

## Internet 內容遞送之演進與評估

測試計畫主持人：林盈達

參與測試人員：田慶明、林柏青

Internet 內容提供者採用傳統中央網站式的內容遞送，會有骨幹頻寬浪費及使用者下載內容時間過長等問題，且採用 Web 複製式的內容遞送，將伺服器分散建置於各家服務提供者，則其營運費用較高。而在新興的內容遞送網路（Content Delivery Network；CDN）中，內容提供者只需將內容交由一家 CDN 服務提供者來散布，省去與多家 ISP 或 HSP 洽商的麻煩，並且 CDN 的資源可以由多家內容提供者來共享，所以可以降低各家內容提供者的營運成本。本文介紹 Internet 內容遞送之演進、CDN 的商業模式以及效能評估。在使用 Inktomi CDN 產品的評估數據中，發現採用內容遞送網路的營運成本會比 Web 複製式的內容遞送減少數成，在散布內容時可以比傳統 FTP 方式減少一半以上的骨幹頻寬消耗，而內容散布時間是 FTP 的數十甚至上百分之一，另外，使用者下載內容的時間是中央網站式的十五分之一，而且 CDN 可以使串流媒體避開網路擁塞的節點，進而提升遞送的服務品質。我們認為內容提供者的內容如果大多為靜態即非動態由原始伺服器產生，或其使用者散布在多家 ISP 網路中，就比較適合轉換到內容遞送網路。

### Internet 內容遞送之演進

Internet 上的內容由伺服器端遞送到終端使用者端，是由許多人彼此合作而達成，如圖一所示。在開始探討內容傳遞網路的演進之前，應先釐清內容傳遞網路的組成分子。以下介紹這些人在內容遞送中所扮演的角色，及其為了要加速內容遞送所採用的一些機制。

Internet 內容提供者（Internet Content Provider；ICP）

ICP 主要的任務就是產生內容，並且透過各種方式將內容遞送給終端使用者。ICP 可選擇自行架設伺服器，或採用 HSP 所提供的網站代管服務。

代管服務提供者（Hosting Service Provider；HSP）

HSP 的網路資料中心，提供 ICP 網站或伺服器主機代管的服務。

骨幹運輸者（Backbone Carrier）

骨幹運輸者提供給各個 ICP 長距離、大範圍的網路傳輸服務，一般來說骨幹運輸者都是透過光纖網路或衛星網路來傳送資料。

Internet 服務提供者（Internet Service Provider；ISP）

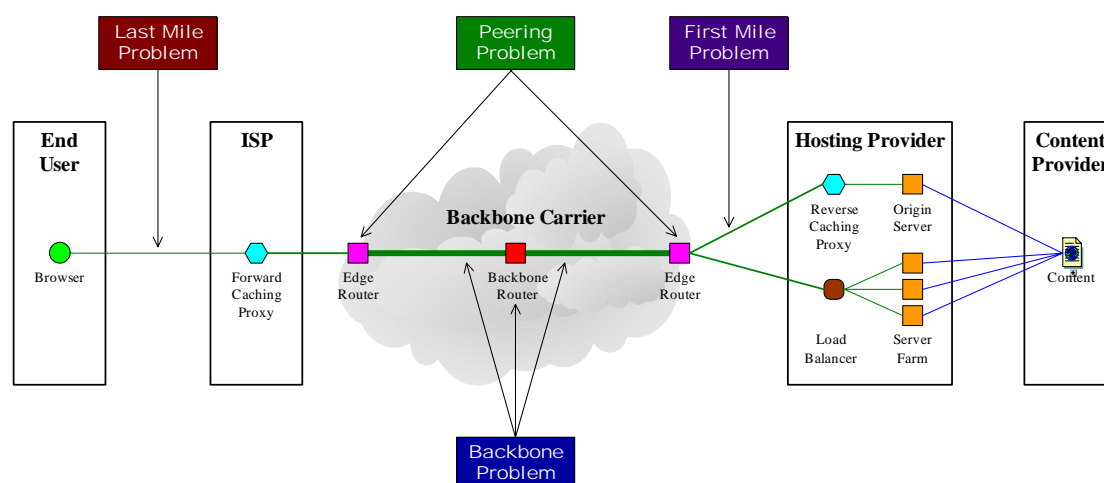
ISP 提供給一般終端使用者連線到 Internet 的服務，有些 ISP 同時也是 HSP 及骨幹運輸者。

終端使用者（End User）

終端使用者也就是 Internet 內容的消費者，其利用瀏覽器（Browser）來瀏覽 Internet 上的內容。

內容遞送網路服務業者（CDN service provider）

傳統上 ICP 是面對多家的 HSP 或 ISP 來遞送內容至終端使用者，而 CDN 服務業者則是一個介於中間的協力廠商（3rd Party），ICP 可直接把內容交給 CDN 服務業者來散布，如此可使 ICP 免於面對多家 HSP 及 ISP。



圖一：參與 Internet 內容遞送的角色及遭遇的瓶頸問題

## 內容遞送的四大瓶頸

整個 Internet 架構是由許多獨立的網路所組成，各個網路之間彼此透過路由器（Router）或是閘道器（Gateway）相互連接。就 ICP 而言，在 Internet 上進行內容遞送會橫跨數個網路，而所遭遇到的瓶頸主要有四個地方，分別為第一哩（first mile）、對等點（peering point）、骨幹（backbone）以及最後一哩（last mile），以下將針對這四個瓶頸逐一說明。

### 一、第一哩問題

所謂「第一哩（first mile）」指的是介於伺服器端及路由器端（edge router）的網路連線。ICP 或是 HSP 架設伺服器時，最容易遇到的就是第一哩的問題，因每個終端使用者所發出的要求大多會經過這一段連線，而隨著終端使用者數量的成長，ICP 必須花費更多資金在伺服器及網路頻寬的建置上，當然同樣的問題也會發生在 ISP 上。為解決伺服器容量不足的問題，ICP 或 HSP 可以採用伺服器群（server farm）的方式，利用負載平衡器（load balancer）將大量連線要求（request）導往後端的多台伺服器上，以達到交換及平衡負載的目的，但是這種作法會造成終端使用者的連線，仍然會被負載平衡器的網路連線頻寬所限制，為解決此一問題，ICP 或 HSP 可以在靠近伺服器端架設反向快取代理伺服器（reverse caching proxy），將某個網站較常被存取的內容儲存在代理伺服器，當收到終端使用者的要求時，代理伺服器便直接把內容回傳給終端使用者，以減少連線到伺服器所需的頻寬消耗及伺服器的負載。總而言之，第一哩的問題主要是在於伺服器端的延展性（scalability）不夠。

### 二、對等問題

所謂對等點指的是兩個獨立網路相互連接的端點，通常對等問題大多發生在商業上，而並不是效能上的問題。試想若有兩家 ISP 投資大量的資金在建構網路上，

第一家 ISP 可能會因為商業上的考量，而不願意幫另一家 ISP 轉送 (forward) 他的網路封包 (packet)，造成第二家 ISP 的封包必須改走其他可能頻寬較小，或是延遲時間 (latency) 較長的 ISP 網路路徑。而這些對等問題會拖慢整個內容傳送的速度。對等問題 (peering problem) 的解決方式，是由 ISP 彼此簽定對等協議 (peering agreement)，利用對方的路由器轉送封包。

### 三、骨幹問題

Internet 骨幹是由數個大型長途運送網路 (long-haul network) 所組成，而這些骨幹的容量就是骨幹問題 (backbone problem)。現今 Internet 的流量愈來愈大，而骨幹的容量也必須要能隨著流量而成長。通常網路的容量是由纜線及路由器來決定，所以想要解決骨幹容量問題，除提升纜線及路由器之容量外，另一個方法就是減少在骨幹上的傳輸流量。作法是各 ISP 在靠近終端使用者端架設正向快取代理伺服器 (forward caching proxy)，使其儲存各種原始伺服器 (origin server) 上被下載過的內容，讓有相同內容的需求能直接由正向快取代理伺服器來提供服務，如此便可減少終端使用者直接連線到伺服器的機會，進而減少在骨幹上的流量，並且可以減少內容下載的時間及伺服器上的負載。

### 四、最後一哩問題

所謂的「最後一哩 (last mile)」指的是介於終端使用者及 ISP 之間的網路連線。終端使用者利用數據機上網，其速度幾乎都被限制在 56Kbps 之內，若是瀏覽到圖片或媒體較多的網頁，網頁物件的下載速度很容易被數據機所限制住。目前解決的方式是終端使用者利用 Cable Modem 或是 DSL Modem 等連線速度比較快的寬頻裝置來上網。

