

負載平衡架構之資料庫最佳配置分析

¹ 李昆達

² 包蒼龍

¹ 銘傳大學資訊網路處

² 大同大學資訊工程學系

摘要

負載平衡架構可以解決伺服器負載過重的問題，由於即時性目前幾乎所有網站所回應之網頁內容都是搭配資料庫系統不斷動態更新產生。因此負載平衡架構中資料庫的配置方式，將會對整體的效能有著極大的影響。本文將分析集中式資料庫架構和非集中式資料庫架構對負載平衡架構效能的影響，藉由實驗來找出資料庫最適合的配置方式。從實驗結果得知，當使用者連線數量小時，使用非集中式架構可有效降低平均回應時間，因為網頁伺服器與資料庫伺服器間不需要透過網路傳輸資料，減少了傳輸延遲。但是，當使用者連線量大時，使用集中式架構能夠可得到較佳的回應時間，這是因為分散式架構，每部網頁伺服器上都安裝有資料庫伺服器，造成網頁伺服器負載加重，影響效能。而集中式架構則因為所有網頁伺服器可以共用一部資料庫伺服器，因此不需要額外的運算處理，雖然會有與資料庫伺服器間的網路傳輸延遲，但整體效能要比分散式資料庫配置者要佳。

關鍵字：負載平衡、集中式資料庫架構、非集中式資料庫架構

1. 簡介

網際網路的發展快速，網路應用也越來越多樣化。當網站的因用戶增加而流量變大時，如何避免伺服器超載是一個很重要的課題。用戶數多的網站，它需要強大的硬體資源及設備來提供服務。然而，隨著網站服務項目及流量的增加，硬體設備的投資成本也跟著水漲船高。為降低建置成本，採用負載平衡架構可能是最適切的解決方案，因為負載平衡架構之運作兼具效能提升與彈性升級[1]，管理者可以視網站負載狀況，輕鬆地增加或移除後端伺服器。靈活的運用負載平衡架構可以有效提高網站整體的運作效能。

然而，受歡迎的網頁內容大多都是利用動態方式不斷更新，動態網頁內容是由資料庫中讀取而來的，所有資料都存儲在資料庫伺服器中。當使用者發出請求瀏覽資料時，網頁伺服器自資料庫讀取資料並客製化回應的內容，針對不同使用者請求，回應其所要求的內容。結合後端資料庫，網頁可以更加多元化，網站可以提供最新的資訊給使用者。

在本篇論文中，我們將針對集中式和非集中式資料庫[4,5]架構的效能進行分析，在我們所建構的實驗環境中，我們採用分配器(dispatcher)來做負載平衡控制，將用戶連線依 Round Robin

演算法分配到多台網頁伺服器。藉由控制多台可依設定發出大量連線的客戶端電腦，實驗存取兩種不同架構資料庫之網頁效能，藉由實驗數據加以分析，以討論在不同負載狀況下資料庫配置對整體效能的影響。

本文架構分成以下幾個部分，第 2 節簡單介紹相關的負載平衡系統，第 3 節討論系統架構及實驗環境，第 4 節分析實驗結果，第 5 節為結論

2. 相關文獻

負載平衡架構及演算法有許多種，它可以運用軟體、硬體或軟硬體混合使用[8]，不同的使用需要有不同的架構，本節將對負載平衡架構及演算法相關研究加以討論。

2.1 DNS 輪詢負載平衡架構

這類負載平衡的方式是透過網域名稱伺服器將單一網域名稱對應到多個 IP 位址的方式來做負載平衡，每個 IP 會對應到一台後端網頁伺服器。當使用者以網域名稱要求連線時，必須先向負責解析該網域之網域名稱伺服器查詢 IP 位址，此伺服器會利用輪詢的方式依序回應不同之後端伺服器之 IP 位址給不同的使用者，最後使用者再向所查到之 IP 位址的伺服器發出連線要求。

