



Prime Service Catalog のアップグレード

概要

この章では、以前のリリースの Cisco Prime Service Catalog からこのリリースにアップグレードする方法について説明します。

リリース アップグレードパス

このアップグレードプロセスでは、サービス カタログ (Service Catalog) リリース 10.1 以上からリリース 12.0.1 へのデータベース コンポーネントの直接アップグレードがサポートされています。データベーススキーマをサポートされているアップグレードレベルにするため、表 6-1 に示す最新サービス パックのデータベース インストーラ プログラムをデータベースに対して実行する必要があります。

表 6-1 直接アップグレードパス

アップグレード前のリリースバージョン	アップグレード後のリリースバージョン
11.0	12.0.1
11.1	12.0.1
11.1.1	12.0.1
12.0	12.0.1

11.0 より前のリリースからアップグレードする場合は、まず 11.0 インストーラを使用してリリース 11.0 にアップグレードします。その後、Prime Service Catalog のインスタンスを起動せずに 12.0.1 インストーラを実行し、インスタンスをリリース 12.0.1 にアップグレードします。

制限事項と注意事項

ここでは、製品の制限事項や、このバージョンにアップグレードする際に考慮する必要がある重要な注意事項を記載しています。

新しいプラットフォーム サポート

このリリースでのプラットフォーム サポートの変更については、『[Cisco Prime Service Catalog Compatibility Matrix](#)』を参照してください。

新しいプラットフォームの特徴は次のとおりです。

- リリース間のアップグレードでは、デフォルト ディレクトリを使用する代わりに、新しいフォルダをインストール ディレクトリとして使用します。既存のデータベースを引き続き使用できます。
- `kek_new.txt` ファイルと `kek_old.txt` ファイルが保存されていることを確認します。最新のインストーラへのアップグレード中に `kek_new.txt` が要求されることがあります。インストーラを続行するには、このファイルの提供が必要です。`kek` が一致しないと、PSC を開始できません。
- オペレーティング システム: Prime Service Catalog 12.0.1 では Windows Server 2012 R2 と Red Hat Enterprise Linux Server 7.2 だけがサポートされます。既存の Prime Service Catalog がこれ以外の Windows または Linux オペレーティング システムにインストールされている場合は、新たにサポートされるオペレーティング システムを使用した新しいマシンをセットアップし、必要な前提条件すべてに対応するようこのマシンを準備します。
- SQL Server: 既存のデータベースが SQL Server 2008 の場合、Prime Service Catalog のアップグレードを行う前に、「[アップグレード環境の準備](#)」の項で説明する作業を行う必要があります。
- Oracle: 既存のデータベースが Oracle 12c の場合、Prime Service Catalog のアップグレードを行う前に、「[アップグレード環境の準備](#)」の項で説明する作業を行う必要があります。
- 監査ログ データ: アップグレードの前に、以前のリリースの Prime Service Catalog 10.1/11.0 から Prime Service Catalog 12.0.1 に監査ログ データを移行します。詳細については、「[監査ログデータの移行](#)」の項を参照してください。
アップグレードが完了したら、変更データ キャプチャ機能を無効にします(詳細については「[アップグレード後の作業の実行](#)」の手順 8 を参照)。
- WebSphere: WebSphere Application Server はサポートされなくなりました。WebSphere を使用しており、Prime Service Catalog 12.0.1 へのアップグレードに関心がある場合は、Cisco TAC にご連絡ください。
- Oracle Java: Windows オペレーティング システムの WildFly と、すべてのオペレーティング システムの WebLogic アプリケーション サーバでは、Oracle JDK 8 Update 91 以上を使用する必要があります。Prime Service Catalog のアップグレードを実行する前に、マシンに Oracle JDK 8 Update 91 以上をインストールする必要があります。WebLogic を使用している場合は、正しいバージョンの JDK を使用するように、WebLogic の `commEnv.cmd` (または `commEnv.sh`) スクリプト内の `JAVA_HOME` プロパティを変更してください。クラスタ WebLogic 環境を使用している場合は、クラスタの各ノードでこのステップを実行する必要があります。
- OpenJDK: Red Hat Enterprise Linux オペレーティング システムの WildFly には、Red Hat OpenJDK 8 (バージョン 1.8.0_91 以上の 1.8.0_x) を使用する必要があります。Prime Service Catalog をインストールする前に、OpenJDK 1.8.0_91 以上の 1.8.0_x をマシンにインストールする必要があります。
- WebLogic: Prime Service Catalog リリース 10.0 以降では、WebLogic 12c (12.2.1) を使用する必要があります。古いリリースの Prime Service Catalog と古いバージョンの WebLogic を使用している場合は、Prime Service Catalog のアップグレードを実行する前に、最初に WebLogic をバージョン 10.3.6 にアップグレードするか、または WebLogic 12c (12.2.1) 環境を新規にインストールしてから、第 5 章の説明に従って WebLogic サーバを設定します。
- ブラウザ: Prime Service Catalog のこのリリースでサポートされる新しいブラウザの一覧については、『Cisco Prime Service Catalog Compatibility Matrix』を参照してください。

Rating ファセット

Rating という表示名のファセットがある場合は、アップグレード前にこのファセット名を必ず変更してください。これは、Cisco Prime Service Catalog 12.0.1 で使用可能な新しい Rating ファセットとの競合を回避するためです。

新しい JDBC ドライバ

Oracle と SQL Server 向けの新しい JDBC ドライバは、このリリースの Prime Service Catalog に付属しています。WebLogic アプリケーション サーバを使用している場合は、既存の REQUESTCENTERDS データソース (newscale_drivers.jar の DataDirect ドライバを使用していたデータソース) を削除し、新しい JDBC ドライバ (Oracle では ojdbc7.jar、SQL Server では sqljdbc4.jar) を使用して REQUESTCENTERDS データソースを再作成する必要があります。詳細については、[アップグレード環境の準備](#)の項で説明します。



(注) newScale 統合型ドライバを使用するカスタマイズがある場合、データベース タイプに基づいて適切な JDBC ドライバに置き換える必要があります。

Service Import/Export には下位互換性がない

Service Import/Export には、以前のリリースとの下位互換性はありません。以前のリリースでエクスポートしたサービスは、このリリースにインポートできません。アップグレードが完了したら、コードリポジトリに保持されているサービスを必ずエクスポートしてください。

