

Sun Java™ System Application Server Enterprise Edition 发行说明

版本： 8.1 2005Q1

文件号码： 819-2028

Sun Java™ System Application Server Enterprise Edition 8.1 2005Q1 产品极大地简化了创建和管理 Web 服务应用程序的任务。它为可伸缩服务提供了较高的性能、群集和高可用性功能，使其在软件和硬件出现故障时可以继续运行。Application Server 为 Web 服务提供了开发路径，从而在提供独特而灵活的发展机会的同时简化了开发过程。

本发行说明包含 Sun Java System Application Server 8.1 2005Q1 发行时可用的重要信息。介绍了新增功能和增强功能、已知问题和限制以及其他信息。开始使用 Application Server 8.1 之前，请先阅读本文档。

可以在以下 Sun Java System 文档 Web 站点找到本发行说明的最新版本：

<http://docs.sun.com/db/prod/slappsrv#hic/> 和 <http://docs.sun.com/db/prod/slappsrv#hic/?l=zh>。

在安装和设置软件之前，请先查看此 Web 站点上的相关信息，并在以后定期查看最新的发行说明和产品文档。

本文档包括以下内容：

- [关于 Application Server Enterprise Edition 8.1 2005Q1](#)
- [已知问题和限制](#)
- [如何报告问题和提供反馈](#)
- [Sun 欢迎您提出意见](#)
- [其他 Sun 资源](#)

本文中引用了第三方 URL，这些 URL 提供了其他相关信息。

注 Sun 对本文档中提到的第三方 Web 站点的可用性不承担任何责任。对于此类站点或资源中的（或通过它们获得的）任何内容、广告、产品或其他材料，Sun 并不表示认可，也不承担任何责任。对于因使用或依靠此类站点或资源中的（或通过它们获得的）任何内容、产品或服务而造成的或连带产生的实际或名义损坏或损失，Sun 概不负责，也不承担任何责任。

关于 Application Server Enterprise Edition 8.1 2005Q1

Sun Java System Application Server Enterprise Edition 8.1 是与 J2EE 1.4 平台兼容的服务器，它用于在大规模生产环境中开发和部署 J2EE 应用程序和基于 Java 技术的 Web 服务。

本节包括：

- [8.1 版中的新增功能](#)
- [硬件和软件要求](#)
- [独立版本](#)
- [相关文档](#)

8.1 版中的新增功能

Sun Java System Application Server Enterprise Edition 8.1 2005Q1 实现了很多新增功能和增强功能，以下各节中介绍了：

- [本版本的增强功能](#)
- [J2EE 支持](#)
- [高性能](#)
- [可扩展性](#)
- [高可用性](#)
- [JavaServer Faces 1.1 支持](#)

本版本的增强功能

Application Server Enterprise Edition 8.1 包括以下增强功能：

- **改进的管理**——Application Server 通过使用基于浏览器的控制台或者可执行脚本的命令行界面，来支持复杂多机企业部署的远程安全管理。它还提供了基于 JMX 的丰富 API，从而允许对管理和监视功能进行远程、安全和可编程化的访问。
- **消息代理**——Application Server 捆绑有集成的企业级消息代理，后者可以提供具有高可用性、可靠性、高性能和可伸缩性的消息传送。

- **扩展平台支持**——支持其他操作系统、数据库、语言环境和硬件。
- **Sun Java Enterprise System**——作为 Sun Java Enterprise System 的主要组件，Application Server 与 portal 和网络身份服务紧密集成。
- **移植和升级工具**——这些工具使您可以检验 J2EE 应用程序是否符合标准以及是否具有可移植性，有助于从其他 J2EE Application Server（JBoss、WebLogic 和 WebSphere）进行移植，并有助于从 Sun ONE Application Server/iPlanet Application Server 的早期版本进行升级。
- **Java 2 Standard Edition 5.0 支持**——Application Server 支持 Java 2 Standard Edition 5.0，此版本包括增强的管理和监视功能，并改进了许多性能和可伸缩性。
- **JDBC 驱动程序**——Application Server 捆绑有 Sun JDBC 驱动程序。
- **Web 服务安全性**——这些容器消息安全机制通过使用 OASIS WS-Security 标准的 X509 和用户名 / 密码配置文件，实现 SOAP Web 服务调用的消息级别验证（例如 XML 数字签名和加密）。
- **WS-I 基本配置文件 1.1**——此版本符合 J2EE 1.4 规范，它实现了 Web 服务互操作性 (WS-I) 基本配置文件 1.1 而使 Web 服务应用程序可具有互操作性。
- **与 iWay 适配器的后端连接**——Sun Microsystem 现在转售和支持针对主要后端系统（SAP、Siebel、Oracle、CICS 和 IBM MQ Series）的二十二种 iWay 适配器，有助于您在 Application Server 环境中充分利用现有 IT 应用程序。这些适配器支持 J2EE 连接器体系结构 1.5 规范和 Web 服务 (SOAP) 标准，并且包括开发者工具以缩短连接至后端应用程序的时间。
- **最新的 HADB 管理系统**——UNIX® 平台包含新的高可用性数据库 (HADB) 管理系统（HADB 4.4 版）。这将消除对 SSH/RSH 的依赖性，但要求将网络配置为可以进行 UDP 多址广播。有关 HADB 要求和限制的详细信息，请参见《Sun Java System Application Server Enterprise Edition 8.1 Installation Guide》。

J2EE 支持

Sun Java System Application Server 8.1 2005Q1 支持 J2EE 1.4 平台。下表介绍了 J2EE 1.4 平台上可用的增强 API。

表 1 J2EE 1.4 平台上的主要 API 更改

API	说明
组件	
应用程序和应用程序客户机	通过 XML 模式实现标准部署描述符
Enterprise JavaBeans (EJB) 2.1	计时器服务和 EJB Web 服务端点
Java Servlet 2.4	Web 服务端点过滤器

表 1 J2EE 1.4 平台上的主要 API 更改 (续)

API	说明
JavaServer Pages (JSP) 2.0 体系结构	表达式语言和标记库
J2EE Connector Architecture 1.5	内嵌资源适配器和 Java 消息服务 (JMS) 插件化
Web 服务	
Java Web Services Developer Pack 1.5	用于构建、测试和部署 XML 应用程序、Web 服务和 Web 应用程序的集成工具包
用于基于 XML 的远程过程调用的 Java API (JAX-RPC) 1.1	用于 WSDL 和 Java 技术的映射以及对开发 Web 服务客户机和端点的支持
WS-I 基本配置文件 1.0	使用 WSDL 和 SOAP 启用互操作性的元素
Java 的带附件的 SOAP API (SAAJ) 1.2	用于基于 SOAP 的消息传送的 API；帮助创建带有附件的 SOAP 消息
用于 XML 注册表的 Java API (JAXR) 1.0	统一且标准的 API，用于访问 XML 注册表，例如用于通用说明、发现和集成 (UDDI 和 ebXML) 的 XML 注册表
其他	
J2EE Deployment 1.1	启用部署 J2EE 组件和应用程序的标准 API
J2EE Management 1.0	管理 J2EE 平台的信息模型的定义
Java Management Extensions (JMX) 1.2	标准管理 API
Java 容器授权合同 (JACC) 1.0	J2EE Application Server 和授权策略提供者之间的安全性合同的定义
用于 XML 处理的 Java API (JAXP) 1.2	应用程序可以用来解析和变换 XML 文档的 API；还添加了对处理 XML 模式的支持
JMS 1.1	一种消息传送标准，使 J2EE 应用程序组件可以创建、发送、接收和读取消息；还添加了对用于队列和主题的统一 API 的支持
JavaMail 1.3	一组建立邮件系统模型的抽象类；还包括对 API 的细微更新

高性能

Application Server 包括一个高性能的 EJB 容器、Web 容器和服务，并使用 Sun Java System Message Queue 软件支持并行消息传送。

可扩展性

Application Server 可以通过服务器实例的群集和请求负载平衡来支持水平可伸缩性。它也可以实现业内领先的支持大型多处理器计算机的垂直可伸缩性。可以对集成消息代理进行群集以获得更佳的可伸缩性和可用性。从 HTTP 客户机访问的客户机、基于 RMI/IIOP 的胖客户机应用程序、Web 服务客户机和 JRM 客户机可被负载平衡到 Application Server 群集。

高可用性

Application Server 包括 HTTP、IIOP 和 JMS 客户机的负载平衡；HTTP 会话故障转移支持；EJB 群集和故障转移支持；具有高可用性的 EJB 计时器；分布式事务恢复；对持续式应用程序升级的支持；以及一个用于存储 J2EE 应用程序瞬间状态的高可用性数据库。

可用性允许对群集中的 Application Server 实例进行故障转移保护。一个 Application Server 实例出现故障时，其他 Application Server 实例将接管指定给该故障服务器的会话。会话信息存储在 HADB 中。HADB 支持 HTTP 会话的持久性、有状态会话 Bean 和单点登录证书。

JavaServer Faces 1.1 支持

Sun Java System Application Server Enterprise Edition 8.1 支持 JavaServer Faces 1.1 技术。JavaServer Faces 技术由一组服务器端 API 组成，这些 API 表示用于管理其状态、事件、处理和输入验证的用户界面组件。这些 API 还定义了页面导航并支持国际化和辅助功能。您可以使用 JSP 自定义标记库添加自定义 UI 组件。

使用 JavaServer Faces 技术进行开发时，开发团队的每个成员可以集中开发整个进程的单个部分，然后由一个简单的编程模型将这些部分连接起来，从而使开发周期更为简单高效。

硬件和软件要求

本节列出了安装 Sun Java System Application Server Enterprise Edition 8.1 产品之前必须满足的要求。

- [平台要求](#)
- [重要修补程序信息](#)
- [JDBC 驱动程序和数据库](#)
- [配置 Oracle](#)
- [配置 PointBase](#)
- [Web 服务器](#)
- [浏览器](#)
- [高可用性要求和限制](#)
- [HADB 文件系统支持](#)
- [升级 Sun Java System Application Server](#)
- [切换到 J2SE 5.0](#)
- [其他要求](#)

平台要求

下表列出了 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 产品所支持的操作系统。此外，还标识了安装和运行 Application Server 所需的最小内存和推荐的内存。

表 2 Sun Java System Application Server 8.1 2005Q1 平台要求

操作系统	最小内存	建议的内存	最小磁盘空间	建议的磁盘空间	JVM
Sun Solaris 9, 10 (SPARC) Solaris 9, 10 (x86)	512 MB	1 GB	250 MB 可用空间	500 MB 可用空间	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 可用空间	300 MB 可用空间	J2SE 1.4.2_06 J2SE 5.0
Microsoft Windows 2000® Advanced Server Service Pack 4+	512 MB	1 GB	250 MB 可用空间	500 MB 可用空间	J2SE 1.4.2_06 J2SE 5.0
Microsoft Windows Server® 2003	512 MB	1 GB	250 MB 可用空间	500 MB 可用空间	J2SE 1.4.2_06 J2SE 5.0

在 UNIX 上，可以使用 `uname` 命令查看操作系统版本。可以使用 `df` 命令查看磁盘空间。

Solaris 修补程序要求

建议 Solaris 9、10（x86、SPARC）用户安装“Sun 推荐的修补程序群集”。此修补程序群集可以在以下位置的“Recommended and Security Patches”下找到：

