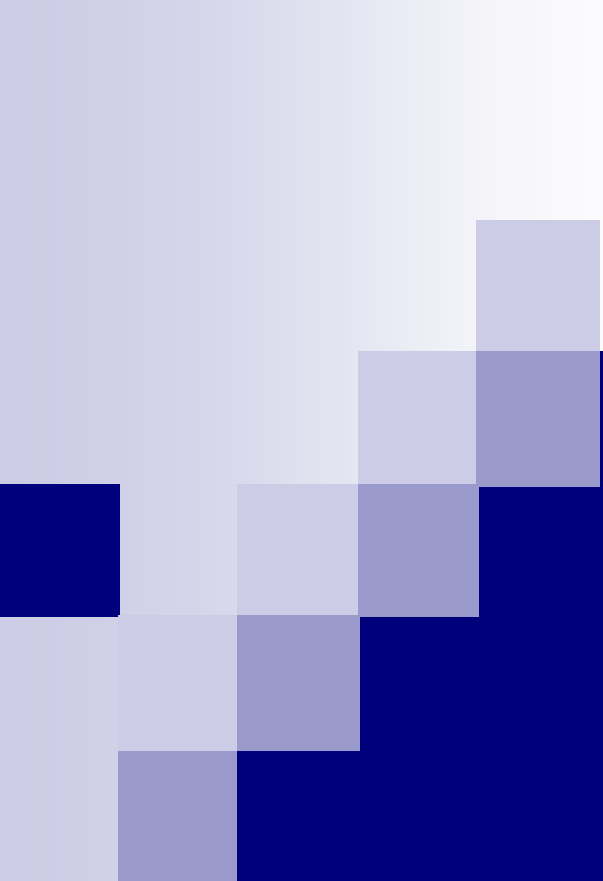




叢集式系統應用

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

前言

- 高效能運算的重要性：
 - 氣象模式模擬(氣象預報機制)
 - 水污染/空氣污染模式(水/空污預報機制)
 - 生化科技(模擬可能的後果)
 - 軍事/衛星/太空模擬
- 單一電腦硬體的高效能運算瓶頸
 - 硬體架構的限制
 - 擴充性與相容性及除錯方面的問題
- 解決方案：
 - 透過網路連線多部主機
 - 將工作分給多部主機同時完成一項複雜工作。

何謂叢集式系統

■ 高效能運算平台的主要分類：

- 集中式多行程（Centralized Multiprocessors）架構
- 分散式系統（Distributed Systems）

■ 分散式系統依架構可區分為：

□ 叢集方式（Cluster）

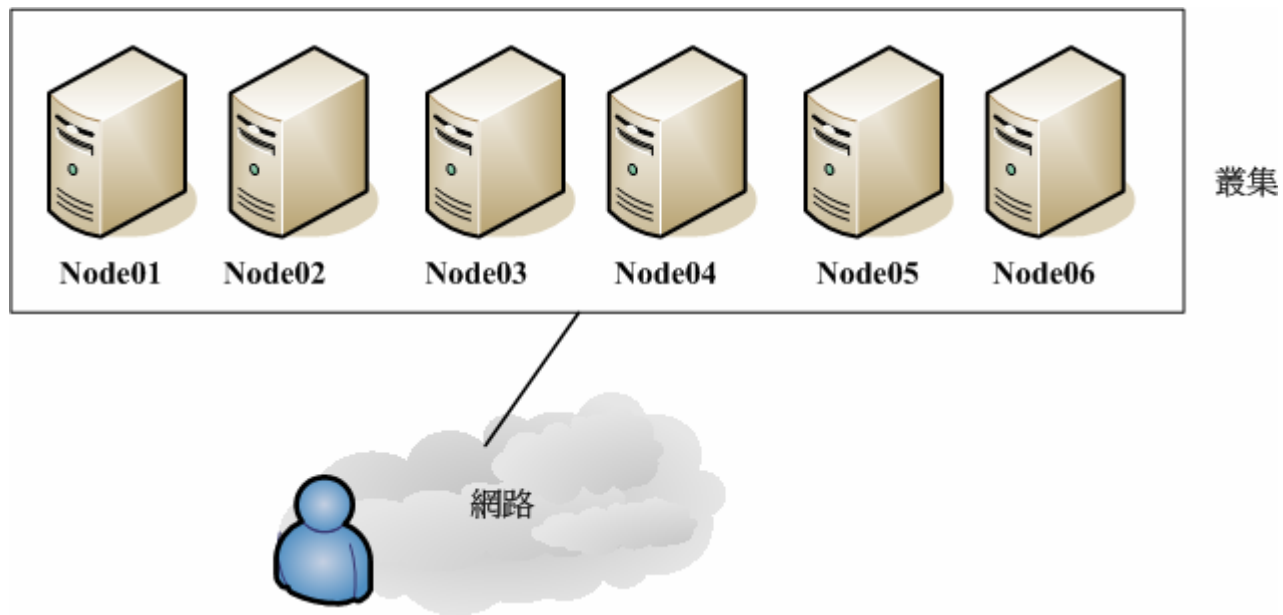
- 透過網路將許多不同的電腦集結起來，提供各自擁有的資源

□ 全然分散式（Fully Distributive）

- 運算資源都是各自獨立在每個node上，只是每個node會主動的取得工作，然後再將工作回饋回去。
- 尋找外星人計畫 <http://setitaiwan.tripod.com/MIRROR/>

