叢集式系統應用

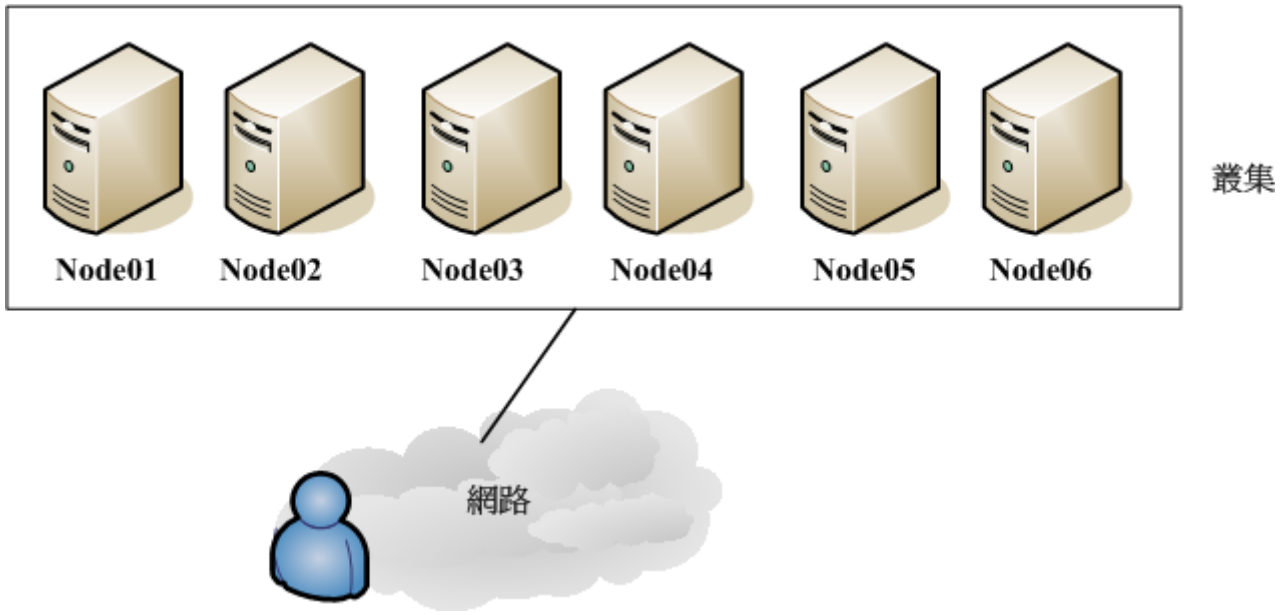
大綱

- 何謂叢集式系統
- 叢集式系統的通訊
- 叢集式系統的檔案系統
- 叢集式系統的同步機制
- 平行程式計算
- 叢集式系統的資源管理
- 本章重點回顧

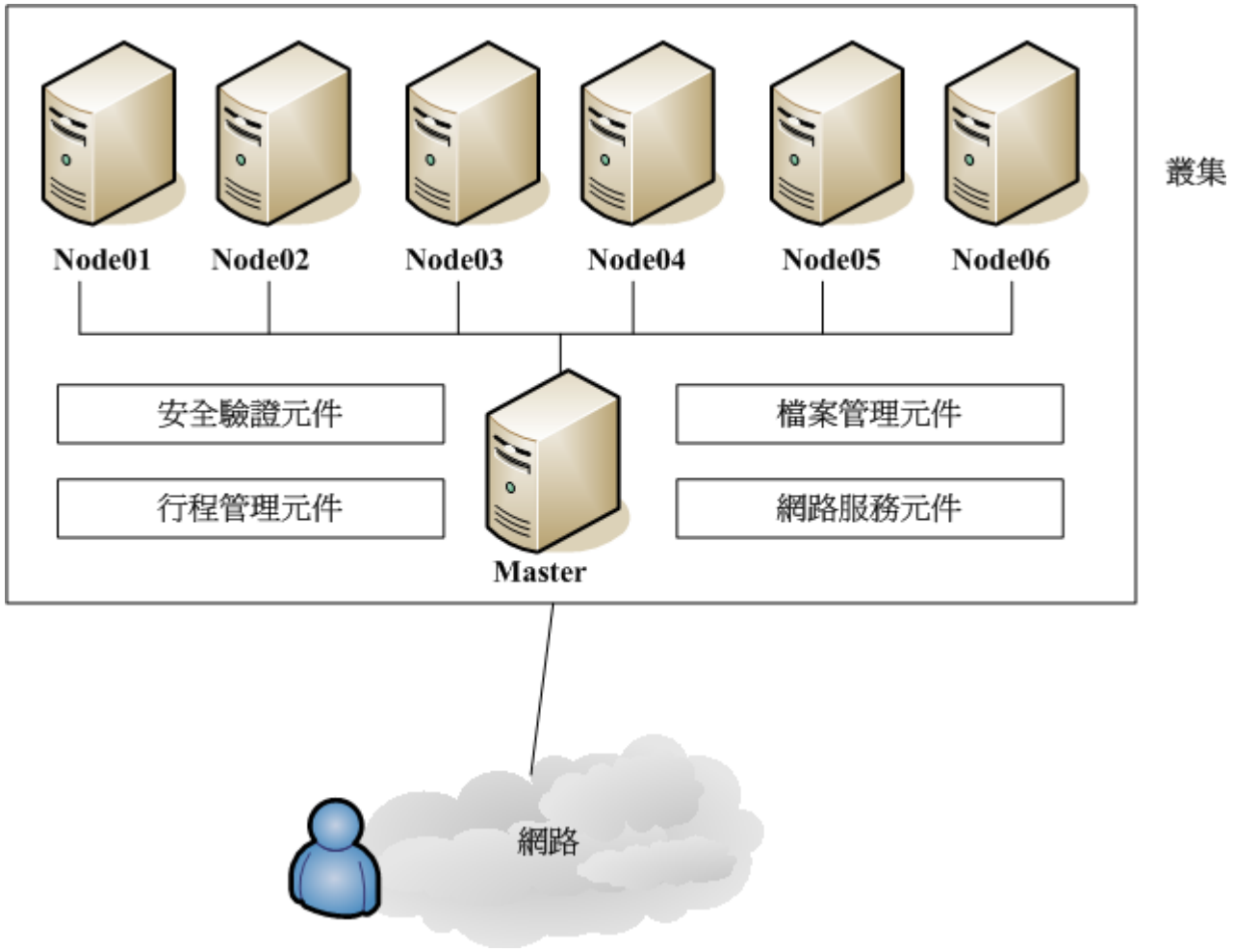
何謂叢集式系統

- 集中式多行程（Centralized Multiprocessors）架構與分散式系統（Distributed Systems）
- 分散式系統依架構可區分為：
  - 叢集方式（Cluster）
  - 全然分散式（Fully Distributive）

