7 Deadly Hadoop Misconfigurations

Kathleen Ting
kathleen@cloudera.com, @kate_ting

cloudera

How to Destroy Your Cluster with 7 Misconfigurations

cloudera

Who Am I?

- Kathleen Ting
  - Apache Sqoop Committer, PMC Member
  - Support Manager, Cloudera

cloudera

Agenda

- Ticket Breakdown
- What are Misconfigurations?
  - Memory Mismanagement
    - TT OOME
    - JT OOME
    - Native Threads
  - Thread Mismanagement
    - Fetch Failures
    - Replicas
  - Disk Mismanagement
    - No File
    - Too Many Files
- Cloudera Manager

cloudera

Agenda

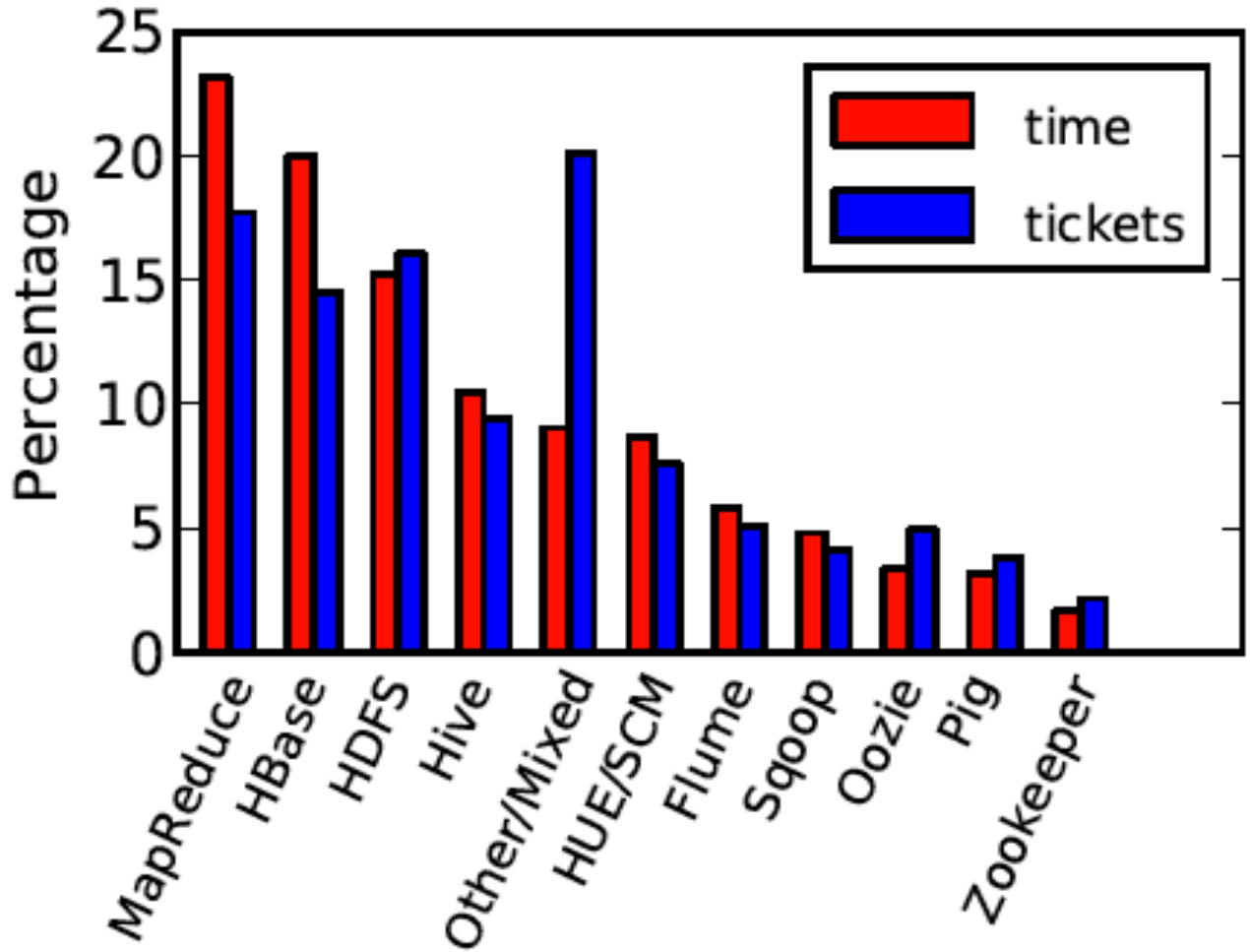
- Ticket Breakdown
- What are Misconfigurations?
  - Memory Mismanagement
    - TT OOME
    - JT OOME
    - Native Threads
  - Thread Mismanagement
    - Fetch Failures
    - Replicas
  - Disk Mismanagement
    - No File
    - Too Many Files
- Cloudera Manager

