



BEA WebLogic Platform™

リリース ノート

バージョン 7.0 Service Pack 5
マニュアルの日付: 2004 年 4 月 8 日

著作権

Copyright © 2004, BEA Systems, Inc. All Rights Reserved.

限定的権利条項

本ソフトウェアおよびマニュアルは、BEA Systems, Inc. 又は日本ビー・イー・エー・システムズ株式会社（以下、「BEA」といいます）の使用許諾契約に基づいて提供され、その内容に同意する場合にのみ使用することができ、同契約の条項通りにのみ使用またはコピーすることができます。同契約で明示的に許可されている以外の方法で同ソフトウェアをコピーすることは法律に違反します。このマニュアルの一部または全部を、BEA からの書面による事前の同意なしに、複製、複製、翻訳、あるいはいかなる電子媒体または機械可読形式への変換も行うことはできません。

米国政府による使用、複製もしくは開示は、BEA の使用許諾契約、および FAR 52.227-19 の「Commercial Computer Software-Restricted Rights」条項のサブパラグラフ (c)(1)、DFARS 252.227-7013 の「Rights in Technical Data and Computer Software」条項のサブパラグラフ (c)(1)(ii)、NASA FAR 補遺 16-52.227-86 の「Commercial Computer Software--Licensing」条項のサブパラグラフ (d)、もしくはそれらと同等の条項で定める制限の対象となります。

このマニュアルに記載されている内容は予告なく変更されることがあり、また BEA による責務を意味するものではありません。本ソフトウェアおよびマニュアルは「現状のまま」提供され、商品性や特定用途への適合性を始めとする（ただし、これらには限定されない）いかなる種類の保証も与えません。さらに、BEA は、正当性、正確さ、信頼性などについて、本ソフトウェアまたはマニュアルの使用もしくは使用結果に関していかなる確約、保証、あるいは表明も行いません。

商標または登録商標

BEA、Jolt、Tuxedo、および WebLogic は、BEA Systems, Inc. の登録商標です。BEA Builder、BEA Campaign Manager for WebLogic、BEA eLink、BEA Liquid Data for WebLogic、BEA Manager、BEA WebLogic Commerce Server、BEA WebLogic Enterprise、BEA WebLogic Enterprise Platform、BEA WebLogic Express、BEA WebLogic Integration、BEA WebLogic Personalization Server、BEA WebLogic Platform、BEA WebLogic Portal、BEA WebLogic Server、BEA WebLogic Workshop および How Business Becomes E-Business は、BEA Systems, Inc. の商標です。

その他の商標はすべて、関係各社がその権利を有します。

目次

BEA WebLogic Platform の概要	2
このマニュアルで使用される用語	2
BEA WebLogic Platform 7.0 Service Pack 5 の新機能.....	2
新しいデータベース ドライバのサポート : Oracle Thin Driver 10g	3
WebLogic Platform 7.0 の SP1、SP2、SP4 および SP5 の相互運用性.....	4
パッケージアップグレード インストーラ拡張機能	4
WebLogic Platform 7.0 SP5 と共にインストールされる WebLogic Integration アダプ タとプラグイン	5
プラットフォームのサポートおよびシステム要件	5
特殊なインストール手順.....	6
WebLogic JRockit SDK と共に WebLogic Platform を使用する	7
WebLogic Platform と共にバンドルされていない JVM を使用する	7
[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する	8
移行スクリプトを実行する	10
アップグレード中に実行されるアクション	11
アップグレードの例	13
再開の例	14
移行を実行するための手動作業の実行	14
WebLogic Platform 7.0 SP1 に移行するための手動作業の実行	14
WebLogic Platform 7.0 SP2 に移行するための手動作業の実行	22
WebLogic Platform 7.0 SP4 に移行するための手動作業の実行	32
WebLogic Platform 7.0 SP5 に移行するための手動作業の実行	34
新しいインストール環境 (アップグレード環境ではない) でスクリプトおよびコン フィグレーション ファイルを更新する	37
Platform Domain で使用されたデータベースの切り替え	38
手順 1: データベースのコンフィグレーション	38
手順 2: データベース環境に合わせた db_settings.properties の編集	39
手順 3: データベース環境に合わせた setDBVars および setDBVarsExt の編集	40
手順 4: WebLogic Server の起動	42

手順 5: WebLogic Administration Console での接続プールおよびレلمの設定	42
手順 6: WebLogic Server の起動	45
手順 7: db_settings.properties の編集によるデータベースのコメントの解除	45
手順 8: create_db の実行	45
手順 9: WebLogic Server の再起動	45
手順 10: sync の実行	46
手順 11: Oracle のみ — インデックスの再構築	46
ベスト プラクティス	47
BEA ホーム ディレクトリ名の文字制限	47
インストール ディレクトリに名前を付ける	47
WebLogic Portal サンプル データ	48
Platform Domain テンプレートを使用する	48
WebLogic Platform サンプル アプリケーションを実行する	49
Netscape ブラウザ 6.x を使用してドキュメントにアクセスする	50
WebLogic Platform 7.0 SP2、SP4、および SP5 の JVM オプションを指定する	50
サンプル スクリプト	52
修正された問題点	52
Service Pack 1 で修正された問題点	53
Service Pack 2 で修正された問題点	54
Service Pack 4 で修正された問題点	57
Service Pack 5 で修正された問題点 END	58
既知の制約	58
追跡された問題	59
WebLogic JRockit SDK のサポート上の制約	74
JVM コンフィグレーション	74
パフォーマンス上の注意点	75
DB2 の注意点	75
Apache 2.0.43 の注意点	75

BEA WebLogic Platform 7.0 リリース ノート

BEA WebLogic Platform リリース : 7.0 Service Pack 5 マニュアルの日付 : 2004 年 4 月

このマニュアルでは、次のトピックについて説明します。

- BEA WebLogic Platform の概要
- このマニュアルで使用される用語
- BEA WebLogic Platform 7.0 Service Pack 5 の新機能
- プラットフォームのサポートおよびシステム要件
- 特殊なインストール手順
- WebLogic JRockit SDK と共に WebLogic Platform を使用する
- [コンフィグレーション] ウィザードを使用して作成されたドメインを移行する
- Platform Domain で使用されたデータベースの切り替え
- 修正された問題点
- 既知の制約

最新のリリース ノート情報については、次の URL にある BEA マニュアルの Web サイトを参照してください。

<http://edocs.beasys.co.jp/e-docs/index.html>

BEA WebLogic Platform の概要

BEA WebLogic Platform は、BEA WebLogic の既存の製品（アプリケーションサーバ、開発、ポータル、および統合）のすべての機能が高度に統合されたソリューションです。この統合されたソリューションによって、一般的なアプリケーション インフラストラクチャの利点を、堅牢で使いやすいフレームワークで活用できます。

このマニュアルで使用される用語

このマニュアルでは、パス名に次の 2 つの仮の文字列を使用しています。

- *BEA_HOME* は、ライセンス ファイルなど、同じマシンにインストールされた複数の BEA 製品が使用するファイルのリポジトリとして機能するディレクトリです。
BEA_HOME の一般的な値は `c:\bea` ですが、どのディレクトリでも *BEA_HOME* として指定できます。
- *WL_HOME* は、WebLogic Platform ソフトウェアがインストールされるディレクトリです。デフォルトでは、*BEA_HOME* の下に置かれます。一般的なデフォルトのパス名は `c:\bea\weblogic700` ですが、*WL_HOME* はどのディレクトリに置くこともできます。

BEA WebLogic Platform 7.0 Service Pack 5 の新機能

BEA WebLogic Platform 7.0 Service Pack 5 (SP5) には、すべての WebLogic Platform 7.0 コンポーネント (WebLogic Server、WebLogic Workshop、WebLogic Integration、WebLogic Portal および WebLogic JRockit) のサービス パックの更新が収録されています。WebLogic Platform 7.0 SP5 では、すべての WebLogic Platform 7.0 コンポーネントに対してアップデートを行うことができます。

WebLogic Platform 7.0 SP5 には、WebLogic Platform 7.0、7.0 SP1、7.0 SP2、および 7.0 SP4 のメンテナンスが含まれています。さらに、以下の節では、SP5 の新機能について説明します。

- 新しいデータベースドライバのサポート : Oracle Thin Driver 10g
- WebLogic Platform 7.0 の SP1、SP2、SP4 および SP5 の相互運用性
- パッケージアップグレード インストーラ拡張機能
- WebLogic Platform 7.0 SP5 と共にインストールされる WebLogic Integration アダプタとプラグイン

WebLogic Platform 7.0 SP5 および以前の 7.0.x リリースの新しい機能について、完全な一覧を参照するには、次の URL の新機能ページを参照してください。

<http://edocs.beasys.co.jp/e-docs/platform/docs70/interm/whatsnew.html>

新しいデータベース ドライバのサポート : Oracle Thin Driver 10g

WebLogic Platform 7.0 SP5 では、Oracle Thin Driver 10g のサポートが拡張されています。詳細については、次の URL にある WebLogic Platform 7.0 のサポート対象コンフィグレーションの『サポート対象のデータベース コンフィグレーション』を参照してください。

http://edocs.beasys.co.jp/e-docs/platform/suppconfigs/configs70/70_over/supported_db.html

注意： Oracle10g GA ドライバには既知の問題があります。日本語文字でマルチバイト文字セットを使用していると、setString() メソッドに対するコールが失敗します。(Oracle TAR 番号 3584729.994、Oracle バグ番号 3437365):

WebLogic Platform 7.0 SP5 は、Oracle 10g ドライバで認定されています。このドライバはキットにバンドルされ、このパッチを含んでいます。Oracle 10g GA ドライバを使用している場合は、Oracle から関連パッチを入手することをお勧めします。

WebLogic Platform 7.0 の SP1、SP2、SP4 および SP5 の相互運用性

WebLogic Platform 7.0 は、サービス パックが異なっても、相互に運用することができます。たとえば、ドメインやスクリプトを修正しなくても、WebLogic Platform 7.0 を実行するマシンが 7.0 SP1、7.0 SP2、7.0 SP4、または 7.0 SP5 を実行するマシンと対話するように設定できます。

ただし、次の制約を伴います。

- 相互運用性は以下の一般的なコンフィグレーションでサポートされています。

- クラスタ化されていないノード間
- 個別のクラスタに設定されたノード間

同じドメインまたはクラスタに属するサーバはすべて、同じ 7.0.x リリースの WebLogic Platform コンポーネントを実行する必要があります。

- WebLogic Portal に付属する E-Business Control Center (EBCC) クライアントは、EBCC と同じサービス パックを実行するサーバとのみ運用できる。たとえば、WebLogic Portal 7.0 SP1 に付属する EBCC クライアントは、WebLogic Server 7.0 SP1 とは運用できますが、WebLogic Server 7.0 SP2 とは運用できません。

パッケージ アップグレード インストーラ拡張機能

WebLogic Platform 7.0 SP5 パッケージ アップグレード インストーラには、以下の拡張機能が含まれています。

- WebLogic JRockit SDK と共に WebLogic Platform 7.0 の以前のサービス パックを実行している場合、利用可能なアップグレード インストーラを使用して、WebLogic Platform インストールをアップグレードすることができます。以前の 7.0.x リリースでは、完全な WebLogic Platform インストーラを使用する必要があります。インストールのアップグレードの詳細については、次の URL にある『WebLogic Platform のインストール』の「サービス パックとローリング パッチのインストール」を参照してください。

<http://edocs.beasys.co.jp/e-docs/platform/docs70/install/update.html>

プラットフォームのサポートおよびシステム要件

- **WebLogic Platform 7.0 SP5** パッケージアップグレード インストーラは、現在のインストールの **JVM** をパッケージアップグレード インストーラにバンドルされる **JVM** と置き換えます。たとえば、現在のインストールが **Sun JVM** を使用する場合、**WebLogic JRockit JVM** をバンドルするパッケージアップグレード インストーラを使用してインストールをアップグレードすると、インストールは **WebLogic JRockit JVM** を使用するようになります。

WebLogic Platform 7.0 SP5 と共にインストールされる WebLogic Integration アダプタとプラグイン

WebLogic Platform 7.0 SP5 と共に以下の **WebLogic Integration** アダプタとプラグインがインストールされます。

- **RDBMS アダプタ**
- **xFile、電子メール、および HTTP プラグイン**

以前の 7.0.x リリース ではこれらのコンポーネントを入手するには個別にダウンロードしていました。[**コンフィグレーション**] ウィザードを使用して作成された **WebLogic Integration** ドメインに、プラグインはあらかじめデプロイされます。**RDBMS** アダプタは手動でデプロイする必要があります。詳細については、次の URL の『**WebLogic Integration** リリース ノート』を参照してください。

<http://edocs.beasys.co.jp/e-docs/wli/docs70/relnotes/index.htm>

プラットフォームのサポートおよびシステム要件

ハードウェア要件およびソフトウェア要件など、**WebLogic Platform** でサポートされるプラットフォームについては、次の URL の『**WebLogic Platform 7.0** サポート対象コンフィグレーション』を参照してください。

http://edocs.beasys.co.jp/e-docs/platform/suppconfigs/configs70/70_over/overview.html

注意： 2003年12月31日で Red Hat は Linux 7.2 のサポートを終了したため、BEA も同日よりサポートを終了しました。Red Hat Enterprise Linux と WebLogic JRockit をアップグレードすることをお勧めします。

サポートされているすべてのプラットフォーム コンフィグレーションに製品をインストールする手順については、次の URL の『WebLogic Platform のインストール』を参照してください。

<http://edocs.beasys.co.jp/e-docs/platform/docs70/install/index.html>

IBM AIX および SuSE Linux プラットフォームなどの特殊なインストール手順については、次の URL の『WebLogic Platform 7.0 サポート対象プラットフォーム』の「使用するプラットフォームでのインストール手順」を参照してください。

http://edocs.beasys.co.jp/e-docs/platform/suppconfigs/configs70/70_over/install_info.html

特殊なインストール手順

オペレーティング システム コンフィグレーションによっては、特殊なインストール手順が必要な場合があります。WebLogic Platform 7.0 では、以下の オペレーティング システムで特殊なインストールと操作手順が必要です。

- IBM AIX 4.3.3、5.1、および 5.2
- IBM zSeries/S390 上での SuSE Linux Enterprise Server 7 および 8

このインストール手順は、次の URL の『WebLogic Platform 7.0 サポート対象コンフィグレーション』を参照してください。

http://edocs.beasys.co.jp/e-docs/platform/suppconfigs/configs70/70_over/install_info.html

WebLogic JRockit SDK と共に WebLogic Platform を使用する

WebLogic JRockit SDK に付随する以下の使用方法ノートを参照してください。

- WebLogic JRockit を使用するために JVM を切り替える場合、7 ページの「WebLogic Platform と共にバンドルされていない JVM を使用する」で説明しているバンドルされていない JVM への切り替えと同じ手順を使うことができます。
- WebLogic JRockit SDK と共に以前の WebLogic Platform 7.0.x リリースを実行している場合、利用可能なアップグレード インストーラを使用して、WebLogic Platform インストーラをアップグレードすることができます。以前の 7.0.x リリースでは、完全な WebLogic Platform インストーラを使用する必要があります。インストーラのアップグレードの詳細については、次の URL にある『WebLogic Platform のインストール』の「サービス パックとローリング パッチのインストール」を参照してください。
<http://edocs.beasys.co.jp/e-docs/platform/docs70/install/update.html>
- サポートの制約については、74 ページの「WebLogic JRockit SDK のサポート上の制約」を参照してください。

WebLogic Platform と共にバンドルされていない JVM を使用する

以下の手順は、WebLogic Platform 7.0 SP5 をすでにインストール済みでバンドルされていない JVM を使用するユーザのためのものです。

注意： 将来のインストール アップグレードを容易にするために、この手順では、インストーラをアップグレードする際に WebLogic Platform と共にバンドルされていない JVM を使用することをお勧めします。

選択したインストーラによって、WebLogic Platform 7.0 SP5 には、Sun Java 2 SDK (jdk131_10) または WebLogic JRockit SDK (jrockit70sp5_131_10 のいずれかの JVM がバンドルされます。

