

Sun Java™ System Application Server Enterprise Edition 版本說明

版本 8.1 2005Q1

文件號碼 819-2029

Sun Java™ System Application Server Enterprise Edition 8.1 2005Q1 產品大大簡化了建立與管理 Web 服務應用程式的工作。它為可延伸式服務提供了出眾的效能、叢集和高可用性功能，即使出現軟體和硬體故障仍可繼續作業。Application Server Web 服務提供了這樣一條開發途徑，既可以簡化開發程序，又可以提供獨一無二的靈活擴展功能。

此版本說明包含 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 發佈時可用的重要資訊。同時還介紹了新功能和增強功能、已知問題和限制以及其他資訊。在您開始使用 Application Server 8.1 之前，請閱讀此文件。

您可以在 Sun Java System 說明文件網站上找到此版本說明的最新版本：

<http://docs.sun.com/db/prod/slappsrv#hic/> 和

http://docs.sun.com/db/prod/slappsrv#hic/?l=zh_TW。請在安裝和設定您的軟體前查閱此網站並在安裝後定期查閱以檢視最新的版本說明和產品說明文件。

本文件包含以下章節：

- [關於 Application Server Enterprise Edition 8.1 2005Q1](#)
- [已知問題和限制](#)
- [如何報告問題與提供回饋](#)
- [Sun 歡迎您提出意見和建議](#)
- [其他 Sun 資源](#)

本文件中參考了協力廠商 URL，並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。對於此類站台或資源中的 (或通過它們獲得的) 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。Sun 對於因使用或依靠此類網站或資源中 (或透過它們所取得的) 任何內容、產品或服務而造成的或連帶產生的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。

關於 Application Server Enterprise Edition 8.1 2005Q1

Sun Java System Application Server Enterprise Edition 8.1 是與 J2EE 1.4 平台相容的伺服器，用於在大規模生產環境中開發與部署 J2EE 應用程式和基於 Java 技術的 Web 服務。

本節包含以下內容：

- [8.1 版本中的新增功能](#)
- [硬體和軟體需求](#)
- [獨立版本](#)
- [相關說明文件](#)

8.1 版本中的新增功能

Sun Java System Application Server Enterprise Edition 8.1 2005Q1 實作了多項新增功能和增強功能，在以下章節中進行介紹：

- [本版本中的增強功能](#)
- [J2EE 支援](#)
- [高效能](#)
- [可延伸性](#)
- [高可用性](#)
- [JavaServer Faces 1.1 支援](#)

本版本中的增強功能

Application Server Enterprise Edition 8.1 包含以下增強功能：

- **改進的管理** — Application Server 支援使用基於瀏覽器的主控制台或可編寫的命令行介面，對複雜的多重機器企業部署進行遠端安全管理。它還提供一個基於 JMX 的豐富型 API，可以對管理功能與監視功能進行有計劃地遠端安全存取。
- **訊息代理程式** — Application Server 隨附一個整合的企業級訊息代理程式，該代理程式具有高度可用的、可靠的、高效能的以及可延伸的訊息傳送功能。

- **延伸的平台支援** — 支援其他的作業系統、資料庫、語言環境和硬體。
- **Sun Java Enterprise System** — 做為 Sun Java Enterprise System 的主要元件，Application Server 與入口服務和網路識別服務緊密整合。
- **遷移和升級工具** — 這些工具可讓您驗證 J2EE 應用程式是否符合標準並具有可攜性，有助於從其他 J2EE Application Server (JBoss、WebLogic、WebSphere) 進行遷移，還有助於從舊版的 Sun ONE Application Server/iPlanet Application Server 進行升級。
- **支援 Java 2 Standard Edition 5.0** — Application Server 支援 Java 2 Standard Edition 5.0，該版本包含增強的管理功能和監視功能，並在效能和可延伸性方面有諸多改進。
- **JDBC 驅動程式** — Application Server 隨附有 Sun JDBC 驅動程式。
- **Web 服務安全性** — 這些容器訊息安全性機制使用 OASIS WS-Security 標準的 X509 和使用者名稱/密碼設定檔，實作 SOAP Web 服務呼叫的訊息層級驗證 (例如 XML 數位簽名和加密)。
- **WS-I Basic Profile 1.1** — J2EE 1.4 規格規定此發行版本實作 Web Services Interoperability (WS-I) Basic Profile 1.1，以啓用 Web 服務應用程式的互通功能。
- **與 iWay 配接卡的後端可連接性** — Sun Microsystem 現在轉售並支援連接至主要後端系統 (SAP、Siebel、Oracle、CICS 和 IBM MQ Series) 的二十二種 iWay 配接卡，以協助您利用 Application Server 環境中的現有的 IT 應用程式。這些配接卡支援 J2EE 連接器架構規格 1.5 和 Web 服務 (SOAP) 標準，包括可縮短與後端應用程式連線時間的開發工具。
- **最新的 HADB 管理系統** — UNIX® 平台包含新的高可用性資料庫 (HADB) 管理系統 (HADB 版本 4.4)。這會消除對 SSH/RSH 的依賴性，但是需要為 UDP 多重播送配置網路。請參閱「*Sun Java System Application Server Enterprise Edition 8.1 Installation Guide*」，以取得有關 HADB 需求與限制的詳細資訊。

J2EE 支援

Sun Java System Application Server 8.1 2005Q1 支援 J2EE 1.4 平台。下表介紹了在 J2EE 1.4 平台上可用的增強的 API。

表 1 J2EE 1.4 平台上主要的 API 變更

API	描述
組件	
應用程式與應用程式用戶端	透過 XML 模式實作標準部署描述元
Enterprise JavaBeans (EJB) 2.1	計時器服務和 EJB Web 服務端點
Java Servlet 2.4	Web 服務端點過濾器
JavaServer Pages (JSP) 2.0 架構	表示式語言和標籤檔案庫
J2EE 連接器架構 1.5	內收資源配接卡與 Java Message Service (JMS) 可插接性

表 1 J2EE 1.4 平台上主要的 API 變更 (續)

API	描述
Web 服務	
Java Web Services Developer Pack 1.5	用於建立、測試和部署 XML 應用程式、Web 服務和 Web 應用程式的整合式工具箱
用於基於 XML 的遠端程序呼叫的 Java API (JAX-RPC) 1.1	WSDL 與 Java 技術的對映和對 Web 服務用戶端和端點的開發的支援
WS-I Basic Profile 1.0	使用 WSDL 和 SOAP 互通功能的啟用元素
具有 Java 的 API 附件的 SOAP (SAAJ) 1.2	用於基於 SOAP 訊息傳送的 API；促使建立具有附件的 SOAP 訊息
用於 XML 登錄的 Java API (JAXR) 1.0	用於存取 XML 登錄的統一且標準的 API，例如用於通用說明開發與整合 (UDDI 和 ebXML) 的 API
其他	
J2EE Deployment 1.1	啟用 J2EE 組件和應用程式部署的標準 API
J2EE Management 1.0	管理 J2EE 平台的資訊模型的定義
Java Management Extensions (JMX) 1.2	標準管理 API
Java 容器授權合約 (JACC) 1.0	J2EE Application Server 與授權策略提供者之間的安全性合約的定義
用於 XML 處理的 Java API (JAXP) 1.2	使用此 API，應用程式可以剖析和變換 XML 文件；也增加了對 XML 模式處理的支援
JMS 1.1	啟用 J2EE 應用程式組件建立、傳送、接收並讀取訊息的訊息傳送標準；還增加了對佇列與主題的統一 API 的支援
JavaMail 1.3	建立郵件系統模型的抽象類別集；也包括 API 的次要更新

高效能

Application Server 包括高效能的 EJB 容器、Web 容器與服務，並且支援使用 Sun Java System Message Queue 軟體進行的同步訊息傳送。

可延伸性

Application Server 透過伺服器實例的叢集功能和請求負載平衡功能，支援水平可延伸性。它還實現了業界領先的垂直可延伸性，支援大型的多處理器機器。整合的訊息代理程式可以進行叢集，以取得更好的可延伸性和可用性。來自 HTTP 用戶端、基於 RMI/IIOP 豐富型用戶端應用程式、Web 服務用戶端和 JRM 用戶端的用戶端存取可以負載平衡至 Application Server 叢集。

高可用性

Application Server 包括 HTTP、IIOP 和 JMS 用戶端的負載平衡功能、HTTP 階段作業防故障備用支援、EJB 叢集功能和防故障備用支援、高度可用的 EJB 計時器、分散式作業事件恢復功能、支援應用程式持續升級的功能和用於儲存 J2EE 應用程式之暫態狀態的高可用性資料庫。

可用性允許對叢集中的 Application Server 實例進行防故障備用保護。如果一個 Application Server 實例出現故障，則其他 Application Server 實例將接管指定給該故障伺服器的階段作業。階段作業資訊儲存在 HADB 中。HADB 支援 HTTP 階段作業、有狀態階段作業 Bean 的持續性和單次登入憑證。

JavaServer Faces 1.1 支援

Sun Java System Application Server Enterprise Edition 8.1 支援 JavaServer Faces 1.1 技術。JavaServer Faces 技術由一組伺服器端 API 組成，其表示了管理其狀態、事件、處理與輸入驗證的使用者介面組件。API 還定義了頁面導航並支援國際化和可存取性。您可以使用 JSP 自訂標籤檔案庫來增加自訂 UI 組件。

使用 JavaServer Faces 技術進行開發時，開發團隊的每個成員都可以專注於程序單獨的一部分。然後簡單的程式設計模型連結各部分，從而使開發週期更加有效、更加簡單。

硬體和軟體需求

本節列出了安裝 Sun Java System Application Server Enterprise Edition 8.1 產品前必須滿足的需求。

- [平台需求](#)
- [重要修補程式資訊](#)
- [JDBC 驅動程式和資料庫](#)
- [配置 Oracle](#)
- [配置 PointBase](#)
- [Web 伺服器](#)
- [瀏覽器](#)
- [高可用性的需求與限制](#)
- [HADB 檔案系統支援](#)
- [升級 Sun Java System Application Server](#)
- [切換到 J2SE 5.0](#)
- [其他需求](#)

平台需求

下表列出了 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 產品支援的作業系統。此外，還給出了安裝和執行 Application Server 了最小的和建議的記憶體需求。

表 2 Sun Java System Application Server 8.1 2005Q1 平台需求

作業系統	最小記憶體	建議的記憶體	最小磁碟空間	建議的磁碟空間	JVM
Sun Solaris 9、10 (SPARC) Solaris 9、10 (x86)	512 MB	1 GB	250 MB 可用 空間	500 MB 可用 空間	J2SE 1.4.2_06 J2SE 5.0
Redhat Enterprise Linux 2.1 Update 2、3.0 Update 1	512 MB	1 GB	220 MB 可用 空間	300 MB 可用 空間	J2SE 1.4.2_06 J2SE 5.0
Microsoft Windows 2000® Advanced Server Service Pack 4+	512 MB	1 GB	250 MB 可用 空間	500 MB 可用 空間	J2SE 1.4.2_06 J2SE 5.0
Microsoft Windows Server® 2003	512 MB	1 GB	250 MB 可用 空間	500 MB 可用 空間	J2SE 1.4.2_06 J2SE 5.0

在 UNIX 上，您可以使用 `uname` 指令檢查作業系統版本。可以使用 `df` 指令檢查磁碟空間。

Solaris 修補程式需求

建議 Solaris 9、10 (x86, SPARC) 使用者安裝「Sun 建議的修補程式叢集」。此修補程式叢集在此處的「建議的安全修補程式」中可以找到：

<http://sunsolve.sun.com/>

RedHat Enterprise Linux 3.0 附加套裝軟體需求

若要執行此產品的原生組件（包括安裝程式），則應該安裝以下套裝軟體（非標準 RedHat Enterprise Linux 3.0 發行軟體的一部分）：`compat-libstdc++-7.3-2.96.118.i386.rpm`

此套裝軟體可以從此處下載：

<http://mirrors.kernel.org/redhat/redhat/linux/9/en/os/i386/RedHat/RPMS/compat-libstdc++-7.3-2.96.118.i386.rpm>

重要修補程式資訊

如需目前 Sun Java System Application Server Enterprise Edition 8.1 所需的修補程式的清單，請移往 <http://sunsolve.sun.com> 並選取「Patches」或「Patch Portal」。請依循 Sun Java System Application Server Enterprise Edition 8.1 連結進行。由於作業系統修補程式需求發生變更和出現可用的 Java Enterprise System 組件的修補程式，在 SunSolve 上更新將最初以建議的修補程式叢集的形式出現。

