



# Red Hat Decision Manager 7.10

## Red Hat Decision Manager のインストールおよび設定

ガイド



# Red Hat Decision Manager 7.10 Red Hat Decision Manager のインストールおよび設定

---

ガイド

Enter your first name here. Enter your surname here.

Enter your organisation's name here. Enter your organisational division here.

Enter your email address here.

## 法律上の通知

Copyright © 2022 | You need to change the HOLDER entity in the en-US/Installing\_and\_configuring\_Red\_Hat\_Decision\_Manager.ent file |.

The text of and illustrations in this document are licensed by Red Hat under a Creative Commons Attribution–Share Alike 3.0 Unported license ("CC-BY-SA"). An explanation of CC-BY-SA is available at

<http://creativecommons.org/licenses/by-sa/3.0/>

. In accordance with CC-BY-SA, if you distribute this document or an adaptation of it, you must provide the URL for the original version.

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

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

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

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

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

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

The OpenStack<sup>®</sup> 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

本書では、Red Hat JBoss EAP クラスター環境、JBoss Web サーバーおよびその他のサポート環境において、Red Hat JBoss EAP に Red Hat Decision Manager コンポーネントをインストールする方法を説明します。

## 目次

前書き .....	6
多様性を受け入れるオープンソースの強化 .....	7
パート I. RED HAT DECISION MANAGER インストールの計画 .....	8
第1章 RED HAT DECISION MANAGER のバージョン管理 .....	9
第2章 RED HAT DECISION MANAGER のコンポーネント .....	11
第3章 RED HAT DECISION MANAGER インストールプラットフォーム .....	12
3.1. RED HAT JBOSS EAP .....	12
3.2. RED HAT JBOSS WEB SERVER .....	12
3.3. RED HAT OPENSIFT CONTAINER PLATFORM .....	12
3.4. ORACLE WEBLOGIC SERVER .....	13
3.5. IBM WEBSHERE APPLICATION SERVER .....	13
第4章 RED HAT DECISION MANAGER でサポートされるリポジトリ .....	14
4.1. GIT リポジトリ .....	14
4.2. APACHE MAVEN .....	14
第5章 RED HAT DECISION MANAGER のインストール環境オプション .....	15
第6章 RED HAT DECISION MANAGER ロールおよびユーザー .....	16
第7章 RED HAT DECISION MANAGER リファレンス実装 .....	17
第8章 RED HAT DECISION MANAGER との統合 .....	18
8.1. RED HAT DECISION MANAGER SPRING BOOT ビジネスアプリケーション .....	18
8.2. RED HAT FUSE の統合 .....	18
8.3. RED HAT SINGLE SIGN-ON の統合 .....	18
第9章 関連資料 .....	19
パート II. RED HAT JBOSS EAP 7.3 への RED HAT DECISION MANAGER のインストールおよび設定 .....	20
第10章 RED HAT DECISION MANAGER について .....	21
第11章 RED HAT DECISION MANAGER ロールおよびユーザー .....	22
第12章 RED HAT DECISION MANAGER インストールファイルのダウンロード .....	23
第13章 RED HAT DECISION MANAGER インストーラーの使用 .....	24
13.1. インタラクティブモードでのインストーラーの使用 .....	24
13.2. CLI モードでのインストーラーの使用 .....	26
第14章 ZIP からの RED HAT DECISION MANAGER のインストール .....	29
14.1. ZIP ファイルからの BUSINESS CENTRAL のインストール .....	29
14.2. ZIP ファイルからの KIE SERVER のインストール .....	29
14.3. ユーザーの作成 .....	30
14.4. BUSINESS CENTRAL に接続する KIE SERVER の設定 .....	32
第15章 キーストアを使用したパスワードセキュリティの確保 .....	35
第16章 SSH が RSA を使用するように設定する手順 .....	37
第17章 RED HAT DECISION MANAGER システムの検証 .....	38

第18章 RED HAT DECISION MANAGER の実行 .....	39
第19章 ヘッドレス DECISION MANAGER コントローラーのインストールおよび実行 .....	41
19.1. ヘッドレス DECISION MANAGER コントローラーのインストール .....	41
19.1.1. ヘッドレス Decision Manager コントローラーの作成 .....	42
19.1.2. KIE Server とヘッドレス Decision Manager コントローラー の設定 .....	42
19.2. ヘッドレス DECISION MANAGER コントローラーの実行 .....	44
第20章 スタンドアロン BUSINESS CENTRAL の設定および実行 .....	46
第21章 RED HAT DECISION MANAGER の MAVEN 設定およびリポジトリ .....	48
21.1. RED HAT DECISION MANAGER の MAVEN 依存関係の追加 .....	48
21.2. BUSINESS CENTRAL および KIE SERVER への外部 MAVEN リポジトリの設定 .....	50
21.3. オフラインで使用する MAVEN ミラーリポジトリの用意 .....	51
第22章 GIT リポジトリからのプロジェクトのインポート .....	54
第23章 BUSINESS CENTRAL ブランディングのカスタマイズ .....	55
23.1. BUSINESS CENTRAL ログインページのカスタマイズ .....	55
23.2. BUSINESS CENTRAL アプリケーションヘッダーのカスタマイズ .....	55
第24章 LDAP と SSL の統合 .....	56
付録A BUSINESS CENTRAL システムプロパティ .....	57
第25章 関連情報 .....	63
第26章 次のステップ .....	64
パート III. RED HAT JBOSS EAP クラスター環境への RED HAT DECISION MANAGER のインストールおよび設定 .....	65
第27章 RED HAT DECISION MANAGER のクラスター .....	66
第28章 開発 (オーサリング) 環境における RED HAT DECISION MANAGER クラスター .....	67
28.1. RED HAT DATA GRID のインストールおよび設定 .....	67
28.2. AMQ BROKER のダウンロードおよび設定 .....	68
28.3. NFS バージョン 4 サーバーの設定 .....	69
28.4. RED HAT JBOSS EAP 7.3 および RED HAT DECISION MANAGER のダウンロードおよび展開 .....	69
28.5. クラスターでの BUSINESS CENTRAL の設定および実行 .....	70
28.6. 高可用性 (HA) オンプレミスインフラストラクチャーのテスト .....	73
28.7. RED HAT DECISION MANAGER クラスターの検証 .....	75
第29章 ランタイム環境における KIE SERVER クラスター .....	76
29.1. RED HAT JBOSS EAP 7.3 および KIE SERVER のダウンロードおよび展開 .....	76
29.2. ヘッドレス DECISION MANAGER コントローラーを使用した KIE SERVER のクラスターリング .....	77
第30章 関連資料 .....	80
パート IV. RED HAT JBOSS WEB SERVER への RED HAT DECISION MANAGER のインストールおよび設定 .....	81
第31章 RED HAT DECISION MANAGER について .....	82
31.1. RED HAT DECISION MANAGER のコンポーネント .....	83
31.2. RED HAT DECISION MANAGER ロールおよびユーザー .....	83
第32章 RED HAT DECISION MANAGER インストールファイルのダウンロード .....	85
第33章 RED HAT DECISION MANAGER インストーラーの使用法 .....	86
33.1. インタラクティブモードでのインストーラーの使用 .....	86

33.2. CLI モードでのインストーラーの使用	88
<b>第34章 KIE SERVER ZIP ファイルのインストールと設定</b>	<b>91</b>
34.1. ZIP ファイルからの KIE SERVER のインストール	91
<b>第35章 キーストアを使用したパスワードセキュリティの確保</b>	<b>93</b>
<b>第36章 KIE SERVER インストールの確認</b>	<b>96</b>
<b>第37章 ヘッドレス DECISION MANAGER コントローラーのダウンロードおよびインストール</b>	<b>97</b>
<b>第38章 スタンドアロン BUSINESS CENTRAL の設定および実行</b>	<b>100</b>
<b>第39章 RED HAT DECISION MANAGER の MAVEN 設定およびリポジトリ</b>	<b>102</b>
39.1. プロジェクト設定ファイル (POM.XML) を使用した MAVEN の設定	102
39.2. MAVEN 設定ファイルの修正	102
39.3. RED HAT DECISION MANAGER の MAVEN 依存関係の追加	103
39.4. オフラインで使用する MAVEN ミラーリポジトリの用意	105
<b>第40章 GIT リポジトリからのプロジェクトのインポート</b>	<b>108</b>
<b>第41章 LDAP と SSL の統合</b>	<b>109</b>
<b>付録B BUSINESS CENTRAL システムプロパティ</b>	<b>110</b>
<b>パート V. IBM WEBSHERE APPLICATION SERVER への KIE SERVER のインストールおよび設定</b>	<b>116</b>
<b>第42章 KIE SERVER</b>	<b>117</b>
<b>第43章 IBM WEBSHERE APPLICATION SERVER</b>	<b>118</b>
<b>第44章 IBM WEBSHERE APPLICATION SERVER のインストールおよび実行</b>	<b>119</b>
<b>第45章 KIE SERVER への IBM WEBSHERE APPLICATION SERVER の設定</b>	<b>121</b>
45.1. 管理セキュリティの有効化	121
45.2. JAVA MESSAGE SERVICE (JMS) の設定	122
45.2.1. サービスバスを作成し、IBM WebSphere Application Server を追加します。	122
45.2.2. JMS 接続ファクトリーの作成	122
45.2.2.1. KIE Server の JMS 接続ファクトリー	123
45.2.3. JMS キューの作成	123
45.2.3.1. KIE Server 向けの JMS キュー	123
45.2.4. JMS アクティベーション指定の作成	124
45.2.4.1. KIE Server の JMS アクティベーション仕様	124
45.3. IBM WEBSHERE APPLICATION SERVER へのシステムプロパティの設定	125
45.4. IBM WEBSHERE APPLICATION SERVER の停止および再起動	128
<b>第46章 IBM WEBSHERE APPLICATION SERVER への KIE SERVER のインストールおよび実行</b>	<b>130</b>
46.1. KIE SERVER のグループおよびロールの作成	131
46.2. KIE SERVER のグループおよびロールのマッピング	131
46.3. KIE SERVER のクラ出力ディレクトリ設定	132
46.4. インストールの検証	132
<b>第47章 IBM WEBSHERE APPLICATION SERVER へのヘッドレス DECISION MANAGER コントローラーのインストールおよび実行</b>	<b>134</b>
47.1. ヘッドレス DECISION MANAGER コントローラーへのクラ出力ディレクトリ設定	135
47.2. ヘッドレス DECISION MANAGER コントローラーのシステムプロパティの設定	135
47.3. ヘッドレス DECISION MANAGER コントローラーのグループおよびロールのマッピング	137
47.4. インストールの検証	137

第48章 IBM WEBSPPHERE APPLICATION SERVER での埋め込みデシジョンエンジンの設定 .....	139
第49章 キーストアを使用したパスワードセキュリティの確保 .....	142
第50章 次のステップ .....	144
パート VI. ORACLE WEBLOGIC SERVER への KIE SERVER のインストールおよび設定 .....	145
第51章 KIE SERVER .....	146
第52章 ORACLE WEBLOGIC SERVER .....	147
第53章 ORACLE WEBLOGIC SERVER のインストールおよび実行 .....	148
第54章 KIE SERVER への ORACLE WEBLOGIC SERVER の設定 .....	150
54.1. KIE SERVER のグループおよびユーザーの設定 .....	150
54.2. JAVA MESSAGE SERVICE (JMS) の設定 .....	150
54.2.1. JMS サーバーの作成 .....	150
54.2.2. JMS モジュールの作成 .....	151
54.2.3. JMS 接続ファクトリーの作成 .....	151
54.2.4. JMS キューの作成 .....	152
54.3. ORACLE WEBLOGIC SERVER へのシステムプロパティの設定 .....	153
54.4. ORACLE WEBLOGIC SERVER の停止および再起動 .....	154
第55章 ORACLE WEBLOGIC SERVER への KIE SERVER のインストール .....	156
55.1. ORACLE WEBLOGIC SERVER への KIE SERVER インストールの確認 .....	156
第56章 ORACLE WEBLOGIC SERVER へのヘッドレス DECISION MANAGER コントローラーのインストールおよび実行 .....	158
56.1. ヘッドレス DECISION MANAGER コントローラーのシステムプロパティの設定 .....	159
56.2. インストールの検証 .....	160
第57章 ORACLE WEBLOGIC SERVER での埋め込みデシジョンエンジンの設定 .....	161
第58章 キーストアを使用したパスワードセキュリティの確保 .....	164
第59章 次のステップ .....	166
パート VII. RED HAT DECISION MANAGER 7.10 の修正およびアップグレード .....	167
第60章 RED HAT DECISION MANAGER のバージョン管理 .....	168
第61章 RED HAT カスタマーポータルで RED HAT DECISION MANAGER の通知の有効化 .....	170
第62章 RED HAT DECISION MANAGER へのパッチ更新およびマイナーリリースアップグレードの適用 ...	171
第63章 関連情報 .....	174
付録C バージョン情報 .....	175
付録D お問い合わせ先 .....	176





## 前書き

開発者またはシステム管理者は、Red Hat JBoss EAP クラスター環境、JBoss Web サーバーおよびその他のサポート環境において、Red Hat JBoss EAP に Red Hat Decision Manager コンポーネントをインストールする方法を説明します。

## 多様性を受け入れるオープンソースの強化

Red Hat では、コード、ドキュメント、Web プロパティにおける配慮に欠ける用語の置き換えに取り組んでいます。まずは、マスター (master)、スレーブ (slave)、ブラックリスト (blacklist)、ホワイトリスト (whitelist) の 4 つの用語の置き換えから始めます。この取り組みにより、これらの変更は今後の複数のリリースに対して段階的に実施されます。詳細は、[弊社の CTO である Chris Wright のメッセージ](#)を参照してください。

## パート I. RED HAT DECISION MANAGER インストールの計画

システム管理者が選択できる Red Hat Decision Manager インストールオプションは複数あります。

## 第1章 RED HAT DECISION MANAGER のバージョン管理

Red Hat Decision Manager のバージョンには、メジャーリリース、マイナーリリース、およびパッチリリースのそれぞれの数字を組み合わせた **メジャーリリース.マイナーリリース.パッチリリース** という書式が使用されています (例: **7.5.1**)。この例では、メジャーリリースが **7.x.x**、マイナーリリースが **7.5.x**、パッチリリースが **7.5.1** となります。メジャーリリースでは、大抵の場合はデータ移行が求められますが、マイナーリリースの更新やパッチの更新は、Red Hat Decision Manager リリースアーティファクトで提供されるアップグレードおよび更新ツールで管理されます。

以下は、Red Hat Decision Manager の一般的なリリースの種類になります。

### メジャーリリースの移行

Red Hat Decision Manager のメジャーリリースには、かなりの数の機能強化、セキュリティ更新、バグ修正が含まれており、再設計された機能も含まれている可能性があります。Red Hat JBoss BRMS 6.4.x から Red Hat Decision Manager 7.0 など、メジャーリリースから別のメジャーリリースにアプリケーションを移行すると、データ移行が必要になる場合があります。移行を容易にするために、Red Hat Decision Manager の新しいメジャーバージョンで、自動化した移行ツールがしばしば提供されますが、場合によっては特定のデータおよび設定を手動で処理する必要があります。サポートされる移行パスは、製品のお知らせおよびドキュメントに指定されています。

たとえば、移行の方法については、[Red Hat JBoss BRMS 6.4 から Red Hat Decision Manager 7.0 への移行](#) を参照してください。

### マイナーリリースのアップグレード

Red Hat Decision Manager のマイナーリリースには、機能拡張、セキュリティ更新、バグ修正が含まれます。Red Hat Decision Manager 7.5.x から 7.6 など、アプリケーションを別のマイナーリリースに移行する場合には、データの移行が必要な場合があります。自動更新ツールは、多くの場合、パッチ更新と Red Hat Decision Manager の新しいマイナーバージョンの両方で提供され、Business Central、KIE Server、ヘッドレス Decision Manager コントローラーなど、Red Hat Decision Manager の特定コンポーネントの更新を容易にします。デシジョンエンジンやスタンドアロンの Business Central など、その他の Red Hat Decision Manager アーティファクトは、各マイナーリリースが含まれる新しいアーティファクトとしてリリースされるため、再インストールして更新を適用する必要があります。

新たなマイナーリリースにアップグレードする前に、お使いの Red Hat Decision Manager に最新のパッチ更新を適用してください。

### パッチの更新

Red Hat Decision Manager のパッチの更新には、最新のセキュリティ更新、およびバグ修正が含まれます。予定されているパッチ更新には、製品のマイナーバージョンに対してすでにリリースされているパッチ更新がすべて含まれるため、更新を最新の状態にするために、パッチの更新を1つ1つ適用していく必要はありません。たとえば、Red Hat Decision Manager 7.5.0 または 7.5.1 は Red Hat Decision Manager 7.5.2 に更新できます。ただし、Red Hat Decision Manager で最適なパフォーマンスを得るには、製品の更新が利用可能になったらすぐに適用する必要があります。また、既存製品の通常の更新サイクルとは別に、予定されていないパッチ更新がリリースされる場合があります。これには、特定の問題を修正するために Red Hat グローバルサポートサービス (GSS) によって提供されるセキュリティまたはその他の更新が含まれる場合があります、累積的な更新ではない場合があります。

自動更新ツールは、多くの場合、パッチ更新と Red Hat Decision Manager の新しいマイナーバージョンの両方で提供され、Business Central、KIE Server、ヘッドレス Decision Manager コントローラーなど、Red Hat Decision Manager の特定コンポーネントの更新を容易にします。デシジョンエンジンやスタンドアロンの Business Central など、その他の Red Hat Decision Manager アーティファクトは、各マイナーリリースが含まれる新しいアーティファクトとしてリリースされるため、再インストールして更新を適用する必要があります。

リリース間での移行を最適に保ち、Red Hat Decision Manager ディストリビューションに最新の機能強化および修正が含まれるようにするには、製品リリースや更新が Red Hat カスタマーポータルで利用可能になった時点で、Red Hat Decision Manager に適用します。Red Hat カスタマーポータルで製品の通知を有効にすることも検討してください。

## 第2章 RED HAT DECISION MANAGER のコンポーネント

Red Hat Decision Manager は、Business Central と KIE Server で設定されます。

- Business Central は、ビジネスルールを作成して管理するグラフィカルユーザーインターフェイスです。Business Central は、Red Hat JBoss EAP インスタンスまたは Red Hat OpenShift Container Platform (OpenShift) にインストールできます。  
Business Central は、スタンドアロンの JAR ファイルとしても使用できます。Business Central スタンドアロンの JAR ファイルとして使用して、アプリケーションサーバーにデプロイせずに Business Central を実行できます。
- KIE Server では、ルール、およびその他のアーティファクトが実行されます。これは、ルールをインスタンス化して実行し、計画の問題を解決するために使用されます。KIE Server は、Red Hat JBoss EAP インスタンス、Red Hat JBoss EAP クラスター、OpenShift、Oracle WebLogic Server インスタンス、IBM WebSphere Application Server インスタンスに、または Spring Boot アプリケーションの一部としてインストールできます。  
KIE Server は、管理モードまたは非管理モードで動作するように設定できます。KIE Server が非管理モードの場合は、手動で KIE コンテナ (デプロイメントユニット) を作成および維持する必要があります。KIE コンテナは、プロジェクトの特定のバージョンです。管理モードの場合は、Decision Manager コントローラーが KIE Server の設定を管理し、ユーザーはコントローラーと対話形式で KIE コンテナを作成して維持します。  
  
Decision Manager コントローラーは Business Central と統合します。Red Hat JBoss EAP に Business Central をインストールする場合は、Execution Server ページを使用して KIE コンテナを作成および維持します。ただし、Business Central をインストールしない場合は、ヘッドレス Decision Manager コントローラーをインストールし、REST API または KIE Server Java Client API を使用してそのコントローラーと対話します。
- Red Hat Business Optimizer は、Business Central および KIE Server に統合されています。また、組み込み可能な軽量プランニングエンジンで、プランニングの問題を最適化します。Red Hat Business Optimizer は、最適化のためのヒューリスティック法およびメタヒューリスティック法を効率的なスコア計算と組み合わせ、Java プログラマーがプランニングの問題を効率的に解決できるようにします。

## 第3章 RED HAT DECISION MANAGER インストールプラットフォーム

Red Hat Decision Manager のリリースはすべて、オペレーティングシステム、JVM、Web ブラウザー、データベースのさまざまな組み合わせで認定されています。Red Hat は、サポートされる設定およびテスト済みの設定に対して、実稼働サポートと開発サポートをお客様のサブスクリプション契約に従って提供します。サポート対象の設定とバージョン番号の詳細は、以下のページを参照してください。

- [Red Hat Decision Manager 7 でサポート対象の設定](#)
- [Red Hat Decision Manager 7 Component Details](#)

Red Hat Decision Manager 7.10 は、以下のアプリケーションプラットフォームで利用できます。

- Red Hat JBoss EAP
- Red Hat JBoss Web Server
- Red Hat OpenShift Container Platform
- Oracle WebLogic Server
- IBM WebSphere Application Server

### 3.1. RED HAT JBOSS EAP

Red Hat JBoss Enterprise Application Platform (Red Hat JBoss EAP) は、Java Enterprise Edition 7 (Java EE 7) の Full Profile 仕様および Web Profile 仕様の認定実装です。Red Hat JBoss EAP には、高可用性クラスタリング、メッセージング、分散キャッシングなどの機能に対する事前設定オプションが用意されています。ユーザーは、Red Hat JBoss EAP が提供するさまざまな API およびサービスを使用して、アプリケーションを開発、デプロイ、および実行することもできます。

Business Central と KIE Server の両方を単一の Red Hat JBoss EAP インスタンスにインストールできます。ただし、実稼働環境では、別のインスタンスにインストールする必要があります。

### 3.2. RED HAT JBOSS WEB SERVER

Red Hat JBoss Web Server は Tomcat をベースとしたエンタープライズレベルの Web サーバーで、中規模および大規模のアプリケーション用に設計されています。Red Hat JBoss Web Server は、Java Server Pages (JSP) および Java Servlet テクノロジー、PHP、ならびに CGI をデプロイするための単一プラットフォームを提供します。

KIE Server とヘッドレス Decision Manager コントローラーは、Red Hat JBoss Web Server にインストールすることができます。

### 3.3. RED HAT OPENSIFT CONTAINER PLATFORM

Red Hat OpenShift Container Platform (OpenShift) は、Kubernetes を使用し、API を提供してこれらのサービスを管理します。OpenShift を使用すると、コンテナを作成および管理できます。

この場合、Red Hat Decision Manager のコンポーネントは、別の OpenShift Pod としてデプロイされます。各 Pod のスケールアップとダウンを個別に行い、特定のコンポーネントに必要な数だけコンテナを提供できます。標準の OpenShift の手法を使用して Pod を管理し、負荷を分散できます。



OpenShift には、Business Central と KIE Server の両方をインストールできます。

### 3.4. ORACLE WEBLOGIC SERVER

Oracle WebLogic Server は、分散型 Java アプリケーションを作成するために API の標準セットを提供する Java EE アプリケーションサーバーで、データベース、メッセージングサービス、外部のエンタープライズシステムへの接続など、さまざまなサービスにアクセスできます。ユーザーは、Web ブラウザクライアントまたは Java クライアントを使用してこれらのアプリケーションにアクセスします。

Oracle WebLogic Server インスタンスに KIE Server をインストールし、ヘッドレス Decision Manager コントローラーと、REST API または KIE Server Java Client API を使用して KIE Server と対話します。また、スタンドアロンの Business Central を使用して KIE Server と対話することもできます。

### 3.5. IBM WEBSPHERE APPLICATION SERVER

IBM WebSphere Application Server は、Java ベースの Web アプリケーションをホストし、Java EE 認定ランタイム環境を提供する、柔軟性がある安全な Web アプリケーションです。IBM WebSphere 9.0 は Java SE 8 に対応しており、バージョン 8.5.5.6 以降の Java EE 7 に完全に準拠しています。

IBM WebSphere Application Server インスタンスに KIE Server をインストールし、ヘッドレス Decision Manager コントローラーを使用して、KIE Server と対話します。また、スタンドアロンの Business Central を使用して KIE Server と対話することもできます。

## 第4章 RED HAT DECISION MANAGER でサポートされるリポジトリ

Git リポジトリと Apache Maven を Red Hat Decision Manager で使用できます。

### 4.1. GIT リポジトリ

Git リポジトリは Business Central 内で使用され、オーサリング環境で作成したプロセス、ルール、その他のアーティファクトをすべて保存します。Git は分散バージョン管理システムです。リビジョンをコミットオブジェクトとして実装します。変更をリポジトリにコミットすると、Git リポジトリに新規コミットオブジェクトが作成されます。Business Central でプロジェクトを作成すると、Business Central に接続される Git リポジトリに追加されます。

他の Git リポジトリにプロジェクトがある場合は、それらを Business Central のスペースにインポートできます。Git フックを使用して、内部 Git リポジトリを外部リポジトリに同期できます。

### 4.2. APACHE MAVEN

[Apache Maven](#) は分散型構築自動化ツールで、ソフトウェアプロジェクトのビルドおよび管理を行うために Java アプリケーション開発で使用されます。Maven を使用して、ご自分の Red Hat Decision Manager プロジェクトをビルド、公開、およびデプロイすることができます。Maven には以下のメリットがあります。

