



Sun Java System Application Server Platform Edition 8.2 发行 说明



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

文件号码 819-5897

版权所有 2006 Sun Microsystems, Inc. 4150 Network Circle, Santa Clara, CA 95054 U.S.A. 保留所有权利。

对于本文中介绍的产品，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一项或多项美国专利，或者在美国和其他国家/地区申请的待批专利。

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。

本发行版可能包含由第三方开发的内容。

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。UNIX 是 X/Open Company, Ltd. 在美国和其他国家/地区独家许可的注册商标。

Sun、Sun Microsystems、Sun 徽标、Solaris 徽标、Java 咖啡杯徽标、docs.sun.com、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。所有的 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。标有 SPARC 商标的产品均基于由 Sun Microsystems, Inc. 开发的体系结构。

OPEN LOOK 和 Sun™ 图形用户界面是 Sun Microsystems, Inc. 为其用户和许可证持有者开发的。Sun 感谢 Xerox 在研究和开发可视或图形用户界面的概念方面为计算机行业所做的开拓性贡献。Sun 已从 Xerox 获得了对 Xerox 图形用户界面的非独占性许可证，该许可证还适用于实现 OPEN LOOK GUI 和其他方面遵守 Sun 书面许可协议的 Sun 许可证持有者。

本发行说明所介绍的产品以及所包含的信息受美国出口控制法制约，并应遵守其他国家/地区的进出口法律。严禁将本产品直接或间接地用于核设施、导弹、生化武器或海上核设施，也不能直接或间接地出口给核设施、导弹、生化武器或海上核设施的最终用户。严禁出口或转口到美国禁运的国家/地区以及美国禁止出口清单中所包含的实体，包括但不限于被禁止的个人以及特别指定的国家/地区的公民。

本文档按“原样”提供，对于所有明示或默示的条件、陈述和担保，包括对适销性、适用性或非侵权性的默示保证，均不承担任何责任，除非此免责声明的适用范围在法律上无效。

目录

1 概述	7
关于本说明	7
发行说明修订历史记录	8
为残疾人士提供的辅助功能	8
相关文档	8
如何报告问题和提供反馈	9
Sun 欢迎您提出意见	10
其他 Sun 资源	10
2 关于 Application Server Platform Edition 8.2	13
8.2 发行版的新增功能	13
硬件和软件要求	14
平台要求	14
重要修补程序信息	15
JDBC 驱动程序和数据库	15
使用捆绑的 Derby 数据库	16
浏览器	19
升级 Sun Java System Application Server	19
其他要求	20
8.2 发行版中修复的错误	20
J2EE 支持	22
切换到其他支持的 J2SE 版本	23
▼ 切换到另一个支持的 J2SE 版本	23
3 已知问题和限制	25
管理	25
如果不存在 domain1，则 package-appclient 脚本将不起作用。(ID 6171458)	25
无法恢复具有其他名称的备份域。(ID 6196993)	26

不支持启动带有附加 JMX 代理的 Application Server。(ID 6200011)	26
如果某个 Web 模块是任一虚拟服务器的默认 Web 模块，则无法对其进行重新部署或取消其部署。(ID 6204799)	26
▼更改默认 Web 模块	27
通过 Application Server GUI 中的 AMX API 将 WAR 和 JAR 部署到 PE 服务器后，出现 FrameworkError 异常。(ID 6201462)	27
配置中的 Java 主目录设置不起作用。(ID 6240672)	27
Selector.select() 抛出 IOException。Application Server 启动失败。(ID 6322825)	28
create-domain 主密码具有特殊字符时，域无法启动。(ID 6345947)	28
AS 8.2 启动时无法正确处理特定的 Java System 属性。(ID 6372759)	28
应用程序客户机	29
封装在应用程序客户机归档文件中的库 JAR 覆写了 MANIFEST 文件。(ID 6193556)	29
不支持诸如 CGI-bin 和 SHTML 功能之类的动态内容技术。(ID 6373043)	29
数据库驱动程序	29
使用 DB2 类型 II 驱动程序时，空闲超时后 DB2 服务器连接数不断增加。(ID 2082209/5022904)	29
Deploytool	29
Deploytool 通常不会在以下 Sun 部署描述符中创建 message-destination 元素 (ID 6197393) :	29
▼编辑现有 JNDI 名称	30
在简体中文版 Deploytool 中，“Home”被错译为“安装目录”。(ID 6203658)	30
文档	30
某些记录的监视功能不适用于 Platform Edition。(ID 6202255)	31
在文档中，AppservPasswordLoginModule 被引用为 AbstractPasswordLoginModule。(ID 6229682)	31
在 8.2 PE 手册页中，--passwordfile 不正确地具有了简短选项 -w。(ID 6373588)	32
多个 AMX 接口和方法的 Javadoc 缺少或不正确（若干 ID）：	32
安装	32
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。(ID 4977191) ...	32
在某些 Linux 系统上，单击“完成”按钮后安装结束过程挂起。(5009728)	32
在 Linux 上安装包装程序时，有时会出现 J2SE 检测和引导问题。(6172980)	33
▼解决 Linux 上的引导问题	33
生命周期管理	33
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-mills 设置为 7000，将导致 set 命令失败，并显示以下错误消息：(ID 6193449)	33
日志记录	34
为 access.failure 设置调试语句会导致 Application Server 在启动时挂起。(ID 6180095) ..	34

样例应用程序	34
managementws 样例需要将 MANIFEST.MF 引用从 castor-0.9.3.9-xml.jar 更新为 castor-0.9.9.1.jar。(ID 6363339)	34
安全性	36
WS 安全性：应用程序客户机容器无法与 JAXRPC 客户机运行时正确集成。(ID 6325469)	36
升级实用程序	36
从 Application Server Platform Edition 8 升级到 Application Server Platform Edition 8.2 时， 不会直接升级在自定义路径（而非 <i>install_dir/domains</i> 目录）中创建的域。(ID 6165528)	36
从 8.0 Platform Edition 升级到 8.2 Platform Edition 后，启动 domain1 或 samples 域时发生 端口冲突。(ID 6202188)	37
▼更改端口值	37
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)	37
▼使用命令行安装模式	37
升级后在“结果”面板中出现乱码 (ID 6376140)	38
Web 容器	38
使用 <code>--precompilejsp=true</code> 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。（仅限于 Windows）(ID 5004315)	38
无法使用基于 Servlet 2.4 的 web.xml（包含空 <code><load-on-startup></code> 元素）来部署 WAR。 (ID 6172006)	39
无法编译资源受限服务器上的 JSP 页面。(ID 6184122)	39
在多 CPU 计算机上运行时性能降低。(ID 6194026)	40
如果收到的 Fast Infoset 文档不规则，则会禁用 Fast Infoset 对已部署的 JAX-RPC 服务的 支持。(ID 6368670)	40

