

IBM WebSphere Commerce



管理手冊

5.5 版

IBM WebSphere Commerce



管理手冊

5.5 版

注意事項：

在使用本資訊及其支援的產品之前，請確定先閱讀第 177 頁的『注意事項』中的資訊。

第一版（2003 年 6 月）

本版本適用於 IBM WebSphere Commerce 5.5（產品號碼 5724-A18）以及所有後續版次與修訂，除非新版中另有指示。請確定您使用的是產品層次的正確版本。

請向 IBM 業務代表或向當地的 IBM 分公司訂購出版品。

IBM 歡迎您提供意見。請利用線上的 IBM WebSphere Commerce 出版品讀者意見調查表提出您的建議，此調查表的 URL 如下：

<http://www.ibm.com/software/webservers/commerce/rcf.html>

當您傳送資訊給 IBM 時，即授與 IBM 非獨占的資訊使用或公佈權利，IBM 不需對您負任何責任。

© Copyright International Business Machines Corporation 2003. All rights reserved.

目錄

關於本書	v
資訊來源	v
WebSphere Commerce 書籍	v
WebSphere Commerce 線上說明	v
Web 中的其他資訊	vi
開始之前	vi
基本知識要求	vii
本書中的使用慣例	vii
路徑變數	vii
第 1 章 WebSphere Commerce 管理簡介	1
管理工具	2
管理主控台	2
配置管理程式	2
WebSphere Commerce Payments	3
組織管理主控台	9
WebSphere Commerce Accelerator	9
WebSphere Application Server	9
資料庫	10
第 2 章 基本管理作業	11
僅一次管理作業	11
安裝與配置	11
日常管理作業	11
啟動及停止 WebSphere Commerce	11
啟動 WebSphere Commerce 配置管理程式	12
建立、配置及刪除實例	14
開啓管理主控台	16
開啓組織管理主控台	16
啓用及停用 WebSphere Commerce 元件	17
啟動及停止 WebSphere Commerce Payments	18
安裝手冊中的管理作業	19
第 3 章 配置	21
配置服務	21
架構配置	21
傳訊服務	21
排定服務排程	21
日誌記載服務程式	22
WebSphere Commerce Payments 問題判定	24
規則服務	25
LDAP -- 輕裝備目錄存取通訊協定	25
暫置作業伺服器	32
連線服務	54
配置作業	57
配置傳訊服務	57
配置排程服務	57
配置日誌記載	58
檢視日誌檔案	62
配置 LDAP	64

配置規則服務	64
配置暫置作業伺服器	64
配置連線	69
配置 WebSphere Commerce Payments	69
第 4 章 代管型商店管理	71
管理轉銷商	71
管理配銷商	71
網站管理	72
第 5 章 動態快取	73
WebSphere Commerce servlet 或 JSP 結果快取	73
WebSphere Application Server 動態快取	73
動態快取	73
啓用動態快取服務與 servlet 快取	74
啓用動態快取監視器	74
WebSphere 動態快取監視器的相依關係 ID 樹狀檢視畫面	74
配置可快取的物件	75
快取預設值	76
整頁與片段快取	76
整頁快取	77
片段快取	77
快取 servlet 過濾	78
商店頁面快取策略	79
WebSphere Commerce 中的快取作廢機制	79
在 cachespec.xml 中定義作廢原則	80
透過快取監視器移除快取項目	82
啓動動態快取作廢 API	83
針對指令型作廢撰寫指令	83
WebSphere Commerce 中的使用指令作廢範例	83
快取作廢範例	84
快取作廢 API (DynaCacheInvalidation)	87
CACHEIVL 表格	87
外部快取作廢	87
Web 伺服器外掛程式快取	87
簡單的檔案 servlet	88
問題判定	89
第 6 章 系統維護	91
資料庫清理公用程式	91
「資料庫清理公用程式」物件	92
資料庫清理指令	104
清理資料庫	108
設定 WebSphere Commerce 公用程式的 PATH 環境變數	109
刪除物件	110
刪除帳戶物件	110
刪除地址物件	110
刪除約定交付庫存物件	111
刪除附件物件	111

刪除拍賣物件	112
刪除拍賣日誌物件	113
刪除自動出價日誌物件	113
刪除基本項目物件	114
刪除出價日誌物件	114
刪除 cachlog 物件	115
刪除計算代碼物件	116
刪除活動物件	116
刪除活動統計物件	117
刪除型錄項目物件	117
刪除型錄群組物件	118
刪除合約物件	118
刪除優待券物件	119
刪除預期的庫存記錄物件	120
刪除有關預期的庫存記錄明細物件	120
刪除討論區訊息物件	121
刪除供貨中心物件	121
刪除庫存代碼調整物件	122
刪除庫存調整物件	122
刪除指定的項目資訊物件	123
刪除成員訊息關係物件	124
刪除訊息物件	124
刪除訂單物件	125
刪除組織物件	125
刪除「產品顧問」統計物件	126
刪除「產品對照」統計物件	127
刪除「產品探索」統計物件	127
刪除原則物件	128
刪除產品配套物件	128
刪除報價要求物件	129
刪除退貨項目物件	129
刪除退貨原因物件	130
刪除「銷售輔助」統計物件	131
刪除暫置物件	131
刪除商店物件	132
刪除使用者物件	132
刪除使用者流量日誌物件	133
刪除供應商物件	134
新增配置到「資料庫清理公用程式」	134
刪除物件範例	135
第 7 章 效能	139
使用 WebSphere Commerce PMI 模組監視效能	140
設定 PMI 模組	140
啟動 Tivoli Performance Viewer	141
WebSphere Commerce PMI 模組報告值	142
其他效能工具	143
維護排程器	143
LDAP 抄寫	144
在暫置作業伺服器上測試網站	144
配置暫置作業伺服器	144

建立自訂表格的觸發指令	144
配置遠端資料庫	144
複製資料到暫置作業資料庫	145
執行「暫置檢查」指令	145
將資料延伸至正式作業資料庫	146
複製檔案到正式作業伺服器	147
刪除暫置物件	147
WebSphere Commerce Payments 效能調整參數	148
管理 iSeries 上的 Payments 實例資料庫	149

第 8 章 疑難排解 151

CrossWorlds® 配接器	151
WebSphere Commerce Payments	152
調整高效能環境	152

附錄 A. LDAP 實務內容 155

LDAP 實務內容：LDAP 伺服器作為成員儲存庫	155
---------------------------	-----

附錄 B. LDAP 檔 157

ldapmap.dtd	157
ldapentry.xml	157

附錄 C. WebSphere Commerce Payments 指導教學 163

步驟 1：存取 WebSphere Commerce Payments 使用者介面	163
步驟 2：建立 WebSphere Commerce Payments 商家及授權使用卡匣	164
步驟 3：定義 WebSphere Commerce Payments 使用者	164
步驟 4：指定使用者職務	165
以商家管理者身份登入	166
步驟 5：建立帳戶	166
指定帳戶設定	167
步驟 6：建立卡別	168
步驟 7：使用「結帳範例」建立訂單	168
步驟 8. 核准訂單	169
從「訂單」頁面核准訂單	170
使用核准並自動請款功能核准訂單	170
步驟 9：請款	171
步驟 10：結清批次	171
步驟 11：發出退款	172
步驟 12：檢視批次總額	173

附錄 D. WebSphere Commerce Payments 的國家語言支援 (NLS) 資訊 . 175

NLS 提示和要訣	175
支援的語言和語言環境	175

注意事項 177

關於本書

本書主要對象為管理者，並且說明如何使用「管理主控台」以及其他管理工具來管理 WebSphere® Commerce 元件。

資訊來源


WebSphere Commerce 中所含的線上資訊和書籍資訊會說明完整的電子商務解決方案。此外，隨附於 WebSphere Commerce 中的軟體產品亦會提供進一步資訊，說明該軟體的特定特性和功能。這一節提供各類型資訊來源的快速總覽。

WebSphere Commerce 書籍

- *WebSphere Commerce 基礎入門* 5.5 版
- *WebSphere Commerce 程式設計手冊與指導教學* 5.5 版
- *WebSphere Commerce 安裝手冊* 5.5 版
- *WebSphere Commerce 快速入門* 5.5 版
- *WebSphere Commerce 附加軟體手冊* 5.5 版
- *WebSphere Commerce 移轉手冊* 5.5 版
- *WebSphere Commerce 商店程式開發手冊* 5.5 版
- *WebSphere Commerce 範例商店手冊* 5.5 版
- *WebSphere Commerce 管理手冊* 5.5 版
- *WebSphere Commerce Security Guide* 5.5 版
- *WebSphere Commerce Payments 程式設計手冊與參考* 5.5 版
- *WebSphere Commerce Payments OfflineCard Cassette 補充資料* 5.5 版
- *WebSphere Commerce Payments CustomOffline Cassette 補充資料* 5.5 版

如果需要這些出版品的更新資料，請造訪下列網址：

 Business Edition (http://www.ibm.com/software/webservers/commerce/wc_be/)

 Professional Edition
(http://www.ibm.com/software/webservers/commerce/wc_pe/)

WebSphere Commerce 線上說明

WebSphere Commerce 線上說明中含有可讓您用 Web 瀏覽器檢視的線上資訊。

您可以執行 Internet Explorer 5.5 版（或以上）Web 瀏覽器，並使用以下的網址來存取線上說明：

`https://host_name:8000/wchelp`

其中 `host_name` 是您的 WebSphere Commerce 機器的名稱。

Web 中的其他資訊

WebSphere Commerce 支援中心

如果要取得支援資訊，包括新聞群組、FAQ、技術需知、疑難排解資訊及下載項目，請造訪下列網址：

WebSphere Commerce 支援中心網頁 (<http://www.ibm.com/software/commerce/support/>)

WebSphere Commerce Technical Library 網頁
(<http://www.ibm.com/software/commerce/library/>)

軟體合作者

我們有許多軟體合作者會提供相關的產品與服務，增強 WebSphere Commerce 的功能。如果需要這些合作者的相關資訊，請造訪 Commerce Zone (<http://www.ibm.com/software/wsdd/zones/commerce/>)，並按 Software Developers 鏈結。

Redbooks™

如需更多進階技術資訊，請造訪 Redbooks 網站 (<http://www.ibm.com/redbooks>)，並搜尋 WebSphere Commerce。

開始之前

IBM® WebSphere Commerce 管理手冊提供了網站管理者或系統管理者在 WebSphere Commerce 網站中執行所需作業的相關資訊。本手冊稱這些職務為網站管理者職務。

重要事項

在本書中，您將看到平台特有的資訊。請確實遵循您平台的特有指示。

本書分成下列幾章：

WebSphere Commerce 管理簡介

基本管理作業

配置

代管型商店管理

動態快取

系統維護

效能

疑難排解

附錄 A：LDAP 實務內容

附錄 B：LDAP 檔案

附錄 C：WebSphere Commerce Payments 指導教學

附錄 D：WebSphere Commerce Payments 的國家語言支援 (NLS) 資訊

基本知識要求

本書適用於需瞭解如何設定和維護 WebSphere Commerce 網站的網站管理者或系統管理者。管理者必須具備下列幾個領域的知識：

- 電子商務解決方案架構
- 安裝和配置網路介面元件，如防火牆、DFS™、DNS、路由器，以及網路集線器
- 資料庫技術
- XML
- 商店商業程序

本書中的使用慣例

本書使用下列慣例：

粗體字型，表示圖形使用者介面 (GUI) 控制項，如欄位名稱、按鈕或功能表選項。

等寬字型，表示輸入必須完全相同的文字範例，以及目錄路徑。

斜體字型，用來強調並代表可換成您所要之值的變數。



表示可協助您完成作業的其餘資訊。

400 專指 WebSphere Commerce for IBM Eserver iSeries™ 400® (舊稱 AS/400®) 的資訊。

AIX 專指適用於 WebSphere Commerce for AIX® 的資訊。

Linux 專指適用於 WebSphere Commerce for Linux® 的資訊。

Solaris 專指適用於 WebSphere Commerce for Solaris 作業環境軟體的資訊。

Windows

專指適用於 WebSphere Commerce for Windows® 的資訊。

Professional 專指適用於 WebSphere Commerce Professional Edition 的資訊。





Business 專指適用於 WebSphere Commerce Business Edition 的資訊。

路徑變數

本手冊使用下列變數來代表目錄路徑：

WC_installdir

這是 WebSphere Commerce 的安裝目錄。下列是 WebSphere Commerce 在各種作業系統上的預設安裝目錄：

	<code>/usr/WebSphere/CommerceServer55</code>
	<code>/QIBM/ProdData/CommerceServer55</code>
	<code>/opt/WebSphere/CommerceServer55</code>
	<code>/opt/WebSphere/CommerceServer55</code>
	<code>C:\Program Files\WebSphere\CommerceServer55</code>






WC_userdir

這是 WebSphere Commerce 使用的所有資料所在的目錄，可以由使用者修改或視需求配置。這類資料的範例如 WebSphere Commerce 實例資訊。這個目錄只適用於 OS/400®。

WC_userdir 變數代表下列目錄：`/QIBM/UserData/CommerceServer55`

WAS_installdir

這是 WebSphere Application Server 的安裝目錄。下列是 WebSphere Application Server 在各種作業系統上的預設安裝目錄：

	<code>/usr/WebSphere/AppServer</code>
	<code>/QIBM/ProdData/WebAS5</code>
	<code>/opt/WebSphere/AppServer</code>
	<code>/opt/WebSphere/AppServer</code>
	<code>C:\Program Files\WebSphere\AppServer</code>

WAS_userdir

這是 WebSphere Application Server 使用的所有資料所在的目錄，可以由使用者修改或視需求配置。 `/QIBM/UserData/WebAS5/Base/WAS_instance_name`

第 1 章 WebSphere Commerce 管理簡介

本書著重於網站管理者所執行的常用作業。本書中所討論的網站管理者職務與許多公司都有的系統管理者職務同義，因此從這個觀點來看，將它稱為網站管理者。

網站管理者是 WebSphere Commerce 隨附的已定義預設職務之一。網站管理者通常負責安裝、配置及維護 WebSphere Commerce 和相關的軟硬體。網站管理者要回應系統警告、警示和錯誤，以及診斷和解決系統問題。這個職務通常要控制存取和授權（建立成員，並為成員指定適當職務）、管理網站、監視網站效能及管理負荷平衡作業。網站管理者也可能要負責建立和維護數個伺服器配置，供測試、暫置和正式作業等不同開發階段使用。這個職務也要處理重要的系統備份及解決效能問題。

如需其他 WebSphere Commerce 職務的詳細資訊，請參閱 WebSphere Commerce 正式作業線上說明。網站管理者執行下列區域中的作業：

存取管理

- 使用者
- 組織
- 職務
- 存取群組
- 原則
- 資源群組
- 動作群組

安全

- 帳戶原則
- 密碼原則
- 帳戶鎖定原則
- 安全檢查者

效能 監視系統效能

配置

- 傳輸
- 訊息類型
- 訊息日誌記載
- 追蹤
- 元件配置
- 排程器
- 登錄
- 關於本產品

WebSphere Commerce Payments

- 使用者
- 商家設定
- WebSphere Commerce Payments 設定
- 卡匣

- 追蹤
- 存取管理
- 定義網站的傳訊系統
- 延伸規則服務到正式作業伺服器中
- 配置 WebSphere Commerce Payments
- 配置和維護快取
- 配置排程器
- 配置日誌記載
- 維護資料庫
- 管理 WebSphere Application Server
- 管理 Web 伺服器
- 監視系統效能
- 確定網站的安全
- 疑難排解

管理工具

有一些工具可以讓「網站管理者」作業更容易執行。本節中會討論主要的工具。後續章節則涵蓋其他工具，以及可以利用它們來完成的作業。

管理主控台

「管理主控台」可讓「網站管理者」執行與網站配置以及商店配置相關的作業。如果需要使用「管理主控台」的詳細資訊，請參閱第 16 頁的『開啓管理主控台』。網站管理者可利用「管理主控台」執行的作業，包括下列幾項：

- 管理使用者、組織、職務與成員群組
- 管理存取控制
- 定義網站的傳輸和訊息類型
- 指定 WebSphere Commerce Payments 設定
- 啓用及停用 WebSphere Commerce 元件
- 排定網站要執行的工作
- 更新登錄元件
- 公佈商店紀錄

商店公佈公用程式

商店公佈公用程式包含在「管理主控台」中，可讓您根據 WebSphere Commerce 隨附的範例，快速建立商店紀錄。這支公用程式也可以讓您藉由在商店中啓用或停用某些功能（比方說，共同作業），來配置已公佈的商店。如需有關公佈商店的詳細資訊，請參閱 WebSphere Commerce 線上說明中的「公佈商店紀錄」一節。

配置管理程式

「配置管理程式」可用來建立和變更部署和執行 WebSphere Commerce 所需要的基礎架構。建立基礎架構包括取得資料庫、Web 伺服器和 WebSphere Commerce Payments 的位置相關資訊，以及配置這些應用程式和 WebSphere Commerce 搭配使用所需要的資訊。同時也包括決定機器的設定資訊，以及起始設定 WebSphere Commerce 應用程式本

身的任何必要資訊。如果需要使用「配置管理程式」的詳細資訊，請參閱第 12 頁的『啓動 WebSphere Commerce 配置管理程式』，或參閱 *WebSphere Commerce 安裝手冊*。

「配置管理程式」可讓「網站管理者」執行管理作業和配置作業，而不需要使用具有語法敏感性的配置檔。以下是一些「網站管理者」可以執行的功能：

- 建立或刪除 WebSphere Commerce 實例
- 變更 WebSphere Commerce 實例的配置設定
- 配置 Web 伺服器
- 配置資料庫來作為暫置作業伺服器
- 輸入 WebSphere Commerce Payments 的參數

WebSphere Commerce Payments

WebSphere Commerce Payments 是 WebSphere Commerce 的一個元件，可以為在網際網路上的商家提供安全的電子付款處理程序。WebSphere Commerce Payments 根據開放式的標準技術，和付款卡匣搭配，支援多重付款通訊協定，包括：

- VisaNet 是全球電信資料及付款處理程序的提供者，具有授權及結算付款的能力。
- BankServACH 是一個與「自動化清算機構網路 (ACH)」接合的 Payment Gateway，可支援線上電子支票付款。
- Paymentech，是線上信用卡付款的商家處理器，直接鏈結到 Visa 和 MasterCard。
- 針對 WebSphere Commerce Payments 撰寫的協力廠商付款卡匣。

簡單來說，WebSphere Commerce Payments 整合了 WebSphere Commerce，並提供類似收銀機的功能來管理付款程序。客戶完全不需要直接和 WebSphere Commerce Payments 接觸，因為 WebSphere Commerce Payments 隱身於網際網路商家的商店門面後面，負責接收付款並處理銀行及其他金融機構間的付款事宜。

註： IBM WebSphere Commerce Payments（此後稱為 WebSphere Commerce Payments）先前稱為 Payment Manager。從 3.1.3 版開始，付款應用程式更名為 WebSphere Commerce Payments，而且產品的參考資料已在本文件中全面變更。

涵蓋多重付款方法

WebSphere Commerce Payments 施行一種多重付款式的組織架構結構，可以提供彈性並且可以延伸的方式，讓商家接受多重付款方法。多重付款組織架構將付款管理（組織架構）與特定付款類型（軟體卡匣）分開，以便讓彼此能各自發展。

WebSphere Commerce Payments 提供一種外掛式結構，因此各種付款方法的軟體卡匣皆與付款組織架構連接。組織架構提供付款與接受付款時所需的同屬基礎功能（不論是採用何種付款方法）。

付款卡匣為一些軟體應用程式，其符合 WebSphere Commerce Payments 組織架構的資料往來與控制法則。每一個付款卡匣含有特定付款方法與通訊協定的施行。

卡匣可以是 IBM 或協力廠商付款系統施行者所撰寫。IBM 支援卡匣的開發，並可為有意撰寫自己之付款卡匣的開發者，提供詳盡的指導。如需有關卡匣開發的詳細資訊，請參閱 *WebSphere Commerce Payments Cassette Kit Programming Guide*，其下載網址為：<http://www.software.ibm.com/commerce/payments/download.html>

WebSphere Commerce Payments 使用者介面

您可以透過 WebSphere Commerce UI 或者獨立式的 UI，存取 Payments 功能。「付款管理者」和使用代管型 WebSphere Commerce Payments 的個別商家，都可以使用這兩種方法執行下列作業：

- 配置 WebSphere Commerce Payments
- 以付款中立方式執行下列例行性的付款處理作業：
 - 核准訂單
 - 請款
 - 結清批次
 - 發出退款
 - 檢視每日的批次總額

WebSphere Commerce Payments 使用者介面以瀏覽器為基礎，並可透過您 Web 瀏覽器的 Secure Sockets Layer 功能從遠端安全存取之。

WebSphere Commerce Payments 職務

WebSphere Commerce Payments 會強制執行職務，根據使用者的職務，為各個使用者呈現不同的檢視畫面，例如，「付款管理者」和「商家管理者」的不同角度。在商家組織中，WebSphere Commerce Payments 採用不同職務的主張，讓商家可以監督他們自己的使用者。例如，「辦事員」只能做核准訂單工作，而只有「商家管理者」或「付款管理者」才能修改與金融機構間的關係。

Payments 組織架構中有四種職務，這些職務分別和 WebSphere Commerce 中的對應職務相關。如果您是負責建立使用者和指定其職務的管理者，您需要瞭解其對應關係。當您在「WebSphere Commerce 組織管理主控台」中建立使用者時，您必須為那些使用者指定以下列出的 WebSphere Commerce 職務。之後使用者就會顯示在 Payments UI 中，您可以在此指定他們的對應 Payments 職務。下表顯示 Payments 使用者職務和對應的 WebSphere Commerce 職務：

表 1. 職務對映

Payments 職務	WebSphere Commerce 職務
付款管理者	網站管理者
商家管理者	網站管理者
監督者	業務或業務經理
辦事員	客戶服務主管

「付款管理者」和「商家管理者」兩者都可以管理 WebSphere Commerce Payments。「客戶服務主管」和「辦事員」都是財務職務。他們不負責管理 WebSphere Commerce Payments，但是會管理付款處理功能。下表說明每一個 Payments 職務的責任：

表 2. 職務責任

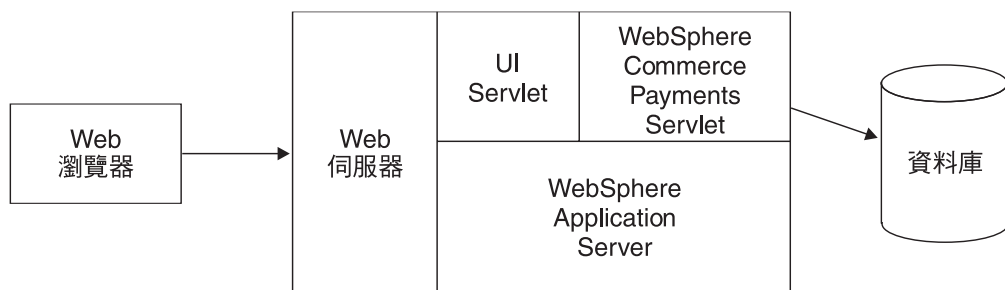
職務	責任
付款管理者	<ul style="list-style-type: none"> • 定義「商家管理者」、「客戶服務主管」和「辦事員」 • 配置商家和商家的卡匣 • 識別 Payments 主機名稱與狀態 • 配置任何已安裝的卡匣 • 新增、刪除以及更新事件接收程式 • 結清付款 • 核准或銷售訂單 • 發出退款以及沖銷退款 • 訂單請款 • 搜尋訂單與批次 • 檢視每日的批次總額
商家管理者	<ul style="list-style-type: none"> • 定義「商家管理者」、「客戶服務主管」和「辦事員」 • 配置商家和商家的卡匣 • 新增、刪除以及更新事件接收程式
監督者	<ul style="list-style-type: none"> • 結清付款 • 核准或銷售訂單 • 發出退款以及沖銷退款 • 訂單請款 • 搜尋訂單與批次 • 檢視每日的批次總額
辦事員	<ul style="list-style-type: none"> • 結清付款 • 核准或銷售訂單 • 訂單請款 • 搜尋訂單與批次 • 檢視每日的批次總額

WebSphere Commerce Payments 元件

如果要瞭解 WebSphere Commerce Payments，最好先看看下列 WebSphere Commerce Payments 以及定義其元件：

- 以 WebSphere Application Server 配置的 Web 伺服器
- WebSphere Commerce Payments Servlet
- 「使用者介面」servlet
- 資料庫

下圖說明 WebSphere Commerce Payments 的內部元件：



WebSphere Commerce Payments Servlet 是 WebSphere Commerce Payments 的主要元件。Servlet 的設計目的是要搭配使用 WebSphere Application Server - 此產品可提供所有平台皆通用的 Servlet 環境。

本節的其餘部分會進一步定義 WebSphere Commerce Payments Servlet 以及其輔助的支援產品：

- Web 伺服器
- WebSphere Application Server
- 資料庫

Web 伺服器: 何謂 Web 伺服器？一如本章前述，WebSphere Commerce Payments 會處理傳入的 HTTP 要求。一開始，HTTP 要求會先交由以 WebSphere Application Server 配置的 Web 伺服器應付。之後，再將這些要求轉交給其他 WebSphere Commerce Payments 元件。

IBM WebSphere Application Server: 為何要以 WebSphere Application Server 來配置您的 Web 伺服器？其原因有二：

1. WebSphere Application Server 是一項可在多種平台上使用各種不同的 Web 伺服器的產品。
2. WebSphere Application Server 提供一個可執行 WebSphere Commerce Payments Servlets 的環境。

WebSphere Commerce Payments Servlet: WebSphere Commerce Payments Servlet，也稱為 Payment Servlet，是 WebSphere Commerce Payments 的核心。它會接收來自使用者介面或其他商家應用程式的要求，並且在執行安全檢查後，呼叫「付款卡匣」來處理任何必要的付款處理程序 (如與金融網路的通訊)，最後再將回應傳回給呼叫應用程式。它是唯一可連接 WebSphere Commerce Payments 資料庫及存取其中所儲存的資料的元件。

WebSphere Commerce Payments 與電子商務實體

本節會說明 WebSphere Commerce Payments 如何支援商家的購物軟體。三種實務內容示範 WebSphere Commerce Payments 如何配合商家軟體以及同屬的付款通訊協定，與下列電子商務實體互動：

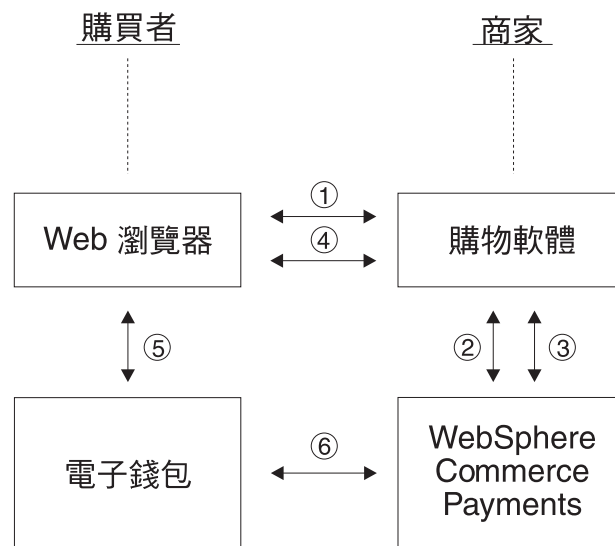
- 商家
- 購買者
- 金融機構

以下的實務內容說明電子商務使用 WebSphere Commerce Payments。特別要注意的是，這些實務內容為同屬的，而不代表某種特定付款通訊協定特有的互動。相反地，這些實務內容是探索常出現在眾多電子商務商業實務內容中的基本交易往來。

實務內容 1：現金與電子錢包： 第一個實務內容採用虛構的現金通訊協定，並涉及兩個實體：購買者與商家。資金會從購買者身上轉送到商家手中。在這項交易中，購買者使用通訊協定特有的電子錢包應用程式。此實務內容圖的交易步驟說明如下：

1. 購買者與商家購物軟體間的互動為購買者發出訂購要求給商家。
2. 在回應訂購要求時，商家的軟體傳送一支付款起始指令（亦即接受付款）給 WebSphere Commerce Payments。
3. WebSphere Commerce Payments 回應一則付款起始訊息，其中含有購買者的電子錢包程式的特定通訊協定資訊。
4. 商家軟體將這則付款起始訊息轉遞至購買者的瀏覽器。
5. 購買者的瀏覽器收到的付款起始訊息起始購買者的電子錢包程式。
6. 購買者透過 WebSphere Commerce Payments 與電子錢包軟體間的互動，完成交易。

實務內容 1： WebSphere Commerce Payments 與現金、電子錢包間的互動：



這個實務內容與下列兩個實務內容不同的地方在於，這個實務內容所示範的電子商務交易只牽涉到兩個實體：

- 購買者，以電子錢包代表
- 使用 WebSphere Commerce Payments 的商家

沒有涉及金融機構。

實務內容 2：信用卡與電子錢包： 第二個實務內容說明一項使用虛構信用卡通訊協定的訂購，其中涉及三種實體：

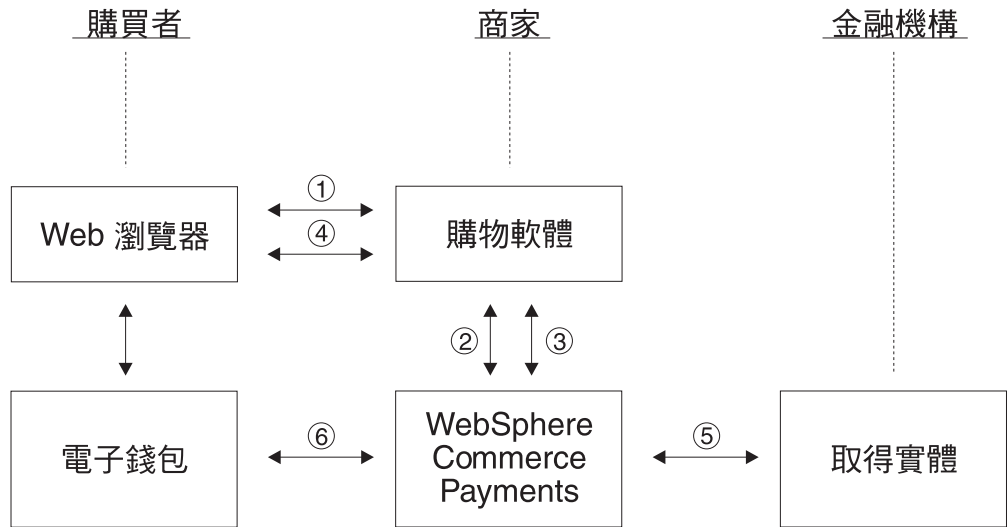
- 商家
- 購買者，同樣使用電子錢包
- 金融機構

這個實務內容流程和第一個實務內容相似，唯一的差異如下：

- 當購買者的電子錢包完成訂購要求時，WebSphere Commerce Payments 會發出核准要求給金融機構的收單機構，以確定持卡人足夠的資金。
- 交易在 WebSphere Commerce Payments 傳送一則確認回應給購買者的電子錢包，指出核准狀態而結束。

下列實務內容圖說明步驟 5 與 6 中的此項差異：

實務內容 2： WebSphere Commerce Payments 與信用卡、電子錢包間的互動：

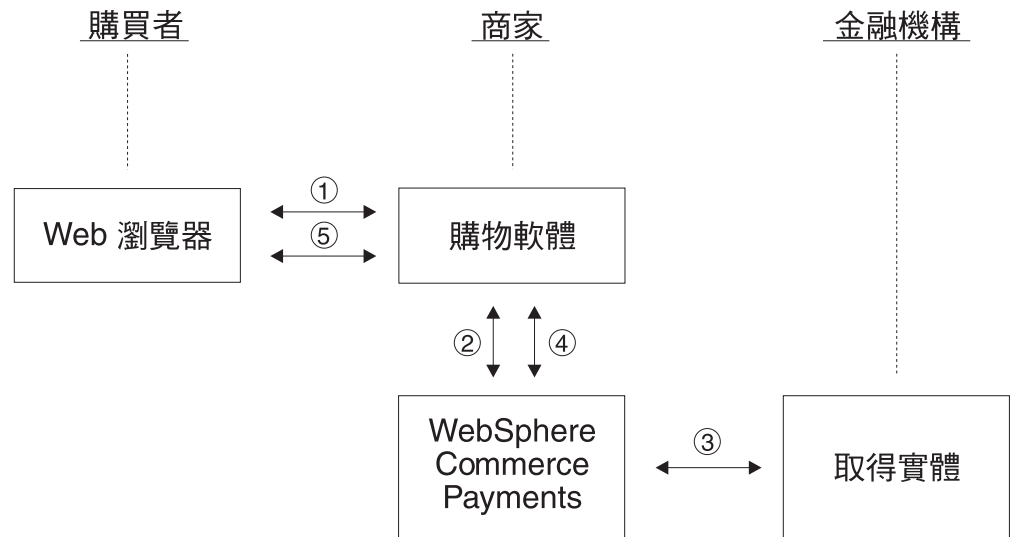


當客戶的電子錢包程式已起始並且交易完成後，金融機構的收單機構會確認持卡人的資金足夠，而 WebSphere Commerce Payments 會向購買者確認請款事宜。

實務內容 3：信用卡與 SSL： 最後一個實務內容說明信用卡通訊協定，其中購買者並無電子錢包或消費者專用軟體。所有消費者資訊都是以安全方式（像是採 SSL Socket）傳給商家軟體。此實務內容圖的交易步驟說明如下：

1. 購買者與商家購物軟體間的互動為購買者發出訂購要求給商家。
2. 在回應訂購要求時，商家軟體傳送一支付款起始指令給 WebSphere Commerce Payments。
3. WebSphere Commerce Payments 嘗試經由收單機構的金融機構確認持卡人的資金充足，來核准交易。
4. WebSphere Commerce Payments 傳回結果（像是已經請款）給商家軟體。
5. 最後，通知購買者的訂購要求成功或失敗。

實務內容 3： WebSphere Commerce Payments 與信用卡、SSL 間的互動：



上圖說明可能的電子商務實務內容，並示範 WebSphere Commerce Payments 在這些實務內容中所充當的職務。還有許多其他可能的商業流程，而 WebSphere Commerce Payments 在這些互動中將擔任類似的付款功能。

組織管理主控台

「組織管理主控台」可讓「網站管理者」控制存取您的網站或商店的組織。授權讓您執行的作業會顯示在「組織管理主控台」首頁的多個功能表中。這些作業是根據使用者職務與權限層次而來。如需詳細資訊，請參閱 WebSphere Commerce 線上說明中的「組織管理主控台」資訊。

WebSphere Commerce Accelerator

從管理您的商店外觀，到建立及維護訂單以追蹤商店活動，您都可以利用 WebSphere Commerce Accelerator 來完成各種不同的商店營運，以維護您的線上商店。WebSphere Commerce Accelerator 首頁功能表中會顯示您的職務有權執行的作業。這些作業都是根據使用者職務和權限層次而定，而使用者職務和權限層次是由網站管理者使用管理主控台定義。如需詳細資訊，請參閱 WebSphere Commerce 線上說明中的 WebSphere Commerce Accelerator 資訊。

註： 線上說明中會說明「網站管理者」使用「組織管理主控台」或 WebSphere Commerce Accelerator 所執行的作業。

WebSphere Application Server

WebSphere Application Server (WAS) 是一個 Java 型的應用程式環境，可用來建置、部署及管理網際網路和企業內部網路的 Web 應用程式。WebSphere Application Server 也提供 Java™ Server Page 檔案的執行時期支援。

WebSphere Application Server 的管理主控台有各種工具和特性，管理者可以利用它們來執行下列動作：

- 停止和啓動 WebSphere Commerce 伺服器
- 停止和啓動 IBM WebSphere Commerce Payments
- 安裝和配置資源（比方說，Servlet 和 Enterprise Bean）

- 指定資源的安全
- 確定所有應用程式都可以使用
- 授與或取消使用者存取權（比方說，建立新員工的新帳號，以及將人員加入特定的使用者群組中）
- 監視 Commerce Server 效能
- 複製應用程式元件以改進效能
- 提供執行 WebSphere 應用程式的追蹤和除錯資訊

WebSphere Application Server 資訊可以在這個 WebSphere Application Server 網站取得：<http://www.ibm.com/software/webservers/appserv>。如果需要 WebSphere Application Server 系統管理的其餘詳細資訊，請參閱下列文件：

- 系統管理
- 調整及移難排解

資料庫

您可以使用 DB2[®] 或 Oracle 作為您的資料庫管理系統。如果需要詳細資訊，請參閱每一個系統適用的 *WebSphere Commerce 安裝手冊*。

第 2 章 基本管理作業

本章涵蓋了日常執行及僅執行一次所需的基本管理作業。您不一定需要執行每一項作業，視您的商業程序而定。

僅一次管理作業

下列是通常僅執行一次的作業。

安裝與配置

「網站管理者」必須做的其中一項作業，就是安裝以及配置 WebSphere Commerce 網站。通常，這是您必需執行的第一個步驟。安裝作業通常包括根據您使用的作業系統，來安裝 WebSphere Commerce 軟體。安裝步驟可以在產品隨附的安裝手冊中找到。如果需要安裝及配置的詳細資訊，請參閱 *WebSphere Commerce 安裝*。進一步的配置資訊可以在 *WebSphere Commerce 線上說明* 中找到。

日常管理作業

下列是日常執行的作業：

- 啟動及停止 WebSphere Commerce
- 開啟配置管理程式
- 建立、配置及刪除 WebSphere Commerce 實例
- 開啟「管理主控台」
- 開啟「組織管理主控台」
- 啟用及停用 WebSphere Commerce 元件
- 啟動及停止 WebSphere Commerce Payments

啟動及停止 WebSphere Commerce


WebSphere Commerce 的每一個實例都含有自己的 WebSphere Commerce Server。WebSphere Commerce Server 是一種 Java 型商務伺服器，用來控制 WebSphere Commerce 系統中的資訊流程。如需有關 WebSphere Commerce Server 中的每一個元件的詳細資訊，請參閱 WebSphere Commerce 正式作業與開發作業線上說明的 WebSphere Commerce Server。

WebSphere Commerce 實例是以一個名稱爲 `WC_instance_name` 的應用程式伺服器啟動和停止，其中的 `instance_name` 是您要啟動和停止的 WebSphere Commerce 實例名稱。例如，若要啟動一個名稱爲 `demo` 的 WebSphere Commerce 實例，您需要啟動名稱爲 `WC_demo` 的應用程式伺服器。

如要啟動或停止 WebSphere Commerce 實例，請執行下列步驟：

1. 決定要啟動或停止哪一個 WebSphere Commerce 實例。
2. 遵循停止和啟動您的平台上的應用程式伺服器的指示。如需有關這些指示的詳細資訊，請參閱 WebSphere Commerce 正式作業與開發作業線上說明中的「在.....<您的平台> 上啟動和停止應用程式伺服器」。

如果 WebSphere Commerce Payments 尚未啟動，您可以在啟動 WebSphere Commerce 之後，將其啟動。如需啟動 WebSphere Commerce Payments 的指示，請參閱第 18 頁的『啟動及停止 WebSphere Commerce Payments』。

註:  如需有關在 WebSphere Commerce Studio 中啟動 WebSphere Commerce 的指示，請參閱 WebSphere Commerce 正式作業線上說明的「在 WebSphere Commerce Studio 中啟動及停止 WebSphere Commerce」。

啟動 WebSphere Commerce 配置管理程式

「配置管理程式」可讓您啟用及配置 WebSphere Commerce 的各種元件。只要是可能的話，應該要透過「配置管理程式」來啟用元件，而不要直接修改 WebSphere Commerce 配置檔。如要建立或配置 WebSphere Commerce 實例，您應該使用「配置管理程式」。

在 AIX、Linux 以及 Solaris 軟體上啟動 WebSphere Commerce 配置管理程式

若要在 AIX、Linux 以及 Solaris 軟體上啟動 WebSphere Commerce 配置管理程式，請執行下列步驟：

1. 執行下列步驟，啟動伺服器：
 - a. 開啓終端機視窗。
 - b. 使用安裝 WebSphere Commerce 時在執行 postinstall script 時所建立的 WebSphere Application Server 使用者登入。預設的 WebSphere Application Server 使用者名稱爲 *wasuser*。
 - c. 發出下列指令：

```
export DISPLAY=host_name:0.0
xhost +host_name
cd /usr/WebSphere/CommerceServer55/bin
./config_server.sh
```



```
export DISPLAY=host_name:0.0
xhost +host_name
cd /opt/WebSphere/CommerceServer55/bin
./config_server.sh
```

其中的 *host_name* 是您要存取「配置管理程式」所在機器的完整主機名稱。

註:

- 1) 請勿關閉您輸入 `config_server.sh` 指令所在的終端機視窗，否則將會停止「配置管理程式」伺服器。
 - 2) 請勿將「配置管理程式」伺服器當作背景處理程序執行 - 這可能會有潛在的安全危機。
2. 執行下列步驟，啟動用戶端：
 - a. 開啓另一個終端機視窗。
 - b. 使用安裝 WebSphere Commerce 時在執行 postinstall script 時所建立的 WebSphere Application Server 使用者登入。預設的 WebSphere Application Server 使用者名稱爲 *wasuser*。

- c. 發出下列指令：

► AIX ► Solaris

```
export DISPLAY=host_name:0.0
xhost +host_name
cd /usr/WebSphere/CommerceServer55/bin
./config_client.sh &
```

► Linux

```
export DISPLAY=host_name:0.0
xhost +host_name
cd /opt/WebSphere/CommerceServer55/bin
./config_client.sh &
```

其中的 *host_name* 是您要存取「配置管理程式」所在機器的完整主機名稱。

在 iSeries 上啟動 WebSphere Commerce 配置管理程式

若要在 iSeries 上啟動 WebSphere Commerce 配置管理程式，請執行下列步驟：

1. 執行下列步驟，啟動伺服器：

- 登入 iSeries，請確定設定檔具備 *SECOFR 使用者類別，並且以英文或您選擇作為實例預設語言的語言之特定設定值設置而成。
- 輸入下列指令，啟動一個 QShell 階段作業：

```
STRQSH
```

並在 QShell 階段作業中執行下列動作：

- 1) 發出下列指令，切換到 WebSphere Commerce 伺服器的 bin 目錄：

```
cd /qibm/proddata/commerceserver55/bin
```

- 2) 發出下列指令，啟動配置管理程式伺服器程式：

```
config_server.sh -port server_port_number
```

其中的 *server_port_number* 是位在 iSeries 伺服器上，「配置管理程式」會接聽的埠號。此參數為選用的，預設值為 1099。這個值必須介於 1024 和 65535 之間，而且不能正在使用中。

註：如果您使用的系統中，主要語言和您建立案例時所用的語言不同，您必須在您的使用者設定檔的檔案庫清單中加入 QSYSlanguage_feature_number 檔案庫。否則設定檔會試圖在 QSYS 下面尋找。如果要加入語言特性檔案庫，請使用 EDTLIBL 指令。

- c. 第一次在系統中執行「配置管理程式」時，您會看到下列訊息：

```
正在將 Java 程式連結到 /QIBM/ProdData/WebCommerce55/lib/wcsconfig.JAR。
正在將 Java 程式連結到 /QIBM/ProdData/WebCommerce55/lib/wcsruntime.JAR。
正在將 Java 程式連結到 /QIBM/ProdData/WebCommerce55/wc.ear/lib
/wcslogging.JAR。
正在將 Java 程式連結到 /QIBM/ProdData/WebCommerce55/lib/xml4j.JAR。
正在將 Java 程式連結到 /QIBM/ProdData/WebCommerce55/lib/sslite.ZIP。
```

- d. 當您收到下列訊息時，請繼續進行下一個步驟：

```
已建立登錄。
CMServer 已連結於登錄中。
```

2. 執行下列步驟，啟動用戶端：

- 在用戶端機器中使用指令提示，切換至 WCS400 目錄。

- b. 執行下列指令以配置用戶端：

```
config_client.bat iSeries_Host_name Server_port_number
```

其中

iSeries_Host_name

是伺服器的完整主機名稱。

Server_port_number

是 iSeries 伺服器上，「配置管理程式」會接聽的埠號。

在 Windows 上啟動 WebSphere Commerce 配置管理程式

若要在 Windows 上啟動 WebSphere Commerce 配置管理程式，請執行下列步驟：

1. 選取開始 → 設定 → 控制台 → 系統管理工具 → 服務，並檢查 IBM WC 「配置管理程式」服務的狀態為「已啟動」，確定 IBM WC 「配置管理程式」伺服器在執行中。

重要事項

如果讓 IBM WC 「配置管理程式」伺服器服務維持執行，可能會有潛在的安全問題。因此，當您不使用「配置管理程式」時，請停止「WC 配置管理程式」伺服器服務。

如果要避免潛在的安全問題，您也要確定 IBM WC 「配置管理程式」伺服器是設定為手動啟動，而非自動啟動。

2. 選取開始 → IBM WebSphere Commerce → 配置。

建立、配置及刪除實例

如要建立、配置或刪除 WebSphere Commerce 實例，您應該使用「配置管理程式」。設計這個工具的目的在於可讓您在配置實例的同時，防止您輸入無效值。

「配置管理程式」應該用來修改大部份的 WebSphere Commerce 配置參數。不過，部份參數的變更必須透過「WebSphere Commerce 配置檔」(*instance_name.xml*) 進行。請參閱線上說明中個別元件的文件，來判定應該透過「配置管理程式」或透過 WebSphere Commerce 配置檔 (*instance_name.xml*) 進行配置。

建立及配置實例

如果要建立實例，請執行下列步驟：

1. 開啓「配置管理程式」。如果需要開啓「配置管理程式」的指示，請參閱第 12 頁的『啟動 WebSphere Commerce 配置管理程式』。
2. 展開具有您機器名稱的節點、選取實例清單，然後從「動作」功能表選取建立實例。
3. 實例建立精靈將引導您完成實例建立程序。在任何畫面上，如果您需要協助，請選取說明。

