

# 序

自從我寫了這本書的第一版，servlet 和 server-side Java 平台的成長情況，遠遠超過任何人最樂觀的預期，頗有改朝換代的氣勢。Web server 的供應商將 servlet 的支援視為標準規格，而 Java 2 Enterprise Edition (J2EE) 也納入 servlet，將它當成核心元件。別以為 servlet 只是廠商的宣傳噱頭而已，如果不提供一套有規模彈性的 servlet 成品，application server 的供應商將提早出局。Servlet 已經成為 JavaServer Page (JSP) 和其它 frameworks 的基礎，而 servlet 技術現在也支援 ESPN.com 和 AltaVista.com 這類的高流量網站。

不意外地，servlet 目前的遠景，看起來已不同於本書第一版剛問世時的初衷。Servlet API 已經歷了兩次改版，而第三版也正在進行中。Live Software 和 New Atlanta 這兩家新興公司，也因為銷售 JRun 和 ServletExec 這兩套「servlet 引擎」（現在的正式名稱為 servlet container）而聲名大噪，賺進不少鈔票，進而分別被 Allaire 和 Unify 這兩家著重 Web 業務的大公司買下。他們現在所提供的功能，已超越基本的 servlet 支援，希望能藉此脫穎而出。

令人訝異地，正式的 `javax.servlet` 和 `javax.servlet.http` 套件 (package) 是首度以 Open Source (開放原始碼) 形式發表的 Java classes。它們被交給 Apache Software Foundation (ASF) 所主持的 Jakarta 計畫 (<http://jakarta.apache.org>)，這套件將繼續遵循 Servlet API 規格，但修正錯誤以及更新規格的任務，則交給一組值得信賴的 Open Source 發展者 - 如果這就是你，那麼你最近將有機會修正一個 bug，以改善 `HttpServletRequest` 對「條件式 GET 申請」(conditional GET request) 的處理過程。此外，當成 Servlet API 參考成品的伺服軟體也已轉移給 ASF，並以 Apache Tomcat 的名義當成 Open Source 軟體開放給公眾使用。也因此，Tomcat 已成為最受歡迎的 servlet container 之一。在 <http://opensource.org> 可獲得更進一步的消息。

Servlet 的世界已經改變了許多，而本書將帶來最新資訊。從開始規劃 Java servlet 程式，到完成後的收尾工作，這整個過程所需知道的點點滴滴，本書都將巨細靡遺地仔細解釋。前五章交代 servlet 的基本概念：什麼是 servlets？它們作了些什麼？以及它們是怎麼辦到的？接下來的十五章是真正的“牛肉”所在，探討你可用 servlet 來做些什麼事，以及你可能會用到的工具。你會在這本書找到大量的範例、一些建議、少數警告，和兩篇讓本書能通過技術校稿者法眼的研究心得。

## Servlet API 2.2

本書第一版只涵蓋到 Servlet API 2.0，而第二版則涵蓋了 Servlet API 2.2，這版本在 1999 年八月公開，而在 1999 年十二月定案。從 2.0 到 2.2 有很大的變革：

- 制訂如何將 servlet 散佈到多台後端伺服器 (backend server) 的規則。
- Servlet 現在運用了“可外掛的 Web 應用程式” (pluggable web application)，這使得規劃與佈署的工作可以不受伺服器平台的限制。
- 大幅提昇 servlet 的安全性。
- Servlet 現在可將 request 轉交給其它伺服元件來的處理。
- Servlet 之間可使用 `ServletContext` 來分享資訊。
- Servlet 的資源現在可以被抽象化，以支援分散式存取。

- 更進一步控制 session 管理的能力。
- 添增對 response 的輸出緩衝能力。
- 加強對 HTTP header 的控制。
- 運用更先進的錯誤處置機制。
- 重新整頓 API，讓 method 的名稱更合理、更加一致。
- Servlet API 現在是以正式的規格書來定義，其未來演進（改版）的工作，以正式的 Java Specification Request（JSR）處理程序來管理。
- Servlet 現在已被整合進 Java 2 Enterprise Edition（J2EE）規格的大傘之下。

