



# Sun GlassFish Enterprise Server 2.1 リリースノート



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

Part No: 820-7428-11  
2009 年 1 月

Sun Microsystems, Inc. (以下米国 Sun Microsystems 社とします) は、本書に記述されている製品に含まれる技術に関連する知的財産権を所有します。特に、この知的財産権はひとつかそれ以上の米国における特許、あるいは米国およびその他の国において申請中の特許を含んでいることがあります。また、それらに限定されるものではありません。

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.

この配布には、第三者によって開発された素材を含んでいることがあります。

本製品の一部は、カリフォルニア大学からライセンスされている Berkeley BSD システムに基づいていることがあります。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他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、Java および Solaris は、米国およびその他の国における米国 Sun Microsystems 社の商標、登録商標もしくは、サービスマークです。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に基づくものです。

OPEN LOOK および Sun<sup>TM</sup> Graphical User Interface は、米国 Sun Microsystems 社が自社のユーザ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・ユーザインタ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 のグラフィカル・ユーザインタフェースを実装するか、またはその他の方法で米国 Sun Microsystems 社との書面によるライセンス契約を遵守する、米国 Sun Microsystems 社のライセンス実施権者にも適用されます。

本書で言及されている製品や含まれている情報は、米国輸出規制法で規制されるものであり、その他の国の輸出入に関する法律の対象となる場合があります。核、ミサイル、化学あるいは生物兵器、原子力の海洋輸送手段への使用は、直接および間接を問わず厳しく禁止されています。米国が禁輸の対象としている国や、限定はされませんが、取引禁止顧客や特別指定国民のリストを含む米国輸出排除リストで指定されているものへの輸出および再輸出は厳しく禁止されています。

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。

# ◆ ◆ ◆ 第 1 章

## 概要

---

Sun GlassFish™ Enterprise Server 製品は、Java™ Platform, Enterprise Edition (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のための 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に使用する場合も費用は発生しません。Sun GlassFish Enterprise Server は開発、配備、および再頒布の用途に無償で使用できます。再頒布を予定している場合は、再頒布ライセンスについて [Sun OEM 販売](#) にお問い合わせください。

Sun GlassFish Enterprise Server は、Web アプリケーションと Web サービスの開発および配信を行うための、Java Platform, Enterprise Edition (Java EE) テクノロジーに基づいた、使いやすく高速な、業界を代表するアプリケーションサーバーです。ソフトウェア障害やハードウェア障害が発生した場合にも稼働し続けるスケーラブルなキャリアアップグレードのサービスを実現するために、優れたパフォーマンス、クラスタリング、および高可用性機能を提供します。

- 3 ページの「このリリースノートについて」
- 5 ページの「アクセシビリティ機能」
- 4 ページの「関連マニュアル」
- 5 ページの「問題の報告とフィードバックの方法」
- 6 ページの「このマニュアルに関するコメント」
- 6 ページの「その他の情報」

### 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 GlassFish Enterprise Server リリース時点における重要な情報が記載されています。拡張機能、既知の問題、およびその他の最新の情報が記載されています。Enterprise Server 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版は、Enterprise Server [ドキュメント Web サイト](#) (<http://docs.sun.com/coll/1343.6>) から入手できます。ソフトウェアをインストールおよび設定する前だけでなく、それ以降も定期的にこの Web サイトをチェックして、最新のリリースノートと製品マニュアルを確認してください。

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。

## 関連マニュアル

表 1-1 Enterprise Server のマニュアルセットの内容

マニュアル名	説明
『Documentation Center』	タスクや主題ごとに整理された Enterprise Server のマニュアルのトピック。
『リリースノート』	ソフトウェアとマニュアルに関する最新情報。サポートされているハードウェア、オペレーティングシステム、Java™ Development Kit (JDK™)、およびデータベースドライバの包括的な表ベースの概要を含みます。
『クイックスタートガイド』	Enterprise Server 製品の使用を開始するための手順。
『Installation Guide』	ソフトウェアとそのコンポーネントのインストール。
『アプリケーション配備ガイド』	アプリケーションおよびアプリケーションコンポーネントの Enterprise Server への配備。配備記述子に関する情報を含みます。
『開発者ガイド』	Enterprise 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。開発者ツール、セキュリティ、デバッグ、ライフサイクルモジュールの作成に関する情報を含みます。
『配備計画ガイド』	配備の計画と設定。
『アップグレードガイド』	旧バージョンの製品から Enterprise Server 2.1 へのアップグレード。
『Java EE 5 Tutorial』	Java EE 5 プラットフォームテクノロジーと API を使用した Java EE アプリケーションの開発。
『Java WSIT Tutorial』	Web サービス相互運用性テクノロジー (WSIT) を使用した Web アプリケーションの開発。WSIT テクノロジーを使用する方法、時期、および理由と、各テクノロジーがサポートする機能およびオプションについて説明します。
『管理ガイド』	設定、監視、セキュリティ、資源管理、および Web サービス管理を含む Enterprise Server のシステム管理。
『高可用性 (HA) 管理ガイド』	クラスタの設定、ノードエージェントの操作、およびロードバランサの使用。

表 1-1 Enterprise Server のマニュアルセットの内容 (続き)

マニュアル名	説明
『Administration Reference』	Enterprise Server 設定ファイル <code>domain.xml</code> の編集。
『パフォーマンスチューニングガイド』	パフォーマンスを向上させるための Enterprise Server の調整。
『Reference Manual』	Enterprise 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。 <code>asadmin</code> コマンド行インタフェースも含まれます。

## 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最新バージョンは、<http://sun.com/software/javaenterprisesystem/get.html> で入手できます。

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。

## 問題の報告とフィードバックの方法

Sun GlassFish Enterprise Server に問題が発生した場合は、次のいずれかの方法で Sun にお問い合わせください。

- **SailFin** メーリングリスト  
(<https://sailfin.dev.java.net/servlets/ProjectMailingListList>) — 多くの情報やフィードバックを得られる、さまざまな SailFin コミュニティーのメーリングリスト
- **Java Developer Connection** のバグデータベース (<http://developer.java.sun.com/servlet/SessionServlet?url=/developer/bugParade/index.jshtml>) — バグを表示または提出するには、Java Developer Connection Bug Parade を使用します。
- **Java Developer Connection** のバグデータベース (<http://developer.java.sun.com/servlet/SessionServlet?url=/developer/bugParade/index.jshtml>) — バグを表示または提出するには、Java Developer Connection Bug Parade を使用します。
- **SailFin Issue Tracker** (<https://sailfin.dev.java.net/servlets/ProjectIssues>)
- **Java Technology Forums** (<http://forum.java.sun.com/>) — Java テクノロジーおよびプログラミング技術に関する情報や質問を共有するための、対話型のメッセージボード。
- **Sun Software Support Services** (<http://www.sun.com/service/sunone/software>) - ナレッジベース、オンラインサポートセンター、Product Tracker、および保守プログラムやサポートに関する問い合わせ番号へのリンク

- 保守契約を結んでいるお客様の場合は、専用ダイヤルをご利用ください。  
最善の問題解決のため、サポートに連絡する際には次の情報をご用意ください。
- 問題が発生した箇所や動作への影響など、問題の具体的な説明
- マシン機種、OSバージョン、および製品のバージョン(問題に関係するパッチおよびその他のソフトウェアを含む)
- 問題を再現するための具体的な手順の説明
- エラーログやコアダンプ

## このマニュアルに関するコメント

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。

コメントを共有するには、<http://docs.sun.com> にアクセスし、「Feedback」をクリックします。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、このドキュメントのタイトルは『Sun GlassFish Enterprise Server 2.1 リリースノート』、Part No. は 820-7428 です。

## その他の情報

以下の場所から有益な情報を入手することができます。

- Enterprise Server 製品情報  
([http://www.sun.com/software/products/appsrvr/home\\_appsrvr.html](http://www.sun.com/software/products/appsrvr/home_appsrvr.html))
- プロフェッショナルサービス (<http://www.sun.com/service/sunps/sunone>)
- ソフトウェア製品とサービス (<http://www.sun.com/software>)
- ソフトウェアサポートサービス (<http://www.sun.com/service/sunone/software>)
- サポートとナレッジベース (<http://www.sun.com/service/support/software>)
- Sun Support およびトレーニングサービス (<http://training.sun.com>)
- コンサルティングと Professional Services  
(<http://www.sun.com/service/sunps/sunone>)
- 開発者向け情報 (<http://developers.sun.com>)
- Sun Developer Support Services (<http://www.sun.com/developers/support>)
- ソフトウェアトレーニング (<http://www.sun.com/software/training>)
- Sun ソフトウェアデータシート (<http://www.sun.com/software>)
- Sun Microsystems 製品ドキュメント (<http://docs.sun.com/>)

# Sun GlassFish Enterprise Server について

---

Sun GlassFish Enterprise Server は、大規模な本稼働環境での 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。

この章で説明する内容は、次のとおりです。

- 7 ページの「Sun GlassFish Enterprise Server 2.1 の新機能」
- 7 ページの「Sun GlassFish Enterprise Server の機能」
- 11 ページの「ハードウェアとソフトウェアの要件」
- 25 ページの「Update Center の使用法」

## Sun GlassFish Enterprise Server 2.1 の新機能

バグの修正と拡張機能 - 次の表に、このリリースで修正されたバグの一覧を示します。

バグ ID	説明
6571044	sunpkcs11.jar がいないためドメインを開始できない
6165528	as-install/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

## Sun GlassFish Enterprise Server の機能

次の機能が、Sun Java System Application Server 9.1 で導入されました。Sun GlassFish Enterprise Server に実装された機能は、次のとおりです。

- **Sun Connection** への登録 - インストーラ、管理コンソール GUI、または Update Center を使用して、製品を **Sun Connection** (<http://www.sun.com/service/sunconnection/index.jsp>) へ登録できます。Sun Connection で Enterprise Server を登録すると、次の利点を受けることができます。
  - パッチ情報とバグ更新
  - スクリーンキャストとチュートリアル
  - ニュースとイベント
  - サポートとトレーニングの提供
- **AIX** オペレーティングシステムのサポート - Enterprise Server は、開発者プロファイルまたはクラスタプロファイルを使用して作成されているドメインにおいて AIX オペレーティングシステムでサポートされています。

---

注 - HADB および NSS が AIX オペレーティングシステムでサポートされていないため、エンタープライズプロファイルは AIX オペレーティングシステムではサポートされていません。

---

- **Ubuntu** オペレーティングシステムのサポート - Enterprise Server は Ubuntu Linux オペレーティングシステムに含まれています。

---

注 - Enterprise Server ドキュメントセット内のインストール情報は、このオペレーティングシステムには該当しません。HADB および NSS が Ubuntu Linux オペレーティングシステムでサポートされていないため、エンタープライズプロファイルは Ubuntu Linux オペレーティングシステムではサポートされていません。

---

- **SuSe Linux 64** ビット版のサポート
- マルチレベルの関係プリフェッチのサポート - コンテナ管理による持続性 (CMP) 2.1 エンティティー Bean に対するマルチレベルの関係プリフェッチがサポートされるようになりました。詳細については、[26 ページの「マルチレベルの関係プリフェッチの有効化」](#)を参照してください。
- 拡張された **JB1** のサポート - 既に配備されているサービスアセンブリの再配備を必要とせずに、管理コンソール GUI またはコマンド行から JB1 コンポーネントを更新できます。
- **Java EE 5** プラットフォームのサポート - Sun GlassFish Enterprise Server 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。
  - 。 Enterprise Server 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、このノートで後述します。

- **Web サービス相互運用性テクノロジー (WSIT) サポート** – Sun は Microsoft と密接に連携して、メッセージ最適化、高信頼性メッセージング、およびセキュリティーなどの Web サービスエンタープライズテクノロジーの相互運用性を実現しています。WSIT の初期リリースはこの協力の成果です。WSIT は、エンタープライズ機能をサポートする多くのオープンな Web サービス仕様を実装したものです。メッセージ最適化、信頼できるメッセージング、およびセキュリティーに加えて、WSIT にはブートストラップと設定のテクノロジーも含まれています。現在 Java プラットフォームに組み込まれているコア XML サポートを基本にして、WSIT は既存の機能を使用または拡張し、相互運用可能な Web サービスのための新しいサポートを追加します。それらには、次のサポートが含まれます。
  - ブートストラップおよび設定
  - メッセージ最適化テクノロジー
  - 高信頼性メッセージングテクノロジー
  - セキュリティーテクノロジー

Enterprise Server での WSIT 統合の詳細については、この章の [26 ページの「WSIT 統合の詳細について」](#) で後述します。

- **JBIC サポート** – JBIC はビジネス統合 Service Provider Interfaces (SPI) によって Java EE を拡張します。これらの SPI を使用することにより、開発者は WSCI、BPEL4WS、および W3C Choreography Working Group などの仕様に合わせ Java ビジネス統合環境を作成または実装できます。JBIC 実装は直接 Enterprise Server インストーラにより通常は *as-install/jbic* ディレクトリにインストールされます。Enterprise Server JVM で JBIC フレームワークを開始する `lifecycle` モジュールを含む、JBIC 用のすべての共通 JAR ファイルとシステムコンポーネントがこのディレクトリに含まれています。
- **インメモリーレプリケーションサポート** – ほかのサーバー上でインメモリーレプリケーションを実行することにより、HADB などの別個のデータベースを入手しなくてもセッション状態データの軽量ストレージを用意できます。このタイプのレプリケーションは、ほかのサーバー上のメモリーを使用して HTTP セッションとステートフルセッション Bean データの高可用性ストレージを実現します。クラスタ化されたサーバーインスタンスはセッション状態をリングトポロジで複製します。各バックアップインスタンスは複製されたデータをメモリーに格納します。セッション状態データをほかのサーバー上のメモリーに複製することによって

、セッションを分散することが可能になります。インメモリーレプリケーションを使用するには、グループ管理サービス (GMS) を有効にする必要があります。デフォルトで有効です。