JDBC 驅動程式和資料庫

Sun Java System Application Server 旨在支援任何具有對應 JDBC 驅動程式的 DBMS 的連接性。若需 Sun 已經測試並發現可用於建構 J2EE 相容的資料庫配置的組件之清單，請參考下表：

表 3 J2EE 相容的 JDBC 驅動程式

JDBC 供應商	JDBC 驅動程式類型	支援的資料庫伺服器
i-net Software	類型 4	Oracle (R) 8.1.7、9i、9.2.0.3 Sybase ASE 12.5.2 Microsoft SQL Server 2000 4.0 Service Pack 1
IBM	類型 2	IBM DB2 8.1 Service Pack 3+
PointBase	類型 4	PointBase Network Server 4.8
DataDirect	類型 4	Oracle (R) 8.1.7、9i、9.2.0.3 Sybase ASE 12.5.2 Microsoft SQL Server IBM DB2 8.1 Service Pack 3+
用於 Oracle 的 Sun Java System JDBC 驅動程式	類型 4	Oracle (R) 9.2.0.3、10G
用於 DB2 的 Sun Java System JDBC 驅動程式	類型 4	IBM DB2 8.1 Service Pack 3+
用於 Sybase 的 Sun Java System JDBC 驅動程式	類型 4	Sybase ASE 12.5.2
用於 Microsoft SQL Server 的 Sun Java System JDBC 驅動程式	類型 4	Microsoft SQL Server 2000 4.0 Service Pack 1
Oracle	類型 4、類型 2	Oracle (R) 9.2.0.3、10G

如需更多關於 i-net Software 的資訊，請參閱：

<http://www.inetsoftware.de/>

如需更多關於 DataDirect Technologies 的資訊，請參閱：

<http://www.datadirect.com/>

配置 Oracle

必須正確配置 Oracle JDBC 驅動程式，以使其與 J2EE 1.4 相容。請使用以下用於類型 2 和類型 4 的驅動程式的配置：

1. 請使用 9.2.0.3 或更高版本的 JDBC 驅動程式。
2. Oracle 資料庫需要在其參數 (init.ora) 檔案中具有 compatible=9.0.0.0.0 或更高的。
3. 使用 ojdbc14.jar 檔案。
4. 請配置 Application Server 來定義以下 JVM 特性：

```
-Doracle.jdbc.J2EE13Compliant=true
```

此外，對於 Type-2 驅動程式，ORACLE_HOME 和 LD_LIBRARY_PATH 變數 (必須包括 \$ORACLE_HOME/lib) 均需在啟動 Application Server 的環境中進行定義。例如，將其增加到 asenv.conf 檔案並確定將其匯出。

配置 PointBase

許多應用程式範例使用 Application Server 中包含的 PointBase 資料庫伺服器。使用 Application Server Enterprise Edition 時，必須在使用前配置 PointBase 資料庫伺服器。然而，使用 PointBase 與 Application Server 之前，請注意支援的配置組合。

表 4 支援的 J2SE/PointBase 組合

Application Server	PointBase
支援的	
J2SE 1.4	J2SE 1.4
J2SE 5.0	J2SE 1.4
不支援的	
J2SE 5.0	J2SE 5.0

有兩種方法可以配置 PointBase：

- 將 JAVA_HOME 環境變數設定為 J2SE 的位置。只有 J2SE 1.4.2 支援 Application Server 8.1 附帶的 PointBase 實作。
- 編輯 Application Server 的 PointBase 配置檔案。

若要使用第一種方法：

1. 請確定您安裝了要使用的 J2SE。

如果您還沒有 J2SE 1.4.2，則請下載 J2SE 1.4.2。

2. 使用您的作業系統和 Shell 相應的指令，將 `JAVA_HOME` 環境變數設定為安裝 J2SE 的目錄；例如：

```
% setenv JAVA_HOME "/opt/SUNWappserver/jdk"
```

若要使用第二種方法，則程序視作業系統而定。

Solaris 和 Linux

請編輯 `install_dir/pointbase/tools/serveroption/pbenv.conf` 配置檔案，將這一行：

```
PB_JAVA=%%PB_JAVA%%
```

變更為

```
PB_JAVA=J2SE_location
```

其中 `J2SE_location` 為安裝 J2SE 的目錄。如果您同時安裝 J2SE 與 Application Server，依預設，其安裝在 `install_dir/jdk`。變更後，您可以使用 `startserver` 程序檔啟動 PointBase。

Windows

請編輯 `install_dir\pointbase\tools\serveroption\pbenv.bat` 配置檔案，將這一行：

```
set PB_JAVA=%%PB_JAVA%%
```

變更為

```
set PB_JAVA=J2SE_location
```

其中 `J2SE_location` 為安裝 J2SE 的目錄。如果您同時安裝 J2SE 與 Application Server，依預設，其安裝在 `install_dir\j2se1.4`。進行此變更後，您可以透過執行 `startserver.bat` 來啟動 PointBase。

Web 伺服器

本節列出了 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 支援的 Web 伺服器。

表 5 支援的 Web 伺服器

Web 伺服器	版本	作業系統
Sun Java System Web Server	6.1+	Solaris SPARC 8、9、10 Solaris x86 9、10 Red Hat Enterprise Linux 2.1 Update 2、3.0 Update 1
Apache Web Server	1.3+, 1.4, 2.0	Solaris SPARC 9、10 Solaris x86 10 Red Hat Enterprise Linux 2.1 Update 2、3.0 Update 1

表 5 支援的 Web 伺服器 (續)

Web 伺服器	版本	作業系統
Microsoft IIS®	5.0+	Windows Server 2003 Windows 2000 Advanced Server Service Pack 4+

以後還將支援其他平台 (包括 Windows 和 HP-UX)。

瀏覽器

本節列出了 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 支援的瀏覽器。

表 6 支援的 Web 瀏覽器

瀏覽器	版本
Mozilla	1.4, 1.5, 1.6, 1.7.x
Netscape Navigator	4.79, 6.2, 7.0
Internet Explorer	5.5 Service Pack 2、6.0

高可用性的需求與限制

配置 Sun Java System Application Server 高可用性組件前，必須滿足以下高可用性的需求：

- HADB 最少需要 512 MB 記憶體，建議使用 1GB 記憶體，以使其與 Application Server 正常工作。
- HADB 僅支援 IPv4。
- 必須為 UDP 多重播送配置網路。
- 請勿對在 create domain、extend domain、hadbm create 或 hadbm addnodes 指令中使用的主機使用動態 IP 位址 (DHCP)。
- 如果在 Red Hat Linux 3.0 上執行 HADB，您必須安裝 Update 4 以避免作業系統過度交換的問題。請注意 AS 尚未在 Red Hat Update 4 環境中進行測試。

HADB 檔案系統支援

如果您要將 HADB 配置為使用以下某一檔案系統，則請注意以下重要事項：

- **ext2 和 ext3** — HADB 支援用於 Red Hat Application Server 3.0 的 ext2 和 ext3 檔案系統。對於 Red Hat Application Server 2.1，HADB 僅支援 ext2 檔案系統。
- **Veritas** — 當在 Solaris 平台上使用 Veritas 檔案系統時，訊息「WRN:Direct disk I/O mapping failed」會寫入歷程記錄檔案。此訊息表示 HADB 無法為資料和記錄裝置開啓直接 I/O。直接 I/O 是一種效能增強功能，可以降低 CPU 寫入磁碟頁面的消耗。它也會減少管理作業系統中的壞資料頁面的耗用時間。

若要同時使用直接 I/O 與 Veritas 檔案系統，請使用以下方法中的一種：

- 在透過選項 `mincache=direct` 掛載的檔案系統上建立資料和記錄裝置。此選項會套用至檔案系統上建立的所有檔案。請參閱 `mount_vxfs(1M)` 指令，以取得詳細資訊。
- 使用 Veritas Quick I/O 工具對檔案系統檔案執行原始 I/O。請參閱「*VERITAS File System 4.0 Administrator's Guide for Solaris*」，以取得詳細資訊。

請注意這些配置尚未在 Application Server 8.1 環境中進行測試。

請參考「*Sun Java System Application Server Enterprise Edition 8.1 Installation Guide*」，以取得關於與 Application Server 8.1 軟體一同安裝和配置 HADB 的詳細資訊。

升級 Sun Java System Application Server

請參考「安裝指南」，以取得關於將舊版 Application Server 升級為 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 的完整說明。

切換到 J2SE 5.0

Sun Java System Application Server 8.1 2005Q1 支援 J2SE 5.0 作為基礎 JVM，但是附帶的 PointBase 資料庫不支援。如果您要使用 J2SE 5.0 代替附帶的 J2SE 1.4.2 來執行 Application Server，請執行以下步驟 (Windows 和 Unix)：

1. 下載 J2SE 5.0 SDK (不是 JRE) 並安裝在您的系統上 (如果您尚未執行此作業)。
可以從 <http://java.sun.com/j2se/1.5.0/> 下載 J2SE 5.0 SDK。
2. 完全停止 Application Server。

您可以使用以下指令行：

```
as-install/bin/asadmin stop-domain
```

或使用管理主控台 GUI：

- a. 請按一下 [Application Server] 節點。
 - b. 按一下 [停止實例]。
3. 編輯 *install_dir/config/asenv.conf* 檔案 (Windows 中的 *asenv.bat*)，變更 `AS_JAVA` 的值以指向 J2SE 5.0 主目錄：
 4. 編輯 *as-install/samples/common.properties* 檔案，變更行起始處「`com.sun.aas.javaRoot...`」以參考 J2SE 5.0 主目錄。
 5. 重新啟動 Application Server。
`as-install/bin/asadmin start-domain`

備註 如果您要從比附帶的版本 (JDK 1.4.2_06) 更早的 JDK 版本進行升級，則您無法使用以上步驟將其升級為 J2SE 5.0。具體來說，除以上步驟外，您還必須刪除所有的網域然後再重新建立。

其他需求

安裝 Sun Java System Application Server 軟體之前，應該滿足以下的附加需求。

- **可用空間：**暫存目錄必須至少擁有 35MB 的可用空間用來安裝 Sun Java System Application Server，以及 250 MB 的可用空間用來安裝 SDK。
- **使用解除安裝程式：**如果您需要從系統中移除 Application Server，則使用軟體附帶的解除安裝程式是非常重要的。如果您嘗試使用其他方法，則在嘗試重新安裝相同的版本或安裝新的版本時將會出現問題。
- **可用的連接埠：**您必須擁有七個未使用的連接埠。
 - 安裝程式會自動偵測使用中的連接埠，並建議用作預設設定的未使用的連接埠。依預設，初始預設連接埠為 8080 (用於 HTTP)，8181 (用於 HTTPS)，和 4849 (用於管理伺服器)。
 - 安裝程式將偵測使用的連接埠並為您指定其他的兩個連接埠。[Sun Java™ System Message Queue](#) (依預設，為 7676) 和 IIOP (依預設，3700 用於 IIOP，1060 和 1061 用於 IIOP/SSL)。如果這些預設連接埠號正在使用中，解除安裝程式將在動態連接埠範圍內隨機指定連接埠號 (請注意其可能不是下一個可用的連接埠號)。
- **啟動先前安裝的伺服器 (UNIX)** — 除非您要取代先前安裝的伺服器，否則您必須在開始 Sun Java System Application Server 8.1 安裝程式之前將其啟動。這使安裝程式可以偵測使用中的連接埠並避免將其指定為其他用途。

- 取代先前安裝的伺服器 (UNIX) — 如果您安裝了舊版的 Sun Java System Application Server 並想用目前的 Application Server 取代它，則您應在安裝新的伺服器前將其停止。使用安裝程式升級精靈來升級伺服器。
- 關閉防火牆 (Microsoft Windows) — 您必須在安裝 Sun Java System Application Server 軟體前停止所有防火牆軟體，因為依預設某些防火牆軟體會停用所有連接埠。安裝程式必須可以確切地決定哪些連接埠是可用的。

如需進一步的相容性資訊，請參閱「升級與遷移指南」，網址為：

<http://docs.sun.com/app/docs/doc/819-0083>

獨立版本

Sun Java System Application Server Enterprise Edition 8.1 的獨立版本與 Java ES Enterprise Edition 版本在一些方面有所不同，具體來說：

