

選課系統問題分析及其解決方案

賴秀香 蔡錫鈞

國立交通大學資訊技術服務中心

carollai@nctu.edu.tw, sctsai@cs.nctu.edu.tw

摘要

近年來國內大學對於相關經費是錙銖必較，不可能僅為了解決選課系統某幾天的高峰期，就花大筆經費添購設備。因此，無法透過增購及置換設備來改善選課系統緩慢及效能不足的問題。我們藉由虛擬技術的支援，採用低成本、不需額外經費的方式，解決選課系統的效能問題。

本文描述解決交通大學選課系統效能問題所實施的三個階段改善方案，從現況描述、問題分析至解決方案，相信這些經驗可提供給面臨類似問題的機構做為參考。

關鍵詞：選課、複寫、Nginx、負載平衡。

Abstract

In this article we show how to resolve the performance problem of online elective system without extra budget. We believe this will provide a very useful guideline to troubleshoot similar performance problems.

Keywords: online elective system, load balance.

1. 前言

許多大學的選課系統都有個一直存在的共同問題，就是每當選課期間，大批的學生同時湧入選課系統時，選課系統就會變得十分緩慢，緩慢到學生無法順利選課，嚴重時甚至會當機而無法使用。通常解決系統緩慢或系統效能不足，最先被提出的方案大都是增購或置換設備。在經費縮編、凡事錙銖必較的今日，花大筆經費添購設備，卻僅為了解決選課某幾天的高峰期？這樣的方案通常都不會被採納，而且其投資效益有限。而選課系統能否正常運作，直接關係著學生是否能選到課程，為確保學生選課的權益，如何維持選課系統於選課期間能順利運作，就成了各大學計算機中心的一個重大的課題。

對於一個每年只有幾個特定時段才會大量使用的系統進行測試及除錯原本就不是件簡單的事，因為使用量不大時，問題是不會出現的，況且該系統已歷經不同的開發者及管理者，問題的掌握更是具有挑戰性。

交通大學資訊技術服務中心對於這個問題進

行了三個階段的改善方案來解決選課系統的問題。首先我們釐清瓶頸的來源，並發現連線的負載須先解決，透過模擬壓力測試，我們適度增加 Web 主機，以三台實體機搭配兩台虛擬主機將網路大量連線的瓶頸解決。之後我們發現資料庫是另一個瓶頸的來源，經過分析資料庫讀和寫的比例，我們發現寫資料到資料庫的比例是相對很低的，於是我們複製資料庫到多台虛擬機，以分散讀取資料庫的負擔，再使用一台虛擬機負責將選課資料寫入資料庫，並將資料同步，取得一致性。至此我們已將選課系統常見的瓶頸解除，即使在尖峰時段系統塞車的問題已不復見。然而在這過程中衍生了另一問題，使用者經過一段時間後會被系統登出，亦即 session 無法共用的問題，於是我們導入 Nginx 的技術將上述問題一併解決。總之，透過新興的網路及虛擬技術，我們以很經濟的方式，有效地解決了選課系統常見的效能及穩定問題。

本文的架構如下，在第二節我們介紹解決 Web Server 的負載問題。第三節介紹資料庫瓶頸問題的解決方式。第四節討論排除 session 共用的問題。第五節是本文的總結。

2. 第一階段：Web Server 負載問題

2.1 現況描述

造成選課系統緩慢的可能原因很多，首先要找出影響系統效能的瓶頸為何。我們採用一般的做法，針對選課系統進行壓力測試，以瞭解選課系統可能的效能瓶頸。壓力測試需在與目前運作系統相同的環境下進行，原本選課系統的主機架構是兩台 Web 主機及一台 Database 主機，而當時只有一台相同規格的 Web 實體機可利用，以當下的經費狀況也不容許添購主機，只有一台 Web 實體機的狀態下是無法以相同環境模擬下測試。選課系統原本的主機架構圖如圖 1。

