



Sun Java System Application Server Enterprise Edition 8.2 ト ラブルシューティングガイド



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

Part No: 820-1614
2005Q2

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

はじめに	9
1 概要	15
事前に計画する	15
▼サーバー設定をバックアップおよび復元する	16
問題を特定する	16
▼設定情報を確認する	16
解決法を探す	18
システム設定を確認する	18
メッセージを評価する	18
ログファイルを検査する	20
その問題を以前に解決したことがないか確認する	22
製品マニュアルを検索する	23
ナレッジベースを検索する	23
▼ナレッジベースを検索する	23
オンラインフォーラムで検索または参加する	23
サポートに問い合わせる	24
2 よくある問題	25
.asadmintruststore ファイルがないためアップグレード時のバグの原因となる	26
解決法	26
削除された .asadmintruststore ファイルの復元	26
▼新しい .asadmintruststore ファイルを作成する	26
asadmin start-domain コマンドが失敗する	27
Error: CLI143 ... には複数のドメインがあります	27
Error: ドメインを起動できません	27
リブート時にドメインまたはノードエージェントを自動的に再起動する	28

UNIX プラットフォーム上での自動再起動	28
Microsoft Windows プラットフォーム上での自動再起動	29
ログファイルが見つからない	32
ローカルサーバーのアクセスが失敗する (http://localhost:8080)	34
サーバーは起動したか?	34
サーバーは正しいポートで起動したか?	35
リモートサーバーへのアクセスが失敗する	35
サーバーをローカルで利用できるか?	36
プロキシの設定が問題を引き起こしていないか?	36
管理コンソールを表示できない	37
Application Server は稼働しているか?	37
想定されるポートで管理コンソールが実行されているか?	37
セキュリティーマネージャーが無効になっていないか?	38
サーバーアプリケーションを使用できない	38
Application Server は稼働しているか?	38
アプリケーションの配備は成功したか?	39
Don't Prompt オプション使用時の無効なユーザーまたはパスワード	39
管理者のユーザー名またはパスワードがわからない	40
▼ ユーザー名とパスワードを完全に削除する	40
Windows 上でサーバーが起動しない (ポートの衝突)	41
サーバーのポートで別のアプリケーションが実行されていないか?	41
以前に稼働していたサーバーで異常な方法での停止が発生しなかったか?	41
同じサーバー上の複数インスタンスのデバッグ時にポートが衝突する	42
説明	42
解決法	42
2つのサーバーインスタンスが Windows 上の同じポートにバインドされる	42
説明	42
解決法	43
Error: 指定したパスがシステムで見つからない	43
説明	43
解決法 1	43
解決法 2	43
解決法 3	44
アプリケーションで Error persistence.support.JDODataStoreException が発生する	44
説明	44

解決法	44
asadmin set コマンドを使用すると予期していない結果になることがある	44
説明	44
jar ファイルへのストリームが開いているためにアプリケーションを配備解除または再配備できない (Windows のみ)	45
説明	45
解決法	45
ディレクトリの手動削除後 Application Server を再インストールできない	46
説明	46
解決法 1	46
解決法 2	46
サーバクラッシュ後に JVM スレッドダンプを生成できない	46
説明	46
解決法	47
▼ JVM スレッドダンプを有効にする	47
main スレッドでバンドル版の ANT から java.lang.NoClassDefFoundError がスローされる	47
解決法	47
分解したアプリケーションファイルを変更しても /generated/xml ディレクトリは更新されない	47
解決法	47
3 HADB の問題	49
HADB データベースの作成が失敗する	50
共有メモリーに関連する問題	50
セマフォが少なすぎる	52
データベースノードにアクセスできないため、データベースが機能しない	52
管理エージェントでドメインを確立できなかった	52
hadbm create または hadbm addnodes コマンドがハングする	53
ma (管理エージェントプロセス) に障害が発生する	54
サーバーが応答する前に長いアイドル状態が発生する	54
説明	54
解決法	54
要求が成功しない	55
ロードバランサのタイムアウトは適切か?	55
システムクロックは同期しているか?	55

Application Server は HADB と通信しているか？	56
セッション持続の問題	56
create-session-store コマンドが失敗した	57
インスタンスレベルのセッション持続設定が機能しなかった	57
セッションデータが壊れている可能性がある	57
HADB のパフォーマンスの問題	58
CPU またはメモリーのリソースが不足していないか、あるいはスワッピングが多すぎないか？	58
ディスクの競合はないか？	59
HADB データデバイス空間が不足していないか？	60
ほかの HADB リソースが不足していないか？	61
高負荷の問題	61
ダブルログの容量が不足していないか？	62
node-internal ログがいっぱいになっていないか？	63
ロックの数は十分か？	63
パフォーマンスチューニングを行えば問題を解決できるか？	64
セマフォリソースの不足で発生する接続の問題	64
説明	64
解決法	65
CPU 利用率の向上	65
説明	65
解決法	65
HADB 管理の問題	65
hadbm コマンドが失敗する: エージェントにアクセスできなかった	66
hadbm コマンドが失敗する: コマンドが見つからない	66
hadbm コマンドが失敗する: JAVA_HOME が定義されていない	67
hadbm createdomain が失敗し、2 つに分割されたドメインが作成される	68
create が失敗する: ホスト上にパスがない	68
データベースが起動しない	69
clear コマンドが失敗した	70
create-session-store が失敗した	71
hadbm コマンドがハングする	72
HADB を再起動できない	72
共有メモリーセグメントキーがすでに存在する (Windows のみ)	73
説明	73
解決法	73

configure-ha-cluster の失敗	73
説明	73
解決法 1	74
解決法 2	74
configure-ha-cluster を実行できない	74
説明	74
解決法	74
hadbm set コマンドが失敗する	75
説明	75
解決法	75
configure-ha-cluster の失敗: HADB インスタンスの作成が失敗する	75
説明	75
解決法	75
 4 セキュリティーの問題	77
java.security.AccessControlException: アクセス拒否エラー	77
説明	77
解決法	78
javax.ejb.AccessLocalException: 承認されていないクライアントエラー	78
説明	78
解決法	78
Solaris レルムで認証が機能していない	79
アプリケーションクライアントで相互認証が機能していない	79
説明	79
解決法	79
 5 よくある質問	81
サーバー側レルムを1つも設定していないとどうなりますか?	81
PKCS12 証明書をクライアント証明書として使用できますか?	82
SSL クライアントの TLS/SSL ハンドシェイク情報を見ることはできますか?	82
キーストアのパスワードを変更できますか?	82
JAX-RPC 内のセッションをどのように維持できますか?	82
スタンドアロン Java クライアントからはどのようにネームサービスにアクセスできますか?	83
▼アプリケーションクライアントからネームサービスにアクセスする	83

リモート EJB にアクセスするために RMI-IIOP スタブは必要ですか？	85
アプリケーションロガーのログレベルを変更する方法を教えてください。	85
索引	87

はじめに

Sun Java System Application Server Enterprise Edition のトラブルシューティング。

ここでは、Sun Java™ System Application Server のマニュアルセット全体の概要と表記規則について説明します。

Application Server のマニュアルセット

Application Server のマニュアルセットは、配備の計画とシステムのインストールについて説明しています。スタンドアロン Application Server のマニュアルの URL (Uniform Resource Locator) は <http://docs.sun.com/app/docs/coll/1310.4> です。Sun Java Enterprise System (Java ES) Application Server のマニュアルの URL は <http://docs.sun.com/app/docs/coll/1659.1> です。Application Server の概要については、次の表に示す順序で各マニュアルを参照してください。

表 P-1 Application Server のマニュアルセットの内容

マニュアル名	説明
『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。サポートされるハードウェア、オペレーティングシステム、JDK™ (Java Development Kit)、およびデータベースドライバの包括的な一覧表が含まれています。
『Quick Start Guide』	Application 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。
『配備計画ガイド』	最適な方法で確実に Application Server を導入するための、システムニーズや企業ニーズの分析。サーバーを配備する際に注意すべき一般的な問題や懸案事項についても説明しています。
『Developer's Guide』	J2EE コンポーネントおよび API 用のオープン Java 標準モデルに従い、Application Server 上で実行することを目的とする Java 2 Platform, Enterprise Edition (J2EE™ 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、デバッグ、配備、ライフサイクルモジュールの作成などについての情報も提供します。

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

マニュアル名	説明
『J2EE 1.4 Tutorial』	J2EE アプリケーションの開発のための J2EE 1.4 プラットフォーム技術および API の使用。
『管理ガイド』	管理コンソールからの、Application Server サブシステムおよびコンポーネントの設定、管理、および配備。
『高可用性 (HA) 管理ガイド』	高可用性データベースのための、インストール後の設定と管理に関する解説。
『Administration Reference』	Application Server 設定ファイル <code>domain.xml</code> の編集。
『アップグレードと移行』	新しい 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 (特に Application Server 6.x または 7 からの移行)。このマニュアルでは、製品仕様の非互換性をもたらす可能性のある、隣接した製品リリース間の相違点や設定オプションについても説明しています。
『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 Application Server の調整。
『トラブルシューティングガイド』	Application Server の問題の解決。
『Error Message Reference』	Application Server のエラーメッセージの解決。
『Reference Manual』	Application Server で使用できるユーティリティコマンド。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。

関連マニュアル

Application Server は、単体で購入することが可能です。あるいは、ネットワークまたはインターネット環境にわたって分散しているエンタープライズアプリケーションをサポートするソフトウェアインフラストラクチャーである Java ES のコンポーネントとして購入することもできます。Application Server を Java ES のコンポーネントとして購入した場合は、<http://docs.sun.com/coll/1286.2> にあるシステムマニュアルをよく読むことをお勧めします。Java ES およびそのコンポーネントに関するすべてのマニュアルの URL は <http://docs.sun.com/prod/entsys.5> です。

ほかの Sun Java System サーバーについては、次のマニュアルを参照してください。

- Message Queue のマニュアル
- Directory Server のマニュアル
- Web Server のマニュアル

さらに、次のリソースが役立つことがあります。

- J2EE 1.4 Specifications (<http://java.sun.com/j2ee/1.4/docs/index.html>)
- J2EE 1.4 Tutorial (<http://java.sun.com/j2ee/1.4/docs/tutorial/doc/index.html>)
- J2EE Blueprints (<http://java.sun.com/reference/blueprints/index.html>)

デフォルトのパスとファイル名

次の表は、このマニュアルで使用するデフォルトのパスやファイル名について説明したものです。

表 P-2 デフォルトのパスとファイル名

ブレースホルダ	説明	デフォルト値
<i>install-dir</i>	Application Server のベースインストールディレクトリを表します。	Solaris™ プラットフォームへの Sun Java Enterprise System (Java ES) インストールの場合: /opt/SUNWappserver/appserver Linux プラットフォームへの Java ES インストールの場合: /opt/sun/appserver/ Solaris および Linux プラットフォームへのインストールで、ルートユーザーでない場合: ユーザーのホームディレクトリ /SUNWappserver Solaris および Linux プラットフォームへのインストールで、ルートユーザーである場合: /opt/SUNWappserver Windows のすべてのインストールの場合: SystemDrive:\Sun\AppServer
<i>domain-root-dir</i>	すべてのドメインを含むディレクトリを表します。	Solaris プラットフォームへの Java ES インストールの場合: /var/opt/SUNWappserver/domains/ Linux プラットフォームへの Java ES インストールの場合: /var/opt/sun/appserver/domains/ そのほかのすべてのインストールの場合: <i>install-dir</i>/domains/
<i>domain-dir</i>	ドメインのディレクトリを表します。 設定ファイルには、次のように表される <i>domain-dir</i> があります。 \${com.sun.aas.instanceRoot}	<i>domain-root-dir</i> / <i>domain-dir</i>
<i>instance-dir</i>	サーバーインスタンスのディレクトリを表します。	<i>domain-dir</i> / <i>instance-dir</i>

表記上の規則

このマニュアルでは、次のような字体や記号を特別な意味を持つものとして使用します。

表 P-3 表記上の規則

字体または記号	意味	例
AaBbCc123	コマンド名、ファイル名、ディレクトリ名、画面上のコンピュータ出力、コード例を示します。	.login ファイルを編集します。 ls -a を使用してすべてのファイルを表示します。 machine_name% you have mail.
AaBbCc123	ユーザーが入力する文字を、画面上のコンピュータ出力と区別して示します。	machine_name% su Password:
<i>aabbcc123</i>	変数を示します。実際に使用する特定の名前または値で置き換えます。	ファイルを削除するには、rm <i>filename</i> と入力します。
『 』	参照する書名を示します。	『コードマネージャー・ユーザーズガイド』を参照してください。
「 」	参照する章、節、ボタンやメニュー名、強調する単語を示します。	第 5 章「衝突の回避」を参照してください。 この操作ができるのは、「スーパーユーザー」だけです。
\	枠で囲まれたコード例で、テキストがページ行幅を超える場合に、継続を示します。	sun% grep '^#define \ XV_VERSION_STRING'

コード例は次のように表示されます。

■ C シェル

```
machine_name% command y|n [filename]
```

■ C シェルのスーパーユーザー

```
machine_name# command y|n [filename]
```

■ Bourne シェルおよび Korn シェル

```
$ command y|n [filename]
```

■ Bourne シェルおよび Korn シェルのスーパーユーザー

```
# command y|n [filename]
```

[] は省略可能な項目を示します。上記の例は、*filename* は省略してもよいことを示しています。

| は区切り文字 (セパレータ) です。この文字で分割されている引数のうち 1 つだけを指定します。

キーボードのキー名は英文で、頭文字を大文字で示します (例: Shift キーを押します)。ただし、キーボードによっては Enter キーが Return キーの動作をします。

ダッシュ (-) は 2 つのキーを同時に押すことを示します。たとえば、Ctrl-D は Control キーを押したまま D キーを押すことを意味します。

記号の表記規則

次の表に、このマニュアルで使用する記号の表記規則を示します。

表 P-4 記号の表記規則

記号	説明	例	意味
[]	任意に指定できる引数とコマンドオプションです。	ls [-l]	-l オプションは省略可能です。
{ }	必須のコマンドオプションの選択肢を囲みます。	-d {y n}	-d オプションでは、y 引数または n 引数を使用する必要があります。
\${ }	変数参照を示します。	\${com.sun.javaRoot}	com.sun.javaRoot 変数の値を参照します。
-	同時に押すキーを示します。	Control-A	Control キーを押しながら A キーを押します。
+	順番に押すキーを示します。	Ctrl+A+N	Control キーを押してから放し、それに続くキーを押します。
→	グラフィカルユーザーインタフェースでのメニュー項目の選択順序を示します。	「ファイル」→「新規」 →「テンプレート」	「ファイル」メニューから「新規」を選択します。「新規」サブメニューから「テンプレート」を選択します。

マニュアル、サポート、およびトレーニング

Sun のサービス	URL	内容
マニュアル	http://jp.sun.com/documentation/	PDF 文書および HTML 文書をダウンロードできます。
サポートおよび トレーニング	http://jp.sun.com/supporttraining/	技術サポート、パッチのダウンロード、および Sun のトレーニングコース情報を提供します。

概要

この章では、Sun Java System Application Server 8.2 Enterprise Edition Platform Edition のトラブルシューティングに使用できるツール、手法、および情報源について説明します。問題の評価および調査のためのガイドラインも提供します。

- 15 ページの「事前に計画する」
- 16 ページの「問題を特定する」
- 18 ページの「解決法を探す」

事前に計画する

アプリケーションの配備、配備解除、再配備を行なったり、異なるサーバー設定を試したりしていくうちに、サーバーが混乱状態または不安定状態になることがあります。そのような場合は、よりどころとなる有効な設定が、前に保存されていると役立ちます。これ自体は問題解決策ではありませんが、まず第一に問題を回避するための方法です。