本書不僅會解釋上述變革，其它比較小的變動也會一併說明。此外，還有五種以 servlet 為基礎的 Open Source 技術，可用來大幅提昇網站產生動態內容的效率，本書也將涵蓋它們，這五種技術分別是：

- JavaServer Pages（JSP），這是 Sun 制定的標準，被發展來搭配 servlet。
- Tea，由 Walt Disney Internet Group（前身是 GO.com）首創的技術，用於應付 ESPN.com、NFL.com、Disney.com、DisneyLand.com、GO.com 和 Movies.com 這類的高流量網站。
- WebMacro，由 Semiotek 首創，被用於 AltaVista 搜尋引擎。
- XMLC，這是 Lutris Technologies 為了解放 XML 技術在 Web 上的潛力，進而發展出來的技術。XMLC 可運用於 customatix.com 之類的創意網站。
- Element Construction Set（ECS），這是 Apache 的傑作，用來應付最迫切的程式設計需求。

本書也會介紹 Wireless Application Protocol（WAP），並解釋如何為無線裝置建置 servlet-based 的網站應用系統。

## Servlet API 2.3

就在寫作本書的同時，Servlet API 2.3 的發展也正在進行中，但尚未完成。有鑑於此，對於預期在 Servlet API 2.3 會有的變動部分，本書將簡略提及。在本書最後一章，將徹底討論 2000 年十月所發表的 Servlet API 2.3，讓讀者對 Servlet API 2.3 規格的新變革有些概念。需要注意的是，這規格書仍然可能會有變動，最終定案的版本，可能會與本書內容有所出入。

## 第一版的讀者

《Java Servlet 設計，第一版》的讀者將發現，本書已依據 Servlet API 2.2 規格大幅改版（甚至有部份是依據 Servlet API 2.3），每一章都將比第一版為你帶來更多閱讀樂趣；此外，本書也添增了六個新章節，涵蓋了多種以 servlet 為基礎的網頁內容編製技術。《第 12 章 Enterprise Servlets 與 J2EE》是第七個新章節，解釋 servlet 是如何被整合進 J2EE 平台的。

由於 pulgable web application 模型已明顯衝擊 servlet 程式設計的整個觀念，我們建議已經讀過第一版的讀者，最好重新閱讀有興趣的每一章節，並注意那些專用來解決傳統任務所設置的新機制。對於時間不夠充裕的讀者，可參考稍後的《章節編排》一節，看看 servlet 技術的主要變革之處。

## 讀者定位

這本書是為你而寫的嗎？如果你打算在 Web 上佈署開發出來的應用系統，那麼答案將是肯定的。說的更明白些，本書是為協助下列人士所寫的：

### J2EE 程式設計師

Servlet 是 J2EE 標準的一部份，為 J2EE server 開發程式的設計師，將會體認到把 servlet 整合進其它後端技術，是件多麼美妙的事。

### JSP 程式設計師

JavaServer Pages (JSP) 是架構在 servlet 之上。要完全享用 JSP 的好處，必須對 servlet 有一定程度的瞭解，而這正是本書所提供的。本書對 JSP 以及其它四種類似用途的技術，都提供了入門教材。

### Java applet 程式設計師

基於一些安全性的考量，要讓 applet 能夠跟伺服器交談，始終是件困難的事；直到 servlet 的出現，這情況才有所改觀。Servlet 可在伺服器端提供一個以 Java 寫成的代理程式 (agent)，這讓 applet 有個可以輕易連線的對象，而同時又兼顧安全性。

### CGI 程式設計師

CGI 是常用來擴充 Web 伺服器功能的技術，但是其效率一直遭到詬病，而 servlet 提供了另一種又精緻、又有效率的選擇。

### 其它伺服器程式的設計師

有不少 CGI 替代方案，包括 FastCGI、PHP、NSAPI、WAI、ISAPI、ASP 和現在的 ASP+。每一種方案在可移植性、安全性、效能，以及後端資源的整合能力方面，或多或少都有某種程度的限制。而 servlet 在這些領域則有優異的表現。