- 使用状況プロファイル - どの管理ドメインもそのドメインの能力を示す使用状況プロファイルと関連付けられています。Enterprise Server には次のプロファイルが用意されています。
  - 開発者 - ドメインを開発環境で実行している場合や、アプリケーションに負荷分散、高可用性、セッションレプリケーションなどのクラスタ化機能が必要でない場合に、このプロファイルを使用します。このプロファイルの実際の名前は「developer」で、大小文字を区別することに注意してください。
  - クラスタ - アプリケーションサーバーインスタンスのクラスタを作成して、配備された Java EE アプリケーションにスケラビリティと高可用性を付与する場合は、このプロファイルを使用します。アプリケーションの状態はインメモリーのままとなります。このプロファイルの実際の名前は「cluster」で、大小文字を区別することに注意してください。
  - エンタープライズ - HADB と NSS が必要な場合に、このプロファイルを使用します。このプロファイルは、HADB と NSS が別々にインストールされていたり、Enterprise Server が Java Enterprise System (Java ES) の一部としてインストールされていたりする場合には使用できません。
- 負荷分散の拡張機能 - 負荷分散プラグインにいくつかの拡張機能が追加されました。簡潔にまとめると、次の機能が含まれます。
  - 重み付けラウンドロビン - オプション属性 `weight` がインスタンス `Loadbalancer.xml` ファイルに追加されています。このオプションを使用することにより、ロードバランサプラグインは重みに応じて要求を経路指定できるようになります。たとえば、500 の要求のうち、100 が `instance1` へ、400 が `instance2` へと毎回振り分けられるようになります。デフォルトの重みは 100 です。重みは管理コンソールまたはコマンド行から各インスタンスに対して割り当てられ、サーバーの `domain.xml` にはインスタンスごとの重みを示す属性があります。
  - ユーザー定義によるロードバランサの方法指定 - ユーザーが負荷分散のカスタムロジックを定義できます。たとえば、ユーザー ID に基づくリダイレクトや MIME ベースの負荷分散があります。この機能は、ロードバランサがロードするユーザー定義の共用ライブラリによって実装されます。このカスタム共用ライブラリは、`as-install/lib/install/templates/` にある `loadbalancer.h` に定義されているとおりにインタフェースを実装します。
  - 管理拡張機能 - 以前のバージョンの Enterprise Server では、`loadbalancer.xml` ファイルを特定のサーバーの `config` ディレクトリに手動でコピーすることが必要でした。Enterprise Server には、このようなコピーを Web サーバーと Enterprise Server の間で実行するプッシュ自動化が組み込まれています。ロードバランサ自体はサーバーの `domain.xml` ファイルで設定されます。

- 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。Enterprise Server 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。
- **Update Center** サポート - Enterprise Server Update Center により、Enterprise Server の更新が自動化され、追加コンポーネントが簡単に利用できるようになりました。詳細は、25 ページの「**Update Center** の使用法」を参照してください。

## ハードウェアとソフトウェアの要件

この節では、Sun GlassFish Enterprise Server 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。

- 11 ページの「サポートされるプラットフォーム」
- 13 ページの「重要パッチ情報」
- 13 ページの「JDK バージョン」
- 14 ページの「サポートされている Java SE バージョンへ切り換える」
- 14 ページの「Apache Ant のバージョン」
- 14 ページの「JDBC ドライバとデータベース」
- 15 ページの「バンドル版 Java DB データベースの使用」
- 17 ページの「Message Queue のバージョン」
- 17 ページの「Web サーバー」
- 18 ページの「ブラウザ」
- 18 ページの「HADB の要件とサポートされているプラットフォーム」
- 19 ページの「Enterprise Server のアップグレード」
- 19 ページの「その他の要件」

### サポートされるプラットフォーム

次の表に、Sun GlassFish Enterprise Server 2.1 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
Solaris 9、10 (x86)					Java SE 6

表 2-1 サポートされるオペレーティングシステム (続き)

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
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
RedHat Enterprise Linux 4.0、5.x	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
SuSE Linux Enterprise Server 10 SP1	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
64ビット版 SuSE Linux Enterprise Server 10 SP1	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
Ubuntu Linux (Hardy Release)	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
AIX 5.2、5.3	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
Windows Server 2000 SP4+  Windows 2000 Advanced Server SP4+  Windows Server 2003  Windows XP Pro SP1+  Windows Vista	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
Macintosh (Intel、Power)  開発のみでサ ポート。	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	Java SE 5
OpenSolaris  評価サポート のみ	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	Java SE 5 Java SE 6

---

注 - Microsoft Windows プラットフォームで Enterprise 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。

---

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 - MacOS は本稼働環境への配備ではサポートされませんが、開発の目的ではサポートされます。Macintosh オペレーティングシステムのダウンロード情報は、GlassFish ダウンロードページか、Mac が一覧に示されている SDK ページから入手できます。たとえば、JavaEE 5 SDK Update 5 のダウンロードページ [https://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS\\_Developer-Site/en\\_US/-/USD/ViewProductDetail-Start/ProductRef=java\\_ee\\_sdk-5\\_05-nojdk-oth-JPR@CDS-CDS\\_Developer](https://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS_Developer-Site/en_US/-/USD/ViewProductDetail-Start/ProductRef=java_ee_sdk-5_05-nojdk-oth-JPR@CDS-CDS_Developer) で入手できます。

---

## 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 製品のテストを行なっています。これは、Sun Java System 製品が、適切な規模と構成の仮想化環境でも、仮想化されていないシステム上の場合と同様に機能することを効果的に確認するためのテストです。仮想化された環境における 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

### Solaris のパッチ要件

Solaris 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) (<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」で入手できます。

### JDK バージョン

Enterprise Server で必要な、動作が保証されている JDK の最小バージョンは、1.5.0\_14 です。

## ▼ サポートされている Java SE バージョンへ切り換える

次の手順で `asenv file` を編集して、サポートされている Java SE バージョンに切り替えることができます。

- 1 システムに新しい Java SE バージョンをインストールします (インストールしていない場合)。  
Java SE SDK は <http://java.sun.com/javase> からダウンロードできます。
- 2 **Enterprise Server** を停止します。
  - コマンド行から、次のように入力します。

```
install_dir/bin/asadmin stop-domain
```
  - 管理コンソールで次のようにします。
    - 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 Java 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、`com.sun.aas.javaRoot...` で始まる行を、新しい Java ホームディレクトリを参照するように変更します。
- 5 **Application Server** を再起動します。
  - コマンド行から、次のように入力します。

```
as-install/bin/asadmin start-domain
```
  - 管理コンソールで次のようにします。
    - 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. 「インスタンスを起動」をクリックします。

### Apache Ant のバージョン

Ubuntu Linux を除くすべてのオペレーティングシステムでは、Apache Ant の Version 「1.6.5」が Enterprise Server に含まれています。Ubuntu Linux では、Apache Ant の Version 「1.7.0」が Enterprise Server に含まれており、このバージョンが使用されます。

### 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 GlassFish Enterprise Server の構成では、バンドル版 Java DB デ

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、Enterprise 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
Derby Network Client	Type 4	Derby 10.2
DataDirect 3.6.x, 3.7.x	Type 4	Oracle 10g Oracle 9i Sybase ASE 15 MS SQL 2005 DB2 9.1 DB2 8.2
MySQL Connector/J Driver 5.1.6	Type 4	MySQL 5.0
Oracle 10G	Type 4	Oracle 10g -RAC
PostGres	Type 4	8.2.x

表 2-3 に、Enterprise Server FCS リリースでサポートを予定している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 JDBC ドライバとデータベース (Enterprise Server FCS)

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
Oracle OCI	Type 2	Oracle 9i
IBM DB2	Type 2	DB2 9.1

通常、Enterprise Server 2.1 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、Enterprise Server 2.1 にバンドルされた Java DB データベース実装の使用方法について説明します。Java DB は Apache Derby データベースに基づいています。

