

Oracle® WebCenter Sites

FatWire Content Server バージョン 7.6 パッチ 1 または
パッチ 2 からのアップグレード

11g リリース 1 (11.1.1)

部品番号 : B69406-01

2012 年 4 月

Oracle® WebCenter Sites: FatWire Content Server バージョン 7.6 パッチ 1 またはパッチ 2 からのアップグレード, 11g リリース 1 (11.1.1)

部品番号 : B69406-01

原本名 : Oracle® WebCenter Sites: Upgrading from FatWire Content Server Version 7.6 Patch 1 or Patch 2, 11g Release 1 (11.1.1)

原本著者 : Promila Chitkara

原本協力著者 : Tatiana Kolubayev

原本協力者 : Eric Gandt, Gaurang Mavadiya, Vipin Kumar, Ankush Chadha

Copyright © 2012 Oracle and/or its affiliates. All rights reserved.

このソフトウェアおよび関連ドキュメントの使用と開示は、ライセンス契約の制約条件に従うものとし、知的財産に関する法律により保護されています。ライセンス契約で明示的に許諾されている場合もしくは法律によって認められている場合を除き、形式、手段に関係なく、いかなる部分も使用、複写、複製、翻訳、放送、修正、ライセンス供与、送信、配布、発表、実行、公開または表示することはできません。このソフトウェアのリバースエンジニアリング、逆アセンブル、逆コンパイルは互換性のために法律によって規定されている場合を除き、禁止されています。

ここに記載された情報は予告なしに変更される場合があります。また、誤りが無いことの保証はいたしかねます。誤りを見つけた場合は、オラクル社までご連絡ください。

このソフトウェアまたは関連ドキュメントが、米国政府機関もしくは米国政府機関に代わってこのソフトウェアまたは関連ドキュメントをライセンスされた者に提供される場合は、次の Notice が適用されます。

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

このソフトウェアまたはハードウェアは様々な情報管理アプリケーションでの一般的な使用のために開発されたものです。このソフトウェアは、危険が伴うアプリケーション (人的傷害を発生させる可能性があるアプリケーションを含む) への用途を目的として開発されていません。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用する場合、それを安全に使用するために、適切な安全装置、バックアップ、冗長性 (redundancy)、その他の対策を講じることは使用者の責任となります。このソフトウェアまたはハードウェアを危険が伴うアプリケーションで使用したこと起因して損害が発生しても、オラクル社およびその関連会社は一切の責任を負いかねます。

Oracle および Java は Oracle Corporation およびその関連企業の登録商標です。その他の名称は、他社の商標の可能性があり得ます。

Intel および Intel Xeon は Intel Corporation の商標または登録商標です。すべての SPARC の商標はライセンスをもとに使用し、SPARC International, Inc. の商標または登録商標です。AMD、Opteron、AMD ロゴ、AMD Opteron ロゴは、Advanced Micro Devices の商標または登録商標です。UNIX は、X/Open Company, Ltd のライセンスによる登録商標です。

このソフトウェアまたはハードウェアおよびドキュメントは、第三者のコンテンツ、製品、サービスへのアクセス、あるいはそれらに関する情報を提供することがあります。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスに関して一切の責任を負わず、いかなる保証もいたしません。オラクル社およびその関連会社は、第三者のコンテンツ、製品、サービスへのアクセスまたは使用によって損失、費用、あるいは損害が発生しても、一切の責任を負いかねます。

目次

このガイドについて	5
対象読者	5
関連ドキュメント	5
表記規則	6
サード・パーティのライブラリ	6
1 アップグレードの前に	7
アップグレード・プロセスの計画立案	8
テスト環境のアップグレード	8
概要	9
アップグレード前の手順	10
アップグレード前の意思決定	14
アップグレード後のサマリー	15
アップグレード・プロセス中の変更	15
インタフェースとアプリケーション	15
ロール	16
Page アセット・タイプと Slot アセット・タイプ	16
属性エディタ	16
プロパティの値	17
WEM の変更 (WEM 以外のインストールからアップグレードする場合)	17
Log4j	17
キャッシング・フレームワーク (以前に切り替えられていない場合)	18
カスタム要素	18
リビジョン追跡	18
データベース表	18
Web アプリケーションの変更	19
タグに対する変更	20
その他の変更	20
2 Content Server 7.6 パッチ 1 またはパッチ 2 の環境の WebCenter Sites へのアップグレード	23
Content Server の環境のアップグレード	24

手順 1. Content Server 7.6 パッチ 1 またはパッチ 2 の環境のバックアップ	24
手順 2. テスト環境の作成	25
手順 3. Content Server パッチ 1 またはパッチ 2 のアップグレード準備	27
手順 4. Content Server 7.6 パッチ 1 またはパッチ 2 のインスタンスの WebCenter Sites 11gR1 へのアップグレード	31
対話型アップグレード.....	32
サイレント・アップグレード.....	33
手順 5. アップグレード・ユーティリティの再実行	38
手順 6. アップグレードの完了	39
手順 7. 環境におけるリモートの Satellite Server のアップグレード	41
手順 8. アップグレードされた環境の検証	41
アクティブ環境のアップグレード.....	42

このガイドについて

このガイドでは、FatWire Content Server 7.6 パッチ 1 またはパッチ 2 から Oracle WebCenter Sites 11gR1 にアップグレードするプロセスについて説明します。

このガイドで説明しているアプリケーションは、旧 FatWire の製品です。命名規則は次のとおりです。

- *Oracle WebCenter Sites* は、以前は *FatWire Content Server* と呼ばれていたアプリケーションの現在の名前です。このガイドでは、*Oracle WebCenter Sites* を *WebCenter Sites* と呼ぶこともあります。
- *Oracle WebCenter Sites: Satellite Server* は、以前は *FatWire Satellite Server* と呼ばれていたアプリケーションの現在の名前です。このガイドでは、*Oracle WebCenter Sites: Satellite Server* を *Satellite Server* と呼ぶこともあります。

対象読者

このガイドは、データベース、アプリケーション・サーバーおよびコンテンツ管理製品など、エンタープライズ・レベルのソフトウェアのインストールおよび構成経験のあるインストール・エンジニアを対象としています。

関連ドキュメント

詳細は、次のドキュメントを参照してください。

- *Oracle WebCenter Sites: Apache Tomcat Application Server* へのインストール
- *Oracle WebCenter Sites: Oracle WebLogic Application Server* へのインストール
- *Oracle WebCenter Sites: IBM WebSphere Application Server* へのインストール
- *Oracle WebCenter Sites: サポート・ソフトウェアの構成*
- *Content Server 7.6 パッチ2 バックアップおよびリカバリ・ガイド*
- 『*Oracle WebCenter Sites 開発者ガイド*』
- 『*Oracle WebCenter Sites 管理者ガイド*』
- *Oracle WebCenter Sites* プロパティ・ファイル・リファレンス

表記規則

このガイドでは、次の表記規則を使用します。

- **太字**は、ユーザーが選択するグラフィカル・ユーザー・インタフェース要素を示します。
- *斜体*は、ドキュメントのタイトル、強調、またはユーザーが特定の値を指定する変数を示します。
- 等幅フォントは、ファイル名、URL、サンプル・コード、または画面に表示されるテキストを示します。
- 等幅太字フォントは、コマンドを示します。

サード・パーティのライブラリ

Oracle WebCenter Sites およびそのアプリケーションには、サード・パーティのライブラリが含まれています。詳細は、*Oracle WebCenter Sites 11gR1: サード・パーティのライセンス*を参照してください。

第 1 章

アップグレードの前に

このガイドでは、FatWire Content Server 環境のリリース 7.6 パッチ 1 または 7.6 パッチ 2 から Oracle WebCenter Sites 11gR1 へのアップグレードについて説明します。

アップグレードの前にこの章に目を通して、方針と前提条件を理解しておいてください。

この章には次の項が含まれます。

- [アップグレード・プロセスの計画立案](#)
- [概要](#)
- [アップグレード前の手順](#)
- [アップグレード前の意思決定](#)
- [アップグレード後のサマリー](#)
- [アップグレード・プロセス中の変更](#)

アップグレード・プロセスの計画立案

本番環境では、アクティブなシステムの動作が中断されるため、停止時間が長引く可能性があります。このような中断を最小限にとどめるために、FatWire Content Server リリース 7.6 パッチ 1 またはパッチ 2 から Oracle WebCenter Sites 11gR1 にアップグレードする際には、この章で概略を示す方針を採用することを強くお勧めします。

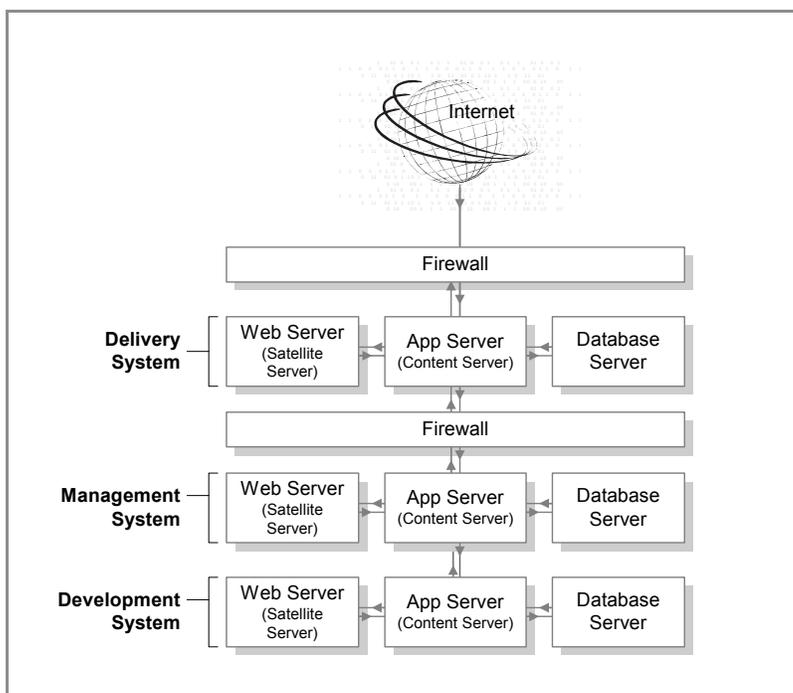
注意

開発システム、管理システム、配信システムのうち、コンテンツに対するサービスを現在提供していて、突然停止すると組織の業務および生産性に影響を与えるシステムを指す用語として、このガイドでは「アクティブ」を使用します。

