



Sun Java™ System

Application Server 7

インストールガイド

2004Q2 Update 1

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

Part No: 819-1000

Copyright © 2004 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

Sun Microsystems, Inc. は、この製品に含まれるテクノロジーに関する知的所有権を保持しています。特に限定されることなく、これらの知的所有権は <http://www.sun.com/patents> に記載されている 1 つ以上の米国特許および米国およびその他の国における 1 つ以上の追加特許または特許出願中のものが含まれている場合があります。

このソフトウェアは SUN MICROSYSTEMS, INC. の機密情報と企業秘密を含んでいます。SUN MICROSYSTEMS, INC. の書面による許諾を受けることなく、このソフトウェアを使用、開示、複製することは禁じられています。

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.

Use is subject to license terms. この配布には、第三者が開発したソフトウェアが含まれている可能性があります。

Sun、Sun Microsystems、Sun のロゴマーク、Java、Solaris、Sun™ ONE、Sun™ ONE Studio、iPlanet、J2EE、J2SE、Enterprise JavaBeans、EJB、JavaServer Pages、JSP、JDBC、JDK、JVM、Java Naming and Directory Interface、JavaMail および Java Coffee Cup のロゴは、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。

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。

UNIX は、X/Open Company, Ltd が独占的にライセンスしている米国およびその他の国における登録商標です。

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であり、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らないものとします。

目次

本書について	7
対象読者	7
マニュアルの用法	8
マニュアルの構成	10
マニュアルの表記規則	11
一般的な表記規則	11
ディレクトリ名の表記規則	12
Sun の連絡先	13
ご意見、ご要望	13
トレーニング	13
製品サポートの問い合わせ	13
マニュアルの参照方法	14
 第 1 章 Standard Edition および Enterprise Edition ソフトウェアのインストール	15
Application Server のインストールについて	16
製品の配布	16
モデルのパッケージ化とディレクトリ構造	17
Solaris SPARC/x86 でのパッケージベースのインストールおよび Linux での RPM ベース のインストール	17
Microsoft Windows および Solaris SPARC/x86 でのファイルベースのインストール	18
インストールコンポーネント	18
インストール方法	21
Application Server ソフトウェアのインストール	23
Web サイトからのダウンロード	23
Solaris SPARC および x86 または Linux 用インストールファイルのダウンロード	23
Microsoft Windows 用のダウンロード	24
Standard Edition のインストール	24

Standard Edition の Solaris SPARC および x86 へのインストール	24
Standard Edition の Linux へのインストール	26
Microsoft Windows の Standard Edition へのインストール	28
Enterprise Edition のインストール	30
Enterprise Edition の Solaris SPARC および x86、Linux へのインストール	30
Microsoft Windows での Enterprise Edition のインストール	34
ロードバランサプラグインコンポーネントのインストール	35
サイレントモードでのインストール	37
インストール設定ファイルの作成	37
インストール設定ファイルの作成の構文	37
インストール設定ファイルの例	37
インストール設定ファイルの変更	38
サイレントモードでのインストール	41
第 2 章 HADB の設定準備	43
Windows での HADB の設定	44
共有メモリとセマフォの設定	45
Solaris での共有メモリの設定	46
Linux での共有メモリの設定	47
ホスト通信の設定	48
HADB 管理のための RSH の設定	49
HADB 管理のための SSH の設定	50
SSH の要件と制限事項	50
Solaris 8 への SSH のインストール	51
SSH の設定	52
Solaris で VERITAS ファイルシステムを使う場合の HADB の使用法	56
Linux で ext ファイルシステムを使う場合の HADB の使用方法	56
ユーザー環境の設定	57
root 以外の特権に対する管理の設定	58
clsetup の使用	59
clsetup の機能	60
入力ファイルの機能	60
clsetup が実行する内容	61
clsetup が使用するコマンド	61
clsetup の要件と制限事項	62
clsetup 入力ファイルの編集	64
clinstance.conf ファイル	64
clpassword.conf ファイル	65
clresource.conf ファイル	66
clsetup の実行	74
clsetup のクリーンアップ手順	76
時刻の同期	77

第3章 Standard Edition および Enterprise Edition ソフトウェアのアンインストール	79
アンインストールについて	79
アンインストールの要件	80
Application Server ソフトウェアのアンインストール	80
サイレントモードでのアンインストール (非対話型)	82
 付録 A Application Server のアップグレード	83
Standard Edition または Enterprise Edition のアップグレード	84
 索引	85

本書について

このマニュアルでは、Sun Java System Application Server 7 Standard Edition および Enterprise Edition のインストール方法について説明します。

この章では次の項目について説明します。

- [対象読者](#)
- [マニュアルの使用法](#)
- [マニュアルの構成](#)
- [マニュアルの表記規則](#)
- [Sun の連絡先](#)
- [マニュアルの参照方法](#)

対象読者

このマニュアルは、企業内で Bean の開発、アセンブル、および配備を行なっている読者を対象としています。

このマニュアルでは、次の項目に精通していることを前提としています。

- Java プログラミング
- Java™ Servlet、JavaServer Pages™ (JSP™)、Enterprise JavaBeans™ (EJB™)、および Java™ Database Connectivity (JDBC™) 仕様で定義されている Java API
- 構造化照会言語 (Structured Query Language、SQL)
- リレーショナルデータベースの概念
- デバッグ、ソースコード管理を含むソフトウェア開発プロセス

マニュアルの使用方法

Sun Java System Application Server Standard Edition および Enterprise Edition のマニュアルは、PDF 形式および HTML 形式のオンラインファイルとして利用できます。

次の表は、Sun Java System Application Server のマニュアルに記述されているタスクと概念を示しています。(AS 7 2004Q2 用に更新)と記載されているマニュアルは、Sun Java System Application Server Standard Edition および Enterprise Edition 7 2004Q2 用に更新されたものです。このように記載されていないマニュアルは、バージョン 7 Enterprise Edition のリリース以降は更新されていないものです。

表 1 Sun Java System Application Server ドキュメントロードマップ

情報の内容	参照するマニュアル
(AS 7 2004Q2 用に更新) ソフトウェアおよびマニュアルの最新情報。サポート対象のハードウェア、オペレーティングシステム、JDK、JDBC、RDBMS の一覧も含まれる	『リリースノート』
サーバーのアーキテクチャの図と説明、Sun Java System Application Server アーキテクチャの利点	『サーバーアーキテクチャの概要』
企業、開発者、および運用向けの、Sun Java System Application Server 7 の新機能	『新機能』
Sun Java System Application Server 製品の基本的な使用方法。サンプルアプリケーションのチュートリアルも含まれる	『入門ガイド』
(AS 7 2004Q2 用に更新) Sun Java System Application Server Standard Edition および Enterprise Edition ソフトウェアとそのコンポーネント (サンプルアプリケーション、管理インタフェースなど) のインストール。Enterprise Edition ソフトウェアについては、高可用性設定の実装手順が記載されている	『インストールガイド』
(AS 7 2004Q2 用に更新) Sun Java System Application Server を確実にサイトに適合させるように配備するための、システム要件とエンタープライズの評価。アプリケーションサーバーを配備する際に注意する必要がある、一般的な問題と懸案事項も記載されている	『System Deployment Guide』
J2EE コンポーネント (サブレット、EJB™ (Enterprise JavaBeans™)、JSP™ (JavaServer Pages™) など) 向け Java オープンスタンダードモデルに準拠した Sun Java System Application Server 上で実行することを目的とした Java™ Platform, Enterprise Edition (J2EE™ プラットフォーム) アプリケーションの作成および実装。アプリケーション設計、開発ツール、セキュリティ、構成、配備、デバッグ、ライフサイクルモジュールの作成方法などについての一般的な情報が含まれる。Sun Java System Application Server のさまざまな用語について解説する用語集も含まれる	『Developer's Guide』

表 1 Sun Java System Application Server ドキュメントロードマップ (続き)

情報の内容	参照するマニュアル
Sun Java System Application Server の Java™ Servlet および JSP (JavaServer Pages) 仕様に準拠した J2EE Web アプリケーションの作成および実装。Web アプリケーションプログラミングの概念とタスクの説明、サンプルコード、実装のヒント、関連資料の紹介など。結果キャッシュ機能、JSP のプリコンパイル、セッション管理、セキュリティ、配備、SHTML、CGI などが含まれる	『Developer's Guide to Web Application』
(AS 7 2004Q2 用に更新) Sun Java System Application Server のエンタープライズ Bean 向け Java オープンスタンダードに準拠した J2EE アプリケーションの作成および実装。Enterprise JavaBeans (EJB) プログラミングの概念とタスクの説明、サンプルコード、実装のヒント、関連資料の紹介など。コンテナ管理持続性、読み取り専用 Bean、エンタープライズ Bean に関連付けられた XML ファイルや DTD ファイルなどが含まれる	『Developer's Guide to Enterprise javaBeans Technology』
(AS 7 2004Q2 用に更新) Sun Java System Application Server 上で J2EE アプリケーションにアクセスする Application Client コンテナ (ACC) の作成	『Developer's Guide to Clients』
Sun Java System Application Server 環境での Web サービスの作成	『Developer's Guide to Web Services』
Java™ Database Connectivity (JDBC™)、トランザクション、Java Naming and Directory Interface™ (JNDI)、Java™ Message Service (JMS)、および JavaMail™ API	『Developer's Guide to J2EE Services and APIs』
カスタム NSAPI プラグインの作成	『NSAPI Developer's Guide』
(AS 7 2004Q2 用に更新) 管理インタフェースとコマンド行インタフェースの両方からの Sun Java System Application Server サブシステムとコンポーネントの設定、管理、および配備の説明と手順。クラスタ管理、高可用性データベース、ロードバランス、およびセッションの持続性について説明する。Sun Java System Application Server のさまざまな用語について解説する用語集も含まれる	『Administration Guide』
server.xml ファイルなどの Sun Java System Application Server の設定ファイルの編集	『管理者用設定ファイルリファレンス』
Sun Java System Application Server 運用環境のセキュリティの設定および管理。一般的なセキュリティ、証明書、および SSL/TLS 暗号化に関する情報など。HTTP サーバーベースのセキュリティについても説明	『セキュリティ管理者ガイド』
Sun Java System Application Server 7 用の J2EE™ CA (Connector Architecture) コネクタのサービスプロバイダ実装の設定と管理。管理ツール、プーリングモニター、JCA コネクタの配備、サンプルコネクタとサンプルアプリケーションなどについて説明する	『J2EE CA Service Provider Implementation Administrator's Guide』
(AS 7 2004Q2 用に更新) 新しい Sun Java System Application Server プログラミングモデルへのアプリケーションの移行 (特に、iPlanet Application Server 6.x、Sun ONE Application Server 7.0 からの移行)。移行例も含まれる	『Migrating and Redeploying Server Applications Guide』

表 1 Sun Java System Application Server ドキュメントロードマップ (続き)

情報の内容	参照するマニュアル
(AS 7 2004Q2 用に更新) Sun Java System Application Server のパフォーマンスを改善する方法とその理由	『Performance Tuning Guide』
(AS 7 2004Q2 用に更新) Sun Java System Application Server に関する問題の解決	『Troubleshooting Guide』
(AS 7 2004Q2 用に更新) Sun Java System Application Server に関するエラーメッセージの解決	『Error Message Reference』
(AS 7 2004Q2 用に更新) マニュアルページに記載されている、Sun Java System Application Server で利用できるユーティリティコマンド	『Utility Reference Manual』
Sun Java System Message Queue 3.5 SP1 ソフトウェアの使用	Sun Java System Message Queue については次の URL を参照： http://docs.sun.com/db/prod/s1.s1msgqu?l=ja#hic

マニュアルの構成

このマニュアルでは次の項目について説明します。

- 第 1 章「[Standard Edition](#) および [Enterprise Edition](#) ソフトウェアのインストール」- Sun Java System Application Server 7 ソフトウェアコンポーネントのインストール手順について説明します。非対話型のサイレントインストールを実行する手順も含まれています。
- 第 2 章「[HADB](#) の設定準備」- 共有メモリの設定手順、および高可用性設定のためのホスト通信とユーザー環境の設定手順について説明します。
- 第 3 章「[Standard Edition](#) および [Enterprise Edition](#) ソフトウェアのアンインストール」- Sun Java System Application Server 7 ソフトウェアのアンインストール手順について説明します。非対話型のサイレントアンインストールを実行する手順も含まれています。
- 付録 A「Application Server のアップグレード」- 以前にインストールした Application Server を現在のバージョンにアップグレードする方法について説明します。

マニュアルの表記規則

この節では、このマニュアルの表記規則について説明します。

- 一般的な表記規則
- ディレクトリ名の表記規則

一般的な表記規則

このマニュアルは、次の表記規則に従っています。

- **ファイルとディレクトリのパス**は、UNIX の形式で表記します (ディレクトリ名を「¥」記号で区切って表記)。Windows の場合はディレクトリのパスは同じです。ただし、ディレクトリは「¥」記号で区切られます。

- **URL** は次の書式で記述します。

`http://server.domain/path/file.html`

server はアプリケーションを実行するサーバー名、*domain* はユーザーのインターネットドメイン名、*path* はサーバー上のディレクトリの構造、*file* は個別のファイル名を示します。URL の斜体文字の部分は可変部分です。

- **フォント**は、次のように使い分けます。
 - モノスペースフォントは、コード例、コードリスト、API および言語要素 (関数名、クラス名など)、ファイル名、パス名、ディレクトリ名、および HTML タグに使用します。
 - 斜体文字はコード変数に使用します。
 - 斜体文字は、変数および可変部分、およびリテラルに使われる文字にも使用します。
 - **太字**は、段落の先頭またはリテラルに使われる文字の強調に使用します。
- このマニュアルでは、**インストールルートディレクトリ**を *install_dir* と記述します。例外は、12 ページの「[ディレクトリ名の表記規則](#)」で説明されています。

デフォルトでは、*install_dir* はほとんどのプラットフォームで次の場所にあります。

- Solaris および Linux ファイルベースのインストールの場合：

ユーザーのホームディレクトリ /sun/appserver7

- Windows、すべてのインストール：

システムドライブ : ¥Sun¥AppServer7

上記のプラットフォームの場合、*default_config_dir* および *install_config_dir* は *install_dir* と同じです。例外と追加情報については、[12 ページの「ディレクトリ名の表記規則」](#)を参照してください。

- このマニュアルでは、**インスタンスルートディレクトリ**は、*instance_dir* と記述します。これは次のパスの省略形式です。

default_config_dir/domains/*domain*/*instance*

- このマニュアルでは、**UNIX 固有の記述**は、特に言及されていない場合 Linux オペレーティングシステムにも適用されます。

ディレクトリ名の表記規則

Solaris のパッケージに含まれる製品または Linux RPM ベースの配布では、アプリケーションサーバーのファイルはデフォルトで複数のルートディレクトリにまたがって保存されます。このマニュアルは、デフォルトの複数のインストールディレクトリに対応した、次の表記規則を使用しています。

- *install_dir* は /opt/SUNWappserver7 を示します。このディレクトリにはインストールイメージの静的な要素が保存されます。アプリケーションサーバーを構成するすべてのユーティリティ、実行可能ファイル、ライブラリが格納されます。
- *default_config_dir* は /var/opt/SUNWappserver7/domains を示します。このディレクトリは、作成したドメインのデフォルトの保存場所です。
- *install_config_dir* は /etc/opt/SUNWappserver7/config を示します。このディレクトリには、ライセンスなどのインストール全体に適用される設定情報や、このインストール用に設定した管理ドメインのマスターストが保存されます。

Sun の連絡先

次のような場合、Sun Microsystems にご連絡ください。

- ご意見、ご要望
- トレーニング
- 製品サポートの問い合わせ

ご意見、ご要望

製品またはマニュアル全般に対するフィードバックは、appserver-feedback@sun.com までご連絡ください。

トレーニング

Application Server のトレーニングコースは、次の URL から申し込むことができます。

http://training.sun.com/US/catalog/enterprise/web_application.html/

Sun Java System Application Server の新しいコースについては、定期的にこのサイトを参照して確認してください。

製品サポートの問い合わせ

ご使用のシステムに問題が発生した場合は、次のいずれかの方法でカスタマサポートにお問い合わせください。

- 次のオンラインサポート Web サイトをご利用ください。

<http://www.sun.com/supporttraining/>

- 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。

サポートのご依頼の前に、次の情報を用意してください。サポート担当がお客様の問題を解決するために必要な情報です。

- 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、問題の原因と思われるパッチやその他のソフトウェアなどの製品バージョンバージョンを調べるために一般に使用するコマンドは次のとおりです。
 - Solaris: pkginfo、showrev