Application Server の `asadmin` コマンドには、指定したドメインをバックアップする `backup-domain` オプションがあります。このオプションを使用して、サーバー設定の定期的な「スナップショット」を取ってください。その後、必要な場合には、`restore-domain` オプションを使用して1つ以上のドメインを正常に機能する既知の状態に復元します。

`asadmin backup-domain` および `restore-domain` オプションの詳細な使用手順については、『Application Server 管理ガイド』を参照してください。ただし、この『トラブルシューティングガイド』の目的の範囲内で、サーバー設定のバックアップおよび復元手順の概要を次に示します。

▼ サーバー設定をバックアップおよび復元する

- 1 **Application Server** を起動します。

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

- 2 ドメインを停止します。

```
install_dir/bin/asadmin stop-domain domain_name
```

- 3 ドメインをバックアップします。

```
install_dir/bin/asadmin backup-domain domain_name
```

デフォルトでは、バックアップしたディレクトリは *install_dir/backups* ディレクトリに保存されます。

- 4 **Application Server** の設定またはドメイン、あるいはその両方に必要な変更を加えます。

- 5 必要に応じて、サーバーやドメインの設定を、前記手順3で保存した状態に復元します。

```
install_dir/bin/asadmin restore-domain --filename backup_file domain_name
```

問題を特定する

一般に、J2EE アプリケーションサーバーは、複雑で高度に洗練されたオペレーティング環境に配備されます。Sun Java System Application Server は、Java、Java サブレット、XML、JSP、JDBC データソース、EJB テクノロジーなど、幅広いテクノロジーに対応しています。Application Server と関連するほかの製品およびツールは、LDAP、Web Server、Sun Java System Message Queue、配備および移行ツールなどです。多数の本質的に異なるコンポーネントが関係する複雑な問題を理解および診断するには、十分な知識と注意深い診断プロセスが必要です。

次に示す情報のいずれかまたはすべてを収集すれば、問題を分類して解決策を探ることが容易になります。システム情報の収集には、Solaris の `pkginfo` や `showrev`、Linux の `rpm` など、オペレーティングシステムのユーティリティーが役立ちます。

▼ 設定情報を確認する

- 1 オペレーティングシステムとインストールされている製品の正確なバージョン番号を確認します。
- 2 パッチが適用されているかどうか調べます。適用されている場合は、製品およびオペレーティングシステムのパッチ番号を指定します。

- 3 システムの構成を調べます。
 - a. システムが備えるシステムリソース(メモリー、ディスク、スワップ空間など)について調べます。
 - b. インストールされているアプリケーションサーバー、**Web** サーバー、およびディレクトリサーバーの数を調べます。
 - c. **Web** サーバーと **Application Server** との接続方法について調べます。それらは同じマシン上にありますか?
 - d. **Application Server** とディレクトリサーバーとの接続方法について調べます。
 - e. アプリケーションサーバーがクラスタ化されているかどうか調べます。
 - f. アップグレードが行われたかどうか調べます。すでに行われている場合は、ソースバージョンとターゲットバージョンを確認します。
 - g. 移行が行われたかどうか調べます。すでに行われている場合は、ソースバージョンとターゲットバージョンを確認します。
- 4 新しいアプリケーションが配備されたかどうか調べます。
- 5 **SSL** に対応しているかどうか調べます。
- 6 **HADB** および使用しているバックエンドデータベースのバージョンを確認します。
- 7 データベースのアクセスに使用している **JDBC** ドライバを特定します。
- 8 使用している **JDK** のバージョンを確認します。
- 9 **JVM** のヒープ、スタック、およびガベージコレクション関連のパラメータの設定値を確認します。
- 10 **JVM** オプションを確認します。
- 11 インストールで使用されているサードパーティーのテクノロジーについて調べます。
- 12 相互運用コンポーネントのバージョンが、リリースノートに記載された互換表に適合しているかどうか調べます。

これらの情報を収集したあと、次の作業を行います。

 - **Web** サーバーのエラーおよびアクセスログのデータを収集します (**Web** サーバーインスタンスによって異なる)。

- Application Server のスタックトレースを収集します。特定の問題に関連するログの最新セットが稼働するように注意してください。これにより、何ギガバイトもの無関係なログ情報を走査することを避けることができます。
- その問題を解決するためにすでに実行した可能性のある手順を含め、問題の最初の出現時に発生したイベントのシーケンスを調べます。

解決法を探す

問題を特定し、何が間違っているかについて予備的な仮説を立てると、調査を行う準備が整います。

この節では、次のトピックについて説明します。

- [18 ページの「システム設定を確認する」](#)
- [18 ページの「メッセージを評価する」](#)
- [20 ページの「ログファイルを検査する」](#)
- [22 ページの「その問題を以前に解決したことがないか確認する」](#)
- [23 ページの「製品マニュアルを検索する」](#)
- [23 ページの「ナレッジベースを検索する」](#)
- [23 ページの「オンラインフォーラムで検索または参加する」](#)
- [24 ページの「サポートに問い合わせる」](#)

システム設定を確認する

もっとも明白な解決法が見過ごされている場合も少なくありません。したがって、最初の手順はシステム設定を確認することです。最新のシステム要件と依存関係については、『Sun Java System Application Server 8.2 リリースノート』を参照してください。

メッセージを評価する

メッセージには通常、試行されたアクション、アクションの結果、さらに場合によっては障害や失敗の原因に関する情報が含まれています。

メッセージの種類

ログファイルに含まれている一般的なメッセージエントリの種類は、次のとおりです。

- エラー - これらのメッセージは、状態が「失敗」と報告される要因となる重大な失敗を示します。エラーメッセージには、通常、発生した問題の性質および原因についての詳細な情報が示されます。

- 警告 - これらのメッセージは重大でない失敗を示します。警告メッセージには、通常、失敗の原因および性質についての情報が含まれ、考えられる救済策も示されます。
- 情報 - これらのメッセージは、各タスクが正常に完了したことを示します。

エラーメッセージ

アプリケーションの続行を妨げる問題には、通常、エラーメッセージが伴います。

- 何が間違っているか、およびそれを修正することが可能な場合、何を行う必要があるかについて、メッセージの内容がきわめて明白な場合もあります。たとえば、`asadmin start-domain` コマンドを使用してドメインを起動する場合、ドメインが起動済みになったあとに不注意で同じコマンドを再度実行すると、次のメッセージが表示されます。

```
userD:\Sun\studio5_se\appserver8\bin>asadmin start-domain
Domain already started : domain1
Domain domain1 Started.
```

この場合、メッセージに示される明白なガイダンスに従って、問題を無視することができます。

- 問題または解決法についての全般的な情報のみを表示するエラーメッセージや、複数の可能性を提示するエラーメッセージもあります。次に例を示します。

```
[16/Jun/2003:22:20:50] SEVERE ( 2204): WEB0200: Configuration error
in web module [JAXBProjectStudio] (while initializing virtual server
[server1]) com.ipplanet.ias.config.ConfigException: Failed to load
deployment descriptor for: JAXBProjectStudio cause:
java.io.FileNotFoundException:
```

この場合は、問題が明確でないか、複数の点で間違いが存在する可能性があります。さまざまな可能性とともに、おそらく多くの解決策を検討することが必要になる可能性があります。時間や費用のかかる解決法が提案された場合、実際に何かを行う前に、その解決法で成功するという確信を得るための手順を踏んでください。

- 問題の解決に役立たず、ガイダンスもほとんど示されないエラーメッセージもあります。次に例を示します。

```
[23/Jun/2003:16:50:45] WARNING ( 1972): for host 127.0.0.1 trying to GET /SupplierServiceClient1/
SupplierServiceClient1_SOAP.html, send-file reports: HTTP4144: error sending D:/Sun/studio5_se/appserver8
/domains/domain1/server1/applications/j2ee-modules/SupplierServiceClient1_1/SupplierServiceClient1_SOAP.html
(Overlapped I/O operation is in progress.)
status=1:5
```

この場合、対策を取るための情報はほとんどありません。何かを行う前に、エラーの原因となった正確な状況と現れた症状を把握することはきわめて重要です。

Application Server のすべてのエラーメッセージの説明については、次の場所にある『Sun Java System Application Server Error Message Reference』を参照してください。

http://docs.sun.com/db/coll/ApplicationServer8_04q4

ログファイルを検査する

多数の Application Server サブシステムでログファイルが作成され、イベントがそれらのファイルに記録されます。これらのログファイルの主な目的は、トラブルシューティング情報を提供することです。

記録されたメッセージから、メッセージテキストに加えて次の情報を得ることができます。

- イベントの日時
- イベントのログレベル — Application Server によって指定されたログレベル ID または名前
- プロセス識別子 (PID) — Application Server プロセスの PID
- (オプション) 仮想サーバー識別子 (VSID) — メッセージの生成元 VSID
- メッセージ識別子 (MID) — サブシステムと 4 桁の整数
- メッセージデータ

Application Server の各問題分野に関連する固有のログについては、このマニュアルの関連する章を参照してください。

ログレベル

Application Server の管理 GUI では、さまざまなログレベルを設定できます (FINEST、FINER、FINE、CONFIG、INFO、WARNING、SEVERE、ALERT、および FATAL)。ログレベルを FINEST に設定するとすべてのメッセージが記録され、ログレベルを FATAL に設定すると重大なエラーメッセージのみが出力されます。

詳細なログレベル (FINEST、FINER、FINE) では、ある特定のイベントのログ情報が大量に生成されるため、一見すると、実際には存在しなくてもエラー条件が存在するように見えることがあります。

デフォルトレベルの INFO より重大でないログレベル (FINEST、FINER、FINE、および CONFIG) のすべてのメッセージには、デバッグに関連する情報が含まれているので、特定して有効にする必要があります。これを行う手順については、『Sun Java System Application Server 管理ガイド』を参照してください。

Application Server には、標準の JDK ログレベルに加えて、Application Server ログファイル (server.log) との関係を直観的に把握でき、Solaris と緊密に統合するように

考案されたログレベルが追加されています。ログレベル **ALERT** および **FATAL** は Application Server に固有なログレベルで、JDK1.4 のログ API には実装されていません。

注 - Microsoft Windows オペレーティング環境で使用されるイベントログ機構については、キーワード「イベントログ」で Windows ヘルプシステムを索引検索してください。Windows server.log ファイルにログを送信する場合、ログレベル **INFO**、**WARNING**、**SEVERE**、**ALERT**、または **FATAL** のメッセージのみが Windows イベントログに記録されます。

ログオプション

管理 GUI は次の 2 つのログオプションを提供します。

- オプション 1 — `stdout` (`System.out.print`) コンテンツをイベントログに記録する
 - オプション 2 — `stderr` (`System.err.print`) コンテンツをイベントログに記録する
- これらのオプションを設定すると、`stdout` と `stderr` のメッセージが `server.log` ファイルに出力されます。イベントログは、Solaris では `syslog` デーモン、Microsoft Windows ではイベントログです。
- 前記のオプションを設定しないと、次のように動作します。
- `stdout` または `stderr` に書き込まれる (つまり、`System.out` または `System.err` を使用する) すべての情報が、ログに表示されない。
 - JDK ログで記録されたメッセージがログに表示される。
 - `stdout` または `stderr` に書き込まれるメッセージは **INFO** レベルとして表示されるが、メッセージ ID は付かない。

クライアント側のログ

アプリケーションクライアントコンテナ (ACC) には、ローカルファイルにのみ出力可能な独自のログサービスがあります。通常、ACC は、Application Server とは別のホスト上の独自のプロセスで実行されます。ログインフラストラクチャーとログファイルは ACC 固有のものです。ACC の設定は、`sun-acc.xml` ファイルに格納されています。詳しくは、『Sun Java System Application Server Developer's Guide』を参照してください。

スレッドダンプの取得

ここでは、Application Server 8.2 のスレッドダンプを取得する方法について説明します。デフォルトでは、サーバーからコアファイルがダンプされ、`server.xml` ファイルにある `-Xrs java-option` フラグに従って再起動が行われます。

UNIX の場合

UNIX でサーバースレッドダンプを取得する方法は次のとおりです。

▼ UNIXでサーバースレッドダンプを取得する

- 1 影響を受けるサーバーインスタンスの `server.xml` ファイルに `-Xrs java-option` フラグが含まれていないことを確認します。含まれている場合は、`-Xrs java-option` フラグを削除します。
- 2 オプションが変更されている場合は、サーバーインスタンスを再起動します。
- 3 `ps` コマンドを使用して、アプリケーションサーバーが稼働している `java` プロセスまたは `appservDAS` プロセス、あるいはその両方を特定します。
- 4 アプリケーションサーバーインスタンスで次のコマンドを実行します。

```
kill -3 pid
```

この `kill` コマンドで、スレッドダンプがサーバーインスタンスの `server.log` ファイルにリダイレクトされます。

Windows の場合

Windows でサーバースレッドダンプを取得する方法は次のとおりです。

▼ Windowsでサーバースレッドダンプを取得する

- 1 サーバーインスタンスの `server.xml` ファイルに `-Xrs java-option` フラグが含まれていないことを確認します。含まれている場合は、`-Xrs java-option` フラグを削除します。
- 2 オプションが変更された場合は、**Application Server** を再起動します。
- 3 「**Application Server**」ウィンドウで `ctrl-brk` を入力します。スレッドダンプがインスタンスの `server.log` ファイルにリダイレクトされます。

その問題を以前に解決したことがないか確認する

最初にこの『トラブルシューティングガイド』に目を通して、その問題が扱われているかどうか確認することをお勧めします。扱われているなら、適切な解決法を選択します。多くの解決法では、詳細な説明や例については、Application Server のマニュアルセット内のほかのマニュアルを参照するように指示されています。

製品マニュアルを検索する

トラブルシューティングする製品の該当バージョンの『リリースノート』を読むことから始めます。

この Application Server 製品リリースのマニュアルは、次の場所から入手できます。

<http://docs.sun.com/db/coll/ApplicationServer81>

Application Server のマニュアルの詳細については、9 ページの「Application Server のマニュアルセット」を参照してください。

ナレッジベースを検索する

ナレッジベースは、製品の問題に関する記事を集めたもので、トラブルシューティングに役立つ情報が含まれています。ナレッジベースを利用するには、次の手順に従います。

▼ ナレッジベースを検索する

- 1 **SunSolve Online** (<http://sunsolve.sun.com/pub-cgi/show.pl?target=home>) に移動します。
- 2 「SunSolve Collections」で「Search Collections」リンクをクリックします。
- 3 検索するコレクションのチェックボックスを選択します。
- 4 「次へ」をクリックします。
- 5 検索基準を入力します。
- 6 「検索」をクリックします。

オンラインフォーラムで検索または参加する

オンラインフォーラム内で直接検索するか、ログインおよび登録してメッセージを投稿できるようにします。Application Server のオンラインフォーラムは、次の場所にあります。<http://swforum.sun.com/jive/index.jsp?cat=7>

サポートに問い合わせる

必要な場合は、これまでに取得した情報を手元に用意して、テクニカルサポート <http://www.sun.com/service/contacting> までお問い合わせください。