テスト環境のアップグレード

企業レベルの環境は通常、WebCenter Sites を原動力とする少なくとも 3 種類のシステムで構成されています。この 3 種類とは、[図 1](#) に示されている、開発、管理、配信の各システムです。それぞれのシステムは独自のデータベース上で稼働し、他のシステムとは、パブリッシュまたは CSDT を介して対話するのが普通です。

図 1: Content Server の環境



Content Server のアクティブ環境をアップグレードする前に、複製環境で試用アップグレードを実行することが不可欠です。試用アップグレードを行うと、シ

システム固有の問題を検出して記録に残し、実際のアップグレードに先だって修正しておくことができます。

注意

7.6 パッチ 1 およびパッチ 2 の Advanced インタフェースと Dash に対する既存のカスタマイズは WebCenter Sites 11gR1 では機能しないため、試用アップグレードが不可欠です。したがって、アクティブなシステムにアップグレードしようとする前に、そのようなカスタマイズを新しい Contributor インタフェース向けに設計されたカスタマイズに置換するために、デュー・ディリジェンス (共同精査) が必要です。

複製環境が用意されていない場合は、管理システムと配信システムの中断を最小限に抑えるために、まず開発システムをアップグレードします。ただし、このアプローチは非常にリスクが高いため、停止時間が長引いても受け入れ可能と見なされる場合以外は試みないでください。

概要

WebCenter Sites 11gR1 には、「Contributor」という新しいインタフェースが用意されています。Dash インタフェースは、今回のリリースでは削除されました。今回のリリースでの変更点としては、このほかに、キャッシング機能の向上、Central Authentication Service (CAS) を介したシングル・サインオン (SSO)、Web エクスペリエンス管理 (WEM) フレームワークがあります。

Content Server 7.6 パッチ 1 および 2 ではオプションだった機能のうち、必須となったものは次のとおりです。

- **Central Authentication Service (CAS)**。CAS Web アプリケーションは、アップグレード・プロセス中にデプロイされます。WebCenter Sites 11gR1 はまた、ユーザー認証の負荷を均衡化させるために、CAS クラスタリングをサポートしています。CAS クラスタのセカンダリ・メンバーは、アップグレード後のプロセスで手動でデプロイされます。
- **inCache フレームワーク**。Apache ライセンスの下で使用可能な Terracotta の Ehcache オープン・ソース製品を実装したものです。inCache フレームワークでは、従来のページ・キャッシング方式と比較して、パフォーマンスが大幅に向上しています。また、アセット・キャッシング機能と resultset キャッシング機能も用意されています。inCache フレームワークは、デフォルトでインストールおよび構成されます。inCache のページ、アセットおよび resultset のキャッシングの情報は、*WebCenter Sites 管理者ガイド*に記載されています。resultset のキャッシングの追加情報は、*WebCenter Sites 開発者ガイド*に記載されています。
- **WEM フレームワーク**。WEM ログイン・ページは、新しい WebCenter Sites ログイン・ページになり、WebCenter Sites のインタフェースにアクセスするやり方が影響を受けます。WEM フレームワークは、次のコンポーネントで構成されています。
 - **REST API**: WEM フレームワークにアプリケーションを構築および実装することを目的として、開発者が WebCenter Sites と連絡できるようにします。

- **Universal UI Container:** WebCenter Sites に統合されているカスタム構築アプリケーションに、WEM フレームワークを介してアクセスするための単一インタフェースを提供します。また、アプリケーションのインタフェースのレンダリングも可能になります。
- **WEM の Admin インタフェース:** WEM フレームワークで稼働するアプリケーションに対してユーザーを結合できるようにし、ユーザーの一元管理に提供します。WEM の Admin インタフェースは、配信システムにはインストールされていません。
- **REST セキュリティ・モデル:** WEM フレームワークで稼働するアプリケーションのリソースに対するアクセス権限を管理者が制御できるようにします。
- **シングル・サインオン:** WebCenter Sites のユーザーが、アクセス権限を付与されているアプリケーションすべてに、各アプリケーションにサインインしないでセッション中にアクセスできるようにします。
- WEM には Central Authentication Service (CAS) が必要であり、これはクラスタ化していてもしていなくてもかまいません。
- 開発システムとコンテンツ管理システムには、InDis Baltic 社の Clarkii Online Image Editor (OIE) がデフォルトでインストールされています。Clarkii は、現在の Online Image Editor のかわりに、WebCenter Sites の Contributor インタフェースで有効化できます。ただし、Contributor インタフェースの Web モードではサポートされていません。Clarkii OIE の詳細は、*WebCenter Sites 開発者ガイド*と *WebCenter Sites ユーザーズ・ガイド*を参照してください。
- **アップグレードのオプション**
 - Apache Log4j ロギング・システム。これは、Admin インタフェースからロガーの動的更新ができるようにします。
 - Oracle Access Manager (OAM) との統合。これについては、*WebCenter Sites: サポートするソフトウェアの構成*に記述されています。

アップグレード前の手順

このガイドは、Content Server の経験があるインストール・エンジニアを対象としています。WebCenter Sites 11gR1 にアップグレードする前に、次の手順を実行します。

- WebCenter Sites リリース・ノートと *Oracle WebCenter Sites 動作保証マトリックス*に目を通してください。
- このガイドの他の部分を読んで、アップグレード手順、インストーラによって行われる変更、およびアップグレード後の手順に習熟してください。
- Content Server 7.6 パッチ 1 またはパッチ 2 のインストールから始めます。使用環境内でのすべての Content Server システムで、WebCenter Sites 11gR1 インストーラを実行します。システム・タイプには、開発、コンテンツ管理、配信の 3 つがあります。開発システムとコンテンツ管理システムは同じタイプであることが普通ですが、用途は異なります。

WebCenter Sites 11gR1 インストーラは、Content Server 7.6 パッチ 1 またはパッチ 2 のインストール・プロセスで選択されたシステム・タイプ (開発、

コンテンツ管理または配信)とデプロイメント・モード(自動または手動)の両方を検出して再利用します。たとえば、Content Server 7.6 パッチ 1 が配信システムとしてインストールされ自動的にデプロイされた場合、WebCenter Sites 11gR1 インストーラは引き続き、このシステムを配信システムとして扱い自動的にデプロイします。

- 現在のプラットフォームが WebCenter Sites 11gR1 でサポートされていない場合は、WebCenter Sites 11gR1 でサポートされているスタックに新しいインストールを実行し、この環境に対するコンテンツとカスタマイズをすべて公開してからアップグレードを続行してください。

注意

- Content Server 7.6 パッチ 1 または 2 で行ったインタフェース・カスタマイズは、WebCenter Sites 11gR1 で大きな変更を行わないと動作しません。そのため、そうした変更を新しい環境で行わないと時間が短縮されます。ただし、インタフェースのカスタマイズ以外のカスタマイズは、アップグレードに先だって新しい環境で行ってください。
 - システム・タイプとデプロイメント・モードは、アップグレード中には変更できません。
 - アップグレード中には、Content Server のプライマリ・クラスタ・メンバーをセカンダリ・クラスタ・メンバーとして再構成することも、Content Server のセカンダリ・クラスタ・メンバーをプライマリ・クラスタ・メンバーとして再構成することもできません。
- 現在の Content Server アプリケーションが最初にデプロイされた後に変更された場合は(一番可能性があります)、次の手順を実行します。
 - まず cs.war ファイル、次に ContentServer.ear ファイルを作成して、現在の Content Server アプリケーションをバックアップします。また、インストール・ディレクトリ、共有ディレクトリおよびデータベースもバックアップします。手順の詳細は、<http://docs.oracle.com> で入手できる *FatWire Content Server 7.6 パッチ 2 バックアップおよびリカバリ・ガイド*を参照してください。

アップグレード後のプロセスでは、バックアップしたインストール・ディレクトリ、共有ディレクトリおよびデータベースを使用して、それらに含まれている可能性のあるカスタマイズを再適用します。
 - 現在の Content Server アプリケーションが最初にデプロイされた後、そのアプリケーションにカスタマイズが行われ、かつ <cs_install_dir>/ominstallinfo/ app にある cs.war ファイルにも ContentServer.ear ファイルにもそれらのカスタマイズが反映されていない場合は、それらのファイルを削除し、現在デプロイされている Content Server アプリケーションの .war ファイルと .ear ファイルを同じディレクトリにコピーします。

アップグレード中、cs.war ファイルは cs-*date-time*.war としてアーカイブされます。アップグレード後のプロセスでは、必要に応じて、アーカイブされたバージョンを使用してカスタマイズを再適用します。