cloudera

Ticket Breakdown

- No one issue was more than 2% of tickets
- First symptom ≠ root cause
- Bad configuration goes undetected

cloudera



Breakdown of Tickets by Component

Agenda

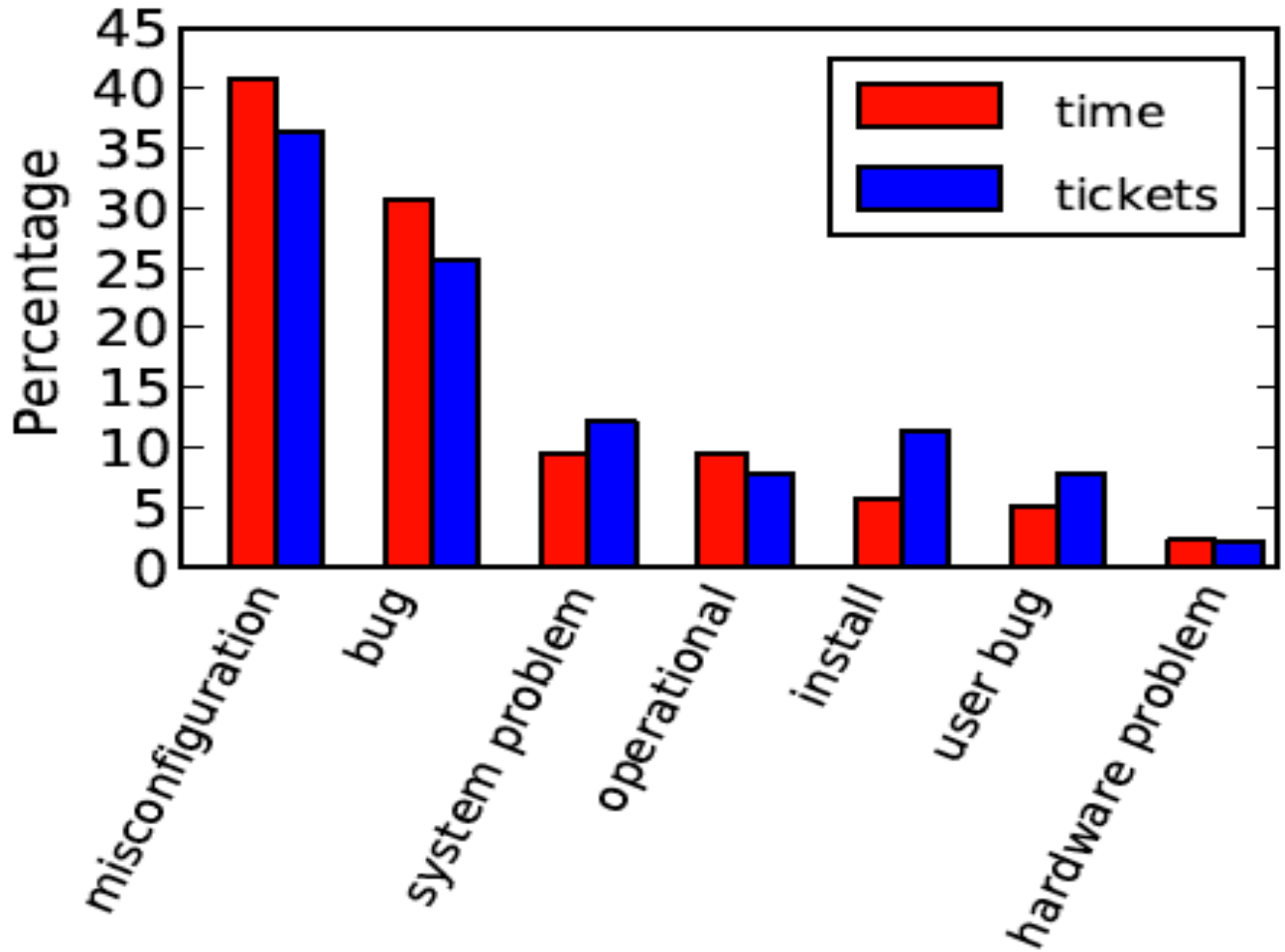
- Ticket Breakdown
- What are Misconfigurations?
  - Memory Mismanagement
    - TT OOME
    - JT OOME
    - Native Threads
  - Thread Mismanagement
    - Fetch Failures
    - Replicas
  - Disk Mismanagement
    - No File
    - Too Many Files
- Cloudera Manager

