

管理者用設定ファイルリファレンス

Sun™ ONE Application Server

Version 7

816-6480-10
2002年9月

Copyright © 2002 Sun Microsystems, Inc., 4150 Network Circle, Santa Clara, California 95054, U.S.A. All rights reserved.

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 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 および Sun ONE のロゴマークは、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。

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。

目次

本書について	9
対象読者	9
マニュアルの使用方法	10
マニュアルの構成	12
マニュアルの表記規則	13
一般的な表記規則	13
ディレクトリ名の表記規則	14
製品サポート	15
第1章 サーバー動作の基本	17
設定ファイル	17
server.xml	18
init.conf	18
obj.conf	18
mime.types	19
動的再設定	19
設定ファイルの手動編集	20
第2章 サーバー設定ファイル	23
sun-server_1_0.dtd ファイル	24
サブ要素	24
データ	25
属性	25
server.xml ファイル内の要素	26
全般的な要素	27
server	27
property	28

description	29
admin-service	30
server-instance	30
リスナーサービス要素	30
http-service	30
http-listener	32
ssl	34
mime	36
acl	37
virtual-server-class	37
virtual-server	39
http-qos	43
auth-db	44
iiop-service	45
orb	45
ssl-client-config	46
iiop-listener	47
コンテナ要素	48
web-container	48
ejb-container	49
mdb-container	53
J2EE サービス要素	55
jms-service	55
log-service	57
security-service	59
auth-realm	60
transaction-service	63
Java 設定要素	65
java-config	65
profiler	67
jvm-options	68
リソース要素	69
resources	69
custom-resource	70
external-jndi-resource	71
jdbc-resource	72
mail-resource	73
jms-resource	75
persistence-manager-factory-resource	77
jdbc-connection-pool	78
アプリケーション要素	82
applications	82
lifecycle-module	83

j2ee-application	84
ejb-module	85
web-module	86
connector-module	87
ユーザーデータベースの選択	88
Sun ONE LDAP スキーマ	89
収束ツリー	89
ドメインコンポーネント (dc) ツリー	90
変数	91
変数の形式	91
id 変数	91
その他の重要な変数	92
変数の評価	93
server.xml ファイルのサンプル	93
第 3 章 init.conf の構文と使い方	99
Init 関数	101
サーバー情報	101
NetSiteRoot	101
TempDir	101
TempDirSecurity	102
User (UNIX のみ)	102
DNS ルックアップ	103
AsyncDNS	103
DNS	103
スレッド、プロセス、および接続	104
ConnQueueSize	105
HeaderBufferSize	105
IOTimeout	105
KeepAliveThreads	106
KeepAliveTimeout	106
KernelThreads (Windows のみ)	106
ListenQ	107
MaxKeepAliveConnections	107
PostThreadsEarly	107
RcvBufSize	107
RqThrottle	107
RqThrottleMin	108
SndBufSize	108
StackSize	108
StrictHttpHeaders	108
TerminateTimeout	108
ThreadIncrement	109

ネイティブのスレッドプール	109
NativePoolStackSize	110
NativePoolMaxThreads	110
NativePoolMinThreads	110
NativePoolQueueSize	110
CGI	111
CGIExpirationTimeout	111
CGIStubIdleTimeout	111
MaxCGIStubs	111
MinCGIStubs	112
WingcgiTimeout (Windows のみ)	112
エラー記録	113
ErrorLogDateFormat	113
LogFlushInterval	113
PidLog	113
ACL	114
ACLCacheLifetime	114
ACLUserCacheSize	114
ACLGroupCacheSize	115
セキュリティ	115
Security	115
SSLCacheEntries	116
SSLClientAuthDataLimit	116
SSLClientAuthTimeout	116
SSLSessionTimeout	116
SSL3SessionTimeout	117
チャンクされたエンコード	117
UseOutputStreamSize	117
ChunkedRequestBufferSize	118
ChunkedRequestTimeout	118
その他	119
ChildRestartCallback	119
HTTPVersion	119
MaxRqHeaders	119
ReentrantTimeFunctions (Solaris のみ)	120
Umask (UNIX のみ)	120
第4章 MIME タイプ	121
はじめに	122
MIME タイプの決定	122
Type の応答への影響	123
クライアントの行う MIME タイプの処理	124
MIME タイプファイルの構文	124

MIME タイプファイルの例	124
第 5 章 その他の設定ファイル	127
dbswitch.conf	127
配備記述子	130
generated.instance.acl	131
nsfc.conf	131
password.conf	133
server.policy	134
付録 A 日時の形式	135
付録 B サーバー設定要素のアルファベット順リスト	137
付録 C init.conf の指令のアルファベット順リスト	141
索引	145

本書について

このマニュアルでは、`server.xml`、`init.conf`、`mime.types` などの Sun™ Open Net Environment (Sun ONE) Application Server 7 の設定ファイルの目的と使い方について説明し、これらの設定ファイルで使用できる要素や指令の包括的なリストを提供します。

この章には次の項目があります。

- 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
- 製品サポート

対象読者

このマニュアルは、Sun ONE Application Server の管理および保守の担当者を対象にしています。

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。

- J2EE 仕様
- HTTP
- HTML
- XML
- Java プログラミング
- サーブレット、JSP、EJB、および JDBC の仕様に定義されている Java API
- リレーショナルデータベースの概念

マニュアルの使用法

このマニュアルは、PDF 形式または HTML 形式でも入手できます。次のサイトを参照してください。

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

次の表は、Sun ONE 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。左側の列にタスクと概念、右側の列に参照するマニュアルを示します。

Sun ONE Application Server マニュアルの概要

情報の内容	参照するマニュアル
ソフトウェアおよびマニュアルの最新情報	リリースノート
サポート対象のプラットフォームと環境	プラットフォーム
アプリケーションサーバーの紹介。アプリケーションサーバーの新機能、評価用バージョンのインストール、アーキテクチャの概要など	入門ガイド
Sun ONE Application Server 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、Sun ONE Message Queue など) のインストール	インストールガイド
Sun ONE Application Server 7 の Java オープンスタンダードモデルに準拠した J2EE アプリケーションの作成方法と実装方法。アプリケーションの設計、開発ツール、セキュリティ、アセンブリ、配備、デバッグ、ライフサイクルモジュールの作成に関する情報など	開発者ガイド
Sun ONE Application Server 7 の Web アプリケーション向け Java オープンスタンダードモデルに準拠した J2EE アプリケーションの作成方法と実装方法。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	Web アプリケーション開発者ガイド
Sun ONE Application Server 7 のエンタープライズ Bean 向け Java オープンスタンダードモデルに準拠した J2EE アプリケーションの作成方法と実装方法。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	Enterprise JavaBeans 開発者ガイド
Sun ONE Application Server 7 上の J2EE アプリケーションにアクセスするクライアントの作成方法	Developer's Guide to Clients
Web サービスの作成方法	Developer's Guide to Web Services

Sun ONE Application Server マニュアルの概要 (続き)

情報の内容	参照するマニュアル
JDBC、JNDI、JTS、JMS、JavaMial、リソース、コネクタなどの J2EE 機能	Developer's Guide to J2EE Features and Services
カスタム NSAPI プラグインの作成方法	NSAPI Developer's Guide
次の管理タスクの実行	管理者ガイド
<ul style="list-style-type: none"> • 管理インタフェースとコマンド行インタフェースの使用 • サーバーの作業環境の設定 • 管理ドメインの使用 • サーバーインスタンスの使用 • サーバーの稼動状況の監視およびログ記録 • Web サーバープラグインの設定 • Java Messaging Service の設定 • J2EE 機能の使用 • CORBA ベースのクライアント機能の設定 • データベース接続性の設定 • トランザクション管理の設定 • Web コンテナの設定 • アプリケーションの配備 • 仮想サーバーの管理 	
サーバー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 ONE Application Server 7 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がある。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 ONE Application Server 7 用の J2EE CA 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、DTD に関する情報やサンプル XML ファイルなど	J2EE CA Service Provider Implementation Administrator's Guide
Netscape Application Server バージョン 2.1 から新しい Sun ONE Application Server 7 プログラミングモデルへのアプリケーションの移行。Sun ONE Application Server に付属するオンラインバンクアプリケーションの移行サンプルなど	サーバーアプリケーションの移行および再配備

Sun ONE Application Server マニュアルの概要 (続き)

情報の内容	参照するマニュアル
Sun ONE Message Queue の使用法	Sun ONE Message Queue については次の URL を参照: http://docs.sun.com/?p=/coll/S1_MessageQueue_30

マニュアルの構成

このマニュアルには、次の章と付録があります。

- 第 1 章「サーバー動作の基本」
この章では、Sun ONE Application Server を制御する主な設定ファイルを紹介し、設定ファイルを編集し有効にする方法について説明します。
- 第 2 章「サーバー設定ファイル」
この章では、サーバー動作のほとんどを制御する `server.xml` ファイルについて説明します。
- 第 3 章「`init.conf` の構文と使い方」
この章では、`init.conf` ファイルに指令を設定して、初期化時に Sun ONE Application Server を設定する方法について説明します。
- 第 4 章「MIME タイプ」
この章では、ファイル拡張子をファイルタイプにマップする MIME タイプファイルについて説明します。
- 第 5 章「その他の設定ファイル」
この章では、その他の重要な設定ファイルを一覧で表示し、その内容が一目で分かるようにしています。
- 付録 A「日時の形式」
この付録では、サーバーログで日時を表すのに使われる形式文字列について説明します。
- 付録 B「サーバー設定要素のアルファベット順リスト」
付録 C「`init.conf` の指令のアルファベット順リスト」
これらの付録には、`server.xml` の要素と `init.conf` の指令のアルファベット順のリストがあり、簡単に検索できます。

マニュアルの表記規則

この節では、このマニュアルで使用する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 (ディレクトリ名をスラッシュ記号で区切って表記)。Windows バージョンでは、ディレクトリパスについては UNIX と同じですが、ディレクトリの区切り記号にはスラッシュ記号ではなく円記号を使用します。

- **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、サンプルコード、コードのリスト、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 タグに使います。
 - 斜体文字はコード変数に使います。
 - 斜体文字は、マニュアルのタイトル、強調、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います。
 - **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います。
- このマニュアルでは、ほとんどのプラットフォームの**インストールルートディレクトリ**を *install_dir* と記述します。例外については、14 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。

デフォルトでは、ほとんどのプラットフォームの *install_dir* は次の位置になります。

- Solaris 8 のパッケージベースでない評価インストール：


```
ユーザーのホームディレクトリ /sun/appserver7
```
- Solaris にアンバンドルの、非評価インストール：

/opt/SUNWappserver7

- Windows のインストール :

C:%Sun¥AppServer7

上記の *default_config_dir* と *install_config_dir* は、*install_dir* と同義です。例外と追加情報については、14 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。これは以下のパスの省略形式です。

default_config_dir/domains/domain/instance

- このマニュアルを通じて、特に明記のない限り、すべての **UNIX 固有の表記**は、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

デフォルトでは、Solaris 8 および 9 パッケージベースのインストールおよび Solaris 9 バンドル版のインストールを使用すると、アプリケーションサーバーファイルはいくつかのルートディレクトリに分散してインストールされます。この節では、これらのディレクトリについて説明します。

- **Solaris 9 バンドル版のインストール**のデフォルトのインストールディレクトリは、次のように表記します。
 - *install_dir*: /usr/appserver/ を表します。このディレクトリには、インストールイメージの静的な部分が格納されます。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、およびライブラリがここに格納されます。
 - *default_config_dir*: 作成されたドメインのデフォルトの格納先となる /var/appserver/domains ディレクトリを表します。
 - *install_config_dir*: /etc/appserver/config ディレクトリを表します。このディレクトリには、インストール全体の設定情報が格納されます。たとえば、このインストールのライセンス、管理ドメインのマスターリストなどが格納されます。
- **Solaris 8 および 9 パッケージベースの、バンドルされていない評価版以外のインストール**ではデフォルトのディレクトリは、次のように表記します。
 - *install_dir*: /opt/SUNWappserver7 ディレクトリを表します。このディレクトリには、インストールイメージの静的な部分が格納されます。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、およびライブラリがここに格納されます。

- *default_config_dir*: 作成されたドメインの格納先となる
/var/opt/SUNWappserver7/domains ディレクトリを表します。
- *install_config_dir*: /etc/opt/SUNWappserver7/config ディレクトリを表します。このディレクトリには、インストール全体の設定情報が格納されます。たとえば、このインストールのライセンス、管理ドメインのマスターリストなどが格納されます。

製品サポート

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。

事前に、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
- 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

サーバー動作の基本

Sun ONE Application Server の設定および動作は設定ファイルで決まります。管理インタフェースを使用すると、これらの設定ファイルの設定を変更できます。これらのファイルは手動で編集することもできます。

この章には次の節があります。

- 設定ファイル
- 動的再設定
- 設定ファイルの手動編集

設定ファイル

Sun ONE Application Server の設定および動作は設定ファイルによって制御されます。設定ファイルは、*instance_dir*/config/ ディレクトリにあります。このディレクトリには、異なるコンポーネントを制御するためのさまざまな設定ファイルがあります。設定ファイルの実際の数と名前は、サーバーで有効になっているコンポーネントまたはサーバーに読み込まれたコンポーネントによって異なります。

ただし、このディレクトリには、サーバーの動作に不可欠な 4 つの設定ファイルが常に含まれています。これらのファイルを次に示します。

- `server.xml` -- ほとんどのサーバー設定情報が含まれている
- `init.conf` -- グローバルサーバー初期化情報が含まれている
- `obj.conf` -- クライアントからの HTTP 要求を処理する命令が含まれている
- `mime.types` -- 要求されたリソースのコンテンツタイプを決定する情報が含まれている

その他の重要な設定ファイルについては、第 5 章「その他の設定ファイル」を参照してください。

server.xml

このファイルには、ほとんどのサーバー設定情報が含まれています。スキーマファイル `sun-server_1_0.dtd` は、`server.xml` ファイルの書式と内容を定義します。

サーバーがどのように `sun-server_1_0.dtd` および `server.xml` を使用するかについての詳細は、第2章「サーバー設定ファイル」を参照してください。

init.conf

このファイルでは、初期化時にサーバーを設定する変数の値を設定します。サーバーはこのファイルを参照し、起動時に設定を実行します。サーバーは、再起動されるまでこのファイルを再び参照することはありません。

`init.conf` で設定できるすべての変数および `Init` 指令のリストについては、第3章「`init.conf` の構文と使い方」を参照してください。

obj.conf

このファイルには、クライアントからの HTTP 要求の処理方法、またネイティブサーバープラグインや CGI プログラムなどの Web サーバーのコンテンツの提供方法をサーバーに示す指示が含まれています。サーバーは、クライアントからの要求を処理するときは必ず、このファイルで定義されている設定を参照します。

すべての `obj.conf` ファイルは、`instance_dir/config/` ディレクトリにあります。複数の仮想サーバーが `obj.conf` ファイルを共有するように設定されている場合を除き、仮想サーバーごとに1つの `obj.conf` ファイルがあります。このマニュアルで示す「`obj.conf` ファイル」とは常に、すべての `obj.conf` ファイル、または説明中の仮想サーバーの `obj.conf` ファイルのことを指します。

プレフィックスがない `obj.conf` というファイルは、Sun ONE Application Server が各仮想サーバーの `obj.conf` ファイルを作成するために使用するテンプレートです。このファイルを編集しても、既存の仮想サーバーは影響を受けませんが、その後で作成されるサーバーは影響を受けます。

デフォルトでは、有効な各 `obj.conf` ファイルには、`virtual_server_name-obj.conf` という名前が付けられます。サーバーインスタンスのデフォルトの仮想サーバーの名前は、そのインスタンスの名前であるため、最初にサーバーインスタンスを作成するときには、その `obj.conf` ファイルの名前は `instance_name-obj.conf` になります。これらのファイルを直接または管理インタフェースから編集すると、仮想サーバーの設定が変わります。

obj.conf ファイルは、Sun ONE Application Server の動作に欠かせない重要なものです。管理インターフェースを使用してサーバーに変更を加えると、システムは obj.conf を自動的に更新します。

obj.conf ファイルには、要求応答プロセスの各段階でどのような処理を行うかを Sun ONE Application Server に指示する、一連の命令 (指令) が含まれています。

サーバーがどのように obj.conf を使用するかについては、『Sun ONE Application Server Developer's Guide to NSAPI』を参照してください。

mime.types

このファイルは、ファイル拡張子を MIME タイプにマップして、要求されたリソースのコンテンツタイプをサーバーが判別できるようにします。たとえば、.html 拡張子を持つリソースの要求は、クライアントが HTML ファイルを要求していることを意味し、.gif 拡張子を持つリソースの要求は、クライアントが GIF フォーマットのイメージファイルを要求していることを意味します。

サーバーがどのように mime.types を使用するかについては、第 4 章「MIME タイプ」を参照してください。

動的再設定

次のファイルに加えた変更を有効にするためにサーバーを再起動する必要はありません。

- obj.conf
- mime.types
- ACL ファイル
- server.xml (applications および resources サブ要素のみ)

ただし、配備は例外であり、変更を有効にするには変更を適用する必要があります。次のように管理インターフェースを使います。

1. サーバーインスタンスのページに移動します。
2. 「一般」タブを選択します。
3. 「変更の適用」ボタンを選択します。

または、次のようにコマンド行を使います。

```
asadmin reconfig --user user [--password password] [--passwordfile password_file]  
[--host machine] [--port port] [--discardmanualchanges=false |  
--keepmanualchanges=false] instance_name
```

次に例を示します。

```
asadmin reconfig --user joeuser --password secret --keepmanualchanges=true  
server1
```

設定を変更してその変更を適用すると、上記に示した動的に設定可能なファイル内のすべての情報を含む新しい設定がメモリーに読み込まれます。

注 設定の変更を適用すると、サーバーインスタンスがエラーメッセージを表示する場合があります。たとえば、セキュリティがオンになっているときに変更を適用すると、設定が正しくない旨のエラーが表示されることがあります。

このエラーメッセージが表示された場合は、「インスタンスの再起動が必要です」または「サーバーの再起動が必要です」などの再起動メッセージが表示されたらただちにインスタンスを再起動します。再起動が遅れると、同じエラーメッセージが繰り返し表示されることがあります。

設定ファイルの手動編集

設定ファイルを手動で編集することはお勧めできませんが、ここでは、手動で編集する際の注意事項をいくつか挙げます。

- 変更が必要な場合には、*instance_dir/config/* ディレクトリにある設定ファイルのみを編集します。ファイルをバックアップディレクトリにコピーしたり、タイムスタンプファイルを削除したりしないでください。手動による変更の上書きを妨げることがあります。
- 手動による変更と管理インタフェースによる変更の両方を行い、まだ変更を適用していない場合は、次のいずれかを行う必要があります。
 - 管理インタフェースを使用して変更を適用します。この場合、手動の変更と管理インタフェースによる変更のどちらかを選択する必要があります。
 - `asadmin reconfig` コマンドに `--keepmanualchanges` または `--discardmanualchanges` オプションを指定して、コマンド行で同じ選択を行います。次に例を示します。

```
asadmin reconfig --user joeuser --password secret --keepmanualchanges=true  
server1
```

このような状況を避けるには、常に変更の直後に変更を適用し、手動の変更と管理インタフェースによる変更を同時に適用しないようにします。

- ファイルを手動で編集してからサーバーを再起動しないでください。サーバーを再起動する前に `asadmin reconfig` コマンドに `--keepmanualchanges` オプションを指定して実行します。上の例を参照してください。
- 管理インタフェースを使用して `server.xml` の設定を変更するたびに、`sun-server_1_0.dtd` ファイルと比較して検証されます。ファイルの相互参照のチェックに関する追加の検証が行われます。設定が完全で正しいことを確認してから、変更を適用してください。
- `server.xml` の `applications` または `resources` サブ要素を変更する（たとえば、J2EE アプリケーションを配備したり JDBC 接続プールを登録したりする）と、管理サーバーは実行しているサーバーインスタンスへの変更を適用しようとしません。設定が完全で正しいことを確認してから、変更を適用してください。
- 管理サーバーは、再起動前の適用されていない変更を記憶しています。このため、管理サーバーが認識しているサーバーインスタンスの設定が実際のサーバーインスタンスの設定とは異なる場合があります。

サーバー設定ファイル

`server.xml` ファイルには、ほとんどのサーバー設定情報が含まれています。一般的な UNIX テキストエディタとの互換性を維持するため、エンコーディングは UTF-8 になっています。`server.xml` ファイルは、`instance_dir/config` ディレクトリにあります。スキーマファイル `sun-server_1_0.dtd` は、`server.xml` ファイルの書式と内容を定義します。

この章には、`server.xml` および `sun-server_1_0.dtd` ファイルについて説明する次の節があります。

- `sun-server_1_0.dtd` ファイル
- `server.xml` ファイル内の要素
- 一般的な要素
- リスナーサービス要素
- コンテナ要素
- J2EE サービス要素
- Java 設定要素
- リソース要素
- アプリケーション要素
- ユーザーデータベースの選択
- Sun ONE LDAP スキーマ
- 変数
- `server.xml` ファイルのサンプル

注 仮想サーバーとサーバーインスタンスは別のものです。各サーバーインスタンスは、1 つ以上の仮想サーバーを含み、完全に個別のサーバーです。

sun-server_1_0.dtd ファイル

sun-server_1_0.dtd ファイルは、server.xml ファイルの構造を定義します。つまり、server.xml に記述できるさまざまな要素と、その要素が持つことのできる属性を定義しています。sun-server_1_0.dtd ファイルは、install_dir/lib/dtds ディレクトリにあります。

注 sun-server_1_0.dtd ファイルを編集しないでください。このファイルの内容は、Sun ONE Application Server のバージョンの改訂にもなって変更されます。

注 sun-server_1_0.dtd インタフェースは、不確定です。不確定なインタフェース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。

DTD ファイルおよび XML の全般的な情報については、次のサイトにある XML 仕様書を参照してください。

<http://www.w3.org/TR/REC-xml>

DTD ファイル (対応する XML ファイルにある場合がある) に定義された各要素には、次のものを含めることができます。

- サブ要素
- データ
- 属性

サブ要素

要素にはサブ要素を含めることができます。たとえば、次のファイルコードは iiop-listener 要素を定義しています。

```
<!ELEMENT iiop-listener (ssl?, property*)>
```

この ELEMENT タグは、iiop-listener 要素に ssl と property 要素をこの順番で含めることができることを示しています。

次の表に、サブ要素のオプションのサフィックス文字によって決まる必要指定数、つまり指定可能なサブ要素の数を示します。左の列にはサブ要素の終了文字、右の列には対応する必要指定数を示しています。

