



Sun Java System Application Server 9.1 发行说明



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

文件号码 820-3517
2007 年 9 月

版权所有 2007 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 SunTM 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 和在其他方面遵守 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。

概述

Sun Java™ System Application Server 9.1 产品显著简化了创建和管理 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。

- 第 3 页中的“关于本说明”
- 第 4 页中的“发行说明修订历史记录”
- 第 4 页中的“为残疾人士提供的辅助功能”
- 第 4 页中的“相关文档”
- 第 5 页中的“如何报告问题和提供反馈”
- 第 6 页中的“Sun 欢迎您提出意见”
- 第 6 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 9.1 发行时可用的重要信息。还介绍了增强功能、已知问题和其他最新问题。开始使用 Application Server 9.1 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/1343.4?q=glassfish>)上找到本发行说明的最新版。在安装和配置软件之前，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

本节列出了在 Sun Java System Application Server 9.1 产品的初始发行版之后，在本发行说明中所做的更改。

表 1-1 发行说明修订历史记录

修订日期	说明
2007 年 2 月	Sun Java System Application Server 9.1 产品的初始版。
2007 年 3 月	Sun Java System Application Server 9.1 产品的 Beta 1 发行版。
2007 年 5 月	Sun Java System Application Server 9.1 产品的 Beta 2 发行版。
2007 年 9 月	Sun Java System Application Server 9.1 产品的 FCS 发行版。

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

表 1-2 Application Server 文档集中的书籍

书名	说明
发行说明	软件和文档的最新信息。包括有关支持的硬件、操作系统、JDK 和 JDBC/RDBMS 的综合表式摘要。
快速入门指南	如何开始使用 Application Server 产品。
Installation Guide	安装软件及其组件。

表 1-2 Application Server 文档集中的书籍 (续)

书名	说明
部署规划指南	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 Application Server。此外还介绍了部署服务器时应该注意的常见问题。
开发者指南	创建和实现要在 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。
J2EE 1.4 Tutorial	使用 J2EE 1.4 平台技术和 API 开发 J2EE 应用程序。
管理指南	从管理控制台配置、管理和部署 Application Server 子系统和组件。
High Availability Administration Guide	有关高可用性数据库的安装后配置和管理说明。
Administration Reference	编辑 Application Server 配置文件 domain.xml。
Upgrade and Migration Guide	将应用程序迁移到新的 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范不兼容的相邻产品发行版和配置选项之间的差异。
Performance Tuning Guide	调优 Application Server 以提高性能。
Troubleshooting Guide	解决 Application Server 问题。
Error Message Reference	解析 Application Server 错误消息。
Reference Manual	可用于 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。

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** (<http://java.sun.com/docs/forms/J2EE14SubmittalForm.html>)—用于提交有关 Application Server 产品反馈的表单
- **Glassfish 邮件列表** (<https://glassfish.dev.java.net/servlets/ProjectMailingListList>)—针对各种问题和反馈的多个 Glassfish 社区邮件列表
- **Java Developer Connection 的错误数据库** (<http://developer.java.sun.com/servlet/SessionServlet?url=/developer/bugParade/index.jshtml>)—要查看错误或提交错误，请使用 Java Developer Connection 的 Bug Parade (错误展示)
- **Java 技术论坛** (<http://forum.java.sun.com/>)—用于共享有关 Java 技术和编程技术的知识和讨论问题的交互式消息板；使用 Java EE 5 SDK 论坛可讨论有关 Sun Java System Application Server 9.1 产品的问题

- Sun 软件支持服务 (<http://www.sun.com/service/sunone/software>)—可以链接至知识库、联机支持中心和 Product Tracker，并取得维护程序和支持联系人电话号码
- 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，本书的标题是《Sun Java System Application Server 9.1 发行说明》，文件号码是 820-3517。

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)
- 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 9.1

Sun Java System Application Server 9.1 是 Java EE 5 平台兼容的服务器，用于在大规模生产环境中开发和部署 Java EE 应用程序和基于 Java 技术的 Web 服务。

本章包括：

- 第 7 页中的 “9.1 发行版的新增功能”
- 第 9 页中的 “硬件和软件要求”
- 第 18 页中的 “Java EE 5 平台 API”
- 第 19 页中的 “Java EE 5 SDK”
- 第 20 页中的 “切换到支持的另一 Java 版本”
- 第 20 页中的 “已知的 Java EE 5 兼容性问题”
- 第 21 页中的 “使用 Application Server 更新中心”
- 第 21 页中的 “有关 WSIT 集成的更多信息”

9.1 发行版的新增功能

Application Server 9.1 包括以下增强功能：

- **Java EE 5 平台支持**—Sun Java System Application Server Enterprise Edition 9.1 实现了 Java EE 5 规范，可以为下一代企业应用程序和 Web 服务提供最佳的应用程序运行时。Application Server 9.1 实现了以下 Java EE 标准：
 - Enterprise Java Beans 3.0
 - JAXB 2.0
 - Java Persistence
 - Java Server Faces 1.2
 - Java Server Pages 2.1 (JSP 2.1)
 - Java Server Pages Standard Tag Library (JSTL) 1.2
 - Streaming API for XML (StAX)
 - Web Services Metadata
 - Java API for XML based Web Services 2.0 (JAX-WS 2.0)
 - Common Annotations for the Java Platform 1.0 (CAJ 1.0)
 - Java Servlet 2.5

本说明的后面部分提供了完整的 Java EE 5 平台技术列表。

- **Web 服务互操作性技术 (Web Services Interoperability Technologies, WSIT) 支持** – Sun 与 Microsoft 密切合作，以确保 Web 服务企业技术（例如消息优化、可靠的消息传送和安全性）的互操作性。WSIT 的初始发行版便是这一通力合作的产品。WSIT 是多个开放式 Web 服务规范的实现，用以支持企业功能。除了消息优化、可靠的消息传送和安全性之外，WSIT 还包括引导和配置技术。从当前 Java 平台上内置的核心 XML 支持开始，WSIT 使用或扩展了现有的功能，并添加了新的对可互操作 Web 服务的支持，其中包括：
 - 引导和配置
 - 消息优化技术
 - 可靠的消息传送技术
 - 安全性技术

有关 Application Server 9.1 中的 WSIT 集成的更多信息，请参见本章后面部分的第 21 页中的“有关 WSIT 集成的更多信息”。

- **Java 业务集成 (Java Business Integration, JBI) 支持** – JBI 使用业务集成服务提供者接口 (Service Provider Interface, SPI) 对 Java EE 进行了扩展。开发者可以使用这些 SPI 为 WSCI、BPEL4WS 和 W3C Choreography Working Group 等规范创建或实现 Java 业务集成环境。JBI 实现由 Application Server 安装程序直接安装，通常安装在 AS_HOME/jbi 目录中。此目录包含 JBI 的所有通用 JAR 文件和系统组件，其中包括在 Application Server JVM 中启动 JBI 框架的 lifecycle 模块。
- **内存复制支持** – 其他服务器上的内存复制提供了会话状态数据的轻量存储，而无需获得单独的数据库（例如 HADB）。此类型的复制使用其他服务器上的内存对 HTTP 会话和有状态会话 Bean 数据进行高可用性存储。群集服务器实例在环状拓扑中复制会话状态。每个备份实例都将复制的数据存储在内存中。通过在其他服务器上的内存中复制会话状态数据，可以对会话进行分配。使用内存复制需要启用组管理服务 (GroupManagement Service, GMS)（启用状态默认为 true）。
- **使用情况配置文件** – 每个管理域都与一个使用情况配置文件关联，此配置文件指定该域的功能。Application Server 9.1 提供以下配置文件：
 - **开发者** – 如果正在开发环境中运行域，并且应用程序不需要群集功能（例如负载均衡、高可用性和会话复制），请使用此配置文件。请注意，此配置文件的实际名称为 "developer"（区分大小写）。
 - **群集** – 如果要创建向已部署的 Java EE 应用程序分配可伸缩性和高可用性的应用服务器实例的群集，请使用此配置文件。应用程序的状态持久保存在内存中。请注意，此配置文件的实际名称为 "cluster"（区分大小写）。
 - **企业** – 如果需要 HADB 和 NSS，请使用此配置文件。除非分别安装了 HADB 和 NSS，或者将 Application Server 作为 Java Enterprise System (Java ES) 的一部分安装，否则此配置文件不可用。
- **负载均衡增强功能** – 已向负载均衡插件中添加了数项增强功能。简而言之，这些增强功能包括：

- **加权循环** – 已向实例 `Loadbalancer.xml` 文件中添加了名为 `weight` 的可选属性。使用此选项，负载均衡器插件可以根据权重路由请求。例如，对于每 500 个请求，100 个将转至 `instance1`，400 个将转至 `instance2`。默认权重为 100。可从管理控制台或命令行为每个实例指定权重，服务器 `domain.xml` 针对每个实例都有一个表示权重的属性。
- **用户自定义的负载均衡器决策** – 可让用户定义用于负载均衡的自定义逻辑；例如，基于用户身份的重定向和基于 `mime` 的负载均衡。此功能通过由负载均衡器装入的用户自定义共享库实现。此自定义共享库可实现 `loadbalancer.h`（位于 `appserver_install_dir/lib/install/templates/` 中）中定义的接口。
- **管理增强功能** – 早期版本的 Application Server 需要将 `loadbalancer.xml` 文件手动复制到特定服务器的 `config` 目录。Application Server 9.1 包括用于在 Web 服务器和 Application Server 之间执行此类复制的推送式自动操作。负载均衡器本身在服务器的 `domain.xml` 文件中进行配置。
- **开放源代码和 GlassFish 社区** – Sun 在 2005 年 6 月启动了 [GlassFish](http://java.sun.com/javaee/glassfish/) (<http://java.sun.com/javaee/glassfish/>) 社区，目的是开发可实现 Java EE 5 平台的最新功能和相关企业技术的免费开放源代码的商业级应用服务器。Application Server Enterprise Edition 9.1 就是基于由 Sun 的工程师和 GlassFish 社区开发的源代码。
- **更新中心支持** – Application Server 9.1 更新中心提供了自动 Application Server 更新，允许用户轻松访问其他组件。有关更多信息，请参见第 21 页中的“使用 Application Server 更新中心”。

硬件和软件要求

本节列出了在安装 Sun Java System Application Server 9.1 产品之前必须满足的要求。

- 第 9 页中的“支持的平台”
- 第 10 页中的“重要修补程序信息”
- 第 11 页中的“JDK 版本”
- 第 11 页中的“JDBC 驱动程序和数据库”
- 第 12 页中的“使用捆绑的 Java DB 数据库”
- 第 15 页中的“Web 服务器”
- 第 15 页中的“浏览器”
- 第 16 页中的“HADB 要求和支持的平台”
- 第 17 页中的“升级 Sun Java System Application Server”
- 第 17 页中的“其他要求”