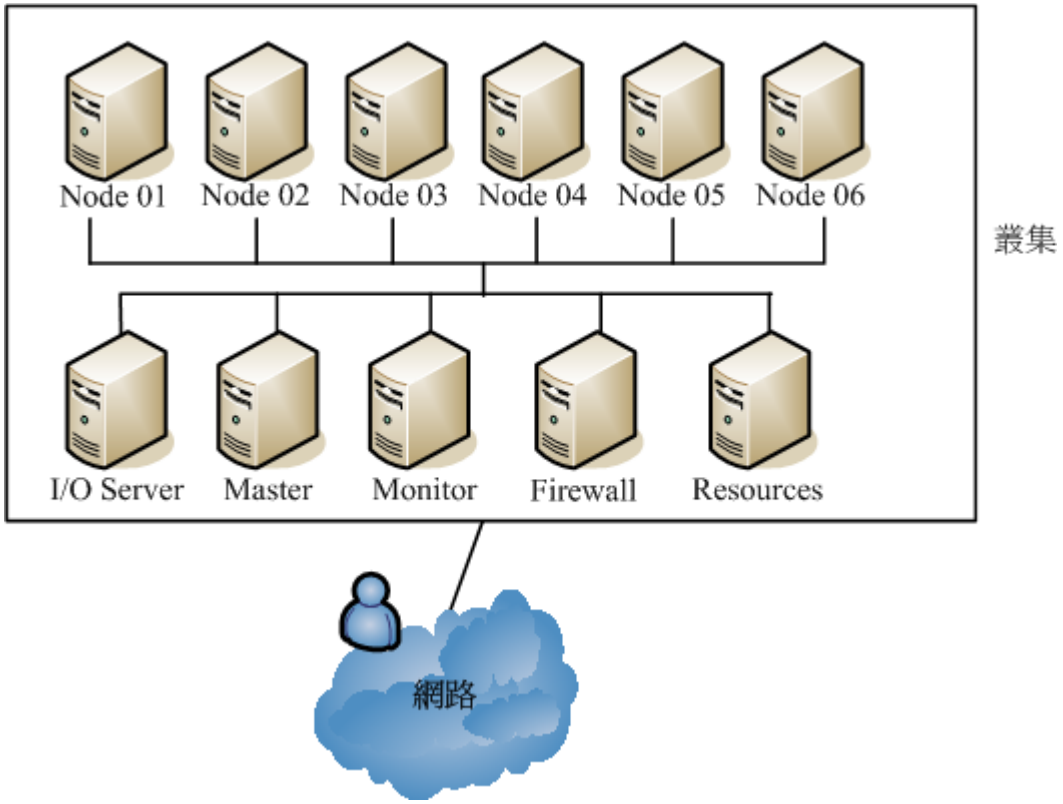
- 最簡單的叢集架構—對稱叢集架構（Symmetric Cluster）



非對稱叢集架構


Node01
Node02
Node03
Node04
Node05
Node06
叢集
安全驗證元件
檔案管理元件
行程管理元件
網路服務元件
Master
網路

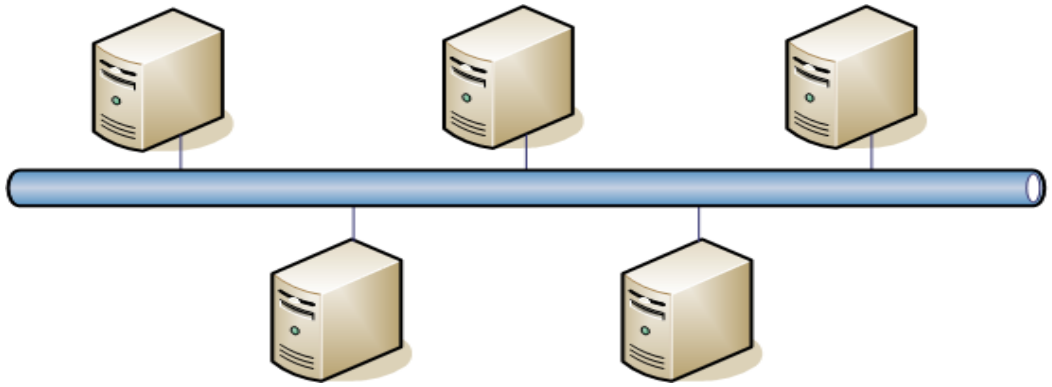
延伸叢集架構


Node 01
Node 02
Node 03
Node 04
Node 05
Node 06
叢集
I/O Server
Master
Monitor
Firewall
Resources
網路