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

RedHat Enterprise Linux 3.0 的附加软件包要求

要运行此产品的本机组件（包括安装程序），应安装以下软件包，该软件包不是标准 RedHat Enterprise Linux 3.0 分发的一部分：`compat-libstdc++-7.3-2.96.118.i386.rpm`

可以从以下位置下载该软件包：

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

重要修补程序信息

有关 Sun Java System Application Server Enterprise Edition 8.1 所需修补程序的当前列表，请访问 <http://sunsolve.sun.com>，然后选择“Patches”或“Patch Portal”。点击 Sun Java System Application Server Enterprise Edition 8.1 链接。操作系统修补程序的要求更改并且 Java Enterprise System 组件的修补程序变得可用后，这些更新可在 SunSolve 上获得，最初的形式为建议的修补程序群集。

JDBC 驱动程序和数据库

Sun Java System Application Server 用于支持使用相应的 JDBC 驱动程序连接任何 DBMS。有关经 Sun 测试发现适合构建符合 J2EE 规范的数据库配置的组件列表，请参阅下表：

表 3 J2EE 兼容的 JDBC 驱动程序

JDBC 供应商	JDBC 驱动程序类型	支持的数据库服务器
i-net Software	类型 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	类型 2	IBM DB2 8.1 Service Pack 3+
PointBase	类型 4	PointBase Network Server 4.8
DataDirect	类型 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+
用于 Oracle 的 Sun Java System JDBC 驱动程序	类型 4	Oracle (R) 9.2.0.3, 10G
用于 DB2 的 Sun Java System JDBC 驱动程序	类型 4	IBM DB2 8.1 Service Pack 3+
用于 Sybase 的 Sun Java System JDBC 驱动程序	类型 4	Sybase ASE 12.5.2
用于 Microsoft SQL Server 的 Sun Java System JDBC 驱动程序	类型 4	Microsoft SQL Server 2000 4.0 Service Pack 1
Oracle	类型 4、类型 2	Oracle (R) 9.2.0.3, 10G

有关 i-net Software 的更多信息，请参见：

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

有关 DataDirect Technologies 的更多信息，请参见：

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

配置 Oracle

必须正确配置 Oracle JDBC 驱动程序才能与 J2EE 1.4 兼容。对类型 2 和类型 4 驱动程序应用以下配置：

1. 使用 9.2.0.3 或更高版本的 JDBC 驱动程序。
2. Oracle 数据库的参数 (init.ora) 文件中应具有 compatible=9.0.0.0.0 或更高版本。
3. 使用 ojdbc14.jar 文件。
4. 配置 Application Server 以定义以下 JVM 属性：

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

此外，对于类型 2 驱动程序，必须在已启动 Application Server 的环境中定义 ORACLE_HOME 和 LD_LIBRARY_PATH 变量（必须包含 \$ORACLE_HOME/lib）。例如，将它们添加到 asenv.conf 文件中并确保将其导出。

配置 PointBase

许多样例应用程序使用 Application Server 附带的 PointBase 数据库服务器。使用 Application Server Enterprise Edition 时，您必须先配置 PointBase 数据库服务器，然后才能使用该服务器。但是，在将 PointBase 与 Application Server 结合使用之前，请记住支持的配置组合。

表 4 支持的 J2SE/PointBase 组合

Application Server	PointBase
受支持	
J2SE 1.4	J2SE 1.4
J2SE 5.0	J2SE 1.4
不受支持	
J2SE 5.0	J2SE 5.0

配置 PointBase 的方法有两种：

- 将 JAVA_HOME 环境变量设置为 J2SE 的位置。仅 J2SE 1.4.2 支持与 Application Server 8.1 捆绑在一起的 PointBase 实现。
- 编辑 Application Server 的 PointBase 配置文件。

若使用第一种方法，请执行以下步骤：

1. 确保您已安装了要使用的 J2SE。

如果尚未安装，请下载 J2SE 1.4.2。

2. 使用适于您的操作系统和 shell 的命令，将 JAVA_HOME 环境变量设置为 J2SE 的安装目录；例如：

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

若使用第二种方法，执行的过程将取决于操作系统。

Solaris 和 Linux

编辑 `install_dir/pointbase/tools/serveroption/pbenv.conf` 配置文件，将行：

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

更改为

```
PB_JAVA=J2SE_location
```

其中 `J2SE_location` 为 J2SE 的安装目录。如果已随 Application Server 一起安装了 J2SE，则 J2SE 默认安装在 `install_dir/jdk`。进行此更改后，您可以使用 `startserver` 脚本启动 PointBase。

Windows

编辑 `install_dir\pointbase\tools\serveroption\pbenv.bat` 配置文件，将行：

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

更改为

```
set PB_JAVA=J2SE_location
```

其中 `J2SE_location` 为 J2SE 的安装目录。如果已随 Application Server 一起安装了 J2SE，则 J2SE 默认安装在 `install_dir\j2se1.4`。进行此更改后，您可以通过运行 `startserver.bat` 来启动 PointBase。

Web 服务器

本节列出了 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 所支持的 Web 服务器。

表 5 支持的 Web 服务器

Web 服务器	版本	操作系统
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
Microsoft IIS®	5.0+	Windows Server 2003 Windows 2000 Advanced Server Service Pack 4+

以后将可以支持其他平台，包括 Windows 和 HP-UX。

浏览器

本节列出了 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 所支持的浏览器。

表 6 支持的 Web 浏览器

浏览器	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

高可用性要求和限制

在配置 Sun Java System Application Server 高可用性组件前，必须满足以下高可用性要求：

- 要与 Application Server 配合正常工作，HADB 需要的最小内存为 512 MB，建议的内存为 1 GB。
- HADB 仅支持 IPv4。
- 必须将网络配置为可以进行 UDP 多址广播。
- 请勿将动态 IP 地址 (DHCP) 用于 create domain、extend domain、hadbm create 或 hadbm addnodes 命令中所使用的主机。
- 如果在 Red Hat Linux 3.0 上运行 HADB，则必须安装 Update 4 以避免操作系统出现过度交换 (swapping) 问题。请注意，尚未使用 Red Hat Update 4 对 AS 进行测试。

HADB 文件系统支持

如果您要将 HADB 配置为使用以下文件系统之一，则需考虑几个重要事项：

- **ext2 和 ext3**——对于 Red Hat Application Server 3.0，HADB 支持 ext2 和 ext3 文件系统。对于 Red Hat Application Server 2.1，HADB 仅支持 ext2 文件系统。
- **Veritas**——在 Solaris 平台上使用 Veritas 文件系统时，将向历史记录文件写入消息“警告：直接磁盘 I/O 映射失败”。此消息指出：HADB 无法打开用于数据和日志设备的直接 I/O。直接 I/O 可以减少写入磁盘页面的 CPU 耗时，从而提高性能。它还可以降低管理操作系统中的脏数据页面的系统开销。

要结合使用直接 I/O 和 Veritas 文件系统，请使用以下方法之一：

- 在以选项 mincache=direct 进行安装的文件系统上创建数据和日志设备。此选项将应用于该文件系统上创建的所有文件。有关详细信息，请参见命令 mount_vxfs(1M)。
- 使用 Veritas Quick I/O 工具对文件的文件执行原始 I/O。有关详细信息，请参见《VERITAS File System 4.0 Administrator's Guide for Solaris》。

请注意，尚未使用 Application Server 8.1 对这些配置进行测试。

有关使用 Application Server 8.1 软件安装和配置 HADB 的详细信息，请参阅《Sun Java System Application Server Enterprise Edition 8.1 Installation Guide》。

升级 Sun Java System Application Server

有关从先前版本的 Application Server 升级到 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 的完整说明，请参阅 Installation Guide。

切换到 J2SE 5.0

Sun Java System Application Server 8.1 2005Q1 支持 J2SE 5.0 作为基础 JVM，但是捆绑的 PointBase 数据库却不支持。如果您要使用 J2SE 5.0 而非捆绑的 J2SE 1.4.2 来运行 Application Server，请执行以下步骤（对于 Windows 和 Unix）：

1. 下载 J2SE 5.0 SDK（而非 JRE）并将其安装在您的系统上（如果尚未这样做）。
可以从 <http://java.sun.com/j2se/1.5.0/> 下载 J2SE 5.0 SDK。

2. 完全停止 Application Server。

您可以使用以下命令行：

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

或管理控制台 GUI：

a. 单击 "Application Server" 节点。

b. 单击 “停止实例”。

3. 编辑 `install_dir/config/asenv.conf` 文件（在 Windows 上为 `asenv.bat`），将 `AS_JAVA` 的值更改为指向 J2SE 5.0 主目录：

4. 编辑 `as-install/samples/common.properties` 文件，将以 "com.sun.aas.javaRoot..." 开头的行更改为引用 J2SE 5.0 主目录。

5. 重新启动 Application Server。

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

注 如果是从捆绑版本 (JDK 1.4.2_06) 以前的 JDK 版本进行升级，则仅使用上述步骤无法升级到 J2SE 5.0。具体而言，除了上述步骤，您还必须删除所有现有的域，然后重新创建它们。

其他要求

安装 Sun Java System Application Server 软件之前，必须满足以下附加要求。

- **可用空间：**您的临时目录必须具有至少 35 MB 的可用空间以便安装 Sun Java System Application Server，以及 250 MB 的可用空间以便安装 SDK。
- **使用卸载程序：**如果要从系统中删除 Application Server，必须使用此软件中包含的卸载程序。如果尝试使用其他方法，则在试图重新安装同一版本或安装新版本时将出现问题。
- **可用端口：**您必须具有七个未使用的可用端口。
 - 安装程序自动检测正在使用的端口，并建议将当前未使用的端口用作默认设置。默认情况下，用于 HTTP 的初始默认端口号为 8080，用于 HTTPS 的初始默认端口号为 8181，用于管理服务器的初始默认端口号为 4849。
 - 安装程序将检测已用端口并为您分配其他两个端口：[Sun Java™ System Message Queue](#)（默认情况下，端口号为 7676）和 IIOP（默认情况下，用于 IIOP 的端口号为 3700 而用于 IIOP/SSL 的端口号为 1060 和 1061）。如果这些默认端口号正在使用，安装程序将从动态端口范围内指定一个随机端口号（请注意，它不一定是下一个可用的端口号）。

- **启动以前安装的服务器 (UNIX)**——除非您要更换以前安装的服务器，否则，应在开始 Sun Java System Application Server 8.1 安装进程之前先将其启动。这样，安装程序才能检测到已使用的端口，从而避免再分配它们用于其他用途。
- **更换以前安装的服务器 (UNIX)**——如果您已安装较旧版本的 Sun Java System Application Server 并且要用当前版本的 Application Server 替换它，则在安装新服务器之前应先将它停止。使用安装程序升级向导来升级服务器。
- **关闭防火墙 (Microsoft Windows)**——您必须在安装 Sun Java System Application Server 软件前停止所有防火墙软件，这是因为某些防火墙软件默认情况下会禁用所有端口。安装程序必须能够准确确定哪些端口可用。

有关兼容性的详细信息，请参见位于以下站点的 Upgrade and Migration Guide:

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

独立版本

Sun Java System Application Server Enterprise Edition 8.1 的独立版本与 Java ES Enterprise Edition 版本存在差异，具体表现在以下几个方面：

- 8.1 独立产品可以由任何用户来安装，而 Java ES 只能以超级用户身份安装。
- 在独立版本中，HADB 组件是作为子组件的可见组件，而在 Java ES 安装中它是共享组件。
- 在独立版本中，Application Server 所需的所有共享组件均安装在同一安装目录下，而在 JES 中，这些组件安装在不同目录下。
- 使用独立安装程序时，Application Server 的产品文件、域和配置数据默认情况下存储在一个目录下，而使用 Java ES 时，它们存储在多个目录下。
- 独立版本允许在已安装有相同或不同版本 Application Server 的系统上进行安装，并且无需卸载现有安装。这是通过保持不同版本之间或同一版本的不同实例之间唯一的安装目录来实现的。
- 独立版本支持将现有 Sun Java System Application Server Platform Edition 8.0 安装或 Sun Java System Application Server Platform Edition 8.1 安装“就地升级”到 Sun Java System Application Server Enterprise Edition 8.1。

相关文档

除了本发行说明以外，Application Server 产品还包含一套完整的文档，可以在以下位置找到：

http://docs.sun.com/app/docs/coll/ApplicationServer8_pe_04q4 和
http://docs.sun.com/app/docs/coll/ApplicationServer8_pe_04q4?l=zh

下表总结了包含在 Application Server 核心应用程序文档集中的文档。

表 7 此文档集中包含的文档

书名	说明
快速入门指南	如何开始使用 Sun Java System Application Server 产品。
<i>Installation Guide</i>	安装 Sun Java System Application Server 软件及其组件。
<i>Deployment Planning Guide</i>	评估系统需求和企业状况，确保以最适合您的站点的方式部署 Sun Java System Application Server。此外还介绍了部署应用服务器时应该注意的常见问题。
<i>Developer's Guide</i>	创建和实现要在 Sun Java System Application Server 上运行的 Java™ 2 Platform, Enterprise Edition (J2EE™ 平台) 应用程序，这些应用程序遵循针对 J2EE 组件和 API 的开放式 Java 标准模型。其中包括有关开发者工具、安全性、汇编、部署、调试和创建生命周期模块的一般信息。
<i>J2EE 1.4 Tutorial</i>	使用 J2EE 1.4 平台技术和 API 开发 J2EE 应用程序，并将这些应用程序部署到 Sun Java System Application Server。
管理指南	从管理控制台配置、管理和部署 Sun Java System Application Server 子系统和组件。
<i>High Availability Administration Guide</i>	有关高可用性数据库的安装后配置和管理说明。
<i>Administration Reference</i>	编辑 Sun Java System Application Server 的配置文件 domain.xml。
<i>Upgrade and Migration Guide</i>	将应用程序移植到新的 Sun Java System Application Server 编程模型，特别是从 Application Server 6.x 和 7 进行移植。该指南还介绍了可导致与产品规范不兼容的相邻产品版本和配置选项之间的差异。
<i>Performance Tuning Guide</i>	优化 Sun Java System Application Server 以提高性能。
<i>Troubleshooting Guide</i>	解决 Sun Java System Application Server 问题。
<i>Error Message Reference</i>	解决 Sun Java System Application Server 错误消息。
<i>Reference Manual</i>	可用于 Sun Java System Application Server 的实用程序命令，以手册页样式编写。其中包括 asadmin 命令行界面。

已知问题和限制

本节介绍了有关 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 软件的已知问题和相应解决方法。如果汇总说明未指明特定平台，则所有平台都可能出现此问题。本部分信息按以下内容进行组织：

- [管理](#)
- [应用程序客户机](#)
- [捆绑的 Sun JDBC 驱动程序](#)
- [连接器](#)
- [容器管理的持久性](#)
- [Deploytool](#)
- [文档](#)
- [高可用性](#)
- [安装](#)
- [J2EE Tutorial](#)
- [生命周期管理](#)
- [日志记录](#)
- [Message Queue](#)
- [监视](#)
- [PointBase](#)
- [样例](#)
- [安全性](#)
- [升级实用程序](#)
- [Web 容器](#)

管理

本节介绍已知的管理问题和相应的解决方法。

如果不存在 domain1， package-appclient 脚本将不起作用。(ID 6171458)

默认情况下，在 \$INSTALL/lib/package-appclient.xml 中有一个用于 domain1（由 asenv.conf 来指向）的 AS_ACC_CONFIG 变量的硬编码值。如果删除 domain1 并创建新域，将不会用新域名更新 AS_ACC_CONFIG 变量，从而导致 package-appclient 脚本失败。

解决方法

请执行以下操作之一：

- 保持 domain1 不变，围绕它创建其他域。
- 删除 domain1 并用新域名替换 \$INSTALL/lib/package-appclient.xml 中的用于 domain1 的硬编码值。如果 domain1 不存在，则每次创建新域时，都必须执行此操作。

无法恢复具有其他名称的备份域。(ID 6196993)

无法使用 backup-domain 和 restore-domain 命令镜像同一 Application Server 安装上的域，这是由于使用不同于原始名称的其他名称不能恢复域，即使 asadmin restore-domain 命令提供了重命名域的选项。重命名备份域似乎已成功，但尝试启动重命名的域却会失败，因为没有更改域配置中的条目，并且 startserv 和 stopserv 使用的是原始域名来设置路径。

解决方法

用于 restore-domain 的域名必须与用于原始 backup-domain 命令的域名相同。Application Server 8.1 中的 backup-domain 和 restore-domain 命令仅用于在同一台计算机上备份和恢复同一个域。

不支持启动带有附加 JMX 代理的 Application Server。(ID 6200011)

J2SE 1.4.x, 5.0 或更高版本可以在 Application Server 上进行配置。J2SE 5.0 平台的完整功能是可以启动 JMX 代理。在服务器启动时，如果您明确设置了系统属性，此功能将被激活。

示例值包括：

```
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"
```


在配置了 JMX 属性并启动服务器之后，将在 Application Server VM 中启动新的 jmx-connector 服务器。此过程的副作用是会对管理功能造成不利影响，并且 Application Server 管理 GUI 和 CLI 可能会产生异常结果。出现此问题的原因在于内置 jmx-connector 服务器与新的 jmx-connector 服务器之间存在一些冲突。

解决方法

如果使用 jconsole（或任何其他 JMX 兼容客户机），请考虑重新使用标准的 JMX Connector Server，它在 Application Server 启动时启动。

当服务器启动时，server.log 中将显示类似于以下所示的内容。您可以连接到其中指定的 JMXServiceURL 中，并在成功提供证书后执行相同的管理 / 配置操作，例如：

```
[#|2004-11-24T17:49:08.203-0800|INFO|sun-appserver-ee8.1|javax.enterprise.system.tools.admin|_ThreadID=10;|ADM1501:这是 JMXConnectorServer 的
JMXServiceURL:[service:jmx:rmi:///jndi/rmi://hostname:8686/management/rmi-jmx-connector].这是远程管理客户机使用 JSR 160 JMX 连接器进行连接的位置。|#]
```

有关更多信息，请参阅《Sun Java System Application Server 8.1 管理指南》。

过度限制对 Application Server start 和 stop 脚本的执行权限。（仅适用于 UNIX）(ID 6206176)

如果您以用户 "A" 身份登录后运行 asadmin restore-domain 命令，这些脚本将以权限 744 (rwxr--r--) 结束。如果随后您尝试以用户 "B" 身份（即使 "B" 为超级用户）启动或停止域，则会由于这些脚本的可执行性仅适用于 "A" 而失败。

解决方法

更改脚本的权限：

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

应用程序客户机

本节介绍已知的应用程序客户机问题和相应的解决方法。

封装在应用程序客户机归档文件中的库 JAR 覆盖 MANIFEST 文件。(ID 6193556)

如果在您的客户机 JAR 中具有顶层 JAR 文件（在此情况下，为 reporter.jar），则当您部署客户机 JAR 时，该 JAR 的 MANIFEST 文件将覆盖客户机 JAR 的 MANIFEST 文件。

解决方法

此时尚未提供。

捆绑的 Sun JDBC 驱动程序

本节介绍已知的捆绑的 Sun JDBC 驱动程序问题和相应的解决方法。

如果两个并行事务正在运行并且其中一个已回滚，则当使用准备的语句进行更新时，结合使用 `TRANSACTION_SERIALIZABLE` 隔离级别和 Microsoft SQL Server 的捆绑 Sun 驱动程序的应用程序可能会挂起。(ID 6165970)

要为连接设置所需的隔离级别，必须以同一隔离级别创建相应的连接池。有关配置连接池的详细信息，请参见 *Application Server 8.1 2005Q1 管理指南*。

解决方法

无。

PreparedStatement 错误 (ID 6170432)

描述 1

如果应用程序在一个事务中生成超过 3000 个 PreparedStatement 对象，则 DB2 中可能会出现以下错误：

```
[sunm] [DB2 JDBC 驱动程序] 无更多可用语句。请重新创建具有较大 dynamicSections 值的软件包。
```

解决方法 1

将以下属性添加到连接池定义中，以使驱动程序可以重新绑定具有较大动态段值的 DB2 软件包：

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

有关配置连接池的详细信息，请参见 *Application Server 8.1 2005Q1 管理指南*。

描述 2

可能抛出的与上述 PreparedStatement 错误有关的其他错误消息如下所示：

```
[sunm] [DB2 JDBC 驱动程序] [DB2] 虚拟存储或数据库资源不可用
```

解决方法 2

增大 DB2 服务器的配置参数 `APPLHEAPSZ`。最佳值为 4096。

描述 3

隔离级别 TRANSACTION_SERIALIZABLE。

如果您的应用程序使用隔离级别 TRANSACTION_SERIALIZABLE 并使用上述建议的参数之一，它可能会在获得连接时挂起。

解决方法 3

要为连接设置所需的隔离级别，必须以同一隔离级别创建相应的连接池。有关说明，请参见 *Application Server 8.1 2005Q1 管理指南*。

对于 Application Server Enterprise Edition 8.1，捆绑的 DB2 Sun JDBC 驱动程序在默认配置中不起作用。这是由于 DB2 JDBC 驱动程序类需要显式 charsetProvider RuntimePermission。(ID 6183492)

解决方法

修改 server.policy 文件以便为使用 JDBC 驱动程序的已部署应用程序提供以下权限：

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

请注意只有捆绑的 DB2 Sun JDBC 驱动程序需要此操作。

使用捆绑的 Sun 驱动程序为 Sybase Adaptive Server 设置隔离级别时出现问题。(ID 6189199)

- 如果两个并行事务正在运行并且其中一个已回滚，则当使用准备的语句进行更新时，结合使用 TRANSACTION_SERIALIZABLE 隔离级别和 Sybase Adaptive Server 的捆绑 Sun 驱动程序的应用程序可能会挂起。连接回滚失败，系统显示以下消息，并且已回滚的连接不能再使用：

```
java.sql.SQLException: [sunm] [Sybase JDBC 驱动程序] 由于线的争用而无法提交请求
```

