軟體工程 作業三 B0229053 黃宇薇

* Architectural Patterns（架構模式）

Architectural Patterns是一個在軟體架構中，對於經常發生的問題，可以使用一般且可以重複使用的解決方案解決。

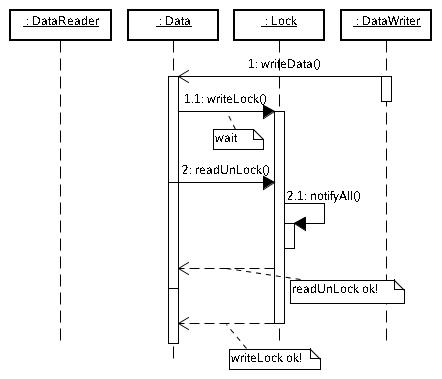
有點類似於軟體的設計模式，不同的是範圍更廣。Architectural Patterns解決各種軟體工程的問題，像是電腦硬體效能的限制、高可使用性以及最小化經營風險。在software frameworks中有一些architectural patterns已經付諸實踐。

（請繼續往下頁閱讀）

* Concurrency（併發）
  + 定義：簡單來說，此的應用為Multiprogramming以及task scheduler。前者用來進行process management，此舉可使效率增加；後者讓想要執行的task進行最有效率的scheduling。
  + 例子：Read write lock pattern

Read write lock pattern是用來保護某一項共享資源不受到兩個process同時讀寫，以免造成此項共享資源的error。

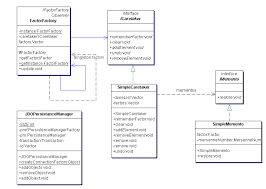
舉例像是open source -“Notepad++”，他不會讓使用者同時開啟相同的檔案兩次，以免造成錯誤，這就使用到了Read write lock pattern。



* Persistence（持久性）
  + 定義：若過去有process曾創建過data，則當process執行時data會持續存在。
  + 例子：Database Management System

Database Management System是用來管理大量資料的軟體，用來建立和維護資料庫。大部分的軟體都需要資料庫的協助跟幫忙。

舉例來說，open source -“Ubuntu”管理帳戶方面必定用了資料庫系統，他需要在資料庫中記錄帳號密碼跟帳戶資料。



* Distribution（分散）
  + 定義：一種在Distribution環境中，系統或組件之間和其他系統或組間的溝通方法。
  + 例子：MapReduce

MapReduce是一種能為大量資料做平行運算處理的software framework，他是雲端運算中最關鍵的技術。他的運作方式很像是選舉開票，誰來開票其實都沒有影響。各投票所的開票任務就像是 Map 程式，負責處理少量資料，而 Reduce 程式就是中選會彙整投票結果的動作。

完成 MapReduce 的程式後，就可以交由雲端運算的平臺來執行Reduce程式，之後再合併。

舉例來看，open source -“Open Cirrus”組織整合了 Open Source 軟體的技術來發展雲端運算，目前發展的雲算運算軟體有Hadoop、Tashi、Zoni 及 HDFS 等。

