Hadoop Distributed File System

Outline

- HDFS 的定義 ?
- HDFS 的特色?
- HDFS 的架構 ?
- HDFS 運作方式 ?
- HDFS 如何達到其宣稱的好處 ?
- HDFS 功能?

HDFS ?

- Hadoop Distributed File System
  - Hadoop : 自由軟體專案，為實現Google的MapReduce架構
  - HDFS: Hadoop專案中的檔案系統
- 實現類似Google File System
  - GFS是一個易於擴充的分散式檔案系統，目的為對大量資料進行分析
  - 運作於廉價的普通硬體上，又可以提供容錯功能
  - 給大量的用戶提供總體性能較高的服務

設計目標 (1)

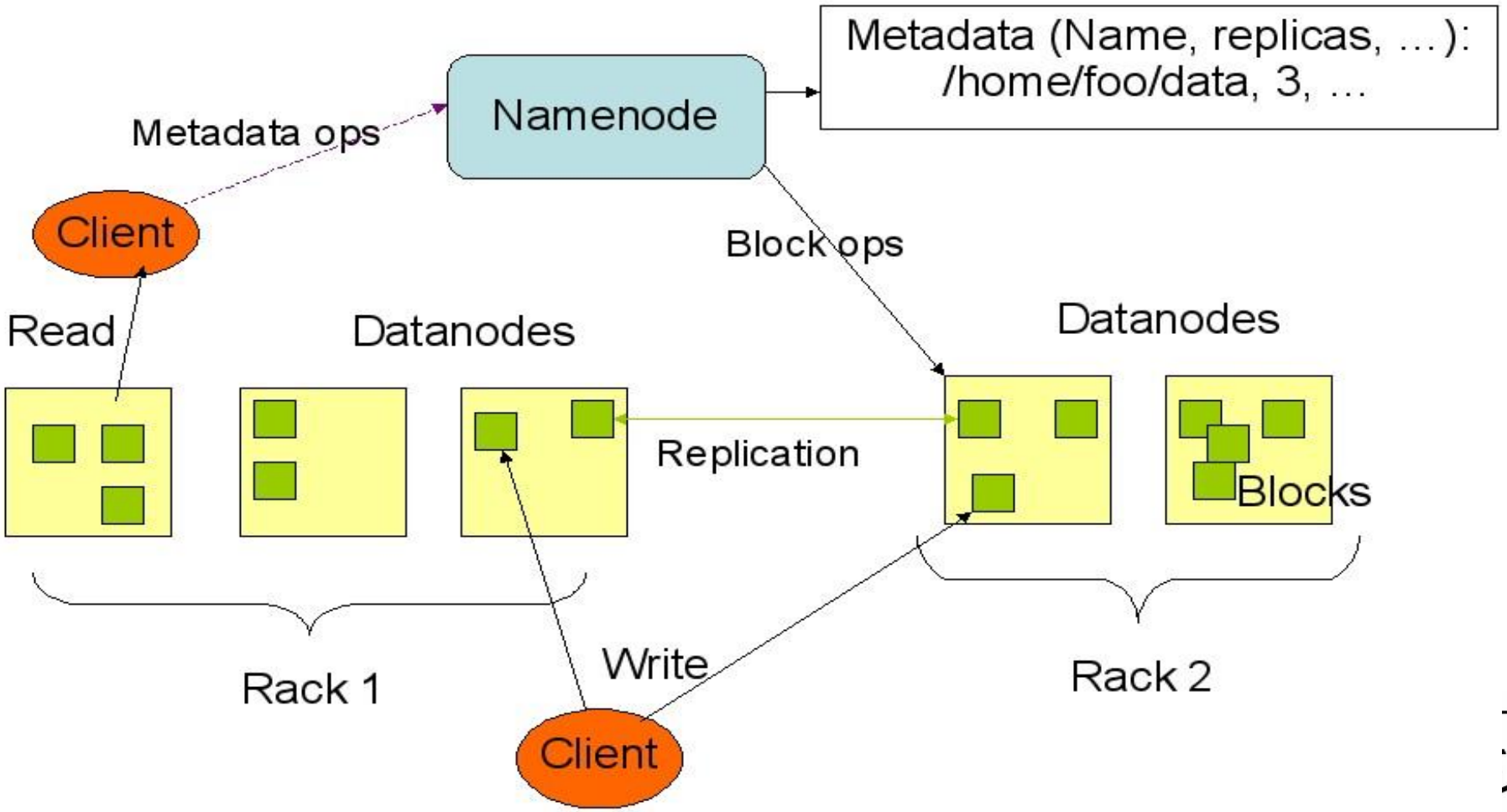
- 硬體錯誤容忍能力
  - 硬體錯誤是正常而非異常
  - 自動恢復或故障排除
- 串流式的資料存取
  - 批次處理多於用戶交互處理
  - **高Throughput** 而非低Latency
- 大規模資料集
  - 支援Perabytes等級的磁碟空間