- Sybase Adaptive Server 不支持 TRANSACTION_REPEATABLE_READ 隔离级别。但是，查询 DatabaseMetaData 时，捆绑的 Sun 驱动程序会返回数据库支持此隔离级别的内容。使用此隔离级别的应用程序将失败。
- 使用捆绑的 Sun 驱动程序的应用程序无法设置 TRANSACTION_READ_UNCOMMITTED 隔离级别。在首次访问 DataBaseMetaData 时，应用程序会抛出以下异常：

```
java.sql.SQLException: [sunm] [Sybase JDBC 驱动程序] [Sybase] 优化程序无法找到唯一的索引，它可以使用该索引对表 "sybssystemprocs.dbo.spt_server_info" 执行隔离级别 0 扫描。
```

解决方法

无。

连接器

本节介绍已知的 J2EE 连接器体系结构问题和相应的解决方法。

`connection-validation` 在 `jdbc-connection-pools` 中不具有动态可重新配置性。
(ID 4930792)

JDBC 连接池在创建之后，其 `is-connection-validation-required` 属性不具有动态可重新配置性。这意味着对于已创建的池连接，不能随意地将验证切换为 `on`（或 `off`）。池的 `validation-method` 属性也同样存在这种情况。

解决方法

为减少侵入正在运行的应用程序的次序，有三种可能的解决方法：

1. 创建 `jdbc-connection-pools`，将 `validation` 切换为 `on`。
2. 删除 `jdbc-connection-pool`，然后重新创建它，将 `validation` 切换为 `on`。
这将仅影响极少数依赖于此特定池的已部署应用程序。
3. 更改 `validation` 属性并重新启动 `Application Server`。
这将影响所有已部署的应用程序（因为执行了重新启动）。

在重新启动 DAS 实例后，如果将级联设置为 `false`，取消部署连接器模块的操作将失败。(ID 6188343)

此方案中，已在 DAS 和连接器连接池中部署了独立或嵌入式连接器模块，并且已为该部署的模块创建了资源。在重新启动 DAS 实例后，如果将级联设置为 `false`，取消部署连接器模块的操作将失败并且会出现以下异常：

```
[#|2004-10-31T19:52:23.049-0800|INFO|sun-appserver-ee8.1|javax.enterprise.system  
.core|_ThreadID=14;|CORE5023: 卸载应用程序时出错 [foo]|#]
```

解决方法

在重新启动 DAS 实例后，使用级联的取消部署（将 `cascade` 选项设置为 `true`）来取消部署独立和嵌入式连接器。

容器管理的持久性

本节介绍已知的容器管理的持久性问题和相应的解决方法。

如果 where 子句包含 OR 运算符和单值型 cmr 导航，则 EJBQL 查询可能不包含所有匹配的结果。(ID 6184864)

如果 EJBQL 查询中的 where 子句包含 OR 运算符和单值型 cmr 导航，则查询结果将不会包含导航路径为 null 的所在行的结果，即使该导航路径位于不同的 OR 子句中。

例如，对于一个由 Employee、Department 和 Insurance 组成的模式。Employee 与 Department 具有一对多关系，而与 Insurance 具有一对一关系：

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

以上查询将不会返回其名为 John 并且不属于任何部门的雇员。

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

以上查询将不会返回其保险名称为 xyz 并且不属于任何部门的任何雇员。也不会返回其部门名称为 Engineering 并且没有任何保险的任何雇员。

解决方法

分别对每个 OR 条件执行查询并合并结果。

Deploytool

本节介绍已知的 Deploytool 问题和相应的解决方法。

Deploytool 通常不会在以下 Sun 部署描述符中创建 message-destination 元素：(ID 6197393)

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

被指定为“消息目标”选项卡中的“JNDI 名称”的 JMS 目标资源可能不会保存到 Sun 描述符中。在指定了“目标名称”（例如，PhysicalQueue——用 create-jmsdest 创建的物理目标）并按 Enter 键后，“目标名称”将显示在“显示名称”下，并且客户机或 Bean 名称将显示在“生产者”列表中。在特定于 Sun 的“JNDI 名称”文本字段中键入“jms/Queue”并按 Enter 键后，在标题栏中应用程序将不会显示为“（已更改）”，并且系统将向 ~/.deploytool/logfile 中写入错误。在保存了应用程序并返回到该选项卡后，“JNDI 名称”字段再次为空。在使用“工具”>“描述符查看器”>“Application Server 描述符”查看 Sun 描述符时，<jndi-name> 元素中的 <message-destination> 元素尚未创建。

此问题的原因在于：在 deploytool 会话期间，第一次为“消息目标 JNDI 名称”输入值后，该值在 Sun 描述符中显示正确，但是，org.netbeans.modules.schema2beans.BeanProp.setElement() 却抛出 IllegalArgumentException。随后对同一应用程序或其他应用程序中的“消息目标 JNDI 名称”的更改或添加将不会保存到 Sun 描述符中。

解决方法

要编辑消息目标的现有 JNDI 名称，请执行以下步骤：

1. 通过将“JNDI 名称”文本字段留空白并按 Enter 键来删除现有 JNDI 名称。
2. 键入新的 JNDI 名称并按 Enter 键。
3. 通过单击“工具”>“描述符查看器”>“Application Server 描述符”来查看 Sun 描述符。
4. 通过单击“文件”>“保存”来保存应用程序。

如果 JNDI 名称没有保存到 Sun 描述符中，请执行以下步骤：

1. 重新启动 deploytool。
2. 在“消息目标”选项卡上选择消息目标或添加新消息目标。
3. 在特定于 Sun 的“JNDI 名称”文本字段中输入消息目标的 JNDI 名称，然后按 Enter 键。
4. 通过单击“工具”>“描述符查看器”>“Application Server 描述符”来查看 Sun 描述符。
5. 通过单击“文件”>“保存”来保存应用程序。

除非是在 deploytool 会话期间第一次在“JNDI 名称”文本字段中输入值，否则每次需要在“消息目标”选项卡上的特定于 Sun 的“JNDI 名称”文本字段中输入值时均应重复以上步骤。

“新建 Web 服务向导”中中断的面板 (ID 6198981)

此问题表现为两种情况：

- 在“WSDL 文件”面板上单击“下一步”时并不转至下一个面板，而是抛出异常：

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

- 在“创建封装的端点”面板上，选择“创建 Web 服务端点和软件包”。在“后续步骤”面板上单击“完成”时并不在输出目录中创建端点模块，而是抛出异常：

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

问题在于 CLASSPATH 中缺少包含 XPathAPI.class 的 xalan.jar。请注意，使用 JDK 5.0 时不在此问题。

解决方法

将 xalan.jar 添加到 *install_dir/lib/processLauncher.xml* 的 *slas-deploytool* 进程的 CLASSPATH includes 中；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"
```

在简体中文版 Deploytool 中，“Home”被错译为“安装目录”。(ID 6203658)

在 deploytool 中创建企业 Bean 后，浏览至此 Bean 节点的“事务”或“安全性”选项卡，可以看到“Local Home”和“Remote Home”标签分别被错译为“本地安装目录”和“远程安装目录”。

文档

本节介绍已知的文档问题和相应的解决方法。

index.html 和 QuickStart.html 文档中的错误 (ID 6193749)

在 Application Server 8.1 文档集的 index.html 文件和 docs-ee/QuickStart.html 文件中存在两组错误。

1. 复制到每个域的 docroot 目录的默认 index.html 页面显示的路径不正确；该路径应为：

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

而非

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

2. 快速入门指南中有关负载均衡器和 Web 服务器的安装说明不正确。有关更新的说明，请参见联机快速入门指南：

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

请注意，此问题只影响 Application Server 8.1 的 Java ES Enterprise Edition 版本，而不影响独立版本。

- asadmin create-domain --help 命令返回不正确的用法，并且记录了无效的选项 (--admin.jmxport)。(ID 6207862)

asadmin create-domain 的 help 命令返回了 --admin.jmxport，而 --admin.jmxport 不是该命令的有效选项。

解决方法

不能将 --admin.jmxport 与 asadmin create-domain 命令结合使用。

Javadoc 不一致（不同的 ID）

缺少多个 AMX 接口和方法的 Javadoc 或该 Javadoc 不正确：

- ConnectorConnectionPoolStats 和 AltJDBCConnectionPoolStats 中缺少用于获得 NumConnAcquired 和 NumConnReleased 统计信息的 Getter 方法。在以后的版本中，将以 getNumConnAcquired() 和 getNumConnReleased() 的形式添加这些 getter 方法。
- 在 EJBCacheStats 中调用以下方法时抛出异常：getPassivationSuccesses()、getExpiredSessionsRemoved()、getPassivationErrors() 和 getPassivations()。在以后的版本中将修复此问题。
- 服务器启动后，可能需要几秒钟才能注册和使用所有的 AMX MBean。在以后的版本中，将可以确定完全装入 AMX MBean 的时间。
- 常数 XTypes.CONNNECTOR_CONNECTION_POOL_MONITOR 拼写错误 ("NNN")。在以后的版本中将纠正此问题。

高可用性

本节介绍已知的高可用性数据库 (HADB) 问题和相应的解决办法。

管理指南的附录 “编译和配置 Apache Web Server” 中有关结合使用 Sun Java System Application Server 和 Apache 的要求已过时。

下面列出了有关结合使用 HADB 和 Apache Web 服务器软件的软件要求。

- 适用于 Apache 1.3 的要求
 - openssl-0.9.7e (源代码)
 - mod_ssl-2.8.16-1.3.33 (源代码)
 - apache_1.3.33 (源代码)
 - gcc-3.3-sol9-sparc-local 软件包 (用于 Solaris 9 SPARC/x86)
 - gcc-3.3-sol9-intel-local 软件包 (用于 Solaris 9 x86)
 - flex-2.5.4a-sol9-sparc-local 软件包 (用于 Solaris 9 SPARC)
 - flex-2.5.4a-sol9-intel-local 软件包 (用于 Solaris 9 x86)

- 适用于 Apache 2.0 的要求
 - openssl-0.9.7e (源代码)
 - httpd-2.0.49 (源代码)
 - gcc-3.3-sol9-sparc-local 软件包 (用于 Solaris 9 SPARC)。
 - gcc-3.3-sol9-intel-local 软件包 (用于 Solaris 9 x86)
 - flex-2.5.4a-sol9-sparc-local 软件包 (用于 Solaris 9 SPARC)
 - flex-2.5.4a-sol9-intel-local 软件包 (用于 Solaris 9 x86)

编译前还有一个附加步骤。在 Solaris 10 平台上，在为 OpenSSL 运行 make 前，请运行以下命令：

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

最后，管理指南中详细说明了安装完成后必须对 Apache Web 服务器进行的修改的“配置负载平衡和故障转移”中的信息不完整。

在所有平台上

1. 在 `apache_install_dir` 下创建一个名为 `sec_db_files` 的目录。
2. 将 `domain/config/*.db` 复制到 `apache_install_dir/sec_db_files`。

在 Solaris 平台上:

- 将路径 `/usr/lib/mps/secv1` 添加到 `apache_install_dir/bin/apachectl` 脚本中的 `LD_LIBRARY_PATH` 中。必须在 `/usr/lib/mps` 之前添加此路径。

在 Linux 平台上

- 将路径 `/opt/sun/private/lib` 添加到 `apache_install_dir/bin/apachectl` 脚本中的 `LD_LIBRARY_PATH` 中。必须在 `/usr/lib` 之前添加此路径。

使用双网络时的 HADB 配置（无 ID）

使用两个子网上的双网络进行配置的 HADB 可以在 Solaris SPARC 上正常工作。但是，由于某些硬件平台上操作系统或网络驱动程序的问题，已发现在 Solaris x86 和 Linux 平台上并不总是能够正确处理双网络。这就导致 HADB 出现以下问题：

- 在 Linux 上，发送消息时某些 HADB 进程会被阻塞。这将导致 HADB 节点重新启动以及进行网络分区操作。
- 在 Solaris x86 上，网络故障后会出现一些问题，导致无法切换到其他网络接口。但并不总是会发这种情况，因此最好还是使用两个网络。这些问题在 Solaris 10 上已部分解决。
- 不支持链路聚合。
- 在 Windows 2003 上，HADB 不支持双网络 (ID 5103186)。

在添加新节点之后创建的新表不在添加的节点上分段。(ID 5042351)

如果创建了一个数据库实例并向其中添加了节点，则以后创建的任何新表都不会在创建数据库之后添加的节点上分段。只有在 `addnodes` 命令之前创建的表才能够在 `hadbm addnodes` 对其进行重新分段时使用添加的节点。这是因为 `create table` 使用执行 `hadbm create` 时创建的 `sysnode` 节点组。

解决方法

在添加任何新表之后都要运行 `hadbm refragment`，或在节点组 `all_nodes` 上创建新表。

不支持 packagepath 的异构路径。(ID 5091349)

不能在不同主机上的不同位置使用相同名称注册同一个软件包；例如：

