



# Red Hat JBoss Data Virtualization 6.3 インストールガイド

---

インストール担当チーム向けのガイド

Red Hat Customer Content  
Services



## インストール担当チーム向けのガイド

Red Hat Customer Content Services

## 法律上の通知

Copyright © 2016 Red Hat, Inc.

This document is licensed by Red Hat under the [Creative Commons Attribution-ShareAlike 3.0 Unported License](https://creativecommons.org/licenses/by-sa/3.0/). If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, OpenShift, Fedora, the Infinity logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

## 概要

本ガイドでは Red Hat JBoss Data Virtualization のインストールオプションについて説明します。

## 目次

<b>第1章 作業を開始する前に</b>	<b>3</b>
1.1. データのバックアップ	3
1.2. 変数名: EAP_HOME	3
<b>第2章 プラットフォームの要件</b>	<b>4</b>
2.1. アーキテクチャーと必要性の評価	4
<b>第3章 製品のダウンロード</b>	<b>7</b>
3.1. Red Hat カスタマーポータル	7
3.2. JBoss Data Virtualization インストーラーのダウンロード	7
3.3. ダウンロードされたファイルの検証	7
<b>第4章 Maven リポジトリ</b>	<b>9</b>
4.1. 提供される Maven リポジトリ	9
4.2. ファイルシステムリポジトリを使用するよう Maven を設定	9
4.3. オンラインリポジトリを使用するよう Maven を設定	12
4.4. 依存関係管理	15
4.5. Red Hat JBoss Data Virtualization に Maven 依存関係を使用	15
4.6. オフラインモード	16
<b>第5章 事前に必要なコンポーネントのインストール</b>	<b>18</b>
5.1. Red Hat Enterprise Linux に OpenJDK をインストール	18
5.2. Maven のインストール	18
<b>第6章 製品のインストール</b>	<b>21</b>
6.1. Red Hat JBoss Data Virtualization のインストール方法	21
6.2. JBoss Data Virtualization のインストール: グラフィカルインストール	21
6.3. テキストベースのインストーラーを使用した JBoss Data Virtualization のインストール	23
6.4. JBoss Data Virtualization 開発ツールのインストール	23
<b>第7章 製品の開始および停止</b>	<b>25</b>
7.1. JBoss Data Virtualization の起動	25
7.2. JBoss EAP 6 を管理対象ドメインとして起動	25
7.3. JBoss Data Virtualization の停止	26
<b>第8章 ドメインモード</b>	<b>27</b>
8.1. ドメインモードでのインストール	27
<b>第9章 JBoss Data Grid キャッシュのインストール</b>	<b>29</b>
9.1. JBoss Data Grid アダプターの設定	29
<b>第10章 ODBC サポート</b>	<b>30</b>
10.1. はじめに	30
10.2. Red Hat Enterprise Linux に ODBC ドライバーをインストール	30
10.3. Microsoft Windows に ODBC ドライバーをインストール	31
10.4. Solaris に ODBC ドライバーをインストール	32
10.5. ODBC の設定	33
10.6. DSN を使用しない接続	36
<b>第11章 自動インストール</b>	<b>37</b>
11.1. 自動インストール	37
<b>第12章 クラウド環境での実行</b>	<b>38</b>
12.1. Amazon AWS Cloud インスタンスでの Red Hat JBoss Data Virtualization の実行	38
12.2. Google Compute インスタンスでの Red Hat JBoss Data Virtualization の実行	38

12.3. Red Hat JBoss Data Virtualization の Azure インスタンスへの接続	39
<b>第13章 OpenShift で Red Hat JBoss Data Virtualization Cartridge を実行</b> .....	<b>41</b>
13.1. カートリッジのインストール	41
13.2. OpenShift でデータ仮想化 Web インターフェースを使用	42
<b>付録A 事前に必要なコンポーネントのインストール</b> .....	<b>46</b>
A.1. Red Hat Enterprise Linux に OpenJDK をインストール	46
<b>付録B 設定情報</b> .....	<b>47</b>
B.1. サポートされるデータソースおよびトランスレーター	47
<b>付録C 改訂履歴</b> .....	<b>49</b>

## 第1章 作業を開始する前に

### 1.1. データのバックアップ



#### 警告

Red Hat は、システム設定をバックアップしてから本書に掲載されている設定タスクを実行することを推奨します。

### 1.2. 変数名: EAP\_HOME

**EAP\_HOME** は、JBoss Data Virtualization がデプロイされた Red Hat JBoss Enterprise Application Platform インストールのルートディレクトリーを示しています。

## 第2章 プラットフォームの要件

### 2.1. アーキテクチャーと必要性の評価

#### 最低推奨サイズ

以下の最低条件を開始点としてください。これらの条件は期待される使用量に応じて調整する必要があります。

JBDS (Teiid Designer) – アプリケーションサーバーなし

- 2 GB の RAM が最低でも必要ですが、大型のモデルにはより多くの RAM が必要です。
- 最新のプロセッサ
- インストールされた製品ファイルには 500 MB のディスク領域が必要です。
- モデルプロジェクトや関連するアーティファクトには 2 GB 以上の領域が必要です。

以下の推奨サイズの目的は、サーバーの開始点 (最小サイズ) を提供することです。

以下は開始点で、推奨される最低条件になります。また、クライアントの情報が取得できないときも推奨サイズとして使用してください。

DV サーバーの最低条件は次のとおりです。

- 16 GB の JVM メモリーサイズ。
- 最新のマルチコア (デュアル以上) プロセッサまたは最新のマルチコアプロセッサを持つマルチソケットシステム
- JBoss のサーバー製品と DV コンポーネントには 20 GB 以上のディスク領域が必要です。
- インストールされた製品のファイルには 1 GB のディスクが必要です。
- ログファイルとデプロイされたアーティファクトには 5 GB 以上が必要です。
- BufferManager の maxBufferSpace には 50 GB (デフォルト) が必要です。
- Modeshape (リポジトリ) が使用される場合、ファイル領域を最低でも 5 GB 増やす必要があります。
- 

JVM フットプリントの最低条件を判断するには、平行性、データボリューム、および計画処理の 3 点を考慮します。

- 同時性: 最大セッション、トランスポートスレッドプール、エンジンスレッドプール/エンジン (特に最大アクティブ値) の設定、および接続プールサイズを考慮します。
- データボリューム: バッチサイズを基にデータソースから読み取られるデータの量を考慮します。デフォルトのプロセッサバッチサイズは 256 で、行ごとに約 2k バイトをターゲットとし、約 512kb のサイズでシステムを通過します。しかし、メモリー容量の多いマシンでは、バッチサイズを 512 に増加し、バッチごとにサイズを約 1mb にすることが推奨されます。
- 計画処理: クエリー計画を基に行われるデータの追加処理を考慮します。通常、これには追加のメモリーが必要です (ソートなど)。

サイズの決定に使用される仮定は次のとおりです。

- サーバーは調整されているため (スレッドプール、接続プールなど)、待機または最大のスループットな

して各クエリーは実行されます。

- ※ 計画ではデータソースごとに 1 つのソースクエリーがあります (複雑なクエリーが増えると、さらに多くのメモリーが必要になります)。
- ※ Teiid と同じ JVM で実行されている他のアプリケーションはありません (同じ JVM で他のアプリケーションが実行される場合は、追加のメモリー要件を考慮する必要があります)。
- ※ 非トランザクションの直線的な読み取りを実行します (Teiid は、事前にバッチフェッチを行います、これはメモリーの要件が増えるため、バッチタイプが 2 倍になります)。
- ※ デフォルトのプロセッサバッチサイズは 512 に設定されています (256 のデフォルトから変更)。これは、バッチのオーバーヘッドを低減するため、メモリー容量の多いマシンで推奨されます。

最小 JVM サイズを推定する式は **(同時クエリー) \* (4 \* バッチバイト) + (2 \* (計画ごとのソースクエリーの数 \* ソースバイトの概算)) + オーバーヘッド** になります。説明は次のとおりです。

- ※ 同時クエリー
- ※ バッチバイト: システムを通過するバッチを表します。256 バッチサイズをデフォルトで使用するとそのバイトサイズは約 512 kb になります。しかし、推奨される 512 バッチサイズを使用すると、各バッチは約 1mb になります。バッチバイトを 2 倍にするのは、別のバッチのプロセス中に部分的なバッチが読み出される場合にワークアイテムでバッチを格納するためです。
- ※ 2 \* バッチバイト
- ※ 計画ごとのソースクエリー: クエリーのデータソースの数ですが、前提を基に制限されます。
- ※ 4 \* バッチバイトのオンヒープのサイズ
- ※ オーバーヘッド: AS (約 300mb) の調整、追加の Teiid オーバーヘッド (キャッシュ、計画など)、および接続プールのオーバーヘッドが含まれます。他の項目を算出するのは難しいため約 300mb のみが使用されますが、パフォーマンスを向上させるにはこれを検討する必要があります。

使用される正確な式は次のとおりです。

- ※  $(\text{同時クエリー}) * (4 * \text{バッチバイト}) + (2 * \text{ソースバイト}) * \text{ソースクエリーの数} + 300\text{mb}$
- ※  $(\text{同時クエリー}) * (4 * 1\text{mb}) + (2 * 512\text{kb}) * \text{ソースクエリーの数} + 300\text{mb}$
- ※  $(\text{同時クエリー}) * (4\text{mb}) + (1\text{mb}) * \text{ソースクエリーの数} + 300\text{mb}$
- ※  $\text{同時実行の数} * (5\text{mb}) * \# \text{ソースクエリー} + 300\text{mb}$

表2.1 設定

同時実行 ソースクエリーの数	100	200
2	1.3 gb	2.3 gb
5	2.8 gb	5.3 gb
10	5.3 gb	10.3 gb

最大同時クエリーを基に、以下を実行してシステムのエンジンを調整します。

- ※ maxActivePlans を最大同時クエリーに設定します。
- ※ maxThreads が maxActivePlans の 2 倍になるように設定します (トランザクションが使用される場合は 3 倍)。
- ※ 各データソースの最大プールサイズが最大同時ソースクエリーと同じになるように設定します (最大同時 maxThreads が maxActivePlans の 5 倍クエリーが最小限になりますが、複数のソースクエリーが作成さ

れる原因となるサブクエリーがある場合、状況に応じて最大プールサイズを増加する必要があります)。

- ※ すべての調整を行った後、サーバーのメモリー容量に余裕がある場合、`processBatchSize` および `connectorBatchSize` を増やし (例: それぞれ 512 および 1024)、データソースおよびエンジンからのスループットを増加することを検討してください。メモリー不足の場合は、JVM サイズを増やしてください。6GB 以下のメモリーを持つマシンは 512 のままにし、これよりも大きなメモリーを持つマシンにはより大きなサイズを使用してください。

## 第3章 製品のダウンロード

### 3.1. Red Hat カスタマーポータル

The Red Hat カスタマーポータルは <https://access.redhat.com/home> にある Web サイトです。サブスクリプションの管理や維持、Red Hat ナレッジベースへのアクセス、および Red Hat やパートナーへの連絡を 1 か所で行うことができます。

### 3.2. JBoss Data Virtualization インストーラーのダウンロード

JBoss Data Virtualization のインストーラーアーカイブは、<https://access.redhat.com/> の Red Hat カスタマーポータルから入手できます。**.jar** アーカイブを使用してグラフィカルインストーラーを実行できます。ここでは、アーカイブをダウンロードする手順を示します。

