



Informatica® Multidomain MDM
10.4

IBM WebSphere を使用する Oracle データベース向け インストールガイド

本ソフトウェアおよびマニュアルは、使用および開示の制限を定めた個別の使用許諾契約のもとでのみ提供されています。本マニュアルのいかなる部分も、いかなる手段（電子的複製、写真複製、録音など）によっても、Informatica LLC の事前の承諾なしに複製または転載することは禁じられています。

米政府の権利プログラム、ソフトウェア、データベース、および関連文書や技術データは、米国政府の顧客に配信され、「商用コンピュータソフトウェア」または「商業技術データ」は、該当する連邦政府の取得規制と代理店固有の補足規定に基づきます。このように、使用、複製、開示、変更、および適応は、適用される政府の契約に規定されている制限およびライセンス条項に従うものとし、政府契約の条項によって適当な範囲において、FAR 52.227-19、商用コンピュータソフトウェアライセンスの追加権利を規定します。

Informatica、Informatica ロゴ、および ActiveVOS は、米国およびその他の国における Informatica LLC の商標または登録商標です。Informatica の商標の最新リストは、Web (<https://www.informatica.com/trademarks.html>) にあります。その他の企業名および製品名は、それぞれの企業の商標または登録商標です。

本ソフトウェアまたはドキュメンテーション（あるいはその両方）の一部は、第三者が保有する著作権の対象となります。必要な第三者の通知は、製品に含まれています。

本マニュアルの情報は、予告なしに変更されることがあります。このドキュメントで問題が見つかった場合は、infa_documentation@informatica.com までご報告ください。

Informatica 製品は、それらが提供される契約の条件に従って保証されます。Informatica は、商品性、特定目的への適合性、非侵害性の保証等を含めて、明示的または黙示的ないかなる種類の保証をせず、本マニュアルの情報を「現状のまま」提供するものとします。

目次

序文	7
Informatica のリソース.....	7
Informatica Network.....	7
Informatica ナレッジベース.....	7
Informatica マニュアル.....	7
Informatica 製品可用性マトリックス.....	8
Informatica Velocity.....	8
Informatica Marketplace.....	8
Informatica グローバルカスタマサポート.....	8
 第 1 章: インストールの概要	9
Multidomain MDM のインストール.....	9
インストールのトポロジ.....	11
インストールのタスク.....	12
 第 2 章: インストール前のタスク	13
インストールの準備.....	13
環境の準備.....	15
データベース環境の設定.....	16
手順 1. Oracle のインストールと設定.....	17
手順 2. データベースとテーブルスペースの作成.....	17
手順 3. データベース特権と接続の設定.....	20
手順 4. ActiveVOS スキーマの作成.....	20
アプリケーションサーバー環境の設定.....	22
Java 仮想マシンの設定.....	22
Transport Layer Security (TLS) を設定する.....	25
セキュアな HTTP 応答ヘッダー.....	25
MDM Hub 環境でのパスワードの暗号化.....	25
WebSphere 環境での安全なプロファイルの作成.....	26
JAAS アプリケーションログインの設定.....	27
ActiveVOS コンソール管理ユーザーの作成.....	27
MDM Hub デプロイメント用の SOAP 要求タイムアウトの設定.....	28
アプリケーションサーバーの追加設定 (オプション).....	28
スタンドアロンプロセスサーバーインスタンス用の WebSphere の設定.....	28
複数の MDM Hub マスタデータベース用の WebSphere の設定.....	32
HTTPS プロトコルの設定.....	32
Informatica Data Director 用の WebSphere の設定.....	33
サイレントインストールのためのプロパティファイルの設定.....	33
Informatica プラットフォームのプロパティファイルの設定.....	33
Hub サーバーのプロパティファイルの設定.....	34

プロセスサーバーのプロパティファイルの設定.	34
第 3 章 : Hub ストアのインストール.	35
MDM Hub マスターデータベースの作成.	35
オペレーショナル参照ストアの作成.	37
MDM Hub マスターデータベースへのメタデータのインポート.	39
オペレーショナル参照ストアへのメタデータのインポート.	40
第 4 章 : Hub ストアのインストール後のタスク.	43
Oracle コンポーネントへのアクセスの検証.	43
第 5 章 : Hub サーバーのインストール.	45
Hub サーバーのインストール.	45
インストーラワークフローの確認.	46
インストールの値の収集.	47
Hub サーバーをインストールウィザードからインストールする.	53
Hub サーバーをコマンドラインからインストールする (UNIX のみ)	54
Hub サーバーをサイレントモードでインストールする.	54
クラスタ内のノードに Hub サーバーをインストールする.	55
第 6 章 : Hub サーバーのインストール後のタスク.	57
インストールログファイルのコピー.	57
バージョンとビルド番号の確認.	58
Elasticsearch のインストールおよび設定.	59
Hub コンソールクライアントの構成 (条件付き)	59
MDM Hub マスターデータベース名の設定.	59
Hub サーバー EAR ファイルの再デプロイ.	60
クラスローダーの設定.	60
アプリケーションサーバー設定の検証と設定 (条件付き)	61
アプリケーションサーバー設定の編集.	61
WebSphere マルチノードまたはクラスタ環境用の Hub サーバーの設定.	62
Hub サーバーアプリケーションのデプロイ (条件付き)	62
Hub サーバーアプリケーションをデプロイするスクリプトの使用 (条件付き)	63
Hub サーバーアプリケーションの手動デプロイ (条件付き)	65
手順 1. データソースの作成.	65
手順 2. JMS メッセージキューの設定.	68
手順 3. Hub サーバーの EAR ファイルの再パッケージ化.	71
手順 4. Hub サーバーアプリケーションのデプロイ.	72
手順 5. クラスローダーの設定.	72
手順 6. Hub サーバーでの JMS メッセージキューの設定.	73
手順 7. Informatica Data Director のサーバーリソースの設定.	75
メタデータキャッシュの構成 (省略可能)	75
Infinispan 属性の編集.	76

Hub コンソールの起動.	77
オペレーショナル参照ストアの登録.	78
第 7 章 : Process サーバーのインストール.	81
プロセスサーバーのインストール.	81
インストーラワークフローの確認.	82
インストールの値の収集.	83
Process サーバーをインストールウィザードからインストールする.	86
プロセスサーバーをコマンドラインからインストールする (UNIX のみ)	87
Process サーバーをサイレントモードでインストールする.	88
クラスタ内のノードに Process サーバーをインストールする.	89
第 8 章 : Process サーバーのインストール後のタスク.	90
インストールログファイルのコピー.	90
バージョンとビルド番号の確認.	91
WebSphere マルチノードまたはクラスタ環境用のプロセスサーバーの設定.	91
プロセスサーバー EAR ファイルの再デプロイ.	92
クラスローダーの設定.	92
プロセスサーバーアプリケーションのデプロイ (条件付き)	93
手順 1. データソースの作成 (条件付き)	93
手順 2. プロセスサーバーアプリケーションのデプロイ (条件付き)	97
手順 3. クラスローダーの設定.	98
プロセスサーバーの保護された通信の有効化.	99
Elasticsearch のインストールおよび設定.	100
一致ポピュレーションの設定.	100
一致ポピュレーションの有効化.	101
プロセスサーバーとクレンジングエンジンの設定.	102
第 9 章 : アプリケーションサーバーに対する ActiveVOS のインストール後のタスク.	103
WebSphere クラスタ環境での ActiveVOS のインストールおよびデプロイ.	103
WebSphere 作業マネージャの設定.	104
WebSphere タイムマネージャの設定.	104
JAAS アプリケーションログインの設定.	105
ActiveVOS サーバーおよび ActiveVOS Central のインストールの完了.	105
ActiveVOS インストールファイルの編集.	106
ActiveVOS および Identity Resolution のデプロイ.	106
WebSphere 環境での信頼されたユーザーの作成.	107
安全なプロファイルへのユーザーとグループの追加.	107
第 10 章 : ビジネスエンティティアダプタに対する ActiveVOS のインストール後のタスク.	108
ActiveVOS Web アプリケーション.	108

ビジネスエンティティワークフローアダプタの ActiveVOS URN の設定.	109
ActiveVOS URL のプロトコルの設定.	109
ActiveVOS プロトコルの HTTPS への設定.	110
プライマリワークフローエンジンの設定.	111
ActiveVOS 用の MDM Identity Service の設定.	111
タスクの設定.	112
第 11 章: リソースキットのインストール.	113
MDM Hub サンプルオペレーショナル参照ストアの設定.	113
Informatica MDM Hub サンプルオペレーショナルリファレンスストアの登録.	116
グラフィカルモードでのリソースキットのインストール.	118
コンソールモードでのリソースキットのインストール.	121
サイレントモードでのリソースキットのインストール.	123
プロパティファイルの設定.	123
サイレントインストーラの実行.	125
第 12 章: リソースキットのインストール後の作業.	127
sip_ant スクリプトの編集.	127
postInstall スクリプトの手動実行.	127
第 13 章: MDM Hub のトラブルシューティング.	129
インストールプロセスのトラブルシューティング.	129
第 14 章: アンインストール.	133
アンインストールの概要.	133
Hub Store のアンインストール.	133
グラフィカルモードでの Process サーバーのアンインストール.	134
UNIX におけるグラフィカルモードでのプロセスサーバーのアンインストール.	134
Windows におけるグラフィカルモードでのプロセスサーバーのアンインストール.	134
グラフィカルモードでの Hub サーバーのアンインストール.	135
UNIX におけるグラフィカルモードでの Hub サーバーのアンインストール.	135
Windows におけるグラフィカルモードでの Hub サーバーのアンインストール.	135
グラフィカルモードでのリソースキットのアンインストール.	136
UNIX でのグラフィカルモードによるリソースキットのアンインストール.	136
Windows でのグラフィカルモードによるリソースキットのアンインストール.	136
コンソールモードでのプロセスサーバーのアンインストール.	137
コンソールモードでの Hub Server のアンインストール.	137
コンソールモードでのリソースキットのアンインストール.	137
手動によるプロセスサーバーのデプロイ解除.	138
手動による Hub Server のデプロイ解除.	138
索引.	139

序文

Multidomain MDM をサポートされるデータベースおよび選択したアプリケーションサーバー環境にインストールおよび設定するには、Informatica[®] *Multidomain MDM* のインストールガイドの指示に従ってください。インストールタスクに加えて、このガイドにはインストール前のタスクおよびインストール後のタスクが含まれています。

Informatica のリソース

Informatica は、Informatica Network やその他のオンラインポータルを通じてさまざまな製品リソースを提供しています。リソースを使用して Informatica 製品とソリューションを最大限に活用し、その他の Informatica ユーザーや各分野の専門家から知見を得ることができます。

Informatica Network

Informatica Network は、Informatica ナレッジベースや Informatica グローバルカスタマサポートなど、多くのリソースへの入口です。Informatica Network を利用するには、<https://network.informatica.com> にアクセスしてください。

Informatica Network メンバーは、次のオプションを利用できます。

- ナレッジベースで製品リソースを検索できます。
- 製品の提供情報を表示できます。
- サポートケースを作成して確認できます。
- 最寄りの Informatica ユーザーグループネットワークを検索して、他のユーザーと共同作業を行えます。

Informatica ナレッジベース

Informatica ナレッジベースを使用して、ハウツー記事、ベストプラクティス、よくある質問に対する回答など、製品リソースを見つけることができます。

ナレッジベースを検索するには、<https://search.informatica.com> にアクセスしてください。ナレッジベースに関する質問、コメント、ご意見の連絡先は、Informatica ナレッジベースチーム (KB_Feedback@informatica.com) です。

Informatica マニュアル

Informatica マニュアルポータルでは、最新および最近の製品リリースに関するドキュメントの膨大なライブラリを参照できます。マニュアルポータルを利用するには、<https://docs.informatica.com> にアクセスしてください。

製品マニュアルに関する質問、コメント、ご意見については、Informatica マニュアルチーム (infa_documentation@informatica.com) までご連絡ください。

Informatica 製品可用性マトリックス

製品可用性マトリックス (PAM) には、製品リリースでサポートされるオペレーティングシステム、データベースなどのデータソースおよびターゲットが示されています。Informatica PAM は、<https://network.informatica.com/community/informatica-network/product-availability-matrices> で参照できます。

Informatica Velocity

Informatica Velocity は、Informatica プロフェッショナルサービスが開発したヒントとベストプラクティスのコレクションで、多数のデータ管理プロジェクトから得た実体験に基づいています。Informatica Velocity には、世界中の組織と連携してデータ管理ソリューションを計画、開発、デプロイ、管理する Informatica コンサルタントによる集合知を表しています。

Informatica Velocity リソースには、<http://velocity.informatica.com> からアクセスしてください。Informatica Velocity についての質問、コメント、またはアイデアがある場合は、ips@informatica.com から Informatica プロフェッショナルサービスにお問い合わせください。

Informatica Marketplace

Informatica Marketplace は、お使いの Informatica 製品を拡張したり強化したりするソリューションを検索できるフォーラムです。Marketplace で、Informatica デベロッパーやパートナーからの多数のソリューションを活用すれば、生産性を向上したり、プロジェクトでの実装時間を短縮したりできます。Informatica Marketplace は、<https://marketplace.informatica.com> からアクセスしてください。

Informatica グローバルカスタマサポート

電話または Informatica Network からグローバルサポートセンターに連絡できます。

各地域の Informatica グローバルカスタマサポートの電話番号は、Informatica Web サイト (<https://www.informatica.com/services-and-training/customer-success-services/contact-us.html>) を参照してください。

Informatica Network でオンラインサポートリソースを見つけるには、<https://network.informatica.com> にアクセスし、eSupport オプションを選択します。

第 1 章

インストールの概要

この章では、以下の項目について説明します。

- [Multidomain MDM のインストール, 9 ページ](#)
- [インストールのトポロジ, 11 ページ](#)
- [インストールのタスク, 12 ページ](#)

Multidomain MDM のインストール

Multidomain MDM は、データの信頼性とデータ管理手順を向上するマスターデータ管理ソリューションです。Multidomain MDM は、MDM Hub と呼ばれます。MDM Hub の機能には Hub コンソールを使用してアクセスできます。

MDM Hub は複数のコンポーネントで構成されています。MDM Hub は、グラフィカルモード、コンソールモード、またはサイレントモードでインストールできます。

コアコンポーネント

インストールのコアコンポーネントを次の表に示します。

コンポーネント	説明
MDM Hub マスターデータベース	MDM Hub のビジネスデータの保存および統合を行うスキーマ。ユーザーアカウント、セキュリティ設定、オペレーショナル参照ストアレジストリ、メッセージキュー設定など、MDM Hub 環境設定が含まれている。MDM Hub マスターデータベースから、オペレーショナル参照ストアにアクセスして管理することができる。MDM Hub マスターデータベースのデフォルトの名前は CMX_SYSTEM ですが、カスタム名を使用できます。 複数の MDM Hub マスターデータベースを作成し、それぞれに独自のオペレーショナルリファレンスストアを設定して、同じデータベースインスタンス内に配置できます。
オペレーショナル参照ストア	MDM Hub のビジネスデータの保存および統合を行うスキーマ。マスターデータ、コンテンツメタデータ、マスターデータを処理および管理するためのルールが含まれます。オペレーショナル参照ストアのデータベースは、地理的に異なる場所、組織内の異なる部署、開発環境およびプロダクション環境ごとに個別に設定できる。オペレーショナル参照ストアのデータベースは、複数のサーバーマシンにわたって分散できる。オペレーショナル参照ストアのデフォルト名は CMX_ORP。

コンポーネント	説明
Hub サーバー	アプリケーションサーバーにデプロイする J2EE アプリケーション。Hub サーバーでは MDM Hub の内部に保存されているデータを処理し、MDM Hub を外部アプリケーションと統合します。Hub サーバーは MDM Hub のコアサービスと共通サービスを管理します。
Process サーバー	アプリケーションサーバーにデプロイする J2EE アプリケーション。Process サーバーは、ロード、BVT の再計算、再検証、データクレンジングの実行、一致操作などのバッチジョブを処理します。Process サーバーは、データを標準化および最適化して一致および統合するように設定したクレンジングエンジンとインターフェースで接続します。
プロビジョニングツール	ビジネスエンティティモデルの作成や、Data Director のエンティティ 360 フレームワークの設定を行うツール。ビジネスエンティティモデルを作成したら、設定を MDM Hub にパブリッシュできます。
Informatica ActiveVOS ^(R)	MDM Hub の内部でデータ処理用に必要とされるビジネスプロセス管理 (BPM) ツールです。Informatica ActiveVOS は、データの変更承認プロセスなど、自動的なビジネスプロセスをサポートします。また、ベストバージョンオブトゥールズ (BVT) レコードに追加する前に、マスタデータの変更が必ず確認と承認のプロセスを経るには、Informatica ActiveVOS も使用できます。 ActiveVOS サーバーを Hub サーバーのインストールの一環としてインストールする場合は、ActiveVOS サーバー、ActiveVOS コンソール、および Process Central をインストールします。また、定義済みの MDM のワークフロー、タスク、およびロールもインストールします。
Data Director (IDD)	MDM Hub に格納されているデータを習得および管理するためのユーザーインターフェースです。IDD では、顧客、サプライヤ、従業員などのビジネスエンティティによってデータが整理されます。ビジネスエンティティは、組織にとって意味があるデータグループです。

オプションコンポーネント

次の表に、インストールのオプションコンポーネントを示します。

コンポーネント	説明
リソースキット	MDM Hub をアプリケーションとワークフローに統合するサンプル、アプリケーション、およびユーティリティのセット。インストールするリソースキットコンポーネントは選択できます。
Informatica Platform	ソースデータのクレンジングおよび MDM Hub への転送に使用する Informatica サービスおよび Informatica クライアントから構成される環境です。データのクレンジングには、MDM Hub で利用可能なクレンジング機能の代わりに Informatica Platform を使用できます。 Informatica Platform を Hub サーバーのインストールの一環としてインストールする場合は、データ統合サービス、モデルリポジトリサービス、および Informatica Developer (Developer tool) をインストールします。
Dynamic Data Masking	MDM Hub とデータベースの間で動作して、機密情報への不正アクセスを防止するデータセキュリティツール。Dynamic Data Masking は、データベースに送信された要求をインターセプトし、その要求にデータマスキングルールを適用し、データをマスクしてから MDM Hub に返送します。

コンポーネント	説明
Informatica Data Controls (IDC)	サブジェクト領域データモデルのみに基づいて、Informatica Data Director (IDD) に適用されます。 IDC は、ビジネスユーザーが使用するサードパーティアプリケーションで MDM Hub データを公開する一連のユーザーインターフェースコントロールです。
Zero Downtime (ZDT) モジュール	MDM Hub のアップグレード時に、アプリケーションが MDM Hub のデータにアクセスできるようにするモジュール。ZDT 環境で、データベースを複製します (ソースデータベースとターゲットデータベース)。MDM Hub のアップグレード時に、ZDT モジュールは、ソースデータベースのデータ変更をターゲットデータベースにレプリケートします。 ZDT モジュールを購入するには、Informatica の担当者にお問い合わせください。Zero Downtime 環境のインストールの詳細については、データベース向けの『 <i>Multidomain MDM Zero Downtime</i> のインストールガイド』を参照してください。

インストールのトポロジ

MDM Hub をインストールする前に、インストールのトポロジを決定します。通常、インフラストラクチャの計画者およびマスタデータ管理ソリューションの設計者が実装するトポロジを決定します。

MDM Hub は、開発環境、テスト環境、プロダクション環境など、複数の環境にインストールできます。各タイプの環境の要件および優先順位は独自のものです。したがって、各環境によってインストールのトポロジも変わってきます。

次の表に、使用できる MDM Hub インストールのトポロジを示します。

トポロジ	説明
スタンドアロンのアプリケーションサーバーインスタンス	すべての MDM Hub コンポーネントがスタンドアロンのアプリケーションサーバーインスタンスにインストールされます。
複数のアプリケーションサーバーインスタンス	MDM Hub コンポーネントが複数のアプリケーションサーバーインスタンスにインストールされます。
アプリケーションサーバークラスター	MDM Hub コンポーネントがアプリケーションサーバークラスターにインストールされます。

インストールのトポロジの詳細については、*Multidomain MDM* のインフラストラクチャプランニングガイドを参照してください。

注: MDM Hub 実装のすべてのコンポーネントはバージョンが同じである必要があります。複数のバージョンの MDM Hub が存在する場合は、各バージョンを別々の環境にインストールする必要があります。

インストールのタスク

MDM Hub コンポーネントをインストールするために、インストール前のタスクを完了します。インストールしたら、インストール後のタスクを実行します。

MDM Hub をインストールするには、以下のタスクを実行します。

1. インストール前のタスクを完了します。正常に Hub サーバーおよび Process サーバーのインストーラを実行し、Hub ストアを作成できるようにするには、インストール前のタスクを完了します。
2. MDM Hub マスターデータベースを作成します。Hub サーバーおよびプロセスサーバーをインストールする前に、MDM Hub マスターデータベースを作成します。
MDM Hub ディストリビューションに付属するセットアップスクリプトを使用し、MDM Hub マスターデータベースを作成します。
3. オペレーショナル参照ストアを作成します。インストール前のタスクの完了後は、いつでもオペレーショナル参照ストアを作成できます。
MDM Hub ディストリビューションに付属するセットアップスクリプトを使用し、オペレーショナル参照ストアを作成します。
4. Hub サーバーをインストールします。MDM Hub インストーラを使用して、Hub サーバーをインストールします。
5. プロセスサーバーをインストールします。MDM Hub インストーラを使用して、プロセスサーバーをインストールします。
6. インストール後の設定タスクの実行。データベースの接続をテストします。MDM Hub 機能を確実に使用できるようにするには、Hub サーバーとプロセスサーバーを設定します。
7. オプションで、リソースキットをインストールします。MDM Hub インストーラを使用して、リソースキットをインストールします。

第 2 章

インストール前のタスク

この章では、以下の項目について説明します。

- [インストールの準備, 13 ページ](#)
- [環境の準備, 15 ページ](#)
- [データベース環境の設定, 16 ページ](#)
- [アプリケーションサーバー環境の設定, 22 ページ](#)
- [アプリケーションサーバーの追加設定（オプション）, 28 ページ](#)
- [サイレントインストールのためのプロパティファイルの設定, 33 ページ](#)

インストールの準備

MDM Hub をインストールする前に、インストールの準備をします。

次の表で、インストールの準備作業について説明します。

タスク	説明
リリースノートの確認	最新の <i>Multidomain MDM</i> のリリースノートを読んで、インストールプロセスおよびアップグレードプロセスの最新情報を確認します。 重要: アプリケーションサーバーとデータベースの一部のバージョンには、Multidomain MDM を実行するときの既知の制限事項があります。推奨されたすべての回避策を確実に実行します。
製品可用性マトリックスの確認	製品の要件とサポートされているプラットフォームに関する情報については、製品可用性マトリックス（PAM）を確認します。PAM には https://network.informatica.com/community/informatica-network/product-availability-matrices からアクセスできます。
MDM インフラストラクチャおよびアーキテクチャプランの把握	組織のインフラストラクチャの計画者または MDM ソリューションの設計者から MDM インフラストラクチャとアーキテクチャのプランを入手し、把握します。インフラストラクチャ計画と MDM アーキテクチャの詳細については、 <i>Multidomain MDM</i> のインフラストラクチャプランニングガイドを参照してください。

タスク	説明
インストーラファイルのダウンロードと抽出	<p>インストールファイルを Informatica 電子ソフトウェアダウンロードサイトからマシン上のディレクトリにダウンロードします。圧縮されたファイルを抽出するには、空のフォルダも抽出する抽出ユーティリティを使用します。</p> <p>次のインストールファイルをダウンロードして抽出します。</p> <ul style="list-style-type: none"> - オペレーティングシステムに対応する MDM Hub インストーラ - データベースファイル - オペレーティングシステムに対応する ActiveVOS サーバーインストーラ <p>Informatica プラットフォームをインストールする場合は、次のファイルをダウンロードします。</p> <ul style="list-style-type: none"> - オペレーティングシステムに対応する Informatica プラットフォームのサーバーインストールファイル - Informatica プラットフォームのクライアントインストールファイル
ライセンスキーの確認	<p>ライセンスキーを持っていることを確認します。ライセンスキーは、Informatica から送られてきた電子メールメッセージ内にあります。製品をインストールするユーザーアカウントにアクセス可能なディレクトリに、ライセンスキーファイルをコピーします。</p> <p>ライセンスキーがない場合は、Informatica グローバルカスタマサポートにお問い合わせください。</p>
インストールの記録を保存するディレクトリの作成	<p>検証結果、環境レポート、データベースデバッグログ、ログファイルなどの、インストールファイルのコピーを保存するディレクトリを作成します。</p> <p>例えば、ディレクトリ install_doc を作成します。インストールのトラブルシューティングを行う場合は、インストールの記録を保存するディレクトリのアーカイブファイルを作成し、それを Informatica グローバルカスタマサポートに送信して、分析を求めることができます。</p>

環境の準備

MDM Hub をインストールする前に、インストール環境を準備します。

次の表で、インストール用の環境を準備するために実行するタスクについて説明します。

タスク	説明
最小システム要件の確認	<p>マシンが、MDM Hub インストールのハードウェア要件とソフトウェア要件を満たしていることを確認します。ハードウェア要件は、データ、処理容量、およびビジネスルールによって異なります。</p> <p>MDM Hub をインストールするには、マシンが次の最小システム要件を満たしている必要があります。</p> <ul style="list-style-type: none">- ディスク容量。4.9GB- 開発環境の RAM。4GB <p>MDM Hub コンポーネントのランタイム物理メモリ要件を確認するには、次の計算式を使用してください。</p> <p>Total run-time memory requirement for MDM Hub components = JDK JVM max heap size of the application server + operating system native heap size</p>
Java Development Kit (JDK) のインストール	<p>MDM Hub をインストールするマシンに、サポートされているバージョンの JDK をインストールします。JDK は、MDM Hub インストーラにバンドルされていません。</p> <p>注: アプリケーションサーバーマシンと Hub コンソールを起動するマシンで、同じ Java バージョンを使用します。</p> <p>Informatica Platform をインストールする場合、JDK は Informatica Platform インストーラにバンドルされています。</p>
Visual Studio 2015 の Visual C++再配付可能パッケージのインストール (Windows の場合のみ)	<p>Windows システムでは、Multidomain MDM は名前検索機能と照合機能をサポートするために Visual Studio 2015 の Visual C++再配付可能パッケージが必要です。</p>
環境変数の設定	<p>MDM Hub をインストールするための環境変数を設定します。</p> <p>適切な JDK を使用するには、次の環境変数を、JDK ディレクトリを参照するように設定します。</p> <ul style="list-style-type: none">- JAVA_HOME。必須- PATH。必須- INFA_JDK_HOME。オプション。AIX に Informatica Platform をインストールする場合は必須です。 <p>Oracle 用の正しいロケール動作を設定するには、Oracle ローダー、MDM Hub コンポーネントなどのクライアントアプリケーション環境に、NLS_LANG 環境変数を設定します。</p> <p>NLS_LANG パラメータは以下の形式で指定します。</p> <p>NLS_LANG = <language>_<territory>.<character set></p> <p>注: 中国語、日本語、韓国語、またはアクセント記号付き文字を含んだレコードを保存および検索する場合は、文字セットを UTF-8 に設定します。</p> <p>NLS_LANG の設定の詳細については、Oracle のマニュアルを参照してください。</p> <p>ドメインでのセキュリティが有効になっている Informatica プラットフォームをインストールするには、INFA_TRUSTSTORE 環境変数を、次のディレクトリを参照するように設定します。</p> <p><Informatica platform installation directory>/Client/clients/shared/security</p>

タスク	説明
オペレーティングシステムのロケールを設定	Hub サーバー、MDM Hub マスタデータベース、オペレーショナル参照ストア、Hub コンソールに、同じオペレーティングシステムのロケールを設定します。
UNIX での X Window System の設定	インストーラを UNIX 上でグラフィカルモードで実行する場合は、X Window System を設定します。X Window System は、グラフィック表示サーバーです。X Window System の設定の詳細については、オペレーティングシステムのマニュアルを参照してください。
アプリケーションサーバーのルートようこそページへのアクセス権の無効化	セキュリティを向上するには、アプリケーションサーバーのルートようこそページへのアクセス権を無効にします。手順については、アプリケーションサーバーのマニュアルを参照してください。
セキュアでない TLS 暗号スイートの無効化	<p>セキュリティを向上するには、Multidomain MDM と一緒に使用する Java Runtime Environment で、セキュアでない TLS 暗号スイートを無効にします。</p> <ol style="list-style-type: none"> 1. <code>../jdk<version>/jre/lib/security/java.security</code> ファイルを開きます。 2. <code>jdk.tls.disabledAlgorithms</code> プロパティを見つけ、セキュアでない暗号スイートの次のリストを含めるように値を更新します。 <pre>jdk.tls.disabledAlgorithms = SSLv3, RC4, MD5withRSA, DH keySize < 1024, EC keySize < 224, DES40_CBC, RC4_40, 3DES_EDE_CBC, EDH-RSA-DES-CBC3-SHA, ECDHE-RSA-DES-CBC3-SHA, DES-CBC3-SHA</pre> <p>プロパティの詳細については、JDK のマニュアルを参照してください。</p>

製品要件およびサポート対象プラットフォームの詳細については、次の Informatica Network で Product Availability Matrix を参照してください。

<https://network.informatica.com/community/informatica-network/product-availability-matrices>

データベース環境の設定

MDM Hub マスターデータベースおよびオペレーショナル参照ストアを作成する前に、データベース環境を設定します。

データベース環境を設定するには、次のタスクを実行します。

1. Oracle のインストールおよび設定
2. データベースインスタンスを設定します。
3. データベースの特権と接続を設定します。
4. ActiveVOS^(R)スキーマを作成します。

手順 1. Oracle のインストールと設定

Oracle のマニュアルに記載されている手順に従って、Oracle データベースをインストールして設定できます。

次の表に、Oracle インスタンスを配置する各マシンで実行する必要がある Oracle のインストールおよび設定タスクを示します。:

タスク	説明
Oracle のインストール	サポートされているバージョンの Oracle データベースのインストール 注: Oracle マルチテナント機能を使用する場合は、MDM Hub インストール用にプラグラブルデータベース (PDB) を設定します。
クライアントとユーティリティのインストール	MDM Hub との通信や MDM Hub プロセスの実行をする Oracle クライアントおよびユーティリティソフトウェアをインストールします。 Hub サーバーまたはプロセスサーバーを実行する各マシンに、次のソフトウェアをインストールします。 - Oracle クライアント - SQL*Loader
リサイクルビンの無効化	リサイクルビン (USER_RECYCLEBIN と DBA_RECYCLEBIN) を無効にします。リサイクルビンによって MDM Hub のプロセスが妨害されることがあります。 リサイクルビンをシステムレベルまたはセッションレベルで無効にします。また、リサイクルビン内の既存のオブジェクトを消去します。
初期化パラメータの設定	Oracle 初期化パラメータを init.ora ファイルに設定します。 Oracle 初期化パラメータの詳細については、『 <i>MDM Multidomain Edition パフォーマンスのチューニングの概要</i> 』を参照してください。

Oracle のインストールと設定の詳細については、Oracle のマニュアルを参照してください。

手順 2. データベースとテーブルスペースの作成

Oracle データベースのインストールと設定の後、データベースとテーブルスペースを作成して設定します。

次の表に、データベースを設定するために実行する必要があるタスクを示します。

タスク	説明
データベースの作成	データベースインスタンスごとにデータベースを作成します。
テーブルスペースの作成	MDM Hub データ用のテーブルスペースを作成します。MDM Hub にロードするデータ量に基づいて、デフォルトのテーブルスペースサイズおよびデータファイル数を調整します。 次のテーブルスペースを作成します。 - CMX_DATA。MDM Hub のメタデータおよびユーザーデータが含まれます。 - CMX_INDX。MDM Hub で作成および使用されるインデックスが含まれます。 - CMX_TEMP。MDM Hub 用の一時テーブルが含まれます。 - BPM_DATA。ビジネスプロセス管理用の ActiveVOS データが含まれます。 注: 複数の MDM Hub マスタデータベースを作成する場合は、MDM Hub マスタデータベースごとに一意のテーブルスペースを作成します。

オンプレミスのテーブルスペースの作成

オンプレミスの Oracle を使用する場合、ローカル管理または手動設定を行う永続テーブルスペースとしてテーブルスペースを作成します。

注: 大量のデータロードのデータベース管理を簡素化するには、ビッグファイルのテーブルスペースを作成します。ただし、必要に応じてスモールファイルのテーブルスペースを作成できます。

1. SYSTEM などのデータベース管理者ユーザーとして Oracle にログインします。
2. テーブルスペースを作成します。

次の表に、テーブルスペースを作成するサンプルの SQL 文を示します。

テーブルスペース名	サンプルの SQL 文
CMX_DATA	注: リポジトリマネージャの検証エラーを防止するために、デフォルトのテーブルスペース名を変更しないでください。 CREATE BIGFILE TABLESPACE CMX_DATA NOLOGGING DATAFILE '<Oracle install directory>/CMX_DATA1.dbf' SIZE 2048M REUSE EXTENT MANAGEMENT LOCAL;
CMX_INDX	CREATE BIGFILE TABLESPACE CMX_INDX NOLOGGING DATAFILE '<Oracle install directory>/CMX_INDX1.dbf' SIZE 2048M REUSE EXTENT MANAGEMENT LOCAL;
CMX_TEMP	CREATE BIGFILE TABLESPACE CMX_TEMP NOLOGGING DATAFILE '<Oracle install directory>/CMX_TEMP1.dbf' SIZE 2048M REUSE EXTENT MANAGEMENT LOCAL;
BPM_DATA	CREATE BIGFILE TABLESPACE BPM_DATA NOLOGGING DATAFILE '<Oracle install directory>/BPM_DATA1.dbf' SIZE 2048M REUSE EXTENT MANAGEMENT LOCAL;

Amazon Relational Database Service でのテーブルスペースの作成

Amazon Relational Database Service (RDS) for Oracle を使用する場合、Amazon RDS でテーブルスペースを永続テーブルスペースとして作成します。

注: 大量のデータロードのデータベース管理を簡素化するには、ビッグファイルのテーブルスペースを作成します。ただし、必要に応じてスモールファイルのテーブルスペースを作成できます。