如果要配置實例，請執行下列步驟：

1. 開啓「配置管理程式」。
2. 展開具有您機器名稱的節點、展開實例清單，然後選取要修改的實例。

3. 選取要修改的元件
4. 如果這個元件已存在，您可以選取它，然後將開啓適當的畫面。接著，您可以根據需要來修改元件。
5. 如果元件不存在，您可以用滑鼠右鍵按一下元件，然後選取要建立元件。有一個精靈將引導您完成元件建立程序。

刪除實例

如要刪除 WebSphere Commerce 實例，請執行下列步驟：

1. 備份您的 WebSphere Commerce 實例（建議做法）。
2. 切換至 `WC_installdir\instances\instance_name\xml` 目錄 400
`WC_userdir/instances/instance_name/xml`，然後將 `instance_name.xml` 檔案複製到備份目錄中。
3. 啓動 WebSphere Application Server。
4. 執行下列步驟，移除 WebSphere Commerce 伺服器：

- 400 從 QSHELL 階段作業指令行中，鍵入：

```
WC_installdir/bin/rmCommerceServer.sh instance_name
```

- AIX Solaris Linux 從指令提示鍵入：

```
WC_userdir/bin
./rmCommerceServer.sh instance_name
```

- Windows 從指令提示中，鍵入：

```
WC_installdir/bin
./rmCommerceServer instance_name
```

5. 在「WebSphere Commerce 配置管理程式」中以滑鼠右鍵按一下您的實例，然後選取刪除。
6. 除去您的 WebSphere Commerce Business Edition 資料庫：

- DB2 如果您所用的是 DB2 資料庫，請執行下列步驟：
 - a. 開啓「DB2 指令視窗」，再鍵入下列指令：

```
db2 drop db db_name
db2 uncatalog node node_name
```

其中 `db_name` 是您在「配置管理程式」中輸入的 WebSphere Commerce 資料庫的名稱，`node_name` 則是您為這些資料庫編目的節點名稱。預設節點名稱是主機名稱的短式名稱（不完整的名稱）。

- Oracle 如果您使用 Oracle 資料庫，請執行下列步驟：

- a. 啓動 SQL*Plus 階段作業，然後鍵入下列指令：

```
sqlplus system/manager@wc_SID
```

其中 `wc_SID` 是 WebSphere Commerce 資料庫的「Oracle 系統識別碼」(SID)。

- b. 在 SQL*Plus 階段作業中發出下列指令：

```
DROP TABLESPACE wc_tablespace_name INCLUDING CONTENTS;
DROP USER wc_Oracle_user_ID CASCADE;
EXIT;
```

註: 您也必須移除實際的表格空間檔案。表格空間檔案稱為 `wc_SID.ora`，其中 `wc_SID` 是 WebSphere Commerce 資料庫的「Oracle 系統識別碼」(SID)。

如果是 **400** 執行下列 SQL 陳述式，刪除實例資料庫：`drop schema instance_name`。如果您嘗試刪除綱目時，收到訊息指出不可以刪除已連接的接收者，請改用下列指令：`ENDJRNP FILE(*ALL) JRN(instance_name/QSQJRN)`。

7. 如果要完整地刪除 WebSphere Commerce 實例，請刪除下列目錄：
`WC_installdir/instances/實例名稱目錄`。**400** `WC_userdir/instances/實例名稱`。

開啓管理主控台

每當您想要執行管理作業時，將可能需要在「管理主控台」中執行它。「管理主控台」可讓您藉由完成管理作業及配置作業，來控制您的網站或商店。如果需要「管理主控台」的詳細資訊，請參閱第 2 頁的『管理主控台』。

1. 執行下列動作之一來啓動「管理主控台」：

- 在您的瀏覽器中存取下列 URL：
`https://host_name:8002/adminconsole`

其中 `host_name` 是您的 WebSphere Commerce Server 的完整 TCP/IP 名稱。

- **Studio** 如果您有安裝 WebSphere Commerce Studio，並且希望從這部機器開啓「管理主控台」，請執行下列步驟：
 - a. 確定已經在 WebSphere Commerce Studio 中啓動 WebSphere Commerce。如需有關在 WebSphere Commerce Studio 中啓動 WebSphere Commerce 的指示，請參閱 WebSphere Commerce 正式作業線上說明的「在 WebSphere Commerce Studio 中啓動及停止 WebSphere Commerce」。
 - b. 在您的瀏覽器中存取下列網址：

`https://host_name/webapp/wcs/tools/servlet/ToolsLogon?XMLFile=adminconsole.AdminConsoleLogon`

其中 `host_name` 是您的 WebSphere Commerce Studio 機器的完整 HTTP 主機名稱。

2. 執行下列動作，登入「管理主控台」：

- a. 從「登入」頁面中，鍵入您的「管理主控台」登入使用者名稱與密碼，然後按一下**登入**。如果您有權使用一個以上的商店或語言，則這時會顯示「管理主控台網站/商店選擇」頁面。如果您已經被授權使用一個商店和語言，就會顯示「管理主控台」首頁。您有權執行的作業會顯示在「管理主控台」首頁中。

3. 從「管理主控台網站/商店」選擇頁面中，選取**網站**或**商店**圓鈕，然後按一下**確定**。
 - 如果您選取了**網站**，這時會顯示「網站管理主控台」首頁。
 - 如果選取了**商店**，您可以從**名稱**和**語言**下拉框選擇商店名稱及語言。這時會顯示「管理主控台」首頁。

開啓組織管理主控台

執行下列動作之一，登入「組織管理主控台」：

- 執行下列動作之一，啓動「組織管理主控台」：
 - 在您的瀏覽器中存取下列 URL：`https://host_name:8004/orgadminconsole`

其中 *host_name* 是您的 WebSphere Commerce Server 的完整 TCP/IP 名稱。

- 如果您有安裝 WebSphere Commerce Studio，並且希望從這部機器開啓「組織管理主控台」，請執行下列步驟：
 1. 確定已經在 WebSphere Commerce Studio 中啓動 WebSphere Commerce。如需詳細資訊，請參閱 WebSphere Commerce 線上說明。
 2. 在您的瀏覽器中存取下列網址：

```
https://host_name/webapp/wcs/tools/servlet/ToolsLogon?XMLFile=buyerconsole.BuyAdminConsoleLogon
```

其中 *host_name* 是您的 WebSphere Commerce Studio 機器的完整 HTTP 主機名稱。
- 如果要登入「組織管理主控台」，請執行下列步驟：
 1. 從「登入」頁面中，鍵入您的「組織管理主控台」登入使用者名稱與密碼，然後按一下「登入」。如果您有權使用多個商店或語言，這時會顯示「組織管理主控台」首頁。您有權執行的作業會顯示在「組織管理主控台」首頁中。
- 如果要登入並變更您的「組織管理主控台」密碼，請執行下列步驟：
 1. 從「登入」頁面中，鍵入您的「組織管理主控台」登入使用者名稱與密碼。
 2. 選取**變更密碼**勾選框，然後按一下**登入**。這時會顯示「變更密碼」頁面。
 3. 在**舊密碼**欄位中，鍵入您目前的「組織管理主控台」登入密碼。這個欄位最多可以接受 128 個英數字元。
 4. 在**新密碼**欄位中，鍵入新的登入密碼。這個欄位最多可以接受 128 個英數字元。
 5. 在**新密碼確認**欄位中，重新鍵入新登入密碼。
 6. 按一下**變更**來儲存新密碼。這時會顯示「組織管理主控台」首頁。您有權執行的作業會顯示在「組織管理主控台」首頁中。

啓用及停用 WebSphere Commerce 元件

身為管理者，您可能想要讓 WebSphere Commerce 的某些元件僅能在特定時間使用。由於不同客戶有不同的需求，WebSphere Commerce 可讓您在必要時，使用管理主控台中的**配置**功能表，來啓用或停用 WebSphere Commerce 元件。

如要啓用或停用 WebSphere Commerce 元件，請執行下列步驟：

1. 開啓管理主控台，然後以網站管理者身分登入。
2. 從**配置**功能表中按一下**元件配置**。這時會顯示一個頁面，列出可供您網站使用的所有元件。
3. 如果要啓用元件，請執行下列步驟：
 - a. 從**可用元件**清單中選取要啓用的元件，然後按一下**新增**
 - b. 驗證所有要啓用的元件都在**選取的元件**清單中，然後按一下**確定**。

註：在啓用任何元件之前，請先參閱有關該元件的文件，以確定已適當地配置了它。

4. 如果要停用元件，請執行下列步驟：
 - a. 從**選取的元件**清單選取要停用的元件，然後按一下**移除**。
 - b. 驗證所有要停用的元件都在**可用元件**清單中，然後按一下**確定**。

啓動及停止 WebSphere Commerce Payments

WebSphere Commerce Payments 是一個適用於線上商家、與通訊協定無關的付款交易伺服器。它為網站提供類似現金登錄的功能，支援採用特定通訊協定專用卡匣的多重付款方法。這些卡匣是可連接到 Payments 組織架構的軟體元件，負責將通用付款和管理指令解譯成特定付款通訊協定的要求，之後，這些要求會轉遞給適當的接受者，如「收單機構」的付款閘道。其最後的結果如同收帳員在傳統商店的結帳櫃台刷付款卡。Payments 會代表商家（或商家群組，如商店群組）來處理網際網路付款的所有背景細節，並提供一個圖形介面來簡化交易管理作業。Payments 會執行下列功能：

- 啓動時，它會根據卡匣來驗證所有必要的「收單機構」憑證都已存在。如果遺漏了憑證，將自動向「收單機構」或 Payment Gateway 要求它。
- 它會永久地執行，以接聽專用埠並處理所有從電子錢包及商家伺服器進入的要求。
- 它會使用 TCP/IP 通訊協定或其他通訊協定（視卡匣而定），透過網際網路與「收單機構」通訊。
- 它會使用資料庫表格來維護交易狀態、核准要求及請款要求的相關資訊。這些表格中的記錄會基於工作目的，以及基於追蹤及記錄保存之目的而被保存下來。資料庫也含有配置表格，以保留下列資訊：商家、付款卡的卡別，以及提供卡片授權和付款之扣款服務給商家的「收單機構」。

商家伺服器係使用 Payments API 與 Payments 互動。設計這些 API 的目的，在於能夠足以支援「安全電子交易」(SET™) 以及其他付款技術。

身為管理者，您將被要求啓動及停止 Payments 服務。啓動及停止付款引擎需要您在安裝了 Payments 的目錄中具有 .payment 檔。 .payment 檔是在安裝 Payments 期間建立的。

啓動 WebSphere Commerce Payments

如要啓動 WebSphere Commerce Payments，請執行下列步驟：

1. 啓動 WebSphere Application Server 的「管理主控台」，並確定 WebSphere Payments Application 伺服器已經啓動。

2. 按下列方式登入：



    以資料庫實例 ID 登入。

 以具有「管理者」的群組專用權的使用者登入。

3. 移至安裝了 WebSphere Commerce Payments 的目錄。預設安裝位置如下：

 /usr/lpp/WebSphere/CommerceServer55/payments

 /QIBM/ProdData/CommercePayments/V55

  /opt/WebSphere/CommerceServer55/payments

 drive:\Program Files\IBM\WebSphere Commerce Payments

4. 輸入下列指令：

   ./IBMPayServer

 IBMPayServer

 執行這個指令之後，您必須要指定一個密碼。

註：如果您在配置 WebSphere Commerce Payments 時選取**需要密碼**，您就必須啓動 IBMPayServer 指令，才能完整起始設定 WebSphere Commerce Payments 應用程式。

IBMPayServer 指令是用來指定 WebSphere Commerce Payments 密碼，此密碼是用來將儲存在資料庫中的任何敏感資料解密。如需這個指令的詳細資訊，請參閱 WebSphere Commerce 正式作業與開發作業線上說明的「發出 IBMPayServer 指令」一節。如需有關配置 WebSphere Commerce Payments 的詳細資訊，請參閱第 69 頁的『配置 WebSphere Commerce Payments』。如需有關配置 WebSphere Commerce Payments 的詳細資訊，請參閱第 163 頁的附錄 C, 『WebSphere Commerce Payments 指導教學』。

停止 WebSphere Commerce Payments

如果要停止 WebSphere Commerce Payments，您需要執行下列步驟：

- 停止 WebSphere Commerce Payments。
- 在 WebSphere Application Server 下停止 Payment Manager 應用程式伺服器。

如果要停止 WebSphere Commerce Payments，請完成第 18 頁的『啓動 WebSphere Commerce Payments』的步驟 1-3，然後對步驟 4 使用下列指令：

   ./StopIBMPayServer

 StopIBMPayServer

註：如需有關透過「配置管理程式」或 WebSphere Commerce Studio 停止 Commerce Payments 的資訊，請參閱 WebSphere Commerce 正式作業線上說明的「在 WebSphere Commerce Studio 啓動及停止 WebSphere Commerce Payments」以及「從配置管理程式啓動及停止 WebSphere Commerce Payments」。

安裝手冊中的管理作業

下列管理作業說明位於 *WebSphere Commerce 安裝手冊* 中：

- 修改 WebSphere Commerce 實例或 Commerce Payments 實例
- 刪除 WebSphere Commerce 實例
- 其他指令行配置作業
- 將遠端 DB2 資料庫編目
- 產生 WebSphere Commerce 加密密碼
- 產生 WebSphere Commerce Payments 加密密碼

第 3 章 配置

本章說明為確保 WebSphere Commerce 網站的最佳運作，「網站管理者」所需執行的配置工具與作業。

配置服務

架構配置

「網站管理者」負責與架構師合作，來實作一個運作 WebSphere Commerce 的配置。比方說，WebSphere Commerce 以及必要的堆疊（Web 伺服器、WAS、資料庫，以及 Commerce Payments）可能全都常駐在同一部實體機器或節點上，形成單一節點架構；或者 WebSphere Commerce 的各層（軟體產品）可能常駐在不同的實體節點（機器）上，而形成多節點架構。根據解決方案的複雜性，實際上可能會有一個叢集化環境，在這裡，不同機器或複本之間會實作負荷分擔。網站管理者也可能要負責建立數個伺服器配置，供測試、暫置和正式作業等不同開發階段使用。如果需要 WebSphere Commerce 可用之不同配置的詳細資訊，請參閱 *WebSphere Commerce 安裝手冊*。

傳訊服務

「管理主控台」中的傳訊服務介面可讓網站管理者設定和管理網站和商店的訊息遞送方式。WebSphere Commerce 傳訊系統可用來管理在定義和傳送 WebSphere Commerce 中產生的訊息的所有作業層面。傳訊系統可以透過如電子郵件及一般檔案之類的傳輸方式來傳送訊息。電子郵件所支援的離埠通訊協定是 SMTP，訊息編碼則根據指定的語言而定。一般檔案訊息會使用 UTF-8 編碼標準。另外，您也可以選擇性地配置傳訊系統，使其利用 WebSphere MQ 以及 IBM CrossWorlds® 傳送訊息至後端系統。傳訊服務也可以配置為在傳送離埠訊息之後，等候一段時間，以取得回覆訊息。若要配置離埠傳訊系統，您可以利用「管理主控台」中的「配置」功能表。

WebSphere Commerce 使用一種外掛程式模型實作 Java 2 Enterprise Edition Connector Architecture (J2EE/CA)，在系統和各種傳輸之間提供一個共同的介面。和外部傳輸使用共同介面的優點為傳輸的實作明細可以和傳訊系統的作業分開來。這種架構得以外掛符合 J2EE/CA 介面的其他傳輸。如此您可以便利地自訂解決方案，來配合您的環境和商業需求。

排定服務排程

排程器是 Commerc Server 的元件之一，主要用來依時間計畫，來排定工作和啟動工作。每個已排定的工作都會由個別的執行緒執行。您可以將多項工作排定為同時執行，不過，建議您將工作分散開來，以便控制工作量並避免閒置時間過長。工作是排定在指定時間或間隔時間執行的 WebSphere Commerce 指令。如果要指定工作的時間，請在 AddJob 指令中使用指令起始和間隔參數。如需有關 AddJob 指令的詳細資訊，請參閱 WebSphere Commerce 正式作業與開發作業線上說明的「AddJob 指令」。

排程器必須在下列情況下執行：

- IBM WebSphere Commerce Payments 正在執行中。
- 有拍賣在執行中。

- 規則服務正在延伸至正式作業環境
- 網站管理者在變更系統，並且要將變更傳遞到所有複本，但不希望一一重新啟動叢集中的每一個實例。
- 使用約定交付 (ATP) 功能。
- 網站傳送交易訊息。

如果沒有這些狀況存在，網站管理者可以選擇停用排程器。

日誌記載服務程式

WebSphere Commerce 伺服器中記載訊息的目的在於一旦 WebSphere Commerce 應用程式中發生非預期的錯誤或異常狀況時能夠通知管理者。訊息記載及追蹤是重要的診斷工具，因為它們可以協助「網站管理者」來判定問題來源。追蹤是一種判斷問題的機制。追蹤可協助程式開發人員在開發階段中進行程式碼除錯，以及協助技術支援小組解決客戶的問題。由於 WebSphere Commerce 的日誌記載使用的是 WebSphere Application Server 的日誌記載功能，因此可以使用「日誌分析程式」。「日誌分析程式」是一種圖形式公用程式，可以協助檢視和分析日誌檔案。如果需要有關「日誌分析程式」的詳細資訊，請參閱 WebSphere Application Server InfoCenter。

WebSphere Commerce 有提供日誌記載功能。對現有的客戶，本系統會繼續支援 ECTrace 和 ECTMessage。若是新的實作系統，我們建議您使用 WebSphere Application Server 提供的 JRAS，並且可以讓 WebSphere Commerce 應用程式使用。

WebSphere Commerce 日誌的 API 包括：

- ECTrace 追蹤資料流程。基於除錯目的，擷取的追蹤項目會放在檔案中。
- ECTMessage 記載診斷訊息。訊息與語言環境有關，而且儲存在檔案系統中的日誌檔內。如果啟用了錯誤通知，技術支援人員會收到警示通知。診斷日誌是用於問題判定。在預設的情況下，日誌檔名為 `activity.log`。
- JRas 單獨的日誌記載工具集，可以提供訊息記載和診斷追蹤根源。

輸出檔案的預設位置為 `WAS_installdir/logs/WC_<instance name>/`。 ▶ 400
`WAS_userdir/logs/WC_instance_name/`。預設輸出檔案為：

- `native_stderr.log` 是程序日誌，其中包含寫入 `stderr` 串流的文字。
- `native_stdout.log` 是程序日誌，其中包含寫入 `stdout` 串流的文字。
- `startServer.log` 是啟動伺服器時的日誌。
- `stopServer.log` 是停止伺服器時的日誌。
- `SystemErr.log` 記載伺服器執行時的任何系統錯誤。
- `SystemOut.log` 記載伺服器執行時的系統輸出檔案。
- `activity.log` 記載持續的活動。此日誌位於 `WAS_installdir/logs` 目錄中。

如需有關 WebSphere Application Server 日誌 API 的資訊，請參閱 WebSphere Application Server InfoCenter 中的日誌記載與追蹤區段。

ECTMessage

診斷日誌是用於問題判定。ECTMessageLog 記載診斷訊息，ECTMessages 則已經過翻譯。ECTMessages 分割成下列幾個種類：

- 系統訊息

- 使用者訊息

系統與使用者訊息： 系統訊息出現在日誌，有利於問題的除錯。系統訊息提供診斷資訊給「網站管理者」。這些訊息可能會在系統失效後出現，或指出其他重要的事件。

系統訊息會被指定一個產品特有的訊息 ID：

CMNnnnns

其中

nnnn：用來識別什麼會受到影響的鍵值號碼。

s：訊息嚴重性。

使用者訊息經常顯示在瀏覽器之上，有利於造訪網站的客戶。使用者訊息可能會提供有關問題的明細，訊息會指出所指定的參數是否無效，因此向客戶指出在重新提出申請時，應該要修正哪一個值。「網站管理者」可以利用訊息 ID 查閱和該訊息相關的其餘明細；客戶可以向支援人員報告取得的訊息，作為疑難排解之用。

日誌記載層次： WebSphere Commerce 記載系統中有 5 種記載層次或嚴重性。系統訊息的嚴重性包括：錯誤、狀態、警告、參考以及除錯。

- 在預設的情況下，始終都會記載錯誤訊息。錯誤訊息暴露出可能導致系統故障的錯誤狀況。錯誤訊息可以透過電子郵件、MQSeries® 訊息，或其他通知形式，傳送給已利用傳訊登錄的「網站管理者」。
- 狀態訊息指出 WebSphere Commerce 所達到的某些狀態。比方說，當載入時，或當訊息已移至特定狀況時，每一個元件都應該存留一則訊息。狀態訊息可協助技術支援人員來瞭解元件及應用程式的狀況。
- 警告訊息顯示潛在的問題。
- 參考訊息是在 WebSphere Commerce 系統內發生的事件後出現。參考訊息是與觸發系統狀況中之變更的事件有關。比方說，當提交了訂單後，就會存留一則參考訊息。
-

除錯訊息與追蹤元件重疊。不過，除錯訊息可讓「網站管理者」自行調查問題，不需技術支援人員介入。

由於 WebSphere Commerce 是使用 WebSphere Application Server 日誌記載機能，而且 WebSphere Application Server 只有三種日誌記載層次，因此 WebSphere Commerce 的日誌記載層次會依如下所示，對映到 WebSphere Application Server：

表 3.

WebSphere Commerce 的日誌記載層次	WebSphere Application Server 的日誌記載層次
ERROR/ERR	TYPE_ERROR/TYPE_ERR
INFORMATION/INFO	TYPE_INFORMATION/TYPE_INFO
DEBUG	TYPE_INFORMATION/TYPE_INFO
WARNING/WARN	TYPE_WARNING/TYPE_WARN
STATUS	TYPE_INFORMATION/TYPE_INFO

系統使用 WebSphere Application GUI 來控制啟用日誌記載層次或嚴重性類型。「網站管理者」可以在「WebSphere Application Server 管理主控台」中，指定要記錄的嚴重性。如需有關啟用日誌記載層次的詳細資訊，請參閱 WebSphere Application Server InfoCenter 中的資訊。

ECTrace

追蹤是用於問題判定。追蹤可協助程式開發人員在開發階段中進行程式碼除錯，以及協助技術支援小組解決客戶的問題。

追蹤資料是爲了日後參照追蹤檔而存留。資料結構是由環境定義資訊（如類別名稱、方法名稱及文字訊息）所組成。多重資料結構說明軟體應用程式內的資料流程。藉由分析資料結構順序，程式開發人員可以瞭解執行的路徑，如此可以協助判定故障的來源。

JRas

JRas 包含多個 Java 套件，提供訊息記載以及診斷追蹤根源。這些根源並不限定於任何特定的產品或平台。JRas 是由幾個元件組成：

- 日誌程式：日誌程式是和使用使用者程式碼互動的主要物件。日誌程式有兩種：訊息日誌程式和追蹤日誌程式。訊息日誌程式只會建立訊息記錄，追蹤日誌程式則會建立追蹤記錄。日誌程式中會包含一或多個「處理常式」，日誌程式會將事件轉遞到處理常式，供進一步處理。
- 處理常式：處理常式從日誌程式接收事件，然後透過輸出裝置或事件消費者提供摘要。範例之一爲檔案處理常式，檔案處理常式知道如何將事件寫入檔案。
- 格式製作器：處理常式會使用格式製作器配置，格式製作器知道如何將特定類型的事件格式化。
- 事件類型 - 訊息和追蹤都有他們自己預先定義的事件類型。
- 事件類別 - 獨立的 JRas 日誌記載工具集會定義訊息與追蹤事件類別。

註：如需有關 JRas 日誌記載工具集的詳細資訊，請參閱 WebSphere Application Server Information Center 中的 WebSphere Application Server JRas 文件。

WebSphere JRas 擴充：爲了整合到 WebSphere Application Server 執行時期，或在 J2EE 環境中使用，WebSphere 提供一組擴充類別。WebSphere Commerce 的日誌記載使用 WebSphere Application Server 的日誌記載機能，而這些擴充類別則針對不同的 WebSphere 產品產生的訊息與追蹤，提供更好的關聯性。這個擴充類別集合稱爲 WebSphere JRas 擴充。WebSphere JRas 擴充提供適當的日誌程式實作類別。這些訊息與追蹤日誌程式類別的實例，都是直接從位於 `com.ibm.websphere.ras` 套件中的 WebSphere Manager 類別取得，這也是唯一的來源。其他元件，如 Payments 和 JCA 傳訊組織架構，也都使用 WebSphere JRas 擴充。如需和 WebSphere Commerce 以及 Payments 追蹤元件相關的 WebSphere JRas 擴充清單，請參閱第 59 頁的『啓用追蹤元件』。

註：WebSphere Commerce 針對 ECTrace 和 ECTrace 提供一個外層，來呼叫 WebSphere JRas 擴充 API，不過建議「網站管理者」直接呼叫 JRas API。

WebSphere Commerce Payments 問題判定

當您開始使用 WebSphere Commerce Payments 時，您可能需要診斷當您真正使用應用程式時所遇到的問題。「WebSphere 日誌分析程式」有提供一個介面，可以讓您用來分析錯誤與追蹤日誌，以找出應採取的動作。Payments 的所有錯誤和追蹤日誌都可以透過「日誌分析程式」分析，但下列訊息例外：

- 協力廠商卡匣
- 獨立式 Payments UI

- SampleCheckout
- Cashier

規則服務

WebSphere Commerce 使用規則服務和「Advisor 規則伺服器」交談。規則服務是簡化兩支應用程式之間通訊的介面。規則服務提供一種非常簡便的方式，可以在不停止整個 WebSphere Commerce 的情況下，更新網站中以規則為基礎的部份。規則服務存在活動的一對一關係中。以 WebSphere Commerce Accelerator 公佈的每個活動，都必須具有對應的規則服務。這個設計有其彈性優點，因為每一個行銷服務以及活動都可以和其他服務分別啟動、停止以及重新整理。每當客戶設定檔、活動或規則服務透過 WebSphere Commerce Accelerator 更新時，就需要將檔案轉送到正式作業伺服器上的適當位置，然後重新整理規則服務。每一個應用程式複本或 Java 虛擬機器，都在其本端具有規則伺服器的實例。雖然跨越應用程式複本界線進行通訊的次數極少，但是有一種狀況需要做這種通訊：更新規則服務的狀態。為了從每一個應用程式複本收到狀態報告，系統會使用「排程器」廣播要求，以更新規則服務的狀態。其結果會被收集起來，並在使用者按「檢視狀態」按鈕時顯示出來。

Advisor 規則伺服器

WebSphere Commerce 使用「Advisor 規則伺服器」來處理規則，並且提供包含廣告和推薦式銷售技術的個人化行銷內容。此伺服器已納入 WebSphere Commerce Server 中。WebSphere Commerce Server 會呼叫規則伺服器，依據目前的購物環境傳遞資訊。規則伺服器會依據商家或行銷人員建立的規則集合，處理這項資訊，然後針對特定的情況編譯適當的輸出。輸出是根據以 WebSphere Commerce Accelerator 定義的準則。規則伺服器是由「授權資訊」小冊子中指出的特定限制控管。您需要向 HNC Inc 取得個別的授權，才能超出這些限制。

LDAP -- 輕裝備目錄存取通訊協定

LDAP 是一種存取目錄服務的主從架構通訊協定。最初，它是作為 X.500 的前端來使用，也可以搭配單機或其他種目錄伺服器。您可以將 LDAP 當作集中式的資訊儲存庫，來支援各種應用程式之間的資訊分享。

LDAP 資訊模型是以項目為基礎，這個項目含有某個物件（比方說，人員）的相關資訊。項目是由具有一個類型及一個或多個值的屬性所組成。每一個屬性都具有一個語法，來判定屬性中容許哪些種類的值，以及那些值在目錄作業期間如何運作。屬性語法的範例包括 IA5 (ASCII) 字串、JPEG 照片、u-law 編碼的聲音、URL，以及 PGP 金鑰。一般情況下，項目是專門由它的識別名稱或 DN 來加以識別。

在 WebSphere Commerce 中，已登錄的使用者具有一個唯一識別碼，它是一個與識別名稱格式相同的字串。如果 WebSphere Commerce 資料庫作為成員儲存庫，則 DN 的格式將是 'uid=logonIDvalue'，其後是使用者所屬之母項組織實體的 DN。因此，如果使用者具有 logonID 'john' 且屬於 IBM 組織內的 SWG 組織單位，則他的唯一識別碼將是 'uid=john, ou=SWG, o=IBM, o=Root Organization'。如果將目錄伺服器作為成員儲存庫，則已登錄的使用者的唯一識別碼將是他在目錄伺服器中的 DN，在這種情況中，'o=Root Organization' 部份可能不是他的 DN 部份，因為 WebSphere Commerce 支援目錄伺服器中可能不在「根組織」下的現有使用者項目。

WebSphere Commerce 也支援將組織實體儲存在目錄伺服器中，而且每一個組織實體都有一個與 DN 同格式的唯一識別碼。如果 WebSphere Commerce 資料庫作為成員儲存

庫，則唯一識別碼的格式將是 'ou=organizationalEntityName'，其後是母項組織實體的 DN。因此，如果 IBM 組織內的 SWG 組織單位有一個稱為 Marketing 的組織單位，唯一識別碼將是 'ou=Marketing,ou=SWG,o=IBM,o=Root Organization'。如果目錄伺服器作為成員儲存庫，則組織實體的唯一識別碼將是它在目錄伺服器中的 DN，在這種情況中，'o=Root Organization' 部份可能不是它的 DN 部份，因為 WebSphere Commerce 支援目錄伺服器中可能不在「根組織」下的現有組織項目。另請注意，DN 內使用 'o' 或 'ou' 視組織實體是組織或組織單位而定。

當目錄伺服器作為成員儲存庫時，使用者及組織實體可以儲存在目錄伺服器中，而且您可以配置哪一個 LDAP 屬性應該作為 RDN 屬性，以及哪一個 WebSphere Commerce 屬性應該提供 RDN 屬性的值。典型 DN 如下：

```
uid=jsmith, l=Toronto, st=Ontario, c=CA
```

其中

uid 這是使用者的唯一 ID。上面範例中的 uid 屬性即是 RDN 屬性。專門用來識別母項項目下具有 DN *l=Toronto, st=Ontario, c=CA* 的 LDAP 項目。在 WebSphere Commerce 中，如果鑑別檢核是 logonID 與 password，則 WebSphere Commerce 會在 LDAP 伺服器上，RDN = logonID 節點的特定搜尋空間內進行搜尋。如果 X.509 憑證是檢核類型，WebSphere Commerce 將使用憑證中的主旨名稱作為 RDN 屬性值。

l 使用者的地區或城市。

st 使用者的州/省（縣/市）。

c 使用者的國家或地區。

下列 IBM(R) Redbooks 含有其他關於 LDAP 的資訊：

- *LDAP Implementation Cookbook*
- *Understanding LDAP*

Redbooks 可從下列網站取得：<http://www.redbooks.ibm.com>。

如果要在 WebSphere Commerce Server 與 LDAP 伺服器之間使用 SSL，請參閱 LDAP 伺服器提供的指示，來執行適當的設定（比方說，建立金鑰環檔案），以及在 *instancename.xml* 檔中使用 *JNDIEnvPropNameX* 和 *JNDIEnvPropValueX* 屬性，來指定 WebSphere Commerce Server 所需的任何環境內容。

使用 LDAP 做鑑別

使用者設定檔資訊可以儲存在 WebSphere Commerce 資料庫或目錄伺服器中。鑑別選項如下：

- **LDAP**

使用者可以利用他的 DN 或 RDN 值以及密碼來登入。如果使用 RDN，將使用 *ldapentry.xml* 檔中配置的搜尋庫來搜尋使用者。使用者是對照 LDAP 伺服器來進行鑑別。LDAP 伺服器上的使用者資訊會抄寫至 WebSphere Commerce 資料庫，以便能在執行時期使用。

- **資料庫**

使用者是透過使用者提供的登入 ID 及密碼，對照 WebSphere Commerce 資料庫來進行鑑別。

- 其他

使用協力廠商介面的鑑別，會將設定檔資料儲存在 WebSphere Commerce 資料庫中。

如果要指定鑑別模式，請在「配置管理程式」中使用「成員子系統」頁面。如果 Web 伺服器使用 X.509 憑證鑑別使用者，則 X.509 憑證可以與 LDAP 或資料庫鑑別搭配使用。在這種情況下，「鑑別」模式設定會判定設定檔資料的儲存位置：LDAP 或 WebSphere Commerce 資料庫。

LDAP 登錄

在 WebSphere Commerce 中，您可以使用 LDAP 或 WebSphere Commerce 資料庫作為成員儲存庫。使用者及組織實體可以儲存在 LDAP 伺服器中。成員群組目前僅能儲存在 WebSphere Commerce 資料庫中。如果您使用 LDAP 作為登錄，這會對應至 *instancename.xml* 檔中的下列設定： `AuthenticationMode=LDAP` 和 `ProfileDataStorage=LDAP`。

資料會在 LDAP 伺服器與 WebSphere Commerce 資料庫之間抄寫。您可以使用 *ldapentry.xml* 檔，來配置哪些資料要抄寫。

下表說明 WebSphere Commerce 中所提供的 LDAP 服務層次：

表 4. 使用者登錄

	作為登錄的 LDAP	沒有 LDAP 支援
透過 WebSphere Commerce 建立的新使用者登錄，或更新的已登錄使用者資訊	使用者資訊是在 WebSphere Commerce 資料庫及 LDAP 伺服器中建立或更新。	使用者資訊是在 WebSphere Commerce 資料庫中建立或更新。
來自另一個 LDAP 應用程式的 LDAP 使用者登錄或更新資訊	僅在使用者登入 WebSphere Commerce Server 時，或僅在 WebSphere Commerce Server 內的邏輯需要使用者資訊時，使用者資訊才會抄寫至 WebSphere Commerce 資料庫。	N/A
LDAP 伺服器的可用性	LDAP 伺服器必須永遠都可以使用。	N/A
LDAP 連線失敗	有一個錯誤頁面會顯示給使用者。 有一則錯誤訊息會記載在系統日誌檔中。	N/A

表 5. 組織實體登錄

	作為登錄的 LDAP	沒有 LDAP 支援
透過 WebSphere Commerce 建立的新組織實體登錄，或更新的組織實體資訊	組織實體資訊是在 WebSphere Commerce 資料庫及 LDAP 伺服器中建立或更新。	組織實體資訊是在 WebSphere Commerce 資料庫中建立或更新。
組織實體資訊是從另一個 LDAP 應用程式建立或更新	僅在 WebSphere Commerce Server 內的邏輯需要組織實體資訊時，組織實體才會在 WebSphere Commerce 資料庫中建立或更新。	不適用。

其登錄是核准擱置中的使用者也會儲存在目錄伺服器中。在預設的情況下，與環境定義相依的屬性不會儲存在目錄伺服器中。

僅有使用者登錄地址才會存留到 LDAP。LDAP 使用者的通訊錄中的朋友、親戚或商業合夥人地址不會存留下來。

附註：如果您選擇了 LDAP 伺服器作為成員儲存庫，您不應該使用「載入器」套件來大量載入使用者和組織實體資訊至 WebSphere Commerce 資料庫。WebSphere Commerce 5.5 中已經引進商業模型，此商業模型會將大量資料載入資料庫。如需有關商業模型的詳細資訊，請參閱 *WebSphere Commerce 商店程式開發手冊*。

LDAP 配置參數

在安裝了您想要使用的 LDAP 系統後，如果要以手動方式配置 LDAP 支援，您需要完成「配置管理程式」的「成員子系統」頁面中的適當欄位，或完成 *instancename.xml* 檔。這個檔案說明 *instancename.xml* 檔的 MemberSubSystem 區段中的參數。請注意，WebSphere Commerce 的每一個實例都含有 *instancename.xml* 檔。如果要避免可能的衝突，請確定每一個實例都有相同的 LDAP 配置資訊。

附註： 以下是範例程式碼；值是虛構的，僅針對本範例使用。若是您自己的檔案，您必須提供有效的參數值。

```
<MemberSubSystem name="Member SubSystem"
    ProfileDataStorage="LDAP"
    AuthenticationMode="LDAP">
<Directory LdapPort="636"
    LdapType="SECUREWAY"
    LdapAuthenticationMode="SIMPLE"
    EntryFileName="d:/WebSphere/CommerceServer55/
xml/ldap/ldapentry.xml"
    LdapAdminPW="EaDPFd9Vaf0="
    LdapVersion="3"
    LdapHost="basswood.torolab.ibm.com"
    SingleSignOn="0"
    LdapAdminDN="cn=root"
    display="false"
    MigrateUsersFromWCdb="ON"
    JNDIEnvPropName1="java.naming.referral"
    JNDIEnvPropValue1="ignore"
    JNDIEnvPropName2="java.naming.security.
protocol"
    JNDIEnvPropValue2="ssl"
    ...
    JNDIEnvPropNameN="java.naming.language"
    JNDIEnvPropValueN="en-US"
    LdapTimeOut="0" />
</MemberSubSystem>
```

參數說明

ProfileDataStorage

這個屬性值判定使用者和組織實體的設定檔資料的儲存位置。請注意，即使值是 LDAP，部份設定檔資料仍會抄寫至 WebSphere Commerce 資料庫。

AuthenticationMode

這個屬性值判定進行使用者鑑別的系統。有效值為：

- LDAP（預設值）
- DB
- 其他（針對協力廠商機制使用）

LdapAdminDN

用來在 LDAP 上執行作業的管理者 DN。這個 DN 必須在 LDAP 上設定，而且給與 LDAP 上的許可權。

LdapAuthenticationMode

這個屬性值視您正在使用的目錄伺服器而定。

Simple 及 None 對於 IBM SecureWay® 和 Netscape iPlanet 都有效。

- Simple (預設值)
- 無

LdapTimeOut

過幾秒後 LDAP 搜尋即算逾時。逾時值也可以在 LDAP 伺服器上指定。兩值中最小者將優先採用。如果 LdapTimeOut 的值是 0，則搜尋逾時僅由 LDAP 伺服器配置來判定。預設值是 0。

LdapVersion

LDAP 通訊協定版本號碼。唯一有效值是 3。

EntryFileName

含有哪一個 WebSphere Commerce 屬性對映至哪一個 LDAP 屬性之相關資訊的 XML 檔。

預設值是：

 Solaris Linux Windows

`WC_installdir/
xml/ldap/ldapentry.xml`

 AIX 400

`WC_installdir/xml/ldap/ldapentry.xml`

LdapPort

LDAP 伺服器所使用的埠。預設值是 389。

SingleSignOn

'0' 值表示單一登入已關閉。'1' 表示已開啓。

LdapAdminPW

沒有預設值 - 空白

「管理者」的已加密密碼。如果 XML 是以手動修改，則它會以 wcs_encrypt.bat 或 wcs_encrypt.sh 產生。

LdapHost

LDAP 伺服器的主機名稱。預設值是現行主機名稱。

MigrateUsersFromWCdb

這個選項應該要開啓，以確定 bootstrap 使用者以及由商店公佈程序建立的任何組織都會推送到 LDAP。不然，這個選項應該關閉 (OFF)。預設值是 OFF。

LdapType

您正在使用的 LDAP 目錄伺服器，如下所示：

 Windows AIX Solaris Linux IBM SecureWay V5.1 (預設值)。

 400

- OS400 - DIRECTORY SERVICES

- IBM SecureWay V3.2.2

JNDIEnvPropNameX 和 JNDIEnvPropValueX

使用這些屬性來指定任何您想要 WebSphere Commerce 伺服器設定的 JNDI 環境內容。您必須在 `instancename.xml` 檔中，以手動方式指定這些屬性。

請注意下列自 WebSphere Commerce 5.1 以來的變更：

- MappingFileName 已變更為 EntryFileName
- ldapmap.xml 已變更為 ldapentry.xml
- - LdapPersonRDN、LdapPersonSearchRoot、LdapPersonDefaultBase、LdapPersonOCS、LdapOrgOCS 以及 LdapOrgUnitOCS，現在已成為 ldapentry.xml 檔案的一部份，並且其名稱已經變更如下：
 - LdapPersonRDN 現在改為 rdnName
 - LdapPersonSearchRoot 現在改為 searchBase
 - LdapPersonDefaultBase 現在改為 defaultBase
 - LdapPersonOCS 現在改為 objClass
 - LdapOrgOCS 現在改為 objClass
 - LdapOrgUnitOCS 現在改為 objClass
- 僅支援 LDAP 通訊協定版本 3

如果需要 ldapentry.xml 和 ldapmap.dtd 檔的相關資訊，請參閱第 157 頁的附錄 B, 『LDAP 檔』。

LDAP 設計特性

下列資訊強調 WebSphere Commerce 中 LDAP 實作的部份設計特性。

- 多個 WebSphere Commerce 屬性可利用指定的分隔字元抄寫成單一 LDAP 屬性。
- 您可以從「多個 LDAP」屬性抄寫成單一 WebSphere Commerce 屬性。如果要這樣做，請在 ldapentry.xml 檔中使用多個 <map> 區段；不過，您應該明白可能會發生改寫的情況。
- 單一 WebSphere Commerce 屬性可以抄寫成「多個 LDAP」屬性。
- 單一 LDAP 屬性可以抄寫成多個 WebSphere Commerce 屬性。
- 僅有使用者的預設地址會存留至 LDAP。預設登錄地址會以 `selfaddress=1` 及 `logonid=nickname` 定義成使用者通訊錄中的地址。其中 `logonID` 是 `USEREG` 表格中的 `logonid`，而 `nickname` 是 `ADDRESS` 表格中的 `NICKNAME`。使用者通訊錄中的其他地址將不會儲存在 LDAP。
- 僅有來自 WebSphere Commerce 的已登錄的使用者將存留至 LDAP，而不會存留訪客使用者。
- WebSphere Commerce 容許在 LDAP 上搜尋使用者項目時，指定多個搜尋庫。不過，如果找到了多個滿足搜尋準則的使用者項目，將會產生錯誤。

下表說明 WebSphere Commerce 中 LDAP 支援的部份限制。

限制	說明
LDAP 中的多值屬性	WebSphere Commerce 不會儲存或擷取多值 LDAP 屬性，不過，在擷取第一個值時，可以容忍 LDAP 中的多值屬性。更新期間，如果新值已存在於 LDAP，將不會執行任何更新；不然，將更新多值 LDAP 屬性的第一個值。
LDAP 轉介及參照	WebSphere Commerce 不支援 LDAP 轉介及參照。
多元件 RDN	僅一個 LDAP 屬性可以指定為 RDN 屬性，而 LDAP 屬性組合則不能。

暫置作業伺服器

大部份的線上商店都是每年 365 天，每天 24 小時，不間斷地營業，使得要對系統進行維護或測試變更時，極為困難。WebSphere Commerce 暫置作業伺服器可讓網站管理者將正式作業資料庫複製到暫置作業資料庫中，以便測試所作的更新，而不會影響到客戶。這項功能對於產品型錄的更新測試非常有用，不過，對於測試新的購物程序指令也非常有幫助。

暫置作業伺服器是由下列元件組成：

WebSphere Commerce 實例

測試及修改資料。

資料庫綱目 Script

建立暫置作業資料庫的暫置表格和觸發指令。暫置作業資料庫含有與正式作業資料庫相同的綱目和表格，加上一組觸發指令，用來記載暫置作業資料庫中所做的變更。暫置作業資料庫綱目 script 會新增觸發指令至資料庫。

變更係使用資料庫觸發指令記載至 STAGLOG 表格（暫置表格）。每當您變更暫置作業資料庫中的資料庫表格記錄時，STAGLOG 表格即會記錄該變更。

暫置複製公用程式

可讓管理者將正式作業資料庫中的資料複製到暫置作業資料庫中。您可以將資料複製到網站相關表格、商家相關表格或個別表格中。

暫置延伸公用程式

可讓管理者將暫置作業資料庫中的變更延伸到正式作業資料庫中。位於 STAGLOG 表格中的資訊會指出，在暫置作業資料庫中的哪些記錄必須在正式作業資料庫中插入、更新或刪除。所指出的記錄即會在正式作業資料庫中更新。經過處理的記錄，會在 STAGLOG 表格的 STGPROCESSED 直欄內以 1 表示。

暫置檢查公用程式

容許管理者在暫置作業伺服器和正式作業伺服器上，檢查兩個表格之間是否有潛在的唯一索引鍵衝突。

暫置作業伺服器限制

在使用暫置作業伺服器之前，您應該注意下列限制：

- 您無法使用暫置作業伺服器與買方組織的自我管理特性搭配。

- 所有暫置表格（不包括 MEMBER、MBRREL、MBRROLE 和 MBRATTRVAL）的 member_id 直欄都必須是組織或成員群組，不得為使用者。
- 對於所有網站表格，member_id 必須是 -2001 或 0。對於同時含有網站和商家資料的所有表格，與網站資料相關之列的 member_id 必須是 0 或 -2001。
- 如果您是在正式運作系統上使用 RFQ 特性，將無法使用「暫置複製」指令。您必須使用推進模式，「暫置複製」因為這種模式而無法使用。在啟動您的正式作業網站之前，請建立暫置作業伺服器，並設定暫置作業資料庫。在暫置作業伺服器上部署和測試您的資料，然後使用「暫置延伸」指令推進至正式作業伺服器。
- 您無法在暫置作業伺服器上建立或更新 RFQ。
- 您無法在暫置作業資料庫上啟用快取觸發指令。

主要鍵分割: 在轉售商市場中，IBM 和「網站管理者」會更新暫置作業伺服器上的資料，而轉售商則更新正式作業伺服器上的資料。這可能會導致主要鍵衝突。

WebSphere Commerce 使用金鑰管理程式來產生表格的主要鍵，且鍵值範圍定義在 KEYS 表格中。如果正式作業伺服器和暫置作業伺服器使用相同的鍵值範圍，則金鑰管理程式可以架構相同的主要鍵值給正式作業和暫置作業伺服器上的相同表格，因而導致主要鍵衝突。結果，在建立暫置作業實例和正式作業實例後，必須立即分割鍵值範圍。這可以確保暫置作業伺服器和正式作業伺服器將使用不同的鍵值範圍。

目前，WebSphere Commerce 使用下列 SQL 來分割暫置作業和正式作業伺服器上的鍵值範圍。SQL 陳述式會將現行鍵值範圍分割成三個相等的部份。

第一部份

```
update keys set upperbound=(upperbound-lowerbound)/3 + lowerbound
where tablename in (select tablename from stgmertab) or tablename
in (select tablename from stgsitetab)
```

第二部份

```
db2 update keys set upperbound = (upperbound-lowerbound)/3*2 + lowerbound,
lowerbound = (upperbound-lowerbound)/3 +lowerbound+1, counter =
counter+(upperbound-lowerbound)/3 +1 where tablename in
(select tablename from stgmertab)
or tablename in (select tablename from stgsitetab)
```

第三部份

```
db2 update keys set lowerbound = (upperbound-lowerbound)/3*2 + lowerbound +1
, counter = counter +(upperbound-lowerbound)/3*2 +1 where tablename in
(select tablename from stgmertab) or tablename in (select tablename from stgsitetab)
```

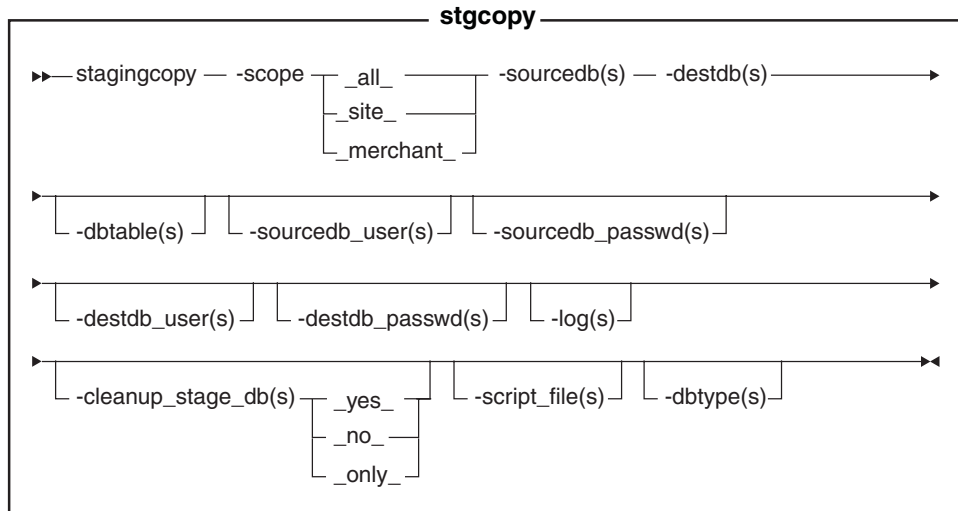
第一個 SQL 陳述式（第一部份）必須在正式作業伺服器上執行。這可以確保正式作業伺服器將佔據完整鍵值範圍的三分之一。第二個 SQL 陳述式（第二部份）必須在暫置作業伺服器上執行。這表示暫置作業伺服器將佔據完整鍵值範圍的第二個三分之一。第三個 SQL 陳述式（第三部份）可以保存下來，以便在未來供第二個可能的暫置作業伺服器使用。

註: 上面的鍵值分割 SQL 已與實例配置程序整合在一起。您不需要以手動方式執行它們。

暫置作業伺服器指令

暫置複製公用程式 (Windows NT®、Windows 2000、AIX、Linux、iSeries 以及 Solaris): 暫置複製公用程式會將正式作業資料庫中的資料複製到暫置作業資料庫中。請注意，如果 RFQ 在您的正式運作系統上，您將無法使用這個指令。如需詳細資訊，請參閱第 32 頁的『暫置作業伺服器限制』。

如果要執行「暫置複製」公用程式，請從可以同時連接至暫置作業伺服器和正式作業伺服器資料庫之機器上的指令行，鍵入下列指令。請將整個指令鍵入在一行上。在這裡以多行顯示，僅是爲了便於呈現而已。



註:

- ▶ **Oracle** 即使您目前正以相同的使用者 ID 執行這個公用程式，您也必須在指令中併入選用參數、登入使用者 ID 和密碼。
- ▶ **400** iSeries 現在改爲使用 stagingCopy.sh。如要在 iSeries 上執行 shell script，請執行下列步驟：
 - 以 CCSID 爲 65535 以外的使用者設定檔登入。
 - 在 OS/400 指令行上鍵入指令：STRQSH，開啓一個 QSHELL 指令視窗。
 - 依照下列方式，執行指令：
/QIBM/ProdData/CommerceServer55/bin/stagingCopy.sh (參數 . . .)

