

Outline

Module 1 : 大數據簡介

Module 2 : Hadoop Ecosystem介紹

Module 3 : Hadoop 平台安裝

Module 4 : Hadoop 分散式檔案系統 (HDFS)

Module 5 : Hadoop MapReduce

Module 6 : Apache Hive

Module 7 : Sqoop與Flume

Module 8 : Apache Spark

Module 9 : Spark 平台安裝

Module 10 : RDD — Resilient distributed dataset

Module 11 : Scala 程式開發基礎

Module 12 : Spark SQL 及 DataFrame

Module 13 : Spark 機器學習函式庫(MLlib)



HDSF簡介

- ▶ Hadoop = HDFS + MapReduce
- ▶ HDFS = **H**adoop **D**istributed **F**ile **S**ystem的縮寫
- ▶ 在分散式的儲存環境中，提供單一的目錄系統
- ▶ 課程內容
 - HDSF的運作原理及架構
 - HDFS的指令操作
 - HDSF的安全性
 - 程式存取HDFS



YARN/Map Reduce V2



Hadoop Distributed File System



HDFS的設計概念

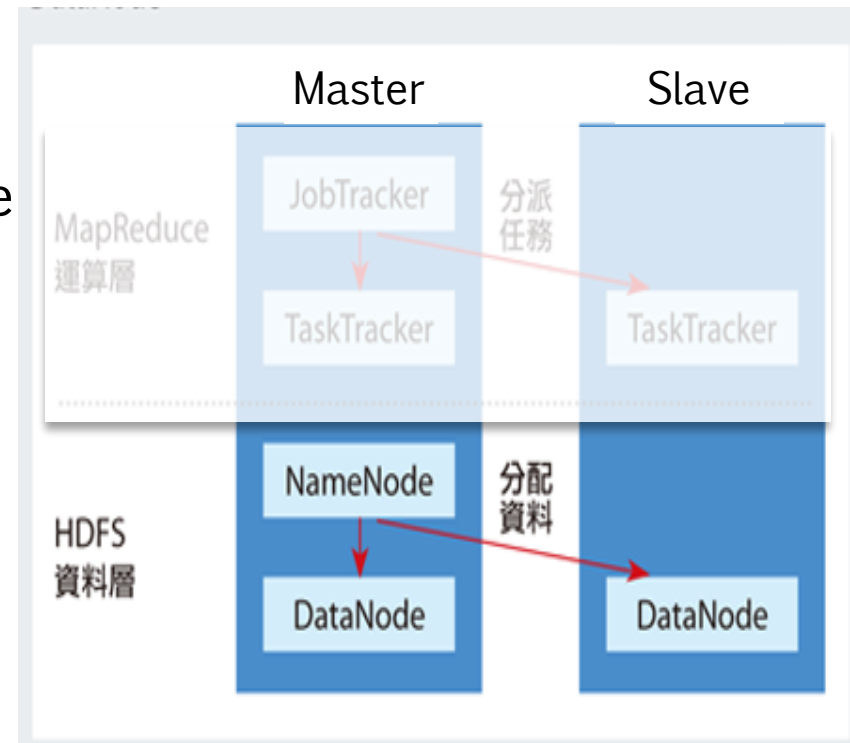
- ▶ 超大型分散式檔案系統(Distributed File System)
 - 上萬節點、上億個檔案、PB等級資料
- ▶ 簡單一致的存取模型
 - Write-once-read-many存取模式
 - 檔案建立後即不允許修改(僅可覆寫)
 - 簡化一致性處理問題
- ▶ 以Block為單位儲存
 - 檔案被分為多個Block、Block大小預設為128MB
 - 每個Block被複製多個複本(replica)分散存放在不同資料節點(DataNode)上

HDFS的設計概念

- ▶ 運算作業儘可能和作業資料在一起(Data Locality)
 - 搬運作業成本比搬運資料來得小
- ▶ 可建置於OS的原生檔案系統、標準化的量產硬體(Commodity Hardware)上(Low Cost)
- ▶ 串流式(Streaming)資料存取
 - 優先考慮大量資料存取(高Throughput)，而非因應低延遲的檔案存取需求
 - 批次處理(Batch Processing)最佳化
- ▶ 多複本儲存
 - 預設複本為3份(由hdfs-site.xml控制)