よくある問題

ここでは、Application Server の使用時に発生する可能性のある、最も一般的な問題について説明します。

- 26 ページの「.asadmintruststore ファイルがないためアップグレード時のバグの原因となる」
- 27 ページの「asadmin start-domain コマンドが失敗する」
- 28 ページの「リブート時にドメインまたはノードエージェントを自動的に再起動する」
- 32 ページの「ログファイルが見つからない」
- 34 ページの「ローカルサーバーのアクセスが失敗する (http://localhost:8080)」
- 35 ページの「リモートサーバーへのアクセスが失敗する」
- 37 ページの「管理コンソールを表示できない」
- 38 ページの「サーバーアプリケーションを使用できない」
- 41 ページの「Windows 上でサーバーが起動しない (ポートの衝突)」
- 42 ページの「同じサーバー上の複数インスタンスのデバッグ時にポートが衝突する」
- 42 ページの「2 つのサーバーインスタンスが Windows 上の同じポートにバインドされる」
- 43 ページの「Error: 指定したパスがシステムで見つからない」
- 44 ページの「アプリケーションで Error persistence.support.JDODataStoreException が発生する」
- 44 ページの「asadmin set コマンドを使用すると予期していない結果になることがある」
- 45 ページの「jar ファイルへのストリームが開いているためにアプリケーションを配備解除または再配備できない (Windows のみ)」
- 46 ページの「ディレクトリの手動削除後 Application Server を再インストールできない」
- 46 ページの「サーバークラッシュ後に JVM スレッドダンプを生成できない」
- 47 ページの「main スレッドでバンドル版の ANT から java.lang.NoClassDefFoundError がスローされる」

- 47 ページの「分解したアプリケーションファイルを変更しても `/generated/xml` ディレクトリは更新されない」

.asadmintruststore ファイルがないためアップグレード時のバグの原因となる

.asadmintruststore ファイルがサーバー管理者のホーム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。
- インストールユーザー以外が実行した場合、.asadmintruststore ファイルが、インストールユーザーのホームディレクトリから、実行ユーザーのホームディレクトリに移動またはコピーされます。
- このファイルをインストールユーザーのホームディレクトリから実行中のユーザーのホームディレクトリに(コピーではなく)移動した場合は、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 6315957、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ることに注意してください。

削除された .asadmintruststore ファイルの復元

ドメインの .asadmintruststore ファイルの削除時に、インスタンスが存在している場合があります。そのような場合は、新しい .asadmintruststore ファイルを作成できます。

▼ 新しい .asadmintruststore ファイルを作成する

- 1 `asadmin start-domain` コマンドを使用して、管理するドメインを起動します。
これはローカルの `asadmin` コマンドなので、ドメインの起動に `~/.asadmintruststore` ファイルは必要ありません。

- 2 リモートの asadmin コマンドのいずれかを実行します。
リモート asadmin コマンドは、`---user`、`---passwordfile` (`--password`)、`---host`、および `---port` オプションをコマンド行に指定する必要があるコマンドで、ターゲットドメインが稼働していることも必要です。よく使用するリモート asadmin コマンドは、`asadmin list` です。
- 3 確認画面で「y」と入力して、新しいドメイン証明書を受け入れます。

asadmin start-domain コマンドが失敗する

コマンド `asadmin start-domain` が、次のいずれかのエラーで失敗します。

- [27 ページの「Error: CLI143 ... には複数のドメインがあります」](#)
- [27 ページの「Error: ドメインを起動できません」](#)

Error: CLI143 ... には複数のドメインがあります

説明

引数を指定しないで実行すると、コマンド `asadmin start-domain` は次のエラーで失敗します。

```
CLI143 There is more than one domain in C:\\Sun\\AppServer\\domains.  
Please use operand to specify the domain.  
CLI156 Could not start the domain null.
```

このエラーは、ドメインのディレクトリに複数のドメインがあり、そのいずれも `domain1` という名前ではなく、`start-domain` コマンドでドメインが1つも指定されていない場合に発生します。

解決法

`start-domain` コマンドの実行時にドメインを指定します。

```
asadmin start-domain domain1
```

Error: ドメインを起動できません

説明

このメッセージは Application Server 8 から出力されます。メッセージ全体は次のいずれかようになります。

```
Could not start the domain.  
There are no domains.
```

または、

```
Could not start the domain.  
No default domain. Need to enter a domain.
```

このエラーは、同じシステム上に Application Server 8 がインストールされており、PATH 上でその `asadmin` コマンド (`/usr/sbin` 内) のほうが、`install_dir/bin` にある Application Server 8 の `asadmin` コマンドより前に見つかる場合に発生します。こうした状況になる可能性が特に高いのは、Solaris/Linux システムで、PATH 変数の一部として `.` が指定されていない場合です。PATH 内に `.` がないと、カレントディレクトリが `install_dir/bin` であっても、`/usr/sbin` 内の `asadmin` コマンドが先に見つかります。

解決法

PATH の中で `install_dir/bin` を `/usr/sbin` より前に指定するか、または `install_dir/bin` ディレクトリに移動してから `asadmin` にアクセスする場合は、PATH の中で `.` を `/usr/sbin` より前に指定します。`asadmin` を実行する際には必ず `install_dir/bin` に移動するのであれば、コマンド名に `./` を含めるようにしてもかまいません。次に例を示します。

```
cd install_dir/bin  
./asadmin
```

リブート時にドメインまたはノードエージェントを自動的に再起動する

マシンの再起動が必要になった場合など、ドメインまたはノードエージェントが予想外に停止される場合にそなえて、リブート時にドメインまたはノードエージェントが自動的に再起動されるようにシステムを設定することが可能です。

- [28 ページの「UNIX プラットフォーム上での自動再起動」](#)
- [29 ページの「Microsoft Windows プラットフォーム上での自動再起動」](#)

UNIX プラットフォーム上での自動再起動

ドメインの再起動

UNIX プラットフォーム上でドメインを再起動するには、適切な `asadmin start-domain` コマンドを含む行を `/etc/inittab` ファイルに追加します。`/etc/rc.local` またはシステム固有の同等のファイルを使用する場合は、目的の `asadmin` コマンドを `/etc/rc.local` に記述します。

たとえば、/opt/SUNWappserver ディレクトリにインストールされた Application Server 用の domain1 を password.txt という名前のパスワードファイルを使用して再起動するには、次の行を /etc/inittab または /etc/rc.local に追加します。

```
das:3:respawn:/opt/SUNWappserver/bin/asadmin start-domain --user admin
--passwordfile /opt/SUNWappserver/password.txt domain1
```

テキストは必ず 1 行で記述してください。先頭の 3 文字はこのプロセスに対する一意の指示子ですが、これは変更可能です。

ノードエージェントの再起動

ノードエージェントを再起動する場合の構文も、これと似ています。たとえば、/opt/SUNWappserver ディレクトリにインストールされた Application Server 用の agent1 を password.txt という名前のパスワードファイルを使用して再起動するには、次の行を /etc/inittab または /etc/rc.local に追加します。

```
das:3:respawn:/opt/SUNWappserver/bin/asadmin start-node-agent --user admin
--passwordfile /opt/SUNWappserver/password.txt agent1
```

Microsoft Windows プラットフォーム上での自動再起動

Microsoft Windows 上で自動的に再起動するには、Windows サービスを作成します。Sun Java System Application Server に同梱されている実行可能ファイル appservService.exe と appserverAgentService.exe を、Microsoft が提供するサービス制御コマンド (sc.exe) と組み合わせて使用します。

- sc.exe コマンドは Windows XP に同梱されており、C:\windows\system32、C:\winnt\system32 のいずれかのディレクトリに格納されています。
- このマニュアルの執筆時点では、Windows 2000 の sc.exe を次の場所からダウンロードできます。<http://ftp.microsoft.com/reskit/win2000/sc.zip>。sc.exe の使い方の詳細は、次を参照してください。

http://msdn.microsoft.com/library/default.asp?url=/library/en-us/dndllpro/html/msdn_scmslite.asp。

appservService.exe および appservAgentService.exe の使用方法は次のとおりです。

```
C:\winnt\system32\sc.exe create service_name binPath= \"fully_qualified_path_to_appservService.exe \"fully_qualified_path_to_asadmin.bat start_command\"
\"fully_qualified_path_to_asadmin.bat stop_command\" start= auto
DisplayName= \"display_name\"
```

ドメインの起動と停止

パスワードファイル C:\Sun\AppServer\password.txt を使用して domain1 を起動および停止する、SunJavaSystemAppServer DOMAIN1 という名前のサービスを作成するには、次のコマンドを実行します。

```
C:\windows\system32\sc.exe create domain1 binPath= "C:\Sun\AppServer\lib\appservService.exe \"C:\Sun\AppServer\bin\asadmin.bat start-domain --user admin --passwordfile C:\Sun\AppServer\password.txt domain1\" \"C:\Sun\AppServer\bin\asadmin.bat stop-domain domain1\" start=auto DisplayName= "SunJavaSystemAppServer DOMAIN1"
```

ノードエージェントの起動と停止

ノードエージェント agent1 を起動および停止するサービスを作成するには、次のコマンドを実行します。

```
C:\windows\system32\sc.exe create agent1 binPath= "C:\Sun\AppServer\lib\appservAgentService.exe \"C:\Sun\AppServer\bin\asadmin.bat start-node-agent --user admin --passwordfile C:\Sun\AppServer\password.txt agent1\" \"C:\Sun\AppServer\bin\asadmin.bat stop-node-agent agent1\" start=auto DisplayName="SJESAS_SE8.1 AGENT1"
```

注 - binPath= パラメータの一部として入力された起動コマンドと停止コマンドは、正しい構文で記述されている必要があります。確認するには、それらのコマンドをコマンドプロンプトから実行します。コマンドを実行してもドメインまたはノードエージェントが正常に起動または停止されない場合、そのサービスは正しく動作しません。

また、asadmin の start/stop コマンドと service start/stop を混在させないでください。両者を混在させると、サーバーの状態の同期が取れなくなります。たとえば、サーバーのコンポーネントが実行されていないのに「コンポーネントが開始された」と表示されたりします。こうした状況を避けるには、サービス使用時には常に、sc.exe コマンドを使ってコンポーネントを開始および停止するようにしてください。

自動再起動時のセキュリティ

起動時に必要となるパスワードとマスターパスワードは、次のいずれかの方法を使って処理します。

- Microsoft Windows 上で、ユーザーにパスワードを尋ねるようにサービスを設定します。
 - サービスコントロールパネルで、作成したサービスをダブルクリックします。
 - 「プロパティ」ウィンドウの「ログオン」タブをクリックします。

- 「デスクトップとの対話をサービスに許可」にチェックマークを付け、必要なパスワードに対するプロンプトがコンポーネント起動時に表示されるようにします。

ログインしてプロンプトを表示させ、入力時にエントリがエコーバックされないことを確認する必要があります。これがサービスオプションを使用する際のもっとも安全な方法ですが、この方法の場合、ユーザーが関与しないとサービスが利用可能になりません。

デスクトップとの対話オプションを設定しなかった場合、サービスは「開始保留」状態のままになり、ハングアップしたように見えます。この状態から抜け出すには、このサービスのプロセスを終了してください。

- Windows または UNIX 上で、`--savemasterpassword=true` オプションを使ってドメインを作成し、管理パスワード格納用のパスワードファイルを作成します。コンポーネント起動時に、`--passwordfile` オプションを使ってパスワードが格納されたファイルを指定します。`asadmin start` コマンドの `--password` オプションを使って管理パスワードを追加することもできます。この方法は、管理パスワードを平文で格納するため、安全性が低いことに注意してください。

次に例を示します。

- ドメイン作成時にマスターパスワードを保存します。この構文では、ユーザーは管理パスワードとマスターパスワードの入力を求められます。

```
asadmin create-domain --adminport 4848 --adminuser admin
--savemasterpassword=true --instanceport 8080 domain1
```

- Windows の場合、サービスを作成します。その際、パスワードファイルを使って管理パスワードを提供します。

```
C:\windows\system32\sc.exe create domain1 binPath=
"C:\Sun\AppServer\lib\appservService.exe
\\"C:\Sun\AppServer\bin\asadmin.bat start-domain --user admin
--passwordfile C:\Sun\AppServer\password.txt domain1\\"
\\"C:\Sun\AppServer\bin\asadmin.bat stop-domain domain1\\"
start= auto DisplayName= "SJESAS_PE8.1 DOMAIN1"
```

パスワードファイル `password.txt` のパスは、`C:\Sun\AppServer\password.txt` です。このファイルには、パスワードが次の形式で格納されています。

```
AS_ADMIN_password=password
```

たとえば、パスワードが `adminadmin` の場合、次のようになります。

```
AS_ADMIN_password=adminadmin
```

- UNIX の場合、`inittab` ファイルに追加する行の中で、`--passwordfile` オプションを使用します。

```
das:3:respawn:/opt/SUNWappserver/bin/asadmin start-domain
--user admin --passwordfile /opt/SUNWappserver/password.txt domain1
```

パスワードファイル password.txt のパスは、
/opt/SUNWappserver/password.txt です。このファイルには、パスワードが次の形式で格納されています。

```
AS_ADMIN_password=password
```

たとえば、パスワードが adminadmin の場合、次のようになります。

```
AS_ADMIN_password=adminadmin
```

- 次のようにして、コマンド行のオプションに指定したパスワードを使ってサービスを作成します。

```
C:\windows\system32\sc.exe create domain1 binPath= "C:\Sun\AppServer\lib\appservService.exe \"C:\Sun\AppServer\bin\asadmin.bat start-domain --user admin --password adminadmin domain1\" \"C:\Sun\AppServer\bin\asadmin.bat stop-domain domain1\" start=auto
DisplayName="SJESAS_PE8.1 DOMAIN1"
```

ログファイルが見つからない

次の Application Server ログは、インストールの問題のトラブルシューティングに有効です。

- サーバーログファイル — サーバーの設定および配備の問題のトラブルシューティングに使用
- HTTP サーバーアクセスログ — HTTP サーバーの問題のトラブルシューティング、および Application Server インスタンスに入力される HTTP 要求の処理の追跡に使用

インストール、アンインストールの両方のプログラムでログファイルが作成され、これらのファイルにインストールおよびアンインストールのすべてのイベントが記録されます。これらのログファイルの主な目的は、トラブルシューティング情報を提供することです。

インストールプログラムのメッセージとログファイルに加え、Solaris の pkginfo や showrev、Linux の rpm などのオペレーティングシステムユーティリティーを使用してシステム情報を収集できます。

ログファイルのエントリには、試行されたアクション、アクションの結果、および該当する場合には、失敗の原因についての情報が含まれます。ログファイルに含まれるメッセージエントリの種類は、次のとおりです。

- **INFO** — これらのメッセージは、各インストールタスクが正常に完了したことを示します。

- **WARNING** — これらのメッセージは重大でない失敗を示します。警告メッセージには、通常、失敗の原因および性質についての情報が含まれ、考えられる救済策も示されます。

- **ERROR** — これらのメッセージは重大な失敗を示します。インストールまたはアンインストールの状態は「失敗」と報告されます。エラーメッセージには、通常、発生した問題の性質および原因についての詳細な情報が含まれます。

ドメイン固有のログは、`install_dir/domains/domain1/logs/` にあります。一般のサーバーインストールのログファイルは、次の場所にあります。

- **Solaris、root** ユーザーのインストール/アンインストール:

`/var/sadm/install/logs`

- **Solaris、root** 以外のインストール/アンインストール:

`/var/tmp`

- **Linux** インストール/アンインストール:

`/var/tmp`

- **Windows** インストール/アンインストール:

`%TEMP%`

ログファイル名は製品の配布形態ごとに異なりますが、プラットフォームとは無関係です。

- Enterprise Edition の SDK と Application Server 両方の配布では、メインのインストール/アンインストールログファイルは次のようになります。

`Install_Application_Server_8PE_<timestamp>.log`
`Uninstall_Application_Server_8PE_<timestamp>.log`

- Enterprise Edition の Application Server のみの配布では、低レベルのインストール/アンインストールログファイルは次のようになります。

`Sun_Java_System_Application_Server_Platform_Edition_install.B<timestamp>`
`Sun_Java_System_Application_Server_Platform_Edition_uninstall.B<timestamp>`

- Enterprise Edition の SDK の配布では、低レベルのインストール/アンインストールログファイルは次のようになります。

`Java_2_Platform_Enterprise_Edition_1.4_SDK_install.B<timestamp>`
`Java_2_Platform_Enterprise_Edition_1.4_SDK_uninstall.B<timestamp>`

- Enterprise Edition のメインのインストール/アンインストールログファイルは次のようになります。

```
Install_Application_Server_8EE_<timestamp>\>.log  
Uninstall_Application_Server_8EE_<timestamp>\>.log
```

- Enterprise Edition の低レベルのインストール/アンインストールログファイルは次のようになります。

```
Sun_Java_System_Application_Server_Enterprise_Edition_  
install.B<timestamp>\>  
Sun_Java_System_Application_Server_Enterprise_Edition_  
uninstall.B<timestamp>\>
```

ローカルサーバーのアクセスが失敗する (http://localhost:8080)

このエラーが発生したときは次の点をチェックします。

- [34 ページ](#)の「サーバーは起動したか?」
- [35 ページ](#)の「サーバーは正しいポートで起動したか?」

サーバーは起動したか?