#### 前提条件

- Red Hat カスタマーポータル <https://access.redhat.com/> でアカウントを設定する必要があります。
- Red Hat のサブスクリプションが最新の状態であることを確認してください。
- サポートされる設定を確認し、ご使用のシステムがサポートされるようにしてください。
- インストールディレクトリーの管理権限があることを確認してください。
- Microsoft Windows サーバーでショートカットを使用する場合、JAVA\_HOME と PATH が環境プロパティに設定されている必要があります。
- Java 6、7 または 8 の JDK が必要です。(Red Hat SSO をお使いになる場合には Java 7 以上が必要になることに注意してください。Java 7 は Impala、Hive および HBase データソースの最小要件でもあります。)
- オプション: JBoss Data Virtualization インストーラーに同梱されている Red Hat JBoss EAP のバージョンを使用しない場合には Red Hat JBoss Enterprise Application Platform 6.4.x が必要です。

#### 手順3.1 JBoss Data Virtualization インストーラーのダウンロード

1. Red Hat カスタマーポータルにログインします。
  - a. ブラウザーのアドレスバーに <https://access.redhat.com/> と入力します。
  - b. ログインをクリックし、**Red Hat Login** と **Password** を入力してカスタマーポータルにアクセスします。アカウントをお持ちでない場合はアカウントを登録する必要があります。
2. JBoss Data Virtualization インストーラーをダウンロードします。
  - a. ダウンロードをクリックし、**Red Hat JBoss Data Virtualization** をクリックします。
  - b. **Red Hat JBoss Data Virtualization [バージョン] Installer** オプションの横にある **Download** をクリックします。
  - c. ファイルを保存します。

### 3.3. ダウンロードされたファイルの検証

## 手順3.2 Red Hat Enterprise Linux でのファイルチェックサムの検証

### 1. ダウンロードされたファイルのチェックサム値を取得

- a. <https://access.redhat.com/jbossnetwork/> へ移動し、必要な場合はログインします。
- b. **Product** で製品を選択し、**Version** でバージョンを指定します。
- c. 検証するパッケージを選択すると、**Software Details** ページに移動します。
- d. **MD5** と **SHA-256** のチェックサム値を書き留めておきます。

### 2. ファイルでチェックサムツールを実行

- a. ターミナルウィンドウで、ダウンロードされたファイルが含まれるディレクトリーへ移動します。
- b. `md5 downloaded_file` を実行します。
- c. `shasum downloaded_file` を実行します。

以下は出力例になります。

```
[localhost]$ md5 jboss-dv-installer-[VERSION]-redhat-[VERSION].jar
MD5 (jboss-dv-installer-[VERSION]-redhat-[VERSION].jar) =
0d1e72a6b038d8bd27ed22b196e5887f
[localhost]$ shasum jboss-dv-installer-[VERSION]-redhat-[VERSION].jar
a74841391bd243d2ca29f31cd9f190f3f1bdc02d  jboss-dv-installer-
[VERSION]-redhat-[VERSION].jar
```

### 3. チェックサム値を比較

- a. **md5** および **shasum** コマンドによって返されたチェックサム値と、**Software Details** ページに表示された対応する値を比較します。
- b. 2つのチェックサム値が同じでない場合は、再度ファイルをダウンロードします。チェックサム値が異なる場合、ダウンロード中にファイルが破損したか、サーバーへアップロードされた後に変更が加えられたことを意味します。数回ダウンロードしてもチェックサムの検証に失敗する場合は、Red Hat サポートまでご連絡ください。
- c. 2つのチェックサム値が同じであれば、ダウンロードされたファイルを安全に使用できます。

#### 結果:

ダウンロードされたファイルの整合性が検証されます。



#### 注記

Microsoft Windows にはチェックサムツールが含まれていません。MD5summerfrom (<http://www.md5summer.org/>) などのサードパーティー製の MD5 アプリケーションをダウンロードしてください。

## 第4章 Maven リポジトリ

### 4.1. 提供される Maven リポジトリ

本リリースでは、アプリケーションを構築するために必要なアーティファクトが含まれるリポジトリのセットが提供されます。適切な構築機能を提供するには、これらのリポジトリと Maven セントラルリポジトリを使用するよう Maven を設定する必要があります。

同じ機能を提供する、交換可能な2つのリポジトリのセットが提供されます。1つ目のセットはローカルファイルシステムでダウンロードおよび保存できます。2つ目のリポジトリはオンラインでホストされ、リモートリポジトリとして使用されます。インストール時に Maven の **settings.xml** ファイルの場所を指定した場合、Maven はオンラインリポジトリを使用するよう設定されています。



#### 重要

オンラインリモートリポジトリのセットは技術プレビューのコンポーネントのソースです。そのため、パッチの対象ではなく、開発環境の使用のみがサポートされます。本番環境でオンラインリポジトリのセットを使用すると、セキュリティの脆弱性の原因となる可能性があるためユースケースではサポートされません。詳細は <https://access.redhat.com/site/maven-repository> を参照してください。

### 4.2. ファイルシステムリポジトリを使用するよう Maven を設定

#### 概要

Red Hat JBoss EAP DV の Maven リポジトリはオンライン上にあるため、ローカルにダウンロードし、インストールする必要はありません。しかし、JBoss EAP の Maven リポジトリをローカルでインストールする場合は、ローカルファイルシステム上のインストール、Apache Web Server 上のインストール、および Maven リポジトリマネージャーを用いたインストールの3つの方法を使用できます。この例では、ローカルのファイルシステムへ JBoss EAP 6 の Maven リポジトリをダウンロードする手順を取り上げます。この方法は、開発環境で Maven の知識を深めることができますが、チームによる本番環境での使用は推奨されません。

#### 手順4.1

1. <https://access.redhat.com/jbossnetwork/restricted/listSoftware.html?product=data.services.platform> からリポジトリをダウンロードします。
2. **ダウンロード** ボタンをクリックし、リポジトリが含まれる **.zip** ファイルをダウンロードします。
3. ローカルファイルシステム上の Zip 形式のファイルを希望のディレクトリで展開します。
4. 展開されていないリポジトリのエントリを Maven の **settings.xml** ファイルに追加します。以下のコード例には、リポジトリがあるプロファイルと、プロファイルのアクティベートエントリが含まれます。

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/xsd/settings-1.0.0.xsd">
  <localRepository/>
```

```

<profiles>
  <!-- Profile with local repositories required by Data
Virtualization -->
  <profile>
    <id>dv-local-repos</id>
    <repositories>
      <repository>
        <id>dv-[VERSION]-repository</id>
        <name>DV [VERSION] GA Repository</name>
        <url>file://<!-- path to the repository -->/jboss-dv-
[VERSION].redhat-[VERSION]-maven-repository/maven-repository</url>
        <layout>default</layout>
        <releases>
          <enabled>true</enabled>
          <updatePolicy>never</updatePolicy>
        </releases>
        <snapshots>
          <enabled>false</enabled>
          <updatePolicy>never</updatePolicy>
        </snapshots>
      </repository>
    </repositories>
    <pluginRepositories>
      <pluginRepository>
        <id>dv-[VERSION].GA-redhat-[VERSION]-repository</id>
        <name>DV [VERSION] GA Repository</name>
        <url>file://<!-- path to the repository -->/jboss-dv-
[VERSION].redhat-[VERSION]-maven-repository/maven-repository</url>
        <layout>default</layout>
        <releases>
          <enabled>true</enabled>
          <updatePolicy>never</updatePolicy>
        </releases>
        <snapshots>
          <enabled>false</enabled>
          <updatePolicy>never</updatePolicy>
        </snapshots>
      </pluginRepository>
    </pluginRepositories>
  </profile>
</profiles>
<activeProfiles>
  <!-- Activation of the Data Virtualization profile -->
  <activeProfile>dv-local-repos</activeProfile>
</activeProfiles>
</settings>

```

**結果:**

Maven リポジトリがダウンロードされ、ローカルファイルシステムで展開された後、Maven の **settings.xml** ファイルに登録されます。Maven の構築を実行するときに使用できます。

**トラブルシューティング**

問： アプリケーションを構築またはデプロイするときにエラーが表示されます。

**答：問題**

プロジェクトをビルドまたはデプロイするときに、以下のエラーの1つまたは両方が発生し、失敗します。

✎ **[ERROR] Failed to execute goal on project *PROJECT\_NAME***

✎ **Could not find artifact *ARTIFACT\_NAME***

**原因**

キャッシュされたローカル Maven リポジトリに古いアーティファクトが含まれている可能性があります。

**解像度**

この問題を解決するには、キャッシュされたローカルディレクトリを削除します。このディレクトリは Linux の場合は `~/.m2/repository/` ディレクトリ、Windows の場合は `%SystemDrive%\Users\USERNAME\.m2\repository\` ディレクトリになります。**`mvn clean install -U`** を実行します。これにより、次回構築を行うと Maven が必要なアーティファクトの適切なバージョンをダウンロードするようになります。

問：JBoss Developer Studio が古い Maven 設定を使用します。

**答：問題**

Maven 設定を更新しても、この設定は JBoss Developer Studio には反映されません。

**原因**

Maven の **`settings.xml`** ファイルの編集時に JBoss Developer Studio が実行されていると、この設定は JBoss Developer Studio には反映されません。

**解像度**

IDE の Maven 設定を更新します。メニューから **Window** → **Preferences** を選択します。**Preferences** ウィンドウで **Maven** を展開し、**User Settings** を選択します。**Update Settings** ボタンをクリックし、JBoss Developer Studio の Maven ユーザー設定を更新します。

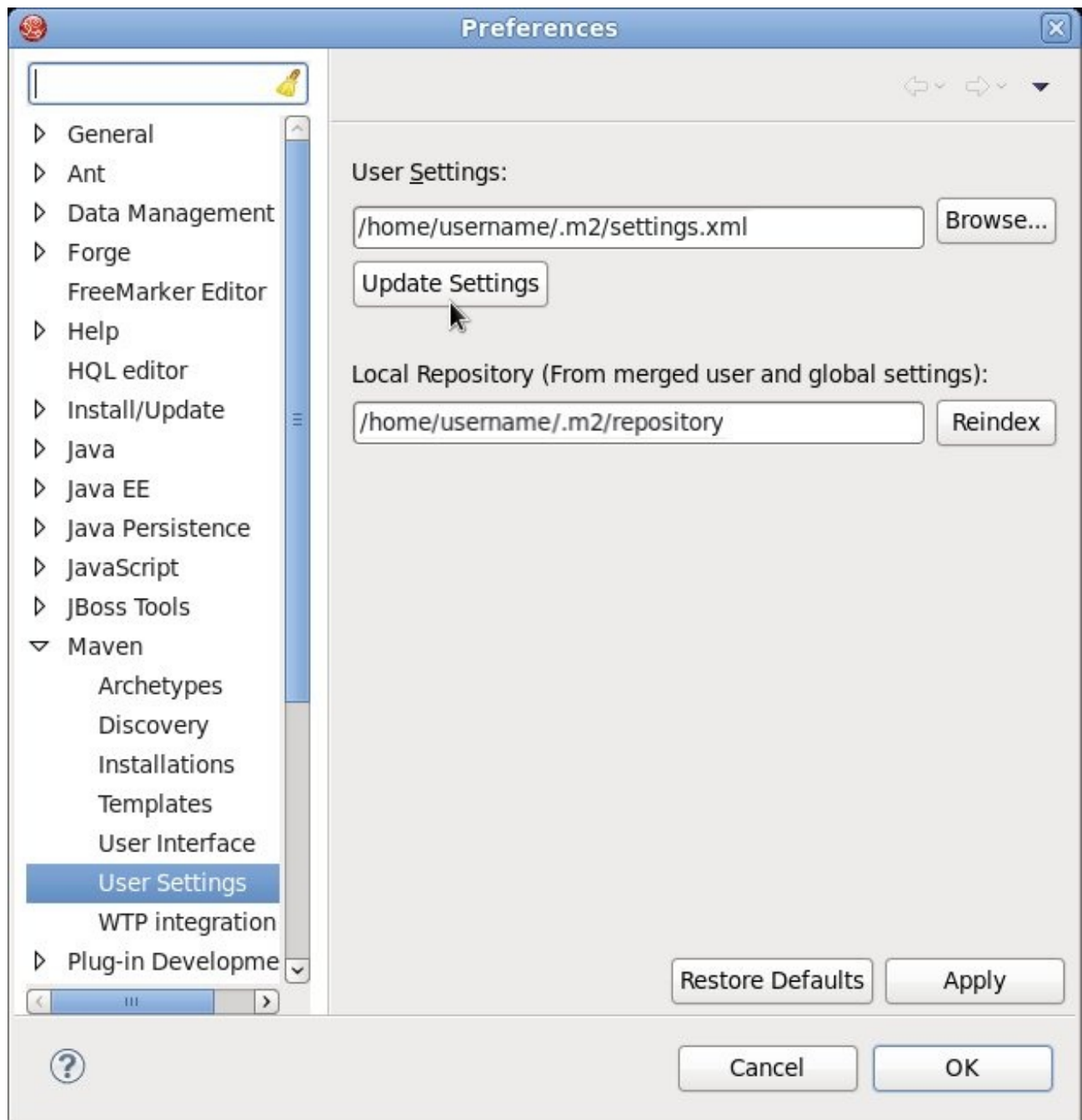


図4.1 Maven ユーザー設定の更新

### 4.3. オンラインリポジトリを使用するよう Maven を設定

Red Hat JBoss Data Virtualization に必要なオンラインリポジトリは  
<http://maven.repository.redhat.com/techpreview/all/> にあります。

インストール中に Maven の **settings.xml** ファイルの場所を指定した場合、Maven はすでにオンラインリポジトリを使用するよう設定されています。

インストール中に Maven リポジトリを設定しなかった場合、以下の手順に従ってリポジトリを設定することができます (プロジェクトの POM ファイルを使用して設定することもできますが、推奨されません)。

#### 手順4.2 オンラインリポジトリを使用するよう Maven を設定

1. 以下のコードサンプルのように、オンラインリポジトリのエントリーを Maven の **settings.xml** ファイルに追加します。