在網域名稱系統中，網域名稱伺服器之管理者可以設定多個不同的 IP 位址對應到同一個網域名稱，管理者對每一個網域名稱也可以個別設定查詢結果的有效存活時間(TTL)，為了避免用戶與管理此網域名稱之伺服器間其它的網域名稱伺服器或使用者電腦本身會記住查詢的結果，進而導致負載不平衡，這類負載平衡系統通常會在設定非常小的存活時間 (TTL) [7,9]。當用戶端發出網域名稱查詢這個負載平衡系統之網域名稱時，負責此網域名稱的伺服器利用輪詢機制選出其中一個 IP 位址，並回傳給使用者端的網域名稱伺服器，使用者主機即可收到該 IP 位址。若用戶端之網域名稱伺服器因存活時間短而每次都重新查詢，藉由此輪詢機制之運作，即可將用戶端的連線請求依序導引至不同的後端網頁伺服器，進而達成負載平衡的目的。

2.2 HTTP 重導式負載平衡架構

這類機制是透過 HTTP 協定[2]中重導機制來達成負載平衡的功能，一般在網頁伺服器前端使用分配器來接收使用者的請求，分配器再將利用 HTTP 重導機制將所選定的伺服器 IP 或網址回應給使用者端，使用者端之瀏覽器會重新再向重導訊息所指定的伺服器發出請求。這種重導機制是透通的，一般使用者不會感覺到有任何的不同。

利用 HTTP 重導機制並不需要修改網頁伺服器之 IP 位址，收到連線請求時，分配器選擇伺服器群中最適切的一台伺服器，接著發送重導封包給使用者端，封包裡包括所選定網頁伺服器之 IP 位址或網域名稱 [10]，使用者端之瀏覽器收到封包後將連線轉向該網頁伺服器。分配器在分配前必須了解網頁伺服器負載狀況，通常的做法是由每台伺服器定時會將負載狀況回報給分配器，訊息包括連線數、記憶體及處理器使用狀況，分配器則依照每台伺服器負載狀況，選擇最佳的在伺服器提供給使用者。

2.3 閘道式分配器式負載平衡架構

閘道式分配器架構是在網頁伺服器群與使用者間透過閘道器介接，由閘道器擔任分配器的角色，由於閘道式分配器轉送封包的效能會直接影響負載平衡系統的可擴充性，例如閘道器一次最多能接受的使用者連線數及後端伺服器的最多個數，另外也會影響到後端網路的使用率，故轉送機制及轉送的效能就變得很重要。現有的轉送機制分為兩大類：一種為第四層封包轉送機制，另外一種為第七層請求 (request) 轉送機制，因為所有使用者的請求都被送往分配器，再由分配器分配給後端伺服器。

分配器為使用者與伺服器的中間溝通橋樑，它提供一個虛擬的 IP 位址(VIP, Virtual IP Address)，一般來說 VIP 就是分配器的 IP 位址。以使用者的觀點來看，使用者只知道 VIP，至於分配器內部如何運作，對使用者而言都是透通的。

3. 系統架構與實驗環境

3.1 系統架構

集中式資料庫配置方式如圖 1 所示，在此架構中，網頁伺服器需要透過網路存取資料庫伺服器的內容，為了避免瓶頸出現在資料庫伺服器上[3,6]，影響我們對負載平衡效能的評估，我們使用性能較高之硬體來擔任集中式架構中的資料庫主機。分配器配合輪詢算法來分配使用者請求到後端之網頁伺服器。當使用者端發出請求，分配器利用重導機制將使用者分配到所選定之網頁伺服器。我們的重點是動態網頁內容，網頁伺服器必須藉由讀取資料庫之內容後產生動態網頁，而該網頁伺服器將持續服務該使用者直到其結束連線為止。