1. データベース管理者ユーザーとして Amazon RDS for Oracle にログインします。
2. テーブルスペースを作成します。

次の表に、テーブルスペースを作成するサンプルの SQL 文を示します。

デフォルトのテーブルスペース名	サンプルの SQL 文
CMX_DATA	<pre>CREATE BIGFILE TABLESPACE CMX_DATA DATAFILE SIZE 2048M AUTOEXTEND ON NEXT 2048M;</pre> <p>リポジトリマネージャの検証エラーを防止するために、デフォルトのテーブルスペース名を変更しないでください。</p>
CMX_INDX	<pre>CREATE BIGFILE TABLESPACE CMX_INDX DATAFILE SIZE 2048M AUTOEXTEND ON NEXT 2048M;</pre>
CMX_TEMP	<pre>CREATE BIGFILE TABLESPACE CMX_TEMP DATAFILE SIZE 2048M AUTOEXTEND ON NEXT 2048M;</pre>

カスタムテーブルスペース名用のデータベース環境の設定

使用するテーブルスペース名が、CMX_INDX または CMX_TEMP のデフォルトテーブルスペース名と異なる場合は、初期化パラメータの DEFERRED_SEGMENT_CREATION を無効にします。このパラメータを無効にするのは、リポジトリマネージャの検証エラーを防止するためです。

- ▶ DEFERRED_SEGMENT_CREATION を無効にするには、次の SQL 文を実行して、データベースを再起動します。
ALTER SYSTEM SET DEFERRED_SEGMENT_CREATION=FALSE SCOPE=BOTH;

手順 3. データベース特権と接続の設定

データベースの特権と接続を設定します。

次の表で、データベースの特権と接続を設定するために実行する必要があるタスクについて説明します。

タスク	説明
データベース管理者ユーザーへの特権の付与	<p>データベース管理者ユーザーを使用して MDM Hub マスタデータベースとオペレーショナル参照ストアを作成する場合は、データベース管理者ユーザーに特権を付与します。ユーザーは、分散トランザクションと DBMS_LOCK のオプションを付与する特権を持っている必要があります。</p> <p>データベース管理者ユーザーに特権を付与するには、オプションを付与する特権を持つユーザーとしてデータベースに接続し、次の SQL 文を実行します。</p> <pre>GRANT SELECT ON sys.pending_trans\$ TO <DBA user> with grant option; GRANT SELECT ON sys.dba_pending_transactions TO <DBA user> with grant option; GRANT SELECT ON sys.dba_2pc_pending TO <DBA user> with grant option; GRANT EXECUTE ON sys.dbms_xa TO <DBA user> with grant option; GRANT EXECUTE ON sys.dbms_lock TO <DBA user> with grant option;</pre>
Oracle TNS 名の追加	<p>Oracle データベースに接続するには、TNS 名前エントリを Hub サーバーおよび Process サーバーマシン上の tnsnames.ora ファイルに追加します。</p> <p>TNS 名前エントリを追加するには、次の構文を使用します。</p> <pre><TNS NAME> = (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(Host = <Oracle server host name>)(Port = <Oracle server port>))) (CONNECT_DATA = (SERVICE_NAME = <Oracle SID>)))</pre> <p>注: Hub サーバーおよび Process サーバーマシン上の TNS 名は同じでなければなりません。</p>
データベース接続のテスト	<p>Hub サーバーまたは Process サーバーを実行する各マシンから、データベースへの接続をテストします。</p> <p>SQL*Plus で、次の SQL 文の構文を使用します。</p> <pre>sqlplus <user name>/<password>@<TNS Name></pre>

手順 4. ActiveVOS スキーマの作成

ActiveVOS をインストールするには、ActiveVOS スキーマを作成する必要があります。スキーマを作成するには、create_bpm スクリプトを実行します。

複数の MDM Hub マスタデータベースを作成する場合は、MDM Hub マスタデータベースごとに ActiveVOS スキーマを作成します。

注: Oracle マルチテナント機能を使用する場合は、プラグブルデータベース (PDB) に ActiveVOS スキーマを作成します。

1. コマンドプロンプトを開き、次のディレクトリに移動します。
 <MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/database/bin
2. 次のコマンドを実行します。
 UNIX の場合: ./sip_ant.sh create_bpm
 Windows の場合: sip_ant.bat create_bpm

3. 表示されるプロンプトに回答します。

プロンプトでは、デフォルトのテキストが括弧内に表示されます。デフォルト値を使用して次のプロンプトに進むには、**Enter** キーを押します。

プロパティ	説明
Database Type	データベースのタイプ。Oracle データベースの場合、Oracle を指定します。データベースタイプは、MDM Hub マスターデータベースとオペレーショナル参照ストア用に選択したデータベースタイプと同じである必要があります。
Oracle Connection Type	接続タイプ。以下の値を使用する。 - SERVICE。Oracle に接続するサービス名を使用する。 - SID。Oracle に接続する Oracle システム ID を使用する。
ActiveVOS Database Host Name	データベースをホストするマシンの名前。
ActiveVOS Database Port	データベースリスナが使用するポート番号。
データベースサービス名	Oracle サービスの名前。このプロパティは、選択した Oracle 接続タイプが SERVICE の場合に必要になります。
Oracle Net Connect Identifier (TNS Name)	Oracle TNS 名。
Database SID	Oracle システム ID の名前。このプロパティは、選択した Oracle 接続タイプが SID の場合に必要になります。
DBA User Name	データベース管理者ユーザーのユーザー名。
DBA Password	管理者ユーザーのパスワード。
ActiveVOS User Name	ActiveVOS サーバーの管理者ユーザーの名前。
ActiveVOS User Password	管理者ユーザーのパスワード。
ActiveVOS User Tablespace	MDM ワークフローに関連するレコードを格納するテーブルスペースの名前。
ActiveVOS User Temp Tablespace	一時テーブルスペースの名前。

4. スキーマを作成したら、次のディレクトリにある sip_ant.log ファイルを確認します。

<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/database/bin

sip_ant.log ファイルには、ActiveVOS データベースのスキーマを作成するために sip_ant スクリプトを実行するときに発生するすべてのエラーが記録されます。

アプリケーションサーバー環境の設定

MDM Hub は、WebSphere クラスタ環境またはスタンドアロンの WebSphere インスタンスにインストールできます。WebSphere のマニュアルで説明されている手順に従って、WebSphere をインストールして設定します。

WebSphere インストールディレクトリのパスにスペースを含めないようにします。

注: データベースサーバーと同じタイムゾーンのアプリケーションサーバーをインストールします。

次の表で、インストール前に設定するプロパティ、値、およびプロパティを設定する場所の簡単な説明を示します。

カスタムプロパティ	値	説明
com.ibm.ws.scripting.echoparams	false	このプロパティは、次のディレクトリにある <code>wsadmin.properties</code> ファイルで設定します: <code><WebSphere のインストールディレクトリ>\WebSphere\AppServer\profiles\<プロファイル名>\properties</code> ActiveVOS データベースのパスワードが <code>patchinstallSetup.log</code> ファイルのスクリプトテキストに表示されないようにするには、この値を <code>false</code> に設定します。デフォルトでは <code>true</code> です。

Java 仮想マシンの設定

Java 仮想マシン (JVM) を設定するには、`JAVA_OPTIONS` 環境変数を使用して、Java のオプションを設定します。Java のオプションを編集または追加した場合は、JVM を再起動します。

WebSphere クラスタ環境を使用する場合は、次のクラスタコンポーネントで Java のオプションを設定します。

- サーバー。クラスタ内の各サーバーで、必要なすべての Java のオプションを設定します。
- デプロイメントマネージャ。必要なすべての Java のオプションを設定します。
- ノードエージェント。Java の `-Xmx` オプションと `-Xms` オプションを使用して、ヒープサイズのみを設定します。

次の表で、Java のオプションの設定を説明します。

Java のオプション	説明
<code>-server</code>	起動速度は遅くなりますが、後続の操作は高速になります。
<code>-Djava.net.preferIPv4Stack</code>	Java で Internet Protocol バージョン 4 (IPv4) を使用するかどうかを指定します。オペレーティングシステムで Internet Protocol バージョン 6 (IPv6) を使用する場合は、 <code>true</code> に設定します。
<code>-Dcom.ibm.crypto.provider.DoRSATypeChecking</code>	Java が秘密キーおよび復号化と公開キーによるデータの RSA 型暗号化を許可するかどうかを指定します。MDM Hub インストーラがライセンス証明書を読み取り、パスワードハッシュが MDM Hub で機能するために必要です。 <code>false</code> に設定します。 <code>-Dcom.ibm.crypto.provider.DoRSATypeChecking</code> を <code>false</code> に設定しないと、Hub サーバーが起動せず、ライセンスエラーが発生する可能性があります。

Java のオプション	説明
-Djgroups.bind_addr	JGroup がメッセージの送受信を行う必要があるインターフェース。 マルチノード環境またはクラスタ環境が必要です。各ノードが専用のネットワークインターフェースにバインドされていることを確認します。
-De360.mdm.host -De360.mdm.port -De360.connection.channel	アプリケーションサーバーの通信プロトコル、ホストおよびポート。 2809 以外の Bootstrap ポートに MDM Hub アプリケーションをデプロイするには、次の Java オプションを設定します。 <ul style="list-style-type: none"> -De360.connection.channel。使用する通信プロトコルに設定します。有効な値は HTTP および HTTPS です。デフォルトは HTTP です。 -De360.mdm.host WebSphere ホストの IP アドレスを設定します。 環境で HTTPS 通信プロトコルを使用していて、セキュリティ証明書が完全修飾ドメイン名 (FQDN) に発行されている場合、FQDN に設定します。 <ul style="list-style-type: none"> -De360.mdm.port 2809 ではなく、設定済みの WebSphere Bootstrap ポートを設定します。 このパラメータを設定しない場合、エンティティ 360 フレームワークに基づく Data Director の画面が期待どおりに動作しない可能性があります。
-Didd.mdm.host -Didd.mdm.port -Didd.protocol	サブジェクト領域を使用する Data Director に必要です。サブジェクト領域を使用する Data Director をデプロイする場合、次の Java オプションを設定します。 <ul style="list-style-type: none"> -Didd.mdm.host。アプリケーションサーバーホストのホスト名または IP アドレスに設定します。 -Didd.mdm.port。デフォルト以外の HTTP または HTTPS リスナポートを使用してデプロイするのに必要です。このプロパティは、サーバーの初期化中にサブジェクト領域のアプリケーションをデプロイするために使用されます。アプリケーション用の JVM で使用する HTTP または HTTPS リスナポートを指定します。デフォルトは 8080 です。 -Didd.protocol。サーバーの初期化中にサブジェクト領域のアプリケーションをデプロイするために使用される必須プロパティ。使用する通信プロトコルが HTTP か HTTPS かを指定します。デフォルトは HTTP です。
-DFrameworksLogConfigurationPath	log4j.xml ファイルのパス。
-Doracle.jdbc.J2EE13Compliant	Oracle ドライバのシステム変数が J2EE に完全に準拠するかどうかを指定します。true に設定します。 このパラメータを true に設定しない場合、Java データベース接続 (JDBC) の問題が発生する可能性があります。
-Dmdm.node.groupid	MDM Hub 実装で Java 仮想マシンのグループ ID を指定します。Hub サーバーとプロセスサーバーを論理的にグループ化する場合にのみ必要です。

Java のオプション	説明
-Dfile.encoding -Dclient.encoding.override	Informatica Data Director を使用する場合、および REST API を使用してレコードを検索する場合に必要です。 UTF-8 文字を含むレコードを確実に検索および保存できるようにするには、両方の Java のオプションを UTF-8 に設定します。
-Dstricttransportsecurity.flag	HTTP 要求を使用して Data Director にアクセスするすべての試行を、代わりに HTTPS 要求に変換する必要があるかどうかを指定します。true に設定します。
-XX:codecachetotal	JIT コードのキャッシュサイズ。MDM Hub 環境のパフォーマンスを高めるには、512m に設定します。
-Xmx	最大 JVM ヒープサイズ。6 GB 以上に設定します。 例えば、-Xmx を 6144m に設定するには、次の JAVA_OPTIONS 環境変数設定を使用します。 <pre>set "JAVA_OPTIONS=-server ... -Xmx6144m"</pre>
-Xms	初期ヒープサイズ 2048m に設定します。
-Xmso	プロセスサーバーの JVM に必要です。オペレーティングシステムスレッドの初期スタックサイズ。システムスレッドのスタックサイズが低いために、アプリケーションサーバーが予期せずシャットダウンしてしまうことがないようにします。4096k に設定します。
-Xss	初期スタックサイズ。2000k に設定します。
XX:+UseCodeCacheFlushing	コードキャッシュがいっぱいになったときに、JVM がコンパイルされたコードを破棄するかどうかを指定します。
-Dtask.pageSize=<タスクの最大数>	各要求に対して取得される ActiveVOS タスクの最大数を指定します。デフォルトは 5000 です。環境に多数のタスクがある場合は、この数値を増やします。

Java 仮想マシンの論理的なグループ化の例

Java 仮想マシン（JVM）をグループ化することにより、Hub サーバーとプロセスサーバーの論理グループを取得できます。Hub サーバーアプリケーションとプロセスサーバーアプリケーションを論理 JVM グループにデプロイすると、Hub サーバーアプリケーションとプロセスサーバーアプリケーション間のすべての通信がグループ内にとどまります。JVM をグループ化するには、MDM Hub 環境の各 JVM にグループ ID を割り当てます。

注: プロセスサーバーのグループ化は、クレンジングと一致プロセスのみに適用されます。論理グループは MDM Hub の内部サーバーキャッシュには適用されません。

次の表は、論理 JVM グループの例を示しています。

JVM グループ	JVM	Hub サーバー	プロセスサーバー
Group1	JVM1	○	はい
Group1	JVM4	-	○
Group2	JVM2	○	はい
Group3	JVM3	-	○

JVM1 では、起動スクリプトに次の Java オプションを追加します。

```
-Dmdm.node.groupid=Group1
```

JVM2 では、起動スクリプトに次の Java オプションを追加します。

```
-Dmdm.node.groupid=Group2
```

JVM3 では、起動スクリプトに次の Java オプションを追加します。

```
-Dmdm.node.groupid=Group3
```

JVM4 では、起動スクリプトに次の Java オプションを追加します。

```
-Dmdm.node.groupid=Group1
```

JVM を設定し、Hub サーバーとプロセスサーバーをデプロイすると、グループには次の特性があります。

- Group1 には 2 つのプロセスサーバーがあり、Group2 には 1 つのプロセスサーバーがあり、Group3 には 1 つのプロセスサーバーがあります。
- すべてのクレンジングとバッチ呼び出しは、検索を除き、自分のグループにとどまります。例えば、Group1 の Hub サーバーでのリアルタイム呼び出しは、Group1 プロセスサーバー（JVM1 および JVM4）にのみ影響します。

Transport Layer Security (TLS) を設定する

重要: 安全な通信を確保するには、Multidomain MDM をデプロイするアプリケーション上で、TLS 1.0 を無効にし、TLS 1.2 を有効にします。

TLS の設定方法の詳細については、アプリケーションサーバーのマニュアルを参照してください。

セキュアな HTTP 応答ヘッダー

アプリケーションサーバー環境を保護するには、セキュアな HTTP 応答ヘッダーを使用します。X-Powered-By や Server などのヘッダーを変更または削除します。

HTTP 応答ヘッダーを変更または削除する方法については、お使いのアプリケーションサーバーのドキュメントを参照してください。

MDM Hub 環境でのパスワードの暗号化

MDM Hub 環境でログファイルに表示されるパスワードなどの機密データを暗号化するには、WebSphere でスク립ティング管理を設定します。

1. 次のディレクトリにある wsadmin.properties ファイルを開きます。

```
<WebSphere installation directory: WebSphere インストールディレクトリ>/profiles/<Application server profile name: アプリケーションサーバーのプロファイル名>/properties
```

2. Java プロパティ `com.ibm.ws.scripting.echoparams` を `false` に設定します。

WebSphere 環境での安全なプロファイルの作成

WebSphere で、Multidomain MDM および Informatica ActiveVOS で使用するためのセキュアプロファイルを設定します。

1. コマンドラインから、次のサンプルコードに示すように安全なプロファイルを作成します。

Windows の場合:

```
<app_server_root>\bin\manageprofiles.bat -create -profileName AppSrv01  
-profilePath <app_server_root>\profiles\AppSrv01  
-templatePath <app_server_root>\profileTemplates\default  
-adminUserName administrator -adminPassword password1 -enableAdminSecurity true
```

UNIX の場合:

```
<app_server_root>/bin/manageprofiles.sh -create -profileName AppSrv01  
-profilePath <app_server_root>/profiles/AppSrv01  
-templatePath <app_server_root>/profileTemplates/default  
-adminUserName administrator -adminPassword password1 -enableAdminSecurity true
```

2. WebSphere コンソールで、セキュリティトランスポートタイプを SSL-Supported に変更します。
 - a. **【セキュリティ】** を展開し、**【グローバル・セキュリティ】** をクリックします。
 - b. **【認証】** の下で **【RMI/IIOP セキュリティ】** を展開し、**【CSlv2 インバウンド通信】** をクリックします。
 - c. CSlv2 Transport Layer の下の **【Transport】** リストから、**【SSL サポート】** を選択します。
 - d. **【適用】** をクリックしてから **【保存】** をクリックします。
 - e. **【CSlv2 アウトバウンド通信】** をクリックします
 - f. CSlv2 Transport Layer の下の **【Transport】** リストから、**【SSL サポート】** を選択します。
 - g. **【適用】** をクリックしてから **【保存】** をクリックします。
3. WebSphere コンソールで、アプリケーションセキュリティが設定されていることを確認します。
 - a. **【セキュリティ】** を展開し、**【グローバル・セキュリティ】** をクリックします。
 - b. **【アプリケーション・セキュリティ】** の下で **【アプリケーション・セキュリティを使用可能にする】** を選択します。
 - c. **【適用】** をクリックしてから **【保存】** をクリックします。
4. 統合リポジトリをセットアップします。
 - a. **【セキュリティ】** を展開し、**【グローバル・セキュリティ】** をクリックします。
 - b. **【ユーザー・アカウント・リポジトリ】** の下の **【使用可能なレルム定義】** リストから、**【統合リポジトリ】** を選択します。
 - c. **【構成】** をクリックします。
 - d. レルムの **【リポジトリ】** の下で、**【組み込みリポジトリを使用する】** をクリックします。
 - e. 管理者ユーザーのパスワードを指定します。
 - f. **【適用】** をクリックしてから **【保存】** をクリックします。
5. WebSphere プロファイルを再起動します。

JAAS アプリケーションログインの設定

WebSphere クラスタ環境でアプリケーションのセキュリティを有効にしている場合、JAAS アプリケーションログインを設定します。ActiveVOS を起動するには、JAAS アプリケーションログイン設定が必要です。

1. WebSphere コンソールを開始します。
2. JAAS ログインモジュールを設定します。
 - a. **【セキュリティ】**を展開し、**【グローバルセキュリティ】** > **【JAAS - アプリケーションログイン】**をクリックします。
 - b. ActiveVOS で指定されたユーザーのアプリケーションログインを追加します。
 - c. エイリアスに ActiveVOSProvidedUser と指定します。
 - d. **【JAAS ログインモジュール】** セクションで、次のモジュールクラスを追加します。
`com.activee.rt.websphere.trustvalidation.AeBasicLoginModule`
 - e. 変更を保存します。
3. **【全般プロパティ】** セクションで、モジュールプロパティを設定します。
 - a. 認証ストラテジを **【必須】** に設定します。
 - b. モジュールのユーザ名とパスワードを入力します。
 - c. 変更を保存します。
4. ActiveVOS ユーザーを作成します。
 - a. **【ユーザーおよびグループ】**を展開し、**【ユーザーの管理】**をクリックします。
 - b. **【作成】**をクリックして、名前とパスワードなど、ユーザーの詳細を入力します。
 - c. **【作成】**をクリックします。
ActiveVOS ユーザーが作成されます。
5. ActiveVOS ユーザーに管理者ロールを追加します。
 - a. **【ユーザーおよびグループ】**で、**【管理者ユーザーロール】** > **【ユーザー】**をクリックします。
 - b. このユーザーに対して **【監視】** ロールを選択します。
 - c. ユーザーを **【使用可能】** リストから **【ロールにマップ】** リストに移動します。
 - d. **【OK】**をクリックします。
6. JAAS アプリケーションログインを使用するように ActiveVOS を設定します。
 - a. ActiveVOS 設定ウィザードを開始します。
 - b. **【グローバルセキュリティ構成】** ページで、**【クラスタ通信で JAAS ログインを使用する】**を選択します。
 - c. **【セキュリティ JAAS ログイン名】** リストで、**【ActiveVOSProvidedUser】**を選択します。
 - d. **【JAAS ログイン構成】** ページで、**【ActiveVOS で指定したユーザー】**を選択します。
 - e. ActiveVOS で指定したユーザーのユーザー名とパスワードを指定します。
 - f. 変更を保存します。

ActiveVOS コンソール管理ユーザーの作成

ActiveVOS を使用する場合、アプリケーションサーバーコンテナで abAdmin ロールを使用して ActiveVOS コンソールの管理者ユーザーを作成します。管理者ユーザーを作成しない場合は、Hub サーバーのデプロイメント

に失敗します。Hub サーバーインストーラから ActiveVOS コンソールの管理者ユーザーの資格情報を入力するように要求されたら、ActiveVOS コンソールの管理者ユーザー名およびパスワードを使用します。

- ▶ WebSphere コンソールにログインして、ActiveVOS コンソール管理者ユーザーを作成します。

注: ActiveVOS コンソールのユーザーは、インストール後またはアップグレード後のプロセスで postInstallSetup または patchInstallSetup スクリプトを実行するときに abAdmin ロールにマップされます。

MDM Hub デプロイメント用の SOAP 要求タイムアウトの設定

MDM Hub コンポーネントのデプロイメントがタイムアウトにならないように、SOAP 要求タイムアウトのプロパティを設定します。インストールが成功したら、このプロパティをデフォルト値にリセットします。

1. 次のディレクトリにある soap.client.props ファイルを開きます。
 <WebSphere installation directory>: WebSphere インストールディレクトリ>/profiles/<Application server profile name>: アプリケーションサーバーのプロファイル名>/properties
2. com.ibm.SOAP.requestTimeout プロパティを 1800 以上に設定します。

アプリケーションサーバーの追加設定（オプション）

MDM Hub 環境の要件に基づいて、WebSphere の追加設定を行います。

次の表に、実行可能な設定を示します。

設定	説明
スタンドアロンプロセスサーバーインスタンス用の WebSphere の設定	次のシナリオでは、スタンドアロンプロセスサーバーインスタンス用に WebSphere を設定する必要があります。 <ul style="list-style-type: none">- Hub サーバーがインストールされていない WebSphere インスタンスに Process サーバーインスタンスをインストールする必要がある。- 複数のスタンドアロン Process サーバーインスタンスをインストールする。
複数の MDM Hub マスタデータベース用の WebSphere の設定	複数の MDM Hub マスタデータベースインスタンスを設定する場合に必要です。
HTTPS プロトコルの設定	MDM Hub 通信用に HTTPS プロトコルを設定する場合に必要です。
Informatica Data Director 用の WebSphere の設定	Informatica Data Director を使用する場合に必要です。

スタンドアロンプロセスサーバーインスタンス用の WebSphere の設定

複数のスタンドアロン Process サーバーインスタンスをインストールする場合、適切なデータソースを使用するように WebSphere を設定します。また、Hub サーバーをインストールしていない WebSphere インスタンスに Process サーバーインスタンスをインストールする場合、データソースを設定します。

以下のタスクを実行して、適切なデータソースを使用するように WebSphere を設定します。

1. JDBC ドライバをインストールします。
2. MDM Hub マスタデータベースのデータソースを作成します。

3. オペレーショナル参照ストアのデータソースを作成します。

手順 1.JDBC ドライバのインストール

MDM Hub マスタデータベースとオペレーショナル参照ストア（ORS）のデータソースを作成する前に、JDBC ドライバをインストールします。

サポートされているバージョンの JDBC ドライバの取得方法については、Oracle にお問い合わせください。

- ▶ JDBC ドライバを次のディレクトリにコピーします。

<WebSphere installation directory>/lib

手順 2.MDM Hub マスタデータベースのデータソースの作成

プロセスサーバマシンに JDBC ドライバをインストールした後で、MDM Hub マスタデータベース用のデータソースを作成します。

1. WebSphere Application Server の管理コンソールを起動します。
2. ドライバライブラリの場所を指定します。
 - a. コンソールナビゲーションツリーで、**[環境]** を展開します。
 - b. **[WebSphere 変数]** リンクをクリックします。
 - c. JDBC 変数が次の JDBC ドライバディレクトリを参照するように更新します。
<WebSphere installation directory: WebSphere インストールディレクトリ>/lib
3. MDM Hub マスタデータベースのデータソースで使用するセキュリティアカウントを作成します。
 - a. コンソールのナビゲーションツリーで、**[セキュリティ]** を展開します。
 - b. **[セキュアな管理、アプリケーション、インフラストラクチャ]** リンクをクリックします。
 - c. **[認証]** で **[Java 認証および承認サービス]** を展開し、**[J2C 認証データ]** をクリックします。
 - d. **[新規]** をクリックし、以下のプロパティを指定します。

プロパティ	説明
エイリアス	MDM Hub マスタデータベースの名前。
ユーザー ID	MDM Hub マスタデータベースに接続するためのユーザー名。
パスワード	MDM Hub マスタデータベースにアクセスするためのパスワード。

- e. **[OK]** をクリックします。
4. JDBC プロバイダを作成します。
 - a. **[リソース]** > **[JDBC]** を展開し、**[JDBC プロバイダ]** をクリックします。
[JDBC プロバイダ] ページが表示されます。
 - b. データソースを使用するアプリケーションのスコープを選択します。

- c. **【新規】** をクリックし、以下のプロパティを指定します。

プロパティ	説明
データベースタイプ	データベースの種類。[Oracle] を選択します。
プロバイダタイプ	JDBC プロバイダのタイプ。[Oracle JDBC ドライバ] を選択します。
実装タイプ	データソース実装のタイプ。[XA データソース] を選択します。
名前	JDBC プロバイダの名前。名前を「Informatica MDM Oracle JDBC Provider (XA)」に変更します。

- d. **【次へ】** をクリックし、**【完了】** をクリックします。

5. MDM Hub マスタデータベースのデータソースを作成します。

- a. 作成した JDBC プロバイダをクリックします。
【設定】 ページが表示されます。
- b. **【追加プロパティ】** の下で、**【データソース】** をクリックします。
【データソース】 ページが表示されます。
- c. **【新規】** をクリックします。
- d. 以下のデータソースのプロパティを指定してください。

プロパティ	説明
名前	データソース名。「MDM Master Data Source」と指定します。
JNDI 名	JDBC データソースが関連付けられる場所への JNDI パス。jdbc/siperian-cmx_system-ds と指定します。 注: JNDI 名は小文字で指定する必要があります。
管理されるコンポーネントの認証エイリアス	マスタデータベースデータソースの認証エイリアス。[<host name>/cmx_system] を選択します。

- e. **【次へ】** をクリックし、**【完了】** をクリックします。

手順 3. オペレーショナル参照ストアのデータソースの作成

プロセスサーバーマシンに JDBC ドライバをインストールした後で、各オペレーショナル参照ストア用のデータソースを作成します。

- WebSphere Application Server の管理コンソールを起動します。
- ドライバライブラリの場所を指定します。
 - コンソールナビゲーションツリーで、**【環境】** を展開します。
 - 【WebSphere 変数】** リンクをクリックします。
 - JDBC 変数が次の JDBC ドライバディレクトリを参照するように更新します。
 <WebSphere installation directory: WebSphere インストールディレクトリ>/lib

3. オペレーショナル参照ストアで使用するセキュリティアカウントを作成します。
 - a. コンソールのナビゲーションツリーで、**【セキュリティ】**を展開します。
 - b. **【セキュアな管理、アプリケーション、インフラストラクチャ】**リンクをクリックします。
 - c. **【認証】**で**【Java 認証および承認サービス】**を展開し、**【J2C 認証データ】**をクリックします。
 - d. **【新規】**をクリックし、以下のプロパティを設定します。

プロパティ	説明
エイリアス	オペレーショナル参照ストアの名前。
ユーザー ID	オペレーショナル参照ストアに接続するためのユーザー名。
パスワード	オペレーショナル参照ストアにアクセスするためのパスワード。

- e. **【OK】**をクリックします。
4. JDBC プロバイダを作成します。
 - a. **【リソース】** > **【JDBC】**を展開し、**【JDBC プロバイダ】**をクリックします。
【JDBC プロバイダ】ページが表示されます。
 - b. データソースを使用するアプリケーションのスコープを選択します。
 - c. **【新規】**をクリックし、以下のプロパティを指定します。

プロパティ	説明
データベースタイプ	データベースの種類。 【Oracle】 を選択します。
プロバイダタイプ	JDBC プロバイダのタイプ。 【Oracle JDBC ドライバ】 を選択します。
実装タイプ	データソース実装のタイプ。 【XA データソース】 を選択します。
名前	JDBC プロバイダの名前。名前を「Informatica MDM Oracle JDBC Provider (XA)」に変更します。

- d. **【次へ】**をクリックし、**【完了】**をクリックします。
5. オペレーショナル参照ストアのデータソースを作成します。
 - a. 作成した JDBC プロバイダをクリックします。
【設定】ページが表示されます。
 - b. **【追加プロパティ】**の下で、**【データソース】**をクリックします。
【データソース】ページが表示されます。
 - c. **【新規】**をクリックします。

- d. 以下のデータソースのプロパティを指定してください。

プロパティ	説明
名前	データソース名。「MDM ORS Data Source」と指定します。
JNDI 名	JDBC データソースが関連付けられる場所への JNDI パス。jdbc/siperian- <Oracle host name>-<Oracle SID>-<Operational Reference Store name>-ds を指 定します。 注: JNDI 名は小文字で指定する必要があります。
管理されるコンポーネントの認証エイリアス	マスターデータベースデータソースの認証エイリアス。<host name>/ <Operational Reference Store name>を選択します。

- e. **[次へ]** をクリックし、**[完了]** をクリックします。

複数の MDM Hub マスターデータベース用の WebSphere の設定

複数の MDM Hub マスターデータベースインスタンスを設定する場合、MDM Hub マスターデータベースインスタンスと同数の WebSphere プロファイルを設定します。各 MDM Hub マスターデータベースインスタンスには、独自の MDM Hub インスタンスが必要です。そのため、同数の WebSphere プロファイルを作成して各 MDM Hub インスタンスを個別の WebSphere プロファイルでデプロイします。

HTTPS プロトコルの設定

Hub サーバー、プロセスサーバー、ActiveVOS などの、MDM Hub コンポーネント間の通信に HTTPS プロトコルを使用するには、WebSphere アプリケーションサーバーで HTTPS プロトコルを設定します。

1. SSL 対応の WebSphere ポートを作成します。
2. 自己署名証明書を許可するように WebSphere を設定します。
3. 次のカスタム JVM プロパティを設定します。

カスタム JVM プロパティ	説明
javax.net.ssl.keyStore	キーストアの場所。
javax.net.ssl.keyStorePassword	キーストアのパスワード。
javax.net.ssl.keyStoreType	キーストアのタイプ。
javax.net.ssl.trustStore	トラストストアの場所。
javax.net.ssl.trustStorePassword	トラストストアのパスワード。
javax.net.ssl.trustStoreType	トラストストアのタイプ。

HTTPS プロトコルの設定の詳細については、WebSphere のマニュアルを参照してください。

Informatica Data Director 用の WebSphere の設定

Data Director を使用する場合、WebSphere を設定し、変更が反映されるように WebSphere を再起動します。

以下の設定を確実に実行します。

- Web コンテナのカスタムプロパティを設定します。
WebSphere コンソールを使用して `com.ibm.ws.webcontainer.invokerequestlistenerforfilter` を `true` に設定します。Web コンテナのカスタムプロパティの設定手順については、WebSphere のマニュアルを参照してください。
- タスクの管理をサポートするには、タイムアウトプロパティの値を 2 倍に増やします。
WebSphere コンソールを使用して以下のタスクを実行します。
 1. **WebSphere コンソールの [サーバー] > [サーバータイプ] > [WebSphere アプリケーションサーバー] > <target server name>** に移動します。
 2. **[コンテナサービス]** カテゴリで、**[トランザクションサービス]** をクリックし、タイムアウトプロパティの値を増やします。

サイレントインストールのためのプロパティファイルの設定

サイレントモードでユーザー操作を省いて Hub サーバーとプロセスサーバーをインストールする場合は、インストールプロパティファイルを設定します。複数回のインストールや、マシクラスタへのインストールが必要なときは、サイレントモードでの実行をお勧めします。サイレントインストールでは、進行状況や失敗に関するメッセージが表示されません。

インストーラによりサイレントインストールプロパティファイルが読み込まれ、インストールオプションが決定されます。プロパティファイルが正しく設定されていることを確認してください。サイレントインストールプロセスは、設定に誤りがあっても正常に完了する場合があります。

設定可能なサイレントインストールプロパティファイルを次に示します。

- Informatica プラットフォーム。MDM Hub のインストールの一環として Informatica プラットフォームをインストールする場合に必要です。
- Hub サーバー。Hub サーバーをサイレントモードでインストールする場合に必要です。
- プロセスサーバー。プロセスサーバーをサイレントモードでインストールする場合に必要です。

注: サイレントインストールプロパティファイルを手動で設定しない場合は、インストール時に `-r` コマンドラインオプションを使用するとサイレントインストールプロパティファイルを生成できます。

Informatica プラットフォームのプロパティファイルの設定

MDM Hub インストールの一環として Informatica プラットフォームをインストールするには、Informatica プラットフォームのプロパティファイルを設定します。プロパティファイルにインストールのオプションを指定して、このファイルを「`SilentInput.properties`」という名前で作成します。

1. `<Distribution directory>/Informatica platform` ディレクトリで、`SilentInput.properties` ファイルを見つけます。
2. ファイルのバックアップコピーを作成します。
3. テキストエディタで `SilentInput.properties` ファイルを開きます。

4. インストールパラメータの値を設定し、ファイルを保存します。

Hub サーバーのプロパティファイルの設定

Hub サーバーをサイレントモードでインストールする場合、Hub サーバーのプロパティファイルを設定します。プロパティファイルにインストールのオプションを指定して、このファイルを新しい名前で保存します。

1. `silentInstallServer_sample.properties` ファイルを `/silent_install/mrmserver` ディレクトリで見つけます。
2. `silentInstallServer_sample.properties` ファイルのバックアップコピーを作成します。
3. このファイルをテキストエディタで開き、インストールパラメータの値を設定します。
4. プロパティファイルを `silentInstallServer.properties` などの新しい名前で保存します。