## 閱讀本書所需的背景知識

當我們一開始寫作這本書時，我們發覺最困難的一件事，就是不知應該假設讀者已經懂了多少。你熟悉 Java 嗎？你寫過 CGI 或其它 Web 程式嗎？你用過 servlet 嗎？你懂 HTTP 及 HTML 嗎？不管我們怎麼想像讀者的定位，總會有些人會認為太簡單，同時也會有人抱怨實在太難。

因此，我們決定只說明原始的主題；至於過於煩瑣的部分，讀者可查詢那些已寫得很好的線上文件或其它書籍，遇到這種情形，我們會提供參考資訊。

當然，外界參考能幫上忙的仍然有限。我們預期本書讀者已學會了 Java 程式語言，並具備物件導向程式設計的基礎，如果你只有其它語言的基礎，我們建議你先閱讀相關的 Java 程式設計書籍，像是 Patrick Niemeyer 與 Jonathan Knudsen 合著的《JAVA 學習手冊》(歐萊禮)，你可以忽略關於 applet 與 AWT 的章節，把重點放在網路與多緒 (multithread) 這些主題上。如果你想從 servlet 著手學習 Java，我們建議在你閱讀本書的同時，手邊放一本 David Flanagan 寫的《JAVA 技術手冊，第三版》(歐萊禮)，或其它 Java 參考手冊。

要閱讀本書，不必具備深入的 Web 程式設計、HTTP 以及 HTML 等知識，但本書也不會細說這些主題，我們只會提及相關的必備基礎知識，讓你能開發出有效率的 servlet，至於其它要點 (像是完整的 HTML tag 與 HTTP 1.1 headers) 就留給其它來源發揮。

## 關於本書範例

本書包括超過上百個 servlet 範例，這些範例的程式碼都印在內文中，如果讀者不想看著書敲鍵盤，可以到 <http://www.oreilly.com/catalog/jservlet2> 或本書的中文網頁 <http://www.oreilly.com.tw/chinese/java/servlet2.htm> 下載程式碼。你也可以到 <http://www.servlets.com> 看看這些 servlet 的樣子。

本書所有程式碼已於 Apache Tomcat 3.3 (包括 Windows 和 Unix 版) 配合 JDK 1.1.8 與 1.2.2 的 Java Virtual Machine (JVM) 測試過。少數幾個比較進階的範例是在其它伺服器上測試的，因為這些範例所用的功能，是 Tomcat 的 standalone 模式所不支援的；對於這種情況，我們會內文裡說明其細節。Apache Tomcat server 是正式的 Servlet API 參考成品，正如其版權宣示，你可在 <http://jakarta.apache.org> 免費取得原始碼。

本書還提供了一套輔助工具類別，許多範例都會用到它們，當你開發自己的 servlets 時，也可能會發現這套工具頗為方便。我將這些類別包裝在 `com.oreilly.servlet` 套件內，它們能幫 servlet 分析參數、處理檔案上傳、產生多部回應 (multipart response, 即 server push)、協調 locale (地域性)、傳回檔案、管理 socket 連線、當成 RMI server、甚至還有一個可協助 applet 與 servlets 溝通的類別。此外，第二版還加入了一些新類別來幫助 servlet 寄發電子郵件、快取應答 (cache responses)，以

及自動偵測是否支援 servlet API。本書已列出了 `com.oreilly.servlet` 套件的大部份原始碼，至於最新的完整版本（連同 javadoc 說明文件），可從 <http://www.servlets.com> 下載【註】。

## 章節編排

本書包含二十個章節與六篇附錄，如下：

### 第一章《簡介》

說明開發 web 應用系統時，Java servlet 所扮演的角色與其優點。第二版已被更新過，額外添增了一些伺服器軟體的相關消息。

### 第二章《HTTP servlet 基礎》

簡要說明 HTTP，以及 HTTP servlet 能進行的工作；示範產生網頁的基本過程，並介紹 pluggable web application 的概念。第二版涵蓋了 web 應用系統與它們的 XML-based deployment descriptor。

### 第三章《Servlet 生命週期》

詳細解釋載入、執行 servlet 的時機與方式，以及如何管理 threads，並探討多緒系統如何處理同步化的議題。狀態延續的能力（persistent state capabilities）也涵蓋在內。第二版增加了有關 context reloading 和註冊 servlet 的新規則，一個有關伺服器端快取的小節，以及 `super.init(config)` 的附帶說明。