- JDK 1.6 の場合は、インストーラの Sun/lib ディレクトリにある jaxb-impl-2.1.12.jar ファイルを、<PATH_TO_JDK_FOLDER>/jre/lib/endorsed にコピーします。

注意

JDK 1.6 または JDK 1.7 に付属している jaxb-impl は使用しないでください。WebCenter Sites 11gR1 は、jaxb-impl の最新バージョンに依存しており、これは、前述で指定された場所に提供されています。(JDK 1.6 に付属している) WebLogic Server との実行時の競合を解決するために、最新の JAR ファイルを使用する必要があります。

- すべてのアプリケーション・サーバーで、次の手順を実行します。
 - 起動スクリプトを更新します。
 - max PermGen パラメータを、128MB-196MB の範囲で設定します。たとえば、-XX:MaxPermSize=196m とします。
 - -Dcs.useJavaURLDecoder を false に設定します。これで、URL 文字のデコードに Apache URLCodec が使用されるようになります。
 - CLASSPATH 環境変数を更新します。
 - <cs_install_dir>/bin を追加します。
 - WebCenter Sites で変更された Microsoft XML Parser のバージョン (WEB-INF/lib ディレクトリにある MSXML.jar) へのパスを追加します。クラス・パスが Microsoft XML Parser の別バージョンを指している場合は、XML の解析時に WebCenter Sites が失敗します。
- サポートされているすべてのオペレーティング・システムでは、<cs_install_dir>/bin を追加することによって、次のライブラリ・パス環境変数を更新します。
 - Linux と Solaris: LD_LIBRARY_PATH
 - AIX: LDPATH
 - Windows: PATH

注意

クラス・パスとライブラリ・パスが適切に設定されていない場合は、WebCenter Sites の Admin インタフェースの「管理」タブにあるシステム・ツールが、機能が制限された状態で使用可能となり、CAS の起動が失敗するため、システムにアクセスできなくなります。

- ContentServer ユーザーのパスワードに特殊文字が含まれている場合は、特殊文字を含まないパスワードに置換します。アップグレード・インストーラでは、パスワードのフィールドに特殊文字が受け入れられません。すべてのアップグレードが完了したら、元のパスワードを使用できます。
 - Tomcat アプリケーション・サーバーではオプションです。データソースに関連するエラーが発生した場合は、<app_server_home>/lib ディレク

トりに次の JAR ファイルを配置して、WebCenter Sites が初めて起動されたときにデータソースが正しく初期化されるようにします。

- commons-dbc-1.3.jar このファイルは、<http://commons.apache.org/dbcp/> からダウンロードできます。
- commons-pool-1.5.5.jar このファイルは、<http://commons.apache.org/pool/> からダウンロードできます。
- 現在の Content Server 7.6 パッチ 1 またはパッチ 2 システムに WEM フレームワークがインストールされている場合は、次の手順を実行します。
 - アップグレードを開始する前に、以前の CAS インストールをアンデプロイして削除します。
 - 表 1 に示されているように、CASHostName フィールドと CASPortNumber フィールドの値を設定することによって、Content Server のすべてのクラスタ・メンバーの omii.ini ファイルを変更します。

注意

グラフィカル・インストーラを使用している場合は、この手順は実行しないでください。インストーラを停止し、omii.ini ファイルを変更してインストーラを再実行する必要があるからです。

表 1: CAS に関連するフィールド

フィールド	CAS	値
CASHostName	クラスタ化されている	ロード・バランサを実行しているサーバーのホスト名。
	クラスタ化されていない	CAS のデプロイ先となるサーバーのホスト名。
CASPortNumber	クラスタ化されている	ロード・バランサを実行しているサーバーのポート番号。
	クラスタ化されていない	CAS のデプロイ先となるサーバーのポート番号。
CASHostNameLocal	クラスタ化されている	内部でアクセス可能な CAS ロード・バランサを実行しているサーバーのホスト名。
	クラスタ化されていない	CAS のデプロイ先となるサーバーの内部ホスト名。
CASPortNumberLocal	クラスタ化されている	内部でアクセス可能な CAS ロード・バランサを実行しているサーバーのポート番号。
	クラスタ化されていない	CAS のデプロイ先となるサーバーの内部ポート番号。
CASHostNameActual	クラスタ化されている / クラスタ化されていない	CAS のデプロイ先であるサーバーのホスト名 / IP アドレス。

アップグレード前の意思決定

- GUI インストーラとサイレント・インストーラのどちらを実行するか

GUI インストーラを使用すると、広範囲にわたるオンライン・ヘルプにアクセスでき、アップグレード・プロセスを順序だてて理解できます。サイレント・インストーラを使用すると、`omii.ini` ファイルで提供されている構成情報に基づいて自動化されるアップグレードを実行できます。これらのアップグレード手順の詳細は、「[対話型アップグレード](#)」と「[サイレント・アップグレード](#)」を参照してください。

- SSO の場合は、アップグレード・プロセス中に、**Central Authentication Service (CAS)** のデプロイメント情報も提供する必要があります。必要となるデプロイメント情報は、デプロイメント・モード (自動か手動か) によって異なります。このデプロイメント・モードは、**Content Server 7.6** パッチ 1 または 2 のインストールから継承されます。
- 次の各ポイントは、CAS のデプロイメント情報を判断する際に役立ちます。
 - **Content Server 7.6** パッチ 1 またはパッチ 2 で有効だったデプロイメント・モードはどれでしょうか。また、そのモードによってデプロイメント・プロセスはどのように決定されるでしょうか。

Content Server のプライマリ・クラスタ・メンバーでインストーラを実行している場合は、WEM フレームワークをインストールまたはアップグレードするためのデプロイメント・シナリオは次のとおりです。

- 自動デプロイメントが有効の場合は、インストーラによって、**WebCenter Sites** と **CAS** が同じサーバーにデプロイされます。
- デプロイメント・モードが手動であるとインストーラが検出した場合は、**WebCenter Sites** を手動でデプロイします。**CAS** もデプロイします。ただし、インストーラと異なり、**CAS** は、**WebCenter Sites** インストールのプライマリ・クラスタ・メンバーと同じサーバーにも、別のサーバーにもデプロイできます。

自動デプロイメント・プロセスが完了したら、**CAS** を別のサーバーに移動する場合は、アップグレード後のプロセスの一環として **CAS** を手動で再デプロイする必要があります。

- **CAS** のセカンダリ・クラスタ・メンバーはどのようにデプロイしますか。

インストール・プロセス中に、**CAS** のデプロイメント情報を **CAS** のセカンダリ・クラスタ・メンバーに提供します。*Oracle WebCenter Sites: サポートするソフトウェアの構成の CAS のセカンダリ・クラスタ・メンバーのデプロイに関する項に記述されているように、これらのメンバーを手動で構成およびデプロイします。*
- アップグレード・プロセス中に、**Content Server** の既存のロギング・システムから **Apache log4j** に移行しますか (推奨)。

その場合は、更新プロセス中に、**log4j** 移行オプションを選択する必要があります。アップグレード・プロセス中に **log4j** に移行しない場合は、後で **log4j** に切り替えることができます。手順については、**WebCenter Sites 11gR1 管理者ガイド**を参照してください。

アップグレード後のサマリー

WebCenter Sites 11gR1 がインストールされたら、カスタマイズを再適用してインストールを検証します。最後に、リモートの Satellite Server を再インストールします。

アップグレード・プロセス中の変更

Content Server 7.6 パッチ 1 またはパッチ 2 から WebCenter Sites 11gR1 にアップグレードすると、既存のインストールに数多くの変更が行われます。この一覧に不明な点がありましたら、アップグレードを開始する前にテクニカル・サポートにご連絡ください。

この項で説明する変更は次のとおりです。

- インタフェースとアプリケーション
- ロール
- Page アセット・タイプと Slot アセット・タイプ
- 属性エディタ
- プロパティの値
- WEM の変更 (WEM 以外のインストールからアップグレードする場合)
- Log4j
- キャッシング・フレームワーク (以前に切り替えられていない場合)
- カスタム要素
- リビジョン追跡
- データベース表
- Web アプリケーションの変更
- その他の変更

インタフェースとアプリケーション

- Content Server の「Advanced」インタフェースは、WebCenter Sites の「Admin」インタフェースに名前が変更されました。デフォルトでは、編集用のインタフェースとして使用されることはなくなりました。ただし、`futuretense_xcel.ini` ファイルでプロパティ `advancedUI.enableAssetForms` を設定することによって、アップグレード後のプロセスで編集機能を再度有効化できます。詳細は、*WebCenter Sites* プロパティ・ファイル・リファレンスを参照してください。
- Advanced インタフェースに対して行ったカスタマイズは、アップグレード後には動作しなくなります。そのため、データベース、インストール・フォルダ、共有フォルダおよび Web アプリケーションに格納されているカスタマイズをすべてバックアップします。WebCenter Sites へのアップグレードが完了したら、このバックアップに格納されているカスタマイズを再適用します。

- WebCenter Sites の Contributor インタフェースがインストールされています。インタフェースについては、*Oracle WebCenter Sites ユーザーズ・ガイド* を参照してください。
- Content Server の Dash インタフェースが完全に削除されています。パッチ 1 または 2 で Dash インタフェースに対して行われたカスタマイズは、アップグレード時には適用されません。
- WEM の管理アプリケーションが AdminSite 以外のサイトに割り当てられていた場合は、割当てが解除されます。アップグレード・プロセス後に、AdminSite 以外のサイトに手動で再割当てを行う必要があります。
- アップグレード元のインストールに WEM フレームワークが含まれていない場合は、AdminSite がインストールされ、全体の管理者に割り当てられません。

