



# Service Registry 3.1 管理指南



Sun Microsystems, Inc.  
4150 Network Circle  
Santa Clara, CA 95054  
U.S.A.

文件號碼：820-0538  
2007年2月

Copyright 2007 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 版權所有。

Sun Microsystems, Inc. 對本文件所述產品所採用的技術擁有相關智慧財產權。特別是(但不僅限於)，這些智慧財產權可能包含一項或多項美國專利，或者一項或多項美國及其他國家/地區的申請中專利。

美國政府權利 — 商業軟體。政府使用者均應遵守 Sun Microsystems, Inc. 的標準授權合約和 FAR 及其增補文件中的適用條款。

本發行物可能包含由協力廠商開發的材料。

本產品中的某些部分可能源自加州大學授權的 Berkeley BSD 系統的開發成果。UNIX 是在美國及其他國家/地區的註冊商標，已獲得 X/Open Company, Ltd. 專屬授權。

Sun、Sun Microsystems、Sun 標誌、Solaris 標誌、Java 咖啡杯標誌、docs.sun.com、Java 與 Solaris 是 Sun Microsystems, Inc. 在美國及其他國家/地區商標或註冊商標。所有 SPARC 商標都是 SPARC International, Inc. 在美國及其他國家/地區商標或註冊商標，經授權後使用。凡具有 SPARC 商標的產品都是採用 Sun Microsystems, Inc. 所開發的架構。

OPEN LOOK 與 Sun<sup>TM</sup> Graphical User Interface (Sun 圖形化使用者介面) 都是由 Sun Microsystems, Inc. 為其使用者與授權者所開發的技術。Sun 感謝 Xerox 公司在研究和開發視覺化或圖形化使用者介面之概念上，為電腦工業所做的開拓性貢獻。Sun 已向 Xerox 公司取得 Xerox 圖形化使用者介面之非獨占性授權，該授權亦適用於使用 OPEN LOOK GUI 並遵守 Sun 書面授權合約的 Sun 公司授權者。

本出版物所涵蓋的產品和包含的資訊受到美國出口控制法規的控制，並可能受到其他國家/地區進出口法規的管轄。嚴禁將本產品直接或間接地用於核武器、飛彈、生化武器或海上核動力裝備，也不得將本產品直接或間接地提供給核武器、飛彈、生化武器或海上核動力裝備的一般使用者。嚴禁將本產品出口或再出口至美國禁運的國家/地區或美國出口除外清單(包括但不僅限於被拒的個人和特別指定的國家/地區的公民清單)中包含的實體。

本文件以其「原狀」提供，對任何明示或暗示的條件、陳述或擔保，包括對適銷性、特殊用途的適用性或非侵權性的暗示保證，均不承擔任何責任，除非此免責聲明的適用範圍在法律上無效。

# 目錄

---

前言 .....	7
<b>1 配置和設定 Service Registry .....</b>	<b>13</b>
配置 Service Registry .....	13
▼ 在 [以後配置] 方式安裝之後以 Root 身份使用自訂特性配置 Service Registry .....	16
▼ 在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry .....	17
管理 Service Registry 的 Application Server 網域 .....	18
▼ 使用 Application Server Admin Console .....	20
▼ 停止並重新啓動 Registry 的 Application Server 網域 .....	20
▼ 將根憑證增加至 Registry 網域中的信任的憑證 .....	21
配置非預設 Application Server 安裝的 Service Registry .....	22
▼ 編輯一份 install.properties 檔案的副本 .....	22
針對 Registry 網域配置 Java 虛擬機器 (JVM) .....	23
▼ 配置 Registry 網域的 JVM 選項 .....	23
建立管理員 .....	24
▼ 建立管理員 .....	24
指定可執行使用者註冊的人 .....	25
▼ 限制使用者註冊 .....	26
啓用登錄 (Registry) 物件的版本控制 .....	26
▼ 啓用 Registry 物件的版本控制 .....	26
停用 WSDL Cataloger .....	27
▼ 停用 WSDL Cataloger .....	27
配置 Web Console .....	28
增加預先定義的查詢 .....	28
▼ 增加預先定義的查詢 .....	28
變更預設查詢 .....	30
▼ 變更預設查詢 .....	30

隱藏分類方案 .....	30
▼ 隱藏分類方案 .....	31
配置搜尋結果顯示 .....	31
▼ 配置搜尋結果顯示的列數 .....	32
▼ 配置搜尋結果區域的欄 .....	32
重新安裝 Service Registry .....	34
▼ 停止並刪除 Registry 的 Application Server 網域 .....	34
▼ 重新安裝 Service Registry 資料庫 .....	34
管理 Java DB 資料庫 .....	35
▼ 要求資料庫認證 .....	36
▼ 執行資料庫的離線備份 .....	36
▼ 從內嵌式模式切換至網路伺服器模式 .....	37
▼ 執行資料庫的線上備份 .....	38
<b>2 使用 Administration Tool .....</b>	<b>41</b>
關於 Admin Tool .....	41
啓動 Admin Tool .....	41
批次模式 .....	42
互動式模式 .....	42
Admin Tool 命令行選項 .....	42
使用 Admin Tool 將內容發佈到登錄 (Registry) .....	44
▼ 讓您自己可以將內容發佈到登錄 (Registry) .....	44
Admin Tool 功能 .....	45
許可權 .....	45
顯示異常 .....	45
識別登錄 (Registry) 物件 .....	45
語言環境對指定名稱的影響 .....	46
大小寫相符性 .....	46
使用 Admin Tool 指令 .....	46
add association .....	47
add user .....	48
cd .....	54
chown .....	55
cp .....	55
echo .....	57

---

help .....	57
import .....	58
keystoreMover .....	59
lcd .....	61
ls .....	62
pwd .....	63
quit .....	63
rm .....	63
select .....	65
set .....	65
show .....	66
users .....	67
索引 .....	69



# 前言

---

「Service Registry 3.1 管理指南」說明如何在安裝後配置 Service Registry (「Registry」)，以及如何使用與 Registry 一起提供的 Administration Tool。本書也說明其他管理作業，例如備份和復原 Registry 資料庫。

Service Registry 屬於 ebXML Registry，是一種聯合登錄與儲存庫，可管理標準和可延伸中介資料說明之所有類型的電子內容。它為服務導向架構 (SOA) 和其他內容與中介資料提供聯合、安全資訊管理功能，可支援 ebXML Registry 3.0 和 UDDI 3.0 登錄協定。

## 本書適用對象

「管理指南」適用於需要安裝、解除安裝和管理 Registry 的使用者，以及要為 Registry 大量建立內容而不使用 Web Console 進行此操作的使用者。

您應該熟悉您作業系統 (Solaris™ 作業系統、Linux 或 HP-UX) 上的 UNIX® 指令 shell 環境基本概念。

## 閱讀本書之前

在閱讀本書之前，您必須依照「Sun Java Enterprise System 5 Installation Guide for UNIX」中的說明安裝 Registry。

Service Registry 是 Sun Java Enterprise System (「Java ES」) 的元件；它是一種軟體基礎架構，支援分散於網路或網際網路環境中的企業應用程式。您應該熟悉 <http://docs.sun.com/coll/1286.2> 和 <http://docs.sun.com/coll/1412.2> 中的 Java ES 文件。

某些管理作業需要您熟悉這些規格的基本概念：

- [ebXML Registry Information Model Version 3.0](#)
- [ebXML Registry Services and Protocols Version 3.0](#)

## 本書架構

本書內容如下：

第 1 章說明如何在安裝之後配置 Service Registry，以及如何執行其他管理作業。

第 2 章說明 Administration Tool 的使用。

## Service Registry 文件集

Service Registry 文件集可從 <http://docs.sun.com/coll/1314.2> 和 <http://docs.sun.com/coll/1413.2> 取得。若要瞭解 Service Registry，請參閱下表中列出的書籍。

表 P-1 Service Registry 文件

文件標題	內容
「Service Registry 3.1 版本說明」	包含有關 Service Registry 的最新資訊 (包括已知問題)。
「Service Registry 3.1 管理指南」	說明安裝 Service Registry 之後如何進行配置，以及如何使用 Registry 隨附的管理工具。還說明如何執行其他管理作業。
「Service Registry 3.1 使用者指南」	說明如何使用 Service Registry Web Console 來搜尋 Service Registry 和將資料發佈到其中。
「Service Registry 3.1 Developer's Guide」	說明如何使用 Java API for XML Registries (JAXR) 來搜尋 Service Registry 和將資料發佈到其中。

## 相關書籍

安裝 Service Registry 時，會將其部署至 Sun Java System Application Server。如需有關管理 Application Server 的資訊，請參閱「Sun Java System Application Server Enterprise Edition 8.2 Administration Guide」。

Java ES 文件集說明部署規劃和系統安裝。系統文件的 URL 為 <http://docs.sun.com/coll/1286.2> 和 <http://docs.sun.com/coll/1412.2>。如需有關 Java ES 的介紹，請按下表所列順序參閱以下書籍。



表 P-2 Java Enterprise System 文件

文件標題	內容
「Sun Java Enterprise System 5 Release Notes for UNIX」	包含有關 Java ES 的最新資訊 (包括已知問題)。此外，元件有各自的版本說明，如版本說明集
「Sun Java Enterprise System 5 Release Notes for Microsoft Windows」	( <a href="http://docs.sun.com/coll/1315.2">http://docs.sun.com/coll/1315.2</a> 和 <a href="http://docs.sun.com/coll/1426.2">http://docs.sun.com/coll/1426.2</a> ) 所列。
「Sun Java Enterprise System 5 Technical Overview」	介紹 Java ES 的技術與概念基礎。說明元件、架構、程序和功能。
「Sun Java Enterprise System Deployment Planning Guide」	介紹基於 Java ES 規劃和設計企業部署解決方案。說明部署規劃和設計的基本概念和原則，討論解決方案生命週期，並提供基於 Java ES 規劃解決方案時要使用的高階範例和策略。
「Sun Java Enterprise System 5 Installation Planning Guide」	協助您開發 Java ES 部署中的硬體、作業系統及網路方面的實作規格。說明諸如元件相依性等安裝和配置規劃時需要解決的問題。
「Sun Java Enterprise System 5 Installation Guide for UNIX」	引導您進行安裝 Java ES 的程序。還說明安裝後如何配置元件，以及如何驗證這些元件是否正常工作。
「Sun Java Enterprise System 5 Installation Guide for Microsoft Windows」	
「Sun Java Enterprise System 5 Installation Reference for UNIX」	指示更多關於配置參數的資訊，提供工作表做為配置規劃之用，並列出參考資料，例如 Solaris 作業系統和 Linux 作業環境中的預設目錄和連接埠號。
「Sun Java Enterprise System 5 Upgrade Guide for UNIX」	提供從先前安裝版本升級至 Java ES 5 的指示。
「Sun Java Enterprise System 5 Upgrade Guide for Microsoft Windows」	
「Sun Java Enterprise System 5 Monitoring Guide」	提供為各個產品元件設定監視架構的說明，以及使用 Monitoring Console 檢視即時資料和建立監視規則的說明。
「Sun Java Enterprise System Glossary」	定義 Java ES 文件中使用的專有名詞。

關於 Java ES 及其元件的所有文件的 URL 為 <http://docs.sun.com/prod/entsys.5> 和 [http://docs.sun.com/prod/entsys.5?l=zh\\_TW](http://docs.sun.com/prod/entsys.5?l=zh_TW)。

## 預設路徑和檔案名稱

下表說明本書中使用的預設路徑和檔案名稱。

表 P-3 預設路徑和檔案名稱

預留位置	說明	預設值
<i>ServiceRegistry-base</i>	代表 Service Registry 的基底安裝目錄。	Solaris 作業系統： <code>/opt/SUNWsrvc-registry</code>  Linux 與 HP-UX 系統： <code>/opt/sun/srvc-registry</code>
<i>RegistryDomain-base</i>	代表 Service Registry 的 Application Server 網域和 Service Registry 資料庫所在的目錄。	Solaris 作業系統： <code>/var/opt/SUNWsrvc-registry</code>  Linux 與 HP-UX 系統： <code>/var/opt/sun/srvc-registry</code>
<i>Ant-base</i>	表示 Ant 工具 Java ES 版本的目錄位置。	Solaris 作業系統： <code>/usr/sfw/bin</code>  Linux 與 HP-UX 系統： <code>/opt/sun/share/bin</code>

## 印刷排版慣例

下表描述本書在印刷排版上所做的變更。

表 P-4 印刷排版慣例

字體	意義	範例
<b>AaBbCc123</b>	指令、檔案及目錄的名稱；螢幕畫面輸出。	請編輯您的 <code>.login</code> 檔案。 請使用 <code>ls -a</code> 列出所有檔案。 <code>machine_name% you have mail.</code>
<b>AaBbCc123</b>	您所鍵入的內容 (與螢幕畫面輸出相區別)。	<code>machine_name%<b>su</b></code> <code>Password:</code>
<i>AaBbCc123</i>	將用實際的名稱或數值取代的預留位置。	移除檔案的指令為 <code>rm 檔案名稱</code> 。
<b>AaBbCc123</b>	新術語，要強調的詞。	<b>快取記憶體</b> 是儲存在本機的副本。 請 <b>不要</b> 儲存此檔案。 <b>備註</b> ：某些重點項目在線上以粗體顯示。

表 P-4 印刷排版慣例 (續)

字體	意義	範例
「AaBbCc123」	用於書名及章節名稱。	請參閱「使用者指南」中的第 6 章。

## 指令範例中的 Shell 提示符號

下表顯示預設系統提示符號和超級使用者提示符號。

表 P-5 Shell 提示符號

Shell	提示符號
UNIX 和 Linux 系統上的 C shell	machine_name%
UNIX 和 Linux 系統上的 C shell 超級使用者	machine_name#
UNIX 和 Linux 系統上的 Bourne shell 和 Korn shell	\$
UNIX 和 Linux 系統上的 Bourne shell 和 Korn shell 超級使用者	#
Microsoft Windows 指令行	C:\

## 符號慣例

下表說明在本書中可能用到的符號。

表 P-6 符號慣例

符號	說明	範例	涵義
[ ]	含有選擇性引數和指令選項。	ls [-l]	-l 非必要選項。
{   }	包含一組必要指令選項的選擇。	-d {y n}	-d 選項需要您使用 y 引數或 n 引數。
\${ }	表明變數參照。	\${com.sun.javaRoot}	參照 com.sun.javaRoot 變數的值。
-	同時按下多個按鍵。	Control-A	當您按住 Control 鍵時按下 A 鍵。
+	連續按下多個按鍵。	Ctrl+A+N	按下 Ctrl 鍵，釋放該鍵，然後按下隨後的鍵。
→	表示選取圖形使用者介面上的功能表項目。	[檔案] → [新增] → [範本]	從 [檔案] 功能表中，選擇 [新增]。從 [新增] 子功能表中，選擇 [範本]。