1. `BEA_HOME/jdk131_10 (Sun)` または `BEA_HOME/jrockit70sp5_131_10 (WebLogic JRockit)` ディレクトリをバックアップ先に移動します。
2. 新しい JVM のダウンロードとインストール
3. 新しい JVM インストール ディレクトリを、`BEA_HOME` に移動またはコピーして、手順 1 でバックアップしたディレクトリの名前と一致する名前にリネームします。
4. インストールにバンドルされているものと異なる JVM ベンダを使っている場合は、**Sun JVM** を使用するならば `Sun` に、**WebLogic JRockit JVM** を使用するならば `BEA` に、`COMM_JAVA_VENDOR` 環境変数が設定されていることを確認する必要があります。`COMM_JAVA_VENDOR` 環境変数は、以下のディレクトリのいずれかにある `commEnv` ファイルに記載されています。

```
BEA_HOME\weblogic700\common\bin\commEnv.cmd  
(Windows)BEA_HOME/weblogic700/common/bin/commEnv.sh (Unix)
```

注意： WebLogic JRockit では、E-Business Control Center はサポートされていません。Sun SDK から WebLogic JRockit に切り替えたり、E-Business Control Center を使用する場合は、さらにファイル `BEA_HOME\weblogic700\ebcc\bin\ide.cfg` 内の `-jdkhome` 行を、手順 1 で保存したバックアップ先の Sun Java 2 SDK 1.3.1_10 インストール ディレクトリを参照するように修正する必要があります。

[コンフィグレーション]ウィザードを使用して作成されたドメインを移行する

[コンフィグレーション]ウィザードは、新しいドメインをすばやく簡単に作成できる WebLogic Platform のツールです。WebLogic Platform 7.0 インストールを新しい 7.0 サービスパック (SP) リリースにアップグレードする場合、[コンフィグレーション]ウィザードを使用して作成したすべてのドメインを、引き続き新しい 7.0 SP リリースで使用するには、移行する必要があります。この節で説明する手順は、WebLogic Platform インストール 7.0 GA、7.0 SP1、7.0 SP2、7.0 SP4、および 7.0 SP5 に適用されます。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

注意: WebLogic Platform を希望する 7.0 SP リリースにまだアップグレードしていない場合は、まずアップグレードを行い、インストールが完了してから以下の手順を実行してください。インストールの手順については、次の URL の『WebLogic Platform のインストール』の「サービス バックとローリング パッチのインストール」を参照してください。

<http://edocs.beasys.co.jp/e-docs/platform/docs70/install/update.html>

ドメインを新リリースへ移行する前に、以前の 7.0.x リリースが正常に実行されていることを確認してください。

移行処理は、2つの手順で構成されます。

1. 移行スクリプトを実行する

ドメイン ディレクトリ内のプロダクト JAR ファイルをアップグレードする移行スクリプトを実行する必要があります。

2. 移行を実行するための手動作業の実行

ドメイン内の特定のファイルを手動で更新する必要があります。

3. 新しいインストール環境 (アップグレード環境ではない) でスクリプトおよびコンフィグレーション ファイルを更新する

この手順は、WebLogic Platform を *BEA_HOME* ディレクトリ以外のディレクトリにインストールした場合のみ必要です。既存の WebLogic Platform 7.0 インストールをすでにアップグレードしている場合は、この手順を省略してください。この手順には、ドメインの起動スクリプト (*startWebLogic* など)、コンフィグレーション ファイル (*config.xml* など) および作成したカスタム スクリプト内の *BEA_HOME* 環境変数の更新が必要です。

これらの手順は、以下の節で詳しく説明します。この手順は、移行するドメインごとに繰り返す必要があります。

注意: この手順は、実行しているのが Sun Java 2 SDK あるいは WebLogic JRockit SDK かどうかに関係なく適用されます。

移行スクリプトを実行する

[コンフィグレーション] ウィザードで作成した既存のドメインを移行する最初の手順は、移行スクリプトを実行して、そのドメインのプロダクト JAR ファイルをアップグレードすることです。そのために、`BEA_HOME\weblogic700\server\bin` に移動して、以下のコマンドのいずれかを入力します。

- Windows の場合 : `migrate.cmd domain mode`
- UNIX の場合 : `migrate.sh domain mode`

以下の表は、コマンド ライン引数の定義を示しています。

表 1 移行スクリプトのコマンド ライン引数

引数	解説
<code>domain</code>	ドメイン ディレクトリの絶対パス名。
<code>mode</code>	移行モード。設定できる値は <code>upgrade</code> または <code>revert</code> です。 <code>upgrade</code> — 必要に応じて、ドメイン ディレクトリにあるプロダクト JAR ファイルをアップグレードするには、このモードを選択する。詳細については、11 ページの「アップグレード中に実行されるアクション」を参照してください。これはデフォルトのモードです。 <code>revert</code> — このモードは、移行したドメインを、移行スクリプトの起動時に生成されたバックアップ ファイル (<code>*.jar.orig</code>) を使って、移行前の状態に戻す場合に選択します。バックアップ ファイルが存在しない場合、コマンドは無視されます。

いずれかのキーを押して処理を開始するよう求めるメッセージが表示されます。

何らかの理由により、前の状態に戻すことになった場合は、まずアップグレードしたすべてのドメインを元の状態に戻します。ドメインを元に戻した場合のみ、**WebLogic Platform** のバージョンを戻してください。移行を元に戻すには、コマンド ラインで `revert` オプションを指定して、移行スクリプトを再度実行します。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

アップグレード中に実行されるアクション

デフォルトの移行モード `upgrade` を指定すると、スクリプトは次のアクションを実行します。

- オリジナルのプロダクト JAR ファイルを保存する。
 - 7.0 SP1 に移行する場合、ファイル名は `*.jar.orig` になります。
 - 7.0 SP2 に移行する場合、ファイル名は `*.jar.orig_bfosp2` になります。
 - 7.0 SP4 に移行する場合、ファイルは同じファイル名で `pre_sp3backup` ディレクトリに保存されます。
 - 7.0 SP5 に移行する場合、ファイルは同じファイル名で `pre_sp3backup` ディレクトリに保存されます。
- ドメインを 7.0 SP2 に移行する場合は、移行スクリプトも以下の変更を行う。
 - Java オプション (`-server`、`-hotspot`) を `COMM_VM` 変数と置き換えます。
 - `%JAVA_VM%` に存在するものをすべて `%COMM_VM%` に置き換えます。
 - 新しいセキュリティ証明書を各 **WebLogic Integration** ドメインにコピーします。
 - `jdk131_03` に存在するものをすべて `jdk131_06` に置き換えます。(すべてのカスタム スクリプトを新しい SDK を参照するように変更する必要があります)。
- ドメインを 7.0 SP4 に移行する場合は、移行スクリプトも以下の変更を行う。
 - `jdk131_0x` に存在するものをすべて `jdk131_08` に置き換えます(すべてのカスタム スクリプトを新しい SDK を参照するように変更する必要があります)。

注意： Sun Java 2 SDK へのすべての参照を、`config.xml` ファイルおよび SDK を参照するカスタム スクリプト内の `jdk131_08` へ手動で置換する必要があります。

- Java オプション (`-server`、`-hotspot`) を `COMM_VM` 変数と置き換えます。
- `%JAVA_VM%` に存在するものをすべて `%COMM_VM%` に置き換えます。
- `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\apps\paymentWSApp` のすべてのファイルを **Web** アプリケーション ディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\paymentWSA` にコピーします。
- `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\apps\jars` のすべてのファイルをディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\portalApp` にコピーします。

- *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\webapps\datsync\WEB-INF\lib のすべてのファイルを Web アプリケーション ディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\portalApp\datsync\WEB-INF\lib にコピーします。
 - *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\webapps\tools\WEB-INF\lib のすべてのファイルを Web アプリケーション ディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\portalApp\tools\WEB-INF\lib にコピーします。
 - *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\webapps\toolSupport\WEB-INF\lib のすべてのファイルを Web アプリケーション ディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\portalApp\toolSupport\WEB-INF\lib にコピーします。
 - *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\apps\jars のすべての .jar ファイルを、アプリケーションの
BEA_HOME\user_projects\domain\beaApps\portalApp ディレクトリ (デフォルト) にコピーします。
 - *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\apps\taxWSApp のすべてのファイルをディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\taxWSApp にコピーします。
- ドメインを 7.0 SP5 に移行する場合は、移行スクリプトも以下の変更を行う。
- jdk131_0x に存在するものをすべて jdk131_10 に置き換えます (You must modify all custom scripts to reference the new SDK.)
- 注意：** Sun Java 2 SDK へのすべての参照を、config.xml ファイルおよび SDK を参照するカスタム スクリプト内の jdk131_10 へ手動で置換する必要があります。
- Java オプション (-server、-hotspot) を COMM_VM 変数と置き換えます。
 - %JAVA_VM% に存在するものをすべて %COMM_VM% に置き換えます。
 - *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\apps\paymentWSApp のすべてのファイルをディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\paymentWSApp にコピーします。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

- *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\apps\jars のすべてのファイルをディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\portalApp にコピーします。
- *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\webapps\datsync\WEB-INF\lib のすべてのファイルを **Web** アプリケーション ディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\portalApp\datsync\WEB-INF\lib にコピーします。
- *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\webapps\tools\WEB-INF\lib のすべてのファイルを **Web** アプリケーション ディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\portalApp\tools\WEB-INF\lib にコピーします。
- *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\webapps\toolSupport\WEB-INF\lib のすべてのファイルを **Web** アプリケーション ディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\portalApp\toolSupport\WEB-INF\lib にコピーします。
- *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\apps\jars のすべての .jar ファイルを、アプリケーションの
BEA_HOME\user_projects\domain\beaApps\portalApp ディレクトリ (デフォルト) にコピーします。
- *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\apps\taxWSApp のすべてのファイルをディレクトリ (デフォルト)
BEA_HOME\user_projects\domain\beaApps\taxWSApp にコピーします。

アップグレードの例

デフォルトのユーザプロジェクト ディレクトリ *BEA_HOME*\user_projects にある mydomain というドメインをアップグレードすると仮定します。使用しているプラットフォームに適したコマンドを実行します。

- Windows の場合 : migrate.cmd c:\bea\user_projects\mydomain upgrade
- UNIX の場合 : migrate.sh /bea/user_projects/mydomain upgrade

再開の例

移行処理で `mydomain` に行われた変更を元に戻すには、使用しているオペレーティングシステムに適したスクリプトを実行します。

- Windows の場合 : `migrate.cmd c:\bea\user_projects\mydomain revert`
- UNIX の場合 : `migrate.sh /bea/user_projects/mydomain revert`

移行を実行するための手動作業の実行

ドメインの作成に使用したドメイン テンプレートによっては、既存の **WebLogic Platform 7.0 SP** リリースをサポートするために、既存のスクリプトやファイルを追加または変更する必要があります。

以下の節では、移行を実行するために必要な手動作業について説明します。

- **WebLogic Platform 7.0 SP1** に移行するための手動作業の実行
- **WebLogic Platform 7.0 SP2** に移行するための手動作業の実行
- **WebLogic Platform 7.0 SP4** に移行するための手動作業の実行
- **WebLogic Platform 7.0 SP5** に移行するための手動作業の実行

注意： 説明された移行手順は、7.0 SP リリースの順番に従って順次実行しなければなりません。たとえば、7.0 から 7.0 SP5 に移行する場合、別途説明された 7.0 SP1、7.0 SP2 および 7.0 SP4 への移行手順を実行してから、7.0 SP5 への移行手順を実行する必要があります。

WebLogic Platform 7.0 SP1 に移行するための手動作業の実行

この節では、ドメインを **WebLogic Platform 7.0** から **WebLogic Platform 7.0 SP1** に移行するために必要な手動作業について説明します。ドメインの作成に使用したドメイン テンプレートに基づいて、該当する節を参照し、記載された手順を実行します。

- **BPM Domain**
- **EAI Domain**

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

- Platform Domain
- WebLogic Workshop Domain
- WLI Domain
- WLP Domain
- WLS Domain
- WLS Examples
- WLS Petstore

注意： ファイルを追加または変更する前に、以下の節で説明するように、元のファイルのバックアップをとることをお勧めします。

BPM Domain

BPM Domain テンプレートに基づくドメインの場合は、以下の手順を実行します。

1. WLISERVERCP 変数を startWebLogic.cmd スクリプト (Windows の場合) または startWebLogic.sh スクリプト (UNIX の場合) に追加して、CLASSPATH を定義します。startWebLogic スクリプトは、*BEA_HOME*\user_projects\domain ディレクトリにあります。

以下のサンプルは startWebLogic.cmd スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

```
:pointbaseREM Invoke a script to finish up workset  
WLISERVERCP=%WLISERVERCP%;%PBCOMMONCP%;set  
SCRIPT=%WLI_HOME%\lib\scripts\PointbaseChecker.xml
```

2. PBCOMMONCP 変数を、startManagedWebLogic.cmd スクリプト (Windows の場合) または startManagedWebLogic.sh スクリプト (UNIX の場合) の CLASSPATH 定義の最後に追加します。startManagedWebLogic スクリプトは、*BEA_HOME*\user_projects\domain ディレクトリにあります。

以下のサンプルは startManagedWebLogic.cmd スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

```
set CLASSPATH=%WLISERVERCP%;%PBCOMMONCP%
```

3. CLASSPATH に定義した JAR ファイル名が適切な PointBase バージョン (183 vs 172) を反映するように、setDBVars.cmd コマンド (Windows の場合) または setDBVars コマンド (UNIX の場合) を変更します。両方のコマンドのファイルは、デフォルトでは以下のディレクトリにあります。

`BEA_HOME\user_projects\domain\dbInfo\pointbase`

以下のサンプルは setDBVars.cmd スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

変更前:

```
set CLCP=-classpath
%WL_HOME%\..\samples\server\eval\pointbase\lib\pbserver42ECF172.jar

set
CLCP=%CLCP%;%WL_HOME%\..\samples\server\eval\pointbase\lib\pbclient4
2ECF172.jar

set
CLCP=%CLCP%;%WL_HOME%\..\samples\server\eval\pointbase\lib\pertools42
ECF172.jar;
```

変更後:

```
set CLCP=-classpath
%WL_HOME%\..\samples\server\eval\pointbase\lib\pbserver42ECF183.jar

set
CLCP=%CLCP%;%WL_HOME%\..\samples\server\eval\pointbase\lib\pbclient4
2ECF183.jar

set
CLCP=%CLCP%;%WL_HOME%\..\samples\server\eval\pointbase\lib\pertools42
ECF183.jar;
```

EAI Domain

EAI Domain テンプレートに基づくドメインの場合は、15 ページの「BPM Domain」に記載されている手順を参照してください。

Platform Domain

Platform Domain テンプレートに基づくドメインの場合は、以下の手順を実行します。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

1. (デフォルトでは、

BEA_HOME\user_projects\domain\beaApps\portalApp\tools\WEB-INF ディレクトリにある) ツール **Web** アプリケーションの web.xml ファイルで、Customer Profile and Order Pages セキュリティ規則を検索し、<url-pattern> 要素を使用してこの規則を適用するリソースを定義します。

以下のサンプルは web.xml ファイルからの抜粋で、更新が必要な箇所は太字で示されています。

```
<security-constraint>  <!-- リソースコレクションを定義する -->
<web-resource-collection>  <web-resource-name>      Customer Profile
and Order Pages  </web-resource-name>  <description>      Customer
Profile and Order Pages  </description>  <!-- URL pattern for the
resource collection -->  <b>url-pattern>/tools/*</b></url-pattern>
<b>url-pattern>/repository/*</b></url-pattern>
<b>url-pattern>/security/*</b></url-pattern>  <http-method>GET</http-method>
<http-method>POST</http-method>
</web-resource-collection></security-constraint>
```