註: 必須要指定使用者參數。

參數值

scope 複製至暫置作業伺服器的範圍層次。請指定下列其中一個：

- **_all_**
鍵入 **_all_**，複製與網站和所有商家相關的記錄。
- **_site_**
鍵入 **_site_**，僅複製與網站相關的記錄。
- **_merchant_**
鍵入 **_merchant_**，僅複製與所有商家相關的記錄。

sourcedb

正式作業伺服器上的資料庫名稱。

▶ 400 正式作業伺服器上的資料庫名稱，如關聯式資料庫目錄中所顯示者。

▶ Oracle 使用 `host:port:sid`。比方說，`myhost:1521:mydb`。

destdb

暫置作業伺服器上的資料庫名稱。

▶ 400 正式作業伺服器上的資料庫名稱，如關聯式資料庫目錄中所顯示者。

▶ Oracle 使用 `host:port:sid`。比方說，`myhost:1521:mydb`。

dbtable

(選用) 要複製之任何特定表格的名稱。將複製這個表格中的所有記錄，但記錄須在 `scope` 參數指定的範圍內；不然，將不複製任何記錄。

sourcedb_user

(選用) 已建立來源資料庫綱目之資料庫管理者的登入 ID。

▶ 400 (必要的) 與 Commerce 實例相關的使用者設定檔。這和來源資料庫綱目相同。

註: 如果未指定的話，將使用目前呼叫公用程式之使用者的 ID。

sourcedb_passwd

(選用) `sourcedb_user` 參數指定之登入 ID 的密碼。

destdb_user

(選用) 已建立目標資料庫綱目之資料庫管理者的登入 ID。▶ 400 (必要的) 與 Commerce 實例相關的使用者設定檔。這和目的地資料庫綱目相同。

註: 如果未指定的話，將使用呼叫公用程式之使用者的 ID。

destdb_passwd

(選用) `destdb_user` 參數指定之登入 ID 的密碼。如果未指定的話，系統將提示您輸入密碼。

log

(選用) 「暫置複製」公用程式記錄其活動和錯誤所在之檔案的路徑和名稱。如果未指定這個參數，將在下列 `log` 目錄中建立稱為 `stagingcopy_yyyy.mm.dd_hh.mm.ss.zzz.log` 的日誌檔。

▶ AIX ▶ 400 `WAS_userdir/logs`

▶ Solaris ▶ Linux `WAS_installdir/logs`

▶ Windows `WC_installdir\logs`

▶ 400 如果未指定這個參數，將會在 `WC_userdir/instances` 目錄中，建立一個名稱為

`stagingcopy_sourcedb_user_destdb_user_yyyy.mm.dd_hh.mm.ss.zzz.log` 的日誌檔。

cleanup_stage_db

(選用) 在使用暫置複製公用程式之前，請利用這個參數來清理暫置表格。當您使用 `-cleanup_stage_db` 參數清理網站資料時，請注意，因為刪除連鎖反應，所以可以刪除商家資料。在清理及複製網站資料後，您應該清理及複製商家資料。Yes 是預設值。如果您指定 `no`，將不會從暫置表格刪除任何資料。如果您的複製資料在主要鍵或唯一索引上產生衝突或重複的鍵值，則您的複製可能會

失敗。如果只要使用暫置複製來清理您的暫置作業資料庫，不從正式作業資料庫複製資料，請將 `-cleanup_stage_db` 指定為 `only`。

script file

(選用) 當根據指定的範圍，使用匯出及匯入，將正式作業資料庫複製至暫置作業資料庫時，「暫置複製」公用程式指令所產生之 SQL script 檔的名稱。如果您使用預設值或將 `-cleanup_stage_db` 指定為 `yes`，script 檔也會產生刪除陳述式，來清理暫置作業資料庫。

註: iSeries 上不支援這個 Script 檔。

在執行 script 之前，請驗證您有足夠的磁碟空間來保存匯出的表格。script 檔位於您呼叫「暫置複製」公用程式所在的「暫置複製」公用程式目錄。

 使用 `db2 -vtd# -f script_file_name` 來執行 script 檔。

dbtype (選用) 資料庫類型 (DB2 或 Oracle)。預設值是 DB2。

如果需要複製暫置作業伺服器的詳細資訊，請參閱範例。

將資料複製至暫置作業資料庫的範例: 下列範例說明如何將正式作業資料庫中的表格複製至暫置作業資料庫。您必須記住，如果 RFQ 在您的正式運作系統上，您將無法使用「暫置複製」公用程式。詳細說明請參閱第 32 頁的『暫置作業伺服器限制』。

請注意，您應該將整個指令鍵入在單一行中。在這裡以多行顯示指令，僅是為了便於呈現而已。

範例 1

在清理暫置作業資料庫後，將 `scope` 設為 `all`，把正式作業資料庫複製至暫置作業資料庫：



1. 設定 PATH 環境變數。  不適用。
2. 配置資料庫。  不適用。
3. 切換至要寫入日誌檔的目錄。  不適用。如果是 iSeries，日誌檔案會預設為 `/QIBM/UserData/CommerceServer55/instances/stagingcopy_{sourcedb_user}_{destdb_user}_{timestamp}.log`
4. 輸入下列指令：
 -  
`stagingcopy -scope _all_ -sourcedb production_database_name -destdb staging_database_name`
 -  
`stagingcopy -scope _all_ -sourcedb production_database_name -destdb staging_database_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`
 -     
`. stagingcopy.sh -scope _all_ -sourcedb production_database_name -destdb staging_database_name`
 -   
`. stagingcopy.sh -scope _all_ -sourcedb production_database_name -destdb`



```
staging_database_name dbtype oracle -sourcedb_user user -sourcedb_passwd
password -destdb_user user -destdb_passwd password
```






附註：使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。




5. 檢查 stagingcopy_yyyy.mm.dd_hh.mm.ss.zzz.log 檔，以驗證指令是否成功。

如果要指定日誌檔名稱和路徑，請使用日誌檔參數：

-   stagingcopy -scope _all_ -sourcedb production_database_name -destdb staging_database_name -log log_file_name

-   stagingcopy -scope _all_ -sourcedb production_database_name -destdb staging_database_name -log log_file_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password

-      . stagingcopy.sh -scope _all_ -sourcedb production_database_name -destdb staging_database_name -log log_file_name

-    . stagingcopy.sh -scope _all_ -sourcedb production_database_name -destdb staging_database_name -log log_file_name dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password

附註：使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。



如果您正在使用 DB2，但未以「資料庫管理者」登入，



則需要提供 -sourcedb_user、-sourcedb_passwd、-destdb_user 和 -destdb_passwd 選項的值。

範例 2




從暫置作業資料庫清理商家表格後，將正式作業資料庫中與商家相關的表格複製至暫置作業資料庫：

1. 設定 PATH 環境變數。
2. 配置資料庫。
3. 切換至要寫入日誌檔的目錄。
4. 輸入下列指令：

-   stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name

-   stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password

-      stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name

-    stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -dbtype oracle

```
-sourcedb_user user -sourcedb_passwd password -destdb_user user  
-destdb_passwd password
```

附註：使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。

5. 檢查 stagingcopy_yyyy.mm.dd_hh.mm.ss.zzz.log 檔，以驗證指令是否成功。

如果僅要清理暫置作業資料庫，請指定 -cleanup_stage_db 參數：

- Windows DB2 stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only
- Windows Oracle stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password
- AIX 400 Solaris Linux DB2 stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only
- AIX Solaris Oracle stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password

附註：使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。

如果僅要複製資料，請指定 -cleanup_stage_db no 參數：

- Windows DB2 stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no
- Windows Oracle stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password
- AIX 400 Solaris Linux DB2 stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no
- AIX Solaris Oracle stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password

附註：使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。













重要事項：當以 scope 設為 merchant 來進行複製時，請先確定您已複製了網站範圍資料。不然，您的複製將失敗。

範例 3

從暫置作業資料庫清理網站表格後，將正式作業資料庫中的網站表格複製至暫置作業資料庫。

1. 設定 PATH 環境變數。













2. 配置資料庫。
3. 切換至要寫入日誌檔的目錄。
4. 輸入下列指令：

-   `stagingcopy -scope _site_ -sourcedb production_database_name -destdb staging_database_name`
-   `stagingcopy -scope _site_ -sourcedb production_database_name -destdb staging_database_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`
-      `stagingcopy.sh -scope _site_ -sourcedb production_database_name -destdb staging_database_name`
-    `stagingcopy.sh -scope _site_ -sourcedb production_database_name -destdb staging_database_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`

附註：使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。

5. 檢查 stagingcopy_yyyy.mm.dd_hh.mm.ss.zzz.log 檔，以驗證指令是否成功。

刪除網站表格可能會影響商家表格，這是因為刪除連鎖反應限制的關係。請先清理商家資料，再清理網站資料，然後複製資料：

-   `stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only`
-   `stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`
-      `stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only`
-    `stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`

附註：使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。

-   `stagingcopy -scope _site_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only`

- Windows Oracle stagingcopy -scope _site_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password

- AIX 400 Solaris Linux DB2 stagingcopy.sh -scope _site_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only

- AIX Solaris Oracle stagingcopy.sh -scope _site_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db only dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password

附註：使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。

- Windows DB2 stagingcopy -scope _site_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no

-

Windows

Oracle

```
stagingcopy -scope _site_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password
```

- AIX 400 Solaris Linux DB2 stagingcopy.sh -scope _site_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no

- AIX Solaris Oracle stagingcopy.sh -scope _site_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password

附註：使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。

- Windows DB2 stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no

- Windows Oracle stagingcopy -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password

- AIX 400 Solaris Linux DB2 stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no

- AIX Solaris Oracle stagingcopy.sh -scope _merchant_ -sourcedb production_database_name -destdb staging_database_name -cleanup_stage_db no dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user

-destdb_passwd *password*

附註：使用 *host:port:sid* 作為 Oracle 資料庫名稱。比方說，*myhost:1521:mydb*。

範例 4

以 `scope all` 產生下列 `script` 來清理並複製正式作業資料庫至暫置作業資料庫。

400 這個範例不適用於 OS/400 for iSeries，因為 iSeries 上不支援 `-script` 選項。

1. 設定 `PATH` 環境變數。
2. 配置資料庫。
3. 切換至要寫入日誌檔的目錄。
4. 輸入下列指令：

- **Windows** **DB2** `stagingcopy -scope _all_ -sourcedb production_database_name -destdb staging_database_name -script_file stage_copy.sql`

- **Windows** **Oracle** `stagingcopy -scope _all_ -sourcedb production_database_name -destdb staging_database_name -script_file stage_copy.sql -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`

- **AIX** **Solaris** **Linux** **DB2** `stagingcopy.sh -scope _all_ -sourcedb production_database_name -destdb staging_database_name -script_file stage_copy.sql`

- **AIX** **Solaris** **Oracle** `stagingcopy.sh -scope _all_ -sourcedb production_database_name -destdb staging_database_name -script_file stage_copy.sql -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`

附註：使用 *host:port:sid* 作為 Oracle 資料庫名稱。比方說，*myhost:1521:mydb*。

5. 檢查 `stagingcopy_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

「暫置複製」公用程式會產生 `stage_copy.sql` `script` 來清理並複製資料庫。 **DB2**
如果您是使用 `DB2`，請執行下列 `script`：

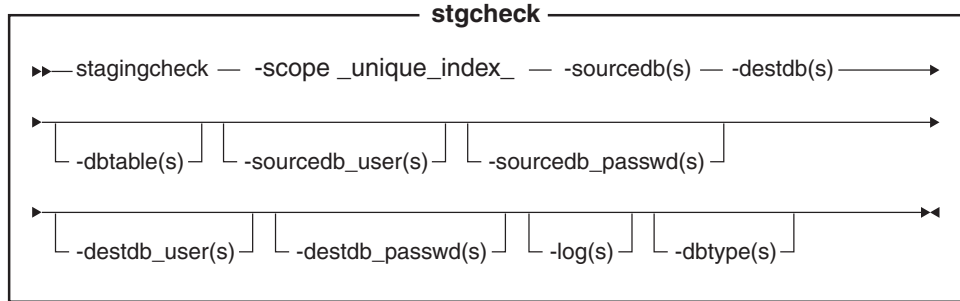
1. 以「資料庫管理者」(DBA) 登入。
2. 開啓 `DB2` 指令視窗。
3. 鍵入：`db2 -vtd# -f stage_copy.sql`

Oracle 如果您正在使用 `Oracle`，請執行下列 `script`：

1. 開啓 `SQLPlus` 視窗。
2. 以 `dba` 連接。
3. 鍵入：`@stage_copy.sql`

暫置檢查指令 (Windows NT、Windows 2000、AIX、iSeries、Linux 以及 Solaris): 「暫置檢查」指令會判定暫置作業資料庫與正式作業資料庫之間是否有唯一索引鍵衝突。如果要執行「暫置檢查」公用程式，請從暫置作業伺服器或正式作業伺服器上的指令行鍵入下列指令。請將整個指令鍵入在一行上。在這裡以多行顯示，僅

是爲了便於呈現而已。



Oracle 附註：即使您目前正以相同的使用者 ID 執行這個公用程式，您也必須在指令中併入選用參數、登入使用者 ID 和密碼。

400 iSeries 現在改爲使用 `stagingCheck.sh`。如要在 iSeries 上執行 shell script，請執行下列步驟：

1. 以 CCSID 爲 65535 以外的使用者設定檔登入。
2. 在 OS/400 指令行上鍵入指令：STRQSH，開啓一個 QSHELL 指令視窗。
3. 依照下列方式，執行指令：

`/QIBM/ProdData/CommerceServer55/bin/stagingCheck.sh (參數 . . .)`

註：必須要指定使用者參數。

參數值

scope 複製至暫置作業伺服器的範圍層次。指定 `_unique_index_`。

sourcedb

暫置作業伺服器上的資料庫名稱。

Oracle 使用 `host:port:sid`。比方說，`myhost:1521:mydb`。

destdb 正式作業伺服器上的資料庫名稱。

Oracle 使用 `host:port:sid`。比方說，`myhost:1521:mydb`。

dbtable

(選用) 要檢查是否有唯一鍵值衝突之特定表格的名稱。

sourcedb_user

(選用) 已建立暫置作業資料庫綱目之資料庫管理者的登入 ID。如果未指定的話，將使用目前呼叫公用程式之使用者的 ID。

sourcedb_passwd

(選用) `sourcedb` 參數指定之登入 ID 的密碼。

destdb_user

(選用) 已建立正式作業資料庫綱目之資料庫管理者的登入 ID。如果未指定的話，將使用呼叫公用程式之使用者的 ID。

destdb_passwd

(選用) `destdb_user` 參數指定之登入 ID 的密碼。如果未指定的話，系統將提示您輸入密碼。

參數值

scope

延伸至正式作業伺服器的範圍層次。請指定下列其中一個：


- `_all_`
鍵入 `_all_`，延伸與網站和所有商家相關的記錄。
- `_site_`
鍵入 `_site_`，僅延伸與網站相關的記錄。
- `_merchant_`
鍵入 `_merchant_`，僅延伸與所有商家相關的記錄。

sourcedb

暫置作業伺服器上的資料庫名稱。

 Oracle 使用 `host:port:sid`。比方說，`myhost:1521:mydb`。

destdb 正式作業伺服器上的資料庫名稱。

 Oracle 使用 `host:port:sid`。比方說，`myhost:1521:mydb`。

dbtable

(選用) 要延伸之任何特定表格的名稱。將延伸這個表格中的所有變更記錄，但記錄須在 `scope` 參數指定的範圍內；不然，將不延伸任何記錄。

sourcedb_user

(選用) 已建立來源資料庫綱目之資料庫管理者的登入 ID。如果未指定的話，將使用目前呼叫公用程式之使用者的 ID。

sourcedb_passwd

(選用) `sourcedb_user` 參數指定之登入 ID 的密碼。

destdb_user

(選用) 已建立目標資料庫綱目之資料庫管理者的登入 ID。如果未指定的話，將使用呼叫公用程式之使用者的 ID。當使用遠端資料庫時，此參數是必要的。

destdb_passwd

(選用) `destdb_user` 參數指定之登入 ID 的密碼。如果未指定的話，系統將提示您輸入密碼。當使用遠端資料庫時，此參數是必要的。


log

(選用) 「暫置延伸」公用程式記錄其活動和錯誤所在之檔案的路徑和名稱。如果未指定這個參數，將在下列 `log` 目錄中建立稱為 `stagingprop_yyyy.mm.dd_hh.mm.ss.zzz.log` 的日誌檔。

 AIX `WAS_userdir/logs`

 Solaris  Linux `WAS_installdir/logs`

 Windows `WC_installdir\logs`

 400 如果未指定這個參數，將會在 `WC_userdir/instances` 目錄中，建立一個名為 `stagingprop_sourcedb_user_destdb_user_yyyy.mm.dd_hh.mm.ss.zzz.log` 的日誌檔。

dbtype (選用) 資料庫類型 (DB2 或 Oracle)。預設值是 DB2。

如果需要延伸至暫置作業伺服器的詳細資訊，請參閱下列範例。













將資料延伸至正式作業資料庫的範例： 下列範例說明如何將暫置作業資料庫中的已變更記錄延伸至正式作業資料庫。

請注意，您應該將指令鍵入在單一行中。在這裡以多行顯示指令，僅是爲了便於呈現而已。

範例 1

將暫置作業伺服器資料庫中的所有變更延伸到正式作業資料庫中。

1. 設定 PATH 環境變數。
2. 配置資料庫。
3. 切換至要寫入日誌檔的目錄。
4. 輸入下列指令：










-   stagingprop -scope _all_ -sourcedb *staging_database_name* -destdb *production_database_name*
-   stagingprop -scope _all_ -sourcedb *staging_database_name* -destdb *production_database_name* -dbtype oracle -sourcedb_user *user* -sourcedb_passwd *password* -destdb_user *user* -destdb_passwd *password*
-      stagingprop.sh -scope _all_ -sourcedb *staging_database_name* -destdb *production_database_name*
-    stagingprop.sh -scope _all_ -sourcedb *staging_database_name* -destdb *production_database_name* dbtype oracle -sourcedb_user *user* -sourcedb_passwd *password* -destdb_user *user* -destdb_passwd *password*

附註：使用 `host:port:sid` 作爲 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

5. 檢查日誌檔，察看指令是否成功。檢查日誌參數的說明，判斷這個日誌檔案在您的資料庫類型與平台的預設位置。

「暫置延伸」公用程式首先將延伸所有網站資料，然後再延伸所有商家資料至正式作業資料庫。如果發生錯誤，整個交易將回復。

如果要指定日誌檔名稱和路徑，請使用日誌檔參數：

-   stagingprop -scope _all_ -sourcedb *staging_database_name* -destdb *production_database_name* -loglog_file_name
-   stagingprop -scope _all_ -sourcedb *staging_database_name* -destdb *production_database_name* -loglog_file_name -dbtype oracle -sourcedb_user *user* -sourcedb_passwd *password* -destdb_user *user* -destdb_passwd *password*
-      stagingprop.sh -scope _all_ -sourcedb *staging_database_name* -destdb *production_database_name* -loglog_file_name

- ▶ AIX ▶ Solaris ▶ Oracle

```
stagingprop.sh -scope _all_ -sourcedb staging_database_name-destdb
production_database_name -loglog_file_name-dbtype oracle -sourcedb_user user
-sourcedb_passwd password -destdb_user user -destdb_passwd password
```

附註：使用 `host:port:sid` 作為 Oracle 資料庫名稱，比方說，`myhost:1521:mydb`。

如果您正在使用 DB2，但未以「資料庫管理者」登入，則需要提供 `-sourcedb_user`、`-sourcedb_passwd`、`-destdb_user` 和 `-destdb_passwd` 選項的值。

範例 2

將暫置作業資料庫中所有修改過的網站資料延伸到正式作業資料庫中。

1. 設定 PATH 環境變數。
2. 配置資料庫。
3. 切換至要寫入日誌檔的目錄。
4. 輸入下列指令：

- ▶ Windows ▶ DB2 `stagingprop -scope _site_ -sourcedb staging_database_name-destdb production_database_name`
- ▶ Windows ▶ Oracle `stagingprop -scope _site_ -sourcedb staging_database_name-destdb production_database_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`
- ▶ AIX ▶ 400 ▶ Solaris ▶ Linux ▶ DB2 `stagingprop.sh -scope _site_ -sourcedb staging_database_name-destdb production_database_name`
- ▶ AIX ▶ Solaris ▶ Oracle `stagingprop.sh -scope _site_ -sourcedb staging_database_name-destdb production_database_name dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`

附註：使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

5. 檢查日誌檔，察看指令是否成功。檢查日誌參數的說明，判斷這個日誌檔案在您的資料庫類型與平台的預設位置。

範例 3

將暫置作業資料庫中所有修改過的商家資料延伸到正式作業資料庫中（在延伸網站資料後）。

1. 設定 PATH 環境變數。
2. 配置資料庫。
3. 切換至要寫入日誌檔的目錄。
4. 輸入下列指令：

- ▶ Windows ▶ DB2 `stagingprop -scope _merchant_ -sourcedb staging_database_name-destdb production_database_name`

- Windows Oracle `stagingprop -scope _merchant_ -sourcedb staging_database_name-destdb production_database_name -dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`
- AIX 400 Solaris Linux DB2 `stagingprop.sh -scope _merchant_ -sourcedb staging_database_name-destdb production_database_name`
- AIX Solaris Oracle `stagingprop.sh -scope _merchant_ -sourcedb staging_database_name-destdb production_database_name dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`

附註：使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

- 檢查日誌檔，察看指令是否成功。檢查日誌參數的說明，判斷這個日誌檔案在您的資料庫類型與平台的預設位置。

自訂的資料庫表格需求

如果您藉由建立新表格，自訂了您的資料庫綱目，您必須符合下列需求，才能使用暫置作業伺服器：

- 您必須定義主要鍵或唯一索引。**
 暫置作業伺服器是根據鍵值來運作。如果要避免在 `STAGLOG` 表格中記載過多的資料，請僅記載鍵值（主要鍵或唯一索引）。暫置公用程式將使用鍵值進行壓縮，以及尋找要延伸的資料。如果沒有鍵值，暫置公用程式將無法運作。
- 表格之間不能有參照完整性 (RI) 限制循環。**
 在延伸子項表格之前，暫置作業伺服器恆會先延伸母項表格。如果有一個 `RI` 限制循環，暫置作業伺服器將無法區分母項與子項表格。
- 資料庫表格僅含有配置資料。**
 在大眾消費型商務實務內容中，配置資料是在「網站管理者」控制之下，如型錄和型錄項目。如果表格含有作業資料，則在「網站管理者」將表格複製到暫置作業資料庫後，客戶就可以變更正式作業資料庫中的相同表格。這將導致潛在的鍵值衝突或 `RI` 限制違規。
- 資料庫表格不能含有作業表格的任何參照。**
 要延伸的表格不應該含有作業表格之主要鍵的任何外來鍵參照。當有如此的參照時，如果客戶在執行暫置複製後刪除了主要鍵，則資料將無法還原至產品資料庫。
- Windows AIX 400 Solaris Linux **在正式作業資料庫中插入兩個表格時，不能有插入觸發指令存在。**
 對於暫置作業伺服器所涵蓋的任何兩個表格（比方說，`R1` 和 `R2`），當插入到正式作業資料庫中的 `R2` 和 `R1` 時，不能有將列插入 `R1` 或 `R2` 的觸發指令。插入觸發指令會在這兩個資料庫中建立更新，因而產生鍵值問題。
- MEMBER 表格不能有唯一索引。**
- 在自訂的資料庫表格上刪除限制時，必須謹慎小心。**
 刪除限制會抑制「資料庫清理」公用程式的效能。當清理暫置作業資料庫時，您可能遭遇困難。在可以清理暫置作業資料庫之前，您必須以手動方式使用「資料庫清理」公用程式指令與 `force` 選項搭配，來清理表格。不然，清理暫置作業資料庫將失敗。

如果要為自訂的表格準備暫置作業伺服器，請參閱為自訂的表格配置暫置作業伺服器。

暫置作業伺服器疑難排解 (Windows NT、Windows 2000、AIX、iSeries、Linux 以及 Solaris)



1. 當使用暫置作業伺服器指令 `stgprop` 時，您可能會收到下列訊息；
SQLSTATE 54001：陳述式太長或太複雜。

確定您已按照配置資料庫的建議，設定資料庫的 `stmthep` 大小。您應該驗證是否有足夠的記憶體來支援增加大小。

2. 如果「暫置複製」公用程式無法完成處理，可能是資料庫日誌太小。從 DB2 指令視窗發出指令：

```
update database configuration for staging_server using logprimary 50  
logfilsiz 1000  
db2 terminate  
db2stop  
db2start
```

其中 `staging_server` 是暫置作業伺服器資料庫的名稱。如果問題依然存在，請嘗試對 `logprimary` 或 `logfilsiz` 參數使用較高的值。

3.   確定 DB2 使用者（如 `db2inst1`）的資源已配置妥當。輸入下列指令以顯示資源限制：`ulimit -a`
將資料區段配置為 240MB，將堆疊配置為 16MB。

如果需要其他資訊，請參閱 *DB2 Command Reference*。

WebSphere Commerce 暫置表格

您可以將正式作業伺服器中的 WebSphere Commerce 暫置表格複製到暫置作業伺服器。表格會根據它們是否含有網站相關資料、商家相關資料，或網站和商家相關資料來分組。

每一個表格群組都會列示在下列幾頁：

- 網站資料範圍
- 商家資料範圍
- 網站和商家資料範圍

暫置表格的觸發指令

下列是已針對 WebSphere Commerce 暫置表格定義的觸發指令。如果這些設定含有相同的資料範圍和鍵值性質，則您可以將它們套用至自訂的表格。請參閱底下的鏈結，按資料範圍來檢視表格。

- 網站資料範圍
- 商家資料範圍
- 網站或商家資料範圍

網站資料範圍： 下圖列出網站資料範圍下的資料庫表格。

表格名稱	唯一索引				
	整數 1	整數 2	字元 1	整數 3	字元 2
ACACGPDESC	是	是			
ACACTACTGP	是	是			
ACACTDESC	是	是			
ACACTGRP	是				
ACACTION	是				
ACATTR	是				
ACATTRDESC	是	是			
ACCCMDTYPE			是		
ACRELATION	是				
ACRELDDESC	是	是			
ACRESACT	是	是			
ACRESATREL	是	是			
ACRESCGRY	是				
ACRESGRP	是				
ACRESGPDES	是	是			
ACRESGPRES	是	是			
ACRESPRIM	是				
ACRESREL	是	是			
ATTACHUSG			是		
ATTRTYPE			是		
BUYERPOTYP	是				
CALUSAGE	是				
CATENTTYPE			是		
CATRELTYPE			是		
CHKARRANG	是	是			
CHKCMD	是				
COUNTCODE			是		是
COUNTRY	是		是		
DEVICEFMT	是				
FLCOMPOSE	是	是		是	
FLDOMNDESC	是	是			
FLOW	是				
FLOWDESC	是	是			
FLOWDOMAIN	是				
FLOWTYPE	是				
FLSTATEDCT	是				
FLSTATEGP	是				
FLSTATEREL	是	是			
FLSTDCTDSC	是	是			
FLSTGPDSC	是	是			

FLTRANSDSC	是	是			
FLTRANSITN	是				
FLTYPEDESC	是	是			
ICDATAREG	是				
INVRSRVTYP	是				
INVRSRVDSC	是	是			
ITEMTYPE			是		
LANGUAGE	是				
LANGUAGEDS	是	是			
MASSOC			是		
MASSOCTYPE			是		
MBRATTR	是				
MBRGRPTYPE	是				
OPERATOR	是				
OPERATRDSC	是	是			
ORDCHNLTY	是				
OUTPUTQ	是				
OUTPUTQDSC	是	是			
PARTROLE	是				
PARTROLED	是	是			
PLCYTYPDSC	是		是		
POLICYTYPE	是				
QTYCONVERT			是		是
QTYUNIT			是		
QTYUNITDSC	是		是		
ROLE	是				
SCHCMD	是				
SETCURR			是		
SETCURRDSC	是		是		
STATECODE			是		是
STATEPROV	是		是		
STORECGRY	是				
TAXTYPE	是				
TCSUBTYPDS	是		是		
TCSUBTYPE			是		
TCTYPE			是		
TFALGOPOL	是				
TFALGOTYPE	是				
TFALGPOLDS	是	是			
TFALGTYPDS	是	是			
TFDOMAIN	是				
TFDOMDSC	是	是			

TFSBDOMAIN	是				
TFSBDOMDSC	是	是			
TRDTYPE	是				
TRDTYPEDSC	是	是			
TXCDScheme	是				

商家資料範圍： 下圖列出商家資料範圍下的資料庫表格。

表格名稱	唯一索引				
	整數 1	整數 2	字元 1	整數 3	字元 2
ACCCMDGRP	是				
ACCCUSTEXC	是	是			
ACCOUNT	是				
ACORGPOL	是	是			
ACPOLDESC	是	是			
ACPOLICY	是				
ATTACHMENT	是				
ATTRIBUTE	是	是			
ATTRVALUE	是	是			
BASEITEM	是				
BASEITEM	是	是			
BUYERPO	是				
CALCODE	是				
CALCODEDSC	是	是			
CALCODEMGP	是	是			
CALCODTXEX	是	是			
CALMETHOD	是				
CALRANGE	是				
CALRLOOKUP	是				
CALRULE	是				
CALRULEMGP	是	是			
CALSCALE	是				
CALSCALEDS	是	是			
CATALOG	是				
CATALOGDSC	是	是			
CATCNTR	是	是			
CATCONFINF	是				
CATENCALCD	是				
CATENTATTR	是				
CATENTDESC	是	是			
CATENTREL	是	是	是		
CATENTRY	是				

CATENTSHIP	是				
CATGPCALCD	是				
CATGPENREL	是	是		是	
CATGROUP	是				
CATGRPATTR	是	是			
CATGRPDESC	是	是			
CATGRPPS	是	是		是	
CATGRPREL	是	是		是	
CATGRPTPC	是	是		是	
CATTOGRP	是	是			
CHARGETYPE	是				
CHRGTYPDSC	是	是			
CNTRDISPLY	是				
CNTRNAME	是		是		
CONTRACT	是				
CREDITLINE	是				
CRULESCALE	是	是			
CURCONVERT	是				
CURCVLIST	是	是	是		是
CURFMTDESC	是	是	是		
CURFORMAT	是		是		
CURLIST	是		是		
DISPCGPREL	是				
DISPENTREL	是				
DISTARRANG	是				
FLOWADMIN	是				
INVADJCODE	是				
INVADJDESC	是	是			
ITEMSPC	是				
ITEMVERSN	是				
JURST	是				
JURSTGPREL	是	是			
JURSTGROUP	是				
LANGPAIR	是	是		是	
LISTPRICE	是		是		
MASSOCCECE	是				
MASSOCGPGP	是				
MBRATTRVAL	是				
MBRGRPUSG	是	是			
MBRREL	是	是			
MBRROLE	是	是		是	
MGPTRDPSCN	是	是			

OFFER	是				
OFFERDESC	是	是			
OFFERPRICE	是		是		
PARTICIPNT	是				
PATTRDESC	是	是			
PATTRIBUTE	是				
PATTRPROD	是	是			
PKGATTR	是				
PKGATTRVAL	是				
PKGITEMREL	是	是		是	
POLICY	是				
POLICYCMD	是		是		
POLICYDESC	是	是			
POLICYTC	是	是			
PRODSETDSC	是	是			
PRODUCTSET	是				
PRSETCEREL	是	是			
QTYFMTDESC	是	是	是		
QTYFORMAT	是		是		
RTNDNYDESC	是	是			
RTNDNYRSN	是				
RTNDSPCODE	是				
RTNDSPDESC	是	是			
RTNREASON	是				
RTNRSNDESC	是	是			
SHPARJURGP	是	是			
SHPARRANGE	是				
SHPJCRULE	是				
SHPMODEDSC	是	是			
STENCALUSG	是	是			
STORECAT	是	是			
STORECENT	是	是			
STORECGRP	是	是			
STORECNTR	是	是			
STOREDEF	是				
STOREITEM	是	是			
STORELANG	是	是			
STOREMBRGP	是	是			
STORITMFFC	是	是		是	
STORLANGDS	是	是		是	
TAXCGRY	是				
TAXCGRYDS	是	是			

TAXJCRULE	是	是		是	
TCDESC	是	是			
TDPSCNCNTR	是	是			
TERMCOND	是				
TRADEPOSCN	是				
TRADING	是				
TRDATTACH	是	是			
TRDESC	是	是			
TXCDCLASS	是				
VENDOR	是				
VENDORDESC	是	是			
VERSIONSPC	是				
VIEWREG	是	是	是		

網站與商家資料範圍： 下圖列出網站和商家資料範圍下的資料庫表格。

表格名稱	唯一索引				
	整數 1	整數 2	字元 1	整數 3	字元 2
CMDREG	是		是		
FFMCENTDS	是	是			
FFMCENTER	是				
MBRGRP	是				
MBRGRPCOND	是				
MEMBER	是				
ORGENCY	是				
SHIPMODE	是				
STADDRESS	是				
STORE	是				
STOREENT	是				
STOREENTDS	是	是			
STOREGRP	是				
URLREG	是		是		
VIEWREG	是	是	是		

連線服務

WebSphere Commerce 使用「HTTP 程式配接器」、WebSphere MQ 配接器以及 CrossWorlds 配接器等，作為和外部系統連線之用。下列幾節說明每一個配接器，以及如何在 WebSphere Commerce 使用它們。

HTTP 程式配接器

HTTP 程式配接器可以讓外部系統藉由透過 HTTP 通訊協定來傳送 XML 要求的方式，和 WebSphere Commerce 通訊。HTTP 程式配接器可為採購系統之類的外部系統提供一種通用方式，使其可以透過 HTTP 與 WebSphere Commerce 通訊，使 WebSphere

Commerce 可以作為這些系統的供應者，來進行買方/供應商之間的交易。 HTTP 程式配接器藉由執行下列動作，來處理入埠 XML 要求：

- 辨識要求以及驗證其是否為 XML 要求。如果要求的下列三種屬性都受支援，就會將其分辨成為 XML 要求。
 - 內容類型
 - 方法
 - 字元編碼
- 受支援的要求屬性是在配接器配置中指定。
- 擷取要求的輸入串流。
- 呼叫訊息對映程式並傳遞輸入串流的内容。
- 接收 CommandProperty 物件，此物件代表由訊息對映程式傳回的 WebSphere Commerce 指令。
- 判斷適當的裝置格式，以使用來產生回應。
- 執行指令。
- 傳送 XML 回應訊息，此訊息是由 JSP 建立，且是根據所執行的指令指定的檢視名稱以及所收到的要求之裝置格式。

每個要求都會被視為個別的階段作業。訊息的認證是在訊息的控制區域中指定。在預設的情況下，HTTP 程式配接器會檢查訊息中的使用者 ID 和密碼參數，來判斷要求的確實性。 HTTP 程式配接器並不支援舊式訊息，因為舊式訊息不支援認證規格。

HTTP 程式配接器的生命週期和 WebSphere Commerce 實例相同。除非配接器的配置參數被移除或配接器被停用，不然它會在案例啟動時起始設定，並且只要案例在執行中，它就會存在。

註：在預設的情況下，「HTTP 程式配接器」會停用。

如需有關 WebSphere Commerce 如何處理自裝置收到要求的架構資訊，請參閱 *WebSphere Commerce 程式設計手冊與指導教學*。如需配置「HTTP 程式配接器」以及其他連線服務的相關資訊，請參閱 WebSphere Commerce 正式作業線上說明。

WebSphere MQ 接收程式

「WebSphere MQ 接收程式」是 WebSphere Commerce 的一個元件，它透過以 WebSphere MQ 處理入埠訊息的方式，來與後端系統整合。「MQSeries 接收程式」是包括使用 WebSphere MQ 擷取 MQ 訊息，以及使用「程式配接器」，會被呼叫來執行那些訊息。

「WebSphereMQ 接收程式」有一組預先定義的訊息，可協助將 WebSphere Commerce 商業處理程序與後端或外部系統處理程序整合起來。每則入埠訊息都會啟動 WebSphere Commerce 中的處理程序，以更新資料庫表格或執行其他作業。如果需要所提供的訊息的詳細資訊，請參閱 WebSphere Commerce 線上說明中的後端整合與供貨整合訊息資訊。另外，除了現有的預先定義的訊息外，「WebSphere MQ 接收程式」也支援訊息擴充和新訊息。

WebSphere MQ 作為中介軟體：「WebSphereMQ 配接器」可讓您利用 WebSphere MQ 作為中介軟體，將後端與外部系統和 WebSphere Commerce 整合。「WebSphereMQ 配接器」讓 WebSphere Commerce 能夠接收來自後端系統和外部系統的訊息。受支援的軟

體為 WebSphere MQ 5.3 或更新版，含 WebSphere MQ 用戶端產品擴充。WebSphere MQ 用戶端是 WebSphere MQ 的用戶端，可以單獨下載。

您可以透過使用連結或用戶端模式，將 WebSphere MQ 設定為您的中介軟體。若 WebSphere Commerce 是安裝在和 WebSphere MQ 伺服器相同的機器上，並且該伺服器透過 WebSphere MQ Java，使用 Java 原生介面 (JNI)，連接到 WebSphere MQ 伺服器，就應該使用連結模式。由於在通訊方面是經由將 JNI 呼叫導向到佇列管理程式 API，而非經由網路，所以連結模式所提供的效能要比使用網路連線的用戶端模式來得好。若 WebSphere Commerce 安裝在某部機器上，並且 WebSphere MQ 伺服器安裝在後端系統上時，就應該使用用戶端模式。

如要驗證 WebSphere MQ 連線、佇列與通道，請執行測試程式，將訊息置於 WebSphere MQ 佇列，並從其中取得訊息。如需細資訊，請參閱 WebSphere MQ 文件與 *WebSphere Commerce 附加軟體手冊*。

CrossWorlds 配接器

CrossWorlds 配接器提供一種新的機制，可以延伸 IBM WebSphere Commerce 商務和 IBM CrossWorlds 的整合。這個新的配接器使用 CrossWorlds Server Access Interface (一個應用程式設計介面, API)，可讓外部程序執行 IBM CrossWorlds InterChange Server (CW ICS) 中的共同作業。使用這個配接器時，WebSphere Commerce 可以透過 IBM CrossWorlds 傳送同步訊息到這些系統，來和外部系統整合。這個配接器可以讓 WebSphere Commerce 傳送訊息到 CrossWorlds 伺服器，並等候回覆。取得回應之後，呼叫指令可以繼續處理其他商業邏輯。

IBM CrossWorlds 系統是一個軟體整合產品套件，其中包括一些預先建立的商業邏輯範本，稱為共同作業，供處理一般商業整合需求，以及包括一些開發與管理工具。共同作業會定義及自動化一般業界特定的商業程序步驟，如訂單管理以及生產物料清單管理。共同作業也可以用來調整以及延伸不同企業軟體產品的商業程序，以及協助兩者之間做有意義的資料交換。

註：如需有關 CrossWorlds 配接器的詳細資訊，請參閱 *WebSphere Commerce 附加軟體手冊*以及 WebSphere Commerce 正式作業與開發作業線上說明。

WebSphere Commerce 與 IBM CrossWorlds InterChange Server 整合：

WebSphere Commerce 設計為和 IBM CrossWorlds InterChange Server (ICS) 整合。這使得 WebSphere Commerce 可以透過 ICS 和後端系統整合。WebSphere Commerce 和 ICS 整合得以支援三種型態的訊息流程：

- 離埠非同步訊息流程可讓 WebSphere Commerce 傳送訊息到 ICS 進行處理。
- 入埠訊息流程可讓 WebSphere Commerce 自 ICS 接收訊息。
- 同步訊息流程可讓 WebSphere Commerce 傳送要求/回應訊息到 ICS。

離埠流程和入埠流程都是使用 WebSphere Business Integration (WBI) Adapter for WebSphere Commerce。要求/回應流程則使用 WebSphere Commerce SAI Adapter for ICS。

WebSphere Commerce 離埠程序： WebSphere Commerce 中的異動可以觸發動作，使訊息透過 WebSphere Commerce 傳訊系統傳送到其他應用程式。透過傳訊系統產生的訊息，可以用來經由 WebSphere Business Integration Server 這類的整合伺服器，和其他應用程式整合。例如，可以從 WebSphere Commerce 產生 OrderCreate XML 訊息，以便在後端系統中建立訂單。

在 WebSphere Commerce 中建立訂單時，WebSphere Commerce 可以配置為將訂單相關資訊傳送到其他系統，做進一步的處理，如訂單供貨。使用 IBM Websphere Business Integration Server 的客戶可以配置 Websphere Commerce 和 IBM CrossWorlds ICS，使其將來自 Websphere Commerce 的 XML 訊息傳送到 IBM CrossWorlds ICS 進行處理。在此情況下，訂單會透過 XML 訊息，從 WebSphere Commerce 傳送到 ICS。此 XML 訊息可以由 IBM CrossWorlds ICS 進行處理，然後傳送到其他系統，如 SAP。

當您在 WebSphere Commerce 中下訂單時，將會產生一則 XML 格式的 OrderCreate 訊息，並放置在 WebSphere MQ 輸出佇列中。WBI Adapter for the WebSphere Commerce 代理程式會不斷地輪詢取得新訊息，然後將新訊息從 WebSphere MQ 輸出佇列傳遞到連接器控制程式。