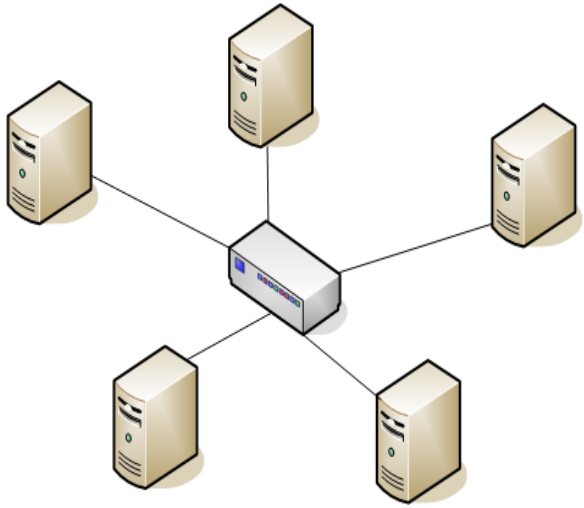
叢集式系統的通訊

- 網路拓樸型態：
  - 匯流排拓樸（Bus Topology）
  - 星狀拓樸（Star Topology）
  - 環狀拓樸（Ring Topology）
  - 網狀拓樸（Mesh Topology）
- 選擇拓撲的依據：
  - 串接時所支付的成本
  - 通訊時的可靠度
  - 通訊時的速度