HDFS 架構說明

- ▶ Master/Slave
 - 實體機器
- ▶ Name Node / Data Node
 - 運行的Daemon
- ▶ Master Daemon
 - Name Node、Secondary Name Node
 - 負責資源 / 工作調配
- ▶ Slave Daemon
 - Data Node
 - 負責執行任務



ref: <http://www.ithome.com.tw/node/73978>

HDFS角色說明 — Name Node

- ▶ 發起資料的讀取 / 寫入工作
- ▶ 儲存資料的Metadata
 - 檔案名稱、權限、目錄
 - Block儲存於那個Data Node
- ▶ 若只有一個Name Node，可能發生單點失敗問題
- ▶ Metadata可儲存於記憶體與磁碟中
 - 建議定期備份Metadata

HDFS角色說明 — Name Node

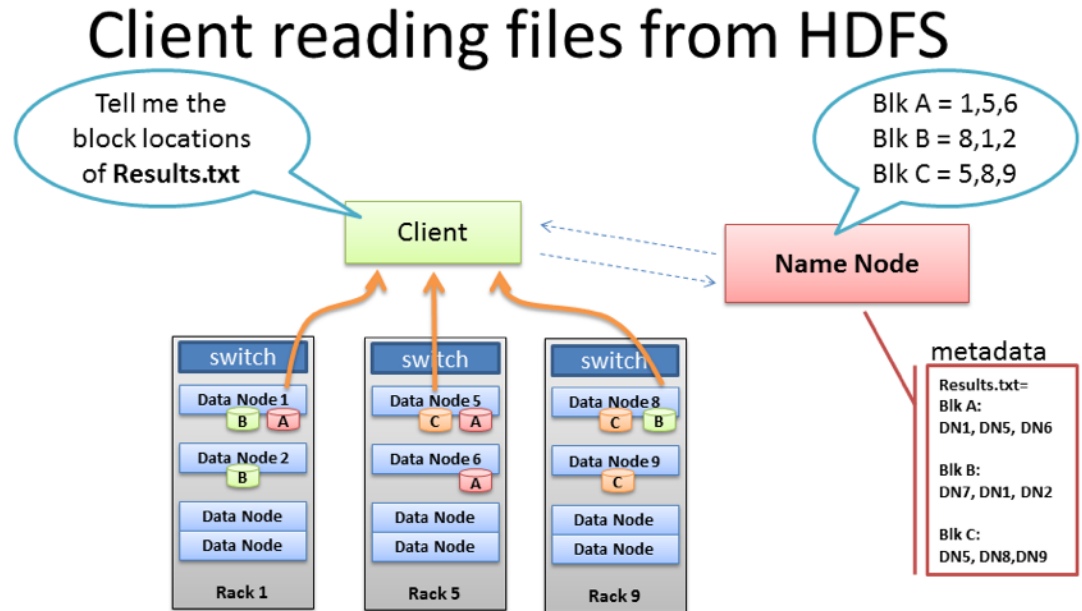
- ▶ Name Node將Metadata儲存於記憶體與磁碟中
 - 需大量記憶體
 - metadata存於兩個檔案
 - fsimage(檔案的Snapshot)
 - edit log(自snapshot後HDFS的變更記錄)
 - fsimage與edit log會定期合併
 - 當Name Node重新啟動時
 - 由Secondary Name Node合併
 - Name Node的儲存建議
 - 使用RAID或NFS進行存放
 - fsimage 需定期備份(每天 / 每週)

HDFS角色說明 – Secondary Name Node

- ▶ 非即時備援Name Node
 - 復原約需花費一小時
- ▶ 負責檢核(Checkpoint) Name Node
- ▶ 記憶體需求和Name Node相當

HDFS角色說明 — Data Node

- ▶ 存放實際資料的節點
- ▶ 用戶端程式向 Data Node 直接存取資料



- Client receives Data Node list for each block
- Client picks first Data Node for each block
- Client reads blocks sequentially

BRAD HEDLUND .com

Name Node管理Data Node的機制

▶ Heartbeats

- Data Node會定期向Name Node送Heartbeats
- 若Data Node 30秒沒心跳，Name Node會告知Client向其它Data Node取資料
- 若Data Node 10分鐘沒心跳，Name Node會將該Data Node中的資料複製到其它Data Node