- Linux: rpm
- その他すべて: `asadmin version --verbose`
- 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
- 次のような設定ファイル:
 - `instance_dir/config/server.xml`
 - Web アプリケーションの `web.xml` ファイル。
Web アプリケーションが問題に関係している場合
- アプリケーションの場合、クラスタ内で実行したときに問題が発生するのか、スタンドアロンで実行したときに発生するのか

マニュアルの参照方法

Sun Java System Application Server のマニュアルは、様々な方法で入手できます。

- マニュアル - HTML 形式および印刷可能な PDF 形式の Sun Java System Application Server のマニュアルおよびリリースノートは、次のサイトからダウンロードできます。

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

- オンラインヘルプ: グラフィカルインタフェースの「ヘルプ」ボタンをクリックするとコンテキストヘルプウィンドウを起動できます。
- マニュアルページ - コマンド行でマニュアルページを表示するには、使用する MANPATH 環境変数に `install_dir/man` を最初に追加する必要があります。変数の設定後、コマンド行に `man command_name` と入力すると、Sun Java System Application Server コマンドのマニュアルページにアクセスできます。次に例を示します。`man asadmin`

Standard Edition および Enterprise Edition ソフトウェアのインストール

この章では、Sun Java System Application Server Standard Edition および Enterprise Edition のインストール方法について説明します。対話形式でインストールするか、サイレントモードを使用して複数のマシンに対して同じ条件でインストールすることもできます。

ここでは次の項目について説明します。

- [Application Server のインストールについて](#)
- [Application Server ソフトウェアのインストール](#)
- [ロードバランサプラグインコンポーネントのインストール](#)
- [サイレントモードでのインストール](#)

最新の更新情報については、『Sun Java System Application Server リリースノート』を参照してください。

インストール後に Application Server を設定する方法の詳細については、『Sun Java System Application Server Administration Guide』を参照してください。

次の Web ページには、製品のダウンロードやその他の役に立つ情報が用意されています。http://www.sun.com/software/products/appsrvr/home_appsrvr.html

Application Server のインストールについて

Sun Java System Application Server 製品は、互いに関連して機能するいくつかのソフトウェアコンポーネントで構成されています。インストールでは、さまざまな選択が可能です。

- 製品の **Standard Edition** または **Enterprise Edition** をインストールできます。
- 製品 CD またはダウンロードサイトからインストールできます。
- 製品のファイルベースまたはパッケージベースの配布をインストールできます。
- コマンド行インタフェースまたはグラフィカルインタフェースを使用してインストールできます。
- 対話形式またはサイレントモードでインストールできます。

注	ファイルベースの配布を使用している場合、同じマシンに複数の Application Server のインスタンスをインストールできます。
---	--

この節では次の項目について説明します。

- [製品の配布](#)
- [モデルのパッケージ化とディレクトリ構造](#)
- [インストールコンポーネント](#)
- [インストール方法](#)

製品の配布

Sun Java System Application Server には、2 つのタイプの配布があります。

- **ファイルベースの配布** (Solaris SPARC、x86、Linux、Microsoft Windows) - root 以外のユーザーで複数のインストールを行うことができます。
- **RPM ベース (Linux) またはパッケージベースの配布** (Solaris SPARC および x86) - root 権限でインストールする必要があります。

製品のこれらの配布は、製品 CD または Web サイトからのダウンロードによりインストールできます。http://www.sun.com/software/download/app_servers.html で、Sun Java System Application Server 製品をダウンロードできます。

注	複数のファイルベース Application Server を 1 台のマシンにインストールできます。
---	---

次の表に、それぞれのプラットフォームで使用可能な配布タイプと、それぞれの配布で使用可能な Application Server Edition を示しています。

表 1-1 製品配布の定義

配布タイプ	プラットフォーム	Application Server Edition
ファイルベース	Solaris SPARC、	Standard および Enterprise
	Solaris x86	Standard および Enterprise
	Linux	Standard および Enterprise
	Microsoft Windows	Standard および Enterprise
パッケージベース	Solaris SPARC、	Standard および Enterprise
	Solaris x86	Standard および Enterprise
RPM ベース	Linux	Standard および Enterprise

サポートされているプラットフォームのバージョンについては、『Sun Java System Application Server 7 リリースノート』を参照してください。

モデルのパッケージ化とディレクトリ構造

パッケージベースおよびファイルベースの配布には、個別のインストーラがあります。システムにインストールされた製品の配布に応じて、Sun Java System Application Server ソフトウェアが単一の root ディレクトリパスまたは複数の root ディレクトリにまたがってインストールされます。

この節では次の項目について説明します。

- [Solaris SPARC/x86 でのパッケージベースのインストールおよび Linux での RPM ベースのインストール](#)
- [Microsoft Windows および Solaris SPARC/x86 でのファイルベースのインストール](#)

Solaris SPARC/x86 でのパッケージベースのインストールおよび Linux での RPM ベースのインストール

Solaris SPARC/x86 でのパッケージベースの配布、および Linux での RPM ベースの配布では、コンポーネントはパッケージとしてインストールされます。デフォルトでは、インストール先は 3 つのディレクトリルートにまたがります。

- /opt/SUNWappserver7 には、インストールイメージの静的な要素が保存されます。Sun Java System Application Server ソフトウェアのユーティリティ、実行可能ファイル、およびライブラリはすべて、ここに保存されます。

- /etc/opt/SUNWappserver7/config には、ライセンスやそのインストール用に設定した管理ドメインのマスターストなど、インストール全体に適用される設定情報が保存されます。
- /var/opt/SUNWappserver7/domains は、デフォルトで管理ドメインが作成される領域です。

注 Solaris SPARC/x86 でのパッケージベースの配布、および Linux での RPM ベースの配布では、**root** としてログインする必要があります。

Microsoft Windows および Solaris SPARC/x86 でのファイルベースのインストール

Microsoft Windows (ファイルベースの配布だけで使用可能)、および Solaris SPARC/x86 でのファイルベースの配布では、コンポーネントは単一のディレクトリパスにインストールされます。デフォルトのディレクトリは次のとおりです。

- Microsoft Windows の場合 : `c:\¥Sun¥AppServer7`
- Solaris SPARC および x86 ファイルベースの配布の場合 :
`appserver_install_dir/sun/appserver7`

これらのケースでは、/config および /domains ディレクトリはインストールディレクトリルートの下に置かれます。

インストールコンポーネント

通常は、Sun Java System Application Server 製品の機能を提供する基本コンポーネントをインストールします。特定のコンポーネントを、インストールしないように選択することもできます。インストールしないことを選択したコンポーネントをあとから追加したい場合、依存関係が満たされていれば、そのコンポーネントの追加インストールができます。

注 部分インストールのあとに、追加インストールを何回でも実行できます。サイレントモードの場合、部分的な初期インストールを実行できますが、2 回目以降のインストールでは対話モードを使用する必要があります。

インストールプログラムは、各コンポーネントに対して指定されているコンポーネントの依存関係を適用します。いったんコンポーネントの依存関係が満たされると、コンポーネントのライフサイクルは独立します。特定のコンポーネントは、追加インストールによって、ほかのコンポーネントに影響を与えることなく動的にインストールまたはアンインストールすることができます。ただし、追加のアンインストールを行うことはできません。

Sun Java System Application Server **Standard Edition** および **Enterprise Edition** 製品には、次のインストールコンポーネントが含まれています。

- **Application Server - Sun Java System Application Server** のすべて。グラフィカル管理ツールおよびコマンド行管理ツール、`asadmin` コマンド、および Sun Java™ System Message Queue が含まれます。

UNIX パッケージベースまたは RPM ベースの配布、および Microsoft Windows の配布では、Application Server ソフトウェアにより Sun Java System Message Queue ソフトウェアが自動的にインストールされます。`install_dir/imq`

UNIX ファイルベースの配布では、Sun Java System Message Queue は `/usr/share/lib`、`usr/share/lib/imq` (lib ファイル) および `/usr/bin` (バイナリ) に配布されます。

同じシステム上に Application Server Enterprise Edition および HADB サーバーのノードをインストールする場合は、両方のコンポーネントを選択します。

- **Application Server 管理クライアント - asadmin ユーティリティ**

管理クライアントコンポーネントを選択して、Application Server がインストールされていないマシンにコマンド行ユーティリティを個別にインストールします。Application Server をインストールすると、管理クライアントもインストールされます。

- **Java 2 Software Development Kit** - インストール中に、Java 2 SDK のバージョンが正しい場合にかぎり、システムにすでにインストールされている Java 2 SDK を再度使用するように選択できます。それぞれの配布のデフォルトのインストール場所は、次のとおりです。

- Solaris SPARC および x86、Linux、および Microsoft Windows ファイルベースの配布の場合 : `install_dir/jdk`
- Solaris SPARC および x86 パッケージベースの配布の場合 : `/usr/j2se`
- Linux RPM ベースの配布の場合 : `/usr/java/j2sdk1.4.2_x`

サポートされている Java 2 SDK のバージョンについては、『Sun Java System Application Server 7 リリースノート』を参照してください。

注 Sun Java System Application Server 製品は、Sun Microsystems の Java 2 SDK で動作することが保証されています。サードパーティの JDK 開発キットはバージョンに関係なくサポート対象外です。

- **サンプルアプリケーション** - サンプルには、ソース、スキーマ、Ant ビルドスクリプト、EAR ファイルが付属しています。データベース関連のサンプルに関連付けられている既存のデータが、データベースで利用できます。これらのサンプルアプリケーションは、次のように分類できます。

- テクノロジサンプル - Java™ 2 Platform, Enterprise Edition (J2EE™) 仕様の技術的な側面を紹介し、Sun Java System Application Server プラットフォームの付加価値機能を示します。
- 相互運用サンプル - Sun Java System Application Server プラットフォーム上で各種テクノロジを相互運用する方法を詳しく示します。

デフォルトでは、サンプルアプリケーションは次の場所にインストールされます。
install_dir/samples

サンプルについての詳細情報は、次の場所にあります。
install_dir/samples/index.html

- **PointBase Server (Standard Edition に限る)** - デフォルトでは、PointBase は次の場所にインストールされます。*install_dir/pointbase*

サポートされている PointBase Server のバージョンについては、『Sun Java System Application Server 7 リリースノート』を参照してください。

Sun Java System Application Server **Enterprise Edition** 製品には、次の追加コンポーネントが含まれています。

- **高可用性データベースサーバー (Enterprise Edition)** - デフォルトでは、HADB サーバーは次の場所にインストールされます。*install_dir/SUNWhadb*
このコンポーネントの追加情報については、『Sun Java System Application Server Administration Guide』の「HADB Configuration」の章を参照してください。
- **高可用性管理クライアント (Enterprise Edition)** - *hadbm* コマンドに限る
これらのユーティリティの使用方法は、『Sun Java System Application Server Administration Guide』、*hadbm* のマニュアルページ、および *asadmin* セッション持続性マニュアルページに記載されています。
- **ロードバランサプラグイン (Enterprise Edition)** - このコンポーネントは、プリインストールされている Web サーバーに依存します。サポートされている Web サーバーは、『Sun Java System Application Server 7 リリースノート』の「プラットフォーム」セクションに記述されています。

このコンポーネントに関する追加情報は、このインストールガイドの「[ロードバランサプラグインコンポーネントのインストール](#)」の節、および『Sun Java System Application Server Administration Guide』の「Configuring Load Balancing」の章を参照してください。

インストール方法

3 つのインストール方法があります。

- **グラフィカル (対話型)** - 一連のグラフィカル画面が表示されます。これは、`./setup` にオプションを指定せずにインストールプログラムを起動した場合のデフォルトのインストール方法です。
- **コマンド行 (対話型)** - 一連のコマンド行のプロンプトとメッセージが表示されます。対話型のコマンド行モードを有効にするには、`-console` オプションを指定してインストールプログラムを `./setup -console` と入力して起動します。コマンド行インタフェースを使用してインストールを実行するには、`root` 権限が必要です。

Telnet でリモートサーバーにアクセスしている場合は、コマンド行インタフェースを使って対話形式で製品をインストールできます。

注 Solaris オペレーティング環境の場合、コマンド行方式を使う必要があります。セキュリティ強化された環境でインストールプログラムを開始するには、『Sun Java System Application Server リリースノート』の「他の要件と制限事項」の節にある手順を実行します。

- **サイレントモード** - インストールプログラムは、指定された設定ファイルからインストールパラメータを読み取ります。設定ファイルの生成の詳細については、[41 ページの「サイレントモードでのインストール」](#)を参照してください。

`setup` コマンドを使用すると、インストール方式を指定することができ、サイレントインストールの設定ファイルを作成することができます。

`setup` コマンド構文は、次のとおりです。 `setup [-console [-savestate]] [-savestate] [-silent config_file] [-h || -help]`

[表 1-2](#) で、`setup` コマンドオプションについて説明します。

表 1-2 `setup` コマンドのオプション

オプション	説明
<code>-console</code>	コマンド行方式を使ってインストールを実行します。
<code>-silent <i>config_file</i></code>	<p>サイレントモードでインストールを実行します。インストールパラメータは、既存のインストール設定ファイルから読み取られます。このオプションは、<code>savestate</code> オプションと一緒に選択することはできません。</p> <p>インストール設定ファイルのパスは、明示的に指定する必要があります。デフォルトのファイルパスはありません。サイレントモードのインストールとインストール設定ファイルの詳細については、37 ページの「サイレントモードでのインストール」を参照してください。</p>

表 1-2 setup コマンドのオプション (続き)

オプション	説明
-savestate	グラフィカル方式かコマンド行方式のどちらかを使ってインストールを実行し、このインストールに基づいてインストール設定ファイルを作成します。このオプションは、silent オプションと一緒に選択することはできません。このオプションを指定しないと、インストール設定ファイルは作成されません。 このファイルは statefile という名前で、install_dir にあります。
-h -help	setup コマンドで使用できるコマンド行引数を表示します。

表 1-3 に、それぞれのインストール方法で使用するコマンドを記述します。

表 1-3 すべてのインストール方法のコマンド

インストール方法	インストールコマンド
グラフィカルインタフェース (デフォルト)	./setup
コマンド行インタフェース	./setup -console
サイレントモードインストール ¹ で設定ファイルを作成するグラフィカルインタフェース	./setup -savestate
サイレントモードインストール ¹ で設定ファイルを作成するコマンド行インタフェース	./setup -console -savestate
既存のインストール設定ファイルに基づくサイレントモード	./setup -silent config_file
setup コマンドで使用できるコマンド行引数を表示	./setup -help

¹install_dir に statefile というファイルが作成されます。

Application Server ソフトウェアのインストール

この章では、Sun Java System Application Server 7 Standard Edition および Enterprise Edition をインストールする方法について説明します。

次のインストール方法について説明します。

- [Web サイトからのダウンロード](#)
- [Standard Edition のインストール](#)
- [Enterprise Edition のインストール](#)

Web サイトからのダウンロード

この節では、Web サイトから Sun Java System Application Server インストールファイルをダウンロードする方法について説明します。Web サイトから製品をダウンロードした場合は、60 日間有効のライセンスでインストールされます。

Solaris SPARC および x86 または Linux 用インストールファイルのダウンロード

1. 次の Sun Java System Application Server ダウンロードサイトでファイルを選択して、使用する Solaris SPARC および x86 または Linux マシンの一時ディレクトリにダウンロードファイルを保存します。

http://www.sun.com/software/download/app_servers.html

2. 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。次に例を示します。

```
cd /temp_dir/as7download/
```

3. `gunzip` コマンドを次の形式で使用して、`gz` ファイルを解凍します。

```
gunzip filename.tar.gz
```

4. `tar -xvf` コマンドを次の形式で使用して、圧縮ファイルを解凍します。

```
tar -xvf filename.tar
```

この処理には多少時間がかかることがあります。ファイルの展開が完了すると、`sun-appserver7` ディレクトリが作成され、その中に `setup` ファイルと `package` ディレクトリが置かれています。