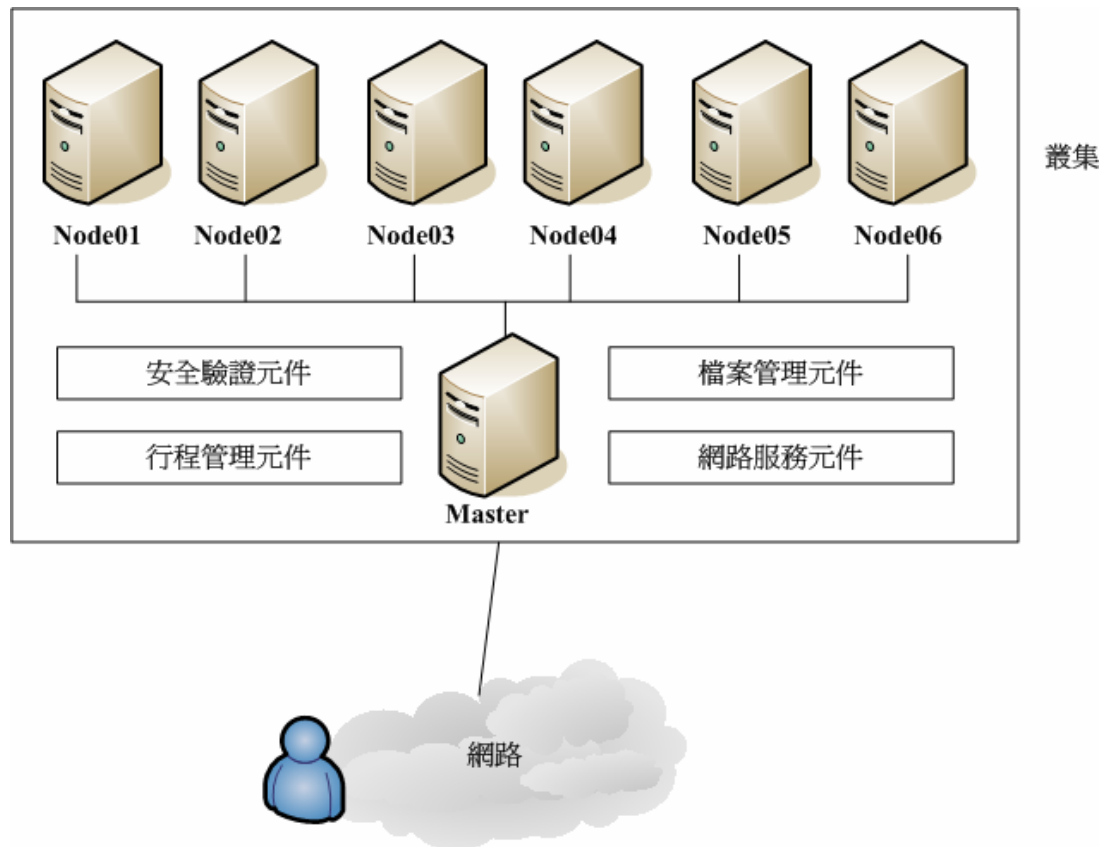
■ 最簡單的叢集架構—對稱叢集架構（Symmetric Cluster）

- 只要透過一個網路連結，即可發佈工作給叢集運作
- 但是檔案的一致性/系統安全性可能較差(沒人管)



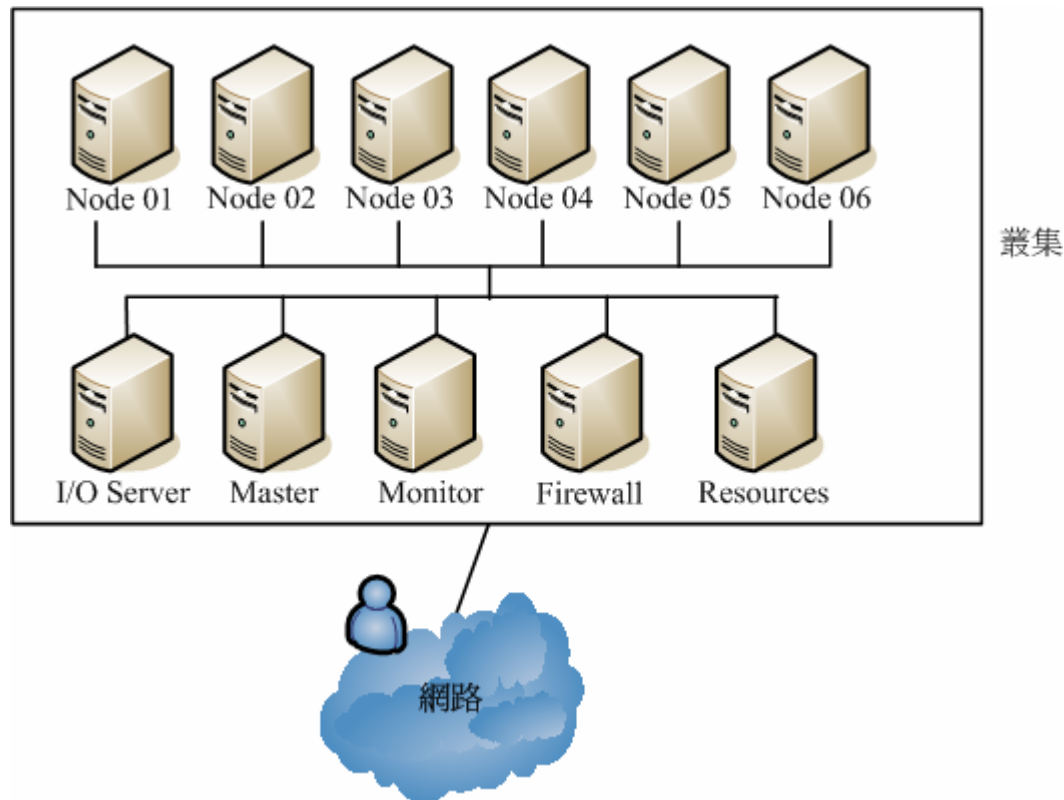
■ 非對稱叢集架構 (Asymmetric Cluster)