控制程式會接收 WebSphere Commerce 特定的商業物件，然後啟動對映，以產生同屬商業物件 (GBO)。這些 GBO 會被傳遞到對應的共同作業，該共同作業隨即處理那些 GBO，並將要求傳送到後端系統。如需有關將 Websphere Commerce 和 IBM CrossWorlds ICS 整合的詳細資料，請參閱 WBI Adapter for Websphere Commerce 的文件，網址為：<http://www-3.ibm.com/software/websphere/crossworlds/library/doc/v411/welcome.html>。

WebSphere Commerce 入埠程序: Websphere Commerce 設計為和 IBM CorssWorlds ICS 整合，以支援將要求從 ICS 送到 WebSphere Commerce。WebSphere Commerce 傳訊系統會啟動 WebSphere Commerce 商業邏輯，以支援來自 ICS 的要求。

在這個實務中，將會使用現有的 WebSphere Commerce 訊息，以啟動來自 IBM CrossWorlds ICS 的商業邏輯觸發要求。

WBI Adapter for WebSphere Commerce 會從共同作業接收商業物件，並使用資料處理常式將商業物件轉換為 XML 格式的訊息，然後將訊息遞送到 WebSphere MQ 佇列。如需有關將 Websphere Commerce 和 IBM CrossWorlds ICS 整合的詳細資料，請參閱 CrossWorlds 配接器的文件，網址為：<http://www-3.ibm.com/software/websphere/crossworlds/library/doc/v411/welcome.html>。

配置作業

配置傳訊服務

在管理系統時，「網站管理者」可以執行下列作業：

- 新增用來傳遞訊息的機制（稱為「傳輸」）
- 配置傳輸
- 檢視訊息傳輸指定
- 配置訊息類型（新增、變更或刪除訊息說明、嚴重性、傳輸或裝置格式）
- 新增商店使用的傳輸方法
- 啟動或停用傳輸方法

如需有關配置傳訊系統的詳細資訊，請參閱 WebSphere Commerce 正式作業線上說明。

配置排程服務

排程器可從管理主控台存取。如需使用排程器的其餘資訊，請參考 WebSphere Commerce 正式作業線上說明。有關維護排程器的其他資訊，請參閱第 143 頁的『維護排程器』。

配置日誌記載

所有日誌記載以及追蹤配置現在都改為在 WebSphere Application Server 管理主控台中完成。下列各節僅涵蓋單獨使用的工具：

- 修改 ECMessage 啟動選項
- 修改 ECTrace 啟動選項

如需有關 WebSphere Application Server 日誌記載與追蹤的詳細資訊，請參閱 WebSphere Application Server Information Center。

單獨使用工具の日誌記載

修改 ECMessage 啟動選項：「網站管理者」可以使用 WebSphere Application Server 設定日誌記載，或直接修改 XML 檔案。

編輯 instance_name.xml 配置檔：訊息是根據它們的嚴重性來過濾的。載入期間的訊息日誌配置是以 XML 配置檔為基礎：

► Windows

drive:\Program Files\WebSphere\CommerceServer\instances\instance_name\logs\ecmsg.log

► AIX

/usr/WebSphere/CommerceServer/instances/instance_name/logs/ecmsg.log

► Solaris ► Linux

/opt/WebSphere/CommerceServer/instance/instance_name/logs/ecmsg.log

► 400

/QIBM/UserData/WebCommerce/instances/instance_name/logs/ecmsg.log

下列是 instance_name.xml 檔中用來配置 ECMessage 記載的 messageLog 元素範例：

```
<messageLog
  fileSize="40"
  messageFile="e:\wcs_wsad\instances\demo\logs\ecmsg.log"
  notification="false"
  notificationClassName="com.ibm.commerce.messaging.ras.ErrorNotificationHandler">
<logSeverity type="ERROR"/>
<logSeverity type="WARNING"/>
<logSeverity type="STATUS"/>
<logSeverity type="DEBUG"/>
<logSeverity type="INFO"/>
</messageLog>
</LogSystem>
```

如果 notification 屬性設為 false，除了 ERROR 外的所有訊息嚴重性都將停用。如果要停用嚴重性，請從嚴重性清單中移除那一行。如果 notification 屬性設為 false，將不會傳送任何錯誤通知。如果需要錯誤通知，notification 屬性必須設為 true。您必須適當地配置了傳訊元件後，錯誤通知才能運作。

fileSize 屬性指定日誌檔的最大檔案大小 (MB)。如果實際檔案大小超出了指定的數量，將開啓新的檔案。檔名的建立是根據下列命名慣例：

fileName_cloneId_timeStamp.ext。比方說，ecmsg.log 檔產生具有如 ecmsg_ploiesti_2000.08.08_18.17.47.768.log 之類名稱的系統檔，其 ploiesti 是複製 ID 且內含的時間戳記是可變因數。

擷取的訊息及追蹤資料可以放入同一檔案中。如果要進行這樣的配置，它們的 `fileSize` 屬性必須具有相同值，而且 XML 配置檔必須指向同一檔案位置。

修改 ECTrace 啟動選項：「網站管理者」可以在獨立使用的工具中，直接使用 XML 檔案來配置 ECTrace 啟動選項。請見下列各節：

編輯 `instance_name.xml` 配置檔： 載入期間的追蹤日誌配置是以 XML 配置檔為基礎：

▶ Windows

```
drive:\Program Files\WebSphere\CommerceServer55\  
instances\instance_name\logs\ecmsg.log
```

▶ AIX

```
/usr/WebSphere/CommerceServer55/instances/instance_name/logs/ecmsg.log
```

▶ Solaris ▶ Linux

```
/opt/WebSphere/CommerceServer55/instance/instance_name/logs/ecmsg.log
```

▶ 400

```
/QIBM/UserData/WebCommerce/instances/instance_name/logs/ecmsg.log
```

下列是 `instance_name.xml` 檔中用來配置 ECTrace 追蹤的 `trace` 元素範例：

```
<trace  
  fileSize="40"  
  traceFile="e:/wcs_wsad\instances\demo\logs\ecmsg.log"  
  <component name="SERVER"/>  
  <component name="EJB"/>  
</trace>
```

`fileSize` 屬性指定日誌檔的最大檔案大小 (MB)。如果實際檔案大小超出了指定的數量，將會開啓一個新的檔案，並由 `traceFile` 提供檔案的基本名稱。檔案名稱是根據下列命名慣例建立：`file name_clone ID_time stamp.ext`。例如，`ecmsg.log` 這個檔案會產生的系統檔案名為 `ecmsg_ploiesti_2000.08.08_18.17.47.768.log`，其複本 ID 為 `ploiesti`，內嵌的時間戳記是一個可變的因素。

擷取的訊息及追蹤資料可以放入同一檔案中。如果要進行這樣的配置，它們的 `fileSize` 屬性必須具有相同值，而且 XML 配置檔必須指向同一檔案位置。

啓用追蹤元件

如要啓用 WebSphere Commerce 系統追蹤，您需要使用 WebSphere Application Server 管理主控台。

在伺服器啓動時啓用追蹤：

1. 開啓 WebSphere Application Server 管理主控台。如需有關如何執行此動作的資訊，請參閱 WebSphere Application Server Information Center。
2. 在主控台導覽樹中，按一下 **疑難排解 > 日誌記載與追蹤**，然後按一下 **伺服器 > 診斷追蹤**。
3. 按一下 **配置**。
4. 選取 **啓用追蹤** 勾選框，以啓用追蹤，或清除勾選框，以停用追蹤。

5. 輸入適當的 TraceString，將追蹤規格設定為屬意的狀態。
6. 選取要將追蹤輸出導入檔案，或導入內建的記憶體循環緩衝區。
7. 針對所產生的追蹤選取屬意的格式。
8. 儲存已變更的配置。
9. 啟動伺服器。

註：如需有關任何這些步驟的詳細資訊，請參閱 WebSphere Application Server InfoCenter。

在執行中的伺服器上啟用追蹤： 您可以使用下列程序，修改追蹤服務的狀態，決定目前執行中的伺服器有哪些元件正在做追蹤。

1. 開啟 WebSphere Application Server 管理主控台。如需有關如何執行此動作的資訊，請參閱
2. 在主控台導覽樹中，按一下疑難排解 > 日誌記載與追蹤，然後按一下伺服器 > 診斷追蹤。
3. 選取執行時期標籤。
4. 將追蹤規格變更為屬意的狀態，變更現有的追蹤狀態。
5. （選用的）如果希望變更現有的追蹤輸出，則配置追蹤輸出。
6. 按一下套用。

註：如需有關任何這些步驟的詳細資訊，請參閱 WebSphere Application Server InfoCenter。

WebSphere Commerce 追蹤元件： 下列是 WebSphere Commerce V5.5 中的追蹤元件：

表 6.

追蹤元件	WebSphere JRAS 擴充追蹤日誌程式
SERVER	com.ibm.websphere.commerce.WC_SERVER
CATALOG	com.ibm.websphere.commerce.WC_CATALOG
DATASOURCE	com.ibm.websphere.commerce.WC_DATASOURCE
ORDER	com.ibm.websphere.commerce.WC_ORDER
USER	com.ibm.websphere.commerce.WC_USER
COMMAND	com.ibm.websphere.commerce.WC_COMMAND
CF	com.ibm.websphere.commerce.WC_CF
NEGOTIATION	com.ibm.websphere.commerce.WC_NEGOTIATION
RAS	com.ibm.websphere.commerce.WC_RAS
DB	com.ibm.websphere.commerce.WC_DB
METAPHOR	com.ibm.websphere.commerce.WC_METAPHOR
SCHEDULER	com.ibm.websphere.commerce.WC_SCHEDULER
DEVTOOLS	com.ibm.websphere.commerce.WC_DEVTOOLS
TOOLSFRAMEWORK	com.ibm.websphere.commerce.WC_TOOLSFRAMEWORK
RULESYSTEM	com.ibm.websphere.commerce.WC_RULESYSTEM
MERCHANDISING	com.ibm.websphere.commerce.WC_MERCHANDISING
MARKETING	com.ibm.websphere.commerce.WC_MARKETING

表 6. (繼續)

REPORTING	com.ibm.websphere.commerce.WC_REPORTING
TRANSPORT_ADAPTER	com.ibm.websphere.commerce.WC_TRANSPORT_ADAPTER
PERFMONITOR	com.ibm.websphere.commerce.WC_PERFMONITOR
MESSAGING	com.ibm.websphere.commerce.WC_MESSAGING
STOREOPERATIONS	com.ibm.websphere.commerce.WC_STOREOPERATIONS
CACHE	com.ibm.websphere.commerce.WC_CACHE
EVENT	com.ibm.websphere.commerce.WC_EVENT
EJB	com.ibm.websphere.commerce.WC_EJB
CURRENCY	com.ibm.websphere.commerce.WC_CURRENCY
CATALOGTOOL	com.ibm.websphere.commerce.WC_CATALOGTOOL
CONTRACT	com.ibm.websphere.commerce.WC_CONTRACT
UBF	com.ibm.websphere.commerce.WC_UBF
BI	com.ibm.websphere.commerce.WC_BI
INVENTORY	com.ibm.websphere.commerce.WC_INVENTORY
UTF	com.ibm.websphere.commerce.WC_UTF
RFQ	com.ibm.websphere.commerce.WC_RFQ
EXCHANGE	com.ibm.websphere.commerce.WC_EXCHANGE
ACCESSCONTROL	com.ibm.websphere.commerce.WC_ACCESSCONTROL
AC_UNITTEST	com.ibm.websphere.commerce.WC_AC_UNITTEST
APPROVAL	com.ibm.websphere.commerce.WC_APPROVAL
COLLABORATION	com.ibm.websphere.commerce.WC_COLLABORATION
THREAD	com.ibm.websphere.commerce.WC_THREAD
SENSITIVE_INFO	com.ibm.websphere.commerce.WC_SENSITIVE_INFO

JCA 連接器追蹤元件： 下列是 WebSphere Commerce V5.5 中的 JCA 連接器追蹤元件：

表 7.

追蹤元件	WebSphere JRAS 擴充追蹤日誌程式
JCA 電子郵件連接器	com.ibm.websphere.commerce.jcaemail
JCA 檔案連接器	com.ibm.websphere.commerce.jcafile
JCA JMS 連接器	com.ibm.websphere.commerce.jcajms
JCA 範例連接器	com.ibm.websphere.commerce.jcasample

Commerce Payments 追蹤元件： 下列是 WebSphere Commerce V5.5 中的 Commerce Payments 追蹤元件：

表 8.

追蹤元件	WebSphere JRAS 擴充追蹤日誌程式
* (提供所有資訊)	com.ibm.websphere.commerce.payments.*
MPF (Payments 組織架構)	com.ibm.websphere.commerce.payments.MPF
MPFUI (Payments UI)	com.ibm.websphere.commerce.payments.MPFUI

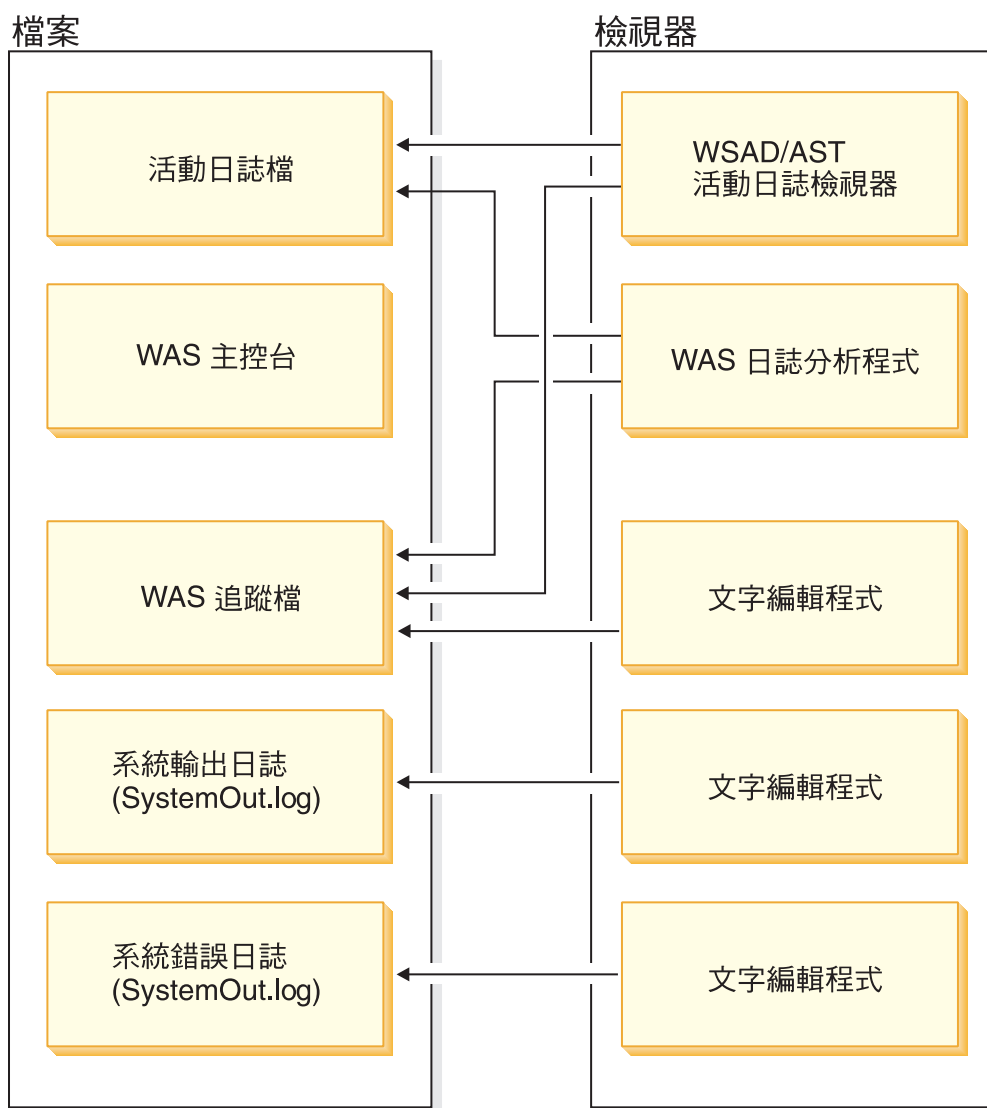
表 8. (繼續)

<p><Cassette_name> (可以追蹤任何 IBM 或協力廠商的卡匣。WebSphere Commerce 隨附的卡匣包括：VisaNet、Paymentech、BankServACH、OfflineCard 以及 CustomOffline。)</p>	<p>com.ibm.websphere.commerce.payments.<Cassette_name></p>
--	--

註：如需有關 WebSphere Commerce Payments 的詳細資訊，請參閱 WebSphere Commerce Payments Library。

檢視日誌檔案

下圖指出日誌檔案可以使用不同的方式檢視：






使用「日誌分析程式」

「日誌分析程式」是屬於 WebSphere Application Server 的一部份，此元件會取得一或多個服務或活動的日誌，合併所有資料，然後再顯示那些項目。這個工具會根據它的

症狀資料庫，分析以及解譯日誌項目中的事件或錯誤狀況，協助您診斷問題。「日誌分析程式」具有一項特殊功能，可以用來從 IBM 網站下載最新的症狀資料庫。如要下載最新的更新到症狀資料庫中，請在「日誌分析程式」中使用「檔案 -> 更新資料庫 -> WebSphere Application Server 症狀資料庫」選項（如果是 WebSphere Application Server），如果是 WebSphere Application Server Network Deployment，則選取 WebSphere Application Server 網路部署症狀資料庫」選項。

服務與活動日誌： WebSphere Application Server 會從各種 WebSphere Application Server 元件的活動，建立服務或活動日誌檔案。服務或活動日誌檔案 `activity.log` 是一個二進位檔案，此檔案位於：`installation_directory/logs/activity log`。您可以使用「日誌分析程式」檢視服務或活動日誌檔案，也可以將服務或活動日誌檔案合併為一個日誌檔。服務或活動日誌不能使用文字編輯程式察看。「日誌分析程式」工具的設計目的就是要用來檢視這個檔案。如需有關 `activity.log` 的詳細資訊，請參閱 *WebSphere Application Server System Administration Guide*。

使用「日誌分析程式」檢視服務與活動日誌： 如果要使用「日誌分析程式」檢視服務或活動日誌，請執行下列步驟：

1. 變更至 `WASinstall_dir/bin` 目錄。
2. 執行 `waslogbr` Script 檔：
 -  `v waslogbr.bat`。
 -   `v waslogbr`。

此動作會啟動「日誌分析程式」介面。

3. 選取 **檔案 > 開啓**。
4. 瀏覽到包含服務或活動日誌檔案的目錄。
5. 選取服務或活動日誌檔案，然後按一下 **開啓**。
6. 如果要分析其記錄，以滑鼠右鍵按一下左邊樹狀結構中的某一筆記錄，然後從快速功能表中選取 **UnitOfWorkView**，再選取 **分析**。


這時記錄旁若有一個綠色勾號，表示該記錄符合症狀資料庫中的一筆記錄。當您選取有勾號的記錄時，就可以在右下窗格中看到其問題說明。如需有關「日誌分析程式」使用者介面的詳細資訊，請參閱 *WebSphere Application Server Information Center*。

使用文字檔案

如果要檢視日誌檔，您可以使用任何文字編輯程式。您可以在安裝 WebSphere Commerce 伺服器的 `logs` 目錄中找到日誌檔。通常這個檔案是位於 `../appserver/logs/` 目錄中，不過可加以變更。下列是您可以用來尋找日誌檔的路徑範例。

`WAS_installdir/logs/WC_instanceName/trace.log`

註：  使用 `vi` 以外的文字編輯程式，來檢視所產生的 WebSphere Commerce 日誌檔。這些字行對於 `vi` 太長，因此，無法適當地檢視。

配置 LDAP

一旦安裝了 WebSphere Commerce，您就可以選擇使用 LDAP 伺服器作為成員儲存庫，或者您可以先使用資料庫作為成員儲存庫，稍後再切換為使用 LDAP 伺服器。如需有關配置 LDAP 的詳細資訊，請參閱 *WebSphere Commerce 附加軟體手冊* 中的「針對 LDAP 配置 WebSphere Commerce」。

配置規則服務

如果要配置規則服務，請登入「管理主控台」，然後從**規則服務**功能表按一下**管理**。「網站管理者」可以使用「管理主控台」執行下列規則服務作業：

- 新增規則服務
- 編輯規則服務
- 檢查規則服務的狀態
- 重新整理規則服務
- 刪除規則服務
- 啟動規則服務
- 停止規則服務

如需有關配置規則服務作業的詳細資訊，請參閱 WebSphere Commerce 線上說明。

配置暫置作業伺服器

使用暫置作業伺服器公用程式

暫置作業公用程式存在於暫置作業伺服器上，並使用標準 DB2/JDBC 通訊，同時連接至正式作業及暫置作業資料庫。管理者必須確定 DB2 主從架構環境已設定妥當。DB2 用戶端會透過 TCP/IP 埠與 DB2 伺服器交談。號碼設定在 DB2 實例變數 SVCENAME 中。您必須注意這個埠號，並確定 proxy 將不會阻擋這個埠。在預設的情況下，DB2 使用 50000。如果需要設定 JDBC 和 DB2 主從架構環境的詳細資訊，請參閱 *DB2 管理手冊*。

一旦您的環境設定完畢，且您準備開始將暫置作業伺服器中的資料複製至正式作業伺服器時，請使用「暫置延伸」公用程式來推進資料。可以複製的表格列示在暫置表格中。為自訂的表格配置暫置作業伺服器後，您已建立來儲存自訂資料的任何新表格也會複製到暫置作業伺服器。

STAGLOG 表格充當一個內部日誌。每當您在暫置作業伺服器上變更表格中的記錄時，就會有一個觸發指令將這個變更記錄在 STAGLOG 表格中。對於每一個修改的記錄，觸發指令會記錄修改類型（插入、刪除或更新）、記錄常駐之表格的名稱，以及記錄的主要鍵或唯一索引。在變更及測試暫置作業伺服器上的資料庫記錄後，請使用「暫置延伸」公用程式，將變更延伸回正式作業伺服器。

在暫置階段作業期間，暫置作業伺服器所涵蓋的資料庫表格絕不能在正式作業資料庫上更新。當您利用暫置複製公用程式來複製正式作業資料庫到暫置作業資料庫時，暫置階段作業便告開始。當您使用「暫置複製」公用程式來開始另一個暫置階段作業時，暫置階段作業就會結束。在您的資料從正式作業資料庫複製至暫置作業資料庫後，就暫置作業伺服器所涵蓋的表格而言，暫置作業資料庫與正式作業資料庫已同步。一旦表格同步，將不允許變更正式作業資料庫上的表格。您僅能更新暫置作業資料庫，然後使用「暫置延伸」公用程式，將變更延伸至正式作業資料庫。如果您更新

這兩個資料庫，您的延伸可能會失敗，因為可能會發生鍵值衝突或參照完整性違規。如果您必須在暫置階段作業中更新正式作業資料庫，請利用暫置複製公用程式來同步化這些資料庫，再起始一個新的暫置階段作業。

如果要保證在暫置階段作業期間，絕不會更新正式作業資料庫上的表格，這些表格必須只能由「網站管理者」來控制。在某些情況下，在使用暫置複製來複製了資料後，正式作業資料庫中的暫置表格將遭到個別客戶或商家的更新。比方說，在複製到正式作業資料庫後，您無法禁止商家修改正式作業資料庫中的 OFFER 表格。在這種情況下，您無法使用暫置作業伺服器。不過，RFQ 物件是一個例外。當您在正式作業資料庫上建立 RFQ 物件時，有幾列將插入正式作業資料庫中的交易表格。如果您正在暫置作業資料庫中建立合約，您也會插入幾列至暫置作業資料庫中的交易表格。在這種情況下，您是在更新暫置作業資料庫與正式作業資料庫上的相同表格。

請注意，當使用 RFQ 物件時，對於暫置作業伺服器有一些限制。您也需要執行「暫置檢查」公用程式來尋找潛在的唯一索引鍵衝突，並加以更正，然後執行「暫置延伸」公用程式，將變更延伸至正式作業資料庫。

在典型的大眾消費型商務網站中，表格可以分成兩個群組：配置資料及作業資料。配置表格含有如商店、型錄、型錄項目、語言、稅金，以及折扣之類的資料。這些表格都是在「網站管理者」控制之下；個別客戶無法修改表格。作業表格含有如客戶資訊、地址、訂單，以及 SET 相關資料之類的資料。客戶可以修改作業表格。暫置作業伺服器僅涵蓋配置表格。請參閱 WebSphere Commerce 暫置表格，以取得暫置作業伺服器所涵蓋的表格清單。

您也必須確定暫置作業伺服器所涵蓋的表格未含有作業表格的任何外來鍵參照。不然，延伸可能會失敗，因為可能會從正式作業資料庫刪除主要鍵。在使用暫置作業伺服器之前，您應該確定只有組織擁有作業資料，而不是個別客戶，如型錄管理者。

在使用暫置作業伺服器之前，您應該注意下列事項：

- 暫置記錄所參照的任何新影像檔、HTML 檔或 JSP 檔都必須從暫置作業伺服器中，手動複製到正式作業伺服器中。
- 暫置作業伺服器無法複製及延伸資料庫綱目變更、影像檔、HTML 檔或 JSP 檔。比方說，如果您在暫置作業資料庫中建立新的索引或表格，您必須在正式作業資料庫中，以手動方式來建立這個索引或表格。
- 「暫置延伸」公用程式無法延伸「載入器」套件（載入模式）或 DB2 Load 公用程式所載入的記錄，因為這兩者會略過暫置觸發指令。如果您使用了這兩個公用程式之一，請使用「暫置複製」，將您的資料庫表格重新同步化，然後開始新的暫置階段作業。在暫置階段作業期間，請絕對不要在暫置作業資料庫或正式作業資料庫上使用「載入器」套件（載入模式）或 DB2 Load。
- 使用「暫置複製」公用程式之後，您必須停止並重新啟動暫置作業伺服器。
- 暫置作業伺服器不支援 DB2 Text Extender。
- 除了清理 STAGLOG 表格，請不要在暫置作業伺服器上執行「資料庫清理」公用程式。
- 有一些暫置作業伺服器限制。在使用暫置作業伺服器之前，確定您瞭解暫置作業伺服器限制。詳細資訊，請參閱第 32 頁的『暫置作業伺服器限制』。

暫置複製公用程式：「暫置複製」公用程式 (iSeries 為 stagingCopy.sh) 會將正式作業資料庫中的資料複製到暫置作業資料庫中。您可以將資料複製在網站相關表格、商家

相關表格或個別表格中。在執行「暫置複製」公用程式之前，您也可以指令語法中使用 `cleanup_stage_db` 參數，來清理暫置作業資料庫。如果您指定了 `yes`，則在複製資料之前，「暫置複製」公用程式將清理所有暫置表格。這可能會影響其他具有刪除連鎖反應的表格。如果您指定了 `no`，「暫置複製」公用程式將不會從暫置表格刪除任何資料。如果您的複製資料在主要鍵或唯一索引上產生衝突或重複的鍵值，則您的複製可能會失敗。如果要使用「暫置複製」公用程式，僅清理暫置作業資料庫，請指定 `only`。

「暫置複製」公用程式和「暫置延伸」公用程式會將資料庫資料分成兩個範圍層次：網站相關範圍和商家相關範圍。網站範圍包括系統中所有商家共用的資料。比方說，系統所使用的語言，以及國家或地區代碼。商家範圍包括與個別商家相關的資料。比方說，商店資訊是針對個別商家自訂的，而且來自商店表格的列可以是每一個商家的特定列。某些資料庫表格同時含有網站及商家資訊。在執行「暫置複製」公用程式期間，如果您將 `scope` 參數指定為 `_all_`，則在複製網站資料後，將接著複製所有商家資料。如果您將 `scope` 指定為 `_site_`，僅會複製網站資料。如果您將 `scope` 指定為 `_merchant_`，僅會複製商家資料。請注意，您無法複製個別商家的資料，僅能複製所有商家的資料。如果您未將 `scope` 設為 `_all_`，請先複製網站資料，然後再複製商家資料，因為所有商家都會使用網站資料。不然，您的複製將失敗，因為外來鍵與主要鍵不符。當您使用 `cleanup_stage_db` 來清理網站資料時，請注意，因為刪除連鎖反應，所以會刪除商家資料。如果未將 `scope` 設為 `_all_`，您應該在清理商家資料後，再清理網站資料，然後在複製網站資料後，再複製商家資料。

「暫置複製」公用程式的另一個選項就是 `script_file` 參數。藉由指定 `script` 檔名稱，「暫置複製」公用程式將產生一個 SQL `script` 檔，根據指定的範圍，使用匯出及匯入，將正式作業資料庫複製至暫置作業資料庫。如果您使用預設值，或將 `cleanup_stage_db` 指定為 `yes`，也會產生刪除陳述式來清理暫置作業資料庫。`script` 檔位於您啟動「暫置複製」公用程式所在的目錄中。`script` 檔會使用匯出及匯入來加快資料庫複製程序的速度。您也可以變更產生的 `script` 檔，來修改「暫置複製」公用程式的行為。比方說，您可以變更 `script` 檔，使用 DB2 載入公用程式，來代替匯入公用程式，進一步加快複製程序的速度。請注意，產生的 `script` 會將所有表格匯出至您執行公用程式所在的目錄。請確定您具有足夠的磁碟空間。

瞭解交易範圍是很重要的。當清理暫置作業資料庫時，「暫置複製」公用程式會在清理每一個表格後確定交易。當複製資料時，「暫置複製」公用程式會在複製每一個表格並將 `KEYS` 表格同步後確定交易。對於產生的 `script`，交易範圍會因為 DB2 匯入公用程式而略有不同。在完成匯入後，DB2 匯入公用程式會自動確定交易。將 `KEYS` 表格同步前，會先確定交易。結果，在個別交易中會完成 `KEYS` 表格的同步。

您可以使用 `dbtable` 參數，來指定要清理或複製的表格。請注意，當您指定要清理或複製的表格時，可能無法隔離表格。某些表格會透過參照限制而彼此相關。如果您清理指定的表格，您也將透過刪除連鎖反應而清理子項表格。如果您複製指定的表格，首先您應該複製母項表格。不然，您的清理或複製將會失敗。

「暫置複製」公用程式是可以配置且延伸。如果要處理自訂的表格，您的表格必須符合某些條件，而且您必須在暫置配置表格中設定。您必須先遵循配置資料庫的步驟，才可以使用「暫置複製」公用程式。如果指令成功，「暫置複製」公用程式會刪除 `STAGLOG` 表格中的所有記錄。

附註：如果 `RFQ` 在您的系統上，您將無法使用「暫置複製」指令。詳細資訊請參閱第 32 頁的『暫置作業伺服器限制』。

暫置檢查公用程式: 當配置和作業資料共用同一表格時，暫置作業資料庫與正式作業資料庫之間可能會發生唯一索引鍵衝突。在將變更延伸至正式作業資料庫前，請先使用「暫置檢查」指令，若是 iSeries 則使用 `stagingCheck.sh`，判斷是否有潛在的唯一索引鍵衝突，並加以更正，然後再進行延伸。

當您在正式作業資料庫上使用 RFQ，且在暫置作業資料庫上建立合約時，就是在更新這兩個資料庫上的相同表格。比方說，「網站管理者」在暫置作業資料庫上建立一個合約，這將在暫置作業資料庫上的 TRADING 表格（以及其他格）中插入一列。同時，使用者在正式作業資料庫上建立一個 RFQ，這將在正式作業資料庫上的 TRADING 表格（以及其他格）中插入一列。這兩列在 TRADING 表格中可能具有相同的唯一索引值。將合約從暫置作業資料庫延伸至正式作業資料庫時，會產生一個唯一索引鍵衝突，因此延伸將失敗。延伸之前，請使用「暫置檢查」公用程式來尋找發生衝突的唯一索引鍵，並更正它們。完成後，您便可以延伸變更。

當您使用「暫置檢查」公用程式時，請將 `-scope` 參數指定為 `_unique_index_`，以檢查暫置作業資料庫中的 `delta` 變更是否有潛在的鍵值衝突。對於所有插入及更新作業，它將檢查 STGUINDTAB 表格中指定的所有表格是否有潛在的索引鍵衝突。對於每一個表格，它將察看所有唯一索引，並檢查正式作業資料庫與暫置作業資料庫之間是否有潛在的鍵值衝突。如果有潛在的鍵值衝突，它將報告表格名稱、唯一索引，以及衝突的鍵值。

「暫置檢查」指令不會變更您的資料庫；它會報告必須解決的潛在鍵值衝突。當使用這個指令時，請將 `-sourcedb` 參數指定為暫置作業資料庫。如果您指定了正式作業資料庫作為來源資料庫，則「暫置檢查」指令將無法正常運作。

暫置檢查公用程式可供配置與延伸。您可以在 STGUINDTAB 表格中新增多個表格或您自訂的表格，並執行「暫置檢查」指令，以驗證是否有任何潛在的鍵值衝突存在。

附註：恆確定您的配置及作業資料不共用同一表格。

暫置延伸公用程式: 在變更及測試暫置作業伺服器上的資料庫記錄後，請使用「暫置檢查」公用程式來檢查是否有潛在的唯一索引鍵衝突，並更正它們。現在您可以將變更延伸至正式作業資料庫。

「暫置延伸」公用程式或 iSeries 為 `stagingProp.sh`，會將暫置作業資料庫中的變更移到正式作業資料庫中。「暫置延伸」公用程式會使用 STAGLOG 表格，來識別暫置作業資料庫中的已變更記錄，然後在正式作業資料庫中更新這些記錄。經過處理的記錄，會在 STAGLOG 表格的 STGPROCESSED 直欄內以 1 表示。

您可以指定 `scope` 參數，來選取要延伸的資料類型。設為 `_site_`，所有變更的網站資料將從暫置作業資料庫延伸至正式作業資料庫。設為 `_merchant_`，將延伸所有商家的已變更資料。您無法延伸個別的商家資料。設為 `_all_`，將同時延伸網站和商家資料。

使用 `dbtable` 參數，您可以延伸特定的表格。在指定表格之前，確定已延伸母項表格。

「暫置延伸」公用程式的交易範圍不同於「暫置複製」公用程式。每一次執行「暫置延伸」公用程式指令都會當作一個交易來計數。比方說，如果您將 `scope` 指定為 `_site_`，則「暫置延伸」公用程式將對所有修改過的網站資料開始新的交易，且在成功延伸後確定交易。如果延伸失敗，延伸將會回復，因此您的正式作業資料庫的狀況將與之前的狀況相同。

「暫置延伸」公用程式是可以配置且延伸。在延伸您自訂的表格之前，表格必須符合某些條件。如果需要詳細資訊，請參閱「為自訂的表格配置暫置作業伺服器」。您必須先遵循配置資料庫的步驟，才可以使用「暫置延伸」公用程式。

配置資料庫


使用「暫置複製」公用程式、「暫置延伸」公用程式或「資料庫清理」公用程式之前，可能需要執行下列步驟：

註：

1. 下列步驟僅是建議而已，並且是在「暫置複製」公用程式發現問題時，才應該使用。

2. 下列說明不適用於 iSeries，除非另以  指出

1. 設定 PATH 環境變數。

2.      如果您是使用 DB2 資料庫，請發出下列指令，來配置暫置作業資料庫和正式作業資料庫：

```
db2 update db config for db_name using LOGPRIMARY 80
db2 update db config for db_name using LOGBUFSZ 512
db2 update db config for db_name using DBHEAP 2048
db2 update db config for db_name using APPLHEAPSZ 2048
db2 update db config for db_name using PCKCACHESZ 8200
```

其中 *db_name* 是您資料庫的名稱。





註：

a. 預設 STMTHEAP 大小為 60000。

b. 預設 LOCKLIST 為 2400。


c. 預設 STAT_HEAP_SZ 為 2048。

d. 預設 APP_CTL_HEAP_SZ 為 4096。

3.     增加緩衝池大小，以改善效能。根據您的 DB2 資料庫大小和可用的記憶體，來判定最佳的緩衝池大小。執行下列指令，來變更預設緩衝池大小：

```
db2 connect to db_name
db2 alter bufferpool IBMDEFAULTBP size n
db2 terminate
```

其中 *n* 是最佳緩衝池大小。

4.  以具有 `secofr` 權限，並且其 `cssid` 不是 65535 的使用者設定檔來登入。

針對自訂的表格配置暫置作業伺服器

如果要使用暫置作業伺服器與您自訂的資料庫表格搭配，請執行下列配置：

1. 識別您自訂的表格範圍（網站資料、商家資料或網站和商家資料）。

2. 使用對應的觸發指令範例，根據您的表格範圍和索引類型，為您的資料庫表格建立觸發指令。

3. 將自訂的表格插入 STGSITETAB、STGMERTAB 以及 STGMRSTTAB 表格。

- 若是網站表格，僅插入 STGSITETAB。
- 若是商家表格，僅插入 STGMERTAB。

- 若是同時含有網站和商家資料的表格，插入 STGSITETAB、STGMERTAB 以及 STGMRSTTAB。

附註：您必須確定所有母項表格都已適當地插入，而且母項表格的 TABNBR 直欄少於子項表格的 TABNBR 直欄。如果您自訂的表格是 WebSphere Commerce 表格的母項表格，您也需要確定您表格的 TABNBR 直欄少於子項表格的 TABNBR 直欄。

配置連線

如需有關配置各種配接器的資訊，請參閱 *WebSphere Commerce 附加軟體手冊* 以及 *WebSphere Commerce 正式作業線上說明*。

配置 WebSphere Commerce Payments

配置 Payments 之前，請先執行下列步驟：

- 確定 WebSphere Commerce Payments 已經隨 WebSphere Commerce 安裝作業安裝好。
- 建立一個 WebSphere Commerce Payments 實例。
- 使用「配置管理程式」新增一個卡匣到 WebSphere Commerce Payments 實例中。
- 啓動 WebSphere Commerce Payments 實例以及 WebSphere Commerce 實例。
- 建立一個商家以及該商家的「商家管理者」。

如果要配置卡匣並使用 Payments，您必須以「商家管理者」身份登入 WebSphere Commerce Payments。WebSphere Commerce Payments 會自動安裝 CustomOffline Cassette 以及 OfflineCard Cassette。這些卡匣支援的基本組織架構為 WebSphere Commerce Payments 5.5 版。如需詳細資訊，請參閱 *WebSphere Commerce 安裝手冊*。

下列是 WebSphere Commerce 與 Payments 的預設安裝目錄：

表 9. 預設安裝目錄

預設安裝目錄	說明
<i>WC_installdir</i>	Windows : <i>drive:\WebSphere\CommerceServer55</i> AIX : <i>/usr/WebSphere/CommerceServer55</i> Solaris、Linux : <i>/opt/WebSphere/CommerceServer55</i> 400 : <i>/QIBM/ProdData/CommerceServer55</i>

表 9. 預設安裝目錄 (繼續)

預設安裝目錄	說明
<i>Payments_installdir</i>	Windows : <i>drive:\WebSphere\CommerceServer55\payments</i> AIX : <i>/usr/lpp/WebSphere/CommerceServer55/payments</i> Solaris 、 Linux : <i>/opt/WebSphere/CommerceServer55/payments</i> 400 : <i>/QIBM/ProdData/CommercePayments/55</i>

如需有關配置 WebSphere Commerce Payments 的詳細資訊，請參閱第 163 頁的附錄 C, 『WebSphere Commerce Payments 指導教學』。

第 4 章 代管型商店管理

在代管型商店實務內容中，網站管理者需要執行幾項作業。本章依下列種類列出網站管理者應執行的作業：

- 管理轉銷商
- 管理配銷商
- 網站管理

管理轉銷商

在代管型商店實務內容中，網站管理者要負責設定轉銷商組織以及維護與轉銷商的關係。這項協調牽涉到下列作業：

- 傳送新轉銷商組織登錄的電子郵件核准。

當轉銷商登錄他們的組織後，網站管理者要負責傳送電子郵件給轉銷商，通知他們的組織申請已被核准。核准實際的組織登錄是在「組織管理主控台」中執行。如需「組織管理主控台」以及可在其中執行的作業詳細資訊，請參閱 *WebSphere Commerce* 正式作業線上說明。

當轉銷商組織經過核准之後，位在 Demand 鏈商業模型中的轉銷商就可以使用 *WebSphere Commerce Accelerator* 修改他們的商店。不過，每一個商店都是在關閉狀態下建立，轉銷商可以使用 *WebSphere Commerce Accelerator* 更新商店和變更商店，最後再開放商店。在「代管」商業模型中，每一個商店都是建立在暫停狀態，並且需要由「網站管理者」或「通路經理」將其回復。在此情況下，商家要等到其商店回復之後，才可以變更商店。如需有關商業模型的詳細資訊，請參閱 *WebSphere Commerce 基礎入門* 以及 *WebSphere Commerce 商店程式開發手冊*。如需有關 *WebSphere Commerce Accelerator* 的詳細資訊，請參閱 *WebSphere Commerce* 正式作業線上說明。

- 將商家代管商店核准傳送給商家。（只適用於使用「代管」商業模型的商店）

設定好代管商店之後，網站管理者要以電子郵件將代管商店申請核准傳送給商家。這個電子郵件中必須包含新的代管商店的鏈結，以及如何管理商店的指示。

註： 如需有關管理轉銷商組織的詳細資訊，請參閱 *WebSphere Commerce* 正式作業線上說明中的「管理轉銷商組織」。

管理配銷商

在代管型商店實務內容中，網站管理者要負責設定配銷商以及建立與轉銷商的代理關係。這項管理牽涉到下列作業：

- 建立新的配銷商。

網站管理者可以透過建立配銷商服務合約的方式，來建立新的配銷商。需要加入合約中的資訊可以直接向 IBM 洽詢取得。如需有關配銷商服務合約的詳細資訊，請參閱 *WebSphere Commerce* 正式作業線上說明。

註： 這項作業也可以由通道管理程式執行。

- 變更配銷商設定。

有時網站管理者可能會收到要求變更配銷商的某些設定，如配銷商的連線逾時設定。配銷商的設定可以透過 SQL 陳述式存取資料庫，加以變更。其他資訊請參閱 *WebSphere Commerce 商店程式開發手冊*。

- 配置與安裝 CrossWorlds 的配接器

CrossWorlds 配接器是用來在 WebSphere Commerce 以及外部與代管商店的後端系統之間，建立交談的連接器。網站管理者要負責配置以及安裝配接器，以確保轉銷商和配銷商網站之間的連線。也可以建立使用 WebSphere MQ，作為 WebSphere Commerce 與 SAP R/3 後端系統之間的交談，並且必須由網站管理者設定。如需有關 IBM CrossWorlds 以及 WebSphere MQ 的詳細資訊，請參考 *WebSphere Commerce 附加軟體手冊*。

網站管理

代管商店特定的網站管理作業包括下列：

- 操作暫置作業伺服器。如需其他資訊，請參閱第 32 頁的『暫置作業伺服器』。
- 操作 DBClean。有關進一步資訊，請參閱第 91 頁的『資料庫清理公用程式』。
- 操作 WebSphere Commerce Analyzer。如需取得其餘資訊，請參閱 WebSphere Commerce 安裝 CD。
- 啟動及停止 WebSphere Commerce 伺服器。有關進一步資訊，請參閱第 11 頁的『啟動及停止 WebSphere Commerce』。

第 5 章 動態快取

本節說明動態快取 WebSphere Commerce servlet 或 JavaServer Pages (JSP) 結果。

WebSphere Commerce servlet 或 JSP 結果快取

當客戶點選某個鏈結以檢視產品或種類頁面時，大部分的時間都是花在剖析 HTTP 要求、存取資料庫以及動態建立頁面。當網站流量大，而且資料庫中的產品和種類項目繁多時，會進一步增加載入 servlet 或 JavaServer Pages (JSP 檔案) 所需要的時間。

伺服器上的大部份 HTTP 要求都是有關型錄資訊。WebSphere Commerce 指令 `--CategoryDisplay`、`ProductDisplay`、`TopCategoriesDisplay` 以及 `StoreCatalogDisplay` 會從資料庫中擷取資訊，再將結果顯示成 JSP 頁面。如果型錄資訊自從前次檢視之後都沒有變動過，當下次客戶要求該頁面時，Servlet 或 JSP 檔案就不需要再以動態方式重建。提供儲存在快取中的相同靜態 servlet 或 JSP 檔案，在速度上會加快許多。Servlet 或 JSP 快取的建構方式，是在位於 Web 模組的 `WEB-INF` 目錄內的 `cachespec.xml` 檔案中定義快取項目元素。

如果存取頁面的對應快取項目不存在於記憶體中，即會以動態方式產生它。之後，該頁面會儲存在記憶體內，且在其據以產生的資料沒有變動時，不需要重新產生。

註：WebSphere Commerce 動態快取以及 URL 重新編寫無法交互運作。如果您啓用 URL 重新編寫，則必須停用 WebSphere Commerce 快取，並且不可以快取 servlet 或 JSP 檔案。如需有關 URL 重新編寫的詳細資訊，請參閱 *WebSphere Commerce Security Guide* 中的階段作業管理一章。

WebSphere Application Server 動態快取

WebSphere Commerce 利用 WebSphere Application Server 動態快取服務來快取 servlet 或 JSP 檔案，以及自 WebSphere Application Server `CacheableCommand` 介面延伸的指令。在建立 WebSphere Commerce 實例時，都會根據預設，啓用動態快取服務、servlet 快取以及磁碟解除負載。

如需有關 WebSphere Application Server 動態快取的詳細資訊，請參考 WebSphere Application Server Information Center 中的主題『透過動態快取服務改進效能』(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

動態快取

快取 servlet 或 JSP 檔案結果可以提高應用程式效能。WebSphere Application Server 會將多種快取活動，包括 servlet、Web 服務以及 WebSphere 指令，合併在一個服務中，稱為動態快取。這些快取活動會一起合作提高應用程式效能，並且會共用應用程式伺服器的動態快取服務中設定的許多配置參數。