Name Node管理Data Node的機制

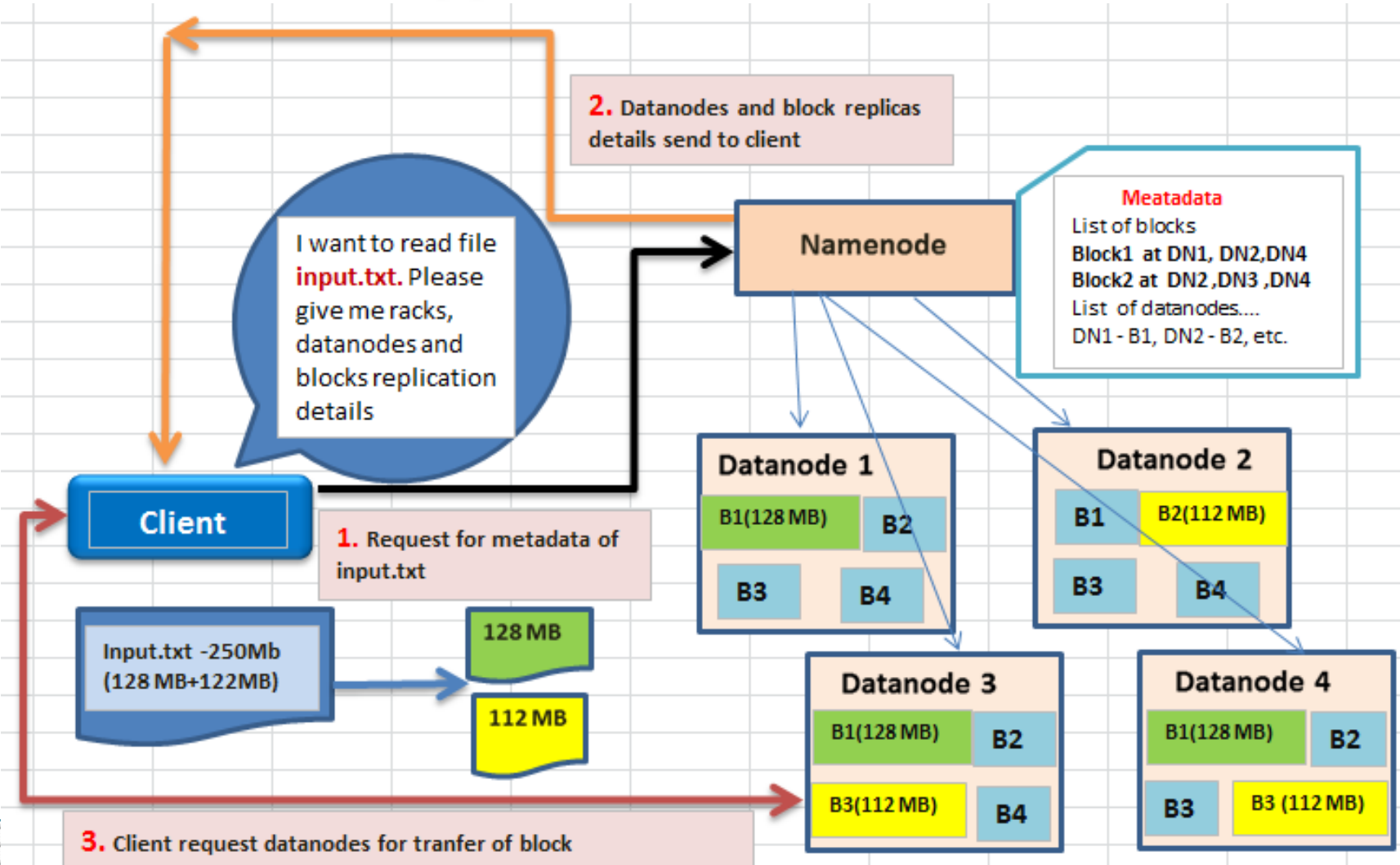
▶ Block Reports

- 每小時Data Node會跟Name Node同步目前內含的Block
- 當Name Node發現Data Node內的Block與當初的分配不符，會重新配置資料，確認符合Replica設定