```
hadbm registerpackage test --packagepath=/var/install11 --hosts europa11
软件包注册成功。
hadbm registerpackage test --packagepath=/var/install12 --hosts europa12
hadbm:Error 22171: 已注册使用软件包名称 test 的软件包。
```

解决方法

HADB 不支持数据库群集中节点之间的异构路径。确保 HADB 服务器的安装目录 (--packagepath) 在所有参与的主机之间均相同。

hadbm set 不检查资源可用性（磁盘和内存空间）。(ID 5091280)

使用 hadbm set 增加设备或缓冲区大小时，管理系统会在创建数据库或添加节点时检查资源可用性，但在更改设备或主内存缓冲区大小时则不会检查是否有足够的可用资源。

解决方法

在增大 devicesize 或 buffersize 配置属性之前，检验所有主机上是否有足够的可用磁盘 / 内存空间。

HADB 与 RedHat AS 3.0 位于同一台计算机上并在负荷下运行时的问题 (ID 6158393)

HADB 在与 AS 位于同一台计算机上的 RedHat 3.0 上运行。事务可能会中止，并影响性能。这是由于操作系统执行的过度交换而引起的。

解决方法

此问题已在 Red Hat EL 3.0 Update 4 中得到修复。已在 RedHat 3.0 update 4 中测试 HADB，并且证明操作系统的过度交换问题已消除。请注意，尚未在 update 4 中测试 Application Server 8.1。

configure-ha-cluster 命令可能会挂起。(ID 6159633)

使用 asadmin configure-ha-cluster 命令在多个主机上创建或配置高可用性群集时，该命令有时会挂起。HADB 管理代理或 Application Server 不抛出异常。

解决方法

HADB 不支持数据库群集中节点之间的异构路径。确保 HADB 服务器安装目录和配置目录在所有参与的主机之间均相同。确保在再次运行此命令之前清除系统信息库目录。

使用 HADB 时的 Application Server 性能 (ID 6172589)

在所有平台上，配置为使用 HADB 的 Application Server 实例的性能比以前版本要差，这是由于对 HADB 使用的 JDBC 驱动程序的更改而造成的。

解决方法

请立即与 Sun 服务部门联系，以获取解决方案。

重新分段期间第二个 addnodes 失败。(ID 6175436)

重新分段期间第二个（和后续）addnodes 命令可能会失败，并显示以下错误消息：

```
hadbm:Error 22042: 无法对数据库进行重新分段。请重试，使用 hadbm refragment 命令对数据库进行重新分段。  
原因：HADB-E-11747: 节点组 all_nodes 已存在
```

解决方法

使用 hadbm refragment 手动对表进行重新分段。

在 Windows 上无法创建大于 2 GB 的数据设备。(ID 6181845)

如果结合使用 hadbm create 或 hadbm set 与 --NumberOfDataDevices=1（默认值）和值大于 2 GB 的 --devicesize，则会出现以下错误：

```
DEVINIT-ERROR: 空间不足，写入 -2147479552 B, 共 -2036330496 B  
尝试在文件开始之前移动文件指针。
```

解决方法

在 Windows 中，如果要创建大于 2 GB 的数据设备，请用 devicesize 除以 2 GB，从而得出需要创建的设备数目。然后，根据计算结果使用 --NumberOfDataDevices 选项创建相应数目的数据设备。例如，如果要创建一个 5 GB 的数据设备：

$$5 \div 2 = 2.5$$

您应当进行向上舍入，并设置 --NumberOfDataDevices=3。

hadbm 帮助中的信息已过时 (ID 6190702)

hadbm 帮助系统中的某些信息已过时。

解决方法

有关最新信息，请参见 *Application Server 8.1 管理员指南* 中的 "HADB" 一章。

addnodes 命令失败，并显示未找到表错误 (ID 6214601)

在这种情况下，`hadbm refragment` 命令失败，并显示以下错误：

```
hadbm:Error 22042: 无法对数据库进行重新分段。请重试，使用 hadbm refragment 命令对数据库进行重新分段。
原因：HADB-E-11701: * 未找到表单点登录 *
```

解决方法

解决方法是使用 `clusql` 命令手动对 App Srv 表进行重新分段：

```
> 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;
```

安装

本节介绍已知的安装问题和相应的解决方法。

有时无法显示安装程序和卸载程序“欢迎”屏幕上的“下一步”导航按钮。此问题只影响本产品的独立版本。(ID 4977191)

在 Solaris x86 平台上有时报告此问题，但此问题可能也会影响 Solaris SPARC 和 Linux 平台。

此问题具体表现为：在安装程序或卸载程序的第一个屏幕上正确显示了全部文本以及“帮助”和“取消”按钮，但并未显示导航到下一个屏幕所需的“下一步”按钮。尽管未显示此按钮，但其所在区域仍处于活动状态。如果单击此区域，将正常地转至下一个屏幕。出现此问题的原因是间歇性 J2SE GUI 重画问题。

解决方法

一种解决方法是单击“帮助”按钮左侧相邻的“下一步”按钮区域。另一种解决方法是通过略微调整屏幕大小或最小化并还原安装程序窗口来强制重画此屏幕。重画之后，缺少的“下一步”按钮将显示出来。

在某些 Linux 系统上，单击“完成”按钮后安装关闭挂起。(5009728)

已在多种 Linux 系统上发现此问题。此问题在 Java Desktop System 2 上最常见，但在 RedHat 分发上也发现了此问题。

在安装程序的最后一个屏幕上单击“完成”按钮后，安装程序无法启动包含产品“关于”页面或产品注册页面的浏览器窗口，同时安装程序将无限期地挂起并且不返回命令提示符。

解决方法

通过在启动安装程序的终端窗口中按 **Ctrl+C** 组合键来退出安装程序。执行此操作后，有时会启动包含产品“关于”页面或注册页面的浏览器窗口。如果不显示此窗口，请启动浏览器并输入以下 URL 以查看“关于”页面：

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

如果您还选择了用于注册产品的安装选项，请通过产品“关于”页面上指向注册页面的链接来访问该页面。

在 Linux 上，安装包装程序中有时会出现 J2SE 检测和引导问题。(6172980)

启动 Linux 安装程序的 `setup` 可执行文件有时会挂起。包装程序不是解析 J2SE 位置并启动安装向导，而是挂起并返回以下消息：

```
正在检查可用磁盘空间 ....  
正在检查 Java(TM) 2 Runtime Environment....  
正在解压缩 Java(TM) 2 Runtime Environment....  
正在删除临时文件 .....
```

仅在某些版本的 Linux 上发现了此问题。此问题好像与环境设置有关（特别是存在 `JAVA_HOME` 变量时）。

解决方法

要解决此问题，请执行以下步骤：

1. 通过运行 `unset` 或 `unsetenv`（这取决于您的 shell）来取消设置 `JAVA_HOME` 变量。
2. 运行带有 `-javahome` 选项的 `setup` 来指定安装程序使用的 `JAVA_HOME`。

在安装过程中需要创建 `imq` 目录（仅限于 Windows）。(ID 6199697)

刚刚在 Windows 上安装 Application Server EE 之后，IMQ 代理启动失败，并显示一条消息，表明 `drive:\astomainstomain1\imq` 不存在。

请注意，如果是在启动 `domain1` 后启动代理，Application Server 将创建该目录，也就不会出现此问题。

解决方法

在创建代理之前创建 `var_home_dir_location`:

```
$imqbrokerd -varhome var_home_dir_location
```

例如:

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

J2EE Tutorial

要在 Sun Java System Application Server Enterprise Edition 8.1 2005Q1 上运行 J2EE 1.4 Tutorial, 请执行以下任务:

- 按照 “关于本教程” 一章的 “关于示例” 一节中所述编辑文件 `examples/common/build.properties` 时, 还要将端口 4848 更改为 4849。
- 使用 Deploytool 时, 在部署示例之前添加服务器 `localhost:4849`。
- 使用管理控制台创建资源时, 使用 “目标” 选项卡将服务器指定为目标。如果使用命令行或 `asant` 目标, 则该服务器为默认目标, 无需其他操作。

生命周期管理

本节介绍已知的生命周期管理问题和相应的解决方法。

将 `ejb-timer-service` 属性 `minimum-delivery-interval` 设置为 9000 之后, 如果尝试将 `ejb-timer-service` 属性 `redelivery-interval-in-millis` 设置为 7000, 将导致 `set` 命令失败, 并显示以下错误: (ID 6193449)