以上所述瓶頸問題雖然有一些解決方式，但似乎仍存在一些不易解決的問題。伺服器群及反向快取雖然可以解決伺服器容量不足的問題，卻無法解決網路擁塞。正向快取雖然能減少網路頻寬的消耗，但也帶來新的問題，例如在快取代理伺服器中，快取的物件無法與原始伺服器上內容同步更新，當伺服器上的內容一有變動，正向快取仍有可能會提供終端使用者舊的內容，另外快取代理伺服器的擊中率 (hit rate) 並不是很高，當快取伺服器發生 cache miss 時，仍然要連線到原始伺服器取回所需要的內容。另外一種解決伺服器容量不足及減少網路連線延遲時間的方法，就是 ICP 直接在靠近終端使用者的 ISP 網路端架設伺服器，然後利用內容複製 (replication) 的機制，主動將內容複製到各個 replication server 上。但要使各個 replication server 內容與原始伺服器同步，便是一件不容易的事，萬一有某台 replication server 出了問題無法同步，那麼連上該伺服器的終端使用者所取得的內容便是有問題的。另外要同時管理眾多 replication server 也是一件麻煩的事。最後營運成本是最大問題，ICP 必須在擁有不同數量客戶的 ISP 端放置不等量的 edge server，每個月都必須付給各 HSP 或 ISP 大量的月租金。

基於以上仍然未解決的問題，一種新的內容遞送方式便被產生出來，也就是「內容遞送網路 (Content Delivery Network; CDN)」。CDN 是由 CDN 服務業者建置 edge server 於靠近用戶的地方，通常是眾多 ISP 及 HSP 端，也有可能是在各個

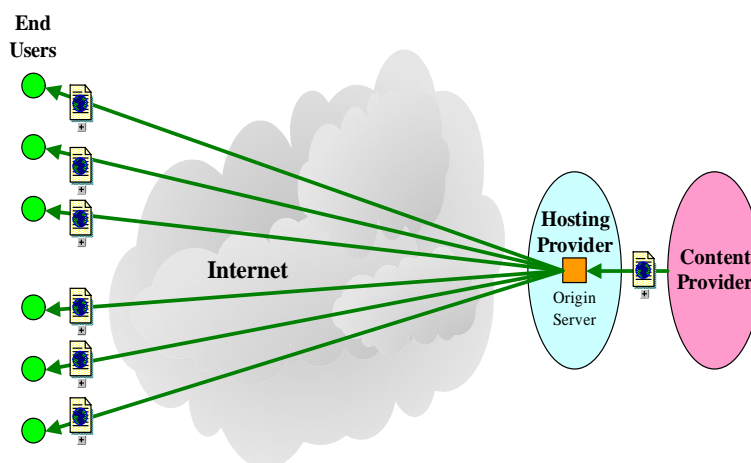
大樓、公共場所、有線電視頭端等，並租用骨幹運輸者的線路來進行長距離的資料傳輸。另外 HSP、骨幹運輸者及 ISP 彼此也可聯盟成為一個 CDN。其想法主要有兩個，一個就是由 CDN 服務提供者為各個 ICP，主動地將其原始伺服器上的內容事先傳送到 CDN 服務業者的 edge server 上，終端使用者的要求直接就由 edge server 來服務，因為只要先傳送一次內容，便可提供成千上萬個在 edge 端的終端使用者存取，所以可大量減少原始伺服器的負載，以及網路骨幹頻寬的消耗，而由於終端使用者非常靠近 edge server，可以減少相當多的網路延遲時間。CDN 的另一個想法就是，由要求路由系統（request-routing system）將終端使用者所發出的要求，重新導向到最靠近終端使用者或負載最輕的 edge server 上，如此終端使用者便可以很迅速地取回內容，而且各個 edge server 之間的負載也可以平衡。所以 CDN 可有效地解決傳統內容遞送所無法解決的問題，同時也讓 ICP 免於面對多家 ISP 或 HSP 的煩惱，只需將內容交由一家 CDN 服務業者來傳送。

### 內容遞送的三類型

內容遞送依其演進主要可分成三個類型，分別為中央網站式（centralized hosting）、Web 複製式（web replication）以及內容遞送網路，以下將介紹這三種內容遞送的類型，並做一個比較。

#### 一、中央網站式內容遞送架構

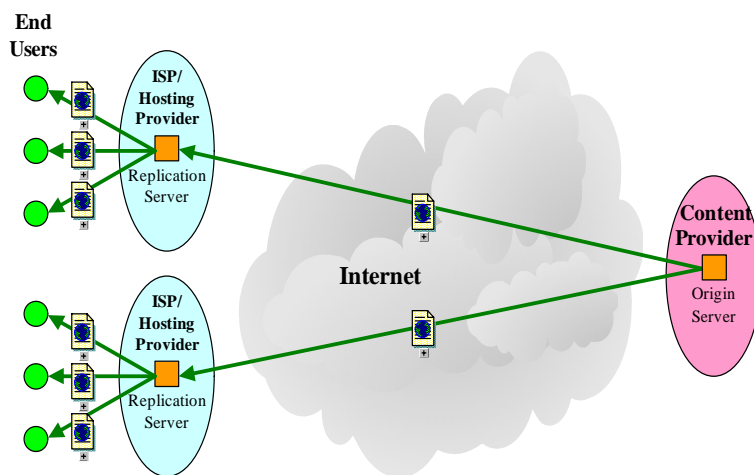
中央網站式的內容遞送架構如圖二所示，參與內容遞送的角色主要有 ICP、HSP 以及終端使用者。在這個架構下，ICP 可自行架設伺服器來服務終端使用者，或利用 HSP 所提供的網站或主機代管服務來把內容遞送給終端使用者。



圖二：中央網站式內容遞送

#### 二、Web 複製式內容遞送架構

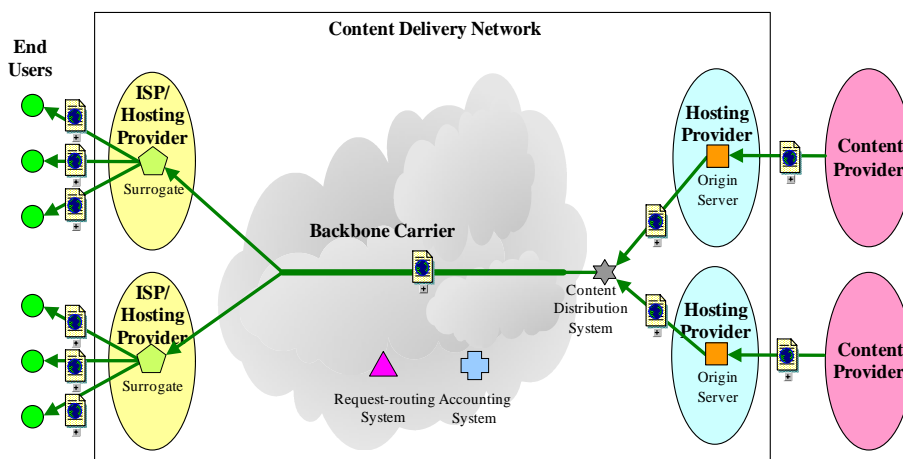
Web 複製式的內容遞送架構如圖三所示，參與內容遞送的角色主要有 ICP、HSP、ISP 以及終端使用者。在這個架構下，ICP 利用多個 HSP 或 ISP 所提供的主機代管或是主機共置服務，橫跨數個網路在各個 edge 端架設伺服器，然後再利用 replication 的技術，將所製作好的內容同步複製到各台 replication server 上，以便就近提供內容給各 ISP 端的用戶。而終端使用者首先連線到原始伺服器，再從所顯示的網頁中手動選取一個最靠近終端使用者的 replication server。



圖三：Web 複製式內容遞送

### 三、內容遞送網路式的架構

內容遞送網路的架構如圖四所示，主要參與內容遞送的角色有 ICP、CDN 服務業者、HSP、骨幹運輸者、ISP 及終端使用者。CDN 服務業者取代各個 ICP 付費給 HSP，在他們的機房中建置 edge server (在圖四中為 surrogate)。另外 CDN 服務業者租用骨幹運輸者的連線，將各個 ICP 客戶的內容散布到各個 ISP 或 HSP 端的 edge server 上。另外 CDN 透過要求路由系統 (request-routing system) 把終端使用者所發出的要求，重新導向到最靠近或負載最輕的 edge server，可減少內容下載的時間，並且各個 edge server 的負載也可得到平衡。在此要釐清散布 (distribution) 及遞送 (delivery) 的差別，所謂的 distribution 是指內容從原始伺服器傳送到 edge server 上的過程；而 delivery 則指內容從 edge server 傳送到終端使用者的過程。



