Oracle® Application Server

管理者ガイド

10*g* リリース 3（10.1.3.2.0）

部品番号 : E05047-01

2007 年 6 月

ORACLE®

Oracle Application Server 管理者ガイド , 10*g* リリース 3（10.1.3.2.0）

部品番号 : E05047-01

原本名 : Oracle Application Server Administrator's Guide, 10*g* Release 3 (10.1.3.2.0)

原本部品番号 : B32196-01

原本著者 : Helen Grembowicz, Kevin Hwang, Peter LaQuerre, Mary Beth Roeser, Harry Schaefer, Deborah Steiner

原本協力者 : Steven Button, Megan Ginter, Pavana Jain, Michael Lehmann, Thomas Van Raalte

目次

3 起動と停止

第 II 部 基本的な管理

4 ポートの管理

5 ログ・ファイルの管理

第 III 部 高度な管理

6 Application Server インスタンスの再構成

7 ネットワーク構成の変更

8 Infrastructure サービスの変更

9 Application Server 中間層インスタンスのクローニング

第 IV 部 Secure Sockets Layer（SSL）

10 Oracle Application Server の Secure Sockets Layer（SSL）の概要

11 Wallet と証明書の管理

12 Infrastructure での SSL の有効化

13 中間層での SSL の有効化

14 SSL のトラブルシューティング

第 V 部 バックアップとリカバリ

15 バックアップとリカバリの概要

16 Oracle Application Server Recovery Manager

17 バックアップ計画と手順

18 リカバリ計画と手順

19 OracleAS Recovery Manager のトラブルシューティング

第 VI 部 付録と用語集

A Application Server Control の管理および構成

H Oracle Application Server のトラブルシューティング

用語集

索引

図一覧

表一覧

はじめに

このマニュアルでは、Oracle Application Server の起動および停止方法、コンポーネントの再構成方法、Oracle Application Server のバックアップおよびリカバリ方法など、Oracle Application Server の管理方法について説明します。

対象読者

このマニュアルは、Oracle Application Server の管理者を対象としています。

ドキュメントのアクセシビリティについて

オラクル社は、障害のあるお客様にもオラクル社の製品、サービスおよびサポート・ドキュメントを簡単にご利用いただけることを目標としています。オラクル社のドキュメントには、ユーザーが障害支援技術を使用して情報を利用できる機能が組み込まれています。HTML 形式のドキュメントで用意されており、障害のあるお客様が簡単にアクセスできるようにマークアップされています。標準規格は改善されつつあります。オラクル社はドキュメントをすべてのお客様がご利用できるように、市場をリードする他の技術ベンダーと積極的に連携して技術的な問題に対応しています。オラクル社のアクセシビリティについての詳細情報は、Oracle Accessibility Program の Web サイト `http://www.oracle.com/accessibility/` を参照してください。

ドキュメント内のサンプル・コードのアクセシビリティについて

スクリーン・リーダーは、ドキュメント内のサンプル・コードを正確に読めない場合があります。コード表記規則では閉じ括弧だけを行に記述する必要があります。しかし JAWS は括弧だけの行を読まない場合があります。

外部 Web サイトのドキュメントのアクセシビリティについて

このドキュメントにはオラクル社およびその関連会社が所有または管理しない Web サイトへのリンクが含まれている場合があります。オラクル社およびその関連会社は、それらの Web サイトのアクセシビリティに関しての評価や言及は行っておりません。

Oracle サポート・サービスへの TTY アクセス

アメリカ国内では、Oracle サポート・サービスへ 24 時間年中無休でテキスト電話（TTY）アクセスが提供されています。TTY サポートについては、(800)446-2398 にお電話ください。

関連ドキュメント

詳細は、次の Oracle ドキュメントを参照してください。

- Oracle Application Server ドキュメント・ライブラリ
- Oracle Application Server プラットフォーム固有のドキュメント

リリース・ノート、インストール関連ドキュメント、ホワイト・ペーパーまたはその他の関連ドキュメントは、OTN-J（Oracle Technology Network Japan）から、無償でダウンロードできます。OTN-J を使用するには、オンラインでの登録が必要です。登録は、次の Web サイトから無償で行えます。

`http://www.oracle.com/technology/documentation/`

表記規則

本文では、次の表記規則を使用します。

規則	意味
太字	太字は、操作に関連するグラフィカル・ユーザー・インタフェース要素、または本文中で定義されている用語および用語集に記載されている用語を示します。
イタリック	イタリックは、特定の値を指定するプレースホルダ変数を示します。
固定幅フォント	固定幅フォントは、パラグラフ内のコマンド、URL、例に記載されているコード、画面に表示されるテキスト、または入力するテキストを示します。

サポートおよびサービス

次の各項に、各サービスに接続するための URL を記載します。

Oracle サポート・サービス

オラクル製品サポートの購入方法、および Oracle サポート・サービスへの連絡方法の詳細は、次の URL を参照してください。

```
http://www.oracle.co.jp/support/
```

製品マニュアル

製品のマニュアルは、次の URL にあります。

```
http://otn.oracle.co.jp/document/
```

研修およびトレーニング

研修に関する情報とスケジュールは、次の URL で入手できます。

```
http://www.oracle.co.jp/education/
```

その他の情報

オラクル製品やサービスに関するその他の情報については、次の URL から参照してください。

```
http://www.oracle.co.jp
http://otn.oracle.co.jp
```

注意： ドキュメント内に記載されている URL や参照ドキュメントには、Oracle Corporation が提供する英語の情報も含まれています。日本語版の情報については、前述の URL を参照してください。

Oracle Application Server 管理の新機能

ここでは、Oracle Application Server 10*g* リリース 3（10.1.3.2.0）の新しい管理機能と変更された管理機能を紹介します。ここで紹介する内容の多くは、Oracle Application Server 10*g* リリース 2（10.1.2）または 10*g* リリース 3（10.1.3.0.0 または 10.1.3.1.0）を含む、以前のリリースの Oracle Application Server を管理していたユーザーに役立ちます。

10*g* リリース 3（10.1.3.2.0）の新機能

Oracle Application Server 10*g* リリース 3（10.1.3.2.0）の新しい管理機能は次のとおりです。

- Oracle WebCenter Framework がサポートされています。Oracle WebCenter Framework の概要は、第 1.5.4 項を参照してください。
- Oracle Content DB がサポートされています。Oracle Content DB の概要は、第 1.5.5 項を参照してください。

第 I 部

概要

この部では、Oracle Application Server の管理の概要について説明します。

この部は、次の章で構成されています。

- 第 1 章「Oracle Application Server のインストール後の作業」
- 第 2 章「管理ツールの概要」
- 第 3 章「起動と停止」

1

Oracle Application Server のインストール後の作業

この章では、Oracle Application Server の管理の概要の理解に役立つ、インストール後の作業について説明します。

この章の項目は次のとおりです。

1.1 Oracle Application Server 10*g* リリース 3（10.1.3.2.0）の概要

Oracle Application Server 10*g* リリース 3（10.1.3.2.0）は、完全な Java 2 Enterprise Edition（J2EE）1.4 準拠の環境です。

インストール・タイプによって、Oracle HTTP Server、Oracle Containers for J2EE（OC4J）、Oracle Process Manager and Notification Server（OPMN）、Application Server Control コンソール、OC4J Java Single Sign-On、Oracle WebCenter Framework、Oracle Content DB および Oracle Business Rules が含まれています。

このリリースは、リリース 10.1.4 またはリリース 2（10.1.2）の Oracle Identity Management サービス、およびリリース 2（10.1.2）の Oracle Application Server Web Cache と併用できます。

1.2 作業 1: 環境変数の設定

インストール作業を行った担当者は、Oracle Application Server のインストール時に特定のユーザーとしてオペレーティング・システムにログインしていました。このユーザーがインストールの Oracle ホームにあるファイルを表示および変更する権限を持つため、インストールを管理するには、常にこのユーザーとしてログインする必要があります。

Oracle Application Server を使用するには、次の表に示すように、環境変数を設定する必要があります。

- 表 1-1「環境変数（UNIX 用）」
- 表 1-2「環境変数（Windows 用）」

表 1-1 環境変数（UNIX 用）

環境変数	値
DISPLAY	*hostname*:*display_number*.*screen_number* Oracle Application Server 10*g* から、`oidadmin` などのごく一部のツールで DISPLAY 変数が必要となりました。

LD_LIBRARY_PATH	Solaris の場合は、この値に次のディレクトリが含まれていることを確認します。
	`$ORACLE_HOME/lib32`
	Linux の場合は、この値に次のディレクトリが含まれていることを確認します。
	`$ORACLE_HOME/lib`
	HP-UX の場合は、この値に次のディレクトリが含まれていることを確認します。
	`$ORACLE_HOME/lib`
	IBM AIX の場合は、この環境変数の設定がないことを確認します。

（IBM AIX のみ）
LIBPATH
	コール元のアプリケーションが 32 ビット・アプリケーションの場合は、この値に次のディレクトリが含まれていることを確認します。
	`$ORACLE_HOME/lib32`
	コール元のアプリケーションが 64 ビット・アプリケーションの場合は、この値に次のディレクトリが含まれていることを確認します。
	`$ORACLE_HOME/lib`

（Solaris のみ） LD_LIBRARY_PATH_64	この値に次のディレクトリが含まれていることを確認します。 `$ORACLE_HOME/lib`
（HP-UX のみ） SHLIB_PATH	この値に次のディレクトリが含まれていることを確認します。 `$ORACLE_HOME/lib32`

表 1-1 環境変数（UNIX 用）（続き）

環境変数	値
ORACLE_HOME	インストールの Oracle ホームのフルパスに設定します。インストール時に root ユーザーとして root.sh を実行すると、この変数が設定されます。

PATH	この値に次のディレクトリが含まれていることを確認します。これらのディレクトリには、すべてのインストールで使用される基本的なコマンドがあります。
	`$ORACLE_HOME/bin`
	`$ORACLE_HOME/opmn/bin`
	特定のコンポーネントで作業を開始するときは、コンポーネントのドキュメントで推奨されているように、必要に応じてさらにディレクトリを追加します。

表 1-2 に、Windows の環境変数を示します。

表 1-2 環境変数（Windows 用）

環境変数	値
ORACLE_HOME	インストールの Oracle ホームのフルパスに設定します。 この値は、Oracle Universal Installer によって自動設定されます。
TEMP	一時ディレクトリに設定します（例、`C:¥temp`）。
TMP	一時ディレクトリに設定します（例、`C:¥temp`）。

UNIX ホストに複数のインストールがある場合の最適な設定例

UNIX ホストに Oracle Application Server の複数のインストールがある場合は、特定のインストールの管理を開始するときに、環境を完全に設定することが重要です。

一部の Oracle Application Server コマンドは、ORACLE_HOME 環境変数を使用して、操作対象のインストールを決定します。また、一部のコマンドは、そのコマンドのあるディレクトリを使用します。そのため、複数のインストール間を移動するときは、そのユーザーの環境変数を再設定したり、`cd` で別の Oracle ホームに移動するだけでは不十分です。次の手順に従って、新しいインストールに完全に変更する必要があります。

1. 操作対象のインストールをインストールしたユーザーとしてログインします。

   UNIX ホストでは、`su` コマンドを使用して目的のユーザーに切り替えることもできますが、実際にそのユーザーとしてログインしたときと同じ環境に設定されるようにダッシュ（-）オプションを必ず使用する必要があります。次に例を示します。

   ```
   su - user
   ```

2. 表 1-1 の説明に従って、操作対象のインストールに適した環境変数を設定します。
3. 操作対象のインストールの Oracle ホームでコマンドを実行します。

同じユーザーによる複数インストール　同じユーザーで複数インストールしている場合は、特定のインストールで作業するときに、正しい Oracle ホームに移動し、正しい環境変数を設定していることを確認してください。スクリプトを設定して、インストール間を簡単に変更できるようにすると便利です。

1.3 作業 2: Oracle Application Server の「ようこそ」ページの使用

Oracle Application Server の「ようこそ」ページは、アプリケーション・サーバーの管理のための重要な開始ポイントです。このページには、次の内容が含まれます。

- このリリースの概要
- Oracle Enterprise Manager 10*g* Application Server Control コンソール（Oracle Application Server を管理する Web ベースのツール）へのリンク
- 追加のリソースへのリンク

図 1-1 に、基本インストールを選択した場合の「ようこそ」ページの一部を示します。

図 1-1 Oracle Application Server の「ようこそ」ページ

「ようこそ」ページへのアクセス

「ようこそ」ページへのアクセスに使用する URL は、インストールの終了画面のテキスト上で見つけることができます。これは、次のファイルにあります。

```
(UNIX) ORACLE_HOME/install/readme.txt
(Windows) ORACLE_HOME¥install¥readme.txt
```

「ようこそ」ページを表示するには、インストールの HTTP リスナー・ポートを使用して接続します。次に例を示します。

```
http://hostname.domain:port
```

「基本インストール」オプションを選択した場合、デフォルトのポートは UNIX では 7777、Windows では 80 です。

ヒント 「ようこそ」ページにアクセスできない場合は、次の手順を実行してください。

1. `readme.txt` をチェックし、正しい URL（ホスト名およびポート番号）を使用していることを確認します。
2. Oracle HTTP Server を再起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server

(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopproc ias-component=HTTP_Server
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startproc ias-component=HTTP_Server
```

1.4 作業 3: ポート番号のチェック

インストール時に、Oracle Application Server により、各種コンポーネントおよびサービスにポート番号が割り当てられます。これらのポート番号をチェックしておくことは、次の理由で重要です。

- アプリケーション・サーバーの管理を開始するには、これらのポート番号を把握している必要があります。
- Oracle Application Server では、いくつかの方法により、ポート番号の割当てが一意であることが確認されます。しかし、インストール時に実行されていない、Oracle Application Server 以外のホスト上のプロセスとは、ポートの割当てが競合する可能性があります。競合があるとわかった場合は、Oracle Application Server 以外のプロセスを停止し、この章の作業を続けてください。この章の作業が完了し、インストールが適切に動作していることが確認されたら、Oracle Application Server のポート番号の変更を検討します。

  関連項目： ポート番号の変更の詳細は、第 4 章を参照してください。

次のコマンドを入力して、使用されているポート番号のリストを参照できます。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl status -l
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl status -l
```

Linux でのインストールにおける出力を次に示します（読みやすさを考慮して、一部の列は省略されています）。

```
Processes in Instance: orcl10132.myhost.ora.com
-------------------------+---------------------+--------+-----+-----------------------------------------
ias-component            | process-type        |  pid-  | ... | ports
-------------------------+---------------------+--------+-----+-----------------------------------------
OC4JGroup:default_group  | OC4J:OC4J_Content   | 27769  | ... | jms:12602,ajp:12502,rmis:12702,rmi:12402
OC4JGroup:default_group  | OC4J:OC4J_WebCent~  | 27501  | ... | jms:12601,ajp:12501,rmis:12701,rmi:12401
OC4JGroup:default_group  | OC4J:home           | 27394  | ... | jms:12603,ajp:12503,rmis:12703,rmi:12403
HTTP_Server              | HTTP_Server         |  26928 | ... |https1:4443,http2:7200,http1:7777
ASG                      | ASG                 |  N/A   | ... | N/A
```

ポート番号は、この章の作業が完了し、すべてのコンポーネントが適切に動作していることが確認されるまでは、そのままにしてください。その後で、ポート番号の変更を検討します。一部のポート番号は変更できません。また、ポート番号によっては、他のコンポーネントを更新する追加手順が必要になる点に注意してください。

Windows では、Windows の「スタート」メニューからポート番号を表示できます。たとえば、Windows 2000 では、「**スタート**」→「**プログラム**」→「**Oracle - *Oracle_home_name***」→「**Oracle Process Manager**」→「**Oracle Assigned Port Numbers**」を選択します。

1.5 作業 4: コンポーネント管理の概要の理解

この項では、コンポーネント管理の開始方法について説明します。コンポーネント管理ツールへのアクセス手順についても説明し、関連情報の参照先を示します。この項の項目は次のとおりです。

- Oracle Process Manager and Notification Server（OPMN）の概要
- Oracle HTTP Server の概要
- Oracle Containers for J2EE（OC4J）の概要
- Oracle WebCenter Framework の概要
- Oracle Content DB の概要

> **関連項目：** 各コンポーネントへのアクセス方法のクイック・リファレンスは、付録 C を参照してください。

1.5.1 Oracle Process Manager and Notification Server（OPMN）の概要

Oracle Process Manager and Notification Server（OPMN）は、ほとんどの Oracle Application Server コンポーネントを管理および監視します。OPMN は、各中間層インストールでインストールおよび構成され、Oracle Application Server の実行に対して重要な役割を持ちます。

OPMN には opmnctl コマンドがあります。この実行可能ファイルは次のディレクトリにあります。

```
(UNIX) ORACLE_HOME/opmn/bin
(Windows) ORACLE_HOME¥opmn¥bin
```

OPMN の概要を理解するには、次の opmnctl コマンドを使用して、それぞれのインストールにおける各コンポーネントのステータスを問い合せます。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl status
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl status
```

例 1-1 に、このコマンドの出力例を示します。コマンド出力には、コンポーネント名、プロセスのタイプ、オペレーティング・システム・プロセス ID（PID）および各プロセスのステータスが表示されます。

例 1-1 opmnctl status コマンドの出力例

```
Processes in Instance: orcl10132.myhost.ora.com
-------------------------+--------------------+---------+---------
ias-component            | process-type       |     pid | status
-------------------------+--------------------+---------+---------
OC4JGroup:default_group  | OC4J:OC4J_Content  |   27769 | Alive
OC4JGroup:default_group  | OC4J:OC4J_WebCent~ |   27501 | Alive
OC4JGroup:default_group  | OC4J:home          |   27394 | Alive
HTTP_Server              | HTTP_Server        |   26928 | Alive
ASG                      | ASG                |     N/A | Down
```

OPMN は、アプリケーション・サーバーの起動と停止、コンポーネントの監視、イベント・スクリプトの構成に使用できます。また、プロセス管理に関連する他の多くのタスクの実行にも使用できます。たとえば、UNIX では、次のコマンドを使用して、OPMN とすべての OPMN 管理プロセス（Oracle HTTP Server や OC4J インスタンスなど）を起動および停止できます。

```
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/opmn/bin/opmnctl stopall
```

Windows では、Windows の「スタート」メニューから、これらのコマンドを起動できます。たとえば、Windows 2000 ですべてのプロセスを起動するには、「**スタート**」→「**プログラム**」→「**Oracle - *Oracle_home_name***」→「**Oracle Process Manager**」→「**Start Oracle Process Manager**」を選択します。

関連項目：『Oracle Process Manager and Notification Server 管理者ガイド』

Oracle Application Server のインストール後に、OracleAS Guard サーバーである ASG プロセスは起動されていないことに注意してください。OracleAS Guard を使用する場合は、インストール後に起動できます。

関連項目：『Oracle Application Server 高可用性ガイド』

1.5.2 Oracle HTTP Server の概要

次のインストール・タイプをインストールすると、Oracle HTTP Server がインストールおよび構成されます。

- 基本インストール
- 拡張インストール : Oracle HTTP Server のある Oracle WebCenter Framework
- 拡張インストール : Oracle Content Database
- 拡張インストール : Oracle HTTP Server

Oracle HTTP Server にアクセスするには、次の形式の URL を使用します。

`http://`*`hostname.domain`*`:`*`port`*

この例では、*port* は Oracle HTTP Server リスニング・ポート番号で、`opmnctl status -l` コマンドを実行した結果に表示されます。

次の例では、ポート 7777 を使用しています。

`http://`*`hostname.domain`*`:7777`

Oracle HTTP Server にアクセスすると、Oracle Application Server の「ようこそ」ページが表示されます。「**Application Server Control**」をクリックして、Application Server Control コンソールにログインします。次に、「Application Server: *server_name*」ホーム・ページにナビゲートして、HTTP サーバーのステータスを確認します。

関連項目：

- 『Oracle HTTP Server 管理者ガイド』
- 第 2.3 項「Application Server Control コンソールの概要」

1.5.3 Oracle Containers for J2EE（OC4J）の概要

Oracle Containers for J2EE（OC4J）は、完全な Java 2 Enterprise Edition（J2EE）環境です。

Oracle Application Server をインストールすると、`home` OC4J インスタンスが作成されます。これは、すべての中間層インストールで作成されるデフォルトの OC4J インスタンスです。Oracle WebCenter Framework をインストールすると、`OC4J_WebCenter` という OC4J インスタンスが作成されます。Oracle Content DB をインストールすると、`OC4J_Content` という別の OC4J インスタンスが作成されます。

このインスタンスのホーム・ページにナビゲートすると、Application Server Control コンソールを使用して OC4J インスタンスを管理できます。

関連項目：

- 『Oracle Containers for J2EE 構成および管理ガイド』
- 第 2.3.5 項「Application Server Control での OC4J インスタンスの表示と管理」

1.5.4 Oracle WebCenter Framework の概要

Oracle WebCenter Framework は Oracle WebCenter Suite の一部です。Oracle WebCenter Suite は、一連のツールおよびサービスであり、これを使用すると、トランザクション処理や分析を行うアプリケーションや動的アプリケーションを短時間で簡単に構築できます。ポートレット、コンテンツ、ランタイム・カスタマイズ、プレゼンス、コラボレーションおよび検索など、サービス指向アプリケーションの潜在能力を最大限に引き出すために必要なものがすべて用意されます。

Oracle WebCenter Framework を使用すると、ポータル機能を Java EE 5 アプリケーションに直接構築し、ポートレットやページのカスタマイズなどの従来のポータル機能をユーザーに用意できます。Oracle WebCenter Framework は、標準的な Faces 構造を使用して、これらのポータル機能を JavaServer Faces ページに直接組み込むことのできる最初の製品です。

関連項目：『Oracle WebCenter Framework チュートリアル』

1.5.5 Oracle Content DB の概要

Oracle Content DB は、統合されたデータベース中心のコンテンツ管理アプリケーションであり、これによって包括的な統合されたソリューションをファイルとドキュメントのライフサイクル管理において実現します。

Oracle Content DB の管理には Application Server Control コンソールを使用できます。「ようこそ」ページから Application Server Control コンソールにナビゲートします。「クラスタ・トポロジ」ページから、OC4J_Content インスタンスを開いて、「**コンテンツ**」をクリックします。そして、「**Content DB の拡張**」をクリックします。

関連項目：『Oracle Content Database Oracle WebCenter Suite 用管理者ガイド』

1.6 作業 5: SSL の有効化（オプション）

インストール時に、SSL は一部のコンポーネントに対して構成されません。SSL を有効にする場合は、第 IV 部「Secure Sockets Layer（SSL）」を参照してください。

2

管理ツールの概要

この章では、Oracle Application Server の管理ツールについて説明します。

この章の項目は次のとおりです。

2.1 Oracle Application Server の管理ツールの概要

アプリケーション・サーバー・コンポーネントの監視および管理に使用される手順は、組織の規模、採用する管理者の数、管理するコンポーネントのタイプなどによって異なります。そのため、Oracle Application Server インストールの管理には、複数のオプションが用意されています。

これらの管理オプションは、次のカテゴリに分類できます。

- Oracle Enterprise Manager 10g Application Server Control を使用した Oracle Application Server の管理
- OPMN コマンドラインを使用した Oracle Application Server の管理
- admin_client.jar ユーティリティを使用した Oracle Application Server の管理
- 組込みパフォーマンス・メトリックを監視する他のツールの使用

2.1.1 Oracle Enterprise Manager 10*g* Application Server Control を使用した Oracle Application Server の管理

Oracle Enterprise Manager 10*g* Application Server Control は、Oracle Application Server のすべてのインスタンスとともにインストールされます。したがって、Web ブラウザを使用して、アプリケーション・サーバーとそのコンポーネントの管理をただちに開始できます。

Application Server Control コンソールからは、単独の Oracle Application Server インスタンスだけでなく、複数のアプリケーション・サーバー・インスタンスとそれらの OC4J インスタンスにデプロイされたアプリケーションで構成されるクラスタ・トポロジも監視および管理できます。

Application Server Control コンソールでは、豊富な種類のパフォーマンス・データと管理機能が、アプリケーション・サーバーと Oracle Containers for J2EE の特定の機能について、個別の Web ベースのホーム・ページとして編成されます。Enterprise Manager ホーム・ページを使用すると、最も重要な監視データと最もよく使用される管理機能のすべてに、Web ブラウザから容易にアクセスできます。

関連項目： 第 2.2 項「Oracle Enterprise Manager 10g Application Server Control について」

2.1.2 OPMN コマンドラインを使用した Oracle Application Server の管理

Application Server Control のほかに、`opmnctl` コマンドライン・ツールも使用できます。このツールは、Oracle Process Manager and Notification Server（OPMN）へのコマンドライン・インタフェースを提供します。たとえば、このコマンドライン・ツール（`opmnctl`）を使用して、次のことを実行できます。

- アプリケーションおよびアプリケーション・サーバー・インスタンスの起動と停止
- クラスタを構成するための複数のアプリケーション・サーバー・インスタンスの関連付け
- クラスタに属するインスタンスのサマリーの表示

関連項目：

- 第 1.5.1 項「Oracle Process Manager and Notification Server（OPMN）の概要」
- `opmnctl` を使用した OC4J クラスタの構成および管理の詳細は、『Oracle Containers for J2EE 構成および管理ガイド』を参照してください。
- OPMN コマンドラインの詳細は、『Oracle Process Manager and Notification Server 管理者ガイド』を参照してください。

2.1.3 admin_client.jar ユーティリティを使用した Oracle Application Server の管理

Oracle Application Server には、アクティブな OC4J インスタンスで操作をする際に使用できる、admin_client.jar というコマンドライン・ユーティリティも用意されています。

admin_client.jar ユーティリティのほとんどの機能は、スタンドアロン構成の Oracle Application Server OC4J サーバー専用として使用される admin.jar ユーティリティのかわりに使用できます。

admin.jar ユーティリティとは異なり、admin_client.jar ユーティリティを使用すると、管理対象の Oracle Application Server 環境に存在する OC4J インスタンスだけでなく、スタンドアロン OC4J 環境の OC4J インスタンスも管理できます。

admin_client.jar ユーティリティを使用すると、次のタスクを実行できます。

- 特定の OC4J インスタンス、またはクラスタ内のすべてのインスタンスへのアプリケーションのデプロイ
- アプリケーションのアンデプロイ
- デプロイ後の EJB モジュールに対するクラス変更の増分更新
- 新規共有ライブラリの作成
- 特定の OC4J インスタンスまたはクラスタ全体での特定アプリケーションの停止、起動、再起動

関連項目：『Oracle Containers for J2EE 構成および管理ガイド』の admin_client.jar ユーティリティの使用に関する項

2.1.4 組込みパフォーマンス・メトリックを監視する他のツールの使用

Oracle Application Server をインストールして起動すると、アプリケーション・サーバーにより、一連の組込みパフォーマンス・メトリックの収集が自動的に開始されます。この組込みパフォーマンス・メトリックは、Oracle Application Server コンポーネントの実装に組み込まれたパフォーマンス・インストルメントを使用して、連続的に測定されます。

Application Server Control コンソールでは、これらのパフォーマンス・メトリックのサブセットが、アプリケーション・サーバー・コンポーネントのホーム・ページに整理された形態で表示されます。たとえば、OC4J のパフォーマンス・メトリックは、「OC4J のパフォーマンス」ページに一連のグラフとして表示されます。

これらのかわりに、すべての組込みパフォーマンス・メトリックを参照する必要がある場合や、アプリケーション・サーバー・コンポーネントの特定のメトリック・セットを監視する必要がある場合があります。Oracle Application Server には、Application Server Control コンソールを使用せずに、Oracle Application Server の組込みパフォーマンス・メトリックを直接表示するコマンドライン・ツールおよびサーブレットベースのツールも用意されています。

関連項目：『Oracle Application Server パフォーマンス・ガイド』

2.2 Oracle Enterprise Manager 10*g* Application Server Control について

Oracle Enterprise Manager 10*g* Application Server Control は、Oracle Application Server 10*g* リリース 3（10.1.3.*x*）専用に設計された Web ベースの管理機能を提供します。

次の各項でその詳細を説明します。

- 10g リリース 3（10.1.3.0.0）での Application Server Control の新機能
- Application Server Control 10g リリース 3（10.1.3.1）の新機能
- Application Server Control の基礎となるテクノロジについて
- Application Server Control コンソールのオンライン・ヘルプの使用

2.2.1 10*g* リリース 3（10.1.3.0.0）での Application Server Control の新機能

Oracle Application Server 10*g* リリース 3（10.1.3.0.0）の Application Server Control によって、次の利点と強化機能が導入されました。これらの機能は、10*g* リリース 3（10.1.3.1）および 10*g* リリース 3（10.1.3.2.0）のリリースにも含まれています。

- 軽量アーキテクチャ
- 標準ベースの管理
- リモート管理
- ロールベースの管理

注意： この後の各項では、10*g* リリース 3（10.1.3.*x*）は、10*g* リリース 3（10.1.3.0.0）、10*g* リリース 3（10.1.3.1）および 10*g* リリース 3（10.1.3.2.0）を意味します。

2.2.1.1 軽量アーキテクチャ

10*g* リリース 3（10.1.3.*x*）の Application Server Control は、作成する OC4J コンテナごとに実行される、標準の J2EE アプリケーション（`ascontrol`）としてデプロイされます。小規模なデプロイ環境では、Application Server Control をホストする OC4J インスタンスを使用して、カスタマ・アプリケーションをデプロイすることもできます。

この新しいアーキテクチャの採用により、Oracle Application Server の以前のリリースで Application Server Control に必要であった個別の Oracle Management Agent は不要となりました。その結果、管理ソフトウェアに必要なディスク領域およびシステム・リソースも削減されています。したがって、アプリケーション・サーバー環境への影響を最小限に抑えながら、アプリケーションを管理および監視できます。

2.2.1.2 標準ベースの管理

10*g* リリース 3（10.1.3.*x*）では、Application Server Control は Java Management Extension（JMX）テクノロジに基づいています。

> **関連項目：** Java Management Extension（JMX）テクノロジの概要：
>
> `http://java.sun.com/j2se/1.5.0/docs/guide/jmx/overview/JMXoverviewTOC.html`

具体的には、Application Server Control は、次の Java テクノロジ標準を実装しています。

- JSR 77 J2EE 管理仕様

  Application Server Control は、この仕様に基づいて開発および構成されたアプリケーションが指定するアプリケーション定義の MBean に加えて、10*g* リリース 3（10.1.3.*x*）OC4J コンテナが指定するコンテナ定義の MBean を利用します。これらの MBean は、構成、監視、状態管理の各機能を提供します。

  > **関連項目：** JSR 77 J2EE 管理仕様：
  >
  > `http://jcp.org/en/jsr/detail?id=077`

  さらに、Application Server Control は、JSR 77 とアプリケーションで定義された MBean 関連の操作（MBean 属性値と統計の表示、MBean メソッドの呼出し、JMX 通知のサブスクリプション、状態管理など）を完全にサポートする新しい MBean ブラウザを備えています。

  > **関連項目：** 第 2.3.6 項「MBean および Application Server Control MBean ブラウザについて」

- JSR 88 J2EE アプリケーション・デプロイメント仕様

  Application Server Control には、J2EE アプリケーションのデプロイおよび再デプロイを簡略化する JSR 88 ベースのデプロイ用ウィザード、デプロイ時に共通のデプロイメント・ディスクリプタの割当てやマッピングを補助するタスク指向のデプロイ・プラン・エディタ、および高度な構成のすべてのデプロイメント・ディスクリプタにアクセスできる汎用デプロイ・プラン・エディタが用意されています。

  関連項目： JSR 88 J2EE アプリケーション・デプロイメント仕様 :

  `http://www.jcp.org/en/jsr/detail?id=88`

- Java Naming and Directory Interface（JNDI）

  Application Server Control には、各アプリケーションの JNDI バインディングを階層的に表示する新しい JNDI ブラウザが用意されています。

  関連項目： Java Naming and Directory Interface（JNDI）の記述 :

  `http://java.sun.com/products/jndi/`

- Web サービスの管理

  Application Server Control では、OC4J インスタンスにデプロイされた Web サービスに対して、監査、ロギング、セキュリティおよび信頼性に関するパラメータを構成できます。また、Oracle Web Services Manager を、排他的な Web サービス管理ソリューションとして使用したり、監査、ロギング、信頼性およびセキュリティの標準管理機能に連動させて使用したりできます。

  関連項目： 10*g* リリース 3（10.1.3.2.0）での Web サービスの開発および管理の詳細は、『Oracle Application Server Web Services 開発者ガイド』を参照してください。

  Oracle Web Services Manager の詳細は、『Oracle Web Services Manager 管理者ガイド』を参照してください。

2.2.1.3 リモート管理

Oracle Application Server 10*g* リリース 3（10.1.3.*x*）では、Oracle Process Manager and Notification Server（OPMN）を使用して複数のアプリケーション・サーバー・インスタンスを関連付け、1 つの Oracle Application Server クラスタとして構成できます。このように環境を構成すると、Application Server Control のシングル・インスタンスを使用して、クラスタ内のすべてのインスタンスをリモート管理できます。

関連項目： 第 2.3.3.1 項「クラスタ・トポロジの表示とアクティブな Application Server Control の検索」

2.2.1.4 ロールベースの管理

一般的な本番環境のデータ・センターでは、管理操作（構成、アプリケーション開発、プロセスの制御と監視など）が、データ・センターの配置やセキュリティ・ポリシーに応じて、様々な管理者グループにより実行されています。

通常は、すべての権限を持つスーパー管理者が、特定の管理操作のみを実行できる、限られた権限を持つ他の管理者に管理操作を委任します。Application Server Control では、各ユーザーに、3 つの標準管理ロールの 1 つを割り当てることができます。

関連項目： 第 2.3.2 項「自分用およびチーム・メンバー用の管理アカウントの作成」

2.2.2 Application Server Control 10*g* リリース 3（10.1.3.1）の新機能

10*g* リリース 3（10.1.3.1）の Application Server Control には、次の新機能が導入されています。これらの機能は 10*g* リリース 3（10.1.3.2.0）でも使用できます。

- Application Server Control コンソールから OC4J インスタンスを作成および削除する機能
- Application Server Control コンソールからグループを作成および削除する機能
- Oracle Application Server クラスタ・トポロジのコンポーネントによって現在使用中のポートがすべて要約表示される、新しい「ランタイム・ポート」ページ
- アプリケーション・サーバー、OC4J インスタンスおよびデプロイ済アプリケーションに割り当てたルーティング ID を表示および変更できる、新しい「ルーティング ID」ページ
- クラスタ・トポロジのメンバーおよびプロパティの構成に使用できる、新しい「トポロジ・ネットワーク構成」ページ

詳細は、第 2.3 項「Application Server Control コンソールの概要」を参照してください。

2.2.3 Application Server Control の基礎となるテクノロジについて

Application Server Control コンソールは、Oracle Application Server 環境の検出、監視および管理において、関連性のある各テクノロジに依存しています。Application Server Control コンソールでオプションおよび機能を選択すると、これらのテクノロジによって、多くの管理タスクが自動的に実行されます。たとえば、各アプリケーション・サーバー・インスタンスのコンポーネントの検出、パフォーマンス・データの収集および処理、アプリケーションの構成情報へのアクセスの提供などが実現されます。

表 2-1 に、Application Server Control コンソールに利用されている基礎となるテクノロジの要約を示します。

表 2-1 Application Server Control の基礎となるテクノロジの要約

テクノロジ	説明
Dynamic Monitoring Service（DMS）	Application Server Control コンソールは、DMS を使用して、Oracle Application Server コンポーネントに関するパフォーマンス・データを収集します。 詳細は、『Oracle Application Server パフォーマンス・ガイド』を参照してください。
Oracle Process Manager and Notification Server（OPMN）	OPMN では、アプリケーション・サーバーのインスタンスとそのコンポーネントに対して、プロセスの制御と監視が実行されます。また、コンポーネントのステータス情報が収集され、その情報に関連するコンポーネントに配布されます。Application Server Control では OPMN を使用して、アプリケーション・サーバー・インスタンスのコンポーネントの起動や停止などのタスクを実行します。 詳細は、『Oracle Process Manager and Notification Server 管理者ガイド』を参照してください。

2.2.4 Application Server Control コンソールのオンライン・ヘルプの使用

Application Server Control コンソールの使用中であれば、いつでもページの一番上にある「**ヘルプ**」をクリックして、詳細を参照できます。ほとんどの場合、「ヘルプ」ウィンドウが表示され、現在のページに関するヘルプ・トピックが表示されます。「ヘルプ」ウィンドウの「**目次**」をクリックしてヘルプ・トピックのリストを参照したり、「**検索**」をクリックして特定の語や句を検索できます。

2.3 Application Server Control コンソールの概要

次の項では、Application Server Control コンソールの使用を開始し、Application Server Control コンソールの Enterprise Manager ホーム・ページについて理解を深めます。

- Application Server Control コンソールの表示
- 自分用およびチーム・メンバー用の管理アカウントの作成
- Application Server Control によるクラスタ・トポロジの管理
- Application Server Control でのアプリケーション・サーバー・コンポーネントの管理
- Application Server Control での OC4J インスタンスの表示と管理
- MBean および Application Server Control MBean ブラウザについて

2.3.1 Application Server Control コンソールの表示

次の項では、Application Server Control コンソールの表示方法について説明し、Application Server Control コンソールを初めて表示したときに最初に表示されるホーム・ページを紹介します。

- Application Server Control コンソール URL の使用
- 「ようこそ」ページからの Application Server Control コンソールの表示

2.3.1.1 Application Server Control コンソール URL の使用

ポート番号を含む Application Server Control コンソールの URL は、Oracle Application Server のインストール手順の最後に表示されるテキスト・ファイルに記載されています。このテキスト・ファイルは、アプリケーション・サーバーのインストール後に、次の場所に保存されます。

(UNIX) *ORACLE_HOME*/install/readme.txt
(Windows) *ORACLE_HOME*¥install¥readme.txt

通常、Application Server Control コンソールの URL は、ホスト・コンピュータの名前とインストール時に Application Server Control コンソールに割り当てられたポート番号で構成されます。たとえば UNIX では、次のように指定します。

```
http://mgmthost1.acme.com:7777/em
```

2.3.1.2「ようこそ」ページからの Application Server Control コンソールの表示

Oracle Application Server の「ようこそ」ページから Application Server Control コンソールを表示する手順は次のとおりです。

1. Web ブラウザに次の URL を入力して、Oracle Application Server の「ようこそ」ページを表示します。

   ```
   http://hostname.domain:port
   ```

   次に例を示します。

   ```
   http://sys42.acme.com:7777
   ```

   注意： Oracle HTTP Server のデフォルト・ポート（「ようこそ」ページ）は、Oracle Application Server のインストール手順の最後に通知されます。また、アプリケーション・サーバーの Oracle ホームの `install` ディレクトリにある次のテキスト・ファイルに記載されています。

   ```
   readme.txt
   ```

2. 「Oracle Enterprise Manager 10*g* Application Server Control コンソールにログイン」をクリックします。

   Enterprise Manager に管理者ログイン・ダイアログ・ボックスが表示されます。

3. Oracle Application Server 管理者のユーザー名とパスワードを入力して、「**OK**」をクリックします。

   管理者ユーザーのデフォルトのユーザー名は、`oc4jadmin` です。これは、Application Server Control コンソールへの初回のログイン時に使用可能なアカウントです。`oc4jadmin` パスワードは、Oracle Application Server のインストール時に指定したものです。

2.3.2 自分用およびチーム・メンバー用の管理アカウントの作成

デフォルトの `oc4jadmin` アカウントを使用して Application Server Control コンソールにログインしたら、次の手順に従って、自分用の新しい管理ユーザー・アカウントと、チーム内の各システム管理者用の管理ユーザー・アカウントをそれぞれ作成します。

`oc4jadmin` アカウントは、日常的な管理業務には使用しないことをお薦めします。`oc4jadmin` アカウントは、クラスタ管理用の資格証明として予約し、排他的に使用してください。詳細は、第 A.2.2 項「oc4jadmin アカウントについて」を参照してください。

タスク 1 自分用の新しい管理者アカウントの作成

1. Application Server Control コンソールの各ページの上部にある「**設定**」をクリックします。
2. 「**ユーザー**」をクリックして、「ユーザー」ページを表示します。
3. 「**作成**」をクリックして、新しい管理者アカウントを作成します。
4. 画面の指示に従って、アカウントの名前とデフォルトのパスワードを指定し、自分用の管理者アカウントに `ascontrol_admin` ロールを割り当てます。

   アカウント名には、名前のイニシャルと姓の組合せを使用することを検討してください（たとえば、`bsmith`）。

   すべての管理タスクを実行し、他の管理ユーザーを作成できるように、必ず ascontrol_admin ロールを割り当ててください。

タスク 2 チーム・メンバー用の追加管理アカウントの作成

前述の手順に従って、チームのメンバー用の追加ユーザー・アカウントを作成します。

各ユーザーに割当て可能な管理ロールの説明は、表 2-2 を参照してください。

表 2-2 Application Server Control 管理者に割り当てることができる管理ロール

ロール	説明
ascontrol_admin	このロールは、すべての管理権限を必要とし、Application Server Control アプリケーションおよび Application Server Control ページにアクセスする管理者に割り当てます。このような管理者は、Oracle Application Server および OC4J 環境を管理するために割り当てられたスーパーユーザーと見なされます。
ascontrol_appadmin	このロールは、デフォルトのアプリケーションと Application Server Control（ascontrol）アプリケーションを除き、デプロイされているすべてのアプリケーションを管理する必要がある管理者に割り当てます。この管理者は、デプロイされたアプリケーションの管理はできますが、新しい管理ユーザーを作成したり、グローバル構成を変更したりすることはできません。

表 2-2 Application Server Control 管理者に割り当てることができる管理ロール（続き）

ロール	説明
ascontrol_monitor	このロールは、Oracle Application Server および OC4J 環境を監視するが、アプリケーションや OC4J インスタンスの構成に変更を加える必要はない管理者に割り当てます。このロールは基本的に読取り専用です。 このロールは、作成するすべての管理ユーザーに自動的に適用されます。ascontrol_admin ロールも ascontrol_appadmin ロールも適用しなかった場合、そのアカウントは監視専用として使用できます。

2.3.3 Application Server Control によるクラスタ・トポロジの管理

Oracle Application Server をインストールし、Application Server Control コンソールにログインすると、最初に「クラスタ・トポロジ」ページが表示されます。次の項では、このページを使用した Oracle Application Server 環境の管理の概要について説明します。

- クラスタ・トポロジの表示とアクティブな Application Server Control の検索
- Application Server Control によるグループの管理
- クラスタ・トポロジの管理タスクの要約

関連項目：

- Oracle Application Server 10*g* リリース 3（10.1.3.1.0）のインストール時にクラスタを構成する方法の詳細は、Oracle Application Server のインストレーション・ガイドを参照してください。
- インストール後にクラスタを構成する方法の詳細は、第 6.2 項「クラスタ・トポロジの構成」を参照してください。

2.3.3.1 クラスタ・トポロジの表示とアクティブな Application Server Control の検索

Oracle Application Server をインストールし、Application Server Control にログインすると、「クラスタ・トポロジ」ページ（図 2-1）が最初に表示されます。このページは、Application Server Control コンソールで「**クラスタ・トポロジ**」をクリックしても表示できます。

この「クラスタ・トポロジ」ページには、クラスタにデプロイされているアプリケーション・サーバー、OC4J インスタンス、Web サービス、アプリケーションの詳細なビューが表示されます。

クラスタ内の各 OC4J インスタンスには、Application Server Control を表す `ascontrol` アプリケーションが 1 つずつ自動的に含まれています。ただし、クラスタ内のすべての Oracle Application Server インスタンスの管理に使用されるのは、1 つの Application Server Control のみです。

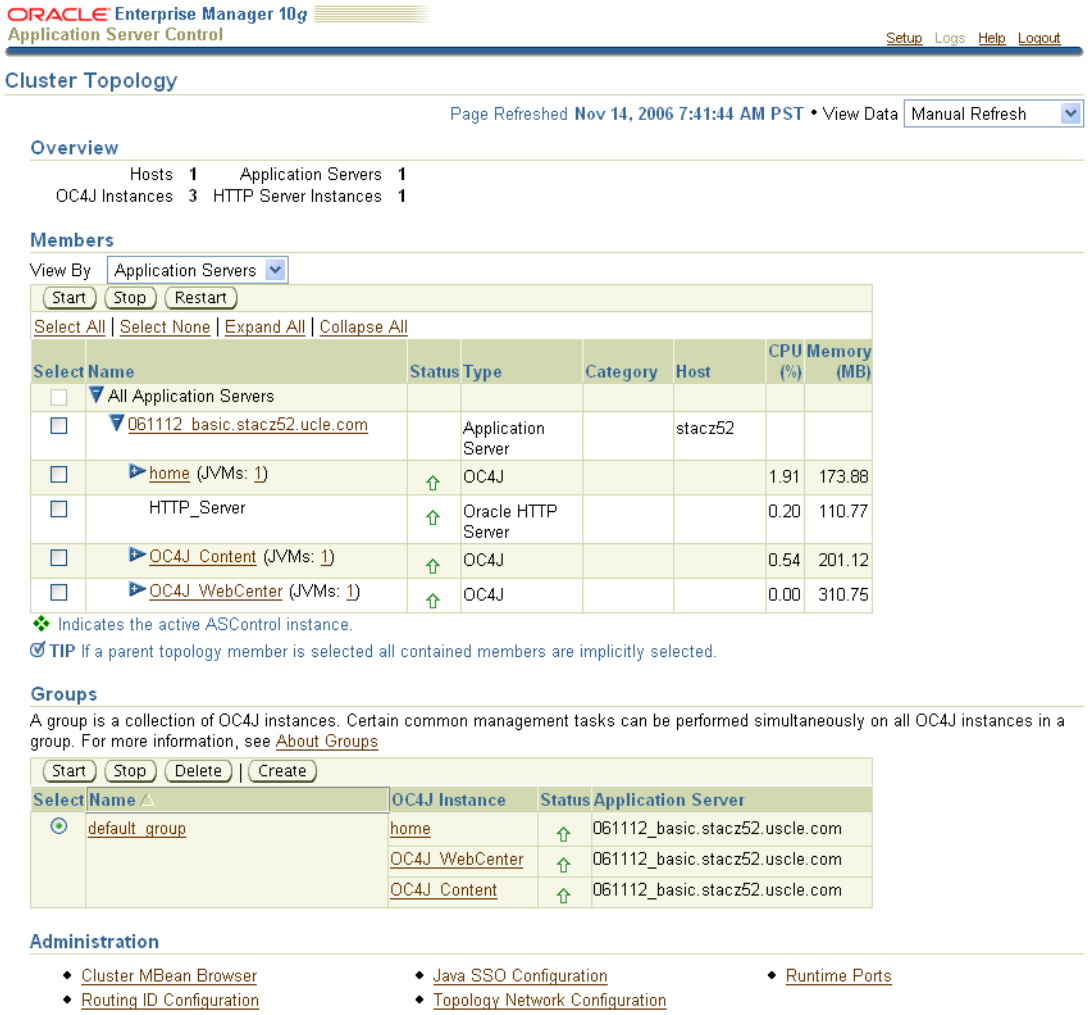
- クラスタの管理に使用されているアクティブな Application Server Control を識別するには、「**すべてを開く**」をクリックしてクラスタ内のすべてのコンポーネントを表示してから、アクティブな `ascontrol` アプリケーションを探します。これは、アクティブな Application Server Control アイコンによって見分けることができます。

アクティブな `ascontrol` アプリケーションのデプロイに使用される OC4J インスタンスは、管理 OC4J インスタンスと呼ばれます。管理 OC4J インスタンスは、Oracle Application Server のインストール時に指定できます。

関連項目： クラスタ・トポロジでのアクティブな Application Server Control の識別方法および構成方法の詳細は、第 A.6 項「アクティブな Application Server Control の管理」を参照してください。

独自のアプリケーションをデプロイしたときは、それらも「クラスタ・トポロジ」ページに表示されます。クラスタ内の OC4J インスタンスにデプロイされているすべてのアプリケーションを表示するには、このページの「メンバー」セクションの上にある「**表示方法**」から「**アプリケーション**」を選択します。Application Server Control によって、リストが特定のカテゴリに整理されます。

図 2-1 クラスタ・トポロジの管理



2.3.3.2 Application Server Control によるグループの管理

Oracle Application Server 10*g* リリース 3（10.1.3.2.0）の用途では、**グループ**とは、同じクラスタ・トポロジに属する OC4J インスタンスのセットです。グループ内のすべての OC4J インスタンスには、特定の構成操作を同時に実行できます。

次の各項で詳細を説明します。

- グループの表示と管理
- グループを使用する利点

2.3.3.2.1 グループの表示と管理 Oracle Application Server の初回インストール時には、デフォルト・グループが自動的に作成されます。この `default_group` には、インストール時に作成されたすべての OC4J インスタンスが含まれます。

インストール後は、クラスタ内の使用可能なグループが、「クラスタ・トポロジ」ページ（図 2-1）の「グループ」セクションに表示されます。このページの「グループ」セクションから、グループを起動、停止、削除および作成できます。また、グループ名をクリックし、表示される「グループ」ページを使用することもできます。「グループ」ページでは、次の操作を実行できます。

- OC4J インスタンスのグループ内外への移動
- アプリケーションのグループへのデプロイ
- グループに対する特定の管理タスクの実行

すべての OC4J インスタンスがグループに所属する必要があります。結果として、新しい OC4J インスタンスを作成するときは、そのインスタンスが所属するグループの指定が必要になります。

関連項目：

- 第 6.2.3 項「追加グループの作成」
- Application Server Control のオンライン・ヘルプのグループに関する項

2.3.3.2.2 グループを使用する利点 グループを使用すると、複数の OC4J インスタンスに対して、いくつかの一般的な管理タスクを自動的に実行できます。

「グループ」ページから複数の OC4J インスタンスに実行できるタスクには、次のものがあります。

- 開始、停止、再起動などのプロセス管理操作
- デプロイ、アンデプロイ、再デプロイなどのデプロイ操作
- JDBC データソースと接続プールの作成、変更、削除などの JDBC 管理操作
- JMS 宛先の作成、削除、JMS 接続ファクトリの作成、変更、削除などの JMS プロバイダ操作

「グループ」ページを表示するには、「クラスタ・トポロジ」ページの「グループ」セクションで、グループの名前をクリックします。

関連項目： Application Server Control のオンライン・ヘルプの OC4J インスタンスとグループの作成ガイドラインに関する項

2.3.3.3 クラスタ・トポロジの管理タスクの要約

「クラスタ・トポロジ」ページにある「管理」セクションから、クラスタ全体に対する一連の管理タスクを実行できます。表 2-3 に、クラスタ・トポロジの管理タスクを要約します。

表 2-3 クラスタ・トポロジの管理タスクの要約

タスク	説明	詳細情報の参照先
クラスタ MBean ブラウザ	クラスタ MBean ブラウザを表示します。このブラウザを使用して、クラスタ全体の操作に固有の管理対象 Bean の階層を表示できます。	第 2.3.6 項「MBean および Application Server Control MBean ブラウザについて」
ルーティング ID 構成	Oracle Application Server クラスタのコンポーネントに割り当てられているルーティング ID を表示または変更します。	Application Server Control のオンライン・ヘルプのルーティング ID の変更に関する項

表 2-3 クラスタ・トポロジの管理タスクの要約（続き）

タスク	説明	詳細情報の参照先
Java SSO 構成	デプロイ済のアプリケーションに OC4J Java Single Sign-On（Java SSO）の使用を構成します。Java SSO は、OC4J に付属する、追加のインフラストラクチャを必要としない軽量のシングル・サインオン・ソリューションです。 基本インストールを選択すると、Java SSO は自動的にデプロイ、構成および起動されます。拡張インストールを選択すると、Java SSO はデプロイされますが、構成および起動は行われません。	『Oracle Containers for J2EE セキュリティ・ガイド』の Java SSO の設定および構成に関する項
トポロジ・ネットワーク構成	現行の Oracle Application Server クラスタ・トポロジのメンバーおよびプロパティを構成します。	第 6.2 項「クラスタ・トポロジの構成」 Application Server Control のオンライン・ヘルプのサポートされているクラスタ・トポロジの要約に関する項
ランタイム・ポート	Oracle Application Server クラスタのコンポーネントによって使用されているポートを表示および変更します。	第 4 章「ポートの管理」

2.3.4 Application Server Control でのアプリケーション・サーバー・コンポーネントの管理

「クラスタ・トポロジ」ページに慣れたら、特定のアプリケーション・サーバー・インスタンスのホーム・ページにドリルダウンできます。

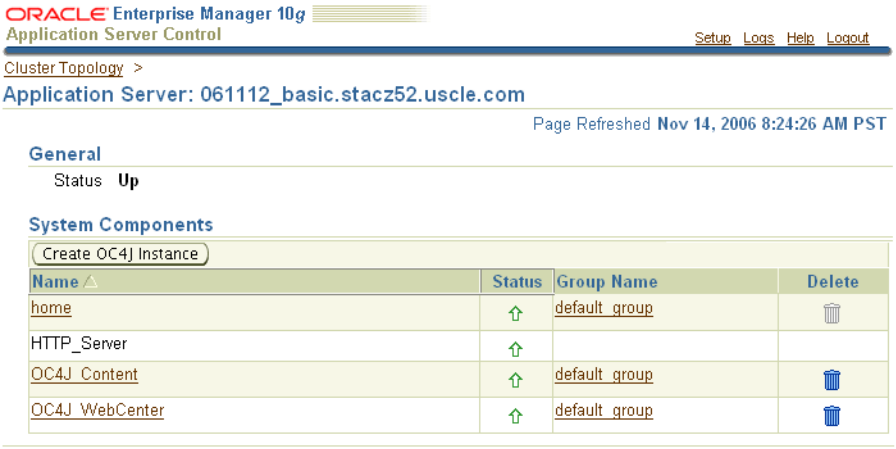
「クラスタ・トポロジ」にあるアプリケーション・サーバー・インスタンス名をクリックして、「アプリケーション・サーバー」ページを表示します。たとえば、図 2-1 では、「**061112_basic.stacz52.ucle.com**」をクリックします。

図 2-2 に示すような「アプリケーション・サーバー」ページが表示されます。このページには、このインスタンスに作成された OC4J インスタンスや Oracle HTTP Server（この Oracle Application Server インスタンスにインストールされている場合）などの、アプリケーション・サーバー・インスタンスのコンポーネントが一覧表示されます。

「アプリケーション・サーバー」ページに表示される実際のコンポーネントのリストは、選択したインストール・タイプに応じて異なります。

「**OC4J インスタンスの作成**」ボタンをクリックして、このアプリケーション・サーバー・インスタンスに新しい OC4J インスタンスを作成します。詳細は、第 6.1 項「OC4J インスタンスの追加と削除」を参照してください。

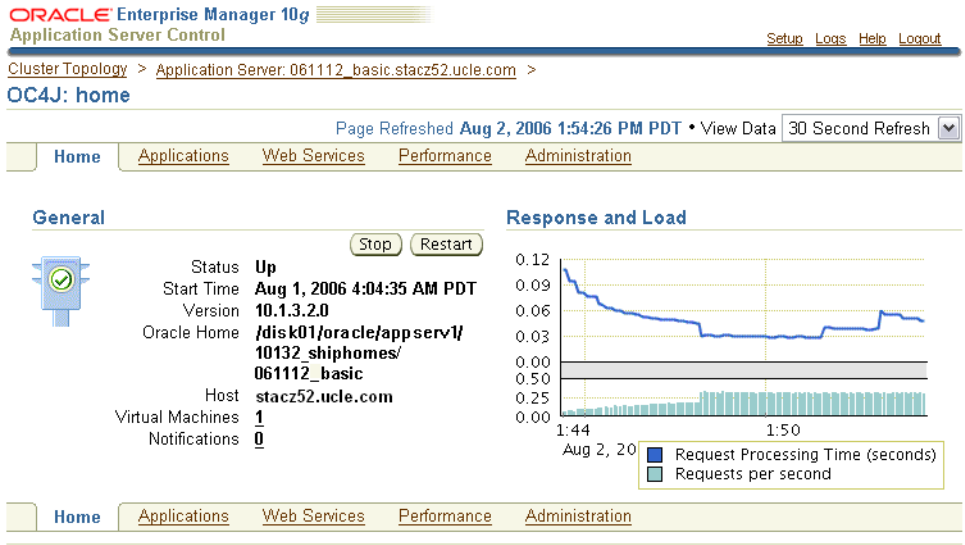
図 2-2 アプリケーション・サーバー・インスタンスのコンポーネントの表示



2.3.5 Application Server Control での OC4J インスタンスの表示と管理

「クラスタ・トポロジ」ページまたは「アプリケーション・サーバー」ページで OC4J インスタンスの名前をクリックすると、OC4J ホーム・ページ（図 2-3）が表示されます。

図 2-3 OC4J ホーム・ページからの OC4J インスタンスの管理



OC4J ホーム・ページは、OC4J インスタンスの一般情報の取得、およびレスポンスと負荷に関するグラフの表示に使用します。一定期間のレスポンスと負荷のメトリックを監視するには、「**データの表示**」ドロップダウン・メニューからリフレッシュ間隔（たとえば、「**30 秒リフレッシュ**」）を選択します。OC4J インスタンスを起動、停止または再起動するには、「クラスタ・トポロジ」ページにナビゲートし、インスタンスを選択してから、「**起動**」、「**停止**」または「**再起動**」をクリックします。

OC4J ホーム・ページと、関連する「**アプリケーション**」、「**Web サービス**」、「**パフォーマンス**」および「**管理**」ページは、OC4J インスタンスとそのインスタンスにデプロイされているアプリケーションおよび Web サービスを、中央から Web ベースで表示できるように設計されています。

OC4J ホーム・ページの使用中は、いつでも「**ヘルプ**」をクリックして詳細を参照できます。オンライン・ヘルプには、各ページのフィールドに関する参照情報のほか、初心者に役立つ関連タスクや関連ドキュメントへのリンクが掲載されています。

2.3.6 MBean および Application Server Control MBean ブラウザについて

管理対象 Bean（MBean）とは、分散環境内の JMX で管理されるリソースを表す Java オブジェクトです。MBean には、アプリケーション、サービス、コンポーネント、デバイスなどがあります。

MBean は、J2EE 環境のアプリケーションの管理に使用する標準インタフェースを作成するための一連の仕様、すなわち Java Management Extension（JMX）の一部である、J2EE 管理仕様（JSR-77）で定義されています。

MBean を作成してアプリケーションとともに OC4J にデプロイすると、アプリケーションまたはアプリケーションのコンポーネントを、Application Server Control コンソールを使用して管理および監視できるようになります。

> **関連項目：** 『Oracle Containers for J2EE 構成および管理ガイド』の OC4J での MBeans の使用に関する項

Application Server Control には、OC4J インスタンス、クラスタ、または選択したアプリケーションの MBean を参照できる、一連の MBean ブラウザが用意されています。MBean ブラウザでは、特定の監視タスクや構成タスクも実行できます。

次の各項で詳細を説明します。

- システム MBean ブラウザの表示
- 選択したアプリケーションの MBean の表示
- クラスタ MBean ブラウザの表示

2.3.6.1 システム MBean ブラウザの表示

選択した OC4J インスタンス固有の MBean を一覧するシステム MBean ブラウザを表示する手順は次のとおりです。

1. OC4J インスタンスの OC4J ホーム・ページにナビゲートします。
2. 「**管理**」をクリックして、OC4J の「管理」ページを表示します。
3. 表の「**システム MBean ブラウザ**」行にあるタスク・アイコンをクリックします。

   Enterprise Manager にシステム MBean ブラウザが表示されます。システム MBean ブラウザの使用方法の詳細を確認するには、「**ヘルプ**」をクリックします。

   オンライン・ヘルプには、MBean ブラウザの使用方法のオンライン・デモなど、MBean ブラウザを説明するツアーのトピックも用意されています。

   > **関連項目：** Application Server Control のオンライン・ヘルプの MBean ブラウザの構造に関する項

2.3.6.2 選択したアプリケーションの MBean の表示

特定のアプリケーションの MBean を表示する手順は次のとおりです。

1. OC4J ホーム・ページで「**アプリケーション**」をクリックして、OC4J インスタンスにデプロイされたアプリケーションのリストを表示します。
2. デプロイされたアプリケーションの名前をクリックします。
3. 「**管理**」をクリックして、アプリケーションの管理ページを表示します。このページには、選択したアプリケーションに実行できる様々な管理タスクの表が表示されます。
4. 表の中の適切なタスク・アイコンをクリックして、選択したアプリケーションに関連付けられているシステム MBean、またはそのアプリケーションに定義されている MBean を表示します。

   Enterprise Manager に、選択した MBean ブラウザのページが表示されます。

2.3.6.3 クラスタ MBean ブラウザの表示

クラスタ・トポロジに関連付けられた MBean を表示するには、「クラスタ・トポロジ」ページで、「**クラスタ MBean ブラウザ**」をクリックします。

Enterprise Manager にクラスタ MBean ブラウザが表示されます。クラスタの管理に使用されている MBean と、クラスタ内に定義されているグループはハイライト表示されます。

3

起動と停止

この章では、Oracle Application Server を起動および停止する手順について説明します。

この章の項目は次のとおりです。

3.1 起動および停止手順の概要

Oracle Application Server は、使用者の要件に応じて、様々な方法で起動および停止できる柔軟性のある製品です。次の各項を参照してください。

- 第 3.2 項「アプリケーション・サーバー・インスタンスの起動と停止」

  ホストの再起動後など、最初からインスタンスを起動するとき、またはシステムの停止の前など、インスタンス全体を停止するときは、この項の手順に従います。

- 第 3.3 項「コンポーネントの起動と停止」

  インスタンスを起動後に個々のコンポーネントを起動または停止するときは、この項の手順を使用します。

- 第 3.4 項「Oracle Application Server 環境の起動と停止」

  この項では、環境全体を正しい順序で停止する方法について説明します。

3.2 アプリケーション・サーバー・インスタンスの起動と停止

この項では、アプリケーション・サーバー・インスタンスを起動および停止する方法について説明します。この項の項目は次のとおりです。

- 中間層インスタンスの起動
- 中間層インスタンスの停止

中間層インスタンスに接続された 10.1.4 または 10.1.2 の OracleAS Infrastructure を含む環境全体に対する停止手順の詳細は、第 3.4 項を参照してください。

3.2.1 中間層インスタンスの起動

この項では、中間層インスタンスのすべてのプロセスを起動する方法について説明します。この手順は、ホストの再起動後やインスタンス全体の起動時に実行できます。

中間層インスタンスを起動するには、次の手順を実行します。

1. 中間層インスタンスが Oracle Identity Management などの OracleAS Infrastructure サービスに関連付けられている場合は、それらのサービスが起動されていることを確認します。詳細は、第 3.4.3 項を参照してください。
2. 中間層コンポーネントを起動します。

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
   (Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
   ```

   このコマンドでは、OPMN とすべての OPMN 管理プロセス（Oracle HTTP Server、OC4J インスタンス、Application Server Control コンソールなど）が起動されます。

また、Windows では「スタート」メニューから中間層を起動できます。たとえば、Windows 2000 で Oracle Application Server を起動するには、「**スタート**」→「**プログラム**」→「**Oracle - *Oracle_home_name***」→「**Oracle Process Manager**」→「**Start Oracle Process Manager**」を選択します。これにより、OPMN および OPMN で管理されているすべてのプロセスが起動します。

3.2.2 中間層インスタンスの停止

この項では、中間層インスタンスのすべてのプロセスを停止する方法について説明します。この手順は、ホストの停止時やインスタンス全体の停止時に実行できます。

中間層インスタンスを停止するには、次のコマンドを使用します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopall
```

このコマンドでは、OPMN とすべての OPMN 管理プロセス（Oracle HTTP Server、OC4J インスタンス、Application Server Control コンソールなど）が停止されます。

また、Windows では「スタート」メニューから中間層を停止できます。たとえば、Windows 2000 で Oracle Application Server インスタンスを停止するには、「**スタート**」→「**プログラム**」→「**Oracle - *Oracle_home_name***」→「**Oracle Process Manager**」→「**Stop Oracle Process Manager**」を選択します。これにより、OPMN および OPMN で管理されているすべてのプロセスが停止します。

3.3 コンポーネントの起動と停止

コンポーネントの起動、停止、再起動およびステータスの表示には、次のツールを使用できます。

- `opmnctl` コマンド。第 3.3.1 項を参照してください。
- Application Server Control コンソール。第 3.3.2 項を参照してください。

これらの両ツールはプロセス管理の基盤となるテクノロジとして OPMN を使用しているため、用途に応じて交換して使用できます。たとえば、`opmnctl` を使用してコンポーネントを起動し、Application Server Control コンソールを使用して停止できます。

ただし、この 2 つのツールは互換性がある一方で、異なる機能も備えています。`opmnctl` コマンドでは、コンポーネント全体に加えて、コンポーネントのサブプロセスを起動および停止できます。Application Server Control コンソールには、起動または停止のステータスが他のコンポーネントに依存しており、直接起動および停止できないコンポーネントを表示できます。

3.3.1 opmnctl を使用したコンポーネントの起動と停止

コンポーネントを起動および停止するには、`opmnctl` コマンドライン・ツールを使用できます。これは次のディレクトリにあります。

```
(UNIX) ORACLE_HOME/opmn/bin
(Windows) ORACLE_HOME¥opmn¥bin
```

コンポーネントを起動、停止または再起動する場合の opmnctl コマンドは次のとおりです。

```
opmnctl stopproc ias-component=component
opmnctl startproc ias-component=component
opmnctl restartproc ias-component=component
```

コンポーネントのサブプロセスを起動、停止または再起動するコマンドは次のとおりです。

```
opmnctl stopproc process-type=process
opmnctl startproc process-type=process
opmnctl restartproc process-type=process
```

Application Server Control のようなアプリケーションを起動、停止または再起動するコマンドは次のとおりです。

```
opmnctl startproc application=app_name
opmnctl stopproc application=app_name
opmnctl restartproc application=app_name
```

コンポーネントおよびプロセスのステータスを表示するコマンドは次のとおりです。

```
opmnctl status -l
```

opmnctl の使用方法の詳細は、『Oracle Process Manager and Notification Server 管理者ガイド』を参照してください。

3.3.2 Application Server Control コンソールを使用したコンポーネントの起動と停止

Application Server ホーム・ページで、コンポーネントの起動、停止、再起動およびステータスの表示ができます。

1. Application Server Control コンソールで「クラスタ・トポロジ」ホーム・ページにナビゲートします。
2. 「メンバー」セクションの「**表示方法**」で、「**アプリケーション・サーバー**」を選択します。
3. 「**選択**」列で、起動、停止または再起動するコンポーネントを選択します（すべてのコンポーネントを表示するには、「**すべてを開く**」をクリックします）。
4. 「メンバー」セクションの右上にある「**起動**」、「**停止**」または「**再起動**」ボタンをクリックします。

各コンポーネントのホーム・ページを使用して、個々のコンポーネントを起動および停止することもできます。

3.4 Oracle Application Server 環境の起動と停止

この項では、Oracle Application Server 環境を起動および停止する手順について説明します。1 つの環境は、複数のホストに分散配置された複数の OracleAS Infrastructure および中間層インスタンスで構成できます。これらのインスタンスは相互に依存するため、正しい順序で起動および停止することが重要です。

Oracle Application Server 環境を完全に停止する必要がある場合は、次の手順を使用できます。環境を完全にバックアップする場合や、パッチを適用する場合などは、この手順を使用します。

この項の項目は次のとおりです。

> **注意：** この章では、OracleAS Infrastructure は、リリース 10.1.4 またはリリース 2（10.1.2）の OracleAS Infrastructure を表しています。
>
> 10*g* リリース 3（10.1.3.2.0）の中間層インスタンスと OracleAS Infrastructure との関連付けの詳細は、第 6.6 項を参照してください。

3.4.1 Oracle Application Server 環境の起動

Oracle Application Server 環境を起動する手順は次のとおりです。

1. OracleAS Metadata Repository のみを含む OracleAS Infrastructure を起動します。

   環境に OracleAS Metadata Repository のみを含む複数の OracleAS Infrastructure インストールがある場合は、それらを任意の順序で起動します。このインストール・タイプでは、OracleAS Metadata Repository のみを起動する必要がある点に注意してください。`opmnctl` でプロセスを起動する必要はありません。Application Server Control コンソールを起動する必要もありません。詳細は、第 3.4.3 項「10.1.4 または 10.1.2 の OracleAS Infrastructure の起動」を参照してください。

2. Oracle Identity Management を含む OracleAS Infrastructure を起動します。

   Oracle Identity Management を環境で使用している場合は、Oracle Internet Directory を含む OracleAS Infrastructure を起動します。この OracleAS Infrastructure で OracleAS Metadata Repository を使用している場合は、Oracle Internet Directory を起動する前に OracleAS Metadata Repository を起動します。詳細は、第 3.4.3 項「10.1.4 または 10.1.2 の OracleAS Infrastructure の起動」を参照してください。

3. 中間層インスタンスを起動します。

   中間層インスタンスを任意の順序で起動します。詳細は、第 3.2.1 項「中間層インスタンスの起動」を参照してください。

3.4.2 Oracle Application Server 環境の停止

Oracle Application Server 環境のすべてのプロセスを停止する手順は次のとおりです。

1. 中間層インスタンスを停止します。

   中間層インスタンスが環境にある場合は、それらを任意の順序で停止します。詳細は、第 3.2.2 項「中間層インスタンスの停止」を参照してください。

2. Oracle Identity Management を含む OracleAS Infrastructure を停止します。

   Oracle Identity Management を環境で使用している場合は、Oracle Internet Directory を含む OracleAS Infrastructure を停止します。この OracleAS Infrastructure に OracleAS Metadata Repository が含まれる場合は、OracleAS Metadata Repository も停止します。

3. OracleAS Metadata Repository のみを含む OracleAS Infrastructure インスタンスを停止します。

   環境に OracleAS Metadata Repository のみを含む複数の OracleAS Infrastructure インスタンスがある場合は、それらを任意の順序で停止します。詳細は、第 3.4.4 項「10.1.4 または 10.1.2 の OracleAS Infrastructure の停止」を参照してください。

3.4.3 10.1.4 または 10.1.2 の OracleAS Infrastructure の起動

中間層がリリース 10.1.4 またはリリース 2（10.1.2）の OracleAS Infrastructure インスタンスに接続されている場合、そのインスタンスから Infrastructure を起動できます。

この手順は、OracleAS Infrastructure のすべてのタイプに適用できます。

- Oracle Identity Management および OracleAS Metadata Repository

  Oracle Identity Management および OracleAS Metadata Repository を起動する場合は、両方の手順を実行します。

- OracleAS Metadata Repository のみ

  OracleAS Metadata Repository を起動する場合は、ステップ 1 のみを実行します。OracleAS Metadata Repository のみのインストールでは、OPMN または Application Server Control コンソールが必要ないため、Oracle Identity Management を起動する 2 番目の手順は実行する必要がありません。

- Oracle Identity Management のみ

  Oracle Identity Management を起動する場合は、ステップ 2 のみを実行します。Oracle Identity Management をサポートする OracleAS Metadata Repository（別の Oracle ホームにあります）がすでに起動されている必要があります。

OracleAS Infrastructure を起動する手順は次のとおりです。

1. OracleAS Infrastructure に OracleAS Metadata Repository が含まれる場合は、次の手順に従い起動します。

   a. ORACLE_HOME 環境変数を OracleAS Infrastructure の Oracle ホームに設定します。

   b. ORACLE_SID 環境変数を OracleAS Metadata Repository SID（デフォルトは `orcl`）に設定します。

   c. Net リスナーを起動します。

      ```
      ORACLE_HOME/bin/lsnrctl start
      ```

d. OracleAS Metadata Repository インスタンスを起動します。

```
ORACLE_HOME/bin/sqlplus /nolog
SQL> connect SYS as SYSDBA
SQL> startup
SQL> quit
```

e. Oracle Enterprise Manager 10*g* Database Control を起動します。

```
(UNIX) ORACLE_HOME/bin/emctl start dbconsole
(Windows) ORACLE_HOME¥bin¥emctl start dbconsole
```

2. OracleAS Infrastructure に Oracle Identity Management が含まれる場合は、次の手順に従い起動します。

a. コンポーネントを起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

このコマンドでは、OPMN とすべての OPMN 管理プロセス（DCM、Oracle HTTP Server、OC4J インスタンス、Oracle Internet Directory など）が起動されます。

b. Application Server Control コンソールを起動します。

```
(UNIX) ORACLE_HOME/bin/emctl start iasconsole
(Windows) ORACLE_HOME¥bin¥emctl start iasconsole
```

また、Windows では、コントロール パネルの「サービス」を使用して Application Server Control コンソールを起動することもできます。Application Server Control のサービス名の形式は次のとおりです。

```
OracleOracleHomeNameASControl
```

Application Server Control の起動方法の詳細は、第 A.1 項を参照してください。

また、Windows では、「**スタート**」→「**プログラム**」→「**Oracle Application Server Infrastructure - *Infra_name***」→「**Start *instanceName***」を選択して、プログラム・メニューから Infrastructure を起動することもできます。

3.4.4 10.1.4 または 10.1.2 の OracleAS Infrastructure の停止

中間層がリリース 10.1.4 またはリリース 2（10.1.2）の OracleAS Infrastructure インスタンスに接続されている場合、そのインスタンスから Infrastructure を停止できます。

この手順は、OracleAS Infrastructure のすべてのタイプに適用できます。

- Oracle Identity Management および OracleAS Metadata Repository

  Oracle Identity Management および OracleAS Metadata Repository を停止する場合は、両方の手順を実行します。

- OracleAS Metadata Repository のみ

  OracleAS Metadata Repository を停止する場合は、ステップ 2 のみを実行します。

- Oracle Identity Management のみ

  Oracle Identity Management を停止する場合は、ステップ 1 のみを実行します。

OracleAS Infrastructure を停止する手順は次のとおりです。

1. OracleAS Infrastructure に Oracle Identity Management が含まれる場合は、次の手順に従い停止します。

a. Application Server Control コンソールを停止します。

```
(UNIX) ORACLE_HOME/bin/emctl stop iasconsole
(Windows) ORACLE_HOME¥bin¥emctl stop iasconsole
```

また、Windows では、コントロール パネルの「サービス」を使用して Application Server Control コンソールを停止することもできます。詳細は、第 A.1 項を参照してください。

b. コンポーネントを停止します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopall
```

このコマンドでは、OPMN とすべての OPMN 管理プロセス（DCM、Oracle HTTP Server、OC4J インスタンス、Oracle Internet Directory など）が停止されます。

2. OracleAS Infrastructure に OracleAS Metadata Repository が含まれる場合は、次の手順に従い停止します。

a. ORACLE_HOME 環境変数を OracleAS Infrastructure の Oracle ホームに設定します。

b. ORACLE_SID 環境変数を OracleAS Metadata Repository SID（デフォルトは `orcl`）に設定します。

c. OracleAS Metadata Repository インスタンスを停止します。

```
ORACLE_HOME/bin/sqlplus /nolog
SQL> connect SYS as SYSDBA
SQL> shutdown
SQL> quit
```

d. Net リスナーを停止します。

```
ORACLE_HOME/bin/lsnrctl stop
```

e. Oracle Enterprise Manager 10*g* Database Control を停止します。

```
(UNIX) ORACLE_HOME/bin/emctl stop dbconsole
(Windows) ORACLE_HOME¥bin¥emctl stop dbconsole
```

また、Windows では、「**スタート**」→「**プログラム**」→「**Oracle Application Server Infrastructure - *Infra_name***」→「**Stop *instanceName***」を選択して、プログラム・メニューから Infrastructure を停止することもできます。

3.5 起動と停止 : 特殊なトピック

この項では、Oracle Application Server の起動および停止に関する特殊なトピックについて説明します。この項の項目は次のとおりです。

- 高可用性環境での起動と停止
- コンポーネントの使用可能および使用不可の設定
- 複数インスタンス起動時の OC4J エラーの解決

3.5.1 高可用性環境での起動と停止

次に示す高可用性環境では、起動および停止について特別な検討と手順が必要です。

- Oracle Application Server Cluster
- Oracle Application Server Cold Failover Cluster
- Oracle Application Server Disaster Recovery （ASG コンポーネントの起動と停止を含む）

関連項目： 高可用性環境における起動と停止の詳細は、『Oracle Application Server 高可用性ガイド』を参照してください。

3.5.2 コンポーネントの使用可能および使用不可の設定

コンポーネントは、使用可能または使用不可に設定できません。ただし、第 6.1 項の説明にあるように、OC4J インスタンスを作成または削除することはできます。

3.5.3 複数インスタンス起動時の OC4J エラーの解決

1 つのホスト上に複数の Oracle Application Server インストールがあり、それらを同時に起動する場合、OPMN によって次のようなエラーが返されることがあります。

```
<process-type id="my_OC4J_instance">
  <process-set id="default_island">
    <process id="93388820" pid="24711" status="Stopped" index="1"
     log="/disk1/oracleas/opmn/logs/OC4J~my_OC4J_instance~default_island~1"
 operation="request" result="failure">
      <msg code="-21" text="failed to restart a managed process
        after the maximum retry limit">
      </msg>
```

このエラーは、OC4J インスタンス（my_OC4J_instance）の起動に失敗したことを示しています。この問題は、同一ホスト上にある 2 つの異なる Oracle ホームで、RMI、JMS および AJP ポート用に同じポート範囲を使用しており、一方の Oracle ホームの OC4J インスタンスが他方の Oracle ホームの OC4J インスタンスと同じポートを使用しようとしたことが原因である可能性があります。

たとえば、1 つのホスト上に 2 つの Oracle Application Server インストールがあり、ORACLE_HOME1 と ORACLE_HOME2 に配置されているとします。各インストールには 1 つ以上の OC4J インスタンスがあり、そのそれぞれの OC4J インスタンスに、AJP、RMI および JMS ポート用のポート範囲が割り当てられます。

OC4J ポート範囲の割当ては、両方の Oracle ホームの opmn.xml ファイルを調べることによってチェックできます。

ORACLE_HOME1/opmn/conf/opmn.xml
ORACLE_HOME2/opmn/conf/opmn.xml

各ファイルで、次のような行で始まる OC4J インスタンス・エントリを検索します。

```
<process-type id="home" module-id="OC4J" ... >
```

各エントリ内で、RMI、JMS および AJP ポート範囲を検索します。次に例を示します。

```
<port id="ajp" range="12501-12600"/>
<port id="rmi" range="12401-12500"/>
<port id="jms" range="12601-12700"/>
```

表 3-1 に、2 つの Oracle ホームに同じ OC4J ポートが割り当てられている問題の例を示します。この例では、ORACLE_HOME1 の AJP、RMI および JMS ポート範囲が、ORACLE_HOME2 の AJP、RMI および JMS ポート範囲と同じになっています（この例では、問題になっている opmn.xml の行のみを示しています）。

表 3-1 2 つの Oracle ホームにおける同一ポート範囲の例

ORACLE_HOME1/opmn/conf/opmn.xml における OC4J ポート範囲	*ORACLE_HOME2*/opmn/conf/opmn.xml における OC4J ポート範囲
<ias-component id="OC4J"> ... <process-type id="home" ... > ... <port id="default-web-site" range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> ... </process-type> <process-type id="OC4J_WebCenter" ... > ... <port id="default-web-site" range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> </process-type>	<ias-component id="OC4J"> ... <process-type id="home" ... > ... <port id="default-web-site range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> ... </process-type> <process-type id="OC4J_WebCenter" ... > ... <port id="default-web-site" range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> </process-type>

Oracle Application Server インスタンス内のすべての OC4J インスタンスに対するポート割当ては、OPMN により制御されます。そのため、1 つの opmn.xml ファイル内でポート範囲が重複していても問題はありません。しかし、1 つのホスト上の 2 つの OPMN によってプロセスが同時に起動されたときは、両プロセス間でポート使用が調整されません。

OPMN でポート割当てに使用されるアルゴリズムは次のとおりです。

1. ローカル・インスタンスの OPMN により管理されるプロセスに対し現在、割当て済となっていないポート範囲からポートを選択します。
2. 選択したポートを割り当てる前に、バインドすることによってそのポートが使用されているかどうかをチェックします。
3. そのポートが使用されていない場合（OPMN によってバインドできた場合）は、バインドを解除して、そのポートをプロセス（OC4J インスタンスなど）に割り当てます。これにより、そのポートがプロセスにバインドされ、この割当て情報で内部データ構造が更新されます。

OPMN がそのポートからバインド解除され、割当てプロセスがそのポートにバインドされる間に、別のプロセスがそのポートにバインドされる可能性があります。たとえば、ホスト上の他の OPMN プロセスや、同じポート番号に偶然バインドしようとした他の任意のプロセスが対象となります。

ポート範囲の割当てが複数の Oracle ホームで同一である場合で、この項の最初に示したようなエラーを受け取ったときは、2 つの OPMN プロセスがそれぞれの OC4J インスタンスに同じポートをバインドしようとしたことが原因として考えられます。この問題を完全に取り除く方法はありませんが（OPMN 以外のプロセスが同時に同じポートにバインドしようとする可能性は少ないながらもあるため）、OPMN を再構成することによって、このエラーが発生する可能性を減らすことはできます。

この問題を解決する方法には、次の 2 つがあります。

- オプション 1: 各 Oracle ホームに一意のポート範囲を割り当てる
- オプション 2: OC4J インスタンスの起動の最大試行回数を増やす

オプション 1: 各 Oracle ホームに一意のポート範囲を割り当てる

表 3-2 に示すように、各 Oracle ホームに重複しない OC4J ポート範囲を割り当てることができます。これにより、ORACLE_HOME1 の OPMN と ORACLE_HOME2 の OPMN は、OPMN ポートの割当て時に同じポート番号を使用しないようになり、同じポートがバインドされなくなります。

表 3-2 2 つの Oracle ホームで一意のポート範囲を使用する例

ORACLE_HOME1/opmn/conf/opmn.xml における OC4J ポート範囲	*ORACLE_HOME2*/opmn/conf/opmn.xml における OC4J ポート範囲
<ias-component id="OC4J"> ... <process-type id="home" ... > ... <port id="default-web-site" range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> ... </process-type> <process-type id="OC4J_WebCenter" ... > ... <port id="default-web-site" range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> </process-type>	<ias-component id="OC4J"> ... <process-type id="home" ... > ... <port id="default-web-site" range="**4601-4700**" protocol="ajp"/> <port id="rmi" range="**4701-4800**"/> <port id="rmis" range="**4901-4999**"/> <port id="jms" range="**4801-4900**"/> ... </process-type> <process-type id="OC4J_WebCenter" ... > ... <port id="default-web-site" range="**4601-4700**" protocol="ajp"/> <port id="rmi" range="**4701-4800**"/> <port id="rmis" range="**4901-4999**"/> <port id="jms" range="**4801-4900**"/> </process-type>

手順は次のとおりです。

1. AJP、RMI および JMS 用に一意のポートを選択します。
2. *ORACLE_HOME2*/opmn/conf/opmn.xml を編集します。
3. このファイルの各 OC4J インスタンスで、新しい一意のポート範囲を使用するように AJP、RMI および JMS を変更します。次に例を示します。

   ```
   <port id="ajp" range="4601-4700"/>
   <port id="rmi" range="4701-4800"/>
   <port id="jms" range="4801-4900"/>
   ```

4. ファイルを保存して閉じます。
5. OPMN をリロードします。

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl reload
   (Windows) ORACLE_HOME¥opmn¥bin¥opmnctl reload
   ```

オプション 2: OC4J インスタンスの起動の最大試行回数を増やす

OPMN では、起動の失敗が通知される前に、決められた回数だけプロセスの起動が再試行されます。ポートに範囲があるプロセス・タイプでは、プロセス起動の失敗が割当て済のポート番号にプロセスをバインドできないことに起因する場合、指定された範囲内の異なるポート番号を使用してプロセスの起動が再試行されます。これを利用して、2 つの Oracle ホームのポート範囲は同じままで、OPMN がプロセスの起動を再試行する回数を増やすことができます。結果として、OPMN により問題のないポートが選択されます。この方法では問題が完全に取り除かれることはありません。10 回試行しても問題のないポートが見つかるとは限らないためですが、問題が発生する可能性は減ります。

再試行回数を制御するパラメータは retry です。デフォルト値は 2 です。各 Oracle ホームで次の手順を実行して、このパラメータを 10 などの大きな数値に増やすことができます。

1. *ORACLE_HOME*/opmn/conf/opmn.xml を編集します。
2. このファイルの各 OC4J インスタンスで、起動および再起動の再試行値を増やします。次に例を示します。

```
<start timeout="600" retry="10"/>
<restart timeout="720" retry="10"/>
```

3. ファイルを保存して閉じます。
4. OPMN をリロードします。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl reload
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl reload
```

表 3-3 に、再試行回数を 10 に増やした後の、同じホスト上の 2 つの Oracle ホームの opmn.xml ファイルの例を示します。

表 3-3 2 つの Oracle ホームで再試行回数を増やす例

***ORACLE_HOME1*/opmn/conf/opmn.xml における OC4J ポート範囲**	***ORACLE_HOME2*/opmn/conf/opmn.xml における OC4J ポート範囲**
<ias-component id="OC4J"> ... <process-type id="home" ... > ... <start timeout="600" retry="**10**"/> ... <restart timeout="720" retry="**10**"/> <port id="default-web-site" range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> ... </process-type> <process-type id="OC4J_WebCenter" ... > ... <start timeout="600" retry="**10**"/> ... <restart timeout="720" retry="**10**"/> <port id="default-web-site" range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> </process-type>	<ias-component id="OC4J"> ... <process-type id="home" ... > ... <start timeout="600" retry="**10**"/> ... <restart timeout="720" retry="**10**"/> <port id="default-web-site" range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> ... </process-type> <process-type id=""OC4J_WebCenter" ... > ... <start timeout="600" retry="**10**"/> ... <restart timeout="720" retry="**10**"/> <port id="default-web-site" range="**12501-12600**" protocol="ajp"/> <port id="rmi" range="**12401-12500**"/> <port id="rmis" range="**12701-12800**"/> <port id="jms" range="**12601-12700**"/> </process-type>

第 II 部

基本的な管理

この部では、基本的な管理作業について説明します。

この部は、次の章で構成されています。

- 第 4 章「ポートの管理」
- 第 5 章「ログ・ファイルの管理」

4

ポートの管理

この章では、Oracle Application Server のポート番号の表示および変更方法について説明します。この章の項目は次のとおりです。

4.1 ポートの管理について

Oracle Application Server の多くのコンポーネントやサービスはポートを使用します。管理者はこれらのサービスが使用するポート番号を把握し、ホスト内で同じポート番号が 2 つのサービスに使用されないようにすることが重要です。

ポート番号のほとんどはインストール時に割り当てられます。すべてのコンポーネントとサービスにはポート範囲が割り当てられています。Oracle Application Server はこのポート範囲内にあるポート番号を使用してポートを割り当てます。Oracle Application Server はポート範囲内の最も小さい番号から次のチェックを実行します。

- ポートがホスト内の別の Oracle Application Server インストールによって使用されているか。

  チェック時にインストールが実行されている場合でも停止している場合でも、Oracle Application Server はポートが使用されているかどうかを検出できます。

- ポートが現在実行中のプロセスによって使用されているか。

  このプロセスには Oracle Application Server 以外のプロセスを含むホストのすべてのプロセスが含まれます。

これらのチェック項目に 1 つでも該当するものがある場合、Oracle Application Server は割り当てられたポート範囲内の次に大きなポート番号に進み、空いているポートが見つかるまでチェックを続けます。

このポートの割当てを無効にして、インストール時にポート番号を指定することもできます。その場合は、`staticports.ini` というテンプレート・ファイルを編集し、特別なオプションを指定して Oracle Universal Installer を起動します。

> **関連項目：** 割り当てられるポート範囲の詳細は、付録 D を参照してください。`staticports.ini` を使用してインストール時のポートの割当てを無効にする方法の詳細は、Oracle Application Server のインストレーション・ガイドを参照してください。

4.2 ポート番号の表示

次のコマンドを使用して、現行のポート番号を表示できます。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl status -l
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl status -l
```

また、Application Server Control コンソールを使用してポート番号を表示することもできます。それには、「クラスタ・トポロジ」ページから「**ランタイム・ポート**」を選択します。

4.3 中間層のポートの変更

この項では、中間層インスタンスのポート番号を変更する手順の詳細について説明します。ポート番号の変更方法とその影響を受ける他のコンポーネントの更新方法についても説明します。

注意： ポート番号は、未使用であるポートの任意のポート番号に変更できます。コンポーネントに割り当てられているポート範囲内のポート番号を使用する必要はありません。割当て済のポート範囲の詳細は、付録 D を参照してください。

この項の項目は次のとおりです。

4.3.1 OC4J リスナー・ポートの変更

Oracle WebCenter Framework インストール・オプションを選択した場合、Oracle HTTP Server はインストールされませんが、OC4J によって HTTP リスナーが提供されます。OC4J リスナーは、Application Server Control コンソールを使用するか、または手動で変更できます。

- Application Server Control コンソールを使用する場合 :

  1. Application Server Control コンソールに対して Java SSO が有効化されている場合は、それを無効化します。

     「クラスタ・トポロジ」ページから、「**Java SSO 構成**」をクリックします。次に、「Java SSO 構成」ページで、「**関連アプリケーション**」をクリックします。「関連アプリケーション」ページで、「ascontrol」の選択が解除されていることを確認します。「**適用**」をクリックします。

  2. 「クラスタ・トポロジ」ページから、「**ランタイム・ポート**」をクリックします。

  3. OC4J インスタンス（デフォルトでは home）の HTTP ポートに対応する「**ポートの構成**」アイコンをクリックします。

  4. 「サーバー・プロパティ」ページの「ポート」セクションで、「**Web サイト**」表を探します。この表で、**default-web-site** のポートを変更します。

  5. 「**適用**」をクリックします。

  6. OC4J インスタンスを再起動します。

     (UNIX) *ORACLE_HOME*/opmn/bin/opmnctl startproc process-type=*OC4J_instance*
     (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl startproc process-type=*OC4J_instance*

- 手動の場合 :

  1. 次のコマンドを発行します。たとえば、OC4J のデフォルト・インスタンスの名前が home の場合は、次のように指定します。

     (UNIX) *ORACLE_HOME*/opmn/bin/opmnctl config port update ias-component=default_group process-type=home range=8888 portid=default-web-site
     (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl config port update ias-component=default_group process-type=home range=8888 portid=default-web-site

  2. OPMN をリロードします。

     (UNIX) *ORACLE_HOME*/opmn/bin/opmnctl reload
     (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl reload

  3. 変更したポート番号を含む OC4J インスタンスを起動します。

     (UNIX) *ORACLE_HOME*/opmn/bin/opmnctl startproc process-type=*OC4J_instance*
     (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl startproc process-type=*OC4J_instance*

たとえば、UNIX では、home インスタンスにあるポート番号を変更した場合は、次のように起動します。

```
ORACLE_HOME/opmn/bin/opmnctl startproc process-type=home
```

4.3.2 その他の OC4J ポートの変更

この項では、次の OC4J ポート番号を変更する方法について説明します。

- AJP
- JMS
- RMI
- RMIS
- IIOP
- IIOPS1（サーバーのみ）
- IIOPS2（サーバーおよびクライアント）

Oracle Application Server のデフォルトの設定では、OC4J ポートのタイプごとに 1 つのポート番号が使用されるわけではありません。かわりに、OC4J ポートのタイプごとにポート範囲が指定され、そのポート範囲がホスト上のすべての OC4J インスタンスに適用されます。つまり、実行時にホスト上の各 OC4J インスタンスには、ポート範囲内の空きポートから 1 つずつ割り当てられます。たとえば、ホスト上のすべての OC4J インスタンスに対するデフォルトの AJP の範囲は 12501-12600 です。そのため、各 OC4J インスタンスにはこの AJP のポート範囲内の空きポートから 1 つずつ割り当てられます。

OC4J ポート番号を変更するときは、通常、新しいポート範囲を指定します。範囲の指定には、単純なポート範囲（12501-12600）、カンマ区切りのポート・リスト（12501, 12504, 12507）またはその両方の組合せ（12501-12580, 12583, 12590-12600）を使用できます。ポート範囲のデフォルトのポート数は、100 ポートです。指定されたポート範囲が狭い場合は、OC4J インスタンスの起動時に問題が生じる場合があります。AJP および RMI ポート範囲は必須ですが、他はオプションです。

注意： IIOP、IIOPS1 および IIOPS2 の各ポートは、デフォルトでは構成されないため、opmn.xml ファイルに一覧表示されていない場合があります。これらを構成するには、ファイルに手動で追加する必要があります。

詳細は、『Oracle Containers for J2EE サービス・ガイド』を参照してください。

OC4J のポート範囲は、Application Server Control コンソールを使用するか、または手動で変更できます。

- Application Server Control コンソールを使用する場合 :
  1. 「クラスタ・トポロジ」ページから、「**ランタイム・ポート**」をクリックします。
  2. 変更するポートに対応する「**ポートの構成**」アイコンをクリックします。
  3. 「サーバー・プロパティ」ページの「ポート」セクションで、変更するポートのポート範囲を変更します。
  4. 「**適用**」をクリックします。
  5. 「クラスタ・トポロジ」ページにナビゲートし、変更した OC4J インスタンスを選択して、「**再起動**」をクリックします。
  6. 確認ページで「**はい**」をクリックします。

- 手動の場合 :

  1. opmn.xml ファイルを開きます。

     (UNIX) *ORACLE_HOME*/opmn/conf/opmn.xml
     (Windows) *ORACLE_HOME*¥opmn¥conf¥opmn.xml

  2. 変更するポート範囲を含む OC4J インスタンスの要素を探します。たとえば、home インスタンスのポート範囲を変更する場合は、次の要素を探します。

     ```
     <process-type id="home" ...>
     ```

  3. OC4J インスタンス要素には、ポート・タイプごとの port 要素があります。たとえば、「基本インストール」オプションを使用してインスタンスをインストールした場合は、次のようなエントリがあります。

     ```
     <port id="default-web-site" range="8888" protocol="http"/>
     <port id="rmi" range="12401-12500"/>
     <port id="rmis" range="12701-12800"/>
     <port id="jms" range="12601-12700"/>
     <port id="iiop" range="13301-13400"/>
     <port id="iiops1" range="13401-13500"/>
     <port id="iiops2" range="13501-13600"/>
     ```

  4. 変更するポートの range パラメータを変更し、ファイルを保存します。

  5. OPMN をリロードします。

     (UNIX) *ORACLE_HOME*/opmn/bin/opmnctl reload
     (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl reload

  6. 変更したポート番号を含む OC4J インスタンスを起動します。

     (UNIX) *ORACLE_HOME*/opmn/bin/opmnctl startproc process-type=*OC4J_instance*
     (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl startproc process-type=*OC4J_instance*

     たとえば、UNIX では、home インスタンスにあるポート番号を変更した場合は、次のように起動します。

     ORACLE_HOME/opmn/bin/opmnctl startproc process-type=home

4.3.3 Oracle HTTP Server リスニング・ポートの変更

Oracle HTTP Server リスニング・ポート（非 SSL または SSL）を変更するには、Oracle HTTP Server Listen ディレクティブを変更します。これを行うときは、多くの場合、関連する依存関係も設定する必要があります。たとえば、OracleAS Web Cache リリース 2（10.1.2）を使用して Oracle Application Server インスタンスのパフォーマンスを向上させている場合は、Oracle HTTP Server リスニング・ポートを変更すると、OracleAS Web Cache のオリジナル・サーバーの設定も変更する必要があります。

次の各項では、Oracle HTTP Server の HTTP または HTTPS リスニング・ポートを変更する方法について説明します。

- 1024 未満に設定されたポート使用時の Oracle HTTP Server の root 実行の有効化（UNIX のみ）
- Oracle HTTP Server の非 SSL リスニング・ポートの変更
- Oracle HTTP Server の SSL リスニング・ポートの変更

4.3.3.1 1024 未満に設定されたポート使用時の Oracle HTTP Server の root 実行の有効化（UNIX のみ）

UNIX システムでリスニング・ポートを 1024 未満の番号に変更する場合は、Oracle HTTP Server のリスニング・ポート番号を変更する前に、次の手順を実行します。

デフォルトでは、Oracle HTTP Server は非 root ユーザー（Oracle Application Server をインストールしたユーザー）として実行されます。UNIX システムでは、Oracle HTTP Server リスニング・ポート番号を 1024 未満の値に変更する場合は、次のように root として実行するように Oracle HTTP Server を有効にする必要があります。

1. root としてログインします。
2. 中間層の Oracle ホームで次のコマンドを実行します。

```
cd ORACLE_HOME/Apache/Apache/bin
chown root .apachectl
chmod 6750 .apachectl
```

4.3.3.2 Oracle HTTP Server の非 SSL リスニング・ポートの変更

Oracle HTTP Server の非 SSL（HTTP）リスニング・ポートを変更するには、次の作業を実行します。UNIX システムでリスニング・ポートを 1024 未満の番号に変更する場合は、最初に第 4.3.3.1 項の手順を実行する必要があります。

- 作業 1: Oracle HTTP Server HTTP Listen ディレクティブの変更
- 作業 2: OracleAS Web Cache の更新
- 作業 3: Oracle Content DB でのポート番号の変更
- 作業 4: 中間層インスタンスの再起動

作業 1: Oracle HTTP Server HTTP Listen ディレクティブの変更

Oracle HTTP Server HTTP Listen ディレクティブを変更する手順は次のとおりです。

1. `httpd.conf` ファイルを開きます。

```
(UNIX) ORACLE_HOME/Apache/Apache/conf/httpd.conf
(Windows) ORACLE_HOME¥Apache¥Apache¥conf¥httpd.conf
```

2. Listen ディレクティブを新しいポート番号で更新します。

   このファイルに複数の Listen ディレクティブが存在する場合があります。SSL 仮想ホスト・コンテナで囲まれていない Listen ディレクティブを変更します。正しい Listen ディレクティブを最も簡単に探すには、ファイルの古いポート番号を検索します。

3. 中間層インスタンスが OracleAS Web Cache をリバース・プロキシとして使用していない場合、Port ディレクティブを更新します。

   Listen および Port の値には、同じポート番号を指定します。次の例では、リスニング・ポートを 7779 に変更しています。

```
Listen 7779
Port 7779
```

   中間層インスタンスが OracleAS Web Cache をリバース・プロキシとして使用している場合、Port ディレクティブを更新する必要はありません。

作業 2: OracleAS Web Cache の更新

中間層インスタンスがリリース 2（10.1.2）の OracleAS Web Cache をリバース・プロキシとして使用している場合は、OracleAS Web Cache を更新する必要があります。

たとえば、リリース 2（10.1.2）の OracleAS Web Cache スタンドアロン・インスタンスの場合、次の手順を実行します。

1. リリース 2（10.1.2）インスタンスの OracleAS Web Cache Manager で、「**Origin Servers, Sites, and Load Balancing**」→「**Origin Servers**」を選択します。
2. HTTP ポートを選択し、「**Edit Selected**」をクリックします。
3. 「Edit Application Web Server」ダイアログ・ボックスで、「**Port**」フィールドの数値を変更します。
4. 「**Submit**」をクリックします。
5. 「**Apply Changes**」をクリックします。
6. 「**Restart**」をクリックします。

作業 3: Oracle Content DB でのポート番号の変更

Oracle Content DB のある環境では、Oracle Content DB Application Port ドメイン・プロパティを変更して OC4J_Content インスタンスを再起動する必要があります。手順の詳細は、『Oracle Content Database Oracle WebCenter Suite 用管理者ガイド』の Oracle Content DB のポート番号の変更に関する項を参照してください。

作業 4: 中間層インスタンスの再起動

アプリケーション・サーバー・インスタンスを再起動します。

- UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl stopall
ORACLE_HOME/opmn/bin/opmnctl startall
```

- Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl stopall
ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

4.3.3.3 Oracle HTTP Server の SSL リスニング・ポートの変更

Oracle HTTP Server の SSL（HTTPS）リスニング・ポートを変更するには、次の作業を実行します。UNIX システムでリスニング・ポートを 1024 未満の番号に変更する場合は、第 4.3.3.1 項の手順を実行する必要があります。