- 16 ページの「Java DB データベースの起動と停止」
- 16 ページの「Java DB ユーティリティースクリプト」

## Java DB データベースの起動と停止

Sun GlassFish Enterprise Server 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に、次の `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。

```
asadmin 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 `as-install/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。

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

## Java DB ユーティリティースクリプト

Enterprise Server 2.1 に付属の Java DB の設定には、Java DB の使用に役立つスクリプトが含まれます。`as-install/javadb/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`、`startNetworkServer.bat`— ネットワークサーバーを開始するスクリプト
- `stopNetworkServer`、`stopNetworkServer.bat`— ネットワークサーバーを停止するスクリプト
- `ij`、`ij.bat`— 対話式の JDBC スクリプト記述ツール
- `dblook`、`dblook.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`、`sysinfo.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`、`NetworkServerControl.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 `JAVA_HOME` 環境変数が JDK のインストールディレクトリを指定していることを確認します。
- 2 `as-install/javadb` ディレクトリをポイントするように `DERBY_HOME` 環境変数を設定します。

参照 これらのユーティリティの詳細については、次の Derby マニュアルを参照してください。

- [Derby Tools and Utilities Guide \(http://db.apache.org/derby/docs/10.1/tools/\)](http://db.apache.org/derby/docs/10.1/tools/)
- [Derby Server and Administration Guide \(http://db.apache.org/derby/docs/10.1/adminguide/\)](http://db.apache.org/derby/docs/10.1/adminguide/)

### Message Queue のバージョン

ファイルベースのインストールでは、Message Queue 4.3 が Enterprise Server と一緒にパッケージされています。UNIX のパッケージベースのインストールでは、Message Queue 4.3 と同等の機能を持つ、Message Queue 4.1.1 が提供されます。バージョンの数字が 4.1.1 と小さいにもかかわらず、Message Queue 4.1.1 は Message Queue 4.2 と下位互換性を持ちます。

Enterprise Server の一部として提供される埋め込みの Sun GlassFish Message Queue コードは、一般的に同等のバージョンの Message Queue プローカに対してのみ、テストおよび認証されています。つまり、提供される埋め込みの Message Queue コードを使用して、別のバージョンの Message Queue コードを実行しているリモート (Enterprise Server で管理されていない) Message Queue プローカに接続することは、サポートされません。

### Web サーバー

この節では、Sun GlassFish Enterprise Server 向けにサポートされている Web サーバーを示します。

表 2-4 サポートされている Web サーバー

Web サーバー	バージョン	オペレーティングシステム
Sun Java System Web Server (32 ビット)	6.1、7.0	Solaris SPARC 8、9、10 Solaris x86 9、10 Red Hat Enterprise Linux 2.1 Update 2、3.0 Update 1
Apache Web Server (32 ビット)	2.0.x、2.2.x	Solaris SPARC 9、10 Solaris x86 10 Red Hat Enterprise Linux 2.1 Update 2、3.0 Update 1
Microsoft IIS (32 ビット)	5.0+	Windows Server 2003 Windows 2000 Advanced Server (Service Pack 4 以上)

32 ビットのプラットフォームのみがサポートされます。

## ブラウザ

ここでは、Sun GlassFish Enterprise Server 2.1 でサポートされているブラウザを示します。

表 2-5 サポートされている Web ブラウザ

ブラウザ	バージョン
Mozilla	1.7.12
Internet Explorer	6.0 Service Pack 2、7.0
Firefox	1.5.x、2.x
Safari	2
Netscape	8.0.4、8.1、9.0、9.0.x

## HADBの要件とサポートされているプラットフォーム

11 ページの「ハードウェアとソフトウェアの要件」に示されている要件に加えて、システムが、下に示されている HADB を実行するための要件を満たしていることを確認してください。

- 18 ページの「サポートされるプラットフォーム」
- 19 ページの「HADB サーバーホストの要件」
- 19 ページの「HADB 管理ホストの要件」
- 19 ページの「HADB クライアントホストの要件」

注 - HADB は Enterprise Server 2.1 Enterprise プロファイルにのみバンドルされています。システムの Java コンポーネントは JDK 5 を使用して構築され、JDK 1.5 上でテストされています。

## サポートされるプラットフォーム

- **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 (ext3 はサポートされず、ext2 ファイルシステムのみがサポートされている)、3.0 U4 (ext2 と ext3 の両方がサポートされている)。U4 より前のアップデートは、過剰なスワッピングのために推奨されない。これらのオペレーティングシステムバージョンでは、HADB が 32 ビットモードでのみテストされていることに注意してください。また、HADB が、オペレーティングシステム内のバグのために 64 ビットモードで動作している RedHat Enterprise Linux 3.0 をサポートしていないことにも注意してください (HADB への影響の詳細は、43 ページの「高可用性」の既知のバグ 6249685 を参照)。

- **Microsoft Windows** – Microsoft Windows 2000 Advanced Server Service Pack 4 および Microsoft Windows 2003 Enterprise Edition。HADB が、間もなく登場する、64 ビットモードの Microsoft Windows オペレーティングシステムバージョンをまったくサポートしていないことに注意してください。

#### HADB サーバーホストの要件

- 最小メモリー - ノードあたり 320M バイト。
- 最小空きディスク容量 - ホストあたり HADB バイナリ用に 70M バイト。さらに、データデバイスにもディスク容量が必要、ノードあたりテストインストール用に 512M バイト。
- 推奨メモリー - ノードあたり 512M バイト。
- 推奨空きディスク容量 - ホストあたり HADB バイナリ用に 70M バイト。さらに、データデバイスにもディスク容量が必要、ノードあたりテストインストール用に 1200M バイト。

---

注 - HADB データとログファイルを格納するデバイスで書き込みキャッシュが無効になっていることを確認してください。一部の Solaris プラットフォーム (たとえば、Solaris x86) では、デフォルトで書き込みキャッシュが有効になります。

---

#### HADB 管理ホストの要件

- 最小メモリー - 128M バイト
- 最小空きディスク容量 - ノードあたり HADB バイナリ用に 70M バイト

#### HADB クライアントホストの要件

- 最小メモリー - 120M バイト
- 最小空きディスク容量 - 20M バイト

#### Enterprise Server のアップグレード

以前のバージョンの Enterprise Server から現在のバージョンへのアップグレードに関する詳細な手順については、『[Sun GlassFish Enterprise Server 2.1 Upgrade guide](#)』を参照してください。

#### その他の要件

Sun GlassFish Enterprise 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。

- **空き領域:** 一時ディレクトリには、少なくとも、Sun GlassFish Enterprise Server のインストール用に 35M 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。

- アンインストールプログラムの使用: システムから Enterprise Server を削除する必要がある場合は、ソフトウェアに付属のアンインストールプログラムを使用することが重要です。ほかの方法を使用して削除すると、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。
- 空きポート: 利用可能な未使用ポートが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ドメインとして示します。デフォルトの初期ポートは次のとおりです。
  - HTTP については 8080、HTTPS については 8181
  - IIOP については 3700、IIOP/SSL については 3820、相互認証の IIOP/SSL については 3920
  - Administration Server については 4848 (HTTP)
  - pure JMX Client については 8686 (RMI) ポート

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 GlassFish Enterprise Server 2.1 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) - 以前のバージョンの Enterprise Server から現在のバージョンへのアップグレードに関する詳細な手順については、『[Sun GlassFish Enterprise Server 2.1 Upgrade guide](#)』を参照してください。
- ファイアウォールのシャットダウン (Microsoft Windows) - Sun GlassFish Enterprise 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、利用可能なポートを正確に検出できる必要があります。

互換性の詳細は、『[Sun GlassFish Enterprise Server 2.1 Upgrade guide](#)』を参照してください。

## Java EE 5 プラットフォームの API

Sun GlassFish Enterprise Server 2.1 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 2-6 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	

表 2-6 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 Platform, Enterprise Edition 5 ( <a href="http://java.sun.com/javaee/5/docs/api/">http://java.sun.com/javaee/5/docs/api/</a> )	JSR 244 ( <a href="http://jcp.org/aboutJava/communityprocess/pr/jsr244/">http://jcp.org/aboutJava/communityprocess/pr/jsr244/</a> )
<b>Web サービステクノロジー</b>	
エンタープライズ Web サービスの実装	JSR 109 ( <a href="http://jcp.org/en/jsr/detail?id=109">http://jcp.org/en/jsr/detail?id=109</a> )
Java API for XML-Based Web Services (JAX-WS) 2.0 ( <a href="https://jax-ws.dev.java.net/">https://jax-ws.dev.java.net/</a> )	JSR 224 ( <a href="http://jcp.org/en/jsr/detail?id=224">http://jcp.org/en/jsr/detail?id=224</a> )
Java API for XML-Based RPC (JAX-RPC) 1.1 ( <a href="https://jax-rpc.dev.java.net/">https://jax-rpc.dev.java.net/</a> )	JSR 101 ( <a href="http://jcp.org/en/jsr/detail?id=101">http://jcp.org/en/jsr/detail?id=101</a> )
Java Architecture for XML Binding (JAXB) 2.0 ( <a href="https://jaxb.dev.java.net/">https://jaxb.dev.java.net/</a> )	JSR 222 ( <a href="http://jcp.org/en/jsr/detail?id=222">http://jcp.org/en/jsr/detail?id=222</a> )
SOAP with Attachments API for Java (SAAJ) ( <a href="https://saaj.dev.java.net/">https://saaj.dev.java.net/</a> )	JSR 67 ( <a href="http://jcp.org/en/jsr/detail?id=67">http://jcp.org/en/jsr/detail?id=67</a> )
Streaming API for XML ( <a href="http://java.sun.com/webservices/docs/1.6/tutorial/doc/SJSXP.html">http://java.sun.com/webservices/docs/1.6/tutorial/doc/SJSXP.html</a> )	JSR 173 ( <a href="http://jcp.org/en/jsr/detail?id=173">http://jcp.org/en/jsr/detail?id=173</a> )
Java プラットフォーム用 Web サービスメタデータ	JSR 181 ( <a href="http://jcp.org/en/jsr/detail?id=181">http://jcp.org/en/jsr/detail?id=181</a> )
<b>コンポーネントモデルテクノロジー</b>	
Enterprise JavaBeans 3.0 ( <a href="http://java.sun.com/products/ejb/">http://java.sun.com/products/ejb/</a> )	JSR 220 ( <a href="http://jcp.org/en/jsr/detail?id=220">http://jcp.org/en/jsr/detail?id=220</a> )
J2EE Connector Architecture 1.5 ( <a href="http://java.sun.com/j2ee/connector/">http://java.sun.com/j2ee/connector/</a> )	JSR 112 ( <a href="http://jcp.org/en/jsr/detail?id=112">http://jcp.org/en/jsr/detail?id=112</a> )
Java Servlet 2.5 ( <a href="http://java.sun.com/products/servlet/">http://java.sun.com/products/servlet/</a> )	JSR 154 ( <a href="http://jcp.org/en/jsr/detail?id=154">http://jcp.org/en/jsr/detail?id=154</a> )
JavaServer Faces 1.2 ( <a href="http://java.sun.com/j2ee/javaserverfaces/">http://java.sun.com/j2ee/javaserverfaces/</a> )	JSR 252 ( <a href="http://jcp.org/en/jsr/detail?id=252">http://jcp.org/en/jsr/detail?id=252</a> )
JavaServer Pages 2.1 ( <a href="http://java.sun.com/products/jsp/">http://java.sun.com/products/jsp/</a> )	JSR 245 ( <a href="http://jcp.org/en/jsr/detail?id=245">http://jcp.org/en/jsr/detail?id=245</a> )
JavaServer Pages Standard Tag Library 1.2 ( <a href="http://java.sun.com/products/jsp/jstl/">http://java.sun.com/products/jsp/jstl/</a> )	JSR 52 ( <a href="http://jcp.org/en/jsr/detail?id=52">http://jcp.org/en/jsr/detail?id=52</a> )
<b>管理テクノロジー</b>	
J2EE Management ( <a href="http://java.sun.com/j2ee/tools/management/">http://java.sun.com/j2ee/tools/management/</a> )	JSR 77 ( <a href="http://jcp.org/en/jsr/detail?id=77">http://jcp.org/en/jsr/detail?id=77</a> )
J2EE Application Deployment ( <a href="http://java.sun.com/j2ee/tools/deployment/">http://java.sun.com/j2ee/tools/deployment/</a> )	JSR 88 ( <a href="http://jcp.org/en/jsr/detail?id=88">http://jcp.org/en/jsr/detail?id=88</a> )

表 2-6 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 ( <a href="http://java.sun.com/j2ee/javaacc/">http://java.sun.com/j2ee/javaacc/</a> )	JSR 115 ( <a href="http://jcp.org/en/jsr/detail?id=115">http://jcp.org/en/jsr/detail?id=115</a> )
その他の <b>Java EE</b> テクノロジー	
Common Annotations for the Java Platform	JSR 250 ( <a href="http://jcp.org/en/jsr/detail?id=250">http://jcp.org/en/jsr/detail?id=250</a> )
Java Transaction API (JTA) ( <a href="http://java.sun.com/products/jta/">http://java.sun.com/products/jta/</a> )	JSR 907 ( <a href="http://jcp.org/en/jsr/detail?id=907">http://jcp.org/en/jsr/detail?id=907</a> )
JavaBeans Activation Framework (JAF) 1.1 ( <a href="http://java.sun.com/products/javabeans/glasgow/jaf.html">http://java.sun.com/products/javabeans/glasgow/jaf.html</a> )	JSR 925 ( <a href="http://jcp.org/en/jsr/detail?id=925">http://jcp.org/en/jsr/detail?id=925</a> )
JavaMail ( <a href="http://java.sun.com/products/javamail/">http://java.sun.com/products/javamail/</a> )	JSR 919 ( <a href="http://jcp.org/en/jsr/detail?id=919">http://jcp.org/en/jsr/detail?id=919</a> )
Java Message Service API ( <a href="http://java.sun.com/products/jms/">http://java.sun.com/products/jms/</a> )	JSR 914 ( <a href="http://www.jcp.org/en/jsr/detail?id=914">http://www.jcp.org/en/jsr/detail?id=914</a> )
Java Persistence API ( <a href="http://java.sun.com/j2ee/persistence/faq.html">http://java.sun.com/j2ee/persistence/faq.html</a> )	JSR 220 ( <a href="http://www.jcp.org/en/jsr/detail?id=220">http://www.jcp.org/en/jsr/detail?id=220</a> )

## Java EE 5 SDK

Sun GlassFish Enterprise Server 2.1 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 2 つのバージョンがあります。

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

さらに、これらの SDK 配布を JDK または NetBeans 5.5.1 ツールとともにダウンロードすることもできます。詳細は、ダウンロードページの

<http://java.sun.com/javaee/downloads/index.jsp> を参照してください。

---

注 - GlassFish v2.1 と Enterprise Server 2.1 は、Java EE アプリケーションの開発では NetBeans 5.5 と連携していません。NetBeans 5.5.1 を使用する必要があります。

---

## サポートされている別の Java バージョンへの切り換え

Sun GlassFish Enterprise Server 2.1 では、基盤となる JVM として Java SE 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)

---

注-新しい Java VM でドメインを作成したあとに以前の Java バージョンにダウングレードすることは推奨されていません。JVM をダウングレードする必要がある場合は、ドメインごとにそうすることをおすすめします。次の手順はその方法を説明したものです。

---

## ▼ サポートされている別の Java バージョンへ切り換えるには

- 1 必要に応じて、希望する Java SDK (JRE ではない) をダウンロードしてシステムにインストールします。

Java SDK は、<http://java.sun.com/j2se> でダウンロードできます。

- 2 JDK を変更するドメインを開始します。

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

- 3 管理コンソールにログインして、ドメインの JVM 属性を変更します。

特に、ドメインの「JVM 設定」ページの JAVA\_HOME 変数を変更することが必要です。

または、asadmin コマンドを使用することもできます。

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

## 既知の Java ES 5 互換性の問題

Enterprise Server 2.1 と Java ES 5 (JES5) の間には、既知の互換性の問題が2つあります。

1. Enterprise Server 2.1 で提供される JSF 1.2 のバージョンのため、JES5 Service Registry は Enterprise Server 2.1 と互換性がありません。Enterprise Server を 2.1 にアップグレードする前に、Service Registry を JES5u1 にアップグレードする必要があります。この制限は Service Registry のドキュメントにも記載されています。
2. Enterprise Server で提供される JSF 1.2 のバージョンのため、JES5 Portal Server は Enterprise Server 2.1 と互換性がありません。2.1 にアップグレードする前に、Portal Server を JES5u1 にアップグレードする必要があります。

Enterprise Server 8.2 上で Portal Server を実行する Java ES 5 設定環境で、Enterprise Server 8.2 を 2.1 にアップグレードすると、Portal Server が使用できなくなります。Enterprise Server 2.1 は JSF 1.2 を使用しますが、Release 5 Portal Server の JSF-Portlet ブリッジは JSF 1.2 をサポートしません。Solaris/Linux では、Portal Server を JavaES5 Update 1 にアップグレードする必要があります。Windows で Release 5 Portal Server

をそのまま使用する場合は、Enterprise Server を 2.1 にアップグレードしないでください。Portal Server は、Windows 上の Java ES 5 Update 1 ではサポートされません。

## 一部のオペレーティングシステムのみでサポートされる機能

Enterprise Server には、一部のオペレーティングシステムでしかサポートされていない機能があります。それらの機能の詳細を以下に示します。

### AIX オペレーティングシステムではサポートされない機能

AIX オペレーティングシステムを使用している場合、次の制限が適用されます。

- Enterprise Server は、開発者プロファイルまたはクラスタプロファイルを使用されたドメインに限り、AIX オペレーティングシステムでサポートされます。HADB および NSS が AIX オペレーティングシステムでサポートされていないため、エンタープライズプロファイルは AIX オペレーティングシステムではサポートされていません。
- AIX オペレーティングシステムの場合、ネイティブ起動プログラムを使用できません。

### Linux オペレーティングシステムではサポートされない機能

Linux オペレーティングシステムを使用している場合、次の制限が適用されます。

- 64 ビット JVM 上の SuSe Linux 64 ビットオペレーティングシステムおよび RHEL 64 ビットオペレーティングシステムでは、ネイティブのアプリケーションサーバー起動ツールを使用できません。32 ビット JVM を使用してください。
- 64 ビット JVM 上の SuSe Linux 64 ビットオペレーティングシステムおよび RHEL 64 ビットオペレーティングシステムでは、インストーラを使用できません。32 ビット JVM を使用してください。
- 次のプラットフォームでは、エンタープライズプロファイルはサポートされません。
  - RedHat Enterprise Linux 64 ビットオペレーティングシステムおよび 64 ビット JVM
  - SuSe Linux 64 ビットオペレーティングシステムおよび 64 ビット JVM

### Ubuntu オペレーティングシステムではサポートされない機能

Ubuntu Linux オペレーティングシステムを使用している場合、次の制限が適用されます。

- Enterprise Server 2.1 は Ubuntu Linux オペレーティングシステムに含まれているため、Enterprise Server 2.1 ドキュメントセット内のインストール情報はこのオペレーティングシステムには該当しません。

- HADB および NSS が Ubuntu Linux オペレーティングシステムでサポートされていないため、エンタープライズプロファイルは Ubuntu Linux オペレーティングシステムではサポートされていません。

## Update Center の使用法

Update Center により、Enterprise Server の更新が自動化され、追加コンポーネントが簡単に利用できるようになりました。

Update Center を有効にすると、自動でソフトウェアの更新を実行します。この自動更新プロセス中に、Update Center は次のデータを収集して Sun Microsystems または購入先に転送します。

- 固有インストール ID (GUID)
- IP アドレス
- オペレーティングシステム情報 (名前、バージョン、アーキテクチャー、ロケール)
- JDK バージョン
- モジュールダウンロード情報 (モジュール名、日付、時刻、状況、ダウンロード時間、ダウンロード済みバイト数)

個人情報は追跡しません。どの個人情報もほかのデータと関連付けられておらず、レポート用に使用されません。

## ▼ Update Center を使用するには

自動アップデートへの同意を明示するため、Update Center はデフォルトでは無効になっています。Update Center を有効にして、定期チェックと自動アップデートを実行するには、次の手順に従います。

- 1 Update Center を開始します。
  - UNIX の場合: `<installdir>/updatecenter/bin/updatetool`
  - Windows の場合: `<installdir>\updatecenter\bin\updatetool.bat`
- 2 「Preferences」タブをクリックします。
- 3 「Update Scheduling」ウィンドウで、「Check for Updates」ドロップダウンボックスの値を「Never (Manual)」から希望する値に変更します。一例として、毎日や毎週などを指定できます。
- 4 アップデートを希望する曜日と時刻を指定します。

- 5 変更を保存するには、「Save」ボタンを選択します。

これで、Update Center は指定したスケジュールに従って、SailFin または Enterprise Server コンポーネントのアップデートを自動的に確認します。アップデートが利用可能である場合には、Update Center が起動して更新可能なコンポーネントを通知します。

## Ubuntu オペレーティングシステムでの Enterprise Server のアップグレード

Ubuntu Linux オペレーティングシステムでは、GlassFish v2 をインストールしても既存の GlassFish v1 のインストールは上書きされません。/usr/bin 内の asadmin スクリプトは GlassFish v2 を管理します。GlassFish v1 を管理するには、/usr/share/sunappserver/bin 内のローカルの asadmin ユーティリティを使用します。v1 ドメインを v2 にアップグレードするには、/usr/share/glassfishv2/bin にある asupgrade ツールを使用します。

グローバルの asadmin スクリプトを使用して v1 ドメインを開始すると、アップグレードツールが自動的に起動します。ただし、これによってエラーが発生します。エラーの原因は、サーバーのドメインの場所に関して、(Ubuntu 以外の) ほかのプラットフォームでのデフォルトの場所が使用されるというバグです。このエラーを回避するには、代わりに asupgrade ツールを使用します。

## マルチレベルの関係プリフェッチの有効化

マルチレベルの関係プリフェッチが CMP 2.1 エンティティ Bean でサポートされています。

マルチレベルの関係プリフェッチを有効にするには、次のプロパティを設定します。

