

系統元件、服務與開發

- OS 之重要元件
  - Process Mgt.
  - Memory Mgt.
  - I/O
  - File Mgt.
  - Protection
  - Networking
  - Command interpreter

- Design Principle
  - 區別 policy & mechanism
    - policy
      - What to be done
      - change
    - mechanism
      - How to be done
      - Not change
  - Layered Approach
    - 定義
      - System 切分為數個元件在分層組織
      - Top Down 方式，上可以用下，下部能用上
      - Bottom up 測試
    - 優點
      - 模組化、利於開發
      - 易於測試、維護
    - 缺點
      - 不易劃分出明確的層次架構

- Types of system
  - Micro kernel
    - By CMU
    - 以 Mach 為代表
    - Kernel 只留下三個必要元件
      - Process mgt.
      - Memory mgt. (basic)
      - Communication
    - 優點
      - Easy to extend
      - More secure
      - Easy to port
    - 缺點
      - 效能差
      - 因為充斥著大量的訊息傳遞處理
  - Monolithic kernel
    - 所有的服務都放在 kernel
    - 一般商用系統都是
    - 優缺點與 Micro kernel 完全相反
  - Loadable kernel module
    - Easy to attach/de-attach
    - 效能比 Micro kernel 好，相較於 layered，模組間可以溝通
  - Hybrid
    - Sun Solaris
    - 實際上現代系統均偏向混合型

System Calls

- 定義/敘述
  - User process & kernel 之間的介面
  - process 請系統執行服務，系統完成後再將結果回傳
- 種類
  - Process Control
    - fork exit wait
  - File mangement
    - open close read write
  - Device mangement
  - Information maintenance
    - getpid
  - Communication
    - 針對 message passing 而言
    - pipe
  - UNIX 限定 Protection
    - chmod umask
- 參數傳遞方式
  - Register
    - 直接以暫存器保存參數，OS 再從而取得
    - 簡單、速度快
    - 不適合大量參數傳遞的狀況
  - Memory (Block in memory)
    - 參數存於記憶體，同時使用一個暫存器紀錄起始位置
    - 適合大量參數傳遞
    - 速度比前者慢
  - stack
    - Push, Pop 操作於 stack
    - 操作簡易
    - stack 空間預留不足會 overflow

Virtual Machine

- 定義/敘述
  - Software simulation 出與底層**完全一樣**的硬體之技術(原文定義)
  - Guest OS as a user process run over host OS
    - host 需滿足 guest 的所有 system call
- 優點
  - Guest OS 獨立於作為 user process 有利於隔離測試、對 host 安全無害
  - 實體機器上可以執行多個 VM、多個 OS，節省成本
  - 因應 cloud computing，VM 技術恰可以模擬出更多的機器服務 user
    - 節省成本
    - 動態調配運算資源
    - 充分發揮實體機器之使用率
- 種類
  - Para-virtualization
    - 模擬類似、但不完全一樣的底層硬體
    - Guest OS 通常需要經過修改
  - VMware
    - 模擬的完全虛擬化
    - 利用硬體(CPU)輔助處理敏感指令
    - Guest OS 不用修改
  - Simulation
    - 軟體模擬不同平台之機器
    - Play Station 模擬器
    - GBA 模擬器
  - JVM
    - flow
      - Java source code
      - Java compiler
      - Java byte code
      - Java interpreter 解譯執行之
      - 為了提升效能，又加入了 Just In Time Compiler
    - 重點
      - Its a spec.** (所以 HW SW 實作皆可)
      - Spec. 包括 (主要元件)
        - Class loader
        - Class verifier
        - Java interpreter
      - 是物件導向語言
      - 提供 garbage collection 功能