



Sun Java System Application Server Platform Edition 9 リリース ノート



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

Part No: 819-7011
2008 年 5 月

本書で説明する製品で使用されている技術に関連した知的所有権は、Sun Microsystems, Inc. に帰属します。特に、制限を受けることなく、この知的所有権には、米国特許、および米国をはじめとする他の国々で申請中の特許が含まれています。

U.S. Government Rights – Commercial software. Government users are subject to the Sun Microsystems, Inc. standard license agreement and applicable provisions of the FAR and its supplements.

本製品には、サードパーティーが開発した技術が含まれている場合があります。

本製品の一部は Berkeley BSD システムより派生したもので、カリフォルニア大学よりライセンスを受けています。UNIX は、X/Open Company, Ltd. が独占的にライセンスしている米国ならびにほかの国における登録商標です。

Sun、Sun Microsystems、Sun のロゴマーク、Solaris のロゴマーク、Java Coffee Cup のロゴマーク、docs.sun.com、Java、Solaris は、米国およびその他の国における米国 Sun Microsystems, Inc. (以下、米国 Sun Microsystems 社とします) の商標もしくは登録商標です。Sun のロゴマークおよび Solaris は、米国 Sun Microsystems 社の登録商標です。すべての SPARC 商標は、米国 SPARC International, Inc. のライセンスを受けて使用している同社の米国およびその他の国における商標または登録商標です。SPARC 商標が付いた製品は、米国 Sun Microsystems 社が開発したアーキテクチャーに基づくものです。

OPEN LOOK および SunTM Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザーおよびライセンス実施権者向けに開発しました。米国 Sun Microsystems 社は、コンピュータ産業用のビジュアルまたはグラフィカルユーザーインターフェースの概念の研究開発における米国 Xerox 社の先駆者としての成果を認めるものです。米国 Sun Microsystems 社は米国 Xerox 社から Xerox Graphical User Interface の非独占的ライセンスを取得しており、このライセンスは、OPEN LOOK GUI を実装するか、または米国 Sun Microsystems 社の書面によるライセンス契約に従う米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

この製品は、米国の輸出規制に関する法規の適用および管理下にあり、また、米国以外の国の輸出および輸入規制に関する法規の制限を受ける場合があります。核、ミサイル、生物化学兵器もしくは原子力船に関連した使用またはかかる使用者への提供は、直接的にも間接的にも、禁止されています。このソフトウェアを、米国の輸出禁止国へ輸出または再輸出すること、および米国輸出制限対象リスト(輸出が禁止されている個人リスト、特別に指定された国籍者リストを含む)に指定された、法人、または団体に輸出または再輸出することは一切禁止されています。

本書は、「現状のまま」をベースとして提供され、商品性、特定目的への適合性または第三者の権利の非侵害の黙示の保証を含みそれに限定されない、明示的であるか黙示的であるかを問わない、なんらの保証も行われないものとします。

目次

1	概要	7
	このリリースノートについて	7
	リリースノートの変更履歴	8
	アクセシビリティ機能	8
	関連マニュアル	8
	コメントをお寄せください	10
	補足情報	10
2	Application Server Platform Edition 9 について	13
	9 リリースの新機能	13
	ハードウェアとソフトウェアの要件	16
	プラットフォーム要件	16
	システムの仮想化のサポート	17
	重要パッチ情報	17
	JDBC ドライバとデータベース	17
	バンドル版 Java DB データベースの使用	18
	ブラウザ	22
	Sun Java System Application Server のアップグレード	22
	その他の要件	22
	Java EE 5 プラットフォームの API	23
	Java EE 5 SDK	25
	サポートされている別の Java バージョンへの切り替え	26
	▼ サポートされている別の Java バージョンへ切り替えるには	26
3	既知の問題点と制限事項	27
	管理	27
	domain1 が存在しない場合、package-appclient スクリプトが動作しない。(ID 6171458)	27

JMX エージェントを追加した Application Server の起動はサポートされていない。 (ID 6200011)	28
display-error-statistics で負の数値が返される	29
アプリケーションクライアント	29
アプリケーションクライアントアーカイブのライブラリ JAR が MANIFEST ファイルを上書きする。(ID 6193556)	29
Windows プラットフォーム-クラスパスに APPCPATH が設定されていない (ID 6419847)	29
インストール	30
一部の Linux システムで、「完了」ボタンをクリックしたあとにインストールの 終了でハングアップする。(5009728)	30
Linux 上のインストールラッパーで J2SE の検出およびブートストラップに問題が 発生することがある。(6172980)	30
ライフサイクル管理	31
ejb-timer-service プロパティ minimum-delivery-interval を 9000 に設定したあ とで、ejb-timer-service プロパティ redelivery-interval-in-mills を 7000 に 設定しようとする、set コマンドが次のエラーで失敗する。(ID 6193449)	31
Java Persistence API	32
WHERE 句でサブクエリーを使用する UPDATE または DELETE クエリーを行うと、クエ リーのコンパイル中に NullPointerException が発生する。	32
クエリーコンパイラが、Java Persistence 言語仕様で定義されたすべてのルールを チェックしない。	32
JOIN 変数別にグループ化し、JOIN 変数を直接選択するクエリーを行うと、 SQLException が発生する。	32
クエリーコンパイラが、間違ったトークンを示した紛らわしいエラーメッセージ をスローする場合がある。	33
現在、同じクラスを含む2つの持続性ユニットを、同一の EAR ファイルに配備で きない。	33
persistence.xml のクラス要素を使用して、MappedSuperclass を明示的に一覧表 示できない。	33
継承されたクラスで、@ManyToMany のエンティティーマッピングが失敗す る。	33
関係に List<> を使用すると、MetadataHelper.getAttributeNameFromMethodName で StringIndexOutOfBoundsException が発生する。	33
Null のデータベース値をプリミティブにマップできない。	34
クライアント側から LAZY 初期化関係にアクセスすると、NullPointerException が 発生する場合がある。	34
関係フィールドを選択するクエリーを行った場合、クエリー結果に Null が含ま れない。	34

単一値の関係フィールドに対して定義された JOIN 識別変数を選択するクエリーを行うと、無効な SQL になる場合がある。	34
EntityManager.find() を行うと、誤って IllegalArgumentException が発生する。	35
関係フィールドを java.util.HashSet に設定したエンティティーを持続できない。	35
JavaServer Faces	35
JSF 1.1 - 1.2 仕様違反: 変数およびプロパティリーゾルバ (ID 6419278)	35
アップグレード	37
Application Server Platform Edition 8 から Application Server Platform Edition 9 へのアップグレードで、install_dir /domains ディレクトリ以外のカスタムパスに作成されたドメインが直接アップグレードされない。(ID 6165528)	37
一部の Linux システムで「アップグレードウィザードを起動」ボタンをクリックしたあとに、「同じ場所にアップグレード」を実行するインストーラがアップグレードツールの起動に失敗する。(6207337)	37
アップグレードツールで、JVM オプションの一部が移行されない (ID 6409650)	38
アップグレード後、静的ドキュメントを docroot ディレクトリにコピーする必要がある (ID 6409641)	38
Windows 上でアップグレードツールを実行するときに、Application Server のファイルを閉じる必要がある	38
XWSClientProvider および XWSServerProvider を使用した場合、ユーザー名トークンが暗号化されない (ID 6409652)	38
Web コンテナ	39
空の <load-on-startup/> 要素を持つ Servlet 2.4 ベースの web.xml を使用して WAR を配備できない。(ID 6172006)	39
リソースに制約のあるサーバー上で JSP ページをコンパイルできない。(ID 6184122)	39
ParserUtils が Web アプリケーションからのパーサーを選択する (ID 6412405)	40
EAR バンドル JAR に TLD がない。(Glassfish の問題 590)	41
プラットフォーム固有の問題	41
Windows XP および Windows Server 2003 で denyRemoteHost および allowRemoteHost プロパティを設定するときに、完全修飾されたドメイン名を使用する必要がある (ID 6377272)	41

◆ ◆ ◆

1

第 1 章

概要

Sun Java™ System **Application Server Platform Edition 9** 製品は、Java EE アプリケーションおよび Java Web サービスの開発と配備に利用できる Java EE 5 プラットフォーム互換サーバーです。このサーバーを業務用を使用する場合も費用は発生しません。Sun Java System Application Server Platform Edition の開発、配備、および再頒布は自由に行うことができます。再頒布を検討している場合は、再頒布ライセンスについて [Sun OEM sales](#)