ダウンロードしたファイルがファイルベース、RPM ベース (Linux)、またはパッケージベースの配布なのかを判断する方法については、17 ページの「モデルのパッケージ化とディレクトリ構造」を参照してください。

使用するプラットフォームに選択した Application Server の Edition をインストールする手順に進みます。

Microsoft Windows 用のダウンロード

1. 次の Sun Java System Application Server ダウンロードサイトでファイルを選択して、使用する Microsoft Windows マシンの一時ディレクトリにダウンロードファイルを保存します。

http://www.sun.com/software/download/app_servers.html

進行状況を表す進捗バーが、ダウンロードの完了までの目安を示します。

2. インストール zip 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。
3. インストール zip ファイルを開いて解凍し、選択したフォルダにコンテンツを展開します。ファイルの展開が完了すると、sun-appserver7 ディレクトリが作成され、その中に setup.exe ファイルと package ディレクトリおよびその他の関連ファイルが置かれています。

Microsoft Windows に Application Server Standard Edition をインストールする手順に進みます。

Standard Edition のインストール

この節では、各種配布のさまざまなプラットフォームに Application Server Standard Edition をインストールする方法について説明します。ここでは次のインストールについて説明します。

- [Standard Edition の Solaris SPARC および x86 へのインストール](#)
- [Standard Edition の Linux へのインストール](#)
- [Microsoft Windows の Standard Edition へのインストール](#)

Standard Edition の Solaris SPARC および x86 へのインストール

特に指定しない限り、ファイルベースおよびパッケージベースの配布には次の手順が適用されます。パッケージベースの配布をインストールするには、root 権限が必要です。ファイルベースの配布には root 権限は必要ありません。

1. インストールプログラムを実行します。
 - a. グラフィカルインタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで setup と入力します。
 - b. コマンド行インタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで setup -console と入力します。
2. インストールウィザード画面に従って使用許諾契約に同意し、Application Server のインストールディレクトリのパスを指定するか、デフォルトのインストールディレクトリを受け入れます。

デフォルトのインストールディレクトリは、インストールする製品によって異なります。17 ページの「モデルのパッケージ化とディレクトリ構造」を参照してください。

3. インストールしたいコンポーネントを選択します。

コンポーネントが有効でない場合、インストールプログラムはそのコンポーネントがすでにシステムにインストールされているものとして検出します。

4. Java 2 SDK のインストールを選択するか、プリインストールされている Java 2 SDK を使用します。

- 正しいバージョンの Java 2 SDK がインストールされている場合は、それが使用されます。あるいは、別の正しいバージョンのパスを入力することもできます。
- Java 2 SDK がインストールされていない場合、インストールプログラムで Java 2 SDK パッケージを自動的にインストールするように選択できます。
- パッケージベースの配布では、デフォルトのパスに正しくないバージョンの Java 2 SDK が見つかった場合、現在のバージョンにアップグレードするよう要求されます。
- ファイルベースの配布では、Java 2 SDK のインストールを選択した場合にプライベートコピーが *install_dir* にインストールされます。

デフォルトの Java 2 SDK インストールパス：

- Solaris SPARC および x86 パッケージベースの配布の場合 : /usr/j2se
- Solaris SPARC および x86 ファイルベースの配布の場合 : *install_dir*/jdk

5. パッケージベースの配布の場合、サーバー設定ディレクトリを指定する必要があります。

デフォルトのサーバー設定ディレクトリは次のとおりです。
/etc/opt/SUNWappserver7

6. パッケージベースの配布の場合、サーバードメインディレクトリを指定する必要があります。

デフォルトのサーバードメインディレクトリは次のとおりです。
/var/opt/SUNWappserver7

7. 「サーバーの設定情報」 ページ (またはコマンド行) で、次のように入力します。

- 「管理ユーザー」- サーバーの管理者の名前 (admin など)
- 「管理ユーザーのパスワード」- 管理サーバーにアクセスするためのパスワード (8 文字以上)
- 「管理サーバーポート」- 管理サーバーにアクセスするポート番号
- 「HTTP サーバーポート」- デフォルトサーバーインスタンスにアクセスするポート番号

注	<p>インストールプログラムにより使用中のポートが自動的に検出され、未使用のポートがデフォルトポートの候補として提示されます。パッケージベースの配布のデフォルトポートは HTTP サーバーでは 80 で、管理サーバーでは 4848 です。</p> <p>root 以外のユーザーとしてインストールする場合、ファイルベースの配布のデフォルトポートは HTTP サーバーでは 1024 で、管理サーバーでは 4848 です。</p>
---	--

インストールプログラムは、マシンの利用可能なディスク容量を確認します。ディスク容量が不足している場合は、エラーメッセージが表示されます。必要な最小ディスク容量については、『Sun Java System Application Server 7 リリースノート』を参照してください。

インストールステータスを示す「インストールの概要」ページが表示されます。インストールに失敗した場合は、次の場所にあるログファイルを確認してください。

- インストールログ `/var/tmp/Sun_ONE_Application_Server_install.log`
- 低レベルログ
 - ファイルベースの Solaris SPARC および x86 root ユーザー :
`/var/sadm/install/logs/Sun_ONE_Application_Server_install.timestamp`
 - ファイルベースの Solaris SPARC および x86 root 以外のユーザー :
`/var/tmp/Sun_ONE_Application_Server_install.timestamp`
 - パッケージベースの Solaris SPARC および x86:
`/var/sadm/install/logs/Sun_ONE_Application_Server_install.timestamp`

Web サイトから製品をダウンロードした場合は、60 日間有効のライセンスでインストールされます。

製品を CD からインストールする場合は、無期限の Application Server Standard Edition 製品ライセンスがインストールされます。

Standard Edition の Linux へのインストール

特に指定がないかぎり、RPM ベースおよびパッケージベースの配布には次の手順が適用されます。パッケージベースの配布をインストールするには、root 権限が必要です。RPM ベースの配布には root 権限は必要ありません。

1. インストールプログラムを実行します。
 - a. グラフィカルインタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで `setup` と入力します。

- b. コマンド行インタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで `setup -console` と入力します。
2. インストールウィザード画面に従って使用許諾契約に同意し、Application Server のインストールディレクトリのパスを指定するか、デフォルトのインストールディレクトリを受け入れます。

デフォルトのインストールディレクトリは、インストールする製品によって異なります。[17 ページの「モデルのパッケージ化とディレクトリ構造」](#)を参照してください。

3. インストールしたいコンポーネントを選択します。
コンポーネントが有効でない場合、インストールプログラムはそのコンポーネントがすでにシステムにインストールされているものとして検出します。
4. Java 2 SDK のインストールを選択するか、プリインストールされている Java 2 SDK を使用します。
 - 正しいバージョンの Java 2 SDK がインストールされている場合は、それが使用されます。あるいは、別の正しいバージョンのパスを入力することもできます。
 - Java 2 SDK がインストールされていない場合、インストールプログラムで Java 2 SDK パッケージを自動的にインストールするように選択できます。
 - RPM ベースでは、デフォルトのパスに正しくないバージョンの Java 2 SDK が見つかった場合、現在のバージョンにアップグレードするよう要求されます。
 - ファイルベースでは、Java 2 SDK のインストールを選択した場合にプライベートコピーが `install_dir` にインストールされます。

デフォルトの Java 2 SDK インストールパス：

- Linux RPM ベースの配布の場合：`/usr/java`
 - Linux ファイルベースの配布の場合：`install_dir/jdk`
5. RPM ベースの配布の場合、サーバー設定ディレクトリを指定する必要があります。
デフォルトのサーバー設定ディレクトリは次のとおりです。
`/etc/opt/SUNWappserver7`
 6. RPM ベースの配布の場合、サーバードメインディレクトリを指定する必要があります。
デフォルトのサーバードメインディレクトリは次のとおりです。
`/var/opt/SUNWappserver7`
 7. 「サーバーの設定情報」ページ (またはコマンド行) で、次のように入力します。
 - 「管理ユーザー」- サーバーの管理者の名前 (admin など)
 - 「管理ユーザーのパスワード」- 管理サーバーにアクセスするためのパスワード (8 文字以上)

- 「管理サーバーポート」- 管理サーバーにアクセスするポート番号
- 「HTTP サーバーポート」- デフォルトサーバーインスタンスにアクセスするポート番号

注 インストールプログラムにより使用中のポートが自動的に検出され、未使用のポートがデフォルトポートの候補として提示されます。パッケージベースの配布のデフォルトポート、またはファイルベースの配布の **root** ユーザーとしてのデフォルトポートは、HTTP サーバーでは **80** で、管理サーバーでは **4848** です。

root 以外のユーザーとしてファイルベースの配布をインストールする場合、デフォルトポートは HTTP サーバーでは **1024** で、管理サーバーでは **4848** です。

インストールプログラムは、マシンの利用可能なディスク容量を確認します。ディスク容量が不足している場合は、エラーメッセージが表示されます。必要な最小ディスク容量については、『Sun Java System Application Server 7 リリースノート』を参照してください。

インストールステータスを示す「インストールの概要」ページが表示されます。インストールに失敗した場合は、次の場所にあるログファイルを確認してください。

- インストールログ `:/var/tmp/Sun_ONE_Application_Server_install.log`
- 低レベルログ
 - ファイルベースの Linux の **root** ユーザーおよび **root** 以外のユーザー：
`/var/tmp/Sun_ONE_Application_Server_install.timestamp`
 - RPM ベースの Linux：
`/var/tmp/Sun_ONE_Application_Server_install.timestamp`

Web サイトから製品をダウンロードした場合は、60 日間有効のライセンスでインストールされます。

製品を CD からインストールする場合は、無期限の Application Server Standard Edition 製品ライセンスがインストールされます。

Microsoft Windows の Standard Edition へのインストール

Application Server ソフトウェアをインストールするには、管理者権限が必要です。

1. インストールファイルを解凍したディレクトリに移動します。setup.exe ファイルがあることを確認します。
2. インストールプログラムを実行します。

- a. グラフィカルインタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで `setup` と入力します。
 - b. コマンド行インタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで `setup -console` と入力します。
3. インストールウィザード画面に従って使用許諾契約に同意し、Application Server のインストールディレクトリのパスを指定するか、デフォルトのインストールディレクトリを受け入れます。
デフォルトのインストールディレクトリは次のとおりです。 `c:\¥Sun¥AppServer7`
4. インストールしたいコンポーネントを選択します。
コンポーネントが有効でない場合、インストールプログラムはそのコンポーネントがすでにシステムにインストールされているものとして検出します。
5. Java 2 SDK のインストールを選択するか、プリインストールされている Java 2 SDK を使用します。
 - o 正しいバージョンの Java 2 SDK がインストールされている場合は、それが使用されます。あるいは、別の正しいバージョンのパスを入力することもできます。
 - o Java 2 SDK がインストールされていない場合、インストールプログラムで Java 2 SDK パッケージを自動的にインストールするように選択できます。
 - o Java 2 SDK のインストールを選択した場合は、プライベートインストールが次の場所にインストールされます。 `c:\¥install¥jdk`
6. 「サーバーの設定情報」 ページ (またはコマンド行) で、次のように入力します。
 - o 「管理ユーザー」- サーバーの管理者の名前 (admin など)
 - o 「管理ユーザーのパスワード」- 管理サーバーにアクセスするためのパスワード (8 文字以上)
 - o 「管理サーバーポート」- 管理サーバーにアクセスするポート番号
 - o 「HTTP サーバーポート」- デフォルトサーバーインスタンスにアクセスするポート番号

注 インストールプログラムにより使用中のポートが自動的に検出され、未使用のポートがデフォルトポートの候補として提示されます。デフォルトポートは HTTP サーバーでは 80 で、管理サーバーでは 4848 です。

インストールプログラムは、マシンの利用可能なディスク容量を確認します。ディスク容量が不足している場合は、エラーメッセージが表示されます。必要な最小ディスク容量については、『Sun Java System Application Server 7 リリースノート』を参照してください。

インストールステータスを示す「インストールの概要」ページが表示されます。インストールに失敗した場合は、`installdir¥install.log` にあるインストールログファイルを確認してください。

Web サイトから製品をダウンロードした場合は、60 日間有効のライセンスでインストールされます。

製品を CD からインストールする場合は、無期限の Application Server Standard Edition 製品ライセンスがインストールされます。

Enterprise Edition のインストール

この節では、さまざまな配布のさまざまなプラットフォームに Application Server Enterprise Edition をインストールする方法について説明します。ここでは次のインストールについて説明します。

- [Enterprise Edition の Solaris SPARC および x86、Linux へのインストール](#)
- [Microsoft Windows での Enterprise Edition のインストール](#)

Enterprise Edition の Solaris SPARC および x86、Linux へのインストール

特に指定しないかぎり、次のインストールにここでの手順が適用されます。

- Solaris SPARC および x86、Linux でのファイルベースの配布
- Linux での RPM ベースの配布
- Solaris SPARC、x86 でのパッケージベースの配布

パッケージベースおよび RPM ベースの配布をインストールするには、root 権限が必要です。ファイルベースの配布には root 権限は必要ありません。

1. トポロジを計画したら、インストールプログラムを実行します。
 - a. グラフィカルインタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで `setup` と入力します。
 - b. コマンド行インタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで `setup -console` と入力します。

注	ロードバランサプラグインをインストールする場合は、Web サーバーをインストールしている必要があります。35 ページの「 ロードバランサプラグインコンポーネントのインストール 」を参照してください。
---	---

2. インストールウィザード画面に従って使用許諾契約に同意し、Application Server のインストールディレクトリのパスを指定するか、デフォルトのインストールディレクトリを受け入れます。

デフォルトのインストールディレクトリは、インストールする製品によって異なります。17 ページの「モデルのパッケージ化とディレクトリ構造」を参照してください。

3. パッケージベースの配布で HADB サーバーコンポーネントだけをインストールする場合は、/opt をインストールディレクトリとして選択し、HADB パッケージをデフォルトの場所である /opt/SUNWhadb にインストールします。
4. パッケージベースの配布の場合は、HADB サーバーのノードをホストするすべてのシステム上で、同じインストールディレクトリを選択する必要があります。

Sun Java System Application Server を HADB とともにインストールする場合に、デフォルトのインストールフォルダを使用しないときは、代わりにディレクトリを作成して /var/opt および /etc/opt ディレクトリからこれらのディレクトリへのソフトリンク (ln -s) を作成します。

5. インストールするコンポーネントを選択します。

コンポーネントが有効でない場合、インストールプログラムはそのコンポーネントがすでにシステムにインストールされているものとして検出します。

Web サーバーがロードバランサプラグインのインストール先のマシンにまだインストールされていない場合、ロードバランサプラグインのインストールを続行することはできません。

6. Java 2 SDK のインストールを選択するか、プリインストールされている Java 2 SDK を使用します。
 - 正しいバージョンの Java 2 SDK がインストールされている場合は、それが使用されます。あるいは、別の正しいバージョンのパスを入力することもできます。
 - Java 2 SDK がインストールされていない場合、インストールプログラムで Java 2 SDK パッケージを自動的にインストールするように選択できます。
 - パッケージベースまたは RPM ベースの配布では、デフォルトのパスに正しくないバージョンの Java 2 SDK が見つかった場合、現在のバージョンにアップグレードするよう要求されます。
 - ファイルベースの配布では、Java 2 SDK のインストールを選択した場合にプライベートコピーが *install_dir* にインストールされます。

デフォルトの Java 2 SDK インストールパス：

- Solaris SPARC および x86 パッケージベースの配布の場合 : /usr/j2se
- Linux RPM ベースの配布の場合 : /usr/java
- Solaris SPARC および x86 ファイルベースの配布の場合 : *install_dir* /jdk

注 Sun Java System Application Server Enterprise Edition ソフトウェアは、Sun Microsystems の Java 2 SDK で動作することが保証されています。サードパーティの Java 2 SDK 開発キットはバージョンに関係なくサポート対象外です。