```
[echo] 正在执行管理任务 set
[exec] [属性 (id=redelivery-interval-internal-in-millis) : Redelivery-Interval (7,000) 应大于或等于 Minimum-delivery-interval-in-millis (9,000)]
[exec] CLI137 命令 set 失败。
```

- `minimum-delivery-interval` 是传送相同周期计时器之间的最小时间间隔。
- `redelivery-interval-in-millis` 是计时器服务在 `ejbTimeout` 失败之后、再次尝试传送之前等待的时间。

问题在于描述重新传送时间间隔属性与最小传送时间间隔属性之间关系的逻辑不正确, 使您无法使用 GUI 或 CLI 来设置使最小传送时间间隔大于重新传送时间间隔的任何值。

必须始终将 `minimum-delivery-interval-in-millis` 设置为等于或大于 `ejb-timer-service` 属性 `redelivery-interval-in-millis`。在 `Application Server` 中有一个错误的验证检查，用于验证 `redelivery-interval-in-millis` 的值是否大于 `minimum-delivery-interval-in-millis` 的值，这是产生上述问题的原因。

解决方法

使用这些属性的默认值，如下所示：

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

使用其他的值将导致产生错误。

日志记录

本节介绍已知的日志记录问题和相应的解决方法。

为 `access, failure` 设置调试语句导致在 `Application Server` 启动时挂起。
(ID 6180095)

为 JVM 设置 `java.security.debug` 选项将导致服务器实例启动停止并死锁；例如，在 `domain.xml` 中设置以下内容将导致出现此问题：

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

解决方法

目前尚无解决方法。请避免设置此标志。

Message Queue

本节介绍已知的 `Java Message Queue` 问题和相应的解决方法。

在某些与时间相关的情况下，`JMS` 重新连接无法成功完成。
(ID 6173308、6189645、6208728、6198481、6199510 或 6199510)

多种问题均可导致在与时间相关的情况下重新连接失败。但是，通常可以通过以下方式解决这些问题：

- 重新启动相关的代理
- 重新启动相关的 `Application Server` 实例

从 8.0 到 8.1, appclient 中的异步消息侦听器行为已更改 (ID 6198465)

由于最近的更改, 当异步消息侦听器为 app-client 容器中唯一的活动线程时, 其余 appclient VM 将作为守护进程存在。对于以前在 ACC 中执行异步接收的应用程序, 此行为是一种退化。此问题将影响设置 JMS 消息侦听器并退出主线程的应用程序客户机。

解决方法

不退出主线程。等待消息侦听器通知主线程, 然后再终止主线程。

消息代理日志包含“无法传送”消息。(ID 6204180)

在群集环境中运行时, 如果代理内存太小, 则代理日志中可能会出现以下错误消息:

```
内部错误, 无法传送 .....: java.lang.NullPointerException
```

对于长期使用者, 尽管使用者已确认消息, 可能仍会在以后的某个时间将消息重新传送给使用者 (设置了重新传送标志)。

出现此错误的原因是系统的内存受到限制时错误地释放了系统仍需要的数据。仅当使用传送到附加到此代理的使用者但已传播到群集中的其他代理的消息时, 才会发生此错误。

解决方法

增加消息代理进程的最大 Java 堆大小 (-Xmx), 以防止系统运行时内存不足。

在故障之后重新启动后, 群集中运行的代理耗尽内存。(ID 6205463)

作为群集的一部分运行的消息代理在重新启动后耗尽内存。导致此问题的可能原因有两个。要确定是一个还是两个原因与此有关, 请查看群集中的 Application Server 和消息代理的日志文件:

- 如果 Application Server 日志文件含有以下格式的警告消息:

```
MQRA:OMR: 端点不可用: 正在将 OMR 标记为 invalid-omrId=1 用于 :ActivationSpec 配置 ...
```

则正在触发内存问题 (如 "在某些与时间相关的情况下, JMS 重新连接无法成功完成。
(ID 6173308、6189645、6208728、6198481、6199510 或 6199510) " 中所述)。

- 如果群集中其他代理的一个或多个代理日志文件显示如下格式的消息:

```
ERROR 内部错误: 已接收 ack 两次 ...
```

则正在触发内存问题 (如 "发生故障后重新启动群集时内存存在消息代理上堆积。
(ID 6208621)" 中所述)。

解决方法

请参见为 6208621 或 6208728 (这取决于导致此问题的原因) 列出的解决方法。

发生故障后重新启动群集时内存在消息代理上堆积。(ID 6208621)

在发生故障后重新启动群集中的消息代理时，内存可能会在代理上堆积，因为未正确传播主题目标上非长期 MDB 的状态。

在消息代理作为活动群集的一部分启动后，代理和应用程序服务器日志中均出现异常。随着时间的推移，重新启动的代理将开始出现运行时内存不足。在这种情况下，群集中应用程序服务器的一个或多个日志将在代理重新启动后抛出与以下类似的警告消息：

```
[#|2004-12-03T17:45:12.821-0800|WARNING|sun-appserver-ee8.1|javax.enterprise.system.stream
.err|_ThreadID=12;|com.sun.messaging.jms.JMSEException: [C4000]: 包确认失败。用户 =admin, 代理
=<brokername>....
```

消息代理日志中通常也会出现相应的消息，其格式如下：

```
ERROR 内部错误：已接收 ack 两次...
```

此问题表现为：当消息代理附加到活动群集时，有关所有活动使用者的信息都将通过群集中的其他代理转发至此消息代理。如果 MDB 在一个远程代理上有非长期订户，它可能会在转发使用者信息时发送错误信息。一旦重新启动的代理接收到无效的使用者信息，该代理会错误地将额外的消息副本路由至其他代理。

发生这种情况时，远程使用者将在 `ApplicationServer` 日志中记录 “[C4000]: 包确认失败” 消息，远程代理上也将记录 “双 ack” 错误。代理日志中每次出现 “双 ack” 消息时，生成方代理都不会正确确认该消息。随着时间的推移，这将导致生成方代理运行时内存不足或耗尽内存。

解决方法

如果当前出现此问题，可以在系统中取消部署 MDB，然后再将其重新部署至系统，以清除内部信息。要防止此问题影响正常操作期间代理的操作，请执行以下步骤：

- 如果不考虑性能，则将 MDB 更改为使用长期（而不是非长期）订户。
- 如果考虑性能，则可以对系统进行配置，使其在一段时间内未收到确认时清除 “泄漏” 消息。可以通过以下两种方式之一执行此操作：
 - 设置 MDB 使用的目标上的 `REMOVE_OLDEST` 的 `limitBehavior` 和消息限制：

```
imqcmd update dst -n <dest name> -t t -o limitBehavior=REMOVE_OLDEST -o
maxNumMsgs=<some limit>
```

- 更改传播至主题的消息的 “生命周期” 设置。

监视

本节介绍已知的监视问题和相应的解决方法。

某些 HTTP 服务监视统计信息不提供有用信息，应该被忽略。(ID 6174518)

查看 HTTP 服务的某些元素的监视统计信息时，所提供的某些值与当前值并不对应或始终为 0。特别是，以下 HTTP 服务统计信息并不提供适用于 Application Server 8.1 的信息，应该被忽略：

- http 服务
 - load1MinuteAverage
 - load5MinuteAverage
 - load15MinuteAverage
 - rateBytesTransmitted
 - rateBytesReceived
- pwc 线程池（元素）

解决方法

在以后的版本中，将删除这些监视器并将其替换为更适当的信息。

未删除已取消部署的 EJB 模块的监视 mbean，即使该监视名称下的所有统计信息均已移动。(ID 6191092)

例如：

```
EJBModuleMonitorMap().size() = 1 即使 ejb 模块已取消部署 EJBModuleMonitor().getName() =
sqe_ejb_s1_01
```

EJB 模块和应用程序都存在这个问题。以编程方式（通过 MBeanAPI）和通过 `asadmin list/get` 执行操作后，空的监视 mbean 仍然存在。

诊断

`asadmin list -m "server.applications"` 显示以下输出：

```
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_e
jb_s1_01
```

您可以查看统计信息：

```
bin/asadmin list -m "server.applications._expo
rt_install_nov-11_domains_domain1_applications_j2ee-modules_sqe_ejb_s1_01"
server.applications._export_install_nov-11_domains_domain1_applications_j2ee-mod
ules_sqe_ejb_s1_01.SQEMessage
server.applications._export_install_nov-11_domains_domain1_applications_j2ee-mod
ules_sqe_ejb_s1_01.TheGreeter
```

一旦您取消部署：

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

如果执行 `list` 命令，您仍然可以看到应用程序：

```
asadmin list -m "server.applications"
server.applications.MEjbApp
server.applications.__ejb_container_timer_app
server.applications._export_install_nov-11_domains_domain1_applications_j2ee-mod
ules_sqe_ejb_s1_01
server.applications.adminapp
server.applications.admingui
server.applications.com_sun_web_ui
```

但它不包含任何监视统计信息：

```
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
```

要获得以某个字符串开头的有效名称，请使用通配符 ('*') 字符。例如，要列出以 `server` 开头的所
有可监视实体的名称，则使用 `list "server.*"` 命令。

解决方法

这是没有危害的。可以安全地重新部署模块而不会出现任何问题。未删除根监视 Mbean，但它为
空。

PointBase

本节介绍与 PointBase 有关的已知问题和相应的解决方法。

对应用程序的连接池设置隔离级别导致 PointBase 中出现异常。(ID 6184797)

对于指向 PointBase 数据库安装的 JDBC 连接池，将 `transaction-isolation-level` 池属性设置为默认值 (`Connection.TRANSACTION_READ_COMMITTED`) 以外的任何值都将导致异常。但是，对于指向其他数据库的池，将此参数设置为非默认值不会抛出异常。

解决方法

对于指向 PointBase 数据库安装的 JDBC 连接池，不要尝试设置 `transaction-isolation-level`。

如果同时使用网络服务器驱动程序和嵌入式驱动程序， PointBase 将抛出异常。(ID 6204925)

如果同时使用网络服务器驱动程序和嵌入式驱动程序，捆绑的 PointBase 有时会抛出异常。

解决方法

只使用嵌入式驱动程序或网络服务器驱动程序两者之一。

样例

本节介绍与 Application Server 8.1 产品附带的样例代码有关的已知问题和相应的解决方法。

`setup-one-machine-cluster` 在 Windows 上挂起，但可以在 Solaris 上工作；`mqfailover` 需要使用 `Ctrl+C` 组合键取消，然后必须重新运行。(ID 6195092)

例如，要再现此错误，请参见

`install_dir\samples\ee-samples\failover\apps\mqfailover\docs\index.html`，然后运行以下命令：

- 控制台 1

```
cd install_dir\samples\ee-samples asant start-mq-master-broker1
```
- 控制台 2

```
cd install_dir\samples\ee-samples asant start-mq-cluster-broker1
```
- 控制台 3

```
cd install_dir\samples\ee-samples asant start-mq-cluster-broker2
```
- 控制台 4

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

如果您已为任何其他 EE 样例执行了 `asant setup-one-machine-cluster-without-ha` 或 `asant setup-one-machine-cluster-with-ha`，然后执行 `asant configure-mq` 或执行 `asant setup-one-machine-cluster-and-configure-mq`。在这种情况下，命令显示为成功：

```
start_nodeagent: [echo] 启动节点代理 cluster1-nodeagent [exec] 命令 start-node-agent 执行成功。
但随后系统将无限期挂起。
```

解决方法

目前尚无解决方法。此问题同样会影响在 Windows 上使用此 ant 目标的所有 EE 样例。一种解决方法是使用 Ctrl+C 组合键退出挂起进程，然后重新运行它。

文档没有明确说明在执行 `asadmin deploy` 指令之后，需要在运行 MQ 故障转移样例应用程序之前创建 JMS 资源。(ID 6198003)

抛出的错误如下：

```
/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: 未找到 JMS 目标对象: jms/durable/TopicA
Nov 18, 2004 10:50:18 PM com.sun.enterprise.naming.NamingManagerImpl bindObjects
SEVERE: javax.naming.NameNotFoundException
javax.naming.NameNotFoundException
```

文档没有明确说明如果使用 `asadmin deploy` 命令进行了手动部署则必须手动创建 JMS 资源，并且应使用提供的 ant 目标来部署样例应用程序。

解决方法

为创建所需 JMS 资源以运行应用程序的 `build.xml` 脚本使用 `asant deploy` 目标。

在 Linux 上，在 Web 服务 / 安全性样例中创建证书过程中发生运行时错误。(ID 6198239)

在 Linux 上，部署 `install_dir/samples/webservices/security` 样例 (`basicSSL`) 时未创建证书，并抛出类似如下的错误：