```
<settings xmlns="http://maven.apache.org/SETTINGS/1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/SETTINGS/1.0.0
http://maven.apache.org/xsd/settings-1.0.0.xsd">

  <profiles>
    <!-- Profile with online repositories required by Data
Virtualization -->
    <profile>
      <id>dv-online-profile</id>
      <repositories>
        <repository>
          <id>jboss-ga-repository</id>

<url>http://maven.repository.redhat.com/techpreview/all</url>
          <releases>
            <enabled>>true</enabled>
          </releases>
          <snapshots>
            <enabled>>false</enabled>
          </snapshots>
        </repository>
      </repositories>
      <pluginRepositories>
        <pluginRepository>
          <id>jboss-ga-plugin-repository</id>

<url>http://maven.repository.redhat.com/techpreview/all</url>
          <releases>
            <enabled>>true</enabled>
          </releases>
          <snapshots>
            <enabled>>false</enabled>
          </snapshots>
        </pluginRepository>

      </pluginRepositories>
    </profile>
  </profiles>

  <activeProfiles>
    <!-- Activation of the Data Virtualization profile -->
    <activeProfile>dv-online-profile</activeProfile>
  </activeProfiles>

</settings>
```

2. JBoss Developer Studio の稼働中に **settings.xml** ファイルを変更した場合は、IDE の Maven 設定を更新する必要があります。メニューで **Window** → **Preferences** と選択します。 **Preferences** ウィンドウで **Maven** を展開し、 **User Settings** を選択します。 **Update Settings** ボタンをクリックし、JBoss Developer Studio の Maven ユーザー設定を更新します。

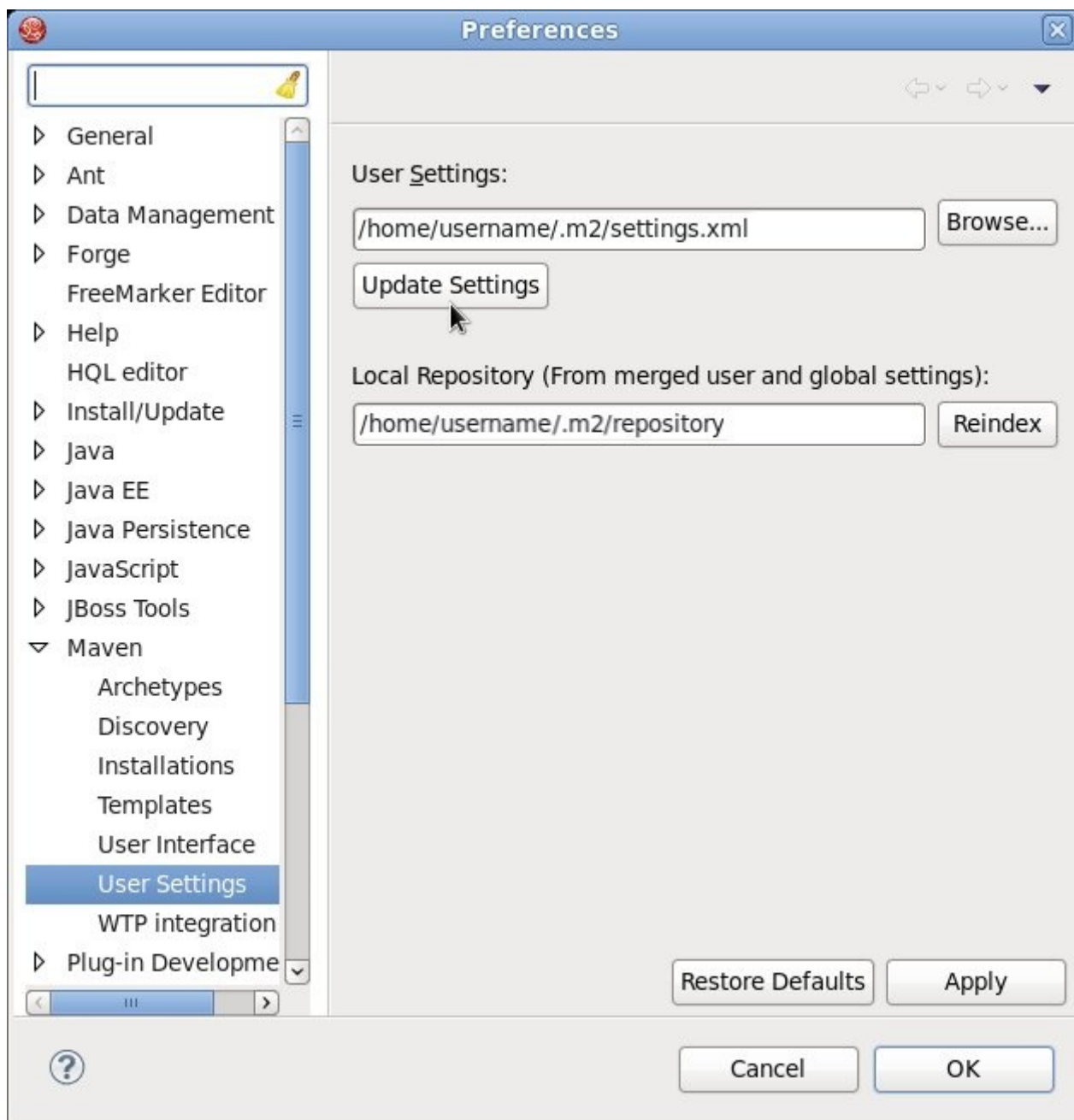


図4.2 Maven ユーザー設定の更新

**結果:**

Red Hat JBoss 製品向けに提供されるオンラインリポジトリを使用するよう Maven が設定されます。

**重要**

キャッシュされたローカル Maven リポジトリに古いアーティファクトが含まれる場合は、プロジェクトをビルドまたはデプロイしたときに以下のいずれかの Maven エラーメッセージが発生することがあります。

- ✧ Missing artifact *ARTIFACT\_NAME*
- ✧ [ERROR] Failed to execute goal on project *PROJECT\_NAME*; Could not resolve dependencies for *PROJECT\_NAME*

この問題を解決するには、キャッシュされたローカルディレクトリーを削除します。このディレクトリーは Linux の場合は `~/.m2/repository/` ディレクトリー、Windows の場合は `%SystemDrive%\Users\USERNAME\.m2\repository\` ディレクトリーになります。これにより、次の構築中に Maven が必要なアーティファクトの適切なバージョンをダウンロードするようになります。

## 4.4. 依存関係管理

Red Hat JBoss Data Virtualization プロジェクトで適切な Maven 依存関係を使用するには、関係する Bill Of Materials (BOM) および親 POM ファイルをプロジェクトの `pom.xml` ファイルに追加する必要があります。BOM および親 POM ファイルを追加して、適切なバージョンのプラグインと提供された Maven リポジトリからの推移的依存関係が確実にプロジェクトに含まれるようにします。

Maven リポジトリは Maven セントラルリポジトリとの組み合わせのみで使用することを目的としており、他のリポジトリは必要ありません。

使用する親 POM ファイルは `org.jboss.dv.component.management:dv-parent-[VERSION].pom` です。

使用する BOM ファイルは `org.jboss.dv.component.management:dv-dependency-management-all-[VERSION].pom` です。

## 4.5. Red Hat JBoss Data Virtualization に Maven 依存関係を使用

Red Hat JBoss Data Virtualization に Maven 依存関係を使用するには、関係する Bill Of Materials (BOM) および親 POM ファイルをプロジェクト POM ファイルに追加する必要があります。BOM および親 POM ファイルを追加して、適切なバージョンのプラグインと推移的依存関係が確実にプロジェクトに含まれるようにします。

例を以下に示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<project xmlns="http://maven.apache.org/POM/4.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
    http://maven.apache.org/maven-v4_0_0.xsd">

  <modelVersion>4.0.0</modelVersion>

  <!-- Example POM file using the DV 6.3.0 and EAP 6.4 component versions.
    - Parent is set to the DV 6.3.0 parent management POM, which will
    - bring in the correct toolchain (plugin) versions.
```

- *DependencyManagement dependencies include the DV 6.3.0 and EAP 6.4*
- *BOMs - which will bring in the correct compile-time (and other*
- *scoped) versions.*
- >

```
<name>Example POM for DV 6.3.0</name>
<groupId>org.jboss.dv</groupId>
<artifactId>dv-example</artifactId>
<version>0.0.1</version>
<packaging>pom</packaging>

<parent>
  <!-- DV version (parent) -->
  <groupId>org.jboss.dv.component.management</groupId>
  <artifactId>dv-parent</artifactId>
  <version>2.3.0.redhat-10</version>
</parent>

<dependencyManagement>
  <dependencies>
    <!-- DV BOM -->
    <dependency>
      <groupId>org.jboss.dv.component.management</groupId>
      <artifactId>dv-dependency-management-all</artifactId>
      <version>2.3.0.redhat-10</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>

</project>
```