必要指定数とサブ要素のサフィックス

サブ要素のサフィックス	必要指定数
要素 *	このサブ要素を含まないか、1 個以上含めることができる
要素 ?	このサブ要素を含まないか、1 個含めることができる
要素 +	このサブ要素を 1 個以上含まなければならない
要素 (サフィックスなし)	このサブ要素を 1 個だけ含まなければならない

要素にほかの要素を含めることができない場合は、カッコで囲まれたサブ要素名のリストの代わりに、EMPTY または (#PCDATA) が表示されます。

データ

要素の中には、サブ要素の代わりに文字データを含むものもあります。これらの要素は、次の形式で定義されます。

```
<!ELEMENT element-name (#PCDATA) >
```

次に例を示します。

```
<!ELEMENT description (#PCDATA) >
```

server.xml ファイル内では、データ要素内の空白スペースはデータの一部として扱われます。そのため、データ要素で区切られたデータの前後には余分な空白がないようにする必要があります。次に例を示します。

```
<description>shopping cart bean</description>
```

属性

ATTLIST タグが付いた要素には、属性 (名前と値のペア) が含まれます。次に例を示します。

```
<!ATTLIST iiop-listener
      id          CDATA          #REQUIRED
      address     CDATA          #REQUIRED
      port        CDATA          "3700"
      enabled     %boolean;     "true">
```

iiop-listener 要素には、id、address、port、または enabled 属性を含めることができます。

REQUIRED ラベルは、値を指定する必要があることを示します。# IMPLIED ラベルは、その属性の指定は省略可能であり、Sun ONE Application Server がデフォルト値を設定することを示します。可能な場合は、オプションの属性の明示的なデフォルト値 ("true" など) が示されます。

属性の宣言では、属性のタイプを指定します。たとえば、CDATA は文字データを、%boolean は事前に定義された列挙型を示します。

server.xml ファイル内の要素

この節では、server.xml ファイル内の XML 要素について説明します。要素は次のグループに分類されます。

- 全般的な要素
- リスナーサービス要素
- コンテナ要素
- J2EE サービス要素
- Java 設定要素
- リソース要素
- アプリケーション要素

注 特に指定しない限り、**サブ要素**は各表の要素欄に示されている順に定義する必要があります。

server.xml の要素のアルファベット順のリストについては、付録 B 「サーバー設定要素のアルファベット順リスト」を参照してください。

全般的な要素

全般的な要素は次のとおりです。

- `server`
- `property`
- `description`
- `admin-service`
- `server-instance`

server

サーバーを定義します。これはルート要素であり、`server.xml` ファイル内には `server` 要素が 1 つだけ存在します。

サブ要素

次の表は、`server` 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示します。

`server` のサブ要素

要素	必要指定数	説明
<code>http-service</code>	1 個のみ	HTTP サービスを定義する
<code>iiop-service</code>	1 個のみ	IIOP サービスを定義する
<code>admin-service</code>	0 または 1 個	実装されていない
<code>web-container</code>	1 個のみ	Web コンテナを設定する
<code>ejb-container</code>	1 個のみ	EJB コンテナを設定する
<code>mdb-container</code>	1 個のみ	メッセージ駆動型 Bean (MDB) コンテナを設定する
<code>jms-service</code>	1 個のみ	Java Message Service プロバイダを設定する
<code>log-service</code>	1 個のみ	システムログサービスを設定する
<code>security-service</code>	1 個のみ	J2EE セキュリティサービスで必要となる情報を定義する
<code>transaction-service</code>	1 個のみ	Java Transaction Service を設定する
<code>java-config</code>	1 個のみ	JVM 設定を含む
<code>resources</code>	1 個のみ	設定されたリソースを含む

server のサブ要素

要素	必要指定数	説明
applications	1 個のみ	配備された J2EE アプリケーション、J2EE モジュール、およびライフサイクルモジュールを含む
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、server 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

server の属性

属性	デフォルト値	説明
name	なし	サーバーインスタンスの名前を指定する
locale	オペレーティングシステムのデフォルト	(省略可能) サーバーインスタンスの言語を指定する
log-root	<i>instance_dir</i> /logs	(省略可能) サーバーインスタンスのログファイルを保存する位置を指定する。サーバーログが保存されるディレクトリには、サーバーで実行されているどのユーザーアカウントも書き込める必要がある。ログについての詳細は、「log-service」の説明を参照
application-root	<i>instance_dir</i> /applications	(省略可能) このサーバーインスタンス内に配備されたアプリケーションの保存されている場所への絶対パスを指定する
session-store	<i>instance_dir</i> /session-store	(省略可能) 非活性化された Beans が格納される、ファイルシステム内のディレクトリを指定する

property

プロパティ (server.xml 内で定義されて、obj.conf 内で参照される変数) を指定します。変数については、91 ページの「変数」を参照してください。

property は、次のどちらかあるいは両方である親要素に設定情報を追加します。

- Sun ONE Application Server では省略可能な要素

- Sun ONE Application Server が認識していない、LDAP サーバーや Java クラスなどのシステムまたはオブジェクトに必要な要素

たとえば、auth-realm 要素には、property サブ要素を含めることができます。

```
<auth-realm name="file"
  classname="com.iplanet.ias.security.auth.realm.file.FileRealm">
  <property name="file" value="instance_dir/config/keyfile"/>
  <property name="jaas-context" value="fileRealm"/>
</auth-realm>
```

auth-realm 要素が使用するプロパティは、auth-realm 要素の name 属性によって決まります。file レルムは、file および jaas-context プロパティを使用します。その他のレルムは、別のプロパティを使用します。

サブ要素

次の表は、property 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

property のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む

属性

次の表は、property 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

property の属性

属性	デフォルト値	説明
name	なし	プロパティまたは変数の名前を指定する
value	なし	プロパティまたは変数の値を指定する

description

親の要素を説明するテキストを含む

サブ要素

なし

属性
なし

admin-service

この要素は実装されていないので、使用できません。

server-instance

この要素は実装されていないので、使用できません。

リスナーサービス要素

リスナーサービス要素は次のとおりです。

- `http-service`
- `http-listener`
- `ssl`
- `mime`
- `acl`
- `virtual-server-class`
- `virtual-server`
- `http-qos`
- `auth-db`
- `iiop-service`
- `iiop-listener`
- `orb`

http-service

HTTP サービスを定義します。

この要素の属性に定義されるサービス品質機能についての詳細は、『Sun ONE Application Server 7 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。

サブ要素

次の表は、`http-service` 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

http-service のサブ要素

要素	必要指定数	説明
http-listener	0 または 1 個以上	HTTP 待機ソケットを定義する
mime	0 または 1 個以上	MIME タイプを定義する
acl	0 または 1 個以上	ACL ファイルを参照する
virtual-server-class	0 または 1 個以上	仮想サーバークラスを定義する
http-qos	0 または 1 個	サービス品質パラメータを定義する
property	0 または 1 個以上	プロパティまたは変数を指定する

注 http-listener、mime、acl、および virtual-server-class 要素はどのような順番で記述してもかまいませんが、http-qos および property 要素はそれぞれ最後から 2 番目と最後に記述する必要があります。

属性

次の表は、http-service 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

http-service の属性

属性	デフォルト値	説明
qos-metrics-interval-in-seconds	30	(省略可能) トラフィックを測定する間隔を指定する
qos-recompute-time-interval-in-millis	100	(省略可能) すべてのサーバーエンティティの帯域幅を再計算する間隔を指定する
qos-enabled	true	(省略可能) サービス品質機能を有効にして、サーバーエンティティに制限を設定したり、帯域幅や接続に関するサーバー統計を表示したりする。有効な値は yes、no、on、off、1、0、true、false

http-listener

HTTP 待機ソケットを定義します。

注 管理インタフェースを使用して安全なリスナーを作成する場合、セキュリティは自動的に `init.conf` 内でグローバルにオンになります。
`sever.xml` で安全なリスナーを手動で作成する場合は、`init.conf` ファイルの `Security` 指令を編集してセキュリティをオンにする必要があります。

サブ要素

次の表は、`http-listener` 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

`http-listener` のサブ要素

要素	必要指定数	説明
<code>ssl</code>	0 または 1 個	SSL パラメータを定義する

属性

次の表は、`http-listener` 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

`http-listener` の属性

属性	デフォルト値	説明
<code>id</code>	なし	一意のリスナー名。 <code>http-listener</code> の名前は数字で始めることはできない
<code>address</code>	なし	リスナーの IP アドレス。ドットで区切ったペア、または IPv6 表記で指定する。すべての IP アドレスを待機する <code>any (INADDR_ANY の)</code> を指定できる。ホスト名も指定できる
<code>port</code>	なし	リスナーが待機するポート番号。有効な値は 1 ~ 65535。UNIX の場合、ポート 1 ~ 1024 で待機するソケットを作成するには、スーパーユーザー権限が必要。SSL リスナーをポート 443 で待機するように設定することを推奨

http-listener の属性 (続き)

属性	デフォルト値	説明
family	inet	(省略可能) ソケットファミリのタイプ。有効な値は <code>inet</code> 、 <code>inet6</code> 、および <code>nca</code> 。IPv6 リスナーの場合は値 <code>inet6</code> を使用。この値が <code>inet6</code> の場合、サーバーログでは、IPv4 アドレスの先頭に <code>::ffff:</code> が付く。Solaris のネットワークキャッシュおよび Accelerator を使う場合は <code>nca</code> を指定する
acceptor-threads	1	(省略可能) リスナーの受け入れ側スレッドの数。推奨値は、マシンのプロセッサ数。有効な値は 1 ~ 1024
blocking-enabled	false	(省略可能) リスナーおよび受け入れたソケットをブロックモードにするかどうかを決定する。ブロックモードを使うと、ベンチマーク結果が向上することがある。有効な値は <code>on</code> 、 <code>off</code> 、 <code>yes</code> 、 <code>no</code> 、 <code>1</code> 、 <code>0</code> 、 <code>true</code> 、 <code>false</code>
security-enabled	false	(省略可能) リスナーが SSL を実行するかどうかを決定する。有効な値は <code>on</code> 、 <code>off</code> 、 <code>yes</code> 、 <code>no</code> 、 <code>1</code> 、 <code>0</code> 、 <code>true</code> 、 <code>false</code> 。要素を使って SSL2 または SSL3 をオンまたはオフにしたり、暗号化方式を設定できる init.conf ファイル内の Security 設定は、サーバーインスタンスで証明書を利用できるようにすることで、SSL をグローバルに有効または無効にする。そのため、init.conf の Security が on になっていることが必要。そうでないと、server.xml の security-enabled は機能しない。詳細については、第 3 章「init.conf の構文と使い方」を参照
default-virtual-server	なし	この特定のリスナー用の、デフォルトの仮想サーバーの id 属性
server-name	なし	クライアントに送信する URL のホスト名セクションに指定する名前を、サーバーに指示する。これは、サーバーが自動的に生成する URL には影響するが、サーバーに格納されているディレクトリやファイルの URL には影響しない。サーバーがエイリアスを使っている場合、この名前はエイリアス名であることが必要 コロンとポート番号を付加すると、サーバーがクライアントに送信する URL でそのポートが使われる

http-listener の属性 (続き)

属性	デフォルト値	説明
enabled	true	(省略可能) リスナーをアクティブにするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

警告 ブロックモードソケットは、実際の配備では使わないようにしてください。ブロックモードソケットを使うと動的な再設定ができなくなり、サーバーがサービスを利用できなくなることがあります。

ssl

SSL (Secure Socket Layer) パラメータを定義します。

http-listener 要素の security-enabled 属性が on に設定されている場合は、ssl 要素が必要です。

ssl 要素は、http-listener または iiop-listener 要素内でのみ使用できます。

サブ要素

なし

属性

次の表は、ssl 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

ssl の属性

属性	デフォルト値	説明
cert-nickname	なし	証明書データベースまたは PKCS#11 トークン内でのサーバー証明書のニックネーム。証明書では、名前の形式は <i>tokennamenickname</i> 。この属性では、名前の <i>tokenname:</i> の部分は省略可能
ssl2-enabled	false	(省略可能) SSL2 を有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false SSL2 と SSL3 の両方が仮想サーバーで有効になっている場合、サーバーは初めに SSL3 暗号化方式を試行する。SSL3 暗号化方式が失敗すると、SSL2 暗号化方式を試行する

ssl の属性 (続き)

属性	デフォルト値	説明
ssl2-ciphers	なし	(省略可能) SSL2 暗号化方式のコンマで区切られたリスト。有効にするにはプレフィックス+を付け、無効にするには-を付ける。たとえば、+rc4 のようになる。有効な値は rc4、rc4export、rc2、rc2export、idea、des、desede3
ssl3-enabled	true	(省略可能) SSL3 を有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false デフォルトは true SSL2 と SSL3 の両方が仮想サーバーで有効になっている場合、サーバーは初めに SSL3 暗号化方式を試行する。SSL3 暗号化方式が失敗すると、SSL2 暗号化方式を試行する
ssl3-tls-ciphers	なし	(省略可能) SSL3 暗号化方式のコンマで区切られたリスト。有効にするにはプレフィックス+を付け、無効にするには-を付ける。たとえば、+rsa_des_sha のようになる。指定できる SSL3 値は rsa_rc4_128_md5、rsa3des_sha、rsa_des_sha、rsa_rc4_40_md5、rsa_rc2_40_md5、rsa_null_md5。指定できる TLS 値は rsa_des_56_sha、rsa_rc4_56_sha
tls-enabled	true	(省略可能) TLS を有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false
tls-rollback-enabled	true	(省略可能) TLS ロールバックを有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false Microsoft Internet Explorer 5.0 および 5.5 の場合には、TLS ロールバックが有効になっていることが必要。詳細については、『Sun ONE Application Server 管理者ガイド』を参照
client-auth-enabled	false	(省略可能) ACL ベースのアクセス制御とは別に、すべての要求で SSL3 クライアント認証を実行するかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

mime

MIME タイプを定義します。

要求されたリソースの MIME タイプをサーバーが決定するには、`obj.conf` ファイルの `ObjectType` セクションにある `type-by-extension` 指令を呼び出す方法が最も一般的です。`type-by-extension` 関数は、`server` 要素で `mime` 要素が定義されていない場合は機能しません。

注 `mime.types` インタフェースは不確定です。不確定なインタフェース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。

サブ要素

なし

属性

次の表は、`mime` 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

mime の属性

属性	デフォルト値	説明
<code>id</code>	なし	MIME タイプのリストの内部名。仮想サーバーが使用する MIME タイプを定義するために、 <code>virtual-server</code> 要素で使用される。MIME タイプ名は数字で始めることはできない
<code>file</code>	なし	MIME タイプのファイルの名前。このファイルの書式設定の詳細については、第 4 章「MIME タイプ」を参照

acl

ACL ファイルを参照します。

注 ACL ファイルインタフェースは、不確定です。不確定なインタフェース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。

サブ要素

なし

属性

次の表は、acl 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

acl の属性

属性	デフォルト値	説明
id	なし	ACL ファイルのリストの内部名。仮想サーバーで使用される ACL ファイルを定義するために、virtual-server 要素で使用される。ACL ファイル名は数字で始めることはできない
file	なし	ACL ファイル。各 ACL ファイルの名前は、一意であることが必要。ACL ファイルの書式については、『Sun ONE Application Server セキュリティ管理者ガイド』を参照

virtual-server-class

仮想サーバークラスを定義します。

サブ要素

次の表は、virtual-server-class 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

virtual-server-class のサブ要素

要素	必要指定数	説明
virtual-server	0 または 1 個以上	仮想サーバーを定義する
http-qos	0 または 1 個	サービス品質パラメータを定義する

virtual-server-class のサブ要素 (続き)

要素	必要指定数	説明
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、virtual-server-class 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

virtual-server-class の属性

属性	デフォルト値	説明
id	なし	仮想サーバークラスの ID。これは、特定の仮想サーバークラスの検索を可能にする一意の ID。仮想サーバークラス ID は数字で始めることはできない
config-file	なし	(省略可能) この仮想サーバークラス用の obj.conf ファイルの名前。virtual-server 要素内でオーバーライドされることがある
default-object	デフォルト	(省略可能) obj.conf ファイルから読み込まれるどのオブジェクトがデフォルトなのかをサーバーに指示する。virtual-server 要素内でオーバーライドされることがある デフォルトのオブジェクトは、仮想サーバー用のすべての名前変換 (NameTrans) 指令を持っているとみなされる。また、デフォルトオブジェクトに設定されているサーバーの動作は、仮想サーバークラス全体に影響する 存在しないオブジェクトを指定した場合、サーバーは、クライアントがドキュメントを取得しようとするまではエラーを報告しない。管理インタフェースでは、デフォルトが default という名前のオブジェクトであるとみなされる。管理インタフェースを使う (または使う予定の) 場合は、この規則に従う必要がある
accept-language	false	(省略可能) true の場合、サーバーは Accept-Language ヘッダーを解析し、クライアントが受け入れることのできる言語に基づいて適切な言語のバージョンを送信する。サーバーが複数の言語をサポートする場合にのみ、この値を true に設定する。virtual-server 要素内でオーバーライドされることがある。有効な値は on、off、yes、no、1、0、true、false。virtual-server 要素内でオーバーライドされることがある
enabled	true	(省略可能) 仮想サーバークラスをアクティブにするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

virtual-server

仮想サーバーを定義します。仮想サーバーは仮想ホストとも呼ばれます。このサーバーは、特定の URL 向けのコンテンツを提供する仮想 Web サーバーです。複数の仮想サーバーが、同じまたは異なるホスト名、ポート番号、または IP アドレスを使ってコンテンツを提供できます。HTTP サービスは、受信する Web 要求を URL に基づいて異なる仮想サーバーに送ることができます。

Sun ONE Application Server を最初にインストールするときに、デフォルトの仮想サーバーが作成されます。作成する新しい http-listener ごとに、デフォルトの仮想サーバーを割り当てることもできます。

注 仮想サーバーとサーバーインスタンスは別のものです。各サーバーインスタンスは、1 つ以上の仮想サーバーを含み、完全に個別のサーバーです。

Sun ONE Application Server は、要求を処理する前に、リスナーを介して要求を受け取り、要求を正しい仮想サーバーに転送する必要があります。仮想サーバーは、次のように決定されます。

- リスナーにデフォルトの仮想サーバーのみが設定されている場合、その仮想サーバーが選択される
- リスナーに複数の仮想サーバーが設定されている場合、要求 Host ヘッダーは仮想サーバーの host 属性と比較される。Host ヘッダーが存在しないか、host 属性が一致しない場合、リスナーのデフォルトの仮想サーバーが選択される

仮想サーバーが SSL リスナーに設定されている場合、起動時にその hosts 属性が証明書のサブジェクトのパターンと比較されます。それらが一致しない場合、警告が生成され、サーバーログに書き込まれます。

サブ要素

次の表は、virtual-server 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

virtual-server のサブ要素

要素	必要指定数	説明
http-qos	0 または 1 個	サービス品質パラメータを定義する
auth-db	0 または 1 個以上	J2EE アプリケーションに適用できない、仮想サーバー用のユーザーデータベースを定義する
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、`virtual-server` 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

virtual-server の属性

属性	デフォルト値	説明
<code>id</code>	なし	仮想サーバー ID。特定の仮想サーバーの検索を可能にする一意の ID。 <code>obj.conf</code> ファイル内の変数 <code>\$id</code> として参照することもできる。仮想サーバー ID は数字で始めることはできない
<code>http-listeners</code>	なし	(省略可能) 仮想サーバーが使用する接続を指定する、 <code>http-listener id</code> のリスト。各 <code>id</code> はコンマで区切られる。 <code>http-listener</code> の <code>default-virtual-server</code> ではない <code>virtual-server</code> の場合にのみ必要
<code>default-web-module</code>	なし	(省略可能) この仮想サーバー用のデフォルトの <code>web-module</code> で、この仮想サーバーに配備されたその他の Web モジュールで解決できないすべての要求に応答する この属性が空になっている場合、この仮想サーバーに割り当てられた <code>context-root</code> 属性が空になっている <code>web-module</code> が使われる。この仮想サーバーに割り当てられた <code>web-module</code> で <code>context-root</code> 属性が空になっているものがない場合、システムのデフォルトの Web モジュールが使われる
<code>config-file</code>	<code>virtual_server_name-obj.conf</code>	(省略可能) この仮想サーバー用の <code>obj.conf</code> ファイルの名前。 <code>virtual-server-class</code> 要素の値をオーバーライドすることがある

virtual-server の属性 (続き)

属性	デフォルト値	説明
default-object	デフォルト	<p>(省略可能) obj.conf ファイルから読み込まれるどのオブジェクトがデフォルトなのかをサーバーに指示する。</p> <p>virtual-server-class 要素の値をオーバーライドすることがある</p> <p>デフォルトのオブジェクトは、仮想サーバー用のすべての名前変換 (NameTrans) 指令を持っているとみなされる。また、デフォルトオブジェクトで設定されているサーバーの動作は、サーバー全体に影響する。デフォルト値は default</p> <p>存在しないオブジェクトを指定した場合、サーバーは、クライアントがドキュメントを取得しようとするまではエラーを報告しない。管理インタフェースでは、デフォルトが default という名前のオブジェクトであるとみなされる。管理インタフェースを使う (または使う予定の) 場合は、この規則に従う必要がある</p>
hosts	なし	<p>仮想サーバーを選択するために Host 要求ヘッダーに指定できる値のリスト。各値は、コンマで区切られる。同じ http-listener に設定された各 virtual-server は、そのリスナーに対して一意の hosts 値を持っていることが必要</p>
mime	なし	<p>仮想サーバーが使用する、mime 要素の id</p>
state	on	<p>(省略可能) virtual-server をアクティブ (on) または非アクティブ (off、disabled) にするかどうかを決定する。デフォルトは on (アクティブ)。非アクティブの場合、virtual-server は要求を処理しない</p> <p>virtual-server が disabled になっている場合、グローバルサーバー管理者だけが on にすることができる</p>
acls	なし	<p>(省略可能) acl 要素の 1 つ以上の id 属性であり、コンマで区切られている。仮想サーバーが使用する ACL ファイルを指定する</p>

virtual-server の属性 (続き)