---

註 本書沒有附贈光碟，因為這會提高本書的成本和售價。我們認為，會有興趣閱讀本書的每位讀者，應該都有能力連上 Internet，並願意從網站下載範例程式，以省下買光碟的錢。另外，由於 servlet 市場變化相當快速，這些附在光碟中的 web server 和 application server 等軟體，可能在書還沒出版之前，就因為過時而面臨淘汰的命運。這些體驗版的軟體，都能在網路上找到，建議你自己去下載最新的版本。在讀這本書之前，找個機會到 <http://www.servlets.com> 去下載範例程式和 Apache Tomcat web server 吧！

#### 第四章《擷取資訊》

介紹 servlet 最常用來取得資訊的方法。這些資訊包括用戶端、伺服器、用戶端的 request、以及 servlet 自己。本章會示範一個 reusable class 如何處理檔案上傳。第二版討論如何設定 deployment descriptor 中的資訊、取得 servlet 的名稱、存取暫時性目錄、處理 context (環境) 的初始參數、判斷 Servlet API 的版本、指派 servlet mappings (對應表)、存取抽象資源等等。此外，還示範了一個改良過、更有彈性的檔案上傳元件。

#### 第五章《傳送 HTML 資訊》

描述 servlet 如何產生 HTML、傳回錯誤訊息及其它狀態碼、對 response 作緩衝處理、重導 request、寫資料到伺服器日誌檔 (log file)、傳送自訂的 HTTP header 資訊。第二版新增的內容，包括 response 的緩衝處理、一個真正有用重導 (redirection) 範例，並新添一章節，專論如何設定錯誤訊息頁與如何處置錯誤。

#### 第六章《傳送多媒體內容》

看看 servlet 能傳回些什麼有趣的事物：給無線裝置的 WAP/WML 內容、動態產生的影像、壓縮過的內容、多部應答 (multipart response)。第二版新增的涵蓋範圍，包括 WAP/WML、歡迎檔清單 (welcome file lists)、PNG，一個改良的伺服器端影像快取，並對深入討論如何產生壓縮的內容。

#### 第七章《進程追蹤》

說明如何在乏態 (stateless) 的 HTTP 協定上建立狀態 (state) 的概念。本章前半展示傳統 CGI 開發者所用的進程追蹤 (session tracking) 技術，後半則詳述如何使用 Servlet API 對 session tracking 的內建支援。第二版還多了 web application session 的規則、新 session method 名稱、討論如何管理逾時 (time-out)，以及 applet-based 的 session tracking。

#### 第八章《安全性》

解釋分散式運算所涉及的安全性議題。示範如何利用標準的 servlet hook 來管理使用者帳號，以及如何用自訂的身份驗證 (authentication) 和授權 (authorization) 機制來建立更強悍的系統。對於 servlet 在 SSL 安全通訊中所扮演的角色，也多有解釋。這一整章在第二版可說是已全面改寫、更新過。



### 第九章《資料庫連結》

示範如何將 servlet 應用到高效率的 Web-database 連結，本章並提供一個 JDBC 的教學課程。第二版新增的內容，包括如何使用 properties 檔來設定連結，一個新的訪客留言簿範例，以及介紹 JDBC 2.0 的新章節。

### 第十章《Applet-Servlet 通訊》

探討需要與伺服器通訊的 applet，如何透過 servlet 居間協調達成作業。本章資訊在第二版已更新過。

### 第十一章《Servlet 之間的分工合作》

討論為何 servlet 需要與其它 servlet 通訊，以及它們之間如何藉由共享資訊來分工合作。本章在第二版已全面改寫過。

### 第十二章《Enterprise Servlets 和 J2EE》

探討企業網站所需的 servlet 先進功能：負載均攤 (load balancing) 與 J2EE 元件整合。這是第二版的全新章節。

### 第十三章《國際化》

展示 servlet 如何讀取、產生多國語文的內容。第二版解釋如何使用 javac 來管理編碼法，以及如何使用新的 API methods 來管理 locale (地域性)。