您可以使用動態快取，從內建的記憶體處理要求，來提高 servlet 以及 JSP 檔案的效能。快取項目中包含 servlet 輸出、servlet 執行結果以及 meta 資料。

如需詳細資訊，請參考 WebSphere Application Server Information Center 中的主題『透過動態快取服務改進效能』

(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

啓用動態快取服務與 **servlet** 快取

如果要啓用快取，您應該要啓用動態快取服務以及配置 **servlet** 快取。如需執行這些步驟的資訊，請參閱 WebSphere Application Server Information Center 中的主題『啓用廣域動態快取服務』以及『配置 **servlet** 快取』

(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

啓用動態快取監視器

WebSphere Application Server 有提供一支「動態快取監視器」應用程式，可以顯示快取統計值、臨界以及磁碟統計值、快取項目、相依關係 ID 以及快取原則資訊。如果要檢查 WebSphere Application Server 動態快取的內容和行爲，您應該要安裝 WebSphere Application Server 動態快取監視器。如果要使用「動態快取監視器」，請安裝 `CacheMonitor.ear` 檔案，這個檔案位於使用動態快取的每一部應用程式伺服器上，`WAS_install_dir` 目錄之下的 `installableApps` 子目錄中。

對於 WebSphere Commerce，基於安全理由，建議您使用虛擬主機 `VH_instance_name_admin` 作為「動態快取監視器」（例如，`VH_demo_admin`），而不要使用 `VH_instance_name`。您可以從「WebSphere Application Server 管理主控台」新增或變更虛擬主機名稱。

您可以使用下列網址，以 Web 瀏覽器存取 Web 應用程式：

`http://host_name:port/cachemonitor`

不過，基於安全理由，建議您存取管理主機機器：

`https://admin_host_name:port/cachemonitor`

例如，如果使用虛擬主機 `VH_instance_name_admin` 安裝「快取監視器」，則可以依下列方式存取「快取監視器」：

`https://admin_host_name:8002/cachemonitor`

註：

1. 您必須針對每一個 WebSphere Commerce 實例，啓動一個動態快取監視器。
2. 每當您需要使用「快取監視器」，並且您已經啓用 WebSphere Application Server EJB 安全性，您就必須執行 *WebSphere Commerce Security Guide* 中的「動態快取監視器」的安全配置」一節說明的某些額外設定。

如需有關安裝「動態快取監視器」的詳細資訊，請參閱 WebSphere Application Server Information Center 中的主題「顯示快取資訊」

(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

WebSphere 動態快取監視器的相依關係 ID 樹狀檢視畫面

WebSphere Commerce 有提供一項「樹狀檢視畫面」工具（屬於「快取監視器」的延伸），可以提供記憶體中的相依關係 ID 之樹狀檢視畫面。「樹狀檢視畫面」工具可以提供相依關係 ID 的樹狀檢視畫面。您可以展開或收合任何樹狀結構節點，取得階層性檢視畫面，以及將該節點下的快取項目群組作廢。原始的檢視畫面不會做任何變動，只會出現一個新的鏈結「相依關係 ID 樹狀檢視畫面」。

如需有關安裝與使用「樹狀檢視畫面」工具的資訊，請參閱以下目錄內的 `readme.txt` 中的「WebSphere 動態快取監視器相依關係 ID 樹狀檢視畫面」：

`WC_installdir/features/dynacache/CacheMonitor/`

 `WC_installdir\features\dynacache\CacheMonitor\`

配置可快取的物件

可快取的物件是在 `cachespec.xml` 檔案中定義，這個檔案位於 Web 應用程式保存 (WAR) WEB-INF 或 Enterprise Bean WEB-INF 目錄內。您可以在應用程式伺服器內容目錄中放置一個廣域的 `cachespec.xml` 檔案，不過，建議將快取配置檔和部署模組放在一起。`cachespec.xml` 檔案的根元素為 `<cache>`，其中包含 `<cache-entry>` 元素。

如果要指定 servlet 或 JSP 結果快取的快取項目，請在 `cachespec.xml` 檔案中加入下列區段：

```
<cache-entry>
  <class>servlet</class>
  <name>name</name>
  ...
</cache-entry>
```

其中的 `name` 是 servlet 或 JSP 的相對 Web 路徑或 servlet 對映

如果要指定指令快取的快取項目，請在 `cachespec.xml` 檔案中加入下列區段：

```
<cache-entry>
  <class>command</class>
  <name>name</name>
  ...
</cache-entry>
```

其中 `name` 是指令類別的完整路徑，例如，`com.ibm.commerce.dynacache.commands.MemberGroupsCacheCmdImpl`。

註： 在本版次中，只支援一個指令作廢功能。

在 `<cache-entry>...</cache-entry>` 元素中，您可以開發 `cache-ID`、`dependency-ID` 以及作廢規則。如果要快取物件，WebSphere Application Server 必須能為該物件的不同呼叫產生唯一的 ID。`<cache-id>` 元素就是用來執行這項作業。每一個快取項目可以擁有多個快取 ID 規則，這些規則會依序執行，直到某個規則傳回非空白的快取 ID 或沒有規則可以執行為止。如果所有快取 ID 產生規則都無法產生有效的快取 ID，就不會快取該物件。

動態快取會對這個檔案中的變更做回應。當偵測到 `cachespec.xml` 的新版本時，就會取代舊原則。透過舊原則檔案快取的物件並不會從快取中自動作廢。那些物件會以新的原則重覆使用，或透過其取代演算法，從快取中清除。如需有關 `cachespec.xml` 檔案的詳細資訊，請參閱 WebSphere Application Server Information Center 中的主題「Cachespec.xml 檔案」

(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。此外，亦請參閱 *WebSphere Commerce 商店程式開發手冊* 中，有關開發商店的其他注意事項。

快取預設值

安裝期間，WebSphere Commerce 會以下列預設值設定快取系統：

- 動態快取已啟用。
- 磁碟解除負載已啟用。
- Servlet 快取已啟用。
- 每一個商業模型的預設快取原則（提供作為範例）會針對下列顯示指令，快取 JSP 檔案：
 - CategoryDisplay
 - ProductDisplay
 - StoreCatalogDisplay
 - TopCategoriesDisplay

您必須手動合併以下的快取項目：



```
WC55_installdir/samples/dynacache/business_model/cachespec.xml
```

複製到：

```
WAS_installdir\installedApps\cell_name\WC_instance_name.ear\Stores.war\WEB-INF\cachespec.xml
```



```
WC55_installdir\samples\dynacache\business_model\cachespec.xml
```

複製到：

```
WAS_installdir\installedApps\cell_name\WC_instance_name.ear\Stores.war\WEB-INF\cachespec.xml
```

如需有關在商店中啟用快取原則的明細，請參考 *WebSphere Commerce 範例商店手冊* 或 *WebSphere Commerce 商店程式開發手冊*。

整頁與片段快取

WebSphere Application Server 動態快取用來快取 JSP 檔案的方法，是根據 JSP 如何撰寫而定。如果特定 WebSphere Commerce 指令的頁面輸出一律是根據 URL 參數和要求屬性產生相同的結果，則此頁面輸出可以使用內容 elementconsume-subfragments (CSF)，加上以 WebSphere Commerce 控制程式 servlet (com.ibm.commerce.server.RequestServlet) 作為伺服器名稱，以快取項目快取輸出。當快取項目是以這種方式定義時，則頁面輸出的快取方法就稱為整頁快取。將 consume-subfragments 和控制程式 servlet 搭配使用的最大優點是在效能，但如果使用這個機制快取網頁，網頁輸出將不會有個人化的資訊。

如果頁面輸出中有部分是和使用者相關時，則其頁面輸出的快取方法就稱為片段快取。亦即，JSP 頁面會快取成爲個別的快取項目，再於接到要求時組合起來。如果是片段 (JSP) 快取，WebSphere Commerce 必須要先執行指令，以判斷應該要執行哪一個 JSP，動態快取機制才能判定是否能從快取提供 JSP。這個方法的優點是彈性大，因爲這可以根據使用者資訊，組合不同的快取項目來組成頁面。

整頁快取

若使用內容元素 `consume-subfragments` (CSF) 時，母項項目（標示為 CSF 者）會在 其快取項目中包括所有片段的所有內容，這會形成一個很大的快取項目，但是沒有併入或轉遞，只有項目的整個樹狀結構的內容。

當快取 `servlet` 時，只會儲存該 `servlet` 的內容。快取會包括快取併入或轉遞的任何其他 片段的位置保留器。當快取有包括子項 `servlet` 時，`Consume-subfragments` (CSF) 會告 知快取不要停止儲存內容。母項項目（標示為 CSF 者）會包括其快取項目中所有片段 的所有內容，這會形成一個很大的快取項目，但是沒有併入或轉遞，只有項目的整個 樹狀結構的內容。如此可以節省許多應用程式伺服器處理時間，但是通常只適用於外 部 HTTP 要求中有包含用來判斷併入片段的整個樹狀結構所需要的資訊時。

例如，如果 `<cache-entry>` 定義如下所示：

```
<cache-entry>
  <class>servlet</class>
  <name>com.ibm.commerce.server.RequestServlet.class</name>
  <property name="consume-subfragments">true</property>
  <property name="save-attributes">false</property>
  <property name="save-cookies">false</property>

  <!-- StoreCatalogDisplay?storeId=<storeId> --->
  <cache-id>
    <component id="" type="pathinfo">
      <required>true</required>
      <value>/StoreCatalogDisplay</value>
    </component>
    <component id="storeId" type="parameter">
      <required>true</required>
    </component>
  </cache-id>
</cache-entry>
```

注意：當 `save-attributes` 內容設定為 `false` 時，要求屬性並不會隨快取項目一起儲 存。當 `save-cookies` 內容設定為 `false` 時，要求 `cookies` 並不會隨快取項目一起儲 存。

在上述範例中，快取 `servlet` 項目會包含一個已耗用的併入 `StoreCatalogDisplay.jsp`， 這是由 `StoreCatalogDisplay` 指令轉遞的 JSP 檔案。

片段快取

每一個以動態形式併入的 JSP 檔案都會有其自己的 `<cache-entry>` 定義在 `cachespec.xml` 檔案中，以便於收到要求時，由動態快取提供。不然的話，每一個動態 併入的 JSP 檔案都要針對每一個要求重新執行。例如，試想如果 `StoreCatalogDisplay.jsp` 動態地併入 `header.jsp` 以及 `footer.jsp`，並且您只為 `StoreCatalogDisplay.jsp` 設定一個 `<cache-entry>`。然後，當您要求 `StoreCatalogDisplay` 頁面時，如果 `header.jsp` 和 `footer.jsp` 兩個檔案沒有快取存 在，則會執行這兩個檔案。以下是如何針對 `StoreCatalogDisplay.jsp` 定義 `<cache-entry>` 的範例：

```
<cache-entry>
  <class>servlet</class>
  <name>/ToolTech/ShoppingArea/CatalogSection/CategorySubsection/StoreCatalogDisplay.jsp</name>
  <property name="save-attributes">false</property>

  <cache-id>
    <component id="storeId" type="parameter">
      <required>true</required>
    </component>
  </cache-id>
</cache-entry>
```

```

</component>
  <component id="catalogId" type="parameter">
    <required>false</required>
  </component>
  </cache-id>
</cache-entry>

```

快取 servlet 過濾

在 WebSphere Commerce 5.5 版之前，WebSphere Commerce 有提供自己的快取機制。使用先前的機制時，網頁可以根據兩種方法快取：

- 階段作業獨立 (SI) 快取 - 頁面依據 URL 參數快取
- 階段作業相依 (SD) 快取 - 頁面依據 URL 參數、使用者的語言、偏好的貨幣、母項組織、合約 ID 以及成員群組等做快取。

SI 快取的快取 ID 是依據 URL 參數產生，如果是 SD 快取，其快取 ID 則是依據 URL 參數加上階段作業資訊兩者建立。

為了提供和先前的階段作業相依快取相同的功能，但是使用 WebSphere Application Server 動態快取機制，WebSphere Commerce 5.5 引進了 servlet 過濾器，稱為快取過濾器。此快取過濾器的設計目的是要以階段作業資訊設定要求屬性，以便讓動態快取用來建構快取 ID。由於階段作業資訊是由 WebSphere Commerce 伺服器執行時期設定，因此快取過濾器要等到第二個要求到達網站時，才能設定所有要求屬性。

下列表格列出由快取過濾器設定的要求屬性：

表 10. 由快取過濾器設定的要求屬性：

要求屬性	說明
DC_curr	使用者偏好的貨幣
DC_lang	使用者偏好的語言
DC_porg	使用者的母項組織
DC_cont	使用者的目前合約
DC_mg	使用者的明確成員群組
DC_storeId	商店 ID
DC_userId	使用者的 ID

由於使用者可能有權使用多份合約，並且隸屬於多個成員群組，因此其要求屬性 DC_cont 和 DC_mg 可能會包含多個值。如果是這類使用者，則會將其值排序，並且連續排列在一起，之間以一個分號 (";") 作為分隔字元。此外，也會定義多份合約和成員群組要求屬性。（例如，DC_cont0, DC_cont1, ... DC_contN 其中 N 是使用者有權使用的合約份數）。比方說，如果使用者有權使用合約 10004 和 10005，則會設定下列要求屬性：DC_cont 為 10004;10005，DC_cont0 為 10004，DC_cont1 為 10005。

設定要求屬性 DC_cont 的目的是要建構具有有限元件數目的快取 ID，而設定個別要求屬性 DC_cont0、DC_cont1, ..., DC_contN 的目的是要建構相依關係 ID，作為更詳細的快取作廢之用。

由於成員群組資訊並未包括在階段作業資料中，因此快取過濾器必須依據使用者 ID，從資料庫擷取這項資訊。為了避免因重複查詢資料庫而導致的效能降低，快取過濾器使用 WebSphere 「指令快取」來完成這項工作。新指令 -

com.ibm.commerce.dynacache.command.MemberGroupsCacheCmdImpl - 直接從「WebSphere 指令組織架構」延伸，用來依據使用者 ID，快取使用者隸屬的成員群組。WebSphere Commerce 會藉由在 cachespec.xml 中加入下列快取項目，通知動態快取服務來快取這個指令：

```
<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.dynacache.commands.MemberGroupsCacheCmdImpl</name>
  <cache-id>
    <component type="method" id="getUserId">
      <required>true</required>
    </component>
  </cache-id>

  <dependency-id>DC_userId</dependency-id>

  <dependency-id>DC_userId
    <component type="method" id="getUserId">
      <required>true</required>
    </component>
  </dependency-id>
</cache-entry>
```

WebSphere Application Server 動態快取會以其 get 方法 getUserId() 作為其快取 ID，來快取類別 MemberGroupsCacheCmdImpl 的指令物件。因此，下一次針對相同使用者 ID 的成員群組要求出現時，動態快取就會傳回已經計算好的成員群組快取指令物件。之後，這項成員群組資訊就隨時可以透過指令方法 getMemberGroups() 存取，快取過濾器會呼叫這個指令，以便在要求屬性 DC_mg 中設定此使用者的成員群組。

註：雖然 WebSphere Commerce 在上述的快取過濾器內部使用 WebSphere 「指令快取」（例如和 MemberGroupsCacheCmdImpl 配合使用），不過，WebSphere Commerce 並沒有正式支援使用 WebSphere Commerce 指令的 WebSphere 「指令快取」。

商店頁面快取策略

如需設計商店頁面時，需要注意的快取考量資訊，請參閱 *WebSphere Commerce 商店程式開發手冊* 中的『快取商店頁面』一節。這包括下列問題的指引：

- 應該快取什麼頁面？
- 應該快取整個頁面或快取頁面片段？
- 發展一套快取策略

此外，亦請參閱 *WebSphere Commerce 商店程式開發手冊* 中，有關商店關係對快取的隱含意義討論一節。

WebSphere Commerce 中的快取作廢機制

由 WebSphere Application Server 動態快取服務產生的快取項目中，所包含的資訊包括 servlet 輸出、servlet 執行結果（如呼叫其他 servlet 或 JSP 檔案），以及有關如逾時與項目優先順序資訊的 meta 資料。WebSphere Commerce 需要不斷地驗證在快取中的這項資訊是否是最新的、是否一致與正確。因此 WebSphere Commerce 需要一套有效率的機制，來識別和移除不再有效的快取資訊。WebSphere Application Server 動態快取服務有提供支援，它採用一個稱為快取作廢的程序，來維護快取資訊。快取作廢可讓 WebSphere Commerce 運用由服務提供的各種作廢機制，作不同情況的作廢動作。這些機制包括規則型、時間型、群組型以及程式化方式。

本節僅著重於說明如何在 WebSphere Commerce 中執行快取作廢，而無意涵蓋一般的快取作業。本節說明由 WebSphere Application Server 動態快取提供的作廢機制：

- 在 `cachespec.xml` 中定義作廢原則
- 透過由 WebSphere Application Server 提供的「快取監視器」移除快取項目
- 在 WebSphere Commerce 應用程式中，程式化地啟動 WebSphere Application Server 動態快取作廢 API

在 `cachespec.xml` 中定義作廢原則

WebSphere Application Server 動態快取支援透過使用 `cachespec.xml` 檔案，配置快取作廢。作廢原則可以在和用來配置可快取的物件相同的 `cachespec.xml` 中定義，這個檔案位於 Web 應用程式保存 (WAR) `WEB-INF` 或 Enterprise Bean `WEB-INF` 目錄中。如需有關 `cachespec.xml` 語法的詳細資訊，請參閱 WebSphere Application Server Information Center 中的主題 "`cachespec.xml` 檔案" (<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

作廢規則可以在快取項目中，使用 `<invalidation>` 標籤指定，這個標籤定義將對應的快取項目作廢的事件，也可以使用 `<timeout>` 標籤，配置要保存快取項目的期限。這些標籤的定義方式和相依關係 ID 完全相同，並且會使用由作廢規則產生的 ID，將具有相同相依關係 ID 的快取項目作廢。每一個快取項目可以有多個作廢規則存在。但所有作廢規則會分別執行。

以下幾節討論在 `cachespec.xml` 中的作廢規則語法，並顯示如何在 WebSphere Commerce 中套用下列規則的範例：

- Servlet 型作廢規則
- 指令型作廢規則
- 逾時與優先順序型作廢規則

Servlet 型作廢規則

這一種由 WebSphere Application Server 動態快取提供的作廢方法會在執行 servlet 要求時，將其他快取項目作廢。

`cachespec.xml` 中的語法:

```
<cache-entry>
  <class>servlet</class>
  <name>servlet_name</name>
  <invalidation>invalidation_id
    <component id="" type="pathinfo">
      ....
    </component>
  </invalidation>
</cache-entry>
```

註:

`<class>`

值為 "servlet" 時，表示作廢是由 servlet 觸發。

`<invalidation>`

用來識別觸發作廢的事件元素，以及要將具有指定 ID 的快取項目作廢。

`pathinfo`

在 WebSphere Commerce 中，所有 servlet 要求都要通過同一個控制程式

servlet，這支控制程式為 `com.ibm.commerce.server.RequestServlet.class` 或 `com.ibm.commerce.tools.common.ToolsRequestServlet.class`。

如果要識別唯一的要求 URI，可以使用 "pathinfo" 元件類型來過濾要求中的 pathinfo 名稱。

cachspec.xml 中定義的原則範例： 在發出 InterestItemDelete URL 刪除喜好項目清單中的型錄項目時，將以參考號碼識別的喜好項目清單快取頁面作廢：

```
<cache-entry>
  <class>servlet</class>
  <name>com.ibm.commerce.server.RequestServlet.class</name>

  <invalidation>listId
    <component id="" type="pathinfo" ignore-value="true">
      <required>true</required>
      <value>/InterestItemDelete</value>
    </component>
    <component id="listId" type="parameter">
      <required>true</required>
    </component>
  </invalidation>
</cache-entry>
```

指令型作廢規則

這個方法會因為執行指令，而順便將其他快取項目作廢。這個方法提供一種事件驅動型機制，在 WebSphere Commerce 的指令層次觸發作廢。使用者可以檢查指令中撰寫的商業邏輯，找出可能會因為執行指令而受影響的特定快取物件，然後據以定義作廢規則。這種指令的準則之一為指令必須實作「WebSphere Commerce 程式設計模型」，並且如果作廢 ID 是要依據那些輸入參數產生，則指令中負責處理輸入參數的方法與欄位必須外部化。

如需此項支援的明細，請參考 WebSphere Application Server 文件。另請參閱第 83 頁的『針對指令型作廢撰寫指令』。

cachspec.xml 中的語法：

```
<cache-entry>
  <class>command</class>
  <name>fully_qualified_class_name_of_the_command</name>
  <invalidation>invalidation_rules</invalidation>
</cache-entry>
```

註：

<class>

值為 `command` 時，表示作廢是由指令觸發。

<name>

包含指令的完整的路徑。

cachspec.xml 中定義的原則範例： 在啟動 CatalogEntryUpdate 指令變更型錄項目時，將以參考號碼快取的產品顯示頁面作廢：

```
<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.catalogmanagement.commands.CatalogEntryUpdateCmdImpl</name>

  <invalidation>productId
    <component id="getCatentryId" type="method">
```

```

        <required>true</required>
</component>
    </invalidation>
</cache-entry>

```

逾時與優先順序型作廢規則

逾時機制在原則中使用 `<timeout>` 標籤，提供一種很簡單的方法，在預先設定的期限之後，移除任何快取項目。如果無法設定特定的作廢規則或觸發事件，以便在指定事件移除快取項目時，這個機制很有用。

WebSphere Application Server 動態快取服務會在修改過 Least Recently Used (LRU) 演算法指出快取滿溢時，自動移除快取項目，這個演算法的優先順序加權使用 `<priority>` 標籤，來決定要從快取中移除哪些項目。特定快取項目與其他快取項目的相對優先順序值愈高，此快取項目留在快取中的時間就比其他項目長。

當快取滿溢時，並不需要從記憶體移除項目，您可以改為配置磁碟解除負載，並將項目複製到檔案系統中供稍後使用。（位置可以自行配置。）當伺服器關閉時，快取項目也會下載到磁碟上，並且可以在伺服器重新啟動時重覆使用。此外，下列動態快取服務設定也會影響作廢程序：

快取大小

決定快取保留的項目數目上限。

預設優先順序

判定在預設的情況下，項目停留在滿溢快取中的時間

磁碟解除負載

指定是否要啓用磁碟解除負載，以及磁碟解除負載的位置。

cacheSpec.xml 中的語法:

```

<cache-entry>
    .....
    <cache-id>
        <component >.....</component>
            <priority>priority_value</priority>
            <timeout>time_in_cache</timeout>
        </cache-id>
</cache-entry>

```

cacheSpec.xml 中定義的原則範例: 在購物車快取內容建立或使用 3600 秒之後，將其移除：

```

    <cache-id>
        <component id="" type="pathinfo">
            <required>true</required>
            <value>/OrderItemDisplay</value>
        </component>
        <component id="orderId" type="parameter">
            <required>true</required>
        </component>
        <timeout>3600</timeout>
    </cache-id>

```

透過快取監視器移除快取項目

WebSphere Application Server 快取監視器提供可用來手動作廢快取項目的選項。其中有一個**清除快取**按鈕，可以清空整個快取內容。每一個快取項目都可以分別作廢，並且一組快取項目可以視為一個範本作廢（例如，和 `ProductDisplay.jsp` 相關的所有項目）。以相依關係 ID 識別的快取項目群組，也可以在監視器上做區塊作廢。

啓動動態快取作廢 API

WebSphere Application Server 動態快取提供下列 API 來支援程式化作廢：

- `com.ibm.websphere.cache.invalidateById`
- `com.ibm.websphere.cache.invalidateByTemplate`

如需有關使用這些 API 的明細，請參考 WebSphere Application Server 文件。

WebSphere Commerce 充分運用這些 API 的優點，提供自家製作的作廢外層 `DynaCacheInvalidation` 指令，在 WebSphere Commerce 中支援表格型作廢 API。這種方法會藉由處理 `CACHEIVL` 表格中的記錄，將快取物件作廢。詳細說明請參閱第 87 頁的『快取作廢 API (`DynaCacheInvalidation`)』。

針對指令型作廢撰寫指令

爲了讓動態快取能截取到指令呼叫，指令必須寫入 WebSphere 「指令組織架構」中，其實作類別是 `CacheableCommandImpl` 的延伸（位於 `com.ibm.websphere.command` 套件中）。爲了簡化指令型作廢的指令撰寫過程，WebSphere Commerce 已經更新抽象類別 `ControllerCommandImpl` 以及 `TaskCommandImpl`，使這兩個類別從 `CacheableCommandImpl` 延伸，因此從這些抽象類別延伸的任何指令，也都會從 `CacheableCommandImpl` 延伸，因此可以適用於指令型作廢。

撰寫這種指令時，您必須要知道作廢 ID，同時也需要瞭解截取呼叫到指令的作廢規則。由於作廢 ID 是依據作爲指令的輸入參數的方法與欄位產生，因此建構作廢 ID 所需要的所有方法都必須在指令介面中提供並實作。

WebSphere Commerce 中的使用指令作廢範例

下列範例顯示 WebSphere Commerce 如何使用指令作廢。當指令 `DeleteMemberGroupMemberCmdImpl`（刪除隸屬於特定成員群組的特定成員）順利執行時，動態快取就會將作廢規則中定義的快取項目作廢。在此範例中，是以 `"DC_userId:userId"` 定義，其中 `userId` 是從 `getMemberId` 方法傳回的值。（請參閱第 78 頁的『快取 servlet 過濾』中，有關 WebSphere Commerce 快取成員群組資訊的明細。）例如，`DC_userId:-1000`，`DC_userId:-1001`，依此類推。這個指令有一個 `get` 方法 `getMemberId()`，會擷取被刪除的使用者 ID，這個方法會用來計算應刪除哪些依據使用者 ID 的相依關係 ID 的快取項目。這個邏輯也會套用到指令 `AddMemberGroupMemberCmdImpl`，這個指令也有一個 `get` 方法 `getMemberId()`：

```
<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.membergroup.commands.AddMemberGroupMemberCmdImpl</name>
  <name>com.ibm.commerce.membergroup.commands.DeleteMemberGroupMemberCmdImpl</name>

  <invalidation>DC_userId
    <component type="method" id="getMemberId">
      <required>true</required>
    </component>
  </invalidation>
</cache-entry>
```

註：上述所有作廢規則都隨附在 WebSphere Commerce 的範例快取原則中。您可以在 WebSphere Commerce 安裝目錄的 `/samples/dynacache/invalidation` 子目錄中，找到更多範例作廢規則。請參閱 `README` 檔中的「Dynacache 範例作廢快取原則」，取得有關如何將作廢規則納入 `cachespec.xml` 檔案的詳細資訊。

快取作廢範例

下列範例顯示如何在 `cachespec.xml` 檔案中設定快取原則，以便為 WebSphere Commerce 中的「消費者市場」商業模型快取 `ProductDisplay` JSP 頁面，以及如何透過在相同的 XML 檔案中定義作廢規則，將快取項目作廢。此範例會定義多個相依關係 ID，以及 JSP 檔案的快取 ID 產生規則。當快取項目在不同的情況下更新時，將會使用相依關係 ID 將快取項目作廢。此範例只顯示將 `ProductDisplay` JSP 作廢所需要的原則的一部分。如需完整的範例與詳細資訊，請參閱 `WC_installdir/samples/dynacache/invalidation` 目錄中的 `README` 檔案。

```
<cache>
<cache-entry>
  <class>servlet</class>
<name>/FashionFlow/ShoppingArea/CatalogSection/CatalogEntrySubsection/ProductDisplay.jsp</name>
<property name="save-attributes">>false</property>

  <!-- 快取 ProductDisplay.jsp -->
  <cache-id>
    <component id="storeId" type="parameter">
      <required>>true</required>
    </component>
    <component id="catalogId" type="parameter">
      <required>>true</required>
    </component>
    <component id="productId" type="parameter">
      <required>>true</required>
    </component>
    <component id="DC_lang" type="attribute">
      <required>>true</required>
    </component>
    <component id="DC_curr" type="attribute">
      <required>>true</required>
    </component>
    <component id="DC_porg" type="attribute">
      <required>>true</required>
    </component>
    <component id="DC_cont" type="attribute">
      <required>>true</required>
    </component>
    <component id="DC_mg" type="attribute">
      <required>>true</required>
    </component>
  </cache-id>

  <!-- 用來將隸屬於特定商店的產品顯示快取項目 -->
  <!-- 作廢 -->
  <dependency-id>storeId
    <component id="storeId" type="parameter">
      <required>>true</required>
    </component>
  </dependency-id>

  <!-- 用來將特定產品的快取項目作廢 -->
  <dependency-id>productId
    <component id="productId" type="parameter">
      <required>>true</required>
    </component>
  </dependency-id>

  <!-- 用來將隸屬於商店中的特定型錄的 -->
  <!-- 產品顯示快取項目作廢 -->
  <dependency-id>storeId:catalogId
    <component id="storeId" type="parameter">
```

```

                <required>true</required>
</component>
    <component id="catalogId" type="parameter">
        <required>true</required>
</component>
</dependency-id>

    <!-- 用來將隸屬於特定合約的 -->
    <!-- 產品顯示快取項目作廢 -->
    <dependency-id>contractId
        <component id="DC_cont0" type="attribute">
            <required>true</required>
</component>
</dependency-id>

</cache-entry>

<cache-entry>
    <class>command</class>
    <sharing-policy>not-shared</sharing-policy>
    <name>com.ibm.commerce.catalogmanagement.commands.AddCatalogDescCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.UpdateCatalogDescCmdImpl</name>

    <!-- ***** -->
    <!-- 將型錄說明變更時，可能會受影響的 -->
    <!-- 所有產品頁面快取項目作廢 -->
    <!-- ***** -->
    <invalidation>storeId:catalogId
        <component id="getStoreId" type="method">
            <required>true</required>
</component>
    <component id="getCatalogId" type="method">
        <required>true</required>
</component>
</invalidation>

</cache-entry>

<cache-entry>
    <class>command</class>
    <sharing-policy>not-shared</sharing-policy>
    <name>com.ibm.commerce.catalogmanagement.commands.ListpriceAddCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.ListpriceDeleteCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.ListpriceUpdateCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.OfferAddCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.OfferDeleteCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.OfferUpdateCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.ProductAttributeUpdateCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.AttributeValueUpdateCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.AddListpriceCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.DeleteListpriceCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.UpdateListpriceCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.AddOfferCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.DeleteOfferCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.UpdateOfferCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.UpdateAttributeCmdImpl</name>
    <name>com.ibm.commerce.catalogmanagement.commands.UpdateAttributeValueCmdImpl</name>

    <!-- ***** -->
    <!-- 當產品更新時，將特定的產品頁面 -->
    <!-- 快取項目作廢 -->
    <!-- ***** -->
    <invalidation>productId
        <component id="getCatentryId" type="method">
            <required>true</required>
</component>
</invalidation>

```

```

</cache-entry>
<cache-entry>
  <class>command</class>
  <sharing-policy>not-shared</sharing-policy>
  <name>com.ibm.commerce.contract.commands.ContractSuspendCmdImpl</name>
  <name>com.ibm.commerce.contract.commands.ContractTCDeployCmdImpl</name>

  <!-- ***** -->
  <!-- 將特定合約中的所有產品頁面 -->
  <!-- 快取項目作廢 -->
  <!-- ***** -->
  <invalidation>contractId
    <component id="getContractId" type="method">
      <required>true</required>
    </component>
  </invalidation>
</cache-entry>

<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.tools.devtools.store.commands.StoreProfileUpdateCmdImpl</name>
  <name>com.ibm.commerce.tools.devtools.flexflow.ui.commands.impl.FlexflowUpdateCmdImpl</name>
  <name>com.ibm.commerce.store.commands.StoreOpenCmdImpl</name>
  <name>com.ibm.commerce.store.commands.StoreCloseCmdImpl</name>

  <!-- ***** -->
  <!-- 將商店更新時，隸屬於商店的 -->
  <!-- 所有產品頁面快取項目作廢 -->
  <!-- ***** -->
  <invalidation>storeId
    <component id="getStoreId" type="method">
      <required>true</required>
    </component>
  </invalidation>
</cache-entry>

<cache-entry>
  <class>command</class>
  <sharing-policy>not-shared</sharing-policy>
  <name>com.ibm.commerce.catalogimport.commands.CatalogImportJobAddCmd</name>

  <!-- ***** -->
  <!-- 將商店型錄更新時，隸屬於商店的 -->
  <!-- 所有產品頁面快取項目作廢 -->
  <!-- ***** -->
  <invalidation>storeId
    <component id="getStoreId" type="method">
      <required>true</required>
    </component>
  </invalidation>
</cache-entry>
</cache>

```

快取作廢 API (DynaCacheInvalidation)

WebSphere Commerce 排程器會定期地呼叫 DynaCacheInvalidation 指令，處理 CACHEIVL 表格中的記錄，然後再呼叫動態快取作廢 API，作廢快取項目。startTime 是從檢查點檔案擷取。檢查點檔案會寫入 *instance_name/cache* 目錄中。第一次執行 DynaCacheInvalidation 時，就會建立這個目錄。在指令暫置時，最後一筆記錄的時間戳記會寫入檢查點檔案中。

CACHEIVL 表格

DynaCacheInvalidation 指令在處理 CACHEIVL 表格時的規則如下所示：

- 如果有設定範本直欄，則 DynaCacheInvalidation 指令會呼叫動態快取作廢 API (invalidateByTemplate)，並使用名稱作為範本 ID (例如，/webapp/wcs/stores/ToolTech/ShoppingArea/CatalogSection/CategorySubsection/StoreCatalogDisplay.jsp)。
- 如果有設定 DATA_ID 直欄，但是沒有設定範本名稱，則 DynaCacheInvalidation 指令會呼叫動態快取作廢 API (invalidateById)，並且使用 DATA_ID 作為 ID (例如，StoreCatalogDisplay:storeId:10151)。
- 當呼叫動態快取作廢 API 時，API 即會將快取項目作廢。

外部快取作廢

結果快取可以推入外部快取中，如 IBM HTTP Server 或 Edge Server。外部快取作廢會在下列情況下發生：

- 快取呈現擁塞
- 發生逾時
- WebSphere Application Server 送出作廢訊息

Web 伺服器外掛程式快取

在 WebSphere Application Server 5.0 中，Web 伺服器外掛程式中包含一個內建的 Edge Side Include (ESI) 處理器。ESI 處理器可以快取整頁，也可以快取片段，提供較高的命中比例。由 ESI 處理器實作的快取是一種內建記憶體快取，不是磁碟快取。因此，當您重新啟動 Web 伺服器時，快取項目並不會儲存起來。

當 Web 伺服器外掛程式收到要求時，會將要求傳送到 ESI 處理器，若 ESI 處理器停用時則例外。(根據預設值會啟用。)如果發生快取遺漏，則會在要求中加入 Surrogate-Capabilities 標頭，並將要求轉遞到 WebSphere Application Server。如果應用程式伺服器中有啟用動態 servlet 快取，並且回應可以移動，則應用程式伺服器會在 WebSphere 外掛程式回應中傳回 Surrogate-Control 標頭。

ESI 處理器可以透過 WebSphere Web 伺服器外掛程式配置檔案 plugin-cfg.xml 配置。例如：

```
<Property Name="esiEnable" Value="true"/>
<Property Name="esiMaxCacheSize" Value="1024"/>
<Property Name="esiInvalidationMonitor" Value="true" />
```

其中：

- `esiEnable` 藉由設定值為 `false`，用來停用 ESI 處理器。在預設的情況下，ESI 會啟用。
- `esiMaxCacheSize` 是快取的最大大小，以 1K 位元組為單位。快取的預設最大大小是 1 MB。如果快取滿溢，則第一個要從快取中移除的項目為最接近有效期限的項目。（ESI 快取項目的預設有效期限是 24 小時。）
- `esiInvalidationMonitor` 指定 ESI 處理器是否應該從應用程式伺服器接收作廢。共有三種方法可以從 ESI 快取中移除快取項目：
 - 啟動項目的有效期限逾時。
 - 項目被清除，以騰出空間給新的項目。
 - 應用程式伺服器傳送一組項目的明確作廢要求。
 若要啟用這個機制，`esiInvalidationMonitor` 必須設為 `true`，並且必須在應用程式伺服器的 `installableApps` 目錄中安裝 `DynaCacheEsi.ear` 應用程式。

註：`DynaCacheEsi.ear` 應該要使用 `VH_instance_name` 作為虛擬主機。

如需有關 ESI 處理器的詳細資訊，請參閱 WebSphere Application Server Information Center 中的主題『配置 Edge Side Include 快取』(<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

簡單的檔案 servlet

在預設的情況下，WebSphere Commerce 不會快取靜態資料，如影像和 HTML。將系統內容 `com.ibm.servlet.file.esi.timeOut` 的值設為 0（零）即可。如果希望快取影像和 HTML，則可以：

- 在 `cachespec.xml` 檔案中，設定 `SimpleFileServlet` 快取項目。在 `cachespec.xml` 中設定快取項目是比較好的做法，因為在控制從 WebSphere Application Server 快取什麼時，讓您在必要時，將快取項目作廢。
- 注意：如果將所有影像當作快取項目儲存在應用程式伺服器上時，如果達到快取項目上限，可能會降低效能。建議將 `SimpleFileServlet` 快取項目的優先順序值設為比其他快取項低。
- 將系統內容 `com.ibm.servlet.file.esi.timeOut` 設定修改為不是 0 的逾時值。

有時候需要將某些影像檔作廢的原因為，在某些商業模型中，標誌和旗標常會修改。

例如：

```
<cache-entry>
  <class>servlet</class>
  <name>com.ibm.ws.webcontainer.servlet.SimpleFileServlet.class</name>
  <property name="EdgeCacheable">true</property>
  <cache-id>
    <component id="" type="pathinfo">
      <required>true</required>
    </component>
  </cache-id>
  <dependency-id>
    <component id="" type="pathinfo">
      <required>true</required>
    </component>
  </dependency-id>
  <timeout>1800</timeout>
</cache-entry>
```

以下是將影像作廢的快取項目範例。

```
<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.tools.devtools.store.commands.StoreLogoUpdateCmdImpl</name>
  <invalidation>FashionFlow/images/logo.gif</invalidation>
</cache-entry>

<cache-entry>
  <class>command</class>
  <name>com.ibm.commerce.tools.devtools.store.commands.StoreBannerUpdateCmdImpl</name>
  <invalidation>FashionFlow/images/banner.gif</invalidation>
</cache-entry>
```

問題判定

如果您在設定使用動態快取頁面時發生問題，請察看下列資訊：

- **問題：**無法存取「快取監視器」。

解決方案：確定您已經重新產生 Web 伺服器外掛程式，並且重新啟動 Web 伺服器與應用程式伺服器，以載入新的虛擬主機與埠號對映。

- **問題：**無法快取任何頁面。

解決方案：驗證 cachespec.xml 檔案是否已載入 WebSphere Application Server。如果 cachespec.xml 已載入，則應該可以從「快取監視器」看到原則，方法為按一下側面列上的快取監視器鏈結。此外，WebSphere Application Server SystemOut.log 檔案中也會顯示類似如下這一行：

```
[6/6/03 21:21:12:635 EDT] 65c60609 ConfigManager I DYNA0047I: Successfully loaded
cache configuration file D:\WebSphere\AppServer5\installedApps\buzz\WC_demo.ear\
Stores.war\WEB-INF/cachespec.xml.
```

注意：上面這行為方便顯示而分割成兩行。

- **問題：**僅快取到部份頁面。

解決方案：這大部分是和 cachespec.xml 檔案有關。您可以開啓追蹤，然後存取在調查的頁面。追蹤應該會指出其問題。

如需有關啓用追蹤的資訊，請參閱第 59 頁的『啓用追蹤元件』。

如要開啓「動態快取」追蹤（WebSphere Commerce 端），**追蹤指定**應該要包含 "com.ibm.websphere.commerce.WC_CACHE=all=enabled"。

如要開啓「動態快取」追蹤（WebSphere Application Server 端），**追蹤指定**應該要包含 "com.ibm.ws.cache.*=all=enabled"。

- **問題：**DynaCacheInvalidation 指令無法執行作廢。

解決方案：驗證指令是否能從 URL 執行。如果您需要從 URL 執行指令，則 cacheinvalidation.jsp 檔案必須位於 docpath。即使該檔案不存在路徑中，指令仍舊會執行，不過您不會收到一個顯示頁面，指出作業已完成。如果 URL 呼叫正確無誤，您可以驗證指令是否已透過 WebSphere Commerce 排程器排定執行，預設值是立即執行。

- **問題：**透過相依關係 ID 作廢未正確執行。

解決方案：驗證屬意的 servlet 或 JSP 檔案的相依關係 ID 是位於相同的快取項目區塊內。如果是，您必須確定在不同快取項目區塊中的相同相依關係 ID 的多重定義，必須具有相同的定義。亦即，如果您的相依關係 id 是針對如下所示的一些快取項目定義，則位在任何其他快取項目區塊中的相依關係 ID 的定義都必須完全相同：

```
<dependency-id>storeId  
  <component id="storeId" type="parameter">  
    <required>true</required>  
  </component>  
</dependency-id>
```

第 6 章 系統維護





本章討論資料庫清理公用程式，以及如何使用此公用程式來維護您的系統上的資料庫。



資料庫清理公用程式

「資料庫清理公用程式」可用來從資料庫中同時刪除多個物件。如果您變更了資料庫中的許多資訊，並且有一些未用的表格或列時，就可以執行此作業。您可以刪除資料庫中的多種物件。如需有關這些物件的詳細資訊，請參閱第 108 頁的『清理資料庫』。

當「資料庫清理公用程式」刪除物件時，物件表格中的記錄也會一併刪除，以維持資料庫的參照完整性。「資料庫清理公用程式」指令會以兩種方式之一清理資料庫：由上往下或由下往上。由上往下法會採用刪除連鎖反應，刪除下層表格的所有列。如果參照完整性中有指定刪除限制，刪除連鎖反應會失敗，您就必須改由下往上方法。

如要使用由下往上方法，請在     的指令語法 FORCE 參數中，指定 yes，以便先刪除下層表格，然後再刪除母項表格。

另一種觸發由下往上方法是在指令語法中，指定     LOGLEVEL 參數為 2。指定 0 不會做記載，指定 1 只會記載上層表格的刪除陳述式。

    LOGLEVEL 2 會記載從每一個被刪除的下層表格到上層表格的刪除陳述式。雖然選取 2 會觸發由下往上方法，但如果參照完整性有指定刪除限制時，並不能保證會順利刪除。如要刪除具有刪除限制的記錄，請指定

    FORCE 參數為 yes。

如果表格中包含許多下層表格，由下往上方法的回應時間會較長。例如，MEMBER 表格中包含 500 個以上的下層表格。則基於效能考量，我們建議使用由上往下方法。

註：僅在 FORCE 參數為 no 並且 Loglevel 0 或 1 失敗時，才使用 Loglevel 2 及/或 FORCE 參數為 yes。

「資料庫清理公用程式」可以配置，也可以延伸和調整。除了可以預先設定清理配置外，您也可以新增物件到資料庫表格中，定義要清理的表格和列。請參考新增配置到「資料庫清理公用程式」。

如果您藉由建立新表格，延伸了您的資料庫綱目，您可以使用「資料庫清理公用程式」來清理新表格。如果您變更您的資料庫綱目（例如新增直欄到某個表格、變更外來鍵與主要鍵的關係，或者新增下層表格到參照完整性路徑中），「資料庫清理公用程式」都會自動處理變更。如果您變更直欄名稱，請更新 CLEANCONF 表格中的配置資料。

「資料庫清理公用程式」會根據資料庫綱目中定義的參照完整性刪除規則，刪除下層表格中的記錄。您可以將刪除規則設定為使用刪除連鎖反應、使用刪除設定空值或使

用刪除限制。如果您要新增表格，請確定已正確定義參照完整性和刪除規則。不然的話，「資料庫清理公用程式」將無法使用您的新表格。

附註：「資料庫清理公用程式」只可用來在暫置作業伺服器上，清理 staglog 物件。暫置作業資料庫和正式作業資料庫不同。暫置作業資料庫只擁有配置資料，但沒有作業資料。刪除配置資料可能會導致引發連鎖反應，而刪除作業資料。「暫置延伸公用程式」將刪除延伸到正式作業資料庫時，可能會導致連鎖反應刪除作業資料（您需要保留這些資料）。如果要清理配置資料，請在正式作業資料庫上執行「資料庫清理公用程式」。

「資料庫清理公用程式」物件

有指定特定的物件與物件類型時，「資料庫清理公用程式」會參照 CLEANCONF 表格，來判斷應該刪除哪些表格和列。下列表格說明 CLEANCONF 表格預先配置的刪除實務內容。您可以在 CLEANCONF 表格中加入類似的列，配置您自己的刪除物件。

物件	類型	陳述式
account	obsolete	delete from account where markfordelete = 1 and trdtype_id = 0 and trading_id not in (select account_id from trading) and trading_id not in (select distinct account_id from ordpaymthd)
address	obsolete	delete from address where status = 'T' and (days(CURRENT TIMESTAMP) - days(lastcreate)) >= ? and (address_id not in (select distinct address_id from orderitems where address_id is not null)) and (address_id not in (select distinct address_id from orders where address_id is not null)) and (address_id not in (select distinct allocaddress_id from orderitems where allocaddress_id is not null))

物件	類型	陳述式
atp_inventory	obsolete	delete from receipt where qtyonhand = 0 and qtyinkits = 0 and receipt_id not in (select distinct receipt_id from ordpickhst where receipt_id is not null) and receipt_id not in (select distinct receipt_id from ordshipst where receipt_id is not null)
attachment	obsolete	delete from attachment where days(current timestamp) - days(timeupdated) >=? and attachment_id not in (select attachment_id from trdattach)
auction	retracted	delete from auction where austatus = 'R' and (days(CURRENT TIMESTAMP) - days(updatetime)) >= ?
auction	settlement_closed	delete from auction where austatus = 'SC' and (days(CURRENT TIMESTAMP) - days(updatetime)) >= ?
auctionlog	obsolete	delete from auctionlog where (days(CURRENT TIMESTAMP) - days(actiontime)) >= ?
autobidlog	obsolete	delete from autobidlog where (days(CURRENT TIMESTAMP) - days(actiontime)) >= ?