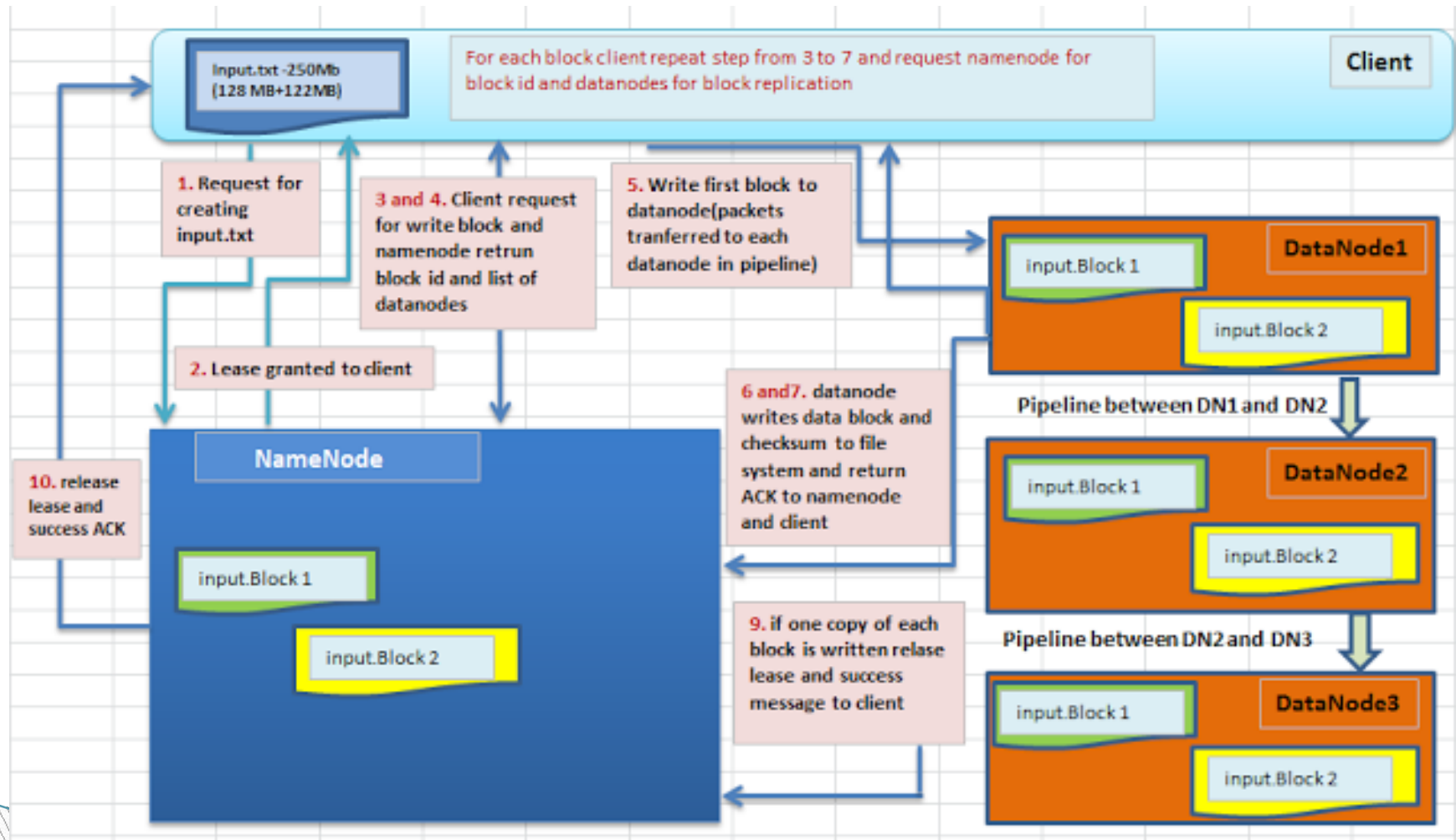
▶ Checksums

- 資料寫入時，會在Block旁增加checksums資訊
- 當讀取時發現資料的checksum不符，則改讀其它複本

HDFS的運作 – Read



HDFS的運作 – Write



HDFS操作介紹

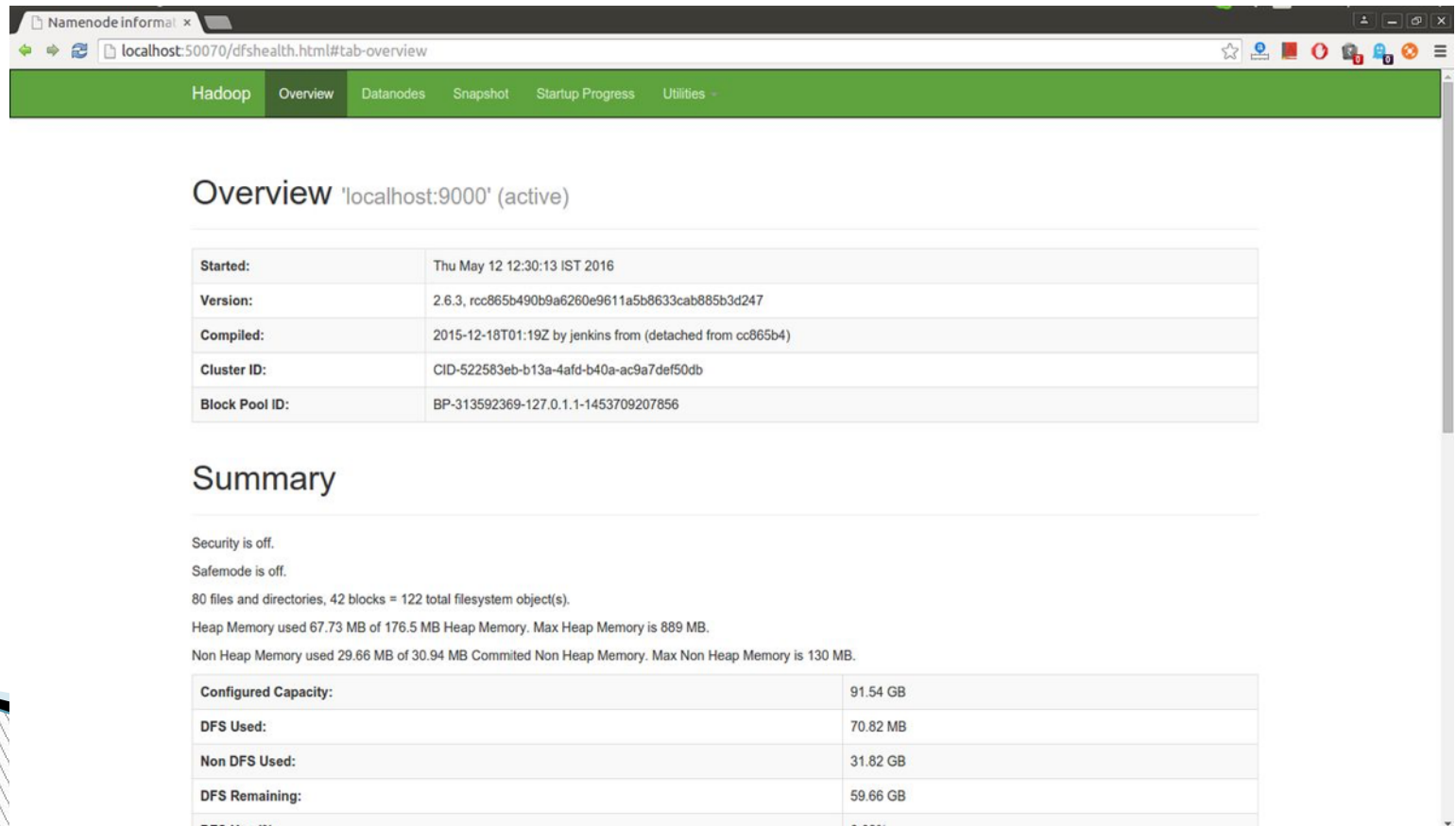
- ▶ 管理者可透過在Terminal視窗下hadoop fs系列指令手動管理HDFS內容
- ▶ 指令格式
 - hadoop fs -操作指令

HDFS操作介紹 — 常用指令

- ▶ 建立目錄－`hadoop fs -mkdir 目錄名稱`
- ▶ 資料上傳－`hadoop fs -copyFromLocal 本機檔案路徑 hdfs目錄路徑`
- ▶ 檢視目錄－`hadoop fs -ls hdfs目錄路徑`
- ▶ 檢視檔案內容－`hadoop fs -cat hdfs檔案路徑`
- ▶ 檢視空間大小－`hadoop fs -df`
- ▶ 檢視目錄大小－`hadoop fs -du hdfs目錄路徑`
- ▶ 下載資料－`hadoop fs -get hdfs檔案路徑 本機目錄`
- ▶ 刪除檔案－`hadoop fs -rm [-f] hdfs檔案路徑`
- ▶ 刪除目錄－`hadoop fs -rm -R [-f] hdfs目錄路徑`

Browse HDFS via Web

- ▶ 在Hadoop的master上瀏覽<http://localhost:50070>
 - Utilities -> browse file system



The screenshot shows the Hadoop NameNode web interface. The browser address bar displays `localhost:50070/dfshealth.html#tab-overview`. The interface has a green navigation bar with tabs: Hadoop, Overview, Datanodes, Snapshot, Startup Progress, and Utilities. The main content area is titled "Overview 'localhost:9000' (active)".

Overview 'localhost:9000' (active)

Started:	Thu May 12 12:30:13 IST 2016
Version:	2.6.3, rcc865b490b9a6260e9611a5b8633cab885b3d247
Compiled:	2015-12-18T01:19Z by jenkins from (detached from cc865b4)
Cluster ID:	CID-522583eb-b13a-4afd-b40a-ac9a7def50db
Block Pool ID:	BP-313592369-127.0.1.1-1453709207856

Summary

Security is off.
Safemode is off.
80 files and directories, 42 blocks = 122 total filesystem object(s).
Heap Memory used 67.73 MB of 176.5 MB Heap Memory. Max Heap Memory is 889 MB.
Non Heap Memory used 29.66 MB of 30.94 MB Committed Non Heap Memory. Max Non Heap Memory is 130 MB.