- 由一部主控電腦(Master)控制所有的主要元件，
- 其他的node只要有基礎的運算能力即可
- 但是Master的壓力比較大，可能易損毀



■ 延伸叢集架構(Extend Cluster)

- 將檔案管理，I/O作業，網路安全控管等分離出單一Master主機的控管



叢集系統的問題

- 所有的電腦是否具備相同的作業系統
- 如何讓所有的電腦都具有相同的檔案系統
- 帳號/群組的一致性
- 平行化處理的模組與機制
- 網路速度可能是效能瓶頸點

叢集系統的優點

- 價格：

- ☐ 各硬體是各自獨立的，較便宜

- 可靠性：

- ☐ 不會因為某一個節點出錯而導致整體系統損毀

- 擴增性：

- ☐ 可以隨時加入額外的node

- 升級：

- ☐ 可將平行化軟體優化，即可升級軟體

叢集式系統的通訊

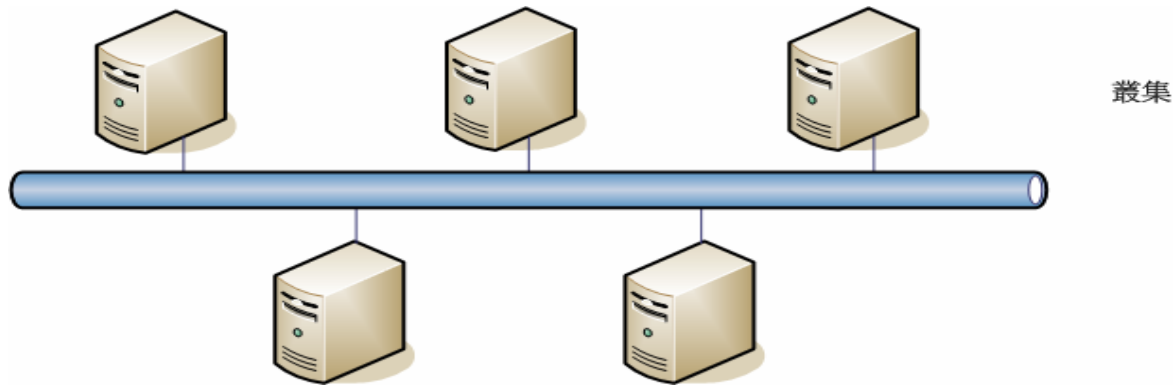
■ 網路拓撲型態：

- ☐ 匯流排拓撲（Bus Topology）
- ☐ 星狀拓撲（Star Topology） ← 目前最適用的拓撲環境
- ☐ 環狀拓撲（Ring Topology）
- ☐ 網狀拓撲（Mesh Topology）