- ビルドプロセスが容易で、すべてのプロジェクトに対して統一された構築システムが実装される。
- プロジェクトに必要なすべての JAR ファイルがコンパイル時に利用可能になる。
- 適切なプロジェクト構造が設定される。
- 依存関係およびバージョンが適切に管理される。
- Maven では事前定義されたさまざまな出力タイプ (JAR および WAR 等) にビルドされるので、追加のビルドプロセスが不要である。

Maven はレポジトリを使用して Java ライブラリー、プラグイン、および他のビルドアーティファクトを格納します。これらのリポジトリは、ローカルまたはリモートいずれかの形態をとることができます。Red Hat Decision Manager によりローカルおよびリモート maven リポジトリが維持され、それをご自分のプロジェクトに追加してルール、プロセス、イベント、およびその他のプロジェクト依存関係にアクセスすることができます。プロジェクトおよびアーキタイプをビルドする際に、Maven はローカルまたはリモートリポジトリから Java ライブラリーおよび Maven プラグインを動的に取得します。これにより、プロジェクト全体を通じて依存関係の共有および再利用が促進されます。

## 第5章 RED HAT DECISION MANAGER のインストール環境オプション

Red Hat Decision Manager を使用して、ビジネスアプリケーションを開発する開発環境、デシジョンをサポートするアプリケーションを実行するランタイム環境、またはその両方を設定できます。

- **開発環境:** 通常、Business Central インストール1つと KIE Server インストール1つ以上で設定されます。開発者は Business Central を使用して、デシジョンやその他のアーティファクトを作成できるほか、KIE Server を使用して、作成したアーティファクトをテストできます。
- **ランタイム環境:** 1つ以上の KIE Server インスタンスに Business Central があり/なしの形で設定されます。Business Central には Decision Manager コントローラーが組み込まれています。Business Central をインストールしている場合は、**Menu → Deploy → Execution Server** ページを使用してコンテナを作成および維持します。Business Central を使用せずに KIE Server の管理を自動化するには、ヘッドレス Decision Manager コントローラーを使用します。

また、開発環境およびランタイム環境の両方をクラスター化することが可能です。クラスター化した開発環境またはランタイム環境は、統合グループまたは2台以上のサーバーのクラスターで設定されます。Red Hat Decision Manager の開発環境をクラスター化では、高可用性や連携の強化という利点があり、ランタイム環境のクラスター化では、高可用性と負荷分散の利点があります。高可用性により、1台のサーバーで障害が発生したときにデータが損失する可能性を減らすことができます。サーバーでの障害発生時に、別のサーバーにより、障害が発生したサーバーにあるデータのコピーを提供して補足します。障害が発生したサーバーが再度オンラインになったら、クラスターに戻ります。



### 注記

ランタイム環境のクラスター化は、Red Hat JBoss EAP 7.3 および Red Hat OpenShift Container Platform でのみサポートされています。

## 第6章 RED HAT DECISION MANAGER ロールおよびユーザー

Business Central または KIE Server にアクセスするには、サーバーを起動する前にユーザーを作成して適切なロールを割り当てます。Business Central または KIE Server のインストール時に、ユーザーとロールを作成できます。

Business Central と KIE Server は、JAVA 認証承認サービス (JAAS) ログインモジュールを使用してユーザーを認証します。Business Central と KIE Server の両方が単一のインスタンスで実行されている場合は、同じ JAAS サブジェクトとセキュリティドメインを共有します。したがって、Business Central に対して認証されたユーザーは、KIE Server にもアクセスできます。

ただし、Business Central と KIE Server が異なるインスタンスで実行されている場合、JAAS ログインモジュールは両方に対して個別にトリガーされます。したがって、Business Central に対して認証されたユーザーは、KIE Server にアクセスするために別々に認証する必要があります。たとえば、Business Central で認証されているものの、KIE Server で認証されていないユーザーが Business Central でプロセステキストを表示または管理しようとする、401 エラーがログファイルに記録され、**Invalid credentials to load data from remote server.Contact your system administrator.** メッセージが Business Central に表示されます。

本セクションでは、Red Hat Decision Manager のユーザーロールを説明します。



### 注記

**admin**、**analyst**、および **rest-all** のロールは Business Central 用に予約されています。**kie-server** ロールは KIE Server 用に予約されています。このため、Business Central または KIE Server のいずれか、またはそれら両方がインストールされているかどうかによって、利用可能なロールは異なります。

- **admin:** **admin** ロールを持つユーザーは Business Central 管理者です。管理者は、ユーザーの管理や、リポジトリの作成、クローン作成、および管理ができます。アプリケーションで必要な変更をすべて利用できます。**admin** ロールを持つユーザーは、Red Hat Decision Manager の全領域にアクセスできます。
- **analyst:** **analyst** ロールを持つユーザーには、すべてのハイレベル機能へのアクセスがあります。プロジェクトのモデル化が可能です。ただし、このユーザーは、**Design → Projects** ビューでスペースに貢献者を追加したり、スペースを削除したりできません。**analyst** ロールを持つユーザーは、管理者向けの **Deploy → Execution Servers** ビューにアクセスできません。ただし、これらのユーザーは、ライブラリーパースペクティブにアクセスするときに **Deploy** ボタンを使用できます。
- **rest-all:** **rest-all** ロールを持つユーザーは、Business Central REST 機能にアクセスできます。
- **kie-server:** **kie-server** ロールを持つユーザーは、KIE Server REST 機能にアクセスできます。

## 第7章 RED HAT DECISION MANAGER リファレンス実装

Red Hat Decision Manager は、スターターアプリケーションとして使用できる参照実装を提供します。**Red Hat Decision Manager 7.10.0 Reference Implementations**は、Red Hat カスタマーポータル の Software Downloads ページにある Red Hat Decision Manager からダウンロードできます。

### 従業員勤務表の参照実装

従業員勤務表の参照実装により、組織内のさまざまなポジションのシフトに従業員を割り当てるアプリケーションの作成が可能となります。たとえば、アプリケーションを使用して、病院での看護師のシフト、さまざまな場所での警備勤務シフト、作業者の組み立てラインのシフトを割り当てます。

### 運搬経路プランニングの参照実装

運搬経路プランニングの参照実装により、実際の地図、道路、および目的地へ向かう物資輸送車(車両ごとに積載量を指定)に関する運搬経路プランニングの問題を解決するアプリケーションを作成できます。詳細は、参照実装のダウンロードに含まれる運搬経路 ZIP ファイルの README ファイルを参照してください。

### 授業の時間割参照実装

授業の時間割参照実装により、授業を教室や時間枠と関連付けて、学生および教師の制約を考慮して競合を回避する REST アプリケーションを Spring Boot に構築できます。

### 高可用性イベント駆動型デシジョン機能の参照実装

高可用性イベント駆動型デシジョン機能の参照実装により、複雑なイベント処理で開発したルールを含む、ステートフル処理を必要とする Drools エンジンコードを OpenShift 環境にデプロイできます。これにより、デシジョンエンジンは、高可用性の複雑なイベントシリーズを処理できるようになります。

## 第8章 RED HAT DECISION MANAGER との統合

Red Hat Decision Manager は、Spring Boot、Red Hat Fuse、Red Hat Single Sign-On などの他の製品やコンポーネントと統合できます。

### 8.1. RED HAT DECISION MANAGER SPRING BOOT ビジネスアプリケーション

Spring Framework は、Java アプリケーション開発用に包括的なインフラストラクチャサポートを提供する Java プラットフォームです。Spring Boot は、Spring Boot スターターをベースにした、軽量フレームワークです。Spring Boot スターターは、**pom.xml** ファイルで、このファイルには Spring Boot プロジェクトに追加可能な依存関係の記述子セットが含まれます。Red Hat は、Red Hat Runtime の一部として、Red Hat Decision Manager Spring Boot プロジェクトのサポートおよびメンテナンスを提供します。

Red Hat Decision Manager Spring Boot ビジネスアプリケーションは、特定のビジネス機能を提供する個別サービスを柔軟に、UI に依存せず、論理的にグループ化します。ビジネスアプリケーションは、Spring Boot スターターをベースにしています。通常、個別にデプロイされ、個別にバージョン管理できます。完全なビジネスアプリケーションを使用すると、ドメインは特定のビジネスゴール (たとえば、順序管理や補助管理など) を達成できます。

[business application](#) の Web サイトで、Process Automation Manager、Decision Manager、または Business Optimizer ビジネスアプリケーションを作成できます。ビジネスアプリケーションを作成して設定した後は、OpenShift で、既存のサービスやクラウドにデプロイできます。

### 8.2. RED HAT FUSE の統合

Fuse は、アジャイル統合ソリューションの一部である、分散型のクラウドネイティブ統合プラットフォームです。チームはこの分散アプローチを使用することで、必要に応じて統合サービスをデプロイできます。

Apache Karaf コンテナプラットフォームまたは Red Hat JBoss Enterprise Application Platform に Red Hat Fuse をインストールしてから、このコンテナに Red Hat Decision Manager をインストールし、設定してください。

### 8.3. RED HAT SINGLE SIGN-ON の統合

Red Hat シングルサインオン (RH-SSO) は、ブラウザーアプリケーションと REST Web サービス、および Git へのアクセスのセキュリティを確保するために使用できるシングルサインオンソリューションです。

Red Hat Decision Manager と RH-SSO を統合する際に、Red Hat Decision Manager 向けに SSO と IDM (アイデンティティ管理) を作成します。RH-SSO のセッション管理機能により、一度認証するだけで、Web 上でさまざまな Red Hat Decision Manager 環境を使用できます。

RH-SSO 7.3 は Red Hat JBoss EAP 7.3 上の Red Hat Decision Manager でサポートされています。

## 第9章 関連資料

### インストール

- [Red Hat JBoss EAP 7.3 への Red Hat Decision Manager のインストールおよび設定](#)
- [Red Hat JBoss EAP クラスター環境への Red Hat Decision Manager のインストールおよび設定](#)
- [Red Hat JBoss Web Server への Red Hat Decision Manager のインストールおよび設定](#)
- [IBM WebSphere Application Server への KIE Server のインストールおよび設定](#)
- [Oracle WebLogic Server への KIE Server のインストールおよび設定](#)

### 統合

- [Spring Boot での Red Hat Decision Manager ビジネスアプリケーションの作成](#)
- [Red Hat Decision Manager と Red Hat Fuse の統合](#)
- [Red Hat Decision Manager と Red Hat Single Sign-On の統合](#)

### 参照実装

- [IDE を使用した Red Hat Business Optimizer 向け従業員勤務表スターターアプリケーションの実行および変更](#)
- [Red Hat Business Optimizer 向けの配送経路プランニングスターターアプリケーションのデプロイおよび使用](#)
- [Red Hat Business Optimizer Spring Boot を使用した授業の時間割の作成](#)
- [デシジョンエンジンを使用した高可用性イベント駆動型デシジョン機能の Red Hat OpenShift Container Platform への実装](#)

### OpenShift

- [『Operator を使用した Red Hat OpenShift Container Platform 4 への Red Hat Decision Manager 環境のデプロイメント』](#)
- [テンプレートを使用した Red Hat OpenShift Container Platform 3 への Red Hat Decision Manager 環境のデプロイメント](#)
- [デシジョンエンジンを使用した高可用性イベント駆動型デシジョン機能の Red Hat OpenShift Container Platform への実装](#)

## パート II. RED HAT JBOSS EAP 7.3 への RED HAT DECISION MANAGER のインストールおよび設定

本書は、Red Hat JBoss Enterprise Application Platform 7.3 インスタンスに Red Hat Decision Manager をインストールする方法を説明します。

### 前提条件

- [Red Hat Decision Manager インストールの計画](#) の内容を確認している。
- Red Hat JBoss Enterprise Application Platform 7.3 の最新の修正リリースがインストールされている。Red Hat JBoss EAP のインストールに関する情報は [Red Hat JBoss Enterprise Application Platform インストールガイド](#) を参照してください。



## 第10章 RED HAT DECISION MANAGER について

Red Hat Decision Manager は、ビジネスルール管理、複合イベント処理、Decision Model & Notation (DMN) 実行、およびプランニングの問題を解決するための Business Optimizer を組み合わせた、オープンソースの意思決定管理プラットフォームです。これにより、ビジネス上の意思決定を自動化し、そのロジックをビジネス全体で利用できるようにします。

ルール、デシジョンテーブル、DMN モデルなどのビジネスアセットはプロジェクト内で組織化され、Business Central リポジトリに保存されます。これにより、ビジネス全体で一貫性や透明性を維持し、監査を行えます。ビジネスユーザーは、IT 担当者からのサポートなしでビジネスロジックを編集できます。

Red Hat Process Automation Manager は、Business Central および KIE Server で設定されます。

- KIE Server では、ルール、およびその他のアーティファクトが実行されます。これは、ルールをインスタンス化して実行し、計画の問題を解決するために使用されます。KIE Server はビジネスアセットのランタイム環境を提供し、アセットリポジトリ (ナレッジストア) に保存されたデータにアクセスします。
- Business Central は、KIE Server が実行するビジネスルールを作成して管理するグラフィカルユーザーインターフェイスです。

Business Central は、スタンドアロンの JAR ファイルとしても使用できます。Business Central スタンドアロンの JAR ファイルとして使用して、アプリケーションサーバーにデプロイせずに Business Central を実行できます。

Red Hat JBoss Enterprise Application Platform (Red Hat JBoss EAP) 7.3 は、Java Enterprise Edition 8 (Java EE 8) の Full Profile 仕様および Web Profile 仕様の認定実装です。Red Hat JBoss EAP には、高可用性クラスタリング、メッセージング、分散キャッシングなどの機能に対する事前設定オプションが用意されています。ユーザーは、Red Hat JBoss EAP が提供するさまざまな API およびサービスを使用して、アプリケーションを開発、デプロイ、および実行することもできます。

本書では、Red Hat JBoss EAP 7.3 サーバーインスタンスに Red Hat Decision Manager をインストールする方法を説明します。

他の環境への Red Hat Decision Manager のインストール方法に関する説明は、以下のドキュメントを参照してください。

- [IBM WebSphere Application Server への KIE Server のインストールおよび設定](#)
- [Oracle WebLogic Server への KIE Server のインストールおよび設定](#)
- [『Operator を使用した Red Hat OpenShift Container Platform 4 への Red Hat Decision Manager 環境のデプロイメント』](#)
- [テンプレートを使用した Red Hat OpenShift Container Platform 3 への Red Hat Decision Manager 環境のデプロイメント](#)

サポートされるコンポーネントについては、以下のドキュメントを参照してください。

- [What is the mapping between Red Hat Decision Manager and the Maven library version?](#)
- [Red Hat Decision Manager 7 でサポート対象の設定](#)

## 第11章 RED HAT DECISION MANAGER ロールおよびユーザー

Business Central または KIE Server にアクセスするには、サーバーを起動する前にユーザーを作成して適切なロールを割り当てます。Business Central または KIE Server のインストール時に、ユーザーとロールを作成できます。

Business Central と KIE Server は、JAVA 認証承認サービス (JAAS) ログインモジュールを使用してユーザーを認証します。Business Central と KIE Server の両方が単一のインスタンスで実行されている場合は、同じ JAAS サブジェクトとセキュリティドメインを共有します。したがって、Business Central に対して認証されたユーザーは、KIE Server にもアクセスできます。

ただし、Business Central と KIE Server が異なるインスタンスで実行されている場合、JAAS ログインモジュールは両方に対して個別にトリガーされます。したがって、Business Central に対して認証されたユーザーは、KIE Server にアクセスするために別々に認証する必要があります。たとえば、Business Central で認証されているものの、KIE Server で認証されていないユーザーが Business Central でプロセスマニフェストを表示または管理しようとする、401 エラーがログファイルに記録され、**Invalid credentials to load data from remote server. Contact your system administrator.** メッセージが Business Central に表示されます。

本セクションでは、Red Hat Decision Manager のユーザーロールを説明します。



### 注記

**admin**、**analyst**、および **rest-all** のロールは Business Central 用に予約されています。**kie-server** ロールは KIE Server 用に予約されています。このため、Business Central または KIE Server のいずれか、またはそれら両方がインストールされているかどうかによって、利用可能なロールは異なります。

- **admin:** **admin** ロールを持つユーザーは Business Central 管理者です。管理者は、ユーザーの管理や、リポジトリの作成、クローン作成、および管理ができます。アプリケーションで必要な変更をすべて利用できます。**admin** ロールを持つユーザーは、Red Hat Decision Manager の全領域にアクセスできます。
- **analyst:** **analyst** ロールを持つユーザーには、すべてのハイレベル機能へのアクセスがあります。プロジェクトのモデル化が可能です。ただし、このユーザーは、**Design → Projects** ビューでスペースに貢献者を追加したり、スペースを削除したりできません。**analyst** ロールを持つユーザーは、管理者向けの **Deploy → Execution Servers** ビューにアクセスできません。ただし、これらのユーザーは、ライブラリーパースペクティブにアクセスするときに **Deploy** ボタンを使用できます。
- **rest-all:** **rest-all** ロールを持つユーザーは、Business Central REST 機能にアクセスできます。
- **kie-server:** **kie-server** ロールを持つユーザーは、KIE Server REST 機能にアクセスできます。

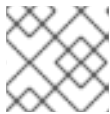
## 第12章 RED HAT DECISION MANAGER インストールファイルのダウンロード

インストーラー JAR ファイルまたはデプロイ可能な ZIP ファイルを使用して Red Hat Decision Manager をインストールすることができます。インストーラーは、インタラクティブモードまたはコマンドラインインターフェイス (CLI) モードで実行できます。もしくは、Business Central および KIE Server のデプロイ可能な ZIP ファイルを展開して設定できます。Business Central をアプリケーションサーバーにデプロイせずに実行する場合は、Business Central スタンドアロン JAR ファイルをダウンロードします。

お使いの環境およびインストール要件に応じた Red Hat Decision Manager ディストリビューションをダウンロードします。

### 手順

1. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product:** Decision Manager
  - **バージョン:** 7.10
2. お好みのインストール方法に従って、以下の製品ディストリビューションのいずれかをダウンロードします。



### 注記

ダウンロードするのは、どれか1つのディストリビューションのみです。

- インストーラーを使用して Red Hat JBoss EAP 7.3 に Red Hat Decision Manager をインストールする場合は、**Red Hat Decision Manager 7.10.0 Installer(rhdm-installer-7.10.0.jar)** をダウンロードします。インストーラーのグラフィックユーザーインターフェイスに従って、インストールプロセスを進めます。
- デプロイ可能な ZIP ファイルを使用して Red Hat JBoss EAP 7.3 に Red Hat Decision Manager をインストールするには、以下のファイルをダウンロードします。
  - Red Hat Decision Manager 7.10.0 KIE Server for All Supported EE8 Containers (rhdm-7.10.0-kie-server-ee8.zip)
  - Red Hat Decision Manager 7.10.0 KIE Server Deployable for EAP (rhdm-7.10.0-decision-central-eap7-deployable.zip)
- アプリケーションサーバーにデプロイせずに Business Central を実行するには、**Red Hat Decision Manager 7.10.0 Business Central Standalone (rhdm-7.10.0-decision-central-standalone.jar)** をダウンロードします。

## 第13章 RED HAT DECISION MANAGER インストーラーの使用方法

本セクションでは、インストーラー JAR ファイルを使用して Business Central および KIE Server をインストールする方法を説明します。JAR ファイルは、既存の Red Hat JBoss EAP 7.3 サーバーインストール環境に Red Hat Decision Manager をインストールする実行ファイルです。インストーラーは、インタラクティブモードまたはコマンドラインインターフェイス (CLI) モードで実行できます。



### 注記

Red Hat Decision Manager JAR ファイルインストーラーは、yum または RPM パッケージマネージャーでインストールした Red Hat JBoss EAP ディストリビューションを **サポートしません**。このような Red Hat JBoss EAP システムに Red Hat Decision Manager をインストールする場合は、Red Hat **Decision Manager 7.10 Deployable for Red Hat JBoss EAP 7.3** ファイルをダウンロードして、「[14章ZIP からの Red Hat Decision Manager のインストール](#)」に記載の手順に従います。



### 注記

IBM JDK は他の JDK で生成されるキーストアを使用できないため、別の JDK で生成されたキーストアを持つ IBM JDK で実行されている既存の Red Hat JBoss EAP システムに Red Hat Decision Manager をインストールすることはできません。

### 次のステップ:

以下のセクションのいずれかに記載される手順を行います。

- [「インタラクティブモードでのインストーラーの使用」](#)
- [「CLI モードでのインストーラーの使用」](#)

### 13.1. インタラクティブモードでのインストーラーの使用

Red Hat Decision Manager のインストーラーは、実行可能な JAR ファイルです。このインストーラーを使用して、既存の Red Hat JBoss EAP 7.3 サーバーインストールに Red Hat Decision Manager をインストールできます。



### 注記

セキュリティ上の理由で、root 以外のユーザーでインストーラーを実行する必要があります。

### 前提条件

- Red Hat Decision Manager 7.10.0 インストーラーをダウンロードしている。手順は、[12章Red Hat Decision Manager インストールファイルのダウンロード](#)を参照してください。
- サポート対象の JDK がインストールされている。サポート対象の JDK に関する一覧は、[Red Hat Decision Manager 7 Supported Configurations](#) を参照してください。
- バックアップ済みの Red Hat JBoss EAP 7.3 のサーバーインストールが利用できる。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。

- `$PATH` 環境変数に含まれている JAR バイナリー。Red Hat Enterprise Linux では、`java-$JAVA_VERSION-openjdk-devel` パッケージに含まれています。



### 注記

Red Hat Decision Manager は、UTF-8 エンコーディングで機能するように設計されています。基礎となる JVM で別のエンコーディングシステムを使用すると、予期せぬエラーが発生する可能性があります。JVM で UTF-8 を使用するするには、`-Dfile.encoding=UTF-8` のシステムプロパティを使用します。システムプロパティの一覧は、[付録A Business Central システムプロパティ](#)を参照してください。

### 手順

1. 端末ウィンドウで、インストーラー JAR ファイルをダウンロードしたディレクトリに移動し、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.10.0.jar
```



### 注記

Windows でインストーラーを実行すると、インストール時に管理者の認証情報が求められる場合があります。この要求を回避するには、インストールコマンドに `izpack.mode=privileged` オプションを追加します。

```
java -Dizpack.mode=privileged -jar  
rhdm-installer-7.10.0.jar
```

また、32 ビットの Java 仮想マシンでインストーラーを実行している場合には、メモリ不足になる可能性があります。この問題を防ぐには、以下のコマンドを実行します。

```
java -XX:MaxHeapSize=4g -jar  
rhdm-installer-7.10.0.jar
```

グラフィカルインストーラーにスプラッシュ画面と使用許諾契約書のページが表示されます。

2. **I accept the terms of this license agreement (本使用許諾契約書の内容に同意します)** をクリックし、**Next** をクリックします。
3. Red Hat Decision Manager をインストールする Red Hat JBoss EAP 7.3 サーバーのホームを指定して、**Next** をクリックします。
4. インストールするコンポーネントを選択し、**Next** をクリックします。



### 注記

Business Central と KIE Server は同じサーバーにインストールできます。ただし、実稼働環境では、Business Central と KIE Server は異なるサーバーにインストールすることが推奨されます。そのためには、インストーラーを 2 回実行します。

5. ユーザーを作成して **Next** をクリックします。デフォルトでは、同じコンテナに Business

Central と KIE Server の両方をインストールする場合は、新しいユーザーに **admin** ロール、**kie-server** ロール、および **rest-all** ロールが割り当てられます。KIE Server のみをインストールした場合には、ユーザーには **kie-server** ロールが割り当てられます。別のロールを選択する場合は、**admin** ロールを外します。ロールの詳細は、[11章 Red Hat Decision Manager ロールおよびユーザー](#) を参照してください。



### 注記

必ず、既存のユーザー、ロール、またはグループとは異なるユーザー名を指定してください。たとえば、**admin** という名前のユーザーは作成しないでください。

パスワードは 8 文字以上で、数字と、英数字以外の文字をそれぞれ 1 文字以上使用する必要があります。ただし & の文字は使用できません。

ユーザー名とパスワードを書き留めておきます。Business Central および KIE Server にアクセスする時に必要になります。

6. **Installation Overview** ページで、インストールするコンポーネントを確認し、**Next** をクリックしてインストールを開始します。
7. インストールが完了したら、**Next** をクリックします。
8. 画面上部に **Processing finished** が表示されたら、**Next** をクリックしてインストールを完了します。
9. 必要に応じて、**Generate Installation Script and Properties File** をクリックして、XML ファイルでインストールデータを保存し、**Done** をクリックします。  
インストーラーは、2つのファイルを生成します。**auto.xml** ファイルは、今後のインストールを自動化し、**auto.xml.variables** ファイルは、ユーザーのパスワードと他の機密情報に関連する変数を保存します。**auto.xml** ファイルを使用して、元のインストールと同じタイプのサーバーおよび同じ設定の複数のシステムで Red Hat Decision Manager のインストールを繰り返します。必要に応じて、**auto.xml** ファイルの **installpath** パラメーターを更新します。XML ファイルを使用してインストールを実行するには、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.10.0.jar <path-to-auto.xml-file>
```