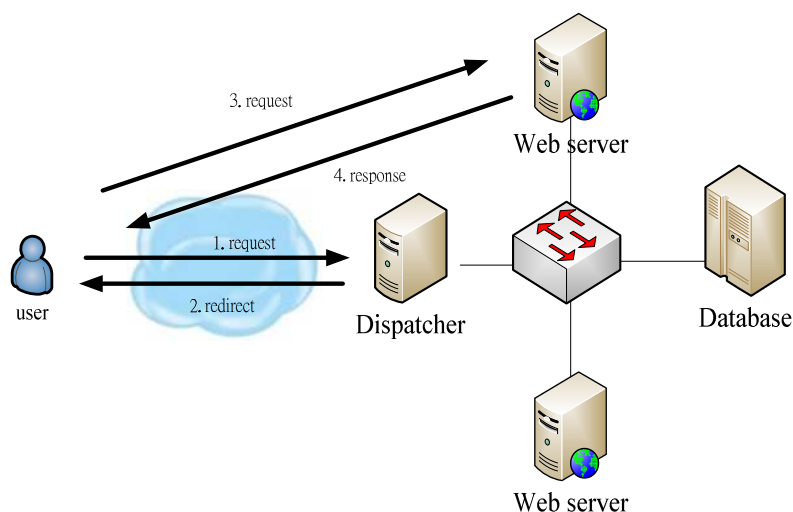


圖 1. 集中式資料庫系統架構

非集中式資料庫配置方式如圖 2 所示，在此架構中，除分配器外，所有主機同時兼具網頁伺服器與資料庫伺服器功能。當使用者發出請求時，分配器將分配使用者到最適切的後端網頁伺服器。由於網頁伺服軟體與資料庫伺服軟體都在同一台伺服器上，提取資料庫之資料時並不需要額外的網路傳輸時間，一般來說，要客製化一個網頁通常都需要多次向資料庫提取資料。因此，若網路傳輸延遲比較明顯，非集中式的效能應該優於集中式配置。

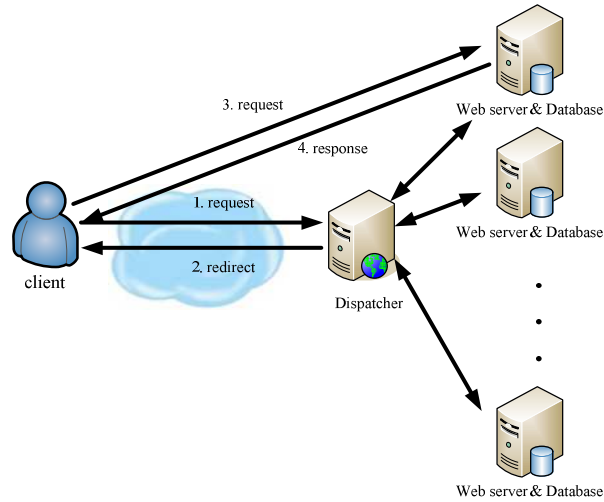


圖 2. 非集中式資料庫

兩種架構比較如表 1，兩種架構都有其優點和缺點。在集中式架構下資料庫軟體只需要一個，但非集中式的架構，如果有多台伺服器，必須取得到多套資料庫軟的授權，在集中式架構的資料庫伺服器，管理者只需要維護一台主機的內容。但是，在非集中式架構，管理者需要負責多台主機的內容。在非集中式架構伺服器為 N 台主機，此時需要伺服器總數為 N+1。此外，集中式架構需承擔單點失效問題，而非集中式架構則有多份資料當備援。在第 4 節將對實驗結果加以分析說明。

表 1. 兩種架構比較(假設共有 N 台網頁伺服器)

配置架構	資料庫授權數	管理維護	伺服器數量	單點失效
集中式	1	容易	N+1	是
非集中式	N	較難	N	否

3.2 實驗環境

表 2 列出我們實驗的硬體設備。我們的重點是在負載平衡系統中找出資料庫的最佳配置方式，因此，我們模擬真實環境，利用四台網頁伺服器服務使用者連線。一般情況下資料庫伺服器只有一台。這資料庫伺服器可能成為整個系統的瓶頸，所以我們採用規格較佳的硬體，以避免這種情況。紀錄伺服器是用來記錄實驗資料。在此實驗中我們利用 20 台主機來發出大量連線。