```
generate_certs: [echo] *** 正在从 NSS 数据库导出证书 [exec] 结果: 1 [echo] *** 正在通过生成的证书
生成 Java 密钥库 [exec] 密钥工具错误: java.lang.Exception: 输入的不是 X.509 证书 [exec] 结果: 1
[echo] *** 正在通过生成的证书生成 Java 信任存储 [exec] 密钥工具错误: java.lang.Exception: 输入的不
是 X.509 证书 [exec] 结果: 1
.
.
.
generate_certs: [echo] *** 正在将服务器证书从 NSS 数据库导出到 PKCS12 证书文件 [exec]
/opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so: 未找到版本 `NSS_3.9'
```

```
(/opt/sun/appserver/lib/pk12util 需要) [exec] /opt/sun/appserver/lib/pk12util:
/usr/lib/libnss3.so: 未找到版本 `NSS_3.6' (/opt/sun/appserver/lib/pk12util 需要) [exec]
/opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so: 未找到版本 `NSS_3.7'
(/opt/sun/appserver/lib/pk12util 需要) [exec] 结果: 1
```

问题在于 NSS 库在 Linux 安装上的位置与其在 Solaris 安装上的位置不同。在 Linux 上部署时，必须确保 LD_LIBRARY_PATH 指向正确的 NSS 库。在您的环境中设置 LD_LIBRARY_PATH，或者在 `install_dir/bin/asant` shell 包装程序脚本中对其进行设置。

解决方法

执行以下操作之一：

- 设置 LD_LIBRARY_PATH=/opt/sun/private/lib。
- 将以下行添加至 `install_dir/bin/asant` 脚本：

```
LD_LIBRARY_PATH=$AS_NSS:$LD_LIBRARY_PATH;export LD_LIBRARY_PATH
```

- `ee samples asadmin deploy` 命令的文档遗漏了用于部署应用程序的 `availabilityenabled=true` 选项，此选项可以确保样例应用程序在默认情况下启用 HADB。(ID 6198796)

解决方法

使用 `asadmin` 命令：

1. 转至样例目录的根，例如：

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

2. 执行 `asadmin deploy` 以将应用程序部署至本地 Application Server 实例；例如：

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

对所有其他 EE 样例（`install_dir/samples/ee-samples/failover/apps/mqfailover` 除外）执行相同的 `asadmin deploy` 命令操作。请注意，MQ 不使用 HADB。

无法使用 asant 脚本为 dukesbookstore EE 样例运行 failover 测试。(ID 6199076)

设置包含两台计算机的群集后，`dukesbookstore failover` 样例遇到错误。此处的方法是每个群集使用一个数据库。当前，如果群集中的实例在两台单独的计算机上运行，则通过此群集中的 DAS 部署样例时，脚本将使用 PointBase 主机作为 localhost。部署 EE 样例时，JDBC 资源使用 PointBase 主机作为 localhost 部署到两个实例。因此，两个实例上的 localhost:9092 要求 PointBase 在两台计算机上运行。

问题在于属于同一群集的两个实例不能使用不同的数据库。为了解决此问题，如果用一个主机名替换 `database.properties` 中的 `localhost`，群集中的两个实例都将能够访问数据库：一个通过 `localhost`，另一个通过指定的主机名。

解决方法

编辑运行 PointBase 服务器的主机上的 `install_dir/samples/database.properties` 文件，将 `pointbase.server` 的值设置为运行 PointBase 的主机的主机名（而不是 `localhost`）。

当前的设置（即 `pointbase.server=localhost`）适用于只包含一台计算机的群集，而不适用于包含两台计算机的群集（其中对于第二个实例来说，PointBase 不能在 `localhost` 上运行）。

在一个 Ant 设置目标中，MQ-failover 样例应用程序有硬编码为 "cluster1" 的群集名称。(ID 6202363)

在一个 Ant 设置目标中，MQ-failover 样例应用程序有硬编码的 `cluster1`。因此，如果将 `ee-samples` 中的 `cluster.properties` 中的 `cluster-name` 从 `cluster1` 修改为其他名称，样例在尝试在 `cluster1` 中设置默认主机时将失败：

```
set-default-jms-host-to-broker1:
[echo] 正在将默认 JMS 主机设置为 samplesbroker1
[exec] 没有与指定的主机名 "cluster1-config.jms-service.default-jms-host" 匹配的对象
[exec] CLI137 命令 set 失败。
```

`install_dir/samples/ee-samples/build.xml` 文件硬编码了对 `cluster1` 的引用，替换该名称时应使用 `install_dir/samples/ee-samples/cluster.properties` 中的 `${cluster.name}`。

解决方法

必须使用 `${cluster.name}` 替换硬编码的 `cluster1` 字符串。手动修改

`install_dir/samples/ee-samples/build.xml`，以便将 `set-default-jms-host-to-broker1` 从 `cluster1` 更改为 `cluster.properties` 中指定的 `${cluster.name}` 或 `customer` 群集名称。

安全性

本节介绍与 Application Server 和 Web 应用程序安全性和证书有关的已知问题和相应的解决方法。

无法通过 java-method 在 client-side message-security-binding 元素中指定目标消息。(ID 6155080)

例如，在 service-ref 元素内通过 port-info 元素内的 java-method 指定 client-side message-security-binding 元素中的目标消息时就会出现上述问题：

```
<!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? )>
```

此处使用 message-security-binding 元素来为 Web 服务端点的特定方法定义消息保护策略。

解决方法

在 message 元素内使用 operation-name 元素，以便通过 WSDL 操作名称来标识为其在包含的 message-security 元素中定义保护策略的消息。

如果 DAS 和远程节点代理计算机时钟不同步，则抛出 "CertificateNotYetValidException"。(ID 6181989)

如果 DAS 服务器和节点代理安装在不同的计算机上，并且这些计算机上的时钟不同步，则尝试运行 `asadmin --start remote-node-agent` 命令将失败，并出现 `CertificateNotYetValidException` 错误。

解决方法

同步 DAS 服务器和所有远程节点代理计算机上的时钟。

无法在 EE 上使用 J2SE 5.0 运行 WebServiceSecurity 应用程序。(ID 6183318)

无法使用 J2SE 5.0 运行 WebServiceSecurity 应用程序，原因如下：

- J2SE 5.0 PKCS11 不支持 UNWRAP 模式
- J2SE 5.0 PKCS11 不支持使用 PKCS11 的 RSA/ECB/OAEPWithSHA1AndMGF1Padding

J2SE 小组已针对此错误归档 “CR 6190389: 为 RSA-PKCS1 和 RSA-OAEP 包装 / 解包机制添加支持”。

解决方法

使用带有任何其他 JCE 提供者（而不是默认包含的提供者）的 J2SE 1.4.2。请注意，此配置中将不提供对硬件加速器的支持。

如果在 `jms-servicSSL` 中设置了 `mq-scheme` 和 `mq-service`，将无法与 MQ 进行 SSL 通信；如果在 `jms-service` 中设置了 `mq-scheme` 和 `mq-service`，将无法与 MQ 进行通信。(ID 6202606)

没有在资源适配器中设置使用的信息，从而使通过资源适配器创建的连接工厂无法使用这些信息。因此，通过资源适配器创建的连接工厂将不具有 SSL 信息。

解决方法

如果需要在 Application Server 和 MQ 之间使用 SSL 通信，请创建使用 SSL 语法明确指定了 `addresslist` 的连接工厂。例如，连接工厂中的 `addresslist` 可以为：

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

MQ 和 Application Server 位于不同位置并且入侵者可以利用它们之间的网络连接时，可能需要 Application Server 和 MQ 之间的 SSL 通信。

未使用端口号指定的指向 https 侦听器的 URL 重定向至位于端口 80 的 http 侦听器 (ID 6207297)

在默认端口 (443) 启用 SSL 侦听器时，在浏览器中指定指向安全端口但不带端口号的 URL 将导致浏览器重定向至非安全 (http) 侦听器的端口 80。

例如：

1. 在端口 443 上创建一个 SSL 侦听器，然后重新启动 Application Server。
2. 将浏览器指向 <https://servername:443>。
页面将正确装入
3. 将浏览器指向 <https://servername>（无端口号）。

浏览器将装入 <http://servername:80>，而不是 <https://servername>。

SSL 侦听器位于默认端口 (443) 以外的端口上时，将不会出现此问题。

解决方法

可以选择以下两种解决方法之一：

- 建议站点访问者并对所有链接进行编码，使其在访问 https 端口 443 上的 SSL 侦听器时始终指定端口号。
- 对于所有 SSL 侦听器，不使用端口 443。

升级实用程序

本节介绍已知的升级实用程序问题和相应的解决方法。

从 Application Server Enterprise Edition 8 升级至 Application Server Enterprise Edition 8.1 时，不直接升级在 *install_dir*/domains 目录以外的自定义路径中创建的域。(ID 6165528)

运行升级实用程序并将 *install_dir* 标识为源安装目录时，升级进程只升级在 *install_dir*/domains 目录下创建的域。在其他位置创建的域不会被升级。

解决方法

启动升级进程前，将所有域目录从不同位置复制到 *install_dir*/domains 目录下。

在从 J2EE 1.4 SDK 升级至 Application Server EE 8.1 的过程中，未正确升级捆绑的 J2SE 安装。此问题只影响本产品的**独立版本**。(ID 6196741)

在从 J2EE 1.4 SDK 升级至 Application Server 8.1 独立安装过程中，会出现此问题。在此升级过程中，捆绑的 J2SE 1.4.2 应升级至 J2SE 5.0。但是，最终的 J2SE 5.0 安装中的某些 JAR 文件未正确升级，导致 J2SE 5.0 安装损坏。

安装程序将报告升级成功，并且在升级进程中运行升级工具时看不出将遇到任何问题。但是，接下来尝试启动升级后的 Application Server 却失败，并抛出以下异常：