属性	デフォルト値	説明
accept-language	false	(省略可能) true の場合、サーバーは Accept-Language ヘッダーを解析し、クライアントが受け入れることのできる言語に基づいて適切な言語のバージョンを送信する。サーバーが複数の言語をサポートする場合にのみ、この値を on に設定する。有効な値は on、off、yes、no、1、0、true、false。virtual-server-class 要素の値をオーバーライドすることがある
log-file	server 要素の log-root 属性で指定されたディレクトリにある server.log	(省略可能) この仮想サーバーのログメッセージを、サーバーログとは別のログファイルに書き込む。仮想サーバーログが保存されるファイルとディレクトリには、サーバーで実行されているどのユーザーアカウントも書き込めることが必要。ログについての詳細は、「log-service」の説明を参照

プロパティ

次の表は、virtual-server 要素のプロパティを示します。左の列にプロパティの名前、中央の列にデフォルト値、右の列にそのプロパティの説明を示します。

virtual-server プロパティ

プロパティ	デフォルト値	説明
sso-enabled	true	true の場合、同じレルムに設定されたこの仮想サーバー上の Web アプリケーションに対してシングルサインオンを有効にする。false の場合、この仮想サーバーに対してシングルサインオンは無効になり、ユーザーは仮想サーバー上のアプリケーションごとに個別に認証を行う必要がある。有効な値は on、off、yes、no、1、0、true、false
sso-max-inactive-seconds	300	クライアントが活動を停止後、何秒後にユーザーのシングルサインオンレコードをパージ可能にするかを指定する。シングルサインオンは同じ仮想サーバー上のいくつかのアプリケーションに適用されるので、それらのいずれかのアプリケーションにアクセスすればシングルサインオンを有効な状態に保てる。値が大きいほど、ユーザーのシングルサインオンは長く持続するが、サーバーのメモリーの使用量が増えることになる
sso-reap-inaterva1-seconds	60	有効期限が切れたシングルサインオンレコードのパージを行う間隔を指定する

http-qos

http-service、virtual-server-class、または virtual-server 要素のサービス品質パラメータを定義します。

http-service 要素の属性は、サービス品質機能をアクティブにします。

詳細は、『Sun ONE Application Server Performance Tuning, Sizing, and Scaling Guide』を参照してください。

サブ要素

なし

属性

次の表は、http-qos 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

http-qos の属性

属性	デフォルト値	説明
bandwidth-limit	なし	enforce-bandwidth-limit が true の場合に必須。server、virtual-server-class、または virtual-server の帯域幅の最大値 (単位は秒あたりのバイト数)
enforce-bandwidth-limit	false	(省略可能) 帯域幅制限を適用するかどうかを指定する。有効な値は yes、no、on、off、1、0、true、false
connection-limit	なし	enforce-connection-limit が true の場合に必須。server、virtual-server-class、または virtual-server の最大同時接続数
enforce-connection-limit	false	(省略可能) 接続制限を適用するかどうかを指定する。有効な値は yes、no、on、off、1、0、true、false

auth-db

virtual-server が使用するユーザーデータベースを定義します。J2EE アプリケーションに適用できません。

指定された仮想サーバーのユーザーデータベースを選択する方法については、88 ページの「ユーザーデータベースの選択」を参照してください。

注 ユーザーデータベースはサーバー自体のセキュリティのみ適用します。J2EE アプリケーションおよびモジュールのセキュリティには関係ありません。詳細については、『Sun ONE Application Server セキュリティ管理者ガイド』を参照してください。

サブ要素

なし

属性

次の表は、auth-db 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

auth-db の属性

属性	デフォルト値	説明
id	なし	仮想サーバーの ACL ファイル内のユーザーデータベース名。ユーザーデータベース名は数字で始めることはできない
database	なし	dbswitch.conf ファイル内のユーザーデータベース名
basedn	なし	(省略可能) dbswitch.conf ファイル内の基本 DN 検索をオーバーライドする。ただし、basedn 値は、dbswitch.conf エントリからの基本 DN 値と相対関係になる
certmaps	なし	(省略可能) LDAP エントリマッピング (certmap.conf で定義されている) に使う証明書を指定する。存在しない場合、すべてのマッピングが使われる。certmap.conf 内のマッピングに基づくすべての検索は、virtual-server の最終の基本 DN と相対関係になる

iiop-service

IIOP サービスを定義します。

サブ要素

次の表は、iiop-service 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

iiop-service のサブ要素

要素	必要指定数	説明
orb	1 個のみ	ORB を設定する
ssl-client-config	0 または 1 個	ORB の SSL パラメータを定義する
iiop-listener	0 または 1 個以上	IIOP 待機ソケットを定義する

属性

なし

orb

ORB を設定します。

送信接続の SSL を有効にするには、ssl-client-config サブ要素を親 iiop-service 要素に含めます。

サブ要素

次の表は、orb 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

orb のサブ要素

要素	必要指定数	説明
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、orb 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

orb の属性

属性	デフォルト値	説明
message-fragment-size	1024	(省略可能) このバイト数より大きいサイズの GIOPv1.2 メッセージは断片化される
steady-thread-pool-size	10	(省略可能) ORB 内のワーカースレッドの最小数
max-thread-pool-size	200	(省略可能) ORB 内のワーカースレッドの最大数
idle-thread-timeout-in-seconds	300	(省略可能) この時間が経過すると、アイドル状態のワーカースレッドがプールから削除される
max-connections	1024	(省略可能) すべての IIOP リスナー上の受信接続の最大数。有効な値は整数
log-level	log-service 要素の level 属性の値	(省略可能) この要素がサーバーログに記録する、メッセージのタイプを制御する。詳細は、log-service 要素の level 属性の説明を参照
monitoring-enabled	false	(省略可能) ORB の監視を有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

ssl-client-config

ORB が送信 SSL 接続を確立してクライアントとして動作するときの、SSL パラメータを定義します。

サブ要素

次の表は、ssl-client-config 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

ssl-client-config のサブ要素

要素	必要指定数	説明
ssl	1 個のみ	SSL パラメータを定義する

属性

なし

iiop-listener

IIOP 待機ソケットを定義します。

このリスナーの SSL を有効にするには、`ssl` サブ要素を含めます。

注 管理インタフェースを使用して安全なリスナーを作成する場合、セキュリティは自動的に `init.conf` 内でグローバルにオンになります。
`sever.xml` で安全なリスナーを手動で作成する場合は、`init.conf` ファイルの `Security` 指令を編集してセキュリティをオンにする必要があります。

サブ要素

次の表は、`iiop-listener` 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

iiop-listener のサブ要素

要素	必要指定数	説明
<code>ssl</code>	0 または 1 個	SSL パラメータを定義する
<code>property</code>	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、`iiop-listener` 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

iiop-listener の属性

属性	デフォルト値	説明
<code>id</code>	なし	リスナー名。 <code>iiop-listener</code> 名は数字で始めることはできない
<code>address</code>	なし	リスナーの IP アドレス。ドットで区切ったペア、または IPv6 表記で指定する
<code>port</code>	3700 (最初のサーバーインスタンス用)	(省略可能) リスナーを作成するポート番号。有効な値は 1 ~ 65535。UNIX の場合、ポート 1 ~ 1024 で待機するソケットを作成するには、スーパーユーザー権限が必要

iiop-listener の属性 (続き)

属性	デフォルト値	説明
enabled	true	(省略可能) リスナーをアクティブにするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

コンテナ要素

コンテナ設定要素は次のとおりです。

- web-container
- ejb-container
- mdb-container

web-container

Web コンテナを設定します。

サブ要素

次の表は、web-container 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

web-container のサブ要素

要素	必要指定数	説明
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、web-container 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

web-container の属性

属性	デフォルト値	説明
monitoring-enabled	false	(省略可能) web コンテナの監視を有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

web-container の属性 (続き)

属性	デフォルト値	説明
log-level	log-service 要素の level 属性の値	(省略可能) この要素がサーバーログに記録する、メッセージのタイプを制御する。詳細は、log-service 要素の level 属性の説明を参照 ServerContext.log メッセージは、デフォルトで INFO レベルのログに記録される

ejb-container

EJB コンテナを設定します。ステートレスセッション Bean は、プールに保持されます。ステートフルセッション Beans は、セッション親和性があり、キャッシュされます。データベースの主キーに関連付けられたエンティティ Bean もキャッシュされます。主キーにまだ関連付けられていないエンティティ Bean は、プールに保持されます。プーリングされたエンティティ Bean は、ejbCreate() とファインダーメソッドを実行するために使用されます。

サブ要素

次の表は、ejb-container 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

ejb-container のサブ要素

要素	必要指定数	説明
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、ejb-container 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

ejb-container の属性

属性	デフォルト値	説明
steady-pool-size	32	<p>(省略可能) プールに保持される Bean の初期数および最小数を指定する。0 以上で、max-pool-size 未満である必要がある</p> <p>Bean インスタンスは、プールから削除され、使用後に戻される。プールはこのサイズを保つために、定期的に補充またはクリーンアップされる</p> <p>ステートレスセッション Bean とエンティティ Bean に適用される</p>
pool-resize-quantity	16	<p>(省略可能) 次の条件を満たす数の Bean を指定する。</p> <ul style="list-style-type: none"> • 利用できる Bean がプールにないときに要求を受信した場合に作成される (max-pool-size の制限を受ける) • pool-idle-timeout-in-seconds に指定された時間を過ぎ、クリーナスレッドが未使用のインスタンスを削除すると、削除される <p>0 以上で、max-pool-size 未満である必要がある。プールサイズが steady-pool-size 未満に変更されることはない</p> <p>ステートレスセッション Bean とエンティティ Bean に適用される</p>
max-pool-size	64	<p>(省略可能) クライアントの要求に対応するために作成できる Bean の最大数を指定する。値が 0 の場合は、プールのサイズに制限がないことを示す</p> <p>ステートレスセッション Bean とエンティティ Bean に適用される</p>

ejb-container の属性 (続き)

属性	デフォルト値	説明
cache-resize-quantity	16	<p>(省略可能) 次の条件を満たす Beans 数を指定する。</p> <ul style="list-style-type: none"> • 利用できる Beans がプールにないときに要求を受信した場合に作成される (max-cache-size の制限を受ける) • cache-idle-timeout-in-seconds に指定された時間を過ぎ、クリーナスレッドが未使用のインスタンスを削除するか、あるいはキャッシュサイズが max-cache-size を超える場合に非活性化される <p>1 以上で、max-cache-size 未満である必要がある</p> <p>ステートフルセッション Beans およびエンティティ Beans に適用される</p>
max-cache-size	512	<p>(省略可能) キャッシュに存在できる Beans の最大数を指定する。値が 0 の場合は、キャッシュのサイズに制限がないことを示す</p> <p>ステートフルセッション Beans およびエンティティ Beans に適用される</p>
pool-idle-timeout-in-seconds	600	<p>(省略可能) Bean がプールにアイドル状態のままではいられる最長時間を指定する。この時間を過ぎると、プールはこの Bean を削除する。0 の値を指定すると、アイドル状態の Beans がプールにいつまでも残ることができる</p> <p>ステートレスセッション Beans とエンティティ Beans に適用される</p>
cache-idle-timeout-in-seconds	600	<p>(省略可能) Bean がキャッシュにアイドル状態のままではいられる最長時間を指定する。この時間を過ぎると、コンテナはこの Bean を非活性化する。0 の値を指定すると、Beans は非活性化の候補になることはない</p> <p>ステートフルセッション Beans およびエンティティ Beans に適用される</p>

ejb-container の属性 (続き)

属性	デフォルト値	説明
removal-timeout-in-seconds	5400	<p>(省略可能) Bean が、非活性化されたまま でいられる時間 (セッションストアから削 除されるまで) を指定する。0 の値を指定 すると、コンテナは非活性化されている Beans を自動的に削除しない</p> <p>removal-timeout-in-seconds が cache-idle-timeout-in-seconds 以下 である場合は、Beans は非活性化されずに ただちに削除される</p> <p>server 要素の session-store 属性によっ て、セッションストアの位置が決まる</p> <p>ステートフルセッション Beans およびエン ティティ Beans に適用される</p>
victim-selection-policy	nru	<p>(省略可能) エンティティ Beans またはス テートフルセッション Beans を非活性化す るための選択方法を指定する。有効な値は fifo、lru、および nru。</p> <ul style="list-style-type: none"> •fifo を指定すると、最も古いインスタンス が選択される •lru を指定すると、最近アクセスしたイン スタンスが選択される •nru を指定すると、最近使っていないイン スタンスが選択される
commit-option	B	<p>(省略可能) エンティティ Beans に使うコミッ トオプションを決定する。有効な値は B また は C</p>
log-level	log-service 要素の level 属性の値	<p>(省略可能) この要素がサーバーログに記録す る、メッセージのタイプを制御する。詳細は、 log-service 要素の level 属性の説明を参 照</p>
monitoring-enabled	false	<p>(省略可能) EJB コンテナの監視を有効にする かどうかを決定する。有効な値は on、off、 yes、no、1、0、true、false</p>

mdb-container

メッセージ駆動型 Beans (MDB) コンテナを設定します。

サブ要素

次の表は、mdb-container 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

mdb-container のサブ要素

要素	必要指定数	説明
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、mdb-container 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

mdb-container の属性

属性	デフォルト値	説明
steady-pool-size	10	(省略可能) プールに保持される Beans の初期数および最小数を指定する
pool-resize-quantity	2	(省略可能) プールが空のときに要求を受信した場合に作成される Beans の数 (max-pool-size の制限を受ける)、または idle-timeout-in-seconds 以上の間アイドル状態の場合に削除する Beans の数を指定する
max-pool-size	60	(省略可能) クライアントの要求に対応するために作成できる Beans の最大数を指定する
idle-timeout-in-seconds	600	(省略可能) Bean がプールにアイドル状態のままえられる最長時間を指定する。この時間が経過すると、Bean は削除される
log-level	log-service 要素の level 属性の値	(省略可能) この要素がサーバーログに記録する、メッセージのタイプを制御する。詳細は、log-service 要素の level 属性の説明を参照

mdb-container の属性 (続き)

属性	デフォルト値	説明
monitoring-enabled	false	(省略可能) メッセージ駆動型 Beans (MDB) コンテナの監視を有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

プロパティ

次の表は、mdb-container 要素のプロパティを示します。左の列にプロパティの名前、中央の列にデフォルト値、右の列にそのプロパティの説明を示します。

mdb-container のプロパティ

プロパティ	デフォルト値	説明
cmt-max-runtime-exceptions	1	コンテナ管理のトランザクションを使用するときに、メッセージ駆動型 Beans の onMessage () メソッドから最高何回 RuntimeExceptions がスローされるかを指定する。推奨されていない
reconnect-enabled	true	true の場合、接続が切断されると、MDB コンテナが JMS プロバイダに自動的に再接続を試みる 接続が切断されると、メッセージの処理段階によっては、onMessage () メソッドを完了させることができないか、トランザクションが JMS 例外によりロールバックされることがある。MDB コンテナが再度接続を確立すると、JMS メッセージの再配信セマンティックスが適用される
reconnect-delay-in-seconds	60	再接続の試みの遅延間隔を指定する
reconnect-max-retries	60	再接続の試みの最大数を指定する

J2EE サービス要素

J2EE サービス要素は次のとおりです。

- `jms-service`
- `log-service`
- `security-service`
- `auth-realm`
- `transaction-service`

jms-service

Sun ONE Application Server で管理される、組み込み Java Message Service (JMS) を設定します。

サブ要素

次の表は、`jms-service` 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

`jms-service` のサブ要素

要素	必要指定数	説明
<code>property</code>	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、`jms-service` 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

`jms-service` の属性

属性	デフォルト値	説明
<code>port</code>	7676	(省略可能) JMS プロバイダが使うポート番号を指定する
<code>admin-user-name</code>	<code>admin</code>	JMS プロバイダの管理者ユーザー名を指定する
<code>admin-password</code>	<code>admin</code>	JMS プロバイダの管理者パスワードを指定する
<code>init-timeout-in-seconds</code>	60	(省略可能) 起動時にサーバーインスタンスが、対応する JMS インスタンスの応答を待つ時間を指定する。応答がない場合、起動は中断される。0 に設定すると、サーバーインスタンスはいつまでも待機する

jms-service の属性 (続き)

属性	デフォルト値	説明
start-args	なし	(省略可能) 対応する JMS インスタンスを起動するための引数の文字列を指定する
log-level	log-service 要素の level 属性の値	(省略可能) この要素がサーバーログに記録する、メッセージのタイプを制御する。詳細は、log-service 要素の level 属性の説明を参照
enabled	true	(省略可能) true に設定すると、Sun ONE Applicatin Server インスタンスが JMS プロバイダの起動と停止を行う。false に設定すると、Sun ONE Application Servr インスタンスは JMS プロバイダの起動と終了を行わない (JMS プロバイダが、使われていないか、Sun ONE Applicatin Server から独立して管理されているため) 有効な値は on、off、yes、no、1、0、true、false

プロパティ

次の表は、jms-service 要素のプロパティを示します。左の列にプロパティの名前、中央の列にデフォルト値、右の列にそのプロパティの説明を示します。

jms-service のプロパティ

プロパティ	デフォルト値	説明
instance-name	<i>domain_instance</i>	ドメインとサーバーインスタンス名をつなげた、完全な Sun ONE Message Queue ブローカ名を指定する。(例: domain1_server1)
instance-name-suffix	なし	完全な Sun ONE Message Queue ブローカインスタンス名に追加するサフィックスを指定する。サフィックスとインスタンス名は、下線文字 () で区切られる。たとえば、インスタンス名が domain1_server1 である場合に、サフィックス xyz を付加すると、インスタンス名は domain1_server1_xyz になる
append-version	false	true の場合、下線文字 () を前に付けて、メジャーおよびマイナーバージョン番号を完全な Sun ONE Message Queue ブローカインスタンス名に付加する。たとえば、インスタンス名が domain1_server1 である場合に、バージョン番号を付加すると、インスタンス名は domain1_server1_7_0 になる

log-service

次のログファイルを含む、システムログサービスを設定します。

- **サーバーログファイル**にはデフォルトの仮想サーバーからのメッセージが格納されます。その他の設定されている仮想サーバーからのメッセージもここに格納されます。ただし、`log-file` 属性が `virtual-server-class` または `virtual-server` 要素内で明示的に指定されている場合を除きます。デフォルトの名前は `server.log` です。
- **アクセスログファイル**にはデフォルトの仮想サーバーからの HTTP アクセスメッセージが格納されます。デフォルトの名前は `access.log` です。アクセスログを設定するには、`init.conf` および `obj.conf` ファイルのサーバーアプリケーションの関数を使います。詳細については、『Sun ONE Application Server Developer's Guide to NSAPI』を参照してください。
- **トランザクションログファイル**にはデフォルトの仮想サーバーからのトランザクションメッセージが格納されます。これらのファイルが格納されているディレクトリのデフォルト名は `tx` です。
- **仮想サーバーログファイル**には、`log-file` 属性が明示的に指定された、`virtual-server-class` または `virtual-server` 要素からのメッセージが格納されます。

サブ要素

次の表は、`log-service` 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

log-service のサブ要素

要素	必要指定数	説明
<code>property</code>	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、`log-service` 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

log-service の属性

属性	デフォルト値	説明
file	server.log	<p>(省略可能) サーバーログの名前または位置をオーバーライドする。サーバーログが保存されるファイルとディレクトリには、サーバーを実行しているユーザーアカウントも書き込めることが必要</p> <p>絶対パスを指定すると、この値が server 要素の log-root 属性をオーバーライドする</p> <p>相対パスを指定する場合、相対パスは server 要素の log-root 属性に相対的になる。log-root 値を指定しない場合は、このパスは instance_dir/config に相対的になる</p>
level	INFO	<p>(省略可能) ほかの要素がサーバーログに記録する、メッセージのデフォルトのタイプを制御する。多くのその他の要素が、log-level 属性のこのデフォルトをオーバーライドできる。許可されている値 (降順) は以下のとおり。FINEST、FINER、FINE、CONFIG、INFO、WARNING、SEVERE、ALERT、FATAL</p> <p>各値は、それよりも低いレベルの値のすべてのメッセージをログに記録する。たとえば、FINEST ではすべてのメッセージが記録され、FATAL では FATAL メッセージだけが記録される。デフォルト値は SEVERE であり、SEVERE、ALERT、FATAL のすべてのメッセージがログに記録される</p>
log-stdout	true	<p>(省略可能) true の場合、stdout 出力がサーバーログにリダイレクトされる。有効な値は on、off、yes、no、1、0、true、false</p>
log-stderr	true	<p>(省略可能) true の場合、stderr 出力がサーバーログにリダイレクトされる。有効な値は on、off、yes、no、1、0、true、false</p>
echo-log-messages-to-stderr	true	<p>(省略可能) true の場合、サーバーログのほかにログメッセージが stderr に送信される。有効な値は on、off、yes、no、1、0、true、false</p>
create-console	false	<p>(オプション。Windows のみ) true の場合、stderr 出力用に Windows コンソールが作成される。有効な値は on、off、yes、no、1、0、true、false</p>
log-virtual-server-id	false	<p>(省略可能) true の場合、仮想サーバー ID が仮想サーバーログに表示される。これは、複数の virtual-server 要素が同じログファイルを共有している場合に有用。有効な値は on、off、yes、no、1、0、true、false</p>

log-service の属性 (続き)

属性	デフォルト値	説明
use-system-logging	false	(省略可能) true の場合、UNIX syslog サービスまたは Windows Event Logging を使ってログの作成と管理が行われる。有効な値は on、off、yes、no、1、0、true、false

security-service

J2EE セキュリティサービスで必要となるパラメータと設定情報を定義します。

サブ要素

次の表は、security-service 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

security-service のサブ要素

要素	必要指定数	説明
auth-realm	1 個以上	認証のレルムを定義する
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、security-service 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

security-service の属性

属性	デフォルト値	説明
default-realm	file	(省略可能) このサーバーインスタンスのアクティブな認証レルム (auth-realm name 属性) を指定する
default-principal	なし	(省略可能) 主体がない場合に、必要に応じて、デフォルトのセキュリティコンテキストの ID として使われる。通常のサーバー操作では、この属性を設定する必要はない
default-principal-password	なし	(省略可能) デフォルトの主体のパスワード。通常のサーバー操作では、この属性を設定する必要はない

security-service の属性 (続き)