インストーラーを使用して Red Hat Decision Manager が正常にインストールされました。

## 13.2. CLI モードでのインストーラーの使用

コマンドラインインターフェイス (CLI) を使用して Red Hat Decision Manager インストーラーを実行できます。



### 注記

セキュリティ上の理由で、root 以外のユーザーでインストーラーを実行する必要があります。

### 前提条件

- Red Hat Decision Manager 7.10.0 インストーラーをダウンロードしている。手順は、[12章 Red Hat Decision Manager インストールファイルのダウンロード](#) を参照してください。



- サポート対象の JDK がインストールされている。サポート対象の JDK に関する一覧は、[Red Hat Decision Manager 7 Supported Configurations](#) を参照してください。
- バックアップ済みの Red Hat JBoss EAP 7.3 のサーバーインストールが利用できる。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。
- `$PATH` 環境変数に含まれている JAR バイナリー。Red Hat Enterprise Linux では、`java-$JAVA_VERSION-openjdk-devel` パッケージに含まれています。



### 注記

Red Hat Decision Manager は、UTF-8 エンコーディングで機能するように設計されています。基礎となる JVM で別のエンコーディングシステムを使用すると、予期せぬエラーが発生する可能性があります。JVM で UTF-8 を使用するようには、`-Dfile.encoding=UTF-8` のシステムプロパティを使用します。システムプロパティの一覧は、[付録A Business Central システムプロパティ](#) を参照してください。

### 手順

1. 端末ウィンドウにおいて、インストーラーファイルをダウンロードしたディレクトリーに移動し、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.10.0.jar -console
```

コマンドラインの対話プロセスが開始し、使用許諾契約書が表示されます。

```
press 1 to continue, 2 to quit, 3 to redisplay.
```

2. 使用許諾契約書を読んで **1** を入力し、**Enter** キーを押して続行します。

Specify the home directory of one of the following servers: Red Hat JBoss EAP 7 or Red Hat JBoss Web Server 5. For more information, see <https://access.redhat.com/articles/3354301>[Red Hat Decision Manager 7 Supported Configurations].

3. 既存の Red Hat JBoss EAP 7.3 インストールの親ディレクトリーを入力します。インストーラーが、指定したインストール場所を確認します。**1** を入力して確認し、続行します。



### 注記

Business Central と KIE Server は同じサーバーにインストールできます。ただし、実稼働環境では、Business Central と KIE Server は異なるサーバーにインストールすることが推奨されます。

4. インストーラーの指示に従って、インストールを完了します。



## 注記

ユーザー名およびパスワードを作成する場合は、指定したユーザー名が既存のロールまたはグループの名前と競合しないようにしてください。たとえば、**admin** というロールがある場合は、**admin** という名前のユーザーは作成しないでください。

パスワードは 8 文字以上で、数字と、英数字以外の文字をそれぞれ 1 文字以上使用する必要があります。ただし **&** の文字を使用することは **できません**。

ユーザー名とパスワードを書き留めておきます。Business Central および KIE Server にアクセスする時に必要になります。

5. インストールが完了すると、以下のメッセージが表示されます。

Would you like to generate an automatic installation script and properties file?

6. **y** を入力してインストールデータが含まれる XML ファイルを作成します。あるいは、**n** を入力してインストールを完了します。**y** を入力すると、XML ファイルのパスを指定するように求められます。
7. パスを入力するか、Enter キーを押して提案されたパスを了承します。  
インストーラーは、2 つのファイルを生成します。**auto.xml** ファイルは、今後のインストールを自動化し、**auto.xml.variables** ファイルは、ユーザーのパスワードと他の機密情報に関連する変数を保存します。複数のシステムで **auto.xml** ファイルを使用して、元のインストールと同じ設定の同じタイプのサーバーに、Red Hat Process Automation Manager を簡単に繰り返しインストールできます。必要に応じて、**auto.xml** ファイルの **installpath** パラメーターを更新します。XML ファイルを使用してインストールを実行するには、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.10.0.jar <path-to-auto.xml-file>
```

8. Business Central だけをインストールした場合は、この手順を繰り返して、別のサーバーに KIE Server をインストールします。



## 第14章 ZIP からの RED HAT DECISION MANAGER のインストール

Red Hat Decision Manager の ZIP ファイル (Business Central 用および KIE Server 用) を使用して、インストーラーを使用せずに Red Hat Decision Manager をインストールできます。



### 注記

Business Central と KIE Server は、実稼働環境では異なるサーバーにインストールすることが推奨されます。

ヘッドレス Decision Manager コントローラーのインストールは、[19章ヘッドレス Decision Manager コントローラーのインストールおよび実行](#)を参照してください。

### 14.1. ZIP ファイルからの BUSINESS CENTRAL のインストール

Business Central は、KIE Server が実行するビジネスルールを作成して管理するグラフィカルユーザーインターフェイスです。デプロイ可能な ZIP ファイルを使用して Business Central をインストールおよび設定できます。

#### 前提条件

- バックアップを作成済みの Red Hat JBoss EAP システム (バージョン 7.3) が利用できる。Red Hat JBoss EAP システムのベースディレクトリーを **EAP\_HOME** とします。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。
- [12章Red Hat Decision Manager インストールファイルのダウンロード](#) に従って以下のファイルがダウンロードされている。  
**rhdm-7.10.0-decision-central-eap7-deployable.zip**

#### 手順

1. **rhdm-7.10.0-decision-central-eap7-deployable.zip** ファイルを一時ディレクトリーに展開します。以下の例では、この名前を **TEMP\_DIR** とします。
2. **TEMP\_DIR/rhdm-7.10.0-decision-central-eap7-deployable/jboss-eap-7.3** ディレクトリーのコンテンツを **EAP\_HOME** にコピーします。プロンプトが表示されたら、ファイルをマージまたは置き換えます。



### 警告

コピーする Red Hat Decision Manager デプロイメントの名前が Red Hat JBoss EAP インスタンスの既存デプロイメントと競合しないことを確認します。

### 14.2. ZIP ファイルからの KIE SERVER のインストール

KIE Server はビジネスアセットのランタイム環境を提供し、アセットリポジトリ (ナレッジストア) に保存されたデータにアクセスします。デプロイ可能な ZIP ファイルを使用して KIE Server をインストールおよび設定します。

### 前提条件

- バックアップを作成済みの Red Hat JBoss EAP システム (バージョン 7.3) が利用できる。Red Hat JBoss EAP システムのベースディレクトリーを **EAP\_HOME** とします。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。
- [12章Red Hat Decision Manager インストールファイルのダウンロード](#) に従って以下のファイルがダウンロードされている。  
**rhdm-7.10.0-kie-server-ee8.zip**

### 手順

1. **rhdm-7.10.0-kie-server-ee8.zip** アーカイブを一時ディレクトリーに展開します。以下の例では、この名前を **TEMP\_DIR** とします。
2. **TEMP\_DIR/rhdm-7.10.0-kie-server-ee8/kie-server.war** ディレクトリーを **EAP\_HOME/standalone/deployments/** にコピーします。



#### 警告

コピーする Red Hat Decision Manager デプロイメントの名前が Red Hat JBoss EAP インスタンスの既存デプロイメントと競合しないことを確認します。

3. **TEMP\_DIR/rhdm-7.10.0-kie-server-ee8/rhdm-7.10.0-kie-server-ee8/SecurityPolicy/** のコンテンツを **EAP\_HOME/bin** にコピーします。
4. ファイルを上書きするように求められたら、**Replace** をクリックします。
5. **EAP\_HOME/standalone/deployments/** ディレクトリーに、**kie-server.war.dodeploy** という名前で空のファイルを作成します。このファイルにより、サーバーが起動すると KIE Server が自動的にデプロイされます。

## 14.3. ユーザーの作成

デプロイ可能な ZIP ファイルを使用して Red Hat Decision Manager をインストールした場合は、Business Central または KIE Server にログインする前にユーザーを作成する必要があります。本セクションでは、**admin**、**rest-all**、および **kie-server** のロールを持つ Business Central ユーザーと、**kie-server** ロールを持つ KIE Server ユーザーを作成する方法を説明します。ロールの詳細は、[11章Red Hat Decision Manager ロールおよびユーザー](#) を参照してください。



### 注記

インストーラーを使用して Red Hat Decision Manager をインストールした場合は、そのプロセス時に少なくとも1人のユーザーを作成している。**EAP\_HOME/standalone/configuration/application-users.properties** ファイルには、アプリケーションユーザーおよびパスワードが含まれます。

### 前提条件

- Red Hat Decision Manager が Red Hat JBoss EAP インストールのベースディレクトリー (**EAP\_HOME**) にインストールされている。

### 手順

1. 端末アプリケーションで **EAP\_HOME/bin** ディレクトリーに移動します。
2. ユーザーを作成して、**admin** ロール、**rest-all** ロール、および **kie-server** ロールを割り当てます。



### 注記

**admin** ロールを持つユーザーは、Business Central の管理者です。**rest-all** ロールを持つユーザーは、Business Central REST 機能にアクセスできます。**kie-server** ロールを持つユーザーは、KIE Server REST 機能にアクセスできます。

以下のコマンドの **<USERNAME>** および **<password>** を、作成するユーザーとそのパスワードに置き換えます。

```
$ ./add-user.sh -a --user <USERNAME> --password <PASSWORD> --role admin,rest-all,kie-server
```



### 注記

必ず、既存のユーザー、ロール、またはグループとは異なるユーザー名を指定してください。たとえば、**admin** という名前のユーザーは作成しないでください。

パスワードは8文字以上で、数字と、英数字以外の文字をそれぞれ1文字以上使用する必要があります。ただし & の文字は使用できません。

3. KIE Server へのログインに使用する **kie-server** ロールのユーザーを作成します。

```
$ ./add-user.sh -a --user <USERNAME> --password <PASSWORD> --role kie-server
```

4. ユーザー名とパスワードを書き留めておきます。



### 注記

Business Central と KIE Server を同じサーバーにインストールした場合は、両方のロールを持つユーザーを1つ作成できます。

```
$ ./add-user.sh -a --user <USERNAME> --password <PASSWORD> --role  
admin,rest-all,kie-server
```

実稼働環境では、Business Central と KIE Server は異なるサーバーにインストールすることが推奨されます。

## 14.4. BUSINESS CENTRAL に接続する KIE SERVER の設定



### 警告

このセクションでは、テスト目的で使用可能なサンプルの設定を紹介します。一部の値は、実稼働環境には適しておらず、その旨を記載しています。

Red Hat Decision Manager 環境で KIE Server が設定されていない場合や、Red Hat Decision Manager 環境に追加の KIE Server が必要な場合は、KIE Server が Business Central に接続するように設定する必要があります。



### 注記

Red Hat OpenShift Container Platform に KIE Server をデプロイする場合は、『[Operator を使用した Red Hat OpenShift Container Platform 4 への Red Hat Decision Manager 環境 のデプロイ](#)』で、Business Central に接続する設定手順を参照してください。

### 前提条件

- 以下のセクションのとおり、Business Central および KIE Server が Red Hat JBoss EAP インストールのベースディレクトリー (**EAP\_HOME**) にインストールされている。
  - 「[ZIP ファイルからの Business Central のインストール](#)」
  - 「[ZIP ファイルからの KIE Server のインストール](#)」
- 以下のロールを持つユーザーが存在している
  - Business Central: **rest-all** ロールを持つユーザー
  - KIE Server: **kie-server** ロールを持つユーザー  
詳細は、「[ユーザーの作成](#)」を参照してください。

### 手順

- Red Hat Decision Manager インストールディレクトリーで、**standalone-full.xml** ファイルに移動します。たとえば、Red Hat Decision Manager に Red Hat JBoss EAP インストールを使用する場合は **\$EAP\_HOME/standalone/configuration/standalone-full.xml** に移動します。

2. **standalone-full.xml** ファイルを開き、**<system-properties>** タグの下に、以下の JVM プロパティーを設定します。

表14.1 管理対象 KIE Server インスタンスの JVM プロパティー

プロパティー	値	注記
<b>org.kie.server.id</b>	<b>default-kie-server</b>	KIE Server ID。
<b>org.kie.server.controller</b>	<b>http://localhost:8080/decision-central/rest/controller</b>	Business Central の場所 Business Central の API に接続する URL。
<b>org.kie.server.controller.user</b>	<b>controllerUser</b>	Business Central にログイン可能な <b>rest-all</b> ロールを持つユーザー名。
<b>org.kie.server.controller.pwd</b>	<b>controllerUser1234;</b>	Business Central にログインできるユーザーのパスワード。
<b>org.kie.server.location</b>	<b>http://localhost:8080/kie-server/services/rest/server</b>	KIE Server の場所 KIE Server の API に接続する URL。

表14.2 Business Central インスタンスの JVM プロパティー

プロパティー	値	注記
<b>org.kie.server.user</b>	<b>controllerUser</b>	<b>kie-server</b> ロールを持つユーザー名。
<b>org.kie.server.pwd</b>	<b>controllerUser1234;</b>	ユーザーのパスワード。

以下の例は、KIE Server インスタンスを設定する方法を示しています。

```
<property name="org.kie.server.id" value="default-kie-server"/>
<property name="org.kie.server.controller" value="http://localhost:8080/decision-central/rest/controller"/>
<property name="org.kie.server.controller.user" value="controllerUser"/>
<property name="org.kie.server.controller.pwd" value="controllerUser1234;"/>
<property name="org.kie.server.location" value="http://localhost:8080/kie-server/services/rest/server"/>
```

以下の例は、Business Central インスタンスに設定する方法を示しています。

```
<property name="org.kie.server.user" value="controllerUser"/>
<property name="org.kie.server.pwd" value="controllerUser1234;"/>
```

3. KIE サーバーが正常に起動したことを確認するには、KIE サーバーが動作しているときに、**http://SERVER:PORT/kie-server/services/rest/server/** に GET リクエストを送信します。KIE Server で Red Hat Decision Manager を実行する方法は、『[Running Red Hat Decision](#)

[Manager](#)』を参照してください。

認証に成功すると、以下の例のような XML 応答が返されます。

```
<response type="SUCCESS" msg="Kie Server info">
  <kie-server-info>
    <capabilities>KieServer</capabilities>
    <capabilities>BRM</capabilities>
    <capabilities>BPM</capabilities>
    <capabilities>CaseMgmt</capabilities>
    <capabilities>BPM-UI</capabilities>
    <capabilities>BRP</capabilities>
    <capabilities>DMN</capabilities>
    <capabilities>Swagger</capabilities>
    <location>http://localhost:8230/kie-server/services/rest/server</location>
    <messages>
      <content>Server KieServerInfo{serverId='first-kie-server', version='7.5.1.Final-redhat-1', location='http://localhost:8230/kie-server/services/rest/server', capabilities=[KieServer, BRM, BPM, CaseMgmt, BPM-UI, BRP, DMN, Swagger]}started successfully at Mon Feb 05 15:44:35 AEST 2018</content>
      <severity>INFO</severity>
      <timestamp>2018-02-05T15:44:35.355+10:00</timestamp>
    </messages>
    <name>first-kie-server</name>
    <id>first-kie-server</id>
    <version>7.5.1.Final-redhat-1</version>
  </kie-server-info>
</response>
```

4. 登録が正常に完了したことを確認します。

- a. Business Central にログインします。
- b. **Menu → Deploy → Execution Servers** の順にクリックします。  
正常に登録されている場合は、登録されたサーバーの ID が表示されます。

## 第15章 キーストアを使用したパスワードセキュリティの確保

キーストアを使用して、Business Central と KIE Server の間の通信に使用するパスワードを暗号化できます。コントローラーと KIE Server のパスワードを暗号化する必要があります。Business Central と KIE Server を別のアプリケーションサーバーにデプロイする場合は、いずれのアプリケーションサーバーもキーストアを使用する必要があります。

Java Cryptography Extension KeyStore (JCEKS) は、対称鍵をサポートするため、キーストアには JCEKS を使用してください。JDK インストールに含まれる KeyTool を使用して、新しい JCEKS を作成します。



### 注記

KIE Server が JCEKS で設定されていない場合、KIE Server のパスワードはシステムプロパティにプレーンテキスト形式で保存されます。

### 前提条件

- Red Hat JBoss EAP に KIE Server がインストールされている。
- Java 8 以降がインストールされている。

### 手順

1. Red Hat JBoss EAP のホームディレクトリーで、以下のコマンドを入力して、**kie-server** ロールを割り当てた KIE Server ユーザーを作成し、パスワードを指定します。以下の例では、**<USER\_NAME>** と **<PASSWORD>** は任意のユーザー名とパスワードに置き換えます。

```
$<EAP_HOME>/bin/add-user.sh -a -e -u <USER_NAME> -p <PASSWORD> -g kie-server
```

2. KeyTool を使用して JCEKS を作成するには、Java 8 のホームディレクトリーで以下のコマンドを実行します。

```
$<JAVA_HOME>/bin/keytool -importpassword -keystore <KEYSTORE_PATH> -keypass  
<ALIAS_KEY_PASSWORD> -alias <PASSWORD_ALIAS> -storepass  
<KEYSTORE_PASSWORD> -storetype JCEKS
```

上記の例では、以下の変数を置き換えてください。

- **<KEYSTORE\_PATH>**: キーストアの保存先のパス
  - **<KEYSTORE\_PASSWORD>**: キーストアのパスワード
  - **<ALIAS\_KEY\_PASSWORD>**: エイリアスで保存した値にアクセスする時に使用するパスワード
  - **<PASSWORD\_ALIAS>**: プロセスへのエントリーに使用するエイリアス
3. プロンプトが表示されたら、作成した KIE Server ユーザーのパスワードを入力します。
  4. **EAP\_HOME/standalone/configuration/standalone-full.xml** ファイルに以下のシステムプロパティを設定して、以下の表に記載のプレースホルダーを置き換えます。

```
<system-properties>  
  <property name="kie.keystore.keyStoreURL" value="<KEYSTORE_URL>"/>
```

```

<property name="kie.keystore.keyStorePwd" value="<KEYSTORE_PWD>"/>
<property name="kie.keystore.key.server.alias" value="<KEY_SERVER_ALIAS>"/>
<property name="kie.keystore.key.server.pwd" value="<KEY_SERVER_PWD>"/>
<property name="kie.keystore.key.ctrl.alias" value="<KEY_CONTROL_ALIAS>"/>
<property name="kie.keystore.key.ctrl.pwd" value="<KEY_CONTROL_PWD>"/>
</system-properties>

```

表15.1 KIE Server JCEKS を読み込む時に使用するシステムプロパティー

システムプロパティー	プレースホルダー	説明
<b>kie.keystore.keyStoreURL</b>	<b>&lt;KEYSTORE_URL&gt;</b>	使用する JCEKS の URL (例: <b>file:///home/kie/keystores/keystore.jceks</b> )
<b>kie.keystore.keyStorePwd</b>	<b>&lt;KEYSTORE_PWD&gt;</b>	JCEKS のパスワード
<b>kie.keystore.key.server.alias</b>	<b>&lt;KEY_SERVER_ALIAS&gt;</b>	パスワードの保存先となる REST サービスのキーのエイリアス
<b>kie.keystore.key.server.pwd</b>	<b>&lt;KEY_SERVER_PWD&gt;</b>	保存したパスワードを使用する REST サービスのエイリアスのパスワード
<b>kie.keystore.key.ctrl.alias</b>	<b>&lt;KEY_CONTROL_ALIAS&gt;</b>	パスワードの保存先のデフォルトの REST Process Automation Controller のキーのエイリアス
<b>kie.keystore.key.ctrl.pwd</b>	<b>&lt;KEY_CONTROL_PWD&gt;</b>	保存したパスワードを使用する、デフォルトの REST Process Automation Controller のエイリアスのパスワード

5. KIE Server を起動して、設定を検証します。



## 第16章 SSH が RSA を使用するように設定する手順

Git リポジトリのクローンには、SSH を使用します。デフォルトで、Business Central には DSA 暗号化アルゴリズムが含まれます。ただし、Fedora 23 環境の SSH クライアントなどは、DSA アルゴリズムではなく、RSA アルゴリズムを使用します。Business Central にはシステムプロパティーが含まれており、必要に応じて DSA から RSA に切り替えることができます。



### 注記

Red Hat Enterprise Linux 7 などの、サポート対象の設定で SSH クライアントを使用している場合は、この問題の影響はありません。サポート対象の設定に関する一覧は、[Red Hat Decision Manager 7 Supported Configurations](#) を参照してください。

### 手順

以下のタスクの1つを実行して、このシステムプロパティーを有効にします。

- 以下のようにクライアント側の `~/.ssh/config` ファイルを変更して、SSH クライアントが強制的に非推奨の DSA アルゴリズムを受け入れるようにします。

```
Host <SERVER_IP>
    HostKeyAlgorithms +ssh-dss
```

- 以下のように、Business Central の起動時に、`-Dorg.uberfire.nio.git.ssh.algorithm=RSA` パラメーターを追加します。

```
$ ./standalone.sh -c standalone-full.xml
-Dorg.uberfire.nio.git.ssh.algorithm=RSA
```

## 第17章 RED HAT DECISION MANAGER システムの検証

Red Hat Decision Manager をインストールしたら、アセットを作成してシステムが機能していることを検証します。

### 手順

1. 端末ウィンドウで **EAP\_HOME/bin** ディレクトリーに移動し、以下のコマンドを実行して Red Hat Decision Manager を起動します。

```
./standalone.sh -c standalone-full.xml
```



### 注記

KIE Server を使用せず、Business Central だけをデプロイした場合は、**standalone.sh** スクリプトに **standalone-full.xml** ファイルを指定せず Business Central だけを起動できます。この場合は、Business Central を起動する前に **standalone.xml** ファイルの設定を変更しておく必要があります。

2. Web ブラウザーで **localhost:8080/decision-central** と入力します。
  - Red Hat Decision Manager をドメイン名から実行するように設定している場合は、以下のように **localhost** をドメイン名に置き換えます。  
**http://www.example.com:8080/decision-central**
  - Red Hat Decision Manager をクラスター環境で実行するように設定している場合は、以下のように **localhost** を特定ノードの IP アドレスに置き換えます。  
**http://<node\_IP\_address>:8080/decision-central**
3. インストール時に作成した **admin** ユーザーの認証情報を入力します。Business Central ホームページが表示されます。
4. **Menu** → **Deploy** → **Execution Servers** の順に選択します。
5. **default-kieserver** が **Server Configurations** 下に記載されていることを確認します。
6. **Menu** → **Design** → **Projects** の順に選択します。
7. **MySpace** スペースを開きます。
8. **Try Samples** → **Mortgages** → **OK** の順にクリックします。**Assets** ウィンドウが表示されます。
9. **Add Asset** → **Data Object** をクリックします。
10. **Data Object** フィールドに **MyDataObject** と入力し、**OK** をクリックします。
11. **Spaces** → **mySpace** → **Mortgages** の順にクリックし、アセットリストに **MyDataObject** があることを確認します。
12. **Mortgages** プロジェクトを選択します。

## 第18章 RED HAT DECISION MANAGER の実行

Red Hat JBoss EAP 上の Red Hat Decision Manager をスタンドアロンモードで実行するには、以下の手順にしたがいます。

### 前提条件

- Red Hat Decision Manager がインストールおよび設定済みである。



#### 注記

デフォルトのホスト (**localhost**) またはデフォルトのポート (**9990**) を変更した場合には、Red Hat Decision Manager を実行する前に **business-central.war/WEB-INF/classes/datasource-management.properties** ファイルおよび **business-central.war/WEB-INF/classes/security-management.properties** ファイルを [Solution 3519551](#) に記載の説明に沿って編集する必要があります。

### 手順

- ターミナルアプリケーションで **EAP\_HOME/bin** に移動します。
- スタンドアロン設定を実行します。

- Linux または UNIX ベースのシステムの場合:

```
$ ./standalone.sh -c standalone-full.xml
```

- Windows の場合:

```
standalone.bat -c standalone-full.xml
```



#### 注記

KIE Server を使用せず、Business Central だけをデプロイした場合は、**standalone.sh** スクリプトに **standalone-full.xml** ファイルを指定せず Business Central だけを起動できます。この場合は、Business Central を起動する前に **standalone.xml** ファイルの設定を変更しておく必要があります。

Linux または UNIX ベースのシステムの場合:

```
$ ./standalone.sh
```

Windows の場合:

```
standalone.bat
```

- Web ブラウザーで URL **localhost:8080/decision-central** にアクセスします。  
Red Hat Decision Manager をドメイン名から実行するように設定している場合は、以下のよう  
に **localhost** をドメイン名に置き換えます。

**http://www.example.com:8080/decision-central**

4. [「ユーザーの作成」](#) の Business Central で作成したユーザーの認証情報を使用してログインします。

## 第19章 ヘッドレス DECISION MANAGER コントローラーのインストールおよび実行

KIE Server は、管理モードまたは非管理モードで動作するように設定できます。KIE Server が非管理モードの場合は、手動で KIE コンテナ (デプロイメントユニット) を作成および維持する必要があります。管理モードの場合は、Decision Manager コントローラーが KIE Server の設定を管理し、ユーザーはコントローラーと対話形式で KIE コンテナを作成して維持します。

Business Central には Decision Manager コントローラーが組み込まれています。Business Central をインストールしている場合は、**Execution Server** ページを使用して KIE コンテナを作成および維持します。Business Central を使用せずに KIE Server の管理を自動化するには、ヘッドレス Decision Manager コントローラーを使用します。