## 文件、支援和培訓

Sun 網站提供有關以下附加資源的資訊：

- 文件 (<http://www.sun.com/documentation/>)
- 支援 (<http://www.sun.com/support/>)
- 培訓 (<http://www.sun.com/training/>)

## 搜尋 Sun 產品文件

除使用 docs.sun.com<sup>SM</sup> 網站搜尋 Sun 產品文件外，您還可以在搜尋欄位中鍵入以下語法來使用搜尋引擎：

```
search-term site:docs.sun.com
```

例如，若要搜尋「broker」，請鍵入以下語法：

```
broker site:docs.sun.com
```

若要使您的搜尋結果中包含其他 Sun 網站 (例如，[java.sun.com](http://java.sun.com)、[www.sun.com](http://www.sun.com) 和 [developers.sun.com](http://developers.sun.com))，請在搜尋欄位中使用 sun.com 代替 docs.sun.com。

## 協力廠商網站參照

本文件中提供了協力廠商 URL 以供參考，另亦提供其他相關的資訊。

---

**備註** – Sun 對本文件中提到的協力廠商網站的可用性不承擔任何責任。對於此類網站或資源中的 (或透過它們所取得的) 任何內容、廣告、產品或其他材料，Sun 並不表示認可，也不承擔任何責任。對於因使用或依靠此類網站或資源中的 (或透過它們所取得的) 任何內容、產品或服務而造成的、名義上造成的或連帶產生的任何實際或名義上之損壞或損失，Sun 概不負責，也不承擔任何責任。

---

## Sun 歡迎您提出寶貴意見

Sun 致力於提高文件品質，因此誠心歡迎您提出意見與建議。若要分享您的意見，請至 <http://docs.sun.com>，並按一下 [Send Comments (傳送您的意見)]。在線上表單中，請提供完整的文件標題和文件號碼。文件號碼是一個七位或九位的數字，可以在書的標題頁面或文件的 URL 中找到。例如，本書的文件號碼為 820-0538。

# 配置和設定 Service Registry

---

本章說明如何在安裝 Service Registry 之後對其進行配置，以及如何執行其他管理作業。

本章包含下列小節：

- 第 13 頁的「配置 Service Registry」
- 第 23 頁的「針對 Registry 網域配置 Java 虛擬機器 (JVM)」
- 第 24 頁的「建立管理員」
- 第 25 頁的「指定可執行使用者註冊的人」
- 第 26 頁的「啓用登錄 (Registry) 物件的版本控制」
- 第 27 頁的「停用 WSDL Cataloger」
- 第 28 頁的「配置 Web Console」
- 第 34 頁的「重新安裝 Service Registry」
- 第 35 頁的「管理 Java DB 資料庫」

## 配置 Service Registry

「Sun Java Enterprise System 5 Installation Guide for UNIX」說明如何使用 Registry 的預設特性設定，執行 Service Registry 的後續安裝配置。若要使用自訂特性設定，請複製 `ServiceRegistry-base/install/install.properties` 或 `ServiceRegistry-base/install/install.properties.template` 檔案，並且在執行配置之前先編輯此檔案。

若要配置 Registry，您可使用 root、超級使用者或非 root 使用者的身份登入。

基於安全性因素，建議您以非 root 使用者的身份配置 Registry。如需指示，請參閱第 17 頁的「在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry」。

`ServiceRegistry-base` 位於 Solaris 作業系統上的 `/opt/SUNWsrvc-registry`，若是 Linux 和 HP-UX 系統，則位於 `/opt/sun/srvc-registry`。

**備註** – 在配置 Service Registry 之前，您必須首先安裝和配置 Sun Java System Application Server (「Application Server」)。Service Registry 的配置程序會將 Registry 安裝至 Application Server 網域中。

建議您將 Application Server 安裝在其預設位置。如果您已將 Application Server 安裝在非預設位置，請在配置 Service Registry 之前依照第 22 頁的「[配置非預設 Application Server 安裝的 Service Registry](#)」中的說明進行操作。

`install.properties` 檔案包含一組可修改的特性設定。配置程序會使用表 1-1 中所列的特性。每個特性名稱都具有前綴 `registry.install.` (以點號結尾)。某些特性都會為 Application Server 網域 (為 Registry 所建立) 設定非預設連接埠。本手冊另有章節說明其他設定的配置特性。

您可以使用 root 或非 root 使用者的身份，設定多個 Service Registry 實例。如果您這麼做，您必須將 `registry.install.dataHome` 特性變更為各個 Registry 實例的唯一特性。如果您要在伺服器上同時執行一個以上的實例，您也必須將八個連接埠特性變更為各個實例唯一的特性。對於多個非 root 實例，您最好將各個實例的 `registry.install.CACertDir` 值都變更成唯一。

表 1-1 Service Registry 配置特性

屬性名稱	說明	預設特性值
DomainName	Application Server 網域名稱	registry
ServerInstanceHost	預設主機名稱，您可在此存取 Service Registry	localhost
ServerInstancePort	Service Registry 的 Application Server HTTP 連接埠	6480
ServerInstanceSecurePort	Service Registry 的 Application Server HTTPS 連接埠	6443
ServerJMSPort	Service Registry 的 Application Server Message Queue 連接埠	6484
ServerIIOPPort	Service Registry 的 Application Server IIOP 連接埠	6485

表 1-1 Service Registry 配置特性 (續)

屬性名稱	說明	預設特性值
ServerIIOPSecurePort	Service Registry 的 Application Server IIOP 安全連接埠	6486
ServerIIOPMutualAuthPort	Service Registry 的 Application Server IIOP 相互認證連接埠	6487
AdministrationJMXPort	Service Registry 的 Application Server JMX 連接埠	6488
AdministrationPort	Service Registry 的 Application Server 管理伺服器連接埠	6489
AdministratorUserID	用於存取 Application Server Administrative Server 的使用者名稱	admin
AdministratorPassword	用於存取 Application Server Administrative Server 的密碼	若您使用 root 的身份進行配置，則密碼為 12345678 如果您使用非 root 使用者的身份進行配置，則無密碼
ApplicationServerKeystorePassword	用於存取 Application Server 金鑰庫的密碼	若您使用 root 的身份進行配置，則密碼為 12345678 如果您使用非 root 使用者的身份進行配置，則無密碼
RegistryServerKeystorePassword	用於存取 Service Registry 金鑰庫的密碼	若您使用 root 的身份進行配置，則密碼為 12345678 如果您使用非 root 使用者的身份進行配置，則無密碼
clientDatabase	決定 Service Registry 資料庫是在內嵌式模式中執行，還是網路伺服器模式中執行 (對於「網路伺服器」模式，其值為 true)	false
RequireDatabaseAuthentication	決定存取 Service Registry 資料庫是否需要使用者認證	false

表 1-1 Service Registry 配置特性 (續)

屬性名稱	說明	預設特性值
DatabaseUserID	存取 Service Registry 資料庫的使用者 ID (如果需要使用者認證)	APP
DatabasePassword	存取 Service Registry 資料庫的密碼 (如果需要使用者認證)	若您使用 root 的身份進行配置，則密碼為 app123 如果您使用非 root 使用者的身份進行配置，則無密碼
backupDir	用於 Service Registry 備份的目錄。通常會標記為註釋；移除註釋可指定替代位置。	若您使用 root 的身份進行配置，則目錄為 <i>RegistryDomain-base/3.0/backup</i> 如果您以非 root 使用者的身份進行配置，則目錄為 <i>\$HOME/srvc-registry/3.0/backup</i>
dataHome	儲存 Service Registry 資料的目錄	若您使用 root 的身份進行配置，則目錄為 <i>RegistryDomain-base</i> 如果您以非 root 使用者的身份進行配置，則目錄為 <i>\$HOME/srvc-registry</i>
CACertDir	Application Server 網域應信任的新增憑證所在目錄	若您使用 root 的身份進行配置，則目錄為 <i>ServiceRegistry-base/install/cacerts</i> 如果您以非 root 使用者的身份進行配置，則目錄為 <i>\$HOME/srvc-registry/cacerts</i>

## ▼ 在 [以後配置] 方式安裝之後以 Root 身份使用自訂特性配置 Service Registry

**開始之前** 這些指示假設您已經以 root 身份登入或已經成為超級使用者。

- 1 變更至 *ServiceRegistry-base/install* 目錄。
- 2 將 *install.properties* 檔案複製到安全的位置。

使用指令，如下所示：

```
cp install.properties $HOME/hidden_dir/sr.properties
```



- 3 變更 `properties` 檔案的權限，使該檔案成為可寫入形式。

使用指令，如下所示：

```
chmod 600 $HOME/hidden_dir/sr.properties
```

- 4 在檔案中編輯可修改的特性。

例如，建議您變更預設值中所有的密碼。

- 5 將權限改回原始的唯讀值。

使用指令，如下所示：

```
chmod 400 $HOME/hidden_dir/sr.properties
```

- 6 在 `ServiceRegistry-base/install` 目錄中，執行下列指令 (全部位於同一行)，指定已修改的 `install.properties` 檔案位置。

使用指令，如下所示：

```
Ant-base /ant -f build-install.xml
-Dinstall.properties=$HOME/hidden_dir/sr.properties configure
```

`ant` 指令需要設定 `JAVA_HOME` 環境變數。通常，您會將此變數設定為下列的值：

```
/usr/jdk/entsys-j2se
```

Registry 配置程序會在 `RegistryDomain-base/domains/${registry.install.DomainName}` 中建立 Application Server 網域。預設網域名稱為 `registry`。然後，配置程序會啟動網域，部署 Registry，並使網域處於執行狀態。

Registry 配置程序會將 Registry 資料庫和伺服器金鑰庫安裝在目錄 `RegistryDomain-base/3.0` 中。解除安裝 Registry 時不會移除此目錄，以便保留資料庫以用於將來的發行版本。管理員可以控制何時移除此目錄和是否移除此目錄。

`RegistryDomain-base` 位於 Solaris 作業系統上的 `/var/opt/SUNWsrcv-registry`，若是 Linux 和 HP-UX 系統，則位於 `/var/opt/sun/srcv-registry`。

Registry 配置程序會建立名稱為 `ServiceRegistry-base/install/cacerts` 的目錄，供您放置 Application Server 網域應信任的新增憑證。

- 7 檢查 `ant configure` 指令的輸出以查看是否有錯誤。

如果沒有錯誤，您可以立即開始使用 Web Console 或 Admin Tool。

## ▼ 在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry

**開始之前** 這些指示假設您是以一般使用者身份登入，而不是以 `root` 身份，而且已經有人以 `root` 身份登入並安裝了 Service Registry。

- 1 變更至 `ServiceRegistry-base/install` 目錄。
- 2 將 `install.properties.template` 檔案複製到主目錄中的某個位置，並且將該檔案重新命名。  
使用指令，如下所示：  

```
cp install.properties.template $HOME/sr.properties
```
- 3 在文字編輯器中開啓檔案副本，並且依您所需修改特性。  
您必須提供下列特性的密碼值，這些值目前在範本檔案中保留為空白。  

```
registry.install.AdministratorPassword=  
registry.install.ApplicationServerKeystorePassword=  
registry.install.RegistryServerKeystorePassword=
```
- 4 變更檔案的權限，設定為只有您可讀取該檔案。  
執行以下類似指令：  

```
chmod 400 $HOME/sr.properties
```
- 5 在 `ServiceRegistry-base/install` 目錄中，執行以下類似指令(全部位於同一行)，以指定已修改檔案的位置。  

```
Ant-base/ant -f build-install.xml -Dinstall.properties=$HOME/sr.properties  
configure
```

使用類似的指令，執行您所需的其他任何配置目標。

Registry 配置程序會在 `$HOME/srvc-registry/domains/${registry.install.DomainName}` 中建立 Application Server 網域。預設網域名稱為 `registry`。

Registry 配置程序會將 Registry 資料庫和伺服器金鑰庫安裝在目錄 `$HOME/srvc-registry/3.0` 中。

Registry 配置程序會建立名稱為 `$HOME/srvc-registry/cacerts` 的目錄，供您放置 Application Server 網域應信任的新增憑證。
- 6 檢查 `ant configure` 指令的輸出以查看是否有錯誤。  
如果沒有錯誤，您可以立即開始使用 Web Console 或 Admin Tool。

## 管理 Service Registry 的 Application Server 網域

依預設，Service Registry 的配置程序會建立名稱為 `registry` 的 Application Server 網域，Service Registry Web 應用程式將被部署至該網域。此網域位於 `RegistryDomain-base/domains/registry` 目錄中。

此位置與 Application Server 網域的預設位置不同，預設位置為  
 /var/opt/SUNWappserver/domains (Solaris 作業系統) 或  
 /var/opt/sun/appserver/domains (Linux 和 HP-UX 系統)。

---

**備註** – 建議您不要在 registry 網域中執行 Service Registry 以外的任何應用程式。

---

若要管理 registry 網域，您可以使用 Application Server Administration Console (「Admin Console」)。您可以使用 Admin Console 啟動和停止網域、檢視伺服器記錄以及執行其他管理作業。請參閱第 20 頁的「使用 Application Server Admin Console」，以取得詳細資訊。

您也可以直接檢查伺服器記錄。記錄位於  
*RegistryDomain-base/domains/registry/logs/server.log* 檔案中。

除了 Admin Console 之外，您可以使用 `asadmin` 指令管理 registry 網域。由於網域未在預設位置，所以當您使用提供 `--domaindir` 選項的 `asadmin` 指令時，必須指定該選項。`--domaindir` 選項的引數是 *RegistryDomain-base/domains*。

如果您要使用 `asadmin` 指令的 `--passwordfile` 選項，您需要有 Registry 網域管理員密碼副本的檔案。若要建立這類檔案，請使用 `build-install.xml` 檔案的 `generate.password.file` 目標。該檔案是  
*RegistryDomain-base/3.0/data/security/pw.txt*。

registry 網域使用一組非預設連接埠，以免與預設 Application Server 網域 `domain1` 發生衝突。這些 Service Registry 連接埠值已向 Internet Assigned Numbers Authority (IANA) 註冊。表 1-2 列出並說明了這些連接埠。如需更多資訊，請參閱「Sun Java System Application Server Enterprise Edition 8.2 Administration Guide」中的「Ports in the Application Server」。

表 1-2 Service Registry 網域預設連接埠

連接埠值	說明
6480	HTTP 連接埠
6443	經由 SSL 的 HTTPS
6484	Message Queue 連接埠
6485	IIOP 連接埠
6486	IIOP SSL 連接埠
6487	IIOP 相互認證連接埠
6488	JMX 連接埠

表 1-2 Service Registry 網域預設連接埠 (續)