(http://www.sun.com/software/products/appsrvr/appsrvr_oem.html) にお問い合わせください。

このリリースノートには、次の内容が含まれています。

- 7 ページの「このリリースノートについて」
- 8 ページの「リリースノートの変更履歴」
- 8 ページの「アクセシビリティ機能」
- 8 ページの「関連マニュアル」
- 10 ページの「コメントをお寄せください」
- 10 ページの「補足情報」

このリリースノートについて

このリリースノートには、Sun Java System Application Server Platform Edition 9 のリリース時点で入手できる重要な情報が含まれています。ここでは、新機能、拡張機能、既知の問題、制限事項などについて説明します。Application Server 9 をご使用になる前に、このリリースノートをお読みください。

このリリースノートの最新バージョンは、Sun の [マニュアル Web サイト](#) (<http://docs.sun.com/app/docs/coll/1343.3>) から入手できます。ソフトウェアのインストールおよびセットアップを行う前に、この Web サイトを確認してください。また、それ以降も定期的にこのサイトをチェックして、最新のリリースノートや製品マニュアルを参照してください。

このマニュアル内で参照している第三者の URL は、追加の関連情報を提供します。

注- このマニュアル内で引用する第三者の Web サイトの可用性について Sun は責任を負いません。こうしたサイトやリソース上の、またはこれらを通じて利用可能な、コンテンツ、広告、製品、その他の素材について、Sun は推奨しているわけではなく、Sun はいかなる責任も負いません。こうしたサイトやリソース上の、またはこれらを経由して利用可能な、コンテンツ、製品、サービスを利用または信頼したことによって発生した(あるいは発生したと主張される)いかなる損害や損失についても、Sun は一切の責任を負いません。

リリースノートの変更履歴

この節では、リリースノートの改訂履歴を示します。

表 1-1 リリースノートの変更履歴

改訂日付	説明
2007 年 5 月	バージョン 9.0 PE の FCS リリース。
2008 年 5 月	「システムの仮想化のサポート」の節を追加。

アクセシビリティ機能

このメディアの出版以降にリリースされたアクセシビリティ機能を入手するには、米国リハビリテーション法 508 条に関する製品評価資料を Sun に請求し、その内容を確認して、どのバージョンが、アクセシビリティに対応したソリューションを配備するためにもっとも適しているかを特定してください。アプリケーションのアップデートされたバージョンは次の場所にあります。<http://sun.com/software/javaenterprisesystem/get.html>

アクセシビリティ機能に対する Sun のコミットメントについては、<http://sun.com/access> を参照してください。

関連マニュアル

リリースノートに加えて、Application Server 製品にはマニュアルセット全体 (<http://docs.sun.com/app/docs/prod/sjs.asse#hic>) が用意されています。

次の表に、Application Server Platform Edition コアアプリケーションのマニュアルセットに含まれるものを要約して示します。

表1-2 マニュアルセットの内容

マニュアル名	説明
『Sun Java System Application Server Platform Edition 9 Documentation Center』	1 か所で Application Server のすべてのトピックにアクセスできる場所。
『Sun Java System Application Server Platform Edition 9 クイックスタートガイド』	Sun Java System Application Server 製品を使い始めるための入門ガイド。
『Sun Java System Application Server Platform Edition 9 Installation Guide』	Sun Java System Application Server ソフトウェアとそのコンポーネントのインストール。
『Sun Java System Application Server Platform Edition 9 Application Deployment Guide』	Sun Java System Application Server での Java EE アプリケーションのアセンブルと配備をする情報。
『Sun Java System Application Server Platform Edition 9 Developer's Guide』	Sun Java System Application Server 上で動作することを目的とし、Java EE コンポーネントおよび API のオープン Java スタンダードモデルに準拠した、Java 2 Platform, Enterprise Edition (Java EE™ プラットフォーム) アプリケーションの作成と実装。開発ツール、セキュリティー、アセンブリ、配備、デバッグ、ライフサイクルモジュールの作成などについての全般的な情報も提供します。
『Sun Java System Application Server Platform Edition 9 Java EE Tutorial』	Java EE アプリケーションを開発するための Java EE 5 プラットフォームテクノロジーおよび API の使用と、それらアプリケーションの Sun Java System Application Server への配備。
『Sun Java System Application Server Platform Edition 9 Administration Guide』	Sun Java System Application Server サブシステムおよびコンポーネントの、管理コンソールからの設定、管理、および配備。
『Sun Java System Application Server Platform Edition 9 Administration Reference』	Sun Java System Application Server 設定ファイル <code>domain.xml</code> の編集。
『Sun Java System Application Server Platform Edition 9 Upgrade and Migration Guide』	Application Server 6.x、7、8.x から新しい Sun Java System Application Server プログラミングモデルへのアプリケーションの移行。このガイドでは、直前の製品リリースとの違いと、製品仕様との互換性がなくなる可能性のある設定オプションについても説明します。
『Sun Java System Application Server Platform Edition 9 Troubleshooting Guide』	Sun Java System Application Server におけるトラブルの解決。
『Sun Java System Application Server Platform Edition 9 Error Message Reference』	Sun Java System Application Server のエラーメッセージの解決。

表 1-2 マニュアルセットの内容 (続き)

マニュアル名	説明
『Sun Java System Application Server Platform Edition 9 Documentation Center』	1 か所で Application Server のすべてのトピックにアクセスできる場所。
『Sun Java System Application Server Platform Edition 9 Reference Manual』	Sun Java System Application Server で使用できるユーティリティーコマンド。マニュアルページのスタイルで記述されています。asadmin コマンド行インタフェースも含まれます。

コメントをお寄せください

弊社では、マニュアルの改善に努めており、お客様からのコメントおよびご忠告をお受けしております。

<http://docs.sun.com> にアクセスして「コメントの送信」をクリックしてください。このオンラインフォームでは、マニュアルのタイトルと Part No. もご記入ください。Part No. は、7桁か9桁の番号で、マニュアルのタイトルページまたは最初のページに記載されています。たとえば、このリリースノートの場合、タイトルは『Sun Java System Application Server Platform Edition 9 リリースノート』、Part No. は 819-7011 です。

補足情報

Application Server 製品に関する便利な情報を、次のサイトから入手できます。

- [Java EE Feedback Submittal Form](http://java.sun.com/docs/forms/javaeesubmittalform.html)
(<http://java.sun.com/docs/forms/javaeesubmittalform.html>) — Sun Java System Application Server に関するフィードバックを送信します
- [J2EE-INTEREST List](http://archives.java.sun.com/archives/j2ee-interest.html) (<http://archives.java.sun.com/archives/j2ee-interest.html>)
— Java EE プラットフォームに関する質問用のメーリングリスト
- [Bug Database on Java Developer Connection](http://developer.java.sun.com/developer/bugParade/index.jshtml)
(<http://developer.java.sun.com/developer/bugParade/index.jshtml>) — Sun Java System Application Server に関連したバグの表示または送信をします
- [Glassfish Community](http://java.sun.com/javaee/glassfish) (<http://java.sun.com/javaee/glassfish>) — Java EE 5 プラットフォームでの Application Server やほかの最新機能の実装を開発する無償オープンソースプロジェクトコミュニティ
- [Java Enterprise Community](http://community.java.net/java-enterprise/) (<http://community.java.net/java-enterprise/>) — Java EE アプリケーション、コンポーネント、またはツールを中心にした java.net (<http://java.net/>) の一部
- [EJB 3.0 Tech Tips](http://java.sun.com/developer/EJTechTips/) (<http://java.sun.com/developer/EJTechTips/>) — [java.sun.com](http://java.sun.com/developer/) (<http://java.sun.com/developer/>) の開発者領域にある EJB 3.0 の有用な技術的ヒント。具体的には次のとおりです。
 - [Accessing the Bean Environment in EJB Session Beans](http://java.sun.com/developer/EJTechTips/2005/tt0930.html#1)
(<http://java.sun.com/developer/EJTechTips/2005/tt0930.html#1>)