cloudera

What are Misconfigurations?

- Any diagnostic ticket requiring a change to Hadoop or to OS config files
- Comprise 35% of tickets
- e.g. resource-allocation: memory, file-handles, disk-space

cloudera



Problem Tickets by Category and Time

Why Care About Misconfigurations?

cloudera

The life of an over-subscribed MR/Hive cluster is nasty, brutish, and short.

(with apologies to Thomas Hobbes)

cloudera

What else you got?

FAILED: Execution Error, return code 2 from org.apache.hadoop.hive.ql.exec.MapRedTask

cloudera

Faulty MR Config Killed Hive

- Compared failing job.xml with successful job.xml
- Shuffle phase for stage 6 of that query was not completing due to io.sort.mb
- They had io.sort.mb = 112M
- Should be io.sort.mb = 512M

cloudera

Agenda

- Ticket Breakdown
- What are Misconfigurations?
  - Memory Mismanagement
    - TT OOME
    - JT OOME
    - Native Threads
  - Thread Mismanagement
    - Fetch Failures
    - Replicas
  - Disk Mismanagement
    - No File
    - Too Many Files
- Cloudera Manager

cloudera

1. Task Out Of Memory Error

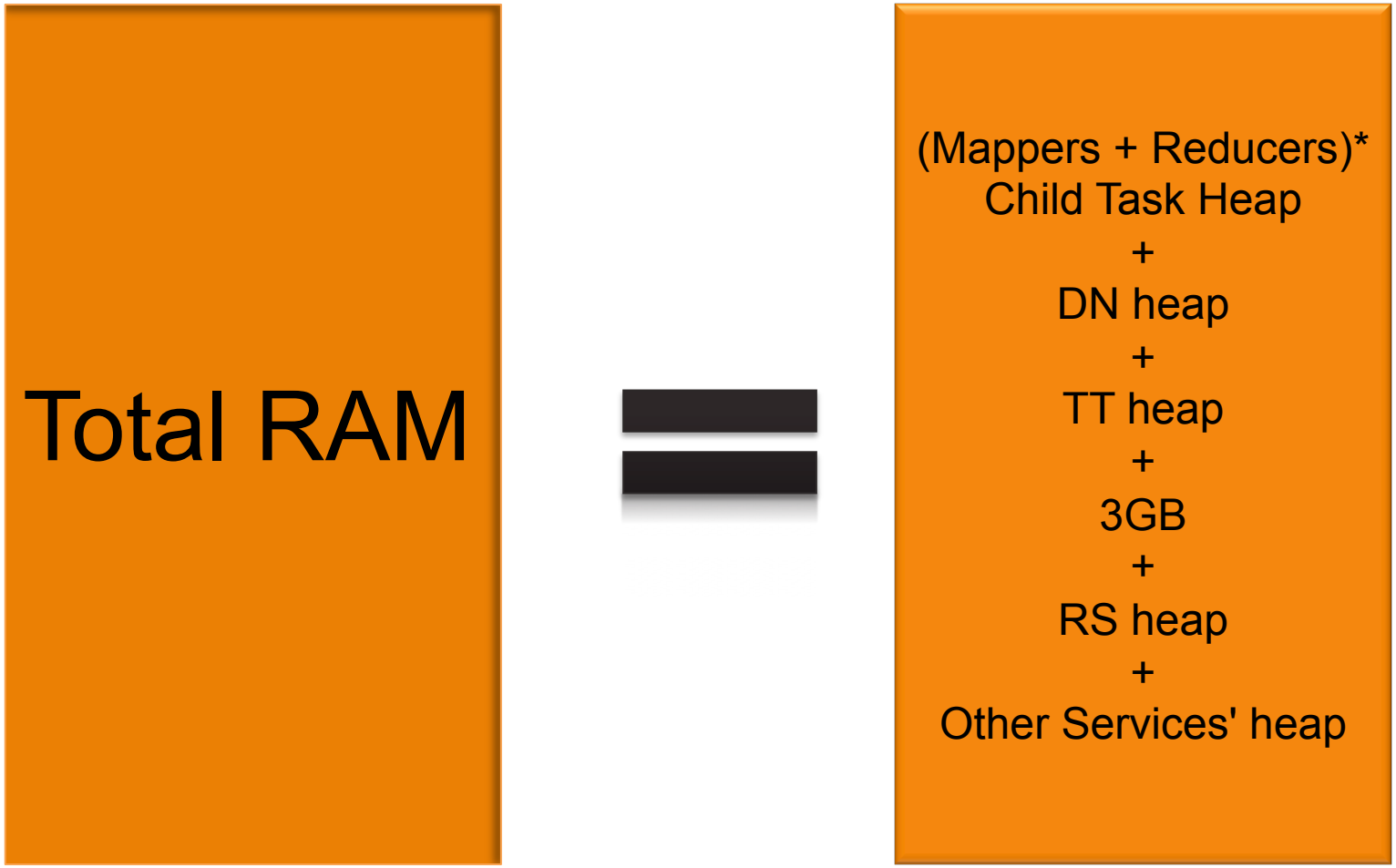
FATAL org.apache.hadoop.mapred.TaskTracker:
Error running child : java.lang.OutOfMemoryError: Java heap space
    at org.apache.hadoop.mapred.MapTask$MapOutputBuffer.<init>
(MapTask.java:781)
    at org.apache.hadoop.mapred.MapTask.runOldMapper
(MapTask.java:350)
    at org.apache.hadoop.mapred.MapTask.run(MapTask.java:307)
    at org.apache.hadoop.mapred.Child.main(Child.java:170)

cloudera

1. Task Out Of Memory Error

- What does it mean?
  - Memory leak in task code
- What causes this?
  - MR task heap sizes will not fit
- How can it be resolved?
  - mapred.child.ulimit (in KB) should be 2x higher than the heap size specified in mapred.child.java.opts (in MB) and left there to prevent runaway child task memory consumption
  - Set io.sort.mb (MB) = between ¼ to ½ of mapred.child.java.opts
    - e.g. 512M for io.sort.mb and 1G for mapred.child.java.opts
    - io.sort.mb = size of map buffer; smaller buffer => more spills
  - Use fewer mappers & reducers: 4:3 mapper:reducer ratio
    - mappers = # of cores on node
      - if not using HBase
      - if # disks = # cores