## 4.6. オフラインモード

### 4.6.1. カスタムオフラインリポジトリのユースケース

プロジェクトの開発段階からデプロイメント段階に進む場合、通常はアーティファクトを必要時にインターネットからダウンロードせずに必要なアーティファクトをすべて事前インストールすると便利です。この場合、デプロイメントに必要なアーティファクトを格納するカスタムのオフラインリポジトリを作成するのが理想的です。しかし、アプリケーションのバンドルおよび機能に関連する**すべての**推移的依存関係を含める必要があるため、カスタムオフラインリポジトリを手動で作成するのは容易ではありません。

カスタムオフラインリポジトリの作成に最も適した方法は、Apache Karaf の **features-maven-plugin** プラグインを利用してカスタムオフラインリポジトリを生成することです。

### 4.6.2. Maven プロジェクトのリポジトリのダウンロード

Maven プロジェクトとそのランタイム依存関係をビルドするためにオフラインリポジトリを作成する必要がある場合は、[Maven 依存関係プラグイン](#)を使用できます。

たとえば、Maven プロジェクトのトップレベルディレクトリー (カレントディレクトリーに **pom.xml** ファイルがある) から以下の Maven コマンドを実行できるはずで

```
mvn org.apache.maven.plugins:maven-dependency-plugin:2.8:go-offline -  
Dmaven.repo.local=/tmp/cheese
```

このコマンドを実行すると、**/tmp/cheese** ディレクトリーにプロジェクトをビルドするために必要な Maven 依存関係およびプラグインがダウンロードされます。

### 4.6.3. オフラインリポジトリの生成

カスタムオフラインリポジトリを生成するには、新しいコマンドプロンプトを開き、***ProjectDir/custom-repo*** ディレクトリーへ移動してから、以下の Maven コマンドを入力します。

```
mvn generate-resources
```

Maven のビルドが正常に完了すれば、カスタムオフラインリポジトリが以下の場所に生成されるはずです。

```
ProjectDir/custom-repo/target/features-repo
```

## 第5章 事前に必要なコンポーネントのインストール

### 5.1. Red Hat Enterprise Linux に OpenJDK をインストール

#### 手順5.1 Red Hat Enterprise Linux に OpenJDK をインストール

##### 1. ベースチャンネルのサブスクリプション

RHN ベースチャンネルから OpenJDK を取得します (Red Hat Enterprise Linux のインストールはデフォルトでこのチャンネルにサブスクリプションされています)。

##### 2. パッケージのインストール

yum ユーティリティを使用して OpenJDK をインストールします: **yum install java-1.7.0-openjdk-devel**

##### 3. OpenJDK がシステムデフォルトであることを確認

以下の手順に従って、正しい JDK がシステムデフォルトとして設定されていることを確認します。

- a. root で **java** の alternatives コマンドを実行します。

```
/usr/sbin/alternatives --config java
```

- b. **/usr/lib/jvm/jre-1.7.0-openjdk/bin/java** を選択します。

- c. 同様に、**javac** の alternative コマンドを実行します。

```
/usr/sbin/alternatives --config javac
```

- d. **/usr/lib/jvm/java-1.7.0-openjdk/bin/javac** を選択します。

### 5.2. Maven のインストール

#### 前提条件

以下のソフトウェアがインストールされている必要があります。

- ※ 圧縮ファイルの内容を抽出するアーカイブツール。
- ※ Open JDK。

#### 手順5.2 Maven のインストール

##### 1. Maven のダウンロード

- a. ブラウザーのアドレスバーに <http://maven.apache.org/download.cgi> を入力します。
- b. **apache-maven-[latest-version]** ZIP ファイルをダウンロードし、ハードドライブに保存します。

##### 2. Maven のインストールおよび設定

## A. Red Hat Enterprise Linux の場合

- a. Maven をインストールするディレクトリーに ZIP ファイルを展開します。
- b. ターミナルで `.bash_profile` ファイルを開きます: **vi ~/.bash\_profile**
- c. `M2_HOME` 環境変数をファイルに追加します。

```
export M2_HOME=/path/to/your/maven
```

- d. さらに、`M2` 環境変数をファイルに追加します。

```
export M2=$M2_HOME/bin
```

- e. ファイルに `PATH` 環境変数を追加します。

```
export PATH=$M2:$PATH
```

- f. `JAVA_HOME` が `JDK` の場所に設定されるようにします。例は次のとおりです。

```
export JAVA_HOME=/usr/lib/jvm/java-1.7.0-openjdk.x86_64
```

- g. `$JAVA_HOME/bin` が `PATH` 環境変数に含まれるようにします。
- h. ファイルを保存して、テキストエディターを終了します。
- i. **source ~/.bash\_profile** コマンドを実行して変更が反映されるようにします。
- j. 以下のコマンドを実行して、ご使用のマシンに Maven が正常にインストールされたことを確認します。

```
mvn --version
```

## B. Microsoft Windows の場合

- a. Maven をインストールするディレクトリーに ZIP ファイルを展開します。アーカイブからサブディレクトリー **apache-maven-[latest-version]** が作成されます。
- b. **Start+Pause|Break** を押します。**System Properties** ダイアログボックスが表示されます。
- c. **Advanced** タブをクリックし、**Environment Variables** をクリックします。
- d. System Variables で **Path** を選択します。
- e. **Edit** をクリックし、区切り文字にセミコロンを使用して Maven のパスを 2 つ追加します。
  - Add the `M2_HOME` variable and set the path to **C:\path\to\your\Maven**.
  - `M2` 変数を追加し、値を `%M2_HOME%\bin` に設定します。
- f. Path 環境変数を更新または作成します。
  - `%M2%` 変数を追加し、コマンドラインから Maven を実行できるようにします。
  - 変数 `%JAVA_HOME%\bin` を追加し、適切な Java インストールへのパスを設定しま

す。

- g. **OK** をクリックして、**System Properties** ダイアログボックスを含むすべてのダイアログボックスを閉じます。
- h. Windows のコマンドプロンプトを開き、以下のコマンドを実行してご使用のマシンに Maven が正常にインストールされたことを確認します。

```
mvn --version
```

**結果:**

マシンに Maven が正常にインストールおよび設定されます。

## 第6章 製品のインストール

### 6.1. Red Hat JBoss Data Virtualization のインストール方法

Red Hat JBoss Data Virtualization をインストールする方法は 3 つあります。

- ※ グラフィカルモード: グラフィカルモードでは、Red Hat JBoss Data Virtualization のインストールおよび設定をステップごとに説明するグラフィカルウィザードが実行されます。インストーラーを使用して、クイックスタートや Maven リポジトリなどの追加設定を行うこともできます。
- ※ テキストモード: テキストモードでもインストーラーを実行できます。テキストモードでは、Red Hat JBoss Data Virtualization のインストールおよび設定をステップごとに説明します。
- ※ 自動化スクリプトモード: 自動化スクリプトを使用すると Red Hat JBoss Data Virtualization の同一インスタンスを複数インストールできます。この自動化スクリプトは、最初のインストールインスタンスの後に生成されます。自動化インストールについての詳細は、本書の末尾で扱われます。

### 6.2. JBoss Data Virtualization のインストール: グラフィカルインストール

グラフィカルインストーラーを使用すると、GUI によるステップごとの手順に従って **JBoss Data Virtualization** をマシン上にインストールできます。ここでは、インストーラーを実行するために必要な手順について説明します。

#### 手順6.1 JBoss Data Virtualization のインストール

1. ターミナルウィンドウを開き、GUI インストーラーをダウンロードした場所に移動します。
2. 以下のコマンドを実行して GUI インストーラーを起動します。

```
java -jar jboss-dv-VERSION-installer.jar
```

3. ダイアログボックスが表示された後に、使用許諾契約書が表示されます。契約の内容に同意する場合は **I accept the terms of this license agreement** をクリックし、**Next** をクリックします。
4. Red Hat JBoss EAP がサーバーにインストールされている場所を Red Hat JBoss Data Virtualization に通知するか、製品にバンドルされているためインストールされていない場合は、新しい場所を指定します (Red Hat JBoss EAP がすでにインストールされている場合は、6.4.x の最新バージョンのパッチが適用されていることを確認します)。 **Next** をクリックします。
5. Teiid インストールおよび Modeshape インストールにチェックマークが付いていることを確認し、**Next** をクリックします。
6. 新しい EAP 管理者、Dashboard 管理者、Teiid データアクセスユーザー、および ModeShape ユーザーを作成するよう要求され、OData のアクセスを有効にするかどうかを問われます。これらの管理者とユーザーが作成されると、EAP 管理者は ManagementRealm へ追加され、管理コンソールへアクセスするために使用できます。他のユーザーは ApplicationRealm に追加され、JBoss Data Virtualization の特定コンポーネントへアクセスするために使用できます。新しいユーザー名とパスワードを適切なフィールドに入力し、**Next** をクリックします。

必ずこれらのパスワードをすべて記憶するようにしてください。これらのパスワードはシステムの異なる部分へのアクセスを可能にします。EAP アカウントおよびパスワードは EAP サーバーの管理、Dashboard のパスワードは Web インターフェースに関連する管理機能、Teiid データアクセスユーザーおよび ModeShape ユーザーは標準的なユーザーアクセスを可能にします。

ユーザー名と同じパスワードは使用できません。また、パスワードは 8 文字以上である必要があり、英字、数字、および英数字以外の文字がそれぞれ 1 文字以上含まれる必要があります。

7. Red Hat JBoss Data Virtualization は、デフォルト設定または追加設定オプションでインストールできます。ここでは、デフォルトのみを使用するため、Perform default configuration を選択し、デフォルトのオプションで Red Hat JBoss Data Virtualization をインストールします。**Next** をクリックします。
8. Configure password vault の画面が表示されます。6 文字以上のパスワードを入力します。**Next** をクリックします。



### 注記

デフォルトの H2 データベースは本番環境のデータベースには適していません。テストおよび評価環境のみで使用してください。

9. インストールの概要が表示されます。Next をクリックしてインストールを開始します。インストールには時間がかかることがあります。すべてのコンポーネントがインストールされたら **Next** をクリックします。
10. 自動スクリプトを生成したい場合は **Generate an automatic installation script** をクリックします。これを選択すると、毎回ウィザードを使用しなくても初回のインストールで指定した設定を使用して、製品を迅速に再インストールしたり、大量にデプロイしたりすることができます。



### 注記

Red Hat JBoss Data Virtualization 6.0 用に作成された自動インストールスクリプトは Red Hat JBoss Data Virtualization 6.1 では動作しないため注意してください。

11. **Done** をクリックし、インストーラーを終了します。

### 結果:

Red Hat JBoss Data Virtualization が正常にインストールされ、設定されます。



### 注記

JBoss Data Virtualization をインストール後に別の場所へ移動すると、FileNotFoundException 例外が発生することがあります。これは、一部のファイルパスは JBoss EAP サーバーによってハードコードされることが原因です。



### 警告

データが格納される JDK とは別の JDK で作成されたキーストアを用いて vault の使用を試みると、サーバーの起動に失敗します。vault にアクセスするには同じ JDK を一貫して使用する必要があります。

## 6.3. テキストベースのインストーラーを使用した JBoss Data Virtualization のインストール

テキストベースのインストーラーを使用して Red Hat JBoss Data Virtualization をインストールできます。このモードでは、グラフィカルウィザードを使用せずにインストールの手順を実行します。ディスプレイサーバーが利用できない場合は、GUI インストーラーは自動的にテキストモードで実行されます。