- [Converting a POJO to a Persistent Entity](http://java.sun.com/developer/EJTechTips/2005/tt1122.html#2)
(<http://java.sun.com/developer/EJTechTips/2005/tt1122.html#2>)
- [Sun Developer Forums](http://developers.sun.com/forums/) (<http://developers.sun.com/forums/>) — 技術およびプログラミング技術に関する知識と問題を共有するための対話型掲示板。Application Server に関連した議論には、[Application & Integration Server](http://swforum.sun.com/jive/category.jspa?categoryID=10) (<http://swforum.sun.com/jive/category.jspa?categoryID=10>) フォーラムを使用します。
- [Java Technology Forums](http://forum.java.sun.com/index.jspa) (<http://forum.java.sun.com/index.jspa>) — 一般的な Java テクノロジに関する情報を共有するための対話型掲示板
- [Java EE SDK Forums](http://forum.java.sun.com/forum.jspa?forumID=136) (<http://forum.java.sun.com/forum.jspa?forumID=136>) — 特に Java EE SDK に関する情報を共有するための対話型掲示板
- [Sun Software Support Service Standard](http://www.sun.com/service/support/software/standard.html)
(<http://www.sun.com/service/support/software/standard.html>) — 購入の契約書を入手できます
- [Sun Developer Network](http://developers.sun.com/user_registration/whyregister.html)
(http://developers.sun.com/user_registration/whyregister.html) — 参加すれば、さらなる支援を受け、最新の Java 開発に精通できます
- docs.sun.com (<http://docs.sun.com/>) — Application Server およびほかの Sun 製品のすべてのマニュアル
- [Sun Java System Application Server home page](http://www.sun.com/appserver) (<http://www.sun.com/appserver>) — Application Server に関するデータシート、プレス情報、ニュース、およびその他の情報

Application Server Platform Edition 9 について

Sun Java™ System **Application Server Platform Edition 9** は、Java EE アプリケーションおよび Java テクノロジベースの Web サービスの開発と配備に利用できる Java EE 5 プラットフォーム互換サーバーです。

この節では、次の点を説明します。

- 13 ページの「9 リリースの新機能」
- 16 ページの「ハードウェアとソフトウェアの要件」
- 23 ページの「Java EE 5 プラットフォームの API」
- 25 ページの「Java EE 5 SDK」
- 26 ページの「サポートされている別の Java バージョンへの切り替え」

9 リリースの新機能

Sun Java System Application Server Platform Edition 9 は、次に示す多くの新機能を実装しています。

- **Java EE 5** プラットフォームのサポート — Sun Java System Application Server Platform Edition 9 では、Java EE 5 仕様を実装して、次世代のエンタープライズアプリケーションおよび Web サービスにもっとも適したアプリケーションランタイムの 1 つを実現します。Application Server 9 は、次の Java EE 標準を実装しています。
 - 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

一連のすべての Java EE 5 プラットフォームテクノロジーについて、このノートの [23 ページの「Java EE 5 プラットフォームの API」](#) で後述します。

- 開発者の生産性の向上 — Application Server 9 は、オープンモジュール方式の拡張可能なアーキテクチャーと、軽量でメモリー占有率が低くパフォーマンスが高いランタイムにより、開発者の操作性を高めます。アプリケーション配備の迅速化により、反復的なアプリケーション開発をすばやく簡単に行えます。製品を一般的な IDE に簡単に統合できます。NetBeans と Sun Java Studio Creator が Application Server 9 と統合し、優れた開発環境を実現します。
- **Java 2 Standard Edition 5.0** のサポート — Application Server 9 は、管理および監視のための拡張機能とパフォーマンスおよびスケラビリティについての改善点を数多く含む [Java 2 Standard Edition 5.0 \(http://java.sun.com/j2se/1.5.0/\)](http://java.sun.com/j2se/1.5.0/) をサポートしています。
- 高度な管理 — Application Server 9 では、ブラウザベースの管理コンソールとスクリプト処理可能なコマンド行インタフェースとを使用して、セキュリティ保護されたりリモート管理を実現します。新しい管理コンソール機能を次に示します。
 - Web サービスの検出、管理、監視、プロビジョニングなど、総合的な Web サービス管理サポート。
 - アプリケーションの監視、視覚化、および診断の強化。
 - JMX を使用したセキュリティ保護されたプログラムによるリモート管理および監視を実現するための JMX MBean API。
 - プログラムによって拡張可能な管理ルールおよびトリガーの組み込み。
- プラットフォームのサポート — Application Server 9 は、幅広いオペレーティングシステム、データベース、Web サーバー、ロケール、およびハードウェアをサポートします。すべてのリストについては、このノートで後述する [16 ページの「プラットフォーム要件」](#) を参照してください。
- **64 ビットサポート** — Application Server 9 では、64 ビット版 Solaris オペレーティングシステムのサポートが追加されました。
- **Java Web Services Developer Pack** との統合 — Application Server 9 は、Web Services Metadata 1.0 (Web サービスの注釈を指定)、JAX-WS 2.0 (Java プラットフォーム用の Web サービス API を指定)、JAXB 2.0 (Java と XML のバイnding を指定) などの最新の Web サービス標準をサポートするために、[Java Web Services Developer Pack \(http://java.sun.com/webservices/downloads/webservicespack.html\)](http://java.sun.com/webservices/downloads/webservicespack.html) (Java WSDP) 2.0 と統合されました。さらに、Java Business Integration (JBI) 標準に基づいた Java EE サービスエンジンを組み込んでいます。
- **JMS 接続** — Application Server 9 は、IBM MQ シリーズおよび Sun Java System Message Queue Server との接続を内蔵しています。
- **長期継続トランザクション** — Application Server 9 では、長期継続の分散トランザクションのサポートが追加されました。

- **JDBC™** ドライバー Application Server 9 では、どのデータベースサーバーへの接続にも JDBC ドライバを使用します。Java EE 互換のデータベースを構成するコンポーネントとして Sun がテストし基準を満たすと判断したもののリストについては、このノートで後述する 17 ページの「**JDBC ドライバとデータベース**」を参照してください。Application Server には、配備に使用できる主要なデータベース用の DataDirect™ JDBC ドライバを含みます。
- **Java DB** データベース Application Server 9 は、**Apache Derby** データベースに基づいた Java DB データベースをバンドルしており、エンドツーエンドの Java EE アプリケーションの開発と配備を可能にします。
- **呼び出しフロー監視** Application Server 9 は、着信要求がアプリケーションサーバー内のさまざまなコンテナおよびユーザーアプリケーションコードを通過する際に、それを監視するように設定できます。たとえば、Application Server は、Web コンテナ、Web アプリケーションコード、EJB コンテナ、および EJB アプリケーションコード内で経過する時間を分析できます。収集した情報はデータベース内に格納され、その後クエリーや分析に利用できます。
- **自己管理ルール** Application Server 9 は、アプリケーションサーバーの管理タスクを自動化するための、強力で柔軟なインフラストラクチャーを備えています。自己管理ルールは、イベントとアクションから構成されます。イベントには、メッセージログ記録、しきい値の監視、タイマー、JMX 通知などがあります。アクションとはユーザーが定義したロジックであり、これはその後 JMX MBean でカプセル化されます。イベントがトリガーされ、このイベントが自己管理ルールで定義されたいずれかのイベントに一致した場合、関連付けられたアクションが実行されます。たとえば、管理者は、SEVERE メッセージがアプリケーションサーバーのログに記録されたときに電子メールを受信するように管理ルールを定義できます。
- **移行ツールとアップグレードツール** 含まれているツールを使用して、Java EE アプリケーションの標準への適合性と移植性について確認できます。また、これらのツールは、ほかの Java EE アプリケーションサーバー (JBoss、WebLogic、WebSphere) からの移行と、以前のバージョンの Sun の Application Server からのアップグレードに役立ちます。これらの移行ツールは、<http://java.sun.com/j2ee/tools/migration/index.html> でも個別に入手できます。
- **オープンソースおよび GlassFish コミュニティー** 2005 年 6 月、Sun は、Java EE 5 プラットフォームの最新機能とそれに関連したエンタープライズテクノロジーを実装した、無償でオープンソースの商用レベルのアプリケーションサーバーを開発することを目標として、**GlassFish** (<http://java.sun.com/javaee/glassfish/>) コミュニティーを立ち上げました。Application Server Platform Edition 9 は、Sun の技術者と GlassFish コミュニティーが開発したソースコードを基盤に作成されています。

ハードウェアとソフトウェアの要件

ここでは、Sun Java System Application Server Platform Edition 9 製品をインストールする前に満たしている必要のある要件を示します。

- 16 ページの「プラットフォーム要件」
- 17 ページの「システムの仮想化のサポート」
- 17 ページの「重要パッチ情報」
- 17 ページの「JDBC ドライバとデータベース」
- 18 ページの「バンドル版 Java DB データベースの使用」
- 22 ページの「ブラウザ」
- 22 ページの「Sun Java System Application Server のアップグレード」
- 22 ページの「その他の要件」

プラットフォーム要件

次の表に、Sun Java System Application Server Platform Edition 9 製品と互換性があるオペレーティングシステムを示します。

