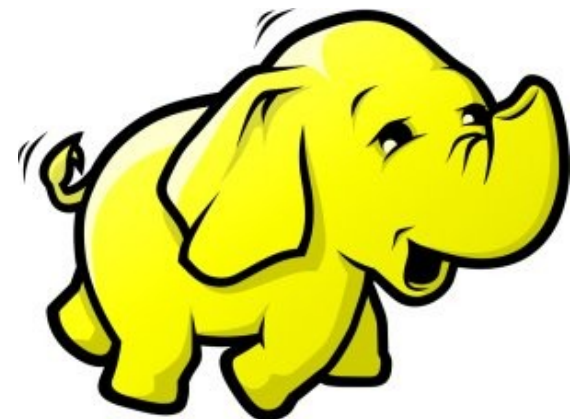




# HDFS 簡介

Introduction to Hadoop Distributed File System

**Jazz Wang**  
**Yao-Tsung Wang**  
**jazz@nchc.org.tw**



# What is HDFS ??

## 什麼是 **HDFS** ??

- **Hadoop Distributed File System**

- 實現類似 Google File System 分散式檔案系統
- Reference from Google File System.
- 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
- **A scalable distributed file system for large data analysis .**
- 運作於廉價的普通硬體上，又可以提供容錯功能
- **based on commodity hardware with high fault-tolerant.**
- 給大量的用戶提供總體性能較高的服務
- **It have better overall performance to serve large amount of users.**

# Features of HDFS ...

## **HDFS** 的特色是 ...

- **硬體錯誤容忍能力 Fault Tolerance**
  - 硬體錯誤是正常而非異常
  - Failure is the norm rather than exception
  - 自動恢復或故障排除
  - automatic recovery or report failure
- **串流式的資料存取 Streaming data access**
  - 批次處理多於用戶交互處理
  - Batch processing rather than interactive user access.
  - 高 Throughput 而非低 Latency
  - High aggregate data bandwidth (throughput)

# Features of HDFS ...

## HDFS 的特色是 ...

- **大規模資料集 Large data sets and files**
  - 支援 Petabytes 等級的磁碟空間
  - Support Petabytes size
- **一致性模型 Coherency Model**
  - 一次寫入，多次存取 Write-once-read-many
  - 簡化一致性處理問題 This assumption simplifies coherency
- **在地運算 Data Locality**
  - 到資料的節點上計算 > 將資料從遠端複製過來計算
  - “move compute to data” > “move data to compute”
- **異質平台移植性 Heterogeneous**
  - 即使硬體不同也可移植、擴充
  - HDFS could be deployed on different hardware

# Parallel Computing using NFS storage

使用 **NFS** 進行平行運算

NFS Client RAM

NFS Client Bridge

NFS Client NIC

NFS Server NIC

NFS Server Bridge

NFS Server Disk

Bus I/O (2)