支持的平台

下表列出了 Sun Java System Application Server Enterprise Edition 9.1 所兼容的操作系统。

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 5.0 Java SE 6
64 位 Sun Solaris 10 (SPARC、x86)	512 MB	512 MB	250 MB 可用空间	500 MB 可用空间	J2SE 5.0 Java SE 6
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 5.0 Java SE 6
SuSE Linux Enterprise Server 10 SP1	512 MB	1 GB	250 MB 可用空间	500 MB 可用空间	J2SE 5.0 Java SE 6
Windows Server 2000 SP4+ Windows 2000 Advanced Server SP4+ Windows Server 2003 Windows XP Pro SP1+ Windows Vista	1 GB	2 GB	500 MB 可用空间	1 GB 可用空间	J2SE 5.0 Java SE 6

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。

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

重要修补程序信息

Solaris 修补程序要求

建议 Solaris 9、10 (x86、SPARC) 用户安装“Sun 推荐的修补程序簇”。此修补程序簇在 SunSolve (<http://sunsolve.sun.com/pub-cgi/show.pl?target=patchpage>) Web 站点上的 "Recommended and Security Patches" 下提供。

JDK 版本

Application Server 9.1 需要的最低（并且经过认证）JDK 版本是 **1.5.0_12**。

JDBC 驱动程序和数据库

表 2-2 列出了满足 Java EE 兼容性要求的数据库和驱动程序。所有受支持的 Sun Java System Application Server Enterprise Edition 配置必须至少包含一个来自此表的数据库/驱动程序组合，例如捆绑的 Java DB 数据库和驱动程序。此外，Application Server 支持使用相应的 JDBC 驱动程序与任何其他 DBMS 进行 JDBC 连接。

表 2-2 兼容 Java EE 的 JDBC 驱动程序

JDBC 驱动程序供应商	JDBC 驱动程序类型	支持的数据库服务器
Derby Network Client	类型 4	Derby 10.2
DataDirect 3.60.29	类型 4	Oracle 10g
DataDirect 3.6	类型 4	DB2 9.1 DB2 8.2
Oracle Thin	类型 4	Oracle 10g
PostGres	类型 4	8.1
MySQL Connector/J Driver 3.1	类型 4	MySQL 5

表 2-3 列出了预定在 Application Server FCS 发行版中受支持的所有其他数据库和驱动程序。

表 2-3 支持的 JDBC 驱动程序和数据库 (Application Server FCS)

JDBC 驱动程序供应商	JDBC 驱动程序类型	支持的数据库服务器
Oracle OCI	类型 2	Oracle 9i
IBM DB2	类型 2	DB2 9.1
Oracle 10G	类型 4	-RAC
DataDirect 3.60.29	类型 4	Sybase ASE 15 MS SQL 2005
MySQL Connector/J Driver 3.1	类型 4	MySQL 5

一般情况下，Application Server Enterprise Edition 9.1 支持符合 Java EE 规范的所有 JDBC 驱动程序。

使用捆绑的 Java DB 数据库

本节介绍了有关使用 Application Server 9.1 捆绑的 Java DB 数据库实现的说明。Java DB 基于 [Apache Derby 数据库](#)。

- 第 12 页中的 “启动和停止 Java DB 数据库”
- 第 12 页中的 “Java DB 实用程序脚本”
- 第 13 页中的 “将 Pointbase 中的表导出到 Java DB”

启动和停止 Java DB 数据库

Sun Java System Application Server 9.1 引入了两个新的 `asadmin` 命令，用于启动和停止 Java DB 网络服务器。

- `asadmin start-database` 命令用于启动 Java DB 网络服务器实例：

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

主机的默认值为 `0.0.0.0`，这将允许 Java DB 在 `localhost` 以及 IP/主机名接口上侦听。`dbhome` 属性值为 Java DB 数据库的驻留位置。默认 `path` 为 `appserver_install_dir/javadb`。

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

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

Java DB 实用程序脚本

注 – 这些脚本将在 Application Server 9.1 中停止使用。

Application Server 9.1 附带的 Java DB 配置还包括几个有用的脚本，可帮助您使用 Java DB。以下脚本可用于 `<appserver_install_dir>/javadb/frameworks/NetworkServer/bin` 目录中：

- `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`—显示有关 Java DB 环境的版本信息的脚本
- `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 Java DB 实用程序脚本

- 1 将 DERBY_INSTALL 环境变量设置为指向 <appserver_install_dir>/javadb 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/) (<http://db.apache.org/derby/docs/10.1/tools/>) 和 [管理指南](http://db.apache.org/derby/docs/10.1/adminguide/) (<http://db.apache.org/derby/docs/10.1/adminguide/>)。

将 Pointbase 中的表导出到 Java DB

本示例介绍如何使用 NetBeans 5.0 捕获 commander 中 Pointbase 表的 DDL 以及在 Java DB 中创建相同的表。也可以通过使用 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。sampleddb.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，以使它仅包含 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 Java DB，需要对 Java 代码和 CREATE PROCEDURE 语句进行某些更改。可在 [Derby 参考手册](http://db.apache.org/derby/docs/10.1/ref/) (<http://db.apache.org/derby/docs/10.1/ref/>) 中查看有关创建 Java DB Java 过程的信息。Java DB 的下一版将支持布尔变量数据类型。

Web 服务器

本节列出了 Sun Java System Application Server 9.1 支持的 Web 服务器。

表 2-4 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 9.1 支持的浏览器。

表 2-5 支持的 Web 浏览器

浏览器	版本
Mozilla	1.4、1.5、1.6、1.7.x、2.x
Internet Explorer	5.5 Service Pack 2、6.0、7.0
Firefox	1.5.x、2.x

HADB 要求和支持的平台

除了第 9 页中的“硬件和软件要求”中列出的要求以外，还需验证系统是否满足下面所列的运行 HADB 的要求。

- 第 16 页中的“支持的平台”
- 第 16 页中的“HADB 服务器主机要求”
- 第 17 页中的“HADB 管理主机要求”
- 第 17 页中的“HADB 客户机主机要求”

注 - HADB 仅捆绑在 Application Server 9.1 Enterprise 配置文件中。可以从 *** 下载 HADB。还请注意，系统的 Java 组件已使用 JDK 5 生成，并已在 JDK 1.5 上进行了测试。

支持的平台

- **Solaris (SPARC)**—Solaris 8 MU7、Solaris 9 MU7、Solaris 10 RR。
- **Solaris (x86)**—Solaris 9 MU7、Solaris 10 RR。
- **RedHat Enterprise Linux**—2.1 U5（仅支持 ext2 文件系统，不支持 ext3）和 3.0 U4（ext2 和 ext3 均受支持。由于会出现过度交换的问题，建议不要使用 U4 之前的更新版本）。请注意，仅在 32 位模式下的操作系统中测试了 HADB。还请注意，HADB 不支持以 64 位模式运行的 RedHat Enterprise Linux 3.0，这是由于该操作系统中的一个错误所致（有关对 HADB 的影响的详细信息，请参见第 32 页中的“高可用性”部分中的已知错误 6249685）。
- **Microsoft Windows**—Microsoft Windows 2000 Advanced Server Service Pack 4 和 Microsoft Windows 2003 Enterprise Edition。请注意，HADB 不支持任何即将推出的 64 位模式的 Microsoft Windows 操作系统版本。

HADB 服务器主机要求

- **最小内存**—每个节点需要 320 MB。
- **最小可用磁盘空间**—每个节点的 HADB 二进制需要 70 MB。此外，对于每个节点上的测试安装，数据设备还需要 512 MB 的磁盘空间。
- **建议的内存**—每个节点需要 512 MB。
- **建议的可用磁盘空间**—每台主机的 HADB 二进制需要 70 MB。此外，对于每个节点上的测试安装，数据设备还需要 1200 MB 的磁盘空间。

注 – 请确保在存储 HADB 数据和日志文件的设备上禁用写高速缓存。默认情况下，在有些 Solaris 平台上写高速缓存处于启用状态；例如，Solaris x86。

HADB 管理主机要求

- 最小内存—128 MB
- 最小可用磁盘空间 — 每台主机的 HADB 二进制需要 70 MB。

HADB 客户机主机要求

- 最小内存—120 MB
- 最小可用磁盘空间—20 MB

升级 Sun Java System Application Server

有关从 Application Server 的早期版本升级到当前版本的完整说明，请参阅《Sun Java System Application Server 9.1 Upgrade and Migration 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。

- **可用空间：**临时目录必须具有至少 35MB 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：**如果需要从系统中删除 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：**必须有七个未使用的可用端口。安装程序将自动检测使用中的端口，并为默认域推荐当前未使用的端口。默认情况下，初始默认端口如下：
 - 8080（用于 HTTP）；8181（用于 HTTPS）
 - 3700（用于 IIOP）；3820（用于 IIOP/SSL）；3920（用于具有相互验证的 IIOP/SSL）
 - 4848 (HTTP)（用于 Administration Server）
 - 8686 (RMI) 端口（用于纯 JMX 客户机）

启动以前安装的服务器 (UNIX)—除非要替换以前安装的服务器，否则在开始安装 Sun Java System Application Server 9.1 之前必须先启动以前安装的服务器。这样，安装程序才能检测到正在使用的端口，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)** — 有关从 Application Server 的早期版本升级的完整说明，请参阅《Sun Java System Application Server 9.1 Upgrade and Migration Guide》。
- **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。

有关进一步的兼容性信息，请参见《Sun Java System Application Server 9.1 Upgrade and Migration Guide》。

Java EE 5 平台 API

Sun Java System Application Server Enterprise Edition 9.1 支持 Java EE 5 平台。下表列出了 Java EE 5 平台上可用的增强 API。

表 2-6 Java EE 5 平台上的主要 API 更改

API	JSR
Java EE 5	
Java Platform, Enterprise Edition 5 (http://java.sun.com/javaee/5/docs/api/)	JSR 244 (http://jcp.org/aboutJava/communityprocess/pr/jsr244/)
Web 服务技术	
Implementing Enterprise Web Services	JSR 109 (http://jcp.org/en/jsr/detail?id=109)
Java API for XML-Based Web Services (JAX-WS) 2.0 (https://jax-ws.dev.java.net/)	JSR 224 (http://jcp.org/en/jsr/detail?id=224)
Java API for XML-Based RPC (JAX-RPC) 1.1 (https://jax-rpc.dev.java.net/)	JSR 101 (http://jcp.org/en/jsr/detail?id=101)
Java Architecture for XML Binding (JAXB) 2.0 (https://jaxb.dev.java.net/)	JSR 222 (http://jcp.org/en/jsr/detail?id=222)
SOAP with Attachments API for Java (SAAJ) (https://saaj.dev.java.net/)	JSR 67 (http://jcp.org/en/jsr/detail?id=67)
Streaming API for XML (http://java.sun.com/webservices/docs/1.6/tutorial/doc/SJSXP.html)	JSR 173 (http://jcp.org/en/jsr/detail?id=173)
Web Service Metadata for the Java Platform	JSR 181 (http://jcp.org/en/jsr/detail?id=181)
组件模型技术	
Enterprise JavaBeans 3.0 (http://java.sun.com/products/ejb/)	JSR 220 (http://jcp.org/en/jsr/detail?id=220)
J2EE Connector Architecture 1.5 (http://java.sun.com/j2ee/connector/)	JSR 112 (http://jcp.org/en/jsr/detail?id=112)
Java Servlet 2.5 (http://java.sun.com/products/servlet/)	JSR 154 (http://jcp.org/en/jsr/detail?id=154)
JavaServer Faces 1.2 (http://java.sun.com/j2ee/javaserverfaces/)	JSR 252 (http://jcp.org/en/jsr/detail?id=252)