- 作業 1: Oracle HTTP Server Listen ディレクティブの変更
- 作業 2: OracleAS Web Cache の更新
- 作業 3: mod_osso の再登録
- 作業 4: Oracle Content DB でのポート番号の変更
- 作業 5: 中間層インスタンスの再起動

作業 1: Oracle HTTP Server Listen ディレクティブの変更

HTTPS ポートを変更する場合は、Oracle HTTP Server の `ssl.conf` ファイルにある SSL の Listen ディレクティブと Port ディレクティブの両方を新しいポート番号に変更します。

1. 次の場所にある `ssl.conf` ファイルを編集します。

```
(UNIX) ORACLE_HOME/Apache/Apache/conf
(Windows) ORACLE_HOME¥Apache¥Apache¥conf
```

2. SSL の Listen および Port ディレクティブ、また VirtualHost _default ディレクティブを新しいポート番号で更新し、ファイルを保存します。

   Listen、Port および VirtualHost _default の値には、同じポート番号を指定します。次の例では、ディレクティブをポート 4445 に変更しています。

   ```
   Listen 4445
   Port 4445
   <VirtualHost _default_:4445>
   ```

   ファイルを保存して閉じます。

3. 中間層インスタンスを再起動します。

   - UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/opmn/bin/opmnctl startall
     ```

   - Windows の場合 :

     ```
     ORACLE_HOME¥opmn¥bin¥opmnctl stopall
     ORACLE_HOME¥opmn¥bin¥opmnctl startall
     ```

作業 2: OracleAS Web Cache の更新

中間層インスタンスがリリース 2（10.1.2）の OracleAS Web Cache をリバース・プロキシとして使用している場合は、OracleAS Web Cache を更新する必要があります。次の手順に従います。

1. リリース 2（10.1.2）インスタンスの OracleAS Web Cache Manager で、「**Origin Servers, Sites, and Load Balancing**」→「**Origin Servers**」を選択します。
2. HTTPS ポートを選択し、「**Edit Selected**」をクリックします。
3. 「Edit Application Web Server」ダイアログ・ボックスで、「**Port**」フィールドの数値を変更します。
4. 「**Submit**」をクリックします。
5. 「**Apply Changes**」をクリックします。
6. 「**Restart**」をクリックします。

作業 3: mod_osso の再登録

OracleAS Single Sign-On 認証を有効にした場合（つまり、mod_osso を登録した場合）、mod_osso を再登録する手順は次のとおりです。

1. Identity Management ホストで、ORACLE_HOME および ORACLE_SID 環境変数を設定します。
2. Identity Management ホストで、`ssoreg` スクリプトを `-remote_midtier` オプションを使用して実行します。このスクリプトは、次のディレクトリにあります。

   ```
   (UNIX) ORACLE_HOME/sso/bin/ssoreg.sh
   (Windows)ORACLE_HOME¥sso¥bin¥ssoreg.bat
   ```

   たとえば Linux では、次のように指定します。

   ```
   $ORACLE_HOME/sso/bin/ssoreg.sh -oracle_home_path $ORACLE_HOME
    -config_mod_osso TRUE
    -site_name myhost.com:7778
    -remote_midtier
    -config_file $ORACLE_HOME/Apache/Apache/conf/osso/myosso.conf
    -mod_osso_url http://myhost.com:7778
   ```

   作成される構成ファイル（この例では `mysso.conf`）は、不明瞭化された osso 構成ファイルです。

3. 不明瞭化された osso 構成ファイルを、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスにコピーします。
4. 中間層ホストで、次のスクリプトを実行して登録を完了させます。

```
(UNIX) ORACLE_HOME/Apache/Apache/bin/osso1013 config_file
(Windows) perl ORACLE_HOME¥Apache¥Apache¥bin¥osso1013 config_file
```

作業 4: Oracle Content DB でのポート番号の変更

Oracle Content DB のある環境では、Oracle Content DB ApplicationUseHttps ドメイン・プロパティを変更して OC4J_Content インスタンスを再起動する必要があります。手順の詳細は、『Oracle Content Database Oracle WebCenter Suite 用管理者ガイド』の Oracle Content DB のポート番号の変更に関する項を参照してください。

作業 5: 中間層インスタンスの再起動

アプリケーション・サーバー・インスタンスを再起動します。

- UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl stopall
ORACLE_HOME/opmn/bin/opmnctl startall
```

- Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl stopall
ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

4.3.4 Oracle HTTP Server 診断ポートの変更

Oracle HTTP Server 診断ポート番号を変更する手順は次のとおりです。

1. dms.conf ファイルを開きます。

```
(UNIX) ORACLE_HOME/Apache/Apache/conf/dms.conf
(Windows) ORACLE_HOME¥Apache¥Apache¥conf¥dms.conf
```

2. ファイルに出現する古いポート番号をすべて新しいポート番号に変更して、ファイルを保存します。これには、Listen ディレクティブ、OpmnHostPort ディレクティブ、Redirect ディレクティブ、および VirtualHost が含まれます。
3. Oracle HTTP Server を再起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server

(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopproc ias-component=HTTP_Server
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startproc ias-component=HTTP_Server
```

4.3.5 Java Object Cache ポートの変更

すべてのインストール・タイプで Java Object Cache ポート番号を変更する手順は次のとおりです。

1. javacache.xml ファイルを開きます。

```
(UNIX) ORACLE_HOME/javacache/admin/javacache.xml
(Windows) ORACLE_HOME¥javacache¥admin¥javacache.xml
```

2. <communication> 要素で、<coordinator> 要素の discovery-port パラメータを新しいポート番号で更新し、ファイルを保存します。

   次に例を示します。

```
<coordinator discovery-port="7010" />
```

3. Java Object Cache を使用する J2EE アプリケーションを含むすべての OC4J インスタンスを再起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=OC4J_instance_name
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl restartproc process-type=OC4J_instance_name
```

4.3.6 OPMN ポート（ONS Local、Request、Remote）の変更

この項では、次のポート番号を変更する方法について説明します。

- ONS Local ポート
- ONS Request ポート
- ONS Remote ポート

これらのポートを変更する手順は次のとおりです。

1. Application Server Control コンソール、OPMN および OPMN が管理するすべてのプロセスを停止します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
(Windows)ORACLE_HOME¥opmn¥bin¥opmnctl stopall
```

2. opmn.xml ファイルを開きます。

```
(UNIX) ORACLE_HOME/opmn/conf/opmn.xml
(Windows) ORACLE_HOME¥opmn¥conf¥opmn.xml
```

3. <notification-server> 要素で、変更するポートに応じて <port> 要素の local、remote または request パラメータを変更し、ファイルを保存します。

   次に例を示します。

```
<port local="6101" remote="6201" request="6004"/>
```

4. Application Server Control コンソール、OPMN および OPMN が管理するすべてのプロセスを起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

4.3.7 ポート・トンネリング・ポートの変更

ポート・トンネリング・ポート番号を変更する手順は次のとおりです。

1. Application Server Control コンソール、OPMN および OPMN が管理するすべてのプロセスを停止します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
(Windows)ORACLE_HOME¥opmn¥bin¥opmnctl stopall
```

2. opmn.xml ファイルを開きます。

```
(UNIX) ORACLE_HOME/opmn/conf/opmn.xml
(Windows) ORACLE_HOME¥opmn¥conf¥opmn.xml
```

3. <ias-component id="IASPT"> 要素で、<port> 要素の range パラメータを新しい範囲で更新します。たとえば、次のように指定します。

```
<port id="ajp" range="7501-7553"/>
```

   opmn.xml で指定したポート番号の範囲は iaspt.conf で指定したポート番号より優先されます。そのため、opmn.xml のポート番号を更新するだけで済みます。

4. Application Server Control コンソール、OPMN および OPMN が管理するすべてのプロセスを起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

4.4 10.1.4 または 10.1.2 の Infrastructure ポートの変更

この項の項目は次のとおりです。

4.4.1 10.1.4 または 10.1.2 の OracleAS Metadata Repository Net リスナー・ポートの変更

環境に 10.1.4 または 10.1.2 の OracleAS Metadata Repository が含まれるときに、そのリスナー・ポート番号を変更する場合は、この項で説明する手順を実行します。

最初に、OracleAS Metadata Repository のリスナー・ポート番号の変更が必要かどうかを判断します。ホスト上に同じポートを使用する別のデータベースがある場合でも、OracleAS Metadata Repository と他のデータベースで同じポートを使用できます。

同じホスト上の複数のデータベースが使用するポートに対しては、次に示すガイドラインを参考にしてください。

- 複数の Oracle9*i* および Oracle Database 10*g* データベースは、同じ Oracle Net リスナー・ポートを共有できます。Oracle9*i* および Oracle Database 10*g* データベースが配置されたホストに OracleAS Metadata Repository をインストールする場合は、そのすべてでポート 1521 を使用できます。OracleAS Metadata Repository のポート番号を変更する必要はありません。
- システム上に Net8 リスナーを実行する Oracle8*i* データベースがある場合、OracleAS Metadata Repository は別のポートを使用する必要があります。両者は同じポートを共有できません。

注意： ホスト上で同じキー値を使用する 2 つのリスナーを実行する場合は、第 4.4.1.1 項「IPC リスナーの KEY 値の変更」を参照してください。

OracleAS Metadata Repository リスナー・ポートの変更が必要な場合は、この項の手順に従います。OracleAS Metadata Repository には、様々な使用方法があります。次の表を参照して、各自の使用方法に応じた変更手順を実行してください。

Metadata Repository の使用方法	Oracle Net リスナー・ポートの変更に必要な作業
■ ID 管理リポジトリおよび製品メタデータ・リポジトリ ■ Oracle Internet Directory に登録済	作業 1: 中間層インスタンスの停止 作業 2: OracleAS Metadata Repository Oracle Net リスナー・ポートの変更 作業 3: Oracle Internet Directory の更新 作業 4: OracleAS Single Sign-On の更新 作業 5: OracleAS Certificate Authority の更新 作業 6: Application Server Control コンソールの更新 作業 7: 中間層インスタンスの更新

Metadata Repository の使用方法	Oracle Net リスナー・ポートの変更に必要な作業
■ ID 管理リポジトリのみ ■ Oracle Internet Directory に登録済	作業 1: 中間層インスタンスの停止 作業 2: OracleAS Metadata Repository Oracle Net リスナー・ポートの変更 作業 3: Oracle Internet Directory の更新 作業 4: OracleAS Single Sign-On の更新 作業 5: OracleAS Certificate Authority の更新 作業 6: Application Server Control コンソールの更新
■ 製品メタデータ・リポジトリ ■ Oracle Internet Directory に登録済	作業 1: 中間層インスタンスの停止 作業 2: OracleAS Metadata Repository Oracle Net リスナー・ポートの変更 作業 3: Oracle Internet Directory の更新 作業 7: 中間層インスタンスの更新

作業 1: 中間層インスタンスの停止

各中間層の Oracle ホームで次のコマンドを実行して、Metadata Repository を使用するすべての中間層インスタンスを停止します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopall
```

作業 2: OracleAS Metadata Repository Oracle Net リスナー・ポートの変更

OracleAS Metadata Repository ホストで、次の手順を実行します。

1. ORACLE_HOME および ORACLE_SID 環境変数が設定されていることを確認します。
2. OPMN が実行されている場合は停止します。

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
   (Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopall
   ```

3. OracleAS Metadata Repository リスナーを停止します。

   ```
   lsnrctl stop
   ```

4. 次の場所にある `listener.ora` ファイルを編集します。

   ```
   (UNIX) ORACLE_HOME/network/admin/listener.ora
   (Windows) ORACLE_HOME¥network¥admin¥listener.ora
   ```

   `LISTENER` エントリの下にある `PORT` の値を更新します。ファイルを保存します。

5. `tnsnames.ora` ファイルを編集します。デフォルトの場所は次のとおりです。

   ```
   (UNIX) ORACLE_HOME/network/admin/tnsnames.ora
   (Windows) ORACLE_HOME¥network¥admin¥tnsnames.ora
   ```

   ファイルに次の変更を加えます。

   a. OracleAS Metadata Repository に適用される各エントリの `PORT` 値を更新します。

   b. 次のエントリを追加します。

      ```
      newnetport =
        (DESCRIPTION =
          (ADDRESS = (PROTOCOL = tcp) (HOST = hostname) (PORT = port)))
      ```

      この例で、*hostname* は完全修飾のホスト名、*port* は新しいポート番号です。

6. OracleAS Metadata Repository リスナーを起動します。

```
lsnrctl start
```

7. SQL*Plus を使用して、SYSDBA 権限を持つ SYSTEM ユーザーとして OracleAS Metadata Repository にログインし、次のコマンドを実行します。

```
SQL> ALTER SYSTEM SET local_listener='newnetport' scope=spfile;
```

8. SQL*Plus を使用して、OracleAS Metadata Repository を再起動します。

```
SQL> SHUTDOWN
SQL> STARTUP
```

9. Oracle Internet Directory を起動します。

   - UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl start
ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=OID
```

   - Windows の場合 :

```
ORACLE_HOME¥opmn¥bin/opmnctl start
ORACLE_HOME¥opmn¥bin¥opmnctl startproc ias-component=OID
```

作業 3: Oracle Internet Directory の更新

Identity Management ホストで、Oracle Internet Directory を新しい Oracle Net リスナー・ポート番号によって更新します。

1. Oracle Directory Manager を起動します。

   - UNIX の場合は、次のコマンドを使用します。

```
ORACLE_HOME/bin/oidadmin
```

   - Windows の場合は、Oracle Directory Manager（「**スタート**」→「**プログラム**」→「**Oracle Application Server Infrastructure - *Oracle_Home***」→「**Integrated Management Tools**」→「**Oracle Directory Manager**」）にナビゲートします。

2. Oracle Directory Manager にログインします。
3. 「システム・オブジェクト」フレームで、次の手順を実行します。
   a. 「**エントリ管理**」を開きます。
   b. 「**cn=Oracle Context**」を開きます。
   c. OracleAS Metadata Repository の DBName を選択します。たとえば、DBName がデフォルトの `orcl` の場合は、「**cn=ORCL**」を選択します。
   d. 「プロパティ」タブで、「**orclnetdescstring**」フィールドの `PORT` パラメータを新しいポート番号で更新します。
4. 「**適用**」をクリックします。
5. 「システム・オブジェクト」フレームで、次の手順を実行します。
   a. 「**cn=Oracle Context**」の下にある、OracleAS Metadata Repository の DBName を選択します。たとえば、DBName がデフォルトの `orcl` の場合は、「**cn=ORCL**」を選択します。
   b. 「**cn=DESCRIPTION_0**」を開きます
   c. 「**cn=ADDRESS_0**」を選択します。
   d. 「プロパティ」タブで、「**orclnetaddressstring**」フィールドの `PORT` パラメータを新しいポート番号で更新します。
6. 「**適用**」をクリックします。

7. Oracle Internet Directory の Oracle ホームにある OPMN を起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

作業 4: OracleAS Single Sign-On の更新

OracleAS Single Sign-On の Oracle ホームで、次の手順を実行します。

1. UNIX システムの場合は、LD_LIBRARY_PATH、LD_LIBRARY_PATH_64、LIB_PATH または SHLIB_PATH 環境変数を、表 1-1 に示されている適切な値に設定します。実際に設定が必要な環境変数および値は、UNIX オペレーティング・システムのタイプによって異なります。
2. 次のコマンドを実行して、新しいリポジトリ・ポート番号で OracleAS Single Sign-On を更新します。
   - UNIX の場合 :

     ```
     $ORACLE_HOME/jdk/bin/java -jar $ORACLE_HOME/sso/lib/ossoca.jar reassoc -repos
     $ORACLE_HOME
     ```

   - Windows の場合 :

     ```
     %ORACLE_HOME%¥jdk¥bin¥java -jar %ORACLE_HOME%¥sso¥lib¥ossoca.jar reassoc -repos
     %ORACLE_HOME%
     ```

3. OC4J を再起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc ias-component=OC4J
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl restartproc ias-component=OC4J
```

作業 5: OracleAS Certificate Authority の更新

Identity Management インストールに OracleAS Certificate Authority が構成されている場合は、次の手順を実行します。

1. 次のコマンドを実行します。

```
(UNIX) ORACLE_HOME/oca/bin/ocactl updateconnection
(Windows) ORACLE_HOME¥oca¥bin¥ocactl updateconnection
```

2. OracleAS Certificate Authority を再起動します。

```
(UNIX) ORACLE_HOME/oca/bin/ocactl stop
(UNIX) ORACLE_HOME/oca/bin/ocactl start

(Windows) ORACLE_HOME¥oca¥bin¥ocactl stop
(Windows) ORACLE_HOME¥oca¥bin¥ocactl start
```

OracleAS Certificate Authority が構成されているかどうかわからない場合は、Application Server Control のホーム・ページの「コンポーネント」セクションに一覧表示されているかどうかを調べます。

作業 6: Application Server Control コンソールの更新

Application Server Control コンソールを新しいポート番号で更新します。

1. Identity Management の Oracle ホームで、次のファイルを編集します。

```
(UNIX) ORACLE_HOME/sysman/emd/targets.xml
(Windows) ORACLE_HOME¥sysman¥emd¥targets.xml
```

2. 古い OracleAS Metadata Repository ポート番号のそれぞれを新しいポート番号で更新し、ファイルを保存します。

   たとえば、PORT パラメータと ConnectDescriptor パラメータを更新します。

3. Application Server Control コンソールをリロードします。

```
(UNIX) ORACLE_HOME/bin/emctl reload
(Windows) ORACLE_HOME¥bin¥emctl reload
```

作業 7: 中間層インスタンスの更新

OracleAS Metadata Repository を使用する各中間層の Oracle ホームで、次の手順を実行します。

1. 次のファイルを新しい Oracle Net リスナー・ポート番号で更新します。

```
(UNIX) ORACLE_HOME/network/admin/tnsnames.ora
(Windows) ORACLE_HOME¥network¥admin¥tnsnames.ora
```

2. 次のファイルをチェックします。

```
(UNIX) ORACLE_HOME/Apache/modplsql/conf/dads.conf
(Windows) ORACLE_HOME¥Apache¥modplsql¥conf¥dads.conf
```

`PlsqlDatabaseConnectString` で始まる行を探します。

- その行が `ServiceNameFormat` または `SIDFormat` で終わる場合は、その行を新しい OracleAS Metadata Repository のポート番号で更新し、ファイルを保存して Oracle HTTP Server を再起動します。
- その行が `NetServiceNameFormat` で終わる場合は、何もする必要ありません。

3. 中間層インスタンスを起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

4.4.1.1 IPC リスナーの KEY 値の変更

2 つのリスナーの IPC プロトコル・アドレスに同じ KEY 値の使用が構成されている場合は、それらを同時に実行できません。デフォルトでは、OracleAS Metadata Repository リスナーの IPC KEY 値は EXTPROC に設定されます。したがって、EXTPROC キーを使用する IPC リスナーが他にもある場合は、EXTPROC1 などの別のキー値を使用するように OracleAS Metadata Repository リスナーを構成する必要があります。

IPC リスナーの KEY 値を変更する手順は次のとおりです。

1. リスナーを停止します（ORACLE_HOME 環境変数が設定されていることを確認します）。

```
lsnrctl stop
```

2. `listener.ora` および `tnsnames.ora` ファイルを編集します。各ファイルで、次の行を探します。

```
(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC))
```

この行を次のように変更します。

```
(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1))
```

3. リスナーを再起動します。

```
lsnrctl start
```

4.4.2 10.1.4 または 10.1.2 の Oracle Internet Directory ポートの変更

10*g* リリース 3（10.1.3.2.0）の中間層インスタンスは、リリース 10.1.4 または リリース 2（10.1.2）の Identity Management サービスに関連付けることができます。

Identity Management のインストールで Oracle Internet Directory の HTTP または HTTPS ポートを変更する場合は、Identity Management インストールを使用する中間層インスタンスをすべて更新する必要があります。

次の作業では、Oracle Internet Directory ポート番号を更新する方法について説明します。Infrastructure のその他のコンポーネントの更新や、そのポートを使用する中間層インスタンスの更新についても説明します。

- 作業 1: 中間層インスタンスの準備
- 作業 2: Infrastructure インスタンスの準備
- 作業 3: Oracle Internet Directory ポートの変更
- 作業 4: OracleAS Certificate Authority の再構成
- 作業 5: Identity Management インスタンスの再起動
- 作業 6: 新しいポート番号を使用するための中間層インスタンスの更新

作業 1: 中間層インスタンスの準備

Identity Management を使用する 10*g* リリース 3（10.1.3.2.0）の各中間層インスタンスが起動していることを確認します。

起動していない場合は、次のコマンドを使用してすべてのプロセスを起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

作業 2: Infrastructure インスタンスの準備

次の手順を実行して、Infrastructure インスタンスを準備します。

1. ポート番号を変更する Infrastructure で、Identity Management および関連する OracleAS Metadata Repository が起動していることを確認します。
2. 製品メタデータに異なる OracleAS Metadata Repository を使用する中間層インスタンスがある場合は、それらのリポジトリが起動していることを確認します。つまり、環境内のすべての Metadata Repository が起動していることを確認します。

作業 3: Oracle Internet Directory ポートの変更

次の手順を実行して、Oracle Internet Directory ポートを変更します。

1. Oracle Internet Directory ホストで、次の手順を実行します。
   a. `mod.ldif` というファイルを作成し、次の内容を記述します。ファイルは任意のディレクトリに作成できます。

      HTTP の場合：

      ```
      dn: cn=configset0, cn=osdldapd, cn=subconfigsubentry
      changetype: modify
      replace: orclnonsslport
      orclnonsslport: new_nonssl_port_number
      ```

      HTTPS の場合：

      ```
      dn: cn=configset0, cn=osdldapd, cn=subconfigsubentry
      changetype: modify
      replace: orclsslport
      orclsslport: new_ssl_port_number
      ```

b. 次のコマンドを実行します。

HTTP（非 SSL）ポートの場合 :

```
ldapmodify -D "cn=orcladmin" -w password -p oid_port -f mod.ldif
```

HTTPS（SSL）ポートの場合 :

```
ldapmodify -D "cn=orcladmin" -w password -p oid_port -U SSLAuth -f mod.ldif
```

oid_port は、Oracle Internet Directory の古いポート番号です。HTTPS ポートを変更する場合は、`-U` 引数を追加して SSL 認証モードを指定する必要があります。*SSLAuth* の値には、認証が不要な場合は `1`、一方向の認証が必要な場合は `2`、双方向の認証が必要な場合は `3` を使用します。

2. Oracle Internet Directory のホストで、Application Server Control コンソールだけでなく、Oracle Internet Directory の入っているインスタンス全体を停止します。

- UNIX の場合 :

```
ORACLE_HOME/bin/emctl stop iasconsole
ORACLE_HOME/opmn/bin/opmnctl stopall
```

- Windows の場合 :

```
ORACLE_HOME¥bin¥emctl stop iasconsole
ORACLE_HOME¥opmn¥bin¥opmnctl stopall
```

3. Oracle Internet Directory の Oracle ホームで次の手順を実行します。この Oracle Internet Directory に登録したその他の Oracle ホームに OracleAS Metadata Repository がインストールされている場合は、その Oracle ホームでもこの手順を実行してください。

a. `ldap.ora` ファイルを開きます。

```
(UNIX) ORACLE_HOME/ldap/admin/ldap.ora
(Windows) ORACLE_HOME¥ldap¥admin¥ldap.ora
```

b. 新しいポート番号を含むように次の行を更新し、ファイルを保存します。

```
DIRECTORY_SERVERS=(myhost.myco.com:non_ssl_port:ssl_port)
```

c. `ias.properties` ファイルを開きます。

```
(UNIX) ORACLE_HOME/config/ias.properties
(Windows) ORACLE_HOME¥config¥ias.properties
```

d. `OIDport`（HTTP ポートの変更の場合）または `OIDsslport`（HTTPS ポートの変更の場合）の値を新しいポート番号に変更し、ファイルを保存します。

4. Oracle Internet Directory のホストで、Oracle Internet Directory の入っているインスタンスを起動し、Application Server Control コンソールを起動します。

- UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/bin/emctl start iasconsole
```

- Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl startall
ORACLE_HOME¥bin¥emctl start iasconsole
```

5. OracleAS Single Sign-On の Oracle ホームで次の手順を実行します。

a. UNIX システムの場合は、LD_LIBRARY_PATH、LD_LIBRARY_PATH_64、LIB_PATH または SHLIB_PATH 環境変数を、表 1-1 に示されている適切な値に設定します。実際に設定が必要な環境変数および値は、UNIX オペレーティング・システムのタイプによって異なります。

b. OracleAS Single Sign-On の Oracle ホームで次のコマンドを実行します。

```
$ORACLE_HOME/jdk/bin/java -jar $ORACLE_HOME/sso/lib/ossoca.jar reassoc -repos
$ORACLE_HOME
```

作業 4: OracleAS Certificate Authority の再構成

OracleAS Certificate Authority を使用している場合は、この作業を実行します。

1. OracleAS Certificate Authority が別の Oracle ホームで実行されている場合は、OracleAS Certificate Authority の Oracle ホームで次の手順を実行します。

   a. `ias.properties` ファイルを開きます。

   (UNIX) *ORACLE_HOME*/config/ias.properties
   (Windows) *ORACLE_HOME*¥config¥ias.properties

   b. `OIDport`（HTTP ポートの変更の場合）または `OIDsslport`（HTTPS ポートの変更の場合）の値を新しいポート番号に変更し、ファイルを保存します。

2. OracleAS Certificate Authority の Oracle ホームで次のコマンドを実行して、OracleAS Certificate Authority を新しい Oracle Internet Directory ポート番号で更新します。

   (UNIX) *ORACLE_HOME*/oca/bin/ocactl changesecurity -server_auth_port *portnum*
   (Windows) *ORACLE_HOME*¥oca¥bin¥ocactl changesecurity -server_auth_port *portnum*

   この例では、*portnum* は、OracleAS Certificate Authority Server Authentication Virtual Host（SSL）ポートで、デフォルトは 6600 です。

   関連項目： 詳細は、『Oracle Application Server Certificate Authority 管理者ガイド』を参照してください。

作業 5: Identity Management インスタンスの再起動

Identity Management インスタンスを再起動します。

- UNIX の場合 :

  ORACLE_HOME/bin/emctl stop iasconsole
  ORACLE_HOME/opmn/bin/opmnctl stopall
  ORACLE_HOME/opmn/bin/opmnctl startall
  ORACLE_HOME/bin/emctl start iasconsole

- Windows の場合 :

  ORACLE_HOME¥bin¥emctl stop iasconsole
  ORACLE_HOME¥opmn¥bin¥opmnctl stopall
  ORACLE_HOME¥opmn¥bin¥opmnctl startall
  ORACLE_HOME¥bin¥emctl start iasconsole

作業 6: 新しいポート番号を使用するための中間層インスタンスの更新

Identity Management インストールを使用する各中間層インスタンスで、ID 管理の変更ウィザードを実行し、インスタンスを起動します。

1. Application Server Control コンソールを使用して、中間層インスタンスの OC4J ホーム・ページにナビゲートします。
2. 「**管理**」をクリックします。
3. 表の「タスク名」列で「**セキュリティ**」が閉じている場合は、それを開きます。「ID 管理」行で、「**タスクに移動**」アイコンをクリックします。
4. 「ID 管理」ページで、「**変更**」をクリックします。
5. ウィザードの手順に従って、新しい Identity Management の情報を指定します。詳細は、第 6.6 項を参照してください。

6. 操作が完了したら、「**再起動**」をクリックして、OC4J インスタンスを再起動します。その後、「確認」ページで「**はい**」をクリックします。

また、Identity Management を使用するリリース 2（10.1.2）の中間層インスタンスは、すべて更新する必要があります。リリース 2（10.1.2）の中間層インスタンスを更新する方法の詳細は、『Oracle Application Server 管理者ガイド』の「ポートの管理」を参照してください。

4.4.3 10.1.4 または 10.1.2 の Identity Management インストールの HTTP Server ポートの変更

この項では、10.1.4 または 10.1.2 の Identity Management インストールで Oracle HTTP Server の HTTP または HTTPS リスニング・ポートを変更する方法について説明します。このポート番号を変更するときは、OracleAS Single Sign-On ポート番号も関連付けて変更します。これは、OracleAS Single Sign-On ポートを使用するすべての中間層インスタンスを更新する必要があることを意味します。

次の作業では、Identity Management で Oracle HTTP Server ポート番号を更新する方法について説明します。Infrastructure のその他のコンポーネントの更新や、そのポートを使用する中間層インスタンスの更新についても説明します。

- 作業 1: 中間層インスタンスの準備
- 作業 2: Infrastructure インスタンスの準備
- 作業 3: Oracle HTTP Server の Listen および Port ディレクティブの変更
- 作業 4: 1024 未満のポート使用時の Oracle HTTP Server の root 実行の有効化（UNIX のみ）
- 作業 5: Application Server Control コンソールの更新
- 作業 6: OracleAS Single Sign-On の更新
- 作業 7: mod_osso の再登録
- 作業 8: Oracle Delegated Administration Services の更新
- 作業 9: OracleAS Certificate Authority の更新
- 作業 10: Identity Management インスタンスの再起動
- 作業 11: OracleAS Certificate Authority の再起動
- 作業 12: 新しいポート番号を使用するための中間層インスタンスの更新

作業 1: 中間層インスタンスの準備

Identity Management を使用する 10*g* リリース 3（10.1.3.2.0）の各中間層インスタンスが起動していることを確認します。

起動していない場合は、次のコマンドを使用してすべてのプロセスを起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

作業 2: Infrastructure インスタンスの準備

次の手順を実行して、Infrastructure を準備します。

1. ポート番号を変更する Infrastructure で、Identity Management および関連する OracleAS Metadata Repository が起動していることを確認します。
2. 製品メタデータに異なる Metadata Repository を使用する中間層インスタンスがある場合は、それらのリポジトリが起動していることを確認します。つまり、環境内のすべての Metadata Repository が起動していることを確認します。

作業 3: Oracle HTTP Server の Listen および Port ディレクティブの変更

HTTP ポートを変更する場合は、Oracle HTTP Server の `httpd.conf` ファイルで、Listen ディレクティブと Port ディレクティブの両方を新しいポート番号に変更します。この作業は、次に示すように Application Server Control コンソールで実行できます。または、リリース 2（10.1.2）の Identity Management インスタンスで、手動で実行することもできます。

- Identity Management インスタンスの Application Server Control コンソールを使用する場合:
  1. Application Server ホーム・ページにナビゲートし、「**ポート**」をクリックします。
  2. 「ポート」ページで Oracle HTTP Server リスニング・ポートを検索し、「**構成**」列のアイコンをクリックします。
  3. 「サーバー・プロパティ」ページで次の操作を実行します。
     - 「**デフォルト・ポート**」フィールドに新しいポート番号を入力します。これは Port ディレクティブの場合です。
     - 「**リスニング・ポート**」列に新しいポート番号を入力します。これは Listen ディレクティブの場合です。複数のリスニング・ポートが一覧表示される場合もあります。どのポートが非 SSL リスニング・ポートかを示すには、古い非 SSL リスニング・ポートの値が指定されたポートを選択します。
  4. ページ下部の「**適用**」をクリックします。
  5. 再起動しない場合は、「確認」ページで「**いいえ**」をクリックします。
- 手動の場合:
  1. `httpd.conf` ファイルを開きます。

     (UNIX) *ORACLE_HOME*/Apache/Apache/conf/httpd.conf
     (Windows) *ORACLE_HOME*¥Apache¥Apache¥conf¥httpd.conf

  2. 非 SSL の Listen および Port ディレクティブを新しいポート番号で更新し、ファイルを保存します。

     Listen および Port の値には、同じポート番号を指定します。次の例では、ディレクティブを 7779 に変更しています。

     ```
     Listen 7779
     Port 7779
     ```

     このファイルに複数の Listen および Port ディレクティブが存在する場合があります。SSL 仮想ホスト・コンテナで囲まれていない Listen および Port ディレクティブを変更します。正しい Listen および Port ディレクティブを最も簡単に探すには、ファイルの古いポート番号を検索します。

  3. 次のコマンドを実行します。

     (UNIX) *ORACLE_HOME*/dcm/bin/dcmctl updateConfig -ct ohs
     (Windows) *ORACLE_HOME*¥dcm¥bin¥dcmctl updateConfig -ct ohs

HTTPS ポートを変更する場合は、Oracle HTTP Server の `ssl.conf` ファイルにある SSL の Listen ディレクティブと Port ディレクティブの両方を新しいポート番号に変更します。この手順は、次のように手動で実行する必要があります。

1. 次の場所にある `ssl.conf` ファイルを編集します。

   (UNIX) *ORACLE_HOME*/Apache/Apache/conf/ssl.conf
   (Windows) *ORACLE_HOME*¥Apache¥Apache¥conf¥ssl.conf

2. SSL の Listen および Port ディレクティブを新しいポート番号で更新し、ファイルを保存します。

   Listen および Port の値には、同じポート番号を指定します。次の例では、ディレクティブを 4445 に変更しています。

   ```
   Listen 4445
   Port 4445
   ```

   ファイルを保存して閉じます。

3. 次のコマンドを実行します。

   ```
   (UNIX) ORACLE_HOME/dcm/bin/dcmctl updateConfig -ct ohs
   (Windows) ORACLE_HOME¥dcm¥bin¥dcmctl updateConfig -ct ohs
   ```

作業 4: 1024 未満のポート使用時の Oracle HTTP Server の root 実行の有効化（UNIX のみ）

UNIX でポート番号を 1024 未満の番号に変更する場合は、この作業を実行します。

デフォルトでは、Oracle HTTP Server は非 root ユーザー（Oracle Application Server をインストールしたユーザー）として実行されます。UNIX システムでは、Oracle Application Server の非 SSL リスニング・ポート番号を 1024 未満の値に変更する場合は、次のように root として実行するように Oracle HTTP Server を有効にする必要があります。

1. root としてログインします。
2. Infrastructure の Oracle ホームで次のコマンドを実行します。

   ```
   cd ORACLE_HOME/Apache/Apache/bin
   chown root .apachectl
   chmod 6750 .apachectl
   ```

作業 5: Application Server Control コンソールの更新

Application Server Control コンソールを新しいポート番号で更新します。

1. `targets.xml` ファイルを開きます。

   ```
   (UNIX) ORACLE_HOME/sysman/emd/targets.xml
   (Windows) ORACLE_HOME¥sysman¥emd¥targets.xml
   ```

2. 古い Oracle HTTP Server リスニング・ポート番号のそれぞれを新しいポート番号で更新し、ファイルを保存します。

   構成によっては、このファイルに Oracle HTTP Server リスニング・ポートが存在しない場合や、多数のポート番号が存在する場合があります。リスニング・ポートは、ポート自身のパラメータである場合や URL の一部である場合があります。このファイルを最も簡単に編集するには、古い Oracle HTTP Server リスニング・ポート番号すべてを検索し、それらの番号を新しいポート番号で置換します。

3. Application Server Control コンソールをリロードします。

   ```
   (UNIX) ORACLE_HOME/bin/emctl reload
   (Windows) ORACLE_HOME¥bin¥emctl reload
   ```

作業 6: OracleAS Single Sign-On の更新

ポートを変更するインストールで OracleAS Single Sign-On が Oracle HTTP Server の HTTP リスニング・ポートを使用するように構成されている場合は、この作業を実行します。

1. UNIX システムの場合は、LD_LIBRARY_PATH、LD_LIBRARY_PATH_64、LIB_PATH または SHLIB_PATH 環境変数を、表 1-1 に示されている適切な値に設定します。実際に設定が必要な環境変数および値は、UNIX オペレーティング・システムのタイプによって異なります。

2. OracleAS Single Sign-On の Oracle ホームで、次のコマンドのいずれかまたは両方を実行します。

   非 SSL ポートを変更する場合は、次のコマンドを実行します。

   ```
   (UNIX) ORACLE_HOME/sso/bin/ssocfg.sh http hostname new_non_ssl_port_number
   (Windows) ORACLE_HOME¥sso¥bin¥ssocfg.bat http hostname new_non_ssl_port_number
   ```

   SSL ポートを変更する場合は、次のコマンドを実行します。

   ```
   (UNIX) ORACLE_HOME/sso/bin/ssocfg.sh https hostname new_ssl_port_number
   (Windows) ORACLE_HOME¥sso¥bin¥ssocfg.bat https hostname new_ssl_port_number
   ```

   この例では、次のようになります。

   - *hostname* は、OracleAS Single Sign-On が稼動するホストです。
   - *new_non_ssl_port_number* は、Oracle HTTP Server の新しい非 SSL リスニング・ポート番号です。
   - *new_ssl_port_number* は、Oracle HTTP Server の新しい SSL リスニング・ポート番号です。

作業 7: mod_osso の再登録

mod_osso を再登録する手順は次のとおりです。

1. 環境変数を設定します。
   - UNIX システムの場合は、LD_LIBRARY_PATH、LD_LIBRARY_PATH_64、LIB_PATH または SHLIB_PATH 環境変数を、表 1-1 に示されている適切な値に設定します。実際に設定が必要な環境変数および値は、UNIX オペレーティング・システムのタイプによって異なります。
   - Windows システムの場合は、PATH=%PATH%;%ORACLE_HOME%¥bin;%ORACLE_HOME%¥lib のように設定します。
2. Oracle HTTP Server のリスニング・ポートを変更する場合、Identity Management の Oracle ホームで次のコマンドを実行して、デフォルトのパートナ・アプリケーションを処理するように mod_osso を再登録します。

   UNIX の場合 :

   ```
   ORACLE_HOME/sso/bin/ssoreg.sh
   -oracle_home_path identity_management_oracle_home
   -site_name identity_management_hostname:new_port_number
   -config_mod_osso TRUE
   -mod_osso_url mod_osso_url
   ```

   Windows の場合 :

   ```
   ORACLE_HOME¥sso¥bin¥ssoreg.bat
   -oracle_home_path identity_management_oracle_home
   -site_name identity_management_hostname:new_port_number
   -config_mod_osso TRUE
   -mod_osso_url mod_osso_url
   ```

   たとえば、UNIX では、ホスト myhost で Oracle HTTP Server のリスニング・ポートを 7779 に変更する場合は、次のように実行します。

   ```
   $ORACLE_HOME/sso/bin/ssoreg.sh
   -oracle_home_path /disk1/oracleas
   -site_name myhost:7779
   -config_mod_osso TRUE
   -mod_osso_url http://myhost.mydomain:7779
   ```

3. Oracle HTTP Server SSL リスニング・ポートを変更する場合は、次の手順を実行します。

   a. 中間層の Oracle ホームで次のコマンドを実行して、mod_osso を新しいポート番号で再登録します。

      UNIX の場合 :

      ```
      ORACLE_HOME/sso/bin/ssoreg.sh
      -oracle_home_path identity_management_oracle_home
      -site_name identity_management_hostname:new_port_number
      -config_mod_osso TRUE
      -update_mode MODIFY
      -remote_midtier
      -config_file path/osso-https.conf
      -mod_osso_url mod_osso_url
      ```

      Windows の場合 :

      ```
      ORACLE_HOME¥sso¥bin¥ssoreg.bat
      -oracle_home_path identity_management_oracle_home
      -site_name identity_management_hostname:new_port_number
      -config_mod_osso TRUE
      -update_mode MODIFY
      -remote_midtier
      -config_file path¥osso-https.conf
      -mod_osso_url mod_osso_url
      ```

      たとえば、UNIX では、myhost で Oracle HTTP Server の SSL リスニング・ポートを 4445 に変更する場合は、次のように実行します。

      ```
      $ORACLE_HOME/sso/bin/ssoreg.sh
      -oracle_home_path /disk1/oracleas
      -site_name myhost:4445
      -config_mod_osso TRUE
      -update_mode MODIFY
      -remote_midtier
      -config_file $ORACLE_HOME/Apache/Apache/conf/osso/osso-https.conf
      -mod_osso_url http://myhost.mydomain:7778
      ```

      関連項目： mod_osso の登録の詳細は、『Oracle Application Server Single Sign-On 管理者ガイド』を参照してください。

   b. 不明瞭化された osso 構成ファイルを、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスにコピーします。

   c. 中間層ホストで、次のスクリプトを実行して登録を完了させます。

      ```
      (UNIX) ORACLE_HOME/Apache/Apache/bin/osso1013 config_file
      (Windows) perl ORACLE_HOME¥Apache¥Apache¥bin¥osso1013 config_file
      ```

   d. 次の場所にある mod_osso.conf ファイルを編集します。

      ```
      (UNIX) ORACLE_HOME/Apache/Apache/conf/mod_osso.conf
      (Windows) ORACLE_HOME¥Apache¥Apache¥conf¥mod_osso.conf
      ```

      mod_osso.conf ファイルの次のディレクティブがコメントアウトされていない場合は、それをコメントアウトします。

      UNIX の場合 :

      ```
      LoadModule osso_module libexec/mod_osso.so
      ```

      Windows の場合 :

      ```
      LoadModule osso_module modules¥ApacheModuleOsso.dll
      ```

e. 同じディレクトリ（conf）にある httpd.conf ファイルに、前の手順でコメントアウトしたディレクティブを追加していない場合は、それを追加します。デフォルトの設定では、このディレクティブを次の行の直後に追加します。

```
LoadModule wchandshake_module libexec/mod_wchandshake.so
```

4. Oracle HTTP Server を再起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc process-type=HTTP_Server
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl restartproc process-type=HTTP_Server
```

5. その他のパートナ・アプリケーションを構成または変更した場合は、そのアプリケーションも再登録する必要があります。

関連項目： mod_osso の登録の詳細は、『Oracle Application Server Single Sign-On 管理者ガイド』を参照してください。

作業 8: Oracle Delegated Administration Services の更新

Oracle Delegated Administration Services を構成して新しいポート番号を使用する場合は、次の手順を実行して、Oracle Internet Directory の Oracle Delegated Administration Services の URL エントリを更新します。

次のコマンドを実行すると、Oracle Delegated Administration Services が使用するポートを検出できます。

```
ldapsearch -h oid_host -p oid_port -D "cn=orcladmin"
-w "password" -b "cn=OperationURLs,cn=DAS,cn=Products,cn=OracleContext"
-s base "objectclass=*" orcldasurlbase
```

Oracle Delegated Administration Services を更新する手順は次のとおりです。

1. 次の内容を記述した mod.ldif というファイルを作成します（任意のディレクトリに作成できます）。

```
dn:cn=OperationURLs,cn=DAS,cn=Products,cn=OracleContext
changetype:modify
replace:orcldasurlbase
orcldasurlbase:http://hostname:new_http_port_number/
```

orcldasurlbase URL の最後にスラッシュを入力してください。

2. 次のコマンドを実行します。

```
ldapmodify -D cn=orcladmin -w password -p oid_port -f mod.ldif
```

作業 9: OracleAS Certificate Authority の更新

OracleAS Certificate Authority を使用している場合は、次の手順を実行します。

1. OracleAS Certificate Authority の Oracle ホームで次のコマンドを実行し、OracleAS Certificate Authority を OracleAS Single Sign-On サーバーに再登録します。

```
(UNIX) ORACLE_HOME/oca/bin/ocactl changesecurity -server_auth_port portnum
(Windows) ORACLE_HOME¥oca¥bin¥ocactl changesecurity -server_auth_port portnum
```

この例では、*portnum* は、OracleAS Certificate Authority Server Authentication Virtual Host（SSL）ポートで、デフォルトは 6600 です。

関連項目：『Oracle Application Server Certificate Authority 管理者ガイド』

2. OracleAS Certificate Authority が OracleAS Single Sign-On サーバー以外の Oracle ホームに配置されている場合は、OracleAS Certificate Authority の Oracle ホームで Oracle HTTP Server と oca インスタンスを再起動します。

   - UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
     ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=oca
     ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server
     ORACLE_HOME/opmn/bin/opmnctl startproc process-type=oca
     ```

   - Windows の場合 :

     ```
     ORACLE_HOME¥opmn¥bin¥opmnctl stopproc ias-component=HTTP_Server
     ORACLE_HOME¥opmn¥bin¥opmnctl stopproc process-type=oca
     ORACLE_HOME¥opmn¥bin¥opmnctl startproc ias-component=HTTP_Server
     ORACLE_HOME¥opmn¥bin¥opmnctl startproc process-type=oca
     ```

作業 10: Identity Management インスタンスの再起動

Identity Management インスタンスを再起動します。

- UNIX の場合 :

  ```
  ORACLE_HOME/bin/emctl stop iasconsole
  ORACLE_HOME/opmn/bin/opmnctl stopall
  ORACLE_HOME/opmn/bin/opmnctl startall
  ORACLE_HOME/bin/emctl start iasconsole
  ```

- Windows の場合 :

  ```
  ORACLE_HOME¥bin¥emctl stop iasconsole
  ORACLE_HOME¥opmn¥bin¥opmnctl stopall
  ORACLE_HOME¥opmn¥bin¥opmnctl startall
  ORACLE_HOME¥bin¥emctl start iasconsole
  ```

作業 11: OracleAS Certificate Authority の再起動

このインスタンスで OracleAS Certificate Authority を構成している場合は、OracleAS Certificate Authority を再起動します。

```
(UNIX) ORACLE_HOME/oca/bin/ocactl start
(Windows) ORACLE_HOME¥oca¥bin¥ocactl start
```

作業 12: 新しいポート番号を使用するための中間層インスタンスの更新

Identity Management インストールで Oracle HTTP Server ポートを変更したので、すべての中間層インスタンスを更新して、新しいポート番号を使用する必要があります。この手順では、ID 管理の変更ウィザードによってポート番号が内部的に取得されるため、HTTP または HTTPS ポート番号を明示的に指定する必要はありません。

Identity Management を使用する 10*g* リリース 3（10.1.3.2.0）の各中間層インスタンスで、次の作業を行います。

1. Application Server Control コンソールを使用して、中間層インスタンスの OC4J ホーム・ページにナビゲートします。
2. 「**管理**」をクリックします。
3. 表の「タスク名」列で「**セキュリティ**」が閉じている場合は、それを開きます。「ID 管理」行で、「**タスクに移動**」アイコンをクリックします。
4. 「ID 管理」ページで、「**変更**」をクリックします。
5. Oracle Internet Directory の現在の情報を入力します。詳細は、第 6.6 項を参照してください。
6. 「**OK**」をクリックします。

7. 操作が完了したら、「**再起動**」をクリックして、OC4J インスタンスを再起動します。その後、「確認」ページで「**はい**」をクリックします。

また、Identity Management を使用するリリース 2（10.1.2）の中間層インスタンスは、すべて更新する必要があります。リリース 2（10.1.2）の中間層インスタンスを更新する方法の詳細は、『Oracle Application Server 管理者ガイド』の「ポートの管理」を参照してください。

4.4.4 10.1.4 または 10.1.2 の OracleAS Certificate Authority ポートの変更

この項では、次のポート番号を変更する方法について説明します。

- OracleAS Certificate Authority Server Authentication Virtual Host（SSL）
- OracleAS Certificate Authority Mutual Authentication Virtual Host（SSL）

これらのポート番号を変更する手順は次のとおりです。

1. OracleAS Certificate Authority が配置された Infrastructure の Oracle ホームにある `ocm_apache.conf` ファイルを開きます。

   (UNIX) *ORACLE_HOME*/Apache/Apache/conf/ocm_apache.conf
   (Windows) *ORACLE_HOME*¥Apache¥Apache¥conf¥ocm_apache.conf

   a. `Server` または `Mutual` ポート、あるいはその両方を変更し、ファイルを保存します。

      このファイルでは、これらのポート番号が次の 2 箇所に指定されていることに注意してください。

      – Listen ディレクティブとして
      – デフォルトの仮想ホストとして

      これらの箇所を探す最も簡単な方法は、元のポート番号でファイルを検索することです。

   b. 次のコマンドを実行します。

      (UNIX) *ORACLE_HOME*/dcm/bin/dcmctl updateConfig -ct ohs
      (Windows) *ORACLE_HOME*¥dcm¥bin¥dcmctl updateConfig -ct ohs

2. 次のコマンドを実行します（ORACLE_HOME 環境変数が設定されていることを確認します）。

   sqlplus oca/*oca_admin_password* @$ORACLE_HOME/oca/sql/ocaportchg

   a. プロンプトが表示されたら、サーバー認証のみのポートを入力します。このポート番号を変更しない場合は、元のポート番号を入力します。

   b. プロンプトが表示されたら、相互認証ポートを入力します。このポート番号を変更しない場合は、元のポート番号を入力します。

3. OracleAS Certificate Authority の Oracle ホームで次のコマンドを実行し、OracleAS Certificate Authority を OracleAS Single Sign-On サーバーに再登録します。

   (UNIX) ORACLE_HOME/oca/bin/ocactl changesecurity -server_auth_port *portnum*
   (Windows) *ORACLE_HOME*¥oca¥bin¥ocactl changesecurity -server_auth_port *portnum*

   この例では、*portnum* は、OracleAS Certificate Authority Server Authentication Virtual Host（SSL）ポートで、デフォルトは 6600 です。

   関連項目：『Oracle Application Server Certificate Authority 管理者ガイド』

4. Oracle HTTP Server を再起動します。

   (UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc type=ohs
   (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl restartproc type=ohs

5. OracleAS Certificate Authority OC4J インスタンスを再起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl restartproc type=oc4j instancename=oca
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl restartproc type=oc4j instancename=oca
```

6. Oracle Application Server Certificate Authority を再起動します。

```
(UNIX) ORACLE_HOME/oca/bin/ocactl start
(Windows) ORACLE_HOME¥oca¥bin¥ocactl start
```

5

ログ・ファイルの管理

Oracle Application Server コンポーネントは、起動および停止情報、エラー、警告メッセージ、HTTP リクエスト時のアクセス情報など、すべての種類のイベントを記録するメッセージが格納されたログ・ファイルを生成します。この章では、システムの動作の監視やシステムに関する問題の診断に役立つ、ログ・ファイルの表示方法と管理方法について説明します。

この章の項目は次のとおりです。

5.1 Application Server Control でのログ・ファイルのリストと表示

Application Server Control コンソールでは、Oracle Application Server コンポーネント全体のログ・ファイルのリストを作成して検索できます。ログ・ファイルは、Application Server Control コンソール・ページで表示したり、ローカル・クライアントにダウンロードしたり、別のツールを使用して表示することができます。

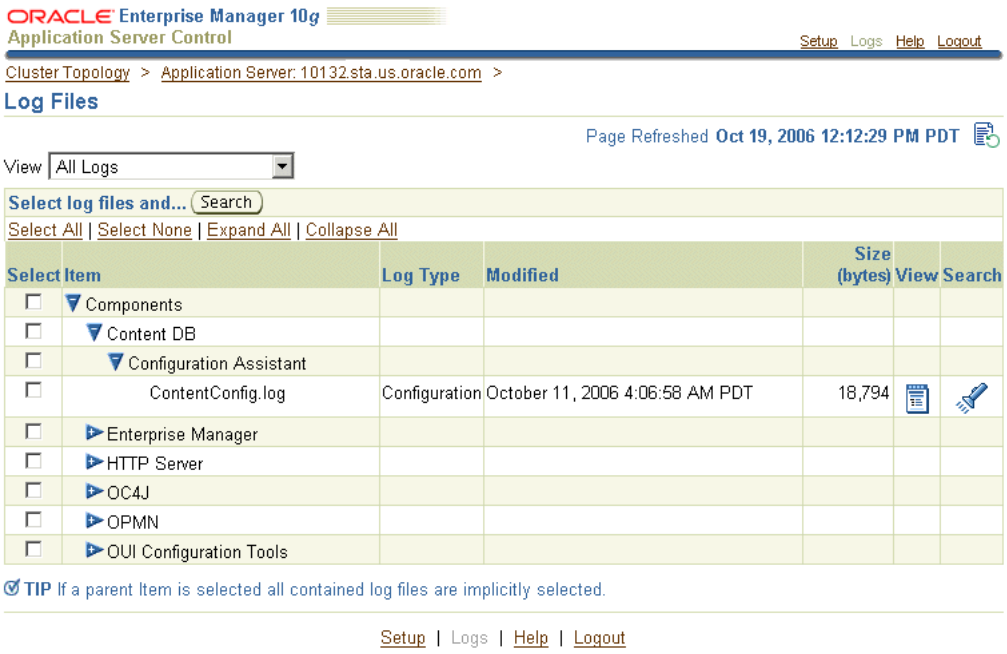
この項の項目は次のとおりです。

5.1.1 ログ・ファイルの表示

Application Server Control コンソールを使用すると、特定のコンポーネントのログ・ファイルを一覧表示できます。

1. Application Server ホーム・ページにナビゲートし、「**ログ**」リンクを選択します。「ログ・ファイル」ページが表示されます。
2. 「**表示**」メニューでオプションを選択すると、アプリケーション・ログ、診断ログ、管理システム・ログ、Web サービス・ログなど、あらゆる種類のログを表示できます。
3. 表にコンポーネントのログ・ファイルのリストが表示されるまで、アイテムを開きます。たとえば、「**コンポーネント**」→「**Content DB**」→「**コンフィギュレーション・アシスタント**」を開きます。図 5-1 に示すように、表にログ・ファイルが一覧表示されます。

図 5-1 Enterprise Manager の「ログ・ファイル」ページ



4. 特定のログ・ファイルを表示するには、「**表示**」をクリックします。ログ・ファイルのテキストが表示されます。

5.1.2 コンポーネントのログ・ファイルの一覧表示

Application Server Control コンソールを使用すると、個々のコンポーネント、コンポーネントの一部、またはすべてのコンポーネントのログ・ファイルを一覧表示できます。ログ・ファイルを一覧表示する手順は次のとおりです。

1. Application Server ホーム・ページにナビゲートし、「**ログ**」リンクを選択します。「ログ・ファイル」ページが表示されます。
2. 「**表示**」メニューで選択すると、アプリケーション・ログ、診断ログ、管理システム・ログ、Web サービス・ログなど、あらゆる種類のログを表示できます。
3. すべてのコンポーネントを表示するには、表の「**コンポーネント**」を選択します。一部のコンポーネントを表示するには、「コンポーネント」アイテムを開いてから、目的の階層レベルが表示されるまで「コンポーネント」の下にあるアイテムを開きます。その後、対象コンポーネントを選択します。
4. 「**検索**」をクリックして、選択したコンポーネントのログ・ファイルのリストを作成します。
5. 検索の終了後に表示される「ログの検索」ページで、「**選択したログ・ファイルを表示**」をクリックします。「ログの検索」ページに、ログ・ファイルの名前が表示されます。

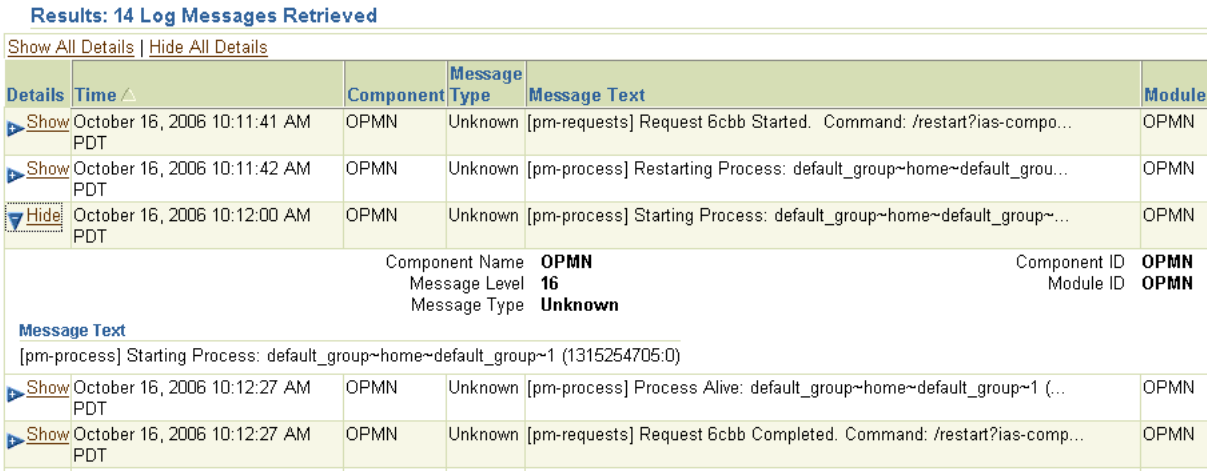
5.1.3 ログ・ファイルの検索とメッセージの表示

Application Server Control コンソールの「ログの検索」ページでは、特定のログ・ファイル属性のフィルタを設定してログ・ファイルを検索できます。

次の手順に従います。

1. Application Server ホーム・ページにナビゲートし、「**ログ**」リンクを選択します。「ログ・ファイル」ページが表示されます。
2. 「**表示**」メニューで、ログ・ファイルのタイプを選択します。
3. 特定のコンポーネントのログ・ファイルを表示するには、表で選択します。
4. 「**検索**」をクリックします。「ログの検索」ページが表示されます。
5. 「**日付範囲**」や「**メッセージ・タイプ**」などの検索基準を指定します。
6. 「**拡張検索オプション**」をクリックして、「ログ・メッセージ・フィールドでフィルタ」表を表示します。
7. 「**ログ・メッセージ・フィールド**」リストでフィールドを選択します。
8. 「**行を追加**」をクリックして、選択したログ・メッセージ・フィールドのための行を追加します。
9. 「**値**」フィールドに、目的の検索値を入力します。
10. 正規表現であることを指定するには、「**正規表現**」をクリックします（正規表現の詳細は、第 5.1.4 項を参照）。
11. さらにフィールドを追加して値を指定する場合は、「**行の追加**」をクリックして、追加する値を入力します。
12. 「**検索**」をクリックして、検索を実行します。検索が終了すると、図 5-2 に示すように、一致するフィールドを含むログ・ファイルが「**結果**」セクションに表示されます。

図 5-2 ログ検索の「結果」セクション



13. ログ・エントリを表示するには、「ログの検索」ページの「結果」領域にある「**詳細**」列の「**表示**」をクリックします。

    エラー・メッセージの詳細が表示されます。この情報には、「コンポーネント名」、「コンポーネント ID」、「メッセージ・レベル」、「モジュール ID」、「メッセージ・タイプ」、「メッセージ・テキスト」およびオプションで「実行コンテキスト ID」（ECID）などが含まれます。

5.1.4 検索での正規表現の使用

正規表現による一致は、「ログの検索」ページで「正規表現」フィールドのチェック・ボックスが選択されている場合に有効になります。正規表現は、「**メッセージ・テキスト**」フィールドと「**ログ・メッセージ・フィールドでフィルタ**」表で指定できます。検索で正規表現を使用すると、パターン表現を入力し、検索で一致する文字列を検索できます。

検索では、Apache Jakarta 正規表現エンジンが使用されます。これは、任意の文字列に "*"、任意の 1 文字に "?" を使用します。また、エントリの最初の文字のみの一致に "^"、エントリの最後の文字のみの一致に "$" を指定する境界一致や、タブに "¥t"、改行に "¥n"、復帰に "¥r"、改ページに "¥f" を指定する特殊文字がサポートされています。

> **関連項目：** サポートされている正規表現の詳細は、http://jakarta.apache.org/regexp を参照してください。

5.2 Oracle Application Server ロギングの概要

この項では、メッセージ形式とログ・ファイルのネーミングに関する情報を提供し、コンポーネントのロギング・オプションを構成する方法について説明します。この項の項目は次のとおりです。

- ログ・ファイルの形式とネーミングについて
- コンポーネント・ロギング・オプションの構成

5.2.1 ログ・ファイルの形式とネーミングについて

Oracle Application Server コンポーネントのログ・ファイルは、テキスト・ベースの形式または Oracle Diagnostic Logging（ODL）を使用します。

ODL を使用すると、ログ・ファイルのネーミングおよび内容の形式は Oracle 標準に準拠し、診断メッセージは XML で記述されます。一部の Oracle Application Server コンポーネントは ODL を使用せず、診断メッセージはコンポーネント固有のテキスト形式で記述されます。その他のコンポーネントは ODL をサポートしますが、デフォルトでは ODL が無効です。

ログ・ファイルに格納されるメッセージ形式（ODL またはテキスト・ベース）に関係なく、ログ・ファイルは Application Server Control コンソールで表示したり、ローカル・クライアントにダウンロードしたり、別のツール（テキスト・エディタやその他のファイル表示ユーティリティ）で表示することができます。

この項の項目は次のとおりです。

- ODL メッセージの形式と ODL ログ・ファイルのネーミング
- コンポーネント別のログ・ファイル・メッセージの形式

注意： 一部の Oracle Application Server コンポーネントは、ODL をサポートしません。その他のコンポーネントは ODL をサポートしますが、デフォルトでは ODL が無効です。

5.2.1.1 ODL メッセージの形式と ODL ログ・ファイルのネーミング

Oracle Application Server コンポーネントを実行して生成された ODL メッセージは、診断ログ・ファイルに XML 形式で書き込まれます。各 ODL メッセージには、メッセージに関する情報のフィールドが含まれる HEADER 要素があり、オプションで、コンポーネント間でメッセージを関連付ける際に役立つ情報が含まれる CORRELATION_DATA 要素およびオプションの引数や関連する値などのメッセージ・テキストが含まれる PAYLOAD 要素があります。

ODL を使用すると、Oracle Application Server コンポーネントは、ロギング・ディレクトリに診断ログ・ファイルを書き込み、コンポーネント固有のネーミング規則に従ってロギング・ディレクトリの名前を決定します。

関連項目：

- 第 5.4.1 項「ODL メッセージと ODL ログ・ファイルについて」
- 第 5.3.1 項「ログ・ファイルおよびコンポーネント間のメッセージの関連付け」

5.2.1.2 コンポーネント別のログ・ファイル・メッセージの形式

表 5-1 に、各 Oracle Application Server コンポーネントがサポートするメッセージ形式を示します。一部のコンポーネントは、ODL 形式をオプションでサポートします。ODL はデフォルトの形式ではありません。

表 5-1 コンポーネント別の診断メッセージ形式

コンポーネント	デフォルトの形式	ODL のサポート	場所[1]
Application Server Control コンソール	テキスト	×	*ORACLE_HOME*/j2ee/home/log/ascontrol.log *ORACLE_HOME*/j2ee/home/log/home_default_group-1/
Content DB: ノード・ログ	テキスト	×	*ORACLE_HOME*/content/log/*domain_namenode_name.log*
Content DB: アプリケーション・ログ	テキスト	○	*ORACLE_HOME*/j2ee/OC4J_Content/application-deployments/Content/OC4J_Content_default_group_1/application.log
HTTP Server	テキスト	○	*ORACLE_HOME*/Apache/Apache/logs/error_log.*time*
OC4J *instance_name*	テキスト	○	*ORACLE_HOME*/j2ee/*instance_name*/log/*instance_group_process*/ *ORACLE_HOME*/j2ee/*instance_name*/application-deployments/*application_name*/application.log

表 5-1 コンポーネント別の診断メッセージ形式（続き）

コンポーネント	デフォルトの形式	ODL のサポート	場所[1]
OC4J *instance_name*	ODL	○	*ORACLE_HOME*/j2ee/*instance_name*/log/*instance_group_process*/oc4j/log.xml
OPMN	テキスト	×	*ORACLE_HOME*/opmn/logs *ORACLE_HOME*/opmn/logs/*component_type*~...
Port Tunneling	テキスト	×	*ORACLE_HOME*/iaspt/logs
Universal Installer	テキスト	×	*ORACLE_HOME*/cfgtoollogs
WebCenter: アプリケーション・ログ	テキスト	○	*ORACLE_HOME*/j2ee/OC4J_WebCenter/application-deployments/*type*/OC4J_WebCenter_default_group_1/application.log

[1] 場所は UNIX 形式で表示されています。Windows 形式では、スラッシュを「¥」と読み替えてください。

5.2.2 コンポーネント・ロギング・オプションの構成

管理者は、ロギング・オプションを構成して、Oracle Application Server コンポーネントによって生成および保存されるログ情報を管理および制限できます。

たとえば、Java ロギングを使用して、OC4J コンポーネントのロギング・オプションを構成する場合は、*ORACLE_HOME*/j2ee/home/config ディレクトリに格納されている `j2ee-logging.xml` ファイルを変更します。詳細は、『Oracle Containers for J2EE 構成および管理ガイド』の「OC4J でのログイン」を参照してください。

コンポーネントのロギング構成オプションには次のものがあります。

- ログ・ファイル名とパス名の指定 : ほとんどの Oracle Application Server コンポーネントでは、診断ログ・ファイルの保存先ディレクトリを指定できます。管理者は、診断ロギング・ディレクトリを指定して、システムやネットワークのリソースを管理できます。
- ログ・ファイル・サイズの制限 : Oracle Application Server コンポーネントを実行して診断メッセージを生成すると、ログ・ファイルのサイズが大きくなります。Oracle Application Server コンポーネントには、ログ・ファイル・サイズを管理するいくつかの方法があります。一部のコンポーネントでは、ログ・ファイルのサイズが増加し続けるため、管理者はログ・ファイルを監視してクリーンアップする必要があります。OC4J などのコンポーネントでは、収集および保存するログ・ファイル・データのサイズを制限する構成オプションを指定できます。
- ログ・ファイルのアーカイブの使用 : 特定の Oracle Application Server コンポーネントでは、診断ロギング・ディレクトリのサイズを制御する構成オプションを指定できます。これにより、コンポーネントのログ・ファイルを格納するディレクトリの最大サイズを指定できます。最大サイズに達すると、古いログ情報が削除されてから、新しいログ情報が保存されます。
- コンポーネントのロギング・レベルの設定 : Oracle HTTP Server などの特定の Oracle Application Server コンポーネントでは、管理者はロギング・レベルを構成できます。ロギング・レベルを構成すると、診断ログ・ファイルに保存されるメッセージ数を減らすことができます。たとえば、システムが重要なメッセージのみをレポートおよび保存するようにロギング・レベルを設定できます。

関連項目 :

- 第 A.4 項「Application Server Control のロギングの構成」
- ロギング構成オプションの設定については、Oracle Application Server コンポーネントのドキュメントを参照してください。

5.3 問題の診断とメッセージの関連付け

通常、管理者やその他のユーザーは、ログ・ファイルのデータを表示して、コンポーネント・エラーおよびエラーを引き起こす可能性のある問題を診断、監視および検索します。Application Server Control コンソールは、統合されたアーキテクチャをサポートしており、これらの作業に役立つ、異なるコンポーネントで使用できるツールを提供します。

この項の項目は次のとおりです。

5.3.1 ログ・ファイルおよびコンポーネント間のメッセージの関連付け

特定の Oracle Application Server コンポーネントは、診断メッセージの**メッセージ相関**情報を提供します。メッセージ相関情報は、診断メッセージを表示したユーザーがコンポーネント間のメッセージの関係を判断する際に役立ちます。実行コンテキスト ID（ECID）は、実行のスレッドに関連した全体的に一意な識別子です。ECID は、ログ・ファイル・エントリを使用して 1 つのアプリケーションからのメッセージまたはアプリケーション・サーバー・コンポーネント間のメッセージを関連付ける際に役立ちます。メッセージ相関情報を使用して関連メッセージを検索すると、複数のメッセージを調べることができ、最初に問題が発生したコンポーネントを識別できます（この方法は、**最初に障害が発生したコンポーネントの切分け**と呼ばれます）。メッセージ相関データを使用すると、コンポーネント全体における診断メッセージのパスが明確になり、エラーや関連する動作を把握できます。

Application Server Control コンソールの「ログ・メッセージの詳細」ページでエントリを表示する際、ECID フィールドが使用可能である場合は、実行コンテキスト ID がリンクとして表示されます。「**実行コンテキスト ID**」リンクを選択すると、同じ ECID を持つログ・リポジトリ内のすべての診断メッセージが表示されます。

ECID を使用して、Oracle Application Server 間で移動するリクエストをトラッキングできます。

ECID の形式は、次のとおりです。

request_id, sequence_number

- *request_id* は、各リクエストに関連付けられている一意な文字列です。
- *sequence_number* は、リクエストが Oracle Application Server（またはコンポーネント）間で移動する際の、リクエストのホップ番号です。

  たとえば、Oracle HTTP Server はリクエストに最初の順序番号として 0 を割り当てます。その後、リクエストが Oracle Application Server コンポーネント間で移動するたびに、順序番号が増えていきます。

表 5-2 に、メッセージ相関情報（ECID を使用）を提供する Oracle Application Server コンポーネントを示します。コンポーネントがメッセージ相関をサポートする場合も、デフォルトではこのオプションが無効です。

表 5-2 メッセージ相関をサポートする Oracle Application Server コンポーネント

コンポーネント	メッセージ相関のサポート
OC4J	メッセージ相関をサポートします。
HTTP Server	メッセージ相関をサポートします。

5.3.2 コンポーネントの問題の診断

Oracle Application Server コンポーネントに問題が発生した場合は、診断メッセージを確認することによって、問題の原因を分離して特定できます。この作業を行う際に役立つ一般的な方法は、次のとおりです。

- 問題に関連するエラーまたは警告を検索します。
- コンポーネント間でエラーを関連付けます。
- 一定の時間間隔でエラーを関連付けます。
- コンポーネント・ベースの分析を行います。

5.4 ロギングに関する高度なトピック

この項の項目は次のとおりです。

- ODL メッセージと ODL ログ・ファイルについて
- コンポーネントの診断ログ・ファイルの登録
- ODL メッセージを生成するためのコンポーネントの構成
- OC4J でリダイレクトされた stderr および stdout ファイルの管理
- ログ・ファイルの構成に関する問題

5.4.1 ODL メッセージと ODL ログ・ファイルについて

この項の項目は次のとおりです。

- ODL メッセージの内容
- ODL ログ・ファイルのローテーションとネーミング

5.4.1.1 ODL メッセージの内容

ODL を使用すると、診断メッセージが XML 形式でログ・ファイルに書き込まれます。各メッセージには、メッセージに関する情報のフィールドが含まれる HEADER 要素があり、オプションで、コンポーネント間でメッセージを関連付ける際に役立つ情報が含まれる CORRELATION_DATA 要素およびオプションの引数や関連する値などのメッセージ・テキストが含まれる PAYLOAD 要素があります。

例 5-1 に、オプションの CORRELATION_DATA 要素を含む、ODL 形式のメッセージの例を示します。

例 5-1 ODL メッセージの内容の例

```
<MESSAGE>
  <HEADER>
    <TSTZ_ORIGINATING>2006-10-19T12:52:16.821-07:00</TSTZ_ORIGINATING>
    <COMPONENT_ID>j2ee</COMPONENT_ID>
    <MSG_TYPE TYPE="ERROR"></MSG_TYPE>
    <MSG_LEVEL>1</MSG_LEVEL>
    <HOST_ID>sta.oracle.com</HOST_ID>
    <HOST_NWADDR>146.87.8.203</HOST_NWADDR>
    <MODULE_ID>security</MODULE_ID>
    <THREAD_ID>10</THREAD_ID>
    <USER_ID>oracle</USER_ID>
  </HEADER>
  <CORRELATION_DATA>
    <EXEC_CONTEXT_ID><UNIQUE_ID>146.87.8.203:41990:1161287536821:0</UNIQUE_
ID><SEQ>0</SEQ></EXEC_CONTEXT_ID>
  </CORRELATION_DATA>
  <PAYLOAD>
```

```
    <MSG_TEXT>          [RealmLoginModule] authentication failed</MSG_TEXT>
  </PAYLOAD>
</MESSAGE>
```

表 5-3 に、ODL メッセージ・ヘッダーの内容を示します。ODL 形式のメッセージを生成する任意のコンポーネントでは、オプションのヘッダー・フィールドは、生成された診断メッセージには表示されません。

表 5-3 ODL 形式のメッセージのヘッダー・フィールド

ヘッダー・フィールド名	説明	必須
COMPONENT_ID	メッセージの作成元となるコンポーネントの製品 ID またはコンポーネント ID。	必須
HOST_ID	DNS ホストのネットワーク ID。	省略可能
HOST_NWADDR	作成元ホストの IP アドレスやその他のネットワーク・アドレス。	省略可能
HOSTING_CLIENT_ID	メッセージが関連付けられるクライアントまたはセキュリティ・グループの ID。	省略可能
MODULE_ID	メッセージの作成元となるモジュールの ID。	省略可能
MSG_GROUP	メッセージが属するグループ名。類似したメッセージの選択に使用します。	省略可能
MSG_ID	メッセージ ID。メッセージを一意に識別します。	省略可能
MSG_LEVEL	メッセージ・タイプ（MSG_TYPE）を修飾する整数値。指定する値のレベルが低いほど、重大度の高いエラーを示します。指定可能な値は 1 ～ 32 です。	省略可能
MSG_TYPE	メッセージのタイプ。INTERNAL_ERROR、ERROR、WARNING、NOTIFICATION、TRACE、UNKNOWN のいずれかのタイプを指定します。MSG_TYPE を指定する際、MSG_TYPE がメッセージ・ヘッダーに含まれる場合は、TYPE 属性が必要です。	必須
ORG_ID	作成元のコンポーネントの組織 ID。通常、これは組織のドメイン名です。	省略可能
PROCESS_ID	メッセージに関連付けられたプロセスまたは実行単位のプロセス ID。Java コンポーネントでは、このフィールドを使用してプロセス ID とスレッド ID またはスレッド ID のみが指定されます。	省略可能
TSTZ_NORMALIZED	ホスト間のクロックのずれの調整用に標準化されたタイムスタンプ。診断メッセージが異なるホストのリポジトリにコピーされる場合に、このフィールドが使用されます。	省略可能
TSTZ_ORIGINATING	ローカル・タイムゾーンのタイムスタンプ。これは、メッセージが生成された日付および時刻を指定します。	必須
USER_ID	メッセージに関連付けられたユーザー ID。	省略可能

5.4.1.2 ODL ログ・ファイルのローテーションとネーミング

ODL を使用する利点は、次のとおりです。

- 保存される診断情報の合計サイズを制限します。
- 古いセグメント・ファイルが削除され、新しいセグメント・ファイルが時系列で保存されます。
- 診断ログ・ファイルのクリーンアップ時に、コンポーネントをアクティブのままにすることができ、停止する必要がありません。

ODL を使用すると、Oracle Application Server コンポーネントは、ロギング・ディレクトリに診断ログ・ファイルを書き込みます。コンポーネントは、コンポーネント固有のネーミング規則に従ってロギング・ディレクトリの名前を決定します。

ODL ログは、現在の ODL ログ・ファイル（通常は log.xml という名前）および古いメッセージが格納された **ODL アーカイブ（セグメント・ファイル）**を含む、ログ・ファイルのセットです（ODL アーカイブは、存在しない場合もあります）。ログ・ファイルが更新される場合は、新しい情報がログ・ファイル log.xml の最後に追加されます。ログ・ファイルがローテーション・ポイントに達すると名前が変更され、新しいログ・ファイル log.xml が作成されます（ローテーション・ポイントは ODL セグメント・サイズの最大値を指定することで設定されますが、一部の OC4J ログでは、コンポーネント固有の構成オプションを使用して、ローテーション時間およびローテーション頻度を指定できます）。

注意： 一部の Oracle Application Server コンポーネント（特に Oracle HTTP Server）は、この項で説明する ODL ログ・ファイルのネーミング・メカニズムをサポートしません。Oracle HTTP Server では、サイズ制限を構成できない log.xml ファイルに、ODL 診断メッセージが書き込まれます。

ODL ログ・ファイル log.xml がローテーション・ポイントに達すると、セグメント・ファイルが作成されます。つまり、コンポーネントが新しい診断メッセージを生成すると、log.xml ファイル名が log*n*.xml（*n* は整数）に変更され、新しい log.xml ファイルが作成されます。

サイズベースのログ・ローテーション

ODL ログのサイズを制限するために、コンポーネントは、ロギング・ディレクトリの最大サイズを指定する構成オプションを使用します。ディレクトリ内の全ファイルの合計サイズが最大値に達すると、合計サイズが指定された制限を超えないように一番古いアーカイブが削除されます。

注意： 最新のセグメント・ファイルは削除されません。

たとえば、log9872 というセグメント・ファイルを使用する場合、最大ディレクトリ・サイズに達すると、ログ・ファイル・ディレクトリに次のようなファイルが作成されていきます。

```
File                    Size

log.xml                 10002
log9872.xml             15000
log9873.xml             15000
log9874.xml             15000
log9875.xml             15000
log9876.xml             15000
```

この例では、log.xml が最大サイズに達すると、log9872.xml が削除され、log.xml は log9877.xml に名前が変更されます。新しい診断メッセージは、新しい log.xml に書き込まれます。

たとえば、petstore という名前の OC4J アプリケーションに対して ODL セグメント・サイズの最大値とディレクトリ・サイズの最大値を指定するには、ファイル *ORACLE_HOME*/j2ee/*instance_name*/application-deployments/petstore/orion-application.xml に次のエントリを追加します。

```
<log>
<odl path="../log/petstore/" max-file-size="1000" max-directory-size="10000" />
</log>
```

OC4J コンポーネントが j2ee-logging.xml ファイルに構成されている場合は、最大セグメント・サイズおよび最大ディレクトリ・サイズに加えて、ローテーション時間およびローテーション頻度を指定できます。

時間ベースのログ・ローテーション

<log_handler> 要素に次のプロパティを指定します。

- baseRotationTime: （オプション）。ローテーションのベース時間。ベース時間は、次のいずれかの書式で指定できます。
  - `hh:mm`: たとえば、04:20。この書式では、ローカル・タイムゾーンが使用されます。
  - `yyyy-MM-dd`: たとえば、2006-08-01。この書式では、ローカル・タイムゾーンが使用されます。
  - `yyyy-MM-ddThh:mm`: たとえば、2006-08-01T04:20。この書式では、ローカル・タイムゾーンが使用されます。
  - `yyyy-MM-ddThh:mm:ss.sTZD`: TZD はタイムゾーン・インジケータです。TZD には、UTC を示す `Z` または `{+|-}hh:mm` を指定できます。たとえば、2006-03-01T04:20:00-08:00 は、US/ 太平洋標準時の 2006 年 3 月 1 日午前 4 時 20 分 00 秒を表します。

  baseRotationTime を指定しない場合、デフォルト値は Jan. 1, 1970, 00:00 UTC です。
- rotationFrequency: ローテーションの頻度（分単位）。また、値として hourly、daily または weekly を指定できます。

これらのプロパティを次のファイルに指定します。

ORACLE_HOME/j2ee/*instance_name*/config/j2ee-logging.xml

たとえば、ローカルタイムの午前 4 時に毎日、またはサイズが 2000000 バイトに達したときにログ・ファイルがローテーションされるように指定するには、次のエントリを使用します。

```
<log_handler name="h1" class="oracle.core.ojdl.logging.ODLHandlerFactory">
     <property name="path" value="log"/>
     <property name="baseRotationTime" value="04:00"/>
     <property name="rotationFrequency" value="daily"/>
     <property name="maxFileSize" value=" 2000000"/>
</log_handler>
```

5.4.2 コンポーネントの診断ログ・ファイルの登録

Application Server Control コンソールは、Oracle Application Server コンポーネントの診断登録ファイルを読み取り、診断ログ・ファイルの名前、場所およびその他の構成情報を決定します。次のディレクトリに、診断ログ・ファイルの登録ファイルが保存されます。

ORACLE_HOME/j2ee/*instance*/applications/ascontrol/ascontrol/WEB-INF/config/registration

Oracle Application Server コンポーネントでは、構成登録ディレクトリ内に複数の登録ファイルが存在する場合があります。

登録ファイルの形式には、Oracle Application Server コンポーネント ID が含まれ、拡張子として `.xml` が設定されます。表 5-4 に、Oracle Application Server コンポーネントおよびそれらに関連付けられる ID のリストを示します。

> **注意：** コンポーネントの診断登録ファイルは、各コンポーネントによって作成されます。通常、Oracle Application Server 管理者はこれらのファイルを変更できません。

表 5-4 診断ログ・ファイル構成のためのコンポーネント ID

コンポーネント名	コンポーネント ID
Enterprise Manager	EM
Oracle HTTP Server	OHS
OC4J	OC4J

表 5-4 診断ログ・ファイル構成のためのコンポーネント ID（続き）

コンポーネント名	コンポーネント ID
OPMN	OPMN
Port Tunneling	IASPT
Oracle Content DB	CONTENT
Oracle Universal Installer	OUI
Web Services	WEBSERVICES

5.4.3 ODL メッセージを生成するためのコンポーネントの構成

この項の項目は次のとおりです。

- ODL メッセージを生成するための Oracle HTTP Server の構成
- ODL メッセージを生成するための OC4J の構成

表 5-5 に、ODL メッセージをサポートするがデフォルトではテキスト・メッセージが生成される、Oracle Application Server コンポーネントのリストを示します。構成を変更することにより、ODL メッセージを生成するように、これらのコンポーネントを構成できます（ODL メッセージを生成する Oracle Application Server コンポーネントの一覧は、表 5-1 を参照）。

表 5-5 ODL をサポートする Oracle Application Server コンポーネントと構成オプション

コンポーネント	デフォルトの形式	ODL のサポート	場所[1]
HTTP Server	テキスト	○	*ORACLE_HOME*/Apache/Apache/logs
OC4J インスタンス	テキスト	○	OC4J にデプロイされたアプリケーション： *ORACLE_HOME*/j2ee/application-deployments/*application_name*/application.log **注意：**次の OC4J ログ・ファイルでは、デフォルトで ODL が使用されます。 Java ロギングを使用する OC4J コンポーネント（OPMN が管理）： *ORACLE_HOME*/j2ee/*instance_name*/log/*instance_group_process*/oc4j/log.xml Java ロギングを使用する OC4J コンポーネント（スタンドアロン OC4J）： *ORACLE_HOME*/j2ee/*instance_name*/log/oc4j/log.xml ログ・ファイルの完全なリストは、『Oracle Containers for J2EE 構成および管理ガイド』の「OC4J でのログイン」を参照してください。

[1] 場所は UNIX 形式で表示されています。Windows 形式では、スラッシュを「¥」と読み替えてください。

5.4.3.1 ODL メッセージを生成するための Oracle HTTP Server の構成

ODL メッセージを生成するように Oracle HTTP Server を構成する手順は次のとおりです。

1. Oracle HTTP Server の ODL メッセージの格納先となる、`oracle` という名前のディレクトリを追加します。次のディレクトリのサブディレクトリとして作成します。

   (UNIX) *ORACLE_HOME*/Apache/Apache/logs
   (Windows) *ORACLE_HOME*¥Apache¥Apache¥logs

2. `httpd.conf` を変更して、OraLogMode および OraLogSeverity ディレクティブの値を設定します。このファイルは次のディレクトリにあります。

   (UNIX) *ORACLE_HOME*/Apache/Apache/conf/httpd.conf
   (Windows) *ORACLE_HOME*¥Apache¥Apache¥conf¥httpd.conf

次に例を示します。

```
OraLogMode oracle
OraLogSeverity NOTIFICATION
```

3. 中間層インスタンスを再起動します。

関連項目： OraLogMode ディレクティブおよび OraLogSeverity ディレクティブの詳細は、『Oracle HTTP Server 管理者ガイド』を参照してください。

5.4.3.2 ODL メッセージを生成するための OC4J の構成

ODL ロギングを有効にするには、OC4J ログ・ファイルの構成ファイルの <log> 要素内に <odl> という新しい要素を追加します。たとえば、petstore という名前のアプリケーションで ODL ロギングを有効にする場合は、ファイル *ORACLE_HOME*/j2ee/*instance_name*/application-deployments/petstore/orion-application.xml に次のエントリを追加します。

```
<log>
<odl path="../log/petstore/" max-file-size="1000" max-directory-size="10000" />
</log>
```

関連項目： 構成ファイルのリストなどの詳細は、『Oracle Containers for J2EE 構成および管理ガイド』の「OC4J でのログイン」を参照してください。

5.4.4 OC4J でリダイレクトされた stderr および stdout ファイルの管理

OC4J ログ・ファイルにランタイム・オプションを設定して、ファイルが特定のサイズに達したときや一日の特定の時刻に、リダイレクトされた `stderr` および `stdout` ログ・ファイルを循環するように指定できます。また、アーカイブとして保持するログ・ファイルの最大数を指定することもできます。次の例では、`stdout` ログ・ファイルが毎日午前 1 時 30 分に循環されること、またログ・ファイルの最大数が 10 であることが指定されています。

```
java -Dstdstream.rotatetime=1:30 -Dstdstream.filenumber=10 -jar oc4j.jar -out
d:¥logs¥oc4j.out
```

関連項目： 詳細は、『Oracle Containers for J2EE 構成および管理ガイド』を参照してください。

5.4.5 ログ・ファイルの構成に関する問題

Application Server Control コンソールの「ログ」リンクでは、様々な Oracle Application Server コンポーネントのログ・ファイルが統合されて表示されます。ただし、特定のログ・ファイルは、コンポーネント・レベルでのみ使用できます。Oracle Application Server コンポーネントは次のディレクトリを使用して、Application Server Control コンソールでログ・ファイルを表示できるようにします。

```
ORACLE_HOME/j2ee/home/applications/ascontrol/ascontrol/WEB-INF/config/registration
```

一部の Oracle Application Server コンポーネントのログ・ファイルは、Application Server Control コンソール・ページに表示されません。

第 III 部

高度な管理

この部では、Oracle Application Server の再構成に関する高度な管理作業について説明します。

この部は、次の章で構成されています。

6

Application Server インスタンスの再構成

Oracle Application Server をインストールしたときに、インストール・タイプとインスタンスをクラスタの一部にするかどうかを指定しました。インストールの後、Oracle Application Server インスタンスのクラスタ化（インストール時に指定しなかった場合）や、OC4J インスタンスの追加または削除が必要になる場合があります。また、OracleAS Web Cache をリバース・プロキシとして使用したり、中間層のインストールで Identity Management を使用する必要性も生じることがあります。この章では、このようなインストール・タイプの変更方法について説明します。

この章の項目は次のとおりです。

Oracle Internet Directory で匿名バインドが無効になっている場合、構成を変更する前に有効にする必要があります。詳細は、第 6.7 項「匿名バインドの有効化と無効化」を参照してください。

6.1 OC4J インスタンスの追加と削除

OC4J インスタンスは、次の各項で説明するように、既存の Oracle ホームに追加したり Oracle ホームから削除したりできます。

- OC4J インスタンスの追加
- OC4J インスタンスの削除

6.1.1 OC4J インスタンスの追加

OC4J インスタンスは、次の方法によって既存の Oracle ホームに追加できます。

- `createinstance` ユーティリティの使用。このユーティリティは Oracle ホームの `bin` ディレクトリにあります。
- Application Server Control コンソールの使用

たとえば、Oracle WebCenter Framework のインストールに、新しい OC4J インスタンスを追加できます。図 6-1 では、`crm` という名前の第 2 の OC4J インスタンスがインストールに追加されています。

コマンドライン・ユーティリティを使用して OC4J インスタンスを追加するには、次の手順を実行します。

1. インスタンスを作成します。

   (UNIX) *ORACLE_HOME*/bin/createinstance -instanceName *OC4J_instanceName* -groupName *groupname* [-httpPort *port*] [-protocol *protocol*]
   (Windows) *ORACLE_HOME*¥bin¥createinstance -instanceName *OC4J_instanceName* -groupName *groupname*[-httpPort *port*] [-protocol *protocol*]

   この例では、次のようになります。

   - `-groupName` オプションを指定しない場合、新しいインスタンスは default_group グループに割り当てられます。
   - Oracle WebCenter Framework のみのインストール・タイプ（別の HTTP サーバーからのリクエストを許可するようインストール時に構成されていない）に OC4J インスタンスを追加する場合は、`-httpPort` オプションまたは `-protocol` オプションのどちらかを指定する必要があります。
   - `-httpPort` オプションは、インスタンスを HTTP モードで実行する場合に使用します。これは Oracle HTTP Server からアクセスされなくなることを意味します。この場合、OC4J インスタンスは、OC4J HTTP リスナーを使用するように構成されます。
   - `-protocol` オプションは、Oracle HTTP Server からアクセスされる必要がある場合に使用し、その値に `ajp` を指定します。この場合、このインスタンスでは Apache JServ Protocol（AJP）が使用されます。この OC4J インスタンスは、Oracle HTTP Server からリクエストを受信して応答します。

   作成処理の途中で、パスワードを入力するよう求められます。このパスワードは、このインスタンスの `oc4jadmin` ユーザーに関連付けられます。一貫性を保つために、`oc4jadmin` ユーザーを使用してデフォルトの OC4J インスタンスにアクセスするときと同じパスワードを入力することもできます。

2. 新しい OC4J インスタンスを起動します。

   (UNIX) *ORACLE_HOME*/opmn/bin/opmnctl startproc process-type=*oc4J_instanceName*
   (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl startproc process-type=*oc4J_instanceName*

Application Server Control コンソールを使用して OC4J インスタンスを追加するには、次の手順を実行します。

1. 「アプリケーション・サーバー *instance_name*」ページにナビゲートします。
2. 「**OC4J インスタンスの作成**」をクリックします。

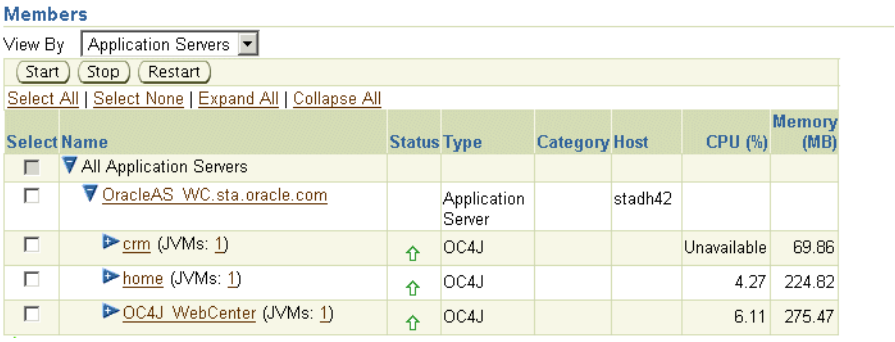
3. 「OC4J インスタンスの作成」ページで、次の情報を入力します。
   - **OC4J インスタンス名 :** インスタンスの名前を入力します。
   - 次のどちらかを選択します。
     - **名前を指定して既存のグループに追加 :**「**既存のグループ名**」からグループを選択します。
     - **名前を指定して新規グループに追加 :**「**新規グループ名**」フィールドに、新しいグループの名前を入力します。
   - 「**作成後にこの OC4J インスタンスを起動します。**」を選択します。
4. 「**作成**」をクリックします。

   インスタンスが作成され、確認画面が表示されます。

この OC4J インスタンスのパスワードは、このインストールの `oc4jadmin` ユーザーのパスワードと同じになることに注意してください。

図 6-1 に、「クラスタ・トポロジ」ページの一部を示します。このページには、クラスタに追加された別の OC4J インスタンスが表示されています。

図 6-1 クラスタに追加された OC4J インスタンス



また、次に示す `opmnctl` コマンドを使用すると、インスタンスが追加されたことを確認できます。