説明

コンソールウィンドウがまだ開いているなら、次のメッセージを探します。

```
Domain domain Started
```

このメッセージの *domain* は、デフォルトドメインの名前です。これは、デフォルトドメインが正常に起動したことを示します。

コンソールウィンドウがすでに閉じているなら、次のログファイルにメッセージがないかチェックします。

```
install_dir/domains/domain1/logs/server.log
```

起動が正常に行われた場合は、ログファイルの末尾にコンソールのメッセージに似た、次のようなメッセージが見つかります。

```
[INFO][...][...][date&time][Application server startup complete .]
```

サーバーは正しいポートで起動したか？

説明

サーバーが、想定される番号とは異なるポートで稼働していることがあります。意図的にそのポートにインストールされたか、そのサーバーのインストール時にすでに別のサーバーがデフォルトポートで稼働していたと考えられます。

▼ サーバーが実際に使用しているポート番号を調べる

- 1 サーバーの設定ファイルを検査します。

```
install_dir/domains/domain1/config/domain.xml
```

- 2 http-listener 要素を見つけます。

- 3 port 属性の値を調べます。

サーバーの起動時には、必ず正しいポート番号を入力してください。

注-サーバーのデフォルトポート番号は8080ですが、次のような場合には、想定される値に変更が生じます。

- インストール時に別のポート番号が指定された。
 - 以前のインストールが存在する。
 - 指定したポート番号が、サーバーの起動時に別のアプリケーションによってすでに取得されている場合、ポート番号は、次に利用可能な大きい番号になります。たとえば、サーバーがデフォルトの8080ポートですでに稼働している場合、新しいApplication Server インスタンスではポート番号8081を使用します。2つのサーバーが稼働しているなら、ポート番号は8082に増え、以下同様に変化していきます。
-

リモートサーバーへのアクセスが失敗する

Application Server の開始ページを開こうとしたとき、最初の画面が表示されません。

次の点をチェックします。

- [36 ページの「サーバーをローカルで使用できるか？」](#)
- [36 ページの「プロキシの設定が問題を引き起こしていないか？」](#)

サーバーをローカルで使用できるか？

説明

サーバーが、Web からはアクセスできないがローカルでは稼働している場合、そのサーバーは実際に稼働しています。

解決法

サーバーがローカルで稼働していることを確認します。

▼ サーバーがローカルで稼働していることを確認する

- 1 サーバーが実行されているマシンにログオンします。
- 2 ローカルの **Web** ページに移動します。たとえば、デフォルトポートが 8080 の場合、次の場所に移動します。

`http://localhost:8080/`

開始ページが表示されるのであれば、サーバーにリモートでアクセスできない原因は Web 接続にあります。開始ページが表示されないときは、[34 ページの「サーバーは起動したか？」](#)を参照してください。

プロキシの設定が問題を引き起こしていないか？

説明

通常、サーバーは、自分自身が稼働しているホスト (localhost) から直接アクセス可能です。たとえば、デフォルトポート 8080 を使用すると次のようになります。

`http://localhost:8080/`

解決法

サーバーホストマシンがプロキシ経由で Web に接続していると、localhost 上で稼働しているサーバーインスタンスにアクセスできないことがあります。この問題を解決するために、次のいずれかを行います。

- localhost にアクセスするときはプロキシサーバーをバイパスするようにブラウザを設定します。設定手順については、ブラウザのヘルプシステムを参照してください。
- システムの完全修飾ホスト名または IP アドレスを使用します。次に例を示します。

`http://myhost.mydomain.com:8080/`

注-localhost マシンのホスト名およびドメインを検索するには、次の手順に従います。

- **Microsoft Windows** の場合 — デスクトップで「マイ コンピュータ」を右クリックし、ポップアップメニューから「プロパティ」を選択します。「システムのプロパティ」ダイアログが表示されます。「ネットワーク ID」をクリックするとコンピュータ名が表示されます。
 - **Solaris** または **Linux** の場合 — コマンドプロンプトに `hostname` と入力します。
-

管理コンソールを表示できない

管理コンソールは、管理機能を実行するためのユーザーインタフェースです。管理コンソールを表示できない場合、その理由は次のいずれかと考えられます。

- 37 ページの「[Application Server は稼働しているか?](#)」
- 37 ページの「[想定されるポートで管理コンソールが実行されているか?](#)」
- 38 ページの「[セキュリティマネージャーが無効になっていないか?](#)」

Application Server は稼働しているか?

説明

管理コンソールを使用する前に、サーバーが稼働している必要があります。

解決法

34 ページの「[サーバーは起動したか?](#)」を参照して、サーバーが稼働中かどうか調べます。

想定されるポートで管理コンソールが実行されているか?

説明

EE および SE の管理コンソールのデフォルトポート番号は 4849 です。PE の管理コンソールでは 4848 です。EE および SE のコンソールの URL には Secure HTTP (`https://servername:4849`) が必要なことに注意してください。これに対し、PE のコンソールでは標準 HTTP (`http://servername:4848`) を使用します。ただし、意図的

に別のポート番号にインストールされたか、サーバーの起動時にポートが使用中だったという理由で、想定されていない別のポート番号上で管理コンソールが実行されることもあります。

解決法

35 ページの「サーバーは正しいポートで起動したか？」にある、管理コンソールを実行するポートを確認するためのガイドラインを参照し、正しいポート番号と HTTP プロトコルを入力して管理コンソールを呼び出すようにします。

セキュリティーマネージャーが無効になっていないか？

説明

J2EE 1.4 仕様上、セキュリティーマネージャーは省略できません。Application Server で有効になっている必要があります。Application Server にセキュリティーマネージャーを無効にする設定インタフェースはないため、domain.xml 設定ファイルから次の行を削除するという直接的な編集操作を行わないかぎり無効にはできません。

```
<jvm-option>-Djava.security.policy=yourPolicy</jvm-option>
```

解決法

管理コンソールを表示するには、domain.xml ファイルに
-Djava.security.policy=yourPolicy オプションが存在する必要があります。

サーバーアプリケーションを使用できない

Application Server 経由で特定のアプリケーションを使用できない場合は、次の点をチェックします。

- 38 ページの「Application Server は稼働しているか？」
- 39 ページの「アプリケーションの配備は成功したか？」
- 39 ページの「Don't Prompt オプション使用時の無効なユーザーまたはパスワード」
- 40 ページの「管理者のユーザー名またはパスワードがわからない」

Application Server は稼働しているか？

説明

Application Server が稼働していなければ、アプリケーションは使用できません。

解決法

34 ページの「サーバーは起動したか?」を参照して、サーバーが稼動中かどうか調べます。サーバーアプリケーションを使用する前に、サーバーが稼働している必要があります。

アプリケーションの配備は成功したか?

説明

アプリケーションを使用するには、その前に配備を正常に行う必要があります。

解決法

次のサーバーのログファイルをチェックします。

```
install_dir/domains/domain1/server.log
```

Don't Prompt オプション使用時の無効なユーザーまたはパスワード

「無効なユーザーまたはパスワード」のエラーが表示されますが、Don't Prompt オプションを指定してシステムをインストールした場合、パスワードは自動的に入力されます。

説明

インストール時に適切なパスワードを指定しなかったか、ドメインの起動時にパスワードが渡されていない可能性があります。

解決法

.asadminprefs ファイル内のパスワードをチェックします。UNIX/Linux システムでは、このファイルはサーバーがインストールされた、ユーザーのホームディレクトリにあります。Windows では、C:\Documents and Settings\username にあります。ファイルの内容は次のようになっています。

```
AS_ADMIN_USER=admin  
AS_ADMIN_PASSWORD=administrator
```

管理者のユーザー名またはパスワードがわからない

管理者のユーザー名を忘れた場合は、前記のセクションの説明に従って `.adminprefs` ファイル内を検索して見つけることができます。

`install_dir/domains/domain1/config/keyfile` を調べることもできます。domain1 はデフォルトドメインです。別のドメインの場合は、パス内の名前を置き換えてください。

管理者のパスワードを忘れた場合は、以前のユーザー名とパスワードを削除して新規に作成し、サーバーを再起動することで、新しいユーザー名とパスワードのペアを使用する必要があります。(パスワードは `keyfile` 内で暗号化されているため、読み取れません。)

▼ ユーザー名とパスワードを完全に削除する

- 1 サーバーが稼働中であれば停止します。
- 2 適切な `WEB-INF` ディレクトリに移動します。次に例を示します。

```
install_dir/lib/install/applications/adminapp/adminapp_war/WEB-INF
```

- 3 `web.xml` ファイル内の `<security-constraint>` 要素全体をコメントにします。
あとで再度有効にできるように、要素の削除はしないでください。この操作により、コマンド行操作のセキュリティが無効になります。

注 - コマンド行で `--username` (または `-u`) と `--password` (または `-w`) に値を指定することには変わりはありませんが、サーバー側でセキュリティはまったく要求されない
ので、ダミーの値でかまいません。

- 4 サーバーを起動します。
この時点で、サーバーにはコマンド行のセキュリティがありません。

- 5 次のコマンドを実行します。

```
asadmin create-file-user --user <dummy> --password <dummy>  
--userpassword <new_secret> --groups asadmin <new_user_id>
```

このコマンドで、次のような新規エントリが作成されます。

```
<install_dir>/domains/domain1/config/keyfile
```

- 6 `web.xml` ファイル内の `<security-constraint>` 要素のコメントを解除します。
- 7 サーバーを再起動して、新しいユーザー名-パスワードを有効にします。

注-サーバーを起動する際、すべてのリモートコマンド行操作で、ユーザー名とパスワードとして `new_user_id` および `new_secret` の指定が必要です。

Windows 上でサーバーが起動しない (ポートの衝突)

Microsoft Windows で Application Server の起動時に次のようなメッセージが表示された場合は、サーバーポートの衝突が発生しています。

Address already in use

このエラーは、Application Server ポート (デフォルトは **8080**) 上で別のアプリケーションが実行されているか、Application Server の以前のインスタンスが正常な方法でシャットダウンしなかった場合に発生します。

それで次の点をチェックします。

- 41 ページの「サーバーのポートで別のアプリケーションが実行されていないか？」
- 41 ページの「以前に稼働していたサーバーで異常な方法での停止が発生しなかったか？」

サーバーのポートで別のアプリケーションが実行されていないか？

別のアプリケーションがサーバーのポートを使用している場合は、そのアプリケーションを停止してから Application Server を再起動します。

注-インストーラには、デフォルトポートが使用中の場合に次に利用可能なポートを選択することによってポートの衝突を回避する機能がありますが、それが機能するのは、デフォルトポートを使用するアプリケーションが、Application Server のインストール時に実行されていた場合だけです。

以前に稼働していたサーバーで異常な方法での停止が発生しなかったか？

`asadmin stop-domain` コマンドを使用してサーバーを停止するか、Java プロセスを `kill` で明示的に終了してから Application Server を再起動します。

同じサーバー上の複数インスタンスのデバッグ時にポートが衝突する

説明

同じサーバー上にあり同じクラスタに属する複数のインスタンスをデバッグしているときに、ポートの衝突エラーが発生することがあります。

解決法

domain.xml ファイルを変更して、クラスタの java-config 要素にある -Xrunjdpw オプションから address 属性を削除します。これにより、JVM でインスタンス用にランダムデバッグポートが選択されます。インスタンス用に選択されたポート番号は、起動時にサーバーログに表示されます。次に例を示します。

- 変更前:

```
debug-options="-Xdebug -Xrunjdpw:transport=dt_socket,server=y,
suspend=n,address=9009"
```

- 変更後:

```
debug-options="-Xdebug -Xrunjdpw:transport=dt_socket,server=y,
suspend=n"
```

2つのサーバーインスタンスが **Windows** 上の同じポートにバインドされる

説明

この問題は、Application Server Enterprise Edition (Platform Edition ではない) ソフトウェアを実行する Windows 2000/XP システムでのみ発生し、Application Server の問題というよりも Windows の既知のセキュリティバグに原因があります。

Application Server の2つ以上のインスタンスを、instanceport オプションに同じポート番号を指定して作成すると、この問題が発生します。次に例を示します。

```
asadmin create-domain -adminport 5001 <options> -instanceport 6001
<domain>
asadmin create-domain -adminport 5002 <options> -instanceport 6001
<domain>
```

2つのドメインをUNIX/Linux システム上で起動すると、ポートの衝突エラーがスローされ、2つめのインスタンスの起動は失敗します。これに対し、2つのドメインをWindows 2000/XP 上で起動すると、エラーがスローされずに両方のサーバーインスタンスが起動しますが、指定したポートでは最初のインスタンスしか利用できません。その後、最初のサーバーインスタンスがシャットダウンすると、2つめのインスタンスが利用可能になります。さらに、両方のインスタンスが稼働している場合、Windows の `netstat` コマンドでは重複したリスナーがアクティブとして示されますが、要求に応答できるのは最初のリスナーのみです。

解決法

Windows システム上のすべてのサーバーインスタンスに、一意のポート番号を割り当てるようにします。

Error: 指定したパスがシステムで見つからない

説明

このエラーメッセージは、インストール時に指定した J2SE ディレクトリの削除後にサーバーを起動しようとする则表示されます。一般にこの状況は、J2SE プラットフォームのアップグレードが必要で、そのアップグレードは Application Server のインストール後に行われる、という通知がインストール中になされたあとに発生します。

解決法 1

すべてのドメインで新しい J2SE を使用するには、`asenv.conf` (Solaris/Linux) または `asenv.bat` (Windows) 内の `AS_JAVA` 変数を変更します。

解決法 2

J2SE のバージョンは、ドメインの `domain.xml` ファイルにある `java-config` 要素の `java-home` 属性を変更することによって、ドメインごとに変更できます。

```
<java-config ...  
  java-home="path"  
... \>
```

解決法 3

これは時間のかかる解決法で、サーバーをアンインストールしてから再インストールします。

アプリケーションで **Error**

`persistence.support.JDODataStoreException` が発生する

説明

`com.sun.jdo.api.persistence.support.JDODataStoreException` が、主キーの重複を示す入れ子にされた `java.sql.SQLException` とともに、アプリケーションから出力されます。

アプリケーションで `CreateException` の発生をチェックしていても、見つかることはありません。Enterprise JavaBeans の仕様上、同じ主キーを持つ 2 つの Bean が同じトランザクションで作成される場合にのみ、`CreateException` のスローが要求されます。したがって、主キーが重複する 2 つのエンティティ Bean が作成される場合でも、トランザクションロールバックで `CreateException` がスローされることはありません。