表 2-6 Java EE 5 平台上的主要 API 更改 (续)

API	JSR
JavaServer Pages 2.1 (http://java.sun.com/products/jsp/)	JSR 245 (http://jcp.org/en/jsr/detail?id=245)
JavaServer Pages Standard Tag Library 1.2 (http://java.sun.com/products/jsp/jstl/)	JSR 52 (http://jcp.org/en/jsr/detail?id=52)
管理技术	
J2EE Management (http://java.sun.com/j2ee/tools/management/)	JSR 77 (http://jcp.org/en/jsr/detail?id=77)
J2EE Application Deployment (http://java.sun.com/j2ee/tools/deployment/)	JSR 88 (http://jcp.org/en/jsr/detail?id=88)
Java Authorization Contract for Containers (http://java.sun.com/j2ee/javaacc/)	JSR 115 (http://jcp.org/en/jsr/detail?id=115)
其他 Java EE 技术	
Common Annotations for the Java Platform	JSR 250 (http://jcp.org/en/jsr/detail?id=250)
Java Transaction API (JTA) (http://java.sun.com/products/jta/)	JSR 907 (http://jcp.org/en/jsr/detail?id=907)
JavaBeans Activation Framework (JAF) 1.1 (http://java.sun.com/products/javabeans/glasgow/jaf.html)	JSR 925 (http://jcp.org/en/jsr/detail?id=925)
JavaMail (http://java.sun.com/products/javamail/)	JSR 919 (http://jcp.org/en/jsr/detail?id=919)
Java Message Service API (http://java.sun.com/products/jms/)	JSR 914 (http://www.jcp.org/en/jsr/detail?id=914)
Java Persistence API (http://java.sun.com/j2ee/persistence/faq.html)	JSR 220 (http://www.jcp.org/en/jsr/detail?id=220)

Java EE 5 SDK

Sun Java System Application Server 9.1 是作为 Java EE 5 SDK 的一部分提供的。

有两个 Java EE 5 SDK 版本：

- Java Application Platform SDK (<http://java.sun.com/javaee/downloads/>)
- Java EE 5 SDK (<http://java.sun.com/javaee/downloads/index.jsp>)

此外，可以下载带有 JDK 或 NetBeans 5.5.1 工具的 SDK 分发版。有关更多信息，请访问下载页，网址为：<http://java.sun.com/javaee/downloads/index.jsp>。

注 - GlassFish V2 和 Application Server 9.1 不能使用 NetBeans 5.5 来开发 Java EE 应用程序。必须使用 NetBeans 5.5.1。

切换到支持的另一 Java 版本

Sun Java System Application Server 9.1 要求使用 J2SE 5.0 或更高版本作为基础 JVM。如果要从一个 Java 版本切换到另一版本，请执行以下常规步骤。（Windows 和 Unix）

注 - 使用较新的 Java VM 创建了域之后，建议不要降级到较早的 Java 版本。如果必须降级 JVM，建议在每个域上都执行此操作。以下过程说明了如何执行此操作。

▼ 切换到支持的另一 Java 版本

- 1 下载所需的 Java SDK（而非 JRE）并将其安装在系统上（如果尚未这样做）。
可以从 <http://java.sun.com/j2se> 下载 Java SDK。

- 2 启动要更改其 JDK 的域：

```
as_install/bin/asadmin start-domain domain_name
```

- 3 登录到管理控制台并更改域的 JVM 属性。

特别是，可能需要在“JVM 设置”页上更改域的 JAVA_HOME 变量。

或者，可以使用 asadmin 命令：

```
as_install/bin/asadmin set "server.java-config.java-home=Path-To-Java-Home"
```

已知的 Java ES 5 兼容性问题

在 Application Server 9.1 和 Java ES 5 (JES5) 之间，存在两个已知的兼容性问题。

1. 由于 Application Server 9.1 提供了 JSF 1.2 版本，因此 JES5 Service Registry 与 Application Server 9.1 不兼容。在 Application Server 升级到 9.1 之前，需要将 Service Registry 升级到 JES5u1。此限制在 Service Registry 文档中也有所记录。
2. 由于 Application Server 9.1 提供了 JSF 1.2 版本，因此 JES5 Portal Server 与 Application Server 9.1 不兼容。在 Application Server 升级到 9.1 之前，需要将 Portal Server 升级到 JES5u1。

在 Portal Server 位于 Application Server 8.2 上的 Java ES 5 设置中，将 Application Server 8.2 升级到 9.1 会使 Portal Server 不可用。Application Server 9.1 使用 JSF 1.2，但是 Portal Server 版本 5 中的 JSF-Portlet 桥不支持 JSF 1.2。在 Solaris/Linux 上，需要将 Portal Server 升级到 JavaES5 Update 1。在 Windows 上，如果要继续使用 Portal Server 版本 5，请勿将 Application Server 升级到 9.1，因为在 Windows 上，Java ES 5 Update 1 不支持 Portal Server。

使用 Application Server 更新中心

Application Server 9.1 更新中心提供了自动 Application Server 更新，允许用户轻松访问其他组件。

更新中心启用后，便会执行自动软件更新。在此自动更新过程中，更新中心将收集以下数据并将其传送给 Sun Microsystems（或其服务提供者）：

- 唯一安装 ID (GUID)
- IP 地址
- 操作系统信息（名称、版本、体系结构、语言环境）
- JDK 版本
- 模块下载信息（模块名称、日期、时间、状态、下载时间、已下载的字节数）

将不会跟踪个人识别信息。也不会将个人识别信息与其他任何数据关联，或用于生成报告。

▼ 使用更新中心

为确保明确同意使用自动更新，默认情况下禁用更新中心。要启用更新中心以执行定期检查 and 自动更新，请执行以下操作：

- 1 启动更新中心。
 - 在 Unix 上：<installdir>/updatecenter/bin/updatetool
 - 在 Windows 上：<installdir>\updatecenter\bin\updatetool.bat
- 2 选择“首选项”选项卡。
- 3 在“更新计划”窗口中，将“检查更新”下拉框的值从“从不（手动）”更改为所需的值。例如，每天或每周。
- 4 指定需要在每周的哪天以及每天的什么时间进行更新。
- 5 选择“保存”按钮以保存更改。

更新中心现在将根据指定的计划，自动检查 Glassfish 或 Application Server 组件的更新。当更新可用时，更新中心将启动，并通知您可进行更新的组件。

有关 WSIT 集成的更多信息

有关 WSIT 状态的详细信息，请参阅 [WSIT Status Notes \(https://wsit.dev.java.net/source/browse/*checkout*/wsit/wsit/status-notes-1-0-FCS.html\)](https://wsit.dev.java.net/source/browse/*checkout*/wsit/wsit/status-notes-1-0-FCS.html) 页。有关将 WSIT 用于 Application Server 的信息，另请参阅《The WSIT Tutorial》。

已知问题和限制

本章介绍 Sun Java System Application Server 9.1 软件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 23 页中的 “管理”
- 第 28 页中的 “Apache 和负载均衡器插件”
- 第 29 页中的 “应用程序客户机”
- 第 30 页中的 “捆绑的 Sun JDBC 驱动程序”
- 第 31 页中的 “文档”
- 第 32 页中的 “高可用性”
- 第 40 页中的 “安装”
- 第 43 页中的 “Java EE Tutorial”
- 第 43 页中的 “Java Persistence”
- 第 44 页中的 “生命周期管理”
- 第 45 页中的 “日志记录”
- 第 46 页中的 “Message Queue”
- 第 46 页中的 “Monitoring”
- 第 47 页中的 “样例”
- 第 49 页中的 “安全性”
- 第 50 页中的 “升级实用程序”
- 第 55 页中的 “Web 容器”
- 第 59 页中的 “Web 服务器”
- 第 59 页中的 “Web 服务”

管理

本节介绍已知的管理问题和相应的解决方法。

如果 domain1 不存在，package-appclient 脚本无法运行 (6171458)

说明

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

执行以下操作之一：

1. 保持 domain1 不变，围绕它创建其他域。
2. 删除 domain1 并用新域名替换 `$INSTALL/lib/package-appclient.xml` 中用于 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。

无法使用其他名称恢复备份域 (6196993)

说明

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` 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 8.1 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。

不支持使用其他 JMX 代理启动 Application Server (6200011)

说明

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 虚拟机中启动新的 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 管理控制台和命令行界面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 JMXService URL，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 9.1 Administration Guide》。

在 UNIX 上，对 Application Server 启动和停止脚本的执行权限限制过度 (6206176)

说明

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。如果随后尝试以用户 "B" 的身份（即使 "B" 为超级用户）启动或停止域，则会因为只有 "A" 可以执行脚本而失败。

解决方法

更改脚本的权限：

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

无法使用任一 Web 服务的端点 URL 创建负载均衡器配置文件 (6236544、6275436)

说明

如果某个应用程序具有可导出 Web 服务 URL 的 EJB 模块，则在用该应用程序设置负载均衡器配置时，Web 服务环境中的超级用户不会包含在结果文件 loadbalancer.xml 中。

解决方法

1. 编辑 loadbalancer.xml 文件，按如下所示添加缺少的 Web 模块：

```
<web-module context-root="context-root-name"
disable-timeout-in-minutes="30" enabled="true"/>
```

2. 用作为 EJB 提供的 Web 服务环境中的超级用户名称替换 *context-root-name* 的值。

使用 sun-appserv-admin 重新启动 Application Server 将导致 LoginException 错误 (628893)

解决方法

1. 将现有的 `<as_install>/bin/asant` 脚本重命名为 `asant.bak`。
2. 将 `<as_install>/lib/install/templates/ee` (适用于 SE/EE 版本) 中的 `asant.template` 文件复制到 `<as_install>/bin/` 目录中, 并将其重命名为 `asant`。
3. 编辑新复制的 `<as_install>/bin/asant` 脚本, 用 `<as_install>/config` 替换 `%CONFIG_HOME%` 标记。
4. 如果对原始 `asant.bak` 文件进行了任何手动更改, 请将其并入新的 `asant` 脚本。

Application Server 文档中未介绍 .asadmintruststore 文件 (6315957)

说明

Application Server 文档中未介绍 `.asadmintruststore` 文件。如果服务器管理员的主目录中不包含此文件, 在升级该服务器上的某些应用程序时可能会出现严重错误。