◆ ◆ ◆ 第 1 章

概述

Sun Java™ System **Application Server Platform Edition 8.2** 产品是与 J2EE 1.4 平台兼容的服务器，它用于开发和部署 J2EE 应用程序和 Java Web 服务。该服务器可以免费用于生产。Sun Java System Application Server Platform Edition 可免费用于开发、部署和再分发。有兴趣进行再分发的客户应与 Sun OEM 销售 (http://www.sun.com/software/products/appsrvr/appsrvr_oem.html) 联系以获取再分发许可证。

本文档包括以下内容：

- 第 7 页中的 “关于本说明”
- 第 8 页中的 “发行说明修订历史记录”
- 第 8 页中的 “为残疾人士提供的辅助功能”
- 第 8 页中的 “相关文档”
- 第 9 页中的 “如何报告问题和提供反馈”
- 第 10 页中的 “Sun 欢迎您提出意见”
- 第 10 页中的 “其他 Sun 资源”

关于本说明

本发行说明包含 Sun Java System Application Server 8.2 发行时可用的重要信息，其中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 8.2 之前，请先阅读本文档。

可以在 Sun Java System 文档 Web 站点 (<http://docs.sun.com/app/docs/coll/1310.1>) 和 (<http://docs.sun.com/app/docs/coll/1386.1>) 上找到本发行说明的最新版本。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。

本文档引用了第三方 URL 以提供其他相关信息。

注 - Sun 对本文中提到的第三方 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、产品或其他资料，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。

发行说明修订历史记录

本节列出了本发行说明的修订历史记录。

表 1-1 发行说明修订历史记录

修订日期	说明
2006 年 1 月	Sun Java System Application Server 8.2 产品的初始版。
2006 年 2 月	常规编辑处理，更新了 Red Hat 修补程序 RPM 的 URL。
2006 年 3 月	记录了其他问题，更新了已修复错误的列表。

为残疾人士提供的辅助功能

欲获得自本介质发行以来所发布的辅助功能，请联系 Sun 索取有关 "Section 508" 法规符合性的产品评估文档，以便确定哪些版本最适合部署辅助功能解决方案。可通过以下网址获取应用程序的更新版本：<http://sun.com/software/javaenterprisesystem/get.html>。

有关 Sun 在辅助功能方面所做出的努力，请访问 <http://sun.com/access>。

相关文档

除了本发行说明以外，Application Server 产品还包括一整套文档 (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 Application Server 核心应用程序文档集中包含的文档。

表 1-2 本文档集中的文档

书名	说明
《Sun Java System Application Server Platform Edition 8.2 Quick Start Guide》	Sun Java System Application Server 产品如何入门。

表 1-2 本文档集中的文档 (续)

书名	说明
《Sun Java System Application Server Platform Edition 8.2 Installation Guide》	安装 Sun Java System Application Server 软件及其组件。
《Sun Java System Application Server Platform Edition 8.2 Developer's Guide》	创建和实现要在 Sun Java System Application Server 中运行的 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。
《Sun Java System Application Server Platform Edition 8.2 J2EE Tutorial》	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。
《Sun Java System Application Server Platform Edition 8.2 Administration Guide》	从管理控制台配置、管理和部署 Sun Java System Application Server 子系统和组件。
《Sun Java System Application Server Platform Edition 8.2 Administration Reference》	编辑 Sun Java System Application Server 配置文件 domain.xml。
《Sun Java System Application Server Platform Edition 8.2 Upgrade and Migration Guide》	将应用程序迁移到新的 Sun Java System Application Server 编程模型, 特别是从 Application Server 6.x 和 7 进行迁移。该指南还介绍了可导致与产品规格不兼容的相邻产品发行版和配置选项之间的差异。
《Sun Java System Application Server Platform Edition 8.2 Troubleshooting Guide》	解决 Sun Java System Application Server 问题。
《Sun Java System Application Server Platform Edition 8.2 Error Message Reference》	解决 Sun Java System Application Server 错误消息。
《Sun Java System Application Server Platform Edition 8.2 Reference Manual》	Sun Java System Application Server 中提供的实用程序命令; 按手册页样式编写。其中包括 asadmin 命令行界面。

如何报告问题和提供反馈

如果您在使用 Sun Java System Application Server 期间遇到问题, 请通过以下方式与 Sun 客户支持部门联系。

- 反馈提交表单—用于提交有关 Application Server 产品反馈的表单 (<http://java.sun.com/docs/forms/J2EE14SubmittalForm.html>)。
- **J2EE-INTEREST** 列表—有关 J2EE 问题的邮件列表 (<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 Java Developer Connection 的 Bug Parade（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 J2EE SDK 论坛 (<http://forum.java.sun.com/>) 可以讨论与 Sun Java System **Application Server Platform Edition 8.2** 产品相关的问题。
- **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，并十分乐意收到您的意见和建议。

为了共享您的意见，请访问 <http://docs.sun.com>，并单击 "Send Comments"（发送意见）。在联机表单中，请提供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，本书的书名为《Sun Java System Application Server Platform Edition 8.2 发行说明》，文件号码是 819-5897。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：819-4707，《Sun Java System Application Server Platform Edition 8.2 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/app/docs/coll/1310.1>) 和 (<http://docs.sun.com/app/docs/coll/1386.1>)
- Sun Java System 文档 (<http://docs.sun.com/prod/java.sys>) 和 (<http://docs.sun.com/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/>)

关于 Application Server Platform Edition 8.2

Sun Java™ System **Application Server Platform Edition 8.2** 是与 J2EE 1.4 平台兼容的服务器，它用于开发和部署 J2EE 应用程序和基于 Java 技术的 Web 服务。

本章包括：

- 第 13 页中的 “8.2 发行版的新增功能”
- 第 14 页中的 “硬件和软件要求”
- 第 20 页中的 “8.2 发行版中修复的错误”
- 第 22 页中的 “J2EE 支持”
- 第 23 页中的 “切换到其他支持的 J2SE 版本”

8.2 发行版的新增功能

Sun Java System Application Server Platform Edition 8.2 实现了很多新增功能：

- 与 J2EE 1.4 兼容。
- （改进）利用 NetBeans 5 以及改进后的部署速度、运行时处理器占用量和服务器启动时间，开发者可以享受非凡的体验。NetBeans 5.0 将 Application Server 8.2 作为默认 J2EE 运行时包括在内。
- （新增）卓越的吞吐性能以及更好的可伸缩性，可支持多核心/多线程体系结构。
- （新增）支持 Fast Infoset 从而大大提高 Web 服务性能。
- （新增）Application Server 8.2 新增对 Red Hat 4 和 My SQL 5 的支持。有关支持的操作系统和数据库驱动程序的完整列表，请参见本发行说明后面的第 14 页中的 “硬件和软件要求”。
- （新增）Application Server 8.2 捆绑了部署就绪的 Derby 数据库，从而可以开发和部署端对端 J2EE 应用程序。
- （新增）通过内置 JMS 资源适配器更加方便地与后端系统进行连接。Application Server 8.2 支持到 IBM MQ Series 和 Sun 的 MQ Server 的连接。
- （更新）改进的 J2EE 1.4 Blueprint，其中包括将 Web 2.0 技术（例如 AJAX）与 Application Server 8.2 配合使用的演示。