```
-Dcom.sun.jdo.spi.persistence.support.sqlstore.MULTILEVEL_PREFETCH=true
```

関係プリフェッチの詳細は、『[Sun GlassFish Enterprise Server 2.1 Developer's Guide](#)』の「[Relationship Prefetching](#)」を参照してください。

## WSIT 統合の詳細について

WSIT ステータスの詳細は、[WSIT Status Notes \(https://wsit.dev.java.net/source/browse/\\*checkout\\*/wsit/wsit/status-notes/status-notes-1-0-1-FCS.html\)](https://wsit.dev.java.net/source/browse/*checkout*/wsit/wsit/status-notes/status-notes-1-0-1-FCS.html) ページを参照してください。Enterprise Server での WSIT の使用について、『[The WSIT Tutorial](#)』も参照してください。

## 既知の問題点と制限事項

---

この章では、Sun GlassFish Enterprise Server 2.1 ソフトウェアに関する既知の問題とそれに関連する回避方法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。

- 28 ページの「管理」
- 37 ページの「Apache とロードバランサプラグイン」
- 38 ページの「アプリケーションクライアント」
- 39 ページの「付属の Sun JDBC ドライバ」
- 41 ページの「配備」
- 42 ページの「マニュアル」
- 42 ページの「EJB」
- 43 ページの「グローバル化」
- 43 ページの「高可用性」
- 53 ページの「インストール」
- 56 ページの「Java EE Tutorial」
- 57 ページの「Java Persistence」
- 57 ページの「ライフサイクル管理」
- 59 ページの「ロギング」
- 60 ページの「メッセージキュー」
- 61 ページの「監視」
- 62 ページの「パッケージ化」
- 62 ページの「サンプル」
- 63 ページの「セキュリティ」
- 65 ページの「アップグレードユーティリティー」
- 68 ページの「Web コンテナ」
- 73 ページの「Web サーバー」
- 73 ページの「Web サービス」

## 管理

ここでは、管理上の既知の問題とその解決方法を示します。

スタンドアロンのインスタンスがほかのインスタンスからファイルを取得することがある (6698604)

### 説明

一部の環境で、特定のインスタンスと同期する目的で DAS にインストールしたファイルが、実際には別のインスタンスに送信されます。

### 解決方法

現在のところ解決策はありません。

**start-cluster** コマンドからの起動メッセージが冗長である (6728317)

### 説明

asadmin start-cluster コマンドが、起動中に重要でないコンポーネントのエラーでも多量のメッセージを表示します。重要でない(クラスタ内のインスタンスに関連する)要素でエラーが発生した場合の、コマンド出力の例を示します。

```
./asadmin start-cluster --port 9898 cluster1
Please enter the admin user name>admin
Please enter the admin password>
The clustered instance, instance2, was successfully started.
error 0 [#|2008-07-17T14:58:16.496+0200|WARNING|sun-appserver9.1|javax.jms|
_ThreadID=10;_ThreadName=main;
_RequestID=90bbbe3a-d654-4480-b295-7e317d945a4a;|[C4003]:
Error occurred on connection creation [localhost:37676]. - cause:
java.net.ConnectException: Connection refused|#]
```

```
error 1 [#|2008-07-17T14:58:17.517+0200|WARNING|sun-appserver9.1|javax.jms|
_ThreadID=10;_ThreadName=main;
_RequestID=90bbbe3a-d654-4480-b295-7e317d945a4a;|[C4003]:
Error occurred on connection creation [localhost:37676]. - cause:
java.net.ConnectException: Connection refused|#]
```

```
error 2 [#|2008-07-17T14:58:30.596+0200|WARNING|sun-appserver9.1|
javax.enterprise.system.container.ejb|
_ThreadID=13;_ThreadName=pool-1-thread-4;TimerBean;
_RequestID=5954a044-df06-4a3e-902a-0c40b4b6cddb;
|EJB5108:Unable to initialize EJB Timer Service.
The likely cause is the database has not been
```

```
started or the timer database table has not been created.|#]
```

```
error 3 [#|2008-07-17T14:58:32.512+0200|WARNING|sun-appserver9.1|
javax.enterprise.resource.resourceadapter|_ThreadID=10;_ThreadName=main;
__CallFlowPool;_RequestID=90bbbe3a-d654-4480-b295-7e317d945a4a;|
RAR5005:Error in accessing XA resource with JNDI name [__CallFlowPool] for recovery|#]
```

The clustered instance, instance1, was successfully started.

```
error 0 [#|2008-07-17T14:58:21.117+0200|WARNING|sun-appserver9.1|
javax.enterprise.system.container.ejb|
_ThreadId=13;_ThreadName=pool-1-thread-4;TimerBean;
_RequestID=30827d9a-72ac-4854-b216-06494b6a9fb5;
|EJB5108:Unable to initialize EJB Timer Service. The likely cause is the database has
not been started or the timer database table has not been created.|#]
```

```
error 1 [#|2008-07-17T14:58:23.106+0200|WARNING|sun-appserver9.1|
javax.enterprise.resource.resourceadapter|
_ThreadId=10;_ThreadName=main;__CallFlowPool;
_RequestID=b41d76fa-0203-49f7-a2ae-83bf242d3e7a;
|RAR5005:Error in accessing XA resource with JNDI name [__CallFlowPool] for recovery|#]
```

Command start-cluster executed successfully.

## 解決方法

現在のところ解決策はありません。これらの例外メッセージは無視できます。

## IBM MQ に対して **Generic RA Resource Adapter** を配備できない (Issue 6605)

### 説明

IBM Message Queue 製品に対する Generic RA アダプタの配備が失敗します。server.policy ファイルで付与される権限は、次のようになります。

```
grant {
    permission java.util.logging.LoggingPermission "control";
    permission java.util.PropertyPermission "*", "read,write";
}
```

### 解決方法

server.policy ファイル中の権限を、次のように変更します。

```
grant codeBase
"file:${com.sun.aas.installRoot}/lib/install/applications/adminapp/-" {
```

```
    permission java.util.logging.LoggingPermission "control";
};
```

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

### 説明

デフォルトでは、*as-install/lib/package-appclient.xml* に、*asenv.conf* から参照される domain1 の AS\_ACC\_CONFIG 変数の値がハードコードされています。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。

1. 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。
2. domain1 を削除し、*as-install/lib/package-appclient.xml* 内のハードコードされた domain1 の値を、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サーバーの起動がサポートされない(6200011)

### 説明

サーバーでは J2SE 1.4.x または 5.0 以降を設定できます。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からサーバーを起動すると、仮想マシン (VM) 内に新たな jmx-connector サーバーが起動します。この場合、思わぬ結果として管理機能が悪影響を受け、管理コンソールやコマンド行インタフェース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は、サーバー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ようにしてください。

サーバーの起動時に、次に示すような行が `server.log` に記録されます。ここで指定されている `JMXService` の URL に接続し、資格を正常に指定した後、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.|#]
```

詳細は、『[Sun GlassFish Enterprise Server 2.1 管理ガイド](#)』を参照してください。

どの **Web** サービスのエンドポイント URL を使用しても、ロードバランサの設定ファイルが作成されない (6236544、6275436)

#### 説明

Web サービスの URL をエクスポートする EJB モジュールを含むアプリケーションを使用してロードバランサを設定しても、作成された `loadbalancer.xml` ファイルに、その Web サービスのコンテキストルートが存在しません。

#### 解決方法

1. `loadbalancer.xml` ファイルを編集して、作成されなかった Web モジュールを次のように追加します。

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

2. `context-root-name` 値を、EJB として公開された Web サービスのコンテキストルート名に置き換えます。

**Application Server** のマニュアルに `.asadmintruststore` ファイルが記述されていない (6315957)

#### 説明

Enterprise Server のマニュアルに `.asadmintruststore` ファイルが記述されていない。このファイルがサーバー管理者の `home` ディレクトリに存在しないと、そのサーバー上にホストされている特定のアプリケーションをアップグレードしたときに重大なバグが発生する場合があります。

#### 解決方法

- 可能であれば、そのサーバーをインストールしたユーザーが `asadmin start-domain domain1` コマンドを実行してください。

- そのユーザーがこのコマンドを実行できない場合は、`.asadmintruststore`を、インストールしたユーザーの home ディレクトリから実行中のユーザーの home ディレクトリに移動またはコピーしてください。
- このファイルをインストールユーザーの home ディレクトリから実行中のユーザーの home ディレクトリに(コピーではなく)移動した場合、アップグレードまたはインストールしたユーザーのホームディレクトリ (Java ES では、通常 root) に `.asadmintruststore` ファイルが存在しなくなるため、バグ 6309079、6310428、および 6312869 で説明されているような、アプリケーションのアップグレードに関する問題が発生する可能性があります。

**JMS ブローカに到達するまでにタイムアウトになるため、クラスタ化されたインスタンスが起動できない (6523663)**

## 説明

Enterprise Server クラスインスタンスのデフォルト MQ 統合モードは LOCAL です。Enterprise Server がインストールされている場所の PATH が長い、つまり短い場合、クラスインスタンス起動時に `imqbrokersvc.exe` で障害が発生します。  
。`imqbrokersvc` のメモリー割り当てに問題があります。

## 解決方法

クラスインスタンスの JMS サービスタイプを、デフォルトの LOCAL から REMOTE に変更する必要があります。この設定では、すべてのインスタンスは DAS ブローカを指します。次の手順に従って、REMOTE モードでクラスタを設定してください。

---

注 - REMOTE モード使用時には、すべてのインスタンスが1つのブローカ (DAS) を使用しているため、Enterprise Server クラスタが起動してもブローカクラスタは作成されません。詳細は、<http://www.glassfishwiki.org/gfwiki/attach/OnePagersOrFunctionalSpecs/as-mq-integration-gfv2.txt> の第 4.1 節「iii. Auto-clustering」を参照してください。それによると、上記の機能は将来使用できなくなります。

---

## ▼ コマンド行の使用

始める前に ご使用の環境に合わせて、ポートおよびパスワードファイルを変更してください。次に示す手順では、クラスタ名が `racluster`、DAS 管理ポートが 5858、および DAS JMS ポートが 7676 になっています。

- 1 **JMS タイプを REMOTE に変更して、クラスタ設定を変更します。**

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

- 2 **DAS JMS** ホストに対応する **JMS** ホストを作成します。

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

- 3 デフォルトの **JMS** ホストが前のステップで作成した **DAS JMS** ホストになるように設定します。

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

## ▼ 管理 GUI の使用

- 1 「設定」->「*cluster-name-config*」->「Java メッセージサービス」->「**JMS** ホスト」の順に移動します。
- 2 「新規」をクリックして新規 **JMS** ホストを作成し、dashost という名前を付けます。
- 3 **DAS** の **JMS** サービスに対応する設定を入力します。デフォルト設定は次のとおりです。

- ホスト名: localhost
- ポート: 7676
- 管理者ユーザー: admin
- パスワード: admin

これらの設定をご使用の **DAS JMS** サービスに適した値に変更してください。

- 4 「Java メッセージサービス」タブに戻って、**JMS** サービスタイプを **REMOTE** に変更します (デフォルトは **LOCAL**)。
- 5 default-jms-host ドロップダウンリストから dashost を選択します。
- 6 変更を保存してから、ノードエージェントまたはクラスタを起動します。

**Netscape 8.1.3**、**Mozilla 1.7**、および **Safari 2.0.4** ブラウザで **jmaki** チャートが表示できない (**6543014**)

### 説明

「ログ統計の監視」ページから一部のサポートされていないブラウザを使用してチャートを表示しようとすると、次のエラーがスローされます。

```
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
```

## 解決方法

サポートされているブラウザを使用します。Enterprise Server 2.1 でサポートされているブラウザについては、[18 ページの「ブラウザ」](#)を参照してください。

AS のメジャーリリースのたびに、デフォルトポートが変わっている (6566481)

## 説明

Enterprise Server の過去 3 回のメジャーリリースのたびに、デフォルト管理ポートが変わっていました。具体的には、7.x、8.x、および 9.x のデフォルト管理ポートは次のようになっています。

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

## 解決方法

これはバグではありませんが、このことに注意してください。デフォルト管理ポートは推奨値に過ぎません。今後の Enterprise Server リリースでは、デフォルトに 4848 ポートが継続して使用されます。

AIX でカスタムマスターパスワードを使用すると create-domain コマンドが失敗する (6628170)

## 説明

AIX オペレーティングシステムで、カスタムマスターパスワードを使用してドメインを作成しようとする、次のエラーが発生して失敗します。

```
keytool error (likely untranslated): java.lang.NullPointerException
Enter keystore password: New keystore password:
```

## ▼ 解決方法:(AIX) カスタムマスターパスワードを使用してドメインを作成する

---

注-次に示す手順では、各手順に必須のオプションのみ指定されています。コマンドにそれ以外のオプションを必要とする場合は、それらのオプションをコマンドに指定してください。Enterprise Server コマンドの詳細は、[『Sun GlassFish Enterprise Server 2.1 Reference Manual』](#)を参照してください。

---

- 1 次のコード行を含むシェルスクリプトを作成します。

```
#!/bin/sh
changeKeystorePass() {
    keytool -storepasswd -keystore ${KEYSTORE} -storepass ${OLD} -new ${NEW}
```

```

}
changeTruststorePass() {
    keytool -storepasswd -keystore ${TRUSTSTORE} -storepass ${OLD} -new ${NEW}
}
changeKeyPass() {
    keytool -keypasswd -alias slas -keystore ${KEYSTORE} -storepass ${NEW} -keypass ${OLD} -new ${NEW}
}
changeDomainPasswordEntry() {
    keytool -storepasswd -storetype JCEKS -keystore ${DOMAINPASSWORDS} -storepass ${OLD} -new ${NEW}
}
deleteMasterPasswordFile() {
    if [ -f ${DOMAIN_PATH}/master-password ] ; then
        echo Deleting ${DOMAIN_PATH}/master-password
        rm -f ${DOMAIN_PATH}/master-password
    fi
}
DOMAIN_PATH=$1
OLD=$2
NEW=$3
if [ $# != 3 ] ; then
    echo Usage: $0 domain-path old-master-pass new-master-pass
    exit 1
fi
echo Processing ...
if [ ! -f ${DOMAIN_PATH}/config/domain.xml ] ; then
    echo "Domain with folder ${DOMAIN_PATH} does not exist, create it first"
    exit 2
else
    KEYSTORE=${DOMAIN_PATH}/config/keystore.jks
    TRUSTSTORE=${DOMAIN_PATH}/config/cacerts.jks
    DOMAINPASSWORDS=${DOMAIN_PATH}/config/domain-passwords
    changeKeystorePass
    changeTruststorePass
    changeKeyPass
    changeDomainPasswordEntry
    deleteMasterPasswordFile
fi

```

- 2 デフォルトのマスターパスワードを指定して、ドメインを作成します。

```

admin create-domain --adminport aminportno --portbase portbase domain-name
Please enter the admin user name>admin-user
Please enter the admin password>admin-user-password
Please enter the admin password again>admin-user-password
Please enter the master password [Enter to accept the default]:>
Please enter the master password again [Enter to accept the default]:>

```

デフォルトのマスターパスワードは `changeit` です。

- 3 作成したドメインのマスターパスワードを変更します。  
マスターパスワードを変更するには、[手順1](#)で作成したスクリプトを実行します。  
*script-name domain-path old-password new-password*
- 4 [手順2](#)で作成したドメインを起動します。  
**asadmin start-domain domain-name**  
ドメインにはカスタムマスターパスワードが設定されているため、マスターパスワードの入力を求めるプロンプトが表示されます。
- 5 プロンプトに応じて、新しいマスターパスワードを入力します。
- 6 クラスタをサポートするように設定されているドメインの場合、ノードエージェントを作成して起動します。
  - a. [手順2](#)で作成したドメインにノードエージェントを作成します。  
**asadmin create-node-agent --port portno --user admin-user**
  - b. [手順a](#)で作成したノードエージェントを起動します。  
**asadmin start-node-agent**  
ドメインにはカスタムマスターパスワードが設定されているため、マスターパスワードの入力を求めるプロンプトが表示されます。
  - c. プロンプトに応じて、新しいマスターパスワードを入力します。