注意： WebLogic Server は、security-constraint 要素内の各

web-resource-collection 要素を検証して、少なくとも 1 つの URL パターンが含まれることを確認します。ドメインに別の **Web** アプリケーションがある場合は、すべての web-resource-collection 要素にセキュリティ規則に準拠する URL パターンが少なくとも 1 つ含まれることを確認してください。

2. *BEA_HOME*\weblogic700\common\templates\domains\shared\bea\portal\webapps\tools\tools ディレクトリからツール **Web** アプリケーションの tools ディレクトリに、以下のファイルをコピーします。カスタマイズしたファイルを上書きしないように気をつけてください。

```
catalog\category_add_remove_items.jspcatalog\item_property_edit.jspcatalog\item_property_edit_mr.jspcatalog\item_property_edit_mu.jspcatalog\item_property_edit_sr.jspcatalog\item_property_edit_su.jspcatalog\item_search.jspusermgmt\groupuser_property_edit_mr.jspusermgmt\groupuser_property_edit_mu.jspusermgmt\groupuser_property_edit_sr.jspusermgmt\groupuser_property_edit_su.jspusermgmt\group_add_remove_users.jspusermgmt\group_edit.jspusermgmt\group_scope_property.jspusermgmt\user_create.jspusermgmt\user_edit_info.jspusermgmt\user_scope_property.jsp
```

3. *BEA_HOME*\weblogic700\samples\portal\p13nDomain\beaApps\p13nApp\META-INF ディレクトリから *BEA_HOME*\user_projects\domain\beaApps\portalApp\META-INF ディレクトリに、weblogic-application.xml ファイルをコピーします。このファイル名で作成された可能性のあるファイルを上書きしないように気をつけてください。
4. ドメインにポータル **Web** アプリケーションを作成した場合は、アプリケーションごとに以下の手順を実行する必要があります。

- a. *BEA_HOME*\weblogic700\common\templates\webapps\portal\baseportal\j2ee\framework ディレクトリから、ポータル Web アプリケーションの framework ディレクトリに、以下のファイルをコピーします。これらのファイル名で作成された可能性のあるファイルを上書きしないように気をつけてください。

```
edit_titlebar.propertieserror\configurationerror.propertieserror\footer.incerror\header.incerror\header.propertieserror\missingformfield.propertieserror\parameters.propertieserror\pipeline.propertieserror\request.propertieserror\runtimeerror.propertieserror\shnav_bar.propertieserror\maximize_titlebar.propertieserror\minimize_titlebar.propertieserror\normal_titlebar.propertieserror\security\help.propertieserror\security\meta.incnvnav_bar.properties
```

- b. *BEA_HOME*\weblogic700\common\templates\webapps\portal\baseportal\j2ee\framework ディレクトリ内の該当ファイルを比較対象として使用して、ポータル Web アプリケーションの framework ディレクトリ内にある以下の JSP ファイルを更新します。ポータルをカスタマイズするために JSP ファイルを変更した可能性があるため、カスタマイズしていないことが確実でない限り、既存のファイルを上書きすることはお勧めしません。

```
edit_titlebar.incerror\configurationerror.jsperror\error.jsperror\missingformfield.jsperror\parameters.jsperror\pipeline.jsperror\request.jsperror\runtimeerror.jsperror\sessiontimeout.jsperror\sessiontimeout.propertieserror\float_portlet.jsperror\shnav_bar.jsperror\maximize_titlebar.incnminimize_titlebar.incnormal_titlebar.incnsecurity\help.jsperror\security\login_header.incnsecurity\need_group.jsperror\security\new_user.jsperror\security\set_password.jsperror\tools\header.jsperror\tools\header.propertieserror\tools\portal_prefs.jsperror\pvnav_bar.jsp
```

- c. *BEA_HOME*\weblogic700\common\templates\webapps\portal\baseportal\j2ee\WEB-INF\lib ディレクトリから、ポータル Web アプリケーションの WEB-INF\lib ディレクトリに、以下のファイルをコピーします。カスタマイズしたファイルを上書きしないように気をつけてください。

```
ent_taglib.jarerror_taglib.jarerror18n_taglib.jarerrorlib_taglib.jarerror13n_servlet.jarerrorportal_servlet.jarerrorportal_taglib.jarerrorportlet_taglib.jarerrorren_taglib.jarerrorres_taglib.jarerrorum_taglib.jarerrorutil_taglib.jarerrorvisitor_taglib.jarerrorwebflow_servlet.jarerrorwebflow_taglib.jarerrorweblogic-tags.jar
```

WebLogic Workshop Domain

WebLogic Workshop テンプレートに基づくドメインの場合は、以下の手順を実行します。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

1. CLASSPATH に定義した JAR ファイル名が適切な PointBase バージョン (183 vs 172) を反映するように、startWebLogic.cmd コマンド (Windows の場合) または startWebLogic.sh コマンド (UNIX の場合) を変更します。両方のコマンドのファイルは、デフォルトでは以下のディレクトリにあります。

BEA_HOME\user_projects\domain

以下のサンプルは startWebLogic.cmd スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

変更前:

```
set
PB_CLASSPATH=%POINTBASEDIR%\eval\pointbase\lib\pbserver42ECF172.jar;
%POINTBASEDIR%\eval\pointbase\lib\pbclient42ECF172.jar
```

変更後:

```
set
PB_CLASSPATH=.\;%POINTBASEDIR%\eval\pointbase\lib\pbserver42ECF183.jar;
%POINTBASEDIR%\eval\pointbase\lib\pbclient42ECF183.jar
```

2. *BEA_HOME*\weblogic700\samples\workshop ディレクトリから WebLogic Workshop ドメインの *BEA_HOME*\user_projects\domain ディレクトリに、以下のファイルをコピーします。これらのファイル名で作成された可能性のあるファイルを上書きしないように気をつけてください。

setWorkshopEnv.cmdsetWorkshopEnv.shstartPointBaseConsole.cmdstartPointBaseConsole.shURLs.dat

WLI Domain

WLI Domain テンプレートに基づくドメインの場合は、以下の手順を実行します。

1. SVRCP 変数を startWebLogic.cmd スクリプト (Windows の場合) または startWebLogic.sh スクリプト (UNIX の場合) に追加して、CLASSPATH を定義します。startWebLogic スクリプトは、*BEA_HOME*\user_projects\domain ディレクトリにあります。

以下のサンプルは startWebLogic.cmd スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

```
:pointbaseREM Invoke a script to finish up workset
WLISERVERCP=%WLISERVERCP%;%PBCOMMONCP%;set
SCRIPT=%WLI_HOME%\lib\scripts\PointbaseChecker.xml
```

2. startManagedWebLogic スクリプトを開きます。

- Windows の場合 :

```
BEA_HOME\user_projects\domain\startManagedWebLogic.cmd
```

- UNIX の場合 : `BEA_HOME\user_projects\domain\startManagedWebLogic`

3. PBCOMMONCP 変数を CLASSPATH 定義の最後に追加します。
4. 7.0 GA で作成した WLI、BPM、または EAI ドメインを実行している場合は、次の行も削除 (またはコメントアウト) します。

```
JAVA_VM=-server
```

以下のサンプルは `startManagedWebLogic.cmd` スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

```
set
CLASSPATH=%WLI_SERVERCP%;%WLI_HOME%\lib\hlcommon.jar;%WLI_HOME%\lib\mekshared.jar;%WLI_HOME%\lib\powerapi.jar;%PBCOMMONCP%
```

5. CLASSPATH に定義した JAR ファイル名が適切な PointBase バージョン (183 vs 172) を反映するように、`setDBVars.cmd` コマンド (Windows の場合) または `setDBVars` コマンド (UNIX の場合) を変更します。両方のコマンドのファイルは、デフォルトでは以下のディレクトリにあります。

```
BEA_HOME\user_projects\domain\dbInfo\pointbase
```

以下のサンプルは `setDBVars.cmd` スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

変更前 :

```
set CLCP=-classpath
%WL_HOME%\..\samples\server\eval\pointbase\lib\pbserver42ECF172.jar

set
CLCP=%CLCP%;%WL_HOME%\..\samples\server\eval\pointbase\lib\pbclient42ECF172.jar

set
CLCP=%CLCP%;%WL_HOME%\..\samples\server\eval\pointbase\lib\pbtools42ECF172.jar;%WLI_DOMAIN_HOME%
```

変更後 :

```
set CLCP=-classpath
%WL_HOME%\..\samples\server\eval\pointbase\lib\pbserver42ECF183.jar

set
CLCP=%CLCP%;%WL_HOME%\..\samples\server\eval\pointbase\lib\pbclient42ECF183.jar
```


[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

```
set
CLCP=%CLCP%;%WL_HOME%\..\samples\server\eval\pointbase\lib\pbtools42
ECF183.jar;%WLI_DOMAIN_HOME%
```

WLP Domain

WLP Domain テンプレートに基づくドメインを移行するには、16 ページの「Platform Domain」に記載されている手順を参照してください。

WLS Domain

WLS Domain テンプレートに基づくドメインの場合は、既存のスクリプトやファイルを追加または変更する必要はありません。

WLS Examples

WLS Examples テンプレートに基づくドメインの場合は、以下の手順を実行します。

1. startExamplesServer.cmd コマンド (Windows の場合) または startExamplesServer.sh コマンド (UNIX の場合) の CLASSPATH 定義を、以下のように変更します。

- 適切な PointBase バージョン (183 vs 172) を JAR ファイル名に反映させるように変更する。
- *BEA_HOME*\server\lib\webservices.jar ファイルを追加する。

両方のコマンドのファイルは、デフォルトでは以下のディレクトリにあります。

```
BEA_HOME\user_projects\domain
```

以下のサンプルでは startExamplesServer.cmd スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

変更前:

```
set
CLASSPATH=c:\bea\jdk131_03\lib\tools.jar;%POINTBASE_HOME%\lib\pbserver42ECF172.jar;%POINTBASE_HOME%\lib\pbclient42ECF172.jar;%CLIENT_CLASSES%;%SERVER_CLASSES%;%COMMON_CLASSES%;%CLIENT_CLASSES%\utils_common.jar
```

変更後:

```
set
CLASSPATH=c:\bea\jdk131_03\lib\tools.jar;%POINTBASE_HOME%\lib\pbserver42ECF183.jar;%POINTBASE_HOME%\lib\pbclient42ECF183.jar;%CLIENT_CLASSES%;%SERVER_CLASSES%;%COMMON_CLASSES%;%CLIENT_CLASSES%\utils_common.jar
```

```
SSES%;%SERVER_CLASSES%;%COMMON_CLASSES%;%CLIENT_CLASSES%\utils_commo  
n.jar;c:\bea\weblogic700\server\lib\webservices.jar
```

2. `BEA_HOME\samples\server\config\examples\applications` ディレクトリから Web アプリケーションの `BEA_HOME\user_projects\WLSEExampleDomain\applications` ディレクトリに、`Webservices_trader.ear` ファイルをコピーします。カスタマイズしたファイルを上書きしないように気をつけてください。

WLS Petstore

WLS Petstore ドメイン テンプレートに基づいたドメインの場合は、`CLASSPATH` に定義した JAR ファイル名が適切な PointBase バージョン (183 vs 172) を反映するように、`startPetStore.cmd` コマンド (Windows の場合) または `startPetStore.sh` コマンド (UNIX の場合) を変更します。

両方のコマンドのファイルは、デフォルトでは以下のディレクトリにあります。

```
BEA_HOME\user_projects\domain
```

以下のサンプルは `startPetStore.cmd` スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

変更前:

```
set  
CLASSPATH=%JAVA_HOME%\lib\tools.jar;%POINTBASE_HOME%\lib\pbserver42ECF1  
72.jar;%POINTBASE_HOME%\lib\pbclient42ECF172.jar;%SERVER_CLASSES%;%COMM  
ON_CLASSES%
```

変更後:

```
set  
CLASSPATH=%JAVA_HOME%\lib\tools.jar;%POINTBASE_HOME%\lib\pbserver42ECF1  
83.jar;%POINTBASE_HOME%\lib\pbclient42ECF183.jar;%SERVER_CLASSES%;%COMM  
ON_CLASSES%
```

WebLogic Platform 7.0 SP2 に移行するための手動作業の実行

この節では、ドメインを WebLogic Platform 7.0 SP2 に移行するために必要な手動作業について説明します。

1. WebLogic Platform 7.0 ドメインを移行する場合は、14 ページの「WebLogic Platform 7.0 SP1 に移行するための手動作業の実行」の説明に従って、ドメインを WebLogic Platform 7.0 SP1 に移行するために必要な手動作業が実行されていることを確認します。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

2. 以下のテンプレートに基づくドメインを、WebLogic Platform 7.0 SP2 に移行するには、以下の節で記載された手順を実行します。

- BPM Domain
- EAI Domain
- WebLogic Workshop Domain
- WLI Domain
- WLP Domain

Platform Domain、WLS Domain、WLS Examples、および WLS Petstore の各テンプレートに基づくドメインを移行する場合は、特別な手順は必要ありません。

注意： ファイルを追加または変更する前に、以下の節で説明するように、元のファイルのバックアップをとることをお勧めします。

BPM Domain

BPM テンプレートに基づくドメインを WebLogic Platform 7.0 SP2 に移行するには、以下の手順を実行します。

1. 必要があれば、データベースとセキュリティ領域データを移行します。詳細については、『WebLogic Integration 移行ガイド』を参照してください。
<http://edocs.beasys.co.jp/e-docs/wli/docs70/migrate/index.htm>

2. BPM JSP Worklist と BPM File Plugin という 2 つの新機能をサポートするには、以下の説明に従って、ドメインの config.xml ファイルを更新し、必要なドプロイ可能なコンポーネントを追加します。

注意： 以下の指示は、指定された場所を除いて、シングルサーバドメインとクラスタドメイン コンフィグレーションに適用されます。

以下の説明では、*name* はシングルサーバまたはクラスタを意味し、*cluster_server_name* は参照するクラスタ内のサーバの名前を意味します。

a. BPM JSP Worklist をサポートするには、<Application Name ="WebLogic Integration"> の下に以下のコンポーネントを追加します。

```
<WebAppComponent Name="WLI-BPM JSP Worklist" Targets="name"
URI="worklist.war"/>
```

b. BPM File Plugin をサポートするには、<Application Name ="WebLogic Integration"> の下に以下のコンポーネントを追加します。

```
<EJBComponent Name="WLI BPM File Plug-in" Targets="name"
URI="fileplugin-ejb.jar"/>
```

JMS Server テキスト ブロック の下に、以下のコンポーネントを追加します。シングルサーバ コンフィグレーションの `cluster_server_name` 変数は無視して、クラスタードメイン コンフィグレーション内の各サーバについてこの手順を繰り返します。

```
<JMSQueue Name="WLI_BPM_FP-cluster_server_name"
JNDIName="com.bea.wli.bpm.FilePluginQueue-cluster_server_name"
StoreEnabled="true"
Template="WLI_JMSTemplate-cluster_server_name"/>
```

クラスタードメイン コンフィグレーションの場合のみ、`<Application Name="WebLogic Integration">` の下に以下のコンポーネントを追加します。

```
<JMSDistributedQueue Name="WLI_BPM_FP"
JNDIName="com.bea.wli.bpm.FilePluginQueue" Targets="name">
<JMSDistributedQueueMember
Name="WLI_BPM_FP-cluster_server_name"
JMSQueue="WLI_BPM_FP-cluster_server_name" Weight="1"/>
<JMSTemplate Name="WLI_BPM_FP"/></JMSDistributedQueue>
```

例：

以下は、**BPM Domain** テンプレートに基づくドメインを移行する方法の例です。また、データベース スキーマを更新し、**BPM File Plugin** をコンフィグレーションする方法についても説明します。

1. **BPM** データベース テーブルを、**BPM File Plug-In** で使用する新しいデータベース テーブル `FILEPOLL` に更新します。この場合、次のスクリプトを実行します。

```
BEA_HOME\weblogic700\integration\dbscripts\database_type\migrate\BPM_70-70SP2.sql
```

