

行政院國家科學委員會專題研究計劃成果報告

協力式 Web 代理伺服器之協力式快取之研究設計(I)

Design of Collaborative Caching for Collaborative Web Proxy Server

計劃編號：NSC87-2213-E-009-015

執行期限：民國 86 年 8 月 1 日至民國 87 年 7 月 31 日

主持人：吳毅成 執行單位：國立交通大學資訊工程系

一、中文摘要

全球資訊網(World Wide Web; WWW)是 1993 年來，最重要的流行趨勢之一，急速的資訊成長增加了網路及伺服器的負擔，同時也增長了文件擷取的時間。對這個問題，最常用的解決辦法是提供一具有快取功能的 Web 代理伺服器 (proxy server)。然而因為資訊量的成長過快，目前的 Web 代理伺服器在運作上仍有一些效能上的問題。

本計畫提出了一個叫物件位置表 (Object Location Table; OLT) 的方法，能結合各 Web 代理伺服器的資源，使得資料擷取的速度能加快，而且各 Web 代理伺服器間的工作量能儘量均衡，我們將得一數據來佐證我們的想法。

關鍵詞：全球資訊網、Web 文件、Web 伺服器、Web 代理伺服器、階層式快取架構、全域式資源索引、物件位置表

Abstract

World Wide Web (WWW) has been the most popular Internet service, since 1993. Thus, the information growth over WWW also cause more load on network and servers.

In order to solve this problem, the common solution is to support a proxy server with cache functions. However, due to the fast growth of the Internet information, there

are some problems in performance of proxy servers.

In this project, we propose a method based on a mechanism, called object location table (OLT), that can combine the resources of Web proxy servers and make information retrievals more efficient. In our experiments, it shows this method has better efficiency.

Keyword: World Wide Web (WWW)、Web Document、Web Server、Web Proxy Server、Hierarchical Cache Arrangement、Global Resource Indexing (GRI)、Object Location Table (OLT)

二、計劃緣起及目的

根據 Braun 等人的報告結果顯示，從 1993 年以來 WWW 在網際網路 (Internet) 上的通訊量，幾乎以每 10 週成長一倍的幾何級數成長。這樣高成長的使用率相對地亦帶來了許多新的問題及挑戰。其中最重要的問題是：由於擷取的資訊量變得越來越大，使得網路頻寬越來越不足，而且伺服器的負載亦越來越大，這增加了文件擷取所需的時間。

對這個問題，最常用的解決辦法是提供一具有 cache 功能的 Web 代理伺服器 (proxy server)。現有的 Web 代理伺服器如 Harvest cache 和 Squid 都有支援階層式的快取架構，使用一種名叫 ICP(Internet

Cache Protocol)的協定來互相查詢有無 Web 文件，以進一步統合各 Web 代理伺服器的資源。但是目前階層式的快取架構會有下列的問題：

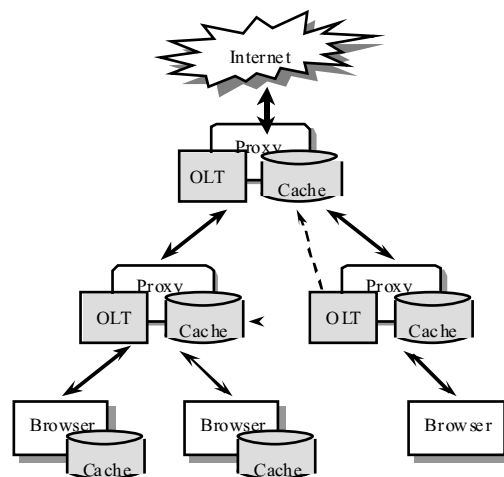
- (1)設定上的麻煩。當有新的 Web 代理伺服器加入時，其他想和它合作的 Web 代理伺服器都必須更改設定。
- (2)會增加 cache miss 的延遲時間。
- (3)使用 ICP 查詢因為只能等一預設的固定時間，造成詢問範圍無法太大。

本子計畫主要研究如何結合各 Web 代理伺服器之間的資源，使得資料擷取的速度更快，伺服器的負載不會提高太多，而且不會有上列的問題。

三、我們的解決方案

針對 Harvest cache 和 Squid 的階層式快取架構的缺點，我們提出一個新的構想來輔助它。我們參考了中正大學 Sun Wu 和 Chang-Chain Liao 所提出一套名叫全域式資源索引 (Global Resource Indexing, 簡稱 GRI) [5] 的方法，進而提出我們自己的解決方案。在我們的新構想中，一個 Web 代理伺服器除了 cache 之外另外維護一個 table —— 我們暫時命名為 OLT (Object Location Table)。只要用 Web 文件的 URL，就可從 OLT 中查出目前有“合作關係”的 Web 代理伺服器之中那一個擁有此一文件。所謂的合作關係，是以人工在 Web 代理伺服器的設定檔中設定的，方式類似設定 squid 的 parent 和 sibling。使用 OLT 技術的快取階層架構如圖一所示。