参照 次の Enterprise Server マニュアルページ

- [create-domain\(1\)](#)
- [create-node-agent\(1\)](#)
- [start-domain\(1\)](#)
- [start-node-agent\(1\)](#)

#### AIX: 0403-027 パラメータリストが長すぎる (6625591)

##### 説明

AIX オペレーティングシステムで、一部の OS に関連する操作で次のエラーが発生して失敗します。

```
0403-027 The parameter list is too long
```

OS に関連する操作の例としては、アプリケーションの配備やアプリケーションクライアントコンテナの実行などがあります。

通常、この問題は CLASSPATH 環境変数のファイルパスが長いために発生します。

## 解決方法

次の解決方法のいずれかを使用します。

- コマンド行の最大長を増加します。詳細については、37 ページの「(AIX) コマンド行の最大長を増加する」を参照してください。
- `xargs` コマンドを使用して、引数リストを構築し、コマンドを開始します。`xargs` コマンドによって、コマンド行の最大長を超えるコマンドを実行できます。

## ▼ (AIX) コマンド行の最大長を増加する

`ncargs` 属性によって、環境変数を含むコマンド行の最大長を決定します。AIX オペレーションシステムでは、`ncargs` 属性のデフォルト値は4ブロック (4K バイト単位) です。Enterprise Server コマンドがコマンド行の最大長を超えないようにするには、この値を16ブロック (4K バイト単位) に増加します。

---

注 - `ncargs` 属性の値を変更しても、デーモンの再起動やリフレッシュは不要です。

---

- 1 `ncargs` 属性の値を確認します。  

```
lsattr -EH -l sys0 | grep ncargs
```
- 2 `ncargs` 属性の値が16ブロック (4K バイト単位) よりも小さい場合、値を16に増加します。  

```
chdev -l sys0 -a ncargs=16
```

## Apache とロードバランサプラグイン

ここでは、Apache Web Server およびロードバランサプラグインに関する既知の問題と、それに関連する解決法を示します。

**Web サービスとして EJB とともに配備した Java EE アプリケーションが、負荷分散プラグインで収集されない (Issue 685)**

### 説明

WAR (または EAR) をサーブレットベースの Web サービスとともに配備すると、HTTP ロードバランサは Web サービスに関する情報で更新されます。EJB ベースの Web サービスを配備する場合、HTTP ロードバランサの設定は新しいオブジェクトを反映するように更新されません。

### 解決方法

これらのコンテキストルートを、ロードバランサの設定ファイル (`loadbalancer.xml`) に手動で追加します。ただし、ロードバランサ設定の動的再構成 (auto-apply 機能を使用) により、以前の手動による編集が失われる場合があります。

DAS で auto-apply 機能を無効にしてから、手動エクスポート機能を使用し、ロードバランサ設定を編集して Web サーバーに適用します。

『**High-Availability Administration Guide**』に、**Apache 2.0**で証明書を使用するための手順が記載されていない(6307976)

Apache のセキュリティを実行するには、証明書を使用する必要があります。認証局から証明書を取得する手順については、[modssl FAQ](http://www.modssl.org/docs/2.8/ssl_faq.html#ToC24) ([http://www.modssl.org/docs/2.8/ssl\\_faq.html#ToC24](http://www.modssl.org/docs/2.8/ssl_faq.html#ToC24)) で証明書に関する情報を参照してください。

**Apache Web Server** をルートとして起動する必要がある (6308021)

Solaris では、Application Server がルートの下にインストールされている場合、Apache Web Server をルートとして起動する必要があります。Java Enterprise System は、ルートとしてインストールされます。Apache 2.0 の場合、ルートとして起動された後、Apache はユーザーが指定した別のユーザーに切り替えて動作します。そのユーザーは、/conf/httpd.conf ファイルで指定します。多くのシステムでは、ルートとして起動するには、httpd.conf ファイルを編集して正しいグループを指定する必要があります。次の行を置き換えます。

Group #-1

次の行に置き換えます。

Group nobody

ユーザーおよびグループの使用に関する詳細情報は、httpd.conf ファイルに記載されています。

## 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 (6193556)

説明

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。

### ACC が常に localhost:3700 に接続しようとする (6527987)

#### 説明

アプリケーションクライアントが常に localhost:3700 に接続しようとし、問題は、クライアントノードを呼び出す前に、一部のシステムプロパティーが読み取られていなければならないことにあります。

#### 解決方法

次をシステムプロパティーとして (JAVA\_CMD に -D を指定して) 設定します。これらを appclient コードで設定しないでください。

```
org.omg.CORBA.ORBInitialHost = server-instance-host
org.omg.CORBA.ORBInitialPort = server-instance-port
```

### sunpkcs11.jar がないため、ドメインを開始できません (6571044)

#### 説明

64 ビット Linux 上で実行している場合、ドメイン開始時に次の例外が発生します。問題は jdk1.5.0\_11/jre/lib/ext/ の下に sunpkcs11.jar がないことです。

#### 解決方法

これは 64 ビット Linux で既知の JDK バグであり、JDK 1.5.0\_13 で修正されます。

ASQuickStartup が SelectionKey の代わりに **NULL** を返して、SocketChannel.keyFor() をブレイクする (**Issue Tracker 3027**)

#### 説明

複数のセレクタに SocketChannel が登録されている場合、socketChannel.keyFor(lastRegisteredSelector) を実行すると SelectionKey の代わりに NULL が返されます。

#### 解決方法

これは JDK バグ 6562829 に関連しており、6.0 U3 で修正される予定です。回避方法は Enterprise Server 2.1 に組み込まれており、keyFor API を読み込む前はセレクタがラップ解除されています。これにより、JDK バグが修正されるまで、keyFor の正常な動作が可能になっています。

## 付属の Sun JDBC ドライバ

ここでは、Sun の JDBC ドライバに関する既知の問題とその解決方法を示します。

## PreparedStatement エラーが発生する (6170432)

### 説明 1

1 つのトランザクションで 3000 を超える PreparedStatement オブジェクトを生成する場合、DB2 では次のエラーが発生する可能性があります。

```
[sunm][DB2 JDBC Driver] No more available statements. Please recreate your package with a larger dynamicSections value.
```

### 解決法 1

次のプロパティを接続プール定義に追加して、ドライバが DB2 パッケージをより大きな動的セクション値に再バインドするようにします。

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

接続プールの設定の詳細は、『[Sun GlassFish Enterprise Server 2.1 管理ガイド](#)』を参照してください。

### 説明 2

前述の PreparedStatement エラーに関連して、次のエラーメッセージがスローされることがあります。

```
[sunm][DB2 JDBC Driver][DB2]Virtual storage or database resource is not available.
```

### 解決法 2

DB2 サーバー設定パラメータ `APPLHEAPSZ` の値を増やします。適度な値は 4096 です。

### 説明 3

遮断レベル `TRANSACTION_SERIALIZABLE`。アプリケーションが遮断レベル `TRANSACTION_SERIALIZABLE` を採用し、前述したパラメータの 1 つを使用している場合、そのアプリケーションは接続を取得するときにハングアップすることがあります。

### 解決法 3

希望の遮断レベルを接続に対して設定するには、対応する接続プールをその遮断レベルで作成する必要があります。手順については、『[Sun GlassFish Enterprise Server 2.1 管理ガイド](#)』を参照してください。

### マシンのリポート後またはサーバーの起動後に、Java DB が起動しない (6515124)

#### 説明

ホストシステムまたは Solaris ゾーンのリポート後、または Enterprise Server 起動後に、付属の Java DB データベースが自動的に再起動しません。これはバグではなく、付属または他社製のアプリケーションで所定の動作です。問題は、Enterprise Server インスタンスの前に Java DB を起動する必要があるということです。

#### 解決方法

ホストマシンまたは Solaris ゾーンのリポート後、必ず Enterprise Server が開始する前に Java DB が起動するようにしてください。一例として、次のようにします。

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

asadmin コマンドのオプションの詳細は、『[Sun GlassFish Enterprise Server 2.1 クイックスタートガイド](#)』の「[管理ツール](#)」を参照してください。

## 配備

### クラスタで自動配備に失敗することがある (6610527)

#### 説明

タイミングの問題によって、クラスタをサポートするように設定されているドメインで自動配備に失敗することがあります。この問題は、クラスタをサポートしないドメインでは発生しません。

#### 解決方法

次の解決方法のいずれかを使用します。

- 次のように自動配備を使用します。
  - 個々のアプリケーションを順次自動的に配備します。
  - 個々のアプリケーションの自動配備の間に遅延を設定します。
- 管理コンソール GUI またはコマンド行のいずれかを使用して、アプリケーションを手動で配備します。

### 付属の ANT によって java.lang.NoClassDefFoundError 例外がスローされる (6265624)

#### 説明

スレッド「main」で java.lang.NoClassDefFoundError: org/apache/tools/ant/launch/Launcher の例外がスローされます。

#### 解決方法

付属の ANT を Enterprise Server の外部で使用することはお勧めできません。

### アプリケーションに固有のクラスローダーが JSP のコンパイルで使用されない (6693246)

#### 説明

アプリケーションに固有のクラスローダー (appliers または --libraries) が、JSP のコンパイルで使用されません。結果として、これらの JAR を参照する JSP がコンパイルされません。

#### 解決方法

現在のところ解決策はありません。

## マニュアル

ここでは、マニュアル上の既知の問題とその解決方法を示します。

### Javadoc に矛盾がある。(さまざまな ID)

いくつかの AMX インタフェースおよびメソッドについて、Javadoc が欠けているか間違っています。

- NumConnAcquired および NumConnReleased 統計情報の取得メソッドが ConnectorConnectionPoolStats および AltJDBCConnectionPoolStats から抜けている。これらの取得メソッドは、将来のリリースで getNumConnAcquired() および getNumConnReleased() として追加される予定。
- EJBCacheStats 内でメソッド getPassivationSuccesses()、getExpiredSessionsRemoved()、getPassivationErrors()、getPassivationErrors() を呼び出すと、例外がスローされる。これは将来のリリースで解決される予定。
- サーバーを起動したあと、すべての AMX MBeans が登録されて利用できるようになるまでに数秒を要することがある。将来のリリースでは、AMX MBeans が完全にロードされたことを確認できるようになる予定。
- 定数 XTypes.CONNNECTOR\_CONNECTION\_POOL\_MONITOR は、スペルに誤りがあります ("NNN")。これは将来のリリースで訂正される予定。

## EJB

### リソースインジェクションが HandlerChain で動作しない (6750245)

#### 説明

EJB の初期化順が原因で、リソースインジェクションが HandlerChain で機能しません。

#### 解決方法

現在のところ解決策はありません。

## グローバル化

### SuSE 10 で CJK サポートを必要とする (6416177)

#### 説明

SuSE 10 では CJK ロケールがインストールされている場合でも CJK のグリフが全て四角形として表示されます。

#### Solution

SuSE 10 上では、ソフトウェアインストーラーと CLI コマンドを C ロケールで実行して下さい。

### Sun Online Account の情報を入力する際、ラテン文字のみ使用可能 (6704238)

#### 説明

ユーザーがインストールの最後に新規 Sun Online Account を作成する上で、氏名を入力する際、マルチバイト文字を入力することができません。

#### Solution

Sun Online アカウントを作成する際は、アルファベットで氏名を入力して下さい

## 高可用性

ここでは、高可用性データベース (HADB) に関する既知の問題とその解決方法を示します。

### ロードバランサプラグインの健全性検査により、バックグラウンド(負荷)で多数の接続と切断が生成される (6453946)

#### 説明

ロードバランサプラグインの健全性検査により、バックグラウンド(負荷)で多数の接続と切断が生成されます。健全性検査の目的で、runDaemonMonitor スレッドが Application Server のリスナーごとに接続と切断を実行します。これにより、Enterprise Server で接続が飽和する場合があります。

#### 解決方法

新しい属性の `monitor-interval-in-seconds` を、`loadbalancer.xml` ファイルで使用できます。この属性を使用すると、ロードバランサプラグインに多数のリスナーが設定されている状況で、接続と切断のイベントの間に一時停止を挿入できます。一時停止のデフォルト値は 0 です。

### ダブルネットワークでの HADB 設定 (ID なし)

2つのサブネット上にダブルネットワークで設定された HADB は、Solaris SPARC 上では正常に動作します。しかし、一部のハードウェアプラットフォームでのオペレーティングシステムまたはネットワークドライバの問題が原因で、Solaris x86 および Linux プラットフォームではダブルネットワークを適切に処理できない場合があります。これにより、HADB について次の問題が発生します。

- Linux では、メッセージ送信の際に HADB プロセスがブロックされることがある。これにより、HADB ノードが再起動し、ネットワークパーティションが発生する。
- Solaris x86 では、ネットワーク障害が発生した場合、もう一方のネットワークインタフェースへの切り替えを妨げる問題が発生することがある。この問題は常に発生するとは限らないため、ネットワークが1つしかないよりも2つあった方が安全である。この問題は、Solaris 10 で部分的に解決されている。
- トランキングがサポートされない。
- Microsoft Windows 2003 では、HADB はダブルネットワークをサポートしていない (ID 5103186)。

### HADB データベースの作成が失敗する (ID なし)

#### 説明

新しいデータベースを作成すると、使用可能な共有メモリーセグメントが少なすぎるといふ、次のエラーで失敗することがあります。

```
HADB-E-21054: System resource is unavailable: HADB-S-05512: Attaching shared memory segment with key "xxxxx" failed, OS status=24 OS error message: Too many open files.
```

#### 解決方法

共有メモリーが設定されており、その設定が機能していることを確認します。特に、Solaris 8 では、`/etc/system` ファイルを調べて、変数 `shmsys:shminfo_shmseg` の値がホストあたりのノード数の6倍以上になっていることを確認します。

リソース (ディスクおよびメモリースペース) が利用可能かどうかを `hadbm set` がチェックしない (5091280)

#### 説明

`hadbm set` を使用してデバイスまたはバッファのサイズを増やす場合、管理システムは、データベースの作成やノードの追加の際にはリソースが利用可能かどうかをチェックしますが、デバイスまたはメインメモリーのバッファサイズを変更するときには利用可能なリソースが十分にあるかどうかをチェックしません。

## 解決方法

設定属性 `devicesize` または `buffersize` を増やす前に、すべてのホスト上にディスクおよびメモリの空きスペースが十分であることを確認してください。

`packagepath` の混在パスがサポートされない (5091349)

## 説明

同一のソフトウェアパッケージを、同じ名前で別のホストの別の位置で登録することはできません。次に例を示します。

```
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.
```