Catalog Deployer パッケージには下位互換性がない

リリース 12.0.1 では Catalog Deployer における XML の書式が変更されたのに伴って、「Deployed」および「Received for Deployment」というステータスは使用できなくなりました。またコンテンツも表示できません。このため、リリース 12.0.1 へのアップグレードを実行する前に、展開が保留されているすべてのパッケージを処理する必要があります。

アップグレードされるオブジェクト

この章で説明するアップグレードプロセスでは、サービス カタログ (Service Catalog) の新機能をサポートするため、サービス カタログ (Service Catalog) ソフトウェアと Service Catalog データベースがアップグレードされます。これらの機能を使用することで、たとえば、データおよび参照整合性をさらに厳しく管理することで、サービス カタログ (Service Catalog) データベースのデータ品質を改善できます。

アップグレードされないオブジェクト

アップグレードプロセスでは、アプリケーションスキーマの一部として認識されない既存のデータベースのすべてのオブジェクトが特定されます。

- 認識されないオブジェクトが サービス カタログ (Service Catalog) のテーブルと対話する場合、そのオブジェクトはデータベースから自動的に削除されます。たとえば、次に示すオブジェクトが (存在する場合) 削除されます。
 - サービス カタログ (Service Catalog) テーブルの認識されないインデックス。
 - サービス カタログ (Service Catalog) テーブルの認識されないトリガー。
 - サービス カタログ (Service Catalog) テーブルの認識されない制約。
 - サービス カタログ (Service Catalog) テーブルを指す認識されない外部キー制約。
- サービス カタログ (Service Catalog) テーブルと関わりがなく認識もされないオブジェクトは報告されるだけで、削除されません。たとえば、存在する場合に削除されないオブジェクトには、テーブル、カラム、シーケンス、ストアドプロシージャ、関数、サービス カタログ (Service Catalog) のテーブルを参照しないインデックス、および サービス カタログ (Service Catalog) のテーブルに影響を与えない制約などがあります。

前提条件とベスト プラクティス

- アップグレードを行う前に、データベース バックアップおよびファイル システム バックアップを作成および検証する必要があります。アップグレードのロールバックは、データベースおよびファイル システムを手動で復元することでのみ可能なので、バックアップを作成しておくことは重要です (ロールバック機能はアップグレードプログラムに組み込まれていません)。
- アップグレード中は実稼働サイトが停止するため、メンテナンス期間内にアップグレードプロセスをスケジュールする必要があります。
- 10.0 以上からアップグレードします。[リリース アップグレードパス](#)。

前提条件

- アップグレードのサンドボックス環境。
- データベース バックアップおよび十分に計画された復元プロセス。
- すべてのカスタマイズの完全なリスト (カスタムのスタイル シート、JavaScript ライブラリ、LDAP Java マッピング コードなど)。
- データベースはすべて、[Service Catalog データベースの設定](#)に記載されている要件を満たしている必要があります。


ハイレベルアップグレードの手順

通常、組織ですでに サービス カタログ (Service Catalog) ソリューションのアップグレード方法を開発しています。または、他の企業のソフトウェアアップグレードのベストプラクティスを収集している場合もあります。本書で説明されている方法は、代替方法として実行したり、特定の新しいアップグレード要件に合わせて開発済みの方法を強化したりするときに役に立ちます。

既存の サービス カタログ (Service Catalog) システムのアップグレード手順の予行練習を行うサンドボックス環境を作ることを推奨します。この実行中に発生する技術的な問題や解決策を記録しておきます。このようにすることで、実稼働システムで実際のアップグレードするとき役に立ちます。予行練習をすることで、アップグレードプロセスの始まりから終わりまで全体的な時間がわかるので、実稼働システムのアップグレードに必要な適切なシステム ダウンタイムを計画するときにも役に立ちます。

サンドボックス環境でシステムを正常にアップグレードし、プロセスに問題がなければ、実稼働システムでのアップグレードを計画できます。アップグレードは、予行練習で準備した技術注記も参考にして、本書の指示に従い実行します。

ハイレベルでのアップグレード手順は次のようになります。

-
- 手順 1** 現在の実稼働システムのバックアップを作成し、データベースの別のセットに復元します。
-  **(注)** 新しい SQL Server 2012 SP23 データベースまたは Oracle 12c データベースにデータベース バックアップを復元します。
-
- 手順 2** リリース 12.0.1 のすべての前提条件 (新しいバージョンのアプリケーション サーバ、Java、および JDBC ドライバを含む) に対応したサンドボックス環境を作成します。これは、リリース 12.0.1 パッケージからアップグレードプログラムを実行する環境で、実稼働データベースのコピーに接続するよう設定する必要があります。
- 手順 3** 「[WildFly アプリケーション サーバでの Prime Service Catalog のインストール](#)」または「[WebLogic アプリケーション サーバでの Prime Service Catalog のインストール](#)」の説明に従い、Service Catalog セットアッププログラムを実行します。インストール ウィザードの [Service Catalog データベース (Service Catalog Database)] パネルで、実稼働データベースのコピーの接続情報を入力します。
- 手順 4** [次へ (Next)] をクリックします。
- 手順 5** [既存のデータベースをアップグレード (Upgrade Existing Database)] をクリックします。
- 手順 6** [スキーマの検証 (Validate Schema)] をクリックします。
- 手順 7** [エラーを表示 (View Errors)] をクリックします。データベースのスキーマ エラーが報告された場合は、データベース管理者やアプリケーション プログラマと連携して、そのスキーマ エラーを修復してください。スキーマ エラーによっては、エラーの修復に SQL コマンドが推奨されています。これがエラー状況に適しているかどうかは、データベース管理者およびアプリケーション プログラマと相談してください。また、スキーマ エラーによっては、適切に修復するために Cisco Technical Assistance Center (TAC) に連絡する必要があります。発生したすべての検証エラーとその解決策をすべて文書化します。
- 手順 8** 検証エラーを修復したら、そのたびに [スキーマの検証 (Validate Schema)] をクリックします。この作業をエラーがレポートされなくなるまで繰り返します。
- 手順 9** [データの検証 (Validate Data)] をクリックします。