設計目標 (2)

- 一致性模型
  - 一次寫入，多次存取
  - 簡化一致性處理問題
- 在地運算
  - **到資料的節點上計算** > 將資料從遠端複製過來計算
- 異質平台移植性
  - 即使硬體不同也可移植、擴充

管理資料

HDFS Architecture

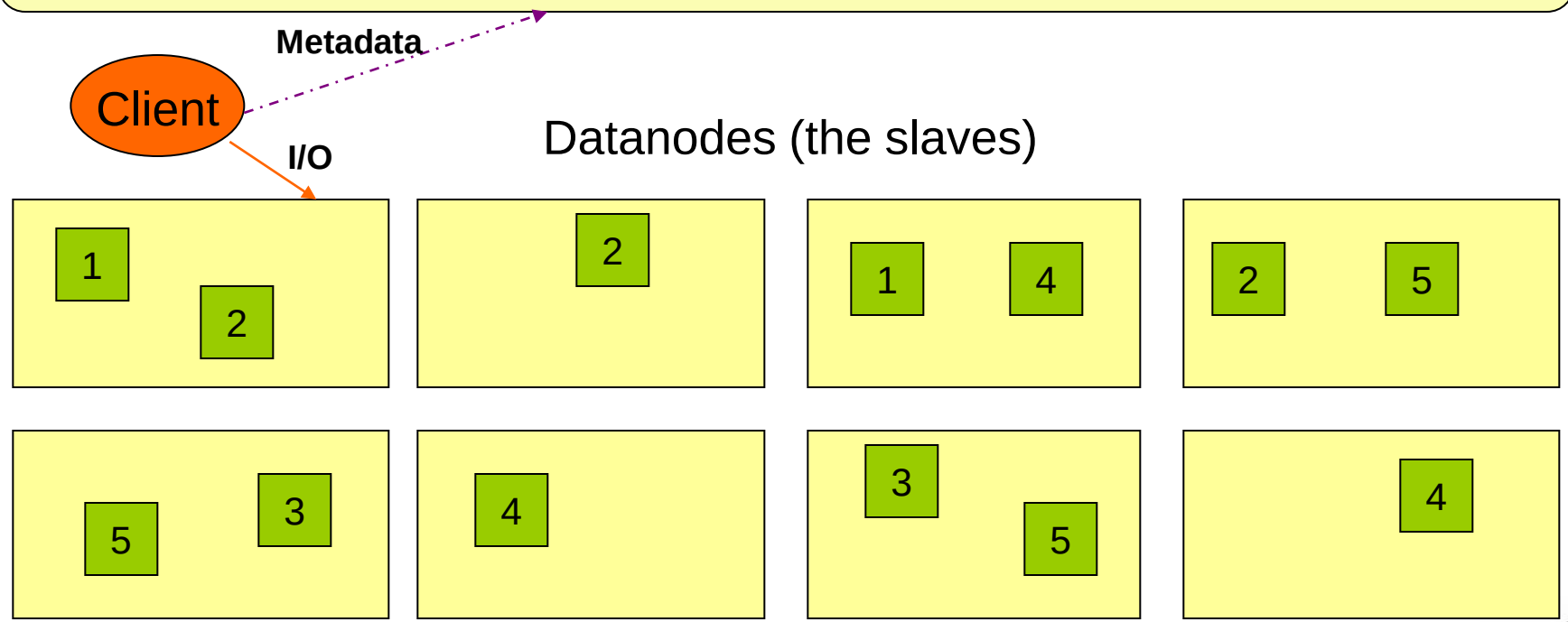


HDFS 運作

Namenode (the master)