## 解決方法

HADB は、データベースクラスタ内のノードをまたがる混在パスをサポートしません。HADB サーバーのインストールディレクトリ (`---packagepath`) は、すべての参加ホストについて同一にしてください。

`hadbm createdomain` が失敗することがある (6173886、6253132)

## 説明

複数のネットワークインタフェースを備えたホスト上で管理エージェントを実行している場合に、すべてのネットワークインタフェースが同じサブネット上に存在しないと、`createdomain` コマンドが失敗することがあります。

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

管理エージェントは、特に設定されていないかぎり、UDP マルチキャスト用の「最初の」インタフェース (`java.net.NetworkInterface.getNetworkInterfaces()` の結果で定義されている最初のインタフェース) を使用します。

## 解決方法

もっとも良い解決法は、使用するサブネットを管理エージェントに通知することです。たとえば、設定ファイル内の `ma.server.mainternal.interfaces` を `ma.server.mainternal.interfaces=10.11.100.0` に設定します。あるいは、サブネット間のルーターを、マルチキャスト packets をルーティングするように設定することもできます。このとき、管理エージェントはマルチキャストアドレス 228.8.8.8 を使用します。

管理エージェントの新しい設定を再試行する前に、管理エージェントリポジトリのクリーンアップが必要になる場合があります。ドメイン内のすべてのエージェントを停止し、管理エージェント設定ファイル中の `repository.dr.path` で指定したりポジトリディレクトリにある、すべてのファイルとディレクトリを削除します。この操作は、新しい設定ファイルを使用してエージェントを再起動する前に、すべてのホスト上で実行する必要があります。

### **HADBの起動、停止、および再設定が失敗またはハングアップすることがある (ID 6230792、6230415)**

#### 説明

Solaris 10 Opteron では、`hadbm` コマンドを使用して HADB を起動、停止、または再設定すると、次のいずれかのエラーで失敗またはハングアップする場合があります。

```
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.
```

このエラーは、`clu_noman_srv` プロセスが使用するファイル (`nomandev`) への読み取り/書き込みに不整合があった場合に発生することがあります。この問題は、HADB 履歴ファイルで次のメッセージを検索することにより検出できます。

```
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.
```

#### 解決方法

問題を手動で再現できていないため、次の回避策はまだ検証されていません。ただし、影響を受けるノードに対してこのコマンドを実行すれば、問題は解決されます。

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

ノードのすべてのデバイスが再初期化されるわけではないことに注意してください。再初期化する前に、ノードの停止が必要になる場合があります。

管理エージェントが、例外「`IPV6_MULTICAST_IF failed`」で終了する (**6232140**)

#### 説明

複数の NIC カードが実装された、Solaris 8 を実行しているホスト上で起動されている場合、IPv6 と IPv4 が有効になったカードが混在していると、管理エージェントが例外「`IPV6_MULTICAST_IF failed`」で終了することがあります。

### 解決方法

環境変数 `JAVA_OPTIONS` を `-Djava.net.preferIPv4Stack=true` に設定します。次に例を示します。

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

あるいは、この問題が発生しない Solaris 9 以降を使用します。

### `clu_trans_srv` を中断できない (6249685)

#### 説明

64 ビットバージョンの Red Hat Enterprise Linux 3.0 のバグにより、非同期 I/O の実行時に `clu_trans_srv` プロセスが中断できないモードになります。この場合、`kill -9` が機能せず、オペレーティングシステムの再起動が必要になります。

### 解決方法

Red Hat Enterprise Linux 3.0 の 32 ビットバージョンを使用します。

### `hadbm` が大文字を含むパスワードをサポートしていない (6262824)

#### 説明

パスワードが `hadb` に格納されるときに、パスワード内の大文字は小文字に変換されます。

### 解決方法

大文字を含むパスワードは使用しないでください。

### **HADB Version 4.4.2.5** から **HADB Version 4.4.1.7** にダウングレードすると、`ma` が各種のエラーコードで失敗する (6265419)

#### 説明

以前の HADB バージョンにダウングレードすると、管理エージェントが各種のエラーコードで失敗する場合があります。

### 解決方法

HADB データベースのダウングレードは可能ですが、リポジトリオブジェクトが変更されている場合は管理エージェントをダウングレードできません。ダウングレードのあとも、最新の HADB バージョンの管理エージェントを使用し続ける必要があります。

インストールまたは削除を行なっても、シンボリックリンクが保持される (6271063)

#### 説明

HADB c パッケージ (Solaris: SUNWhadb、Linux: sun-hadb-c) バージョン <m.n.u-p> のインストールまたは削除に関しては、symlink /opt/SUNWhadb/<m> はいったん作成されると、そのあとは何も手を加えられません。そのため、切り離された symlink が存在することがあります。

#### 解決方法

使用中の場合を除き、インストールの前またはアンインストールの後に symlink を削除します。

大域ゾーンとローカルゾーンの管理エージェントが干渉することがある (6273681)

#### 説明

Solaris 10 では、大域ゾーンで ma-initd スクリプトを使用して管理エージェントを停止すると、ローカルゾーンの管理エージェントも停止されます。

#### 解決方法

管理エージェントを大域ゾーンとローカルゾーンの両方にインストールしないでください。

セッションオブジェクトがタイムアウトし、MA で削除されたとき、hadbm/ma はより適切なエラーメッセージを出力するべきである (6275103)

#### 説明

場合によっては、サーバー上のリソース競合の問題によって管理クライアントが切断されることがあります。再接続時、「hadbm: Error 22184: A password is required to connect to the management agent」という紛らわしいエラーメッセージが返されることがあります。

#### 解決方法

場合によっては、サーバー上のリソース競合の問題によって管理クライアントが切断されることがあります。再接続時、「hadbm: Error 22184: A password is required to connect to the management agent」という紛らわしいエラーメッセージが返されることがあります。

サーバー上にリソースに関する問題があるかどうかを確認し、適切な処置 (たとえば、リソースの追加) を取ってから、操作を再実行します。

## ルート以外のユーザーが HADB を管理できない (6275319)

### 説明

Java Enterprise System を使用して (ルートとして) インストールすると、ルート以外のユーザーは HADB を管理できなくなります。

### 解決方法

HADB を管理するには、常にルートとしてログインします。

## 管理エージェントは特殊用途のインタフェースを使用するべきではない (6293912)

### 説明

0.0.0.0 のような IP アドレスを含む特殊用途のインタフェースを、管理エージェント内の HADB ノードが使用する有効なインタフェースとして登録するべきではありません。このようなインタフェースを登録すると、IP アドレスの代わりにホスト名を使用して `hadbm create` コマンドを発行するユーザーによってこのインタフェース上に HADB ノードが設定された場合に、問題が発生する場合があります。その場合、これらのノードは通信できなくなり、`create` コマンドはハングアップします。

### 解決方法

複数のインタフェースを備えたホスト上で `hadbm create` を使用する場合は、DDN 形式を使用して IP アドレスを常に明示的に指定します。

## Windows 上で再構築が失敗する (6291562)

### 説明

Windows プラットフォームでは、特定の設定および負荷の下で、オペレーティングシステム内で多数の再構築の失敗が発生する場合があります。この問題は、20 を超えるノードが設定されている状況で、複数のテーブルスキャン (`select *`) を並列に実行している場合に発生しています。症状としては、トランザクションが頻繁に中止され、修復またはリカバリの完了に長い時間がかかるため、システムのさまざまな部分で頻繁なタイムアウトが発生していることが考えられます。

### 解決方法

この問題を修正するには、Windows レジストリ変数

`HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters` をデフォルトの 100 より大きい値に設定します。この値を 0x1000 (4096) に増やすことをお勧めします。詳細は、Microsoft サポートページの記事 [811003](http://support.microsoft.com/default.aspx?scid=kb;en-us;811003)

(<http://support.microsoft.com/default.aspx?scid=kb;en-us;811003>) を参照してください。

パスが / である別の **cookie** をブラウザが取ると、セッション状態が保持されない  
(6553415)

#### 説明

cookie のパスが「/」と等価であった場合、持続タイプとしてインメモリーレプリケーションを使用する「/」以外のコンテキストルートに配備された高可用性 Web アプリケーションの cookie と干渉し、高可用性 Web アプリケーションが HTTP セッション状態をまったく保持できなくなります。この状況になる可能性のある 1 つの一般的なシナリオは、同じブラウザを使用して「/」に配備されている管理 GUI と高可用性 Web アプリケーションとの両方にアクセスしている場合です。

#### 解決方法

別のブラウザから「/」に配備されている Web アプリケーションにアクセスします。

*as-install/lib* の下に SASL32.DLL と ZLIB.DLL がいないため、ロードバランサが IIS 6 と連動しない (6572184)

#### 説明

SASL32.DLL および ZLIB.DLL は、Windows IIS 6 でロードバランサを使用する際に必要です。これらのファイルが現在 *as-install/lib* にありません。

#### 解決方法

2 つの DLL ファイルを手動で *as-install/lib* にコピーします。これらのファイルは次の場所からダウンロードできます。

<http://download.java.net/javaee5/external/OS/aslb/jars/>

OS は希望するプラットフォームを表し、次の値のいずれかになります。

- SunOS
- SunOS\_X86
- Linux
- WINNT

大域ゾーンで DAS 作成/起動および HA パッケージ伝播に問題がある (6573511)

#### 説明

大域ゾーンで Enterprise Server を高可用性パッケージとともにインストールまたはアンインストールすると、次の 2 つの問題が起きます。

1. HA パッケージがすべてのゾーンにインストールされます。これは望ましくありません。
2. アンインストール時に、HA、MQ、JDK パッケージがすべてのゾーンから削除されます。これも望ましくありません。

ルートローカルゾーンからインストールまたはアンインストールする場合には、この問題は起きません。

### 解決方法

グローバルゾーンではなくローカルルートゾーンからインストールまたはアンインストールを実行します。

「/」に配備された高可用性 Web アプリケーションがインメモリーレプリケーションで複製された HTTP セッションを再開できない (Issue Tracker 2972)

### 説明

「/」に配備された高可用性 Web アプリケーションは、持続タイプとしてインメモリーレプリケーションを使用している場合、HTTP セッションを保持できません。

### 解決方法

インメモリーレプリケーションを持続タイプとして使用する高可用性 Web アプリケーションを「/」以外のコンテキストルートに配備します。このような Web アプリケーションを「/」で利用できるようにするには、Web アプリケーションが配備されている仮想サーバーの default-web-module として指定します。

AS ロードバランサインストーラが /usr/lib/mps パスを apachectl LD\_LIBRARY\_PATH に書き込まないため、Apache SSL を起動できない (6591878)

### 説明

Solaris 上で Apache 用の Enterprise Server ロードバランサをインストール中に、インストーラは apachectl スクリプトの LD\_LIBRARY\_PATH を更新します。しかし、インストーラは /usr/lib/mps パスを正しく書き込みません。Solaris の場合、LD\_LIBRARY\_PATH にこのパスがないと、Apache セキュリティーインスタンスは開始しません。

### 解決方法

この問題は Solaris プラットフォームでのみ起きます。この問題を回避するために、LD\_LIBRARY\_PATH に /opt/SUNWappserver/appserver/lib/lbplugin/lib を追加します。

インスタンスまたはクラスタの「ロードバランサの有効化/無効化」が正しい状態を示すようにする必要がある (6595113)

### 説明

domain.xml に保存されている内容にかかわらず、「クラスタ/インスタンス」一般情報ページで「ロードバランサの有効化」ボタンが常にオンになっています。

## 解決方法

- クラスタ化されたインスタンスの場合、「インスタンス」タブを選択し、表のプルダウンから「休止」アクションをクリックします。
- スタンドアロンインスタンスの場合、インスタンスが実行中であることを確認してから、インスタンスの「一般」画面で「休止」ボタンをクリックします。

**AS9.1 EE IFR b58f/JES5 UR1: 不完全な HA が検出されたため、レジストリサーバーをインストールできない(6602508)**

## 説明

「(Solaris のみ)」HADB 導入済みの SPARC Solaris 10 に Enterprise Server 2.1 をインストールしたあと、Enterprise Server を開始して JES 5 UR 1 をレジストリサーバーとともにインストールしようとする、次のエラーを受け取ることがあります。

```
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.
```

## 解決方法

Enterprise Server 2.1 IFR が Solaris マシンにインストールされた状態で、JES 5 UR1 からレジストリサーバーをインストールすることは不可能です。次の JES5 UR1 配布ディレクトリから、pkgadd コマンドを使用して、手動でレジストリサーバーパッケージをインストールする必要があります。

```
path/OS/Products/registry-svr/Packages
```

**Internet Explorer 6.0/7.0 ブラウザ固有: ロードバランサ設定ファイルをエクスポートすると、エラーがスローされる (6516068)**

## 説明

「(Internet Explorer 6 および 7 のみ)」ロードバランサ設定ファイル (loadbalancer.xml) を Internet Explorer 6 または 7 からエクスポートしようとする、sun-loadbalancer\_1\_2.dtd DTD ファイルが見つからないことを示すエラーメッセージがブラウザに表示されます。

## 解決方法

設定ファイルを保存するには、次の回避方法を使用します。

1. Internet Explorer の「ロードバランサ」ページで、「エクスポート」をクリックします。  
「XML page cannot be displayed」メッセージが表示されます。
2. エラーフレームをクリックしてから、Internet Explorer で「ファイル」->「名前を付けて保存」を選択します。

3. loadbalancer.xml ファイルを選択したディレクトリに保存します。

## インストール

ここでは、インストール上の既知の問題とその解決方法を示します。

### Enterprise Server インストーラが Linux でクラッシュする (6739013)

#### 説明

この問題は、環境変数 `MALLOC_CHECK_` に 2 を設定して Linux を実行しているシステムで観察されています。

#### 解決方法

環境変数 `MALLOC_CHECK_` を 0 に設定してください。 `export` コマンドを次のように実行します。

```
export MALLOC_CHECK_
```

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)

#### 説明

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、Linux Red Hat ディストリビューションでも見られます。

インストールプログラムの最後の画面で「完了」ボタンをクリックすると、インストールプログラムは製品の「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ます。

#### 解決方法

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、インストールプログラムを終了します。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://as-install/docs-ee/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。

**Windows** では、インストール中に `imq` ディレクトリを作成する必要がある (6199697)

#### 説明

Windows では、インストールした直後に、ディレクトリ `drive:\as\domains\domain1\imq` が存在しない旨のメッセージを出力して Message Queue ブローカが起動に失敗します。

`domain1` を起動してからブローカを起動した場合には、Application Server によってディレクトリが作成され、この問題は発生しません。

#### 解決方法

1. ブローカを作成する前に `var_home_dir_location` を作成します。次のようにします。

```
$imqbrokerd -varhome var_home_dir_location
```

次に例を示します。

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

アンインストール時に `productregistry` ファイルが正しく更新されない。サイレントモードを使用してインストールできない (6571598)