ここで、`BEA_HOME` は **WebLogic Platform** のホーム ディレクトリを表します。

2. シングルサーバドメインの `config.xml` ファイルに **BPM File Plug-in** をコンフィグレーションして、このドメインを移行する準備をします。
 - a. `fileplugin-ejb.jar` を **WebLogic Integration** アプリケーションの 1 つのコンポーネントとしてデプロイするには、次の行を追加します。

```
<EJBComponent Name="WLI-BPM File
Plug-in"Targets="Customer_Server_Name" URI="fileplugin-ejb.jar"/>
```

- b. **BPM File Plug-in** の **JMS** キューを追加するには、次の行を追加します。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

```
<JMSQueue  
JNDIName="com.bea.wli.bpm.FilePluginQueue"Name="WLI_BPM_FP"  
Template="WLI_JMSTemplate"/>
```

3. クラスタドメインの config.xml ファイルに **BPM File Plug-in** をコンフィグレーションして、このドメインを移行する準備をします。

注意： この例は、1 台の管理サーバ (myserver) と 2 台の管理対象サーバ (c1、c2) が存在するクラスタシステム (mycluster) を示しています。

- a. fileplugin-ejb.jar をクラスタサーバにデプロイするには、次の行を追加します。

```
<EJBComponent Name="WLI BPM File Plug-in" Targets="mycluster"  
URI="fileplugin-ejb.jar"/>
```

- b. **BPM File Plug-in** の JMS キューをコンフィグレーションするには、次の行を追加します。

```
<JMSDistributedQueue Name="WLI_BPM_FP"  
JNDIName="com.bea.wli.bpm.FilePluginQueue" Targets="mycluster">  
<JMSDistributedQueueMember Name="WLI_BPM_FP-c1"  
JMSQueue="WLI_BPM_FP-c1" Weight="1"/>  
<JMSDistributedQueueMember Name="WLI_BPM_FP-c2"  
JMSQueue="WLI_BPM_FP-c2" Weight="1"/> <JMSTemplate  
Name="WLI_BPM_FP"/></JMSDistributedQueue>
```

- c. クラスタ ノード c1 JMS サーバで、次の行を追加します。

```
<JMSQueue Name="WLI_BPM_FP-c1"  
JNDIName="com.bea.wli.bpm.FilePluginQueue-c1" StoreEnabled="true"  
Template="WLI_JMSTemplate-c1"/>
```

- d. クラスタ ノード c2 JMS サーバで、次の行を追加します。

```
<JMSQueue Name="WLI_BPM_FP-c2"  
JNDIName="com.bea.wli.bpm.FilePluginQueue-c2" StoreEnabled="true"  
Template="WLI_JMSTemplate-c2"/>
```

EAI Domain

EAI Domain テンプレートに基づくドメインを移行するには、23 ページの「BPM Domain」に記載されている手順を参照してください。

WebLogic Workshop Domain

WebLogic Workshop Domain テンプレートに基づくドメインを WebLogic Platform 7.0 SP2 に移行するには、以下の手順に従って、startweblogic スクリプトを更新する必要があります。

1. 以下のコマンドを追加します。

```
call %WL_HOME%\common\bin\commEnv.cmd
```

2. 以下のように、JAVA_DEBUG 変数を設定します。

```
JAVA_DEBUG=%COMM_CLIENT_VM%
```

3. 以下のように、JAVA_HOME 変数を設定します。

```
JAVA_HOME=%JAVA_HOME%
```

WLI Domain

WLI Domain テンプレートに基づくドメインを移行するには、23 ページの「BPM Domain」に記載されている手順を実行します。

WLP Domain

WLP Domain テンプレートに基づくドメインを WebLogic Platform 7.0 SP2 に移行するには、以下の 3 つの段階の処理が必要です。

- 段階 1: ファイルのコピー
- 段階 2: web.xml ファイルの修正
- 段階 3: weblogic.xml ファイルの修正

段階 1: ファイルのコピー

注意: カスタマイズしたファイルを上書きしないように気をつけてください。

1. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\webapps\datsync` のすべてのファイルを `datsync Web` アプリケーション ディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\portalApp\datsync` にコピーします。
2. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\webapps\tools` のすべてのファイルを `tools Web` アプリケーション ディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\portalApp\tools` にコピーします。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

3. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\webapps\toolSupport` のすべてのファイルを `toolSupport Web` アプリケーション ディレクトリ (デフォルト)
`BEA_HOME\user_projects\domain\beaApps\portalApp\toolSupport` にコピーします。
4. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\apps\jars` のすべての `.jar` ファイルを、アプリケーションの
`BEA_HOME\user_projects\domain\beaApps\portalApp` ディレクトリ (デフォルト) にコピーします。
5. ドメインにポータル Web アプリケーションが作成されている場合は、すべての `.jar` ファイルを
`BEA_HOME\weblogic700\common\templates\webapps\portal\baseportal\j2ee\WEB-INF\lib` ディレクトリから各ポータル Web アプリケーションの `WEB-INF\lib` ディレクトリにコピーします。

この移行手順を実行した後で、必要なファイルがすべてコピーされたことを確認するため、**Portal Consistency Checker** を実行することをお勧めします。**Portal Consistency Checker** は、にある `dev2dev` サイトから入手できます。

http://dev2dev.bea.com/codelibrary/code/portal_consistency_checker.jsp

段階 2: `web.xml` ファイルの修正

Web アプリケーションの `web.xml` ファイルを修正するには、次の 2 つの方法を使用できます。

1. Web アプリケーションの `weblogic.xml` ファイルを、次のディレクトリの `weblogic.xml.stock` バージョンに置き換えます。
`BEA_HOME\weblogic700\common\templates\webapps\portal\baseportal\j2ee\WEB-INF`
その後、以前に変更した内容を新しくコピーしたファイルに組み込みます。
2. 以下の手順 a ~ d に示す新しいコードを、Web アプリケーション (`<listener>`、`<servlet-mapping>`、`<taglib>`、および `<ejb-ref>`) の `weblogic.xml` ファイルの各指定セクションに貼り付けます。
 - a. 次の行を `<listener>` セクションの後に追加します。

```
<!-- Filter to fire click through events -->
<filter>
  <filter-name>ClickThroughEventFilter</filter-name>
-->
```

```

<filter-class>com.bea.pl3n.tracking.clickthrough.ClickThroughEventFilter</filter-class>
</filter>
<filter-mapping>
  <filter-name>ClickThroughEventFilter</filter-name>
  <url-pattern>/application/*</url-pattern>
</filter-mapping>

```

b. 次の行を <servlet-mapping> セクションの後に追加します。

```

<!-- The ShowDoc Servlet -->
<servlet>
  <servlet-name>ShowDocServlet</servlet-name>
  <servlet-class>com.bea.pl3n.content.servlets.ShowDocServlet</servlet-class>
  <!-- Make showdoc always use the local ejb-ref DocumentManager -->
  <init-param>
    <param-name>contentHome</param-name>
    <param-value>java:comp/env/ejb/DocumentManager</param-value>
  </init-param>
</servlet>
<!-- The AdClickThru Servlet -->
<servlet>
  <servlet-name>adClickThru</servlet-name>
  <servlet-class>com.bea.pl3n.ad.servlets.AdClickThruServlet</servlet-class>
</servlet>
<!-- The ClickThrough Servlet -->
<servlet>
  <servlet-name>clickThroughServlet</servlet-name>
  <servlet-class>com.bea.pl3n.tracking.clickthrough.ClickThroughServlet</servlet-class>
</servlet>
<servlet-mapping>
  <servlet-name>ShowDocServlet</servlet-name>
  <url-pattern>/ShowDoc/*</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>adClickThru</servlet-name>
  <url-pattern>/adClickThru/*</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>adClickThru</servlet-name>
  <url-pattern>/AdClickThru/*</url-pattern>
</servlet-mapping>
<servlet-mapping>
  <servlet-name>clickThroughServlet</servlet-name>
  <url-pattern>/clickThroughServlet/*</url-pattern>
</servlet-mapping>

```

c. 次の行を <taglib> セクションに追加します。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

```
<taglib>
  <taglib-uri>cat.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/cat_taglib.jar</taglib-location>
</taglib>
<taglib>
<taglib-uri>eb.tld</taglib-uri>
<taglib-location>/WEB-INF/lib/eb_taglib.jar</taglib-location>
</taglib>
<taglib>
<taglib-uri>productTracking.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/productTracking_taglib.jar</taglib-location>
</taglib>
<taglib>
  <taglib-uri>ad.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/ad_taglib.jar</taglib-location>
</taglib>
<taglib>
  <taglib-uri>cm.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/cm_taglib.jar</taglib-location>
</taglib>
<taglib>
  <taglib-uri>ph.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/ph_taglib.jar</taglib-location>
</taglib>
<taglib>
  <taglib-uri>ps.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/ps_taglib.jar</taglib-location>
</taglib>
<taglib>
  <taglib-uri>pz.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/pz_taglib.jar</taglib-location>
</taglib>
<taglib>
  <taglib-uri>tracking.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/tracking_taglib.jar</taglib-location>
</taglib>
<taglib>
  <taglib-uri>dam.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/dam_taglib.jar</taglib-location>
</taglib>
<taglib>
  <taglib-uri>vum.tld</taglib-uri>
  <taglib-location>/WEB-INF/lib/vum_taglib.jar</taglib-location>
</taglib>
```

d. 次の行を <ejb-ref> セクションに追加します。

```
<!-- これは各種の <cm:> タグに使用される -->
<ejb-ref>
```

```

<description>
    The ContentManager EJB for this webapp
</description>
<ejb-ref-name>ejb/ContentManager</ejb-ref-name>
<ejb-ref-type>Session</ejb-ref-type>
<home>com.bea.p13n.content.document.DocumentManagerHome</home>
<remote>com.bea.p13n.content.document.DocumentManager</remote>
</ejb-ref>
<!-- This is used by the ShowDocServlet -->
<ejb-ref>
    <description>
        The DocumentManager for this webapp
    </description>
    <ejb-ref-name>ejb/DocumentManager</ejb-ref-name>
    <ejb-ref-type>Session</ejb-ref-type>
    <home>com.bea.p13n.content.document.DocumentManagerHome</home>
    <remote>com.bea.p13n.content.document.DocumentManager</remote>
</ejb-ref>
<!-- This is used by the Placeholder tag -->
<ejb-ref>
    <description>
        The PlaceholderService Session EJB for the placeholder tag.
    </description>
    <ejb-ref-name>ejb/PlaceholderService</ejb-ref-name>
    <ejb-ref-type>Session</ejb-ref-type>
    <home>com.bea.p13n.placeholder.PlaceholderServiceHome</home>
    <remote>com.bea.p13n.placeholder.PlaceholderService</remote>
</ejb-ref>
<!-- This is used by the AdClickThruServlet and the adTarget tag-->
<ejb-ref>
    <description>
        The AdService for this webapp
    </description>
    <ejb-ref-name>ejb/AdService</ejb-ref-name>
    <ejb-ref-type>Session</ejb-ref-type>
    <home>com.bea.p13n.ad.AdServiceHome</home>
    <remote>com.bea.p13n.ad.AdService</remote>
</ejb-ref>
<!-- This is used by the AdClickThruServlet -->
<ejb-ref>
    <description>
        The AdBucketService for this webapp
    </description>
    <ejb-ref-name>ejb/AdBucketService</ejb-ref-name>
    <ejb-ref-type>Session</ejb-ref-type>
    <home>com.bea.p13n.ad.AdBucketServiceHome</home>
    <remote>com.bea.p13n.ad.AdBucketService</remote>
</ejb-ref>
<!-- This is used by the various <pz:> tags -->

```

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

```
<ejb-ref>
<description>
  The EjbAdvisor for this webapp
</description>
<ejb-ref-name>ejb/EjbAdvisor</ejb-ref-name>
<ejb-ref-type>Session</ejb-ref-type>
<home>com.bea.pl3n.advisor.EjbAdvisorHome</home>
<remote>com.bea.pl3n.advisor.EjbAdvisor</remote>
</ejb-ref>
```

段階 3 : weblogic.xml ファイルの修正

Web アプリケーションの weblogic.xml ファイルを修正するには、次の 2 つの方法を使用できます。

1. Web アプリケーションの weblogic.xml ファイルを、次のディレクトリの weblogic.xml.stock バージョンに置き換えます。

```
BEA_HOME\weblogic700\common\templates\webapps\portal\baseportal\j2ee\WEB-INF
```

その後、以前に変更した内容を新しくコピーしたファイルに組み込みます。

2. 以下の新しいコードを、Web アプリケーションの weblogic.xml ファイルの <ejb-reference-description> セクションに貼り付けます。

```
<ejb-reference-description>
  <ejb-ref-name>ejb/ContentManager</ejb-ref-name>
  <jndi-name>${APPNAME}.BEA_personalization.DocumentManager</jndi-name>
</ejb-reference-description>
<ejb-reference-description>
  <ejb-ref-name>ejb/DocumentManager</ejb-ref-name>
  <jndi-name>${APPNAME}.BEA_personalization.DocumentManager</jndi-name>
</ejb-reference-description>
<ejb-reference-description>
  <ejb-ref-name>ejb/PlaceholderService</ejb-ref-name>
  <jndi-name>${APPNAME}.BEA_personalization.PlaceholderService</jndi-name>
</ejb-reference-description>
<ejb-reference-description>
  <ejb-ref-name>ejb/AdService</ejb-ref-name>
  <jndi-name>${APPNAME}.BEA_personalization.AdService</jndi-name>
</ejb-reference-description>
<ejb-reference-description>
  <ejb-ref-name>ejb/AdBucketService</ejb-ref-name>
  <jndi-name>${APPNAME}.BEA_personalization.AdBucketService</jndi-name>
</ejb-reference-description>
<ejb-reference-description>
  <ejb-ref-name>ejb/EjbAdvisor</ejb-ref-name>
```

```
<jndi-name>${APPNAME}.BEA_personalization.EjbAdvisor</jndi-name>  
</ejb-reference-description>
```

WebLogic Platform 7.0 SP4 に移行するための手動作業の実行

この節では、ドメインを WebLogic Platform 7.0 SP4 に移行するために必要な手動作業について説明します。

1. 以下の表に記載された必要な手順を実行していることを確認してください。

以下の 7.0.x リリースで作成したドメインを移行する場合	ドメインを下記のものに移行するために必要な手動作業を実行していることを確認してください。
WebLogic Platform 7.0	<ul style="list-style-type: none">■ WebLogic Platform 7.0 SP1。14 ページの「WebLogic Platform 7.0 SP1 に移行するための手動作業の実行」の説明を参照。■ WebLogic Platform 7.0 SP2。22 ページの「WebLogic Platform 7.0 SP2 に移行するための手動作業の実行」の説明を参照。
WebLogic Platform 7.0 SP1	WebLogic Platform 7.0 SP2。22 ページの「WebLogic Platform 7.0 SP2 に移行するための手動作業の実行」の説明を参照。
WebLogic Platform 7.0 SP2	あらかじめ必要な手動作業はありません。手順 2 に進みます。

2. Platform Domain、WebLogic Workshop Domain、または WLP Domain に基づくドメインを移行する場合は、以下の該当する節の手順を実行します。

- Platform Domain
- WebLogic Workshop Domain
- WLP Domain

BPM Domain、EAI Domain、WLI Domain、WLS Examples、および WLS Petstore の各テンプレートに基づくドメインを移行する場合は、特別な手順は必要ありません。

注意： ファイルを追加または変更する前に、以下の節で説明するように、元のファイルのバックアップをとることをお勧めします。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

Platform Domain

Platform Domain テンプレートに基づくドメインを、WebLogic Platform 7.0 SP4 に移行するには、以下の手順を実行します。

1. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\webapps\datsync` のすべてのファイルを `datsync Web` アプリケーション ディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\portalApp\datsync` にコピーします。
2. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\webapps\tools` のすべてのファイルを `tools Web` アプリケーション ディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\portalApp\tools` にコピーします。
3. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\webapps\toolSupport` のすべてのファイルを `toolSupport Web` アプリケーション ディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\portalApp\toolSupport` にコピーします。
4. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\apps\jars` のすべての `.jar` ファイルを、アプリケーションの `BEA_HOME\user_projects\domain\beaApps\portalApp` ディレクトリ (デフォルト) にコピーします。
5. ドメインにポータル Web アプリケーションが作成されている場合は、すべての `.jar` ファイルを `BEA_HOME\weblogic700\common\templates\webapps\portal\baseportal\j2ee\WEB-INF\lib` ディレクトリから各ポータル Web アプリケーションの `WEB-INF\lib` ディレクトリにコピーします。