Configured Capacity:	91.54 GB
DFS Used:	70.82 MB
Non DFS Used:	31.82 GB
DFS Remaining:	59.66 GB
DFS Health:	OK

操作練習

- ▶ 下載維基百科的瀏覽記錄(<https://dumps.wikimedia.org/other/pagecounts-raw/2016/2016-01/pagecounts-20160101-000000.gz>)
- ▶ 解壓縮.gz檔
- ▶ 透過hadoop fs指令在HDFS上建立data資料夾
- ▶ 透過hadoop fs指令將解壓縮後的檔案上傳至HDFS
- ▶ 透過指令瀏覽 HDFS上的data資料夾內容
- ▶ 透過Web介面瀏覽 HDFS上的data資料夾內容
- ▶ 透過hadoop fs指令刪除HDFS上的data資料夾

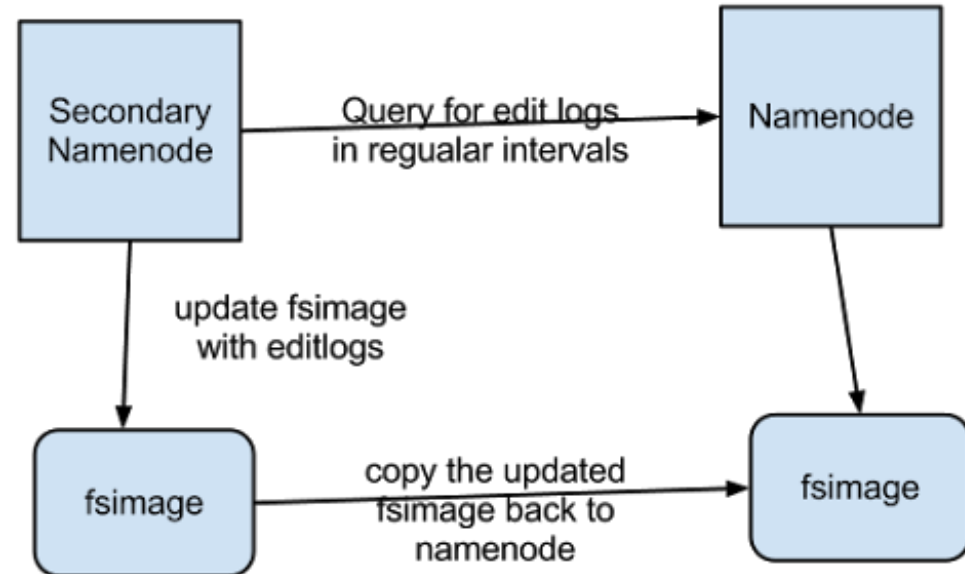
不適用HDFS的情境

- ▶ HDFS 不適用於存放小檔案
 - 小檔案仍會以Block為單位存放，浪費存儲空間
- ▶ HDFS 不適用於隨機存取(Random Access)
 - 由開頭或結尾循序讀取
- ▶ 不建議以SAN或NAS作為HDFS背後的儲存機制
 - 巨量資料傳輸會造成瓶頸
- ▶ HDFS 不是關聯式資料庫
 - 分散式 VS 集中式
 - MapReduce VS SQL
 - 循序讀取 VS 隨機讀取
 - 沒有ACID的特性



Secondary Name Node

- ▶ 非Name Node的即時備源(not hot standby)
- ▶ 監控Name Node的執行狀態
- ▶ 排程(每小時執行)
 - 由Name Node備份fsimage及edit log
 - 將fsimage及edit log合併為大的fsimage