連接埠值	說明
6489	Application Server 網域管理連接埠

## ▼ 使用 Application Server Admin Console

- 1 在 Web 瀏覽器中，移至 URL `https://hostname:6489/`。  
*hostname* 是執行 Application Server 和 Service Registry 的系統。
- 2 接受提供的憑證。  
螢幕上顯示登入頁面。
- 3 在登入頁面上，在 [使用者名稱] 欄位中鍵入 `admin`。  
配置 Registry 時，如果您曾經變更 `registry.install.AdministratorUserID` 特性的預設值，請鍵入您先前指定的值。
- 4 在 [密碼] 欄位中鍵入 Application Server 管理員密碼。使用您在配置 Registry 時為 `registry.install.AdministratorPassword` 特性指定的值。預設為 `12345678`。
- 5 按一下 [登入]。

另請參閱 如需有關使用 Admin Console 的詳細資訊，請參閱 Admin Console 的線上說明或「Sun Java System Application Server Enterprise Edition 8.2 Administration Guide」。

## 變更 Service Registry 記錄層級

若要變更 Service Registry 的記錄層級，請按照 Admin Console 線上說明中的指示作業。要在 [其他特性] 區域中指定的特性是 `org.freebxml.omar`。

若要變更特定 Service Registry 子元件的記錄功能，請參閱下列檔

案：`RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes/log4j.properties`。您可以指定此檔案中所列的任何子元件名稱。請勿指定字串 `log4j.logger`。例如，您可以經由指定 `org.freebxml.omar.server`，以記錄伺服器呼叫。

## ▼ 停止並重新啟動 Registry 的 Application Server 網域

Registry 的配置程序會啟動部署 Registry 的 Application Server 網域。執行某些管理作業後，您需要停止網域並將其重新啟動。這類作業的範例包括第 23 頁的「針對 Registry 網域配置 Java 虛擬機器 (JVM)」和第 24 頁的「建立管理員」。

如果您需要重新啟動網域，Admin Console 會通知您。您可以使用 Admin Console 執行此作業。如果您使用的是 `asadmin` 指令，則可使用 Ant 作業停止和啟動網域。

- 1 變更為 Service Registry 安裝目錄。

```
cd ServiceRegistry-base/install
```

- 2 執行以下指令 (全部位於同一行)：

```
Ant-base/ant -f build-install.xml -Dinstall.properties= props-file  
appserver.domain.bounce
```

其中 *props-file* 是您在第 16 頁的「在 [以後配置] 方式安裝之後以 Root 身份使用自訂特性配置 Service Registry」或第 17 頁的「在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry」中所編輯 `install.properties` 檔案副本的路徑名稱。

`appserver.domain.bounce` 目標會停止網域，然後將它重新啟動。

`build-install.xml` 檔案也包含分別用於停止和啟動 Registry 網域的 Ant 目標。若要停止網域，請使用 Ant 目標 `appserver.domain.stop`。若要啟動網域，請使用 Ant 目標 `appserver.domain.start`。

## ▼ 將根憑證增加至 Registry 網域中的信任的憑證

此作業可延伸 Application Server registry 網域中信任的憑證清單。

只有在您使用協力廠商憑證，而且協力廠商的根憑證授權單位 (CA) 憑證尚未在 Application Server 信任清單存放區時，才執行此作業。如果您只有使用 registry 發出的憑證，請勿執行此作業。

若要判斷您需要的 CA 憑證是否已可供使用，您可以使用 `build-install.xml` 檔案的 `list.cacerts` 目標：

```
Ant-base/ant -f build-install.xml -Dinstall.properties=props-file list.cacerts
```

其中 *props-file* 是您在第 16 頁的「在 [以後配置] 方式安裝之後以 Root 身份使用自訂特性配置 Service Registry」或第 17 頁的「在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry」中所編輯 `install.properties` 檔案副本的路徑名稱。

- 1 下載您要支援的所有根憑證。

提供根憑證的網站包含：

- <http://www.entrust.net/developer/>
- [http://www.geotrust.com/resources/root\\_certificates/](http://www.geotrust.com/resources/root_certificates/)
- <http://www.thawte.com/roots/>
- <http://www.verisign.com/support/roots.html>

- 2 如有必要，使用 `unzip` 指令從下載歸檔擷取 `.cer` 檔案。

---

備註 – 某些檔案有 `.der` 後綴。

---

- 將 `.cer` 檔案複製到以 `install.properties` 檔案副本中的 `registry.install.CACertDir` 特性指定的目錄中。

如果您以 `root` 的身份配置，此值通常是 `ServiceRegistry-base/install/cacerts`；如果您以非 `root` 使用者的身份配置，則為 `$HOME/srvc-registry/cacerts`。

- 變更至目錄 `ServiceRegistry-base/install`。

- 執行以下指令(全部位於同一行)：

```
Ant-base/ant -f build-install.xml -Dinstall.properties= props-file install.cacerts
```

此指令會將 `registry.install.CACertDir` 特性指定的目錄中所找到的任何憑證，安裝至 Application Server 網域信任清單存放區。

您可以再次使用 `list.cacerts` 目標，以確定已正確安裝憑證。

- 依照第 20 頁的「[停止並重新啟動 Registry 的 Application Server 網域](#)」中的說明進行操作。

## 配置非預設 Application Server 安裝的 Service Registry

安裝 Application Server 的預設位置是 `/opt/SUNWappserver/appserver` (在 Solaris 作業系統上) 和 `/opt/sun/appserver` (在 Linux 和 HP-UX 系統上)。如果您已將 Application Server 安裝在其他位置上，則在配置 Service Registry 之前，必須先編輯一份 `install.properties` 檔案。

### ▼ 編輯一份 `install.properties` 檔案的副本

- 變更至 `ServiceRegistry-base/install` 目錄。
- 將 `install.properties` 檔案複製到安全的位置，並將它重新命名。

使用指令，如下所示：

```
cp install.properties $HOME/hidden_dir/sr.properties
```

- 變更 `properties` 檔案的權限，使檔案成為可寫入形式。

使用類似以下的指令：

```
chmod 600 $HOME/hidden_dir/sr.properties
```

- 在文字編輯器中開啓檔案。

- 5 尋找將 `appserver.root.dir` 特性標記為註釋的定義。
- 6 移除註釋字元 (`#`)，並使用 Application Server 的實際位置替代特性定義。
- 7 儲存並關閉檔案。

接下來的步驟 繼續依照第 13 頁的「配置 Service Registry」中的說明進行操作。

## 針對 Registry 網域配置 Java 虛擬機器 (JVM)

Service Registry 需要下列條件，才能正常運作：

- Registry 必須能夠存取外部網站
- Registry 的 Application Server 網域必須有足夠的記憶體可供使用

任何登錄 (Registry) 物件均可具有外部連結 (ExternalLink) 物件，該物件可指定與該登錄 (Registry) 物件關聯的外部 URL。任何服務 (Service) 物件都可具有服務連結 (ServiceBinding)，這也可指向外部 URL。為了讓使用者建立外部連結 (ExternalLink) 和服務連結 (ServiceBinding) 物件，Service Registry 必須能夠驗證 URL，而且此作業需要存取外部網站。如果 Registry 是部署在防火牆後端，您需要設定允許此存取權的代理伺服器配置。

代理伺服器配置需要您將 Web 代理主機和連接埠，指定為已部署 Service Registry 之 Application Server 網域的 Java 虛擬機器 (JVM) 選項。

Registry 也可能記憶體不足。若要避免發生這種問題，請配置 JVM 選項，增加 Application Server 網域之可用記憶體。

執行下列作業，配置 Registry 的 JVM 選項。

### ▼ 配置 Registry 網域的 JVM 選項

- 1 依照第 20 頁的「使用 Application Server Admin Console」中的說明，登入至 Application Server Admin Console。
- 2 展開配置節點。
- 3 展開伺服器節點 `server-config` (Admin Config)。
- 4 按一下 [JVM 設定]。
- 5 按一下 [JVM 選項] 標籤。
- 6 按一下 [增加 JVM 選項]。

- 7 在文字欄位中，鍵入下列指令(全部位於同一行)：

```
-Dhttp.proxyHost=hostname.domainname -Dhttp.proxyPort=8080 -Dhttp.nonProxyHosts=localhost
```

連接埠值通常為 8080。如果您位置的連接埠不同，請指定正確的值。

- 8 再按一下 [增加 JVM 選項]。

- 9 在文字欄位中，鍵入下列指令(全部位於同一行)：

```
-XX:MaxPermSize=128m
```

- 10 按一下 [儲存]。

- 11 依照第 20 頁的「[停止並重新啟動 Registry 的 Application Server 網域](#)」中的說明進行操作。

## 建立管理員

在 Service Registry Administration Tool 中有一些作業，僅有註冊為管理員的使用者才能執行。此外，可能會呼叫管理員對其他使用者提交的物件執行生命週期變更(例如，批准)。

管理員也可以變更預設存取控制策略(ACP)。然而，ACP 目前必須手動編寫，因此需要有關 OASIS 可延伸存取控制標記語言(XACML)的知識。如需詳細資訊，請參閱 ebXML RIM 3.0 的第 9 章「Access Control Information Model」，尤其是 9.7.6 小節至 9.7.8 小節中的範例。請參閱第 7 頁的「[閱讀本書之前](#)」，以獲得有關如何尋找 ebXML RIM 3.0 規格的資訊。

### ▼ 建立管理員

若要將自己註冊為管理員，請執行以下步驟：

- 1 執行「[Service Registry 3.1 使用者指南](#)」中的「[建立使用者帳號](#)」中所述的使用者註冊程序，或按照第 48 頁的「[add user](#)」所述，使用 Admin Tool 的 add user，將自己加入成為使用者。
- 2 如果您曾經使用 Web Console 註冊，請取得使用者 (User) 物件的唯一識別碼，如下所示：
  - a. 使用 Web Console 執行基本查詢，在此將 [物件類型] 設定為 [使用者 (User)]。
  - b. 按一下 [詳細資訊] 連結，檢視 Registry 為您建立的使用者 (User) 物件。
  - c. 記下 [唯一識別碼] 欄位的值，或複製該值然後貼入檔案中。



如果您曾經使用 `add user` 指令，請使用 `users` 指令取得使用者的清單，然後複製自己的使用者名稱的識別碼值。

- 3 變更至目錄 `RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes`。
- 4 在文字編輯器中開啓檔案 `omar.properties`。
- 5 尋找特性 `omar.security.authorization.registryAdministrators` 的定義。
- 6 加入垂直列 (`|`)，後面再加上在步驟 2 中複製的唯一識別碼字串，以編輯特性定義。特性定義必須全部位於同一行，且不得包含空格。完成之後，結果會如下所示 (全部位於同一行)：
 

```
omar.security.authorization.registryAdministrators=
urn:freebxml:registry:predefinedusers:registryoperator|
urn:uuid:77f5c196-79de-4286-8483-8d80def3583b
```
- 7 儲存並關閉 `omar.properties` 檔案。
- 8 依照第 20 頁的「[停止並重新啓動 Registry 的 Application Server 網域](#)」中的說明進行操作。

**接下來的步驟** 若要建立額外的管理員，您不必編輯 `omar.properties` 檔案。您可以使用 Admin Tool 或 Web Console 增加使用者，並且可以使用 Web Console 將使用者分類為管理員。

## 指定可執行使用者註冊的人

依預設，可存取 Service Registry URL 的使用者均能以使用者註冊精靈，執行使用者註冊。任何註冊的使用者都可將內容發佈到登錄 (Registry)。

您可以定義 `omar.properties` 檔案中的 `omar.security.selfRegistration.acl` 特性，以限制此功能，如此只有特定人員可執行使用者註冊。

在您完成此作業之後，唯一可使用精靈註冊的人員，會是由特性定義指定姓名的人員。其他嘗試註冊的人員會在填寫使用者認證詳細資訊表單 (使用者註冊精靈的步驟 3)，並且按 [下一步] 按鈕之後，看見錯誤訊息。錯誤訊息顯示 `User registration failed`。該訊息之後有一行文字顯示 `UserNotFoundException`。

如果 `omar.security.selfRegistration.acl` 特性未出現在 `omar.properties` 中，或者該特性已定義為空字串，則任何註冊過的使用者都可以將內容發佈到登錄 (Registry)。

## ▼ 限制使用者註冊

### 1 變更至目錄

*RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes*。

### 2 在文字編輯器中開啓檔案 `omar.properties`。

### 3 增加下列特性的定義：

```
omar.security.selfRegistration.acl
```

您可以將此特性定義放在檔案中的任何位置。不過特性定義的位置，比較適合具有 `omar.security` 前綴的其他特性定義所在區域。

將特性值定義成爲以逗號分隔的人員姓名清單，這些人員都經過授權可執行使用者註冊，如以下範例所示：

```
omar.security.selfRegistration.acl=Vijay Patel, Jane Doe,
```

### 4 儲存並關閉 `omar.properties` 檔案。

### 5 依照第 20 頁的「停止並重新啓動 Registry 的 Application Server 網域」中的說明進行操作。

## 啓用登錄 (Registry) 物件的版本控制

依預設，登錄 (Registry) 物件的版本控制處於停用狀態。如果啓用版本控制，只要任何物件屬性變更，就會建立新的物件版本。啓用版本控制是一種管理作業。

## ▼ 啓用 Registry 物件的版本控制

### 1 變更至目錄

*RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes*。

### 2 在文字編輯器中開啓檔案 `omar.properties`。

### 3 找到特性 `omar.server.lcm.VersionManager.versionableClassList` 的定義。

依預設，此特性不含任何值：

```
omar.server.lcm.VersionManager.versionableClassList=
```

- 4 指定您需要 Registry 在物件經過修改之後即建立新版本的任何物件。以垂直列 (|) 分隔物件。  
例如：  
`omar.server.lcm.VersionManager.versionableClassList=Service|Organization`  
使用檔案中標記為註釋的特性設定副本做為範例。
- 5 儲存並關閉 `omar.properties` 檔案。
- 6 依照第 20 頁的「[停止並重新啟動 Registry 的 Application Server 網域](#)」中的說明進行操作。

## 停用 WSDL Cataloger

依預設，當 WSDL 檔案發佈至登錄 (Registry) 成為外部物件 (ExtrinsicObject) 物件時，Registry 會使用 [ebXML Registry Profile for Web Services](#) 所定義的功能，建立物件的中介資料。您可以停用此功能。

### ▼ 停用 WSDL Cataloger

- 1 變更至目錄  
`RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes`。
- 2 在文字編輯器中開啓檔案 `omar.properties`。
- 3 找出特性  
`omar.server.cms.classMap.urn\:oasis\:names\:tc\:ebxml-regrep\:profiles\:ws\:wsdl\[:cataloging\:Service\:default` 的定義。
- 4 在定義的前面放置井字號 (#)，可將此特性定義標記為註釋。
- 5 儲存並關閉 `omar.properties` 檔案。
- 6 依照第 20 頁的「[停止並重新啟動 Registry 的 Application Server 網域](#)」中的說明進行操作。