NFS Client CPU

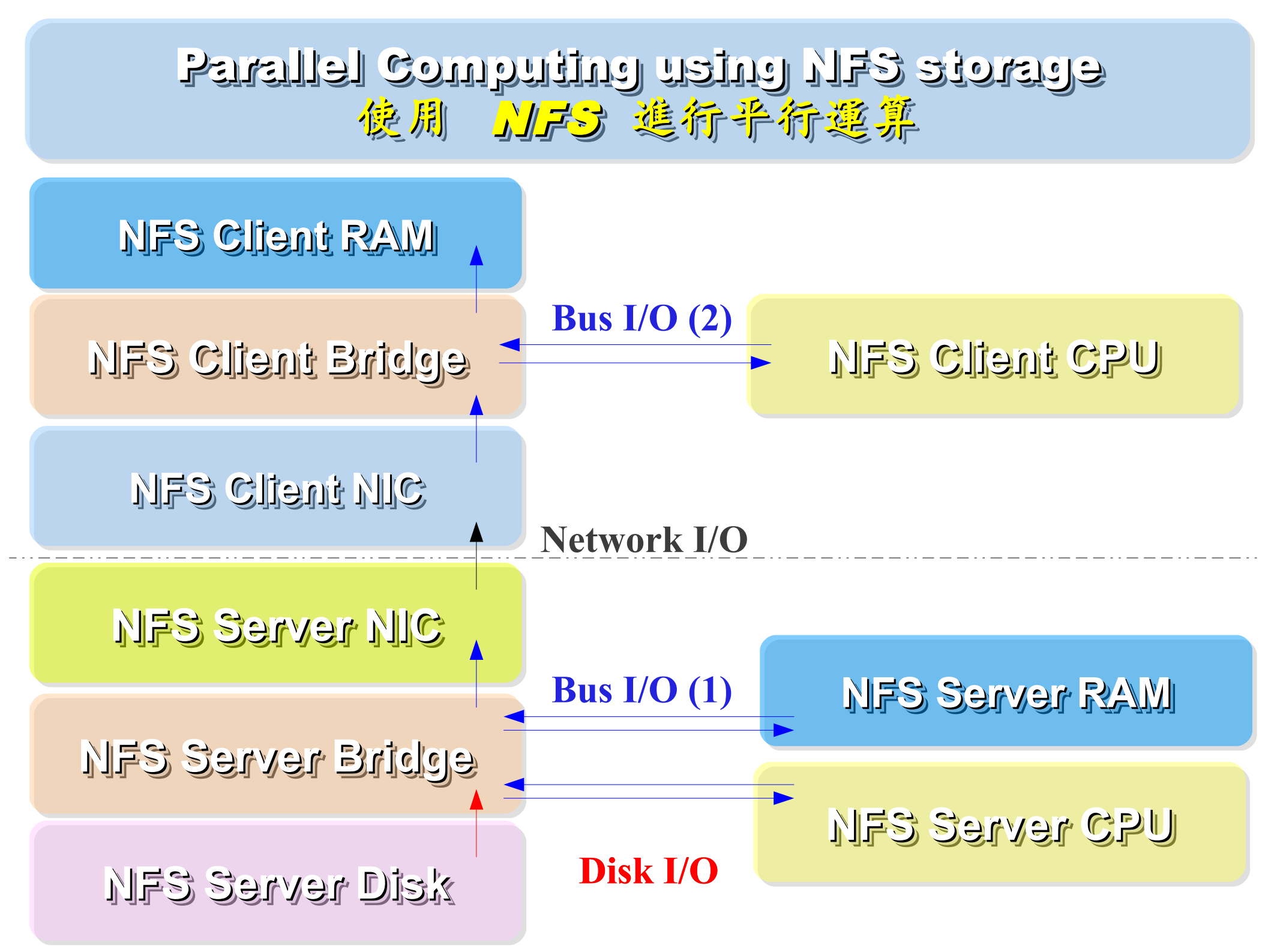
Network I/O

Bus I/O (1)

NFS Server RAM

NFS Server CPU

Disk I/O



# Parallel Computing using HDFS

使用 **HDFS** 進行平行運算

TaskTracker RAM

TaskTracker Bridge

Disk I/O x N Node

DataNode Local Disk

Bus I/O (2)