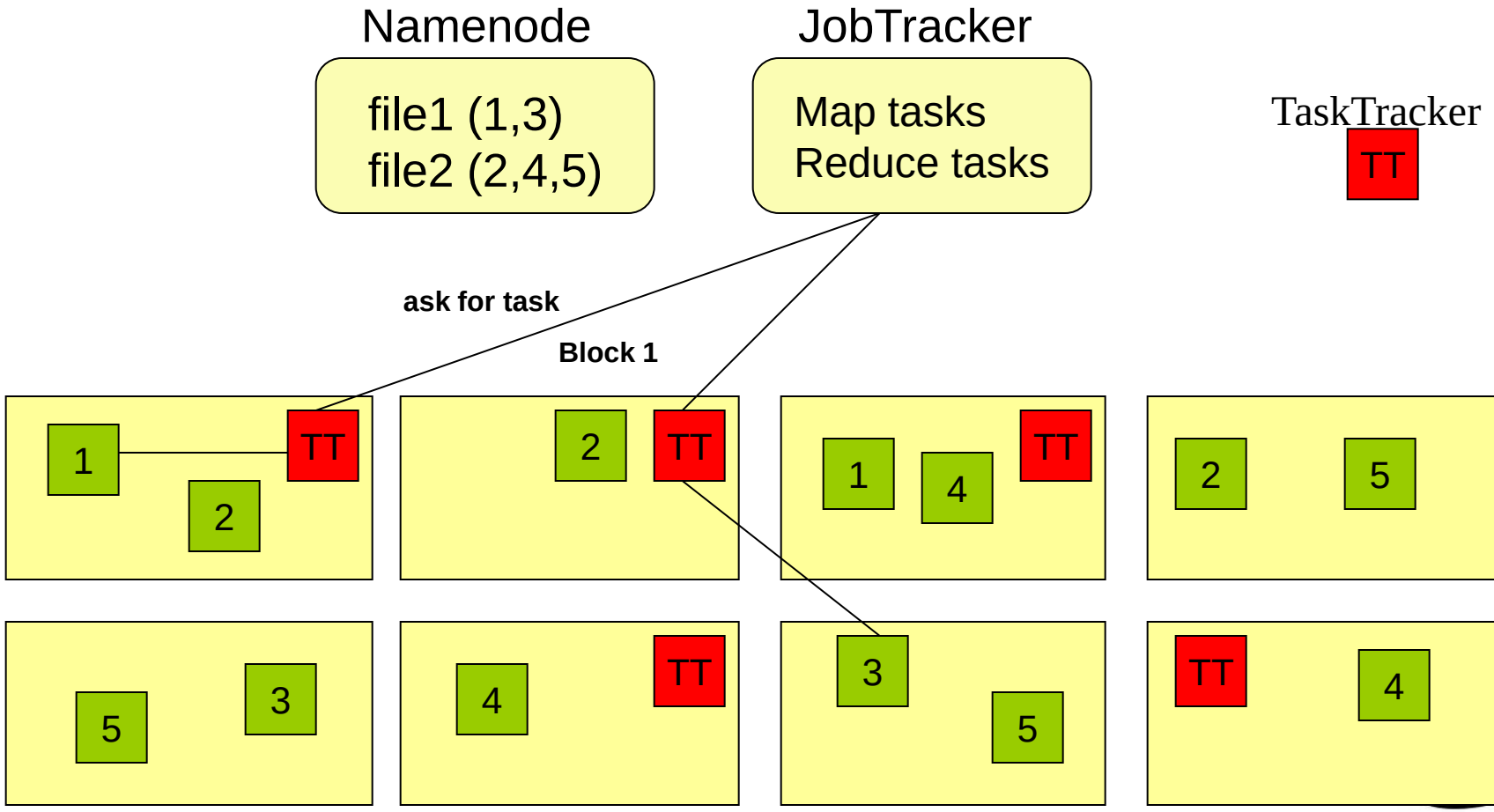
檔案路徑－ 副本數， 由哪幾個block組成

name:/users/joeYahoo/myFile - copies:2, blocks:{1,3}
name:/users/bobYahoo/someData.gzip, copies:3, blocks:{2,4,5}