- JavaServer Faces 支持—通过在页面中汇编可重复使用的 UI 组件、将这些组件连接至数据源以及将客户机生成的事件与服务器端事件处理程序相连，开发者可以迅速构建 Web 应用程序。
- 改进的管理控制台—Application Server 8.2 管理控制台添加了新的查看和监测功能，例如 JNDI 名称空间浏览器、JDBC 连接验证、部署描述符查看器、错误日志查看器和改进的监视功能。
- Web 服务安全性：容器消息安全性机制通过使用 OASIS WS-Security 标准的 X509 和用户名/密码配置文件，来实现 SOAP Web 服务调用的消息级别验证（例如 XML 数字签名和加密）。
- JavaServer Pages 标准标记库 1.1 支持：该库封装了许多 JSP 应用程序通用的核心功能。
- （更新）捆绑了 J2SE 5.0_06。

Sun Java System Application Server Platform Edition 可免费用于开发、部署和再分发。有兴趣进行再分发的客户应与 Sun OEM 销售 (http://www.sun.com/software/products/appsrvr/appsrvr_oem.html) 联系以获取再分发许可证。Sun 提供需额外付费的支持。如果要进行联系以获取有关 Java 2 Platform, Enterprise Edition 的许可证，请填写此表单 (http://java.sun.com/j2ee/license_form.html)。

硬件和软件要求

本节列出了安装 Sun Java System Application Server Platform Edition 8.2 产品之前必须满足的要求。

- 第 14 页中的“平台要求”
- 第 15 页中的“重要修补程序信息”
- 第 15 页中的“JDBC 驱动程序和数据库”
- 第 16 页中的“使用捆绑的 Derby 数据库”
- 第 19 页中的“浏览器”
- 第 19 页中的“升级 Sun Java System Application Server”
- 第 20 页中的“其他要求”

平台要求

下表列出了 Sun Java System Application Server Platform Edition 8.2 产品所支持的操作系统。

表 2-1 支持的操作系统

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	512 MB	250 MB 可用空间	500 MB 可用空间	J2SE 1.4.2_10, J2SE 5_06
Sun Java Desktop System	512 MB	1 GB	250 MB 可用空间	500 MB 可用空间	J2SE 1.4.2_10, J2SE 5_06

表 2-1 支持的操作系统 (续)

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
Redhat Enterprise Linux 3.0 U1, 4.0	512 MB	1 GB	250 MB 可用空间	500 MB 可用空间	J2SE 1.4.2_10, J2SE 5_06
Windows Server 2000 SP4+ Windows 2000 Advanced Server SP4+ Windows Server 2003 Windows XP Pro SP1+	1 GB	2 GB	500 MB 可用空间	1 GB 可用空间	J2SE 1.4.2_10, J2SE 5_06

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

注 - 在任一 Microsoft Windows 平台上运行 Application Server 时，必须使用 NTFS 文件系统，而不是 FAT 或 FAT32。

重要修补程序信息

有关 Sun Java System Application Server Platform Edition 8.2 当前所需的修补程序列表，请访问 <http://sunsolve.sun.com>，然后选择 "Patches" 或 "Patch Portal"。单击 Sun Java System Application Server Platform Edition 8.2 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。

Solaris 修补程序要求

建议 Solaris 8, 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 SunSolve (<http://sunsolve.sun.com/>) 网站上的 "Recommended and Security Patches" 下找到。

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 <http://rpm.pbone.net/index.php3/stat/4/idpl/843376/com/compat-libstdc++-7.3-2.96.118.i386.rpm.html> 下载该软件包。

JDBC 驱动程序和数据库

Sun Java System Application Server Platform Edition 支持使用相应的 JDBC 驱动程序连接至任何 DBMS。有关经 Sun 测试发现适合构建与 J2EE 兼容的数据库配置的组件列表，请参见下表。

表 2-2 与 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+, 10.1.x, 10.2.x 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+
Derby	类型 4	Apache Derby 10.1.2.1
PointBase	类型 4	PointBase Network Server 5.2
MySQL	类型 4	5.x
DataDirect	类型 4	Oracle (R) 8.1.7, 9i, 9.2.0.3+, 10.1.x, 10.2.x Sybase ASE 12.5.2 Microsoft SQL Server IBM DB2 8.1 Service Pack 3+
Oracle	类型 4、类型 2	Oracle (R) 9.2.0.3+, 10.1.x, 10.2.x

使用捆绑的 Derby 数据库

本节介绍了有关使用与 Application Server 8.2 捆绑在一起的 Derby 数据库实现的说明。

- 第 16 页中的“启动和停止 Derby 数据库”
- 第 17 页中的“Derby 实用程序脚本”
- 第 17 页中的“将 Pointbase 中的表导出到 Derby”

启动和停止 Derby 数据库

Sun Java System Application Server 8.2 引入了两个用于启动和停止 Derby 网络服务器的新 `asadmin` 命令。

- `start-database` 命令可用于启动 Derby 网络服务器实例：

```
start-database [--dbhost 0.0.0.0] [--dbport 1527] [--dbhome path/derby]
```

主机的默认值是 `0.0.0.0`，这将允许 Derby 侦听 `localhost` 以及 IP/主机名接口。dbhome 属性值用于表示 Derby 数据库的驻留位置。默认的 path 是 `<appserver_install_dir>/derby`。

- `asadmin stop-database` 命令用于关闭正在运行的 Derby 网络服务器实例：

```
stop-database [--dbhost 0.0.0.0] [--dbport 1527]
```


Derby 实用程序脚本

Application Server 8.2 附带的 Derby 配置还包括若干有用的脚本，这些脚本可有助于您使用 Derby。可以在 `<appserver_install_dir>/derby/frameworks/NetworkServer/bin` 目录中使用以下脚本：

- `startNetworkServer.ksh/bat`—用于启动网络服务器的脚本
- `stopNetworkServer.ksh/bat`—用于停止网络服务器的脚本
- `ij.ksh/bat`—交互式 JDBC 脚本工具
- `dblook.ksh/bat`—用于查看数据库的所有或部分 DDL 的脚本
- `sysinfo.ksh/bat`—用于显示有关 Derby 环境的版本化信息的脚本
- `NetworkServerControl.ksh/bat`—为在 `NetworkServerControl` API 上执行命令提供方法的脚本