解決法

アプリケーションで主キーが重複するエンティティ Bean を作成する場合は、作成を呼び出す前に `findByPrimaryKey` を呼び出して主キーが存在するかどうかチェックします。

asadmin set コマンドを使用すると予期していない結果になることがある

説明

次のようなコマンドで変数を設定すると、予期していない結果が返されます。

```
asadmin set name=${a-b}
```

この場合、name は `{a-b}` ではなく `b` に設定されます。シェル構文 `${a=b}` は「変数 `a` が未設定ならば、値 `b` を代わりに使う、そうでなければ `a` の値を代わりに使う」と解釈されるからです。これは標準のシェル動作です。たとえば、次の例を考えてください。

```
asadmin set default-config.http-service.http-listener.http-listener-1.port=
${http-listener-1.port}
```

この場合、`default-config.http-service.http-listener.http-listener-1.port` は `listener-1-port` に設定されますが、この結果には意味がありません。

jar ファイルへのストリームが開いているためにアプリケーションを配備解除または再配備できない (Windows のみ)

説明

Windows システムでは、アプリケーションの実行後引き続きそのアプリケーションを配備解除または再配備しようとする、サーバーでファイルの削除やディレクトリ名の変更を行えないという例外がスローされます。

Windows システムでは、アプリケーションで `getClass().getResource` または `getResourceAsStream` メソッドを使用して、アプリケーション内部のリソース、特にアプリケーション内にあるまたはアプリケーションからアクセス可能な jar ファイルのリソースを取得できます。ストリームが開いたままになっていると、それ以降のアプリケーションの再配備または配備解除が失敗する可能性があります。加えて、Java ランタイムは、パフォーマンス上の理由からデフォルトで jar ファイルへのストリームをキャッシュします。

解決法

アプリケーションで開いたストリームは必ず閉じてください。さらに、アプリケーションの再配備または配備解除を繰り返す行う必要があり、アプリケーションで `getResource` または `getResourceAsStream` を使用して jar ファイルからリソースを取得する必要もある場合は、URL オブジェクトを返す `getClass().getResource` の使用を検討してください。その後、`url.setUseCaches` メソッドを呼び出してその jar ファイルのキャッシュを無効にしてから、`url.getInputStream()` を使用してストリームを取得します。

jar ファイルアクセスのキャッシュを無効にするとパフォーマンスが低下することがありますが、この手法により確実にアプリケーションを配備解除または再配備でき

ます。getClass().getResourceAsStream メソッドを代わりに使用すると、リソースが存在する jar ファイルがキャッシュされ(これがデフォルトの Java ランタイム設定)、サーバーが停止するまで開いたままになることにも注意してください。

ディレクトリの手動削除後 **Application Server** を再インストールできない

説明

Application Server のディレクトリを付属のアンインストールプログラムを利用しないで手動で削除すると、それ以降、同じディレクトリへの Application Server の再インストールが失敗します。これは、/tmp/productregistry ファイルに保存されたインストールディレクトリ情報が、プログラムディレクトリの削除後も残るからです。

解決法 1

/tmp/productregistry ファイルの <location> プロパティーエントリから Application Server のディレクトリ情報を削除します。たとえば、次の行を変更します。

```
<location>/opt/SUNWappserver/jdk</location>
```

変更後は次のようになります。

```
<location></location>
```

解決法 2

Application Server を別のディレクトリに再インストールします。

サーバークラッシュ後に **JVM** スレッドダンプを生成できない

説明

Application Server がクラッシュすると、サーバーからコアファイルがダンプされ、デフォルトでは -xrs フラグに基づいて再起動が行われますが、これにより JVM スレッドダンプの生成が妨げられます。

解決法

▼ JVM スレッドダンプを有効にする

- 1 **Application Server** の `server.xml` ファイルにある `-Xrs` フラグをコメントにします。
- 2 サーバプロセスを終了させます (**UNIX** では `kill -3`、**Windows** では **Ctrl+Break**)。

main スレッドでバンドル版の **ANT** から `java.lang.NoClassDefFoundError` がスローされる

Application Server 外の作業にバンドル版の Ant ディストリビューションを使用すると、main スレッドで `java.lang.NoClassDefFoundError: org/apache/tools/ant/launch/Launcher` がスローされることがあります。

解決法

バンドル版の ANT を Application Server 環境外の作業に使用することはお勧めできません。

分解したアプリケーションファイルを変更しても /generated/xml ディレクトリは更新されない

JES Service Registry (SR) は、JES 5 リリースの一部として Application Server 8.2 Enterprise Edition に配備されます。SR では、クライアント証明書認証を使用して、登録ユーザーを認証します。SR には、Application Server の TrustStore に格納されている RegistryOperator 証明書によって署名された x509 証明書を生成するオプションがあります。

問題は、このクライアント証明書認証手法が、生成される /generated/xml ディレクトリの `sun-web.xml` ファイルの場所に依存しているということです。配備したアプリケーションを分解して再配備すると、SR 側で `sun-web.xml` ファイルを検索する場所を知ることはできなくなります。

解決法

配備後の配備記述子の変更は推奨されていません。配備後に配備記述子を変更する必要がある場合は、`.reload` 機能を使用してアプリケーションの再ロードを強制しま

す。Service Registry 製品の場合は、ログイン情報が含まれるようにドメインの web.xml ファイルを変更する必要もあります。ログインで問題が発生している場合は、server.log をチェックして、アプリケーションが本当に再ロードされていることを再確認してください。

HADB の問題

ここでは、Application Server 8.2 Enterprise Edition 製品で、高可用性データベース (HADB) モジュール、HADB 管理クライアント、またはロードバランサプラグインを利用する際に発生する可能性のある問題について説明します。これらのコンポーネントは、Application Server の残りの部分と別にインストールすることも、ともにインストールすることもできます。

この章では、次の項目を説明します。

- 50 ページの「HADB データベースの作成が失敗する」
- 54 ページの「サーバーが応答する前に長いアイドル状態が発生する」
- 55 ページの「要求が成功しない」
- 56 ページの「セッション持続の問題」
- 58 ページの「HADB のパフォーマンスの問題」
- 61 ページの「高負荷の問題」
- 64 ページの「セマフォリソースの不足で発生する接続の問題」
- 65 ページの「CPU 利用率の向上」
- 65 ページの「HADB 管理の問題」
- 73 ページの「共有メモリーセグメントキーがすでに存在する (Windows のみ)」
- 73 ページの「configure-ha-cluster の失敗」
- 74 ページの「configure-ha-cluster を実行できない」
- 75 ページの「hadbm set コマンドが失敗する」
- 75 ページの「configure-ha-cluster の失敗: HADB インスタンスの作成が失敗する」

HADB データベースの作成が失敗する

ここでは、次のようにデータベースの作成が失敗する場合に考えられる理由について説明します。

failed to start database : HADB Database creation failed

問題の原因を識別するには、ログビューアを使用するか、*install_dir/hadb/4/log* ディレクトリを調べるか、またはその両方を行います。発生する可能性がある主なエラーは、次のとおりです。

- 50 ページの「共有メモリーに関連する問題」
- 52 ページの「セマフォが少なすぎる」
- 52 ページの「データベースノードにアクセスできないため、データベースが機能しない」
- 52 ページの「管理エージェントでドメインを確立できなかった」
- 53 ページの「*hadbm create* または *hadbm addnodes* コマンドがハングする」
- 54 ページの「*ma* (管理エージェントプロセス) に障害が発生する」

共有メモリーに関連する問題

説明

この問題は、次のいずれかの理由で発生する可能性があります。

原因 1

共有メモリーが設定されていないか、その設定が機能していない。

解決法 1

『Sun Java System Application Server 8.2 Installation Guide』で説明されている手順に従います。共有メモリーの設定後、システムをリブートしてください。

原因 2

物理メモリーの容量がノードの要件を満たしていない。次のエラーメッセージが表示される場合があります。

```
HADB-S-05512: Attaching shared memory segment with key <xx\> failed,  
OS status=12 OS message: Not enough space.
```

解決法 2

前述のように、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。

本稼働システムでは、ホスト上のノード数を減らすか、ホスト上の物理メモリーを増設します。

テストおよび開発システムでは、LogBufferSize および DataBufferPoolSize を設定して共有メモリー使用量を減らし、それぞれデフォルト値の 48 および 200MB よりも小さい値にします。これらの変数に設定可能な最小値は、それぞれ 32 および 64MB です。

原因 3

共有メモリーセグメントのサイズが、許容される最大サイズを超えている。

```
HADB-S-05510: Getting shared memory segment with key <xx\> failed,  
OS status=22. OS message: Invalid argument.
```

解決法 3

前述のように、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。

共有メモリーが正しく設定されている場合は、いずれかの共有メモリーセグメントサイズ (LogBufferSize または DataBufferPoolSize) を、オペレーティングシステムの設定ファイルで設定されているシステム設定済みの最大値 (Solaris では /etc/system の shmsys:shminfo_shmmax) より大きく設定していないかどうかチェックします。

原因 4

指定した識別子で共有メモリーセグメントがすでに作成されている。

```
HADB-S-05515: Shared memory segment with key <segment_key\> exists already.
```

解決法 4

共有メモリーセグメントの一覧を表示してチェックします。UNIX では、ipcs を使用してセグメントのリストを表示できます。Windows では、メモリーがマップされたファイルを使用して共有メモリーを実現しています。HADB では、getTempPath システムコールを使用して、f_segmentid という名前を持つこれらのファイルが格納されているシステム定義の一時ディレクトリを取得します。

稼働中の別のデータベースやその他のプログラムが、この識別子の共有メモリーセグメントをすでに使用していないかどうかチェックします。使用している場合は、別のポートベースでデータベースを作成します。このセグメントを使用している稼働中のデータベースやその他のプログラムがない場合は、hadbm delete unused_database でセグメントを解放します。

セグメントが解放されているかどうかチェックします。セグメントがまだある場合は、それらを削除します (UNIX では `ipcrm` を使用、Windows では `$TMP/f_*` を削除)。ファイル名は、`f_` プレフィックスのあとに、16 進に変換された `segment_key` が続く形式になっています。たとえば、エラーメッセージにセグメントキー `15201` がまだ残っていると表示された場合、その一時ファイルの名前は `f_3B61` になります。

セマフォが少なすぎる

説明

HADB-E-05521: Operation on semaphore with key "46025" failed, OS status=28 :
No space left on device

このエラーは、セマフォの数が少なすぎる場合に発生することがあります。セマフォはオペレーティングシステムからグローバルリソースとして提供されるため、その設定は、HADB だけでなくホスト上で実行されるすべてのプロセスに依存しています。このエラーは、HADB の起動中または実行中に発生する可能性があります。

解決法

`/etc/system` ファイルを編集して、セマフォの設定を変更します。手順とガイドラインについては、『Sun Java System Application Server Installation Guide』の「Preparing for HADB Setup」の章にある「Configuring Shared Memory and Semaphores」のセクションを参照してください。

データベースノードにアクセスできないため、データベースが機能しない

解決法

関係するホストの IP アドレスを固定してください。HADB では、データベース作成時に存在した固定 IP アドレスを使用するため、本稼働システムで動的 IP アドレス (DHCP) は使用できません。

管理エージェントでドメインを確立できなかった

説明

HADB 管理システムは、マルチキャストアドレス `228.8.8.8` 上の UDP マルチキャストメッセージに依存しています。これらのメッセージを伝達できない場合、`createdomain` コマンドが失敗して次のメッセージが表示されます。

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

考えられる原因は次のとおりです。

- エージェントが、異なるサブネットに複数のネットワークインタフェースで接続するホスト上で実行されている。
- マルチキャストメッセージを転送しないスイッチがネットワーク上に存在する。
- アドレス 228.8.8.8 のマルチキャストメッセージを経路指定しないルーターがネットワーク上に存在する。
- マルチキャストメッセージがオペレーティングシステム (たとえば、Solaris 10) で無効にされている。

解決法 1

ホスト上に、異なるサブネットに接続する複数のネットワークインタフェースがある場合は、それらのサブネットの 1 つを使用するように管理エージェントを設定する必要があります。 `ma.server.mainternal.interfaces` 属性を設定します。

解決法 2

マルチキャストメッセージのサポートに必要なネットワークインフラストラクチャーを構成します。

hadbm create または hadbm addnodes コマンドがハングする

説明

`hadbm create` または `addnodes` に指定したホストリストの一部のホストに複数のネットワークインタフェースがあり、ほかのホストにはネットワークインタフェースが 1 つしかない場合、`hadbm create/addnodes` コマンドがハングします。

解決法

ネットワークインタフェースを複数備えるホストでは、`hadbm create/addnodes` の実行時に `hadb` で使用するネットワークインタフェースの IP アドレスをドット区切りで指定します (たとえば、129.241.111.23)。IP アドレスの代わりにホスト名を使用すると、ホストに登録された最初のインタフェースが使用されるため、ノードが通信可能状態になる保証はありません。

ma (管理エージェントプロセス) に障害が発生する

説明

ma (管理エージェントプロセス) には、さまざまな理由で障害が発生します。

解決法

hadbm listdomain を使用して診断情報を表示します。一般に、対処法は障害の発生したエージェントを再起動することです。それで解決しない場合は、すべてのエージェントを順番に再起動します。

サーバーが応答する前に長いアイドル状態が発生する

説明

長いアイドル状態のあとでサーバーが要求を処理するため応答に長い時間がかかり、サーバーログに次のような「接続が失われた」というメッセージが表示されます。

```
java.io.IOException:..HA Store: Lost connection to the server.
```

そのような状況では、サーバーで HADB 用の JDBC プールを再作成する必要があります。

解決法

タイムアウト値を変更します。デフォルトの HADB 接続タイムアウト値は 1800 秒です。この期間中にアプリケーションサーバーから JDBC 接続を介して要求が 1 つも送信されないと、HADB によって接続が閉じられるため、アプリケーションサーバーで接続を再確立する必要が生じます。タイムアウト値を変更するには、hadbm set SessionTimeout= コマンドを使用します。

注-HADB 接続タイムアウトがJDBC 接続プールタイムアウトよりも必ず大きくなるようにしてください。JDBC 接続タイムアウトがHADB 接続タイムアウトよりも大きいと、接続がHADB 側から閉じられ、アプリケーションサーバーの接続プールに残ることになります。その後、アプリケーションでその接続を使用しようとすると、アプリケーションサーバーで接続を再作成する必要性が生じ、重大なオーバーヘッドを招きます。

要求が成功しない

ここでは、次の問題について説明しています。

- 55 ページの「ロードバランサのタイムアウトは適切か？」
- 55 ページの「システムクロックは同期しているか？」
- 56 ページの「Application Server はHADB と通信しているか？」

ロードバランサのタイムアウトは適切か？

説明

loadbalancer.xml ファイルの `response-timeout-in-seconds` プロパティを設定する場合は、実行中のすべてのアプリケーションの最大タイムアウトを考慮に入れる必要があります。この応答タイムアウトを非常に小さい値に設定すると、Application Server が要求に応答する前にロードバランサがタイムアウトするため、実行中の多数の要求が失敗します。

逆に、この応答タイムアウトを異常に大きい値に設定すると、応答を中止したインスタンスのキューに要求が格納されるため、多数の要求が失敗することになります。

解決法

`response-timeout-in-seconds` の値をすべてのアプリケーションのうちで最長の応答時間に設定します。

システムクロックは同期しているか？

説明

セッションがHADB に格納される際には、そのセッションの最終アクセス時刻、最終変更時刻などの時刻情報も保存されます。クロックが同期していないと、インス

タンスに障害が発生し別のマシン上の別のインスタンスがテイクオーバーする場合に、後者のインスタンスで、セッションの有効期限が切れていないのに切れていると判断されたり、さらに悪いことにセッションの最終アクセス時刻が将来の時刻になったりすることがあります。

注-非共存構成では、HADB ノードをホスティングしているそれらのマシン上のクロックを同期させることは重要です。詳しくは、『Installation Guide』の「Preparing for HADB Setup」の章を参照してください。