この移行手順を実行した後で、必要なファイルがすべてコピーされたことを確認するため、Portal Consistency Checker を実行することをお勧めします。Portal Consistency Checker は、dev2dev サイトから入手できます。

http://dev2dev.bea.com/codelibrary/code/portal_consistency_checker.jsp

WebLogic Workshop Domain

WebLogic Workshop Domain テンプレートに基づくドメインを WebLogic Platform 7.0 SP4 に移行するには、ドメインの `startweblogic` スクリプト内の `MEM_ARGS` の値を更新する必要があります。デフォルトでは、スクリプトは下記のディレクトリにあります。

```
BEA_HOME\user_projects\domain
```

以下は startWebLogic.cmd スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所は太字で示されています。

変更前:

```
set MEM_ARGS=-Xms64m -Xmx128m
```

変更後:

```
set MEM_ARGS=%COMM_MEDIUM_MEM_ARGS%
```

WLP Domain

WLP Domain テンプレートに基づくドメインを移行するには、33 ページの「Platform Domain」に記載されている手順を参照してください。

WebLogic Platform 7.0 SP5 に移行するための手動作業の実行

この節では、ドメインを WebLogic Platform 7.0 SP5 に移行するために必要な手動作業について説明します。

1. 以下の表に記載された必要な手順を実行していることを確認してください。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

以下の 7.0.x リリースで作成したドメインを移行する場合	ドメインを下記のものに移行するために必要な手動作業を実行していることを確認してください。
WebLogic Platform 7.0	<ul style="list-style-type: none">■ WebLogic Platform 7.0 SP1。14 ページの「WebLogic Platform 7.0 SP1 に移行するための手動作業の実行」の説明を参照。■ WebLogic Platform 7.0 SP2。22 ページの「WebLogic Platform 7.0 SP2 に移行するための手動作業の実行」の説明を参照。■ WebLogic Platform 7.0 SP4。32 ページの「WebLogic Platform 7.0 SP4 に移行するための手動作業の実行」の説明を参照。
WebLogic Platform 7.0 SP1	<ul style="list-style-type: none">■ WebLogic Platform 7.0 SP2。22 ページの「WebLogic Platform 7.0 SP2 に移行するための手動作業の実行」の説明を参照。■ WebLogic Platform 7.0 SP4。32 ページの「WebLogic Platform 7.0 SP4 に移行するための手動作業の実行」の説明を参照。
WebLogic Platform 7.0 SP2	WebLogic Platform 7.0 SP4。32 ページの「WebLogic Platform 7.0 SP4 に移行するための手動作業の実行」の説明を参照。
WebLogic Platform 7.0 SP4	あらかじめ必要な手動作業はありません。手順 2 に進みます。

2. Platform Domain、WebLogic Workshop Domain、または WLP Domain テンプレートに基づくドメインを移行する場合は、以下の該当する節の手順を実行します。

- WebLogic Workshop Domain
- WLP Domain

BPM Domain、EAI Domain、Platform Domain、WLI Domain、WLS Domain、WLS Examples、および WLS Petstore の各テンプレートに基づくドメインを移行する場合は、特別な手順は必要ありません。

注意： ファイルを追加または変更する前に、以下の節で説明するように、元のファイルのバックアップをとることをお勧めします。

WebLogic Workshop Domain

注意： この節で記載された手順は、WebLogic JRockit SDK を使用する場合にのみ必要です。Sun Java 2 SDK を使用する場合は、追加の移行手順は不要です。

WebLogic JRockit SDK を使用する場合、WebLogic Workshop Domain テンプレートに基づくドメインを WebLogic Platform 7.0 SP5 に移行するには、ドメインの `startweblogic` スクリプト内の `JAVA_VM` の値を更新して、`COMM_VM` 変数を削除する必要があります。デフォルトでは、スクリプトは下記のディレクトリにあります。

```
BEA_HOME\user_projects\domain
```

以下は `startWebLogic.cmd` スクリプト (Windows の場合) からの抜粋で、更新が必要な箇所を示しています。

変更前：

```
set JAVA_VM=%COMM_VM% %JAVA_DEBUG% %JAVA_PROFILE%
```

変更後：

```
set JAVA_VM=%JAVA_DEBUG% %JAVA_PROFILE%
```

WLP Domain

Platform Domain テンプレートに基づくドメインを、WebLogic Platform 7.0 SP5 に移行するには、以下の手順を実行します。

1. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\webapps\datsync` のすべてのファイルを `datsync Web` アプリケーションディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\portalApp\datsync` にコピーします。
2. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\webapps\tools` のすべてのファイルを `tools Web` アプリケーションディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\portalApp\tools` にコピーします。
3. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\webapps\toolSupport` のすべてのファイルを `toolSupport Web` アプリケーションディレクトリ (デフォルト) `BEA_HOME\user_projects\domain\beaApps\portalApp\toolSupport` にコピーします。

[コンフィグレーション] ウィザードを使用して作成されたドメインを移行する

4. `BEA_HOME\weblogic700\common\templates\domains\shared\bea\portal\apps\jars` のすべての `.jar` ファイルを、アプリケーションの `BEA_HOME\user_projects\domainbeaApps\portalApp` ディレクトリ (デフォルト) にコピーします。
5. ドメインにポータル Web アプリケーションが作成されている場合は、すべての `.jar` ファイルを `BEA_HOME\weblogic700\common\templates\webapps\portal\baseportal\j2ee\WEB-INF\lib` ディレクトリから各ポータル Web アプリケーションの `WEB-INF\lib` ディレクトリにコピーします。

この移行手順を実行した後で、必要なファイルがすべてコピーされたことを確認するため、Portal Consistency Checker を実行することをお勧めします。Portal Consistency Checker は、dev2dev サイトから入手できます。

http://dev2dev.bea.com/codelibrary/code/portal_consistency_checker.jsp

新しいインストール環境 (アップグレード環境ではない) でスクリプトおよびコンフィグレーション ファイルを更新する

注意: この手順は、WebLogic Platform を `BEA_HOME` ディレクトリ以外のディレクトリにインストールした場合のみ必要です。既存の WebLogic Platform 7.0 インストールをすでにアップグレードしている場合は、この手順を省略してください。

`startWebLogic` などのドメイン起動スクリプトや、`config.xml` などのコンフィグレーション ファイルによって、`BEA_HOME` ディレクトリ内のファイルの絶対パスが定義されます。これらの絶対パスを検索して、新しい `BEA_HOME` ディレクトリを参照するように更新する必要があります。さらに、新しい `BEA_HOME` ディレクトリの場所を反映するように、`BEA_HOME` ディレクトリ内のファイルの絶対パス名を定義するカスタム スクリプト (ビルドスクリプトなど) も更新する必要があります。

注意: 多くの起動スクリプトは、`BEA_HOME` ディレクトリを参照する変数などの環境変数を現在のシェルに設定します。スクリプト ファイル内の `BEA_HOME` 参照を更新した後、新しいシェルを開いて、最新の環境設定が使用されていることを確認します。

Platform Domain で使用されたデータベースの切り替え

この節では、Platform Domain テンプレートから作成したドメインによって使用されたデータベースを切り替える処理について、順を追って説明します。データベースを切り替えるのは、たとえば、デフォルトのデータベースである WebLogic Platform や PointBase を Oracle などの WebLogic Platform がサポートする別のデータベースに置き換える場合などです。

WebLogic Portal または WebLogic Integration ドメインで使用されたデータベースの切り替えについての詳細は、下記のマニュアルを参照してください。

- WebLogic Portal ドメインについては、下記 URL にある『WebLogic Portal 管理者ガイド』の「システム管理」の「データベース管理」に記載されている手順を実行します。

<http://edocs.beasys.co.jp/e-docs/wlp/docs70/admin/sysadmin.htm>

- WebLogic Integration ドメインについては、Integration Database Wizard を使用するか、次の URL にある『BEA WebLogic Integration の起動、停止およびカスタマイズ』の「WebLogic Integration のカスタマイズ」で説明されている wliconfig コマンドを使用します。

<http://edocs.beasys.co.jp/e-docs/wli/docs70/config/custom.htm>

手順 1: データベースのコンフィグレーション

デフォルトの PointBase データベースを別のサポート対象のデータベースに切り替えるには、その前に、切り替え先のデータベースをコンフィグレーションする必要があります。コンフィグレーションの手順については、次の場所にあるデータベースの README.html ファイルを参照してください。

```
BEA_HOME\weblogic700\portal\db\db_type\version\admin
```

手順 2: データベース環境に合わせた db_settings.properties の編集

ドメイン ディレクトリの db_settings.properties ファイルを開きます。使用しているデータベースの種類に応じて、@...@ 変数に実際の値を代入し、ファイルを保存します。

2つのデータベース ユーザとアカウント（異なるユーザ名とパスワードを持つ）を設定することをお勧めします。1つは WebLogic Portal で使用するデータベースにアクセスするためのもので、もう 1つは WebLogic Integration で使用するデータベースにアクセスするためのものです。

以下のコードサンプルは、変更前の db_settings.properties ファイルの Oracle セクションを示しています。

コード リスト 1 db_settings.properties ファイル : 変更前

```
#-----Oracle Thin
Driver-----#####IF_USING_ORACLE_THIN#database=ORACLE_TH
IN#db_version=817#jdbcdriver=oracle.jdbc.driver.OracleDriver#server=@ORAC
LE_NET_SERVICE_NAME#@port=@ORACLE_PORT#@dblogin=@ORACLE_USER#@dbpassword
=@ORACLE_PASSWORD#@wliidblogin=@ORACLE_WLI_USER#@wliidbpassword=@ORACLE_WLI
I_PASSWORD#@connection=jdbc:oracle:thin:@ORACLE_SERVER@:@ORACLE_PORT@:@
ORACLE_SID#@ENDIF_USING_ORACLE_THIN@
```

以下のコード サンプルは変更されたファイルの例です。

コード リスト 2 db_settings.properties ファイル : 変更後

```
#-----Oracle Thin
Driver-----#####database=ORACLE_THIN#db_version=817#jdbcd
river=oracle.jdbc.driver.OracleDriver#server=MY817SVC#port=1521#dblogin=m
y_portal_db_username#dbpassword=my_portal_db_password#wliidblogin=my_wli_
db_username#wliidbpassword=my_wli_db_password#connection=jdbc:oracle:thin
:@myhost:1521:MY817SID#@ENDIF_USING_ORACLE_THIN@
```

注意： このファイルでは、任意の行に # 記号 (コメント行であることを示します) を付けたり、既存のコメント行から # 記号を削除したりしないでください。「手順 7: db_settings.properties の編集によるデータベースのコメントの解除」では、必要に応じて # 記号の追加と削除を行います。新しい値は **WebLogic Server** を起動してから有効となります。

db_version は使用されている DDL を制御するため、正確な数字で設定することが重要です。以下の表は、一覧されているデータベースの各タイプに対して db_version パラメータとして指定する必要がある値を示しています。

以下のデータベースバージョンの場合	以下の数字を db_version の値として使用する。
DB2 7.2	7
Oracle 8.1.7	817
Oracle 9i	901
Oracle 9.2.0	920
PointBase 4.2	42
Microsoft SQL Server 2000	2000
Sybase 12.5	125

注意： この表に一覧されているデータベースには、現在サポートされていないものもあります。現在サポートされているデータベースの一覧については、次の URL の『WebLogic Platform 7.0 サポート対象コンフィグレーション』の「サポート対象データベース コンフィグレーション」を参照してください。

http://edocs.beasys.co.jp/e-docs/platform/supconfig/configs70/70_over/supported_db.html

手順 3: データベース環境に合わせた setDBVars および setDBVarsExt の編集

データベース環境に合わせて setDBVars および setDBVarsExt を編集するには

Platform Domain で使用されたデータベースの切り替え

1. ドメイン ディレクトリ内の dbInfo ディレクトリに移動します。次に、使用するデータベースのディレクトリ (DB2、MSSQL、Oracle、PointBase、または Sybase) に移動します。
2. 以下のいずれかのセットのファイルを開きます。
 - Windows の場合 : `setDBVars.cmd` および `setDBVarsExt.cmd`
 - UNIX の場合 : `setDBVars` および `setDBVarsExt`
3. 各ファイルの環境変数を、データベースに適した値に設定します。詳細については、次の URL の「環境変数」を参照してください。

<http://edocs.beasys.co.jp/e-docs/wli/docs70/config/keycmd.htm>

手順 4: WebLogic Server の起動

WebLogic Administration Console にアクセスするには、先に WebLogic Server を起動する必要があります。

ドメイン ディレクトリで、WebLogic Server を起動します。

- Windows の場合 : `startWeblogic.cmd`
- UNIX の場合 : `startWeblogic.sh`

注意 : データベースとして PointBase を使用する場合は、データベースを作成するための特別な手順は必要ありません。スクリプトによって自動的に作成されます。

手順 5: WebLogic Administration Console での接続プールおよびレلمの設定

この手順では、ドメイン フォルダに `db_settings.properties` ファイルを開いて（「手順 2: データベース環境に合わせた `db_settings.properties` の編集」で更新済み）、その中の値を WebLogic Administration Console にコピーします。

1. WebLogic Server を実行した状態で `http://hostname:port/console` に移動し、WebLogic Administration Console を起動します。
たとえば、WebLogic Server がインストールされているマシンで作業している場合は、`http://localhost:7501/console` に移動します。
2. WebLogic Server システム管理者のユーザ名およびパスワードを入力します。デフォルトでは、ユーザ名とパスワードはどちらも `weblogic` に設定されています。
3. Console で、[ドメイン | サービス | JDBC | 接続プール] の順に選択します。
4. `commercePool` をクリックし、`db_settings.properties` ファイルの値を貼り付けて編集します。詳細については、表 2 を参照してください。
5. [適用] をクリックして、ハイパーリンクが設定されているフィールドをクリックするか、または次に表示されたタブに移動します。
6. `dataSyncPool` をクリックし、`db_settings.properties` ファイルの値を貼り付けて編集します。詳細については、表 2 を参照してください。

Platform Domain で使用されたデータベースの切り替え

7. [適用] をクリックして、ハイパーリンクが設定されているフィールドをクリックするか、または次に表示されたタブに移動します。
8. wliPool をクリックし、db_settings.properties ファイルの値を貼り付けて編集します。詳細については、表 2 を参照してください。
9. [適用] をクリックして、ハイパーリンクが設定されているフィールドをクリックするか、または次に表示されたタブに移動します。

表 2 Oracle または MS SQL Thin ドライバを使用する場合の接続プールの値

タブ	フィールド	値
General	URL	使用しているデータベースの種類に応じて、db_settings.properties ファイルの connection の値をコピーする。
	Driver Classname	使用しているデータベースの種類に応じて、db_settings.properties ファイルの jdbcdriver の値をコピーする。
	Properties (key=value)	<p>物理的なデータベースの接続を作成するときを使用できるよう、JDBC Driver に渡されたプロパティのリストを入力する。プロパティは使用しているデータベースドライバによって異なる。たとえば、Oracle Thin ドライバを使用している場合は、ユーザキーのみが必要である。MS SQL を使用している場合は、ユーザキーとサーバキーが必要になる。使用するプロパティを特定するには、ドライバのマニュアルを参照する。</p> <p>commercePool または dataSyncPool のプロパティを指定する場合、使用しているデータベースの種類に応じて db_settings.properties ファイル内の dblogin の値をコピーする。たとえば、Oracle Thin ドライバを使用している場合は、user=my_portal_db_username を指定する。</p> <p>wliPool のプロパティを指定する場合、使用しているデータベースの種類に応じて db_settings.properties ファイル内の wlidb_login の値をコピーする。たとえば、Oracle Thin ドライバを使用している場合は、user=my_wli_db_username を指定する。</p>