解决方法

- 如果可能, 应该由安装服务器的用户运行 `asadmin start-domain domain1` 命令。
- 如果不是由该用户运行的, 应将 `.asadmintruststore` 从安装用户的主目录移动或复制到运行用户的主目录中。
- 请注意, 如果将该文件从安装用户的主目录移动 (而非复制) 到运行用户的主目录, 可能会出现错误 6309079、6310428 和 6312869 所述的应用程序升级问题, 原因是升级/安装用户 (通常是 Java ES 中的超级用户) 的主目录中不再具有 `.asadmintruststore` 文件。

由于在访问 JMS 代理时超时, 无法启动群集实例 (6523663)

说明

Application Server 群集实例的默认 MQ 集成模式为 LOCAL。如果将 Application Server 安装在长 (即 "not short") 位置 (PATH) 上, 当群集实例启动时, `imqbrokersvc.exe` 会崩溃。此问题是 `imqbrokersvc` 中的内存分配问题。

解决方法

必须将群集实例的 JMS 服务类型从默认的 LOCAL 更改为 REMOTE。在此配置中, 所有实例都指回 DAS 代理。请遵照下面的说明, 以 REMOTE 模式配置群集。

注 - 使用 REMOTE 模式时，所有实例都使用一个代理 (DAS)，因此当 Application Server 群集启动时，不会创建任何代理群集。有关更多信息，请参见

<http://www.glassfishwiki.org/>

[gfwiki/attach/OnePagersOrFunctionalSpecs/as-mq-integration-gfv2.txt](http://www.glassfishwiki.org/gfwiki/attach/OnePagersOrFunctionalSpecs/as-mq-integration-gfv2.txt) 这一页上的第 4.1 条，第 iii 部分中的 "Auto-clustering"。上述功能将不可用！

▼ 使用命令行

开始之前 根据您的环境修改端口和密码文件。请注意，在以下说明中，群集名称为 `racluster`，DAS 管理端口为 5858，DAS JMS 端口为 7676。

- 1 修改群集配置，将 JMS 类型更改为 REMOTE。

```
$AS91_HOME/bin/asadmin.bat set --port 5858 --user admin --passwordfile \
$AS91_HOME/bin/password_file racluster.jms-service.type=REMOTE
```

- 2 创建对应于 DAS JMS 主机的 JMS 主机。

```
$AS91_HOME/bin/asadmin.bat create-jms-host --port 5858 --user admin --passwordfile \
$AS91_HOME/bin/password_file --target racluster --mqhost localhost --mqport 7676 \
--mquser admin --mqpassword admin dashost
```

- 3 将默认的 JMS 主机设置为在上一步骤中创建的 DAS JMS 主机。

```
$AS91_HOME/bin/asadmin.bat set --port 5858 --user admin --passwordfile \
$AS91_HOME/bin/password_file racluster.jms-service.default-jms-host=dashost
```

▼ 使用管理 GUI

- 1 转至“配置”->“*cluster_name-config*”->“Java 消息服务”->“JMS 主机”。

- 2 单击“新建”以创建新的 JMS 主机，将其命名为 `dashost`。

- 3 输入对应于 DAS JMS 服务的配置设置，默认设置如下所示：

- 主机名：localhost
- 端口：7676
- 管理员用户：admin
- 密码：admin

根据您的 DAS JMS 服务修改这些设置。

- 4 导航回“Java 消息服务”选项卡，并将 JMS 服务类型更改为 REMOTE（默认为 LOCAL）。

- 5 从“default-jms-host”下拉式列表中，选择“dashost”。

- 6 保存更改，然后启动节点代理或群集。

无法在 Netscape 8.1.3、Mozilla 1.7 和 Safari 2.0.4 浏览器中显示 jmaki 图表 (6543014)

说明

当尝试使用某些不支持的浏览器显示“日志统计信息监视”页中的图表时，可能会抛出以下错误：

```
Error loading jmaki.widgets.jmaki.charting.line.Widget : id=form1:jmaki_chart11
Script: http://easqelx5.red.iplanet.com:4848/resources/jmaki/charting/ \
line/component.js (line:5437).
Message: area.initialize is not a function
```

解决方法

使用支持的浏览器。有关 Application Server 9.1 支持的浏览器的列表，请参阅第 15 页中的“浏览器”。

默认端口在每个 AS 主要发行版中都有所不同 (6566481)

说明

默认管理端口已在过去三个主要 Application Server 发行版中都有所不同。具体而言，7.x、8.x 和 9.x 中的默认管理端口如下所示：

- AS 7.x: 4848
- AS 8.x: 4849
- AS 9.x: 4848

解决方法

这不是错误，但是一个需要注意的事项。默认管理端口只是一个建议端口。希望在以后的 Application Server 版本中将保留默认 4848 端口。

Apache 和负载均衡器插件

本节介绍 Apache Web 服务器和负载均衡器插件的已知问题和相应的解决方法。

《高可用性管理指南》中所包含的将 openssl 用于 Apache 的说明不正确 (6306784)

在编译和生成 openssl 时，请使用以下命令：

```
cd openssl-0.9.7e
config
make
```

另外，对于 Apache 1.3，mod_ssl 源的目录名称会因使用的 Apache 版本而异。例如，对于 Apache 1.3.33，该名称为 mod_ssl-2.8.22-1.3.33。

《高可用性管理指南》中不包含使用 Apache 2.0 证书的说明 (6307976)

要运行 Apache 安全性，就必须使用证书。有关从证书授权机构获取证书的说明，请参见 [modssl 常见问题解答](http://www.modssl.org/docs/2.8/ssl_faq.html#ToC24) (http://www.modssl.org/docs/2.8/ssl_faq.html#ToC24) 中有关证书的信息。

必须以超级用户的身份启动 Apache Web 服务器 (6308021)

在 Solaris 上，如果 Application Server 由超级用户安装，则必须以超级用户的身份启动 Apache Web 服务器。必须以超级用户的身份来安装 Java Enterprise System 软件。对于 Apache 2.0，在以超级用户的身份启动后，Apache 会切换到您指定的另一用户并以该用户的身份运行。在 `/conf/httpd.conf` 文件中指定该用户。要以超级用户的身份启动，在很多系统中都必须编辑 `httpd.conf` 文件以指定正确的组。将行：

```
Group #-1
```

替换为：

```
Group nobody
```

有关用户/组使用的信息包含在 `httpd.conf` 文件中。

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 将覆盖 MANIFEST 文件 (6193556)

说明

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

目前尚无解决方法。

ACC 始终尝试连接到 localhost:3700 (6527987)

说明

应用程序客户机始终尝试连接到 `localhost:3700`。问题在于在调用客户机代码之前，需要读取多个系统属性。

解决方法

将以下各项设置为系统属性（`JAVA_CMD` 中的 `-D`）。请勿在应用程序客户机代码中设置它们：

```
org.omg.CORBA.ORBInitialHost = server_instance_host  
org.omg.CORBA.ORBInitialPort = server_instance_port
```

无法启动域，缺少 sunpkcs11.jar (6571044)

说明

如果在 64 位 Linux 上运行，则启动域时，会出现以下异常。问题在于 jdk1.5.0_11/jre/lib/ext/ 下缺少 sunpkcs11.jar。

解决方法

这是 64 位 Linux 中的已知 JDK 错误，将在 JDK 1.5.0_13 中得以修正。

ASQuickStartup 中断了 SocketChannel.keyFor()，返回 null 而不是 SelectionKey (问题跟踪器 3027)

说明

在多个选择器上注册 SocketChannel 时，执行 socketChannel.keyFor(lastRegisteredSelector) 会返回 null 而不是 SelectionKey。

解决方法

此问题与 JDK 错误 (6562829) 相关，希望在 6.0 U3 中得以修正。Application Server 9.1 中已包括解决方法，因此，在调用 keyFor API 之前打开选择器。这样可使 keyFor 继续作用，直至 JDK 错误得以修正。

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。

PreparedStatement 错误 (6170432)

描述 1

如果应用程序在一个事务中生成 3000 个以上 PreparedStatement 对象，DB2 可能会发生以下错误：

[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《Sun Java System Application Server 9.1 Administration Guide》。

描述 2

可能抛出的与上述 `PreparedStatement` 错误相关的另一条错误消息为：

```
[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

隔离级别为 `TRANSACTION_SERIALIZABLE`。如果应用程序使用隔离级别 `TRANSACTION_SERIALIZABLE` 并使用上面建议的某个参数，该应用程序可能会在获取连接时挂起。

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《Sun Java System Application Server 9.1 Administration Guide》。

重新引导计算机或启动 Application Server 之后，无法启动 Java DB (6515124)

说明

重新引导主机系统或 Solaris 区域或者启动 Application Server 之后，捆绑的 Java DB 数据库无法自动重新启动。这不是错误，而是任何捆绑的应用程序或第三方应用程序的预期行为。问题在于必须在 Application Server 实例之前启动 Java DB。

解决方法

重新引导主机或 Solaris 区域之后，请务必在启动 Application Server 之前启动 Java DB，例如：

```
/opt/SUNWappserver/appserver/bin/asadmin start-database
```

有关 `asadmin` 命令选项的更多信息，请参阅《Sun Java System Application Server 9.1 Quick Start Guide》中的“Application Server Administration Tools”。

文档

本节介绍已知的文档问题和相应的解决方法。

Javadoc 不一致（各种 ID）

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

- ConnectorConnectionPoolStats 和 AltJDBCConnectionPoolStats 中缺少用于获取 NumConnAcquired 和 NumConnReleased 统计信息的 getter 方法。这些 getter 方法将以 getNumConnAcquired() 和 getNumConnReleased() 的形式添加到将来的发行版中。
- 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。
- 常数 XTypes.CONNNECTOR_CONNECTION_POOL_MONITOR 拼写错误 ("NNN")。在以后的版本中将纠正此问题。

捆绑的 ANT 抛出 java.lang.NoClassDefFoundError (6265624)

说明

线程 "main" 中会抛出以下异常：java.lang.NoClassDefFoundError: org/apache/tools/ant/launch/Launcher。

解决方法

建议不要对 Application Server 外的对象使用捆绑的 ANT。

Id:

6615557

摘要:

本地化联机帮助的“内容”选项卡下丢失 3 页

说明:

本地化联机帮助丢失“内容”选项卡下的以下 3 个已本地化页面:

- 资源 -> JavaMail Session -> 管理 JavaMail Session 目标

- 资源 -> 连接器 -> 编辑连接池属性

- 连接器 -> 编辑连接持属性 -> 管理安全映射

解决方法:

请使用这些页面的英文版联机帮助。

高可用性

本节介绍已知的高可用性数据库 (HADB) 问题和相应的解决方法。

使用双网络进行 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，由于操作系统或网络驱动程序在某些硬件平台上的问题，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。但并不总是会发生这种情况，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)。

HADB 数据库创建失败（无 ID）

说明

创建新数据库可能会失败并出现以下错误，说明可用的共享内存段太少：

HADB-E-21054:系统资源不可用:HADB-S-05512:用关键字 "xxxxx" 连接共享内存段失败，操作系统状态=24，操作系统错误消息:打开的文件太多。

解决方法

请确认已配置共享内存且配置能正常工作。特别是在 Solaris 8 上，请检查文件 `/etc/system`，然后确定变量 `shmsys:shminfo_shmseg` 的值至少为每个主机的节点数的六倍。

hadbm set 无法检查资源可用性（磁盘和内存空间）(5091280)

说明

使用 `hadbm set` 增大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，确认所有主机上都有足够的可用磁盘空间/内存空间。

不支持 packagepath 的异构路径 (5091349)

说明

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/install1 --hosts europa11
Package successfully registered.
hadbm registerpackage test --packagepath=/var/install2 --hosts europa12
hadbm:Error 22171: A software package has already been registered with
the package name test.
```