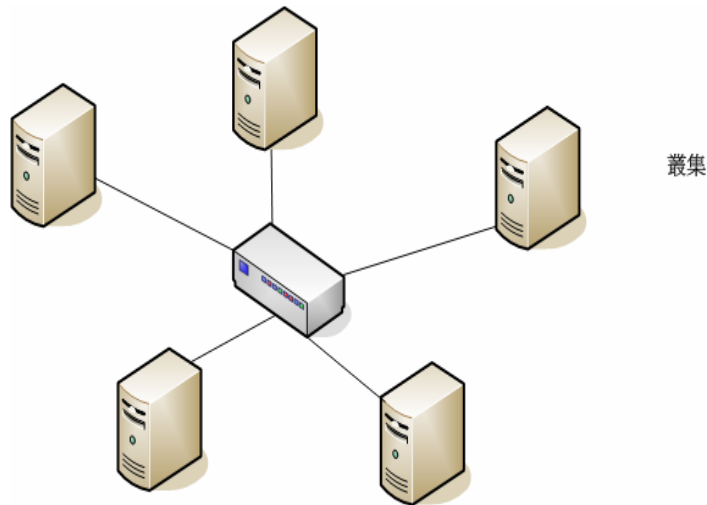
■ 選擇拓撲的依據：

- ☐ 串接時所支付的成本
- ☐ 通訊時的可靠度
- ☐ 通訊時的速度

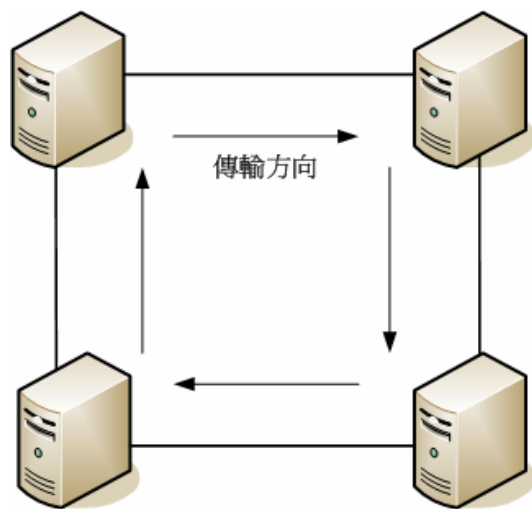
■ 匯流排拓樸 (Bus Topology)



■ 星狀拓樸 (Star Topology)

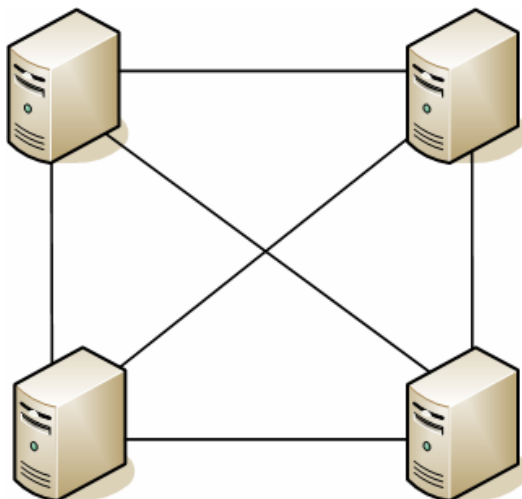


■ 環狀拓撲 (Ring Topology)



叢集

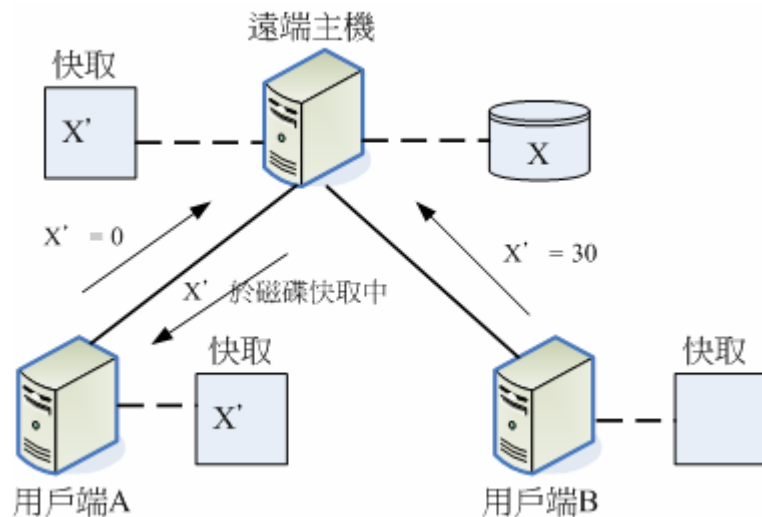
■ 網狀拓撲 (Mesh Topology)