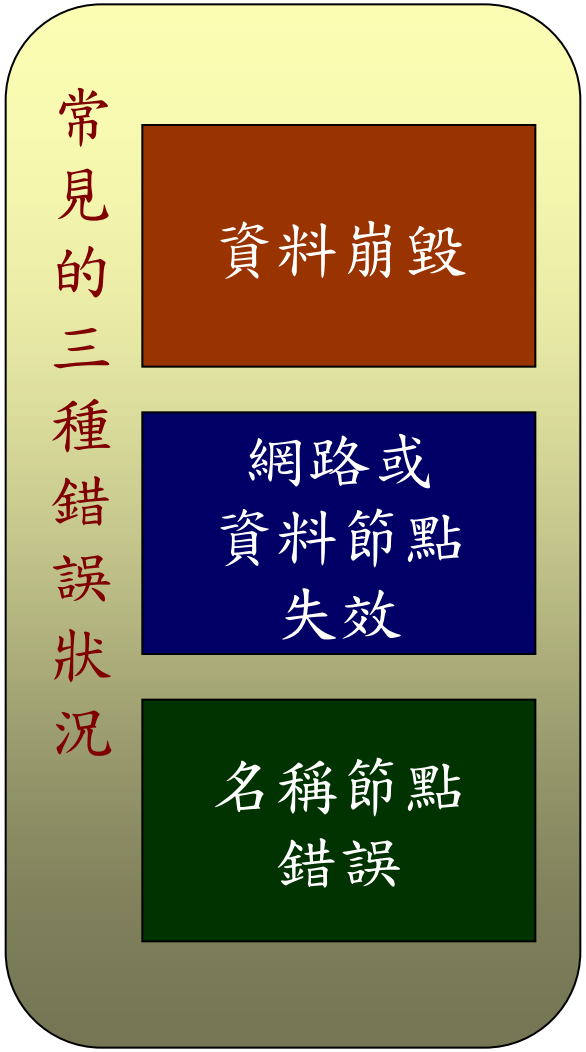


HDFS 運作

- 目的：提高系統的可靠性與讀取的效率
  - 可靠性：節點失效時讀取副本已維持正常運作
  - 讀取效率：分散讀取流量（但增加寫入時效能瓶頸）



可靠性機制



- 資料完整性
  - checked with CRC32
  - 用副本取代出錯資料
- Heartbeat
  - Datanode 定期向Namenode送heartbeat
- Metadata
  - FSImage、Editlog為核心印象檔及日誌檔
  - 多份儲存，當NameNode壞掉可以手動復原

一致性與效能機制

- 檔案一致性機制
  - 刪除檔案╲新增寫入檔案╲讀取檔案皆由Namenode負責
- 巨量空間及效能機制
  - 以Block為單位： 64M為單位
  - 在HDFS上得檔案有可能大過一顆磁碟
  - 大區塊可提高存取效率
  - 區塊均勻散佈各節點以分散讀取流量

HDFS的功能

- 類POXIS指令
- 權限控管
- 超級用戶模式
- Web 瀏覽
- 用戶配額管理
- 分散式複製檔案

POSIX Like

```
hadoop fs [-fs <local | file system URI>] [-conf <configuration file>]
        [-D <property=value>] [-ls <path>] [-lsr <path>] [-du <path>]
        [-dus <path>] [-mv <src> <dst>] [-cp <src> <dst>] [-rm <src>]
        [-rmr <src>] [-put <localsrc> <dst>] [-copyFromLocal <localsrc> <dst>]
        [-moveFromLocal <localsrc> <dst>] [-get <src> <localdst>]
        [-getmerge <src> <localdst> [addnl]] [-cat <src>]
        [-copyToLocal <src><localdst>] [-moveToLocal <src> <localdst>]
        [-mkdir <path>] [-report] [-setrep [-R] [-w] <rep> <path/file>]
        [-touchz <path>] [-test -[ezd] <path>] [-stat [format] <path>]
        [-tail [-f] <path>] [-text <path>]
        [-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
        [-chown [-R] [OWNER][:[GROUP]] PATH...]
        [-chgrp [-R] GROUP PATH...]
        [-help [cmd]]
```

結論

- HDFS使得普通硬體卻可支援處理大量的工作運算
  - 聯集叢集內的磁碟空間，且易於擴充
  - 容錯設計與自動處理
  - 讀取效率遠高於寫入效率

- 問題：
  - HDFS是否適合用來做
    - 大規模檔案備份系統？
    - 即時存取的線上文件系統？
  - hdfs 的空間是否與linux 的互相空間獨立？
  - 執行hdfs 的namenode format 時是否會刪除linux的檔案？