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) 在所有参与的主机上都相同。

createdomain 可能会失败 (6173886、6253132)

说明

在具有多个网络接口的主机上运行管理代理时，如果所有网络接口不是在同一子网中，则 createdomain 命令可能会失败：

```
hadbm:Error 22020: The management agents could not establish a
domain, please check that the hosts can communicate with UDP multicast.
```

管理代理将（如果不是采用其他配置）使用 UDP 多址广播的“第一个”接口（“第一个”接口由 `java.net.NetworkInterface.getNetworkInterfaces()` 的结果定义）。

解决方法

最佳解决方法是告诉管理代理要使用哪个子网（在配置文件中设置 `ma.server.mainternal.interfaces`，例如，`ma.server.mainternal.interfaces=10.11.100.0`）。此外，也可以配置子网之间的路由器，以便路由多址广播数据包（管理代理使用多址广播地址 228.8.8.8）。

在重试管理代理的新配置之前，可能需要清除管理代理系统信息库。停止域中的所有代理，并删除系统信息库目录（由管理代理配置文件中的 `repository.dr.path` 标识）中的所有文件和目录。必须先在所有主机上执行此操作，然后才能用新配置文件重新启动代理。

启动、停止和重新配置 HADB 可能会失败或挂起 (6230792、6230415)

说明

在 Solaris 10 Opteron 上，使用 hadbm 命令启动、停止或重新配置 HADB 可能会失败或挂起，并产生以下错误消息：

```
hadbm:Error 22009: The command issued had no progress in the last
300 seconds.
HADB-E-21070: The operation did not complete within the time limit,
but has not been cancelled and may complete at a later time.
```

如果 `clu_noman_srv` 进程所使用的文件 (`nomandevic`) 存在不一致的读/写操作，就可能出现这种情况。通过在 HADB 历史文件中查找以下消息，可以检测到此问题：

```
n:3 NSUP INF 2005-02-11 18:00:33.844 p:731 Child process noman3 733
does not respond.
n:3 NSUP INF 2005-02-11 18:00:33.844 p:731 Have not heard from it in
104.537454 sec.
n:3 NSUP INF 2005-02-11 18:00:33.844 p:731 Child process noman3 733
did not start.
```

解决方法

以下解决方法未经验证，原因是此问题尚未手动再现。但是，对受影响的节点运行此命令应该能解决此问题。

```
hadbm restartnode --level=clear nodeno dbname
```

请注意，该节点的所有设备都将重新初始化。在重新初始化之前可能必须停止该节点。

管理代理会终止并产生异常“IPV6_MULTICAST_IF 失败”(6232140)

说明

在运行 Solaris 8 且装有多个 NIC 卡的主机上启动时，如果混合启用了 IPv6 卡和 IPv4 卡，管理代理可能会终止并产生异常“IPV6_MULTICAST_IF 失败”。

解决方法

将环境变量 JAVA_OPTIONS 设置为 -Djava.net.preferIPv4Stack=true；例如：

```
export JAVA_OPTIONS="-Djava.net.preferIPv4Stack=true"
```

此外，也可以使用 Solaris 9 或更高版本，这些版本不会出现此问题。

clu_trans_srv 无法中断 (6249685)

说明

在 Red Hat Enterprise Linux 3.0 的 64 位版本中有一个错误，使 clu_trans_srv 进程在执行异步 I/O 时以不可中断模式结束。这意味着 kill -9 不起作用，必须重新引导操作系统。

解决方法

使用 Red Hat Enterprise Linux 3.0 的 32 位版本。

hadbm 不支持包含大写字母的密码 (6262824)

说明

当密码存储在 hadb 中时，其中的大写字母会转换成小写字母。

解决方法

请勿使用包含大写字母的密码。

从 HADB 版本 4.4.2.5 降级到 HADB 版本 4.4.1.7 会导致 ma 失败，并产生不同的错误代码 (6265419)

说明

降级到先前的 HADB 版本时，管理代理会失败并产生不同的错误代码。

解决方法

可以降级 HADB 数据库，但如果已经对系统信息库对象进行更改，则无法降级管理代理。在降级后，必须仍然使用来自最新 HADB 版本的管理代理。

安装/删除和 symlink 保留 (6271063)

说明

对于 HADB c 软件包 (Solaris: SUNWhadb, Linux: sun-hadb-c) <m.n.u-p> 版的安装/删除，symlink /opt/SUNWhadb/<m> 在创建后不会有任何改动。因此，可能会存在孤立的 symlink。

解决方法

除非正在使用 symlink，否则请在安装前或卸载后将其删除。

全局区域和局部区域中的管理代理可能会发生冲突 (6273681)

说明

在 Solaris 10 上，在全局区域中使用 ma-initd 脚本停止管理代理会导致局部区域中的管理代理也停止。

解决方法

请勿在全局区域和局部区域中都安装管理代理。

当会话对象在 MA 上已超时并被删除时，hadbm/ma 应该提供更准确的错误消息 (6275103)

说明

有时，服务器上的资源争用问题可能会导致管理客户机断开连接。在重新连接时，可能会返回误导性的错误消息“hadbm: 错误 22184: 必须提供密码才能连接到管理代理”。

解决方法

有时，服务器上的资源争用问题可能会导致管理客户机断开连接。在重新连接时，可能会返回误导性的错误消息“hadbm: 错误 22184: 必须提供密码才能连接到管理代理”。

检查服务器上是否存在资源问题，采取适当的措施（例如添加更多资源），然后重试操作。

非超级用户无法管理 HADB (6275319)

说明

与 Java Enterprise System 一起安装（以超级用户的身份）仅允许超级用户管理 HADB。

解决方法

要管理 HADB，请始终以超级用户的身份登录。

管理代理不应使用特殊用途的接口 (6293912)

说明

IP 地址类似于 0.0.0.0 的特殊用途的接口不应注册为可供管理代理中的 HADB 节点使用的接口。如果用户使用主机名称而不是 IP 地址发出 `hadbm create` 命令，将 HADB 节点设置到这些接口上，则注册这些接口可能会导致出现此问题。此后，节点将无法通信，从而导致 `create` 命令挂起。

解决方法

在具有多个接口的主机上使用 `hadbm create` 时，始终明确使用 DDN 表示法来指定 IP 地址。

在 Windows 上重汇编失败 (6291562)

说明

在具有某些配置和负载的 Windows 平台上，操作系统中可能会出现大量的重汇编失败。已经发现 20 多个节点的配置在并行运行多个表扫描 (`select *`) 时有此问题。症状可能是事务频繁地异常中止，修复或恢复会持续很长时间才能完成，系统的各个部分会频繁超时。

解决方法

要修复此问题，可将 Windows 注册表变量

`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters` 设置为高于默认值 100 的值。建议将此值增大到 0x1000 (4096)。有关更多信息，请参见 Microsoft 支持页中的文章 811003

(<http://support.microsoft.com/default.aspx?scid=kb;en-us;811003>)。

如果浏览器具有其他路径为 / 的 cookie，无法维护会话状态 (6553415)**说明**

路径为 "/" 的 Cookie 与在 "/" 以外的上下文根目录中部署的高度可用 Web 应用程序（使用内存复制作为其持久性类型）的 Cookie 发生冲突，从而使高度可用的 Web 应用程序无法维护任何 HTTP 会话状态。通常，当使用同一浏览器访问管理 GUI（部署在 "/" 中）和高度可用的 Web 应用程序时，可能会出现这种情况。

解决方法

通过其他浏览器访问部署在 "/" 中的 Web 应用程序。

LB 无法使用 IIS 6；<appserver-install>/lib 下缺少 SASL32.DLL 和 ZLIB.DLL (6572184)**说明**

负载均衡器使用 Windows IIS 6 时需要 SASL32.DLL 和 ZLIB.DLL 文件。当前在 <appserver-install>/lib 下没有这些文件。

解决方法

将这两个 DLL 文件手动复制到 <appserver-install>/lib 中。可以从以下网址下载这些文件：

<http://download.java.net/javaee5/external/<OS>/aslb/jars/aslb-9.1-MS4-b5.jar>

其中 <OS> 表示所需的平台，可以是以下值之一：

- SunOS
- SunOS_X86
- Linux
- WINNT

全局区域中存在 DAS 创建/启动和 HA 软件包传播问题 (6573511)**说明**

在全局区域中安装或卸载具有高可用性软件包的 Application Server 时，会出现两个问题：

1. 在所有区域中都安装 HA 软件包，这可能不是所希望的。
2. 卸载时，将从所有区域中删除 HA、MQ、JDK 软件包，这可能不是所希望的。

从局部根区域中安装或卸载时，不会出现此问题。

解决方法

从局部根区域而不是全局区域执行安装和卸载。

部署在 "/" 中的高度可用的 Web 应用程序无法恢复在内存中复制的 HTTP 会话（问题跟踪器 2972）

说明

当使用内存复制作为其持久性类型时，部署在 "/" 中的高度可用的 Web 应用程序无法维护任何 HTTP 会话。

解决方法

将使用内存复制作为其持久性类型的高度可用的 Web 应用程序部署到 "/" 以外的上下文根目录中。如果要使此类 Web 应用程序位于 "/" 中，可以将其指定为 Web 应用程序已部署到的虚拟服务器的默认 Web 模块。

AS LB 安装程序未将 /usr/lib/mps 路径放在 apachectl LD_LIBRARY_PATH 中，无法启动 Apache SSL (6591878)

说明

在 Solaris 上针对 Apache 安装 Application Server 负载均衡器期间，安装程序将更新 apachectl 脚本中的 LD_LIBRARY_PATH。但是，安装程序未正确写入 /usr/lib/mps 路径。在 Solaris 上，如果 LD_LIBRARY_PATH 中没有此路径，则 Apache 安全性实例将无法启动。

解决方法

只有 Solaris 平台存在此问题。要解决此问题，请将 /opt/SUNWappserver/appserver/lib/lbplugin/lib 添加到 LD_LIBRARY_PATH。

针对实例/群集启用/禁用 LB 应该显示正确的状态 (6595113)

说明

无论 domain.xml 中保存了哪些内容，在“群集/实例”常规页上，“启用负载均衡”按钮都始终处于启用状态。

解决方法

- 对于群集实例，请选择“实例”选项卡，然后在下拉菜单中单击“停止”操作。
- 对于独立实例，请确保实例正在运行，然后在实例的“常规”屏幕上单击“停止”按钮。

AS9.1 EE IFR b58f/JES5 UR1。无法安装 Registry Server，因为已检测到“不完整的”HA。(6602508)

说明

（仅 Solaris）在具有 HADB 的 SPARC Solaris 10 上安装 Application Server 9.1 之后，启动 Application Server 并尝试安装具有 Registry Server 的 JES 5 UR1 时，可能会出现以下错误：

Dependency Error: Installation can not proceed because the version of HA Session Store 4.4.3 detected on this host is incomplete , and a compatible version is required by Servservice Registry Deployment Support.