7. パッケージベースおよび RPM ベースの配布の場合、サーバー設定ディレクトリを指定する必要があります。
デフォルトのサーバー設定ディレクトリは次のとおりです。
`/etc/opt/SUNWappserver7`
8. パッケージベースおよび RPM ベースの配布の場合、サーバードメインディレクトリを指定する必要があります。
デフォルトのサーバードメインディレクトリは次のとおりです。
`/var/opt/SUNWappserver7`
9. ロードバランサプラグインコンポーネントを選択した場合は、Web サーバーを特定します。
10. 「サーバーの設定情報」 ページ (またはコマンド行) で、次のように入力します。
 - 「管理ユーザー」- サーバーの管理者の名前 (admin など)
 - 「管理ユーザーのパスワード」- 管理サーバーにアクセスするためのパスワード (8 文字以上)
 - 「管理サーバーポート」- 管理サーバーにアクセスするポート番号
 - 「HTTP サーバーポート」- デフォルトサーバーインスタンスにアクセスするポート番号

注 インストールプログラムにより使用中のポートが自動的に検出され、未使用のポートがデフォルトポートの候補として提示されます。パッケージベースおよびファイルベースの配布の **root** 以外のユーザーとしてのデフォルトポートは、HTTP サーバーでは 80 で、管理サーバーでは 4848 です。

root 以外のユーザーとしてファイルベースの配布をインストールする場合、デフォルトポートは HTTP サーバーでは 1024 で、管理サーバーでは 4848 です。

インストールプログラムは、マシンの利用可能なディスク容量を確認します。ディスク容量が不足している場合は、エラーメッセージが表示されます。必要な最小ディスク容量については、『Sun Java System Application Server 7 リリースノート』を参照してください。

インストールステータスを示す「インストールの概要」ページが表示されます。インストールに失敗した場合は、次の場所にあるログファイルを確認してください。

- インストールログ `/var/tmp/Sun_ONE_Application_Server_install.log`
- 低レベルログ
 - ファイルベースの Solaris SPARC および x86 root ユーザー :
`/var/sadm/install/logs/Sun_ONE_Application_Server_install.timestamp`
 - ファイルベースの Solaris SPARC および x86 root 以外のユーザー :
`/var/tmp/Sun_ONE_Application_Server_install.timestamp`
 - パッケージベースの Solaris SPARC および x86:
`/var/sadm/install/logs/Sun_ONE_Application_Server_install.timestamp`
 - ファイルベースの Linux の root ユーザーおよび root 以外のユーザー :
`/var/tmp/Sun_ONE_Application_Server_install.timestamp`
 - RPM ベースの Linux:
`/var/tmp/Sun_ONE_Application_Server_install.timestamp`

Web サイトから製品をダウンロードした場合は、60 日間有効のライセンスでインストールされます。

製品を CD からインストールする場合は、無期限の Application Server Standard Edition 製品ライセンスがインストールされます。

11. サーバーを起動します。

12. HADB コンポーネントを選択した場合は、各ホストで次のように入力して、HADB ソフトウェアが正常にインストールされていることを確認します。hadbm --help

hadbm ユーティリティを使用するすべての使用可能なコマンドのリストが表示されます。hadbm は `install_dir/SUNWhadb/4/bin` にあります。

これで、システムで高可用性を利用するための設定をする準備ができました。43 ページの「HADB の設定準備」に進みます。

13. インストール後にロードバランサプラグインを選択した場合は、提供されている `loadbalancer.xml.example` ファイルを編集して実際のアプリケーションサーバーインスタンスに参照を組み込みます。このファイルは次の場所にあります。
`webserver_instance_dir/config/loadbalancer.xml.example`

HTTP ロードバランサおよびフェイルオーバーの設定方法の詳細については、『Sun Java System Application Server Administration Guide』を参照してください。

14. `loadbalancer.xml.example` ファイルを `loadbalancer.xml` として同じディレクトリに保存します。

Microsoft Windows での Enterprise Edition のインストール

Application Server ソフトウェアをインストールするには、管理者権限が必要です。

1. インストールファイルを解凍したディレクトリに移動します。
2. setup.exe を選択してインストールプログラムを実行します。
3. インストールウィザード画面に従って使用許諾契約に同意し、Application Server のインストールディレクトリのパスを指定するか、デフォルトのインストールディレクトリを受け入れます。

デフォルトのインストールディレクトリは次のとおりです。c:\Sun\AppServer7

4. インストールしたいコンポーネントを選択します。

コンポーネントが有効でない場合、インストールプログラムはそのコンポーネントがすでにシステムにインストールされているものとして検出します。

注	ロードバランサブラグインをインストールする場合は、Web サーバーをインストールしている必要があります。35 ページの「 ロードバランサブラグインコンポーネントのインストール 」を参照してください。
----------	---

5. バンドルされている Java 2 SDK のインストールを選択するか、プリインストールされている Java 2 SDK を使用します。
 - 正しいバージョンの Java 2 SDK がインストールされている場合は、既存のバージョンを使用できます。バージョンが一致しない場合は、正しいバージョンのインストール先を示すパスを入力します。
 - Java 2 SDK がインストールされていない場合は、バンドルされている Java 2 SDK のインストールを選択します。バンドルされている Java 2 SDK は次の場所にインストールされます。c:\install\jdk
6. 「サーバーの設定情報」ページ (またはコマンド行) で、次の情報を入力します。
 - 「管理ユーザー」- サーバーの管理者の名前 (admin など)
 - 「管理ユーザーのパスワード」- 管理サーバーにアクセスするためのパスワード (8 文字以上)
 - 「管理サーバーポート」- 管理サーバーにアクセスするポート番号
 - 「HTTP サーバーポート」- デフォルトサーバーインスタンスにアクセスするポート番号

注	インストールプログラムにより使用中のポートが自動的に検出され、未使用のポートがデフォルトポートの候補として提示されます。デフォルトポートは HTTP サーバーでは 80 で、管理サーバーでは 4848 です。
----------	--

インストールプログラムは、マシンの利用可能なディスク容量を確認します。ディスク容量が不足している場合は、エラーメッセージが表示されます。必要な最小ディスク容量については、『Sun Java System Application Server 7 リリースノート』を参照してください。

インストールステータスを示す「インストールの概要」ページが表示されます。インストールに失敗した場合は、Windows の temp ディレクトリの `Sun_Java_system_Application_Server_install.log` にあるインストールログファイルを確認してください。

Web サイトから製品をダウンロードした場合は、60 日間有効のライセンスでインストールされます。

製品を CD からインストールする場合は、無期限の Application Server Enterprise Edition 製品ライセンスがインストールされます。

ロードバランサプラグインコンポーネントのインストール

次の手順に従って、ロードバランサプラグインコンポーネントを個別にインストールします。

1. Web サーバーとロードバランサプラグインをホストするシステムをチェックして、インストール済みのロードバランサプラグインまたはリバースプロキシプラグインが存在するかどうかを確認します。存在する場合は、インストールプログラムを使用して削除します。
2. ロードバランサプラグインのインストール先のマシンにサポートされている Web サーバーがインストールされていることを確認します。

注	同じ権限を使用して、Web サーバーとロードバランサプラグインをインストールする必要があります。
---	--

現在サポートされている Web サーバーのバージョンについては、『Sun Java System Application Server 7 リリースノート』を参照してください。

3. Web サーバーと Web サーバーインスタンスパスを特定します。
4. 設定プログラムの手順に従ってインストールを完了させます。

注	Apache Web サーバーの場合、ロードバランサプラグインインストールプログラムは必要なファイルを Apache 1.3 では libexec フォルダに、Apache 2.0 では modules フォルダに展開します。
---	---

- インストールステータスを示す「インストールの概要」ページが表示されます。
インストールに失敗した場合は、
`/var/tmp/Sun_Java_System_application_Server_install.log`にあるインストールログファイルを確認してください。また、低レベルインストールログファイルが次の場所に作成されます。
`/var/sadm/install/logs/Sun_Java_System_Application_Server_install.timestamp`

Windows の場合、`install_dir¥install.log` のインストールログファイルが利用できます。

- Windows の場合、インストーラは `sun-passthrough.properties` ファイルに次のプロパティを追加し、それは `C:¥Inetpub¥wwwroot¥sun-passthrough¥` にインストールされます。

lb-config-file: `loadbalancer.xml` へのパス。デフォルトの場所は
`IIS_www_root¥sun-passthrough¥loadbalancer.xml`

log-file: ロードバランサログファイルへのパス。デフォルトの場所は
`IIS_www_root¥sun-passthrough.log`

log-level: ロードバランサプラグインのログレベル。デフォルトは INFO

- 5. インストール後に、提供されている `loadbalancer.xml.example` ファイルを編集して、実際のアプリケーションサーバーのインスタンスに参照を組み込みます。
このファイルは次の場所にあります。
`webserver_instance_dir/config/loadbalancer.xml.example`

Windows の場合: `ISS_www_root¥sun-passthrough¥loadbalancer.xml.example`

- 6. `loadbalancer.xml.example` ファイルを `loadbalancer.xml` として同じディレクトリに保存します。

注 Windows for IIS または Apache でロードバランサプラグインをインストールしたら、Application Server のパスを Path 環境変数に追加します。

- 「スタート」->「設定」->「コントロールパネル」->「システム」->「詳細」->「環境変数」->「システム変数」->「Path」と進み、次を追加します。`appserver_install_dir¥bin`
 - マシンを再起動する必要があります。
-

HTTP ロードバランサおよびフェイルオーバーの設定方法の詳細については、『Sun Java System Application Server Administration Guide』を参照してください。

サイレントモードでのインストール

サイレントモードインストールでは、インストール情報を得るために設定ファイルの確認を行うなどの入力操作を行わずに、インストールが実行されます。ここでは次の項目について説明します。

- [インストール設定ファイルの作成](#)
- [サイレントモードでのインストール](#)

インストール設定ファイルの作成

インストール設定ファイルは、`savestate` オプションを `setup` コマンドとともに使用して対話型インストールを開始するときに作成されます。対話型インストール中、入力は指定した設定ファイル内に収集され保管されます。この内容はサイレントインストールのテンプレートとなり、あとから 1 台または複数のマシンに製品をインストールするときに使用できます。必要に応じて、インストール設定ファイルを変更することができます。

ここでは次の項目について説明します。

- [インストール設定ファイルの作成の構文](#)
- [インストール設定ファイルの例](#)
- [インストール設定ファイルの変更](#)

インストール設定ファイルの作成の構文

インストール設定ファイルの作成の構文は、次のとおりです。

グラフィカル方式の場合 `./setup -savestate`

コマンド行方式の場合 `./setup -console -savestate`

詳細については、[21 ページの「インストール方法」](#)を参照してください。

インストール設定ファイルの例

インストール設定ファイルは次のようになります。

```
# Wizard Statefile created: Mon Jan 17 16:25:26 PST 2004

#           Wizard path:/tmp/sun-appserver7/./appserv.class

# Install Wizard Statefile for Sun Java System Application Server 7.1 Enterprise
Edition

#
```

```
[STATE_BEGIN Sun Java System Application Server
108a4222b3a6a8ed98832d45238c7e8bb16c67a5]

defaultInstallDirectory = /opt/SUNWappserver7
currentInstallDirectory = /opt/SUNWappserver7

SELECTED_COMPONENTS = Java 2 SDK, Standard Edition 1.4.21_02#Application
Server#Sun ONE Message Queue 3.5#Sample Applications#Load Balancing
Plugin#Uninstall#Startup

USE_BUNDLED_JDK = FALSE
JDK_LOCATION = /usr/j2se
JDK_INSTALLTYPE = PREINSTALLED
INSTALL_DEFAULT_CONFIG_DIR = /etc/opt/SUNWappserver7
INSTALL_CONFIG_DIR = /etc/opt/SUNWappserver7
INSTALL_DEFAULT_VAR_DIR = /var/opt/SUNWappserver7
INSTALL_VAR_DIR = /var/opt/SUNWappserver7
DOMAINS_DIR = /var/opt/SUNWappserver7/domains
WEBSERVER_INSTALL_DEFAULT_DIR = /usr/iplanet/servers
WEBSERVER_INSTALL_DIR = /opt/iplanet/servers/https-tesla.red.iplanet.com
INST_ASADMIN_USERNAME = admin
INST_ASADMIN_PASSWORD = adminadmin
INST_ASADMIN_PORT = 4848
INST_ASWEB_PORT = 81
INSTALL_STATUS = SUCCESS
[STATE_DONE Sun Java System Application Server
108a4222b3a6a8ed98832d45238c7e8bb16c67a5]
```

インストール設定ファイルの変更

インストール設定ファイルは、[表 1-4](#)に記載されている変数と値を編集することによって変更できます。

表 1-4 インストール設定ファイルの変数

変数名	有効な値 (適用可能な場合)	内容	コメント
defaultInstallDirectory		デフォルトのインストールディレクトリパス	インストールプログラムがアクティブに使用していない値
currentInstallDirectory		選択したインストールディレクトリパス	
SELECTED_COMPONENTS		インストールのために選択した製品コンポーネントのリスト	ボンド (#) 文字はリストの区切り文字として使用される

表 1-4 インストール設定ファイルの変数 (続き)

変数名	有効な値 (適用可能な場合)	内容	コメント
USE_BUNDLED_JDK	TRUE FALSE	製品にバンドルされている JDK をインストールするかどうか	
JDK_LOCATION		JDK パス	USE_BUNDLED_J2SE が false に設定されている場合はプリインストールされている JDK パス。それ以外の場合は、バンドルされている J2SE のインストール先
JDK_INSTALLTYPE	PREINSTALLED CANNOTUPGRADE UPGRADABLE CLEANINSTALL	既存の JDK インストールの処理方法	サイレントインストールの設定ファイルに有効な値は、PREINSTALLED と CLEANINSTALL だけ
INSTALL_DEFAULT_CONFIG_DIR		デフォルトの設定ファイルのディレクトリパス	インストールプログラムがアクティブに使用していない値
INSTALL_CONFIG_DIR		選択した設定ファイルのディレクトリパス	
INSTALL_DEFAULT_VAR_DIR		デフォルトドメイン設定ファイルのディレクトリパス	インストールプログラムがアクティブに使用していない値
INSTALL_VAR_DIR		選択したドメイン設定ファイルのディレクトリパス	
DOMAINS_DIR		選択したドメイン設定ファイルのディレクトリパスと、ドメインのサブディレクトリ	AS_INSTALL_VAR_DIR と DOMAINS_DIR は通常は同じ値。ただし、従来のインストールプログラムコードは両方のエントリを必要とする

表 1-4 インストール設定ファイルの変数 (続き)

変数名	有効な値 (適用可能な場合)	内容	コメント
WEBSERVER_INSTALL_DEFAULT_DIR		デフォルトの Web サーバーインスタンスのディレクトリパス	Enterprise Edition に限る インストールプログラムがアクティブに使用していない値
WEBSERVER_INSTALL_DIR		選択した Web サーバーインスタンスのディレクトリパス	Enterprise Edition に限る
INST_ASADMIN_USERNAME		初期ドメインの管理者のユーザー名	
INST_ASADMIN_PASSWORD		初期ドメインの管理者のパスワード	
INST_ASADMIN_PORT	0 - 65535	初期ドメインの管理サーバーのポート番号	
INST_ASWEB_PORT	0 - 65535	初期サーバーインスタンスのサーバーのポート番号	
INSTALL_STATUS	SUCCESS FAILURE	インストール結果	インストーラの実装によって定まる。インストールプログラムがアクティブに使用していない値

注 ある配布のために作成された設定ファイルは、Application Server の別の配布や Edition で使用することはできません。

 サイレントインストールの結果を確認するには、ログファイルを参照してください。

サイレントモードでのインストール

Sun Java System Application Server ソフトウェアをサイレントモードでインストールするには、次の手順を実行します。

1. 設定ファイルを確認して、サイレントインストールで使いたいものが含まれていることを確認します。
2. Sun Java System Application Server ソフトウェアをインストールしようとしている各マシンに、インストール設定ファイルをコピーします。
3. Application Server をインストールする各マシンに、Sun Java System Application Server インストールファイルをコピーします。
4. インストールファイルとインストール設定ファイルをコピーしたディレクトリに移動します。
5. スーパーユーザーとして、コマンド行に次の形式のコマンドを入力してサイレントインストールを開始します。 `./setup -silent config_file`