### 第十四章《Tea Framework》

展現 Tea Framework 這套既優雅又強悍的樣版引擎 (templating engine)。這是第二版的全新章節。

### 第十五章《WebMacro》

討論 WebMacro framework，它很像 Tea，但是有著不同的設計理念。這是第二版的全新章節。

### 第十六章《Element Construction Set》

簡略說明 ECS，這是一套以物件導向手法製作網頁的工具程式。第二版的全新章節。

### 第十七章 《XMLC》

介紹 XMLC 的概念，這是使用 XML 來製作網頁的解決方案。第二版的全新章節。

### 第十八章 《JavServer Pages》

JSP 是 Sun 公司的新技術，可自動將網頁編譯成 servlets。這是第二版的全新章節。

### 第十九章 《奇門遁甲》

有些實用的 servlet 範例與技巧，實在很難編排在其它章節，我將它們收錄在本章。第二版新收錄了一個 localized parameter parser、一個全新的 email class，更新了正規表示法 (regular expression) 章節的內容，並另闢兩個章節分別討論協力廠商所提供的工具，以及提昇效率的技巧。

### 第二十章 《Servlet 2.3 API 的變革》

闡述即將到來的 Servlet 2.3 API 之變革。2.3 版到目前 (2001 年六月) 尚未正式定案，但應該快了。這是第二版的全新章節。

### 附錄 A 《Servlet API 速查參考》

完整條列 javax.servlet 套件中的所有 class、method 與變數。第二版已經更新到 Servlet API 2.2。

### 附錄 B 《HTTP Servlet API 速查參考》

完整條列 javax.servlet.http 套件中的所有 class、method 與變數。第二版已經更新到 Servlet API 2.2。

### 附錄 C 《Deployment Descriptor DTD 參考》

描述並註解 web.xml deployment descriptor 的 Document Type Definition。這是第二版新添增的內容。

### 附錄 D 《HTTP 狀態碼》

列出 HTTP 所定義的所有狀態碼，以及供 servlet 所用的對應助憶常數 (mnemonic constants)。

### 附錄 E 《特殊字碼表》

列出 HTML 定義的所有字元代碼 (character entities)，以及等效的 Unicode escape value。

### 附錄 F 《字碼集》

當 servlet 需要產生多種不同語言的內容時，所建議使用的字碼集。

讀者可依自己喜歡的順序閱讀本書。依照章節順序從頭讀到尾，當然是最紮實的方式，而且可以避免看不懂往前查的情況。如果你想跳著看，倒也無妨，特別是第五章之後的內容，本來就彼此獨立。最後的良心建議，倘若你發覺任何有程式碼不按照你所想像的方式工作，請參考第十九章的《19.7 除錯》一節。

## 本書印刷體裁

粗體字 (BOLD) 代表：

- 檔名、路徑名稱、程式名稱。
- 第一次出現的術語，或是術語的定義。

定寬字 (Constant Width) 代表：

- 任何在 Java 程式碼中所出現的字樣，包括關鍵字、資料型態、常數、方法名稱、變數、類別名稱與界面名稱。
- 所有表列出來的 Java 程式碼。
- HTML 文件、tag 與屬性。

斜體字 (Italic) 代表：

- 在你的程式中應該要被代換成的實際值的項目。

粗體定寬字 (Constant width bold) 代表：

- 用鍵盤在命令列敲入的字樣。(以便與電腦的輸出有所區別)

## 歡迎批評指教！

我們非常歡迎讀者提供任何批評指教，幫助我們改善本書的下一版本。若讀者發現本書內容上有任何錯誤、錯字、不夠精確、缺陷、誤導、或語焉不詳，請務必寫 email 到敝公司專設的 [bookquestions@oreilly.com.tw](mailto:bookquestions@oreilly.com.tw) 信箱。任何指教經過查證後，我們會公佈在本書中文網頁 <http://www.oreilly.com.tw/chinese/java/servlet2.htm> 的勘誤表，並於下一刷時改正。在您回報錯誤之前，也請您先看看該錯誤是不是已經有人發現了。

除了內容上的指正之外，我們也非常希望能知道我們可以怎樣讓這本書更有用，像是該添增哪些新章節，或是範例程式的舉例方向 ... 等等，我們會認真研議您提出的建言，作為下一版的改版參考。