# 配置 Web Console

身為管理員，您可以編輯配置檔案，自訂某些部分的 Web Console 顯示畫面。

本小節說明以下作業：

- 第 28 頁的「增加預先定義的查詢」
- 第 30 頁的「變更預設查詢」
- 第 30 頁的「隱藏分類方案」
- 第 31 頁的「配置搜尋結果顯示」

如需有關使用 Web Console 的資訊，請參閱「Service Registry 3.1 使用者指南」。

## 增加預先定義的查詢

Service Registry 包含一些預先定義的查詢，這些查詢顯示在 [Web Console 搜尋] 表單中的 [選取預先定義的查詢] 下拉式清單中。身為管理員，您可以將新的查詢增加至 Registry 安裝的特定下拉式清單中。

### ▼ 增加預先定義的查詢

- 1 使用 Web Console 將即時查詢 (AdhocQuery) 物件發佈至登錄 (Registry)。

針對查詢指定的名稱和說明會出現在預先定義查詢的下拉式清單中。在查詢的 SQL 敘述中，指定使用者所提供資料的預留位置，方法是將這些資料括在一組單引號內，如下所示：

```
select * from registryobject where id = '$lid'
```

- 2 記下即時查詢 (AdhocQuery) 物件和 SQL 敘述中任何預留位置的唯一識別碼，或是將這些識別碼複製並貼入到檔案中。
- 3 變更至目錄 *RegistryDomain-base/3.0/jaxr-ebxml*。
- 4 在文字編輯器中，開啓檔案 *registry-browser-config.xml*。  
製作原始檔案的副本，以便在必要時回復所做的變更。
- 5 使用下列格式，在 *registry-browser-config.xml* 檔案中增加項目。在 SQL 敘述中，針對各個預留位置指定 *Parameter* 元素。

```
<Query>
  <AdhocQueryRef id="unique-identifier"/>
  <Parameter parameterName="$placeholder-name" datatype="string">
    <rim:Name>
      <rim:LocalizedString xml:lang="en" charset="UTF-8"
        value="parameter-name-in-en-locale"/>
    </rim:Name>
  </Parameter>
</Query>
```

```

    <rim:LocalizedString xml:lang="fr" charset="UTF-8"
      value="parameter-name-in-fr-locale"/>
  </rim:Name>
  <rim:Description>
    <rim:LocalizedString xml:lang="en" charset="UTF-8"
      value="parameter-description-in-en-locale"/>
    <rim:LocalizedString xml:lang="fr" charset="UTF-8"
      value="parameter-description-in-fr-locale"/>
  </rim:Description>
</Parameter>
...
</Query>

```

*unique-identifier* 是即時查詢 (AdhocQuery) 物件的唯一識別碼。

各個參數的 *parameterName* 屬性值必須來自查詢的 SQL 敘述預留位置。

*datatype* 屬性可含有下列任何值：

- *string*：此參數會顯示為 [搜尋] 表單中的文字欄位。
- *taxonomyElement*：此參數會顯示為 [搜尋] 表單中的下拉式清單。如果您指定 *taxonomyElement* 資料類型，「名稱」和「說明」元素後面必須接著 *SlotList* 元素，如下所示：

```

<rim:SlotList>
  <rim:Slot name="domain">
    <rim:ValueList>
      <rim:Value>
        classification-scheme-or-concept-id
      </rim:Value>
    </rim:ValueList>
  </rim:Slot>
</rim:SlotList>

```

*classification-scheme-or-concept-id* 是分類方案或概念的唯一識別碼，其中的概念 (或子概念) 必須出現在下拉式清單中。如果分類方案未出現在登錄 (Registry) 中，您必須發佈分類方案。

槽名稱必須是「domain」。

- *boolean*：此參數會顯示為 [搜尋] 表單中的核取方塊。

如果 *datatype* 是 *string* 或 *boolean*，您也可以將 *defaultValue* 屬性增加至 *Parameter* 元素中，以指定要出現在 [搜尋] 表單中的預設值。

針對您支援的任何語言環境，指定各個參數名稱和描述的本土化字串值。目前語言環境的 *parameter-name* 會在 [搜尋] 表單中顯示為參數標籤。

使用 *registry-browser-config.xml* 檔案中的現有項目做為參照。

## 6 儲存並關閉 *registry-browser-config.xml* 檔案。

- 7 依照第 20 頁的「[停止並重新啟動 Registry 的 Application Server 網域](#)」中的說明進行操作。

## 變更預設查詢

[選取預先定義的查詢] 下拉式清單中出現的預設查詢是「基本查詢」，這可供使用者按名稱、說明和分類來搜尋登錄 (Registry) 物件。

身為管理員，您可以將此預設值變更為安裝環境適用的查詢。例如，您可能希望預設查詢是已加入 Registry 的新預先定義查詢，如第 28 頁的「[增加預先定義的查詢](#)」中所述。若要進行變更，請編輯配置檔案的特性。

### ▼ 變更預設查詢

- 1 變更至目錄

`RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes`。

- 2 在文字編輯器中開啓檔案 `jaxr-ebxml.properties`。

- 3 找出特性 `jaxr-ebxml.thin.defaultQueryPanel` 的定義。依預設，此特性已標記為註釋：

```
#jaxr-ebxml.thin.defaultQueryPanel=
```

- 4 移除註釋字元 (#)。

- 5 經由指定將做為預設值的查詢邏輯識別碼，來設定特性值，如以下範例所示：

```
jaxr-ebxml.thin.defaultQueryPanel=urn:oasis:names:tc:ebxml-regrep:query:MyQuery
```

- 6 儲存並關閉 `jaxr-ebxml.properties` 檔案。

- 7 依照第 20 頁的「[停止並重新啟動 Registry 的 Application Server 網域](#)」中的說明進行操作。

## 隱藏分類方案

分類方案的樹狀結構會出現在下列的 Web Console 區域：

- 當您按一下 [搜尋] 功能表區域中的 [選擇分類節點 (Classification Node)]，或需要選取某些登錄 (Registry) 物件種類的概念時，便會出現 [分類方案 (ClassificationSchemeor)/概念選擇器] 視窗
- [檔案管理] 功能表區域

身為管理員，如果您不想讓 Service Registry 的使用者使用分類方案，可將分類方案隱藏不顯示。若要隱藏分類方案，請在配置檔案中定義特性。

## ▼ 隱藏分類方案

### 1 變更至目錄

*RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes*。

### 2 在文字編輯器中開啓檔案 *jaxr-ebxml.properties*。

### 3 使用下列語法，設定特性

*jaxr-ebxml.registryBrowser.ConceptsTreeModel.hiddenSchemesList*。所有的特性定義必須全部位於同一行，且不得包含空格。

```
jaxr-ebxml.registryBrowser.ConceptsTreeModel.hiddenSchemesList=
class-scheme-id1|class-scheme-id2|...
```

指定要隱藏的各個分類方案的邏輯識別碼。如果您指定一個以上的識別碼，請使用垂直列 (|) 分隔各個識別碼，如以下範例所示：

```
jaxr-ebxml.registryBrowser.ConceptsTreeModel.hiddenSchemesList=
urn:oasis:names:tc:ebxml-regrep:classificationScheme:StatusType|
urn:oasis:names:tc:ebxml-regrep:profile:ws:classificationScheme:BindingType
```

### 4 儲存並關閉 *jaxr-ebxml.properties* 檔案。

### 5 依照第 20 頁的「停止並重新啓動 Registry 的 Application Server 網域」中的說明進行操作。

## 配置搜尋結果顯示

依預設，Web Console 會針對各個查詢一次顯示 25 筆搜尋結果。如果搜尋後傳回 25 筆以上的結果，使用者可顯示額外的結果頁面。身為管理員，您可以修改各頁面上顯示的搜尋結果數量。

依預設，Web Console 會將某些欄顯示在搜尋結果區域中。對於各個物件，這時會顯示物件類型、名稱、說明、版本和版本註釋。對於某些物件類型，則配置非預設的顯示畫面。例如，對於服務連結 (ServiceBinding) 物件，顯示畫面包含的是端點，而非版本資訊。身為管理員，您可以增加配置資訊，以顯示您所選擇物件類型的非預設資料。

若要執行各個作業，需要編輯配置檔案。

## ▼ 配置搜尋結果顯示的列數

### 1 變更至目錄

*RegistryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes*。

### 2 在文字編輯器中，開啓檔案 *jaxr-ebxml.properties*。

### 3 尋找特性 *omar.client.thinbrowser.numSearchResults* 的定義：

```
omar.client.thinbrowser.numSearchResults=25
```

### 4 將值 25 變更為您所要的值。

### 5 儲存並關閉 *jaxr-ebxml.properties* 檔案。

### 6 依照第 20 頁的「停止並重新啓動 Registry 的 Application Server 網域」中的說明進行操作。

## ▼ 配置搜尋結果區域的欄

您可以針對物件類型，配置 [搜尋結果] 區域的欄。這些欄會顯示物件的屬性。

### 1 變更至目錄 *RegistryDomain-base/3.0/jaxr-ebxml*。

### 2 在文字編輯器中，開啓檔案 *registry-browser-config.xml*。

### 3 在 *registry-browser-config.xml* 檔案中增加項目，或編輯現有的項目。使用下列格式。

此範例將配置 Service 物件的非預設顯示。

```
<ObjectTypeConfig
  className="org.freeebxml.omar.client.xml.registry.infomodel.ServiceImpl"
  id="urn:oasis:names:tc:ebxml-regrep:ObjectType:RegistryObject:Service">
<SearchResultsConfig>
  <SearchResultsColumn columnClass="java.lang.Object"
    columnHeader="Object Type" columnWidth="25" editable="false"
    method="getObjectType"/>
  <SearchResultsColumn columnClass="java.lang.Object"
    columnHeader="Name" columnWidth="25" editable="true" method="getName"/>
  <SearchResultsColumn columnClass="java.lang.Object"
    columnHeader="Description" columnWidth="30" editable="true"
    method="getDescription"/>
  <SearchResultsColumn columnClass="java.lang.Object"
    columnHeader="Status" columnWidth="15" method="getStatusAsString"/>
  <SearchResultsColumn columnClass="java.lang.Object"
    columnHeader="Version" columnWidth="5" method="getVersionName"/>
```



```
</SearchResultsConfig>
</ObjectTypeConfig>
```

registry-browser-config.xml 檔案提供 ObjectTypeConfig 元素的語法。使用檔案中已有的元素做為範例。這些元素會配置登錄 (Registry) 物件的預設顯示，並配置外部連結 (ExternalLink)、外部物件 (ExtrinsicObject) 和服務連結 (ServiceBinding) 物件的非預設顯示。

您可配置的欄數上限是 30。

對於 SearchResultsColumn 元素：

- columnClass 屬性值一定是 java.lang.Object。
- columnHeader 屬性值是 Web Console 資源束檔案中訊息的金鑰。這些檔案位在目錄 *registryDomain-base/domains/registry/applications/j2ee-modules/soar/WEB-INF/classes/org/freebxml/omar/client/ui/thin/*。例如，如果您針對 columnHeader 值鍵入 Object Type，Web Console 的背景 Bean 會使用 WebResourceBundle 類別查找具有該金鑰的訊息。如果 WebResourceBundle 找不到訊息，會將金鑰轉換為小寫，然後再試一次。如果仍然找不到，它會將訊息值設定為 "???" + key + "???"，並且記錄關於遺失資源束項目的警告訊息。因此，若要增加新的本土化 columnHeader 值，您必須將新的訊息金鑰輸入至本目錄所包含的 ResourceBundle 檔案中。
- Web Console 未使用 columnWidth 屬性。
- Web Console 未使用 editable 屬性。
- 在大多數狀況下，您可以從「ebXML Registry Information Model Version 3.0」規格的類別屬性推演出 method 屬性的方法名稱 (如需詳細資訊，請參閱第 7 頁的「[閱讀本書之前](#)」)。`getStatusAsString` 方法可以在 RegistryObjectImpl 實作類別中找到。(不過，此發行版本的 Service Registry 不含 API 文件。)

各個 omar.client.xml.registry.infomodel 類別名稱只能有一個 ObjectTypeConfig 元素。

- 4 儲存並關閉 registry-browser-config.xml 檔案。
- 5 依照第 20 頁的「[停止並重新啟動 Registry 的 Application Server 網域](#)」中的說明進行操作。
- 6 若要驗證重新配置，請使用 Web Console 的 [搜尋] 或 [檔案管理] 功能表，顯示您已變更欄的物件。

## 重新安裝 Service Registry

如果您需要解除安裝再重新安裝 Service Registry，請在重新安裝之前執行以下作業：

- 如果 Registry 資料庫中含有您要保留的資料，請依照第 35 頁的「管理 Java DB 資料庫」中的說明備份資料庫。
- 停止 Registry 的 Application Server 網域，然後刪除該網域。如果不刪除網域，則重新安裝之 Registry 的安裝後配置將會失敗。

如果您需要重新安裝 Service Registry 資料庫 (例如，如果資料庫已毀壞)，請按照第 34 頁的「重新安裝 Service Registry 資料庫」中的指示進行操作。在重新安裝資料庫之前，您不需要解除安裝資料庫。

### ▼ 停止並刪除 Registry 的 Application Server 網域

- 1 變更至 *ServiceRegistry-base/install* 目錄。
- 2 執行以下指令 (全部位於同一行)：

```
Ant-base/ant -f build-install.xml Dinstall.properties= props-file  
appserver.domain.delete
```

其中 *props-file* 是您在第 16 頁的「在 [以後配置] 方式安裝之後以 Root 身份使用自訂特性配置 Service Registry」或第 17 頁的「在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry」中所編輯 *install.properties* 檔案副本的路徑名稱。

此目標會停止網域，然後再將其刪除。

### ▼ 重新安裝 Service Registry 資料庫

此作業會刪除現有的資料庫內容、納入已註冊的使用者，並且重新建立預設的資料庫。

- 1 變更至 *ServiceRegistry-base/install* 目錄。
- 2 執行以下指令 (全部位於同一行)：

```
Ant-base/ant -f build-install.xml Dinstall.properties= props-file install.db
```

- 3 執行以下指令 (全部位於同一行)：

```
Ant-base/ant -f build-install.xml Dinstall.properties= props-file  
appserver.domain.stop export.registryOperatorCert install.cacerts  
appserver.domain.start
```

# 管理 Java DB 資料庫

Registry 會使用 Java DB 資料庫。Java DB 是 Apache 軟體基金會開放原始碼關聯式資料庫計劃的商用發行版本。這個 Apache 計劃的名稱是 Derby。

依預設，資料庫位在 `RegistryDomain-base/3.0/data/registry/soar/` 目錄中。資料庫備份會位在 `RegistryDomain-base/3.0/backup/` 目錄中，其中包括一個含備份日期 (例如，`20060419-004759`) 的子目錄名稱。