```
ORACLE_HOME¥opmn¥bin¥opmnctl status
Processes in Instance: OracleAS_WC.sta.oracle.com
---------------------------------+--------------------+---------+---------
ias-component                    | process-type       |     pid | status
---------------------------------+--------------------+---------+---------
OC4JGroup:default_group          | OC4J:crm           |    9228 | Alive
OC4JGroup:default_group          | OC4J:OC4J_WebCent~ |    8616 | Alive
OC4JGroup:default_group          | OC4J:home          |    8615 | Alive
ASG                              | ASG                |     N/A | Down
```

注意 : Secure Sockets Layer（SSL）を使用するように Remote Management Interface（RMI）を構成した場合は、作成する各 OC4J インスタンスの `rmi.xml` ファイルに、適切な <ssl-config> 要素を追加する必要があります。そうしないと、管理 OC4J インスタンスの opmn.xml ファイルに設定されている接続プロトコル・プロパティの値によっては、Application Server Control から OC4J インスタンスへの管理接続ができないか、セキュアでない RMI プロトコルが使用されます。詳細は、第 A.3 項「Application Server Control コンソールのセキュリティの構成」を参照してください。

関連項目 :『Oracle Containers for J2EE 構成および管理ガイド』の「別の OC4J インスタンスの作成と管理」

6.1.2 OC4J インスタンスの削除

OC4J インスタンスは、次の方法によって削除できます。

- `removeinstance` ユーティリティの使用。このユーティリティは Oracle ホームの `bin` ディレクトリにあります。
- Application Server Control コンソールの使用

どちらの方法でも、インスタンス用に作成されたディレクトリが `j2ee` ディレクトリ構造から削除され、インスタンスの構成データが `opmn.xml` から削除されます。

たとえば、コマンドライン・ユーティリティを使用してインスタンスを削除するには、次の手順を実行します。

1. インスタンスを停止します。

   (UNIX) *ORACLE_HOME*/opmn/bin/opmnctl stopproc process-type=*oc4J_instanceName*
   (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl stopproc process-type=*oc4J_instanceName*

2. インスタンスを削除します。

   (UNIX) *ORACLE_HOME*/bin/removeinstance -instanceName *oc4J_instanceName*
   (Windows) *ORACLE_HOME*¥bin¥removeinstance -instanceName *oc4J_instanceName*

Application Server Control コンソールを使用して OC4J インスタンスを削除するには、次の手順を実行します。

1. 「アプリケーション・サーバー *instance_name*」ページにナビゲートします。
2. 削除するインスタンスに対応する「**削除**」アイコンをクリックします。
3. 確認ページで「**はい**」をクリックします。
4. インスタンスが削除され、確認画面が表示されます。

OC4J インスタンスを削除する際は、次のガイドラインを参考にしてください。

- インストール時に Oracle Application Server によって作成された OC4J `home` インスタンスは削除できません。
- インストール後にユーザーが作成した OC4J インスタンスは削除することができます。

6.2 クラスタ・トポロジの構成

クラスタ・トポロジは、2 つ以上の接続された Oracle Application Server ノードとして定義されます。

クラスタを作成する理由には、次のものがあります。

- Application Server Control の単一インスタンス（管理 OC4J インスタンス）を使用して、クラスタ内のすべてのインスタンスを管理するため。
- 複数の J2EE インスタンスを複数のホストにインストールすることで、J2EE サーバーの高可用性を実現するため。Oracle HTTP Server が J2EE コンテナにリクエストをルーティングし、アプリケーションがデプロイされたときに、J2EE コンテナが新しいアプリケーションのバインディングを Oracle HTTP Server に動的に通知できるようにするため。このシナリオについては、第 6.2.2 項で説明します。
- グループを使用することで、一般的な管理タスクを複数の OC4J インスタンスにおいて自動的に実行するため。**グループ**とは、同じクラスタ・トポロジに属する OC4J インスタンスの集まりです。グループ内の実行されているすべての OC4J インスタンスに対して、構成操作を同時に実行できます。グループの詳細は第 2.3.3.2 項を、追加グループの作成方法は第 6.2.3 項を参照してください。

このリリースでは、次のタイプのクラスタ・トポロジを作成できます。

- 動的ノード検出 : 各ノードのクラスタ・トポロジ・マップが、ノードの追加または削除に伴い自動的に更新され、クラスタの自己管理が可能になります。
- 検出サーバーとしての静的ハブ : クラスタ内の特定のノードが検出サーバーとして機能するように構成されます。この検出サーバーにはクラスタのトポロジ・マップが保持され、残りのノードはこのサーバーを介して相互に接続されます。トポロジ内のハブは、別のトポロジ内のハブと接続できます。
- ゲートウェイによる分離されたトポロジの接続 : この構成は、ファイアウォールによって分離されたトポロジや、異なるサブネット上にあるトポロジを、ゲートウェイとして指定したノードで接続する場合に使用します。
- 手動によるノード構成 : クラスタ内の各ノードのホスト・アドレスとポートを、手動で指定して構成に含めます。これは、Oracle Application Server リリース 2（10.1.2）でサポートされているものと同じクラスタリング・メカニズムです。このメカニズムは、主に下位互換性を維持するためにサポートされています。

クラスタ・トポロジは、次の方法で構成できます。

- インストール時に、「クラスタ・トポロジ構成」ページで、「**このインスタンスを Oracle Application Server クラスタ・トポロジの一部として構成**」オプションを選択します。この方法を使用すると、動的ノード検出クラスタ・トポロジが作成されます。

  詳細は、Oracle Application Server のインストレーション・ガイドを参照してください。

- インストール後に、Application Server Control コンソールを使用します。

  1. 「クラスタ・トポロジ」ページから、「**トポロジ・ネットワーク構成**」をクリックします。
  2. 「トポロジ」セクションで、次の構成のいずれかを選択します。
     - マルチキャストを使用して動的ノード検出の構成中 : 動的なノード検出を構成する場合は、マルチキャスト・アドレスおよびポートを入力します。次に例を示します。

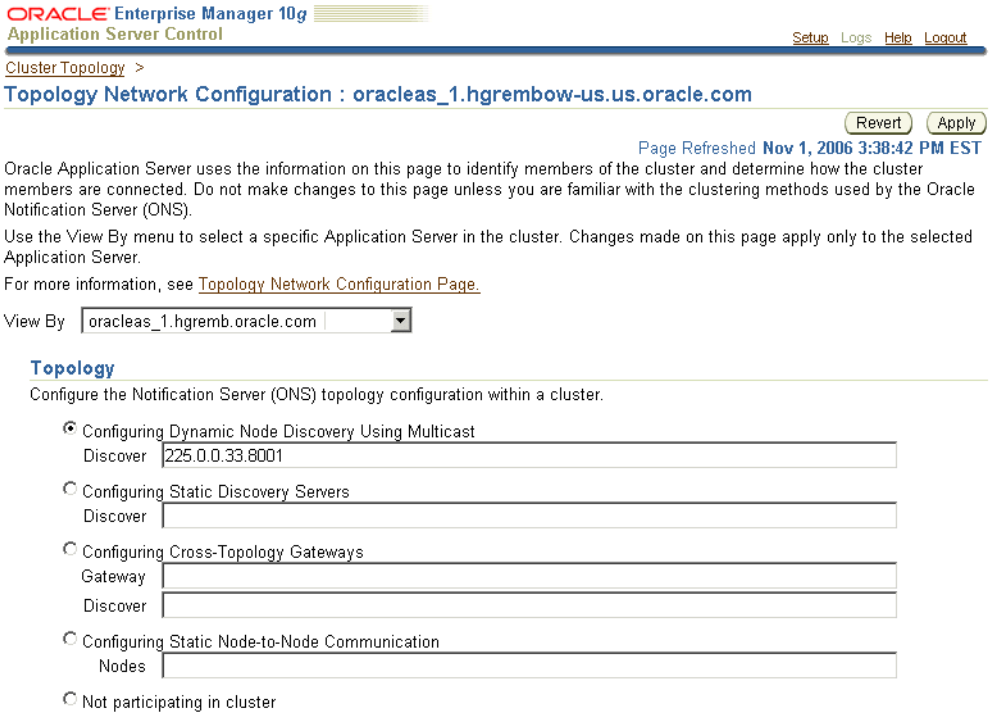
       ```
       225.0.0.33:8001
       ```

       マルチキャスト・アドレスは、224.0.1.0 ～ 239.255.255.255 の範囲内で指定する必要があります。
     - 静的検出サーバーの構成中 : 静的検出を構成する場合は、静的検出サーバーのホスト名または IP アドレスと OPMN リモート・ポートをカンマ区切りのエントリとして入力します。
     - トポロジ間ゲートウェイの構成中 : 各ソース・ノードおよびターゲット・ノードを構成する場合は、各サーバーのホスト名または IP アドレスと OPMN リモート・ポートを指定します。各ノードのデータは、アンパサンド（&）で区切ります。

       また、各ノード固有のクラスタ内で動的検出に使用するマルチキャスト・アドレスおよびポートを指定します。
     - 静的ノード対ノード通信の構成中 : クラスタに含めるすべてのノードの IP アドレスと OPMN リモート・ポートをリスト化します。

     図 6-2 に、「トポロジ・ネットワーク構成」ページを示します。

図 6-2「トポロジ・ネットワーク構成」ページ



3. 「**適用**」をクリックします。

- インストールの後、次の Oracle Process Manager and Notification Server（OPMN）コマンドのいずれかを実行します。

  – `opmnctl`: このユーティリティには、クラスタにインスタンスを追加するのに必要なマルチキャスト・アドレスとポート（multicastAddress:multicastPort）、および Web サイト構成データで、opmn.xml ファイルを更新するためのコマンドが含まれています。構文は次のとおりです。

    ```
    opmnctl config topology update discover=*multicastAddress:multicastPort
    ```

  – `opmnassociate`: このユーティリティを使用すると、1 つの手順のみでインスタンスがクラスタに追加されます。構文は次のとおりです。

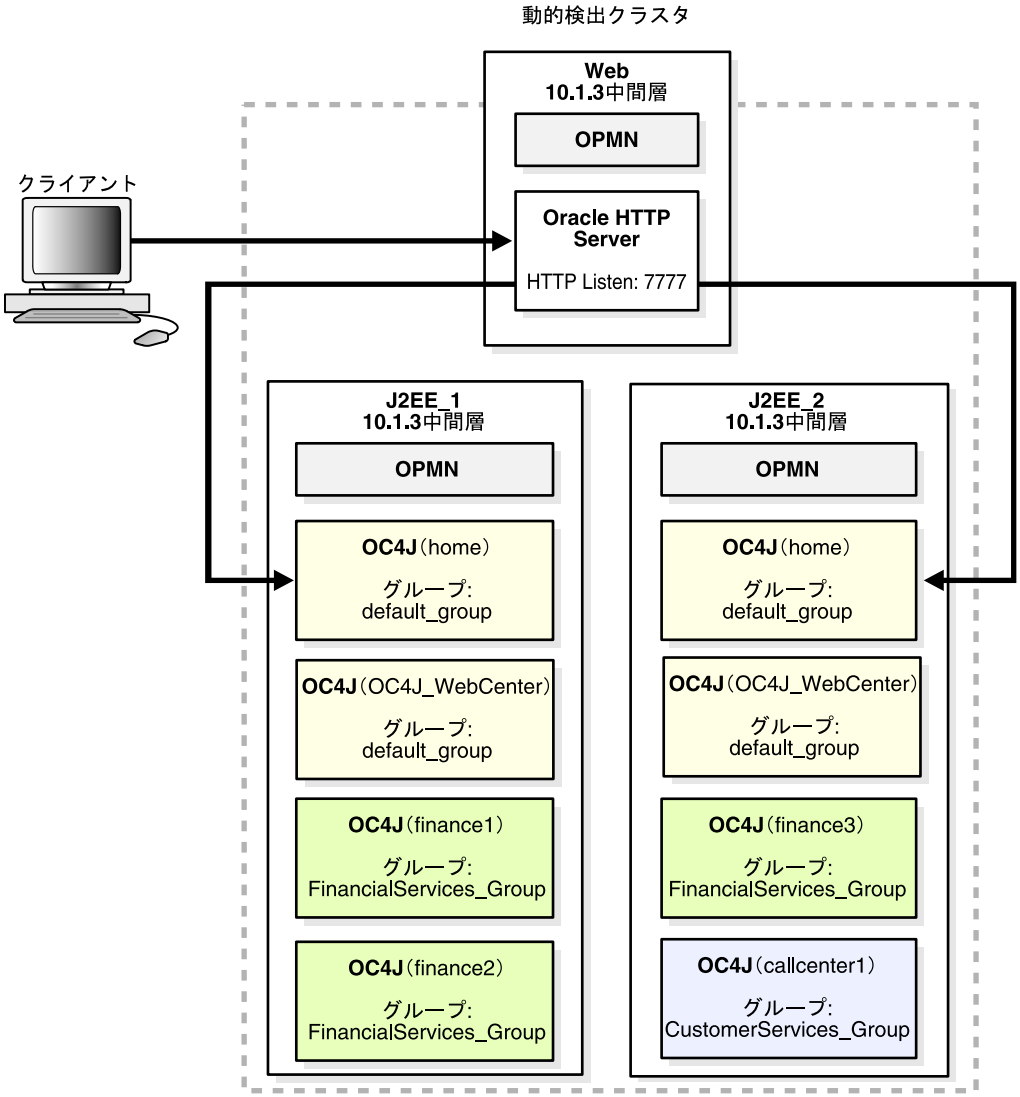
    ```
    opmnassociate *multicastAddress:multicastPort -restart
    ```

    関連項目： クラスタ・トポロジの構成の詳細は、『Oracle Containers for J2EE 構成および管理ガイド』の「クラスタの構成と管理」を参照してください。

次の項では、3 つのノードからなるクラスタを作成し、OC4J インスタンスの 2 つのグループを作成します。次に、OC4J インスタンスを 2 つのノードに追加してグループに割り当て、作成した OC4J インスタンスに複数の JVM を指定します。

図 6-3 に、この構成を示します。

図 6-3 クラスタにおける複数の OC4J 中間層、追加の OC4J インスタンスおよび 1 つの Web サーバー中間層

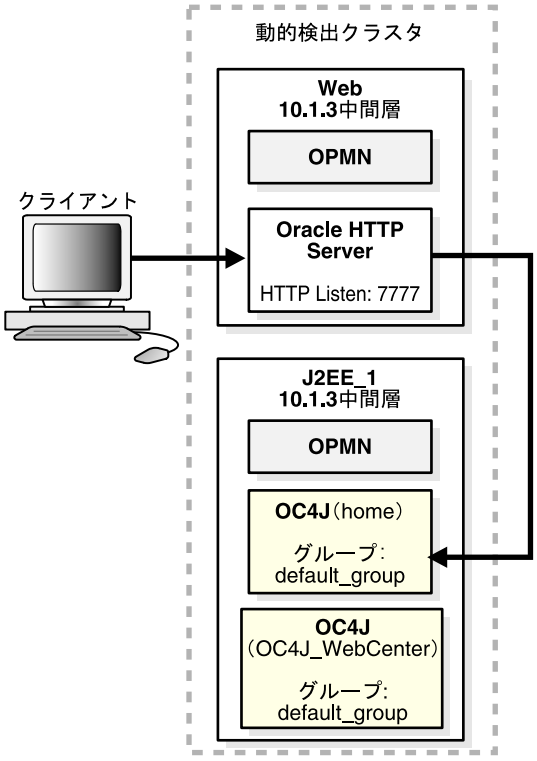


6.2.1 Web サーバーと OC4J の個別ホストへの構成

この例では、1 台のホストに Web サーバーである Oracle HTTP Server をインストールし、別のホストに Oracle WebCenter Framework（Oracle Containers for J2EE（OC4J）が含まれている）をインストールします。その後、動的ノード検出を使用して、インスタンスをクラスタ化します。これにより、Oracle HTTP Server は OC4J にリクエストをルーティングし、OC4J はアプリケーションがデプロイされたときに、新しいアプリケーションのバインディングを Oracle HTTP Server に動的に通知できるようになります。

図 6-4 に、この環境を示します。

図 6-4 クラスタ内の別のホストにおける Web サーバー中間層と Oracle WebCenter Framework 中間層



このシナリオでは、次の Oracle Application Server 中間層インスタンスを別のホストにインストールします（このシナリオでは、インストール時にクラスタを構成しません）。

- Oracle HTTP Server（このシナリオでは Web となります）。これには Oracle HTTP Server と OPMN が配置されます。

  Oracle HTTP Server のインストール時は、「クラスタ・トポロジ構成」ページのデフォルトを受け入れます。

- Oracle WebCenter Framework のみ（このシナリオでは J2EE_1 となります）。これには OC4J、Oracle WebCenter Framework および OPMN が配置されます。

  Oracle WebCenter Framework のインストール時は、「管理 (Administration) 設定」ページで「**このインスタンスで Oracle Enterprise Manager 10*g* ASControl を起動**」を選択します。これにより、OC4J インスタンスが管理 OC4J インスタンスに設定されます。

  「クラスタ・トポロジ構成」ページでは、デフォルトを受け入れます。インスタンスは後の手順でクラスタに追加します。

  次の点に注意してください。

  – OC4J インスタンスを管理 OC4J インスタンスとして構成するように指定すると、そのインスタンス内でホストされている Application Server Control コンソールにより、ローカル OC4J インスタンスと、クラスタ内で管理 OC4J インスタンスとして指定されていないすべてのインスタンスが管理されます。

    このオプションを選択しなかった場合は、インスタンスで ascontrol アプリケーションを起動することにより、インストール後に構成できます。

    ```
    (UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc application=ascontrol
    (Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startproc application=ascontrol
    ```

  – 管理 OC4J インスタンスとして指定されていないこれらのインスタンスの場合、このインスタンスに Application Server Control コンソールがデプロイされますが、起動はされません。

– クラスタ内では、1 つの OC4J インスタンスのみを管理 OC4J インスタンスとして構成することをお薦めします。

中間層インスタンスをインストールしたら、次の手順に従って、これらのインスタンスを動的ノード検出用として構成します。

1. Oracle HTTP Server インスタンスに動的ノード検出を構成するには、`opmnctl config topology` コマンドを使用して、OPMN マルチキャスト検出アドレスを設定します（Application Server Control コンソールはこのインスタンス上で実行されていないため使用できず、`opmnassociate` はデフォルトの OC4J インスタンスに `home` 以外の名前を使用しているため使用できません）。

   たとえば、UNIX 上の Oracle HTTP Server インスタンスをマルチキャスト・アドレス 225.0.0.33 に関連付けるには、次のコマンドを使用します。

   ```
   ORACLE_HOME/opmn/bin/opmnctl config topology update discover=*225.0.0.33:8001
   ORACLE_HOME/opmn/bin/opmnctl reload
   ```

2. Oracle WebCenter Framework インスタンスに動的ノード検出を構成するには、`opmnctl config topology` コマンドまたは Application Server Control コンソールを使用します。この例では、Application Server Control コンソールを使用して、次の手順を実行します。

   a. Application Server Control コンソールの「クラスタ・トポロジ」ページから、「**トポロジ・ネットワーク構成**」をクリックします。

   b. 「トポロジ」セクションで、「**マルチキャストを使用して動的ノード検出の構成中**」を選択します。次に、Oracle HTTP Server インスタンスと同じマルチキャスト・アドレスおよびポートを入力します。このシナリオでは、次を入力します。

      ```
      225.0.0.33:8001
      ```

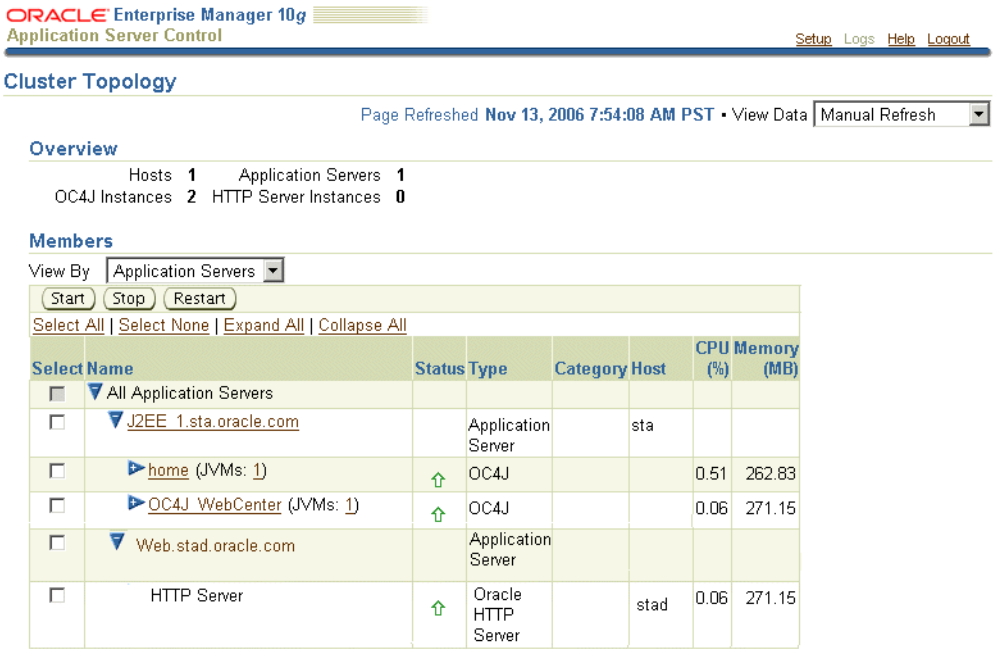
   c. 「**適用**」をクリックします。

   これで、両方のインスタンスが同じクラスタ・トポロジの一部になりました。

次のいずれかの方法で構成を確認します。

- Application Server Control コンソールを使用して、「クラスタ・トポロジ」ページにナビゲートします。図 6-5 に示すように、このページに両方のインスタンスが表示されます。

図 6-5 クラスタ・トポロジの確認



- opmnctl コマンドを、@cluster オプションを使用して実行します。出力例を次に示します。

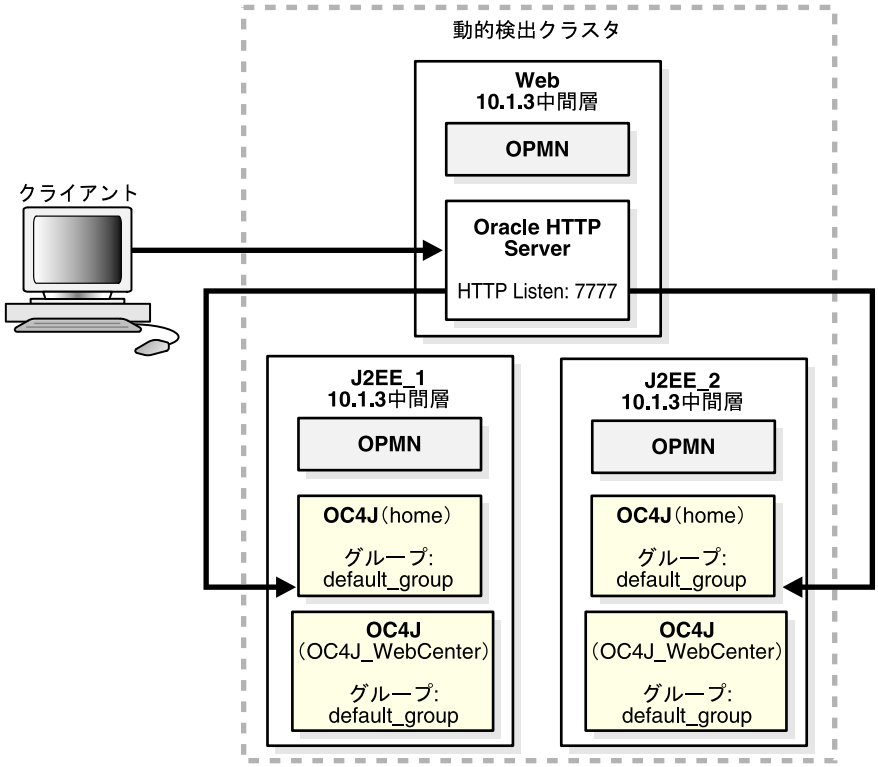
```
ORACLE_HOME/opmn/bin/opmnctl @cluster status
Processes in Instance: J2EE_1.sta.oracle.com
---------------------------------+--------------------+---------+---------
ias-component                    | process-type       |     pid | status
---------------------------------+--------------------+---------+---------
OC4JGroup:default_group          | OC4J:OC4J_WebCent~ |    8616 | Alive
OC4JGroup:default_group          | OC4J:home          |    8615 | Alive
ASG                              | ASG                |     N/A | Down

Processes in Instance: Web.stad.oracle.com
---------------------------------+--------------------+---------+---------
ias-component                    | process-type       |     pid | status
---------------------------------+--------------------+---------+---------
HTTP_Server                      | HTTP_Server        |   25118 | Alive
```

6.2.2 クラスタへの複数の J2EE サーバー中間層の構成

この項で示す例は、前述の第 6.2.1 項の例をベースにしています。ここでは、別の J2EE サーバー中間層（Oracle WebCenter Framework インストール）をクラスタ・トポロジに追加して、テスト用または本番用の高可用性環境をサポートします。

図 6-6 に、この環境を示します。

図 6-6 クラスタにおける複数の J2EE サーバー中間層と 1 つの Web サーバー中間層



このシナリオでは、次を行います。

- 第 6.2.1 項の説明に従ってインスタンスをインストールし、構成します。
- 別の Oracle WebCenter Framework インスタンスをインストールします（このシナリオでは J2EE_2 となります）。

  インストール時は、「管理 (Administration) 設定」ページで「**このインスタンスで Oracle Enterprise Manager 10*g* ASControl を起動**」を選択しないでください。このインスタンスは、クラスタへの追加後に、J2EE_1 の管理 OC4J インスタンスによって管理されます。

次に、このインスタンスに動的ノード検出を構成するには、`opmnctl` コマンドを使用して、前のインスタンスと同じクラスタに追加します。

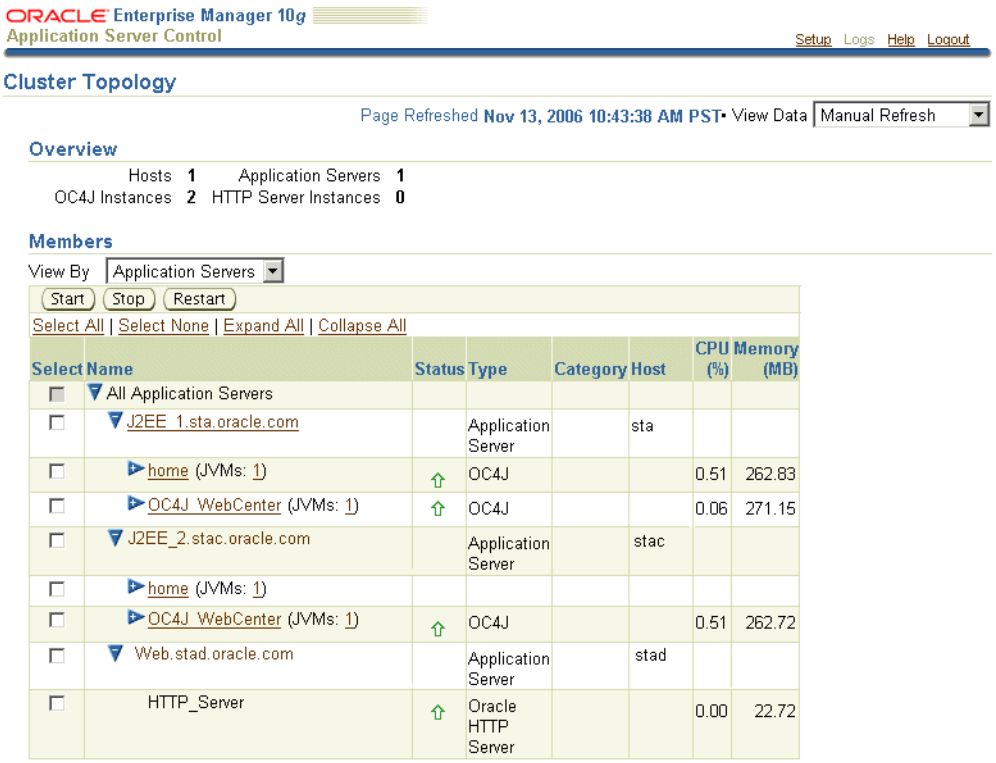
たとえば、UNIX 上の J2EE_2 インスタンスをマルチキャスト・アドレス 225.0.0.33 に関連付けるには、次のコマンドを使用します。

```
ORACLE_HOME/opmn/bin/opmnctl config topology update discover=*225.0.0.33:8001
ORACLE_HOME/opmn/bin/opmnctl reload
```

これで、このインスタンスはクラスタ・トポロジの一部になり、J2EE_1 の管理 OC4J インスタンスによって管理されます。この OC4J インスタンスは、Apache JServ Protocol（AJP）を使用して、Oracle HTTP Server からのリクエストを受信して応答します。

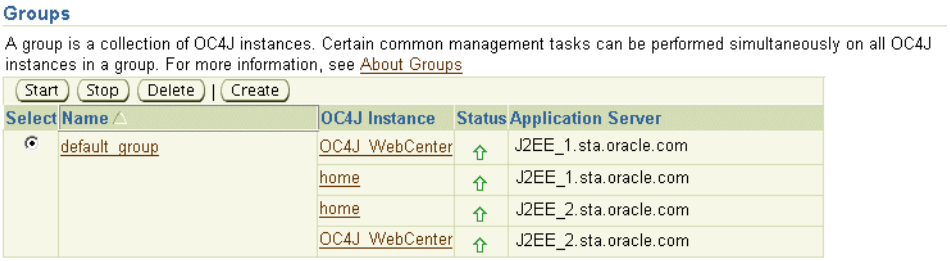
Application Server Control コンソールまたは `opmnctl @cluster status` コマンドを使用して、構成を確認します。たとえば、Application Server Control コンソールを使用して確認する場合は、「クラスタ・トポロジ」ページにナビゲートします。このページの「メンバー」セクションには、図 6-7 に示すように、3 つのインスタンスがすべて表示されます。

図 6-7 更新したクラスタ・トポロジの確認



デフォルトの OC4J インスタンスである home インスタンスおよび OC4J_WebCenter インスタンスは、default_group グループの一部です。図 6-8 は、「クラスタ・トポロジ」ページの「グループ」セクションを示しています。

図 6-8 default_group グループ



6.2.3 追加グループの作成

グループとは、同じクラスタ・トポロジに属する OC4J インスタンスの集まりです。グループ内の実行されているすべての OC4J インスタンスに対して、構成操作を同時に実行できます。

グループは追加作成できます。このシナリオでは、次の 2 つの空グループを作成します。

- FinancialServices_Group
- CustomerServices_Group

グループごとに次の手順を実行します。

1. 「クラスタ・トポロジ」ページの「グループ」セクションで、「**作成**」をクリックします。
2. 「**グループ名**」に「FinancialServices_Group」を入力します。
3. 「**作成**」をクリックします。
4. ステップ 1 から 3 を繰り返します。ただし、「**グループ名**」には「CustomerServices_Group」を入力します。

次の項では、新しい OC4J インスタンスを作成して、それらをグループに追加します。

6.2.4 OC4J インスタンスの追加とグループへの追加

第 6.1 項の説明では、OC4J インスタンスを既存の Oracle ホームに追加しました。この項では、次の表に従って、インスタンスを追加作成し第 6.2.3 項で作成したグループに追加します。

アプリケーション・サーバー・インスタンス	OC4J インスタンス名	グループ名
J2EE_1	finance1	FinancialServices_Group
J2EE_1	finance2	FinancialServices_Group
J2EE_2	finance3	FinancialServices_Group
J2EE_2	callcenter1	CustomerServices_Group

4 つの OC4J インスタンスを作成するには、この表の情報を使用して、インスタンスごとに次の手順を実行します。

1. 「アプリケーション・サーバー J2EE_1.*hostname*」などの、「アプリケーション・サーバー *instance_name*」ページにナビゲートします。
2. 「**OC4J インスタンスの作成**」をクリックします。
3. 「OC4J インスタンスの作成」ページで、次の情報を入力します。
   - **OC4J インスタンス名 :** インスタンスの名前を入力します。たとえば、J2EE_1 インスタンスでは「`finance1`」などを入力します。
   - 「**名前を指定して既存のグループに追加**」を選択し、「**既存のグループ名**」から適切なグループを選択します。
4. 「**作成後にこの OC4J インスタンスを起動します。**」を選択します。
5. 「**作成**」をクリックします。

   インスタンスが作成され、確認画面が表示されます。

図 6-9 に、新しい OC4J インスタンスが表示された、「クラスタ・トポロジ」ページの「メンバー」セクションを示します。

図 6-9 新しい OC4J インスタンスが表示された「クラスタ・トポロジ」ページ

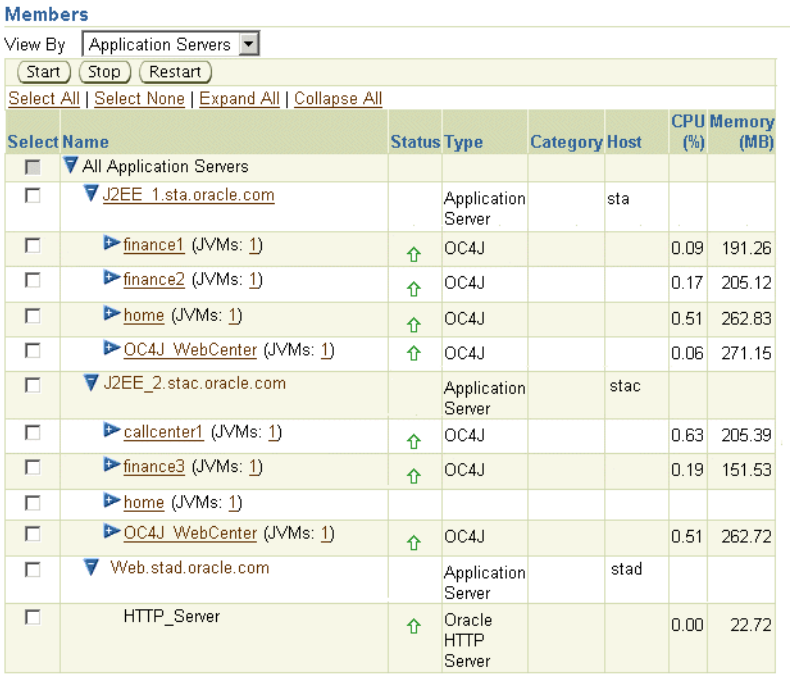
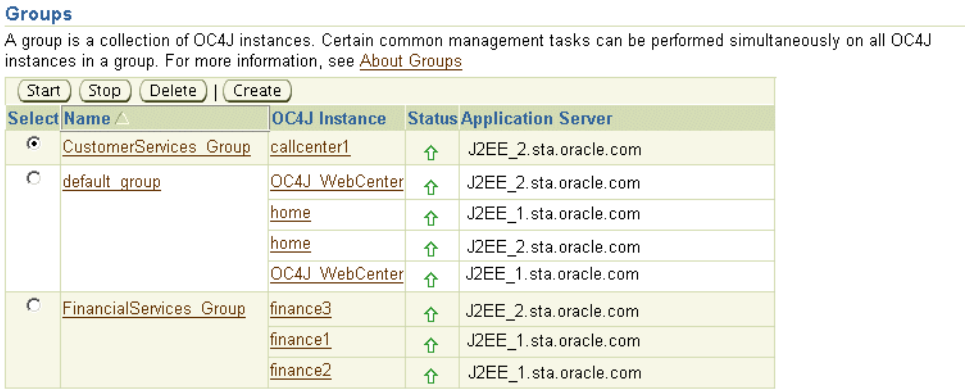


図 6-10 に、新しいグループとそのメンバーが表示された、「クラスタ・トポロジ」ページの「グループ」セクションを示します。

図 6-10 新しいグループが表示された「クラスタ・トポロジ」ページ



これで、第 6.2 項の図 6-3 に示されたクラスタが構成されました。

6.2.5 複数の JVM の作成

OC4J は、標準の Java Development Kit（JDK）の Java 仮想マシン（JVM）上で実行されます。デフォルトでは、各 OC4J インスタンスは 1 つの JVM を使用します。しかし、1 つの OC4J インスタンスが複数の JVM 上で実行されるように構成できます。

この場合、OC4J インスタンスは複数のプロセスで実行されるのが基本です。これによって、デプロイ済のアプリケーションのパフォーマンスが向上し、一定レベルのフォルト・トレラントが実現されます。ただし、複数の JVM が効果的に動作するには、追加のハードウェア・リソースが必要です。

注意： アクティブな Application Server Control（`ascontrol` アプリケーションで表される）をホストする OC4J インスタンスは複数の JVM を実行するように構成できません。

この例では、第 6.2.4 項で作成した各 OC4J インスタンスに追加の JVM を作成します。各 OC4J インスタンスで次の手順を実行します。

1. OC4J インスタンスのホーム・ページにナビゲートします。
2. 「**管理**」をクリックします。
3. 必要に応じて「開く」アイコンをクリックし、表の「プロパティ」セクションを開きます。「**サーバー・プロパティ**」行の「**タスクに移動**」をクリックします。
4. 「**VM のプロセス数**」フィールドに、構成する JVM の数を入力します。
5. 「**適用**」をクリックします。
6. 「クラスタ・トポロジ」ページにナビゲートし、変更した OC4J インスタンスを選択して、「**再起動**」をクリックします。確認ページで「**はい**」をクリックします。

6.3 リバース・プロキシとしての 10.1.2 OracleAS Web Cache の構成

10*g* リリース 3（10.1.3.2.0）の中間層インスタンスには、リリース 2（10.1.2）の OracleAS Web Cache をリバース・プロキシとして使用できます。リバース・プロキシ・サーバーとして、OracleAS Web Cache は中間層サーバーへのゲートウェイとして機能します。

次の各トピックでは、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスに、OracleAS Web Cache リリース 2（10.1.2）をリバース・プロキシとして構成する方法を説明します。

- リバース・プロキシとしての OracleAS Web Cache インスタンスの構成
- リバース・プロキシとしての OracleAS Web Cache クラスタの構成

> **関連項目：** OracleAS Web Cache のリバース・プロキシとしての使用の詳細、および OracleAS Web Cache クラスタの詳細は、『Oracle Application Server Web Cache 管理者ガイド』を参照してください。

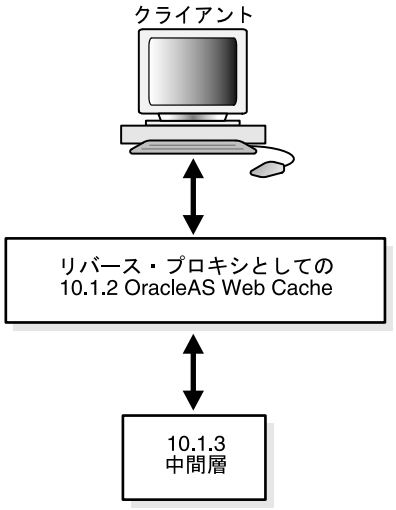
6.3.1 リバース・プロキシとしての OracleAS Web Cache インスタンスの構成

中間層インスタンスには、リリース 2（10.1.2）の OracleAS Web Cache をリバース・プロキシとして使用できます。この項の手順では、次のことを前提にしています。

- リリース 2（10.1.2）の OracleAS Web Cache のスタンドアロン・キットがインストールされていること。このキットは、Oracle Application Server Companion CD（OTN で入手可能）に収録されています。
- 『Oracle Application Server Web Cache 管理者ガイド』の説明に従って、OracleAS Web Cache が構成されていること。
- 10*g* リリース 3（10.1.3.2.0）の中間層インスタンスがインストールされていること。

図 6-11 に、この項で説明するシナリオを示します。

図 6-11 リバース・プロキシとしての OracleAS Web Cache



リリース 2（10.1.2）のスタンドアロン OracleAS Web Cache インスタンスから、次の手順を実行します。

1. OracleAS Web Cache のユーザー名とパスワードを使用して、OracleAS Web Cache Manager にログインします。デフォルトのユーザー名は `ias_admin`、パスワードはインストール時に指定したものです。次の URL を使用します。*`port`* には、OracleAS Web Cache の管理ポートを指定します。

   `http://`*`hostname:port`*`/webcacheadmin`

デフォルトのポートは、9400 です。OracleAS Web Cache スタンドアロン・インストールの管理ポート番号は、*Oracle_Home*/webcache/webcache.xml ファイルで確認できます。Oracle Application Server インストールの一部である OracleAS Web Cache のポート番号を探すには、Application Server Control コンソールで「**ポート**」リンクをクリックします。

2. ナビゲータ・フレームで、「**Origin Servers, Sites, and Load Balancing**」→「**Origin Servers**」を選択します。
3. 「Origin Servers」ページで、「Application Web Servers」セクションの「**Add**」をクリックします。
4. 「Add Application Web Server」ダイアログ・ボックスに、次の情報を入力します。
   - 「**Hostname**」フィールドに、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスにあるオリジナル・サーバー（Oracle HTTP Server）のホスト名を入力します。
   - 「**Port**」フィールドに、オリジナル・サーバーが OracleAS Web Cache のリクエストを受信するリスニング・ポート番号を入力します。
   - 「**Routing**」フィールドで、「ENABLED」を選択して、OracleAS Web Cache がオリジナル・サーバーにリクエストをルーティングできるようにします。

   このダイアログ・ボックスにおける他のフィールドの詳細は、オンライン・ヘルプまたは『Oracle Application Server Web Cache 管理者ガイド』を参照してください。
5. 「**Submit**」をクリックします。
6. オプションとして、新しいサイトを追加してオリジナル・サーバーにマップすることも、既存のサイトを使用することもできます。新しいサイトを追加するには、ナビゲータ・フレームで、「**Origin Servers, Sites, and Load Balancing**」→「**Site Definitions**」を選択します。

   サイトの追加方法の詳細は、オンライン・ヘルプまたは『Oracle Application Server Web Cache 管理者ガイド』を参照してください。
7. ナビゲータ・フレームで、「**Origin Servers, Sites, and Load Balancing**」→「**Site-to-Server Mapping**」を選択し、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスのオリジナル・サーバーにサイトをマップします。
8. 「Site-to-Server Mapping」ページで、マッピングを選択し、「**Insert Above**」または「**Insert Below**」をクリックします。
9. 「Edit/Add Site-to-Server Mapping」ダイアログ・ボックスで、次の操作を行います。
   - 「**Select from Site definitions**」を選択して、使用するサイト定義を選択します。
   - 「**Select Application Web Servers**」フィールドで、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスからアプリケーション Web サーバーを選択します。

6.3.2 リバース・プロキシとしての OracleAS Web Cache クラスタの構成

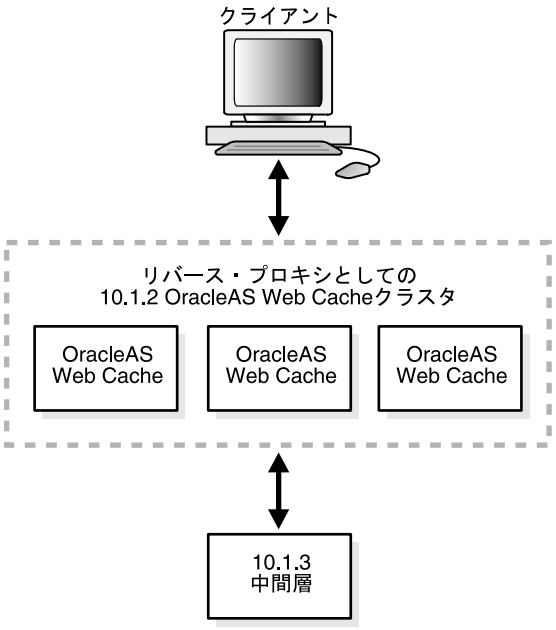
10*g* リリース 3（10.1.3.2.0）の中間層インスタンスには、リリース 2（10.1.2）の OracleAS Web Cache インスタンスのクラスタをリバース・プロキシとして使用できます。

この項の手順では、次のことを前提にしています。

- リリース 2（10.1.2）の OracleAS Web Cache スタンドアロン・キットのインスタンスが 1 つ以上インストールされていること。このキットは、Oracle Application Server Companion CD（OTN で入手可能）に収録されています。
- 『Oracle Application Server Web Cache 管理者ガイド』の説明に従って、OracleAS Web Cache が構成されていること。
- 10*g* リリース 3（10.1.3.2.0）の中間層インスタンスがインストールされていること。

図 6-12 に、この項で説明するシナリオを示します。

図 6-12 リバース・プロキシとしての OracleAS Web Cache クラスタ



OracleAS Web Cache クラスタをリバース・プロキシとして構成するには、次の手順を実行します。

1. 第 6.3.1 項の説明に従って、1 つの OracleAS Web Cache インスタンスをリバース・プロキシとして設定します。
2. OracleAS Web Cache のユーザー名とパスワードを使用して、そのインスタンスの OracleAS Web Cache Manager にログインします。デフォルトのユーザー名は `ias_admin`、パスワードはインストール時に指定したものです。次の URL を使用します。*`port`* には、OracleAS Web Cache の管理ポートを指定します。

   `http://`*`hostname:port`*`/webcacheadmin`

3. 次の手順に従って、キャッシュ・クラスタのプロパティを構成します。
   a. OracleAS Web Cache Manager のナビゲータ・フレームで、「**Properties**」→「**Clustering**」を選択します。
   b. 「Clustering」ページの「**General Cluster Information**」セクションで、「**Edit**」をクリックします。

      オンライン・ヘルプの説明または『Oracle Application Server Web Cache 管理者ガイド』の第 10 章の説明に従ってください。
4. クラスタ内に配置するインスタンスごとに、次の手順を実行して、他のキャッシュをクラスタに追加します。
   a. OracleAS Web Cache Manager のナビゲータ・フレームで、「**Properties**」→「**Clustering**」を選択します。
   b. 「Clustering」ページの「**Cluster Members**」セクションで、「**Add**」をクリックします。

      オンライン・ヘルプの説明または『Oracle Application Server Web Cache 管理者ガイド』の第 10 章の説明に従ってください。

5. すべてのキャッシュをクラスタに追加したら、次の手順に従って、クラスタの構成をクラスタのメンバーに伝播します。

   a. OracleAS Web Cache Manager のナビゲータ・フレームで、「**Operations**」→「**Cache Operations**」を選択します。

   b. 「**All Caches**」を選択し、「**Propagate**」をクリックします。

   c. 「**All Caches**」を選択し、「**Restart**」をクリックして、すべてのキャッシュを再起動します。

6.4 Oracle Application Server 10.1.3 での Oracle Application Server 10.1.2 の構成

既存の Oracle Application Server リリース 2（10.1.2）のコンポーネントおよびアプリケーションで、Oracle Application Server 10*g* リリース 3（10.1.3.2.0）の最新の J2EE 機能を使用するには、Oracle Application Server リリース 2（10.1.2）の中間層にある Oracle HTTP Server を、Oracle Application Server 10*g* リリース 3（10.1.3.2.0）の中間層のフロントエンドとして使用できます。この項では、相互運用性を確保するために、Oracle Application Server 10*g* リリース 3（10.1.3.2.0）と Oracle HTTP Server リリース 2（10.1.2）のインストールおよび構成を行う手順を示します。

ファームまたはクラスタで、次のコンポーネントをインストールするか、または探します。

- **サーバー 1**: Oracle Application Server リリース 2（10.1.2）の J2EE and Web Cache タイプの中間層、または Oracle HTTP Server が存在する他の Oracle Application Server リリース 2（10.1.2）の中間層。
- **サーバー 2**: Oracle Application Server 10*g* リリース 3（10.1.3.2.0）の Oracle WebCenter Framework の中間層。Oracle Application Server 10*g* リリース 3（10.1.3.2.0）のインストールの手順は、10*g* リリース 3（10.1.3.2.0）用の Oracle Application Server のインストレーション・ガイドを参照してください。このインスタンスでは AJP プロトコルを使用する必要があります。HTTP プロトコルを使用している場合は、次のコマンドを実行して、AJP プロトコルに変更してください。

  ```
  ORACLE_HOME_SERVER2/opmn/bin/opmnctl config port update ias-component=default_group
  process-type=instance name portid=default-web-site protocol=ajp
  ORACLE_HOME_SERVER2/opmn/bin/opmnctl reload
  ORACLE_HOME_SERVER2/opmn/bin/opmnctl restartproc ias-component=default_group
  process-type=instance name
  ```

Infrastructure に関連付けられているリリース 2（10.1.2）中間層では、リリース 2（10.1.2）中間層の `ons.conf` ファイルがこの構成では更新されないため、次の手順のステップ 3 から開始してください。この構成では、Oracle Application Server 10*g* リリース 3（10.1.3.2.0）によって 2 つのインスタンス間における接続が開始します。Infrastructure に関連付けられていない中間層では、ステップ 1 から開始します（J2EE and Web Cache タイプの中間層は、Infrastructure に関連付けられている場合と関連付けられていない場合があります）。

構成手順

2 台のサーバーを構成するには、次の手順を実行します。

1. サーバー 1 で、次のように DCM を使用してサーバー 2 を追加します。

   ```
   ORACLE_HOME/dcm/bin/dcmctl addOPMNLink server2_ip:server2_ons_remote_port
   ```

   この例では、次のようになります。

   - *server2_ip* はサーバー 2 の IP アドレスです。IP アドレスを確認するには、次の ping コマンドを使用できます。

     ```
     ping server_name
     ```

- *server2_ons_remote_port* は、サーバーのリモート ONS ポートです。ポート番号は opmn.xml ファイルに記載されています。次の例では、リモート・ポートは 6200 です。

```
<notification-server interface="ipv4">
      <port local="6100" remote="6200" request="6003"/>
```

2. *ORACLE_HOME*/opmn/conf ディレクトリの ons.conf の内容を調べて、サーバー 2 が追加されたことを確認します。このファイルには、カンマで区切られた *hostname/ip:ons_remote_port* エントリのリストが含まれています。リモート・ポートとはサーバー 2 上のポートで、サーバー 1 上の OPMN がサーバー 2 との通信に使用するものです。リストには、次のようなエントリがあります。

```
127.2.148.142:6200
```

3. サーバー 2 で、静的ノード対ノード通信を使用するクラスタにサーバー 1 を追加します。これには、次のように、*ORACLE_HOME*/opmn/conf/opmn.xml のトポロジ・セクションを編集します。

```
<notification-server>
...
   <topology>
      <nodes list="server1_ip:remote_port,server2_ip:remote_port"/>
   </topology>
</notification-server>
```

この例で、*server*_ip* はサーバー 1 またはサーバー 2 の IP アドレスを、*remote_port* は他のサーバーがこのサーバーとの通信に使用するポート番号を示します。たとえば、次のように指定します。

```
127.2.148.142:6200
```

opmn.xml に ONS を構成するには、マスターだけでなく、すべての Oracle RAC インスタンス・ノードを一覧表示する必要があります。host1、host2、host3、host4 を持つ Oracle RAC の場合、リストは次のようになります。

```
list="host1:ONSRemotePort,host2:ONSRemotePort,host3:ONSRemotePort,host4:ONSRemotePo
rt"
```

Oracle RAC 環境では、SSL 設定はすべてのノードで同じ（有効または無効）にする必要があります。

4. サーバー 2 で、OPMN をリロードします。

```
ORACLE_HOME_SERVER2/opmn/bin/opmnctl reload
```

5. 次のコマンドを実行して、両方のサーバーが相互に通信できることを確認します。

- サーバー 1 の場合

```
ORACLE_HOME_SERVER1/opmn/bin/opmnctl @farm status
```

- サーバー 2 の場合

```
ORACLE_HOME_SERVER2/opmn/bin/opmnctl @cluster status
```

これらのコマンドを実行すると、ファームまたはクラスタの一部であるサーバーのリストが生成されます。

6. サーバー 1 で、次のように、*ORACLE_HOME*/Apache/Apache/conf/mod_oc4j.conf ファイルに OC4J マウント・ディレクティブを設定します。

```
Oc4jMount /MyApp instance://server2_instance_name:oc4j_instance_name
Oc4jMount /MyApp/* instance://server2_instance_name:oc4j_instance_name
```

10*g* リリース 3（10.1.3.2.0）にデプロイされる J2EE アプリケーションごとに、マウント・ポイントを 1 つ追加する必要があります。新しいアプリケーションを追加したら、新しいマウント・ポイントも追加する必要があります。

7. サーバー 1 で、構成を更新します（`mod_oc4J.conf` を手動で編集したときは、そのたびに構成の更新が必要です）。

   ```
   ORACLE_HOME/dcm/bin/dcmctl updateConfig
   ```

   関連項目：『Distributed Configuration Management 管理者ガイド』

8. Oracle Application Server 10*g* リリース 3（10.1.3.2.0）インスタンスに Oracle Content DB が含まれている場合は、Oracle Content DB のプロパティを、10.1.2.0.2 Oracle HTTP Server のホスト名およびポート番号を参照するように変更する必要があります。

   プロパティは、Application Server Control コンソールを使用して次のように変更します。

   a. OC4J_Content ホーム・ページにナビゲートして、「**アプリケーション**」を選択します。

   b. 「**コンテンツ**」→「**Content DB の拡張**」をクリックします。

   c. 「管理」タブを選択します。

   d. 「**ドメインのプロパティ**」行で、「**タスクに移動**」アイコンをクリックします。

   e. 「**IFS.DOMAIN.APPLICATION.ApplicationHost**」をクリックします。「**値**」フィールドでホスト名を変更します。「**OK**」をクリックします。

   f. 「**IFS.DOMAIN.APPLICATION.ApplicationPort**」をクリックします。「**値**」フィールドでポート番号を変更します。「**OK**」をクリックします。

9. Oracle Application Server 10*g* リリース 3（10.1.3.2.0）インスタンスで OracleAS Single Sign-On を使用している場合は、次の手順を実行します。

   a. サーバー 1 で、第 6.6 項の「作業 1: SSO 認証の有効化（オプション）」のステップ 1 と 2 を実行します。

   b. サーバー 2（10*g* リリース 3（10.1.3.2.0））インスタンスの次の場所に、新しく作成した osso 構成ファイルをコピーします。

      ```
      (UNIX) ORACLE_HOME/Apache/Apache/conf/osso
      (Windows) ORACLE_HOME¥Apache¥Apache¥conf¥osso
      ```

   c. httpd.conf ファイルで、`mod_osso.conf` が含まれている行からコメント文字（#）を削除します。

   d. mod_oc4j.conf ファイルで、osso 構成ファイル用のエントリを追加します。

      ```
      OssoConfig new_osso.conf_file_path
      ```

   e. サーバー 1 で、構成を更新します

      ```
      ORACLE_HOME/dcm/bin/dcmctl updateConfig
      ```

10. 第 6.6 項の「作業 2: Identity Management の中間層インスタンスの構成」 で説明したように、サーバー 2 を Oracle Internet Directory に関連付けます。

11. サーバー 1 で、Oracle HTTP Server を再起動します。

    ```
    ORACLE_HOME_SERVER1/opmn/bin/opmnctl restartproc ias-component=HTTP_Server
    ```

これで、サーバー 1 の `appserverInstance` を指しているブラウザは、サーバー 2 の `appserverInstance` にある OC4J アプリケーションにアクセスできるようになります。

注意： J2EE and Web Cache タイプの中間層を除くリリース 2（10.1.2）の中間層のインストールでは、Oracle Application Server 10*g* リリース 3（10.1.3.2.0）の中間層を起動する前に、Oracle Application Server リリース 2（10.1.2）の中間層を起動してください。そうしないと、最大 2 分の遅延が生じる場合があります。

この構成では、J2EE 10*g* リリース 3（10.1.3.2.0）インスタンスのアプリケーション停止機能は使用しないでください。Oracle HTTP Server リリース 2（10.1.2）が、アプリケーションが停止した J2EE 10*g* リリース 3（10.1.3.2.0）インスタンスにルーティングを行うと、エラーが発生する場合があります。

注意： リリース 2（10.1.2）は、10*g* リリース 3（10.1.3.2.0）の Application Server Control コンソールから管理できません。10.1.2 インスタンスは Application Server Control コンソール 10*g* リリース 3（10.1.3.2.0）の「クラスタ・トポロジ」ページに表示されますが、その一部の情報は利用できないか正しくない場合があります。次に例を示します。

- 「ポート」ページでは、10.1.2 インスタンスのポートが表示されない場合や、ポート・タイプが NA として表示される場合があります。
- トポロジ・ページの「グループ」セクションでは、グループに 10.1.2 インスタンスが含まれる場合、そのインスタンスのステータスが間違って表示されることがあります。

6.5 OC4J Java Single Sign-On を使用するためのインスタンスの構成

OC4J に付属する軽量のシングル・サインオン・ソリューションである OC4J Java Single Sign-On（Java SSO）を使用するようにインスタンスを構成できます。Java SSO は、追加のインフラストラクチャを必要とせず（OracleAS Single Sign-On や Oracle Access Manager シングル・サインオンでは必要）、使用する ID 管理システムから OC4J を分離します。

基本インストールを選択すると、Java SSO は自動的にデプロイ、構成および起動されます。拡張インストールを選択すると、Java SSO はデプロイされますが、構成および起動は行われません。

Java SSO を使用するためのインスタンスの構成の詳細は、『Oracle Containers for J2EE セキュリティ・ガイド』の Java SSO の設定および構成に関する項を参照してください。

Oracle Internet Directory を使用するための Java SSO の構成の詳細は、『Oracle Containers for J2EE セキュリティ・ガイド』の「Oracle Internet Directory と OC4J の関連付け」を参照してください。

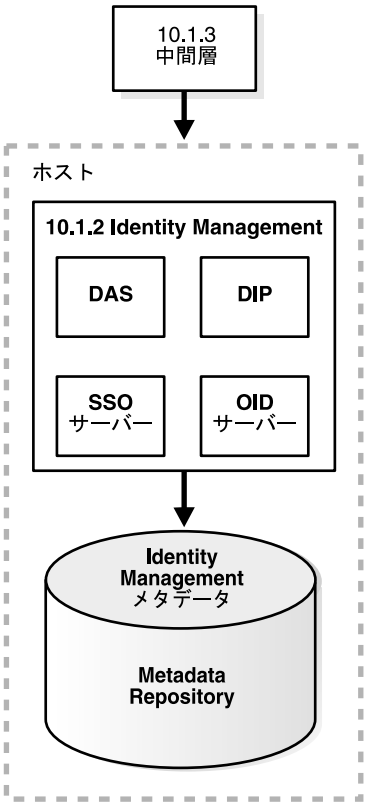
Java SSO を使用するための Oracle WebCenter Framework の構成の詳細は、『Oracle WebCenter Framework 開発者ガイド』の Java Single Sign-On を使用するための WebCenter アプリケーションの構成に関する項を参照してください。

6.6 10.1.4 または 10.1.2 の Oracle Identity Management を使用するためのインスタンスの構成

10.1.3 の中間層インスタンスは、リリース 10.1.4 またはリリース 2（10.1.2）の Oracle Identity Management を使用するように構成できます。

この項では、10.1.3 の中間層インスタンスを、リリース 10.1.4 またはリリース 2（10.1.2）の Oracle Identity Management を使用するように構成する方法について説明します。図 6-13 に、リリース 2（10.1.2）の Oracle Identity Management が構成された中間層インスタンスを示します。

図 6-13 10.1.2 の Identity Management を使用する中間層



開始する前に、次を確認してください。

- Oracle Identity Management インスタンスが起動している（ステータスが「稼働中」になっている）こと。
- Oracle Internet Directory のホストとポート番号がわかっていること。
- `cn=orcladmin`、または `iASAdmins` グループの別のユーザーのパスワードがわかっていること。
- Oracle Internet Directory と OracleAS Single Sign-On を使用するための Oracle WebCenter Framework の構成の詳細は、『Oracle WebCenter Framework 開発者ガイド』の LDAP と Single Sign-On を使用するための WebCenter アプリケーションの構成に関する項を参照してください。

作業 1: SSO 認証の有効化（オプション）

デプロイされているアプリケーションの SSO 認証を有効にするには、「作業 2: Identity Management の中間層インスタンスの構成」で説明する ID 管理ウィザードを使用する前に、次の手順を実行する必要があります。

注意： OracleAS Single Sign-On は、Oracle HTTP Server が含まれる Oracle Application Server のインストール・タイプがインストールされている場合にのみ使用できます。他のインストール・タイプでは、Java SSO を使用できます。第 6.5 項を参照してください。

---

1. Identity Management ホストで、ORACLE_HOME および ORACLE_SID 環境変数を設定します。
2. Identity Management ホストで、ssoreg スクリプトを -remote_midtier オプションを使用して実行します。このファイルは、次のディレクトリにあります。

   (UNIX) *ORACLE_HOME*/sso/bin/ssoreg.sh
   (Windows)*ORACLE_HOME*¥sso¥bin¥ssoreg.bat

   たとえば Linux では、次のように指定します。

```
$ORACLE_HOME/sso/bin/ssoreg.sh -oracle_home_path $ORACLE_HOME
 -config_mod_osso TRUE
 -site_name myhost.com:7778
 -remote_midtier
 -config_file $ORACLE_HOME/Apache/Apache/conf/osso/myosso.conf
 -mod_osso_url http://myhost.com:7778
```

   作成される構成ファイル（この例では mysso.conf）は、不明瞭化された osso 構成ファイルです。
3. 不明瞭化された osso 構成ファイルを、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスにコピーします。
4. 中間層ホストで、次のスクリプトを実行して登録を完了させます。

   (UNIX) *ORACLE_HOME*/Apache/Apache/bin/osso1013 *config_file*
   (Windows) perl *ORACLE_HOME*¥Apache¥Apache¥bin¥osso1013 *config_file*

作業 2: Identity Management の中間層インスタンスの構成

Identity Management を使用するように中間層インスタンスを構成するには、次の手順を実行します。

1. Application Server Control コンソールを使用して、中間層インスタンスの OC4J ホーム・ページにナビゲートします。
2. 「**管理**」をクリックします。
3. 表の「タスク名」列で「**セキュリティ**」が閉じている場合は、それを開きます。「**ID 管理**」行で、「**タスクに移動**」アイコンをクリックします。
4. 「ID 管理」ページで、「**構成**」をクリックします。
5. 「ID 管理の構成 : 接続情報」ページで次のように入力します。
   - **Oracle Internet Directory ホスト :** Oracle Internet Directory ホストの完全修飾名。
   - **Oracle Internet Directory ユーザー DN:** iASAdmins グループの cn=orcladmin などの、ユーザーの識別名。
   - **パスワード :** そのユーザーのパスワード。

     このパスワードは、Oracle Internet Directory で作成した oc4jadmin ユーザーのデフォルト・パスワードとして使用されます。
   - **Internet Directory へ SSL 接続のみを使用** : SSL を使用して中間層コンポーネントを Oracle Internet Directory に接続する場合は、このオプションを選択します。

     「**Oracle Internet Directory SSL ポート**」フィールドに、Oracle Internet Directory の SSL ポート番号を入力します。
   - **Internet Directory へ非 SSL 接続を使用** : SSL 以外の接続を使用して中間層コンポーネントを Oracle Internet Directory に接続する場合は、このオプションを選択します。

     次に、「**Oracle Internet Directory ポート**」フィールドに、Oracle Internet Directory の SSL 以外のポート番号を入力します。

   「**次へ**」をクリックします。

6. 「ID 管理の構成 : Application Server Control」ページでは、Identity Management を管理ユーザーの認証および認可用のセキュリティ・プロバイダとして使用するように Application Server Control を構成するかどうかを指定できます。構成する場合は、「**Oracle Identity Management セキュリティ・プロバイダを使用**」を選択します。

   次の事項に注意してください。

   – 現在のセキュリティ・プロバイダに作成されている Application Server Control 管理者ユーザーは、この変更を行うと、Application Server Control コンソールにアクセスできなくなります。Application Server Control コンソールにアクセスできるのは、Oracle Internet Directory に定義されているユーザーおよびグループのみです。

   – Application Server Control のセキュリティ・プロバイダは後から変更できます。これには、「**設定**」→「**セキュリティ・プロバイダ**」をクリックします。

7. 「ID 管理の構成 : デプロイ済アプリケーション」ページでは、この OC4J インスタンスにデプロイされているアプリケーションのセキュリティ・オプションを指定できます。各アプリケーションに対して、次を実行します。

   - **OID セキュリティ・プロバイダを使用** : Identity Management を認証および認可用のセキュリティ・プロバイダとして使用するようにアプリケーションを構成する場合は、このオプションを選択します。

     デフォルトのアプリケーションのセキュリティ・プロバイダは変更できないことに注意してください。

   - **SSO 認証の有効化** :「**OID セキュリティ・プロバイダを使用**」を選択した場合は、このオプションを選択すると Single Sign-On 認証を使用できます。ただし、最初に Oracle Application Server のインスタンスを OracleAS Single Sign-On サーバーに登録しておく必要があります。詳細は、「作業 1: SSO 認証の有効化（オプション）」を参照してください。

   「**構成**」をクリックします。

8. 操作が完了したら、OC4J インスタンスを再起動する必要があります。「確認」ページで「再起動」をクリックしないでください。かわりに、「クラスタ・トポロジ」ページにナビゲートし、OC4J インスタンスを選択してから、「**再起動**」をクリックします。

これで、中間層は Oracle Identity Management サービスを使用するように構成されました。

関連項目 : 『Oracle Identity Management 概要および配置プランニング・ガイド』

6.7 匿名バインドの有効化と無効化

リリース 2（10.1.2.0.2）から、Oracle Internet Directory で匿名バインド（匿名認証）を有効化および無効化できるようになりました。デフォルトでは、匿名バインドは有効です。

匿名バインドは多くの実行環境で無効になっていると便利ですが、次のようなほとんどの構成の変更時には匿名バインドを有効にする必要があります。

- Oracle Universal Installer での新しいコンポーネントのインストール
- Application Server Control コンソールでのコンポーネントの構成
- Oracle Application Server をインストールしたホストのホスト名、ドメイン名、IP アドレスの変更

6.7.1 実行環境の匿名バインドの無効化

匿名バインドを無効にする手順は次のとおりです。

1. 第 3.2.1 項「中間層インスタンスの起動」で説明しているように、OracleAS Infrastructure に接続しているすべての中間層を停止します。
2. すべての Infrastructure Oracle ホームで OracleAS Infrastructure を停止します。

   (UNIX) *ORACLE_HOME*/opmn/bin/opmnctl stopall
   (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl stopall

3. この手順において必要なので Oracle Internet Directory を起動します。

   (UNIX) *ORACLE_HOME*/bin/oidmon connect=*db_connect_string* start
   (Windows) *ORACLE_HOME*¥bin¥oidmon connect=*db_connect_string* start

4. OracleAS Infrastructure に接続されているすべての中間層と OracleAS Single Sign-On および Oracle Delegated Administration Services を含む Infrastructure Oracle ホームの ias.properties ファイルを変更します。ias.properties ファイルは次のディレクトリにあります。

   (UNIX) *ORACLE_HOME*/config
   (Windows) *ORACLE_HOME*¥config

   ias.properties ファイルに、OIDAnonymousDisabled プロパティを追加して、これを true に設定します。

   OIDAnonymousDisabled=true

5. OracleAS Infrastructure に接続されているすべての中間層と OracleAS Single Sign-On および Oracle Delegated Administration Services を含む Infrastructure Oracle ホームの dads.conf ファイルを変更します。dads.conf ファイルは次のディレクトリにあります。

   (UNIX) *ORACLE_HOME*/Apache/modplsql/conf
   (Windows) *ORACLE_HOME*¥Apache¥modplsql¥conf

   PlsqlDatabaseConnectString パラメータには、デフォルトで、次のような LDAP による名前解決の形式の値が含まれます。

   PlsqlDatabaseConnectString cn=*orcl*, cn=*oraclecontext* NetServiceNameFormat

   この行をコメントアウトします（将来、必要に応じて匿名バインドに戻せるように、この行は削除しないでください）。

   次の行を追加します。これにより、LDAP による名前解決ではなく host:port:service 形式が使用されるように PlsqlDatabaseConnectString パラメータの値が変更されます。

   PlsqlDatabaseConnectString *db_host*:*db_hostdb_listener_port*:*db_service_name*

   この例では、*db_host* は OracleAS Single Sign-On の OracleAS Metadata Repository がインストールされているホストの名前、*db_listener_port* はその OracleAS Metadata Repository のリスナー・ポート、*db_service_name* は OracleAS Metadata Repository のサービス名です。

6. ldapmodify コマンドを使用して匿名バインドを無効にします。このコマンドは、Oracle Internet Directory のある Oracle ホームで使用します。

   次の手順に従います。

   a. テキスト・ファイルを作成し、次の行を含めます。

      ```
      dn:
      changetype: modify
      replace: orclanonymousbindsflag
      orclanonymousbindsflag: 0
      ```

b. ldapmodify コマンドを使用し、前の手順で作成した入力テキスト・ファイルを呼び出します。次の例では、テキスト・ファイルは anon_off.ldif という名前です。

```
(Unix) ORACLE_HOME/bin/ldapmodify -h host -p port -D cn=orcladmin -w password
-v -f anon_off.ldif
(Windows) ORACLE_HOME¥bin¥ldapmodify -h host -p port -D cn=orcladmin -w
password -v -f anon_off.ldif
```

7. Oracle Internet Directory を停止します。

```
(UNIX) ORACLE_HOME/bin/oidmon connect=db_connect_string stop
(Windows) ORACLE_HOME¥bin¥oidmon connect=db_connect_string stop
```

8. 次のように Oracle Internet Directory を含む OracleAS Infrastructure を起動します。Oracle Internet Directory の Oracle ホーム、次いで他のすべての OracleAS Infrastructure の Oracle ホームにおいて OracleAS Infrastructure を起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

9. 第 3.2.1 項「中間層インスタンスの起動」で説明しているように、Infrastructure に接続しているすべての中間層を起動します。

6.7.2 構成変更時の匿名バインドの有効化

匿名バインドを無効にしている場合、Oracle Application Server 中間層または OracleAS Infrastructure の構成を変更する前に、次の手順に従って匿名バインドを有効にする必要があります。

1. 第 3.2.2 項「中間層インスタンスの停止」で説明しているように、OracleAS Infrastructure に接続しているすべての中間層を停止します。
2. すべての Infrastructure Oracle ホームで OracleAS Infrastructure を停止します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopall
```

3. この手順において必要なので Oracle Internet Directory を起動します。

```
(UNIX) ORACLE_HOME/bin/oidmon connect=db_connect_string start
(Windows) ORACLE_HOME¥bin¥oidmon connect=db_connect_string start
```

4. OracleAS Infrastructure に接続されているすべての中間層と OracleAS Single Sign-On および Oracle Delegated Administration Services を含む Infrastructure Oracle ホームの ias.properties ファイルを変更します。ias.properties ファイルは次のディレクトリにあります。

```
(UNIX) ORACLE_HOME/config
(Windows) ORACLE_HOME¥config
```

ias.properties ファイルで、OIDAnonymousDisabled プロパティを false に設定します。

```
OIDAnonymousDisabled=false
```

ファイルにこのプロパティが存在しない場合、または false に設定されている場合、匿名バインドは有効になっています。

5. OracleAS Infrastructure に接続されているすべての中間層と OracleAS Single Sign-On および Oracle Delegated Administration Services を含む Infrastructure Oracle ホームの dads.conf ファイルを変更します。dads.conf ファイルは次のディレクトリにあります。

```
(UNIX) ORACLE_HOME/Apache/modplsql/conf
(Windows) ORACLE_HOME¥Apache¥modplsql¥conf
```

LDAP で名前解決される形式の値を持つ PlsqlDatabaseConnectString パラメータを含む次のような行がコメントアウトされている場合、その行を非コメント化します。この行が削除されている場合、次の形式の行を追加します。

```
PlsqlDatabaseConnectString cn=orcl, cn=oraclecontext NetServiceNameFormat
```

host:port:service 形式の値を持つ PlsqlDatabaseConnectString パラメータを含む行が追加されている場合、それをコメントアウトします。

```
PlsqlDatabaseConnectString db_host:db_hostdb_listener_port:db_service_name
```

6. ldapmodify コマンドを使用して匿名バインドを有効にします。このコマンドは、Oracle Internet Directory のある Oracle ホームで使用します。

   次の手順に従います。

   a. テキスト・ファイルを作成し、次の行を含めます。

      ```
      dn:
      changetype: modify
      replace: orclanonymousbindsflag
      orclanonymousbindsflag: 1
      ```

   b. ldapmodify コマンドを使用し、前の手順で作成した入力テキスト・ファイルを呼び出します。次の例では、テキスト・ファイルは anon_on.ldif という名前です。

      ```
      (Unix) ORACLE_HOME/bin/ldapmodify -h host -p port -D cn=orcladmin -w password
      -v -f anon_on.ldif
      (Windows) ORACLE_HOME¥bin¥ldapmodify -h host -p port -D cn=orcladmin -w
      password -v -f anon_on.ldif
      ```

7. Oracle Internet Directory を停止します。

   ```
   (UNIX) ORACLE_HOME/bin/oidmon connect=db_connect_string stop
   (Windows) ORACLE_HOME¥bin¥oidmon connect=db_connect_string stop
   ```

8. 次のように Oracle Internet Directory を含む OracleAS Infrastructure を起動します。Oracle Internet Directory の Oracle ホーム、次いで他のすべての OracleAS Infrastructure の Oracle ホームにおいて OracleAS Infrastructure を起動します。

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
   (Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
   ```

9. 次のコマンドを使用して、Infrastructure に接続しているすべての中間層を起動します。

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl startall
   (Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startall
   ```

7

ネットワーク構成の変更

この章では、Oracle Application Server ホストのネットワーク構成を変更する手順について説明します。

この章の項目は次のとおりです。

7.1 ネットワーク構成の変更手順の概要

この章では、ネットワーク構成変更のための次の手順を説明します。

- ホスト名、ドメイン名または IP アドレスの変更

  この項では、ホストのホスト名、ドメイン名または IP アドレスを変更する際に、Oracle Application Server を更新する方法について説明します。

- ネットワーク接続のオン / オフの切替え

  この項では、Oracle Application Server ホストを、ネットワークに接続された状態と接続されていない状態の間で切り替える手順について説明します。ネットワークに接続された状態では、DHCP または静的 IP アドレスを使用できます。これらの手順は、Oracle Application Server をラップトップにインストールした後で異なるネットワークに接続するような場合に使用できます。

- 静的 IP アドレスと DHCP の切替え

  この項では、静的 IP アドレスから DHCP、または DHCP から静的 IP アドレスに変更するための手順を説明します。この手順は、静的 IP アドレスでインストールを行った後、より移動性を高めるために DHCP を使用することにした場合や、DHCP を使用していて、静的 IP アドレスを使用してネットワークに接続しなければならない場合に使用できます。

Oracle Internet Directory で匿名バインドが無効になっている場合、構成を変更する前に有効にする必要があります。詳細は、第 6.7 項「匿名バインドの有効化と無効化」を参照してください。

7.2 ホスト名、ドメイン名または IP アドレスの変更

Oracle Application Server をインストールした後、ホストのホスト名、ドメイン名または IP アドレスの変更が必要となる場合があります。

表 7-1 では、ホスト名、ドメイン名および IP アドレスの変更をサポートするインストール・タイプと、該当する手順の参照先をまとめます。

表 7-1 ホスト名、ドメイン名および IP アドレス変更のためにサポートされている手順

インストール・タイプ	ホスト名またはドメイン名の変更	IP アドレスの変更
中間層	サポート対象 詳細は、第 7.2.2 項「中間層インストールのホスト名またはドメイン名の変更」を参照してください。	サポート対象 オペレーティング・システムでアドレスを変更します。Oracle Application Server の更新は必要ありません。
10.1.4 または 10.1.2 の Infrastructure: Identity Management のみ 次のコンポーネントが構成されている Identity Management インストールに適用されます。 ■ Oracle Internet Directory のみ ■ OracleAS Single Sign-On、Oracle Delegated Administration Services および（場合によっては）Oracle Directory Integration and Provisioning ■ Oracle Internet Directory、OracleAS Single Sign-On、Oracle Delegated Administration Services および（場合によっては）Oracle Directory Integration and Provisioning	サポート対象 詳細は、第 7.2.3 項「10.1.4 または 10.1.2 の Identity Management インストールのホスト名、ドメイン名または IP アドレスの変更」を参照してください。	サポート対象 詳細は、第 7.2.3 項「10.1.4 または 10.1.2 の Identity Management インストールのホスト名、ドメイン名または IP アドレスの変更」を参照してください。

表 7-1 ホスト名、ドメイン名および IP アドレス変更のためにサポートされている手順（続き）

インストール・タイプ	ホスト名またはドメイン名の変更	IP アドレスの変更
10.1.4 または 10.1.2 の Infrastructure: Identity Management および Metadata Repository	サポート対象外	サポート対象 詳細は、第 7.2.4 項「Metadata Repository を含む 10.1.4 または 10.1.2 の Infrastructure の IP アドレスの変更」を参照してください。
10.1.4 または 10.1.2 の Infrastructure: Metadata Repository のみ	サポート対象外	サポート対象 詳細は、第 7.2.4 項「Metadata Repository を含む 10.1.4 または 10.1.2 の Infrastructure の IP アドレスの変更」を参照してください。

この項の手順の多くは、chgiphost コマンドを使用します。このコマンドの詳細は、第 7.2.1 項を参照してください。

7.2.1 chgiphost コマンドの概要

chgiphost コマンドライン・ユーティリティでは、中間層インスタンス、Infrastructure、または Identity Management インストールのホスト名、ドメイン名または IP アドレスを変更できます。

このユーティリティは、次のディレクトリにあります。

- UNIX の場合 :

  ORACLE_HOME/chgip/scripts/chgiphost.sh

- Windows の場合 :

  ORACLE_HOME¥chgip¥scripts¥chgiphost.bat

表 7-2 に、このコマンドのオプションを示します。

表 7-2 chgiphost コマンドのオプション

オプション	説明
-version	ユーティリティのバージョンを表示します。
-mid	中間層インスタンスのホスト名、ドメイン名、または IP アドレスを変更します。
-silent	サイレント・モードでコマンドを実行します。
-infra	10.1.4 または 10.1.2 の Infrastructure インスタンスの IP アドレスを変更します。10*g* リリース 3（10.1.3.2.0）には適用しないでください。
-idm	10.1.4 または 10.1.2 の Identity Management インスタンスのみのホスト名、ドメイン名または IP アドレスを変更します。10*g* リリース 3（10.1.3.2.0）には適用しないでください。

chgiphost を使用してホスト名やドメイン名を変更した場合、インスタンス名は更新されないので注意してください。たとえば、元のインスタンス名はホスト名、ドメイン名が付加されて次のような名前であるとします。

```
1013mid.myhost1.mydomain.com
```

ここでホスト名を myhost2 に変更しても、インスタンス名は変わりません。

関連項目：

- 第 7.2.5.1 項「chgiphost のログ・レベルの設定」
- 第 7.2.5.2 項「chgiphost コマンドのカスタマイズ」

7.2.2 中間層インストールのホスト名またはドメイン名の変更

この項では、いずれかの中間層インストール・タイプが含まれるホストのホスト名またはドメイン名、あるいはその両方を変更する方法について説明します。

次の各項では、この手順について説明します。

- 設定前の確認事項
- 作業 1: ホストの準備
- 作業 2: ホスト名の変更
- 作業 3: chgiphost コマンドの実行
- 作業 4: SSO 認証の再有効化
- 作業 5: 他のクラスタ・メンバーのホスト情報の更新
- 作業 6: ファイル内のホスト名の手動更新
- 作業 7: 環境の再起動

設定前の確認事項

手順を開始する前に、次の項目を確認してください。

- 中間層インスタンスが Oracle Internet Directory に登録されている場合は、手順の実行時に `cn=orcladmin` パスワードを指定する必要があります。
- `chgiphost` コマンドを実行する前に、より詳細な情報が表示されるようにログ・レベルを変更することを検討してください。詳細は、第 7.2.5.1 項を参照してください。
- 元のホスト名と同じ文字列が構成ファイル内に記述されている可能性がある場合は、構成ファイルの更新時に `chgiphost` コマンドを実行すると問題が生じることがあります。この問題を回避する方法は、第 7.2.5.2 項を参照してください。
- 手順を開始する前に、元のホスト名および IP アドレスを書き留めておいてください。これらの値の入力が求められます。
- この手順を開始する前に、環境のバックアップを行うことをお薦めします。詳細は、第 V 部「バックアップとリカバリ」を参照してください。

作業 1: ホストの準備

すべてのプロセスを停止して、ホストの変更準備をします。

1. ホストの各中間層インスタンスを停止します。これには、各 Oracle ホームで次のコマンドを実行します。
   - UNIX の場合：

     ORACLE_HOME`/opmn/bin/opmnctl stopall`

   - Windows の場合：

     ORACLE_HOME`¥opmn¥bin¥opmnctl stopall`

2. `/etc/init.d` スクリプトなどの自動スタートアップ・スクリプトを設定している場合は、それらを無効にして、ホストの再起動後に Oracle Application Server プロセスが自動的に起動されないようにします。
3. 中間層インスタンスが使用している Oracle Internet Directory が起動していることを確認します。

作業 2: ホスト名の変更

オペレーティング・システムで、ホスト名、ドメイン名、またはその両方を更新します。ここでの手順を実行する方法の詳細は、該当するオペレーティング・システムのドキュメントを参照してください。必要に応じて IP アドレスも変更できます。

1. オペレーティング・システムで、ホスト名またはドメイン名、またはその両方を適切に変更します。
2. オペレーティング・システムで再起動が必要な場合は、ホストを再起動します。
3. ネットワークの別のホストから、このホストに ping が実行できることを確認します。必ず新しいホスト名を使用して ping を実行し、すべてが正しく解決されていることを確認してください。

作業 3: chgiphost コマンドの実行

ホスト上の各中間層インスタンスについてこれらの手順を実行します。1 つの中間層インスタンスですべての手順を完了してから次のインスタンスの作業を行ってください。

1. 中間層インスタンスをインストールしたユーザーとしてホストにログインします。
2. ORACLE_HOME 環境変数に中間層の Oracle ホームが設定されていることを確認します。変数を指定するときは、スラッシュ（UNIX）や円記号（Windows）を最後に使用しないでください。
3. UNIX システムの場合は、LD_LIBRARY_PATH、LD_LIBRARY_PATH_64、LIB_PATH または SHLIB_PATH 環境変数を、表 1-1 に示されている適切な値に設定します。実際に設定が必要な環境変数および値は、UNIX オペレーティング・システムのタイプによって異なります。
4. 中間層の Oracle ホームで次のコマンドを実行します。

   - UNIX の場合 :

     ```
     cd ORACLE_HOME/chgip/scripts
     ./chgiphost.sh -mid
     ```

   - Windows の場合 :

     ```
     cd ORACLE_HOME¥chgip¥scripts
     cmd /c chgiphost.bat -mid
     ```

   chgiphost コマンドでは、表 7-3 に示す情報の入力を求めるプロンプトが表示されます。プロンプトには、値がカッコに囲まれて表示される場合があります。別の値を入力するか、[Enter] キーを押して、自動入力された値を受け入れます。

表 7-3 chgiphost -mid のプロンプトとアクション

プロンプト	アクション
`Enter fully qualified hostname (hostname.domainname) of destination`	新しい完全修飾ホスト名を入力します。これは、新しいホスト名、ドメイン名、またはその両方です。
`Enter fully qualified hostname (hostname.domainname) of source`	元の完全修飾ホスト名またはドメイン名、あるいはその両方を入力します。
`Enter valid IP Address of destination`	ホストの IP アドレスを変更した場合は、新しい IP アドレスを入力します。それ以外の場合は、現在の IP アドレスを入力します。
`Enter valid IP Address of source`	ホストの IP アドレスを変更した場合は、古い IP アドレスを入力します。それ以外の場合は、現在の IP アドレスを入力します。
`OIDAdmin Password:`	この中間層インスタンスが登録されている Oracle Internet Directory の `cn=orcladmin` パスワードを入力します。

5. 次のディレクトリ内にあるファイルで、エラーが発生していないかチェックして、ツールが正常に実行されたことを確認します。

   - UNIX の場合 :

     ORACLE_HOME/chgip/log

   - Windows の場合 :

     ORACLE_HOME¥chgip¥log

作業 4: SSO 認証の再有効化

中間層インスタンスが SSO 認証用に有効化された場合は、再び有効化する必要があります。その場合は、第 6.6 項の「作業 1: SSO 認証の有効化（オプション）」の手順に従います。

作業 5: 他のクラスタ・メンバーのホスト情報の更新

中間層インスタンスがトポロジ・クラスタのメンバーである場合、他のクラスタ・メンバーで opmn.xml ファイルにあるトポロジ情報の更新が必要になる場合があります。

中間層インスタンスが動的検出クラスタの一部である場合、opmn.xml ファイルの変更は不要です。

ただし、クラスタが次のタイプのいずれかである場合、opmn.xml ファイルを更新する必要があります。

- 検出サーバーとしての静的ハブのタイプである場合、<discover> 要素でホスト名またはドメイン名を変更する必要があります。次に要素の例を示します。

```
<topology>
   <discover list="node1.com:6201,node2.com:6202"/>
</topology>
```

- トポロジ間ゲートウェイのタイプの場合、<gateway> 要素でホスト名またはドメイン名を変更する必要があります。次に要素の例を示します。

```
<topology>
 <gateway list="node1.com:6201&node2.com:6202&node3.com:6203"/>
 <discover list="*224.0.1.37:8205"/>
</topology>
```

- 静的なノード間タイプである場合、<nodes> 要素でホスト名またはドメイン名を変更する必要があります。次に要素の例を示します。

```
<topology>
 <nodes list="node1-sun:6201,node2-sun:6202"/>
</topology>
```

作業 6: ファイル内のホスト名の手動更新

状況によっては、ファイル内のホスト名を手動で更新する必要があります。

- ファイルを編集し、Oracle ホームのパスなどユーザー定義パラメータの一部としてホスト名を入力している場合、そのホスト名は *chgiphost* コマンドを実行しても自動的には更新されません。このような場合にホスト名を更新するには、そのファイルを手動で編集する必要があります。たとえば、UNIX では plsql.conf ファイルに、/net/dsun1/private/... のようにホスト名を含む NFS パスが記述されている場合があります。
- 通常、WebCenter アプリケーションでは、同じホスト上で実行されているポートレットを使用します。たとえば、OC4J インスタンスの OC4J_Apps で実行されている WebCenter アプリケーションが、OC4J インスタンスの OC4J_WebCenter で実行されているポートレットを使用する場合があります。

  WebCenter アプリケーションのホスト名またはドメイン名（あるいはその両方）を変更する場合は、これらのポートレットに対する内部参照を手動で更新する必要があります。

WebCenter アプリケーションでは、ポートレット・プロデューサの接続情報が、次のディレクトリにある connections.xml ファイルに格納されます。

ORACLE_HOME/j2ee/OC4J_instance/applications/application_name/adf/META-INF

テキスト・エディタを使用して、ホスト名を検索して、すべてのホスト名を適切な新しい値に置き換えます。この情報は、次の例の太字部分のようになります。

```
<urlconnection name="PdkPortletProducer1_
115996420445613411da8-010e-1000-8002-8c5707c5e057-urlconn"
url="http://apphost1.mycompany.com:8890/jpdk/providers"/>
.
.
.
<wsconnection
description="http://apphost1.mycompany.com:6688/richtextportlet/portlets/wsrp2?WSDL
">
    <soap
addressUrl="http://apphost1.mycompany.com:6688/richtextportlet/portlets/WSRP_v2_
PortletManagement_Service" xmlns="http://schemas.xmlsoap.org/wsdl/soap/" />
```

同様に、Oracle Content DB を使用していて、接続情報が変更された場合は、次のように connections.xml を更新できます。

```
<StringRefAddr addrType="jcr_oracle.ifs.jcr.configuration.serverUrl">
<Contents>https://ctdbhost1.mycompany.com:4444/content/ws</Contents></StringRefAddr
>
```

- また、chgiphost コマンドでは、ドキュメンテーション・ファイル内に記述されたホスト名も変更されません。これらのファイルについては、手動で編集してホスト名を更新する必要があります。*ORACLE_HOME*/Apache/Apache/htdocs ディレクトリの index.html.* ファイルは、このようなファイルの例です。

作業 7: 環境の再起動

中間層インスタンスを再起動して、これまでの手順を開始する前の状態に構成を戻します。

1. ホストの各中間層インスタンスを起動します。これには、各 Oracle ホームで次のコマンドを実行します。

   (Unix) *ORACLE_HOME*/opmn/bin/opmnctl startall
   (Windows) *ORACLE_HOME*¥opmn¥bin¥opmnctl startall

2. この手順の開始前に、Oracle Application Server が自動的に起動するプロセスを無効にしていた場合は、有効化します。

7.2.3 10.1.4 または 10.1.2 の Identity Management インストールのホスト名、ドメイン名または IP アドレスの変更

10.1.4 または 10.1.2 の Identity Management インストールに関連付けられている 10*g* リリース 3（10.1.3.2.0）中間層インスタンスが存在する場合があります。

この項では、Identity Management インストールを含むホストのホスト名、ドメイン名または IP アドレスを変更する方法について説明します。この手順は、次のものを含む Identity Management のみのインストールすべてに適用されます。

- Oracle Internet Directory のみが構成されている Identity Management
- OracleAS Single Sign-On、Oracle Delegated Administration Services および Oracle Directory Integration and Provisioning が構成されている Identity Management
- Oracle Internet Directory、OracleAS Single Sign-On、Oracle Delegated Administration Services および Oracle Directory Integration and Provisioning が構成されている Identity Management

次の各項では、この手順について説明します。

設定前の確認事項

手順を開始する前に、次の項目を確認してください。

- `chgiphost` コマンドを実行する前に、より詳細な情報が表示されるようにログ・レベルを変更することを検討してください。詳細は、第 7.2.5.1 項を参照してください。
- 元のホスト名と同じ文字列が構成ファイル内に記述されている可能性がある場合は、構成ファイルの更新時に `chgiphost` コマンドを実行すると問題が生じることがあります。この問題を回避する方法は、第 7.2.5.2 項を参照してください。
- 手順を開始する前に、元のホスト名および IP アドレスを書き留めておいてください。これらの値の入力が求められます。
- この手順を開始する前に、環境のバックアップを行うことをお薦めします。詳細は、第 V 部「バックアップとリカバリ」を参照してください。

作業 1: 中間層インスタンスの停止

Identity Management を使用する各 10*g* リリース 3（10.1.3.2.0）中間層インスタンスで、次のコマンドを使用して Application Server Control コンソールと中間層インスタンスを停止します。

- UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl stopall
```

- Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl stopall
```

作業 2: ホストの準備

リリース 2（10.1.2）Identity Management インスタンス上ですべてのプロセスを停止して、ホストのホスト名変更を準備します。

1. ORACLE_HOME 環境変数を設定します。
2. Oracle ディレクトリ・サーバー、Directory Integration and Provisioning データ・サーバー（構成されている場合）、レプリケーション・サーバーなどのサーバーだけでなく Application Server Control コンソールも含めて Identity Management インストールを停止します。たとえば、UNIX の場合、次のコマンドを入力します。

```
ORACLE_HOME/bin/emctl stop iasconsole
ORACLE_HOME/bin/oidctl server=odisrv instance=instance_number stop
ORACLE_HOME/bin/oidctl connect=global_db_name server=oidrepld instance=instance_
number stop
ORACLE_HOME/bin/oidctl server=oidldapd instance=instance_number stop
ORACLE_HOME/opmn/bin/opmnctl stopall
```

3. `/etc/init.d` スクリプトなどの自動スタートアップ・スクリプトを設定している場合は、それらを無効にして、ホストの再起動後に Oracle Application Server プロセスが自動的に起動されないようにします。

作業 3: ホスト名または IP アドレスの変更

オペレーティング・システムで、ホスト名、ドメイン名または IP アドレスを更新します。ここでの手順を実行する方法の詳細は、該当するオペレーティング・システムのドキュメントを参照してください。

1. オペレーティング・システムで、ホスト名またはドメイン名、またはその両方を適切に変更します。
2. オペレーティング・システムで再起動が必要な場合は、ホストを再起動します。
3. ネットワークの別のホストから、このホストに ping が実行できることを確認します。必ず新しいホスト名を使用して ping を実行し、すべてが正しく解決されていることを確認してください。

作業 4: chgiphost コマンドの実行

Identity Management インスタンスにおいて次の手順を実行します。

1. Identity Management をインストールしたユーザーとしてホストにログインします。
2. ORACLE_HOME 環境変数を設定します。ORACLE_HOME 変数を指定するときは、スラッシュ（UNIX）や円記号（Windows）を最後に使用しないでください。
3. UNIX システムの場合は、LD_LIBRARY_PATH、LD_LIBRARY_PATH_64、LIB_PATH または SHLIB_PATH 環境変数を、表 1-1 に示されている適切な値に設定します。実際に設定が必要な環境変数および値は、UNIX オペレーティング・システムのタイプによって異なります。
4. Identity Management の Oracle ホームで次のコマンドを実行します。

   - UNIX の場合 :

     ```
     cd ORACLE_HOME/chgip/scripts
     ./chgiphost.sh -idm
     ```

   - Windows の場合 :

     ```
     cd ORACLE_HOME¥chgip¥scripts
     cmd /c chgiphost.bat -idm
     ```

   chgiphost コマンドでは、表 7-4 に示す情報の入力を求めるプロンプトが表示されます。プロンプトには、値がカッコに囲まれて表示される場合があります。別の値を入力するか、[Enter] キーを押して、自動入力された値を受け入れます。

表 7-4 chgiphost -idm のプロンプトとアクション

プロンプト	アクション
Enter fully qualified hostname (hostname.domainname) of destination	システムのホスト名またはドメイン名を変更した場合は、新しい完全修飾ホスト名を入力します。それ以外の場合は、現在の完全修飾ホスト名を入力します。
Enter fully qualified hostname (hostname.domainname) of source	システムのホスト名またはドメイン名を変更した場合は、古い完全修飾ホスト名を入力します。それ以外の場合は、現在の完全修飾ホスト名を入力します。
Enter valid IP Address of destination	システムの IP アドレスを変更した場合は、新しい IP アドレスを入力します。それ以外の場合は、現在の IP アドレスを入力します。
Enter valid IP Address of source	システムの IP アドレスを変更した場合は、古い IP アドレスを入力します。それ以外の場合は、現在の IP アドレスを入力します。

5. 次のディレクトリ内にあるファイルで、エラーが発生していないかチェックして、ツールが正常に実行されたことを確認します。

```
(UNIX) ORACLE_HOME/chgip/log
(Windows) ORACLE_HOME¥chgip¥log
```

作業 5: 環境の再起動

Identity Management インストールと、この手順の中で停止した他のすべての Infrastructure インスタンスを再起動します。

1. 次のコマンドを使用して、Identity Management インスタンスを再起動します。

   - UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl stopall
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/bin/emctl start iasconsole
```

   - Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl stopall
ORACLE_HOME¥opmn¥bin¥opmnctl startall
ORACLE_HOME¥bin¥emctl start iasconsole
```

2. この手順の開始前に、Oracle Application Server が自動的に起動するプロセスを無効にしていた場合は、有効化します。

作業 6: 環境の更新

この作業には、新しいホスト名、ドメイン名、または IP アドレス用に環境を更新する手順が含まれます。実行する手順は、環境の構成方法により異なります。次を含むホストのホスト名または IP アドレスを変更した場合、それぞれの手順に従います。

- **Oracle Internet Directory のみ :**「構成 1: Oracle Internet Directory のみ」を参照してください。これは、Oracle Internet Directory があるホストにインストールされ、他の Identity Management コンポーネントが別のホストにインストールされている場合に、Oracle Internet Directory を含むホストが変更されたケースです。この場合、他の Identity Management コンポーネントとこの Identity Management を使用する中間層を更新する必要があります。
- **Oracle Internet Directory 以外の Identity Management コンポーネント :**「構成 2: OracleAS Single Sign-On、Oracle Delegated Administration Services および Oracle Directory Integration and Provisioning」を参照してください。これは、Oracle Internet Directory があるホストにインストールされ、他の Identity Management コンポーネントが別のホストにインストールされている場合に、他の Identity Management コンポーネントを含むホストが変更されたケースです。この場合、この Identity Management を使用する中間層を更新する必要があります。
- **Oracle Internet Directory と他の Identity Management コンポーネント :**「構成 3: Oracle Internet Directory、OracleAS Single Sign-On、Oracle Delegated Administration Services および Oracle Directory Integration and Provisioning」を参照してください。これは、Oracle Internet Directory と他の Identity Management コンポーネントが同じホストにインストールされているケースです。この場合、この Identity Management を使用する中間層を更新する必要があります。

環境の Oracle Internet Directory で LDAP ベースのレプリケーションが使用されており、Oracle Internet Directory が OracleAS Metadata Repository とは別のホストにある場合、Master（サプライヤ）または Replica（コンシューマ）の Oracle Internet Directory を含むホストのホスト名、ドメイン名、または IP アドレスを変更できます。詳細は、7-14 ページの「作業 7: LDAP ベースのレプリケーションが使用されている場合の Oracle Internet Directory の更新」を参照してください。

構成 1: Oracle Internet Directory のみ　これは、Oracle Internet Directory があるホストにインストールされ、他の Identity Management コンポーネントが別のホストにインストールされている場合に、Oracle Internet Directory を含むホストが変更されたケースです。次の手順に従います。

1. OracleAS Single Sign-On インストールで、Infrastructure のプロセスと Application Server Control コンソールを停止します。

   - UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/bin/emctl stop iasconsole
     ```

   - Windows の場合 :

     ```
     ORACLE_HOME¥opmn¥bin¥opmnctl stopall
     ORACLE_HOME¥bin¥emctl stop iasconsole
     ```

2. Oracle Internet Directory を使用するすべての OracleAS Infrastructure インスタンスで、`ias.properties` ファイルを更新します。これには、他の Identity Management インスタンス（OracleAS Single Sign-On、Oracle Delegated Administration Services および Oracle Directory Integration and Provisioning（構成されている場合））が含まれます。

   各 Oracle ホームで、次のファイルを更新します。

   ```
   (UNIX) ORACLE_HOME/config/ias.properties
   (Windows) ORACLE_HOME¥config¥ias.properties
   ```

   ファイルの `OIDhost` パラメータを、次のように新しいホスト名で更新します。

   ```
   OIDhost=newhost.us.oracle.com
   ```

3. Oracle Internet Directory を使用するすべての OracleAS Infrastructure インスタンスで、`ldap.ora` ファイルを更新します。これには、他の Identity Management インスタンス（OracleAS Single Sign-On、Oracle Delegated Administration Services および Oracle Directory Integration and Provisioning（構成されている場合））が含まれます。

   各 Oracle ホームで、次のファイルを編集します。

   ```
   (UNIX) ORACLE_HOME/ldap/admin/ldap.ora
   (Windows) ORACLE_HOME¥ldap¥admin¥ldap.ora
   ```

   ファイルの `DIRECTORY_SERVERS` パラメータを、新しい完全修飾ホスト名で更新します。

4. 他の Identity Management コンポーネントの Oracle ホームで、OPMN と Application Server Control コンソールを再起動します。

   - UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl start
     ORACLE_HOME/bin/emctl start iasconsole
     ```

   - Windows の場合 :

     ```
     ORACLE_HOME¥opmn¥bin¥opmnctl start
     ORACLE_HOME¥bin¥emctl start iasconsole
     ```

5. 中間層インスタンスの Oracle ホームで、OPMN と Application Server Control コンソールを再起動します。Application Server Control コンソールを起動するには、デフォルトの OC4J インスタンスを起動します。Application Server Control コンソールは、デフォルトの OC4J インスタンスのアプリケーションとして実行されるからです。

   - UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl start
     ORACLE_HOME/opmn/bin/opmnctl startproc process-type=home
     ```

- Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl start
ORACLE_HOME¥opmn¥bin¥opmnctl startproc process-type=home
```

6. 中間層インスタンスが SSO 認証用に有効化された場合は、再び有効化する必要があります。この Oracle Internet Directory インスタンスを使用する各中間層インスタンスについて、第 6.6 項の「作業 1: SSO 認証の有効化（オプション）」の手順を実行します。
7. 他の Identity Management コンポーネントと各中間層の Oracle ホームで、ID 管理の変更ウィザードを実行し、Oracle Internet Directory の新しい情報を指定します。たとえば、10.1.3 中間層の場合、次の手順を実行します。
   a. Application Server Control コンソールを使用して、中間層インスタンスの OC4J ホーム・ページにナビゲートします。
   b. 「**管理**」をクリックします。
   c. 表の「**タスク名**」列で「**セキュリティ**」が閉じている場合は、それを開きます。「ID 管理」行で、「**タスクに移動**」アイコンをクリックします。
   d. 「ID 管理」ページで、「**変更**」をクリックします。
   e. ウィザードの手順に従って、新しい Identity Management の情報を指定します。詳細は、第 6.6 項を参照してください。
   f. 操作が完了したら、OC4J インスタンスを再起動する必要があります。「確認」ページで「再起動」をクリックしないでください。かわりに、「クラスタ・トポロジ」ページにナビゲートし、OC4J インスタンスを選択してから、「**再起動**」をクリックします。

   ページに Internet Directory の新しいホストやポートが表示されても、この手順を実行する必要があります。リリース 2（10.1.2）の場合、Application Server Control コンソールに仮想ホスト名が表示されるのは、更新された `ias.properties` ファイルからそれを読み取っているためです。
8. OracleAS Certificate Authority がインストールされている場合、次の手順を実行します。
   a. OracleAS Certificate Authority が実行されているホストで OracleAS Certificate Authority、OC4J `oca` プロセス、および Oracle HTTP Server を停止します。たとえば、UNIX の場合、次のコマンドを実行します。

      ```
      ORACLE_HOME/oca/bin/ocactl stop
      ORACLE_HOME/opmn/bin/opmnctl stopproc process-type=oca
      ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=HTTP_Server
      ```

   b. 次のファイルを編集して、ファイル内にリストされているホスト名を変更します。

      ```
      (UNIX) ORACLE_HOME/oca/conf/oca.conf
      (Windows) ORACLE_HOME¥oca¥conf¥oca.conf
      ```

   c. OracleAS Single Sign-On と Oracle Internet Directory を再度関連付けます。たとえば UNIX では、次のように指定します。

      ```
      ORACLE_HOME/oca/bin/ocactl changesecurity -server_auth_port OcaSslPort
      ```

   d. Oracle HTTP Server、OC4J `oca` プロセスおよび OracleAS Certificate Authority を起動します。たとえば UNIX では、次のように指定します。

      ```
      ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=HTTP_Server
      ORACLE_HOME/opmn/bin/opmnctl startproc process-type=oca
      ORACLE_HOME/oca/bin/ocactl start
      ```

構成 2: OracleAS Single Sign-On、Oracle Delegated Administration Services および Oracle Directory Integration and Provisioning　これは、Oracle Internet Directory があるホストにインストールされ、他の Identity Management コンポーネントが別のホストにインストールされている場合に、他の Identity Management コンポーネントを含むホストが変更されたケースです。

各中間層インストールで、次の手順を実行します。

1. OPMN と Application Server Control コンソールを起動します。Application Server Control コンソールを起動するには、デフォルトの OC4J インスタンスを起動します。Application Server Control コンソールは、デフォルトの OC4J インスタンスのアプリケーションとして実行されるからです。

   - UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl start
     ORACLE_HOME/opmn/bin/opmnctl startproc process-type=home
     ```

   - Windows の場合 :

     ```
     ORACLE_HOME¥opmn¥bin¥opmnctl start
     ORACLE_HOME¥opmn¥bin¥opmnctl startproc process-type=home
     ```

2. 各中間層の Oracle ホームで、ID 管理の変更ウィザードを実行して Oracle Internet Directory の新しい情報を指定します。

   a. Application Server Control コンソールを使用して、中間層インスタンスの OC4J ホーム・ページにナビゲートします。

   b. 「**管理**」をクリックします。

   c. 表の「**タスク名**」列で「**セキュリティ**」が閉じている場合は、それを開きます。「ID 管理」行で、「**タスクに移動**」アイコンをクリックします。

   d. 「ID 管理」ページで、「**変更**」をクリックします。

   e. ウィザードの手順に従って、新しい Identity Management の情報を指定します。詳細は、第 6.6 項を参照してください。

   f. 操作が完了したら、OC4J インスタンスを再起動する必要があります。「確認」ページで「再起動」をクリックしないでください。かわりに、「クラスタ・トポロジ」ページにナビゲートし、OC4J インスタンスを選択してから、「**再起動**」をクリックします。

   ページに Internet Directory のホストやポートが表示されても、この手順を実行する必要があります。

3. 影響を受けるコンポーネントを再起動します。各 Oracle ホームで次のコマンドを実行します。

   – UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/opmn/bin/opmnctl startall
     ```

   – Windows の場合 :

     ```
     ORACLE_HOME¥opmn¥bin¥opmnctl stopall
     ORACLE_HOME¥opmn¥bin¥opmnctl startall
     ```

構成 3: Oracle Internet Directory、OracleAS Single Sign-On、Oracle Delegated Administration Services および Oracle Directory Integration and Provisioning

これは、Oracle Internet Directory と他の Identity Management コンポーネントが同じホストにインストールされている場合に、このホストが変更されたケースです。次の手順に従います。

各中間層インストールで、次の手順を実行します。

1. OPMN と Application Server Control コンソールを起動します。Application Server Control コンソールを起動するには、デフォルトの OC4J インスタンスを起動します。Application Server Control コンソールは、デフォルトの OC4J インスタンスのアプリケーションとして実行されるからです。

   - UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl start
     ORACLE_HOME/opmn/bin/opmnctl startproc process-type=home
     ```

   - Windows の場合 :

     ```
     ORACLE_HOME¥opmn¥bin¥opmnctl start
     ORACLE_HOME¥opmn¥bin¥opmnctl startproc process-type=home
     ```

2. 中間層インスタンスが SSO 認証用に有効化された場合は、再び有効化する必要があります。各中間層インスタンスについて、第 6.6 項の「作業 1: SSO 認証の有効化（オプション）」の手順を実行します。

3. 各中間層インストールで、ID 管理の変更ウィザードを実行します。

   a. Application Server Control コンソールを使用して、中間層インスタンスの OC4J ホーム・ページにナビゲートします。

   b. 「**管理**」をクリックします。

   c. 表の「**タスク名**」列で「**セキュリティ**」が閉じている場合は、それを開きます。「ID 管理」行で、「**タスクに移動**」アイコンをクリックします。

   d. 「ID 管理」ページで、「**変更**」をクリックします。

   e. ウィザードの手順に従って、新しい Identity Management の情報を指定します。詳細は、第 6.6 項を参照してください。

   f. 操作が完了したら、OC4J インスタンスを再起動する必要があります。「確認」ページで「再起動」をクリックしないでください。かわりに、「クラスタ・トポロジ」ページにナビゲートし、OC4J インスタンスを選択してから、「**再起動**」をクリックします。

4. 影響を受けるコンポーネントを再起動します。各 Oracle ホームで次のコマンドを実行します。

   – UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/opmn/bin/opmnctl startall
     ```

   – Windows の場合 :

     ```
     ORACLE_HOME¥opmn¥bin¥opmnctl stopall
     ORACLE_HOME¥opmn¥bin¥opmnctl startall
     ```

作業 7: LDAP ベースのレプリケーションが使用されている場合の Oracle Internet Directory の更新

環境の Oracle Internet Directory で LDAP ベースのレプリケーションが使用されており、Oracle Internet Directory が OracleAS Metadata Repository とは別のホストにある場合、Master（サプライヤ）または Replica（コンシューマ）の Oracle Internet Directory を含むホストのホスト名、ドメイン名、または IP アドレスを変更できます。

- 構成 A: Master の Oracle Internet Directory があるホストが変更されたケース
- 構成 B: Replica の Oracle Internet Directory のあるホストが変更されたケース

構成 A: Master の Oracle Internet Directory があるホストが変更されたケース

Master の Oracle Internet Directory を含むホストのホスト名、ドメイン名、または IP アドレスを変更した場合、次の手順を実行します。

1. Master の Oracle Internet Directory のレプリカ ID を取得します。

   ```
   ldapsearch -p master_port -h master_host -b "" -s base "objectclass=*"
   orclreplicaid
   ```

2. Master の Oracle Internet Directory のレプリカ・エントリに `orclreplicauri` と `orclreplicasecondaryuri` が存在する場合、Master と Replica の両方で、*orclreplicauri* と *orclreplicasecondaryuri* のいずれか、または両方を更新します。次の手順に従います。

   a. `mod.ldif` という名前のファイルを作成し、次の行を入力します。

      ```
      dn: orclreplicaid=master_replicaID,cn=replication configuration
      changetype:modify
      replace: orclreplicauri
      orclreplicauri: ldap://new_master_host:new_master_port/
      ```

      この例では、*master_replicaID* はステップ a で取得した ID で、*new_master_host* は Master の Oracle Internet Directory の新しいホスト名です。*new_master_port* は Master の Oracle Internet Directory のポート番号です。

   b. Master で次のコマンドを実行します。

      ```
      ldapmodify -p master_port -h master_host -f mod.ldif
      ```

   c. Replica で次のコマンドを実行します。

      ```
      ldapmodify -p replica_port -h replica_host -f mod.ldif
      ```

3. Replica でレプリケーション・サーバーを再起動します。

   ```
   oidctl server=oidrepld inst=inst_num connect=connect_string flags="-h
    replica_host -p replica_port -m false" stop
   oidctl server=oidrepld inst=inst_num connect=connect_string flags="-h
    replica_host -p replica_port -m false" start
   ```

   この例では、*replica_host* は Replica の Oracle Internet Directory のホスト名で、*replica_port* は Replica の Oracle Internet Directory のポートです。

構成 B: Replica の Oracle Internet Directory のあるホストが変更されたケース

Replica の Oracle Internet Directory を含むホストのホスト名、ドメイン名、または IP アドレスを変更した場合、次の手順を実行します。

1. Replica の Oracle Internet Directory のレプリカ ID を取得します。

   ```
   ldapsearch -p replica_port -h replica_host -b "" -s base "objectclass=*"
   orclreplicaid
   ```

2. Replica の Oracle Internet Directory のレプリカ・エントリに `orclreplicauri` と `orclreplicasecondaryuri` が存在する場合、Master と Replica の両方で、*orclreplicauri* と *orclreplicasecondaryuri* のいずれか、または両方を更新します。次の手順に従います。

   a. `mod.ldif` という名前のファイルを作成し、次の行を入力します。

      ```
      dn: orclreplicaid=replica_replicaID,cn=replication configuration
      changetype:modify
      replace: orclreplicauri
      orclreplicauri: ldap://new_replica_host:new_replica_port/
      ```

      この例では、*replica_replicaID* はステップ a で取得した ID で、*new_replica_host* は Replica の Oracle Internet Directory の新しいホスト名です。*new_replica_port* は Replica の Oracle Internet Directory のポート番号です。

b. Master で次のコマンドを実行します。