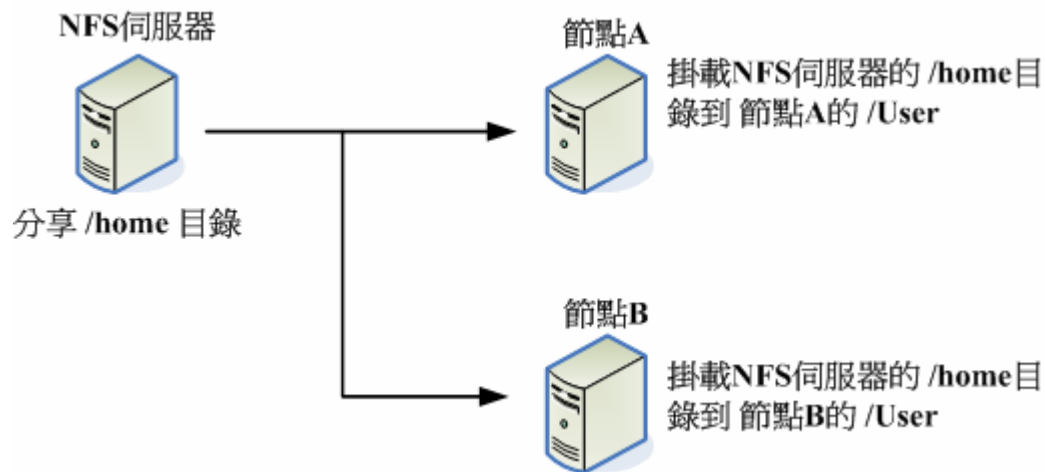
叢集

叢集式系統的檔案系統

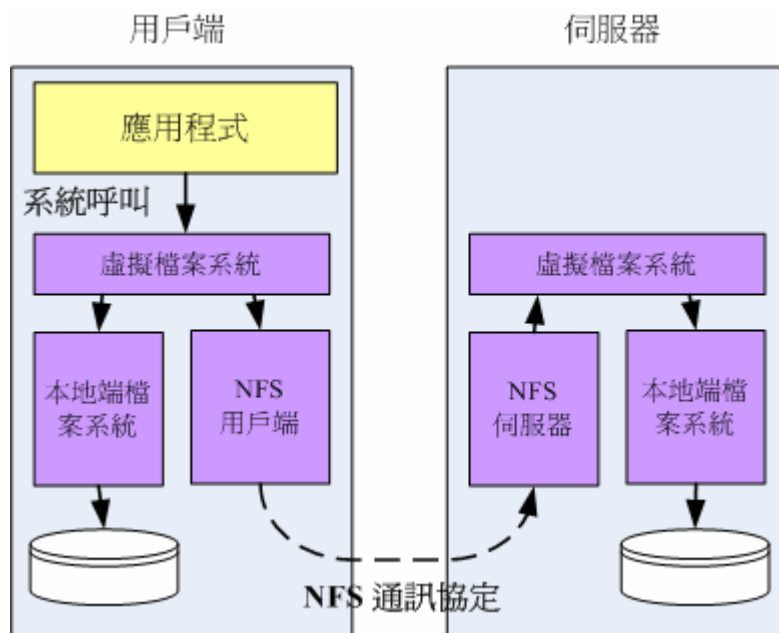
- 叢集式架構下，檔案的命名模式：
 - 主機名稱加上路徑，例如，hcservers/home
 - 將遠端檔案系統掛載到本地端的檔案階層中
 - 使用單一名稱，使所有電腦所看到的名稱皆相同
- 快取一致性問題（Cache-consistency Problem）
- 直接寫入演算法（Write-through Algorithm）