依預設，Java DB 資料庫會在內嵌式模式中執行作業。這表示，它會與 Service Registry 在相同的 JVM 中執行，而且只接受來自 Service Registry 這個用戶端的連線。您無法進行遠端連線。當 Java DB 在內嵌式模式中執行時，只有在它未執行的狀況下，才可以備份資料庫 (離線備份)。如需操作說明，請參閱第 36 頁的「執行資料庫的離線備份」。

在 Registry 網域執行時，如果您需要能夠備份資料庫 (線上備份)，您必須在網路伺服器模式中執行 Java DB 資料庫。如需操作說明，請參閱第 37 頁的「從內嵌式模式切換至網路伺服器模式」和第 38 頁的「執行資料庫的線上備份」。

當 Java DB 在網路伺服器模式中執行時，Java DB 可在常見用戶端/伺服器配置中，接受多個用戶端連線。例如，Service Registry 和 SQL 用戶端可同時與 Java DB 進行通訊。當 Java DB 在網路伺服器模式中執行時，會使用預設值為 1527 的資料庫連接埠。Service Registry 這類的用戶端會使用此連接埠與資料庫通訊。

當 Java DB 在網路伺服器模式中執行時，Service Registry 會在網路用戶端模式中執行。

在網路伺服器模式中執行的資料庫必須以密碼加以保護，讓只有經過認證的用戶端才可以使用資料庫。當資料庫在內嵌式模式中執行時，您也可以使用相同的方式保護資料庫，但這並不是必要的步驟。

依預設，此時會設定 `ServiceRegistry-base/install/install.properties` 檔案中的特性，讓 Java DB 在內嵌式模式中執行。表 1-3 將顯示這些設定。

表 1-3 Java DB 的預設特性設定

特性設定	說明
<code>registry.install.clientDatabase=false</code>	啓用內嵌式模式
<code>registry.install.RequireDatabaseAuthentication=false</code>	不要求資料庫認證
<code>registry.install.DatabaseUserID=APP</code>	將使用者 ID 設定為 APP (未使用)
<code>registry.install.DatabasePassword=app123</code>	將密碼設定為 app123 (未使用)

如需有關 Java DB 的更多詳細資訊，請參閱 Java DB 文件，文件位在 [Java DB 網站](#)。

## ▼ 要求資料庫認證

依預設，不需要資料庫認證。當資料庫在內嵌式模式中執行時，可能會要求認證，而當資料庫在網路伺服器模式中執行時，則必須要求認證。

### 1 變更為 Service Registry 安裝目錄：

```
cd ServiceRegistry-base/install
```

### 2 在文字編輯器中，開啓 `install.properties` 檔案副本。

### 3 將 `registry.install.RequireDatabaseAuthentication` 特性的設定從 `false` 變更為 `true`。

### 4 編輯 `registry.install.DatabaseUserID` 特性的設定。

對於內嵌式模式，此值可以是 APP 或空白：

```
registry.install.DatabaseUserID=APP
```

```
registry.install.DatabaseUserID=
```

對於網路伺服器模式，此值必須是 APP。

### 5 編輯 `registry.install.DatabasePassword` 特性的設定。

密碼必須至少包含 6 個字元。預設值為 `app123`。

### 6 儲存並關閉檔案。

### 7 停止並重新啟動 Registry 的 Application Server 網域。若要這麼做，請執行以下指令(全部位於同一行)：

```
Ant-base/ant -f build-install.xml Dinstall.properties= props-file  
appserver.domain.bounce
```

其中 `props-file` 是您在第 16 頁的「在 [以後配置] 方式安裝之後以 Root 身份使用自訂特性配置 Service Registry」或第 17 頁的「在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry」中所編輯 `install.properties` 檔案副本的路徑名稱。

## ▼ 執行資料庫的離線備份

如果資料庫在內嵌式模式中執行，您必須執行離線備份。

### 1 變更為 Service Registry 安裝目錄：

```
cd ServiceRegistry-base/install
```

- 2 執行以下指令 (全部位於同一行) :

```
Ant-base/ant -f build-install.xml Dinstall.properties= props-file  
appserver.domain.stop
```

其中 *props-file* 是您在第 16 頁的「在 [以後配置] 方式安裝之後以 Root 身份使用自訂特性配置 Service Registry」或第 17 頁的「在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry」中所編輯 *install.properties* 檔案副本的路徑名稱。

- 3 執行以下指令 (全部位於同一行) :

```
Ant-base/ant -f build-install.xml Dinstall.properties= props-file backup.db
```

- 4 執行以下指令 (全部位於同一行), 重新啟動網域 :

```
Ant-base/ant -f build-install.xml Dinstall.properties=props-file  
appserver.domain.start
```

## ▼ 從內嵌式模式切換至網路伺服器模式

若要從內嵌式模式 (預設) 切換至網路伺服器模式, 請將 `registry.install.clientDatabase` 特性的設定從 `false` 變更為 `true`, 並且要求資料庫認證。

在編輯特性之後, 重新建立資料庫資源, 然後停止再重新啟動 Application Server。

- 1 變更為 Service Registry 安裝目錄 :

```
cd ServiceRegistry-base/install
```

- 2 在文字編輯器中, 開啓 `install.properties` 檔案副本。
- 3 將 `registry.install.clientDatabase` 特性的設定從 `false` 變更為 `true`。
- 4 將 `registry.install.RequireDatabaseAuthentication` 特性的設定從 `false` 變更為 `true`。
- 5 必要時, 請編輯 `registry.install.DatabaseUserID` 特性的設定。  
對於網路伺服器模式, 此值必須是 APP。
- 6 編輯 `registry.install.DatabasePassword` 特性的設定。  
任何長度都有效。預設值為 `app123`。
- 7 儲存並關閉檔案。

- 8 重新建立資料庫連線池和相關資源。若要這麼做，請執行以下指令 (全部位於同一行)：

```
Ant-base/ant -f build-install.xml Dinstall.properties= props-file
appserver.jdbcResource.update
```

其中 *props-file* 是您在第 16 頁的「在 [以後配置] 方式安裝之後以 Root 身份使用自訂特性配置 Service Registry」或第 17 頁的「在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry」中所編輯 *install.properties* 檔案副本的路徑名稱。

- 9 停止並重新啟動 Registry 的 Application Server 網域。若要這麼做，請執行以下指令 (全部位於同一行)：

```
Ant-base/ant -f build-install.xml Dinstall.properties= props-file
appserver.domain.bounce
```

- 10 啟動 Java DB 資料庫。若要這麼做，請執行以下指令：

```
asadmin start-database --dbhome database-directory
```

依預設，此指令會將資料庫及其記錄檔放在目前的目錄中。使用 `--dbhome` 選項指定資料庫的位置 (通常是在 *RegistryDomain-base /3.0/data/registry/soar*)。

**接下來的步驟** 如果資料庫在網路伺服器模式中執行，您可以執行資料庫的線上備份。

稍後，如果您需要返回內嵌式模式，請按照相同步驟進行，但有以下例外狀況：

- 將 `registry.install.clientDatabase` 特性的設定從 `true` 變更為 `false`。
- 如果您需要停止要求資料庫認證，請將 `registry.install.RequireDatabaseAuthentication` 特性的設定從 `true` 變更為 `false`。
- 如果您需要繼續要求資料庫認證，請依照所需變更使用者 ID 和密碼。  
`registry.install.DatabaseUserID` 的值必須是 APP 或空白。  
`registry.install.DatabasePassword` 的值可以是任何長度。
- 請勿執行步驟 10。您不需要分別啟動資料庫。

## ▼ 執行資料庫的線上備份

- 1 變更為 Service Registry 安裝目錄：

```
cd ServiceRegistry-base/install
```

- 2 檢查 `install.properties` 檔案副本的 `registry.install.clientDatabase` 特性是否設定為 `true`。

- 3 執行以下指令 (全部位於同一行)：

```
Ant-base/ant -f build-install.xml Dinstall.properties= props-file backup.db
```

其中 *props-file* 是您在第 16 頁的「在 [以後配置] 方式安裝之後以 Root 身份使用自訂特性配置 Service Registry」或第 17 頁的「在 [以後配置] 方式安裝之後以非 Root 使用者身份使用自訂特性配置 Service Registry」中所編輯 `install.properties` 檔案副本的路徑名稱。





## 使用 Administration Tool

---

本章說明如何使用 Service Registry 的 Administration Tool (「Admin Tool」)。

本章包含下列小節：

- 第 41 頁的「關於 Admin Tool」
- 第 41 頁的「啓動 Admin Tool」
- 第 45 頁的「Admin Tool 功能」
- 第 46 頁的「使用 Admin Tool 指令」

### 關於 Admin Tool

Service Registry Administration Tool 提供簡單的指令行介面，用於一般的管理工作，例如將關聯增加至登錄 (Registry) 和從登錄 (Registry) 移除物件。

此工具可以在以下兩種模式中作業：

- 在批次模式中，您可以在此工具的指令行中指定一條或多條指令。
- 在互動式模式中，您可以在此工具的互動式 shell 中輸入指令。

有幾個指令 (像是 `ls` 和 `rm`) 可模擬 UNIX® 中處理檔案和資料夾之常見指令的名稱和運作方式。其他指令沒有相應的 UNIX 指令。

### 啓動 Admin Tool

若要啓動 Admin Tool，請執行 `admin-tool.jar` 檔案：

```
java java-options -jar ServiceRegistry-base/lib/admin-tool.jar [admin-tool-options]...
```

java 指令通常位在 `/usr/jdk/entsys-j2se/bin` 目錄中。

`ServiceRegistry-base` 位於 Solaris 作業系統上的 `/opt/SUNWsrcv-registry`，若是 Linux 和 HP-UX 系統，則位於 `/opt/sun/srcv-registry`。

您可放心略過啓動工具時出現的警告訊息。

若要結束 Admin Tool，請使用 `quit` 指令。

## 批次模式

若要在批次模式中執行 Admin Tool，請在啓動 Admin Tool 時在指令行中指定 `-command` 選項。

例如，下列指令會執行 `ls` 指令：

```
java -jar ServiceRegistry-base/lib/admin-tool.jar -command "ls *.html"
```

Admin Tool 會將您的指令以及工具的回應，回應 (echo) 至螢幕上，然後在指令執行之後結束。

請確定正確退出對 shell 很重要的所有字元。

## 互動式模式

若要在互動式模式中執行 Admin Tool，請透過在指令行中指定除 `-command` 之外的任何選項 (或不指定選項)，來啓動 Admin Tool shell：

```
java -jar ServiceRegistry-base/lib/admin-tool.jar
```

Admin Tool 會顯示以下提示並等待您的輸入：

```
admin>
```

## Admin Tool 指令行選項

Admin Tool 可辨識之指令行選項的清單和說明，請分別參閱第 42 頁的「提要」和第 42 頁的「選項」。

### 提要

```
[-alias alias] [-command commands] [-debug] [-help] [-keypass keypass]  
[-localdir localdir] [-locale locale] [-registry url] [-root locator [-create]]  
[-sqlselect SQL-statement] [-verbose]
```

### 選項

`-alias` 存取金鑰庫中的使用者憑證時要使用的別名。指定您註冊為使用者時所使用的別名。如果您要使用 Admin Tool 發佈資料到登錄 (Registry)，則需要此選項。

- command 要執行(而非從互動式 shell 取得指令)的 Admin Tool 指令序列。使用分號(;)分隔多條指令。您不需要在 *commands* 中包含 quit 指令。如果您需要使用分號(不是指令分隔符)，請在分號前加反斜線：
- ```
\;
```
- 執行 Admin Tool 的 shell 可能需要您使用另一個反斜線來退出此反斜線：
- ```
\\;
```
- 如果任何指令中含有空格，請以單引號或雙引號括住整個指令序列，工具便可將序列視為一個(而非數個)指令行參數。此外，如果 shell 也會將分號解譯為分隔 shell 指令，則您務必始終以引號括住多條 Admin Tool 指令的序列。
- create 如有必要，依需要建立 -root 選項指定的 Registry 套裝軟體 (RegistryPackage) 以及父系 Registry 套裝軟體 (RegistryPackage) 物件。僅當執行 Admin Tool 的使用者被授權建立物件時此選項才有效。
- debug 輸出有助於除錯的附加資訊。
- help 提供這些選項的清單。
- keypass 存取金鑰庫中的使用者憑證時要使用的密碼。指定您註冊為使用者時所使用的密碼。如果您要使用 Admin Tool 發佈資料到 Registry，則需要此選項。
- localdir 與本機檔案系統中的檔案相關的指令在本機檔案系統中的基底目錄。
- locale 要用於選取資源束以用於錯誤和狀態訊息的語言環境(例如，en 或 fr)。預設的語言環境由 Java 虛擬機器 (JVM) 決定。
- registry 要連線至的 ebXML Registry 的 URL。預設為 `http://localhost:6480/soar/registry/soap`。
- root Registry 套裝軟體 (RegistryPackage) 的定位器(例如 `/registry/userData`)，該定位器將用作將儲存庫視為 Registry 套裝軟體 (RegistryPackage) 物件(每個均含有其他 RegistryObject 和 Registry 套裝軟體 (RegistryPackage) 物件)之樹狀結構的指令的基底。預設為針對所有使用者資料定義的 Registry 套裝軟體 (RegistryPackage)：`/registry/userData`。
- sqlselect 執行 *SQL-statement* 以選取登錄 (Registry) 物件。敘述必須是以 `select` 起始的完整 SQL 敘述。SQL 敘述必須以引號括住，但不必以分號終止。如果您指定此選項，然後使用不帶引數的 `select` 指令，則該指令將執行 *SQL-statement*，直到您使用帶有 *SQL-statement* 以外其他引數的 `select` 指令為止。

`-v` | `-verbose` 指定狀態訊息的詳細輸出。

---

備註 `-help` 選項的輸出列出此發行版本不支援的兩個選項：`-class` 和 `-property`。

---

## 使用 Admin Tool 將內容發佈到登錄 (Registry)

某些 Admin Tool 指令可用來將內容發佈到登錄 (Registry)：例如，`cp` 和 `import`。此外，`rm` 指令可用來刪除登錄 (Registry) 中的內容。使用這些指令之前，您必須執行某些額外的步驟。

### ▼ 讓您自己可以將內容發佈到登錄 (Registry)

- 1 依照「Service Registry 3.1 使用者指南」中的「建立使用者帳號」一節的說明，執行使用者註冊。

請記住您下載 PKCS12 憑證的位置，以及您指定的使用者名稱和密碼。

- 2 啓動 Admin Tool：

```
java -jar ServiceRegistry-base/lib/admin-tool.jar
```

- 3 執行 `keystoreMover` 指令，將 PKCS12 憑證匯出至 JKS 金鑰庫。如需詳細資訊，請參閱第 59 頁的「`keystoreMover`」。