### 前提条件

[カスタマーポータル](#)から Red Hat JBoss Data Virtualization の jar ファイルをダウンロードしておく必要があります。

### 手順6.2 JBoss Data Virtualization のインストール

1. ターミナルウィンドウを開き、GUI インストーラーをダウンロードした場所に移動します。
2. 以下のコマンドを入力し、インストールプロセスを開始します。

```
java -jar jboss-dv-VERSION-installer.jar -console
```

3. ターミナルに表示されるインストールプロンプトに従います。デフォルト設定でインストールするか、追加の設定を行います。
4. 最後の手順で、自動インストールスクリプトを生成します。このスクリプトを使用すると、ヘッドレスインストールを実行したり、複数のインスタンスで同一インストールを実行したりすることができます。

### 結果:

Red Hat JBoss Data Virtualization が正常にインストールされ、設定されます。

## 6.4. JBoss Data Virtualization 開発ツールのインストール

### 前提条件

以下のソフトウェアがインストールされている必要があります。

- ※ Red Hat JBoss Developer Studio (『Red Hat JBoss Developer Studio Installation Guide』を参照してください)。
- ※ 圧縮ファイルの内容を抽出するアーカイブツール。
- ※ Open JDK (『Red Hat JBoss Data Virtualization インストールガイド』を参照) またはサポートされる他の Java 仮想マシン

### 手順6.3 JBoss Developer Studio Integration Stack のインストール

1. **Red Hat JBoss Developer Studio** を起動します。
2. **Red Hat JBoss Developer Studio** でメインメニューから **Help > Install New Software...** アクションをクリックします。
3. **Available Software** ページで **Add ...** ボタンをクリックします。
4. **Add Repository** ダイアログで以下の詳細を入力します。

**Name** フィールドに JBDSIS (または別の一意な名前) を入力します。

**Location** フィールドに <https://devstudio.redhat.com/9.0/stable/updates/integration-stack/earlyaccess/> を入力します。

5. **OK** をクリックします。

6. アップデートサイトのツリービューで **JBoss Data Virtualization Development** フォルダーとすべての子を選択します。



#### 注記

JBDSIS がすでにインストールされている場合は、次の手順に進みます。**What is already installed** リンクをクリックすると、JBDSIS がマシンにインストールされているかを確認できます。

7. **Next** をクリックします。

8. 追加の依存関係とライセンス契約に同意し、**Finish** をクリックして完了します。

インストールに完了すると、新しい機能が完全に動作するよう **Red Hat JBoss Developer Studio** の再起動を要求されます。

#### 結果:

Teiid Designer および ModeShape Tools が正常にインストールされます。

## 第7章 製品の開始および停止

### 7.1. JBoss Data Virtualization の起動

#### 前提条件

以下のソフトウェアがインストールされている必要があります。

- ✧ Red Hat JBoss Data Virtualization

#### 手順7.1 JBoss Data Virtualization の起動

##### ✧ JBoss EAP サーバーの起動

JBoss EAP server を起動すると JBoss Data Virtualization を実行できます。以下のように JBoss EAP server を起動します。

##### A. Red Hat Enterprise Linux

ターミナルを開き、以下のコマンドを入力します。

```
$ EAP_HOME/bin/standalone.sh
```

##### B. Microsoft Windows

ターミナルを開き、以下のコマンドを入力します。

```
$ EAP_HOME\bin\standalone.bat
```

#### 注記

エラーがないことを確認するには、サーバーログ `EAP_HOME/LOG/server.log` を確認します。[管理コンソール](#)を開き、登録された JBoss EAP 管理ユーザーのユーザー名とパスワードでログインします。管理コンソールの使用についての詳細は『Red Hat JBoss Enterprise Application Platform 管理および設定ガイド』を参照してください。

#### 注記

詳細な起動オプションについては『Red Hat JBoss Enterprise Application Platform 管理および設定ガイド』を参照してください。

### 7.2. JBoss EAP 6 を管理対象ドメインとして起動

#### 操作の順序

ドメイン内のサーバーグループのスレーブサーバーを起動する前にドメインコントローラーを起動する必要があります。最初に、この手順をドメインコントローラーで使用した後に、関連するホストコントローラーおよびドメインに関連する他のホストに対して使用してください。開始する前に、Red Hat JBoss EAP ドキュメント (<https://access.redhat.com/documentation/en/red-hat-jboss-enterprise-application-platform/>) を参照してください。

## 手順7.2 プラットフォームサービスを管理対象ドメインとして起動

### 1. Red Hat Enterprise Linux の場合

コマンド `EAP_HOME/bin/domain.sh` を実行します。

### 2. Microsoft Windows Server の場合

コマンド `EAP_HOME\bin\domain.bat` を実行します。

### 3. 他のパラメーターを起動スクリプトに渡す (任意)

起動スクリプトに使用できるすべてのパラメーターを表示するには、**-h** パラメーターを使います。

#### 結果:

JBoss EAP 6 管理対象ドメインインスタンスが起動します。

## 7.3. JBoss Data Virtualization の停止

**JBoss Data Virtualization** を停止するには **JBoss EAP server** を停止する必要があります。JBoss EAP を停止する方法は、JBoss EAP を起動した方法によって異なります。ターミナルで **CTRL+C** を押すと JBoss EAP を停止できます。



#### 注記

他の方法で JBoss EAP サーバーを停止する場合は『Red Hat JBoss Enterprise Application Platform 管理および設定ガイド』を参照してください。

#### 結果:

サーバーがシャットダウンします。シャットダウンには数分かかることに注意してください。

## 第8章 ドメインモード

### 8.1. ドメインモードでのインストール

複数の Red Hat JBoss EAP インスタンスがサーバーファームにあり、それらすべてをドメインモードで起動する場合、このサーバーファームのすべての設定オプションは一元的に管理することができます。たとえば、単一の CLI ベースの呼び出しにより、すべてのインスタンス間で VDB をデプロイしたり、データソースを作成したりすることができます。Red Hat JBoss Data Virtualization はこの設定コンセプトを拡張し、VDB やトランスレーターのサーバーファーム全体でのデプロイを可能にします。

ドメインモードと「HA (high availability)」プロファイルが組み合わさると、デプロイされている Red Hat JBoss EAP サーバーインスタンスをクラスター化できます。(HA プロファイルは「domain.xml」ファイルでデフォルトとして設定されています。)「domain.xml」ファイルを使用してサーバーを起動する場合、ResultSet キャッシュと内部マテリアライズキャッシュに使用される分散キャッシュが自動的に設定されます。管理 API の使用法はスタンドアロンモードとドメインモードのどちらの場合でも同じです。

クラスター内で複数の Red Hat JBoss Data Virtualization インスタンスが使用可能な場合、負荷分散やフェイルオーバー機能を使用することができます。

1. 「Domain」(ドメイン) モードでサーバーを起動するには、クラスターの一部になるすべてのサーバーに Red Hat JBoss Data Virtualization をインストールします。それらのサーバーの 1 つを「マスター」ドメインコントローラーとして選択します。(残りのサーバーは、すべての管理操作に関連して「マスター」ドメインコントローラーに接続するスレーブになります。)
2. すべてのサーバーを設定したら、「マスター」ノードを起動します: **/bin/domain.sh**
3. 「スレーブ」ノードを起動します: **/bin/domain.sh**

スレーブノードは「マスター」ノードからそれぞれのドメイン設定の設定情報を取得します。

「JBoss EAP 6 を管理対象ドメインとして起動」セクションを参照してください。

4. すべてのサーバーが実行されたら、「マスター」ノードで以下のコマンドを実行してインストールを完了し、ドメインモードで実行します: **/bin/jboss-cli.sh --file=[JBOSS\_HOME\_DIR]/cli-scripts/teiid-domain-mode-install.cli**

これはドメイン (クラスター) のインストールごとに 1 回のみ実行する必要があります。このスクリプトは Red Hat JBoss Data Virtualization を ha および full-ha プロファイルでインストールします。さらに、HA プロファイルを開始するために main-server-group を再設定します。ドメインモードに入ると、リソースをドメイン/デプロイメントディレクトリーにドロップすることでそれらを静的にデプロイすることができないため、このスクリプトは、CLI インターフェースでデフォルトリソース (ファイル、ldap、salesforce および ws connector など) をデプロイします。

5. Red Hat JBoss Data Virtualization を HA 以外のプロファイルでインストールする必要がある場合は、インストールの前に teiid-domain-mode-install.cli ファイルを編集し、プロファイル、ソケットバインディングおよびサーバーグループに適宜変更を加えます。
6. VDB がデプロイされると、ユーザーはそれぞれの JDBC アプリケーションを Red Hat JBoss Data Virtualization に接続できるようになります。



## 注記

Teiid Designer は「ドメイン」モードでは Red Hat JBoss Data Virtualization Server に接続できません。ドメインモードは本番環境で使用されることが想定されるため、Red Hat ではデプロイとテストに他のデプロイメント方法を採用されることを推奨しています (CLI、web コンソールなど)。Teiid Designer は開発時の作業のみを支援することを目的としているため、テスト環境のみで使用する必要があります。

## 第9章 JBoss Data Grid キャッシュのインストール

### 9.1. JBoss Data Grid アダプターの設定

infinispan-cache および infinispan-cache-dsl トランスレーターはサーバーの起動時に動作するよう事前設定されていません。そのため、手動で設定する必要があります。

1. **docs/teiid/datasources/infinispan/** ディレクトリーへ移動します。
2. **add-infinispan-cache-translator.cli** または **add-infinispan-cache-dsl-translator.cli** のスクリプトを実行します。

## 第10章 ODBC サポート

### 10.1. はじめに

#### 10.1.1. JBoss Data Virtualization と ODBC

JBoss Data Virtualization は、PostgreSQL データベースサーバーをエミュレートして、デプロイされた VDB へ Open Database Connectivity (ODBC) のアクセスを提供します。

ODBC 準拠のクライアントアプリケーションは、サポートされる PostgreSQL ODBC ドライバーを使用して VDB へ接続できます。サポートされる PostgreSQL ODBC ドライバーは <http://access.redhat.com> からダウンロードできます。

アプリケーションによる ODBC の使用を可能にするには、最初にアプリケーションが実行されているマシンに ODBC ドライバーをインストールし、VDB の接続プロファイルを表すデータソース名 (DSN) を作成する必要があります。



#### 警告

現在 JBoss Data Virtualization は、ODBC のプレーンテキストパスワード認証のみをサポートしています。クライアントとサーバーが SSL を使用するよう設定されていない場合は、ネットワーク上でプレーンテキストのパスワードが送信されます。SSL を使用せずに送信するパスワードをセキュア化する場合、クライアントからの安全なパスワード値 (暗号化やハッシュ化など) を許可するセキュリティードメインのインストールを検討してください。

### 10.2. Red Hat Enterprise Linux に ODBC ドライバーをインストール

#### 10.2.1. Red Hat Enterprise Linux に ODBC ドライバーをインストール

##### 前提条件

- ✦ 管理者権限が必要です。

##### 手順10.1 Red Hat Enterprise Linux に ODBC ドライバーをインストール

###### 1. ドライバーのダウンロード

<https://access.redhat.com/jbossnetwork/> から適切な ODBC ドライバーパッケージ (`jboss-dv-psqlodbc-[version]-X.rpm`) をダウンロードしてください。