- 手順 10 [エラーを表示 (View Errors)] をクリックします。データベースのデータ エラーが報告された場合は、データベース管理者やアプリケーション プログラマと連携して、そのデータ エラーを修復してください。データ エラーには、検証エラーと自動修復可能エラーの 2 種類があります。検証エラーによっては、エラーの修復に SQL コマンドが推奨されています。これがエラー状況に適しているかどうかは、データベース管理者およびアプリケーション プログラマと相談してください。また、スキーマ エラーによっては、適切に修復するために Cisco Technical Assistance Center (TAC) に連絡する必要があります。発生したすべての検証エラーとそれに対応する解決策をすべて文書化します。
- 手順 11 検証エラーを修復したら、そのたびに [データの検証 (Validate Data)] をクリックします。この作業を検証エラーがレポートされなくなるまで繰り返します。[自動修復可能なエラー (Auto-Repairable Errors)] はいくつかあっても問題ありません。これらは、次のステップでプログラムにより修復されます。
- 手順 12 [データベースの修復 (Repair Database)] をクリックします。この機能を使用すると、直前の手順で報告されたすべての自動修復可能エラーがプログラムによって修復されます。
- 手順 13 データベース修復機能が完了したら、[次へ (Next)] をクリックし、サービス カタログ (Service Catalog) インストール ウィザードの残りのページで必要な処理をすべて行います。
- 手順 14 インストール ウィザードによるアップグレード インストールが完了したら、必要なカスタマイズをサンドボックス環境に再適用します。
- 手順 15 サンドボックス環境のアップグレード システムに対してユーザ受け入れテストを実行します。
- 手順 16 プロセスにおいてこれまでに作成した技術注記をすべて収集します。
- 手順 17 この時点で、アップグレード プロセスに問題があると感じた場合、サンドボックス環境をクリーンアップして、サンドボックス環境ですべてのステップをやり直すこともできます。この場合、これまで収集した技術注記を参照し、本書の指示に従います。
- 手順 18 準備ができれば、アップグレード プロセス全体を実稼働環境で繰り返します。

Prime サービス カタログ (Service Catalog) のアップグレード

プラットフォームの中には一部、リリース 12.0 以降にサポートが終了したものがあります。したがって、Prime Service Catalog をアップグレードする前に、「リリース アップグレードパス」を参照し、データベース、アプリケーション サーバ、Java、Web サーバ、またはオペレーティング システムのバージョンをアップグレードする必要があるかどうかを確認してください。

既存の サービス カタログ (Service Catalog) が稼働するプラットフォームがサポートされなくなっている場合、第 2 章「インストール要件」で説明されているように、新しくサポートされるいずれかのプラットフォームのための新しい環境を準備する必要があります。

サービス カタログ (Service Catalog) のアップグレードでは次の操作を行います。

- 現行バージョンのアプリケーションの実行中に、アップグレード前の作業を行います
- サービス カタログ (Service Catalog) リリース 12.0.1 のインストール用に、新たにサポートされるプラットフォーム条件および前提条件を満たしているサンドボックス環境を準備します
- サービス カタログ (Service Catalog) インストール ウィザードを実行し、[既存のデータベースをアップグレード (Upgrade Existing Database)] をクリックします
- アップグレード前のデータベースの整合性を検証し、検出されたスキーマまたはデータの問題を修復します
- アップグレード後の作業を行います

以降の項で、詳しいアップグレード手順を説明します。

- [アップグレード前の作業の実行 \(6-7 ページ\)](#)
- [アップグレード環境の準備](#)
- [インストール ウィザードの実行 \(検証のページまで\)](#)
- [データベースの検証](#)
- [スキーマの検証](#)
- [データの検証](#)
- [データベースの修復](#)
- [インストールの完了](#)
- [アップグレード後の作業の実行](#)

アップグレード前の作業の実行

実稼働環境で、次に示すアップグレード前の必須作業を行います。

- 手順 1** Advanced Reporting モジュールを使用しない場合、手順 2 に進みます。Advanced Reporting モジュールを使用する場合、[Reporting のアップグレード前の作業の実行](#)で説明されている、そのためのいくつかのアップグレード前の作業を実行する必要があります。Advanced Reporting モジュールのアップグレード前の作業だけを実行したら、この項に戻り、ここで説明する手順を完了します。
- 手順 2** アプリケーション サーバのバージョンが変更された場合、Service Link の JMS キューに残っている未処理のメッセージが、新しいアプリケーション サーバへ自動的に移行されることはありません。アップグレード前に、未処理のメッセージがキューにないかチェックし、メッセージがある場合は処理しておく必要があります。キューがクリアになったら、すべての Service Link エージェントを停止します。これにより、アップグレード後にエージェントが再起動する前に Service Link 通信を確認できます。
- 手順 3** Catalog Deployer では、異なるリリース レベルの サービス カタログ (Service Catalog) 間でのパッケージの展開はサポートされません。そのため、アップグレード前に、展開できるすべてのアセンブルパッケージを展開したことを確認します。データベースがリリース 12.0.1 にアップグレードされた後でこれらを展開することはできません。また、アップグレードを実行する場合、現在のシステムでアセンブルする新しいパッケージに対して、説明されているリリース バージョンの サービス カタログ (Service Catalog) を含めることもできます。これにより、異なるリリース バージョンのパッケージを簡単に識別できます。
- 手順 4** アップグレード前の作業として、展開されるパッケージのリストを確認します。オンラインにしておく必要がないパッケージをエクスポートおよび削除できます。これらのパッケージは(アップグレードを実行すると)展開できなくなるので、これらをオンラインにしておくことは、展開履歴の照会のみ役に立ちます。これらのパッケージを削除することで、データベース容量が増加します。このようなクリーンアップ作業はすべてのシステム(開発、テスト/QA および実稼働)で実行できます。
- 手順 5** アプリケーション サーバのすべての サービス カタログ (Service Catalog) サービスを停止します。
- 手順 6** すべての サービス カタログ (Service Catalog) データベースをバックアップします。複数ある場合、すべての サービス カタログ (Service Catalog) 関連データベースをバックアップします。たとえば、Service Catalog データベースの他に、別個の Data Mart データベースまたは Content Store データベース (Cognos により使用される)がある場合があります。この場合は、すべてのデータベースをバックアップする必要があります。