解決法

クラスタ内のすべてのシステムでクロックが同期していることを確認します。

Application Server は HADB と通信しているか？

説明

HADB が作成され稼働していても、持続性ストアが未作成の場合は、Application Server は HADB と通信できません。このような状況では、次のメッセージが表示されます。

```
WARNING (7715): ConnectionUtilgetConnectionsFromPool failed using
connection URL: connection URL
```

解決法

次のようなコマンドを使用して、HADB 内にセッションストアを作成します。

```
asadmin create-session-store --storeurl connection URL --storeuser haadmin
--storepassword hapasswd --dbsystempassword super123
```

セッション持続の問題

ここでは、次の問題について説明しています。

- [57 ページの「create-session-store コマンドが失敗した」](#)
- [57 ページの「インスタンスレベルのセッション持続設定が機能しなかった」](#)
- [57 ページの「セッションデータが壊れている可能性がある」](#)

create-session-store コマンドが失敗した

説明

asadmin create-session-store コマンドは、ファイアウォールを越えて実行できません。したがって、create-session-store コマンドが機能するには、アプリケーションサーバーインスタンスと HADB がファイアウォールの同じ側に存在する必要があります。

create-session-store コマンドは HADB と通信しますが、アプリケーションサーバーインスタンスとは通信しません。

解決法

HADB とアプリケーションサーバーインスタンスをファイアウォールの同じ側に配置します。

インスタンスレベルのセッション持続設定が機能しなかった

アプリケーションレベルのセッション持続設定は、インスタンスレベルのセッション持続設定より常に優先されます。アプリケーションを配備したあとにインスタンスレベルのセッション持続設定を変更した場合でも、アプリケーションの設定値がアプリケーションサーバーインスタンスの設定値より優先されます。

セッションデータが壊れている可能性がある

説明

次のような状況でシステムログからエラーが報告される場合、セッションデータが壊れている可能性があります。

- セッション持続中
 - セッション有効化中のセッション状態読み取り時
 - セッションフェイルオーバー後のセッション状態読み取り時
- データが壊れている場合、セッションストアを一貫性のある状態に戻すため、考えられる解決法が次に示すように3つあります。

解決法 1

asadmin clear-session-store コマンドを使用して、セッションストアをクリアします。

解決法 2

セッションストアのクリアーで解決しない場合は、すべてのノード上のデータ領域を再初期化し、`hadbm clear` コマンドを使用して HADB 内のデータをクリアーします。

解決法 3

HADB のクリアーで解決しない場合は、データベースを削除して再作成します。

前記の解決法 2 と 3 では、HADB のクリアー後、セッションストアを再作成してデータベーススキーマを再確立します。

HADB のパフォーマンスの問題

HADB へのトランザクションが遅延したり異常終了したりすると、パフォーマンスが影響を受けます。一般にこうした状況は、システムリソースの不足が原因です。待機時間が 5 秒を超えると、トランザクションの異常終了を引き起こします。ノードの障害も、クラッシュ時にそのノード上のアクティブトランザクションが異常終了する原因となります。二重障害 (両方のミラーの障害) が発生すると、HADB が使用不能になります。障害の原因は、通常、HADB の履歴ファイルに記録されています。

問題を特定するには、次の点を考慮します。

- 58 ページの「CPU またはメモリーのリソースが不足していないか、あるいはスワッピングが多すぎないか？」
- 59 ページの「ディスクの競合はないか？」
- 60 ページの「HADB データデバイス空間が不足していないか？」
- 61 ページの「ほかの HADB リソースが不足していないか？」

CPU またはメモリーのリソースが不足していないか、あるいはスワッピングが多すぎないか？

説明

「Process blocked for *x* sec, max block time is 2.500000 sec」というエラーが原因で、ノードの再起動または二重障害が発生します。この場合、*x* はプロセスがブロックされた時間の長さで、それが 2.5 秒を超えました。

HADB ノードスーパーバイザプロセス (NSUP/clu_nsup_srv) では、前回、プロセスで何らかの監視操作を実行した時刻からの経過時間を追跡しています。その時間が指定された最大値 (デフォルトでは 2500ms) を超えると、NSUP は、ブロックされた時間が長すぎると判断しノードを再起動します。

NSUPが2.5秒より長くブロックされると、ノードの再起動が引き起こされます。ミラーノードが同じホスト上に配置されていると、二重障害の恐れが高まります。ミラーホスト上でブロッキングが同時に発生するときも、二重障害になる可能性があります。

こうした状況が発生する恐れが特に高いのは、共存構成の場合のように、システム内にCPUやメモリーの使用で競合するほかのプロセスが存在する場合です。プロセスが再スケジュールされるたびに、大規模なスワッピングや複数のページフォルトが発生するからです。

システムクロックの負の調整もNSUPがブロックされる原因になります。

解決法

HADB ノードに十分なシステムリソースが割り当てられるようにします。また、時刻同期デーモンで2秒を超える大幅な調整が行われないようにします。

ディスクの競合はないか？

説明

ディスクの競合は、ディスク装置に対するユーザーデータの読み取り/書き込み、およびHADBから履歴ファイルへの書き込みに悪影響を与えることがあります。重度のディスク競合は、ユーザートランザクションの遅延や異常終了の原因となります。履歴ファイルへの書き込みが遅延すると、ノードの再起動が発生することがあり、最悪の場合、二重障害になります。

ディスクの競合は、データ、ログデバイス、および履歴ファイル用に使用しているディスクのディスク入出力をOSから監視することで識別できます。履歴ファイルへの書き込み遅延は、HADB履歴ファイルに記録されます。「NSUP BEWARE timestamp Last flush took too long (x msec)」というメッセージが遅延を表しています。

この警告は、ディスク入出力に要した時間が異常に長いという意味です。遅延が10秒を超えると、ノードスーパーバイザによってtransプロセスが再起動され、次のようなエラーメッセージが表示されます。

```
Child process trans0 10938 does not respond.
Child died - restarting nsup.
Psup::stop: stopping all processes.
```

このメッセージは、trans (clu_trans_srv) プロセスが、ほかの動作 (たとえば、履歴ファイルへの書き込みの待機中) のためにビジー状態になったままで、trans プロセスのハートビートをチェックするノードスーパーバイザの要求に応答できないことを示しています。このため、nsupでtransが終了していると判断され、プロセスが再起動されます。

プロセスが多すぎる (多くの HADB ノードはほかのプロセスと共存している) ためにオペレーティングシステムがオーバーロード状態になると、入出力動作のスケジューリングに遅延が生じることがあります。HADB 関連の入出力処理が遅延すると、HADB ノードから履歴ファイルに「HADB warning: Schedule of async <read,write> operation took ...」というメッセージが書き込まれます。

この問題は特に、Red Hat AS 2.1 で複数の HADB ノードが同じホスト上に配置され、すべてのノードでデバイスの配置に同じディスクを使用する場合に生じることが報告されています。

解決法

ノードで使用するデバイスの配置には、ノードごとに1つのディスクを使用します。ノードに複数のデータデバイスがあってディスクの競合が生じている場合は、1つのデータデバイスを別のディスクに移動します。履歴ファイルにも同じ方法を適用できます。

すべてのデータおよびログデバイス、またすべての履歴ファイルが、NFS マウントディスクやほかのリモートマウントディスクではなく、ローカルディスク上に存在することを確認してください。

監視ツールで HADB ディスクの競合がなくならない場合は、データバッファプールサイズを大きくすることができます。

HADB データデバイス空間が不足していないか？

説明

トランザクションが失敗する1つの理由として、データデバイス空間の不足も考えられます。このような状況が生じると、HADB から履歴ファイルに警告が書き込まれ、データを挿入したり更新したりするトランザクションが異常終了します。

通常、次のようなメッセージが表示されます。

```
HIGH LOAD: about to run out of device space, ...
```

```
HIGH LOAD: about to run out of device space on mirror node, ...
```

一般には、データデバイスにはユーザーデータの4倍以上の空き容量が必要とされています。詳しくは、『パフォーマンスチューニングガイド』を参照してください。

解決法 1

次のコマンドを使用して、データデバイスのサイズを大きくします。

```
hadbm set DeviceSize=size
```

この解決法は、すべてのノードで HADB データデバイス用に使用されている物理ディスクに空き容量がある場合にのみ適用できます。

HADBM により、データベースの各ノードが自動的に再起動されます。

解決法 2

HADB を停止して削除し、ノード数を増やしたり、データデバイスのサイズを大きくしたり、ノードごとに複数のデータデバイスを配置したりした、新しいインスタンスを作成します。ただし、この解決法では、Application Server で作成したすべての持続データおよびスキーマが消去されます。この手順の詳細については、『管理ガイド』を参照してください。

ほかの HADB リソースが不足していないか？

HADB ノードの起動時に、次の割り当てが行われます。

- 固定サイズの複数の共有メモリーセグメント
- 固定サイズの内部データ構造

HADB ノードでリソースの不足が発生すると、トランザクションが遅延したり異常終了したりします。ミラーノード間でリソース使用量情報がやり取りされるので、ミラーノード上で失敗する可能性が高い操作を、ノード側で遅延させたり中止したりできます。

トランザクションが繰り返し遅延すると、タイムアウトになってクライアントにエラーメッセージが返される場合があります。タイムアウトしない場合、クライアント側でわかるのは、システムでリソースが不足している時間帯にはパフォーマンスが低下するという現象のみです。

これらの問題は、「高負荷」の状況で頻繁に発生します。詳しくは、[61 ページ](#)の「[高負荷の問題](#)」を参照してください。

高負荷の問題

次のような症状が出るシナリオは、高負荷であると判断できます。

- ユーザー要求が失敗する
- データベースで複数のタイムアウトが発生し、「トランザクションが異常終了した」というメッセージが表示される
- 履歴ファイルに「HIGH LOAD」警告が頻繁に出力される
- 障害が散発する

高負荷の問題が疑われる場合は、次の点を考慮してください。

- 62 ページの「タプルログの容量が不足していないか?」
- 63 ページの「node-internal ログがいっぱいになっていないか?」
- 63 ページの「ロックの数は十分か?」
- 64 ページの「パフォーマンスチューニングを行えば問題を解決できるか?」

注-多くの場合、これらの問題すべては、より多くの CPU パワーを利用できるようにすることで解決できます。

タプルログの容量が不足していないか?

すべてのユーザー操作 (delete、insert、update) はタプルログに記録されて、実行されます。タプルログは、次の理由でいっぱいになる可能性があります。

- CPU またはディスク入出力の競合が原因で実行速度が低下する
- ミラーノードでログレコードを受信する速度が遅い。これには次のような原因があります。
 - ネットワーク競合のため、ログレコードをミラーノードに転送できない
 - ミラーノード側の CPU およびディスクの競合のため、受信済みのログレコードの処理速度が不十分 (履歴ファイルの「log throw due to...」メッセージ)。
タプルログの容量がなくなると、履歴ファイルにタプルログの HIGH LOAD を示すメッセージが出力されます。

解決法 1

65 ページの「CPU 利用率の向上」の説明に従って、CPU の使用量をチェックします。

解決法 2

CPU の利用状況に問題がなければ、ディスク入出力をチェックします。ディスク競合の兆候がある場合は、`hadbm set DataBufferPoolSize=...` でデータバッファサイズを大きくして、ログレコード処理中のページフォルトを回避します。ディスク競合が発生する場合は、59 ページの「ディスクの競合はないか?」で説明している解決法に従います。

解決法 3

ネットワーク競合の兆候がないかどうかチェックし、ボトルネックを解決します。

解決法 4

`hadbm set LogBufferSize=...` を使用して、タプルログバッファを大きくします。

node-internal ログがいっぱいになっていないか？

CPU または ディスク 入出力に問題があると、スケジュールされても処理されない node-internal 操作が異常に増加します。

node-internal ログの容量がなくなると、履歴ファイルに node-internal ログの HIGH LOAD を示すメッセージが出力されます。

解決法 1

65 ページの「CPU 利用率の向上」の説明に従って、CPU の使用量をチェックします。

解決法 2

CPU の利用状況に問題がなければ、ディスク入出力をチェックします。ディスク競合の兆候がある場合は、`hadbm set DataBufferPoolSize=...` でデータバッファサイズを大きくして、ログレコード処理中のページフォルトを回避します。ディスク競合が発生する場合は、59 ページの「ディスクの競合はないか？」で説明している解決法に従います。

ロックの数は十分か？

この状況を識別するための主な追加の症状は、次のとおりです。

- エラーコード 2080 または 2096 がクライアントに配信される。
- `hadbm resourceinfo --locks` で表示される割り当て済みのすべてのロックが、常時使用中である

解決法 1: 長いトランザクションを分割する

ノード上で実行されるトランザクションで、そのノードに割り当てられたロックの 25% を超える数のロックを使用することはできません。「repeatable read」遮断レベルで実行される読み取りトランザクション、および更新/挿入/削除トランザクションでは、トランザクションが終了するまでロックが保持されます。したがって、長いトランザクションを小さいトランザクションに分割することをお勧めします。

解決法 2: ロックの数を増やす

`hadbm set NumberOfLocks=` を使用して、ロックの数を増やします。

パフォーマンスチューニングを行えば問題を解決できるか？

ほとんどの状況では、負荷を減らすかリソースの可用性を高めることでホストのパフォーマンスを改善できます。次に示す手順はさらに広範囲に適用できます。

- ハードウェアの性能が高い (大容量内部メモリー、高速プロセッサ、マルチプロセッサ) ホスト上でノードを実行します。
- 物理ディスクを追加し、複数のデータデバイスを使用します。1 台の物理ディスク上に複数のデバイスを配置しないようにします。
- 新しいホスト上にノードを追加し、新しいノードを活用できるようにデータを再断片化します。
- 設定変数を変更して、サイズの大きいメモリーセグメントまたは内部データ構造を割り当てます。

これに加え、『パフォーマンスチューニングガイド』の説明に従って次のリソースを調整することにより、「HIGH LOAD」の問題を改善できます。

表 3-1 HADB のパフォーマンスチューニングプロパティ

リソース	プロパティ
データベースバッファのサイズ	hadbm attribute DataBufferPoolSize
タブルログバッファのサイズ	hadbm attribute LogBufferSize
ノード内部ログバッファのサイズ	hadbm attribute InternalLogBufferSize
データベースロックの数	hadbm attribute NumberOfLocks

セマフォリソースの不足で発生する接続の問題

説明

この問題が発生すると、履歴ファイルに次のようなメッセージが出力されます。

HADB-E-05521: Operation on semaphore with key "46025" failed,
OS status=28 : No space left on device

ホストコンピュータ上にさらに多くのセマフォ構造を設定する必要があります。各オペレーティングシステムに semmnu を設定する方法については、『Sun Java System Application Server 8.2 Installation Guide』の「Preparing for HADB Setup」の章を参照してください。

解決法

影響を受けている HADB ノードを停止して、影響を受けているホストを再設定およびリブートし、HADB ノードを再起動します。HADB は、これらの処理中も使用可能です。

CPU 利用率の向上

説明

パフォーマンスは、使用可能な CPU サイクルおよび入出力容量によって重大な制限を受けることがあります。システムパフォーマンスを最適化するには、HADB を最適に設定することに加えて、そのような問題を解決および防止することが必要です。

解決法

利用されていない余分の CPU がホスト上にあれば、同じホストに新しいノードを追加します。それができない場合、新しいマシンを追加して、それらに新しいノードを追加します。

マシンに十分なメモリーがあれば、`DataBufferPoolSize` を増やし、ログファイルに警告を出力する可能性のあるほかの内部バッファのサイズを大きくします。それができない場合、新しいマシンを追加して、それらに新しいノードを追加します。

このセクションについての詳細は、『パフォーマンスチューニングガイド』を参照してください。

HADB 管理の問題

`hadbm` コマンドとその多数のサブコマンドおよびオプションを使用して、高可用性データベース (HADB) を管理することができます。`hadbm` コマンドは、`install_dir/SUNWhadb/4/bin` ディレクトリにあります。