ロール

- AdvancedUser ロールは、WebCenter Sites の Admin インタフェースへのアクセス権限を全体の管理者に提供するために保持されています。そのため、AdvancedUser ロールが割り当てられているユーザーは、デフォルトでコントリビューター向けに構成されなくなったインタフェースへのアクセス権限を保持します。このロールは、管理ユーザーでないユーザーから削除するのが賢明と考えられます。
- SitesUser ロールは、WebCenter Sites の Contributor インタフェースへのアクセス権限をユーザーに提供するための新規のロールです。
- アップグレード元のインストールに WEM フレームワークが含まれていない場合は、SiteAdmin ロールが全体の管理者に割り当てられます。全体の管理者は、RestAdmin グループにも追加されます。
- DashUser ロールが削除されます。

Page アセット・タイプと Slot アセット・タイプ

- Page アセット・タイプは、拡張可能な Page アセット・タイプにアップグレードされ、(フレックス・アセットの場合に類似した) 属性、フィルタおよび定義を含むようになります。サブタイプが定義に変更されます。また、外部テンプレートが、テンプレートの新規「レイアウト」タイプに変更されます。詳細は、*WebCenter Sites 開発者ガイド* を参照してください。

注意

アップグレード後、ユーザーは、PageAttribute、PageDefinition および PageFilter のスタート・メニューを作成して、これらのアセット・タイプを使用できるようにする必要があります。

属性エディタ

- RememberME は非推奨で、サポートされていません。

- 値が複数ある属性がすべて、並替えに対応するようになりました。
- デフォルトですべてのインタフェースにおいて、CKEditor がリリース 3.6.2 に更新され、遅延ロードされます。開発者は、フレックス・アセットとベシック・アセットの両方に対して、以前から存在していた FCKEditor の属性定義を引き続き使用できます。

注意

アップグレード後には、属性エディタを FCKEditor から CKEditor に変更することをお勧めします。

- 次の属性エディタが新規にインストールされます。
 - TypeAhead
 - Uploader
- URL 属性が BLOB に置換されます。ただし、既存の URL 属性は、アップグレード後も引き続き機能します。

プロパティの値

- デフォルト・プロパティのカスタム値はアップグレード・インストーラによってオーバーライドされ、デフォルト値に置換されます。
- カスタム・プロパティのカスタム値は、アップグレード・プロセスの影響を受けません。

WEM の変更 (WEM 以外のインストールからアップグレードする場合)

- CAS シングル・サインオン Web アプリケーションがデプロイされます。
- CAS は、WebCenter Sites と並行して、または別のクラスタにインストールできます。
- REST サブレットがインストールされます。REST セキュリティは、*WebCenter Sites: WEM フレームワーク開発者ガイド*と *WebCenter Sites: WEM フレームワーク管理者ガイド*に記述されています。
- 新規サイト : AdminSite
- 新規セキュリティ・モデル
- CAS で保護される URL
- WebCenter Sites の新規ログイン・ページ

Log4j

Apache Log4j に切り替えることができ、切り替えると、WebCenter Sites の Admin インタフェース内から、そのシステム・ツールを使用することによって、ログ・レベルを変更できます。

キャッシング・フレームワーク (以前に切り替えられていない場合)

- Content Server で以前使用されていた既存のデータベース・ベースおよびハッシュ・ベースのキャッシュは、Ehcache by Terracotta に基づくメモリー・ベースのキャッシュである inCache に置換されます。詳細は、*WebCenter Sites 開発者ガイド*を参照してください。
- キャッシュを切り替えた結果、キャッシングの再調整が必要になります。
- inCache はメモリー・ベースなので、追加の JVM ヒープが必要になります。

カスタム要素

- AssetStubElementCatalog 表にある要素はすべて、WebCenter Sites 11gR1 のネイティブ・コードで上書きされます。
- Advanced インタフェースにあるカスタマイズされた要素は、WebCenter Sites 11gR1 のネイティブ・コードで上書きされます。

リビジョン追跡

リビジョン追跡が変更されます。詳細は、*WebCenter Sites ユーザーズ・ガイド*を参照してください。

データベース表

アップグレード・プロセス中に、データベース表に対して次の変更が行われる場合があります。次に示すような、データベース・タイプに適用される変更のみが実行されます。

- DB2 と MSSQL では、StringValue が 255 から 2000 に変更されます。
- assettypes の _TGROUPEL 表に序数列が追加されます。
- ApprovalQueue 表に cs_target 列が追加されます。
- ApprovedAssets 表に changedby 列が追加されます。
- AssetPublistList 表に tag 列が追加されます。
- アップグレードを実行すると、txRTInfo 表が作成されます。この表は、アップグレード後に手動で削除します。
- IndexSourceMetaDataConfig 表で、fielddescriptor 列が urlfielddescriptor に変更され、その値が 2048 から 255 に変更されます。
- Mimetype 表で、mimetype 列のサイズが 64 から 128 に変更されます。
- FW_PUBDATASTORE 表で、エンコーディング列のサイズが 255 に変更されます。
- SAVESEARCH 表で、spostpage 列が NULL 値可能に変更されます。
- すべての属性アセット・タイプに Editorid 列が追加されます。
- Flex Definition アセット・タイプと親定義アセットが変更されて、Mungo 表に cs_ordinal 列が含まれるようになります。
- Page 表で、flextemplateid 列が追加されます。
- Publication 表で、cs_wrapperasset 列が追加されます。

- PubTarget 表で、FACTORS 列のサイズが 1000 に変更されます。
- RTInfo 表で、newerformat 列が追加されます。
- Template_Subtypes 表で、サブタイプ列のサイズが 64 に変更されます。
- Workflow_DlgA、WorkflowGroups_Dlock、WorkflowRoutes_Comp、WorkflowRoutes_Cond、WorkflowRoutes_Dead の各表で、Ordinal 列が追加されます。
- AssetEditPane が削除されます。ただし、アップグレードも残っていたら、手動で削除する必要があります。
- 追加される表は次のとおりです。

FW_AttributeEditor	Page_Attribute_Publish
FW_AttributeEditor_Config	Page_Attribute_Subtypes
FW_AttributeEditor_ConfigVals	Page_Definition
FW_AttributeEditor_Dim	Page_Definition_Dim
FW_AttributeEditor_DimP	Page_Definition_DimP
FW_AttributeEditor_Publish	Page_Definition_Publish
FW_Tag	Page_Definition_TAttr
FW_UIConfiguration	Page_Definition_TFilter
FW_UIConfiguration_Dim	Page_Filter
FW_UIConfiguration_DimP	Page_Filter_Args
Page_Mungo	Page_Filter_Dim
Page_Publish	Page_Filter_Dim_P
Page_Ratings	Page_Filter_Publish
Page_Attribute	PublishedTags
Page_AttributeArgs	Slots
Page_Attribute_Dim	Slots_Dim
Page_Attribute_DimP	Slots_DimP
Page_Attribute_Extension	Slots_Publish

- フレックスまたは親のアセット・タイプすべてに _Ratings 表が追加されます。定義または属性のアセット・タイプは無視されます。これは、一部の特殊なコアのアセット・タイプを除き、ベーシック・アセット・タイプにも追加されます。
- すべての属性アセット・タイプに _SubTypes 表が追加されます。たとえば、MEDIA_A_SUBTYPES です。

Web アプリケーションの変更

WebCenter Sites の Web アプリケーションから Dash インタフェースが削除されるため、次の変更が発生します。

- web.xml ファイル:

- csRedirect、trinidad、adfFaces の各フィルタが削除されます。
- org.apache.myfaces.webapp.StartupServletContextListener リスナーが削除されます。
- FacesServlet と faces サブレットが削除されます。
- WEB-INF フォルダで、web.xml、weblogic.xml (該当する場合)、applicationContext.xml 以外のファイルがすべて削除されます。

タグに対する変更

- WebCenter Sites 11gR1 で新たに追加されたタグは次のとおりです。
 - insite:list
 - insite:slotlist
 - insite:ifedit
 - WebCenter Sites 11gR1 で非推奨になったタグは次のとおりです。
 - insite:addrref は、insite:list タグに置換されています
 - insite:addvalue は、insite:list タグに置換されています
 - insite:beginlist は、使用方法に応じて、insite:list タグまたは insite:slotlist タグに置換されています
 - insite:endlist は、使用方法に応じて、insite:list タグまたは insite:slotlist タグに置換されています
 - insite:beginref は、field/assetid/assettype バリエーションを使用する insite:calltemplate に置換されています
 - insite:endref は、insite:calltemplate タグに置換されています
- タグの詳細は、*WebCenter Sites タグ・リファレンス*を参照してください。

その他の変更

- デフォルトでは、WebCenter Sites は、日付とタイムスタンプにミリ秒を使用します。
- アップグレード後、属性フォームには、新規インストールでは見られない追加のフィールドが表示されます。この属性フォームと新規インストール時の属性フォームの間で整合性を維持するには、mwb.externalattributes プロパティを false に設定することによって、gator.ini ファイルを編集します (gator.ini ファイルは、<cs_install_folder> の中にあります)。
- アップグレード前に CSDT を使用していた場合は、CSDT プラグインが、WebCenter Sites 11gR1 インストーラに付属している最新バージョンに更新されていることを確認してください。
- アップグレード前にアセット・タイプのいずれかに対してアセット・タイプ検索が構成されている場合は、アップグレード後には、検索結果に日付が適切に表示されなくなります。日付を正しく表示するには、ユーザーはアセット・タイプの索引を再作成する必要があります。
- eWebEditPro がサポートされていないため、futuretense_xcel.ini ファイルで、xcelerate.ewebeditpro プロパティの値が削除されます。
- レイアウトのテンプレート:

使用タイプ「エレメントは HTML ページ全体を定義するため、外部から呼び出すことができます。」のテンプレートは、インストーラによって、使用タイプ「エレメントはレイアウトとして使用されます。」に変換されます。ただし、それ以外の使用タイプはいっさい変換されません。そのようなテンプレートを使用するアセットを編集するときには、レイアウトのテンプレートを割り当てる必要があります。

- 承認されたアセットのパブリッシュが正しく機能するようになるのは、アップグレード・ユーティリティが実行されたときです。WebCenter Sites 11gR1 では、日付はミリ秒単位の精度で管理されます。前のバージョンでは、承認関連データに対してミリ秒単位の精度は提供されていなかったため、アップグレード後に日付が不正確に解釈される可能性があります。このような状況が発生しないようにするために、アップグレード・ユーティリティでは、承認の表にミリ秒単位の精度を追加して、アップグレード後も承認状態が正しく保持されるようにしています。ApprovedAssets、ApprovedAssetDeps、PublishAssets の各表のエントリは、このユーティリティの影響を受けません。

第 2 章

Content Server 7.6 パッチ 1 またはパッチ 2 の環境の WebCenter Sites へのアップグレード

この章には次の項が含まれます。

- [Content Server の環境のアップグレード](#)
- [アクティブ環境のアップグレード](#)

Content Server の環境のアップグレード

この項では、Content Server 7.6 パッチ 1 またはパッチ 2 のテスト環境を WebCenter Sites 11gR1 にアップグレードし、テスト環境を検証してアクティブ環境をアップグレードする手順について説明します。

注意

- WebCenter Sites 11gR1 では各種インタフェースが変更されているため、最初にテスト環境をアップグレードすることを強くお勧めします。アクティブ環境のみをアップグレードしようとする場合は、停止時間が長引くと予想できます。
- この項に記載されている手順を開始する前に、第1章「アップグレードの前に」に目を通し、使用環境に該当する必要手順を実行しておいてください。
- クラスタ化された環境では、まずプライマリ・クラスタ・メンバー、その後ですべてのセカンダリ・クラスタ・メンバーをアップグレードします。インストールを検証する前に、すべてのメンバーをアップグレードする必要があります。
- Content Server データベースに対する権限が、WebCenter Sites: サポートするソフトウェアの構成で推奨されているデフォルトにリセットされていることを確認します。これは、稼働中のシステムでは不要な権限がインストーラでは必要となる場合があるからです。これらの権限は、アップグレードの完了後に変更して、任意のカスタマイズを含むようにすることができます。

手順 1. Content Server 7.6 パッチ 1 またはパッチ 2 の環境のバックアップ

手順 2. テスト環境の作成

手順 3. Content Server パッチ 1 またはパッチ 2 のアップグレード準備

手順 4. Content Server 7.6 パッチ 1 またはパッチ 2 のインスタンスの WebCenter Sites 11gR1 へのアップグレード

手順 5. アップグレード・ユーティリティの再実行

手順 6. アップグレードの完了

手順 7. 環境におけるリモートの Satellite Server のアップグレード

手順 8. アップグレードされた環境の検証

手順 1. Content Server 7.6 パッチ 1 またはパッチ 2 の環境のバックアップ

この項では、予防策としてアクティブ環境をバックアップします(テスト環境を作成することにした場合は、バックアップをリカバリして行います)。

環境をバックアップするには

1. ガイドラインと具体的な手順については、Content Server 7.6 パッチ 2 バックアップおよびリカバリ・ガイドを確認してください。

2. 必要に応じて、あるシステムのコンテンツを別のシステムにパブリッシュすることによって (たとえば、開発システムから管理システム、さらに配信システムへ)、アクティブ環境内のシステムをすべて同期化します。
3. パブリッシュ・スケジュールをすべて無効にして、パブリッシュ・セッションが実行されないようにします。アクティブ環境のアップグレードと検証が済んだら、パブリッシュ・スケジュールを再度有効にします。
4. カスタマイズされたエレメントを書き留めます。
 - a. ベーシック・アセット・タイプごとに、(ElementCatalog 表内にある) エレメントの中でカスタマイズされたコードがあれば書き留めておきます。
 - b. Content Server の Advanced インタフェースのいずれかの部分をカスタマイズした場合は、アップグレード後には既存のカスタマイズが機能せず、変更する必要があるので、そのカスタム・コードを書き留めておきます。
5. すべてのイベントを無効にしておいてください。
6. 次のアセット・タイプに対して、リビジョン追跡を無効にします。
 - ASSOCNAMED 表
 - CSElement
 - Template
 - Page
7. 外部データ・フィールドがあれば無効にします。
8. このガイドに記述されているように、アクティブ環境内のシステムをすべてバックアップします。これには、Content Server のインスタンスがすべて、そしてリモートの Satellite Server がすべて含まれています。
9. Web サーバーの構成、アプリケーション・サーバーの構成、データベースの構成、LDAP の構成など、既存のインストールに関する情報をすべて書き留めておきます。これらはアップグレード中には変更されませんが、その一方で、別の場所 (たとえば、新規キャッシュの調整) で変更が必要になった場合に、既存の設定を参照できます。

手順 2. テスト環境の作成

注意

アクティブ環境をアップグレードする前に、テスト環境で試用アップグレードを完了します。テスト環境が用意されていない場合は、まず開発システムのアップグレードから始めます。このとき、停止時間が長引くと予想されます。WebCenter Sites で行われる変更の程度を考慮し、テスト環境を使用することを強くお勧めします。

1. テスト環境を作成する準備ができたら、前の手順で作成したバックアップをリカバリします。リカバリのガイドラインについては、*Content Server 7.6 パッチ 2 バックアップおよびリカバリ・ガイド*を参照してください。

2. アクティブ環境でリモートの **Satellite Server** が実行されている場合は、少なくともそのうちの 1 つを、テスト環境に忘れずに複製してください。
3. 次に示すガイドラインに従って、テスト環境を完成します。
 - Content Server
 - 1) Content Server を、前のホストで占有していたのと同じ場所にコピーします。
 - 2) SystemSatellite 表のエントリを除去します。ただし、ローカル・システムと、このアップグレードのテスト対象とする 1 つのリモート Satellite Server は除きます。
 - 3) ContentServer.ear ファイルと cs.war ファイルの中にある次の各ファイルで、ホスト名と IP アドレスを変更します。

ホスト名と IP の変更	ファイル	ディレクトリ
必須	satellite.properties (ホスト)	<deploy_dir>/WEB-INF/classes
オプション	SampleSites.html	<deploy_dir>/Xcelerate
オプション	AssetSet.wsdl AssetType.wsdl Asset.wsdl Miscellaneous.wsdl SitePlan.wsdl	<deploy_dir>/Xcelerate/wsdl

- 4) <cs_install_dir> ディレクトリの中にある次の各プロパティ・ファイルで、ホスト名と IP アドレスを変更します。

ホスト名と IP の変更	プロパティ・ファイル	プロパティ
必須	futuretense.ini databaseloader.ini futuretense_xcel.ini	cs.eventhost db1.loginurl xcelerate.batchhost

- 5) WEM フレームワークを事前にインストールしてある場合は、次のファイルも編集します。

ホスト名と IP の変更	ファイル	プロパティ
必須	<cs_home>/Bin/ jbossTicketCacheReplication Config.xml <cs_home>/bin/cas.properties <cs_home>/Bin/host.properties CS/WEB-INF/classes/ SSOConfig.xml CAS/WEB-INF/ deployerConfigContext.xml	これらはプロパティ・ファイルではありません。必要に応じて、値の検索と置換を行います。

- アプリケーション・サーバー
 - 1) 前のホストで使用されていたのと同じパスを使用して、アプリケーション・サーバーをインストールします。
 - 2) 前のホストで使用されていたのと同じ JNDI 名を使用して、データソースを作成します。リストアされたデータベースのデータベース情報を使用します。
 - 3) アプリケーション・サーバーでの必要に応じて、Content Server アプリケーションをデプロイします。

手順 3. Content Server パッチ 1 またはパッチ 2 のアップグレード準備

この項では、Content Server でサポートしているソフトウェアを最初にアップグレードするために、*Oracle WebCenter Sites 動作保証マトリックス* および現在のリリース・ノートを使用します。

A. 環境の準備

注意

クラスタ化されている環境が対象。クラスタ・メンバーをすべて停止します。プライマリ・メンバーのみをアップグレードするように準備します。プライマリ・メンバーのアップグレードが終了したら、セカンダリ・メンバーでも WebCenter Sites をアップグレードします。

1. Content Server でサポートしているコンポーネントを、*Oracle WebCenter Sites 動作保証マトリックス* に一覧表示されているバージョンにアップグレードします。サポートしているソフトウェアには次のようなものがあります。
 - オペレーティング・システム
 - アプリケーション・サーバー
 - Java SDK
 - Content Server のデータベース
 - (条件付き)。LDAP サーバーと Web サーバー。LDAP を使用していて、LDAP サーバーをアップグレード (または変更) することにした場合は、LDAP データのすべてを、古いサーバーから新しいサーバーに手動で移行します。