指定した *config_file* が読み込まれ、ディスク容量がチェックされます。その後、*config_file* のデータに基づいて製品がインストールされます。

プロンプトが再度表示された時点で、サイレントインストールは完了し、インストールコンポーネントがシステムにインストールされています。

6. 『Sun Java System Application Server Administration Guide』の説明に従って、Application Server ソフトウェアを開始できます。

管理コンソールが起動すると、Application Server グラフィカルインタフェースの初期ページが表示されます。

これで、システムで高可用性を利用するための設定をする準備ができました。[43 ページ](#)の「[HADB の設定準備](#)」に進んで、このプロセスを開始します。

サイレントモードでのインストール

HADB の設定準備

クラスタの一部であるサーバーに高可用性コンポーネントをインストールすると、高可用性の設定準備が整います。

ここでは次の項目について説明します。

- [Windows での HADB の設定](#)
- [共有メモリとセマフォの設定](#)
- [ホスト通信の設定](#)
- [ユーザー環境の設定](#)
- [root 以外の特権に対する管理の設定](#)
- [clsetup の使用](#)
- [時刻の同期](#)

ここで説明するタスクを実行したあと、クラスタ、ロードバランサプラグイン、および高可用性データベース (HADB) の設定と管理の詳細な手順について『Sun Java System Application Server Administration Guide』を参照してください。

高可用性のトポロジについては、『Sun Java System Application Server システム配備ガイド』を参照してください。

Windows での HADB の設定

Windows プラットフォームで使用可能な HADB のバージョンは、UNIX プラットフォームではバージョン 4.3.x であるのに対し 4.4.x です。このため、Windows での HADB の設定および管理に関し、UNIX バージョンからの変更点がいくつかあります。これらの変更は、この章の適切な箇所で説明されています。

HADB データベースとともに Application Server を設定する場合、[59 ページの「clsetup の使用」](#)で説明されているように Windows プラットフォームでは、clsetup ユーティリティを使用できます。この節の例は UNIX 専用のものです。Windows プラットフォームには、適切な変更を行ってください。

Windows プラットフォームでの HADB の設定および実行の詳細については、『Sun Java System Application Server 7 2004Q2 Update 1 Administration Guide』の第 21 章「Administering High Availability on Windows」を参照してください。

2 つの HADB バージョン間の主な違いは次のとおりです。

- 新しい HADB 管理システム。これにより、`hadbm` にはいくつかの新しいコマンドが追加され、いくつかの従来のコマンドは推奨されなくなる
- Windows プラットフォームでは RSH/SSH 設定が不要
- 新しい `adminpassword` オプションが `hadbm` コマンドに追加される。この管理パスワードは `dbpassword` とは異なる
- HADB の 4.3 バージョンのいくつかの重要な制限事項を 4.4 バージョンで削除
- `hadbm` からの管理要求を処理する管理エージェントプロセスが実行されてから、HADB インスタンスが作成可能

HADB 4.4.x のこれらおよび他の変更点については、『Sun Java System Application Server 7 2004Q2 Update 1 Administration Guide』を参照してください。

注	<p>HADB が 60-80 以上の SQL サーバープロセスを作成する場合、Windows を設定し直す必要があります。この設定は、HA ストアに過大な JDBC 接続を設定した場合に発生することがあり、さらにアプリケーションサーバーに障害が発生して再起動した後にも一時的に発生することがあります。</p> <p>設定済みではない Windows では、ポップアップウィンドウに次のエラーメッセージが表示されます。clu_sql_srv.exe: The application failed to initialize properly (0xc00000142)。このメッセージはイベントログにも登録されます。HADB サーバーログには、「Server time out waiting for sub-process, waited for 10 seconds」という警告メッセージが表示されます。</p> <p>この問題を防止するには、次のように Windows を設定します。</p> <ol style="list-style-type: none">1. データベースに対して作成される JDBC 接続の数を少なくします。あるいは、2. 次の手順に従って Windows レジストリを更新します。 <p>http://support.microsoft.com/default.aspx?scid=kb;[LN];184802 (Cause 2)</p>
---	--

共有メモリとセマフォの設定

このセクションには、UNIX プラットフォームでの HADB ホストマシンの共有メモリ設定についての手順が記載されています。HADB を使用する前に、共有メモリを設定する必要があります。

注	Windows プラットフォームの場合、この設定は必要ありません。
---	-----------------------------------

- [Solaris での共有メモリの設定](#)
- [Linux での共有メモリの設定](#)

Solaris での共有メモリの設定

1. root としてログインします。
2. 次のコマンドを共有メモリ用の /etc/system ファイルに追加します。

```
set shmsys:shminfo_shmmax=0x80000000
set shmsys:shminfo_shmseg=20
```

この例では、共有メモリ shmmax を 2G バイト (16 進 0x80000000) に設定しています。これは、ほとんどの設定に十分な値です。

shmsys:shminfo_shmmax の設定は 256M バイトごとに 0x10000000 として計算され、ホストのメモリサイズと同じに設定します。ホストのメモリを決定するには、次のコマンドを実行します。

```
prtconf | grep Memory
```

そのあと、ホストのメモリ値を次の式に入力します。

```
((<host> M バイト / 256M バイト) * 0x10000000)
```

注 shmsys:shminfo_shmmax の最大値が 0xffffffff であることに注意してください。

セマフォの場合：

/etc/system ファイルには、あらかじめ semmni、semmns、および semmnu エントリが含まれている場合があります。次に例を示します。

```
set semsys:seminfo_semmni=10
set semsys:seminfo_semmns=60
set semsys:seminfo_semmnu=30
```

このエントリが存在する場合は、次のように 16、128、および 1000 をそれぞれ追加することで、値を増加させます。

```
set semsys:seminfo_semmni=26
set semsys:seminfo_semmns=188
set semsys:seminfo_semmnu=1030
```

/etc/system ファイルに上述のエントリが含まれていない場合は、ファイルの末尾に次のエントリを追加します。

```
set semsys:seminfo_semmni=16
set semsys:seminfo_semmns=128
set semsys:seminfo_semmnu=1000
```

これは、コンピュータ上の最大 16 個の HADB ノードを起動するために十分な値です。

3. コンピュータを再起動して共有メモリの設定を有効にします。

ノード数が 16 個を超える場合については、『Sun Java System Performance Tuning Guide』の「HADB」の章を参照してください。

Linux での共有メモリの設定

1. 共有メモリを 512M バイトに増やすには、次のコマンドを実行します。

```
echo 536870912 > /proc/sys/kernel/shmmax
echo 536870912 > /proc/sys/kernel/shmall
```

shmmax は単一の共有メモリセグメントの最大サイズで、shmall は利用可能な共有メモリの合計です。

デフォルトの値を使用する標準 HADB ノードの場合は、この値で十分です。より大きなサイズのメモリを使用する場合は、この値を大きくする必要があります。

2. shmmax で指定されているデフォルトの共有メモリ制限は、proc ファイル内でマシンを再起動することなく変更できます。さらに、sysctl(8) を使用して値を変更することもできます。
3. これらの変更を適用するには、kernel.shmmax=536870912 の行を /etc/sysctl.conf ファイルに追加します。このファイルは、ブートプロセス中に使用されます。

```
echo kernel.shmmax=536870912 /etc/sysctl.conf
```

HADB ノードの説明については、『Sun Java System Application Server Administration Guide』の「Administering the High-Availability Database (Enterprise Edition)」の章の「Configuring the HADB」を参照してください。また、ストレステストとパフォーマンステストについては、『Sun Java System Application Server Performance Tuning Guide』を参照してください。

ホスト通信の設定

注 Windows プラットフォームの場合、RSH/SSH 設定は必要ありません。

HADB の管理用のリモートアクセスを実装するには、HADB サーバーと HADB 管理クライアントを稼働させるために使用するすべてのマシンが、リモートシェル (RSH) またはセキュアシェル (OpenSSH/SSH) 用に設定されている必要があります。この手順は UNIX プラットフォームの場合だけ必要です。

RSH は単純なリモートシェルコマンドであり、セキュリティ機能はありません。SSH 通信チャネルは、HADB ノード間で渡されるデータを暗号化することによって、一定レベルのセキュリティを提供します。

注 Solaris 9 および 10 の場合は、SSH をデフォルトでインストールすることをお奨めします。

Solaris 8 では、デフォルトでは SSH はインストールされません。使用している Solaris 8 システム上に SSH がない場合は、[51 ページの「Solaris 8 への SSH のインストール」](#)の手順に従ってください。

SSH を使用したいが設定されていない場合は、`hadbm` コマンドを使用できません。SSH が認識されていることを確認するには、[50 ページの「SSH の要件と制限事項」](#)を参照してください。

このセクションには、次の説明があります。

- [HADB 管理のための RSH の設定](#)
- [HADB 管理のための SSH の設定](#)

HADB 管理のための RSH の設定

SSH ではなく RSH を使用する場合は、set managementProtocol オプションを使って RSH を明示的に指定する必要があります。clresource.conf ファイルでこのパラメータを設定する方法については、[66 ページの表 2-3](#) を参照してください。

SSH は RSH よりも安全であるため、hadbm create コマンドのデフォルトとして強くお勧めします。

RSH を実装するには、次の手順を実行します。

1. root としてログインします。
2. Linux プラットフォームの場合だけ、/etc/securetty ファイルに次の内容を追加します。

```
rexec
rsh
rlogin
pts/0
pts/1
```

さらに、/etc/xinetd.d/ の下の rexec、rlogin、および rsh ファイルで、disable=no に変更します。

3. /etc/hosts ファイルを編集して、ローカルホストのホスト名など、選択したすべての HADB ホストのエントリを組み込みます。localhost の形式を使用します。次に例を示します。

```
computer1.xbay.company.com
computer99.zmtn.company.com
```

4. このファイルを、選択したすべてのインストールホストの /etc/hosts ファイルに追加します。
5. HADB ユーザーの \$HOME ディレクトリで、.rhosts ファイルを作成します。
6. グループやその他に対して読み取り専用のアクセス権が設定されていることを確認します。

各 HADB ホストのホスト名を追加します。このホスト名にはローカルホストの名前が含まれ、そのあとにデータベースユーザーの名前が続きます。たとえば、データベースユーザーが Jon の場合は次のようになります。

```
computer1.xbay.company.com    Jon
computer99.zmtn.company.com   Jon
mine456.red.mycompany.com     Jon
```

7. このファイルを各 HADB ホストの .rhosts に追加します。
8. 各ホストのホスト通信をチェックします。次に例を示します。

```
rsh computer99.zmtn.company.com uname -a
```

ID は他のホストから返されます。

HADB 管理のための SSH の設定

SSH は RSH よりも安全であるため、`hadbm create` コマンドを使用する場合に強くお勧めします。

注	セキュリティの観点から、RSA ベースのバージョン 1 のプロトコルではなく、DSA ベースのバージョン 2 のプロトコルをお勧めします。選択するバージョンは、サイトで使用している SSH クライアントソフトウェアによって異なります。
---	---

この節では、次のトピックを取り上げます。

- [SSH の要件と制限事項](#)
- [Solaris 8 への SSH のインストール](#)
- [SSH の設定](#)

SSH の要件と制限事項

注	SSH は、Solaris 9 システムにはデフォルトでインストールされますが、Solaris 8 にはデフォルトでインストールされません。Solaris 8 に SSH をインストールするには、 51 ページの「Solaris 8 への SSH のインストール」 を参照してください。
---	---

SSH を設定するときは、次の要件の一部またはすべてを考慮する必要があります。

- SSH バイナリの場所 - 高可用性の管理クライアントは、`ssh` バイナリおよび `scp` バイナリが各 HADB ホスト上の次の場所にあるものと予想します。
`/usr/bin`
 - バイナリがシステム上にあっても、その場所が正しくない場合は、`/usr/bin` から正しい場所へのシンボリックリンクを作る必要があります。
 - Solaris 8 システムの場合、SSH バイナリはデフォルトではインストールされないため、存在しない可能性があります。この場合は、[51 ページの「Solaris 8 への SSH のインストール」](#)の手順に進んでください。
- サポート - サポートは、SunSSH と OpenSSH だけが対象です。別のバージョンの SSH を使用する場合は、該当の製品のマニュアルに記載されている設定手順を参照して、SSH 通信が正しく機能することを確認してください。

- **OpenSSH クライアントおよびデーモン - OpenSSH クライアントおよびデーモン**
がある環境で実行中の場合、キーファイルに次のように名前を付けてください。
`~/.ssh/authorized_keys2` または `~/.ssh/authorized_keys`
- **root 特権で実行 - HADB 管理クライアントを root として実行している場合は、**
すべてのマシンの `sshd` 設定 (`/etc/ssh/sshd_config`) で `PermitRootLogin` パラ
メータを必ず `yes` に設定してください。

注 デフォルトでは、Sun SSH は root 特権でのログインを許可しないため、こ
れは `no` に設定されています。sshd 設定を変更した場合は、sshd を再起
動する必要があります。サービスを再起動するには、次のように入力しま
す。

`/etc/init.d/sshd stop/start`

- **SSH プロトコルのバージョン 2 をサポートしない - SSH クライアントおよびデー
モンが SSH プロトコルのバージョン 2 をサポートしていない場合は、オプション**
を付けずに `ssh-keygen` を実行する必要があります。このキーファイルに、
`id_dsa.pub` ではなく `identity.pub` という名前が付きます。このファイルは
`~/.ssh/authorized_keys` に追加する必要があります。
 - **混在した SSH 環境 - 混在した SSH 環境で操作する場合は、**
`~/.ssh/authorized_keys2` ファイルと `~/.ssh/authorized_keys` ファイルの
両方を作成する必要があります。後者には、バージョン 1 とバージョン 2 の両方
のキーが含まれている場合があります。
 - **同じ場所に置く - Sun Java System Application Server と HADB を同じマシン上の**
同じ場所に置く場合は、次のコマンドのどちらかを実行して、`.ssh` ディレクトリ
の下に `known_hosts` ファイルを作成する必要があります。
- `ssh localhost`
- または
- `ssh hostname`

Solaris 8 への SSH のインストール

Solaris 8 システムでは、ssh バイナリおよび scp バイナリはデフォルトではインス
トールされません。これらのバイナリが Solaris 8 システム上にない場合は、次の手順
を実行します。

1. 次のサイトに移動します。

<http://www.sunfreeware.com/openssh8.html>

このサイトでは、次のようなメッセージが表示されることがあります。

===PLEASE NOTE!!!..... make a note of some of the mirror sites so that if the servers are down, you can still download from a mirror site.

このようなメッセージが表示された場合は、「FTP/Mirror Sites」のリンクのリストにあるミラーサイトのいずれかにアクセスしてみてください。次に例を示します。

<http://sunfreeware.secsup.org/>

2. このサイトで、インストールの手順に従って、必要な OpenSSH のパッケージとパッチのすべてをダウンロードしてインストールします。

これで SSH を設定する準備が整いました。

SSH の設定

ssh バイナリおよび scp バイナリがすでにインストールされているシステム上に SSH を設定するには、次の節で説明する手順のどちらかを実行します。

- [マウントされていないホームディレクトリの SSH](#)
- [マウントされているホームディレクトリの SSH](#)

マウントされていないホームディレクトリの SSH

ホームディレクトリがマウントされていないシステムに SSH を実装するには、次の手順を実行します。

1. SSH 要件が理解されていて、[50 ページの「SSH の要件と制限事項」](#)に示す要件が満たされていることを確認します。
2. HADB ユーザーとしてホストにログインします。
3. 次のコマンドを実行して、キーを生成します。

```
ssh-keygen -t dsa
```

SSH1 と OpenSSH/1 の場合、通常は ssh-keygen コマンドにパラメータを付ける必要はありません。

4. 次の 3 つのプロンプトに対して、Enter を押してデフォルトのオプションを受け入れます。
5. クラスタ内のすべてのマシンに対して手順 1、2、および 3 を繰り返します。

~/.ssh ディレクトリ内にある identity.pub または id_dsa.pub というファイル (SSH のバージョン 1 とバージョン 2 のどちらを使うかによって異なる) に公開鍵が格納されます。マシンに接続する際のパスワード入力を不要にするには、このファイルの内容を、すべてのマシン上の authorized_keys というファイルに追加する必要があります。

6. ログイン ID を設定するには、ユーザーディレクトリに移動します。

~/.ssh.