プロセスサーバーのプロパティファイルの設定

プロセスサーバーをサイレントモードでインストールする場合、Hub サーバーのプロパティファイルを設定します。プロパティファイルにインストールのオプションを指定して、このファイルを新しい名前で保存します。

1. `silentInstallCleanse_sample.properties` ファイルを `/silent_install/mrmcleanse` ディレクトリで見つけます。
2. `silentInstallCleanse_sample.properties` ファイルのバックアップコピーを作成します。
3. このファイルをテキストエディタで開き、インストールパラメータの値を設定します。
4. プロパティファイルに `silentInstallCleanse.properties` などの名前を付けて保存します。

第 3 章

Hub ストアのインストール

この章では、以下の項目について説明します。

- [MDM Hub マスターデータベースの作成, 35 ページ](#)
- [オペレーショナル参照ストアの作成, 37 ページ](#)
- [MDM Hub マスターデータベースへのメタデータのインポート, 39 ページ](#)
- [オペレーショナル参照ストアへのメタデータのインポート, 40 ページ](#)

MDM Hub マスターデータベースの作成

Oracle をインストールした後に、MDM Hub マスターデータベースを作成します。複数の MDM Hub マスターデータベースを作成する場合は、各 MDM Hub マスターデータベースを異なるスキーマに作成していることを確認してください。MDM Hub マスターデータベースのデフォルト名は CMX_SYSTEM ですが、カスタム名を使用できます。

注: 配布ディレクトリ内のフォルダ名を変更すると、メタデータのインポートが失敗します。

Oracle マルチテナント機能を使用する場合は、プラグブルデータベース (PDB) に MDM Hub マスターデータベースを作成します。

1. コマンドプロンプトを開き、次のディレクトリに移動します。
<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/database/bin
2. MDM Hub マスターデータベースを作成するには、次のコマンドを実行します。
UNIX の場合: ./sip_ant.sh create_system
Windows の場合: sip_ant.bat create_system
3. 以下の表に説明する内容に従ってプロンプトに回答します。

注: プロンプトでは、デフォルトのテキストが括弧内に表示されます。デフォルト値を使用して次のプロンプトに進むには、**Enter** キーを押します。

プロンプト	説明
データベースタイプを入力 (ORACLE、MSSQL、DB2)	データベースタイプ。ORACLE を指定。
Oracle 接続タイプ (SERVICE、SID) を入力。[SERVICE]	接続タイプ。以下の値を使用する。 - SERVICE。Oracle に接続するサービス名を使用する。 - SID。Oracle に接続する Oracle システム ID を使用する。 デフォルトは SERVICE。
マスタユーザー名を入力 [cmx_system]	MDM Hub マスタデータベースにアクセスするためのユーザー名。デフォルトは cmx_system です。
データベースホスト名を入力 [localhost]	データベースを実行するホストの名前。デフォルトは localhost。 重要: クラスタ環境では、キャッシュの問題を回避するために、絶対ホスト名または IP アドレスを指定します。
データベースポート番号を入力します ([1521])	データベースリスナーが使用するポート番号。デフォルトは 1521。
データベースサービス名を入力 [orcl]	Oracle サービスの名前。このプロンプトは、選択した Oracle 接続タイプが SERVICE の場合に表示される。デフォルトは orcl。
データベース SID を入力。[orcl]	Oracle システム ID の名前。このプロンプトは、選択した Oracle 接続タイプが SID の場合に表示される。
Oracle Net の接続 ID (TNS 名) を入力。[orcl]	Oracle データベースに接続するために使用される TNS 名。デフォルトは orcl。
マスタデータベースの接続 URL: "jdbc:oracle:thin:@// <host_name>:<port>/<service_name>"。	Oracle 接続タイプ SERVICE の接続 URL。
マスタデータベースのユーザーパスワードを入力	MDM Hub マスタデータベースにアクセスするためのパスワード。
リストからロケール名を入力します (de、en_US、fr、ja、ko、zh_CN) [en_US]	オペレーティングシステムのロケール。デフォルトは en_US。
DBA ユーザー名を入力 [SYS]	管理者ユーザーの名前。デフォルトは SYS。
DBA のパスワードを入力	管理者ユーザーのパスワード。
MDM インデックステーブルスペースの名前を入力 [CMX_INDX]	MDM Hub マスタデータベース用のインデックスコンポーネントが含まれているテーブルスペースの名前。デフォルトは CMX_INDX。

プロンプト	説明
MDM 一時テーブルスペース名の入力 (Oracle 一時テーブルスペースではない) [CMX_TEMP]	MDM Hub マスターデータベース用の一時コンポーネントが含まれているテーブルスペースの名前。デフォルトは CMX_TEMP。
Oracle 一時テーブルスペース名を入力 [TEMP]	Oracle 一時テーブルスペースの名前。デフォルトは TEMP。

4. MDM Hub マスターデータベースが正常に作成されていることを確認するには、次のディレクトリの sip_ant.log を確認します。
- <MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/database/bin
- sip_ant.log ファイルには、MDM Hub マスターデータベースを作成するために sip_ant スクリプトを実行するときに発生するすべてのエラーが記録されます。

オペレーショナル参照ストアの作成

インストール前のタスクを完了したら、オペレーショナル参照ストア（ORS）を作成します。ORS のデフォルト名は CMX_ORS です。

注: 配布ディレクトリ内のフォルダ名を変更すると、メタデータのインポートが失敗します。

Oracle マルチテナント機能を使用する場合は、プラガブルデータベース（PDB）に ORS を作成します。

1. コマンドプロンプトを開き、次のディレクトリに移動します。
<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/database/bin
2. ORS を作成するには、次のコマンドを実行します。
UNIX の場合: ./sip_ant.sh create_ors
Windows の場合: sip_ant.bat create_ors

3. 以下の表に説明する内容に従ってプロンプトに回答します。

注: プロンプトでは、デフォルトのテキストが括弧内に表示されます。デフォルト値を使用して次のプロンプトに進むには、**Enter** キーを押します。

プロンプト	説明
データベースタイプを入力 (ORACLE、MSSQL、DB2)	データベースタイプ。ORACLE を指定。
Oracle 接続タイプ (SERVICE、SID) を入力。 [SERVICE]	接続タイプ。以下の値を使用する。 - SERVICE。Oracle に接続するサービス名を使用する。 - SID。Oracle に接続する Oracle システム ID を使用する。 デフォルトは SERVICE。
オペレーショナル参照ストアスキーマのホスト名を入力 [localhost]	データベースを実行しているホストの名前。デフォルトは localhost。

プロンプト	説明
オペレーショナル参照ストアスキーマのポート番号を入力 [1521]	データベースリスナが使用するポート番号。デフォルトは 1521。
オペレーショナル参照ストアのデータベースサービス名を入力 [orcl]	Oracle サービスの名前。このプロンプトは、選択した Oracle 接続タイプが SERVICE の場合に表示される。
Oracle Net の接続 ID (TNS 名) を入力。[orcl]	Oracle TNS 名。デフォルトは orcl。
ORS DB の接続 URL: "jdbc:oracle:thin:@//<host_name>:<port>/<service_name>"。 接続 URL を変更しますか (y/n) [n]	Oracle 接続タイプ SERVICE の接続 URL。デフォルトの接続 URL を変更する場合は、y を入力する。デフォルトの接続 URL を使用する場合は、n を入力する。
データベース SID を入力。[orcl]	Oracle システム ID の名前。このプロンプトは、選択した Oracle 接続タイプが SID の場合に表示される。
オペレーショナル参照ストアのデータベースユーザー名を入力[cmx_ors]	オペレーショナル参照ストアにアクセスするためのユーザー名。デフォルトは cmx_ors。
オペレーショナル参照ストアのデータベースのユーザーパスワードを入力	オペレーショナル参照ストアにアクセスするためのパスワード。
リストからロケール名を入力します (de、en_US、fr、ja、ko、zh_CN) [en_US]	オペレーティングシステムのロケール。
DBA ユーザー名を入力 [SYS]	管理者ユーザーの名前。デフォルトは SYS。
DBA のパスワードを入力	管理者ユーザーのパスワード。
MDM インデックステーブルスペースの名前を入力 [CMX_INDX]	オペレーショナル参照ストア用のインデックスコンポーネントが含まれているテーブルスペースの名前。デフォルトは CMX_INDX。
MDM 一時テーブルスペースの名前を入力 [CMX_TEMP]	オペレーショナル参照ストア用の一時コンポーネントが含まれているテーブルスペースの名前。デフォルトは CMX_TEMP。
Oracle 一時テーブルスペース名を入力 [TEMP]	Oracle 一時テーブルスペースの名前。デフォルトは TEMP。

4. ORS が正常に作成されていることを確認するには、次のディレクトリの sip_ant.log を確認します。

<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/database/bin

sip_ant.log ファイルには、ORS を作成するために sip_ant スクリプトを実行するときに発生するすべてのエラーが記録されます。

MDM Hub マスターデータベースへのメタデータのインポート

MDM Hub マスターデータベースを作成したら、初期メタデータを MDM Hub マスターデータベースにインポートします。初期メタデータには、リポジトリテーブルや、MDM Hub が Hub ストアで必要とするその他のオブジェクトがあります。

注: 配布ディレクトリ内のフォルダ名を変更すると、メタデータのインポートが失敗します。

1. コマンドプロンプトを開き、次のディレクトリに移動します。

<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/database/bin

2. 初期メタデータをインポートするには、次のコマンドを実行します。

UNIX の場合: `./sip_ant.sh import_system`

Windows の場合: `sip_ant.bat import_system`

3. 以下の表に説明する内容に従ってプロンプトに回答します。

注: プロンプトでは、デフォルトのテキストが括弧内に表示されます。デフォルト値を使用して次のプロンプトに進むには、**Enter** キーを押します。

プロンプト	説明
データベースタイプを入力 (ORACLE、MSSQL、DB2)	データベースタイプ。ORACLE を指定。
Oracle 接続タイプ (SERVICE、SID) を入力。 [SERVICE]	接続タイプ。以下の値を使用する。 SERVICE Oracle に接続するサービス名を使用する。 SID Oracle に接続する Oracle システム ID を使用する。 デフォルトは SERVICE。
マスタユーザー名を入力 [cmx_system]	MDM Hub マスタデータベースにアクセスするためのユーザー名。デフォルトは cmx_system です。
データベースホスト名を入力 [localhost]	データベースを実行しているホストの名前。
データベースポート番号を入力 [1521]	データベースリスナーが使用するポート番号。デフォルトは 1521。
MDM Hub マスタデータベースのサービス名を入力 [orcl]	Oracle サービスの名前。このプロンプトは、選択した Oracle 接続タイプが SERVICE の場合に表示される。デフォルトは orcl。
データベース SID を入力	Oracle システム ID の名前。このプロンプトは、選択した Oracle 接続タイプが SID の場合に表示される。
Oracle Net の接続 ID (TNS 名) を入力。 [orcl]	Oracle データベースに接続するために使用される TNS 名。デフォルトは orcl。

プロンプト	説明
マスターデータベースの接続 URL: "jdbc:oracle:thin:@//<host_name>:<port>/<service_name>"。	Oracle 接続タイプ SERVICE の接続 URL。
マスターデータベースのユーザーパスワードを入力	MDM Hub マスターデータベースにアクセスするためのパスワード。
リストからロケール名を入力します (de、en_US、fr、ja、ko、zh_CN) [en_US]	オペレーティングシステムのロケール。デフォルトは en_US。

4. 初期メタデータをインポートした後、次のログファイルでエラーがないかどうかを確認します。
 - seed.log。データベースエラーがあります。
seed.log ファイルは次のディレクトリにあります。<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/database/bin/oracle
 - sip_ant.log。ユーザー入力エラーがあります。
sip_ant.log ファイルは次のディレクトリにあります。<distribution directory>/database/bin

オペレーショナル参照ストアへのメタデータのインポート

オペレーショナル参照ストアを作成したら、初期メタデータをオペレーショナル参照ストアにインポートします。初期メタデータには、リポジトリテーブルや、MDM Hub が Hub ストアで必要とするその他のオブジェクトがあります。

注: 配布ディレクトリ内のフォルダ名を変更すると、メタデータのインポートが失敗します。

1. コマンドプロンプトを開き、次のディレクトリに移動します。
<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/database/bin
2. 初期メタデータをインポートするには、次のコマンドを実行します。
UNIX の場合: ./sip_ant.sh import_ors
Windows の場合: sip_ant.bat import_ors
3. 以下の表に説明する内容に従ってプロンプトに回答します。

注: プロンプトでは、デフォルトのテキストが括弧内に表示されます。デフォルト値を使用して次のプロンプトに進むには、**Enter** キーを押します。

プロンプト	説明
データベースタイプを入力 (ORACLE、MSSQL、DB2)	データベースタイプ。ORACLE を指定。
Oracle 接続タイプ (SERVICE、SID) を入力。 [SERVICE]	接続タイプ。以下の値を使用する。 SERVICE Oracle に接続するサービス名を使用する。 SID Oracle に接続する Oracle システム ID を使用する。 デフォルトは SERVICE。
オペレーショナル参照ストアのデータベースホスト名を入力 [localhost]	データベースを実行しているホストの名前。
オペレーショナル参照ストアのデータベースポート番号を入力 ([1521])	データベースリスナーが使用するポート番号。デフォルトは 1521。
オペレーショナル参照ストアのデータベースサービス名を入力	Oracle サービスの名前。このプロンプトは、選択した Oracle 接続タイプが SERVICE の場合に表示される。
Oracle Net の接続 ID (TNS 名) を入力。[orcl]	Oracle データベースに接続するために使用される TNS 名。デフォルトは orcl。
ORS DB の接続 URL: "jdbc:oracle:thin:@//<host_name>:<port>/<service_name>"。 接続 URL を変更しますか (y/n) [n] :	Oracle 接続タイプ SERVICE の接続 URL。デフォルトの接続 URL を変更する場合は、y を入力する。デフォルトの接続 URL を使用する場合は、n を入力する。
データベース SID を入力	Oracle システム ID の名前。このプロンプトは、選択した Oracle 接続タイプが SID の場合に表示される。
オペレーショナル参照ストアのデータベースユーザー名を入力[cmx_ors]	オペレーショナル参照ストアにアクセスするためのユーザー名。デフォルトは cmx_ors。
オペレーショナル参照ストアのデータベースのユーザーパスワードを入力	オペレーショナル参照ストアにアクセスするためのパスワード。
リストからロケール名を入力します (de、en_US、fr、ja、ko、zh_CN) [en_US]	オペレーティングシステムのロケール。デフォルトは en_US。
オペレーショナル参照ストアのタイムラインの粒度の整数コードを入力: 5 (年)、4 (月)、3 (日)、2 (時間)、1 (分)、0 (秒) [3]	使用するタイムラインの単位を指定する。デフォルトは日 (日数)。 注: 設定したタイムラインの粒度は後から変更できない。 タイムラインの詳細については、『 <i>Multidomain MDM の設定ガイド</i> 』を参照。

4. 初期メタデータをインポートした後、次のログファイルでエラーがないかどうかを確認します。
- seed.log。データベースエラーがあります。
seed.log ファイルは次のディレクトリにあります。<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/database/bin/oracle
 - sip_ant.log。ユーザー入力エラーがあります。
sip_ant.log ファイルは次のディレクトリにあります。<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/database/bin

第 4 章

Hub ストアのインストール後のタスク

- [Oracle コンポーネントへのアクセスの検証, 43 ページ](#)

Oracle コンポーネントへのアクセスの検証

Hub ストアが必要な Oracle コンポーネントにアクセスできることを確認します。

Hub ストアは、次の Oracle コンポーネントにアクセスする必要があります。

Oracle Java Virtual Machine
Oracle XML Database
ALL_CONSTRAINTS
ALL_CONS_COLUMNS
ALL_DIRECTORIES
ALL_INDEXES
ALL_IND_COLUMNS
ALL_JOBS (移行で使用)
ALL_TABLES
ALL_TAB_COLUMNS
ALL_VIEWS
DBMS_APPLICATION_INFO
DBMS_JOB
DBMS_OUTPUT
DBMS_STANDARD
DBMS_SQL
DBMS_STATS
DBMS_UTILITY
DUAL
PLITBLM
STANDARD

SYS_STUB_FOR_PURITY_ANALYSIS
USER_CONSTRAINTS
USER_CONS_COLUMNS
USER_EXTERNAL_TABLES
USER_INDEXES
USER_JAVA_POLICY
USER_OBJECTS
USER_SEQUENCES
USER_SOURCE
USER_TABLES
USER_TAB_COLS
USER_TAB_COLUMNS
USER_TRIGGERS
UTL_FILE
V\$NLS_PARAMETERS
V\$VERSION

第 5 章



Hub サーバーのインストール

この章では、以下の項目について説明します。

- [Hub サーバーのインストール, 45 ページ](#)
- [インストーラワークフローの確認, 46 ページ](#)
- [インストールの値の収集, 47 ページ](#)
- [Hub サーバーをインストールウィザードからインストールする, 53 ページ](#)
- [Hub サーバーをコマンドラインからインストールする \(UNIX のみ\) , 54 ページ](#)
- [Hub サーバーをサイレントモードでインストールする, 54 ページ](#)
- [クラスタ内のノードに Hub サーバーをインストールする, 55 ページ](#)

Hub サーバーのインストール

インストールウィザード、サイレントインストールスクリプト、または UNIX システムではコマンドラインスクリプトを使用して、Hub サーバーをインストールできます。インストール前のタスクを完了し、インストーラの開始前に必要な情報を収集した場合、インストールプロセスには約 15 分かかります。

	STOP! インストール前のタスクを完了しましたか? インストーラを実行する前にインストール前のタスクを完了しないと、インストールは失敗します。
	インストール準備状況のチェックリスト <ul style="list-style-type: none"><input type="checkbox"/> MDM 実装プランを作成しました。<input type="checkbox"/> サーバーがシステム要件を満たしていることを確認しました。<input type="checkbox"/> オペレーティングシステムとソフトウェアのバージョンがサポートされていることを確認しました。<input type="checkbox"/> オペレーティングシステムとソフトウェアバージョンの既知の制限事項を確認しました。<input type="checkbox"/> サポートされているバージョンのアプリケーションサーバーをインストールして設定しました。<input type="checkbox"/> サポートされているバージョンのデータベース管理システムをインストールして設定しました。<input type="checkbox"/> 環境に合わせてインストール前の環境設定タスクを実行しました。<input type="checkbox"/> MDM ライセンスファイルをアクセス可能な場所に保存しました。

タスクを実行しなかった場合は、前の章に戻ってタスクを完了するためのヘルプを参照してください。

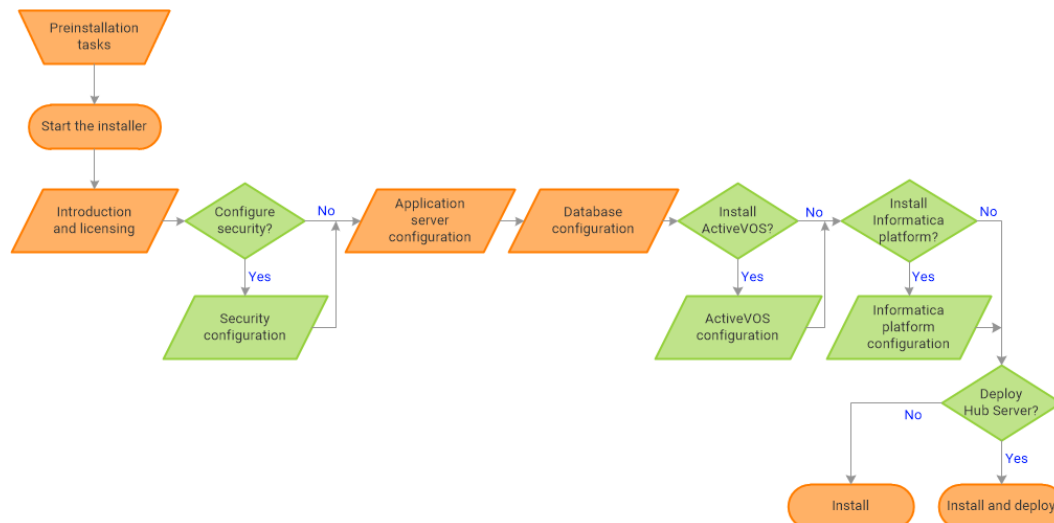
続行する準備ができたなら、次の手順を実行します。

1. インストールワークフローを確認します。
2. インストール中に入力するすべての値を収集します。
3. インストールウィザードまたはコマンドラインを使用するか、サイレントモードでインストールを開始します。

インストーラワークフローの確認

インストールウィザード、コマンドラインプロンプト、サイレントインストールスクリプトのいずれを使用しても、インストーラは同じ順序で手順を実行します。決定ポイントを念頭に置いて、インストールワークフローに従う必要があります。

次の図は、Hub サーバーのインストーラワークフローの手順を示しています。



インストールプロセスの決定ポイントについて確認してください。実装プランを参照して、次の各決定ポイントで実行するパスを把握してください。

1. **セキュリティを設定しますか?** デフォルトのセキュリティを使用するか、パスワードハッシュアルゴリズムと証明書プロバイダを指定して独自のセキュリティを設定します。
2. **ActiveVOS をインストールしますか?** 管理者は Informatica ActiveVOS を使用して確認ワークフローを設定します。承認済みユーザーは、マスタデータが更新される前に、レコードに対して提案されている変更を検証できます。

注: Informatica ActiveVOS がすでにインストールされている場合は、Hub サーバーのインストール中に再度インストールする必要はありません。

3. **Informatica プラットフォームをインストールしますか?** 管理者は、ソースシステムから MDM へのデータのマッピングのために Informatica Platform を使用します。代替手段として、MDM でランディングテーブルを使用します。

注: Informatica Platform がすでにインストールされている場合は、Hub サーバーのインストール中に再度インストールする必要はありません。後で個別に Informatica Platform をインストールすることもできます。

4. **Hub サーバーをデプロイしますか?** インストーラが postInstallSetup スクリプトを実行できるようにします。他の重要なタスクの中で、スクリプトは Hub サーバーをアプリケーションサーバーにデプロイします。または、インストーラの終了後に postInstallSetup スクリプトを実行することもできます。

インストールの値の収集

インストーラを実行する前に、インストールの値を収集します。インストーラはアプリケーションサーバー、データベース管理システム、およびその他のコンポーネントに関する情報を求めるプロンプトを表示します。これらの表を印刷して環境に応じた値を追加することをお勧めします。

アプリケーションサーバー: IBM WebSphere

次の表を使用して、インストールに必要な WebSphere の詳細を収集します。

プロパティ	説明	デフォルト値	サーバーインスタンス1の値
WebSphere インストールディレクトリ	WebSphere がインストールされている場所。	-	
WebSphere セキュリティを有効にしますか?	WebSphere セキュリティが有効な場合、 【はい】 を選択し、ユーザー名およびパスワードを指定します。	いいえ	
サーバー名	Hub サーバーをデプロイする WebSphere アプリケーションサーバーの名前。 クラスタ環境では、クラスタサーバー名、その対応するブートストラップポート情報、および SOAP コネクタポート情報のいずれかを入力します。	-	
ブートストラップポート	指定するサーバーにより使用されるブートストラップポート番号。 ヒント: ポート情報を検索するには、WebSphere 管理コンソールに移動し、 【アプリケーションサーバー】 > 【<server name>】 > 【ポート】 をクリックします。	2809	
SOAP コネクタポート	指定するサーバーにより使用される SOAP コネクタポート番号。 ヒント: ポート情報を検索するには、WebSphere 管理コンソールに移動し、 【アプリケーションサーバー】 > 【<server name>】 > 【ポート】 をクリックします。	8880	
プロファイル名	Hub サーバーをデプロイする WebSphere アプリケーションサーバーを含む WebSphere プロファイルの名前。	-	
ユーザー名	管理特権のある WebSphere ユーザーの名前。	admin	
パスワード	WebSphere 管理者ユーザーのパスワード。	-	

クラスタ環境では、その他の WebSphere インスタンスの詳細を次の表に追加します。

プロパティ	サーバーインスタンス 2 の値	サーバーインスタンス 3 の値	サーバーインスタンス 4 の値
WebSphere インストールディレクトリ			
WebSphere セキュリティを有効にしますか?			
サーバー名			
ブートストラップポート			
SOAP コネクタポート			
プロファイル名			
ユーザー名			
パスワード			

データベース: Oracle データベース

次の表を使用して、インストールに必要な Oracle データベースの詳細を収集します。

プロパティ	説明	デフォルト値	インストールの値
サーバー	Oracle データベースが存在するサーバーのホスト名。	localhost	
ポート	Oracle データベースサーバー上で実行される Oracle リスナの TCP ポート番号。 Hub サーバーはこのポートを使用して Oracle データベースと通信します。	1521	
接続タイプ	Oracle データベースを設定したときに定義した接続のタイプ。	Service	
サービス名	定義した Oracle サービスの名前。	orcl	
Oracle SID	tnsnames.ora ファイルに定義した Oracle システム ID の名前。 注: TNS 名は Hub サーバーと MDM プロセスサーバーマシンの両方で同じである必要があります。	orcl	

プロパティ	説明	デフォルト値	インストールの値
マスタデータベーススキーマ名	MDM マスタデータベースのスキーマ名。	cmx_system	
マスタデータベーススキーマパスワード	MDM マスタデータベースユーザーのパスワード。	-	

セキュリティ: オプション

ハッシュアルゴリズム、証明書プロバイダ、またはその両方を設定することを選択できます。

ハッシュキーおよびハッシュアルゴリズム

ハッシュアルゴリズムを設定するには、siperian-server-hash.jar ファイルに Java 抽象クラス `com.siperian.sam.security.hashing.algorithms.HashAlgorithm` を実装する必要があります。JAR ファイルは、以下のディレクトリに格納されます。<MDM installation directory>/hub/server/lib/hashing

次の表を使用して、siperian-server-hash.jar ファイルへの Java 抽象クラスの実装に使用したハッシュアルゴリズムの詳細を収集します。

プロパティ	説明	デフォルト値	インストールの値
ハッシュキー	オプション。区切り文字なしで最大 32 文字の 16 進数文字のシーケンス。キーサイズは最大 128 ビットです。キーは安全に保管してください。 注意: カスタムハッシュキーが公開または失われた場合、すべてのユーザーパスワードをリセットする必要があります。	-	
ハッシュアルゴリズム	設定されているアルゴリズムのリスト。カスタムアルゴリズムを設定するには、 【その他】 を選択します。	SHA-3	
名前	カスタムハッシュアルゴリズムの名前。	-	
インプリメンテーションファイル	カスタムハッシュアルゴリズムのアーカイブの場所。 注: 圧縮されたファイルには、必要なすべての JAR ファイルおよびサポートするファイルが含まれる必要があります。	-	
標準クラス名	ハッシュアルゴリズム実装の標準クラス名。 例: \$HASHING_CLASS_NAME\$	-	

証明書プロバイダ

カスタム証明書プロバイダを実装するには、siperian-server-pkiutil.jar ファイルの `PKIUtil.java` インタフェースを実装する必要があります。JAR ファイルは、以下のディレクトリに格納されます。<MDM installation directory>/hub/server/lib/pkiutils

次の表を使用して、siperian-server-pkiutil.jar ファイルへの Java 抽象クラスの実装に使用したカスタム証明書プロバイダの詳細を収集します。

プロパティ	説明	デフォルト値	インストールの値
証明書プロバイダ	デフォルトでは、MDM は信頼されたアプリケーションを認証します。カスタムプロバイダを設定するには、 [その他] を選択します。	デフォルト	
インプリメンテーションファイル	カスタム証明書プロバイダの名前。	-	
クラス名	証明書プロバイダ実装のクラス名。	-	

Hub コンソール

Hub コンソールでは、Hub サーバマシンに接続するためにサーバーの詳細が必要です。

次の表を使用して、Hub コンソールが必要なサーバーの詳細を収集します。

プロパティ	説明	デフォルト値	インストールの値
公的にアクセス可能なホスト名	アプリケーションサーバーがバインドするサーバーの IP アドレスまたは公的にアクセス可能なホスト名 (FQDN)。	-	
HTTP ポート	Hub コンソールが使用する必要のあるサーバーの HTTP ポート。	-	

Informatica ActiveVOS: オプション

Informatica ActiveVOS、ActiveVOS Central または ActiveVOS サーバーの詳細については、[Informatica ActiveVOS](#) のマニュアルを参照してください。

次の表を使用して、ActiveVOS のインストールに必要な詳細を収集します。

プロパティ	説明	デフォルト値	インストールの値
ActiveVOS サーバーのインストールディレクトリ	ActiveVOS サーバーをインストールする空のディレクトリ。	<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/avos/server	
接続タイプ	Oracle データベースを設定したときに定義した接続のタイプ。	サービス名	
サービス名	定義した Oracle サービスの名前。	orcl	

プロパティ	説明	デフォルト値	インストールの値
Oracle SID	tnsnames.ora ファイルに定義した Oracle システム ID の名前。	orcl	
サーバー	ActiveVOS スキーマを含む Oracle データベースをホストするマシンのホスト名または IP アドレス。	localhost	
ポート	サーバー上で実行される Oracle リスナの TCP ポート番号。Hub サーバーはこのポートを使用して Oracle データベースサーバーと通信します。	1521	
ActiveVOS スキーマ名	ActiveVOS スキーマユーザーの名前。	avos	
ActiveVOS スキーマパスワード	ActiveVOS スキーマユーザーのパスワード。	-	
Web サービスの URL	ActiveVOS サーバーが実行するホストおよびポート。http または https プロトコルを使用することを選択します。 Hub サーバーは ActiveVOS Central (Process Central と呼ばれる) が ActiveVOS サーバーを呼び出すのに使用するのと同じ URL を使用します。この URL は Informatica ActiveVOS のマニュアルで Process Central AeTaskService URL と呼ばれ、形式は次のとおりです: http://[hostname.domainname]:[port]/active-bpel/services/AeB4PTaskClient-taskOperations。	http://localhost:2809	
ActiveVOS サーバーのインストーラファイル	Multidomain MDM 配布パッケージには、ActiveVOS サーバーのインストーラが含まれます。	ActiveVOS_Server_<operating system>_<version>	
プロセスコンソールユーザー名	ActiveVOS プロセスコンソールの認証ユーザー。	-	
プロセスコンソールのパスワード	ActiveVOS プロセスコンソールユーザーのパスワード。	-	

Informatica Platform: オプション

Informatica Platform をインストールする予定がある場合は、次の場所を記録します。

プロパティ	説明	デフォルト値	インストールの値
インストールプロパティファイル	Informatica Platform 配布ディレクトリの <code>SilentInstall.properties</code> ファイルへの完全修飾パス。 Informatica Platform のサイレントインストールの場合のインストールプロパティを含みます。	-	
Informatica Platform インストールファイル	Informatica Platform 配布ディレクトリのインストーラファイルへの完全修飾パス。	-	

製品使用ツールキット

製品使用ツールキットにより、MDM 環境に関する情報が Informatica に送信されます。この情報は、トラブルシューティングに役立てるため、および使用環境に対する推奨事項を提供するために Informatica グローバルカスタマサポートによって使用されます。ツールキットによる Informatica への情報送信を希望しない場合は、MDM をインストールした後にツールキットを無効化することができます。

次の表を使用して、製品使用ツールキットのインストールに必要な詳細を収集します。

プロパティ	説明	デフォルト値	インストールの値
業種	組織のビジネスに最も一致する業種。	-	
環境	インストールする環境のタイプ。 コマンドラインからインストールする場合、次のいずれかの番号を入力します。 - 1. プロダクション環境 - 2. テストまたは QA 環境 - 3. 開発環境	-	
ネットワークにプロキシサーバーが設定されていますか?	設定されている場合、プロキシサーバーの詳細を指定します。	いいえ	
ホスト	プロキシサーバーの名前または IP アドレス。	-	
ポート	プロキシサーバーにより使用されるポート番号。	-	
ドメイン名	プロキシサーバーがドメインの一部である場合は、ドメインの名前。	-	
ユーザー名	安全なプロキシサーバーを使用する場合は、MDM にアクセスできるユーザーの名前。	-	
パスワード	ユーザーのパスワード。	-	

Hub サーバーをインストールウィザードからインストールする

Hub サーバーをグラフィカルモードでインストールする場合、インストールウィザードを使用します。インストールウィザードでは手順を追ってインストールすることができます。

Hub サーバーとプロセスサーバーをインストールするには、同じユーザー名を使用する必要があります。

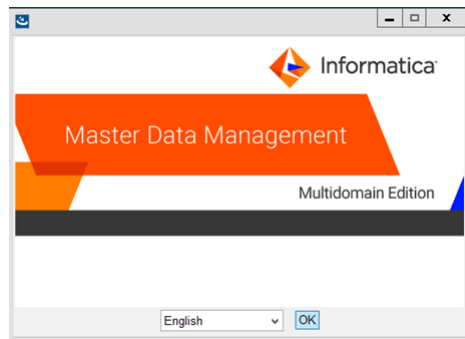
1. アプリケーションサーバーを起動します。
2. 次のディレクトリに移動します。

<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/<operating system name: オペレーティングシステム名>/mrmsrver

3. オペレーティングシステムに応じたタスクを実行することでインストーラを開始します。

- **UNIX。** コマンドラインから、./hub_install.bin を実行します。
- **Windows。** File Explorer から、hub_install.exe をダブルクリックします。

Hub サーバーのインストールウィザードが開始します。



ヒント: インストールウィザードが開始しない場合、サポートされているバージョンの Java がインストールされ、クラスパスまたは環境変数に含まれていることを確認します。

4. 言語を選択して、**[OK]** をクリックします。

【概要】 ウィンドウが表示されます。

5. オンラインの指示に従います。プロンプトが表示されたら、収集したインストールの値を入力します。
6. インストールの最後に、**【設定のサマリ】** ウィンドウで選択したオプションを確認します。
7. 変更が必要な場合、**【前へ】** をクリックして、適切なウィンドウに戻ります。完了したら、**【次へ】** をクリックして最後のウィンドウに戻ります。
8. **【インストール】** をクリックします。
9. **【次の手順】** : 次の手順はインストーラから Hub サーバーのデプロイを選択したかどうかによって異なります。
 - Hub サーバーをインストーラからデプロイすることを選択した場合、Hub サーバーをインストール後のタスクの一部としてデプロイする必要はありません。
 - Hub サーバーを後でデプロイすることを選択した場合、Hub サーバーをインストール後のタスクの一部としてデプロイする必要があります。

Hub サーバーをコマンドラインからインストールする (UNIX のみ)