属性	デフォルト値	説明
anonymous-role	ANYONE	(省略可能) デフォルトまたは匿名のロールの名前として使われる。匿名のロールは常にすべての主体に割り当てられる。このロール値を J2EE 配備記述子を使って、誰にでもアクセス権を与えることができる
audit-enabled	false	(省略可能) true の場合、監査情報を提供するために追加のアクセスログが実行される。有効な値は on、off、yes、no、1、0、true、false ○監査情報には、次の内容が含まれる。 <ul style="list-style-type: none"> • 認証の成功および失敗イベント • サブレットおよび EJB へのアクセスの許可と拒否
log-level	log-service 要素の level 属性の値	(省略可能) この要素がサーバーログに記録する、メッセージのタイプを制御する。詳細は、log-service 要素の level 属性の説明を参照

auth-realm

認証のレルムを定義します。

認証のレルムには、プロバイダ固有のプロパティが必要となり、このプロパティは特定の実装のニーズによって異なります。

レルムを定義する方法については、『Sun ONE Application Server 開発者ガイド』を参照してください。

次に、デフォルトの file レルムの例を示します。

```
<auth-realm name="file"
  classname="com.ipplanet.ias.security.auth.realm.file.FileRealm">
  <property name="file" value="instance_dir/config/keyfile"/>
  <property name="jaas-context" value="fileRealm"/>
</auth-realm>
```

auth-realm 要素が使用するプロパティは、auth-realm 要素の name 属性によって決まります。file レルムは、file および jaas-context プロパティを使用します。その他のレルムは、別のプロパティを使用します。

サブ要素

次の表は、auth-realm 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

auth-realm のサブ要素

要素	必要指定数	説明
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、auth-realm 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

auth-realm の属性

属性	デフォルト値	説明
name	なし	このレルムの名前を指定する
classname	なし	このレルムを実装する Java クラスを指定する

プロパティ

Sun ONE Application Server が提供する標準のレルムには、必須および省略可能なプロパティがあります。カスタムレルムのプロパティは、標準レルムのプロパティとは異なるものを使用できます。

次の表は、auth-realm 要素のプロパティを示します。左の列にプロパティの名前、中央の列にプロパティを使用する標準のレルム、右の列にそのプロパティの説明を示します。

auth-realm のプロパティ

プロパティ	レルム	説明
jaas-context	file, ldap, solaris	JAAS (Java Authentication and Authorization Service) コンテキストを指定する
file	file	ユーザー名を保存するファイルを指定する。デフォルトは、 <i>instance_dir/config/keyfile</i> である

auth-realmのプロパティ (続き)

プロパティ	レルム	説明
assign-groups	certificate	(省略可能) このプロパティを設定すると、その値はグループ名をコンマで区切ったリストの形になる。有効な証明書を提示するすべてのクライアントは、Web および EJB コンテナで承認を行うために、これらのグループのメンバーシップを割り当てられる
directory	ldap	LDAP URL をサーバーに指定する
base-dn	ldap	ユーザーデータの位置の LDAP 基本 DN を指定する。ツリー範囲検索が実行されるため、この基本 DN は、ユーザーデータのレベルより上に置く。検索ツリーが小さければ小さいほど、パフォーマンスは良くなる
search-filter	ldap	(省略可能) ユーザーを検索するために使う検索フィルタを指定する。デフォルトは、uid=%s である (%s はサブジェクト名に展開する)
group-base-dn	ldap	(省略可能) グループデータの位置の基本 DN を指定する。デフォルトでは、base-dn と同じであるが、必要に応じて変更することができる
group-search-filter	ldap	(省略可能) ユーザーのグループメンバーシップを検索するために使う検索フィルタを指定する。デフォルトは、uniquemember=%d である (%d はユーザー要素 DN に展開する)
group-target	ldap	(省略可能) グループ名エントリを含む LDAP の属性名を指定する。デフォルトは、CN である
search-bind-dn	ldap	(省略可能) search-filter 検索を実行するためにディレクトリに対する認証に使うオプションの DN を指定する。匿名検索が認められていないディレクトリにだけ必要
search-bind-password	ldap	(省略可能) search-bind-dn に指定された DN の LDAP パスワードを指定する

transaction-service

Java Transaction Service (JTS) を設定します。

サブ要素

次の表は、transaction-service 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

transaction-service のサブ要素

要素	必要指定数	説明
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、transaction-service 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

transaction-service の属性

属性	デフォルト値	説明
automatic-recovery	false	(省略可能) true の場合、サーバーインスタンスは起動時にトランザクションの復旧を実行する。有効な値は on、off、yes、no、1、0、true、false
timeout-in-seconds	0	(省略可能) トランザクションを中止するまでの時間を指定する。0 に設定すると、トランザクションはタイムアウトにならない
tx-log-dir	server 要素の log-root 属性で指定したディレクトリ	(省略可能) トランザクションログディレクトリの位置をオーバーライドする。トランザクションログが保存されるディレクトリには、サーバーで実行されているどのユーザーカウントも書き込めることが必要。ログについての詳細は、「log-service」の説明を参照
heuristic-decision	rollback	(省略可能) 復旧中に、トランザクションの結果をログから決定できない場合、このプロパティが結果を決定する。有効な値は rollback および commit

transaction-service の属性 (続き)

属性	デフォルト値	説明
keypoint-interval	2048	(省略可能) ログ内でのキーポイント処理間のトランザクションの数を指定する。キーポイント処理は、トランザクションログファイルを圧縮してサイズを小さくする。この属性の値を大きくすると (たとえば 4096)、トランザクションログファイルが大きくなるが、キーポイント処理が少なくなり、パフォーマンスが向上する可能性がある。小さい値 (たとえば 100) にするとログファイルは小さくなるが、キーポイント処理の頻度が増えるためパフォーマンスが少し低下する
log-level	log-service 要素の level 属性の値	(省略可能) この要素がサーバーログに記録する、メッセージのタイプを制御する。詳細は、log-service 要素の level 属性の説明を参照
monitoring-enabled	false	(省略可能) JTS の監視を有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false。デフォルトは false

プロパティ

次の表は、transaction-service 要素のプロパティを示します。左の列にプロパティの名前、中央の列にデフォルト値、右の列にそのプロパティの説明を示します。

transaction-service のプロパティ

プロパティ	デフォルト値	説明
oracle-xa-recovery-workaround	false	true の場合、Oracle XA Resource の回避策がトランザクションの復旧に使用される。有効な値は on、off、yes、no、1、0、true、false
sybase-xa-recovery-workaround	false	true の場合、Sybase XA Resource の回避策がトランザクションの復旧に使用される。有効な値は on、off、yes、no、1、0、true、false
disable-distributed-transaction-logging	false	true の場合、トランザクションのログへの記録を無効にするので、パフォーマンスが向上する。有効な値は on、off、yes、no、1、0、true、false
xaresource-txn-timeout	使用される XAResource に固有	XAResource タイムアウトを変更する。XAResource のデフォルトのタイムアウトによりトランザクションが中止されることがあるので、この値を変更することが望ましい

Java 設定要素

Java 設定要素は次のとおりです。

- java-config
- profiler
- jvm-options

java-config

Java Virtual Machine (JVM) 設定パラメータを指定します。

サブ要素

次の表は、java-config 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

java-config のサブ要素

要素	必要指定数	説明
profiler	0 または 1 個	Sun ONE Application Server 用にプロファイラを設定する
jvm-options	0 または 1 個以上	JVM コマンド行オプションを含む
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、java-config 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

java-config の属性

属性	デフォルト値	説明
java-home	なし	JDK がインストールされているディレクトリへのパス
debug-enabled	false	(省略可能) true の場合、サーバーは JPDA ベースのデバッガに接続できるようにデバッグモードで起動する。有効な値は on、off、yes、no、1、0、true、false

java-config の属性 (続き)

属性	デフォルト値	説明
debug-options	-Xdebug -Xrunjdwpt:transport =dt_socket,server=y ,suspend=n	(省略可能) JPDA (Java Platform Debugger Architecture) オプションを指定する。含めることのできるデバッグオプションのリストは、次のサイトを参照 http://java.sun.com/products/jpda/doc/conninv.html#Invocation デバックの詳細は、『Sun ONE Application Server 開発者ガイド』を参照
rmic-options	-iiop -poa -alwaysgenerate -keepgenerated -g	(省略可能) アプリケーションの配備時に RMI コンパイラに渡すオプションを指定する。 -keepgenerated オプションは、生成されたソースをスタブとタイ用に保存する。 rmic コマンドの詳細は、『Sun ONE Application Server Enterprise Java Beans 開発者ガイド』を参照
javac-options	-g	(省略可能) アプリケーションの配備時に Java コンパイラに渡すオプションを指定する。
classpath-prefix	なし	(省略可能) システムクラスパスのプレフィックスを指定する。XML パーサークラスなどのシステムクラスをオーバーライドしたい場合にだけ、システムクラスパスにプレフィックスを付ける。この属性を使用する際には注意が必要である
server-classpath	なし	(省略可能) サーバーが起動された環境のクラスパスを指定する。このクラスパスには、 <code>System.getProperty("java.class.path")</code> を使ってアクセスできる
classpath-suffix	なし	(省略可能) システムクラスパスのサフィックスを指定する
native-library-path-prefix	なし	(省略可能) ネイティブライブラリパスのプレフィックスを指定する ネイティブライブラリパスとは、Application Server のネイティブ共有ライブラリの相対パス、標準の JRE ネイティブライブラリパス、シェル環境設定 (UNIX の <code>LD_LIBRARY_PATH</code>)、および profiler 要素に指定されたパスを、自動的に連結して作成したパスのことである。このパスは合成されるので、サーバー設定としては表示されない

java-config の属性 (続き)

属性	デフォルト値	説明
native-library-path-suffix	なし	(省略可能) ネイティブライブラリパスのサフィックスを指定する
bytecode-preprocessors		(省略可能) それぞれに <code>com.sun.appserv.BytecodePreprocessor</code> インタフェースを実装する必要があるクラス名をコンマで区切ったリストの形で指定する。指定された各プリプロセッサクラスは、指定された順に呼び出される
env-classpath-ignored	true	(省略可能) <code>false</code> の場合、 <code>CLASSPATH</code> 環境変数が読み取られ、 <code>Sun ONE Application Server</code> のクラスパスに付加される。 <code>CLASSPATH</code> 環境変数は、一番最後の、 <code>classpath-suffix</code> の後に追加される 開発環境では、この値を <code>false</code> に設定することが必要。本稼動環境では、この値を <code>true</code> に設定して、環境変数による影響を避ける必要がある 有効な値は <code>on</code> 、 <code>off</code> 、 <code>yes</code> 、 <code>no</code> 、 <code>1</code> 、 <code>0</code> 、 <code>true</code> 、 <code>false</code>

profiler

Sun ONE Application Server 用にプロファイラを設定します。プロファイラの詳細は、『Sun ONE Application Server 開発者ガイド』を参照してください。

サブ要素

次の表は、`profiler` 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

profiler のサブ要素

要素	必要指定数	説明
<code>jvm-options</code>	0 または 1 個以上	プロファイラ固有の JVM コマンド行オプションを含む
<code>property</code>	0 または 1 個以上	プロパティまたは変数を指定する

注 profiler 要素のサブ要素は任意の順序で記述できます。

属性

次の表は、profiler 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

profiler の属性

属性	デフォルト値	説明
name	なし	プロファイラの名前を指定する
classpath	なし	(省略可能) プロファイラのクラスパスを指定する
native-library-path	なし	(省略可能) プロファイラのネイティブライブラリパスを指定する
enabled	true	(省略可能) プロファイラを有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

jvm-options

次のような JVM コマンド行オプションを含みます。

```
<jvm-options>-Xdebug -Xmx128m</jvm-options>
```

使用可能なオプションについては、次のサイトを参照してください。

<http://java.sun.com/docs/hotspot/VMOptions.html>

サブ要素

なし

属性

なし

リソース要素

リソース要素は次のとおりです。

- resources
- custom-resource
- external-jndi-resource
- jdbc-resource
- mail-resource
- jms-resource
- persistence-manager-factory-resource
- jdbc-connection-pool

resources

データベース接続、JavaMail セッションなどの設定されたリソースが含まれています。

注 各リソースに JNDI 名を指定する必要があります。JNDI のその他の企業リソースの名前との衝突や移植の際の問題を回避するためには、Sun ONE Application Server アプリケーションでは名前をすべて、文字列 `java:comp/env` で始める必要があります。

サブ要素

次の表は、resources 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

resources のサブ要素

要素	必要指定数	説明
custom-resource	0 または 1 個以上	カスタムリソースを定義する
external-jndi-resource	0 または 1 個以上	外部 JNDI リポジトリに配備されているリソースを定義する
jdbc-resource	0 または 1 個以上	JDBC (Java Database Connectivity) リソースを定義する
mail-resource	0 または 1 個以上	JavaMail リソースを定義する
jms-resource	0 または 1 個以上	JMS リソースを定義する

resources のサブ要素 (続き)

要素	必要指定数	説明
persistence-manager-factory-resource	0 または 1 個以上	CMP 用の持続マネージャファクトリリソースを定義する
jdbc-connection-pool	0 または 1 個以上	JDBC 接続プールの作成に必要なプロパティを定義する

注 resources 要素のサブ要素は任意の順序で記述できます。

属性
なし

custom-resource

カスタムサーバー全体のリソースオブジェクトファクトリを指定する、カスタムリソースを定義します。そのようなオブジェクトファクトリは、`javax.naming.spi.ObjectFactory` インタフェースを実装します。

サブ要素

次の表は、`custom-resource` 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

custom-resource のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、`custom-resource` 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

custom-resource の属性

属性	デフォルト値	説明
jndi-name	なし	リソースの JNDI 名を指定する

custom-resource の属性 (続き)

属性	デフォルト値	説明
res-type	なし	リソースの完全指定されたタイプを指定する
factory-class	なし	javax.naming.spi.ObjectFactory を実装する、ユーザーが記述したファクトリクラスの完全指定名を指定する
enabled	true	(省略可能) リソースを実行時に有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

external-jndi-resource

外部 JNDI リポジトリに配備されているリソースを定義します。たとえば、汎用 Java オブジェクトを LDAP サーバーに格納します。外部 JNDI ファクトリは、javax.naming.spi.InitialContextFactory インタフェースを実装する必要があります。

サブ要素

次の表は、external-jndi-resource 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

external-jndi-resource のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、external-jndi-resource 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

external-jndi-resource の属性

属性	デフォルト値	説明
jndi-name	なし	リソースの JNDI 名を指定する
jndi-lookup-name	なし	リソースの JNDI ルックアップ名を指定する

external-jndi-resource の属性 (続き)

属性	デフォルト値	説明
res-type	なし	リソースの完全指定されたタイプを指定する
factory-class	なし	javax.naming.spi.InitialContextFactory を実装する、ファクトリクラスの完全指定名を指定する JNDI の詳細は、『Sun ONE Application Server Developer's Guide to J2EE Features and Services』を参照
enabled	true	(省略可能) リソースを実行時に有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

jdbc-resource

JDBC (javax.sql.DataSource) リソースを定義します。

サブ要素

次の表は、jdbc-resource 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

jdbc-resource のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、jdbc-resource 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

jdbc-resource の属性

属性	デフォルト値	説明
jndi-name	なし	リソースの JNDI 名を指定する
pool-name	なし	jdbc-connection-pool 要素で定義されている、関連付けられた JDBC 接続プールの name を指定する

jdbc-resource の属性 (続き)

属性	デフォルト値	説明
enabled	true	(省略可能) リソースを実行時に有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

mail-resource

JavaMail (javax.mail.Session) リソースを定義します。

サブ要素

次の表は、mail-resource 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

mail-resource のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、mail-resource 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

mail-resource の属性

属性	デフォルト値	説明
jndi-name	なし	リソースの JNDI 名を指定する
store-protocol	imap	(省略可能) ストアプロトコルサービスを指定する。このサービスは、メールサーバーへの接続、メッセージの取得、およびフォルダへのメッセージの保存を実行する。値の例は imap および pop3
store-protocol-class	com.sun.mail.imap.IMAPStore	(省略可能) ストア用のサービスプロバイダ実装クラスを指定する
transport-protocol	smtp	(省略可能) メッセージを送信するトランスポートプロトコルサービスを指定する

mail-resource の属性 (続き)

属性	デフォルト値	説明
transport-protocol-class	com.sun.mail.smtp. .SMTPTransport	(省略可能) トランスポート用のサービスプロバイダ実装クラスを指定する
host	なし	メールサーバーのホスト名
user	なし	メールサーバーのユーザー名
from	なし	メッセージの送信者を示すためにメールサーバーが使う電子メールアドレス
debug	false	(省略可能) リソースのデバッグを有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false
enabled	true	(省略可能) リソースを実行時に有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

プロパティ

mail-resource 要素のプロパティを設定してから、後で `JavaMail Session` オブジェクトから設定したプロパティを取得できます。すべてのプロパティ名は、mail- というプレフィックスで始める必要があります。Sun ONE Application Server は、プロパティ名内のダッシュ (-) をペリオド (.) に変換し、プロパティを `MailConfiguration` および `JavaMail Session` オブジェクトに保存します。プロパティの名前が mail- で始まらない場合は、プロパティは無視されます。

たとえば、`JavaMail Session` オブジェクト内に `mail.password` プロパティを定義したい場合は、まず次のように `server.xml` を編集します。

```
...
<mail-resource jndi-name="mail/Session" ...>
  <property name="mail-password" value="adminadmin"/>
</mail-resource>
...
```

`JavaMail Session` オブジェクトを取得後、次のようにして、値 `adminadmin` を得るために `mail.password` プロパティを取得できます。

```
String password = session.getProperty("mail.password");
```

jms-resource

JMS (Java Message Service) リソースを定義します。

サブ要素

次の表は、`jms-resource` 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

`jms-resource` のサブ要素

要素	必要指定数	説明
<code>description</code>	0 または 1 個	この要素を説明するテキストを含む
<code>property</code>	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、`jms-resource` 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

`jms-resource` の属性

属性	デフォルト値	説明
<code>jndi-name</code>	なし	リソースの JNDI 名を指定する
<code>res-type</code>	なし	リソースの完全修飾されたタイプを指定する。次のいずれかのタイプを指定できる <code>javax.jms.Topic</code> <code>javax.jms.Queue</code> <code>javax.jms.TopicConnectionFactory</code> <code>javax.jms.QueueConnectionFactory</code>
<code>enabled</code>	<code>true</code>	(省略可能) リソースを実行時に有効にするかどうかを決定する。有効な値は <code>on</code> 、 <code>off</code> 、 <code>yes</code> 、 <code>no</code> 、 <code>1</code> 、 <code>0</code> 、 <code>true</code> 、 <code>false</code>

プロパティ

次の表は、`jms-resource` 要素のプロパティを示します。左の列にプロパティの名前、中央の列にデフォルト値、右の列にそのプロパティの説明を示します。

jms-service のプロパティ

プロパティ	デフォルト値	説明
imqDestinationName	なし	この JMS リソースに関連付けられた物理的な JMS 送信先名を指定する。javax.jms.Topic または javax.jms.Queue の res-type で jms-resource 要素のプロパティを指定する必要がある 『Sun ONE Message Queue Administrator's Guide』には、このプロパティのデフォルトの値が示されているが、この値は Sun ONE Application Server 環境に適用しない
imqBrokerHostName	Sun ONE Application Server インスタンスと同じホスト名 (localhost)	JMS サービス (Sun ONE Message Queue ブローカ) が実行されているホストの名前を指定する。res-type が javax.jms.TopicConnectionFactory または javax.jms.QueueConnectionFactory の jms-resource 要素用
imqBrokerHostPort	jms-service 要素の port 属性	JMS サービス (Sun ONE Message Queue ブローカ) が実行されているポートを指定する。res-type が javax.jms.TopicConnectionFactory または javax.jms.QueueConnectionFactory の jms-resource 要素用
imqConfiguredClientID	なし	QueueConnectionFactory および TopicConnectionFactory クラスそれぞれに対応する createQueueConnection および createTopicConnection JMS API を使用して作成した Connection に関連付けられた JMS クライアント ID を指定する res-type が javax.jms.TopicConnectionFactory または javax.jms.QueueConnectionFactory の jms-resource 要素用 永続的なサブスクリプション名は一意であり、クライアント ID の範囲内でのみ有効である。永続的サブスクリプションを作成またはアクティブにし直すには、接続には有効なクライアント ID が必要である。JMS 仕様によると、クライアント ID が一意であり、特定のクライアント ID は 1 度に 1 つのアクティブな接続にしか使用できない

persistence-manager-factory-resource

コンテナ管理持続性 (CMP) 用の持続マネージャファクトリリソースを定義します。

サブ要素

次の表は、persistence-manager-factory-resource 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

persistence-manager-factory-resource のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、persistence-manager-factory-resource 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

persistence-manager-factory-resource の属性

属性	デフォルト値	説明
jndi-name	なし	リソースの JNDI 名を指定する
factory-class	com.sun.jdo.spi.persistence.support.sqlstore.impl.PersistenceManagerFactoryImpl	(省略可能) ファクトリクラスの名前を指定する。この属性はサードパーティ製の CMP 持続マネージャファクトリをサポートしている。サードパーティの CMP 実装が必要とする名前を使用する。この属性は、組み込み CMP 実装には使わないこと
jdbc-resource-jndi-name	なし	(省略可能) データベース接続が取得される jdbc-resource を指定する。これは、既存の jdbc-resource の jndi-name であることが必要
enabled	true	(省略可能) リソースを実行時に有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

jdbc-connection-pool

JDBC 接続プールの作成に必要なプロパティを定義します。

ヒント プール定義を作成し、次にコピー、ペースト、および編集して、複数の JDBC データソースを設定することができます。

サブ要素

次の表は、jdbc-connection-pool 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

jdbc-connection-pool のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、jdbc-connection-pool 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

jdbc-connection-pool の属性

属性	デフォルト値	説明
name	なし	接続プールの名前を指定する。 jdbc-resource 要素の pool-name 属性は、この name に指定された値を参照する
datasource-classname	なし	ベンダーが提供するデータソースに関連付けられたクラス名を指定する。このクラスは、 java.sql.DataSource、 java.sql.XADataSource、あるいはその両方を実装していることが必要

jdbc-connection-pool の属性 (続き)

属性	デフォルト値	説明
res-type	javax.sql.DataSource	(省略可能) データソースクラスが実装するインタフェースを指定する。この属性の値として、javax.sql.DataSource または javax.sql.XADataSource を指定することができる。値がこれらのインタフェースのいずれでもない場合は、デフォルトが使用される。この属性に有効な値が指定されていても、指定されたインタフェースがデータソースクラスに実装されていない場合は、エラーが発生する
steady-pool-size	8	(省略可能) プールに保持される接続の初期数および最小数を指定する
max-pool-size	32	(省略可能) クライアントの要求に対応するために作成できる接続の最大数を指定する
max-wait-time-in-millis	60000	(省略可能) 呼び出し側が接続を取得するまで待機する時間をミリ秒単位で指定する。0 の場合、呼び出し側はリソースが利用できるようになるまで、またはエラーが発生するまで待機し続ける
pool-resize-quantity	2	(省略可能) 既存の接続の数が steady-pool-size (max-pool-size による制限を受ける) を超える場合に破棄される接続の数を指定する。これは、idle-time-out-in-seconds 間隔で定期的に行われる。アイドル状態の接続は、idle-time-out-in-seconds の間使用されない接続である
idle-timeout-in-seconds	300	(省略可能) プールで接続がアイドル状態のままにいられる最長時間を指定する。この時間を過ぎると、プールはこの接続を閉じる
transaction-isolation-level	デフォルトの JDBC ドライバ遮断レベル	(省略可能) プールされた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。有効な値は read-uncommitted、read-committed、repeatable-read、serializable プールされた接続の遮断レベルをアプリケーションでプログラマ的に変更すると、プールのしくみに支障をきたす場合があり、エラーが発生することがある。詳細については、is-isolation-level-guaranteed を参照

jdbc-connection-pool の属性 (続き)