cloudera



cloudera

2. JobTracker Out of Memory Error

ERROR org.apache.hadoop.mapred.JobTracker: Job initialization failed:
java.lang.OutOfMemoryError: Java heap space
at org.apache.hadoop.mapred.TaskInProgress.<init>(TaskInProgress.java:122)
at org.apache.hadoop.mapred.JobInProgress.initTasks(JobInProgress.java:653)
at org.apache.hadoop.mapred.JobTracker.initJob(JobTracker.java:3965)
at org.apache.hadoop.mapred.EagerTaskInitializationListener$InitJob.run
(EagerTaskInitializationListener.java:79)
at java.util.concurrent.ThreadPoolExecutor$Worker.runTask
(ThreadPoolExecutor.java:886)
at java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:
908)
at java.lang.Thread.run(Thread.java:662)

cloudera

2. JobTracker Out of Memory Error

- What does it mean?
  - Total JT memory usage > allocated RAM
- What causes this?
  - Tasks too small
  - Too much job history

cloudera

2. JobTracker Out of Memory Error

- How can it be resolved?
  - Verify by summary output: sudo -u mapred jmap -J-d64 -histo:live <PID of JT>
  - Increase JT's heap space
  - Separate JT and NN
  - If going by the JT Web UI heap display, could be MAPREDUCE-3807
  - Set mapred.job.tracker.handler.count to ln(#TT)*20 and restart JT
  - Decrease mapred.jobtracker.completeuserjobs.maximum to 5
    - JT cleans up mem more often, reducing RAM

cloudera

3. Unable to Create New Native Thread

ERROR mapred.JvmManager: Caught Throwable in JVMRunner. Aborting TaskTracker.
java.lang.OutOfMemoryError: unable to create new native thread
at java.lang.Thread.start0(Native Method)
at java.lang.Thread.start(Thread.java:640)
at org.apache.hadoop.util.Shell.runCommand(Shell.java:234)
at org.apache.hadoop.util.Shell.run(Shell.java:182)
at org.apache.hadoop.util.Shell$ShellCommandExecutor.execute(Shell.java:375)
at org.apache.hadoop.mapred.DefaultTaskController.launchTask
(DefaultTaskController.java:127)
at org.apache.hadoop.mapred.JvmManager$JvmManagerForType
$JvmRunner.runChild(JvmManager.java:472)
at org.apache.hadoop.mapred.JvmManager$JvmManagerForType$JvmRunner.run
(JvmManager.java:446)

cloudera

3. Unable to Create New Native Thread

- What does it mean?
  - DN show up as dead even though processes are still running on those machines
- What causes this?
  - nproc default of 4200 processes is too low
- How can it be resolved?
  - Adjust /etc/security/limits.conf:
    - hbase soft/hard nproc 50000
    - hdfs soft/hard nproc 50000
    - mapred soft/hard nproc 50000
    - Restart DN, TT, JT, NN

cloudera

Agenda

- Ticket Breakdown
- What are Misconfigurations?
  - Memory Mismanagement
    - TT OOME
    - JT OOME
    - Native Threads
  - Thread Mismanagment
    - Fetch Failures
    - Replicas
  - Disk Mismanagement
    - No File
    - Too Many Files
- Cloudera Manager

cloudera

4. Too Many Fetch-Failures

INFO org.apache.hadoop.mapred.JobInProgress: Too many fetch-failures for output of task:

cloudera

4. Too Many Fetch-Failures

- What does it mean?
  - Reducer fetch operations fail to retrieve mapper outputs
  - Too many fetch failures occur on a particular TT
    - => blacklisted TT

