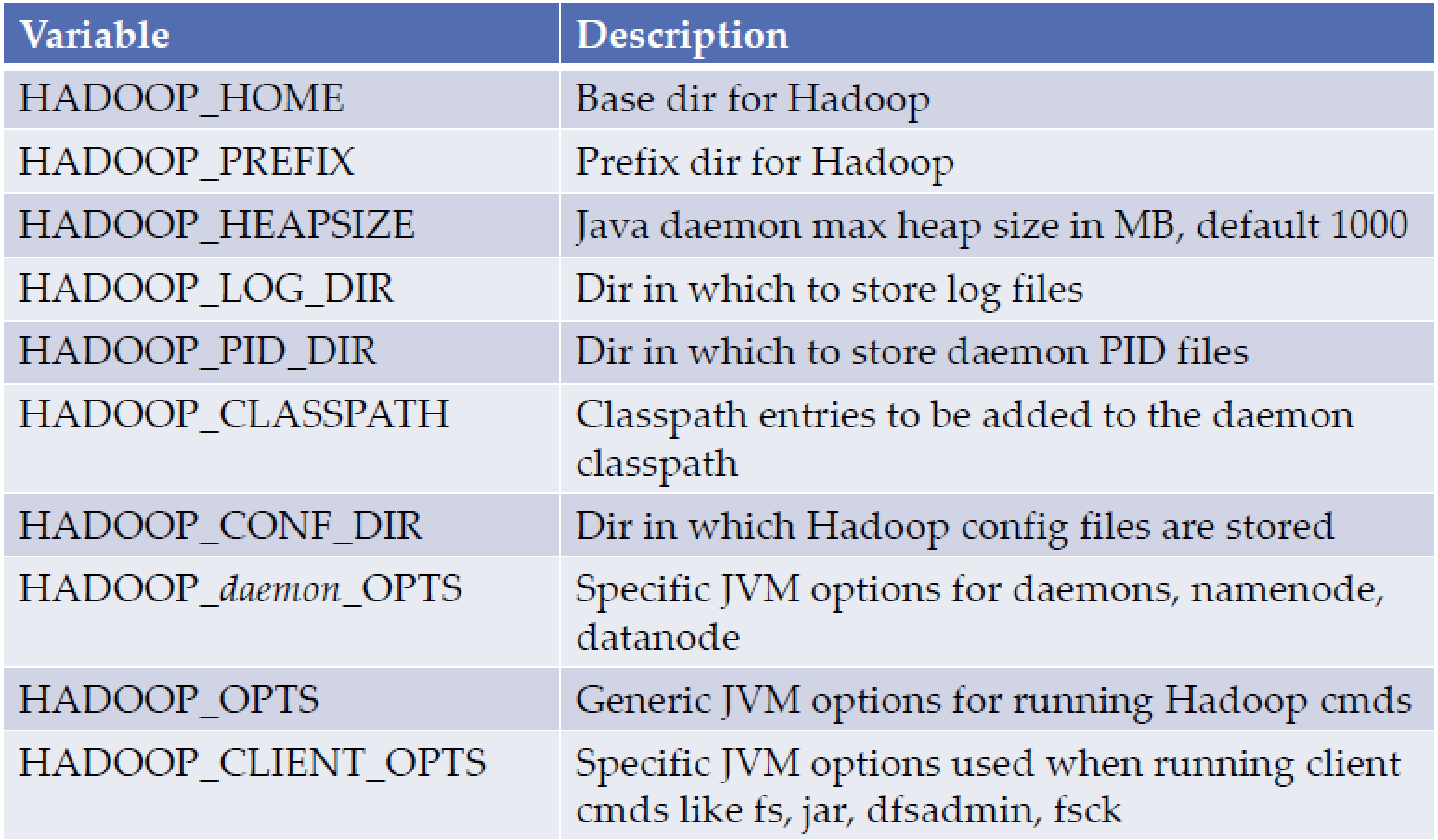
capacity-scheduler.xml: optional, specifies queues and settings for the Capacity Scheduler plugin for MR

hadoop-policy.xml: defines which uesrs/groups are permitted to invoke specific RPC functions when communicating with Hadoop

mapred-queue-acls.xml: defines which users/groups are permitted to submit jobs to which MR job queues

taskcontroller.cfg: Java property file that defines values used by the setuid task-controller MR helper program used when running in secure mode