一般而言，您只需要指定指令範例中所顯示的四個選項。

- 4 停止 Admin Tool：

```
quit
```

- 5 再次啓動 Admin Tool。此時指定選項，如下所示：

```
java -Djaxr-ebxml.security.storetype=JKS \  
-Djaxr-ebxml.security.keystore=security/filename \  
-Djaxr-ebxml.security.storepass=ebxmlrr \  
-jar ServiceRegistry-base/lib/admin-tool.jar -alias alias -keypass password
```

其中 `filename` 是憑證檔案的名稱，通常是 `keystore.jks`。位置 `security/filename` 與目錄 `$HOME/soar/3.0/jaxr-ebxml` 相關。`alias` 和 `password` 值是您在建立使用者帳號時指定的值。

若要節省鍵入的時間，可建立程序檔執行此指令。

# Admin Tool 功能

本小節說明 Admin Tool 的以下功能：

- 第 45 頁的「許可權」
- 第 45 頁的「顯示異常」
- 第 45 頁的「識別登錄 (Registry) 物件」
- 第 46 頁的「語言環境對指定名稱的影響」
- 第 46 頁的「大小寫相符性」

## 許可權

使用 Admin Tool 時，您僅能執行在啟動此工具時為其指定金鑰別名和密碼的使用者可以執行的動作。只有身為管理員角色的使用者，才能執行指令以變更非使用者所擁有的物件。如需詳細資訊，請參閱第 24 頁的「建立管理員」。

## 顯示異常

Admin Tool 可讓您避免在指令失敗時檢視很長的堆疊追蹤。

指令失敗時，Admin Tool 會輸出堆疊追蹤的第一行和以下訊息：

```
An error occurred when executing the function. Use the show exception command to view messages.
```

如果您需要更多資訊，請接著執行 `show exception` 指令，以查看完整堆疊追蹤。

`show exception` 指令會自動顯示前一緊接指令的堆疊追蹤。

## 識別登錄 (Registry) 物件

識別登錄 (Registry) 物件的主要方法是依名稱。然而，您通常可按照從 `registry` 根至 `RegistryPackage` 的路徑來識別 Registry 套裝軟體 (RegistryPackage) 物件。例如，`/registry/userData` 是 `userData` Registry 套裝軟體 (RegistryPackage) 的路徑。

某些名稱的比對支援萬用字元。使用問號 (?) 可比對單一字元。使用星號 (\*) 可比對零個或多個字元。

某些指令 (例如，`cd` 和 `chown`) 支援依物件的單一資源名稱 (URN) (必須包含前導字元 `urn:`) 來識別物件。例如，`urn:uuid:2702f889-3ced-4d49-82d1-e4cd846cb9e4` 是有效的 URN。

chown 和 cp 指令還支援使用 `%number` 來參考前一 `users` 指令列出的**使用者 (User)**。

對於某些指令，如果要輸入含有空格的名稱，請以雙引號括住整個名稱，或在名稱中的每個空格前加上反斜線。

select 指令支援使用 SQL 萬用字元：百分比符號 (%) 用於比對多個字元，而底線 (\_) 則用於比對單一字元。

## 語言環境對指定名稱的影響

RegistryObject (或 Registry 套裝軟體 (RegistryPackage)) 可具有多個名稱，每個名稱關聯一個不同的語言環境。

您指定的路徑和物件名稱僅根據目前的語言環境計算。嘗試依名稱選取具有多個名稱的登錄 (Registry) 物件時，Registry 會嘗試將您提供的名稱僅與一個登錄 (Registry) 物件名稱 (其語言環境與目前的語言環境最相符) 選項相比對，而非與登錄 (Registry) 物件的所有名稱相比對。

例如，假定目前 Registry 套裝軟體 (RegistryPackage) 的成員物件具有兩個名稱，每個名稱關聯一個不同的語言環境：在 en (英文) 語言環境中為 red，在 fr (法文) 語言環境中為 rouge。目前的語言環境是 en 時，指令 `ls rouge` 不會顯示該成員物件，但當語言環境是 fr (或其一個變體) 時，則會顯示該成員物件。

## 大小寫相符性

Admin Tool 辨識的指令名稱和文字參數無須大小寫相符。例如，`ls`、`Ls` 和 `LS` 均指同一個指令。

您為其提供值的選項會被逐字地傳送至使用該選項的代碼。

## 使用 Admin Tool 指令

以下小節說明可用的指令。

- 第 47 頁的 「add association」
- 第 48 頁的 「add user」
- 第 54 頁的 「cd」
- 第 55 頁的 「chown」
- 第 55 頁的 「cp」
- 第 57 頁的 「echo」
- 第 57 頁的 「help」
- 第 58 頁的 「import」

- 第 59 頁的 「keystoreMover」
- 第 61 頁的 「lcd」
- 第 62 頁的 「ls」
- 第 63 頁的 「pwd」
- 第 63 頁的 「quit」
- 第 63 頁的 「rm」
- 第 65 頁的 「select」
- 第 65 頁的 「set」
- 第 66 頁的 「show」
- 第 67 頁的 「users」

對於每個指令，選項和運算元的提要 and 說明遵守以下印刷排版慣例：

- **斜體字**代表在執行指令時應以實際值替代的選項引數或運算元。
- 大括號 ( { } ) 分隔選項或運算元的選擇，其中必須包含一個選項或運算元。選項或運算元以垂直列 ( | ) 分隔。
- 方括號 ( [ ] ) 分隔可以省略的選項、運算元，或是選項或運算元的選擇。
- 選項或運算元後面的省略號 ( ... ) 指示您可以重複引數或運算元。

其他內容是執行指令時必須包含的文字。

## add association

將**關聯** (Association) 物件增加至登錄 (Registry)。

### 提要

**add association** -type *association-type* *sourceURN* *targetURN*

### 說明

add association 指令會將指定類型的**關聯** (Association) 物件增加至登錄 (Registry)。

您可以使用以下任何類型：

- AccessControlPolicyFor
- AffiliatedWith (具有子概念 EmployeeOf 和 MemberOf)
- Contains
- ContentManagementServiceFor
- EquivalentTo
- Extends

- ExternallyLinks
- HasFederationMember
- HasMember
- Implements
- InstanceOf
- InvocationControlFileFor (具有子概念 CatalogingControlFileFor 和 ValidationControlFileFor)
- OffersService
- OwnerOf
- RelatedTo
- Replaces
- ResponsibleFor
- SubmitterOf
- Supersedes
- Uses

## 選項

`-type` 關聯 (Association) 物件的類型。

## 運算元

`sourceURN` 來源物件的 URN。

`targetURN` 目標物件的 URN。

## 範例

以下指令 (全部位於同一行) 會在具有指定的兩個 URN 的物件之間建立 RelatedTo 關係。

```
admin> add association -type RelatedTo
urn:uuid:ab80d8f7-3bea-4467-ad26-d04a40045446
urn:uuid:7a54bbca-2131-4a49-8ecc-e7b4ac86c4fd
```

## add user

將使用者增加至登錄 (Registry)。



## 提要

```

add user [-edit] [- load pathname] [-firstname string] [-lastname string]
[-middleName string] -alias string -keypass string [-post1. -type string]
[-post1.city string] [-post1.country string] [-post1.postalcode string]
[-post1.stateOrProvince string] [-post1.street string] [-post2.streetNumber string]
[-post2. -type string] [-post2.city string] [-post2.country string]
[-post2.postalcode string] [-post2.stateOrProvince string] [-post2.street string]
[-post2.streetNumber string] [-post3. -type string] [-post3.city string]
[-post3.country string] [-post3.postalcode string] [-post3.stateOrProvince string]
[-post3.street string] [-post3.streetNumber string] [-telephone1. -type string]
[-telephone1.areaCode string] [-telephone1.countryCode string]
[-telephone1. -extension string] [-telephone1.number string]
[-telephone1. -URL string] [-telephone2. -type string] [-telephone2.areaCode string]
[-telephone2.countryCode string] [-telephone2. -extension string]
[-telephone2.number string] [-telephone2. -URL string] [-telephone3. -type string]
[-telephone3.areaCode string] [-telephone3.countryCode string]
[-telephone3. -extension string] [-telephone3.number string]
[-telephone3. -URL string] [-email1. -type string] [- email1.address string]
[-email2. -type string] [-email12address string] [-email3. -type string]
[-email3.address string]

```

## 說明

`add user` 指令會增加**使用者 (User)** 物件。**使用者 (User)** 物件一般含有至少一個**郵遞地址 (PostalAddress)**、**電話號碼 (TelephoneNumber)** 和**電子郵件地址 (EmailAddress)** 物件。

在指令行本身中或透過使用 `-load` 選項指定有關使用者的資訊，以指定具有該資訊的 Java 特性檔案。資訊選項和 `-load` 選項依其在指令行中顯示的順序計算。例如，您可以在指令行中指定某些特性，從特性檔案載入其他特性，然後使用後續的指令行選項置換特性檔案中的資訊。

您可以為新的使用者最多指定三個地址、電話號碼和電子郵件地址。如果您需要更多，可以在稍後使用 Web Console 或 JAXR 予以增加。

指定地址、電話號碼或電子郵件時，您必須提供其類型的值：例如，`-emailType OfficeEmail`。

對於每個使用者都需要的共用資訊，您可以在指令行使用快捷選項 (例如，`-fn`)。然而，在特性檔案中提供資訊時，您必須使用較長的格式。例如，您可以使用 `-email1.address`、`-emailAddress` 或 `-email`，在指令行指定使用者的第一個電子郵件地址。不過，當您在特性檔案中指定第一個電子郵件地址時，則須使用 `email1.address=`。由於使用者的第二個電子郵件地址僅有一個選項，所以您必須在指令行中使用 `-email2.address`，在特性檔案中使用 `email2.address=`。

如果您指定 `-edit` 選項，則 Admin Tool 會啟動編輯器，以便您可以編輯新使用者的資訊。請參閱選項說明，以取得詳細資訊。

指令會在主目錄中的 `$HOME/soar/3.0/jaxr-ebxml/security/keystore.jks` 檔案內，建立新使用者的憑證金鑰庫。如果您正以 `root` 的身份執行工具，主目錄會是 `/` 或 `/root`。

---

**備註** – 您使用 `-load` 載入或使用 `-edit` 編輯的特性檔案使用 ISO-8859-1 字元集，所有 Java 特性檔案均如此。請參閱 `java.util.Properties.load(InputStream)` 的文件，以取得有關如何在特性檔案中表示 ISO-8859-1 以外其他字元的詳細資訊。

---

## 選項

### `-edit`

使 Admin Tool 啟動編輯器，以便您可以編輯新使用者的資訊。在計算其他指令行參數之後，工具會啟動編輯器。因此，計算指令行或特性檔案中指定的資訊有了結果時，就要開始進行編輯。編輯程式必須無錯誤終止，指令才能繼續。Admin Tool 啟動 `set editor` 指令 (請參閱 [第 65 頁的「set」](#)) 指定的編輯器；依預設為 `vi` 編輯器。

---

**備註** – 在此發行版本中，`-edit` 可以與 `emacsclient` 和 `NetBeans™` 指令 `bin/runide.sh --open` 搭配使用 (但效果不佳)，但不能與 `vi` 搭配使用。

---

### `-load`

指定其內容可指定使用者特性的 Java 特性檔案。特性名稱與 `add user` 指令選項 (例如，`lastName` 和 `postal.type`) 的長格式特性名稱相同。

### `-fn | -firstName`

指定使用者的名字。

### `-ln | -lastName`

指定使用者的姓氏。您必須在指令行或特性檔案中指定姓氏 (必要項目)。

### `-mn | -middleName`

指定使用者的中間名。

### `-alias`

存取金鑰庫中的使用者憑證時要使用的別名。此選項是必要項目。別名必須至少具有三個字元。

### `-keypass`

存取金鑰庫中的使用者憑證時要使用的密碼。此選項是必要項目。密碼必須至少具有六個字元。

### `-postalType | -postal.type`

第一個 `PostalAddress` 的類型。您必須在指令行或特性檔案中指定類型 (必要項目)。該值是任意字串 (例如，`Office` 或 `Home`)。

- city | -post1.city  
第一個 PostalAddress 的城市。
- country | -post1.country  
第一個 PostalAddress 的國家/地區。
- postalCode | -postcode | -zip | -post1.postalcode  
第一個 PostalAddress 的郵遞區號。
- stateOrProvince | -state | -province | -post1.stateOrProvince  
第一個 PostalAddress 的州或省。
- street | -post1.street  
第一個 PostalAddress 的街道名稱。
- streetNumber | -number | -post1.streetNumber  
第一個 PostalAddress 的街道號碼。
- post2.type  
第二個 PostalAddress 的類型。如果指定第二個 PostalAddress，則必須在指令行或特性檔案中指定類型(必要項目)。該值是任意字串(例如，Office 或 Home)。
- post2.city  
第二個 PostalAddress 的城市。
- post2.country  
第二個 PostalAddress 的國家/地區。
- post2.postalcode  
第二個 PostalAddress 的郵遞區號。
- post2.stateOrProvince  
第二個 PostalAddress 的州或省。
- post2.street  
第二個 PostalAddress 的街道名稱。
- post2.streetNumber  
第二個 PostalAddress 的街道號碼。
- post3.type  
第三個 PostalAddress 的類型。如果指定第三個 PostalAddress，則必須在指令行或特性檔案中指定類型(必要項目)。該值是任意字串(例如，Office 或 Home)。
- post3.city  
第三個 PostalAddress 的城市。
- post3.country  
第三個 PostalAddress 的國家/地區。
- post3.postalcode  
第三個 PostalAddress 的郵遞區號。

- post3.stateOrProvince  
第三個 PostalAddress 的州或省。
- post3.street  
第三個 PostalAddress 的街道名稱。
- post3.streetNumber  
第三個 PostalAddress 的街道號碼。
- phoneType | -telephone1.type  
第一個 TelephoneNumber 的類型。您必須在指令行或特性檔案中指定類型 (必要項目)。該值是任意字串，但您可以指定以下一種已知類型：Beeper、FAX、HomePhone、MobilePhone 或 OfficePhone。
- areaCode | -telephone1.areaCode  
第一個 TelephoneNumber 的區域代碼。
- countryCode | -telephone1.countryCode  
第一個 TelephoneNumber 的國家/地區代碼。
- extension | -telephone1.extension  
第一個 TelephoneNumber 的分機號碼。
- number | -telephone1.number  
第一個 TelephoneNumber 的電話號碼尾碼 (不包含國碼或區碼)。您必須在指令行或特性檔案中指定號碼 (必要項目)。
- URL | -telephone1.URL  
第一個 TelephoneNumber 的 URL (可透過電子方式撥打此號碼的 URL)。
- telephone2.type  
第二個 TelephoneNumber 的類型。如果指定第二個 TelephoneNumber，則必須在指令行或特性檔案中指定類型 (必要項目)。該值是任意字串，但您可以指定以下一種已知類型：Beeper、FAX、HomePhone、MobilePhone 或 OfficePhone。
- telephone2.areaCode  
第二個 TelephoneNumber 的區域代碼。
- telephone2.countryCode  
第二個 TelephoneNumber 的國碼。
- telephone2.extension  
第二個 TelephoneNumber 的分機號碼。
- telephone2.number  
第二個 TelephoneNumber 的電話號碼尾碼 (不包含國碼或區碼)。如果指定第二個 TelephoneNumber，則必須在指令行或特性檔案中指定號碼 (必要項目)。
- telephone2.URL  
第二個 TelephoneNumber 的 URL (可透過電子方式撥打此號碼的 URL)。