属性	デフォルト値	説明
is-isolation-level-guaranteed	true	(省略可能) transaction-isolation-level が明示的に設定されているときに限り適用される。true の場合、プールから取得されたすべての接続は希望する遮断レベルを持つことができる。これは、いくつかの JDBC ドライバのパフォーマンスに影響を与えることがある。ホストされているアプリケーションが変更された遮断レベルで接続を返さないことが確かな場合は、この属性を false に設定できる
is-connection-validation-required	false	(省略可能) アプリケーションに対して接続が確立される前に接続を検証する必要があるかどうかを指定する。リソースの検証が失敗すると破棄され、新しいリソースが作成されて返される。有効な値は on、off、yes、no、1、0、true、false
connection-validation-method	auto-commit	(省略可能) 有効な値は次のとおり <ul style="list-style-type: none"> • auto-commit (デフォルト)。 Connection.setAutoCommit (Connection.getAutoCommit ()) を使用する • meta-data。 Connection.getMetaData () を使用する • table。validation-table-name 属性で指定されているテーブルでクエリを実行する
validation-table-name	なし	(省略可能) 接続を検証するクエリを実行するために使用するテーブル名を指定する。connection-validation-type が table に設定されている場合に限り、このパラメータは必須である
fail-all-connections	false	(省略可能) true の場合、1 つでも検証が失敗するとプールのすべての接続が閉じられる。is-connection-validation-required が true に設定されている場合に限り、このパラメータは必須である。有効な値は on、off、yes、no、1、0、true、false

プロパティ

ほとんどの JDBC 2.0 ドライバは、標準のプロパティリストを使って、ユーザー、パスワード、およびその他のリソース設定情報を指定することができます。Sun ONE Application Server ではプロパティはオプションですが、中にはほとんどのデータベースで必要なプロパティもあります。詳細は、『JDBC 2.0 Standard Extension API』の第 5.3 節を参照してください。

プロパティが指定されているときは、`setName(value)` メソッドを使って、`datasource-classname` 属性で指定されたベンダーのデータソースクラスに渡されます。

コンテナ管理認証が指定されていて、`default-resource-principal` がアプリケーション配備記述子内がない場合、`user` および `password` プロパティがデフォルトの主体として使われます。

次の表は、`jdbc-connection-pool` 要素の共通プロパティを示します。左の列にはプロパティ名、右の列にはプロパティの説明を示しています。

jdbc-connection-pool プロパティ

プロパティ	説明
<code>user</code>	接続プールのユーザー名を指定する
<code>password</code>	接続プールのパスワードを指定する
<code>databaseName</code>	接続プールのデータベースを指定する
<code>serverName</code>	接続プールのデータベースサーバーを指定する
<code>port</code>	データベースサーバーが要求を待機するポートを指定する
<code>networkProtocol</code>	通信プロトコルを指定する
<code>roleName</code>	最初の SQL ロール名を指定する
<code>datasourceName</code>	基礎となる XADataSource を指定するか、または、接続プールが実行されている場合は ConnectionPoolDataSource を指定する
<code>description</code>	説明テキストを指定する
<code>url</code>	接続プールの URL を指定する。標準のプロパティではないが、広く使用される

アプリケーション要素

Application 要素は次のとおりです。

- applications
- lifecycle-module
- j2ee-application
- web-module
- ejb-module
- connector-module

applications

配備された J2EE アプリケーション、J2EE モジュール、およびライフサイクルモジュールを含む

サブ要素

次の表は、applications 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

applications のサブ要素

要素	必要指定数	説明
lifecycle-module	0 または 1 個以上	配備されたライフサイクルモジュールを指定する
j2ee-application	0 または 1 個以上	配備された J2EE アプリケーションを指定する
ejb-module	0 または 1 個以上	配備された EJB モジュールを指定する
web-module	0 または 1 個以上	配備された Web モジュールを指定する
connector-module	0 または 1 個以上	配備されたコネクタモジュールを指定する

注 applications 要素のサブ要素は任意の順序で記述できます。

属性

次の表は、applications 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

applications の属性

属性	デフォルト値	説明
dynamic-reload-enabled	false	(省略可能) 動的再読み込みを有効にするかどうかを指定する。この属性は、開発環境では true に設定し、本稼動環境では false に設定することが必要である。有効な値は on、off、yes、no、1、0、true、false
dynamic-reload-poll-interval-in-seconds	2	(省略可能) アプリケーションとモジュールのコードが変更されていないかを確認し、動的再読み込みする間隔を指定する

lifecycle-module

配備されたライフサイクルモジュールを指定します。ライフサイクルモジュールの詳細は、『Sun ONE Application Server 開発者ガイド』を参照してください。

サブ要素

次の表は、lifecycle-module 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

lifecycle-module のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む
property	0 または 1 個以上	プロパティまたは変数を指定する

属性

次の表は、lifecycle-module 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

lifecycle-module の属性

属性	デフォルト値	説明
name	なし	ライフサイクルモジュールの名前

lifecycle-module の属性 (続き)

属性	デフォルト値	説明
class-name	なし	ライフサイクルモジュールのクラスファイルの完全指定名。 com.sun.appserv.server.LifecycleListener インタ フェースを実装していることが必要である
classpath	server 要素の application- root 属性の値	(省略可能) ライフサイクルモジュールのクラスパス。モジュー ルの位置を指定する
load-order	なし	(省略可能) 起動時にどのライフサイクルモジュールを読み込む かどうかを決定する。モジュールに指定された数値が小さいほ ど、早く読み込まれる。値の範囲は 101 ~ オペレーティング システムの MAXINT まで。1 ~ 100 までの値は予約されている
is-failure-fatal	false	(省略可能) ライフサイクルモジュールが失敗したときにサー バーをシャットダウンするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false
enabled	true	(省略可能) ライフサイクルモジュールを有効にするかどうかを 決定する。有効な値は on、off、yes、no、1、0、true、 false

j2ee-application

配備された J2EE アプリケーションを指定します。

サブ要素

次の表は、j2ee-application 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

j2ee-application のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む

属性

次の表は、j2ee-application 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

j2ee-application の属性

属性	デフォルト値	説明
name	なし	アプリケーションの名前
location	なし	Sun ONE Application Server ファイルシステムでのアプリケーションの位置
virtual-servers	すべての仮想サーバー	(省略可能) このアプリケーション内の Web モジュールが配備される仮想サーバー
enabled	true	(省略可能) アプリケーションを有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

ejb-module

配備された EJB モジュールを指定します。

サブ要素

次の表は、ejb-module 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

ejb-module のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む

属性

次の表は、ejb-module 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

ejb-module の属性

属性	デフォルト値	説明
name	なし	EJB モジュールの名前
location	なし	Sun ONE Application Server ファイルシステムでの EJB モジュールの位置
enabled	true	(省略可能) EJB モジュールを有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

web-module

配備された Web モジュールを指定します。

サブ要素

次の表は、web-module 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

web-module のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む

属性

次の表は、web-module 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

web-module の属性

属性	デフォルト値	説明
name	なし	Web モジュールの名前
context-root	空文字列	Web モジュールをインストールするコンテキストルート (先頭に / が付かないコンテキストパス) サブレット 2.3 仕様書の第 5.4 節を参照 この属性が空文字列 (デフォルト値) の場合、この Web モジュールが、virtual-servers 属性で指定される virtual-server 要素のデフォルトの Web モジュールになる 仮想サーバーに割り当てられた複数の Web モジュールが空のコンテキストルートを持つ場合、それらの 1 つだけが読み込まれ、残りに対しては「Virtual server [virtual_server] already has a web module loaded at [/]」などのエラーメッセージが生成される
location	なし	.war ファイルの内容を抽出したディレクトリへの完全修飾パスまたは相対パス。相対パスの場合、次のディレクトリと相対関係になる instance_dir/applications/j2ee-modules/
virtual-servers	すべての仮想サーバー	(省略可能) Web モジュールが配備される virtual-server 要素

web-module の属性 (続き)

属性	デフォルト値	説明
enabled	true	(省略可能) Web モジュールを有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

connector-module

配備されたコネクタモジュールを指定します。

サブ要素

次の表は、connector-module 要素のサブ要素を示します。左の列にはサブ要素名、中央の列には必要指定数、右の列には要素の説明を示しています。

connector-module のサブ要素

要素	必要指定数	説明
description	0 または 1 個	この要素を説明するテキストを含む

属性

次の表は、connector-module 要素の属性を示します。左の列に属性の名前、中央の列にデフォルト値、右の列にその属性の説明を示します。

connector-module の属性

属性	デフォルト値	説明
name	なし	コネクタモジュールの名前
location	なし	Sun ONE Application Server ファイルシステムでのコネクタモジュールの位置
enabled	true	(省略可能) コネクタモジュールを有効にするかどうかを決定する。有効な値は on、off、yes、no、1、0、true、false

ユーザーデータベースの選択

注 ユーザーデータベースはサーバー自体のセキュリティのみに適用します。J2EE アプリケーションおよびモジュールのセキュリティには関係ありません。

`server.xml` の `auth-db` 要素は、次のように親 `virtual-server` 要素のユーザーデータベースを選択します。

- `auth-db` 要素の `id` 属性は、ACL ファイルの `database` 属性にマップする
- `auth-db` 要素の `database` 属性は、`dbswitch.conf` エントリにマップする

ACL ファイルと `dbswitch.conf` ファイル間のこのレイヤは、サーバー管理者に、仮想サーバーの管理者とユーザーがアクセスするデータベースを完全に制御する権限を与えます。

注 `dbswitch.conf` インタフェースは、不確定です。不確定なインタフェース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。

`dbswitch.conf` ファイルは、次のように LDAP データベースの検索ツリーのルートを確認します。

- `dbswitch.conf` にある LDAP URL の基本 DN は、すべての詳細な DN 仕様のルートオブジェクトを定義します。最新のインストールでは、基本 DN が空になっていることがあります。それは、最終の基本 DN が、ほかの方法、つまり DC ツリー検索または `auth-db` 要素の明示的な `basedn` 値のいずれかで決定されるためです。
- LDAP データベースの `dbswitch.conf` 属性である `dcsuffix` は、DC ツリーのルートを定義します。このルートは、LDAP URL の基本 DN と相対関係になります。データベースがスキーマに準拠している場合、`dcsuffix` を使うことができます。スキーマ準拠の要件は、89 ページの「Sun ONE LDAP スキーマ」のリストに記載されています。

仮想サーバー用のユーザーデータベースは、次のように選択されます。

- `virtual-server` に `auth-db` サブ要素がない場合、ユーザーベースまたはグループベースの ACL は失敗します。

- 仮想サーバーの ACL 定義に database 属性が存在しないときは、virtual-server は default の id 属性を伴う auth-db サブ要素を持っている必要があります。auth-db の database 属性は、dbswitch.conf のデータベースを指します。database 属性が存在しない場合は、default が使われます。
- LDAP データベースがスキーマに準拠している場合、クライアントの提供する Host ヘッダーと一致する virtual-server 要素の host 属性の DC ツリー検索を使って、アクセスの基本 DN が計算されます。host 属性が 1 つも一致しない場合、親 http-listener の server-name 属性が使われます。DC ツリー検索は dcsuffix DN に基づきます。検索結果には、基本 DN を含む inetDomainBaseDN 属性が含まれている必要があります。この基本 DN は、そのまま取得され、いずれの基本 DN 値とも関連がありません。
- auth-db 要素の basedn 属性が存在せず、データベースがスキーマに準拠していない場合、Application Server の前のバージョンでの場合と同様に、アクセスは dbswitch.conf エントリの基本 DN と関連して行われます。

Sun ONE LDAP スキーマ

LDAP データベースがこの節で説明した要件を満たす場合、dbswitch.conf ファイルにある dcsuffix 属性を使うことができます。

`o=isp` のように ISP エントリをルートとするサブツリーは、収束ツリーと呼ばれます。このツリーには、ISP が提供する、組織 (顧客) に関連するすべてのディレクトリデータが含まれます。

`o=internet` をルートとするサブツリーは、ドメインコンポーネントツリーまたは dc ツリーと呼ばれます。このサブツリーは、顧客ドメインのエントリを持つ簡単な DNS ツリーを含んでいます。これらのエントリは、そのドメインのデータがある、収束ツリーの適切な位置へのリンクになっています。

ディレクトリツリーは、ルートが 1 つだけの場合 (推奨) と、異なる 2 つのルート (収束ツリー用と dc ツリー用) がある場合があります。ルートが 1 つというのは、たとえば、`o=root` の下に `o=isp` と `o=internet` がある場合です。

収束ツリー

収束ツリーの最上位レベルには、組織のエントリが、各顧客 (または組織) ごとに 1 つ、そして ISP 自体に 1 つ必要です。

各組織の下には、2 つの `organizationalUnit` エントリ (`ou=People` および `ou=Groups`) が必要です。組織のデバイスデータを格納する場合、3 番目の `ou=Devices` が表示されます。

各ユーザーエントリーは、指定した組織内で一意の uid 値を持っている必要があります。このサブツリーの下にネーム空間をさまざまな ou エントリーに区分し、ユーザーエントリーを ou=eng や ou=corp のように便利なグループにまとめることができます。その場合でも、ユーザーの uid 値は People サブツリー全体で一意である必要があります。

収束ツリーのユーザーエントリーは、inetOrgPerson タイプです。cn、sn、および uid 属性が必要です。uid 属性は、有効な電子メール名 (特に、RFC822 で定義されている有効なローカルパート) である必要があります。cn には名前のイニシャル sn を含め、ユーザーエントリーの RDN は uid 値にすることをお勧めします。ユーザーエントリーを有効またはサービスで有効と見なされるようにするには、AUXILIARY クラスの inetUser を含める必要があります。

ユーザーエントリーは、アカウントの管理目的に使われる AUXILIARY クラス inetSubscriber も含みます。inetUseStatus 属性がエントリーにあり、その値が inactive または deleted の場合、エントリーは無視されます。

グループは Groups サブツリーの下に配備され、groupOfUniqueNames タイプの LDAP エントリーで構成されます。

ドメインコンポーネント (dc) ツリー

dc ツリーには、階層構造の domain エントリーが含まれていて、それぞれが DNS 名コンポーネントになっています。

ユーザーのドメイン名を表すエントリーは、LDAP の AUXILIARY クラス inetDomain でオーバーレイされます。たとえば、2 つの LDAP エントリー dc=customer1,dc=com,o=Internet,o=root および dc=customer2,d=com,o=Internet,o=root は inetDomain クラスを含みますが、dc=com,o=Internet,o=root は含みません。後者は、ツリーに構造を持たせるためにのみ存在します。

inetDomain 属性を含むエントリーは、仮想ドメインと呼ばれます。これらのエントリーは、最上位レベルの組織エントリーの DN が入力されている inetDomainBaseDN 属性を持っている必要があります。最上位レベルの組織エントリーは収束ツリーでこのドメインのデータが格納されている場所です。たとえば、dc=cust2,dc=com,o=Internet,o=root の仮想ドメインエントリーは、o=Cust2,O=root の値を持つ inetDomainBaseDN 属性を含みます。

inetDomainStatus 属性がエントリーにあり、その値が inactive または deleted の場合、エントリーは無視されます。

変数

変数の中には、obj.conf ファイルで使うために server.xml で定義されているものがあります。次のファイルコードは docroot 変数を定義します。

```
<property name="docroot" value="/server/docs/class2/acme" />
```

docroot 変数を使うと、異なるドキュメントルートディレクトリを異なる仮想サーバーに割り当てることができます。この変数は、obj.conf ファイルで使われます。次に例を示します。

```
NameTrans fn=document-root root="$docroot"
```

この docroot 変数を使うと、同じ仮想サーバークラス内で異なるドキュメントルートを異なる仮想サーバーに定義することができます。

注 変数は、http-service、virtual-server-class、または virtual-server レベルで、server.xml ファイルに定義する必要があります。http-service または virtual-server-class レベルでデフォルト値を持つ変数を定義し、virtual-server レベルでそれらの変数をオーバーライドすることをお勧めします。

変数の形式

次の正規表現と一致するとき、変数は obj.conf にあります。

```
¥$ [A-Za-z] [A-Za-z0-9_]*
```

この表現は、\$ の後に 1 つまたは複数のアルファベットおよび数値文字が続いていることを表します。\${property} のような区切り形式はサポートされていません。正規の \$ 文字を取得するには、\$\$ を使って変数を置き換えます。

id 変数

特殊な変数 id は、virtual-server 要素内で常に利用でき、id 属性の値を参照します。この変数は事前に定義されているのでオーバーライドすることはできません。id 属性は仮想サーバーを一意に識別します。次に例を示します。

```
<property name="docroot" value="/export/$id" />
```

親 virtual-server を含む id 属性が myserver の場合、docroot 変数の値は /export/myserver に設定されます。

その他の重要な変数

次の変数は、Sun ONE Application Server 設定のさまざまな部分で使われます。`$sid` 変数とは異なり、これらの変数はサーバーで事前に定義されていないので、オーバーライドすることができます。

一般的な変数

次の表は、一般的な `server.xml` 変数を示しています。左の列に変数、右の列にそれらの変数の説明を示しています。

一般的な変数

変数	説明
<code>docroot</code>	仮想サーバーのドキュメントルート。通常は、 <code>obj.conf</code> ファイル内の <code>document-root</code> 関数へのパラメータとして評価される
<code>accesslog</code>	仮想サーバーのアクセスログファイル

send-cgi 変数

次の表は、`obj.conf` ファイルの `send-cgi` 関数で使われる `server.xml` 変数の一覧です。左の列に変数、右の列にそれらの変数の説明を示しています。

send-cgi 変数

変数	説明
<code>user</code>	<code>user</code> CGI パラメータの値
<code>group</code>	<code>group</code> CGI パラメータの値
<code>chroot</code>	<code>chroot</code> CGI パラメータの値
<code>dir</code>	<code>dir</code> CGI パラメータの値
<code>nice</code>	<code>nice</code> CGI パラメータの値

`send-cgi` 関数の詳細は、『Sun ONE Application Server Developer's Guide to NSAPI』を参照してください。

変数の評価

変数は各仮想サーバー固有の **objsets** を生成するときに評価されます。評価は再帰的に行われるので、変数の値にほかの変数を含めることができます。次に例を示します。

```

...
<virtual-server-class>
  ...
  <virtual-server ...>
    ...
    <property name=docroot value="$docrootbase/nonjava/$id" />
  </virtual-server>
  <virtual-server ...>
    ...
    <property name=docroot value="$docrootbase/java/$id" />
  </virtual-server>
  ...
  <property name=docrootbase value="/export" />
</virtual-server-class>
...