- 任何使用者均可以安裝 8.1 獨立產品，而僅可以 root 使用者身份安裝 Java ES。
- HADB 組件在獨立版本中顯示為子組件，而在 Java ES 安裝中則為共用組件。
- 獨立版本將 Application Server 需要的全部共用組件安裝在一個安裝目錄下，而在 JES 中這些組件則安裝在不同的目錄下。
- 依預設，獨立安裝程式將 Application Server 的產品檔案、網域和配置資料儲存在一個單獨的目錄下，而在 Java ES 中則儲存在多個目錄下。
- 獨立版本允許在已經安裝相同或不同版本的 Application Server 系統上進行，無需解除安裝現有的安裝。透過在不同的版本之間或相同版本的不同實例之間保持唯一的安裝目錄，可以達到此目的。
- 獨立版本支援將現有的 Sun Java System Application Server Platform Edition 8.0 安裝或 Sun Java System Application Server Platform Edition 8.1 安裝升級到 Sun Java System Application Server Enterprise Edition 8.1 的「原地升級」。

相關說明文件

除本版本說明之外，Application Server 產品還包括整套的說明文件，均可以在此位置找到：

http://docs.sun.com/app/docs/coll/ApplicationServer8_pe_04q4 和
http://docs.sun.com/app/docs/coll/ApplicationServer8_pe_04q4_zh_TW

下表總結了 Application Server 核心應用程式說明文件集中包含的書籍。

表 7 此文件集中包含的文件

書名	描述
快速入門指南	如何開始使用 Sun Java System Application Server 產品。
<i>Installation Guide</i>	安裝 Sun Java System Application Server 軟體及其組件。
<i>Deployment Planning Guide</i>	評估系統需求和企業狀況，確保以最適合您的站點的方式部署 Sun Java System Application Server。此外還描述了部署應用伺服器時應該注意的常見問題。
<i>Developer's Guide</i>	建立和實作要在 Sun Java System Application Server 上執行的 Java™ 2 Platform, Enterprise Edition (J2EE™ 平台) 應用程式，這些應用程式遵循針對 J2EE 元件和 API 的開放式 Java 標準模型。其中包括有關開發者工具、安全性、組合、部署、除錯和建立生命週期模組的一般資訊。
<i>J2EE 1.4 Tutorial</i>	使用 J2EE 1.4 平台技術和 API 開發 J2EE 應用程式，並將這些應用程式部署到 Sun Java System Application Server。
管理指南	從管理主控台配置、管理和部署 Sun Java System Application Server 子系統和組件。
<i>High Availability Administration Guide</i>	高可用性資料庫安裝後的配置和管理說明。
<i>Administration Reference</i>	編輯 Sun Java System Application Server 的配置檔案 domain.xml。
<i>Upgrade and Migration Guide</i>	將應用程式遷移到新的 Sun Java System Application Server 程式設計模型，特別是從 Application Server 6.x 和 7 進行遷移。該指南還介紹了可導致與產品規格不相容的相鄰產品版本和配置選項之間的差異。
<i>Performance Tuning Guide</i>	微調 Sun Java System Application Server 以改善效能。
<i>Troubleshooting Guide</i>	解決 Sun Java System Application Server 問題。
<i>Error Message Reference</i>	解決 Sun Java System Application Server 錯誤訊息。
<i>Reference Manual</i>	可用於 Sun Java System Application Server 的公用程式指令，以線上說明手冊樣式編寫。其中包括 asadmin 指令行介面。

已知問題和限制

本節介紹了 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 軟體的已知問題及相關的解決方法。如果摘要敘述未指明特定的平台，則所有平台都可能出現此問題。這些資訊按以下章節進行分類：

- [管理](#)
- [應用程式用戶端](#)
- [附帶的 Sun JDBC 驅動程式](#)
- [連接器](#)
- [容器管理式的持續性](#)
- [Deploytool](#)
- [說明文件](#)
- [高可用性](#)
- [安裝](#)
- [J2EE 指導](#)
- [生命週期管理](#)
- [記錄](#)
- [Message Queue](#)
- [監視](#)
- [PointBase](#)
- [範例](#)
- [安全性](#)
- [升級公用程式](#)
- [Web 容器](#)

管理

本節介紹已知的管理問題以及相關的解決方案。

如果沒有 domain1，則 package-appclient 程序檔無法工作。(ID 6171458)

依預設，在 \$INSTALL/lib/package-appclient.xml 中，會有一個用於由 asenv.conf 指向的 domain1 的變數 AS_ACC_CONFIG 的硬編碼的值。如果刪除 domain1 並建立新的網域，則 AS_ACC_CONFIG 變數不會與新的網域一起更新，這將導致 package-appclient 程序檔失敗。

解決方案

執行以下作業之一：

- 保持 domain1 不變，並在其周圍建立其他的網域。
- 移除 domain1，並用新的網域名稱取代 \$INSTALL/lib/package-appclient.xml 中的 domain1 的硬編碼值。如果沒有 domain1，則每次建立新的網域均必須執行此作業。

無法使用其他名稱復原備份的網域。(ID 6196993)

無法使用 backup-domain 和 restore-domain 指令執行在同一 Application Server 安裝上的網域的鏡像，因為網域無法使用與原始名稱不同的名稱進行復原，即使 asadmin restore-domain 指令提供了重新命名的選項。重新命名備份的網域表面上成功了，但啟動重新命名的網域的嘗試會失敗，因為網域配置中的項目未變更，並且 startserv 和 stopserv 仍使用原始的網域名稱來設定路徑。

解決方案

用於 restore-domain 的網域名稱必須與用於原始 backup-domain 指令的名稱相同。Application Server 8.1 中的 backup-domain 和 restore-domain 指令僅適用於在相同的機器上備份和儲存相同的網域。

不支援附帶附加的 JMX 代理程式啟動 Application Server。(ID 6200011)

J2SE 1.4.x、5.0 或更高版本可以在 Application Server 上進行配置。啟動 JMX 代理程式是 J2SE 5.0 平台不可或缺的功能。如果您在伺服器啟動時明確設定系統特性，則會啟動此功能。

範例值包含：

```
name="com.sun.management.jmxremote" value="true"  
name="com.sun.management.jmxremote.port" value="9999"  
name="com.sun.management.jmxremote.authenticate" value="false"  
name="com.sun.management.jmxremote.ssl" value="false"
```

配置 JMX 特性並啟動伺服器後，新的 jmx-connector 伺服器將在 Application Server VM 中啟動。其副作用就是管理功能會受到不利影響，且 Application Server 管理 GUI 和 CLI 可能產生非預期的結果。問題在於內建 jmx-connector 伺服器與新的 jmx-connector 伺服器之間存在衝突。

解決方案

如果使用 jconsole (或任何其他與 JMX 相容的用戶端)，請考量重新使用與 Application Server 啓動一同啓動的標準 JMX 連接器伺服器。

當伺服器啓動時，類似於下面顯示的行會顯示在 server.log 中。您可以在成功提供憑證後連線至其中指定的 JMXServiceURL 並執行相同的管理 / 配置作業；例如：

```
[#|2004-11-24T17:49:08.203-0800|INFO|sun-appserver-ee8.1|javax.enterprise.system.tools.admin|_ThreadID=10;|ADM1501: 以下是用於 JMXConnectorServer 的 JMXServiceURL:
[service:jmx:rmi:///jndi/rmi://hostname:8686/management/rmi-jmx-connector]。這是遠端管理用戶端應使用 JSR 160 JMX 連接器連線的位置。|#]
```

如需更多資訊，請參閱「*Sun Java System Application Server 8.1 Administration Guide*」。

嚴格限制對 Application Server start 和 stop 程序檔的執行許可權。(僅限於 UNIX) (ID 6206176)

如果您以使用者「A」的身份登入時執行 asadmin restore-domain 指令，程序檔將結束作為 744 (rwxr--r--) 的許可權。如果您在以後嘗試以使用者「B」的身份啓動或停止網域 (即使「B」是根)，則會因為程序檔僅可由「A」執行而失敗。

解決方案

變更程序檔的權限：

```
chmod 755 <appserv>/domains/<domain-name>/bin/*
```

應用程式用戶端

本節介紹已知的應用程式用戶端問題以及相關的解決方案。

封裝在應用程式用戶端歸檔檔案中的程式庫 JAR 會覆寫可取得的檔案。(ID 6193556)

如果您在用戶端 JAR 內部具有頂層的 JAR 檔案 (這種情況下，為 reporter.jar)，當您部署用戶端 JAR 時，該 JAR 的可取得的檔案將覆寫用戶端 JAR 的可取得的檔案。

解決方案

目前尚無解決方案。

附帶的 Sun JDBC 驅動程式

本節介紹已知的附帶的 Sun JDBC 驅動程式問題以及相關的解決方案。

如果兩個並列作業事件正在執行並且其中一個被轉返，則當使用預備敘述來更新時，使用的 TRANSACTION_SERIALIZABLE 隔絕層級與附帶 (用於 Microsoft SQL Server 的) Sun 驅動程式的應用程式可能會當機。(ID 6165970)

若要為連線設定所需的隔絕層級，必須在相同的隔絕層級建立相應的連線區。請參閱「*Application Server 8.1 2005Q1 Administration Guide*」，以取得有關配置連線區的詳細資訊。

解決方案

無。

PreparedStatement 錯誤 (ID 6170432)

描述 1

如果應用程式在一個作業事件中產生 PreparedStatement 物件多於 3000 個，則 DB2 可能發生以下錯誤：

```
[sunm] [DB2 JDBC Driver] 無其他可用的描述。請使用更大的動態區段值重新建立封裝。
```

解決方案 1

將以下特性增加到連線區定義中，以使用更大的動態區段值來使驅動程式重新連結 DB2 封裝：

```
createDefaultPackage=true  
replacePackage=true  
dynamicSections=1000
```

請參閱「*Application Server 8.1 2005Q1 Administration Guide*」，以取得有關配置連線區的詳細資訊。

描述 2

與以上 PreparedStatement 錯誤相關，可能擲回的另一個錯誤訊息是：

```
[sunm] [DB2 JDBC Driver] [DB2] 虛擬儲存或資料庫資源不可用
```

解決方案 2

增大 DB2 伺服器配置參數 APPLHEAPSZ。合適的值為 4096。

描述 3

隔絕層級 TRANSACTION_SERIALIZABLE。

如果您的應用程式使用的是隔絕層級 TRANSACTION_SERIALIZABLE 並使用以上建議的參數中的一個，則其在取得連線時可能當機。

解決方案 3

若要為連線設定所需的隔絕層級，必須在此隔絕層級建立相應的連線區。請參閱「*Application Server 8.1 2005Q1 Administration Guide*」，以取得說明。

在 Application Server Enterprise Edition 8.1 上，附帶的 DB2 Sun JDBC 驅動程式在預設配置中不起作用。這是因為 DB2 JDBC 驅動程式類別需要明確的 charsetProvider RuntimePermission。(ID 6183492)

解決方案

修改 server.policy 檔案，為使用 JDBC 驅動程式的部署的應用程式提供以下權限：

```
grant codeBase "file:${DEPLOYED_APPLICATION_DIR}" { permission java.lang.RuntimePermission
    "charsetProvider";
};
```

請注意，只有附帶的 DB2 Sun JDBC 驅動程式需要進行此作業。

使用附帶的 Sun 驅動程式設定隔絕層級的問題 (針對 Sybase Adaptive Server)。(ID 6189199)

- 如果兩個並列作業事件正在執行並且其中一個被轉返，則當使用預備敘述來更新時，使用附帶的 Sun 驅動程式 (用於 Sybase Adaptive Server) TRANSACTION_SERIALIZABLE 隔絕層級的應用程式可能會當機。連線轉返失敗，並顯示以下訊息，且轉返的連線無法再被使用：

```
java.sql.SQLException: [sunm] [Sybase JDBC Driver] 由於連線的原因，無法提交請求
```

- Sybase Adaptive Server 不支援 TRANSACTION_REPEATABLE_READ 隔絕層級。然而，查詢 DatabaseMetaData，附帶的 Sun 驅動程式會傳回資料庫支援此隔絕層級。使用此隔絕層級的應用程式將失敗。
- 使用附帶的 Sun 驅動程式的應用程式無法設定 TRANSACTION_READ_UNCOMMITTED 隔絕層級。在第一次存取 DataBaseMetaData 時，應用程式會擲回以下異常：

```
java.sql.SQLException: [sunm] [Sybase JDBC Driver] [Sybase] 最佳化程式無法找到唯一的索引，此索引可用於在表格「sybssystemprocs.dbo.spt_server_info」中執行隔絕層級掃描。
```