### 19.1. ヘッドレス DECISION MANAGER コントローラーのインストール

ヘッドレス Decision Manager コントローラーをインストールし、REST API または KIE Server Java Client API を使用してコントローラーを操作することができます。

#### 前提条件

- バックアップを作成済みの Red Hat JBoss EAP システム (バージョン 7.3) が利用できる。Red Hat JBoss EAP システムのベースディレクトリーを **EAP\_HOME** とします。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。

#### 手順

1. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product:** Decision Manager
  - **バージョン:** 7.10
2. Red Hat Decision Manager 7.10.0 Add Ons(**rhdm-7.10.0-add-ons.zip** ファイル) をダウンロードします。
3. **rhdm-7.10.0-add-ons.zip** ファイルを展開します。**rhdm-7.10.0-controller-ee7.zip** ファイルは展開したディレクトリーにあります。
4. **rhdm-7.10.0-controller-ee7** アーカイブを一時ディレクトリーに展開します。以下の例では、この名前を **TEMP\_DIR** とします。
5. **TEMP\_DIR/rhdm-7.10.0-controller-ee7/controller.war** ディレクトリーを **EAP\_HOME/standalone/deployments/** にコピーします。

**警告**

コピーするヘッドレス Decision Manager コントローラーデプロイメントの名前が、Red Hat JBoss EAP インスタンスの既存デプロイメントと競合しないことを確認します。

6. **TEMP\_DIR/rhdm-7.10.0-controller-ee7/SecurityPolicy/** ディレクトリーの中身を **EAP\_HOME/bin** にコピーします。
7. ファイルの上書きを求めるプロンプトが出されたら、**Yes** を選択します。
8. **EAP\_HOME/standalone/deployments/** ディレクトリーに、**controller.war.dodeploy** という名前で空のファイルを作成します。このファイルにより、サーバーが起動するとヘッドレス Decision Manager コントローラーが自動的にデプロイされます。

### 19.1.1. ヘッドレス Decision Manager コントローラーの作成

ヘッドレス Decision Manager コントローラーを使用する前に、**kie-server** ロールを持つユーザーを作成する必要があります。

#### 前提条件

- ヘッドレス Decision Manager コントローラーが Red Hat JBoss EAP インストールのベースディレクトリー (**EAP\_HOME**) にインストールされている。

#### 手順

1. 端末アプリケーションで **EAP\_HOME/bin** ディレクトリーに移動します。
2. 以下のコマンドを入力し、**<USER\_NAME>** および **<PASSWORD>** を、作成するユーザー名およびパスワードに置き換えます。

```
$ ./add-user.sh -a --user <username> --password <password> --role kie-server
```

**注記**

必ず、既存のユーザー、ロール、またはグループとは異なるユーザー名を指定してください。たとえば、**admin** という名前のユーザーは作成しないでください。

パスワードは 8 文字以上で、数字と、英数字以外の文字をそれぞれ 1 文字以上使用する必要があります。ただし & の文字は使用できません。

3. ユーザー名とパスワードを書き留めておきます。

### 19.1.2. KIE Server とヘッドレス Decision Manager コントローラー の設定

KIE Server をヘッドレス Decision Manager コントローラーで管理する場合は、KIE Server インストールの **standalone-full.xml** とヘッドレス Decision Manager コントローラーの **standalone.xml** ファイルを編集する必要があります。

## 前提条件

- KIE Server が **EAP\_HOME** にインストールされている。
- ヘッドレス Decision Manager コントローラーが **EAP\_HOME** にインストールされている。



### 注記

実稼働環境では KIE Server およびヘッドレス Decision Manager コントローラーを異なるサーバーにインストールすることを推奨します。ただし、開発環境のように KIE Server およびヘッドレス Decision Manager コントローラーを同じサーバーにインストールする場合は、併せて共有の **standalone-full.xml** ファイルを変更します。

- KIE Server ノードに、**kie-server** ロールのあるユーザーが作成されている。
- サーバーノードに、**kie-server** ロールのあるユーザーが作成されている。  
詳細は、「[ユーザーの作成](#)」を参照。

## 手順

1. **EAP\_HOME/standalone/configuration/standalone-full.xml** ファイルの **<system-properties>** セクションに以下のプロパティを追加し、**<USERNAME>** および **<USER\_PWD>** を、**kie-server** ロールを持つユーザーの認証情報に置き換えます。

```
<property name="org.kie.server.user" value="<USERNAME>"/>
<property name="org.kie.server.pwd" value="<USER_PWD>"/>
```

2. KIE Server の **EAP\_HOME/standalone/configuration/standalone-full.xml** ファイルの **<system-properties>** セクションに以下のプロパティを追加します。

```
<property name="org.kie.server.controller.user" value="<CONTROLLER_USER>"/>
<property name="org.kie.server.controller.pwd" value="<CONTROLLER_PWD>"/>
<property name="org.kie.server.id" value="<KIE_SERVER_ID>"/>
<property name="org.kie.server.location" value="http://<HOST>:<PORT>/kie-
server/services/rest/server"/>
<property name="org.kie.server.controller" value="<CONTROLLER_URL>"/>
```

3. このファイルで、以下の値を置き換えます。
  - **<CONTROLLER\_USER>** および **<CONTROLLER\_PWD>** を **kie-server** ロールを持つユーザーの認証情報に置き換えます。
  - **<KIE\_SERVER\_ID>** を KIE Server システムの ID または名前に置き換えます (例: **rhdm-7.10.0-kie-server-1**)。
  - **<HOST>** を KIE Server ホストの ID または名前に置き換えます (例: **localhost** または **192.7.8.9**)。
  - **<PORT>** を KIE Server ホストのポートに置き換えます (例: **8080**)。

**注記**

**org.kie.server.location** プロパティで KIE Server の場所を指定します。

- **<CONTROLLER\_URL>** をヘッドレス Decision Manager コントローラーの URL で置き換えます。起動中に KIE Server がこの URL に接続します。

## 19.2. ヘッドレス DECISION MANAGER コントローラーの実行

ヘッドレス Decision Manager コントローラーを Red Hat JBoss EAP にインストールしたら、以下の手順に従ってヘッドレス Decision Manager コントローラーを実行します。

### 前提条件

- ヘッドレス Decision Manager コントローラーが Red Hat JBoss EAP インストールのベースディレクトリー (**EAP\_HOME**) にインストールされている。

### 手順

1. ターミナルアプリケーションで **EAP\_HOME/bin** に移動します。
2. ヘッドレス Decision Manager コントローラーを、KIE Server をインストールした Red Hat JBoss EAP インスタンスと同じ Red Hat JBoss EAP インスタンスにインストールしている場合は、以下のいずれかのコマンドを実行します。
  - Linux または UNIX ベースのシステムの場合:
 

```

$ ./standalone.sh -c standalone-full.xml

```
  - Windows の場合:
 

```

standalone.bat -c standalone-full.xml

```
3. ヘッドレス Decision Manager コントローラーを、KIE Server をインストールした Red Hat JBoss EAP インスタンスとは別の Red Hat JBoss EAP インスタンスにインストールしている場合は、**standalone.sh** スクリプトでヘッドレス Decision Manager コントローラーを開始できます。

**注記**

この場合は、**standalone.xml** ファイルに必要な設定変更を加えます。

- Linux または UNIX ベースのシステムの場合:

```

$ ./standalone.sh

```

- Windows の場合:

```

standalone.bat

```

4. ヘッドレス Decision Manager コントローラーが Red Hat JBoss EAP 上で動作していることを確認するには、以下のコマンドを入力します。ここで、**<CONTROLLER>** と **<CONTROLLER\_PWD>** は、「[ヘッドレス Decision Manager コントローラーの作成](#)」で作成



したユーザー名とパスワードの組み合わせです。このコマンドにより、KIE Server インスタンスに関する情報が出力されます。

```
curl -X GET "http://<HOST>:<PORT>/controller/rest/controller/management/servers" -H  
"accept: application/xml" -u '<CONTROLLER>:<CONTROLLER_PWD>'
```



### 注記

別の方法として、KIE Server Java API Client を使用してヘッドレス Decision Manager コントローラーにアクセスすることもできます。

## 第20章 スタンドアロン BUSINESS CENTRAL の設定および実行

Business Central スタンドアロンの JAR ファイルとして使用して、アプリケーションサーバーにデプロイせずに Business Central を実行できます。設定ファイルのサンプルを使用して、追加設定なしで Business Central スタンドアロン JAR ファイルを実行するか、要件に合わせてサンプルファイルをカスタマイズできます。



### 注記

この JAR ファイルは、Red Hat Enterprise Linux で実行している場合にのみサポートされます。

### 前提条件

- 12章 *Red Hat Decision Manager インストールファイルのダウンロード* の記載通りに、Red Hat Decision Manager 7.10.0 Business Central Standalone (**rhdm-7.10.0-decision-central-standalone.jar**) ファイルおよび Red Hat Decision Manager 7.10.0 Add Ons (**rhdm-7.10.0-addons.zip**) ファイルが、Red Hat Decision Manager 7.10 の「[Software Downloads](#)」ページからダウンロードしている。

### 手順

- ダウンロードした **rhdm-7.10.0-addons.zip** ファイルを一時ディレクトリーに展開します。このアーカイブには **rhdm-7.10.0-standalone-sample-configuration.zip** ファイルが含まれます。
- rhdm-7.10.0-standalone-sample-configuration.zip** ファイルを、**rhdm-7.10.0-decision-central-standalone.jar** ファイルが含まれるディレクトリーに展開します。**rhdm-7.10.0-standalone-sample-configuration.zip** ファイルには、以下の設定ファイルのサンプルが含まれています。
  - application-roles.properties**
  - application-users.properties**
  - sample-standalone-config.yml**  
設定ファイルで提供されるサンプルデータを使用して **rhdm-7.10.0-decision-central-standalone.jar** ファイルを実行するか、要件に合わせてデータをカスタマイズできます。
- 設定データをカスタマイズするには、以下の手順を実行します。
  - application-users.properties** ファイルを編集します。管理者ユーザーを含めます。この Business Central インスタンスが KIE Server の Decision Manager コントローラーになる場合は、Decision Manager コントローラーユーザーを含めます。以下に例を示します。

```
rhdmAdmin=password1
```

- application-roles.properties** ファイルを編集し、**application-users.properties** ファイルに含まれるユーザーにロールを割り当てます。以下に例を示します。

```
rhdmAdmin=admin
controllerUser=kie-server
```

詳細は、11章 *Red Hat Decision Manager ロールおよびユーザー* を参照してください。

- sample-standalone-config.yml** 設定ファイルの内容を確認し、必要に応じて更新します。

4. Business Central スタンドアロン JAR ファイルを実行するには、以下のコマンドを入力します。

```
java -jar rhdm-7.10.0-decision-central-standalone.jar -s sample-standalone-config.yml
```

5. JAR ファイルの実行時にアプリケーションプロパティを設定するには、コマンドに -**D<PROPERTY>=<VALUE>** パラメーターを追加します。ここで、<PROPERTY> はサポートされるアプリケーションプロパティの名前で、<VALUE> はプロパティの値になります。

```
java -jar rhdm-7.10.0-decision-central-standalone.jar -s sample-standalone-config.yml -  
D<PROPERTY>=<VALUE> -D<PROPERTY>=<VALUE>
```

たとえば、Business Central を実行し、ユーザー **controllerUser** として KIE Server に接続するには、次のコマンドを実行します。

```
java -jar {PRODUCT_FILE}-{URL_COMPONENT_CENTRAL}-standalone.jar \  
-s sample-standalone-config.yml \  
-Dorg.kie.server.user=controllerUser  
-Dorg.kie.server.pwd=controllerUser1234
```

これにより、コンテナを KIE Server にデプロイできます。詳細は、[付録A Business Central システムプロパティ](#)を参照してください。

## 第21章 RED HAT DECISION MANAGER の MAVEN 設定およびリポジトリ

Red Hat Decision Manager プロジェクトの作成時に、Business Central は Business Central 用に設定された Maven リポジトリを使用します。Maven project object model (POM) ファイル (**pom.xml**) を変更することで、Maven グローバルまたはユーザー設定を使用し、すべての Red Hat Decision Manager プロジェクトに対して Red Hat Decision Manager の公開リポジトリから依存関係を取得するように指示できます。また、Business Central と KIE Server が外部の Maven リポジトリを使用するか、Maven のミラーを準備してオフラインでできるように設定できます。

Red Hat Decision Manager プロジェクトのパッケージ化およびデプロイメントオプションに関する詳細は、[Red Hat Decision Manager プロジェクトのパッケージ化およびデプロイ](#) を参照してください。

### 21.1. RED HAT DECISION MANAGER の MAVEN 依存関係の追加

ご自分の Red Hat Decision Manager プロジェクトで適切な Maven 依存関係を使用するには、プロジェクトの **pom.xml** ファイルに Red Hat Business Automation の BOM (bill of materials) ファイルを追加します。Red Hat Business Automation BOM は、Red Hat Decision Manager と Red Hat Process Automation Manager の両方に適用されます。BOM ファイルを追加すると、提供される Maven リポジトリから、推移的依存関係の適切なバージョンがプロジェクトに含まれます。

Red Hat Business Automation BOM (Bill of Materials) についての詳細情報は、[What is the mapping between Red Hat Decision Manager and the Maven library version?](#) を参照してください。

#### 手順

1. Red Hat Business Automation BOM を **pom.xml** ファイルで宣言します。

```
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>com.redhat.ba</groupId>
      <artifactId>ba-platform-bom</artifactId>
      <version>7.10.0.redhat-00004</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
<dependencies>
  <!-- Your dependencies -->
</dependencies>
```

2. **<dependencies>** タグでお使いのプロジェクトに必要な依存関係を宣言します。製品の BOM をプロジェクトにインポートしたら、ユーザー向け製品依存関係のバージョンが定義されるため、**<dependency>** 要素のサブ要素 **<version>** を指定する必要はありません。ただし、**<dependency>** 要素を使用して、プロジェクトで使用する依存関係を宣言する必要があります。
3. Business Central に作成されない標準のプロジェクトでは、お使いのプロジェクトに必要な依存関係をすべて指定します。Business Central に作成するプロジェクトでは、基本的なデシジョンエンジンの依存関係が Business Central に自動的に提供されます。

- 標準的な Red Hat Decision Manager プロジェクトでは、使用する機能に応じて、以下の依存関係を宣言します。
- 標準的な Red Hat Decision Manager プロジェクトでは、以下の依存関係を宣言します。

### 埋め込みデシジョンエンジン依存関係

```
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-compiler</artifactId>
</dependency>

<!-- Dependency for persistence support. -->
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-persistence-jpa</artifactId>
</dependency>

<!-- Dependencies for decision tables, templates, and scorecards.
For other assets, declare org.drools:business-central-models-* dependencies. -->
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-decisiontables</artifactId>
</dependency>
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-templates</artifactId>
</dependency>
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-scorecards</artifactId>
</dependency>

<!-- Dependency for loading KJARs from a Maven repository using KieScanner. -->
<dependency>
  <groupId>org.kie</groupId>
  <artifactId>kie-ci</artifactId>
</dependency>
```

- KIE Server を使用するには、以下の依存関係を宣言します。

### クライアントアプリケーション KIE Server の依存関係

```
<dependency>
  <groupId>org.kie.server</groupId>
  <artifactId>kie-server-client</artifactId>
</dependency>
```

- Red Hat Decision Manager にリモートクライアントを作成するには、以下の依存関係を宣言します。

### クライアントの依存関係

```
<dependency>
  <groupId>org.uberfire</groupId>
```

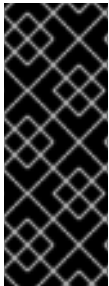
```
<artifactId>uberfire-rest-client</artifactId>
</dependency>
```

- ルール、プロセス定義など、アセットを含む JAR ファイルを作成する場合は、お使いの Maven プロジェクトのパッケージングの種類を **kjar** と指定し、**org.kie:kie-maven-plugin** を使用して、**<project>** 要素に置かれた **kjar** パッケージングタイプを処理します。以下の例の **\${kie.version}** は、[What is the mapping between Red Hat Decision Manager and the Maven library version?](#) に記載されている Maven ライブラリーのバージョンです。

```
<packaging>kjar</packaging>
<build>
  <plugins>
    <plugin>
      <groupId>org.kie</groupId>
      <artifactId>kie-maven-plugin</artifactId>
      <version>${kie.version}</version>
      <extensions>true</extensions>
    </plugin>
  </plugins>
</build>
```

## 21.2. BUSINESS CENTRAL および KIE SERVER への外部 MAVEN リポジトリの設定

Business Central および KIE Server が、内部のリポジトリではなく、Nexus や Artifactory などの外部の Maven リポジトリを使用するように設定できます。このように設定することで、Business Central と KIE Server は外部の Maven リポジトリで管理されているアーティファクトにアクセスしてダウンロードできます。



### 重要

Maven ではアーティファクトが不変である必要があるため、リポジトリ内のアーティファクトは自動セキュリティパッチを受け取りません。その結果、既知のセキュリティ問題のパッチがないアーティファクトはリポジトリに残り、これらに依存するビルドが破損しないようにします。パッチが適用されたアーティファクトのバージョン番号がインクリメントされます。詳細は、[JBoss Enterprise Maven リポジトリ](#) を参照してください。



### 注記

Red Hat OpenShift Container Platform のオーサリング環境向けに外部の Maven リポジトリを設定する方法については、以下のドキュメントを参照してください。

- 『[Operator を使用した Red Hat OpenShift Container Platform 4 への Red Hat Decision Manager 環境のデプロイメント](#)』
- [テンプレートを使用した Red Hat OpenShift Container Platform 3 への Red Hat Decision Manager 環境のデプロイメント](#)

### 前提条件

- Business Central および KIE Server がインストールされている。インストールオプションは、[Red Hat Decision Manager インストールの計画](#) を参照してください。

## 手順

1. 外部リポジトリの接続およびアクセスの詳細が含まれる Maven **settings.xml** ファイルを作成します。**settings.xml** ファイルの詳細は Maven の [Settings Reference](#) を参照してください。
2. 既知の場所 (例: **/opt/custom-config/settings.xml**) にファイルを保存します。
3. Red Hat Decision Manager インストールディレクトリーで、**standalone-full.xml** ファイルに移動します。たとえば、Red Hat Decision Manager に Red Hat JBoss EAP インストールを使用する場合は **\$EAP\_HOME/standalone/configuration/standalone-full.xml** に移動します。
4. **standalone-full.xml** の **<system-properties>** タグで、**kie.maven.settings.custom** プロパティーに **settings.xml** ファイルのフルパス名を設定します。  
以下に例を示します。

```
<property name="kie.maven.settings.custom" value="/opt/custom-config/settings.xml"/>
```

5. Business Central と KIE Server を起動または再起動します。

## 次のステップ

KJAR アーティファクトとして外部の Maven リポジトリにエクスポートまたはプッシュする Business Central のプロジェクトごとに、プロジェクトの **pom.xml** ファイルにリポジトリの情報を追加する必要があります。詳細は [Red Hat Decision Manager プロジェクトのパッケージ化およびデプロイ](#) を参照してください。

## 21.3. オフラインで使用する MAVEN ミラーリポジトリの用意

公開インターネットへの送信アクセスが、Red Hat Decision Manager のデプロイメントに設定されていない場合は、必要なすべてのアーティファクトのミラーが含まれる Maven リポジトリを用意して、このリポジトリを使用できるようにする必要があります。



### 注記

Red Hat Decision Manager デプロイメントがインターネットに接続されている場合は、この手順を実行する必要はありません。

## 前提条件

- 公開インターネットへの送信アクセスが設定されているコンピューターが利用できる。

## 手順

1. 公開インターネットに送信アクセスができるコンピューターで、以下のアクションを実行します。
2. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **製品:** Red Hat Decision Manager
  - **バージョン:** 7.10
    - a. **Red Hat Process Automation Manager 7.10.0 Offliner Content List (rhdm-7.10.0-offliner.zip)** の製品配信可能ファイルをダウンロードして展開します。



b. **rhdm-7.10.0-offliner.zip** ファイルの内容を任意のディレクトリーに展開します。

c. ディレクトリーに移動し、以下のコマンドを入力します。

```
./offline-repo-builder.sh offliner.txt
```

このコマンドは、**repository** サブディレクトリーを作成し、必要なアーティファクトをこのサブディレクトリーにダウンロードします。これはミラーリポジトリーです。

一部のダウンロードが失敗したことを示すメッセージが表示された場合は、同じコマンドを再度実行してください。ダウンロードが再び失敗する場合は、Red Hat サポートに連絡してください。

3. Business Central 外でサービスを開発し、追加の依存関係がある場合は、ミラーリポジトリーにその依存関係を追加します。サービスを Maven プロジェクトとして開発した場合は、以下の手順を使用し、これらの依存関係を自動的に用意します。公開インターネットへに送信接続できるコンピューターで、この手順を実行します。

a. ローカルの Maven キャッシュディレクトリー (`~/.m2/repository`) のバックアップを作成して、ディレクトリーを削除します。

b. **mvn clean install** コマンドを使用してプロジェクトのソースをビルドします。

c. すべてのプロジェクトで以下のコマンドを入力し、Maven を使用してプロジェクトで生成したすべてのアーティファクトのランタイムの依存関係をすべてダウンロードするようにします。

```
mvn -e -DskipTests dependency:go-offline -f /path/to/project/pom.xml --batch-mode -Djava.net.preferIPv4Stack=true
```

`/path/to/project/pom.xml` を、プロジェクトの **pom.xml** ファイルのパスに置き換えます。

d. ローカルの Maven キャッシュディレクトリー (`~/.m2/repository`) の内容を、作成した **repository** サブディレクトリーにコピーします。

4. **repository** サブディレクトリーのコンテンツを、Red Hat Decision Manager をデプロイしたコンピューターのディレクトリーにコピーします。このディレクトリーがオフラインの Maven ミラーリポジトリーになります。

5. 「[Business Central および KIE Server への外部 Maven リポジトリーの設定](#)」の説明に従って、Red Hat Decision Manager デプロイメント向けに、**settings.xml** ファイルを作成して設定します。

6. **settings.xml** ファイルで以下を変更します。

- `<profile>` タグの下に、`<repositories>` タグまたは `<pluginRepositores>` タグがない場合は、不足しているタグを追加します。
- `<repositories>` の下に、以下のコンテンツを追加します。

```
<repository>
  <id>offline-repository</id>
  <url>file:///path/to/repo</url>
  <releases>
    <enabled>true</enabled>
  </releases>
  <snapshots>
```



```
<enabled>false</enabled>
</snapshots>
</repository>
```

**/path/to/repo** は、ローカルの Maven ミラーリポジトリのディレクトリーに対する完全パスに置き換えます。

- **<pluginRepositories>** の下に、以下のコンテンツを追加します。

```
<repository>
  <id>offline-plugin-repository</id>
  <url>file:///path/to/repo</url>
  <releases>
    <enabled>true</enabled>
  </releases>
  <snapshots>
    <enabled>false</enabled>
  </snapshots>
</repository>
```

**/path/to/repo** は、ローカルの Maven ミラーリポジトリのディレクトリーに対する完全パスに置き換えます。

7. Business Central の **kie.maven.offline.force** プロパティを **true** に設定します。Business Central のプロパティの設定手順は、[Red Hat JBoss EAP 7.3 への Red Hat Decision Manager のインストールおよび設定](#) を参照してください。

## 第22章 GIT リポジトリからのプロジェクトのインポート

Git は分散バージョン管理システムです。リビジョンをコミットオブジェクトとして実装します。リポジトリに変更を保存すると、Git リポジトリに新しいコミットオブジェクトが作成されます。

Business Central は Git を使用してプロジェクトデータ (ルールやプロセスなどのアセットを含む) を格納します。Business Central でプロジェクトを作成すると、Business Central に埋め込まれている Git リポジトリに追加されます。他の Git リポジトリにプロジェクトがある場合は、Business Central スペースから、そのプロジェクトを Business Central の Git リポジトリにインポートできます。

### 前提条件

- Red Hat Decision Manager プロジェクトが外部の Git リポジトリに存在している。
- 外部の Git リポジトリへの読み取りアクセスに必要な認証情報がある。

### 手順

1. Business Central で **Menu → Design → Projects** の順にクリックします。
2. プロジェクトをインポートするスペースを選択または作成します。デフォルトのスペースは **MySpace** です。
3. プロジェクトをインポートするには、次のいずれかを実行します。
  - **Import Project** をクリックします。
  - ドロップダウンリストから **Import Project** を選択します。
4. **Import Project** ウィンドウに、インポートするプロジェクトが含まれる Git リポジトリの URL および認証情報を入力し、**Import** をクリックします。プロジェクトが Business Central の Git リポジトリに追加され、現在のスペースで利用できるようになります。