- What causes this?
  - DNS issues
  - Not enough http threads on the mapper side for the reducers
  - JVM bug

cloudera

4. Too Many Fetch-Failures

- How can it be resolved?
  - mapred.reduce.slowstart.completed.maps = 0.80
    - Allows reducers from other jobs to run while a big job waits on mappers
  - tasktracker.http.threads = 80
    - Specifies # threads used by the TT to serve map output to reducers
  - mapred.reduce.parallel.copies = SQRT(NodeCount) with a floor of 10
    - Specifies # parallel copies used by reducers to fetch map output
  - Stop using 6.1.26 Jetty, which is fetch-failure prone and upgrade to CDH3u2 (MR-2980, MR-2524, MR-2529), MR-3184 (CDH3u3)
  - Set mapred.tasktracker.shuffle.fadvise to "false" on the TTs (only for CDH3u3)

cloudera

5. Not Able to Place Enough Replicas

WARN org.apache.hadoop.hdfs.server.namenode.FSNamesystem: Not able to place enough replicas

cloudera

5. Not Able to Place Enough Replicas

- What does it mean?
  - NN is unable to choose some DNs

cloudera

5. Not Able to Place Enough Replicas

- What causes this?
  - dfs replication > # avail DNs
    - # avail DNs is low due to low disk space
    - mapred.submit.replication default of 10 is too high
  - NN is unable to satisfy block placement policy
    - If # racks ≥ 2, a block has to exist in at least 2 racks
  - DN being decommissioned
  - DN has too much load on it (too many transfer threads in use)
  - Block-size unreasonably large
  - Not enough xcievers threads
    - Default 256 threads that DN can manage is too low
    - Note the accidental misspelling

cloudera

5. Not Able to Place Enough Replicas

- How can it be resolved?
  - Set dfs.datanode.max.xcievers = 4096 then restart DN (set to 512 if not using HBase)
  - Look for nodes down (or rack down)
  - Check disk space
    - Log dir may be filling up or a runaway task may fill up an entire disk
  - Rebalance

cloudera

Agenda

- Ticket Breakdown
- What are Misconfigurations?
  - Memory Exhaustion
    - TT OOME
    - JT OOME
    - Native Threads
  - Thread Exhaustion
    - Fetch Failures
    - Replicas
  - Data Space Exhaustion
    - No File
    - Too Many Files
- Cloudera Manager