- 手順 7 すべてのカスタマイズ スクリプトまたはファイルをバックアップします。アップグレード プロセスでは、既存のインストールに対するカスタマイズの内容は維持されません。そのため、アップグレード後も適切な場合、これらのカスタマイズの一部またはすべてをシステムに再適用する必要があります。
- 手順 8 インストール ディレクトリをバックアップします。元々 サービス カタログ (Service Catalog) ソフトウェアがインストールされていたディレクトリをバックアップします。
- 手順 9 監査ログ データを移行します。[監査ログ データの移行](#)を参照してください。
- 手順 10 次の[アップグレード環境の準備](#)に進みます。

監査ログ データの移行

古いリリース (Oracle 11g の 10.1 など) から現在のリリース (Oracle 12c の 12.0.1) に監査ログ データを移行するには、CDCADMIN スキーマを現在のリリースにインポートする必要があります。

アップグレード プロセスを実行する前に、次のコマンドを実行して、現在のリリースに適切な テーブル スペースを作成します。

1. create table space cdc_tbsp

```
CREATE TABLESPACE cdc_tbsp DATAFILE
'cdctbsp.dbf' SIZE 525M AUTOEXTEND ON NEXT 512M MAXSIZE 20G
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 1m
SEGMENT SPACE MANAGEMENT AUTO;
```

2. create schema cdcadmin

```
CREATE USER cdcadmin
IDENTIFIED BY cdcadmin
DEFAULT TABLESPACE cdc_tbsp
TEMPORARY TABLESPACE temp
QUOTA UNLIMITED ON cdc_tbsp;
```

上記のコマンドを実行したら、前に使用していた名前でも RCUser スキーマが準備されていることを確認し、作成された CDCADMIN スキーマに監査ログ データをインポートします。

アップグレード環境の準備

実稼働環境でのアップグレードの実行準備ができていない場合、この項をスキップします。

この項では、アップグレード プロセスの予行練習を実行するときに使用するサンドボックス環境を作成します。アップグレード プロセスに問題がなく、予行練習中に収集した技術注記を準備したら、実際の実稼働システムでアップグレード手順を開始できます。

はじめる前に

[アップグレード前の作業の実行 \(6-7 ページ\)](#)に記載されている必須のアップグレード前作業を実行します。

サンドボックス環境を準備するには、次の手順を実行します。

リリース番号の徹底的な確認が必要。

- 手順 1** 実稼働データベース バックアップを別のデータベース セットに復元します。SQL Server 環境の Service Catalog バージョン 10.1 で監査ログを有効にしている場合は、Service Catalog バージョン 10.1 から 12.0.1 へのアップグレード時にデータベースを復元するときに、次のコマンドを実行して監査ログ レコードを表示します。

同一ホスト上で復元を実行する場合:

```
RESTORE DATABASE DB_101_RESTORE FROM DISK='C:\Databases\Backup\DB_101.bak'
    その背景にあるのは
    MOVE 'DB_101' TO 'C:\Databases\Backup\DB_101.mdf',
    MOVE 'DB_101_log' TO 'C:\Databases\Backup\DB_101_log.ldf'
    , KEEP_CDC
```

異なるホスト上で復元を実行する場合:

```
RESTORE DATABASE DB_101_RESTORE FROM DISK='C:\Databases\Backup\DB_101.bak'

    その背景にあるのは
    MOVE 'DB_101' TO 'C:\Databases\Backup\DB_101.mdf',
    MOVE 'DB_101_log' TO 'C:\Databases\Backup\DB_101_log.ldf'
    , KEEP_CDC
```

```
USE DB_101_RESTORE
GO
EXEC sys.sp_cdc_add_job 'capture'
GO
EXEC sys.sp_cdc_add_job 'cleanup'
GO
```



- (注) デフォルトでは、Service Catalog 10.1 から Service Catalog 12.0 への直接アップグレードでデータベース バックアップ操作の監査ログが表示されます。

- 手順 2** Oracle DBMS を使用する場合、復元後に各データベースで統計情報の再コンパイルを実行することを推奨します。この手順は、大規模なデータベースでのアップグレードプロセスのパフォーマンスを改善するために不可欠です。第 3 章の Oracle の設定に関する項に説明されているすべての要件を満たしていることを確認します。

- 手順 3** データベースが SQL Server の場合、次の手順を実行して、READ_COMMITTED_SNAPSHOT をアクティブにする必要があります。

- a. SQL Server に「sa」ユーザとして接続し、SQL Server をシングルユーザ モードで設定します。
- b. 次のコマンドを実行します。<database_name> は、RequestCenter データベースの名前に置き換えます。

```
ALTER DATABASE <database_name> SET READ_COMMITTED_SNAPSHOT ON
GO
```

```
ALTER DATABASE <database_name> SET COMPATIBILITY_LEVEL=110
GO
```

- c. SQL Server をマルチユーザ モードに戻します。

- 手順 4** 第 1 章のデータベースの設定に関する項の説明に従い、データベース ユーザ「CPSCUser」に追加の権限を付与します。このリリースに必要な権限が「CPSCUser」に付与されているかどうかは確実ではない場合は、データベースの設定に関する項で説明するコマンドを再度実行できます。これらのコマンドを複数回実行しても、問題は発生しません。