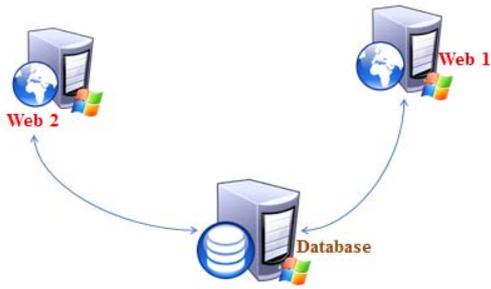


圖 1 選課系統原本主機架構圖

2.2 問題分析及解決方案

由於本校資訊技術服務中心已導入虛擬技術，在虛擬技術的支援下，複製多台與選課系統 Web Server 相同環境的虛擬主機進行壓力測試。壓力測試的過程由一台 Web Server 陸續增加 Web Server 的數量，並針對不同的同時上線人數來測試。圖 2 為壓力測試結果，結果反應增加 Web Server 數量能有效的降低平均回應時間，並提高使用人數。依壓力測試結果顯示原本選課系統兩台 Web Server 無法應付尖峰時間大量的使用者，以至於 Web Server 反應間隔時間增加而傳回大量的 500 錯誤。根據壓力測試的結果，增加至四台 Web Server 雖足以應付選課系統尖峰時間的人數，但為了能再降低平均回應時間，將選課系統增加至五台 Web Server。選課系統 Web 主機由原本兩台實體機，增加一台備援機，調整成三台實體機，再搭配兩台虛擬機，圖 3 為調整後的選課系統主機架構圖。

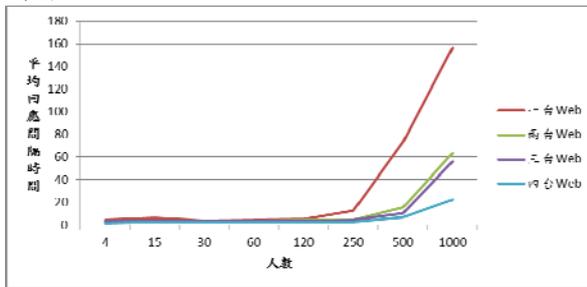


圖 2 壓力測試結果



圖 3 選課系統第一階段調整後主機架構圖

調整 Web Server 主機數量後，在選課尖峰時刻當機的次數明顯減少，系統瀏覽速度也有顯著的改善，學生也能順利選課。但是，在系統調整後，卻引發了另一個問題，課務組後端管理系統在選課尖峰時間會有無法使用的狀況。因此，需要再研擬下個階段的問題改善方案。

3. 第二階段：Database Server 負載問題

3.1 現況描述

課務組所使用的後端管理系統是由 Visual Basic 6.0 開發的 Application 系統，該後端管理系統是透過課務組桌機直接連至資料庫存取資料。為什麼後端管理系統會於選課尖峰時間無法使用？針對 Database Server 做觀察紀錄，以了解問題發生原因。

選課系統使用的 Database Server 是微軟的 MSSQL 2000 Server，選擇 MSSQL 本身提供的 SQL Profiler 工具，並搭配一些 MSSQL 監控程式，來確認後端管理系統無法使用的問題點。監控的數據顯示，後端管理系統在無法使用期間，Database Server 之 wait type 有大量 LATCH_*、LCK_*、PAGELATCH_*、PAGEIOLATCH_*、CXPACKET 等訊息。微軟亞太資料庫技術支持組官方博客網站中有提到：「Latch 是 SQL server 內部用來同步資源訪問的一個資料結構，和作業系統的 critical section 或 ReaderWriterLock 類似。Latch 保護了那些想保護的資源，使得訪問同步有序。比方說，當某個執行緒獲得某個資源的 Latch 的獨佔使用權的時候，別的執行緒如果也需要訪問這個 latch 則它必須等待[1]。Database Server 出現大量 Latch 相關的訊息，表示 Database Server 的資源已無法負荷 5 台 web server 及後端管理系統同時使用。」