解決方案

無。

連接器

本節介紹已知的 J2EE 連接器架構問題以及相關的解決方案。

在 `jdbc-connection-pools` 中，無法動態重新配置 `connection-validation`。
(ID 4930792)

建立 JDBC 連線區後，無法動態重新配置其 `is-connection-validation-required` 屬性。這表示對於已經建立的區連線，無法在過程中將驗證切換為 `on` (或 `off`)。對於區的 `validation-method` 屬性同樣適用。

解決方案

依對執行應用程式的侵入的遞減順序，有三個可能的解決方法：

1. 在將 `validation` 切換為 `on` 的情況下，建立 `jdbc-connection-pools`。
2. 在將 `validation` 切換為 `on` 的情況下，刪除 `jdbc-connection-pool` 並重新建立。
這樣最多只會影響依賴此特定區的少數部署的應用程式。
3. 變更 `validation` 特性並重新啓動 `Application Server`。
這將影響所有部署的應用程式 (因為執行了重新啓動)。

重新啓動 DAS 實例後，在將重疊設定為 `false` 時，取消部署連接器模組會失敗。
(ID 6188343)

在此情況下，獨立的或內嵌的連接器模組便部署在 DAS 和連接器連線區中，並為部署的模組建立了資源。重新啓動 DAS 實例後，將重疊設定為 `false` 時，取消部署連接器模組會失敗，並顯示以下異常：

```
[#|2004-10-31T19:52:23.049-0800|INFO|sun-appserver-ee8.1|javax.enterprise.system  
.core|_ThreadID=14;|CORE5023: 卸載應用程式 [foo] 時發生錯誤|#]
```

解決方案

在重新啓動 DAS 實例後，使用重疊的取消部署 (將 `cascade` 選項設定為 `true`) 來取消部署獨立的和內嵌的連接器。

容器管理式的持續性

本節介紹已知的容器管理式的持續性問題以及相關的解決方案。

如果 where 子句包含 OR 運算子和具有單一值的 cmr 導航，則 EJBQL 詢問可能不包含所有相符的結果。(ID 6184864)

如果 EJBQL 詢問中的 where 子句包含 OR 運算子和具有單一值的 cmr 導航，則詢問結果不會包含在列中航路徑設定為 null 的列的結果 (即使導航路徑位於不同的 OR 子句中)。

例如，考量組成 Employee、Department 和 Insurance 的綱目。Employee 與 Department 具有一對多的關係，與 Insurance 具有一對一的關係：

```
select Distinct Object(e) from Employee e
       where e.name = 'John' OR e.department.name = 'Engineering'
```

以上詢問將不會傳回名為 John 且不屬於任何部門的員工。

```
select Distinct Object(e) from Employee e
       where e.department.name = 'Engineering' OR e.insurance.name = 'xyz'
```

以上詢問將不傳回保險名稱為 xyz 且不屬於任何部門的員工。也不會傳回部門名稱為 Engineering 且無保險的員工。

解決方案

為每個 OR 條件分別執行詢問並合併結果。

Deploytool

本節介紹已知的 Deploytool 問題以及相關的解決方案。

在以下 Sun 部署描述元中，Deploytool 通常不會建立 message-destination 元素：(ID 6197393)

- sun-application-client.xml
- sun-ejb-jar.xml
- sun-web.xml

被指定為 [訊息目標] 標籤中的 JNDI 名稱的 JMS 目標資源可能不會儲存到 Sun 描述元中。指定目標名稱後 (例如：PhysicalQueue，create-jmsdest 將建立實際的目標)，並按下 Enter，在顯示名稱下面會顯示目標名稱，且在提供者清單中顯示用戶端或 Bean 名稱。在 Sun 特定的 [JNDI 名稱] 文字欄位中鍵入「jms/Queue」，並按下 Enter，在標題列中應用程式不會顯示為「(changed)」，且錯誤訊息會被寫入到 ~/.deploytool/logfile 中。在儲存應用程式和返回到該標籤時，[JNDI 名稱] 欄位會重新變為空白。在使用 [工具]>[描述元檢視器]>[Application Server 描述元] 檢視 Sun 描述元時，<jndi-name> 元素內未建立 <message-destination> 元素。

問題在於在 Deploytool 階段作業中，首次為訊息目標 JNDI 名稱輸入值，此值在 Sun 描述元中看起來正確，但 org.netbeans.modules.schema2beans.BeanProp.setElement() 卻擲回 IllegalArgumentException。以後對在相同的或其他的應用程式中的訊息目標 JNDI 名稱所做的變更或增加將不儲存到 Sun 描述元中。

解決方案

若要編輯現有的訊息目標的 JNDI 名稱，請執行以下步驟：

1. 請保持 [JNDI 名稱] 文字欄位空白並按下 Enter，刪除現有的 JNDI 名稱。
2. 鍵入新的 JNDI 名稱並按下 Enter。
3. 按一下 [工具]>[描述元檢視器]>[Application Server 描述元]，複查 Sun 描述元。
4. 按一下 [檔案]>[檔案]，儲存應用程式。

如果 JNDI 名稱未儲存到 Sun 描述元中：

1. 重新啟動部署工具。
2. 在 [訊息目標] 標籤上，選取一個訊息目標或增加一個新的訊息目標。
3. 在 [Sun 特定的 JNDI 名稱] 文字欄位中輸入訊息目標的 JNDI 名稱，然後按下 Enter 鍵。
4. 按一下 [工具]>[描述元檢視器]>[Application Server 描述元]，來複查 Sun 描述元。
5. 按一下 [檔案]>[儲存]，以儲存應用程式。

除非在部署工具階段作業過程中已經在 [JNDI 名稱] 文字欄位中輸入了一個值，否則請在每次需要在 [訊息目標] 標籤上的 [Sun 特定的 JNDI 名稱] 中輸入值時，重複上述步驟。

新的 Web 服務精靈中損壞的面板 (ID 6198981)

該問題表明兩組徵兆：

- 在 WSDL 檔案面板上按 [下一步] 後未前進至下一個面板，而是拋出異常：
java.lang.NoClassDefFoundError: com/sun/org/apache/xpath/internal/XPathAPI

- 在 [建立封裝的端點] 面板上，選取 [建立 Web 服務端點並封裝]。在 [後續步驟] 面板上按一下 [完成] 後未在輸出目錄中建立 [端點] 模組，而是拋出異常：

```
java.lang.NoClassDefFoundError: com/sun/org/apache/xpath/internal/XPathAPI。
```

問題在於包含 XPathAPI.class 的 xalan.jar 從 CLASSPATH 中遺失。請注意 JDK 5.0 不存在此問題。

解決方案

將 xalan.jar 增加至 CLASSPATH includes，以用於 *install_dir/lib/processLauncher.xml* 的 *s1as-deploytool* 程序；例如：

```
includes="appserv-assemblytool.jar,activation.jar,appserv-admin.jar,appserv-cmp.jar,appserv-rt.jar,j2ee.jar,jaxrpc-impl.jar,appserv-ext.jar,deployhelp.jar,admin-cli.jar,dom.jar,xercesImpl.jar,xalan.jar"
```

簡體中文的部署工具，將「Home」錯誤地翻譯為「安裝目錄」。(ID 6203658)

如果您使用部署工具建立一個企業 Bean，然後導覽至 Bean 節點的 [作業事件] 標籤或 [安全性] 標籤，會將「Local Home」和「Remote Home」標籤錯誤地翻譯為「本機安裝目錄」與「遠端安裝目錄」。

說明文件

本節描述已知的文件問題以及相關的解決方案。

index.html 與 QuickStart.html 文件中的錯誤 (ID 6193749)

Application Server 8.1 文件集中的 index.html 檔案與 docs-ee/QuickStart.html 檔案中有兩組錯誤。

- 複製到每個網域的 docroot 目錄的預設 index.html 頁面顯示的路徑不正確，應該為：

```
as_instance/domains/domain_name/docroot/
```

而非

```
install_dir/domains/domain_name/docroot/index.html
```

- 「快速入門指南」提供的負載平衡程式與 Web 伺服器的安裝說明不正確。如需更新的說明，請參閱線上「快速入門指南」，位於：

<http://docs.sun.com/app/docs/doc/819-1003>

請注意，這僅影響 Application Server 8.1 的 Java ES Enterprise Edition 版本，而不影響獨立版本。

- `asadmin create-domain --help` 指令產生錯誤用法，並且無效選項已被歸檔 (`--admin.jmxport`)。(ID 6207862)

`asadmin create-domain` 的 `help` 指令描述對該指令而言為無效選項的 `--admin.jmxport`。

解決方案

`--admin.jmxport` 無法與 `asadmin create-domain` 指令配合使用。

Javadoc 不一致 (ID 不同)

用於多個 AMX 介面與方法的 Javadoc 缺漏或不正確：

- `ConnectorConnectionPoolStats` 與 `AltJDBCConnectionPoolStats` 中缺少 `NumConnAcquired` 與 `NumConnReleased` 統計資料的獲取方法。這些獲取方法將增加至未來的版本中，如 `getNumConnAcquired()` 和 `getNumConnReleased()`。
- 在 `EJBCacheStats` 中呼叫以下方法將拋出異常：`getPassivationSuccesses()`、`getExpiredSessionsRemoved()`、`getPassivationErrors()` 和 `getPassivations()`。這將在未來的版本中進行修正。
- 啟動伺服器後，AMX MBeans 可能需要數秒鐘才能完全註冊和使用。未來的版本將可能確定完全載入 AMX MBeans 的時間。
- 常數 `XTypes.CONNNECTOR_CONNECTION_POOL_MONITOR` 拼字錯誤 (「NNN」)。這將在未來的版本中進行校正。

高可用性

本節描述已知的高可用性資料庫 (HADB) 問題以及相關的解決方案。

將 Apache 與「管理指南」中「編譯和配置 Apache Web Server」附錄中的 Sun Java System Application Server 配合使用的需求已過期。

下面列示的是將 Apache Web 伺服器軟體與 HADB 配合使用的軟體需求。

- 適用於 Apache 1.3 的需求
 - `openssl-0.9.7e` (源代碼)
 - `mod_ssl-2.8.16-1.3.33` (源代碼)
 - `apache_1.3.33` (源代碼)

- gcc-3.3-sol9-sparc-local 套裝軟體 (用於 Solaris 9 SPARC/ x86)
- gcc-3.3-sol9-intel-local 套裝軟體 (用於 Solaris 9 x86)
- flex-2.5.4a-sol9-sparc-local 套裝軟體 (用於 Solaris 9 SPARC)
- flex-2.5.4a-sol9-intel-local 套裝軟體 (用於 Solaris 9 x86)
- 適用於 Apache 2.0 的需求
 - openssl-0.9.7e (源代碼)
 - httpd-2.0.49 (源代碼)
 - gcc-3.3-sol9-sparc-local 套裝軟體 (用於 Solaris 9 SPARC)。
 - gcc-3.3-sol9-intel-local 套裝軟體 (用於 Solaris 9 x86)
 - flex-2.5.4a-sol9-sparc-local 套裝軟體 (用於 Solaris 9 SPARC)
 - flex-2.5.4a-sol9-intel-local 套裝軟體 (用於 Solaris 9 x86)

編譯之前還需要一個附加步驟。在 Solaris 10 平台上，請在執行 OpenSSL 的 make 之前，執行以下指令：

```
/usr/local/lib/gcc-lib/sparc-sun-solaris2.9/3.3/install-tools/mkheaders。
```

最後，有關您在安裝後必須對 Apache web 伺服器進行的修改的詳細資訊 (在「管理指南」中的「配置負載平衡和防故障備用」中) 是不完整的。

在所有平台上

1. 在 `apache_install_dir` 下建立一個名為 `sec_db_files` 的目錄。
2. 將 `domain/config/*.db` 複製到 `apache_install_dir/sec_db_files`。

在 Solaris 平台上：

- 將路徑 `/usr/lib/mps/secv1` 增加至 `apache_install_dir/bin/apachectl` 程序檔中的 `LD_LIBRARY_PATH` 中。此路徑必須增加在 `/usr/lib/mps` 的前面。