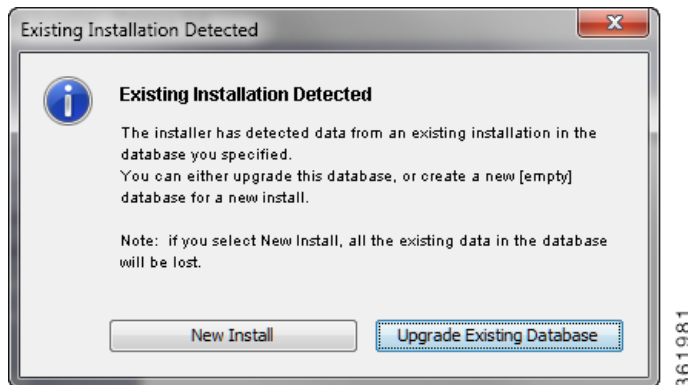
- 手順 5 『Cisco Prime Service Catalog Compatibility Matrix』で説明するサポートされるプラットフォームと前提要件情報を参照してください。プラットフォームのサポートが終了している場合は、サンドボックス環境で、サポートされているオペレーティングシステムに、正しいバージョンのアプリケーション サーバ、Web サーバおよび JDK をインストールします。
- 手順 6 「インストール前の作業: WildFly」または「インストール前の設定: WebLogic」の手順をすべて読み、既存のアプリケーション サーバ環境で設定を再設定または調整する必要があるかどうかを確認します。新しい jar ファイルをコピーする必要がある場合や、このリリースの Prime Server Catalog で新しい設定を追加する必要がある場合があります。
- 手順 7 Prime Service Catalog のこのリリースには新しい JDBC ドライバが付属しているため、WebLogic 環境の既存の REQUESTCENTERDS データソースと、DATAMARTDS データソースが存在する場合は DATAMARTDS データソースも削除する必要があります。WildFly 環境ではこの作業を行う必要はありません。新しい JDBC ドライバを使用するには、REQUESTCENTERDS データソースを再作成する必要があります。WebLogic でのデータソースの作成手順については、JDBC データソースの設定を参照してください。DATAMARTDS データソースは不要になったことに注意してください。
- 手順 8 次のインストール ウィザードの実行(検証のページまで)に進みます。

インストール ウィザードの実行(検証のページまで)

インストール ウィザードを実行してアップグレードを行うには、次の手順を実行します。

- 手順 1 「WildFly アプリケーション サーバでの Prime Service Catalog のインストール」または「WebLogic アプリケーション サーバでの Prime Service Catalog のインストール」の説明に従って、サービス カタログ (Service Catalog) インストール ウィザードで標準(Typical)およびカスタム(Custom)インストールを実行します。
- 手順 2 インストール ウィザードの [Service Catalog データベース (Service Catalog Database)] パネルで、データベースの値を入力します。
- 手順 3 [次へ(Next)] をクリックして、ウィザードの次のページに進みます。
- 図 6-1 に示す [既存のインストールが検出されました (Existing Installation Detected)] ダイアログボックスが表示されます

図 6-1 既存のインストールが検出されました (Existing Installation Detected)



- 手順 4 [既存のデータベースをアップグレード(Upgrade Existing Database)] をクリックします。

インストール ウィザードの [検証 (Validation)] ページが表示されます。

既存のデータベースは、検証および修復が正常に完了するまで、サービス カタログ (Service Catalog) インストール ウィザードでアップグレードすることはできません。[検証 (Validation)] ページでは、次の各機能を上から順に実行する必要があります。

1. [スキーマの検証 (Validate Schema)]
2. [データの検証 (Validate Data)]
3. [データベースの修復 (Repair Database)]

[スキーマの検証 (Validate Schema)] をまだ一度も実行していない段階で、[データの検証 (Validate Data)] を実行することはできません。また、[データの検証 (Validate Data)] をまだ一度も実行していない段階で、[データベースの修復 (Repair Database)] を実行することはできません。

各機能は複数回実行できます。ただし、[スキーマの検証 (Validate Schema)] を実行するたびに、プログラムは再初期化されるため、検証プロセスは実質的に最初から実行されます。たとえば、[データの検証 (Validate Data)] まで完了している場合は、[データベースの修復 (Repair Database)] を開始することができますが、代わりに [スキーマの検証 (Validate Schema)] を再度実行することも可能です。ただしこの場合には、[スキーマの検証 (Validate Schema)] を実行したことによりシステムが再初期化されるため、[スキーマの検証 (Validate Schema)] が完了した後は、ただちに [データベースの修復 (Repair Database)] を実行することはできず、次に [スキーマの検証 (Validate Schema)] を実行する必要があります。

手順 5 次のデータベースの検証に進みます。

データベースの検証

Prime Service Catalog のアップグレード作業におけるデータベースの検証プロセスについて十分な知識がある場合は、この項をスキップしてスキーマの検証に進んでください。[スキーマの検証 (Validate Schema)] および [データの検証 (Validate Data)] の実行時にエラーが発生したときは、この項に戻ればいつでもその内容を確認することができます。

[検証 (Validation)] ページの各機能は、既存のスキーマおよびデータベースの整合性を検証するためのものです。いずれの検証スクリプトも、その結果はデータベース内の SchValidationLog というテーブルに格納されます。このログ ファイルの表示方法については、SchValidationLog テーブルを参照してください。次の表 6-2 に、検証エントリの全エラー レベル (ErrorLevel) とその内容に関する説明を示します。

表 6-2 検証エラー レベル

エラーレベル	説明
inform	<p>検証テストにより異常が検出されたものの、アプリケーションやアップグレードプロセスには悪影響が及ばないことを表します。データベース スキーマに属さないテーブルが検出された場合などが、このエラーに該当します。このようなテーブルが存在しても、アップグレードが正常に実行されないことや、アプリケーションに悪影響が及ぶことはありません。</p> <p>SchValidationLog テーブルでは、サービス カタログ (Service Catalog) テーブルと関わりがなく認識もされないこうしたオブジェクトには、「inform」というステータスが表示されます。サービス カタログ (Service Catalog) テーブルと関わりがないテーブル、列、シーケンス、ストアドプロシージャ、関数、インデックス、制約の各オブジェクトは、存在したとしても報告されるだけです。これらのオブジェクトは報告されるだけで、その後の処理は行われません。</p> <p>ErrorLevel が「inform」の検証エントリはすべて無視されます。</p> <p>これらのエントリに対処するためのユーザによる操作は不要です。</p>

表 6-2 検証エラー レベル(続き)