表 2. 硬體設備規格表

Device	CPU	RAM	OS	No.
Web Server	Intel Pentium 3.0GHz	2G	FreeBSD7.0	4
Database Server	Xeon1.6 GHz*4	16G	FreeBSD7.0	1
Log Server	Xeon 1.6 GHz	2G	FreeBSD7.0	1
Client	Intel Pentium 3.40GHz	2G	FreeBSD7.0	20

4. 實驗結果

在本節中，我們將描述實驗結果和分析。針對平均回應時間加以分析說明。使用者分別發出大量的請求，計算出回應時間。首先我們觀察集中式和非集中式架構的差異。我們在集中式資料庫架構下，分別比較二台、三台以及四台 web server 效能資料。在圖 3 中可以發現，在可承載的平均連線數上，二台 web server 每秒 580 個連線數時呈現負載過重，四台 web server 的連線數在每秒 1000 個左右呈現負載過重，伺服器的處理器在程序的分配上產生忙碌狀態。

在非集中式資料庫的架構下，由圖 4 可以得知，二台網頁伺服器在每秒 570 人時伺服器處理器負載量開始升高，當網頁伺服器增加三台及四台反應時間可以有較好的表現，由平均回應時間 0.01 秒來看，每秒連線可以到 700 人左右。所以實驗指出 database 及 web server 安裝在同一台主機上，系統同時處理網頁及資料庫的情況下，出現系統忙碌需要更多的資源來幫忙處理，在 CPU 與記憶體的需求比集中式資料庫來得多。

