

[首頁](#) > [學習](#) > [PHP on Windows](#) > IIS 7 輸出快取

焦點

IIS 7 安裝與部署

IIS 7.0 操作手冊

IIS 擴充功能

規劃您的 IIS 7 架構

在 IIS 7 上部署網站

管理與維護 IIS 7

在 IIS 7 上裝載應用程式

Web Platform Installer

影片

虛擬實驗室(Virtual Lab)

PHP on Windows

IIS 7 輸出快取

IIS 7 可以讓我們設定輸出快取來改善網頁伺服器、網站或應用程式的效能。啟用輸出快取之後，IIS會複製一份處理過的網頁，並儲存在自己的記憶體，如果後續用戶端的要求需要相同的資源，會直接從自己的記憶體取出並傳回給用戶端。這樣就不需要在每次處理用戶端要求時，都重新處理網頁，因此能提高伺服器效能，縮短用戶端的等待時間。尤其，如果網站內容是以程式動態產生（例如 PHP 或 ASP.NET），或者是以外部來源資料作為網站內容（例如資料庫），開啟輸出快取將對網站伺服器產生更大的效益。

本文將討論 IIS 7 提供的輸出快取功能，包括輸出快取的簡介，以及如何在 IIS 7 設定輸出快取。

輸出快取二三事

輸出快取之所能改善動態產生網頁內容所造成的效能瓶頸，簡單來說，就是利用快取的特性，預先儲存動態產生的網頁內容，如果遇到用戶端提出相同的要求（且內容未過期），就直接將快取的內容回應給用戶端，而不需重新執行網頁程式。

快取是為了提高效率，因此會將快取的結果（快取物）存放在記憶體（因為存取速度快），但多數電腦的記憶體數量大多有限，至少是相對有限，因此要空出記憶體給真的能提高效能的快取物。

就快取的角度而言，網頁內容可以分成靜態、動態、半動態。靜態內容的更動機率低，例如 HTML 檔案、CSS 檔案、圖檔等。由於內容更動機會低，因此很適合成為輸出快取的快取物。IIS 會自動快取靜態內容，但也會自動偵測這類檔案的內容是不是有所更動，而視情況更新其快取物。

PHP 或 ASP.NET 網頁都可以動態產生內容。動態內容的更動機率很高，這種內容是由程式依據要求當時的情況即時產生，通常高度個人化的網頁，在每次要求時都會產生不同的結果。如果動態內容的更動機率太高，每次要求的結果都不同，就沒有快取的價值。

半動態內容介於靜態內容和動態內容之間，通常是由程式依據要求當時的情況即時產生，但是若要求的情況沒有改變，其結果也不會有異。例如其目的是查詢資料庫的 PHP 網頁，雖然這看似動態內容，但如果資料庫內容及查詢條件都沒變，查詢結果也不會有變，而這就有快取的價值。

換句話說，根據 URL 或標頭資訊來判斷，資料雖是動態產生，但每次產生的資料確定不會有變的動態網頁，就是輸出快取的最佳來源。再舉另一個適合輸出快取的例子，相簿網站會視情況動態調整網頁所要顯示的相片大小，但顯示的相片並沒有變，就很適合輸出快取，因為這能省下伺服器必須因為每次要求而前置調整相片大小的時間。IIS 7 的輸出快取是以半動態內容為目標，可以快取動態要求的靜態回應，並能根據查詢字串值和用戶端送到伺服端的 HTTP 標頭來更改已快取的輸出，而且為了達到更好的效能，也已經整合到核心模式的驅動程式 Http.sys。而且 IIS 7 的快取也能增加延展性。

IIS 7 輸出快取簡介

IIS 7 支援兩種快取方式：

- varyByQueryString：URL 相同但查詢字串值有變。
- varyByHeaders：根據用戶端傳送到伺服端的 HTTP 標頭來快取。

關於 varyByHeaders，舉例來說，如果站台或應用程式裡有搭配不同瀏覽器版本的網頁內容，可以根據 HTTP User-Agent 標頭所傳回的瀏覽器類型，設定要快取不同網頁內容的輸出快取規則。因此當用戶端要求站台或應用程式裡的網頁時，如果快取裡有符合用戶端瀏覽器版本的網頁，IIS 就會將該網頁的快取版本直接送給用戶端。

至於 varyByQueryString，假設有一個會依據使用者所輸入的產品識別碼而傳回不同網頁內容的產品型錄應用程式，而所依據的產品識別碼是以查詢字串傳送。因此，我們就可以根據查詢字串裡的產品識別碼來快取不同網頁。

快取失效

動態的內容如果有變，就必須更新快取，也就是重新快取新的內容；IIS 7 的作法是提供兩種使快取物失效的方式（一旦快取物失效，就有機會重新快取而更新內容）。第1種方式是以 CacheForTimePeriod 屬性設定快取逾時的時間，另一種是快取物有變就使之失效，這可利用 CacheUntilChange 屬性設定。