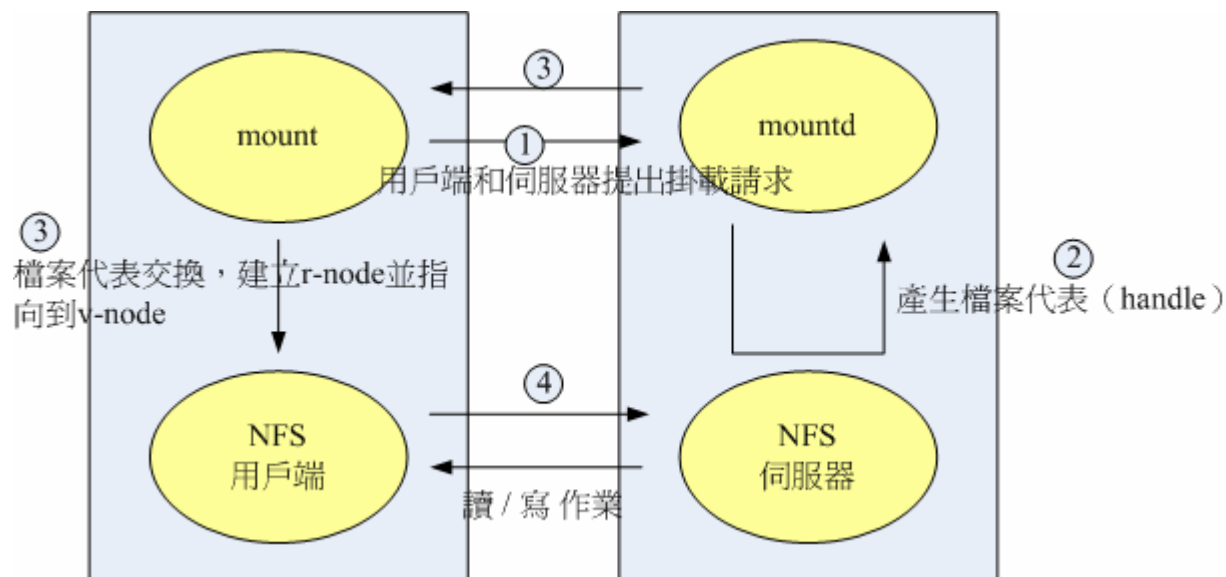
■ 網路檔案系統（Network File System，NFS）



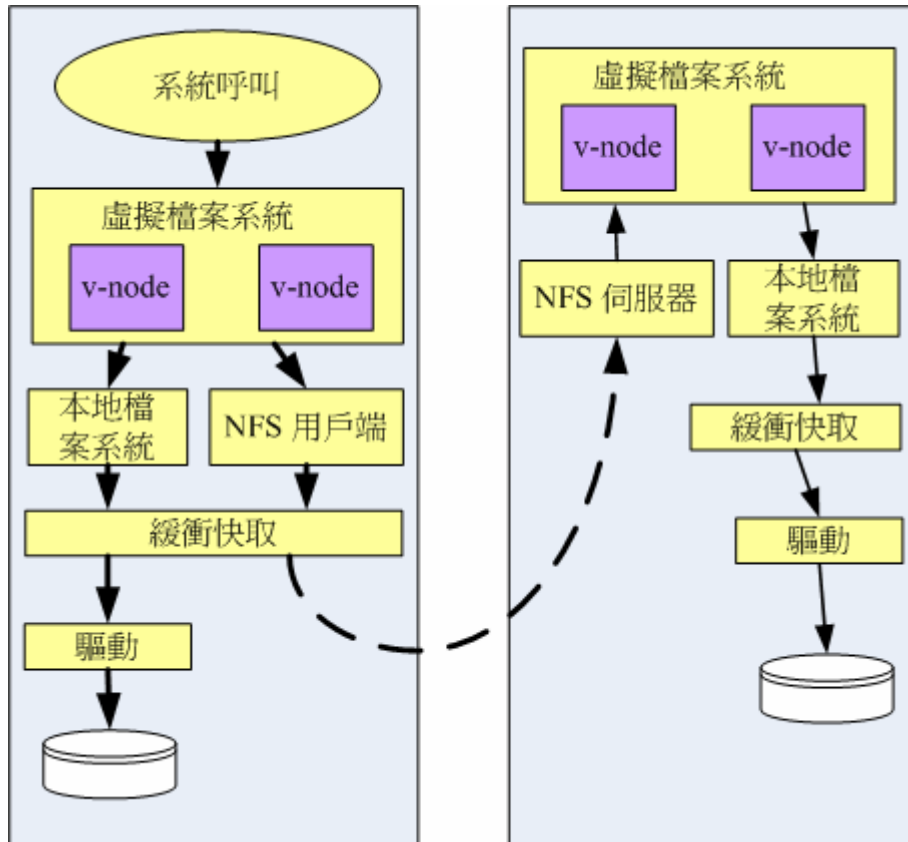
■ NFS運作層次



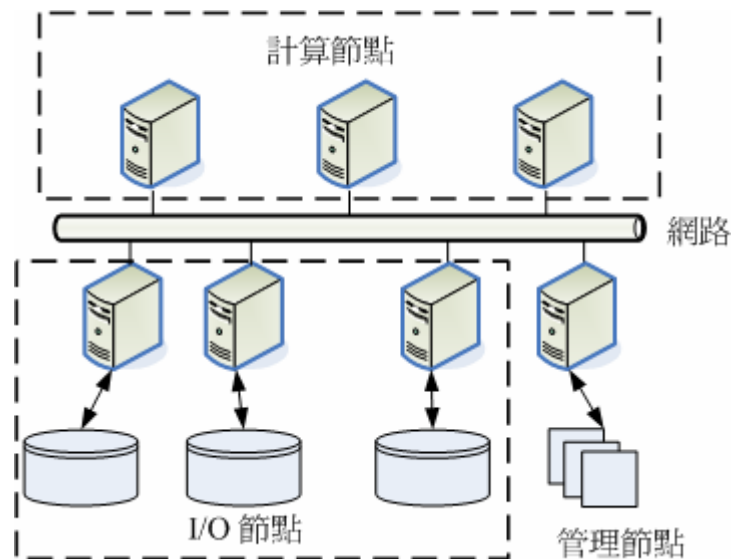
■ NFS遠端掛載運作方式



■ NFS檔案存取

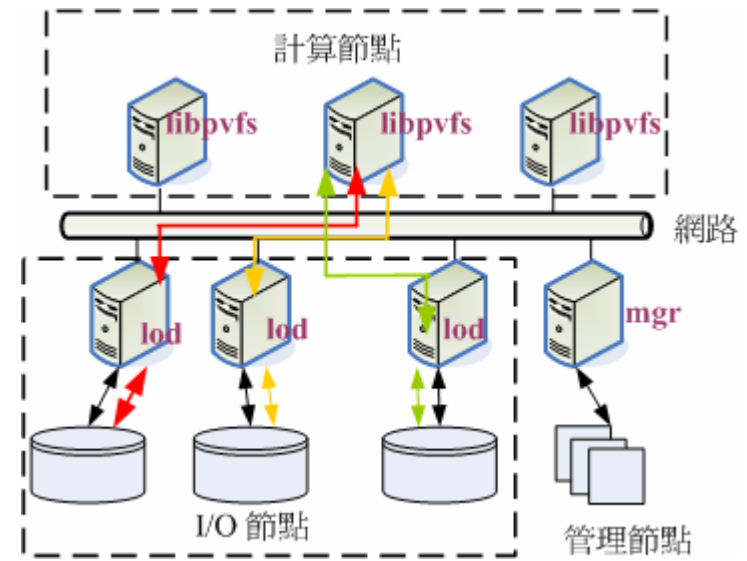
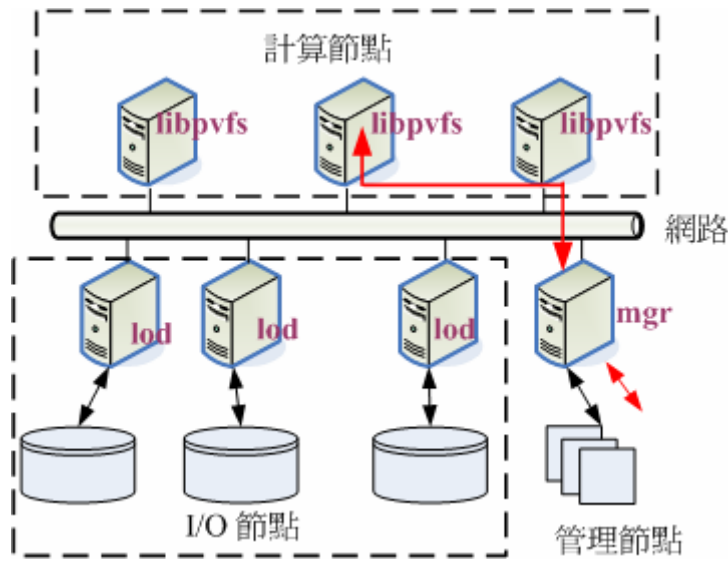
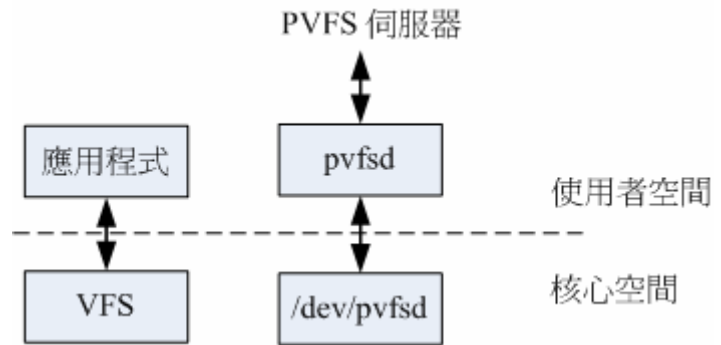