物件位置表和 ICP query 是否能並存？答案是肯定的，因為使用物件位置表雖然減少向其他 Web 代理伺服器查詢的時間，但萬一物件位置表的資料“有錯”時，仍要依靠原先的階層式快取架構來取得資料。



圖一、使用物件位置表的快取階層架構

接下來的內容會敘述當一個使用物件位置表技術的 Web 代理伺服器遇到 cache miss 時，如何正確地取得資料的步驟。有一用戶端向 Web 代理伺服器 P 要求擷取一個 Web 文件 D，在 P 本身的硬碟找不到：

- (1) 查詢 P 本身的物件位置表，如果物件位置表沒有關於此一 Web 文件的資料，就直接以 ICP 的方式來處理。
- (2) 如果查到 Web 代理伺服器 PX 擁有 Web 文件 D，則以 P 向 PX 要求擷取 D。但在同時，向其他有合作關係的每一個 Web 代理伺服器發出 ICP query。
- (3) 如果 PX 在兩秒內開始傳回 D，P 就等 D 傳完，然後再傳給用戶端。同時刪除 D 在物件位置表中的記錄；如果 P 設定成不保留由 PX 傳過來的 Web 文件，則不用刪除。
- (4) 假設 PX 在兩秒（這個等待時間可以視需要來改）內沒有回應或說它沒有 D，則 P 還是以 ICP 的方式來處理。

物件位置表的更新分為主要和次要兩種。每個支援物件位置表的 Web 代理伺服器都必須通知有合作關係的 Web 代理伺服器自己新增了那些 Web 文件，同時也接收它們的通知。

物件位置表的主要更新是指，Web 代

理伺服器要找出從上次主要更新到目前這段時間內存到自己本身硬碟的 Web 文件，並且把它們的 URL 列表，然後通知其他有合作關係的 Web 代理伺服器。由於主要更新很耗運算時間和網路頻寬，所以建議每天做一次。至於次要更新基本上很類似主要更新，只是規模較小，一至兩個小時做一次。

和原先的 Web 代理伺服器架構相比，新架構最主的優點是可以節省 cache miss 之時，用 ICP 向其他 Web 代理伺服器查詢的時間，因為查詢物件位置表就可以知道附近那一個 Web 代理伺服器有我們想要的 Web 文件，然後就直接向它要。而且 Document Index 的數目可以很多，因此涵蓋的範圍可以很大。

目前實驗所得到的結果是，一個使用者多的 Web 代理伺服器 P 和一個使用者少的 Web 代理伺服器 Q ，由 Q 以物件位置表來得到 P 的援助，可以讓 Q 的效能獲得明顯的改善。

四、結果與討論

本計劃目前主要的研究方法是，利用 Squid 1.1.x 的 access log file，進行模擬與分析。所使用的 access log files 是分別從交大計中 Web 代理伺服器和交大資工 Web 代理伺服器取得。目前拿來做實驗的資料的如表一和表二所示。

CC	1997/11/2	1997/11/3	1997/11/4	1997/11/5	平均
Count	902331	893996	1000654	924280	930315
Bytes	12.65G	12.22G	14.08G	13.64G	13.15G

表一、交大計中代理伺服器 access log file

CSIE	1997/11/2	1997/11/3	1997/11/4	1997/11/5	平均
Count	9947	23144	38293	39909	27823
Bytes	211.0M	473.5M	708.3M	821.5M	553.6M

表二、交大資工代理伺服器 access log file

我們檢驗兩種 hit rate。Counts hit rate

是 Web 代理伺服器傳回給用戶端的 Web 文件的數目除以所有 Web 文件的總數；而 Bytes hit rate 是 Web 代理伺服器傳回給用戶端的 Web 文件的檔案大小總和除以所有 Web 文件檔案大小的總和。

當虛擬 Web 代理伺服器的硬碟空間不夠，需要選出一些 Web 文件將之從硬碟中刪除時，目前使用的演算法是一種近似的 LRU algorithm——從每 32 個 Web 文件之中找出 access time 離現在最久的一個將之刪除，直到硬碟空間足夠為止。

我們首先試驗快取硬碟容量對 hit rate 的影響，這個實驗我們分別對交大計中和交大資工的 log files 各做一組實驗。每組的實驗各測試六種硬碟容量，分別是無限大、每天的平均流量，以及每天的平均流量的 1/2、1/4、1/8、1/16。

CC	Infinite	1.331G	6656M	3328M	1664M	832.0M
Counts hit rate	72.63%	69.30%	60.78%	50.83%	41.97%	33.92%
Bytes hit rate	54.49%	51.25%	44.29%	36.71%	30.08%	23.99%