## 第23章 BUSINESS CENTRAL ブランディングのカスタマイズ

独自のイメージに置き換えることで、Business Central ログインページやアプリケーションヘッダーのブランディングをカスタマイズできます。

### 23.1. BUSINESS CENTRAL ログインページのカスタマイズ

Business Central ログインページで、企業のロゴおよびプロジェクトロゴをカスタマイズできます。

#### 手順

1. Red Hat JBoss EAP を起動し、Web ブラウザーで Business Central を開きます。
2. SVG 形式のイメージを、Red Hat Decision Manager インストールの **EAP\_HOME/standalone/deployments/decision-central.war/img/** ディレクトリーにコピーします。
3. **EAP\_HOME/standalone/deployments/decision-central.war/img/** ディレクトリーで、既存の **redhat\_logo.png** ファイルを移動するか名前を変更します。
4. PNG ファイル **redhat\_logo.png** の名前を変更します。
5. **User name** フィールドおよび **Password** フィールドの上に表示されるプロジェクトロゴを変更するには、デフォルトのイメージ **BC\_Logo.png** を新しい SVG ファイルに置き換えます。
6. ログインページを強制的に完全に再読み込みし、キャッシュを消去して変更を表示します。たとえば、Linux および Windows の多くの Web ブラウザーでは、Ctrl+F5 を押します。

### 23.2. BUSINESS CENTRAL アプリケーションヘッダーのカスタマイズ

Business Central アプリケーションヘッダーをカスタマイズできます。

#### 手順

1. Red Hat JBoss EAP を起動し、Web ブラウザーで Business Central を開き、お使いのユーザー認証情報でログインします。
2. SVG 形式の新しいアプリケーションヘッダーイメージを、Red Hat Decision Manager システムの **EAP\_HOME/standalone/deployments/decision-central.war/banner/** ディレクトリーにコピーします。
3. テキストエディターで **EAP\_HOME/standalone/deployments/decision-central.war/banner/banner.html** ファイルを開きます。
4. **<img>** タグの **logo.png** を、新しい **image:admin-and-config/** のファイル名に置き換えます。

```

```

5. ログインページを強制的に完全に再読み込みし、キャッシュを消去して変更を表示します。たとえば、Linux および Windows の多くの Web ブラウザーでは、Ctrl+F5 を押します。

## 第24章 LDAP と SSL の統合

Red Hat Decision Manager を使用して、RH-SSO を介して LDAP および SSL を統合できます。詳細は [Red Hat Single Sign-On サーバー管理ガイド](#) を参照してください。

## 付録A BUSINESS CENTRAL システムプロパティー

このセクションに記載の Business Central のシステムプロパティーは **standalone\*.xml** ファイルに渡されます。

### Git ディレクトリー

以下のプロパティーを使用して、Business Central Git ディレクトリーの場所と名前を設定します。

- **org.uberfire.nio.git.dir**: Business Central の Git ディレクトリーの場所。
- **org.uberfire.nio.git.dirname**: Business Central の Git ディレクトリーの名前。デフォルト値は **.niogit** です。
- **org.uberfire.nio.git.ketch**: Git ketch を有効化または無効化。
- **org.uberfire.nio.git.hooks**: Business Central の Git ディレクトリーの場所。

### HTTP 経由の Git

次のプロパティーを使用して、HTTP 経由で Git リポジトリにアクセスできるように設定します。

- **org.uberfire.nio.git.proxy.ssh.over.http**: SSH が HTTP プロキシを使用するかどうかを指定します。デフォルト値は **false** です。
- **http.proxyHost**: HTTP プロキシのホスト名を定義します。デフォルト値は **null** です。
- **http.proxyPort**: HTTP プロキシのポート (整数値) を定義します。デフォルト値は **null** です。
- **http.proxyUser**: HTTP プロキシ名を定義します。
- **http.proxyPassword**: HTTP プロキシのユーザーパスワードを定義します。
- **org.uberfire.nio.git.http.enabled**: HTTP デモンを有効または無効にします。デフォルト値は **true** です。
- **org.uberfire.nio.git.http.host**: このデモンは、HTTP デモンが有効な場合にホストの識別子としてこのプロパティーを使用します。これは、HTTP 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTP は、継続してサブレットコンテナに依存します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.http.hostname**: HTTP デモンが有効な場合に、このデモンはホスト名の識別子としてこのプロパティーを使用します。これは、HTTP 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTP は、継続してサブレットコンテナに依存します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.http.port**: このデモンは、HTTP デモンが有効な場合にポート番号としてこのプロパティーを使用します。これは、HTTP 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTP は、継続してサブレットコンテナに依存します。デフォルト値は **8080** です。

### HTTPS 経由の Git

次のプロパティーを使用して、HTTPS 経由で Git リポジトリにアクセスできるように設定します。

- **org.uberfire.nio.git.proxy.ssh.over.https**: SSH が HTTPS プロキシを使用するかどうかを指定します。デフォルト値は **false** です。

- **https.proxyHost**: HTTPS プロキシのホスト名。デフォルト値は **null** です。
- **https.proxyPort**: HTTPS プロキシのポート (整数値)。デフォルト値は **null** です。
- **https.proxyUser**: HTTPS プロキシ名を定義します。
- **https.proxyPassword**: HTTPS プロキシのユーザーパスワードを定義します。
- **user.dir**: ユーザーディレクトリーの場所。
- **org.uberfire.nio.git.https.enabled**: HTTPS デーモンを有効または無効にします。デフォルト値は **false** です。
- **org.uberfire.nio.git.https.host**: このデーモンは、HTTPS デーモンが有効な場合にホストの識別子としてこのプロパティを使用します。これは、HTTPS 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTPS は、継続してサブレットコンテナに依存します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.https.hostname**: このデーモンは、HTTPS デーモンが有効な場合にホスト名の識別子としてこのプロパティを使用します。これは、HTTPS 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTPS は、継続してサブレットコンテナに依存します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.https.port**: このデーモンは、HTTPS デーモンが有効な場合にポート番号としてこのプロパティを使用します。これは、HTTPS 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTPS は、継続してサブレットコンテナに依存します。デフォルト値は **8080** です。

## JGit

- **org.uberfire.nio.jgit.cache.instances**: JGit キャッシュサイズを定義します。
- **org.uberfire.nio.jgit.cache.overflow.cleanup.size**: JGit キャッシュオーバーフローのクリーンアップサイズを定義します。
- **org.uberfire.nio.jgit.remove.eldest.iterations**: 最も古い JGit の反復を削除するかどうかを定義します。
- **org.uberfire.nio.jgit.cache.evict.threshold.duration**: JGit 退避のしきい値の期間を定義します。
- **org.uberfire.nio.jgit.cache.evict.threshold.time.unit**: JGit 退避のしきい値の時間単位を定義します。

## Git デーモン

次のプロパティを使用して、Git デーモンを有効にして設定します。

- **org.uberfire.nio.git.daemon.enabled**: Git デーモンを有効または無効にします。デフォルト値は **true** です。
- **org.uberfire.nio.git.daemon.host**: Git デーモンが有効な場合は、このプロパティをローカルホストの識別子として使用します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.daemon.hostname**: Git デーモンが有効な場合は、このプロパティをローカルホスト名の識別子として使用します。デフォルト値は **localhost** です。

- **org.uberfire.nio.git.daemon.port**: Git デーモンが有効な場合は、このプロパティーをポート番号として使用します。デフォルト値は **9418** です。
- **org.uberfire.nio.git.http.sslVerify**: Git リポジトリを確認する SSL 証明書を有効または無効にします。デフォルト値は **true** です。



### 注記

デフォルトポートまたは割り当てられたポートが既に使用されている場合は、別のポートが自動的に選択されます。ポートが利用可能であることを確認し、詳細についてはログをチェックします。

## Git SSH

次のプロパティーを使用して、Git SSH デーモンを有効にして設定します。

- **org.uberfire.nio.git.ssh.enabled**: SSH デーモンを有効または無効にします。デフォルト値は **true** です。
- **org.uberfire.nio.git.ssh.host**: SSH デーモンが有効な場合は、このプロパティーをローカルホスト識別子として使用します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.ssh.hostname**: SSH デーモンが有効な場合は、このプロパティーをローカルホスト名の識別子として使用します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.ssh.port**: SSH デーモンが有効な場合は、このプロパティーをポート番号として使用します。デフォルト値は **8001** です。



### 注記

デフォルトポートまたは割り当てられたポートが既に使用されている場合は、別のポートが自動的に選択されます。ポートが利用可能であることを確認し、詳細についてはログをチェックします。

- **org.uberfire.nio.git.ssh.cert.dir**: ローカルの証明書が保存される **.security** ディレクトリーの場所。デフォルトは作業ディレクトリーです。
- **org.uberfire.nio.git.ssh.idle.timeout**: SSH のアイドルタイムアウトを設定します。
- **org.uberfire.nio.git.ssh.passphrase**: SCP スタイルの URL を持つ Git リポジトリのクローンを作成する場合に、オペレーティングシステムの公開キーストアにアクセスするためのパスフレーズ。たとえば、**git@github.com:user/repository.git** です。
- **org.uberfire.nio.git.ssh.algorithm**: SSH で使用されるアルゴリズム。デフォルト値は **RSA** です。
- **org.uberfire.nio.git.gc.limit**: GC の制限を設定します。
- **org.uberfire.nio.git.ssh.ciphers**: コンマ区切りの暗号化の文字列。利用可能な暗号化は **aes128-ctr**、**aes192-ctr**、**aes256-ctr**、**arcfour128**、**arcfour256**、**aes192-cbc**、**aes256-cbc** です。このプロパティーを使用しない場合は、すべての暗号化が読み込まれます。
- **org.uberfire.nio.git.ssh.macs**: コンマ区切りのメッセージ認証コード (MAC) の文字列。利用可能な MAC は **hmac-md5**、**hmac-md5-96**、**hmac-sha1**、**hmac-sha1-96**、**hmac-sha2-256**、**hmac-sha2-512** です。このプロパティーを使用しない場合は、すべての MAC が読み込まれます。



### 注記

RSA、または DSA 以外のアルゴリズムを使う場合は、Bouncy Castle JCE ライブラリーを使用するようにアプリケーションサーバーを設定します。

## KIE Server ノードおよび Decision Manager コントローラー

以下のプロパティーを使用して Decision Manager コントローラーと KIE Server ノードの接続を設定します。

- **org.kie.server.controller:** URL は Decision Manager コントローラーへの接続に使用されます。たとえば、**ws://localhost:8080/decision-central/websocket/controller** です。
- **org.kie.server.user:** Decision Manager コントローラーから KIE Server ノードに接続する時に使用するユーザー名。このプロパティーが必要なのは、この Business Central システムを Decision Manager コントローラーとして使用する場合があります。
- **org.kie.server.pwd:** Decision Manager コントローラーから KIE Server ノードに接続する時に使用するパスワード。このプロパティーが必要なのは、この Business Central システムを Decision Manager コントローラーとして使用する場合があります。

## Maven など

以下のプロパティーを使用して、Maven などの機能を設定します。

- **kie.maven.offline.force:** Maven のオフライン動作を強制します。true に設定すると、オンラインの依存関係解決が無効になります。デフォルト値は **false** です。



### 注記

このプロパティーは、Business Central にのみ使用してください。他のコンポーネントとランタイム環境を共有する場合は、設定を分離して、Business Central にだけ適用してください。

- **org.uberfire.gzip.enable:** **GzipFilter** 圧縮フィルターで Gzip の圧縮を有効にするか、または無効にします。デフォルト値は **true** です。
- **org.kie.workbench.profile:** Business Central プロファイルを選択します。許容値は、**FULL** または **PLANNER\_AND\_RULES** です。プリフィックス **FULL\_** で、プロファイルを設定し、管理者設定にこのプロファイルの設定が表示されないようにします。デフォルト値は **FULL** です。
- **org.appformer.m2repo.url:** Business Central は依存関係を検索する時に、Maven リポジトリのデフォルトの場所を使用します。デフォルト値は、**http://localhost:8080/business-central/maven2** など、Business Central 内の Maven リポジトリを参照します。このプロパティーは、Business Central が起動する前に設定してください。デフォルト値は、内部の **m2** リポジトリへのファイルパスです。
- **appformer.ssh.keystore:** クラス名を指定して Business Central で使用する、カスタムの SSH キーストアを定義します。このプロパティーが指定されていない場合はデフォルトの SSH キーストアを使用します。
- **appformer.ssh.keys.storage.folder:** このプロパティーは、デフォルトの SSH キーストアを使用する場合にユーザーの SSH 公開鍵の保存フォルダーを定義します。このプロパティーを指定しないと、この公開鍵は Business Central の **.security** フォルダーに保存されます。



- **appformer.experimental.features**: 実験的機能のフレームワークを有効にします。デフォルト値は **false** です。
- **org.kie.demo**: GitHub 外部へのデモアプリケーションのクローン作成を有効にします。
- **org.uberfire.metadata.index.dir**: Lucene の **.index** ディレクトリーが保存される場所。デフォルトは作業ディレクトリーです。
- **org.uberfire.ldap.regex.role\_mapper**: LDAP プリンシパル名をアプリケーションのロール名にマッピングするのに使用する regex パターン。プリンシパルの値とロール名が一致する場合は、アプリケーションのロール名が変数ロールに置き換えられるため、変数ロールはパターンの一部でなければならない点に注意してください。
- **org.uberfire.sys.repo.monitor.disabled**: 設定モニターを無効にします。無効にした場合の影響を正しく理解していない場合は、無効にしないでください。デフォルト値は **false** です。
- **org.uberfire.secure.key**: パスワードの暗号化で使用するパスワード。デフォルト値は **org.uberfire.admin** です。
- **org.uberfire.secure.alg**: パスワードの暗号化で使用する暗号化アルゴリズム。デフォルト値は **PBEWithMD5AndDES** です。
- **org.uberfire.domain**: uberfire が使用するセキュリティドメイン名。デフォルト値は **ApplicationRealm** です。
- **org.guvnor.m2repo.dir**: Maven リポジトリディレクトリーが保存される場所。デフォルト値は **<working-directory>/repositories/kie** です。
- **org.guvnor.project.gav.check.disabled**: グループ ID、アーティファクト ID、およびバージョン (GAV) のチェックを無効にします。デフォルト値は **false** です。
- **org.kie.build.disable-project-explorer**: Project Explorer で選択したプロジェクトの自動ビルドを無効にします。デフォルト値は **false** です。
- **org.kie.builder.cache.size**: プロジェクトビルダーのキャッシュサイズを定義します。デフォルト値は **20** です。
- **org.kie.library.assets\_per\_page**: プロジェクト画面のページごとのアセット数をカスタマイズできます。デフォルト値は **15** です。
- **org.kie.verification.disable-dtable-realtime-verification**: デシジョンテーブルのリアルタイム確認および検証を無効にします。デフォルト値は **false** です。

## Decision Manager コントローラー

以下のプロパティを使用して、Decision Manager コントローラーへの接続方法を設定します。

- **org.kie.workbench.controller**: Decision Manager コントローラーとの接続に使用する URL。たとえば、**ws://localhost:8080/kie-server-controller/websocket/controller** です。
- **org.kie.workbench.controller.user**: Decision Manager コントローラーのユーザー名。デフォルト値は **kieserver** です。
- **org.kie.workbench.controller.pwd**: Decision Manager コントローラーのパスワード。デフォルト値は **kieserver1!** です。

- **org.kie.workbench.controller.token:** Decision Manager コントローラーとの接続に使用するトークン文字列。

### Java Cryptography Extension KeyStore (JCEKS)

JCEKS を設定するには、以下のプロパティを使用します。

- **kie.keystore.keyStoreURL:** Java Cryptography Extension KeyStore (JCEKS) の読み込みに使用する URL。たとえば、**file:///home/kie/keystores/keystore.jceks** です。
- **kie.keystore.keyStorePwd:** JCEKS に使用するパスワード。
- **kie.keystore.key.ctrl.alias:** デフォルトの REST Decision Manager コントローラーに使用するキーのエイリアス。
- **kie.keystore.key.ctrl.pwd:** デフォルトの REST Decision Manager コントローラーのエイリアスのパスワード。

### レンダリング

以下のプロパティを使用して、Business Central と KIE Server のレンダリングフォームを切り替えます。

- **org.jbpm.wb.forms.renderer.ext:** Business Central と KIE Server のフォームのレンダリングを切り替えます。デフォルトでは、フォームのレンダリングは Business Central が行います。デフォルト値は **false** です。
- **org.jbpm.wb.forms.renderer.name:** Business Central と KIE Server のレンダリングフォームを切り替えることができます。デフォルト値は **workbench** です。

## 第25章 関連情報

- [Red Hat JBoss EAP クラスター環境への Red Hat Decision Manager のインストールおよび設定](#)

## 第26章 次のステップ

- [Red Hat Decision Manager と Red Hat Single Sign-On の統合](#)
- [デシジョンサービスのスタートガイド](#)

## パート III. RED HAT JBOSS EAP クラスター環境への RED HAT DECISION MANAGER のインストールおよび設定

システムエンジニアは、Red Hat Decision Manager クラスター環境を作成して、開発環境およびランタイム環境に高可用性および負荷分散を提供できます。

### 前提条件

- [Red Hat Decision Manager インストールの計画](#) の内容を確認している。

## 第27章 RED HAT DECISION MANAGER のクラスター

2 台以上のコンピューターをクラスターリングすると、高可用性、コラボレーションの強化、負荷分散の利点があります。高可用性により、1 台のコンピューターで障害が発生したときにデータが損失する可能性を減らすことができます。その障害が発生したコンピューターにあるデータのコピーを提供することで、コンピューターに障害が発生したときに、別のコンピューターが不足を補います。障害が発生したコンピューターが再度オンラインになったら、クラスターに戻ります。

Red Hat Decision Manager コンポーネントのクラスターリングを行う方法は複数あります。本書は、以下のシナリオにおけるクラスターリングの方法を説明します。

- [28章 開発 \(オーサリング\) 環境における Red Hat Decision Manager クラスター](#)
- [29章 ランタイム環境における KIE Server クラスター](#)

## 第28章 開発 (オーサリング) 環境における RED HAT DECISION MANAGER クラスター

Red Hat Decision Manager をクラスター開発環境として設定すると、高可用性の利点が得られます。クラスター環境では、開発者がノードで作業していて、そのノードで障害が発生した場合、この開発者が作業した内容はクラスターの別のノードに保存され、そちらで確認できます。

多くの開発環境には、Business Central と最低でも 1 台 KIE Server が含まれています。

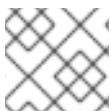
Red Hat Decision Manager のクラスター開発環境を構築するには、以下のタスクを実行する必要があります。

1. クラスターのノードとなる各システムに以下のコンポーネントを設定します。
  - Red Hat JBoss EAP 7.3 with Red Hat Data Grid 7.3 を設定します。  
Red Hat Data Grid は Infinispan のオープンソースソフトウェアプロジェクトで構築されています。Red Hat Data Grid は、インデックス化の機能が含まれた、分散型インメモリーキー/値のデータストアで、大量のデータを素早くほぼリアルタイムに保存、検索、および解析できます。Red Hat Decision Manager のクラスター環境では、クラスターノード全体にわたる複雑な検索を効率的に実施できます。
  - Java メッセージングサーバー (JMS) ブローカーである、AMQ Broker を設定します。  
JMS ブローカーは、ローカルでメッセージを受信して保存し、そのメッセージを受信者に転送するソフトウェアコンポーネントです。AMQ Broker を使用すると、アプリケーションがメッセージングプロバイダーと通信できます。また、メッセージ駆動型 Bean、Enterprise JavaBean、servlet などのコンポーネントがどのようにメッセージを送受信するかを指定します。
  - NFS ファイルサーバーを設定します。
2. Red Hat JBoss EAP 7.3 および Red Hat Decision Manager 7.10 をダウンロードして、各システムにインストールします。各システムがクラスターノードになります。
3. クラスターの各ノードで Business Central を設定して起動します。

### 28.1. RED HAT DATA GRID のインストールおよび設定

クラスターノード全体でより効率的な検索を行うために、Red Hat Decision Manager のクラスター化環境に Red Hat Data Grid をインストールして設定します。

Red Hat Data Grid の高度なインストールおよび設定オプション、ならびに Red Hat JBoss EAP の Red Hat Data Grid モジュールに関する情報は、[Red Hat Data Grid User Guide](#)を参照してください。



#### 注記

Business Central と同じノードに Red Hat Data Grid はインストールしないでください。

#### 前提条件

- Java 8.0 以降と互換性のある Java Virtual Machine (JVM) 環境がインストールされている。

#### 手順

1. Red Hat カスタマーポータル[の Software Downloads ページ](#)に移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product:** Data Grid
  - **Version:** 7.3
2. お使いのシステムの任意の場所に、**Red Hat JBoss Data Grid 7.3.0 Server(jboss-datagrid-7.3.0-1-server.zip)** のインストールファイルをダウンロードして展開します。以下の例では、展開したディレクトリーは **JDG\_HOME** と呼ばれます。
3. Red Hat Data Grid を最新のバージョンに更新します。詳細は、[Red Hat Data Grid User Guide](#) を参照してください。
4. Red Hat Data Grid を実行するには、**JDG\_HOME/bin** に移動して以下のコマンドを入力します。

```
$ ./standalone.sh -c clustered.xml -b <HOST>
```

**<HOST>** は、Red Hat Data Grid をインストールしたサーバーの IP アドレスまたはホスト名に置き換えます。

## 28.2. AMQ BROKER のダウンロードおよび設定

Red hat AMQ Broker により、アプリケーションがメッセージングプロバイダーに接続できるようになります。また、メッセージ駆動型 Bean、Enterprise JavaBean、servlet などのコンポーネントがどのようにメッセージを送受信するかを指定します。

高度なインストールおよび設定オプションの詳細は、[Getting started with AMQ Broker](#) を参照してください。

### 手順

1. Red Hat カスタマーポータル[の Software Downloads ページ](#)に移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product:** AMQ Broker
  - **バージョン:** 7.8.0
2. Red Hat AMQ Broker 7.8.0(**amq-broker-7.8.0-bin.zip**) の横の **Download** をクリックします。
3. **amq-broker-7.8.0-bin.zip** ファイルを展開します。
4. **amq-broker-7.8.0-bin/amq-broker-7.8.0/bin** のディレクトリーに移動します。
5. 以下のコマンドを入力し、ブローカーおよびブローカーユーザーを作成します。

```
./artemis create --host <HOST> --user <AMQ_USER> --password <AMQ_PASSWORD> --require-login <BROKER_NAME>
```

上記の例で、以下のプレースホルダーを置き換えてください。

- **<HOST>** は、AMQ Broker をインストールしたサーバーの IP アドレスまたはホスト名に置き換えます。



- **<AMQ\_USER>** および **<AMQ\_PASSWORD>** は、任意のユーザー名とパスワードの組み合わせに置き換えます。
  - **<BROKER\_NAME>** は作成するブローカーの名前に置き換えます。
6. AMQ Broker を実行するには、**amq-broker-7.8.0-bin/amq-broker-7.8.0/bin** ディレクトリーに移動し、以下のコマンドを入力します。

```
<BROKER_NAME>/bin/artemis run
```

## 28.3. NFS バージョン 4 サーバーの設定

Business Central クラスター環境には、NFS バージョン 4 サーバーの共有ファイルシステムが必要で、その共有ファイルシステムに、各クラスターノードからアクセスできる必要がある。

### 手順

1. NFS バージョン 4 共有をエクスポートするようにサーバーを設定します。Red Hat Enterprise Linux での NFS 共有のエクスポートの方法については、[ファイルシステムの管理](#) の **NFS 共有のエクスポート** を参照してください。NFS サーバーの作成に関する情報は、[RHEL 7 で NFS を設定する](#) を参照してください。
2. テキストエディターで このサーバーの **/etc/exports** ファイルを開きます。
3. **/etc/exports** ファイルに以下の行を追加します。**<HOST\_LIST>** は、サーバーへの接続が許可されているホストのスペース区切りリストに置き換えます。

```
/opt/kie/data <HOST_LIST>
```

以下に例を示します。

```
/opt/kie/data 192.268.1.0/24(rw,sync) 192.268.1.1/24(no_root_squash)
```

これにより、NFS に必要となる最小オプション **rw,sync,no\_root\_squash** を使用して **/opt/kie/data** 共有が作成されます。



### 注記

**/opt/kie/data** の代わりに別の共有名を使用できます。別の共有名を使用する場合には、Business Central を実行する全ノードの設定時に、この名前を使用する必要があります。

## 28.4. RED HAT JBOSS EAP 7.3 および RED HAT DECISION MANAGER のダウンロードおよび展開

クラスターの各ノードで Red Hat JBoss EAP 7.3 および Red Hat Decision Manager 7.10 をダウンロードして、インストールします。

### 手順

1. クラスターの各ノードに Red Hat JBoss EAP 7.3 をインストールします。

- a. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product: Enterprise Application Platform**
  - **Version: 7.3**
- b. Red Hat JBoss Enterprise Application Platform 7.3.0の横にある **Download** をクリックします。 (**jboss-eap-7.3.0/jboss-eap-7.3.0.zip**)。
2. **jboss-eap-7.3.0.zip** ファイルを展開します。以下の例では、**EAP\_HOME** を **jboss-eap-7.3/jboss-eap-7.3** ディレクトリーとします。
3. 最新の Red Hat JBoss EAP パッチが利用できる場合には、ダウンロードして適用します。
4. クラスターの各ノードで Red Hat Decision Manager をダウンロードします。
  - a. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し、ドロップダウンオプションから製品およびバージョンを選択します。
    - **Product: Decision Manager**
    - **Version: 7.10**
  - b. Red Hat Decision Manager 7.10.0 Business Central Deployable for Red Hat JBoss EAP 7 (**rhdm-7.10.0-decision-central-eap7-deployable.zip**) をダウンロードします。
5. **rhdm-7.10.0-decision-central-eap7-deployable.zip** ファイルを一時ディレクトリーに展開します。以下のコマンドでは、このディレクトリーを **TEMP\_DIR** とします。
6. **TEMP\_DIR/rhdm-7.10.0-decision-central-eap7-deployable/jboss-eap-7.3** の内容を **EAP\_HOME** にコピーします。
7. 最新の Red Hat Decision Manager のパッチが利用できる場合には、ダウンロードして適用します。
8. 高可用性環境向けに Red Hat Single Sign-On を設定します。詳細は、[Red Hat Decision Manager と Red Hat Single Sign-On の統合](#) および [Red Hat Single Sign-On サーバー管理ガイド](#) を参照してください。