このコマンドの詳細な説明については、『Sun Java System Application Server 管理ガイド』の「Configuring the High Availability Database」の章を参照してください。各 `hadbm` サブコマンドの詳細については、`hadbm` のマニュアルページを参照してください。

ここでは、次の問題について説明しています。

- 66 ページの「`hadbm` コマンドが失敗する: エージェントにアクセスできなかった」

- 66 ページの「`hadbm` コマンドが失敗する: コマンドが見つからない」
- 67 ページの「`hadbm` コマンドが失敗する: `JAVA_HOME` が定義されていない」
- 68 ページの「`hadbm createdomain` が失敗し、2 つに分割されたドメインが作成される」
- 68 ページの「`create` が失敗する: ホスト上にパスがない」
- 69 ページの「データベースが起動しない」
- 70 ページの「`clear` コマンドが失敗した」
- 71 ページの「`create-session-store` が失敗した」
- 72 ページの「`hadbm` コマンドがハングする」
- 72 ページの「HADB を再起動できない」

hadbm コマンドが失敗する: エージェントにアクセスできなかった

説明

コマンドが次のエラーで失敗します。

```
The agents <url\> could not be reached.
```

URL のホストにアクセスできなかった可能性があります。原因は、ホストがダウンしている、通信経路が確立されていない、URL のポート番号が違う、管理エージェントがダウンしている、のいずれかです。

解決法

URL が正しいかどうか確認します。URL が正しい場合は、ホストが稼動しておりすぐに通信できる状態にあることを確認します。次に例を示します。

```
ping hostname1ping hostname2...
```

hadbm コマンドが失敗する: コマンドが見つからない

説明

`hadbm` コマンドはカレントディレクトリから実行できます。検索 `PATH` を設定してどこからでも `hadb` コマンドを使用できるようにすれば、一層便利です。「`hadbm: Command not found`」というエラーは、これらの条件のいずれにも当てはまらなかったことを示します。

解決法 1

hadbm コマンドを含むディレクトリに cd で移動し、そこから実行します。

```
cd install_dir/SUNWhadb/4/bin/  
./hadbm
```

解決法 2

hadbm コマンドのフルパスを指定して呼び出します。

```
install_dir/SUNWhadb/4/bin/hadbm
```

解決法 3

PATH 変数を設定すると、どこからでも hadbm コマンドを使用できます。PATH 変数の設定手順については、『Sun Java System Application Server 8.2 Installation Guide』の「Preparing for HADB Setup」の章を参照してください。

PATH の設定が正しいかどうか確認するには、次のコマンドを実行します。

```
which asadmin  
which hadbm
```

ユーティリティに返されるパスを確認してください。

hadbm コマンドが失敗する: JAVA_HOME が定義されていない

説明

「hadbm: <path\>: Invalid Java home location」というメッセージは、JAVA_HOME 環境変数が適切に設定されていないことを示します。

解決法

システムに複数のバージョンの Java がインストールされている場合は、JAVA_HOME 環境変数に正しいバージョンの Java (Enterprise Edition では 1.4.1_03 以上) が設定されているかどうか確認します。

PATH 変数の設定手順については、『Sun Java System Application Server 8.2 Installation Guide』の「Preparing for HADB Setup」の章を参照してください。

hadbm createdomain が失敗し、2 つに分割されたドメインが作成される

説明

ネットワークインタフェースが複数あるホスト上で HADB 管理エージェントを実行する場合、それらのネットワークインタフェースがすべて同一サブネット上にないと、createdomain コマンドが次のように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.
```

管理エージェントのデフォルトでは、UDP マルチキャスト用に「最初の」(`java.net.NetworkInterface.getNetworkInterfaces()` で返される「最初の」) インタフェースが使用されます。

解決法

最善の解決法は、設定ファイルで `ma.server.mainternal.interfaces` を設定して、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。次に例を示します。

```
ma.server.mainternal.interfaces=10.11.100.0
```

サブネット間のルーターで、マルチキャストパケットを経路指定するように設定する方法もあります。デフォルトでは、管理エージェントで使用するマルチキャストアドレスは、228.8.8.8 です。

create が失敗する: ホスト上にパスがない

説明

`hadbm create` コマンドの実行後、コンソールに次のようなエラーが表示されます。

```
./hadbm create ...  
...  
hadbm: Error 22022: Path path does not exist on host host
```

このエラーメッセージは、マシン上に存在しないパスを指定して新しいノードを追加した場合にも表示されることがあります。

解決法

ホストにログインし、HADB デバイスおよび HADB 履歴ファイルのパスを作成します。hadbm create を実行し、作成したパスを --devicepath および --historypath オプションで指定します。ホスト上で管理エージェントを実行するユーザーに、これらのディレクトリに対する読み取りおよび書き込みアクセス権があることも確認してください。

注-HADB の実行可能ファイルを複数の異なるホスト上の異なるパスにインストールすることはできません。

データベースが起動しない

create または start コマンドが失敗し、次のコンソールエラーメッセージが表示されます。

```
hadbm: Error 22095: Database could not be started...
```

次の点を考慮してください。

共有メモリーのセグメント取得エラーがなかったか？

説明

ノードの停止後に、そのノードに割り当てられたリソース (共有メモリー、ディスク領域) がほかのプロセスに占有されていると、起動が失敗することがあります。

解決法

この問題を解決する方法については、[50 ページの「共有メモリーに関連する問題」](#)を参照してください。

履歴ファイルにエラーが記録されていないか？

説明

問題がなくなる場合は、HADB 履歴ファイルを調べます。見つかる可能性が高い主なエラーメッセージは、次のとおりです。

- Could not verify node address

このメッセージは、HADB サーバーが処理しようとしたポートを別のプロセスが使用している場合に出力されます。次のような状況が考えられます。

- `portBase` が、このホストマシン上で実行されている別のプロセスで使用されている。

次のコマンドを使用して、`PortBase` 属性を別の値に設定します。

```
hadbm set portbase=value
```

- 保守のために HADB ノードを停止しようとしたが失敗した。

`hadbm` コマンドを使用して、ノードの停止を再試行します。これが失敗する場合は、このノードの OS プロセス `clu_nsup_srv` を `-9` オプションを指定せずに `kill` します。これで、`nsup` プロセスの `hadb` 子プロセスが停止します。親プロセスの `nsup` が存在しない場合は、`kill -9` で、すべての子プロセスを `kill` します。

- 保守のために HADB ノードを停止したが、起動予定時刻より前に `inetd` プロセスによって HADB ノードが再起動された。

予定より前に `inetd` によって HADB ノードが起動されないように注意してください。

- `hadbm command fails with internal error: "The database could not be started"`
次のことをチェックします。

- HADB 設定内のすべてのマシン上の共有メモリが正常であること。
- マシン上でほかの HADB データベースが稼働していないこと。同じポート番号を使用する可能性のあるほかのプロセスも実行されていないこと。
- 必要なディレクトリがすべて存在し、それらに対する書き込み権があること。
- デバイスの書き込み先となるディレクトリに十分な容量があること。

解決法

前述の項目のいずれにも問題が発生していないことを確認したら、次の救済策を順番に試してください。

- データベースを削除して再試行します。
- データベースを削除し、リブートして再試行します。
- データベースを削除し、HADB ソフトウェアを再インストールして再試行します。
- [24 ページの「サポートに問い合わせる」](#)。

詳しくは、『Error Message Reference』を参照してください。

clear コマンドが失敗した

`clear` コマンドでは、ディスク上に存在するデータベースデバイスファイルを再初期化します。このコマンドは、ディスクまたはディスクアクセスの問題が原因で失敗することがあります。このことを示す `hadbm` からのエラーメッセージがないかどうか

かチェックします。もしないなら、`ma.log` ファイル内を検索して、`devinit` から出力されたエラーメッセージがないかどうかチェックします。

create-session-store が失敗した

`asadmin create-session-store` コマンドは、次のいずれかの理由で失敗することがあります。

- 71 ページの「無効なユーザー名またはパスワード」
- 71 ページの「SQLException: No suitable driver」

無効なユーザー名またはパスワード

このエラーは、`create-session-store` コマンドに指定した `--dbsystempassword` が、データベース作成時に指定したパスワードと異なる場合に発生します。

解決法 1

正しいパスワードを指定してコマンドを再試行します。

解決法 2

`dbssystem` パスワードを忘れた場合は、`hadbm clear` でデータベースをクリアし、新しい `dbssystem` システムパスワードを指定する必要があります。

SQLException: No suitable driver

`create-session-store` から次のエラーが出力されます: `SessionStoreException: java.sql.SQLException: No suitable driver.`

解決法 1

`asadmin` で、Application Server config ディレクトリの `asenv.conf` で定義した `AS_HADB` パスから `hadbjdbc4.jar` を見つけられない場合に、このエラーが発生することがあります。

解決法は、HADB のインストール場所を指すように `AS_HADB` を変更することです。

次に、`asenv.conf` ファイルから `AS_HADB` エントリの例を示します。

```
AS_HADB=/export/home0/hercules/0815/SUNWhadb/4.4.0-8
```

解決法 2

このエラーは、`--storeUrl` に間違った値を指定した場合にも発生することがあります。この問題を解決するには、`hadbm get jdbcURL` を使用して正しい URL を取得します。

hadbm コマンドがハングする

hadbm との間で通信する管理エージェントが、操作が完了する前に停止すると、hadbm プロセスがハングすることがあります。すべてのエージェントが稼働しているかどうかチェックしてください。

HADB を再起動できない

説明

二重ノード障害が起きると、HADB 再起動は機能しません。HADB を再起動するには、付加的な回復アクションが必要です。

二重ノード障害の主な症状は、次のとおりです。

- hadbm status で、HADB が作動不能状態であることが示される。
- node status で、ノードが Starting または Recovering 状態であることが示される。各ノードを停止し再起動したあとでも、ノードは Starting 状態のままです。最終的に、ノード状態は Stopped に変わります。

この問題は、停電後などにミラー HADB ホストマシンで障害が発生したりリブートが行われたりした場合、単一マシンインストールで HADB を先に停止することなくマシンがリブートされる場合、または両方のデータ冗長ユニット (DRU) からミラーマシンのペアがリブートされる場合に発生します。

HADB は、このような「二重障害」の状況から自動的に自己回復できません。ペアーノード上に存在したデータの一部が失われるからです。このような場合、hadbm start コマンドは失敗し、hadbm status コマンドにより HADB が作動不能状態であることが示されます。

DRU と HADB の詳細な説明については、『管理ガイド』の「Administering the High Availability Database」、および『配備計画ガイド』を参照してください。

ヒント - HADB が異常な動作 (たとえば、定常的にタイムアウトする) を示している場合、再起動でその問題が解決するかどうかを調べるには、hadbm restart コマンドを使用します。

この方法で HADB が再起動するなら、HADB のサービスを引き続き利用できます。これとは異なり、HADB の起動と停止を hadbm stop および hadbm start を使用して別個の操作で行う場合は、HADB の停止中は HADB サービスを利用できません。

解決法

ノードの状態が **Starting** か **Recovering** を示すことを確認してから、データベースをリセットします。『管理ガイド』の「Administering the High Availability Database」の章にある「Recovering from Session Data Corruption」の手順に従います。

共有メモリーセグメントキーがすでに存在する (Windows のみ)

説明

hadbm プロセスから次のエラーが返されます。

```
HADB-S-05515: Shared memory segment with key "NNNNN" exists already
```

制御による停止後に HADB インスタンスを作成する際、以前に作成した、同じポートベースを使用するインスタンスを削除せずに行うと、このエラーが発生することがあります。この問題は、何らかの理由で HADB インスタンスの削除が失敗したことが原因で生じる場合もあります。

解決法

停止中のすべての hadb インスタンスを削除し、HADB リソースを再利用する前にすべての HADB リソースが解放されていることを確かめます。

問題がなくなるなら、\$TMP/f * の HADB ファイルを削除することによって、HADB の共有メモリーセグメントを手動で削除します。

configure-ha-cluster の失敗

説明

いくつかのホスト名から成る HADB ドメインの作成が成功したので、listdomain コマンドでそれを確認します。

```
$hadbm listdomain -w admin
Hostname Enabled? Running? Release Interfaces
host1 Yes      Yes      V4-4-1-3 128.139.113.110
host2 Yes      Yes      V4-4-1-3 128.139.113.111
```

次に、`hadbm create` コマンドでデータベースを作成し、`--hosts` オプションのパラメータとして完全ドメイン名を含む妥当なホスト名を指定します。

```
$ hadbm create --hosts=host1.xyz.abc.com,host2.xyz.abc.com...
```

このとき、次のエラーが返されます。

```
hadbm:Error 22176: The host host1.xyz.abc.com is not registered in the
HADB management domain. Use hadbm createdomain to set up the management
domain or hadbm extenddomain to include new hosts in an existing domain.
```

解決法 1

`listdomain` で表示される名前を使用します。次に例を示します。

```
hadbm create --hosts=host1,host2...
```

解決法 2

10 進の IP アドレス (DDN) を使用します。次に例を示します。

```
hadbm create --hosts=128.139.113.110,128.139.113.111
```

configure-ha-cluster を実行できない

説明

2 つの異なる HADB インストールが設定されています。1 つはサーバーホスト上に、もう 1 つは `hadbm` クライアントホスト上にあり、それぞれ異なるバージョンの HADB を実行しています。管理エージェントが一方の HADB バージョンで起動され、その後、`hadbm create` が他方のバージョンで実行されます。次のエラーが返されます。

```
HADBGMGT007:hadbm create command failed. Return value: 1 STDOUT:
STDERR: hadbm:Error 22170: The software package V4.4.x could not be found at path <packagepath>/4.4-x on
host <hostname>.
CLI137 Command configure-ha-cluster failed.
```

解決法

管理エージェントとすべての `hadbm` クライアントで、同じバージョンの HADB を使用します。

hadbm set コマンドが失敗する

説明

hadbm set コマンドを実行すると、データベースインスタンスが回復困難な状態に陥ります。

hadbm set コマンドでデータベース設定変数を変更しようとするとう失敗します。たとえば、DataBufferPoolSize の大きいサイズへの変更は、node-0 の共有メモリーの不足が原因で失敗します。この hadbm set コマンドにより、データベースに、停止状態の node-0 と稼動状態の node-1 が設定されます。hadbm set を使用してプールサイズを元の値にリセットしようとするとう、次のメッセージが表示されて失敗します。

```
22073: The operation requires restart of node 1. Its mirror node is
currently not available. Use hadbm status --nodes to see the status of
the nodes.
```

この状況では、hadbm startnode 0 コマンドも使えません。

解決法

データベースを停止し、hadbm set を使用して前の値を復元してから、データベースを再起動します。

configure-ha-cluster の失敗: **HADB** インスタンスの作成が失敗する

説明

HADB クラスタの作成が、次のメッセージで失敗します。

```
cresqldict: HADB-E-00208: The transaction was aborted.
```

これは、SQL 辞書テーブルを生成するブートランザクションが異常終了したことを示します。

解決法

configure-ha-cluster をもう一度実行します。hadbm create コマンドが前記のメッセージで失敗する場合は、コマンドを再実行します。

セキュリティの問題

この章では、セキュリティ設定の結果として発生する可能性のある、次の問題について説明しています。

- 77 ページの「`java.security.AccessControlException`: アクセス拒否エラー」
- 78 ページの「`javax.ejb.AccessLocalException`: 承認されていないクライアントエラー」
- 79 ページの「Solaris レルムで認証が機能していない」
- 79 ページの「アプリケーションクライアントで相互認証が機能していない」

`java.security.AccessControlException`: アクセス拒否エラー

説明

次のエラーが、アプリケーションクライアントから、または `server.log` 内で発生します。