▼ 配置环境以运行 Derby 实用程序脚本

- 1 将 `DERBY_INSTALL` 环境变量设置为指向 `<appserver_install_dir>/derby` 目录。
- 2 取消 `CLASSPATH` 环境变量的设置。
- 3 还可以有选择地设置以下属性：
 - a. 将 `DERBY_SERVER_HOST` 设置为网络服务器将侦听的主机。
还可以设置为 `0.0.0.0` 来启用所有侦听器。
 - b. 将 `DERBY_SERVER_PORT` 设置为网络服务器将侦听的端口号。

另请参见 有关这些实用程序的更多信息，请参见 Derby 工具 (<http://db.apache.org/derby/docs/10.1/tools/>) 和管理指南 (<http://db.apache.org/derby/docs/10.1/adminguide/>)。

将 Pointbase 中的表导出到 Derby

本示例介绍如何使用 Netbeans 5.0 捕获 Pointbase 中表的 DDL 并在 Derby 中创建相同的表。也可以使用 `commander` 工具和 `unload database` 命令来执行此操作：

```
./startcommander.sh
Do you wish to create a new Database. (Yes (Y) or No (N))? [default: N]:
Enter product to connect with: (Embedded (E) or Server (S))? [default: E]: e
Enter driver to use? [default: [com.pointbase.jdbc.jdbcUniversalDriver]:
Enter database URL? [default: [jdbc:pointbase:embedded:sample]:
Enter Username? [default: PBPUBLIC]:
Enter Password? [default: PBPUBLIC]:
```

PointBase Commander 5.2 ECF build 294 size restricted version EMBEDDED

Interactive SQL command language. SunOS/5.9

(C) Copyright 2004 DataMirror Mobile Solutions, Inc. All rights reserved.

Licensed to: Sun_customer_demo_use
 For commercial version contact PointBase at:
 pointbase.com
 PHONE: 1-877-238-8798 (US & CANADA)
 1-408-961-1100 (International)
 WEBSITE: www.pointbase.com

```
SQL>unload database sampledb.sql;
SQL> unload database sampledb.sql;
SQL> 13 Row(s) Unloaded. (PBPUBLIC.CUSTOMER_TBL)
SQL> 4 Row(s) Unloaded. (PBPUBLIC.DISCOUNT_CODE_TBL)
SQL> 30 Row(s) Unloaded. (PBPUBLIC.MANUFACTURE_TBL)
SQL> 11 Row(s) Unloaded. (PBPUBLIC.MICRO_MARKETS_TBL)
SQL> 9 Row(s) Unloaded. (PBPUBLIC.OFFICE_TBL)
SQL> 4 Row(s) Unloaded. (PBPUBLIC.OFFICE_TYPE_CODE_TBL)
SQL> 15 Row(s) Unloaded. (PBPUBLIC.ORDER_TBL)
SQL> 6 Row(s) Unloaded. (PBPUBLIC.PRODUCT_CODE_TBL)
SQL> 30 Row(s) Unloaded. (PBPUBLIC.PRODUCT_TBL)
SQL> 10 Row(s) Unloaded. (PBPUBLIC.SALES_REP_DATA_TBL)
SQL> 10 Row(s) Unloaded. (PBPUBLIC.SALES_REP_TBL)
SQL> 52 Row(s) Unloaded. (PBPUBLIC.SALES_TAX_CODE_TBL)
SQL> 12 Table(s) Unloaded.
SQL> quit;
```

在上述示例中，unload database 命令的执行结果将被写入 sampledb.sql 文件。sampledb.sql 文件中包含创建必要的表和索引所需的所有 DDL。它还包含将数据插回到数据库中的 DML。通过 commander 命令 RUN 使用生成的脚本可以将数据导入到其他 Pointbase 数据库中。以下是有关生成的文件中的 INSERT 语句和相关数据的示例：

```
INSERT INTO "ADVENTURE"."CATEGORY" (
"CATID", "LOCALE", "NAME", "DESCRIPTION", "IMAGEURI" )
VALUES( ?, ?, ?, ?, ? );
{
'ISLAND          ', 'en_US', 'Island Adventures', 'Experience an island /
paradise in a way fit for your needs.', 'Island_Adventures.gif'
'JUNGLE          ', 'en_US', 'Jungle Adventures', 'Experience a jungle /
paradise in a way fit for your needs.', 'Jungle_Adventures.gif'
'MOUNTAIN        ', 'en_US', 'Mountain Adventures', 'Experience an /
elevated paradise with a view.', 'Mountain_Adventures.gif'
'ORBITAL         ', 'en_US', 'Orbital Adventures', 'Experience a vacuum /
paradise with a beautiful view and where no one can hear you scream.', /
'Space_Adventures.gif'
'WESTERN         ', 'en_US', 'Western Adventures', 'Enjoy the Wild West. /
', 'Western_Adventures.gif'
```

```
'SOUTH_POLE          ', 'en_US', 'South Pole Adventures', 'Experience a /
frozen paradise in a way fit for your needs.', 'SouthPole_Adventures.gif'
};
```

可以轻松地编辑通过 `commander unload database` 命令生成的文件，以使其仅包含 DDL（例如，编写用于处理 `insert` 语句的程序并不困难）。在简单测试中，我们可以对 Pointbase 样例数据库使用 `unload database` 命令，然后编辑生成的脚本，可做出如下更改：

- 删除所有 `CREATE Table` 语句末尾的短语 `Organization Heap`
- 删除 `COMMIT` 命令
- 将布尔变量 `datatype` 更改为 `smallint`
- 删除所有 `INSERT` 语句和相关数据

接下来，使用简单的 Ant 脚本执行使用 `sql` 目标的 DDL。最后，对 `sun-appserv-samples` 数据库重复执行相同的操作，并对生成的 SQL 文件做出如下更改：

- 对样例数据库做出所有上述更改
- 删除 `create user` 命令
- 删除 `SET PATH` 命令
- 将 `Decimal` 精度从 38 更改为最大 31
- 将 `float` 精度从 64 更改为最大 52
- 当前不支持 `CREATE PROCEDURE` 的 `SPECIFIC` 关键字
- 删除 `GRANT` 命令

要将 Pointbase Java 过程转换为使用 Derby，需要对 Java 代码和 `CREATE PROCEDURE` 语句进行一些更改。有关创建 Derby Java 过程的信息，请参见 Derby 参考手册 (<http://db.apache.org/derby/docs/10.1/ref/>)。Derby 的下一发行版将支持布尔变量数据类型。

浏览器

本节列出了 Sun Java System Application Server Platform Edition 8.2 管理控制台和快速入门指南所支持的浏览器。在 Application Server 上运行的应用程序不同，支持的浏览器也不同。

表 2-3 支持的浏览器

浏览器	版本
Mozilla	1.4, 1.5, 1.6, 1.7.x
Netscape Navigator	6.2, 7.0
Internet Explorer	5.5 Service Pack 2, 6.0
Firefox	1.x