```
ldapmodify -p master_port -h master_host -f mod.ldif
```

c. Replica で次のコマンドを実行します。

```
ldapmodify -p replica_port -h replica_host -f mod.ldif
```

3. Replica でレプリケーション・サーバーを再起動します。

```
oidctl server=oidrepld inst=inst_num connect=connect_string flags="-h
 new_replica_host -p new_replica_port -m false" stop
oidctl server=oidrepld inst=inst_num connect=connect_string flags="-h
 new_replica_host -p new_replica_port -m false" start
```

この例では、*new_replica_host* は Replica の Oracle Internet Directory の新しいホスト名で、*new_replica_port* は Replica の Oracle Internet Directory のポートです。

7.2.4 Metadata Repository を含む 10.1.4 または 10.1.2 の Infrastructure の IP アドレスの変更

この項では、Infrastructure インストール・タイプが次のいずれかであるホストの IP アドレスを変更する方法について説明します。

- Metadata Repository のみ
- Identity Management および Metadata Repository

次の各項では、この手順について説明します。

設定前の確認事項

手順を開始する前に、次の項目を確認してください。

- 開始する前に、元の IP アドレスを書き留めておいてください。手順の実行中に、この入力が求められます。
- この手順を開始する前に、環境のバックアップを行うことをお薦めします。詳細は、第 V 部「バックアップとリカバリ」を参照してください。

作業 1: 中間層インスタンスの停止

Infrastructure インストールを使用するすべての中間層インスタンスを停止します。別のホストにあるインスタンスも停止します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopall
```

作業 2: ホストの準備

すべてのプロセスを停止して、ホストの変更準備をします。

1. ORACLE_HOME および ORACLE_SID 環境変数を設定します。

2. Infrastructure を停止します。

a. すべての Application Server Control コンソールおよびすべてのプロセスを停止します。

   – UNIX の場合 :

   ```
   ORACLE_HOME/bin/emctl stop iasconsole
   ORACLE_HOME/opmn/bin/opmnctl stopall
   ```

   – Windows の場合 :

   ```
   ORACLE_HOME¥bin¥emctl stop iasconsole
   ORACLE_HOME¥opmn¥bin¥opmnctl stopall
   ```

b. ディレクトリを Oracle ホームの bin サブディレクトリに変更します。次に、リスナーとデータベースを停止します。

   ```
   lsnrctl stop

   sqlplus /nolog
   SQL> connect SYS as SYSDBA
   SQL> shutdown
   SQL> quit
   ```

3. すべての Oracle Application Server プロセスが停止したことを確認します。
4. /etc/init.d スクリプトなどの自動スタートアップ・スクリプトを設定している場合は、それらを無効にして、ホストの再起動後に Oracle Application Server プロセスが自動的に起動されないようにします。

作業 3: IP アドレスの変更

オペレーティング・システムの IP アドレスを変更して再起動し、ホストがネットワーク上で正しく機能することを確認します。ここでの手順を実行する方法の詳細は、該当するオペレーティング・システムのドキュメントを参照してください。

1. オペレーティング・システムの IP アドレスを適切に変更します。
2. オペレーティング・システムで再起動が必要な場合は、ホストを再起動します。
3. ネットワークの別のホストから、このホストに ping が実行できることを確認します。必ず新しい IP アドレスを使用して ping を実行し、すべてが正しく解決されていることを確認してください。

作業 4: Infrastructure の更新

ホストの Infrastructure の IP アドレスを更新します。

1. Infrastructure をインストールしたユーザーとしてホストにログインします。
2. ORACLE_HOME および ORACLE_SID 環境変数を設定します。ORACLE_HOME 変数を指定するときは、スラッシュ（UNIX）や円記号（Windows）を最後に使用しないでください。
3. UNIX システムの場合は、LD_LIBRARY_PATH、LD_LIBRARY_PATH_64、LIB_PATH または SHLIB_PATH 環境変数を、表 1-1 に示されている適切な値に設定します。実際に設定が必要な環境変数および値は、UNIX オペレーティング・システムのタイプによって異なります。
4. ディレクトリを Oracle ホームの bin サブディレクトリに変更します。次に、データベースとリスナーを起動します。

   ```
   sqlplus /nolog
   SQL> connect SYS as SYSDBA
   SQL> startup
   SQL> quit

   lsnrctl start
   ```

5. OPMN を起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl start
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl start
```

6. Oracle Internet Directory を起動します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=OID process-type=OID
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl startproc ias-component=OID process-type=OID
```

7. Infrastructure の Oracle ホームで次のコマンドを実行します。

   - UNIX の場合 :

```
cd ORACLE_HOME/chgip/scripts
./chgiphost.sh -infra
```

   - Windows の場合 :

```
cd ORACLE_HOME¥chgip¥scripts
cmd /c chgiphost.bat -infra
```

   chgiphost コマンドでは、古い IP アドレスと新しい IP アドレスの入力を求めるプロンプトが表示されます。

8. 次のディレクトリ内にあるファイルで、エラーが発生していないかチェックして、ツールが正常に実行されたことを確認します。

```
(UNIX) ORACLE_HOME/chgip/log
(Windows) ORACLE_HOME¥chgip¥log
```

作業 5: 環境の再起動

Infrastructure の残りのコンポーネントを起動し、それを使用する中間層インスタンスを起動します。

1. Infrastructure を起動します。

   - UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl startall
ORACLE_HOME/bin/emctl start iasconsole
```

   - Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl startall
ORACLE_HOME¥bin¥emctl start iasconsole
```

2. 中間層インスタンスが Infrastructure と同一のホストに存在する場合は、中間層インスタンスを再起動する前に、中間層インスタンスに対して chgiphost コマンドを実行する必要があります。

3. 中間層インスタンスを起動します。

   - UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl startall
```

   - Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl startall
```

4. この手順の開始前に、Oracle Application Server が自動的に起動するプロセスを無効にしていた場合は、有効化します。

7.2.5 ホスト名またはドメイン名の変更に関する特殊なトピック

この項では、Oracle Application Server ホストのホスト名またはドメイン名を変更する際に該当する特殊なトピックについて説明します。この項のトピックは次のとおりです。

7.2.5.1 chgiphost のログ・レベルの設定

デフォルトでは、chgiphost コマンドのコンソールのログ・レベルは SEVERE です。この設定では、chgiphost の実行時に出力される情報はクリティカルなもののみになります。これ以外の進捗情報を表示する必要がある場合は、次の手順を実行して、コンソールのログ・レベルを CONFIG に設定します。

1. 次のファイルを編集します。

   (UNIX) *ORACLE_HOME*/chgip/config/chgip.log.properties
   (Windows) *ORACLE_HOME*¥chgip¥config¥chgip.log.properties

2. java.util.logging.ConsoleHandler.level パラメータを CONFIG に変更します。

   java.util.logging.ConsoleHandler.level = CONFIG

7.2.5.2 chgiphost コマンドのカスタマイズ

デフォルトでは、chgiphost コマンドを実行すると、Oracle ホーム内の主要な構成ファイルが更新され、新しいホスト名が使用されるようになります。現行のインストールに次のいずれかの項目が該当する場合は、chgiphost コマンドの動作のカスタマイズを検討する必要があります。

- ホスト名が記述された構成ファイルを追加作成しており、chgiphost コマンドを使用してそれらのファイルも更新したい。

  これらのファイルを更新するには、次のファイルにフルパス名を追加してから、chgiphost を実行します。

  (UNIX) *ORACLE_HOME*/chgip/config/hostname.lst
  (Windows) *ORACLE_HOME*¥chgip¥config¥hostname.lst

- 元のホスト名が非常に短い（1 文字か 2 文字）、または元のホスト名と同じ文字列が構成ファイル内に記述されている可能性がある。

  chgiphost を実行する前に、hostname.lst に一覧表示されている個々のファイルを調べて、元のホスト名と同じ文字列がこれらのファイルの設定に出現するかどうかを判定します。一致する文字列があった場合は、chgiphost の実行後に、該当する設定を訂正します。

- Oracle ホームのフルパスにホスト名が含まれる。

  この場合は、chgiphost コマンドを実行しても、構成ファイルを適切に更新できません。この問題を回避するには、FileFixer という Java ユーティリティを使用します。FileFixer では、正規表現と照合することでファイル内の特定のテキスト文字列が検索され、それらが新しい値に更新されます。FileFixer のパターン検索は行単位で実行されることに注意してください。複数行にわたるパターンは照合できません。

  FileFixer を使用する手順は次のとおりです。

  1. 次のファイルのコピーを作成します。

     (UNIX) *ORACLE_HOME*/chgip/config/hostname_short_sample.lst.xml
     (Windows) *ORACLE_HOME*¥chgip¥config¥hostname_short_sample.lst.xml

2. ファイルのコピーを編集して、元のホスト名と新しいホスト名に対して必要な正規表現照合を指定します。このファイルには、指定方法の例が用意されています。
3. chgiphost コマンドの実行時に、ファイルを次のように指定します。

   ```
   ./chgiphost option -hostnameShortXml full_path_to_your_xml_file
   ```

   たとえば、/mydir/my_sample.lst.xml と命名したファイルを使用して UNIX 上で中間層インストールを更新する場合は、chgiphost を次のように実行します。

   ```
   ./chgiphost -mid -hostnameShortXml /mydir/my_sample.lst.xml
   ```

7.2.5.3 Windows 2000 から Windows 2003 へのアップグレード後のホスト名変更

Windows 2000 から Windows 2003 にアップグレードすると、ホスト名に含まれる小文字が大文字に変更される場合があります。たとえば、アップグレード前のホスト名が myhost の場合、MYHOST に変更される場合があります。この現象が発生すると、Oracle Application Server の一部のプロセスが正常に機能しないおそれがあります。

この問題を解決するために、chgiphost コマンドを使用して Oracle Application Server を実行する必要はありません。小文字のホスト名を含むエントリを、次のホスト・ファイルに追加するだけです。

```
OS_path¥system32¥drivers¥etc¥hosts
```

たとえば、アップグレード前の完全修飾ホスト名が myhost.mydomain で、IP アドレスが 1.2.3.4 の場合は、次の行を追加します。

```
1.2.3.4 myhost.mydomain myhost
```

7.2.5.4 ホスト名変更時のエラーからのリカバリ

この項では、chgiphost コマンドの使用時に発生する一般的なエラーからのリカバリ方法について説明します。この項の項目は次のとおりです。

- ケース 1: 変更後の名前の指定間違い
- ケース 2: chgiphost の実行時に発生するエラー

ケース 1: 変更後の名前の指定間違い

chgiphost コマンドの実行時に、変更後の名前を間違って指定したとします。この場合は、chgiphost をもう一度実行することによってエラーを修正できます。次に具体的な例を示します。

現在の変更前のホスト名を loire985、間違って指定した変更後のホスト名を mqa985、正しい変更後のホスト名を sqb985 とします。最初に、変更前のホスト名を loire985、変更後のホスト名を mqa985 として chgiphost を実行しました。

このエラーからリカバリする手順は次のとおりです。

1. 変更前のホスト名を mqa985、変更後のホスト名を sqb985 として chgiphost を実行します。
2. 変更前のホスト名を loire985、変更後のホスト名を sqb985 としてもう一度 chgiphost を実行します。

ケース 2: chgiphost の実行時に発生するエラー

たとえば、Oracle Internet Directory のパスワードを間違えて入力すると、エラーが発生します。この場合は、opmnctl stopall コマンドを使用してインスタンスのすべてのプロセスを停止してから、前と同じ変更前および変更後のホスト名を使用して chgiphost をもう一度実行し、プロンプトが表示されたときに確実に正しいパスワードを指定します。

chgiphost の実行時にエラーが発生する場合は、そのエラーを修正してからもう一度 chgiphost を実行する必要があります。

7.3 ネットワーク接続のオン / オフの切替え

この項では、Oracle Application Server ホストがネットワークに接続された状態と接続されていない状態を切り替える方法について説明します。前提および制限は次のとおりです。

- ホストには、Infrastructure を使用しないインスタンスが含まれている必要があります。または、中間層インスタンスおよび Infrastructure が同じホスト内にある必要があります。
- DHCP は、ループバック・モードで使用する必要があります。詳細は、Oracle Application Server のインストレーション・ガイドを参照してください。
- IP アドレスの変更のみがサポートされています。ホスト名は変更できません。
- DHCP モードでは、デフォルトのホスト名（`localhost.localdomain`）は使用しないでください。標準のホスト名を使用するようにホストを設定します。ループバック IP アドレスからそのホスト名に解決する必要があります。
- ネットワークに接続されていない状態のインストール（DHCP または静的 IP）では、ループバック・アダプタが常に必要です。詳細は、Oracle Application Server のインストレーション・ガイドを参照してください。

7.3.1 ネットワーク接続のオフからオンへの変更（静的 IP アドレス）

この手順では、標準のホスト名（`localhost` ではない）を使用する、ネットワークに接続されていないホストに Oracle Application Server をインストールした後で、ネットワークに接続して静的 IP アドレスを使用する場合を想定しています。IP アドレスは、デフォルトのループバック IP または任意の標準的な IP アドレスです。

ネットワークに接続された状態に変更するには、ホストをネットワークに接続します。Oracle Application Server の更新は必要ありません。

7.3.2 ネットワーク接続のオフからオンへの変更（DHCP）

この手順では、標準のホスト名（`localhost` ではない）を使用する、ネットワークに接続されていないホストにインストールを行った後で、ネットワークに接続して DHCP を使用する場合を想定しています。ホストの IP アドレスには、任意の静的 IP アドレスまたはループバック IP アドレスを使用できます。この IP アドレスは、そのホスト名に対して設定されている必要があります。

ネットワークに接続をオンにする手順は次のとおりです。

1. DHCP を使用してホストをネットワークに接続します。
2. ループバック IP アドレスに対してのみホスト名を構成します。

7.3.3 ネットワーク接続のオンからオフへの変更（静的 IP アドレス）

この手順は、ホストがネットワークに接続され、静的 IP アドレスを使用している状態を、ネットワークに接続されていない状態に変更する場合に実行します。

1. `/etc/hosts` ファイルを構成して、IP アドレスとホスト名がローカルで解決できるようにします。
2. ホストを、ネットワークから切断します。
3. ホスト名または IP アドレスを変更する必要はありません。
4. インスタンスがクラスタの一部として構成された場合、インスタンスを起動する前に、クラスタから削除する必要があります。たとえば、動的検出を使用していた場合にインスタンスをクラスタから削除するには、次のコマンドを使用します。
   - UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl config topology delete discover
     ORACLE_HOME/opmn/bin/opmnctl reload
     ```

- Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl config topology delete discover
ORACLE_HOME¥opmn¥bin¥opmnctl reload
```

クラスタから削除する方法は、クラスタの構成方法によって異なります。詳細は、『Oracle Containers for J2EE 構成および管理ガイド』の「クラスタの構成と管理」を参照してください。

7.3.4 ネットワーク接続のオンからオフへの変更（DHCP）

この手順は、ホストがネットワークに接続され、ループバック・モードの DHCP を使用している状態を、ネットワークに接続されていない状態に変更する場合に実行します。

1. ホストを、ネットワークから切断します。
2. ホスト名または IP アドレスを変更する必要はありません。
3. インスタンスがクラスタの一部として構成された場合、インスタンスを起動する前に、クラスタから削除する必要があります。たとえば、動的検出を使用していた場合にインスタンスをクラスタから削除するには、次のコマンドを使用します。

   - UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl config topology delete discover
ORACLE_HOME/opmn/bin/opmnctl reload
```

   - Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl config topology delete discover
ORACLE_HOME¥opmn¥bin¥opmnctl config reload
```

   クラスタから削除する方法は、クラスタの構成方法によって異なります。詳細は、『Oracle Containers for J2EE 構成および管理ガイド』の「クラスタの構成と管理」を参照してください。

7.4 静的 IP アドレスと DHCP の切替え

この項では、静的 IP アドレスと DHCP を切り替える方法を説明します。前提および制限は次のとおりです。

- ホストには、Infrastructure を使用しないインスタンスが含まれている必要があります。または、中間層インスタンスおよび Infrastructure が同じホスト内にある必要があります。
- DHCP は、ループバック・モードで使用する必要があります。詳細は、Oracle Application Server のインストレーション・ガイドを参照してください。
- IP アドレスの変更のみがサポートされています。ホスト名は変更できません。
- DHCP モードでは、デフォルトのホスト名（`localhost.localdomain`）は使用しないでください。標準のホスト名を使用するようにホストを設定します。ループバック IP アドレスからそのホスト名に解決する必要があります。

7.4.1 静的 IP アドレスから DHCP への切替え

ホストの設定を静的 IP アドレスから DHCP に変更する手順は次のとおりです。

1. DHCP に変更する前に、ホストを設定してホスト名とループバック IP アドレスを関連付けます。
2. ホストを DHCP に変更します。Oracle Application Server を更新する必要はありません。

7.4.2 DHCP から静的 IP アドレスへの切替え

ホストの設定を DHCP から静的 IP アドレスに変更する手順は次のとおりです。

1. ホストが静的 IP アドレスを使用するように変更します。
2. Oracle Application Server を更新する必要はありません。

8

Infrastructure サービスの変更

この章では、中間層インスタンスによって使用される Infrastructure サービスを変更する手順について説明します。

この章の項目は次のとおりです。

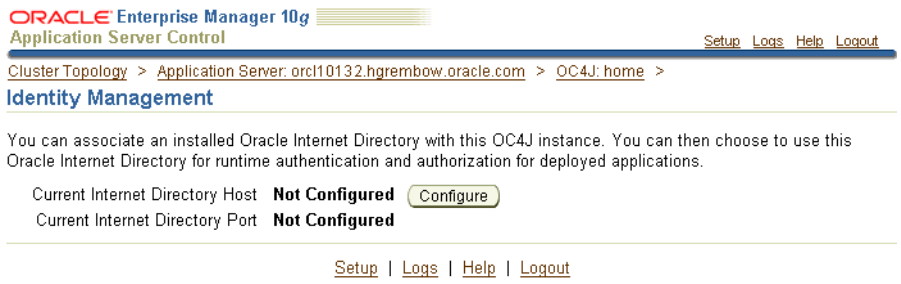
8.1 Identity Management サービスの変更手順の概要

このリリースでは、第 6.6 項で説明されているとおり、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスをリリース 10.1.4 またはリリース 2（10.1.2）の Identity Management サービスに関連付けることができます。

中間層インスタンスを Identity Management サービスに関連付けた後、中間層インスタンスによって使用される Identity Management サービスを変更しなければならないことがあります。たとえば、ID 管理サービスを別のホストで使用しなければならない場合などです。

Identity Management サービスは、図 8-1 に示す Application Server Control コンソールの「ID 管理」ページで変更できます。

図 8-1 Application Server Control コンソールの「ID 管理」ページ



Identity Management サービスを変更しなければならないのは、次のいずれかを変更する場合です。

- Identity Management インストールの HTTP OracleAS Single Sign-On ポート番号
- Oracle Internet Directory の非 SSL または SSL のポート番号
- Oracle Internet Directory のモード（デュアル・モードまたは SSL）
- Identity Management がインストールされているホスト

Oracle Internet Directory で匿名バインドが無効になっている場合、構成を変更する前に有効にする必要があります。詳細は、第 6.7 項「匿名バインドの有効化と無効化」を参照してください。

単にウィザードを使用するだけでは、特定の Infrastructure サービスを別の Infrastructure サービスに変更することはできません。新しい Infrastructure サービスを作成および準備するには、まず手動の作業を実行する必要があります。この章では、Infrastructure サービスの変更に対応する次の手順について説明します。

- Oracle Internet Directory のデュアル・モードから SSL モードへの変更

  Oracle Internet Directory のモードを非 SSL から SSL に変更する場合は、この手順を実行します。モードの変更とともに、中間層インスタンスが新しいモード情報を使用するように変更する必要があるため、Infrastructure サービスの変更が必要になります。

- 新しいホストへの 10.1.4 または 10.1.2 Identity Management の移動

  インストールされた Identity Management とそれに関連付けられた Metadata Repository を新しいホストに移動する場合は、この手順を実行します。移動後は、中間層インスタンスが Identity Management の新しいホスト情報を使用するように変更する必要があるため、Infrastructure サービスの変更が必要になります。

ポートの変更方法の詳細は、次の各項を参照してください。

- Identity Management インストールで Oracle Internet Directory の非 SSL または SSL ポートを変更する場合は、第 4.4.2 項「10.1.4 または 10.1.2 の Oracle Internet Directory ポートの変更」を参照してください。
- Identity Management インストールで Oracle HTTP Server の非 SSL または SSL リスナー・ポートを変更する（これにより OracleAS Single Sign-On ポートが変更される）場合は、第 4.4.3 項「10.1.4 または 10.1.2 の Identity Management インストールの HTTP Server ポートの変更」を参照してください。

8.2 Oracle Internet Directory のデュアル・モードから SSL モードへの変更

Identity Management をインストールすると、Oracle Internet Directory のモードを選択するよう求められます。デフォルトのモードはデュアル・モードです。デュアル・モードでは、一部のコンポーネントは非 SSL 接続を使用して Oracle Internet Directory にアクセスできます。インストール時には、SSL モードを選択することができます。SSL モードでは、ディレクトリに接続する際にすべてのコンポーネントが SSL を使用する必要があります。

インストール時に SSL モードを選択せず、インストール後に SSL に変更する場合は、この項の手順を実行します。手順には、Oracle Internet Directory のモードの変更と、中間層インスタンスで新しいモードを使用するための変更処理が含まれます。

8.2.1 Application Server Control のセキュリティ・プロバイダの制限

この手順を実行する前に、Application Server Control でファイルベースのセキュリティ・プロバイダが使用されていることを確認する必要があります。そうでない場合は、Oracle Internet Directory モードの変更後に追加の手順を実行します。

セキュリティ・プロバイダのタイプを確認する手順は次のとおりです。

1. Application Server Control コンソールで、OC4J ホーム・ページにナビゲートします。
2. 「**設定**」をクリックします。
3. 「設定」ページで、「**セキュリティ・プロバイダ**」を選択します。

   「セキュリティ・プロバイダ」ページに、使用されているセキュリティ・プロバイダのタイプが表示されます。
4. ファイルベースではないセキュリティ・プロバイダを変更する場合、「**セキュリティ・プロバイダの変更**」をクリックします。次に、「**ファイルベースのセキュリティ・プロバイダ**」を選択して、XML ファイルの場所を指定します。

セキュリティ・プロバイダが Oracle Internet Directory であり、この手順を実行するまで変更しない場合は、「作業 3: jazn.xml の変更（Oracle Internet Directory セキュリティ・プロバイダにのみ該当）」の手順を実行する必要があります。

8.2.2 手順

Oracle Internet Directory を SSL モードに変更するには、次の作業を実行します。

- 作業 1: 中間層プロセスの停止後の Application Server Control コンソールの起動
- 作業 2: Oracle Internet Directory のモードの変更
- 作業 3: jazn.xml の変更（Oracle Internet Directory セキュリティ・プロバイダにのみ該当）
- 作業 4: 中間層インスタンスが SSL モードを使用するように変更

作業 1: 中間層プロセスの停止後の Application Server Control コンソールの起動

Oracle Internet Directory を使用するすべての中間層インスタンスで、次の手順を実行します。

1. 次のコマンドを使用して、すべての中間層インスタンスを停止します。

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopall
   (Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopall
   ```

2. この後の手順で Application Server Control コンソールを使用するため、次のコマンドを使用して OPMN と Application Server Control を起動します。Application Server Control コンソールを起動するには、デフォルトの OC4J インスタンスを起動します。Application Server Control コンソールは、デフォルトの OC4J インスタンスのアプリケーションとして実行されるからです。

   - UNIX の場合 :

     ```
     ORACLE_HOME/opmn/bin/opmnctl start
     ORACLE_HOME/opmn/bin/opmnctl startproc process-type=home
     ```

   - Windows の場合 :

     ```
     ORACLE_HOME¥opmn¥bin¥opmnctl start
     ORACLE_HOME¥opmn¥bin¥opmnctl startproc process-type=home
     ```

作業 2: Oracle Internet Directory のモードの変更

この作業は、Oracle Internet Directory を含むリリース 2（10.1.2）Infrastructure に対して実行します。

1. `mod.ldif` という名前のファイルを作成し、次の行を入力します。

   ```
   dn:cn=configset0,cn=osdldapd,cn=subconfigsubentry
   changetype:modify
   replace:orclsslenable
   orclsslenable:1
   ```

2. 次のコマンドを実行します。

   ```
   ldapmodify -D "cn=orcladmin" -w orcladmin_passwd -p oid_port -v -f mod.ldif
   ```

   この例で、*oid_port* は、非 SSL の Oracle Internet Directory ポートです。この値は、*ORACLE_HOME*/config/ias.properties 内の OIDport で確認できます。

   OracleAS Cold Failover Cluster を使用している場合は、次のコマンドを使用する必要があります。

   ```
   ldapmodify -D cn=orcladmin -w orcladmin_passwd -h virtual_hostname
   -p oid_port -v -f mod.ldif
   ```

   この例で、*virtual_hostname* は、OracleAS Cold Failover Cluster の仮想ホスト名です。

3. Oracle Internet Directory を含むインスタンス全体を停止します。

   - UNIX の場合 :

     ```
     ORACLE_HOME/bin/emctl stop iasconsole
     ORACLE_HOME/opmn/bin/opmnctl stopall
     ```

   - Windows の場合 :

     ```
     ORACLE_HOME¥bin¥emctl stop iasconsole
     ORACLE_HOME¥opmn¥bin¥opmnctl stopall
     ```

4. 次のファイルを編集します。

   (UNIX) *ORACLE_HOME*/ldap/admin/ldap.ora
   (Windows) *ORACLE_HOME*¥ldap¥admin¥ldap.ora

   a. 次の行から非 SSL ポート番号を削除します。

      DIRECTORY_SERVERS=(*myhost.myco.com*:*nonsslport*:*sslport*)

      この行は次のようになります。

      DIRECTORY_SERVERS=(*myhost.myco.com*::*sslport*)

   b. ファイルを保存して閉じます。

5. OracleAS RepCA を使用して OracleAS Metadata Repository を作成している場合、次の手順を実行します。

   a. ldap.ora ファイルを Identity Management の Oracle ホームから OracleAS Metadata Repository の Oracle ホームへコピーします。たとえば、リリース 2（10.1.2）の場合、場所は次のとおりです。

      (UNIX) *ORACLE_HOME*/ldap/admin
      (Windows) *ORACLE_HOME*¥ldap¥admin

   b. OracleAS Metadata Repository の Oracle ホーム内の次の場所にある sqlnet.ora ファイルを編集します。

      (UNIX) *ORACLE_HOME*/network/admin
      (Windows) *ORACLE_HOME*¥network¥admin

      次の例に示すように、LDAP を NAMES.DIRECTORY_PATH エントリに追加します。

      NAMES.DIRECTORY_PATH= (LDAP, TNSNAMES, ONAMES, HOSTNAME)

6. 次のファイルを編集します。

   (UNIX) *ORACLE_HOME*/config/ias.properties
   (Windows) *ORACLE_HOME*¥config¥ias.properties

   a. SSLOnly パラメータを次のように変更します。

      SSLOnly=true

   b. ファイルを保存して閉じます。

7. Oracle Internet Directory を含むインスタンス全体を再起動します。

   - UNIX の場合 :

     ORACLE_HOME/opmn/bin/opmnctl stopall
     ORACLE_HOME/bin/emctl stop iasconsole
     ORACLE_HOME/opmn/bin/opmnctl startall
     ORACLE_HOME/bin/emctl start iasconsole

   - Windows の場合 :

     ORACLE_HOME¥opmn¥bin¥opmnctl stopall
     ORACLE_HOME¥bin¥emctl stop iasconsole
     ORACLE_HOME¥opmn¥bin¥opmnctl startall
     ORACLE_HOME¥bin¥emctl start iasconsole

作業 3: jazn.xml の変更（Oracle Internet Directory セキュリティ・プロバイダにのみ該当）

Application Server Control のセキュリティ・プロバイダが Oracle Internet Directory である場合は、アクティブな ascontrol アプリケーションを含むインスタンスの jazn.xml を変更しないと、その中間層インスタンスで SSL モードを使用するように変更できません（Application Server Control で使用するセキュリティ・プロバイダのタイプを決定する方法は、第 8.2.1 項を参照）。

アクティブな ascontrol アプリケーションを含むインスタンスで変更する手順は次のとおりです。

1. 次のファイルを編集します。

   (Unix) *ORACLE_HOME*/j2ee/*OC4J_InstanceName*/config/jazn.xml
   (Windows) *ORACLE_HOME*¥j2ee¥*OC4J_InstanceName*¥config¥jazn.xml

2. location 属性を変更し、SSL ポートを使用します。次に例を示します。

   ```
   location="ldap://myoid.us.oracle.com:636"
   ```

3. ldap.protocol のプロパティ値を変更し、ssl を指定します。次に例を示します。

   ```
   <property name=:ldap.protocol" value="ssl"/>
   ```

4. ファイルを保存して閉じます。

作業 4: 中間層インスタンスが SSL モードを使用するように変更

各中間層インスタンスで、ID 管理の変更ウィザードを起動し、インスタンスを再起動します。

1. Application Server Control コンソールを使用して、中間層インスタンスの OC4J ホーム・ページにナビゲートします。
2. 「**管理**」をクリックします。
3. 表の「タスク名」列で「**セキュリティ**」が閉じている場合は、それを開きます。「ID 管理」行で、「**タスクに移動**」アイコンをクリックします。
4. 「ID 管理」ページで、「**変更**」をクリックします。
5. 「ID 管理の変更」ページで次のように入力します。
   - **Oracle Internet Directory ホスト :** Oracle Internet Directory ホストの完全修飾名。
   - **Oracle Internet Directory ユーザー :** cn=orcladmin または iASAdmins グループのユーザーの識別名。
   - **パスワード :** そのユーザーのパスワード。

     このパスワードは、Oracle Internet Directory で作成した oc4jadmin ユーザーのデフォルト・パスワードとして使用されます。
   - **Internet Directory へ SSL 接続のみを使用 :** このオプションを選択します。

     「**Oracle Internet Directory SSL ポート**」フィールドに、Oracle Internet Directory の SSL ポート番号を入力します。

   「**OK**」をクリックします。
6. 操作が完了したら、OC4J インスタンスを再起動する必要があります。「確認」ページで「再起動」をクリックしないでください。かわりに、「クラスタ・トポロジ」ページにナビゲートし、OC4J インスタンスを選択してから、「**再起動**」をクリックします。

注意 : これで非 SSL の Oracle Internet Directory ポートが無効になったため、LDAP コマンドライン・ユーティリティ（ldapsearch、ldapmodify、ldapaddmt など）を使用して SSL ポートに接続するときは、「-U 1」オプションを指定する必要があります。

8.3 新しいホストへの 10.1.4 または 10.1.2 Identity Management の移動

第 6.6 項で説明されているとおり、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスをリリース 10.1.4 またはリリース 2（10.1.2）の Identity Management サービスに関連付けた後、Identity Management を新しいホストに移動する場合は、この項の手順を実行します。

この手順では、元の Identity Management のレプリカ（またはコピー）と、そのレプリカの新しい Metadata Repository を別のホストに作成し、中間層インスタンスが新しい Identity Management を使用するように変更します。

8.3.1 この手順の使用例

この手順の使用例は次のとおりです。

- 既存のリリース 10.1.4 またはリリース 2（10.1.2）の Identity Management とそれに関連付けられた Metadata Repository があり、これを 1 つまたは複数の 10*g* リリース 3（10.1.3.2.0）中間層インスタンスで使用しています。この組織では、現在の Identity Management ホストを新しいシステムに交換しようと考えています。この手順を実行すると、Identity Management のレプリカと、そのレプリカの Metadata Repository を作成し、中間層インスタンスが新しい Identity Management を使用するように変更できます。その後は、元のホストを廃棄してもかまいません。
- リリース 10.1.4 またはリリース 2（10.1.2）の Identity Management 用のフェイルオーバー環境を作成したいと考えています。この手順を実行すると、現在の Identity Management のレプリカと、そのレプリカの Metadata Repository を作成することができます。このレプリカは、元の Identity Management と同期を保った状態で実行することができます。元の Metadata Repository のデータは、定期的にエクスポートして保存しておくことができます。元の Identity Management を失った場合には、保存しておいたデータを新しい Metadata Repository にインポートし、10*g* リリース 3（10.1.3.2.0）の中間層インスタンスが新しい Identity Management を使用するように変更できます。詳細は、第 8.3.4 項「この手順を使用してフェイルオーバーを実施する方法」を参照してください。

8.3.2 前提と制限

- 元のインストールと新しいインストールで、Identity Management と Metadata Repository は、同じ Oracle ホームを使用することも、別の Oracle ホーム（同一または異なるホスト上）を使用することもできます。異なる Oracle ホームを使用する場合は、それぞれの Oracle ホームで操作を実行します。
- 元のインストールおよび新しいインストールで、Identity Management のコンポーネント（OracleAS Single Sign-On、Oracle Internet Directory、Delegated Administration Services および Directory Integration and Provisioning）は、同じ Oracle ホームを使用することも、別の Oracle ホーム（同一または異なるホスト上）を使用することもできます。異なる Oracle ホームを使用する場合は、それぞれの Oracle ホームで操作を実行します。
- この手順では、OracleAS Certificate Authority については考慮されていません。

関連項目： Identity Management サービスを変更する際に OracleAS Certificate Authority を更新する方法は、『Oracle Application Server Certificate Authority 管理者ガイド』を参照してください。

8.3.3 新しいホストに Identity Management を移動する手順

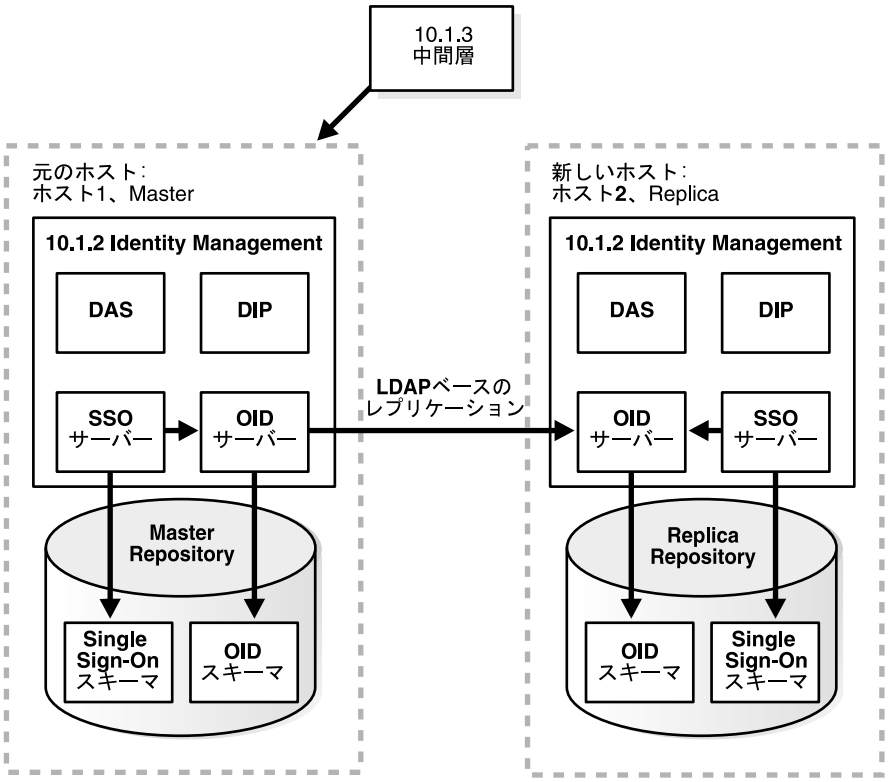
この項では、リリース 10.1.4 またはリリース 2（10.1.2）の Identity Management を新しいホストに移動する方法について説明します。

その手順の概要は次のとおりです。

1. 1 つまたは複数の中間層インスタンスによって使用される元のリリース 10.1.4 またはリリース 2（10.1.2）の Identity Management（Master）があります。この Identity Management には Metadata Repository があります。ここで、新しい Identity Management（Replica）をインストールおよび設定します。この Identity Management には固有の Metadata Repository があります。新しい Identity Management の Oracle Internet Directory は、元の Oracle Internet Directory の LDAP ベースのレプリカです。元の Oracle Internet Directory から新しい Oracle Internet Directory へのレプリケーションは常時行われます。

   図 8-2 に、リリース 2（10.1.2）の Identity Management を使用する設定を示します。

図 8-2 元のホスト（Master）と新しいホスト（Replica）



   関連項目：「作業 1: 新しい Identity Management と Metadata Repository のインストールおよび設定」

2. 元の Metadata Repository（Master）から新しい Metadata Repository（Replica）へ OracleAS Single Sign-On および Directory Integration and Provisioning データを移行します。

   関連項目：「作業 2: OracleAS Single Sign-On および Directory Integration and Provisioning データの移行」

3. 中間層インスタンスが新しい Identity Management を使用するように変更します。

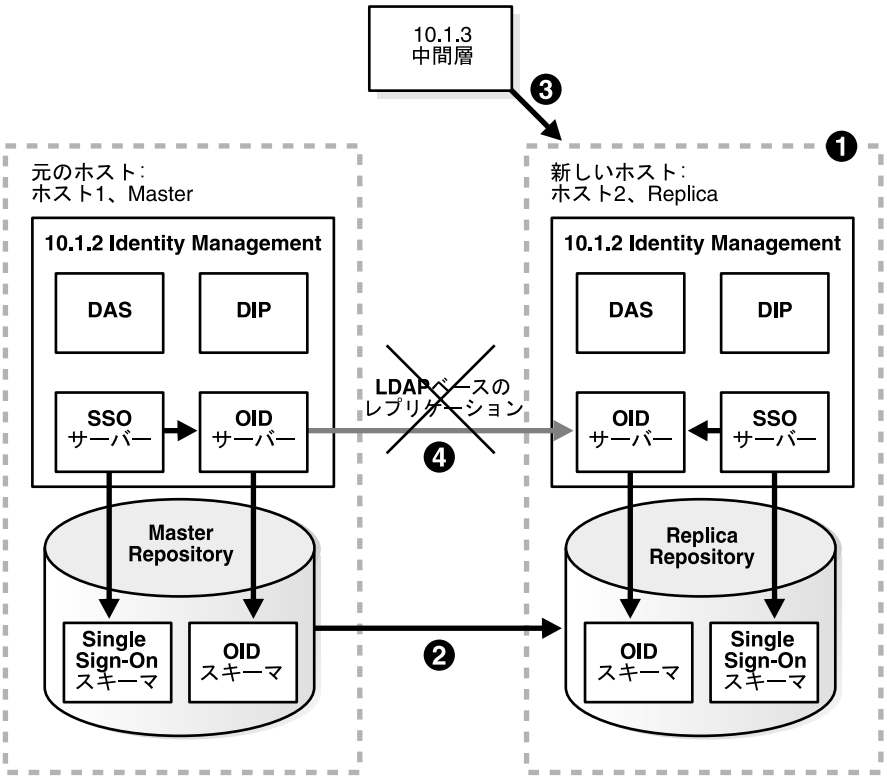
   関連項目：「作業 3: 中間層インスタンスが新しい Identity Management を使用するように変更」

4. LDAP ベースのレプリケーションを停止します。

   関連項目：「作業 4: レプリケーションの停止」

図 8-3 に、これらの手順を示します。

図 8-3 元の Identity Management から新しい Identity Management への変更



作業 1: 新しい Identity Management と Metadata Repository のインストールおよび設定

この作業では、新しいリリース 10.1.4 またはリリース 2（10.1.2）の Identity Management と、それに関連付けられた Metadata Repository をインストールおよび設定します。新しい Identity Management は、元の Identity Management の LDAP ベースのレプリカです。

1. LDAP ベースのレプリカおよびこの手順での使用方法は、第 F.1 項「LDAP ベースのレプリカについて」を参照してください。
2. 第 F.2 項「LDAP ベースのレプリカのインストールと設定」の手順に従って、新しい Identity Management と Metadata Repository をインストールおよび設定します。

作業 2: OracleAS Single Sign-On および Directory Integration and Provisioning データの移行

この作業では、元の Metadata Repository から新しい Metadata Repository に OracleAS Single Sign-On および Directory Integration and Provisioning データを移行します。移行元は元の Metadata Repository（Master）で、移行先は新しい Metadata Repository（Replica）です。

この作業には次の下位作業があります。

- OracleAS Single Sign-On データの移行
- Directory Integration and Provisioning データの移行

注意： 開始する前に、環境変数の ORACLE_HOME および ORACLE_SID が設定されていることを確認します。これはすべてのプラットフォームに適用されます。

OracleAS Single Sign-On データの移行

OracleAS Single Sign-On データを移行する手順は次のとおりです。

1. マスターの ORASSO スキーマ・パスワードを取得します。

```
MASTER_HOME/bin/ldapsearch -p master_oid_port -h master_host
 -D "cn=orcladmin" -w master_orcladmin_passwd
 -b "orclresourcename=orasso,  orclreferencename=master_global_db_name,
cn=ias infrastructure databases, cn=ias, cn=products, cn=oraclecontext"
-s base "objectclass=*" orclpasswordattribute
```

   このコマンドを実行すると、ORASSO パスワードが次のように行に出力されます。

```
orclpasswordattribute=LAetjdQ5
```

2. マスターから OracleAS Single Sign-On データをエクスポートします（このコマンドを実行する前に、環境変数 ORACLE_HOME が設定されていることを確認してください）。

```
MASTER_HOME/sso/bin/ssomig -export -s orasso -p master_orasso_passwd
-c master_db_name -log_d $MASTER_HOME/sso/log
```

   この例では、*master_orasso_passwd* は前の手順で取得した ORASSO パスワードです。

3. `ssomig.dmp` および `ssoconf.log` ファイルをマスターからレプリカにコピーします（各ファイルの正確なフルパスを保持）。

4. レプリカの ORASSO スキーマ・パスワードを取得します。

```
REPLICA_HOME/bin/ldapsearch -p replica_oid_port -h replica_host
-D "cn=orcladmin" -w replica_orcladmin_password -b "orclresourcename=orasso,
orclreferencename=replica_global_db_name, cn=ias infrastructure databases,
cn=ias, cn=products, cn=oraclecontext" -s base "objectclass=*"
orclpasswordattribute
```

5. OracleAS Single Sign-On データをレプリカにインポートします。

```
REPLICA_HOME/sso/bin/ssomig -import -overwrite -s orasso
-p replica_orasso_passwd -c replica_db_name
-log_d $REPLICA_HOME/sso/log -discoforce
```

   この例では、*replica_orasso_passwd* は前の手順で取得した ORASSO パスワードです。

6. OracleAS Single Sign-On のエクスポートとインポートが正常に完了したことを確認します。

   OracleAS Single Sign-On 移行ツールによって成功がレポートされていることを確認します。次のログ・ファイルでエラーをチェックすることもできます。

```
MASTER_HOME/sso/log/ssomig.log
REPLICA_HOME/sso/log/ssomig.log
```

   関連項目： ログ・ファイルのメッセージ解析の詳細は、『Oracle Application Server Single Sign-On 管理者ガイド』を参照してください。

7. 第 6.6 項の「作業 1: SSO 認証の有効化（オプション）」の説明に従って、SSO 認証を再度有効化します。

Directory Integration and Provisioning データの移行

Directory Integration and Provisioning データを移行する手順は次のとおりです。

> **関連項目：** Oracle Internet Directory HTTP ポートが無効になっている環境で、HTTPS ポートを使用して次のコマンドを実行する方法の詳細は、『Oracle Internet Directory 管理者ガイド』の Directory Integration and Provisioning データに関する説明を参照してください。

1. マスターの Directory Integration and Provisioning データ・サーバーを停止します。

   ```
   MASTER_HOME/bin/oidctl server=odisrv instance=1 stop
   ```

2. Directory Integration and Provisioning データを移行します。

   ```
   MASTER_HOME/bin/dipassistant reassociate -src_ldap_host master_host
   -src_ldap_port master_oid_port -dst_ldap_host replica_host
   -dst_ldap_port replica_oid_port -src_ldap_passwd master_orcladmin_passwd
   -dst_ldap_passwd replica_orcladmin_passwd
   ```

   ログ・メッセージが次のファイルに出力されます。

   ```
   MASTER_HOME/ldap/odi/log/reassociate.log
   ```

3. レプリカの Directory Integration and Provisioning データ・サーバーを停止します。

   ```
   REPLICA_HOME/bin/oidctl server=odisrv instance=1 stop
   ```

4. Directory Integration and Provisioning データ・サーバーをレプリカに登録します。

   ```
   REPLICA_HOME/bin/odisrvreg -D "cn=orcladmin" -w replica_orcladmin_passwd
   -h replica_host -p replica_oid_port
   ```

5. レプリカの Directory Integration and Provisioning データ・サーバーを起動します。

   ```
   REPLICA_HOME/bin/oidctl server=odisrv instance=1 flags="port=replica_oid_port"
   start
   ```

作業 3: 中間層インスタンスが新しい Identity Management を使用するように変更

各中間層インスタンスで、次の手順を実行します。

1. Application Server Control コンソールを使用して、中間層インスタンスの OC4J ホーム・ページにナビゲートします。
2. 「**管理**」をクリックします。
3. 表の「**タスク名**」列で「**セキュリティ**」が閉じている場合は、それを開きます。「ID 管理」行で、「**タスクに移動**」アイコンをクリックします。
4. 「ID 管理」ページで、「**変更**」をクリックします。
5. ウィザードの手順に従って、新しい Identity Management の情報を指定します。詳細は、第 6.6 項を参照してください。
6. 操作が完了したら、OC4J インスタンスを再起動する必要があります。「確認」ページで「再起動」をクリックしないでください。かわりに、「クラスタ・トポロジ」ページにナビゲートし、OC4J インスタンスを選択してから、「**再起動**」をクリックします。

中間層インスタンスが新しいホストを使用するように変更する際に問題が発生した場合は、レプリケーションが実行されていることを確認してからもう一度やりなおしてください。

作業 4: レプリケーションの停止

元の Identity Management と新しい Identity Management（Replica）の間のレプリケーションを停止します。これには、新しい Identity Management の Oracle ホームで次のコマンドを実行します。

```
oidctl connect=global_db_name server=oidrepld instance=1 flags="-p oid_port" stop
```

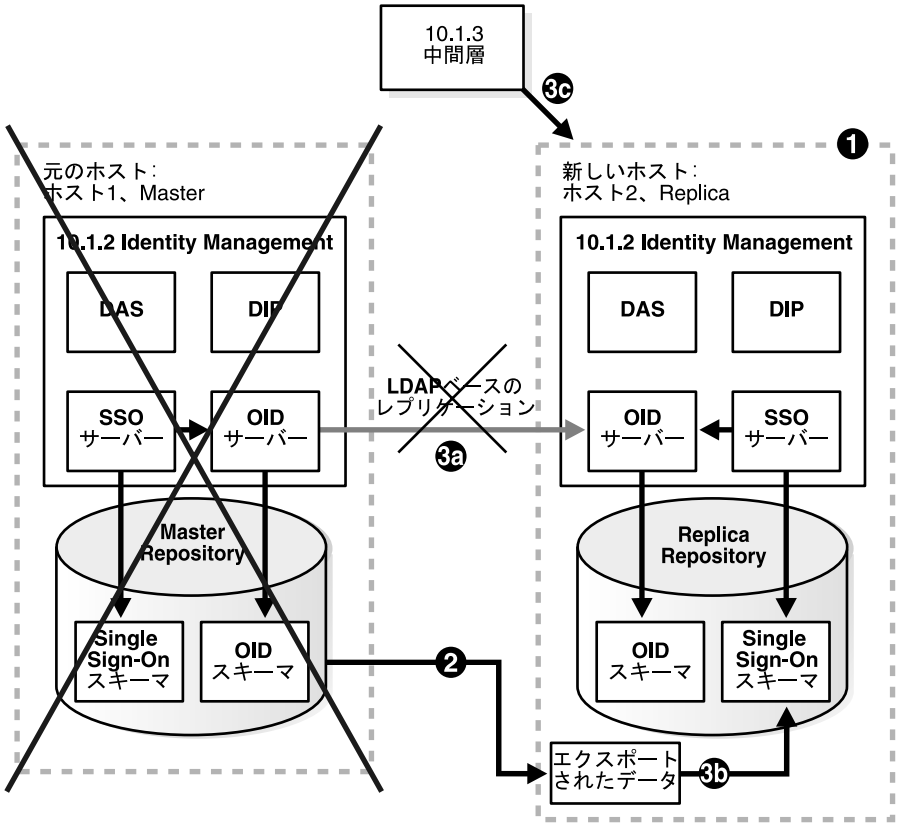
この例では、次のようになります。

- *global_db_name* は、新しい Identity Management のグローバル・データベース名です。
- *oid_port* は、新しい Identity Management の非 SSL の Oracle Internet Directory ポートです。

8.3.4 この手順を使用してフェイルオーバーを実施する方法

第 8.3.1 項で説明されているとおり、この手順に変更を加えると、Identity Management のフェイルオーバーを実施できます。これにより、元の Identity Management を失った場合に、中間層インスタンスを新しい Identity Management に移動することができます。

図 8-4 新しい Identity Management へのフェイルオーバー



フェイルオーバー環境を設定する手順は次のとおりです。

1. 新しい Identity Management をインストールおよび設定します。手順は、「作業 1: 新しい Identity Management と Metadata Repository のインストールおよび設定」を参照してください。
2. 元の Metadata Repository から OracleAS Single Sign-On データと Directory Integration and Provisioning データを定期的にエクスポートします。データを新しい Metadata Repository にインポートする必要はありません。データをエクスポートし、ファイルを新しい Metadata Repository ホストにコピーするだけでかまいません。詳細は、「作業 2: OracleAS Single Sign-On および Directory Integration and Provisioning データの移行」を参照してください。
3. 元の Identity Management を失った場合は、次の手順を実行します。
   a. 「作業 4: レプリケーションの停止」の説明に従って、レプリケーションを停止します。
   b. OracleAS Single Sign-On データと Directory Integration and Provisioning データの最新のコピーを新しい Identity Management リポジトリにインポートします。詳細は、「作業 2: OracleAS Single Sign-On および Directory Integration and Provisioning データの移行」を参照してください。
   c. 「作業 3: 中間層インスタンスが新しい Identity Management を使用するように変更」の手順に従って、中間層インスタンスが新しい Identity Management を使用するように変更します。

9

Application Server 中間層インスタンスのクローニング

この章では、Oracle Application Server 中間層インスタンスのインストールのクローニングについて説明します。

この章の項目は次のとおりです。

9.1 クローニングの概要

クローニングとは、既存のインストールを元の構成のまま別の場所へコピーするプロセスです。Oracle Application Server のインストールのクローニングは、次のような場合に役立ちます。

- 作成するインストールが、本番、テスト、または開発用インストールのコピーである場合。クローニングによって、すべてのパッチが適用された新しいインストールを 1 回の手順で作成できます。これは、Oracle Application Server のインストール、構成、パッチの適用を個別に行うこととはまったく異なる方法です。
- インスタンスおよびそのインスタンスがホストするアプリケーションを迅速にデプロイする場合。
- パッチを適用したホームの「ゴールド」イメージを作成し、それを多数のホストに配置する場合。

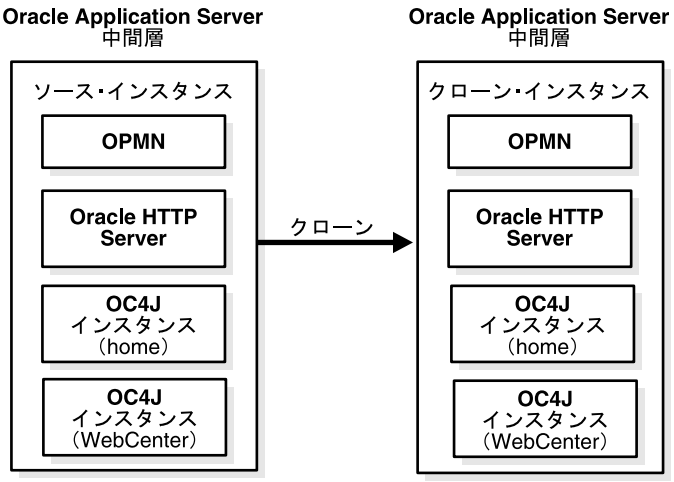
クローン・インストールは、ソース・インストールと同様に動作します。たとえば、クローン・インスタンスは、Oracle Universal Installer を使用して削除またはパッチを適用できます。また、クローン・インストールから別のクローンを作成することもできます。

コマンドラインのクローニング・スクリプトを使用して、テスト用、開発用または本番用のインストールをクローニングしたコピーを作成できます。

大半の用途には、デフォルトのクローニング手順で十分に対応できます。さらに、カスタム・ポート割当ての指定や、カスタム設定の保存など、クローニング・プロセスの様々な側面をカスタマイズすることもできます。

図 9-1 に、Oracle Identity Management に接続されていない Oracle WebCenter Framework および Oracle HTTP Server 中間層のクローニングを示します。

図 9-1 Oracle WebCenter Framework および Oracle HTTP Server 中間層のクローニング



クローニング・プロセスでは、ソース Oracle ホームにあるすべてのファイルをクローニング先 Oracle ホームにコピーします。したがって、ソース・インスタンスによって使用されているファイルの中に、ソース Oracle ホームのディレクトリ構造の外部に置かれているものがある場合、それはクローニング先にコピーされません。

ファイルをコピーした後は、一連のスクリプトを使用して、主要な構成ファイルの情報を更新します。たとえば、`httpd.conf` 内のホスト名と Oracle ホームへの参照はすべて、新しい値に更新されます。

ソース・インスタンスにデプロイされたアプリケーションのうち、ソース Oracle ホームのディレクトリ構造内にあるものも、クローン・インスタンスにコピーされ、自動的にデプロイされます。

9.2 クローニングできるインストール・タイプ

このリリースでは、次のタイプの中間層インストールをクローニングできます。

- Oracle WebCenter Framework
- Oracle HTTP Server のある Oracle WebCenter Framework
- Oracle HTTP Server

クローン Oracle ホームにおける特定のコンポーネントに影響する検討項目および制限の詳細は、第 9.5 項を参照してください。

次の事項に注意してください。

- Oracle Content DB が含まれるインスタンスはクローニングできません。ただし、別のインスタンスにある Oracle Content DB を Oracle WebCenter Framework のリポジトリとして使用するインスタンスはクローニングできます。
- OracleAS Infrastructure はクローニングできませんが、Oracle Identity Management に接続されている中間層はクローニングできます。ただし、クローン・インスタンスは Oracle Identity Management に関連付けられないため、手動で Oracle Identity Management に関連付ける必要があります。手順は、第 6.6 項「10.1.4 または 10.1.2 の Oracle Identity Management を使用するためのインスタンスの構成」を参照してください。
- クローン・インスタンスには、ソース・インスタンスと異なるインスタンス名を付ける必要があります。第 9.4.3 項に示されているように、インスタンス名はインスタンスのクローニング時に指定します。
- クラスタのメンバーである中間層インスタンスは、マルチキャスト検出または静的ノード検出がベースとなっている場合のみクローニングできます。詳細は、第 9.4.5 項を参照してください。

9.3 クローニング・プロセスの概要

クローニング・プロセスでは、Oracle Universal Installer のクローニング機能を利用します。処理は Oracle Application Server インストールに含まれている一連のスクリプトによって実行されます。次の項では、インスタンスのクローニングに関連するプロセスについて説明します。

1. ソース準備フェーズ
2. クローニング・フェーズ

9.3.1 ソース準備フェーズ

ソースで、`prepare_clone.pl` というスクリプトを実行します。これはクローニングに向けてソースを準備する Perl スクリプトです。クローニングに必要な情報のスナップショットを取得します。

このフェーズで、`prepare_clone.pl` はソース Oracle ホームのファイルを解析して、必要な値を抽出および格納し、必要なファイルをバックアップします。

その後で、Oracle ホーム・ディレクトリに対して tar を実行します。

ソース・インスタンスを準備する具体的な手順は、第 9.4.2 項を参照してください。

9.3.2 クローニング・フェーズ

クローニング先で、tar ファイルから Oracle ホームを抽出します。次に、`clone.pl` というスクリプトを実行します。これは、クローニング処理のすべてを自動的に実行する Perl スクリプトで、必要に応じて、他のユーティリティや Oracle Universal Installer を起動します。`clone.pl` スクリプトを起動すると、次の 3 つのフェーズが実行されます。

1. プリクローニング・フェーズ

   このフェーズでは、`clone.pl` スクリプトによって、クローニングを実行できるようにするために必要な土台を作ります。

2. クローニング・フェーズ

   このフェーズでは、clone.pl スクリプトによって、Oracle Universal Installer に必要な引数を指定してクローン・モードで起動し、Oracle Universal Installer ホームのクローニングを実行します。これによって、すべてのファイルの再インスタンス化（インスタンス化された既存ファイルのバックアップ作成後）、環境変数の設定、リンクの更新などが行われます。つまり、ファイルのコピーを除き、インストール時に実行された作業をすべて繰り返します。

3. ポストクローニング・フェーズ

   ポストインストール構成アシスタントは、クローニング時に再度実行するようには設計されていません。したがって、構成アシスタントによって更新されるインスタンス固有の構成ファイルの一部は、Oracle Universal Installer のクローニング・セッションが終了しても更新されません。かわりに、Oracle が用意した一連のポストクローニング・スクリプトで、これらのファイルを更新し、クローン・ホームを作業可能な状態にします。

   スクリプトによって実行されるポストクローニング手順は次のとおりです。

   a. 新規 Oracle ホームを設定します。

   b. 構成ファイルを更新します。この手順では、クローニング・フェーズで Oracle Universal Installer によって再インスタンス化された多数の構成ファイルが、バックアップからリストアされます。これらのファイルは必要に応じて、新しい環境を反映した新しい値で更新されます。たとえば、ファイルにソース Oracle ホームを参照する記述がある場合、その記述はクローニング先 Oracle ホームを参照するように更新されます。

   c. ホームの chgiphost コマンドを呼び出すことによって、クローン・ホームのホスト名と IP アドレスが変更されます。chgiphost を呼び出す前に、スクリプトは chgiphost をサイレント・モードで起動するために必要な次の情報を集める必要があります。

      – ソースのホスト名
      – ソースの IP アドレス
      – クローニング先のホスト名
      – クローニング先の IP アドレス

      クローニングの一部として chgiphost を実行する場合、（ホスト名やドメイン名の変更などで）chgiphost をスタンドアロンで実行するときとは違い、すべての構成ツールが実行されるわけではありません。

   d. ソース・インスタンスが Oracle Internet Directory に接続されている場合、クローンについての情報を Oracle Internet Directory に追加します。

   e. クローニング処理がすべて完了した後、サービスと Application Server Control コンソールを起動して、クローニング処理が正常に行われたことを確認します。

各フェーズを手動で実行する必要はありません。clone.pl スクリプトが 3 つのフェーズをすべて自動的に処理します。ここで示した情報は、概念を理解するためのものにすぎません。

クローニング先での具体的な作業手順は、第 9.4.3 項を参照してください。

ポストクローニング・フェーズで更新されるファイル

ポストクローニング・フェーズでは、いくつかの重要な構成ファイルがバックアップからリストアされ、更新されます。ファイルに対する典型的な変更は、ホスト名、Oracle ホーム、ポート番号などの環境固有の変数を新しい値に更新することです。

次のリストに、更新される重要なファイルの一部を示します。これは、更新されるファイルをすべて網羅しているわけではありません。

- *Oracle_Home*/config/ias.properties
- *Oracle_Home*/sysman/j2ee/application-deployments/ascontrol/orion-web.xml

- *Oracle_Home*/Apache/Apache/conf/httpd.conf
- *Oracle_Home*/Apache/Apache/conf/mod_oc4j.conf
- *Oracle_Home*/Apache/Apache/conf/oracle_apache.conf
- *Oracle_Home*/Apache/modplsql/conf/dads.conf
- *Oracle_Home*/Apache/modplsql/conf/plsql.conf
- *Oracle_Home*/Apache/modplsql/conf/cache.conf
- *Oracle_Home*/Apache/oradav/conf/moddav.conf
- *Oracle_Home*/opmn/conf/opmn.xml
- *Oracle_Home*/backup_restore/config/config_misc_files.inp

ここでのパスの形式は、UNIX 形式で示されています。Windows では、スラッシュが円記号になります。

9.4 Oracle Application Server インスタンスのクローニング

Oracle Application Server インスタンスをクローニングするには、Companion CD からスクリプトをコピーします。最初にソース Oracle ホームを準備し、次に、相手先をクローニングします。

9.4.1 クローニングの前提条件

クローニングを実行するには、Perl 5.83 以降をシステムにインストールしておく必要があります。クローニングを行う Perl スクリプトを実行する前に、PERL5LIB 環境変数に、Oracle ホームの Perl ディレクトリへのパスを設定する必要があります。このパスは、変数定義の最初に示されているパスと一致する必要があります。たとえば、次のように指定します。

- UNIX の場合 :

```
export PERL5LIB=$ORACLE_HOME/perl/lib/5.8.3/i686-linux-thread-multi:$ORACLE_
HOME/perl/lib/5.8.3:$ORACLE_HOME/perl/lib/site_perl/5.8.3/i686-linux-thread-multi/
```

- Windows の場合 :

```
set PERl5LIB=%ORACLE_HOME%¥perl¥5.8.3¥lib;%ORACLE_
HOME%¥perl¥5.8.3¥lib¥MSWin32-x86-multi-thread;%ORACLE_
HOME%¥perl¥site¥5.6.1¥lib;%ORACLE_HOME%¥perl¥site¥5.8.3¥lib
```

9.4.2 ソースの準備

ソース Oracle ホームのクローニングを準備するには、ソース・インスタンスで次の手順を実行します。

1. 次のディレクトリに移動します。

   (UNIX) *ORACLE_HOME*/clone/bin
   (Windows) *ORACLE_HOME*¥clone¥bin

2. prepare_clone.pl スクリプトを実行します。このスクリプトによって、ソースをクローニングする準備ができます。

   このスクリプトのコマンドラインは、次のような形式になります。

   perl prepare_clone.pl [ORACLE_HOME=*OH_dir*]
   [-silent]
   [-debug]
   [-export]
   [-help]

この例では、*perl* はそれぞれ、次のように置き換えます。

- UNIX の場合 :

  ```
  $ORACLE_HOME/perl/bin/perl
  ```

- Windows の場合 :

  ```
  %ORACLE_HOME%¥perl¥5.8.3¥bin¥MSWin32-x86-multi-thread¥perl5.8.3
  ```

表 9-1 に、prepare_clone.pl スクリプトのパラメータとオプションを示します。

表 9-1 prepare_clone.pl スクリプトのパラメータとオプション

パラメータまたはオプション	説明
ORACLE_HOME	ソース Oracle ホームの完全なディレクトリ指定。このパラメータを指定せずにスクリプトを実行すると、ORACLE_HOME 環境変数が存在する場合はそれが使用されます。この環境変数が存在しない場合、このスクリプトでは、スクリプトの実行場所であるディレクトリが ORACLE_HOME と想定されます。 Oracle ホームを指定するときは、スラッシュ（UNIX）や円記号（Windows）を最後に使用しないでください。 インストール時に提供された値を使用します。シンボリック・リンクは使用しないでください。 ORACLE_HOME が無効の場合、スクリプトは終了し、標準出力（STDOUT）にエラーが記録されます。
-silent	スクリプトがサイレント・モードで実行されます。パスワードに関連する必須オプションがコマンドラインに含まれていない場合、スクリプトは終了します。
-debug	スクリプトがデバッグ・モードで実行されます。
-export	ソース・インスタンス上の MDS に格納されているページ・カスタマイズ・データやポートレット・メタデータを .ear ファイルにエクスポートします。また、ポートレットのカスタマイズ・データ（プリファレンス・データ）を .ear ファイルにエクスポートします。クローン・インスタンスで別の場所の MDS を使用する場合は、このオプションを使用して、WebCenter アプリケーションに関連付けられているカスタマイズ・データを別の場所に移行します。 このオプションでは、Oracle WebCenter Framework Predeployment ツールのエクスポート・モードをコールします。Predeployment ツールの詳細は、『Oracle WebCenter Framework 開発者ガイド』の WebCenter アプリケーションのデプロイに関する項を参照してください。 スクリプトによって、_clone_export.ear という接尾辞の .ear ファイルが、次のディレクトリに作成されます。 (UNIX) *ORACLE_HOME*/j2ee/*instance*/applications/*app_name*/*app_name*_clone_export.ear (Windows) *ORACLE_HOME*¥j2ee¥*instance*¥applications¥*app_name*¥*app_name*_clone_export.ear
-help	スクリプトの使用方法が出力されます。

3. アーカイブ用のツールを使用して、ソース Oracle ホームをアーカイブおよび圧縮します。たとえば、Windows では WinZip、UNIX では tar と gzip が使用できます。使用しているツールが、ファイルの権限とタイムスタンプを保存することを確認してください。UNIX でソースをアーカイブおよび圧縮する方法を、次の例で示します。

```
cd Source_Oracle_Home
tar cf - * | gzip > oracleas.tar.gz
```

ファイルの中に sticky bit に設定されているものがある場合、tar ユーティリティは警告を発する場合があります。この警告は無視してかまいません。

Oracle ホームのアーカイブおよび圧縮には、jar ユーティリティを使用しないようにしてください。

9.4.3 インスタンスのクローニング

相手先でソース・インスタンスをクローニングするには、次の手順を実行します。

1. 圧縮した Oracle ホームを、ソース・マシンからクローニング先マシンにコピーします。
2. 圧縮した Oracle ホームを、クローニング先の新しい Oracle ホームとなるディレクトリに解凍します。圧縮ファイルの解凍には、それに適したツールを使用します。たとえば、Windows では WinZip、UNIX では tar と gunzip が使用できます。使用しているツールが、ファイルの権限とタイムスタンプを保存することを確認してください。UNIX でファイルを解凍する方法を、次の例で示します。

```
mkdir -p Destination_Oracle_Home
cd Destination_Oracle_Home
gunzip  <  Dir_Containing_Tar/oracleas.tar.gz | tar xf -
```

注意： ソース・マシンとクローニング先マシンで、tar および gzip（または gunzip）のバージョンが同じである必要があります。バージョンが異なると、アーカイブを解凍するときに問題が発生する場合があります。

---

3. 次のディレクトリに移動します。

```
(UNIX) ORACLE_HOME/clone/bin
(Windows) ORACLE_HOME¥clone¥bin
```

4. clone.pl スクリプトを実行します。Oracle インベントリ・ファイルが含まれたディレクトリには、書込み権限が必要です（Oracle インベントリ・ディレクトリの場所の詳細は、第 9.4.4 項を参照してください）。

このスクリプトのコマンドラインは、次のような形式になります。

```
perl clone.pl ORACLE_HOME=OH_dir
              ORACLE_HOME_NAME=OH_Name
              -instance Instance_Name
              {-oc4jadmin_old_password old_admin_pass |
               -oc4jadmin_obf_old_password old_obf_admin_pass}
              {-oc4jadmin_new_password new_admin_pass |
               -oc4jadmin_obf_new_password new_obf_admin_pass}
              [-Ostring]
              [-silent]
              [-debug]
              [-import]
              [-help]
```

この例では、*perl* はそれぞれ、次のように置き換えます。

- UNIX の場合 :

  ```
  $ORACLE_HOME/perl/bin/perl
  ```

- Windows の場合 :

  ```
  %ORACLE_HOME%¥perl¥5.8.3¥bin¥MSWin32-x86-multi-thread¥perl5.8.3
  ```

表 9-2 に、clone.pl スクリプトのパラメータとオプションを示します。

表 9-2 clone.pl スクリプトのパラメータとオプション

パラメータまたはオプション	説明
ORACLE_HOME	**必須。**クローニング先 Oracle ホームの完全なディレクトリ指定。このパラメータは必須です。このパラメータを指定しない場合、またはその値が無効の場合は、スクリプトは終了します。 Oracle ホームを指定するときは、スラッシュ（UNIX）や円記号（Windows）を最後に使用しないでください。
ORACLE_HOME_NAME	**必須。**クローニング先 Oracle ホーム（クローンの Oracle ホーム）の名前。
-instance	**必須。**クローンのインスタンス名。このインスタンスには、ソース・インスタンス、またはそれ以外に、同じ OracleAS Infrastructure を使用するインスタンスや同じクラスタ・トポロジの一部であるインスタンスとは、異なる名前を付ける必要があります。
-oc4jadmin_old_password	**-oc4jadmin_obf_old_password が使用されていない場合は必須。**ソース・インスタンスに対する Oracle Application Server の管理者 oc4jadmin 用パスワード。このオプションも -oc4jadmin_obf_old_password も指定せず、スクリプトがサイレント・モードで実行されていない場合は、スクリプトから、パスワードを入力するように求められます。
-oc4jadmin_obf_old_password	**-oc4jadmin_old_password が使用されていない場合は必須。**ソース・インスタンスに対する Oracle Application Server の管理者 oc4jadmin 用の不明瞭化されたパスワード。このオプションも -oc4jadmin_old_password も指定せず、スクリプトがサイレント・モードで実行されていない場合は、スクリプトから、パスワードを入力するように求められます。
-oc4jadmin_new_password	**-oc4jadmin_obf_new_password が使用されていない場合は必須。**クローン・インスタンスに対する Oracle Application Server の管理者 oc4jadmin の新規パスワード。このオプションも -oc4jadmin_obf_new_password も指定せず、スクリプトがサイレント・モードで実行されていない場合は、スクリプトから、パスワードを入力するように求められます。 このパスワードはデフォルトの OC4J インスタンスに使用され、それ以外の OC4J インスタンスには使用されません。詳細は、第 9.5.3 項を参照してください。
-oc4jadmin_obf_new_password	**-oc4jadmin_new_password が使用されていない場合は必須。**クローン・インスタンスに対する Oracle Application Server の管理者 oc4jadmin の不明瞭化された新規パスワード。このオプションも -oc4jadmin_new_password も指定せず、スクリプトがサイレント・モードで実行されていない場合は、スクリプトから、パスワードを入力するように求められます。 このパスワードはデフォルトの OC4J インスタンスに使用され、それ以外の OC4J インスタンスには使用されません。詳細は、第 9.5.3 項を参照してください。

表 9-2 clone.pl スクリプトのパラメータとオプション（続き）

パラメータまたはオプション	説明
-O	このオプションに続くテキストが、Oracle Universal Installer コマンドラインに渡されるように指定されます。たとえば、次のコードでこのオプションを使用すると、Oracle Universal Installer が使用する oraparam.ini ファイルの場所を渡せます。 `'-O-paramFile C:¥OraHome_1¥oui¥oraparam.ini'` 渡すテキストに空白などの区切り文字が含まれている場合は、そのオプションを二重引用符（"）で囲む必要があります。 このオプションを使用して複数のパラメータを Oracle Universal Installer に渡すには、すべてのパラメータに -O オプションを 1 つ付けて渡すか、個々のパラメータに複数の -O オプションを使用して渡します。
-silent	スクリプトがサイレント・モードで実行されます。パスワードに関連する必須オプションがコマンドラインに含まれていない場合、スクリプトは終了します。
-debug	スクリプトがデバッグ・モードで実行されます。
-import	ソース・インスタンス上の MDS に格納されているページ・カスタマイズ・データやポートレット・メタデータをクローン・インスタンスにインポートします。また、ポートレットのカスタマイズ・データ（プリファレンス・データ）もインポートします。prepare_clone.pl コマンドラインの -export オプションで生成された .ear ファイルからカスタマイズ・データをインポートします。新しいインスタンスで別の場所の MDS を使用する場合は、このオプションを使用して、WebCenter アプリケーションに関連付けられているカスタマイズ・データを別の場所に移行します。 このオプションでは、Oracle WebCenter Framework Predeployment ツールのインポート・モードをコールします。Predeployment ツールの詳細は、『Oracle WebCenter Framework 開発者ガイド』の WebCenter アプリケーションのデプロイに関する項を参照してください。 スクリプトによって、`prepare_clone.pl -export` オプションで作成した ear ファイルがインポートされます。
-help	スクリプトの使用方法が出力されます。

たとえば、次のように指定します。

```
perl clone.pl ORACLE_HOME=/scratch/oracle/Ora_10131_B
              ORACLE_HOME_NAME=OH_10131_B
              -instance orcl_B
              -oc4jadmin_old_password my_old_admin_pass
              -oc4jadmin_new_password my_new_admin_pass
              '-O-paramFile /var/opt/oracle/oui/oraparam.ini'
              -import
              -silent
```

5. デプロイされた WebCenter アプリケーションがソース・インスタンスに含まれる場合、同じ場所の MDS を使用するのか別の場所の MDS を使用するのかをクローニング・スクリプトによって尋ねられます。

   たとえば、テスト環境から本番環境に移行する場合、新しい場所の MDS を指定できます。ただし、新しいインスタンスを追加することによって環境を拡張する場合は、ソース・インスタンスと同じ MDS を使用することをお薦めします。

   ```
   Specified mds path is mds_path. Do you want to keep the original settings [n|y]
   [y]:
   ```

   同じ場所の MDS を使用するには、**y** を指定します。

   別の場所の MDS を使用するには、**n** を指定します。次に、プロンプトで、新しい場所を入力します。絶対パスを指定してください。また、MDS の場所はアクセス可能である必要があります。つまり、ユーザーには読取りと書込みができる権限が必要です。

   誤った場所を指定した場合は、クローニング処理の完了後に場所を変更できます。場所の変更は、次のディレクトリにある `adf-config.xml` ファイルで行います。

   ```
   ORACLE_HOME/j2ee/OC4J_Instance/applications/apps_name/adf/META-INF
   ```

   ディレクトリの指定において、*OC4J_instance* は OC4J インスタンスの名前、*apps_name* はアプリケーションの名前です。

6. ソース・インスタンスがマルチキャスト動的ノード検出または静的ノード検出クラスタのメンバーである場合は、スクリプトによって、元のクラスタ設定を保持するかどうか尋ねられます。詳細は、第 9.4.5 項を参照してください。

7. UNIX では、クローン・インスタンスが正しく機能するように、Oracle ホームで `root.sh` スクリプトを実行します。このスクリプトを実行するには root ユーザーとしてログインしている必要があります。このスクリプトは、クローン・インスタンスの Oracle ホーム・ディレクトリにあります。

   たとえば、次のように指定します。

   ```
   $ORACLE_HOME/root.sh
   ```

8. UNIX では、これがコンピュータ上の最初の Oracle インストールである場合、root ユーザーとして `orainstRoot.sh` スクリプトを実行し、Oracle インベントリ・ディレクトリを登録する必要があります。このスクリプトは oraInventory ディレクトリにあります。

   oraInventory ディレクトリの場所は、次のファイルに含まれます。

   ```
   ORACLE_HOME/clone/logs/clonetimestamp.log
   ```

9. 新しい場所の MDS を使用する場合は、Application Server Control コンソールを使用して、クローン・インスタンスにアプリケーションを再デプロイします。WebCenter アプリケーションのデプロイの詳細は、『Oracle WebCenter Framework 開発者ガイド』の WebCenter アプリケーションのデプロイに関する項を参照してください。

10. クローン・インスタンスで別の場所の MDS が使用されており、WebCenter アプリケーションが含まれている場合の追加手順の詳細は、『Oracle WebCenter Framework 開発者ガイド』のクローニングによるステージング環境から本番環境への移行に関する項を参照してください。

11. クローン・インスタンスを再起動します。

    - UNIX の場合 :

      ```
      ORACLE_HOME/opmn/bin/opmnctl stopall
      ORACLE_HOME/opmn/bin/opmnctl startall
      ```

    - Windows の場合 :

      ```
      ORACLE_HOME¥opmn¥bin¥opmnctl stopall
      ORACLE_HOME¥opmn¥bin¥opmnctl startall
      ```

これで、クローン・インスタンスは、ソース・インスタンスと同じ構成になります。OC4J カスタム・インスタンスなど、クローン・インスタンスのプロセスはすべて、Application Server Control コンソールと OPMN で起動および停止できます。デプロイされたアプリケーションはすべて表示され、正しく実行できます。

9.4.4 ログ・ファイルの検索と表示

クローニング・スクリプトは複数のツールを呼び出しますが、これらのツールは個別にログ・ファイルを生成します。ただし、次に示すログ・ファイルは、Oracle Universal Installer およびクローニングを行うスクリプトによって生成されるもので、診断を目的とした主要なログ・ファイルです。

- *Oracle_inventory*/logs/cloneActions*timestamp*.log: ここには、クローニングの Oracle Universal Installer 部分で発生したアクションの詳細なログが記録されます。
- *Oracle_inventory*/logs/oraInstall*timestamp*.err: ここには、Oracle Universal Installer の実行中に発生したエラーに関する情報が記録されます。
- *Oracle_inventory*/logs/oraInstall*timestamp*.out: ここには、Oracle Universal Installer で生成された他の様々なメッセージが記録されます。
- *Oracle_Home*/clone/logs/clone*timestamp*.log: ここには、プリクローニングおよびクローニングの処理中に発生したアクションの詳細なログが記録されます。
- *Oracle_Home*/clone/logs/error*timestamp*.log: ここには、プリクローニングおよびクローニングの処理中に発生したエラーに関する情報が記録されます。また、クローニング・スクリプトにより呼び出された複数のツールによって書き込まれた標準エラー（STDERR）メッセージがすべて記録されます。メッセージは、ツールによって参考メッセージであったりエラー・メッセージであったりします。

ここでのパスの形式は、UNIX 形式で示されています。Windows では、スラッシュが円記号になります。

注意： Oracle インベントリ・ディレクトリの場所は、次のファイルに含まれます。

Oracle_Home/clone/logs/clone*timestamp*.log

次に例を示します。

```
Wed Jul  5 09:42:51 2006  INFO: Please check
/scratch/oracleas/oraInventory/logs/cloneActions2006-07-05_
09-38-30AM.log for more details.
```

Windows システムの場合、場所は、レジストリ HKEY_LOCAL_MACHINE¥SOFTWARE¥ORACLE¥INST_LOC から取得できます。

---

clone.pl スクリプトの実行後にこれらのログ・ファイルを開いて、クローニング・プロセスの詳細を調べてください。Application Server Control コンソールでログ・ファイルを表示する手順は次のとおりです。

1. ホーム・ページで「**ログ**」を選択します。
2. 「ログの表示」ページで、「**使用可能なコンポーネント**」ボックスから「**ASClone**」を選択します。「**移動**」をクリックして、選択したコンポーネントを「**選択したコンポーネント**」ボックスに移動します。
3. 「**検索**」をクリックします。

   「**結果**」表にログ・ファイルが表示されます。
4. このログを表示するには、「**ログ・ファイル**」列でログ名をクリックします。

9.4.5 クラスタ・トポロジのメンバーであるインスタンスのクローニング

クラスタのメンバーである中間層インスタンスは、マルチキャスト動的ノード検出または静的ノード検出がベースとなっている場合のみクローニングできます。この場合、クローニング・スクリプトによって、元のクラスタ設定を保持するかどうか尋ねられます。次に例を示します。

```
cluster Config:<Multi Casting>detected for Source Instance:Do you want to keep
the original cluster settings(n|y)[y]:
```

y と答えると、クローン・インスタンスは、ソース・インスタンスと同じクラスタの一部となります。

n と答えると、クローニング・スクリプトによって、クラスタの新しい IP アドレスとポートを入力するよう求められます。この場合、クラスタのタイプは維持されます。たとえば、ソース・インスタンスが動的ノード検出クラスタのメンバーである場合、n と答えると、新しいマルチキャスト検出 IP アドレスおよびポートを入力するよう求められます。クローニングの実行中は、クラスタのタイプを静的ノード検出クラスタなどの別のタイプに変更できません。

トポロジ間ゲートウェイまたは手動のノード間構成をベースとしているクラスタのメンバーであるインスタンスはクローニングできません。これらは、最初にクラスタから削除する必要があります。

9.5 クローニングに関する検討事項と制限事項

次の項では、クローニング全般およびクローン Oracle ホームの特定のコンポーネントに影響する検討事項と制限事項について、詳しく説明します。

- クローニングに関する一般的な検討事項と制限事項
- Oracle HTTP Server のクローニングに関する検討事項
- Oracle Containers for J2EE（OC4J）のクローニングに関する検討事項
- Application Server Control のクローニングに関する検討事項
- Oracle WebCenter Framework のクローニングに関する検討事項

9.5.1 クローニングに関する一般的な検討事項と制限事項

このリリースでは、次のものはクローニングできません。

- OracleAS Infrastructure のコンポーネント（Oracle Identity Management および OracleAS Metadata Repository）
- Oracle Content DB

クローニングについては、これ以外にも次の点に注意してください。

- 新しいホスト名と一致するように、セキュリティ証明書を更新して設定することが必要になる場合があります。Wallet および証明書の管理の詳細は、第 11 章を参照してください。
- Oracle Identity Management に接続されている Oracle Application Server 中間層をクローニングできます。ただし、クローニングされた中間層インスタンスは Oracle Identity Management に関連付けられません。クローニングされた中間層インスタンスは、手動で Oracle Identity Management に関連付ける必要があります。中間層を Oracle Identity Management に関連付けるには、第 6.6 項「10.1.4 または 10.1.2 の Oracle Identity Management を使用するためのインスタンスの構成」を参照してください。
- 構成ファイルのデフォルトのファイル権限を変更した場合、クローニングを実行したときに、その変更後のファイル権限は保存されません。
- 次のコンポーネントに対して、ユーザーが行ったカスタマイズの内容は保存されません。これらのコンポーネントのステータスは、デフォルトにリセットされます。
  - Oracle Application Development Framework
  - Port Tunneling

  - UIX
  - XDK

- クローニングを実行しても、ロード可能なモジュールやアプリケーション固有のライブラリなど、ソース Oracle ホームのすべての依存状態がクローン Oracle ホームに継承されるわけではありません。クローニングでは、ソース Oracle ホーム全体がクローニング先 Oracle ホームにコピーされるためです。ソース Oracle ホームの外部にあるファイルは、自動的にはコピーされません。そのため、ソース Oracle ホームの外部にあるファイルを参照するアプリケーションは、クローン Oracle ホームで正しく機能しない場合があります。

  アーカイブ済のソース Oracle ホームを抽出してから、`clone.pl` スクリプトを実行するまでに、ファイルをクローニング先ホストに手動でコピーすることが必要な場合があります。

- ソース Oracle ホームの外部にあるファイルまたはアプリケーション（例、デフォルトの場所に格納されていない Oracle Wallet ファイル）に対してシンボリック・リンクを作成してある場合、アプリケーションを正しく機能させるために、クローン Oracle ホームでリンクを手動で作成しなおす必要があります。

- クローニング処理を実行すると、クローン・インスタンスのデフォルト・ポートが生成されます。他のポートを指定するには、第 9.6.2 項に示すように `staticports.ini` ファイルを使用できます。UNIX で 1024 未満のポートを指定した場合は、クローニング処理中にクローン・インスタンスが起動することはありません。クローニング処理の完了後、root 権限で `root.sh` スクリプトを実行してから、このプロセスを開始する必要があります。

- クローニング・プロセスを実行しても、ロード・バランシング・ルーターがクローン・インスタンスを認識するように構成されるわけではありません。ユーザー環境でロード・バランシング・ルーターを使用する場合は、無効化ポートを含め、ロード・バランシング・ルーターを手動で構成する必要があります。

- クローニング処理が失敗したにもかかわらず、Oracle ホームが Oracle インベントリに登録された場合、以降のクローニング処理ではその Oracle ホームを使用できません。この場合、以降のクローニング処理で Oracle ホームとして別のディレクトリ名を使用するか、次のクローニング処理の前に Oracle ホームを削除します。

9.5.2 Oracle HTTP Server のクローニングに関する検討事項

次に、Oracle HTTP Server のクローニングに関する重要な情報を説明します。

- 次のファイルの構成情報はすべて更新されます。
  - *Oracle_Home*/Apache/Apache/conf/httpd.conf
  - *Oracle_Home*/Apache/Apache/conf/mod_oc4j.conf
  - *Oracle_Home*/Apache/Apache/conf/oracle_apache.conf
  - *Oracle_Home*/Apache/modplsql/conf/cache.conf
  - *Oracle_Home*/Apache/modplsql/conf/dads.conf
  - *Oracle_Home*/Apache/modplsql/conf/plsql.conf
  - *Oracle_Home*/Apache/oradav/conf/moddav.conf

  ここでのパスの形式は、UNIX 形式で示されています。Windows では、スラッシュが円記号になります。

  クローニング・スクリプトは、ソースの設定を保存し、これらのファイルを新しい環境パラメータで更新します。

  クローニングで更新されるのは、既知のファイル、つまり元のインストールに含まれていたファイルのみです。特に、クローニングでは、`httpd.conf`、`oracle_apache.conf`、`dads.conf`、`plsql.conf`、`olap.conf`、`moddav.conf` などのファイルの「include」リストにユーザーが追加した構成ファイルは更新されません。ただし、クローニングで更新されるファイルのリストに「include」ファイルを明示的に追加できます。カスタム設定の更新方法の詳細は、第 9.6.3 項を参照してください。

- クローニングでは、httpd.conf 内の VirtualHost 命令ディレクティブがすべて保存されます。これらのディレクティブの中に、ソース・ホームへの参照があれば、すべて置き換えられます。ただし、クローニングでは、これらの仮想ホストがリスニングする IP アドレスまたはポート番号は変更されません。

  これらの値がクローニング先の環境で有効ではない場合、次のいずれかの手順を実行する必要があります。

  – クローン・スクリプトにこれらの変更を登録して、クローニング時に更新されるようにします。詳細は、第 9.6.3 項を参照してください。

  – クローニング後、httpd.conf で、手動で更新します。

- httpd.conf のポート番号を変更して、ローカルの Oracle HTTP Server ポートでなく、ロード・バランシング・ルーターのポートを使用するようにしていた場合、その変更はクローニング後に破棄されます。クローン・ホームの httpd.conf ファイルを編集して、ポート番号をロード・バランシング・ルーターのポートに変更する必要があります。

- 使用している Oracle HTTP Server のベースが Apache 1.3 または Apache 2.0 の場合、クローニングはサポートされません（これらはデフォルトではインストールされませんが、付属の CD-ROM に同梱されています）。

9.5.3 Oracle Containers for J2EE（OC4J）のクローニングに関する検討事項

次に、OC4J のクローニングについての検討事項を示します。

- クローニング処理の準備フェーズ時にソースの Oracle ホームで、prepare_clone.pl スクリプトの実行中に、OC4J アプリケーションをアンデプロイしようとしたり、OC4J アプリケーションのその他の管理作業を実行しないでください。

- カスタム OC4J インスタンスに環境固有の情報を含むファイルは手動で登録し、それらのファイルがクローニング時に更新されるようにする必要があります。oc4j.properties はそうしたファイルの一例です。詳細は、第 9.3 項を参照してください。

- OC4J インスタンスが、ソース Oracle ホームに含まれない Oracle HTTP Server インスタンスを使用している場合、クローニングを実行しても、Oracle HTTP Server の mod_oc4j.conf ファイルは更新されません。mod_oc4j.conf ファイルに、インスタンスを手動で追加する必要があります。

- クローニング時には、デフォルトの OC4J インスタンスのパスワードを指定します。ソース・インスタンスにデフォルト以外の OC4J インスタンスが含まれる場合、クローン・インスタンスの OC4J インスタンスは、ソースの OC4J インスタンスと同じパスワードを使用します。つまり、デフォルトの OC4J インスタンスに指定されているパスワードは使用されません。パスワードは Application Server Control コンソールを使用して変更できます。

  グループ内の各 OC4J インスタンスに同じ oc4jadmin パスワードを指定する必要があることに注意してください。詳細は、第 2.3.3.2 項「Application Server Control によるグループの管理」を参照してください。

- OPMN は、クローニングされたデフォルトおよびカスタムの OC4J インスタンスを、すべて管理できます。

- クローン・インスタンス上の Grid Control コンソールは、デフォルトおよびカスタムの OC4J インスタンスを管理できます。

次に、どの OC4J コンポーネントが保存されるかを説明します。

- デフォルトの OC4J インスタンスはすべて保存されます。

- ユーザーが作成したカスタム OC4J インスタンスと、その中にデプロイされたアプリケーションは保存されます。ただし、これらのアプリケーションが Oracle ホームの中にない場合、アプリケーションの外部依存関係は、クローン・ホームにはコピーされずに破棄されます。

- data-sources.xml の中のデータソース情報は保存されます。

- jms.xml、java2.policy、jazn.xml、jazn-data.xml、global-web-application.xml および application.xml に含まれるユーザー構成は保存されます。

9.5.4 Application Server Control のクローニングに関する検討事項

Application Server Control コンソールをクローニングする際は、次の検討事項を参考にしてください。

- ソース・インスタンスにソース専用の Application Server Control が含まれる場合、クローン・インスタンスにはクローン専用の Application Server Control が含まれ、クローン・インスタンスの管理に使用されます。
- ソース・インスタンスが、別の Oracle ホームにデプロイされている Application Server Control によって管理されている場合、クローン・インスタンスは同じ Application Server Control によって管理されます。

default-web-site.xml ファイルを維持することで、ソース・インスタンスの SSL 設定が維持されます。つまり、ソースの Application Server Control コンソールが HTTPS 用に構成されている場合は、クローンの Application Server Control コンソールも HTTPS 用に構成されます。

9.5.5 Oracle WebCenter Framework のクローニングに関する検討事項

Oracle WebCenter Framework をクローニングする際は、次の検討事項を参考にしてください。

- WebCenter アプリケーションで使用されている Oracle Metadata Services（MDS）の場所が Oracle ホーム内でなく、共有ドライブ上でもない場合、クローン・インスタンスではその場所の MDS を使用できません。
- デプロイされた WebCenter アプリケーションがソース・インスタンスに含まれる場合、同じ場所の MDS を使用するのか別の場所の MDS を使用するのかをクローニング・スクリプトによって尋ねられます。

  たとえば、テスト環境から本番環境に移行する場合、新しい場所の MDS を指定できます。ただし、新しいインスタンスを追加することによって環境を拡張する場合は、ソース・インスタンスと同じ MDS を使用することをお薦めします。
- カスタマイズ・データ（デプロイされたアプリケーションのプロデューサに対して行われたカスタマイズ・データ）をソース・インスタンスから新しい場所の MDS にコピーするには、prepare_clone コマンドラインの -export オプションおよび clone コマンドラインの -import オプションを使用します。第 9.4.3 項を参照してください。

クローニングおよび Oracle WebCenter Framework の詳細は、『Oracle WebCenter Framework 開発者ガイド』の WebCenter アプリケーションのクローニングに関する項を参照してください。

9.6 クローニング・プロセスのカスタマイズ

大半の場合には、デフォルトのクローニング・プロセスで十分に対応できます。さらに、次の項で説明する手動の構成手順を実行することによって、クローニング・プロセスのいくつかの側面をカスタマイズすることもできます。

- Oracle Universal Installer のパラメータの指定
- カスタム・ポートの割当て
- カスタム・データの更新

9.6.1 Oracle Universal Installer のパラメータの指定

Oracle Application Server 管理者ガイド

インスタンスをクローニングするときには、Oracle Universal Installer を直接起動しません。ただし、Oracle Universal Installer に間接的に情報を渡すことはできます。それには、通常はコマンドラインに指定する Oracle Universal Installer のパラメータを構成ファイル cs.properties で指定します。このファイルは次のディレクトリにあります。

```
(UNIX) ORACLE_HOME/clone/ias/config
(Windows) ORACLE_HOME¥clone¥ias¥config
```

たとえば、UNIX で Oracle のインベントリ・ファイルにデフォルト以外の場所を指定するには、cs.properties ファイルに次の行を追加できます。

```
clone_command_line= -invptrloc /private/oracle/oraInst.loc
```

複数の引数を指定するには、clone_command_line に、それぞれの引数を空白で区切って追加します。clone_command_line 行を追加しないでください。次の例は、Linux 上で 2 つの引数を指定する方法を示しています。

```
clone_command_line= -silent -invptrloc /private/oracle/oraInst.loc
oracle.as.j2ee.top:szl_PortListSelect="{YES,/tmp/staticports.ini}"
```

さらに、-O 文字列オプションを使用して、Oracle Universal Installer コマンドラインに渡す情報を指定できます。たとえば、次のコードでこのオプションを使用すると、Oracle Universal Installer が使用する oraparam.ini ファイルの場所を渡せます。

```
'-O-paramFile C:¥OraHome_1¥oui¥oraparam.ini'
```

9.6.2 カスタム・ポートの割当て

デフォルトでは、クローニング・スクリプトが自動的に、コンポーネントに空きポートを割り当てます。クローニング時にデフォルトのポートを割り当てるアルゴリズムは、Oracle Application Server のインストール時に使用するものと同じです。

新しい Oracle Application Server インスタンスをインストールするときに、使用するポートを staticports.ini ファイルに列挙して指定できます。次に、このファイルは、Oracle Universal Installer を呼び出すときのパラメータの値として渡されます。ポートの割当て方法および staticports.ini ファイルの使用方法の詳細は、使用しているプラットフォームの Oracle Application Server のインストレーション・ガイドを参照してください。

インスタンスをクローニングするときには、Oracle Universal Installer を直接起動しません。したがって、コマンドラインで staticports.ini ファイルを指定しても、カスタム・ポートは割り当てられません。ただし、Oracle Universal Installer に間接的にポート情報を渡すことはできます。それには、staticports.ini ファイルの場所を、次の構成ファイルで指定します。

```
(UNIX) ORACLE_HOME/clone/ias/config/cs.properties
(Windows) ORACLE_HOME¥clone¥ias¥config¥cs.properties
```

たとえば、1024 未満のポートを使用する場合は、staticports.ini ファイルでポートを指定でき、cs.properties ファイルで staticports.ini ファイルの場所を指定できます。

クローニング時にカスタム・ポートを割り当てる手順は次のとおりです。

1. staticports.ini ファイルにポート番号を列挙します。詳細は、使用しているプラットフォームの Oracle Application Server のインストレーション・ガイドを参照してください。
2. staticports.ini ファイルの場所を指定するには、cs.properties ファイルの clone_command_line に情報を追加します。たとえば Linux では、次のように指定します。

   ```
   clone_command_line= -silent oracle.as.j2ee.top:szl_
   PortListSelect="{YES,/tmp/staticports.ini}"
   ```

staticports.ini ファイルに列挙されているポートは、クローニング時に読み取られ、それにしたがって Oracle Universal Installer はポート番号を割り当てます。

UNIX で 1024 未満のポートを指定した場合は、クローニング処理中にクローン・インスタンスが起動することはありません。クローニング処理の完了後、root 権限で root.sh スクリプトを実行してから、このプロセスを開始する必要があります。

注意： デフォルトでは、Oracle Universal Installer はインストール時のすべてのユーザー入力を保存し、それをクローニング時のアクションを自動化するために使用します。その結果、ソース・インスタンスのインストール時に staticports.ini ファイルを使用した場合には、Oracle Universal Installer は、同じ staticports.ini ファイルをデフォルトで使用します。これは、インスタンスをクローニングする際に staticports.ini ファイルを指定しない場合でも同じです。この動作を取り消し、Oracle Universal Installer で新しいポートを生成するには、cs.properties ファイルに次の行を追加します。

```
oracle.as.j2ee.top:szl_PortListSelect="{¥"NO¥", ¥"¥"}"
```

9.6.3 カスタム・データの更新

デフォルトでは、クローニング・スクリプトを実行すると、Oracle ホーム内の主要な構成ファイルが更新され、そこに含まれる情報は、クローニング先の環境に対応するものとなります。第 9.4.3 項は、更新されるファイルの一部のリストです。

デフォルトのクローニング・プロセスを変更して、デフォルトでは更新されないカスタム・データを更新するようにできます。クローニング時にどのファイルを更新し、それらのファイルのどのエントリを更新するかについての情報は、別のファイルのセットに含まれており、それがクローニング・スクリプトによって読み取られます。これらのファイルを編集すると、次のことができます。

- ソース Oracle ホーム内にあって、デフォルトではクローニング時に更新されないファイルに対する変更を保存します。
- デフォルトでクローニング時に更新されるものの、通常はクローニング・プロセスによって保存されないファイルに対する変更を保存します。

これらの変更は、FileFixer という Java ユーティリティで行います。FileFixer では、正規表現と照合することでファイル内の特定のテキスト文字列が検索され、それらが新しい値に更新されます。FileFixer のパターン検索は行単位で実行されることに注意してください。複数行にわたるパターンは照合できません。

可能な変更には、次のようなものがあります。

- ファイル内のホスト名の変更

  これを行うには、ホスト名の変更が必要なファイルのパスを（Oracle ホームからの相対位置で）次のようにファイルに追加します。

  (UNIX) *ORACLE_HOME*/chgip/config/hostname.lst
  (Windows) *ORACLE_HOME*¥chgip¥config¥hostname.lst

- ファイル内に出現するすべての Oracle ホームを、古い値から新しい値に更新

  これを行うには、XML 構成ファイル fixup_script.xml.tmpl に replace 要素タグを追加します。このファイルは次のディレクトリにあります。

  (UNIX) *ORACLE_HOME*/clone/ias/config
  (Windows) *ORACLE_HOME*¥clone¥ias¥config

file_name 属性の値は、置換を行うファイルの名前と場所を指定します。たとえば、次のタグは server.xml ファイル内の Oracle ホームの値を更新します。

```
<cfw:operation>
   <replace file_name="%NEW_HOME%/j2ee/home/config/server.xml">
         <cfw:replaceCommand>
            <cfw:pattern>(%OLD_HOME%)</cfw:pattern>
            <cfw:value_ref>1</cfw:value_ref>
            <cfw:new_value>%NEW_HOME%</cfw:new_value>
         </cfw:replaceCommand>
   </replace>
</cfw:operation>
```

9.7 例 : クローニングによる Oracle Application Server クラスタの拡張

クローニングの一般的な用途は、Oracle Application Server クラスタ・トポロジのサイズの拡張です。構成とアプリケーション・デプロイが同一であり、複数の Oracle WebCenter Framework および Oracle HTTP Server 中間層で構成されるクラスタを考えてみましょう。クラスタを拡張するには、他のインスタンスと同じ構成の新しい中間層インスタンスを作成し、同じクラスタの一部とします。

この例では、次のことを前提としています。

- ソース・インスタンスがマルチキャスト動的ノード検出をベースとするクラスタのメンバーです。クローン・インスタンスは、ソース・インスタンスと同じクラスタの一部となります。
- ソース・インスタンスにはファイルベースの MDS（Oracle ホームに存在して NFS 共有ディスク上にあります）が含まれています。

クラスタ・トポロジを拡張する手順は次のとおりです。

1. 第 9.4.2 項で説明されている手順に従って、ソース・インスタンスのクローニングを準備します。
   a. 第 9.4.2 項の説明にあるように、ステップ 1 を実行します。
   b. 第 9.4.2 項のステップ 2 を実行します。

      このステップで、-export オプションを使用して、デプロイされたアプリケーションのプロデューサに対して行われたカスタマイズ・データをソース・インスタンスから .ear ファイルにエクスポートします。たとえば、次のように指定します。