物件	類型	陳述式
baseitem	obsolete	delete from baseitem where markfordelete = 1 and baseitem_id not in (select baseitem_id from catentry) and baseitem_id not in (select distinct baseitem_id from itemspc where markfordelete = 0 and itemspc_id in (select distinct itemspc_id from orderitems) or itemspc_id in (select distinct itemspc_id from oicomplst) or itemspc_id in (select distinct itemspc_id from versionspc where versionspc_id in (select distinct versionspc_id from receipt)) or itemspc_id in (select distinct itemspc_id from radetail) or itemspc_id in (select distinct itemspc_id from bkordalloc) or itemspc_id in (select distinct itemspc_id from invreserve) or itemspc_id in (select distinct itemspc_id from rmaitem) or itemspc_id in (select distinct itemspc_id from rmaitemcmp) or itemspc_id in (select distinct itemspc_id from catentry))
bidlog	obsolete	delete from bidlog where (days(CURRENT TIMESTAMP) - days(actiontime)) >= ?
cachlog	obsolete	delete from cachlog where (days(CURRENT TIMESTAMP) - days(cacstmp)) >= ?

物件	類型	陳述式
calculation_code	obsolete	delete from calcode where published = 2 and calcode_id not in (select distinct calcode_id from ordadjust where calcode_id is not null) and calcode_id not in (select distinct calcode_id from stencalug where calcode_id is not null) and calcode_id not in (select distinct calcode_id from ordcalcd where calcode_id is not null) and calcode_id not in (select distinct calcode_id from ordicalcd where calcode_id is not null)
catalog_group	obsolete	delete from catgroup where markfordelete = 1
catentry	without_orderitems	delete from catentry where markfordelete = 1 and(days(CURRENT_TIMESTAMP) - days(lastupdate)) >= ? and catentry_id not in (select distinct catentry_id from auction) and catentry_id not in (select distinct catentry_id from orderitems where catentry_id is not null) and catentry_id not in (select distinct catentry_id from oicomplis where catentry_id is not null) and catentry_id not in (select distinct catentry_id from rmaitem where catentry_id is not null) and catentry_id not in (select distinct catentry_id from offer where offer_id in (select distinct offer_id from orderitems))

物件	類型	陳述式
catentry	without_orderitems-iitems	<pre> delete from catentry where markfordelete = 1 and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and catentry_id not in (select distinct catentry_id from auction) and catentry_id not in (select distinct catentry_id from orderitems) and catentry_id not in (select distinct catentry_id from oicomplst where catentry_id is not null)and catentry_id not in (select distinct catentry_id from iitem) and catentry_id not in (select distinct catentry_id from rmaitem where catentry_id is not null) and catentry_id not in (select distinct catentry_id from offer where offer_id in (select distinct offer_id from orderitems)) </pre>

物件	類型	陳述式
contract	obsolete	<p>1.delete from trading where markfordelete = 1 and trdtype_id = 1 and trading_id not in (select distinct trading_id from orderitems) and trading_id not in (select distinct trading_id from rma) and trading_id not in (select distinct trading_id from ordpaymthd) and trading_id not in (select distinct account_id from ordpaymthd)</p> <p>2.delete from productset where markfordelete = 1 and productset_id not in (select distinct productset_id from tradeposcn where tradeposcn_id in (select distinct tradeposcn_id from offer where offer_id in (select distinct offer_id from orderitems)))</p> <p>3.delete from tradeposcn where markfordelete = 1 and tradeposcn_id not in (select distinct tradeposcn_id from offer where offer_id in (select distinct offer_id from orderitems))</p>
coupon_promotion	expired	delete from cppmn where days(current timestamp) - days(enddate) >=?
cpgnlog	obsolete	delete from cpgnlog
cpgnstats	obsolete	delete from cpgnstats
expected_inventory_records	obsolete	delete from ra where markfordelete = 1 and ra_id not in (select distinct ra_id from receipt, radetail where receipt.radetail_id = radetail.radetail_id)
expected_inventory_record_details	obsolete	delete from radetail where markfordelete = 1 and radetail_id not in (select distinct radetail_id from receipt)

物件	類型	陳述式
forummsg	obsolete	delete from forummsg where msgstatus = 'D' or (days(CURRENT TIMESTAMP) - days(posttime)) >= ?
fulfillment_center	obsolete	delete from ffmcenter where markfordelete = 1 and ffmcenter_id not in (select distinct ffmcenter_id from radetail) and ffmcenter_id not in (select distinct ffmcenter_id from inventory) and ffmcenter_id not in (select distinct ffmcenter_id from rma) and ffmcenter_id not in (select distinct ffmcenter_id from orderitems) and ffmcenter_id not in (select distinct allocffmc_id from orderitems) and ffmcenter_id not in (select distinct ffmcenter_id from store) and ffmcenter_id not in (select distinct rtnffmctr_id from store) and ffmcenter_id not in (select distinct ffmcenter_id from receipt) and ffmcenter_id not in (select distinct ffmcenter_id from auction) and ffmcenter_id not in (select distinct ffmcenter_id from auctionlog)
inventory_adjustments	obsolete	delete from invadjust where days(CURRENT TIMESTAMP) - days(adjustmentdate) >= ?
inventory_adjustment_codes	obsolete	delete from invadjcode where markfordelete = 1 and invadjcode_id not in (select distinct invadjcode_id from invadjust)

物件	類型	陳述式
itemspecification	obsolete	delete from itemspc where markfordelete = 1 and itemspc_id not in (select distinct itemspc_id from orderitems) and itemspc_id not in (select distinct itemspc_id from oicomplst) and itemspc_id not in (select distinct itemspc_id from versionspc where versionspc_id in (select distinct versionspc_id from receipt)) and itemspc_id not in (select distinct itemspc_id from radetail) and itemspc_id not in (select distinct itemspc_id from bkordalloc) and itemspc_id not in (select distinct itemspc_id from invreserve) and itemspc_id not in (select distinct itemspc_id from rmaitem) and itemspc_id not in (select distinct itemspc_id from rmaitemcmp) and itemspc_id not in (select distinct itemspc_id from catentry)
message	obsolete	delete from message where message_id not in (select message_id from msgmemrel) or (days(CURRENT TIMESTAMP) - days(posttime)) >= ?
msgmemrel	obsolete	delete from msgmemrel where message_id in (select m.message_id from message ms, msgmemrel m where ms.message_id = m.message_id and (status = 'D' or ((status = 'O' or sendstat = 'S') and (days(CURRENT TIMESTAMP) - days(posttime)) >= ?)))
order	canceled	delete from orders where status = 'X' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?

物件	類型	陳述式
order	completed	delete from orders where status = 'C' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?
order	deposited	delete from orders where status='D' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?
order	shipped	delete from orders where status='S' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?
order	stale_guest	delete from orders where (status='P' or status='I' or status='W' or status='N') and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and orders.member_id in (select distinct users_id from users where registertype='G') and (orders_id not in (select distinct orders_id from orderitems where inventorystatus != 'NALC' and inventorystatus is not null))
order	stale_non_guest	delete from orders where (status='P' or status='I' or status='W' or status='N') and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and orders.member_id in (select distinct users_id from users where registertype != 'G') and (orders_id not in (select distinct orders_id from orderitems where inventorystatus != 'NALC' and inventorystatus is not null))
organization	obsolete	delete from member where member_id in (select orgentity_id from orgentity where orgentity_id = ?)
pastats	obsolete	delete from pastats
pcstats	obsolete	delete from pcstats

物件	類型	陳述式
pestats	obsolete	delete from pestats
policy	obsolete	delete from policy where days(current timestamp) - days(endtime) > ? and policy_id not in (select distinct policy_id from ordpaymthd) and policy_id not in (select distinct policy_id from rma)
rfq	obsolete	delete from trading where markfordelete = 1 and trdtype_id in (2, 3, 4) and trading_id not in (select distinct trading_id from orderitems) and trading_id not in (select distinct trading_id from rma) and trading_id not in (select distinct trading_id from ordpaymthd) and trading_id not in (select distinct account_id from ordpaymthd)
rma	abandoned	delete from rma where status in ('PRC', 'EDT') and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and rma_id not in (select rma_id from rmaitem where rmaitem.status in ('APP', 'MAN')) and rma_id not in (select rma_id from rtnreceipt)
rma	approved_or_partly_ approved	delete from rma where status = 'PND' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and rma_id in (select rma_id from rmaitem where rmaitem.status in ('APP', 'MAN')) and rma_id not in (select rma_id from rtnreceipt)
rma	canceled	delete from rma where status = 'CAN' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ?

物件	類型	陳述式
rma	completed	delete from rma where status = 'CLO' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and rma_id not in (select rma_id from rtnreceipt)
rma	not_approved	delete from rma where status = 'PND' and (days(CURRENT TIMESTAMP) - days(lastupdate)) >= ? and rma_id not in (select rma_id from rmaitem where rmaitem.status in ('APP', 'MAN')) and rma_id not in (select rma_id from rtnreceipt)
rtnreasons	obsolete	delete from rtnreason where markfordelete = 1 and rtnreason_id not in (select distinct rtnreason_id from rtnrcptdsp) and rtnreason_id not in (select distinct rtnreason_id from rmaitem)
sastats	obsolete	delete from sastats
staglog	obsolete	delete from staglog where stgprocessed = 1 and (days(CURRENT TIMESTAMP) - days(stgstmp)) >= ?
store	obsolete	delete from storeent where storeent_id = ? and type='S'

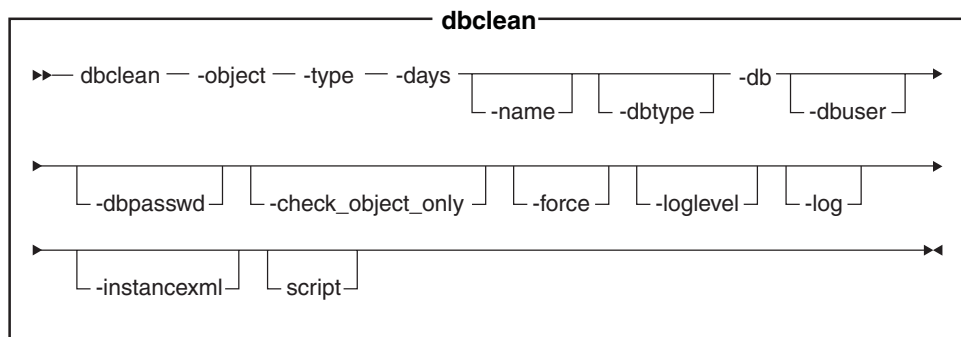
物件	類型	陳述式
users	guest	delete from member where member_id in (select users_id from users where registertype='G' and (days(CURRENT TIMESTAMP) - days(lastsession)) >= ? And (users_id not in (select member_id from orders where status != 'Q')) and (users_id > 0) and (users_id not in (select member_id from address where address_id in (select address_id from orderitems where address_id is not null and status != 'Q') or address_id in (select allocaddress_id from orderitems where allocaddress_id is not null and status != 'Q') or address_id in (select address_id from orders where address_id is not null and status != 'Q')))))
users	registered	delete from member where member_id in (select users_id from users where registertype= 'R' and (days(CURRENT TIMESTAMP) - days(lastsession)) >= ? and (users_id not in (select member_id from orders where status != 'Q')) and (users_id > 0) and (users_id not in (select member_id from address where address_id in (select address_id from orderitems where address_id is not null and status != 'Q') or address_id in (select allocaddress_id from orderitems where allocaddress_id is not null and status != 'Q') or address_id in (select address_id from orders where address_id is not null and status != 'Q')))))

物件	類型	陳述式
usrtraffic	obsolete	delete from usrtraffic where (days(CURRENT_TIMESTAMP) - days(stmp)) >= ?
vendor	obsolete	delete from vendor where markfordelete = 1 and vendor_id not in (select distinct vendor_id from ra) and vendor_id not in (select distinct vendor_id from receipt where vendor_id is not null)
productset	obsolete	delete from productset where markfordelete = 1 and productset_id not in (select productset_id from tradeposcn where productset_id is not null)
productset	obsolete	delete from productset where productset_id in (select productset_id from tradeposcn where productset_id is not NULL and markfordelete = 1 and type = 'C')
tradeposcn	obsolete	delete from tradeposcn where markfordelete = 1 and type = 'S'

資料庫清理指令

資料庫清理公用程式 (iSeries、AIX、Solaris、Linux、Windows NT 以及 Windows 2000)

「資料庫清理公用程式」會從您的資料庫中移除物件。如果要執行「資料庫清理公用程式」，請從指令行鍵入下列指令。請將整個指令鍵入在一行上。



▶ 400 iSeries 現在改為使用 dbclean.sh。如要在 iSeries 上執行 shell script，請執行下列步驟：

1. 以 CCSID 為 65535 以外的使用者設定檔登入。
2. 在 OS/400 指令行上鍵入指令：STRQSH，開啓一個 QSHELL 指令視窗。
3. 依照下列方式，執行指令：/QIBM/ProdData/CommerceServer55/bin/dbclean.sh (參數 . . .)

註：必須要指定參數。

Oracle 附註：

- 即使您目前正以相同的使用者 ID 執行這支公用程式，您也必須在指令中包括選用參數、登入使用者 ID 和密碼。
- **Linux** Oracle 參照不適用於 Linux 平台。

參數值：

物件 (object)

要刪除其已作廢記錄的物件名稱。請鍵入下列物件名稱之一：

- account 刪除帳戶物件。
- address 刪除地址物件。
- atp_inventory 刪除收據資訊物件。
- attachment 刪除附件物件。
- auction 刪除拍賣物件。
- auctionlog 刪除拍賣日誌物件。
- autobidlog 刪除自動出價物件。
- baseitem 刪除產品資訊物件。
- bidlog 刪除出價日誌物件。
- cachlog 刪除已快取物件。
- calculation_code 刪除計算代碼物件。
- catentry 刪除型錄項目物件。
- catalog_group 刪除型錄群組物件。
- contract 刪除合約物件。
- coupon_promotion 刪除優待券物件。
- cpnlog 刪除活動物件。
- cpnstats 刪除活動統計物件。
- expected_inventory_records 刪除庫存物件。
- expected_inventory_records_details 刪除庫存明細物件。
- forummsg 刪除「網站管理者」和客戶之間的訊息物件。
- fulfillment_center 刪除供貨中心物件。
- inventory_adjustments 刪除庫存物件。
- inventory_adjustment_codes 刪除庫存代碼物件。
- itemspecification 刪除指定的項目物件。
- jdbccustomizer 指定自訂參數的位置，和 dbclean 配合使用。如要配置 dbclean，使其將工具箱 JDBC 驅動程式和 iSeries 配合使用，您必須指定 DB2/iSeries 作為 dbtype，並且您的自訂單檔案應該要包含：

`jdbcDriver=com.ibm.as400.access.AS400JDBCdriver`
`jdbcUrlPrefix=jdbc:as400://`。iSeries 上的 DB2 預設值為：
`jdbcDriver=com.ibm.db2.jdbc.app.DB2Driver` `jdbcUrlPrefix=jdbc:db2://`。
其他平台上的 DB2 預設值為：`jdbcDriver=COM.ibm.db2.jdbc.app.DB2Driver`
`jdbcUrlPrefix=jdbc:db2`。

註：其他組合也可能可行，但是未做測試。

- `message` 刪除與拍賣有關的訊息物件。
- `msgmemrel` 刪除訊息成員關係物件。
- `orders` 刪除訂單物件。
- `organization` 刪除組織物件。
- `pastats` 刪除「產品顧問」統計物件。
- `pcstats` 刪除「產品對照」統計物件。
- `pestats` 刪除「產品探索」統計物件。
- `policy` 刪除原則物件。
- `product_sets` 刪除產品配套物件。
- `rfq` 刪除報價要求物件。
- `rma` 刪除退貨項目物件。
- `rtnreasons` 刪除退貨原因物件。
- `sastats` 刪除「銷售輔助」統計物件。
- `staglog` 刪除暫置物件。
- `store` 刪除商店物件。
- `user` 刪除使用者物件。
- `usrtraffic` 刪除使用者流量日誌物件。
- `vendor` 刪除供應商物件。

type 您要刪除的物件類型。請參閱清理資料庫下的個別指令。

days 要刪除記錄的最小存在天數。

name (選用的) 要刪除的物件 ID。如果 `organization` 參數的值為 `member`，並且 `type` 的值為 `organization`，則此為必要參數。

dbtype

(選用) 資料庫類型為 DB2、DB2/iSeries 或 Oracle。預設值是 DB2。預設值為 DB2/iSeries。

db 資料庫的名稱。  如在關聯式資料庫目錄中出現的資料庫名稱。

註：如果您配置 `jdbccustomizer` 參數使用工具箱 JDBC 驅動程式，則您必須指定資料庫所在的機器主機名稱，而不是資料庫名稱，例如，`-db hostname.ibm.com`。

 Oracle

使用 `host:port:sid`。比方說，`myhost:1521:mydb`。

dbuser

(選用) 建立資料庫綱目的管理者或網站管理者的登入 ID。如果未指定此參

數，則會使用呼叫公用程式的使用者 ID。▶ 400 (必要的) 與 WebSphere Commerce 實例相關的使用者設定檔。這和綱目名稱相同。

dbpasswd

(選用) `dbuser` 參數為 DB2 資料庫指定的登入 ID 的密碼。如果未指定，系統會提示您輸入密碼。

check_object_only

(選用) 「資料庫清理公用程式」會列出會受資料庫清理影響的所有下層表格，如果您指定 `yes`，則會刪除限制。如果將參數保留為 `no` (預設值)，此公用程式不會執行檢查。

force (選用) `force` 選項可以設定為 `yes` 或 `no`。如果指定 `yes`，公用程式會先刪除下層表格，然後再刪除母項表格。

loglevel

(選用) 做資料庫清理時，要執行的日誌記載層次。如果沒有指定值，則會使用預設日誌記載層次 0。

- 鍵入 0 指定不要記錄日誌活動。此為預設值。
- 鍵入 1 指定只針對指定的表格記錄 DELETE 陳述式。
- 鍵入 2 指定針對指定的表格以及任何下層表格記錄 DELETE 陳述式。記載層次 2 會強制「資料庫清理公用程式」使用由下往上方法。

log

(選用) 公用程式記錄其活動所在的日誌檔案路徑和名稱。發出此指令的使用者必須對指定的路徑擁有寫入權限，並且該路徑必須已存在。如果未指定這個參數，將在下列 `log` 目錄中，建立名稱為 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 的日誌檔案。

▶ AIX ▶ 400 `WAS_userdir/logs`

▶ Solaris ▶ Linux `WAS_installdir/logs`

▶ NT ▶ 2000 `WC_installdir/logs`

▶ 400 如果未指定這個參數，將在 `WC_userdir/instances/log` 目錄中，建立名稱為 `dbclean_dbuser_yyyy.mm.dd_hh.mm.ss.zzz.log` 的日誌檔案。

instancexml

刪除商店時使用。`instancexml` 會以

`WC_installdir/instances/instancexml/xml/instancexml.xml` 格式，取得絕對檔案名稱的值。

註: 擁有檔案資產的某些物件，如使用者、組織、商店以及連接裝置等，需要這個參數。

script

(選用) 指定要儲存的輸出 `script` 路徑和名稱。在預設的情況下，輸出會儲存








為 ▶ Windows `WC_installdir/logs\assetclean_userName_timestamp.cmd`


▶ 400 `/WC_userdir/instances/assetclean_userName_timestamp.sh`

其他平台為 `WC_installdir/logs/assetclean_userName_timestamp.sh`。

清理資料庫

如要從 WebSphere Commerce 資料庫中刪除未使用或已作廢物件，請使用「資料庫清理公用程式」，並選取下列其中一項：

- 帳戶物件
- 地址物件
- 約定交付 (atp) 庫存物件
- 附件物件
-  拍賣物件
-  拍賣日誌物件
-  自動出價日誌物件
- 基本項目物件
-  出價日誌物件
- 已快取物件
- 計算代碼物件
- 活動物件
- 活動統計物件
- 型錄項目物件
- 型錄群組物件
- 合約物件
- 優待券物件
- 預期的庫存記錄物件
- 預期的庫存記錄明細物件
- 討論區訊息物件
- 供貨中心物件
- 庫存調整代碼物件
- 庫存調整物件
- 項目規格物件
- 成員訊息物件
- 訊息物件
- 訂單物件
- 組織物件
-  「產品顧問」統計物件
-  「產品對照」統計物件
-  「產品探索」統計物件
- 原則物件
- 產品配套物件
- 報價要求 (RFQ) 物件






- 退貨項目物件
- 退貨原因物件
-  「銷售輔助」統計物件
- 暫置物件
- 商店物件
- 使用者物件
- 使用者流量日誌物件
- 供應商物件

附註： 在使用「資料庫清理公用程式」之前，您必須先配置資料庫並設定 PATH 環境變數。

設定 WebSphere Commerce 公用程式的 PATH 環境變數

註： 本節不適用於 iSeries。

在使用「資料庫清理公用程式」或暫置作業伺服器之前，請確定您已經在 PATH 環境變數中設定好下列變數：

-  
`WC_installdir/bin`
 以及 `WC_installdir\sqllib\bin`。
-  
`drive:\WebSphere\CommerceServer\bin` 以及
`drive:\WebSphere\CommerceServer\ora9i\bin`
-  
`WC_installdir/bin` 以及 `WC_installdir\sqllib\bin`
-  
`WC_installdir/bin` 以及 `WC_installdir\ora9i\bin`
-  
`WC_installdir/bin` 和 `WC_installdir/lpp/db2_08_01`
-  
`WC_installdir/bin` 和 `/usr/ora9i/bin`
-  
`WC_installdir/bin` 和 `WC_installdir/IBMdb2/V8.1`
-  
`WC_installdir/bin` 和 `opt/IBM/db2/V8.1`
-  
`WC_installdir/bin` 和 `WC_installdir/ora9i/bin`
-  這個步驟不適用於 OS/400 for iSeries。














刪除物件

本節包含使用「資料庫清理公用程式」，從 WebSphere Commerce 資料庫中刪除未使用或已作廢的物件。

刪除帳戶物件

如果要刪除帳戶物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object account -type obsolete -db dbname -days daysold -loglevel loglevel`
-  
`dbclean -object account -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object account -type obsolete -db dbname -days daysold -loglevel loglevel`
-   
`. dbclean.sh -object account -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('account') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。









2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。






如需有關帳戶物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除地址物件

如果要刪除地址物件，請執行下列步驟：

1. 輸入下列指令：














-  
`dbclean -object address -type obsolete -db dbname -days daysold -loglevel loglevel`
-  
`dbclean -object address -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object address -type obsolete -db dbname -days daysold -loglevel loglevel`

-   
`. dbclean.sh -object address -type obsolete -db dbname -days daysold
 -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
 -  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
 OBJECT('address') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)`
 使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。
2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。
- 如需有關刪除地址物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除約定交付庫存物件

如果要刪除約定交付 (ATP) 庫存物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object atp_inventory -type typename -db dbname -loglevel loglevel`
-  
`dbclean -object atp_inventory -type typename -db dbname -loglevel loglevel
 -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object atp_inventory -type typename -db dbname -loglevel
loglevel`
-   
`. dbclean.sh -object atp_inventory -type typename -db dbname -loglevel
loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
 OBJECT('atp_inventory') TYPE(typename) LOGLEVEL(loglevel)
 STATUS(orderstatus)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。



2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。












如需有關刪除 ATP 庫存物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除附件物件

如果要刪除附件物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object attachment -type obsolete -db dbname -days daysold
 -loglevel loglevel`

-  
`dbclean -object attachment -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object attachment -type obsolete -db dbname -days daysold -loglevel loglevel`
-   
`. dbclean.sh -object attachment -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('attachment') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)`
 使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

註：連接裝置的 `-type` 也可以是 `markfordelete`。在本範例中，應該是 `-instancexml`。例如：

```
dbclean.sh -object attachment -type markfordelete -instancexml
/WC_userdir/instances/instance/xml/instance.xml
-db dbname -dbuser instance -loglevel
```














2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除連接裝置物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除拍賣物件

如果要刪除拍賣物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object auction -type typename -db dbname -days daysold -loglevel loglevel`
-  
`dbclean -object auction -type typename -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object auction -type typename -db dbname -days daysold -loglevel loglevel`
-   
`. dbclean.sh -object auction -type typename -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('auction') TYPE(typename) LOGLEVEL(loglevel) DAYS(daysold)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。附註：針對 `-type` 參數，您可以指定 `settlement_closed`，指出已完成的拍賣記錄，或 `retracted` 指出已撤回的拍賣記錄。

2. 檢查 `dbclean.log_yyyy.mm.dd_hh.mm.ss.zzz` 檔，驗證指令是否順利完成。

如需有關刪除拍賣物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除拍賣日誌物件

如果要刪除拍賣日誌物件，請執行下列步驟：

1. 輸入下列指令：

- Windows DB2

```
dbclean -object auctionlog -type obsolete -db dbname -days daysold  
-loglevel loglevel
```

- Windows Oracle

```
dbclean -object auctionlog -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

•

- AIX Solaris Linux DB2

```
. dbclean.sh -object auctionlog -type obsolete -db dbname -days daysold  
-loglevel loglevel
```

- AIX Solaris Linux

```
. dbclean.sh -object auctionlog -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

- 400 DB2

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)  
OBJECT('auctionlog') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除拍賣日誌物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除自動出價日誌物件

如果要刪除自動出價日誌物件，請執行下列步驟：

1. 輸入下列指令：

- Windows DB2

```
dbclean -object autobidlog -type obsolete -db dbname -days daysold  
-loglevel loglevel
```

- Windows Oracle

```
dbclean -object autobidlog -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

- ▶ AIX
 - ▶ Solaris
 - ▶ Linux
 - ▶ DB2

```
. dbclean.sh -object autobidlog -type obsolete -db dbname -days daysold
-loglevel loglevel
```
- ▶ AIX
 - ▶ Solaris
 - ▶ Oracle

```
. dbclean.sh -object autobidlog -type obsolete -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```
- ▶ 400
 - ▶ DB2

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('autobidlog') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)
```

使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。

- 檢查 dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log 檔，以驗證指令是否成功。

如需有關刪除自動出價日誌物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除基本項目物件

如果要刪除基本項目物件，其中包含有關具有相同名稱與說明的一般商品系列資訊，請執行下列步驟：

- 輸入下列指令：

- ▶ Windows
 - ▶ DB2

```
dbclean -object baseitem -type obsolete -db dbname -loglevel loglevel
```
- ▶ Windows
 - ▶ Oracle

```
dbclean -object baseitem -type obsolete -db dbname -loglevel loglevel
-dbtype oracle -dbuser user -dbpasswd password
```
- ▶ AIX
 - ▶ Solaris
 - ▶ Linux
 - ▶ DB2

```
. dbclean.sh -object baseitem -type obsolete -db dbname -loglevel loglevel
```
- ▶ AIX
 - ▶ Solaris
 - ▶ Oracle

```
. dbclean.sh -object baseitem -type obsolete -db dbname -loglevel loglevel
-dbtype oracle -dbuser user -dbpasswd password
```
- ▶ 400
 - ▶ DB2

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('baseitem') TYPE('obsolete') LOGLEVEL(loglevel)
```

使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。

- 檢查 dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log 檔，以驗證指令是否成功。

如需有關刪除基本項目物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除出價日誌物件

如果要刪除出價日誌物件，請執行下列步驟：

- 輸入下列指令：

•  

```
dbclean -object bidlog -type obsolete -db dbname -days daysold -loglevel loglevel
```

•  

```
dbclean -object bidlog -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

•    

```
. dbclean.sh -object bidlog -type obsolete -db dbname -days daysold -loglevel loglevel
```

•   

```
. dbclean.sh -object bidlog -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

•  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('bidlog') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除出價日誌物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除 `cachlog` 物件

如果要刪除 `cachlog` 物件，請執行下列步驟：

1. 輸入下列指令：

•  

```
dbclean -object cachlog -type obsolete -db dbname -days daysold -loglevel loglevel
```

•  

```
dbclean -object cachlog -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

•    

```
. dbclean.sh -object cachlog -type obsolete -db dbname -days daysold -loglevel loglevel
```

•   

```
. dbclean.sh -object cachlog -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

•  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('cachlog') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。














2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除已快取物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除計算代碼物件

如果要刪除計算代碼物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object calculation_code -type obsolete -db dbname -loglevel loglevel`
-  
`dbclean -object calculation_code -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object calculation_code -type obsolete -db dbname -loglevel loglevel`
-   
`. dbclean.sh -object calculation_code -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('calculation_code') TYPE('obsolete') LOGLEVEL(loglevel)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。














2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關計算代碼物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除活動物件

如果要刪除活動物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object cpnlog -type all -db dbname -loglevel loglevel`
-  
`dbclean -object cpnlog -type all -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object cpnlog -type all -db dbname -loglevel loglevel`
-   
`. dbclean.sh -object cpnlog -type all -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('cpnlog') TYPE('all') LOGLEVEL(loglevel)`

- 使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。
- 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。
- 如需有關刪除活動物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除活動統計物件

如果要刪除活動統計物件，請執行下列步驟：

- 輸入下列指令：

- Windows DB2

```
dbclean -table cpgnstats -type all -db dbname -loglevel loglevel
```

- Windows Oracle

```
dbclean -table cpgnstats -type all -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

- AIX Solaris Linux DB2

```
. dbclean.sh -object cpgnstats -type all -db dbname -loglevel loglevel
```

- AIX Solaris Oracle

```
. dbclean.sh -object cpgnstats -type all -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

- 400 DB2

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('cpgnstats') TYPE('all') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

- 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關從 `CPGNSTATS` 表格刪除活動統計記錄的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除型錄項目物件

如果要刪除型錄項目物件，請執行下列步驟：

- 輸入下列指令：

- Windows DB2




```
dbclean -object catentry -type typename -db dbname -days daysold -loglevel loglevel
```

- Windows Oracle



```
dbclean -object catentry -type typename -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

- AIX Solaris Linux DB2

```
. dbclean.sh -object catentry -type typename -db dbname -days daysold -loglevel loglevel
```

-   

```
. dbclean.sh -object catentry -type typename -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('catentry') TYPE(typename) LOGLEVEL(loglevel) DAYS(daysold)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。附註：針對 `-type` 參數，您可以指定 `without_orderitems`，指出沒有參照任何訂單項目的型錄項目，或指定 `without_orderitems_iitems`，指出沒有參照任何訂單項目或喜好項目清單的型錄項目。



2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除型錄項目物件的其他範例，請參閱第 135 頁的『刪除物件範例』。



刪除型錄群組物件

如果要刪除型錄群組物件，請執行下列步驟：





1. 輸入下列指令：

-  




```
dbclean -object catalog_group -type obsolete -db dbname -loglevel loglevel
```

-  



```
dbclean -object catalog_group -type obsolete -db dbname -loglevel loglevel
-dbtype oracle -dbuser user -dbpasswd password
```

-    

```
. dbclean.sh -object catalog_group -type obsolete -db dbname -loglevel
loglevel
```

-   

```
. dbclean.sh -object catalog_group -type obsolete -db dbname -loglevel
loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('catalog_group') TYPE('obsolete') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除型錄群組物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除合約物件

如果要刪除合約物件，請執行下列步驟：

1. 輸入下列指令：

-  

```
dbclean -object contract -type obsolete -db dbname -loglevel loglevel
```


-  

```
dbclean -object contract -type obsolete -db database -loglevel loglevel
-dbtype oracle -dbuser user -dbpasswd password
```

-    

```
. dbclean.sh -object contract -type obsolete -db dbname -loglevel loglevel
```

-   

```
. dbclean.sh -object contract -type obsolete -db dbname -loglevel loglevel
-dbtype oracle -dbuser user -dbpasswd password
```

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('contract') TYPE('obsolete') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。附註：

針對     `-dbor`  `DATABASE` 參數，您必須指定三個資料庫表格之一：`productset`、`tradeposn` 或 `trading`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除合約物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除優待券物件

如果要刪除優待券物件，請執行下列步驟：

1. 輸入下列指令：

-  

```
dbclean -object coupon_promotion -type expired -db dbname -days daysold
-loglevel loglevel
```

-  

```
dbclean -object coupon_promotion -type expired -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-    

```
. dbclean.sh -object coupon_promotion -type expired -db dbname -days
daysold -loglevel loglevel
```

-   

```
. dbclean.sh -object coupon_promotion -type expired -db dbname -days
daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('coupon_promotion') TYPE('expired') LOGLEVEL(loglevel) DAYS(daysold)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。














2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除優待券物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除預期的庫存記錄物件

如果要刪除預期的庫存記錄物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object expected_inventory_records -type obsolete -db dbname -days daysold -loglevel loglevel`
-  
`dbclean -object expected_inventory_records -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object expected_inventory_records -type obsolete -db dbname -days daysold -loglevel loglevel`
-   
`. dbclean.sh -object expected_inventory_records -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('expected_inventory_records') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。












2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除預期庫存記錄物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除有關預期的庫存記錄明細物件

如果要刪除有關預期的庫存記錄明細物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object expected_inventory_record_details -type obsolete -db dbname -loglevel loglevel`
-  
`dbclean -object expected_inventory_record_details -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object expected_inventory_record_details -type obsolete -db dbname -loglevel loglevel`
-   
`. dbclean.sh -object expected_inventory_record_details -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('expected_inventory_record_details') TYPE('obsolete')
LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除預期庫存記錄明細物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除討論區訊息物件

如果要刪除討論區訊息物件，請執行下列步驟：

1. 輸入下列指令：

-  

```
dbclean -object forummsg -type obsolete -db dbname -days daysold -loglevel
loglevel
```

-  

```
dbclean -object forummsg -type obsolete -db dbname -days daysold -loglevel
loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-    

```
. dbclean.sh -object forummsg -type obsolete -db dbname -days daysold
-loglevel loglevel
```

-   

```
. dbclean.sh -object forummsg -type obsolete -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('forummsg') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除討論區訊息物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除供貨中心物件

如果要刪除供貨中心物件，請執行下列步驟：

1. 輸入下列指令：

-  

```
dbclean -object fulfillment_center -type obsolete -db dbname -loglevel
loglevel
```

-  

```
dbclean -object fulfillment_center -type obsolete -db dbname -loglevel
loglevel -dbtype oracle -dbuser user -dbpasswd password
```

- AIX
 - Solaris
 - Linux
 - DB2

```
. dbclean.sh -object fulfillment_center -type obsolete -db dbname -loglevel loglevel
```
- AIX
 - Solaris
 - Oracle

```
. dbclean.sh -object fulfillment_center -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```
- 400
 - DB2

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('fulfillment_center') TYPE('obsolete') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

- 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除供貨中心物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除庫存代碼調整物件

如果要刪除庫存代碼調整物件，請執行下列步驟：

- 輸入下列指令：

- Windows
 - DB2

```
dbclean -object inventory_adjustment_codes -type obsolete -db dbname -loglevel loglevel
```
- Windows
 - Oracle

```
dbclean -object inventory_adjustment_codes -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```
- AIX
 - Solaris
 - Linux
 - DB2

```
. dbclean.sh -object inventory_adjustment_codes -type obsolete -db dbname -loglevel loglevel
```
- AIX
 - Solaris
 - Oracle

```
. dbclean.sh -object inventory_adjustment_codes -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```
- 400
 - DB2

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) DBTABLE('inventory_adjustment_codes') TYPE('obsolete') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。














- 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除庫存代碼調整物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除庫存調整物件

如果要刪除庫存調整物件，請執行下列步驟：

- 輸入下列指令：

-  
`dbclean -object inventory_adjustments -type obsolete -db dbname -days daysold -loglevel loglevel`
-  
`dbclean -object inventory_adjustments -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object inventory_adjustments -type obsolete -db dbname -days daysold -loglevel loglevel`
-   
`. dbclean.sh -object inventory_adjustments -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('inventory_adjustments') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。














2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除庫存調整物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除指定的項目資訊物件

如果要刪除指定的項目資訊物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object itemspecification -type obsolete -db dbname -loglevel loglevel`
-  
`dbclean -object itemspecification -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object itemspecification -type obsolete -db dbname -loglevel loglevel`
-   
`. dbclean.sh -object itemspecification -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('itemspecification') TYPE('obsolete') LOGLEVEL(loglevel)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。














2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除指定的項目資訊物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除成員訊息關係物件

如果要刪除成員訊息關係物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object msgmemrel -type obsolete -db dbname -days daysold -loglevel loglevel`
-  
`dbclean -object msgmemrel -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object msgmemrel -type obsolete -db dbname -days daysold -loglevel loglevel`
-   
`. dbclean.sh -object msgmemrel -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('msgmemrel') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。












2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除訊息成員關係物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除訊息物件

如果要刪除訊息物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object message -type obsolete -db dbname -days daysold -loglevel loglevel`
-  
`dbclean -object message -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object message -type obsolete -db dbname -days daysold -loglevel loglevel`
-   
`. dbclean.sh -object message -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('message') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除訊息物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除訂單物件

如果要刪除訂單物件，請執行下列步驟：

1. 輸入下列指令：

-  

```
dbclean -object order -type typename -db dbname -days daysold -loglevel
loglevel
```

-  

```
dbclean -object order -type typename -db dbname -days daysold -loglevel
loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-    

```
. dbclean.sh -object order -type typename -db dbname -days daysold
-loglevel loglevel
```

-   

```
. dbclean.sh -object order -type typename -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('order') TYPE(typename) LOGLEVEL(loglevel) DAYS(daysold)
STATUS(orderstatus)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。附註：針對 `-type` 參數，您可以指定六種不同類型之一：`completed` 指出已完成訂單，`canceled` 指出已取消的訂單，`shipped` 指出已出貨的訂單，`deposited` 指出已請款的訂單，`stale_guest` 指出訪客的陳舊訂單，或 `stale_non_guest` 指出非訪客型的客戶陳舊訂單。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除訂單物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除組織物件

如果要刪除組織物件，請執行下列步驟：

1. 輸入下列指令：

-  

```
dbclean -object organization -type specified -db dbname -loglevel loglevel
```

```
- name organization - instancexml
WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

•  

```
dbclean -object organization -type specified -db dbname -loglevel loglevel
-dbtype oracle -dbuser user -dbpasswd password -name organization
-instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

•    

```
. dbclean.sh -object organization -type specified -db dbname -loglevel
loglevel - name organization - instancexml
WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

•   

```
. dbclean.sh -object organization -type specified -db dbname -loglevel
loglevel -dbtype oracle -dbuser user -dbpasswd password -name
organization - instancexml
WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

•  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('organization') TYPE('specified') LOGLEVEL(loglevel)
NAME(organization) - instancexml
WC_userdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

使用 host:port:sid 作為 Oracle 資料庫名稱。比方說，myhost:1521:mydb。附註：針對 -type 參數，您可以指定 organization，指出任何組織記錄。

2. 檢查 dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log 檔，以驗證指令是否成功。

如需有關刪除組織物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除「產品顧問」統計物件

如果要刪除「產品顧問」統計物件，請執行下列步驟：

1. 輸入下列指令：

•  

```
dbclean -object pastats -type all -db dbname -loglevel loglevel
```

•  

```
dbclean -object pastats -type all -db dbname -loglevel logLevel -dbtype
oracle -dbuser user -dbpasswd password
```

•    

```
. dbclean.sh -object pastats -type all -db dbname -loglevel loglevel
```

•   

```
. dbclean.sh -object pastats -type all -db dbname -loglevel loglevel
-dbtype oracle -dbuser user -dbpasswd password
```

•  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('pastats') TYPE('obsolete') LOGLEVEL(loglevel)
```


- 使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。
- 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除「產品顧問」統計物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除「產品對照」統計物件

如果要刪除「產品對照」（產品顧問）統計物件，請執行下列步驟：

- 輸入下列指令：

•  

```
dbclean -object pcstats -type all -db dbname -loglevel loglevel
```

•  

```
dbclean -object pcstats -type all -db dbname -loglevel loglevel -dbtype  
oracle -dbuser user -dbpasswd password
```

•    

```
. dbclean.sh -object pcstats -type all -db dbname -loglevel loglevel
```

•   

```
. dbclean.sh -object pcstats -type all -db dbname -loglevel loglevel  
-dbtype oracle -dbuser user -dbpasswd password
```

•  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)  
OBJECT('pcstats') TYPE('all') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

- 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除「產品對照」（「產品顧問」）統計物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除「產品探索」統計物件

如果要刪除「產品探索」（產品顧問）統計物件，請執行下列步驟：

- 輸入下列指令：

•  


```
dbclean -object pestats -type all -db dbname -loglevel loglevel
```

•  

```
dbclean -object pestats -type all -db dbname -loglevel loglevel -dbtype  
oracle -dbuser user -dbpasswd password
```

•    

```
. dbclean.sh -object pestats -type all -db dbname -loglevel loglevel
```

•   

```
. dbclean.sh -object pestats -type all -db dbname -loglevel loglevel  
-dbtype oracle -dbuser user -dbpasswd password
```

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('pestats') TYPE('all') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除「產品探索」（「產品顧問」）統計物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除原則物件

如果要刪除原則物件，請執行下列步驟：

1. 輸入下列指令：

-  


```
dbclean -object policy -type obsolete -db dbname -days daysold -loglevel
loglevel
```

-  

```
dbclean -object policy -type obsolete -db dbname -days daysold -loglevel
loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-    

```
. dbclean.sh -object policy -type obsolete -db dbname -days daysold
-loglevel loglevel
```

-   

```
. dbclean.sh -object policy -type obsolete -db dbname -days daysold
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('policy') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除原則物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除產品配套物件

如果要刪除產品配套物件，請執行下列步驟：

1. 輸入下列指令：

-  

```
dbclean -object product_sets -type obsolete -db dbname -loglevel loglevel
```

-  

```
dbclean -object product_sets -type obsolete -db dbname -loglevel loglevel
-dbtype oracle -dbuser user -dbpasswd password
```

- ▶ AIX
 - ▶ Solaris
 - ▶ Linux
 - ▶ DB2

```
. dbclean.sh -object product_sets -type obsolete -db dbname -loglevel loglevel
```
- ▶ AIX
 - ▶ Solaris
 - ▶ Oracle

```
. dbclean.sh -object product_sets -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```
- ▶ 400
 - ▶ DB2

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('product_sets') TYPE('obsolete') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

- 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除產品配套物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除報價要求物件

如果要刪除報價要求 (RFQ) 物件，請執行下列步驟：

- 輸入下列指令：

- ▶ Windows
 - ▶ DB2

```
dbclean -table rfq -type obsolete -db dbname -loglevel loglevel
```
- ▶ Windows
 - ▶ Oracle

```
dbclean -object rfq -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```
- ▶ AIX
 - ▶ Solaris
 - ▶ Linux
 - ▶ DB2

```
. dbclean.sh -object rfq -type obsolete -db dbname -loglevel loglevel
```
- ▶ AIX
 - ▶ Solaris
 - ▶ Oracle

```
. dbclean.sh -object rfq -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```
- ▶ 400
 - ▶ DB2

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('rfq') TYPE('obsolete') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

- 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除 RFQ 物件的其他範例，請參閱第 135 頁的『刪除物件範例』。












刪除退貨項目物件

如果要刪除退貨項目物件，請執行下列步驟：

- 輸入下列指令：

- ▶ Windows
 - ▶ DB2

```
dbclean -object rma -type typename -db dbname -days daysold -loglevel loglevel
```

-  
`dbclean -object rma -type typename -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object rma -type typename -db dbname -days daysold -loglevel loglevel`
-   
`. dbclean.sh -object rma -type typename -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('rma') TYPE(typename) LOGLEVEL(loglevel) DAYS(daysold)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。附註：針對 `-type` 參數，您可以指定 `abandoned` 指出已放棄的記錄，`canceled` 指出已取消的記錄，`not_approved` 指出已拒絕的記錄，`approved_or_partly_approved` 指出已核准或局部核准的記錄，或者 `completed` 指出已完成的記錄。














2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除退貨項目物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除退貨原因物件

如果要刪除客戶對商品不滿意的原因，或是退貨原因物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object rtnreason -type obsolete -db dbname -loglevel loglevel`
-  
`dbclean -object rtnreason -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object rtnreason -type obsolete -db dbname -loglevel loglevel`
-   
`. dbclean.sh -object rtnreason -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('rtnreason') TYPE('obsolete') LOGLEVEL(loglevel)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除退貨原因物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除「銷售輔助」統計物件

如果要刪除「銷售輔助」（產品顧問）統計物件，請執行下列步驟：

1. 輸入下列指令：

•  




```
dbclean -object sastats -type all -db dbname -loglevel loglevel
```

•  

```
dbclean -object sastats -type all -db dbname -loglevel loglevel -dbtype  
oracle -dbuser user -dbpasswd password
```

•    

```
. dbclean.sh -object sastats -type all -db dbname -loglevel loglevel
```

•   

```
. dbclean.sh -object sastats -type all -db dbname -loglevel loglevel  
-dbtype oracle -dbuser user -dbpasswd password
```

•  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)  
OBJECT('sastats') TYPE('all') LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除「銷售輔助」（「產品顧問」）統計物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除暫置物件

如果要刪除暫置物件，請執行下列步驟：

1. 輸入下列指令：

•  

```
dbclean -object staglog -type obsolete -db dbname -days daysold -loglevel  
loglevel
```

•  

```
dbclean -object staglog -type obsolete -db dbname -days daysold -loglevel  
loglevel -dbtype oracle -dbuser <user> -dbpasswd <password>
```

•    

```
. dbclean.sh -object staglog -type obsolete -db dbname -days daysold  
-loglevel loglevel
```

•   

```
. dbclean.sh -object staglog -type obsolete -db dbname -days daysold  
-loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password
```