表 2-1 サポートされているオペレーティングシステム

オペレーティングシステム	最小メモリー	推奨メモリー	最小ディスク容量	推奨ディスク容量	JVM
Sun Solaris 9、 10 (SPARC)	512M バイト	512M バイト	250M バイトの 空き領域	500M バイトの 空き領域	J2SE 5.0 Java SE 6 (利用 可能な場合)
Solaris 9、 10 (x86)					
64 ビット版 Sun Solaris 10 (SPARC、 x86)	512M バイト	512M バイト	250M バイトの 空き領域	500M バイトの 空き領域	J2SE 5.0 Java SE 6 (利用 可能な場合)
Redhat Enterprise Linux 3.0 U1、 4.0	512M バイト	1G バイト	250M バイトの 空き領域	500M バイトの 空き領域	J2SE 5.0 Java SE 6 (利用 可能な場合)
Windows Server 2000 SP4+	1G バイト	2G バイト	500M バイトの 空き領域	1G バイトの空 き領域	J2SE 5.0 Java SE 6 (利用 可能な場合)
Windows 2000 Advanced Server SP4+					
Windows Server 2003					
Windows XP Pro SP1+					

UNIX™ では、オペレーティングシステムのバージョンを `uname` コマンドで確認できます。ディスク容量は `df` コマンドで確認できます。

注 - Microsoft Windows プラットフォームで Application Server を実行している場合は、FAT や FAT32 ではなく NTFS ファイルシステムを使用する必要があります。

システムの仮想化のサポート

システムの仮想化は、複数のオペレーティングシステム (OS) インスタンスを共有ハードウェア上で個別に実行できるようにするテクノロジーです。機能的にいうと、仮想化された環境でホストされる OS に配備されたソフトウェアは、通常はベースとなるプラットフォームが仮想化されていることを認識しません。Sun では、精選されたシステムの仮想化と OS の組み合わせについて、その Sun Java System 製品のテストを行っています。これは、Sun Java System 製品が、適切な規模と構成の仮想化された環境で、仮想化されていないシステム上の場合と同様に引き続き機能することを実証するためのテストです。仮想化された環境での Sun Java System 製品のサポートについては、[System Virtualization Support in Sun Java System Products](#) を参照してください。

重要パッチ情報

現時点で Sun Java System Application Server Platform Edition 9 に必要なパッチについては、<http://sunsolve.sun.com> にアクセスして、「[パッチやアップデート](#) (<http://sunsolve.sun.com/pub-cgi/show.pl?target=patchpage>)」を選択してください。Sun Java System Application Server Platform Edition 9 のリンクをたどります。オペレーティングシステムのパッチ要件に変更があり、Java Enterprise System コンポーネントのパッチが利用可能になった場合、更新は SunSolve から入手できます。最初は、推奨パッチクラスタの形式で提供されます。

Solaris のパッチ要件

Solaris 9 または 10 (x86 または SPARC) を使用する場合には、「Sun 推奨パッチクラスタ」をインストールすることをお勧めします。このパッチクラスタは [SunSolve](#) (<http://sunsolve.sun.com/pub-cgi/show.pl?target=patchpage>) Web サイトの「[推奨 & セキュリティパッチ](#)」から入手できます。

JDBC ドライバとデータベース

表 2-2 に、Java EE の互換性要件を満たすデータベースとドライバを示します。サポートされるすべての Sun Java System Application Server Platform Edition の構成では、バンドル版 Java DB データベースとドライバなどの、この表に基づくデータベースとドライバの組み合わせを少なくとも 1 つ含める必要があります。さらに、Application Server は、対応する JDBC ドライバを使用してほかのどの DBMS に対する JDBC 接続もサポートするように設計されています。

表 2-2 Java EE 互換の JDBC ドライバ

JDBC ドライバベンダー	JDBC ドライバのタイプ	サポートされるデータベースサーバー
Java DB ネットワーククライアント	Type 4	Apache Derby 10.1
DataDirect 3.5	Type 4	Oracle 9i、10g

表 2-3 に、Sun が内部的にテストしたその他のデータベースとドライバをすべて示します。

表 2-3 サポートされるその他の JDBC ドライバとデータベース

JDBC ドライバベンダー	JDBC ドライバのタイプ	サポートされるデータベースサーバー
DataDirect 3.5	Type 4	Sybase ASE 12.5、15 Microsoft SQL Server 2000、2005 IBM DB2 8.1、8.2
IBM	Type 2	IBM DB2 8.1、8.2
Microsoft MS SQL	Type 4	SQLServer 2000、2005
MySQL Connector/J Driver 3.1	Type 4	MySQL 5
Oracle	Type 4	Oracle 9i、10g
Postgres	Type 4	Postgres 8.1
Sybase	Type 4	ASE 12.5、15

通常、Application Server Platform Edition 9 は、Java EE 仕様に適合するすべての JDBC ドライバをサポートします。

バンドル版 Java DB データベースの使用

この節では、Application Server 9 にバンドルされている Java DB データベースの実装を使用する手順について説明します。Java DB は、[Apache Derby データベース](#)を基盤としています。

- 18 ページの「[Java DB データベースの起動と停止](#)」
- 19 ページの「[Java DB ユーティリティスクリプト](#)」
- 20 ページの「[Pointbase から Java DB へのテーブルのエクスポート](#)」

Java DB データベースの起動と停止

Sun Java System Application Server 9 には、Java DB ネットワークサーバーの起動と停止を行うための 2 つの新しい `asadmin` コマンドが導入されています。

- Java DB Network Server のインスタンスを開始するには、`asadmin start-database` コマンドを使用します。

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

ホストのデフォルト値は `0.0.0.0` です。この値によって、Java DB は IP/ホスト名のインタフェースと同様に `localhost` で待機できます。 `dbhome` プロパティの値は、Java DB データベースの格納場所です。デフォルトの `path` は、`appserver_install_dir/javadb` です。

- Java DB Network Server の実行中のインスタンスを停止するには、次のように `asadmin stop-database` コマンドを使用します。

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

Java DB ユーティリティースクリプト

Application Server 9 とともに出荷される Java DB の構成には、Java DB の使用に役立つ便利なスクリプトも含まれます。 `<appserver_install_dir>/javadb/frameworks/NetworkServer/bin` ディレクトリには、次の使用可能なスクリプトがあります。

- `startNetworkServer.ksh/bat`: ネットワークサーバーを開始するスクリプト
- `stopNetworkServer.ksh/bat`: ネットワークサーバーを停止するスクリプト
- `ij.ksh/bat`: 対話式の JDBC スクリプト記述ツール
- `dblook.ksh/bat`: データベースの DDL の全部または一部を表示するスクリプト
- `sysinfo.ksh/bat`: Java DB 環境に関するバージョン情報を表示するスクリプト
- `NetworkServerControl.ksh/bat`: `NetworkServerControl` API でコマンドを実行するためのスクリプト

▼ Java DB ユーティリティースクリプトを実行するために環境を設定するには

- 1 `<appserver_install_dir>/javadb` ディレクトリをポイントするように、`DERBY_INSTALL` 環境変数を設定します。
- 2 `CLASSPATH` 環境変数の設定を解除します。
- 3 次のプロパティを設定することもできます。
 - a. `DERBY_SERVER_HOST` をネットワークサーバーの待機先ホストとして設定します。`0.0.0.0` に設定すると、すべてのリスナーを有効にできます。

- b. DERBY_SERVER_PORT をネットワークサーバーの待機先ポート番号として設定し
ます。

参照 これらのユーティリティーの詳細については、Derby のツール
(<http://db.apache.org/derby/docs/10.1/tools/>) および Admin
(<http://db.apache.org/derby/docs/10.1/adminguide/>) に関するガイドを参照してく
ださい。

Pointbase から Java DB へのテーブルのエクスポート

この例は、コマンドの Pointbase テーブルの DDL をキャプチャーし、NetBeans 5.0 を
使用して Java DB に同じテーブルを作成する方法を示しています。これと同じ事を行
うには、コマンドツールと unload database コマンドを使用する方法もあります。

```
./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 sampledб.sql;
SQL> unload database sampledб.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;
```

この例の `unload database` コマンドを実行した結果は、`sampledb.sql` ファイルに書き込まれます。`sampledb.sql` ファイルには、必要なテーブルとインデックスを作成するために必要となる DDL がすべて含まれています。また、データベースにデータを挿入しなおすための DML も含まれます。コマンドの `RUN` コマンドは、生成されたスクリプトを使用して別の Pointbase データベースにデータをインポートするために使用されます。生成されたファイル内の `INSERT` 文と関連するデータの例を次に示します。

```
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'
};
```