#### 説明

Enterprise Server の `productregistry` ファイルに共用コンポーネント設定が含まれている場合、Enterprise Server のアンインストール処理で `productregistry` ファイルが正しく更新されないため、`productregistry` ファイルを名前変更または削除しないかぎり、それ以降のインストールでサイレントモードを使用できなくなります。  
。`productregistry` ファイル内の共用コンポーネントエントリを変更せずに残しておくことは意図的ですが、そのためにそれ以後のインストールで混乱が生じます。

#### 解決方法

アンインストールが正常に完了したことがアンインストールログファイルによって報告されたあとに、続けてインストールを実行する前に `productregistry` ファイルを削除します。以前のアンインストールが正常に完了したことを確認するには、`as-install` に `appserv_uninstall.class` ファイルがあるかどうかを調べます。アンインストールが正常に完了した場合には、このファイルはありません。

---

注-インストールが正常に完了していない場合は、`productregistry` ファイルを削除しないでください。

---

`productregistry` ファイルは、Solaris では `/var/sadm/install` に、Linux では `/var/tmp` にあります。

## IFR: MQパッケージの問題で、ASを疎ローカルゾーンにインストールできない(6555578)

### 説明

疎ローカルゾーンに Enterprise Server をインストールする場合、Message Queue (MQ) が先にインストールされていないとインストールが失敗します。インストーラは MQ をインストールしようとしませんが、インストール全体が失敗します。

### 解決方法

Enterprise Server を疎ローカルゾーンにインストールする前に、MQ をグローバルゾーンに手動でインストールする必要があります。この問題には2つの回避方法があります。

1. 最新の MQ パッケージを入手するために、Enterprise Server 2.1 IFR インストールが収録されている同一のメディアから、MQ 2.1 を手動でグローバルゾーンにインストールします。

- a. ご使用のプラットフォームに対応するインストーラを使用してください。

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

- b. 圧縮ファイルを解凍して、インストーラを実行します。  
インストーラは mq4\_1-installer ディレクトリにあります。

2. IFR インストールのすべてのコンポーネントをグローバルゾーンにインストールします。この処理で、GZ の MQ のバージョンをチェックし、必要に応じて Enterprise Server 2.1 IFR に付属のバージョンにアップグレードします。サンプルアプリケーションコンポーネントを選択してインストールするだけでも、MQ は IFR バージョンにアップグレードされます。

- a. グローバルゾーンで Enterprise Server インストールを実行しますが、サンプルコンポーネントのみを選択します。

サンプルコンポーネントインストールでは、MQ と Enterprise Server 共用コンポーネントもすべてのゾーンにインストールされます。

- b. Enterprise Server インストールを再度実行しますが、今回は疎ローカルゾーンで実行します。

何の問題もなくインストールが完了するはずです。

## IFR インストールの -console プロンプト「Do you want to upgrade from previous Application?」を削除する必要がある (6592454)

### 説明

Enterprise Server 2.1 IFR インストーラに -console オプション(コマンド行モード)を指定して実行すると、次のプロンプトが表示されます。

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

残念ながら、IFR インストーラではこのようなアップグレードをサポートしていないため、このプロンプトは誤りです。プロンプトに「yes」と応答すると、インストーラは正常に続行しますが、アップグレードされないばかりか、インストーラが完全に実行されたことが表示されません。

### 解決方法

Enterprise Server インストールのアップグレードを希望する場合は、アップグレードツールを使用してください。

アップグレードのあと、ドメインの起動時に次の例外がログに記録される (6774663)

### 説明

次の例外がスローされます。

```
#
```

```
|2008-11-19T01:44:37.422+0530|SEVERE|sun-appserver9.1|org.apache.catalina.session.ManagerB  
uestID=cc0ddf54-a42e-400a-9788-e30d79a25d88;|PWC2768: IOException while loading  
persisted sessions: java.io.InvalidClassException: org.apache  
.catalina.session.StandardSession; local class incompatible: stream classdesc  
serialVersionUID = 8647852380089530442, local class serialVersi onUID =  
-8515037662877107054 java.io.InvalidClassException:  
org.apache.catalina.session.StandardSession; local class incompatible: stream  
classdesc serialVersionUID = 864 7852380089530442, local class serialVersionUID  
= -8515037662877107054..... .....
```

### 解決方法

アップグレードを完了して、アップグレードしたドメインを起動したあと、これらの例外は発生しても無視できます。

## Java EE Tutorial

管理コンソールを使用して何らかのリソースを作成する場合には、「ターゲット」タブを使用してサーバーをターゲットとして指定します。コマンド行または asant ターゲットを使用する場合、サーバーがデフォルトのターゲットになるため、特別な処置は必要ありません。

## Java Persistence

**TopLink** は Collection フィールド/プロパティがクローン可能であることを想定する (**Issue Tracker 556**)

### 説明

Object[] から Collection への変換に java.util.Arrays.asList() API を使用している場合、JDK はクローン可能ではない java.util.ArrayList の実装を返します。結果として次の例外が発生します。

```
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
```

この問題は、[https://glassfish.dev.java.net/issues/show\\_bug.cgi?id=556](https://glassfish.dev.java.net/issues/show_bug.cgi?id=556) で説明されています。

### 解決方法

別のコレクションをそのコンストラクタを使用して作成します。一例として、次のようにします。

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

GenerationType.IDENTITY および SyBase とともに使用する **DataDirect** ドライバ (**Issue Tracker 2431**)

### 説明

DataDirect ドライバを SyBase とともに使用すると、GenerationType.IDENTITY を使用するエンティティの挿入に失敗します。この失敗は、パラメータ化された PreparedStatement ごとに DataDirect ドライバがストアプロシージャを作成するために発生します。

### 解決方法

domain.xml ファイルで、対応するデータソースのプロパティ PrepareMethod=direct を設定します。

## 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 プロパティを設定すると set コマンドが失敗する (6193449)****説明**

ejb-timer-service プロパティの `minimum-delivery-interval` を `9000` に設定したあと、ejb-timer-service プロパティの `redelivery-interval-in-millis` を `7000` に設定しようとする、set コマンドが次のエラーにより失敗します。

```
[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` 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サーバー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。

**DAS 以外の config 内の JMS 物理送信先を一覧表示すると、エラーがスローされる (6532532)****説明**

`default-config` を使用して JMS 物理送信先を表示しようとする、エラーメッセージが表示されます。

## 解決方法

これは予想どおりの動作です。Enterprise Server 2.1 では、default-config は設定情報のテンプレートであるため、JMS 操作 (list や create など) を default-config に対して実行できません。しかし、これらの JMS 操作をクラスタまたはスタンドアロンインスタンスの設定に対して実行することは可能です。

**Win2003 のみ:** ページング不可能なプールでメモリーリークが発生し、TCP スタックおよび RichAccess テストが中断する (6575349)

## 説明

(Windows 2003 のみ) Windows 2003 では、リッチアクセス機能実行時にメモリーリークが発生します。Win32 ページング不可能プールが増大し続け、ついには TCP/IP スタック全体が停止してしまうために起きる問題です。一度この障害が発生すると、TCP/IP スタックは回復可能状態のままになり、TCP/IP スタックを復元するには Windows 2003 システムをリブートするしかありません。

## 回避策

この問題には 2 つの回避方法があります。

- domain.xml http-listener 属性の blocking-enabled="true" を設定して Grizzly プロックモードを使用するか、次の http-listener プロパティを追加します。

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

- Windows Vista または Windows XP を使用します。

## ロギング

この節では、ログに関する既知の問題とその解決方法を示します。

access, failure のデバッグ文を設定すると、サーバーの起動時にハングアップする (6180095)

## 説明

JVM の java.security.debug オプションを設定すると、サーバーインスタンスの起動がデッドロックで動かなくなります。たとえば、domain.xml で次の設定を行うと、この問題が発生します。

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

## 解決方法

現時点ではありません。このフラグは設定しないでください。

## メッセージキュー

ここでは、Java メッセージキューに関する既知の問題とその解決方法を示します。

### MQ ブローカが起動していないと、Enterprise Server が起動しない (6740797)

#### 説明

JMS を REMOTE モードの Enterprise Server として設定している場合、MQ ブローカが起動していないと起動に失敗します。

#### 解決方法

現在のところ解決策はありません。JMS を EMBEDDED モードで設定してください。

### タイミングに依存する特定の場合に、JMS 再接続が正常に完了しない (ID 6173308、6189645、6198481、6199510、6208728)

#### 説明

タイミングに依存する場面での再接続の失敗は、さまざまな問題によって引き起こされます。

#### 解決方法

これらの問題は、次の方法で回避できます。

- 関連するブローカを再起動する
- 関連するインスタンスを再起動する

### Linux 上にクラスタプロファイルがあると MQ ブローカが起動に失敗する (6524871)

#### 説明

ドメインを作成しクラスタプロファイルを Linux システム上に配置すると、`java.lang.OutOfMemoryError: Java heap space` エラーが発生する場合があります、MQ ブローカが起動しないためにサーバーインスタンスが再起動できないことがあります。この状況になったシステムが回復することはありません。問題は `/etc/hosts` ファイルの設定ミスです。特に、サーバーホスト名がループバックアドレス `127.0.0.1` を指している場合です。

#### 解決方法

設計上、ネットワークデバイスがループバックアドレスを指すように設定された状態で MQ ブローカクラスタが開始することはできません。これはバグではありません。回避方法は、Enterprise Server ホストの `/etc/hosts` ファイルが `127.0.0.1` を指さないようにすることです。

アップグレードの前に `imqjmsra.jar` を読み込む際、旧バージョンと新バージョンのクラスが一致しない (6740794)

#### 説明

サーバーは起動中に Message Queue のバージョンをチェックします。Message Queue のバージョンが正しくない場合、サーバーは `imqjmsra.jar` を使用してアップグレードを行います。このアップグレード用の JAR とそのクラスは、次に再起動されるまでサーバーから利用できません。この状況は、Message Queue が単独でアップグレードされる場合、または Application Server のみにパッチが適用される場合に発生します。この状況の副次的な影響として、サーバーが起動しない場合があります。

#### 解決方法

Message Queue と Enterprise Server の両方を同じパッチレベルに維持する必要があります。または、サーバーを再起動してください。

## 監視

ここでは、監視上の既知の問題とその解決方法を示します。

**HTTP サービスの監視統計には有益な情報を提示しないため、無視する必要があるものがある (6174518)**

#### 説明

HTTP サービスの特定要素の監視統計を表示しているときに、表示される一部の値が現在の値に対応していないか、常に 0 になります。特に、次の HTTP サービスの統計は、Enterprise Server に適用できる情報を示していません。これらの値は無視してください。

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

#### 解決方法

これらの監視情報は将来のリリースで削除され、より適切な情報で置き換えられる予定です。

管理 GUI から JNDI ブラウザを開くと、server.log に大量の例外がダンプされる (6591734)

説明

管理 GUI から JNDI ブラウザを開くと、多くの例外がスローされます。

解決方法

現時点ではありません。

## パッケージ化

この節では、Enterprise Server 2.1 製品のソフトウェアコードのパッケージ化に関する既知の問題とその解決方法を示します。

**AIX:monitor コマンドが AIX 上で機能しない (6655731)**

説明

libcliutil.so ライブラリファイルが Enterprise Server 2.1 にパッケージ化されていないため、AIX オペレーティングシステムでは monitor コマンドを実行できません。

### ▼ 解決方法: 欠落している libcliutil.so ライブラリファイルをインストールするには

- 1 <http://download.java.net/javaee5/external/AIX/appserv-native/jars/appserv-native-9.1.1-b16a.jar> から JAR ファイルをダウンロードします。
- 2 appserv-native-9.1.1-b16a.jar ファイルをダウンロードしたディレクトリに移動します。  
prompt% `cd destination-dir`
- 3 appserv-native-9.1.1-b16a.jar ファイルの内容を抽出します。  
prompt% `jar xf appserv-native-9.1.1-b16a.jar`
- 4 libcliutil.so ファイルを `as-install/lib` ディレクトリにコピーします。  
prompt% `cp libcliutil.so as-install/lib`

## サンプル

この節では、Enterprise Server 2.1 製品に付属するサンプルコードに関する既知の問題とその解決方法を示します。

アップグレード後に、**Enterprise Server** のサンプルと **JES5** ポータルサンプルが **derby** ポート 1527 で競合する (**6574563**)

#### 説明

Windows で、Enterprise Server 2.1 にアップグレードしたあと、そのサンプルと JES 5 ポータルサンプルが Derby ポート 1527 で競合します。特に、Enterprise Server 2.1 は 0.0.0.0:1527 (APP:APP 設定) で自動的に JavaDB を起動しますが、JES:5 ポータル JavaDB は hostnameIP:1527 (portal:portal 設定) へのバインドを希望します。

このバグは、JES5 のバグ 6472173 で確認済みの問題を表しています。バグ 6472173 の回避策は、<http://docs.sun.com> で『Sun Java Enterprise System 5 Installation Guide for Microsoft Windows』を参照してください。

#### 解決方法

次のコマンドを使用して Derby データベースを開始します。

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

## セキュリティ

ここでは、Enterprise Server と Web アプリケーションのセキュリティおよび証明書に関する既知の問題とその解決方法を示します。

負荷が非常に高い場合に **SSL** のシナリオで **OutOfMemory** エラーが発生する (**JDK 6 Issue 23**)

#### 説明

非常に負荷の高い状況で特定の SSL シナリオを実行すると、JDK6 Sun PKCS11 Provider の JDK バグ ([https://jdk6.dev.java.net/issues/show\\_bug.cgi?id=23](https://jdk6.dev.java.net/issues/show_bug.cgi?id=23) を参照) により、OutOfMemory エラーが発生する場合があります。

#### 解決方法

この問題が発生した場合は、JRE インストールの java.security ファイルから sun.security.pkcs11.SunPKCS11 プロバイダを削除してください。

**AIX:** サーバー側の証明書の検証でエラーが発生したために **WSS** 動的暗号化鍵のテストに失敗する (**6627379**)

#### 説明

AIX プラットフォームで、応答用の暗号化鍵を判別するための動的な暗号化に失敗します。失敗は、サーバー側で証明書の検証中に発生します。

失敗に対して、次のエラーメッセージがサーバーのログファイル server.log に書き込まれます。

Unable to validate certificate

Error occurred while resolving key information  
com.sun.xml.wss.impl.WssSoapFaultException: Certificate validation failed

### 解決方法

Enterprise Server 2.1 に Metro 1.1 をインストールします。

**AIX: EJB モジュール承認テストで @RunAs のエラー AccessLocalException: クライアントが承認されない (6627385)**

### 説明

@RunAs 注釈を使用して定義されている run-as (伝達された) セキュリティー ID を持つエンタープライズ Bean 内のメソッドが、別のエンタープライズ Bean 内のメソッドを呼び出そうとしました。run-as 主体が sun-ejb-jar.xml 配備記述子ファイルで定義されていない場合、呼び出しは javax.ejb.AccessLocalException 例外で失敗します。

javax.ejb.AccessLocalException: Client not authorized for this invocation.

### 解決方法

sun-ejb-jar.xml 配備記述子ファイルの principal-name 要素で、run-as ロールが指定されている主体名を定義します。

**SSL 終了が機能しない (6269102)**

### 説明

SSL 終了が機能しない。ロードバランサ (ハードウェア) を SSL 終了用に設定すると、リダイレクト中に Enterprise Server がプロトコルを https から http に変更する。

### 解決方法

ハードウェアロードバランサと Enterprise Server の間にソフトウェアロードバランサを追加します。

**SSL でソケット接続リークが発生する (6492477)**

### 説明

JVM バグのため、HTTP リスナで security-enabled を true に設定すると、一部の JDK バージョンでリークの問題が起きます。具体的には、このバグを再現する手順は次のようになります。

1. HTTP リスナで security-enabled を true に設定します。