UNIX では、Hub サーバーをコマンドラインからインストールできます。コマンドラインインストールを開始するスクリプトを実行します。

1. アプリケーションサーバーを起動します。
2. コマンドラインから次のディレクトリに移動します。
<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/<operating system name: オペレーティングシステム名>/mrmsserver
3. 次のコマンドを実行します。
。 /hub_install.bin -i console
Hub サーバーのインストールを要求するプロンプトが表示されます。
4. 収集したインストールの値を入力します。
括弧内に表示されたデフォルト値を使用するには、**Enter** を押します。
5. **【次の手順】**：インストールの完了後、次の手順は Hub サーバーのデプロイを選択したかどうかによって異なります。
 - Hub サーバーをインストーラからデプロイすることを選択した場合、Hub サーバーをインストール後のタスクの一部としてデプロイする必要はありません。
 - Hub サーバーを後でデプロイすることを選択した場合、Hub サーバーをインストール後のタスクの一部としてデプロイする必要があります。

Hub サーバーをサイレントモードでインストールする

サイレントモードで Hub サーバーをインストールできます。サイレントインストールを開始する前に、サイレントインストールのプロパティファイルが設定されていることを確認します。

1. アプリケーションサーバーを起動します。
2. サイレントインストールプロパティファイルをターゲット環境にコピーします。
3. ターゲット環境で、オペレーティングシステムに応じたコマンドを実行します。
 - **UNIX**。 /hub_install.bin -f <absolute path to edited installer properties file>
 - **Windows**。 hub_install.exe -f <absolute path to edited installer properties file>自動インストーラがバックグラウンドで実行します。プロセスにしばらく時間がかかる場合があります。
4. インストーラによる Hub サーバーのデプロイを選択した場合、postinstallSetup.log をチェックして、正常にインストールされたかどうかを確認します。
log ファイルは、次のディレクトリにあります。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/logs
5. **【次の手順】**：インストールの完了後、次の手順は Hub サーバーのデプロイを選択したかどうかによって異なります。

- Hub サーバーをインストーラからデプロイすることを選択した場合、Hub サーバーをインストール後のタスクの一部としてデプロイする必要はありません。
- Hub サーバーを後でデプロイすることを選択した場合、Hub サーバーをインストール後のタスクの一部としてデプロイする必要があります。

クラスタ内のノードに Hub サーバーをインストールする

アプリケーションサーバークラスタ環境では、Hub サーバーアプリケーションをデプロイする必要のあるクラスタのすべてのノードに、Hub サーバーをインストールします。クラスタ内の 1 つのノードでインストールが完了してから、クラスタ内の別のノードのインストールを開始してください。

例えば、WebSphere クラスタには 4 つのサーバーがあり、それらは host1、host2、host3、および host4 で実行され、サーバーは RMI ポート 2812、2813、2814、および 2815 を使用しています。各サーバーにはノードがあります。Hub サーバーは node1、node2、node3 および node4 にインストールする必要があります。Hub サーバーは任意の順序でノードにインストールできます。Hub サーバーのインストールは、1 つのノード、例えば node2 で完了してから、もう 1 つのノードすなわち node1 や node4 でインストールを開始する必要があります。

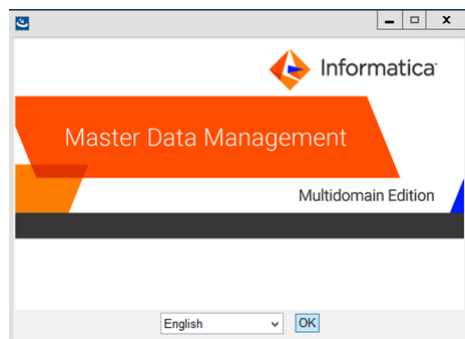
Hub サーバーがインストールされたディレクトリの構造がすべてのノードで確実に同じになるようにしてください。

1. WebSphere クラスタを開始するには、次の手順を実行します。
 - a. WebSphere デプロイメントマネージャを開始します。
 - b. Hub サーバーをインストールする WebSphere クラスタのノードを開始します。
 - c. WebSphere クラスタを開始します。
2. 次のディレクトリに移動します。
 <MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/<operating system name: オペレーティングシステム名>/mrmsrver
3. クラスタノードで Hub サーバーのインストーラを起動するには、お使いのオペレーティングシステム用のコマンドを実行します。

UNIX。 `./hub_install.bin -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false`

Windows。 `.hub_install.exe -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false`

Hub サーバーインストールウィザードが開始します。



4. オンラインの指示に従います。プロンプトが表示されたら、収集したインストールの値を入力します。

5. **【次の手順】**：インストールの完了後、インストールしたすべてのノードに手動で Hub サーバーをデプロイします。

プライマリノードに対して自動デプロイを実行した場合、プライマリノードには手動デプロイは必要ありません。

第 6 章

Hub サーバーのインストール後のタスク

この章では、以下の項目について説明します。

- [インストールログファイルのコピー, 57 ページ](#)
- [バージョンとビルド番号の確認, 58 ページ](#)
- [Elasticsearch のインストールおよび設定, 59 ページ](#)
- [Hub コンソールクライアントの構成（条件付き）, 59 ページ](#)
- [MDM Hub マスターデータベース名の設定, 59 ページ](#)
- [Hub サーバー EAR ファイルの再デプロイ, 60 ページ](#)
- [クラスローダーの設定, 60 ページ](#)
- [アプリケーションサーバー設定の検証と設定（条件付き）, 61 ページ](#)
- [Hub サーバーアプリケーションのデプロイ（条件付き）, 62 ページ](#)
- [Hub サーバーアプリケーションをデプロイするスクリプトの使用（条件付き）, 63 ページ](#)
- [Hub サーバーアプリケーションの手動デプロイ（条件付き）, 65 ページ](#)
- [メタデータキャッシュの構成（省略可能）, 75 ページ](#)
- [Hub コンソールの起動, 77 ページ](#)
- [オペレーショナル参照ストアの登録, 78 ページ](#)

インストールログファイルのコピー

インストールログファイルは、Hub サーバーのインストールに失敗した場合に、問題を解決するのに役立ちます。ログファイルは、インストールドキュメントディレクトリにコピーしておきます。インストールの問題に

ついて問い合わせした場合は、Informatica グローバルカスタマサポートによりログファイルのコピーを求められることがあります。

次の表に、さまざまなタイプのインストールログファイルの説明を示します。

ログファイルのタイプ	説明
インストールログ	<ul style="list-style-type: none">- ファイル名。Informatica_MDM_Hub_Server_Install_<timestamp>.xml- 場所。 <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/logs- 内容。作成されたディレクトリおよびレジストリエントリ、インストールされたファイルおよび実行されたコマンドの名前、およびインストールされた各ファイルのステータス。
インストール前提条件ログ	<ul style="list-style-type: none">- ファイル名。installPrereq.log- 場所。 <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/logs- 内容。インストーラによって実行される前提条件チェックのログ。
デバッグログ	<ul style="list-style-type: none">- ファイル名。infamdm_installer_debug.txt- 場所。 <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server- 内容。インストール中に行う選択、およびインストーラによって実行されるアクションに関する詳細情報。
インストール後のセットアップログ	<ul style="list-style-type: none">- ファイル名。postInstallSetup.log- 場所。 <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/logs- 内容。インストール後プロセス中にインストーラによって実行されるアクション、およびインストール後プロセスで発生するエラーのサマリ。
Hub サーバーログ	<ul style="list-style-type: none">- ファイル名。cmxserver.log- 場所。 <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/logs- 内容。Hub サーバーの操作のサマリ。
WebSphere ログ	<ul style="list-style-type: none">- ファイル名。startServer.log、stopServer.log、SystemErr.log、および SystemOut.log- 場所。 <WebSphere installation directory: WebSphere インストールディレクトリ>/profiles/<Application server profile name: アプリケーションサーバーのプロファイル名>/logs/<server name: サーバー名>- 内容。WebSphere サーバーステータス、およびパフォーマンス情報。

バージョンとビルド番号の確認

バージョンとビルド番号が正しい Hub サーバーがインストールされていることを確認します。

1. コマンドプロンプトを開き、次のディレクトリに移動します。 <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/bin
2. Hub サーバーバージョンとビルド番号を確認するには、次のコマンドを実行します。

UNIX の場合: versionInfo.sh

Windows の場合: versionInfo.bat

注: AIX システムの場合は、< >/jre/bin ディレクトリから Java を実行するように、Java home: Java homeversionInfo.sh スクリプトを変更します。

Elasticsearch のインストールおよび設定

検索を使用するには、MDM Hub 用に Elasticsearch をインストールおよび設定します。

検索のインストールおよび構成の詳細については、*Multidomain MDM の設定ガイド*の「Elasticsearch を使用した検索」の章を参照してください。

Hub コンソールクライアントの構成（条件付き）

Hub サーバマシンに接続するには、Hub コンソールで、`build.properties` ファイルにホスト名およびポートのプロパティが必要です。

次のシナリオで、プロパティファイルを編集します。

- アプリケーションサーバーで HTTPS が有効で、HTTPS ポートを使用する必要がある場合。
 - マルチノードまたはクラスタ環境で、Hub サーバインストールディレクトリを 1 つのノードから別のノードへコピーした場合は、それぞれのノードで編集します。
1. 次のディレクトリにある `build.properties` ファイルを開きます。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/bin
 2. 次のプロパティを編集します。
 - `console.hostname`。アプリケーションサーバーがバインドするサーバーの IP アドレスまたは公的にアクセス可能なホスト名（FQDN）を指定します。
 - `console.webport`。Hub コンソールが使用する必要のある、現在のノードの HTTP または HTTPS ポートを指定します。
 3. ファイルを保存します。

`build.properties` ファイルを編集した後、`postInstallSetup` スクリプトを実行して Hub サーバアプリケーションをデプロイするようにします。

MDM Hub マスターデータベース名の設定

MDM Hub マスターデータベースに `cmx_system` 以外の名前が付いている場合は、`cmxserver.properties` ファイルで MDM Hub マスターデータベースの名前を設定します。

1. 次のディレクトリにある `cmxserver.properties` ファイルを開きます。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/resources
2. `cmx.server.masterdatabase.schemaname` プロパティを、MDM Hub マスターデータベースに対して指定した名前に設定します。
MDM Hub マスターデータベースの名前は、作成時に指定しています。

Hub サーバー EAR ファイルの再デプロイ

postInstallSetup スクリプトを手動または Hub サーバーのインストールの一部として実行した後、WebSphere サーバー管理コンソールを使用して、Hub サーバーの EAR ファイル siperian-mrm.ear をデプロイ解除およびデプロイします。EAR ファイルは、Hub サーバーのインストールディレクトリからデプロイする必要があります。

1. WebSphere サーバー管理コンソールにログインします。
2. siperian-mrm.ear をデプロイ解除します。
3. siperian-mrm.ear をデプロイします。

EAR ファイルは、次のディレクトリにあります。

<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server

次のデプロイメントオプションを設定します。

- **[アプリケーションのインストール準備]** パネルで、デプロイメントにデフォルトのバインディングを生成させるオプションを有効にします。
- **[モジュールのメタデータ]** パネルで、siperian-ejb.jar モジュールの metadata-complete 属性を無効にして、モジュールが読み込まれるたびに注釈ベースのメタデータをスキャンします。

詳細については、WebSphere サーバーのマニュアルを参照してください。

クラスローダーの設定

Hub サーバーアプリケーションのクラスローダーを設定するには、WebSphere デプロイメントマネージャを使用します。

1. Hub サーバーアプリケーションのクラスローダー siperian-mrm.ear、provisioning-ear.ear、および entity360view-ear.ear が、最後に親クラスローダーでクラスを読み込むように設定されていることを確認します。

最初に親クラスローダーでクラスを読み込むようにクラスローダーが設定されている場合は、アプリケーションのクラスローダーを設定します。

- a. **[アプリケーション]** > **[アプリケーションタイプ]** > **[WebSphere エンタープライズアプリケーション]** を選択します。
- b. **[エンタープライズアプリケーション]** ページで、いずれかのアプリケーションをクリックします。
- c. アプリケーションの設定ページで、**[クラスロードおよび更新の検出]** リンクをクリックします。
- d. **クラスローダーの設定** ページで、**[最初にローカルクラスローダーでロードしたクラス (親が最後)]** クラスローダーの順序オプションを選択します。
- e. **[適用]** をクリックしてから、**[OK]** をクリックします。

2. 次のアプリケーション EAR ファイルの Web モジュールのクラスローダーを設定します。

アプリケーション EAR ファイル	Web モジュール	クラスローダーの順序
siperian-mrm.ear	zds-gui.war	最初にローカルクラスローダーで読み込むクラス (親が最後)
provisioning-ear.ear	provisioning.war	最初にローカルクラスローダーで読み込むクラス (親が最後)

- 【アプリケーション】 > 【アプリケーションタイプ】 > 【WebSphere エンタープライズアプリケーション】を選択します。
 - 【エンタープライズアプリケーション】 ページで、いずれかのアプリケーション EAR ファイルの名前をクリックします。
 - アプリケーションの設定ページで、【モジュールの管理】リンクをクリックします。
 - モジュールのリストで、Web モジュールのリンクをクリックします。
 - Web モジュールの設定ページで、クラスローダーの順序を選択します。
 - 【適用】をクリックしてから、【OK】をクリックします。
3. WebSphere を再起動して、Hub サーバーアプリケーションを起動します。

アプリケーションサーバー設定の検証と設定（条件付き）

MDM Hub 環境の要件に基づいて、アプリケーションサーバー設定を検証および設定します。

次の表に、実行可能な設定タスクを示します。

環境設定タスク	説明
アプリケーションサーバー設定の編集	インストール中に postInstallSetup スクリプトを実行し、不適切なアプリケーションサーバー設定が原因でスクリプトに失敗した場合は必須です。
WebSphere クラスタの Hub サーバーの設定	WebSphere クラスタに Hub サーバーをインストールした場合は必須です。

アプリケーションサーバー設定の編集

インストール中に postInstallSetup スクリプトを実行しても、アプリケーションサーバー設定が正しくないためスクリプトが失敗する場合、build.properties ファイルを編集します。また、アプリケーションサーバー設定を変更する場合も、このファイルを編集します。

- 次のディレクトリにある build.properties ファイルを開きます。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/bin
- アプリケーションサーバー設定を編集して保存します。

build.properties ファイルを編集した後、postInstallSetup スクリプトを実行して Hub サーバーアプリケーションをデプロイするようにします。

WebSphere マルチノードまたはクラスタ環境用の Hub サーバーの設定

WebSphere マルチノードまたはクラスタ環境に Hub サーバーをインストールした場合は、WebSphere 環境用に Hub サーバーを設定します。WebSphere 環境用に Hub サーバーを設定するには、cmxserver.properties ファイルで Hub サーバーのプロパティを設定します。

例えば、WebSphere クラスタまたはマルチノード環境には 2 つのサーバーがあり、それらは host1 および host2 で実行し、RMI ポート 2812 および 2813 を使用しているとします。両方のサーバーで WebSphere のプロパティを設定する必要があります。

1. 各サーバーで、次のディレクトリにある cmxserver.properties ファイルを開きます。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/resources
2. 以下のプロパティを設定します。

プロパティ	説明
cluster.flag	手動で追加する必要があります。クラスタリングが有効かどうかを指定します。クラスタリングを有効にするには、true に設定します。デフォルトは false です。
cmx.appserver.hostname	すべてのサーバーのマシン名をカンマで区切って指定します。 例えば、WebSphere クラスタ環境に 2 つのサーバーがあり、それらが host1 および host2 で実行している場合、プロパティを cmx.appserver.hostname=host1,host2 に設定します。
cmx.appserver.rmi.port	サーバーが使用する RMI ポート番号をカンマで区切って指定します。 例えば、WebSphere クラスタまたはマルチノード環境のサーバーが RMI ポート 2812 および 2813 を使用している場合、プロパティを cmx.appserver.rmi.port=2812,2813 に設定します。

プロパティの記述で、1 つ目のサーバーのホスト名およびポート番号は host1 および 2812 です。2 つ目のサーバーのホスト名およびポート番号は host2 および 2813 です。

Hub サーバーアプリケーションのデプロイ（条件付き）

Hub サーバーアプリケーションは、Hub サーバーをインストールするマシンと同じマシンにデプロイする必要があります。

Hub サーバーアプリケーションから、属する Hub サーバーインストールを見つけられるようにする必要があります。そのため、EAR ファイルを移動して別のマシンにアプリケーションをデプロイしないでください。例えば、Hub サーバーをテストマシンにインストールし、アプリケーションを本番マシンにデプロイしたとします。本番マシンにデプロイしたアプリケーションは、テストマシン上のインストールにアクセスしてロギング設定などの情報を見つけることができません。

次のいずれかのシナリオで Hub サーバーアプリケーションをデプロイする必要があります。

- インストールは、アプリケーションサーバーのマルチノード環境またはクラスタ環境にあります。
- インストールは完了したが、インストール中に実行した postInstallSetup スクリプトが失敗する。
- インストール中に postInstallSetup スクリプトをスキップした。

インストールがアプリケーションサーバーのマルチノード環境またはクラスタ環境にある場合は、次の手順を実行してデプロイを完了します。

1. Hub サーバーアプリケーションを 1 つのノードにデプロイします。
2. すべてのファイルを、デプロイメントを含むノード上の certificates ディレクトリから、その他のすべてのノード上の certificates ディレクトリにコピーします。certificates ディレクトリは次の場所にあります。

<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/resources

3. 証明書がコピーされる先のすべてのノード上の Hub サーバーの EAR ファイルを再パッケージ化します。再パッケージプロセスは正しい証明書を使用して hubConsole.jar ファイルを更新します。

- a. 次のディレクトリに移動します。

<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/bin

- b. 次のコマンドを実行します。

UNIX の場合:

./sip_ant.sh repack

Windows の場合:

sip_ant.bat repack

4. Hub サーバーアプリケーションを他のノードにデプロイします。

Hub サーバーアプリケーションをデプロイするには、以下の表で説明されている次のいずれかの手順を使用します。

プロシージャ	説明
デプロイメント用スクリプトの使用	postInstallSetup スクリプトを実行して、Hub サーバーアプリケーションをデプロイします。また、スクリプトによってデータソースの作成と JMS メッセージキューの設定も行われます。
手動デプロイ	Hub サーバーアプリケーションを手動でデプロイします。また、手動でデータソースを作成して JMS メッセージキューを設定する必要もあります。

Hub サーバーアプリケーションをデプロイするスクリプトの使用（条件付き）

インストール中に postInstallSetup スクリプトをスキップした場合は、スクリプトを実行します。インストール後のプロセスでは、Hub サーバーアプリケーションをデプロイし、データソースを作成し、JMS メッセージキューを設定します。

1. コマンドプロンプトを開き、次のディレクトリに移動します。

<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server

2. postInstallSetup スクリプトを実行します。

注: MDM Hub インストーラにバンドルされている ActiveVOS をインストールしなかった場合は、コマンドに ActiveVOS のユーザー名とパスワードを含めないでください。

WebSphere のセキュリティが有効になっていない場合は、次のコマンドを実行します。

UNIX の場合:

```
./postInstallSetup.sh -Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

注: パスワードに感嘆符 (!) が含まれている場合、感嘆符の前にバックスラッシュを付ける必要があります。例えば、パスワードが !! cmx!! の場合、次のパスワードを入力します: \! \!cmx\! \!

Windows の場合:

```
postInstallSetup.bat
-Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

WebSphere のセキュリティを有効にした場合は、次のコマンドを実行します。

UNIX の場合:

```
postInstallSetup.sh
-Dwebsphere.password=<WebSphere password>
-Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

Windows の場合:

```
postInstallSetup.bat
-Dwebsphere.password=<WebSphere password>
-Ddatabase.password=<MDM Hub Master database password>
-Davos.username=<ActiveVOS Console username>
-Davos.password=<ActiveVOS Console password>
-Davos.jdbc.database.username=<ActiveVOS database username>
-Davos.jdbc.database.password=<ActiveVOS database password>
```

ActiveVOS コンソールの資格情報は、アプリケーションサーバーの管理者ユーザーの資格情報と同じです。

ActiveVOS データベースの資格情報は、create_bpm スクリプトを実行するために使用した資格情報と同じです。

3. siperian-ejb.jar モジュールの注釈ベースのメタデータのスキャンを有効にするには、WebSphere サーバー管理コンソールを使用して、EAR ファイル siperian-mrm.ear をデプロイ解除およびデプロイします。

詳細については、[「Hub サーバー EAR ファイルの再デプロイ」 \(ページ 60\)](#)を参照してください。

Hub サーバーアプリケーションの手動デプロイ（条件付き）

postInstallSetup スクリプトをインストール中にスキップしたか、postInstallSetup スクリプトが失敗する場合、Hub サーバーアプリケーションを手動でデプロイできます。Hub サーバーアプリケーションのデプロイは、Hub サーバーのインストールディレクトリから行う必要があります。

Hub サーバーアプリケーションをデプロイするには、次のタスクを実行します。

1. データソースの作成
2. JMS メッセージキューの設定
3. Hub サーバーアプリケーションの再パッケージ化
4. Hub サーバーアプリケーションのデプロイ
5. クラスローダーの設定
6. Hub サーバーでの JMS メッセージキューの設定
7. Informatica Data Director (IDD) 用のサーバーリソースの設定

手順 1. データソースの作成

Hub サーバーアプリケーションを手動でデプロイする前に、データソースを作成します。また複数のプロセスサーバーを設定したり、インストール問題をトラブルシューティングしたりする場合も、データソースを作成します。

1. JDBC ドライバをインストールします。
2. MDM Hub マスタデータベースのデータソースを作成します。
3. オペレーショナル参照ストアのデータソースを作成します。

手順 1.JDBC ドライバのインストール

MDM Hub マスタデータベースとオペレーショナル参照ストア（ORS）のデータソースを作成する前に、JDBC ドライバをインストールします。

サポートされているバージョンの JDBC ドライバの取得方法については、Oracle にお問い合わせください。

- ▶ JDBC ドライバを次のディレクトリにコピーします。

```
<WebSphere installation directory>/lib
```

手順 2.MDM Hub マスタデータベースのデータソースの作成

プロセスサーバーマシンに JDBC ドライバをインストールした後で、MDM Hub マスタデータベース用のデータソースを作成します。

1. WebSphere Application Server の管理コンソールを起動します。
2. ドライバライブラリの場所を指定します。
 - a. コンソールナビゲーションツリーで、**[環境]** を展開します。
 - b. **[WebSphere 変数]** リンクをクリックします。
 - c. JDBC 変数が次の JDBC ドライバディレクトリを参照するように更新します。

```
<WebSphere installation directory: WebSphere インストールディレクトリ>/lib
```

3. MDM Hub マスタデータベースのデータソースで使用するセキュリティアカウントを作成します。
 - a. コンソールのナビゲーションツリーで、**【セキュリティ】**を展開します。
 - b. **【セキュアな管理、アプリケーション、インフラストラクチャ】**リンクをクリックします。
 - c. **【認証】**で**【Java 認証および承認サービス】**を展開し、**【J2C 認証データ】**をクリックします。
 - d. **【新規】**をクリックし、以下のプロパティを指定します。

プロパティ	説明
エイリアス	MDM Hub マスタデータベースの名前。
ユーザー ID	MDM Hub マスタデータベースに接続するためのユーザー名。
パスワード	MDM Hub マスタデータベースにアクセスするためのパスワード。

- e. **【OK】**をクリックします。
4. JDBC プロバイダを作成します。
 - a. **【リソース】** > **【JDBC】**を展開し、**【JDBC プロバイダ】**をクリックします。
【JDBC プロバイダ】ページが表示されます。
 - b. データソースを使用するアプリケーションのスコープを選択します。
 - c. **【新規】**をクリックし、以下のプロパティを指定します。

プロパティ	説明
データベースタイプ	データベースの種類。 【Oracle】 を選択します。
プロバイダタイプ	JDBC プロバイダのタイプ。 【Oracle JDBC ドライバ】 を選択します。
実装タイプ	データソース実装のタイプ。 【XA データソース】 を選択します。
名前	JDBC プロバイダの名前。名前を「Informatica MDM Oracle JDBC Provider (XA)」に変更します。

- d. **【次へ】**をクリックし、**【完了】**をクリックします。
5. MDM Hub マスタデータベースのデータソースを作成します。
 - a. 作成した JDBC プロバイダをクリックします。
【設定】ページが表示されます。
 - b. **【追加プロパティ】**の下で、**【データソース】**をクリックします。
【データソース】ページが表示されます。
 - c. **【新規】**をクリックします。

- d. 以下のデータソースのプロパティを指定してください。

プロパティ	説明
名前	データソース名。「MDM Master Data Source」と指定します。
JNDI 名	JDBC データソースが関連付けられる場所への JNDI パス。jdbc/siperian-cmx_system-ds と指定します。 注: JNDI 名は小文字で指定する必要があります。
管理されるコンポーネントの認証エイリアス	マスターデータベースデータソースの認証エイリアス。[<host name>/cmx_system] を選択します。

- e. **【次へ】** をクリックし、**【完了】** をクリックします。

手順 3.オペレーショナル参照ストアのデータソースの作成

プロセスサーバマシンに JDBC ドライバをインストールした後で、各オペレーショナル参照ストア用のデータソースを作成します。

- WebSphere Application Server の管理コンソールを起動します。
- ドライバライブラリの場所を指定します。
 - コンソールナビゲーションツリーで、**【環境】** を展開します。
 - 【WebSphere 変数】** リンクをクリックします。
 - JDBC 変数が次の JDBC ドライバディレクトリを参照するように更新します。
 <WebSphere installation directory: WebSphere インストールディレクトリ>/lib
- オペレーショナル参照ストアで使用するセキュリティアカウントを作成します。
 - コンソールのナビゲーションツリーで、**【セキュリティ】** を展開します。
 - 【セキュアな管理、アプリケーション、インフラストラクチャ】** リンクをクリックします。
 - 【認証】** で **【Java 認証および承認サービス】** を展開し、**【J2C 認証データ】** をクリックします。
 - 【新規】** をクリックし、以下のプロパティを設定します。

プロパティ	説明
エイリアス	オペレーショナル参照ストアの名前。
ユーザー ID	オペレーショナル参照ストアに接続するためのユーザー名。
パスワード	オペレーショナル参照ストアにアクセスするためのパスワード。

- e. **【OK】** をクリックします。
- JDBC プロバイダを作成します。
 - 【リソース】** > **【JDBC】** を展開し、**【JDBC プロバイダ】** をクリックします。
【JDBC プロバイダ】 ページが表示されます。
 - データソースを使用するアプリケーションのスコープを選択します。

- c. **【新規】** をクリックし、以下のプロパティを指定します。

プロパティ	説明
データベースタイプ	データベースの種類。[Oracle] を選択します。
プロバイダタイプ	JDBC プロバイダのタイプ。[Oracle JDBC ドライバ] を選択します。
実装タイプ	データソース実装のタイプ。[XA データソース] を選択します。
名前	JDBC プロバイダの名前。名前を「Informatica MDM Oracle JDBC Provider (XA)」に変更します。

- d. **【次へ】** をクリックし、**【完了】** をクリックします。

5. オペレーショナル参照ストアのデータソースを作成します。

- a. 作成した JDBC プロバイダをクリックします。
【設定】 ページが表示されます。
- b. **【追加プロパティ】** の下で、**【データソース】** をクリックします。
【データソース】 ページが表示されます。
- c. **【新規】** をクリックします。
- d. 以下のデータソースのプロパティを指定してください。

プロパティ	説明
名前	データソース名。「MDM ORS Data Source」と指定します。
JNDI 名	JDBC データソースが関連付けられる場所への JNDI パス。jdbc/siperian- <Oracle host name>-<Oracle SID>-<Operational Reference Store name>-ds を指 定します。 注: JNDI 名は小文字で指定する必要があります。
管理されるコンポーネントの認証エイリアス	マスターデータベースデータソースの認証エイリアス。<host name>/ <Operational Reference Store name>を選択します。

- e. **【次へ】** をクリックし、**【完了】** をクリックします。

手順 2。JMS メッセージキューの設定

Hub サーバーアプリケーションを手動でデプロイする前に、JMS メッセージキューを設定します。また問題をトラブルシューティングする場合にも、JMS メッセージキューを手動で設定しなければならない場合があります。例えば、自動化されたキュー作成プロセスが失敗したり、インストール後に誤ってキューを削除した場合などには、メッセージキューを手動で設定する必要があります。

サービス統合フレームワーク（SIF）は、JMS メッセージキュー上のメッセージ駆動型 Bean を使用して、受信非同期 SIF 要求を処理します。MDM Hub の実装に使用するアプリケーションサーバーに対して、メッセージキューと接続ファクトリを設定します。JMS メッセージキューを設定する際に、接続ファクトリも作成できます。

JMS メッセージキューを設定するには、次のタスクを実行します。

1. WebSphere Server でバスを作成します。
2. キュー接続ファクトリを設定します。
3. SIF 要求の JMS メッセージキューを設定します。
4. 検索の JMS メッセージキューを設定します。(条件付き)

手順 1.WebSphere Server でのバスの作成

JMS メッセージキューを作成するには、WebSphere コンソールを使用してバスを作成します。

1. WebSphere コンソールで、**【サービスの統合】 > 【バス】** に移動します。
2. SiperianBus という名前のバスを作成し、変更を保存します。
3. 作成した [SiperianBus] に移動して、**【バスのメンバ】** をクリックします。
4. **【サーバー】** リストに新しいメンバを追加します。
5. アプリケーションが実行されているサーバーを選択して **【次へ】** をクリックし、**【データストア】** を選択します。
6. **【次へ】** をクリックし、**【生成された JNDI でデフォルトのデータソースを作成する】** の名前がチェックされていることを確認します。
7. **【次へ】** をクリックし、**【完了】** をクリックします。
8. **【保存】** をクリックします。
9. 作成した SiperianBus に移動して、**【接続先】** をクリックします。
10. **【新規】** をクリックし、送信先のタイプとして **【キュー】** を選択して、**【次へ】** をクリックします。
11. 名前 SiperianQueue を識別子として使用し、**【次へ】** をクリックします。
12. 作成したバスメンバを選択し、**【次へ】** をクリックします。
13. **【完了】** をクリックし、**【保存】** をクリックします。

手順 2.キュー接続ファクトリの設定

作成したバスに対してキュー接続ファクトリを作成、設定します。

1. WebSphere コンソールで、**【リソース】 > 【JMS】 > 【JMS プロバイダ】** に移動します。
2. **【Node=<servername>Node01】** などのノードを一覧から選択します。
3. 使用する JMS プロバイダを選択します。
4. **【キュー接続ファクトリ】** をクリックします。
5. 名前と JNDI 名として siperian xaconnectionfactory を指定します。
6. キュー接続ファクトリのバス名として **【SiperianBus】** を選択します。
7. **【保存】** をクリックします。
8. **【キュー】** をクリックし、スコープの同じノード (**【Node=<servername>Node01】** など) を選択します。

手順 3.JMS キューの設定

バスと接続ファクトリを設定した後、システム JMS メッセージキューとカスタム JMS メッセージキューを設定します。次に、キューの有効化指定を作成します。

1. **【キュー】** ページで **【新規作成】** をクリックします。
2. JMS プロバイダを選択し、**【OK】** をクリックします。

3. クラスタ環境では、**【スコープ】** を1つ以上のノードに設定できます。

ヒント: 1つのノードでキューを設定することから始めることができます。その後、複数のノードに展開できます。

4. JMS キューの以下のオプションを指定します。

オプション	システムキューの値	カスタムキューの値
名前	siperian.sif.jms.queue	siperian.sif.test.jms.queue
JNDI 名	queue/siperian.sif.jms.queue	queue/siperian.sif.test.jms.queue
説明	MDM JMS システムキュー	MDM JMS カスタムキュー
バス名	SiperianBus	SiperianBus
キュー名	SiperianQueue	SiperianQueue

5. **【OK】** をクリックします。
6. **【JMS の有効化指定】** をクリックし、リストからスコープを選択します。
7. 有効化を設定するには、**【新規】** をクリックします。
8. 次のオプションを指定します。

オプション	値
名前	SiperianActivation
JNDI 名	SiperianActivation
宛先タイプ	キュー
送信先の JNDI 名	queue/siperian.sif.jms.queue
バス名	SiperianBus

9. **【OK】** をクリックします。

手順 4. 検索の JMS キューの設定

検索機能では、Data Director 内でフルテキスト検索を有効にするために JMS キューが必要です。検索 JMS キューをアクティブ化したり、Hub コンソールでキューを設定したりする必要はありません。

検索用の JMS キューを設定する前に、Elasticsearch を設定する必要があります。

1. **【キュー】** ページで **【新規作成】** をクリックします。
2. JMS プロバイダを選択し、**【OK】** をクリックします。
3. クラスタ環境では、**【スコープ】** を1つ以上のノードに設定します。

ヒント: 1つのノードでキューを設定することができます。その後、複数のノードに展開できます。

4. 次のオプションを指定します。

オプション	値
名前	informatica.mdm.sss.jms.queue
JNDI 名	queue/informatica.mdm.sss.jms.queue
説明	Siperian JMS キュー検索用
バス名	SiperianBus
キュー名	SiperianQueue
配信モード	アプリケーション

5. **[OK]** をクリックします。

手順 3.Hub サーバーの EAR ファイルの再パッケージ化

cmxserver.properties ファイルで cmx.home プロパティを編集するか、アプリケーションサーバークラスタをインストールした場合は、Hub サーバーの EAR ファイルを再パッケージ化します。

1. EAR という名前のディレクトリを作成します。
 - a. 次のディレクトリに移動します。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/lib
 - b. 次のコマンドを実行します。
mkdir ear
2. カスタム JAR ファイルがある場合は、各カスタム JAR ファイルを前の手順で作成した EAR ディレクトリにコピーします。

カスタム JAR ファイルを EAR ディレクトリにコピーするには、次のコマンドを実行します。