表 2 Oracle または MS SQL Thin ドライバを使用する場合の接続プールの値 (続き)

ACLName	空白。
Password	<p>[変更] をクリックする。編集中の接続プールのデータベース ユーザ パスワードを入力し、確認のためにもう一度入力するよう求めるメッセージが表示される。</p> <p>commercePool または dataSyncPool のパスワードを指定する場合は、使用しているデータベースの種類に応じて db_settings.properties ファイル内の dbpassword の値をコピーする。</p> <p>wliPool のプロパティを指定する場合、使用しているデータベースの種類に応じて db_settings.properties ファイル内の wliadbpassword の値をコピーする。</p>

10. Console で、[domain_name | Compatibility Security | Realms | wlsRealm] の順に移動します。
11. [Database] タブで、前の手順で入力したものと同一 [Driver Classname] と [URL] を表 2 の説明に従って入力します。使用しているデータベースの種類に応じて、db_settings.properties ファイルの dblogin の値をコピーします。
12. [適用] をクリックします。
13. [パスワード] フィールドで [変更] をクリックします。次にデータベース ユーザ パスワードを入力して、確認のためもう一度入力します。このパスワードは、db_settings.properties ファイルの dbpassword の値として指定したものと一致する必要があります。
14. [適用] をクリックします。
15. [続行] をクリックします。
16. [Schema] タブの [Schema Properties] (key=value) フィールドに、前の手順で入力したものと同一プロパティを前の表の説明に従って入力します。
17. [適用] をクリックします。

手順 6: WebLogic Server の起動

ドメイン ディレクトリで、次の該当スクリプトを実行して WebLogic Server を停止します。

- Windows の場合 : `stopWeblogic.cmd`
- UNIX の場合 : `stopWeblogic.sh`

手順 7: db_settings.properties の編集によるデータベースのコメントの解除

手順 2: データベース環境に合わせた `db_settings.properties` の編集 で定義した値をコメント解除するには、`db_settings.properties` ファイルを編集します。

1. `db_settings.properties` ファイルを開きます。
2. 使用するデータベースのプロパティを指定している各行の先頭から、# 記号を外します。
3. PointBase データベースの設定の行の先頭に # 記号を付けてコメントアウトし、ファイルを保存します。

手順 8: create_db の実行

ドメイン フォルダ内の `create_db` スクリプトを実行して、データベースを作成し、データを入力します。`create_db.log` に出力された内容に致命的なエラーがないことを検証します。

手順 9: WebLogic Server の再起動

ドメイン ディレクトリで、WebLogic Server を再起動します。

- Windows の場合 : `startWeblogic.cmd`

- UNIX の場合: `startWeblogic.sh`

手順 10: sync の実行

ドメイン フォルダ内の `sync` コマンドを実行して、新しいデータベース データでサーバを更新します。

以上で設定は完了です。データベースを使用する Platform ドメイン アプリケーションを起動して、実行することができます。

手順 11: Oracle のみ — インデックスの再構築

`create_db` スクリプトは、WebLogic Portal テーブルのインデックスをテーブルスペース `WEBLOGIC_INDEX` に配置します。デフォルトの `WEBLOGIC_INDEX` テーブルスペースを使用していない場合は、テーブルスペース名に合わせて `rebuild_indexes.sql` を編集する必要があります。スクリプトを実行してインデックスを再構築するには、以下の手順を実行します。

1. コマンド ウィンドウで、次のディレクトリに移動します。

```
PORTAL_HOME/db/oracle/817/admin
```

ここで、`PORTAL_HOME` は WebLogic Portal ソフトウェアがインストールされているディレクトリのパス名です。一般には、`BEAHOME/weblogic700/portal` になります。

2. 次のコマンドを実行して、SQL*Plus セッションを開始します。

```
sqlplus username/password@net_service_name
```

- `username` は、Oracle ユーザ アカウントの名前です。デフォルトでは `WEBLOGIC` に設定されています。
- `password` は、Oracle ユーザ アカウントのパスワードです。デフォルトでは `WEBLOGIC` に設定されています。
- `net_service_name` は、Oracle データベースに定義した Net Service の名前です。

3. 次のコマンドを実行して、インデックスを再構築します。

```
@rebuild_indexes.sql
```

ベスト プラクティス

この節では、WebLogic Platform のインストール、[コンフィグレーション] ウィザードの使用、すぐに使用できる Example の実行のためのベスト プラクティスに基づいたヒントを紹介しします。

- BEA ホーム ディレクトリ名の文字制限
- インストールディレクトリに名前を付ける
- WebLogic Portal サンプルデータ
- Platform Domain テンプレートを使用する
- WebLogic Platform サンプル アプリケーションを実行する
- Netscape ブラウザ 6.x を使用してドキュメントにアクセスする
- WebLogic Platform 7.0 SP2、SP4、および SP5 の JVM オプションを指定する

BEA ホーム ディレクトリ名の文字制限

BEA ホーム ディレクトリに名前を付けるときは、12 文字以下にすることをお勧めします。最大値である 12 文字近くまで使用した場合、CLASSPATH 解決が正しく実行されないおそれがあります。

インストール ディレクトリに名前を付ける

WebLogic Platform は、名前にスペースを使用していないディレクトリにインストールすることをお勧めします。インストールディレクトリの名前にスペースが含まれていると、サンプルおよび E-Business Control Center (EBCC) の起動スクリプトで問題が発生するおそれがあります。

WebLogic Portal サンプル データ

『*BEA WebLogic Platform のインストール*』

(<http://edocs.beasys.co.jp/e-docs/platform/docs70/install/index.html>) で説明したように、E-Business Control Center (EBCC) または WebLogic Portal Example のいずれかをインストールすると、WebLogic Portal のサンプル データがインストールされます。ただし、WebLogic Portal Example を実行する前に、WebLogic Portal サーバをインストールする必要があります。

注意： EBCC または WebLogic Portal Example をアンインストールする際、WebLogic Portal のサンプル データはシステムから削除されます。

Platform Domain テンプレートを使用する

Platform Domain テンプレートを使用してすべての WebLogic Platform コンポーネントをサポートするドメインを作成するときは、以下のガイドラインに留意してください。

- Platform Domain Cluster を作成するときは、管理コンソールで、`b2bconsole.war` のターゲットをクラスタから管理サーバに変更する必要があります。
- [コンフィグレーション] ウィザードの実行後にユーザ名とパスワードを変更する場合は、`sync.cmd` または `sync.sh` を手動で変更する必要があります。
[コンフィグレーション] ウィザードを実行してドメインを作成した後に、管理コンソールでユーザ名およびパスワードを変更した場合は、`BEA_HOME\user_projects\domain\beaApps\portalApp-project` にある `sync.cmd` スクリプトまたは `sync.sh` スクリプトも同様に変更する必要があります。
- `portalApp` ツールにアクセスする前に、`sync.cmd` または `sync.sh` を実行します。
サーバの起動後、`sync.cmd` スクリプトまたは `sync.sh` スクリプトを実行して、ドメイン内にデプロイされる WebLogic Portal アプリケーションに必要な実行時データをロードします。
- ドメイン名は、以下のファイル内で `platformDomain` としてハードコード化されているので、変更しないでください。
 - `create_db.cmd`
 - `create_db.sh`
 - `create_wli.cmd`

- create_wli.sh
- db_settings.properties

ドメイン名のこの設定により、**WebLogic Integration create_db** スクリプトで platformDomain に必要な DDL をロードできるようになります。これらの値を変更しないでください。

- BEA により提供されるアプリケーションは、
BEA_HOME\user_projects\domain\beaApps に保存されています。ただし、
WebLogic Workshop IDE を使用して作成される新規アプリケーションはすべて、
BEA_HOME\user_projects\domain\applications ディレクトリ内に作成されます。

WebLogic Platform サンプル アプリケーションを 実行する

この節では、WebLogic Platform に付属する WebLogic Platform サンプル アプリケーション
を実行する際のベスト プラクティスを紹介します。サンプルを実行するための具体的な手
順については、次の URL の『*BEA WebLogic Platform サンプル アプリケーションのツ
アー*』を参照してください。

<http://edocs.beasys.co.jp/e-docs/platform/docs70/tour/index.html>

- WebLogic Platform をインストールした後、create_db スクリプトを実行して
WebLogic Integration データベースの手順に従ってデータベースを切り替えた場合は、
config.xml ファイルを編集して E2EappView_conneccionFactory Application
Integration (AI) アプリケーションを削除します。
- サーバのポートを変更した場合は、アプリケーション ビューを再デプロイします。
e2eSetupAppView のポート番号を変更した後、それを実行してアクティブ サーバにデ
プロイします。
- サンプル アプリケーション実行時に Doc Portlet リンクにアクセスして技術情報を参照
するには、リンクを右クリックし、[リンクを新しいウィンドウで開く]を選択しま
す。ブラウザの [戻る] ボタンを使用すると実行時エラーが発生する場合があるため、
代わりにこの方法をお勧めします。
- Solaris で QuickStart ページから [Live Platform ツアー] を開始するには、まず URL を
ブラウザにコピーします。

- サンプル アプリケーションを実行する際、新しいパスワード、ポート番号などでコンフィグレーションを変更しないでください。パッケージ化されたコンフィグレーションに対する変更はサポートされていません。

Netscape ブラウザ 6.x を使用してドキュメントにアクセスする

Netscape 6.x を使用して WebLogic Platform オンラインドキュメントを表示すると、フォーマットやフォントに不具合が発生する場合があります。最適な表示のためには、Netscape 4.7 または 5.x の使用をお勧めします。

WebLogic Platform 7.0 SP2、SP4、および SP5 の JVM オプションを指定する

WebLogic Platform 7.0 SP2、SP4、および SP5 では、JVM に必要な `-client` などのオプションを簡単に指定できる `commEnv` スクリプトが用意されています。

`commEnv` スクリプトは、クライアントとサーバのどちらの JVM を実行すべきかを判断する、次の環境変数を定義します。

- `COMM_CLIENT_VM`
- `COMM_SERVER_VM`
- `COMM_VM`

7.0 SP4 および SP5 では、Java 実行可能ファイル (WebLogic JRockit JVM または Sun JVM) に渡される標準のメモリ引数を設定する次の `commEnv` 環境変数が、WebLogic Platform にも含まれています。

- `COMM_SMALL_MEM_ARGS`
- `COMM_MEDIUM_MEM_ARGS`
- `COMM_LARGE_MEM_ARGS`

以下の表は、`commEnv` 環境変数の使い方を示しています。どの JVM オプションが有効で利用可能であるかは、使用しているプラットフォームと実行する Java 実行可能ファイルによって異なります。

表 3 commEnv 環境変数の使い方

使用する環境	対象となるアプリケーションの種類	例
COMM_CLIENT_VM	クライアント (GUI など)	-hotspot または -client
COMM_SERVER_VM	サーバ (WebLogic Server など)	-server または -jrockit
COMM_VM	クライアントまたはサーバ。アプリケーションに COMM_CLIENT_VM や COMM_SERVER_VM が必要な場合に推奨される設定。	COMM_CLIENT_VM または COMM_SERVER_VM
COMM_SMALL_MEM_ARGS	クライアントまたはサーバ	WebLogic JRockit JVM: -Xms64m -Xmx64m Sun JVM: -Xms64m -Xmx64m -XX:MaxPermSize=64m
COMM_MEDIUM_MEM_ARGS	クライアントまたはサーバ	WebLogic JRockit JVM: -Xms128m -Xmx128m Sun JVM: -Xms128m -Xmx128m -XX:MaxPermSize=128m
COMM_LARGE_MEM_ARGS	クライアントまたはサーバ	WebLogic JRockit JVM: -Xms256m -Xmx256m Sun JVM: -Xms256m -Xmx256m -XX:MaxPermSize=256m

注意： SP4 および SP5 と共に WebLogic JRockit JVM を使用している場合は、WebLogic JRockit を起動したときに、有効なコマンドライン オプションを使用しているかどうか確認してください。無効なオプション (特に -xx オプション) を使用している

場合は、WebLogic JRockit が終了します (前バージョンでは、無効なオプションは無視されます)。有効なコマンドライン オプションの一覧については、『BEA WebLogic JRockit 7.0 SDK User Guide』を参照してください。

commEnv 変数は、WebLogic Platform サンプルと、[コンフィグレーション] ウィザードで作成されたカスタム ドメインの両方に使用されます。

JVM オプションは、commEnv の環境変数を使用して設定することをお勧めします。使用しているプラットフォームに適したコマンドを参照してください。

- Windows の場合 : `BEAHOME\weblogic700\common\bin\commEnv.cmd`
- UNIX の場合 : `BEAHOME/weblogic700/common/bin/commEnv.sh`

環境変数をハード コーディングする代わりに commEnv を使用すると、スクリプトを実行するときに、プラットフォームまたは WebLogic JRockit SDK を指定する必要がなくなります。

サンプル スクリプト

次の Windows スクリプトは、commEnv.cmd を使用して WebLogic JRockit JVM を実行します。

```
SET WL_HOME=c:\bea\weblogic700SET JDK_HOME=c:\bea\jrockit70sp5_131_10CALL %WL_HOME%\common\bin\commEnv.cmd "%JDK_HOME%\bin\java" %COMM_SERVER_VM% myclass
```

この例では、WL_HOME は WebLogic Platform がインストールされたディレクトリを示します。

修正された問題点

この節では、WebLogic Platform で検出された問題の解決策について説明します。特に、次の点について説明します。

- Service Pack 1 で修正された問題点
- Service Pack 2 で修正された問題点
- Service Pack 4 で修正された問題点

■ Service Pack 5 で修正された問題点 END

サービス パックの内だけの製品変更が行われる場合があるため、解決された問題の一覧はすべてが含まれているわけではありません。製品で問題が発生した場合は、問題の根本原因を調査分析する手助けと正しい解決を見つけるために、BEA カスタマサポートにお問い合わせください。

WebLogic Platform 製品コンポーネントで修正された問題を確認するには、WebLogic Server、WebLogic Workshop、WebLogic Integration、WebLogic Portal、および WebLogic JRockit の各リリース ノートを参照してください。

Service Pack 1 で修正された問題点

以下の表は、BEA WebLogic Platform 7.0 SP1 で修正された問題点とその CR (変更要求) 番号のうちのいくつかを示しています。これらの問題点の一部は、元は BEA WebLogic Platform 7.0 の『リリースノート』に記載されていたものです。

表 4 BEA WebLogic Platform 7.0 SP1 で修正された問題点

変更要求番号	解説
CR078423	起動スクリプトでサーバを起動できない。 この問題は、入力行の長さが原因で発生することがあります。CLASSPATH の長さが縮小されています。BEA_HOME の定義は、最大 10 から 12 文字に制限する必要があります。
CR078737	Platform Domain テンプレートは、シングル サーバだけでなくクラスタ化をサポートする必要がある。
CR080029	WebLogic Integration チュートリアル QuickStart リンクが、誤ったドキュメントにリンクされている。
CR080377	Netscape 4.7 で、b2c から実行時エラーが送出される。ポートレット チェックアウト用のポートレットの状態を検索できない。

Service Pack 2 で修正された問題点

以下の表は、BEA WebLogic Platform 7.0 SP2 で修正された問題点とその CR (変更要求) 番号のうちのいくつかを示しています。これらの問題点の多くは、元は BEA WebLogic Platform 7.0 または BEA WebLogic Platform 7.0 Service Pack 1 の『リリースノート』に記載されていたものです。