コマンドの `unload database` コマンドによって生成されたファイルを、DDL だけで構成されるように簡単に編集できます。たとえば、`insert` 文を処理するプログラムを記述することは難しくありません。簡単なテストとして、Pointbase `sample` データベースに対して `unload database` コマンドを使用し、生成されたスクリプトを編集して次のように変更します。

- すべての `CREATE Table` 文の末尾から `Organization Heap` という部分を削除します。
- `COMMIT` コマンドを削除します。
- プール値 `datatype` を `smallint` に変更します。
- すべての `INSERT` 文とそれに関連するデータを削除します。

次に、簡単な Ant スクリプトを使用し、`sql` ターゲットを使用して DDL を実行します。最後に、`sun-appserv-samples` データベースに対して同じ試行を繰り返します。このとき、生成された SQL ファイルに次の変更を追加する必要があります。

- サンプルのデータベースに対して、ここで説明した変更をすべて加えます。
- create user コマンドを削除します。
- SET PATH コマンドを削除します。
- Decimal の精度を 38 から 31 以下に変更します。
- float の精度を 64 から 52 以下に変更します。
- CREATE PROCEDURE コマンドの SPECIFIC キーワードは現在サポートされていません。
- GRANT コマンドを削除します。

Pointbase の Java プロシージャを Java DB で動作するように変換するには、CREATE PROCEDURE 文と同様に Java コードも変更する必要があります。Java DB の Java プロシージャの詳細については、[Derby Reference manual](http://db.apache.org/derby/docs/10.1/ref/) (<http://db.apache.org/derby/docs/10.1/ref/>) を参照してください。Boolean データ型は、次期リリースの Java DB でサポートされる予定です。

ブラウザ

この節では、Sun Java System Application Server Platform Edition 9 管理コンソールおよび『Quick Start Guide』でサポートされるブラウザを示します。Application Server 上のアプリケーションの実行時にサポートされるブラウザは、実行中のアプリケーションによって異なります。

表 2-4 サポートされるブラウザ

ブラウザ	バージョン
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

Sun Java System Application Server のアップグレード

以前のバージョンの Application Server から Sun Java System Application Server Platform Edition 9 にアップグレードするための詳細な手順については、『[Sun Java System Application Server Platform Edition 9 Upgrade and Migration Guide](#)』を参照してください。

その他の要件

Sun Java System Application Server ソフトウェアをインストールする前に、次の追加の要件が満たされるようにしてください。

- 空き容量: 一時ディレクトリに、Sun Java System Application Server のインストール用に少なくとも 200M バイトの空き領域と、SDK のインストール用に 250M バイトの空き領域が必要です。

- アンインストーラの使用: システムから Application Server を削除する場合、ソフトウェアに含まれるアンインストーラを使用することが重要です。他の方法を使って削除すると、同じバージョンの再インストールや新しいバージョンのインストールを行うときに問題が発生する可能性があります。
- 使用可能なポート: 未使用のポートが7つ必要です。
 - インストールプログラムは、使用されているポートを自動的に検出し、その時点で未使用のポートをデフォルトの設定として示します。デフォルトでは、HTTP については 8080、HTTPS については 8181、Administration Server については 4848 が初期デフォルトポートです。
 - インストールプログラムは、使用されているポートを検出し、それ以外の2つのポートを割り当てます。1つは Sun Java™ System Message Queue (デフォルトで 7676)、もう1つは IIOP (デフォルトで、IIOP については 3700 および IIOP/SSL については 3820 と 3890) です。これらのデフォルトポート番号が使用されている場合、インストールプログラムは動的ポートの範囲から無作為にポート番号を割り当てようとします (これは利用可能な次のポート番号とはかぎらない点に注意)。

インストール済みサーバーの起動 (UNIX): インストール済みサーバーを置き換える場合以外は、Sun Java System Application Server 9 のインストール処理を開始する前に、インストール済みサーバーを起動してください。インストールプログラムが使用中のポートを検出するため、使用中のポートがほかの目的に使用されることはありません。

- インストール済みサーバーの置き換え (UNIX): 古いバージョンの Sun Java System Application Server がインストールされており、それを現在のバージョンの Application Server に置き換える場合は、新しいサーバーをインストールする前に、インストール済みサーバーを停止してください。
- ファイアウォールのシャットダウン (Microsoft Windows) – Sun Java System Application Server ソフトウェアをインストールする前に、すべてのファイアウォールソフトウェアを停止する必要があります。一部のファイアウォールソフトウェアはデフォルトですべてのポートを無効にするからです。インストールプログラムは、使用可能なポートを正確に検出できる必要があります。

互換性の詳細については、『[Sun Java System Application Server Platform Edition 9 Upgrade and Migration Guide](#)』を参照してください。

Java EE 5 プラットフォームの API

Sun Java System Application Server Platform Edition 9 は、Java EE 5 プラットフォームをサポートしています。Java EE 5 プラットフォームで使用できる拡張 API を、次の表に示します。

表 2-5 Java EE 5 プラットフォームの主要な API の変更点

API	JSR
Java EE 5	
Java Platform, Enterprise Edition 5 (http://java.sun.com/javaee/5/)	JSR 244 (http://jcp.org/aboutJava/communityprocess/pr/jsr244/)
Web サービステクノロジー	
エンタープライズ Web サービスの実装	JSR 109 (http://jcp.org/en/jsr/detail?id=109)
Java API for XML-Based Web Services (JAX-WS) 2.0	JSR 224 (http://jcp.org/en/jsr/detail?id=224)
Java API for XML-Based RPC (JAX-RPC) 1.1	JSR 101 (http://jcp.org/en/jsr/detail?id=101)
Java Architecture for XML Binding (JAXB) 2.0 (http://java.sun.com/webservices/jaxb/)	JSR 222 (http://jcp.org/en/jsr/detail?id=222)
SOAP with Attachments API for Java (SAAJ)	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)
Java プラットフォーム用 Web サービスメタデータ	JSR 181 (http://jcp.org/en/jsr/detail?id=181)
コンポーネントモデルテクノロジー	
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)
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)
管理テクノロジー	
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)

表 2-5 Java EE 5 プラットフォームの主要な API の変更点 (続き)

API	JSR
Java Authorization Contract for Containers (http://java.sun.com/j2ee/javaacc/)	JSR 115 (http://jcp.org/en/jsr/detail?id=115)
その他の Java EE テクノロジー	
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 は、Java EE 5 SDK の一部として使用できます。

Java EE 5 SDK には、次の 3 つのバージョンがあります。

- Java EE 5 SDK (<http://java.sun.com/javaee/downloads/>)
- Java EE 5 SDK SOA Starter Kit Preview
(<http://java.sun.com/integration/openesb/releasenotes.jsp>)
- Java EE 5 Tools Preview Bundle
(<http://www.netbeans.info/downloads/download.php?type=5.5>)

Application Server 以外にも、Java EE 5 SDK には Java 2 Platform Standard Edition 5.0 Update 6、Java EE 5 サンプル、Java BluePrints、および API ドキュメント (Javadoc) が含まれます。

Java EE 5 SDK SOA Starter Kit Preview には、Java Based Integration (JBI) の JSR 208 仕様に基づいたエンタープライズサービスバス (ESB) ランタイムを実装した Project Open ESB Starter Kit も含まれています。BPEL サービスエンジン、Java EE サービスエンジン、HTTP SOAP バインディングコンポーネント、および BPEL サービスエンジンの使用例も含まれます。

Java EE 5 Tools Preview Bundle は、NetBeans 5.5 Beta、NetBeans Enterprise Pack 5.5、および Sun Java System Access Manager 7.1 を追加しますが、J2SE 5.0 は追加しません。

その他の SDK コンポーネントの詳細については、次のサイトを参照してください。

- [Project Open ESB Starter Kit](#)
- [NetBeans 5.5](#)
- [NetBeans Enterprise Pack 5.5](#)

サポートされている別の **Java** バージョンへの切り替え

Sun Java System Application Server 9 では、基盤となる JVM として J2SE 5.0 以上が必要です。Java を別のバージョンに切り替える場合は、次の一般的な手順に従います。
(Windows および Unix)

▼ サポートされている別の **Java** バージョンへ切り替えるには

- 1 必要に応じて、**Java SDK (JRE ではない)** をダウンロードしてインストールします。
Java SDK は、<http://java.sun.com/j2se> からダウンロードできます。
- 2 **Application Server** を完全に停止します。
次のコマンド行が使えます。