```
线程 "main" 中出现异常
[#|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 位于
com.sun.enterprise.security.SecurityLifecycle.onInitialization(SecurityLifecycle.java:59)
位于
com.sun.enterprise.server.ApplicationServer.onInitialization(ApplicationServer.java:215) 位
于 com.sun.enterprise.server.PEMain.run(PEMain.java:277) 位于
com.sun.enterprise.server.PEMain.main(PEMain.java:219)
```

解决方法

此问题有以下几种解决方法：

- 安装适当的受支持版本的独立 J2SE。在 Application Server 升级期间，在 Application Server 安装程序的“Java 配置”屏幕中选择用于重新使用现有 Java 2 SDK 的选项，而不要选择用于安装 Java 2 SDK 5.0 的默认选项。提供独立 J2SE 安装的路径。
- 运行升级程序之前，删除或重命名现有的 J2EE 1.4 SDK *install_dir*/jdk 子目录。从而可以在 Application Server 8.1 升级进程中正确安装 J2SE 5.0。请注意，您只能在已浏览安装程序的目录选择屏幕并且已提示您允许对现有安装执行“就地升级”之后才应当删除或重命名此目录。

- 如果已执行升级并且遇到了服务器启动问题，请安装适当的受支持版本的独立 J2SE，然后修改 *install_dir/config/asenv.conf* 文件（Linux 和 Solaris）或 *install_dir\config\asenv.bat* 文件（Windows）中的 AS_JAVA 变量。此变量的值应该指向新的独立 J2SE 安装的位置。

在某些 Linux 系统中，单击“启动升级向导”按钮之后，执行“就地升级”的安装程序无法启动升级工具。(6207337)

此问题已在多个 Linux 系统上出现，是 Java Desktop System 2 上最常见的问题，但在 RedHat 版本中也发现了此问题。

在最终安装程序屏幕上单击“启动升级工具”按钮后，安装程序无法启动升级工具以完成升级过程，而且无限期挂起，并未返回命令提示符。

解决方法

如果使用命令行安装模式来运行就地升级，将不会遇到此问题。

1. 如果您以 GUI 模式运行就地升级并且遇到此问题，请通过在启动安装程序的终端窗口中按 **Ctrl+C** 组合键来退出安装程序。
2. 使用以下命令从终端窗口启动升级工具：

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

adminuser 和 *adminpassword* 的值应匹配要升级的安装所使用的值。

3. 在升级工具完成升级过程后，您还可以启动浏览器并输入以下 URL 来查看“关于”页面：
file://install_dir/docs-ee/about.html

如果您还选择了用于注册产品的安装选项，请点击产品“关于”页面上提供的指向注册页面的链接。

Web 容器

本节介绍已知的 Web 容器问题和相应的解决办法。

使用 `--precompilejsp=true` 部署应用程序时，会锁定该应用程序中的 JAR 文件，从而导致以后无法取消部署或重新部署。（仅适用于 Windows）(ID 5004315)

如果您在 Windows 上部署应用程序时要求预编译 JSP，则以后尝试取消部署该应用程序或重新部署该应用程序（或任何具有相同模块 ID 的应用程序）的操作将不会按预期进行。出现此问题的原因是：JSP 预编译会打开应用程序中的 JAR 文件，但不能关闭这些文件，Windows 将禁止执行取消部署或重新部署操作以避免删除或覆盖它们。

请注意，取消部署在某种程度上是成功的，因为应用程序从 **Application Server** 中被逻辑删除。另外，请注意，**asadmin** 实用程序不会返回任何错误消息，但应用程序的目录以及锁定的 **jar** 文件将保留在服务器中。服务器的日志文件将包含用于说明未能删除文件和应用程序的目录的消息。

在取消部署后尝试重新部署应用程序的操作会失败，这是由于服务器尝试删除现有文件和目录，而这些尝试也失败了。如果您尝试部署的应用程序所使用的模块 **ID** 与最初部署的应用程序的模块 **ID** 相同，会出现这种情况，这是由于服务器在选择目录名来保存应用程序的文件时会使用模块 **ID**。

如果没有先取消部署应用程序而尝试重新部署该应用程序，也将会由于同样的原因而失败。

诊断

如果您要尝试重新部署应用程序或在取消部署后部署它，**asadmin** 实用程序将返回一个类似于如下所示的错误。

```
运行命令时出现异常。异常消息是：CLI171 命令部署失败：在域中部署应用程序失败，无法部署。模块目录被锁定，无法将其删除
```

解决方法

如果您在部署应用程序时指定 `--precompilejsp=false`（默认设置），则不会出现此问题。请注意，第一次使用应用程序时会触发 **JSP** 编译，因此第一个请求的响应时间将会长于随后的请求的响应时间。

另外，请注意，如果您确实进行了预编译，则在取消部署或重新部署应用程序之前，应先停止并重新启动服务器。关闭服务器后将释放锁定的 **JAR** 文件，这样在重新启动服务器后，取消部署或重新部署便可以成功。

无法使用基于 Servlet 2.4 的 web.xml（包含空 `<load-on-startup>` 元素）来部署 WAR。(ID 6172006)

web.xml 中的可选 `load-on-startup` **servlet** 元素表示的是：相关的 **servlet** 将作为对其进行声明的 **Web** 应用程序启动的一部分被装入和初始化。

此元素的可选内容是一个整数，用于表示该 **servlet** 相对于 **Web** 应用程序的其他 **servlet** 而被装入和初始化的顺序。空 `<load-on-startup>` 表示顺序无关紧要，只要该 **servlet** 在其包含 **Web** 应用程序的启动期间被装入并进行了初始化就可以。

web.xml 的 **Servlet 2.4** 模式不再支持空 `<load-on-startup>`，这意味着在使用基于 **Servlet 2.4** 的 **web.xml** 时，必须指定一个整数。如果像在 `<load-on-startup/>` 中那样指定空 `<load-on-startup>`，则 **web.xml** 针对 **web.xml** 的 **Servlet 2.4** 模式的验证将会失败，从而导致 **Web** 应用程序的部署失败。

向下兼容性问题。指定空 `<load-on-startup>` 在基于 **Servlet 2.3** 的 **web.xml** 中仍起作用。

解决方法

在使用基于 Servlet 2.4 的 web.xml 时，指定 `<load-on-startup>0</load-on-startup>`，以表明 servlet 装入顺序无关紧要。

使用 AMX API 时，从服务器中删除 J2EE 应用程序引用将会删除该应用程序，但仍然可以访问该应用程序。(ID 6173248)

使用 AMX API 时，在没有首先明确停止应用程序的情况下，删除对该应用程序的引用将导致该应用程序仍处于可访问状态。此行为是故意设计的，并且在文档中被忽略。

解决方法

要删除某个应用程序使其不再可访问，请执行以下步骤：

1. 停止应用程序
2. 删除对应用程序的引用。
3. 取消部署应用程序。

无法编译资源受限服务器上的 JSP 页面。(ID 6184122)

已访问 JSP 页面但是无法对其进行编译，并且服务器日志包含错误消息“无法执行命令”和以下堆栈跟踪：

```
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)
```

解决方法

将 JSP 编译开关 "fork" 设置为 "false"。

可以通过以下两种方式之一来实现：

- 全局设置，通过将 `/${S1AS_HOME}/domains/domain1/config/default-web.xml` 中 JspServlet 的 fork init 参数设置为 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>
```

- 在每个 Web 应用程序基础上，通过将 sun-web.xml 中的 fork JSP 配置属性设置为 false：

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

以上任何一种设置都将阻止 ant 产生用于 javac 编译的新进程。

如何报告问题和提供反馈

如果您在使用 Sun Java System Application Server 时遇到了问题，请用以下方式之一与 Sun 客户支持联系：

- **反馈提交表单**——用于提交有关 Application Server 产品反馈的表单，位于：
<http://java.sun.com/docs/forms/J2EE14SubmittalForm.html>
- **J2EE-INTEREST 列表**——有关 J2EE 问题的邮件列表，位于：
<http://archives.java.sun.com/archives/j2ee-interest.html>
- **Java Developer Connection 的错误数据库**——要查看错误或提交错误，请使用位于以下位置的 Java Developer Connection 的错误展示：
<http://developer.java.sun.com/servlet/SessionServlet?url=/developer/bugParade/index.jshtml>
- **Java 技术论坛**——用来共享关于 Java 技术和编程技术的知识和问题的交互式留言板。使用位于以下位置的 J2EE SDK 论坛可以开展与 Sun Java System Application Server Enterprise Edition 8.1 产品有关的讨论：
<http://forum.java.sun.com/>
- **Sun 软件支持服务**——网址为：
<http://www.sun.com/service/sunone/software>

此站点上有一些链接，通过这些链接可以访问知识库、联机支持中心和产品跟踪器，还可了解维护方案以及用于联系支持部门的电话号码。

- 随维护合同一起分发的电话号码

为了使我们能更好地帮助您解决问题，请在联系客户支持时提供以下信息：

- 问题的说明，包括问题发生时出现的情况以及它对操作的影响
- 计算机类型、操作系统版本，以及产品版本，包括可能导致问题的任何修补程序和其他软件

- 操作的详细步骤，以便再现问题
- 任何错误日志或信息转储

Sun 欢迎您提出意见

Sun 致力于提高其文档的质量，并十分乐意接收到您的意见和建议。

为了分享您的意见和建议，请访问 <http://docs.sun.com> 并单击 "Send Comments"。在联机表单中，提供了文档标题和文件号码。文件号码是一个七位或九位的数字，可以在书的标题页或文档的顶部找到。例如，本书的标题是《Sun Java System Application Server Enterprise Edition 8.1 2005Q1 发行说明》，文件号码是 819-2028。在您提出意见时，可能需要在表单中输入英文版书名和文件号码，本书的英文版文件号码和书名为：819-0214， Sun Java System Application Server Enterprise Edition Release Notes。

其他 Sun 资源

可以在以下位置找到有用的信息：

- Application Server 产品信息：
http://www.sun.com/software/products/appsrvr/home_appsrvr.html
- Application Server 产品文档：
<http://docs.sun.com/db/prod/slappsrv#hic> 和
<http://docs.sun.com/db/prod/slappsrv#hic?l=zh>
- Sun Java System 文档
<http://docs.sun.com/prod/java.sys> 和
<http://docs.sun.com/db/prod/java.sys?l=zh>
- Sun Java System 专业服务
<http://www.sun.com/service/sunps/sunone>
- Sun Java System 软件产品和服务
<http://www.sun.com/software>

- Sun Java System 软件支持服务
<http://www.sun.com/service/sunone/software>
- Sun Java System 支持和知识库
<http://www.sun.com/service/support/software>
- Sun 支持和培训服务
<http://training.sun.com>
- Sun Java System 咨询和专业服务
<http://www.sun.com/service/sunps/sunone>
- Sun Java System 开发者信息
<http://developers.sun.com>
- Sun 开发者支持服务
<http://www.sun.com/developers/support>
- Sun Java System 软件培训
<http://www.sun.com/software/training>
- Sun 软件数据表
<http://www.sun.com/software>
- Sun Microsystems 产品文档:
<http://docs.sun.com/>

版权所有 © 2004-2005 Sun Microsystems, Inc. 保留所有权利。

对于本文档中介绍的产品，Sun Microsystems, Inc. 对其所涉及的技术拥有相关的知识产权。需特别指出的是（但不局限于此），这些知识产权可能包含在 <http://www.sun.com/patents> 中列出的一项或多项美国专利，以及在美国和其他国家 / 地区申请的一项或多项其他专利或待批专利。

SUN 专有 / 机密。

美国政府权利——商业软件。政府用户应遵循 Sun Microsystems, Inc. 的标准许可协议，以及 FAR（Federal Acquisition Regulations，即“联邦政府采购法规”）的适用条款及其补充条款。

必须依据许可证条款使用。

本软件可能包括由第三方开发的产品。

本产品的某些部分可能是从 Berkeley BSD 系统衍生出来的，并获得了加利福尼亚大学的许可。

Sun、Sun Microsystems、Sun 徽标、Java 和 Solaris 是 Sun Microsystems, Inc. 在美国和其他国家 / 地区的商标或注册商标。所有 SPARC 商标的使用均已获得许可，它们是 SPARC International, Inc. 在美国和其他国家 / 地区的商标或注册商标。