解决方法

在 Solaris 计算机上，不能使用 Application Server 9.1 IFR 从 JES 5 UR1 安装 Registry Server。必须使用 pkgadd 命令从以下 JES5 UR1 分发目录手动安装 Registry Server 软件包：

<path>/<OS>/Products/registry-svr/Packages

特定于 IE 6.0 浏览器：导出负载均衡器配置文件会抛出错误 (6516068)

说明

（仅 Internet Explorer 6）当尝试从 Internet Explorer 6 导出负载均衡器配置文件 (loadbalancer.xml) 时，浏览器会显示错误消息，表示找不到 sun-loadbalancer_1_2.dtd DTD 文件。

解决方法

要保存此文件，请使用以下解决方法：

1. 在 Internet Explorer 中，在“负载均衡器”页上单击“导出”。
将显示“XML page cannot be displayed”消息。
2. 单击错误框，然后从 Internet Explorer 中选择“文件”->“另存为”。
3. 将 loadbalancer.xml 文件保存到所选目录中。

安装

本节介绍已知的安装问题和相应的解决方法。

在某些 Linux 系统上，单击“完成”按钮之后，安装关闭会挂起 (5009728)

说明

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 Linux Red Hat 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，请点击产品“关于”页面上提供的指向注册页面的链接。

在 Windows 上，需要在安装期间创建 imq 目录 (6199697)

说明

刚刚在 Windows 上安装 Application Server Enterprise Edition 之后，Message Queue 代理启动失败，并显示一条消息，说明目录 drive:\as\domains\domain1\imq 不存在。

请注意，如果在启动 domain1 后启动该代理，则 Application Server 将创建该目录，因此不会出现上述问题。

解决方法

1. 在创建代理之前创建 var_home_dir_location :

```
$imqbrokerd -varhome var_home_dir_location
```

例如：

```
$imqbrokerd -varhome D:\as\domains\domain1\imq
```

在 Windows Vista 上安装 SDK 时出现虚假警告 (6533646)

说明

在 Windows Vista 上安装捆绑的 SDK 时，可能会遇到错误“Unsupported Installation Platform Detected”。但是，安装却成功而没有出现任何问题。

解决方法

这实际上并不是问题。Application Server 在 Windows Vista 上运行，此错误消息将在以后的产品版本中删除。

卸载无法正确更新 productregistry 文件；无法使用无提示模式进行安装 (6571598)

说明

如果 Application Server productregistry 文件包含共享组件配置，则 Application Server 卸载过程无法正确更新 productregistry 文件，您将无法在后续安装中使用无提示模式，除非重命名或删除 productregistry 文件。按照设计，productregistry 文件中的共享组件条目保持不变，但是会导致与后续无提示安装发生混淆。

解决方法

通过卸载日志文件报告卸载成功之后，先删除 productregistry 文件，然后再运行后续安装。要检验先前的卸载是否已成功完成，请在 <install_dir> 中查找 appserv_uninstall.class 文件。如果卸载成功，此文件将不会存在。

注 – 如果卸载失败，请勿删除 productregistry。

在 Solaris 上，productregistry 文件位于 /var/sadm/install 中；在 Linux 上，位于 /var/tmp 中。

IFR。无法在稀疏局部区域中安装 AS，MQ 软件包问题。(6555578)

说明

在稀疏局部区域中安装 Application Server 时，如果未先安装 Message Queue (MQ)，安装便会失败。安装程序尝试安装 MQ，随后整个安装失败。

解决方法

在稀疏局部区域中安装 Application Server 之前，必须在全局区域中手动安装 MQ。此问题有两种解决方法：

1. 通过包含 Application Server 9.1 IFR 安装的媒体，在全局区域中手动安装 MQ 4.1，以获得最新的 MQ 软件包。

- a. 使用与您的平台相对应的安装程序：

```
mq4_1-installer-SunOS.zip
mq4_1-installer-SunOS_X86.zip
mq4_1-installer-Linux_X86.zip
mq4_1-installer-WINNT.zip
```

- b. 解压缩位并运行安装程序。

安装程序将位于 mq4_1-installer 目录中。

2. 在全局区域中安装任何 IFR 安装组件。此操作将检查 GZ 中的 MQ 版本，确定是否需要将其升级到 Application Server 9.1 IFR 中捆绑的版本。均等选择并安装样例应用程序组件将 MQ 升级到 IFR 版本。

- a. 在全局区域中运行 Application Server 安装，但是仅选择样例组件。

样例组件安装也会在所有区域中安装 MQ 和 Application Server 共享组件。

- b. 再次运行 Application Server 安装，这次是在局部稀疏区域中。

安装应该顺利完成，不出现任何问题。

必须删除 IFR 安装 - console 提示 “Do you want to upgrade from previous Application?” (6592454)

说明

使用 -console 选项（命令行模式）运行 Application Server 9.1 IFR 安装程序时，系统会对您进行以下提示：

```
Do you want to upgrade from previous Application Server version?
```

遗憾的是，IFR 安装程序不支持此升级，因此该提示是错误的。如果您对该提示回答是“是”，则安装将正常进行，但是不会出现表明已执行完整安装而不是升级的指示。

解决方法

如果要升级 Application Server 安装，请使用升级工具。

Java EE Tutorial

要运行有关 Sun Java System Application Server 9.1 的《Java EE Tutorial》，请执行以下任务：

- 按照“关于本教程”一章的“关于示例”一节中的描述编辑文件示例 `/common/build.properties` 时，还要将端口 4848 更改为 4849。

注 - Application Server 9.1 中的默认管理端口也是 4848。有关更多信息，请参见第 28 页中的“默认端口在每个 AS 主要发行版中都有所不同 (6566481)”。

- 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。

MDB 在 Java EE Tutorial 示例中失败 (6591307)

说明

在《The Java EE 5 Tutorial》中的第 32 章“Java EE Examples Using the JMS API”，《The Java EE 5 Tutorial》中的“An Application Example That Consumes Messages from a Remote Server”，此示例不再起作用。MDB 无法接收消息。在两个系统之间发送消息的其他两个示例仍正确运行（《The Java EE 5 Tutorial》中的“Running JMS Client Programs on Multiple Systems”和《The Java EE 5 Tutorial》中的“An Application Example That Deploys a Message-Driven Bean on Two Servers”）。

解决方法

将在以后的 Application Server 内部版本中得以修复。

Java Persistence

TopLink 期望 Collection 字段/属性可复制（问题跟踪器 556）

说明

如果 `java.util.Arrays.asList()` API 用于将 `Object[]` 转换为 `Collection`，则 JDK 将返回不可复制的 `java.util.ArrayList` 的实现。这将导致以下异常：

```
The method invocation of the method [protected native java.lang.Object
java.lang.Object.clone() throws java.lang.CloneNotSupportedException] on the object
[[pkg.A id = xxx]], of class [class java.util.Arrays$ArrayList], triggered an
exception. Internal Exception: java.lang.reflect.InvocationTargetException Target
Invocation Exception: java.lang.CloneNotSupportedException:
java.util.Arrays$ArrayList
```

将在 https://glassfish.dev.java.net/issues/show_bug.cgi?id=556 中跟踪此问题。

解决方法

使用其构造函数创建其他集合，例如：

```
myCollection = new ArrayList(java.util.Arrays.asList(a))
```

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。

设置 `ejb-timer-service` 属性会导致 `set` 命令失败 (6193449)

说明

将 `ejb-timer-service` 属性 `minimum-delivery-interval` 设置为 `9000` 之后，尝试将 `ejb-timer-service` 属性 `redelivery-interval-in-millis` 设置为 `7000` 会导致 `set` 命令失败，并出现以下错误：

```
[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`。Application Server 在确认 `redelivery-interval-in-millis` 的值是否大于 `minimum-delivery-interval-in-millis` 的值时使用了错误的验证检查，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。

当列出非 DAS 配置中的 JMS 物理目的地时，会抛出错误(6532532)

说明

如果要尝试使用 default-config 查看 JMS 物理目的地，将会出现错误消息。

解决方法

这是预期行为。在 Application Server 9.1 中，default-config 是配置信息的模板，因此，无法针对 default-config 执行 JMS 操作（例如 list 和 create）。但是，可以针对群集或独立实例的配置执行这些 JMS 操作。

仅 Win2003：非分页池泄漏内存，破坏 tcp 栈和丰富访问测试(6575349)

说明

（仅 Windows 2003）在 Windows 2003 系统上，当执行丰富访问功能时，会出现内存泄漏。出现此问题是因为 Win32 非分页池不断增长，最终破坏整个 TCP/IP 栈。出现故障后，TCP/IP 栈将保持可恢复状态，并且仅可通过重新引导 Windows 2003 系统对其进行恢复。

解决方法

这是 Microsoft 问题（问题编号：SRX070906600011），已具有可用于此问题的修补程序。有关更多信息，请与 [Microsoft 支持](#) 联系。

解决方法

除了上述的修补程序之外，此问题还有两种解决方法。

1. 通过配置 domain.xml http-listener 属性 blocking-enabled="true" 使用 Grizzly 阻塞模式，或者添加以下 http-listener 属性：