最後要說明的是，就算開啟了輸出快取，IIS 7 也不會立即快取，而必須等待一段時間，直

到IIS 7認為有快取價值。快取價值可透過 ServerRuntime 來設定，決定快取價值的兩個屬性分別是frequentHitTimePeriod、frequentHitThreshold，亦即「命中頻率的時間長度」，以及「命中頻率的門檻」；也就是說，如果可以快取的URL在frequentHitTimePeriod之內到達，且超過frequentHitThreshold，就能被快取。frequentHitTimePeriod的預設值是10秒，而frequentHitThreshold的預設值是2秒。

使用者模式快取和核心模式快取

此外，IIS 7的輸出快取處理可以利用使用者模式和核心模式：使用者模式是以 IIS 7 的工作者行程來處理快取，而核心模式則是以核心模式的驅動程式 Http.sys 來處理。使用者模式和核心模式的輸出快取IIS 7預設都會啟用，但核心模式快取設定值的優先順序高於使用者模式快取。

核心模式的快取雖然因速度較快而能提高網站的效能，但不支援諸如驗證或授權等只能在使用者模式執行的模組或功能。此外，核心模式輸出快取只支援 varyByHeaders，不支援 varyByQueryString。

設定輸出快取

警過簡單的說明之後，接著將示範輸出快取的設定。首先說明如何利用 IIS 管理員開啟快取，以及設定快取相關參數。最後我們將示範命令列的快取設定方式。

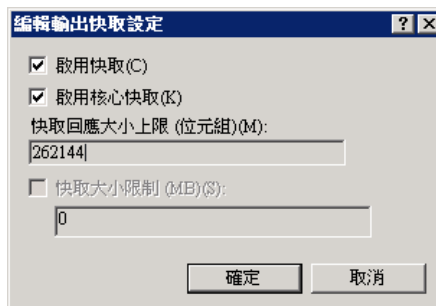
啟用快取

- 1.開啟 IIS 管理員，展開左側『連線』窗格裡的『站台』節點，並點選欲設定輸出快取的網站。
- 2.請以滑鼠雙按 IIS 功能群組裡的『輸出快取處理』（如下圖）。



點擊放大

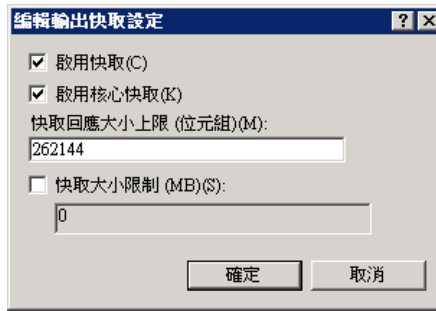
- 3.按一下右側『動作』窗格裡的『編輯功能設定』，會出現『編輯輸出快取設定』對話窗（如下圖）。



- 4.『編輯輸出快取設定』對話窗裡的兩個核取方塊預設都以勾選，如果取消『啟用快取』的勾選，表示要關閉使用者模式輸出快取的功能；如果取消『啟用核心快取』，則表示要關閉核心模式快取。

- 5.『快取回應大小上限』裡的數值是指定可以儲存在輸出快取的回應大小上限（以位元組為單位）。預設值為 262144 個位元組，也就是 256 KB。如果將這個值設定為 0，可讓 IIS 允許任何大小的回應儲存在輸出快取。輸出快取所使用的記憶體總數仍然會受到記憶體限制之快取大小上限的規範。

- 6.『快取大小限制』裡的數值是指定 IIS 用於輸出快取的記憶體總數量，先勾選核取方塊，即可輸入數值。要提醒您的是，設定太大可能會導致輸出快取使用過多的記憶體，因而可能會對伺服器 and 應用程式元件的效能有不利的影響。預設值為 0，表示讓 IIS 自動管理快取記憶體的大小。再者，不論上一個『快取回應大小上限』或者『快取大小限制』，都必須從網頁伺服器層級設定，上圖是由站台進入設定，因此呈現灰色無法更改。下圖是從網站伺服器進入設定，因此可以設定這兩個數值。



7.完成設定之後，請按下『編輯輸出快取設定』交談窗裡的『確定』按鈕。

設定快取規則

啟用輸出快取之後，還必須設定快取規則，讓 IIS 依據我們設定的規則來進行快取。

- 1.開啟 IIS 管理員，展開左側『連線』窗格裡的『站台』節點，並點選欲設定輸出快取的網頁伺服器、站台或應用程式。
- 2.請以滑鼠雙按 IIS 功能群組裡的『輸出快取處理』（如下圖）。



[點擊放大](#)