- telephone3.type  
第三個 TelephoneNumber 的類型。如果指定第三個 TelephoneNumber，則必須在指令行或特性檔案中指定類型 (必要項目)。該值是任意字串，但您可以指定以下一種已知類型：Beeper、FAX、HomePhone、MobilePhone 或 OfficePhone。
- telephone3.areaCode  
第三個 TelephoneNumber 的區域代碼。
- telephone3.countryCode  
第三個 TelephoneNumber 的國家/地區代碼。
- telephone3.extension  
第三個 TelephoneNumber 的分機號碼。
- telephone3.number  
第三個 TelephoneNumber 的電話號碼尾碼 (不包含國碼或區碼)。如果指定第三個 TelephoneNumber，則必須在指令行或特性檔案中指定號碼 (必要項目)。
- telephone3.URL  
第三個 TelephoneNumber 的 URL (可透過電子方式撥打此號碼的 URL)。
- emailType | -email1.type  
第一個 EmailAddress 的類型。您必須在指令行或特性檔案中指定類型 (必要項目)。該值是任意字串，但您可以指定以下一種已知類型：HomeEmail 或 OfficeEmail。
- emailAddress | -email | -email1.address  
第一個電子郵件地址。第一個電子郵件地址是必需的。
- email2.type  
第二個 EmailAddress 的類型。如果指定第二個 EmailAddress，則必須在指令行或特性檔案中指定類型 (必要項目)。該值是任意字串，但您可以指定以下一種已知類型：HomeEmail 或 OfficeEmail。
- email2.address  
第二個電子郵件地址。
- email3.type  
第三個 EmailAddress 的類型。如果指定第三個 EmailAddress，則必須在指令行或特性檔案中指定類型 (必要項目)。該值是任意字串，但您可以指定以下一種已知類型：HomeEmail 或 OfficeEmail。
- email3.address  
第三個電子郵件地址。

## 範例

以下指令會從使用者主目錄中的檔案 JaneSmith.properties 載入**使用者 (User)** 特性。

```
admin> add user -load ~/JaneSmith.properties
```

以下指令 (全部位於同一行) 會指定建立**使用者 (User)** 物件所需的最少特性。

```
admin> add user -ln Smith -postaltype Office -country US
-phonenumber Office -number 333-3333 -emailtype OfficeEmail
-emailaddress JaneSmith@JaneSmith.com -alias 123 -keypass 123456
```

## cd

變更 Registry 套裝軟體 (RegistryPackage) 位置。

## 提要

```
cd {locator | URN}
```

## 說明

cd 指令會將目錄變更 (隱喻地) 至指定路徑下的或具有指定 URN 的 Registry 套裝軟體 (RegistryPackage)。

當多個 Registry 套裝軟體 (RegistryPackage) 物件具有相同路徑 (對於目前的語言環境) 時，指令會變更至指定的 URN。

## 運算元

*locator* 登錄 (Registry) 物件名稱的路徑，從儲存庫根到儲存庫中的物件，每個名稱前均加有正斜線 (/)。

例如，registry Registry 套裝軟體 (RegistryPackage) (本身不是任何 Registry 套裝軟體 (RegistryPackage) 的成員) 所屬成員 userData Registry 套裝軟體 (RegistryPackage) 的定位器是 /registry/userData。userData Registry 套裝軟體 (RegistryPackage) 所屬成員 folder1 Registry 套裝軟體 (RegistryPackage) 的定位器是 /registry/userData/folder1。

*URN* RegistryPackage 的 URN，必須是以 urn: 起始的 URN。

## 範例

以下指令會將目錄變更至 URN 為 urn:uuid:92d3fd01-a929-4eba-a5b4-a3f036733017 的 Registry 套裝軟體 (RegistryPackage)。

```
admin> cd urn:uuid:92d3fd01-a929-4eba-a5b4-a3f036733017
```

以下指令會將目錄變更至位置 /registry/userData/myData。

```
admin> cd /registry/userData/myData
```

## chown

變更 RegistryObject 的所有者。

### 提要

**chown** {URN | %index}

### 說明

chown 指令會將使用前一 select 指令所選取物件的所有權變更至 URN 指定的使用者，或變更至前一 users 指令所列出使用者 URN 參照指定的使用者。

僅具有管理員角色的使用者才能成功執行此指令。

### 運算元

URN URN 指定的**使用者 (User)** 物件。

%index 在前一 users 指令中所列出**使用者 (User)**物件的 URN 數字參照。

### 範例

以下指令會將已選取物件的所有權變更至 URN

urn:uuid:26aa17e6-d669-4775-bfe8-a3a484d3e079 指定的使用者。

```
admin> chown urn:uuid:26aa17e6-d669-4775-bfe8-a3a484d3e079
```

以下指令會將已選取物件的所有權變更至前一 users 指令中編號為 2 的使用者。

```
admin> chown %2
```

### 另請參閱

- [第 65 頁的「select」](#)
- [第 67 頁的「users」](#)

## cp

將檔案和資料夾複製到登錄 (Registry)。

### 提要

**cp** [-owner {URN | % index}] [- exclude *pattern*]... [-include *pattern*]... *pattern*...

## 說明

cp 指令會將資料夾和檔案分別複製到登錄 (Registry) 做為 Registry 套裝軟體 (RegistryPackage) 和外部物件 (ExtrinsicObject) 物件。

從其上複製檔案和資料夾的本機檔案系統上的本機目錄預設為啓動 Admin Tool 的目前目錄。啓動 Admin Tool 時，您可以使用 `-localdir` 選項變更本機目錄。啓動 Admin Tool 之後，您可以使用 `lcd` 指令變更本機目錄。您可以使用 `show localdir` 指令，取得目前本機目錄的絕對路徑。

此指令是遞迴的。也就是說，如果您指定目錄，此指令會複製該目錄下的所有檔案和資料夾。

## 選項

- `-owner` 將複製的登錄 (Registry) 物件所有者設定為 *URN* 或 *%index* 引數指定的使用者。請參閱 `chown` 指令的說明，以取得這些引數的說明。您必須具有管理員的角色，才能指定除自己之外的所有者。
- `-exclude` 複製除名稱中含有指定式樣的檔案之外的所有檔案，其中 *pattern* 是一種式樣，包含文字字元和特殊字元星號 (\*) (表示零個或多個字元) 及問號 (?) (表示有且僅有一個字元)。  
您可以多次指定此選項。
- `-include` 複製除名稱中含有指定式樣的檔案之外的所有檔案，其中 *pattern* 是一種式樣，包含文字字元和特殊字元星號 (\*) (表示零個或多個字元) 及問號 (?) (表示有且僅有一個字元)。  
您可以多次指定此選項。

## 運算元

*pattern* 要複製的檔案或資料夾，由包含文字字元和特殊字元星號 (\*) (表示零個或多個字元) 及問號 (?) (表示有且僅有一個字元) 的式樣指定。您可以指定多個 *pattern*。

## 範例

以下指令會將目錄 `mydir` 複製到登錄 (Registry)，由前一 `users` 指令中編號為 4 的使用者所有。

```
admin> cp -owner %4 mydir
```

以下指令會將目錄 `mydir` 複製到登錄 (Registry)，排除以字串 `.z` 或 `.c` 結束的檔案和目錄。



```
admin> cp mydir -exclude \\.z -exclude \\.c
```

## 另請參閱

- 第 61 頁的「`lcd`」
- 第 66 頁的「`show`」

## echo

回應字串。

## 提要

`echo string`

## 說明

`echo` 指令會將指定的 *string* 回應至輸出。在批次模式中執行 Admin Tool 時，於 `-command` 選項中指定此指令會非常有用。

## 運算元

*string* 字元序列。

## 範例

以下指令會將 `ls` 指令的日期和結果輸出至記錄檔。

```
java -jar admin-tool.jar -command "echo "date"; ls" > admin.log
```

## help

顯示有關指令的資訊。

## 提要

`help [command_name]`

## 說明

`help` 指令會顯示有關可用指令或指定指令的資訊。

對於含有子指令的指令 (例如 `add` 和 `show`)，`help` 指令會顯示有關子指令的資訊。

如果您未指定引數，則 `help` 指令會顯示所有指令的用法資訊。

## 運算元

`command_name`     Admin Tool 指令的名稱。

## 範例

以下指令會顯示所有指令的用法資訊。

```
admin> help
```

以下指令會顯示 `lcd` 指令的用法資訊。

```
admin> help lcd
```

以下指令會顯示 `add` 子指令的用法資訊。

```
admin> help add
```

## import

匯入定義登錄 (Registry) 物件的 XML 檔案。

## 提要

```
import [ {-a | --attach} pathname, mimeType, id]... submitObjectsRequest
```

## 說明

`import` 指令會提交符合 ebXML Registry Services and Protocols Version 3.0 規格中所述 `SubmitObjectsRequest` 協定的 XML 檔案，將一或多個新物件或儲存庫項目增加至登錄 (Registry)。

XML 檔案以及 SOAP 訊息內容提供的所有附件會傳送到登錄 (Registry) 進行處理。因此此操作相當低階，適合非常熟悉 ebXML 規格的使用者使用。

## 選項

`-a | --attach` 附加檔案做為外部物件的儲存庫項目。`pathname` 是所要增加檔案的路徑名稱。`mimeType` 指定檔案的 MIME 類型。`id` 是外部物件的唯一識別碼，而此物件是儲存庫項目 (??)。您可以多次指定此選項。

## 運算元

`submitObjectsRequest` 包含登錄 (Registry) 物件定義的 XML 檔案。

## 範例

下列指令會匯入 `MyRequest.xml` 檔案中定義的物件群組：

```
admin> import MyRequest.xml
```

下列指令 (全部位於同一行) 會匯入外部物件和其儲存庫項目 (影像檔)：

```
admin> import --attach chicken.jpg, image/jpeg, urn:bird:poultry:chicken
ChickenRequest.xml
```

## keystoreMover

將一或多個金鑰從一個金鑰庫格式匯出成另一種格式。

## 提要

```
keystoreMover [-sourceKeystoreType {JKS | PKCS12}] -sourceKeystorePath pathname
-sourceKeystorePassword password
[-sourceAlias alias [-sourceKeyPassword password]] [-destinationKeystoreType {JKS
| PKCS12}] -destinationKeystorePath pathname
-destinationKeystorePassword password [-destinationAlias newAlias]
[-destinationKeyPassword password]
```

## 說明

Registry 的開發者介面需要使用 JKS 金鑰庫，而 Web Console 需要您可匯入至 Web 瀏覽器的 PKCS12 或 DER 憑證。

如果您使用登錄 (Registry) 產生的 PKCS12 憑證建立使用者帳號，並且希望使用 Admin Tool 將內容發佈至登錄 (Registry)，便可使用 `keystoreMover` 指令將憑證匯出至 JKS 金鑰庫。您也可以使用此指令，針對 Registry 執行開發者應用程式。

如果您曾經使用 `add user` 指令建立使用者，並且希望該使用者能夠使用 Web Console，您可以使用此指令將 `add user` 建立的 JKS 金鑰庫匯出為 PKCS12 格式。

如需有關使用 Web Console 建立使用者帳號的詳細資訊，請參閱「Service Registry 3.1 使用者指南」中的「建立使用者帳號」一節。如需有關搭配 Admin Tool 使用 `keystoreMover` 的資訊，請參閱第 44 頁的「使用 Admin Tool 將內容發佈到登錄 (Registry)」。如需有關使用 `add user` 指令的資訊，請參閱第 48 頁的「`add user`」。

如需有關開發 Registry 應用程式的資訊，請參閱「Service Registry 3.1 Developer's Guide」。

## 選項

<code>-sourceKeystoreType</code>	指定要匯出的金鑰庫類型。引數必須是 PKCS12 或 JKS。預設值為 PKCS12。
<code>-sourceKeystorePath</code>	指定包含來源金鑰庫的檔案路徑名稱。此選項是必要項目。通常，這是您建立使用者時所建立的憑證檔案路徑名稱。
<code>-sourceKeystorePassword</code>	指定來源金鑰庫的密碼。通常，這是您建立使用者時指定的密碼。此選項是必要項目。
<code>-sourceAlias</code>	指定要匯出的別名。如果您不指定此選項，該指令會匯出金鑰庫中所有的別名。從 Web Console 下載的金鑰庫只包含一個別名。
<code>-sourceKeyPassword</code>	指定別名特有的密碼 (相對於金鑰庫密碼)。如果您不指定此選項，該密碼就會與金鑰庫的密碼相同 (這是一般狀況)。
<code>-destinationKeystoreType</code>	指定目標金鑰庫的類型。引數可以是 PKCS12 或 JKS。預設值為 JKS。
<code>-destinationKeystorePath</code>	指定包含目標金鑰庫的檔案路徑名稱。此選項是必要項目。通常，此引數是 <code>HOME/soar/3.0/jaxr-ebxml/security/keystore.jks</code> ，其中 <code>HOME</code> 是使用者的主目錄。
<code>-destinationKeystorePassword</code>	指定目標金鑰庫的密碼。此引數是必要項目。此特性的預設值為 <code>ebxmlrr</code> 。
<code>-destinationAlias</code>	指定新的別名 (如果您要為別名重新命名)。如果您不指定此選項，新別名就會與來源憑證的別名相同。
<code>-destinationKeyPassword</code>	指定別名特有的密碼 (相對於金鑰庫密碼)。如果您不指定此選項，該密碼就會與金鑰庫的密碼相同 (這是一般狀況)。

---

備註 - 所有密碼的長度都必須至少為 6 個字元。

---

## 範例

下列指令將使用者主目錄中 `generated-key.p12` 的憑證匯出至 `soar/3.0/jaxr-ebxml/security/keystore.jks` 以及使用者主目錄中的 JKS 金鑰庫。來源金鑰庫密碼是使用者向 Registry 註冊時所提供的密碼。目標金鑰庫密碼是 `ebxmlrr` 的預設值。在同一行指定全部指令。

```
admin> keystoreMover -sourceKeystorePath /home/myname/generated-key.p12
-sourceKeystorePassword mypass -destinationKeystorePath
/home/myname/soar/3.0/jaxr-ebxml/security/keystore.jks
-destinationKeystorePassword ebxmlrr
```

## lcd

變更本機檔案系統上的目前目錄。

## 提要

`lcd [pathname]`

## 說明

`lcd` 指令會變更本機檔案系統上的目前本機目錄。

如果您未指定引數，則 `lcd` 指令會將目前目錄變更至預設主目錄。

## 運算元

*pathname* 目錄名稱可以是絕對的，也可以是相對的。

## 範例

以下指令會將目前本機目錄變更至 `/usr/share` 目錄。

```
admin> lcd /usr/share
```

以下指令會將目前本機目錄變更至本機檔案系統上的預設主目錄。

```
admin> lcd
```

## ls

列出目前 Registry 套裝軟體 (RegistryPackage) 中的物件。

## 提要

ls [ {*pattern* | *URN*}...]