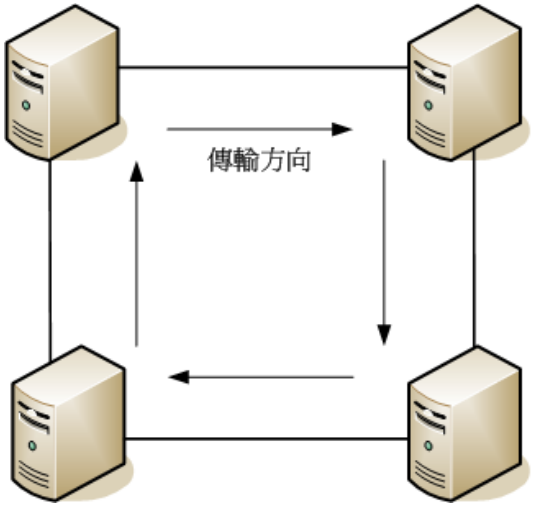
- 匯流排拓樸（Bus Topology）



- 星狀拓樸（Star Topology）

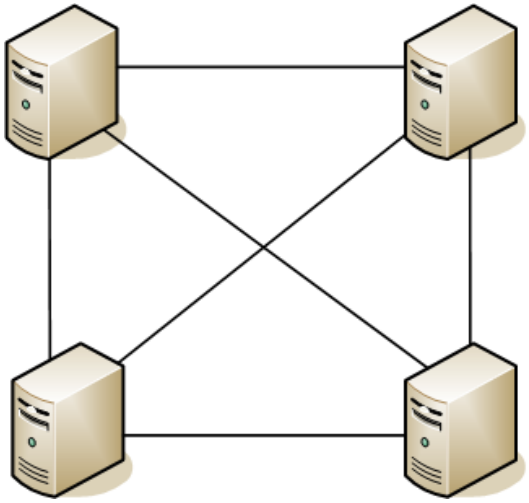


- 環狀拓樸（Ring Topology）



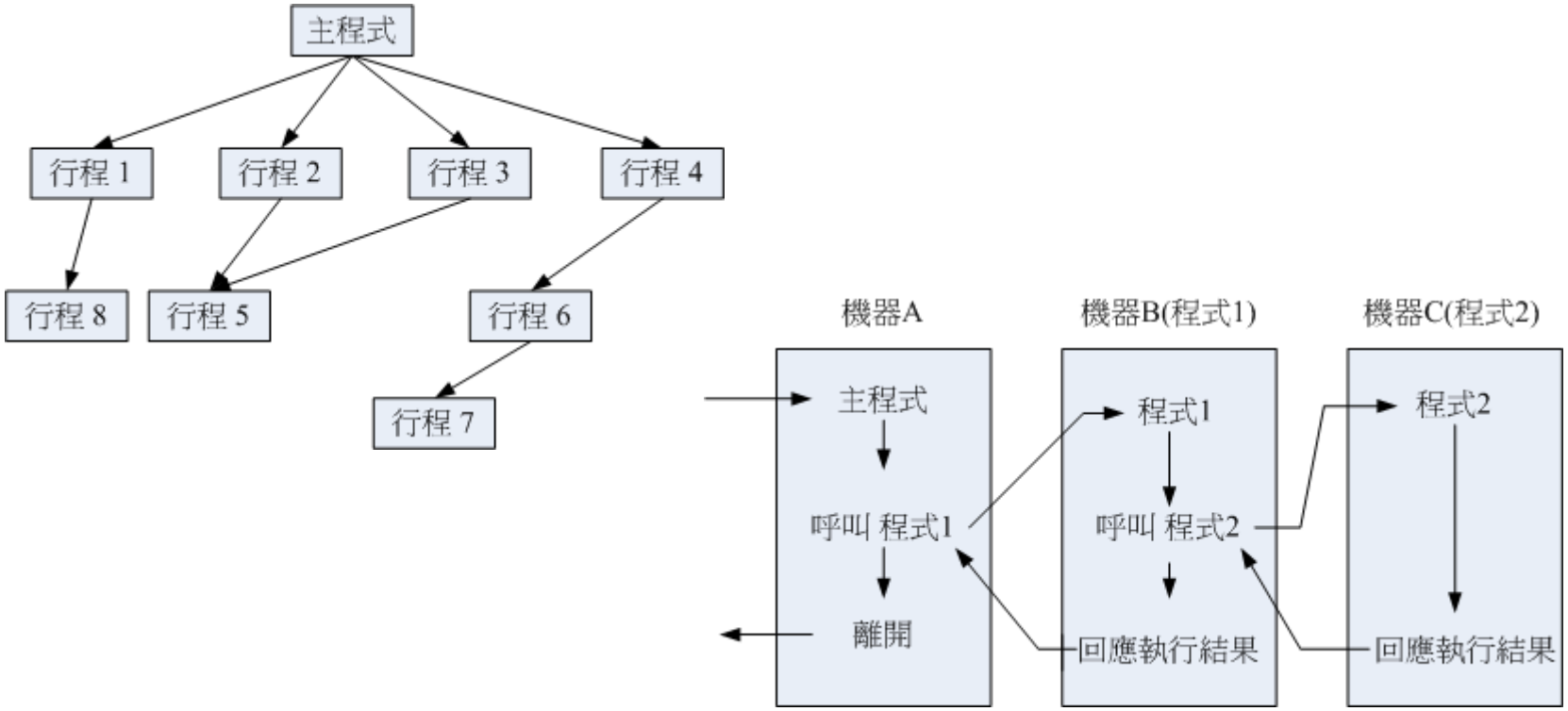
叢集

- 網狀拓樸（Mesh Topology）



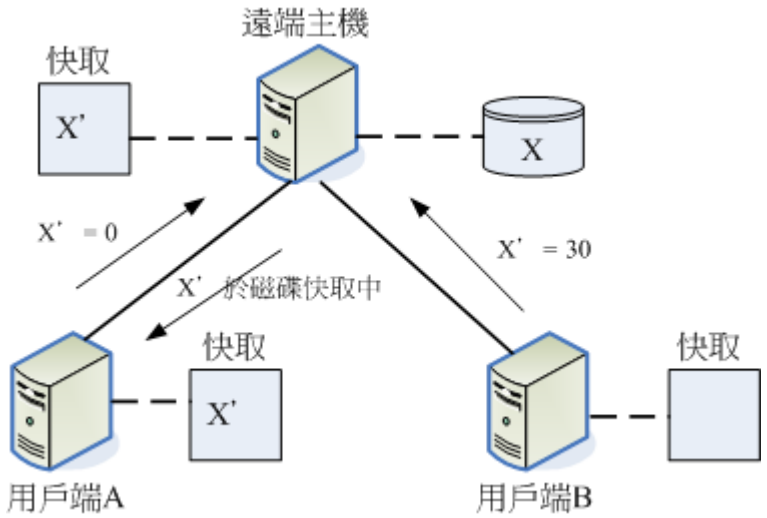
叢集

- 主從式架構（Client/Server）與對等式網路架構（Peer-to-Peer）