エラーレベル	説明
inform: auto-repairable	<p>データ検証テストにより、悪影響を及ぼすようなデータ エラーが検出されたことを表します。ただし、このタイプのエラーは [データベースの修復 (Repair Database)] を使用すれば、安全かつプログラマ的に修復することができます。</p> <p>このタイプのほとんどのエラーは、内部不一致が原因です。これは、以前のアップグレードまたはインポート ユーティリティにより発生したと考えられます。修復するには、通常、参照およびデータの整合性を復元します。</p> <p>SchValidationLog テーブルの RepairScript 列には、エラー修復に使用される SQL ステートメントが表示されます。</p> <p>ユーザの操作は不要です。</p>
inform: auto-repaired	<p>[データベースの修復 (Repair Database)] 機能を使用して RepairScript 列に記述されている SQL ステートメントを実行したことにより、前述の「inform: auto-repairable」としてレポートされたエラーが修復されたことを表します。</p> <p>ErrorLevel が「inform:auto-repaired」である検証エントリはすべて、SchValidationLog テーブルに表示されるほか、[データベースの修復 (Repair Database)] の実行時に [エラーを表示 (View Errors)] をクリックすれば Validation Errors テーブルにも表示されます。</p> <p>ユーザの操作は不要です。</p>
inform: pending-repair	<p>検証テストにより、アップグレードプロセスに悪影響を及ぼすような異常が検出されたものの、アップグレードプロセスの中でプログラマ的に修復できることを表します。</p> <p>SchValidationLog テーブルでは、欠落したオブジェクトや変更されたオブジェクトに「inform: pending repair」というステータスが表示されます。このタイプのエラーの例としては、インデックスまたはプライマリ キー制約の欠落があります。欠落したインデックスやプライマリ キー制約は、もし存在しなければ適切に作成されます。</p> <p>ユーザの操作は不要です。</p>
inform: pending-removal	<p>検証テストは、サービス カタログ (Service Catalog) テーブルを示す認識されないデータベース オブジェクトを検出しました。このオブジェクトが存在すると、アップグレードが正常に完了しないことがあります。SchValidationLog テーブルでは、これらの認識されないオブジェクトには、「inform: pending-repair」というステータスが表示されます。</p> <p>たとえば、存在している場合に削除対象としてフラグが設定されるオブジェクトには、a) サービス カタログ (Service Catalog) テーブルの認識されないインデックス、b) サービス カタログ (Service Catalog) テーブルの認識されないトリガー、c) サービス カタログ (Service Catalog) テーブルの認識されない制約、および d) サービス カタログ (Service Catalog) テーブルを指す認識されない外部 キー制約があります。これらのオブジェクトは自動的に削除されます。</p> <p>ユーザの操作は不要です。</p>

表 6-2 検証エラー レベル(続き)

エラーレベル	説明
error	<p>検証テストにより、プログラムの修復できないハード「エラー」が検出されたことを表します。通常、このタイプのエラーは、欠落行や重複エントリなど、不正なデータ関係に関連します。</p> <p>ErrorLevel が「error」である検証エントリはすべて、SchValidationLog テーブルに表示されるほか、[スキーマの検証 (Validate Schema)] または [データの検証 (Validate Data)] の実行時に [エラーを表示 (View Errors)] をクリックすれば Validation Errors テーブルにも表示されます。</p> <p>TestType 列は、エラーのタイプを示し、TestDetail 列は、検証テストに使用された SQL ステートメントを示します。この SQL ステートメントは、エラーに関するいくつかのヒントを示します。このエラーが発生すると、アップグレードが正常に実行されない場合があります。そのため、このようなエラーが検出された場合は、アップグレードを開始する前にそのエラーを修復する必要があります。</p> <p>アプリケーション管理者またはデータベース管理者は、このタイプのすべてのエラーを手動で修復し、エラーがレポートされなくなるまで、同じ検証機能を繰り返し実行する必要があります。</p> <p>検証エラーの修復方法については、Cisco Technical Assistance Center (TAC) にお問い合わせください。場合によっては、RepairScript 列に、エラー修復に使用できる推奨 SQL ステートメントが示されます。アプリケーション管理者またはデータベース管理者と協力して、このような修復スクリプトが実際のエラー修復に使用できるか確認してください。</p> <p>各検証エラーがどのように修復されるかを簡潔に説明します。別の環境でアップグレード手順を繰り返す必要がある場合、メモを取っておく必要があります。</p>

上記の表 6-2 に記載されているように、[スキーマの検証 (Validate Schema)] または [データの検証 (Validate Data)] の実行時に [エラーを表示 (View Errors)] をクリックして表示した Validation Errors テーブルで、ErrorLevel が「error」となっている検証エントリのみ、手動で修復する必要があります。SchValidationLog テーブルで報告されているその他の検証エラーはすべて、自動的に処理されます。

ErrorLevel が「error」である検証エントリをすべて修復したら、同じ検証機能を再度実行して、これ以上エラーがレポートされないことを確認します。手動修復の結果として、新しい検証エラーが表示されることもあります。この場合には、再び検証機能を繰り返し実行して、検証エラーをすべて修復する必要があります。

すべてのスキーマ検証エラーを確認および修復してから、すべてのデータ検証エラーを確認および修復することを強く推奨します。このようにすることで、スキーマ検証エラーの修復とデータ検証エラーの修復が混在することによる回帰エラーの可能性を軽減できます。

SchValidationLog テーブル

すべての検証スクリプトの結果は、検証エラーがあるかどうかに関係なく、データベースの SchValidationLog というテーブルに保存されます。

SchValidationLog テーブルを表示するには、次の手順を実行します。

- 手順 1** データベースにスキーマ所有者(つまり、CPSCUser)として接続し、テーブル SchValidationLog を参照して検証結果を確認します。SQL Analyzer(図 6-2 を参照)や SQL*Plus などのユーティリティを使用してデータベースに接続できます。