- 平行虛擬檔案系統（Parallel Virtual File System，PVFS）
- PVFS軟體中主要功能：
 - 一致性的檔案命名方式。
 - 支援現有的檔案系統存取模式。
 - 資料分散於叢集架構上的節點硬碟內。
 - 為應用程式提供更高性能的資料存取方式。



■ 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

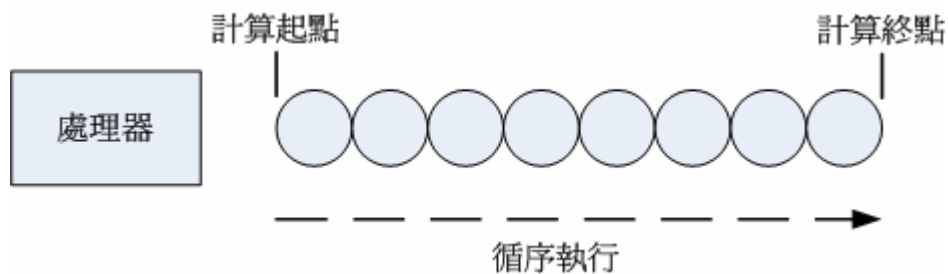
■ 在Linux作業系統中對於檔案的時間戳記表示方式大致上有三種類別，分別為：Modify time（mtime）、Access time（atime）、Inode Change time（ctime）