- 遠端程序呼叫（Remote Procedure Call，RPC）

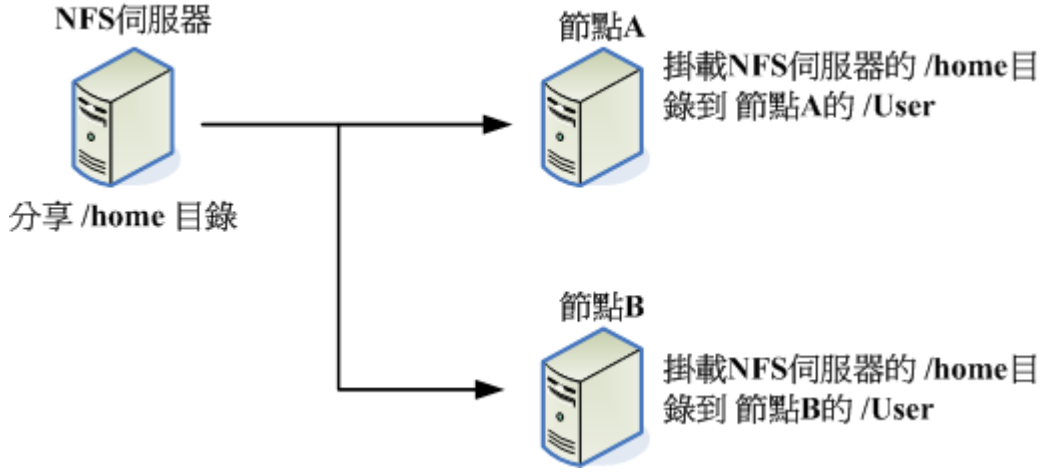


叢集式系統的檔案系統

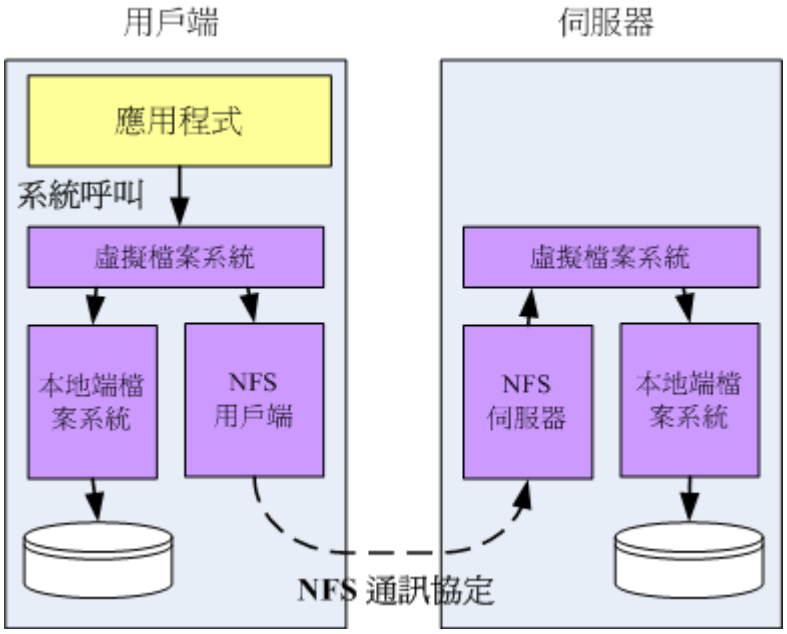
- 叢集式架構下，檔案的命名模式：
  - 主機名稱加上路徑，例如，hcserver:/home
  - 將遠端檔案系統掛載到本地端的檔案階層中
  - 使用單一名稱，使所有電腦所看到的名稱皆相同
- 快取一致性問題（Cache-consistency Problem）
- 直接寫入演算法（Write-through Algorithm）