表 5 BEA WebLogic Platform 7.0 SP2 で修正された問題点

変更要求番号	解説
CR080423	UNIX プラットフォームで Netscape ブラウザを使用すると、QuickStart の画面が空白で表示される。このため、Netscape ブラウザが起動されている間は QuickStart にアクセスできない。
CR080526	e2eAppTools の説明で HTTP 404 エラーが返される。
CR081153	一般的なインストーラで、インストールを完了するための十分な空き容量がないことを検出できない。
CR081523	Linux 以外の UNIX プラットフォーム (AIX、HP-UX、Solaris) に、WebLogic Workshop Application View Control をインストールできない。
CR081842	実行中のサーバを停止するためにドメイン内で停止コマンドを実行すると、例外が <code>weblogic.log</code> ファイルに書き込まれる。
CR82012	<code>WebservicesEJB.jsp</code> が、WLS Example ドメイン テンプレートに基づいて [コンフィグレーション] ウィザードで作成されたドメインでコンパイルを行わない。
CR083994	<code>startWebLogic.sh</code> が <code>nopointbase</code> 引数を処理しない。
CR083997	複数あるドメイン テンプレート (Platform、WLP、WLS 用など) のうちのいずれかを使ってドメインを生成すると、デフォルトでリスン アドレスが <code>BEA_HOME\user_projects\domain</code> の <code>config.xml</code> ファイルに追加されない。
CR084109	WebLogic Platform のクラスタ化されたドメイン : 管理対象サーバをシャットダウンしようとする時、例外が管理サーバのログ ファイルに記録される。
CR084155	QuickStart アプリケーションが Red Hat Linux 7.2 プラットフォームで正しく表示されない。

表 5 BEA WebLogic Platform 7.0 SP2 で修正された問題点（続き）

変更要求番号	解説
CR084292	WLP テンプレート：管理対象サーバの SSL 情報が、管理サーバの config.xml ファイルに保存されていない。
CR084464	[コンフィグレーション] ウィザードの Platform Domain テンプレートで表明例外が発生する。
CR084695	UNIX システム上の WebLogic Platform サンプルアプリケーションでエラーが発生する。
CR085203	RosettaNet2 のファイアウォール コンフィグレーションおよび IIS プロキシで SSL 例外が発生する。
CR085531	GPR (Global Product Registry) を更新して、WebLogic Workshop を、適格なリリースすべてのコンポーネントとして反映させる。
CR085583	[標準インストール] を選択しても完全インストールが行われない。
CR087360	Windows 対応の JRockit 7.0 SP1 SDK で Portal EBCC コンポーネントがサポートされない。
CR087956	JRockit 7.0 SP1 SDK 版 Red Hat Linux Advanced Server 2.1 で WebLogic Integration サンプルまたは WebLogic Integration RosettaNet サンプルを実行しているときに SIGHUP を受信すると、JRockit がデッドロックしたりクラッシュしたりする。
CR090148	JRockit の他に PointBase を使って WebLogic Integration サンプルを実行していると、途中で PointBase サーバが突然動作不能になる。
CR090186	WLS 7.0 SP1 - java.sql.SQLException: ORA-01591: 疑わしい分散トランザクションによるロック。
CR091228	ファイアウォール RN2.0 のテストで NPE (NullPointerException) が発生する。
CR091239	WLW 7.0 SP1 - Application View Control でイベントが実行されると、多数のトランザクションがロールバックする。
CR091702	復帰：Solaris で管理対象サーバを実行しているときにコア ダンプが出力される。
CR092483	WLS SP2 版 JRockit で、サーバの起動およびシャットダウン時に問題が発生する。

表 5 BEA WebLogic Platform 7.0 SP2 で修正された問題点（続き）

変更要求番号	解説
CR092681	ORA-01591: BPM 回復テスト中の Oracle Thin XA ドライバを使用した疑わしい分散トランザクションによるロック。
CR092703	ファイアウォール RN2.0 のテストで certicom により NPE (NullPointerException) が発生する。
CR092742	RN11 ワークフローをクラスタ内で完了できない。
CR092851	XML ドキュメントの DTD 部分を使用すると、XML パーサによって CPU が 100% 消費されたり、メモリが大量に消費される可能性がある。
CR092895	新しい BPM 機能を使用するには、ドメインをアップグレードするための手順が必要である。
CR093003	async からのコールバックが起動しない。
CR093007	aiimportexport スクリプトを使用して Application View をインポートすると例外が発生する。
CR093116	AdapterDesignTimeTestCase ログイン方法が JRockit で機能しない。
CR093119	AdapterDesginTimeTestCase ログイン方法が JRockit で機能しない。
CR093127	AppView をクラスタ内の管理対象サーバの 1 つにデプロイできない。
CR093195	Oracle 9.2.0.1 Thin XA ドライバと 9.2.0.1 サーバを使用すると、WLI 回復時に XA エラーが発生する。
CR093280	WebLogic Platform 7.0sp2 ja アップグレード インストーラの実行時に xmlx.jar が更新されない。
CR093281	組織 ID が日本語で指定されていると、Post XML Event によってワークフローが起動されない。
CR093307	JRockit: async からの AppView 制御コールバックが起動しない。
CR093382	管理サーバがダウンした後、以前のコンフィグレーション ファイルを使って管理対象サーバを起動できない。
CR093510	RN2Security サンプルが Solaris で outOfMemory エラーとなり異常停止する。

表 5 BEA WebLogic Platform 7.0 SP2 で修正された問題点（続き）

変更要求番号	解説
CR093663	Exclusive の同時実行戦略を使ったエンティティ EJB が、クラスタ間でロックされない。
CR093724	管理対象サーバ (WLI-Other) を停止する ([Ctrl] + [C]) と、JVM がクラッシュ (コア ダンプ) する。
CR093753	HTTP アダプタによるリポジトリのアクセス エラーが発生する。
CR093828	WLS 6.1 SP4 - RJVM のハートビート信号が途絶えたために EJB 層が ConnectionManager を閉じると、Web 層と EJB 層の通信が異常停止する。
CR093929	管理サーバがダウンした後、以前のコンフィグレーションファイルを使って 1 台の管理対象サーバを起動できない。
CR094008	JVM が Application View をクラスタ内の管理対象サーバの 1 つにデプロイできない。
CR095162	PosixSocketMuxer でなく、Windows XP の NTSocketMuxer がロードされる。

Service Pack 4 で修正された問題点

以下の表は、BEA WebLogic Platform 7.0 SP4 で修正された問題点とその CR (変更要求) 番号のうちのいくつかを示しています。これらの問題点の多くは、元は BEA WebLogic Platform 7.0 または BEA WebLogic Platform 7.0 SP1 および SP2 の『リリースノート』に記載されていたものです。

表 6 BEA WebLogic Platform 7.0 SP4 で修正された問題点

変更要求番号	解説
CR094041	競合状態により、JTA および JMS の移行に失敗することがあります。
CR094109	Apache プラグインのフェイルオーバー? 機能が動作しない。
CR094722	管理対象サーバの起動中に、クラスタ内の管理サーバで socketWrite エラーが発生する。

表 6 BEA WebLogic Platform 7.0 SP4 で修正された問題点 (続き)

変更要求番号	解説
CR099887	Smart Update が HP-UX で動作しない。
CR100933	JRockit 7.0 SP2 RP1 が <code>-XX:MaxNewSize</code> オプションを認識しない。
CR101796	テンプレートをサイレント モードで使用すると、Platform テンプレートが問題のあるドメインを作成する。

Service Pack 5 で修正された問題点 END

以下の表は、BEA WebLogic Platform 7.0 SP5 で修正された問題点とその CR (変更要求) 番号を示しています。

表 7 BEA WebLogic Platform 7.0 SP5 で修正された問題点

変更要求番号	解説
CR120193	Linux インストーラで SmartUpdate を実行中に、警告メッセージが表示される。

既知の制約

この節では、BEA WebLogic Platform の現在のサービス パックで認識されている制約について説明します。制約の説明の後に、回避策がある場合はその方法も記載します。

- 追跡された問題
- WebLogic JRockit SDK のサポート上の制約
- JVM コンフィグレーション
- パフォーマンス上の注意点
- DB2 の注意点
- Apache 2.0.43 の注意点

WebLogic Platform 製品コンポーネントの既知の問題点を確認するには、WebLogic Server、WebLogic Workshop、WebLogic Integration、WebLogic Portal、および WebLogic JRockit の各リリース ノートを参照してください。

追跡された問題

以下の表に、このサービス パックで確認されている制約の一覧を記載します。それぞれの問題には、**CR** (変更要求) 番号が付いています。これらの番号は、**BEA** が開発過程で問題の解決を追跡する際に使用されます。未解決の問題を追跡する場合は、**BEA** カスタマ サポートにお問い合わせください。その際には、該当する **CR** 番号をお知らせください。

連絡先については、次の **URL** にアクセスしてください。

<http://www.beasys.co.jp/service/support/index.html>

1 CR072310 インストーラで、非表示の .Workshop ファイルが削除されない。

問題 WebLogic Platform をアンインストールする際、非表示の .Workshop ファイルが削除されません。 .Workshop ファイルには、WebLogic Workshop のビジュアルな開発環境用のユーザ固有のコンフィグレーション パラメータが含まれています。このため、再インストール後に WebLogic Workshop を起動しようとする、エラーが発生することがあります。これは、WebLogic Workshop がファイル内の古いコンフィグレーション情報を参照するためです。

.Workshop ファイルの詳細については、以下の URL を参照してください。
<http://edocs.beasys.co.jp/e-docs/workshop/docs70/help/index.html#reference/configfiles/conDotWorkshopConfigurationFile.html>

プラットフォーム すべて

回避策 WebLogic Workshop を再インストールする前に、非表示の .Workshop ファイルを削除します。

.Workshop ファイルは、ユーザのホーム ディレクトリにあります。
Windows プラットフォームでは、ユーザのホーム ディレクトリは、%USERPROFILE% 環境変数によって指定されます。UNIX および Linux プラットフォームでは、.Workshop ファイルは、~(チルダ)ディレクトリ(または \$HOME 環境変数)に保存されています。

2 CR072761 [Ctrl] + [C] を押して WebLogic Portal サーバを停止したときに、PointBase サーバが自動的に終了しない。

問題 停止スクリプトを実行したり、[Close] アイコンをクリックしてコマンド ウィンドウを閉じるのではなく、[Ctrl] + [C] を押して WebLogic Portal サーバを停止した場合に、PointBase サーバがバックグラウンドで実行し続けます。

プラットフォーム すべて

回避策 停止スクリプトを実行するか、[Close] アイコンをクリックしてコマンド ウィンドウを閉じて、サーバを停止します。[Ctrl] + [C] でコマンド ウィンドウを閉じてサーバを停止する場合は、バックグラウンドの java.exe プロセスを終了させて PointBase を停止します。

3	CR076141	Solaris のユーザ ドメインに、startWeblogic ファイルの実行可能な権限がない。
問題	BPM Domain、EAI Domain、および WLI Domain テンプレートを使ってユーザ ドメインが作成された後、 <code>user_domain</code> 内の一部のファイル (<code>wliconfig</code> 、 <code>startWeblogic</code> など) に実行権限が設定されていません。	
プラットフォーム	Solaris	
回避策	ドメイン内でスクリプトを実行する前に、スクリプトに実行権限があることを確認してください。実行権限がない場合は、 <code>chmod</code> コマンドを使って実行権限を与えます。たとえば、 <code>startWebLogic.sh</code> スクリプトに実行権限を与えるには、次のコマンドを入力します。 <pre>chmod +x startWebLogic.sh</pre> 詳細については、 <code>chmod</code> の <code>man</code> ページを参照してください。	
4	CR078315	wliconfig (コンソール モード) コマンドが Solaris 上で成功しない。
問題	<code>wliconfig</code> コマンドを実行した結果として Solaris プラットフォームで PointBase データベースが実行されている状態で、 <code>wliconfig</code> コマンドを再実行しようとする、コマンドは成功しません。	
プラットフォーム	UNIX (Solaris)	
回避策	この状態を回避するには、 <code>wliconfig</code> コマンドを再実行する前に、PointBase サーバをシャットダウンします。ただし、過去に選択したデータベース タイプに対して、 <code>wliconfig</code> コマンドを使って既存のパラメータを変更することはできません。	

5 CR078518 **[コンフィグレーション] ウィザードで選択可能なオプションのうち、WebLogic Integration でサポートされないものがある。**

問題

BPM Domain、EAI Domain、WLI Domain テンプレートでは、[コンフィグレーション] ウィザードの 4 つのサーバタイプのうち 2 つしかサポートしません。以下のいずれかのサーバタイプを選択する必要があります。

- Single Server (Standalone Server)
- Admin Server with Clustered Managed Server(s)

管理対象サーバまたは管理対象サーバのオプションは、管理サーバに対してコンフィグレーションが無効になります。

プラットフォーム

すべて

回避策

BPM Domain、EAI Domain、WLI Domain のテンプレートが使用されている場合は、以下のいずれかのサーバタイプを選択してください。

- Single Server (Standalone Server)
- Admin Server with Clustered Managed Server(s)

以下のサーバタイプは選択できません。

- Admin Server with Managed Server(s)
 - Managed Server (with owning Admin Server Configuration)
-

-
- 6 CR078529 **wliconfig** コマンドが同じデータベースに対して発行されている場合、データベース コンフィグレーション オプションを **wliconfig** コマンドに切り替えると 2 度目には失敗する。
-

問題 設計では、過去に選択したデータベース タイプに対して、**wliconfig** コマンドを使って既存のパラメータを変更することはできません。たとえば、データベース タイプとして **Oracle** を選択している場合、**Oracle** データベースのパラメータを変更するのに、**wliconfig** コマンドを使用することはできません。ただし、別のデータベース タイプに切り替えるのにコマンドを使用することは可能です。

プラットフォーム すべて

回避策 **setDBVars** ファイルおよび **setDBVarsExt** ファイルを編集して、データベース接続パラメータを設定または変更します。**setDBVars** ファイルおよび **setDBVarsExt** ファイルが保存されている場所は以下のとおりです。

- `SAMPLES_HOME\integration\config\samples\dbInfo\database_type`
- `SAMPLES_HOME\integration\config\samples\RN2Security\config\peer1\dbInfo\database_type`
- `SAMPLES_HOME\integration\config\samples\RN2Security\config\peer2\dbInfo\database_type`

上記のパス名で、**SAMPLES_HOME** はサンプルのインストールディレクトリで、**database_type** は接続しているデータベースのタイプです。

setDBVars ファイルおよび **setDBVarsExt** ファイルの詳細については、『**WebLogic Integration** の起動、停止およびカスタマイズ』の「**WebLogic Integration** コマンド」を参照してください。

7 CR078971 Platform Domain テンプレートのアイコンが [コンフィグレーション] ウィザードに表示されない。

問題 WebLogic Platform サンプル アプリケーション (Platform ツアー) をアンインストールすると、[コンフィグレーション] ウィザードで Platform Domain テンプレートを使用できなくなります。

プラットフォーム すべて

回避策 WebLogic Platform サンプル アプリケーション (Platform ツアー) がインストールされていることを確認します。以下のアプリケーションもインストールされている必要があります。

- WebLogic Server
 - WebLogic Integration
 - WebLogic Portal
 - WebLogic Workshop
 - WebLogic Workshop サンプル
-

8 CR079101 PointBase 接続が閉じていないことをユーザに示す必要がある。

問題 PointBase コマンド ウィンドウが表示されなくなったため、ユーザは PointBase が稼動中であることを知る手段がありません。サーバが停止しても、PointBase 接続はすぐには終了されません。PointBase 接続が閉じる前に PointBase のサーバを再起動しようとする、データベース接続を作成できないことを示すエラー メッセージが表示されます。

プラットフォーム すべて

回避策 PointBase 接続の終了時、接続が閉じるまで 4、5 分かかることがあります。サーバを再起動しようとしてエラー メッセージが表示されたら、2、3 分待つてからもう一度実行してください。

9	CR079490	SecurityConfiguration 資格の変更時に java.security.PrivilegedActionException がログ記録される。
問題	config.xml ファイルの SecurityConfiguration 要素を変更して、その後サーバを起動しようとする、java.security.PrivilegedActionException がログ記録されます。	
プラットフォーム	すべて	
回避策	サーバをシャットダウンし、_ServletContextidnameDbmsEvent*****.xml ファイルおよび _ServletContextidnameDbmsEvent*****_temp ファイルを e2eDomain から削除します。その後サーバを再起動します。	