      ```
      perl prepare_clone.pl ORACLE_HOME=/scratch/oracleas/Ora_10132 -export
      ```

   c. 第 9.4.2 項の説明にあるように、ステップ 3 を実行します。
2. 第 9.4.3 項で説明されている手順に従って、ソース・インスタンスをクローニングし、新しいインスタンスを作成します。
   a. 第 9.4.3 項の説明にあるように、ステップの 1、2 および 3 を実行します。
   b. 第 9.4.3 項のステップ 4 を実行します。

      このステップでは、クローン・インスタンスの名前を、コマンドラインで指定して変更する必要があります。さらに、-import オプションを指定して、prepare_clone 処理で生成された .ear ファイルをインポートする必要があります。次に例を示します。

      ```
      perl clone.pl ORACLE_HOME=/scratch/oracle/Ora_10132_B
                    ORACLE_HOME_NAME=OH_10132B
                    -instance WebC
                    -oc4jadmin_old_password my_old_admin_pass
                    -oc4jadmin_new_password my_new_admin_pass
                    -import
      ```

      この例では、クローン・インスタンスのインスタンス名は WebC です。

3. 必要に応じて、第 9.4.3 項のステップ 6 ～ 8 を実行します。
4. 必要に応じて、第 9.4.3 項のステップ 9 ～ 10 を実行します。
5. 第 9.4.3 項のステップ 11 を実行します。

第 IV 部

Secure Sockets Layer（SSL）

この部は、次の章で構成されています。

10

Oracle Application Server の Secure Sockets Layer（SSL）の概要

Oracle Application Server では、コンポーネント間でリクエストの送信とレスポンスの受信が行われます。これらのコンポーネントは、Oracle Application Server コンポーネント（Oracle HTTP Server、OC4J アプリケーション、OracleAS Single Sign-On など）またはブラウザなどの外部クライアントのいずれかです。

注意： この章では、次の Oracle Application Server 製品を参照する情報は、リリース 10.1.4、リリース 2（10.1.2）またはそれ以前のソフトウェアにのみ該当します。

- OracleAS Single Sign-On
- OracleAS Web Cache
- OracleAS Certificate Authority
- Oracle Identity Management
- OracleAS Portal

これらの通信を保護するには、SSL を使用するように Oracle Application Server を構成します。SSL は、通信を保護するための業界標準です。Oracle Application Server では SSL バージョン 2 および 3 に加えて、TLS バージョン 1 をサポートしています。

この章では、SSL の概要と、Oracle Application Server での SSL の使用方法について説明します。この章の項目は次のとおりです。

10.1 SSL の機能

SSL は、メッセージの暗号化、整合性および認証を提供することで、通信を保護します。SSL 標準により、関係するコンポーネント（ブラウザや HTTP サーバーなど）は、どの暗号化、認証および整合性メカニズムを使用するかのネゴシエーションができます。

- 暗号化を行うと、正当な受信者のみがメッセージを読めるようになります。SSL では、様々な暗号化アルゴリズムを使用してメッセージを暗号化できます。各 SSL セッションの開始時に行われる SSL ハンドシェイク中、クライアントとサーバーは、どのアルゴリズムを使用するかのネゴシエーションを行います。SSL がサポートしている暗号化アルゴリズムには、AES、RC4、3DES などがあります。
- 整合性により、クライアントが送信したメッセージが改ざんされずに正当なサーバーに届きます。メッセージの整合性を確保するために、クライアントはハッシュ機能を使用してメッセージをダイジェストにハッシュし、この**メッセージ・ダイジェスト**をサーバーに送信します。サーバーは、さらにこのメッセージをダイジェストにハッシュし、これら 2 つのダイジェストを比較します。SSL では、2 つの異なるメッセージから同じダイジェストをコンピュータで生成するのが不可能なハッシュ機能を使用しているため、サーバーでは 2 つのダイジェストが一致しない場合には、何者かによりメッセージが改ざんされたと見なすことができます。SSL がサポートしているハッシュ機能の 1 つに、SHA1 があります。
- 認証を使用すると、サーバーとクライアントは互いに相手の身元を確認できます。クライアントが SSL セッションを開始すると、サーバーは通常、サーバー自身の証明書をクライアントに送信します。証明書とは、VeriSign などの信頼できる認証局によって発行されるデジタル ID です。証明書の詳細は、第 10.5 項「証明書と Oracle Wallet」を参照してください。

  クライアントは、サーバー証明書内の証明連鎖を検証することで、サーバーが本物であることを確認します。サーバー証明書は、サーバー証明書に署名した認証局（CA）により保証されています。

  また、サーバーがクライアントの識別を認証する必要がある場合には、サーバーがクライアントに証明書の所持を要求することもできます。

10.2 秘密鍵と公開鍵の暗号化について

メッセージの整合性、認証および暗号化を提供するために、SSL では秘密鍵と公開鍵の両方の暗号化を使用します。

秘密鍵の暗号化

秘密鍵、つまり対称鍵の暗号化では、通信を保護する目的で 2 者以上が共有する 1 つの秘密鍵が必要です。この鍵は、当事者間で送信された安全なメッセージを暗号化および復号化するために使用します。これを行うには、安全な方法で各当事者に鍵を事前に配布しておく必要があります。この方法における問題点は、鍵を安全に転送し、格納することが困難なことです。

SSL では、各当事者は互いに認識している乱数を使用して秘密鍵を個別に計算します。次に、その秘密鍵を使用して暗号化したメッセージを送信します。

公開鍵の暗号化

公開鍵の暗号化では、公開鍵と秘密鍵のペアおよび安全なキーの配布方法を使用して、この問題を解決します。自由に使用可能な公開鍵は、関連する秘密鍵の保持者のみが復号化できるメッセージを暗号化するために使用します。秘密鍵は、他のセキュリティ資格証明とともに、Oracle Wallet などの暗号化されたコンテナ内に安全に格納されます。

公開鍵のアルゴリズムでは、メッセージの秘密は保証されますが、安全な通信は必ずしも保証されません。その理由は、通信者間の識別が検証されないためです。安全な通信を確立するには、メッセージの暗号化に使用される公開鍵が相手の受信者に実際に属していることを確認することが重要です。そうしないと、第三者が通信を傍受し、公開鍵のリクエストに割り込み、正当な鍵を独自の公開鍵に置き換えることが可能になります（**介在者**攻撃）。

このような攻撃を避けるためには、公開鍵の所有者を確認する必要があります。これは認証と呼ばれるプロセスです。認証は、認証局（CA）を介して行うことができます。CA は、両通信者間によって信頼されている第三者です。

CA は、エンティティの名前、公開鍵および他のセキュリティ資格証明を含む公開鍵証明書を発行します。通常、このような資格証明には、CA 名、CA の署名および証明書の有効日（開始日、終了日）が含まれています。

CA では独自の秘密鍵を使用してメッセージを暗号化します。一方、そのメッセージの復号化には公開鍵が使用されるため、メッセージが CA によって暗号化されたものであるかどうかが確認されます。CA 公開鍵は広く一般に知られているため、アクセスするたびに認証する必要はありません。このような CA 公開鍵は Wallet に格納されます。

10.3 SSL セッションの設定方法（SSL ハンドシェイク）

SSL プロトコルには、ハンドシェイク・フェーズとデータ転送フェーズという 2 つのフェーズがあります。ハンドシェイク・フェーズでは、サーバーおよびオプションでクライアントを認証し、データ転送フェーズで転送されるデータを保護するための暗号化鍵を設定します。

クライアントがサーバーへの SSL 接続を要求すると、クライアントとサーバーはまずハンドシェイク・フェーズでメッセージを交換します（一般的なシナリオとしては、`http://` ではなく `https://` プロトコルを使用して、サーバーからページを要求するブラウザがあります。HTTPS プロトコルは、HTTP で SSL を使用することを示します）。

図 10-1 に、Web サーバーとブラウザ間の一般的な SSL 接続用ハンドシェイク・メッセージを示します。この図では次の手順を示しています。

1. クライアントは、サーバーにハロー・メッセージを送信します。

   メッセージには、クライアントがサポートしているアルゴリズムの一覧および鍵を生成するための乱数が含まれています。

2. サーバーは、クライアントにハロー・メッセージを送信してレスポンスを返します。このメッセージには、次の内容が含まれています。

   - 使用するアルゴリズム。これは、クライアントが送信した一覧から、サーバーによって選択されます。
   - 鍵の生成に使用する乱数。

3. サーバーは、クライアントに証明書を送信します。
4. クライアントがサーバーの証明書を使用してサーバーを認証します。
5. クライアントが乱数（プリマスタ・シークレット）を生成し、サーバーの公開鍵を使用して暗号化し、サーバーに送信します。
6. サーバーは、秘密鍵を使用してメッセージを復号化し、プリマスタ・シークレットを取得します。
7. クライアントとサーバーは、SSL セッションで使用される鍵を個別に計算します。

   これらの鍵は、互いに認識しているプリマスタ・シークレットと乱数に基づいて計算されるため、それぞれの相手には送信されません。鍵には次の内容が含まれています。

   - クライアントがサーバーへの送信前にデータを暗号化するために使用する暗号化鍵
   - サーバーがクライアントへの送信前にデータを暗号化するために使用する暗号化鍵
   - クライアントがデータのメッセージ・ダイジェストを作成するために使用する鍵
   - サーバーがデータのメッセージ・ダイジェストを作成するために使用する鍵

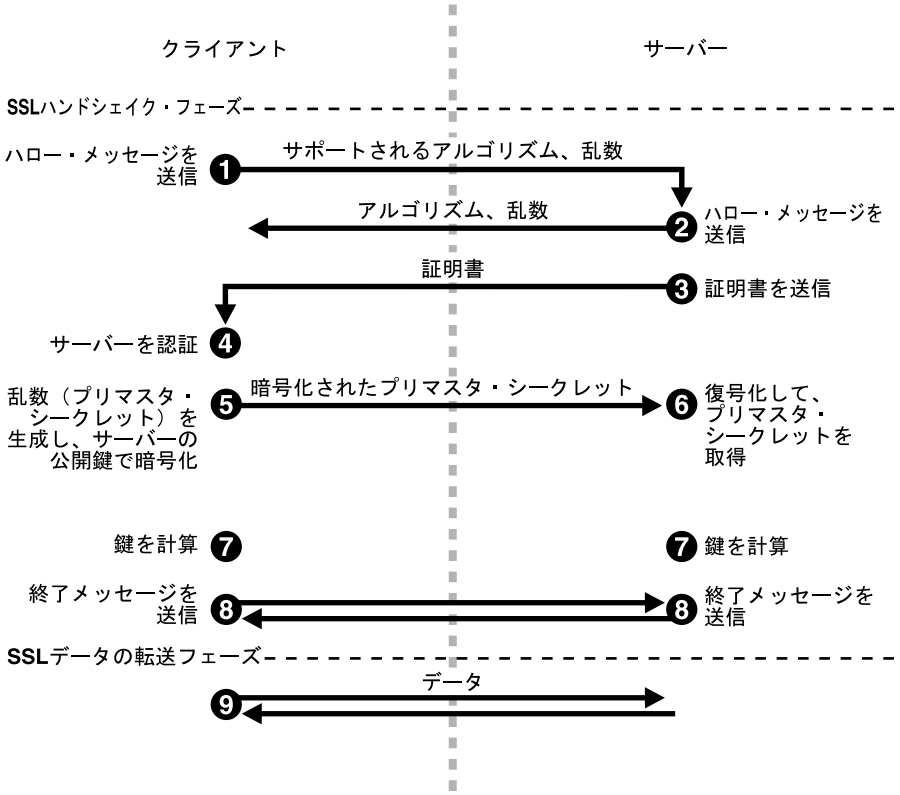
   暗号化鍵は、対称的です。つまり、データの暗号化と復号化には同じ鍵が使用されます。

8. クライアントとサーバーは、相互に終了メッセージを送信します。これらは前の手順で生成した鍵を使用して送信される最初のメッセージ（最初の安全なメッセージ）です。

   終了メッセージには、各当事者が送信した以前のハンドシェイク・メッセージがすべて含まれています。各当事者は、受信した以前のメッセージが、終了メッセージに含まれているメッセージに一致するかどうかを確認します。これは、ハンドシェイク・メッセージが改ざんされていないことを確認するためです。

9. クライアントとサーバーは、暗号化鍵とハッシュ鍵およびアルゴリズムを使用してデータを転送します。

図 10-1 SSL ハンドシェイク



10.4 Oracle Application Server で SSL を使用するための要件

Oracle Application Server で SSL を使用する手順は次のとおりです。

- サイトの証明書と Oracle Wallet が必要です。この証明書は、偽りのサイトに接続していないことを確認するためにクライアントが使用します。
- クライアントを認証する必要がある場合は、クライアントにも証明書が必要です。
- SSL を介してメッセージを受け入れ、転送できるように Oracle Application Server のコンポーネント（Oracle HTTP Server など）を構成する必要があります。
- SSL はリソースを消費します。SSL トラフィックの負荷が高くなることが予想される場合は、SSL アクセラレータの使用を検討してください。

この章の次の項で、これらのトピックについて詳しく説明します。

10.5 証明書と Oracle Wallet

サイトには証明書が必要です。SSL 通信を要求しているサイトにクライアントが接続する場合、接続先のサイトでは証明書をクライアントに送信し、クライアントがサイトを認証できるようにする必要があります。

Oracle Application Server では、X.509 V3 証明書および PKIX 標準（RFC 3280）に準拠する証明書をサポートしています。

10.5.1 証明書の取得方法

証明書は認証局（CA）から取得します。CA は、独自の秘密鍵を使用して発行した証明書に署名する、信頼されたエンティティです。クライアントは（CA の公開鍵を使用して）証明書の発行者を確認できます。CA の例には、VeriSign（`http://www.verisign.com`）や Thawte（`http://www.thawte.com`）などがあります。

Oracle Application Server にも、OracleAS Certificate Authority（OCA）と呼ばれる認証局があります。OCA を使用すると、独自の認証局を設定できます。詳細は、『Oracle Application Server Certificate Authority 管理者ガイド』を参照してください。

証明書を取得するには、証明書リクエストを CA に送信します。証明書リクエストには、独自の公開鍵を含む情報が含まれています。証明書リクエストを生成するには、ツールを使用できます。これらのツールは、秘密鍵と公開鍵のペアを生成します。証明書リクエストを生成できるツールには、Oracle Wallet Manager や Sun 社の `keytool`（OC4J 専用）などがあります。Oracle Wallet Manager の詳細は、第 11 章「Wallet と証明書の管理」を参照してください。

証明書には、他の項目とともに、次のデータが含まれています。

- 証明書の所有者名
- 証明書の所有者の公開鍵
- CA 名
- 証明書の有効期限
- 証明書のシリアル番号

証明書は、期限が切れるか、取り消されるまで有効です。

OracleAS Certificate Authority（OCA）を使用してサーバーの証明書を作成した場合、ほとんどのブラウザではブラウザのユーザーからの入力がないかぎり、これらの証明書を受け入れません。これは、ほとんどのブラウザは特定の CA からの証明書のみを受け入れるように事前に構成されており、OCA はその中に含まれていないためです。ブラウザでは、ユーザーがサーバーからの証明書を受け入れるか、CA の証明書をインポートしないかぎり、サーバーからの証明書を拒否します。

この問題は、CA の証明書がブラウザにインポートされるまで、すべての CA に対して発生します。詳細は、『Oracle Application Server Certificate Authority 管理者ガイド』を参照してください。

10.5.2 Oracle Wallet

Oracle Wallet は、証明書、証明書リクエスト、秘密鍵などの資格証明を格納するコンテナです。Oracle Wallet は、ファイル・システムまたは Oracle Internet Directory などの LDAP ディレクトリに格納できます。Oracle Wallet は、パスワードで保護されています。

Oracle Wallet は、Oracle Wallet Manager を使用して管理します。Oracle Wallet Manager では、Oracle Wallet の作成、証明書リクエストの作成、Wallet への証明書のインポート、LDAP ディレクトリへの Wallet のアップロードなどのタスクを実行できます。

Oracle Wallet Manager は、PKCS #11 と PKCS #12 の Wallet をサポートしています。

- ケース 1: Oracle Wallet Manager を使用して証明書リクエストを生成し、秘密鍵をファイル・システムに格納することにしました。CA から証明書を取得するとき、それを Oracle Wallet にインポートできます。この Wallet は、PKCS #12 形式を使用します。詳細は、第 11.1.4.2.1 項「標準 Wallet の作成」を参照してください。

- ケース 2: Oracle Wallet Manager を使用して証明書リクエストを生成し、秘密鍵をハードウェア・セキュリティ・モジュールに格納することにしました。CA から証明書を取得するとき、それを Oracle Wallet にインポートできます。この Wallet は、PKCS #11 形式を使用します。詳細は、第 11.1.4.2.2 項「ハードウェア・セキュリティ・モジュールに資格証明を格納する Wallet の作成」を参照してください。
- ケース 3: PKCS #12 形式を使用する証明書が Wallet にすでに存在し、これを Oracle Application Server で使用することを予定しています。Wallet はサード・パーティのツールを使用して作成されています。この場合は、Wallet の作成に使用したツールで、Wallet をファイル・システム上のファイルにエクスポートします。次に、Wallet をインポートします。詳細は、第 11.1.5.1.3 項「サード・パーティ製ツールによって作成された証明書のインポート」を参照してください。

Oracle Wallet を使用するコンポーネント

SSL サーバーとして機能する Oracle Application Server コンポーネントには、Oracle Wallet が必要です（Wallet にはサーバーが使用する証明書がすでに含まれています）。これらのコンポーネントには、Oracle HTTP Server、OracleAS Web Cache、OPMN、Oracle Internet Directory、ポート・トンネリング・デーモン（`iaspt`）などがあります。

コンポーネントは、Oracle Wallet の場所を使用して構成します。たとえば、SSL 対応の Oracle HTTP Server を構成するには、`SSL Wallet` ディレクティブを使用して Wallet の場所を指定します。コンポーネントの Wallet の場所を指定する方法については、コンポーネントのマニュアルを参照してください。

注意： OC4J コンポーネントでは、Oracle Wallet ではなくキーストアを使用して証明書を格納します。証明書をキーストアにインポートするには、`keytool` というツールを使用します。キーストアと `keytool` の詳細は、『Oracle Containers for J2EE セキュリティ・ガイド』を参照してください。

10.5.3 クライアント証明書

クライアントを認証する必要がある場合は、クライアントに証明書の送信を要求するように Oracle HTTP Server を構成します。クライアントは CA から証明書を取得することもできます。

クライアントが Oracle のコンポーネントである場合（たとえば OracleAS Web Cache は Oracle HTTP Server と通信する際にクライアントとして機能できる）、クライアントのコンポーネントは、クライアントの証明書を Oracle Wallet に格納できます。OPMN が SSL に対応するように構成されているときは、OPMN もクライアントして機能します。

クライアントがブラウザである場合、クライアントには Oracle Wallet は必要ありません。証明書はブラウザにインポートできます。

SOAP や Web サービス・クライアントなどの他の種類のクライアントには、証明書と証明書ストアを構成する独自の方法があります。

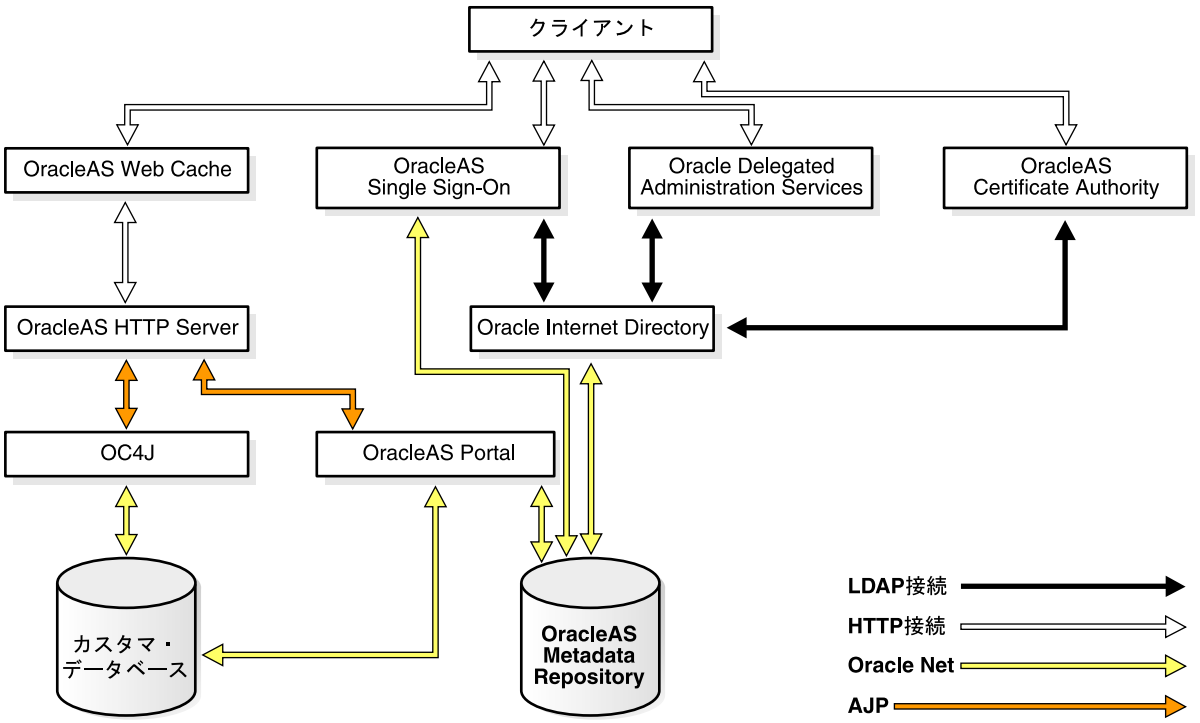
10.6 SSL 構成の概要

Oracle Application Server のコンポーネントが SSL を使用できるようにするには、Application Server Control を使用します。場合によっては、構成ファイルを手動で編集します。

SSL は、クライアントとサーバー間の通信を保護します。たとえばクライアント・ブラウザ、OracleAS Web Cache、Oracle HTTP Server および OC4J など、3 者以上が関与している場合は、すべてのコンポーネントで SSL を使用するように構成する必要があります。

図 10-2 に、Oracle Application Server コンポーネントと、これらのコンポーネントが使用するプロトコル間の一般的な通信パスを示します。たとえば、ブラウザは HTTP を使用して OracleAS Web Cache と通信し、Oracle HTTP Server は AJP を使用して OC4J と通信します。これらすべてのプロトコルは、SSL に対応しています。

図 10-2 Oracle Application Server のコンポーネント間の通信パス



10.6.1 デフォルトの SSL 構成

Oracle Application Server のインストールでデフォルトのオプションを選択した場合は、いずれのコンポーネントも SSL に対応するように構成されていません。

Oracle Internet Directory のホストとポートを指定するインストール画面には、「Oracle Internet Directory には SSL 接続のみ使用」というオプションがあります。このオプションを選択した場合は、Oracle Internet Directory の SSL ポート番号を指定する必要があります。インストーラは、実行時に SSL のみを Oracle Internet Directory との通信に使用するようにコンポーネントを構成します。

10.6.2 部分的な SSL 構成

Oracle Application Server では、保護するパスに対してのみ SSL を構成できます。図 10-2 に示すように、コンポーネントが使用するパスは数多くあります。

次のような理由から、すべてのパスを保護することは好ましくない場合があります。

- SSL はリソースを消費します。SSL トラフィックの負荷が高い場合は、SSL 処理を SSL アクセラレータに委託する必要があります。詳細は、第 10.7 項「ハードウェア・セキュリティ・モジュールとの統合」を参照してください。
- コンピュータがファイアウォールの内側にある場合は、外部からアクセスされるパスのみを保護する必要があります。たとえば、外部が OracleAS Web Cache と Oracle HTTP Server にのみアクセスできる場合には、これらのパスのみを保護する必要があります。

10.7 ハードウェア・セキュリティ・モジュールとの統合

クライアントが SSL を使用してサイトに接続すると、SSL に必要な追加の処理によってサイトのサーバーの負荷が大きくなり、サイト全体（SSL および SSL 以外の接続）のパフォーマンスとスループットが低下します。このような場合は、SSL アクセラレータ・ハードウェアに SSL の計算を委託して、パフォーマンスを改善することを考慮してください。

SSL アクセラレータのタイプ :

- 第 10.7.1 項「プロトコル・コンバータ」
- 第 10.7.2 項「演算アクセラレータ（PKCS #11 の統合）」

10.7.1 プロトコル・コンバータ

プロトコル・コンバータは、HTTPS トラフィックを HTTP に変換します。プロトコル・コンバータは、スタンドアロンのハードウェア・マシンです。Oracle Application Server では次のような企業のプロトコル・コンバータをサポートしています。

- F5（`http://www.f5.com`）
- Cisco（`http://www.cisco.com`）
- SonicWall（`http://www.sonicwall.com`）

注意 : プロトコル・コンバータへの SSL 接続は、プロトコル・コンバータで終了します。コンバータがリクエストを Oracle Application Server に転送するとき、そのほとんどは**暗号化されない**状態で転送されます。

SSL を使用してリクエストを Oracle Application Server に転送するプロトコル・コンバータでも、プロトコル・コンバータを使用しない場合より依然として高速です。プロトコル・コンバータを使用すると、高負荷な SSL の鍵交換操作のほとんどが不要になるためです。

10.7.2 演算アクセラレータ（PKCS #11 の統合）

演算アクセラレータは、SSL が使用する数式演算の速度を向上します。このようなデバイスは、通常は（TCP/IP を経由して）サーバーに接続します。また、多くの場合、このようなデバイスには鍵の管理や安全なキーストアなどの追加機能が備わっています。

Oracle Application Server では、PKCS #11 標準に準拠している演算アクセラレータをサポートしています。認定済アクセラレータの一覧は、Oracle*MetaLink* サイト `http://www.oracle.com/support/metalink/index.html` を参照してください。

11

Wallet と証明書の管理

この章では、Oracle Application Server のリソースに対するセキュリティ資格証明を取得および管理する方法を説明します。セキュリティ管理者は、Oracle Wallet Manager およびそのコマンドライン・ユーティリティ `orapki` を使用して、Oracle クライアントおよびサーバー上の公開鍵インフラストラクチャ（PKI）の管理を行います。これらのツールで作成した資格証明は、Oracle Database、Oracle Application Server および Oracle Identity Management インフラストラクチャによる読込みが可能です。

注意： この章では、次の Oracle Application Server 製品を参照する情報は、リリース 10.1.4、リリース 2（10.1.2）またはそれ以前のソフトウェアにのみ該当します。

- Oracle Identity Management
- Oracle Internet Directory

この章の項目は次のとおりです。

注意： 割り当てられた証明書がすでにある場合は、次の項をお読みください。

第 11.1.2 項「Oracle Wallet Manager の起動」

第 11.3 項「X.509 証明書との相互運用性」

11.1 Oracle Wallet Manager の使用

この項では、PKI 証明書の管理に使用するグラフィカル・ユーザー・インタフェース・ツールである Oracle Wallet Manager について説明します。この項の項目は次のとおりです。

11.1.1 Oracle Wallet Manager の概要

Oracle Wallet Manager は、Oracle Wallet 内のセキュリティ資格証明を管理および編集するためのアプリケーションです。Wallet はパスワードで保護されたコンテナで、強固な認証のために SSL で必要な秘密鍵、証明書、信頼できる証明書など、認証および署名用の資格証明を格納します。Oracle Wallet Manager を使用して実行できるタスクは次のとおりです。

- Wallet の作成
- 証明書リクエストの生成
- PKI ベースのサービスにアクセスするために Wallet を開く
- 公開鍵暗号規格 #11 仕様準拠の API を使用した、ハードウェア・セキュリティ・モジュールへの資格証明の保存（PKCS #11 を参照）
- LDAP ディレクトリへの Wallet のアップロードおよび LDAP ディレクトリからのダウンロード
- Oracle 環境で使用するための、サード・パーティの PKCS #12 形式の Wallet のインポート
- サード・パーティ環境への Oracle Wallet のエクスポート

Oracle Wallet Manager の機能について、次の各項目で説明します。

11.1.1.1 Wallet のパスワードの管理

Oracle Wallet は、パスワードで保護されています。Oracle Wallet Manager には、強化された Wallet パスワード管理モジュールが含まれており、次のパスワード管理ポリシー・ガイドラインが強制されます。

- パスワードの最小文字数（8 文字）
- パスワード最大長は無制限
- 英数字の混在が必須

11.1.1.2 強度の高い Wallet 暗号化

Oracle Wallet Manager には、X.509 証明書に関連付けられる秘密鍵が格納され、Triple-DES 暗号化が使用されます。

11.1.1.3 Microsoft Windows レジストリへの Wallet の格納

Oracle Wallet Manager では、Microsoft Windows システム・レジストリのユーザー・プロファイル領域または Windows ファイル管理システムに複数の Oracle Wallet を格納できます。Wallet をレジストリ内に保存することの利点は次のとおりです。

- **より優れたアクセス制御。** レジストリのユーザー・プロファイル領域に格納される Wallet には、関連付けられたユーザー以外はアクセスできません。したがって、システムのユーザー・アクセスを制御することは、Wallet のアクセスを制御することにもなります。さらに、ユーザーがシステムからログアウトすると、そのユーザーの Wallet へのアクセスは不可能になります。
- **容易な管理。** Wallet は特定のユーザー・プロファイルと関連付けられているため、ファイル権限を管理する必要がなく、プロファイルに格納されている Wallet はユーザー・プロファイルが削除されると自動的に削除されます。Oracle Wallet Manager は、Wallet をレジストリ内で作成および管理するために使用できます。

11.1.1.3.1 サポートされるオプション

- レジストリから Wallet を開きます。
- レジストリに Wallet を保存します。
- レジストリの他の場所に別名保存します。
- レジストリから Wallet を削除します。
- ファイル・システムから Wallet を開き、レジストリに保存します。
- レジストリから Wallet を開き、ファイル・システムに保存します。

11.1.1.4 下位互換性

Oracle Wallet Manager には、リリース 8.1.7 のデータベースとの間の下位互換性があります。

11.1.1.5 サード・パーティの Wallet のサポート

Oracle Wallet Manager では、次のサード・パーティ製アプリケーションからの PKI 資格証明を使用できます。

- Microsoft Internet Explorer 5.0 以降
- Netscape Communicator 4.7.2 以降
- OpenSSL

Microsoft Internet Explorer および Netscape からのブラウザの PKI 資格証明ストアには、ユーザー証明書が格納されています。この証明書には、サブジェクトの公開鍵と ID、および関連する信頼できる証明書が含まれています。このような資格証明を使用するには、サード・パーティ環境から資格証明をエクスポートして PKCS #12 形式で保存する必要があります。その後、この資格証明を Oracle Wallet Manager で開くと、SSL で使用できるようになります。

> **関連項目：** 第 11.1.5.1.3 項「サード・パーティ製ツールによって作成された証明書のインポート」

11.1.1.6 LDAP ディレクトリのサポート

Oracle Wallet Manager では、LDAP 準拠ディレクトリとの間で Wallet のアップロードや取得ができます。集中化された LDAP 準拠ディレクトリに Wallet を格納すると、ユーザーは複数の場所やデバイスから Wallet にアクセスできるので、一貫性があり信頼性の高いユーザー認証が行われるようになります。また Wallet のライフ・サイクルを通して、集中的な Wallet 管理が可能になります。機能する Wallet を誤って上書きするのを防ぐため、インストールされた証明書を含む Wallet のみをアップロードできます。

Oracle Wallet Manager を使用してユーザーの Wallet をアップロードまたはダウンロードするには、LDAP ディレクトリ内のディレクトリ・ユーザー・エントリを事前に定義および構成しておく必要があります。ディレクトリに Oracle8*i* 以前のユーザーが含まれている場合、Wallet のアップロードやダウンロードの機能を使用できるよう、初めて使用する際にユーザーが自動的にアップグレードされます。

Oracle Wallet Manager では、単純なパスワードを使用して LDAP ディレクトリに接続し、ユーザー Wallet をダウンロードします。ただし、開いた Wallet に SSL Oracle PKI 証明書使用による証明書が含まれている場合、アップロードには SSL 接続が使用されます。SSL 証明書が Wallet 内に存在しない場合は、パスワードによる認証が行われます。

注意： ディレクトリ・パスワードと Wallet パスワードは独立しているので、異なっていてもかまいません。これらのパスワードは、一貫して別々に管理することをお薦めします。一方のパスワードからもう一方を類推できないようにしてください。

関連項目：

- 第 11.1.4.7 項「LDAP ディレクトリへの Wallet のアップロード」
- 第 11.1.4.8 項「LDAP ディレクトリからの Wallet のダウンロード」
- 第 11.3.2 項「複数の証明書のサポート」

11.1.2 Oracle Wallet Manager の起動

Oracle Wallet Manager を起動する手順は次のとおりです。

- Windows の場合：「**スタート**」→「**プログラム**」→「**Oracle - *Oracle_Home_Name***」→「**Integrated Management Tools**」→「**Wallet Manager**」を選択します。
- UNIX の場合：コマンドラインに `owm` と入力します。

11.1.3 完全な Wallet の作成方法：プロセスの概要

Wallet は、ピアの証明書の検証に必要なユーザー証明書および様々なトラスト・ポイントをセキュアに格納する必須のリポジトリです。

完全な Wallet を作成するプロセスの概要を次に示します。

1. Oracle Wallet Manager を使用して新しい Wallet を作成します。
   - Wallet のパスワードの作成は、第 11.1.4.1 項「Wallet のパスワード作成に必要なガイドライン」を参照してください。
   - 標準の Wallet（資格証明をファイル・システム上に格納）およびハードウェア・セキュリティ・モジュール Wallet の作成方法は、第 11.1.4.2 項「新しい Wallet の作成」を参照してください。
2. 証明書リクエストを生成します。Oracle Wallet Manager を使用して新しい Wallet を作成するときに、証明書リクエストの作成を求めるメッセージが表示されます。証明書リクエストの作成方法は、第 11.1.5.1.1 項「証明書リクエストの追加」を参照してください。
3. 使用する CA に、証明書リクエストを送信します。証明書リクエストのテキストをコピーして電子メール・メッセージに貼り付けるか、証明書リクエストをファイルにエクスポートします。詳細は、第 11.1.5.1.7 項「ユーザー証明書リクエストのエクスポート」を参照してください。証明書リクエストは Wallet の一部となります。証明書リクエストは、関連する証明書が削除されるまで存在している必要があることに注意してください。

4. 署名付きユーザー証明書および関連する信頼できる証明書が CA から送信されたら、これらの証明書を次の順序でインポートします（PKCS #7 形式のユーザー証明書および信頼できる証明書は同時にインポートできます）。

   - 初めに、CA の信頼できる証明書を Wallet にインポートします。詳細は、第 11.1.5.2.1 項「信頼できる証明書のインポート」を参照してください。新しいユーザー証明書の発行元である CA の信頼できる証明書がデフォルトで Oracle Wallet Manager 内に存在する場合は、この手順を省略できます。
   - 信頼できる証明書が正常にインポートされたら、次に、CA から送信されたユーザー証明書を Wallet にインポートします。詳細は、第 11.1.5.1.2 項「Wallet へのユーザー証明書のインポート」を参照してください。

   注意： ほとんどの認証局で使用されている、BASE64 エンコードの PKCS #7 形式では、一般に、次のヘッダー行とフッター行が使用されています。

   ```
   -----BEGIN PKCS7-----
   -----END PKCS7-----
   ```

   通常の証明書には、次のヘッダー行とフッター行があります。

   ```
   -----BEGIN CERTIFICATE-----
   -----END CERTIFICATE-----
   ```

   ただし、認証局によっては、PKCS #7 形式の証明書の中で BEGIN CERTIFICATE および END CERTIFICATE のヘッダー行およびフッター行も使用しています。PKCS #7 形式の証明書がインポートされるときに、認証局の証明書は信頼できる証明書としてインポートされます。

   認証局の証明書がないユーザー証明書をインポートすると、Oracle Wallet Manager では、ユーザー証明書を発行した認証局の証明書を要求するメッセージが表示されます。

   ---

5. （オプション）Wallet の自動ログイン機能を設定します。詳細は、第 11.1.4.14 項「自動ログインの使用」を参照してください。

   これはパスワードを使用せずに PKI ベースでサービスにアクセスするための機能ですが、通常は、ほとんどの Wallet に必要です。データベース・サーバーおよびクライアントの Wallet には必須です。起動の時点で Wallet パスワードを受け取る製品の場合にのみ、これを省略できます。

ここに示したプロセスが完了すると、ユーザー証明書とそのトラスト・ポイントが含まれた Wallet が完成します。

11.1.4 Wallet の管理

この項では、新しい Wallet の作成方法と、関連する Wallet 管理タスクの実行方法を説明します。この項の項目は次のとおりです。

- Wallet のパスワード作成に必要なガイドライン
- 新しい Wallet の作成
- 既存の Wallet を開く
- Wallet を閉じる
- サード・パーティ環境への Oracle Wallet のエクスポート
- PKCS #12 をサポートしていないツールへの Oracle Wallet のエクスポート
- LDAP ディレクトリへの Wallet のアップロード
- LDAP ディレクトリからの Wallet のダウンロード

- 変更の保存
- 開いている Wallet の新しい場所への保存
- システムのデフォルトへの保存
- Wallet の削除
- パスワードの変更
- 自動ログインの使用

11.1.4.1 Wallet のパスワード作成に必要なガイドライン

Oracle Wallet には、ユーザーを複数のデータベースについて認証する際に使用できるユーザー資格証明が含まれているので、強度の高い Wallet パスワードを選択することが重要です。悪意のあるユーザーが、Wallet パスワードを推測し、Wallet 所有者がアクセス権を持つすべてのデータベースにアクセスすることもあり得ます。

パスワードは、数字または特殊文字を含む 8 文字以上のアルファベットで構成する必要があります。

注意： 「admin0」、「oracle1」、「2135551212A」のような、ユーザーの名前、電話番号、政府発行の ID 番号に由来した、容易に推測可能なパスワードは選択しないことを強くお薦めします。これは、不正アクセス者が個人情報からユーザー・パスワードを推測するのを防ぐためです。また、毎月または 3 か月に 1 回程度、定期的にパスワードを変更することもセキュリティ上重要な習慣です。

パスワードを変更した場合、自動ログイン Wallet を再生成する必要があります。

関連項目：

- 第 11.1.1.1 項「Wallet のパスワードの管理」
- 第 11.1.4.14 項「自動ログインの使用」

11.1.4.2 新しい Wallet の作成

Oracle Wallet Manager を使用すると、PKCS #12 Wallet（デフォルトの標準タイプの Wallet）を作成できます。この Wallet では、資格証明はファイル・システム上のディレクトリに格納されます。また、PKCS #11 Wallet も作成できます。これは、資格証明をサーバーのハードウェア・セキュリティ・モジュールに格納する、または秘密鍵をクライアントのトークンに格納する Wallet です。次の各項では、Oracle Wallet Manager を使用してこれらのタイプの Wallet を作成する方法を説明します。

11.1.4.2.1 標準 Wallet の作成 ハードウェア・セキュリティ・モジュール（PKCS #11 のデバイス）がない場合は、ファイル・システム上のディレクトリに資格情報を格納する標準 Wallet を使用してください。

標準 Wallet を作成するには、次の作業を実行します。

1. メニュー・バーから「**ウォレット**」→「**新規**」を選択します。「新規ウォレット」ダイアログ・ボックスが表示されます。
2. 第 11.1.4.1 項「Wallet のパスワード作成に必要なガイドライン」に従って、「**ウォレット・パスワード**」フィールドにパスワードを入力します。このパスワードは、ユーザーの資格証明が無許可で使用されるのを防ぎます。
3. 「**パスワードの確認**」フィールドにパスワードを再入力します。
4. 「**ウォレット・タイプ**」リストで、「**標準**」を選択します。

5. 次に進むには「**OK**」をクリックします。入力したパスワードが必要なガイドラインに準拠していない場合は、次のメッセージが表示されます。

```
Password must have a minimum length of eight characters,
and contain alphabetic characters combined with numbers
or special characters.
Do you want to try again?
```

6. 空の Wallet が新しく作成されたことを示す警告が表示されます。証明書リクエストを追加するかどうかを尋ねられます。詳細は、第 11.1.5.1.1 項「証明書リクエストの追加」を参照してください。

   「**いいえ**」を選択すると、Oracle Wallet Manager のメイン・ウィンドウに戻ります。作成したばかりの新しい Wallet が左のウィンドウに表示されます。証明書のステータスは「**空**」で、Wallet はデフォルトの信頼できる証明書を表示します。

7. 「**ウォレット**」→「**システム・デフォルトに保存**」を選択して、新しい Wallet を保存します。

   システムのデフォルトに Wallet を保存する権限がない場合は、別の場所に Wallet を保存します。クライアントおよびサーバーの SSL 構成で、この場所を使用します。

   Wallet が正常に保存されたことを確認するメッセージが、ウィンドウの一番下に表示されます。

11.1.4.2.2 ハードウェア・セキュリティ・モジュールに資格証明を格納する Wallet の作成

PKCS #11 準拠のハードウェア・セキュリティ・モジュールに資格証明を格納する Wallet を作成するには、次のタスクを実行します。

1. メニュー・バーから「**ウォレット**」→「**新規**」を選択します。「新規ウォレット」ダイアログ・ボックスが表示されます。
2. 第 11.1.4.1 項「Wallet のパスワード作成に必要なガイドライン」に従って、「**ウォレット・パスワード**」フィールドにパスワードを入力します。
3. 「**パスワードの確認**」フィールドにパスワードを再入力します。
4. 「**ウォレット・タイプ**」リストで「**PKCS11**」を選択し、「**OK**」をクリックして次に進みます。新規 PKCS11 Wallet ウィンドウが表示されます。
5. 「**ハードウェア・ベンダーの選択**」リストで、ベンダー名を選択します。

   注意： Oracle Wallet Manager の現行リリースでは、Oracle Wallet との相互運用が保証されているのは nCipher ハードウェアのみです。

6. 「**PKCS11 ライブラリのファイル名**」フィールドに、PKCS11 ライブラリが格納されているディレクトリへのパスを入力します。または、「**参照**」をクリックして、ファイル・システム上でそのディレクトリを検索します。
7. 「**スマートカードのパスワード**」を入力して、「**OK**」を選択します。

   スマートカードのパスワード（Wallet のパスワードとは別）は、Wallet の中に格納されます。

8. 空の Wallet が新しく作成されたことを示す警告が表示されます。証明書リクエストを追加するかどうかを尋ねられます。詳細は、第 11.1.5.1.1 項「証明書リクエストの追加」を参照してください。

   「**いいえ**」を選択すると、Oracle Wallet Manager のメイン・ウィンドウに戻ります。作成したばかりの新しい Wallet が左のウィンドウに表示されます。証明書のステータスは「**空**」で、Wallet はデフォルトの信頼できる証明書を表示します。

9. 「**ウォレット**」→「**システム・デフォルトに保存**」を選択して、新しい Wallet を保存します。

   システムのデフォルトに Wallet を保存する権限がない場合は、別の場所に Wallet を保存します。

   Wallet が正常に保存されたことを確認するメッセージが、ウィンドウの一番下に表示されます。

注意： スマートカードのパスワードを変更した場合や、PKCS #11 ライブラリを移動した場合は、Wallet を開こうとするときにエラー・メッセージが表示されます。このとき、スマートカードの新しいパスワードまたはライブラリの新しいパスを入力するように要求されます。

11.1.4.3 既存の Wallet を開く

ファイル・システム・ディレクトリにすでに存在する Wallet を開くには、次の手順に従います。

1. メニュー・バーから「**ウォレット**」→「**開く**」を選択します。「ディレクトリの選択」ダイアログ・ボックスが表示されます。
2. Wallet が格納されたディレクトリの場所へナビゲートし、ディレクトリを選択します。
3. 「**OK**」を選択します。「ウォレットを開く」ダイアログ・ボックスが表示されます。
4. 「**ウォレット・パスワード**」フィールドに、Wallet のパスワードを入力します。
5. 「**OK**」を選択します。

   メイン・ウィンドウに戻り、Wallet が開かれたことを示すメッセージがウィンドウの一番下に表示されます。Wallet の証明書と信頼できる証明書が左のウィンドウに表示されます。

11.1.4.4 Wallet を閉じる

現在選択されているディレクトリで開いている Wallet を閉じるには、次の手順に従います。

「**ウォレット**」→「**閉じる**」を選択します。

ウィンドウの一番下にメッセージが表示され、Wallet が閉じたことを確認できます。

11.1.4.5 サード・パーティ環境への Oracle Wallet のエクスポート

Oracle Wallet Manager では、Wallet をサード・パーティ環境にエクスポートできます。

Wallet をサード・パーティ環境にエクスポートするには：

1. Oracle Wallet Manager を使用して、Wallet ファイルを保存します。
2. サード・パーティ製品ごとのインポート手順に従って、Oracle Wallet Manager で作成したオペレーティング・システムの PKCS #12 Wallet ファイルをインポートします（UNIX および Windows プラットフォームでは `ewallet.p12` と呼ばれます）。

注意：

- Oracle Wallet Manager は、1 つの Wallet に対する複数の証明書をサポートしていますが、現時点では、ブラウザでサポートされているのは一般に単一証明書の Wallet のインポートのみです。これらのブラウザでは、単一鍵ペアを持つ Oracle Wallet をエクスポートする必要があります。
- Oracle Wallet Manager では、Netscape Communicator 4.7.2 以降、OpenSSL、および Microsoft Internet Explorer 5.0 以降への Wallet のエクスポートのみをサポートします。

11.1.4.6 PKCS #12 をサポートしていないツールへの Oracle Wallet のエクスポート

PKCS #12 をサポートしていないツールに Wallet を追加する場合は、テキストベースの PKI フォーマットに Wallet をエクスポートします。個別のコンポーネントは表 11-1 に示す標準に従ってフォーマットされます。Wallet 内では、SSL 鍵を使用する証明書だけが Wallet でエクスポートされます。

Wallet をテキストベースの PKI フォーマットにエクスポートするには :

1. 「**操作**」→「**ウォレットのエクスポート**」を選択します。「Wallet のエクスポート」ダイアログ・ボックスが表示されます。
2. Wallet のエクスポート先になるファイル・システム・ディレクトリを入力するか、「**フォルダ**」の下のディレクトリ構造にナビゲートします。
3. Wallet のエクスポート先ファイルの名前を入力します。
4. 「**OK**」を選択して、メイン・ウィンドウに戻ります。

表 11-1 PKI Wallet のエンコーディング規格

コンポーネント	エンコーディング規格
証明連鎖	X509v3
信頼できる証明書	X509v3
秘密鍵	PKCS #8

11.1.4.7 LDAP ディレクトリへの Wallet のアップロード

指定された Wallet に SSL 証明書がある場合、Oracle Wallet Manager は SSL を使用して Wallet を LDAP ディレクトリにアップロードします。証明書がない場合、ユーザーはディレクトリ・パスワードの入力を求められます。

誤って Wallet が破壊されるのを防ぐために、対象となる Wallet が現在開かれていて、少なくとも 1 つのユーザー証明書が含まれていないかぎり、Oracle Wallet Manager はアップロード・オプションの実行をユーザーに許可しません。

Wallet をアップロードするには :

1. 「**ウォレット**」→「**ディレクトリ・サービス内へのアップロード**」を選択します。現在開かれている Wallet が保存されていない場合は、次のメッセージを示すダイアログ・ボックスが表示されます。

   アップロードを実行する前に Wallet を保存してください。

   「**はい**」を選択して、操作を続行します。
2. Wallet 証明書で SSL 鍵使用の有無が確認されます。Wallet で SSL 鍵を使用した証明書が見つかったかどうかによって、次のような結果になります。
   - **少なくとも 1 つの証明書で SSL 鍵が使用されている場合 :** LDAP ディレクトリ・サーバーのホスト名とポート情報の入力を求められたら、これらの情報を入力して「**OK**」をクリックします。Oracle Wallet Manager によって、SSL を使用する LDAP ディレクトリ・サーバーへの接続が試行されます。Wallet が正常にアップロードされたかどうかを通知するメッセージが表示されます。
   - **SSL 鍵を使用した証明書が存在しない場合 :** ユーザーの識別名（DN）、LDAP サーバーのホスト名とポート情報の入力を求められたら、これらの情報を入力して「**OK**」をクリックします。Oracle Wallet Manager によって、単純なパスワード認証モードを使用して LDAP ディレクトリ・サーバーへの接続が試行されます。この際、Wallet のパスワードはディレクトリ・パスワードと同じであるものと想定されます。

接続に失敗した場合は、指定されている DN のディレクトリ・パスワードの入力を求めるダイアログ・ボックスが表示されます。Oracle Wallet Manager によって、このパスワードを使用した LDAP ディレクトリ・サーバーへの接続が試行され、接続に失敗した場合には警告メッセージが表示されます。接続に成功すると、正常にアップロードされたことを示すメッセージがウィンドウの一番下に表示されます。

11.1.4.8 LDAP ディレクトリからの Wallet のダウンロード

Wallet は LDAP ディレクトリからダウンロードされると、作業メモリーに入れられます。次の項で説明するいずれかの保存オプションを使用して明示的に保存しないと、ファイル・システムには保存されません。

関連項目 :

- 第 11.1.4.9 項「変更の保存」
- 第 11.1.4.10 項「開いている Wallet の新しい場所への保存」
- 第 11.1.4.11 項「システムのデフォルトへの保存」

LDAP ディレクトリから Wallet をダウンロードするには :

1. 「**ウォレット**」→「**ディレクトリ・サービスからのダウンロード**」を選択します。
2. ユーザーの識別名（DN）、LDAP ディレクトリ・パスワード、ホスト名およびポート情報の入力を求めるダイアログ・ボックスが表示されます。Oracle Wallet Manager では、単純なパスワード認証を使用した LDAP ディレクトリへの接続が試行されます。

   ダウンロード操作が成功したかどうかによって、次のような結果になります。

   - **ダウンロード操作に失敗した場合 :** ユーザーの DN と LDAP サーバーのホスト名およびポート情報を正しく入力したことを確認してください。
   - **ダウンロードに成功した場合 :** 「**OK**」を選択し、ダウンロードした Wallet を開きます。Oracle Wallet Manager により、ディレクトリ・パスワードを使用して Wallet を開く試みがなされます。ディレクトリ・パスワードを使用して開くことができない場合は、Wallet のパスワードを入力するためのダイアログ・ボックスが表示されます。

     Wallet のパスワードを使用しても Wallet を開くことができない場合は、入力したパスワードが正しいことを確認してください。成功すると、Wallet が正常にダウンロードされたことを示すメッセージがウィンドウの一番下に表示されます。

11.1.4.9 変更の保存

現在開いている Wallet に変更を保存するには、次の手順に従います。

「**ウォレット**」→「**保存**」を選択します。

選択されたディレクトリ内の Wallet に変更が正常に保存されたことを確認するメッセージが、ウィンドウの一番下に表示されます。

11.1.4.10 開いている Wallet の新しい場所への保存

Wallet を新しい場所に保存するには、「**別名保存**」メニュー・オプションを使用します。

1. 「**ウォレット**」→「**別名保存**」を選択します。「ディレクトリの選択」ダイアログ・ボックスが表示されます。
2. Wallet を保存するディレクトリの場所を選択します。
3. 「**OK**」を選択します。

   選択した場所に Wallet がすでに存在する場合は、次のメッセージが表示されます。

   ```
   A wallet already exists in the selected path. Do you want to overwrite it?
   ```

   既存の Wallet を上書きするには、「**はい**」を選択します。Wallet を別の場所に保存するには、「**いいえ**」を選択します。

選択されたディレクトリの場所に Wallet が正常に保存されたことを確認するメッセージが、ウィンドウの一番下に表示されます。

11.1.4.11 システムのデフォルトへの保存

Wallet をデフォルトのディレクトリの場所に保存するには、「**システム・デフォルトに保存**」メニュー・オプションを使用します。

「**ウォレット**」→「**システム・デフォルトに保存**」を選択します。

システムのデフォルトの Wallet の場所に正常に保存されたことを確認するメッセージが、ウィンドウの一番下に表示されます。UNIX および Windows プラットフォームのデフォルト・ディレクトリは次のとおりです。

- （UNIX）`/etc/ORACLE/WALLETS/$USER/`
- （Windows）`%USERPROFILE%¥ORACLE¥WALLETS¥`

注意：

- SSL は、システムのデフォルト・ディレクトリの場所に保存された Wallet を使用します。
- Oracle アプリケーションによっては、Wallet がシステムのデフォルトの場所にないと使用できないものもあります。個々のアプリケーションに対する Oracle マニュアルで、デフォルトの Wallet ディレクトリの場所に Wallet を置く必要があるかどうかを確認してください。

11.1.4.12 Wallet の削除

現在開いている Wallet を削除するには、次の手順に従います。

1. 「**ウォレット**」→「**削除**」を選択します。「ウォレットの削除」ダイアログ・ボックスが表示されます。
2. 表示された Wallet の場所を確かめて、削除する Wallet であることを確認します。
3. Wallet パスワードを入力します。
4. 「**OK**」を選択します。Wallet が正常に削除されたことを示すダイアログ・パネルが表示されます。

注意： アプリケーション・メモリー内の開いている Wallet は、アプリケーションが終了するまでメモリー内に残ります。このため、現在使用中の Wallet を削除しても、システム・オペレーションにただちに影響するわけではありません。

11.1.4.13 パスワードの変更

パスワードの変更はただちに有効になります。Wallet は、新たに暗号化されたパスワードとともに、現在選択されているディレクトリに保存されます。

注意： 自動ログインを有効にして Wallet を使用している場合は、パスワードの変更後、自動ログインを再生成する必要があります。詳細は、第 11.1.4.14 項「自動ログインの使用」を参照してください。

現在開いている Wallet のパスワードを変更するには、次の手順に従います。

1. 「**ウォレット**」→「**パスワードの変更**」を選択します。「ウォレット・パスワードの変更」ダイアログ・ボックスが表示されます。
2. 既存の Wallet パスワードを入力します。
3. 新しいパスワードを入力します。
4. 新しいパスワードを再入力します。
5. 「**OK**」を選択します。

パスワードが正常に変更されたことを確認するメッセージが、ウィンドウの一番下に表示されます。

関連項目：

- 第 11.1.4.1 項「Wallet のパスワード作成に必要なガイドライン」
- 第 11.1.1.1 項「Wallet のパスワードの管理」

11.1.4.14 自動ログインの使用

Oracle Wallet Manager の自動ログイン機能では、Wallet のあいまいなコピーを作成し、その Wallet の自動ログイン機能が無効になるまで、パスワードなしでサービスに PKI アクセスすることを可能にします。ファイル・システム権限では、Wallet の自動ログインに必要なセキュリティが提供されます。

複数の Oracle データベースにシングル・サインオンでアクセスする場合は、自動ログインを有効化する必要があります（デフォルトでは無効）。これらはシングル・サインオン機能を持つため、SSO Wallet と呼ばれることもあります。

11.1.4.14.1 自動ログインの有効化

自動ログインを有効化するには、次の手順に従います。

1. メニュー・バーから「**ウォレット**」を選択します。
2. 「**自動ログイン**」を選択します。自動ログインが有効化されたことを示すメッセージが、ウィンドウの一番下に表示されます。

11.1.4.14.2 自動ログインの無効化

自動ログインを無効化するには、次の手順に従います。

1. メニュー・バーから「**ウォレット**」を選択します。
2. 「**自動ログイン**」の選択を解除します。自動ログインが無効化されたことを示すメッセージが、ウィンドウの一番下に表示されます。

11.1.5 証明書の管理

Oracle Wallet Manager では、ユーザー証明書と信頼できる証明書の 2 種類の証明書が使用されます。証明書はすべて、ネットワーク ID を対応する公開鍵にバインドする署名付きデータ構造です。ユーザー証明書は、公開鍵 / 秘密鍵の交換において、サーバー・アプリケーションなどエンド・エンティティの ID を確認するために使用されます。これに対し、信頼できる証明書は、ユーザーが信頼する任意の証明書です。たとえば、CA が発行するユーザー証明書を確認するために CA で提供される証明書などです。

この項では、この 2 種類の証明書の管理方法を説明します。この項の項目は次のとおりです。

- ユーザー証明書の管理
- 信頼できる証明書の管理

注意： ユーザー証明書をインストールするには、そのユーザー証明書を発行した認証局を表す信頼できる証明書が Wallet に含まれている必要があります。ただし、新しい Wallet を作成するときは必ず、いくつかの、一般的な信頼できる証明書が自動的にインストールされます。これは、これらの証明書が幅広く使用されているためです。必要な認証局を表す証明書が存在しない場合は、認証局の証明書を最初にインストールする必要があります。

また、PKCS #7 証明連鎖形式を使用してインポートすることもできます。この方法では、ユーザー証明書と CA 証明書を同時に入手できます。

11.1.5.1 ユーザー証明書の管理

ユーザー証明書はエンド・ユーザー、スマート・カード、または Web サーバーなどのアプリケーションで使用されます。サーバー証明書は、ユーザー証明書の 1 つです。たとえば、CA で Web サーバーの証明書が発行され、件名のフィールドにその識別名（DN）が記述されていれば、その Web サーバーが証明書の所有者、つまりこのユーザー証明書のユーザーとなります。

ユーザー証明書の管理には、次のようなタスクが含まれます。

- 証明書リクエストの追加
- Wallet へのユーザー証明書のインポート
- サード・パーティ製ツールによって作成された証明書のインポート
- Wallet からのユーザー証明書の削除
- 証明書リクエストの削除
- ユーザー証明書のエクスポート
- ユーザー証明書リクエストのエクスポート

11.1.5.1.1 証明書リクエストの追加 Oracle Wallet Manager を使用して、複数の証明書リクエストを追加できます。複数のリクエストを追加すると、Oracle Wallet Manager では自動的に最初のリクエストの内容を、それに続く各リクエストのダイアログ・ボックスに取り込みます。これは後から編集できます。

実際の証明書リクエストは、Wallet の一部になります。証明書リクエストを再利用して、新しい証明書を取得できます。ただし、既存の証明書リクエストの編集はできません。Wallet には、正しく入力された証明書だけを格納してください。

PKCS #10 証明書リクエストを作成するには、次の手順に従います。

1. 「**操作**」→「**証明書リクエストの追加**」を選択します。「証明書リクエストの追加」ダイアログ・ボックスが表示されます。
2. 表 11-2 に示す情報を入力します。
3. 「**OK**」を選択します。証明書リクエストが正常に作成されたことを示すメッセージが表示されます。このダイアログ・パネルの本文から証明書リクエストのテキストをコピーし、電子メール・メッセージに貼り付けて認証局に送信するか、または証明書リクエストをファイルにエクスポートします。
4. 「**OK**」を選択して、Oracle Wallet Manager のメイン・ウィンドウに戻ります。証明書のステータスが「**リクエスト済**」に変化します。

関連項目： 第 11.1.5.1.7 項「ユーザー証明書リクエストのエクスポート」

表 11-2 証明書リクエストのフィールドと説明

フィールド名	説明
共通名	必須。ユーザーまたはサービスの識別名を入力します。ユーザー名は、名前 / 姓の書式で入力します。 例 : Eileen.Sanger
組織単位	オプション。識別対象の組織単位名を入力します。例 : Finance
組織	オプション。識別対象の組織名を入力します。例 : XYZ Corp.
市町村	オプション。識別対象が所在する市区町村名を入力します。
都道府県	オプション。識別対象が所在する都道府県名を略さずに入力します。 認証局によっては略称を受け入れない場合があるので、フル・ネームで入力してください。
国	必須。国の略称リストを表示します。組織の所在地の国を選択します。
鍵のサイズ	必須。公開鍵と秘密鍵のペアを作成する際に使用する鍵サイズのリストを表示します。鍵サイズの評価には、表 11-3 を参照してください。
詳細	オプション。「**詳細**」を選択すると、「証明書リクエストの詳細」ダイアログ・パネルが表示されます。このフィールドを使用して、識別名（DN）を編集またはカスタマイズします。たとえば、都道府県や市区町村のフル・ネームを編集できます。

表 11-3 に、使用できる鍵サイズおよび各サイズで提供されるセキュリティを示します。通常、CA では 1024 または 2048 の鍵サイズが使用されます。証明書の所有者が長期間鍵を保存する場合は、3072 または 4096 ビット鍵を選択します。

表 11-3 使用できる鍵サイズ

鍵のサイズ	関連するセキュリティ・レベル
512 または 768	セキュアであるとは見なされません。
1024 または 2048	セキュアです。
3072 または 4096	非常にセキュアです。

11.1.5.1.2 Wallet へのユーザー証明書のインポート 認証局から証明書が交付されるとき、証明書は電子メールでテキスト（BASE64）形式で送信されることも、バイナリ・ファイルとして添付されることもあります。

注意 : 認証局から送信される証明書には、PKCS #7 証明連鎖の場合と独立した X.509 証明書の場合があります。Oracle Wallet Manager ではこれら両方のタイプをインポートできます。

PKCS #7 証明連鎖とは、複数の証明書の集合です。これには、ユーザー証明書と、サポートする信頼できる CA および下位 CA の証明書がすべて含まれます。

対照的に、X.509 証明書ファイルに格納されるのは 1 つの独立した証明書だけで、サポートする証明連鎖は含まれません。

ただし、このような独立した証明書をインポートするには、署名者の証明書が、信頼できる証明書として Wallet 内に存在している必要があります。

認証局の電子メールのテキストからユーザー証明書をインポートするには、テキスト（BASE64）で記述されている証明書を電子メール・メッセージからコピーします。Begin Certificate および End Certificate の行を取り込みます。

1. 「**操作**」→「**ユーザー証明書のインポート**」を選択します。「証明書のインポート」ダイアログ・ボックスが表示されます。
2. 「**証明書の貼付け**」を選択して「**OK**」をクリックします。次のメッセージを示すもう 1 つの「証明書のインポート」ダイアログ・ボックスが表示されます。

   ```
   Please provide a base64 format certificate and paste it below.
   ```

3. 証明書をダイアログ・ボックスにコピーして、「**OK**」を選択します。
   a. 受け取った証明書が PKCS #7 形式の場合、これがインストールされ、PKCS #7 データに関連する他のすべての証明書が「信頼できる証明書」リストに表示されます。
   b. 受け取った証明書が PKCS #7 形式でなく、その CA の証明書が「信頼できる証明書」リストに含まれていない場合は、さらに操作が必要です。Oracle Wallet Manager によって、証明書を発行した CA の証明書をインポートするように求められます。この CA 証明書は「信頼できる証明書」リストに表示されます（CA 証明書がすでに「信頼できる証明書」リストにある場合は、ユーザーの証明書は追加手順なしでインポートされます）。

   a または b のいずれかを完了すると、証明書が正常にインストールされたことを確認するメッセージが、ウィンドウの一番下に表示されます。Oracle Wallet Manager のメイン・ウィンドウが再び表示され、左のパネルのサブツリーにある対応するエントリのステータスが「**待機中**」に変化します。

注意：

標準の X.509 証明書には、次に示す開始と終了のテキストが含まれています。

```
-----BEGIN CERTIFICATE-----
-----END CERTIFICATE-----
```

前述したように、一般的な PKCS #7 の証明書には、この他に次に示す開始と終了のテキストがあります。

```
-----BEGIN PKCS7-----
-----END PKCS7-----
```

標準の [Ctrl] + [C] を使用してコピーし（すべてのダッシュを含む）、[Ctrl] + [V] を使用して貼り付けます。

証明書をファイルからインポートするには：

ファイル内のユーザー証明書は、テキスト（BASE64）とバイナリ（der）のどちらの形式でもかまいません。

1. 「**操作**」→「**ユーザー証明書のインポート**」を選択します。「証明書のインポート」ダイアログ・ボックスが表示されます。
2. 「**証明書を含むファイルを選択**」を選択して、「**OK**」をクリックします。もう 1 つの「証明書のインポート」ダイアログ・ボックスが表示されます。
3. 証明書を含むファイルの場所のパスまたはフォルダ名を入力します。
4. 証明書ファイルの名前（たとえば cert.txt や cert.der）を選択します。
5. 「**OK**」を選択します。
   a. 受け取った証明書が PKCS #7 形式の場合、これがインストールされ、PKCS #7 データに関連する他のすべての証明書が「信頼できる証明書」リストに表示されます。

b. 受け取った証明書が PKCS #7 形式でなく、その CA の証明書が「信頼できる証明書」リストに含まれていない場合は、さらに操作が必要です。Oracle Wallet Manager によって、証明書を発行した CA の証明書をインポートするように求められます。この CA 証明書は「信頼できる証明書」リストに表示されます（CA 証明書がすでに「信頼できる証明書」リストにある場合は、ユーザーの証明書は追加手順なしでインポートされます）。

a または b のいずれかを完了すると、証明書が正常にインストールされたことを確認するメッセージが、ウィンドウの一番下に表示されます。Oracle Wallet Manager のメイン・パネルに戻ります。左のパネルのサブツリーにある対応するエントリが「**待機中**」に変化します。

11.1.5.1.3 サード・パーティ製ツールによって作成された証明書のインポート 第三者の証明書とは、Oracle Wallet Manager を使用して生成されたのではない証明書リクエストから作成された証明書を指します。このような第三者の証明書には、ユーザー証明書以外も含まれるので、Oracle の観点では実際には Wallet です。第三者の証明書には、その証明書の秘密鍵も含まれています。さらに、信頼できる証明書の連鎖が含まれており、証明書を作成したのが信頼できる機関であることが立証されます。

PKCS #12 形式でインポートすると、Oracle Wallet Manager でこのような Wallet を利用できるようになります。この形式には、前述の 3 つの要素（ユーザー証明書、秘密鍵、信頼できる証明書）がすべて含まれます。サポートされる PKCS #12 形式の証明書は次のとおりです。

- Netscape Communicator 4.x
- Microsoft Internet Explorer 5.x 以降

Oracle Wallet Manager は PKCS #12 規格に準拠しています。したがって、PKCS #12 準拠のツールでエクスポートされた証明書は、すべて Oracle Wallet Manager で使用できます。

このような第三者の証明書を既存の Oracle Wallet に格納することはできません。秘密鍵や信頼できる機関の連鎖が欠如している可能性があるためです。したがって、このような証明書に対しては、それぞれ独立した PKCS #12 ファイルとして、つまり個別の Wallet としてエクスポートおよび取出しを実行します。

サード・パーティ製ツールで作成された証明書をインポートするには、初めに、使用しているアプリケーションから証明書をエクスポートします。次に、Oracle Wallet Manager で読取り可能な Wallet ファイルとして証明書を保存します。

サード・パーティ製ツールで作成された証明書をインポートするには :

1. 使用する製品に応じた手順に従って、証明書をエクスポートします。秘密鍵もエクスポートする場合は、エクスポートする製品で示されているアクションを実行してください。また、エクスポートされた証明書を保護するための新しいパスワードも指定してください。関係するトラスト・ポイントもすべてエクスポートされるようにしてください（PKCS #12 規格では、ブラウザが署名者自身の証明書以外の**信頼できる証明書**を必ずしもエクスポートするわけではありません。ピアを認証するには、その他にも証明書が必要な場合があります。Oracle Wallet Manager を使用して、信頼できる証明書をインポートできます）。

   作成されたファイル（証明書、秘密鍵およびトラスト・ポイントを含む）は新しい Wallet であり、これによって第三者の証明書を利用できるようになります。

2. エクスポートされた証明書に、オペレーティング・システムに応じた適切なファイル名を付けて、Oracle Wallet Manager が使用するディレクトリに保存します。

   UNIX および Windows の場合、ファイル名は `ewallet.p12` です。

   その他のオペレーティング・システムについては、該当するオペレーティング・システムの Oracle ドキュメントを参照してください。

3. Oracle Wallet Manager を使用して、`ewallet.p12` ファイルが保存されているディレクトリにナビゲートしてファイルを開き、格納されている PKI 資格証明を使用します。

注意： 関係するアプリケーションの起動時など、証明書が必要とされるときに、パスワードが要求されます。このようなアクセスを自動化するには、第 11.1.4.14 項「自動ログインの使用」を参照してください。

ただし、証明書の秘密鍵が、独立したハードウェア・セキュリティ・モジュール内に保持されている場合は、その証明書をインポートすることはできません。

信頼できる証明書を別にエクスポートした場合は、インポートした第三者のユーザー証明書が含まれている `ewallet.p12` ファイルを開く前に、信頼できる証明書をインポートしてください。

関連項目： 第 11.1.5.2.1 項「信頼できる証明書のインポート」

11.1.5.1.4 Wallet からのユーザー証明書の削除 Wallet からユーザー証明書を削除するには、次の手順に従います。

1. 左のパネルのサブツリーで、削除する証明書を選択します。
2. 「**操作**」→「**ユーザー証明書の削除**」を選択します。Wallet からユーザー証明書を削除してよいかどうかを確認するダイアログ・パネルが表示されます。
3. 「**はい**」を選択して、Oracle Wallet Manager のメイン・パネルに戻ります。証明書のステータスが、「**リクエスト済**」になります。

11.1.5.1.5 証明書リクエストの削除 関連するリクエストを削除する前に、証明書を削除する必要があります。

証明書リクエストを削除するには、次の手順に従います。

1. 左のパネルのサブツリーで、削除する証明書リクエストを選択します。
2. 「**操作**」→「**証明書リクエストの削除**」を選択します。
3. 「**はい**」をクリックします。証明書のステータスが、「**空**」になります。

11.1.5.1.6 ユーザー証明書のエクスポート 証明書をファイル・システム・ディレクトリに保存するには、次の手順に従って証明書をエクスポートします。

1. 左のパネルのサブツリーで、エクスポートする証明書を選択します。
2. メニュー・バーから「**操作**」→「**ユーザー証明書のエクスポート**」を選択します。「証明書のエクスポート」ダイアログ・ボックスが表示されます。
3. 証明書を保存するファイル・システム・ディレクトリの場所を入力するか、「**フォルダ**」の下のディレクトリ構造にナビゲートします。
4. 証明書のファイル名を「**ファイル名**」フィールドに入力します。
5. 「**OK**」を選択します。証明書がファイルへ正常にエクスポートされたことを確認するメッセージが、ウィンドウの一番下に表示されます。Oracle Wallet Manager のメイン・ウィンドウに戻ります。

関連項目： Wallet のエクスポートの詳細は、第 11.1.4.5 項「サード・パーティ環境への Oracle Wallet のエクスポート」を参照してください。Oracle Wallet Manager では複数の証明書を 1 つの Wallet に格納する機能をサポートしていますが、現時点では、ブラウザでサポートされているのは一般に証明書が 1 つだけの Wallet のみです。これらのブラウザでは、単一鍵ペアを持つ Oracle Wallet をエクスポートする必要があります。

11.1.5.1.7 ユーザー証明書リクエストのエクスポート 証明書リクエストをファイル・システム・ディレクトリに保存するには、次の手順に従って証明書リクエストをエクスポートします。

1. 左のパネルのサブツリーで、エクスポートする証明書リクエストを選択します。
2. 「**操作**」→「**証明書リクエストのエクスポート**」を選択します。「証明書リクエストのエクスポート」ダイアログ・ボックスが表示されます。
3. 証明書リクエストを保存するファイル・システム・ディレクトリの場所を入力するか、「**フォルダ**」の下のディレクトリ構造にナビゲートします。
4. 証明書リクエストのファイル名を「**ファイル名**」フィールドに入力します。
5. 「**OK**」を選択します。証明書リクエストがファイルへ正常にエクスポートされたことを確認するメッセージが、ウィンドウの一番下に表示されます。Oracle Wallet Manager のメイン・ウィンドウに戻ります。

11.1.5.2 信頼できる証明書の管理

信頼できる証明書の管理には、次のようなタスクが含まれます。

- 信頼できる証明書のインポート
- 信頼できる証明書の削除
- 信頼できる証明書のエクスポート
- すべての信頼できる証明書のエクスポート

11.1.5.2.1 信頼できる証明書のインポート 信頼できる証明書を Wallet にインポートするには、認証局から受信した電子メールから信頼できる証明書を貼り付ける方法と、信頼できる証明書をファイルからインポートする方法の 2 つがあります。

Oracle Wallet Manager では、Wallet を新規作成すると、信頼できる証明書が VeriSign、RSA、Entrust および GTE CyberTrust から自動的にインストールされます。

テキストのみ（BASE64）の信頼できる証明書をコピーおよび貼り付けるには：

ユーザー証明書が含まれている、受信した電子メール・メッセージの本文から、信頼できる証明書をコピーします。`Begin Certificate` および `End Certificate` の行を取り込みます。

1. メニュー・バーから「**操作**」→「**信頼できる証明書のインポート**」を選択します。「信頼できる証明書のインポート」ダイアログ・パネルが表示されます。
2. 「**証明書の貼付け**」を選択して「**OK**」をクリックします。次のメッセージを示すもう 1 つの「信頼できる証明書のインポート」ダイアログ・パネルが表示されます。

   ```
   Please provide a base64 format certificate and paste it below.
   ```

3. 証明書をウィンドウに貼り付け、「**OK**」をクリックします。信頼できる証明書が正常にインストールされたことを示すメッセージが、ウィンドウの一番下に表示されます。
4. 「**OK**」を選択します。Oracle Wallet Manager のメイン・パネルに戻り、「信頼できる証明書」ツリーの一番下に、インストールした信頼できる証明書が表示されます。

証明書をコピーおよび貼り付けるためのキーボード・ショートカット：

コピーするには [Ctrl] + [C]、貼り付けるには [Ctrl] + [V] を使用します。

信頼できる証明書を含むファイルをインポートするには :
信頼できる証明書が含まれているファイルは、テキスト（BASE64）とバイナリ（der）のいずれかの形式で保存されています。

1. 「**操作**」→「**信頼できる証明書のインポート**」を選択します。「信頼できる証明書のインポート」ダイアログ・パネルが表示されます。
2. 信頼できる証明書がある場所のパスまたはフォルダ名を入力します。
3. 信頼できる証明書のファイル名（たとえば cert.txt）を選択します。
4. 「**OK**」を選択します。信頼できる証明書が Wallet へ正常にインポートされたことを示すメッセージが、ウィンドウの一番下に表示されます。
5. 「**OK**」を選択してダイアログ・パネルを終了します。Oracle Wallet Manager のメイン・パネルに戻り、「信頼できる証明書」ツリーの一番下に、インストールした信頼できる証明書が表示されます。

11.1.5.2.2 信頼できる証明書の削除 ユーザー証明書の署名に使用した信頼できる証明書は、そのユーザー証明書がまだ Wallet 内に残っている間は削除できません。このような信頼できる証明書を削除するには、まず署名した証明書を削除する必要があります。また、Wallet から信頼できる証明書が削除された後は、証明書を確認できません。

信頼できる証明書を Wallet から削除するには、次の手順に従います。

1. 「信頼できる証明書」ツリーに表示されている信頼できる証明書を選択します。
2. メニュー・バーから「**操作**」→「**信頼できる証明書の削除**」を選択します。

   署名に使用された信頼できる証明書を削除すると、使用しているユーザー証明書を受信者が検査できなくなることを警告するダイアログ・パネルが表示されます。
3. 「**はい**」を選択します。「信頼できる証明書」ツリーから、選択された信頼できる証明書が削除されます。

11.1.5.2.3 信頼できる証明書のエクスポート 信頼できる証明書をファイル・システムの別の場所にエクスポートするには、次の手順に従います。

1. 左のパネルのサブツリーで、エクスポートする信頼できる証明書を選択します。
2. 「**操作**」→「**信頼できる証明書のエクスポート**」を選択します。「信頼できる証明書のエクスポート」ダイアログ・ボックスが表示されます。
3. 信頼できる証明書の保存先になるファイル・システム・ディレクトリを入力するか、「**フォルダ**」の下のディレクトリ構造にナビゲートします。
4. 信頼できる証明書を保存するファイル名を入力します。
5. 「**OK**」を選択します。Oracle Wallet Manager のメイン・ウィンドウに戻ります。

11.1.5.2.4 すべての信頼できる証明書のエクスポート 信頼できる証明書すべてをファイル・システムの別の場所にエクスポートするには、次の手順に従います。

1. 「**操作**」→「**すべての信頼できる証明書のエクスポート**」を選択します。「信頼できる証明書のエクスポート」ダイアログ・ボックスが表示されます。
2. 信頼できる証明書の保存先になるファイル・システム・ディレクトリを入力するか、「**フォルダ**」の下のディレクトリ構造にナビゲートします。
3. 信頼できる証明書を保存するファイル名を入力します。
4. 「**OK**」を選択します。Oracle Wallet Manager のメイン・ウィンドウに戻ります。

11.2 orapki ユーティリティによる証明書検証と CRL 管理の実行

orapki ユーティリティは、証明書失効リスト（CRL）の管理、Oracle Wallet の作成と管理、およびテスト用の署名付き証明書の作成を行うためのコマンドライン・ツールです。

次の各項目で、このツールの概要と使用方法を説明します。

11.2.1 orapki の概要

orapki は、公開鍵インフラストラクチャ（PKI）の要素（Wallet や証明書失効リストなど）の管理をコマンドラインで行うためのユーティリティです。このユーティリティで実行するタスクを、スクリプトの中に組み込むことができます。このユーティリティを利用すると、PKI の保守に関する定常的なタスクの多くを自動化できます。

このコマンドライン・ユーティリティを使用して実行できるタスクは次のとおりです。

- テスト用の署名付き証明書の作成
- Oracle Wallet の管理
  - Oracle Wallet の作成と表示
  - 証明書リクエストの追加と削除
  - 証明書の追加と削除
  - 信頼できる証明書の追加と削除
- 証明書失効リスト（CRL）の管理
  - 証明書検証のためのハッシュ値による CRL 名の変更
  - Oracle Internet Directory 内の CRL のアップロード、一覧表示、個別表示および削除

11.2.1.1 orapki ユーティリティの構文

orapki コマンドライン・ユーティリティの基本的な構文は次のとおりです。

```
orapki module command -parameter value
```

このコマンドの *module* は、wallet（Oracle Wallet）、crl（証明書失効リスト）、cert（PKI デジタル証明書）のいずれかです。使用できるコマンドは、使用する module によって異なります。たとえば、wallet の操作を行う場合は、add コマンドを使用して証明書や鍵を Wallet に追加できます。次の例では、/private/lhale/cert.txt にあるユーザー証明書を、ORACLE_HOME/wallet/ewallet.p12 にある Wallet に追加します。

```
orapki wallet add -wallet ORACLE_HOME/wallet/ewallet.p12
-user_cert -cert /private/lhale/cert.txt
```

11.2.2 orapki のヘルプの表示

特定のモードで使用可能な orapki のコマンドをすべて表示するには、コマンドラインで次のように入力します。

```
orapki mode help
```

たとえば、証明書失効リスト（CRL）の管理に使用できるコマンドをすべて表示するには、コマンドラインで次のように入力します。

```
orapki CRL help
```

> **注意：** -summary、-complete、-wallet の各コマンド・オプションは省略可能です。これらのコマンド・オプションを指定しなくても、コマンドは実行されます。

11.2.3 テスト用の署名付き証明書の作成

このコマンドライン・ユーティリティを利用すると、テスト用の署名付き証明書を簡単に作成できます。署名付き証明書の作成および表示を行うための構文を次に示します。

テスト用の署名付き証明書を作成するには：

```
orapki cert create [-wallet wallet_location] -request
 certificate_request_location
-cert certificate_location -validity number_of_days [-summary]
```

このコマンドを実行すると、証明書リクエストから署名付き証明書が作成されます。-wallet パラメータでは、証明書リクエストに署名するために使用されるユーザー証明書および秘密鍵を含む Wallet を指定します。-validity パラメータでは、この証明書の有効期間を、実行日からの日数として指定します。このコマンドでは、証明書と証明書リクエストの指定は必須です。

証明書を表示するには：

```
orapki cert display -cert certificate_location [-summary | -complete]
```

このコマンドでは、orapki で作成したテスト証明書を表示できます。-summary と -complete のいずれかを選択して、表示する詳細のレベルを指定できます。-summary を選択すると、証明書とその有効期限が表示されます。-complete を選択すると、シリアル番号や公開鍵などの、その他の証明書情報も表示されます。

11.2.4 orapki ユーティリティによる Oracle Wallet の管理

次の各項では、orapki コマンドライン・ユーティリティを使用して Oracle Wallet を作成および管理するための構文を説明します。orapki ユーティリティの wallet モジュールのコマンドをスクリプトで使用すると、Wallet 作成プロセスを自動化できます。

- orapki による Oracle Wallet の作成と表示
- orapki による Oracle Wallet への証明書および証明書リクエストの追加
- orapki による Oracle Wallet からの証明書および証明書リクエストのエクスポート

> **注意：** wallet モジュールのコマンドでは、-wallet パラメータは常に必須です。

11.2.4.1 orapki による Oracle Wallet の作成と表示

Oracle Wallet を作成するには :

```
orapki wallet create -wallet wallet_location
```

このコマンドを実行すると、Wallet のパスワードの入力と再入力を要求されます。Wallet は、`-wallet` で指定した場所に作成されます。

Oracle Wallet を作成して自動ログインを有効化するには :

```
orapki wallet create -wallet wallet_location -auto_login
```

このコマンドを実行すると、自動ログイン可能な Wallet が作成されます。既存の Wallet に対して自動ログインを有効化することもできます。`wallet_location` にすでに Wallet が含まれる場合は、その Wallet の自動ログインが有効化されます。自動ログイン機能を無効にするには、Oracle Wallet Manager を使用します。詳細は、第 11.1.4.14 項「自動ログインの使用」を参照してください。

注意 : 自動ログインが有効な Wallet に対しては、`add` などの、Wallet を変更する操作を行う場合にのみパスワードの入力が要求されます。

Oracle Wallet を表示するには :

```
orapki wallet display -wallet wallet_location
```

このコマンドを実行すると、Wallet に含まれている証明書リクエスト、ユーザー証明書および信頼できる証明書が表示されます。

11.2.4.2 orapki による Oracle Wallet への証明書および証明書リクエストの追加

証明書リクエストを Oracle Wallet に追加するには :

```
orapki wallet add -wallet wallet_location -dn user_dn -keySize 512|1024|2048
```

このコマンドを実行すると、指定した識別名（`user_dn`）のユーザーの Wallet に証明書リクエストが追加されます。リクエストする証明書の鍵サイズ（512 ビット、1024 ビットまたは 2048 ビット）も指定します。リクエストの署名およびエクスポート・オプションによるリクエストのエクスポートについては、第 11.2.4.3 項「orapki による Oracle Wallet からの証明書および証明書リクエストのエクスポート」を参照してください。

信頼できる証明書を Oracle Wallet に追加するには :

```
orapki wallet add -wallet wallet_location -trusted_cert -cert
certificate_location
```

このコマンドを実行すると、指定した場所（`-cert certificate_location`）にある信頼できる証明書が Wallet に追加されます。ユーザー証明書を追加する前に、ユーザー証明書の証明連鎖内の信頼できる証明書をすべて追加する必要があります。そうしなければ、ユーザー証明書を追加するコマンドは失敗します。

ルート証明書を Oracle Wallet に追加するには :

```
orapki wallet add -wallet wallet_location -dn
certificate_dn -keySize 512|1024|2048 -self_signed -validity number_of_days
```

このコマンドを実行すると、新しい自己署名（ルート）証明書が作成されて Wallet に追加されます。`-validity` パラメータ（必須）では、この証明書の有効期間を、実行日からの日数として指定します。このルート証明書の鍵サイズ（`-keySize`）には、512 ビット、1024 ビット、2048 ビットのいずれかを指定できます。

ユーザー証明書を Oracle Wallet に追加するには :

```
orapki wallet add -wallet wallet_location -user_cert -cert certificate_location
```

このコマンドを実行すると、-cert パラメータで指定した場所にあるユーザー証明書が、*wallet_location* にある Wallet に追加されます。Wallet にユーザー証明書を追加する前に、証明連鎖を構成するすべての信頼できる証明書を追加する必要があります。ユーザー証明書を追加する前に、Wallet にすべての信頼できる証明書がインストールされていない場合、ユーザー証明書の追加に失敗します。

11.2.4.3 orapki による Oracle Wallet からの証明書および証明書リクエストのエクスポート

Oracle Wallet から証明書をエクスポートするには :

```
orapki wallet export -wallet wallet_location -dn
certificate_dn -cert certificate_filename
```

このコマンドを実行すると、サブジェクトの識別名（-dn）を持つ証明書が、Wallet から、-cert で指定されたファイルにエクスポートされます。

Oracle Wallet から証明書リクエストをエクスポートするには :

```
orapki wallet export -wallet wallet_location -dn
certificate_request_dn -request certificate_request_filename
```

このコマンドを実行すると、サブジェクトの識別名（-dn）を持つ証明書リクエストが、Wallet から、-request で指定したファイルにエクスポートされます。

11.2.5 orapki ユーティリティによる証明書失効リスト（CRL）の管理

CRL の管理には orapki を使用する必要があります。このユーティリティによって、CRL 発行者名のハッシュ値が作成されます。このハッシュ値は、システム内の CRL の場所を特定するためのものです。orapki を使用しないと、PKI デジタル証明書を検証するための CRL を Oracle サーバーが見つけることができなくなります。次の各項では、CRL の概要と使用方法、および orapki を使用した管理方法を説明します。

- 第 11.2.5.1 項「証明書失効リストを使用した証明書の検証について」
- 第 11.2.5.2 項「証明書失効リストの管理」

11.2.5.1 証明書失効リストを使用した証明書の検証について

特定の証明書が特定の状況において使用可能かどうかを判断するプロセスを、証明書の検証と呼びます。証明書の検証では、次のことを判断します。

- 信頼できる認証局（CA）によって証明書にデジタル署名が付加されているかどうか
- 証明書のデジタル署名が、証明書自身の別途計算されたハッシュ値および証明書署名者（CA）の公開鍵と対応しているかどうか
- 証明書の有効期限が切れていないかどうか
- 証明書が失効していないかどうか

最初の 3 つの検証は SSL ネットワーク層によって自動的に実行されますが、証明書が失効していないことを確認するために、管理者は証明書失効リスト（CRL）チェックを構成する必要があります。CRL とは署名付きデータ構造で、失効した証明書のリストが格納されています。CRL は一般に、元の証明書を発行したのと同じ機関によって発行および署名されます。

11.2.5.1.1 どの CRL を使用するか 信頼できるトラスト・ポイントすべてについて、CRL が必要です。トラスト・ポイントとは、特定の信頼レベルを満たした第三者の ID からの信頼できる証明書です。一般的に、信頼できる認証局をトラスト・ポイントと呼びます。

11.2.5.1.2 CRL チェックのしくみ 証明書の失効ステータスは、ファイル・システム・ディレクトリまたは Oracle Internet Directory 内にある CRL、または証明書の CRL 配布ポイント（CRL DP）拡張で指定されている場所からダウンロードされた CRL と比較してチェックされます。CRL をローカル・ファイル・システムまたはディレクトリに格納する場合は、管理者が定期的に更新する必要があります。CRL DP を使用する場合は、証明書の使用のたびに CRL がダウンロードされるので、CRL を定期的にリフレッシュする必要はありません。

サーバーは、次に示す順序で CRL を探します。証明書 CA の DN と一致する CRL が見つかると、検索は終了します。

1. ローカル・ファイル・システム

   `sqlnet.ora` ファイルに `SSL_CRL_FILE` パラメータがあるかどうかを調べます。次に、`SSL_CRL_PATH` パラメータを探します。この 2 つのパラメータが指定されていない場合は、Wallet の保存場所に CRL があるかどうかを調べます。

   注意 : CRL をローカル・ファイル・システムに格納する場合は、`orapki` ユーティリティを使用して CRL を定期的に更新する必要があります。第 11.2.5.2.1 項「証明書検証のためのハッシュ値による CRL 名の変更」を参照してください。

2. Oracle Internet Directory

   ローカル・ファイル・システム上で CRL が見つからないけれども、`ORACLE_HOME/ldap/admin/ldap.ora` ファイル内でディレクトリ接続情報が構成されている場合は、そのディレクトリ内を検索します。CA の識別名（DN）と CRL サブツリーの DN を使用して CRL サブツリーを検索します。

   このディレクトリ内で CRL を検索できるようにするには、サーバーの `ldap.ora` ファイルが正しく構成されている必要があります。Oracle Internet Directory のドメイン・ネーム・システム（DNS）探索機能は使用できません。また、CRL をディレクトリ内に格納する場合は、`orapki` ユーティリティを使用して CRL を定期的に更新する必要があります。第 11.2.5.2.2 項「Oracle Internet Directory への CRL のアップロード」を参照してください。

3. CRL DP

   証明書の発行時に X.509 バージョン 3 の CRL DP 証明書拡張で CA によって場所が指定されている場合は、その証明書の失効情報を含む CRL がダウンロードされます。現時点では、Oracle Advanced Security は HTTP および LDAP を介した CRL のダウンロードをサポートしています。

注意 :

- パフォーマンス上の理由から、チェックの対象はユーザー証明書のみとなっています。
- CRL は、ローカル・ファイル・システムではなく、ディレクトリ内に格納することをお薦めします。

11.2.5.2 証明書失効リストの管理

証明書失効ステータスのチェックを有効にする前に、使用している CA から受け取る CRL が、システムで使用可能な形式（ハッシュ値で名前変更済）であること、または場所（ディレクトリにアップロード済）に存在することを確認する必要があります。Oracle Advanced Security には、次の作業の実行が可能なコマンドライン・ユーティリティ `orapki` が用意されています。

- 証明書検証のためのハッシュ値による CRL 名の変更
- Oracle Internet Directory への CRL のアップロード
- Oracle Internet Directory に格納されている CRL の一覧表示
- Oracle Internet Directory 内の CRL の表示
- Oracle Internet Directory からの CRL の削除

注意： 検証が正しく行われるようにするには、CRL を定期的に（期限切れになる前に）更新する必要があります。このタスクを自動化するには、スクリプトの中で orapki のコマンドを実行します。

LDAP のコマンドライン・ツールを使用して、Oracle Internet Directory の中にある CRL を管理することもできます。

関連項目： LDAP のコマンドライン・ツールおよびその構文の説明は、『Oracle Identity Management アプリケーション開発者ガイド』のコマンドライン・ツールの構文に関する項を参照してください。

11.2.5.2.1 証明書検証のためのハッシュ値による CRL 名の変更 証明書を作成した CA によって発行された CRL を、証明書の検証時にシステムが検出できるようにしておく必要があります。システムは証明書内の発行者名と CRL 内の発行者名を対応させ、該当する CRL を検出します。

Oracle Net Manager の「**証明書失効リスト・パス**」フィールドで CRL の格納場所を指定するときに（sqlnet.ora ファイル内の SSL_CRL_PATH パラメータを設定する）、orapki ユーティリティを使用して、発行者名を表すハッシュ値によって CRL の名前を変更してください。ハッシュ値を作成すると、サーバーが CRL をロードできるようになります。

UNIX システムでは、orapki によって CRL へのシンボリック・リンクが作成されます。Windows システムでは、CRL ファイルのコピーが作成されます。どちらの場合も、orapki によって作成されたシンボリック・リンクまたはコピーの名前には、発行者名のハッシュ値が使用されます。その後で、システムが証明書を検証するときに、同じハッシュ関数を使用してリンクまたはコピーの名前が計算されるので、正しい CRL をロードできるようになります。

オペレーティング・システムに応じて次のいずれかのコマンドを使用し、ファイル・システム内に格納されている CRL の名前を変更します。

UNIX のファイル・システム内に格納されている CRL の名前を変更するには：

```
orapki crl hash -crl crl_filename [-wallet wallet_location]
-symlink crl_directory [-summary]
```

Windows のファイル・システム内に格納されている CRL の名前を変更するには：

```
orapki crl hash -crl crl_filename
[-wallet wallet_location] -copy crl_directory [-summary]
```

このコマンドの crl_filename は CRL ファイルの名前、*wallet_location* は CRL を発行した CA の証明書が含まれている Wallet の場所、*crl_directory* は CRL が存在するディレクトリです。

-wallet および -summary は、省略可能です。-wallet を指定すると、CRL 名を変更する前に、CA の証明書と比較して CRL の有効性が検証されます。-summary オプションを指定すると、CRL の発行者名が表示されます。

11.2.5.2.2 Oracle Internet Directory への CRL のアップロード CRL をディレクトリ内で公開すると、CRL 検証を企業全体で行うことができ、個々のアプリケーションで独自の CRL を構成する必要はなくなります。集中管理可能なディレクトリに格納された CRL をすべてのアプリケーションで利用できるので、CRL の管理および利用における管理オーバーヘッドが大幅に削減されます。

orapki を使用して CRL をディレクトリにアップロードするユーザーは、ディレクトリ・グループ CRLAdmins（cn=CRLAdmins,cn=groups,%s_OracleContextDN%）のメンバーである必要があります。この操作の実行に権限が必要とされるのは、CRL が企業全体からアクセス可能であるためです。実行するユーザーをこの管理用ディレクトリ・グループに追加するには、ディレクトリ管理者に連絡してください。

CRL をディレクトリにアップロードするには、コマンドラインで次のように入力します。

```
orapki crl upload -crl crl_location
-ldap hostname:ssl_port -user username [-wallet wallet_location] [-summary]
```

このコマンドで、*crl_location* は CRL が存在するファイルの名前または URL、*hostname* および *ssl_port*（認証なしの SSL ポート）はディレクトリがインストールされているシステムのホスト名とポート、*username* は CRL サブツリーに CRL を追加する権限を持つディレクトリ・ユーザー、*wallet_location* は CRL を発行した CA の証明書が含まれている Wallet の場所です。

`-wallet` および `-summary` は、省略可能です。`-wallet` を指定すると、ディレクトリへのアップロードの前に、CA の証明書と比較して CRL の有効性が検証されます。`-summary` オプションを指定すると、CRL の発行者名およびディレクトリ内に CRL が格納されている LDAP エントリが出力されます。

注意：

- `orapki` ユーティリティを使用してこの操作を実行するときに、ディレクトリのパスワードの入力が要求されます。
- Diffie-Hellman ベースの SSL サーバーが実行されているディレクトリ SSL ポートを必ず指定してください。これは、認証を実行しない SSL ポートです。サーバー認証や相互認証を行う SSL ポートは、`orapki` ユーティリティではサポートされていません。

11.2.5.2.3 Oracle Internet Directory に格納されている CRL の一覧表示 ディレクトリに格納されているすべての CRL を、`orapki` を使用して一覧表示できます。これは CRL を探して表示したり、ローカル・システムにダウンロードしたりするときに便利です。このコマンドを実行すると、CRL を発行した CA（発行者）およびディレクトリの CRL サブツリー内の場所（DN）が出力されます。

Oracle Internet Directory 内の CRL を一覧表示するには、コマンドラインで次のように入力します。

```
orapki crl list -ldap hostname:ssl_port
```

このコマンドの *hostname* および *ssl_port* は、ディレクトリがインストールされているシステムのホスト名と SSL ポートです。前述のとおり、これは認証なしのディレクトリ SSL ポートです。

11.2.5.2.4 Oracle Internet Directory 内の CRL の表示 Oracle Internet Directory に格納されている特定の CRL を要約形式で表示できます。また、特定の CRL に対応する失効済証明書の完全なリストを表示することもできます。要約形式のリストには、CRL の発行者名とその有効期間が表示されます。完全なリストには、特定の CRL に含まれているすべての失効済証明書が一覧表示されます。

Oracle Internet Directory 内の CRL の要約リストを表示するには、コマンドラインで次のように入力します。

```
orapki crl display -crl crl_location [-wallet wallet_location] -summary
```

このコマンドの *crl_location* は、ディレクトリ内の CRL の場所です。`orapki crl list` コマンドの使用時に表示されるリストから CRL の場所を貼り付けると便利です。第 11.2.5.2.3 項「Oracle Internet Directory に格納されている CRL の一覧表示」を参照してください。

Oracle Internet Directory に格納されている特定の CRL の、すべての失効済証明書を一覧表示するには、コマンドラインで次のように入力します。

```
orapki crl display -crl crl_location [-wallet wallet_location] -complete
```

たとえば、次のような orapki コマンドの場合、

```
orapki crl display -crl $T_WORK/pki/wlt_crl/nzcrl.txt -wallet $T_WORK/pki/wlt_crl
-complete
```

出力は次のようになります。CRL 発行者の DN、発行日、次回更新日および CRL 内の失効済証明書が表示されます。

```
issuer = CN=root,C=us, thisUpdate = Sun Nov 16 10:56:58 PST 2003,
nextUpdate = Mon Sep 30 11:56:58 PDT 2013, revokedCertificates =
{(serialNo = 153328337133459399575438325845117876415,
revocationDate - Sun Nov 16 10:56:58 PST 2003)}
CRL is valid
```

orapki crl display コマンドの実行時に -wallet オプションを使用すると、CRL が CA の証明書と比較して検証されます。

CRL のサイズによっては、-complete オプションを指定すると表示に時間がかかることがあります。

ディレクトリ内の CRL を表示するには、Oracle Internet Directory 付属のグラフィカル・ユーザー・インタフェース・ツールである Oracle Directory Manager を使用することもできます。CRL は、次に示すディレクトリ内の場所に格納されています。

```
cn=CRLValidation,cn=Validation,cn=PKI,cn=Products,cn=OracleContext
```

11.2.5.2.5 Oracle Internet Directory からの CRL の削除 orapki を使用してディレクトリから CRL を削除するユーザーは、ディレクトリ・グループ CRLAdmins のメンバーである必要があります。このディレクトリ管理グループの詳細は、第 11.2.5.2.2 項「Oracle Internet Directory への CRL のアップロード」を参照してください。

CRL をディレクトリから削除するには、コマンドラインで次のように入力します。

```
orapki crl delete -issuer issuer_name -ldap hostname:ssl_port
-user username [-summary]
```

このコマンドの *issuer_name* は CRL を発行した CA の名前、*hostname* および *ssl_port* はディレクトリがインストールされているシステムのホスト名と SSL ポート、*username* は CRL サブツリーから CRL を削除する権限を持つディレクトリ・ユーザーです。これは認証なしのディレクトリ SSL ポートであることに注意してください。このポートの詳細は、第 11.2.5.2.2 項「Oracle Internet Directory への CRL のアップロード」を参照してください。

-summary オプションを使用すると、削除された CRL の LDAP エントリが出力されます。

たとえば、次のような orapki コマンドの場合、

```
orapki crl delete -issuer "CN=root,C=us"
-ldap machine1:3500 -user cn=orcladmin -summary
```

出力は次のようになります。ディレクトリ内の削除された CRL の場所が一覧表示されます。

```
Deleted CRL at cn=root
cd45860c.rN,cn=CRLValidation,cn=Validation,cn=PKI,cn=Products,cn=OracleContext
```

11.2.6 orapki ユーティリティのコマンドの要約

この項では、次の orapki コマンドについて説明します。

11.2.6.1 orapki cert create

次の各項では、このコマンドについて説明します。

11.2.6.1.1 目的 テスト用に署名付き証明書を作成するときに使用します。

11.2.6.1.2 構文

```
orapki cert create [-wallet wallet_location]
-request certificate_request_location
-cert certificate_location -validity number_of_days [-summary]
```

- -wallet パラメータでは、証明書リクエストに署名するために使用されるユーザー証明書および秘密鍵を含む Wallet を指定します。
- -request パラメータ（必須）では、作成する証明書の証明書リクエストの場所を指定します。
- -cert パラメータ（必須）では、新しい署名付き証明書の出力先となるディレクトリの場所を指定します。
- -validity パラメータ（必須）では、この証明書の有効期間を、実行日からの日数として指定します。

11.2.6.2 orapki cert display

次の各項では、このコマンドについて説明します。

11.2.6.2.1 目的 特定の証明書の詳細を表示するときに使用します。

11.2.6.2.2 構文

```
orapki cert display -cert certificate_location [-summary|-complete]
```

- -cert パラメータでは、表示する証明書の場所を指定します。
- -summary パラメータまたは -complete パラメータを使用すると、次の情報を表示できます。
  - -summary を指定すると、証明書とその有効期限が表示されます。
  - -complete を指定すると、シリアル番号や公開鍵など、その他の証明書情報も表示されます。

11.2.6.3 orapki crl delete

次の各項では、このコマンドについて説明します。

11.2.6.3.1 目的 Oracle Internet Directory から CRL を削除するときに使用します。orapki を使用してディレクトリから CRL を削除するユーザーは、CRLAdmins（cn=CRLAdmins,cn=groups,%s_OracleContextDN%）ディレクトリ・グループのメンバーである必要があります。