3.2 問題分析及解決方案

經由分析監控數據得知，選課 Database Server 有很大的比例都是在提供 Web Servers 的資料讀取，寫入的比例相對的很低。因此，我們採取的作法是將讀取及寫入資料分別使用不同 Database Server 的方式，以期能分散資料庫的負載量。讀取及寫入資料庫的方案，需要同時搭配「讀取資料庫」、「寫入資料庫」與「同步資料的機制」。「讀取資料庫」是利用原本在運行的選課系統 Web Server 虛擬主機再加裝 MSSQL 2005 Server，Web 與資料庫同在一個主機上，這樣的架構不僅能節省建置 Database Server 的成本，還可以加速資料讀取速度；「寫入資料庫」則由原選課系統 Database Server 負責處理；「同步資料的機制」是使用 MSSQL 提供的交易式複寫機制。

圖 4 是選課系統「讀取」及「寫入」資料分別使用不同資料庫的架構圖，選課系統 Web Server

在讀取一般性資料時，是透過本機 Database Server(local DB)取得資料；寫入資料還是維持寫入選課系統 Database Server。然而遇到重要資料需要確認時，還是必須透過選課系統 Database Server 取得資料，避免因資料同步時差而產生其他問題。

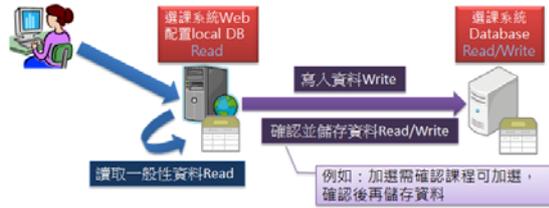


圖 4 選課系統「讀取」及「寫入」資料分別使用不同資料庫的架構圖

MSSQL 的複寫機制中有三種角色，分別為「發行者」、「散發者」、「訂閱者」。圖 5 是 MSSQL 交易式複寫架構圖，「發行者」產生發行集，發行集可以是資料表、預存程序、檢視表...等。「散發者」會先利用「快照集代理程式」取得一份「發行者」所提供的發行集快照，之後只要發行集有任何交易紀錄，會再透過「紀錄讀取器代理程式」將交易紀錄傳送至「散發者」端。「散發者」確認與各「訂閱者」連線正常，會透過「散發代理程式」將發行集資料送至各「訂閱者」資料庫中。

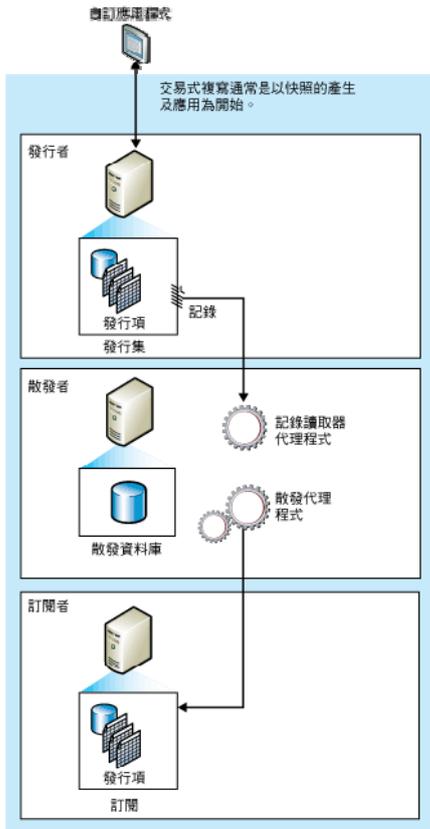


圖 5 MSSQL 交易式複寫架構圖[2]

「發行者」及「散發者」原可以同時由一個 Database Server 擔任，但「散發者」在記錄交易紀

錄資料及散發資料給各「訂閱者」時會影響正式資料庫的運作，因此，如圖 6 所示，選課系統的資料複寫架構建立了三種角色的 Database Server 各司其職，將選課系統 Web Server 「寫入」到選課系統 Database Server 的異動資料，複寫回 Web Server 的資料庫(local DB)中，以供「讀取」使用。

實行讀取及寫入資料庫的改善方案，不僅改善了後端管理系統於選課尖峰時間會無法使用的問題，也大幅改善 Web Server 在尖峰時間，瀏覽速度會降低的狀況，尖峰時間使用選課系統已與一般時間無異。