```
copy <location of custom JAR file: カスタム JAR ファイルの場所>/<custom JAR file name: カスタム JAR ファイル名>.jar ear
```

カスタムユーザーイグジット用にカスタム JAR ファイルが必要になる場合があります。
3. EAR ファイルを再パッケージ化します。
 - a. 次のディレクトリに移動します。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/bin
 - b. 次のコマンドを実行します。
UNIX の場合: ./sip_ant.sh repack
Windows の場合: sip_ant.bat repack

手順 4. Hub サーバーアプリケーションのデプロイ

Hub サーバーアプリケーションを手動でデプロイできます。Hub サーバーアプリケーションのデプロイは、Hub サーバーのインストールディレクトリから行う必要があります。

1. 既存のデプロイメントがある場合は、WebSphere サーバー管理コンソールを使用して、次のデプロイメントファイルのデプロイを解除します。

デプロイメントファイル名	説明
siperian-mrm.ear	必須。Hub サーバーアプリケーション。
provisioning-ear.ear	必須。プロビジョニングツールアプリケーション。
entity360view-ear.ear	オプション。エンティティ 360 フレームワーク。
informatica-mdm-platform-ear.ear	オプション。Informatica Platform アプリケーション。

2. WebSphere 管理コンソールを使用して、前の手順で表示されたファイルをデプロイします。

デプロイメントファイルは次のディレクトリにあります。

<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server

次のデプロイメントオプションを設定します。

- **【アプリケーションのインストール準備】** パネルで、デプロイメントにデフォルトのバインディングを生成させるオプションを有効にします。
- **【モジュールのメタデータ】** パネルで、siperian-ejb.jar モジュールの metadata-complete 属性を無効にして、モジュールが読み込まれるたびに注釈ベースのメタデータをスキャンします。
- クラスターノードでデプロイする場合、**【インストールオプションの選択】** パネルで、アプリケーションを配布するデプロイメントのオプションを有効にします。

3. クラスターノードに Hub サーバーアプリケーションをデプロイした場合は、次の手順を実行します。

- a. クラスター、デプロイメントマネージャ、およびノードを停止します。
- b. ノード、デプロイメントマネージャ、クラスターの順に開始します。

詳細については、WebSphere サーバーのマニュアルを参照してください。

手順 5. クラスローダーの設定

Hub サーバーアプリケーションを手動でデプロイしたら、各 Hub サーバーアプリケーションにクラスローダーを設定します。

1. 次の Hub サーバーアプリケーションのクラスローダーを設定します。siperian-mrm.ear、provisioning-ear.ear、および entity360view-ear.ear
 - a. **【アプリケーション】** > **【アプリケーションタイプ】** > **【WebSphere エンタープライズアプリケーション】** を選択します。
 - b. **【エンタープライズアプリケーション】** ページで、いずれかのアプリケーションをクリックします。
 - c. アプリケーションの設定ページで、**【クラスロードおよび更新の検出】** リンクをクリックします。
 - d. **クラスローダー** の設定ページで、**【最初にローカルクラスローダーでロードしたクラス（親が最後）】** クラスローダーの順序オプションを選択します。
 - e. **【適用】** をクリックしてから、**【OK】** をクリックします。

2. 次のアプリケーション EAR ファイルの Web モジュールのクラスローダーを設定します。

アプリケーション EAR ファイル	Web モジュール	クラスローダーの順序
siperian-mrm.ear	zds-gui.war	最初にローカルクラスローダーで読み込むクラス (親が最後)
provisioning-ear.ear	provisioning.war	最初にローカルクラスローダーで読み込むクラス (親が最後)

- a. **【アプリケーション】** > **【アプリケーションタイプ】** > **【WebSphere エンタープライズアプリケーション】** を選択します。
 - b. **【エンタープライズアプリケーション】** ページで、いずれかのアプリケーション EAR ファイルの名前をクリックします。
 - c. アプリケーションの設定ページで、**【モジュールの管理】** リンクをクリックします。
 - d. モジュールのリストで、Web モジュールのリンクをクリックします。
 - e. Web モジュールの設定ページで、クラスローダーの順序を選択します。
 - f. **【適用】** をクリックしてから、**【OK】** をクリックします。
3. WebSphere を再起動して、Hub サーバーアプリケーションを起動します。

手順 6. Hub サーバーでの JMS メッセージキューの設定

Hub サーバーアプリケーションを手動でデプロイしたら、Hub サーバーに JMS メッセージキューを設定します。

Hub サーバーに JMS メッセージキューを設定するには、次のタスクを実行します。

1. Hub コンソールを起動します。
2. メッセージキューサーバーを追加します。
3. メッセージキューを追加します。

手順 1. Hub コンソールの起動

MDM Hub にアクセスするには、Hub コンソールを起動します。

1. ブラウザウィンドウを開いて、以下の URL を入力します。
`http://<MDM Hub host: MDM Hub ホスト>:<port number: ポート番号>/cmx/`
【ハブコンソールの起動】 ページが表示されます。
2. ユーザー名とパスワードを入力し、**【ダウンロード】** をクリックします。
Hub コンソールのダウンロードを起動するために必要な MDM Hub アプリケーション JAR ファイル。
注: MDM Hub アプリケーション JAR ファイルをダウンロードできない場合は、MDM 管理者に連絡してください。管理者は、次のディレクトリから JAR ファイルを配布できます。 <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/resources/hub
3. アプリケーション JAR ファイルを実行します。
アプリケーションの最大メモリ割り当てプールを指定するには、次のコマンドを実行します。
`java -Xmx<n>G -jar hubConsole.jar`

<n>は、GB 単位の最大メモリ割り当てです。

[Informatica MDM Hub ログイン] ダイアログボックスが表示されます。

4. ユーザー名とパスワードを入力して、**[OK]** をクリックします。

[データベースの変更] ダイアログボックスが表示されます。

5. ターゲットデータベースを選択します。

ターゲットデータベースは MDM Hub マスターデータベースです。

6. リストから言語を選択して、**[接続]** をクリックします。

Hub コンソールのユーザーインターフェースが、選択した言語で表示されます。Hub コンソールユーザーインターフェースを表示する言語を変更する場合は、言語を選択して Hub コンソールを再起動します。

手順 2.メッセージキューサーバーの追加

メッセージキューを追加する前に、メッセージキューサーバーを MDM Hub 実装に追加する必要があります。

1. Hub コンソールの設定ワークベンチで、**[メッセージキュー]** をクリックします。
2. **[書き込みロック]** > **[ロックの取得]** の順にクリックします。
3. メッセージキューツールの中央のペインを右クリックして、**[メッセージキューサーバーの追加]** をクリックします。

[メッセージキューサーバーの追加] ダイアログボックスが表示されます。

4. メッセージキューサーバーの詳細情報を入力します。

次の表に、JMS メッセージキューサーバーを設定するために使用するフィールドを示します。

フィールド名	値
接続ファクトリ名	接続ファクトリの名前。 siperian.mrm.jms.xaconnectionfactory を指定します。
表示名	Hub コンソールに表示される必要のあるメッセージキューサーバーの名前。 siperian.mrm.jms.xaconnectionfactory を指定します。

5. **[OK]** をクリックします。

メッセージキューサーバーが追加されました。

手順 3.メッセージキューの追加

メッセージキューサーバーにカスタムメッセージキューを追加できます。

1. Hub コンソールの設定ワークベンチで、**[メッセージキュー]** をクリックします。
2. **[書き込みロック]** > **[ロックの取得]** の順にクリックします。
3. メッセージキューツールの中央のペインでメッセージキューサーバーを右クリックしてから、**[メッセージキューの追加]** をクリックします。

[メッセージキューの追加] ダイアログボックスが表示されます。

4. JMS メッセージキューの詳細を入力します。

次の表に、JMS メッセージキューのフィールドを示します。

フィールド名	値
Queue Name	メッセージキューの名前。 queue/siperian.sif.test.jms.queue を指定します。
表示名	Hub コンソールに表示されるメッセージキューの名前。 queue/siperian.sif.test.jms.queue を指定します。

5. **[OK]** をクリックします。
メッセージキューがメッセージキューサーバーに追加されました。
6. 右ペインで、**[メッセージトリガで使用]** オプションを選択します。
7. **[テスト]** をクリックします。
メッセージキューのテスト結果が表示されます。

手順 7. Informatica Data Director のサーバーリソースの設定

Informatica Data Director (IDD) を使用する場合は、JNDI URL リソースを設定します。

1. WebSphere Server 管理コンソールで、**[リソース]** > **[URL]** をクリックします。
2. JNDI URL リソースを設定するには、次のプロパティを設定します。

プロパティ	値
スコープ	Hub サーバーのスコープを指定します。
名前	Hub サーバーホームディレクトリ
JNDI 名	url/hubserver/home
仕様	file:///<Hub サーバーインストールディレクトリ>

メタデータキャッシュの構成（省略可能）

メタデータキャッシュは、データオブジェクト、リポジトリオブジェクト、検索トークンなどのアイテムを管理します。MDM Hub は、Infinispan を使用してメタデータキャッシュを実行します。Infinispan は、Hub サーバーと共にインストールされます。Hub サーバーが使用するキャッシュの場合、Infinispan 設定ファイルにはデフォルトの属性値が含まれます。

キャッシュのデフォルトの属性値を使用して MDM Hub ハブを実行します。パフォーマンスの問題が発生した場合は、環境にさらに適合するように属性値を微調整できます。

次の表は、デフォルトの属性値をまとめたものです。

Infinispan の要素と属性	デフォルト値	説明
locking acquire-timeout	60000	Hub サーバーがロックを取得しようとする最大時間。
transaction stop-timeout	30000	キャッシュが停止したときに、この属性は、Hub サーバーがリモートおよびローカルトランザクションを完了するまでに Infinispan が待機する最大時間を設定します。
transport cluster	infinispan-cluster	基本になるグループ通信クラスタの名前。
transport stack	UDP	設定のタイプ: UDP または TCP。設定は、jgroups-udp.xml ファイルおよび jgroups-tcp.xml ファイルで定義されます。
transport node-name	\$node\$	現在のノードの名前。Hub サーバーはこの属性を設定します。 ノード名のデフォルトは、ホスト名と乱数の組み合わせです。この番号は、同じホスト上の複数のノードを区別します。
transport machine	\$machine\$	ノードが実行されるマシンの ID。Hub サーバーはこの属性を設定します。
expiration lifespan	--	キャッシュエントリの最大存続期間（ミリ秒単位）。キャッシュエントリが存続期間を超えると、エントリはクラスタ内で有効期限が切れます。 パフォーマンスを最適化する必要がある場合は、DISABLE_WHEN_LOCK、DATA_OBJECTS、および REPOS_OBJECTS の各キャッシュの存続期間を長くします。 例えば、存続期間を長くして、1 時間（3600000）から 1 日（86400000）にすることができます。 各キャッシュには、この属性の独自のデフォルト値があります。デフォルト値を見つけるには、infinispanConfig.xml ファイルを開きます。
expiration interval	--	存続期間を確認するための最大間隔。パフォーマンスを最適化する必要がある場合は、DISABLE_WHEN_LOCK、DATA_OBJECTS、および REPOS_OBJECTS の各キャッシュの間隔を長くします。 例えば、間隔を 5 秒（5000）から 5 分（300000）に増やすことができます。 各キャッシュには、この属性の独自のデフォルト値があります。デフォルト値を見つけるには、infinispanConfig.xml ファイルを開きます。

Infinispan 属性の編集

メタデータキャッシュ属性を設定するには、Hub サーバーの `infinispanConfig.xml` ファイルを編集します。
Infinispan 設定のヘルプについては、Infinispan のマニュアルを参照してください。

注: プロセスサーバーには、Infinispan 設定ファイルもあります。デフォルトの属性値は十分に機能するはずですが、プロセスサーバーのパフォーマンスに問題があることに気付く場合は、属性値を微調整できます。

1. 次のディレクトリに移動します。 <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/resources
2. 次のファイルのバックアップコピーを作成します。infinispanConfig.xml
3. infinispanConfig.xml ファイルを開き、Infinispan バージョン番号を見つけます。これは xsi:schemaLocation 属性内にあります。

4. その Infinispan バージョンのマニュアルを確認します。
注: 次の URL で、パスに # が含まれている場合は、バージョン番号を置き換えます。#。
 - 設定スキーマを表示するには、ファイルの `xsi:schemaLocation` 属性に含まれている URL に移動します。
 - 属性の詳細を知るには、<https://docs.jboss.org/infinispan/<#.#.x>/configdocs/> に移動してください。
 - Infinispan の詳細を知るには、<http://infinispan.org/docs/<#.#.x>/> に移動し、「Frequently Asked Questions」リンクを選択します。
5. ファイルを編集して保存します。

Hub コンソールの起動

MDM Hub にアクセスするには、HTTP または HTTPS 接続を使用して Hub コンソールを起動します。

Hub コンソールを開始する前に、次の情報があることを確認してください。

- ホスト名と URL のポート番号
- ユーザー名とパスワード

1. ブラウザウィンドウを開いて、以下の URL を入力します。

`http://<MDM Hub host: MDM Hub ホスト>:<port number: ポート番号>/cmx/`

「ハブコンソールの起動」ページが表示されます。

2. ユーザー名とパスワードを入力し、**【ダウンロード】** をクリックします。

Hub コンソールのダウンロードを起動するために必要な MDM Hub アプリケーション JAR ファイル。

注: MDM Hub アプリケーション JAR ファイルをダウンロードできない場合は、MDM 管理者に連絡してください。管理者は、次のディレクトリから JAR ファイルを配布できます。 <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/resources/hub

3. アプリケーション JAR ファイルを実行します。

アプリケーションの最大メモリ割り当てプールを指定するには、次のコマンドを実行します。