```

サブ要素内の変数は、親要素内の変数をオーバーライドします。たとえば、仮想サーバークラスの変数を設定し、その変数を、各仮想サーバー内の同じ変数の定義でオーバーライドすることができます。

server.xml ファイルのサンプル

Sun ONE Application Server の Windows バージョンの最初のインストール時には、server.xml ファイルは次のようになっています。

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- server_7_0_dtd version: 1.1.2.39 -->
<!DOCTYPE server PUBLIC "-//Sun Microsystems Inc.//DTD Sun ONE Application Server 7.0//EN" "file:///C:/Sun/AppServer7/lib/dtds/sun-server_1_0.dtd">
<server name="server1" log-root="C:/Sun/AppServer7/domains/domain1/server1/logs"
application-root="C:/Sun/AppServer7/domains/domain1/server1/applications"
session-store="C:/Sun/AppServer7/domains/domain1/server1/session-store">
  <http-service qos-enabled="false">
    <http-listener id="http-listener-1" address="0.0.0.0" port="1024"
acceptor-threads="1" blocking-enabled="false" security-enabled="false"
default-virtual-server="server1" server-name="goliath" enabled="true"/>
    <mime id="mime1" file="mime.types"/>

```

```
<acl id="acl1"
file="C:/Sun/AppServer7/domains/domain1/server1/config/generated.server1.acl"/>
  <virtual-server-class id="defaultclass" config-file="obj.conf"
default-object="default" accept-language="false" enabled="true">
    <virtual-server id="server1" http-listeners="http-listener-1"
hosts="goliath" mime="mime1" state="on" accept-language="false">
      <auth-db id="default" database="default"/>
      <property name="dir" value=""/>
      <property name="nice" value=""/>
      <property name="user" value=""/>
      <property name="group" value=""/>
      <property name="chroot" value=""/>
      <property name="docroot"
value="C:/Sun/AppServer7/domains/domain1/server1/docroot"/>
      <property name="accesslog"
value="C:/Sun/AppServer7/domains/domain1/server1/logs/access"/>
    </virtual-server>
  </virtual-server-class>
</http-service>
<iiop-service>
  <orb message-fragment-size="1024" steady-thread-pool-size="10"
max-thread-pool-size="200" idle-thread-timeout-in-seconds="300"
max-connections="1024" monitoring-enabled="false"/>
  <iiop-listener id="orb-listener-1" address="goliath" port="3700"
enabled="true">
    </iiop-listener>
  </iiop-service>
</web-container monitoring-enabled="false" >
</web-container>
  <ejb-container steady-pool-size="32" pool-resize-quantity="16"
max-pool-size="64" cache-resize-quantity="32" max-cache-size="512"
pool-idle-timeout-in-seconds="600" cache-idle-timeout-in-seconds="600"
removal-timeout-in-seconds="5400" victim-selection-policy="nru" commit-option="B"
monitoring-enabled="false">
  </ejb-container>
```

```
<mdb-container steady-pool-size="10" pool-resize-quantity="2"
max-pool-size="60" idle-timeout-in-seconds="600" monitoring-enabled="false">
  </mdb-container>

  <jms-service port="7676" admin-user-name="admin" admin-password="admin"
init-timeout-in-seconds="30" enabled="true">
    </jms-service>

    <log-service file="C:/Sun/AppServer7/domains/domain1/server1/logs/server.log"
level="INFO" log-stdout="true" log-stderr="true"
echo-log-messages-to-stderr="true" create-console="true"
log-virtual-server-id="false" use-system-logging="false">
    </log-service>

    <security-service default-realm="file" anonymous-role="ANYONE"
audit-enabled="false">
      <auth-realm name="file"
classname="com.iplanet.ias.security.auth.realm.file.FileRealm">
        <property name="file"
value="C:/Sun/AppServer7/domains/domain1/server1/config/keyfile"/>
        <property name="jaas-context" value="fileRealm"/>
      </auth-realm>

      <auth-realm name="ldap"
classname="com.iplanet.ias.security.auth.realm.ldap.LDAPRealm">
        <property name="directory" value="ldap://localhost:389"/>
        <property name="base-dn" value="o=isp"/>
        <property name="jaas-context" value="ldapRealm"/>
      </auth-realm>

      <auth-realm name="certificate"
classname="com.iplanet.ias.security.auth.realm.certificate.CertificateRealm">
    </auth-realm>

      <!--
      <auth-realm name="solaris"
classname="com.iplanet.ias.security.auth.realm.solaris.SolarisRealm">
        <property name="jaas-context" value="solarisRealm"/>
      </auth-realm>

      -->
    </security-service>
```

```

    <transaction-service automatic-recovery="false" timeout-in-seconds="0"
tx-log-dir="C:/Sun/AppServer7/domains/domain1/server1/logs"
heuristic-decision="rollback" keypoint-interval="2048"
monitoring-enabled="false">

    </transaction-service>

    <java-config java-home="C:/Sun/AppServer7/jdk"
server-classpath="C:/Sun/AppServer7/lib/appserv-rt.jar;C:/Sun/AppServer7/jdk/lib/
tools.jar;C:/Sun/AppServer7/lib/appserv-ext.jar;C:/Sun/AppServer7/lib/appserv-cmp
.jar;C:/Sun/AppServer7/lib/appserv-ideplugin.jar;C:\Sun\%AppServer7%\imq\lib\imq.ja
r;C:\Sun\%AppServer7%\imq\lib\jaxm-api.jar;C:\Sun\%AppServer7%\imq\lib\imqadmin.jar;C:
\Sun\%AppServer7%\imq\lib\imqutil.jar;C:/Sun/AppServer7/lib/appserv-admin.jar;C:\Sun
\%AppServer7%\share\lib\jaxrpc-impl.jar;C:\Sun\%AppServer7%\share\lib\jaxrpc-api.jar;C
:\Sun\%AppServer7%\share\lib\jaxr-impl.jar;C:\Sun\%AppServer7%\share\lib\jaxr-api.jar;
C:\Sun\%AppServer7%\share\lib\jhall.jar;C:\Sun\%AppServer7%\share\lib\activation.jar;C
:\Sun\%AppServer7%\share\lib\mail.jar;C:\Sun\%AppServer7%\share\lib\saaj-api.jar;C:\Su
n\%AppServer7%\share\lib\saaj-impl.jar;C:\Sun\%AppServer7%\share\lib\commons-logging.j
ar;C:\Sun\%AppServer7%\imq\lib\fscontext.jar;C:\Sun\%AppServer7%\imq\lib/providerutil.
jar;C:/Sun/AppServer7/lib/appserv-jstl.jar"
classpath-suffix="C:/Sun/AppServer7/pointbase/client_tools/lib/pbclient42RE.jar"
env-classpath-ignored="true" debug-options="-Xdebug
-Xrunjdw:transport=dt_socket,server=y,suspend=n" debug-enabled="false"
javac-options="-g">

        <jvm-options>-Dorg.xml.sax.parser=org.xml.sax.helpers.XMLReaderAdapter
</jvm-options>

        <jvm-options>-Dorg.xml.sax.driver=org.apache.crimson.parser.XMLReaderImpl
</jvm-options>

<jvm-options>-Djava.security.policy=C:/Sun/AppServer7/domains/domain1/server1/con
fig/server.policy</jvm-options>

<jvm-options>-Djava.security.auth.login.config=C:/Sun/AppServer7/domains/domain1/
server1/config/login.conf</jvm-options>

<jvm-options>-Dcom.sun.jdo.api.persistence.model.multipleClassLoaders=reload
</jvm-options>

<jvm-options>-Djava.util.logging.manager=com.ipplanet.ias.server.logging.ServerLog
Manager</jvm-options>

<jvm-options>-Dcom.sun.aas.configRoot=C:/Sun/AppServer7\config</jvm-options>

        <jvm-options>-Dcom.sun.aas.imqLib=C:\Sun\%AppServer7%\imq\lib</jvm-options>
        <jvm-options>-Dcom.sun.aas.imqBin=C:\Sun\%AppServer7%\imq\bin</jvm-options>

<jvm-options>-Dcom.sun.aas.webServicesLib=C:\Sun\%AppServer7%\share\lib</jvm-option
s>

```



```
<jvm-options>-Dsun.rmi.dgc.server.gcInterval=300000</jvm-options>
<jvm-options> -Xrs -Xms128m -Xmx256m </jvm-options>
</java-config>
<resources>
</resources>
<applications dynamic-reload-enabled="false"
dynamic-reload-poll-interval-in-seconds="2">
  </applications>
</server>
```

server.xml ファイルのサンプル

init.conf の構文と使い方

Sun ONE Application Server は起動時に、*instance_dir/config* ディレクトリの *init.conf* というファイルを検索して、サーバーインスタンスの動作と設定に関するグローバル変数セットを設定します。Sun ONE Application Server は *init.conf* で定義した指令をすべて実行します。

Init 関数を除く *init.conf* の指令では変数と値を 1 つずつ指定します。たとえば次のように指定します。

```
TempDir /tmp
```

指令の順序は重要ではありません。

注 ブール値の指令には次のような等価の値があります。on/yes/true と off/no/false。

注 *init.conf* ファイルを編集した場合、サーバーを再起動して変更内容を反映させる必要があります。

注 *init.conf* インタフェースは、不確定です。不確定なインタフェース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。

この章では、Sun ONE Application Server 7 の `init.conf` で指定できるグローバル設定について示します。

この章には次の節があります。

- Init 関数
- サーバー情報
- DNS ルックアップ
- スレッド、プロセス、および接続
- ネイティブのスレッドプール
- CGI
- エラー記録
- ACL
- セキュリティ
- チャンクされたエンコード
- その他

指令のアルファベット順のリストは、付録 C 「`init.conf` の指令のアルファベット順リスト」を参照してください。

注 ほとんどのファイルキャッシュ機能は `nsfc.conf` という設定ファイルで制御されます。`nsfc.conf` については、131 ページの「`nsfc.conf`」と『Sun ONE Application Server 7 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。

Init 関数

Init 関数は、サーバーモジュールとプラグインを読み込んで初期化し、ログファイルを初期化します。これらの関数の詳細については、『Sun ONE Application Server Developer's Guide to NSAPI』を参照してください。

サーバー情報

この節では、サーバーに関する情報を指定する `init.conf` の指令について示します。次の指令があります。

- `NetSiteRoot`
- `TempDir`
- `TempDirSecurity`
- `User` (UNIX のみ)

NetSiteRoot

サーバーの `bin` ディレクトリと `lib` ディレクトリがある最上位ディレクトリへの絶対パス名を指定します。デフォルト値はないので値を指定する必要があります。

構文

`NetSiteRoot path`

TempDir

サーバーが一時ファイルとして使用するローカルボリュームのディレクトリを指定します。UNIX では、サーバーを実行するユーザーがそのディレクトリを所有し、書き込める必要があります。「`User (UNIX のみ)`」指令および「`TempDirSecurity`」指令も参照してください。

構文

`TempDir path`

デフォルト値

`/tmp` (UNIX)

`TEMP` (Windows の環境変数)

TempDirSecurity

TempDir ディレクトリのセキュリティをサーバーがチェックするかどうかを指定します。UNIX では TempDirSecurity をオフ (TempDirSecurity off) にすると、サーバーは /tmp を一時ディレクトリとして使用します。

警告 UNIX では TempDirSecurity off を指定したり、/tmp を一時ディレクトリとして使用しないでください。/tmp を一時ディレクトリとして使用すると、セキュリティ上のリスクが大きくなります。

構文

TempDirSecurity [on|off]

デフォルト値

on

User (UNIX のみ)

User 指令ではサーバーの UNIX ユーザーアカウントを指定します。サーバーをスーパーユーザーまたはルートユーザーとして起動すると、指定したポートがバインドされ、ユーザー ID が User 指令で指定したユーザーアカウントに変更されます。サーバーを root で起動しない場合、この指令は無視されます。指定するユーザーアカウントには、サーバールートおよびサブディレクトリへの読み込みアクセス権を与える必要があります。ユーザーアカウントには、ログディレクトリへの書き込みアクセス権と CGI プログラムの実行権限を与える必要があります。ユーザーアカウントには設定ファイルに対する書き込み権を与えないようにします。これは、たとえ第三者がサーバーに侵入しても、設定ファイルを変更してマシンへアクセスすることを防ぐためです。nobody ユーザーは使用できますが、お勧めしません。

構文

User *name*

name は 8 文字以下の UNIX ユーザーアカウントのログイン名です。

デフォルト値

User 指令がなければ、サーバーは起動時のユーザーアカウントで実行します。

例

User http

User server

User nobody

DNS ルックアップ

この節では、DNS ルックアップに関係がある `init.conf` の指令を示します。次の指令があります。

- AsyncDNS
- DNS

AsyncDNS

非同期 DNS を使用するかどうかを指定します。この指令を有効にするには、DNS 指令を `on` に設定する必要があります。値は `on` または `off` です。DNS が有効であれば、非同期 DNS を有効にするとサーバーのパフォーマンスが向上します。

デフォルト値

デフォルトは `off` です。

DNS

DNS 指令では、サーバーにアクセスするクライアント上でサーバーが DNS ルックアップを実行するかどうかを指定します。クライアントがサーバーに接続すると、サーバーはクライアントの IP アドレスを認識しますが、クライアントのホスト名を認識しません (たとえばサーバーはクライアントを `198.95.251.30` として認識できてもホスト名の `www.a.com` を認識できません)。サーバーはクライアントの IP アドレスからホスト名を解決し、アクセスコントロール、CGI、エラーレポート、アクセスログなどを実行します。

サーバーが 1 日に応答する要求が多ければ、ホスト名の解決を停止して DNS サーバーや NIS サーバーの負荷を減らすこともできます。

構文

DNS [on|off]

デフォルト値

デフォルトの DNS ホスト名の解決は `on` です。

例

DNS on

スレッド、プロセス、および接続

Sun ONE Application Server 7 では、待機ソケットの受け入れスレッドは接続すると接続キューで待機します。次にセッションスレッドがキューから接続を受け取って要求を処理します。セッションスレッドは、必要に応じてさらに多くのセッションスレッドを要求の最後に登録します。接続キューが次のような状態の場合に新しいスレッドが追加されます。

- 新しい接続が返されるたびに、キューで待機している接続数 (接続のバックログ) と作成済みセッションスレッドの数が比較される。待機している接続数が作成済みスレッドの数よりも多ければ、次の要求完了後スレッドがさらに追加される
- 前のバックログが追跡される。バックログが時間の経過とともに増えていて、その増え方が `ThreadIncrement` の値よりも大きく、セッションスレッドからバックログを引いた数が `ThreadIncrement` の値よりも小さければ、`ThreadIncrement` で指定したスレッド数が新たに追加される
- 新しくセッションスレッドを追加する過程が `RqThrottle` の値で厳密に制限されている
- ベンチマーク負荷の開始などによってバックログが急に増えても、作成されるスレッドが多くなるようにするために、16 回または 32 回ごとの接続時にスレッドが必要かどうかが決まる。この回数は既に存在するセッションスレッドの数によって決まる

この節では、スレッド、プロセスおよび接続数とそれぞれのタイムアウトに関係がある `init.conf` の指令を示します。次の指令があります。

- `ConnQueueSize`
- `HeaderBufferSize`
- `IOTimeout`
- `KeepAliveThreads`
- `KeepAliveTimeout`
- `KernelThreads` (Windows のみ)
- `ListenQ`
- `MaxKeepAliveConnections`
- `PostThreadsEarly`
- `RcvBufSize`
- `RqThrottle`
- `RqThrottleMin`
- `SndBufSize`

- StackSize
- StrictHttpHeaders
- TerminateTimeout
- ThreadIncrement

ネイティブスレッドのプールを制御する指令については、109 ページの「ネイティブのスレッドプール」も参照してください。

ConnQueueSize

アプリケーションサーバーが持てる未処理の接続数 (サービスが実行されていない接続数) を指定します。この値は、オペレーティングシステムで規定されている 1 プロセス当たりのオープンファイル記述子の最大値を超えるように設定することをお勧めします。

デフォルト値

デフォルト値は 5000 です。

HeaderBufferSize

各要求スレッドがクライアントからの要求データの読み取りに使用できるバッファサイズをバイト単位で指定します。要求処理スレッドの最大値は RqThrottle で設定します。

デフォルト値

デフォルト値は 8192 (8 KB) です。

IOTimeout

クライアントからのデータが到着するのをサーバーが待つ時間を秒で指定します。時間内にデータが到着しない場合は、接続が失われます。デフォルトの 30 秒より短く設定すると、スレッドを早く解放できます。ただし、接続速度が遅いユーザーの接続も切斷してしまうことになります。

構文

IOTimeout *seconds*

デフォルト値

ハードウェア暗号化デバイスを使用しない場合は 30 秒、暗号化デバイスを使用する場合は 300 秒です。

KeepAliveThreads

この指令では、キープアライブサブシステムのスレッド数を指定します。スレッド数はシステムのプロセッサの最小倍数にすることをお勧めします。たとえば、CPU が 2 つあるシステムでは、キープアライブのスレッド数は 2 または 4 にします。この値は、キープアライブ接続の最大許容数 (MaxKeepAliveConnections) も考慮して設定する必要があります。

デフォルト値

1

KeepAliveTimeout

この指令では、サーバーが HTTP キープアライブ接続をオープンにしている最大時間、またはクライアントとサーバー間の接続が持続している最大時間を指定します。以前のバージョンでは、サーバーがクライアント要求を処理している間はキープアライブ機能によってクライアントとサーバー接続をオープンにしておくことができました。デフォルトの接続は、持続接続 (サーバーが接続を閉じるまではオープン) であるか、または KeepAliveTimeout で指定した時間よりも長く接続がオープンになっています。

接続がキープアライブサブシステムに移ると、タイムアウトへの秒読みが開始されます。タイムアウトまでに接続でアクティビティがない場合は、接続が閉じられます。

デフォルト値

デフォルト値は 30 秒です。最大値は 300 秒 (5 分) です。

KernelThreads (Windows のみ)

Sun ONE Application Server は、オペレーティングシステムがカーネルレベルのスレッドをサポートしていれば、カーネルレベルとユーザーレベルの両方のスレッドをサポートします。ローカルスレッドはプロセスの NSPR でスケジュールしますが、カーネルスレッドはホストのオペレーティングシステムでスケジュールします。通常、標準のデバグとコンパイラはカーネルレベルのスレッドを対象にしています。KernelThreads を 1 (オン) に設定すると、サーバーではカーネルレベルのスレッドだけが使用され、ユーザーレベルのスレッドは使用されません。KernelThreads を 0 (オフ) に設定すると、サーバーではユーザーレベルのスレッドだけが使用されるので、パフォーマンスが向上することもあります。

デフォルト値

デフォルト値は 0 (オフ) です。

ListenQ

待機ソケットで保留される接続の最大数を指定します。バックログキューがいっぱいになっている待機ソケットがタイムアウトになると接続は失敗します。

デフォルト値

デフォルト値はプラットフォームによって異なります。AIX の場合は 4096、Windows の場合は 200、その多の場合は 128 です。

MaxKeepAliveConnections

サーバーが同時にオープンにできるキープアライブと持続接続の最大数を指定します。0 ~ 32768 の値を取ります。

デフォルト値

256

PostThreadsEarly

この指令が 1 (オン) に設定されていると、接続を受け入れてから要求に対する応答を送信するまでの間、待機ソケットの最小スレッド数が有効かどうかをサーバーがチェックします。データベース接続のように処理に時間がかかる要求をサーバーが扱う場合はこの指令を使用します。

デフォルト値

0 (オフ)

RcvBufSize

ソケットが使用する受信バッファサイズをバイト単位で指定します。使用できる値はオペレーティングシステムによって決まります。

デフォルト値

デフォルト値はオペレーティングシステムによって決まります。通常は 4096 (4K)、または 8192 (8K) がデフォルトです。

RqThrottle

サーバーが同時に扱える要求処理スレッドの最大数を指定します。それぞれの要求は各スレッドで実行されます。

サーバーの設定やパフォーマンスのチューニングについては、『Sun ONE Application Server Performance Tuning, Sizing, and Scaling Guide』で詳しく説明されています。

デフォルト値

128

RqThrottleMin

サーバーの起動時に作成される要求処理スレッドの数を指定します。サーバーの負荷が増すにつれて、作成される要求処理スレッドが多くなります。RqThrottle スレッドの最大数まで作成できます。

デフォルト値

48

SndBufSize

ソケットが使用する送信バッファサイズをバイト単位で指定します。

デフォルト値

デフォルト値はオペレーティングシステムによって決まります。通常は 4096 (4K)、または 8192 (8K) がデフォルトです。

StackSize

要求処理スレッドごとの最大スタックサイズを指定します。

デフォルト値

各マシンに適したマシン固有のスタックサイズです。

StrictHttpHeaders

厳密な HTTP ヘッダーチェックを制御します。厳密な HTTP ヘッダーチェックがオンの場合、サーバーは不適切にヘッダーが重複している接続を拒否します。

構文

```
StrictHttpHeaders [on|off]
```

デフォルト値

off

TerminateTimeout

シャットダウンする前に既存の接続が終了するのをサーバーが待つ時間を指定します。

デフォルト値

30 秒

ThreadIncrement

サーバー負荷の増加に対応するための追加または新規要求処理スレッド数です。要求処理キューに入っている保留接続数がアイドル状態の要求処理数より大きい場合などに追加されます。

サーバーの起動時に `RqThrottleMin` の数だけ要求処理スレッドが作成されます。負荷が増すと `RqThrottle` 指令による要求処理スレッドが作成されるまで `ThreadIncrement` 指令による追加要求処理スレッドが作成されます。

デフォルト値

デフォルト値は 10 です。

ネイティブのスレッドプール

この節では、ネイティブのスレッドプールのサイズを制御する指令を示します。このスレッドプールは、ネイティブの OS レベルのスレッドからのみ構成されています。すべてのスレッドは OS レベルであるため、通常、UNIX ではネイティブのプールは使用されません。UNIX でネイティブプールを使用するとコンテキストの切り替えが必要になり、パフォーマンスオーバーヘッドが少し増えますが、このネイティブプールを `stickyAttach` 効果のローカライズ、リソースの制御と管理、プラグインの単独スレッド動作のエミュレートなどの目的に使用できます。

`stickyAttach` は、`server.xml` ファイルの `java-config` 要素のプロパティとして次のように指定できます。

```
<java-config>
  ...
  <property name="stickyAttach" value="1" >
</java-config>
```

Windows では常にデフォルトのネイティブプールが使用され、Sun ONE Application Server では初期要求処理用のファイバ (ユーザーがスケジュール設定したスレッド) を使用します。Windows でカスタムの追加プールを使用してもオーバーヘッドは増えません。

次の指令があります。

- `NativePoolStackSize`
- `NativePoolMaxThreads`
- `NativePoolMinThreads`
- `NativePoolQueueSize`

NativePoolStackSize

ネイティブスレッドプールのスレッドごとにスタックサイズを指定します。

デフォルト値

0 (オペレーティングシステム固有のデフォルト値)

NativePoolMaxThreads

ネイティブスレッドプールの最大スレッド数を指定します。

デフォルト値

128

NativePoolMinThreads

ネイティブスレッドプールの最小スレッド数を指定します。

デフォルト値

1

NativePoolQueueSize

キューでスレッドプールを待機するスレッド数を指定します。プールのスレッドがすべてビジー状態の場合、次の要求処理スレッドはネイティブプールのスレッドが空くまでキューで待機します。キューが満杯の場合は次の要求処理スレッドがキューに入ろうとしても拒否され、クライアントにはビジー応答が返されます。要求処理スレッドは別の受信要求を扱い、いつまでもキューで待機することはありません。

デフォルト値

0 (キューのサイズは無制限)

CGI

この節では、CGI プログラムの要求と関係がある `init.conf` の指令を示します。次の指令があります。

- `CGIExpirationTimeout`
- `CGIStubIdleTimeout`
- `MaxCGIStubs`
- `MinCGIStubs`
- `WingcgiTimeout` (Windows のみ)

CGIExpirationTimeout

この指令では、CGI プロセスが終了される前に実行できる最大秒数を指定します。

`CGIExpirationTimeout` の値はあまり小さくしないようにします。対話型の CGI では 300 秒 (5 分) が適当です。ただし処理に時間がかかる CGI の場合は、CGI プログラムが処理に必要とする最大持続時間を指定します。値を 0 に設定すると CGI の有効期限がなくなり、CGI の処理時間は無制限になります。

Windows プラットフォームでは `init-cgi` タイムアウトは使用できません。代わりに、`CGIExpirationTimeout` を使用します。

デフォルト値
0 (無制限)

CGIStubIdleTimeout

指令で設定した秒数の間、アイドル状態にある `CGIStub` プロセスをサーバーで終了します。プロセスの数が `MinCGIStubs` になると、サーバーはそれ以上にプロセスを終了しなくなります。

デフォルト値
30

MaxCGIStubs

サーバーが生成する `CGIStub` プロセスの最大値を制御します。これは `CGIStub` 実行時の最大同時プロセス数であり、保留要求の最大数ではありません。デフォルト値はほとんどのシステムに適合します。この値を大きくしすぎると実際のスループットが減少します。

デフォルト値

10

MinCGIStubs

デフォルトで起動するプロセス数を制御します。最初の CGIStub プロセスは CGI プログラムが呼び出されると起動します。init.conf ファイルに init-cgi 指令があれば、最小数の CGIStub プロセスが起動時に生成されます。この値は MaxCGIStubs の値より小さくする必要があります。

デフォルト値

2

WingcgiTimeout (Windows のみ)

この値より時間がかかる WinCGI プロセスは、このタイムアウト (秒単位) になると終了します。

デフォルト値

60

エラー記録

この節では、エラー記録に関係がある指令を示します。次の指令があります。

- ErrorLogDateFormat
- LogFlushInterval
- PidLog

ErrorLogDateFormat

ErrorLogDateFormat 指令では、サーバーログで使用する日付形式を指定します。

構文

ErrorLogDateFormat *format*

format には付録 A 「日時の形式」の形式がすべて使用できます。

デフォルト値

%d/%b/%Y:%H:%M:%S

LogFlushInterval

この指令では、ログフラッシュスレッドの間隔を秒単位で指定します。

デフォルト値

30

PidLog

PidLog では、ベースサーバープロセスのプロセス ID (**pid**) を記録するファイルを指定します。このログがサーバールート `logs/pid` にあるとみなすサーバーサポートプログラムもあります。

サーバーをシャットダウンするには、`-TERM` シグナルを使って `pid` ログファイルに表示されているベースサーバープロセスを終了します。設定ファイルを再び読み取ってログファイルを再度開くようサーバーに指示するには、`kill` を `-HUP` シグナルとともに使用します。

サーバーのユーザーアカウントを使って PidLog ファイルに書き込めない場合、サーバーにはプロセス ID を記録する場所がありません。プロセス ID が記録できなければサーバーは起動しません。

構文

PidLog *file*

file はプロセス ID が保存されているファイルへのフルパスとそのファイル名です。

デフォルト値

デフォルトはありません。

例