SSH1、OpenSSH/1 の場合、次の手順を行います。

- a. identity.pub ファイルをコピーして、authorized_keys という名前を付けます。
- b. クラスタ内のその他のマシンでそれぞれ、identity.pub ファイルの内容をコピーし、それをローカルの authorized_keys ファイルに追加します。

OpenSSH/2 の場合、次の手順を行います。

- a. id_dsa.pub ファイルをコピーして、authorized_keys2 という名前を付けます。
 - b. クラスタ内のその他のマシンでそれぞれ、id_dsa.pub ファイルの内容をコピーし、それをローカルの authorized_keys2 ファイルに追加します。
7. すべての HADB マシンの ~/.ssh ディレクトリに、authorized_keys ファイルをコピーします。
 8. ssh ディレクトリ、HADB ユーザーのホームディレクトリ、および ~/.ssh/authorized_keys ファイルに対して、グループとその他の書き込み権がないことを確認します。

必要に応じて、これらのグループやその他の書き込み権を、次のようにして無効にします。

```
chmod og-w ~/.ssh
chmod og-w ~/.ssh/authorized_keys
chmod og-w $HOME
```

\$HOME を HADB ユーザーのホームディスクと置き換えます。たとえば、次のようにします。

```
chmod og-w ~/johnsmith
```

注	<p>~/.ssh ディレクトリの下ファイルに対してグループやその他に読み取り権が与えられている場合でも、自動 SSH ログイン ID を設定することはできません。この場合、ssh machine_name を実行しようとすると、システムによりアクセス権が正しくないことが示され、パスワードの入力が求められます。このため、自動ログインを有効にする場合は、グループとその他にアクセス権を与えないことをお勧めします。</p>
---	--

9. ユーザーが入力しなくてもログインできるようにするには、SSH 環境を設定したあとにはじめて SSH を使用するときに、/.ssh ディレクトリの下 known_hosts ファイルへ次のようにノードマシン名を追加する必要があります。

- a. 次のように入力します。

```
ssh machine_name
```

`machine_name` を `known_hosts` ファイルに追加するかどうかを確認する「Yes/No」の質問が表示されます。

- b. 「Yes」と答えます。

これで、入力しなくてもログインできるようになります。

10. SSH が正しく設定されたことを確認するには、HADB の管理ツールを実行する前に、クラスタ内の各ホストに対して SSH の確認を行います。

パスワードの入力は必要なく、自動ログインします。

マウントされているホームディレクトリの SSH

ホームディレクトリがマウントされているシステムに SSH を実装するには、次の手順を実行します。

1. 50 ページの「SSH の要件と制限事項」に示すように、SSH 要件が満たされていることを確認します。
2. HADB ユーザーとしてホストにログインします。
3. 次のコマンドを実行して、キーを生成します。

```
ssh-keygen -t dsa
```

SSH1 と OpenSSH/1 の場合、通常は `ssh-keygen` コマンドにパラメータを付ける必要はありません。

4. 次の 3 つのプロンプトに対して、Enter を押してデフォルトのオプションを受け入れます。

~/.ssh ディレクトリ内にある `identity.pub` または `id_dsa.pub` というファイル (SSH のバージョン 1 とバージョン 2 のどちらを使うかによって異なる) に公開鍵が格納されます。マシンに接続する際にパスワードの入力が求められないようにするには、このファイルの内容を、すべてのマシン上の `authorized_keys2` というファイルに追加する必要があります。これは次のようにして行います。

5. ログイン ID を設定するには、ユーザーディレクトリに移動します。

```
~/.ssh.
```

SSH1、OpenSSH/1 - `identity.pub` ファイルをコピーし、それに `authorized_keys` という名前を付けます。

OpenSSH/2 - `id_dsa.pub` ファイルをコピーし、それに `authorized_keys` という名前を付けます。

6. `ssh` ディレクトリと `~/.ssh/authorized_keys` ファイルに、グループとその他に対する書き込み権がないことを確認します。

必要に応じて、これらのグループやその他の書き込み権を、次のようにして無効にします。

```
chmod og-w ~/.ssh
chmod og-w ~/.ssh/authorized_keys
chmod og-w /$HOME
```

HOME を HADB ユーザーのホームディスクと置き換えます。次に例を示します。

```
chmod og-w ~/johnsmith
```

注

~/.ssh ディレクトリの下ファイルに対してグループやその他に読み取り権が与えられている場合でも、自動 SSH ログイン ID を設定することはできません。この場合、ssh *machine_name* を実行しようとすると、システムによりアクセス権が正しくないことが示され、パスワードの入力が求められます。このため、自動ログインを有効にする場合は、グループとその他にアクセス権を与えないことをお勧めします。

7. ユーザーが入力しなくてもログインできるようにするには、SSH 環境を設定したあとにはじめて SSH を使用するとき、/.ssh ディレクトリの下 known_hosts ファイルへノードマシン名を追加する必要があります。

```
ssh machine_name
```

machine_name を known_hosts ファイルに追加するかどうかのメッセージが表示されたら、「Yes」と答えます。これで、何も入力することなくログインできます。

8. SSH が正しく設定されたことを確認するには、HADB の管理ツールを実行する前に、クラスタ内の各ホストに対して SSH の確認を行います。

パスワードの入力は必要なく、自動ログインします。

Solaris で VERITAS ファイルシステムを使う場合の HADB の使用法

Solaris で VERITAS ファイルシステムを使用する場合、HADB 履歴ファイルのメッセージ WRN: Direct disk I/O mapping failed を取得します。

このメッセージは、HADB がデータおよびログデバイスの「direct I/O」をオンすることを不可能であることを示します。direct I/O とは、ディスクのページを記述する CPU コストを削減し、さらにオペレーティングシステムのダーティデータページを管理するオーバーヘッドも少なくするパフォーマンス強化策です。

VERITAS で direct I/O を使用するには、オプション mincache=direct でマウントされるファイルシステムに、データおよびログデバイスを作成する必要があります。このオプションは、ファイルシステムで作成されるすべてのファイルに適用されることに注意してください。詳細については、コマンド mount_vxfs(1M) を参照してください。

もう 1 つの方法は VERITAS Quick I/O 機能を使用することです。事実上、この製品は raw I/O を行い、ファイルシステムのファイルを可能にします。詳細については、VERITAS のマニュアル『VERITAS File System 4.0 Administrator's Guide for Solaris』を参照してください。

注

この説明は入手可能なマニュアルに基づくものに限られます。

Sun Database Technology Group はこれらの設定をテストしていません。

Linux で ext ファイルシステムを使う場合の HADB の使用方法

HADB は Red Hat Advanced Server 2.1 での ext3 ファイルシステムをサポートしていません。ext2 だけをサポートしています。ext3 ファイルシステムは Red Hat Advanced Server 3.0 の HADB でサポートされます。

ユーザー環境の設定

注 この章に記載されている手順は UNIX 専用の例です。注記がある場合を除いて、同じコマンドおよび例は適切に変更することで、Microsoft Windows プラットフォームにも適用できます。

ホスト通信を設定すると、次のようにして、`hadbm` コマンドを `install_dir/SUNWhadb/4/bin` ディレクトリから実行できます。

```
./hadbm
```

ただし、高可用性の管理クライアントのコマンドをどこからでも使用できるようにローカル環境を設定すると、より便利です。これを設定するには、次の手順を実行します。

注 この節の例では、`csch` を使用します。別のシェルを使用する場合は、使用するシェルのマニュアルページで変数の設定手順を参照してください。

1. `PATH` 変数を次のように設定します。

```
setenv PATH ${PATH}:install_dir/bin:install_dir/SUNWhadb/4/bin
```

2. 次のコマンドを実行して、`PATH` 設定が正しいことを確認します。

```
which asadmin
which hadbm
```

3. 複数の Java バージョンがインストールされている場合は、`JAVA_HOME` 環境が Enterprise Edition の JDK バージョン 1.4.2_05 にアクセスしていることを確認します。

```
setenv JAVA_HOME java_install_dir
setenv PATH ${PATH}:${JAVA_HOME}/bin
```

4. `HADB` デバイスファイルおよびログファイルがデフォルトの場所がない場合 (`appserver_install_dir/SUNWhadb/4`)、次の `hadbm` コマンドを使用してこれら重要なファイルを探索します。

```
hadbm get configpath
hadbm get devicepath
hadbm get historypath
hadbm get installpath
```

これらのコマンドで表示された場所をバックアップします。

root 以外の特権に対する管理の設定

デフォルトでは、Sun Java System Application Server の初期インストールまたは設定時、Sun Java System Application Server 用に作成されたファイルとパスの書き込み権はルートだけに与えられます。ルート以外のユーザーが Sun Java System Application Server を作成したり管理したりする場合、関連するファイルの書き込み権を、その特定のユーザー、またはそのユーザーが属するグループに与える必要があります。影響を受けるファイルとそのデフォルトの場所は次のとおりです。

- Sun Java System Application Server 設定ファイル - *install_config_dir/cl*.conf*
- Sun Java System Application Server 設定および管理スクリプト - *install_dir/bin/cl**
- HADB バイナリ - *install_dir/SUNWhadb*
- HADB 設定 - */etc/opt/SUNWhadb*
- clsetup ログファイルの場所 - */var/tmp*

次の手順で説明するように、Sun Java System Application Server を管理するためのユーザーグループを作成できます。もう 1 つの方法は、特定のユーザーにアクセス権と所有権を設定することです。

Sun Java System Application Server のユーザーグループを作成し、インストールルートディレクトリのアクセス権を設定するには、影響を受けるファイルのそれぞれに対して次の手順を繰り返します。

1. root としてログインします。
2. コマンドプロンプトから、Sun Java System Application Server ユーザーグループを作成します。次に例を示します。

```
# groupadd slasuser
```

コマンド行で groupadd と入力して、適切な使用法を表示します。

3. 影響を受けるファイルのそれぞれのグループ所有権を、新たに作成したグループに変更します。次に例を示します。

```
chgrp -R sjsasuser install_config_dir/cl*.conf
```

4. 新しく作成したグループの書き込み権を設定します。

```
chmod -R g+rw install_config_dir/cl*.conf
```

5. 影響を受けるファイルのそれぞれで、手順 3 と 4 を繰り返します。

6. clsetup コマンドと cladmin コマンドを、新しく作成したグループで作成できるようにします。次に例を示します。

```
chmod -R g+x install_dir/bin/cl*
```

7. `--sysuser` オプションを使って、デフォルトのドメインである `domain1` を削除してから再度作成します。`sysuser` も新たに作成したグループに属する必要があります。次に例を示します。

```
asadmin delete-domain domain1

asadmin create-domain --sysuser bleonard --adminport 4848
--adminuser admin --adminpassword password domain1
```

clsetup の使用

`clsetup` ユーティリティの目的は、一般的な設定で基本クラスタを設定するプロセスを自動化することです。`clsetup` コマンドは `install_dir/bin` にあります。`install_dir` は、Sun Java System Application Server ソフトウェアがインストールされているディレクトリです。

`clsetup` ユーティリティは、`cladmin` コマンドとともに Sun Java System Application Server ソフトウェアにバンドルされています。

注	<code>cladmin</code> コマンドは、インストールと設定のすべてのタスクが完了したあとで、クラスタの設定と管理のプロセスを効率化するために使用します。HADB の作成方法と <code>cladmin</code> コマンドの使用方法については、『Sun Java System Application Server Administration Guide』を参照してください。
---	--

ここでは次の項目について説明します。

- [clsetup の機能](#)
- [clsetup の要件と制限事項](#)
- [clsetup 入力ファイルの編集](#)
- [clsetup の実行](#)
- [clsetup のクリーンアップ手順](#)

clsetup の機能

clsetup ユーティリティは、Sun Java System Application Server コマンドのセットです。これを使えば、事前に内容が設定されている入力ファイルに基づいて、自動的にクラスタを設定できます。クラスタ設定の一部として、HADB が作成されます。ただし、『Sun Java System Application Server Administration Guide』で説明されているように、hadbm コマンドを使用して作業用クラスタを設定する必要があります。

注	clsetup ユーティリティインタフェースは不確定です。不確定なインタフェースとは試験的または一時的なインタフェースであるため、次のリリースで互換性がなくなったり、削除されたり、または安定したインタフェースに置き換えられたりする場合があります。
---	---

この節では次の項目について説明します。

- [入力ファイルの機能](#)
- [clsetup が実行する内容](#)
- [clsetup が使用するコマンド](#)

入力ファイルの機能

clsetup ユーティリティでは、次の 3 つの入力ファイルを使ってクラスタを設定します。

- `clinstance.conf` - このファイルは、アプリケーションサーバーインスタンスの `server1` と `server2` に関する情報が事前に設定されています。このファイルの内容の詳細については、[64 ページの「clinstance.conf ファイル」](#)を参照してください。
- `clpassword.conf` - このファイルには、Sun Java System Application Server 7.1 Enterprise Edition ソフトウェアをインストールしたときに指定した、`domain1` の管理サーバーパスワードが事前に設定されています。このファイルの内容の詳細については、[65 ページの「clpassword.conf ファイル」](#)を参照してください。
- `clresource.conf` - このファイルには、クラスタリソースである HADB、JDBC 接続プール、JDBC リソース、およびセッションストアとセッション持続性に関する情報が事前に設定されています。このファイルの内容の詳細については、[66 ページの「clresource.conf ファイル」](#)を参照してください。

注	クラスタの設定に必要な設定パラメータは常に入力ファイルから読み取られ、コマンド行から指定することはできません。
---	---

一般的なクラスタの設定をするようあらかじめ設定されている、clsetup 設定パラメータを使用します。別の設定をサポートするには、設定ファイルの一部、またはすべてを編集します。

clsetup が実行する内容

clsetup 入力ファイルに事前に設定されている値を使って、clsetup ユーティリティコマンドは次のことを行います。

- domain1 というデフォルトのドメイン内に、server2 という新しいサーバーインスタンスを作成します。server2 の HTTP ポート番号は、インストール時に server1 に指定した HTTP ポート番号の次に続く番号です。たとえば、インストール時に server1 にポート番号 80 を指定すると、server2 のポート番号は 81 です。
- ローカルマシン上に 2 つのノードを持つ hadb という HADB を作成します。ポートベースは 15200 で、データベースパスワードは password です。
- HADB にセッション情報を格納するために必要な HADB 表を作成します。
- clinstance.conf ファイル (server1、server2) にリストされているすべてのインスタンスに、appservCPL という接続プールを作成します。
- clinstance.conf ファイル (server1、server2) にリストされているすべてのインスタンスに、jdbc/hastore という JDBC リソースを作成します。
- clinstance.conf ファイル (server1、server2) にリストされているすべてのインスタンスに、セッション持続性情報を設定します。
- clinstance.conf ファイル (server1、server2) にリストされているすべてのインスタンスに RMI/IIOP クラスタを設定し、RMI/IIOP フェイルオーバーを有効にします。
- clinstance.conf ファイル (server1、server2) にリストされているすべてのインスタンスで、SFSB フェイルオーバーを設定します。
- clinstance.conf ファイル (server1、server2) にリストされているすべてのインスタンスで高可用性を有効にします。

注 clresource.conf および clpassword.conf 入力ファイルはパスワードを格納するため、0600 アクセス権でアクセス保護されています。

clsetup が使用するコマンド

clsetup ユーティリティは、多数の hadbm コマンドと asadmin コマンドを使ってクラスタを設定します。表 2-1 では、左側の列に clsetup のタスクを示し、右側の列にタスクの実施に使われるコマンドを示しています。