升级 Sun Java System Application Server

有关从 Application Server 的前先版本升级至 Sun Java System Application Server Platform Edition 8.2 的完整说明，请参阅《Sun Java System Application Server Platform Edition 8.2 Installation Guide》。

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。

- 空闲空间—临时目录必须具备至少 200 MB 的空闲空间以便安装 Sun Java System Application Server，以及 250 MB 的空闲空间以便安装 SDK。
- 使用卸载程序—如果需要从系统中删除 Application Server，请务必使用本软件附带的卸载程序。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。
- 空闲端口—必须有七个空闲端口。
 - 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 4848。
 - 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 3820 和 3890）。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)—除非要替换以前安装的服务器，否则应在开始安装 Sun Java System Application Server 8.2 之前启动以前安装的服务器。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。

- 替换以前安装的服务器 (UNIX)—如果已安装了旧版本的 Sun Java System Application Server，并希望将其替换为当前的 Application Server，则在安装新服务器之前应先停止旧服务器。
- 关闭防火墙 (Microsoft Windows)—在安装 Sun Java System Application Server 软件之前，必须停止所有防火墙软件，因为默认情况下某些防火墙软件会禁用所有端口。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《Sun Java System Application Server Platform Edition 8.2 Upgrade and Migration Guide》。

8.2 发行版中修复的错误

- | | |
|---------|---|
| 6184864 | 如果 where 子句包含 OR 运算符和单值型 cmr 导航，则 EJBQL 查询可能不会包含所有匹配的结果。 |
| 6198981 | “新建 Web 服务向导”中的面板不完整。 |
| 6207862 | - 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)。 |
| 6155080 | 无法通过 java-method 在 client-side message-security-binding 元素中指定目标消息。 |
| 6173248 | 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。 |

- 6360646 AS 8.2 PE/EE: 使用 PlatformMBeanServer (如果有)。
- 6295215 DOCS: java.sql.statement.getConnection() 不符合池连接的 javadoc 规范。
- 6290666 即使在停止并删除后, 域仍没有释放端口 8080。
- 6286688 首次创建 JMS 连接工厂时无法保存事务支持值。
- 6298257 对于 Application Server 8.1 UR2 PE, 根本无法在 Windows 2000 上运行 asant; 返回“命令太长”错误。
- 6320008 丰富客户机 RMI-IIOP 故障转移测试失败。
- 6347544 在 8.2 中支持 Grizzly ARA 包。
- 6275566 未更新 Application Server 8.1 虚拟服务器访问日志的位置。
- 6351023 JMS 的通用 RA 需要与 AS 8.2 集成。
- 6288752 XA 事务日志的磁盘 I/O 太高。
- 6356910 通过解除与控制 TTY 的关联, 使 Application Server 8.x 本地启动器可以使用 Bourne shell。
- 6307510 S1AS 7.0/SJAS7.1: EJBC/RMIC 生成不具有全限定包名称的 STUB/Skel。
- 6286783 服务器必须拒绝带有双 Content-Length 标头的请求。
- 6207862 asadmin create-domain --help 生成一些 CVS 合并字符, 且出现乱码。
- 6377830 当下一用户使用同一连接时, setAutoCommit 仍将设置为 false。
- 6317857 取消部署: 取消注册 mbean 时发生错误。
- 6284124 Servlet 容器 UTF-8 URI 映射问题。
- 6276218 如果在安装路径中使用空格, 则 Deploytool 将无法工作。
- 6211979 如果采用基于文件的非 root 用户安装, 则部署命令将失败。
- 6354545 在 Windows 中部署会挂起。
- 6283805 升级后无法访问已部署的应用程序。
- 6327037 需要改进部署性能。
- 6270387 重新部署有时会失败, 并显示错误消息“运行 EJBC 时发生错误 -- EJB 编译器发生致命错误”。
- 6258619 取消部署后, 没有释放所有文件。
- 6276021 WAR 文件的重新部署 (远程部署创建器) 失败。
- 6330332 在部署/取消部署方案中发生 AS8102 内存泄漏 (来自 6324399 EJBCClassLoader 的 SubCR)。

J2EE 支持

Sun Java System Application Server 8.2 支持 J2EE 1.4 和 Java EE 5 平台。下表介绍了 J2EE 1.4 平台上可用的增强 API。

表 2-4 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
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

表 2-4 J2EE 1.4 平台上的主要 API 更改 (续)

API	说明
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

切换到其他支持的 J2SE 版本

Sun Java System Application Server 8.2 支持将 J2SE 1.4.2 和 J2SE 5.0 作为基础 JVM。如果要从一个 J2SE 版本切换到另一个版本，请执行以下常规步骤。（Windows 和 Unix）

▼ 切换到另一个支持的 J2SE 版本

- 1 下载 J2SE SDK（而非 JRE）并将其安装到系统中（如果尚未这样做）。
可以从 <http://java.sun.com/j2se> 下载 J2SE SDK。
- 2 完全停止 Application Server。
您可以使用以下命令行：
`as-install/bin/asadmin stop-domain`
或者，也可以使用管理控制台 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` 的值更改为指向新的 J2SE 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新的 J2SE 主目录。
- 5 重新启动 Application Server。
`as-install/bin/asadmin start-domain`

接下来的操作 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 或更高版本。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。

已知问题和限制

本章介绍有关 Sun Java System Application Server Platform Edition 8.2 产品的已知问题和相应的解决方法。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：

- 第 25 页中的 “管理”
- 第 29 页中的 “应用程序客户机”
- 第 29 页中的 “数据库驱动程序”
- 第 29 页中的 “Deploytool”
- 第 30 页中的 “文档”
- 第 32 页中的 “安装”
- 第 33 页中的 “生命周期管理”
- 第 34 页中的 “日志记录”
- 第 34 页中的 “样例应用程序”
- 第 36 页中的 “安全性”
- 第 36 页中的 “升级实用程序”
- 第 38 页中的 “Web 容器”

管理

如果不存在 domain1，则 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，将不会使用新域名更新 `AS_ACC_CONFIG` 变量，从而导致 `package-appclient` 脚本失败。

解决方法

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 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` 命令对安装的另一 Application Server 上的域进行镜像。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时却会失败，原因是域配置中的条目并没有更改，并且 `startserv` 和 `stopserv` 仍然使用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 8.2 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)

可以在 Application Server 上配置 J2SE 1.4.x, 5.0 或更高版本。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。出现此问题的原因存在于内置 `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，它在启动 Application Server 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`，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: Here is the JMXServiceURL for the
JMXConnectorServer: [service:jmx:rmi:///jndi/rmi://hostname:8686/management/
rmi-jmx-connector]. This is where the remote administrative clients should
connect using the JSR 160 JMX Connectors.|#]
```

有关更多信息，请参阅《Sun Java System Application Server 8.2 Administration Guide》。

如果某个 Web 模块是任一虚拟服务器的默认 Web 模块，则无法对其进行重新部署或取消其部署。(ID 6204799)

如果将 Web 模块指定为虚拟服务器的默认 Web 模块，并试图对其进行重新部署或取消其部署，则将会收到以下错误消息：