圖四：內容遞送網路的架構

### 內容遞送方式之比較

在了解中央網站式、Web 複製式及 CDN 這三種內容遞送方式後，列舉表一來比較這三者的差異及優缺點。

表一：Centralized hosting、web replication 及 CDN 三者的比較

Item	Centralized Hosting	Web Replication	CDN
Investment cost	server, bandwidth	server, bandwidth	distribution service

Server group	dedicated	dedicated	shared
Server load	centralized	distributed but unbalanced	distributed and globally balanced
End users connect to	origin server	replication server	best edge server
Server proximity	far	Near but not true edge	Near and true edge
User-perceived latency	long	shorter	shortest
Content dispersibility	single point	multiple points	globally wide
Content availability	low	low	high
Backbone bandwidth consumption	high	lower	lowest

首先是 ICP 所要投資的費用，採用中央網站式及 Web 複製式，均要花費資金在建置伺服器軟硬體及網路頻寬上，並且隨著終端使用者數量的成長，而要花費更多資金來提升伺服器容量及網路頻寬。如果在某個時間內發生非常熱門的事件（例如美國 911 恐怖攻擊事件），ICP 為了要應付突如其來的擁塞（flash crowd），則必須在事先大幅提升伺服器及頻寬的容量，藉以應付大量的終端使用者連線擠入網站。若是 ICP 採用 CDN 則只需付費給 CDN 服務業者，將其內容透過 CDN 散布出去。另外中央網站式及 Web 複製式的伺服器均是 ICP 專用的（dedicated），平常傳輸量很少時便浪費掉了，而 CDN 的 edge server 是可由多個 ICP 共享（shared）並且散布在各地，比較不容易產生擁塞的問題。若一個 CDN 服務業者在一個 ISP 端租 50 台機器來裝載 10 家 ICP 的內容，效能可能等同於這 10 家 ICP 各租 10 台機器（即共 100 台），這是統計多工（statistical multiplexing）的效果。另外由於 CDN 會將終端使用者的要求，導向到最好的 edge server 上，所以各 edge server 上的負載會比較輕，且能夠做到全球化的負載平衡（global load balancing）。

在終端使用者的連線方面，由於一般的終端使用者大多會在自己的瀏覽器上使用快取伺服器，或是 ISP 利用透明化的代理伺服器（transparent proxy），將終端使用者的連線重新導向到快取代理伺服器上，如此讓終端使用者所發出的要求首先會連到快取代理伺服器，然後再連到真正的伺服器。在中央網站式中，終端使用者所發出的要求還是會連到原始伺服器，所以終端使用者所感受到的內容下載時間比較長。在 Web 複製式中，終端使用者所發出的要求是先連到快取代理伺服器，再連到在 edge 端的 replication server 上，所以時間可以改善一些。

一般的 ICP 會想要藉由分析自己伺服器上的存取記錄檔，來了解終端使用者存取內容的狀況，做為財務上的統計。而中央網站式及 Web 複製式的原始伺服器和 replication server，並非是離終端使用者最近的伺服器，真正離終端使用者最近的是快取代理伺服器，如果快取代理伺服器在 cache hit 的狀況下，會使得 ICP 的伺服器無法真正記錄到終端使用者的連線。但在 CDN 的架構下，CDN 的 edge server 就是快取代理伺服器，所以終端使用者下載內容的時間可大幅改善，ICP 也可很容易地取得在快取代理伺服器上的存取記錄檔。

在中央網站式的架構中，終端使用者的要求會連線到原始伺服器，途中可能會經過骨幹網路，所以在骨幹網路上的頻寬消耗無法避免；而在 Web 複製式的架構下，終端使用者要是選了不好的 replication server，則也有可能經過骨幹網路。

但是在 CDN 的架構下，ICP 的內容已經被散布到 edge server 上，再加上終端使用者的要求會被重新導向到最好的 edge server，所以可避免終端使用者的連線經過骨幹網路，進而節省骨幹網路頻寬的消耗。

### 內容遞送網路之組成要素

整個 CDN 主要是由四個要素所組成，如圖四分別為：要求路由系統（request-routing system）、內容散布系統（content distribution system）、記帳系統（accounting System）以及代理伺服器（surrogate），以下簡介這四個組成要素。

#### （1）要求路由系統

Request-routing system 主要作用是將終端使用者所發出的要求，重新導向到最靠近終端使用者端或負載最輕的 edge server 上，也就是將要求交由最「好」的一台 edge server 來服務。通常要判斷哪一台 edge server 是最好的，可依據網路拓樸接近度（topological proximity）、內容有效度（content availability）、伺服器負載量（server load）以及自訂的政策（customized policies）來決定。另外 request-routing system 重新導向的方式主要有 DNS（domain name service）重新導向、傳輸層（transport layer）重新導向及應用層（application layer）重新導向等三種。

#### （2）內容散布系統

內容散布系統主要的作用，是把在原始伺服器上的內容散布到 edge server，而散布的方式主要有兩種，第一種是由伺服器端主動把內容散佈（server side push）到 edge server 上；第二種是由 edge server 定期主動詢問原始伺服器是否有內容要散布，若有則取回內容（cache based pull）。值得一提的是，傳統快取代理伺服器是被動式取回內容，也就是若發生 cache miss 的情況，快取代理伺服器才會連線到原始伺服器取回內容。

#### （3）記帳系統

記帳系統會分析在各個 edge server 上的存取記錄檔，讓 CDN 服務業者能根據終端使用者下載 ICP 內容的量來向 ICP 收費。

#### （4）代理伺服器

所謂的代理伺服器也就是本文中所提到的 edge server，其實就是一台快取代理伺服器，常見的快取代理伺服器種類有 forward explicit caching proxy、forward transparent caching proxy 及代理伺服器。代理伺服器的作用是接收終端使用者所發出的要求，並將正確的內容傳回給終端使用者，它的動作行為就像是原始伺服器一樣。

### 內容遞送網路的商業模式

CDN 的商業模式主要有三種，第一種是根據 ICP 的需求來看，可稱之為「以內容為中心的內容遞送網路（Content-centric CDN）」；第二種則是根據 ISP 的需求來看，稱之為「以存取為中心的內容遞送網路（Access-centric CDN）」；第三種則是 HSP、骨幹運輸者及 ISP 所組成的內容同盟（Content Bridge and Content Alliance），以下便探討在這三種商業模式中的付費及內容遞送方式。表二為各

CDN 服務業者及相關廠商的分類。

表二：內容遞送網路服務提供者及相關廠商的分類

Type	Service Providers
Content-centric	Adero, Akamai, AT&T, Cidera, Digital Island, Edgix, epicRealm, Globix, iBeam, MadgeWeb, Mirror Image, Orblynx, SkyServ, Speedera, Teleglobe
Access-centric	AT&T, Cidera, Digital Island, Edgix, Globix, iBeam, MadgeWeb, Mirror Image, Orblynx, SkyServ, Teleglobe
Content Bridge	Adero, AOL, Digital Island, Exodus, Genuity, MadgeWeb, Inktomi, MadgeWeb, Mirror Image, Nextra
Content Alliance	AOL, AT&T, Cidera, Cisco, Digital Island, epicRealm, Mirror Image, Speedera, Teleglobe

### 以內容為中心的 CDN 商業模式

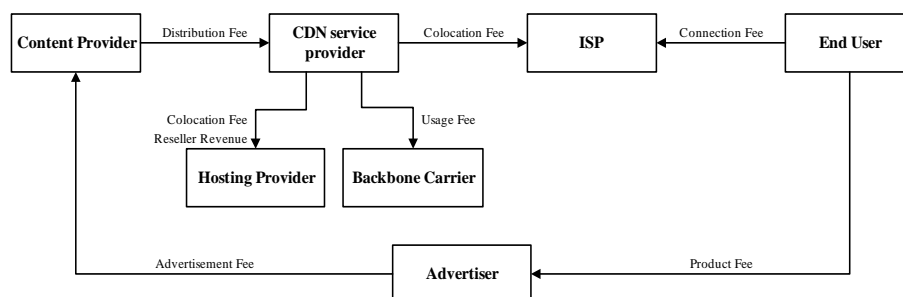
以內容為中心的 CDN 價值鏈如圖五所示，這種模式是比較適合較少商業價值的內容 (valueless content)。ICP 付費給 CDN 服務業者來散布他們的內容，主要目的是要加速將內容遞送給終端使用者，以避免終端使用者因存取這家 ICP 的內容速度過慢而轉向其他的 ICP。