```
PidLog /var/ns-server/logs/pid
```

```
PidLog /tmp/ns-server.pid
```

ACL

この節では、アクセス制御リスト (ACL) に関連する `init.conf` の指令を示します。次の指令があります。

- `ACLCacheLifetime`
- `ACLUserCacheSize`
- `ACLGroupCacheSize`

ACLCacheLifetime

`ACLCacheLifetime` では、キャッシュエントリが無効になるまでの時間を秒数で指定します。キャッシュのエントリが参照されるたびに、経過時間が計算され `ACLCacheLifetime` と照合されます。経過時間が `ACLCacheLifetime` 以上の場合、このエントリは使用されません。この値を 0 にすると、キャッシュが無効になります。

この値を大きくすると、LDAP エントリを変更した場合に `Sun ONE Application Server` を再起動する必要が生じることがあります。たとえば 120 秒に設定すると、`Sun ONE Application Server` は 2 分間 LDAP サーバーと同期が取れなくなることもあります。LDAP が頻繁に変更されない場合、大きな値を設定できます。

デフォルト値

120

ACLUserCacheSize

`ACLUserCacheSize` ではユーザーキャッシュのユーザー数を指定します。

デフォルト値

200

ACLGroupCacheSize

ACLGroupCacheSize では、1つの UID/ キャッシュエントリに対してキャッシュできるグループ ID の数を指定します。

デフォルト値

4

セキュリティ

この節では、Sun ONE Application Server のサーバーアクセスとセキュリティに関する `init.conf` の指令を示します。次の指令があります。

- Security
- SSLCacheEntries
- SSLClientAuthDataLimit
- SSLClientAuthTimeout
- SSLSessionTimeout
- SSL3SessionTimeout

Security

Security 指令で SSL をグローバルに有効または無効にするには、サーバーインスタンスへの証明書を有効にします。仮想サーバーで SSL を使用する場合は on にしておきます。on に設定した場合、ユーザーは次の入力をするよう要求されます。

- サーバーの秘密キーが含まれている信頼データベースへのパスワード
- インストールされた暗号化されたハードウェアに必要な PIN

注	管理インタフェースでセキュア待機ソケットを作成すると、 <code>init.conf</code> のグローバルなセキュリティが自動的に有効になります。 <code>server.xml</code> でセキュア待機ソケットを手動で作成する場合は、 <code>init.conf</code> を編集してセキュリティをオンにします。
---	--

仮想サーバーの SSL を有効にする方法については、第 2 章「サーバー設定ファイル」を参照してください。

構文

Security [on|off]

デフォルト値

off

例

Security off

SSLCacheEntries

キャッシュできる SSL のセッション数を指定します。上限はありません。

構文

SSLCacheEntries *number*

number が 0 の場合は、デフォルト値の 10000 が使用されます。

SSLClientAuthDataLimit

クライアント証明書のハンドシェイクフェーズ時にバッファに入れるアプリケーションデータの最大数をバイト単位で指定します。

デフォルト値

デフォルト値は 1048576 (1 MB) です。

SSLClientAuthTimeout

クライアント証明書のハンドシェイクフェーズのタイムアウト時間を秒数で指定します。

デフォルト値

60

SSLSessionTimeout

SSLSessionTimeout 指令は、SSL2 セッションのキャッシュ化を制御します。

構文

SSLSessionTimeout *seconds*

seconds 値は、キャッシュされた SSL2 セッションが無効になるまでの秒数です。SSLSessionTimeout 指令が指定されている場合、この秒数の値は暗黙的に 5 ~ 100 秒間に制限されます。

デフォルト値

デフォルト値は 100 です。

SSL3SessionTimeout

SSL3SessionTimeout 指令は、SSL3 セッションのキャッシュ化を制御します。

構文

SSL3SessionTimeout *seconds*

seconds 値は、キャッシュされた SSL3 セッションが無効になるまでの秒数です。デフォルト値は 86400 (24 時間) です。SSL3SessionTimeout 指令が指定されている場合、この秒数の値は暗黙的に 5 ~ 86400 秒間に制限されます。

チャンクされたエンコード

この節では、チャンクされたエンコードを制御する指令を示します。

- UseOutputStreamSize
- ChunkedRequestBufferSize
- ChunkedRequestTimeout

これらの指令は `obj.conf` の Service SAF パラメータと同等です。 `obj.conf` パラメータはこれらの指令を無視します。詳細については、『Sun ONE Application Server NSAPI Developer's Guide』を参照してください。

UseOutputStreamSize

UseOutputStreamSize 指令では、`net_read` および `netbuf_grab` NSAPI 関数の出力ストリームバッファサイズのデフォルト値を指定します。これらの関数の詳細については、『Sun ONE Application Server NSAPI Developer's Guide』を参照してください。

注 `obj.conf` ファイルの UseOutputStreamSize パラメータをゼロに設定すると、出力ストリームバッファを無効にできます。 `init.conf` ファイルの UseOutputStreamSize をゼロに設定しても無効になりません。

構文

UseOutputStreamSize *size*

size の値はバイト数です。

デフォルト値

デフォルト値は 8192 (8 KB) です。

ChunkedRequestBufferSize

ChunkedRequestBufferSize 指令では、チャンクしない要求データのデフォルトバッファサイズを指定します。

構文

ChunkedRequestBufferSize *size*

size の値はバイト数です。

デフォルト値

デフォルト値は 8192 です。

ChunkedRequestTimeout

ChunkedRequestTimeout 指令では、チャンクしない要求データのデフォルトタイムアウトを指定します。

構文

ChunkedRequestTimeout *seconds*

seconds の値は秒数です。

デフォルト値

デフォルト値は 60 (1 分) です。

その他

この節では、`init.conf` のその他の指令を示します。

- `ChildRestartCallback`
- `HTTPVersion`
- `MaxRqHeaders`
- `ReentrantTimeFunctions` (Solaris のみ)
- `Umask` (UNIX のみ)

ChildRestartCallback

この指令は、サーバーの再起動時やシャットダウン時に `daemon_atrestart` NSAPI 関数を使って登録された NSAPI 関数をコールバックさせます。値は `on`、`off`、`yes`、`no`、`true`、または `false` です。`daemon_atrestart` の詳細については、『Sun ONE Application Server NSAPI Developer's Guide』を参照してください。

デフォルト値

`no`

HTTPVersion

現在サーバーが使用している HTTP バージョンは、`m.n` の形式で表します。`m` はバージョンのメジャー番号で、`n` はマイナー番号です。

デフォルト値

デフォルト値は `1.1` です。

MaxRqHeaders

要求の最大ヘッダ行数を指定します。0 ~ 512 の値を取ります。

デフォルト値

`64`

ReentrantTimeFunctions (Solaris のみ)

ReentrantTimeFunctions 指令では、サーバーがサーバーの再入可能な時間フォーマット実装をオペレーティングシステムの実装の代わりに使用するかどうかを指定します。サーバーの再入可能な実装を使うと、多数の CPU が搭載されているコンピュータのパフォーマンスがあまり向上しないことがあります。

デフォルト値は `off` で、サーバーはオペレーティングシステムの実装を使用します。この値を推奨します。

デフォルト値

`off`

Umask (UNIX のみ)

この指令では、NSAPI 関数 `system_fopenWA` および `system_fopenRW` で使用する `umask` 値を指定して、異なるモードでファイルを開きます。この指令で有効な値は標準の UNIX `umask` 値です。

これらの関数の詳細については、『Sun ONE Application Server NSAPI Developer's Guide』の `system_fopenWA` および `system_fopenRW` を参照してください。

MIME タイプ

この付録では、MIME タイプのファイルについて説明します。次の節があります。

- はじめに
- MIME タイプの決定
- Type の応答への影響
- クライアントの行う MIME タイプの処理
- MIME タイプファイルの構文
- MIME タイプファイルの例

`mime.types` ファイルは `obj.conf` ファイルと相互に作用します。`obj.conf` の詳細については、『Sun ONE Application Server Developer's Guide to NSAPI』を参照してください。

注 `mime.types` インタフェースは不確定です。不確定なインタフェース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る場合があります。

はじめに

`instance_dir/config` ディレクトリにある MIME タイプファイルには、MIME (Multipurpose Internet Mail Extensions) タイプとファイル拡張子との間のマッピングが含まれています。たとえば、MIME タイプファイルは次のように拡張子 `.html` や `.htm` を `text/html` タイプにマッピングします。

```
type=text/html exts=htm,html
```

Sun ONE Application Server はクライアントから Web のリソース要求を受け取ると、MIME タイプマッピングを使って要求されているリソースの種類を判断します。

MIME タイプは、言語 (`lang`)、エンコーディング (`enc`)、およびコンテンツタイプ (`type`) の 3 つの属性で定義されます。各タイプにこれらの属性の 1 つ以上が存在する必要があります。最もよく使われる属性は、`type` です。サーバーは、通常、クライアントへの応答の生成方法を決定する際 `type` を使います。 `enc` および `lang` 属性を使うことはほとんどありません。

デフォルトの MIME タイプファイルは `mime.types` と呼ばれます。

MIME タイプの決定

要求処理プロセスの `ObjectType` ステップで、サーバーはクライアントが要求するリソースの MIME タイプ属性を判断します。MIME タイプの決定にはいくつかの異なる関数が使われますが、通常使われるのは `type-by-extension` です。この関数は、MIME タイプテーブルにある要求されたリソースのファイル拡張子に従って MIME タイプを検索するようサーバーに伝えます。

次に拡張子に従って MIME タイプを検索するようサーバーに伝える `obj.conf` の指令を示します。

```
ObjectType fn=type-by-extension
```

サーバーが `force-type` のような別の関数を使って `type` を決定する場合、特定の要求のために MIME タイプテーブルが使われることはありません。

Type の応答への影響

サーバーは、`obj.conf` でクライアントへの応答の生成に使う `Service` 指令を決定するときに、`type` 属性の値を考慮します。

デフォルトでは、`type` が `magnus-internal/` で始まっていない場合、サーバーは要求されたファイルをクライアントに送信するだけです。この命令を含む `obj.conf` の指令は、次のとおりです。

```
Service method=(GET|HEAD|POST) type=~magnus-internal/* fn=send-file
```

慣例では、要求されたリソースをクライアントに送信すること以外の作業をサーバーに要求する `type` のすべての値は、`magnus-internal/` で始まります。

たとえば、要求されたリソースのファイル拡張子が `.map` の場合、タイプは `magnus-internal/imagemap` にマッピングされます。拡張子が `.cgi`、`.exe` または `.bat` の場合、タイプは `magnus-internal/cgi` にマッピングされます。

```
type=magnus-internal/imagemap      exts=map
type=magnus-internal/cgi            exts=cgi,exe,bat
```

`type` が `magnus-internal/` で始まっている場合、サーバーは `obj.conf` にある `Service` 指令のうち、指定されたタイプと一致するすべてを実行します。たとえば、タイプが `magnus-internal/imagemap` の場合、サーバーは次の指令の指示に従い `imagemap` 関数を使ってクライアントへの応答を生成します。

```
Service method=(GET|HEAD) type=magnus-internal/imagemap fn=imagemap
```

クライアントの行う MIME タイプの処理

Service 関数はデータを生成し、要求元のクライアントに送信します。サーバーはデータをクライアントに送信するとき、ヘッダーも送信します。これらのヘッダーには、既知のすべての MIME タイプ属性 (通常は type) が含まれます。

クライアントはデータを受信すると、MIME タイプを使ってデータの処理内容を決定します。ブラウザクライアントの場合、通常の処理はデータをブラウザウィンドウに表示することです。

要求されたリソースをブラウザに表示できず、ほかのアプリケーションで処理する必要がある場合、その type は application/ から始まります。たとえば、.bin ファイル拡張子では application/octet-stream、.fm ファイル拡張子では application/x-maker のようになります。クライアントにはユーザーが編集可能なマッピングの独自のセットがあり、どのアプリケーションを使ってどのデータを処理するのかを伝えます。

たとえば、type が application/x-maker の場合、クライアントは通常 Adobe FrameMaker を開いてファイルを表示し、処理します。

MIME タイプファイルの構文

MIME タイプファイルの最初の行では、次のようにしてファイル形式を明示します。

```
#--Sun Microsystems MIME Information
```

コメント以外の行は次の形式にします。

```
type=type/subtype exts=[file extensions]
```

- type/subtype はタイプとサブタイプです。
- exts は、このタイプに関連づけられたファイル拡張子です。

MIME タイプファイルの例

MIME タイプファイルの例を次に示します。

```

#--Sun Microsystems MIME Information
# 上の行は削除しないでください。ファイルタイプの識別に使われます。
type=application/octet-stream      exts=bin, exe
type=application/oda                exts=oda
type=application/pdf                exts=pdf
type=application/postscript         exts=ai, eps, ps
type=application/rtf                exts=rtf
type=application/x-mif              exts=mif, fm
type=application/x-gtar             exts=gtar
type=application/x-shar             exts=shar
type=application/x-tar              exts=tar
type=application/mac-binhex40      exts=hqx

type=audio/basic                    exts=au, snd
type=audio/x-aiff                   exts=aif, aiff, aifc
type=audio/x-wav                    exts=wav

type=image/gif                      exts=gif
type=image/ief                      exts=ief
type=image/jpeg                     exts=jpeg, jpg, jpe
type=image/tiff                     exts=tiff, tif
type=image/x-rgb                     exts=rgb
type=image/x-xbitmap                exts=xbm
type=image/x-rgb                     exts=xpm
type=image/x-xwindowdump            exts=xwd

type=text/html                      exts=htm, html
type=text/plain                     exts=txt
type=text/richtext                  exts=rtx
type=text/tab-separated-values      exts=tsv
type=text/x-setext                  exts=etx

type=video/mpeg                     exts=mpeg, mpg, mpe
type=video/quicktime                exts=qt, mov
type=video/x-msvideo                exts=avi

enc=x-gzip                           exts=gz
enc=x-compress                       exts=z
enc=x-uencode                        exts=uu, uue

type=magnus-internal/imagemap       exts=map
type=magnus-internal/parsed-html    exts=shtml
type=magnus-internal/cgi             exts=cgi, exe, bat
type=magnus-internal/jsp            exts=jsp

```

MIME タイプファイルの例

その他の設定ファイル

この章では、他の章で説明されていない重要な設定ファイルの**目的**、**位置**、および**内容**または**構文**を要約します。次に、各ファイルで使うことのできるすべての指令やパラメータを表に示します。他の章またはマニュアルに指令やパラメータの詳細な説明がある場合、**関連項目**の見出しの後に相互参照を示します。変更してはならない設定ファイルはこの章のリストには記載されていません。

次の設定ファイルをアルファベット順に示します。

- `dbswitch.conf`
- 配備記述子
- `generated.instance.acl`
- `nsfc.conf`
- `password.conf`
- `server.policy`

dbswitch.conf

目的

Sun ONE Application Server が使用する LDAP ディレクトリを指定します。

注

`dbswitch.conf` インタフェースは、不確定です。不確定なインタフェース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る場合があります。

位置

`instance_dir/config`

構文

```
directory name LDAP_URL
name:property1 [value1]
name:property2 [value2]
...
```

このファイルのデフォルトの内容は次のとおりです。

```
directory default null:///none
```

SSL を介して匿名バインドを実行するには、ファイルを次のように編集します。

```
directory default ldaps://directory.sun.com:636:/dc%3Dcom
```

SSL を介さないで匿名バインドを実行するには、ファイルを次のように編集します。

```
directory default ldap://directory.sun.com:389:/dc%3Dcom
```

関連項目

88 ページの「ユーザーデータベースの選択」

次の表に、dbswitch.conf ファイルのプロパティを示します。左の列にプロパティ名、左から 2 番目の列に許容値、左から 3 番目の列にデフォルト値、右の列にプロパティの説明を示します。

dbswitch.conf

プロパティ	許容値	デフォルト値	説明
nsessions	正の整数	8	データベースへの LDAP 接続の数
dyngroups	off、on、recursive	on	動的グループの処理方法を決定。off の場合、動的グループはサポートされない。on の場合、動的グループはサポートされる。recursive の場合、動的グループにほかのグループを含めることができる
binddn	有効な DN		データベースへの接続に使用する DN。binddn と bindpw の両方がない場合、バインドは匿名
bindpw			データベースへの接続に使用するパスワード。binddn と bindpw の両方がない場合、バインドは匿名

dbswitch.conf (続き)

プロパティ	許容値	デフォルト値	説明
dcsuffix	有効な DN (LDAP URL と関連して)	なし	<p>指定した場合、要求の仮想サーバーの基本 DN のデフォルト値は、仮想サーバーの host 属性のうち、クライアントの提供する Host ヘッダと一致する属性の dcsuffix DN で始まる DC ツリー検索で決定される。host 属性が1つも一致しない場合、親 http-listener の server-name 属性が使われる</p> <p>指定しない場合、基本 DN のデフォルト値は LDAP URL での base DN 値</p> <p>server.xml ファイルの auth-db 要素の basedn 属性は、この値をオーバーライドする</p>
digestauth	off、on	off	データベースがダイジェスト認証できるかどうかを指定する。on の場合、特別なディレクトリサーバープラグインが必要。このプラグインのインストール方法の詳細については、『Sun ONE Application Server 管理者ガイド』を参照

LDAP データベースがスキーマに準拠している場合、アクセスの基本 DN は、クライアントの提供する Host ヘッダと一致する virtual-server 要素の host 属性の DC ツリー検索を使って計算されます。host 属性が1つも一致しない場合、親 http-listener の server-name 属性が使われます。結果には、基本 DN を含む inetDomainBaseDN 属性が含まれている必要があります。この基本 DN は、そのまま取得され、いずれの基本 DN 値とも関連がありません。

配備記述子

目的

配備されたモジュールおよびアプリケーションに対応した Sun ONE Application Server の機能を設定します。

位置

モジュールまたはアプリケーションの META-INF または WEB-INF ディレクトリ

関連項目

次の表に、Sun ONE Application Server の配備記述子についての詳細情報の参照先を示します。左の列は配備記述子、右の列はそれらの記述子に関する詳細情報の参照先を示しています。

Sun ONE Application Server 記述子

配備記述子	詳細情報の参照先
sun-application.xml	『Sun ONE Application Server 開発者ガイド』
sun-web.xml	『Sun ONE Application Server Web アプリケーション開発者ガイド』
sun-ejb-jar.xml および sun-cmp-mapping.xml	『Sun ONE Application Server Enterprise JavaBeans 開発者ガイド』
sun-application-client.xml および sun-acc.xml	『Sun ONE Application Server Developer's Guide to Clients』
sun-ra.xml	『Sun ONE J2EE CA Service Provider Implementation 管理者ガイド』

generated.*instance.acl*

目的

サーバーインスタンスへのアクセス権を設定します。これはデフォルトの ACL ファイルですが、ユーザーが作成したりほかのファイルを使うこともできます。

注

ACL ファイルインタフェースは、不確定です。不確定なインタフェース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る場合があります。

位置

instance_dir/config

関連項目

『Sun ONE Application Server セキュリティ管理者ガイド』

nsfc.conf

目的

ファイルキャッシュのパラメータを設定します。このファイルは、ファイルのキャッシュパラメータがデフォルトから変更されている場合にのみ存在します。

注

nsfc.conf インタフェースは、不確定です。不確定なインタフェース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。

位置

instance_dir/config

構文

parameter=value

関連項目

『Sun ONE Application Server 7 パフォーマンスチューニングガイド』

次の表に、nsfc.conf ファイルのパラメータを示します。左の列にパラメータ名、左から 2 番目の列に許容値、左から 3 番目の列にデフォルト値、右の列にパラメータの説明を示します。

nsfc.conf

パラメータ	許容値	デフォルト値	説明
FileCacheEnable	on、off	on	ファイルキャッシュを有効にする
CacheFileContent	on、off	on	MediumFileSizeLimit より小さい (TransmitFiles が on になっている場合は、SmallFileSizeLimit より小さい) ファイルについて、ファイル情報だけでなくファイルの内容のキャッシュも有効にする
MaxAge	秒数	30	有効なキャッシュエントリの最大生存期間。この設定は、ファイルがキャッシュされた後の、キャッシュされた情報の使用期間を制御する。MaxAge より古いエントリは、同じファイルの新しいエントリに置き換えられる
MediumFileSizeLimit	利用可能なメモリーサイズ	537600 (525K)	(UNIX のみ) TransmitFiles が off になっている場合に、メモリーマップファイルとしてキャッシュできる 1 つのファイルの最大サイズ
MediumFileSpace	利用可能なメモリーサイズ	10485760 (10 M)	TransmitFiles が off になっている場合に、メモリーマップファイルとしてキャッシュされるファイルの合計サイズ
SmallFileSizeLimit	利用可能なメモリーサイズ	2048 (2K)	(UNIX のみ) メモリーに読み込めるファイルの最大サイズ
SmallFileSpace	利用可能なメモリーサイズ	1048576 (UNIX, 1 M), 0 (Windows)	メモリーに読み込まれるすべてのファイルの合計サイズ
TransmitFiles	on、off	on (Windows), off (UNIX)	TransmitFile システム呼び出しの使用を有効にする。IRIX、Compaq、Linux ではサポートされない
MaxFiles		1024	ファイルキャッシュ内の最大ファイル数
HashInitSize	利用可能なメモリーサイズ	0	ハッシュバケットの初期番号。0 の場合、ハッシュバケットの番号は $2 * \text{MaxFiles} + 1$ として動的に決定される

nsfc.conf (続き)

パラメータ	許容値	デフォルト値	説明
CopyFiles	on、off	on	(Windows のみ) ファイルを一時ディレクトリにコピーして共有違反を防ぐようにする
TempDir	パス名	<i>system_temp/instance</i>	CopyFiles が on の場合、ファイルキャッシュの一時ディレクトリを指定する

password.conf

目的

デフォルトでは、アプリケーションサーバーの起動時に、管理者が SSL キーデータベースのパスワードを入力する必要があります。アプリケーションサーバーを人の介入なしで再起動できるようにするには、password.conf ファイルにパスワードを保存する必要があります。システムを十分にセキュリティ保護して、このファイルとキーデータベースが危険にさらされないようにする必要があります。

位置

instance_dir/config

デフォルトでは、このファイルは存在しません。必要な場合はファイルを作成する必要があります。

構文

PKCS#11_module_namepassword

サーバーに付属している内部 PKCS#11 ソフトウェア暗号化モジュールを使用している場合には、次の情報を入力します。

internal:password

それ以外の PKCS#11 モジュールを (たとえばハードウェアの暗号化またはハードウェアアクセラータ用に) 使用している場合は、PKCS#11 モジュールの名前を指定し、続けて次のようにパスワードを入力する必要があります。

internal:password

関連項目

『Sun ONE Application Server 管理者ガイド』

server.policy

目的

アプリケーションがアクセスするリソースを制御します。これは標準の J2SE ポリシーファイルです。

位置

instance_dir/config

構文