在 Linux 平台上

- 將路徑 `/opt/sun/private/lib` 增加至 `apache_install_dir/bin/apachectl` 程序檔中的 `LD_LIBRARY_PATH` 中。此路徑必須增加在 `/usr/lib` 的前面。

具有雙網路的 HADB 配置 (無 ID)

在兩個子網路上配置為具有雙網路的 HADB，在 Solaris SPARC 上工作正常。然而，我們發現由於作業系統的問題或同一硬體平台上的網路驅動程式，Solaris x86 與 Linux 平台並不總能正確處理雙網路。這將引起 HADB 的以下問題：

- 在 Linux 上，在傳送訊息時會封鎖某些 HADB 程序。這將引起 HADB 節點重新啟動和網路分割。
- 在 Solaris x86 上，網路失敗後會出現一些問題，這會阻止切換至其他網路介面。這種情況不會經常發生，因此最好還是具有兩個網路。這些問題中的一部分在 Solaris 10 中得到解決。
- 不支援幹線。
- HADB 在 Windows 2003 上不支援雙網路 (ID 5103186)。

增加新節點後建立的新表格不會在增加的節點上分段。(ID 5042351)

如果您建立資料庫實例，然後在其上增加節點，則以後建立的所有新表格均不會在資料庫建立之後增加的節點上分段。僅在 `addnodes` 指令之前建立的表格在 `hadbm addnodes` 使其重新分段後才能使用增加的節點。這是因為 `create table` 所使用的 `sysnode` 節點群組是在執行 `hadbm create` 時建立的。

解決方案

在增加所有的新表格後執行 `hadbm refragment`，或在節點群組 `all_nodes` 上建立新表格。

不支援 `packagepath` 的不同路徑。(ID 5091349)

不可能在不同主機的不同位置使用相同名稱註冊同一套裝軟體，例如：

```
hadbm registerpackage test --packagepath=/var/install11 --hosts europa11
套裝軟體註冊成功。
hadbm registerpackage test --packagepath=/var/install12 --hosts europa12
hadbm:Error 22171: 已用過該套裝軟體名稱測試註冊套裝軟體。
```

解決方案

HADB 不支援資料庫叢集中跨節點的不同路徑。請確定所有參與的主機的 HADB 伺服器安裝目錄 (`--packagepath`) 均相同。

`hadbm set` 不檢查資源可用性 (磁碟與記憶體空間)。(ID 5091280)

當使用 `hadbm set` 增加裝置或緩衝區大小時，管理系統會在建立資料庫或增加節點時檢查資源可用性，但不會檢查裝置或主記憶體緩衝區大小變更時是否有足夠的可用資源。

解決方案

在增加任何 `devicesize` 或 `buffersize` 配置屬性之前，驗證所有主機上是否有足夠的可用磁碟 / 記憶體空間。

在存在負載時，RedHat AS 3.0 處於同位模式時出現 HADB 問題。(ID 6158393)

HADB 與 AS 同位在 RedHat 3.0 上執行。作業事件可能中斷並影響效能。這是由作業系統執行的過度交換引起的。

解決方案

該問題已在 Red Hat EL 3.0 Update 4 中得到修正。HADB 已經在 RedHat 3.0 update 4 環境中進行測試，並證實由作業系統執行的過度交換已經消失。請注意，Application Server 8.1 尚未在 update 4 環境中進行測試。

`configure-ha-cluster` 指令可能當機。(ID 6159633)

當使用 `asadmin configure-ha-cluster` 指令在多台主機上建立或配置高度可用的叢集時，指令有時會當機。HADB 管理代理程式或 Application Server 不會拋出異常。

解決方案

HADB 不支援資料庫叢集中跨節點的不同路徑。請確定所有參與的主機的 HADB 伺服器安裝目錄與配置目錄均相同。請確定在再次執行指令之前，清除儲存庫目錄。

使用 HADB 的 Application Server 的效能下降 (ID 6172589)

在所有平台上，配置為使用 HADB 的 Application Server 實例的效能將不及舊版的效能，這是因為對 HADB 所使用的 JDBC 驅動程式進行了變更。

解決方案

請立即聯絡 Sun 服務以獲得解決方法。

第二個 `addnodes` 在重新分段過程中失敗。(ID 6175436)

第二個 (及後續的) `addnodes` 指令可能會在重新分段過程中失敗，並出現以下錯誤：

```
hadbm:Error 22042: 無法將資料庫分段。請使用 hadbm 重新分段指令重試以將資料庫重新分段。  
原因: HADB-E-11747: Nodegroup all_nodes 已經存在
```

解決方案

使用 `hadbm` 重新分段來手動將表格重新分段。

無法在 Windows 上建立大於 2GB 的資料裝置。(ID 6181845)

如果您將 `hadbm create` 或 `hadbm set` 與 `--NumberOfDataDevices=1` (預設) 和值大於 2GB 的 `--devicesize` 配合使用，則會出現以下錯誤：

```
DEVINIT-ERROR: out of space, wrote -2147479552 B of -2036330496 B
嘗試將檔案指標移至文件開始之前。
```

解決方案

如果您需要在 Windows 中建立大於 2GB 的資料裝置，請用所需的 `devicesize` 除以 2 GB，得到您需要的裝置數目。然後，使用 `--NumberOfDataDevices` 選項，根據計算建立多個資料裝置。例如，如果您需要建立一個 5GB 的資料裝置，則：

$$5 \div 2 = 2.5$$

然後應對計算得到的數字進行上捨入，設定 `--NumberOfDataDevices=3`。

hadbm 說明中的資訊已過期 (ID 6190702)

hadbm 說明系統中的某些資訊已過期。

解決方案

請參閱「*Application Server 8.1 管理員指南*」中的 HADB 章節，以取得最新資訊。

Addnodes 指令用於表格時失敗，未找到錯誤 (ID 6214601)

在此方案中，`hadbm` 重新分段指令由於以下錯誤失敗：

```
hadbm: 錯誤 22042：無法將資料庫分段。請使用 hadbm 重新分段指令重試以將資料庫重新分段。原因：
HADB-E-11701:* 未找到表格 Singlesignon*
```

解決方案

解決方法是使用 `clusql` 指令手動將 App Srv 表格重新分段：

```
> clusql <server:port list> system+<dbpassword specified at database create>
SQL: set autocommit on;
SQL: set schema haschema;
SQL: alter table sessionattribute nodegroup all_nodes;
SQL: alter table singlesignon nodegroup all_nodes;
SQL: alter table statefulsessionbean nodegroup all_nodes;
SQL: alter table sessionheader nodegroup all_nodes;
SQL: alter table blobsessions nodegroup all_nodes;
SQL: quit;
```

安裝

本節描述已知的安裝問題以及相關的解決方案。

間歇性失敗使安裝程式與解除安裝程式 [歡迎] 螢幕上的 [下一步] 導航按鈕無效。此問題僅影響產品的獨立版本。(ID 4977191)

此問題會間歇性地出現在 Solaris x86 平台上，但可能還影響 Solaris SPARC 與 Linux 平台。

問題是，安裝程式與解除安裝程式的第一螢幕正確顯示完整文字與 [說明] 和 [取消] 按鈕，但導覽至下一螢幕所必需的 [下一步] 按鈕不可見。儘管按鈕不可見，但其區域仍在作用中，且如果在其上按一下，會繼續正常導覽至下一螢幕。問題的原因是間歇性 J2SE GUI 重新繪製問題。

解決方案

解決方法之一是按一下就在 [說明] 按鈕左側的 [下一步] 按鈕區域。另一個解決方案是，透過微調螢幕大小或最小化和復原安裝程式視窗來強制重新繪製螢幕。重新繪製後，遺失的 [下一步] 按鈕將變為可見的。

按一下 [完成] 按鈕之後，某些 Linux 系統上的安裝關閉當機。(5009728)

已在多個 Linux 系統中發現此問題。在 Java Desktop System 2 中最常見，但還見於 RedHat 發行軟體中。

在最後的安裝程式螢幕上按一下 [完成] 按鈕之後，安裝程式無法啟動包含產品 [關於] 頁面或產品註冊頁面的瀏覽器視窗，且無限期當機，並不返回指令提示。

解決方案

在啟動安裝程式的終端機視窗中按下 **Ctrl+C** 退出安裝程式。執行完此步驟後，有時會啟動包含產品 [關於] 頁面或註冊頁面的瀏覽器視窗，但如果未顯示該視窗，請啟動瀏覽器並輸入以下 URL 以檢視 [關於] 頁面：

```
file://install_dir/docs-ee/about.html
```

如果您還選取安裝選項以註冊產品，請使用產品 [關於] 頁面上的連結進入註冊頁面。

Linux 上安裝包裝函式中的間歇性 J2SE 偵測與啟動程式問題。(6172980)

用來啟動 Linux 安裝程式的可執行 `setup` 有時當機。包裝函式會當機並傳回以下訊息，而不是解析 J2SE 位置並啟動安裝精靈：

```
正在檢查可用磁碟空間 ....
正在檢查 Java (TM) 2 執行環境 ....
正在擷取 Java (TM) 2 執行環境 ....
正在刪除暫存檔 .....
```

該問題僅在某些 Linux 版本中才會見到，並且似乎取決於環境設定，尤其是 `JAVA_HOME` 變數。

解決方案

若要解決此問題，請：

1. 根據您的 shell，透過執行 `unset` 或 `unsetenv` 來取消 `JAVA_HOME` 變數設定。
2. 使用 `-javahome` 選項執行 `setup` 以指定由安裝程式使用的 `JAVA_HOME`。

需要在安裝過程中建立 `imq` 目錄 (僅 Windows)。(ID 6199697)

在 Windows 上安裝 Application Server EE 後，IMQ 代理程式立即啟動失敗，並顯示目錄 `drive:\astomainstomain1\imq` 不存在的訊息。

請注意，如果代理程式在啟動 `domain1` 之後啟動，則會由 Application Server 建立目錄，且不會出現問題。

解決方案

建立代理程式之前建立 `var_home_dir_location`：

```
$imqbrokerd -varhome var_home_dir_location
```

例如：

```
$imqbrokerd -varhome D:\astomainstomain1\imq
```

J2EE 指導

若要在 Sun Java System 上執行 J2EE 1.4 Tutorial，Application Server Enterprise Edition 8.1 2005Q1 會執行以下工作：

- 當您編輯「關於本指導」一章中「關於範例」一節中所描述的檔案範例 `/common/build.properties` 時，將連接埠 `4848` 變更爲 `4849`。
- 使用部署工具時，會在部署範例之前增加伺服器本地主機：`4849`。
- 當使用 [管理主控台] 建立任何來源時，請使用 [目標] 標籤將伺服器指定爲目標。如果使用指令行或 `asant` 目標，則伺服器爲預設目標，無需其他動作。

生命週期管理

本節描述已知的生命週期管理問題以及相關的解決方案。

將 `ejb-timer-service` 特性 `minimum-delivery-interval` 設定為 9000 之後，嘗試將 `ejb-timer-service` 特性 `redelivery-interval-in-millis` 設定為 7000 會導致 `set` 指令失敗，並出現以下錯誤：(ID 6193449)

```
[echo] Doing admin task set
[exec] [Attribute(id=redelivery-interval-internal-in-millis) : Redelivery-Interval (7,000)
應大於或等於 Minimum-delivery-interval-in-millis (9,000)]
[exec] CLI137 指令設定失敗。
```

- `minimum-delivery-interval` 是同一計時器的兩個傳送之間的最小間隔持續時間。
- `redelivery-interval-in-millis` 是計時器服務在 `ejbTimeout` 失敗後嘗試重新傳送之前的等待時間。

問題在於，將重新傳送間隔特性與最小傳送特性相關係的邏輯不正確，並阻止您使用 GUI 或 CLI 設定任何最小傳送間隔大於重新傳送間隔的值。

必須總是將 `minimum-delivery-interval-in-millis` 設定為等於或高於 `ejb-timer-service` 特性 `redelivery-interval-in-millis`。問題在於，Application Server 中有錯誤驗證檢查，它可驗證 `redelivery-interval-in-millis` 的值是否大於 `minimum-delivery-interval-in-millis` 的值。

解決方案

使用這些特性的預設值，如下所示：

```
minimum-delivery-interval(default)=7000
redelivery-interval-in-millis(default)=5000
```