•  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)  
OBJECT('staglog') TYPE('obsolete') DAYS(daysold) LOGLEVEL(loglevel)
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除暫置物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除商店物件

如果要刪除商店物件，請執行下列步驟：

1. 輸入下列指令：

- Windows DB2

```
dbclean -object store -type specified -db dbname -loglevel loglevel -name storeid -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```
- Windows Oracle

```
dbclean -object store -type specified -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password -name storeid -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```
- AIX Solaris Linux DB2

```
. dbclean.sh -object store -type specified -db dbname -loglevel loglevel -name storeid -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```
- AIX Solaris Oracle

```
. dbclean.sh -object store -type specified -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password -name storeid -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```
- 400 DB2

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('store') TYPE('specified') LOGLEVEL(loglevel) NAME(storeid) -instancexml WC_userdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除商店物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除使用者物件

如果要刪除使用者物件，請執行下列步驟：










1. 輸入下列指令：

- Windows DB2

```
dbclean -object user -type typename -db dbname -days daysold -loglevel loglevel -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```
- Windows Oracle

```
dbclean -object user -type typename -db dbname -days daysold -loglevel
```

```
loglevel -dbtype oracle -dbuser user -dbpasswd password -instancexml
WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml
```

-    
• `dbclean.sh -object user -type typename -db dbname -days daysold -loglevel loglevel -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml`
-   
• `dbclean.sh -object user -type typename -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password -instancexml WC_installdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('user') TYPE(typename) LOGLEVEL(loglevel) DAYS(daysold) -instancexml WC_userdir/instances/INSTANCE_NAME/xml/INSTANCE_NAME.xml`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。附註：針對 `-type` 參數，您可以指定 `guest` 指出訪客，或指定 `registered` 指出已登錄的客戶。













2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除使用者物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除使用者流量日誌物件

如果要刪除使用者流量日誌物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object usrtraffic -type obsolete -db dbname -days daysold -loglevel loglevel`
-  
`dbclean -object usrtraffic -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
• `dbclean.sh -object usrtraffic -type obsolete -db dbname -days daysold -loglevel loglevel`
-   
• `dbclean.sh -object usrtraffic -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('usrtraffic') TYPE('obsolete') LOGLEVEL(loglevel) DAYS(daysold)`

使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。














2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除使用者流量日誌物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

刪除供應商物件

如果要刪除供應商物件，請執行下列步驟：

1. 輸入下列指令：

-  
`dbclean -object vendor -type obsolete -db dbname -loglevel loglevel`
-  
`dbclean -object vendor -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-    
`. dbclean.sh -object vendor -type obsolete -db dbname -loglevel loglevel`
-   
`. dbclean.sh -object vendor -type obsolete -db dbname -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  
`dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password) OBJECT('vendor') TYPE('obsolete') LOGLEVEL(loglevel)`








使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

2. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如需有關刪除供應商物件的其他範例，請參閱第 135 頁的『刪除物件範例』。

新增配置到「資料庫清理公用程式」

如果要新增配置到「資料庫清理公用程式」，請使用下列語法作為參考。例如，物件 `o1` 中包含表格 `R1`，`R1` 具有下列直欄：`col1`、`col2`、`lastupdate` 以及 `col3`。如果要配置「資料庫清理公用程式」，以刪除具有 `col1 > 10`，並且 `lastupdate` 為 `n` 天之前的所有物件，請執行下列步驟：

- 
- 1.      開啟一個 DB2 指令提示視窗。
- 2. 輸入下列指令：
`db2 insert into cleanconf (objectname, type, statement, namearg, sequence, daysarg) values ('o1', 'obsolete', 'delete from r1 where col1 > 10 and (days(CURRENT TIMESTAMP) - days(lastupdate)) > ?', 'no', 1, 'yes')`
- 1.  執行下列 SQL 陳述式：
`insert into cleanconf (objectname, type, statement, namearg, sequence, daysarg) values ('o1', 'obsolete', 'delete from r1 where col1 > 10 and (days(CURRENT TIMESTAMP) - days(lastupdate)) > ?', 'no', 1, 'yes')`

▶ Oracle



1. 開啓一個 SQLPlus 指令視窗。
2. 輸入下列指令：



```
insert into cleanconf (objectname, type, statement, namearg, sequence,
daysarg) values ('o1', 'obsolete', 'delete from r1 where col1 > 10 and (sysdate
- lastupdate) > ?', 'no', 1, 'yes')
```





其中的 ? 會由來自下列指令行的 `-days` 參數取代。'no' 指出陳述式中不使用名稱參數。'yes' 指出陳述式中有使用 `-days` 參數。'obsolete' 說明針對物件 o1 所做的清理類型。您可以使用其他字詞，不過，一定要和您在啓動「資料庫清理」公用程式指令時，在 `-type` 引數中所使用的字詞相同。




範例



如果要啓動「資料庫清理公用程式」指令，以清除已經存在新表格中超過 2 天的記錄，請鍵入下列指令：

-  

```
dbclean -object o1 -db dbname -type obsolete -days 2 -loglevel 1
```
-  

```
dbclean -object o1 -db dbname -type obsolete -days 2 -loglevel 1 -dbtype
oracle -dbuser user -dbpasswd password
```
-    

```
. dbclean.sh -object o1 -db dbname -type obsolete -days 2 -loglevel 1
```
-   

```
. dbclean.sh -object o1 -db dbname -type obsolete -days 2 -loglevel 1 -dbtype
oracle -dbuser user -dbpasswd password
```
-  

```
dbclean.sh DATABASE(dbname) SCHEMA(schema_name) PASSWD(instance_password)
OBJECT('o1') TYPE('obsolete') LOGLEVEL(1) DAYS(2)
```



附註：如果是 Oracle `dbname` 參數，請使用 `host:port:sid`。例如，`myhost:1521:mydb`。



刪除物件範例

下列是使用來自「資料庫清理公用程式」指令的選用參數，從資料庫表格中刪除物件的範例。如需詳細的參數資訊，請參考「資料庫清理公用程式」指令。

範例 1

如果要確認哪些表格有指定刪除限制參數，請鍵入下列指令：

-  

```
dbclean -object objectname -type typename -db database -check_object_only
yes
```
-  

```
dbclean -object objectname -type typename -db host:port:sid -check_object_only
yes -dbtype oracle -dbuser user -dbpasswd <password>
```

- ▶ AIX
 - ▶ Solaris
 - ▶ Linux
 - ▶ DB2

```
. dbclean.sh -object objectname -type typename -db database -check_object_only yes
```
- ▶ AIX
 - ▶ Solaris
 - ▶ Oracle

```
. dbclean.sh -object objectname -type typename -db host:port:sid -check_object_only yes -dbtype oracle -dbuser user -dbpasswd password
```
- ▶ 400
 - ▶ DB2

```
dbclean.sh DATABASE(dbname)
SCHEMA(schema_name) PASSWD(instance_password) OBJECT(objectname)
TYPE(typename) CHECKONLY(*YES)
```

範例 2

如果要針對有指定刪除限制參數的表格使用 `force` 選項，請鍵入下列指令：

- ▶ Windows
 - ▶ DB2

```
dbclean -object objectname -type typename -db database -days daysold -loglevel loglevel -force yes
```
- ▶ Windows
 - ▶ Oracle

```
dbclean -object objectname -type typename -db host:port:sid -days daysold -loglevel loglevel -force yes -dbtype oracle -dbuser user -dbpasswd <password>
```
- ▶ AIX
 - ▶ Solaris
 - ▶ Linux
 - ▶ DB2

```
. dbclean.sh -object objectname -type typename -db database -days daysold -loglevel loglevel -force yes
```
- ▶ AIX
 - ▶ Solaris
 - ▶ Oracle

```
. dbclean.sh -object objectname -type typename -db host:port:sid -days daysold -loglevel loglevel -force yes -dbtype oracle -dbuser user -dbpasswd <password>
```
- ▶ 400
 - ▶ DB2

```
dbclean.sh DATABASE(dbname)
SCHEMA(schema_name) PASSWD(instance_password) OBJECT(objectname)
TYPE(typename) FORCE(*YES) DAYS(daysold) LOGLEVEL(loglevel)
```

範例 3

預設日誌檔案名稱一律是 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 的變量。如果要指定日誌檔名，請鍵入下列指令：

- ▶ Windows
 - ▶ DB2

```
dbclean -object objectname -type typename -db database -days daysold -loglevel loglevel -log logfilename
```
- ▶ Windows
 - ▶ Oracle

```
dbclean -object objectname -type typename -db host:port:sid -days daysold -loglevel loglevel -log logfilename -dbtype oracle -dbuser user -dbpasswd <password>
```

- ▶ AIX
▶ Solaris
▶ Linux
▶ DB2

```
. dbclean.sh -object objectname -type typename -db database -days daysold
-loglevel loglevel -log logfile
```
- ▶ AIX
▶ Solaris
▶ Oracle

```
. dbclean.sh -object objectname -type typename -db host:port:sid -days daysold
-loglevel loglevel -log logfile -dbtype oracle -dbuser user -dbpasswd
<password>
```
- ▶ 400
▶ DB2

```
dbclean.sh                               DATABASE(dbname)
SCHEMA(schema_name) PASSWD(instance_password) OBJECT(objectname)
TYPE(typename) DAYS(daysold) LOGLEVEL(loglevel) LOG(logfile)
```

第 7 章 效能

WebSphere Commerce 是一些產品之間的複雜互動。每一項產品都有自己的效能性質，而且在各種元件的互動之內，有一些地方的效能會因為配置不正確或資源不足而受到影響。效能目標包括以及時方式處理下列類型的要求：

- 處理多項客戶申請
- 存取 WebSphere Commerce 資料庫中的資料
- 將資料製成網頁格式
- 將回應傳回至購物者的瀏覽器

如果要將 WebSphere Commerce 最佳化，請考慮下列元件：

- 硬體

請確定您的機器符合 *WebSphere Commerce* 安裝手冊指出的基本機器需求。

在正式作業環境中，將會有許多並行的使用者存在，因此多部 CPU 有助於提高效能。使用高速的 CPU 通常可以加快大部份的作業。

- 資料庫

如果是 DB2 資料庫調整，請參閱紅皮書：*DB2/UDB WebSphere Performance Tuning Guide*（可以在此網站取得：

<http://publib-b.boulder.ibm.com/Redbooks.nsf/RedbookAbstracts/sg246417.html?Open>）。

請確定最大資料庫連線儲存池大小足以應付所有並行的作業（例如，HTTP 連線、排程器執行緒等）。

- WebSphere Commerce

請確定伺服器具有 I/O 界限 - 如果有許多檔案存取或網路存取，WebSphere Commerce 系統的效能會受影響。例如，如果啓用所有的日誌記載與追蹤，則系統可能會用掉大部份的時間將資料寫入磁碟，而不是用來處理工作量。

依照第 73 頁的第 5 章，『動態快取』的說明使用動態快取

如果您是使用伺服器型的階段作業管理，請參考 *WebSphere Application Server 5.0 Tuning Guide* 中的指示。這本手冊可以從 WebSphere Application Server Information Center 取得：<http://www.ibm.com/software/webservers/appserv/infocenter.html>）。

展開 **Monitor and Troubleshooting --> Performance --> Tuning Performance**。

- WebSphere Application Server

使用 *WebSphere Application Server 5.0 Tuning Guide* 作為如何調整系統的指南。這本手冊可以從 WebSphere Application Server Information Center 取得：

<http://www.ibm.com/software/webservers/appserv/infocenter.html>）。

- 展開 **Monitor and Troubleshooting --> Performance --> Tuning Performance**。
- 其他注意事項：
 - WebSphere 資料來源（最小與最大連線儲存池大小、陳述式快取大小）
 - 網站設計
 - 安全（配置、逾時、鑑別和存取控制）
 - Web 伺服器問題（程序處理、資源用法、快速回應快取加速器）

- WebSphere 引擎問題 (Java 虛擬機器或 JVM、傳輸佇列、JSP 檔快取、EJB 配置區)
- WebSphere Commerce 階段作業管理 (快取、將階段作業儲存在記憶體中或資料庫中)
- WebSphere Application Server 階段作業管理 (設定記憶體內階段作業計數、容許溢位、逾時間隔以及「分散式環境」設定)。
- NFS (網路檔案系統) 效能調整 (檔案伺服器調整)

使用 WebSphere Commerce PMI 模組監視效能

您可以使用 WebSphere Application Server Performance Monitoring Infrastructure (PMI) 監視您的 WebSphere Commerce 系統效能。以前的 WebSphere Commerce 效能監視器功能現在已經改到 WebSphere Commerce PMI 模組中。之後，效能資料可以使用 WebSphere Application Server 提供的各種工具加以監視和分析。網站管理者可以使用從那些工具收集的資訊，來偵測效能問題以及分析效能趨勢。那些工具可用來從本端或遠端機器上，測量 WebSphere Commerce 應用程式伺服器的效能。

WebSphere Commerce 應用程式伺服器會收集 URL、作業和 JSP 的統計值。每一個資料鍵都有一組相關的計數器，提供下列資訊：

- 平均回應時間
- 最後回應時間
- 最短回應時間
- 最長回應時間
- 命中數
- 合計回應時間
- 標準偏差

如果要設定 PMI，請參閱 WebSphere Application Server Information Center 的「監視效能」一節 (<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

設定 PMI 模組

Performance Monitoring Infrastructure (PMI) 使用一套主從架構結構。其伺服器會從 WebSphere Application Server 的各個元件收集效能資料。用戶端從一或多部伺服器擷取效能資料，然後處理那些資料。

WebSphere Application Server 伺服器收集記憶體中的 PMI 資料。這項資料包括一些計數器，如 servlet 回應時間以及資料連線儲存池用量。之後再以 Web 用戶端、Java 用戶端或 JMX 用戶端擷取這些資料點。WebSphere Application Server 有提供 Tivoli® Performance Viewer，這是一種 Java 用戶端，可以顯示以及監督效能資料。

若要使用效能監視工具，您需要依照下列方式，設定 WebSphere Commerce PMI 模組：

1. 如果要透過 PMI 介面監視效能資料，您必須先透過 WebSphere Application Server 管理主控台啟用效能監視服務，然後重新啟動伺服器。如需詳細步驟，請參閱 WebSphere Application Server Information Center 的「透過管理主控台，在應用程式伺服器中啟用 PMI 服務」一節 (<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

2. 您也可以透過 PMI 介面監視效能資料，方法為透過管理主控台，在 NodeAgent 中啟用效能監視服務。如需詳細步驟，請參閱 WebSphere Application Server Information Center 的「透過管理主控台，在 NodeAgent 中啟用效能監視服務」一節 (<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。
3. 做好上述設定之後，您就可以開始收集資料。

判定要啟用哪些資料計數器的監視層次可以動態設定，並不需要重新啟動伺服器。其做法有下列三種：

 - 透過 WebSphere Application Server 管理主控台啟用資料收集。
 - 透過 Tivoli Performance Viewer (舊稱為「資源分析程式」) 啟用效能監視器服務。
 - 使用指令行啟用效能監視服務。

如需詳細步驟，請參閱 WebSphere Application Server Information Center 中，有關上述作業的對應區段 (<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。
4. 監視與分析效能資料。

您可以採用多種工具來監視以及分析資料：

 - a. 使用 Tivoli Performance Viewer 監督效能資料。這套工具隨附在 WebSphere Application Server 中。
 - b. 使用其他 Tivoli 監督工具監督效能資料。
 - c. 使用使用者開發的監視工具監視效能資料。撰寫您自己的應用程式來監視效能資料。
 - d. 使用協力廠商的監視工具監視效能。

如需詳細步驟，請參閱 WebSphere Application Server Information Center 中，有關上述作業的對應區段 (<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

啓動 Tivoli Performance Viewer

Tivoli Performance Viewer 是 WebSphere Application Server 5.0 版中隨附的 WebSphere Application Server 效能監視器。其檢視器會藉由定期輪詢管理伺服器，擷取效能資料。這些資料會持續地收集，並在需要時從檢視器中擷取。啓動「效能檢視器」，並指定要從 WebSphere Application Server 管理主控台收集的資料層次。使用其圖形介面以表格或圖表擷取以及檢視資料，或將資料儲存在日誌檔中。您不但可以使用檢視器監視 Commerce PMI，也可以監視和 WebSphere Commerce 相關的其他同屬 WebSphere Application Server 效能資料。

如需詳細資訊，請參閱 WebSphere Application Server Information Center 中的「使用 Tivoli Performance Viewer (舊稱為「資源分析程式」) 監督效能」。

當您針對 WebSphere Commerce 啓動 Tivoli 效能檢視器時，您需要為 `WC_instance_name` 應用程式伺服器指定 SOAP 或 RMI 埠號，以測量 WebSphere Commerce 的相關計數器。不然的話，在預設的情況下，Tivoli Performance Viewer 會嘗試連接 `server1` 應用程式伺服器。判斷 `WC_instance_name` 應用程式伺服器的 SOAP 或 RMI 埠號的方法之一，是察看 `WC_instance_name` 的 `System.Out` 日誌檔。當伺服器啓動時，就會顯示 SOAP 或 RMI 埠號。

以下是啓動 Tivoli Performance Viewer 的步驟

1. 透過管理主控台啟用 PMI 服務。
2. 啟動 Tivoli Performance Viewer。其做法有下列兩種：
 - a. 從指令行啟動效能監視作業。移至 `WAS_installdir/bin` 目錄，然後執行 `tperfviewer script`。

在 Windows 2000 環境中，您可以依下列方式指定主機與埠號：

```
tperfviewer.bat host_name port_number connector_type
```

在 AIX 以及其他 UNIX® 平台上，請使用：

```
tperfviewer.sh host_name port_number connector_type
```

例如：`tperfviewer.bat localhost 8879 SOAP`

`connector_type` 可以是 SOAP 或 RMI。8879 是 SOAP 連接器的預設 ND 埠號。9809 是 RMI 連接器的預設 ND 埠號。

- b. 按一下開始 --> 程式集 --> **IBM WebSphere** --> **Application Server v.5.0** > **Tivoli Performance Viewer**。

Tivoli Performance Viewer 會偵測您使用的是哪一套 WebSphere Application Server 套件，然後使用預設的「遠端方法呼叫 (RMI)」連接埠進行連線。如果連線失敗，就會顯示一個對話框，以便提供新的連線參數。您可以透過指令行啟動效能檢視器方式，連接到遠端主機或不同的埠號。

3. 調整資料收集設定。

請參考 WebSphere Application Server Information Center 中，「設定效能監視層次」一節的指示 (<http://www.ibm.com/software/webservers/appserv/infocenter.html>)。

WebSphere Commerce PMI 模組報告值

在啟動 Tivoli Performance Viewer 之後，在「資源選擇」畫面中展開 **Commerce 計數器群組**。資料會根據商店 ID 下的「作業」、URL 以及「檢視畫面」分組。注意，所有作業指令都會在 `StoreId=0` 之下尋找。如果從「資源選擇」畫面中選取特定的 URL、「檢視畫面」或「作業」，「計數器選擇畫面」就會顯示 WebSphere Commerce 計數器。

這些是每一項「作業」、URL 或「檢視畫面」中包含的計數器，請注意所有計時值都是經歷時間：

計數器名稱

定義

平均回應時間

作業的平均回應時間。

最後回應時間

作業最後的回應時間。

最短回應時間

作業最短的回應時間。

最長回應時間

作業最長的回應時間。

命中 作業被呼叫的合計次數。

合計回應時間

作業的合計回應時間。

標準偏差

回應時間的標準偏差。用來計算標準偏差的公式假設資料符合標準分送。

其他效能工具

網站管理者可能也需要使用下列工具：

- IBM Tivoli Web Site Analyzer

IBM Tivoli Web Site Analyzer 可以藉由擷取、分析、儲存以及報告網站的用量、健全狀態、完整性以及網站內容等，看出訪客與網站的互動，以及網站的整體效能。您可以運用這項內視能力，將網站最佳化，以提高客戶的忠誠度和電子商業的效益。Web Site Analyzer 可以追蹤頁面內容的受歡迎程度，以及針對目標優惠或活動的產品購買，追蹤到特定的訪客或客戶區段。這項資訊可以指出由於極少人造訪某些網頁或產品頁面，因此可以減少投資或變更 Web 導覽。

如需有關 Web Site Analyzer 的詳細資訊，請造訪下列網址：

<http://www.ibm.com/software/sysmgmt/products/web-site-analyzer.html>

- Commerce Studio Page Detailer

用來分析網頁以及顯示識別、大小、來源以及遞送頁面上的每一個項目所需要的時間。您可以使用這些明細來找出哪些部分的效能可以改善，以增強一般使用者的瀏覽經驗。Page Detailer 可以和 Commerce Studio 的其他元件分開執行。

維護排程器

為了確保適當的效能，排程器必須做好充分的維護。提供日常維護以確定「排程器」正常執行是「網站管理者」的責任。

SCHSTATUS 表格含有一個項目，代表 SCHCONFIG 表格中每一次執行的工作。因為排程器所執行的工作相當多，所以保存在 SCHSTATUS 表格中的工作狀態記錄將變得非常的多。這可能會對 WebSphere Commerce 伺服器的效能帶來負面影響。因此，建議您使用「排程器狀態顯示」頁面來定期清理 SCHSTATUS 表格。這是位於「管理主控台」的**配置**功能表之下。刪除工作狀態記錄會減少 SCHSTATUS 表格的大小，因為它會根據時間戳記及工作參考號碼來清理工作。

另外，您可以排定 CleanJob 工作的執行時間，並自動縮減 SCHSTATUS 表格的大小。使用 endTime 參數來指定是否要移除前一週或前一個月的工作狀態記錄。如需有關 CleanJob 指令的語法與排程資訊，請參考您的平台的 WebSphere Commerce 正式作業與開發作業線上說明。

註：當您執行 CleanJob 時，商店公佈狀態會從商店公佈頁面中移除。

如果您不需要保存工作的歷程，可以將排程器的 autoClean 內容設為 ON。此設定會使得工作執行完成時，自動從 SCHSTATUS 表格中移除工作。

如果需要「排程器」的詳細資訊，請參閱第 21 頁的『排定服務排程』或參閱 WebSphere Commerce 線上說明。

LDAP 抄寫

WebSphere Commerce 支援使用 LDAP 進行鑑別，以及儲存鑑別和設定檔資料。部份資料會在 WebSphere Commerce 資料庫與 LDAP 伺服器之間抄寫。您可以使用 `ldapentry.xml` 檔來配置大部份抄寫方式。不同於前一版的 WebSphere Commerce 僅從特定的「成員子系統」指令觸發抄寫邏輯，現行版本的 WebSphere Commerce 是在物件層次執行抄寫，這表示只要需要就會發生抄寫。因此，如果 LDAP 伺服器關閉，將會產生錯誤。

在暫置作業伺服器上測試網站

如果要在暫置作業伺服器上測試網站，請遵循下列步驟：

1. 配置暫置作業伺服器。
2. 建立任何自訂表格的觸發指令。
3. 配置您的遠端資料庫（若適用的話）。
4. 將資料複製至暫置作業資料庫。
5. 測試網站。
6. 執行「暫置檢查」指令以確定沒有唯一索引鍵衝突。
7. 將資料延伸至正式作業資料庫。
8. 將檔案複製至正式作業伺服器。
9. 從 STAGLOG 表格刪除暫置物件。

配置暫置作業伺服器

任何 WebSphere Commerce 機器都可設定成暫置作業伺服器。暫置作業伺服器可以在安裝期間或安裝之後配置。如果需要在安裝 WebSphere Commerce 期間設定暫置作業伺服器的說明，請參閱 *WebSphere Commerce 安裝手冊*。如果要在安裝後設定暫置作業伺服器，請執行下列步驟：

1. 建立並配置個別的 WebSphere Commerce 實例，作為與「配置管理程式」搭配的暫置作業伺服器。
2. 確定您在資料庫畫面上選取了**使用暫置作業伺服器**勾選框，將實例配置成您的暫置作業伺服器。
3. 確定未在「配置管理程式」的**快取**畫面中啟用快取。

附註：您無法在暫置作業資料庫上啟用快取觸發指令。

建立自訂表格的觸發指令

觸發指令會在 STAGLOG 表格中建立一個項目，來識別資料庫記錄變更。如果表格含有相同的資料範圍和鍵值性質，您就可以修改新表格的現有觸發指令設定。

如果您未建立任何新表格，將不需要執行這個步驟。如果您已建立了新表格，請參閱為自訂的表格配置暫置作業伺服器的指示。

配置遠端資料庫

如果您在正式作業伺服器以外的機器上設定了暫置作業伺服器，則必須配置遠端資料庫。如果您計劃從暫置作業伺服器執行暫置公用程式，您需要將正式作業資料庫配置

成暫置作業伺服器中的遠端資料庫。如果您計劃從正式作業伺服器執行暫置公用程式，您需要將暫置作業資料庫配置成正式作業伺服器中的遠端資料庫。

DB2 若是 DB2 資料庫，請參閱 *DB2 管理手冊*。

Oracle 若是 Oracle 資料庫，請參閱該產品的文件。

複製資料到暫置作業資料庫

如果要將正式作業資料庫中的資料複製至暫置作業資料庫，請執行下列步驟：

1. 設定 PATH 環境變數。
2. 配置資料庫。
3. 切換至要寫入日誌檔的目錄。
4. 輸入下列指令：

• **Windows** **DB2**

```
stagingcopy -scope _all_ -sourcdb production_database_name  
-destdb staging_database_name
```

• **Windows** **Oracle**

```
stagingcopy -scope _all_ -sourcdb production_database_name  
-destdb staging_database_name -dbtype oracle -sourcdb_user  
user -sourcdb_passwd password -destdb_user  
user -destdb_passwd password
```

• **AIX** **Solaris** **Linux** **DB2**

```
. stagingcopy.sh -scope _all_ -sourcdb production_database_name  
-destdbstaging_database_name
```

• **AIX** **Solaris** **Oracle**

```
. stagingcopy.sh -scope _all_ -sourcdb production_database_name  
-destdb staging_database_name dbtype oracle -sourcdb_user  
user -sourcdb_passwd password  
-destdb_user user -destdb_passwd password
```

• **400** **DB2**

```
stagingcopy.sh SOURCEDB(production_database_name)  
SRCINST(production_instance_name)  
DESTDB(staging_database_name)  
DESTINST(staging_instance_name)  
SCOPE(_all_)
```

註：使用 *host:port:sid* 作為 Oracle 資料庫名稱。比方說，*myhost:1521:mydb*。

5. 檢查 *stagingcopy_yyyy.mm.dd_hh.mm.ss.zzz.log* 檔，以驗證指令是否成功。
6. 停止並重新啟動暫置作業伺服器實例。

執行「暫置檢查」指令

如果要檢查暫置作業資料庫與正式作業資料庫之間是否有潛在的唯一索引鍵衝突，請執行下列步驟：

如果要執行「暫置檢查」指令以確定沒有唯一索引鍵衝突，請執行下列步驟：

1. 設定 PATH 環境變數。

2. 配置資料庫。
3. 切換至要寫入日誌檔的目錄。
4. 輸入下列指令：

-  

```
stagingcheck -scope _unique_index_ -sourcedb staging_database_name
              -destdb production_database_name
```

-  

```
stagingcheck -scope _unique_index_ -sourcedb staging_database_name
              -destdb production_database_name -dbtype oracle -sourcedb_user
              user -sourcedb_passwd password -destdb_user
              user -destdb_passwd password
```

-    

```
. stagingcheck.sh -scope _unique_index_ -sourcedb staging_database_name
                 -destdb production_database_name
```

-   

```
. stagingcheck.sh -scope _unique_index_ -sourcedb staging_database_name
                 -destdb production_database_name dbtype oracle -sourcedb_user
                 user -sourcedb_passwd password -destdb_user
                 user -destdb_passwd password
```

-  

```
stagingcheck.sh SOURCEDB(staging_database_name)
                 SRCINST(staging_instance_name)
                 DESTDB(production_database_name)
                 DESTINST(production_instance_database_name)
                 SCOPE(_unique_index_)
```

註：使用 *host:port:sid* 作為 Oracle 資料庫名稱。比方說，*myhost:1521:mydb*。

5. 檢查 `stagingcheck_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

將資料延伸至正式作業資料庫

如果要將暫置作業資料庫中的資料延伸至正式作業資料庫，請執行下列步驟：

1. 設定 PATH 環境變數。
2. 配置資料庫。
3. 切換至要寫入日誌檔的目錄。
4. 輸入下列指令：






-  

```
stagingprop -scope _all_ -sourcedb staging_database_name -destdb
            production_database_name
```

-  

```
stagingprop -scope _all_ -sourcedb staging_database_name
            -destdb production_database_name -dbtype oracle -sourcedb_user
            user -sourcedb_passwd password -destdb_user
            user -destdb_passwd password
```

-    

- `stagingprop.sh -scope _all_ -sourcedb staging_database_name -destdb production_database_name`
-   
 - `stagingprop.sh -scope _all_ -sourcedb staging_database_name -destdb production_database_name dbtype oracle -sourcedb_user user -sourcedb_passwd password -destdb_user user -destdb_passwd password`
-  

```
stagingprop.sh SOURCEDB(staging_database_name)
SRCINST(staging_instance_name)
DESTDB(production_database_name)
DESTINST(production_instance_name)
SCOPE(_all_)
```

註：使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

5. 檢查 `stagingprop_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如果需要將資料延伸至正式作業資料庫的進一步資訊，請參閱範例。

複製檔案到正式作業伺服器

如果您在暫置作業伺服器上新增、變更或刪除影像或 HTML 檔（相對於資料庫中的資料），您必須以手動方式將這些檔案複製至正式作業伺服器。



如果要將檔案複製至正式作業伺服器，請執行下列步驟：



1. 將檔案複製至正式作業伺服器中的正確目錄。
 - a. 建立一個壓縮檔，來包含所有靜態 HTML 檔、相關的影像檔，以及其他內含的檔案。這個檔案應該包含新的、更新的，以及不變的檔案。
 - b. 將壓縮檔轉送至正式作業伺服器。
 - c. 將每一個壓縮檔解壓縮至正式作業伺服器目錄結構中的對應目錄。如果您已移動 HTML 檔，請編輯 WebSphere Commerce 配置，來指向新目錄中的檔案。
2. 刪除正式作業伺服器上未使用的目錄。





刪除暫置物件






如果要刪除暫置物件，請執行下列步驟：

1. 設定 PATH 環境變數。
2. 切換至要寫入日誌檔的目錄。
3. 輸入下列指令：

-  

```
dbclean -object staglog -type obsolete -db dbname
-days daysold -loglevel loglevel
```
-  

```
dbclean -object staglog -type obsolete -db dbname
-days daysold -loglevel loglevel -dbtype oracle
-dbuser user -dbpasswd password
```
-    

- `dbclean.sh -object staglog -type obsolete -db dbname -days daysold -loglevel loglevel`
-   
 - `dbclean.sh -object staglog -type obsolete -db dbname -days daysold -loglevel loglevel -dbtype oracle -dbuser user -dbpasswd password`
-  

```
dbclean.sh DATABASE(dbname)
          SCHEMA(schema_name)
          PASSWD(instance_password)
          OBJECT('staglog') TYPE('obsolete')
          DAYS(daysold)
          LOGLEVEL(loglevel)
```

註：使用 `host:port:sid` 作為 Oracle 資料庫名稱。比方說，`myhost:1521:mydb`。

4. 檢查 `dbclean_yyyy.mm.dd_hh.mm.ss.zzz.log` 檔，以驗證指令是否成功。

如果需要刪除暫置物件的其他範例，請參閱刪除物件的範例。

WebSphere Commerce Payments 效能調整參數

WebSphere Commerce Payments 含有可用來控制 WebSphere Commerce Payments 內部資源配置的調整參數。

注意：只有對 WebSphere Commerce Payments 非常熟悉的管理者才能修改這些參數。這些參數值若設定太高，則可能導致效能降低或 WebSphere Commerce Payments 啟動完全失敗。

強烈建議您：

- 變更時，僅做小幅度的更改。
- 一次只變更一個參數；衡量效果後再做其他變更。
- 在不重要的系統上進行徹底的變更測試後，才將變更引用到正式運作系統上。

WebSphere Commerce Payments 會管理下列執行緒儲存池集：

通訊協定執行緒儲存池

這個儲存池中的執行緒主要的任務是處理卡匣內伺服器端通訊協定特有的訊息。每一個定義了自己的 ComPoints 之卡匣，都有一個這樣的儲存池。

 內容 `wpm.poolsize` 可變更通訊協定執行緒儲存池大小。

您可以在 WebSphere Application Server 管理主控台中完成下列步驟，變更儲存池大小。

1. 展開伺服器。
2. 按一下 **Application Servers>wpm Commerce Payments Server>程序定義>Java 虛擬機器>自訂內容>wpm.poolsize**。
3. 鍵入屬意的值。預設值為 8。
4. 按一下**套用**。

在處理的過程中，通常會涉及到一些伺服器端的通訊協定訊息（增加的網路通訊層次），所以這些執行緒通常會針對每一個給定的要求保留一段時間。因此，依賴通訊協定執行緒來處理通訊協定訊息的卡匣，應配置足以容納一般數量的並行交易（包括重要的通訊協定訊息）之通訊協定執行緒儲存池。

服務執行緒儲存池

這個儲存池中的執行緒是供卡匣和組織架構在背景執行目前或未來作業。這個儲存池的預設執行緒數目為 6。

400 內容 `wpm.spoolsize` 可變更「服務」執行緒儲存池的大小。

您可以在 WebSphere Application Server 管理主控台中完成下列步驟，變更儲存池大小。

1. 展開伺服器。
2. 按一下 **Application Servers>wpm Commerce Payments Server>程序定義>Java 虛擬機器>自訂內容>wpm.spoolsize**。
3. 在**值**欄位中，指定屬意的儲存池大小。預設值為 6。
4. 按一下**套用**。

最初，服務執行緒最常用來處理排定於未來執行的作業。這些作業像是：重試與卡匣的後端處理程式間的異動，以及一些定期維護作業。

`wpm.disableDuplicateOrderCheck`

在處理新訂單及消除資料庫存取權時，此參數可指示 WebSphere Commerce Payments 略過重覆訂單檢查。這將造成更高的交易產量。因此我們建議僅在您的商家不會產生重覆訂單號碼時，才應該使用此參數。若這樣做，將導致這些訂單失敗。若要啓用這個參數，請透過 WebSphere Application Server 管理主控台完成下列步驟：

1. 展開伺服器。
2. 按一下 **Application Servers>wpm Commerce Payments Server>程序定義>Java 虛擬機器>自訂內容**。
3. 按一下**新建**。
4. 在**名稱**欄位中，鍵入參數名稱 `wpm.spoolsize`。
5. 在**值**中鍵入 **1**。
6. 按一下**套用**。

要訣：您的資料庫產品可能有並行連線數量的某些限制，限制您可以在指定時間對資料庫建立的連線數目。這可能是技術上的限制，或是您訂購資料庫產品的授權合約上所加諸的限制。這些因素是您在調整上述參數時所應考慮到的。如果您因為將其中的參數設定太高而超出此限制，則 `Payment Servlet` 起始設定將失敗。

管理 iSeries 上的 Payments 實例資料庫

WebSphere Commerce Payments 處理新訂單和付款時，Payments 的實例資料庫集合大小會跟著成長。如果 iSeries 儲存體空間開始填滿，系統操作員可能會接收到警告訊息，例如，CPI099C - 嚴重，儲存體到達下限。如果此警告訊息發生，您必須考慮釋放 WebSphere Commerce Payments 實例資料庫的空間。

日誌記載設定為 *YES，即會建立 WebSphere Commerce Payments 實例資料庫集合。這意味實例程式庫將包含資料庫日誌接收者（即指出有事件發生，例如，變更資料庫檔案的登錄）。在 iSeries 上，日誌 QSQJRN 是建立在 WebSphere Commerce Payments 實例程式庫中，而且名為 QSQJRNXXXX 的日誌接收者會建立在實例程式庫內。如果 WebSphere Commerce Payments 實例程式庫成長太大，則實例程式庫中的舊日誌接收者（名為 QSJRNXXXX，類型為 *JRNRCV）可能會被刪除，以釋放空間。大部份最近的日誌接收者仍將會附屬於資料庫而且應該不會被刪除。此外，資料庫表格也可以使用 IBM 服務程式提供的 Commerce Payments Pruneorders 公用程式刪改。

有關清除儲存體的資訊，請參閱 *AS/400 System Operation* 手冊 (SC41-4203)。

第 8 章 疑難排解

「網站管理者」負責排解疑難並尋找系統問題的解決方案。本篇涵蓋 Crossworlds® 配接器以及 WebSphere Commerce Payments 的疑難排解。請參閱 *WebSphere Commerce* 安裝，以取得排解下列疑難的相關資訊：

- 可下載的工具
- 日誌檔
- WebSphere Application Server 問題
- Web 伺服器問題
- WebSphere Commerce 問題
- 資料庫問題

CrossWorlds® 配接器

本節列出配置配接器時可能會遇到的問題，以及該採取哪些動作來解決這些問題。

- **問題：**InterChange Server (ICS) 以及儲存庫位在同一部機器上，而 WebSphere Commerce Server 位在另一部機器上。需要哪些配置資訊，Visigenic ORB 代理程式才能跨越子網路運作？此症狀是在嘗試連線時，蹦現的對話框，該對話框指出 WebSphere Commerce Server 「找不到 InterChange Server '<伺服器名稱>。伺服器可能不在執行中。」，即使您可以單獨確認該伺服器在執行中。

解決方案 1：

1. 建立一個使用者層次的環境變數，稱為 VBROKER_ADM，這個變數指向 Visibroker 安裝系統的 adm 目錄（例如 d:\inprise\vbroker\adm）。
2. 在 adm 目錄中，建立一個文字檔案，檔名為 agentaddr，這個檔案中包含要連接到此 ICS 的用戶端機器 IP 位址或主機名稱。
3. 重新啟動 osagent 程序，確定 VBROKER_ADM 存在啟動此程序的使用者或 shell 的環境中。您現在應該可以將 WebSphere Commerce Server 連接到 ICS。注意：每一個 osagent 都必須擁有其自己的 agentaddr 文字檔，其中包含 WebSphere Commerce Server 的遠端 IP。

註：agentaddr 沒有延伸，並且只應該在 ICS 端執行 osagent。

解決方案 2：其他人已成功地使用替代的環境變數 OSAGENT_ADDR_FILE，其中指定 agentaddr 檔案的路徑與名稱。

由於 ORB 需要幾分鐘的探索時間，因此請於 10 分鐘後再重新連線，如果不行再改用其他方法。

- **問題：**WebSphere Commerce Server 無法連接到多部 InterChange Server。WebSphere Commerce Server 機器上有設定 OSAGENT_ADDR 環境變數，但是只能連接到在指定機器上執行的 InterChange Server。您有兩部 InterChange Servers 分別在兩個不同的子網路上執行，並且 WebSphere Commerce Server 是在第三部機器上執行。每一部 InterChange Server 機器上都有執行 osagent，並且每一部機器都有正確配置 VBROKER_ADM 環境變數，使其指向包含 agentaddr 檔案（其中包含每一部 InterChange Server/osagent 機器的 IP 位址）的目錄。

解決方案：在每一部 osagent 機器的 VBROKER_ADM 環境變數中指定的目錄內，建立一個 localaddr 文字檔案，並指定其他 osagent 機器的 IP 位址或 DNS 別名。

- **問題：**您嘗試將 NT 機器的 WebSphere Commerce Server 連接到在 Solaris 上執行的 ICS。

解決方案：請確定 WebSphere Commerce Server 以及 ICS 位置的 agentaddr 檔案中，不只包含要連接的機器 IP 位址，同時也包括機器自己的 IP 位址。因此，NT 機器上的 agentaddr 檔案必須要有其自己的 IP 位址，也要有 Solaris 機器的 IP 位址。

註：

1. 如果有其他問題沒有在本節中提到，請聯絡您的 IBM 服務代表。
2. 在 iSeries、Linux、IBM eServer™ zSeries® 或 S/390® Linux 上，不支援 CrossWorlds® 配接器。

註：如果有其他問題沒有在本節中提到，請聯絡您的 IBM 服務代表。

WebSphere Commerce Payments

每一種企業運作模式都會處理唯一需求，而在運用 WebSphere Commerce Payments 的功能上則稍有不同。通常，WebSphere Commerce Payments 可以針對給定的商業環境調整，使其效能與功能達到最高。

調整高效能環境

傳輸控制通訊協定/網際網路通訊協定 (TCP/IP) Socket 等候時間

對 WebSphere Commerce Payments 提出的每一個要求，都會使 TCP Socket 處於 TIME_WAIT 狀態，並持續數分鐘。對於需要處理高容量要求的機器而言，可能會有大量的 Socket 處於 TIME_WAIT 或 TIME_CLOSED 狀態，而導致要求遭到拒絕，亦即產生回覆碼：無法連接 WebSphere Commerce Payments。對所有 TCP 連線而言，這是正常且必要的行為。

TCP Socket 會移至 TIME_WAIT 狀態一段時間，以確定未將 Socket 上任何後續的通訊誤認為新近連結之 Socket 上的新通訊。理論上來說，此段時間為 2 MSL（亦即，最大區段生命週期的兩倍）。實際上，TIME_WAIT 預設值在 Windows 作業系統與 Solaris 上為四分鐘，在 AIX 上為兩分鐘。在 iSeries 系統上，預設值大約為 2 分鐘。藉由改變作業系統的 TIME_WAIT 值，高容量使用者可減少此問題發生。下列範例說明如何在 Windows、AIX 以及 Solaris 作業系統上重新配置時間間隔。

例外：如果採用替代的堆疊，可能需要其他測量方式。

Windows:

1. 定位在「登錄」上：

```
HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\tcpip\Parameters  
\TcpTimeWaitDelay
```

如果此項目不存在於 Windows 登錄中，您需要依下列步驟 2 建立它。

2. 將此項目編輯成新 DWORD 項目。
3. 將之設為 30 到 300 間的任一值（此值代表秒數）。建議您將此值設為 30。

AIX:

```
no -o tcp_timewait=1
```

此值 (1) 採 15 秒遞增方式。例如，1 等於 15 秒，2 等於 30 秒，依此類推。建議您設為 1 或 2。

Solaris:

```
ndd -set /dev/tcp tcp_close_wait_interval 30000
```

此值 (30000) 是以毫秒計。

附錄 A. LDAP 實務內容

LDAP 實務內容：LDAP 伺服器作為成員儲存庫

使用 LDAP 伺服器作為成員儲存庫的實務內容，就是您可以建立 WebSphere Commerce 的新實例，然後指定它來使用 LDAP 伺服器作為成員儲存庫。在這個實務內容中，WebSphere Commerce 資料庫將移入「根組織」、「預設組織」，以及 'wcsadmin' 使用者項目的 Bootstrap 資料。

對於這個實務內容，您需要完成下列動作：

1. 建立目錄伺服器中所需要的字尾。WebSphere Commerce 將使用的使用者和組織實體項目將存在於這些字尾之下。
2. 在目錄伺服器上建立代表「根組織」、「預設組織」及 'wcsadmin' 的項目，其中「預設組織」及 'wcsadmin' 兩者位在「根組織」之下。請參閱“在 LDAP 伺服器中建立 Bootstrap 項目”相關鏈結，來取得範例 LDIF 檔。
3. 在 LDAP 伺服器中為 'wcsadmin' 使用者項目建立一個密碼。
4. 在目錄伺服器中建立了項目後，您應該以「根組織」（具有 MEMBER_ID -2001）及「預設組織」（具有 MEMBER_ID -2000）的正確識別名稱，來更新 ORGENTITY 表格中的 DN 直欄。
5. 設定 ldapentry.xml 檔，讓 WebSphere Commerce 屬性對映至 LDAP 屬性。確定使用者的搜尋庫已在 ldapentry.xml 檔中指定，以便可以找到 'wcsadmin' 使用者。
6. 以 'wcsadmin' 並提供適當的密碼來登入。

當透過 WebSphere Commerce 或另一個應用程式，在目錄伺服器上建立了更多使用者時，使用者便可以使用他們的 RDN 值或 DN 值來登入 WebSphere Commerce。如果使用者將使用 RDN 來登入，請確定搜尋庫已適當地在 ldapentry.xml 檔中指定。

附註：如果您建立了多個 WebSphere Commerce 實例，而且它們共用了同一個 LDAP 伺服器，則您應該確定每一個 WebSphereCommerce 實例內的 LDAP 伺服器配置都是相同的。

附錄 B. LDAP 檔

本附錄提供下列 LDAP 檔的詳細說明：

- ldapmap.dtd
- ldapentry.xml

ldapmap.dtd

ldapentry.dtd 取代了前一版 WebSphere Commerce 中所使用的 ldapmap.dtd 檔。下列是針對 LDAP 使用的 DTD。您不應該修改這個 DTD。

```
<!-- new mapping file -->
<!ELEMENT ldapentry (entry+)>
<!ELEMENT entry (ldapsetting,ldapmap)>
<!ELEMENT ldapsetting (ldaprtn,ldapocs,ldapbase)>
<!ELEMENT ldaprtn EMPTY>
<!ELEMENT ldapocs EMPTY>
<!ELEMENT ldapbase EMPTY>
<!ELEMENT ldapmap (map+)>
<!ELEMENT map (objectAttribute+,objectSeparator?,ldapAttribute)>
<!ELEMENT objectAttribute EMPTY>
<!ELEMENT objectSeparator EMPTY>
<!ELEMENT ldapAttribute EMPTY>

<!ATTLIST entry
entryName      (User|Organization|OrganizationalUnit) #REQUIRED>

<!ATTLIST ldapbase
defaultBase    CDATA      #REQUIRED
searchBase     CDATA      #REQUIRED>

<!ATTLIST ldaprtn
rdnName        CDATA      #REQUIRED keyAttrName    CDATA      #REQUIRED
keyObjName     CDATA      #REQUIRED>