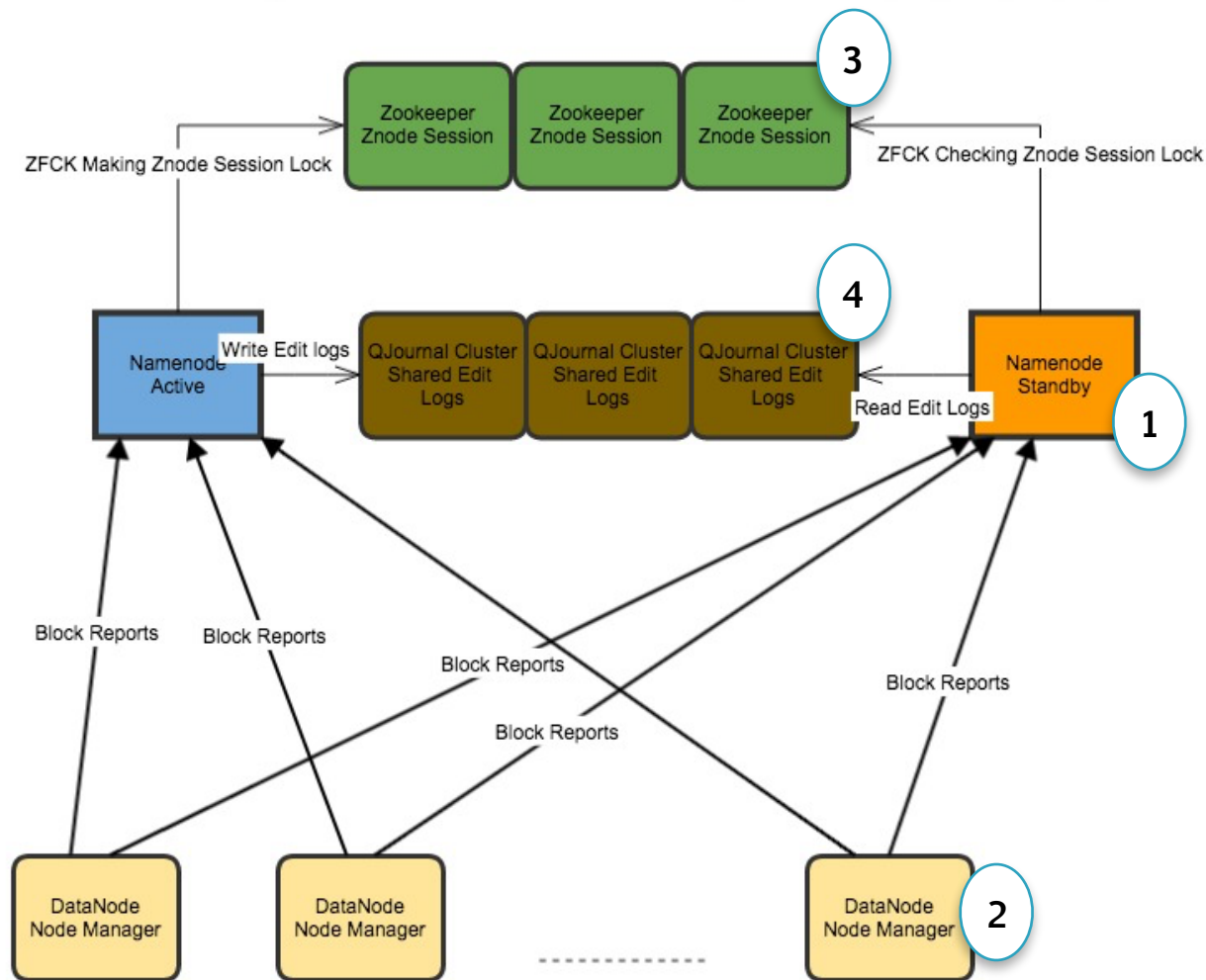
Secondary Name Node備源

- ▶ Secondary Name Node需要跟Name Node相當的記憶體
 - 最好與Name Node分開安裝於不同機器上
- ▶ 若Name Node損毀時可使用Secondary Name Node取代
 - 回復時間約晚一小時

HDFS High Availability

- ▶ <https://hadoop.apache.org/docs/r2.7.1/hadoop-project-dist/hadoop-hdfs/HDFSHighAvailabilityWithQJM.html>
- ▶ 要解決的問題
 - Name Node可能發生單點失敗問題
 - Secondary Name Node無法即時備源
- ▶ Hadoop HA
 - 建置Standby Name Node
 - Client讀寫指令同時發送給Active及Standby NN
 - DataNode Block Report同時與Active及Standby NN同步
 - NN的metadata(fsimage及edit log)透過共享磁碟儲存

HDFS HA Architecture



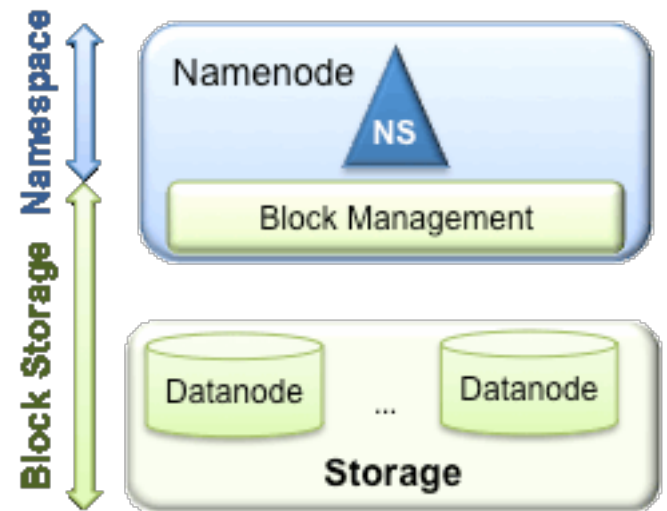
1. 建置Standby NN
2. DN同時與兩台NN作 Block Report
3. 以Zookeeper自動偵測及切換(或下指令手動切換)
4. 透過NFS或QJM共享 metadata(避免 metadata損毀)