表三、快取硬碟容量之影響（交大計中組）

CSIE	Infinite	550.0M	275.0M	137.5M	68.75M	34.38M
Counts hit rate	62.06%	55.76%	46.28%	38.57%	32.31%	26.35%
Bytes hit rate	53.90%	49.47%	40.88%	32.28%	24.75%	18.01%

表四、快取硬碟容量之影響（交大資工組）

兩組實驗的結果，得到的 hit rate 比一般實際運作的 Web 代理伺服器的 hit rate 要來得好。原因有下列幾個：第一、當天兩個 Web 代理伺服器的 hit rate 本來就很高。第二、我們的虛擬 Web 代理伺服器還無法模擬真正 Web 代理伺服器在磁碟存取上的細節。有一點值得觀察的是，Object bytes hit rate 都比 Object counts hit rate 來低，這顯示在這幾天小檔案的使用率比大檔案來得高。

在實驗二中，我們想觀察兩個 Web 代理伺服器合併，會不會讓 hit rate 提高。作法是將同一天的兩個 Web 代理伺服器的 access log files 合併起來，按存取時間來排序。之後就照實驗一的方法來做，快取容量的設定也一樣。

CC & CSIE	Infinite	1.386 G	6931 M	3466 M	1733 M	866.4 M
Counts	72.72	69.19	60.76	51.11	42.40	34.56
hit rate	%	%	%	%	%	%
Bytes	54.97	51.50	44.73	37.55	31.15	25.40
hit rate	%	%	%	%	%	%

表五、兩個 Web 代理伺服器合併

表五的數字和表三相去不遠，表示交大計中和交大資工這兩個 Web 代理伺服器如果合併在一起，對原先計中 Web 代理伺服器的使用者並沒有提高多少 hit rate，沒什麼好處，當然資工 Web 代理伺服器的使用者得利不少。這是因為交大計中 Web 代理伺服器的使用者人數遠遠多於交大資工 Web 代理伺服器。

實驗三是測試 OLT 的效果。我們假設的情形是這樣的，因為交大計中 Web 代理伺服器的使用者人數較多，所以計中 Web 代理伺服器會去更新資工 Web 代理伺服器的物件位置表。換言之，在測量 hit rate 之前，資工 Web 代理伺服器的物件位置表中有記錄計中 Web 代理伺服器含有那些 Web 文件。至於 Cache 的容量和實驗一的交大資工組完全一樣。

由表六的結果顯示，在不增加快取硬碟容量的情形下，和表四的結果比起來，兩種 hit rate 都有顯著的改善。雖然這麼做對使用者多的 Web 代理伺服器並沒有好處，但是對使用者人數少的 Web 代理伺服器卻能不增加成本，不用拉攏其他使用者的情形下，讓 hit rate 明顯提高。

CSIE with GRI	Infinite	550.0 M	275.0 M	137.5 M	68.75 M	34.38 M
Counts	76.71	75.33	71.30	66.55	62.59	58.83

hit rate	%	%	%	%	%	%
Bytes hit rate	61.95%	60.87%	58.01%	55.53%	53.63%	51.44%

表六、使用物件位置表之結果

五、計劃成果自評

從上面的幾個實驗，能不能顯示物件位置表技術真的有幫助？初步的結果是肯定的，不過要評估到底實不實用，還要取得更多的 Web 代理伺服器的 access logs 來進行模擬。而且不能只有兩台 Web 代理伺服器的 log，應該要三台以上才行。同時程式也還必須改寫，考慮更真實的狀況，如物件位置表每個小時之間的次要更新等。

未來的工做有幾個困難點，第一是 Web 代理伺服器的 log 取得不易。第二點是，我們的模擬能做到什麼程度，例如 Web 代理伺服器之間的網路延遲的因素要如何考慮。第三，雖然物件位置表能使查詢的範圍變大，但還是要網路通暢才会有好的效果。

六、參考文獻

- [1] Michael Accetta. RFC 887: Resource location protocol, December 1983.
- [2] Guy T. Almes, Andrew P. Black, Edward D. Lazowska, and Jerre D. Noe. The Eden system: A technical review. IEEE Transactions on Software Engineering, SE-11(1):43--59, January 1985.
- [3] Rafael Alonso and Matthew Blaze. Long-term caching strategies for very large distributed file systems. Proceedings of the USENIX Summer Conference, pages 3--16, June 1991.
- [4] David R. Boggs. Internet broadcasting, January 1982. Ph.D. Thesis, available as Technical Report CSL-83-3, Xerox Palo Alto Research Center, October 1983.
- [5] Sun Wu and Chang-Chain Liao, "Virtual Proxy Servers for WWW and Intelligent Agents on the Internet", 30th Hawaii International Conference on System Science, January 1997.