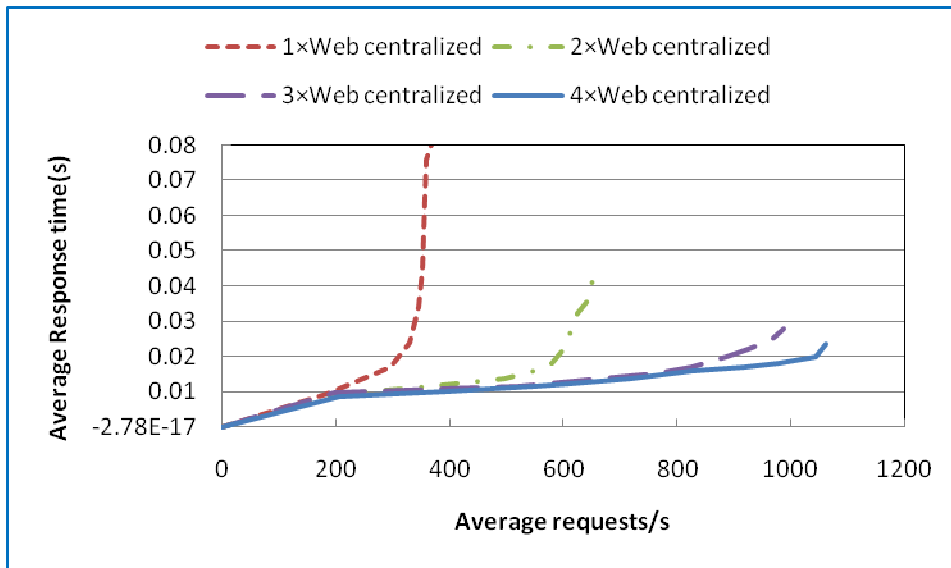


圖 3 集中式架構下的平均回應時間

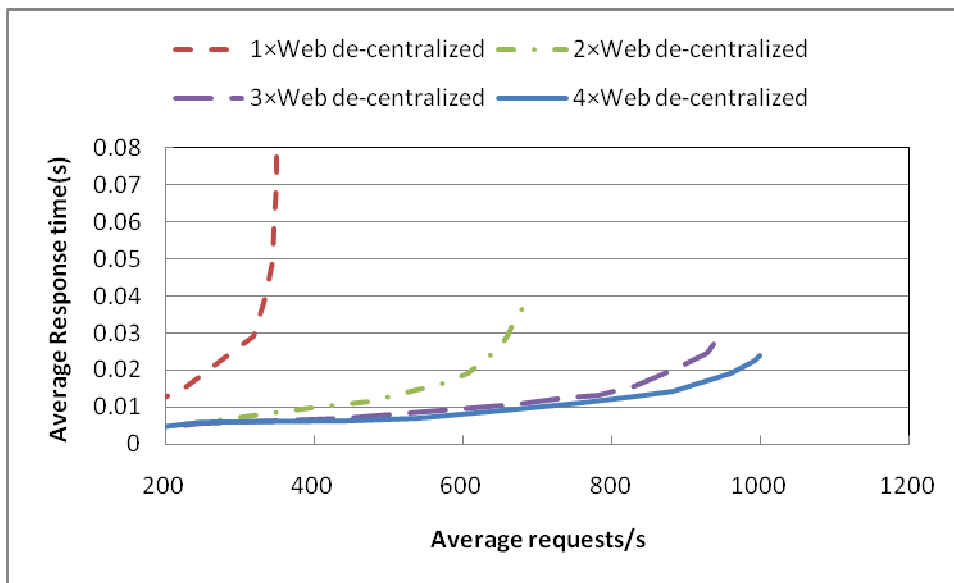


圖 4 非集中式架構下的平均回應時間

在實驗中我們在另外實做測試廣域網路對效能的影響，我們在台北及桃園各一台 web server 來分析，由圖 5 可以看出在平均每秒 580 人左右時集中式與非集中式會有交叉點，對照圖 6 所示兩台 web server 都在台北來分析連線時間及每秒連線數，由實驗結果得知在平均連線時間上伺服器在桃園所花的時間較長。