表 2-1 clsetup ユーティリティが使用する hadbm および asadmin コマンド

clsetup で実行するタスク	コマンド
データベースが存在するかどうか をチェックします	hadbm status
HADB を作成して起動します	hadbm create
JDBC URL を取得します	hadbm get jdbcURL
セッションストアを作成します	asadmin create-session-store
インスタンスの状態をチェックし ます	asadmin show-instance-status
インスタンスを作成します	asadmin create-instance
JDBC 接続プールを作成します	asadmin create-jdbc-connection-pool
データソースを登録します	asadmin create-jdbc-resource
持続タイプを設定します	asadmin configure-session-persistence
RMI/IIOP フェイルオーバーを設定 します	asadmin add-iiop-cluster-endpoint
SFSB フェイルオーバーを設定しま す	asadmin set
インスタンスを再設定します	asadmin reconfig -u admin

clsetup の要件と制限事項

clsetup ユーティリティには、次の要件と制限が適用されます。

- インストールパス、デバイスパス、設定パスなどは、クラスタの一部を構成するすべてのマシン上で同じでなければいけません。
- clsetup を使用する前に、ローカルマシンで asadmin ユーティリティおよび hadbm ユーティリティが利用可能になっている必要があります。clsetup は、次のコンポーネントがインストールされているマシンだけで実行できます。
 - Sun Java System Application Server コンポーネントまたは Sun Java System Application Server 管理クライアントのコンポーネント
 - HADB コンポーネントまたは HADB 管理クライアントのコンポーネント
- clsetup を使用するには、[45 ページ](#)の「[共有メモリとセマフォの設定](#)」の説明に従って、UNIX プラットフォーム用の共有メモリを設定します。clsetup ユーティリティは共有メモリの値を設定しません。

Windows プラットフォームの場合、この手順は必要ありません。

- clsetup ユーティリティを使用するには、[48 ページの「ホスト通信の設定」](#)の説明に従って、UNIX プラットフォームで SSH または RSH 用の HADB クラスタホスト通信を設定します。

Windows プラットフォームの場合、この手順は必要ありません。

- デフォルトではない RSH を使用する場合は、clresource.conf ファイルの次の行のコメント指定を外します。

```
#set managementProtocol=rsh
```

- SSH を使用する同じマシン上で Application Server と HADB を同じ場所に置く場合は、known_hosts ファイルが .ssh ディレクトリの下に存在している必要があります。このファイルが存在しない場合は、clsetup を使用する前に、ssh localhost または ssh hostname コマンドのどちらかを実行します。
- clsetup を実行する前に、クラスタの一部である Sun Java System Application Server のすべてのインスタンスの管理サーバーを起動する必要があります。
- 管理者のパスワードは、クラスタの一部であるすべてのドメイン間で同じである必要があります。
- 処理するエントリ (HADB ノードと Application Server インスタンス) がすでに存在する場合、clsetup コマンドはそれらを削除したり再設定したりしないで、各設定手順は省略されます。
- 入力ファイルで指定した値は、クラスタ内のすべてのインスタンスで同じになります。clsetup は異なる値でインスタンスを設定するように設計されていません。たとえば、clsetup は、インスタンスごとに設定が異なる JDBC 接続プールは作成できません。
- clsetup では、inetd は設定されません。HADB は inetd 設定なしで作成されます。inetd の設定手順については、『Sun Java System Application Server Administration Guide』を参照してください。

Windows プラットフォームの場合、この設定は必要ありません。

- シェル初期化ファイル内のホスト名 - .cshrc または .login ファイルにホスト名とともにプロンプトを組み込むと、clsetup が停止したように見えることがあります。リモートコマンドの実行中のプロンプトと余剰な出力を削除します。たとえば、hostB で hostname コマンドを実行したときに、プロンプトなしで hostB が印刷される必要があります。
- clsetup を root 以外のユーザーとして実行するには、[「root 以外の特権に対する管理の設定」](#)で説明されている手順に従ってください。

clsetup 入力ファイルの編集

clsetup コマンドに必要な入力ファイルは、インストール処理の中で設定インストールディレクトリ (デフォルトは、UNIX では /etc/opt/SUNWappserver7、Windows では c:\¥Sun¥AppServer7¥config¥) の下にインストールされます。これらの入力ファイルには通常設定の値が事前に入力されており、使用する設定に応じて変更できます。

このセクションには、次の内容があります。

- [clinstance.conf ファイル](#)
- [clpassword.conf ファイル](#)
- [clresource.conf ファイル](#)

clinstance.conf ファイル

clsetup を動作させるため、クラスタの一部であるすべてのアプリケーションサーバーのインスタンスが clinstance.conf ファイルに定義されている必要があります。インストール中に、2 つのインスタンスのエントリにより clinstance.conf ファイルが作成されます。クラスタにインスタンスを追加する場合は、そのインスタンスに関する情報を次のように追加する必要があります。

```
# Comment
instancename instance_name
user user_name
host localhost
port admin_port_number
domain domain_n
instanceport instance_port_number
```

クラスタの一部であるインスタンスごとに 1 組のエントリが必要です。ハッシュマーク (#) で始まる行はコメントとして扱われます。

注

これらのエントリの clinstance.conf ファイルでの順序は重要であるため、ここで示す順序は変更しないでください。その他のアプリケーションサーバーインスタンスに関する情報を追加する場合、これらのインスタンスのエントリは同じ順序で表示される必要があります。コメントはファイルの任意の場所に追加できます。

表 2-2 では、clinstance.conf ファイル内のエントリに関する情報を説明しています。左側の列にはパラメータ名、中央の列にはパラメータの定義、右側の列にはデフォルト値を示しています。

表 2-2 clinstance.conf ファイルのエントリ

パラメータ	定義	デフォルト値
instancename	Application Server のインスタンス名	server1、server2
user	管理サーバーのユーザー名	admin
host	ホスト名	localhost
port	管理サーバーのポート番号	4848
domain	管理ドメイン名	domain1
instanceport	Application Server インスタンスポート	80, 81
master	マスターインスタンス (クラスタ検証に使用)	false

clinstance.conf ファイルの例

この clinstance.conf ファイルには、2 つのインスタンスに関する情報が含まれています。

```
#Instance 1

instancename server1
user admin
host localhost
port 4848
domain domain1
instanceport 80

#Instance 2

instancename server2
user admin
host localhost
port 4848
domain domain1
instanceport 81
```

clpassword.conf ファイル

clsetup を実行するとき、asadmin コマンドには管理サーバーのパスワードが必要です。このパスワードは、インストール時に clpassword.conf ファイルに指定されます。

clpassword.conf ファイルの形式は次のとおりです。

```
AS_ADMIN_PASSWORD= password
```

password は管理サーバーのパスワードです。

アクセス権 0600 が `clpassword.conf` ファイルに事前設定されています。このファイルにアクセスできるのは `root` ユーザーだけです。

clresource.conf ファイル

インストール中に `clresource.conf` ファイルが作成されて、通常の設定が行われます。`clresource.conf` ファイルには、クラスタの一部である次のリソースに関する情報が含まれています。

- HADB
- セッションストア
- JDBC 接続プール
- JDBC リソース
- セッション持続性

UNIX プラットフォームでは、アクセス権 0600 が `clresource.conf` ファイルに事前設定されています。このファイルにアクセスできるのは `root` ユーザーだけです。

注	<div>clsetup を実行する前に、最適化や別の設定のために、 clresource.conf ファイルに指定した値を変更することができます。値 を編集する場合は、ファイルの順序と形式を変更しないでください。 ハッシュマーク (#) で始まる行はコメントとして扱われます。</div>
---	--

次の表では、`clresource.conf` ファイルのパラメータについて説明しています。左側の列にはパラメータ名、中央の列にはパラメータの定義、右側の列にはデフォルト値を示しています。

表 2-3 では、`clresource.conf` ファイルの HADB パラメータについて説明しています。

表 2-3 clresource.conf ファイルの HADB パラメータ

パラメータ	定義	デフォルト値
historypath	履歴ファイルのパス	/var/tmp Windows の場合 : <i>REPLACEDIR</i> (<i>REPLACEURL</i> は実行時の実際の URL と置き換えられる)

表 2-3 clresource.conf ファイルの HADB パラメータ (続き)

パラメータ	定義	デフォルト値
devicepath	データおよびログデバイスのパス	/opt/SUNWappserver7/SUNWhadb/4 Windows の場合 : C:\Sun\JavaServer7\SUNWhadb\4.4.0-14
datadevices	各ノード上のデータデバイスの数	1
portbase	ノード 0 に使われるポートベース番号。別のノードには、ここで示す番号から 20 ずつ増分されたポート番号 (10000 ~ 63000 の範囲のランダムな番号) が割り当てられます	15200
spares	スペアノードの数	0
set	コンマで区切られた、データベース設定属性のリスト	有効なデータベース設定属性の説明については、『Sun Java System Application Server 7.1 Administration Guide』を参照 たとえば、SSH (デフォルト) ではなく RSH を使用するよう指定するには、次の行のコメント指定を外します。#set managementProtocol=rsh
inetd	HADB が inet デーモンを使って実行されているかどうかを示します Windows プラットフォームでは適用不可	false
inetdsetupdir	inet デーモンの設定ファイルがあるディレクトリ Windows プラットフォームでは適用不可	/tmp
devicesize	デバイスのサイズ (M バイト)。このサイズはすべてのデバイスに適用される	512
dbpassword	HADB ユーザーのパスワード	password
hosts	すべてのデータノードに使用されるすべてのホスト	値は、インストール時に指定したホストに基づいて自動的に設定される

データベース名は、clresource.conf ファイルの [HADBINFO] セクションの末尾に指定されます。

表 2-4 では、clresource.conf ファイルのセッションストアパラメータについて説明しています。

表 2-4 clresource.conf ファイルのセッションストアパラメータ

パラメータ	定義	デフォルト値
storeurl	HADB ストアの URL	REPLACEURL 注：値は実行時の実際の URL と置き換えられます
storeuser	セッションストアへのアクセス権を持つユーザー	appservusr 注：表 2-5 の username プロパティと一致する必要があります
storepassword	storeuser のパスワード	password 注：表 2-5 の password プロパティと一致する必要があります
dbssystempassword	HADB システムユーザーのパスワード	password
adminpassword	ドメインを管理する管理者のパスワード。hadbm createdomain または hadbm create とともに adminpassword オプションを使用する場合には、hadbm コマンドを使うたびにこのパスワードを入力する必要がある (Windows の場合に限る)	なし
adminpasswordfile	adminpasswordfile オプションは、パスワードを含むファイルへのパスとしてパスワードを提供するために使用 (Windows の場合に限る)	なし
no-adminauthentication	no-adminauthentication オプションで、管理者は管理者のパスワードを入力することなくすべての hadbm コマンドが使用可能 (Windows の場合に限る)	なし

表 2-5 では、clresource.conf ファイルの JDBC 接続プールパラメータについて説明しています。

表 2-5 clresource.conf ファイルの JDBC 接続プールパラメータ

パラメータ	定義	デフォルト値
steadypoolsize	プールに保持される接続の最小数および初期数	8
maxpoolsize	作成できる接続の最大数	32
datasourceclassname	ベンダーから提供される JDBC データソースの名前 ベンダーから提供される JDBC データソースに対応するデータソースクラスの名前は、javax.sql.XADataSource インタフェースを実装する XA に対応していない、またはローカルトランザクション専用のデータソースでは、javax.sql.DataSource インタフェースが実装される	com.sun.hadb.jdbc.ds.HaDataSource
isolationlevel	プールされているデータベース接続のトランザクション遮断レベルを指定する	repeatable-read
isisolationguaranteed	トランザクション遮断レベルが保証される	true
validationmethod	検証方法の種類を指定する	meta-data
property	ユーザー名、パスワード、およびリソース設定の指定に使われるプロパティ	username=appservusr:password=password:cacheDataBaseMetaData=false:eliminateRedundantEndTransaction=true:serverList=REPLACEURL 注: ユーザー名とパスワードのプロパティでは、必ず、セッションストアパラメータの表に示されている値と同じ値を使用してください。REPLACEURL は、実行時の実際の URL と置き換えられます

接続プール名は、clresource.conf ファイルの [JDBC_CONNECTION_POOL] セクションの末尾に指定されます。

表 2-6 では、clresource.conf ファイルの JDBC リソースパラメータについて説明しています。

表 2-6 clresource.conf ファイルの JDBC リソースパラメータ

パラメータ	定義	デフォルト値
connectionpoolid	接続プールの名前	appservCPL
注：接続プール名は表 2-5 に指定されています。		

JDBC リソース名は、clresource.conf ファイルの [JDBC_RESOURCE] セクションの末尾に定義されています。

表 2-7 では、clresource.conf ファイルのセッションの持続性パラメータについて説明しています。

表 2-7 clresource.conf ファイルのセッション持続性パラメータ

パラメータ	定義	デフォルト値
type	セッション持続性タイプ	ha
frequency	セッション頻度	web-method
scope	セッションの範囲	セッション
store	セッションストア	jdbc/hastore
注：ストア名は、[JDBC_RESOURCE] セクションの末尾に定義されています		

表 2-8 では、clresource.conf ファイルのステートフルセッション Bean パラメータについて説明しています。

表 2-8 clresource.conf ファイルのステートフルセッション Bean パラメータ

パラメータ	定義	デフォルト値
sfsb	ステートフルセッション Bean フェイルオーバー	false

表 2-9 では、clresource.conf ファイルの RMI/IIOP フェイルオーバーパラメータについて説明しています。

表 2-9 clresource.conf ファイルの RMI/IIOP フェイルオーバーパラメータ

パラメータ	定義	デフォルト値
rmi_iiop	RMI/IIOP クラスタ設定	false

表 2-10 では、clresource.conf ファイルのクラスタ識別パラメータについて説明しています。

表 2-10 clresource.conf ファイルの クラスタ識別パラメータ

パラメータ	定義	デフォルト値
cluster_id	クラスタ ID	cluster1

clresource.conf ファイルの例

```
[HADBINFO]
historypath /var/tmp
devicepath /opt/SUNWappserver7/SUNWhadb/4
datadevices 1
portbase 15200
spares 0
#set      managementProtocol=rsh
inetd false
inetdsetupdir /tmp
devicesize 512
dbpassword password
hosts machine1,machine1
hadb

[SESSION_STORE]
storeurl          REPLACEURL
storeuser         appservusr
storepassword     password
dbssystempassword password

[JDBC_CONNECTION_POOL]
steadypoolsize    8
maxpoolsize       32
datasourceclassname com.sun.hadb.jdbc.ds.HadbDataSource
isolationlevel    repeatable-read
```

```
validationmethod      meta-data
property
username=appservusr:password=password:cacheDataBaseMetaData=false:eliminateRedundantEndTransaction=true:serverList=REPLACEURL

appservCPL

[JDBC_RESOURCE]
connectionpoolid      appservCPL
jdbc/hastore

[SESSION_PERSISTENCE]
type                  ha
frequency             web-method
scope                 session
store                 jdbc/hastore
```

Windows での clresource.conf ファイルの例

```
[HADBINFO]

package V4.4

historypath    REPLACEDIR

devicepath     C:\Sun\AppServer7\SUNWhadb\4.4.0-12

datadevices    1

portbase       15200

spares         0

#set    LogbufferSize=32,DataBufferPoolSize=128

devicesize     208

dbpassword     password

hosts          machine1,machine2

adminpassword  password

hadb

[SESSION_STORE]

storeurl       EPLACEURL

storeuser      appservusr

storepassword  password

dbssystempassword password

[JDBC_CONNECTION_POOL]

steadypoolsize 8
```



```

maxpoolsize      2
datasourceclassname    com.sun.hadb.jdbc.ds.HadbDataSource
isolationlevel      repeatable-read
--isolationguaranteed=true
validationmethod      meta-data
property      username=appservusr:password=password:cacheDataBaseMeta
Data=false:eliminateRedundantEndTransaction=true:serverList=REPLACE
URL
appservCPL
[JDBC_RESOURCE]
connectionpoolid      appservCPL
jdbc/hastore
[SESSION_PERSISTENCE]
type      ha
frequency      web-method
scope      session
store      jdbc/hastore
[EJB_FAILOVER]
sfsb      true
[RMI_IIOP_FAILOVER]
rmi_iiop      true
[CLUSTER_ID]
cluster_id      cluster1

```

clsetup の実行

clsetup の実行の構文は次のとおりです。

```
clsetup [--help] [--instancefile instance_file_location] [--resourcefile  
resource_file_location] [--passwordfile password_file_location] verify
```

引数が指定されていない場合、clsetup は次のデフォルトを使用します。

--instancefile は *install_config_dir*/clinstance.conf

--resourcefile は *install_config_dir*/clresource.conf

--passwordfile は *install_config_dir*/clpassword.conf

カスタムの入力ファイルの場所を指定することで、これらの引数を無効にできます。次に例を示します。

```
./clsetup --resourcefile /tmp/myappservresource.conf
```