<!ATTLIST ldapocs
objClass       CDATA      #REQUIRED>

<!ATTLIST objectAttribute
attrName       CDATA      #REQUIRED>

<!ATTLIST objectSeparator
attrSeparator  CDATA      #IMPLIED>
<!ATTLIST ldapAttribute
name           CDATA      #REQUIRED
operation      (replace|add) #REQUIRED
flow          (ldapToWcs|wcsToLdap|bothDirections) #REQUIRED>
<!-- End of DTD -->
```

ldapentry.xml

ldapentry.xml 檔取代了前一版 WebSphere Commerce 中所使用的 ldapmap.xml 檔。確定先前使用的 ldapmap.xml 檔的內容已轉送至 ldapentry.xml 檔。ldapentry.xml 檔提供最常用欄位的預設對映。預設綱目對映中的 LDAP 屬性是以預設使用者物件類別（person、organizationalPerson、inetOrgPerson 及 ePerson）和預設組織實體物件

類別 (organization 和 organizationalUnit) 為基礎。這些物件類別是針對 WebSphere Commerce 支援的所有 LDAP 伺服器而定義的。如果要延伸預設綱目對映超出這些物件類別，請確定要先在 LDAP 伺服器上執行綱目延伸，然後新屬性的對映就可以放置在 ldapentry.xml 檔中。建議您不要將下列屬性存留到 LDAP：WebSphere Commerce 內部產生的鍵值、經常變更的資料，如 USERS 表格中的 Lastsession。

如果需要 LDAP 屬性的相關資訊，請造訪下列網站：www.as400.ibm.com/ldap/schema。

```
<?xml version="1.0"?>
<!DOCTYPE ldapentry SYSTEM "ldapentry.dtd">
<ldapentry>
  <entry entryName="User">
    <ldapsetting>

<ldaprdn rdnName="uid"
  keyAttrName="logonId" keyObjName="UserRegistry"/>
    <ldapocs
objClass="top;person;organizationalPerson;inetOrgPerson"/>
    <ldapbase
defaultBase="o=Default Organization,o=Root Organization"
searchBase="o=Root Organization"/>
    </ldapsetting>
    <ldapmap>
      <map>

        <
objectAttribute attrName="logonPassword"/>

<ldapAttribute name="userPassword"
  operation="replace" flow="wcsToLdap"/>
      </map>
      <map>

        <objectAttribute attrName="lastName"/>

        <objectAttribute attrName="firstName"/>

        <objectSeparator attrSeparator=""/>

<ldapAttribute name="cn" operation="replace"
  flow="wcsToLdap"/>

      </map>
      <map>

        <objectAttribute attrName="lastName"/>

        <ldapAttribute name="sn" operation="replace"
" flow="bothDirections"/>

      </map>
      <map>

        <objectAttribute attrName="firstName"/>

        <ldapAttribute name="givenName" operation="replace
```



```

" flow="bothDirections"/>
    </map>
  <map>

  <objectAttribute attrName="phone1"/>

  <ldapAttribute name="homePhone" operation="replace"
" flow="bothDirections"/>
    </map>
  <map>

  <objectAttribute attrName="zipCode"/>

  <ldapAttribute name="postalCode" operation="replace"
" flow="bothDirections"/>
    </map>
  <map>

  <objectAttribute attrName="address1"/>

  <objectAttribute attrName="address2"/>

  <objectAttribute attrName="address3"/>

  <objectSeparator attrSeparator="/" />

  <ldapAttribute name="postalAddress" operation="replace"
" flow="bothDirections"/>
    </map>
  </ldapmap>
</entry>

entry entryName="Organization">
  <ldapsetting>

  <ldaprdn rdnName="o" keyAttrName="orgEntityName"
keyObjName="Organization"/>
  <
  ldapocs objClass="top;organization"/>
  <
  ldapbase defaultBase="o=Root Organization" searchBase="
o=Root Organization"/>
  </ldapsetting>
  <ldapmap>
  <map>

  <objectAttribute attrName="businessCategory"/>

  <ldapAttribute name="businessCategory" operation="
replace" flow="bothDirections"/>
    </map>
  <map>

  <objectAttribute attrName="description"/>

  <ldapAttribute name="description" operation="
replace" flow="bothDirections"/>
    </map>
  <map>

  <objectAttribute attrName="address1"/>

  <objectAttribute attrName="address2"/>

  <objectAttribute attrName="address3"/>

```

```

<objectSeparator attrSeparator="/" />

<ldapAttribute name="postalAddress" operation="replace
" flow="bothDirections" />
    </map>
    <map>

<objectAttribute attrName="phone1" />

<ldapAttribute name="telephoneNumber" operation="replace
" flow="bothDirections" />
    </map>
</ldapmap>
</entry>
    <entry entryName=
"OrganizationalUnit">
    <ldapsetting>

        <ldaprdn rdnName="ou" keyAttrName="orgEntityName"
keyObjName="Organization" />

<ldapocs objClass="top;organizationalUnit" />

<ldapbase defaultBase="o=Root Organization"
searchBase="o=Root Organization" />
    </ldapsetting>

    <ldapmap>
        <map>

<objectAttribute attrName="businessCategory" />

<ldapAttribute name="businessCategory" operation="replace
" flow="bothDirections" />
    </map>
    <map>

<objectAttribute attrName="description" />

<ldapAttribute name="description" operation="replac
e" flow="bothDirections" />
    </map>
    <map>

<objectAttribute attrName="address1" />

<objectAttribute attrName="address2" />

<objectAttribute attrName="address3" />

<objectSeparator attrSeparator="/" />

<ldapAttribute name="postalAddress" operation="replace
" flow="bothDirections" />
    </map>
    <map>

<objectAttribute attrName="phone1" />

<ldapAttribute name="telephoneNumber" operation="replace
" flow="bothDirections" />
    </map>
</ldapmap>
</entry>
</ldapentry>

```

ldapentry.xml 檔的格式如下：

entry entry 元素用來識別對映是針對哪一類型的成員。有效值是 User、Organization 或 OrganizationalUnit。

ldapsetting - ldaprDN - rdnName

指定哪一個 LDAP 屬性是 RDN 屬性。

ldapsetting - ldaprDN - keyAttrName

指定哪一個 WebSphere Commerce 屬性對映至 RDN 屬性。若是使用者，屬性名稱將與 UserRegistrationAdd 指令語法中所說明的內容名稱對應。屬性名稱會區分大小寫。同樣地，若是組織及組織單位，應使用 OrgEntityAdd 指令。

ldapsetting - ldaprDN - keyObjName

指定哪一個 WebSphere Commerce 存取 Bean 可以找到 keyAttrName 中所指定的 WebSphere Commerce 屬性。

ldapsetting - ldapocs - objClass

指定用來建立 LDAP 項目的 LDAP 物件類別。

ldapsetting - ldapbase - defaultBase

指定第一個搜尋庫 DN，如果使用 RDN 登入，則登入期間， WebSphere Commerce 將在這個搜尋庫 DN 下搜尋使用者。對於組織實體，defaultBase 指定第一個搜尋庫 DN，需要時，WebSphere Commerce 將在這個搜尋庫 DN 下搜尋組織實體。

ldapsetting - ldapbase - searchBase

指定基本 DN 的其餘部份，WebSphere Commerce 將在這個部份搜尋使用者和組織實體。

map -objectAttribute - attrName

WebSphere Commerce 屬性名稱。若是使用者，屬性名稱將與 UserRegistrationAdd 指令語法中所說明的內容名稱對應。屬性名稱會區分大小寫。同樣地，若是組織及組織單位，應使用 OrgEntityAdd 指令。

map -ldapAttribute - name

要對映至 attrName 中所指定的 WebSphere Commerce 屬性的 LDAP 屬性名稱。

map -ldapAttribute - flow

指定屬性值是從 LDAP 讀取、寫入 LDAP，或是既讀取也寫入。有效值是 ldapToWcs、wcsToLdap 或 bothDirections。

map -ldapAttribute - operation

指定是否應該為 LDAP 修改屬性值。有效值是 replace 或 add。replace 值將以提供的新資訊來更新目前的使用者資訊（比方說，將現有的電話號碼換成新的電話號碼）。add 值將併入另一個使用者項目（比方說，新增電話號碼至目前的使用者資訊、製作使用者的電話號碼清單）。

map - objectSeparator - attrSeparator

當儲存多個 WebSphere Commerce 至單一 LDAP 屬性或是從其中擷取它們時，所使用的分隔字元。

附錄 C. WebSphere Commerce Payments 指導教學

本指導教學會引導您進行第一次 WebSphere Commerce Payments 設定使用 OfflineCard Cassette。在這項起始設定作業過程中，爲了示範最常用的管理與付款功能， WebSphere Commerce Payments 以 OfflineCard Cassette 以及「結帳範例」提供教學指導支援。如需有關管理、配置與付款功能的詳細資訊，請參閱 WebSphere Commerce Payments 使用者介面的線上說明。

以下是設定可運作的 WebSphere Commerce Payments 使用 OfflineCard Cassette 時，必須進行的六項配置作業：

1. 存取 WebSphere Commerce Payments 使用者介面
2. 建立 WebSphere Commerce Payments 商家並授權使用付款卡匣
3. 定義 WebSphere Commerce Payments 使用者
4. 指定使用者的職務
5. 建立帳戶
6. 建立卡別

配置作業完成之後，您就可以準備開始下訂單，以及進行商家日常執行的付款處理作業：

7. 建立訂單
8. 核准訂單
9. 請款
10. 結清批次
11. 發出退款
12. 檢視每日的批次總額

步驟 1：存取 WebSphere Commerce Payments 使用者介面

第一個步驟是以「付款管理者」身份登入 Payments。

如果要登入 WebSphere Commerce Payments 使用者介面，請執行下列步驟：

1. 在 Web 瀏覽器中指向 `http://host_name:port/webapp/PaymentManager` 其中 `host_name` 是執行 Payments 的 Web 伺服器主機名稱，而 `port` 是 Payments 執行所在的埠號。
2. 鍵入您的 WebSphere Commerce Site 管理者 ID。
3. 鍵入您的 WebSphere Commerce Site 管理者密碼。
4. 按一下登入。

重要事項：如果 WebSphere Commerce Payments 實例使用的 HTTP 伺服器所配置的埠號不是預設埠號，在本指導教學中，請在 WebSphere Commerce Payments 網址鏈結的主機名稱之後包含埠號。

使用者介面右上角的圖示有下列用法：

- 按一下多頭箭頭可以重新整理頁面。
- 按一下左向箭頭可以返回前次造訪的頁面。

- 按一下問號可以存取頁面的環境定義相關線上說明。

步驟 2：建立 WebSphere Commerce Payments 商家及授權使用卡匣

如果您尚未登入，請以「付款管理者」身份登入 WebSphere Commerce Payments。此時您擁有廣域的檢視畫面與廣域的權限。配置 WebSphere Commerce Payments 的第一步是建立商家並授權該商家使用付款卡匣：執行下列步驟，建立商家並授權其使用卡匣：

1. 從導覽頁框中，按一下**商家設定**。
2. 從「商家設定」頁面中，按一下**新增商家**。
3. 在「商家設定」頁面中，鍵入下列資訊（請注意：在本指導教學中，必須在這些欄位中輸入等寬字體文字）：

表 11. 建立商家欄位

欄位名稱	說明
商家名稱	鍵入 Test Store 。這是您指定給商家的名稱。其作用只是顯示在使用者介面中。
商家號碼	鍵入 123456789 。這是您指定給商家的唯一號碼，以便在所有交易資料中識別該商家。
有權使用的卡匣	勾選 OfflineCard 旁的勾選框。勾選此框是授權商家使用此付款卡匣。

4. 按一下**建立商家**，儲存商家配置。

如果您已建立商家，並要授權其使用此卡匣，請執行下列步驟：

1. 按一下**商家設定**。
2. 按一下「商家名稱」。
3. 選取 **OfflineCard** 旁的勾選框。
4. 按一下**建立商家**。

該商家就會取得使用這個卡匣的授權。

步驟 3：定義 WebSphere Commerce Payments 使用者

在本指導教學中，您會用到下列使用者：

- 安裝期間建立的預設使用者（如需詳細資訊，請參閱 *WebSphere Commerce 安裝手冊*。）
- *Pat*，您將定義的使用者

使用 WebSphere Commerce 「組織管理主控台」，來達成如定義及管理使用者的作業。在 WebSphere Commerce Payments 中，定義使用者是兩個步驟的程序。例如，如要定義使用者 *Pat*，您必須使用 WebSphere Commerce 「組織管理主控台」，並為 *Pat* 指定「網站管理者」職務。然後，您可以直接在 Payments UI 中為 *Pat* 指定「商家管理者」使用者職務，或透過「組織管理主控台」指定。注意在指定存取權給使用者之前，您必須先建立一個商家。

如果要配置 Payments 使用者，請執行下列步驟：

1. 在 Web 瀏覽器中指向 https://host_name:port/adminconsole

2. 按一下**存取管理>使用者**。
3. 按一下**新建**。
4. 使用「新建使用者」精靈，建立新使用者 Pat。
5. 從「存取權限」頁面中，為 Pat 指定「網站管理者」職務。
6. 按一下 **Payments>使用者**。
7. 鍵入**使用者名稱** Pat。
8. 按一下**搜尋**。
9. 按一下 Pat 的名字。這時會顯示「使用者配置」頁面。
10. 選取屬意的商家名稱。
11. 選取屬意的職務圓鈕，在此範例中是「商家管理者」。
12. 按一下**更新**，儲存指定。

如果您將 WebSphere Commerce Payments 安裝在和 WebSphere Commerce 產品以外的遠端系統上，並且您希望在 WebSphere Commerce Payments 使用者介面中使用 WebSphere Commerce 樣式表，則您必須將位於 `Payments_installdir\samples\wcs\PMCustomUI.properties` 目錄中的 `PMCustomUI.properties` 檔案，複製到主要的 WebSphere Commerce Payments 安裝目錄中。

步驟 4：指定使用者職務

您必須指定下列 WebSphere Commerce Payments 職務之一給使用者，這些職務和對應的 WebSphere Commerce 職務相對映：

表 12. 職務對映

Payments 職務	WebSphere Commerce 職務
付款管理者	網站管理者
商家管理者	網站管理者
監督者	業務或業務經理
辦事員	客戶服務主管

建立下列使用者之後：

- 使用者 Pat
- 商家 Test Store

您可開始在 WebSphere Commerce Payments 配置中指定 Pat 的職務。

例外：您也可以指定沒有 WebSphere Commerce Payments 存取權的職務，拒絕使用者存取 WebSphere Commerce Payments。這種方式適用於當您只需要暫時性的拒絕某位使用者存取，例如，某位員工缺勤。如需有關 WebSphere Commerce Payments 職務許可權的詳細資訊，請參閱 *WebSphere Commerce Payments 程式設計與參考手冊* 中的「職務許可權表格」。

指定 Pat 擔任 Test Store 的「商家管理者」職務，在 Payments UI 中執行下列步驟：

1. 從導覽頁框中，按一下**使用者**。
2. 在「使用者搜尋」頁面中，輸入使用者名稱 Pat 然後按一下**搜尋**。

3. 從「使用者」頁面中，按一下使用者名稱 **Pat**。
4. 從**商家**捲動框中，選取 **Test Store**。
5. 選取**商家管理者的**圓鈕。
6. 按一下**更新**，儲存使用者配置。

現在請登出 WebSphere Commerce Payments 使用者介面，然後重新登入，這時請以「商家管理者」Pat 身份登入。

重要事項：在代管型環境中，由商務服務公司 (CSP) 建立 WebSphere Commerce Payments 在遠端為多個商家提供服務時，是由 CSP 扮演「付款管理者」角色，並且依照每個商家與該服務公司間的合約，由該公司代理所有商家。在這種實務內容中，商家是以 CSP (亦即付款管理者) 授予他的「商家管理者」權限，來配置自家的商家設定。

以商家管理者身份登入

如果要登出然後重新登入，請執行下列步驟：

1. 從導覽頁框中，按一下**登出管理**，返回主要 WebSphere Commerce Payments 登入頁面。
2. 鍵入**使用者 ID** Pat。
3. 鍵入為 Pat 定義的**密碼**。
4. 按一下**確定**。

在本指導教學中，從現在開始，您的職務將是使用者 Pat，擁有 Test Store 的「商家管理者」權限。您的 WebSphere Commerce Payments 使用者介面檢視畫面會被限制在商家管理功能範圍內；亦即身為「付款管理者」，您同時具有「商家」與「付款管理者」功能的廣域檢視畫面。

步驟 5：建立帳戶

到目前為止，您已定義一個商家 (Test Store)，並啓用一個付款卡匣 (OfflineCard Cassette)。身為「商家管理者」，您的第一項任務是要為 OfflineCard Cassette 建立一個帳戶。

帳戶是商家與金融機構 (負責處理該商家的交易) 間的關係。每一個付款卡匣可有多個帳戶。基於本指導教學，您將為 OfflineCard Cassette 建立一個帳戶。

如果要建立帳戶，請執行下列步驟：

1. 從導覽頁框中，按一下**商家設定**。
2. 從「商家設定」頁面中，按一下 Test Store 的 OfflineCard Cassette 圖示。
3. 從 OfflineCard Cassette 頁面中，按一下**帳戶**。
4. 在「帳戶」頁面中，按一下**新增帳戶**。
5. 填入下列欄位 (請注意：在本指導教學中，必須在這些欄位中輸入等寬字體文字)：

表 13. 新增帳戶欄位

欄位名稱	說明
帳戶名稱	鍵入 Test Account。這是您指定給帳戶的名稱。其作用只是顯示在使用者介面中。
帳戶號碼	鍵入 111111111。這是代管的服務公司或「商家管理者」指定給帳戶的唯一號碼，以便在所有交易資料中識別該帳戶。
金融機構名稱	鍵入 Test Bank。這是您所持帳戶往來的金融機構名稱。其作用只是在使用者介面中提供顯示資訊。
貨幣	選取這個帳戶要收集的貨幣。
批次結束時間	午夜過後幾分鐘後，卡匣會自動結束此帳戶的批次。若值為 0（零）代表午夜。1439 為容許的最大值。空值表示停用自動結束批次特性。

6. 按一下**建立帳戶**，為 OfflineCard Cassette 建立帳戶。

指定帳戶設定

若要使用指導教學，您不需要對剛建立給卡匣的帳戶及卡別指定任何特殊基本或進階設定。然而，您應該明白您可以指定某些選項給帳戶。例如，身為「商家」管理者的您可以定義帳戶是否應該具有與付款核准有關的有效期限。

若要從 OfflineCard Cassette 的 WebSphere Commerce Payments 使用者介面檢視帳戶設定，請執行下列步驟：

1. 在導覽頁框選取**商家設定**。
2. 從「商家設定」視窗中，按一下「測試商店」視窗中的 OfflineCard Cassette 圖示。
3. 從 OfflineCard Cassette 視窗中，按一下**帳戶**。
4. 選取「帳戶」視窗中的 **測試帳單**。
5. 從「測試帳戶」視窗中，選取**帳戶設定**。
6. 您可以按一下適當的選擇，檢視或變更基本或進階設定。例如，選取**進階設定**，以查看「自動付款處理程序選項」的「核准有效期限」設定。
7. 輸入適當的選項並選取**更新**，以更新帳戶設定。（若要取消，請選取「帳戶設定」視窗上的倒退箭頭。）

如需關於處理選項設定的詳細資訊，可以參閱線上說明。不過現在我們要探索其中一個選項--「核准有效期限」選項。

您可能想要設定一個核准有效期限，以控制付款核准的有效時間長度。設定一個核准有效期限可能有助於避免您的財務機構在核准有效期限後，因為請款而強加費用。例如，對於信用卡訂單，付款核准僅能持續信用卡發卡機構所建立的特定天數。如果您嘗試在核准過期後申請訂單付款，則您的財務機構可能會拒絕請款，或是要求您支付更多的請款費用。在 WebSphere Commerce Payments 中設定核准有效期限，您就可以避免這種情況，因為 WebSphere Commerce Payments 將阻止核准授權過期後的扣款。

如果付款核准已過期，而您嘗試進行請款，您會收到一則訊息，指出付款不處於核准狀態。在 WebSphere Commerce Payments 使用者介面中，您可以利用金額 0，使得付款作廢。一旦付款作廢，您就可以執行「核准」作業，將付款放入核准狀態。

若要使用核准有效期限選項，您使用的卡匣必須支援核准有效期限。WebSphere Commerce Payments 隨附的 OfflineCard、VisaNet、Paymentech 以及 CustomOffline 等卡匣，都支援核准有效期限。

重要事項： 您輸入作為核准有效期限的值會套用到所有與帳戶相關的卡別。如果您的每一個卡別都採用不同的付款核准有效期限規則，則請設定核准有效期限值作為最低公分母，或是為每一個卡別建立個別的帳戶。如果您為每一個卡別建立個別的帳戶，則您必須為每一個帳戶個別結算批次付款。

本章稍後將提供有關核准訂單、請款及結算批次的詳細資訊。

步驟 6：建立卡別

如果要使用 OfflineCard Cassette，您必須先針對和您的商店往來的信用卡公司，配置其卡別名稱。如此當購物者在結帳時，就可以從信用卡清單中選取適當的信用卡。如果要建立 OfflineCard Cassette 卡別，請執行下列步驟（注意在本指導教學中，ROBO 用來指卡別）：

1. 從導覽頁框中，按一下**商家設定**。
2. 從「商家設定」頁面中，按一下 Test Store 的 OfflineCard Cassette 圖示。
3. 從 OfflineCard Cassette 頁面中，按一下**帳戶**。
4. 從「帳戶」頁面中，按一下帳戶名稱 Offline Account。
5. 從 Offline Account 頁面中，按一下**卡別**。
6. 從「卡別」頁面中，按一下**新增卡別**。
7. 在**卡別名稱**欄位中，鍵入 ROBO。
8. 按一下**建立卡別**。

步驟 7：使用「結帳範例」建立訂單

身為「商家管理者」，您擁有廣域的商家權限，這意味著您可以執行：

- 商家特有的管理功能
- 所有付款處理功能

在現實的商業環境中，您可以選擇將付款處理作業委任給其他由商家定義的使用者，他們擁有有限付款處理權限，像是：監督者與辦事員。在本指導教學中，這些作業將由身為「商家管理者」的您來執行。當完成開始付款處理程序所需的所有 WebSphere Commerce Payments 與商家管理作業時，您就可以準備開始：

- 核准訂單
- 請款
- 結清批次
- 發出退款
- 檢視每日的批次總額

在本指導教學中，您要使用「結帳範例」工具來建立三張訂單，供進行付款處理。「結帳範例」工具提供一個使用者介面，可讓您用來建立範例訂單，以便測試卡匣實作。如果要存取「結帳範例」，您必須先依照以下說明，變更預設的使用者 ID 與密碼。

如果要存取 WebSphere Commerce Payments 的「結帳範例」並建立訂單，請執行下列步驟：

1. 從目錄：

```
WAS_installdir/installedApps/host_name/  
payments_instance_Commerce_Payments_App.ear/SampleCheckout.war
```

開啓配置檔 SampleCheckout.xml。

2. 在 SampleCheckout 元素處，變更下列屬性值：

```
pmHostName="fully_qualified_host_name "  
pmPort="port "  
default_userid="wc_userid "  
password="wc_password "
```

3. 儲存檔案。將您的瀏覽器指向 `http://host_name:port/webapp/SampleCheckout`，其中的 `host_name` 是執行 Payments 的 Web 伺服器機器主機名稱，而 `port` 是 Payments 執行所在的埠號。
4. 在「結帳範例」頁面中，輸入下列資訊（請注意：在本指導教學中，必須在這些欄位中輸入等寬字體文字）：

表 14. 「結帳範例」欄位

名稱	說明
商家號碼	鍵入 OfflineCard 結帳範例商家號碼 987654321。
訂單號碼	輸入任一數字，以代表訂單號碼。
金額	輸入任一金額，以代表訂單的總額。
貨幣	鍵入美元。代表下此訂單時所用的貨幣。
付款方式	選取 OfflineCard 作為付款類型。
卡別	選取要使用的 OfflineCard 卡別。
信用卡號碼	鍵入 4111111111111111。
有效期限	選取您的信用卡將要到期的月份與年度。

▶ 400 如果使用 iSeries 系統，您必須指定您使用的信用卡卡別。

5. 按一下**購買**。

再重複執行這些步驟兩次，如此您便有三份要處理其付款的訂單。

步驟 8. 核准訂單

當您使用「結帳範例」建好三張訂單後，就可以核准那些訂單。按下列這些步驟來核准訂單：

1. 再度將您的瀏覽器指向 `http://host_name:port/webapp/PaymentManager`，其中的 `host_name` 是執行 Payments 的 Web 伺服器機器主機名稱，而 `port` 是 Payments 執行所在的埠號，並以 Pat 身份登入。
2. 從導覽頁框中，按一下**核准**。
3. 從「核准」頁面中，選取您下的其中一張訂單的方框。
4. 按下**核准已選取**。「核准結果」頁面中會顯示您的核准要求狀態。
5. 當您的核准完成時，請按一下**回到「核准」螢幕**。

另有兩份訂單等著您核准。您可以在「核准」頁面中按一下**全部核准**，一次核准其全部金額。不過，為了做核准功能的更好示範，本指導教學會說明如何分別處理每一張訂單。

從「訂單」頁面核准訂單

在本節中，您要從「訂單」頁面核准訂單總金額的一部份。若訂單中的一些商品無法於處理訂單時交貨，例如商品補貨中，那麼，僅核准訂單的一部份很有幫助。

1. 從「核准」頁面中，針對等候核准的剩餘訂單之一，按一下其**訂單號碼**。
2. 在「訂單號碼」頁面中，您可以看到訂單的明細。請按一下**核准**來核准此訂單。
3. 「訂單號碼核准」頁面會顯示下列欄位：

表 15. 訂單號碼核准欄位

欄位名稱	說明
貨幣	下此訂單時所用的貨幣類型。此為唯讀欄位。
訂單金額	以下訂單時所用貨幣來表示的訂單總額。此為唯讀欄位。
已核准金額	由於尚未核准任何訂單金額，因此這個唯讀欄位顯示零。
請款的金額	由於尚未核准或已請款任何金額，因此這個唯讀欄位顯示零。
核准金額	訂單的總金額。
授權代碼	手動離線授權要求程序傳回的授權代碼。付款狀態變為已核准。
拒絕原因	手動離線授權要求程序傳回的拒絕原因。付款狀態變為已拒絕。
AVS 結果碼	「地址驗證系統」結果碼。

將核准金額改為 3.00。您可以選擇性地指定一個授權原因來核准該金額，或指定拒絕原因，以表示核准遭到拒絕。

4. 按一下**核准**，核准此訂單的金額 3 元。當核准程序完成時，「訂單」頁面會重新整理，然後顯示核准狀態。

使用核准並自動請款功能核准訂單

由於您只核准上一張處理訂單的一部份，在「核准」頁面還有兩張訂單項目。在這個步驟中，您將要使用核准並自動請款功能來核准剩下的訂單。

核准並自動請款功能可讓您核准訂單，並將訂單直接移到已請款狀態，而跳過已核准狀態。核准並自動請款功能會自動核准，並進行訂單付款的請款。因此，您可以將核准並自動請款視為核准帶有自動請款。使用「核准並自動請款」功能可以儘快將商品交貨給購買者並保證能夠扣款，例如，您銷售的是可下載軟體或電子資訊。不過，當您使用「核准並自動請款」功能時，您便無法設定有關核准的授權原因或拒絕原因。由於「核准並自動請款」會將核准與請款合併成一筆交易，因此如果您需要支付每一筆交易的費用時，這個功能可以節省成本。

執行下列步驟，以核准並自動請款功能核准訂單：

1. 從導覽頁框中，按一下**核准**。
2. 從「核准」頁面中，按一下**核准並自動請款全部**。當處理程序完成時，就會顯示每一張已提交進行核准並自動請款的訂單核准狀態。

- 當您的核准並自動請款程序完成時，請按一下**回到「核准」螢幕**。

步驟 9：請款

「請款」功能可讓您做訂單付款請款。單一訂單號碼可有多項相關的付款。在同一列示中您可能會看到同一訂單號碼出現多次，而每一次都有不同的付款資訊。

如果要進行請款，請執行下列步驟：

- 從導覽頁框中，按一下**請款**。
- 勾選列出的某一項付款旁的勾選框，並按一下**請款選擇項**。當處理程序完成時，會顯示已提交進行請款的付款項目之請款狀態。
- 完成請款時，按一下**回到「請款」螢幕**。

您也可以做付款的**局部請款**。如果要進行局部請款，請執行下列步驟：

- 從「請款」頁面中，針對等候請款的付款項目之一，按一下其**付款號碼**。
- 從「付款」頁面中，按一下**請款**。
- 在「訂單付款」頁面中，將請款金額改為 2.00，然後按一下**請款**。

步驟 10：結清批次

「批次」是付款與退款的集合，金融機構將其當成一個單元來處理。批次與商家及帳戶有關。您在前面練習中已請款的付款處理程序，此時會出現在批次中。您必須結清此批次，使金融機構開始處理程序。完成結清時，金融機構會負責資金的轉移。

如果要結清批次，請執行下列步驟：

- 從導覽頁框中，按一下**批次搜尋**。另外，您也可以按一下**結清**。
- 在「批次搜尋」頁面中，您可以輸入下列資訊縮小搜尋範圍（注意在本指導教學中，您不需要填入這些欄位）：

表 16. 批次搜尋欄位

欄位名稱	說明
商家	您要搜尋批次的商家名稱。如果 WebSphere Commerce Payments 資料庫中的商家少於 500 個，請從下拉清單中選取商家名稱。如果 WebSphere Commerce Payments 資料庫中的商家數目超過 500 個，請輸入商家名稱。
批次號碼	批次在商家中的唯一識別號碼。在請款時指定。
狀況	批次的狀況有： <ul style="list-style-type: none">• 開啓• 已結束
結帳狀態	此批次的結帳狀態為： <ul style="list-style-type: none">• 已抵銷：批次已順利抵銷，亦即，所有金額相符。• 餘額不符：嘗試對此批次進行抵銷失敗，亦即總金額不相符。
付款類型	識別下訂單時所用的付款類型或通訊協定，例如 OfflineCard。
批次開啓日期	使用之後以及之前欄位，搜尋在指定時間範圍開啓的批次： <ul style="list-style-type: none">• 之後指定一個日期，搜尋在此日期（含此日）之後開啓的所有批次。• 之前指定一個日期，搜尋在此日期（含此日）之前開啓的所有批次。

表 16. 批次搜尋欄位 (繼續)

欄位名稱	說明
批次結束日期	使用之前以及之後欄位，搜尋在指定時間範圍結束的批次： <ul style="list-style-type: none"> • 之後指定一個日期，搜尋在此日期（含此日）之後結束的所有批次。 • 之前指定一個日期，搜尋在此日期（含此日）之前結束的所有批次。
帳戶	針對其處理本訂單的帳戶。如果已定義的帳戶數目超過 500 個，請在輸入欄位中鍵入帳戶號碼。

3. 按一下**搜尋**。

要訣：您也可以使用之前以及之後欄位，排除某些批次以縮小搜尋結果。例如，您可以搜尋在 2003/08/01 之前開啓的所有批次，以及搜尋在 2003/08/15 之後開啓的所有批次，因而排除 2001/08/02 到 2001/08/14 之間開啓的批次。

4. 按一下**批次號碼**以檢視批次的相關資訊。

5. 按一下**批次明細**，查看此批次中所有付款與退款的明細。

6. 按一下**結清**以結清批次。處理程序完成時，結清狀態會顯示在「結清結果」頁面中。

如果要刪除老舊的資訊，您也可以按一下「結清結果」頁面中的**刪除**，刪除已結清的批次。當批次刪除時，亦會連帶刪除該批次所有的輔助資訊，亦即付款、退款與卡匣特有的資料。如果您需要保留所有付款資料，例如，供審核用，則不可刪除批次。

步驟 11：發出退款

退款是針對訂單發出，且最多可以給予訂單的總金額。

如果要發出退款，請執行下列步驟：

1. 如果要尋找要發出退款的訂單，從導覽頁框中按一下**訂單搜尋**。
2. 在「批次搜尋」頁面中，您可以輸入下列資訊以縮小搜尋範圍（注意在本指導教學中，您不需要填入這些欄位）：

表 17. 訂單搜尋欄位

欄位名稱	說明
商家	您要搜尋訂單的商家名稱。如果 WebSphere Commerce Payments 資料庫中的商家少於 500 個，請從下拉清單中選取商家名稱。如果 WebSphere Commerce Payments 資料庫中的商家數目超過 500 個，請輸入商家名稱。
訂單號碼	商家指定而唯一代表該「訂單」的號碼。
狀態	「訂單」的狀態有： <ul style="list-style-type: none"> • 已申請 • 已訂購 • 可退款 • 已取消 • 已結束
付款類型	識別下訂單時所用的付款類型或通訊協定，例如 OfflineCard。

表 17. 訂單搜尋欄位 (繼續)

欄位名稱	說明
訂單日期	使用以下的之後以及之前欄位，搜尋在指定時間範圍內開啓的訂單： <ul style="list-style-type: none"> • 之後指定一個日期，搜尋在此日期（含此日）之後開啓的所有訂單。 • 之前指定一個日期，搜尋在此日期（含此日）之前開啓的所有訂單。
訂單金額	<ul style="list-style-type: none"> • 貨幣下此訂單時所用的貨幣。從下拉列示中選取貨幣類型。 • 大於指定一值，以擷取訂單金額大於或等於此指定值的所有訂單。 • 小於指定一值，以擷取訂單金額小於或等於此指定值的所有訂單。
帳戶	針對其處理本訂單的帳戶。如果已定義的帳戶數目超過 500 個，請在輸入欄位中鍵入帳戶號碼。

3. 按一下**搜尋**。
4. 從「訂單搜尋結果」頁面中，按一下訂單狀態為「可退款」的訂單號碼，檢視該訂單的明細。
5. 從「訂單」頁面中，按一下**退款**，針對此訂單建立一項退款。
6. 「建立退款」頁面會顯示下列資訊：

表 18. 建立退款欄位

欄位名稱	說明
貨幣	下此訂單時所用的貨幣類型。此為唯讀欄位。
訂單金額	以下訂單時所用貨幣來表示的訂單總額。此為唯讀欄位。
已核准金額	以下訂單時所用的貨幣來表示已核准訂單的總金額。此為唯讀欄位。
請款的金額	以下訂單時所用的貨幣來表示已請款的訂單總金額。此為唯讀欄位。
退款金額	此欄位必須由「商家管理者」填入要退還給購物者的總金額。
授權原因	手動離線授權要求程序傳回的授權代碼。退款狀態變為「已退款」。
拒絕原因	手動離線授權要求程序傳回的拒絕原因。退款狀態變為「已拒絕」。

鍵入退款金額，然後按一下**退款**。

當退款處理程序完成時，「訂單」頁面會重新整理，並顯示退款狀態。剛剛建立的退款記錄會顯示在**退款**下。

步驟 12：檢視批次總額

本指導教學的最後一個步驟是檢視每日的批次總額。WebSphere Commerce Payments 的報告功能可讓您檢視處於已結束狀況的批次的**每日總額**。

如果要產生每日批次總額報告，請執行下列步驟：

1. 從導覽頁框中，按一下**報告**。
2. 從「報告」頁面中，按一下**每日批次總額**。

3. 在「批次總額報告」頁面中，鍵入要取得批次總額報告的**日期**。若將此欄位留白，可以產生當日的報告。
4. 鍵入或選取**商家名稱**。如果您沒有輸入商家名稱，則會列出指定日期的所有批次清單。如果有 500 個以上的批次，則會顯示前 500 個批次。
5. 按一下**搜尋**。

「每日批次總額」報告會計算您在**搜尋**頁面中指定的日期結束的所有批次的總額。由於您未指定搜尋日期，產生之報告中所含的是本日的批次總額。這些總額是以每種貨幣為基礎計算出來的，因此一種貨幣各佔一行。請注意：這些總額涵蓋所有付款類型發生的所有付款與退款，而非只有透過 **OfflineCard Cassette** 進行的這些款項。

現在您已經完成「付款管理者」與「商家管理者」在一天中的工作。儘管個別的商业模型可能不同，本指導教學中描述了建立可運作的 **WebSphere Commerce Payments** 的基本途徑，並示範經由 **WebSphere Commerce Payments** 施行的基礎付款程序。如需有關 **WebSphere Commerce Payments** 使用者介面中特定欄位的詳細資訊，請參閱線上說明。

附錄 D. WebSphere Commerce Payments 的國家語言支援 (NLS) 資訊

本章提供有關 WebSphere Commerce Payments 的國家語言支援 (NLS) 補充資訊。如果要安裝 WebSphere Commerce Payments 的國家語言版本 (NLV)，請一定要檢視本資訊。

NLS 提示和要訣

Adobe Acrobat Reader 限制

對於 AIX 平台上的某些語言來說，可能無法正確顯示 PDF 檔。若無法檢視或列印 WebSphere Commerce Payments PDF 檔（即文件檔），您可以在 Windows[®] 或 Windows 2000 作業系統上，使用瀏覽器檢視及列印檔案。這是 AIX Adobe Acrobat Reader 的限制。

字碼頁 若無法正確顯示西班牙文字元，請確定是使用字碼頁 ISO-8859-1。

WebSphere Commerce Payments 使用者名稱

WebSphere Commerce Payments 僅支援由下表中的字元組成的使用者名稱。無法存取含有這個字集以外的字元的使用者名稱。

本表格列示用來指定 WebSphere Commerce Payments 使用者名稱的已支援字集。

表 19. WebSphere Commerce Payments 使用者名稱和其他的可攜性字集 (PCS)

ASCII 十六進位值	20	21	22	23	24	25	26	27	28	29	2A	2B	2C	2D	2E	2F
字元	空格	!	"	#	\$	%	&	'	()	*	+	,	-	.	/
ASCII 十六進位值	30	31	32	33	34	35	36	37	38	39	3A	3B	3C	3D	3E	3F
字元	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
ASCII 十六進位值	40	41	42	43	44	45	46	47	48	49	4A	4B	4 C	4D	4E	4F
字元	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
ASCII 十六進位值	50	51	52	53	54	55	56	57	58	59	5A	5B	5 C	5D	5E	5F
字元	P	Q	R	S	T	U	V	W	X	Y	Z	[\]	^	--
ASCII 十六進位值	60	61	62	63	64	65	66	67	68	69	6A	6B	6 C	6D	6E	6F
字元	'	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
ASCII 十六進位值	70	71	72	73	74	75	76	77	78	79	7A	7B	7 C	7D	7E	
字元	p	q	r	s	t	u	v	w	x	y	z	{		}	~	

支援的語言和語言環境

下表列出 WebSphere Commerce Payments 支援的每一種語言和對應的語言環境。

表 20. *WebSphere Commerce Payments* 支援的語言和語言環境

語言	語言環境
巴西葡萄牙文	pt_BR
英文	en_US
法文	fr_FR
德文	de_DE
義大利文	it_IT
日文	Ja_JP
韓文	ko_KR
簡體中文	zh_CN
西班牙文	es_ES
繁體中文	Zh_TW

注意事項

Note to U.S. Government Users -- -- Documentation relating to restricted rights --
-- Use, duplication, or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract with IBM Corp.

本資訊是針對在美國所提供之產品與服務開發出來的。而在其他國家或地區中，IBM 不見得有提供本書中所提的各項產品、服務、或功能。要知道在您所在之區是否可用到這些產品與服務時，請向當地的 IBM 服務代表查詢。凡提及 IBM 產品、程式或服務時，不表示或暗示只能使用 IBM 產品、程式或服務。只要未侵犯 IBM 的智慧財產權，任何功能、產品或服務都可以取代 IBM 的產品。不過，其他非 IBM 產品、程式或服務在運作上的評價與驗證，其責任屬於使用者。

IBM 可能已經申請與本書有關（包括本書的主題內容）的各項專利權，或者具有正在審理中的專利權申請書。本書使用者並不享有前述專利之任何授權。您可以書面方式來查詢授權，來函請寄到：

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

若要查詢有關二位元組 (DBCS) 資訊的特許權限事宜，請聯絡您國家或地區的 IBM 智慧財產部門，或者用書面方式寄到：

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chrome, Minato-ku
Tokyo 1061, Japan

下列段落若與該國之法律條款抵觸，即視為不適用：

IBM 就本書僅提供「交付時之現況」保證，而並不提供任何明示或默示之保證，如默示保證書籍之適售性或符合客戶之特殊使用目的；若有些地區在某些交易上並不允許排除上述保證，則該排除無效。

本書中可能有技術上或排版印刷上的訛誤。因此 IBM 會定期修訂；並將修訂後的內容納入新版中。同時，IBM 得隨時改進並（或）變動本書中所提及的產品及（或）程式。

資料中提供的非 IBM 網站僅供用戶參考方便，絕不代表為那些網站背書。那些網站上的內容並非本 IBM 產品內容的一部份，用戶使用該網站時應自行承擔風險。

IBM 對您以任何方式提供的資訊隨時享有使用或公開權，且不須負法律責任。

本程式的獲授權者若希望取得相關資料，以便使用下列資訊者可洽詢 IBM。其下列資訊指的是：（1）獨立建立的程式與其他程式（包括此程式）之間更換資訊的方式（2）相互使用已交換之資訊方法。若有任何問題請聯絡：

Lab Director
IBM Canada Ltd. Laboratory
8200 Warden Avenue
Markham, Ontario
L6G 1C7
加拿大

上述資料之取得有其特殊要件，在某些情況下必須付費方得使用。

IBM 基於雙方之「IBM 客戶合約」、「IBM 國際程式授權合約」（或任何同等合約）條款，提供本資訊中所述的授權程式與其所有適用的授權資料。

本書所提及之非 IBM 產品資訊，取自產品的供應商，或其公佈的聲明或其他公開管道。IBM 並未測試那些產品，所以無法確認性能準確度、相容性或任何其他與非 IBM 產品有關的索賠。有關非 IBM 產品的性能問題應直接洽詢該產品供應商。

本資訊中的範例包含了用於日常商業活動的資料及報告。爲了盡可能詳細，範例中涵蓋了個人、公司、品牌及產品的名稱。所有名稱純屬虛構，如有雷同純屬巧合。

本文件可能包含其他公司的產品，包括參考這些公司的網際網路網址。IBM 對這類資訊是否正確、完整及可使用不負任何責任。

本產品採用 SET 通信協定。

著作權授權：

本書包含範例應用程式的原始檔語言，用來說明在各種作業平台上的程式設計技術。客戶不需付費給 IBM 即可以任何格式複製、修改或散布這些範例程式，作爲開發、使用、行銷或散布符合範例程式針對的目標作業平台應用程式設計介面的應用程式。這些範例並未在所有條件下進行完整測試。IBM 不保證或暗示這些程式的可靠性、有用性或功能。客戶不需付費給 IBM 即可以任何格式複製、修改或散布這些範例程式，作爲開發、使用、行銷或散布符合 IBM 應用程式設計介面的應用程式。

這些範例程式或任何衍生程式的全部或部分拷貝，都必須包括下列版權聲明：

(C) (貴公司名稱) (年度)。此程式碼衍生自 IBM Corp. 範例程式。(C) Copyright IBM Corp. 1996, 2003. All rights reserved.

商標

IBM 標誌以及下列專有名詞是 International Business Machines Corporation 在美國與（或）其他國家或地區的商標或註冊商標：

- AIX
- CrossWorlds
- DB2 Extenders™
- DB2 Universal Database™
- IBM iSeries
- Intelligent Miner™
- Lotus®
- MQSeries

- OS/400
- pSeries™
- S/390
- VisualAge®
- WebSphere
- xSeries®
- zSeries
- Redbooks
- SecureWay
- DFS
- VisualAge
- DB2
- AS/400

Solaris、Java 與所有包含 Java 的商標與標誌是 Sun Microsystems, Inc. 在美國與（或）其他國家或地區中的商標或註冊商標。

Microsoft®、Windows 與 Windows 標誌為 Microsoft Corporation 在美國與（或）其他國家或地區中的註冊商標。

SET 與 SET 標誌是 SET Secure Electronic Transaction™ LLC 的商標。

UNIX 是 The Open Group 公司在美國和其他國家或地區的註冊商標。

其他公司名稱、產品名稱或服務名稱可能是其他者的商標或服務標記。

讀者意見表

為使本書盡善盡美，本公司極需您寶貴意見；懇請您閱讀後，撥冗填寫下表，惠予指教。

請於下表適當空格內，填入記號（√）；我們會在下一版中，作適當修訂，謝謝您的合作！

評估項目	評估意見	備註
正確性	內容說明與實際程序是否符合 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
	參考書目是否正確 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
一致性	文句用語及風格，前後是否一致 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
	實際產品介面訊息與本書中所提是否一致 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
完整性	是否遺漏您想知道的項目 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
	字句、章節是否有遺漏 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
術語使用	術語之使用是否恰當 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
	術語之使用，前後是否一致 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
可讀性	文句用語是否通順 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
	有否不知所云之處 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
內容說明	內容說明是否詳盡 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
	例題說明是否詳盡 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
排版方式	本書的形狀大小，版面安排是否方便閱讀 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
	字體大小，顏色編排，是否有助於閱讀 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
目錄索引	目錄內容之編排，是否便於查找 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
	索引語錄之排定，是否便於查找 <input type="checkbox"/> 是 <input type="checkbox"/> 否	
	※評估意見為"否"者，請於備註欄提供建議。	

其他：（篇幅不夠時，請另外附紙說明。）

上述改正意見，一經採用，本公司有合法之使用及發佈權利，特此聲明。

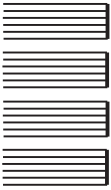
註：您也可將寶貴的意見以電子郵件寄至 NLSC01@tw.ibm.com，謝謝。

IBM WebSphere Commerce
管理手冊
5.5 版

折疊線

台北市 105 敦化南路一段 2 號 4 樓

臺灣國際商業機器股份有限公司 啟
大中華研發中心 軟體國際部



廣告回信
台灣地區郵政管理局
登記
北台字第 00176 號

(免貼郵票)

寄件人 姓名：
地址：

寄

折疊線

讀者意見表

IBM