```
Trying to undeploy application from domain failed; Virtual Servers [server]
have <WEB-MODULE-NAME> as default web module. Please remove the default web
module references first. ; requested operation cannot be completed Virtual
Servers [server] have <WEB-MODULE-NAME> as default web module. Please
remove the default web module references first.
```

此时，`domain.xml` 处于错误状态，管理控制台可能会无法显示已部署的 Web 应用程序表。即使停止并再次启动域，仍将出现这种情况。

解决方法

更改默认 Web 模块。

▼ 更改默认 Web 模块

- 1 使用管理控制台，转至虚拟服务器页面，将默认 Web 模块更改为空或指定其他 Web 模块。
- 2 使用 CLI，通过将 `domain` 指定为目标来取消 Web 模块的部署。

```
# asadmin undeploy --target domain <WEB-MODULE-NAME>
```

现在，管理控制台应该可以正常工作了，如果需要，可以再次部署 Web 模块。

通过 Application Server GUI 中的 AMX API 将 WAR 和 JAR 部署到 PE 服务器后，出现 FrameworkError 异常。(ID 6201462)

如果使用 AMX API 在 PE 上部署了某个应用程序但未对其进行引用，则 Application Server GUI 在显示该应用程序时将抛出错误。AMX 要求显式引用应用程序。例如，部署应用程序时，需要显式创建 `DeployedItemRefConfig`。如果为了简化部署过程而假定在 PE 中进行了引用，则将导致 Application Server GUI 出现问题。

解决方法

创建资源或应用程序后，始终创建对资源或应用程序的引用。

配置中的 Java 主目录设置不起作用。(ID 6240672)

Application Server 域/服务器未使用由相关配置的 `java-config` 元素的 `java-home` 属性指向的 JDK。

解决方法

对于安装的给定服务器，`appserver-installation-dir/config/asenv.conf` 文件确定了 Application Server 进程针对所有域使用的 JDK。该文件中的属性 `AS_JAVA` 确定了所使用的 JDK，此属性是在安装时设置的。如果安装完成后，Application Server 进程使用的是其他 JDK，则可以将该值改为指向所需的 JDK。请注意，安装的此服务器中的所有域均会受此更改影响。

注 - 不会检查对 `asenv.conf` 文件的手动更改的有效性，因此更改时应谨慎行事。修改 `AS_JAVA` 的值时，请检查产品文档以了解最低的 JDK 版本要求。

Selector.select() 抛出 IOException。Application Server 启动失败。(ID 6322825)

在当前 JDK 代码中，`/dev/poll` 选定器会将包含 8192 个 `pollfd` 条目的数组分配给该选定器使用。这将超出 `nofiles` `ulimit`，从而导致分配失败，并显示错误“参数无效”。`selector.select()` 的中断导致在启动期间连接至 MQ 的 Application Server 套接字服务失败，并抛出 `IOException`。

解决方法

放宽对 `pollfd` 文件描述符的限制。有两种方法可完成此操作：

1. 以 `root` 用户身份在 shell 上执行 `ulimit -n 8193`。
2. 将文件描述符数目的严格限制增加为 8193 或更高：
 - a. 使用 `ulimit -n -H` 检查严格限制。
 - b. 如果小于 8193，请编辑 `/etc/system`，以添加 `set rlim_fd_max=8193` 命令。
 - c. 重新引导计算机。

create-domain 主密码具有特殊字符时，域无法启动。(ID 6345947)

域的主密码包含百分比 (%) 字符时，域无法启动。

解决方法

域的主密码不应包含百分比字符 (%)。创建新域或更改现有域的主密码时亦如此。

AS 8.2 启动时无法正确处理特定的 Java System 属性。(ID 6372759)

将以下内容添加到 JVM 代理设置会导致服务器无法启动：