CDN 服務業者付費給 HSP 骨幹運輸者及 ISP，以建置 edge server 及 storage server 在他們的機房中，大部分所付出的費用是在租借場地及網路連線上。有的時候 HSP 會幫 CDN 服務業者轉售內容散布的服務給 ICP，CDN 服務業者便會將部分盈收與 HSP 共享。

另外廣告商 (Advertiser) 付費給 ICP，希望能夠在 ICP 所製作出來的內容中加入廣告，而終端使用者在看了廣告之後，就有可能會付費購買所廣告的商品。

在以內容為中心的商業模式下，骨幹運輸者及 ISP 的確把內容遞送給終端使用者，但他們並沒有從內容遞送中得到任何盈收，因為提供主機代管及共置服務 (collocation) 有可能是他們傳統的業務。另外有的 CDN 服務業者想了解有多少數量的終端使用者在使用 CDN 服務，所以 CDN 服務業者必須要收集在 edge server 上的存取記錄資料，但一般的終端使用者可能會使用 ISP 的快取代理伺服器，造成 CDN 所遞送的內容可能會被 ISP 的快取代理伺服器所 cache，因此 CDN 服務業者會另外付費給 ISP，購買在快取代理伺服器上的存取記錄資訊。

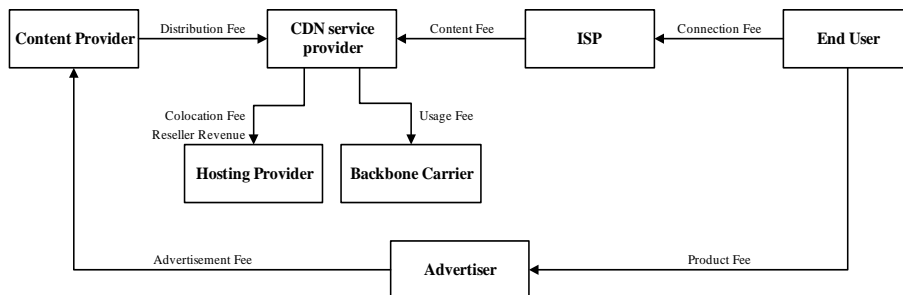
另外有些 CDN 服務業者會 OEM 內容遞送網路服務給 ISP 或骨幹運輸者，或是利用本身的技術幫他們建置私人的 CDN。因為有些跨國性的企業會想要建置私人的 CDN，以方便企業進行內部通訊、教育訓練、產品發行或視訊會議等，所以這些企業找上 ISP，希望 ISP 能提供私人 CDN 的服務。



圖五：以內容為中心的內容遞送網路價值鏈

### 以存取為中心的 CDN 商業模式

以存取為中心的 CDN 價值鏈如圖六所示，這種模式是比較適合有價的內容（valuable content）。ISP 付費給 CDN 服務業者，購買在 CDN 服務業者的 edge server 中較受歡迎的內容，如此 ISP 便可就近提供受歡迎的內容給 ISP 的客戶，而使得 ISP 客戶的連線不必離開 ISP 的網路到原始伺服器上取得內容，如此可減少 ISP 對外的網路頻寬（upstream bandwidth）消耗，進而減少 ISP 投資在擴充網路頻寬上的費用。另外由於 ISP 的客戶能很快地存取到所想要的內容，因此會覺得該家 ISP 的服務品質不錯，所以就長期繼續使用該 ISP 的連線服務，對 ISP 來說就可避免他的客戶轉移到其他 ISP 的情況發生。



圖六：以存取為中心的內容遞送網路價值鏈

### 內容同盟

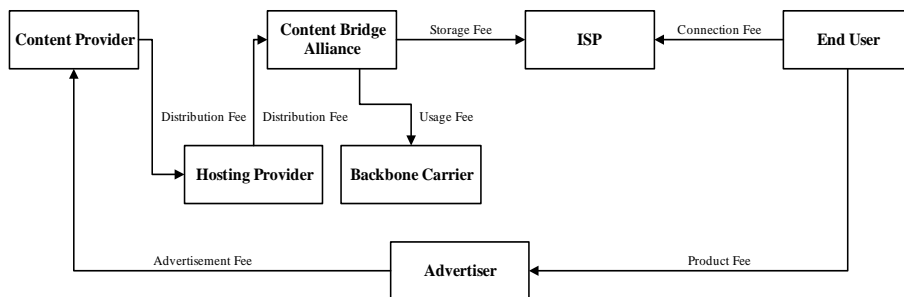
由於 CDN 市場的興起，HSP、骨幹運輸者及 ISP 想要分食這塊 CDN 的大餅，所以就有一種新的商業模式產生，這種商業模式是各家業者共同加入一個同盟，彼此合作提供自己的服務，來組成一個非常大的內容遞送網路。傳統的 CDN 模式大多是由某家 CDN 服務業者自行建構網路架構，但其 CDN 的延展性有限，無法一直無限制地擴充其規模，因為有可能某些 HSP 或 ISP，因本身的網路服務架構或軟硬體設備，無法配合 CDN 服務業者來建置其 edge 伺服器。而內容同盟是採用加盟的方式來營運，並制訂一個開放式的標準，讓各家業者的網路及軟硬設備彼此之間都能配合運作，所以其 CDN 的延展性便會比前面兩種的 CDN 模式來得好。

目前 CDN 市場上主要有兩個同盟，一個由 Inktomi 所提出的 Content Bridge；另一個則是由 Cisco 所提出的 Content Alliance。Content Bridge 主要是提供跨網路的內容遞送服務，並且將所獲得的盈收和參與遞送的業者共享，而 Content Alliance 則著重在發展 CDN 內容對等的標準。至於一般業者可以根據其本身現存網路軟體設備的種類，來選擇加入哪一個同盟，例如某家 ISP 的快取代理伺服器是採用 Inktomi 的產品，則可選擇加入 Content Bridge，當然也有些業者同時加入兩個同盟。

內容同盟的價值鏈如圖七所示，在這個模式下 HSP、骨幹運輸者及 ISP，可提供超過自身網路範圍的內容散布服務，然後便可根據這個服務來獲得盈收，而且這種模式可使 ICP 的能見度（visibility）擴展到在 ISP 端的 edge server 上，並能進

行有效的控制。內容同盟主要是依靠 HSP 來販售內容散布服務給 ICP，因為 HSP 原本主要的客戶就是 ICP，且內容同盟依靠骨幹運輸者，來將內容散布到各家 ISP 的 edge server 上，而 ISP 則負責提供 edge server 來存放所散布的內容。內容同盟的營運者（operator）如 Digital Island，他的角色是在提供技術上的支援、內容遞送的處理，以及統計在各個 edge server 上的存取記錄來向 ICP 收費。

內容同盟的商業模式很像目前的信用卡模式，銀行發行信用卡就等同於 ICP 發行內容，消費者在商店刷卡消費就等同於終端使用者從 edge server 上取得內容，而內容同盟的角色就類似 Visa，本身只負責處理信用卡的交易，而 Visa 會根據終端使用者的交易記錄來向發卡銀行扣款。



圖七：內容同盟的價值鏈

## 內容遞送網路的評估

在了解內容遞送網路的運作方式及其效益之後，以下將實際評估內容遞送網路的效能，利用一些實際的數據來證明，內容遞送網路的確優於傳統的內容遞送方式。在評估內容遞送網路方面，我們從 ICP 的觀點來比較，Web 複製式及 CDN 的內容遞送所需花費的建置及營運費用，雖然各家 CDN 服務業者收費差異很大，但長期來看，相信前面所說的 CDN 統計多工的效果，將可為 ICP 降低營運成本。另外對 ICP 而言，用較短的時間及耗用較少的骨幹頻寬，將內容散布到 edge server 出去是比較有效率的內容散布方式，所以我們評估了內容被散布時，骨幹頻寬的消耗及散布的時間。此外也量測了終端使用者所感受到的內容下載時間。在所測試的內容方面，我們分別測試了靜態內容（static content）及隨選串流媒體（on-demand streaming media），因為目前在 Internet 這兩種型態的內容占大多數。至於動態產生的內容（dynamic generated content）與即時串流（live streaming）的測試方式較為複雜，故沒有列入這次的測試。