注意

サポートしているコンポーネントをアップグレードする手順については、該当するベンダーのドキュメントを参照してください。

2. インストールしたものが全面的に機能することを検証します。

B. Content Server のインスタンスの準備

1. Content Server が Oracle データベースとともに稼働している場合は、Oracle データベースのユーザーに次のシステム権限が割り当てられるようにします。これらの権限は、WebCenter Sites の運用に必須である次の権限と同一です。

- GRANT UNLIMITED TABLESPACE TO <USER>;
- GRANT CREATE SESSION TO <USER>;
- GRANT CREATE TABLE TO <USER>;
- GRANT CREATE VIEW TO <USER>;

2. アップグレード全体でカスタマイズを維持します。

注意

デプロイメント・ディレクトリの中にある js/src フォルダは、アップグレード中に削除されるので、この手順では、デプロイメント・カスタマイズがあればバックアップします。

カスタム JARS をデプロイしているか、または (<cs_install_dir>/ominstallinfo/app の中にある) cs.war ファイルおよび ContentServer.ear ファイルの現在のコピーに反映されていない構成変更を行っている場合は、次のようにして、現在デプロイされている Content Server アプリケーションから、クリーンな cs.war ファイルおよび ContentServer.ear ファイルを作成します。

- a. ominstall/app にある cs.war と ContentServer.ear をバックアップします。
- b. WEM フレームワークがインストールされている場合は、ominstall ディレクトリにある cas.war と cas.ear をバックアップします。
- c. デプロイされて開いている cs.war ファイルを見つけ、このファイルを圧縮して (WebCenter Sites インストーラで使用される同一バージョンの JAR を使用)、ominstall/app ディレクトリの中に配置します。
- d. バックアップされた ContentServer.ear ファイルを、その独自のディレクトリに開きます。
- e. 開いている ContentServer.ear ファイル内で、新しい cs.war で既存ファイルを上書きコピーし、このファイルを圧縮します (WebCenter Sites インストーラで使用される同一バージョンの JAR を使用)。

- f. 新たに圧縮された ContentServer.ear を、ominstall/app ディレクトリに配置します。

注意

WebCenter Sites インストーラは、<cs_install_dir>/ominstallinfo/app ディレクトリの中に Content Server アプリケーションを検出するように構成されています。cs.war ファイルと ContentServer.ear ファイルはいずれも、名前変更も移動も行ってはなりません。インストーラが Content Server アプリケーションを検出できない場合は、アップグレード・プロセスが失敗します。

3. <cs_install_dir> ディレクトリに格納されている .ini ファイルをすべてバックアップします。
4. プロパティ・エディタを使用して、次のプロパティの値を futuretense.ini ファイルに記録します。
 - secure.CatalogManager (「基本」タブ)
 - ft.sync (「クラスタ」タブ)このインストーラでは、アップグレード・プロセス中にそれらの値が変更されます。それらはリストアする必要があります。
5. システムが読取り専用の LDAP と統合されている場合は、(WebCenter Sites で導入された) 次のロールをユーザーごとに追加します。
 - AdvancedUser: WebCenter Sites の Admin インタフェース (旧名は、Content Server の Advanced 管理インタフェース) にユーザーがログインできるようにします。

注意

AdvancedUser ロールは、WEM フレームワークを前もってインストールしてあれば、すでに存在しています。

- DashUser: Content Server の Dash インタフェースにユーザーがログインできるようにします。このインタフェースは削除されているため、このロールも LDAP から削除する必要があります。

注意

次のものについて、DashUser ロールは SitesUser ロールに置換されています。

- スタート・メニュー
- ツリー・タブ
- ワークフロー・ステップ、プロセス、割当ておよび機能権限
- 保存済検索
- パブリッシュの宛先
- アクセス権限

- SitesUser: WebCenter Sites の Contributor インタフェースにユーザーがログインできるようにします。LDAP では、SitesUser ロールを手動で作成し、それまで DashUser ロールが割り当てられていたユーザーすべてに割り当てる必要があります。

6. 現在 inCache を使用していて、ディスクの永続性によってディスクからキャッシュが削除された場合は、これらのファイルは java temp ディレクトリ内に格納されます。削除対象と考えられるディレクトリ名は、cascache、cscache、linkedcache および sscache です。
7. この手順では、WebCenter Sites アップグレード・ユーティリティを実行して、データベース・スキーマのアップグレード前ステータスをログに記録します。アップグレード後にこのユーティリティを再度実行して、インストーラによってデータベース・スキーマ (および索引) にどのような変更が加えられたのかを判別します。アップグレード・ユーティリティの詳細は、Misc¥UpgradeUtility¥upgrade-util.zip に格納されている ReadMe ファイルを参照してください。

アップグレード・ユーティリティを実行する手順は次のとおりです。

- a. 抽出されたインストーラの Misc ディレクトリに移動し、アップグレード・ユーティリティを抽出して、次のスクリプトのいずれかを開きます。
Windows: `cssystem.bat`
Unix/Linux: `./cssystem.sh`
- b. システムに関する次の情報を入力して、選択したスクリプトを編集します。
 - ドライバ、URL、ユーザー名 / パスワードという、Content Server のデータベース設定を指定します。
 - データベース・タイプに該当する DATABASEJAR ファイルを指定します。
 - `-i` パラメータを使用して、Content Server の予想デプロイメント・パスを指定します。

- c. 選択したスクリプトを実行します。

スクリプトを実行すると、(¥Misc¥UpgradeUtility ディレクトリの中に) Systeminfo.log ファイルが生成されます。このファイルには、WebCenter Sites の予想データベース・スキーマ・アップデートが格納されます。

注意

アップグレード・ユーティリティの使用の詳細は、¥Misc¥upgrade-util.zip ディレクトリに格納されている ReadMe.txt ファイルを参照してください。

8. アプリケーション・サーバーから、古い Content Server アプリケーションをアンデプロイします。アンデプロイの手順については、プラットフォームおよび Content Server のバージョンに応じたインストレーション・ガイドを参照してください。

手順 4. Content Server 7.6 パッチ 1 またはパッチ 2 のインスタンスの WebCenter Sites 11gR1 へのアップグレード

この項では、Content Server 7.6 パッチ 1 またはパッチ 2 の各インスタンスに対して、やりやすい方法 (GUI またはサイレント) で WebCenter Sites アップグレード・インストーラを実行します。

注意

- 「手順 3. Content Server パッチ 1 またはパッチ 2 のアップグレード準備」で大きな変更を行った場合は、環境をバックアップします。それを行わないと、アップグレードが失敗した場合に、「手順 1. Content Server 7.6 パッチ 1 またはパッチ 2 の環境のバックアップ」に記述されているアップグレード手順を再度実行することが必要になります。
- Content Server 7.6 パッチ 1 またはパッチ 2 では：
 - log4j が有効化されていた場合は、アップグレードでは log4j が引き続き使用されます。かわりに commons-logging.properties が使用されていた場合は、アップグレード・プロセス中に、それを log4j に変更する機会があります。
 - 従来のページ・キャッシング・フレームワークが有効化されていた場合は、そのキャッシュはフラッシュされ、inCache フレームワークに置換されます。

Content Server のインスタンスを WebCenter Sites にアップグレードするには、次のいずれかを行います。

- 対話型アップグレード
- サイレント・アップグレード

対話型アップグレード

1. WebCenter Sites インストーラ・アーカイブを一時ディレクトリに抽出し、次のインストーラ・スクリプトを実行します。
 - a. Windows の場合: `csinstall.bat`
 - b. Unix の場合: `./csInstall.sh`

注意

アップグレード・プロセス中およびアップグレード後のテストの際に、インストーラのログ (`install_log.log`)、WebCenter Sites アプリケーションのログ (デフォルトでは `futuretense.txt`)、およびアプリケーション・サーバーのログを監視します。エラーが発生した場合は、ログを確認することによってその原因を追跡できます。

アップグレードを進めていく際には、次の点を念頭に置いてください。

- インストーラのほとんどのフィールドには、元々インストールされているものからインストーラが検出した値があらかじめ埋め込まれます。あらかじめ埋め込まれている値をチェックします。古くなっている場合は、現在値を埋め込むことが必要になります (パスワードなどの情報は、手動で入力する必要があります)。
 - 値を変更する権限のないフィールドは無効 (グレー表示) になります。
 - インストーラでは、各画面にオンライン・ヘルプが用意されており、その画面で使用可能なオプションの詳細な説明が表示されます。アップグレード・プロセス中に問題が発生した場合は、オンライン・ヘルプを参照して、考えられる原因と解決策を検討してください。
2. プラットフォームに応じたインストレーション・ガイドを参照し、そこに記述されている WebCenter Sites のインストール手順に従ってください。インストールが完了したら、このガイドに戻って作業を続行してください。

注意

この時点では、Content Server のアップグレードが完了し、WebCenter Sites の有効なインストールを実行していると想定されます。アップグレードが失敗した場合は、前の手順とログを見なおし、問題があれば解決します。インストーラは、前の手順をリストアしてプロセスをもう一度開始することが必要になるため、終了しないでください。