## 說明

ls 指令 (此指令沒有引數) 會列出目前 Registry 套裝軟體 (RegistryPackage) 中的物件。若是提供 *pattern* 或 *URN*，該指令會列出其名稱 (在目前的語言環境中) 或唯一識別碼符合 *pattern* 或 *URN* 之目前 Registry 套裝軟體 (RegistryPackage) 中的物件。

## 運算元

*pattern* 包含文字字元和特殊字元星號 (\*) (表示零個或多個字元) 及問號 (?) (表示有且僅有一個字元) 的式樣。您可以指定多個 *pattern*。

*URN* 以 urn: 起始的 URN，例如  
urn:uuid:4a6741e7-4be1-4cfb-960a-e5520356c4fd。您可以指定多個 *URN*。  
URN 必須是物件的唯一識別碼，而非邏輯識別碼。

## 範例

以下指令會列出目前 Registry 套裝軟體 (RegistryPackage) 中的所有物件。

```
admin> ls
```

以下指令會列出名稱與式樣 urn:bird:poultry:chicken 相符或 ID 為 urn:bird:poultry:chicken 的所有物件。

```
admin> ls urn:bird:poultry:chicken
```

以下指令會列出名稱與式樣 \*bird\* 相符的所有物件。(如果 \*bird\* 是有效 ID，則還會列出 ID 為 \*bird\* 的物件。)

```
admin> ls *bird*
```

以下指令會列出名稱與式樣 \*bird\* 或 urn:bird:poultry:chicken 相符，或 ID 為 urn:bird:poultry:chicken 的所有物件。

```
admin> ls *bird* urn:bird:poultry:chicken
```

## pwd

顯示目前 Registry 套裝軟體 (RegistryPackage) 的路徑。

## 提要

**pwd**

## 說明

pwd 指令會使用目前語言環境的最相符名稱顯示目前 Registry 套裝軟體 (RegistryPackage) 的一條或多條路徑。該指令也會顯示路徑的語言環境。

## 範例

```
admin> pwd
(en_US) /registry/userData
```

## quit

結束 Admin Tool。

## 提要

**quit**

## 說明

quit 指令會結束 Admin Tool。

## 範例

```
admin> quit
```

## rm

從 Registry 套裝軟體 (RegistryPackage) 中移除物件。

## 提要

```
rm [-d] [-r] { pattern | URN}...
```

## 說明

`rm` 指令會移除目前 Registry 套裝軟體 (RegistryPackage) 中名稱 (在目前的語言環境中) 與 *pattern* 或 URN 指定的式樣相符的成員物件。

如果相符的 RegistryObject 是多個 Registry 套裝軟體 (RegistryPackage) 物件的成員，則此指令僅移除目前 Registry 套裝軟體 (RegistryPackage) 和該物件之間的關聯。只有在移除關聯時，物件不再與其他任何 RegistryObject 相關聯 (包括其他含有 Registry 套裝軟體 (RegistryPackage) 物件的項目) 的狀況下，才會從登錄 (Registry) 中移除物件。

如果相符的成員物件本身是含有其他物件的 Registry 套裝軟體 (RegistryPackage)，則物件和目前 Registry 套裝軟體 (RegistryPackage) 與成員 Registry 套裝軟體 (RegistryPackage) 之間的關聯均不會移除，除非指定 `-r` 或 `-d` 選項。

如果指定了 `-d` 和 `-r` 選項，則會遞迴套用 `-d` 選項，所以 `-r` 要選取的所有物件 (及其關聯) 均會被移除，無論其是否具有其他關聯。

## 選項

- `-d` 移除目前的 Registry 套裝軟體 (RegistryPackage) 與指定的 Registry 套裝軟體 (RegistryPackage) 之間的關聯。只有在指定的 Registry 套裝軟體 (RegistryPackage) 僅剩餘與其成員物件的關聯時，才會將其移除。未透過成為其他 HasMember 關聯目標固定之立即移除 Registry 套裝軟體 (RegistryPackage) 的成員物件，現在可以成為 Registry 根的成員進行存取。
- `-r` 移除指定的 Registry 套裝軟體 (RegistryPackage) 物件及其所有派生物件 (物件具有其他關聯時除外)。

## 運算元

*pattern* 包含文字字元和特殊字元星號 (\*) (表示零個或多個字元) 及問號 (?) (表示有且僅有一個字元) 的式樣。您可以指定多個 *pattern*。

URN 以 urn: 起始的 URN，例如  
urn:uuid:4a6741e7-4be1-4cfb-960a-e5520356c4fd。您可以指定多個 URN。

## 範例

以下指令會移除含有字串 "stat" 的所有 Registry 套裝軟體 (RegistryPackage) 物件及其所有派生物件。

```
admin> rm -r *stat*
```



## select

執行 SQL select 敘述。

### 提要

**select** [SQL]

### 說明

**select** 指令會選取並列出透過計算整個指令 (將其視為 SQL 查詢進行計算) 指定的物件。如果未指定引數，則該指令會列出前一 **select** 指令或 **-sqlselect** 選項選取的所有物件。

### 運算元

SQL SQL select 敘述 (沒有前導字元 **select**，因為其已作為指令的名稱)。

### 範例

以下指令會列出登錄 (Registry) 中的所有**分類方案** (ClassificationScheme) 物件：

```
admin> select s.* from ClassificationScheme s
```

## set

設定特性值。

### 提要

**set** *property value*

### 說明

**set** 指令會設定 Admin Tool shell 的特性值。

工具支援以下特性和值。

**set debug** {true | on | yes | false | off | no}

啟用或停用除錯訊息輸出。

**set editor** *string*

設定 Admin Tool 啟動互動式編輯器時要使用的指令。在 UNIX 和 Linux 系統上，預設值是 `/bin/vi`。

**set verbose** {true | on | yes | false | off | no}

執行指令時，啟用或停用更詳細訊息的輸出。

## 運算元

*property* 以下特性之一：debug、editor 和 verbose。

*value* 指定特性的支援值。請參閱「說明」小節，以取得詳細資訊。

## 範例

以下指令會將編輯器設定為 `/usr/bin/vi`，而非預設 `/bin/vi`。

```
admin> set editor /usr/bin/vi
```

以下指令會啟動除錯。

```
admin> set debug true
```

以下指令會關閉詳細輸出。

```
admin> set verbose off
```

## show

顯示特性值。

## 提要

```
show [property]
```

## 說明

show 指令會顯示 Admin Tool shell 的特性值。

如果未指定引數，則此指令會顯示所有特性的值。

此指令支援以下特性：

**debug** 是否已啟用除錯輸出。

**editor** Admin Tool 啟動互動式編輯器時要使用的編輯器。

<code>exception</code>	來自前一緊接執行的指令的異常堆疊追蹤 (如果有)。
<code>localdir</code>	本機檔案系統上的目前目錄。使用 <code>lcd</code> 指令可設定此特性。請參閱 <a href="#">第 61 頁的「lcd」</a> ，以取得詳細資訊。
<code>locale</code>	目前的語言環境。
<code>verbose</code>	是否已啟用詳細輸出。

## 運算元

*property* 其目前值要顯示的特性。特性 `exception` 和 `locale` 可以顯示，但您無法使用 `set` 指令對其進行設定。

## 範例

以下指令會顯示來自前一指令的異常。

```
admin> show exception
```

## users

列出目前的**使用者** (User) 物件。

## 提要

**users**

## 說明

`users` 指令會列出目前在登錄 (Registry) 中的**使用者** (User) 物件。

輸出格式如下：

```
%index: URN lastname, firstname middlename
```

在輸出中，*index* 是一個數值，您可以在執行 `chown` 或 `cp` 指令時，使用該數值 (包含百分比符號%) 參考使用者。*lastname*、*firstname* 和 *middlename* 是使用者的姓氏、名字和中間名。

## 範例

以下指令會顯示目前的使用者：

```
admin> users
%0: urn:freebxml:registry:predefinedusers:registryoperator Operator, Registry
%1: urn:freebxml:registry:predefinedusers:registryguest Guest, Registry
%2: urn:freebxml:registry:predefinedusers:farrukh Najmi, Farrukh Salahudin
%3: urn:freebxml:registry:predefinedusers:nikola Stojanovic, Nikola
%4: urn:uuid:799cc524-b7cd-4e51-8b34-d93b79ac52de User, Test
%5: urn:uuid:85428d8e-1bd5-473b-a8c8-b9d595f82728 Parker, Miles
```

## 另請參閱

- [第 55 頁的「chown」](#)
- [第 55 頁的「cp」](#)

# 索引

---

## A

- Admin Console, Application Server, 20
- Admin Tool
  - 用來發佈到登錄 (Registry), 44
  - 指令行選項, 42-44
  - 停止, 63
  - 啟動, 41-44
  - 簡介, 41
  - 權限, 45
- alias 指令行選項, 42
- Application Server Admin Console, 20
- Application Server 網域
  - 停止並重新啟動, 20-21
  - 將根憑證增加至信任清單存放區, 21-22
  - 管理, 18-21

## C

- cd 指令, 54
- chown 指令, 55
- command 指令行選項, 42
- cp 指令, 55-57
- create 指令行選項, 43

## D

- debug 指令行選項, 43

## E

- echo 指令, 57

## H

- help 指令, 57-58
- help 指令行選項, 43

## I

- import 指令, 58-59

## J

- Java 虛擬機器 (JVM) 選項, 配置, 23-24

## K

- keypass 指令行選項, 43
- keystoreMover 指令, 59-61

## L

- lcd 指令, 61
- localdir 指令行選項, 43
- locale 指令行選項, 43
- ls 指令, 62

## P

pwd 指令, 63

## Q

quit 指令, 63

## R

-registry 指令行選項, 43

Registry 套裝軟體 (RegistryPackage) 位置, 變更, 54

Registry 套裝軟體 (RegistryPackage) 物件

- 列出內容, 62

- 建立, 43

- 移除成員物件, 63-64

- 顯示路徑, 63

Registry 網域

- 停止並重新啟動, 20-21

- 將根憑證增加至信任清單存放區, 21-22

- 管理, 18-21

rm 指令, 63-64

-root 指令行選項, 43

## S

select 指令, 65

Service Registry 資料庫, 管理, 35-39

set 指令, 65-66

show 指令, 66-67

SQL 敘述, 執行, 65

SQL 選取敘述, 執行, 43

-sqlselect 指令行選項, 43

## U

users, 列出, 67-68

users 指令, 67-68

## V

-v 指令行選項, 43

-verbose 指令行選項, 43

## W

Web Console

- 配置, 28-33

- 配置搜尋結果顯示, 31-33

- 增加預先定義的查詢, 28-30

- 隱藏分類方案, 30-31

- 變更預設查詢, 30

WSDL Cataloger, 停用, 27

## X

XML 檔案, 匯入, 58-59

大小寫相符性, 在 Admin Tool 中, 46

分類方案, 隱藏不顯示, 30-31

互動式模式, 42

目前目錄, 變更, 61

目錄, 變更, 61

外部連結 (ExternalLink) 物件, 允許 URL 驗證, 23-24

外部網站, 允許存取, 23-24

代理主機和連接埠, 設定, 23-24

字彙表, 連結至, 9

安裝特性, 14-16

批次模式, 42

即時查詢 (AdhocQuery) 物件, 增加至 Web

- Console, 28-30

服務連結 (ServiceBinding) 物件, 允許 URL 驗

- 證, 23-24

建立管理員, 24-25

使用者, 增加至登錄 (Registry), 48-54

使用者名稱, 在指令行中指定, 42

使用者註冊, 限制, 25-26

查詢, 變更預設, 30

重新安裝 Service Registry, 34

除錯特性

- 設定, 65-66

- 顯示值, 66-67

指令

- cd, 54

## 指令 (續)

- chown, 55
- cp, 55-57
- echo, 57
- help, 57-58
- import, 58-59
- keystoreMover, 59-61
- lcd, 61
- ls, 62
- pwd, 63
- quit, 63
- rm, 63-64
- select, 65
- set, 65-66
- show, 66-67
- users, 67-68
- 增加使用者, 48-54
- 增加關聯, 47-48
- 指令行選項, 42-44
  - alias, 42
  - command, 42
  - create, 43
  - debug, 43
  - help, 43
  - keypass, 43
  - localdir, 43
  - locale, 43
  - registry, 43
  - root, 43
  - sqlselect, 43
  - v, 43
  - verbose, 43
- 根憑證, 增加至 registry 網域信任清單存放區, 21-22
- 記錄層級, 變更, 20
- 記憶體, 配置, 23-24
- 連接埠, Service Registry 預設, 19-20
- 配置 Service Registry, 13-23
  - 以 root 的身份, 16-17
  - 以非 root 使用者的身份, 17-18
- 特性值
  - 設定, 65-66
  - 顯示, 66-67
- 設定特性值, 65-66
- 停止 Admin Tool, 63
- 從 Registry 套裝軟體 (RegistryPackage) 中刪除物件, 63-64
- 從 Registry 套裝軟體 (RegistryPackage) 中移除物件, 63-64
- 啓動 Admin Tool, 41-44
- 將檔案和資料夾複製到登錄 (Registry), 55-57
- 登錄 (Registry) 物件
  - 以 Admin Tool 識別, 45-46
  - 列出, 62
  - 匯入, 58-59
  - 變更所有者, 55
- 登錄 (Registry) 物件的版本控制, 啓用, 26-27
- 異常, 以 Admin Tool 顯示, 45
- 異常特性, 顯示值, 66-67
- 結束 Admin Tool, 63
- 萬用字元, 使用於 Admin Tool, 45-46
- 搜尋結果, 配置顯示, 31-33
- 詳細特性
  - 設定, 65-66
  - 顯示值, 66-67
- 預先定義的查詢, 增加至 Web Console, 28-30
- 預設查詢, 變更, 30
- 管理員, 建立, 24-25
- 語言環境, 在 Admin Tool 中指定名稱的影響, 46
- 語言環境特性, 顯示值, 66-67
- 增加使用者指令, 48-54
- 增加關聯指令, 47-48
- 編輯器特性
  - 設定, 65-66
  - 顯示值, 66-67
- 檔案, XML, 匯入, 58-59
- 檔案系統, 本機
  - 基底目錄, 43
  - 變更目前目錄, 61
- 檔案和資料夾, 複製到登錄 (Registry), 55-57
- 關聯, 增加至登錄 (Registry), 47-48
- 顯示, 特性值, 66-67