図 6-2 SchValidationLog テーブルの参照

SchFunctions	dbo	User	11/5/2007 5:28:25 PM
SchIndColumns	dbo	User	11/5/2007 5:28:25 PM
SchIndexes	dbo	User	11/5/2007 5:28:25 PM
SchObjectActionLog	dbo	User	11/5/2007 5:28:25 PM
SchObjectExceptions	dbo	User	11/5/2007 5:28:25 PM
SchPackages	dbo	User	11/5/2007 5:28:25 PM
SchPrimaryKeys	dbo	User	11/5/2007 5:28:25 PM
SchProcedures	dbo	User	11/5/2007 5:28:25 PM
SchSequences	dbo	User	11/5/2007 5:28:25 PM
SchTabColumns	dbo	User	11/5/2007 5:28:25 PM
SchTables	dbo	User	11/5/2007 5:28:25 PM
SchTriggers	dbo	User	11/5/2007 5:28:25 PM
SchValidationLog	dbo	User	11/5/2007 5:28:25 PM
SchValidationRunAttributes	dbo	User	11/5/2007 5:28:25 PM
SchViewColumns	dbo	User	11/5/2007 5:28:25 PM
SchViews	dbo	User	11/5/2007 5:28:25 PM

手順 2 SchValidationLog テーブルを開いて内容を確認します(図 6-3 を参照)。

図 6-3 SchValidationLog の内容

RunType	TestType	ObjectType	ObjectName	ObjectShortName	ErrorLevel	TestDetail
Check Schema	Table column ty...	Column	DirProject.Perce...	DirProject.Perce...	inform	Expected type: ...
Check Schema	Table column siz...	Column	DirPerson.SSN	DirPerson.SSN	inform	Expected type: ...
Check Schema	Unexpected tabl...	Table	CUSTOMER_TAB...	CUSTOMER_TAB...	inform	NULL
Check Schema	Unexpected tabl...	Table	CUSTOMER_TAB...	CUSTOMER_TAB...	inform	NULL
Check Schema	Unexpected tabl...	Table Column	DirPermission.C...	DirPermission.C...	inform	NULL
Check Schema	Unexpected tabl...	Table Column	DirPerson.CUST...	DirPerson.CUST...	inform	NULL
Check Schema	Unexpected FK ...	FK	FK_CUSTOMER_...	Table name: CU...	pending removal	NULL
Check Schema	Expected index ...	Index	TxInvocationAtt...	TxInvocationAtt...	inform:pending-r...	NULL
Check Schema	Expected index ...	Index	TxObjectRelatio...	TxObjectRelation3	inform:pending-r...	NULL
Check Schema	Expected index ...	Index	TxSatisfaction.T...	TxSatisfaction 1	inform:pending-r...	NULL
Check Schema	Expected index ...	Index	XtrAgent.XtrAge...	XtrAgent3	inform:pending-r...	NULL

手順 3 SchValidationLog テーブルの ErrorLevel 列で次の値をチェックして、推奨されている操作を実行します。

手順 4 SchValidationLog テーブルには、多数のエントリが示されます。そのため、次の SQL コマンドを使用して、内容をフィルタリングできます。

```
SELECT * FROM SchValidationLog WHERE ErrorLevel= "error"
AND RunType= "Check Data";
```

アップグレードプロセスを開始する前に手動で修復する必要がある検証エラーを確認する場合は、WHERE 句「ErrorLevel= 'error'」を指定します。SchValidationLog テーブルの他のエントリを参照する場合、WHERE 句を削除するか、値「error」を別の値に変更します(たとえば、「inform:auto-repairable」です。詳細については、表 6-2 を参照してください)。

SchValidationLog テーブルの RunType 列を参照します。

- [スキーマの検証 (Validate Schema)] を実行すると、RunType が「Check Schema」であるエントリが挿入されます。
- [データの検証 (Validate Data)] を実行すると、RunType が「Check Data」であるエントリが挿入されます。
- [データベースの修復 (Repair Database)] を実行すると、ErrorLevel が「inform: auto-repairable」であるすべてのエントリについて、ErrorLevel が「inform: auto-repaired」に、RunType が「Fix Data」にそれぞれ変更されます。

スキーマの検証

スキーマを検証するには、次の手順に従います。

手順 1 [スキーマの検証 (Validate Schema)] をクリックします。

スキーマの検証テストが完了した時点で検証エラーが検出されなかった場合は、「完了 (Completed)」というメッセージが表示されます。

エラーがレポートされない場合は、[データの検証](#)に進みます。

スキーマの検証テストが完了した時点で検証エラーが検出された場合は、次のように「完了しましたがエラーが発生しました (Completed with errors)」というメッセージが表示されます。

手順 2 [エラーを表示 (View Errors)] をクリックします。

[検証エラー (Validation Errors)] ウィンドウが表示されます。ウィンドウのサイズを変更すれば、次の例のようにテーブル全体を表示することができます (図 6-4)。

図 6-4 検証エラー (スキーマの検証)

Error Level	Test Type	Object Type	Test Detail	Object Short Name	Runtime Timestamp
error	Invalid data relationship	ColConstraint	SELECT * FROM DefService WHERE AreaID IS NOT NULL AND NOT EXISTS (SELECT AreaID FROM DefArea DefArea2 WHERE DefArea2.AreaID = DefService.AreaID)	FK_DefS_DefArea	2012-07-18 18:15:46.233

手順 3 Validation Errors テーブルに示されるエラーは、アップグレードプロセスを続行する前に手動で修復する必要があります。詳細については、[データベースの検証](#)を参照してください。

手順 4 すべての検証エラーが修復されたら、次の[データの検証](#)に進みます。

データの検証

データを検証するには、次の手順を実行します。

-
- 手順 1** [データの検証 (Validate Data)] をクリックします。
データの検証テストが完了した時点で検証エラーが検出されなかった場合は、次のように「完了 (Completed)」というメッセージが表示されます。
エラーがレポートされない場合は、[データベースの修復](#)に進みます。
データの検証テストが完了した時点で検証エラーが検出された場合は、「完了しましたがエラーが発生しました (Completed with errors)」というメッセージが表示されます。
- 手順 2** [エラーを表示 (View Errors)] をクリックします。
[検証エラー (Validation Errors)] ウィンドウが表示されます。
- 手順 3** Validation Errors テーブルに示されるエラーは、アップグレードプロセスを続行する前に手動で修復する必要があります。詳細については、[データベースの検証](#)を参照してください。
- 手順 4** すべての検証エラーが修復されたら、次の[データベースの修復](#)に進みます。
-