11.2.6.3.2 構文

```
orapki crl delete -issuer issuer_name
-ldap hostname:ssl_port -user username [-summary]
```

- -issuer パラメータでは、CRL を発行した認証局（CA）の名前を指定します。
- -ldap パラメータでは、削除する CRL が存在するディレクトリのホスト名および SSL ポートを指定します。これは認証なしのディレクトリ SSL ポートであることに注意してください。このポートの詳細は、第 11.2.5.2.2 項「Oracle Internet Directory への CRL のアップロード」を参照してください。
- -user パラメータでは、ディレクトリ内の CRL サブツリーから CRL を削除する権限を持つディレクトリ・ユーザーのユーザー名を指定します。
- -summary パラメータは省略可能です。このパラメータを使用すると、削除された CRL の LDAP エントリが出力されます。

11.2.6.4 orapki crl display

次の各項では、このコマンドについて説明します。

11.2.6.4.1 目的 Oracle Internet Directory 内に格納されている特定の CRL を表示するときに使用します。

11.2.6.4.2 構文

```
orapki crl display -crl crl_location
[-wallet wallet_location] [-summary|-complete]
```

- -crl パラメータでは、ディレクトリ内の CRL の場所を指定します。orapki crl list コマンドの使用時に表示されるリストから CRL の場所を貼り付けると便利です。第 11.2.6.6 項「orapki crl list」を参照してください。
- -wallet パラメータ（オプション）では、CRL を発行した認証局（CA）の証明書を含む Wallet の場所を指定します。このパラメータを使用すると、CRL を表示する前に、CA の証明書と比較して CRL の有効性が検査されます。
- -summary パラメータまたは -complete パラメータを選択すると、次の情報が表示されます。
  - -summary を指定すると、CRL の発行者名と CRL の有効期間も表示されます。
  - -complete を指定すると、その CRL に含まれているすべての失効済証明書が一覧表示されます。CRL のサイズによっては、このオプションを指定すると表示に時間がかかることがあります。

11.2.6.5 orapki crl hash

次の各項では、このコマンドについて説明します。

11.2.6.5.1 目的 証明書失効リスト（CRL）発行者のハッシュ値を生成するときに使用します。これにより、証明書の検証時にファイル・システム内の CRL の場所を特定します。

11.2.6.5.2 構文 `orapki crl hash -crl crl_filename|URL [-wallet wallet_location] [-symlink|-copy] crl_directory [-summary]`

- `-crl` パラメータでは、CRL が含まれるファイルの名前、または CRL が存在する場所の URL を指定します。
- `-wallet` パラメータ（オプション）では、CRL を発行した認証局（CA）の証明書を含む Wallet の場所を指定します。このパラメータを使用すると、CRL をディレクトリへアップロードする前に、CA の証明書と比較して CRL の有効性が検査されます。
- 次に示すように、オペレーティング・システムに応じて、`-symlink` パラメータまたは `-copy` パラメータのいずれかを使用します。
  - （UNIX の場合）`-symlink` を使用すると、*crl_directory* で指定した場所にある CRL へのシンボリック・リンクが作成されます。
  - （Windows の場合）`-copy` を使用すると、*crl_directory* で指定した場所にある CRL のコピーが作成されます。
- `-summary` パラメータ（オプション）を指定すると、CRL の発行者名が表示されます。

11.2.6.6 orapki crl list

次の各項では、このコマンドについて説明します。

11.2.6.6.1 目的 Oracle Internet Directory 内に格納されている CRL を一覧表示するときに使用します。リストから特定の CRL を探して表示したり、ローカル・ファイル・システムにダウンロードしたりするときに便利です。

11.2.6.6.2 構文 `orapki crl list -ldap hostname:ssl_port`

`-ldap` パラメータでは、一覧表示する CRL が存在するディレクトリ・サーバーのホスト名および SSL ポートを指定します。これは認証なしのディレクトリ SSL ポートであることに注意してください。このポートの詳細は、第 11.2.5.2.2 項「Oracle Internet Directory への CRL のアップロード」を参照してください。

11.2.6.7 orapki crl upload

次の各項では、このコマンドについて説明します。

11.2.6.7.1 目的 証明書失効リスト（CRL）を Oracle Internet Directory 内の CRL サブツリーにアップロードするときに使用します。CRL をディレクトリにアップロードするには、ディレクトリ管理グループ `CRLAdmins`（`cn=CRLAdmins,cn=groups,%s_OracleContextDN%`）のメンバーである必要があります。

11.2.6.7.2 構文 `orapki crl upload -crl crl_location -ldap hostname:ssl_port -user username [-wallet wallet_location] [-summary]`

- `-crl` パラメータでは、ディレクトリにアップロードする CRL が存在するディレクトリの場所または URL を指定します。
- `-ldap` パラメータでは、CRL のアップロード先であるディレクトリのホスト名および SSL ポートを指定します。これは認証なしのディレクトリ SSL ポートであることに注意してください。このポートの詳細は、第 11.2.5.2.2 項「Oracle Internet Directory への CRL のアップロード」を参照してください。
- `-user` パラメータでは、ディレクトリ内の CRL サブツリーに CRL を追加する権限を持つディレクトリ・ユーザーのユーザー名を指定します。
- `-wallet` パラメータでは、CRL を発行した認証局（CA）の証明書を含む Wallet の場所を指定します。このパラメータは省略可能です。このパラメータを使用すると、CRL をディレクトリへアップロードする前に、CA の証明書と比較して CRL の有効性が検査されます。

- -summary パラメータも省略可能です。このパラメータを使用すると、CRL の発行者名およびディレクトリ内に CRL が格納されている LDAP エントリが出力されます。

11.2.6.8 orapki wallet add

次の各項では、このコマンドについて説明します。

11.2.6.8.1 目的 証明書リクエストおよび証明書を Oracle Wallet に追加するときに使用します。

11.2.6.8.2 構文 証明書リクエストを追加するには:

```
orapki wallet add -wallet wallet_location -dn user_dn -keySize 512|1024|2048
```

- -wallet パラメータでは、証明書リクエストの追加先である Wallet の場所を指定します。
- -dn パラメータでは、証明書の所有者の識別名を指定します。
- -keySize パラメータでは、証明書の鍵サイズを指定します。
- リクエストの署名およびエクスポート・オプションによるリクエストのエクスポートについては、第 11.2.6.11 項「orapki wallet export」を参照してください。

信頼できる証明書を追加するには:

```
orapki wallet add -wallet wallet_location -trusted_cert -cert certificate_location
```

- -trusted_cert パラメータを指定すると、-cert で指定した場所にある信頼できる証明書が Wallet に追加されます。

ルート証明書を追加するには:

```
orapki wallet add -wallet wallet_location -dn
certificate_dn -keySize 512|1024|2048 -self_signed -validity number_of_days
```

- -self_signed パラメータを指定すると、ルート証明書が作成されます。
- -validity パラメータは必須です。このパラメータでは、ルート証明書の有効期間を、実行日からの日数として指定します。

ユーザー証明書を追加するには:

```
orapki wallet add -wallet wallet_location -user_cert -cert certificate_location
```

- -user_cert パラメータを指定すると、-cert パラメータで指定した場所にあるユーザー証明書が Wallet に追加されます。Wallet にユーザー証明書を追加する前に、証明連鎖を構成するすべての信頼できる証明書を追加する必要があります。ユーザー証明書を追加する前に、Wallet にすべての信頼できる証明書がインストールされていない場合、ユーザー証明書の追加に失敗します。

11.2.6.9 orapki wallet create

次の各項では、このコマンドについて説明します。

11.2.6.9.1 目的 Oracle Wallet を作成するときや、Oracle Wallet の自動ログインを有効化するときに使用します。

11.2.6.9.2 構文 `orapki wallet create -wallet wallet_location [-auto_login]`

- -wallet パラメータでは、新しい Wallet を作成する場所、または自動ログインを有効にする Wallet の場所を指定します。
- -auto_login パラメータを指定すると、自動ログイン Wallet が作成されます。または、-wallet オプションで指定した Wallet の自動ログインが有効になります。自動ログイン Wallet の詳細は、第 11.1.4.14 項「自動ログインの使用」を参照してください。

11.2.6.10 orapki wallet display

次の各項では、このコマンドについて説明します。

11.2.6.10.1 目的 Oracle Wallet 内の証明書リクエスト、ユーザー証明書および信頼できる証明書を表示するときに使用します。

11.2.6.10.2 構文 `orapki wallet display -wallet wallet_location`

- `-wallet` パラメータでは、開こうとする Wallet が現在の作業用ディレクトリにない場合に、Wallet の場所を指定します。

11.2.6.11 orapki wallet export

次の各項では、このコマンドについて説明します。

11.2.6.11.1 目的 証明書リクエストおよび証明書を Oracle Wallet からエクスポートするときに使用します。

11.2.6.11.2 構文 `orapki wallet export -wallet wallet_location -dn certificate_dn -cert certificate_filename`

- `-wallet` パラメータでは、証明書のエクスポート元である Wallet の場所を指定します。
- `-dn` パラメータでは、証明書の識別名を指定します。
- `-cert` パラメータでは、エクスポートされた証明書を含むファイルの名前を指定します。

Oracle Wallet から証明書リクエストをエクスポートするには :

`orapki wallet export -wallet wallet_location -dn certificate_request_dn -request certificate_request_filename`

- `-request` パラメータでは、エクスポートされた証明書リクエストを含むファイルの名前を指定します。

11.3 X.509 証明書との相互運用性

Oracle Wallet Manager の機能は、すでに証明書が割り当てられているユーザーをサポートしています。Oracle Wallet Manager を使用して証明書を作成しない場合でも、作成済の証明書の管理および格納に Oracle Wallet Manager を使用できます。

11.3.1 公開鍵暗号規格（PKCS）のサポート

Oracle Wallet Manager は、X.509 証明書および秘密鍵を公開鍵暗号規格（PKCS）#12 形式で格納します。また、RSA Laboratories によって開発された PKCS #10 仕様に従って証明書リクエストを生成します。そのため、Oracle Wallet の構造と、サポートされているサード・パーティの PKI アプリケーションとの相互運用性が確保され、異なるオペレーティング・システム間でも Wallet が移植可能になります。

Oracle Wallet Manager の Wallet は、PKCS #11 仕様に準拠する API を使用して資格証明をハードウェア・セキュリティ・モジュール上に格納するように設定できます。Wallet の作成時に Wallet のタイプとして `PKCS11` が選択されている場合は、その Wallet に格納される鍵はすべてハードウェア・セキュリティ・モジュールまたはトークン（秘密鍵の格納または暗号化操作の実行、またはその両方を行う、スマートカード、PCMCIA カード、スマート・ディスケットなどのポータブル・ハードウェア・デバイス）に保存されます。

関連項目：

- 第 11.1.5.1.3 項「サード・パーティ製ツールによって作成された証明書のインポート」
- 第 11.1.4.5 項「サード・パーティ環境への Oracle Wallet のエクスポート」
- 第 11.1.4.2.2 項「ハードウェア・セキュリティ・モジュールに資格証明を格納する Wallet の作成」
- PKCS 規格に関するドキュメントを参照するには、次の URL にアクセスしてください。

  `http://www.rsasecurity.com/rsalabs/`

11.3.2 複数の証明書のサポート

Oracle Wallet Manager では、個々の Wallet に複数の証明書を格納できます。これによって、次のような Oracle PKI 証明書使用がサポートされます。

- SSL
- S/MIME 署名
- S/MIME 暗号化
- コード署名
- CA 証明書署名

Oracle Wallet Manager では、1 つのデジタル・エンティティの証明書を複数サポートしています。各証明書は一連の Oracle PKI 証明書での使用に利用できますが、1 つの証明書をすべての使用に利用できるわけではありません（使用の正当な組合せについては、表 11-4 および表 11-5 を参照）。証明書リクエストと証明書は、1 対 1 でマッピングされている必要があります。同一の証明書リクエストを使用して複数の証明書を取得することはできますが、各証明書リクエストの複数の証明書を、同時に同じ Wallet にインストールすることはできません。

Oracle Wallet Manager では、X.509 バージョン 3 の `KeyUsage` 拡張タイプを使用して Oracle PKI 証明書使用を定義します。鍵使用目的拡張のタイプはそれぞれ、証明書内で設定されるオプションのビットです。このビットを設定することで、証明書の鍵をどのような目的に使用できるか定義します。証明書を発行するとき、認証局では要求された証明書のタイプに応じてこれらのビットを設定します。表 11-4 ではこれらの鍵の用途について説明しています。

表 11-4 X.509 バージョン 3 の KeyUsage 拡張タイプ、値および説明

KeyUsage 拡張タイプ	値	説明
digitalSignature	0	エンティティ認証、およびデータ発信者認証に使用します。
nonRepudiation	1	署名するエンティティが事実と異なってアクションを否定することを防止するために使用します。
keyEncipherment	2	サブジェクトの公開鍵が鍵の転送に使用される場合に使用します。
dataEncipherment	3	サブジェクトの公開鍵が暗号化鍵以外のデータの暗号化に使用される場合に使用します。
keyAgreement	4	サブジェクトの公開鍵が SSL 接続ネゴシエーション時の鍵合意に使用される場合に使用します。
keyCertSign	5	サブジェクトの公開鍵が証明書の署名の検証に使用される場合に使用します。CA 証明書内でのみ使用できます。
cRLSign	6	サブジェクトの公開鍵が証明書失効リストの署名の検証に使用される場合に使用します。
encipherOnly	7	encipherOnly ビットが設定されている場合は、keyAgreement ビットも設定する必要があります。この 2 つのビットが設定されている場合は、サブジェクトの公開鍵は鍵合意実行中のデータの暗号化にのみ使用できます。
decipherOnly	8	encipherOnly ビットの場合と同様に、decipherOnly ビットが設定されている場合は、keyAgreement ビットも設定する必要があります。decipherOnly と keyAgreement の 2 つのビットが設定されているときは、サブジェクトの公開鍵は鍵合意実行中のデータ暗号化解除にのみ使用できます。

関連項目： KeyUsage 拡張タイプの説明は、次の URL にある Internet Engineering Task Force の RFC #2459『Internet X.509 Public Key Infrastructure Certificate and CRL Profile』を参照してください。

```
http://www.ietf.org/rfc/rfc2459.txt
```

Oracle Wallet Manager で証明書（ユーザー証明書または信頼できる証明書）をインストールするとき、表 11-4 および表 11-5 に示すとおり、KeyUsage 拡張の値が Oracle PKI 証明書使用にマッピングされます。

表 11-5 Oracle Wallet Manager による信頼できる証明書の Oracle Wallet へのインポート

KeyUsage の値	クリティカルか否か[1]	用途
なし	該当せず	インポート可能
5 を除く組合せ	○	インポート不可能
5 を除く組合せ	×	インポート可能
5 のみ、または 5 を含む組合せ	該当せず	インポート可能

[1] KeyUsage 拡張がクリティカルである場合は、その証明書を他の目的に使用することはできません。

必要な Oracle PKI 証明書使用に該当する KeyUsage 値で、認証局から証明書を取得します。1 つの Wallet に、同じ用途の鍵ペアが複数含まれることもあります。表 11-4 および表 11-5 に示すように、各証明書では Oracle PKI 証明書使用を複数サポートできます。Oracle PKI アプリケーションでは、必要な PKI 証明書使用を含んでいる最初の証明書を使用します。

たとえば、SSL で使用する場合、SSL Oracle PKI 証明書使用を含む最初の証明書が使用されます。

SSL で使用する証明書が 1 つもない場合は、ORA-28885 エラー（「必須の鍵使用方法のある証明書が見つかりません」）が返されます。

12

Infrastructure での SSL の有効化

この章では、Oracle Application Server Infrastructure インストールで SSL を有効にする方法について説明します。

注意： この章では、次の Oracle Application Server 製品を参照する情報は、リリース 10.1.4、リリース 2（10.1.2）またはそれ以前のソフトウェアにのみ該当します。

- OracleAS Infrastructure
- OracleAS Single Sign-On
- OracleAS Web Cache
- OracleAS Certificate Authority
- Oracle Delegated Administration Services
- Oracle Identity Management

この章の項目は次のとおりです。

12.1 Infrastructure での SSL 通信経路

この項では、Oracle Application Server Infrastructure で使用されるすべての SSL 通信経路を特定し、Oracle Application Server ドキュメント・ライブラリ内のコンポーネント・ガイドに記載されている構成手順への相互参照を示します。

注意： Oracle Identity Management をインストールする際に、Oracle Internet Directory のモードを選択するよう求められます。デフォルトのモードはデュアル・モードです。デュアル・モードでは、一部のコンポーネントは非 SSL 接続を使用して Oracle Internet Directory にアクセスできます。インストール時に SSL モードを選択した場合、インストールしたすべてのコンポーネントは、ディレクトリへの接続時に SSL を使用する必要があります。

SSL の構成を開始する前に、Oracle Internet Directory のモードを指定してください。`oidadmin` ツールを起動し、Oracle Directory Manager で SSL モードを表示します。ディレクトリ・サーバーに移動し、「**プロパティの表示**」→「**SSL 設定**」を選択します。

Oracle Application Server Infrastructure で使用される各通信経路と、それらに関連する SSL の構成手順を次に示します。

- **Oracle HTTP Server から OC4J_SECURITY インスタンス**

  SSL を介した AJP 通信を構成するには、mod_oc4j と `iaspt` デーモンの通信を構成する必要があります。この通信を構成するには、『Oracle HTTP Server 管理者ガイド』の mod_oc4j での SSL 使用の構成に関する項の手順に従ってください。

- **Oracle HTTP Server から iaspt（ポート・トンネリング）を経由し、OC4J_SECURITY インスタンス**

  SSL 用にこの接続経路を構成するには、『Oracle HTTP Server 管理者ガイド』の「ポート・トンネリングの概要」の手順に従ってください。

- **OC4J_SECURITY インスタンスから Oracle Internet Directory**

  SSL 用にこの接続経路を構成するには、『Oracle Application Server Single Sign-On 管理者ガイド』の「シングル・サインオン中間層での SSL の有効化」の手順に従ってください。このマニュアルでは、ブラウザと OracleAS Single Sign-On サーバー間での SSL 通信の構成方法が説明されています。

  Oracle HTTP Server に SSL を構成すると、Oracle Delegated Administration Services が SSL 対応になります。Oracle Delegated Administration Services の Oracle Internet Directory への通信は常に SSL 対応です。対応させるために、特別な構成作業を行う必要はありません（OracleAS Single Sign-On、Oracle Application Server Certificate Authority および Oracle Delegated Administration Services は、デフォルトでは SSL モードで Oracle Internet Directory と通信します）。

- **Oracle Directory Integration and Provisioning から Oracle Internet Directory、および Oracle Internet Directory レプリケーション・サーバーから Oracle Internet Directory**

  図 12-1 に示すように、様々なコンポーネントおよび通信経路に SSL を構成できます。それぞれの構成手順の参照先は次のとおりです。

  – Oracle Internet Directory レプリケーション・サーバーと Oracle Internet Directory サーバー間の通信：『Oracle Internet Directory 管理者ガイド』の「Secure Sockets Layer（SSL）と Oracle Internet Directory レプリケーション」

  – Oracle Directory Integration and Provisioning と Oracle Internet Directory サーバー間の通信：『Oracle Identity Management 統合ガイド』の「Oracle Directory Integration and Provisioning Server の管理」

- **OC4J_SECURITY インスタンスから Metadata Repository データベース、および Oracle Internet Directory から Metadata Repository データベース**

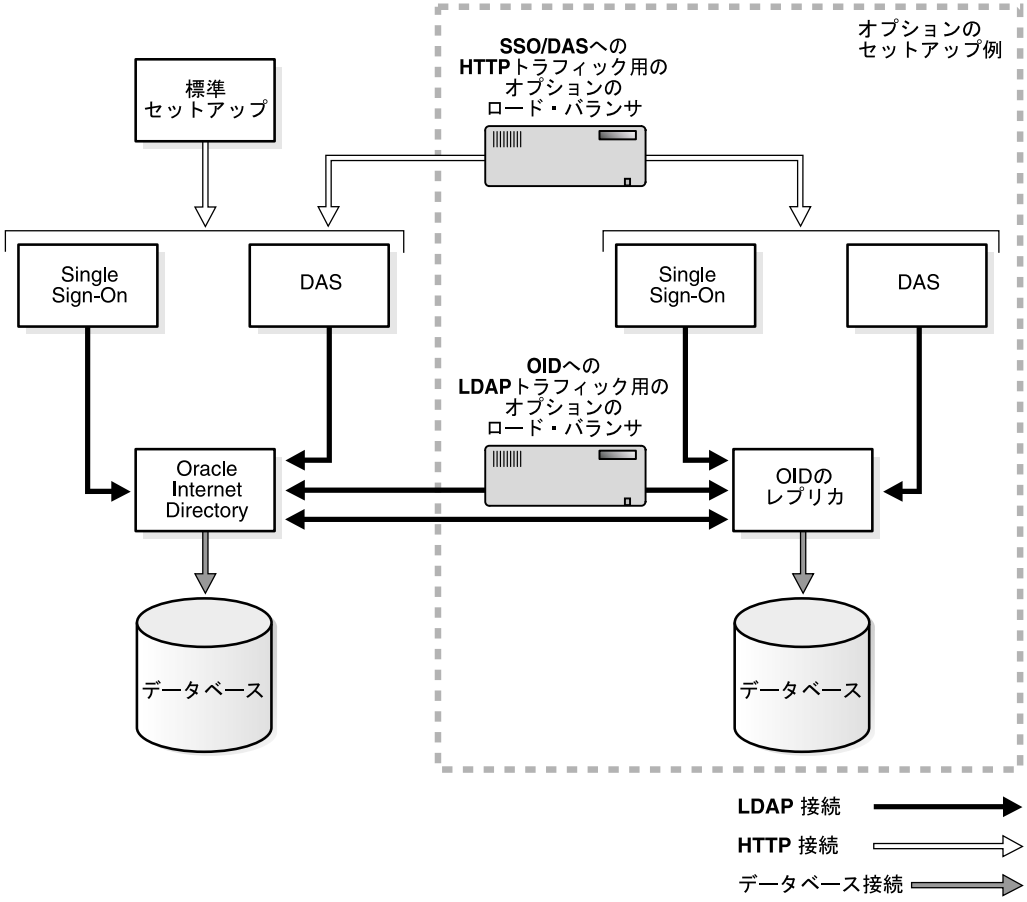
  Oracle Internet Directory が指定された SSL ポートで SSL 接続を許可するように構成されている場合、アプリケーションをリクエストする JDBC URL では、次のように SSL プロトコルと SSL ポートのみを指定する必要があります。

  ldaps://*host:sslport*/...

  セキュアな接続を使用している場合は、プロトコルの名前に s を付ける必要があることに注意してください。たとえば、ldap ではなく ldaps を使用します。

  Oracle Internet Directory が SSL ポートで SSL 接続を許可するように構成されていない場合は、構成を変更する必要があります。『Oracle Internet Directory 管理者ガイド』の「Secure Sockets Layer（SSL）とディレクトリ」を参照してください。

図 12-1 Oracle Identity Management コンポーネントと SSL 接続経路



12.2 推奨される SSL 構成

『Oracle Application Server セキュリティ・ガイド』では、セキュリティの概念について詳しく説明されており、様々な構成でのセキュリティ構成の推奨事項が示されています。「推奨される配置トポロジ」の章では、アーキテクチャの例が示されています。SSL を有効にする必要のあるコンポーネントを特定したら、この章および第 13 章「中間層での SSL の有効化」で説明している方法に従って、コンポーネントを構成します。

OracleAS Single Sign-On および Oracle Delegated Administration Services での SSL の構成は、通常、推奨の配置トポロジで行います（第 12.3.1 項「OracleAS Single Sign-On および Oracle Delegated Administration Services に対する SSL の構成」を参照）。すべての Infrastructure 通信経路での SSL の構成方法は、第 12.1 項「Infrastructure での SSL 通信経路」で説明しています。

12.3 一般的な SSL 構成作業

この項では、個別のコンポーネントで SSL を構成する方法について説明している、Oracle Application Server ドキュメント・ライブラリ内のコンポーネントのマニュアルへの参照を示します。

12.3.1 OracleAS Single Sign-On および Oracle Delegated Administration Services に対する SSL の構成

次のコンポーネント間に SSL を構成するには、『Oracle Application Server Single Sign-On 管理者ガイド』の手順に従ってください。

- ブラウザと OracleAS Single Sign-On サーバー（「シングル・サインオン中間層での SSL の有効化」）
- OracleAS Single Sign-On サーバーと Oracle Internet Directory サーバー（Single Sign-On と Oracle Internet Directory 間の SSL の構成に関する項）

Oracle HTTP Server に SSL を構成すると、Oracle Delegated Administration Services が SSL 対応になります（「シングル・サインオン中間層での SSL の有効化」で説明されています）。Oracle Delegated Administration Services の Oracle Internet Directory への通信は常に SSL 対応です。対応させるために、特別な構成作業を行う必要はありません

12.3.2 Oracle Internet Directory に対する SSL の構成

Oracle Internet Directory での SSL 通信の構成手順は、次のマニュアルに説明されています。

- 『Oracle Internet Directory 管理者ガイド』の「Secure Sockets Layer（SSL）とディレクトリ」
- 『Oracle Internet Directory 管理者ガイド』の「SSL パラメータの構成」
- 『Oracle Internet Directory 管理者ガイド』の「10*g* リリース 2（10.1.2）での SSL の使用制限事項」

12.3.3 Oracle Internet Directory レプリケーション・サーバーと Oracle Directory Integration and Provisioning に対する SSL の構成

図 12-1 に示すように、様々なコンポーネントおよび通信経路に SSL を構成できます。それぞれの構成手順の参照先は次のとおりです。

- Oracle Internet Directory レプリケーション・サーバーと Oracle Internet Directory サーバー間の通信：『Oracle Internet Directory 管理者ガイド』の「Secure Sockets Layer（SSL）と Oracle Internet Directory レプリケーション」
- Oracle Directory Integration and Provisioning と Oracle Internet Directory サーバー間の通信：『Oracle Identity Management 統合ガイド』の「Oracle Directory Integration and Provisioning Server の管理」

12.3.4 Identity Management データベースでの SSL の構成

SSL 通信を Identity Management データベースに構成するには、『Oracle Application Server Single Sign-On 管理者ガイド』の「Identity Management インフラストラクチャ・データベースの再構成」の手順に従ってください。

12.3.5 OC4J_SECURITY インスタンスでの追加の SSL 構成

この項では、mod_oc4j および OC4J の SSL 構成情報への参照を示します。

12.3.5.1 mod_oc4j から OC4J_SECURITY への SSL の構成

SSL を介した AJP 通信を構成するには、mod_oc4j と `iaspt` デーモンの通信を構成する必要があります。この通信を構成するには、『Oracle HTTP Server 管理者ガイド』の「mod_oc4j と OC4J 間での SSL の有効化」の手順に従ってください。

12.3.5.2 mod_oc4j から OC4J_SECURITY インスタンスへのポート・トンネリングの使用

SSL 用にこの接続経路を構成するには、『Oracle HTTP Server 管理者ガイド』の「ポート・トンネリングの概要」の手順に従ってください。

12.3.5.3 JDBC/SSL（ASO サポート）の構成

Oracle Internet Directory が指定された SSL ポートで SSL 接続を許可するように構成されている場合、アプリケーションをリクエストする JDBC URL では、次のように SSL プロトコルと SSL ポートのみを指定する必要があります。

```
ldaps://host:sslport/...
```

セキュアな接続を使用している場合は、プロトコルの名前に s を付ける必要があることに注意してください（たとえば、`ldap` ではなく `ldaps` を使用します）。

Oracle Internet Directory が SSL ポートで SSL 接続を許可するように構成されていない場合は、構成を変更する必要があります。『Oracle Internet Directory 管理者ガイド』の「Secure Sockets Layer（SSL）とディレクトリ」を参照してください。

12.3.6 Oracle Application Server Certificate Authority での SSL

Oracle Application Server Certificate Authority の SSL は、デフォルトで有効になっているため、このコンポーネントについての構成作業はありません。

> **ヒント：** OracleAS Certificate Authority は、Oracle Identity Management のユーザーが証明書をプロビジョニングする作業を単純化します（証明書は SSO 認証ユーザーに対して自動的にプロビジョニングされます）。
>
> OCA OracleAS Single Sign-On を使用して証明書ベースの認証を有効にするには、『Oracle Application Server Certificate Authority 管理者ガイド』を参照してください。OracleAS Single Sign-On への証明書ベースの認証を有効にするには、『Oracle Application Server Single Sign-On 管理者ガイド』を参照してください。

12.3.7 Oracle Enterprise Manager 10*g* に対する SSL の構成

Oracle Enterprise Manager 10*g* は、Grid Control コンソールと Application Server Control コンソールの 2 つのコンポーネントで構成されています。これらにはそれぞれ SSL 通信を構成できます。

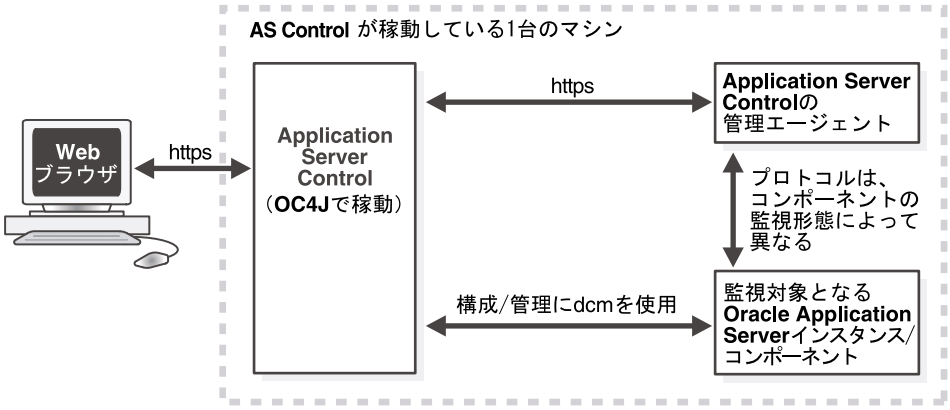
12.3.7.1 Grid Control のセキュリティの構成

『Oracle Enterprise Manager 構成ガイド』の Grid Control のセキュリティ（SSL および HTTPS）の構成に関する項の手順に従います。

12.3.7.2 Application Server Control コンソールのセキュリティの構成

Application Server Control コンソールの SSL 構成に関連する通信経路は、図 12-2 に **https** として示されています。

図 12-2 Oracle Enterprise Manager 10*g* の SSL 接続経路



Web ブラウザと Application Server Control コンソール間、および Application Server Control コンソールと Oracle Application Server コンポーネント間の通信を保護するには、第 A.3 項「Application Server Control コンソールのセキュリティの構成」を参照してください。

13

中間層での SSL の有効化

この章では、Oracle Application Server 中間層インストールで SSL を有効にする方法について説明します。

注意： この章では、次の Oracle Application Server 製品を参照する情報は、リリース 10.1.4、リリース 2（10.1.2）またはそれ以前のソフトウェアにのみ該当します。

- OracleAS Single Sign-On
- OracleAS Web Cache
- Oracle Internet Directory

この章の項目は次のとおりです。

13.1 中間層での SSL 通信経路

この項では、Oracle Application Server 中間層のインストール・タイプで使用するすべての SSL 通信経路を特定し、Oracle Application Server ドキュメント・ライブラリ内のコンポーネントのマニュアルに記載されている構成手順への相互参照を示します。

Oracle Application Server 中間層で使用される各通信経路と、それらに関連する SSL の構成手順を次に示します。

- **外部クライアントまたはロード・バランサから Oracle HTTP Server**

  Oracle HTTP Server に SSL を構成するには、『Oracle HTTP Server 管理者ガイド』の「SSL の有効化」の手順に従ってください。

- **外部クライアントまたはロード・バランサから OracleAS Web Cache**

  OracleAS Web Cache に SSL を構成するには、『Oracle Application Server Web Cache 管理者ガイド』の「HTTPS リクエストをサポートするための OracleAS Web Cache の構成」の手順に従ってください。

- **OracleAS Web Cache から Oracle HTTP Server**

  OracleAS Web Cache に SSL を構成するには、『Oracle Application Server Web Cache 管理者ガイド』の「HTTPS リクエストをサポートするための OracleAS Web Cache の構成」の手順に従ってください。

- **Oracle HTTP Server から OC4J アプリケーション（AJP）**

  SSL を介した AJP 通信を構成するには、mod_oc4j と `iaspt` デーモンの通信を構成する必要があります。この通信を構成するには、『Oracle HTTP Server 管理者ガイド』の mod_oc4j での SSL 使用の構成に関する項の手順に従ってください。

- **Oracle HTTP Server から iaspt を経由して OC4J**

  SSL 用にこの接続経路を構成するには、『Oracle HTTP Server 管理者ガイド』の「ポート・トンネリングの概要」の手順に従ってください。

- **OC4J（JAAS プロバイダ）から Oracle Internet Directory**

  プロバイダを構成するには、『Oracle Containers for J2EE セキュリティ・ガイド』の手順に従ってください。プロバイダに SSL を構成するには、`SSL_ONLY_FLAG` を `true` に設定します。

- **OC4J からデータベース（ASO）**

  Oracle Internet Directory が指定された SSL ポートで SSL 接続を許可するように構成されている場合、アプリケーションをリクエストする JDBC URL では、次のように SSL プロトコルと SSL ポートのみを指定する必要があります。

  `ldaps://`*`host.sslport`*`/...`

  セキュアな接続を使用している場合は、プロトコルの名前に s を付ける必要があることに注意してください。たとえば、ldap ではなく ldaps を使用します。

  Oracle Internet Directory が SSL ポートで SSL 接続を許可するように構成されていない場合は、構成を変更する必要があります。『Oracle Internet Directory 管理者ガイド』の「Secure Sockets Layer（SSL）とディレクトリ」を参照してください。

- **ORMI（Oracle Remote Method Invocation、カスタム・ワイヤ・プロトコル）over SSL**

  SSL 用にこの接続経路を構成するには、『Oracle Containers for J2EE セキュリティ・ガイド』を参照してください。

- **スタンドアロン OC4J での SSL（HTTPS）**

  SSL 用にこの接続経路を構成するには、『Oracle Containers for J2EE セキュリティ・ガイド』の OC4J での SSL の構成に関する項の手順に従ってください。SSL を使用してクライアントと OC4J インスタンス間の通信を保護する方法について説明しています。

- **OracleAS Portal Parallel Page Engine（OC4J_PORTAL インスタンス内のサーブレット）から OracleAS Web Cache（HTTPS）**

  SSL 用にこの接続経路を構成するには、『Oracle Containers for J2EE セキュリティ・ガイド』の OC4J での SSL の構成に関する項の手順に従ってください。

13.2 推奨される SSL 構成

『Oracle Application Server セキュリティ・ガイド』では、セキュリティの概念について詳しく説明されており、様々な構成でのセキュリティ構成の推奨事項が示されています。「推奨される配置トポロジ」の章では、インストール・タイプに対するアーキテクチャの例が示されています。SSL を有効にする必要のあるコンポーネントを特定したら、この章および第 12 章「Infrastructure での SSL の有効化」で説明している方法に従って、コンポーネントを構成します。

13.3 中間層の一般的な SSL 構成作業

この項では、Oracle Application Server 中間層インストール・タイプで一般的に使用される一部の SSL 構成を特定し、Oracle Application Server ドキュメント・ライブラリ内のコンポーネントのマニュアルに記載されている構成手順への相互参照を示します。

13.3.1 OracleAS Web Cache での SSL の有効化

OracleAS Web Cache は、Oracle Application Server 中間層インストールの構成要素です。これに SSL を構成するには、『Oracle Application Server Web Cache 管理者ガイド』の「HTTPS リクエストをサポートするための OracleAS Web Cache の構成」の手順に従ってください。

13.3.2 Oracle HTTP Server での SSL の有効化

Oracle HTTP Server は、あらゆる Oracle Application Server 中間層インストールの構成要素です。これに SSL を構成するには、『Oracle HTTP Server 管理者ガイド』の「SSL の有効化」の手順に従ってください。

13.3.3 OC4J での SSL の有効化

OC4J クライアントへの SSL 接続を構成するには、『Oracle Containers for J2EE セキュリティ・ガイド』のクライアント接続に対する Oracle HTTPS に関する項の手順に従ってください。

13.3.3.1 Oracle HTTP Server から OC4J への SSL の構成

SSL を介した AJP 通信を構成するには、`mod_oc4j` と `iaspt` デーモンの通信を構成する必要があります。この通信を構成するには、『Oracle HTTP Server 管理者ガイド』の「mod_oc4j と OC4J 間での SSL の有効化」の手順に従ってください。

13.3.3.2 Oracle HTTP Server から OC4J へのポート・トンネリング（iaspt）の使用

SSL 用にこの接続経路を構成するには、『Oracle HTTP Server 管理者ガイド』の「ポート・トンネリングの概要」の手順に従ってください。

13.3.3.3 ORMI/HTTP SSL の構成

ORMI over SSL はサポートされていません。同種の機能を構成するには、最初に ORMI over HTTP を構成してから、HTTP over SSL を構成します。

ORMI または HTTP の構成手順は、『Oracle Containers for J2EE サービス・ガイド』の「HTTP を介した ORMI トンネリングの構成」を参照してください。

13.3.3.4 Oracle Internet Directory による Oracle Application Server Java Authentication and Authorization Service (JAAS) Provider への SSL の構成

プロバイダを構成するには、『Oracle Application Server エンタープライズ・デプロイメント・ガイド』の「アプリケーションの認証と認可の構成」の手順に従ってください。プロバイダに SSL を構成するには、`SSL_ONLY_FLAG` を `true` に設定します。

13.3.3.5 Oracle HTTP Server への SSL の構成

『Oracle Containers for J2EE セキュリティ・ガイド』の「OC4J での SSL の有効化」では、Oracle HTTP Server に SSL を構成する方法について説明しています。

13.3.3.6 スタンドアロン OC4J インストールでの SSL の構成

『Oracle Containers for J2EE セキュリティ・ガイド』の「OC4J での SSL の有効化」では、SSL を使用してクライアントと OC4J インスタンス間の通信を保護する方法について説明しています。

13.3.4 J2EE and Web Cache インストールでの SSL の有効化

セキュリティの要求や Oracle Application Server J2EE and Web Cache インストールの構成に応じて、インストールされているコンポーネントの 1 つ以上にセキュアな通信を実装できます。最初のリスナー（OracleAS Web Cache または Oracle HTTP Server）を構成するだけで十分な場合もあります。

Oracle HTTP Server に SSL を構成するには、『Oracle HTTP Server 管理者ガイド』の「Oracle HTTP Server での SSL の有効化」の手順に従ってください。

OracleAS Web Cache に SSL を構成するには、『Oracle Application Server Web Cache 管理者ガイド』の「HTTPS リクエストをサポートするための OracleAS Web Cache の構成」の手順に従ってください。

13.3.5 Oracle Content DB での SSL の有効化

『Oracle Content Database Oracle WebCenter Suite 用管理者ガイド』の Oracle Content DB の SSL 構成に関する項では、Oracle Content DB で SSL を有効にする方法について説明しています。

13.3.6 仮想ホストでの SSL の有効化

仮想ホストを使用して、1 つの Oracle HTTP Server に複数の Web サイトをデプロイできます（たとえば、HTTP プロトコルおよび HTTPS プロトコルでアプリケーションを使用可能にできます）。

『Oracle Application Server Single Sign-On 管理者ガイド』の「仮想ホストでの mod_osso の構成」では、mod_osso によって保護される SSL 仮想ホストの構成手順が記載されています。名前ベースの仮想ホスティングは使用できません。IP ベースまたはポートベースの仮想ホスティングを使用する必要があります。

説明のシナリオでは、次の条件が満たされていることを想定しています。

- アプリケーション中間層のホスト名が app.mydomain.com（この名前をアプリケーション中間層のホスト名に置き換える）となっていること。
- 中間層が非 SSL パートナ・アプリケーションとしてすでに構成されていること（通常はインストール時に構成）。
- アプリケーション中間層のデフォルトの SSL ポート番号が 4443 であること。

13.3.7 Oracle Enterprise Manager 10*g* に対する SSL の構成

第 12.3.7 項「Oracle Enterprise Manager 10g に対する SSL の構成」を参照してください。

14

SSL のトラブルシューティング

この章では、SSL に関する一般的な疑問点とエラーについて説明します。

この章の項目は次のとおりです。

14.1 名前ベースの仮想ホスティングと SSL

SSL に名前ベースの仮想ホスティングを使用することはできません。これは SSL の制限です。

SSL を使用して複数の仮想ホストを構成する必要がある場合は、次のような回避策があります。

- IP ベースの仮想ホスティングを使用します。これを行うには、コンピュータに複数の IP アドレスを構成し、各 IP アドレスに異なる仮想名をマップします。
- 非標準ポート番号の使用が可能な場合は、同じ IP に異なる名前を関連付けることができますが、それぞれの名前は異なるポート番号を使用して構成する必要があります（例：*name1*：443、*name2*：553）。これによって、同じ IP を使用できるようになります。ただし、非標準のポート番号を使用する必要があります。標準の 443 ポートを使用できるのは 1 つの名前のみです。他の名前は別のポート番号を使用する必要があります。

14.2 SSL に関する一般的な ORA エラー

エラーの原因を判断するには、Oracle Net トレースを有効にすることが必要になる場合があります。Oracle Net トレース・パラメータの設定の詳細は、『Oracle Database Net Services 管理者ガイド』を参照してください。

ORA-28759: ファイルのオープンに失敗しました

原因： 指定されたファイルを開くことができませんでした。通常、このエラーは Oracle Wallet が見つからないために発生します。

処置： 次の項目をチェックします。

- Oracle Wallet が、デフォルトの場所（`ORACLE_HOME/Apache/Apache/conf/ssl.wlt/default`）か、`ORACLE_HOME/Apache/Apache/conf/ssl.conf` ファイルの `SSLWallet` ディレクティブで指定された場所にあることを確認します。このディレクトリは、Wallet を保存した場所と同じになります。
- Oracle Net トレースを有効にして、開くことのできないファイルの名前とその原因を確認します。
- Oracle Wallet を保存したときに自動ログインが有効になっていたことを確認します。詳細は、第 11.1.4.14 項「自動ログインの使用」を参照してください。

ORA-28786: 暗号化された秘密鍵の復号化に失敗しました

原因： 暗号化された秘密鍵の復号化に正しくないパスワードが使用されました。多くの場合、Wallet の自動ログインが使用されていないために発生します。

処置： Oracle Wallet Manager を使用して、Wallet の自動ログイン機能を有効にします。次に、Wallet を再保存します。詳細は、第 11.1.4.14 項「自動ログインの使用」を参照してください。

ORA-28858: SSL プロトコル・エラーが発生しました

原因： これは、2 つのプロセス間の SSL ハンドシェイク・ネゴシエーション中に発生する一般的なエラーです。

処置： Oracle Net トレースを有効にして接続を再試行し、トレース出力を生成します。次に、トレース出力を手元に用意して、Oracle サポート・サービスに連絡してください。

ORA-28859: SSL でネゴシエーションに失敗しました

原因： SSL プロトコルの一部である 2 つのプロセス間のネゴシエーション中にエラーが発生しました。このエラーは、接続の両端のサーバーとクライアントで共通の暗号スイートがサポートされていないときに発生する場合があります。

処置： Oracle HTTP Server とクライアント（ブラウザ）で構成されている暗号スイートが、クライアントとサーバーの両方に対して互換性があるかどうかを確認します。

Oracle HTTP Server で構成されている暗号スイートを確認するには、`ORACLE_HOME/Apache/Apache/conf/ssl.conf` ファイル内の `SSLCipherSuite` ディレクティブをチェックします。

ブラウザで構成されている暗号スイートをチェックするには、ブラウザのドキュメントを参照してください。ブラウザにはそれぞれ独自の暗号スイートの設定方法があります。

また、クライアントとサーバーの SSL のバージョンが同じであるか、互換性があることを確認してください。たとえば、サーバーが SSL 3.0 のみに対応し、クライアントが TLS 1.0 のみに対応している場合、SSL 接続は失敗します。

ORA-28862: SSL 接続に失敗しました

原因 : このエラーは、ピアが接続をクローズしたために発生しました。

処置 : 次の項目をチェックします。

- Oracle Wallet が、デフォルトの場所（`ORACLE_HOME/Apache/Apache/conf/ssl.wlt/default`）か、`ORACLE_HOME/Apache/Apache/conf/ssl.conf` ファイルの `SSLWallet` ディレクティブで指定された場所にあることを確認します。このディレクトリは、Wallet を保存した場所と同じになります。
- 暗号スイートが、クライアントとサーバーの両方に対して互換性があるかどうかをチェックします。暗号スイートのチェック方法の詳細は、「ORA-28859: SSL でネゴシエーションに失敗しました」を参照してください。
- 暗号スイートの名前のスペルが正しいことを確認します。
- クライアントとサーバーの SSL のバージョンが同じであるか、または互換性があることを確認してください。このエラーは、サーバーとクライアントで指定されている SSL のバージョンが一致していないために発生することがあります。たとえば、サーバーが SSL 3.0 のみに対応し、クライアントが TLS 1.0 のみに対応している場合、SSL 接続は失敗します。
- 詳細な診断情報を参照するには、ピアで Oracle Net トレースを有効にします。

ORA-28865: SSL 接続はクローズしました

原因 : 基盤となるトランスポート層でエラーが発生したか、ピア・プロセスが予期せず終了したために、SSL 接続がクローズされました。

処置 : 次の項目をチェックします。

- クライアントとサーバーの SSL のバージョンが同じであるか、または互換性があることを確認してください。このエラーは、サーバーとクライアントで指定されている SSL のバージョンが一致していないために発生することがあります。たとえば、サーバーが SSL 3.0 のみに対応し、クライアントが TLS 1.0 のみに対応している場合、SSL 接続は失敗します。
- Diffie-Hellman 匿名暗号スイートを使用していて、`ssl.conf` ファイル内の `SSLVerifyClient` ディレクティブが `require` に設定されている場合、クライアントは証明書をサーバーに渡しません。サーバーがクライアントの証明書を受信しないと、サーバーはクライアントを認証できないため、接続がクローズします。この問題を解決するには、別の暗号スイートを使用するか、`SSLVerifyClient` ディレクティブを `none` または `optional` に設定します。

  暗号スイートのチェック方法の詳細は、「ORA-28859: SSL でネゴシエーションに失敗しました」を参照してください。
- Oracle Net トレースを有効にして、ネットワーク・エラーのトレース出力をチェックします。

ORA-28868: ピア証明連鎖のチェックに失敗しました

原因 : ピアが提示した証明連鎖が確認されましたが、その確認に失敗しました。この失敗は、次のようないくつかの問題が原因になっている可能性があります。

- 連鎖内の証明書のいずれかが期限切れになっています。
- 連鎖内の証明書のいずれかに対する認証局がトラスト・ポイントとして認識されていません。
- いずれかの証明書の署名が検証できません。

処置 : 第 11.1.4.3 項「既存の Wallet を開く」の手順に従って、Oracle Wallet Manager を使用して Wallet を開き、次の点を確認します。

- Wallet にインストールされている証明書がすべて現行のものである（期限切れになっていない）ことを確認します。
- ピアの証明連鎖からの証明局の証明書が、Wallet 内で信頼できる証明書として追加されていることを確認します。Oracle Wallet Manager を使用して信頼できる証明書をインポートするには、第 11.1.5.2.1 項「信頼できる証明書のインポート」を参照してください。

ORA-28885: 必須の鍵使用方法のある証明書が見つかりません。

原因 : 証明書が、X.509 バージョン 3 の適切な鍵使用目的拡張属性を使用して作成されていません。

処置 : Oracle Wallet Manager を使用して、証明書の鍵の使用方法を確認します。詳細は、表 11-4「X.509 バージョン 3 の KeyUsage 拡張タイプ、値および説明」を参照してください。

ORA-29024: 証明書の妥当性チェックに失敗しました

原因 : 相手から送信された証明書の妥当性を確認できませんでした。これは、証明書が、期限切れか、取り消されているか、他の理由によって無効になっているときに発生する場合があります。

処置 : 次の項目をチェックします。

- 証明書が有効であるかどうかを確認します。必要に応じて、新しい証明書を取得するか、送信者に証明書の検証に失敗したことを伝えるか、再送信します。
- サーバーの Wallet に、クライアントの証明書の妥当性をチェックするトラスト・ポイントがあることを確認します。トラスト・ポイントがない場合には、Oracle Wallet Manager を使用して、適切なトラスト・ポイントを Wallet にインポートします。詳細は、第 11.1.5.2.1 項「信頼できる証明書のインポート」を参照してください。
- 証明書が取り消されておらず、証明書失効リスト（CRL）のチェックが有効になっていることを確認します。詳細は、第 11.2.5 項「orapki ユーティリティによる証明書失効リスト（CRL）の管理」を参照してください。

ORA-29223: 証明連鎖を作成できませんでした

原因 : インストールされている証明書の既存のトラスト・ポイントを使用して、証明連鎖を作成できません。通常、このエラーが返されるのは、ピアから完全な連鎖が提供されず、証明連鎖を完成するための適切なトラスト・ポイントがない場合です。

処置 : Oracle Wallet Manager を使用して、連鎖を完成するのに必要なトラスト・ポイントをインストールします。詳細は、第 11.1.5.2.1 項「信頼できる証明書のインポート」を参照してください。

第 V 部

バックアップとリカバリ

バックアップとリカバリとは、ハードウェア障害およびデータの損失に備え、損失が発生した場合にデータを再構成するための様々な計画および手順を意味します。この部では、Oracle Application Server のバックアップおよびリカバリ方法について説明します。

この部は、次の章で構成されています。

15

バックアップとリカバリの概要

この章では、Oracle Application Server のバックアップおよびリカバリの概要について説明します。

この章の項目は次のとおりです。

15.1 Oracle Application Server のバックアップとリカバリの考え方

この項では、Oracle Application Server 環境のバックアップおよびリカバリに対する考え方について説明します。Oracle Application Server 環境には、様々なコンポーネントや構成を含めることができます。どのコンポーネントと構成が要件に最も適しているかを判断するには、Oracle Application Server のインストレーション・ガイドおよび『Oracle Application Server 概要』を参照してください。

通常の Oracle Application Server 環境には、1 つ以上の中間層インストールが含まれます。

Oracle Application Server 環境内のインストールでは、構成情報、アプリケーションおよびデータの同期が保たれ、相互に依存します。たとえば、構成変更を実行すると、中間層インストールで構成ファイルの更新が必要な場合があります。あるアプリケーションをデプロイすると、すべての中間層インストールへのデプロイが必要な場合があります。

したがって、バックアップおよびリカバリを実行するときは、Oracle Application Server 全体の環境を考慮することが重要になります。Oracle Application Server 環境全体を一度にバックアップしてください。そうすれば、ファイルやデータなどが失われた場合でも、環境全体を一貫性のある状態にリストアできます。

バックアップおよびリカバリのためのファイル・タイプには、次のものがあります。

- **Oracle ソフトウェア・ファイル**

  バイナリやライブラリなどの静的なファイルです。これらは、中間層の Oracle ホームに格納されます。これらは、インストール時に作成されます。

- **構成ファイル**

  これらのファイルには、構成情報およびデプロイされたアプリケーションが含まれます。これらは、中間層の Oracle ホームに格納されます。これらは、インストール時または実行時に作成され、アプリケーション・サーバーの通常の操作中に更新されます。構成ファイルのタイプには、Oracle HTTP Server、OC4J および OPMN があります。

- **Oracle システム・ファイル**

  これらのファイルは、`/var/opt/oracle` または `/etc` ディレクトリ、および `oraInventory` ディレクトリ内に格納されています。これらは、Oracle Application Server 環境内の各ホスト上に存在します。これらは、通常 Oracle Application Server インストールの外部に格納されますが、`oraInventory` ディレクトリは、Oracle ホーム内に存在する場合があります。これらのファイルは、Oracle Universal Installer によってインストール時に作成または更新されるもので、インストールに関する情報を含んでいます。Windows では、インストーラによりレジストリの一部が作成されます。

このマニュアルで説明する計画および手順では、Oracle Application Server 環境の一貫性を維持する方法で、これらの異なるタイプのファイルをバックアップおよびリカバリします。

注意： Oracle Application Server 環境には、ログ・ファイルなど、この項に記載されている追加のファイルがあります。さらに、静的な HTML ファイルや CGI スクリプトなど、Oracle ホームにデプロイされるファイルも含まれます。これらのファイルはいずれもバックアップ・リストに追加できます。

15.2 バックアップ計画の概要

この項では、このマニュアルで使用するバックアップ計画について説明します。この項の項目は次のとおりです。

15.2.1 バックアップのタイプ

Oracle Application Server のバックアップ計画には、次の 2 つのタイプがあります。

- イメージのバックアップ
- インスタンスのバックアップ

イメージのバックアップ

Oracle Application Server インスタンスのイメージのバックアップには、そのインスタンスの Oracle ホーム・ディレクトリ、OraInventory ディレクトリ、oratab ファイル、そのノードの Windows レジストリ、およびその Oracle Application Server インスタンスのコールド・バックアップが含まれます。Oracle ホーム・ディレクトリには、Oracle Application Server インスタンスのすべてのバイナリ・ファイル、実行可能ファイル、初期化ファイル、構成ファイル、ログ・ファイルなどや、そのインスタンスのすべてのコンポーネントおよびデプロイされたアプリケーションが含まれます。OraInventory ディレクトリには、インスタンスのインストール情報があります。

インスタンスのバックアップ

Oracle Application Server コンポーネントおよびデプロイされたアプリケーションの構成情報が含まれます。OracleAS Recovery Manager では、バックアップ入力ファイルでそれぞれの構成された中間層コンポーネントに対して指定された構成ファイルのすべてのローカル・コピーがバックアップされます。

15.2.2 Oracle Application Server コンポーネント・バックアップ入力ファイル

Oracle Application Server コンポーネントごとにバックアップ入力ファイルがあり、そのコンポーネント用にバックアップが必要なすべての構成ファイルのリストが含まれています。コンポーネントがインストールされ構成されていると、バックアップ操作時に OracleAS Recovery Manager ではコンポーネントのバックアップ入力ファイルを呼び出してバックアップするファイルを判断します。コンポーネント・バックアップ入力ファイルのファイル拡張子は `.inp` であり、*`Oracle_Home`*`/backup_restore/config` ディレクトリに格納されます。表 15-1 は、このディレクトリに存在する場合があるコンポーネント・バックアップ入力ファイルを示します。

表 15-1 Oracle Application Server コンポーネント・バックアップ入力ファイル

コンポーネント名	バックアップ入力ファイル
Oracle Enterprise Manager	config_em_files.inp
バックアップ中に除外されるファイルのリスト	config_exclude_files.inp
Oracle Application Server インストール情報	config_install_files.inp
Oracle Enterprise Manager ログ・ローダー	config_logloader_files.inp
その他のバックアップ対象ファイル	config_misc_files.inp
Oracle Containers for J2EE アプリケーション	config_oc4j_files.inp
Oracle HTTP Server	config_ohs_files.inp
Oracle iASPT	config_iaspt_files.inp
Oracle Java Object Cache	config_javaobjcache_files.inp
Oracle Portal	config_portal_files.inp
Oracle Process Manager and Notification Server	config_opmn_files.inp
Oracle WebCenter Framework メタデータ・ストア	config_misc_files.inp

15.2.3 プラグイン・バックアップ入力ファイル

プラグイン・バックアップ入力ファイルを作成して、インストール後にファイルをバックアップに追加できます。各プラグイン・バックアップ入力ファイルは Oracle Application Server コンポーネントまたは Oracle アプリケーションに属し、バックアップする追加ファイルのリストを含んでいます。

プラグイン・バックアップ入力ファイルに指定されているファイルのリストは、ローカルの Oracle ホーム・ディレクトリに存在する必要があります。リストされているファイルは、backup_config コマンドの実行時にバックアップされます。これらのファイルのデータと、同じ Oracle ホーム内のその他すべてのコンポーネント構成ファイルのデータとは相互に依存している場合があるため、Recovery Manager によってすべてのファイルが、リストア操作のために 1 つの JAR アーカイブ・ファイルにまとめられます。プラグイン・ファイルの最初のファイルは、キー・ファイルである必要があり、Recovery Manager によってアクセスできる必要があります。最初のファイルを検索してバックアップできなかった場合は、バックアップ構成操作全体が終了し、エラー・メッセージがログに記録されます。

プラグイン・バックアップ入力ファイルのエントリ形式は次のとおりです。

バックアップに特定のファイルを指定するには、次のように指定します。

```
${ORACLE_HOME}/directorypath/filename
```

ディレクトリを指定するには、次のように指定します。

```
${ORACLE_HOME}/directorypath
```

ワイルドカードを使用するには、次のように指定します。

```
${ORACLE_HOME}/directorypath/*.conf
```

プラグイン・バックアップ入力ファイルのファイル・リストにある最初のエントリでは、ワイルドカードを使用することはできません。最初のファイルは、キー・ファイルである必要があり、Recovery Manager によってアクセスできる必要があります。

プラグイン・バックアップ入力ファイルの作成後、このファイルを *Oracle_Home*/backup_restore/plugin_config ディレクトリに追加します。プラグイン・バックアップ入力ファイルの名前は、次の形式で指定する必要があります。

```
config_component_name_plugin.inp
```

次に例を示します。

```
config_rules_plugin.inp
config_oc4j_plugin.inp
config_ohs1_plugin.inp
```

プラグイン・バックアップ入力ファイルの有効化

Recovery Manager がプラグイン・バックアップ入力ファイルに指定されているファイルをバックアップするには、プラグイン・バックアップ入力ファイルを有効にする必要があります。enable_component_inp コマンドを実行すると、入力ファイルが有効になります。次の例は、このコマンドの構文を示しています。

UNIX の場合 :

```
bkp_restore.sh [-d -s -v] -m enable_component_inp -y "component_name[, component_
name]…"
```

Windows の場合 :

```
bkp_restore.bat [-d -s -v] -m enable_component_inp -y "component_name[, component_
name]…"
```

コマンドと構文の詳細は、第 16.4.2 項を参照してください。

プラグイン入力ファイルを有効にしたら、リストア構成操作（restore_config）を実行する前に、新しいバックアップ構成操作（backup_config）を実行する必要があります。

15.2.4 推奨されるバックアップ計画

この項では、バックアップの実行に際し推奨される計画について説明します。この計画に従って、このマニュアルで説明するリカバリ手順を実行することができます。

- **完全なイメージのバックアップを実行します。**

  Oracle Application Server をインストールした直後に、Oracle Application Server 環境の各ノードで完全なイメージのバックアップを実行する必要があります。このバックアップには、各ノードを初期状態にリストアするために必要なものがすべて含まれます。このバックアップは、以降のすべての実行時バックアップに対するベースラインとして機能します。

- **インスタンスのバックアップを定期的に実行します。**

  管理上の変更を実行するたびに、または（これが不可能な場合は）定期的に、Oracle Application Server 環境のインスタンスのバックアップを実行してください。これにより、構成とアプリケーションを最後にバックアップした時点の一貫性のある状態に、環境をリストアすることができます。バックアップの矛盾を防ぐために、バックアップが完了するまで、いずれの Oracle Application Server インスタンスの構成も変更しないでください。

  > **関連項目：** 管理上の変更の詳細は、付録 E「管理上の変更の例」を参照してください。

- **大きな変更の後に、新たに完全なイメージのバックアップを実行します。**

  Oracle Application Server 環境に大きな変更を行う場合、新たに完全なイメージのバックアップを実行します。このバックアップは、以降のすべてのオンライン・バックアップに対するベースラインとして機能します。

  次の処理の後に、新たに完全なイメージのバックアップを実行します。

  - オペレーティング・システム・ソフトウェアのアップグレード
  - Oracle Application Server ソフトウェアのアップグレードまたはパッチ適用

  アップグレードやパッチを取り消す場合、最後の完全なイメージのバックアップまで戻ります。その後、ソフトウェア・アップグレードやパッチと Oracle Application Server 環境の最後の完全なイメージのバックアップとの間で発生した、任意のインスタンス・バックアップを適用できます。イメージの前回の完全バックアップをリストアせずにインスタンスのバックアップをリストアすると、新たにアップグレードされた互換性のないソフトウェアと、古い構成ファイルが混在するおそれがあります。

- **インスタンスのバックアップを定期的に実行します。**

  Oracle Application Server 環境の完全なイメージのバックアップを新たに作成した後、定期的なインスタンス・バックアップを続けて実行します。

- **ポートレットを広範囲にカスタマイズおよびパーソナライズできるように、ポートレット・プロデューサのバックアップを実行します。**

  そうすることにより、ポートレット・プロデューサで管理および格納されているカスタマイズ・データやパーソナライズ・データがバックアップされます。ポートレット・プロデューサのカスタマイズ・データやパーソナライズ・データをバックアップするユーティリティは、リモートのポートレット・プロデューサが実行されているノードで実行する必要があります。

15.3 リカバリ計画の概要

このマニュアルで使用する Oracle Application Server のリカバリ計画には、次の 2 つのタイプがあります。

- データ損失、ホスト障害またはメディア障害に対するリカバリ計画（クリティカル）
- プロセスのクラッシュまたはシステムの停止に対するリカバリ計画（非クリティカル）

データ損失、ホスト障害またはメディア障害に対するリカバリ計画（クリティカル）

これらの計画により、実データの損失などの重大な障害からのリカバリが可能になります。損失のタイプにもよりますが、次のファイル・タイプのどのような組合せでもリカバリできます。

- Oracle ソフトウェア・ファイル
- 構成ファイル
- Oracle システム・ファイル

すべてのケースで、これらの計画により、すべてのインストールにわたって一貫した状態が確保されます。

プロセスのクラッシュまたはシステムの停止に対するリカバリ計画（非クリティカル）

これらの計画により、停止または失敗したプロセスが再起動されます。データはリストアされません。このマニュアルでは、リカバリ計画の万全を期すために、これらについて説明しています。

15.4 OracleAS Recovery Manager とは

OracleAS Recovery Manager は、中間層の構成ファイルをバックアップおよびリカバリするために使用できるアプリケーションです。

OracleAS Recovery Manager は、Oracle Application Server をインストールするときにデフォルトでインストールされます。このアプリケーションは、*Oracle_Home*/backup_restore ディレクトリにインストールされます。OracleAS Recovery Manager を手動でインストールする方法は、第 16.2 項を参照してください。

15.5 前提と制限

次の前提と制限は、このマニュアルに記載されているバックアップおよびリカバリ手順に適用されます。

- OracleAS Recovery Manager には、以前のリリースの OracleAS Recovery Manager との間に下位互換性がありません。以前のリリースの OracleAS Recovery Manager を使用して作成されたアーカイブは、現在のリリースでリカバリすることはできません。
- サポートされているインストール・タイプは次のとおりです。
  - Oracle WebCenter Framework
  - Oracle HTTP Server のある Oracle WebCenter Framework
  - Oracle HTTP Server
- バックアップおよびリカバリ手順では、Oracle Content Database のインストール・タイプをサポートしていません。
- OracleAS Cold Failover Cluster または Disaster Recovery を使用している場合は、『Oracle Application Server 高可用性ガイド』の特別な考慮事項を参照してください。
- このリリースでは、Recovery Manager はコマンドラインからのみ実行できます。
- Windows では、リモート・ファイル・システムにバックアップを格納する場合、ローカルのマップされたドライブを作成して、バックアップ格納ディレクトリに指定する必要があります。たとえば、`Z:¥ASbackups` がバックアップ用にマップされたドライブである場合、構成ファイルおよびリポジトリのバックアップ・ディレクトリは `Z:¥ASbackups` になります。

15.6 バックアップとリカバリを初めて実行する場合の手引き

この項には、Oracle Application Server のバックアップおよびリカバリを初めて実行する際の手引きが記載されています。

1. **OracleAS Recovery Manager を構成します。**

   OracleAS Recovery Manager を構成し、その機能についての知識を習得することをお薦めします。

2. **バックアップ計画をインプリメントします。**

   推奨されるバックアップ計画およびバックアップ手順の概要は、第 17 章「バックアップ計画と手順」を参照してください。このバックアップ計画に従って、このマニュアルに記述されているリカバリ手順を実行することができます。

3. **必要に応じてリカバリを実行します。**

   システム障害またはデータの損失が発生した場合は、第 18 章「リカバリ計画と手順」を参照してください。この章では、様々なタイプの障害について解説し、リカバリを実行する手順について説明します。

16

Oracle Application Server Recovery Manager

この章では、Oracle Application Server Recovery Manager のインストール、構成および使用方法について説明します。

この章の項目は次のとおりです。

16.1 OracleAS Recovery Manager の入手方法

OracleAS Recovery Manager は、Oracle Application Server の一部としてインストールされます。このアプリケーションは *Oracle_Home*/backup_restore ディレクトリに配置されます。この backup_restore ディレクトリに常駐する場合があるファイルを表 16-1 に示します。

表 16-1 OracleAS Recovery Manager のファイル

ファイル[1]	説明
bkp_restore.sh	UNIX でマネージャを実行するために使用するシェル・スクリプト。
bkp_restore.bat	Windows でマネージャを実行するために使用するバッチ・コマンド・ファイル。
config/config.inp	環境に応じて OracleAS Recovery Manager をカスタマイズするためのパラメータを含む主要な構成ファイル。orainst_loc_path フィールドは、–invPtrLoc のインストーラ・コマンドライン・オプションを指定してインスタンスをインストールした場合にのみ変更する必要があります。oraInst.loc の場所が標準以外である場合に、それを反映して変更する必要があります。
config/config_*component*_files.inp	コンポーネント構成ファイル。各ファイルには、特定のコンポーネントに対する構成ファイルのリストが含まれます。これにより、インスタンスのバックアップを実行する際、どのファイルをバックアップするかが決定されます。コンポーネント構成ファイルの一覧は、第 15.2.2 項「Oracle Application Server コンポーネント・バックアップ入力ファイル」を参照してください。

[1] パスは、OracleAS Recovery Manager ディレクトリのルートに対する相対パスです。

関連項目： Oracle Application Server のインストールの詳細は、Oracle Application Server のインストレーション・ガイドを参照してください。

16.2 OracleAS Recovery Manager の手動による構成方法

この項では、OracleAS Recovery Manager を手動で構成する方法について説明します。環境内の各インストールに、これらの手順を実行する必要があります。

Windows ユーザーの皆様へ： OracleAS Recovery Manager ディレクトリ内のファイルを編集するときは、ワードパッドなどのリッチ・テキスト・エディタは使用しないでください。行末ごとに改行記号が挿入されるため、マネージャにエラーが発生するおそれがあります。メモ帳などの基本的なテキスト・エディタを使用することをお薦めします。

1. OracleAS Recovery Manager を実行する前に、ORACLE_HOME 環境変数を設定します。
2. OracleAS Recovery Manager により、ログ・ファイルとバックアップ・ファイルが書き出されます。これらを格納する次のディレクトリを指定する必要があります。デフォルトのログ・ファイル・ディレクトリは、*ORACLE_HOME*/backup_restore/logs です。config.inp を編集して次のディレクトリを作成します。
   - **ログ・ファイル・ディレクトリ**：このディレクトリは、マネージャによって作成されたログ・ファイルを格納します。このディレクトリには、数メガバイトの容量が必要です。
   - **構成ファイルのバックアップ・ディレクトリ**：このディレクトリは、構成ファイルのバックアップを格納します。このディレクトリには、数百メガバイトの容量が必要です。

これらのディレクトリを作成する際の推奨事項は、次のとおりです。

- バックアップ・ディレクトリは、Oracle Application Server の Oracle ホームとは別のディスク（そして、可能であれば別のディスク・コントローラ）上のファイル・システムに作成します。これにより、ハードウェア障害が発生したときに、データをリカバリできる可能性が高くなります。
- バックアップ・ディレクトリは、Oracle Application Server をインストールしたユーザーが書込みできるようにします。

  たとえば、ログ・ファイル・ディレクトリおよび構成ファイルのバックアップ・ディレクトリを /disk1 に作成する手順は次のとおりです。

  UNIX の場合 :

```
mkdir -p /disk1/backups/log_files
mkdir -p /disk1/backups/config_files
cd /disk1/backups
chmod 755 log_files config_files
chown OracleAS_user log_files config_files
```

  Windows の場合 :

```
mkdir C:¥backups¥log_files
mkdir C:¥backups¥config_files
```

3. config.inp を編集し、表 16-2 に示すようにパラメータを変更します。

表 16-2 config.inp のパラメータ

パラメータ	値
oracle_home（オプション）	これに値を挿入しないでください。コマンドライン・インタフェースを使用して、まずシェル環境で ORACLE_HOME を設定します。
log_path（オプション）	ログ・ファイル・ディレクトリのフルパスを指定します。フルパスを指定しないと、-m configure コマンドを実行したときに、デフォルトのログ・ディレクトリ *ORACLE_HOME*/backup_restore/logs が自動的に作成されます。config.inp ファイルで log_path が指定されていて、指定したディレクトリが存在しない場合、OracleAS Recovery Manager では、-m configure コマンドで -f（force）オプションが使用されているかどうかにかかわらず、指定したログ・ディレクトリが自動的に作成されます。ただし、構成ファイルのバックアップ・ディレクトリは、-f オプションが指定されていないかぎり、自動的に作成されません。
config_files_list	これに値を挿入しないでください。config_files_list=DO_NOT_SET のままにします。 このパラメータは、bkp_restore.pl -m configure の実行時に、インストールに対応する適切な構成ファイルのリストで更新されます。
config_backup_path（必須）	構成ファイルのバックアップ・ディレクトリのフルパスを指定します。
install_type	これに値を挿入しないでください。install_type=DO_NOT_SET のままにします。 このパラメータは、bkp_restore.pl -m configure の実行時に、インストールに対応する適切な値で更新されます。
orainst_loc_path（オプション）	このパラメータは UNIX プラットフォームでのみ使用します。インストール中にデフォルトのパスが上書きされた場合は、oraInst.loc ファイルが存在するディレクトリのフルパスを指定します。それ以外の場合は、デフォルト値のままにします。
plugin_config_files_list=DO_NOT_SET	このパラメータは更新しないでください。このパラメータは、プラグイン・バックアップ入力ファイルを含むコンポーネントを有効にしたときに更新されます。

-m configure オプションで OracleAS Recovery Manager を実行し、このアプリケーションを構成します。たとえば、次のコマンドを使用します。

- UNIX の場合 :

```
./bkp_restore.sh -m configure
```

- Windows の場合 :

```
bkp_restore.bat -m configure
```

これで OracleAS Recovery Manager を使用する準備ができました。

16.3 構成ファイルに対応した OracleAS Recovery Manager のカスタマイズ

デフォルトでは、OracleAS Recovery Manager により、Oracle Application Server インストールの再構成に必要なすべての Oracle Application Server 構成ファイルをバックアップします。OracleAS Recovery Manager をカスタマイズすると、定期的にバックアップする必要のあるファイルを追加したり、バックアップする必要のないファイルを除外できます。

16.3.1 構成ファイルのバックアップ時の OracleAS Recovery Manager の動作

OracleAS Recovery Manager をカスタマイズする前に、OracleAS Recovery Manager の動作について理解している必要があります。マネージャを使用して構成ファイルをバックアップするとき、次のことが実行されます。

1. -e オプションで別の環境ファイルが指定されていない場合、config.inp を開き、config_files_list を取得します。
2. config_files_list 内の各入力ファイルを開くよう試行し、すべてのファイルを開けない場合はエラーで終了します。
3. config_exclude_files.inp の内容を確認します。マネージャは、このファイルにリストされているファイルのバックアップは実行しません。
4. config_files_list 内の各ファイルを調べ、各ファイルの最初のエントリを確認します。このエントリは、キー・ファイルです。キー・ファイルは、このインストールにコンポーネントが存在するかどうかを判断するために使用されます。
   - キー・ファイルが検出されると、コンポーネントがインストールされていることが確認され、ファイル内にあるすべてのエントリのバックアップが試行されます。キー・ファイルが検出されない場合は、ログにエラーが記録されます。その他のファイルが検出されない場合は、警告が出力され、バックアップが続行されます。
   - キー・ファイルが存在しない場合は、コンポーネント入力ファイルのどのエントリのバックアップも試行されません。ログ・ファイルにエラーが記録され、次のコンポーネント入力ファイルに進みます。
5. 構成ファイルは、config.inp ファイルの config_backup_path パラメータで指定されたディレクトリにある jar ファイルに格納されます。

   ```
   config_bkp_2006-05-10_18-33-10.jar
   ```

16.3.2 OracleAS Recovery Manager のカスタマイズ方法

OracleAS Recovery Manager では、インストールにどの構成ファイルが存在するかを判断できるため、OracleAS Recovery Manager をカスタマイズする必要はありません。ただし、次のような場合にマネージャのカスタマイズが必要になることもあります。

- バックアップへのファイルの追加

  定期的にバックアップする必要のある、独自のローカル構成ファイルまたは他の任意のファイル（ログ・ファイルなど）を追加する場合です。

- バックアップからのファイルの除外

  バックアップからファイルを除外する場合です。

バックアップへのファイルの追加

Oracle Application Server コンポーネント固有のログ・ファイルなどのファイルをバックアップに追加するには、次のように config_misc_files.inp ファイルにエントリを追加します。

- 特定のファイルを指定するには、次のように指定します。

  ${ORACLE_HOME}/*directorypath*/*file*

- ディレクトリ全体を指定するには、次のように指定します。

  ${ORACLE_HOME}/*directorypath*/

- ワイルドカードを使用するには、次のように指定します。

  ${ORACLE_HOME}/*directorypath*/*.html

エントリはいくつでも追加できます。config_misc_files.inp ファイルは、常に config.inp 内の config_files_list パラメータに含まれます。したがって、config.inp を編集する必要はありません。

場合によっては、OracleAS Recovery Manager で、通常のディレクトリ構造の外部に格納されている追加のファイルが認識されないことがあります。たとえば、次の場合は、config_misc_files.inp を編集して、次の追加のファイルが適切にバックアップされるようにする必要があります。

- Oracle HTTP Server 構成ファイル（httpd.conf や moddav.conf など）に定義されている仮想パスまたはデフォルト以外のパス。他のファイルまたはディレクトリをポイントするようにこれらの Web サーバー構成ファイルを変更した場合は、新しいパスを実行時バックアップに含めることを検討してください。
- OC4J コンテナにデプロイされていて、コンテナ・ディレクトリの外部にあるファイルを使用するアプリケーション。OracleAS Recovery Manager では、コンテナ・ディレクトリ内のすべてのファイルが自動的にバックアップされます。アプリケーションでその他のディレクトリを使用する場合は、それらを構成バックアップの一部として処理することを検討してください。
- ファイルベースの永続性のある Java Message Service（JMS）。JMS ランタイム・データ（メッセージ）は、物理的なファイルに格納されるため、バックアップ・プロセスの一部として処理してください。

config_misc_files.inp に、キー・ファイルを指定する必要はありません。

バックアップからのファイルの除外

次のいずれかの方法を使用して、バックアップからファイルを除外することができます。

- 該当する config_*component*.inp ファイルからファイル・エントリを削除します。
- config_*component*.inp ファイルでディレクトリ全体をバックアップするよう指定してある場合、そのディレクトリ内の特定ファイルを除外するには、そのファイルのエントリを config_exclude_files.inp に追加します。これにより、指定されたファイルを除くディレクトリ全体がバックアップされます。config_exclude_files.inp 内でディレクトリを指定、またはワイルドカードを使用することはできません。単一ファイルのエントリのみを使用できます。

  config_exclude_files.inp に、キー・ファイルを指定する必要はありません。

16.4 OracleAS Recovery Manager の使用方法のまとめ

この項では、OracleAS Recovery Manager の使用方法についてまとめます。

この項の項目は次のとおりです。

16.4.1 OracleAS Recovery Manager を実行する際の前提条件

OracleAS Recovery Manager を実行する前に、次の手順を実行します。

- Oracle Application Server をインストールしたユーザーとしてログインします。
- ORACLE_HOME 環境変数が設定されていることを確認します。

16.4.2 構文

OracleAS Recovery Manager の構文は次のとおりです。

UNIX の場合 :

```
bkp_restore.sh [-defsv] -m mode [args]
```

Windows の場合 :

```
bkp_restore.bat [-defsv] -m mode [args]
```

次のオプションが有効です。

-d 実行せずにトレースを出力する。
-e 環境ファイルを指定する（デフォルトは config.inp）。
-f 現行のコマンドで、ログ・ファイルおよび構成ファイルのディレクトリが必要とされているのにそれらのディレクトリが存在しない場合、強制的に作成する。
-n プロンプトを非表示にして、マネージャをバッチ・モードで実行できるようにする。
-o Loss of Host Automation（LOHA）操作。
-s サイレント・モードで実行する。
-v 冗長モードで実行する。
-y プラグイン・バックアップ入力ファイルに関連付けられているコンポーネントを有効にする。

実行モードを指定するには、-m オプションを使用します。一部のモードでは、引数が取得されます。表 16-3 に、OracleAS Recovery Manager のモードとその引数を示します。すべてのモードと引数では大文字と小文字が区別されます。

bkp_restore 操作の間隔は、最低 1 分間空けて行う必要があります。バックアップの jar が存在するとき、そのタイムスタンプと現在実行中のバックアップ操作の実行時刻との間隔が 1 分未満である場合は、現在のバックアップ操作が失敗します。

表 16-3 OracleAS Recovery Manager のモードと引数

モードと引数	説明
backup_config	構成の全体バックアップを実行します。このコマンドを実行すると、次の操作が行われます。 ■ config.inp（または、-e オプションで指定されている代替のファイル）を開き、config_files_list、config_backup_path および log_path を取得します。 ■ config_files_list 内の各ファイルを開くよう試行します。すべてのファイルが開けない場合は、エラーで終了します。 ■ config_files_list 内の各ファイルに対して、最初のエントリ（キー・ファイル）が存在するかどうかをチェックします。キー・ファイルが存在しない場合は、致命的エラーとして処理されます。存在する場合は、リスト内のすべてのファイルをバックアップします。存在しないファイルが他にある場合は、ログにエラーを記録し、続行します。 ■ config_exclude_files.inp 内にリストされているファイルを除外します。 ■ 完了したら、config_backup_path/config_bkp_*timestamp* にバックアップを格納します。 ■ エラーが発生した場合は、*log_path*/config_bkp_*timestamp* にログ・ファイルを作成します。
backup_config_incr	構成ファイルの増分バックアップを実行します。 backup_config と同様の動作ですが、これは構成ファイルの最後の全体バックアップまたは増分バックアップ以降に変更されたすべての構成ファイルをバックアップします。
backup_instance_cold	Oracle Application Server インスタンスの完全なコールド・バックアップを実行します。このコマンドを実行すると、次の操作が行われます。 ■ OPMN が管理するすべてのプロセスを停止します。 ■ OPMN 管理プロセスを起動します。 ■ OPMN が管理するプロセスをすべてチェックして、プロセスが停止していることを確認します。そうでない場合は、もう一度停止します。それでもプロセスを停止できない場合は、致命的エラーを発行します。 ■ 構成の全体バックアップを実行します。 ■ OPMN が管理するすべてのプロセスを起動します。 ■ すべての OPMN プロセスをチェックして、実行中であることを確認します。そうでない場合は、警告メッセージを発行します。
backup_instance_cold_incr	Oracle Application Server インスタンスの増分コールド・バックアップを実行します。このコマンドを実行すると、次の操作が行われます。 ■ OPMN が管理するすべてのプロセスを停止します。 ■ OPMN 管理プロセスを起動します。 ■ OPMN が管理するプロセスをすべてチェックして、プロセスが停止していることを確認します。そうでない場合は、もう一度停止します。それでもプロセスを停止できない場合は、致命的エラーを発行します。 ■ 構成の増分バックアップを実行します。 ■ OPMN が管理するすべてのプロセスを起動します。 ■ すべての OPMN プロセスをチェックして、実行中であることを確認します。そうでない場合は、警告メッセージを発行します。
backup_instance_online	Oracle Application Server インスタンスのオンライン・バックアップを実行します。
backup_instance_online_incr -1 *level number*	Oracle Application Server インスタンスの増分オンライン・バックアップを実行します。
configure	マネージャを構成します。このコマンドを実行すると、config.inp 内の config_files_list および install_type が、インストールに対する適切な情報で更新されます。
help	使用方法に関するメッセージを出力します。

表 16-3 OracleAS Recovery Manager のモードと引数（続き）

モードと引数	説明
`list_backups`	該当するインスタンスの構成バックアップをリストします。
`list_instance_backups`	該当するインスタンスのインスタンス・レベルのバックアップをリストします。
`list_changed_config`	最後の全体バックアップまたは増分バックアップ以降に変更されたすべての構成ファイルをリストします。このコマンドでは、各ファイルの変更日付はチェックされますが、ファイルの実際の内容はチェックされません。ファイルのリストをログ・ファイルに書き込み、ログ・ファイルの名前を出力します。削除されたファイルまたは削除されたディレクトリは、`list_changed_config` に一覧表示されません。一覧表示されるのは、変更されたファイル、または変更されたファイルを含むディレクトリのみです。
`node_backup -o image_backup -P` *`directory for the image archive`*	元のホストのイメージ・アーカイブを作成します。このイメージには、インストールに応じて、元の Oracle ホーム、oratab、セントラル・インベントリなどが含まれます。UNIX では、`root` としてこのコマンドを実行する必要があります。
`node_backup -o prepare`	ノードに対してバックアップの準備を行います。この準備操作を実行すると、オペレーティング・システムの種類、ホスト名 /IP アドレス、ユーザー / グループ ID、インストール・タイプ、セントラル・インベントリの場所、Oracle ホームの場所（複数ある場合）などが検出されます。また、Windows レジストリと Windows サービス・データベースがスキャンされ、Oracle ホーム用に作成されたすべてのサービスが検出されます。この情報はファイルに保存され、ノードのリストア時に使用されます。 このモードでは、構成のバックアップも作成されます。
`node_restore -o inst_reconfigure -t config_bkp_timestamp`	新しいホストでインスタンスを再構成します。再構成では、インストール・タイプに応じて、IP の変更、構成バックアップ、リストアなどが実行されます。
`node_restore -o inst_register`	oratab およびセントラル・インベントリにインスタンスを登録します。また、`root.sh` の実行によって、デーモンの起動および停止スクリプトが設定され、Windows の場合は Windows サービスが作成されます。 UNIX システムでは、`root` としてこのコマンドを実行する必要があります。
`node_restore -o sys_init`	oratab（UNIX）、Windows レジストリ（Windows）、セントラル・インベントリなどの、Oracle Universal Installer 関連のメタデータをリストアします。このコマンドは、新しいホストで 1 回のみ実行します。 UNIX システムでは、`root` としてこのコマンドを実行する必要があります。

表 16-3 OracleAS Recovery Manager のモードと引数（続き）

モードと引数	説明
restore_config [-t config_bkp_*timestamp*] [-n]	構成ファイルをリストアします。このコマンドを実行すると、次の操作が行われます。 ■ config.inp（または、-e オプションで指定されている代替のファイル）を開き、config_backup_path および log_path を取得します。 ■ -t オプションが指定され、それが全体バックアップからのタイムスタンプである場合は、該当する全体バックアップをリストアします。 ■ -t オプションが指定され、それが増分バックアップからのタイムスタンプである場合は、全体バックアップおよび指定の増分バックアップまでのすべての増分バックアップをリストアします。 ■ -t オプションが指定されない場合、config_backup_path 内の構成ファイルのバックアップのリストを表示し、終了します。終了後、コマンドを再実行し、-t オプションでこれらのファイルの 1 つを指定できます。 ■ 構成ファイルのバックアップからすべてのファイルを Oracle ホームにリストアします。所有者、グループ、権限およびタイムスタンプは保持されます。 ■ エラーが発生した場合は、*log_path*/config_rst_*timestamp* にログ・ファイルを作成します。 -n オプションでプロンプトを非表示にして、マネージャをバッチ・モードで使用できます。 プロセスの前提条件の詳細は、backup_config オプションを参照してください。 restore_config は、J2EE クラスタ内の複数のノードで同時に実行しないでください。これを行うと、restore_config は失敗します。restore_config は、一度に 1 つのノードで実行してください。
restore_instance -t *timestamp*	Oracle Application Server のインスタンスをリストアします。タイムスタンプ引数を指定しない場合、バックアップ・タイムスタンプのリストが表示されます。このコマンドを実行すると、次の操作が行われます。 ■ OPMN が管理するすべてのプロセスを停止します。 ■ OPMN プロセスをチェックして、停止していることを確認します。OPMN プロセスを停止できない場合は（opmn.xml ファイルがない可能性があります）、ファイル・システムのリストアが実行されます。その後、OPMN プロセスの停止を再試行します。それでも OPMN プロセスを停止できない場合は、致命的エラーを発行します。 ■ OPMN 管理プロセスを起動します。 ■ 構成のリストアを実行します。 ■ OPMN が管理するすべてのプロセスを起動します。 ■ OPMN が管理するすべてのプロセスをチェックして、実行中であることを確認します。そうでない場合は、警告メッセージを発行します。

16.4.3 使用例

この項では、OracleAS Recovery Manager の使用例を示します。最初に UNIX コマンド、次に Windows コマンドを示します。

- デフォルトの config.inp ファイルを使用して、マネージャを構成する場合 :

```
bkp_restore.sh -m configure
bkp_restore.bat -m configure
```

- myconfig.inp という構成ファイルを使用して、マネージャを構成する場合 :

```
bkp_restore.sh -m configure -e myconfig.inp
bkp_restore.bat -m configure -e myconfig.inp
```

- Oracle Application Server インスタンスのコールド・バックアップを実行する場合 :

```
bkp_restore.sh -m backup_instance_cold
bkp_restore.bat -m backup_instance_cold
```

- Oracle Application Server インスタンスの増分コールド・バックアップを実行する場合 :

```
bkp_restore.sh -m backup_instance_cold_incr
bkp_restore.bat -m backup_instance_cold_incr
```

- Oracle Application Server インスタンスのオンライン・バックアップを実行する場合 :

```
bkp_restore.sh -m backup_instance_online
bkp_restore.bat -m backup_instance_online
```

- Oracle Application Server インスタンスのオンライン増分バックアップを実行する場合 :

```
bkp_restore.sh -m backup_instance_online_incr
bkp_restore.bat -m backup_instance_online_incr
```

- Oracle Application Server インスタンスを特定の時点の状態にリストアする場合 :

```
bkp_restore.sh -m restore_instance -t 2006-09-21_06-12-45
bkp_restore.bat -m restore_instance -t 2006-09-21_06-12-45
```

- Loss of Host Automation（LOHA）を使用して、ノードのバックアップ準備を行う場合 :

```
bkp_restore.sh -m node_backup -o prepare
bkp_restore.bat -m node_backup -o prepare
```

- LOHA を使用して、元のホストのイメージ・バックアップを作成する場合 :

```
bkp_restore.sh -m node_backup -o image_backup -P directory for image archive
bkp_restore.bat -m node_backup -o image_backup -P directory for image archive
```

- LOHA を使用して、新しいホストで Oracle Universal Installer 関連のメタデータをリストアする場合 :

```
bkp_restore.sh -m node_restore -o sys_init
bkp_restore.bat -m node_restore -o sys_init
```

- LOHA を使用して、新しいホストでインスタンスを登録する場合 :

```
bkp_restore.sh -m node_restore -o inst_register
bkp_restore.bat -m node_restore -o inst_register
```

- LOHA を使用して、新しいホストでインスタンスを構成する場合 :

```
bkp_restore.sh -m node_restore -o inst_reconfigure -t config_bkp_timestamp
bkp_restore.bat -m node_restore -o inst_reconfigure -t config_bkp_timestamp
```

16.4.4 バックアップのパージおよび 3 次ストレージへの移動

OracleAS Recovery Manager では、正常なバックアップの記録が backup_restore ディレクトリ内のカタログ・ファイル（data/catalog.txt）に保存されます。各バックアップはタイムスタンプで識別されます。インスタンスまたは構成のみのバックアップの場合は、構成ファイルのバックアップ・ディレクトリに保存される jar ファイルのファイル名に、タイムスタンプが付加されます。あるタイムスタンプに対応するすべての .jar ファイルを削除した場合や、他の場所（オフライン・ストレージなど）に移動した場合は、カタログにタイムスタンプの記録が残っていても、-m list_backups を実行したときにこの記録が表示されず、このタイムスタンプを -t 値として使用したリストアもできません。これは予期された動作です。

17

バックアップ計画と手順

この章では、Oracle Application Server のバックアップ計画および手順について説明します。

この章の項目は次のとおりです。

17.1 推奨されるバックアップ計画

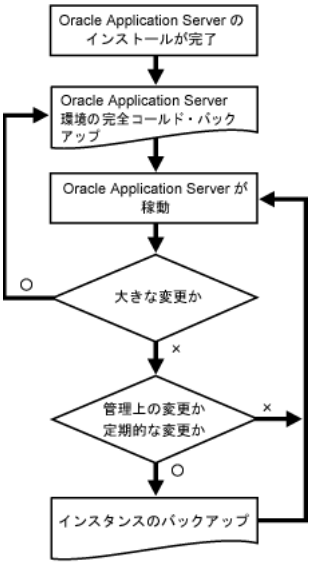
この項では、Oracle Application Server に推奨されるバックアップ計画について説明します。この計画に従って、このマニュアルで説明するリカバリ手順を実行することができます。

バックアップ計画には、次のものがあります。

- 作業 1: Oracle Application Server 環境の完全コールド・バックアップの実行
- 作業 2: インスタンスのバックアップの定期的な実行
- 作業 3: 環境の完全バックアップの再実行（大きな変更があった場合）
- 作業 4: インスタンスのバックアップの定期的な実行（作業 2 に戻る）
- 作業 5: ポートレット・プロデューサのバックアップの実行

図 17-1 のフロー・チャートは、特定の状況に適したバックアップ・タイプの決定方法の概要を示しています。

図 17-1 必要なバックアップ・タイプの決定



作業 1: Oracle Application Server 環境の完全コールド・バックアップの実行

最初に実行するバックアップは、イメージのバックアップです。このバックアップには、環境内のすべてのファイルが含まれます。環境の記録も作成してください。

1. Oracle Application Server 環境の完全バックアップを実行します。

   このバックアップは、以降のすべてのインスタンス・バックアップに対するベースラインとして機能します。

   詳細は、第 17.2.3 項「Oracle Application Server 環境の完全バックアップの実行」を参照してください。

2. Oracle Application Server 環境の記録を作成します。

   環境を再構成する必要がある場合、この記録を参照できます。

   詳細は、第 17.2.1 項「Oracle Application Server 構成の記録の作成」を参照してください。

作業 2: インスタンスのバックアップの定期的な実行

管理上の変更を実行するたびに、または（これが不可能な場合は）定期的に、Oracle Application Server 環境のインスタンスのバックアップを実行してください。

> **関連項目：** 管理上の変更の詳細は、付録 E「管理上の変更の例」を参照してください。

詳細は、第 17.2.2 項「コマンドラインからの Oracle Application Server インスタンスのバックアップの実行」を参照してください。

作業 3: 環境の完全バックアップの再実行（大きな変更があった場合）

Oracle Application Server 環境に大きな変更を加えた場合は、Oracle Application Server 環境のイメージのバックアップを改めて実行する必要があります。このバックアップは、以降のすべてのインスタンス・バックアップに対するベースラインとして機能します。また、環境の記録を新しい構成情報で更新する必要もあります。

次の処理の後に、イメージのバックアップを実行します。

- オペレーティング・システム・ソフトウェアのアップグレード
- Oracle Application Server ソフトウェアのアップグレードまたはパッチ適用

そのためには、次の手順を実行します。

1. Oracle Application Server 環境の記録を更新します。

   詳細は、第 17.2.1 項「Oracle Application Server 構成の記録の作成」を参照してください。

2. Oracle Application Server 環境の完全バックアップを実行します。

   詳細は、第 17.2.3 項「Oracle Application Server 環境の完全バックアップの実行」を参照してください。

作業 4: インスタンスのバックアップの定期的な実行（作業 2 に戻る）

改めて Oracle Application Server 環境の完全バックアップを実行した後で、作業 2 に戻り、インスタンスの定期的なバックアップを実行します。

作業 5: ポートレット・プロデューサのバックアップの実行

アプリケーションにリモートのポートレット・プロデューサが含まれている場合は、ポートレット・プロデューサのカスタマイズ・データおよびパーソナライズ・データをバックアップする必要があります。

その他のヒント：

- システム上に JRE/JDK のバックアップを作成します。これは Oracle 製品ではありませんが、Oracle Application Server によって使用されます。誤って損失や破損が発生した場合、Oracle Application Server が機能するためには、これをリストアする必要があります。これは、HP-UX、HP Tru64 および IBM AIX システムにのみ適用します。
- バックアップがリストアできることを定期的に確認して、バックアップの有効性を確保します。

17.2 バックアップ手順

この項では、バックアップ手順の詳細を説明します。構成データの一貫性を維持するには、各 Oracle Application Server インスタンスのバックアップを同時に作成する必要があります。ある Oracle Application Server インスタンスをバックアップしている間は、他のインスタンスの構成を変更しないでください。

この項の項目は次のとおりです。

- Oracle Application Server 構成の記録の作成
- コマンドラインからの Oracle Application Server インスタンスのバックアップの実行
- Oracle Application Server 環境の完全バックアップの実行
- ポートレット・プロデューサのバックアップの実行

17.2.1 Oracle Application Server 構成の記録の作成

Oracle Application Server 環境のリストアおよびリカバリが必要な場合、必要なすべての情報を入手し、対処することが重要です。これは、特に Oracle Application Server 環境全体（またはその一部）を新しいディスクまたはホストに再構成する必要があるような、ハードウェアの損失が発生した場合に当てはまります。

この項で説明されている情報を含む、Oracle Application Server 環境の最新記録を維持管理する必要があります。この情報は、印刷物と電子形式の両方で保管してください。電子形式のデータは、Oracle Application Server 環境とはまったく別のホストまたは電子メール・システム上に格納する必要があります。

Oracle Application Server のハードウェアおよびソフトウェア構成の記録には、次のものが含まれます。

- 環境内のホストごとに次の情報が必要です。
  - ホスト名
  - 仮想ホスト名（存在する場合）
  - ドメイン名
  - IP アドレス
  - ハードウェア・プラットフォーム
  - オペレーティング・システムのリリース・レベルおよびパッチ情報
- 環境内の Oracle Application Server インストールごとに次の情報が必要です。
  - インストールが常駐するホスト
  - Oracle ホームを所有するオペレーティング・システム・ユーザーのユーザー名、ユーザー ID 番号、グループ名、グループ ID 番号、環境プロファイルおよびシェル・タイプ（`/etc/passwd` および `/etc/group` エントリ）
  - *`ORACLE_HOME`* のディレクトリ構造、マウント・ポイントおよびフルパス
  - インストールで使用されるディスク領域の量
  - インストールで使用されるポート番号

注意： `opmnctl status -l` を使用して、使用中のポートを確認します。

17.2.2 コマンドラインからの Oracle Application Server インスタンスのバックアップの実行

この項では、Oracle Application Server インスタンスの各種バックアップをコマンドラインから実行する方法について説明します。インスタンス・レベルのバックアップでは、構成ファイル、中間層用のリポジトリを含む、アプリケーション・サーバー・インスタンスに必要なすべてのコンポーネントがバックアップされます。

Oracle Application Server 環境の完全バックアップを実行したら、それ以降は、管理上の変更があるたびに、またはそれが不可能であれば定期的に、インスタンス・レベルのバックアップを実行する必要があります。

Oracle Application Server インスタンスのコールド・バックアップの実行

次のコマンドを使用して、Oracle Application Server インスタンスのコールド・バックアップを実行します。

```
bkp_restore.sh -m backup_instance_cold
bkp_restore.bat -m backup_instance_cold
```

Oracle Application Server インスタンスの増分コールド・バックアップの実行

次のコマンドを使用して、Oracle Application Server インスタンスの増分コールド・バックアップを実行します。

```
bkp_restore.sh -m backup_instance_cold_incr
bkp_restore.bat -m backup_instance_cold_incr
```

Oracle Application Server インスタンスのオンライン・バックアップの実行

次のコマンドを使用して、Oracle Application Server インスタンスのオンライン・バックアップを実行します。

```
bkp_restore.sh -m backup_instance_online
bkp_restore.bat -m backup_instance_online
```

Oracle Application Server インスタンスの増分オンライン・バックアップの実行

次のコマンドを使用して、Oracle Application Server インスタンスの増分オンライン・バックアップを実行します。

```
bkp_restore.sh -m backup_instance_online_incr -l level
bkp_restore.bat -m backup_instance_online_incr -l level
```

17.2.3 Oracle Application Server 環境の完全バックアップの実行

この項では、Oracle Application Server 環境の完全バックアップを実行する方法について説明します。インストールまたはアップグレードの後には、ノードのバックアップを実行する必要があります。ホスト上のインスタンスごとに次の作業を実行します。

ノードの構成のバックアップ

次のコマンドを実行して、ノードの構成のバックアップを作成します。

UNIX の場合 :

```
bkp_restore.sh -m configure
```

Windows の場合 :

```
bkp_restore.bat -m configure
```

ノードのバックアップの準備

次のコマンドを実行して、ノードのバックアップを準備します。

UNIX の場合 :

```
bkp_restore.sh -m node_backup -o prepare
```

Windows の場合 :

```
bkp_restore.bat -m node_backup -o prepare
```

インスタンスのイメージのバックアップの作成

この作業では、Oracle ホーム、oratab、セントラル・インベントリ、Windows レジストリなどを含むインスタンスのアーカイブを作成します。UNIX の場合、ルートからコマンドを実行する必要があります。次のコマンドを実行して、インスタンスのイメージのバックアップを作成します。

UNIX の場合 :

```
bkp_restore.sh -m node_backup  -o image_backup -P archive path
```

Windows の場合 :

```
bkp_restore.bat -m node_backup  -o image_backup -P archive path
```

コマンドが完了すると、バックアップは *archive path* で指定されているディレクトリに格納されます。

17.2.4 ポートレット・プロデューサのバックアップの実行

リモート・ポートレット・プロデューサを含むアプリケーションを完全にバックアップおよびリカバリするには、次の 2 つの項目を追加でバックアップおよびリカバリする必要があります。

- ポートレット・プロデューサの Web アプリケーション自体 : プロデューサのアプリケーションは、Oracle Application Server 全体のバックアップおよびリカバリの一環としてバックアップおよびリカバリされます。ポートレット・プロデューサをリモートの Oracle Application Server インストールで実行している場合は、そのアプリケーション・サーバーもバックアップする必要があります。
- プロデューサのプリファレンス・ストア（ポートレットのパーソナライズ・データおよびカスタマイズ・データが含まれるプリファレンス・ストア）: プリファレンス・ストアをバックアップするには、プリファレンス・ストア移行ユーティリティを使用する必要があります。

Predeployment ツールを使用すると、Oracle Metadata Services（OMS）を構成するときに、`-backup` オプションを使用して OMS リポジトリの場所を指定できます。これにより、Recovery Manager では、`config_misc_files.inp` ファイルにこの場所を追加することによって、OMS リポジトリをバックアップできます。

注意 : ポートレットのプリファレンス・データは、リレーショナル・データベースまたはファイル・システムに格納できます。ポートレットのプリファレンス・ストアを構成する方法の詳細は、『Oracle WebCenter Framework 開発者ガイド』を参照してください。

注意 : 障害時のバックアップおよびリカバリの詳細は、『Oracle Application Server 高可用性ガイド』の Disaster Recovery に関する項を参照してください。

ポートレット・プロデューサのプリファレンス・ストアをバックアップするには、JPS および PDK-Java のプリファレンス・ストア移行ユーティリティを使用できます。

- JPS プリファレンス・ストアのバックアップ
- PDK-Java プリファレンス・ストアのバックアップ

17.2.4.1 JPS プリファレンス・ストアのバックアップ

JPS プリファレンス・ストアをバックアップするには、次の手順を実行します。

1. ポートレット・コンテナが実行している OC4J インスタンスを停止します。
2. 移行ツールを実行して、ソースのプリファレンス・ストアからバックアップ先のプリファレンス・ストアにデータをバックアップします。次に例を示します。

```
java -classpath wsrp-container.jar:cache.jar:saaj-api.jar:orasaaj.jar:ojdbc14.jar
oracle.portlet.server.containerimpl.PersistenceMigrationTool
-sourceType db
-destType file
-sourceDatabase portaldb.mycompany.com:1521:orcl
```

```
-sourceUsername p1
-sourcePassword p1
-destPath /tmp/portletbkp
```