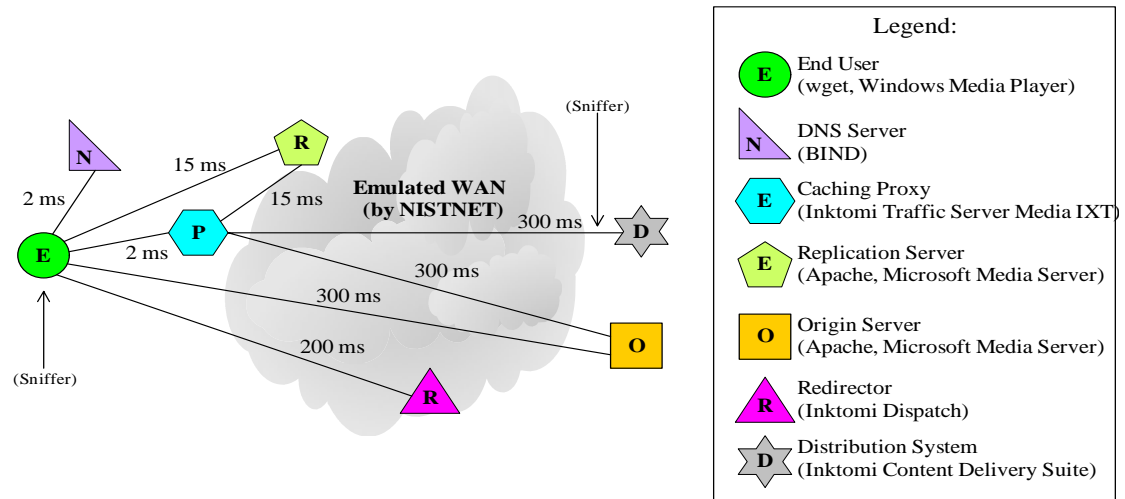
表三為本次測試活動所邀請參與測試內容遞送網路產品的廠商。Inktomi 非常積極地提供產品給我們進行測試；而其他廠商如 Cisco 無法派出工程師來設置測試平台，因此退出測試。Network Appliance 的代理商沒有多餘的機器可供借測。至於 Cache Flow 及 F5 因新產品的發布時間無法趕上測試日期，所以選擇不參加。因此本次的測試活動以 Inktomi 的產品為代表。

表三：內容遞送網路產品

Vendor	Content Distribution System	Surrogate	Request-routing System
Inktomi	Content Delivery Suite	Traffic Server, Media IXT	Dispatch
Cisco	Content Distribution Manager	Content Engine	Content Distribution Manager
Network Appliance	Content Director	NetCache	Global Request Manager
Cache Flow	ciQ Director	End user Accelerator, Server Accelerator	
F5	GLOBAL-SITE Controller	EDGE-FX Cache	3DNS Controller

測試平台、工具以及模擬的網路延遲時間如圖八所示，各 PC 的作業系統軟體採用 Microsoft Windows 2000 Advanced Server 及 RedHat Linux 7.0，硬體則採用 Pentium III 1G Hz CPU、512MB memory、20 GB IDE hard disk 及 Intel EEPro 100 Mb/s 網路卡。首先我們利用 Inktomi 的 Content Network Platform 產品來建構一個 CDN，接著用 NIST Net 模擬了在 Internet 上，終端使用者到各種伺服器或 system 的網路延遲時間（latency）及封包遺失（packet loss），使測試床單（testbed）儘可能接近真實 Internet 的狀況，網路延遲時間及封包遺失率都會在所設的範圍內小幅變化。因為一般的終端使用者都會使用快取代理伺服器來連線，故終端使用者到快取代理伺服器的延遲時間（2ms），會比終端使用者到 replication server

的延遲時間 ( 15ms ) 來得短。Redirector 指的是 request-routing system , 通常是由 CDN 服務業者來管理 , 所以我們設定從終端使用者到 redirector 的 latency 是 200ms , 其餘從原始伺服器及 distribution system 到其他地方的 latency 均設為 300ms。在進行測試時我們均用 Sniffer 來統計網路流量及分析延遲時間。



圖八：測試平台、測試工具及模擬的網路延遲時間

### 建置及營運費用評估

對於 ICP 來說，如何用最有效率且最節省費用的方式，將內容遞送給終端使用者是其所追求的目標，在此我們比較 Web 複製式和 CDN 兩者所需花費的資金差異。首先假設某家 ICP 想要提供內容給散布在五個 ISP 網路的終端使用者，而整個營運的時間是兩年，也就是 24 個月。表四是比較 ICP 採用 Web 複製及內容遞送網路所需之費用。

表四：Web 複製式內容遞送及 CDN 投資費用的比較

( 數據資料來源：<http://www.business2.com/articles/mag/0,1640,9562,FF.html>,  
<http://techupdate.zdnet.com/techupdate/stories/main/0,14179,2321710,00.html> )

Investment in Web Replication				
Cost Type	Item	Price per Item	Number of Items	Subtotal
Deployment	Hardware and software	\$9,000	5	\$45,000
	Network equipments	\$2,000	5	\$10,000
Operation	Leased rack	\$600	5*24	\$72,000
	10Mb/s Leased line	\$5000	5*24	\$600,000
Total				<b>\$727,000</b>
Investment in Content Delivery Network				
Cost Type	Item	Price per Item	Number of Items	Subtotal
Operation	Average bandwidth	\$2000	10*24	\$480,000
Total				<b>\$480,000</b>

ICP 若採用 Web 複製的方式，初期建置的費用是花在 5 台伺服器的軟硬體設備及網路設備上，之後是營運的成本，包括租用機房場架 ( rack ) 費用以及 10Mb/s 網路專線 ( leased line ) 費用，整個加總之後所需要花費的資金為美金 727,000

元。Web 複製式的內容遞送缺點就是，ICP 必須先考量伺服器應付擁塞時的網路流量，所以在租用專線時需要採用能承受大量傳輸流量的網路頻寬，但在平常傳輸流量很小的時候，所花費高額資金的租用專線便浪費掉了。

若 ICP 採用 CDN 的營運方式，則 ICP 不必花費任何資金在購買伺服器軟體、租用場地及專線上，只單純地每個月付費給 CDN 服務業者來散布內容即可。CDN 服務業者的收費方式，是根據終端使用者使用 CDN 網路的量來估算，也就是每個月統計終端使用者所存取的總量，用總量除以這個月的時間，便可得到平均有多少 Mb/s，每個月每 1 Mb/s 的量是收費大約美金 2,000 元（Akamai 的收費），假設每個月平均會有 10 Mb/s 的量，一共營運了 24 個月，全部資金需要花費 480,000 元。比起採用 Web 複製式的營運方式，CDN 的方式可省下許多付出在閒置頻寬上的成本。

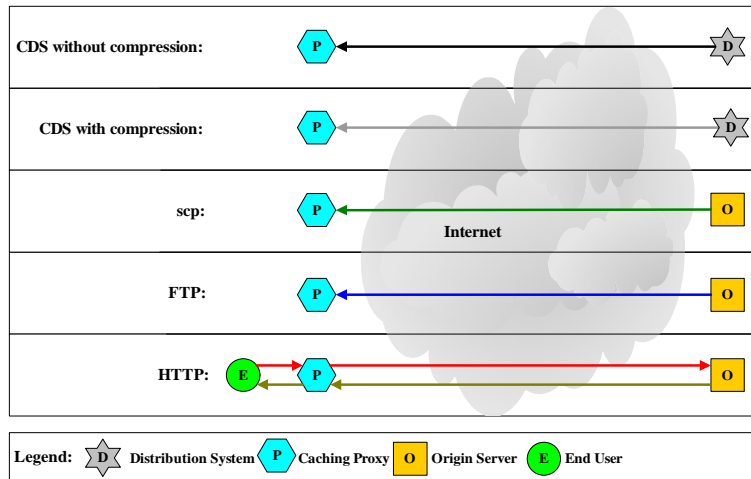
另外不可忽略的是 CDN 提供較快的內容下載速度及較高的可用性（availability）。根據 Zona Research 的研究指出，如果終端使用者等待下載內容的時間超過 8 秒，將有 30% 的終端使用者選擇停止瀏覽網頁。透過 CDN 較佳的服務品質，可以為 ICP 吸引到更多客戶，這同時意味著盈收的增加，這點對一個商業網站而言是不容忽視的。雖然實際可節省的成本隨各家 CDN 的計費方式而有所不同，但對有大量客戶需求的網站來說，CDN 不啻為一項較經濟的選擇。

#### **網路頻寬消耗及散布時間評估**

對於 ICP 及 CDN 服務業者來說，如果能以最少的網路頻寬消耗，及在最短的時間內將內容散布到各個 edge server 上，將可節省許多費用及時間。在此針對靜態內容及隨選串流媒體的散布來進行測試。