TaskTracker CPU

Network I/O

TaskTracker NIC

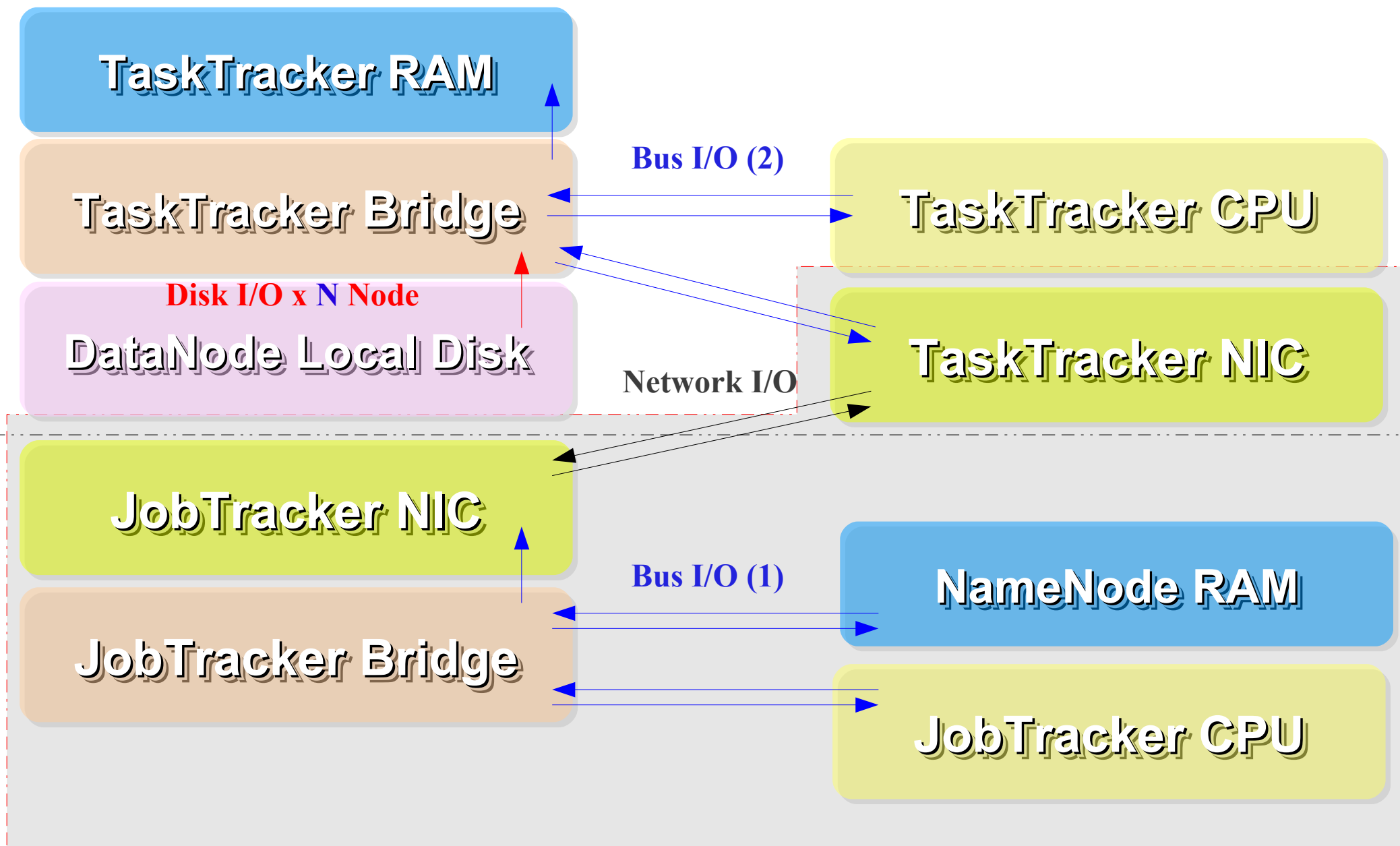
JobTracker NIC

Bus I/O (1)