```
java -Xmx<n>G -jar hubConsole.jar
```

<n>は、GB 単位の最大メモリ割り当てです。

【Informatica MDM Hub ログイン】 ダイアログボックスが表示されます。

4. ユーザー名とパスワードを入力して、**【OK】** をクリックします。

【データベースの変更】 ダイアログボックスが表示されます。

5. ターゲットデータベースを選択します。

ターゲットデータベースは MDM Hub マスターデータベースです。

6. リストから言語を選択して、**【接続】** をクリックします。

Hub コンソールのユーザーインターフェースが、選択した言語で表示されます。Hub コンソールユーザーインターフェースを表示する言語を変更する場合は、言語を選択して Hub コンソールを再起動します。

オペレーショナル参照ストアの登録

オペレーショナル参照ストアを作成したら、それを Hub コンソールで登録する必要があります。オペレーショナル参照ストアを単一の MDM Hub マスターデータベースに登録します。

1. Hub コンソールを起動します。
[データベースの変更] ダイアログボックスが表示されます。
2. [MDM Hub マスタデータベース] を選択して、[接続] をクリックします。
3. [設定] ワークベンチで、[データベース] ツールをクリックします。
4. [書き込みロック] メニューから [ロックの取得] をクリックします。
5. [データベース] ペインで、[データベースの登録] ボタンをクリックします。

Informatica MDM Hub 接続ウィザードが表示されます。

6. Oracle データベースタイプオプションを選択して、[次へ] をクリックします。
7. データベースの接続プロパティを設定します。
 - a. Oracle の接続方式を選択して、[次へ] をクリックします。

次の表に、選択できる Oracle の接続方式を示します。

接続方式	説明
サービス	サービス名を使用して Oracle に接続します。
SID	Oracle のシステム ID を使用して Oracle に接続します。

サービス名と SID 名の詳細については、Oracle のマニュアルを参照してください。

[接続プロパティ] ページが表示されます。

- b. 選択する接続タイプの接続プロパティを指定し、[次へ] をクリックします。

以下の表に、接続のプロパティを示します。

プロパティ	説明
データベース表示名	Hub コンソールに表示される必要のあるオペレーショナル参照ストアの名前。
マシン識別子	Hub ストアインスタンスからのレコードを一意に識別するためにキーに割り当てられるプレフィックス。
データベースホスト名	Oracle データベースをホストするサーバーの IP アドレスまたは名前。
SID	サーバー上で実行される Oracle データベースのインスタンスを参照する Oracle システム識別子。[SID] フィールドは、[SID] 接続タイプを選択した場合に表示される。
サービス	Oracle データベースへの接続に使用する Oracle サービスの名前。[サービス] フィールドは、[サービス] 接続タイプを選択した場合に表示される。

プロパティ	説明
ポート	Oracle データベースサーバー上で実行される Oracle リスナの TCP ポート。デフォルトは 1521。
Oracle TNS 名	TNSNAMES.ORA ファイルで定義されている、ネットワーク上のデータベース名。 Oracle TNS 名は、Oracle データベースのインストール時に設定します。Oracle TNS 名の詳細については、Oracle のマニュアルを参照してください。
スキーマ名	オペレーショナル参照ストアの名前。
パスワード	オペレーショナル参照ストアユーザー名に関連付けられているパスワード。
DDM 接続 URL	オプション。Dynamic Data Masking アプリケーションに接続するための URL。この Dynamic Data Masking アプリケーションの URL は、Dynamic Data Masking のホスト名およびポート番号を使用する点を除けば、データベースへの接続に使用する URL と似ています。

注: スキーマ名とユーザー名は、どちらもオペレーショナル参照ストアを作成するときに指定したオペレーショナル参照ストアの名前です。この情報が必要な場合はデータベース管理者に問い合わせます。

サマリページが表示されます。

- c. サマリを確認し、追加の接続プロパティを指定します。

以下の表に、設定可能な追加の接続プロパティを示します。

プロパティ	説明
接続 URL	接続 URL。接続ウィザードでは、デフォルトで接続 URL が生成されます。 次のリストに、Oracle 接続タイプの接続 URL のフォーマットを示します。 - サービス接続タイプ。jdbc:oracle:thin:@//database_host:port/service_name - SID 接続タイプ。jdbc:oracle:thin:@//database_host:port:sid [サービス] 接続タイプの場合、カスタマイズし、後で別の接続 URL をテストするオプションがあります。
登録後にデータソースを作成する	登録後にアプリケーションサーバーのデータソースを作成する場合は選択します。このオプションを選択しない場合、手動でデータソースを設定する必要があります。 注: アプリケーションサーバークラスタ環境で、オペレーショナル参照ストア用のデータソースと接続プールを手動で作成します。

- d. サービス接続タイプには、デフォルトの URL を変更する場合、**【編集】** ボタンをクリックし、URL を指定して **【OK】** をクリックします。

8. **【完了】** をクリックします。

【データベースの登録】 ダイアログボックスが表示されます。

9. **【OK】** をクリックします。

MDM Hub でオペレーショナル参照ストアが登録されます。

10. 登録したオペレーショナル参照ストアを選択して、**【データベース接続のテスト】** ボタンをクリックします。

データベース接続をテストする前に、アプリケーションサーバーを再起動する必要があります。

【データベースのテスト】 ダイアログに、データベース接続テストの結果が表示されます。

11. **【OK】** をクリックします。

オペレーショナル参照ストアが登録され、データベースへの接続がテストされます。

第 7 章



Process サーバーのインストール

この章では、以下の項目について説明します。

- [プロセスサーバーのインストール, 81 ページ](#)
- [インストーラワークフローの確認, 82 ページ](#)
- [インストールの値の収集, 83 ページ](#)
- [Process サーバーをインストールウィザードからインストールする, 86 ページ](#)
- [プロセスサーバーをコマンドラインからインストールする \(UNIX のみ\), 87 ページ](#)
- [Process サーバーをサイレントモードでインストールする, 88 ページ](#)
- [クラスタ内のノードに Process サーバーをインストールする, 89 ページ](#)

プロセスサーバーのインストール

インストールウィザード、サイレントインストールスクリプト、または UNIX システムではコマンドラインスクリプトを使用して、プロセスサーバーをインストールできます。インストール前のタスクを完了し、インストーラの開始前に必要な情報を収集した場合、インストールプロセスには約 15 分かかります。

	STOP! インストール前のタスクを完了しましたか? インストーラを実行する前にインストール前のタスクを完了しないと、インストールは失敗します。
	インストール準備状況のチェックリスト <ul style="list-style-type: none"><input type="checkbox"/> MDM 実装プランを作成しました。<input type="checkbox"/> サーバーがシステム要件を満たしていることを確認しました。<input type="checkbox"/> オペレーティングシステムとソフトウェアのバージョンがサポートされていることを確認しました。<input type="checkbox"/> オペレーティングシステムとソフトウェアバージョンの既知の制限事項を確認しました。<input type="checkbox"/> サポートされているバージョンのアプリケーションサーバーをインストールして設定しました。<input type="checkbox"/> サポートされているバージョンのデータベース管理システムをインストールして設定しました。<input type="checkbox"/> 環境に合わせてインストール前の環境設定タスクを実行しました。<input type="checkbox"/> MDM ライセンスファイルをアクセス可能な場所に保存しました。

タスクを実行しなかった場合は、前の章に戻ってタスクを完了するためのヘルプを参照してください。

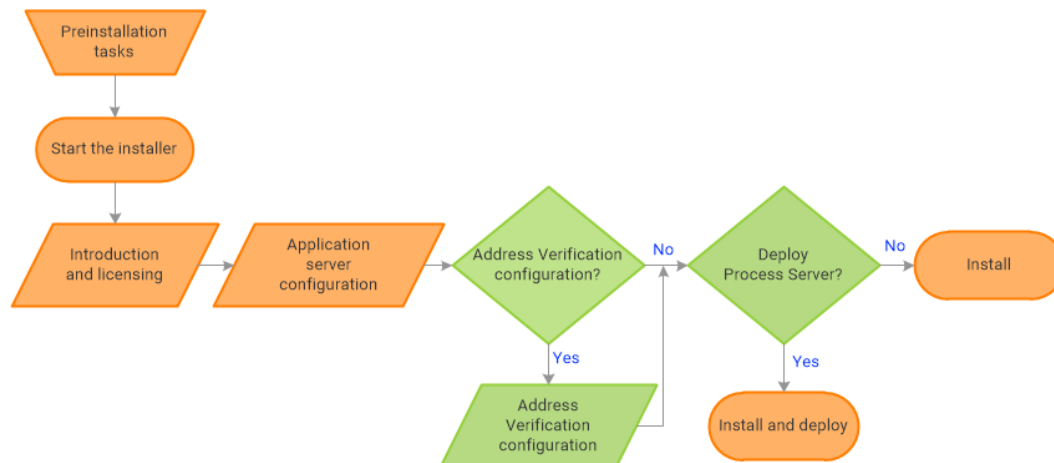
続行する準備ができたなら、次の手順を実行します。

1. インストールワークフローを確認します。
2. インストール中に入力するすべての値を収集します。
3. インストールウィザードまたはコマンドラインを使用するか、サイレントモードでインストールを開始します。

インストーラワークフローの確認

インストールウィザード、コマンドラインプロンプト、サイレントインストールスクリプトのいずれを使用しても、インストーラは同じ順序で手順を実行します。決定ポイントを念頭に置いて、インストールワークフローに従う必要があります。

次の図は、Process サーバーのインストーラワークフローの手順を示しています。



インストールプロセスの決定ポイントについて確認してください。実装プランを参照して、次の各決定ポイントで実行するパスを把握してください。

1. **Address Verification を設定しますか?** 管理者は、レコードに含まれるアドレスを解釈、処理、およびフォーマットするために Informatica Address Verification を使用します。承認済みユーザーは、マスターデータが更新される前に、レコードに含まれているアドレスを検証、修正、および認証できます。

注: サポートされているバージョンの Informatica Address Verification をすでにインストールしている場合は、設定およびパラメータファイルの場所と実行するアドレス処理のタイプを設定します。それ以外の場合は、デフォルトのファイルの場所を使用して、実行するアドレス検証のタイプを指定します。

2. **プロセスサーバーをデプロイしますか?** インストーラが postInstallSetup スクリプトを実行できるようにします。他の重要なタスクの中で、スクリプトはプロセスサーバーをアプリケーションサーバーにデプロイします。または、インストーラの終了後に postInstallSetup スクリプトを実行することもできます。

インストールの値の収集

インストーラを実行する前に、インストールの値を収集します。インストーラはアプリケーションサーバー、データベース管理システム、およびその他のコンポーネントに関する情報を求めるプロンプトを表示します。これらの表を印刷して環境に応じた値を追加することをお勧めします。

アプリケーションサーバー: IBM WebSphere

次の表を使用して、インストールに必要な WebSphere の詳細を収集します。

プロパティ	説明	デフォルト値	サーバーインスタンス1の値
WebSphere インストールディレクトリ	WebSphere がインストールされている場所。	-	
WebSphere セキュリティを有効にしますか?	WebSphere セキュリティが有効な場合、 【はい】 を選択し、ユーザー名およびパスワードを指定します。	いいえ	
サーバー名	プロセスサーバーをデプロイする WebSphere アプリケーションサーバーの名前。 クラスタ環境では、クラスタサーバー名、その対応するブートストラップポート情報、および SOAP コネクタポート情報のいずれかを入力します。	-	
ブートストラップポート	指定するサーバーにより使用されるブートストラップポート番号。 ヒント: ポート情報を検索するには、WebSphere 管理コンソールに移動し、 【アプリケーションサーバー】 > 【<server name>】 > 【ポート】 をクリックします。	2809	
SOAP コネクタポート	指定するサーバーにより使用される SOAP コネクタポート番号。 ヒント: ポート情報を検索するには、WebSphere 管理コンソールに移動し、 【アプリケーションサーバー】 > 【<server name>】 > 【ポート】 をクリックします。	8880	
プロファイル名	プロセスサーバーをデプロイする WebSphere アプリケーションサーバーを含む WebSphere プロファイルの名前。	-	
ユーザー名	管理特権のある WebLogic ユーザーの名前。	admin	
パスワード	WebSphere 管理者ユーザーのパスワード。	-	

クラスタ環境では、その他の WebSphere インスタンスの詳細を次の表に追加します。

プロパティ	サーバーインスタンス 2 の値	サーバーインスタンス 3 の値	サーバーインスタンス 4 の値
WebSphere インストールディレクトリ			
WebSphere セキュリティを有効にしますか?			
サーバー名			
ブートストラップポート			
SOAP コネクタポート			
プロファイル名			
ユーザー名			
パスワード			

Informatica Address Verification

注: Informatica Address Verification は、旧名 AddressDoctor です。

Informatica Address Verification をインストールする予定がある場合は、次のプロパティを記録します。

プロパティ	説明	デフォルト値	サーバーインスタンス 1 の値
構成ファイル	<p>Informatica Address Verification 構成ファイル SetConfig.xml の場所。このファイルを使用して、参照アドレスデータベース、Informatica Address Verification のロック解除コード、メモリ設定などの一般的な設定を行います。</p> <p>すでに Address Verification をインストールしている場合は、構成ファイルの場所を使用します。それ以外の場合は、デフォルトのファイルの場所を使用します</p>	<MDM Hub installation directory>\hub\cleanse\resources\AddressDoctor\5\SetConfig.xml	
パラメータファイル	<p>Informatica Address Verification パラメータファイル Parameters.xml の場所。このファイルを使用して、Informatica Address Verification のアドレス解釈、処理、およびフォーマットの方法を設定します。</p> <p>すでに Address Verification をインストールしている場合は、パラメータファイルの場所を使用します。それ以外の場合は、デフォルトのファイルの場所を使用します</p>	<MDM Hub installation directory>\hub\cleanse\resources\AddressDoctor\5\Parameters.xml	
修正タイプ	<p>実行するアドレス処理のタイプ。</p> <p>次のいずれかの修正タイプを使用します。</p> <ul style="list-style-type: none"> - PARAMETERS_DEFAULT。デフォルトの修正タイプ。Parameters.xml ファイルで定義済みの修正タイプの使用を示します。 - PARSE_ONLY。アドレス要素を解析して適切なフィールドに割り当てます。 - CORRECT_ONLY。アドレスを郵便データと照合して検証し、アドレスを修正します。 - CERTIFY_ONLY。各国ごとの郵便当局の要件を満たすために、郵便認証に従ってアドレスを検証します。 - CORRECT_THEN_CERTIFY。アドレスを郵便データと照合して検証し、アドレスを修正します。次に、各国ごとの郵便当局の要件を満たすために、郵便認証に従ってアドレスを検証します。 - TRY_CERTIFY_THEN_CORRECT。各国ごとの郵便当局の要件を満たすために、郵便認証に従ってアドレスを検証します。アドレスの検証に失敗した場合、処理によりアドレスが郵便データと照合、検証され、アドレスが修正されます。 	PARAMETERS_DEFAULT	

製品使用ツールキット

製品使用ツールキットにより、MDM 環境に関する情報が Informatica に送信されます。この情報は、トラブルシューティングに役立てるため、および使用環境に対する推奨事項を提供するために Informatica グローバルカスタマサポートによって使用されます。ツールキットによる Informatica への情報送信を希望しない場合は、MDM をインストールした後にツールキットを無効化することができます。

次の表を使用して、製品使用ツールキットのインストールに必要な詳細を収集します。

プロパティ	説明	デフォルト値	インストールの値
業種	組織のビジネスに最も一致する業種。	-	
環境	インストールする環境のタイプ。 コマンドラインからインストールする場合、次のいずれかの番号を入力します。 - 1. プロダクション環境 - 2. テストまたは QA 環境 - 3. 開発環境	-	
ネットワークにプロキシサーバーが設定されていますか?	設定されている場合、プロキシサーバーの詳細を指定します。	いいえ	
ホスト	プロキシサーバーの名前または IP アドレス。	-	
ポート	プロキシサーバーにより使用されるポート番号。	-	
ドメイン名	プロキシサーバーがドメインの一部である場合は、ドメインの名前。	-	
ユーザー名	安全なプロキシサーバーを使用する場合は、MDM にアクセスできるユーザーの名前。	-	
パスワード	ユーザーのパスワード。	-	

Process サーバーをインストールウィザードからインストールする

プロセスサーバーをグラフィカルモードでインストールする場合、インストールウィザードを使用します。インストールウィザードでは手順を追ってインストールすることができます。

Hub サーバーとプロセスサーバーをインストールする際には、同じユーザー名を使用する必要があります。

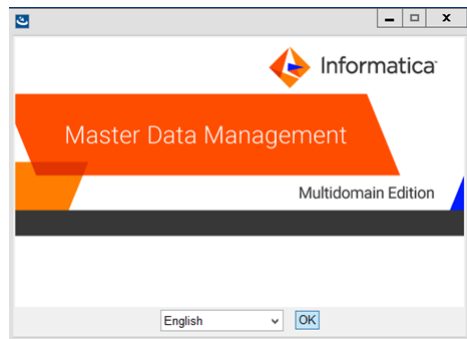
1. アプリケーションサーバーを起動します。
2. 次のディレクトリに移動します。

<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/<operating system name: オペレーティングシステム名>/mrncleanse

3. オペレーティングシステムに応じたタスクを実行することでインストーラを開始します。

- **UNIX**。コマンドラインから、`./hub_cleansetup_install.bin` を実行します。
- **Windows**。File Explorer から、`hub_cleansetup_install.exe` をダブルクリックします。

Process サーバーのインストールウィザードが開始します。



4. 言語を選択して、**[OK]** をクリックします。
【概要】ウィンドウが表示されます。
5. オンラインの指示に従います。プロンプトが表示されたら、収集したインストールの値を入力します。
6. インストールの最後に、**[設定のサマリ]** ウィンドウで選択したオプションを確認します。
7. 変更が必要な場合、**[前へ]** をクリックして、適切なウィンドウに戻ります。完了したら、**[次へ]** をクリックして最後のウィンドウに戻ります。
8. **[インストール]** をクリックします。
9. **[次の手順]** : 次の手順はインストーラから Process サーバーのデプロイを選択したかどうかによって異なります。
 - Process サーバーをインストーラからデプロイすることを選択した場合、Process サーバーをインストール後のタスクの一部としてデプロイする必要はありません。
 - Process サーバーを後でデプロイすることを選択した場合、Process サーバーをインストール後のタスクの一部としてデプロイする必要があります。

プロセスサーバーをコマンドラインからインストールする (UNIX のみ)

UNIX では、Process サーバーをコマンドラインからインストールできます。コマンドラインインストールを開始するスクリプトを実行します。

1. アプリケーションサーバーを起動します。
2. コマンドラインから次のディレクトリに移動します。
<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/<operating system name: オペレーティングシステム名>/mrmcleanup
3. 次のコマンドを実行します。
`./hub_cleansetup_install.bin -i console`
Process サーバーのインストールを要求するプロンプトが表示されます。

4. 収集したインストールの値を入力します。
括弧内に表示されたデフォルト値を使用するには、**Enter** を押します。
5. **【次の手順】**：インストールの完了後、次の手順は Process サーバーのデプロイを選択したかどうかによって異なります。
 - Process サーバーをインストーラからデプロイすることを選択した場合、Process サーバーをインストール後のタスクの一部としてデプロイする必要はありません。
 - Process サーバーを後でデプロイすることを選択した場合、Process サーバーをインストール後のタスクの一部としてデプロイする必要があります。

Process サーバーをサイレントモードでインストールする

サイレントモードでプロセスサーバーをインストールできます。サイレントインストールを開始する前に、サイレントインストールのプロパティファイルが設定されていることを確認します。

1. アプリケーションサーバーを起動します。
2. サイレントインストールプロパティファイルをターゲット環境にコピーします。
3. ターゲット環境で、オペレーティングシステムに応じたコマンドを実行します。
 - **UNIX**。 `/hub_install.bin -f <absolute path to edited installer properties file>`
 - **Windows**。 `hub_install.exe -f <absolute path to edited installer properties file>`自動インストーラがバックグラウンドで実行します。プロセスにしばらく時間がかかる場合があります。
4. インストーラによる Process サーバーのデプロイを選択した場合、`postinstallSetup.log` をチェックして、正常にインストールされたかどうかを確認します。
log ファイルは、次のディレクトリにあります。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/logs
5. **【次の手順】**：インストールの完了後、次の手順は Process サーバーのデプロイを選択したかどうかによって異なります。
 - Process サーバーをインストーラからデプロイすることを選択した場合、Process サーバーをインストール後のタスクの一部としてデプロイする必要はありません。
 - Process サーバーを後でデプロイすることを選択した場合、Process サーバーをインストール後のタスクの一部としてデプロイする必要があります。

クラスタ内のノードに Process サーバーをインストールする

アプリケーションサーバークラスタ環境では、Process サーバーアプリケーションをデプロイする必要があるクラスタのすべてのノードに、Process サーバーをインストールします。クラスタ内のあるノードでのインストールが完了してから、別のノードのインストールに進んでください。

Process サーバーがインストールされたディレクトリの構造がすべてのノードで確実に同じになるようにしてください。

1. WebServer クラスタを起動するには、次の手順を実行します。
 - a. WebSphere デプロイメントマネージャを開始します。
 - b. Process サーバーをインストールする WebServer クラスタのノードを開始します。
 - c. WebServer クラスタを開始します。

2. 次のディレクトリに移動します。

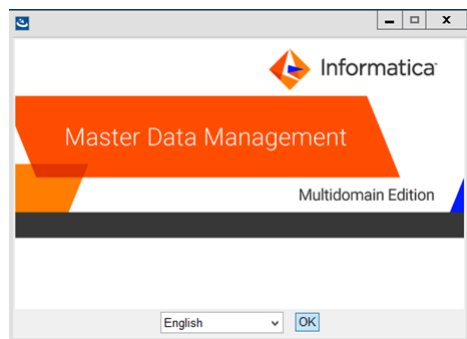
<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/<operating system name: オペレーティングシステム名>/mrncleanse

3. クラスタノードで Process サーバーインストーラを起動するには、オペレーティングシステムに応じたコマンドを実行します。

UNIX。 ./hub_cleanse_install.bin -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false

Windows。 hub_cleanse_install.exe -DSIPERIAN_INSTALL_PREREQ_VALIDATION=false

Process サーバーのインストールウィザードが開始します。



4. オンラインの指示に従います。プロンプトが表示されたら、収集したインストールの値を入力します。
5. **[次の手順]** : インストールの完了後、インストールを含むすべてのノードで Process サーバーを手動でデプロイします。

注: プライマリノードに対する自動デプロイメントを実行した場合、プライマリノードでの手動デプロイは必要ありません。

第 8 章

Process サーバーのインストール後のタスク

この章では、以下の項目について説明します。

- [インストールログファイルのコピー, 90 ページ](#)
- [バージョンとビルド番号の確認, 91 ページ](#)
- [WebSphere マルチノードまたはクラスタ環境用のプロセスサーバーの設定, 91 ページ](#)
- [プロセスサーバー EAR ファイルの再デプロイ, 92 ページ](#)
- [クラスローダーの設定, 92 ページ](#)
- [プロセスサーバーアプリケーションのデプロイ \(条件付き\), 93 ページ](#)
- [プロセスサーバーの保護された通信の有効化, 99 ページ](#)
- [Elasticsearch のインストールおよび設定, 100 ページ](#)
- [一致ポピュレーションの設定, 100 ページ](#)
- [プロセスサーバーとクレンジングエンジンの設定, 102 ページ](#)

インストールログファイルのコピー

インストールログファイルは、Process サーバーのインストールに失敗した場合に、問題を解決するのに役立ちます。ログファイルは、インストールドキュメントディレクトリにコピーしておきます。インストールの問題について問い合わせた場合は、Informatica グローバルカスタマサポートによりログファイルのコピーを求められることがあります。

次の表に、さまざまなタイプのインストールログファイルの説明を示します。

ログファイルのタイプ	説明
インストールログ	<ul style="list-style-type: none">- ファイル名。Informatica_MDM_Cleanse_Match_Server_Install_<timestamp>.xml- 場所。<MDM Hub installation directory>/hub/cleanse/UninstallerData/Logs- 内容。作成されたディレクトリ、インストールされたファイルおよび実行されたコマンドの名前、インストールされた各ファイルのステータス。
インストール前提条件ログ	<ul style="list-style-type: none">- ファイル名。installPrereq.log- 場所。<MDM Hub installation directory>/hub/cleanse/Logs- 内容。インストーラによって実行される前提条件チェックのログ。

ログファイルのタイプ	説明
デバッグログ	<ul style="list-style-type: none"> - ファイル名。infamdm_installer_debug.txt - 場所。<MDM Hub installation directory>/hub/cleanse/ - 内容。インストール中に行う選択、およびインストーラによって実行されるアクションに関する詳細情報。
インストール後のセットアップログ	<ul style="list-style-type: none"> - ファイル名。postInstallSetup.log - 場所。<MDM Hub installation directory>/hub/cleanse/logs - 内容。インストール後プロセス中にインストーラによって実行されるアクション、およびインストール後プロセスで発生するエラーのサマリ。
プロセスサーバーのログ	<ul style="list-style-type: none"> - ファイル名。cmxserver.log - 場所。<MDM Hub installation directory>/hub/cleanse/logs - 内容。プロセスサーバーの操作のサマリ。
WebSphere ログ	<ul style="list-style-type: none"> - ファイル名。startServer.log、stopServer.log、SystemErr.log、および SystemOut.log - 場所。<WebSphere installation directory>/profiles/AppSrv01/logs/<server name> - 内容。サーバーの開始および停止、パフォーマンスについての情報が含まれます。

バージョンとビルド番号の確認

正しいバージョンとビルド番号のプロセスサーバーがインストールされていることを確認します。

1. コマンドプロンプトを開き、次のディレクトリに移動します。<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/cleanse/bin
2. プロセスサーバーのバージョンとビルド番号を確認するには、次のコマンドを実行します。

UNIX の場合: versionInfo.sh

Windows の場合: versionInfo.bat

注: AIX システムの場合は、</jre/bin ディレクトリから Java を実行するように、Java home: Java homeversionInfo.sh スクリプトを変更します。

WebSphere マルチノードまたはクラスタ環境用のプロセスサーバーの設定

WebSphere マルチノードまたはクラスタ環境にプロセスサーバーをインストールした場合は、WebSphere 環境用にプロセスサーバーを設定します。WebSphere 環境用に Hub サーバーを設定するには、cmxcleanse.properties ファイルで cluster.flag プロパティを追加します。

1. 次のディレクトリにある cmxcleanse.properties ファイルを開きます。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/cleanse/resources
2. cluster.flag プロパティを追加します。
このプロパティは、クラスタリングを有効にするかどうかを指定します。クラスタリングを有効にするには、true に設定します。デフォルトは false です。

プロセスサーバー EAR ファイルの再デプロイ

postInstallSetup スクリプトを手動またはプロセスサーバーのインストールの一部として実行した後、WebSphere サーバー管理コンソールを使用して、プロセスサーバーの EAR ファイル siperian-mrm-cleanse.ear をデプロイ解除およびデプロイします。EAR ファイルは、プロセスサーバーのインストールディレクトリからデプロイする必要があります。

1. WebSphere サーバー管理コンソールにログインします。
2. siperian-mrm-cleanse.ear をデプロイ解除します。
3. siperian-mrm-cleanse.ear ファイルをデプロイします。

EAR ファイルは、次のディレクトリにあります。

<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/cleanse

次のデプロイメントオプションを設定します。

- **【アプリケーションのインストール準備】** パネルで、デプロイメントにデフォルトのバインディングを生成させるオプションを有効にします。
- **【モジュールのメタデータ】** で、モジュールが読み込まれるたびに注釈ベースのメタデータをスキャンする、siperian-cleanse-ejb.jar モジュールの metadata-complete 属性を無効にします。

アプリケーションのデプロイの詳細については、WebSphere Server サーバーのマニュアルを参照してください。

クラスローダーの設定

プロセスサーバーアプリケーションのクラスローダーを設定するには、WebSphere デプロイメントマネージャを使用します。

1. プロセスサーバーアプリケーションのクラスローダー siperian-mrm-cleanse.ear が、最後に親クラスローダーでクラスを読み込むように設定されていることを確認します。

最初に親クラスローダーでクラスを読み込むようにクラスローダーが設定されている場合は、アプリケーションのクラスローダーを設定します。

- a. **【アプリケーション】** > **【アプリケーションタイプ】** > **【WebSphere エンタープライズアプリケーション】** を選択します。
- b. **【エンタープライズアプリケーション】** ページで、いずれかのアプリケーションをクリックします。
- c. アプリケーションの設定ページで、**【クラスロードおよび更新の検出】** リンクをクリックします。
- d. **クラスローダーの設定** ページで、**【最初にローカルクラスローダーでロードしたクラス（親が最後）】** クラスローダーの順序オプションを選択します。
- e. **【適用】** をクリックしてから、**【OK】** をクリックします。

2. 次のアプリケーション EAR ファイルの Web モジュールのクラスローダーを設定します。

アプリケーション EAR ファイル	Web モジュール	クラスローダーの順序
siperian-mrm-cleanse.ear	siperian-mrm-cleanse.war	最初にローカルクラスローダーで読み込むクラス（親が最後）
siperian-mrm-cleanse.ear	MDESolr.war	最初に親クラスローダーで読み込むクラス

- 【アプリケーション】 > 【アプリケーションタイプ】 > 【WebSphere エンタープライズアプリケーション】 を選択します。
 - 【エンタープライズアプリケーション】 ページで、いずれかのアプリケーション EAR ファイルの名前をクリックします。
 - アプリケーションの設定ページで、【モジュールの管理】 リンクをクリックします。
 - モジュールのリストで、Web モジュールのリンクをクリックします。
 - Web モジュールの設定ページで、クラスローダーの順序を選択します。
 - 【適用】 をクリックしてから、【OK】 をクリックします。
3. WebSphere を再起動して、プロセスサーバーアプリケーションを起動します。

プロセスサーバーアプリケーションのデプロイ（条件付き）

プロセスサーバーアプリケーションのデプロイメントが必要なシナリオがある場合は、プロセスサーバーアプリケーションをデプロイします。

次のいずれかのシナリオでプロセスサーバーアプリケーションをデプロイする必要があります。

- インストールは、アプリケーションサーバーのマルチノード環境またはクラスタ環境にあります。
- インストールは完了したが、インストール中に実行した postInstallSetup スクリプトが失敗する。
- インストール中に postInstallSetup スクリプトをスキップした。

次の手順を実行して、プロセスサーバーアプリケーションをデプロイします。

- プロセスサーバーが Hub サーバーと同じアプリケーションサーバーインスタンスにインストールされていない場合は、データソースを作成します。
- プロセスサーバーアプリケーション siperian-mrm-cleanse.ear をデプロイします。
- クラスローダーを設定します。

手順 1. データソースの作成（条件付き）

Process サーバーが Hub サーバーと同じアプリケーションサーバーインスタンスにデプロイされていない場合は、アプリケーションサーバーのデータソースを設定します。

- JDBC ドライバをインストールします。
- MDM Hub マスタデータベースのデータソースを作成します。

3. オペレーショナル参照ストアのデータソースを作成します。

手順 1.JDBC ドライバのインストール

MDM Hub マスタデータベースとオペレーショナル参照ストア（ORS）のデータソースを作成する前に、JDBC ドライバをインストールします。

サポートされているバージョンの JDBC ドライバの取得方法については、Oracle にお問い合わせください。

- ▶ JDBC ドライバを次のディレクトリにコピーします。

<WebSphere installation directory>/lib

手順 2.MDM Hub マスタデータベースのデータソースの作成

プロセスサーバマシンに JDBC ドライバをインストールした後で、MDM Hub マスタデータベース用のデータソースを作成します。

1. WebSphere Application Server の管理コンソールを起動します。
2. ドライバライブラリの場所を指定します。
 - a. コンソールナビゲーションツリーで、**[環境]** を展開します。
 - b. **[WebSphere 変数]** リンクをクリックします。
 - c. JDBC 変数が次の JDBC ドライバディレクトリを参照するように更新します。
<WebSphere installation directory: WebSphere インストールディレクトリ>/lib
3. MDM Hub マスタデータベースのデータソースで使用するセキュリティアカウントを作成します。
 - a. コンソールのナビゲーションツリーで、**[セキュリティ]** を展開します。
 - b. **[セキュアな管理、アプリケーション、インフラストラクチャ]** リンクをクリックします。
 - c. **[認証]** で **[Java 認証および承認サービス]** を展開し、**[J2C 認証データ]** をクリックします。
 - d. **[新規]** をクリックし、以下のプロパティを指定します。

プロパティ	説明
エイリアス	MDM Hub マスタデータベースの名前。
ユーザー ID	MDM Hub マスタデータベースに接続するためのユーザー名。
パスワード	MDM Hub マスタデータベースにアクセスするためのパスワード。

- e. **[OK]** をクリックします。
4. JDBC プロバイダを作成します。
 - a. **[リソース]** > **[JDBC]** を展開し、**[JDBC プロバイダ]** をクリックします。
[JDBC プロバイダ] ページが表示されます。
 - b. データソースを使用するアプリケーションの範囲を選択します。

- c. **【新規】** をクリックし、以下のプロパティを指定します。

プロパティ	説明
データベースタイプ	データベースの種類。[Oracle] を選択します。
プロバイダタイプ	JDBC プロバイダのタイプ。[Oracle JDBC ドライバ] を選択します。
実装タイプ	データソース実装のタイプ。[XA データソース] を選択します。
名前	JDBC プロバイダの名前。名前を「Informatica MDM Oracle JDBC Provider (XA)」に変更します。

- d. **【次へ】** をクリックし、**【完了】** をクリックします。

5. MDM Hub マスタデータベースのデータソースを作成します。

- a. 作成した JDBC プロバイダをクリックします。
【設定】 ページが表示されます。
- b. **【追加プロパティ】** の下で、**【データソース】** をクリックします。
【データソース】 ページが表示されます。
- c. **【新規】** をクリックします。
- d. 以下のデータソースのプロパティを指定してください。

プロパティ	説明
名前	データソース名。「MDM Master Data Source」と指定します。
JNDI 名	JDBC データソースが関連付けられる場所への JNDI パス。jdbc/siperian-cmx_system-ds と指定します。 注: JNDI 名は小文字で指定する必要があります。
管理されるコンポーネントの認証エイリアス	マスタデータベースデータソースの認証エイリアス。[<host name>/cmx_system] を選択します。

- e. **【次へ】** をクリックし、**【完了】** をクリックします。

手順 3. オペレーショナル参照ストアのデータソースの作成

プロセスサーバーマシンに JDBC ドライバをインストールした後で、各オペレーショナル参照ストア用のデータソースを作成します。

- WebSphere Application Server の管理コンソールを起動します。
- ドライバライブラリの場所を指定します。
 - コンソールナビゲーションツリーで、**【環境】** を展開します。
 - 【WebSphere 変数】** リンクをクリックします。
 - JDBC 変数が次の JDBC ドライバディレクトリを参照するように更新します。
 <WebSphere installation directory: WebSphere インストールディレクトリ>/lib

3. オペレーショナル参照ストアで使用するセキュリティアカウントを作成します。
 - a. コンソールのナビゲーションツリーで、**【セキュリティ】**を展開します。
 - b. **【セキュアな管理、アプリケーション、インフラストラクチャ】**リンクをクリックします。
 - c. **【認証】**で**【Java 認証および承認サービス】**を展開し、**【J2C 認証データ】**をクリックします。
 - d. **【新規】**をクリックし、以下のプロパティを設定します。

プロパティ	説明
エイリアス	オペレーショナル参照ストアの名前。
ユーザー ID	オペレーショナル参照ストアに接続するためのユーザー名。
パスワード	オペレーショナル参照ストアにアクセスするためのパスワード。

- e. **【OK】**をクリックします。
4. JDBC プロバイダを作成します。
 - a. **【リソース】** > **【JDBC】**を展開し、**【JDBC プロバイダ】**をクリックします。
【JDBC プロバイダ】ページが表示されます。
 - b. データソースを使用するアプリケーションのスコープを選択します。
 - c. **【新規】**をクリックし、以下のプロパティを指定します。

プロパティ	説明
データベースタイプ	データベースの種類。 【Oracle】 を選択します。
プロバイダタイプ	JDBC プロバイダのタイプ。 【Oracle JDBC ドライバ】 を選択します。
実装タイプ	データソース実装のタイプ。 【XA データソース】 を選択します。
名前	JDBC プロバイダの名前。名前を「Informatica MDM Oracle JDBC Provider (XA)」に変更します。

- d. **【次へ】**をクリックし、**【完了】**をクリックします。
5. オペレーショナル参照ストアのデータソースを作成します。
 - a. 作成した JDBC プロバイダをクリックします。
【設定】ページが表示されます。
 - b. **【追加プロパティ】**の下で、**【データソース】**をクリックします。
【データソース】ページが表示されます。
 - c. **【新規】**をクリックします。

- d. 以下のデータソースのプロパティを指定してください。

プロパティ	説明
名前	データソース名。「MDM ORS Data Source」と指定します。
JNDI 名	JDBC データソースが関連付けられる場所への JNDI パス。jdbc/siperian-<Oracle host name>-<Oracle SID>-<Operational Reference Store name>-ds を指定します。 注: JNDI 名は小文字で指定する必要があります。
管理されるコンポーネントの認証エイリアス	マスターデータベースデータソースの認証エイリアス。<host name>/<Operational Reference Store name>を選択します。

- e. [次へ] をクリックし、[完了] をクリックします。

手順 2. プロセスサーバーアプリケーションのデプロイ（条件付き）

インストールがアプリケーションサーバーのマルチノード環境またはクラスタ環境内にある場合、あるいは postInstallSetup スクリプトがスキップされたか失敗した場合は、プロセスサーバーアプリケーションをデプロイします。

プロセスサーバーアプリケーションを、プロセスサーバーをインストールしたのと同じマシンにデプロイします。プロセスサーバーアプリケーションは、それに関連付けられているプロセスサーバーのインストールを見つけることができる必要があります。そのため、アプリケーション EAR ファイルを別のマシンへのデプロイメントのためにコピーしないでください。例えば、プロセスサーバーをテストマシンにインストールし、アプリケーションを本番マシンにデプロイしたとします。本番マシンにデプロイされたアプリケーションは、テストマシン上のインストールを見つけることができません。

次のいずれかの手順を使用して、プロセスサーバーアプリケーションをデプロイします。

デプロイメント用スクリプトの使用

postInstallSetup スクリプトを実行して、プロセスサーバーアプリケーションをデプロイします。

手動デプロイ

プロセスサーバーアプリケーションを手動でデプロイします。

プロセスサーバーアプリケーションのデプロイでのスクリプトの使用（条件付き）

PostInstallSetup スクリプトを実行して、プロセスサーバーアプリケーションをデプロイできます。

重要: インストールがアプリケーションサーバーのマルチノード環境またはクラスタ環境にある場合は、まず 1 つのノードにプロセスサーバーアプリケーションをデプロイし、次に他方のノードにプロセスサーバーアプリケーションをデプロイします。プロセスサーバーアプリケーションは、必ずプロセスサーバーをインストールしたのと同じマシンにデプロイしてください。

1. コマンドプロンプトを開き、次のディレクトリに移動します。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/cleanse
2. PostInstallSetup スクリプトを実行します。

UNIX の場合: `./postInstallSetup.sh`

注: WebSphere でセキュリティを有効にしている場合は、`postInstallSetup.sh - Dwebsphere.password=<WebSpherePassword>`を実行します。

Windows の場合: `postInstallSetup.bat`

注: WebSphere でセキュリティを有効にしている場合は、`postInstallSetup.bat - Dwebsphere.password=<WebSphere のパスワード>`を実行します。

3. `siperian-cleanse-ejb.jar` モジュールの注釈ベースのメタデータのスキャンを有効にするには、WebSphere サーバー管理コンソールを使用して、EAR ファイル `siperian-mrm-cleanse.ear` をデプロイ解除およびデプロイします。

詳細については、[「プロセスサーバー EAR ファイルの再デプロイ」 \(ページ 92\)](#)を参照してください。

プロセスサーバーアプリケーションの手動でのデプロイ (条件付き)

プロセスサーバーアプリケーションは手動でデプロイできます。プロセスサーバーアプリケーションは、プロセスサーバーのインストールディレクトリからデプロイする必要があります。

1. 既存のデプロイメントがある場合は、WebSphere Server 管理コンソールを使用して、`siperian-mrm-cleanse.ear` をデプロイ解除します。
2. WebSphere Server 管理コンソールを使用して、`siperian-mrm-cleanse.ear` ファイルをデプロイします。
デプロイメントファイルは、次のディレクトリにあります。
<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/cleanse
次のデプロイメントオプションを設定します。
 - **【アプリケーションのインストール準備】** パネルで、デプロイメントにデフォルトのバインディングを生成させるオプションを有効にします。
 - **【モジュールのメタデータ】** で、モジュールが読み込まれるたびに注釈ベースのメタデータをスキャンする、`siperian-cleanse-ejb.jar` モジュールの `metadata-complete` 属性を無効にします。
 - クラスターノードでデプロイする場合、**【インストールオプションの選択】** パネルで、アプリケーションを配布するデプロイメントのオプションを有効にします。
3. クラスターノードにデプロイした場合は、次の手順を実行します。
 - a. クラスター、デプロイメントマネージャ、およびノードを停止します。
 - b. ノード、デプロイメントマネージャ、クラスターの順に開始します。

アプリケーションのデプロイの詳細については、WebSphere Server サーバーのマニュアルを参照してください。

手順 3. クラスローダーの設定

プロセスサーバーアプリケーションを手動でデプロイした後、アプリケーションのクラスローダーを設定します。

1. プロセスサーバーアプリケーションのクラスローダーの `siperian-mrm-cleanse.ear` を設定します。
最初に親クラスローダーでクラスを読み込むようにクラスローダーが設定されている場合は、アプリケーションのクラスローダーを設定します。
 - a. **【アプリケーション】** > **【アプリケーションタイプ】** > **【WebSphere エンタープライズアプリケーション】** を選択します。
 - b. **【エンタープライズアプリケーション】** ページで、いずれかのアプリケーションをクリックします。

- c. アプリケーションの設定ページで、**【クラスロードおよび更新の検出】** リンクをクリックします。
 - d. **クラスローダー**の設定ページで、**【最初にローカルクラスローダーでロードしたクラス（親が最後）】** クラスローダーの順序オプションを選択します。
 - e. **【適用】** をクリックしてから、**【OK】** をクリックします。
2. 次のアプリケーション EAR ファイルの Web モジュールのクラスローダーを設定します。

アプリケーション EAR ファイル	Web モジュール	クラスローダーの順序
siperian-mrm-cleanse.ear	siperian-mrm-cleanse.war	最初にローカルクラスローダーで読み込むクラス（親が最後）
siperian-mrm-cleanse.ear	MDESolr.war	最初に親クラスローダーで読み込むクラス

- a. **【アプリケーション】** > **【アプリケーションタイプ】** > **【WebSphere エンタープライズアプリケーション】** を選択します。
 - b. **【エンタープライズアプリケーション】** ページで、いずれかのアプリケーション EAR ファイルの名前をクリックします。
 - c. アプリケーションの設定ページで、**【モジュールの管理】** リンクをクリックします。
 - d. モジュールのリストで、Web モジュールのリンクをクリックします。
 - e. Web モジュールの設定ページで、クラスローダーの順序を選択します。
 - f. **【適用】** をクリックしてから、**【OK】** をクリックします。
3. WebSphere を再起動して、プロセスサーバーアプリケーションを起動します。

プロセスサーバーの保護された通信の有効化

各プロセスサーバーには、署名済み証明書が必要です。Hub コンソールを使用して HTTPS プロトコルを有効化し、プロセスサーバーごとに保護されたポートを指定します。

1. 証明書ストアで、プロセスサーバー向けの署名済み証明書を作成します。
2. アプリケーションサーバーが証明書ストアにアクセスできることを確認します。
3. Hub コンソールにログインします。
4. オペレーショナルリファレンスストアデータベースを選択します。
5. 書き込みロックを取得します。
6. **【ユーティリティ】** ワークベンチで、**【プロセスサーバー】** を選択します。
7. プロセスサーバーを選択して、**プロセスサーバーの編集** アイコンをクリックします。
【プロセスサーバーの追加/編集】ダイアログボックスが表示されます。
8. **【ポート】** が保護されたポートであることを確認します。
9. **【保護された接続を有効にする（HTTPS）】** チェックボックスを選択します。

Process Server Information	
Server	
Port	8443
Enable Cleanse Operations	<input checked="" type="checkbox"/>
Threads for Cleanse Operations	2
Cleanse Mode	Online and Batch ▼
Enable Match Processing	<input checked="" type="checkbox"/>
Match Mode	Online and Batch ▼
Offline	<input type="checkbox"/>
Enable Batch Processing	<input checked="" type="checkbox"/>
Threads for Batch Processing	20
CPU Rating	1.0
Enable Search Processing	<input checked="" type="checkbox"/>
Enable Secured Connection (HTTPS)	<input checked="" type="checkbox"/>

10. [OK] をクリックします。
11. リストに表示されるその他のプロセスサーバーを確認します。

Elasticsearch のインストールおよび設定

検索を使用するには、MDM Hub 用に Elasticsearch をインストールおよび設定します。

検索のインストールおよび構成の詳細については、*Multidomain MDM の設定ガイド*の「Elasticsearch を使用した検索」の章を参照してください。

一致ポピュレーションの設定

一致ポピュレーションには、マッチプロセスに使用する標準ポピュレーションセットが含まれます。サポートされている国、言語、またはポピュレーションごとに標準ポピュレーションセットがあります。マッチルールに使用する一致ポピュレーションを有効にする必要があります。

一致ポピュレーションは、Informatica MDM Hub のインストールの *population.jsp* ファイルとして用意されています。ポピュレーション名は ysp ファイル名と同じです。Japanese ポピュレーションを追加する際に、Person_Name_Kanji 一致フィールドを使用する場合は、_Kanji をポピュレーション名に追加します。例えば、Japan_Kanji または Japan_i_Kanji になります。この場合、標準の Person_Name 一致フィールドは使用できません。

使用するポピュレーションには、SSA-Name3 バージョンの MDM Hub との互換性が必要です。追加のポピュレーションファイルが必要な場合、または更新されたポピュレーションファイルを新しいバージョンにアップグレードする必要がある場合は、Informatica グローバルカスタマサポートにお問い合わせください。この製品で要求する最初のポピュレーションファイルは無料です。他の国用のポピュレーションファイルが必要な場合や、MDM Hub の最新バージョンにアップグレードするために更新されたポピュレーションファイルが必要な場合があります。

一致ポピュレーションの有効化

一致ルールに使用する一致ポピュレーションを有効にする必要があります。

1. `<population>.ysp` ファイルを以下の場所にコピーします。
UNIX の場合:`<infamdm_install_directory>/hub/cleanse/resources/match`
Windows の場合:`<infamdm_install_directory>\hub\cleanse\resources\match`
2. C_REPOS_SSA_POPULATION メタデータテーブルで、ポピュレーションが登録されていることを確認します。
MDM Hub インストールのシードデータベースには、C_REPOS_SSA_POPULATION テーブルに登録されたいくつかのポピュレーションがありますが、有効にはなっていません。
3. C_REPOS_SSA_POPULATION テーブルにポピュレーションが含まれていない場合、それをテーブルに追加して有効にします。
ポピュレーション名は ysp ファイル名と同じです。例えば、ysp ファイル名が US.ysp の場合、ポピュレーション名は US です。
ポピュレーションをオペレーショナルリファレンスストアに追加するには、以下の手順を実行します。
 - a. ポピュレーションを追加するオペレーショナルリファレンスストアスキーマに接続します。
 - b. SQL*Plus で、以下のディレクトリにある `add_std_ssa_population.sql` スクリプトを実行します。
UNIX の場合:`<infamdm_install_directory>/server/resources/database/custom_scripts/oracle`
Windows の場合:`<infamdm_install_directory>\server\resources\database\custom_scripts\oracle`
 - c. 以下の表に説明する内容に従ってプロンプトに回答します。

プロンプト	説明
追加するポピュレーションを入力します	ポピュレーションの名前。
ROWID_SSA_POP の値を入力します (例: INFA.0001)。デフォルトは [INFA.0001]	C_REPOS_SSA_POPULATION メタデータテーブルの ROWID_SSA_POP カラムの一意の値。デフォルトは INFA.0001 です。

ポピュレーションは C_REPOS_SSA_POPULATION テーブルに登録されています。

- d. 次のコマンドを実行して、ポピュレーションを有効にします。
`UPDATE c_repos_ssa_population SET enabled_ind = 1 WHERE population_name = '<Your Population>';`
`COMMIT;`
4. ポピュレーションが有効化されたら、Process サーバーを再起動します。
 5. Hub コンソールにログインし、ポピュレーションが有効になっていることを確認します。
ポピュレーションは、ベースオブジェクトの【一致/マージ設定】ユーザーインターフェースに表示されます。

プロセスサーバーとクレンジングエンジンの設定

Process サーバーをインストールすると、Process サーバーを使用してクレンジングエンジンを設定できます。

クレンジングエンジンの設定に関する詳細については、『*Multidomain MDM のクレンジアダプタガイド*』を参照してください。

第 9 章

アプリケーションサーバーに対する ActiveVOS のインストール後のタスク

この章では、以下の項目について説明します。

- [WebSphere クラスタ環境での ActiveVOS のインストールおよびデプロイ, 103 ページ](#)
- [WebSphere 環境での信頼されたユーザーの作成, 107 ページ](#)
- [安全なプロファイルへのユーザーとグループの追加, 107 ページ](#)

WebSphere クラスタ環境での ActiveVOS のインストールおよびデプロイ

WebSphere クラスタ環境で、ActiveVOS コンソールを使用して MDM ID サービスプロバイダに接続するには、MDM Hub のインストール後に追加の手順を実行します。

Hub サーバー、埋め込み ActiveVOS およびプロセスサーバーのインストール後、次の手順を完了します。

1. WebSphere 作業マネージャの設定。
2. WebSphere タイムマネージャの設定。
3. JAAS アプリケーションログインの設定。
4. ActiveVOS サーバーおよび ActiveVOS Central のインストール。
5. ActiveVOS インストールファイルの編集。
6. ActiveVOS および Identity Resolution のデプロイ。

注: インストールおよびデプロイ手順は、WebSphere のバージョンによって異なる可能性があります。お使いの WebSphere のバージョンに特有の情報については、そのバージョン用の WebSphere のマニュアルを参照してください。

WebSphere 作業マネージャの設定

WebSphere 管理コンソールを使用して、ActiveVOS のエンタープライズおよびシステム作業マネージャを作成します。作業マネージャは、非同期 Bean を使用する ActiveVOS アプリケーションコンポーネント用のスレッドプールとして動作します。

1. WebSphere 管理コンソールから、[リソース] > [非同期 Bean] > [作業マネージャ] に移動します。
2. 次のプロパティを指定して作業マネージャを追加します。

プロパティ	作業マネージャ 1	作業マネージャ 2
名前	ActiveVOS エンタープライズ 作業マネージャ	ActiveVOS システム作業マネージャ
JNDI 名	wm/ActiveVOS	wm/ActiveVOSSystem
スレッドの最小数	10	(5 ページ)
スレッドの最大数	150	50
スコープ	クラスタ	クラスタ
サービスのセキュリティおよび 作業領域	有効	有効

WebSphere タイムマネージャの設定

WebSphere 管理コンソールを使用して、ActiveVOS のタイムマネージャを作成します。タイムマネージャは非同期 Bean を使用する ActiveVOS アプリケーションコンポーネント用のスレッドプールとして動作します。

1. WebSphere 管理コンソールから、[リソース] > [非同期 Bean] > [タイムマネージャ] に移動します。
2. 次のプロパティを指定してタイムマネージャを追加します。

プロパティ	値
名前	ActiveVOS Enterprise Time Manager
JNDI 名	tm/ActiveVOS
アラームスレッド数	10
スコープ	クラスタ
サービスのセキュリティおよび作業領域	有効

JAAS アプリケーションログインの設定

WebSphere 管理コンソールを使用して、ActiveBPELIdentityAssertion および ActiveBPELProvidedUser アプリケーションへのアプリケーションログインを設定します。

1. ActiveBPEL ID アサーション用の JAAS ログインモジュールを設定します。
 - a. WebSphere 管理コンソールから、**[セキュリティ]** > **[グローバルセキュリティ]** > **[JAAS 構成]** > **[アプリケーションログイン構成]** に移動します。
 - b. アプリケーションログインを追加し、エイリアスを ActiveBPELIdentityAssertion と指定します。
 - c. **[JAAS ログインモジュール]** セクションで、次のモジュールクラスを指定した順序で追加します。
 1. `com.activeee.rt.websphere.trustvalidation.AeIdentityAssertionLoginModule`
 2. `com.ibm.wsspi.security.common.auth.module.IdentityAssertionLoginModule`
 - d. 変更を保存します。
2. ActiveBPEL 指定ユーザー用の JAAS ログインモジュールを設定します。
 - a. アプリケーションログインを追加し、エイリアスを ActiveBPELProvidedUser と指定します。
 - b. **[JAAS ログインモジュール]** セクションで、次のログインモジュールクラスを追加します。
`com.activeee.rt.websphere.trustvalidation.AeBasicLoginModule`
 - c. モジュールに次のカスタムプロパティを追加します。

プロパティ	説明
username	ActiveVOS 管理者ユーザーのユーザー名。
パスワード	ActiveVOS 管理者ユーザーのパスワード。

- d. 変更を保存します。
3. ActiveVOS 管理者ユーザーに管理者ロールを追加します。
 - a. **[ユーザーおよびグループ]** で、**[管理者ユーザーロール]** > **[ユーザー]** をクリックします。
 - b. このユーザーに対して **[監視]** ロールを選択します。
 - c. ユーザーを **[使用可能]** リストから **[ロールにマップ]** リストに移動します。
 - d. **[OK]** をクリックします。

ActiveVOS サーバーおよび ActiveVOS Central のインストールの完了

WebSphere 環境を設定した後、ActiveVOS サーバーおよび ActiveVOS Central のインストールを完了するために、インストーラユーティリティを実行します。

1. コマンドプロンプトを開き、次のディレクトリに移動します。
`<MDM Hub installation directory>/<ActiveVOS directory>/server-enterprise/websphere_config/bin`
2. ActiveVOS サーバーおよび ActiveVOS Central のインストールを完了するには、次のインストーラユーティリティを実行します。
UNIX の場合: `./config_deploy.sh`
Windows の場合: `config_deploy.bat`
3. 画面の指示に従います。

ActiveVOS インストールファイルの編集

ActiveVOS をデプロイする前に、ActiveVOS インストールファイルを編集します。

1. 次のディレクトリに移動します。

<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/conf

2. avos.install.properties ファイルを編集して、install.web.application プロパティを false に設定します。

3. 次のディレクトリに移動します。

<MDM Hub installation directory>/avos/server/server-enterprise/websphere_config/bin

4. 次のコードのブロックをコメントアウトして、deployer.xml を編集します。

```
<!-- <target name="deploy.resources" depends="deploy.timer.manager,deploy.work.manager"/>
<target name="deploy.timer.manager">
<echo message="${basedir}"/>

<run.wsadmin script="${basedir}
/scripts/timermanager.jacl"/>
</target>
<target name="deploy.work.manager">
<run.wsadmin script="${basedir}/scripts/workmanager.jacl"/>

<run.wsadmin script="${basedir}
/scripts/systemworkmanager.jacl"/>
</target>
<target name="deploy.jaas">
<run.wsadmin script="${basedir}/scripts/jaaslogin.jacl"/>

</target>

<target name="deploy.apps">

<run.wsadmin script="${basedir}
/scripts/installapp.jacl"/>
</target> -->
```

ActiveVOS および Identity Resolution のデプロイ

ActiveVOS インストールファイルを編集した後、ActiveVOS サーバー、ActiveVOS Central および MDM Identity Resolution をデプロイします。

詳細については、WebSphere サーバーのマニュアルを参照してください。

1. 次のディレクトリに移動します。

<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server/bin

2. ActiveVOS サーバーおよび MDM Identity Resolution をデプロイするには、次のコマンドを実行します。