#### **靜態內容散布測試**

關於靜態內容散布的測試，我們一共測試 5 種散布的方式，如圖九所示，第一種方式是利用 Inktomi CDS 直接把內容散布出去，並不進行內容的壓縮；第二種方式則是利用 Inktomi CDS 先把內容縮壓後再進行散布；第三種方式是在透過 UNIX 的 secure copy( scp )指令，將原始伺服器上的內容複製到快取代理伺服器；第四種則是透過 FTP 來傳送；最後一種則是由終端使用者發出 HTTP 1.0，要求將在原始伺服器上的內容一一抓回去。我們拿 Apache HTTP server 及相關模組，在安裝好之後放在 document 主目錄的網頁內容，來當作要散布的檔案，總計共有 1,488 個檔案 9 個目錄，整體內容大小為 6475K bytes。



圖九：靜態內容散布測試

表五為靜態內容散布測試的結果，其中 Outbound 代表 distribution system 或原始伺服器流出到 Internet 的傳輸流量，而 Inbound 代表從 Internet 流入到 distribution system，或是原始伺服器的流量。從結果可看出利用 Inktomi CDS 來進行靜態內容散布，有非常不錯的效能，由於 CDS 是先將所有的檔案先打包成一個大檔案，然後再進行散布，所以在散布時間方面比後三者要少很多，如果打開 CDS 壓縮內容功能的話，更可減少頻寬的消耗。scp 是在一個連線內傳送多個檔案，而 HTTP 則是一個連線傳送一個檔案，但兩者都是一個檔案一個檔案傳，故所需要的時間比 CDS 要來的久。最差的就是用 FTP 來傳，時間花費了一個多小時，FTP 如此之慢是因為每傳一個檔案就要建立一次 ftp data 連線 (port 20)，而且會有許多 FTP 協定的 overhead，所以從這可以看出用 FTP 來散布內容是非常沒有效率的。另外如果在原始伺服器上的內容有部分檔案變更過的話，那我們會希望只要傳送有變更過的檔案即可，並不需要把整個目錄內的所有檔案重新傳送出去，這點 Inktomi CDS 就做得非常好，因為 CDS 在進行第一次的散布時，便會記下哪些檔案已被散布出去，下次要傳送變更過的檔案，CDS 會自動找出那些被變更或新增的檔案然後傳送出去。相較於傳統用 FTP 或 scp 方式需要人工去判斷哪些是有變更過檔案，Inktomi CDS 可以說是相當方便。

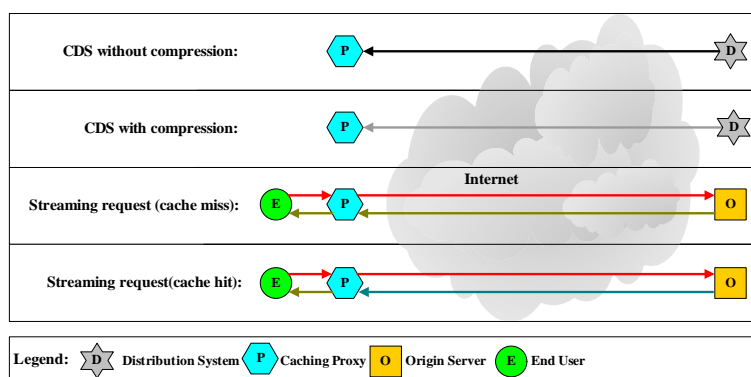
表五：靜態內容散布測試結果

Case	Outbound	Inbound	Distribution Time
CDS without content compression	14,044 KB	172 KB	2 min 26 sec
CDS with content compression	6,775 KB	81 KB	1 min 13 sec
Scp	7,095 KB	845 KB	19 min 30 sec
FTP	15,698 KB	493 KB	1hr 17 min 21 sec
HTTP	14,989 KB	977 KB	19 min 56 sec

### 隨選串流媒體散布測試

接著進行隨選串流媒體的散布測試，測試方式如圖十所示。一共測試四種散布的方式，第一及第二種是利用 Inktomi CDS 來進行散布，第三種則是由終端使用者端發出串流要求，透過快取代理伺服器連到原始伺服器，但快取代理伺服器是

cache miss，第四種方式和第三種類似是 cache hit。此處要注意的是雖然有 cache hit，但快取代理伺服器仍會連線到原始伺服器，檢查原始的 streaming 檔案是否存在，這種檢查的作法，是要確保終端使用者在快取代理伺服器上所取得 streaming media，和原始伺服器上的檔案是相同的。我們測試用的串流媒體檔案是 Microsoft Media Server 安裝好後在 ASFRoot 目錄底下的 sample.asf( 影片長度 59 秒 )，再加上設定 media server 時所產生的 sample.asx，兩個檔案一共為 1121K bytes，其中 sample.asx 僅有 125 bytes。



圖十：隨選串流媒體散布測試

隨選串流媒體散布的測試結果如表六所示。從結果可看出用 Inktomi CDS 來散布，可很快傳送出去。而第三種方式（cache miss）是採用「播放」的方式，除原本影片的長度是 59 秒，還要加上網路連線時間及快取代理伺服器儲存影片在自己快取中的時間，以及媒體播放程式預先緩衝（buffer）的時間，所花費的時間是 1 分 15 秒，但從原始伺服器流出到 Internet 的流量卻只有 537KB，這是因為雖然影片大小是 1M 多，實際上 media server 流出去的傳輸流量，不見得就和原始影片檔案大小一樣。第四種方式是在 cache hit 的情況下，快取代理伺服器只到原始伺服器取回影片的 metafile 來檢查，所以原始伺服器流出的傳輸流量很小。此外雖然第四種方式的散布時間是 1 分 9 秒，但實際上快取代理伺服器到原始伺服器的連線，大部分時間都是沒有傳送資料，也就是 idle 的。它僅僅在開始及結束播放影片時有傳送資料，在終端使用者所播放的影片是由快取代理伺服器來提供。

表六：隨選串流媒體散布測試結果

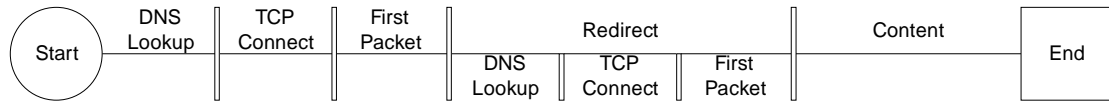
Case	Outbound	Inbound	Distribution Time
CDS without content compression	2,340 KB	32 KB	34 sec
CDS with content compression	1,014 KB	16 KB	20 sec
Streaming request ( cache miss )	537 KB	16 KB	1 min 15 sec
Streaming request ( cache hit )	11 KB	4 KB	1 min 9 sec

### 內容下載時間評估

對於一般的終端使用者而言，內容下載時間的長短會影響到終端使用者是否有耐性繼續瀏覽該網站。在此先分析整體內容下載的時間，然後再量測靜態內容及隨選串流媒體在各種遞送架構下的下載時間。

## 整體內容下載時間之分析

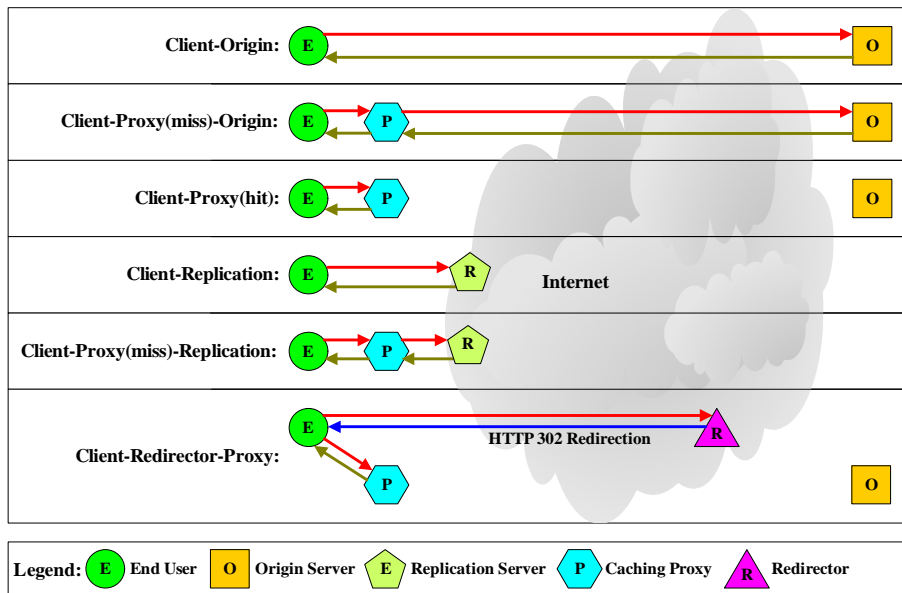
終端使用者所感受的內容下載時間，是從發出一個要求一直到收到所要的內容，當中所需要花費的時間（如圖十一）。其中包括 DNS 查詢時間、TCP 連線時間、伺服器延遲時間（伺服器傳回第一個封包給終端使用者）、重新導向時間以及真正內容下載時間，其中重新導向時間包括了新伺服器的 DNS 查詢時間、TCP 連線時間以及新伺服器傳回第一個封包的時間。