```
<property name="blocking" value="true"/>
```

2. 使用 Windows Vista 或 Windows XP。

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。

设置 access, failure 的调试语句会导致 Application Server 启动挂起 (6180095)

说明

为 JVM 设置 java.security.debug 选项会导致服务器实例的启动停止并死锁；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。

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（6173308、6189645、6198481、6199510、6208728）

说明

多种问题均可导致在与时间相关的情况下重新连接失败。

解决方法

可以通过以下方法解决这些问题：

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

在 Linux 上，MQ 代理无法与群集配置文件一起启动 (6524871)

说明

在 Linux 系统上，创建具有群集配置文件的域之后，可能会遇到 java.lang.OutOfMemoryError: heap space 错误，并且服务器实例可能会因为 MQ 代理未启动而无法重新启动。出现此情况之后，系统便永不会恢复。问题在于 /etc/hosts 文件配置错误；具体而言，服务器主机名称正在指向回送地址 127.0.0.1。

解决方法

按照设计，MQ 代理群集无法在网络设备配置为指向回送地址的情况下启动。这不是错误。解决方法是确保 Application Server 主机的 /etc/hosts 文件不指向 127.0.0.1。

Monitoring

本节介绍已知的监视问题和相应的解决方法。

某些 HTTP 服务监视统计信息未提供有用的信息，应该被忽略 (6174518)

说明

查看 HTTP 服务的某些元素的监视统计信息时，显示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 的信息，应该被忽略：

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

从管理 UI 打开 JNDI 浏览会在 server.log 中转储大量异常 (6591734)

说明

从管理 GUI 打开 JNDI 浏览器时，会抛出许多异常。

解决方法

目前尚无解决方法。

样例

本节介绍与 Application Server 9.1 产品附带的样例代码相关的已知问题和相应的解决方法。

文档没有明确说明需要创建 JMS 资源 (6198003)

说明

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

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: JMS Destination object not found: 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

将 `asant` 部署目标用于 `build.xml` 脚本，该脚本用于创建运行应用程序所需的 JMS 资源。

在 Linux 上，在 Web 服务/安全性样例中创建证书期间，会显示运行时错误 (6198239)

说明

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

```
generate_certs: [echo] ***Exporting certificate from NSS database
[exec] Result: 1 [echo] ***Generating Java Keystore from generated
certificate [exec] keytool error: java.lang.Exception: Input not an
X.509 certificate [exec] Result: 1 [echo] ***Generating Java trust
store from generated certificate [exec] keytool error: java.lang.
Exception: Input not an X.509 certificate [exec] Result: 1
.
.
.
generate_certs: [echo] ***Exporting server certificate from NSS database to
a PKCS12 certificate file [exec] /opt/sun/appserver/lib/pk12util: /usr/lib/
libnss3.so: version 'NSS_3.9' not found (required by /opt/sun/appserver/lib/
pk12util) [exec] /opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so:
version 'NSS_3.6' not found (required by /opt/sun/appserver/lib/pk12util)
[exec] /opt/sun/appserver/lib/pk12util: /usr/lib/libnss3.so: version
'NSS_3.7' not found (required by /opt/sun/appserver/lib/pk12util) [exec]
Result: 1
```

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。在您的环境中或在 `install_dir/bin/asant` shell 包装程序脚本中设置 `LD_LIBRARY_PATH`。

解决方法

执行以下操作之一：

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

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

升级之后，AS9.1 样例和 JES5 Portal 样例将争用 Derby 端口 1527 (6574563)

说明

在 Windows 上，升级到 Application Server 9.1 之后，样例和 JES5 Portal 样例将争用 Derby 端口 1527。具体而言，Application Server 9.1 将自动在具有 APP:APP 的端口 0.0.0.0:1527 上启动 JavaDB，但是 JES5 Portal JavaDB 要绑定到具有 portal:portal 的 hostnameIP:1527。

此错误所描述的问题已在 JES 5 错误 6472173 中出现。错误 6472173 的解决方法在《Sun Java Enterprise System 5 Installation Guide for Microsoft Windows》有所记录。

解决方法

使用以下命令启动 Derby 数据库：

```
<JES installation dir>\appserver\bin\asadmin start-database --dbhome <JES installation dir>\portal\data\derby
```

安全性

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

SSL 终止不起作用 (6269102)

说明

SSL 终止不起作用；如果为 SSL 终止配置了负载均衡器（硬件），则 Application Server 会在重定向过程中将协议从 https 更改为 http。

解决方法

在硬件负载均衡器与 Application Server 之间添加软件负载均衡器。

SSL 出现套接字连接泄漏 (6492477)

说明

由于 JVM 错误，在 HTTP 侦听器上将 security-enabled 设置为 true 时，某些 JDK 版本会出现泄漏问题。具体而言，产生此错误的步骤如下：

1. 在 HTTP 侦听器上，将 security-enabled 设置为 true：

```
<http-listener acceptor-threads="1" address="0.0.0.0"
blocking-enabled="false" default-virtual-server="server" enabled="true"
family="inet" id=" http-listener-1" port="8080" security-enabled="true"
server-name="" xpowered-by="true">
```

2. 对在快速查找测试结束时停止域做出注释。
3. 运行快速查找测试。
4. 检查套接字使用情况：

```
netstat -an | grep 8080
```

以下显示的是处于使用状态的内容：

```
*.8080          *.*           0      0 49152      0 LISTEN
*.8080          *.*           0      0 49152      0 BOUND
```

此问题在 Glassfish 站点 https://glassfish.dev.java.net/issues/show_bug.cgi?id=849 上也有记录。

解决方法

升级到最新的 JDK 版本。

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。

无法直接升级在 *install_dir/domains* 目录以外的自定义路径中创建的域 (6165528)

说明

从 Application Server Enterprise Edition 8 升级到 Application Server Enterprise Edition 8.1 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。

如果运行升级实用程序并将 *install_dir* 标识为源安装目录，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* 目录中。

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

说明

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 Red Hat 分发中也发现了此问题。

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，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。

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-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。

从 8.0 Platform Edition (PE) 升级到 8.1 Enterprise Edition (EE) UR2 期间和之后，自签名证书都不受信任 (6296105)

解决方法

将以下条目从目标 `domain.xml` 删除（在升级后），然后重新启动服务器：

```
<jvm-options>-Djavax.net.ssl.keyStore=${com.sun.aas.instanceRoot}
/config/keystore.jks</jvm-options>-
<jvm-options>Djavax.net.ssl.trustStore=${com.sun.aas.instanceRoot}
/config/cacerts.jks</jvm-options>
```

升级工具将覆盖实例（群集成员）的已修改 `index.html` 的内容 (6386451)

说明

升级工具将覆盖所有服务器实例的任何现有的 `index.html` 文件。

解决方法

在运行升级工具之前，备份现有的 `index.html` 文件，稍后恢复这些文件。

（`sbs-manual`、`sbs-installer`）抛出一个例外，说明“服务器实例”服务器没有名为 `null` 的系统连接器 (6545145)

说明

从 Application Server 8.0PE 升级到 9.1 时，会抛出错误，说明服务器没有名为 `null` 的系统连接器，并且出现 `sbs-manual` 中所示的无效用户信息。即使在更改硬编码值之后，也会出现相同的错误消息。出现这种情况是因为 8.0 中的 `domain.xml` 在 9.1 中已更改。

解决方法

仅在从 8.0 PE 升级到 9.1 时才会遇到此错误。解决方法是先升级到 8.1、8.2 或 9.0，然后升级到 9.1。

使用不同的内部版本组合进行升级时，会丢失不同的域 (6546130)

说明

执行就地升级时，如果源中有多个域，则即使升级过程被中止，安装程序也会调用升级工具。以 GUI 模式进行调用时，便会出现这种情况。

解决方法

1. 以 CLI 模式进行就地安装，当安装过程结束时，在安装程序提示您选择升级工具时退出。这不会删除域目录中的任何域。应该从 bin 目录手动调用升级工具。
2. 以 GUI 模式进行就地安装时，对域根目录中的域进行备份，以免在安装过程中丢失任何域。当安装过程结束时，在安装程序提示您调用升级工具时退出。如果域已经丢失，请将所有备份域复制到域目录。手动启动升级工具以执行升级。

Solaris：在升级工具中预先输入的主密码是 AS9.1（而不是 AS8.2）中的主密码 (6565825)

说明

从 AS8.2 升级到 9.1 时，9.1 安装中不会继承 8.2 安装中的主密码。这随后会导致在下次管理登录时出现验证错误。

解决方法

Application Server 9.1 中的默认管理密码为 `changeit`。从 8.2 升级之后，要在登录到 9.1 服务器时避免出现问题，请执行以下三种操作之一：

- 在执行升级之前，将 8.2 管理密码更改为 `changeit`。
- 在升级过程中，请勿接受默认管理密码，而是明确输入要使用的密码。
- 使用默认密码登录到 9.1，然后立即更改此密码。

AS9.1 IFR 升级工具未将 JES5U1 MQ 存储迁移到新的 var 主目录 (6573635)

说明

升级工具无法升级数据库或任何形式数据库表，以后也不会支持此类升级。将传输资源引用配置，Application Server 应该继续使用原始数据库和表。如果要更改数据库或传输数据库表，请使用可用于使用中的数据库的工具。

解决方法

执行以下步骤以迁移 MQ 存储：

注 - 请在关闭 AS 8.2 及运行 AS9.1 升级工具之后，**首次启动 AS9.1 之前**执行以下步骤。如果已在 IFR 安装/升级之后启动了 AS 9.1，**请勿**执行这些步骤，因为它们可能会破坏 MQ 消息存储的稳定性。

1. 将整个 domains/domain1/imq 子目录从 AS 8.x domains 目录复制到 AS 9.1 domains 目录。
2. 确保目录和文件归要运行 Application Server 的用户所有。

执行上述步骤之后，便可启动 Application Server 9.1，并且 Application Server 9.1 domains 目录中的 MQ 存储将从其 JES5 U1 格式迁移到 MQ 4.1 格式。请注意，执行此过程或者在 AS 9.1 启动 MQ4.1 时，将保留 AS 8.2 下的原始 JES5 U1 MQ 存储，不对其进行修改。

将 JES5 (AS 8.2) 升级到 AS 9.1。PS Community 样例会破坏，出现许多 javax.faces 错误 (6576700)

说明

从 JES5 (Application Server 8.2) 升级到 Application Server 9.1 时，Portal Server Community 样例不再起作用，并且会抛出许多 javax.faces.application.ApplicationFactory 错误。

解决方法

如果使用 JES5 Portal Server 安装 Application Server 8.2，则不支持从 Application Server 8.2 升级到 9.1。在 Application Server 升级到 9.1 之前，需要将 Portal Server 升级到 Java ES 5 Update 1。

AS91 IFR b58，Linux：安装程序安装完新的 Java 之后，未更新 /usr/jdk/entsys-j2se 链接 (6591697)

说明

在 Linux 平台上，使用 IFR 安装程序从 Application Server 8.2 升级到 9.1 时，如果选择“安装 JDK”选项，则在成功完成安装之后，大多数 JES 组件停止运行。

解决方法

此问题仅影响在 Linux 平台上进行的 Application Server 9.1 IFR 安装，并且仅当选择“安装 JDK”选项时才会产生影响。要解决此问题，请在安装之后，立即将 /usr/jdk/entsys-j2se 手动链接到 /usr/java/jdk1.5.0_12 目录。

AS9.1 IFR Win 就地备份未与 asupdate.bat 窗体值集成，检查路径错误 (6596412)

说明

在 Windows 上执行 Application Server 9.1 IFR 升级时，就地备份未正确与 asupdate.bat 窗体值集成。具体而言，如果在 ASupdate.bat GUI 屏幕中输入错误的信息，然后单击“下一步”，则升级安装程序将尝试检测它是否为就地升级。如果是，则在升级之前，会将 domain1 移至备份目录。继续升级时，由于信息不正确，会显示错误消息。当您尝试立即更正此错误时，会抛出路径错误，因为 domain1 已被移动。

解决方法

将源目录更改为 {current source path}/backup 中的 domain1_{timestamp} 目录，或者使用“取消”按钮退出安装程序并再次启动。

AS9.1 IFR Windows，内联升级，将 asupgrade.bat 与短路径和长路径一起使用时，会删除 Domain1 (6596394)

说明

(仅 Windows) 如果在程序目录路径中使用特殊字符或 DOS 样式的短名称安装了早期版本的 Application Server，则在以后仍使用这些目录路径名称就地升级到 Application Server 9.1 时，会导致升级失败。

例如，如果将 Application Server 8.2 安装在以下目录中：