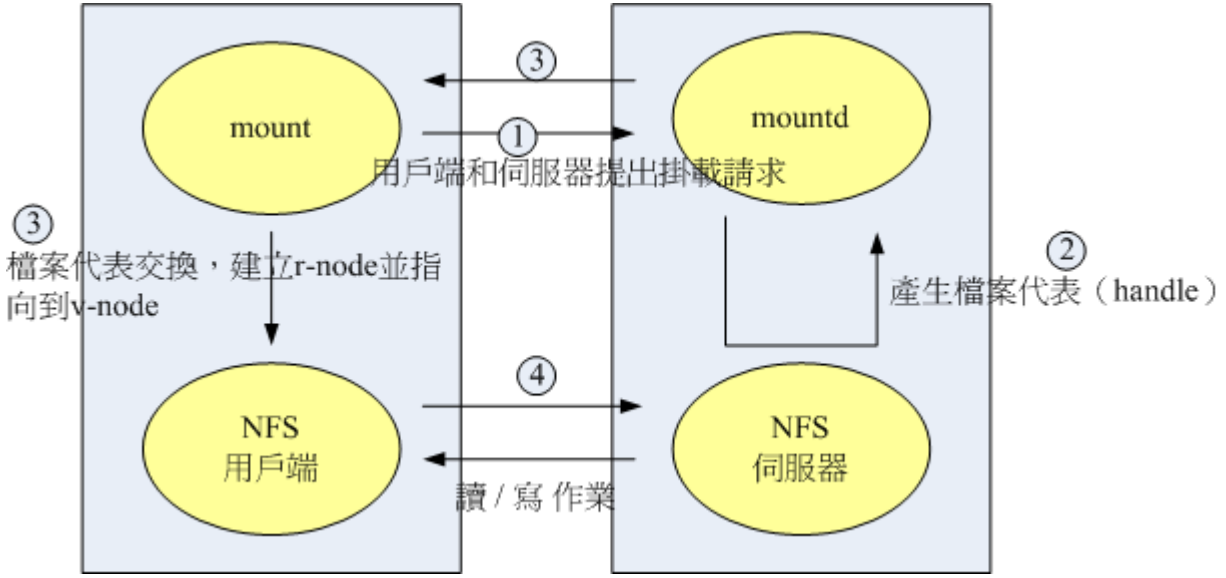
- 網路檔案系統（Network File System，NFS）



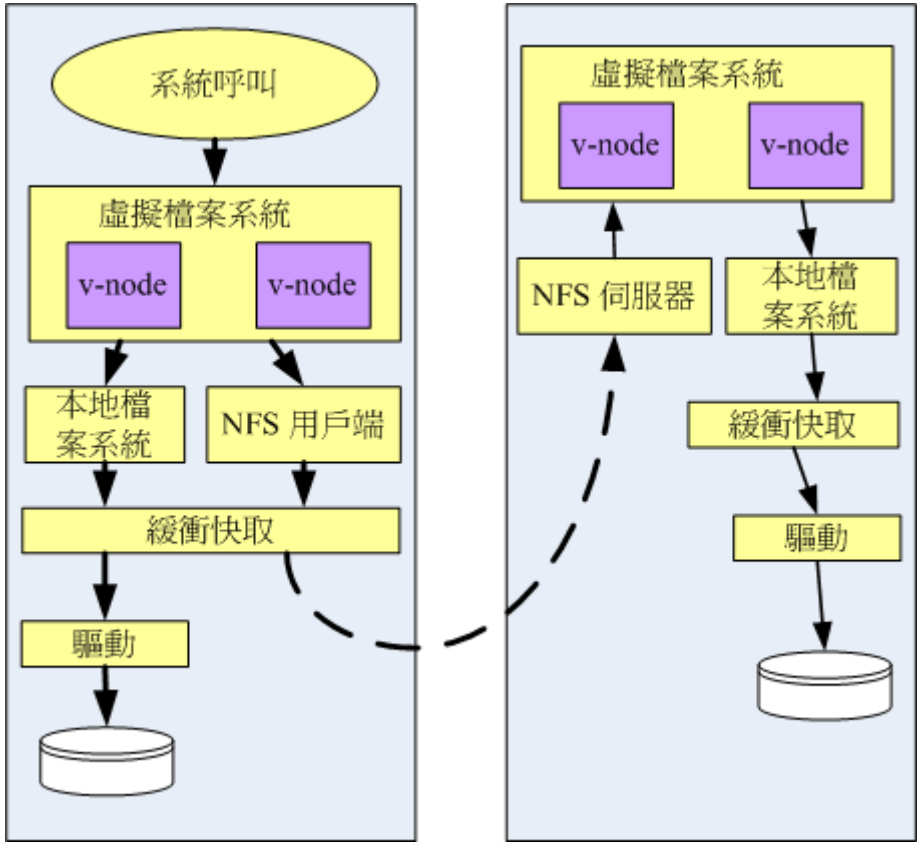
NFS運作層次


用戶端
伺服器
應用程式
系統呼叫
虛擬檔案系統
本地端檔案系統
NFS用戶端
虛擬檔案系統
NFS伺服器
本地端檔案系統
NFS 通訊協定

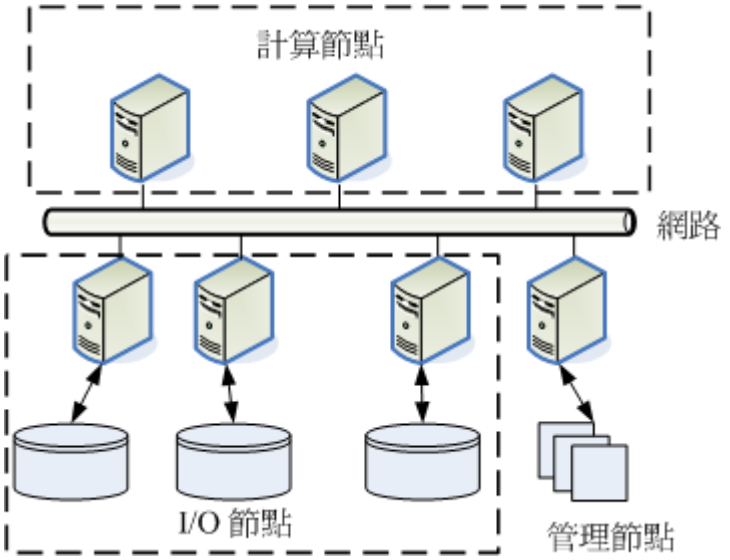
NFS遠端掛載運作方式



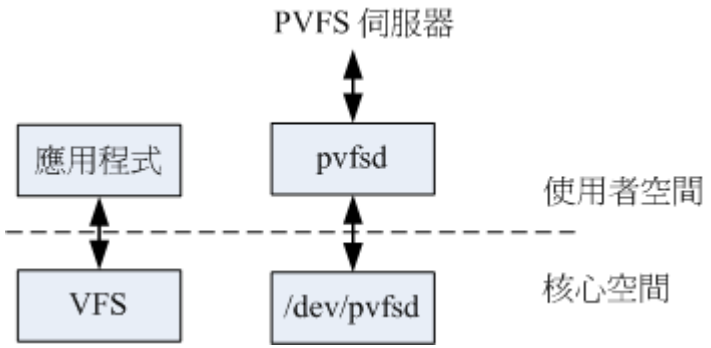
NFS檔案存取

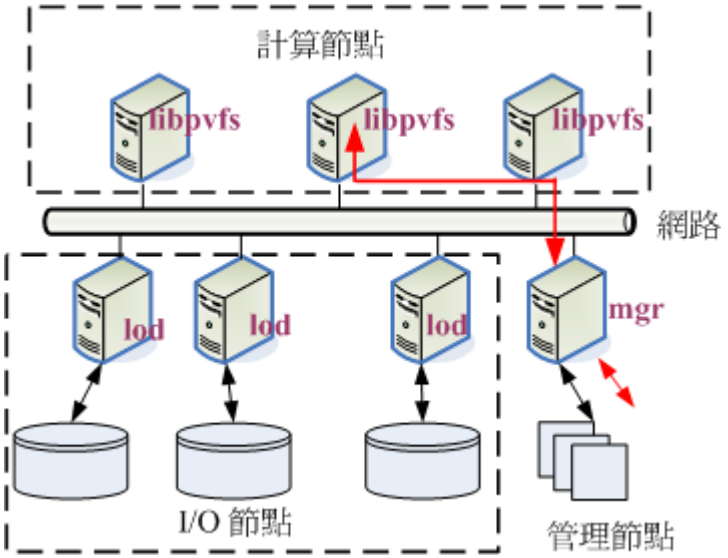

系統呼叫
虛擬檔案系統
v-node
v-node
本地檔案系統
NFS 用戶端
緩衝快取
驅動
虛擬檔案系統
v-node
v-node
NFS 伺服器
本地檔案系統
緩衝快取
驅動

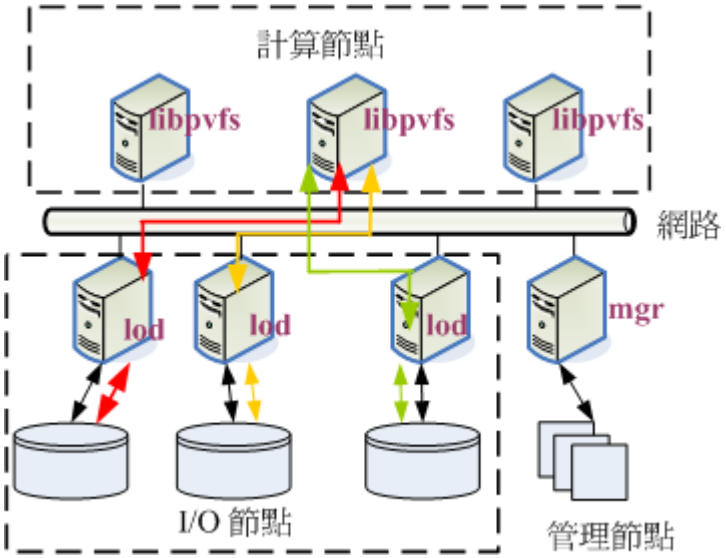
- 平行虛擬檔案系統（Parallel Virtual File System，PVFS）
- PVFS軟體中主要功能：
  - 一致性的檔案命名方式。
  - 支援現有的檔案系統存取模式。
  - 資料分散於叢集架構上的節點硬碟內。
  - 為應用程式提供更高性能的資料存取方式。