## 28.5. クラスターでの BUSINESS CENTRAL の設定および実行

Red Hat JBoss EAP と Business Central のインストール後に、Red Hat Data Grid と AMQ Broker を使用してクラスターを設定できます。クラスターの各ノードで以下の手順を実行します。



### 注記

この手順では、基本的なクラスター設定を説明します。より詳細な設定は [Red Hat JBoss EAP 7.3 設定ガイド](#) を参照してください。



## 注記

高可用性 (HA) のオンプレミス環境で KIE Server を Business Central に接続しないでください。

Business Central インスタンスは、各 KIE Server のステータスと同期し続けることができません。たとえば、KIE Server が稼働中だが同期していない場合、Business Central はそのインスタンスからデプロイできなくなります。

## 前提条件

- 「[Red Hat Data Grid のインストールおよび設定](#)」の記載どおりに、Red Hat Data Grid 7.3 がインストールされている。
- 「[AMQ Broker のダウンロードおよび設定](#)」の説明通りに AMQ Broker がインストールされ、設定されている。
- 「[Red Hat JBoss EAP 7.3 および Red Hat Decision Manager のダウンロードおよび展開](#)」の説明通りに、クラスターの各ノードに Red Hat JBoss EAP および Red Hat Decision Manager がインストールされている。
- 「[NFS バージョン 4 サーバーの設定](#)」で記載されているように、共有ディレクトリーを備えた NFS サーバーを利用できる。

## 手順

1. NFS で共有されているディレクトリーを **/data** としてマウントします。Root ユーザーで以下のコマンドを入力します。

```
mkdir /data
mount <NFS_SERVER_IP>:<DATA_SHARE> /data
```

**<NFS\_SERVER\_IP>** は、NFS サーバーシステムの IP アドレスまたはホスト名に置き換えてください。**<DATA\_SHARE>** は、設定した共有名 (例: **/opt/kie/data**) に置き換えます。

2. **/data** NFS ディレクトリーに **kie-wb-playground** ディレクトリーを作成します。

```
mkdir /kie-wb-playground
```

3. **EAP\_HOME/bin** ディレクトリーに **kie-wb-playground** ディレクトリーを作成し、ディレクトリーをマウントします。

```
mount -o rw,sync,actimeo=1 <NFS_SERVER_IP>:<DATA_SHARE>/kie-wb-playground kie-wb-playground
```

4. テキストエディターで **EAP\_HOME/standalone/configuration/standalone-full.xml** ファイルを開きます。

5. **<system-properties>** 要素でプロパティを編集または追加し、以下のプレースホルターを置き換えます。

- **<AMQ\_USER>** および **<AMQ\_PASSWORD>** は、AMQ Broker の作成時に定義した認証情報に置き換えます。
- **<AMQ\_BROKER\_IP\_ADDRESS>** は AMQ Broker の IP アドレスに置き換えます。

- **<DATA\_GRID\_NODE\_IP>** は、Red Hat Data Grid のインストール先の IP アドレスに置き換えます。

```
<system-properties>
  <property name="appformer-jms-connection-mode" value="REMOTE"/>
  <property name="appformer-jms-username" value="<AMQ_USER>"/>
  <property name="appformer-jms-password" value="<AMQ_USER_PASSWORD>"/>
  <property name="appformer-jms-url"
    value="tcp://<AMQ_BROKER_IP_ADDRESS>:61616?
ha=true&retryInterval=1000&retryIntervalMultiplier=1.0&reconnectAttempts=
-1"/>
  <property name="org.appformer.ext.metadata.infinispan.port"
    value="11222"/>
  <property name="org.appformer.ext.metadata.infinispan.host"
    value="<DATA_GRID_NODE_IP>"/>
  <property name="org.appformer.ext.metadata.infinispan.realm"
    value="ApplicationRealm"/>
  <property name="org.appformer.ext.metadata.infinispan.cluster"
    value="kie-cluster"/>
  <property name="org.appformer.ext.metadata.index"
    value="infinispan"/>
  <property name="org.uberfire.nio.git.dir"
    value="/data"/>
  <property name="es.set.netty.runtime.available.processors"
    value="false"/>
</system-properties>
```

6. オプション: Red Hat Data Grid のデプロイメントで認証が必要な場合は、**<system-properties>** 要素でプロパティを編集または追加し、以下のプレースホルターを置き換えます。

- **<SERVER\_NAME>** は、Red Hat Data Grid のサーバー設定で指定したサーバー名に置き換えます。
- **<SASL\_QOP>** は、Red Hat Data Grid サーバー設定の auth、auth-int、および auth-conf の値に置き換えます。

```
<property name="org.appformer.ext.metadata.infinispan.server.name"
  value="<SERVER_NAME>"/>
<property name="org.appformer.ext.metadata.infinispan.sasl.qop"
  value="<SASL_QOP>"/>
<property name="org.appformer.ext.metadata.infinispan.username"
  value=""/>
<property name="org.appformer.ext.metadata.infinispan.password"
  value=""/>
```

7. **standalone-full.xml** ファイルを保存します。
8. クラスターを起動するには **EAP\_HOME/bin** に移動して、以下のコマンドを入力します。

```
$ ./standalone.sh -c standalone-full.xml -b <HOST>
```

**<HOST>** は、Red Hat Decision Manager をインストールしたサーバーの IP アドレスまたはホスト名に置き換えます。

## 28.6. 高可用性 (HA) オンプレミスインフラストラクチャーのテスト

Business Central の実稼働環境可能な高可用性 (HA) オンプレミスインフラストラクチャーを作成する場合は、実行可能な HA 環境のハードウェアおよびパフォーマンスの最低要件を満たしていることを確認する必要があります。HA オンプレミスインフラストラクチャーは、Business Central、メッセージシステム (AMQ)、インデックスサーバー (Red Hat Data Grid)、および共有ファイルシステム (NFS/GlusterFS/Ceph) の 4 つの主要コンポーネントで設定されます。

### 前提条件

- 3 つ以上のノードのネットワーク環境は、以下のレイアウトで設定される。

ノード 1: Business Central

ノード 2: Business Central

ノード 3: AMQ、Red Hat Data Grid、および NFS

### 手順

1. ネットワーク速度をテストします。

- a. 各サーバーノードのコマンド端末で、**iPerf3** をインストールします。

```
$ dnf install iperf3
```

- b. NFS サーバーノード (サーバーノード 3) のコマンドターミナルで、**iPerf3** をサーバーモードで起動します。

```
$ iperf3 -s
```

- c. 各 Business Central サーバーノードのコマンド端末で以下のコマンドを入力して、ホストとして NFS サーバーノードを設定して、クライアントモードで **iPerf3** を起動します。

```
$ iperf3 -c <NFS_SERVER_IP>
```

+

In this example, replace ``<NFS_SERVER_IP>`` with the IP address of the NFS server.

- d. 各サーバーノードの結果を以下の最小値の例と比較します。

```
iperf3 -c 172.31.47.103
Connecting to host 172.31.47.103, port 5201
[ 5] local 172.31.39.4 port 44820 connected to 172.31.47.103 port 5201
[ ID] Interval      Transfer  Bitrate   Retr  Cwnd
[ 5] 0.00-1.00 sec  143 MBytes 1.20 Gbits/sec  0  419 KBytes
[ 5] 1.00-2.00 sec  111 MBytes 928 Mbits/sec  6  848 KBytes
[ 5] 2.00-3.00 sec  53.8 MBytes 451 Mbits/sec  0  1.08 MBytes
[ 5] 3.00-4.00 sec  52.5 MBytes 440 Mbits/sec  1  1022 KBytes
[ 5] 4.00-5.00 sec  53.8 MBytes 451 Mbits/sec  1  935 KBytes
[ 5] 5.00-6.00 sec  53.8 MBytes 451 Mbits/sec  1  848 KBytes
[ 5] 6.00-7.00 sec  52.5 MBytes 440 Mbits/sec  0  1.08 MBytes
[ 5] 7.00-8.00 sec  53.8 MBytes 451 Mbits/sec  1  1.01 MBytes
[ 5] 8.00-9.00 sec  53.8 MBytes 451 Mbits/sec  1  953 KBytes
[ 5] 9.00-10.00 sec 52.5 MBytes 440 Mbits/sec  1  856 KBytes
-----
```

```
[ ID] Interval      Transfer  Bitrate    Retr
[ 5] 0.00-10.00 sec 680 MBytes 570 Mbits/sec 12      sender
[ 5] 0.00-10.04 sec 677 MBytes 566 Mbits/sec      receiver
```

```
iperf Done.
```

## 2. NFS 情報を確認します。

- a. 各 Business Central サーバーノードのコマンド端末で、NFS ノードをマウントします。

```
$ mount -o actimeo=1 <NFS_SERVER_IP>:/opt/nfs/kie /opt/kie/niogit
```

- b. マウントされた各ノードのコマンド端末で、**nfsiostat** と入力します。

```
$ nfsiostat
```

- c. 各サーバーノードの結果を以下の最小値の例と比較します。

```
nfsiostat
ops/s rpc bklog
6.415 0.000

read:
ops/s kB/s kB/op retrans avg RTT (ms) avg exe (ms) avg queue (ms) errors
0.031 0.045 1.452 0 (0.0%) 0.129 0.166 0.019 0 (0.0%)

write:
ops/s kB/s kB/op retrans avg RTT (ms) avg exe (ms) avg queue (ms) errors
0.517 0.467 0.903 0 (0.0%) 1.235 1.269 0.018 0 (0.0%)
```

## 3. ディスクが SSD であることを確認します。

- a. NFS サーバーのコマンド端末で、**df -h** を入力して、以下の例で示すようにディスクを特定します。

```
$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        3.8G   0 3.8G   0% /dev
tmpfs           3.9G   0 3.9G   0% /dev/shm
tmpfs           3.9G  33M 3.8G   1% /run
tmpfs           3.9G   0 3.9G   0% /sys/fs/cgroup
/dev/xvda2      25G  3.2G 22G  13% /
tmpfs           781M   0 781M   0% /run/user/1000
172.31.47.103:/root/nfs 25G  2.1G 23G   9% /root/nfs
```

- b. **lsblk -d** を入力して、ディスクが SSD であることを確認します。

```
$ lsblk -d
```

- c. **hdparm -Tt** と入力して、ディスクをテストします。

```
$ hdparm -Tt /dev/xvda2
```

- d. 各サーバーノードの結果を以下の最小値の例と比較します。

```
$ hdparm -Tt /dev/xvda2
```

```
/dev/xvda2:
```

```
Timing cached reads: 18670 MB in 1.99 seconds = 9389.01 MB/sec
```

```
Timing buffered disk reads: 216 MB in 3.03 seconds = 71.40 MB/sec
```

## 28.7. RED HAT DECISION MANAGER クラスターの検証

Red Hat Decision Manager のクラスターを設定したら、アセットを作成してシステムが機能していることを検証します。

### 手順

1. Web ブラウザーで、**<node-IP-address>:8080/decision-central** を入力します。**<node-IP-address>** は特定のノードの IP アドレスに置き換えます。
2. インストール時に作成した **admin** ユーザーの認証情報を入力します。Business Central ホームページが表示されます。
3. **Menu → Design → Projects** の順に選択します。
4. **MySpace** スペースを開きます。
5. **Try Samples → Mortgages → OK** の順にクリックします。**Assets** ウィンドウが表示されます。
6. **Add Asset → Data Object** をクリックします。
7. **Data Object** フィールドに **MyDataObject** と入力し、**OK** をクリックします。
8. **Spaces → mySpace → Mortgages** の順にクリックし、アセットリストに **MyDataObject** があることを確認します。
9. Web ブラウザーに以下の URL を入力します。**<node\_IP\_address>** には、クラスターの別のノードのアドレスに置き換えます。  
**http://<node\_IP\_address>:8080/decision-central**
10. **MyDataObject** アセットを作成した最初のノードの Business Central にログインするときに使用した認証情報と同じものを入力します。
11. **Menu → Design → Projects** の順に選択します。
12. **MySpace** スペースを開きます。
13. **Mortgages** プロジェクトを選択します。
14. **MyDataObject** がアセットリストにあることを確認します。
15. **Mortgages** プロジェクトを選択します。

## 第29章 ランタイム環境における KIE SERVER クラスター

KIE Server ランタイム環境をクラスターリングする主な利点は負荷分散です。クラスターのノードの1つでアクティビティが増えると、そのアクティビティはクラスターの残りのノードと共有されるため、パフォーマンスが改善します。

KIE Server のクラスター化ランタイム環境を作成するには、Red Hat JBoss EAP 7.3 および KIE Server をダウンロードして展開します。次に、ドメインモードクラスターに Red Hat JBoss EAP 7.3 を設定し、クラスターを起動し、各クラスターノードに KIE Server をインストールします。

任意で、ヘッドレス Decision Manager コントロールパッドをクラスター化できます。



### 注記

高可用性 (HA) のオンプレミス環境で KIE Server を Business Central に接続しないでください。

Business Central インスタンスは、各 KIE Server のステータスと同期し続けることができません。たとえば、KIE Server が稼働中だが同期していない場合、Business Central はそのインスタンスからデプロイできなくなります。

### 29.1. RED HAT JBOSS EAP 7.3 および KIE SERVER のダウンロードおよび展開

本セクションの手順を行い、Red Hat JBoss EAP 7.3 をダウンロードしてインストールし、クラスター環境にインストールするために KIE Server をダウンロードして再パッケージ化します。

#### 手順

1. クラスターの各ノードに Red Hat JBoss EAP 7.3 をインストールします。
  - a. Red Hat カスタマーポータル[の Software Downloads ページ](#)に移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
    - **Product: Enterprise Application Platform**
    - **Version: 7.3**
  - b. Red Hat JBoss Enterprise Application Platform 7.3.0(**jboss-eap-7.3.0.zip**) の横にある **Download** をクリックします。
2. **jboss-eap-7.3.0.zip** ファイルを展開します。 **jboss-eap-7.3/jboss-eap-7.3** ディレクトリーは **EAP\_HOME** とします。
3. 最新の Red Hat JBoss EAP パッチが利用できる場合には、ダウンロードして適用します。
4. KIE Server のダウンロード:
  - a. Red Hat カスタマーポータル[の Software Downloads ページ](#)に移動し、ドロップダウンオプションから製品およびバージョンを選択します。
    - **Product: Decision Manager**
    - **Version: 7.10**



- b. Red Hat Decision Manager 7.10.0 KIE Server for All Supported EE8 Containers(`rhdm-7.10.0-kie-server-ee8.zip`) をダウンロードします。
5. `rhdm-7.10.0-kie-server-ee8.zip` アーカイブを一時ディレクトリーに展開します。以下の例では、この名前を `TEMP_DIR` とします。
6. `TEMP_DIR/rhdm-7.10.0-kie-server-ee8/rhdm-7.10.0-kie-server-ee8/kie-server.war` ディレクトリーを `EAP_HOME/standalone/deployments/` にコピーします。



### 警告

コピーする Red Hat Decision Manager デプロイメントの名前が Red Hat JBoss EAP インスタンスの既存デプロイメントと競合しないことを確認します。

7. `TEMP_DIR/rhdm-7.10.0-kie-server-ee8/rhdm-7.10.0-kie-server-ee8/SecurityPolicy/` のコンテンツを `EAP_HOME/bin` にコピーします。
8. ファイルを上書きするように求められたら、**Replace** をクリックします。
9. `EAP_HOME/standalone/deployments/` ディレクトリーに、`kie-server.war.dodeploy` という名前で空のファイルを作成します。このファイルにより、サーバーが起動すると KIE Server が自動的にデプロイされます。
10. 最新の Red Hat Decision Manager のパッチが利用できる場合には、ダウンロードして適用します。
11. `EAP_HOME/bin` ディレクトリーに移動します。
12. KIE Server へのログインに使用する `kie-server` ロールのユーザーを作成します。

```
$ ./add-user.sh -a --user <USERNAME> --password <PASSWORD> --role kie-server
```

13. ユーザー名とパスワードを書き留めておきます。
14. クラスターを起動するには `EAP_HOME/bin` に移動して、以下のコマンドの1つを入力します。

- Linux または UNIX ベースのシステムの場合:

```
$ ./standalone.sh -c standalone-full.xml
```

- Windows の場合:

```
standalone.bat -c standalone-full.xml
```

## 29.2. ヘッドレス DECISION MANAGER コントローラーを使用した KIE SERVER のクラスターリング

Decision Manager コントローラーは Business Central と統合します。ただし、Business Central をインストールしない場合は、ヘッドレス Decision Manager コントローラーをインストールし、REST API または KIE Server Java Client API を使用してそのコントローラーと対話します。

## 前提条件

- バックアップを作成してある Red Hat JBoss EAP システム (バージョン 7.3 またはそれ以降) が利用できる。Red Hat JBoss EAP システムのベースディレクトリーを **EAP\_HOME** とします。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。
- [「NFS バージョン 4 サーバーの設定」](#) で記載されているように、共有ディレクトリーを備えた NFS サーバーを利用できる。

## 手順

1. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product: Decision Manager**
  - **Version: 7.10**
2. Red Hat Decision Manager 7.10.0 Add Ons(**rhdm-7.10.0-add-ons.zip** ファイル) をダウンロードします。
3. **rhdm-7.10.0-add-ons.zip** ファイルを展開します。**rhdm-7.10.0-controller-ee7.zip** ファイルは展開したディレクトリーにあります。
4. **rhdm-7.10.0-controller-ee7** アーカイブを一時ディレクトリーに展開します。以下の例では、この名前を **TEMP\_DIR** とします。
5. **TEMP\_DIR/rhdm-7.10.0-controller-ee7/controller.war** ディレクトリーを **EAP\_HOME/standalone/deployments/** にコピーします。



### 警告

コピーするヘッドレス Decision Manager コントローラーデプロイメントの名前が、Red Hat JBoss EAP インスタンスの既存デプロイメントと競合しないことを確認します。

6. **TEMP\_DIR/rhdm-7.10.0-controller-ee7/SecurityPolicy/** ディレクトリーの中身を **EAP\_HOME/bin** にコピーします。
7. ファイルの上書きを求めるプロンプトが出されたら、**Yes** をクリックします。
8. **EAP\_HOME/standalone/deployments/** ディレクトリーに、**controller.war.dodeploy** という名前で空のファイルを作成します。このファイルにより、サーバーが起動するとヘッドレス Decision Manager コントローラーが自動的にデプロイされます。

9. テキストエディターで **EAP\_HOME/standalone/configuration/standalone.xml** ファイルを開きます。
10. 以下のプロパティを **<system-properties>** 要素に追加し、**<NFS\_STORAGE>** を、テンプレート設定が保存されている NFS ストレージへの絶対パスに置き換えます。

```
<system-properties>
  <property name="org.kie.server.controller.templatefile.watcher.enabled" value="true"/>
  <property name="org.kie.server.controller.templatefile" value="<NFS_STORAGE>"/>
</system-properties>
```

テンプレートファイルには、特定のデプロイメントシナリオのデフォルト設定が含まれます。

**org.kie.server.controller.templatefile.watcher.enabled** プロパティの値を **true** に設定すると、別のスレッドが開始してテンプレートファイルの修正を監視します。この確認の間隔はデフォルトで 30000 ミリ秒になり、**org.kie.server.controller.templatefile.watcher.interval** システムプロパティで制御できます。このプロパティの値を **false** に設定すると、テンプレートファイルへの変更の検出が、サーバーの再起動時に制限されます。

11. ヘッドレス Decision Manager コントローラーを開始するには、**EAP\_HOME/bin** に移動して、以下のコマンドを実行します。

- Linux または UNIX ベースのシステムの場合:

```
$ ./standalone.sh
```

- Windows の場合:

```
standalone.bat
```

## 第30章 関連資料

- [Red Hat JBoss EAP 7.3 への Red Hat Decision Manager のインストールおよび設定](#)
- [Red Hat Decision Manager インストールの計画](#)
- [『Operator を使用した Red Hat OpenShift Container Platform 4 への Red Hat Decision Manager 環境のデプロイメント』](#)
- [テンプレートを使用した Red Hat OpenShift Container Platform 3 への Red Hat Decision Manager 環境のデプロイメント](#)

## パート IV. RED HAT JBOSS WEB SERVER への RED HAT DECISION MANAGER のインストールおよび設定

本書は、JBoss Web Server に Red Hat Decision Manager 7.10 をインストールする方法を説明します。

### 前提条件

- [Red Hat Decision Manager インストールの計画](#)の内容を確認している。
- Red Hat JBoss Web Server 5.4.0 がインストールされている。Red Hat JBoss Web Server のインストールに関する情報は、[Red Hat JBoss Web Server インストールガイド](#)を参照してください。

## 第31章 RED HAT DECISION MANAGER について

Red Hat Decision Manager は、ビジネスルール管理、複合イベント処理、Decision Model & Notation (DMN) 実行、およびプランニングの問題を解決するための Business Optimizer を組み合わせた、オープンソースの意思決定管理プラットフォームです。これにより、ビジネス上の意思決定を自動化し、そのロジックをビジネス全体で利用できるようにします。

ルール、デシジョンテーブル、DMN モデルなどのビジネスアセットはプロジェクト内で組織化され、Business Central リポジトリに保存されます。これにより、ビジネス全体で一貫性や透明性を維持し、監査を行えます。ビジネスユーザーは、IT 担当者からのサポートなしでビジネスロジックを編集できます。

Red Hat Process Automation Manager は、Business Central および KIE Server で設定されます。

- KIE Server では、ルール、およびその他のアーティファクトが実行されます。これは、ルールをインスタンス化して実行し、計画の問題を解決するために使用されます。KIE Server はビジネスアセットのランタイム環境を提供し、アセットリポジトリ (ナレッジストア) に保存されたデータにアクセスします。
- Business Central は、KIE Server が実行するビジネスルールを作成して管理するグラフィカルユーザーインターフェイスです。

Business Central は、スタンドアロンの JAR ファイルとしても使用できます。Business Central スタンドアロンの JAR ファイルとして使用して、アプリケーションサーバーにデプロイせずに Business Central を実行できます。

Red Hat JBoss Web Server は Tomcat をベースとしたエンタープライズレベルの Web サーバーで、中規模および大規模のアプリケーション用に設計されています。Red Hat JBoss Web Server は、Java Server Pages (JSP) および Java Servlet テクノロジー、PHP、ならびに CGI をデプロイするための単一プラットフォームを提供します。

Red Hat JBoss Web Server インストールに KIE Server およびヘッドレス Decision Manager コントローラーをインストールできます。または、スタンドアロンの Business Central JAR ファイルを実行できます。

本書では、Red Hat JBoss Web Server インスタンスに Red Hat Decision Manager をインストールする方法を説明します。

他の環境への Red Hat Decision Manager のインストール方法に関する説明は、以下のドキュメントを参照してください。

- [Red Hat JBoss EAP 7.3 への Red Hat Decision Manager のインストールおよび設定](#)
- [IBM WebSphere Application Server への KIE Server のインストールおよび設定](#)
- [Oracle WebLogic Server への KIE Server のインストールおよび設定](#)
- [『Operator を使用した Red Hat OpenShift Container Platform 4 への Red Hat Decision Manager 環境のデプロイメント』](#)
- [テンプレートを使用した Red Hat OpenShift Container Platform 3 への Red Hat Decision Manager 環境のデプロイメント](#)

サポートされるコンポーネントについては、以下のドキュメントを参照してください。

- [What is the mapping between Red Hat Decision Manager and the Maven library version?](#)

- [Red Hat Decision Manager 7 でサポート対象の設定](#)

## 31.1. RED HAT DECISION MANAGER のコンポーネント

Red Hat Decision Manager は、Business Central と KIE Server で設定されます。

- Business Central は、ビジネスルールを作成して管理するグラフィカルユーザーインターフェイスです。Business Central は、Red Hat JBoss EAP インスタンスまたは Red Hat OpenShift Container Platform (OpenShift) にインストールできます。  
Business Central は、スタンドアロンの JAR ファイルとしても使用できます。Business Central スタンドアロンの JAR ファイルとして使用して、アプリケーションサーバーにデプロイせずに Business Central を実行できます。
- KIE Server では、ルール、およびその他のアーティファクトが実行されます。これは、ルールをインスタンス化して実行し、計画の問題を解決するために使用されます。KIE Server は、Red Hat JBoss EAP インスタンス、Red Hat JBoss EAP クラスター、OpenShift、Oracle WebLogic Server インスタンス、IBM WebSphere Application Server インスタンスに、または Spring Boot アプリケーションの一部としてインストールできます。  
KIE Server は、管理モードまたは非管理モードで動作するように設定できます。KIE Server が非管理モードの場合は、手動で KIE コンテナ (デプロイメントユニット) を作成および維持する必要があります。KIE コンテナは、プロジェクトの特定のバージョンです。管理モードの場合は、Decision Manager コントローラーが KIE Server の設定を管理し、ユーザーはコントローラーと対話形式で KIE コンテナを作成して維持します。