這些預設值以外的值將會產生錯誤。

記錄

本節描述已知的記錄問題以及解決方案。

設定 `access, failure` 的除錯描述會導致 Application Server 啟動當機。(ID 6180095)

設定 JVM 的 `java.security.debug` 選項將會導致伺服器實例啟動凍結並成為死結；例如，在 `domain.xml` 中進行以下設定會導致該問題：

```
<jvm-options>-Djava.security.debug=access,failure</jvm-options>
```

解決方案

尚不存在。請避免設定此標幟。

Message Queue

本節描述已知的 Java Message Queue 問題以及相關的解決方案。

在某些與時間相關的情況下，JMS 重新連線不能成功完成。(ID 6173308、6189645、6208728、6198481、6199510、6199510)

在與時間相關的方案中重新連線失敗可能是由多個問題導致的。但是，您通常可以透過以下方式解決這些問題：

- 重新啟動涉及的代理程式
- 包含重新啟動 Application Server 實例

appclient 8.0 至 8.1 中變更了非同步訊息偵聽程式運作方式 (ID 6198465)

由於最新的變更，當非同步訊息偵聽程式為 app-client 容器中唯一的活性執行緒時，其餘的 appclient VM 以常駐程式存在。此運作方式對於 ACC 中執行非同步接收的過去的應用程式是一種回歸。該問題會影響設定 JMS 訊息偵聽程式並結束主執行緒的應用程式用戶端。

解決方案

請勿結束主執行緒。等待訊息偵聽程式告知主執行緒後，再終止主執行緒。

訊息代理程式記錄包含「無法傳送」訊息。(ID 6204180)

在叢集環境中執行時，如果代理程式的記憶體容量很小，則可能會在代理程式記錄中看到下列錯誤訊息：

```
Internal error, unable to deliver .....: java.lang.NullPointerException
```

對於持久性使用者來講，雖然使用者已確認此訊息，但是以後該訊息可能被重新傳送至使用者（含有重新傳送標幟設定）。

發生此錯誤是由於當系統記憶體受到限制時，系統仍需要的資料也受到限制。此錯誤僅發生於傳送至使用者的貼附於此代理程式而非傳輸至叢集中的其他代理程式的訊息。

解決方案

增加訊息代理程式程序 (-Xmx) 的最大 Java 堆疊大小，以避免系統執行時記憶體容量小。

發生故障重新啟動後，叢集中正在執行的代理程式記憶體不足。(ID 6205463)

作為叢集的一部份的訊息代理程式重新啟動後，該正在執行的訊息代理程式記憶體不足。有兩種不同的問題可能會導致此問題。若要確定其中的一種問題或兩種問題是否是相關的，請檢查叢集中的 Application Servers 和 Message Brokers 的記錄檔：

- 如果 Application Server 記錄檔具有使用下列格式的「警告」訊息：

```
MQRA:OMR:Endpoint Unavailable:Marking OMR as invalid-omrId=1 for:ActivationSpec
configuration ...
```

那麼如 " 在某些與時間相關的情況下，JMS 重新連線不能成功完成。(ID 6173308、6189645、6208728、6198481、6199510、6199510)" 中所述，觸發記憶體問題。

- 如果一個或多個代理程式記錄檔位於叢集中的其他代理程式中，並以下列格式顯示訊息：

```
ERROR Internal Error: received ack twice on ...
```

那麼如 " 發生故障後，重新啟動叢集時訊息代理程式記憶體伸展。(ID 6208621)" 中所述，觸發記憶體問題。

解決方案

請參閱 6208621 或 6208728 列示的解決方案 (依據導致此問題的問題)。

發生故障後，重新啟動叢集時訊息代理程式記憶體伸展。(ID 6208621)

當發生故障後重新啟動叢集中的訊息代理程式時，由於主題目標中非持久性 MDB 的狀態為錯誤傳輸，因此代理程式中的記憶體可能會伸展。

啟動作為作用中的叢集的一部份的訊息代理程式後，在代理程式和應用程式伺服器記錄中均可看到異常。如果逾時，則重新啟動的代理程式將執行容量小的記憶體。在這種情況下，重新啟動代理程式後，叢集中應用程式伺服器的一個或多個記錄將拋出一個類似於以下訊息的「警告」訊息：

```
[#|2004-12-03T17:45:12.821-0800|WARNING|sun-appserver-ee8.1|javax.enterprise.system.stream
.err|_ThreadID=12;|com.sun.messaging.jms.JMSEXception: [C4000]: Packet acknowledge failed.
user=admin, broker=<brokername>....
```

訊息代理程式記錄中還可以經常看到相應訊息，格式如下：

```
ERROR Internal Error: received ack twice on ...
```

問題是，當訊息代理程式貼附至某個作用中的叢集時，叢集中的其他代理程式會將有關所有作用中的使用者資訊轉寄至該訊息代理程式。如果 MDB 具有其中一個遠端代理程式的非長期用戶，當 MDB 轉寄使用者資訊時，可能會發送不正確的資訊。重新啟動的代理程式收到無效使用者資訊後，該代理程式會將訊息的附加副本錯誤地路由至其他代理程式。

發生這種情況後，遠端使用者會將一項「[C4000]: Packet acknowledge failed」訊息記錄於 ApplicationServer 記錄中，並且在遠端代理程式上將記錄一項「double ack」錯誤。提供訊息的代理程式將不正確確認每次在代理程式記錄中看到的「double ack」訊息。逾時將導致提供訊息的代理程式執行容量小的，或不足的記憶體。

解決方案

如果此問題正在發生，那麼系統將清除內部資訊取消部署並且重新部署 MDB。若要避免此問題影響一般作業期間代理程式的作業：

- 如果不存在效能問題，那麼變更 MDB 以使用長期（而不是非長期性）用戶。
- 如果存在效能問題，那麼在一段時間內未收到確認時，配置系統以便清除「遺漏的」訊息。可以使用下列任何一種完成此動作：
 - 設定 REMOVE_OLDEST 中的 limitBehavior，並且限制 MDB 使用的目標中的訊息：

```
imqcmd update dst -n <dest name> -t t -o limitBehavior=REMOVE_OLDEST -o  
maxNumMsgs=<some limit>
```
 - 變更傳輸至主題的訊息中的「使用期」設定。

監視

本節介紹已知的監視問題和相關的解決方案。

某些 HTTP 服務的監視統計資訊中不存在有用的資訊，應該將其忽略。 (ID 6174518)

當檢視 HTTP 服務中某些元素的監視統計資訊時，存在的某些值與目前值不對應，或始終為 0。具體來說，下列 HTTP 服務統計資訊不存在適用於 Application Server 8.1 的資訊，應該將其忽略：

- http-service
 - load1MinuteAverage
 - load5MinuteAverage
 - load15MinuteAverage
 - rateBytesTransmitted
 - rateBytesReceived
- pwc-thread-pool (元素)

解決方案

在以後的版本中，將移除這些監視，並由更適當的資訊取代。

未移除取消部署的 EJB 模組的監視 mbean (雖然移動了該監視名稱下的所有統計資訊)。(ID 6191092)

例如：

```
EJBModuleMonitorMap().size() = 1  eventhough ejb module is undeployed
EJBModuleMonitor().getName() = sqe_ejb_s1_01
```

EJB 模組與應用程式也一樣。都是程式化地 (透過 MBeanAPI) 並且透過 `asadmin list/get` 移動監視名稱下的所有統計資訊，但空監視 mbean 仍然存在。

診斷

`asadmin list -m "server.applications"` 顯示下列輸出：

```
server.applications.MEjbApp
server.applications.__ejb_container_timer_app
server.applications.adminapp
server.applications.admingui
server.applications.com_sun_web_ui
server.applications._export_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01
```

您可以查看下列統計資訊：

```
bin/asadmin list -m "server.applications._export_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01"
server.applications._export_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01.SQEMessage
server.applications._export_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01.TheGreeter
```

取消部署後：

```
_export_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01
```

如果執行某項 `list` 指令，仍可看到下列應用程式：

```
asadmin list -m "server.applications"
server.applications.MEjbApp
server.applications.__ejb_container_timer_app
server.applications._export_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01
server.applications.adminapp
server.applications.admingui
server.applications.com_sun_web_ui
```

但不包含任何監視統計資訊：

```
asadmin list -m "server.applications._expo  
rt_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01"  
Nothing to list at server.applications.-export-install-nov-11-domains-domain1-ap  
plications-j2ee-modules-sqe-ejb-s1-01.
```

若要取得以字串開頭的有效名稱，請使用萬用字元 ('*') 字元。例如，若要列示以 server 開始的所有監視實體的名稱，請使用 list "server.*"。

解決方案

此項是無害的。可以安全地重新部署模組，不會出現任何問題。雖然沒有移除根監視 Mbean，但它為空監視。

PointBase

本節介紹與 PointBase 相關的已知問題及其相應解決方案。

設定應用程式連線區中的隔絕層級，將導致 PointBase 異常。(ID 6184797)

對於指向 PointBase 資料庫安裝的 JDBC 連線區，設定 transaction-isolation-level 連線區屬性為除預設值 (Connection.TRANSACTION_READ_COMMITTED) 外的任何值，都將導致異常。但對於指向其他資料庫的連線區，將同一參數設定為非預設值卻不會拋出異常。

解決方案

對於指向 PointBase 資料庫安裝的 JDBC 連線區，請勿嘗試設定 transaction-isolation-level。

如果網路伺服器與內嵌式驅動程式配合使用，則 PointBase 拋出異常。(ID 6204925)

如果同時使用網路伺服器驅動程式與內嵌式驅動程式，那麼附帶的 PointBase 有時會拋出異常。

解決方案

要麼使用內嵌式驅動程式，要麼網路伺服器驅動程式，切勿同時使用。

範例

本節介紹與包含 Application Server 8.1 產品的範例碼相關的已知問題及其相應解決方案。

`setup-one-machine-cluster` 在 Windows 上出現懸置現象，但在 Solaris 中工作正常；需要按 **[Ctrl]+[C]** 來取消 `mqfailover`，然後必須重新執行。(ID 6195092)

例如，若要重現錯誤，請參閱

`install_dir\samples\ee-samples\failover\apps\mqfailover\docs\index.html`，然後執行下列指令：

- 主控台 1

```
cd install_dir\samples\ee-samples asant start-mq-master-broker1
```
- 主控台 2

```
cd install_dir\samples\ee-samples asant start-mq-cluster-broker1
```
- 主控台 3

```
cd install_dir\samples\ee-samples asant start-mq-cluster-broker2
```
- 主控台 4

```
cd install_dir\samples\ee-samples asadmin start-domain domain1
```

如果您已經執行任何其他 EE 範例的 `asant setup-one-machine-cluster-without-ha` 或者 `asant setup-one-machine-cluster-with-ha`，那麼請執行 `asant configure-mq`，否則執行 `asant setup-one-machine-cluster-and-configure-mq`。在這種情況下，指令顯示成功：

```
start_nodeagent: [echo] Start the node agent cluster1-nodeagent [exec] Command
start-node-agent executed successfully.
```

然後系統會無限期懸置。

解決方案

此時為 None。這種問題同樣影響在 Windows 中使用 `ant` 目標的所有 EE 範例。一種解決方法是，按 **[Ctrl]+[C]** 登出懸置程序，然後再重新執行。

有關您需要依照 `asadmin deploy` 說明建立 JMS 資源後，再執行 MQ 防故障備用範例應用程式的內容，說明文件中沒有明確地陳述。(ID 6198003)

拋出的錯誤如下：

```
/opt/SUNWappserver/domains/domain1/config/sun-acc.xml -name MQFailoverTestClient -textauth
-user j2ee -password j2ee
Nov 18, 2004 10:50:17 PM com.sun.enterprise.naming.NamingManagerImpl bindObjects
SEVERE: NAM0006: 找不到 JMS 目標物件: jms/durable/TopicA
Nov 18, 2004 10:50:18 PM com.sun.enterprise.naming.NamingManagerImpl bindObjects
SEVERE: javax.naming.NameNotFoundException
javax.naming.NameNotFoundException
```

對於如果使用 `asadmin deploy` 指令進行手動部署，那麼必須手動建立 JMS 資源，以及應該使用提供的 `ant` 目標來部署範例應用程式的相關內容，說明文件中沒有明確地陳述。