NameNode RAM

JobTracker Bridge

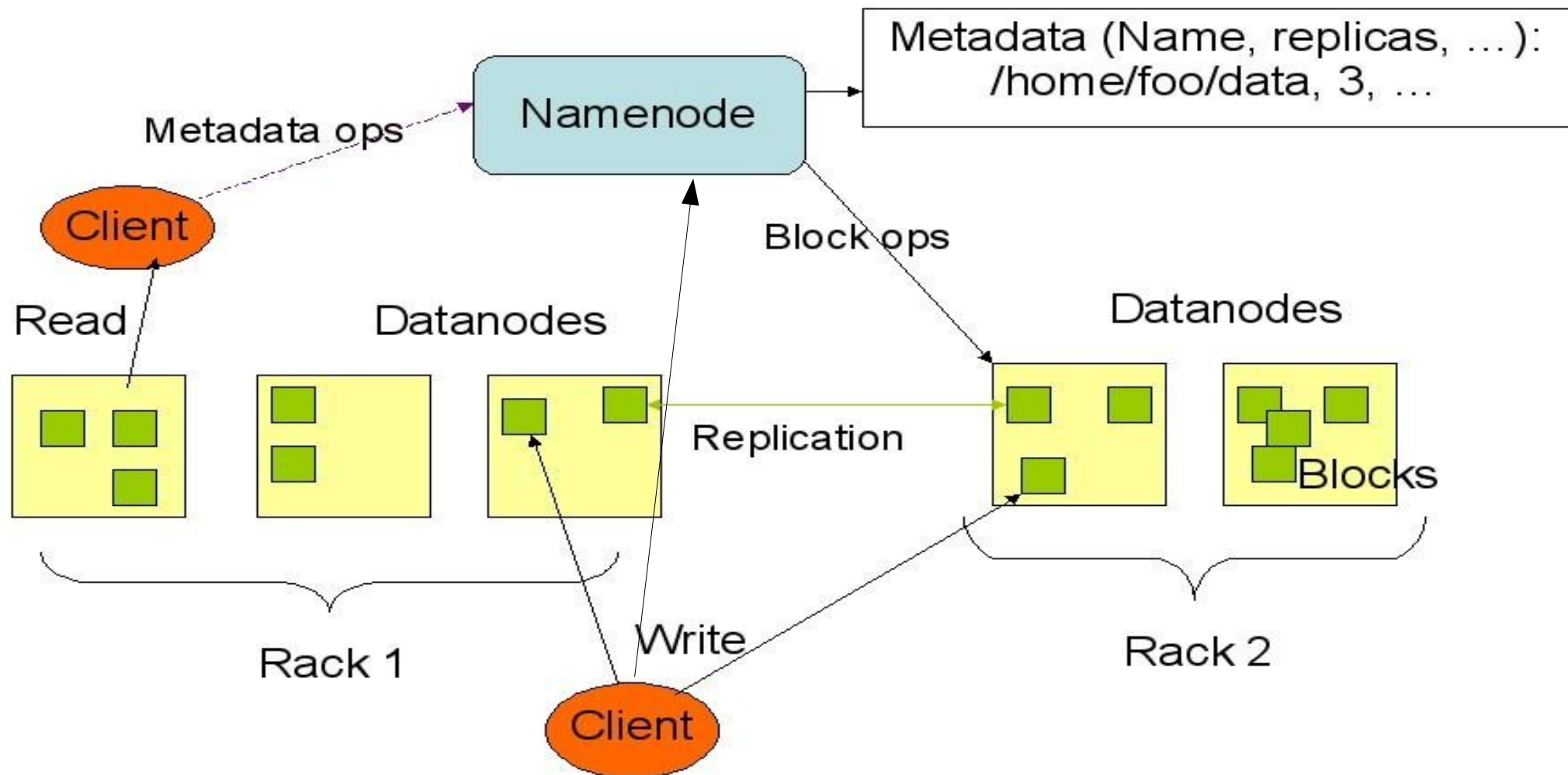
JobTracker CPU



# How HDFS manage data ...