QJM : Quorum Journal Manager

ref: <http://prashantblogs4all.wordpress.com/2014/09/04/setting-up-a-hadoop-namenode-ha/>

HDFS Federation

- ▶ <https://hadoop.apache.org/docs/r2.7.2/hadoop-project-dist/hadoop-hdfs/Federation.html>
- ▶ 要解決的問題
 - NameNode/Namespace水平延展性不足
 - 單一NN有效能問題
 - 僅支援60K個Task
 - 單一NN無法作到Job的隔離
 - 大的Job影響其它Job執行

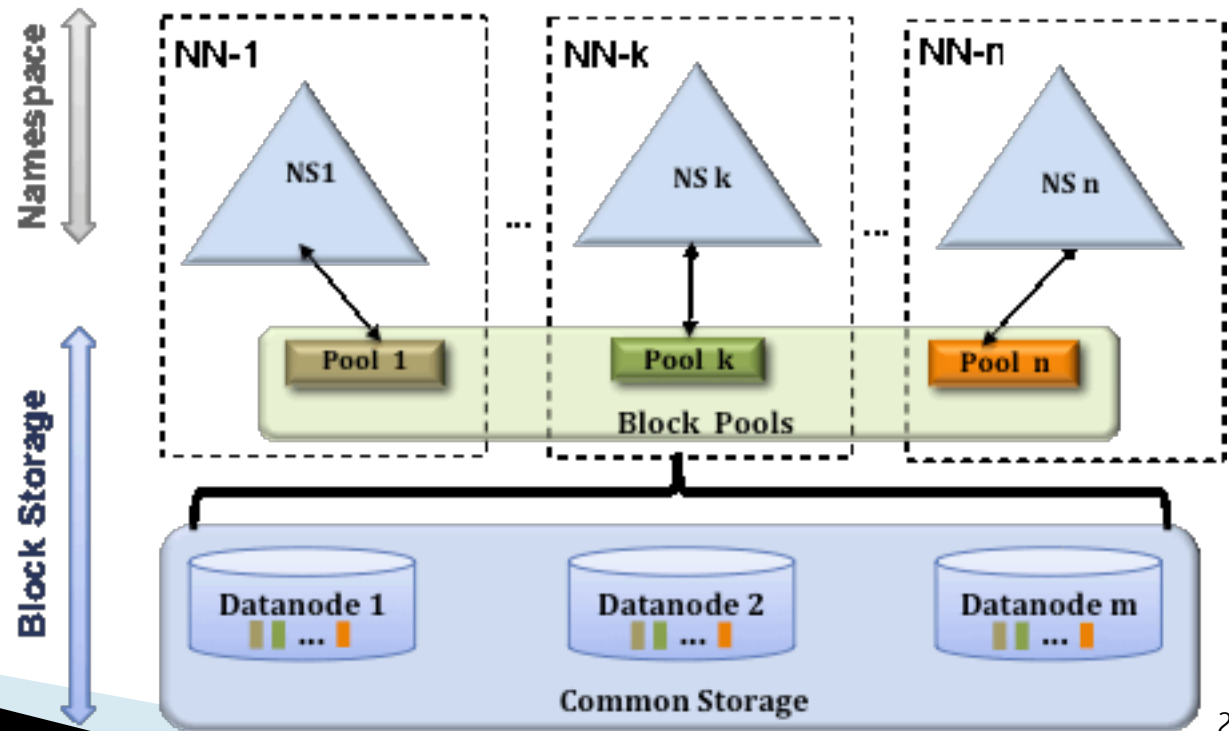


HDFS單一NN架構

HDFS Federation Architecture

- ▶ 透過設定提供建置多個NN功能
- ▶ 提供NN的水平延展性
- ▶ 可依不同用途規劃所屬Namespace
 - by user
 - by app

[注意] 每個NN各自為獨立的Namespace，無法彼此備源(不是HA架構)



HDFS Security

- ▶ HDFS透過實作類似POSIX的權限模型來管理文件及目錄 (User、Group、Others)
 - 644 (rw-r--r--)
 - 755 (rwxr-xr-x)
- ▶ x表示可瀏覽
- ▶ w表示可刪除
- ▶ 指令
 - `hadoop fs -chmod 755 HDFS檔案路徑`
 - `hadoop fs -chgrp groupName HDFS檔案路徑`

HDFS Security

- ▶ User Identity
 - simple(預設) — 由作業系統來決定process的身份，相當於whoami及groups
 - kerberos — 由Kerberos credentials決定process的執行身份