Red Hat JBoss Web Server インストールに KIE Server およびヘッドレス Decision Manager コントローラーをインストールできます。または、スタンドアロンの Business Central JAR ファイルを実行できます。

## 31.2. RED HAT DECISION MANAGER ロールおよびユーザー

Business Central または KIE Server にアクセスするには、サーバーを起動する前にユーザーを作成して適切なロールを割り当てます。Business Central または KIE Server のインストール時に、ユーザーとロールを作成できます。

Business Central と KIE Server は、JAVA 認証承認サービス (JAAS) ログインモジュールを使用してユーザーを認証します。Business Central と KIE Server の両方が単一のインスタンスで実行されている場合は、同じ JAAS サブジェクトとセキュリティドメインを共有します。したがって、Business Central に対して認証されたユーザーは、KIE Server にもアクセスできます。

ただし、Business Central と KIE Server が異なるインスタンスで実行されている場合、JAAS ログインモジュールは両方に対して個別にトリガーされます。したがって、Business Central に対して認証されたユーザーは、KIE Server にアクセスするために別々に認証する必要があります。たとえば、Business Central で認証されているものの、KIE Server で認証されていないユーザーが Business Central でプロセステータスを表示または管理しようとすると、401 エラーがログファイルに記録され、**Invalid credentials to load data from remote server.Contact your system administrator.** メッセージが Business Central に表示されます。

本セクションでは、Red Hat Decision Manager のユーザーロールを説明します。



### 注記

**admin**、**analyst**、および **rest-all** のロールは Business Central 用に予約されています。**kie-server** ロールは KIE Server 用に予約されています。このため、Business Central または KIE Server のいずれか、またはそれら両方がインストールされているかどうかによって、利用可能なロールは異なります。

- **admin: admin** ロールを持つユーザーは Business Central 管理者です。管理者は、ユーザーの管理や、リポジトリの作成、クローン作成、および管理ができます。アプリケーションで必要な変更をすべて利用できます。**admin** ロールを持つユーザーは、Red Hat Decision Manager の全領域にアクセスできます。
- **analyst: analyst** ロールを持つユーザーには、すべてのハイレベル機能へのアクセスがあります。プロジェクトのモデル化が可能です。ただし、このユーザーは、**Design → Projects** ビューでスペースに貢献者を追加したり、スペースを削除したりできません。**analyst** ロールを持つユーザーは、管理者向けの **Deploy → Execution Servers** ビューにアクセスできません。ただし、これらのユーザーは、ライブラリーパースペクティブにアクセスするときに **Deploy** ボタンを使用できます。
- **rest-all: rest-all** ロールを持つユーザーは、Business Central REST 機能にアクセスできます。
- **kie-server: kie-server** ロールを持つユーザーは、KIE Server REST 機能にアクセスできます。



## 第32章 RED HAT DECISION MANAGER インストールファイルのダウンロード

インストーラー JAR ファイルまたはデプロイ可能な ZIP ファイルを使用して Red Hat Decision Manager をインストールすることができます。インストーラーは、インタラクティブモードまたはコマンドラインインターフェイス (CLI) モードで実行できます。もしくは、Business Central および KIE Server のデプロイ可能な ZIP ファイルを展開して設定できます。Business Central をアプリケーションサーバーにデプロイせずに実行する場合は、Business Central スタンドアロン JAR ファイルをダウンロードします。

お使いの環境およびインストール要件に応じた Red Hat Decision Manager ディストリビューションをダウンロードします。

### 手順

1. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product:** Decision Manager
  - **バージョン:** 7.10
2. お好みのインストール方法に従って、以下の製品ディストリビューションのいずれかをダウンロードします。



### 注記

ダウンロードするのは、どれか1つのディストリビューションのみです。

- インストーラーを使用して Red Hat JBoss Web Server に Red Hat Decision Manager をインストールする場合は、**Red Hat Decision Manager 7.10.0 Installer(rhdm-installer-7.10.0.jar)** をダウンロードします。インストーラーのグラフィックユーザーインターフェイスに従って、インストールプロセスを進めます。
- デプロイ可能な ZIP ファイルを使用して Red Hat JBoss Web Server 上に KIE Server をインストールするには、**Red Hat Decision Manager 7.10.0 Add Ons(rhdm-7.10.0-add-ons.zip)** ファイルをダウンロードします。  
ZIP ファイルには、グラフィカルユーザーインターフェイスは必要ありません。
- アプリケーションサーバーにデプロイせずに Business Central を実行するには、**Red Hat Decision Manager 7.10.0 Business Central Standalone (rhdm-7.10.0-decision-central-standalone.jar)** をダウンロードします。

## 第33章 RED HAT DECISION MANAGER インストーラーの使用方 法

本セクションでは、インストーラー JAR ファイルを使用して Business Central および KIE Server をインストールする方法を説明します。JAR ファイルは、既存の Red Hat JBoss Web Server 5.4.0 サーバーインストール環境に Red Hat Decision Manager をインストールする実行ファイルです。インストーラーは、インタラクティブモードまたはコマンドラインインターフェイス (CLI) モードで実行できます。

### 次のステップ:

以下のセクションのいずれかに記載される手順を行います。

- [「インタラクティブモードでのインストーラーの使用」](#)
- [「CLI モードでのインストーラーの使用」](#)

### 33.1. インタラクティブモードでのインストーラーの使用

Red Hat Decision Manager のインストーラーは、実行可能な JAR ファイルです。このインストーラーを使用して、既存の Red Hat JBoss Web Server 5.4.0 のサーバーインストールに Red Hat Decision Manager をインストールできます。

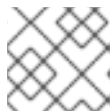


#### 注記

セキュリティ上の理由で、root 以外のユーザーでインストーラーを実行する必要があります。

#### 前提条件

- Red Hat Decision Manager 7.10.0 インストーラーをダウンロードしている。手順は、[32章Red Hat Decision Manager インストールファイルのダウンロード](#)を参照してください。
- サポート対象の JDK がインストールされている。サポート対象の JDK に関する一覧は、[Red Hat Decision Manager 7 Supported Configurations](#) を参照してください。
- バックアップを作成してある Red Hat JBoss Web Server 5.4.0 のサーバーインストールが利用できる。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。



#### 注記

Tomcat への書き込み権限のあるユーザーとしてログインします。

- `$PATH` 環境変数に含まれている JAR バイナリー。Red Hat Enterprise Linux では、`java-$JAVA_VERSION-openjdk-devel` パッケージに含まれています。



## 注記

Red Hat Decision Manager は、UTF-8 エンコーディングで機能するように設計されています。基礎となる JVM で別のエンコーディングシステムを使用すると、予期せぬエラーが発生する可能性があります。JVM で UTF-8 を使用するようには、**-Dfile.encoding=UTF-8** のシステムプロパティを使用します。システムプロパティの一覧は、[付録B Business Central システムプロパティ](#)を参照してください。

## 手順

1. 端末ウィンドウで、インストーラー JAR ファイルをダウンロードしたディレクトリーに移動し、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.10.0.jar
```



## 注記

Windows でインストーラーを実行すると、インストール時に管理者の認証情報が求められる場合があります。この要求を回避するには、インストールコマンドに **izpack.mode=privileged** オプションを追加します。

```
java -Dizpack.mode=privileged -jar  
rhdm-installer-7.10.0.jar
```

また、32 ビットの Java 仮想マシンでインストーラーを実行している場合には、メモリ不足になる可能性があります。この問題を防ぐには、以下のコマンドを実行します。

```
java -XX:MaxHeapSize=4g -jar  
rhdm-installer-7.10.0.jar
```

グラフィカルインストーラーにスプラッシュ画面と使用許諾契約書のページが表示されます。

2. **I accept the terms of this license agreement (本使用許諾契約書の内容に同意します)** をクリックし、**Next** をクリックします。
3. Red Hat Decision Manager をインストールする Red Hat JBoss Web Server 5.4.0 サーバーのホームを指定して、**Next** をクリックします。
4. インストールするコンポーネントを選択し、**Next** をクリックします。  
Business Central は Red Hat JBoss Web Server にインストールすることができません。Business Central をインストールできるのは Red Hat JBoss EAP だけです。KIE Server およびヘッドレス Decision Manager コントローラーを Red Hat JBoss Web Server にインストールできます。ヘッドレス Decision Manager コントローラーは、KIE Server の管理に使用されます。複数の KIE Server インスタンスを管理する予定がある場合は、ヘッドレス Decision Manager コントローラーをインストールします。
5. ユーザーを作成して **Next** をクリックします。デフォルトでは、同じコンテナに Business Central と KIE Server の両方をインストールする場合は、新しいユーザーに **admin** ロール、**kie-server** ロール、および **rest-all** ロールが割り当てられます。KIE Server のみをインストールした場合には、ユーザーには **kie-server** ロールが割り当てられます。KIE Server REST 機能にアクセスするには **kie-server** ロールが必要です。



### 注記

必ず、既存のユーザー、ロール、またはグループとは異なるユーザー名を指定してください。たとえば、**admin** という名前のユーザーは作成しないでください。

パスワードは 8 文字以上で、数字と、英数字以外の文字をそれぞれ 1 文字以上使用する必要があります。ただし & の文字は使用できません。

ユーザー名とパスワードを書き留めておきます。Business Central および KIE Server にアクセスする時に必要になります。

6. **Installation Overview** ページで、インストールするコンポーネントを確認し、**Next** をクリックしてインストールを開始します。
7. インストールが完了したら、**Next** をクリックします。
8. 画面上部に **Processing finished** が表示されたら、**Next** をクリックしてインストールを完了します。
9. 必要に応じて、**Generate Installation Script and Properties File** をクリックして、XML ファイルでインストールデータを保存し、**Done** をクリックします。  
インストーラーは、2 つのファイルを生成します。**auto.xml** ファイルは、今後のインストールを自動化し、**auto.xml.variables** ファイルは、ユーザーのパスワードと他の機密情報に関連する変数を保存します。**auto.xml** ファイルを使用して、元のインストールと同じタイプのサーバーおよび同じ設定の複数のシステムで Red Hat Decision Manager のインストールを繰り返します。必要に応じて、**auto.xml** ファイルの **installpath** パラメーターを更新します。XML ファイルを使用してインストールを実行するには、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.10.0.jar <path-to-auto.xml-file>
```

インストーラーを使用して Red Hat Decision Manager が正常にインストールされました。

## 33.2. CLI モードでのインストーラーの使用

コマンドラインインターフェイス (CLI) を使用して Red Hat Decision Manager インストーラーを実行できます。



### 注記

セキュリティ上の理由で、root 以外のユーザーでインストーラーを実行する必要があります。

### 前提条件

- Red Hat Decision Manager 7.10.0 インストーラーをダウンロードしている。手順は、[32章 Red Hat Decision Manager インストールファイルのダウンロード](#)を参照してください。
- サポート対象の JDK がインストールされている。サポート対象の JDK に関する一覧は、[Red Hat Decision Manager 7 Supported Configurations](#) を参照してください。
- バックアップを作成してある Red Hat JBoss Web Server 5.4.0 のサーバーインストールが利用できる。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。



### 注記

Tomcat への書き込み権限のあるユーザーとしてログインします。

- **\$PATH** 環境変数に含まれている JAR バイナリー。Red Hat Enterprise Linux では、**java-\$JAVA\_VERSION-openjdk-devel** パッケージに含まれています。



### 注記

Red Hat Decision Manager は、UTF-8 エンコーディングで機能するように設計されています。基礎となる JVM で別のエンコーディングシステムを使用すると、予期せぬエラーが発生する可能性があります。JVM で UTF-8 を使用するようには、**-Dfile.encoding=UTF-8** のシステムプロパティを使用します。システムプロパティの一覧は、[付録B Business Central システムプロパティ](#)を参照してください。

## 手順

1. 端末ウィンドウにおいて、インストーラーファイルをダウンロードしたディレクトリーに移動し、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.10.0.jar -console
```

コマンドラインの対話プロセスが開始し、使用許諾契約書が表示されます。

```
press 1 to continue, 2 to quit, 3 to redisplay.
```

2. 使用許諾契約書を読んで **1** を入力し、**Enter** キーを押して続行します。

```
Specify the home directory of one of the following servers: Red Hat JBoss EAP 7 or Red Hat
JBoss Web Server 5. For more information, see
https://access.redhat.com/articles/3354301[Red Hat Decision Manager 7 Supported
Configurations].
```

3. 既存の Red Hat JBoss Web Server 5.4.0 インストールの親ディレクトリーを入力します。インストーラーが、指定したインストール場所を確認します。**1** を入力して確認し、続行します。
4. インストーラーの指示に従って、インストールを完了します。



### 注記

ユーザー名およびパスワードを作成する場合は、指定したユーザー名が既存のロールまたはグループの名前と競合しないようにしてください。たとえば、**admin** というロールがある場合は、**admin** という名前のユーザーは作成しないでください。

パスワードは 8 文字以上で、数字と、英数字以外の文字をそれぞれ 1 文字以上使用する必要があります。ただし **&** の文字を使用することは **できません**。

ユーザー名とパスワードを書き留めておきます。Business Central および KIE Server にアクセスする時に必要になります。

5. インストールが完了すると、以下のメッセージが表示されます。

```
Would you like to generate an automatic installation script and properties file?
```

6. **y** を入力してインストールデータが含まれる XML ファイルを作成します。あるいは、**n** を入力してインストールを完了します。**y** を入力すると、XML ファイルのパスを指定するように求められます。
7. パスを入力するか、Enter キーを押して提案されたパスを了承します。  
インストーラーは、2つのファイルを生成します。**auto.xml** ファイルは、今後のインストールを自動化し、**auto.xml.variables** ファイルは、ユーザーのパスワードと他の機密情報に関連する変数を保存します。複数のシステムで **auto.xml** ファイルを使用して、元のインストールと同じ設定の同じタイプのサーバーに、Red Hat Process Automation Manager を簡単に繰り返しインストールできます。必要に応じて、**auto.xml** ファイルの **installpath** パラメーターを更新します。XML ファイルを使用してインストールを実行するには、以下のコマンドを入力します。

```
java -jar rhdm-installer-7.10.0.jar <path-to-auto.xml-file>
```

8. KIE Server だけをインストールした場合は、この手順を繰り返して、別のサーバーにヘッドレス Decision Manager コントローラーをインストールします。

## 第34章 KIE SERVER ZIP ファイルのインストールと設定

カスタマーポータルでの Red Hat Decision Manager 7.10.0 Add Ons(**rhdm-7.10.0-add-ons.zip**) ファイルで利用可能な **rhdm-7.10.0-kie-server-jws.zip** ファイルを使用して KIE Server をインストールできます。

### 34.1. ZIP ファイルからの KIE SERVER のインストール

KIE Server はビジネスアセットのランタイム環境を提供し、アセットリポジトリ (ナレッジストア) に保存されたデータにアクセスします。ZIP ファイルを使用して既存の Red Hat JBoss Web Server 5.4.0 サーバーインスタンスに KIE Server をインストールすることができます。



#### 注記

インストーラー JAR ファイルを使用した KIE Server のインストール方法は、[33章 Red Hat Decision Manager インストーラーの使用方法](#)を参照してください。

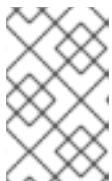
- [32章 Red Hat Decision Manager インストールファイルのダウンロード](#) に従って Red Hat Decision Manager 7.10.0 Add Ons (**rhdm-7.10.0-add-ons.zip**) ファイルがダウンロードされている。
- バックアップを作成してある Red Hat JBoss Web Server 5.4.0 のサーバーインストールが利用できる。Red Hat JBoss Web Server インストールのベースディレクトリーは **JWS\_HOME** とします。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。

#### 手順

1. **rhdm-7.10.0-add-ons.zip** ファイルを展開します。
2. 展開した **rhdm-7.10.0-add-ons.zip** ファイルから、以下のファイルを展開します。
  - **rhdm-7.10.0-kie-server-jws.zip**
  - **rhdm-7.10.0-decision-engine.zip**

以下の説明では、展開した **rhdm-7.10.0-kie-server-jws.zip** ファイルが含まれるディレクトリーは **JWS\_TEMP\_DIR** となり、展開した **rhdm-7.10.0-decision-engine.zip** ファイルが含まれるディレクトリーは **ENGINE\_TEMP\_DIR** となります。

3. **JWS\_TEMP\_DIR/rhdm-7.10.0-kie-server-jws/kie-server.war** ディレクトリーを **JWS\_HOME/tomcat/webapps** ディレクトリーにコピーします。



#### 注記

コピーする Red Hat Decision Manager デプロイメントの名前が Red Hat JBoss Web Server インスタンスの既存デプロイメントと競合しないことを確認します。

4. **kie-server.war** ディレクトリーから **.war** 拡張子を削除します。
5. **kie-tomcat-integration-7.48.0.Final-redhat-00004.jar** ファイルを **ENGINE\_TEMP\_DIR** ディレクトリーから **JWS\_HOME/tomcat/lib** ディレクトリーに移動します。



6. **jboss-jacc-api-<VERSION>.jar** ファイル、**slf4j-api-<VERSION>.jar** ファイル、および **slf4j-jdk14-<VERSION>.jar** ファイルを **ENGINE\_TEMP\_DIR/lib** ディレクトリーから **JWS\_HOME/tomcat/lib** ディレクトリーに移動します。**<VERSION>** は **lib** ディレクトリーのバージョンのアーティファクトファイル名に置き換えます。
7. 最後の Valve 定義の後にある **JWS\_HOME/tomcat/conf/server.xml** ファイルの **<host>** 要素に、以下の行を追加します。

```
<Valve className="org.kie.integration.tomcat.JACCValve" />
```

8. テキストエディターで **JWS\_HOME/tomcat/conf/tomcat-users.xml** ファイルを開きます。
9. **JWS\_HOME/tomcat/conf/tomcat-users.xml** ファイルにユーザーおよびロールを追加します。以下の例で、**<ROLE\_NAME>** は Red Hat Decision Manager でサポートされるロールです。**<USER\_NAME>** および **<USER\_PWD>** は、選択したユーザー名とパスワードです。

```
<role rolename="<ROLE_NAME>"/>
<user username="<USER_NAME>" password="<USER_PWD>" roles="<ROLE_NAME>"/>
```

ユーザーに複数のロールを割り当てる場合は、以下の例に示すようにロールをコンマで区切ります。

```
<role rolename="admin"/>
<role rolename="kie-server"/>
<user username="rhdmUser" password="user1234" roles="admin,kie-server"/>
```

10. **JWS\_HOME/tomcat/bin** ディレクトリーで以下の手順の1つを実行します。
  - Linux または UNIX の場合は、以下の内容の **setenv.sh** ファイルを作成します。

```
CATALINA_OPTS="-Xmx1024m
-Dorg.jboss.logging.provider=jdk
-Dorg.jbpm.server.ext.disabled=true
-Dorg.jbpm.ui.server.ext.disabled=true
-Dorg.jbpm.case.server.ext.disabled=true"
```

- Windows の場合は、以下の内容の **setenv.bat** ファイルを作成します。

```
set CATALINA_OPTS="-Xmx1024m -Dorg.jboss.logging.provider=jdk -
Dorg.jbpm.server.ext.disabled=true -Dorg.jbpm.ui.server.ext.disabled=true -
Dorg.jbpm.case.server.ext.disabled=true"
```



## 第35章 キーストアを使用したパスワードセキュリティの確保

キーストアを使用して、Business Central と KIE Server の間の通信に使用するパスワードを暗号化できます。コントローラーと KIE Server のパスワードを暗号化する必要があります。Business Central と KIE Server を別のアプリケーションサーバーにデプロイする場合は、いずれのアプリケーションサーバーもキーストアを使用する必要があります。

Java Cryptography Extension KeyStore (JCEKS) は、対称鍵をサポートするため、キーストアには JCEKS を使用してください。JDK インストールに含まれる KeyTool を使用して、新しい JCEKS を作成します。



### 注記

KIE Server が JCEKS で設定されていない場合、KIE Server のパスワードはシステムプロパティにプレーンテキスト形式で保存されます。

### 前提条件

- Red Hat JBoss Web Server に KIE Server がインストールされている。
- Java 8 以降がインストールされている。

### 手順

1. テキストエディターで **JWS\_HOME/tomcat/conf/tomcat-users.xml** ファイルを開きます。
2. **kie-server** ロールを割り当てた KIE Server ユーザーを **JWS\_HOME/tomcat/conf/tomcat-users.xml** ファイルに追加します。以下の例では、**<USER\_NAME>** と **<PASSWORD>** は任意のユーザー名とパスワードに置き換えます。

```
<role rolename="kie-server"/>
<user username="<USER_NAME>" password="<PASSWORD>" roles="kie-server"/>
```

3. KeyTool を使用して JCEKS を作成するには、Java 8 のホームディレクトリーで以下のコマンドを実行します。

```
$<JAVA_HOME>/bin/keytool -importpassword -keystore <KEYSTORE_PATH> -keypass
<ALIAS_KEY_PASSWORD> -alias <PASSWORD_ALIAS> -storepass
<KEYSTORE_PASSWORD> -storetype JCEKS
```

上記の例では、以下の変数を置き換えてください。

- **<KEYSTORE\_PATH>**: キーストアの保存先のパス
  - **<KEYSTORE\_PASSWORD>**: キーストアのパスワード
  - **<ALIAS\_KEY\_PASSWORD>**: エイリアスで保存した値にアクセスする時に使用するパスワード
  - **<PASSWORD\_ALIAS>**: プロセスへのエントリーに使用するエイリアス
4. プロンプトが表示されたら、作成した KIE Server ユーザーのパスワードを入力します。
  5. システムプロパティを設定するには、**JWS\_HOME/tomcat/bin** ディレクトリーで以下の手順の1つを実行し、変数は以下の表で記載されているように置き換えます。



## 注記

Business Central またはスタンドアロンのコントローラーが Red Hat JBoss Web Server とは別のインスタンスにインストールされている場合は、**kie.keystore.key.server.alias** と **kie.keystore.key.server.pwd** のプロパティを **CATALINA\_OPTS** に追加しないでください。

- Linux または UNIX の場合は、以下の内容の **setenv.sh** ファイルを作成します。

```
set CATALINA_OPTS="
-Dkie.keystore.keyStoreURL=<KEYSTORE_URL>
-Dkie.keystore.keyStorePwd=<KEYSTORE_PWD>
-Dkie.keystore.key.server.alias=<KEY_SERVER_ALIAS>
-Dkie.keystore.key.server.pwd=<KEY_SERVER_PWD>
-Dkie.keystore.key.ctrl.alias=<KEY_CONTROL_ALIAS>
-Dkie.keystore.key.ctrl.pwd=<KEY_CONTROL_PWD>
```

- Windows の場合は、以下の内容の **setenv.bat** ファイルを作成します。

```
set CATALINA_OPTS="
-Dkie.keystore.keyStoreURL=<KEYSTORE_URL>
-Dkie.keystore.keyStorePwd=<KEYSTORE_PWD>
-Dkie.keystore.key.server.alias=<KEY_SERVER_ALIAS>
-Dkie.keystore.key.server.pwd=<KEY_SERVER_PWD>
-Dkie.keystore.key.ctrl.alias=<KEY_CONTROL_ALIAS>
-Dkie.keystore.key.ctrl.pwd=<KEY_CONTROL_PWD>
```

表35.1 KIE Server JCEKS を読み込む時に使用するシステムプロパティ

システムプロパティ	プレースホルダー	説明
<b>kie.keystore.keyStoreURL</b>	<b>&lt;KEYSTORE_URL&gt;</b>	使用する JCEKS の URL (例: <b>file:///home/kie/keystores/keystore.jceks</b> )
<b>kie.keystore.keyStorePwd</b>	<b>&lt;KEYSTORE_PWD&gt;</b>	JCEKS のパスワード
<b>kie.keystore.key.server.alias</b>	<b>&lt;KEY_SERVER_ALIAS&gt;</b>	パスワードの保存先となる REST サービスのキーのエイリアス
<b>kie.keystore.key.server.pwd</b>	<b>&lt;KEY_SERVER_PWD&gt;</b>	保存したパスワードを使用する REST サービスのエイリアスのパスワード
<b>kie.keystore.key.ctrl.alias</b>	<b>&lt;KEY_CONTROL_ALIAS&gt;</b>	パスワードの保存先のデフォルトの REST Process Automation Controller のキーのエイリアス
<b>kie.keystore.key.ctrl.pwd</b>	<b>&lt;KEY_CONTROL_PWD&gt;</b>	保存したパスワードを使用する、デフォルトの REST Process Automation Controller のエイリアスのパスワード

6. KIE Server を起動して、設定を検証します。

## 第36章 KIE SERVER インストールの確認

KIE Server が正しくインストールされていることを確認します。

### 前提条件

- KIE Server がインストールされ、設定されている。

### 手順

1. KIE Server を起動するには、**JWS\_HOME/tomcat/bin** ディレクトリーから端末を開いて以下のコマンドのいずれかを入力します。
  - Linux または UNIX ベースのシステムの場合:  

```
$ ./startup.sh
```
  - Windows の場合:  