```
sip_ant.bat deploy_mdm_identity_resolution
sip_ant.bat deploy_avos_server
```

3. WebSphere 管理コンソールから、次の ActiveVOS アプリケーションをデプロイします。

- ave_websphere.ear
- activevos-central.war

ActiveVOS アプリケーションは次のディレクトリにあります: <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server

注: デプロイでデフォルトのバインドの生成を可能にするオプションが選択されていることを確認します。

WebSphere 環境での信頼されたユーザーの作成

ActiveVOS ワークフローエンジンを使用するには、信頼されたユーザーを作成して、abTrust、abServiceConsumer、および abTaskClient ロールにマップします。

この信頼されたユーザーは、Hub コンソールの ActiveVOS ワークフローアダプタユーザーと同じユーザーです。信頼されたユーザーの名前は、アプリケーションサーバーの管理者ユーザーと同じ名前にすることはできません。

1. WebSphere コンソールで、ave_websphere EAR アプリケーションを停止します。
2. 信頼されたユーザーを作成します。
3. ave_websphere.ear ファイルを開きます。
4. ave_websphere.ear ファイルで、信頼されたユーザーを abTrust、abServiceConsumer、および abTaskClient ロールにマップします。
5. WebSphere プロファイルを再起動します。

安全なプロファイルへのユーザーとグループの追加

MDM Hub の管理者およびユーザーのためにユーザーとグループを作成します。ユーザーとグループの作成方法の詳細については、WebSphere のマニュアルを参照してください。

1. WebSphere コンソールで、ActiveVOS サーバーで認証する MDM Hub の管理者およびユーザーのそれぞれに対し、ユーザーを作成します。
2. MDM Hub 管理者のグループを作成します。
3. MDM Hub ユーザーのグループを作成します。
4. MDM Hub 管理者グループに管理者を追加します。
5. MDM Hub ユーザーグループにユーザーを追加します。

第 10 章

ビジネスエンティティアダプタに対する ActiveVOS のインストール後のタスク

この章では、以下の項目について説明します。

- [ActiveVOS Web アプリケーション, 108 ページ](#)
- [ビジネスエンティティワークフローアダプタの ActiveVOS URN の設定, 109 ページ](#)
- [ActiveVOS URL のプロトコルの設定, 109 ページ](#)
- [ActiveVOS プロトコルの HTTPS への設定, 110 ページ](#)
- [プライマリワークフローエンジンの設定, 111 ページ](#)
- [ActiveVOS 用の MDM Identity Service の設定, 111 ページ](#)
- [タスクの設定, 112 ページ](#)

ActiveVOS Web アプリケーション

バンドルとしてライセンス供与された ActiveVOS サーバーのバージョンをインストールする場合には、2 つの ActiveVOS Web アプリケーションを使用できるライセンスも供与されています。それらのアプリケーションを使用するには、アプリケーションサーバーのコンテナにユーザーを追加します。

以下の Web アプリケーションがあり、これらはいくつかの目的に使用できます。

ActiveVOS コンソール

管理者は、ActiveVOS コンソールを使用して、デプロイされたプロセス、警告システム、およびエンドポイントの場所を管理します。また、パフォーマンスの監視および管理を行うようにエンジンを設定することもできます。

ActiveVOS Central

ビジネスユーザーは、ActiveVOS Central を使用して、タスク、要求、およびレポートを管理できます。ただし、通常、タスクの管理には Data Director (IDD) アプリケーションが使用されます。これは、確認対象のエンティティをタスクマネージャから開くことができるからです。

ActiveVOS Central を使用するには、MDM Hub ユーザーをアプリケーションサーバー コンテナに追加する必要があります。

これらの Web アプリケーションの詳細については、Informatica ActiveVOS のマニュアルを参照してください。

ビジネスエンティティワークフローアダプタの ActiveVOS URN の設定

ActiveVOS サーバーには、内部的に使用する 2 つの定義済みの Uniform Resource Name (URN) があります。URN マッピングの URL は、ActiveVOS サーバーが動作しているホスト名とポート番号に変更する必要があります。

1. ActiveVOS コンソールを起動します。ブラウザで、正しいホスト名とポート番号に置き換えて、以下の URL を入力します。
暗号化接続: `https://[host]:[port]/activevos`
非暗号化接続: `http://[host]:[port]/activevos`
2. ActiveVOS コンソールのホームページで、**[Administration] > [Configure Server] > [URN Mappings]** をクリックします。
3. 以下の URN について、ActiveVOS サーバーのホスト名とポート番号を反映するようにパスを更新します。

URN	URL パス
ae:internal-reporting	暗号化接続: <code>https://[host]:[port]/activevos/internalreports</code> 非暗号化接続: <code>http://[host]:[port]/activevos/internalreports</code>
ae:task-inbox	暗号化接続: <code>https://[host]:[port]/activevos-central/avc</code> 非暗号化接続: <code>http://[host]:[port]/activevos-central/avc</code>

4. **urn:mdm:service** が MDM Hub サーバーのホスト名とポート番号にマッピングされていることを確認します。
暗号化接続: `https://[host]:[port]/cmx/services/BeServices`
非暗号化接続: `http://[host]:[port]/cmx/services/BeServices`

ActiveVOS URL のプロトコルの設定

ActiveVOS URL のプロトコルは、`build.properties` ファイルで設定できます。

1. 次のディレクトリにある `build.properties` ファイルを検索します。
 - UNIX の場合: <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/bin
 - Windows の場合: <MDM Hub installation directory: MDM Hub のインストールディレクトリ>\hub\bin
2. 次のパラメータのプロトコルを `http` から `https` に変更します。
 - `activevos.mdm.sif.url`
 - `activevos.mdm.cs.url`
3. `build.properties` ファイルを保存します。
4. 次のディレクトリに移動します。
 - UNIX の場合: <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/server
 - Windows の場合: <MDM Hub installation directory: MDM Hub のインストールディレクトリ>\hub\server

5. 次のコマンドを実行して、Hub サーバーアプリケーションをデプロイし、セキュリティ設定に変更を適用します。

UNIX の場合:

WebLogic

```
patchInstallSetup.sh -Dweblogic.password=<WebLogic password: WebLogic パスワード> -  
Ddatabase.password=<your database password: データベースパスワード>
```

WebSphere

```
patchInstallSetup.sh -Ddatabase.password=<your database password: データベースパスワード>
```

JBoss

```
patchInstallSetup.sh -Ddatabase.password=<your database password: データベースパスワード>
```

Windows の場合:

WebLogic

```
patchInstallSetup.bat -Dweblogic.password=<WebLogic password: WebLogic パスワード> -  
Ddatabase.password=<your database password: データベースパスワード>
```

WebSphere

```
patchInstallSetup.bat -Ddatabase.password=<your database password: データベースパスワード>
```

JBoss

```
patchInstallSetup.bat -Ddatabase.password=<your database password: データベースパスワード>
```

注: UNIX でパスワードに感嘆符 (!) を含める場合、感嘆符 (!) の前にバックスラッシュ (\) を付ける必要があります。例えば、パスワードが!!cmx!!の場合は、\!!cmx\!!と入力します。

ActiveVOS プロトコルの HTTPS への設定

ActiveVOS と MDM Hub の間の安全な通信を有効にするには、Hub コンソールの Workflow Manager で HTTPS にプロトコルを設定します。

最初に、HTTPS 通信用のアプリケーションサーバーを設定する必要があります。

1. Hub コンソールを起動します。
2. 書き込みロックを取得します。
3. 設定ワークベンチの **[Workflow Manager]** をクリックします。
4. Workflow Manager で **[ワークフローエンジン]** タブをクリックします。
5. ActiveVOS ワークフローエンジンを選択し、**[編集]** ボタンをクリックします。
6. **[ワークフローの編集]** ダイアログボックスで、プロトコルを HTTPS に設定します。
7. WebLogic 環境では、**[ワークフローの編集]** ダイアログボックスで、abAdmin ロールに属するユーザーのユーザー名およびパスワードを入力します。

プライマリワークフローエンジンの設定

プライマリワークフローエンジンを設定するには、ビジネスエンティティに基づく ActiveVOS ワークフローのワークフローエンジンを追加します。セカンダリワークフローエンジンは、廃止されたワークフローエンジンを使用して既存タスクを処理する既存顧客のためにあります。

1. Hub コンソールの設定ワークベンチで **[Workflow Manager]** をクリックします。
2. 書き込みロックを取得します。
3. **[ワークフローエンジン]** タブを選択し、**[追加]** ボタンをクリックします。
4. **[ワークフローの追加]** ダイアログボックスで、ワークフローエンジンのプロパティを入力します。

以下の表に、ワークフローエンジンのプロパティを示します。

フィールド	説明
ワークフローエンジン	ワークフローエンジンの表示名
アダプタ名	ビジネスエンティティに基づいて ActiveVOS ワークフローアダプタに [BE ActiveVOS] を選択します。
ホスト	Informatica ActiveVOS インスタンスのホスト名。
ポート	Informatica ActiveVOS インスタンスのポート名。
ユーザー名	信頼されたユーザーのユーザー名。
パスワード	信頼されたユーザーのパスワード。
プロトコル	MDM Hub と ActiveVOS 間の通信プロトコル。プロトコルには http または https が使用できます。

5. **[OK]** をクリックします。

ActiveVOS 用の MDM Identity Service の設定

埋め込み ActiveVOS を使用している場合は、MDM Identity Service を使用するように ActiveVOS を設定してください。ActiveVOS 用の MDM Identity Service を設定するには、ActiveVOS コンソールを使用して、MDM Hub ワークフローエンジンユーザーのパスワードを Identity Service のパスワードに設定します。

1. ActiveVOS コンソールで、**[管理]** > **[サービスの設定]** > **[Identity Service]** を選択します。
2. **[プロバイダ設定]** セクションで、**[有効]** チェックボックスを有効にして、**[プロバイダタイプ]** から **[MDM]** を選択します。
3. **[接続]** タブで、ユーザー名を admin として MDM Hub ユーザーのパスワードを入力します。

注: 後で管理者ユーザーのパスワードを変更する場合、ActiveVOS ID サービス設定で新しいパスワードを入力する必要があります。

4. **[更新]** をクリックします。

5. ActiveVOS が admin ユーザーとして MDM Hub にログインできるかどうか、**【テストのユーザー】**として指定したユーザーのロールのリストを ActiveVOS が取得できるかどうかをテストします。
 - a. **【テスト】** タブを選択します。
 - b. **【テストのユーザー】** フィールドに、ロールに割り当てられた MDM Hub ユーザーを入力します。
 - c. **【テスト設定】** をクリックします。

注: オペレーショナル参照ストアが設定されていない場合、このテストは失敗し、テストのユーザーはロールに属さなくなる、またはそのロール名にスペースが格納されます。

タスクの設定

Informatica Data Director でタスクワークフローの使用を開始する前に、プロビジョニングツールでタスクテンプレート、タスクトリガ、およびタスクタイプを設定します。

詳細については、『*Multidomain MDM のプロビジョニングツールガイド*』を参照してください。

第 11 章

リソースキットのインストール

この章では、以下の項目について説明します。

- [MDM Hub サンプルオペレーショナル参照ストアの設定, 113 ページ](#)
- [Informatica MDM Hub サンプルオペレーショナルリファレンスストアの登録, 116 ページ](#)
- [グラフィカルモードでのリソースキットのインストール, 118 ページ](#)
- [コンソールモードでのリソースキットのインストール, 121 ページ](#)
- [サイレントモードでのリソースキットのインストール, 123 ページ](#)

MDM Hub サンプルオペレーショナル参照ストアの設定

MDM Hub サンプルオペレーショナルリファレンスストアを使用するには、それを設定しておく必要があります。MDM Hub サンプルオペレーショナル参照ストアの設定は、リソースキットをインストールする前に行います。MDM Hub サンプルオペレーショナル参照ストアを設定するには、オペレーショナル参照ストアを作成し、それに `mdm_sample` をインポートします。

1. 分散ディレクトリの以下の場所に移動します。

UNIX の場合: `<distribution directory>/database/bin`

Windows の場合: `<配布ディレクトリ>\database\bin`

2. 次のコマンドを実行します。

UNIX の場合: `./sip_ant.sh create_ors`

Windows の場合: `sip_ant.bat create_ors`

3. 表示されるプロンプトに回答します。

注: プロンプトでは、デフォルトのテキストが括弧内に表示されます。デフォルト値を使用して次のプロンプトに進むには、**Enter** キーを押します。

データベースタイプの入力 (ORACLE、MSSQL、DB2)

データベースタイプ。ORACLE を指定します。

Oracle **接続タイプ (SERVICE、SID) を入力。** [SERVICE]

接続タイプ。以下の値を使用する。

- SERVICE。Oracle に接続するサービス名を使用する。

- SID。Oracle に接続する Oracle システム ID を使用する。

デフォルトは SERVICE。

オペレーショナル参照ストアのデータベースホスト名を入力します。[localhost]

データベースをホストするマシンの名前。デフォルトは localhost です。

オペレーショナル参照ストアデータベースのポート番号を入力します。[1521]

データベースリスナーが使用するポート番号。デフォルトは 1521 です。

オペレーショナル参照ストアのデータベースサービス名を入力します。[orcl]

Oracle サービスの名前。このプロンプトは、選択した Oracle 接続タイプが SERVICE の場合に表示される。

オペレーショナル参照ストアデータベースの SID を入力します。[orcl]

Oracle システム ID の名前。このプロンプトは、選択した Oracle 接続タイプが SID の場合に表示される。

Oracle Net の接続 ID (TNS 名) を入力。[orcl]

Oracle TNS 名。デフォルトは orcl。

接続 URL。[jdbc:oracle:thin:@//<ホスト名>:<ポート>/<service_name または SID>]

データベース接続の接続 URL。

オペレーショナル参照ストアのデータベースユーザー名を入力します。[cmx_ors]

MDM Hub サンプルオペレーショナルリファレンスストアデータベースのユーザー名。デフォルトは cmx_ors。

オペレーショナル参照ストアのデータベースのユーザーパスワードを入力します。

MDM Hub サンプルオペレーショナルリファレンスストアユーザーのパスワード。

リストからロケール名を入力します (de、en_US、fr、ja、ko、zh_CN) [en_US]

オペレーティングシステムのロケール。デフォルトは en_US です。

DBA ユーザー名を入力。[SYS]

管理者ユーザーの名前。デフォルトは SYS です。

DBA のパスワードを入力します。

管理者ユーザーのパスワード。

MDM インデックステーブルスペース名の入力。[CMX_INDX]

オペレーショナル参照ストア用のインデックスコンポーネントが含まれているテーブルスペースの名前。デフォルトは CMX_INDX。

MDM 一時テーブルスペース名の入力。[CMX_TEMP]

オペレーショナル参照ストア用の一時コンポーネントが含まれているテーブルスペースの名前。デフォルトは CMX_TEMP。

Oracle 一時テーブルスペース名の入力。[TEMP]

Oracle 一時テーブルスペースの名前。デフォルトは TEMP。

4. オペレーショナル参照ストアを作成したら、次のディレクトリの sip_ant.log を確認します。

UNIX の場合: <distribution directory>/database/bin

Windows の場合: <配布ディレクトリ>\database\bin

sip_ant.log ファイルには、オペレーショナル参照ストアを作成するために sip_ant スクリプトを実行するときに発生するすべてのエラーが記録されます。

5. mdm_sample をインポートするには、次のコマンドを実行します。

UNIX の場合: ./sip_ant.sh import_schema

Windows の場合: sip_ant.bat import_schema

6. 表示されるプロンプトに回答します。

注: プロンプトでは、デフォルトのテキストが括弧内に表示されます。デフォルト値を使用して次のプロンプトに進むには、**Enter** キーを押します。

データベースタイプの入力 (ORACLE、MSSQL、DB2)

データベースタイプ。ORACLE を指定します。

Oracle 接続タイプ (SERVICE、SID) を入力。[SERVICE]

接続タイプ。以下の値を使用する。

- SERVICE。Oracle に接続するサービス名を使用する。
- SID。Oracle に接続する Oracle システム ID を使用する。

デフォルトは SERVICE。

オペレーショナル参照ストアのデータベースホスト名を入力します。[localhost]

データベースをホストするマシンの名前。デフォルトは localhost です。

オペレーショナル参照ストアデータベースのポート番号を入力します。[1521]

データベースリスナーが使用するポート番号。Default is 1521.

オペレーショナル参照ストアのデータベースサービス名を入力します。[orcl]

Oracle サービスの名前。このプロンプトは、選択した Oracle 接続タイプが SERVICE の場合に表示される。

オペレーショナル参照ストアデータベースの SID を入力します。[orcl]

Oracle システム ID の名前。このプロンプトは、選択した Oracle 接続タイプが SID の場合に表示される。

Oracle Net の接続 ID (TNS 名) を入力。[orcl]

Oracle TNS 名。デフォルトは orcl。

接続 URL。[jdbc:oracle:thin:@//<ホスト名>:<ポート>/<service_name または SID>]

データベース接続の接続 URL。

オペレーショナル参照ストアのデータベースユーザー名を入力します。[cmx_ors]

MDM Hub サンプルオペレーショナルリファレンスストアデータベースの名前。デフォルトは cmx_ors。

オペレーショナル参照ストアのデータベースのユーザーパスワードを入力します。

MDM Hub サンプルオペレーショナルリファレンスストアデータベースユーザーの名前。

リストからロケール名を入力します (de、en_US、fr、ja、ko、zh_CN) [en_US]

オペレーティングシステムのロケール。デフォルトは en_US です。

ZIP 形式のダンプファイルのパスを入力します。 [<配布ディレクトリ>\resources\database]

mdm_sample.zip ファイルのパス。

ZIP 形式のダンプファイルの名前を入力します。[mdm_sample.zip]

ZIP 形式のダンプファイルの名前。デフォルトは mdm_sample.zip です。

Informatica MDM Hub サンプルオペレーショナルリファレンスストアの登録

MDM Hub サンプルオペレーショナルリファレンスストアは設定後に登録する必要があります。MDM Hub サンプルオペレーショナルリファレンスストアを登録するには、Hub コンソールを使用します。

1. Hub コンソールを開始します。
[データベースの変更] ダイアログボックスが表示されます。
2. MDM Hub マスターデータベースを選択して、[接続] をクリックします。
3. 設定ワークベンチにあるデータベースツールを起動します。
4. [書き込みロック] > [ロックの取得] の順にクリックします。
5. [データベースの登録] ボタンをクリックします。

Informatica MDM Hub 接続ウィザードが表示され、データベースタイプの選択が求められます。

6. データベースのタイプを選択して [次へ] をクリックします。
7. データベースの接続プロパティを設定します。
 - a. Oracle の接続方式を選択して、[次へ] をクリックします。
次の表に、選択できる Oracle の接続方式を示します。

接続方式	説明
サービス	サービス名を使用して Oracle に接続します。
SID	Oracle のシステム ID を使用して Oracle に接続します。

[接続プロパティ] ページが表示されます。

- b. 選択する接続タイプの接続プロパティを指定し、[次へ] をクリックします。
以下の表に、接続プロパティの種類と説明を示します。

プロパティ	説明
データベース表示名	Hub コンソールに表示する必要があるオペレーショナルリファレンスストアの名前。
マシン識別子	Hub ストアインスタンスからのレコードを一意に識別するためにキーに付けられるプレフィックス。
データベースホスト名	Oracle データベースをホストするサーバーの IP アドレスまたは名前。
SID	サーバー上で実行される Oracle データベースのインスタンスを参照する Oracle システム識別子。 [SID] フィールドは、[SID] 接続タイプを選択した場合に表示される。

プロパティ	説明
サービス	Oracle データベースへの接続に使用する Oracle サービスの名前。 【サービス】 フィールドは、 【サービス】 接続タイプを選択した場合に表示される。
ポート	Oracle データベースサーバー上で実行される Oracle リスナの TCP ポート。デフォルトは 1521。
Oracle TNS 名	ネットワーク上で認識されているデータベースの名前（アプリケーションサーバーの TNSNAMES.ORA ファイルで定義）。 例: mydatabase.mycompany.com Oracle TNS 名は、Oracle データベースのインストール時に設定します。Oracle TNS 名の詳細については、Oracle のマニュアルを参照してください。
スキーマ名	オペレーショナルリファレンスストアの名前。mdm_sample を指定します。
パスワード	オペレーショナルリファレンスストアのユーザー名に関連付けられているパスワード。 Oracle の場合、パスワードでは大文字と小文字の区別がない。 デフォルトでは、これがオペレーショナルリファレンスストアを作成するときに指定するパスワードです。

【サマリ】 ページが表示されます。

- c. サマリを確認し、追加の接続プロパティを指定します。
以下の表に、設定可能な追加の接続プロパティを示します。

プロパティ	説明
接続 URL	接続 URL。接続ウィザードでは、デフォルトで接続 URL が生成されます。次のリストに、Oracle 接続タイプの接続 URL のフォーマットを示します。 サービス接続タイプ jdbc:oracle:thin:@//database_host:port/service_name SID 接続タイプ jdbc:oracle:thin:@//database_host:port/sid [サービス] 接続タイプの場合、カスタマイズし、後で別の接続 URL をテストするオプションがあります。
登録後データソースを作成する	登録後にアプリケーションサーバーのデータソースを作成する場合は選択します。 注: このオプションを選択しない場合、手動でデータソースを設定する必要があります。

- d. サービス接続タイプでは、デフォルトの URL を変更する場合には、**【編集】** ボタンをクリックし、URL を指定して **【OK】** をクリックします。
8. **【完了】** をクリックします。
【データベースの登録】 ダイアログボックスが表示されます。
9. **【OK】** をクリックします。
MDM Hub により、MDM Hub サンプルオペレーショナルリファレンスストアが登録されます。

10. 登録した MDM Hub サンプルオペレーショナルリファレンスストアを選択し、**【データベース接続のテスト】** ボタンをクリックしてデータベース設定をテストします。
[データベースのテスト] ダイアログに、データベース接続テストの結果が表示されます。
11. **【OK】** をクリックします。
オペレーショナルリファレンスストアが登録され、データベースとの接続がテストされます。

グラフィカルモードでのリソースキットのインストール

グラフィカルモードでリソースキットをインストールできます。

リソースキットをインストールする前に、MDM Hub をインストールおよび設定しておく必要があります。

1. アプリケーションサーバーを起動します。
2. コマンド プロンプトを開き、リソースキットインストーラに移動します。デフォルトでは、インストーラは以下のディレクトリにあります。

UNIX の場合:<配布ディレクトリ>/<オペレーティングシステム名>/mrmresourcekit

Windows の場合:<distribution directory>\windows\mrmresourcekit

3. 次のコマンドを実行します。
UNIX の場合:hub_resourcekit_install.bin
Windows の場合:hub_resourcekit_install.exe
4. インストールの言語を選択し、**【OK】** をクリックします。
【概要】 ウィンドウが表示されます。
5. **【次へ】** をクリックします。
【ライセンスキー】 ウィンドウが表示されます。
6. **【使用許諾契約に同意する】** オプションを選択し、**【次へ】** をクリックします。
【機能のインストール】 ウィンドウが表示されます。
7. インストールするリソースキットの機能を選択し、**【次へ】** をクリックします。

次のオプションを選択することができます。

サンプルスキーマ

MDM Hub サンプルスキーマリソースをインストールします。サンプルアプリケーションをインストールする前に、サンプルスキーマを作成して Hub サーバーに登録する必要があります。

サンプルとユーティリティ

サンプルアプリケーションとサンプルユーティリティをインストールします。

デプロイされるサンプルアプリケーションのリストは、次のディレクトリの build.properties ファイルに保存されています。

<Resourcekit_Home>\samples

SIF SDK および Javadoc

SIF SDK に関連する Javadoc、ライブラリ、およびリソースがインストールされます。

BPM SDK

BPM SDK に関連するリソースがインストールされます。

Jaspersoft

Jaspersoft インストーラがリソースキットのホームにコピーされます。

SSA-NAME3

SSA-NAME3 インストーラがリソースキットのホームにコピーされます。

MDM Hub でサンプルスキーマを作成、登録しておかなければならないというメッセージが表示されます。

8. **[OK]** をクリックします。

[インストールフォルダの選択] ウィンドウが表示されます。

9. リソースキットをインストールする場所を選択します。

- デフォルトの場所を選択するには、**[次へ]** をクリックします。
- パスを入力するには、インストールフォルダのパスを入力して **[次へ]** をクリックします。

注: ディレクトリまたはフォルダ名にスペースが含まれているパスを指定すると、インストールは失敗します。

- デフォルトのインストール場所に戻す場合は、**[デフォルトフォルダの復元]** をクリックする。
- 別の場所を指定する場合は、**[選択]** をクリックして **[次へ]** をクリックする。

UNIX の場合は、**[リンクフォルダの選択]** ウィンドウが表示されます。

Windows の場合は、**[ショートカットフォルダの選択]** ウィンドウが表示されます。

10. UNIX の場合は、リンクフォルダを選択するか、またはリンクを作成しないオプションを選択して、**[次へ]** をクリックします。Windows の場合は、製品アイコンを作成する場所を選択するか、または製品アイコンを作成しないオプションを選択します。

[設定の選択] ウィンドウが表示されます。

11. 設定オプションを選択し、**[次へ]** をクリックします。

以下のいずれかのオプションを選択することができます。

サンプルの設定

サンプルをインストールして設定します。

ソースのみ

サンプルのソースをインストールしますが、設定まではしません。

[サンプルの設定] を選択した場合は、**[リソースキットのアプリケーションサーバー]** ウィンドウが表示されます。**[ソースのみ]** を選択した場合は、**[インストール前のサマリ]** ウィンドウが表示されます。

12. **[リソースキットのアプリケーションサーバー]** ウィンドウで、リソースキットをインストールするアプリケーションサーバーを選択し、**[次へ]** をクリックします。

選択したアプリケーションサーバーの **[アプリケーションサーバー: ホーム]** ウィンドウが表示されます。

13. アプリケーションサーバーを設定します。

- a. WebSphere Application Server のパスを選択し、**[次へ]** をクリックします。

[警告] ウィンドウが表示されます。

- b. インストール条件を満たしていることを確認し、**[OK]** をクリックします。

[WebSphere セキュリティの選択] ウィンドウが表示されます。

c. WebSphere のセキュリティを有効にするかどうかを選択し、**【次へ】** をクリックします。

- **【いいえ】** を選択して **【次へ】** をクリックすると、**【WebSphere Application Server: ポート】** ウィンドウが表示されます。デフォルトは **【いいえ】** です。

サーバー名を設定し、WebSphere Application Server の RMI ポートと SOAP ポートを設定します。

- **【はい】** を選択して **【次へ】** をクリックすると、**【WebSphere Application Server: ポートおよびユーザークレデンシャル】** ウィンドウが表示されます。WebSphere のユーザー名と WebSphere のパスワードを指定します。

【Informatica MDM Hub サーバー】 ウィンドウが表示されます。

14. Hub サーバーのインストールの情報を入力し、**【次へ】** をクリックします。

次のフィールドに値を入力します。

サーバー名

Hub サーバーをホストするサーバーの名前。

サーバーの HTTP ポート

Hub サーバーのポート番号。

Informatica MDM の管理パスワード

MDM Hub にアクセスするためのパスワード。

MDM Hub ホームディレクトリ

Hub サーバーインストールのディレクトリ。

【リソースキットの ORS ID】 ウィンドウが表示されます。

15. リストからリソースキットの ORS ID を選択し、**【次へ】** をクリックします。

リストには、作成したオペレーショナル参照ストア ID が表示されています。サンプルスキーマに関連するオペレーショナル参照ストア ID を選択します。

サンプルスキーマを登録していない場合、サンプルスキーマのオペレーショナル参照ストア ID は表示されません。サンプルのオペレーショナル参照ストアを登録し、インストールを再起動します。

【デプロイメントの選択】 ウィンドウが表示されます。

16. 次のいずれかのオプションを選択し、**【次へ】** をクリックします。

このインストール中に実行する。

インストール中にリソースキットをデプロイし、設定します。

後でデプロイする。

後で手動でデプロイと設定を行う場合は、このオプションを選択します。

サンプルとユーティリティ機能のインストールが選択済みの場合は、このインストール手順でリソースキットをデプロイ、設定します。この手順でリソースキットをデプロイしない場合は、変更を行ったときに、リソースキットに付属している postInstallSetup スクリプトを使用してサンプルを再デプロイすることはできません。

インストール後のセットアップを手動で実行する場合は、後で postInstallSetup スクリプトを使用して EAR ファイルをデプロイすることはできません。インストールを変更する場合は、手動で EAR ファイルを編集、デプロイする必要があります。

【インストール前のサマリ】 ウィンドウが表示されます。

17. インストール前のサマリを確認してインストールの設定を確定し、**【インストール】** をクリックします。

インストールが完了すると、**【インストールの完了】** ウィンドウが表示されます。

18. **【完了】** をクリックしてリソースキットのインストーラを終了します。

コンソールモードでのリソースキットのインストール

コンソールモードでリソースキットをインストールできます。

リソースキットをインストールする前に、MDM_SAMPLE スキーマを登録しておきます。

1. アプリケーションサーバーを起動します。
2. MDM Hub ディストリビューション内の次のディレクトリに移動します。
UNIX の場合:<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/<operating system name: オペレーティングシステム名>/resourcekit
Windows の場合:<MDM Hub distribution directory: MDM Hub ディストリビューションディレクトリ>/windows/resourcekit
3. コマンドプロンプトで次のコマンドを実行します。
UNIX の場合:./hub_resourcekit_install.bin -i console
Windows の場合:hub_resourcekit_install.exe -i console
4. インストールで選択するロケールに対応する番号を入力し、**Enter** を押します。
インストールに関する概要情報が表示されます。
5. **Enter** キーを押します。
使用許諾契約が表示されます。
6. 使用許諾契約に目を通します。使用許諾契約の条項に同意する場合は **Y** と入力し、同意しない場合は **N** と入力してインストールプログラムを終了します。
7. **Enter** キーを押します。
前の手順で **Y** と入力した場合、インストールフォルダに関する情報が表示されます。
8. カンマで区切られたインストール対象のリソースキット機能の数を入力し、**Enter** キーを押します。
サンプルスキーマのインストールを要求するプロンプトが表示されます。
9. リソースキットをインストールするフォルダを選択します。
 - デフォルトのフォルダを選択する場合は、**Enter** キーを押します。
 - パスを変更する場合は、インストールフォルダの絶対パスを入力し、**Enter** キーを押します。
10. インストールフォルダの場所を確認します。インストールフォルダを確認して **[OK]** を入力するか、または **[キャンセル]** を入力してインストールフォルダを変更します。
11. **Enter** キーを押します。
リンク場所のオプションのリストが表示されます。
12. リンク場所のオプションを番号で入力します。
リンクファイルの場所を尋ねるプロンプトが表示されます。
13. リンクファイルの場所を絶対パスで入力し、**Enter** キーを押します。
ソースのサンプルの設定オプションが表示されます。
14. 設定オプションを入力し、**Enter** キーを押します。

オプション	説明
1	サンプルをインストールして設定する

オプション	説明
2	サンプルのソースをインストールしますが、設定まではしない

1を入力した場合は、アプリケーションサーバーのオプションのリストが表示されます。2を入力した場合は、インストール前のサマリが表示されます。

15. 1を入力した場合は、選択するアプリケーションサーバーの数を入力し、**Enter** キーを押します。
アプリケーションサーバーの情報を要求するプロンプトが表示されます。
16. WebSphere 設定を設定します。
 - a. アプリケーションサーバーのインストールディレクトリを指定し、**Enter** キーを押します。
インストーラで、WebSphere の JDBC ドライバの前提条件が表示されます。
 - b. データベースの JDBC ドライバファイルの場所を確認し、**Enter** キーを押します。
データベースの JDBC ドライバファイルのコピーが<WebSphere_install_dir>/AppServer/lib ディレクトリに作成されます。WebSphere セキュリティの選択の情報が表示されます。
 - c. **【いいえ】**を選択すると、WebSphere アプリケーションサーバーのポート情報が表示されます。**【はい】**を選択すると、WebSphere アプリケーションサーバーのポート情報とユーザークレデンシャル情報が表示されます。
 - **【いいえ】**を選択する場合、サーバー名、RMI ポート、SOAP ポート、およびプロファイル名を入力するか、デフォルト値を受け入れて、**Enter** キーを押します。
 - **【はい】**を選択する場合、サーバー名、RMI ポート、SOAP ポート、プロファイル名、ユーザー名、およびパスワードを入力するか、デフォルト値を受け入れて、**Enter** キーを押します。
Hub サーバーの情報を要求するプロンプトが表示されます。
17. Hub サーバーのインストールの情報を入力し、**Enter** キーを押します。
以下の表で、Hub サーバーのインストール情報を要求するプロンプトについて説明します。

プロンプト	説明
サーバー名	Hub サーバーをホストするサーバーの名前。
サーバーの HTTP ポート	Hub サーバーのポート番号。
Informatica MDM の管理パスワード	MDM Hub にアクセスするためのパスワード。
MDM Hub ホームディレクトリ	Hub サーバーのインストール用ディレクトリ

MDM Hub ORS ID のリストが表示されます。

18. MDM サンプルスキーマのオペレーショナル参照ストア ID を入力し、**Enter** キーを押します。
サンプルスキーマを登録していない場合、サンプルスキーマのオペレーショナル参照ストア ID は表示されません。サンプルのオペレーショナル参照ストアを登録し、インストールを再起動します。
デプロイメントの選択プロンプトが表示されます。
19. インストール中に postInstallSetup を実行するか、後で手動で実行するか、いずれかの方法を選択します。
20. **Enter** キーを押します。
インストールの設定のサマリが表示されます。

21. インストール前のサマリに表示された情報を確認します。設定内容に問題がなければ、**Enter** キーを押してインストールを開始します。
指定した設定情報に従ってリソースキットがインストールされます。プロセスが完了すると、インストールの完了に関する情報が表示されます。
22. **Enter** キーを押してインストーラを終了します。

サイレントモードでのリソースキットのインストール

サイレントモードでは、ユーザーとの対話なしでリソースキットをインストールできます。複数のインストールや、マシクラスタにインストールをする必要があるときは、サイレントモードでの実行をお勧めします。サイレントインストールでは、進捗や失敗に関するメッセージが表示されません。

リソースキットのサイレントインストールを実行する前に、サイレントインストールのプロパティファイルを設定する必要があります。インストーラによりこのファイルが読み込まれ、インストールオプションが決定されます。サイレントインストールのプロセスは、不正なアプリケーションサーバーのパスまたはポートなど、設定が正しくない場合でも正常に完了する場合があります。プロパティファイルに正しい設定がされていることを確認する必要があります。

リソースキットのインストール先となるマシンのハードディスクに、リソースキットのインストールファイルをコピーします。サイレントモードでインストールするには、以下のタスクを完了します。

1. インストールプロパティファイルを設定し、そのプロパティファイル内でインストールオプションを指定する。
2. インストールプロパティファイルを使用してインストーラを実行する。

プロパティファイルの設定

Informatica では、インストーラが必要とするパラメータを含むサンプルのプロパティファイルが提供されています。このサンプルのプロパティファイルをカスタマイズして、インストールのオプションを指定できます。次に、サイレントインストールを実行します。

自動インストーラは、プロパティファイルの設定を検証しません。自動インストーラを実行する前に、正しい設定を指定したことを確認し、検証する必要があります。

1. 次のディレクトリにある `silentInstallResourceKit_sample.properties` ファイルを検索します。
UNIX の場合: `/silent_install/mrmresourcekit`
Windows の場合: `\silent_install\mrmresourcekit`
ファイルをカスタマイズしたら、保存します。ファイルの名前を変更し、マシン上のどこかに保存しておくことができます。
注: サイレントプロパティファイルでは、スラッシュおよびバックスラッシュが特殊文字として扱われます。インストールパスなどの情報をこのファイルに入力するときは、これらの文字のいずれかを 2 つ入力する必要があります。例えば、サーバーディレクトリのパスを入力するには、`\\u1\infamd\hub\resourcekit` と入力する必要があります。
2. `silentInstallResourceKit_sample.properties` ファイルのバックアップコピーを作成します。
3. テキストエディタを使用してファイルを開き、インストールパラメータの値を変更します。
4. プロパティファイルを `silentInstallresourcekit.properties` などの新しい名前で作成して保存します。

以下の表に、変更可能なインストールパラメータを示します。

プロパティ名	説明
INSTALLER_UI	インストールのモードを指定します。silent に設定。
SIP.INSTALL.TYPE	インストールのタイプを指定します。SIPERIAN_SAMPLE_INSTALL に設定。
SIP.INSTALL.SAMPLE.SCHEMA	サンプルスキーマをインストールするかどうかを指定します。 次のいずれかの値を指定。 - 0. サンプルスキーマをインストールしない - 1. サンプルスキーマをインストールする
SIP.INSTALL.SAMPLES	サンプルとユーティリティをインストールするかどうかを指定します。 次のいずれかの値を指定。 - 0. サンプルとユーティリティをインストールしない - 1. サンプルとユーティリティをインストールする
SIP.INSTALL.SIF.SDK	サービス統合フレームワーク（SIF）SDK をインストールするかどうかを指定します。 次のいずれかの値を指定。 - 0. SIF SDK をインストールしない - 1. SIF SDK をインストールする
SIP.INSTALL.BPM.SDK	BPM SDK をインストールするかどうかを指定します。 次のいずれかの値を指定。 - 0. BPM SDK をインストールしない - 1. SIF SDK をインストールする
SIP.INSTALL.JASPERSOFT	Jaspersoft レポートツールをインストールするかどうかを指定します。 次のいずれかの値を指定。 - 0. BPM SDK をインストールしない - 1. SIF SDK をインストールする
SIP.INSTALL.SSASNAME3	SSA-NAME3 をインストールするかどうかを指定します。 次のいずれかの値を指定。 - 0. SSA-NAME3 をインストールしない - 1. SSA-NAME3 をインストールする
USER_INSTALL_DIR	リソースキットのインストール先ディレクトリ。C:\<infamdm_install_directory>\hub\resourcekit など。
RUN_CONFIGURE_FLAG	サンプルを設定するかどうかを指定します。 - 0. サンプルを設定しない - 1. サンプルをインストールして設定する デフォルトは1です。 RUN_CONFIGURE_FLAG プロパティを1に設定する場合は、RUN_CONFIGURE_SETUP プロパティをコメントアウトするか、0に設定します。 サンプルを設定する場合は、必ずアプリケーションサーバーと Hub サーバーを起動し、Hub コンソールでサンプルスキーマを登録しておきます。

プロパティ名	説明
RUN_CONFIGURE_SETUP	<p>サンプルのソースのみを設定するかどうかを指定します。</p> <ul style="list-style-type: none"> - 0. サンプルのソースをインストールしない - 1. サンプルのソースをインストールする <p>RUN_CONFIGURE_SETUP プロパティを 1 に設定する場合は、RUN_CONFIGURE_FLAG プロパティをコメントアウトするか、0 に設定します。 RUN_CONFIGURE_SETUP プロパティを 1 に設定する場合には、後からサンプルを設定、デプロイすることはできません。</p>
SIP.AS.CHOICE	アプリケーションサーバーの名前。WebSphere を指定する。
SIP.AS.HOME	WebSphere インストールディレクトリへのパス。
SIP.AS.SERVER	サーバーの名前。
SIP.AS.PROFILENAME	アプリケーションサーバーのプロファイル名。
SIP.AS.PORT_2	RMI ポート番号を指定します。
SIP.AS.PORT_3	SOAP ポート番号を指定します。
SIP.WEBSPHERE.SECURITY.ENABLED.Yes=1	<p>WebSphere のセキュリティが有効な場合、このプロパティを設定します。</p> <p>SIP.WEBSPHERE.SECURITY.ENABLED.Yes=1 であれば、以下のプロパティを設定します。</p> <ul style="list-style-type: none"> - SIP.APPSERVER.USERNAME - SIP.APPSERVER.PASSWORD
SIP.APPSERVER.USERNAME	WebSphere にアクセスするために必要なユーザー名。
SIP.APPSERVER.PASSWORD	WebSphere にアクセスするために必要なパスワード。
SIP.SERVER.NAME	Hub サーバーがデプロイされているサーバーの名前。
SIP.SERVER.HTTP.PORT	Hub サーバーがリスンするポート。
SIP.ADMIN.PASSWORD	Hub サーバーにアクセスするためのパスワード。
HUB_SERVER_HOME	Hub サーバーインストールのディレクトリ。
SIP.ORS.ID	MDM Hub サンプルスキーマのオペレーショナル参照ストア ID。
RUN_DEPLOYMENT_FLAG	<p>サイレントインストール中の postInstallSetup スクリプトの実行。</p> <ul style="list-style-type: none"> - 0. postInstallSetup スクリプトを実行しない - 1. postInstallSetup スクリプトを実行する

サイレントインストーラの実行

プロパティファイルの設定後、サイレントインストールを開始できます。

1. アプリケーションサーバーが動作していることを確認します。
2. コマンドウィンドウを開きます。

3. 次のコマンドを実行します。

UNIX の場合: `./hub_resourcekit_install.bin -f <location_of_silent_properties_file_for_resourcekit>`

Windows の場合: `.\hub_resourcekit_install.exe -f <location_of_silent_properties_file_for_resourcekit>`

サイレントインストーラがバックグラウンドで実行されます。このプロセスには時間がかかる場合があります。postinstallSetup.log ファイルを確認して、インストールが成功したことを検証します。

ログファイルは、以下のディレクトリにあります。

UNIX の場合: `<infadm_install_directory>/logs/postInstall.log`

Windows の場合: `<infadm_install_directory>\logs\postInstall.log`

第 12 章

リソースキットのインストール後の作業

この章では、以下の項目について説明します。

- [sip_ant スクリプトの編集, 127 ページ](#)
- [postInstall スクリプトの手動実行, 127 ページ](#)

sip_ant スクリプトの編集

インストールタスクを実行したら、sip_ant スクリプトを編集します。

1. テキストエディタで sip_ant スクリプトを開きます。
sip_ant スクリプトは次のディレクトリにあります。
UNIX の場合: <Resource Kit install directory>/deploy/bin
Windows の場合: <Resource Kit install directory>\deploy\bin
2. 次と同様の行を見つけます。
UNIX の場合: "\$JAVA_HOME/bin/java" \$USER_INSTALL_PROP -Xms128 -Xmx1024m -m -classpath "%WAS_CLASSPATH%";
Windows の場合: "%JAVA_HOME%\bin\java" %USER_INSTALL_PROP% -Xms128 -Xmx1024m -m -classpath "%WAS_CLASSPATH%";
3. 次と同様のコードで置換し、JAVA_HOME を設定します。
UNIX の場合: "\$JAVA_HOME/bin/java" -Djava.endorsed.dirs="<WebSphere installation directory>\endorsed_apis" \$USER_INSTALL_PROP -Xms128 -Xmx1024m -m -classpath "%WAS_CLASSPATH%";
Windows の場合: "%JAVA_HOME%\bin\java" -Djava.endorsed.dirs="<WebSphere installation directory>\endorsed_apis" %USER_INSTALL_PROP% -Xms128 -Xmx1024m -m -classpath "%WAS_CLASSPATH%";
4. 変更を保存し、sip_ant スクリプトを閉じます。

postInstall スクリプトの手動実行

すべての必要なデプロイメントファイルをアプリケーションサーバーに確実にデプロイするには、手動で postInstall スクリプトを実行します。

1. コマンドプロンプトを開きます。

2. 次のディレクトリにある postInstallSetup スクリプトに移動します。

UNIX の場合: <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/hub/resourcekit/deploy

Windows の場合:<MDM Hub installation directory: MDM Hub のインストールディレクトリ>\hub\resourcekit\deploy

3. 次のコマンドを実行します。

UNIX の場合: postInstall.sh -Ddatabase.password=<MDM Hub Master Database Password>

注: パスワードに感嘆符 (!) が含まれている場合、感嘆符の前にバックスラッシュを付ける必要があります。例えば、パスワードが!!cmx!!の場合、そのパスワードを「\!cmx\!」と入力します。

Windows の場合:postInstall.bat -Ddatabase.password=<MDM Hub Master Database Password>

注: WebSphere でセキュリティを有効にしている場合は、postInstallSetup スクリプトに-Dwebsphere.password=<Secure WebSphere Password>のオプションを付けて実行します。

第 13 章

MDM Hub のトラブルシューティング

- [インストールプロセスのトラブルシューティング, 129 ページ](#)

インストールプロセスのトラブルシューティング

インストールが失敗した場合は、次の情報を利用してエラーのトラブルシューティングを行います。

Hub コンソールを起動できない

インストール後、Hub コンソールが起動に失敗し、ログファイルに次のエラーが生成されます。

SIP-09131: General Decryption failure and [ERROR] com.delos.util.StringUtil: Unable to decrypt

MDM Hub マスタデータベースのパスワードまたはオペレーショナル参照ストアのパスワードを暗号化して更新します。

1. データベーススキーマパスワードを暗号化するには、コマンドプロンプトで以下のコマンドを実行します。