```
<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">
```

- クイックルックテストの末尾にあるドメイン停止をコメントにします。
- クイックルックテストを実行します。
- ソケット使用状況を確認します。

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

使用状況が次のように表示されます。

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

この問題は、GlassFish サイトの [https://glassfish.dev.java.net/issues/show\\_bug.cgi?id=849](https://glassfish.dev.java.net/issues/show_bug.cgi?id=849) で説明されています。

### 解決方法

最新の JDK バージョンにアップグレードします。

## アップグレードユーティリティ

この節では、アップグレードユーティリティ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「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)

### 説明

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、Red Hat ディストリビューションでも発生しています。

インストールプログラムの最後の画面で「アップグレードツールの起動」ボタンをクリックすると、そのインストールプログラム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。

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。

```
as-install/bin/asupgrade --source as-install/domains --target
as-install --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。

3. 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す。

```
file://as-install/docs-ee/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。

## 8.0 Platform Edition (PE) から 8.1 Enterprise Edition (EE) UR2 へのアップグレード中およびそのあと、自己署名付き証明書が信頼されない (6296105)

### 解決方法

アップグレード後、ターゲットの `domain.xml` から次のエントリを削除し、サーバーを再起動します。

```
<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>
```

(*sbs-manual*、*sbs-installer*) 「サーバーインスタンスサーバーに `null` という名前のシステムコネクタがない」というエラーをスローする (6545145)

### 説明

Enterprise Server 8.0 PE から 2.1 へのアップグレード時に、サーバーに `null` という名前のシステムコネクタがないというエラーがスローされ、*sbs-manual* に示されているような無効なユーザー情報が表示されます。ハードコードされた値を変更しても、同じエラーメッセージが表示されます。

## 解決方法

このバグは、8.0 PE から Enterprise Server にアップグレードする場合にのみ発生します。回避策として、8.1、8.2、または9.0のいずれかにアップグレードしてから、Enterprise Server にアップグレードします。

異なるビルドの組み合わせを使用すると、アップグレード中に異なるドメインが失われる (6546130)

## 説明

インプレースアップグレード実行時に、ソースに複数のドメインがある場合、プロセスが中止になっても、インストーラはアップグレードツールを呼び出します。GUI モードで呼び出した場合にそうなります。

## 解決方法

1. インプレースを CLI モードでインストールし、インストール処理の最後にインストーラがアップグレードツールを選択するプロンプトを表示したときに終了します。この操作を行っても、ドメインディレクトリにあるどのドメインも削除されません。アップグレードツールは bin ディレクトリから手動で呼び出す必要があります。
2. GUI モードでインプレースをインストールする場合は、処理中にドメインが失われることがないようにするために、ドメインルートにあるドメインのバックアップを取ります。インストール処理の最後に、インストーラがアップグレードツールを選択するプロンプトを表示したときに終了します。失われているドメインがあれば、ドメインバックアップしたドメインをドメインディレクトリにコピーします。アップグレードツールを起動して、手動でアップグレードを実行します。

**Solaris:** アップグレードツールであらかじめ入力されるマスターパスワードに、**AS8.2**ではなく **Enterprise Server 2.1** のパスワードが使用される (6565825)

## 説明

AS8.2からのアップグレードで、8.2 インストール環境のマスターパスワードがターゲットのインストール環境に継承されません。これが原因となって、次の管理者ログイン時に認証エラーとなります。

## 解決方法

Enterprise Server 2.1 のデフォルト管理パスワードは `changeit` です。8.2 からアップグレードしたあと、Enterprise Server にログインするときの問題を回避するために、次の3つの手順のいずれかを行なってください。

- アップグレードを実行する前に、8.2 管理者パスワードを `changeit` に変更する。
- アップグレード処理時にデフォルトの管理者パスワードを受け入れず、使用するパスワードを明示的に入力する。

- デフォルトパスワードで Enterprise Server 2.1 にログインしたあと、ただちにパスワードを変更する。

asupgrade GUI のローカライズされたオンラインヘルプが存在しない (6610170)

#### 説明

英語以外の言語で asupgrade GUI を実行すると、この GUI のオンラインヘルプが選択した英語以外の言語にローカライズされていません。

#### 解決方法

現時点ではありません。オンラインヘルプは英語以外のすべての目標言語にローカライズされる予定です。

複数ドメインのアップグレード (逆順序) 時にアップグレードツールが nodeagents ディレクトリを削除する (6636871)

#### 説明

複数のドメインを含む構成をサイドバイサイドアップグレードすると、最後に処理されたドメインのノードエージェントだけが存在します。この問題は、アップグレードツールがドメインを処理するたびに、ターゲットの nodeagents ディレクトリを削除してから再作成するために発生します。

### ▼ 解決方法: 複数ドメインのサイドバイサイドアップグレードですべてのノードエージェントを保持する

- 1 各ドメインの処理後に、nodeagents ディレクトリの zip ファイルを作成します。
- 2 すべてのドメインが処理されたあとで、作成した zip ファイルを解凍します。これで、すべてのノードエージェントが存続します。

## 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。

**Windows** で、`--precompilejsp=true` を使用してアプリケーションを配備すると、アプリケーションの JAR ファイルがロックされ、そのあとの配備取り消しや再配備に失敗する (5004315)

#### 説明

Microsoft Windows にアプリケーションを配備するときに JSP のプリコンパイルを要求すると、それ以降、そのアプリケーションの配備取り消しや、そのアプリケーション (または同一モジュール ID を持つ任意のアプリケーション) の再配備を試みても

、予期したとおりに動作しません。この問題は、JSPのプリコンパイル処理でアプリケーションのJARファイルが開かれたまま閉じられないため、Microsoft Windowsがこれらのファイルを配備取り消しで削除することや、これらのファイルを再配備で上書きすることを許可しないことにあります。

配備取り消しは、Application Server からアプリケーションが論理的に削除されるという点では成功します。また、asadmin ユーティリティーからエラーメッセージは返されませんが、そのアプリケーションのディレクトリとロックされた jar ファイルはサーバー上に残っています。サーバーのログファイルには、ファイルとアプリケーションディレクトリの削除に失敗した旨のメッセージが出力されます。

配備取り消し後のアプリケーションの再配備が失敗するのは、既存のファイルとディレクトリをサーバーが削除しようとして失敗するからです。これは、最初に配備されたアプリケーションと同じモジュール ID を持つアプリケーションを配備しようとしたときにも発生します。アプリケーションのファイルを保持するディレクトリの名前を、サーバーはモジュール ID から決定するからです。

同様の理由から、配備取り消しをせずにアプリケーションを再配備しようとするとう失敗します。

## 診断

アプリケーションを再配備しようとする、または、配備取り消しを行ってから配備しようとする、asadmin ユーティリティーは次のようなエラーを返します。

```
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.
```

## 解決方法

アプリケーションを配備するときに `--precompilejsps=false` (デフォルトの設定) を指定すると、この問題は発生しません。そのアプリケーションを最初に使用するとき JSP コンパイルが起動されるため、最初の要求に対する応答時間は、その後の要求に比べて長くなります。

また、プリコンパイルを行う場合には、そのアプリケーションを配備取り消しまたは再配備する前に、サーバーを終了して再起動する必要があります。シャットダウンすると、ロックされている JAR ファイルが解放されるため、再起動後の配備取り消しや再配備が成功します。

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 (**6172006**)

## 説明

`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。

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>` が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` は `web.xml` の Servlet 2.4 スキーマに対する妥当性検証に失敗し、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** (**6184122**)

## 説明

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、*domain-dir/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。

## Enterprise Server で、auth-passthrough Web Server 6.1 アドオンがサポートされない (6188932)

### 説明

Sun GlassFish Enterprise Server 2.1 では、Sun GlassFish Enterprise Server Enterprise Edition 7.1 で使用できる auth-passthrough プラグインが提供する機能に対するサポートが追加されています。ただし、ProductName; 2.1 での auth-passthrough プラグイン機能の設定方法は異なります。

Enterprise Server Enterprise Edition 7.1 での auth-passthrough プラグイン関数は、次に示す 2 層配備のシナリオで有効でした。

- Application Server インスタンスは、企業ファイアウォールの内側にある 2 番目のファイアウォールによって保護される。
- Application Server インスタンスへの直接のクライアント接続は許可されない。

このようなネットワークアーキテクチャの場合、クライアントは、service-passthrough プラグイン関数で設定されたフロントエンド Web サーバーに接続し、HTTP 要求を、プロキシされた Application Server インスタンスに転送して処理します。Application Server インスタンスは、要求をクライアントホストから直接にではなく、Web サーバープロキシからしか受信できません。その結果、プロキシさ

れた Application Server インスタンス上に配備され、クライアントの IP アドレスなどのクライアント情報を照会する任意のアプリケーションは、中継された要求の実際の発信元ホストであるプロキシホストの IP を受信します。

## 解決方法

Application Server Enterprise Edition 7.1 では、プロキシされた Application Server インスタンス上で、そのインスタンスに配備された任意のアプリケーションがリモートクライアントの情報を直接使用するように auth-passthrough プラグイン関数を設定できました。その場合は、プロキシされた Application Server インスタンスが、service-passthrough プラグインを実行している中間の Web サーバー経由ではなく、要求を直接受信したかのように見えます。

Enterprise Server 2.1 では、domain.xml 内の <http-service> 要素の authPassthroughEnabled プロパティを TRUE に設定することにより、auth-passthrough 機能を有効にすることができます。次に例を示します。

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

Application Server Enterprise Edition 7.1 の auth-passthrough プラグインと同じセキュリティの問題が、Enterprise Server の authPassthroughEnabled プロパティにも適用されます。authPassthroughEnabled は、認証に使用される情報 (要求の送信元 IP アドレスや SSL クライアント証明書など) を上書き可能にするため、authPassthroughEnabled を TRUE に設定した信頼できるクラスタまたはサーバーのみに、Enterprise Server 2.1 インスタンスへの接続を許可する必要があります。予防措置として、authPassthroughEnabled を TRUE に設定するのは、企業ファイアウォールの内側にあるサーバーだけにすることをお勧めします。インターネット経由でアクセス可能なサーバーでは、決して authPassthroughEnabled を TRUE に設定しないでください。

プロキシ Web サーバーが service-passthrough プラグインを使用して設定されており、要求を authPassthroughEnabled が TRUE に設定された Enterprise Server インスタンスに転送するシナリオでは、SSL クライアント認証は Web サーバープロキシ上で有効になり、プロキシされた Enterprise Server インスタンス上で無効になる可能性があることに注意してください。この場合、プロキシされた Enterprise Server インスタンスは、SSL 経由で認証されたかのように引き続き要求を処理し、クライアントの SSL 証明書を、それを要求している任意の配備されたアプリケーションに提供します。

## Web サーバー

**AS 9.1 b50e.Linux: AS ロードバランサインストール後に WS インストールを開始できない: libjvm.so: cannot open shared エラーになる (6572654)**

### 説明

この問題は、Linux システムで Sun GlassFish Web Server を Enterprise Server およびロードバランサとともに使用している場合にのみ起きます。このような場合、Enterprise Server とロードバランサのインストール後に、libicui18n.so.2 と libicuuc.so.2 が競合するため Web Server が起動に失敗することがあります。これらのライブラリは /opt/sun/private/lib と /opt/sun/appserver/lib の両方にあります。

### 解決方法

使用すべき正しいライブラリは /opt/sun/appserver/lib にあるライブラリです。このライブラリに対して lbplugin が構築されているからです。/opt/sun/private/lib からこの2つのライブラリを削除すると、Web Server はエラーを出さずに起動します。

または、/opt/sun/private/lib からライブラリを削除できない場合は、Web サーバーの startserv スクリプトで、LD\_LIBRARY\_PATH の /opt/sun/private/lib の前に /opt/sun/appserver/lib を配置することもできます。つまり、

```
# 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
```

これを、次の項目で置き換えます。

```
# 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 サービス

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。

**Ant タスク wsimport が Java EE SDK b33d (JDK 1.6 を使用) で NoClassDefFoundError を出して失敗する(6527842)**

### 説明

Java EE SDK b33d に含まれる JDK 1.6 を使用して JAX-WS テストを実行するときに、問題が発生する場合があります。テストはただちに停止し、次のメッセージが表示されます。

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

webservices-tools.jar に  
com/sun/tools/ws/WsImport.class、com/sun/tools/ws/ant/WsImport.class、および  
com/sun/tools/ws/ant/WsImport2.class が含まれていても、このエラーは起きます。  
さらに、1.5.0-10 JDK を使用すると、同じテスト作業空間が問題なく機能します。

### 解決方法

JAX-WS テストを実行する前に、webservices-api.jar を  
\$JAVA\_HOME/jre/lib/endorsed にコピーします。

### IFREE ビルドで publish-to-registry コマンドが失敗する (6602046)

#### 説明

JAXR は SAAJ を使用して SOAP メッセージをレジストリに送信します。IFR 以外では、SAAJ impl クラスは lib/webservices-rt.jar の下にあります。IFR の場合、SAAJ クラスは引き続き lib/webservices-rt.jar の下にあります。また、saaj-impl.jar は /usr/share/lib ディレクトリにあります。この jar ファイルは Enterprise Server で格上げされ、webservices-rt.jar のクラスよりも優先されています。この jar ファイルには、SOAP メッセージを Web サービスレジストリに送信するために必要なセキュリティ権限がありません。/usr/share/lib ディレクトリ内の jar に権限を付与するか、/usr/share/lib の jar に依存しないように、パッケージを変更する必要があります。

### 解決方法

server.policy ファイルに次の内容を追加します。

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

### JDK6 u4 b3 で wscompile は「package javax.xml.rpc does not exist」エラーが発生する (6638567)

#### 説明

wscompile Ant タスクは、JDK 6 Update 4 では失敗します。JAX-RPC API クラスごとに、次のエラーメッセージが表示されます。

```
package package-name does not exist
```

### 解決方法

wscompile Ant タスクを実行する前に j2ee.jar ではなく javaee.jar がクラスパスに指定されていることを確認します。