###### 2. パッケージのインストール

`sudo yum localinstall jboss-dv-psqlodbc-[version]-X.rpm` を実行します。



#### 注記

他のオペレーティングシステムのインストールパッケージは <https://access.redhat.com/jbossnetwork/> からダウンロードできます。

## 10.2.2. ODBC 環境の設定

### ✧ 環境の設定

Run the `/opt/redhat/jboss-dv/v6/psqlodbc/etc/setenv.sh` スクリプトを実行します。

```
[localhost etc]$ ./setenv.sh
```

このスクリプトは必要なディレクトリーを `LD_LIBRARY_PATH` および `PATH` 環境変数へ追加します。このスクリプトは、ドライバーを使用するたびに実行する必要があります。

## 10.2.3. Linux インストールの DSN の設定

- ✧ `/opt/redhat/jboss-dv/v6/psqlodbc/etc/odbc.ini` ファイルを編集し、適切なユーザー名、パスワードおよびデータベースを適用して更新します。データベース名は VDB 名です。

JBoss Data Virtualization では、ODBC はデフォルトで **35432** ポートで有効になっています。

## 10.3. Microsoft Windows に ODBC ドライバーをインストール

### 10.3.1. Microsoft Windows に ODBC ドライバーをインストール

#### 前提条件

- ✧ 管理者権限が必要です。

#### 手順10.2 Microsoft Windows に ODBC ドライバーをインストール

1. <https://access.redhat.com/jbossnetwork/> から適切なドライバーパッケージ (`jboss-dv-psqlodbc-[バージョン]-X.zip`) をダウンロードしてください。
2. インストールパッケージを展開します。
3. `jboss-dv-psqlodbc-[version]-X.msi` ファイルをダブルクリックしてインストーラーを起動します。
4. インストーラーウィザードが表示されます。**Next** をクリックします。
5. 使用許諾契約書が表示されます。契約書の内容に同意する場合は **I accept the terms in the License Agreement** をクリックし、**Next** をクリックします。
6. 表示されたデフォルト以外のディレクトリーにインストールしたい場合は、**Browse** ボタンをクリックし、ディレクトリーを選択します。**Next** をクリックします。
7. 確認画面が表示されます。選択した内容を確認し、正しければ **Next** をクリックしてインストールを開始します。
8. インストールが終了したら、インストールプロセスの終了を知らせる画面が表示されます。**Finish** をクリックしてウィザードを終了します。

**注記**

他のオペレーティングシステムのインストールパッケージは <http://access.redhat.com> からダウンロードできます。

### 10.3.2. Windows インストールの DSN の設定

#### 手順10.3 Windows インストールの DSN の設定

1. ODBC ドライバーの基本オプションを設定します。
2. ODBC ドライバーのデータソースオプションを設定します。
3. ODBC ドライバーのグローバルオプションを設定します。

## 10.4. Solaris に ODBC ドライバーをインストール

### 10.4.1. Solaris に ODBC ドライバーをインストール

#### 前提条件

- ※ 管理者権限が必要です。

#### 手順10.4 Solaris に ODBC ドライバーをインストール

1. ドライバーのダウンロード

<https://access.redhat.com/jbossnetwork/> から適切な ODBC ドライバーパッケージ (**jboss-dv-psqlodbc-[VERSION]-X.zip**) をダウンロードしてください。

2. インストールパッケージの展開

**/opt** ディレクトリーでインストールパッケージを展開します。

3. **PATH** プロパティの設定

ドライバーを展開したディレクトリーから ODBC バイナリーが使用されるよう **PATH** プロパティを設定します。

```
$ export PATH = $PATH:/opt/redhat/jboss-dv/v6/psqlodbc/bin
```

4. 環境変数の設定

既存の **odbc.ini** に **ODBCINI** 環境変数を設定します。

```
$ export ODBCINI=/opt/redhat/jboss-dv/v6/psqlodbc/etc/odbc.ini
```

**注記**

Solaris ターミナルとして Bourne Shell を使用している場合、上記 2 つのエクスポートコマンドを `~/.profile` ファイルに追加すると、毎回実行する必要がなくなります。同様に、Bash を使用している場合は `~/.bash_profile` ファイルに追加します。

### 10.4.2. Solaris インストールの DSN の設定

- ※ `/opt/redhat/jboss-dv/v6/psqlodbc/etc/odbc.ini` ファイルを編集し、適切なユーザー名、パスワードおよびデータベースを適用して更新します。データベース名は VDB 名です。

JBoss Data Virtualization では、ODBC はデフォルトで **35432** ポートで有効になっています。

## 10.5. ODBC の設定

### 10.5.1. Microsoft Windows での ODBC オプションの設定

#### 前提条件

- ※ 管理権限を使用してワークステーションにログインする必要があります。
- ※ コントロールパネルのデータソース (ODBC) アプレットを使用して新しいデータソース名が追加されている必要があります。

設定する各データソース名は Teiid System 内の 1 つの VDB のみにアクセスできます。複数の VDB を利用できるようにするには、複数のデータソース名を設定する必要があります。

#### 手順10.5 Microsoft Windows でのデータソース名 (DSN) の設定

1. **Start** メニューから **Settings - Control Panel** と選択します。
2. **Control Panel** が表示されます。**Administrative Tools** をダブルクリックします。
3. **Data Sources (ODBC)** をダブルクリックします。
4. **ODBC Data Source Administrator** アプレットが表示されます。追加する DSN のタイプに関連するタブをクリックします。
5. Create New Data Source ダイアログボックスが表示されます。**Select a driver for which you want to set up a data source** テーブルで **PostgreSQL Unicode** を選択します。
6. **Finish** をクリックします。
7. **Data Source Name** 編集ボックスで、このデータソースに割り当てる名前を入力します。
8. **Database** 編集ボックスで、このデータソースからアクセスする仮想データベースの名前を入力します。
9. **Server** 編集ボックスで、Teiid ランタイムのホスト名または IP アドレスを入力します。



## 注記

ファイアウォールまたは NAT アドレスを介して接続している場合は、ファイアウォールアドレスまたは NAT アドレスを入力する必要があります。

10. **Port** 編集ボックスで、システムが ODBC リクエストをリッスンするポート番号を入力します (Red Hat JBoss Data Virtualization のデフォルトでは、ODBC リクエストをポート 35432 でリッスンします)。
11. **User Name** および **Password** 編集ボックスで、Teiid ランタイムアクセスのユーザー名およびパスワードを入力します。
12. SSL モードは無効 (disabled) のままにしておきます (現在、SSL 接続はサポートされていません)。
13. データソースの詳細を **Description** フィールドに入力します。
14. **Datasource** ボタンをクリックし、オプションを設定します。Parse Statements、Recognize Unique Indexes、Maximum、Text as LongVarChar、および Bool as Char にチェックマークを入れます。MaxVarChar を 255、Max LongVarChar を 8190、Cache Size を 100、SysTable Prefixes を dd\_ に設定します。  
  
2 ページ目では、LF、Server side prepare、default、7.4+ をクリックします。Extra Opts を 0x0 に設定します。
15. **Save** をクリックします。

Red Hat JBoss Data Virtualization が稼働している場合は、任意で **Test** をクリックすると接続を検証できます。

## 結果:

Excel や Access などのアプリケーションを使用して VDB でデータをクエリーできるようになりました。

**表10.1 Red Hat JBoss Data Virtualization 用の主な ODBC 設定**

名前	説明
Updateable Cursor および Row Versioning	使用不可。
serverside prepare、Parse Statements および Disallow Premature の使用	「Use serverside prepare」を有効にし、「Parse Statements」/「Disallow Premature」を無効にすることを推奨します。
SSL モード	『セキュリティガイド』を参照。
Declare/Fetch カーソルおよび Fetch Max Count の使用	大規模な結果セットが使用される場合にリソースを効率よく管理するために使用する必要があります。

ログ/デバッグ設定を必要に応じて使用できます。

「Show SystemTables」、「True is -1」、「Backend genetic optimizer」、「Bytea as LongVarBinary」、「Bools as Char」などのデータタイプ、メタデータ、または最適化を操作する設定は Teiid サーバーによって無視され、クライアントへの影響はありません。これら、または影響を定義するための設定が必要な場合は、製品/プロジェクトに関連するケースとして報告してください。

必要な場合は「LFto CR/LF conversion」などのクライアント側に影響を与える他の設定を使用できますが、現時点ではこの設定をサーバー側で使用することはできません。

## 10.5.2. Red Hat Enterprise Linux での ODBC オプションの設定

## 手順10.6 Red Hat Enterprise Linux での ODBC オプションの設定

1. **yum install unixODBC** コマンドを実行し、ドライバーマネージャーをインストールします。
2. **odbcinst -q -d** コマンドを実行し、PostgreSQLdriver が正しくインストールされたことを確認します。
3. DSN を作成するには、**sudo vi /opt/redhat/odbc.ini** を実行して、テキストエディターで設定ファイルを開きます。



## 注記

このファイルを開くには、sudo を使用するか root でログインしている必要があります。

4. 以下の設定をファイルに追加します。

```
[<DSN name>] Driver = /usr/lib/psqlodbc.so
Description = PostgreSQL Data Source
Servername = <Teiid Host name or ip>
Port = 35432 Protocol = 7.4
UserName = <user-name>
Password = <password>
Database = <vdb-name>
ReadOnly = no
ServerType = Postgres
ConnSettings = UseServerSidePrepare=1
ByteaAsLongVarBinary=1
Optimizer=0
Ksqo=0 Trace =
No TraceFile = /var/log/trace.log
Debug = No DebugFile = /var/log/debug.log
```

5. ファイルを保存して、テキストエディターを終了します。
6. 以下のコマンドを実行して DSN をテストします。

```
isql <DSN-name> [<user-name> <password>] < commands.sql
```



## 注記

DSN を使用せずに接続するには、以下の DSN を使用しない接続文字列を使用します。

```
ODBC;DRIVER={PostgreSQL};DATABASE=<vdb-name>;SERVER=<host-name>;PORT=<port>;Uid=<username>;Pwd=<password>
```



## 注記

isql を実行した後にエラーが発生して「Can't open lib '/opt/redhat/jboss-dv/v6/psqlodbc/lib64/psqlodbc.so': file not found」というメッセージが表示される場合、それは一部の postgres ライブラリーがないことを意味します。

この問題を解決するには、root で **yum install postgres** コマンドを実行します。

パッケージがインストールされていることを確認するには、**rpm -qa|grep post** コマンドを実行します。

postgresql と postgresql-jdbc パッケージが一覧表示されるはずです。

## 10.6. DSN を使用しない接続

DSN を明示的に作成せずに ODBC を使用して JBoss Data Virtualization VDB へ接続することもできます。しかし、このような場合にはアプリケーションに DSN を使用しない接続文字列が必要になります。接続文字列の例を以下に示します。

Windows の場合:

```
ODBC;DRIVER={PostgreSQL Unicode};DATABASE=<vdb-name>;SERVER=<host-name>;PORT=<port>;Uid=<username>;Pwd=<password>;c4=0;c8=1;
```

\*nix の場合

```
ODBC;DRIVER={PostgreSQL};DATABASE=<vdb-name>;SERVER=<host-name>;PORT=<port>;Uid=<username>;Pwd=<password>;c4=0;c8=1;
```

## 第11章 自動インストール

### 11.1. 自動インストール

同じ設定で Red Hat JBoss 製品をインストールする必要がある場合、インストールスクリプトを使用すると時間を節約できます。事前定義された設定のインストールスクリプトを使用すると、インストールの手順を毎回行う必要がなく、1つのコマンドを実行してインストール全体を実行できます。インストールスクリプトを生成するには、インストーラー (グラフィカルまたはテキストモード) を実行し、希望の設定を指定した後、プロセスの最後の方で指示されたときにスクリプトの生成を選択します。