```
java -classpath siperian-api.jar;siperian-common.jar;siperian-server.jar  
com.delos.util.PublicKeyBasedEncryptionHelper <plain text password> <Hub Server  
installation directory>
```

結果は端末のウィンドウにエコーされます。

2. マスタデータベースのパスワードまたはオペレーショナル参照ストアのパスワードを更新するには、cmx_system ユーザーとして接続して次の文を実行します。

```
UPDATE C_REPOS_DATABASE SET PASSWORD = '<new_password>' WHERE USER_NAME =  
<user_name>;  
COMMIT;
```

3. postInstallSetup スクリプトを実行します。

デフォルトディレクトリにアプリケーションサーバープロファイルがインストールされていない。

アプリケーションサーバープロファイルをデフォルトディレクトリにインストールしなかった場合、Hub サーバーおよび Process サーバーのアプリケーションを postInstallSetup スクリプトでアプリケーションサーバーにデプロイすることができません。

次のデフォルトディレクトリを使用します。

UNIX の場合: <Websphere_install_home>/profiles

Windows の場合: <Websphere_install_home>\profiles

問題を解決するには、Hub サーバーと Process サーバーの EAR ファイルを再パッケージしてから、手動で Hub サーバーおよび Process サーバーアプリケーションをアプリケーションサーバーがインストールされているカスタムディレクトリにデプロイします。

ディレクトリに siperian-mrm.ear ファイルがあると PostInstallSetup スクリプトが失敗する

Hub サーバーアプリケーションを、siperian-mrm.ear という名前のファイルが含まれているディレクトリにデプロイしようとすると、次のエラーが表示されます。

```
[wsadmin] ADMA5016I: Installation of siperian-mrm.ear started.
```

```
[wsadmin] A composition unit with name siperian-mrm.ear already exists. Select a different application name.
```

問題を解決するには、siperian-mrm.ear ファイルが含まれているすべてのディレクトリを削除してから postinstallSetup を再度実行して、ear ファイルをデプロイします。

注: Hub サーバーアプリケーションをデプロイ解除しても、siperian-mrm.ear ファイルがアプリケーションサーバーディレクトリにまだ存在していることがあります。

処理がタイムアウトすると、PostInstallSetup スクリプトが失敗します。

Hub サーバーを WebSphere 環境にインストールするときに、インストール後のセットアッププロセスが失敗して、次のエラーが表示されます。

```
[wsadmin] Starting siperian-mrm.ear ...  
[wsadmin] WASX7017E: Exception received while running file "wsinstall.jacl"; exception information:  
com.ibm.websphere.management.exception.ConnectorException  
[wsadmin] org.apache.soap.SOAPException: [SOAPException: faultCode=SOAP-ENV:Client; msg=Read timed out;  
targetException=java.net.SocketTimeoutException: Read timed out]
```

この問題は、SOAP 要求がタイムアウトしたときに発生します。

この問題を解決するには、次の手順を実行します。

1. 次のディレクトリに移動します。
 <WebSphere プロファイルルートディレクトリ>/properties
2. soap.client.props ファイルで、com.ibm.SOAP.requestTimeout プロパティの値を増やします。
3. WebSphere を再起動し、postInstallSetup スクリプトを再度実行します。

MDM Hub ユーザーがログインできない

Hub サーバーのインストール後に CMX_SYSTEM スキーマを再作成すると、MDM Hub はハッシュされたパスワードを認識できません。その結果、ユーザーは MDM Hub にログインできません。

この問題を解決するには、postInstallSetup スクリプトを再度手動で実行します。このスクリプトにより、MDM Hub ユーザーのパスワードが再度ハッシュされ、ユーザーがログインできるようになります。

または、postInstallSetup スクリプトを再実行しない場合は、次のコマンドを実行して、ユーザーパスワードをハッシュされたパスワードに移行し、アプリケーションユーザーを作成します。

UNIX の場合:

```
cd <MDM Hub installation directory: MDM Hub のインストールディレクトリ>/server/bin  
./sip_ant.sh hash_users_passwords  
./sip_ant.sh add_application_users
```

Windows の場合:

```
cd <MDM Hub installation directory: MDM Hub のインストールディレクトリ>\server\bin  
sip_ant.bat hash_users_passwords  
sip_ant.bat add_application_users
```

注: WebSphere 環境では、MDM Hub ユーザーは、次のディレクトリに対するアクセス権限と書き込み権限を持っている必要があります。

<MDM Hub installation directory: MDM Hub のインストールディレクトリ>/server/bin/resources/certificates

ActiveVOS サーバーのデプロイメントがタイムアウトすると、PostInstallSetUp スクリプトが失敗する

Hub サーバーをインストールする際、インストール後のセットアッププロセスが ActiveVOS サーバーのデプロイ時に失敗することがあります。

この問題を解決するには、次のディレクトリの build.properties ファイルに含まれる deploy.wait.time プロパティの値を大きくします。

UNIX の場合: <infamdm のインストールディレクトリ>/hub/server/bin

Windows の場合: <infamdm のインストールディレクトリ>\hub\server\bin

オペレーショナル参照ストア開始または終了のタイムスタンプが正確でない

作成したオペレーショナル参照ストアをエクスポートするとき、MDM Hub は .dmp ファイルを生成します。オペレーショナル参照ストアの .dmp ファイルを再利用してオペレーショナル参照ストアをもう一つ作成できます。オペレーショナル参照ストアの C_REPOS_DB_VERSION テーブルの開始日および終了日の列には、正確でない開始日と終了日が記録されています。このエラーは機能には一切影響を与えないため、無視できます。

Hub サーバーが cmx_system スキーマに接続できない

Hub サーバーが cmx_system スキーマに接続できないことを確認するには、アプリケーションサーバーログを確認します。

例えば、データベースへアクセスできないことを示す次のエラーが見つかる場合があります。

Caused by: java.sql.SQLException: ORA-28000: the account is locked
DSRA0010E: SQL State = 99999, Error Code = 28,000

問題を解決するには、データベース接続の問題を解決します。アプリケーションサーバーコンソールを使用してデータベースへの接続をテストします。cmx_system スキーマへの接続を解決できない場合は、cmx_system スキーマを再作成します。

レコードのトークン化の必要性の検証に失敗する

マッチプロセスの実行時、以下のようなエラーメッセージが表示されることがあります。

SIP-16062: Failed to verify the need to tokenize records.

以下の環境変数の設定を確認します。

- ライブラリパス環境変数には以下のパスが含まれる必要があります。

UNIX の場合: <infamdm_install_directory>/hub/ cleanse/lib

Windows の場合: <infamdm_install_directory>\hub\ cleanse\lib

ライブラリパス環境変数はオペレーティングシステムに依存します。

- AIX。LIBPATH

- Suse または RedHat Linux。LD_LIBRARY_PATH

- Windows。PATH

- SSAPR 環境変数にはすべてのユーザーに対して次のパスが含まれる必要があります。

UNIX の場合: <infamdm_install_directory>/server_install_dir/ cleanse/resources

Windows の場合: <infamdm_install_directory>\server_install_dir\ cleanse\resources

プロセスサーバーのロード時の major.minor バージョンエラー

Process サーバーをロードしようとして、コンソールに何度も major.minor のエラーが起きる場合は、システムに正しいバージョンの Java がインストールされていることを確認します。

CORBA TRANSACTION_ROLLEDBACK の例外

データマネージャまたはマージマネージャを使用しているとき、MDM Hub によって CORBA TRANSACTION_ROLLEDBACK の例外が生成される場合があります。この例外が生成された場合は、WebSphere 管理コンソールを開いて、WebSphere プロセス定義に -Djava.vendor=IBM Java オプションを手動で設定してください。

認証中の Informatica Address Verification の例外

認証中に Informatica Address Verification が例外を生成します。JVM のスタックサイズが充分であることを確認します。

1. WebSphere コンソールを開きます。
2. [サーバー] > [アプリケーションサーバー] > <Your Server> > [プロセス定義] > [Java 仮想マシン] に移動します。
3. [JVM 汎用引数] に以下の引数を追加します。
 - Xss2000k - Initializes the stack size to 2000k
 - Xms128m - Initializes the heap with at least 128 MB
 - Xmx1024m - Initializes the heap with a maximum of 1024 MB
4. 設定を保存します。
5. WebSphere サーバーを再起動します。

オペレーショナル参照ストアにワークフローエンジンが設定されていません

MDM Hub をインストールし、前のバージョンからオペレーショナル参照ストア（ORS）をインポートすると、ORS にワークフローエンジンが設定されていないことを示す致命的なエラーが表示されます。このエラーの原因は、Siperian BPM ワークフローエンジンがデフォルトで登録されていないためです。ORS に設定する必要がある Siperian BPM ワークフローエンジンを登録するには、Workflow Manager を使用します。

プロセスサーバーの.ear ファイルをデプロイした後、エラーが発生する

WebSphere 8.5.5.9 を使用する Linux 環境では、Process サーバーをデプロイした後、次のエラーが発生します。

Too many open files. Unable to start cleanse ear.

Linux で ulimit パラメータの値を増やし、Process サーバーをデプロイします。

第 14 章

アンインストール

この章では、以下の項目について説明します。

- [アンインストールの概要, 133 ページ](#)
- [Hub Store のアンインストール, 133 ページ](#)
- [グラフィカルモードでの Process サーバーのアンインストール, 134 ページ](#)
- [グラフィカルモードでの Hub サーバーのアンインストール, 135 ページ](#)
- [グラフィカルモードでのリソースキットのアンインストール, 136 ページ](#)
- [コンソールモードでのプロセスサーバーのアンインストール, 137 ページ](#)
- [コンソールモードでの Hub Server のアンインストール, 137 ページ](#)
- [コンソールモードでのリソースキットのアンインストール, 137 ページ](#)
- [手動によるプロセスサーバーのデプロイ解除, 138 ページ](#)
- [手動による Hub Server のデプロイ解除, 138 ページ](#)

アンインストールの概要

MDM Hub をアンインストールするには、Process サーバー、Hub サーバー、Hub Store を MDM Hub の実装から削除する必要があります。

MDM Hub をアンインストールするには、以下の手順を実行します。

1. Hub Store をアンインストールします。
2. Process サーバーをアンインストールします。
3. Hub サーバーをアンインストールします。

Hub Store のアンインストール

Hub Store スキーマを削除し、Hub Store スキーマへのユーザーログインを削除して、Hub Store をアンインストールできます。Hub Store スキーマを削除する前に、Hub コンソールを使用して Hub Store スキーマの登録を解除します。

Hub Store スキーマを削除するには、管理者特権を持っている必要があります。

1. Hub コンソールを起動します。

2. 設定ワークベンチにある**データベースツール**をクリックします。
[データベース情報] ページが表示されます。
3. [書き込みロック] > [ロックの取得] の順にクリックします。
4. データベースのリストから、登録を解除するオペレーショナルリファレンスストアを選択します。
5. [データベースの登録解除] ボタンをクリックします。
オペレーショナルリファレンスストアの登録解除の確認を求めるメッセージが表示されます。
6. [はい] をクリックします。
7. SQL*Plus を使用して、Oracle インスタンスに接続します。
8. 各 Hub Store スキーマに対して次のコマンドを使用して、スキーマを削除します。
`drop user <user name> cascade;`
[カスケード] オプションを使用してスキーマを削除する場合、関連付けられたスキーマが削除されます。

グラフィカルモードでの Process サーバーのアンインストール

グラフィカルモードで Process サーバーをアンインストールできます。

UNIX におけるグラフィカルモードでのプロセスサーバーのアンインストール

MDM Hub をアンインストールするには、Process サーバーを削除する必要があります。Process サーバーをアンインストールする手順は、MDM Hub の実装の Process サーバーごとに実行する必要があります。

1. アプリケーションサーバーを停止します。
2. 次のディレクトリに移動します。
`<infamdm_install_directory>/hub/cleanse/UninstallerData`
3. アンインストーラを実行します。
`./"Uninstall Informatica MDM Hub Cleanse Match Server"`
4. [アンインストール] をクリックします。
アンインストールのプロセスが完了すると、[アンインストールの完了] ウィンドウが表示されます。
5. [完了] をクリックします。

Windows におけるグラフィカルモードでのプロセスサーバーのアンインストール

MDM Hub をアンインストールするには、Process サーバーを削除する必要があります。Process サーバーをアンインストールする手順は、MDM Hub の実装の Process サーバーごとに実行する必要があります。

1. アプリケーションサーバーを停止します。
2. [スタート] をクリックし、[プログラム] > [INFAMDM] > [Hub] > [クレンジング] > [Uninstaller Data] > [Informatica MDM Hub クレンジング一致サーバーのアンインストール] の順にクリックします。
[アンインストールの概要] ウィンドウが表示されます。

3. **【アンインストール】** をクリックします。
アンインストールのプロセスが完了すると、**【アンインストールの完了】** ウィンドウが表示されます。
4. **【完了】** をクリックします。

グラフィカルモードでの Hub サーバーのアンインストール

グラフィカルモードで Hub サーバーをアンインストールできます。

UNIX におけるグラフィカルモードでの Hub サーバーのアンインストール

MDM Hub をアンインストールするには、MDM Hub の実装から Hub サーバーを削除する必要があります。

1. アプリケーションサーバーが停止していることを確認します。
2. 次のディレクトリに移動します。
`<infamdm_install_directory>/hub/server/UninstallerData`
3. アンインストーラを実行します。
`./"Uninstall Informatica MDM Hub Server"`
【アンインストールの概要】 ウィンドウが表示されます。
4. **【アンインストール】** をクリックします。
アンインストールのプロセスが完了すると、**【アンインストールの完了】** ウィンドウが表示されます。
5. **【完了】** をクリックします。

Windows におけるグラフィカルモードでの Hub サーバーのアンインストール

MDM Hub をアンインストールするには、MDM Hub の実装から Hub サーバーを削除する必要があります。

1. アプリケーションサーバーが停止していることを確認します。
2. **【スタート】** ボタンをクリックし、**【プログラム】** > **【Infamdm】** > **【Hub】** > **【サーバー】** > **【UninstallerData】** > **【Informatica MDM Hub サーバーのアンインストール】** の順にクリックします。
【アンインストールの概要】 ウィンドウが表示されます。
3. **【アンインストール】** をクリックします。
アンインストールのプロセスが完了すると、**【アンインストールの完了】** ウィンドウが表示されます。
4. **【完了】** をクリックします。

グラフィカルモードでのリソースキットのアンインストール

グラフィカルモードでリソースキットをアンインストールできます。

UNIX でのグラフィカルモードによるリソースキットのアンインストール

リソースキットをアンインストールするには、MDM Hub の実装からリソースキットを削除する必要があります。

1. アプリケーションサーバーを停止します。
2. 次のディレクトリに移動します。
`<infamdm_install_directory>/hub/resourcekit/UninstallerData`
3. 次のコマンドを実行します。
`./"Uninstall Informatica MDM Hub Resource Kit"`
[Informatica MDM Hub リソースキットのアンインストール] ウィンドウが表示されます。
4. **[アンインストール]** をクリックします。
[アンインストールの完了] ウィンドウが開き、削除できなかった項目のリストが表示されます。
5. **[完了]** をクリックします。
6. 以下のディレクトリを手動で削除します。
`<infamdm_install_dir>/hub/resourcekit`

Windows でのグラフィカルモードによるリソースキットのアンインストール

リソースキットをアンインストールするには、MDM Hub の実装からリソースキットを削除する必要があります。

1. アプリケーションサーバーを停止します。
2. 次のディレクトリに移動します。
`<ResourceKit_install_dir>\deploy\UninstallerData`
3. Uninstall Informatica MDM Hub Resource Kit.exe をダブルクリックします。
[Informatica MDM Hub リソースキットのアンインストール] ウィンドウが表示されます。
4. **[アンインストール]** をクリックします。
[アンインストールの完了] ウィンドウが開き、削除できなかった項目のリストが表示されます。
5. **[完了]** をクリックします。
6. 以下のディレクトリを手動で削除します。
`<infamdm_install_dir>\hub\resourcekit`

コンソールモードでのプロセスサーバーのアンインストール

UNIX では、Process サーバーをコンソールモードでアンインストールできます。コンソールモードで Process サーバーをインストールした場合、コンソールモードで Process サーバーをアンインストールします。

1. 次のディレクトリに移動します。

```
<infamdm_install_dir>/hub/cleanse/UninstallerData
```

2. 以下のコマンドを入力してアンインストーラを実行します。

```
./"Uninstall Informatica MDM Hub Cleanse Match Server"
```

コンソールモードでの Hub Server のアンインストール

UNIX では、Hub サーバーをコンソールモードでアンインストールできます。コンソールモードで Hub サーバーをインストールした場合、コンソールモードで Hub サーバーをアンインストールします。

1. 次のディレクトリに移動します。

```
<infamdm_install_dir>/hub/server/UninstallerData
```

2. 以下のコマンドを入力してアンインストーラを実行します。

```
./"Uninstall Informatica MDM Hub Server"
```

コンソールモードでのリソースキットのアンインストール

コンソールモードでリソースキットをアンインストールできます。コンソールモードでリソースキットをインストールした場合、コンソールモードでリソースキットをアンインストールします。

1. 次のディレクトリに移動します。

UNIX の場合:<infamdm_install_dir>/hub/resourcekit/UninstallerData

Windows の場合:<infamdm_install_dir>\hub\resourcekit\UninstallerData

2. コマンドプロンプトで次のコマンドを実行します。

UNIX の場合:"Uninstall Informatica MDM Hub Resource Kit.bin" -i console

Windows の場合:"Uninstall Informatica MDM Hub Resource Kit.exe" -i console

手動によるプロセスサーバーのデプロイ解除

場合によっては、WebSphere 環境から Process サーバーのデプロイを手動で解除する必要があります。

- ▶ WebSphere Application Server 管理コンソールを使用して siperian-mrmcleanse.ear ファイルのデプロイを手動で解除します。

詳細については、WebSphere のマニュアルを参照してください。

手動による Hub Server のデプロイ解除

場合によっては、WebSphere 環境から Hub サーバーのデプロイを手動で解除する必要があります。

- ▶ WebSphere アプリケーションサーバーの管理コンソールを使用して、次のデプロイメントファイルのデプロイを解除します。

デプロイメントファイル名	説明
siperian-mrm.ear	必須。Hub サーバーアプリケーション。
entity360view-ear.ear	オプション。エンティティ 360 フレームワーク。
informatica-mdm-platform-ear.ear	オプション。Informatica プラットフォームアプリケーション。

詳細については、WebSphere のマニュアルを参照してください。

索引

A

ActiveVOS

- URN、設定 [109](#)
- インストール [103](#)
- インストールファイル [106](#)
- タイムマネージャの設定 [104](#)
- デプロイ [103](#), [106](#)
- 作業マネージャの設定 [104](#)

ActiveVOS Central

- インストール [105](#)

ActiveVOS コンソールの管理者ユーザー

- abAdmin ロール [28](#)
- 作成 [28](#)

ActiveVOS サーバー

- インストール [105](#)

ActiveVOS ファイル

- deployer.xml [106](#)
- install.properties [106](#)

H

HTTPS

- プロセスサーバー向け [99](#)

Hub コンソール

- 起動 [73](#), [77](#)

Hub コンソールクライアント

- build.properties
- 構成 [59](#)
- クラスタ環境 [59](#)
- マルチノード環境 [59](#)
- 構成 [59](#)

Hub サーバー

- EAR ファイルの再パッケージ化 [71](#)
- postInstallSetup スクリプト [63](#)
- WebSphere クラスタへのインストール [55](#)
- アンインストール [137](#)
- インストール [118](#)
- インストールログファイル [58](#)
- ウィザードを使用したインストール [53](#)
- カスタム JAR ファイルの再パッケージ化 [71](#)
- コマンドラインからのインストール [54](#)
- サイレントインストール [123](#)
- サイレントモードでのインストール [54](#)
- デプロイメント [62](#)
- デプロイメントスクリプト [62](#)
- バージョン情報 [58](#)
- ビルド番号 [58](#)
- 再デプロイ [60](#)
- 手動デプロイ [62](#), [65](#)

Hub サーバーのプロパティファイル

- 設定 [34](#)

Hub ストア

- アンインストール [133](#)
- テーブルスペース、作成 [17](#)

I

Identity Resolution

- デプロイ [106](#)

Infinispan

- 設定 [75](#), [76](#)

Informatica ActiveVOS

- スキーマの作成 [20](#)

Informatica Platform

- プロパティファイル [33](#)

J

JAAS アプリケーションログイン

- ActiveVOS の場合 [105](#)
- 構成 [105](#)

Java のオプション

- 設定 [22](#)

JMS メッセージキュー

- 設定 [68](#), [73](#)

JVM パラメータ

- 設定 [22](#)

M

MDM Hub

- Java Development Kit (JDK) の要件 [15](#)
 - インストールのタスク [12](#)
 - インストールのトポロジ [11](#)
 - インストールの設計 [11](#)
 - オペレーティングシステムのロケールの設定 [15](#)
 - コンポーネント [9](#)
 - システム要件 [15](#)
 - 概要 [9](#)
 - 環境変数の設定 [15](#)
- ### MDM Hub EAR ファイル
- 再パッケージ化 [71](#)

O

Oracle の設定

- init.ora パラメータの設定 [17](#)
- MDM Hub 用 [17](#)
- Oracle リサイクルビンの無効化 [17](#)

P

postInstallSetup スクリプト

- Hub サーバー用 [63](#)
- プロセスサーバー用 [97](#)
- 実行 [63](#), [97](#)

T

TLS
設定 [25](#)
TNS 名
追加 [17](#)

U

URN
ActiveVOS の設定 [109](#)

W

WebSphere
設定 [22](#), [61](#)
WebSphere クラスタ
Hub サーバーのインストール [55](#)
プロセスサーバーのデプロイ [89](#)
WebSphere の設定
Informatica Data Director 用の [33](#)

あ

アンインストール
Hub サーバー [137](#)
Hub ストア [133](#)
プロセスサーバー [137](#)

い

一致ポピュレーション
有効化 [100](#), [101](#)
インストーラ
ワークフロー [46](#), [82](#)
インストール
Hub サーバー
ウィザード [53](#)
コマンドライン [54](#)
サイレント [54](#)
プロセスサーバー
ウィザード [86](#)
コマンドライン [87](#)
サイレント [88](#)
リソースキット [118](#), [121](#)

お

オペレーショナル参照ストア
メタデータのインポート [40](#)
作成 [37](#)
登録 [78](#)

さ

サンプルスキーマ
インストール [113](#)
登録 [116](#)

た

ターゲットデータベース
選択 [77](#)

て

データベース
ターゲットデータベース [77](#)
接続テスト [17](#)
テーブルスペース
作成 [17](#)

と

トラブルシューティング
インストール後のプロセス [129](#)

ふ

プロセスサーバー
HTTPS、有効化 [99](#)
postInstallSetup スクリプト [97](#)
WebSphere クラスタへのデプロイ [89](#)
アンインストール [137](#)
インストールログファイル [90](#)
ウィザードを使用したインストール [86](#)
コマンドラインからのインストール [87](#)
サイレントモードでのインストール [88](#)
手動デプロイ [93](#), [97](#)
データソースの作成 [93](#)
デプロイ [97](#)
デプロイメント [93](#), [97](#)
デプロイメントスクリプト [93](#), [97](#)
バージョン情報 [91](#)
ビルド番号 [91](#)
再デプロイ [92](#), [98](#)
プロセスサーバーのプロパティファイル
設定 [34](#)

ま

マスターデータベース
メタデータのインポート [39](#)
作成 [35](#)

り

リソースキット
アンインストール [136](#)
インストール [118](#), [121](#)
サイレントプロパティファイル [123](#)

ろ

ログファイル
Hub サーバーのログファイル [58](#)
JBoss のログファイル [58](#), [90](#)
インストール後の設定ログファイル [58](#), [90](#)
インストール前提条件ログファイル [58](#), [90](#)
インストールログファイル [58](#), [90](#)
デバッグログファイル [58](#), [90](#)

ログファイル (続く)
プロセスサーバーのログファイル [90](#)

わ
ワークフローエンジン
追加 [111](#)