Hadoop Environment Variables



Variable	Description
HADOOP_HOME	Base dir for Hadoop
HADOOP_PREFIX	Prefix dir for Hadoop
HADOOP_HEAPSIZE	Java daemon max heap size in MB, default 1000
HADOOP_LOG_DIR	Dir in which to store log files
HADOOP_PID_DIR	Dir in which to store daemon PID files
HADOOP_CLASSPATH	Classpath entries to be added to the daemon classpath
HADOOP_CONF_DIR	Dir in which Hadoop config files are stored
HADOOP_*daemon*_OPTS	Specific JVM options for daemons, namenode, datanode
HADOOP_OPTS	Generic JVM options for running Hadoop cmds
HADOOP_CLIENT_OPTS	Specific JVM options used when running client cmds like fs, jar, dfsadmin, fsck

HDFS – Identification & Location

Note: All files are in hdfs-site.xml unless noted otherwise

fs.default.name / fs.defaultFS: (core-site.xml) URL that specifies the default filesystem used by clients. This tells the NN what IP and port to bind on and tells the DNs where to heartbeat. Example: hdfs://NNhostname:port. Default port is 8020.

dfs.name.dir / dfs.namenode.name.dir: specifies a comma separated list of local dirs (no spaces) in which NN should store a copy of HDFS metadata. Use at least 2 internal disks and a NFS mount. A full copy of the metadata is stored in each dir. Metadata size is usually well under 1 TB. Warning: default is a tmp dir.

dfs.data.dir / dfs.datanode.data.dir: indicates where DNs should store HDFS block data via a comma separated list. DN round robins blocks between disks

fs.checkpoint.dir / dfs.namenode.checkpoint.dir: specifies comma separated list of dirs used by SNN to store FS metadata during checkpointing.

dfs.permissions.supergroup / dfs.permissions.superusergroup: users in this group are permitted to perform any FS operation. Default is supergroup, so there’s no confusion with linux. This privilege should not be given to everyday users.

HDFS – Optimization & Tuning

Note: All files are in hdfs-site.xml unless noted otherwise

dfs.block.size / dfs.blocksize: default block size for all newly created files. Does not affect files already in cluster. Clients can override this. Recommended is 128 MB.

dfs.datanode.du.reserved: specifies the amount of space in bytes to be researved on each disk in dfs.data.dir for MR spill data (mapred.local.dir). This is only needed if MR spill data is on same disks as DN blocks. Recommended setting is at least 10 GB per disk.

dfs.datanode.failed.volumes.tolerated: by default a DN will completely fail if even one of its local disks fails. This specifies the # of disks that are permitted to die before failing the entire DN. Recommended is 1 or 2. dfs.hosts: newline separated list of hostnames or IPs that are explicitly allowed to join the NN. If configured all others will be denied. By default, all DNs with the correct namespaceID (generated when HDFS is formatted) are allowed to join the cluster.

Fs.trash.interval: (core-site.xml) When trash is enabled, deleted files from the cmd line are moved to .Trash dir in user's HDFS home dir. This interval specifies the amount of time in minutes the file is retained in .Trash. Default is 0, so trash is disabled. Admins can explicitly empty trash with fs –expunge.

MR – Identification & Location

mapred.job.tracker: (mapred-site.xml) Hostname/IP and port of the JobTracker. JT listens on this pair for RPC communication and TTs use this to find the JT. Typical port used is 8021.

mapred.local.dir: stores MR intermediate/spill data. Recommended to use more than one dir. mapred.local.dir and dfs.data.dir can share disks (recommended for a multi-tenant cluster). If you use dedicated disks for mapred.local.dir, then there is no need to specify dfs.datanode.du.reserved.

MR – Optimization & Tuning

mapred.tasktracker.map & reduce.tasks.maximum: Allows admins to specify the max # of each type of task. Maps prefer data locality and use network as little as possible. Reduce have no locality preference and must always fetch data over network. Each task is run in a separate JVM. Start with 2 tasks for each CPU core.

io.sort.mb: Map task output is stored in a memory circular buffer before being written to disk. This specifies size of the buffer. When buffer fills, it spills to mapred.local.dir. In buffer map output is partitioned by key. After all the buffers spill, they are merged into larger files and served to reducers by the TT. Default is 100 MB.

mapred.jobtracker.taskScheduler: Specifies the Java class name of the scheduler plugin that should be used by the JT to decide which tasks execute in which order between jobs. Default is FIFO. Recommended is Fair or Capacity

mapred.reduce.tasks: Specifies the # of reduce tasks to use for the job. This is commonly overwritten by the Job. Default is 1. Should be changed to 50% of the total reducer slots on the cluster.

Thank You