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


または、管理コンソールの GUI で次の手順に従います。
 - a. 「アプリケーションサーバー」ノードをクリックします。
 - b. 「インスタンスの停止」をクリックします。
- 3 `install_dir/config/asenv.conf` ファイル (**Windows** では `asenv.bat`) を編集して、新しい **J2SE** のホームディレクトリをポイントするように `AS_JAVA` の値を変更します。
- 4 `as-install/samples/common.properties` ファイルを編集して、**J2SE** の新しいホームディレクトリを参照するように `com.sun.aas.javaRoot...` で始まる行を変更します。
- 5 **Application Server** を再起動します。

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

既知の問題点と制限事項

この節では、Sun Java System Application Server Platform Edition 9 製品に関する既知の問題とそれに関連する回避策について説明します。問題の説明にプラットフォームが明記されていない場合、その問題はすべてのプラットフォームに当てはまります。この節は次の項目から構成されています。

- 27 ページの「管理」
- 29 ページの「アプリケーションクライアント」
- 30 ページの「インストール」
- 31 ページの「ライフサイクル管理」
- 32 ページの「Java Persistence API」
- 35 ページの「JavaServer Faces」
- 37 ページの「アップグレード」
- 39 ページの「Web コンテナ」
- 41 ページの「プラットフォーム固有の問題」

管理

domain1 が存在しない場合、package-appclient スクリプトが動作しない。(ID 6171458)

デフォルトでは、asenv.conf によってポイントされる domain1 の AS_ACC_CONFIG 変数のハードコードされた値が \$INSTALL/lib/package-appclient.xml にあります。domain1 を削除して新たなドメインを作成した場合、AS_ACC_CONFIG 変数は新たなドメイン名で更新されません。その結果、package-appclient スクリプトの処理が失敗します。

解決方法

次のいずれかの操作を行います。

- domain1 はそのままにしておき、その前後に別のドメインを作成します。

- domain1 を削除し、\$INSTALL/lib/package-appclient.xml の domain1 にハードコードされた値を新しいドメイン名で置き換えます。domain1 がない場合、新たなドメインが作成されるたびにこれを行う必要があります。

JMX エージェントを追加した Application Server の起動はサポートされていない。(ID 6200011)

J2SE 1.4.x、5.0、またはそれ以降のバージョンは、Application Server で設定できます。J2SE 5.0 プラットフォームの重要な特徴は、JMX エージェントを起動できることです。サーバーの起動時にシステムプロパティを明示的に設定すると、JMX エージェントがアクティブになります。

次に例を示します。

```
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"
```

JMX プロパティを設定してサーバーを起動すると、Application Server VM 内で新しい jmx-connector サーバーが起動します。この場合は、望ましくない副作用の 1 つとして、管理機能が悪影響を受け、Application Server の管理 GUI や CLI で予期しない結果が発生することがあります。問題は、組み込みの jmx-connector サーバーと新たな jmx-connector サーバーとの間で衝突が発生することにあります。

解決方法

jconsole または何らかの JMX 互換クライアントを使用する場合には、Application Server とともに起動する標準の JMX コネクタサーバーを再利用することを検討してください。

サーバーの起動時に、次に示すような行が server.log に作成されます。ここで指定されている JMXServiceURL に接続し、資格を正常に指定したあと、同様の管理および設定操作を実行することができます。次に例を示します。

```
[#|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.|#]
```

詳細については、『Sun Java System Application Server 9 管理ガイド』を参照してください。

display-error-statistics で負の数値が返される

管理コンソール GUI では、実際のゼロ (0) と統計が利用できない状態 (N/A) とを区別できます。現在、コマンド行インタフェースでは、N/A の代わりにマイナス 1 (-1) が表示されます。たとえば、次の display-error-statics コマンドの出力では、N/A の代わりに -1 が表示されています。

```
# asadmin display-error-statistics
Please enter the admin user name>admin
Please enter the admin password>
Timestamp                               Severity  Warning
-----
1143659837750(Mar 29, 2006 11:17:17 AM)    0         0
1143656237750(Mar 29, 2006 10:17:17 AM)    0         0
1143652637750(Mar 29, 2006 9:17:17 AM)     -1        -1
1143649037750(Mar 29, 2006 8:17:17 AM)     -1        -1
1143645437750(Mar 29, 2006 7:17:17 AM)     -1        -1
-----
Command display-error-statistics executed successfully.
```

解決方法

エラー統計値を表示するには、管理コンソール GUI を使用してください。この問題は、今後のリリースで解決されます。

アプリケーションクライアント

ここでは、アプリケーションクライアントに関する既知の問題とその解決方法を示します。

アプリケーションクライアントアーカイブのライブラリ JAR が **MANIFEST** ファイルを上書きする。(ID 6193556)

クライアント JAR (たとえば reporter.jar) 内に最上位レベルの JAR ファイルがある場合、クライアント JAR を配備すると、その JAR の MANIFEST ファイルがクライアント JAR の MANIFEST ファイルを上書きします。

解決方法

現時点ではありません。

Windows プラットフォーム - クラスパスに APPCPATH が設定されていない (ID 6419847)

Windows での実行時に、appclient.bat コマンドが APPCPATH を受け付けていないことがわかっています。この問題は、appclient.bat ファイルで APPCPATH 文を引用符で囲むと確認できます。たとえば、次の文は失敗します。

```
set CLASSPATH="%JAXP_IMPL_JARS%";%JVM_CLASSPATH%
if not %APPCPATH%x == x set CLASSPATH=%CLASSPATH%;"%APPCPATH%"
```

これに対し、次の文は正しく機能します。

```
set CLASSPATH=%JAXP_IMPL_JARS%;%JVM_CLASSPATH%
if not %APPCPATH%x == x set CLASSPATH=%CLASSPATH%;%APPCPATH%
```

解決方法

APPCPATH (および JAXP_IMPL_JARS) への参照を囲む二重引用符を削除してください。

インストール

この節では、インストールおよびアンインストールに関する既知の問題とその解決方法を示します。

一部の Linux システムで、「完了」ボタンをクリックしたあとにインストールの終了でハングアップする。(5009728)

この問題は、いくつかの Linux システム上で発生していました。これは Java Desktop System 2 でもっとも一般的に見られますが、RedHat ディストリビューションでも見られます。

インストーラの最後の画面で「完了」ボタンをクリックすると、インストーラは製品の「バージョン情報」ページまたは製品登録ページを表示するブラウザウィンドウの起動に失敗し、コマンドプロンプトに戻ることなくハングアップしたままになります。

解決方法

インストーラを起動した端末ウィンドウで Ctrl+C を押してインストーラを終了します。そのあとで、製品の「バージョン情報」ページまたは登録ページを表示するブラウザウィンドウが起動することがあります。ブラウザウィンドウが現れない場合には、ブラウザを起動してから次の URL を入力して「バージョン情報」ページを確認してください。

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

製品を登録するインストールオプションを選択した場合には、「バージョン情報」ページ上の登録ページへのリンクをたどってください。

Linux 上のインストーララッパーで J2SE の検出およびブートストラップに問題が発生することがある。(6172980)

Linux インストーラを起動する setup 実行可能ファイルがハングすることがあります。J2SE の場所を解決してインストーラウィザードを起動せずに、ラッパーがハングし、次のメッセージを返します。

```

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

```

この問題は Linux の一部のバージョンで見られ、特に JAVA_HOME 変数が存在する場合など環境設定に依存するようです。

解決方法

この問題を回避するには、次の手順を実行します。

▼ Linux 上のブートストラップの問題を回避するには

- 1 シェルに応じて unset または unsetenv を実行して、JAVA_HOME 変数の設定を解除します。
- 2 -javahome オプションを使用して setup を実行して、インストーラが使用する JAVA_HOME を指定します。