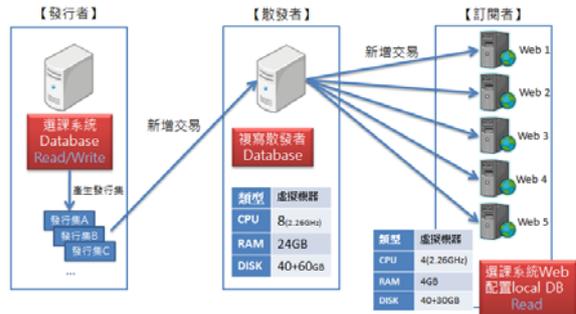


圖 6 選課系統資料複寫架構

4. 第三階段：Session 無法共用問題

4.1 現況描述

選課系統歷經兩階段改善方案後，已無使用者反應系統過慢問題，但卻陸續有使用者反應會有登入選課系統後隨即被登出的現象。經由交叉比對 IIS Log 與使用者登入資訊 Log，發現同一個帳號登入前後分別使用不同的 Web Server。當時，選課系統 Web Server 是用 DNS Round Robin 的方式隨機分配給使用者，只要網路中斷或者其他原因導致重新分配 Web Server 時，使用者會因為登入的 Session 不存在而被登出。一般解決此問題常使用的方式是將所有 Web Server 的 Session 共用就可以解決，但選課系統的開發語言是 ASP，ASP 的 Session 並不像 PHP 是以檔案形式儲存，而是存放在主機記憶體中，無法直接共用。除了共用 Session 外，還有另一種解決方式，讓同一個 IP 固定連相同的 Web Server 也能解決這個問題。Nginx 是兼有負載平衡的 Web Server，它的負載平衡機制只要設定成 ip_hash，就可以將同一個 C 網段的 IP 指向到相同的後端 Web Server。於是我們決定導入 Nginx 的負載平衡機制來解決 Session 的問題。

4.2 問題分析及解決方案

圖 7 是選課系統導入 Nginx 後的選課系統架構圖，選課系統 Web Server 被建構在 Intranet 中，使用者只能透過 Nginx 指向到選課系統 Web

Server，所有與 Web Server 的連線也都只能透過內部網路溝通，這樣的架構除了能解決 Session 無法共用問題，也能增加選課系統 Web Server 的安全性。

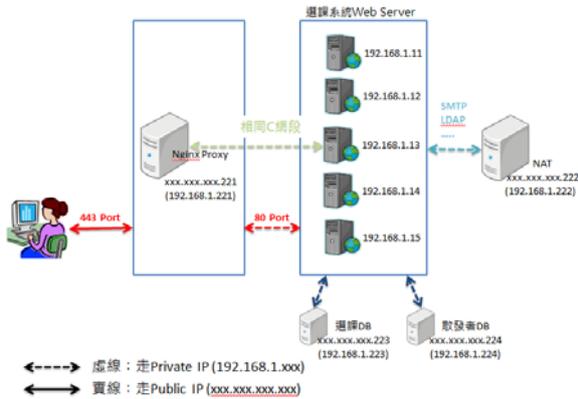


圖 7 選課系統導入 Nginx 架構圖

5. 結論

在科技日新月異的年代，系統為了因應快速變遷的環境，時時刻刻都可能面臨一些莫名、未知的問題。而且經常一個問題解決，卻又引發另一個問題的發生。解決問題除了憑藉系統負責人日積月累的經驗，再搭配他人研究發展的技術，慢慢的抽絲剝繭分析問題，找出可能的解決方案。找出快速又有效的解決問題方式，是所有系統負責人都希望的，交大在解決這個問題的過程中獲得許多寶貴的經驗，希望能提供給面臨類似問題的機構一個參考。

參考文獻

- [1] 什麼是 PAGELATCH 和 PAGEIOLATCH,
<http://blogs.msdn.com/b/apgcdsd/archive/2011/11/29/pagelatch-pageiolatch.aspx>,
Retrieved July31, 2013.
- [2] Transactional Replication,
<http://msdn.microsoft.com/zh-tw/library/ms151176.aspx>,
Retrieved July31, 2013.