3. WebCenter Sites アプリケーションと CAS アプリケーションを再起動します。
4. 「手順 5. アップグレード・ユーティリティの再実行」に進みます。

サイレント・アップグレード

1. <cs_install_dir>/ominstallinfo 中にある omii.ini ファイルを、<cs_install_dir> の外のフォルダにコピーし、必要に応じてコピーの名前を変更します。サイレント・インストーラでは、アップグレード・プロセス中にこのコピーが使用されます。
2. ContentServer ユーザーまたは SatelliteServer ユーザーのデフォルトのユーザー名またはパスワード、あるいはその両方が Content Server 7.5 より後で変更された場合は、名前変更された omii.ini ファイルにある次のプロパティを更新します。サイレント・インストーラは、これらの資格情報を参照することによって認証します。それらが古くなっていると、インストーラが失敗します。

プロパティ	説明
CSInstallAccountName	ContentServer ユーザーの現行ユーザー名を指定します。 デフォルト値は ContentServer です。
CSInstallAccountPassword	ContentServer ユーザーの暗号化パスワードを指定します。
SSUserPassword	SatelliteServer ユーザーの暗号化パスワードを指定します。

注意

Content Server のプロパティ・エディタを使用して、暗号化パスワードを取得します。

1. プロパティ・エディタで futuretense.ini を開きます。
 2. cs.mirrorpassword プロパティを検索します。移入されている場合は、その値をテキスト・ファイルに一時的に格納します。
 3. cs.mirrorpassword プロパティの値を、暗号化するパスワードに置換します。
 4. プロパティ・ファイルを保存して、パスワードを暗号化します。
 5. 暗号化パスワードを omii.ini ファイルにコピーします。
 6. cs.mirrorpassword の値が移入されていた場合はリストアします。
3. Content Server の既存のログイン・システムから Apache log4j に移行する場合は (推奨)、名前変更した omii.ini ファイルに次のプロパティを追加します。log4j がすでに手動で構成されていた場合、または既存のログイン・システムを保持する場合は、このプロパティを追加しないでください。

プロパティ	説明
ConvertToLog4J	log4j に移行する場合は、このプロパティを true に設定します。

注意

インストール・プロセス中に log4j に移行しない場合は、後で log4j に切り替えることができます。手順については、*WebCenter Sites 管理者ガイド*を参照してください。

4. この手順は、WEM フレームワークと FatWire Content Server Developer Tools (CSDT) に当てはまります。表 2 にあるプロパティを、名前変更された omii.ini ファイルに追加します。指定する情報は、Content Server のプライマリ・クラスタ・メンバーとセカンダリ・クラスタ・メンバーのどちらかで WEM フレームワークをインストールするかアップグレードするか、また CAS をクラスタ化するかどうかに応じて異なります。

表 2: WEM のプロパティ

プロパティ	説明
WEM	WEM フレームワークをインストールまたはアップグレードする場合は、このプロパティを true に設定します。
IsPrimaryClusterMember	Content Server のプライマリ・クラスタ・メンバーで WEM フレームワークをアップグレードまたはインストールする場合は、このプロパティを true に設定します。それ以外の場合は、false に設定します。
CASHostNameLocal	クラスタ化されている CAS: 内部でアクセス可能な CAS ロード・バランサを実行しているサーバーのホスト名。 クラスタ化されていない CAS: CAS のデプロイ先となるサーバーの内部ホスト名。
CASPortNumberLocal	クラスタ化されている CAS: 内部でアクセス可能な CAS ロード・バランサを実行しているサーバーのポート番号。 クラスタ化されていない CAS: CAS のデプロイ先となるサーバーの内部ポート番号。
CASHostName	クラスタ化されている CAS: CAS ロード・バランサを実行しているサーバーのホスト名。 クラスタ化されていない CAS: CAS のデプロイ先となるサーバーのホスト名。
CASPortNumber	クラスタ化されている CAS: CAS ロード・バランサを実行しているサーバーのポート番号。 クラスタ化されていない CAS: CAS のデプロイ先となるサーバーのポート番号。
CASHostNameActual	クラスタ化されている CAS/ クラスタ化されていない CAS: CAS が実際にデプロイされているサーバーのホスト名。

注意

Content Server のプライマリ・クラスタ・メンバーをアップグレードする際には、次の点に注意してください。

- 自動デプロイメントが有効の場合は、インストーラによって、WebCenter Sites と CAS が同じサーバーにデプロイされます。
Oracle WebCenter Sites: サポートするソフトウェアの構成の新しいサーバーへの CAS の再デプロイに関する項に記述されているように、インストール・プロセスが完了したら、別のサーバーに CAS を手動で再デプロイできます。
- 手動デプロイメントが有効な場合は、WebCenter Sites をデプロイします。CAS は、同じサーバーにも別のサーバーにもデプロイします。
Oracle WebCenter Sites: サポートするソフトウェアの構成の新しいサーバーへの CAS の再デプロイに関する項に記述されているように、インストール・プロセスが完了し、別のサーバーに CAS を再デプロイすることにしたなら、その操作を手動で行うことができます。

WEM フレームワークをインストールする際のサンプル構成

- a. Content Server のプライマリ・クラスタ・メンバーで WEM フレームワークをインストールする場合は、参考資料として次のサンプル構成を使用します。インストール・プロセスを続行するには、手順 5 に進んでください。

- CAS がクラスタ化されている場合：

```
WEM=true
IsPrimaryClusterMember=true
CASHostName=< ロード・バランサを使用しているサーバーのホスト名 >
CASPortNumber=< ロード・バランサを使用しているサーバーのポート番号 >
CASHostNameLocal=< 内部でアクセス可能で、ロード・バランサを使用しているサーバーのホスト名 >
CASPortNumberLocal=< 内部でアクセス可能で、ロード・バランサを使用しているサーバーのポート番号 >
CASHostNameActual=< CAS が実際にデプロイされているサーバーのホスト名 >
```

- CAS がクラスタ化されていない場合：

```
WEM=true
IsPrimaryClusterMember=true
CASHostName=< CAS のデプロイ先となるサーバーのホスト名 >
CASPortNumber=< CAS のデプロイ先となるサーバーのポート番号 >
CASHostNameLocal=< CAS のデプロイ先となるサーバーの内部ホスト名 >
CASPortNumberLocal=< CAS のデプロイ先となるサーバーの内部ポート番号 >
CASHostNameActual=< CAS が実際にデプロイされているサーバーのホスト名 >
```

- b. Content Server のセカンダリ・クラスタ・メンバーで WEM フレームワークをインストールする場合は、参考資料として次のサンプル構成を使用します。インストール・プロセスを続行するには、手順 5 に進んでください。
- CAS がクラスタ化されている場合：
WEM=true
IsPrimaryClusterMember=false
CASHostName=<CAS のデプロイ先となっているサーバーのホスト名 >
CASPortNumber=<CAS のデプロイ先となっているサーバーのポート番号 >
CASHostNameLocal=<CAS のデプロイ先となるサーバーの内部ホスト名 >
CASPortNumberLocal=<CAS のデプロイ先となるサーバーの内部ポート番号 >
CASHostNameActual=<CAS が実際にデプロイされているサーバーのホスト名 >
 - CAS がクラスタ化されていない場合：
WEM=true
IsPrimaryClusterMember=false
CASHostName=<CAS のデプロイ先となっているサーバーのホスト名 >
CASPortNumber=<CAS のデプロイ先となっているサーバーのポート番号 >
CASHostNameLocal=<CAS のデプロイ先となるサーバーの内部ホスト名 >
CASPortNumberLocal=<CAS のデプロイ先となるサーバーの内部ポート番号 >
CASHostNameActual=<CAS が実際にデプロイされているサーバーのホスト名 >

WEM フレームワークをアップグレードする際のサンプル構成

- a. Content Server のプライマリ・クラスタ・メンバーで WEM フレームワークをアップグレードする場合は、参考資料として次のサンプル構成を使用します。インストール・プロセスを続行するには、手順 5 に進んでください。
- CAS がクラスタ化されている場合：
WEM=true
IsPrimaryClusterMember=true
CASHostName=< ロード・バランサを使用しているサーバーのホスト名 >
CASPortNumber=< ロード・バランサを使用しているサーバーのポート番号 >
CASHostNameLocal=< 内部でアクセス可能で、ロード・バランサを使用しているサーバーのホスト名 >
CASPortNumberLocal=< 内部でアクセス可能で、ロード・バランサを使用しているサーバーのポート番号 >
CASHostNameActual=<CAS が実際にデプロイされているサーバーのホスト名 >
 - CAS がクラスタ化されていない場合：
WEM=true
IsPrimaryClusterMember=true
CASHostName=<CAS のデプロイ先となるサーバーのホスト名 >
CASPortNumber=<CAS のデプロイ先となるサーバーのポート番号 >
CASHostNameLocal=<CAS のデプロイ先となるサーバーの内部ホスト名 >

CASPortNumberLocal=<CAS のデプロイ先となるサーバーの内部ポート番号 >

CASHostNameActual=<CAS が実際にデプロイされているサーバーのホスト名 >

- b. Content Server のセカンダリ・クラスタ・メンバーで WEM フレームワークをアップグレードする場合は、参考資料として次のサンプル構成を使用します。
 - CAS がクラスタ化されている場合：

```
WEM=true
IsPrimaryClusterMember=false
CASHostName=<ロード・バランサを使用しているサーバーのホスト名 >
CASPortNumber=<ロード・バランサを使用しているサーバーのポート番号 >
CASHostNameLocal=<内部でアクセス可能で、ロード・バランサを使用しているサーバーのホスト名 >
CASPortNumberLocal=<内部でアクセス可能で、ロード・バランサを使用しているサーバーのポート番号 >
CASHostNameActual=<CAS が実際にデプロイされているサーバーのホスト名 >
```
 - CAS がクラスタ化されていない場合：

```
WEM=true
IsPrimaryClusterMember=false
CASHostName=<CAS のデプロイ先となっているサーバーのホスト名 >
CASPortNumber=<CAS のデプロイ先となっているサーバーのポート番号 >
CASHostNameLocal=<CAS のデプロイ先となるサーバーの内部ホスト名 >
CASPortNumberLocal=<CAS のデプロイ先となるサーバーの内部ポート番号 >
CASHostNameActual=<CAS が実際にデプロイされているサーバーのホスト名 >
```
5. アップグレード・インストーラのファイルを解凍します。このファイルは、WebCenter Sites 11gR1 インストーラのファイルと同じです。インストーラを実行すると、アップグレードが自動的に検出されます。
6. 抽出された Rollup フォルダのルートにある install.ini ファイルを編集します。
 - a. nodisplay プロパティを true に設定します。