解決方案

對於 `build.xml` 程序檔 (該程序檔建立執行應用程式所要求的 JMS 資源)，請使用 `asant deploy` 目標。

Linux 上的 web 服務/安全性範例中的證書建立期間發生執行期間錯誤。(ID 6198239)

在 Linux 上部署 `install_dir/samples/webservices/security` 範例 (basicSSL) 時，證書未被建立，並且拋出類似於下列錯誤的錯誤：

```
generate_certs: [echo] ***Exporting certificate from NSS database [exec] Result: 1 [echo]
***Generating Java Keystore from generated certificate [exec] keytool error:
java.lang.Exception: Input not an X.509 certificate [exec] Result: 1 [echo] ***Generating
Java trust store from generated certificate [exec] keytool error: java.lang.Exception:
Input not an X.509 certificate [exec] Result: 1
.
.
generate_certs: [echo] ***Exporting server certificate from NSS database to a PKCS12
certificate file [exec] /opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so: version
`NSS_3.9' not found (required by /opt/sun/appserver/lib/pk12util) [exec]
/opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so: version `NSS_3.6' not found (required
by /opt/sun/appserver/lib/pk12util) [exec] /opt/sun/appserver/lib/pk12util:
/usr/lib/libnss3.so: version `NSS 3.7' not found (required by
/opt/sun/appserver/lib/pk12util) [exec] Result: 1
```

問題是 NSS 程式庫在 Linux 安裝中的位置與在 Solaris 安裝中的位置不同。當在 Linux 中部署時，您需要確保 `LD_LIBRARY_PATH` 指向正確 NSS 程式庫。在您所處的環境中，或在 `install_dir/bin/asant shell` 包裝程序檔中設定 `LD_LIBRARY_PATH`。

解決方案

執行下列動作之一：

- 設定 `LD_LIBRARY_PATH=/opt/sun/private/lib`。
- 加入至 `install_dir/bin/asant` 程序檔中的以下行中：
`LD_LIBRARY_PATH=$SAS_NSS:$LD_LIBRARY_PATH;export LD_LIBRARY_PATH`

有關部署應用程式的 `- ee samples asadmin deploy` 指令省略 `availabilityenabled=true` 選項的說明文件，確保範例應用程式為依預設啟用的 HADB。(ID 6198796)

解決方案

使用 `asadmin` 指令：

1. 移至範例目錄的根目錄；例如：

```
cd install_dir/samples/ee-samples/failover/apps/dukesbookstore
```

2. 執行 `asadmin deploy`，將應用程式部署至本機 Application Server 實例；例如：

```
install_dir/bin/asadmin deploy -user adminuser -password adminpassword --host DASHost --port DAsport --availabilityenabled=true --target cluster-name dukesbookstore.ear
```

以同樣的方式執行其他所有 EE 範例 (除 `install_dir/samples/ee-samples/failover/apps/mqfailover` 外) 中的 `asadmin deploy` 指令。請注意，MQ 不使用 HADB。

無法使用 `asant` 程序檔，對 `dukesbookstore` EE 範例執行 `failover` 測試。(ID 6199076)

設定雙機叢集後，`dukesbookstore failover` 範例將會產生錯誤。除非每個叢集中均具有一個資料庫。現在，當您在具有同時在兩個單獨的機器中執行的實例的叢集中從 DAS 部署範例時，程序檔將使用 `PointBase` 主機作為 `localhost`。當部署一個 EE 範例後，在兩個實例中 `PointBase` 主機均作為 `localhost`，從而部署 JDBC 資源。因此，兩個實例中的 `localhost:9092` 均希望在兩部機器上同時執行 `PointBase`。

問題是屬於同一叢集的兩個實例，無法使用不同的資料庫。若要解決此問題，如果以主機名稱取代 `database.properties` 中的 `localhost` 後，叢集中的兩個實例將能夠存取資料庫：其中一個透過 `localhost`，而另一個則透過指定的主機名稱。

解決方案

編輯 `PointBase` 伺服器正在其上執行的主機上的 `install_dir/samples/database.properties` 檔案，同時將 `pointbase.server` 值設定為 `PointBase` 正在其上執行的主機 (而非 `localhost`) 名稱。

目前的設定即為，`pointbase.server=localhost` 適用於單機叢集，而不適用於雙機叢集（第二個實例的 `localhost` 上可能未執行 PointBase）。

MQ-failover 範例應用程式具有硬編碼至 Ant 設定目標中一種中的「cluster1」的叢集名稱。(ID 6202363)

在一個 Ant 設定目標中，MQ-failover 範例應用程式具有 `cluster1` 的硬編碼。因此，如果您要修改 `ee-samples` 中的 `cluster.properties` 中的 `cluster-name` 從 `cluster1` 至某個不同的名稱，嘗試在 `cluster1` 中設定預設主機時，該範例失敗：

```
set-default-jms-host-to-broker1:  
[echo] Setting default JMS host to samplesbroker1  
[exec] No object matches the specified name "cluster1-config.jms-service.default-jms-host"  
[exec] CLI137 Command set failed.
```

當使用 `${cluster.name}` 替換 `install_dir/samples/ee-samples/cluster.properties` 時，`install_dir/samples/ee-samples/build.xml` 檔案的硬編碼將參考 `cluster1`。

解決方案

硬編碼的 `cluster1` 字串必須使用替換的 `${cluster.name}`。手動修改 `install_dir/samples/ee-samples/build.xml` 以變更 `set-default-jms-host-to-broker1`，從 `cluster1` 至 `${cluster.name}` 或者 `cluster.properties` 中指定的 `customer` 叢集名稱。

安全性

本節介紹已知的問題、與 Application Server 相關的解決方案、web 應用程式安全性以及證書。

透過 client-side 指定目標訊息對 client-side message-security-binding 元素無效。(ID 6155080)

例如，`client-side message-security-binding` 元素中的目標訊息透過 `java-method` 在 `service-ref` 元素的 `port-info` 元素中指定時，會產生此問題。

```
<!ELEMENT service-ref ( service-ref-name, port-info*, call-property*, wsdl-override?,  
service-impl-class?, service-qname? )>  
<!ELEMENT port-info ( service-endpoint-interface?, wsdl-port?, stub-property*,  
call-property*, message-security-binding? )>  
<!ELEMENT message-security-binding ( message-security* )>  
<!ELEMENT message-security ( message+, request-protection?, response-protection? )>  
<!ELEMENT message ( java-method? | operation-name? )>
```

在此使用 `message-security-binding` 元素，以便定義 web 服務端點的特定方法的訊息保護策略。

解決方案

使用 `message` 元素內的 `operation-name` 元素透過 WSDL 作業名稱識別在包含 `message-security` 元素中定義的訊息保護策略是否適用。

「CertificateNotYetValidException」如果 DAS 與遠端節點代理程式機器時鐘無法同步化。(ID 6181989)

如果 DAS 伺服器與節點代理程式安裝在不同的機器上，並且這些機器上的時鐘無法同步化，嘗試執行 `asadmin --start remote-node-agent` 指令失敗，並且產生 `CertificateNotYetValidException` 錯誤。

解決方案

將 DAS 伺服器與所有遠端節點代理程式機器上的時鐘同步化。

使用 J2SE 5.0 版本，無法在 EE 上執行 `WebServiceSecurity` 應用程式。(ID 6183318)

由於以下原因，使用 J2SE 5.0 版本無法執行 `WebServiceSecurity` 應用程式：

- J2SE 5.0 PKCS11 不支援 UNWRAP 模式
- J2SE 5.0 PKCS11 不支援有 PKCS11 的 RSA/ECB/OAEPWithSHA1AndMGF1Padding。

J2SE 工作小組已將此錯誤歸檔為「CR 6190389：為 RSA-PKCS1 與 RSA-OAEP wrap/unwrap 機制新增支援」。

解決方案

配合使用 J2SE 1.4.2 與任何其他 JCE 提供者（非依預設所包含的）。請注意，該配置中不存在硬體加速器支援。

如果在 `jms-serviceSSL` 中設定 `mq-scheme` 和 `mq-service`，與 MQ 的 SSL 通訊將不工作；如果在 `jms-service` 中設定 `mq-scheme` 和 `mq-service`，與 MQ 的 SSL 通訊將不工作。(ID 6202606)

從資源配接卡建立的連線工廠，使用資源配接卡中未設定的資訊。因此，從資源配接卡建立的連線工廠將不具有 SSL 資訊。

解決方案

如果需要在 Application Server 與 MQ 之間使用 SSL 通訊，請建立使用 SSL 語法明確指定的 `addresslist` 的連線工廠。例如，連線工廠中的 `addresslist` 可以為：

```
mq://mqserver-1:7676/ssljms,mq://mqserver-2:7676/ssljms
```

當 MQ 與 Application Server 位於不同的位置，並且它們之間的網路連接性可被闖入者利用時，可能會需要 Application Server 與 MQ 之間的 SSL 通訊。

位於連接埠 80 的 https 偵聽程式的 URL 沒有指定重新導向至 http 偵聽程式的連接埠號。(ID 6207297)

在預設連接埠 (443) 上啓用 SSL 偵聽程式時，將瀏覽器中的 URL 指定給安全連接埠而沒有指定連接埠號，將導致瀏覽器重新指向位於非安全性 (http) 偵聽程式的連接埠 80。

例如：

1. 在連接埠 443 上建立 SSL 偵聽程式，然後重新啓動 Application Server。
2. 將瀏覽器指向 <https://servername:443>。
該頁面正確載入
3. 將瀏覽器指向 <https://servername> (無連接埠號)。
瀏覽器載入 <http://servername:80>，而非載入 <https://servername>。

當 SSL 偵聽程式位於非預設連接埠 (443) 時，不會產生此問題。

解決方案

選擇兩種解決方案之一：

- 建議存取 https 連接埠 443 上的 SSL 偵聽程式時，所有連結的網站訪問者及程式碼總是指定連接埠號。
- 請勿將連接埠 443 用於任何 SSL 偵聽程式。

升級公用程式

本節介紹已知的升級公用程式問題和相關的解決方案。

從 Application Server Enterprise Edition 8 升級至 Application Server Enterprise Edition 8.1 時，在自訂路徑而非 *install_dir*/domains 目錄中建立的網域無法直接升級。(ID 6165528)

當執行升級公用程式，並識別 *install_dir* 為來源安裝目錄時，升級程序僅升級在 *install_dir*/domains 目錄下建立的網域。在其他位置建立的網域不能進行升級。

解決方案

在啓動升級程序之前，將不同位置的所有網域目錄複製到 *install_dir*/domains 目錄中。

從 J2EE 1.4 SDK 升級至 Application Server EE 8.1 期間，隨附的 J2SE 安裝無法正確升級。此問題僅影響獨立版本版本的產品。(ID 6196741)

從 J2EE 1.4 SDK 升級至 Application Server 8.1 獨立安裝期間，會產生此問題。此升級中，隨附的 J2SE 1.4.2 應該升級至 J2SE 5.0。但是，已有的 J2SE 5.0 安裝中的一些 JAR 檔案無法正確升級，從而產生損壞的 J2SE 5.0 安裝。

安裝程式將報告成功的升級，在升級程序期間執行升級工具時，將不會遇到任何問題。但是，以後嘗試啟動升級的 Application Server 會失敗，並帶有以下異常：

```
Exception in thread "main"
[#|2004-11-17T18:12:24.033-0800|WARNING|sun-appserver-ee8.1|javax.enterprise.system.stream
.err|_ThreadID=10;|java.lang.NoClassDefFoundError: javax/net/ssl/TrustManager at
com.sun.enterprise.security.SecurityLifecycle.onInitialization(SecurityLifecycle.java:59)
at
com.sun.enterprise.server.ApplicationServer.onInitialization(ApplicationServer.java:215)
at com.sun.enterprise.server.PEMain.run(PEMain.java:277) at
com.sun.enterprise.server.PEMain.main(PEMain.java:219)
```

解決方案

針對此問題，有多個解決方法：

- 安裝適當支援版本的獨立 J2SE。在 Application Server 升級期間，請在 Application Server 安裝程式的「Java 配置」螢幕中選擇選擇重新使用現有 Java 2 SDK 的選項，而非安裝 Java 2 SDK 5.0 的預設選項。提供獨立 J2SE 安裝的路徑。
- 執行升級前，請移除或重命名現有的 J2EE 1.4 SDK *install_dir*/jdk 子目錄。則在 Application Server 8.1 升級程序期間，將正確安裝 J2SE 5.0。請注意，僅在透過安裝程式的目錄選取螢幕導覽並且被提示允許現有安裝的「適當的升級」時，才可移除或重新命名此目錄。
- 如果已經執行了升級，並且遇到一些伺服器啟動問題，那麼請安裝適當支援版本的獨立 J2SE，然後修改 *install_dir*/config/asenv.conf 檔案 (Linux 與 Solaris)，或者 *install_dir*\config\asenv.bat 檔案 (Windows) 中的 AS_JAVA 變數。此變數的值應指向新的獨立 J2SE 安裝的位置。