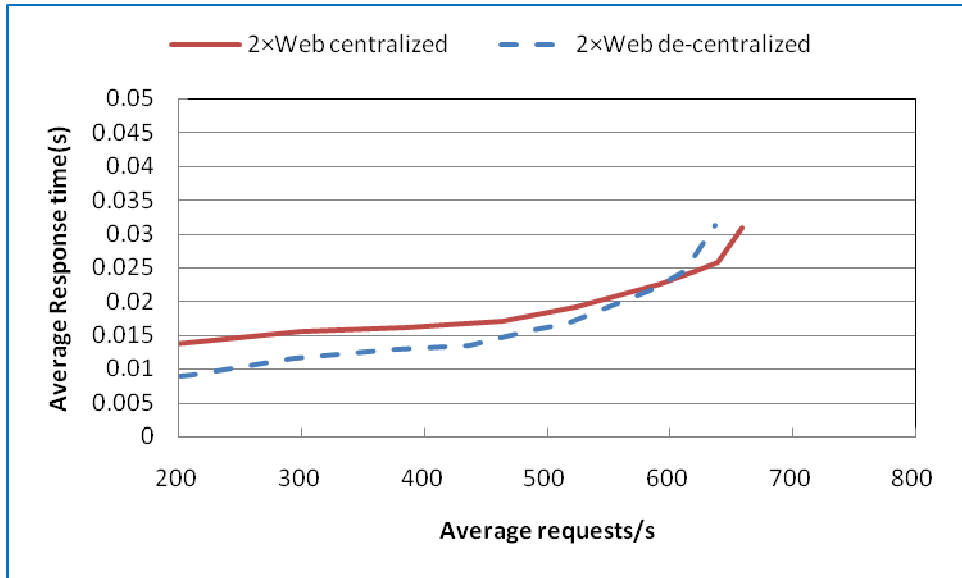


圖 5 台北桃園各一台伺服器

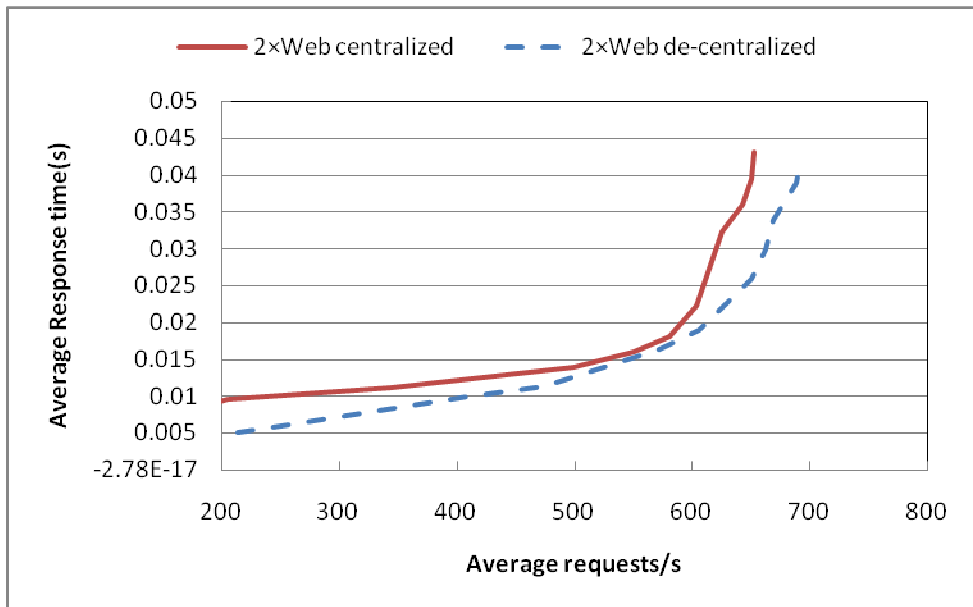


圖 6 兩台伺服器在台北

5. 結論

動態網頁讓網站更具吸引力，為了產生動態網頁，網頁伺服器必須查詢後端資料庫的內容。資料庫伺服器配置成為一個重要因素，它可以決定負載平衡系統的效能。實驗以集中式和非集中式資料庫伺服器架構分別加以評比。根據實驗結果，我們可以得知，當大量的使用者請求在短時間內到達時，非集中式較集中式好，因為資料庫就在網頁伺服器主機中，不需要透過網路傳輸。但是，當使用者的請求在一定數量時，集中式效果優於非集中式，因為 Web server 共享 database server，Web server 不需處理資料的查詢。基於這些實驗結果，管理員可以選擇合適的安排，他的負載平衡系統。

6. 參考文獻

- [1] Chang Li, Gang Peng, Kartik Gopalan, and Tzi-cher Chiueh, "Performance Guarantees for Cluster-Based Internet Services," Proc. 23th International Conference on Distributed Computing Systems, pp. 378-385, May 2003
- [2] C. Kopperapu, "Load Balancing Servers, Firewalls, and Caches," Wiley Computer, Jan. 2002.
- [3] D. Haney and K.S. Madsen, "Load-balancing for MySQL," Kobenhavns Universitet, 2003.
- [4] Eunmi Choi, Yoojin Lim, and Dugki Min, "Performance Comparison of Various Web Cluster Architectures," Lecture Notes in Computer Science, vol. 3398, pp. 617-624, May 2005.
- [5] Jiani Guo and Laxmi N. Bhuyan, "Load Balancing in a Cluster-Based Web Server for Multimedia Applications," IEEE Trans. Parallel and Distributed Systems, vol. 17, no. 11, pp. 1321-1334, November 2006.
- [6] J.M. Hellerstein, M. Stonebraker and J. Hamilton "Architecture of a Database System," Foundations and Trends, pp.141-259. Oct. 2007.
- [7] Michele Colajanni and Philip S. Yu, "Adaptive TTL schemes for Load Balancing of Distributed Web Servers", ACM Trans. Sigmetrics Performance Evaluation Review, vol. 25, no. 2, pp. 36-42, September 1997.
- [8] Philips S. Yu, Valeria Cardellini, and Michele Colajanni, "Dynamic Load Balancing on Web-server Systems," IEEE Trans. Internet Computing, pp. 28-39, May/June 1999.
- [9] S. Yu Philip, M. Colajannin and V. Cardellini, "Dynamic Load Balancing in Geographically Distributed Heterogeneous Web Servers," IEEE International Conference on Distributed Computing Systems (ICDCS'98), Amsterdam, Netherlands pp.26-29 , May. 1998.
- [10] W3C World Wide Web Consortium, <http://www.w3c.org>.