3. ポートレット・コンテナが実行している OC4J インスタンスを起動します。

PersistenceMigrationTool の構文

PersistenceMigrationTool の構文は次のとおりです。

```
java oracle.webdb.wsrp.server.PersistenceMigrationTool
-sourceType file | db
-destType file | db
{-sourcePath dir |
 -sourceUsername username -sourcePassword password -sourceDatabase db}
{-destPath dir | destUsername username -destPassword password -destDatabase db}
[-debug]
```

前述の構文に関する説明を次に示します。

sourceType は、ソース・ストアがファイルまたはデータベースのどちらに存在するかを示します。ソース・ストアとバックアップ先ストアのタイプが同じ場合があります。したがって、異なるデータベース間やファイル・システム間での移行が可能です。

destType は、バックアップ先ストアがファイルまたはデータベースのどちらに存在するかを示します。ソース・ストアとバックアップ先ストアのタイプが同じ場合があります。したがって、異なるデータベース間やファイル・システム間での移行が可能です。

sourcePath は、ファイルベースのプリファレンス・ストアの場所です。この引数は、sourceType が file である場合に必要です。

sourceUsername は、プリファレンス・ストアのデータベースのデータベース・ユーザー名です。この引数は、sourceType が db である場合に必要です。

sourcePassword は、プリファレンス・ストアのデータベースのデータベース・パスワードです。この引数は、sourceType が db である場合に必要です。

sourceDatabase は、プリファレンス・ストアのデータベースの名前です。この引数は、sourceType が db である場合に必要です。

destPath は、ファイルベースのプリファレンス・ストアの場所です。この引数は、destType が file である場合に必要です。

destUsername は、プリファレンス・ストアのデータベースのデータベース・ユーザー名です。この引数は、destType が db である場合に必要です。

destPassword は、プリファレンス・ストアのデータベースのデータベース・パスワードです。この引数は、destType が db である場合に必要です。

destDatabase は、プリファレンス・ストアのデータベースの名前です。この引数は、destType が db である場合に必要です。

debug を指定すると、標準出力を使用するフル・ロギングが有効になり、ツール実行時に発生する問題を診断できます。

注意： 次のコマンドを入力すると、コマンドラインの構文を調べることができます。

```
java -classpath
ORACLE_HOME/j2ee/OC4J_WebCenter/shared-lib/oracle.wsrp/1.0/
   wsrp-container.jar:
ORACLE_HOME/javacache/lib/cache.jar:
ORACLE_HOME/j2ee/OC4J_WebCenter/webservices/lib/saaj-api.jar:
ORACLE_HOME/j2ee/OC4J_WebCenter/webservices/lib/orasaaj.jar:
ORACLE_HOME/j2ee/OC4J_WebCenter/jdbc/lib/ojdbc14.jar
oracle.portlet.server.containerimpl.PersistenceMigrationTool
```

プリファレンス・ストアの指定例

WSRP プロデューサのプリファレンス・ストアのタイプを調べるには、web.xml ファイル（例 17-1）を確認します。

例 17-1 web.xml ファイル内の persistentStore 変数および fileStoreRoot 変数

```
<env-entry>
    <env-entry-name>oracle/portal/wsrp/server/persistentStore</env-entry-name>
    <env-entry-type>java.lang.String</env-entry-type>
    <env-entry-value>File</env-entry-value>
</env-entry>
<env-entry>
    <env-entry-name>oracle/portal/wsrp/server/fileStoreRoot</env-entry-name>
    <env-entry-type>java.lang.String</env-entry/type>
    <env-entry-value>portletdata</env-entry-value>
</env-entry>
```

JPS ポートレット移行ユーティリティの詳細は、『Oracle WebCenter Framework 開発者ガイド』を参照してください。

17.2.4.2 PDK-Java プリファレンス・ストアのバックアップ

PDK-Java プリファレンス・ストアをバックアップするには、次の手順を実行します。

1. ポートレット・コンテナが実行している OC4J インスタンスを停止します。
2. 移行ツールを実行して、ソースのプリファレンス・ストアからバックアップ先のプリファレンス・ストアにデータをバックアップします。次に例を示します。

```
java -classpath $ORACLE_HOME/portal/jlib/pdkjava.jar
 oracle.portal.provider.v2.preference.MigrationTool
 -mode dbtofile
 -remap locale
 -countries AR,MX
 -pref1UseHashing true
 -pref1User portlet_prefs
 -pref1Password portlet_prefs
 -pref1URL jdbc:oracle:thin:@myserver.mydomain.com:1521:mysid
 -pref2RootDirectory /tmp/portletbkp
```

3. ポートレット・コンテナが実行している OC4J インスタンスを起動します。

移行ツールの構文

移行ユーティリティの構文は次のとおりです。

```
java -classpath $ORACLE_HOME/portal/jlib/pdkjava.jar
 oracle.portal.provider.v2.preference.MigrationTool
  -mode [file | db | filetodb | dbtofile | dbtodb]
  [-remap language | locale]
  [-countries iso_country_code]
  [-pref1UseHashing true | false]
  {-pref1RootDirectory directory |
   -pref1User username -pref1Password password -pref1URL url}
  [-pref1UseHashing true | false]
  {-pref2RootDirectory directory |
   -pref2User username -pref2Password password -pref2URL url}
  [-upfixwpi filename]
```

前述の構文に関する説明を次に示します。

-mode は、プリファレンス・ストアの移行およびアップグレードを行うユーティリティの実行モードです。

- file または db を指定すると、アップグレード・モードで実行します。
- filetodb、dbtofile または dbtodb を指定すると、移行モードで実行します。

-remap は、localePersonalizationLevel（language または locale）です。このオプションを使用する必要があるのは、アップグレードや移行の一環として localePersonalizationLevel を変更する場合のみです。

-countries を使用して、ISO 国コードに優先順位を付けたリストを指定できます。このリストは、複数の国のプリファレンスを再マップして競合が発生した場合のプリファレンスの優先順位を示します。-countries は、-remap オプションも同時に指定した場合にのみ機能します。

-pref1UseHashing は、この操作のソースでハッシングを使用するかどうかを示します。

-pref1RootDirectory は、ソース・ファイル・システムのパス（j2ee/home/applications/jpdk/jpdk/WEB-INF/providers/sample など）です。

-pref1User は、ソース・データベースのユーザー名です。

-pref1Password は、ソース・データベースのパスワードです。

-pref1URL は、ソース・データベースの URL（jdbc:oracle:thin:@myserver.mydomain.com:1521:mysid など）です。

-pref2UseHashing は、アップグレードまたは移行でハッシングを使用するかどうかを示します。

-pref2RootDirectory は、アップグレードまたは移行を行う対象となるファイル・システムのパス（j2ee/home/applications/jpdk/jpdk/WEB-INF/providers/sample など）です。

-pref2User は、アップグレードまたは移行を行う対象となるデータベースのユーザー名です。

-pref2Password は、アップグレードまたは移行を行う対象となるデータベースのパスワードです。

-pref2URL は、アップグレードまたは移行を行う対象となるデータベースの URL（jdbc:oracle:thin:@myserver.mydomain.com:1521:mysid など）です。

-upfixwpi は、操作のログ・ファイルを示します。

注意： 次のコマンドを入力すると、コマンドラインの構文を調べることができます。

```
java -classpath C:¥JDEV_HOME¥adfp¥lib¥pdkjava.jar;
  C:¥jdev_10132_wcs_4007¥adfp¥lib¥ptlshare.jar
  oracle.portal.provider.v2.preference.MigrationTool
```

プリファレンス・ストアの指定例

PDK-Java プロデューサのプリファレンス・ストアのタイプを調べるには、provider.xml ファイル（例 17-2）を確認します。

例 17-2

```
<provider class="oracle.portal.provider.v2.DefaultProviderDefinition">
  <localePersonalizationLevel>none</localePersonalizationLevel>
  <session>true</session>
  <defaultLocale>en</defaultLocale>
  <preferenceStore
    class="oracle.portal.provider.v2.preference.FilePreferenceStore">
```

```
    <name>prefStore1</name>
  </preferenceStore>
  <portlet
   class="oracle.portal.sample.v2.devguide.prefstore.GuestBookPortletDefinition">
   <id>1</id>
   <name>GuestBook</name>
   <title>Guest Book Portlet</title>
   <shortTitle>Guest Book</shortTitle>
   <description>Demonstration of using a Preference Store to drive
     portlet content</description>
   <timeout>100</timeout>
   <timeoutMessage>Guest Book Portlet timed out</timeoutMessage>
   <renderer class="oracle.portal.provider.v2.render.RenderManager">
    <showPage>/htdocs/prefstore/guest_book.jsp</showPage>
    <editPage>/htdocs/prefstore/store_comment.jsp</editPage>
   </renderer>
  </portlet>
</provider>
```

PDK-Java の移行とアップグレードを行うユーティリティの詳細は、『Oracle WebCenter Framework 開発者ガイド』を参照してください。

17.3 ホストの破損の自動リカバリ

OracleAS Recovery Manager では、1 つのホスト上のインスタンスの完全バックアップを実行して、元の動作環境が損なわれた場合にそれらのインスタンスを新しいホストにリストアする手順が自動化されています。

Loss of Host Automation（LOHA）によって、Oracle Application Server 管理者が異なるホスト間で Oracle Application Server インスタンスを移行する場合に必要な作業が自動化されます。新しいホストは、同じオペレーティング・システムを実行する別のホストであっても、システムのイメージを再導入した後の同一ホストであってもかまいません。LOHA には、ホストが喪失した場合に、インスタンスの再インストールおよびアプリケーション・データの保存を行うことなく、元のインスタンスを新しい環境にリストアするためのソリューションが用意されています。

LOHA では、すべての中間層インストールがサポートされており、新しいホスト名を元のホスト名と同じにすることも別の名前にすることもできます。ホスト名が異なる場合は、手動の作業が必要になります。

LOHA では、新しいホストにすでに実行している別の Oracle Application Server インスタンスがない場合に、1 つのホストから新しいホストにすべての Oracle Application Server インスタンスを移動できます。インスタンスのサブセットについては、元のホストに残っているインスタンスとの依存関係がない場合に、これらのサブセットを新しいホストにリストアできます。複数のホストから単一のホストにインスタンスをリストアすることはできません。

LOHA を使用すると、同じホスト上にある他のインスタンスに影響することなく、破損したインスタンスをリカバリすることもできます。

この項の項目は次のとおりです。

- Loss of Host Automation 使用の準備
- Loss of Host Automation の有効化
- 新しいホストでのノードのリストア
- 同じホストのインスタンスのリカバリ

17.3.1 Loss of Host Automation 使用の準備

Loss of Host Automation サービスは、OracleAS Recovery Manager の一部としてインストールされます。これは、次のディレクトリにインストールされます。

UNIX の場合 :

ORACLE_HOME/backup_restore/loha

Windows の場合 :

ORACLE_HOME¥backup_restore¥loha

Loss of Host Automation サービスを使用するには、第 16 章「Oracle Application Server Recovery Manager」 の説明に従って、OracleAS Recovery Manager を構成する必要があります。

Loss of Host Automation サービスには、次の前提条件があります。

- 新しいホストのオペレーティング・システムのバージョンとパッチのレベルは、Oracle Application Server に必要なものと同じである必要があります。
- `-invPtrLoc` のインストーラ・コマンドライン・オプションを指定してインスタンスをインストールした場合のみ、`config.inp` ファイルの `orainst_loc_path` フィールドを変更する必要があります。`oraInst.loc` の場所が標準以外である場合に、それを反映して変更する必要があります。
- Windows プラットフォームの場合、Windows Support Files（WSF）をインストールする必要があります。WSF は、Oracle Application Server のインストール CD-ROM から取得できます。Windows システム・ファイルのインストール方法は、Oracle Application Server のインストレーション・ガイドを参照してください。
- Windows プラットフォームの場合、Microsoft 社のサービス・ユーティリティ `sc.exe` を元のホストと新しいホストの両方にインストールする必要があります。Microsoft 社によると、このユーティリティは NT Resource Kit の一部です。Windows XP の場合、このユーティリティはインストールの一部になっています。Windows 2000 プラットフォームの場合、このユーティリティをインストールする必要があります。これが、実行パス内に存在することを確認します。
- UNIX プラットフォームの場合は、ORACLE_HOME 環境変数の最後にスラッシュ（/）がないことを確認します。
- 新しいホストでは、`jar`（Windows）または `tar`（UNIX）を使用したノード・アーカイブの解凍が可能である必要があります。ご使用のシステムに独自の tar プログラムがある場合は、GNU tar のかわりにそのプログラムを使用します。
- ユーザーには、system レベルまたは root レベルの作業を実行できるようにシステムの管理権限が必要です。
- 新しいホストに他の Oracle 製品がインストールされていないことを確認してください。たとえば、新しいホストに Oracle Application Server インスタンスがある場合は、これを正しく停止してアンインストールする必要があります。
- 新しいホストのユーザーまたはグループ ID は、元のホストのものと一致させる必要があります。
- 新しいホストでのポートの使用方法を確認します。リストアする Oracle Application Server インストールと同じポートを使用するプロセスがないことを確認します。同じポートを使用するプロセスがある場合は、Oracle Application Server インスタンスをリストアする前に、別のポートを使用するようにそのプロセスを再構成します。
- リストアを完了すると、元の中間層の Oracle ホームと同じマウント・ポイントおよびフルパスが保存されます。Oracle ホームの親ディレクトリが、中間層インストールを保持するのに十分な容量を持つファイル・システムにあり、また元のホストと同じユーザーとグループがこのディレクトリを所有していることを確認します。

17.3.2 Loss of Host Automation の有効化

Loss of Host Automation サービスを有効化するには、元のホストの各インスタンスに対して次の作業を実行する必要があります。

ノードの構成のバックアップ

インストールまたはアップグレードの後には、ノードのバックアップを実行する必要があります。次のコマンドを実行して、ノードの構成のバックアップを作成します。

UNIX の場合 :

```
bkp_restore.sh -m configure
```

Windows の場合 :

```
bkp_restore.bat -m configure
```

ノードのバックアップの準備

ノードのバックアップの準備では、現在のホストに関する次の情報の調査が、Loss of Host Automation サービスによって行われます。

- オペレーティング・システム
- ホスト名
- IP アドレス
- ユーザーまたはグループ ID
- インストール・タイプ
- セントラル・インベントリの場所
- Oracle ホームの場所
- Oracle ホームに対して作成されている Windows レジストリおよびすべての Windows サービス

この処理では、Loss of Host Automation サービスによってインスタンスのバックアップも作成されます。

次のコマンドを実行して、ノードのバックアップを準備します。

UNIX の場合 :

```
bkp_restore.sh -m node_backup -o prepare
```

Windows の場合 :

```
bkp_restore.bat -m node_backup -o prepare
```

元のホストのイメージのバックアップの作成

この作業では、元の Oracle ホーム、oratab、セントラル・インベントリ、Windows レジストリなどを含むインスタンスのアーカイブが作成されます。UNIX の場合、ルートからコマンドを実行する必要があります。次のコマンドを実行して、元のインスタンスのイメージのバックアップを作成します。

UNIX の場合 :

```
bkp_restore.sh -m node_backup  -o image_backup -P archive path
```

Windows の場合 :

```
bkp_restore.bat -m node_backup  -o image_backup -P archive path
```

コマンドが完了すると、バックアップは *archive path* で指定されているディレクトリに格納されます。

17.3.3 新しいホストでのノードのリストア

この項で示すコマンドによって、ホストの損失後に新しいホストでノードがリストアされます。次の手順を実行する前に、第 17.3.1 項「Loss of Host Automation 使用の準備」のすべての前提条件が満たされていることを確認します。

次の各コマンドを順序正しく実行する必要があります。

1. 古いノードのバックアップ・アーカイブを解凍します。

   UNIX の場合、次のように root としてログインします。

   ```
   cd /
   tar -xvpf archive_name
   ```

   Windows の場合 :

   ```
   jar -xvf archive_name
   ```

2. 次のコマンドによって、oratab（UNIX）、Windows レジストリ、セントラル・インベントリなど、Oracle Universal Installer に関連するメタデータが新しいホストにリストアされます。複数のインスタンスをリストアする場合は、最初のインスタンスにのみこの操作を実行してください。コマンドは UNIX 上で root として実行する必要があります。

   UNIX の場合 :

   ```
   bkp_restore.sh -m node_restore -o sys_init
   ```

   Windows の場合 :

   ```
   bkp_restore.bat -m node_restore -o sys_init
   ```

3. 次のコマンドによって、oratab とセントラル・インベントリにインスタンスが登録されます。また、UNIX の場合は root.sh の実行によってデーモンの起動および停止スクリプトが設定され、Windows の場合は Windows サービスが作成されます。コマンドは UNIX 上で root として実行する必要があります。

   UNIX の場合 :

   ```
   bkp_restore.sh -m node_restore -o inst_register
   ```

   Windows の場合 :

   ```
   bkp_restore.bat -m node_restore -o inst_register
   ```

4. このコマンドによって、新しいホストでインスタンスが再構成されます。再構成では、インストール・タイプに応じて、IP の変更、構成のバックアップのリストアなどが実行されます。このコマンドを実行する前に、opmnctl shutdown を実行して、再構成のプロセスに必要なポートが OPMN プロセスと Enterprise Manager プロセスで使用されていないことを確認します。Windows に Infrastructure と Metadata Repository がインストールされている環境では、このコマンドを実行する前に手動で flashback_recovery_area を作成する必要があります。このコマンドは、インスタンスの所有者として実行する必要があります。インスタンスのバックアップへのパスが有効である必要があります。

   UNIX の場合 :

   ```
   bkp_restore.sh -m node_restore -o inst_reconfigure -t config_bkp_timestamp
   ```

   Windows の場合 :

   ```
   bkp_restore.bat -m node_restore -o inst_reconfigure -t config_bkp_timestamp
   ```

   タイムスタンプ引数を指定しない場合、このコマンドによって使用可能なインスタンス・バックアップがすべて表示されます。この操作を正しく完了するには、他の必要なサービスがすべて、このインスタンスに属していない場合に稼動していることを確認します。

LOHA では、新しいホストでのポート競合は検出されません。リストアするインスタンスで使用する TCP ポートを使用する他のアプリケーションを実行しないことをお薦めします。ポート競合が発生した場合、この操作は失敗します。

5. `Oracle_Home/backup_restore/config/config_misc_files.inp_<time_stamp>` ファイルが存在し、既存の `config_misc_files.inp` ファイルより新しい場合は、`config_misc_files.inp` ファイルを、最新の `config_misc_files.inp_<time_stamp>` ファイルで上書きします。

   `Oracle_Home/backup_restore/plugin_config` ディレクトリ内のコンポーネント・プラグイン・ファイルごとに、`config_component_name_plugin.inp` ファイルのタイムスタンプより新しいタイムスタンプのファイルがあるかどうかを確認します。新しいタイムスタンプのファイルがある場合は、`config_component_name_plugin.inp` ファイルを最新のタイムスタンプのファイルで上書きします。

17.3.4 同じホストのインスタンスのリカバリ

Oracle Application Server インスタンスの問題を解決するためにイメージのリストアが必要な場合、LOHA を使用してインスタンスをリカバリできます。次の手順を実行して、インスタンスをリカバリします。

1. インスタンスを完全に停止します。
2. 第 17.3.3 項「新しいホストでのノードのリストア」のステップ 1 を実行して、インスタンスの最新のイメージ・バックアップを解凍します。
3. 第 17.3.3 項「新しいホストでのノードのリストア」のステップ 3、4 および 5 を実行し、インスタンスを登録して構成します。

   このインスタンスが Oracle Application Server の他のインスタンスとなんらかの依存関係にある場合、他のインスタンスは稼動している必要があります。

18

リカバリ計画と手順

この章では、Oracle Application Server のリカバリ計画および手順について、障害および停止のタイプ別に説明します。

この章の項目は次のとおりです。

18.1 リカバリ計画

この項では、様々な障害および停止のタイプ別に、Oracle Application Server のリカバリ計画について説明します。この項の項目は次のとおりです。

- データ損失、ホスト障害またはメディア障害に対するリカバリ計画（クリティカル）
- プロセスの障害およびシステムの停止に対するリカバリ計画（非クリティカル）

18.1.1 データ損失、ホスト障害またはメディア障害に対するリカバリ計画（クリティカル）

この項では、ホストまたはディスクが再起動できず、永久に失われるような、実データの損失や破損、ホスト障害、またはメディア障害などが関係する停止に対するリカバリ計画について説明します。このタイプの障害では、Oracle Application Server 環境を再起動し、通常の処理を続行する前に、ある種のデータ・リストアが必要です。

この項で説明する計画では、中間層の Point-in-Time リカバリを使用します。

前提

この項で説明するリカバリ計画には、次の前提を適用します。

- 最後のバックアップ以降、管理上の変更は加えられていない。最後のバックアップ以降、管理上の変更が加えられている場合は、リカバリの完了後にそれらの変更を再適用する必要があります。

  関連項目： 管理上の変更の詳細は、付録 E「管理上の変更の例」を参照してください。

使用する計画の決定

表 18-1 に、リカバリ計画を示します。

中間層インストールでデータの損失、ホスト障害またはメディア障害が発生した場合は、この表の情報を参照します。該当する損失のタイプを検索し、推奨される手順を実行してください。

表 18-1 中間層インスタンスにおけるデータの損失、ホスト障害およびメディア障害に対するリカバリ計画

損失のタイプ	リカバリ計画
ホストの破損	ホストが破損した場合、次の 2 つのオプションがあります。 ■ 同じホスト名および IP アドレスを持つ新しいホストにリストアできます。 ■ 別のホスト名および IP アドレスを持つ新しいホストにリストアできます。 いずれの場合も、第 18.2.2 項「新しいホストへの中間層インストールのリストア」の手順に従います。 元のホストに中間層インストールと Infrastructure がある場合、別のホスト名または IP アドレスを持つホストに中間層をリストアすることはできません。
Oracle ソフトウェア / バイナリの削除または破損	Oracle バイナリで損失または破損が発生した場合、中間層全体を同じホストにリストアする必要があります。 第 18.2.1 項「同じホストへの中間層インストールのリストア」の手順に従います。
構成ファイルの削除および破損	中間層の Oracle ホームにある構成ファイルのいずれかを損失した場合、それらをリストアできます。 第 18.2.3 項「中間層の構成ファイルのリストア」の手順に従います。

18.1.2 プロセスの障害およびシステムの停止に対するリカバリ計画（非クリティカル）

この項では、プロセスの障害およびシステムの停止に対するリカバリ計画について説明します。このタイプの停止にはデータの損失は含まれないため、ファイルをリカバリする必要はありません。一部のケースでは、障害が透過的であるため、障害が発生したコンポーネントのリカバリに手動で介入する必要がない場合もあります。ただし、プロセスまたはコンポーネントの再起動に、手動による介入が必要な場合もあります。これらの計画は、バックアップおよびリカバリのカテゴリには厳密には一致しませんが、万全を期すためこのマニュアルに含まれています。

使用する計画の決定

表 18-2 に、プロセスの障害およびシステムの停止に対するリカバリ計画を示します。

中間層インストールで障害または停止が発生した場合は、この表を参照します。該当する停止のタイプを検索し、推奨される手順を実行してください。この表に記載しているのは UNIX のコマンドです。Windows ではスラッシュを「¥」と読み替えると、同じコマンドを使用できます。

表 18-2 中間層インスタンスにおけるプロセスの障害およびシステムの停止に対するリカバリ計画

停止のタイプ	ステータス確認と再起動の方法
ホストの障害（データの損失なし）	**再起動するには：** 1. ホストを再起動します。 2. 中間層を起動します。第 3.2.1 項「中間層インスタンスの起動」を参照してください。

Application Server Control コンソールの障害	**ステータスを確認するには：**
	`opmnctl status`
	再起動するには：
	`opmnctl startproc process-type=OC4J_instance_name`

Oracle HTTP Server プロセスの障害	**ステータスを確認するには：**
	`opmnctl status`
	再起動するには：
	`opmnctl startproc ias-component=HTTP_Server`

OC4J インスタンスの障害	**ステータスを確認するには：**
	`opmnctl status`
	再起動するには：
	`opmnctl startproc process-type=OC4J_instance_name`

OPMN デーモンの障害	**ステータスを確認するには：**
	`opmnctl status`
	再起動するには：
	`opmnctl start`

18.2 リカバリ手順

この項では、様々なタイプのリカバリを実行するための手順について説明します。

この項の項目は次のとおりです。

- 同じホストへの中間層インストールのリストア
- 新しいホストへの中間層インストールのリストア
- 中間層の構成ファイルのリストア
- Oracle Application Server インスタンスのリストア
- ポートレット・プロデューサのプリファレンス・ストアのリカバリ

18.2.1 同じホストへの中間層インストールのリストア

同じホストに中間層インストールをリストアするには、第 17.3.4 項「同じホストのインスタンスのリカバリ」を参照してください。

18.2.2 新しいホストへの中間層インストールのリストア

この項では、新しいホストに中間層インストールをリストアおよびリカバリする方法について説明します。この手順を使用して、次のことを実行できます。

- オペレーティング・システムの再インストール後、同じホストに中間層インストールをリストアします。
- 新しいホストに中間層インストールをリストアします。新しいホストは、元のホストと同じホスト名および IP アドレスを持つ場合もあり、またホスト名か IP アドレスまたはその両方が異なる場合もあります。

第 17.3.3 項「新しいホストでのノードのリストア」の手順を実行して、イメージのバックアップ、システム・ファイルおよびインスタンスの再構成をリストアします。中間層の構成は、元のインスタンスの状態と同じままです。ホスト名が同じままの場合、インスタンスのリストアを実行し、目的の時点のインスタンスに戻します。ホスト名が異なる場合、元のホストのバックアップは別のホスト名が有効でないので、状態を変更できません。

18.2.3 中間層の構成ファイルのリストア

この項では、中間層の Oracle ホーム内の構成ファイルをリストアする方法について説明します。構成ファイルの損失または破損が発生した場合は、この手順を使用します。

この項は、次の作業で構成されています。

- 作業 1: 中間層インスタンスの停止
- 作業 2: 中間層の構成ファイルのリストア
- 作業 3: 最近実行した管理上の変更の適用
- 作業 4: 中間層インスタンスの起動

作業 1: 中間層インスタンスの停止

手順の詳細は、第 3.2.2 項「中間層インスタンスの停止」を参照してください。

作業 2: 中間層の構成ファイルのリストア

最新のバックアップからすべての構成ファイルをリストアします。この作業は、独自の手順または OracleAS Recovery Manager を使用して実行できます。たとえば、OracleAS Recovery Manager を使用して次のコマンドを実行します。

- UNIX の場合 :

```
bkp_restore.sh -m restore_config -t timestamp
```

- Windows の場合：

```
bkp_restore.bat -m restore_config -t timestamp
```

関連項目： 詳細は、第 16 章「Oracle Application Server Recovery Manager」を参照してください。

作業 3: 最近実行した管理上の変更の適用

最後のオンライン・バックアップ以降、管理上の変更を加えた場合は、この時点でそれらの変更を再適用します。

関連項目： 管理上の変更の詳細は、付録 E「管理上の変更の例」を参照してください。

作業 4: 中間層インスタンスの起動

手順の詳細は、第 3.2.1 項「中間層インスタンスの起動」を参照してください。

18.2.4 Oracle Application Server インスタンスのリストア

次のコマンドを使用して、Oracle Application Server インスタンスを特定の時点の状態にリストアします。

```
bkp_restore.sh -m restore_instance -t 2006-09-21_06-12-45

bkp_restore.bat -m restore_instance -t 2006-09-21_06-12-45
```

クラスタ内のインスタンスでリストア操作（restore_instance または restore_config）を実行する前に、クラスタ全体のすべての OC4J プロセスを停止する必要があります。次のコマンドを使用してプロセスを停止します。

```
ORACLE_HOME/opmn/bin/opmnctl @cluster stopproc ias-component=OC4J
```

一部の OC4J コンポーネントには ias-component=OC4J がありません。このようなコンポーネントについては、uniqueid 値を使用して OC4J プロセスを停止します。uniqueid を持つコンポーネントを確認するには、次のコマンドを使用します。

```
ORACLE_HOME¥opmn¥bin¥opmnctl @cluster status -fmt %typ%uid%prt -noheaders
```

次に、このコマンドを実行した結果の例を示します。

```
CUSTOM      | N/A        | ASG
LOGLDR      | N/A        | logloaderd
OHS         | 1500577870 | HTTP_Server
performance | 1500577873 | performance_server
messaging   | 1500577874 | messaging_server
```

次のコマンドを実行して、2 番目の列（uid）の値が N/A ではない、すべての OC4J プロセスを停止します。

```
ORACLE_HOME¥opmn¥bin¥opmnctl @cluster stopproc uniqueid=1500577865

opmnctl: stopping opmn managed processes...
```

リストア操作の完了後、次のコマンドを使用してクラスタ全体の OC4J プロセスを再起動します。

```
ORACLE_HOME/opmn/bin/opmnctl @cluster startproc ias-component=OC4J
```

uniqueid を使用するコンポーネントについては、適切な ias-component 値を使用するか次のコマンドを実行してプロセスを再起動します。

```
opmnctl startall
```

18.2.5 ポートレット・プロデューサのプリファレンス・ストアのリカバリ

JPS ポートレット・プロデューサのプリファレンス・ストアをリカバリするには、次の手順を実行します。

1. コンテナが実行している OC4J インスタンスを停止します。
2. 移行ツールを実行して、ソースのバックアップ・ファイルからリストア先のプリファレンス・ストアにデータをリストアします。次に例を示します。

```
java -classpath
  wsrp-container.jar:cache.jar:saaj-api.jar:orasaaj.jar:ojdbc14.jar
  oracle.portlet.server.containerimpl.PersistenceMigrationTool
-sourceType file
-destType db
-sourcePath /tmp/portletbkp
-destUsername p1
-destPassword p1
-destDatabase portaldb.mycompany.com:1521:orcl
```

3. ポートレット・コンテナが実行している OC4J インスタンスを起動します。

JPS ポートレット移行ユーティリティの詳細は、『Oracle WebCenter Framework 開発者ガイド』を参照してください。

PDK-Java ポートレット・プロデューサのプリファレンス・ストアをリカバリするには、次の手順を実行します。

1. コンテナが実行している OC4J インスタンスを停止します。
2. 移行ツールを実行して、ソースのバックアップ・ファイルからリストア先のプリファレンス・ストアにデータをリストアします。次に例を示します。

```
java -classpath $ORACLE_HOME/portal/jlib/pdkjava.jar
   oracle.portal.provider.v2.preference.MigrationTool
  -mode filetodb
  -remap locale
  -countries AR,MX
  -pref1UseHashing true
  -pref1RootDirectory /tmp/portletbkp
  -pref2User portlet_prefs
  -pref2Password portlet_prefs
  -pref2URL jdbc:oracle:thin:@myserver.mydomain.com:1521:mysid
```

3. ポートレット・コンテナが実行している OC4J インスタンスを起動します。

PDK-Java の移行とアップグレードを行うユーティリティの詳細は、『Oracle WebCenter Framework 開発者ガイド』を参照してください。

19

OracleAS Recovery Manager のトラブルシューティング

この章では、OracleAS Recovery Manager の使用時に発生する場合がある一般的な障害と、その解決方法について説明します。この章は、次の項で構成されています。

19.1 障害と解決策

この項では、一般的な障害と解決策について説明します。この項の項目は次のとおりです。

- restore_config 操作時にファイルが見つからないことを示すメッセージの受信
- opmn.xml ファイルの消失または破損による失敗
- opmnctl stopall コマンドによるプロセス停止時におけるタイムアウトの発生

19.1.1 restore_config 操作時にファイルが見つからないことを示すメッセージの受信

restore_config 操作によって、ファイルが見つからないことを示すメッセージが生成されます。

障害

restore_config 操作時に、ファイルが見つからないことを示すメッセージを受信しました。次に例を示します。

```
Could not copy file C:¥Product¥OracleAS¥Devkit_1129/testdir/ to
C:¥Product¥OracleAS¥Devkit_1129¥backup_restore¥cfg_bkp/2006-12-01_03-26-22.
```

解決策

restore_config 操作時には、構成の一時バックアップが作成されます。これにより、リストアに失敗した場合に、この一時バックアップをリストアしてインスタンスをリストア前と同じ状態に戻せます。

リストア操作の前に一部のファイル（config_misc_files.inp に指定されているファイルとディレクトリを含む）が削除されている場合は、一時バックアップ時に、特定のファイルが見つからないことを示すメッセージが表示されます。見つからないファイルは restore_config 操作の一環としてリストアされるため、このエラーまたは警告メッセージは無視できます。

19.1.2 opmn.xml ファイルの消失または破損による失敗

opmn.xml ファイルの消失または破損により操作に失敗しました。

障害

opmn.xml ファイルの消失または破損により次のエラーが発生します。

```
ADMN-906025
Base Exception:
The exception, 100999, occurred at Oracle Application Server instance
"J2EE_1123.stad.oracle.com"
```

解決策

次の手順に従って、opmn.xml ファイルをリストアします。

1. 次のコマンドを実行します。

   ```
   bkp_restore.bat -m restore_config -t timestamp
   ```

2. このコマンドに失敗した場合は、OC4J プロセスを停止します。
3. 次のコマンドを再実行します。

   ```
   bkp_restore.bat -m restore_config -t timestamp
   ```

19.1.3 opmnctl stopall コマンドによるプロセス停止時におけるタイムアウトの発生

backup_instance_cold、backup_instance_cold_incr および restore_instance 操作時に opmnctl stopall コマンドを使用してプロセスを停止すると、タイムアウトが発生する場合があります。

障害

サーバー・インスタンスのバックアップやリストアなどの一部の操作では、opmnctl stopall コマンドを使用してプロセスを停止すると、タイムアウトが発生する場合があります。この障害は、マシンへの負荷が高いか、プロセスの停止に時間がかかっているときに発生する場合があります。この状況では、次のようなエラー・メッセージが表示されます。

```
Oracle Application Server instance backup failed.
Stopping all opmn managed processes ...

Failure : backup_instance_cold_incr failed

Unable to stop opmn managed processes !!!
```

解決策

opmnctl stopall を再度実行すると、この障害は解決されます。

第 VI 部

付録と用語集

この部は、次の付録と用語集で構成されています。

- 付録 A「Application Server Control の管理および構成」
- 付録 B「Oracle Application Server のコマンドライン・ツール」
- 付録 C「コンポーネントの URL」
- 付録 D「Oracle Application Server のポート番号」
- 付録 E「管理上の変更の例」
- 付録 F「LDAP ベースのレプリカ構成の補助手順」
- 付録 G「Oracle Application Server のリリース番号の確認」
- 付録 H「Oracle Application Server のトラブルシューティング」
- 「用語集」

A

Application Server Control の管理および構成

Oracle Application Server をインストールすると、Oracle Enterprise Manager 10*g* Application Server Control およびその関連プロセスが自動的に起動します。起動後は、すぐに Application Server Control コンソールを使用してアプリケーション・サーバーのコンポーネントを管理できます。

Application Server Control の制御および構成もできます。たとえば、Application Server Control の起動と停止、Application Server Control コンソールのパスワード変更、Application Server Control のセキュリティ構成などができます。

この付録では、Application Server Control の管理と構成の方法について説明します。この付録の項目は次のとおりです。

A.1 Application Server Control の起動と停止

10*g* リリース 3（10.1.3.2.0）では、Application Server Control は標準の J2EE アプリケーションとしてデプロイされます。Application Server Control アプリケーション（ascontrol）は、作成するすべての OC4J インスタンスに自動的にデプロイされます。

そのため、第 3.3.1 項「opmnctl を使用したコンポーネントの起動と停止」の手順を使用して、コマンドラインから ascontrol アプリケーションを起動および停止することができます。

アクティブな ascontrol アプリケーションは Application Server Control コンソールからも停止および再起動できますが、このリリースでデプロイされる他の J2EE アプリケーションと異なり、ascontrol アプリケーションの Application Server Control コンソールからの起動および停止には次のような制限が伴います。

- 1 つのスタンドアロン OC4J インスタンスを管理している場合は、Application Server Control コンソールから ascontrol アプリケーションを起動、停止または再起動することはできません。ascontrol アプリケーションを停止すると、Application Server Control コンソールを表示したり、使用したりできなくなります。
- 複数の OC4J インスタンスを管理するクラスタ環境では、「クラスタ・トポロジ」ページを使用して、アクティブな ascontrol アプリケーションを起動、停止または再起動できます。ただし、アクティブな ascontrol アプリケーションを停止することによる影響を警告するメッセージが表示されます。

  アクティブな ascontrol アプリケーションとは、Oracle Application Server 環境の管理に現在使用されている Application Server Control です。アクティブな ascontrol アプリケーションを停止すると、そのアプリケーションを起動するまで、Application Server Control コンソールを使用できなくなります。

  Application Server Control コンソールを使用して 2 つ目の ascontrol アプリケーションを起動することはできません。起動しようとすると、Application Server Control にアクティブな ascontrol アプリケーションがすでに実行中であることを示すエラー・メッセージが表示されます。

アクティブな ascontrol アプリケーションのデプロイに使用される OC4J インスタンスは、**管理 OC4J インスタンス**と呼ばれます。管理 OC4J インスタンス以外の OC4J インスタンスは、リモート OC4J インスタンスと呼ばれます。ほとんどの場合、リモート OC4J インスタンスで ascontrol を起動する必要はありません。

ただし、リモート OC4J インスタンスで ascontrol アプリケーションを実行しなければならない場合もあります。詳細は、Application Server Control のオンライン・ヘルプのリモート・ログ・ファイルを表示する場合の ascontrol の起動に関する項を参照してください。

A.1.1 Application Server Control が稼動中であることの確認

Application Server Control コンソールの URL に移動することによって、Application Server Control が起動されていることを確認できます。

```
http://hostname.domain:port/em
```

たとえば、次のように指定します。

```
http://mgmthost.acme.com:7777/em
```

Application Server Control コンソールのポート番号を探すには、次のコマンドを使用して HTTP_Server の番号を確認します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl status -l
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl status -l
```

関連項目： 第 2.3.1 項「Application Server Control コンソールの表示」

A.2 Application Server Control の管理パスワードの変更

Application Server Control を使用するには、Application Server Control の管理者アカウントが必要です。環境を管理するときの権限は、Application Server Control コンソールにログインするときに使用するユーザー・アカウントとパスワードによって決まります。

Oracle Application Server をインストールすると、デフォルトのスーパー管理者アカウントが作成されます。これは `oc4jadmin` というアカウントで、このアカウントのパスワードは、Oracle Application Server のインストール時に設定します。この oc4jadmin アカウントは、Application Server Control コンソールへの初回ログイン時に使用できます。その後、管理ユーザー・アカウントを追加作成して、日常の管理タスクに使用できます。

> **関連項目：** 第 2.3.2 項「自分用およびチーム・メンバー用の管理アカウントの作成」

管理者自身の管理アカウントのパスワードは、Application Server Control コンソールへのログイン時に使用するユーザー・アカウントに関係なく、いつでも変更できます。ただし、`oc4jadmin` パスワードを変更する際は、特別な注意事項があります。

次の各項で詳細を説明します。

- 自身の管理者アカウント・パスワードの変更
- oc4jadmin アカウントについて
- 管理 OC4J インスタンスの oc4jadmin パスワードの変更
- Application Server Control を使用したリモート OC4J インスタンスの oc4jadmin パスワードの変更
- コマンドラインを使用したリモート OC4J インスタンスの oc4jadmin パスワードの変更

A.2.1 自身の管理者アカウント・パスワードの変更

自身の管理者アカウントを変更する手順は次のとおりです。

1. 自身の管理者ユーザー名とパスワードを使用して、Application Server Control コンソールにログインします。
2. Application Server ホーム・ページにナビゲートして、ページの上部にある「**設定**」を選択します。

   「パスワード」ページが表示されます。このページの「**ユーザー**」フィールドに、変更するアカウントが表示されます。`oc4jadmin` ユーザー・アカウントを変更する場合の詳細は、第 A.2.3 項を参照してください。
3. 現行の管理者パスワードと新しいパスワードを入力し、さらに確認のために新しいパスワードをもう一度入力します。

   セキュリティを強化するために、新しいパスワードには次のような制限があります。

   - 文字列の長さは、5 ～ 30 文字にする必要があります。
   - 文字列の先頭には英文字を使用します。パスワードの先頭には、数字、アンダースコア（_）、ドル記号（$）または番号記号（#）を使用できません。
   - 少なくとも数字を 1 つ使用します。
   - 英数字および特殊文字（ドル記号（$）、番号記号（#）またはアンダースコア（_））のみを含めることができます。
   - Oracle 予約語（VARCHAR など）は使用できません。

   これらは、Application Server Control および Oracle Universal Installer による制限であり、OC4J の `system-jazn.xml` ファイルや、アプリケーションベースのセキュリティ構成ファイルによる制限ではありません。

4. 「**OK**」をクリックして、パスワードをリセットします。

   次回ログインするときには、この新しいパスワードを使用する必要があります。

A.2.2 oc4jadmin アカウントについて

Oracle Application Server 環境では、デフォルトの `oc4jadmin` 管理ユーザー・アカウントは 2 つの異なる用途に使用されます。それぞれの用途について、次の項で説明します。

- 初回ログイン用の oc4jadmin アカウントの使用
- 管理資格証明用の oc4jadmin アカウントの使用

A.2.2.1 初回ログイン用の oc4jadmin アカウントの使用

Oracle Application Server のインストール時に、`oc4jadmin` アカウントのパスワードを定義する必要があります。この `oc4jadmin` アカウントは、Application Server Control への初回ログイン時に使用できます。

`oc4jadmin` ユーザー・アカウントには、自動的に一連の管理ロールが割り当てられます。これにより、このアカウントでログインしたユーザーは、OC4J インスタンスのあらゆる側面を管理および構成できます。しかし、この `oc4jadmin` アカウントを日常の管理タスクには使用しないでください。日常の管理タスクに使用する管理アカウントは、別途追加作成します。

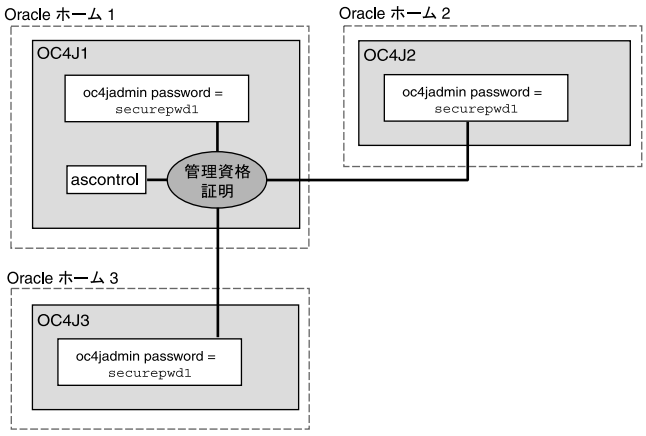
> **関連項目：** 第 2.3.2 項「自分用およびチーム・メンバー用の管理アカウントの作成」

A.2.2.2 管理資格証明用の oc4jadmin アカウントの使用

`oc4jadmin` アカウントは、Application Server Control への初回ログイン・アカウントとしての用途のほかに、Application Server Control ソフトウェアでの管理タスクの実行にも使用されます。特に、OC4J インスタンスで構成タスクを実行する場合、Application Server Control は `oc4jadmin` ユーザー・アカウントとパスワードを使用してインスタンスに接続します。このような場合の `oc4jadmin` ユーザー・アカウントとパスワードは、**管理資格証明**と呼ばれます。

具体的には、クラスタ・トポロジ・ページ上で OC4J インスタンス名をクリックすると、Application Server Control は管理資格証明を使用して OC4J インスタンスに接続します。Application Server Control はこれらの資格証明を使用して、ローカル管理 OC4J およびクラスタ内のリモート OC4J インスタンスの両方に接続します。この概念について、図 A-1 に説明しています。

図 A-1 Application Server Control でクラスタ内 OC4J インスタンスの管理に管理資格証明を使用する方法



各クラスタには管理資格証明が 1 組のみ存在します。デフォルトでは、管理 OC4J（図 A-1 の Oracle ホーム 1）用に定義した oc4jadmin アカウントとパスワードが、クラスタの管理資格証明として保存されています。

そのため、クラスタ内の各 OC4J インスタンスに oc4jadmin ユーザー・アカウントが存在する必要があり、その oc4jadmin アカウントのパスワードは管理 OC4J に定義されたパスワードと同じである必要があります。そうしないと、クラスタに定義した管理資格証明は機能せず、Application Server Control は OC4J インスタンスに接続できなくなります。

後で oc4jadmin アカウントのパスワードを変更することはできますが、Oracle Application Server インスタンスのクラスタを管理する場合は、oc4jadmin パスワードを変更する際に注意が必要です。

A.2.3 管理 OC4J インスタンスの oc4jadmin パスワードの変更

管理 OC4J インスタンスの oc4jadmin パスワードを変更する手順は、自身の管理者パスワードを変更する手順と同じです。oc4jadmin ユーザー名とパスワードを使用してログインし、「**設定**」をクリックします。

ただし、oc4jadmin パスワードを変更すると、Application Server Control コンソールから実行する一部の操作に影響が生じる可能性があります。これは、oc4jadmin のユーザー名とパスワードがクラスタ・トポロジの管理資格証明として使用されているためです。詳細は、第 A.2.2.2 項を参照してください。

Application Server Control コンソールの任意のページで「**設定**」をクリックして oc4jadmin パスワードを変更した場合は、管理 OC4J インスタンスの oc4jadmin アカウントのパスワードのみが変更されます。

「設定」リンクからのパスワード変更は、リモート OC4J インスタンスで使用されている oc4jadmin パスワードには適用されません。リモート OC4J インスタンスとは、アクティブな Application Server Control をホストしていないクラスタ・トポロジ内の OC4J インスタンスです。

そのため、Application Server Control コンソールの「設定」リンクを使用して、リモート OC4J インスタンスで定義したものと異なるように oc4jadmin パスワードを変更すると、「クラスタ・トポロジ」ページからこれらのインスタンスにナビゲートできなくなります。管理資格証明が無効になった OC4J インスタンスのホーム・ページを表示しようとすると、Enterprise Manager によって次のようなエラー・メッセージが表示されます。

```
Unable to make a connection to OC4J instance instance_name on Application
Server application_server_name. A common cause for this failure is an
authentication error. The administrator password for each OC4J instance in the
Cluster must be the same as the administrator password for the OC4J instance on
which Application Server Control is running.
```

この問題を解決するには、リモート OC4J インスタンスの管理資格証明を、管理 OC4J の管理資格証明と一致するように変更する必要があります。

A.2.4 Application Server Control を使用したリモート OC4J インスタンスの oc4jadmin パスワードの変更

クラスタ・トポロジで複数の OC4J インスタンスを管理している場合は、「クラスタ・トポロジ」ページの上部にある「設定」リンクを使用すると、ascontrol アプリケーションをホストする管理 OC4J のパスワードを変更できます。

一方、クラスタ・トポロジのリモート OC4J インスタンスの oc4jadmin パスワードを変更するには、次の手順を実行する必要があります。

この手順では、次のことを前提としています。

- oc4jadmin アカウントが、クラスタの管理資格証明として使用されているものとします。

- oc4jadmin パスワードは、現時点では管理 OC4J インスタンスとリモート OC4J インスタンスで同じものとします。それらが異なる場合、Application Server Control を使用してリモートの oc4jadmin パスワードを変更することはできません。その場合は、第 A.2.5 項で説明している手順を実行します。

Application Server Control を使用してリモート OC4J インスタンスの oc4jadmin パスワードを変更するには、次の手順を実行します。

1. 「クラスタ・トポロジ」ページで、変更するリモート OC4J インスタンスの名前をクリックします。

   アクティブな ascontrol アプリケーションをホストする管理 OC4J ではなく、必ずリモート OC4J インスタンスを選択してください。

   関連項目： 第 2.3.3.1 項「クラスタ・トポロジの表示とアクティブな Application Server Control の検索」

   選択したリモート・インスタンスの OC4J ホーム・ページが表示されます。
2. 「**管理**」をクリックして、選択した OC4J インスタンスで実行できる管理タスクの一覧を表示します。
3. 表の「**セキュリティ・プロバイダ**」行にあるタスク・アイコンをクリックします。
4. 「セキュリティ・プロバイダ」ページで、「**インスタンス・レベルのセキュリティ**」をクリックします。
5. 「インスタンス・レベルのセキュリティ」ページで、「**レルム**」をクリックします。
6. 「結果」表の jazn.com 行で、「ユーザー」列内の数値（3 など）をクリックします。

   選択したセキュリティ・プロバイダに定義されているユーザーが一覧表示されます。
7. **oc4jadmin** をクリックして、oc4jadmin ユーザー・アカウントを変更します。
8. 「ユーザー」ページのパスワード・フィールドを使用して、このリモート OC4J インスタンスの oc4jadmin アカウントのパスワードを変更し、「**適用**」をクリックします。
9. 「クラスタ・トポロジ」ページに戻り、リモート OC4J インスタンスを再起動します。

A.2.5 コマンドラインを使用したリモート OC4J インスタンスの oc4jadmin パスワードの変更

Application Server Control を使用せずにリモート・アカウントの oc4jadmin パスワードを変更するには、次の手順を実行します。

1. リモート OC4J がインストールされ稼動しているコンピュータにログインします。
2. OC4J インスタンスの Oracle ホームで次の構成ファイルを探します。

   ORACLE_HOME/j2ee/*oc4j_instance_name*/config/system-jazn-data.xml
3. テキスト・エディタを使用して system-jazn-data.xml ファイルを開き、oc4jadmin ユーザーの次のエントリを探します。

```
<user>
     <name>oc4jadmin</name>
     <display-name>OC4J Administrator</display-name>
     <guid>41A2E560C96711DABFD08D3BF8B780C4</guid>
     <description>OC4J Administrator</description>
     <credentials>{903}4nlfYYDwaqMJipVbGXuS2ce8egfwBPqp</credentials>
</user>
```

4. <credentials> 要素の値を、新しいパスワードに置き換えます。パスワードの前に感嘆符（!）を必ず付けてください。

たとえば、次のように指定します。

```
<credentials>!abcdefg1234</credentials>
```

この例の *abcdefg1234* の部分を、実際に使用するパスワードに置き換えます。

感嘆符（!）を使用することにより、パスワードはこの構成ファイル内で暗号化されます。

> **関連項目：** 『Oracle Containers for J2EE 構成および管理ガイド』の OC4J 構成ファイルでのパスワードの不明瞭化に関する項

5. 変更を保存して、`system-jazn-data.xml` ファイルを終了します。
6. OC4J インスタンスを再起動します。

   たとえば、次の Oracle Process Manager and Notification Server（OPMN）コマンドを使用して、Oracle Application Server インスタンスを停止した後に起動します。

```
ORACLE_HOME/opmn/bin/opmnctl stopall
ORACLE_HOME/opmn/bin/opmnctl startall
```

A.3 Application Server Control コンソールのセキュリティの構成

Application Server Control コンソールを保護するには、次の 2 種類の通信リンクを保護する必要があります。

- ブラウザ・クライアントとサーバー間の通信
- Oracle Application Server コンポーネント間の通信

セキュリティを有効にすると、トレードオフが発生します。つまり、セキュリティの強化には、SSL の使用や、より高い処理能力とより多くのメモリーが必要となります。したがって、セキュリティ対策は、環境に応じて必要な部分に施すことをお薦めします。

次の各項では、Application Server Control アプリケーションのセキュリティを構成する方法について説明します。

- ブラウザ・クライアントと Application Server Control コンソールをホストする Web サーバー間の通信の保護
- Oracle Application Server のコンポーネント間の通信の保護

> **注意：** この項では、Application Server Control コンソールを保護するために実行する手順の概要を示します。この項に記載されているセキュリティ設定およびセキュリティ・オプションの詳細は、次のドキュメントを参照してください。
>
> - 『Oracle Containers for J2EE セキュリティ・ガイド』
> - 『Oracle HTTP Server 管理者ガイド』

A.3.1 ブラウザ・クライアントと Application Server Control コンソールをホストする Web サーバー間の通信の保護

デフォルトでは、Application Server Control のユーザー資格証明は、企業ネットワークまたはインターネットを介して、ブラウザから Web サーバーにクリア・テキストとして送信されます。そのため、セキュリティ攻撃を受けやすくなります。

ブラウザ・クライアントと、Application Server Control をホストする Web サーバーとの間の通信を保護するには、Application Server Control のユーザー資格証明を含め、Application Server Control の通信をすべて暗号化する必要があります。

セキュアな構成では、ブラウザ・クライアントは HTTPS 経由で管理 OC4J インスタンスに直接接続し、Application Server Control コンソールにアクセスします。この構成は、OC4J のスタンドアロン・インストール環境と Oracle Application Server 環境の両方に推奨されます。

次の手順では、HTTPS を使用して Application Server Control コンソール・クライアントを処理するための管理 OC4J インスタンスの構成方法を説明します。

作業 1: 管理 OC4J 用のキーストアおよび SSL 証明書の作成

管理 OC4J インスタンスのキーストアおよび SSL 証明書を作成するには、次の手順を実行します。

1. 管理 OC4J インスタンスを停止します。
2. keytool 実行可能ファイルを使用して、RSA 秘密鍵と公開鍵のペアが含まれるキーストアを作成します。これにより、OC4J がブラウザ・クライアントとのセキュアな HTTP 通信に使用できる SSL 証明書が作成されます。keytool 実行可能ファイルは、*ORACLE_HOME*/jdk/bin ディレクトリにあります。次のコマンドを使用します。

   ```
   keytool -genkey -keyalg "RSA" -keystore keystore -storepass passwd -validity days
   ```

   key パスワードの入力を求められたら、別のパスワードを入力せずに、[Enter] を押します。key パスワードは、生成した鍵ペアの秘密鍵の保護に使用されます。SSL を正しく機能させるには、キーストアと同じパスワードを使用する必要があります。

   関連項目：

   - 『Oracle Containers for J2EE セキュリティ・ガイド』の OC4J および Oracle HTTP Server での鍵と証明書の使用に関する項
   - 次の Sun 社の Web サイトにある、JSSE keytool コマンドの説明

     http://java.sun.com/j2se/1.3/docs/tooldocs/win32/keytool.html

作業 2: セキュアでない Web サイトからの ascontrol アプリケーションのバインド解除

デフォルトのセキュアでない Web サイトから ascontrol Web アプリケーションのバインドを解除するには、次の手順を実行します。

1. Application Server Control コンソール（ascontrol）の Web モジュールがバインドされている Web サイトの構成ファイルを編集します。デフォルトでは、このファイルは次の場所にあります。

   (UNIX) *ORACLE_HOME*/j2ee/*Admin_OC4J_instance_name*/config/default-web-site.xml
   (Windows) *ORACLE_HOME*¥j2ee¥*Admin_OC4J_instance_name*¥config¥default-web-site.xml

2. ascontrol アプリケーションをバインドしている <web-app> 要素を削除します。たとえば、次の行を削除します。

   ```
   <web-app application="ascontrol" name="ascontrol" root="/em" load-on-startup="true"
   ohs-routing="true" />
   ```

3. ファイルを保存して閉じます。

作業 3: ascontrol アプリケーション用の新しい HTTPS Web サイトの作成

Application Server Control（ascontrol）アプリケーションに新しい Web サイトを作成します。これには、HTTPS を使用する管理 OC4J インスタンスに新しい構成ファイルを作成します。次の手順に従います。

1. *ORACLE_HOME*/j2ee/*Admin_OC4J_instance_name*/config ディレクトリにある既存の *-web-site.xml ファイルをコピーして、新しい Web サイトを作成します。たとえば、default-web-site.xml を ascontrol-web-site.xml にコピーします。
2. 新しく作成した ascontrol-web-site.xml ファイルの <web-site> 要素に、次の変更を行います。
   - display-name 属性を変更して、Web サイトの表示名を ASControl Secure HTTP Web Site に変更します。
   - Web サイトが HTTPS を使用するように構成します。これには、protocol 属性を http に、secure 属性を true に設定します。

- ブラウザ・クライアントが Application Server Control コンソールの Web サイトへのアクセスに使用するポートを構成します。これには、port 属性に新しいポート番号を設定します。たとえば、port を 1156 に設定します。
- <ssl-config> 要素と、必要な keystore および keystore-password プロパティを追加して、前述の作業で作成したキーストアが参照されるようにします。
- <access-log> 要素の path 属性を、新しい Web サイトのアクセス・ログを保存する新しいログ・ファイルを指すように変更します。

3. 次の操作を行って、ascontrol の Web モジュールをこの Web サイトにバインドします。
   - <web-site> 要素内の <default-web-app> 要素の application 属性および name 属性を、ascontrol に設定します。
   - <default-web-app> 要素の root 属性を、"/ " に設定します。
   - <web-site> 要素内の他の <web-app> 要素をすべて削除します。

次の例は、ascontrol Web アプリケーション専用の Web サイトを定義する、ascontrol-web-site.xml という名前の Web サイト構成ファイルの抜粋です。

```
<web-site xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="http://xmlns.oracle.com/oracleas/schema/web-site-10_
0.xsd"
          port="1156" protocol="http" secure="true"
          display-name="ASControl Secure HTTP Web Site""
                      schema-major-version="10" schema-minor-version="0" >

    <default-web-app application="ascontrol" name="ascontrol" root="/ " />
    <access-log path="../log/ascontrol-web-access.log" split="day" />
    <ssl-config keystore="private/OracleAS_2/jdk/bin/mykeystore"
                keystore-password="welcome"/>
</web-site>
```

keystore 属性の値は、絶対パスまたは *-web-site.xml ファイルの位置からの相対パスです。

この例の場合、Application Server Control コンソールのユーザーは、次の URL にアクセスすることで、このコンソールにアクセスします。

```
https://hostname:1156
```

作業 4: 新しい ascontrol HTTPS Web サイトの登録

管理 OC4J インスタンスに新しい Web サイトを登録します。

1. *ORACLE_HOME*/j2ee/*Admin_OC4J_instance_name*/config ディレクトリで、server.xml ファイルを探します。
2. <application-server> 要素に、新しい ascontrol-web-site.xml ファイルを指す <web-site> 要素を追加します（path 属性は、絶対パスまたは server.xml ファイルの位置からの相対パスです）。たとえば、次のように指定します。

   ```
   <web-site path="./ascontrol-web-site.xml" />
   ```

3. 管理 OC4J インスタンスがクラスタ環境に属している場合は、次のファイルを変更して、新しい Web サイトを OPMN に登録します。

   (UNIX) *ORACLE_HOME*/opmn/conf/opmn.xml
   (Windows) *ORACLE_HOME*¥opmn¥conf¥opmn.xml

管理 OC4J の <ias-component> 要素（ias-component ID は OC4J、process-type ID は管理 OC4J 名と同じ）を探します。管理 OC4J セクションに、新しい Web サイトの新しい <port> 要素を追加します。たとえば、次のように指定します。

```
<ias-instance id="yellow.stadm21.ora.com" name="yellow.stadm21.ora.com">
 . . .
        <ias-component id="OC4J">
           <process-type id="home" module-id="OC4J" status="enabled">
 . . .
              <port id="default-web-site" range="8989" protocol="http"/>
              <port id="secure-web-site" range="1156" protocol="https"/>
              <port id="rmi" range="12401-12500"/>
              <port id="jms" range="12601-12700"/>
              <process-set id="default_group" numprocs="1"/>
 . . .
           </process-type>
        </ias-component>
```

この例では、Oracle Application Server インスタンスの名前が yellow.stadm21.ora.com、管理 OC4J インスタンスの名前が home になっています。

作業 5: 管理 OC4J インスタンスの起動

Oracle Application Server 環境で、新しい opmn.xml ファイルで OPMN を再構成します。これには、opmn.xml ファイルをリロードし、管理 OC4J インスタンスを起動します。次のコマンドを使用します。

- UNIX の場合 :

```
ORACLE_HOME/opmn/bin/opmnctl reload
ORACLE_HOME/opmn/bin/opmnctl startproc ias-component=OC4J
```

- Windows の場合 :

```
ORACLE_HOME¥opmn¥bin¥opmnctl reload
ORACLE_HOME¥opmn¥bin¥opmnctl startproc ias-component=OC4J
```

関連項目 :

- 『Oracle Containers for J2EE セキュリティ・ガイド』の Oracle Application Server の OC4J での SSL の使用に関する項
- 『Oracle Containers for J2EE セキュリティ・ガイド』のスタンドアロン OC4J での SSL の使用に関する項

A.3.2 Oracle Application Server のコンポーネント間の通信の保護

環境によっては、Oracle Application Server のコンポーネント間の通信を保護したほうがよい場合があります。それぞれの通信リンクは互いに依存していないため、保護するリンクと保護しないリンクを柔軟に決めることができます。次の方法を使用できます。

- 管理 OC4J インスタンスとリモート OC4J インスタンス間の通信を暗号化します（したがって、リモート OC4J の oc4jadmin パスワードも暗号化します）。
- 信頼できる Oracle Application Server インスタンスのみがクラスタのメンバーとなるように OPMN を保護します。

リモート OC4J インスタンスとは、Application Server Control によってリモート管理される OC4J インスタンスです。リモート OC4J インスタンスは、管理 OC4J インスタンスと同じ Oracle ホーム、別の Oracle ホーム、同じホストまたは異なるホストに配置されている可能性があります。

次の各項では、これらの手順について説明します。

- 管理 OC4J インスタンスとリモート OC4J インスタンス間の通信の保護
- Oracle Application Server クラスタの OPMN 通信の保護

A.3.2.1 管理 OC4J インスタンスとリモート OC4J インスタンス間の通信の保護

Oracle Application Server 環境では、Application Server Control を使用して管理 OC4J インスタンス以外の OC4J インスタンスを管理する場合、Remote Method Invocation（RMI）プロトコルによってリモート OC4J インスタンスとの JMX 接続が確立されます。Application Server Control は、リモート OC4J への JMX 接続を確立する際に、リモート OC4J の `oc4jadmin` ユーザー資格証明を送信することで、自身を認証します。デフォルトでは、この通信はクリア・テキスト形式で行われます。

管理 OC4J インスタンスとリモート OC4J インスタンス間の通信を保護するには、セキュアな Remote Method Invocation（ORMIS）プロトコルを使用します。

次の手順では、管理 OC4J インスタンス、および Application Server Control で管理する各 OC4J インスタンスに対して、RMIS を有効にするために実行する作業を示します。

この手順が必要なのは、Oracle Universal Installer と Oracle Application Server インストール手順を使用してインストールされた管理対象の Oracle Application Server 環境のみです。

> **関連項目：** デプロイおよび管理用に ORMI 接続を保護する方法、およびスタンドアロン環境で ORMIS を構成する方法については、『Oracle Containers for J2EE セキュリティ・ガイド』を参照してください。

作業 1: RMIS ポートによる各 OC4J インスタンスの構成

管理 OC4J インスタンスと、Application Server Control コンソールで管理する各リモート OC4J インスタンスに、セキュアな RMI ポートを構成します。

1. `keytool` 実行可能ファイルを使用して、RSA 秘密鍵と公開鍵のペアが含まれるキーストアを作成します。これにより、OC4J がブラウザ・クライアントとのセキュアな HTTP 通信に使用できる SSL 証明書が作成されます。`keytool` 実行可能ファイルは、*ORACLE_HOME*`/jdk/bin` ディレクトリにあります。次のコマンドを使用します。

   ```
   keytool -genkey -keyalg "RSA" -keystore keystore -storepass passwd -validity days
   ```

   `key` パスワードの入力を求められたら、別のパスワードを入力せずに、[Enter] を押します。`key` パスワードは、生成した鍵ペアの秘密鍵の保護に使用されます。SSL を正しく機能させるには、キーストアと同じパスワードを使用する必要があります。

   関連項目：

   - 『Oracle Containers for J2EE セキュリティ・ガイド』の OC4J および Oracle HTTP Server での鍵と証明書の使用に関する項
   - 次の Sun 社の Web サイトにある、JSSE `keytool` コマンドの説明

     `http://java.sun.com/j2se/1.3/docs/tooldocs/win32/keytool.html`

2. OC4J インスタンスの `rmi.xml` 構成ファイルを探します。

   このファイルは、通常、次の場所にありますが、OC4J インスタンスの `server.xml` ファイルで `<rmi-config>` 要素の値をチェックすることにより場所を確認できます。

   ```
   (UNIX) ORACLE_HOME/j2ee/instance_name/config/rmi.xml
   (Windows) ORACLE_HOME¥j2ee¥instance_name¥config¥rmi.xml
   ```

3. テキスト・エディタで `rmi.xml` ファイルを開き、`<ssl-config>` 要素をファイルに追加します。

4. `<ssl-config>` 要素を使用して、ステップ 1 で作成したキーストアのパスとキーストアのパスワードを指定します。たとえば、次のように指定します。

   ```
   <ssl-config keystore="path_to_keystore" keystore-password="keystore_pwd" />
   ```

5. <rmi-server> 要素の ssl-port 属性を使用して、SSL リスナー・ポートを指定します。たとえば、次のように指定します。

```
<rmi-server ... port="23791" ssl-port="23943" ... >
```

作業 2: 各リモート OC4J インスタンスの SSL 証明書の管理 OC4J インスタンスへの配布

各リモート OC4J インスタンスの SSL 証明書を管理 OC4J インスタンスに配布する必要があります。これには、各リモート OC4J インスタンスに、管理 OC4J のキーストアで信頼されている証明局によって署名された SSL 証明書を使用させるか、または各リモート OC4J インスタンスの SSL 証明書を管理 OC4J のキーストアにインポートします。

各リモート OC4J インスタンスの SSL 証明書を管理 OC4J のキーストアにインポートするには、各リモート OC4J インスタンスで次の手順を実行します。

1. リモート OC4J の Oracle ホームから、keytool コマンドを使用して、RSA 公開鍵が含まれる OC4J の SSL 証明書をエクスポートします。これにより、管理 OC4J からアクセス可能なファイルに証明書が配置されます。

```
keytool -export -file cert_file_name -keystore keystore_file_name
```

2. 管理 OC4J の Oracle ホームから次のコマンドを実行して、OC4J の SSL 証明書を管理 OC4J のキーストアにインポートします。

```
keytool -import -file cert_file_name -keystore keystore_file_name
```

作業 3: RMIS を有効にするための OPMN の構成

環境内の OC4J インスタンスをホストする各 Oracle Application Server インスタンスで、次の手順を実行します。

1. Oracle ホームで次の構成ファイルを探します。

```
(UNIX) ORACLE_HOME/opmn/conf/opmn.xml
(Windows) ORACLE_HOME¥opmn¥conf¥opmn.xml
```

2. テキスト・エディタで opmn.xml ファイルを開き、このファイルに定義されている OC4J インスタンスごとに、RMIS プロトコル用の新しい <port> 要素を追加します。

```
<port id="rmis" range="12701-12800"/>
```

作業 4: セキュアな RMIS 接続ポリシーに準拠するための管理 OC4J インスタンスの構成

1. 管理 OC4J がインストールされている Oracle ホームで、OPMN 構成ファイルを探します。

```
(UNIX) ORACLE_HOME/opmn/conf/opmn.xml
(Windows) ORACLE_HOME¥opmn¥conf¥opmn.xml
```

2. 管理 OC4J の <ias-component> 要素（ias-component ID は OC4J、process-type ID は管理 OC4J 名と同じ）を探します。opmn.xml ファイルの、管理 OC4J に対する java-options の開始パラメータに、次のプロパティを追加します。

```
oracle.oc4j.jmx.internal.connection.protocol
```

   Application Server Control は、このプロパティを使用して、リモート OC4J インスタンスとの通信にセキュアな RMI プロトコルを使用するタイミングを決定します。

表 A-1 に、環境に適用するセキュリティ・レベルに応じてこのプロパティに指定できる値を示します。

次の例は、RMIS プロパティが RMIS に設定された管理 OC4J の、<ias-component> 要素の一般的な構成を示しています。

```
<ias-component id="OC4J">
  <process-type id="home" module-id="OC4J" status="enabled">
    <module-data>
       <category id="start-parameters">
          <data id="java-options" value="-server
            -Doracle.oc4j.jmx.internal.connection.protocol=RMIS
```

```
            -Djava.security.policy=$ORACLE_HOME/j2ee/home/config/java2.policy
            -Djava.awt.headless=true -Dhttp.webdir.enable=false"/>
        </category>
    </module-data>
  </process-type>
</ias-component>
```

この例では、管理 OC4J の名前が home になっています。

すべての OC4J インスタンスとアプリケーションの管理において、セキュアな接続を維持する場合は、管理対象の各 OC4J インスタンスの rmi.xml ファイルに、<ssl-config> 要素を追加する必要があります。そうしないと、管理 OC4J インスタンスの opmn.xml ファイルに設定されている接続プロトコル・プロパティの値によっては、Application Server Control から OC4J インスタンスへの管理接続ができないか、セキュアでない RMI プロトコルが使用されます。

表 A-1 jmx.internal.connection.protocol プロパティの設定可能な値

プロパティ値	説明
RMIS_RMI	可能な場合は RMIS を使用し、そうでない場合は RMI を使用します。 これは、RMI 接続プロトコルが opmn.xml ファイルに見つからない場合のデフォルト設定です。
RMI_RMIS	可能な場合は RMI を使用し、そうでない場合は RMIS を使用します。
RMIS	RMIS を使用します。RMIS が使用できない場合は、接続エラーが報告されます。
RMI	RMI を使用します。RMI が使用できない場合は、接続エラーが報告されます。

詳細は、『Oracle Containers for J2EE セキュリティ・ガイド』の OC4J での ORMIS の有効化に関する項を参照してください。

A.3.2.2 Oracle Application Server クラスタの OPMN 通信の保護

環境にクラスタ・トポロジが使用されている場合は、信頼できる Oracle Application Server インスタンスのみがクラスタのメンバーとなるように、クラスタを保護する必要があります。そうしないと、悪意のある Oracle Application Server インスタンスがクラスタのメンバーとなり、「クラスタ・トポロジ」ページのクラスタ・コンポーネントのプロセスが制御される可能性があります。

次の各項で詳細を説明します。

- OPMN の保護とその目的
- OPMN SSL 接続での共通 Wallet の配布によるクラスタの保護

A.3.2.2.1 OPMN の保護とその目的 インストール時に、OPMN は、デフォルトの SSL 証明書が含まれるデフォルトの Wallet を使用するように構成されます。デフォルトの Wallet は、各 Oracle Application Server の Oracle ホームにある次のディレクトリに保存されます。

(UNIX) *ORACLE_HOME*/opmn/conf/ssl.wlt/default
(Windows) *ORACLE_HOME*¥opmn¥conf¥ssl.wlt¥default

OPMN を効果的に保護するには、このデフォルトの Wallet を有効な一意の証明書が含まれる新しい Wallet に置き換える必要があります。この作業は、クラスタの各 Oracle Application Server インスタンスで実行する必要があります。

また、OPMN の Wallet と証明書を構成する場合は、相互認証（サーバーとクライアント認証モードともいう）を使用する必要があります。

相互認証モードでは、OPMN クライアントと OPMN サーバーは両方とも、接続を行う前に X.509 証明書を使用して相互に認証します。まず、クライアントが証明書を検証してサーバーを認証します。それに応答し、サーバーも、信頼性を立証するための証明書を送信するようクライアントに要求します。

クラスタ内の各 OPMN でデフォルトの Wallet を置き換えず、クラスタのすべてのインスタンスで相互認証を使用する場合、デフォルトの Wallet および証明書を使用している Oracle Application Server インスタンスはいずれもクラスタのメンバーになることができます。クラスタのメンバーとなったら、そのインスタンスの Application Server Control で、「クラスタ・トポロジ」ページからクラスタのコンポーネントを起動および停止することが可能になります。

A.3.2.2.2 OPMN SSL 接続での共通 Wallet の配布によるクラスタの保護 クラスタ内の Oracle Application Server インスタンス間の OPMN 接続を最も簡単に保護するには、新しい Wallet を作成し、それをクラスタ内の各 Oracle Application Server インスタンスに配布します。

次の例では、自己署名（ルート）証明書を使用して Oracle Application Server クラスタを保護します。より複雑な環境や本番環境では、認証局から取得した証明書を使用できます。

次の例で、自己署名証明書を使用した相互認証の設定方法を学習します。

1. `orapki` ユーティリティを使用して、クラスタ・トポロジの最初の Oracle Application Server インスタンスに新しい Wallet と新しい自己署名（ルート）証明書を作成します。

   関連項目： 第 11.2 項「orapki ユーティリティによる証明書検証と CRL 管理の実行」

2. そのインスタンスの OPMN 構成ファイル（`opmn.xml`）で、`<ssl>` 要素が新しく作成した Wallet を指すように変更します。たとえば、次のように指定します。

   ```
   <ssl enabled="true"
        wallet-file="$ORACLE_HOME/opmn/conf/ssl.wlt/new_wallet"
        wallet-password="welcome2"/>
   ```

3. 新しい Wallet（新しく作成されたルート証明書を含む）をクラスタ内の各 Oracle Application Server インスタンスにコピーして、ステップ 2 の説明にあるように、各インスタンスの `opmn.xml` ファイルを更新します。

4. 次のコマンドを使用して、クラスタ内の各 Oracle Application Server インスタンスを再起動します。

   UNIX の場合 :

   ```
   ORACLE_HOME/opmn/bin/opmnctl stopall
   ORACLE_HOME/opmn/bin/opmnctl startall
   ```

   Windows の場合 :

   ```
   ORACLE_HOME¥opmn¥bin¥opmnctl stopall
   ORACLE_HOME¥opmn¥bin¥opmnctl startall
   ```

5. Application Server Control コンソールを使用して、クラスタが正しく構成されていることを確認します。

   a. Application Server Control コンソールにログインして、「クラスタ・トポロジ」ページの内容を確認します。

      このページの「メンバー」セクションに、クラスタ内の各 Oracle Application Server インスタンスが表示されている必要があります。

   b. このページの「管理」セクションにスクロールし、「**トポロジ・ネットワーク構成**」をクリックします。

   c. 「トポロジ・ネットワーク構成」ページの上方にある「**表示方法**」メニューで、クラスタ内の Oracle Application Server インスタンスを 1 つ選択します。

   d. 「トポロジ・ネットワーク構成」ページの「SSL」セクションにスクロールし、SSL が有効であり、この手順の前半に作成した新しい Wallet がそのインスタンスで使用されていることを確認します。

OPMN およびセキュリティの詳細は、『Oracle Process Manager and Notification Server 管理者ガイド』を参照してください。

A.4 Application Server Control のロギングの構成

Application Server Control には独自のログ・ファイルのセットが用意されており、これらは構成ファイルを変更することで構成できます。ロギングの構成方法は、Oracle Diagnostic Logging（ODL）を有効にするかどうかによって異なります。

次の各項で詳細を説明します。

- Application Server Control のログ・ファイルに対する ODL の有効化と構成
- ODL が無効である場合のロギング・プロパティの構成
- ログ・ファイルの検索時に取得するエントリ数の制御

A.4.1 Application Server Control のログ・ファイルに対する ODL の有効化と構成

デフォルトでは、Application Server Control で生成されるログ・ファイルは、テキスト形式で保存されます。ただし、Oracle Diagnostic Logging（ODL）形式でログ・ファイルを保存するように、Application Server Control を構成できます。

Application Server Control のログ・ファイルに対して ODL を有効にすると、ロギング情報および診断情報が XML 形式で保存され、それぞれのログ・メッセージが ODL 規格に準拠するようにフォーマットされます。

関連項目： 第 5 章「ログ・ファイルの管理」

デフォルトでは、アプリケーション・サーバーのホーム・ディレクトリにある次のログ・ファイルに、Application Server Control によって情報およびエラー・メッセージが記録されます。

(UNIX) *ORACLE_HOME*/j2ee/home/log/ascontrol.log
(Windows) *ORACLE_HOME*¥j2ee¥home¥log¥ascontrol.log

第 A.4.1.1 項の手順を実行した後、Application Server Control では、前述のファイルではなく次のファイルにログ情報およびエラー・メッセージが記録されます。このファイルのデータは、ODL 規格に従ってフォーマットされます。

(UNIX) *ORACLE_HOME*/sysman/log/log.xml
(Windows) *ORACLE_HOME*¥sysman¥log¥log.xml

詳細は、次の項を参照してください。

- 第 A.4.1.1 項「ODL を有効にするための Application Server Control ロギング・プロパティの構成」
- 第 A.4.1.2 項「Application Server Control の ODL ロギング・プロパティについて」
- 第 A.4.2 項「ODL が無効である場合のロギング・プロパティの構成」

A.4.1.1 ODL を有効にするための Application Server Control ロギング・プロパティの構成

ODL をサポートするように Application Server Control を構成する手順は次のとおりです。

1. Oracle Application Server の Oracle ホームにある次のディレクトリにナビゲートします。

   (UNIX) *ORACLE_HOME*/j2ee/home/applications/ascontrol/ascontrol/WEB-INF/config
   (Windows) *ORACLE_HOME*¥j2ee¥home¥applications¥ascontrol¥ascontrol¥WEB-INF¥config

2. テキスト・エディタを使用して、config ディレクトリにある次の構成ファイルを編集します。

   ascontrollogging.properties

3. ファイルに記述されている指示に従って、デフォルトのプロパティを、デフォルトでコメントアウトされているプロパティに置き換えます。

   例 A-1 に、emiasconsolelogging.properties ファイル内にある、Application Server Control のログ・ファイルに対して ODL を有効にするプロパティを示します。

4. ascontrollogging.properties ファイルを保存して閉じます。

5. Application Server Control を再起動します。

例 A-1 Application Server Control コンソールの ODL ロギング・プロパティ

```
# To support the ODL log appender, replace the lines above
# with the following and restart EM.  The resulting ODL log files
# will be read by the Log Loader and written to the Log Repository.
#
# log4j.appender.emiaslogAppender=oracle.core.ojdl.log4j.OracleAppender
# log4j.appender.emiaslogAppender.ComponentId=EM
# log4j.appender.emiaslogAppender.LogDirectory=/private/shiphomes/m21_infra/sysman/log
# log4j.appender.emiaslogAppender.MaxSize=20000000
# log4j.appender.emiaslogAppender.MaxSegmentSize=5000000
```

A.4.1.2 Application Server Control の ODL ロギング・プロパティについて

表 A-2 は、emiasconsolelogging.properties ファイルで使用できる Oracle Diagnostic Logging（ODL）ロギング・プロパティとその説明です。

表 A-2 Oracle Diagnostic Logging（ODL）プロパティ

プロパティ	説明
log4j.appender.emiaslogAppender.LogDirectory	log.xml ファイルの保存先ディレクトリを指定します。
log4j.appender.emiaslogAppender.MaxSize	log.xml ファイルおよびロギング・ロールオーバー・ファイルで使用する最大ディスク領域を指定します。
log4j.appender.emiaslogAppender.MaxSegmentSize	log.xml ファイルの最大サイズを指定します。log.xml ファイルのサイズがこの上限値に達すると、ロールオーバー・ファイルが作成されます。

ODL を有効にすることによって作成される log.xml ファイルのサイズは、情報が書き込まれるに従って大きくなっていきます。このファイルは、表 A-2 で説明した MaxSegmentSize プロパティに指定された最大サイズに達するように設計されています。ファイルが事前定義済の最大サイズに達すると、Application Server Control では、ロギング情報およびトレース情報が新しいファイル名に変更（またはロール）され、新しいログ・ファイルまたはトレース・ファイルへの記録が開始されます。このプロセスにより、ログ・ファイルのサイズが過大になることはなくなります。

重要なログ情報にアクセスできるようにするため、Application Server Control では、ログ・ファイルおよびそのロールオーバー・ファイルが、例 A-1 で示した MaxSize プロパティに指定された最大ディスク領域に達するまで、log.xml ファイルをロールオーバーします。ログ・ファイルおよびそのロールオーバー・ファイルがこの上限値に達すると、Application Server Control では、一番古いロールオーバー・ファイルが削除されます。

その結果、ログ・ディレクトリに複数のログ・ファイルが存在することもよくあります。次の例は、現在ログ・ディレクトリ内にある Application Server Control の 3 つのロールオーバー・ファイルおよびログ・ファイルを示しています。

```
log.xml
log1.xml
log2.xml
log3.xml
```

A.4.2 ODL が無効である場合のロギング・プロパティの構成

ODL が無効である場合も、ascontrollogging.properties ファイルを変更することで、Application Server Control のロギング・プロパティを構成できます。

1. Oracle Application Server のホーム・ディレクトリにある次のディレクトリにナビゲートします。

   (UNIX) *ORACLE_HOME*/j2ee/home/applications/ascontrol/ascontrol/WEB-INF/config/
   (Windows) *ORACLE_HOME*¥j2ee¥home¥applications¥ascontrol¥ascontrol¥WEB-INF¥config¥

2. テキスト・エディタを使用して、config ディレクトリにある次の構成ファイルを編集します。

   ascontrollogging.properties

3. 表 A-3 に一覧されているロギング・プロパティを変更します。
4. ascontrollogging.properties ファイルを保存して閉じます。
5. Application Server Control を再起動します。

表 A-3 ODL が無効である場合のロギング・プロパティ

プロパティ	説明
log4j.appender.ascontrollogAppender.File	Application Server Control（ascontrol）アプリケーションの位置と名前です。
log4j.appender.ascontrollogAppender.MaxFileSize	ascontrol アプリケーションのログ・ファイルおよびそのロールオーバー・ログ・ファイルで使用する最大ディスク領域を指定します。
log4j.appender.ascontrollogAppender.MaxBackupIndex	Application Server Control が一番古いロールオーバー・ログ・ファイルを削除するまでに、ログ・ファイルを新しいファイル名にロールオーバーする回数を指定します。

A.4.3 ログ・ファイルの検索時に取得するエントリ数の制御

アプリケーション・エラーやパフォーマンスの問題の診断を支援するために、Enterprise Manager には Application Server Control コンソールから OC4J ログ・ファイルを検索するメカニズムが用意されています。

現在の Oracle Application Server インスタンスに関連するすべての OC4J インスタンスのログ・ファイルだけでなく、インスタンスにデプロイされたアプリケーションに関連するログ・ファイルも検索できます。

関連項目：

- 第 5.1 項「Application Server Control でのログ・ファイルのリストと表示」
- Application Server Control のオンライン・ヘルプの OC4J ログ・ファイルの検索に関する項

Application Server Control コンソールの「ログの検索」ページで一連の検索基準を入力すると、Enterprise Manager によって、その基準に一致するログ・ファイル・エントリのリストが返されます。デフォルトでは、詳細情報にブラウズまたはフィルタ可能なログ・ファイル・エントリが最大で 5,000 個返されます。

ログ・ファイル検索時に 5,000 個を超えるエントリを取得したり、返されるエントリが 5,000 個未満になるように検索を制限するには、次の手順を実行します。

1. テキスト・エディタを使用して、emiaslogviewer.xml 構成ファイルを開きます。これは、管理 OC4J の Oracle ホーム内の次のディレクトリにあります。

   (UNIX) *ORACLE_HOME*/j2ee/home/applications/ascontrol/ascontrol/WEB-INF/config
   (Windows) *ORACLE_HOME*¥j2ee¥home¥applications¥ascontrol¥ascontrol¥WEB-INF¥config

2. emiaslogviewer.xml ファイル内の次の要素を変更します。

```
<LogViewerConfig maxEntries="5000">
```

A.5 Enterprise Manager のアクセシビリティ・モードの有効化

次の項では、Enterprise Manager をアクセシビリティ・モードで実行することの利点、およびアクセシビリティ・モードを有効にするための手順について説明します。

- HTML ページに対するアクセスのしやすさの強化
- Enterprise Manager のグラフのテキストによる説明
- uix-config.xml ファイルの変更によるアクセシビリティ・モードの有効化

A.5.1 HTML ページに対するアクセスのしやすさの強化

Enterprise Manager では、一部のユーザー操作のレスポンスを向上させるユーザー・インタフェース開発テクノロジを利用しています。たとえば、表中の新しいレコード・セットへ移動するとき、Enterprise Manager では HTML ページ全体の再表示が行われません。

ただし、このパフォーマンス強化テクノロジは、一般にスクリーン・リーダーではサポートされません。アクセシビリティ・モードを有効にしている場合は、この機能を無効にします。それにより、Enterprise Manager の HTML ページは、身体に障害のあるユーザーにもアクセスしやすくなります。

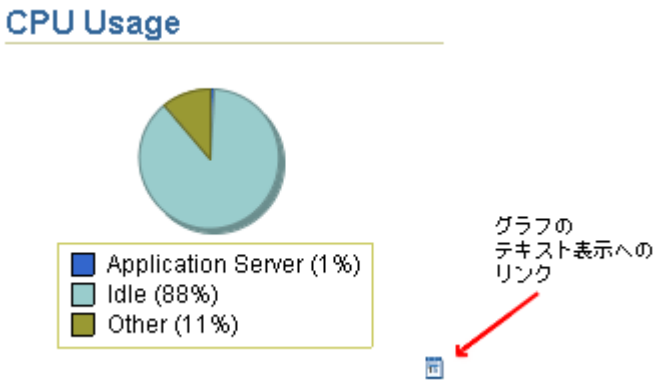
A.5.2 Enterprise Manager のグラフのテキストによる説明

Enterprise Manager では、パフォーマンス・データの表示にグラフが使用されます。グラフによるデータ表示は、ほとんどのユーザーにとっては利用価値があり、トレンドを明確にして、パフォーマンス・メトリックの最大値および最小値が特定しやすくなります。

しかし、グラフに表示される情報をスクリーン・リーダーで読み取って伝達することはできません。この問題を解消するため、各パフォーマンス・グラフをすべてテキストで表示するように Enterprise Manager を構成できます。アクセシビリティ・モードを有効にすると、Enterprise Manager では、各グラフの小さなアイコンが表示され、テキスト表示へのドリルダウン・リンクとして使用できるようになります。

アクセシビリティ・モードを有効にした場合に各グラフの下に表示されるアイコンの例を図 A-2 に示します。

図 A-2 グラフのテキスト表示を表すアイコン



A.5.3 uix-config.xml ファイルの変更によるアクセシビリティ・モードの有効化

uix-config.xml 構成ファイルを変更するには、次の手順を実行します。

1. 次に示す Oracle Application Server のホーム・ディレクトリで、uix-config.xml 構成ファイルを検索します。

   (UNIX) *ORACLE_HOME*/j2ee/home/applications/ascontrol/WEB-INF
   (Windows) *ORACLE_HOME*¥j2ee¥home¥applications¥ascontrol¥WEB-INF

2. テキスト・エディタを使用して uix-config.xml ファイルを開き、次のエントリを検索します。

```
<!-- An alternate configuration that disables accessibility features  -->
<default-configuration>
  <accessibility-mode>inaccessible</accessibility-mode>
</default-configuration>
```

3. accessibility-mode プロパティの値を、inaccessible から accessible に変更します。
4. ファイルを保存して閉じます。
5. Application Server Control コンソールを再起動します。

A.6 アクティブな Application Server Control の管理

次の各項では、アクティブな Application Server Control の管理および構成について詳細に学習します。

- アクティブな Application Server Control について
- アクティブな Application Server Control の管理のベスト・プラクティス
- ascontrol のインスタンスの停止とアプリケーション起動の回避
- 新しいアクティブな Application Server Control の特定と構成
- HTTP を介した管理 OC4J への直接アクセス
- 同じ OC4J インスタンス内の別の Web サイトへの Application Server Control の公開

A.6.1 アクティブな Application Server Control について

デフォルトでは、Oracle Application Server 10*g* リリース 3（10.1.3.2.0）クラスタの各 OC4J インスタンスには、ascontrol アプリケーションが含まれます。これは Application Server Control のインスタンスを表します。

ただし、クラスタ内では ascontrol アプリケーションを 1 つのみ稼動させる必要があります。このアプリケーションの他のインスタンスは、停止または無効にする必要があります。新しい OC4J インスタンスを作成すると、そのインスタンスには OC4J インスタンスにデプロイされる ascontrol アプリケーションが含まれます。ただしデフォルトでは、OC4J インスタンスの特性を定義する server.xml 構成ファイル内の設定により、OC4J インスタンスの起動時に ascontrol アプリケーションは自動的に起動されません。

> **関連項目：** 第 A.6.3 項「ascontrol のインスタンスの停止とアプリケーション起動の回避」

Application Server Control コンソールにログインすると、常に 1 つの ascontrol アプリケーションのみがクラスタの管理に使用されます。その ascontrol アプリケーションを、アクティブな Application Server Control と呼びます。

- アクティブな Application Server Control を特定するには、「クラスタ・トポロジ」ページにナビゲートして、「**すべてを開く**」をクリックし、クラスタのすべてのコンポーネントを表示してから、アクティブな Application Server Control アイコンで示される ascontrol アプリケーションを探します。

アクティブな Application Server Control をホストする OC4J インスタンスは、**管理 OC4J** と呼ばれます。

A.6.2 アクティブな Application Server Control の管理のベスト・プラクティス

表 A-4 に、アクティブな Application Server Control を管理する際に考慮する必要がある一連のベスト・プラクティスのガイドラインを示します。

表 A-4 アクティブな Application Server Control の管理のベスト・プラクティス

推奨事項	説明	詳細情報の参照先
OC4J インスタンスを 1 つ選択し、管理 OC4J として使用します。	このインスタンスをデプロイ済アプリケーションのホストに使用しないでください。このインスタンスは Application Server Control のホスト専用とし、アプリケーションのデプロイには別途 OC4J インスタンスを作成します。 Java SSO を使用している場合は、管理 OC4J を使用してアクティブな `javasso` アプリケーションをホストすることもできます。	第 6.1 項「OC4J インスタンスの追加と削除」 第 6.5 項「OC4J Java Single Sign-On を使用するためのインスタンスの構成」
管理 OC4J インスタンスにデプロイされていない `ascontrol` アプリケーションのインスタンスは停止または無効にします。	アクティブな Application Server Control のみ稼動させておく必要があります。	第 A.6.3 項「ascontrol のインスタンスの停止とアプリケーション起動の回避」
Application Server Control を分離して、独自の Web サイトや URL を使用します。	Oracle Application Server 管理者のみが Application Server Control コンソールにアクセスする必要があるので、コンソールのアクセスに個別の URL とポート番号を使用することは適切です。	第 A.6.5 項「HTTP を介した管理 OC4J への直接アクセス」 第 A.6.6 項「同じ OC4J インスタンス内の別の Web サイトへの Application Server Control の公開」

A.6.3 ascontrol のインスタンスの停止とアプリケーション起動の回避

クラスタ・トポロジ内で複数の `ascontrol` アプリケーションが稼動している場合、「クラスタ・トポロジ」ページの上部に次の警告メッセージが表示されます。

```
You have more than one instance of the Application Server Control application
running in this cluster. This is not a recommended configuration and could lead
to unexpected problems. Please stop the additional instances of Application
Server Control or disable routing to these instances.
```

この問題は、たとえば複数の Oracle Application Server インスタンスを別々にインストールし、それらを後から 1 つのクラスタ・トポロジにまとめた場合などに発生します。クラスタのメンバーになる前の各 Oracle Application Server インスタンスには、インスタンスの管理に使用する個別の Application Server Control があります。複数のインスタンスを 1 つのクラスタにまとめた後、クラスタ全体を管理するために必要となる Application Server Control は 1 つだけです。

この問題を解決するには、次の手順を実行します。

1. 「クラスタ・トポロジ」ページにナビゲートし、「**すべてを開く**」をクリックして、インスタンスにデプロイされているすべての OC4J インスタンスとアプリケーションを表示します。
2. アクティブな ascontrol アプリケーションを示す緑色のひし形アイコンの印が付いていない任意の `ascontrol` アプリケーションを選択します。

3. 「**停止**」をクリックします。

   この操作により、選択された非アクティブな ascontrol アプリケーションが停止されます。さらにクラスタ内の非アクティブな ascontrol を停止すると、Application Server Control では自動的に次の処理が実行されます。

   - 次回ホスト OC4J インスタンスが再起動するときに非アクティブな ascontrol アプリケーションを起動させないようにするため、その OC4J インスタンスの `server.xml` ファイルを変更します。具体的には、そのアプリケーションの `start` パラメータを `false` に設定します。
   - すべての Oracle HTTP Server インスタンスで非アクティブな ascontrol アプリケーションにリクエストをルーティングしないようにするため、その OC4J インスタンスの default-web-site.xml ファイルを変更します。具体的には、ascontrol アプリケーションの `ohs-routing` パラメータを `false` に設定します。

   その結果、後で非アクティブな ascontrol アプリケーションを起動する必要が発生した場合（たとえば、その Oracle Application Server インスタンスをクラスタから分離して個別に管理する場合など）、その構成設定を元の値に再設定する必要があります。

   詳細は、第 A.6.4 項「新しいアクティブな Application Server Control の特定と構成」を参照してください。

A.6.4 新しいアクティブな Application Server Control の特定と構成

状況に応じて、`ascontrol` アプリケーションの異なるインスタンスをアクティブな Application Server Control として特定する必要がある場合があります。たとえば、専用の管理 OC4J インスタンスを使用していない場合、個別のホストの個別の OC4J インスタンスで `ascontrol` アプリケーションが稼動するように環境を再構成できます。

同様に、新しい Oracle Application Server インスタンスをインストールして、インストール時にそれを管理 OC4J として特定しなかった場合、`ascontrol` アプリケーションを稼動せずに終了することも可能です。その場合は、いずれかの `ascontrol` アプリケーションを起動して、それをアクティブな Application Server Control として特定する必要があります。

いずれのケースも、次の手順を実行します。

1. クラスタにデプロイされている ascontrol アプリケーションの各インスタンスで、次の手順を実行します。

   a. テキスト・エディタを使用して、OC4J インスタンスの `server.xml` ファイルを開きます。

      UNIX の場合 :

      ORACLE_HOME/j2ee/*oc4j_instance_name*/config/server.xml

      Windows の場合 :

      ORACLE_HOME¥j2ee¥*oc4j_instance_name*¥config¥server.xml

      この例では、*oc4j_instance_name* は `ascontrol` アプリケーションがデプロイされている OC4J インスタンスの名前を示します。

   b. アクティブな `ascontrol` アプリケーションの `start` 引数が `true` に設定されていることを確認します。これにより、OC4J インスタンスの再起動時に必ずこのアプリケーションが自動的に起動されるようになります。

      ```
      <application name="ascontrol"
                   path="../../oc4j_instance_name/applications/ascontrol.ear"
                   parent="system"
                   start="true" />
      ```

      `ascontrol` アプリケーションのその他すべてのインスタンスは、start 引数が `false` に設定されていることを確認します。これにより、選択されたアプリケーションのインスタンスは、OC4J インスタンスの再起動時に自動的に起動されなくなります。

c. 変更を保存して server.xml ファイルを閉じます。

d. default-web-site.xml ファイルを開きます。これは、server.xml ファイルと同じディレクトリに格納されています。

e. アクティブな ascontrol アプリケーションの web-app 要素の ohs-routing 属性を true に変更します。

```
<web-app application="ascontrol"
         name="ascontrol"
         load-on-startup="true"
         root="/em"
         ohs-routing="true" />
```

ascontrol アプリケーションのその他すべてのインスタンスは、ohs-routing 属性が false に設定されていることを確認します。これにより、Oracle HTTP Server で HTTP リクエストが ascontrol アプリケーションの非アクティブなインスタンスにルーティングされなくなります。

f. 変更を保存して default-web-site.xml ファイルを閉じます。

2. クラスタ内の各 OC4J インスタンスで server.xml および default-web-site.xml ファイルを構成した後、次の手順を実行します。

a. 「クラスタ・トポロジ」ページから、現在アクティブな ascontrol アプリケーションのデプロイに使用されるインスタンス以外の、すべての OC4J インスタンスを停止して再起動します。

現在アクティブな ascontrol アプリケーションは、緑色のひし形アイコンで示されます。

b. 他の OC4J インスタンスを再起動した後に、現在アクティブな ascontrol アプリケーションをホストする管理 OC4J インスタンスを再起動します。

現在アクティブな ascontrol アプリケーションを再起動すると、Application Server Control コンソールをログアウトすることになります。

c. 新しいアクティブな ascontrol アプリケーションの URL を使用して、Application Server Control コンソールにログインします。

たとえば、次のように指定します。

http://*new_admin_oc4j_hostname*:*port*/em

A.6.5 HTTP を介した管理 OC4J への直接アクセス

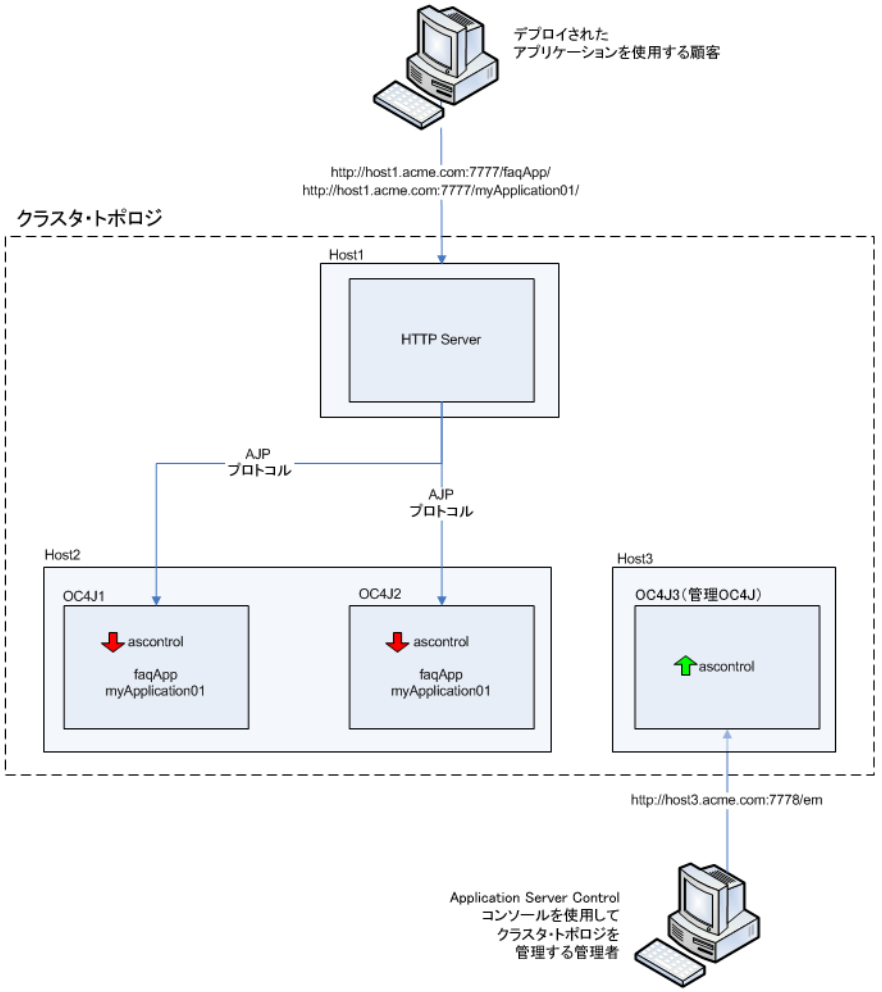
専用の管理 OC4J インスタンスを使用するメリットの 1 つに、管理 OC4J で独自の HTTP リスナーを使用するように容易に構成できる点が挙げられます。

たとえば、現在 Oracle HTTP Server を使用して管理 OC4J にアクセスしている場合は、このインスタンスで組込み OC4J HTTP リスナーを使用するように再構成できます。これにより、Oracle HTTP Server を経由せずに、HTTP プロトコルを使用して管理 OC4J に直接アクセスできるようになります。

ここで、Oracle Application Server 管理者は個別の URL とポート番号を使用して Application Server Control コンソールにアクセスできます。一方、他の OC4J インスタンスにデプロイしたアプリケーションには、アプリケーション・ユーザーはそのまま Oracle HTTP Server を使用してアクセスできます。

Application Server Control コンソールへのアクセス保護を強化するために、管理 OC4J を別のホストに配置して、そこからクラスタ・トポロジのコンポーネントを管理することもできます。図 A-3 はその構成例を示しています。

図 A-3 管理 OC4J HTTP リスナーを使用したクラスタ・トポロジの管理



OC4J インスタンスでフロントエンドの Oracle HTTP Server のかわりに独自の HTTP URL を使用するように構成するには、OC4J インスタンスのプロトコルを AJP から HTTP に変更して、HTTP ポートを指定します。手順は次のとおりです。

1. 「クラスタ・トポロジ」ページで、管理 OC4J インスタンスの名前をクリックします。
2. OC4J ホーム・ページから、「**管理**」をクリックします。
3. OC4J の「管理」ページで、表の「**サーバー・プロパティ**」行にあるタスク・アイコンをクリックします。
4. 「Web サイト」表の「**default-web-site**」行で、「**プロトコル**」ドロップダウン・メニューから「**http**」を選択して、「**ポート**」フィールドにポート番号またはポートの範囲を入力します。

   管理 OC4J インスタンスが Oracle HTTP Server と同じホスト上に存在する場合は、Oracle HTTP Server で使用されるポートとは異なるポートまたはポート範囲を使用してください。
5. 「**適用**」をクリックして変更内容を適用します。
6. 「クラスタ・トポロジ」ページにナビゲートして、管理 OC4J インスタンスを再起動します。

   管理 OC4J を再起動すると、次の URL を使用して Application Server Control コンソールにアクセスできるようになります。

   `http://admin_oc4j_hostname:port/em/`

A.6.6 同じ OC4J インスタンス内の別の Web サイトへの Application Server Control の公開

専用の管理 OC4J を使用しない場合は、管理者が Oracle HTTP Server を経由せずに個別の HTTP ポートを使用して直接 Application Server Control コンソールにアクセスするように、OC4J インスタンスを構成することもできます。

このケースでは、OC4J インスタンスに別の Web サイトを構成して、その Web サイトに既存の Application Server Control のバインディングを移行します。その後、その Web サイトを認識するように OPMN を構成します。既存の Web サイト（通常、default-web-site）は、デプロイ済アプリケーションのホストにそのまま使用できます。一方、Oracle Application Server 管理者は新しい Web サイトとポートを使用して Application Server Control コンソールにアクセスします。

次の手順を実行して、Application Server Control を独自の OC4J Web サイトに移行します。

1. *ORACLE_HOME*/j2ee/home/config/default-web-site.xml ファイルを、*ORACLE_HOME*/j2ee/home/config/ascontrol-web-site.xml（または任意のファイル名）にコピーします。

2. ascontrol-web-site.xml ファイルを編集して、既存の Web アプリケーションのバインディングをすべて削除します。Application Server Control コンソール・アプリケーションの <default-web-app> エントリおよび <web-app> エントリ（次の例の太字部分）のみ残します。これにより、ascontrol アプリケーションが Web サイトのルート・コンテキスト /em にマップされます。<web-site> 要素に protocol="http" および port="1810" が指定されており、display-name が一意の名前になっていることを確認してください。