圖十一：整體內容的下載時間

## 靜態內容的下載時間測試

圖十二為各種靜態內容下載時間的測試，我們一共測試了六種狀況，第一種是終端使用者直接連線到原始伺服器；第二種是終端使用者透過快取代理伺服器在 cache miss 的情形下連線到原始伺服器；第三種是快取代理伺服器在 cache hit 的情況下，直接把快取中的內容直接回傳給終端使用者；第四種是終端使用者直接連線到 Web 複製式的 replication server；第五種是終端使用者透過快取代理伺服器，在 cache miss 的情形下連線到 replication server；最後一種狀況是終端使用者發出要求到 request-routing system，然後 request-routing system 回給終端使用者一個 HTTP 302 redirection 訊息，告訴終端使用者瀏覽器最好的快取代理伺服器是哪一台，然後瀏覽器會再到所指定的快取代理伺服器上取回內容。我們準備了大小為 5K bytes 的網頁來進行測試。



圖十二：靜態內容的下載時間測試

表七及圖十三為靜態內容下載時間測試的結果：第一、二種狀況會直接連線到原始伺服器，所以終端使用者感受到的下載時間最久；但如果是第三種狀況也就是

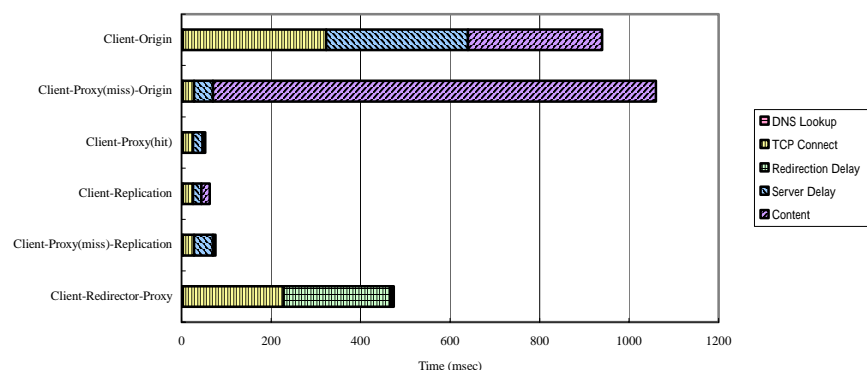
cache hit 的話，則下載時間可大幅縮短；第四、五種狀況是終端使用者直接或透過快取代理伺服器連到 replication server，所測量出來的下載時間很短；最後一種狀況則有透過 request-routing system，而在最好的快取代理伺服器上取得內容，我們可以發現，主要的延遲都是發生終端使用者與 request-routing system 建立 TCP 連線以及重新導向上。

從以上測試結果得知，事先將內容放置在 replication server 或架設快取代理伺服器，將可大幅減少終端使用者所感受的內容下載時間。這樣好像有 replication server 就夠了，終端使用者先連到 request-routing system 反而會增加下載時間。事實不然，因為 ICP 不可能在每一個靠近終端使用者的地方，都放置一台 replication server。就算可以，終端使用者要如何知道要選擇哪一台 replication server 才是最好的呢？一般的作法都是終端使用者先連上原始伺服器，然後所顯示的網頁會列出所有的 replication server 以供終端使用者選擇，但終端使用者在進行選擇的時候必須冒一些風險，像是所選擇的 replication server 實際上已不存在，或是該 replication server 的負載過重或網路擁塞，所以終端使用者在手動選擇 replication server 時，所需要花費的時間及承擔的風險遠比利用 request-routing system 來得高。

表七：靜態內容下載時間的測試結果

Case	DNS Lookup	TCP Connect	Redirection Delay	Server Delay	Content	Total Download Time
End user-Origin	3 ms	320 ms		317 ms	300 ms	940 ms
End user-Proxy ( miss ) -Origin	3 ms	25 ms		42 ms	990 ms	1,060 ms
End user-Proxy ( hit )	3 ms	23 ms		21 ms	6 ms	53 ms
End user- Replication	3 ms	23 ms		18 ms	19 ms	63 ms
End user-Proxy( miss )-Replication	3 ms	25 ms		42 ms	6 ms	76 ms
End user-Redirector-Proxy	3 ms	225 ms	238 ms	4 ms	4 ms	474 ms

Total Download Time: Static Content

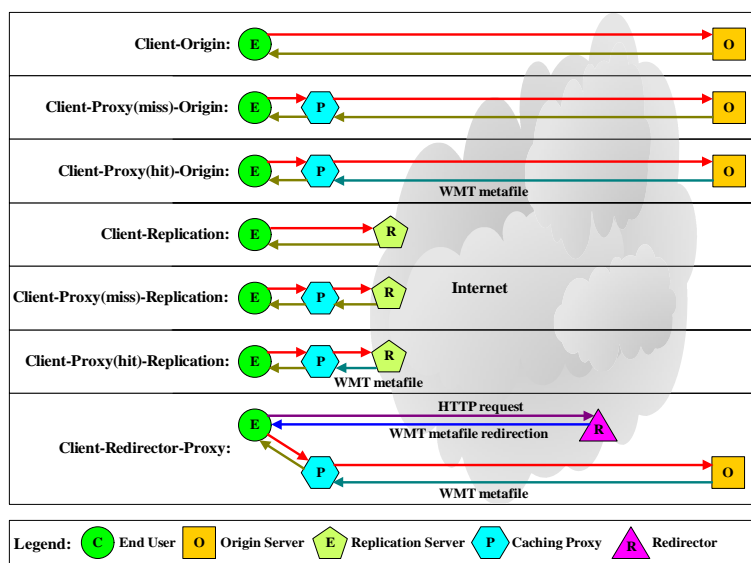


圖十三：靜態內容下載時間的測試結果

隨選串流媒體的下載時間測試

圖十四為隨選串流媒體下載時間測試，此測試共有 10 種狀況，但可分成四大類。

第一類為前三種狀況，是在原始伺服器連線的封包遺失率為 10%，終端使用者直接或透過快取代理伺服器連到原始伺服器，然後有 cache miss 及 cache hit 的狀況。第四到六種為第二類，和第一類類似但封包遺失率提高為 30%。第七到九種為第三類，也是和第一類類似但所不同的是，把原始伺服器取代為 replication server，而沒有任何封包遺失。最後一種狀況也就是第四類，則是終端使用者先建立 HTTP 連線到 request-routing system，爾後 request-routing system 回傳一個 metafile，在此 metafile 指定一台最好的伺服器給終端使用者。我們用的串流媒體檔案也就是先前測試所提到的 sample.asf，該影片長度為 59 秒。



圖十四：隨選串流媒體遞送下載時間測試

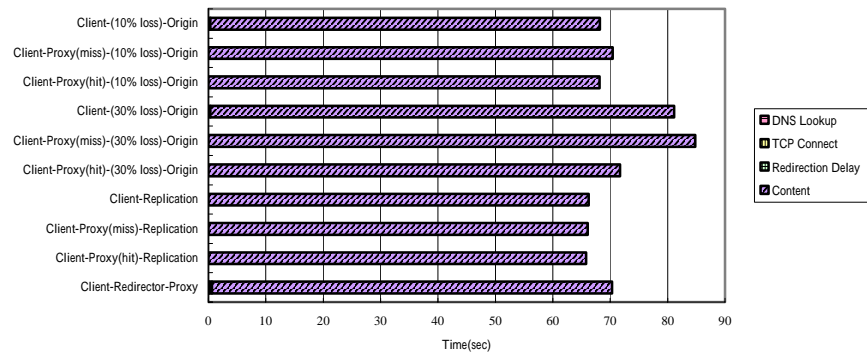
表八及圖十五為隨選串流媒體下載時間的測試結果，從結果中可看出在封包遺失率為 10% 的情況下，終端使用者播放影片的時間大約接近 70 秒，當然如果 cache miss 的話所需要花費的時間要久一點點。但如果封包遺失率提高為 30% 的話，終端使用者直接或透過快取代理伺服器連到原始伺服器，則影片播放時間需要 80 多秒，但如過 cache hit 的話，那延遲的狀況就可改善很多。接下來是終端使用者直接或透過快取代理伺服器連線到 replication server，其影片播放時間大約都是 66 秒。最後是終端使用者透過 request-routing system 來選擇一台好的伺服器來播放影片，其重新導向的時間大約是 0.4 秒。採用 request-routing system 的好處是可以幫終端使用者挑選一台好的伺服器，以避免終端使用者的連線經過，像是封包遺失率為 30% 的線路上。