ライフサイクル管理

ここでは、ライフサイクル管理に関する既知の問題とその解決方法を示します。

ejb-timer-service プロパティ `minimum-delivery-interval` を `9000` に設定したあとで、ejb-timer-service プロパティ `redelivery-interval-in-millis` を `7000` に設定しようとすると、set コマンドが次のエラーで失敗する。(ID 6193449)

```

[echo] Doing admin task set
[exec] [Attribute(id=redelivery-interval-interval-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` は、同一の周期タイマーの最小発生間隔。
- `redelivery-interval-in-millis` は、失敗した `ejbTimeout` のあとに再発生を試みるまでタイマーサービスが待機する時間。

これは、再発生間隔のプロパティを最小発生間隔のプロパティと関連付けるロジックが間違っていて、GUI または CLI を使用して再発生間隔よりも最小発生間隔が大きくなるような値を設定できないという問題です。

`minimum-delivery-interval-in-millis` を、`ejb-timer-service` プロパティの `redelivery-interval-in-millis` 以上の値に設定する必要があります。問題は、`redelivery-interval-in-millis` の値が `minimum-delivery-interval-in-millis` の値よりも大きいことを検証するための Application Server の処理にエラーがあることです。

解決方法

次のように、これらプロパティのデフォルト値を使用します。

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

これらデフォルト以外の値を指定するとエラーが発生します。

Java Persistence API

この節では、Java Persistence API に関する既知の問題とその解決方法を示します。

WHERE 句でサブクエリーを使用する UPDATE または DELETE クエリーを行うと、クエリーのコンパイル中に `NullPointerException` が発生する。

詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=572 を参照してください。

解決方法

この種のクエリーは使用しないでください。

クエリーコンパイラが、**Java Persistence** 言語仕様で定義されたすべてのルールをチェックしない。

クエリーコンパイラは、仕様の Java Persistence 言語の章で定義されたすべてのルールをチェックするわけではありません。具体的には、次のチェックは行われません。

- 関数引数の型の互換性。
- 算術、論理、および比較演算子のオペランドの型の互換性。
- ORDER BY クエリーの SELECT 句の要件。

無効なクエリーは、コンパイルできても実行時に `SQLException` が発生する可能性があります。また、配下のデータベースでは制約が少なく、生成された SQL がサポートされても、別のデータベースで同じクエリーを実行すると `SQLException` が発生して失敗する場合があります。

解決方法

上の条件を手動で確認してください。

JOIN 変数別にグループ化し、JOIN 変数を直接選択するクエリーを行うと、`SQLException` が発生する。

JOIN 変数別にグループ化し、JOIN 変数を直接選択するクエリーを行うと、グループ化していない式が選択されていることを知らせる `SQLException` が発生する可能性があります。詳細について

は、https://glassfish.dev.java.net/issues/show_bug.cgi?id=197 を参照してください。次に例を示します。


```
SELECT AVG(o.totalPrice), c FROM Order o JOIN o.customer c GROUP BY c
```

解決方法

JOIN 変数を定義するのではなく、たとえば次のように直接関係を結びます。

```
SELECT AVG(o.totalPrice), o.customer FROM Order GROUP BY o.customer
```

クエリーコンパイラが、間違っただトークンを示した紛らわしいエラーメッセージをスローする場合があります。

詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=550 を参照してください。

解決方法

クエリー内の間違いを手動で確認してください。

現在、同じクラスを含む2つの持続性ユニットを、同一のEARファイルに配備できない。

解決方法

異なるクラス名を使用してください。

persistence.xml のクラス要素を使用して、MappedSuperclass を明示的に一覧表示できない。

詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=558 を参照してください。

解決方法

MappedSuperclass の明示的な一覧表示はしないでください。

継承されたクラスで、@ManyToMany のエンティティーマッピングが失敗する。

詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=578 を参照してください。

解決方法

サブクラスでは ManyToMany 関係を使用しないでください。

関係に List<> を使用すると、MetadataHelper.getAttributeNameFromMethodName で StringIndexOutOfBoundsException が発生する。

詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=557 を参照してください。

解決方法

代わりに `java.util.Collection` を使用してください。

Null のデータベース値をプリミティブにマップできない。

解決方法

Null にできるデータベース列にマップするには、Java ラッパータイプを使用してください。

クライアント側から LAZY 初期化関係にアクセスすると、`NullPointerException` が発生する場合があります。

詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=404 を参照してください。

解決方法

クライアントヘインスタンスを返す前に、サーバー側コードで LAZY 初期化関係にアクセスしてください。

関係フィールドを選択するクエリーを行った場合、クエリー結果に **Null** が含まれない。

関係フィールドを選択するクエリーを行った場合、関係フィールドの値が Null のときにクエリー結果に Null が含まれません。このエントリはたとえば次のように、クエリー結果から省略されます。

```
SELECT o.customer FROM Order o WHERE ...
```

詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=637 を参照してください。

解決方法

関連するインスタンスから状態フィールドを選択してください。

```
SELECT o.customer.customerId FROM Order o WHERE ...
```

単一値の関係フィールドに対して定義された JOIN 識別変数を選択するクエリーを行うと、無効な SQL になる場合があります。

次に例を示します。

```
SELECT c FROM Order o LEFT OUTER JOIN o.customer c
```

詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=638 を参照してください。

解決方法

関連するインスタンスから状態フィールドを選択してください。

```
SELECT c.customerId FROM Order o LEFT OUTER JOIN o.customer c
```

EntityManager.find() を行うと、誤って IllegalArgumentException が発生する。

主キークラスが @IdClass 注釈で定義されている場合、EntityManager.find() を行うと、別のエントリのサブクラスであるエンティティに対して IllegalArgumentException が誤って発生します。詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=595 を参照してください。

解決方法

find() メソッドへの引数として、最上位のスーパークラスのクラスを使用し、その結果をサブクラスへキャストしてください。

関係フィールドを java.util.HashSet に設定したエンティティを継続できない。

詳細については、https://glassfish.dev.java.net/issues/show_bug.cgi?id=643 を参照してください。

解決方法

初期値として java.util.ArrayList を使用してください。

JavaServer Faces

ここでは、JavaServer Faces (JSF) に関する既知の問題とその解決方法を示します。JSF プロジェクト (<https://javaserverfaces.dev.java.net/>) の Web サイトにアクセスすると、JSF プロジェクトの追加情報を確認したり、発生した JSF のバグを報告できます。

JSF 1.1-1.2 仕様違反: 変数およびプロパティリゾルバ (ID 6419278)

VariableResolver 装飾を使用して Expression Language の機能を拡張する JavaServer Faces テクノロジアプリケーションが、正しく機能しない場合があります。

JavaServer Faces 技術仕様 (<http://jcp.org/en/jsr/detail?id=252>) の第 10.4.5 項には、次のように記述されています。

「デフォルトの PropertyResolver、VariableResolver、ActionListener、NavigationHandler、ViewHandler、または StateManager の置換値を与えると、デコレータデザインパターンが活用されます。この結果、該当するタイプの単一の引数をとるコンストラクタを与えた場合に、カスタム実装は、以前にロールを実行していた実装への参照を受け取ります。このようにして、カスタム実装は、機能のサブセットだけを変更し、または一部の追加機能だけを提供して、残りを既存の実装に委任することができます。」

Application Server 9では、VariableResolverのカスタム実装は、変数解決のロールを十分には実行していない「以前」のVariableResolverを受け取ります。

解決方法

式の解決を「以前」のVariableResolverに委任するのではなく、ValueExpressionを作成して、これを評価することをお勧めします。

例 3-1 ValueExpression の評価

```
public class CustomVR extends VariableResolver {

    private VariableResolver previous = null;

    public CustomVR(VariableResolver previous) {
        this.previous = previous;
    }

    public Object resolveVariable(FacesContext context, String name)
        throws EvaluationException {
        Object result = null;

        // Do some action that may resolve the variable.  If not, you
        // may be tempted to simply do:

        // result = previous.resolveVariable(context, name);

        // But this would not work due to bug 6419278.  A fix is
        // available, please see the Release Notes.  However, a
        // workaround is the following.