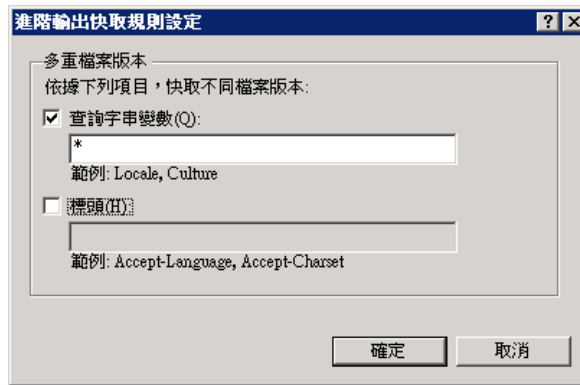
- 3.按一下右側『動作』窗格裡的『新增』，隨即出現『新增快取規則』交談窗（如下圖）。



4.您可以在『新增快取規則』交談窗的『副檔名』文字方塊輸入『.php』（或者您所設定的 PHP 程式副檔名，或者其他您想快取的檔案副檔名），然後視情況勾選『使用者模式快取處理』或『核心模式快取處理』。『檔案快取監視』裡的『使用檔案變更通知』和『時間間隔』，就是前述 IIS 7 提供的兩種使快取物失效的方式（前者是 CacheUntilChange，後者是 CacheForTimePeriod）。『防止所有快取處理』會防止快取指定的檔案類型。

5.按下『新增快取規則』交談窗的『進階』按鈕，這會出現『進階輸出快取規則設定』交談窗（如下圖），目的是讓您使用標頭或查詢字串來設定輸出快取處理規則。『進階輸出快取規則設定』交談窗裡的『查詢字串變數』和『標頭』就是前述 IIS 7 所支援的兩種快取

方式 (前者是 varyByQueryString，後者是 varyByHeaders)。您可視情況勾選『查詢字串變數』或『標頭』核取方塊，再輸入適當的變數或標頭。不論字串或標頭，如果要輸入一筆以上的字串或標頭，請以逗號分隔。



6.完成之後請分別按下『進階輸出快取規則設定』對話窗、『新增快取規則』對話窗的『確定』按鈕來關閉這些對話窗。

觀察 web.config 檔案

透過IIS管理員完成設定之後，這些設定也會反映在 web.config 檔案，例如以下簡單的範例。

```
<configuration>
  <system.webServer>
    <cacheing>
      <profiles>
        <add extension=".php" policy="CacheUntilChange"
              kernelCachePolicy="DontCache"
              varyByQueryString="*" />
      </profiles>
    </cacheing>
  </system.webServer>
</configuration>
```

在上述的設定：

- 快取的檔案副檔名為 .php
- 快取失效的方式是根據快取物的變化，也就是 CacheUntilChange
- 核心模式快取已開啟但未使用，因此 kernelCachePolicy="DontCache"
- 要以所有的查詢字串為依據 (varyByQueryString="*")

以命令列程式 appcmd 設定輸出快取

除了 IIS 管理員，我們也可以利用命令列程式 appcmd 來設定輸出快取，例如以下的例子：

```
%windir%\system32\inetsrv\appcmd set config "Default Web Site/abc"
    -section:cacheing /+profiles.[extension='.jpg',duration='00:00:10',
    policy='CacheForTimePeriod',varyByHeaders='Accept-Language']
```

以上例子的說明如下：

- 快取的對象是 Default Web Site 網站下的應用程式 abc
- 快取的檔案副檔名為.jpg
- 快取失效的方式是時間逾時 (0時0分10秒)
- 要以 HTTP 標頭為依據，所依據的標頭是 Accept-Language

總結

對於動態產生內容的網站後端技術來說，常被討論的缺點就是每經用戶端要求，就要執行一次程式來產生網頁內容回應用戶端。因為網頁內容隨用戶端要求而透

程式的執行才動態產生，因此容易加重伺服器的負擔。因此有人想到了利用輸出快取來改善動態產生網頁內容所造成的效能瓶頸。

善用快取可以提高電腦運作的效能，對於 IIS 7 也一樣，尤其當IIS經常需要回應大量相同的檔案 (例如圖檔)，適當的設定輸出快取規則，對效能的改善會有相當的助益。

探索：你的 IIS

伺服器	工具	社群	部落格	更多資訊...
Windows Server	Visual Studio	ASP.NET Community	MSDN Blogs	IT professionals (TechNet)
SQL Server	Expression Studio	Visual Studio Community	Web Tools Team Blog	Web Developers (MSDN)
IIS	Windows Internet Explorer	IIS Community	Web Platform Team Blog	

技術	網站應用	Silverlight Community	Silverlight Team Blog	Web Designers (Expression)
.NET Framework	SharePoint Server	MSDN Community Center	IE Team Blog	Enterprise Organizations
ASP.NET	Commerce Server	Channel 8	Mix Online	Hosters
Silverlight	BizTalk Server	Channel 9	Scott Guthrie's Blog	
PHP on Windows	Web App Gallery	Channel 10	Scott Hanselman's Blog	
IIS Media Services		CodePlex		