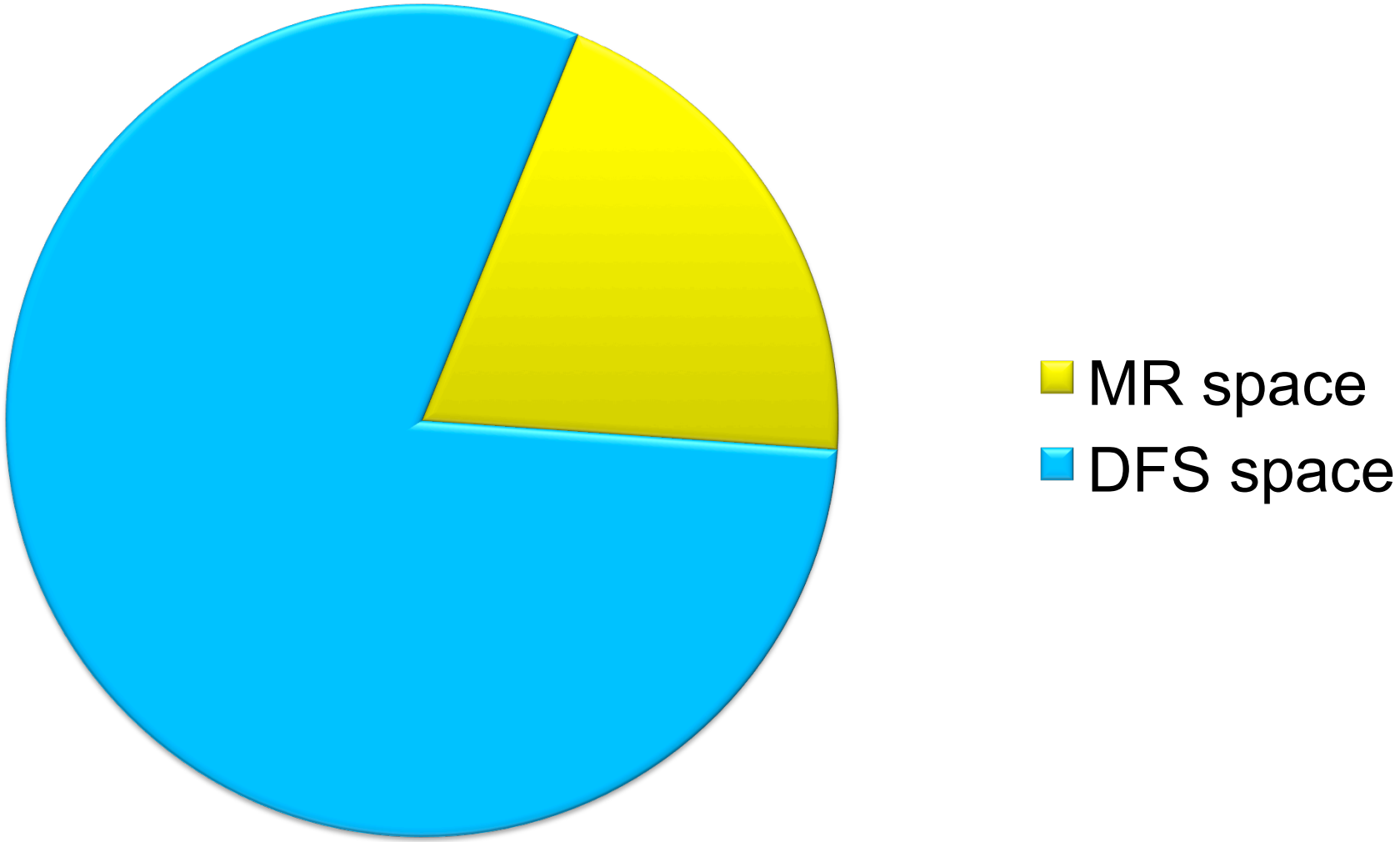
cloudera

6. No Such File or Directory

ERROR org.apache.hadoop.mapred.TaskTracker: Can not start task tracker because ENOENT: No such file or directory
at org.apache.hadoop.io.nativeio.NativeIO.chmod(Native Method)
at org.apache.hadoop.fs.RawLocalFileSystem.setPermission (RawLocalFileSystem.java:496)
at org.apache.hadoop.fs.RawLocalFileSystem.mkdirs(RawLocalFileSystem.java:319)
at org.apache.hadoop.fs.FilterFileSystem.mkdirs(FilterFileSystem.java:189)
at org.apache.hadoop.mapred.TaskTracker.initializeDirectories(TaskTracker.java:666)
at org.apache.hadoop.mapred.TaskTracker.initialize(TaskTracker.java:734)
at org.apache.hadoop.mapred.TaskTracker.<init>(TaskTracker.java:1431)
at org.apache.hadoop.mapred.TaskTracker.main(TaskTracker.java:3521)

cloudera

Total Storage



cloudera

6. No Such File or Directory

- What does it mean?
  - TT failing to start or jobs are failing
- What causes this?
  - Diskspace filling up on the TT disk
  - Permissions on the directories that TT writes to
- How can it be resolved?
  - Job logging dir dfs.datanode.du.reserved = 10% total vol
  - Permissions should be 755 and owner should be mapred for userlogs and mapred.local.dir