        ValueExpression ve = context.getApplication().getExpressionFactory().
            \
            createValueExpression(context.getELContext(), "#{ " + name + " }", Object.class);
        try {
            result = ve.getValue(context.getELContext());
        }
        catch (PropertyNotFoundException pnfe) {
            throw new EvaluationException(pnfe);
        }
        catch (ELException ele) {
            throw new EvaluationException(ele);
        }
        return result;
    }
}
```

アップグレード

この節では、アップグレードとアップグレードツールの使用に関する既知の問題とその解決方法について説明します。

Application Server Platform Edition 8 から Application Server Platform Edition 9 へのアップグレードで、*install_dir* /domains ディレクトリ以外のカスタムパスに作成されたドメインが直接アップグレードされない。(ID 6165528)

アップグレードユーティリティを実行しているときに、*install_dir* をソースインストールディレクトリとして指定すると、そのアップグレードプロセスは、*install_dir* /domains ディレクトリの下に作成されたドメインだけをアップグレードします。その他の場所に作成されたドメインはアップグレードされません。

解決方法

アップグレードプロセスを起動する前に、すべてのドメインディレクトリを、それぞれの場所から *install_dir* /domains ディレクトリに移動します。

一部の Linux システムで「アップグレードウィザードを起動」ボタンをクリックしたあとに、「同じ場所にアップグレード」を実行するインストーラがアップグレードツールの起動に失敗する。(6207337)

この問題はさまざまな Linux システムで発生しています。Java Desktop System 2 でもっとも一般的ですが、RedHat ディストリビューションでも発生しています。

インストーラの最後の画面で「アップグレードツールの起動」ボタンをクリックすると、インストーラはアップグレード処理を完了するためのアップグレードツールの起動に失敗し、コマンドプロンプトに戻ることなくハングアップしたままになります。

解決方法

この問題は、コマンド行インストールモードを使って代替アップグレードを実行している場合には発生しません。

▼ コマンド行インストールモードを使用するには

- 1 GUI モードで代替アップグレードを実行してこの問題が発生した場合には、インストールプログラムを起動した端末ウィンドウで **Ctrl+C** を押すことにより、そのインストールプログラムを終了します。
- 2 その端末ウィンドウから次のコマンドを使ってアップグレードツールを起動します。

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

`adminuser` および `adminpassword` は、アップグレード中のインストールで使用されている値に一致する必要があります。

- アップグレードツールがアップグレードプロセスを完了したあとは、ブラウザを起動して次の URL を入力することにより、「バージョン情報」ページを参照できません。

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

製品を登録するインストールオプションを選択した場合には、「バージョン情報」ページ上の登録ページへのリンクをたどってください。

アップグレードツールで、JVM オプションの一部が移行されない (ID 6409650)

アップグレードツールでは、スタックサイズとヒープサイズに関連した JVM オプションなどは、インストール中に指定された値を保持すべきであるので移行しません。これは、これらのオプションの値が、現バージョンの製品に対する特定の理由のために設定されることが多いからです。アップグレードツールは、どのオプションが移行されなかったかを示すメッセージを記録します。

アップグレード後、静的ドキュメントを docroot ディレクトリにコピーする必要がある (ID 6409641)

Application Server インストールをアップグレードしたら、docroot ディレクトリ内のすべての静的ドキュメントを新しい docroot ディレクトリにコピーしてください。同じ場所でアップグレードを行った場合は、インストール時の `domaindir` の下にあるバックアップされたドメインディレクトリからドキュメントをコピーしてください。

Windows 上でアップグレードツールを実行するときに、Application Server のファイルを閉じる必要がある

Windows 上でアップグレードツールを実行している間は、アップグレードするインストールですべてのファイルを閉じてください。ファイルを開いておくと、アップグレードツールがハングアップすることが報告されています。

XWSClientProvider および XWSServerProvider を使用した場合、ユーザー名トークンが暗号化されない (ID 6409652)

アップグレード後の Application Server で XWSClientProvider および XWSServerProvider を使用するには、これらのプロバイダを使用するアプリケーションの配備記述子を、XWSClientProvider および XWSServerProvider を参照するように変更する必要があります。

変更する必要がある配備記述子は、`sun-web.xml` と `sun-ejb-jar.xml` です。アップグレードツールは、`sun-web.xml` または `sun-ejb-jar.xml` に ClientProvider および ServerProvider を見つけると、アップグレードログにメッセージを記録します。

Web コンテナ

ここでは、Web コンテナに関する既知の問題とその解決方法を示します。

空の `<load-on-startup/>` 要素を持つ **Servlet 2.4** ベースの `web.xml` を使用して **WAR** を配備できない。 (**ID 6172006**)

`web.xml` のオプションの `load-on-startup` 要素は、サーブレットを宣言する Web アプリケーションの起動の一環として、そのサーブレットをロードおよび初期化すべきことを示します。

注 - この制限は、Servlet 2.4 ベースの `web.xml` だけに適用されます。Servlet 2.5 ベースの `web.xml` を使用して、空の `load-on-startup` 要素を指定することは可能です。

この要素のオプションの内容は、Web アプリケーションのその他のサーブレットとの関係で、そのサーブレットをロードおよび初期化する順序を示す整数です。空の `<load-on-startup/>` は、そのサーブレットを含む Web アプリケーションの起動時にそのサーブレットがロードおよび初期化される場合、その順序は意味を持たないことを表します。

`web.xml` の Servlet 2.4 スキーマでは、空の `<load-on-startup/>` はサポートされなくなりました。つまり、Servlet 2.4 ベースの `web.xml` を使用する場合は整数値を指定する必要があります。`<load-on-startup/>` の場合と同様に、空の `<load-on-startup/>` を指定すると、`web.xml` が `web.xml` のサーブレット 2.4 スキーマに対する妥当性検証に失敗するため、Web アプリケーションの配備も失敗します。

下位互換性の問題もあります。空の `<load-on-startup/>` は、Servlet 2.3 ベースの `web.xml` では有効です。

解決方法

Servlet 2.4 ベースの `web.xml` を使用する場合は、`<load-on-startup/>0</load-on-startup/>` を指定して、サーブレットの読み込み順序が問題にならないことを示します。

リソースに制約のあるサーバー上で JSP ページをコンパイルできない。 (**ID 6184122**)

JSP ページにアクセスしてもコンパイルに失敗し、サーバーログには「Unable to execute command」というエラーメッセージと次のスタックトレースが記録されます。

```
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)
```

解決方法

JSP のコンパイルスイッチを `fork` から `false` に設定します。

これは、次のいずれかの方法で行えます。

- グローバルに行うには、次のように、
`${S1AS_HOME}/domains/domain1/config/default-web.xml` 内の `JspServlet` の `fork` `init` パラメータを `false` に設定します。

```
<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>
```

- Web アプリケーションごとに、`sun-web.xml` の JSP 設定プロパティ `fork` を `false` に設定します。次のようにします。

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

これらのいずれかを設定することにより、`ant` が `javac` コンパイルのための新規プロセスを生成することが防止されます。

ParserUtils が Web アプリケーションからのパーサーを選択する (ID 6412405)

JSP にアクセスすると、特定の状況下で次の例外が記録されます。

```
java.lang.AbstractMethodError: org.apache.xerces.dom.DeferredDocumentImpl.  
setDocumentURI(Ljava/lang/String;)V
```

解決方法

次のシステムプロパティを `domain.xml` ファイルに追加してください。

```
<jvm-options>-Djavax.xml.parsers.DocumentBuilderFactory=com.sun.org.apache.xerces.  
internal.jaxp.DocumentBuilderFactoryImpl</jvm-options>
```

または、`org.apache.xerces` パッケージを含む JAR ファイルを、Web アプリケーションの `WEB-INF/lib` から削除します。

EARバンドルJARにTLDがない。(Glassfishの問題590)

EARファイルにバンドルされたJARパックJSPタグライブラリは、EARの一部であるWARファイルのJSPページから参照できません。WARファイルのMETA-INF/MANIFEST.MFリソースが、そのClass-Pathマニフェスト属性でタグライブラリJARファイルをリストしている場合でも参照できません。

EARファイルのJARファイルにパッケージされたタグライブラリをインポートするJSPページにアクセスすると、次のようなエラーメッセージがサーバーログに表示され、JSPページのコンパイルが失敗します。

```
The absolute uri: <taglib_uri> cannot be resolved in either web.xml or
the jar files deployed with this application
```

解決方法

すべてのJSPタグライブラリJARファイルを、これらのタグライブラリをインポートするJSPページを持つすべてのWARファイルのWEB-INF/libディレクトリにバンドルしてください。

このバグに関する詳細について

は、https://glassfish.dev.java.net/issues/show_bug.cgi?id=590 のGlassfishのサイトを参照してください。

プラットフォーム固有の問題**Windows XP および Windows Server 2003 で denyRemoteHost および allowRemoteHost プロパティを設定するときに、完全修飾されたドメイン名を使用する必要がある (ID 6377272)**

Windows システムでは、hostname や ipconfig /all などの Windows コマンドや、Java API InetAddress.getLocalHost().getHostName() が簡易ホスト名を返す場合でも、IP ベースの要求フィルタリングプロパティの設定時には、Windows ホストに対し簡易名 (単なる myhost など) ではなく、完全修飾されたドメイン名 (myhost.mydomain.com など) を使用する必要があります。

たとえば、次のコマンドは、Windows ホスト foobar によるアクセスを拒否します。

```
asadmin --user adminuser --password adminpasswd
--echo server.http-service.virtual-server.server.property.denyRemoteHost=foobar.sun.com
```

ホスト名は、DNS ドメイン名 foobar.sun.com で完全修飾されています。