```
<jvm-options>-Dhttp.proxyHost=webcache.east.sun.com</jvm-options>
<jvm-options> -Dhttp.proxyPort=8080</jvm-options>
<jvm-options>-Dhttp.nonProxyHosts="mssp.ctu.gov|*.ctu.gov|localhost"
</jvm-options>
```

插入 * 字符会导致发生“找不到类 Def”错误（线程 `main java.lang.NoClassDefFoundError: com/sun/enterprise/security/store/IdentityManager` 中的异常）。插入 | 字符会导致启动脚本等待服务器启动超时。

此功能对于支持驻留在防火墙后且需要访问外部和内部服务器的 Application Server 部署（和门户部署）十分重要。Portal Server URL Scraper 便是一个例子。要允许 URL Scraper 从外部源获取内容，这些设置是必需的。

解决方法

编辑 `install-dir/config/asenv.conf` 文件，将行 `AS_NATIVE_LAUNCHER="true"` 更改为 `AS_NATIVE_LAUNCHER="false"`。

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 覆写了 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 文件将覆写客户机 JAR 的 MANIFEST 文件。

解决方法

目前尚无解决方法。

不支持诸如 CGI-bin 和 SHTML 功能之类的动态内容技术。(ID 6373043)

不再支持诸如 CGI-bin 和 SHTML 之类的动态内容技术。

解决方法

使用 JSP 和 Web 服务技术代替。

数据库驱动程序

本节介绍已知的数据库驱动程序问题和相应的解决方法。

使用 DB2 类型 II 驱动程序时，空闲超时后 DB2 服务器连接数不断增加。(ID 2082209/5022904)

当从其他应用服务器接入应用程序后，在连接超时后无法正确关闭物理连接。当对同一 DB2 7.1.x 数据库服务器使用 DB2 8.1.x 版的客户机库（类型 II）驱动程序时，会出现该问题。

解决方法

将 `SteadyPoolSize` 和 `MaxPoolSize` 设置为相同的数值，另外，还要将 `Idle Connection` 超时值设置为 0（零）。这会禁用空闲连接的超时，而且用户将拥有一组完整的可用连接。

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。指定了“目标名称”（例如，`PhysicalQueue`，一个用 `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` 中写入错误消息。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。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。

将出现以下问题：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() 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：

▼ 编辑现有 JNDI 名称

- 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：
- 5 重新启动 `deploytool`。
- 6 在“消息目标”选项卡上选择消息目标或添加新消息目标。
- 7 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。
- 8 通过单击“工具”\>“描述符查看器”\>“Application Server 描述符”来查看 Sun 描述符。
- 9 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。

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。

某些记录的监视功能不适用于 Platform Edition。(ID 6202255)

有关 AMX (Application Server Management eXtensions) 的文档部分中未指明某些监视功能在 Application Server Platform Edition 8.2 中不可用。具体来讲，在 Platform Edition 中不能监视的组件如下所示：

- 生产 **Web 容器 (Production Web Container, PWC)** :
 - PWC HTTP 服务
 - PWC 连接队列
 - PWC 线程池
 - PWC DNS
 - PWC 保持活动
 - PWC 文件高速缓存
 - PWC 虚拟服务器
 - PWC 请求

Web 模块

- SessionSize
- ContainerLatency
- SessionPersistTime
- CachedSessionsCurrent
- PassivatedSessionsCurrent

StatefulSessionStore

- CheckpointCount
- CheckpointSuccessCount
- CheckpointErrorCount
- CheckpointedBeanSize
- CheckpointTime

解决方法

无需解决方法。Platform Edition 不需要这些统计信息。

在文档中，AppservPasswordLoginModule 被引用为 AbstractPasswordLoginModule。(ID 6229682)

《Sun Java System Application Server Platform Edition 8.2 Developer's Guide》中的第 2 章“Securing Applications”中的《Sun Java System Application Server Platform Edition 8.2 Developer's Guide》中的“Realm Configuration”一节错误地引用了扩展 `com.sun.appserv.AbstractLoginModule`，但是此类现命名为 `com.sun.appserv.AppservLoginModule`。

解决方法

引用 `com.sun.appserv.AppservLoginModule` 而非 `com.sun.appserv.AbstractLoginModule`。

在 8.2 PE 手册页中，`--passwordfile` 不正确地具有了简短选项 `-W`。(ID 6373588)

`--passwordfile` 不应具有简短选项。当前，在该手册页中记录的是 `-W --passwordfile`。这是不正确的。

解决方法

在 Application Server 8.2 Platform Edition 中，请勿尝试将 `-W` 选项与 `--passwordfile` 一起使用。计划在以后的 Application Server 发行版中添加此简短选项。

多个 AMX 接口和方法的 Javadoc 缺少或不正确（若干 ID）：

- `ConnectorConnectionPoolStats` 和 `AltJDBCConnectionPoolStats` 中缺少用于获得 `NumConnAcquired` 和 `NumConnReleased` 统计信息的 Getter 方法。这些 Getter 方法将以 `getNumConnAcquired()` 和 `getNumConnReleased()` 的形式添加到以后的版本中。
- 在 `EJBCacheStats` 中调用以下方法时将抛出异常：`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。
- 常量 `XTypes.CONNECTOR_CONNECTION_POOL_MONITOR` 拼写错误 ("NNN")。在以后的版本中将纠正此问题。

安装

本节介绍已知的安装/卸载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。(ID 4977191)

在 Solaris x86 平台上有时会报告此问题，但此问题也可能会影响 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”按钮后安装结束过程挂起。(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 分发软件中也发现了此问题。

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/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。

在 Linux 上安装包装程序时，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：

```
Chcking available disk space....
Checking Java(TM) 2 Runtime Environment....
Extracting Java(TM) 2 Runtime Environment....
Deleting temporary files.....
```

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：

▼ 解决 Linux 上的引导问题

- 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`。

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-mills` 设置为 `7000`，将导致 `set` 命令失败，并显示以下错误消息：**(ID 6193449)**

```
[echo] Doing admin task set
[exec] [Attribute(id=redelivery-interval-internal-in-millis) : Redelivery-Interval (7,000) should be greater than or equal to Minimum-delivery-interval-in-millis (9,000)]
[exec] CLI137 Command set failed.
```

- `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` 失败后尝试重新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` 设置调试语句会导致 `Application Server` 在启动时挂起。(ID 6180095)

为 JVM 设置 `java.security.debug` 选项会导致服务器实例的启动冻结并死锁；例如，在 `domain.xml` 中进行以下设置将导致出现此问题：

```
<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。

样例应用程序

本节介绍与 `Application Server 8.2` 产品附带的样例代码相关的已知问题和相应的解决方法。

`managementws` 样例需要将 `MANIFEST.MF` 引用从 `castor-0.9.3.9-xml.jar` 更新为 `castor-0.9.9.1.jar`。(ID 6363339)

在 `<install_dir>/samples/webservices/jaxrpc/apps/managementws` 上运行检验器时，将会出现以下警告消息：

```
[exec] WARNING: /var/tmp/exploded20051214111425/managementws/ \
managementwsEjb_jar contains library/castor-0.9.3.9-xml.jar in Class-Path
manifest attribute, but it is not found in ear file
[exec] Dec 14, 2005 11:14:30 AM Archive getBundledArchives
[exec] WARNING: /var/tmp/exploded20051214111425/managementws/ \
managementwsEjb_jar contains library/castor-0.9.3.9-xml.jar in Class-Path
manifest attribute, but it is not found in ear file
```

已在 Application Server 8.2 发行版中更新了 Castor jar，因此对旧的 `castor-0.9.3.9-xml.jar` 的所有引用应更改为指向新的 `castor-0.9.9.1.jar`。具体来讲，需要将 MANIFEST.MF 文件中的引用更改为使用 `castor-0.9.9.1.jar` 而不是使用旧的 `castor-0.9.3.9-xml.jar`。

解决方法

将以下对旧的 Castor jar 的引用更改为指向新的 Castor jar：

旧的：

```
src/conf/MANIFEST.MF:Class-Path: library/castor-0.9.3.9-xml.jar
src/conf/MANIFEST.MF:Name: library/castor-0.9.3.9-xml.jar
managementws-ejb/src/conf/MANIFEST.MF:Class-Path: \
library/castor-0.9.3.9-xml.jar
```

新的：

```
src/conf/MANIFEST.MF:Class-Path: library/castor-0.9.9.1.jar
src/conf/MANIFEST.MF:Name: library/castor-0.9.9.1.jar
managementws-ejb/src/conf/MANIFEST.MF:Class-Path: \
library/castor-0.9.9.1.jar
```