## *HDFS* 如何管理資料 ...

### HDFS Architecture



# How does HDFS work ...

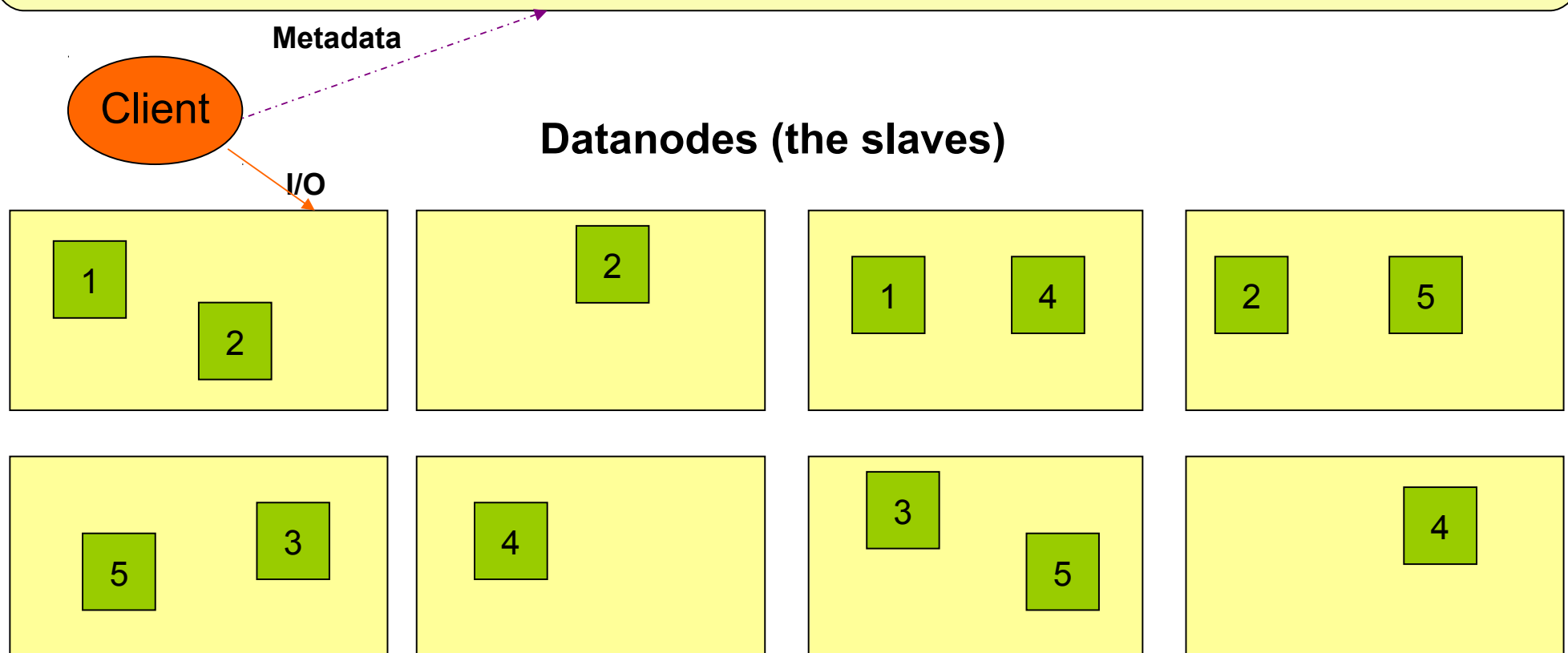
## **HDFS** 如何運作 ...