データベースの修復

データベースを修復するには、次の手順を実行します。

-
- 手順 1** [データベースの修復 (Repair Database)] をクリックします。
データベースの修復が完了すると、「完了 (Completed)」というメッセージが表示されます。
- 手順 2** (オプション)[エラーの表示 (View Errors)] をクリックします。
[検証エラー (Validation Errors)] ウィンドウが表示されます。テーブル全体が表示されるようにウィンドウのサイズを変更します。
表示されている検証エラーは、ErrorLevel が「inform:auto-repaired」となっているもの、つまり [データベースの修復 (Repair Database)] を使用してプログラマ的に修復された検証エラーのみです。
- 手順 3** 次の[インストールの完了](#)に進みます。
-

インストールの完了

データベースの検証および修復が完了すれば、いつでもアップグレードプログラムを開始することができます。

-
- 手順 1** [次へ (Next)] をクリックして、サービス カタログ (Service Catalog) インストール ウィザードの次のページに進みます。
- 手順 2** 「[WildFly アプリケーション サーバでの Prime Service Catalog のインストール](#)」または「[WebLogic アプリケーション サーバでの Prime Service Catalog のインストール](#)」の手順に従って、インストール ウィザードを続行します。

インストール ウィザードの [インストール前の要約 (Pre-Installation Summary)] ページで [インストール (Install)] をクリックすると、インストーラによりアップグレード スクリプトが実行され、データベースのスキーマおよびコンテンツが変更されます。データベースのサイズによっては、アップグレード スクリプトの実行に時間がかかることがあります。アップグレード スクリプトによりデータベース スキーマおよびコンテンツが変更されたら、インストーラは WAR ファイルの作成を開始します。「WildFly のインストール後の作業」の項または「インストール後の設定: WebLogic」の項に記載されている同じ手順に従って、サービス カタログ (Service Catalog) 製品の WAR ファイルを展開します。

- 手順 3** WAR ファイルの展開が終了し、アプリケーション サーバを起動できるようになると、アップグレード プロセスは実質的に完了です。これで、サービス カタログ (Service Catalog) アプリケーションがリリース 12.0 になりました。この時点で、必要に応じて、データベースおよびインストール ディレクトリのバックアップを作成できます。Oracle DBMS を使用する場合、システムのランタイム パフォーマンスを改善するため、アップグレードしたデータベースで統計情報の再コンパイルを再実行することを推奨します。

アップグレード後の作業の実行

アップグレード後の作業を行うには、次のようにします。

- 手順 1** 必要に応じて、インストーラによりデータベースから削除されたカスタム データベース オブジェクトを再作成します。
- 手順 2** カスタム コードは、新しいバージョンの JDK に対応している必要があります。
- Service Link カスタム アダプタは Service Link ADK リリース 12.0 バージョンを使用して再構築する必要があります。
 - カスタマー サイトで開発されたカスタム Java コードは、新しい JDK を使用して再構築する必要があります。
 - Service Catalog Portlet の展開先エンタープライズ ポータルは、JDK 8 Update 91 を使用している必要があります。
- 手順 3** Service Import/Export には、以前のリリースとの下位互換性はありません。以前のリリースでエクスポートしたサービスをリリース 12.0 にインポートすることはできません。アップグレード前のコード リポジトリにあるサービス エクスポート ファイルを保守する場合は、そのファイルを再エクスポートしてリリース 12.0 用にマークします。



(注) 異なるデータベースから (たとえば Catalog Deployer を使用して) インポートされたユーザは、アプリケーションにログインするためのパスワードをリセットする必要があります。これはデータベースごとにキー暗号キーが異なるためです。

- 手順 4** 以前に組織で使用した手順に従い、アプリケーションのすべてのカスタマイズを再実装します。
- 手順 5** サービス カタログ (Service Catalog) アプリケーションに管理者ユーザとして接続します。「Administration」モジュールに移動し、[設定 (Settings)] タブをクリックします。[カスタマイズ設定 (Customizations Settings)] で、[ブラウザキャッシュ (Browser Cache)] を探します (図 6-5 を参照)。

図 6-5 ブラウザ キャッシュ設定の有効化

Browser Cache:	<input checked="" type="radio"/> Enabled <input type="radio"/> Disabled	The Browser Cache setting enables the browser-side caching of images, javascripts, css, etc., which may improve performance. When the Version setting value is incremented, the login process is interrupted until the browser's cache is deleted. Default is Disabled.
Version:	2 <input style="float: right;" type="button" value="+"/>	
<input type="button" value="Update"/>		

361973

- 手順 6** [ブラウザキャッシュ (Browser Cache)] 設定の [有効 (Enabled)] オプション ボタンを選択します。[バージョン (Version)] テキスト ボックスの右にある [+] ボタンをクリックします。クリックすると、[バージョン (Version)] の数値が 1 つずつ増加します。次に [更新 (Update)] をクリックします。この設定により、サービス カタログ (Service Catalog) システムのアップグレード後にユーザが初めて サービス カタログ (Service Catalog) URL に接続するときにブラウザのキャッシュをクリアするよう通知されます。
- 手順 7** <ServiceCatalog_Install_Dir>/dist ディレクトリにあるマスター キー パスワード ファイル kek_new.txt と kek_old.txt のバックアップを安全な場所に作成する必要があります。
- 手順 8** 監査ログ データを移行したら、データベース ベンダー提供の変更データ キャプチャ機能を無効にします (これ以降、Prime Service Catalog 12.0 がこの機能を使用しないため)。
- 次のストアード プロシージャを実行して、SQL Server の変更データ キャプチャを無効にします。
`sp_DisableCDC()`
 - Oracle 12c で、CDCADMIN スキーマを削除します。

次の章では、Reporting モジュールの Cognos コンポーネントのアップグレード手順について説明します。アップグレード前の サービス カタログ (Service Catalog) システムに Reporting モジュールがインストールされていた場合は、次の章に進み、Reporting モジュールがリリース 12.0 で機能するように、Cognos コンポーネントのアップグレードプロセスを完了する必要があります。