- 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

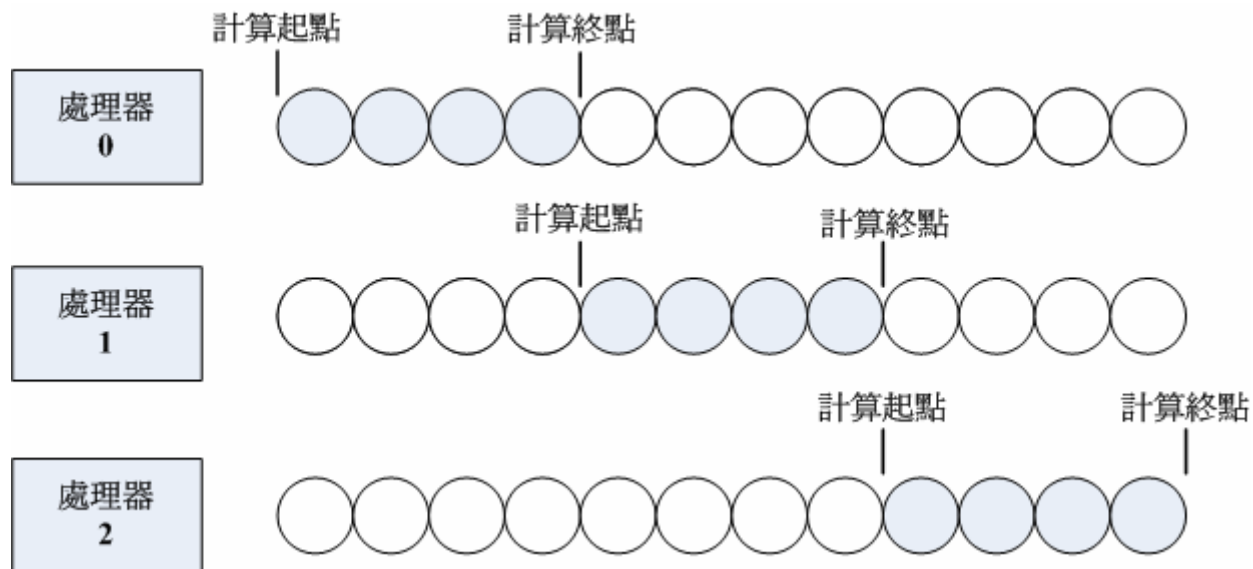
平行程式計算

- 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

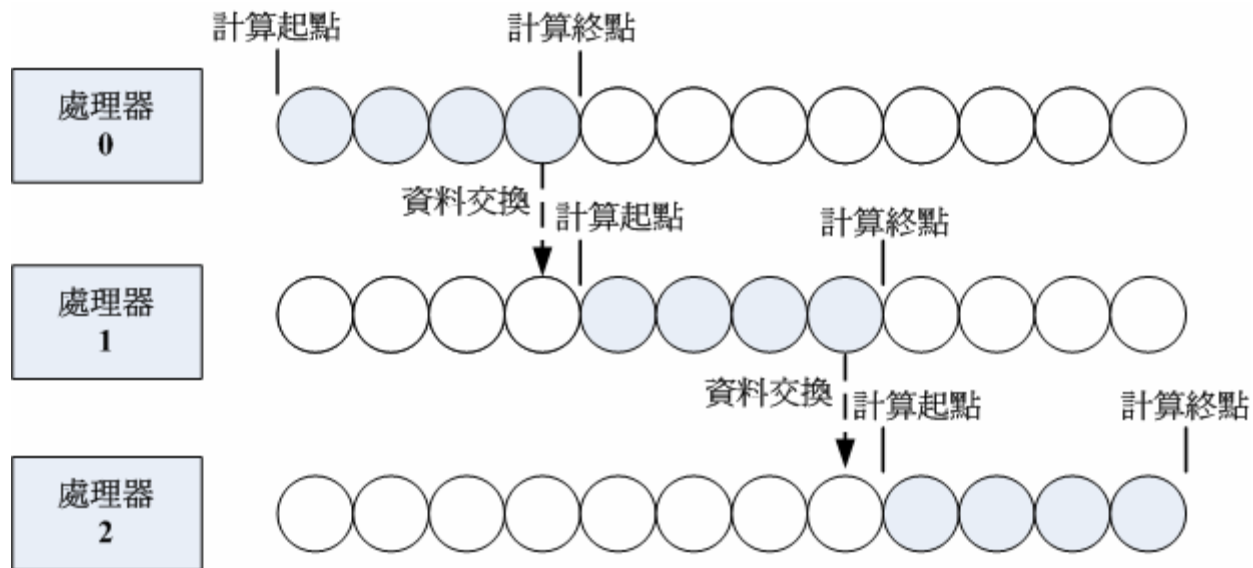
■ 單處理器循序程式運作方式



■ 計算切割而資料不切割的運作方式



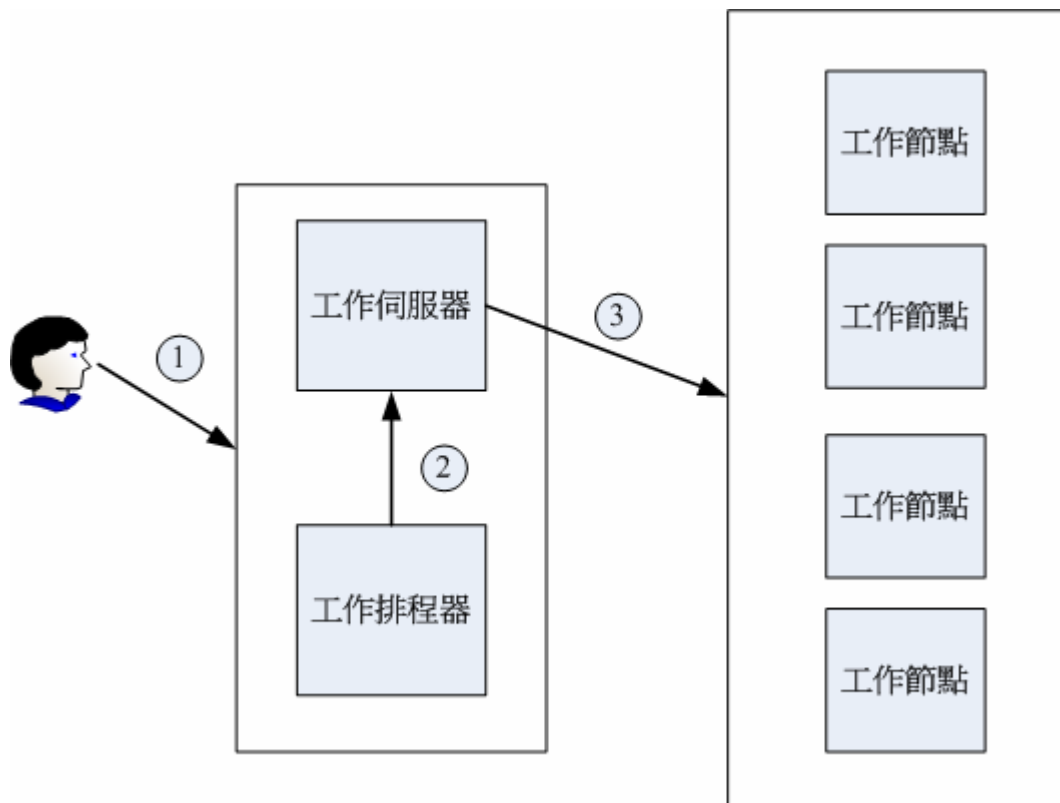
■ 計算與資料均切割的運作方式



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。