```
<?xml version = '1.0' standalone = 'yes'?>
<web-site
   protocol="http"
   port="1810
   display-name="OC4J 10g (10.1.3) ASControl Web Site"

   <default-web-app application="default" name="defaultWebApp" root="/j2ee" />
   <web-app application="ascontrol" name="ascontrol" root="/em" />

   <!-- Access Log, where requests are logged to -->
   <access-log path="../log/default-web-access.log"/>
   <!-- Uncomment this if you want to use ODL logging capabilities
   <odl-access-log path="../log/default-web-access" max-file-size="1000"
max-directory-size="10000"/>
   -->
   <web-app application="bc4j" name="webapp" root="/webapp"
load-on-startup="false"/>
</web-site>
```

3. access-log path を、ascontrol の Web サイト用として一意のログ・ファイルを指定するように変更します。

4. *ORACLE_HOME*/j2ee/home/config/server.xml を編集して、次の例の太字部分のように、ascontrol-web-site.xml ファイルを指定する新しい <web-site> 要素を追加します。

```
<application-server ...>
...
        <web-site default="true" path="./default-web-site.xml" />
        <web-site default="false" path="./ascontrol-web-site.xml" />...
</application-server>
```

5. *ORACLE_HOME*/j2ee/home/config/default-web-site.xml を編集して、ascontrol アプリケーションの web-app バインディングを削除するか、コメントアウトします。

```
<web-site
   protocol="http"
   port="1810"
   display-name="OC4J 10g (10.1.3) ASControl Web Site"
...
<!--
   <web-app application="ascontrol" name="ascontrol" root="/em" / -->

</web-site>
```

6. OPMN が新しい ascontrol Web サイトのポート設定を認識できるよう、OPMN 構成を新しい ascontrol Web サイトで更新します。*ORACLE_HOME*/opmn/bin から、次のコマンドを発行します。

```
opmnctl config port update ias-component=OC4J process-type=home
 portid=ascontrol-web-site protocol="http" range=1810-1820
```

7. *ORACLE_HOME*/opmn/bin から次のコマンドを発行して、サーバーを再起動します。

```
opmnctl stopall
opmnctl startall
```

これで、Application Server Control は AppHost1:1810/em でアクセスできるようになり、Oracle HTTP Server から分離されました。ただし、デフォルトのアプリケーションと、デフォルトのアプリケーションの子としてデプロイされている他のアプリケーションは、そのまま Oracle HTTP Server を使用します。

B

Oracle Application Server のコマンドライン・ツール

表 B-1 は、Oracle Application Server で使用可能なコマンドライン・ツールについて、その説明と関連マニュアルを示しながらまとめます。

表 B-1 Oracle Application Server のコマンドライン・ツール

コマンド	Oracle ホームからのパス	説明
bkp_restore	（UNIX）backup_restore/bkp_restore.sh （Windows）backup_restore¥bkp_restore.bat	Oracle Application Server インスタンスをバックアップおよびリストアします。 **関連項目：**第 16 章
chgiphost	（UNIX）chgip/scripts/chgiphost.sh （Windows）chgip¥scripts¥chgiphost.bat	中間層インスタンス、Infrastructure または Identity Management インストールのホスト名、ドメイン名または IP アドレスを変更します。 **関連項目：**第 7.2.1 項
clone	（UNIX）clone/bin/clone.pl （Windows）clone¥bin¥clone.pl	Oracle Application Server インスタンスをクローニングします。 **関連項目：**第 9 章
createinstance	（UNIX）bin/createinstance.sh （Windows）bin¥createinstance.bat	新しい OC4J インスタンスを作成します。 **関連項目：**第 6.1 項
dmstool	（UNIX）bin/dmstool （Windows）bin¥dmstool.bat	パフォーマンス・メトリックを表示し、レポート作成の間隔を設定します。 **関連項目：**『Oracle Application Server パフォーマンス・ガイド』
jazn.jar	（UNIX）j2ee/home/jazn.jar （Windows）j2ee¥home¥jazn.jar	XML ベースと LDAP ベースの JAAS データを管理します。 **関連項目：**『Oracle Containers for J2EE セキュリティ・ガイド』
ojspc	（UNIX）bin/ojspc （Windows）bin¥ojspc.bat	JSP バック・プリコンパイラ。 **関連項目：**『Oracle Containers for J2EE JavaServer Pages 開発者ガイド』
opmnassociate	（UNIX）bin/opmnassociate （Windows）bin¥opmnassociate.cmd	OC4J インスタンスをクラスタに追加します。 **関連項目：**『Oracle Process Manager and Notification Server 管理者ガイド』
opmnctl	（UNIX）opmn/bin/opmnctl （Windows）opmn¥bin¥opmnctl.exe	OPMN によって管理されるプロセスの起動、停止およびステータスの取得を行います。 **関連項目：**『Oracle Process Manager and Notification Server 管理者ガイド』

表 B-1 Oracle Application Server のコマンドライン・ツール（続き）

コマンド	Oracle ホームからのパス	説明
orapki	（UNIX）bin/orapki.sh （Windows）bin¥orapki.bat	証明書失効リスト（CRL）、Wallet を管理します。 **関連項目：**第 11.2.1 項
prepare_clone	（UNIX）clone/bin/prepare_clone.pl （Windows）clone¥bin¥prepare_clone.pl	Oracle Application Server インスタンスのクローニングを準備します。 **関連項目：**第 9 章
removeinstance	（UNIX）bin/removeinstance.sh （Windows）bin¥removeinstance.bat	OC4J インスタンスを削除します。 **関連項目：**第 6.1 項

C

コンポーネントの URL

表 C-1 は、インストール後にコンポーネントへのアクセスに使用する URL とログイン ID を示しています。

表には、URL とともにデフォルト・ポートが示されています。環境によっては、コンポーネントで異なるポートが使用される場合があります。コンポーネントのポート番号を確認するには、次のコマンドを使用します。

```
(UNIX) ORACLE_HOME/opmn/bin/opmnctl status -l
(Windows) ORACLE_HOME¥opmn¥bin¥opmnctl status -l
```

表 C-1 コンポーネントの URL

コンポーネント	URL（およびデフォルト・ポート番号）	opmnctl status 出力のエントリ	ログインおよびパスワード
「ようこそ」ページ	UNIX の場合 : http://*host*:7777 Windows の場合 : http://*host*:80	HTTP_Server	該当なし
Oracle HTTP Server	UNIX の場合 : http://*host*:7777 Windows の場合 : http://*host*:80	HTTP_Server	該当なし
Application Server Control コンソール	UNIX の場合 : http://*host*:7777/em Windows の場合 : http://*host*:80/em	HTTP_Server	`oc4jadmin` インストール時に指定した `oc4jadmin` パスワードを使用します。
Oracle Content DB	UNIX の場合 : http://*host*:7777/em Windows の場合 : http://*host*:80/em	HTTP_Server	`oc4jadmin` インストール時に指定した `oc4jadmin` パスワードを使用します。
リッチ・テキスト・ポートレット	UNIX の場合 : http://*host*:7777/richtextportlet/portlets/wsrp2?WSDL Windows の場合 : http://*host*:80/richtextportlet/portlets/wsrp2?WSDL	HTTP_Server	該当なし

表 C-1 コンポーネントの URL（続き）

コンポーネント	URL（およびデフォルト・ポート番号）	opmnctl status 出力のエントリ	ログインおよびパスワード
OmniPortlet	UNIX の場合 : http://*host*:7777/portalTools/omniPortlet/providers/omniPortlet Windows の場合 : http://*host*:80/portalTools/omniPortlet/providers/omniPortlet	HTTP_Server	該当なし
Web クリッピング	UNIX の場合 : http://*host*:7777/portalTools/webClipping/providers/webClipping Windows の場合 : http://*host*:80/portalTools/webClipping/providers/webClipping	HTTP_Server	該当なし

D

Oracle Application Server のポート番号

この付録では、Oracle Application Server のポート番号について説明します。

この付録の項目は次のとおりです。

- ポート番号とその割当て方法

  この項では、割当て済のポート範囲、デフォルトのポート番号、およびポート番号が割り当てられるタイミングとポート番号の変更に関する情報の入手元について説明します。

- ポート番号（番号別）

  この項では、すべての割当て済ポート範囲を番号別にソートして一覧表示します。特定のポート番号が Oracle Application Server で使用されているかどうかを確認する場合に便利です。

- ファイアウォールで開くポート

  この項では、Oracle Application Server がファイアウォールの内側にインストールされている場合に開く必要のあるポートを一覧表示します。

D.1 ポート番号とその割当て方法

ほとんどのポート番号は、インストール時に Oracle Application Server によって割り当てられます。この場合、割当て済のポート範囲から空いているポートが選択されます。

この項では、ポートを使用する Oracle Application Server の各サービスについて次の情報を示します。

- **コンポーネント / サービス :** コンポーネントとサービスの名前、およびポート番号が割り当てられるタイミングとポート番号の変更（可能な場合）に関する情報の入手元についての情報。
- **割当て済のポート範囲 :** ポートを割り当てる際に Oracle Application Server が使用を試みる一連のポート番号。
- **デフォルトのポート番号 :** Oracle Application Server がサービスに割り当てようとする最初のポート番号。通常は、割当て済ポート範囲の最小値になります。
- **プロトコル :** 使用するプロトコル。

ポートは次のカテゴリにソートされています。

- OC4J、OPMN および Oracle HTTP Server のポート
- Oracle WebCenter Framework および Oracle Content DB
- その他のコンポーネントのポート番号

D.1.1 OC4J、OPMN および Oracle HTTP Server のポート

表 D-1 は、10*g* リリース 3（10.1.3.2.0）のインストールにおける様々なポートの一覧です。表の「コンポーネント / サービス」列に特に記載のないかぎり、次のことが当てはまります。

- ポート番号はインストール時に割り当てられます。
- ポート番号がインストール時に割り当てられる場合、`staticports.ini` ファイルを使用すると、インストール時にそのポート番号を無効にできます。

  ほとんどのポートでは、`staticports.ini` にポート番号を指定することにより、デフォルトで割り当てられるポート番号をインストール時に無効にできます。目的のポート番号でテンプレート `staticports.ini` を作成し、特別なオプションを指定して Oracle Universal Installer を起動します。

  > **関連項目：** `staticports.ini` の使用方法の詳細は、Oracle Application Server のインストレーション・ガイドを参照してください。

- ポート番号は、インストール後に変更できます。

表 D-1 OC4J、OPMN および Oracle HTTP Server のポート

コンポーネント / サービス	割当て済のポート範囲	デフォルトのポート番号	プロトコル
Oracle HTTP Server			
Listen Port ポート番号を変更する場合は、第 4.3.3 項「Oracle HTTP Server リスニング・ポートの変更」を参照してください。また、これが Oracle WebCenter Framework のみのインストールである場合は、第 4.3.1 項「OC4J リスナー・ポートの変更」を参照してください。	80, 7777 - 7877, 8888	7777[1]	HTTP
Port ポート番号を変更する場合は、第 4.3.3 項「Oracle HTTP Server リスニング・ポートの変更」を参照してください。	80, 7777 - 7877, 8888	7777[1]	HTTP

表 D-1 OC4J、OPMN および Oracle HTTP Server のポート（続き）

コンポーネント / サービス	割当て済のポート範囲	デフォルトのポート番号	プロトコル
Listen（SSL）Port このポートは、インストール後に SSL を有効にしないかぎり使用しません。詳細は、『Oracle HTTP Server 管理者ガイド』を参照してください。 ポート番号を変更する場合は、第 4.3.3 項「Oracle HTTP Server リスニング・ポートの変更」を参照してください。	4443	4443	HTTPS
SSL Port このポートは、インストール後に SSL を有効にしないかぎり使用しません。詳細は、『Oracle HTTP Server 管理者ガイド』を参照してください。 ポート番号を変更する場合は、第 4.3.3 項「Oracle HTTP Server リスニング・ポートの変更」を参照してください。	443, 4443	4443	HTTPS
Diagnostic ポート番号を変更する場合は、第 4.3.4 項「Oracle HTTP Server 診断ポートの変更」を参照してください。	7200 - 7299	7200	TCP
OC4J			
AJP このポート番号は、インストール時に無効にできません。 ポート番号を変更する場合は、第 4.3.2 項「その他の OC4J ポートの変更」を参照してください。	12501 - 12600	12501	TCP
HTTP プロトコルを使用するデフォルトの Web サイト このポート番号は、インストール時に無効にできません。 ポート番号を変更する場合は、第 4.3.1 項「OC4J リスナー・ポートの変更」を参照してください。	8888-8987	8888	HTTP
JMS このポート番号は、インストール時に無効にできません。 ポート番号を変更する場合は、第 4.3.2 項「その他の OC4J ポートの変更」を参照してください。	12601 - 12700	12601	TCP
RMI このポート番号は、インストール時に無効にできません。 ポート番号を変更する場合は、第 4.3.2 項「その他の OC4J ポートの変更」を参照してください。	12401 - 12500	12401	TCP
RMIS このポート番号は、インストール時に無効にできません。 ポート番号を変更する場合は、第 4.3.2 項「その他の OC4J ポートの変更」を参照してください。	12701 - 12800	12701	TCP
IIOP ポートはインストール後、IIOP の構成時に割り当てられます。詳細は、『Oracle Containers for J2EE サービス・ガイド』を参照してください。 ポート番号を変更する場合は、第 4.3.2 項「その他の OC4J ポートの変更」を参照してください。	13301 - 13400	13301	TCP

表 D-1 OC4J、OPMN および Oracle HTTP Server のポート（続き）

コンポーネント / サービス	割当て済のポート範囲	デフォルトのポート番号	プロトコル
IIOPS1（サーバーのみ） ポートはインストール後、IIOPS1 の構成時に割り当てられます。 ポート番号を変更する場合は、第 4.3.2 項「その他の OC4J ポートの変更」を参照してください。	13401 - 13500	13401	TCP
IIOPS2（サーバーおよびクライアント） ポートはインストール後、IIOPS2 の構成時に割り当てられます。 ポート番号を変更する場合は、第 4.3.2 項「その他の OC4J ポートの変更」を参照してください。	13501 - 13600	13501	TCP
OPMN			
ONS Local ポート番号を変更する場合は、第 4.3.6 項「OPMN ポート（ONS Local、Request、Remote）の変更」を参照してください。	6100 - 6199	6100	HTTP/TCP
ONS Remote ポート番号を変更する場合は、第 4.3.6 項「OPMN ポート（ONS Local、Request、Remote）の変更」を参照してください。	6200 - 6299	6200	HTTP/TCP
ONS Request ポート番号を変更する場合は、第 4.3.6 項「OPMN ポート（ONS Local、Request、Remote）の変更」を参照してください。	6003 - 6099	6003	HTTP/TCP
その他のサービス			
Java Object Cache ポート番号を変更する場合は、第 4.3.5 項「Java Object Cache ポートの変更」を参照してください。	7000 - 7099	7000	TCP
Port Tunneling ポートはインストール後、ポート・トンネリングの構成時に割り当てられます。 ポート番号を変更する場合は、第 4.3.7 項「ポート・トンネリング・ポートの変更」を参照してください。	7501 - 7599	7501	TCP

[1] Windows では、デフォルトのポートは 80 です。

D.1.2 Oracle WebCenter Framework および Oracle Content DB

Oracle WebCenter Framework コンポーネントおよび Oracle Content DB では、次のポートを使用します。

- ほとんどのインストール・タイプの場合、Oracle WebCenter Framework コンポーネントでは Oracle HTTP Server ポートを使用します。ただし、Oracle WebCenter Framework のみをインストールする場合は、OC4J_WebCenter HTTP ポートが使用されます。
- Oracle Content DB では、Oracle HTTP Server ポートを使用します。

これらのポートの一覧は、第 D.1.1 項を参照してください。

D.1.3 その他のコンポーネントのポート番号

その他のポート番号は、Oracle Application Server とは別にインストールされているコンポーネントによって使用される場合があります。ポート番号の詳細は、そのコンポーネントのドキュメントを参照してください。

たとえば、OracleAS Infrastructure リリース 2（10.1.2）の詳細は、リリース 2（10.1.2）用の『Oracle Application Server 管理者ガイド』を参照してください。

D.2 ポート番号（番号別）

表 D-2 は、ポート番号で昇順にソートした Oracle Application Server のポート番号とサービスの一覧です。

表 D-2 ポート番号（番号別）

ポート番号	サービス
80	Oracle HTTP Server Listen および Oracle HTTP Server Port（Windows のみ）
443	Oracle HTTP Server Port（SSL）
4443	Oracle HTTP Server Listen（SSL）および Oracle HTTP Server Port（SSL）
6003 - 6099	OPMN ONS Request
6100 - 6199	OPMN ONS Local
6200 - 6299	OPMN ONS Remote
7000 - 7099	Java Object Cache
7200 - 7299	Oracle HTTP Server Diagnostic
7501 - 7599	Port Tunneling
7777 - 7877	Oracle HTTP Server Listen および Oracle HTTP Server Port
7890 - 7895	Oracle Application Server Guard
8250 - 8350	Oracle HTTP Server Listen（SSL）および Oracle HTTP Server Port（SSL）
8888-8987	OC4J HTTP リスナー・ポート
12401 - 12500	OC4J RMI
12501 - 12600	OC4J AJP
12601 - 12700	OC4J JMS
12701 - 12800	OC4J RMIS
13301 - 13400	OC4J IIOP
13401 - 13500	OC4J IIOPS1（サーバーのみ）
13501 - 13600	OC4J IIOPS2（サーバーおよびクライアント）

D.3 ファイアウォールで開くポート

Oracle Application Server をファイアウォールの内側にインストールする場合は、インストール中と実行時にファイアウォールで特定のポートを開く必要があります。

10*g* リリース 3（10.1.3.2.0）中間層インスタンスの場合は、Oracle Notification Server と AJP のポートにアクセスする必要があります。これらのコンポーネントが使用する次のポートをファイアウォールで開く必要があります。

- OPMN ONS Remote ポート
- OC4J AJP ポート

E

管理上の変更の例

この付録では、Oracle Application Server 環境で実行可能な管理上の変更の例を示します。これは、このマニュアルの第 V 部「バックアップとリカバリ」および『Oracle Application Server 高可用性ガイド』の Disaster Recovery に関する項の姉妹編となります。

この付録の項目は次のとおりです。

E.1 この付録の使い方

一部の管理操作は、Oracle Application Server 環境の構成の変更を伴います。これらの変更を**管理上の変更**と呼びます。管理上の変更には、アプリケーションのデプロイとアンデプロイ、トポロジの変更、ポートの変更、ユーザーの作成と削除、パスワードの変更などがあります。環境のバックアップやなんらかの同期処理が必要となる場合があるため、管理者は管理上の変更が生じるタイミングについて認識しておく必要があります。

この付録では、管理上の変更の例をコンポーネント別に一覧表示します。この一覧は、次の各手順を実行する際の手引きとして利用できます。

- バックアップとリカバリ

  環境に対して管理上の変更を行った場合は、バックアップを実行することをお薦めします。この付録を使用すると、環境のバックアップが必要な管理上の変更のタイプを確認できます。

  関連項目： 第 V 部「バックアップとリカバリ」

- プライマリ・サイトとスタンバイ・サイトの障害時リカバリの同期

  Disaster Recovery を実装する場合は、環境に対する管理上の変更を行うときにスタンバイ・サイトを更新する必要があります。この付録を使用すると、スタンバイ・サイトの更新が必要な管理上の変更のタイプを確認できます。

  関連項目：『Oracle Application Server 高可用性ガイド』

E.2 管理上の変更の例（コンポーネント別）

表 E-1 に、管理上の変更の例をコンポーネント別に示します。これらの操作の詳細は、各コンポーネントのマニュアルを参照してください。

表 E-1 管理上の変更の例

コンポーネント	管理上の変更の例
Dynamic Monitoring Service（DMS）	Application Server Control コンソールで実行する DMS 管理操作および構成操作 DMS 構成ファイル（`dms.conf` など）の手動による編集
Oracle Containers for J2EE（OC4J）	Application Server Control コンソールを使用した OC4J 管理操作および構成操作（アプリケーションのデプロイとアンデプロイ、OC4J インスタンスの作成など） OC4J 構成ファイルの手動による編集
Oracle Application Server Java Authentication and Authorization Service (JAAS) Provider（JAZN）	Application Server Control コンソールで実行する JAZN 管理操作および構成操作 `admintool` ユーティリティで実行する JAZN 管理操作および構成操作（ユーザーの追加と削除、ロール、権限、特権、パスワードの変更など）
Oracle Enterprise Manager 10*g* Application Server Control コンソール	Application Server Control コンソールで実行する、アプリケーション・サーバー全体またはコンポーネント固有の管理操作および構成操作（`oc4jadmin` パスワードの変更、アプリケーションのデプロイとアンデプロイ、構成ファイルの変更を伴う操作など）
Oracle HTTP Server	Application Server Control コンソールで実行する Oracle HTTP Server の管理操作および構成操作（VM 数の変更、仮想ホストの作成など） Oracle HTTP Server 構成ファイルの手動による編集
Oracle Process Manager and Notification Server（OPMN）	Application Server Control コンソールで実行する OPMN 管理操作および構成操作 OPMN 構成ファイル（`opmn.xml` など）の手動による編集

F

LDAP ベースのレプリカ構成の補助手順

この付録では、第 8.3 項「新しいホストへの 10.1.4 または 10.1.2 Identity Management の移動」で言及されている補助手順について説明します。

この付録の項目は次のとおりです。

- LDAP ベースのレプリカについて
- LDAP ベースのレプリカのインストールと設定

F.1 LDAP ベースのレプリカについて

この項では、LDAP ベースのレプリカをインストールおよび構成する方法について説明します。この項の項目は次のとおりです。

- LDAP ベースのレプリカとは
- Infrastructure サービスの変更における LDAP ベースのレプリカの使用方法

F.1.1 LDAP ベースのレプリカとは

Oracle Internet Directory のレプリケーションは、複数のディレクトリ・サーバーにある同じデータ（ネーミング・コンテキスト）をコピーして保持するプロセスです。簡単に言うと、レプリケーションとは、同じ情報を格納した同じディレクトリを 2 つ持つことを意味します。一方のディレクトリをマスター（サプライヤ）と呼びます。このディレクトリには、ネーミング・コンテキストのマスター・コピーを格納します。もう一方のディレクトリをレプリカ（コンシューマ）と呼びます。マスターは、レプリケーションの更新をレプリカに提供します。これにより、マスターとレプリカは同期します。

レプリカには様々なタイプがあります。この手順では、LDAP ベースのレプリカを使用します。このレプリカでは、マスターとレプリカ間のデータ転送が LDAP プロトコルで行われます。

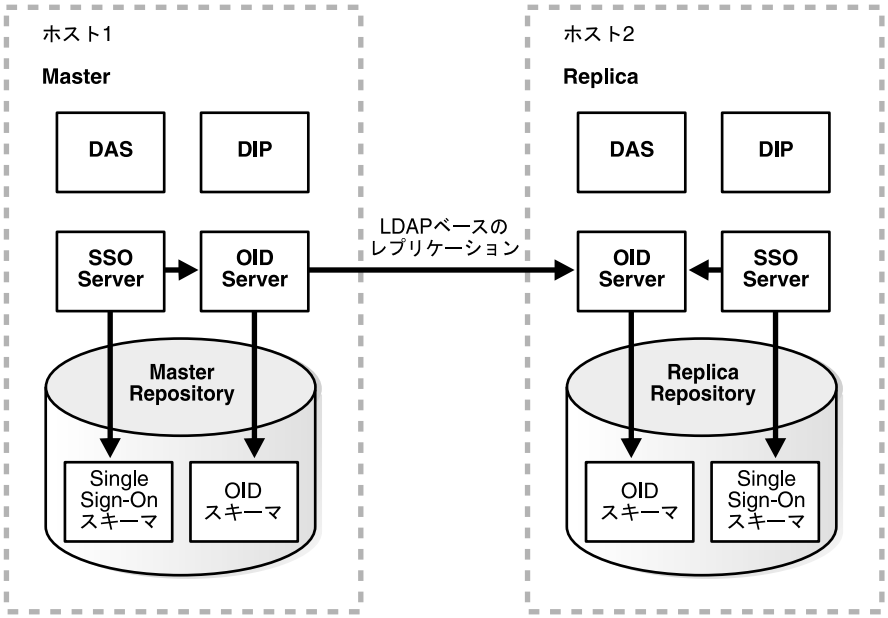
> **関連項目：** ディレクトリのレプリケーションと LDAP ベースのレプリカの詳細は、『Oracle Internet Directory 管理者ガイド』を参照してください。

この手順では、マスターとレプリカのディレクトリは、これらのディレクトリが含まれる Identity Management インストールとこれらのディレクトリをサポートする Metadata Repository で構成される、より大きな環境の一部になります。これは LDAP ベースのレプリカ環境と呼ばれ、次の要素で構成されています。

- **Master**: ネーミング・コンテキストのマスター・コピーを保持する Oracle Internet Directory を含む Identity Management インストールです。レプリケーションの更新をレプリカに提供します。
- **Master Repository**: Identity Management のスキーマを格納するためにマスターで使用される Metadata Repository です。
- **Replica**: レプリケートされた Oracle Internet Directory を含む Identity Management インストールです。
- **Replica Repository**: Identity Management のスキーマを格納するためにレプリカで使用される Metadata Repository です。

図 F-1 に、LDAP ベースのレプリカ環境を示します。

図 F-1 LDAP ベースのレプリカ環境



F.1.2 Infrastructure サービスの変更における LDAP ベースのレプリカの使用方法

一般に LDAP ベースのレプリカは、ディレクトリ・ユーザーに高可用性を提供し、パフォーマンスを強化するために使用されます。

第 8.3 項の Infrastructure サービスの変更を目的とする場合、LDAP ベースのレプリカは、あるホストから別のホストへ Identity Management を移行する手段として作成されます。Master は元の Identity Management インストールであり、Replica は新しい Identity Management インストールになります。この場合は、レプリケーションによって元の Identity Management の同一コピーが新しいホストに作成されます。これにより、古い Identity Management（Master）から新しい Identity Management（Replica）に中間層を移行し、Master を廃棄することができます。

F.2 LDAP ベースのレプリカのインストールと設定

この項では、LDAP ベースのレプリカ環境をインストールおよび設定する方法について説明します。

F.2.1 構成にあたっての注意

次の重要な留意点を確認してから手順を開始してください。

- この手順では、Identity Management と Metadata Repository を含む、単一の Infrastructure Oracle ホームを使用します。ただし、Infrastructure インストールを分割して、一方の Oracle ホームに Identity Management を、もう一方の Oracle ホームに Metadata Repository を格納することもできます。また、Identity Management のコンポーネント（OracleAS Single Sign-On、Oracle Internet Directory、Delegated Administration Services、Directory Integration and Provisioning）を様々なホストに分散することもできます。その場合は、それぞれの Oracle ホームの各コンポーネントで操作を実行します。
- レプリカでは、Oracle Universal Installer によってレポートされる内容にかかわらず、非 SSL の Oracle Internet Directory ポートにはポート 389 が、SSL の Oracle Internet Directory ポートにはポート 636 が常に使用されます。レプリカ・ホストのポート 389 と 636 を他のプロセスが使用していないことを確認してから、この手順を開始してください。

- *ORACLE_HOME*/bin にある ldapsearch および ldapmodify コマンドを使用してください（オペレーティング・システムによっては、これらのコマンドの独自のバージョンが付属していますが、そのバージョンは使用しないでください）。
- この手順では、remtool コマンドおよび oidpasswd コマンドを使用します。これらのコマンドから返されるメッセージは UTF-8 エンコードであり、英語以外のほとんどの環境では読み取ることができません。これを回避するには、これらのコマンドを実行する前に NLS_LANG 環境変数を american_america.*character_set* に設定してください。これにより、ほとんどのキャラクタ・セット（US7ASCII など）が機能するようになります。

  関連項目：『Oracle Application Server グローバリゼーション・ガイド』

- ORACLE_HOME および ORACLE_SID 環境変数が設定されていることを確認します。これはすべてのプラットフォームに適用されます。

F.2.2 手順

この項では、LDAP ベースのレプリカを設定する手順を説明します。この項は、次の作業で構成されています。

- 作業 1: Master および Master Repository の取得
- 作業 2: 中間層インスタンスのインストール（オプション）
- 作業 3: Replica のインストールと構成

作業 1: Master および Master Repository の取得

ほとんどの場合、Master および Master Repository はすでに取得されています。

第 8.3 項「新しいホストへの 10.1.4 または 10.1.2 Identity Management の移動」の手順を実行する場合、Master および Master Repository は新しいホストへ移行するインストールであり、LDAP ベースのレプリカは再配置のインストールです。

最初から始める場合は、Master および Master Repository を次のようにインストールできます。

1. Oracle Universal Installer を使用して Oracle Application Server をインストールします。
2. Infrastructure インストールを選択します。
3. Identity Management および OracleAS Metadata Repository のインストールを選択します。
4. Oracle Internet Directory、OracleAS Single Sign-On、Delegated Administration Services および Directory Integration and Provisioning の各コンポーネントを構成するように選択します。

作業 2: 中間層インスタンスのインストール（オプション）

ほとんどの場合、Identity Management サービスの Master を使用する中間層インスタンスはすでにインストールされています。これらのインスタンスはそのまま使用できますが、必要な場合は、Master を使用するための追加インスタンスのインストールと構成をここで行うか、Replica の構成後、手順の最後に行うこともできます（またはその両方）。

これらの中間層インスタンスでは、製品メタデータの Master Repository を使用することも、別のリポジトリを使用することもできます。

作業 3: Replica のインストールと構成

Oracle Universal Installer を使用して Replica をインストールおよび構成できます。Replica は、Master とは別のホストにインストールしてください。

関連項目： Oracle Internet Directory のレプリカのインストールについては、Oracle Application Server のインストレーション・ガイドを参照してください。

インストールが終了すると、レプリケーションが構成され、すべてのコンポーネントが起動します。作業を開始した主要な手順（第 8.3 項「新しいホストへの 10.1.4 または 10.1.2 Identity Management の移動」）に戻ることができます。

G

Oracle Application Server のリリース番号の確認

この付録では、Oracle Application Server のリリース番号の表記方法について説明します。

この付録の項目は次のとおりです。

注意： Oracle Application Server インストールに適用した、すべての個別パッチのログを記録しておくことをお薦めします。

G.1 リリース番号の書式

Oracle で使用されるリリース・レベルの命名体系を理解するには、図 G-1 に示した Oracle Application Server のリリース番号の例を確認してください。

図 G-1 Oracle Application Server のリリース番号の例

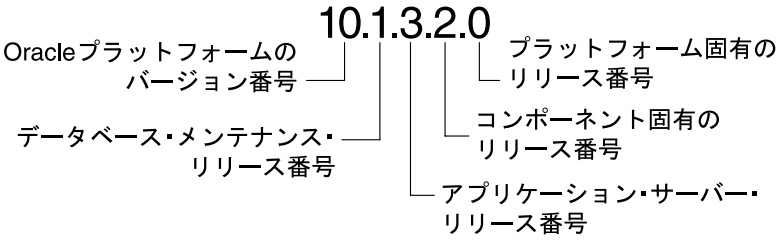


図 G-1 の各数字のラベルの意味は次のとおりです。

- Oracle プラットフォームのバージョン番号

  これは最も一般的な識別子です。この数字は、Oracle データベース・サーバーや Oracle Application Server などのアプリケーションの主要な新規版（バージョン）を表すものであり、そのリリースに重要な新機能が含まれることを表します。

- データベース・メンテナンス・リリース番号

  この数字は、メンテナンス・リリース・レベルを表します。新機能がいくつか含まれている場合もあります。

- アプリケーション・サーバー・リリース番号

  この数字は、Oracle Application Server のリリース・レベルを表します。

- コンポーネント固有のリリース番号

  この数字によって、コンポーネント固有のリリース・レベルが識別されます。コンポーネントが異なると、ここに表示される数値も異なることがあります。これは、コンポーネントのパッチ・セットや仮リリースなどに基づきます。

- プラットフォーム固有のリリース番号

  この数字は、プラットフォーム固有のリリースを表します。

G.2 Oracle Application Server インストールのリリース番号の確認

すべての Oracle Application Server インストールにはリリース番号があります。この番号は、パッチ・セット・リリースの適用時やインストールのアップグレード時に更新されます。

Oracle Application Server インストールのリリース番号を確認するには、Oracle Universal Installer で次の操作を行います。

1. Oracle Universal Installer を起動します。

   (UNIX) *ORACLE_HOME*/oui/bin/runInstaller.sh
   (Windows) *ORACLE_HOME*¥oui¥bin¥setup.exe

2. 「**インストール済の製品**」をクリックして、「インベントリ」ページを表示します。
3. 「インベントリ」ページで、「**Oracle ホーム**」を展開します。ホスト上のすべてのインストールのエントリが表示されます。
4. 目的のインストールの「Oracle ホーム」エントリを展開します。
5. 元のインストールのリリース番号が付いたエントリと、これまでに適用されたすべてのパッチ・セットのエントリが表示されます。

G.3 コンポーネント・リリース番号の確認

すべての Oracle Application Server のコンポーネントにはリリース番号があり、コンポーネントの多くはリリース番号を持つサービスを備えています。これらの番号は、パッチ・セット・リリースの適用時やインストールのアップグレード時に更新されることがあります。

コンポーネントとそのサービスのリリース番号は、次の方法で確認できます。

- ファイル・システムの使用
- Oracle Universal Installer の使用

ファイル・システムの使用

UNIX では、次のコマンドを実行すると、コンポーネント・リリース番号を確認できます。

```
cd ORACLE_HOME/inventory
ls -d Components*/*/*
```

Oracle Universal Installer の使用

Oracle Universal Installer を使用して Oracle Application Server をインストールした場合、コンポーネント・リリース番号を表示するには次の操作を行います。

1. Oracle Universal Installer を起動します。

   ```
   (UNIX) ORACLE_HOME/oui/bin/runInstaller.sh
   (Windows) ORACLE_HOME¥oui¥bin¥setup.exe
   ```

2. 「**インストール済の製品**」をクリックして、「インベントリ」ページを表示します。
3. 「インベントリ」ページで、「**Oracle ホーム**」を展開します。ホスト上のすべてのインストールのエントリが表示されます。
4. 目的のインストールの「Oracle ホーム」エントリを展開します。
5. 元のインストールのリリース番号が付いたエントリと、これまでに適用されたすべてのパッチ・セットのエントリが表示されます。
6. 最初のエントリを展開して、インストール時のコンポーネント・リリース番号を確認します。以降のパッチ・セットのエントリがある場合は、そのエントリを展開して、パッチ・セットごとに更新されたコンポーネント・リリース番号を確認します。

G.4 OPatch ユーティリティの使用方法

OPatch ユーティリティは、Oracle Application Server などの Oracle 製品に対して、個別パッチの適用とロールバックを可能にするツールです。OPatch ユーティリティの最新情報、および更新の確認は、次の OracleMetaLink を参照してください。

```
http://www.oracle.com/support/metalink/index.html
```

G.4.1 要件

OPatch ユーティリティの要件は次のとおりです。

- Perl 環境。これは、Oracle Application Server に付属しており、パッチ・セットとともにダウンロードすることもできます。
- Oracle ホームの環境変数（ORACLE_HOME）は、有効な Oracle ホーム・ディレクトリを指し、Oracle ホーム・ディレクトリのインストール時に使用された値と一致している必要があります。
- インストール時に `-invPtrLoc` コマンドライン引数を使用した場合は、OPatch ユーティリティを使用するときにもこれを使用する必要があります。プラットフォームでデフォルトになっている中央インベントリを使用することをお薦めします。
- `jar`、`java`、`ar`、`cp`、`make` の各コマンドが、PATH 文で使用できることが必要です。これらのコマンドは、すべてのプラットフォームで使用可能とは限りません。

- Oracle Real Application Clusters 環境のライブラリ・パスが正しく設定されている必要があります。詳細は、opatch/doc ディレクトリの FAQ 資料を参照してください。

関連項目： OPatch ユーティリティの最新情報、および更新の確認は、次の Oracle*MetaLink* を参照してください。

http://www.oracle.com/support/metalink/index.html

G.4.2 OPatch ユーティリティの実行

OPatch ユーティリティは、ORACLE_HOME/OPatch ディレクトリにあります。OPatch ユーティリティの構文を次に示します。

path_to_opatch/opatch *option -command_line_arguments*

前述の例では、次のようになります。

- *option*: OPatch のオプション。値は表 G-1 で説明します。
- *command_line_arguments*: 各オプションに対するコマンドライン引数。値は次の項で説明します。

表 G-1 OPatch ユーティリティのオプション

オプション	説明
apply	個別パッチをインストールします。詳細は、第 G.4.2.1 項を参照してください。
lsinventory	システムに現在何がインストールされているかを一覧表示します。詳細は、第 G.4.2.2 項を参照してください。
query	特定のパッチに特定の詳細を問い合せます。詳細は、第 G.4.2.3 項を参照してください。
rollback	個別パッチを削除します。詳細は、第 G.4.2.4 項を参照してください。
version	パッチ・ツールの現行バージョンを出力します。詳細は、第 G.4.2.5 項を参照してください。

オプションの詳細情報を表示するには、次のコマンドを使用します。

path_to_OPatch/opatch *option* -help

Perl を使用している場合は、次のコマンドを使用します。

perl opatch.pl *option* -help

G.4.2.1 apply オプション

apply オプションは、指定した Oracle ホームに個別パッチを適用します。ORACLE_HOME 環境変数は、パッチが適用される Oracle ホームに設定する必要があります。このオプションは、次の構文で使用します。

```
path_to_opatch/opatch apply patch_location [-delay value] [-force] ¥
[-invPtrLoc path] [-jdk location] [-jre location] [-local] ¥
[-minimize_downtime] [-no_bug_superset] [-no_inventory] ¥
[-oh Oracle_home_location] ¥
[-post options_to _be_passed_into_post [-opatch_post_end]]¥
[-pre options_to_be_passed_into_pre [-opatch_pre_end]] ¥
[-retry value] [-silent] [-verbose]
```

次の表に、apply オプションのコマンドライン引数を示します。

引数	説明
delay	前にエラーがあった場合に、インベントリのロックを試行するまでに待機する秒数を指定します。
force	競合するパッチをシステムから削除します。競合があるためにパッチの適用が妨げられている場合は、-force 引数を使用するとパッチを適用できます。
invPtrLoc	oraInst.loc ファイルの場所を指定します。この引数は、インストール時に -invPtrLoc 引数が使用された場合に必要です。プラットフォームでデフォルトになっている中央インベントリを使用することをお薦めします。
jdk	Oracle ホーム・ディレクトリのデフォルトの場所でなく、使用する特定の JDK（jar）の場所を指定します。
jre	Oracle ホーム・ディレクトリのデフォルトの場所でなく、使用する特定の JRE（Java）の場所を指定します。
local	OPatch ユーティリティがローカル・ノードにパッチを適用し、ローカル・ノードのインベントリを更新することを指定します。パッチ、またはインベントリの更新は、他のノードに伝播しません。 この引数は、Oracle Real Application Clusters 環境およびクラスタリングされていない環境で使用できます。パッチを適用する前にクラスタ全体が停止した場合、この引数は、ローリング以外のパッチに使用できます。
minimize_ downtime	OPatch ユーティリティでパッチを適用するノードの順序を指定します。 この引数は、Oracle Real Application Clusters 環境にのみ適用されます。-local 引数とともに使用したり、ローリング・パッチで使用することはできません。
no_bug_superset	現在のパッチの修正対象バグが、Oracle ホーム・ディレクトリにインストール済のパッチの修正済バグと同じか、そのスーパーセットである場合に、エラーを出力するように指定します。
no_inventory	インベントリの読取りと更新を迂回します。この引数は、-local 引数とともに使用することはできません。この引数により、インストールはサポートされない状態になります。
oh	デフォルトのかわりに使用する Oracle ホーム・ディレクトリを指定します。
opatch_post_end	post オプションの最後をマークします。この引数は、post 引数とともに使用されます。この引数を使用しない場合は、post の後にあるものがすべて、post に渡されます。
opatch_pre_end	pre オプションの最後をマークします。この引数は、pre 引数とともに使用されます。この引数を使用しない場合は、pre の後にあるものがすべて、pre に渡されます。
post	標準的なパラメータ以外に、post スクリプト内に渡されるパラメータを指定します。
pre	標準的なパラメータ以外に、pre スクリプト内に渡されるパラメータを指定します。
retry	インベントリのロックが失敗した場合に、OPatch ユーティリティが試行する回数を指定します。
patch_location	個別パッチのディレクトリを指定します。これは、パッチと同じ名前のディレクトリにする必要があります。
silent	ユーザーが操作する手間を減らし、あらゆる答えに「はい」をデフォルト設定します。
verbose	結果を画面とログ・ファイルに出力します。

注意： パッチが SQL による変更で構成されている場合、それらの変更のみがステージングされます。パッチに付属している手順に従って、影響を受けるインスタンスに対して手動でパッチを適用してください。一部の製品では、SQL アプリケーションは、このツールによってステージング後のアクションとして実行されます。このようなパッチはロールバックできません。

G.4.2.2 lsinventory オプション

lsinventory オプションは、特定の Oracle ホーム・ディレクトリ、またはすべてのインストールについて、システムに何がインストールされているかを報告します。このオプションは、次の構文で使用します。

```
path_to_opatch/opatch lsinventory [-all] [-detail] [-invPrtLoc path] ¥
[-jre location] [-oh Oracle_home_location]
```

次の表に、lsinventory オプションのコマンドライン引数を示します。

引数	説明
all	検出された Oracle ホーム・ディレクトリごとに、その名前とインストール・ディレクトリを報告します。
detail	インストールされた製品などの詳細を報告します。この引数は、-all 引数とともに使用することはできません。
invPtrLoc	oraInst.loc ファイルの場所を指定します。この引数は、インストール時に -invPtrLoc コマンドライン引数が使用された場合に必要です。プラットフォームでデフォルトになっている中央インベントリを使用することをお薦めします。
jre	Oracle ホーム・ディレクトリのデフォルトの場所でなく、使用する特定の JRE（Java）の場所を指定します。
oh	デフォルト・ディレクトリのかわりに使用する Oracle ホーム・ディレクトリを指定します。

opatch lsinventory -detail の出力例を次に示します。

```
ORACLE_HOME      LOCATION
-----------      --------
Home1            /private/phi_local/OraHome1
   There is no Interim Patch
Home2            /private/phi_local/OraHome2
   There is no Interim Patch
Home3           /private/phi_local/OraHome6
   Installed Patch List:
    ====================
    1) Patch 20 applied on Mon Jul 11 15:53:51 PDT 2006
       [ Base Bug(s): 21  ]
    2) Patch 80 applied on Fri Jul 01 16:15:52 PDT 2006
       [ Base Bug(s): 80 81  ]
```

G.4.2.3 query オプション

query オプションは、特定のパッチに特定の詳細を問い合せます。これによって、パッチおよびパッチの対象となるシステムの情報が得られます。このオプションは、次の構文で使用します。

```
path_to_opatch/opatch query [-all] [-get_base_bug] [-get_component] ¥
[-invPtrLoc path] [-get_date] [-get_os] [-get_system_change] [-is_rolling] ¥
```

次の表に、query オプションのコマンドライン引数を示します。

引数	説明
all	パッチについてのすべての情報を取得します。これは、すべてのコマンドライン引数を設定した場合と同じです。
get_base_bug	パッチによって修正される基本バグを記述します。
get_component	オプションも必須も含め、パッチを適用する Oracle コンポーネントを記述します。
get_date	パッチのビルド日付を示します。
get_os	パッチでサポートされるオペレーティング・システムを記述します。
get_system_change	パッチによってシステムに行われる変更を記述します。この引数は使用できません。
invPtrLoc	oraInst.loc ファイルの場所を指定します。この引数は、インストール時に -invPtrLoc コマンドライン引数が使用された場合に必要です。プラットフォームでデフォルトになっている中央インベントリを使用することをお薦めします。
is_rolling	パッチが Oracle Real Application Clusters 用のローリング・パッチであるかどうかを指定します。パッチのセットを、クラスタ全体に同時に適用する必要はありません。パッチは、選択したノードのセットに対して一度に適用できます。

G.4.2.4 rollback オプション

rollback オプションは、特定の個別パッチを該当する Oracle ホーム・ディレクトリから削除します。このオプションは、次の構文で使用します。

```
path_to_opatch/opatch rollback -id patch_id -ph patch_directory ¥
[-delay value] [-invPtrLoc path] [-jdk location] [-jre location]¥
[-local] [-oh Oracle_home_location] ¥
[-post options_to_be_passed_into_post [-opatch_post_end]] ¥
[-pre options_to_be_passed_into_pre [-opatch_pre_end]] [-retry value] ¥
[-silent] [-verbose]
```

次の表に、rollback オプションのコマンドライン引数を示します。

引数	説明
delay	-retry 引数が apply オプションとともに使用されている場合に、OPatch ユーティリティがインベントリのロックを再試行するまでに待機する秒数を指定します。
id	ロールバックするパッチを指示します。すべてのパッチ ID を表示するには、-lsinventory オプションを使用します。パッチを正常にロールバックするには、パッチ ID を指定する必要があります。
invPtrLoc	oraInst.loc ファイルの場所を指定します。この引数は、インストール時に -invPtrLoc コマンドライン引数が使用された場合に必要です。プラットフォームでデフォルトになっている中央インベントリを使用することをお薦めします。
jdk	Oracle ホーム・ディレクトリのデフォルトの場所でなく、使用する特定の JDK（jar）の場所を指定します。
jre	Oracle ホーム・ディレクトリのデフォルトの場所でなく、使用する特定の JRE（Java）の場所を指定します。

引数	説明
local	OPatch ユーティリティがローカル・ノードにパッチを適用し、ローカル・ノードのインベントリを更新することを指定します。パッチ、またはインベントリの更新は、他のノードに伝播しません。 この引数は、Oracle Real Application Clusters 環境およびクラスタリングされていない環境で使用できます。パッチを適用する前にクラスタ全体が停止した場合、この引数は、ローリング以外のパッチに使用できます。
oh	デフォルト・ディレクトリのかわりに使用する Oracle ホーム・ディレクトリを指定します。
opatch_post_end	post オプションの最後をマークします。この引数は、post 引数とともに使用されます。この引数を使用しない場合は、post の後にあるものがすべて、post に渡されます。
opatch_pre_end	pre オプションの最後をマークします。この引数は、pre 引数とともに使用されます。この引数を使用しない場合は、pre の後にあるものがすべて、pre に渡されます。
ph	有効なパッチ・ディレクトリ領域を指定します。このユーティリティでは、パッチ・ディレクトリ内で検出されるコマンド・タイプを使用して、現行オペレーティング・システムに対して使用するコマンドが特定されます。
post	標準的なパラメータ以外に、post スクリプト内に渡されるパラメータを指定します。
pre	標準的なパラメータ以外に、pre スクリプト内に渡されるパラメータを指定します。
retry	インベントリのロックが失敗した場合に、OPatch ユーティリティが試行する回数を指定します。
silent	ユーザーが操作する手間を減らし、あらゆる答えに「はい」をデフォルト設定します。
verbose	結果を画面とログ・ファイルに出力します。

G.4.2.5 version オプション

version オプションは、OPatch ユーティリティの現行バージョン番号を表示します。このオプションは、次の構文で使用します。

path_to_opatch/opatch version

H

Oracle Application Server のトラブルシューティング

この付録では、Oracle Application Server を使用しているときに発生する可能性のある障害のトラブルシューティング方法について説明します。この付録の項目は次のとおりです。

- Oracle Application Server の障害の診断
- 一般的な障害と解決策
- Application Server Control のトラブルシューティング
- まだ解決しない場合

関連項目：

- SSL のトラブルシューティングの詳細は、第 14 章「SSL のトラブルシューティング」を参照してください。
- OracleAS Recovery Manager のトラブルシューティングの詳細は、第 19 章「OracleAS Recovery Manager のトラブルシューティング」を参照してください。

H.1 Oracle Application Server の障害の診断

Oracle Application Server コンポーネントは、起動および停止情報、エラー、警告メッセージ、HTTP リクエスト時のアクセス情報など、すべての種類のイベントを記録するメッセージが格納されたログ・ファイルを生成します。このログ・ファイルを、障害の特定と診断に使用できます。ログ・ファイルの使用と読取りの詳細は、第 5 章「ログ・ファイルの管理」を参照してください。

H.2 一般的な障害と解決策

この項では、一般的な障害と解決策について説明します。この項の項目は次のとおりです。

H.2.1 ガベージ・コレクションの一時停止によって、アプリケーションのパフォーマンスが低下する

アプリケーションのパフォーマンスが遅くなるか、アプリケーションが応答しなくなります。

この問題の原因と解決策の詳細は、『Oracle Containers for J2EE 構成および管理ガイド』のガベージ・コレクションの一時停止によって影響を受けるアプリケーションのパフォーマンスに関する項を参照してください。

H.2.2 アプリケーション・サーバーから接続拒否エラーが返される

高負荷の状況では（アプリケーション・サーバーへの同時接続ユーザー数が短期間に急増した場合など）、サーバーから次のようなエラー・メッセージが返されることがあります。

```
IOException in sending request - Connection refused
```

障害

同時接続ユーザー数が増加した場合に、リクエスト処理に可能な Oracle HTTP Server の最大子プロセス数がすべて使用されることがあります。

解決策

Oracle HTTP Server の MaxClients ディレクティブ値を大きくする必要があります。
MaxClients ディレクティブにより、同時接続が可能なクライアント数が制限されます。

これが原因かどうかを調べるには、次のいずれかの方法を使用します。

- Oracle HTTP Server エラー・ログ・ファイルで、次のエラー・メッセージを検索します。

  ```
  server reached MaxClients setting, consider raising the MaxClients setting
  ```

デフォルトでは、エラー・ログ・ファイルは次の場所にあります。

```
(UNIX) ORACLE_HOME/Apache/Apache/logs/error_log
(Windows) ORACLE_HOME¥Apache¥Apache¥logs¥error_log
```

- Application Server Control コンソールのメトリックを使用して、子プロセスの状況を対話的に監視します。特に、HTTP_Server のホーム・ページで、次のメトリックを調べます。
  - 「ステータス」セクションの「**アクティブな接続**」には、HTTP リクエストを現在実行中のクライアント数が表示されます。
  - 「レスポンスと負荷」セクションの「**アクティブ・リクエスト**」には、現在処理中のアクティブ・リクエストの総数が表示されます。
- mod_status が収集する情報を調べます。mod_status モジュールにより、現行サーバーに関する統計を表示する HTML ページが用意されています。すべてのプロセスがビジーかどうかをチェックします（デフォルトでは、mod_status は localhost アクセスに対してのみ有効化されています）。詳細は、次を参照してください。

  ```
  http://httpd.apache.org/docs/mod/mod_status.html
  ```

さらには、保留中の接続に適用する最大キュー長（ListenBackLog ディレクティブ）の値を大きくすることや永続的な接続（KeepAlive ディレクティブ）の影響についても調べます。

Oracle HTTP Server ディレクティブとその値の変更方法の詳細は、『Oracle HTTP Server 管理者ガイド』を参照してください。Oracle HTTP Server プロセスのチューニングの詳細は、『Oracle Application Server パフォーマンス・ガイド』を参照してください。

H.2.3 ポートの競合により Oracle HTTP Server が起動できない

ポートの競合が原因で Oracle HTTP Server を起動できない場合は、次のエラーが表示されることがあります。

```
[crit] (98) Address already in use: make_sock: could not bind to port 7778
```

この問題の原因と解決策の詳細は、『Oracle HTTP Server 管理者ガイド』の付録「Oracle HTTP Server のトラブルシューティング」の「ポートの競合により Oracle HTTP Server が起動できない」を参照してください。

H.2.4 多数の HTTPD プロセスによるマシンのオーバーロード

1 台のマシン上で実行中の httpd プロセスが多すぎると、レスポンス時間が急激に低下します。

この問題の原因と解決策の詳細は、『Oracle HTTP Server 管理者ガイド』の付録「Oracle HTTP Server のトラブルシューティング」の「多数の HTTPD プロセスによるマシンのオーバーロード」を参照してください。

H.2.5 Oracle Application Server プロセスが起動しない

OPMN を使用して、Oracle Application Server プロセスを起動できません。

この問題の原因と解決策の詳細は、『Oracle Process Manager and Notification Server 管理者ガイド』の第 A.1.1 項「Oracle Application Server プロセスが起動しない」を参照してください。

H.2.6 OPMN の起動時に CPU 使用率が増加する

一部のコンピュータでは、OPMN の起動時に CPU 使用率が大幅に上昇します。

この問題の原因と解決策の詳細は、『Oracle Process Manager and Notification Server 管理者ガイド』の「OPMN の起動時に CPU 使用率が増加する」を参照してください。

H.2.7 ページを表示できないエラーがブラウザに表示される

ページを表示できないエラーがブラウザに表示されます。

OracleAS Web Cache リリース 2（10.1.2）をリバース・プロキシとして使用する場合、この問題の原因と解決策の詳細は、『Oracle Application Server Web Cache 管理者ガイド』の「ページを表示できないエラーがブラウザに表示される」を参照してください。

H.2.8 スタンバイ・サイトが同期化されない

OracleAS Disaster Recovery スタンバイ・サイトの OracleAS Metadata Repository が、プライマリ・サイトの OracleAS Metadata Repository と同期化されない場合があります。

この問題の原因と解決策の詳細は、『Oracle Application Server 高可用性ガイド』の第 A.1.1 項「スタンバイ・サイトが同期化されない」を参照してください。

H.2.9 フェイルオーバーまたはスイッチオーバー後にスタンバイ・インスタンスの起動に失敗する

スタンバイ・インスタンスが、フェイルオーバーまたはスイッチオーバー操作の後に起動しません。

この問題の原因と解決策の詳細は、『Oracle Application Server 高可用性ガイド』の第 A.1.2 項「フェイルオーバーまたはスイッチオーバー後にスタンバイ・インスタンスの起動に失敗する」を参照してください。

H.3 Application Server Control のトラブルシューティング

次の各項では、Application Server Control を使用したときに発生する問題について説明します。

- 管理者（oc4jadmin）のパスワードの再設定
- Internet Explorer 6.0 および Netscape Navigator 7.0 でのデプロイのパフォーマンス
- OC4J のメモリー不足エラーのトラブルシューティング
- Web モジュールまたは Web サービスのテスト時に発生する「403 Forbidden - Directory browsing not allowed」エラー
- クラスタ・トポロジの OC4J ホーム・ページへのアクセス時の管理者資格証明エラー

H.3.1 管理者（oc4jadmin）のパスワードの再設定

Oracle Application Server のインスタンスを管理するには、管理者（`oc4jadmin`）の現在のパスワードを使用して Application Server Control コンソールにログインする必要があります。

障害

`oc4jadmin` のパスワードを忘れてしまった場合やパスワードを知らされていない場合は、Application Server Control コンソールでアプリケーション・サーバーまたはそのコンポーネントの監視や管理を行うことはできません。

解決策

`oc4jadmin` のパスワードを再設定するには、Oracle Application Server インスタンスをインストールしたユーザーとしてログインして、次の手順を実行します。

1. OC4J と Application Server Control を停止します。

   アプリケーション・サーバー・インスタンスの Oracle ホームで次のコマンドを入力します。

   ```
   (UNIX) ORACLE_HOME/opmn/bin/opmnctl stopproc ias-component=OC4J
   (Windows) ORACLE_HOME¥opmn¥bin¥opmnctl stopproc ias-component=OC4J
   ```

2. 次のファイルを探し、テキスト・エディタで開きます。

   (UNIX) *ORACLE_HOME*/j2ee/home/config/system-jazn-data.xml
   (Windows) *ORACLE_HOME*¥j2ee¥home¥config¥system-jazn-data.xml

3. oc4jadmin ユーザーの資格証明プロパティを定義している行を見つけます。

   system-jazn-data.xml のセクションの例を次に示します。太字の部分が、暗号化された credentials エントリです。

```
<jazn-realm>
   <realm>
      <name>jazn.com</name>
      <users>
       .
       .
       .
         <user>
            <name>oc4jadmin</name>
            <display-name>OC4J Administrator</display-name>
            <description>OC4J Administrator</description>
            <credentials>{903}4L50lHJWIFGwLgHXTub7eYK9e0AnWLUH</credentials>
         </user>
```

4. 暗号化された既存のパスワードを、新しいパスワードで置き換えます。

   パスワードの前に感嘆符（!）を必ず付けてください。次に例を示します。

```
<credentials>!mynewpassword123</credentials>
```

   oc4jadmin ユーザーのパスワードは、次のガイドラインに従って作成してください。

   - 文字列の長さは、5 ～ 30 文字にする必要があります。
   - 文字列の先頭には英文字を使用します。パスワードの先頭には、数字、アンダースコア（_）、ドル記号（$）または番号記号（#）を使用できません。
   - 少なくとも数字を 1 つ使用します。
   - 英数字および特殊文字（ドル記号（$）、番号記号（#）またはアンダースコア（_））のみを含めることができます。
   - Oracle 予約語（VARCHAR など）は使用できません。

   関連項目： Oracle Application Server のインストレーション・ガイドの「oc4jadmin ユーザーとそのパスワードの制限」

5. 次のディレクトリの内容を削除して、キャッシュされたパスワードを削除します。

   (UNIX)
   ORACLE_HOME/j2ee/*oc4jinstance*/persistence/ascontrol/ascontrol/securestore/
   (Windows)
   ORACLE_HOME¥j2ee¥*oc4jinstance*¥persistence¥ascontrol¥ascontrol¥securestore¥

6. OC4J と Application Server Control を起動します。

   Application Server Control の再起動後は、管理者（oc4jadmin）の新しいパスワードが使用されます。このパスワードは、system-jazn-data.xml ファイルの中に暗号化されて格納されます。

   関連項目： 第 A.1 項「Application Server Control の起動と停止」

H.3.2 Internet Explorer 6.0 および Netscape Navigator 7.0 でのデプロイのパフォーマンス

障害

Microsoft Internet Explorer 6.0 または Netscape 7.0 を使用しているときに OC4J アプリケーションをデプロイしようとすると、ファイルのアップロードに極端に時間がかかることがあります（たとえば、Netscape 7.1 なら 15 秒でアップロードできる 45 MB の `.ear` ファイルが、アップロードに 10 分かかる）。

解決策

Netscape Navigator を使用している場合は、Netscape 7.1 にアップグレードしてください。

Internet Explorer を使用している場合は、この障害の対処方法を記述した次の Microsoft サポート技術情報を参照してください。

```
http://support.microsoft.com/default.aspx?scid=kb;en-us;329781
```

H.3.3 OC4J のメモリー不足エラーのトラブルシューティング

障害

OC4J インスタンスにデプロイするアプリケーションのサイズや数によっては、メモリー不足エラーが発生する場合があります。

解決策

OC4J プロセス用の Java Virtual Machine（JVM）のヒープ・サイズを調整します。

関連項目：

- 『Oracle Application Server パフォーマンス・ガイド』の「OC4J での J2EE アプリケーションの最適化」の「OC4J プロセスの JVM ヒープ・サイズの設定」
- Application Server Control コンソールのオンライン・ヘルプの、OC4J サーバー・プロパティの管理に関する項

H.3.4 Web モジュールまたは Web サービスのテスト時に発生する「403 Forbidden - Directory browsing not allowed」エラー

障害

Application Server Control コンソールで、Web モジュールをテストして、正しく機能していることを確認できます。ただし、多くの場合、「**モジュールのテスト**」または「**サービスのテスト**」のボタンをクリックすると、次のエラーが Web ブラウザで発生します。

- Mozilla Firefox の場合：

  ```
  403 Forbidden - Directory browsing not allowed
  ```

- Microsoft Internet Explorer の場合：

  ```
  You are not authorized to view this page.
  You might not have permission to view this directory or page using the credentials
  you supplied.
  ```

解決策

Application Server Control では、テストしている Web モジュールまたは Web サービスの完全な URL を判断できない場合があります。かわりに、Application Server Control では Web モジュールまたは Web サービスのルート・コンテキストに関する情報に基づいて、URL を組み立てます。

この問題を回避するには、「Web モジュールのテスト」ページまたは「Web サービスのテスト」ページのテキスト・フィールドを使用してアプリケーションの完全な URL を入力し、「**モジュールのテスト**」または「**サービスのテスト**」をクリックします。

H.3.5 クラスタ・トポロジの OC4J ホーム・ページへのアクセス時の管理者資格証明エラー

障害

「クラスタ・トポロジ」ページで OC4J インスタンスの名前をクリックすると、管理者資格証明を入力するように求められます。ただし、適切な管理者資格証明を入力しても、Application Server Control に次のエラーが表示されます。

```
Administrator credentials were saved but they cannot be used to make a
connection.  Enter new credentials or cancel.
```

解決策

管理 OC4J インスタンスが安全な Remote Method Invocation（RMIS）プロトコルを使用するように構成されていることを確認します。管理しているリモート OC4J インスタンスが RMIS 用に構成されていない場合、構成によっては、OC4J ホーム・ページにアクセスできないことがあります。

Application Server Control コンソールの RMIS 接続を有効にする方法の詳細は、第 A.3 項「Application Server Control コンソールのセキュリティの構成」を参照してください。

H.4 まだ解決しない場合

この他の解決策は、Oracle*MetaLink*（`http://metalink.oracle.com`）で公開されています。発生した障害の解決策が見つからない場合は、サービス・リクエストを発行してください。

関連項目： Oracle Application Server のリリース・ノート。Oracle Technology Network（次の Web サイト）で入手できます。

`http://www.oracle.com/technology/documentation/index.html`

用語集

Advanced Encryption Standard

米国標準技術局（NIST）によって承認されている、DES にかわる暗号化アルゴリズム。AES 規格は米国連邦情報処理標準の公告 197 で入手できる。AES アルゴリズムは対称型のブロック暗号で、128、192 および 256 ビットの暗号鍵を使用して、128 ビットのデータ・ブロックを処理できる。

AES

「**Advanced Encryption Standard**」を参照。

CA

「**認証局**」を参照。

CRL

「**証明書失効リスト**」を参照。

CRL DP

「**CRL 配布ポイント**」を参照。

CRL 配布ポイント（CRL Distribution Point）

X.509 バージョン 3 証明書標準で指定されているオプションの拡張機能。証明書の失効情報が格納されているパーティション化された CRL の場所を示す。通常、この拡張機能の値は URL 形式で示される。CRL 配布ポイント（CRL DP）によって、1 つの**認証局**ドメインに属する失効情報が複数の CRL に配布可能になる。CRL DP では、失効情報がより管理しやすい断片に分割されることによって、巨大な CRL が急増する事態を防げるため、パフォーマンスの向上が見込める。たとえば、証明書で CRL DP を指定することによって、証明書の失効情報がダウンロード可能な Web サーバー上のファイルを参照することができる。

DES

「**データ暗号化規格**」を参照。

Diffie-Hellman 鍵交換アルゴリズム（Diffie-Hellman key negotiation algorithm）

非保護チャネルで通信を行う二者間で、当事者だけにしかわからない乱数を取り決める方法。Diffie-Hellman 鍵交換アルゴリズムの実行中は、当事者が非保護チャネルで情報を交換しても、攻撃者がネットワーク通信を分析し、当事者の間で取り決めた乱数を計算によって推定するのはほぼ不可能である。Oracle Advanced Security では、セッション鍵の生成に Diffie-Hellman 鍵交換アルゴリズムが使用されている。

FIPS

「**米国連邦情報処理標準**」を参照。

HTTP

Hypertext Transfer Protocol。メッセージを書式化して送信し、各種コマンドへのレスポンスのために Web サーバーとブラウザで実行する必要がある処理を決定するために Web で使用される、基礎となる形式。HTTP は Oracle Application Server とクライアントの間で使用されるプロトコルである。

HTTPS

セキュアな Hypertext Transfer Protocol。標準の **HTTP** アプリケーション・レイヤーのサブレイヤーとして **Secure Sockets Layer（SSL）**を使用するプロトコル。ユーザーのページ・リクエストおよびオリジナル・サーバーによって戻されたページを暗号化したり復号化したりする。

HTTP サーバー（HTTP Server）

リモート・ブラウザからの HTTP リクエストを受信し、要求された URL をファイル名に変換して、リクエスト元にファイルを返す**サーバー**。

ID（identity）

エンティティに対する公開鍵とその他の公開情報の組合せ。公開情報には、電子メール・アドレスなどのユーザー認証データが含まれる。宣言どおりのエンティティとして証明されているユーザー。

ID 管理（identity management）

オンライン・エンティティ（デジタル・エンティティ）を作成、管理および使用すること。ID 管理には作成（デジタル ID のプロビジョニング）から、メンテナンス（電子リソースへのアクセスに関する企業ポリシーの強制）、終了までの、デジタル ID のライフ・サイクル全般におけるセキュアな管理が関係する。

ID 管理レルム（identity management realm）

Oracle Internet Directory のサブツリーで、**Oracle コンテキスト**に加えて、ユーザーおよびグループ用の追加サブツリーが含まれる。各サブツリーはアクセス制御リストで保護される。

IIOP

Internet Inter-ORB Protocol。CORBA オブジェクトが相互に通信するために使用するインターネット転送プロトコル。Oracle Application Server のコンテキストでは、IIOP は Java および EJB ベースのアプリケーションで使用される。また、Oracle Application Server のコンポーネント間でも使用される。

Java Database Connectivity（JDBC）

Java プログラムからリレーショナル・データベースに接続できるようにするために、Sun 社によって定義された業界標準の Java インタフェース。

JDBC

「**Java Database Connectivity（JDBC）**」を参照。

Kerberos

分散環境のセキュリティ強化を図るためにマサチューセッツ工科大学の Athena プロジェクトで開発されたネットワーク認証サービス。Kerberos は共有シークレットに依存し、第三者の安全性を前提とした信頼度の高い第三者認証システムである。Kerberos には、**シングル・サインオン**機能とデータベース・リンク認証機能（MIT Kerberos のみ）があり、パスワードを集中的に保管できるため、PC のセキュリティを向上できる。

LDAP

「**Lightweight Directory Access Protocol（LDAP）**」を参照。

ldap.ora ファイル（ldap.ora file）

ディレクトリ・サーバーへのアクセスに関する次の情報を含むファイル。

- ディレクトリ・サーバーの種類
- ディレクトリ・サーバーの位置
- クライアントまたはサーバーが使用するデフォルトの ID 管理レルムまたは Oracle コンテキスト（ポートを含む）

Lightweight Directory Access Protocol（LDAP）

標準的で拡張可能なディレクトリ・アクセス・プロトコル。LDAP クライアントとサーバーが通信に使用する共通言語。業界標準のディレクトリ製品（Oracle Internet Directory など）をサポートする設計規則のフレームワーク。

NIST

「**米国標準技術局**」を参照。

Oracle PKI 証明書使用（Oracle PKI certificate usage）

証明書に含まれる鍵の目的を定義する。Oracle PKI 証明書使用は、X.509 バージョン 3 の標準に定義されている鍵の使用方法に基づく。

Oracle コンテキスト（Oracle Context）

LDAP 準拠のインターネット・ディレクトリのエントリの 1 つで、`cn=OracleContext` として参照される。このエントリの下には、チェックサム・セキュリティ用のエントリを含む、Oracle ソフトウェア関連のあらゆる情報が格納される。

ディレクトリには、1 つ以上の Oracle コンテキストを設定できる。通常、Oracle コンテキストは **ID 管理レルム**に配置される。

PCMCIA カード（PCMCIA card）

Personal Computer Memory Card International Association（PCMCIA）標準に準拠する、クレジット・カード・サイズの小さなコンピュータ・デバイス。PC カードとも呼ばれ、メモリーやモデムを追加したり、ハードウェア・セキュリティ・モジュールとして使用される。ハードウェア・セキュリティ・モジュールとして使用される PCMCIA カードには、**公開鍵と秘密鍵のペア**の秘密鍵コンポーネントが安全に格納される。暗号化操作を実行できるカードもある。

PEM

Privacy-Enhanced Mail プロトコル規格の略。Internet Architecture Board によって採用されており、インターネット上でのセキュアな電子メール通信が可能になる。PEM プロトコルは、暗号化、認証、メッセージ整合性および鍵管理を保証する。PEM は包括的な規格で、データ暗号化鍵を暗号化する各種の鍵管理アプローチ（対称型メソッドと公開鍵メソッドの両方を含む）との互換性が図られている。PEM の仕様は、Internet Engineering Task Force（IETF）の 4 つのドキュメント、RFC 1421、1422、1423 および 1424 で規定されている。

PKCS #10

RSA Security 社の Public-Key Cryptography Standards（PKCS）仕様。証明書リクエストの構文について規定されている。**証明書リクエスト**は、識別名、公開鍵および一連のオプション属性からなり、証明をリクエストするエンティティによって、一括に署名される。

PKCS #11

RSA Security 社の Public-Key Cryptography Standards（PKCS）仕様。暗号化情報を保持して暗号化操作を実行するハードウェア・デバイスに対する、Cryptoki という名前の Application Program Interface（API）が定義されている。「**PCMCIA カード**」も参照。

PKCS #12

RSA Security 社の Public-Key Cryptography Standards（PKCS）仕様。個人的な認証資格証明を、通常 **Wallet** と呼ばれる形式で保管および転送する際の転送構文が規定されている。

PKI

「**公開鍵インフラストラクチャ**」を参照。

Secure Sockets Layer（SSL）

ネットワーク接続を保護するために Netscape 社が設計した業界標準のプロトコル。SSL は、公開鍵インフラストラクチャ（PKI）を使用して認証、暗号化およびデータ整合性を提供する。

Sniffer

ネットワークからプライベート・データ通信を不正に傍受または取得するために使用されるデバイス。

SSL

「**Secure Sockets Layer（SSL）**」を参照。

SSO

「**シングル・サインオン**」を参照。

Wallet

Wallet とは、個々のエンティティに対するセキュリティ資格証明の格納と管理に使用されるデータ構造である。**Wallet Resource Locator（WRL）**は、Wallet の位置を特定するために必要なすべての情報を提供する。

Wallet Resource Locator（WRL）

Wallet の位置を特定するために必要なすべての情報を提供する。Wallet の保存場所であるオペレーティング・システムのディレクトリへのパスである。

Wallet の不明瞭化（wallet obfuscation）

アクセスを許可する前にユーザーにパスワードを要求しなくても、Oracle **Wallet** を格納およびアクセスできるようにするために使用される（**シングル・サインオン**をサポートする）。

Windows ネイティブ認証（Windows native authentication）

Windows サーバーとそのサーバー上で動作するデータベースに対し、クライアントのシングル・ログイン・アクセスを可能にする**認証方式**。

WRL

「**Wallet Resource Locator（WRL）**」を参照。

X.509

デジタル**証明書**用の業界標準仕様。

アクセス制御（access control）

特定のクライアントまたはクライアントのグループに対して、特定データへのアクセス権限を付与または制限するシステムの機能。

アクセス制御リスト（Access Control List: ACL）

管理者が定義するアクセス権に関する一連のディレクティブ。このディレクティブによって、特定のクライアントまたはクライアントのグループ、あるいはその両方に対して、特定データへのアクセス権のレベルを付与する。

暗号化（cryptography）

データのエンコードとデコードを行って、セキュアなメッセージを生成する操作。

暗号化（encryption）

メッセージの内容を、予定された受信者以外の第三者が読むことのできないフォーマットに変換する処理。

暗号化テキスト（encrypted text）

暗号化アルゴリズムを使用して暗号化されたテキストで、暗号化処理の出力ストリームのこと。暗号化テキストは、最初に**復号化**されないかぎり、文面を見ても読取り不能で判読できない。**暗号文**とも呼ばれる。暗号化テキストは、元となる**平文**から作成されている。

暗号スイート（cipher suite）

ネットワークのノード間でメッセージを交換するのに使用される認証、暗号化およびデータ整合性のアルゴリズムのセット。たとえば、SSL ハンドシェイク時には、2 つのノード間でネゴシエーションして、メッセージを送受信するときに使用する暗号スイートを確認する。

暗号スイート名（cipher suite name）

暗号スイートとは、特定のセッションの接続で使用される暗号化保護の種類を表す。

暗号ブロック連鎖（Cipher Block Chaining: CBC）

暗号化方式の 1 つ。先行するすべてのブロックに依存した暗号ブロックの暗号化を行うことにより、ブロック再生攻撃からデータを保護する。この方式は、許可されていない復号化が段階的により困難になるように設計されている。Oracle Advanced Security では、外部暗号ブロック連鎖が使用されている。内部暗号ブロック連鎖よりもセキュアで、実質的なパフォーマンスが低下しないためである。

暗号文（ciphertext）

暗号化されたメッセージ・テキスト。

インスタンス（instance）

アプリケーション・サーバーのインストール内で構成されたコンポーネントの実行に必要な一連のプロセス。アプリケーション・サーバー・インスタンスは、アプリケーション・サーバー・インストールに 1 つずつしか存在しない。インストールとインスタンスという用語は同義で使用されることもあるが、インストールは Oracle ホームにインストールされたファイルのセットであるのに対し、インスタンスはこれらのファイルに関連付けられたプロセスのセットを意味する。

インフラストラクチャ・サービス（infrastructure service）

アプリケーションのデプロイの効率化を図るよう設計された包括的なデプロイ・プラットフォーム。すべてのアプリケーションに対する単一のセキュリティ、ディレクトリおよび製品メタデータ・フレームワークを提供する。フレームワークには、Oracle Identity Management や OracleAS Metadata Repository などのコンポーネントが含まれる。

エントリ（entry）

ディレクトリ・サービスのコンテキストでは、エントリはディレクトリのビルディング・ブロックを指す。エントリは、ディレクトリ内のオブジェクトに関する情報の集まりである。各エントリは、オブジェクトのある 1 つの特徴を表す属性のセットで構成される。たとえば、ディレクトリ・エントリが人物を示す場合、エントリには姓、名、電話番号、電子メール・アドレスなどの属性が含まれる。

オブジェクト・クラス（object class）

名前を持った**属性**のグループ。属性をエントリに割り当てるときは、その属性を保持しているオブジェクト・クラスをそのエントリに割り当てる。同じオブジェクト・クラスに関連付けられているオブジェクトはすべて、そのオブジェクト・クラスの属性を共有する。

介在者（man-in-the-middle）

第三者によるメッセージの不正傍受などのセキュリティ攻撃。第三者、つまり介在者は、メッセージを復号化して再暗号化し（元のメッセージを変更する場合と変更しない場合がある）、元のメッセージの宛先である受信者に転送する。これらの処理はすべて、正当な送受信者が気付かないうちに行われる。この種のセキュリティ攻撃は、**認証**が行われていない場合にのみ発生する。

外部認証（external authentication）

Kerberos などの第三者の認証サービスによって、ユーザー ID を確認すること。

鍵（key）

データの暗号化時に、指定したアルゴリズムによって指定した平文から生成される**暗号文**を決定する値。また、データの復号化時に、暗号文を正しく復号化するために必要な値。暗号文は、正しい鍵が提供された場合にのみ正しく復号化される。

対称型暗号化アルゴリズムでは、同一データの暗号化と復号化の両方に同じ鍵が使用される。非対称型暗号化アルゴリズム（公開鍵暗号化アルゴリズムまたは公開鍵暗号方式とも呼ばれる）では、同一データの暗号化と復号化に異なる鍵が使用される。

鍵のペア（key pair）

公開鍵とそれに対応する**秘密鍵**のペア。「**公開鍵と秘密鍵のペア**」を参照。

管理 OC4J インスタンス（administration OC4J instance）

アクティブな `ascontrol` アプリケーションのデプロイに使用される OC4J インスタンス。

機密保護（confidentiality）

暗号化の機能の 1 つ。機密保護では、メッセージの予定された受信者のみがメッセージを参照（暗号文を復号化）できることが保証される。

クライアント（client）

サービス、データ、他のアプリケーションやコンピュータ（**サーバー**）の処理などを要求するユーザー、ソフトウェア・アプリケーション（ブラウザなど）またはコンピュータである。クライアントはサービスに依存する。

クラスタ（cluster）

複数の接続された Oracle Application Server インスタンス、およびこれらのアプリケーション・サーバー内の OC4J インスタンスにデプロイされたアプリケーション。

クリアテキスト（cleartext）

暗号化されていない平文。

グリッド・コンピューティング（grid computing）

多数のサーバーとストレージを 1 つの巨大なコンピュータとして動作するように調整するコンピューティング・アーキテクチャ。Oracle Grid Computing は、企業のあらゆるコンピューティング・ニーズに対応できる、柔軟性に優れたオンデマンド・コンピューティング・リソースを実現する。Oracle 10*g* グリッド・コンピューティング・インフラストラクチャ上で動作するアプリケーションは、フェイルオーバー、ソフトウェア・プロビジョニングおよび管理において、共通のインフラストラクチャ・サービスを利用できる。Oracle Grid Computing では、リソースに対する需要が分析され、それに応じた供給の調整が行われる。

グループ（group）

同じクラスタ・トポロジに属する OC4J インスタンスの集合。グループ内で実行されているすべての OC4J インスタンスに対する構成操作の同時実行が可能。

公開鍵（public key）

公開鍵暗号化における一般に公開される鍵。主に暗号化に使用されるが、署名の確認にも使用できる。「**公開鍵と秘密鍵のペア**」を参照。

公開鍵暗号化（public key encryption）

メッセージの送信側が受信側の公開鍵でメッセージを暗号化する処理。配信されたメッセージは、受信側の秘密鍵で復号化される。

公開鍵インフラストラクチャ（public key infrastructure: PKI）

公開鍵暗号化の原理を利用する情報セキュリティ・テクノロジ。公開鍵暗号化では、共有されている公開鍵と秘密鍵のペアを使用して情報の暗号化と復号化を行う。パブリック・ネットワークでのセキュアでプライベートな通信を可能にする。

公開鍵と秘密鍵のペア（public and private key pair）

暗号化と**復号化**に使用される 2 つの数字のセット。1 つは**秘密鍵**、もう 1 つは**公開鍵**と呼ばれる。公開鍵は通常広く使用可能であるのに対して、秘密鍵はその各所有者によって保有される。2 つの数字は関連付けられているが、公開鍵から秘密鍵を算出することは一般的にほぼ不可能である。公開鍵と秘密鍵は、非対称型暗号化アルゴリズム（公開鍵暗号化アルゴリズムまたは公開鍵暗号方式とも呼ばれる）でのみ使用される。**鍵のペア**の公開鍵または秘密鍵のどちらかで暗号化されたデータは、鍵のペアの関連する鍵で復号化できる。ただし、公開鍵で暗号化されたデータを同じ公開鍵では復号化できず、秘密鍵で暗号化されたデータを同じ秘密鍵では復号化できない。

サーバー（server）

1. Oracle Application Server で、分散およびオブジェクト指向のアプリケーションに対して、スケーラブル、堅牢、セキュアおよび拡張可能なプラットフォームを提供するミドルウェア・サービスとツールのコレクション。Oracle Application Server により、Java と J2EE、Web サービス、XML、SQL および PL/SQL の単一統合プラットフォームが実現される。2. Oracle Database Server で、利用可能なインタフェースを任意の数だけ使用して、クライアントのためにデータ管理作業を実行するための専用リレーショナル・データベース・サーバー。

サービス（service）

1. クライアントが使用するネットワーク・リソースで、Oracle Application Server や Oracle データベース・サーバーなどがある。2. Windows **レジストリ**にインストールされ、Windows で管理される実行可能プロセス。作成されて起動されたサービスは、ユーザーがコンピュータにログインしていなくても実行できる。

サービス名（service name）

クライアントの観点から見たデータベースで、データベースの論理的な表現。データベースは複数のサービスで構成可能で、このサービスも複数のデータベース・インスタンスとして実装可能である。サービス名とは、グローバル・データベース名を表す文字列である。つまり、インストール時またはデータベース作成時に入力する、データベース名とドメイン名で構成されている。

資格証明（credentials）

Oracle Database、Oracle Application Server や Oracle Identity Management インフラストラクチャのアクセスで使用するユーザー名、パスワードまたは証明書。

識別名（distinguished name: DN）

LDAP ベースのディレクトリ・エントリの一意の名前。識別名は、親エントリの個々の名前がすべて、下からルートに向かって順に結合された形で構成されている。

システム識別子（system identifier: SID）

Oracle インスタンスの一意な名前。Oracle データベースを切り替える場合は、該当する SID を指定する必要がある。SID は tnsnames.ora ファイル内の**接続記述子**の `CONNECT DATA` 部分と listener.ora ファイル内の**ネットワーク・リスナー**の定義に含まれている。

自動ログイン Wallet（auto login wallet）

アクセス時に資格証明を提示することなく、サービスに対する PKI またはパスワード・ベースのアクセスを実現する Oracle Wallet Manager の機能。この自動ログイン・アクセスは、その Wallet の自動ログイン機能が無効になるまで、有効である。ファイル・システム権限では、Wallet の自動ログインに必要なセキュリティが提供される。Wallet への自動ログインが有効な場合でも、その Wallet に対して読取り権限のあるオペレーティング・システム・ユーザーのみが使用できる。これらはシングル・サインオン機能を持つため、SSO Wallet と呼ばれることもある。

証明書（certificate）

公開鍵に対して ID を安全にバインドする ITU X.509 バージョン 3 の標準データ構造。

エンティティの公開鍵に信頼できる機関、つまり認証局が署名するとき、証明書が作成される。証明書によって、エンティティの情報が正しいこと、および公開鍵が実際にそのエンティティに属していることが保証される。

証明書には、エンティティの名前、ID および公開鍵が記載されている。また、シリアル番号、有効期限、ならびにその証明書に関連する権利、使用および権限についての情報が記載されていることもある。最後に、発行元である認証局に関する情報が記載されている。

証明書失効リスト（certificate revocation list）

失効した**証明書**のリストを含む署名付きデータ構造。証明書失効リスト（CRL）の信頼性と整合性は、添付されたデジタル署名によって保証される。通常、CRL の署名者は、発行済の証明書に署名したエンティティと同じである。

証明書リクエスト（certificate request）

リクエストは、証明書リクエスト情報、署名アルゴリズムの識別子、および証明書リクエスト情報のデジタル署名の 3 つの部分で構成される。証明書リクエスト情報は、リクエスト対象の識別名、公開鍵および一連のオプション属性からなる。属性では、リクエスト対象の ID に関する追加情報（郵便宛先など）や、その対象エンティティが後に証明書の失効化をリクエストする際のチャレンジ・パスワードなどが指定される。「**PKCS #10**」を参照。

証明連鎖（certificate chain）

エンド・ユーザーまたはサブスクライバの証明書とその認証局の証明書を含む、順序付けられた証明書のリスト。

シングル・サインオン（single sign-on: SSO）

ユーザーが一度認証を受けると、その後の他のデータベースやアプリケーションへの接続時には厳密な認証が透過的に発生する機能のこと。シングル・サインオンにより、ユーザーは 1 回の接続時に入力した 1 つのパスワードで、複数のアカウントおよびアプリケーションにアクセスできるようになる。単一パスワードによる単一認証を指す。

信頼できる証明書（trusted certificate）

信頼できる証明書は、一定の信頼度を有すると認定された第三者の ID であり、ルート鍵証明書とも呼ばれる。信頼できる証明書は、エンティティが本人であるという ID の確認が行われるときに使用される。通常は、信頼する認証局のことを信頼できる証明書という。複数レベルの信頼できる証明書がある場合、証明連鎖における下位レベルの信頼できる証明書は、それより上のレベルの証明書をすべて再検証する必要はない。

信頼できる認証局（trusted certificate authority）

「**認証局**」を参照。

スキーマ（schema）

1. データベース・スキーマ : 表、ビュー、クラスタ、プロシージャ、パッケージ、属性、オブジェクト・クラスなどのオブジェクトの集合に名前を付けたもの。それらのオブジェクトに対応する一致ルールは、特定のユーザーに関連付けられている。2. **LDAP** ディレクトリ・スキーマ : 属性、オブジェクト・クラスおよびそれらに対応する一致ルールの集合。

スマート・カード（smart card）

ユーザー名やパスワードなどの情報を格納するため、また認証交換に関連する計算を実行するための IC が組み込まれた（クレジット・カードに似た）プラスティック製のカード。スマート・カードはクライアントまたはサーバーにあるハードウェア・デバイスで読み取る。

スマート・カードは、ワンタイム・パスワードとして使用できる乱数を生成できる。この場合、スマート・カードはサーバー上のサービスと同期化されているので、サーバーはスマート・カードによって生成されるパスワードと同じパスワードを要求する。

制限（restriction）

セキュリティ・スキームの 1 つで、サーバーが提供するファイルへのアクセス権を、IP アドレスまたは DNS ドメインの一部のグループ内のクライアント・マシンに制限する。

整合性（integrity）

受信メッセージの内容が、送信時の元のメッセージの内容から変更されていないことを保証すること。

セッション鍵（session key）

少なくとも二者間（通常はクライアントとサーバー）で共有され、単一の通信セッション継続中のデータ暗号化に使用される鍵。セッション鍵は通常、ネットワーク通信を暗号化するのに使用される。クライアントとサーバーは、セッションの開始時に使用するセッション鍵を取り決めることができる。セッション継続中は、関係者間の全ネットワーク通信の暗号化にその鍵が使用される。クライアントとサーバーが新しいセッションで再び通信する場合は、新しいセッション鍵を取り決める。

接続記述子（connect descriptor）

ネットワーク接続の宛先を指定するために特別に書式化された記述。接続記述子には、宛先**サービス**とネットワーク・ルート情報が含まれる。宛先サービスは、Oracle データベースのサービス名を使用して示される。ネットワーク・ルートでは、最低でも**リスナー**の位置がネットワーク・アドレスを使用して示される。「**接続識別子**」を参照。

接続識別子（connect identifier）

接続記述子または接続記述子にマッピングされた名前。接続識別子は、**ネット・サービス名**、データベース・**サービス名**または**ネット・サービス別名**である。ユーザーは、次に示すように、ユーザー名およびパスワードとともに、接続を希望するサービスの接続文字列内にある接続識別子を渡すことにより、接続リクエストを開始する。

```
CONNECT username/password@connect_identifier
```

接続文字列（connect string）

特定のデータベース・インスタンスに接続するためにユーザーが**サービス**に渡す情報。情報には、ユーザー名、パスワードおよび**ネット・サービス名**を含めることができる。次に例を示す。

```
CONNECT username/password@net_service_name
```

属性（attribute）

LDAP ディレクトリにおいて、エントリの性質を説明する断片的な情報項目。1 つのエントリは一連の属性から構成され、それぞれが**オブジェクト・クラス**に所属する。さらに、各属性にはタイプと値があり、タイプは属性の情報の種類を説明するものであり、値には実際のデータが格納されている。

単一鍵ペア Wallet（single key-pair wallet）

単一のユーザー**証明書**とその関連する**秘密鍵**が含まれる **PKCS #12** 形式の **Wallet**。**公開鍵**は証明書に埋め込まれている。

中間層（middle tier）

3 層アーキテクチャでは、中間層はアプリケーション・ロジック・レイヤーである。中間層では、クライアントに計算力とリソースが提供される。Oracle Application Server では、中間層は Oracle HTTP Server、OC4J、OPMN などのコンポーネントで構成されている。

データ暗号化規格（Data Encryption Standard: DES）

米国のデータ暗号化規格。

データ・ディクショナリ（data dictionary）

データベースに関する情報を提供する一連の読取り専用の表。

データベース別名（database alias）

「**ネット・サービス名**」を参照。

ディレクトリ・ネーミング（directory naming）

中央ディレクトリ・サーバーに格納されているデータベース・サービス、**ネット・サービス名**または**ネット・サービス別名**を**接続記述子**に変換するネーミング・メソッド。

ディレクトリ・ネーミング・コンテキスト（directory naming context）

ディレクトリ・サーバー内で重要なサブツリー。通常、ディレクトリ・ネーミング・コンテキストは、組織サブツリーの最上部となっている。一部のディレクトリでは、固定のコンテキストのみが可能である。また、別のディレクトリでは、ディレクトリ管理者による構成の対象をゼロから多数までとすることができる。

デジタル署名（digital signature）

デジタル署名は、公開鍵アルゴリズムを使用して、送信側の秘密鍵で送信側のメッセージに署名すると作成される。デジタル署名によって、文書が信頼できるものであること、別のエンティティによって偽造されていないこと、変更されていないこと、送信者によって否認されないことが保証される。

トークン・カード（token card）

ユーザーが容易に認証サービスを利用できるように、数種類のメカニズムを提供するデバイス。一部のトークン・カードは、認証サービスと同期化されているワンタイム・パスワードを提供する。サーバーは認証サービスと連絡を取ることによって、トークン・カードが提供するパスワードをいつでも検証できる。チャレンジ / レスポンスに基づいて機能するトークン・カードもある。この場合は、サーバーがチャレンジ（番号）を提供し、ユーザーがその番号をトークン・カードに入力する。そして、トークン・カードは別の番号（チャレンジから暗号的に導出される番号）を提供し、それをユーザーがサーバーに渡す。

ドメイン（domain）

ドメイン・ネーム・システムのネームスペース内の任意のツリーまたはサブツリー。ホスト名が共通の接尾辞、つまりドメイン名を共有しているコンピュータのグループを指す。

ドメイン・ネーム・システム（Domain Name System: DNS）

コンピュータやネットワーク・サービスのネーミング・システムであり、**ドメイン**を階層的に編成している。DNS は、ユーザー・フレンドリな名前でコンピュータの位置を識別するために TCP/IP ネットワークで使用される。DNS は、ユーザー・フレンドリな名前をコンピュータが理解できる IP アドレスに変換する。

トラスト・ポイント（trust point）

「**信頼できる証明書**」を参照。

認可（authorization）

メッセージを送信したりリクエストを行うためのセキュリティ制約の評価。認可では特定の基準を使用して、リクエストを許可する必要があるかどうかを決定する。この基準が**認証**および制限である。

認証（authentication）

ユーザー、デバイスまたはコンピュータ・システム内のその他のエンティティの ID が正当なものであるかを確認するプロセス。多くの場合、システム内のリソースへのアクセス権を付与する際に、このプロセスを事前に行う必要がある。認証済メッセージの受信者には、そのメッセージの発信元（送信者）がわかる。認証は、第三者が送信者のふりをする可能性を排除する目的で行われる。

認証局（certificate authority）

ユーザー、データベース、管理者、クライアント、サーバーなどが本人（本物）であることを証明する、信頼できる第三者。ユーザーを認証するときには、認証局はまずそのユーザーが証明書失効リスト（CRL）に記載されていないかを確認する。次に、ユーザーの ID が正当なものであるかを確認してから証明書を与え、認証局の秘密鍵を付けて署名する。認証局には、自局で発行する独自の証明書と公開鍵がある。サーバーおよびクライアントはこの証明書と公開鍵を使用して、認証局が作成した署名が正当なものであるかを確認する。認証局は、証明書サービスを提供する外部の企業であったり、社内の MIS 部門のような内部組織である場合もある。

認証方式（authentication method）

分散環境におけるユーザー、クライアントおよびサーバーの ID を確認するセキュリティ方式。ネットワーク認証方式を利用すると、ユーザーは**シングル・サインオン**の恩恵を受けることもできる。次の認証方式が Oracle Application Server でサポートされる。

- **Kerberos**
- **Secure Sockets Layer（SSL）**
- **Windows ネイティブ認証**

ネット・サービス別名（net service alias）

ディレクトリ・サーバー内の**ディレクトリ・ネーミング**・オブジェクトの代替名。ディレクトリ・サーバーには、任意の定義済**ネット・サービス名**またはデータベース・サービスのネット・サービス別名が格納される。ネット・サービス別名のエントリは、接続記述子情報を持たない。かわりに、ネット・サービス別名は、元となるオブジェクトの位置のみを参照する。クライアントがディレクトリ内にあるネット・サービス別名のルックアップをリクエストすると、そのエントリがネット・サービス別名であることが認識され、その別名が参照している実際のエントリであるかのようにルックアップが実行される。

ネット・サービス名（net service name）

データベース・サーバーを識別する目的でクライアントが使用する名前。ネット・サービス名は、ポート番号とプロトコルにマッピングされる。**接続文字列**または**データベース別名**とも呼ばれる。

ネットワーク認証サービス（network authentication service）

分散環境における認証方法の 1 つ。サーバーに対するクライアントの認証、サーバー間の認証、およびクライアントとサーバーの両方に対するユーザーの認証を行う。ネットワーク認証サービスは、ユーザーやユーザーがアクセスする様々なサーバー上のサービスに関する情報に加えて、ネットワーク上のクライアントとサービスに関する情報を格納するリポジトリである。認証サーバーは物理的に別々のマシンである場合や、システム内の別のサーバー上に共存する場合がある。単一の障害ポイントを避け、可用性を確保するために、レプリケートが行われる認証サービスもある。

ネットワーク・リスナー（network listener）

サーバー上にあるリスナー。1 つ以上のプロトコルで、1 つ以上のデータベースに対する接続リクエストをリスニングする。「**リスナー**」を参照。

ピアの ID（peer identity）

SSL 接続セッションは、特定のクライアントとサーバーとの間で確立される。ピアの ID は、セッションの開始時に確定される。ピアは **X.509 証明連鎖**によって識別される。

否認防止（non-repudiation）

メッセージの発信元、配信、送信または転送に関する明白な証明。

平文（plaintext）

暗号化されていないメッセージ・テキスト。

秘密鍵（private key）

公開鍵暗号化における秘密鍵。主に復号化に使用されるが、デジタル署名とともに暗号化にも使用される。「**公開鍵と秘密鍵のペア**」を参照。

フェイルオーバー（failover）

コンポーネントに障害が発生したときに、類似するアクティブな代替コンポーネントを利用して、コンピューティング・システムを再構成する機能。

復号化（decryption）

暗号化されたメッセージの内容（暗号文）を、元の読取り可能なフォーマット（平文）に戻す変換処理。

不明瞭化（obfuscation）

情報を判読不能な形式にスクランブルすること。スクランブルに使用したアルゴリズムが不明の場合は、スクランブル解除が非常に困難になる。

プロキシ認証（proxy authentication）

ファイアウォールなどの中間層がある環境で一般的に採用されている処理。エンド・ユーザーは中間層で認証され、次に中間層がユーザーにかわって、つまりプロキシとしてディレクトリで認証される。中間層はディレクトリにプロキシ・ユーザーとしてログインする。プロキシ・ユーザーは ID を切り替えることができるため、ディレクトリにログインした後は、エンド・ユーザーの ID に切り替える。プロキシ・ユーザーは、元のエンド・ユーザーに適用されている認可を使用して、エンド・ユーザーのかわりに操作を実行できる。

ベース（base）

LDAP 準拠ディレクトリにおけるサブツリー検索のルート。

米国標準技術局（National Institute of Standards and Technology: NIST）

米国商務省に属する政府機関。コンピュータ・システムや通信システムにおける暗号化ベースのセキュリティ・システムの設計、取得および実装に関連するセキュリティ標準の開発を担当する。この標準は、政府業務の遂行に必要な情報処理を連邦政府にかわって行う連邦政府機関、連邦政府機関の契約機関またはその他の組織によって運営されているシステムに適用される。

米国連邦情報処理標準（Federal Information Processing Standard: FIPS）

暗号化モジュールのセキュリティ要件を定義する米国連邦政府の標準規格。コンピュータ・システムや通信システム内の非機密情報を保護するセキュリティ・システムで使用される。**米国標準技術局**によって公開されている。

メッセージ・ダイジェスト（message digest）

テキストを 1 桁の文字列として表したもの。メッセージを 1 つの数字文字列に変換するアルゴリズムである、一方向ハッシュ関数という計算式を使用して作成される。一方向というのは、数字文字列から元のメッセージを生成するのがほぼ不可能であることを意味する。計算済のメッセージ・ダイジェストは、メッセージが改ざんされていないことを確認するため、**公開鍵**を使用して復号化されたメッセージ・ダイジェストと比較される。

ユーザー検索ベース（user search base）

LDAP ディレクトリ内のユーザーが配置されているノード。

リスナー（listener）

サーバー上に常駐するプロセス。クライアントの着信接続リクエストをリスニングし、サーバーへの通信量を管理する。リスナーとは、着信リクエストを処理し、ディスパッチャにルーティングする HTTP サーバーのことである。

クライアントがサーバーとのネットワーク・セッションをリクエストするたびに、リスナーは実際のリクエストを受信する。クライアントの情報がリスナーの情報と一致すると、リスナーはサーバーへの接続を認める。

リモート OC4J インスタンス（remote OC4J instance）

管理 OC4J インスタンス以外の OC4J インスタンス。

リモート・コンピュータ（remote computer）

ローカル・コンピュータ以外のネットワーク上にあるコンピュータ。

ルート鍵証明書（root key certificate）

「**信頼できる証明書**」を参照。

レジストリ（registry）

コンピュータの構成情報を格納する Windows のリポジトリ。

レルム（realm）

1. **ID 管理レルム**の短縮名。2. **Kerberos** オブジェクト。1 つの鍵配布センター / チケット認可サービス（KDC/TGS）の下で稼動するクライアントとサーバーのセット。名前が同じでも異なるレルムにあるサービスは一意である。

レルムの Oracle コンテキスト（realm Oracle Context）

Oracle Internet Directory の **ID 管理レルム**の一部である **Oracle コンテキスト**。

索引

数字

A

B

C

D

E

F

G

H

P

R

S

T

U

W

X

あ

か

さ

た

な

は

ま

や

ら

わ