```
java.security.AccessControlException: access denied
(java.util.PropertyPermission name write...
```

ポリシーファイルにアクセス権の問題があります。アプリケーションクライアント用の `client.policy` またはサーバー側コンポーネント用の `server.policy` のいずれかに、プロパティを設定するためのアクセス権がありません。

解決法

プロパティーを設定する必要があるアプリケーション用に、`client.policy`(アプリケーションクライアントの場合) または `server.policy`(EJB/Web モジュールの場合) でアクセス権を追加します。デフォルトでは、アプリケーションにはプロパティーの「読み取り」権しかありません。

たとえば、コードベースディレクトリ内のすべてのファイルに対する読み取り/書き込み権を付与するには、`client.policy` または `server.policy` に次の内容を追加または付加します。

```
grant codeBase "file:/.../build/sparc_SunOS/sec/-" {  
    permission java.util.PropertyPermission "*", "read,write";  
};
```

javax.ejb.AccessLocalException:承認されていないクライアントエラー

説明

Sun 固有の XML (たとえば、`sun-ejb-jar.xml`) でロールマッピング情報を利用でき、認証も成功しますが、次のエラーメッセージが表示されます。

```
[...INFO|sun-appserver-pe8.0|javax.enterprise.system.container.ejb|...|  
javax.ejb.AccessLocalException: Client not authorized for this invocation.  
at com.sun.ejb.containers.BaseContainer.preInvoke(BaseContainer.java:...  
at com.sun.ejb.containers.EJBObjectInvocationHandler.invoke(...)
```

解決法

EJB モジュール (`.jar`) または Web モジュール (`.war`) がアプリケーションにパッケージ化 (`.ear`) され、アプリケーションレベルで Sun 固有の `sun-application.xml` にロールマッピング情報を持っているかどうかチェックします。どのアプリケーション (`.ear`) でも、セキュリティロールマッピング情報が `sun-application.xml` に指定されている必要があります。モジュールレベル XML とアプリケーションレベル XML の両方が設定されていてもかまいません。

Solaris レルムで認証が機能していない

インストールおよびサーバー起動が root ユーザーとしてではなく、ローカルユーザーとして実行されたかどうかチェックします。Solaris レルムは root ユーザーでのみ機能するため、Application Server は必ず root ユーザーとして起動してください。ほかのローカルユーザーでは、機能するように設計されていません。ローカルユーザーではロールマッピングが発生する可能性がある、ということにも注意してください。

アプリケーションクライアントで相互認証が機能していない

説明

この障害は、KeyStore と TrustStore のプロパティが正しく設定されていない場合に発生する可能性があります。

解決法

JVM に次のプロパティを設定します。

```
javax.net.ssl.keyStore=  
<keystore-file-path\>;javax.net.ssl.trustStore=<truststore-file-path\>
```

アプリケーションクライアントを使用するには、環境変数 VMARGS を次の値に設定します。

```
-Djavax.net.ssl.keyStore=  
${admin.domain.dir}/${admin.domain}/config/keystore.jks  
-Djavax.net.ssl.trustStore=  
${admin.domain.dir}/${admin.domain}/config/cacerts.jks
```


よくある質問

ここでは、Application Server に関してよく質問される事項について説明します。

- 81 ページの「サーバー側レルムを1つも設定していないとどうなりますか?」
- 82 ページの「PKCS12 証明書をクライアント証明書として使用できますか?」
- 82 ページの「SSL クライアントの TLS/SSL ハンドシェイク情報を見ることはできますか?」
- 82 ページの「キーストアのパスワードを変更できますか?」
- 82 ページの「JAX-RPC 内のセッションをどのように維持できますか?」
- 83 ページの「スタンドアロン Java クライアントからはどのようにネームサービスにアクセスできますか?」
- 85 ページの「リモート EJB にアクセスするために RMI-IIOP スタブは必要ですか?」
- 85 ページの「アプリケーションロガーのログレベルを変更する方法を教えてください。」

サーバー側レルムを1つも設定していないとどうなりますか?

アプリケーションが XML ファイル内で設定されているのに、サーバー側レルムが1つも設定されていない場合は、アプリケーションがデフォルトレルムで認証されます。「指定したレルムはありません。」というエラーはスローされません。

PKCS12 証明書をクライアント証明書として使用できますか？

PKCS12 証明書を、相互認証時の SSL アプリケーションクライアントまたはスタンドアロンクライアントで使用方法はありますか？

いいえ。PKCS12 証明書を直接使用することはできません。ただし、JSSE を使用して独自のクライアントを記述することができ、その場合、`storetype=PKCS12` (キーストアの読み取り専用、書き込みなし) がサポートされます。

SSL クライアントの TLS/SSL ハンドシェイク情報を見ることはできますか？

はい。JVM 上で Java デバッグプロパティを設定してください。アプリケーションクライアントからハンドシェイク情報を見るには、次の設定を付加します。

```
-Djavax.net.debug=ssl,handshake to the VMARGS variable.
```

キーストアのパスワードを変更できますか？

はい。次の J2SE プロパティを使用して、キーストアのパスワードを変更できます。

```
-Djavax.net.ssl.keyStorePassword=password  
-Djavax.net.ssl.trustStorePassword=password
```

キーに対する操作を実行するには、キーストアのパスワードが個々のキーのパスワードと一致する必要があることに注意してください。そのため、前述のプロパティを使用してキーストアのパスワードを変更した後、そのパスワードと一致するように各キーへのパスワードを変更する必要があります。

JAX-RPC 内のセッションをどのように維持できますか？

JAX-RPC エンドポイントを持つセッションをクライアントで維持することはできません。セッションにはクライアント/サーバーの側面があるため、これを設定する明確な方法はありません。

考えられるのは、クライアントがサービスの呼び出しを行い、サーバーが応答して接続上に Cookie を設定するという方法です。それ以降、クライアントは呼び出しのたびにその同じ Cookie を送り返し、サーバーでそれをチェックすることが可能になります。

Cookie が送り返されても、通常、JAX-RPC スタブには影響を与えません。
SESSION_MAINTAIN_PROPERTY を true に設定すると、サーバーでスタブに設定したすべての Cookie が送り返されます。

サーバー側では、使用するクラスに 1 つのフィールドを追加し、そのフィールドを設定するメソッドを追加する必要があります。エンドポイントでは javax.xml.rpc.server.ServiceLifecycle を実装し、さらに 2 つのメソッド destroy() (空でも可) および init(Object context) を追加する必要があります。

エンドポイントに ServletEndpointContext オブジェクト、たとえば、myServletEndpointContext を追加します。init(Object context) メソッドは次のように設定できます。

```
myServletEndpointContext = (ServletEndpointContext) context;
```

これで、ビジネスメソッドから myServletEndpointContext.getHttpSession() を使用して HttpSession にアクセスできます。getHttpSession の最初の呼び出しで、セッションが存在していない場合はセッションが作成されます。

このモデルでは、クライアントから呼び出す任意のメソッドで、セッションの取得、セッション属性の設定、セッションからの値の取得などを行うことができます。この時点以降、クライアントから同じ Cookie 情報が送り返されます。

HttpSession オブジェクトの詳細については、<http://java.sun.com/j2ee/1.4/docs/api/javax/servlet/http/HttpSession.html> を参照してください。

スタンドアロン **Java** クライアントからはどのようにネームサービスにアクセスできますか？

▼ アプリケーションクライアントからネームサービスにアクセスする

- 1 クライアント **Java VM** 起動時の CLASSPATH に appserv-rt.jar を追加します。
JNDI ブートストラップ機構で、appserv-rt.jar にある jndi.properties という名前のファイルが検索されます。このファイルには、Application Server のネームサービスに関するすべてのブートストラッププロパティが含まれています。これらのプロパティは、クライアント起動スクリプトまたはアプリケーションコードに明示的に記述するよりも、appserv-rt.jar から読み取るようにすることをお勧めします。
- 2 スタンドアロンクライアントからリモート **EJB** にアクセスする場合は、配備からクライアント **JAR** を取り出したり、**JAR** をクライアント **JVM** の CLASSPATH に追加したりす

る必要はありません。**Application Server** ネームサービスの使用時に静的 **RMI-IIOP** スタブは必要ないからです。これにより、以前のリリースで必要とされた手順は不要になります。詳しくは、[85 ページの「リモート EJB にアクセスするために RMI-IIOP スタブは必要ですか？」](#)を参照してください。

- 3 引数を必要としないデフォルトコンストラクタ `InitialContext` を使用するようにクライアントコードを記述します。次に例を示します。

```
InitialContext ic = new InitialContext();
```

クライアントから `CosNaming` サービスを明示的に参照するコーディングが強く推奨されるというのは、よくある誤解です。`CosNaming` は **Application Server** オブジェクトの一部の種類にのみ使用されるため、そのように記述しても、**JMS** キュー、接続ファクトリなどクライアントで必要となる可能性があるリソースの多くにアクセスできるようにはなりません。加えて、`CosNaming` を明示的に使用すると、**Application Server** のネームサービスコードの実行が省略されます。これにより、多くの場合、**Application Server** のネームサービスに組み込まれた便利で付加価値の高い処理をクライアントから利用できなくなります。

- 4 検索時には、ターゲットリソースのグローバル **JNDI** 名を使用します。`java:comp/env` をスタンドアロン **Java** クライアントから使用することはできません。そのようなクライアントは **J2EE** コンテナの外部で実行するように定義されているからです。

`java:comp/env` を使用できるクライアントコンポーネントは、**J2EE** アプリケーションクライアントだけです。

- 5 クライアントがサーバーインスタンスとは別のホストマシンで実行されている場合は、**Java VM** 起動時に次のシステムプロパティを設定します。

```
-Dorg.omg.CORBA.ORBInitialHost=hostname_of_target_server
```

デフォルト値は `localhost` です。したがって、このプロパティが必要なのは、クライアントとサーバーのインスタンスが共存していない場合だけです。次に例を示します。

```
java -Dorg.omg.CORBA.ORBInitialHost=server1 ... com.foo.MyMainClass
```

- 6 デフォルトでは、クライアントからサーバーのネームサービスを使用するためポート 3700 への接続が試行されます。3700 は **Application Server** で使用されるデフォルトのネームサービスポートなので、クライアントで追加のポート設定は必要ありません。ポートの衝突が原因で、サーバーインスタンスで別のネームサービスポートが使用される場合もあります。サーバーインスタンスで使用されるネームサービスポートは、`domain.xml` 内の `<iiop-listener id="orb-listener-1" port="3700">` 要素に示されます。

クライアントで使用されるネームサービスポートを変更するには、クライアント **Java VM** 起動時に次のシステムプロパティを設定します。

```
-Dorg.omg.CORBA.ORBInitialPort=naming_port_of_target_server
```

リモート EJB にアクセスするために RMI-IIOP スタブは必要ですか？

いいえ。以前のリリースの Application Server とは異なり、現在のバージョンは実行時に静的 RMI-IIOP スタブを必要としません。

この要件がなくなったことで、次のような利点が生じます。

- リモート EJB またはリモート EJB のクライアントを含むアプリケーションの配備および再配備時間の短縮
- スタブ CLASSPATH 設定の問題が原因で発生する実行時エラーの減少

加えて Application Server では、RMI-IIOP 相互運用性を完全に維持しながら、実行時のパフォーマンスに大きな影響を与えることなくこれらの利点の実現します。

現在でも RMI-IIOP スタブが必要とされるシナリオは、スタンドアロンクライアントで CosNaming ネームサービス用に InitialContext を明示的にインスタンス化する場合だけです。これは、Application Server のネームサービスを使用するための推奨される手法ではありません。ただし、この種のスタンドアロンクライアントとの互換性を維持するため、RMI-IIOP スタブを以前のリリースと一致する方法で強制的に生成する配備時オプションが用意されています。これを使用するには、次のように設定します。

```
--generatemitubs=true
```

この設定は、asadmin または管理コンソールを使用して配備する際に行えます。RMI-IIOP スタブは、以前のリリースの場合と同様に client.jar ファイルに格納されます。

アプリケーションログのログレベルを変更する方法を教えてください。

アプリケーションでは、それぞれ固有のアプリケーションログを使用してメッセージを記録しています。特定のアプリケーションのログレベルを設定するには、次の2つの方法のいずれかを使用します。

- 管理 GUI のログレベル設定ページで、ログ名を表すプロパティ `name`、および `FINEST`、`FINER`、`FINE`、`CONFIG`、`INFO`、`WARNING`、`SEVERE` の7つのログレベルのいずれかまたは `OFF` を表すプロパティ `value` から成るプロパティを追加します。
たとえば、`com.X.Y` という名前のアプリケーションログのログレベルを `FINEST` に変更するには、プロパティ `name` を `com.X.Y` に、プロパティ `value` を `FINEST` に設定します。変更は `domain.xml` ファイルに反映され、ただちに有効になります。サーバーの再起動は必要ありません。

- 次に示すように、domain.xml 内の <module-log-levels> 要素にプロパティを直接追加します。

```
<module-log-levels admin="INFO" classloader="INFO" cmp="INFO"
cmp-container="INFO" configuration="INFO" connector="INFO"
corba="INFO" deployment="INFO" ejb-container="INFO" javamail="INFO"
jaxr="INFO" jaxrpc="INFO" jdo="INFO" jms="INFO" jta="INFO"
jts="INFO" mdb-container="INFO" naming="INFO" node-agent="INFO"
resource-adapter="INFO" root="INFO" saaj="INFO" security="INFO"
server="INFO" synchronization="INFO" util="INFO" verifier="INFO"
web-container="INFO">
  <property name="com.X.Y" value="FINEST" /\>
</module-log-levels>
```

索引

A

ACC ログサービス, 21
.asadmintruststore, 26-27

H

HADB, 49
 環境の設定, 66
hadbm コマンドが機能しない, 66
HADB が起動しない, 72-73
HTTP サーバーアクセスログ, 32

J

JAVA_HOME 設定, 67

P

pkginfo, 16

R

response-timeout-in-seconds, 55
rpm, 16

S

server.log, 21

showrev, 16

start-domain コマンドが失敗する, 27-28
SunSolve, 23

W

Windows
 サーバーが起動しない, 41
 ログレベル, 21

あ

アプリケーション, 使用できない, 38-41

い

異常な方法での停止, 72-73

え

エラーメッセージ, 19

お

オンラインフォーラム, 23

か

管理コンソール, 表示できない, 37-38

管理コンソールを表示できない, 37-38

管理パスワード

無効, 39

わからない, 40-41

管理ユーザー名

無効, 39

わからない, 40-41

管理ユーザー名またはパスワードがわからない, 40-41

こ

高負荷の状況, 61

さ

サーバー

Windows 上で起動しない, 41

デフォルトポート番号が変更される場合, 35

ログファイル, 説明, 32

ログファイル, 場所, 33

サーバーアプリケーションを使用できない, 38-41

サーバーの再起動, 28-32

す

スレッドダンプ、取得, 21

て

停電と HADB, 72-73

と

ドメイン

起動, 28-32

検索, 37

な

ナレッジベース, 23

の

ノードエージェント、起動, 28-32

ふ

フォーラム, 23

ほ

ポート番号, デフォルト値が変更される場合, 35

ホスト名, 検索, 37

ま

マニュアル, 23

む

無効な管理ユーザー名またはパスワード, 39

め

メッセージタイプ, 32

も

問題の診断, 16

問題の調査, 18

り

リモートサーバー, アクセスが失敗する, 35-37

リモートサーバーにアクセスできない, 35-37

ろ

ローカルサーバー, アクセスが失敗する, 34-35

ローカルサーバーにアクセスできない, 34-35

ログ

ディレクトリ, 33

見つからない, 32-34

ログファイル, 20

ログファイルが見つからない, 32-34

ログレベル, 20