Namenode (the master)

Path and Filename – **Replication** , **blocks**

name:/users/joeYahoo/myFile - copies:2, blocks:{1,3}

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

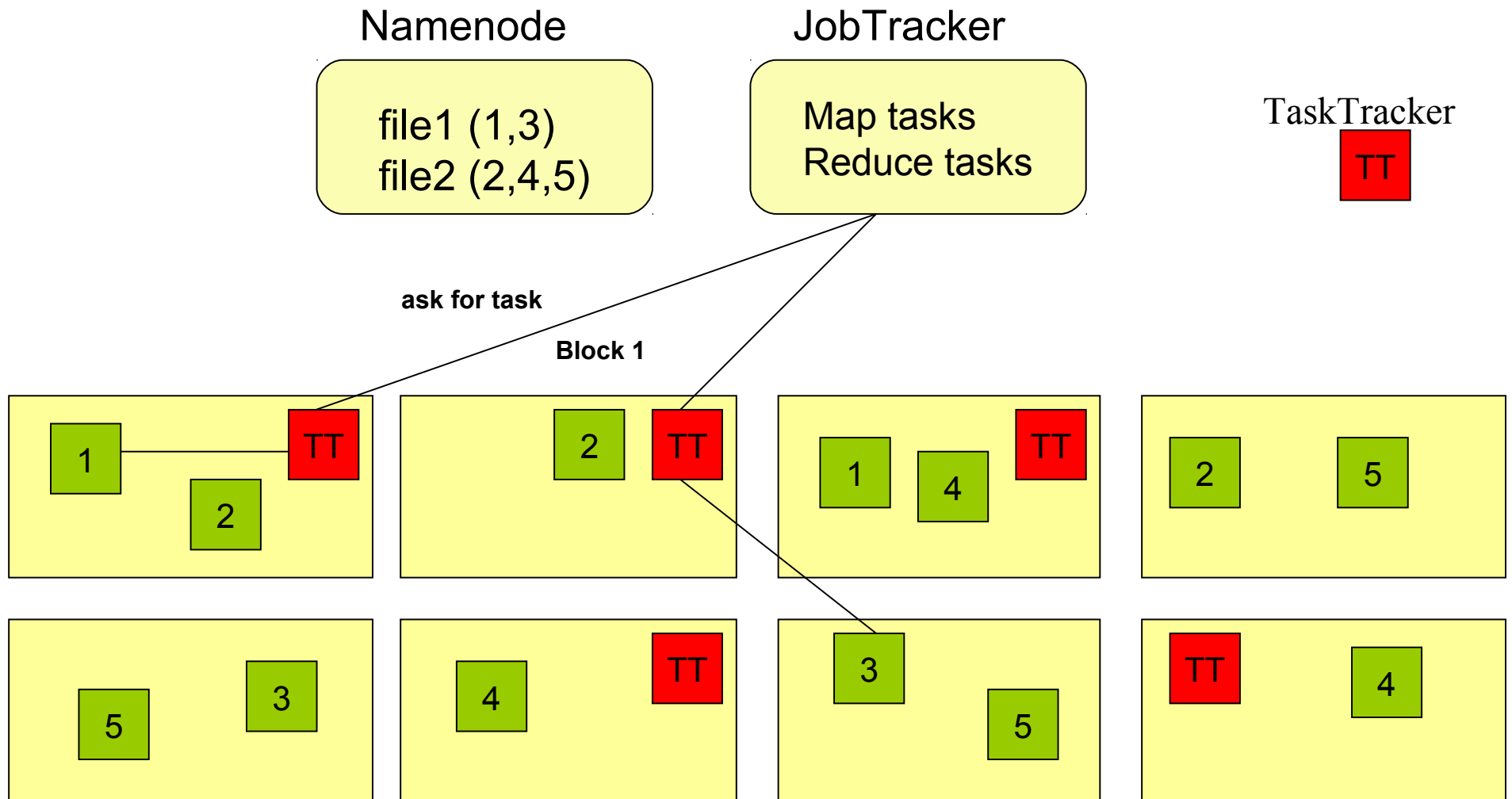




# About Data locality ...

## **HDFS** 如何達成在地運算 ...

- Increase reliability and read bandwidth
  - robustness : read replication while found any failure
  - High read bandwidth : distribute read ( but increase write bottleneck )



# About Fault Tolerance ...

## **HDFS** 如何達成容錯機制 ...

資料崩毀  
Data Corrupt

網路或資料  
節點失效  
Network Fault  
DataNode Fault

名稱節點錯誤  
NameNode Fault

- 資料完整性 Data integrity
  - checked with CRC32
  - 用副本取代出錯資料
  - Replcae corrupt block with replication one
- Heartbeat
  - Datanode send **heartbeat** to Namenode
- Metadata
  - FSImage 、 Editlog 為核心印象檔及日誌檔
  - FSImage – core file system mapping image
  - Editlog – like. SQL transaction log
  - 多份儲存，當名稱節點故障時可以手動復原
  - Multiple backups of FSImage and Editlog
  - Manually recovery while NameNode Fault

# Coherency Model and Performance of HDFS

## **HDFS 的一致性機制與效能 ...**

- **檔案一致性機制 Coherency model of files**
  - 刪除檔案\新增寫入檔案\讀取檔案皆由名稱節點負責
  - NameNode handle the operation of write, read and delete.
- **巨量空間及效能機制 Large Data Set and Performance**
  - 預設每個區塊大小以 64MB 為單位
  - By default, the block size is 64MB
  - 大區塊可提高存取效率
  - Bigger block size will enhance read performance
  - 檔案有可能大過一顆磁碟
  - Single file stored on HDFS might be larger than single physical disk of DataNode.
  - 區塊均勻散佈各節點以分散讀取流量
  - Fully distributed blocks increase throughput of reading.

# POSIX like HDFS commands

## 與 **POSIX** 相似的操作指令 ...

```
jazz@hadoop:~$ hadoop fs
Usage: java FsShell
    [-ls <path>]
    [-lsr <path>]
    [-du <path>]
    [-dus <path>]
    [-count[-q] <path>]
    [-mv <src> <dst>]
    [-cp <src> <dst>]
    [-rm <path>]
    [-rmr <path>]
    [-expunge]
    [-put <localsrc> ... <dst>]
    [-copyFromLocal <localsrc> ... <dst>]
    [-moveFromLocal <localsrc> ... <dst>]
    [-get [-ignoreCrc] [-crc] <src> <localdst>]
    [-getmerge <src> <localdst> [addnl]]
    [-cat <src>]
    [-text <src>]
    [-copyToLocal [-ignoreCrc] [-crc] <src> <localdst>]
    [-moveToLocal [-crc] <src> <localdst>]
    [-mkdir <path>]
    [-setrep [-R] [-w] <rep> <path/file>]
    [-touchz <path>]
    [-test -[ezd] <path>]
    [-stat [format] <path>]
    [-tail [-f] <file>]
    [-chmod [-R] <MODE[,MODE]... | OCTALMODE> PATH...]
    [-chown [-R] [OWNER][:[GROUP]] PATH...]
    [-chgrp [-R] GROUP PATH...]
    [-help [cmd]]
```