- b. loadfile プロパティを、<手順 1 で名前変更された omii.ini のパスと名前> に設定します。

注意

ファイル・システム・パスを正しく指定したことを検証します。たとえば、Windows の場合は次のようになります。

```
CSInstallDirectory=C¥:/csinstall
```

または

```
c¥:¥¥install
```

7. サイレント・インストーラのスクリプトを次のように実行します。
 - Windows の場合: csInstall.bat -silent
 - Unix の場合: csInstall.sh -silent
8. プラットフォームに応じたインストレーション・ガイドを参照し、そこに記述されている WebCenter Sites のインストール手順に従ってください。インストールが完了したら、このガイドに戻って作業を続行してください。

注意

この時点では、Content Server のアップグレードが完了し、WebCenter Sites の有効なインストールを実行していると想定されません。アップグレードが失敗した場合は、前の手順とログを見なおし、問題があれば解決します。インストーラは、前の手順をリストアしてプロセスをもう一度開始することが必要になるため、終了しないでください。

9. WebCenter Sites アプリケーションと CAS アプリケーションを再起動します。
10. 「手順 5. アップグレード・ユーティリティの再実行」に進みます。

手順 5. アップグレード・ユーティリティの再実行

この項では、データベース・スキーマが正しくデプロイされるようにするために、アップグレード・ユーティリティを再実行します。

1. (Content Server 7.6 パッチ 1 またはパッチ 2 のインスタンスのアップグレードを準備したときに、30 ページのステップ 7 で構成および実行した) 次のスクリプトのいずれかを再実行します。
 - Windows: cssystem.bat
 - UNIX: cssystem.sh
2. アップグレード・ユーティリティを再実行すると、更新された systeminfo.log ファイルが生成されます。このファイルを、30 ページのステップ 7 で生成された systeminfo.log ファイルと比較して、データベース・スキーマが予想どおりに変更されたことを確認します。

手順 6. アップグレードの完了

1. プロパティ・エディタを使用して、(<cs_install_dir> 中にある) `futuretense.ini` ファイルを開き、`secure.CatalogManager`、`secure.CatalogManager`、`ft.sync` の各プロパティを元の値にリセットします。
2. アプリケーション・サーバーを再起動します。
3. 全体の管理者として **WebCenter Sites** にログインできるようにします。
4. アップグレード後に、検索エンジンが自動的に再起動されます。ただし、検索エンジンを使用するには、検索索引を再作成する必要があります (この手順では、新たに追加されたサイト・プレビュー機能が全面的に有効となります。この機能では、時間に左右される任意のアセットの「メタデータ」タブに開始 / 終了日を設定する必要があります)。
5. 任意のサイトで多言語のアセットとロケールを使用する予定がある場合は、デフォルトのロケールを作成して、サイト上のすべてのアセットに割り当てます。手順については、*WebCenter Sites 開発者ガイド*を参照してください。
6. オプション。log4j にアップグレードする場合は、`commons_logging.properties` ファイルにカスタム・ロガーが前もって見つかっていれば、`log4j.properties` ファイルに追加します。次のようにして、`log4j.properties` ファイルを更新します。
 - a. `commons_logging.properties` ファイルにカスタム・ロガーが前もって見つかっていれば追加します。
 - b. `log4j.properties` ファイルに次のプロパティを追加します。

```
log4j.logger.com.fatwire.logging.cs.file=INFO
log4j.logger.com.fatwire.logging.cs.framework=INFO
log4j.logger.com.fatwire.logging.cs.realtime=INFO
log4j.logger.com.fatwire.logging.cs.sseed=INFO
log4j.logger.com.fatwire.services=INFO
log4j.logger.com.fatwire.services.dao=INFO
log4j.logger.com.fatwire.services.time=INFO
```
7. ベーシック・アセット・タイプごとに、アセット・エレメントを再登録します。
 - a. ベーシック・アセットが有効になっているコンテンツ管理サイトにログインします。
 - b. 左側のナビゲーション・ツリーで、「管理」タブを選択します。
 - c. 「アセット・タイプ」ノードを開きます。
 - d. ベーシック・アセット・タイプをダブルクリックします。
 - e. 右側のペインで、「アセット・エレメントの登録」をクリックします。
 - f. 確認画面で、「アセット・エレメントの登録」をクリックします。
 - g. ベーシック・アセット・タイプごとに、ElementCatalog 表内にある各エレメントに対して、カスタマイズを再適用します。
8. 次のアセット・タイプに対して、リビジョン追跡があらかじめ無効になっていたら有効にします。

- ASSOCNAMED 表
- CSElement
- Template
- Page

注意

ページ・アセット・タイプがアップグレードされ、部分的にフレックス・ファミリのように動作します。そのため、アップグレード後、ページ・アセット・タイプには、PageAttribute、PageDefinition および PageFilter も含まれるようになります。

9. WEM Admin を AdminSite に割り当てます。
10. Content Server インストールでカスタマイズされたプロパティ値が、アップグレード・プロセス中にデフォルト値に置換された場合は、ここでカスタム値をリストアします。
11. Content Server の Advanced インタフェースの一部をカスタマイズした場合は、カスタム・コードを再適用します。
12. アップグレード中に新しいプロパティが追加されます。それらの値を変更する場合は、*Oracle WebCenter Sites* プロパティ・ファイル・リファレンスを参照してください。そのプロパティは次のとおりです。

プロパティ	プロパティ・ファイル
advancedUI.enableAssetForms	futuretense_xcel.ini
cc.BlobServerTimeout	futuretense.ini
cc.BlobServerCacheCSz	futuretense.ini
cs.HTTP_HOST	futuretense.ini
com.fatwire.logging.cs.realtime	commons-logging.properties
xcelerate.imageeditor.clarkii.basepath	futuretense_xcel.ini

13. これまでの手順で書き留めた事項があれば、ここでまとめます。こうした事項は、アクティブ環境をアップグレードする際に使用できます。

手順 7. 環境におけるリモートの Satellite Server のアップグレード

注意

デプロイしたリモートの Satellite Server ごとに、次の手順を繰り返します。

1. リモートの Satellite Server をバックアップして、行ったカスタマイズがあれば追跡するようにします。それらのカスタマイズはすべて失われてしまうためです。
2. SatelliteServer のホーム・ディレクトリをアンデプロイして削除します。
3. リモートの Satellite Server をインストールします (Oracle WebCenter Sites: *Installing Satellite Server* のインストールを参照)。

注意

WebCenter Sites 11gR1 ではキャッシングが変更されたため、JVM ヒープの調整が必要になります。

手順 8. アップグレードされた環境の検証

アップグレードされた環境で、WebCenter Sites の各インスタンスに対して次の手順を完了します。

注意

クラスタがあれば、まずプライマリ・クラスタ・メンバー、さらにそれぞれのセカンダリ・クラスタ・メンバーをテストします。最後には、ロード・バランサでクラスタをテストします。

1. 全体の管理者として WebCenter Sites にログインします。
2. WebCenter Sites のインタフェース (Admin と Contributor) と、通常の使用で日常的に使用される機能をすべてテストします。たとえば、検索、カスタム・アセット・タイプの作成 / 編集、アセットの作成 / 編集、サイト・プレビュー、ワークフロー、承認 / パブリッシュなどです。インタフェースについては、WebCenter Sites インストール・ガイドのいずれかを参照してください。
3. 環境が検証されたら、次のいずれかを行います。
 - テスト環境をアップグレードした場合は、「[アクティブ環境のアップグレード](#)」に進みます。
 - アクティブ環境をアップグレードした場合は、次の手順を行います。
 - 1) パブリッシュ・スケジュールを有効化します。
 - 2) イベントがあれば有効化します。
 - 3) アップグレードした環境をバックアップします。

- 4) 新しいキャッシング構造でチューニングが必要であれば適用します。
- 5) クライアント・ブラウザのキャッシュをクリアします (キャッシュされた既存のコンテンツが問題を引き起こす可能性があるため)。
- 6) 作業を再開します。

アクティブ環境のアップグレード

テスト環境をアップグレードした場合に該当する手順は次のとおりです。

1. テスト環境の検証が済んでいるので、アクティブ環境をアップグレードします。
 - a. アクティブ環境は試用アップグレード中も運用を続行したと想定して、その Content Server のシステムを同期化します (必要に応じて)。
 - 1) 開発システムからコンテンツ管理システムにパブリッシュします。
 - 2) コンテンツ管理システムから配信システムにパブリッシュします。
 - b. この章のこれまでの手順を参照し、試用アップグレード中に作成したメモがあればそれも参照することによって、アクティブ環境をアップグレードします。
2. アップグレードした環境をバックアップします。
3. 作業を再開します。