#### 前提条件

- ※ <https://access.redhat.com/jbossnetwork/> から関連するインストーラー JAR ファイルをダウンロードしておく必要があります。
- ※ インストーラー (グラフィカルまたはテキストモード) を使用してインストールを行ったときにスクリプトを生成し、XML ファイルとして保存しておく必要があります。

#### 手順11.1 スクリプトを用いたインストール

```
※ java -jar jboss-PRODUCT-installer-VERSION.jar SCRIPT.xml
```

#### 結果:

製品が正常にインストールされ、スクリプトの指定どおりに設定されます。

## 第12章 クラウド環境での実行

### 12.1. Amazon AWS Cloud インスタンスでの Red Hat JBoss Data Virtualization の実行

標準的なインストール手順を行った後、以下の手順に従って Red Hat Data Virtualization を AWS と使用します。

#### 手順12.1 Amazon AWS Cloud インスタンスでの Red Hat JBoss Data Virtualization の実行

1. セキュリティーグループを更新し、ポートを開きます (最低でも TCP、HTTP、および SSH ポートを開く必要があります)。
2. サーバーを起動するには、パラメーター `-Djboss.bind.address.management=0.0.0.0` および `-b 0.0.0.0 example ./standalone.sh -Djboss.bind.address.management=0.0.0.0 -b 0.0.0.0` を追加して、管理およびホストポートをバインドします。
3. Teiid Designer から AWS インスタンスにアクセスするには、JDBS の設定に移動し、**General**, **Network Connections SSH2** を選択します。

次に、**Key Management** タブの **Load Existing Key** を使用して Amazon によって生成されたキーを追加します。

4. サーバー接続を作成するには、**Server Configuration Overview Panel** の **Server Behavior** で **Remote System Deployment** を選択します。必ず **Server is externally managed...** にもチェックマークを入れるようにしてください。

**New Host** ボタンをクリックし、**SSH Only** オプションを選択した後、**Next** をクリックします。

Amazon パブリック IP アドレスと一致するようホスト名を設定し、接続名が同じになるようにします。

**Finish** をクリックします。

5. **Remote Systems** タブを開きます。

新しい接続を右クリックし、**connect** をクリックします。ユーザー ID を入力します (SSH キーが設定されている場合はパスワードを入力する必要はありません)。

6. Server Configuration Overview Panel に戻り、ホストのドロップダウンメニューに新たに作成したホストが選択されていることを確認します。
7. サーバーを起動します (起動したサーバーの状態を切り替えます)。

### 12.2. Google Compute インスタンスでの Red Hat JBoss Data Virtualization の実行

#### 手順12.2 Google Compute インスタンスでの Red Hat JBoss Data Virtualization の実行

1. **Google Developers Console - Compute - Compute Engine - VM Instance** - [インスタンスの名前] - **Network** と選択し、必要なポートを開きます。
2. **Google Developers Console - Compute - Compute Engine - VM Instance** - [インスタンスの名前] - **SSH KEys** と選択し、パブリック SSH キーをアップロードします。

3. 管理ポート (jboss.bind.address.management) を外部インターフェースへバインドします (管理ポートのデフォルト値は **localhost** です)。

## 12.3. Red Hat JBoss Data Virtualization の Azure インスタンスへの接続

以下の手順に従って Azure データベースに接続します。

### 手順12.3 Azure への接続

1. Azure データベースの JDBC 接続文字列を取得します。Azure 管理ポータルでデータベースをクリックしてから「Connect to your database」セクションを見つけます。
2. 「View SQL Database connection strings...」リンクをクリックします。必要な JDBC 文字列が提供されます。
3. このクラウドデータベースでは、これにアクセスするマシンの IP アドレスをファイアウォールルールに登録する必要があります。Azure データベースダッシュボードで、「Manage allowed IP addresses」をクリックしてから、お使いのサーバーの IP アドレスを追加します。これは DV インスタンスがローカルサーバーに置かれている場合には簡単な手順になります。
4. DV インスタンスが OpenShift にデプロイされている場合は OpenShift インスタンスに ssh を実行します。コマンドラインに `ping $OPENSIFT_GEAR_DNS` と入力します。ping 応答の最初の行は以下ようになります。

```
PING ec2-54-221-126-53.compute-1.amazonaws.com
(10.181.128.66) 56(84) bytes of data.
```

ec2 名からサーバーの IP を推測することができます。

```
ec2-54-221-126-53 --> ( 54.221.126.53 )
```

派生した IP アドレスを cloud db firewall rule (クラウドデータベースファイアウォールルール) に登録します。

5. JBoss Data Virtualization はモデル駆動型アプローチを導入しています。最初に、ソースとソースの構造をインポートしてソースモデルを作成します。

Teiid Designer で Teiid Designer perspective を開きます。次に「File」 - 「New」 - 「Teiid Model Project」を選択して新規の「Model Project」を作成します。最初のウィザードページでプロジェクト名として「MyProject」と入力し、「Finish」をクリックします。これでプロジェクトが作成されます。

6. Model Explorer で「MyProject」をクリックしてから「Import...」 - 「JDBC Database」 - 「Source Model」を右クリックします。「Next」をクリックします。
7. 「Import Database via JDBC」ウィザードで、「New...」ボタンをクリックして新規の Connection Profile (接続プロファイル) を作成します。タイプとして「Generic JDBC」を選択し、名前として「AzureCP」を入力します。「Next」をクリックします。
8. 「Drivers」コンボボックスの横にある「New Driver Definition」アイコンを選択して新規ドライバーを作成します。「Generic JDBC Driver template」を選択してから名前に「AzureDriver」を入力します。

9. 「JAR List」タブで「Add JAR/Zip...」ボタンをクリックしてから、ファイルシステムにダウンロードしている SQL Server type 4 driver jar を選択します。「OK」をクリックして「Driver Definition」(ドライバーの定義)の作成を終了します。
10. 接続については、cloud db 接続文字列で指定される実際のデータベース名などを使用した一般的なプロパティーを入力します。
11. 接続文字列から「Optional」(オプション)プロパティーを入力します。
12. 「Test Connection」ボタンをクリックして接続が正常であることを確認します。「Finish」をクリックします。
13. 「JDBC import」ウィザードで操作を継続し、インポートする必要があるテーブルなどを選択します。ソースモデル (AzureSourceModel など) の名前を選択します。インポートウィザードの完了時に、ソースモデルがモデルプロジェクト内に作成されます。

ソースモデルが作成された後に、クラウドデータベースソースのデータをプレビューできます。

#### 手順12.4 データのプレビュー

1. Designer でサーバーが実行中であることを確認します。
2. テーブルを選択し、次にツールバーの「running man」アイコンをクリックします。これにより、クラウドデータベーステーブルの内容が表示されます。
3. 作成したばかりのソースモデルを使ってさらに複雑な変換を作成することもできます。任意の方法でソースの変換と結合を実行するための追加のビューを作成できます。

## 第13章 OpenShift で Red Hat JBoss Data Virtualization Cartridge を実行

### 13.1. カートリッジのインストール

The Data Virtualization Cartridge は、OpenShift 環境向けの Teiid、ModeShape、および Dashboard Builder を提供します。

1. OpenShift ユーザーアカウントを作成します。
2. OpenShift Web コンソールを起動し、<https://openshift.redhat.com/app/console/applications> の Applications ページに移動します。
3. Application タブへ移動し、**Add Application...** を選択します。
4. xPaaS セクションへ移動し、JBoss Data Virtualization cartridge を選択します。
5. アプリケーションの名前 (例: jbossdv) を入力します。
6. アカウントが許可する場合は Medium Gear を選択します。
7. **Create Application** ボタンをクリックします。
8. Red Hat JBoss Data Virtualization Cartridge がデプロイされます。



#### 重要

今後使用するときのために、ユーザー名とパスワードの情報をコピーし、安全な場所に保管しておいてください。

9. **Continue to the application overview page** をクリックし、Data Virtualization カートリッジの概要を確認します。

状態が **Started** であることを確認します。

10. アプリケーションリンクをクリックすると、Data Virtualization の Welcome ページが表示されます。このページにはカートリッジ情報や役立つリンクが含まれています。
11. カートリッジがデプロイされた後に、[http://\[MYAPP\]-\[MYDOMAIN\].rhcloud.com](http://[MYAPP]-[MYDOMAIN].rhcloud.com) へ移動し、状態を確認します。

ユーザー、OData、および REST ロールとともにユーザーが自動的に生成されます。

12. [http://\[MYAPP\]-\[MYDOMAIN\].rhcloud.com/modeshape-webdav](http://[MYAPP]-[MYDOMAIN].rhcloud.com/modeshape-webdav) および [http://\[MYAPP\]-\[MYDOMAIN\].rhcloud.com/modeshape-rest](http://[MYAPP]-[MYDOMAIN].rhcloud.com/modeshape-rest) の ModeShape へアクセスします。

msuser と msadmin の 2 人の ModeShape ユーザーがインストールとともに生成されます。

13. [http://\[MYAPP\]-\[MYDOMAIN\].rhcloud.com/dashboard](http://[MYAPP]-[MYDOMAIN].rhcloud.com/dashboard) の Dashboard Builder へアクセスします。

Dashboard 管理者である dbadmin がインストールとともに生成されます (Teiid ユーザーは Dashboard の読み取り専用アクセスのみ許可されます)。

**重要**

パスワードを取得するため、パブリック OpenShift ssh キーをアプリケーションのメインページに追加し、ssh 経由でマシンに接続するようにしてください。Want to log in to your application? をクリックし、マシンへ接続したらターミナルを開き、`env | grep PASSWORD` を実行してパスワードを取得します。

## 13.2. OpenShift でデータ仮想化 Web インターフェースを使用

### 手順13.1 Box Grinder プラグインのインストール

- 最初に、MySQL データベースを OpenShift Data Virtualization インスタンスに割り当てます。ご使用のデータ仮想化アプリケーションの概要ページに移動し、**Add MySQL 5.5** リンクをクリックします (Databases セクションの下にあります)。

MySQL カートリッジがデプロイされます。完了すると、正常にデプロイされたことを伝える画面が表示されます。

**重要**

後で参照できるよう、必ずクレデンシャルの情報を保存してください。

- Tools and Support** の下にある **Add phpMyAdmin 4.0** リンクをクリックします。これにより、Web インターフェースが追加され、MySQL データベースの管理が容易になります。
- これで、MySQL データベースと管理インターフェースのデプロイメントが完了します。
- phpMyAdmin インターフェースを使用して、一部のデータをデータベースにロードします。  
`https://myApp-myDomain.rhcloud.com/phpmyadmin` で phpMyAdmin カートリッジをインストールしたときに表示されたリンクを使用します。