在按下「啟動升級精靈」按鈕後，執行「適當的升級」的安裝程式，在某些 Linux 系統上無法啟動升級工具。(6207337)

此問題已在多個 Linux 系統中發現，在 Java Desktop System 2 中最常見，但也見於 RedHat 發行軟體中。

在最終的安裝程式螢幕上按一下 [啟動升級工具] 按鈕之後，安裝程式無法啟動升級工具以完成升級程序，且無限期當機，並不返回指令提示。

解決方案

如果使用指令行安裝模式現地執行升級，則不會遇到此問題。

1. 如果在 GUI 模式中現地執行升級並遇到此問題，請在啓動安裝程式的終端機視窗中按下 **Ctrl+C**，以退出安裝程式。
2. 使用以下指令從終端機視窗中啓動升級工具：

```
install_dir/bin/asupgrade --source install_dir/domains --target install_dir --adminuser adminuser  
--adminpassword adminpassword --masterpassword changeit
```

adminuser 與 *adminpassword* 應與您正在升級的安裝所使用的值相匹配。

3. 當升級工具完成升級程序後，您還可以啓動瀏覽器並輸入以下 URL 以檢視 [關於] 頁面：
file://install_dir/docs-ee/about.html

如果您還選取安裝選項以註冊產品，請使用產品 [關於] 頁面上的連結進入註冊頁面。

Web 容器

本節描述已知的 Web 容器問題以及相關的解決方案。

使用 `--precompilejsp=true` 部署應用程式可能會鎖定應用程式中的 JAR 檔案，進而導致以後的取消部署或重新部署失敗。(僅 Windows) (ID 5004315)

如果您在 Windows 上部署應用程式時請求 JSP 的預先編譯，則以後無法按預期嘗試取消部署或重新部署該應用程式 (或任何具有相同模組 ID 的應用程式)。問題在於 JSP 預先編譯會開啓應用程式中的 JAR 檔案，但不會關閉它們，同時 Windows 會防止取消部署刪除這些檔案或防止重新部署覆寫它們。

請注意，取消部署會前進至某個點，在此處，應用程式會在邏輯上從 Application Server 中移除。還請注意，`asadmin` 公用程式不會傳回任何錯誤訊息，但公用程式的目錄和鎖定的 jar 檔案會保留在伺服器上。伺服器的記錄檔將包含描述無法刪除檔案和應用程式目錄的訊息。

取消部署失敗後會嘗試重新部署應用程式，因為伺服器會嘗試移除現有檔案與目錄，此嘗試仍失敗。如果您嘗試部署使用與原來部署的應用程式具有相同模組 ID 的任何應用程式，便會出現這種情況，因為伺服器使用該模組 ID 選擇目錄名稱以存放應用程式檔案。

基於同樣原因，不先取消部署即嘗試重新部署應用程式將會失敗。

診斷

如果您嘗試重新部署應用程式或在取消部署後再部署該應用程式，`asadmin` 公用程式會傳回一個類似如下的錯誤。

```
執行指令時出現異常。異常訊息爲：CLI1171 指令部署失敗：在網域中部署應用程式失敗；無法部署。模組目錄被鎖定且無法刪除
```

解決方案

如果您在部署應用程式時指定 `--precompilejsp=false` (預設設定)，則不會出現此問題。請注意，第一次使用應用程式將觸發 JSP 編譯，因此第一次請求的回應時間會比以後的請求的回應時間長。

還請注意，如果進行預編譯，應先停止並重新啟動伺服器，然後再取消部署或重新部署應用程式。關機會釋放鎖定的 JAR 檔案，因此重新啟動後才能成功取消部署或重新部署。

使用基於 Servlet 2.4 的包含空的 `<load-on-startup>` 元素的 `web.xml` 無法部署 WAR。(ID 6172006)

`web.xml` 中可選的 `load-on-startup servlet` 元素表示要載入相關的 `servlet` 並將其初始化為宣告 `servlet` 的 Web 應用程式啟動的一部分。

該元素的可選內容是一個整數，表示要載入並初始化與 Web 應用程式之其他 `servlet` 相關的 `servlet` 的順序。只要在啟動其包含的 Web 應用程式過程中載入並初始化 `servlet`，空的 `<load-on-startup>` 即表示順序錯誤。

`web.xml` 的 Servlet 2.4 模式不再支援空的 `<load-on-startup>`，即在使用基於 Servlet 2.4 的 `web.xml` 時必須指定一個整數。如果指定與 `<load-on-startup/>` 中的一樣的空 `<load-on-startup>`，`web.xml` 將無法針對 `web.xml` 之 Servlet 2.4 模式進行驗證，進而導致無法部署 Web 應用程式。

返回至相容性問題。指定空的 `<load-on-startup>` 仍可使用基於 `web.xml` 的 Servlet 2.3。

解決方案

使用基於 `web.xml` 的 Servlet 2.4 時指定 `<load-on-startup>0</load-on-startup>`，以表示 `servlet` 載入順序並不重要。

使用 AMX API 從伺服器中移除 J2EE 應用程式參照將移除該應用程式，但該應用程式仍可存取。(ID 6173248)

使用 AMX API 時，不先明確地停止應用程式即移除應用程式參照，該應用程式仍可存取。此運作方式是預先設計的，屬於文件的疏忽。

解決方案

若要移除應用程式以使其不可存取，請：

1. 停止應用程式
2. 移除應用程式的參照。
3. 取消佈署應用程式。

無法編譯資源受限伺服器上的 JSP 頁面。(ID 6184122)

已存取 JSP 頁面但無法編譯，且伺服器記錄包含錯誤訊息「Unable to execute command」，以及以下堆疊追蹤：

```
at org.apache.tools.ant.taskdefs.Execute$Javal3CommandLauncher.exec(Execute.java:655) at
org.apache.tools.ant.taskdefs.Execute.launch(Execute.java:416) at
org.apache.tools.ant.taskdefs.Execute.execute(Execute.java:427) at
org.apache.tools.ant.taskdefs.compilers.DefaultCompilerAdapter.executeExternalCompile(DefaultCompilerAdapter.java:448) at
org.apache.tools.ant.taskdefs.compilers.JavacExternal.execute(JavacExternal.java:81) at
org.apache.tools.ant.taskdefs.Javac.compile(Javac.java:842) at
org.apache.tools.ant.taskdefs.Javac.execute(Javac.java:682) at
org.apache.jasper.compiler.Compiler.generateClass(Compiler.java:396)
```

解決方案

設定 JSP 編譯切換「fork」為「false」。

有兩種方法可以執行此操作：

- 通常，將 `${S1AS_HOME}/domains/domain1/config/default-web.xml` 中 `JspServlet` 的 `fork init` 參數設定為 `false`：

```
<servlet> <servlet-name>jsp</servlet-name>
<servlet-class>org.apache.jasper.servlet.JspServlet</servlet-class> ... <init-param>
<param-name>fork</param-name> <param-value>>false</param-value> </init-param>
...</servlet>
```

- 在每個 Web 應用程式基礎上，將 `sun-web.xml` 中的 `fork JSP` 配置特性設定為 `false`：

```
<sun-web-app> <jsp-config> <property name="fork" value="false" /> </jsp-config>
</sun-web-app>
```

每一種設定均將防止 ant 產生新的 javac 編譯程序。

如何報告問題與提供回饋

如果您的 Sun Java System Application Server 有問題，請使用下列途徑之一與 Sun 客戶支援聯絡：

- **Feedback Submittal form** — 是用來提交關於 Application Server 產品回饋的表單，位於：
<http://java.sun.com/docs/forms/J2EE14SubmittalForm.html>
- **J2EE-INTEREST list** — 是 J2EE 問題的郵件清單，位於：
<http://archives.java.sun.com/archives/j2ee-interest.html>
- **有關 Java 開發人員連線的錯誤資料庫** — 若要檢視錯誤或提交錯誤，請使用 Java 開發人員連線 Bug Parade，位於：
<http://developer.java.sun.com/servlet/SessionServlet?url=/developer/bugParade/index.jshtml>
- **Java Technology Forums** — 是互動式的討論板，可讓開發人員共有有關 Java 技術和程式設計技術方面的知識並討論此類問題。使用此處的 J2EE SDK 論壇來討論與 Sun Java System Application Server Enterprise Edition 8.1 產品相關的問題：
<http://forum.java.sun.com/>
- **線上 Sun 軟體支援服務** — 位於：
<http://www.sun.com/service/sunone/software>

該網站可連結至知識庫、線上支援中心和 Product Tracker，以及維護程式和支援聯絡電話號碼。

- 與維護合約相關的電話派送號碼

如此我們將可為您提供最完善的解決問題方法，請在與支援聯絡時提供以下資訊：

- 對問題進行描述，包括問題發生的情形以及對作業的影響
- 機器類型、作業系統版本以及產品版本，包括任何可能對問題造成影響的修補程式及其他軟體
- 您的詳細操作步驟以重現問題
- 所有的錯誤記錄或核心傾印

Sun 歡迎您提出意見和建議

Sun 非常重視改善其文件品質，歡迎您提出意見和建議。

若要共用您的意見和建議，請移至 <http://docs.sun.com> 並按一下 [Send Comments]。在線上表單中，提供了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頂部找到。例如，本書的標題是「*Sun Java System Application Server Enterprise Edition 8.1 2005Q1 版本說明*」，文件號碼是 819-2029。在您提出意見時，可能需要在表單中輸入英文版書名和文件號碼，本書的英文版文件號碼和書名為：819-0214 和「*Sun Java System Application Server Enterprise Edition Release Notes*」。

其他 Sun 資源

可在下列位置找到有用的資訊：

- Application Server 產品資訊：
http://www.sun.com/software/products/appsrvr/home_appsrvr.html
- Application Server 產品說明文件：
<http://docs.sun.com/db/prod/slappsrv#hic/> 和
http://docs.sun.com/db/prod/slappsrv#hic/?l=zh_TW
- Sun Java System 文件
<http://docs.sun.com/prod/java.sys> 和
http://docs.sun.com/db/prod/java.sys?l=zh_TW
- Sun Java System 專業服務
<http://www.sun.com/service/sunps/sunone>
- Sun Java System 軟體產品與服務
<http://www.sun.com/software>
- Sun Java System 軟體支援服務
<http://www.sun.com/service/sunone/software>
- Sun Java System 支援和知識庫
<http://www.sun.com/service/support/software>

- Sun 支援和培訓服務
<http://training.sun.com>
- Sun Java System 諮詢和專業服務
<http://www.sun.com/service/sunps/sunone>
- Sun Java System 開發人員資訊
<http://developers.sun.com>
- Sun 開發人員支援服務
<http://www.sun.com/developers/support>
- Sun Java System 軟體培訓
<http://www.sun.com/software/training>
- Sun 軟體資料表
<http://www.sun.com/software>
- Sun Microsystems 產品說明文件：
<http://docs.sun.com/>

Copyright © 2004-2005 Sun Microsystems, Inc.。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產權。特別是 (但不僅限於)，這些智慧產權可能包括一項或多項在 <http://www.sun.com/patents> 上列出的美國專利，以及一項或多項美國和其他國家 / 地區的其他專利或待批專利。

SUN 專有權/機密。

美國政府權利 — 商業軟體。政府使用者均應遵守 Sun Microsystems, Inc. 標準軟體許可授權合約和 FAR 及其增補文件中的適當規定。

使用應遵守授權合約的條款。

本發行物可能包括由協力廠商開發的材料。

有些部分可能源自 Berkeley BSD 系統，並經加州大學授權。

Sun、Sun Microsystems、Sun 標誌、Java 和 Solaris 是 Sun Microsystems, Inc. 在美國和其他國家 / 地區的商標或註冊商標。所有 SPARC 商標均經過授權使用，它們是 SPARC International, Inc. 在美國和其他國家 / 地區的商標或註冊商標。