- PVFS系統主要組成的模組：
  - 元資料伺服器（Metadata Server）
  - I/O伺服器（I/O Server，iod）
  - PVFS本地API（PVFS native API，Llibpvfs）
  - PVFS Linux核心支援（PVFS Linux Kernel Support）







- PVFS提供了三種模式存取存放於PVFS上的檔案
  - 透過PVFS提供的API介面
  - Linux核心介面
  - ROMIO、MPI-IO介面
- 通用平行檔案系統（General Parallel File System，GPFS）

叢集式系統的同步機制

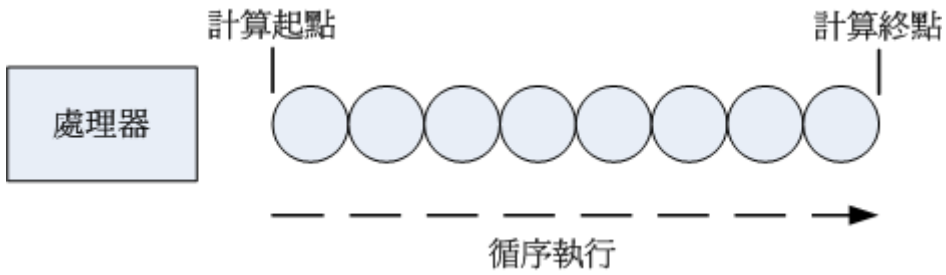
- 分散式演算的特性：
  - 相關資訊散佈在多部電腦之間
  - 行程只能依靠本地資訊來作決定
  - 系統必須避免一點錯誤而故障
  - 沒有共同的時鐘來源存在
- 在Linux作業系統中對於檔案的時間戳記表示方式大致上有三種類別，分別為：Modify time（mtime）、Access time（atime）、Inode Change time（ctime）

- 邏輯時鐘（Logical Clocks）與發生在前（Happened-before）判斷式
- 並行控制演算法（Concurrency Control Algorithm）
  - 鎖定
  - 最佳化並行控制
  - 時間戳記

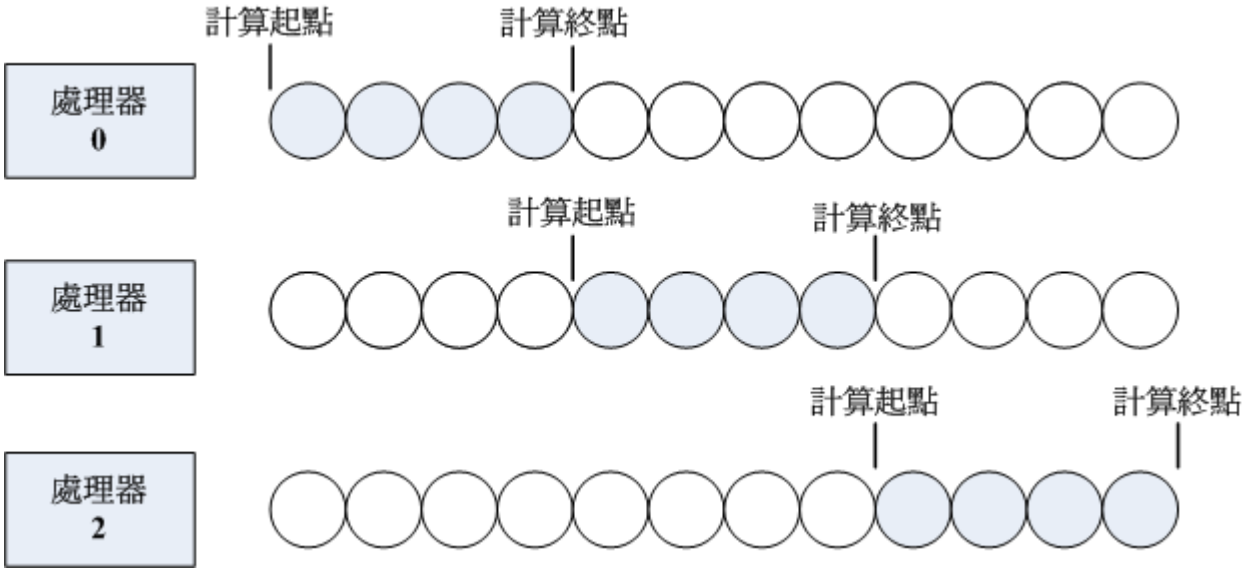
平行程式計算

- 計算節點對於被交付的行程在處理處理器之間的資料傳送方式通常有兩種：
  - 點對點通訊（Point to Point Communication）
  - 集體通訊（Collective Communication）
- MPI_Send與MPI_Recv