```
grant [codeBase "path"] {  
    permission permission_class "package", "permission_type";  
    ...  
};
```

関連項目

『Sun ONE Application Server 開発者ガイド』

<http://java.sun.com/docs/books/tutorial/security1.2/tour2/index.html>

日時の形式

この付録では、サーバーログで日時に使われる形式文字列について説明します。
init.conf の ErrorLogDateFormat 指令では、これらの形式を使います。

次の表に、日付と時刻の形式を示します。左の列は日時の形式を表す記号、右の列は各記号の説明です。

日時の形式	
記号	意味
%a	曜日の名前 (3 文字の短縮形)
%d	月内の日付 (01 から 31 までの 10 進数)
%S	秒 (00 から 59 までの 10 進数)
%M	分 (00 から 59 までの 10 進数)
%H	時 (00 から 23 までの 24 時間形式)
%Y	年 (2099 までの 10 進数)
%b	月の名前 (3 文字の短縮形)
%h	月の名前 (3 文字の短縮形)
%T	"HH:MM:SS" 形式の時刻
%X	"HH:MM:SS" 形式の時刻
%A	曜日の名前 (完全形)
%B	月の名前 (完全形)
%C	"%a %b %e %H:%M:%S %Y"
%c	"%m/%d/%y %H:%M:%S" 形式の日付と時刻
%D	"%m/%d/%y" 形式の日付
%e	月内の日付 (先頭にゼロを付加しない、1 から 31 までの 10 進数)

日時の形式 (続き)

記号	意味
%I	時 (01 から 12 までの 12 時間形式)
%j	年内の日付 (001 から 366 までの 10 進数)
%k	時 (先頭にゼロを付加しない、0 から 23 までの 24 時間形式)
%l	時 (先頭にゼロを付加しない、1 から 12 までの 12 時間形式)
%m	月 (01 から 12 までの 10 進数)
%n	改行文字
%p	A.M./P.M. 表示 (12 時間形式の場合)
%R	"%H:%M" 形式の時刻
%r	"%I:%M:%S %p" 形式の時刻
%t	タブ文字
%U	年内の週 (00 から 51 までの 10 進数。日曜日を週の開始日とする)
%w	曜日 (0 から 6 までの 10 進数。日曜日を 0 とする)
%W	年内の週 (00 から 51 までの 10 進数。月曜日を週の開始日とする)
%x	"%m/%d/%y" 形式の日付
%y	年 (下 2 桁だけの、00 から 99 までの 10 進数)
%%	パーセント記号

サーバー設定要素のアルファベット順リスト

A

- acl 37
- admin-service 30
- applications 82
- auth-db 44
- auth-realm 60

C

- connector-module 87
- custom-resource 70

D

- description 29

E

- ejb-container 49
- ejb-module 85
- external-jndi-resource 71

H

http-listener 32

http-qos 43

http-service 30

I

iiop-listener 47

iiop-service 45

J

j2ee-application 84

java-config 65

jdbc-connection-pool 78

jdbc-resource 72

jms-resource 75

jms-service 55

jvm-options 68

L

lifecycle-module 83

log-service 57

M

mail-resource 73

mdb-container 53

mime 36

O

orb 45

P

persistence-manager-factory-resource 77

profiler 67

property 28

R

resources 69

S

security-service 59

server 27

server-instance 30

ssl 34

ssl-client-config 46

T

transaction-service 63

V

virtual-server 39

virtual-server-class 37

W

web-container 48

web-module 86

init.conf の指令のアルファベット順リスト

A

ACLCacheLifetime 114
ACLGroupCacheSize 115
ACLUserCacheSize 114
AsyncDNS 103

C

CGIExpirationTimeout 111
CGIStubIdleTimeout 111
ChildRestartCallback 119
ChunkedRequestBufferSize 118
ChunkedRequestTimeout 118
ConnQueueSize 105

D

DNS 103

E

ErrorLogDateFormat 113

H

HeaderBufferSize 105

HTTPVersion 119

I

IOTimeout 105

K

KeepAliveThreads 106

KeepAliveTimeout 106

KernelThreads (Windows のみ) 106

L

ListenQ 107

LogFlushInterval 113

M

MaxCGIStubs 111

MaxKeepAliveConnections 107

MaxRqHeaders 119

MinCGIStubs 112

N

NativePoolMaxThreads 110

NativePoolMinThreads 110

NativePoolQueueSize 110

NativePoolStackSize 110

NetSiteRoot 101

P

PidLog 113

PostThreadsEarly 107

R

RcvBufSize 107

ReentrantTimeFunctions (Solaris のみ) 120

RqThrottle 107

RqThrottleMin 108

S

Security 115

SndBufSize 108

SSL3SessionTimeout 117

SSLCacheEntries 116

SSLClientAuthDataLimit 116

SSLClientAuthTimeout 116

SSLSessionTimeout 116

StackSize 108

StrictHttpHeaders 108

T

TempDir 101
TempDirSecurity 102
TerminateTimeout 108
ThreadIncrement 109

U

Umask (UNIX のみ) 120
UseOutputStreamSize 117
User (UNIX のみ) 102

W

WincgiTimeout (Windows のみ) 112

索引

A

accept-language 属性, 38, 42
acceptor-threads 属性, 33
accesslog 変数, 92
ACLCacheLifetime init.conf 指令, 114
ACLGroupCacheSize init.conf 指令, 115
acls 属性, 41
ACLUserCacheSize init.conf 指令, 114
ACL ファイル, 37, 88
 デフォルト, 131
 の init.conf 指令, 114
acl 要素, 37
address 属性, 32, 47
admin-password 属性, 55
admin-service 要素, 30
admin-user-name 属性, 55
anonymous-role 属性, 60
append-version プロパティ, 56
application-root 属性, 28
application 要素, 82
assign-groups プロパティ, 62
AsyncDNS init.conf 指令, 103
audit-enabled 属性, 60
auth-db 要素, 44
auth-realm 要素, 60
automatic-recovery 属性, 63

B

bandwidth-limit 属性, 43
basedn 属性, 44
base-dn プロパティ, 62
binddn プロパティ, 128
bindpw プロパティ, 128
blocking-enabled 属性, 33
bytecode-preprocessors 属性, 67

C

CacheFileContent パラメータ, 132
cache-idle-timeout-in-seconds 属性, 51
cache-resize-quantity 属性, 51
certmaps 属性, 44
cert-nickname 属性, 34
CGI
 init.conf での設定, 111
 の変数, 92
CGIExpirationTimeout init.conf 指令, 111
CGIStubIdleTimeout init.conf 指令, 111
ChildRestartCallback init.conf 指令, 119
chroot 変数, 92
ChunkedRequestBufferSize init.conf 指令, 118
ChunkedRequestTimeout init.conf 指令, 118
class-name 属性, 84

classname 属性, 61
classpath-prefix 属性, 66
classpath-suffix 属性, 66
classpath 属性, 68, 84
client-auth-enabled 属性, 35
CMP, 77
cmt-max-runtime-exceptions プロパティ, 54
commit-option 属性, 52
config-file 属性, 38, 40
config ディレクトリ、位置, 17
connection-limit 属性, 43
connection-validation-method 属性, 80
connector-module 要素, 87
ConnQueueSize init.conf 指令, 105
context-root 属性, 86
CopyFiles パラメータ, 133
create-console 属性, 58
custom-resource 要素, 70

D

databaseName プロパティ, 81
database 属性, 44
datasource-classname 属性, 78
datasourceName プロパティ, 81
dbswitch.conf ファイル, 88, 89, 127
dcsuffix プロパティ, 129
debug-enabled 属性, 65
debug-options 属性, 66
default-object 属性, 38, 41
default-principal-password 属性, 59
default-principal 属性, 59
default-realm 属性, 59
default-virtual-server 属性, 33
default-web-module 属性, 40
description プロパティ, 81
description 要素, 29

digestauth プロパティ, 129
directory プロパティ, 62
dir 変数, 92
disable-distributed-transaction-logging プロパティ, 64
DNS init.conf 指令, 103
docroot 変数, 92
Domain Name Service、「DNS」を参照
dyngroups プロパティ, 128

E

echo-log-messages-to-stderr 属性, 58
ejb-container 要素, 49
ejb-module 要素, 85
enc パラメータ, 122
enforce-bandwidth-limit 属性, 43
enforce-connection-limit 属性, 43
env-classpath-ignored 属性, 67
ErrorLogDateFormat init.conf 指令, 113
external-jndi-resource 要素, 71

F

factory-class 属性, 71, 72, 77
fail-all-connections 属性, 80
family 属性, 33
FileCacheEnable パラメータ, 132
file 属性, 58
 acl 要素の, 37
 mime 要素の, 36
file プロパティ, 61

G

group-base-dn プロパティ, 62

group-search-filter プロパティ, 62
group-target プロパティ, 62
group 変数, 92

H

HashInitSize パラメータ, 132
HeadBufferSize init.conf 指令, 105
heuristic-decision 属性, 63
hosts 属性, 41
 サブジェクトのパターンとの比較, 39
http-listeners 属性, 40
http-listener 要素, 32
http-qos 要素, 43
http-service 要素, 30
HTTPVersion init.conf 指令, 119
HTTP 待機ソケット, 32
HTTP 要求、処理, 18
HUP シグナル
 PidLog, 113

I

idle-thread-timeout-in-seconds 属性, 46
idle-timeout-in-seconds 属性, 53, 79
id 属性
 acl 要素の, 37
 auth-db 要素の, 44
 http-listener 要素の, 32, 47
 mime 要素の, 36
 virtual-server-class 要素の, 38
 virtual-server 要素の, 40
id 変数, 91
iiop-listener 要素, 47
 sun-server_1_0.dtd ファイルの定義, 24
iiop-service 要素, 45
IIOP 待機ソケット, 47

imqBrokerHostName プロパティ, 76
imqBrokerHostPort プロパティ, 76
imqConfiguredClientID プロパティ, 76
imqDestinationName プロパティ, 76
init conf ファイル, 18, 99
 指令のアルファベット順リスト, 141
init-timeout-in-seconds 属性, 55
Init 関数, 101
instance-name-suffix プロパティ, 56
instance-name プロパティ, 56
IOTimeout init.conf 指令, 105
is-connection-validation- required 属性, 80
is-failure-fatal 属性, 84
is-isolation-level-guaranteed 属性, 80

J

j2ee-application 要素, 84
jaas-context プロパティ, 61
Java Database Connectivity、「JDBC」を参照
Java Message Service、「JMS」を参照
Java Naming and Directory Interface、「JNDI」を参
照
Java Platform Debugger Architecture、「JPDA」を
参照
Java Transaction Service、「JTS」を参照
Java Virtual Machine、「JVM」を参照
javac-options 属性, 66
java-home 属性, 65
JavaMail, 73
jdbc-connection-pool 要素, 78
jdbc-resource-jndi-name 属性, 77
jdbc-resource 要素, 72
JDBC 接続プール, 78
JDBC リソース (JDBC resource), 72
JMS, 55, 75
jms-resource 要素, 75
jms-service 要素, 55

JNDI, 71
jndi-lookup-name 属性, 71
jndi-name 属性
 custom-resource 要素の, 70
 external-jndi-resource 要素の, 71
 jdbc-resource 要素の, 72
 jms-resource 要素の, 75
 mail-resource 要素の, 73
 persistence-manager-factory-resource 要素の, 77
JPDA デバッグオプション, 66
java-config 要素, 65
JTS, 63
JVM, 65
 サーバーへのオプションの追加, 68
jvm-options 要素, 68

K

KeepAliveThreads init.conf 指令, 106
KeepAliveTimeout init.conf 指令, 106
KernelThreads init.conf 指令, 106
keypoint-interval 属性, 64

L

lang パラメータ, 122
LDAP, 88
 Sun ONE スキーマ, 89
 マッピングに使う証明書を指定, 44
level 属性, 58
lifecycle-module 要素, 83
ListenQ init.conf 指令, 107
load-order 属性, 84
locale 属性, 28
location 属性, 85, 86, 87
log-file 属性, 42
LogFlushInterval init.conf 指令, 113

log-level 属性, 58
log-root 属性, 28
log-service 要素, 57
log-stderr 属性, 58
log-stdout 属性, 58
log-virtual-server-id 属性, 58

M

mail-resource 要素, 73
MaxAge パラメータ, 132
max-cache-size 属性, 51
MaxCGIStubs init.conf 指令, 111
max-connections 属性, 46
MaxFiles パラメータ, 132
MaxKeepAliveConnections init.conf 指令, 107
max-pool-size 属性, 50, 53, 79
MaxRqHeaders init.conf 指令, 119
max-thread-pool-size 属性, 46
max-wait-time-in-millis 属性, 79
mdb-container 要素, 53
MediumFileSizeLimit パラメータ, 132
MediumFileSpace パラメータ, 132
message-fragment-size 属性, 46
mime.types ファイル
 server.xml 要素の参照, 36
 概要, 19
 構文, 124
 サンプル, 124
 デフォルト, 122
mime 属性, 41
MIME タイプ, 121
mime 要素, 36
MinCGIStubs init.conf 指令, 112
monitoring-enabled 属性, 46, 48, 52, 54, 64

N

name 属性

- auth-realm 要素の, 61
- connector-module 要素の, 87
- ejb-module 要素の, 85
- j2ee-application 要素の, 85
- jdbc-connection-pool 要素の, 78
- lifecycle-module 要素の, 83
- profiler 要素の, 68
- property 要素, 29
- server 要素, 28
- web-module 要素の, 86

native-library-path- prefix 属性, 66

native-library-path-suffix 属性, 67

native-library-path 属性, 68

NativePoolMaxThreads init.conf 指令, 110

NativePoolMinThreads init.conf 指令, 110

NativePoolQueueSize init.conf 指令, 110

NativePoolStackSize init.conf 指令, 110

networkProtocol プロパティ, 81

nice 変数, 92

nsessions プロパティ, 128

nsfc.conf ファイル, 131

O

obj.conf ファイル, 18, 40

と仮想サーバー, 18

oracle-xa-recovery-workaround プロパティ, 64

orb 要素, 45

P

password.conf ファイル, 133

password プロパティ, 81

persistence-manager-factory-resource 要素, 77

PidLog init.conf 指令, 113

pool-idle-timeout-in-seconds 属性, 51

pool-name 属性, 72

pool-resize-quantity 属性, 50, 53, 79

port 属性, 32, 47, 55

port プロパティ, 81

PostThreadsEarly init.conf 指令, 107

profiler 要素, 67

property 要素, 28

R

RcvBufSize init.conf 指令, 107

reconnect-delay-in-seconds プロパティ, 54

reconnect-enabled プロパティ, 54

reconnect-max-retries プロパティ, 54

ReentrantTimeFunctions init.conf 指令, 120

removal-timeout-in-seconds 属性, 52

resources 要素, 69

res-type 属性, 71, 72, 75, 79

rmic-options 属性, 66

roleName プロパティ, 81

RqThrottle init.conf 指令, 107

RqThrottleMin init.conf 指令, 108

S

search-bind-dn プロパティ, 62

search-bind-password プロパティ, 62

search-filter プロパティ, 62

Secure Socket Layer、「SSL」を参照

Security init.conf 指令, 115

security-enabled 属性, 33

security-service 要素, 59

send-cgi 関数、の変数, 92

server.policy ファイル, 134

server.xml ファイル, 18

- 位置, 23
- の例, 93
- 要素, 26
 - アルファベット順リスト, 137
- server-classpath 属性, 66
- server-instance 要素, 30
- server-name 属性, 33
- serverName プロパティ, 81
- server 要素, 27
- session-store 属性, 28
- SmallFileSizeLimit パラメータ, 132
- SmallFileSpace パラメータ, 132
- SndBufSize init.conf 指令, 108
- SSL
 - init.conf での設定, 115
 - 設定, 34
- ssl2-ciphers 属性, 35
- ssl2-enabled 属性, 34
- ssl3-enabled 属性, 35
- SSL3SessionTimeout init.conf 指令, 117
- ssl3-tls-ciphers 属性, 35
- SSL3 クライアント認証, 35
- SSLCacheEntries init.conf 指令, 116
- SSLClientAuthDataLimit init.conf 指令, 116
- SSLClientAuthTimeout init.conf 指令, 116
- ssl-client-config 要素, 46
- SSLSessionTimeout init.conf 指令, 116
- ssl 要素, 34
- sso-enabled プロパティ, 42
- sso-max-inactive-seconds プロパティ, 42
- sso-reap-inaterval-seconds プロパティ, 42
- StackSize init.conf 指令, 108
- start-args 属性, 56
- state 属性, 41
- stderr 出力, 58
- stdout 出力, 58
- steady-pool-size 属性, 50, 53, 79
- steady-thread-pool-size 属性, 46
- store-protocol-class 属性, 73
- store-protocol 属性, 73
- StrictHttpHeaders init.conf 指令, 108
- Sun ONE Application Server、「サーバー」を参照
- Sun ONE Message Queue, 56
- sun-acc.xml ファイル, 130
- sun-application.xml ファイル, 130
- sun-application-client.xml ファイル, 130
- sun-cmp-mapping.xml ファイル, 130
- sun-ejb-jar.xml ファイル, 130
- sun-ra.xml ファイル, 130
- sun-server_1_0.dtd ファイル, 24
- sun-web.xml ファイル, 130
- Sun カスタマサポート, 15
- sybase-xa-recovery-workaround プロパティ, 64

T

- TempDir init.conf 指令, 101
- TempDirSecurity init.conf 指令, 102
- TempDir パラメータ, 133
- TerminateTimeout init.conf 指令, 108
- TERM シグナル, 113
- ThreadIncrement init.conf 指令, 109
- timeout-in-seconds 属性, 63
- tls-enabled 属性, 35
- tls-rollback-enabled 属性, 35
- transaction-isolation-level 属性, 79
- transaction-service 要素, 63
- TransmitFiles パラメータ, 132
- transport-protocol-class 属性, 74
- transport-protocol 属性, 73
- tx-log-dir 属性, 63
- type-by-extension 関数, 122
- type パラメータ, 122

U

Umask init.conf 指令, 120
UNIX ユーザーアカウント、指定, 102
url プロパティ, 81
UseOutputStreamSize init.conf 指令, 117
User init.conf 指令, 102
user プロパティ, 81
user 変数, 92
use-system-logging 属性, 59

V

validation-table-name 属性, 80
value 属性, 29
victim-selection-policy 属性, 52
virtual-server-class 要素, 37
virtual-servers 属性, 85, 86
virtual-server 要素, 39

W

web-container 要素, 48
web-module 要素, 86
Web モジュール、デフォルト, 40
WingcgiTimeout init.conf 指令, 112
Windows コンソール, 58

X

xaresource-txn-timeout プロパティ, 64

あ

アクセスログファイル, 57

アプリケーション
位置, 28
の設定, 84
アルファベット順参照
init.conf 指令, 141
server.xml 要素, 137

え

エンコード、チャンクされた, 117

か

仮想サーバー
と obj.conf ファイル, 18
のログファイル, 57

き

記録する
init.conf での設定, 113
server.xml の設定, 57

け

形式、時刻, 135

こ

コンソール、Windows, 58
コンテナ管理持続性, 77

な

サーバー

HUP シグナル, 113

init.conf の初期化指令, 99

TERM シグナル, 113

基本動作, 17

設定する init.conf 指令, 101

プロセスを終了, 113

変更, 17

サーバーログファイル, 57

位置, 28, 58

サービス品質, 43

サブ要素、について, 24

し

時刻形式, 135

収束ツリー

AUXILIARY クラスの inetSubscriber, 90

LDAP スキーマの, 89

組織の, 89

ユーザーエントリは inetOrgPerson と呼ばれる,
90

収束ツリーの inetOrgPerson, 90

証明書, 62

init.conf での設定, 115

初期化の設定, 18, 99

シングルサインオン, 42

す

スレッド、init.conf での設定, 104

スレッドプール、init.conf での設定, 109

せ

セキュリティ, 59

init.conf での設定, 115

待機ソケットの, 32, 47

接続、init.conf での設定, 104

設定、動的な, 19

設定ファイル, 17

そ

属性、について, 25

た

待機ソケット, 32, 47

ち

チャンクされたエンコード, 117

て

データベースの選択, 44, 88

デフォルトの仮想サーバー

http-listener 要素の, 33

サーバー全体の, 39

と

動的再設定, 19

ドメインコンポーネントツリー (dc), 90

トランザクションログファイル, 57, 63

ね

ネイティブのスレッドプール、`init.conf`での設定、109
ネイティブライブラリパス、設定、66

は

配備記述子、130

ひ

必要指定数、24

ふ

ファイル、マッピングタイプ、122
ファイル名拡張子、MIMEタイプ、122
プロセス、`init.conf`での設定、104
プロパティ、について、28

へ

変数、91
「`obj.conf`」で参照、91
インタフェースで使われる、92
置き換えを許可、91
形式、91
評価、93

め

メッセージ駆動型 Beans、53

ゆ

ユーザーデータベースの選択、44, 88

よ

要求応答プロセス、19
要求処理プロセス、19

り

リソースアダプタ、87