cloudera

7. Too Many Open Files

ERROR org.apache.hadoop.hdfs.server.datanode.DataNode: DatanodeRegistration(<ip address>:50010, storageID=<storage id>, infoPort=50075, ipcPort=50020):DataXceiver
java.io.IOException: Too many open files
    at sun.nio.ch.IOUtil.initPipe(Native Method)
    at sun.nio.ch.EPollSelectorImpl.<init>(EPollSelectorImpl.java:49)
    at sun.nio.ch.EPollSelectorProvider.openSelector(EPollSelectorProvider.java:18)
    at org.apache.hadoop.net.SocketIOWithTimeout$SelectorPool.get(SocketIOWithTimeout.java:407)
    at org.apache.hadoop.net.SocketIOWithTimeout$SelectorPool.select(SocketIOWithTimeout.java:322)
    at org.apache.hadoop.net.SocketIOWithTimeout.doIO(SocketIOWithTimeout.java:157)
    at org.apache.hadoop.net.SocketInputStream.read(SocketInputStream.java:155)
    at org.apache.hadoop.net.SocketInputStream.read(SocketInputStream.java:128)
    at java.io.BufferedInputStream.fill(BufferedInputStream.java:218)
    at java.io.BufferedInputStream.read(BufferedInputStream.java:237)
    at java.io.DataInputStream.readShort(DataInputStream.java:295)
    at org.apache.hadoop.hdfs.server.datanode.DataXceiver.run(DataXceiver.java:97)

cloudera

7. Too Many Open Files

- What does it mean?
  - Hitting open file handles limit of user account running Hadoop
- What causes this?
  - Nofile default of 1024 files is too low
- How can it be resolved?
  - Adjust /etc/security/limits.conf:
    - hdfs - nofile 32768
    - mapred - nofile 32768
    - hbase - nofile 32768
    - Restart DN, TT, JT, NN
    - Upgrade to 1.0.6 JSVC, which fixed DAEMON-192

cloudera

Agenda

- Ticket Breakdown
- What are Misconfigurations?
  - Memory Mismanagement
    - TT OOME
    - JT OOME
    - Native Threads
  - Thread Mismanagement
    - Fetch Failures
    - Replicas
  - Disk Mismanagement
    - No File
    - Too Many Files
- Cloudera Manager

cloudera

Configuration History & Rollback

Save with notes: Save Changes

Compression
Advanced
Logs
Security
Metrics
JobTracker
Performance
Ports and Addresses
Paths
Jobs
Classes
Security
Plugins
Advanced
Logs
Metrics
Client
Jobs
Performance
Compression
Advanced
Monitoring

Property	Value	Description
Client Settings		
Default Number of Reduce Tasks per Job mapred.reduce.tasks	2 Reset to the default value: 1 Default Number of Reduce Tasks per Job is at least 50% of the number of reduce slots across all TaskTrackers. Suggested minimum value: 2	The default number of reduce tasks per job. Will be part of generated client configuration.
Number of Tasks to Run per JVM mapred.job.reuse.jvm.num.tasks	1	Number of tasks to run per JVM. If set to -1, there is no limit. Will be part of generated client configuration.
Map Tasks Speculative Execution mapred.map.tasks.speculative.execution	☐	If enabled, multiple instances of some map tasks may be executed in parallel.
Reduce Tasks Speculative Execution mapred.reduce.tasks.speculative.execution	☐	If enabled, multiple instances of some reduce tasks may be executed in parallel.
Number of Map Tasks to Complete Before Reduce Tasks mapred.reduce.slowstart.completed.maps	0.8	Fraction of the number of map tasks in the job which should be completed before reduce tasks are scheduled for the job.

Additional Resources

- Avoiding Common Hadoop Administration Issues: http://www.cloudera.com/blog/2010/08/avoiding-common-hadoop-administration-issues/
- Tips for Improving MapReduce Performance: http://www.cloudera.com/blog/2009/12/7-tips-for-improving-mapreduce-performance/
- Basic Hardware Recommendations: http://www.cloudera.com/blog/2010/03/clouderas-support-team-shares-some-basic-hardware-recommendations/
- Cloudera Knowledge Base: http://ccp.cloudera.com/display/KB/Knowledge+Base

cloudera

Takeaways

- Configuration is up to you.
- Misconfigurations are hard to diagnose.
- Get it right the first time with tools.
  - "Yep - we were able to download/install/ configure/setup a Cloudera Manager cluster from scratch in minutes :)"

cloudera