表八：隨選串流媒體下載時間的測試結果

Case	DNS Lookup	TCP Connect	Redirection Delay	Content	Total Download Time
End user- ( 10% loss ) -Origin	0.003 s	0.304 s		67.883 s	68.190 s
End user-Proxy ( miss ) - ( 10% loss ) -Origin	0.003 s	0.006 s		70.408 s	70.417 s
End user-Proxy( hit )( 10% loss )-Origin	0.003 s	0.006 s		68.122 s	68.131 s
End user- ( 30% loss ) -Origin	0.003 s	0.305 s		80.827 s	81.135 s
End user-Proxy ( miss ) - ( 30% loss )	0.003 s	0.006 s		84.830 s	84.839 s

-Origin					
End user-Proxy( hit )( 30% loss )-Origin	0.003 s	0.006 s		71.682 s	71.691 s
End user-Replication	0.003 s	0.019 s		66.240 s	66.262 s
End user-Proxy ( miss ) - Replication	0.003 s	0.005 s		66.090 s	66.098 s
End user-Proxy ( hit ) - Replication	0.003 s	0.005 s		65.770 s	65.778 s
End user-Redirector-Proxy	0.003 s	0.204 s	0.401 s	69.717 s	70.325 s

**Total Download Time: On Demand Streaming**



圖十五：隨選串流媒體下載時間的測試結果

## 結論

如何在 Internet 上有效率地將內容遞送給終端使用者，一直是 ICP 所努力的目標，而採用傳統中央網站式及 Web 複製式的內容遞送方式，將會遭遇到許多問題。內容遞送網路主動地把內容散布到各個 ISP 端的 edge server 上，並且將終端使用者所發出的要求，重新導向到最好的 edge server 上。另外由於統計多工的效果，ICP 採用內容遞送網路的方式，將可減少投資的費用、節省骨幹頻寬消耗並減少終端使用者所感受到的內容下載時間。

究竟哪種類型的 ICP 適合使用內容遞送網路？我們認為內容為靜態，即非由原始伺服器動態產生，以及終端使用者散布在多家 ISP 網路，或是其網站容易發生網路擁塞的 ICP，比較適合使用內容遞送網路。ICP 的網站依其內容可分類為資訊網站（例如新聞網站）、電子商務網站（例如網路書店）、查詢網站（例如搜尋引擎）、串流媒體網站（例如音訊或視訊）等，在這些網站中的內容有些是靜態、有些是動態，內容遞送網站比較適合含有大量靜態內容的網站，例如新聞網站的新聞內容、網路商書的商品展示、入口網站的網站分類資料、隨選串流媒體的媒體檔案等，都可透過內容遞送網路，將內容事先散布到靠近終端使用者端的 edge server 上以加速終端使用者的存取。另外即時廣播串流頻道也適合透過 CDN 廣播至 edge server，增加同時可收看串流的用戶數量（即 fan-out 數量）。至於像是電子商務網站的網路下單、搜尋引擎等，由於內容是由網站根據終端使用者所輸入的資料而動態產生，故沒有辦法事先將內容散布到 edge server 上，所以目前比較不適合採用內容遞送網路。

內容遞送網路可朝更有效率、且更多元化的遞送方式邁進，例如 edge 端的快取

代理伺服器可 cache 動態所產生網頁的內容，如此可更減少頻寬的消耗。另外也針對即時串流進行最佳化的遞送，讓即時串流的遞送避開網路擁塞之處，讓終端使用者得到更好的服務品質（Quality of Service；QoS）。還有一個有趣的趨勢就是，網頁的內容可針對個別終端使用者，提供個別化的服務。例如在 Amazon 的網站可以根據終端使用者的查書記錄推測出其喜好，並據此推薦新書。另外若個別化的工作全都在原始伺服器上做，顯然會有較長延遲性的問題，且使現行的 CDN 架構無法表現其優異。至於如何把部分工作推至 edge 端進行，或是在 edge 端對內容提供加值型的服務，將會是個有趣的課題。讓我們期待內容遞送網路的技術更加成熟發展。

### **Inktomi 廠商的回覆：**

如今 CDN 除了在具有散布 HTTP 內容的能力之外，還應該要具有散布 live 及 on demand streaming 的能力。Inktomi 做為一個業界領先的 content networking 技術軟體供應商，已經開發了內容網路平台（Content Networking Platform；CNP）來滿足市場的需要。CNP 不僅具有傳統的快取（caching）及散布（distribution）的能力，而且透過建立在現有 IP 網路之上的媒體散布網路（Media Distribution Network；MDN）來強化 live 和 on demand streaming 對於網路頻寬的使用。Inktomi 的 CNP 是滿足當今和未來需求的 CDN 和 MDN 的統一平台。此外 Inktomi 的 CNP 還結合許多 3rd-party 的加值服務（value-added services）商的技術，例如內容過濾（content filtering）、病毒掃描（virus scanning）和計費（billing）等，並將這些技術無縫地結合來提供一個完整的解決方案。

## **參考資料**

### **Papers:**

- [1] P. Mills and C. Loosley, “A performance Analysis of 40 e-Business Web Sites,” “CMG Journal of Computer Resource Management, Issue 102, spring 2001.

### **IETF Drafts：**

- [2] A. Barbir, F. Douglass, M. Green, M. Hofmann, R. Nair, D. Potter, and O. Spatscheck, “Known CN Request-Routing Mechanisms,” draft-cain-cdnp-known-request-routing-04.txt（work in progress），  
<http://www.ietf.org/internet-drafts/draft-cain-cdnp-known-request-routing-04.txt>, November 2001.
- [3] B. Cain, G. Tomlinson, S. Thomas, P. Rzewski, “Content Internetworking Architectural Overview,” draft-green-cdnp-gen-arch-03.txt（work in progress），  
<http://www.ietf.org/internet-drafts/draft-green-cdnp-gen-arch-03.txt>, March 2001.

- [4] M. Day, B. Cain, G. Tomlinson, and P. Rzewski, "A Model for CDN Peering," draft-day-cdn-model-09.txt ( work in progress ) ,  
<http://www.ietf.org/internet-drafts/draft-day-cdn-model-09.txt>, November 2001.

**White papers:**

- [5] Amamai Technologies Inc., "Internet Bottlenecks: the Case for Edge Delivery Services," Akamai White Paper,  
[http://www.akamai.com/en/html/services/white\\_paper\\_library.html](http://www.akamai.com/en/html/services/white_paper_library.html), 2000.
- [6] stardust.com Inc., " White Paper - The Ins and Outs of Content Delivery Networks, " <http://www.stardust.com/cdnweek/whitepapers/cdnspring01>, February 2001.
- [7] Zona Research Inc., "The Economic Impacts of Unacceptable Web-Site Download Speeds,"  
[http://www.keynote.com/solutions/assets/applets/wp\\_downloadspeed.pdf](http://www.keynote.com/solutions/assets/applets/wp_downloadspeed.pdf), April 1999.

**Reports:**

- [8] B. Hulsebosch, "Content Distribution Networks: State of the Art," Telematica Report, <http://www.telin.nl/>, June 2001.
- [9] M. Posey and C. Munroe, "U.S. Content Distribution/Delivery Services Market Forecast and Analysis," IDC Report,  
<http://www.idc.com/getdoc.jhtml?containerId=22873>, August 2000.
- [10] R. Wetzel, "CDN Business Models – Not All Cast from the Same Mold," Business Communications Review, April 2001 Issue, pp. 40-46, 2001.

**Books:**

- [11] D. G. Messerschmitt, "Understanding Networked Applications: A First Course," Morgan Kaufmann, 2000.

**Web sites:**

- [12] Content Bridge, <http://www.content-bridge.com/>.
- [13] Content Alliance, <http://www.content-peering.org/>.
- [14] NIST Net, <http://snad.ncsl.nist.gov/itg/nistnet/>.