```
C:\Program Files (x86)\dirs\appserver  
c:\progra-2\dirs\appserver
```

尝试执行就地升级到 9.1 将失败，因为安装程序无法将短名称或特殊字符转换为所需的长名称格式。

解决方法

安装 Application Server 时，强烈建议路径名称不要包含特殊字符或 DOS 样式的短名称截断（例如 progra~2），因为它会妨碍后续的升级安装。如果存在这样的安装，请在升级之前使用长路径名称重新安装，或者在全新的目录中安装新版本的 Application Server。

AS9.1 IFR : Web UI 验证被破坏 (6569813)

说明

升级 Application Server 之后，<jsp:forward> 标记在 Authenticate.jsp 中未按预期那样发挥作用。<jsp:forward> 调用在服务器日志中生成一个错误，并在 WebUI 中显示空白页。问题在于 Authenticate.jsp 中的 <jsp:forward> 需要 <jsp:forward page="{redirectPage}"/> 之类的页面属性，但是，正在被传送的值却是相对路径（例如 /registry/thin/{pagename}.jsp），即使 Authenticate.jsp 是纯 JSP 页，此类路径也无效。

解决方法

完成 Application Server 升级之后，请使用 `asadmin` 工具运行以下命令，以在 `domain.xml` 中设置 `<auth-realm>`：

1. 转至 `<appserver9.1-install-dir>/bin` 并运行以下命令：

```
./asadmin delete-auth-realm --host localhost --port 6489 certificate
```

这将删除旧的 `auth-realm` 证书（如果存在）。

2. 运行以下命令：

```
./asadmin create-auth-realm --terse=false --echo=true --interactive=true \
--user admin --host localhost --port 6489 --classname \
com.sun.enterprise.security.auth.realm.certificate.CertificateRealm \
--property assign-groups=have.client.cert certificate
```

这将创建具有 `assign-groups` 属性的新的 `<auth-realm>`。

3. 停止并重新启动 Application Server registry 域。

不存在 `asupgrade` GUI 的本地化联机帮助 (6610170)

说明

在非英语语言中运行 `asupgrade` GUI 时，没有针对所选的非英语语言对 GUI 联机帮助进行本土化。

解决方法

目前尚无解决方法。已计划使用所有非英语目标语言对联机帮助进行本地化。

Web 容器

本节介绍已知的 Web 容器问题和相应的解决办法。

在 Windows 上，使用 `--precompilejsp=true` 部署应用程序会锁定应用程序中的 JAR 文件，从而导致以后的取消部署或重新部署失败 (5004315)

说明

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，因为应用程序会从 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 文件会保留在服务器中。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如下的错误。

```
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't
be deleted.
```

解决方法

如果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 且包含空 `<load-on-startup>` 元素的 `web.xml` 部署 WAR (6172006)

说明

`web.xml` 中的可选 `load-on-startup` servlet 元素表示相关的 servlet 将在启动对其进行声明的 Web 应用程序期间被加载和初始化。

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。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/>` 中那样指定空 `<load-on-startup>`，则 `web.xml` 会无法通过依照 Servlet 2.4 模式对其进行的验证，从而导致 Web 应用程序的部署失败。

向下兼容性问题。指定空的 `<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 的装入顺序无关紧要。

无法在资源限定服务器上编译 JSP 页面 (6184122)

说明

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$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 初始化参数设置为 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。

Application Server 不支持 auth-passthrough Web Server 6.1 附加软件 (6188932)

说明

Sun Java System Application Server 9.1 添加了对 Sun Java System Application Server Enterprise Edition 7.1 中可用的 auth-passthrough 插件功能所提供的功能的支持。但是，在 Application Server 9.1 中，auth-passthrough 插件功能的配置有所不同。

Application Server Enterprise Edition 7.1 中的 auth-passthrough 插件功能在双层部署方案中非常有用，其中：

- Application Server 实例受公司防火墙之后的第二层防火墙的保护。
- 不允许客户机直接连接到 Application Server 实例。

在这种网络体系架构中，客户机连接到前端 Web 服务器，而该 Web 服务器配置有 service-passthrough 插件功能，会将 HTTP 请求转发到代理的 Application Server 实例以供处理。Application Server 只能从 Web 服务器代理接收请求，而决不会从任何客户机主机接收请求。因此，当部署在代理的 Application Server 实例上的任何应用程序查询客户机信息时，该应用程序将收到代理主机的信息（例如，当该应用程序查询客户机 IP 地址时，会收到代理主机的 IP），这是因为代理主机才是中继请求的真正发出主机。

解决方法

在 Application Server Enterprise Edition 7.1 中，可以在代理 Application Server 实例上配置 auth-passthrough 插件功能，以使该实例上所部署的所有应用程序可以直接获得远程客户机的信息，就像代理 Application Server 实例直接收到请求那样，而不是通过运行 service-passthrough 插件的中间 Web 服务器接收。

在 Application Server 9.1 中，可以通过将 domain.xml 中 <http-service> 元素的 authPassthroughEnabled 属性设置为 TRUE，来启用 auth-passthrough 功能，如下所示：

```
<property name="authPassthroughEnabled" value="true"/>
```

Application Server Enterprise Edition 7.1 中 auth-passthrough 插件功能的安全性注意事项同样也适用于 Application Server 9.1 中的 authPassthroughEnabled 属性。由于 authPassthroughEnabled 使得能够覆盖可用于进行验证的信息（例如发出请求的 IP 地址或 SSL 客户机证书），因此，仅允许信任的客户机或服务器连接到 authPassthroughEnabled 设置为 TRUE 的 Application Server 9.1 实例是非常重要的。作为一项预防措施，建议仅将公司防火墙之后的服务器的 authPassthroughEnabled 设置为 TRUE，而不要将可通过 Internet 访问的服务器的 authPassthroughEnabled 设置为 TRUE。

请注意，当代理 Web 服务器已配置了 service-passthrough 并且将请求转发到将 authPassthroughEnabled 设置为 TRUE 的 Application Server 8.1 Update 2 实例时，Web 服务器代理上可能启用了 SSL 客户机验证，而在代理的 Application Server 8.1 Update 2 实例

上却禁用了该验证。在这种情况下，代理的 Application Server 8.1 Update 2 仍会将请求当作通过了 SSL 验证，并向部署在其上的发出请求的所有应用程序提供客户机 SSL 证书。

Web 服务器

AS 9.1 b50e。Linux。无法在安装 AS LB 之后启动 WS：libjvm.so:cannot open shared (6572654)

说明

只有在 Linux 系统上将 Sun Java System Web Server 与 Application Server 9.1 和负载均衡器一起使用时才会出现此问题。在此情况下，安装 Application Server 和负载均衡器之后，Web Server 可能无法启动，因为 libicui18n.so.2 和 libicuuc.so.2 发生冲突。这些库同时位于 /opt/sun/private/lib 和 /opt/sun/appserver/lib 中。

解决方法

要使用的正确的库只能位于 /opt/sun/appserver/lib 中，因为 lbplugin 根据这些库生成。一旦从 /opt/sun/private/lib 中删除这两个库，Web Server 便应该能够顺利启动，不会出现任何错误。

或者，如果不希望从 /opt/sun/private/lib 中删除这些库，可以在 Web Server startserv 脚本的 LD_LIBRARY_PATH 中，将 /opt/sun/appserver/lib 放在 /opt/sun/private/lib 之前；也就是将：

```
# Add instance-specific information to LD_LIBRARY_PATH for Solaris and Linux
LD_LIBRARY_PATH="${SERVER_LIB_PATH}:${SERVER_JVM_LIBPATH}:${LD_LIBRARY_PATH}:/opt/sun/appserver/lib:/opt/sun/appserver/lbplugin/lib"; export LD_LIBRARY_PATH
```

替换为：

```
# Add instance-specific information to LD_LIBRARY_PATH for Solaris and Linux
LD_LIBRARY_PATH="/opt/sun/appserver/lib:/opt/sun/appserver/lbplugin/lib:${SERVER_LIB_PATH}:${SERVER_JVM_LIBPATH}:${LD_LIBRARY_PATH}"; export LD_LIBRARY_PATH
```

Web 服务

本节介绍已知的 Web 容器问题和相应的解决办法。

Ant 任务 wsimport 不可用于 Java EE SDK b33d (使用 JDK 1.6)，并出现 NoClassDefFoundError (6527842)

说明

当使用 Java EE SDK b33d 附带的 JDK 1.6 运行 JAX-WS 测试时，可能会遇到问题。测试将立即中止，并出现以下消息：

```
[wsimport] Exception in thread "main" java.lang.NoClassDefFoundError: \
com/sun/tools/ws/WsImport
```

即使 `webservices-tools.jar` 不包含 `com/sun/tools/ws/WsImport.class`、`com/sun/tools/ws/ant/WsImport.class` 和 `com/sun/tools/ws/ant/WsImport2.class`，也会发生此错误。而且，同一个测试工作区可以使用 1.5.0-10 JDK 运行，而不会出现任何问题。

解决方法

在运行 JAX-WS 测试之前，将 `webservices-api.jar` 复制到 `$JAVA_HOME/jre/lib/endorsed`。

publish-to-registry 命令在 IFREE 内部版本中失败 (6602046)

说明

JAXR 使用 SAAJ 将 SOAP 消息发送到注册表。在非 IFR 的情况下，SAAJ impl 类位于 `lib/webservices-rt.jar` 下。在 IFR 的情况下，SAAJ 类仍位于 `lib/webservices-rt.jar` 下。此外，`saaj-impl.jar` 位于 `/usr/share/lib` 目录中。此 jar 文件由 Application Server 拾取，并且其优先级高于 `webservices-rt.jar` 中的类。此 jar 文件不具有将 SOAP 消息发送到 Web 服务注册表所需的必要安全权限。此打包应该修改为向 `/usr/share/lib` 目录下的 jar 授予权限，或者与 `/usr/share/lib` jar 无关。

解决方法

将以下内容添加到 `server.policy` 文件：

```
grant codeBase "file:/usr/share/lib/saaj-impl.jar" {
    permission java.security.AllPermission;
};
```