10	CR079492	サーバが他のポートで起動されると、アプリケーションビューの E2EAppView.sav がデプロイに失敗する。
問題	e2eAppView は、ポート 7501 と密接に結合されています。このポート番号は、e2eAppView のデプロイ時に使用される e2eAppViewDeployer.java ソース ファイルで定義されています。	
プラットフォーム	すべて	
回避策	7501 以外のポートでアプリケーションビューをデプロイするには、e2eSetupAppView.cmd スクリプトまたは e2eSetupAppView.sh スクリプト内、および e2eSetupAppView スクリプトから呼び出される SetSampleData スクリプト内で、すべてのポート番号参照を 7501 から新しいポート番号に変更する必要があります。	

11 CR079752	[コンフィグレーション] ウィザードおよび WLI テンプレートで作成したドメイン内で停止スクリプトを使用すると、PointBase がシャットダウンしない。
問題	[コンフィグレーション] ウィザードおよび WLI テンプレートを使用して作成したドメインに配置されている stopWeblogic スクリプトで、PointBase がシャットダウンされません。
プラットフォーム	すべて
回避策	stoppointbase スクリプトを使用して PointBase をシャットダウンしてください。
12 CR079917	一部のユーザドメインのショートカットが作成されない。
問題	[コンフィグレーション] ウィザードをインストール処理の一部として実行すると、[スタート] メニューにごく少数のユーザドメイン ショートカットしか作成されません (数はドメインの作成順序によって決まる)。
プラットフォーム	Windows NT (SP6)
回避策	[コンフィグレーション] ウィザードをインストール処理とは別に起動して、ユーザドメインを作成します。インストール処理とは別に [コンフィグレーション] ウィザードを起動してユーザドメインを作成すると、この問題は発生しません。

13 CR080402 インストール時、QuickStart にシェルがない。

問題 Solaris 版インストールの最後に QuickStart を実行すると、QuickStart が起動してインストールプログラムが終了しますが、シェルは実行されないままです。QuickStart で起動したサーバから出力があっても、それを受信するシェルが存在しないため、リンクをクリックしたときに、サーバが正常に起動したかどうか、あるいはいつ起動したかを通知できません。

QuickStart ページのほとんどのリンクでは、サーバの起動完了時、アプリケーションごとに個別のブラウザが起動します。ただし、[Portal Example サーバ] をクリックした場合、サンプルを実行するには 2 番目のリンク [Portal Example を起動] をクリックする必要があります。サーバの起動が完了したことを見分けるのは難しいため、正しいタイミングで [Portal Example を起動] リンクをクリックしてサンプルを実行することは容易ではありません。

プラットフォーム UNIX (Solaris)

回避策 *BEA_HOME*\weblogic700\common\bin ディレクトリにある quickstart.sh スクリプトから QuickStart を起動します。対応する weblogic.log ファイルをチェックして、サーバが起動しているかどうかを確認することもできます。たとえば、Portal Example サーバの weblogic.log ファイルは、*BEA_HOME*\weblogic700\samples\portal\sampleportalDomain\logs にあります。

**14 CR082039 ログ ファイルにエラー メッセージが表示される :
ManagedConnectionFactory**

問題 アップグレード インストーラの使用中、以下のようなエラー メッセージがログ ファイルに表示されることがあります。

```
####<Jul 17, 2002 11:30:49 AM MDT> <Error> <Connector>  
<joe-2k> <e2eServer> <Thread-7> <kernel identity> <> <190004>  
<ManagedConnectionFactory not found for jndiName  
com.bea.wlai.connectionFactories.E2EAppView.sav_connectionF  
actoryInstance.>
```

**プラット
フォーム** すべて

回避策 このエラー メッセージは、製品の機能には影響しないので無視してください。このエラー メッセージは、コードのデプロイメント 順序が原因で表示されますが、悪影響はありません。

15 CR082062 QuickStart で Netscape 4.7x if LANG=C を表示できない。

問題 UNIX プラットフォーム (Solaris と HP-UX のみ) で QuickStart を起動した後にリンクを選択すると、Web ブラウザを選択することを要求するダイアログ ボックスが表示されることがあります。正しい Netscape ブラウザを選択しても、QuickStart でアプリケーションを起動できないことがあります。

**プラット
フォーム** UNIX (Solaris と HP-UX のみ)

回避策 QuickStart を終了し、(手動で、またはラッパー機能を使って) LANG 環境変数の割り当てを解除してから、QuickStart アプリケーションを再起動します。

16	CR083467	wliconfig.cmd を使用してデータベースを作成すると、断続的に <code>ArrayIndexOutOfBoundsException</code> が発生する。
問題	<p>wliconfig スクリプトを実行すると、以下のエラーメッセージが表示されることがあります。</p> <pre>Exception occurred during event dispatching: java.lang.ArrayIndexOutOfBoundsException: No such child: 0</pre> <p>このエラーには害はありません。選択したタイプのデータベースが問題なく作成されます。</p>	
プラットフォーム	すべて	
回避策	このエラーメッセージは、製品の機能には影響しないので無視してください。	
17	CR084490	wliconfig ウィンドウが Red Hat Linux 7.2 上で縮小される。
問題	<p>Red Hat Linux 7.2 が稼動するシステム上で、最初に wliconfig ツールが GUI モードで起動されると、ウィンドウが標準境界線のついたゼロサイズ ウィンドウとして表示されます。</p>	
プラットフォーム	UNIX (Red Hat Linux 7.2)	
回避策	ウィンドウの境界線を選択して、ウィンドウのサイズを変更します。ウィンドウはこれを正しいサイズと見なします。	
18	CR086668	-Xthinthreads を使用して JRockit 7.0 SP1 SDK を実行することができない。
問題	<p>JRockit 7.0 SP1 は、J2SE 1.3.1 での -Xthinthreads オプションの使用をサポートしていません。したがって、このオプションは、WebLogic Platform 7.0 SP1 を実行するときに使用できません。</p>	
プラットフォーム	なし	
回避策	なし	

19 CR094185 データベースの接続を作成しているときに、管理対象サーバで例外エラーが発生する。

問題 [コンフィグレーション]ウィザードでは、localhost が、生成される config.xml 内の JDBCConnectionPool および RDBMSRealm のデータベース URL として使用されます。これにより、データベースのホスト マシンでないマシンでサーバ (管理サーバを含む) を実行すると、問題が発生します。たとえば、クラスタ化された platformDomain を生成すると、管理サーバのホスト マシンでないマシンに常駐する管理対象サーバは起動しません。

プラットフォーム なし

回避策 使用するドメインの種類に応じた手順に従って、「localhost」をデータベースの DNS 名に変更します。Platform テンプレートで作成したドメインを使用している場合は、38 ページの「Platform Domain で使用されたデータベースの切り替え」に記載される手順に従ってください。その他のテンプレートで作成したドメインを使用している場合は、BEA WebLogic Platform マニュアルの『コンフィグレーションウィザードの使い方』を参照してください。
<http://edocs.beasys.co.jp/e-docs/platform/docs70/configwiz/index.html>

20 CR094921 WLI 7.0 SP2 Samples ドメインで、ポート番号 "7001" がハードコーディングされている。

問題 ポート番号 7001 はよく使用される番号です。この番号は、WLI Samples ドメインだけでなく、WLW および WLS サンプルでも使用されます。この番号が複数のコンポーネントでハードコーディングされていることを覚えておいてください。

プラットフォーム なし

回避策 問題が発生した場合は、同じマシンの他のユーザが同じポートを使用していないかどうか調べてください。

21	CR107512	プロキシで Smart Update を使用する際の問題
問題	プロキシで Smart Update を使用する際に問題が発生します。Smart Update の処理中に、ハード コーディングされたプロキシ設定が実行されないか、または認識されません。	
プラットフォーム	なし	
回避策	なし	

22	CR108708	WebLogic Workshop ドメインにデプロイされたアプリケーションが参照できるのは、WebLogic Workshop に付属する log4j.jar ファイルだけ。
問題	WebLogic Workshop ドメインにデプロイされたアプリケーションが参照できるのは、WebLogic Workshop に付属する log4j.jar ファイルだけです。WebLogic Workshop は log4j をデバッグに使用して、サーバが起動するときにそのインストール済みのバージョンをロードします。他の log4j.jar ファイルを参照するアプリケーションは、デプロイメント時に失敗します。	
プラットフォーム	なし	
回避策	なし	

23	CR128767	日本語文字でマルチバイト文字セットを使用する Oracle 10g ドライバ - setString() が失敗する。
問題	Oracle10g GA ドライバには既知の問題があります。日本語文字でマルチバイト文字セットを使用していると、setString() メソッドに対するコールが失敗します。 Oracle 10g GA に対するパッチが必要です (Oracle TAR 番号 3584729.994、Oracle バグ番号 3437365)。	
プラットフォーム	なし	
回避策	WebLogic Platform 7.0 SP5 は、Oracle 10g ドライバで認定されています。このドライバはキットにバンドルされ、このパッチを含んでいます。Oracle 10g GA ドライバを使用している場合は、Oracle から関連パッチを入手することをお勧めします。	

24	CR136530	Windows 2003 上で WebLogic Workshop が起動しない。
問題	Windows 2003 プラットフォーム上では WebLogic Workshop が正しく起動しません。起動時に初期画面が表示してすぐ閉じた後で、ファイル javaw.exe がタスク マネージャ上で実行し続けます。	
プラットフォーム	なし	
回避策	なし	
25	CR172462	Oracle 9.2 で AL32UTF8 文字セットを使用する WebLogic Server jDriver の問題
問題	AL32UTF8 文字セットを使用すると、WebLogic Server jDriver は Oracle 9.2 と正しく動作しません。	
プラットフォーム	なし	
回避策	ソフトウェア パッチが入手可能です。カスタム サービス担当者にお問い合わせで、パッチを入手してください。	
26	CR173360	クラスタ内の cgJMSServer をターゲットにすると、[コンフィグレーション] ウィザードのサイレント モードが失敗する。
問題	サイレント モードで [コンフィグレーション] ウィザードを実行して、クラスタを使用する WebLogic Platform ドメインを作成すると、cgJMSServer が正しくターゲットされません。ターゲットの値を、Targets="" と設定します。このシナリオは、GUI モードで [コンフィグレーション] ウィザードを実行しているときに有効です。	
プラットフォーム	なし	
回避策	WebLogic ドメインを起動する前に、config.xml ファイルを編集して cgJMSServer をターゲットに指定します。	

27	CR173725	Oracle Thin Driver 10g の使用時での無効なバッチ値の例外
問題	Oracle Thin Driver 10g を使用すると、バッチ値が 16383 を超えることができません。	
プラットフォーム	なし	
回避策	バッチのサイズが 16383 を超えないことを確認します。あるいは、Oracle 9.2.0 ドライバを使用します。	

28	CR175635	PA-RISC および Windows 上での HP-UX 11.0 版 WebLogic Server インストール : WebLogic Workshop ドメイン作成時の問題
問題	PA-RISC および Windows 上での HP-UX 11.0 版 WebLogic Server インストールでは、[コンフィグレーション]ウィザードを使用して WebLogic Workshop ドメインを作成するときに、管理サーバのリスン ポートの入力画面で [Next] を押すと、エラー メッセージが表示されます。 例： 管理サーバのリスン ポートを入力するか、または [Exit Previous Next] の順に選択すると、次に This Installer Is Broken!! というメッセージが表示されます。	
プラットフォーム	HP-UX 11.0 PA-RISC および Windows	
回避策	管理サーバのリスン ポート入力画面が表示されたら、有効なリスン ポートアドレスを入力します。画面上で、[Next] と入力しないようにします。	

WebLogic JRockit SDK のサポート上の制約

以下は、WebLogic JRockit 7.0 SP1/SP2/SP4/SP5 SDK のサポート上の制約の一覧を示しています。

- WebLogic JRockit SDK は、JVMDI (Java Virtual Machine Debug Interface) をサポートしていません。したがって、WebLogic Workshop Server を起動するときは、`startWebLogic.cmd` または `startWebLogic.sh` コマンドラインに `nodebug` オプションを追加してください。コマンドラインは次のいずれかになります。
 - `./startWebLogic.sh nodebug`
 - `startWebLogic nodebug`
- WebLogic Workshop IDE は、`BEA_HOME/weblogic700/workshop/jdk1.4` にインストールされている Sun JRE 1.4.0 上で動作します。ただし、WebLogic JRockit SDK で JVMDI (Java Virtual Machine Debug Interface) がサポートされていないため、WebLogic Workshop Server が WebLogic JRockit SDK 上で実行されている間は、WebLogic Workshop IDE をテストしてデバッグ情報を収集することはできません。

JVM コンフィグレーション

Windows で Java HotSpot Server VM (`-server` オプション) を実行するように Java 2 SDK バージョン 1.3.1_06、1.3.1_08 および 1.3.1_10 をコンフィグレーションすると、問題が発生することが確認されています。これを回避するには、Java HotSpot Client VM を起動する `-hotspot` オプションを使用してください。たとえば、サーバを起動するために HotSpot Client を起動するには、起動スクリプトに次の行を追加します。

```
set JAVA_VM=-hotspot
```

HotSpot の詳細については、次の URL にアクセスしてください。

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

パフォーマンス上の注意点

WebLogic Platform の相対的なパフォーマンスは、SDK の選択とアプリケーションの性質によって異なる場合があります。適切なコンフィグレーション オプションについては、次のドキュメントを参照してください。

- 『WebLogic Server パフォーマンス チューニング ガイド』
<http://edocs.beasys.co.jp/e-docs/wls/docs70/perform/index.html>
- 『WebLogic JRockit 7.0 SDK Performance Tuning Guide』
<http://e-docs.bea.com/wljrookit/docs70/tuning/index.html>
- 適切な Java SDK ドキュメント

DB2 の注意点

- DB2 コンフィグレーション ファイルに `DB2_RR_TO_RS=ON` フラグを設定せずに、DB2 データベースを使用して **WebLogic Integration** サンプルを実行しても、正しく実行されません。フラグは必ず設定してください。詳細については、DB2 のマニュアルを参照してください。
- DB2 Type 2 ドライバの `COM.ibm.db2.jdbc.app.DB2Driver` には、アプリケーションドライバとネットワークドライバが両方とも含まれています。この URL は `jdbc:db2:db-alias` です。ここで、`db-alias` は、該当するデータベース インスタンスの名前を示します。

Apache 2.0.43 の注意点

HP-UX プラットフォームで、BEA Apache プラグイン `mod_wl_20.so` を使用したときに Apache サーバが停止する場合、古いバージョンの `ld` または `libdld.sl` (またはその両方) を使用している可能性があります。Apache バイナリでは、B.11.32 以上のバージョンの `ld` および `libdld.sl` が必要です。

システムにインストールされているバージョンの `ld` を判断するには、以下の手順を実行します。

1. `ld -v` と入力します。

2. 次に、`what /usr/lib/libdld.sl` と入力します。
ld および `libdld.sl` のバージョン番号が表示されます。
 - ld と `libdld.sl` がどちらも同じバージョン番号で、その番号が 11.32 以上であれば、適切なバージョンが使用されています。この手順を終了して、`mod_wl_20.so` プラグインを実行することができます。
 - ld と `libdld.sl` のバージョン番号がそれぞれ異なっている場合、または番号は同じ場合でも 11.32 より前のバージョン番号である場合は、HP パッチをインストールする必要があります。手順 3 に進みます。
3. システムに必要なパッチを以下のように確認します。
 - HP-UX 11.00 プラットフォームの場合は **PHSS_26559** が必要です。
 - HP-UX 11i (11.11) 以上のプラットフォームの場合は、**PHSS_26560** が必要です。
4. 次の該当する IT リソース・センタにアクセスします。
 - アメリカおよびアジア / 太平洋地域：<http://us-support.external.hp.com>
 - ヨーロッパ：<http://europe-support.external.hp.com>
5. Maintenance/Support で [Individual Patches] をクリックします。
6. 次の項目までスクロールします。
Retrieve a specific patch by entering the patch name
7. 入力フィールドにパッチ番号を入力します。
8. パッチをダウンロードし、システムにインストールします。