然后，清除 `build.xml` 文件，这样就不会通过该文件在部署期间将 `Castor.jar` 复制到 `install_dir/lib`，以及在取消部署期间删除 `Castor.jar`。以下是新旧 `build.xml` 文件的不同之处。

```
% cvs diff build.xml Index: build.xml
=====
RCS file: /m/jws/samples/samples8x/webservices/jaxrpc/apps/managementws/ \
managementws-standalone-client/ Attic/build.xml,v retrieving revision \
1.1.2.3
diff -r1.1.2.3 build.xml
80,89d79
< <target name="remove_castor_from_classpath">
< <delete file="${com.sun.aas.installRoot}/lib/castor-0.9.9.1.jar"/>
< </target>
< <target name="add_castor_to_classpath">
< <delete file="${com.sun.aas.installRoot}/lib/castor-0.9.9.1.jar"/>
< <copy file="..lib/castor-0.9.9.1.jar" \
todir="${com.sun.aas.installRoot}/lib" />
< </target>
<
< <target name="setup" depends="add_castor_to_classpath, restart.server"/>
< jbenoit/galapago 196 >pwd
/net/galapago.east/files/share/8.2ws/samples/samples8x/webservices/jaxrpc \
/apps/managementws/managementws-standalone-client
jbenoit/galapago 197 >cd ..
jbenoit/galapago 198 >cvs diff build.xml
Index: build.xml
=====
```

```

RCS file: /m/jws/samples/samples8x/webservices/jaxrpc/apps/managementws/ \
Attic/build.xml
v retrieving revision 1.1.2.4
diff -r1.1.2.4 build.xml
28,36d27
< <target name="setup">
< <ant antfile="build.xml" inheritAll="true" dir="${sample.name}$ \
{standalone-client-dir-suffix}" target="setup"/>
< </target>
<
< <target name="unsetup">
< <ant antfile="build.xml" inheritAll="true" dir="${sample.name}$ \
{standalone-client-dir-suffix}" target="remove_castor_from_classpath"/>
< </target>
<
<
53,54c44,45
< <target name="deploy" depends="select_binary_common, deploy_common,
setup" />
< <target name="undeploy" depends="init, undeploy_common, unsetup"/>
---
> <target name="deploy" depends="select_binary_common, deploy_common" />
> <target name="undeploy" depends="init, undeploy_common"/>

```

安全性

本节介绍已知的安全性问题和相应的解决方法。

WS 安全性：应用程序客户机容器无法与 JAXRPC 客户机运行时正确集成。(ID 6325469)

应用程序客户机无法将用户名和密码传递给其他 Web 服务客户机。

解决方法

如果需要，将用户名/密码组合显式传递给客户机程序，如下所示：

```

((Stub)yourWSPort)._setProperty(Stub.USERNAME_PROPERTY, "yourUsername");
((Stub)yourWSPort)._setProperty(Stub.PASSWORD_PROPERTY, "yourPassword");

```

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 Platform Edition 8 升级到 Application Server Platform Edition 8.2 时，不会直接升级在自定义路径（而非 *install_dir/domains* 目录）中创建的域。(ID 6165528)

如果运行升级实用程序并将 *install_dir* 标识为源安装目录，则升级进程只升级在 *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，将所有域目录从其原有位置复制到 *install_dir/domains* 目录中。

从 **8.0 Platform Edition** 升级到 **8.2 Platform Edition** 后，启动 *domain1* 或 *samples* 域时发生端口冲突。(ID 6202188)

升级具有多个域的 8.0 Application Server 后，这些域可能无法同时启动，这是由于为 JMX 连接器配置了相同的端口号。

解决方法

更改端口值。

▼ 更改端口值

- 1 检查 *install_dir/domains/domain1/config/domain.xml* 文件中的以下条目：

```
<jmx-connector accept-all="false" address="0.0.0.0" auth-realm-name="admin-realm" enabled="true" name="system" port="8686" protocol="rmi_jrmp" security-enabled="false"/>" -- and in file <as 8.1 install_dir\>/domains/domain1/samples/config/domain.xml, notice it used the same port "8686", so it failed to start domain due to port conflict.
```

- 2 将端口值 8686 更改为 8687，然后重新启动 *domain1*。

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 最常见的问题，而且在 RedHat 分发软件中也发现了此问题。

在安装程序的最终屏幕上单击“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，并且无限期挂起，而不会返回命令提示符。

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。

▼ 使用命令行安装模式

- 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* 的值应与要升级的安装所使用的值匹配。

- 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/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。

升级后在“结果”面板中出现乱码 (ID 6376140)

当从多语言版本的 Application Server 8.2 升级到使用某些语言环境的更新版本时，“结果”面板可能出现乱码，并且 /opt/SUNWappserver/domains/upgrade.log 文件也可能含有乱码。

解决方法

目前尚无解决方法。此问题将在未来的 Application Server 发行版中得到修复。

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。（仅限于 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，因为会从 Application Server 中逻辑删除应用程序。另外请注意，asadmin 实用程序不会返回任何错误消息，但 application 的目录以及锁定的 jar 文件会保留在服务器中。server 的日志文件将包含用于说明未能删除文件和 application 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，则会出现这种情况，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 实用程序将返回与以下错误消息类似的错误消息。

```
An exception occurred while running the command. The exception message
is: CLI171 Command deploy failed : Deploying application in domain failed;
Cannot deploy. Module directory is locked and can\qt be deleted
```

解决方法

如果部署应用程序时指定了 `--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 中可选的 `load-on-startup` servlet 元素表示在启动声明相关 servlet 的 Web 应用程序时, 将加载及初始化此 servlet。

此元素的可选内容是一个整数, 用于表示该 servlet 相对于 Web application 的其他 servlet 而被装入和初始化的顺序。只要该 servlet 在包含它的 Web 应用程序启动期间被加载和初始化, 空的 `<load-on-startup>` 就表示顺序无关紧要。

web.xml 的 Servlet 2.4 模式不再支持空的 `<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/>` 中一样指定空的 `<load-on-startup>`, 则 web.xml 将无法针对 web.xml 的 Servlet 2.4 模式进行验证, 从而导致 Web 应用程序的部署失败。

向下兼容性问题。在基于 Servlet 2.3 的 web.xml 中仍可以像往常那样指定空的 `<load-on-startup>`。

解决方法

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

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”和以下堆栈跟踪消息:

```
at org.apache.tools.ant.taskdefs.Execute$Java13CommandLauncher.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：

- 在全局范围内，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>
```

- 对于每个 Web 应用程序，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。

在多 CPU 计算机上运行时性能降低。(ID 6194026)

Application Server PE 的默认配置在多 CPU 计算机上无法表现出最佳性能。可以交替使用以使启动更快，但这又会对 Web 应用程序的性能产生负面影响。

解决方法

配置 Application Server 以使用以下 JVM 选项：

```
-Dcom.sun.enterprise.server.ss.ASQuickStartup=false
```

如果收到的 Fast Infoset 文档不规则，则会禁用 Fast Infoset 对已部署的 JAX-RPC 服务的支持。(ID 6368670)

如果将不符合 Fast Infoset 编码的 SOAP 消息发送给 JAX-RPC 服务，则服务会发生响应故障，这是正常的。但将后续的符合 Fast Infoset 编码的 SOAP 消息发送到同一服务或使用同一 JAX-RPC 运行时部署的服务时，则会不正常地发生故障。

解决方法

可以使用以下解决方法：

- 禁用客户机上的 Fast Infoset 支持，以便仅发送 XML 编码的 SOAP 消息。
- 重新启动部署服务的容器，以便可以发送符合 Fast Infoset 编码的 SOAP 消息。