カスタムの入力ファイルを指定するときは、入力ファイルにある必須形式に従ってください。この手順については、[64 ページの「clsetup 入力ファイルの編集」](#)を参照してください。

clsetup を実行するには、次の手順を行います。

1. [62 ページの「clsetup の要件と制限事項」](#)に記載されている要件が満たされていることを確認します。

注

clsetup を root 以外のユーザーとして実行する場合は、[58 ページの「root 以外の特権に対する管理の設定」](#)を参照して設定を行ってください。

2. 入力ファイルにクラスタの設定に必要な情報があることを確認します。必要に応じて、[64 ページの「clsetup 入力ファイルの編集」](#)のガイドラインに従って入力ファイルを編集します。
3. RSH を使用する場合は、clresource.conf ファイルを編集して次の行のコメント指定を外します (# 記号を削除します)。#set managementProtocol
4. Sun Java System Application Server インストールディレクトリ下の /bin ディレクトリに移動します。
5. clsetup コマンドを実行します。

UNIX の場合: ./clsetup

Windows の場合: clsetup.bat

clsetup コマンドにより詳細モードで実行されます。さまざまなコマンドが実行中に画面に表示され、出力はログファイル /var/tmp/clsetup.log (UNIX の場合)、またはデフォルトの Windows temp ディレクトリ (Windows の場合) にリダイレクトされます。

重大なエラーが発生した場合は、設定が停止し、ログファイルにエラーが記録されます。ログファイルがすでに存在している場合は、出力が既存のログファイルに追加されます。

処理するエントリ (HADB ノードと Application Server インスタンス) がすでに存在する場合、clsetup コマンドはそれらを削除したり再設定したりしないで、各設定手順は省略されます。この種のイベントは、ログファイルに記録されます。

6. clsetup が設定を完了すると、それぞれの設定を確認したあとでログファイルのスキャンを行います。
7. 完了すると、clsetup により、表 2-11 に記載されている終了コードが返されます。

表 2-11 clsetup コマンドの終了コード

終了コード	説明
0	正常に終了
2	使用法のエラー
3	インスタンスファイルが見つからない
4	インスタンスファイルを読み取れない
5	リソースファイルが見つからない
6	リソースファイルを読み取れない
7	パスワードファイルが見つからない
8	パスワードファイルを読み取れない
10	スクリプトが asadmin を見つけられない
11	スクリプトが hadbm を見つけられない
12	一時ファイルを作成できない
13	セッションストアの設定に失敗した
14	HADB の作成に失敗した
15	HADB が get jdbc URL の取得に失敗した
16	ユーザーが Welcome メッセージで終了する

clsetup のクリーンアップ手順

clsetup を実行したあと、発生したエラーがログファイル `/var/tmp/clsetup.log` に記録されます。clsetup コマンドの実行後に毎回ログファイルを調べ、報告される重大なエラー (例: 存在しないインスタンスの作成に失敗) を修正してください。

設定のすべてまたは一部は、次のようにして元に戻すことができます。

- Application Server インスタンスを削除するには、次のコマンドを実行します。
`asadmin delete-instance instance_name`
- HADB を削除するには、次のコマンドを実行します。
 - a. `hadbm stop database_name`
 - b. `hadbm delete database_name`
- セッションストアをクリアするには、次のコマンドを実行します。cladmin
`clear-session-store --storeurl URL_information --storeuser storeUsername --storepassword store_user_name`
- JDBC 接続プールを削除するには、次のコマンドを実行します。asadmin
`delete-jdbc-connection-pool connectionpool_name`
- JDBC リソースを削除するには、次のコマンドを実行します。cladmin
`delete-jdbc-resource JDBCresource_Name`

これらのコマンドの詳細な例については、マニュアルページを参照してください。HADB の設定と、クラスタ、ロードバランサプラグイン、および HADB の管理を行うには、『Sun Java System Application Server Administration Guide』を参照してください。

時刻の同期

HADB は、内部イベントの管理やデバッグ目的でシステムクロックのタイムスタンプを使用するため、HADB を実行しているホストの時刻を同期させておくことを強くお勧めします。イベントは、先頭にタイプスタンプが付いた履歴ファイルに書き込まれます。HADB は分散システムのため、トラブルシューティングですべての HADB ノードの履歴ファイルがまとめて分析されます。また、HADB は内部でシステムクロックを使用して、タイムアウトのような時間依存のイベントを管理します。

HADB を実行しているシステムのシステムクロックを調整することはお奨めしません。HADB は通常システムクロックを使用するように実装されていますが、次の点に注意する必要があります。

- ホスト上のオペレーティングシステムまたはその他のソフトウェアコンポーネントの問題が、時計が調整されたときにシステム全体の問題を引き起こす場合があります。よく起こるのが、ノードのハングや再起動です。
- 時計を遅らせると、時計が調整されている間 HADB サーバープロセスがハングする場合があります。時計を進ませる場合には、同じ問題は起こりません。

時計を同期させるには、Solaris では "xntpd" (ネットワークタイムプロトコルデーモン)、Linux では "ntpd" を使用できます。

Standard Edition および Enterprise Edition ソフトウェアのアンインストール

この章では、Sun Java System Application Server ソフトウェアをシステムからアンインストールする方法について説明します。

ここでは次の項目について説明します。

- [アンインストールについて](#)
- [Application Server ソフトウェアのアンインストール](#)
- [サイレントモードでのアンインストール \(非対話型\)](#)

アンインストールについて

インストールプログラムは、各コンポーネントに対して指定されているコンポーネントの依存関係を適用します。いったんコンポーネントの依存関係が満たされると、コンポーネントのライフサイクルは独立します。

アンインストールに失敗すると、インストールが完全にロールバックし、製品の再インストールが必要になります。

注	アンインストールが失敗した場合は、新しくインストールを行う前に、残ったファイルやプロセスをクリーンアップする必要があります。『Sun Java System Application Server Troubleshooting Guide』を参照してください。
---	--

すべてのコンポーネントがアンインストールされます。ソフトウェアの部分的なアンインストールはサポートされていません。

アンインストールの要件

Enterprise Edition のアンインストールを成功させるには、次のことが必要です。

- アンインストールを実行する前に、すべてのデータベースを停止して無効にします。

HADB の停止方法に関するガイドラインについては、『Sun Java System Application Server Administration Guide』を参照してください。

- すべてのデータベースホストには、root ユーザーとして SSH または RSH によってアクセスできます。

HADB 通信をこのように設定する手順については、[48 ページの「ホスト通信の設定」](#)を参照してください。

- アンインストールプログラムは、元のインストールホストから実行されます。

Application Server ソフトウェアのアンインストール

アンインストールが実行される前に、アンインストールプログラムにより、Sun Java System Application Server の実行中のプロセスがすべて検出され、停止されます。

注 使用する製品内の JDK がデフォルトとは異なるディレクトリにインストールされている場合は、次のコマンドを実行してください。

```
uninstall -javahome valid_j2se_directory
```

valid_j2se_directory は、使用する JDK のインストール先を示すパスです。

Application Server ソフトウェアをアンインストールするには、次の手順を実行します。

1. Sun Java System Application Server ソフトウェアをアンインストールするマシンでインストールを実行するユーザーと同じユーザーでログインします。
2. Sun Java System Application Server のインストールディレクトリに移動します。
3. インストール方法を選択します。
 - グラフィカルユーザーインターフェースを使用してアンインストールを実行するには、次のコマンドを入力します。 `./uninstall`
 - コマンド行インターフェースによるアンインストールを実行するには、次のコマンドを入力します。 `./uninstall -console`

4. 「アンインストール」をクリックして、アンインストールプロセスを開始します。

詳細リストにログファイルの先頭部分が表示されます。アンインストールに関する詳細な情報は、この詳細リストの末尾に示されているアンインストールログファイルで確認できます。

- Solaris SPARC、x86 および Linux のアンインストールログは次のとおりです。
/var/tmp/Sun_ONE_Application_Server_uninstall.log
- Microsoft Windows のアンインストールログは次のとおりです。
installdir¥uninstall.log
- 低レベルログ
 - ファイルベースの Solaris SPARC および x86 root ユーザー：
/var/sadm/install/logs/Sun_ONE_Application_Server_uninstall.timestamp
 - ファイルベースの Solaris SPARC および x86 root 以外のユーザー：
/var/tmp/Sun_ONE_Application_Server_uninstall.timestamp
 - パッケージベースの Solaris SPARC および x86:
/var/sadm/install/logs/Sun_ONE_Application_Server_uninstall.timestamp
 - ファイルベースの Linux の root ユーザーおよび root 以外のユーザー：
/var/tmp/Sun_ONE_Application_Server_uninstall.timestamp
 - RPM ベースの Linux:
/var/tmp/Sun_ONE_Application_Server_uninstall.timestamp

5. 「閉じる」を選択してアンインストールプログラムを終了します。

6. システムから Application Server コンポーネントが削除されていることを確認して、アンインストールが成功したことを確認します。

サイレントモードでのアンインストール (非対話型)

Application Server ソフトウェアを非対話型サイレントモードでアンインストールするには、次の手順を実行します。

1. Application Server ソフトウェアをアンインストールしたいマシンでインストールを実行するユーザーと同じユーザーでログインします。
2. 次のように入力して、サイレントアンインストールを開始します。`uninstall -silent`

プロンプトに戻ったら、サイレントアンインストールは完了です。
3. システムから Sun Java System Application Server コンポーネントが削除されていることを確認して、アンインストールが成功したことを確認します。
4. アンインストールするサーバーごとに、このプロセスを繰り返します。

Application Server のアップグレード

Sun Java System Application Server インストーラは、Application Server の以前のインストールを現在のバージョンにアップグレードすることができます。

Sun Java System Application Server インストーラで使用可能なアップグレードオプションについては、『Sun Java System Application Server リリースノート』を参照してください。

Application Server のインストールをアップグレードする場合は、次の点に留意してください。

- パッケージベースのインストールにはパッケージベース、ファイルベースにはファイルベース、RPM ベースには RPM ベースのアップグレードのみがサポートされています。
- アップグレードインストールオプションは、グラフィカルインタフェースとコンソールインストールを使用した場合に使用可能で、サイレントモードでのアップグレードはサポートされていません。
- すべての Application Server コンポーネントをまとめて、または個別にアップグレードできます。さらに、追加インストールを使用して各コンポーネントを個別にアップグレードすることもできます。
- Solaris 9 にバンドルされている Application Server 7 Platform Edition がインストールされている場合は、prodreg ユーティリティを使用して Application Server をアンインストールする必要があります。このユーティリティを使用すると、インストールされているバンドルソフトウェアをすべて表示できます。インストールを選択して、それをアンインストールします。この操作により、すべての依存パッケージがマシンと Solaris 製品レジストリから削除されます。

ここでは次の項目について説明します。

- [Standard Edition](#) または [Enterprise Edition](#) のアップグレード

Standard Edition または Enterprise Edition のアップグレード

特に指定しないかぎり、ファイルベースおよびパッケージベースの配布には次の手順が適用されます。

1. インストールプログラムを実行します。
 - a. グラフィカルインタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで `setup` と入力します。
 - b. コマンド行インタフェースを使用するインストールプログラムを実行するには、コマンドプロンプトで `setup -console` と入力します。
2. インストールウィザード画面に従って使用許諾契約に同意し、**Application Server** のインストールディレクトリのパスを指定するか、デフォルトのインストールディレクトリを受け入れます。

デフォルトのインストールディレクトリは、インストールする製品によって異なります。[17 ページの「モデルのパッケージ化とディレクトリ構造」](#)を参照してください。
3. システムに以前のバージョンの **Application Server** がインストールされている場合は、インストールプログラムがそれを検出すると、使用可能なアップグレードオプションが提示され、次のいずれかの操作が必要となります。
 - ファイルベースのインストールでは、新しいディレクトリを選択するか、アップグレードを継続します。
 - パッケージベースおよび RPM ベースのインストール
 - 選択したインストールディレクトリが、以前にインストールされた古いバージョンの **Application Server** と同じ場合は、「継続」を選択します。

選択したインストールディレクトリが、以前にインストールされた古いバージョンの **Application Server** と異なる場合は、ディレクトリを変更してからアップグレードを続ける必要があります。
4. 「コンポーネントの選択」画面で、すでにインストールされているコンポーネントが無効となります。インストールしたい追加コンポーネントを選択します。
5. インストールの残りのプロセスに進みます。

Application Server をアップグレードしたあと、またはコンポーネントを **Enterprise Edition** に追加したあとで、`clsetup` プログラムを実行してクラスタ設定を行う必要があります。

索引

A

AS_ADMIN_PASSWORD, 65
asadmin コマンド, 61

C

cladmin clear-session-store, 76
cladmin コマンド, 59
clinstance.conf ファイル, 64, 65
clpassword.conf ファイル, 65
clresource.conf ファイル, 66 ~ 72
clsetup コマンド, 59
 cleanup プロシージャ, 76
 log, 75
 root 以外の特権に対する設定, 58
 構文, 74
 実行, 74
 終了コード, 75
 入力ファイル, 64
 要件, 62
clsetup の終了コード, 75
clsetup の制限事項, 62
console オプション, 21

E

/etc/opt/SUNWappserver7/config, 18
/etc/ssh/sshd_config, 51

G

gunzip, 23

H

HADB, 63
 clsetup パラメータ, 66
 root 以外の特権に対する設定, 58
 環境の設定, 57
 共有メモリの設定, 45
 通信設定, 48
 リモートアクセスの設定, 48
HADB_ROOT の PATH 設定, 57
hadbm, 33, 61
hadbm delete, 76
hadbm マニュアルページ, 20
HADB 管理クライアント, 57
HADB の通信設定, 48
HADB のリモートアクセス, 48

I

inetd 設定 , 63

J

J2SE

 サードパーティ , 19, 32

JAVA_HOME 設定 , 57

JDBC 接続プール , 63

JDBC 接続プールパラメータ , 69

O

OpenSSH, 48, 51

/opt/SUNWappserver7, 17

R

.rhosts ファイル , 49

root 以外の特権に対する設定 , 58, 63

rpm, 14

RSH, 48

S

savestate オプション , 22

setenv JAVA_HOME, 57

setenv PATH, 57

set shmsys, 46

setup コマンドのオプション , 21

setup コマンドの構文 , 21

setup コマンドの使用方法 , 21

showrev, 13

Solaris 8, 48, 50

Solaris 9, 48

SSH, 48

ssh-keygen, 54

statefile, 22

sunfreeware, 52

Sun カスタマサポート , 13

T

tar, 23

V

/var/opt/SUNWappserver7/domains, 18

W

Web サーバー

 高可用性の要件 , 30, 34, 35

あ

アンインストール , 79 ~ 82

 要件 , 80

い

依存関係 , 18

インストール , 15 ~ 41

 サイレントモードの設定ファイル , 37

 追加 , 18

 ディレクトリ構造 , 17

 評価 , 18

か

環境変数 HADB の設定 , 57

管理クライアント , 19

き

共有メモリ、HADB の設定 , 45

こ

高可用性 , 43

概要 , 20

ホスト通信の設定 , 48

さ

サードパーティの J2SE, 19, 32

サーバーの起動 , 41

サイレントインストール , 37 ~ 41

サイレントオプション , 21

サイレントモード , 82

せ

セキュアシェル (SSH) の設定 , 48

セッションストアパラメータ , 67

セッションの持続性パラメータ , 70

設定ファイル (サイレントモード), 37

変更 , 38

変数 , 38

セマフォ , 45

つ

追加インストール , 18

は

パッケージ化 , 17

パッケージベースのモデル , 17

パラメータ駆動型インストール , 37

ひ

評価インストール , 18

ふ

部分インストール , 18

プラグイン

ロードバランサ , 20

フリーウェア , 52

へ

ヘルプオプション , 22

ま

マニュアル , 14

一般的な表記規則 , 11

マニュアルページ , 14, 20

よ

要件

clsetup, [62](#)

り

リモートシェル (RSH) の設定, [48](#)

ろ

ロードバランサプラグイン, [20](#)

インストール, [33](#), [35](#) ~ [36](#)

ログ

clsetup, [75](#)