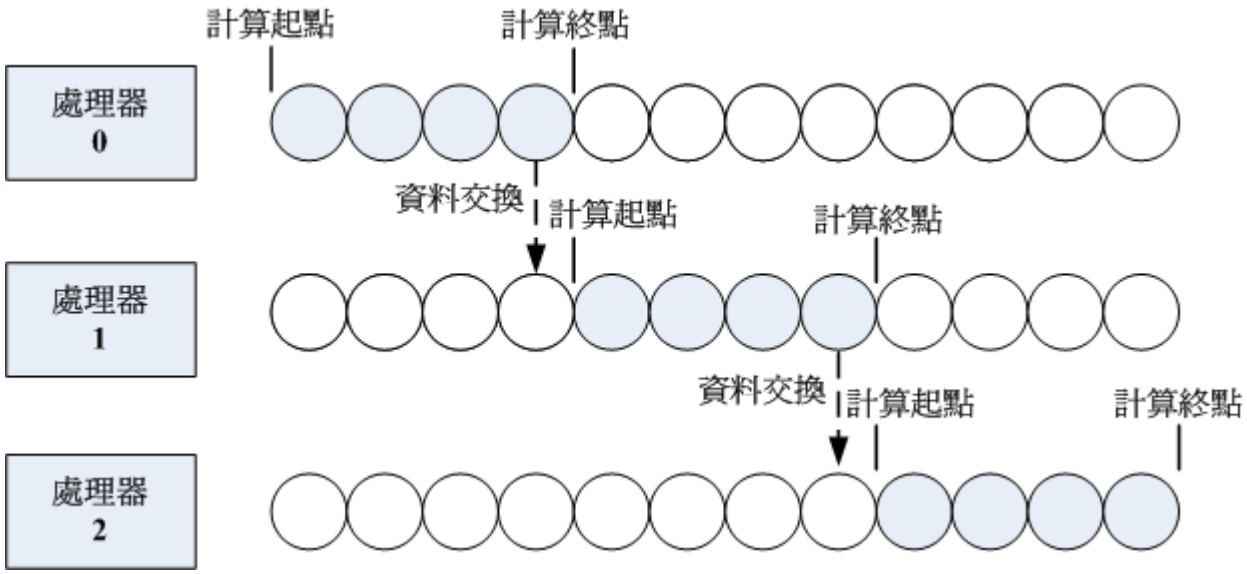
- 單處理器循序程式運作方式



- 計算切割而資料不切割的運作方式



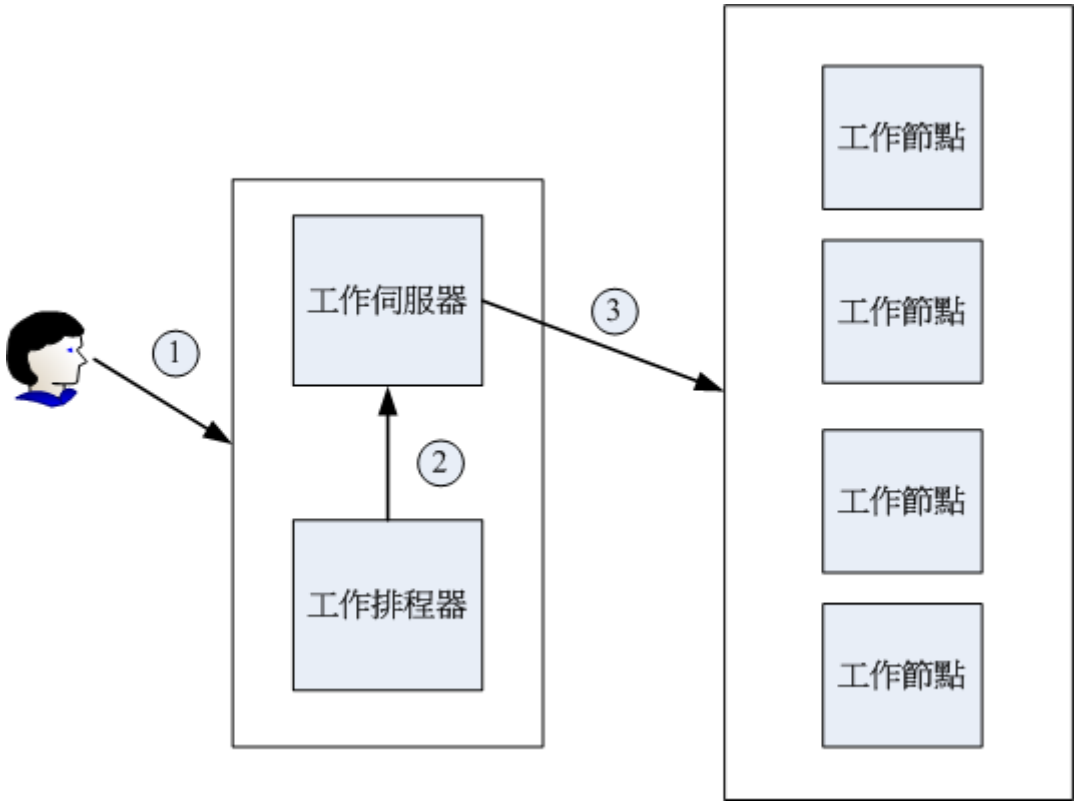
- 計算與資料均切割的運作方式



叢集式系統的資源管理

- 常見的佇列系統很多，例如：
  - NQS（Network Queue System）
  - DQS（Distributed Queue System）
  - Altair的PBS Pro（Portable Batch System）等
- 佇列系統本身需具備下列幾項特點
  - 可以公平分配計算節點上的運算資源，使這些資源可以被充分的運用。
  - 可以針對不同的使用者屬性進行分配可用資源。
  - 隨時監控計算節點的執行狀態與資源使用率。
  - 排程演算法的變更，與計算平衡負載等。

- 佇列系統運作的架構



本章重點回顧

- 了解叢集式系統的架構與特性。
- 了解叢集式系統的資料通訊方式。
- 了解叢集式系統的檔案系統架構與特性。
- 了解叢集式系統的同步機制運作方式。
- 了解如何透過佇列管理程式於叢集系統上進行資源管理。