カートリッジがインストールされたときに取得した root ユーザーと root パスワードを使用します。

左側のツリーパネルで、DV アプリケーション (例: jbosssdv1) の名前と一致するデータベースをクリックします。**SQL** タブを選択します。SQL タブ上でこの DDL をコピーし、テキストエリアに張り付けます。

```
CREATE TABLE PricesTable
(
  ProdCode    CHAR(40) NOT NULL,
  SalePrice   DECIMAL,
  PRIMARY KEY (ProdCode)
);

INSERT INTO PricesTable VALUES
('GC1020', 3499.0),
('GC1040', 19990.0),
('GC1060', 75000.0),
('GC3020', 10200.0),
('GC3040', 38000.0),
('GC3060', 95000.0),
('GC5020', 28000.0),
```

```
( 'GC5040', 59900.0 ),
( 'GC5060', 110000.0 ),
( 'IN7020', 4000.0 ),
( 'IN7040', 16000.0 ),
( 'IN7060', 42000.0 ),
( 'IN7080', 69000.0 ),
( 'SL9020', 4999.0 ),
( 'SL9040', 9999.0 ),
( 'SL9060', 14999.0 ),
( 'SL9080', 19999.0 );
```

最後に、管理インターフェースの右端にある **GO** ボタンをクリックします。

PricesTable テーブルが作成され、事前設定されます。内容をクリックするとその内容を確認できます。

4. Data Virtualization カートリッジを再起動します。これは、MySQL データソースを初期化するために必要になります。アプリケーション概要ページの **Restart application** アイコンをクリックします。
5. <http://www.developerforce.com/events/regular/registration.php?d=701300000000EjHb> へアクセスし、Salesforce アカウントを登録します (アカウントへログインするリンクが含まれる電子メールを受け取ります)。
6. ユーザー名とパスワードを入力して Salesforce インターフェースへログインし、**Personal Setup - Reset My Security Token** へ移動します。セキュリティトークンをリセットします。セキュリティトークンが含まれる別のメールを受け取ります。
7. 以下の例で Salesforce インスタンスへの接続を確立したら、Salesforce のユーザー名を使用します (パスワードは Salesforce パスワードとセキュリティトークンの組み合わせになります)。
8. <http://APP-NAME-DOMAIN-NAME.rhcloud.com/dv-ui> の OpenShift にデプロイされた Web インターフェースに移動します (APP-NAME はアプリケーションに付けた名前に置き換え、DOMAIN-NAME は OpenShift ドメイン名に置き換えます)。
9. **Data Library** 画面から **Create Data Services** リンクをクリックします。**Create Data Service** ページへ移動されます。
10. **Manage Sources** ボタンをクリックします。**Manage Data Sources** ページへ移動されます。
11. MySQLDS ソースをクリックします。これは設定されていますがアクティブでない状態です。ソースの **mysql5** トランスレーターを選択し、**Save Changes** ボタンをクリックします。デプロイされると、ソースはアクティブになるはず (緑色のチェックマークが表示されます)。
12. **Add Source** ボタンをクリックします。MyNewSource というデフォルトの H2 ソースが作成されます。
13. MyNewSource をクリックします。  
表示されたプロパティで Source Type の Name に SalesforceDS を入力し、Salesforce ボタンをクリックします。Ok ボタンをクリックすると、ソースが Salesforce タイプに変更され、トランスレーターが Salesforce に設定されます。Connection Properties で Salesforce アカウントのユーザー名を入力します。パスワードとトークンの組み合わせをパスワードとして入力します。
14. **Save Changes** ボタンをクリックします。各ダイアログボックスに対して Ok をクリックします。Salesforce ソースがデプロイされます (終了するまで数分かかります)。
15. バックリンクをクリックし、Create Data Service ページに戻ります。

16. 次に、MySQL データベーステーブルにアクセスするデータサービスを作成します。サービス名に対して MySQLService を入力します。サービスの説明を入力します。
17. Service Helpers Active Sources タブの MySQLDS をクリックします。
18. Tables で dv61.PricesTable をクリックし、チェックボックスを使って両方の列を選択します。
19. Service View Definition を作成するには Create Service View ボタンをクリックします。Service View Definition テキストエリアが事前設定されます。
20. Test Service ボタンをクリックしてサービスをテストします。
21. Create Data Service をクリックします。エントリーが許可され、サービスが作成されます。Data Service Details ページへリダイレクトされます。
22. Salesforce 専用サービスの作成手順も同様になります。Data Library へ戻り、Create Data Service ボタンをクリックして新しいサービスを作成します。
23. サービス名に SalesforceProductService を入力します。サービスの説明を入力します。
24. Service Helpers Active Sources タブで SalesforceDS をクリックします。
25. Tables で Product2 をクリックし、チェックボックスを使って Name および ProductCode 列を選択します。
26. Service View Definition を作成するには Create Service View ボタンをクリックします。Service View Definition テキストエリアが事前設定されます。
27. 前回と同様にサービスをテストしてサンプルデータを確認し、Create Service をクリックしてサービスを作成します。
28. マッシュアップサービスの作成手順も同様になります。Data Library へ戻り、Create Data Service ボタンをクリックして新しいサービスを作成します。
29. サービス名に MashupService を入力します。サービスの説明を入力します。
30. Service Helpers セクションで Join Definition タブをクリックします (ここで結合を定義します)。
31. Available Sources ドロップダウンメニューで MySQLDS を選択します。Source Tables で PricesTable をクリックします。左側のボタンをクリックし、Left (左) テーブルとして指定します。
32. Available Sources ドロップダウンメニューで SalesforceDS を選択します。Source Tables で Product2 テーブルをクリックします。右側のボタンをクリックし、Right (右) テーブルとして指定します。
33. 左側の Prices テーブルで ProdCode と SalesPrice 列を確認します。右側の Product2 テーブルで Name 列をチェックします。
34. JoinType は Inner Join のままにしておきます。
35. 左側の Join Criteria 列で ProdCode を選択します。
36. 右側の Join Criteria 列で ProductCode を選択します。
37. Apply ボタンをクリックし、Service Definition DDL を生成します。
38. Test Service ボタンをクリックしてサンプルデータを確認します。
39. Create Service ボタンをクリックし、マッシュアップサービスを作成します。Data Service Details ページへリダイレクトされます。Data Service Details ページにはデータのサンプルが表示され、異なる接続オプションの接続手順と URL も提供されます。

40. **Back to Library** リンクをクリックし、Data Library に戻ります。作成した 3 つのサービスが Data Library ページに表示されます。



### 注記

各サービスに使用できるアクションは他にもあります。

- ※ サービスの編集: Edit Data Service ページへリダイレクトします。
- ※ サービスの複製: 選択したサービスのコピーを作成します。
- ※ サービスのテスト: Data Service Details ページへリダイレクトします。
- ※ サービスの削除: 選択したサービスを削除します。
- ※ ファイルへの保存: サービスは Teiid の「動的 VDB」によってサポートされます。このアクションによって動的な VDB xml がファイルへ保存されます。

## 付録A 事前に必要なコンポーネントのインストール

### A.1. Red Hat Enterprise Linux に OpenJDK をインストール

#### 手順A.1 Red Hat Enterprise Linux に OpenJDK をインストール

##### 1. ベースチャンネルのサブスクライブ

RHN ベースチャンネルから OpenJDK を取得します (Red Hat Enterprise Linux のインストールはデフォルトでこのチャンネルにサブスクライブされています)。

##### 2. パッケージのインストール

yum ユーティリティを使用して OpenJDK をインストールします: **yum install java-1.7.0-openjdk-devel**

##### 3. OpenJDK がシステムデフォルトであることを確認

以下の手順に従って、正しい JDK がシステムデフォルトとして設定されていることを確認します。

- a. root で **java** の alternatives コマンドを実行します。

```
/usr/sbin/alternatives --config java
```

- b. **/usr/lib/jvm/jre-1.7.0-openjdk/bin/java** を選択します。

- c. 同様に、**javac** の alternative コマンドを実行します。

```
/usr/sbin/alternatives --config javac
```

- d. **/usr/lib/jvm/java-1.7.0-openjdk/bin/javac** を選択します。

## 付録B 設定情報

### B.1. サポートされるデータソースおよびトランスレーター

以下の表は Red Hat によってサポートされるデータソースおよびトランスレーターの一覧になります。

表B.1 サポートされるデータソースおよびトランスレーター

データソース	ドライバー	トランスレーター
Action Analytics Express 2.0	2.0	action-vector
Amazon Redshift	postgresql 9.2	redshift
Apache Accumulo 1.5.0	該当なし	accumulo
Apache Cassandra 2.2.4	該当なし	cassandra
Apache Hive 2.0.0	2.0.0	hive
Apache Solr 4.9.0	該当なし	solr
Apache Spark 1.6.0	Hive 1.2.1	hive
Cloudera Hadoop 5.5.1	5.5.1	impala
EDS 5.x	現行の Teiid Driver	teiid
ファイル - 区切り、固定長	該当なし	file
汎用データソース -JDBC ansi	該当なし	jdbc-ansi
汎用データソース-JDBC simple (postgresql84)	postgresql 8.4	jdbc-simple
Greenplum 4.x	postgresql 9.0	greenplum
HBase 1.1	phoenix 4.5.1 HBase 1.1	hbase
Hortonworks Hadoop	2.3.4	hive
IBM DB2 10	4.12.55	db2
IBM DB2 9.7	4.12.55	db2
Informix 12.10	4.10.JC5DE	informix
Ingres 10	4.1.4	ingres
JBoss Data Grid 6.4 (ライブラリーモード)	該当なし	infinispan-cache
JBoss Data Grid 6.4 (リモートクライアント - hotrod)	該当なし	infinispan-cache-dsl
LDAP/ActiveDirectory v3	該当なし	ldap
MariaDB	mysql 5.1.22	mysql5
ModeShape/JCR 3.1	3.8.4	modeshape
MongoDB 3.0	該当なし	mongodb
MS Access 2013	該当なし	アクセス
MS Excel 2010	該当なし	excel
MS Excel 2013	該当なし	excel
MS SQL Server 2008	4.0.2206.100	sqlserver
MS SQL Server 2012	4.0.2206.100	sqlserver
MySQL 5.1	mysql 5.1.22	mysql5
MySQL 5.5	mysql 5.1.22	mysql5
Netezza 7.2.x	7.2.1.1	netezza
Oracle 11g RAC	12.1.0.2.0	oracle
Oracle 12c	12.1.0.2.0	oracle

データソース	ドライバー	トランスレーター
PostgreSQL 8.4	postgresql 8.4	postgresql
PostgreSQL 9.2	postgresql 9.2	postgresql
HTTP 上の REST/JSON	該当なし	ws
RHDS 9.0	該当なし	ldap
Salesforce.com API 22.0	NA	salesforce
Salesforce.com API 34.0	NA	salesforce-34
SAP_Hana 1.00.102.01.1444147999	1.00.82.00_03942 70-1510	hana
SAP Netweaver Gateway	odata 4	sap-nw-gateway
Sybase ASE 15	jconn4-26502	sybase
Sybase IQ 16 Express	jconn4-v7	sybaseIQ
Teradata Express 15	15.10.00.05	teradata
Vertica 7.2.1	7.2.1-0	vertica
Webservices	該当なし	ws
XML ファイル	該当なし	FILE



### 注記

MS Excel は書き込みプロシージャがあるためサポートされます。



### 注記

MySQL InnoDB ストレージエンジンは外部マテリアライズのターゲットとしての使用には適していません。

## 付録C 改訂履歴

<b>改訂 6.30-10.1</b>	<b>Mon Jan 23 2017</b>	<b>Terry Chuang</b>
翻訳ファイルを XML ソースバージョン 6.30-10 と同期		
<b>改訂 6.30-10</b>	<b>Wed Oct 26 2016</b>	<b>David Le Sage</b>
バージョン 6.3 用に更新。		
<b>改訂 6.2.0-032</b>	<b>Thu Dec 10 2015</b>	<b>David Le Sage</b>
6.2 用に更新		