```
startup.bat
```
2. 数分後に **JWS\_HOME/tomcat/logs** ディレクトリーのファイルを確認し、エラーが発生している場合は修正します。
3. Web ブラウザーに **http://localhost:8080/kie-server/services/rest/server** を入力し、KIE Server が正常に動作していることを確認します。
4. **tomcat-users.xml** ファイルに保存したユーザー名とパスワードを入力します。

## 第37章 ヘッドレス DECISION MANAGER コントローラーのダウンロードおよびインストール

KIE Server は、管理モードまたは非管理モードで動作するように設定できます。KIE Server が非管理モードの場合は、手動で KIE コンテナ (デプロイメントユニット) を作成および維持する必要があります。管理モードの場合は、Decision Manager コントローラーが KIE Server の設定を管理し、ユーザーはコントローラーと対話形式で KIE コンテナを作成して維持します。

Decision Manager コントローラーは Business Central と統合します。Business Central をインストールしている場合は、**Execution Server** ページを使用して KIE コンテナを作成および維持します。ただし、Business Central をインストールしない場合は、ヘッドレス Decision Manager コントローラーをインストールし、REST API または KIE Server Java Client API を使用してそのコントローラーと対話します。

### 前提条件

- [32章 Red Hat Decision Manager インストールファイルのダウンロード](#) に従って **Red Hat Decision Manager 7.10.0 Add Ons (rhdm-7.10.0-add-ons.zip)** ファイルがダウンロードされている。
- Red Hat JBoss Web Server 5.4.0 サーバーインストールが利用できる。Red Hat JBoss Web Server インストールのベースディレクトリーは **JWS\_HOME** とします。
- インストールを完了するのに必要なユーザーパーミッションが付与されている。

### 手順

1. **rhdm-7.10.0-add-ons.zip** ファイルを展開します。**rhdm-7.10.0-controller-jws.zip** ファイルは、展開したディレクトリーにあります。
2. **rhdm-7.10.0-controller-jws.zip** アーカイブを一時ディレクトリーに展開します。以下の例では、この名前を **TEMP\_DIR** とします。
3. **TEMP\_DIR/rhdm-7.10.0-controller-jws.zip/controller.war** ディレクトリーを **JWS\_HOME/tomcat/webapps** ディレクトリーにコピーします。



#### 注記

コピーする Red Hat Decision Manager デプロイメントの名前が Red Hat JBoss Web Server インスタンスの既存デプロイメントと競合しないことを確認します。

4. **controller.war** ディレクトリーから **.war** 拡張子を削除します。
5. **TEMP\_DIR/rhdm-7.10.0-controller-jws/SecurityPolicy/** ディレクトリーの内容を **JWS\_HOME/bin** にコピーします。
6. ファイルの上書きを求めるプロンプトが出されたら、**Yes** を選択します。
7. **JWS\_HOME/tomcat/conf/tomcat-users.xml** ファイルに **kie-server** ロールおよびユーザーを追加します。以下の例で、**<USER\_NAME>** と **<PASSWORD>** は、任意のユーザー名とパスワードです。

```
<role rolename="kie-server"/>
<user username="<USER_NAME>" password="<PASSWORD>" roles="kie-server"/>
```

8. KIE Server を実行するインスタンスの **JWS\_HOME/tomcat/bin** ディレクトリーで、以下のタスクの1つを実行します。

- Linux または UNIX の場合は、以下の内容の **setenv.sh** ファイルを作成します。

```
CATALINA_OPTS="-Xmx1024m
-Dorg.jboss.logging.provider=jdk
-Dorg.jbpm.server.ext.disabled=true
-Dorg.jbpm.ui.server.ext.disabled=true
-Dorg.jbpm.case.server.ext.disabled=true
-Dorg.kie.server.controller.user=<CONTROLLER_USER>
-Dorg.kie.server.controller.pwd=<CONTROLLER_PWD>
-Dorg.kie.server.id=<KIE_SERVER_ID>
-Dorg.kie.server.location=http://<HOST>:<PORT>/kie-server/services/rest/server
-Dorg.kie.server.controller=http://<HOST>:<PORT>/controller/rest/controller"
```

- Windows の場合は、以下の内容の **setenv.bat** ファイルを作成します。

```
set CATALINA_OPTS= -Xmx1024m
-Dorg.jbpm.server.ext.disabled=true
-Dorg.jbpm.ui.server.ext.disabled=true
-Dorg.jbpm.case.server.ext.disabled=true
-Dorg.kie.server.controller.user=<CONTROLLER_USER>
-Dorg.kie.server.controller.pwd=<CONTROLLER_PWD>
-Dorg.kie.server.id=<KIE_SERVER_ID>
-Dorg.kie.server.location=http://<HOST>:<PORT>/kie-server/services/rest/server
-Dorg.kie.server.controller=http://<CONTROLLER_HOST>:
<CONTROLLER_PORT>/controller/rest/controller
```

上記の例では、以下の変数に置き換えてください。

- **<CONTROLLER\_USER>** および **<CONTROLLER\_PWD>** を、この手順の最初の方で定義した **kie-server** ロールを持つユーザー名とパスワードに置き換えます。
- **<KIE\_SERVER\_ID>** を一意の識別子に置き換えます。
- **<CONTROLLER\_HOST>:<CONTROLLER\_PORT>** をコントローラーの IP アドレス (ホストとポート) に置き換えます。KIE Server とコントローラーで同じサーバーを使用する場合は、**<CONTROLLER\_HOST>:<CONTROLLER\_PORT>** は **localhost:8080** に置き換えます。

9. ヘッドレス Decision Manager コントローラーを実行しているインスタンスの **JWS\_HOME/tomcat/bin** ディレクトリーに、以下の内容で読み取り可能な **setenv.sh** ファイルを作成します。ここで、**<USERNAME>** は KIE Server ユーザーで、**<USER\_PWD>** はそのユーザーのパスワードに置き換えます。

```
CATALINA_OPTS="-Dorg.kie.server.user=<USERNAME> -Dorg.kie.server.pwd=
<USER_PWD>"
```

10. ヘッドレス Decision Manager コントローラーを起動するには、**JWS\_HOME/tomcat/bin** ディレクトリーに以下のいずれかのコマンドを実行します。

- Linux または UNIX ベースのシステムの場合:

■

```
$ ./startup.sh
```

- Windows の場合:

```
startup.bat
```

11. 数分後に **JWS\_HOME/tomcat/logs** ディレクトリーを確認し、エラーが発生している場合には修正します。
12. ヘッドレス Decision Manager コントローラーが正常に動作していることを確認するには、Web ブラウザーに **http://<CONTROLLER\_HOST>:<CONTROLLER\_PORT>/controller/rest/controller/management/servers** を入力します。KIE Server とコントローラーで同じサーバーを使用する場合は、**<CONTROLLER\_HOST>:<CONTROLLER\_PORT>** は **localhost:8080** に置き換えます。
13. **tomcat-users.xml** ファイルに保存したユーザー名とパスワードを入力します。

## 第38章 スタンドアロン BUSINESS CENTRAL の設定および実行

Business Central スタンドアロンの JAR ファイルとして使用して、アプリケーションサーバーにデプロイせずに Business Central を実行できます。設定ファイルのサンプルを使用して、追加設定なしで Business Central スタンドアロン JAR ファイルを実行するか、要件に合わせてサンプルファイルをカスタマイズできます。



### 注記

この JAR ファイルは、Red Hat Enterprise Linux で実行している場合にのみサポートされます。

### 前提条件

- 32章 [Red Hat Decision Manager インストールファイルのダウンロード](#) の記載通りに、Red Hat Decision Manager 7.10.0 Business Central Standalone (**rhdm-7.10.0-decision-central-standalone.jar**) ファイルおよび Red Hat Decision Manager 7.10.0 Add Ons (**rhdm-7.10.0-addons.zip**) ファイルが、Red Hat Decision Manager 7.10 の「[Software Downloads](#)」ページからダウンロードしている。

### 手順

1. ダウンロードした **rhdm-7.10.0-addons.zip** ファイルを一時ディレクトリーに展開します。このアーカイブには **rhdm-7.10.0-standalone-sample-configuration.zip** ファイルが含まれます。
2. **rhdm-7.10.0-standalone-sample-configuration.zip** ファイルを、**rhdm-7.10.0-decision-central-standalone.jar** ファイルが含まれるディレクトリーに展開します。**rhdm-7.10.0-standalone-sample-configuration.zip** ファイルには、以下の設定ファイルのサンプルが含まれています。

- **application-roles.properties**

- **application-users.properties**

- **sample-standalone-config.yml**

設定ファイルで提供されるサンプルデータを使用して **rhdm-7.10.0-decision-central-standalone.jar** ファイルを実行するか、要件に合わせてデータをカスタマイズできます。

3. 設定データをカスタマイズするには、以下の手順を実行します。

- a. **application-users.properties** ファイルを編集します。管理者ユーザーを含めます。この Business Central インスタンスが KIE Server の Decision Manager コントローラーになる場合は、Decision Manager コントローラーユーザーを含めます。以下に例を示します。

```
rhdmAdmin=password1
```

- b. **application-roles.properties** ファイルを編集し、**application-users.properties** ファイルに含まれるユーザーにロールを割り当てます。以下に例を示します。

```
rhdmAdmin=admin
controllerUser=kie-server
```

詳細は、「[Red Hat Decision Manager ロールおよびユーザー](#)」を参照してください。

- c. **sample-standalone-config.yml** 設定ファイルの内容を確認し、必要に応じて更新します。



4. Business Central スタンドアロン JAR ファイルを実行するには、以下のコマンドを入力します。

```
java -jar rhdm-7.10.0-decision-central-standalone.jar -s sample-standalone-config.yml
```

5. JAR ファイルの実行時にアプリケーションプロパティを設定するには、コマンドに **-D<PROPERTY>=<VALUE>** パラメーターを追加します。ここで、**<PROPERTY>** はサポートされるアプリケーションプロパティの名前で、**<VALUE>** はプロパティの値になります。

```
java -jar rhdm-7.10.0-decision-central-standalone.jar -s sample-standalone-config.yml -  
D<PROPERTY>=<VALUE> -D<PROPERTY>=<VALUE>
```

たとえば、Business Central を実行し、ユーザー **controllerUser** として KIE Server に接続するには、次のコマンドを実行します。

```
java -jar {PRODUCT_FILE}-{URL_COMPONENT_CENTRAL}-standalone.jar \  
-s sample-standalone-config.yml \  
-Dorg.kie.server.user=controllerUser  
-Dorg.kie.server.pwd=controllerUser1234
```

これにより、コンテナを KIE Server にデプロイできます。詳細は、[付録A Business Central システムプロパティ](#)を参照してください。

## 第39章 RED HAT DECISION MANAGER の MAVEN 設定およびリポジトリ

Red Hat Decision Manager プロジェクトの作成時に、Business Central は Business Central 用に設定された Maven リポジトリを使用します。Maven project object model (POM) ファイル (**pom.xml**) を変更することで、Maven グローバルまたはユーザー設定を使用し、すべての Red Hat Decision Manager プロジェクトに対して Red Hat Decision Manager の公開リポジトリから依存関係を取得するように指示できます。また、Business Central と KIE Server が外部の Maven リポジトリを使用するか、Maven のミラーを準備してオフラインでできるように設定できます。

Red Hat Decision Manager プロジェクトのパッケージ化およびデプロイメントオプションに関する詳細は、[Red Hat Decision Manager プロジェクトのパッケージ化およびデプロイ](#) を参照してください。

### 39.1. プロジェクト設定ファイル (POM.XML) を使用した MAVEN の設定

Maven を使用して Red Hat Decision Manager プロジェクトをビルドおよび管理するには、POM ファイル (**pom.xml**) を作成および設定する必要があります。このファイルにはプロジェクトの設定情報が含まれます。詳細は [Apache Maven Project](#) を参照してください。

#### 手順

1. Maven プロジェクトを生成します。**pom.xml** ファイルは、Maven プロジェクトの作成時に自動的に生成されます。
2. **pom.xml** ファイルを編集して、追加の依存関係および新たなリポジトリを追加します。プロジェクトのコンパイルおよびパッケージング時に、Maven がすべての JAR ファイルおよび依存関係にある JAR ファイルを Maven リポジトリからダウンロードします。

**pom.xml** ファイルのスキーマは、[http://maven.apache.org/maven-v4\\_0\\_0.xsd](http://maven.apache.org/maven-v4_0_0.xsd) から取得できます。POM ファイルの詳細は [Apache Maven Project POM](#) を参照してください。

### 39.2. MAVEN 設定ファイルの修正

Red Hat Decision Manager では、Maven の **settings.xml** ファイルを使用して Maven の実行を設定します。**settings.xml** ファイルにプロファイルを作成およびアクティベートし、Red Hat Decision Manager プロジェクトが使用する Maven リポジトリを宣言します。

Maven の **settings.xml** ファイルの詳細は Apache Maven Project の [Setting Reference](#) を参照してください。

#### 手順

1. **settings.xml** ファイルでは、Red Hat Decision Manager プロジェクトが使用するリポジトリを宣言します。通常、これは Red Hat カスタマーポータルからダウンロードする [オンラインの Red Hat Decision Manager Maven リポジトリ](#) または Red Hat Decision Manager Maven リポジトリと、使用するカスタムアーティファクトのリポジトリです。
2. Business Central または KIE Server が **settings.xml** ファイルを使用するように設定されていることを確認します。たとえば、**kie.maven.settings.custom=<SETTINGS\_FILE\_PATH>** プロパティを指定し、**<SETTINGS\_FILE\_PATH>** は **settings.xml** ファイルのパスに置き換えます。

Red Hat JBoss Web Server で、KIE Server の場合は **-Dkie.maven.settings.custom=<SETTINGS\_FILE\_PATH>** を **setenv.sh** (Linux) または **setenv.bat** (Windows) ファイルの **CATALINA\_OPTS** セクションに追加します。スタンドアロンの Business Central の場合は、以

下のコマンドを入力します。

```
java -jar rhpam-7.10.0-business-central-standalone.jar -s application-config.yaml -
Dkie.maven.settings.custom=<SETTINGS_FILE_PATH>
```

### 39.3. RED HAT DECISION MANAGER の MAVEN 依存関係の追加

ご自分の Red Hat Decision Manager プロジェクトで適切な Maven 依存関係を使用するには、プロジェクトの **pom.xml** ファイルに Red Hat Business Automation の BOM (bill of materials) ファイルを追加します。Red Hat Business Automation BOM は、Red Hat Decision Manager と Red Hat Process Automation Manager の両方に適用されます。BOM ファイルを追加すると、提供される Maven リポジトリから、推移的依存関係の適切なバージョンがプロジェクトに含められます。

Red Hat Business Automation BOM (Bill of Materials) についての詳細情報は、[What is the mapping between Red Hat Decision Manager and the Maven library version?](#) を参照してください。

#### 手順

1. Red Hat Business Automation BOM を **pom.xml** ファイルで宣言します。

```
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>com.redhat.ba</groupId>
      <artifactId>ba-platform-bom</artifactId>
      <version>7.10.0.redhat-00004</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
<dependencies>
  <!-- Your dependencies -->
</dependencies>
```

2. **<dependencies>** タグでお使いのプロジェクトに必要な依存関係を宣言します。製品の BOM をプロジェクトにインポートしたら、ユーザー向け製品依存関係のバージョンが定義されるため、**<dependency>** 要素のサブ要素 **<version>** を指定する必要はありません。ただし、**<dependency>** 要素を使用して、プロジェクトで使用する依存関係を宣言する必要があります。
3. Business Central に作成されない標準のプロジェクトでは、お使いのプロジェクトに必要な依存関係をすべて指定します。Business Central に作成するプロジェクトでは、基本的なデシジョンエンジンの依存関係が Business Central に自動的に提供されます。
  - 標準的な Red Hat Decision Manager プロジェクトでは、使用する機能に応じて、以下の依存関係を宣言します。
  - 標準的な Red Hat Decision Manager プロジェクトでは、以下の依存関係を宣言します。

#### 埋め込みデシジョンエンジン依存関係

```
<dependency>
  <groupId>org.drools</groupId>
```

```

<artifactId>drools-compiler</artifactId>
</dependency>

<!-- Dependency for persistence support. -->
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-persistence-jpa</artifactId>
</dependency>

<!-- Dependencies for decision tables, templates, and scorecards.
For other assets, declare org.drools:business-central-models-* dependencies. -->
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-decisiontables</artifactId>
</dependency>
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-templates</artifactId>
</dependency>
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-scorecards</artifactId>
</dependency>

<!-- Dependency for loading KJARs from a Maven repository using KieScanner. -->
<dependency>
  <groupId>org.kie</groupId>
  <artifactId>kie-ci</artifactId>
</dependency>

```

- KIE Server を使用するには、以下の依存関係を宣言します。

#### クライアントアプリケーション KIE Server の依存関係

```

<dependency>
  <groupId>org.kie.server</groupId>
  <artifactId>kie-server-client</artifactId>
</dependency>

```

- Red Hat Decision Manager にリモートクライアントを作成するには、以下の依存関係を宣言します。

#### クライアントの依存関係

```

<dependency>
  <groupId>org.uberfire</groupId>
  <artifactId>uberfire-rest-client</artifactId>
</dependency>

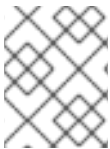
```

- ルール、プロセス定義など、アセットを含む JAR ファイルを作成する場合は、お使いの Maven プロジェクトのパッケージングの種類を **kjar** と指定し、**org.kie:kie-maven-plugin** を使用して、**<project>** 要素に置かれた **kjar** パッケージングタイプを処理します。以下の例の **\${kie.version}** は、[What is the mapping between Red Hat Decision Manager and the Maven library version?](#) に記載されている Maven ライブラリーのバージョンです。

```
<packaging>kjar</packaging>
<build>
  <plugins>
    <plugin>
      <groupId>org.kie</groupId>
      <artifactId>kie-maven-plugin</artifactId>
      <version>${kie.version}</version>
      <extensions>true</extensions>
    </plugin>
  </plugins>
</build>
```

### 39.4. オフラインで使用する MAVEN ミラーリポジトリの用意

公開インターネットへの送信アクセスが、Red Hat Decision Manager のデプロイメントに設定されていない場合は、必要なすべてのアーティファクトのミラーが含まれる Maven リポジトリを用意して、このリポジトリを使用できるようにする必要があります。



#### 注記

Red Hat Decision Manager デプロイメントがインターネットに接続されている場合は、この手順を実行する必要はありません。

#### 前提条件

- 公開インターネットへの送信アクセスが設定されているコンピューターが利用できる。

#### 手順

- 公開インターネットに送信アクセスができるコンピューターで、以下のアクションを実行します。
- Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - 製品:** Red Hat Decision Manager
  - バージョン:** 7.10

- Red Hat Process Automation Manager 7.10.0 Offliner Content List(rhdm-7.10.0-offliner.zip)** の製品配信可能ファイルをダウンロードして展開します。
- rhdm-7.10.0-offliner.zip** ファイルの内容を任意のディレクトリーに展開します。
- ディレクトリーに移動し、以下のコマンドを入力します。

```
./offline-repo-builder.sh offliner.txt
```

このコマンドは、**repository** サブディレクトリーを作成し、必要なアーティファクトをこのサブディレクトリーにダウンロードします。これはミラーリポジトリです。

一部のダウンロードが失敗したことを示すメッセージが表示された場合は、同じコマンドを再度実行してください。ダウンロードが再び失敗する場合は、Red Hat サポートに連絡してください。

3. Business Central 外でサービスを開発し、追加の依存関係がある場合は、ミラーリポジトリにその依存関係を追加します。サービスを Maven プロジェクトとして開発した場合は、以下の手順を使用し、これらの依存関係を自動的に用意します。公開インターネットへに送信接続できるコンピューターで、この手順を実行します。

- a. ローカルの Maven キャッシュディレクトリー (`~/.m2/repository`) のバックアップを作成して、ディレクトリーを削除します。
- b. **mvn clean install** コマンドを使用してプロジェクトのソースをビルドします。
- c. すべてのプロジェクトで以下のコマンドを入力し、Maven を使用してプロジェクトで生成したすべてのアーティファクトのランタイムの依存関係をすべてダウンロードするようにします。

```
mvn -e -DskipTests dependency:go-offline -f /path/to/project/pom.xml --batch-mode -Djava.net.preferIPv4Stack=true
```

`/path/to/project/pom.xml` を、プロジェクトの **pom.xml** ファイルのパスに置き換えます。

- d. ローカルの Maven キャッシュディレクトリー (`~/.m2/repository`) の内容を、作成した **repository** サブディレクトリーにコピーします。
4. **repository** サブディレクトリーのコンテンツを、Red Hat Decision Manager をデプロイしたコンピューターのディレクトリーにコピーします。このディレクトリーがオフラインの Maven ミラーリポジトリになります。
  5. 「[Maven 設定ファイルの修正](#)」の説明に従って、Red Hat Decision Manager デプロイメント向けに、**settings.xml** ファイルを作成して設定します。
  6. **settings.xml** ファイルで以下を変更します。

- **<profile>** タグの下に、**<repositories>** タグまたは **<pluginRepositories>** タグがない場合は、不足しているタグを追加します。
- **<repositories>** の下に、以下のコンテンツを追加します。

```
<repository>
  <id>offline-repository</id>
  <url>file:///path/to/repo</url>
  <releases>
    <enabled>true</enabled>
  </releases>
  <snapshots>
    <enabled>>false</enabled>
  </snapshots>
</repository>
```

`/path/to/repo` は、ローカルの Maven ミラーリポジトリのディレクトリーに対する完全パスに置き換えます。

- **<pluginRepositories>** の下に、以下のコンテンツを追加します。

```
<repository>
  <id>offline-plugin-repository</id>
  <url>file:///path/to/repo</url>
  <releases>
```

```
<enabled>true</enabled>
</releases>
<snapshots>
  <enabled>false</enabled>
</snapshots>
</repository>
```

**/path/to/repo** は、ローカルの Maven ミラーリポジトリのディレクトリーに対する完全パスに置き換えます。

## 第40章 GIT リポジトリからのプロジェクトのインポート

Git は分散バージョン管理システムです。リビジョンをコミットオブジェクトとして実装します。リポジトリに変更を保存すると、Git リポジトリに新しいコミットオブジェクトが作成されます。

Business Central は Git を使用してプロジェクトデータ (ルールやプロセスなどのアセットを含む) を格納します。Business Central でプロジェクトを作成すると、Business Central に埋め込まれている Git リポジトリに追加されます。他の Git リポジトリにプロジェクトがある場合は、Business Central スペースから、そのプロジェクトを Business Central の Git リポジトリにインポートできます。

### 前提条件

- Red Hat Decision Manager プロジェクトが外部の Git リポジトリに存在している。
- 外部の Git リポジトリへの読み取りアクセスに必要な認証情報がある。

### 手順

1. Business Central で **Menu → Design → Projects** の順にクリックします。
2. プロジェクトをインポートするスペースを選択または作成します。デフォルトのスペースは **MySpace** です。
3. プロジェクトをインポートするには、次のいずれかを実行します。
  - **Import Project** をクリックします。
  - ドロップダウンリストから **Import Project** を選択します。
4. **Import Project** ウィンドウに、インポートするプロジェクトが含まれる Git リポジトリの URL および認証情報を入力し、**Import** をクリックします。プロジェクトが Business Central の Git リポジトリに追加され、現在のスペースで利用できるようになります。



## 第41章 LDAP と SSL の統合

Red Hat Decision Manager を使用して、RH-SSO を介して LDAP および SSL を統合できます。詳細は [Red Hat Single Sign-On サーバー管理ガイド](#) を参照してください。

## 付録B BUSINESS CENTRAL システムプロパティ

このセクションに記載の Business Central のシステムプロパティは **standalone\*.xml** ファイルに渡されます。

### Git ディレクトリー

以下のプロパティを使用して、Business Central Git ディレクトリーの場所と名前を設定します。

- **org.uberfire.nio.git.dir**: Business Central の Git ディレクトリーの場所。
- **org.uberfire.nio.git.dirname**: Business Central の Git ディレクトリーの名前。デフォルト値は **.niogit** です。
- **org.uberfire.nio.git.ketch**: Git ketch を有効化または無効化。
- **org.uberfire.nio.git.hooks**: Business Central の Git ディレクトリーの場所。

### HTTP 経由の Git

次のプロパティを使用して、HTTP 経由で Git リポジトリにアクセスできるように設定します。

- **org.uberfire.nio.git.proxy.ssh.over.http**: SSH が HTTP プロキシを使用するかどうかを指定します。デフォルト値は **false** です。
- **http.proxyHost**: HTTP プロキシのホスト名を定義します。デフォルト値は **null** です。
- **http.proxyPort**: HTTP プロキシのポート (整数値) を定義します。デフォルト値は **null** です。
- **http.proxyUser**: HTTP プロキシ名を定義します。
- **http.proxyPassword**: HTTP プロキシのユーザーパスワードを定義します。
- **org.uberfire.nio.git.http.enabled**: HTTP デモンを有効または無効にします。デフォルト値は **true** です。
- **org.uberfire.nio.git.http.host**: このデモンは、HTTP デモンが有効な場合にホストの識別子としてこのプロパティを使用します。これは、HTTP 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTP は、継続してサブレットコンテナに依存します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.http.hostname**: HTTP デモンが有効な場合に、このデモンはホスト名の識別子としてこのプロパティを使用します。これは、HTTP 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTP は、継続してサブレットコンテナに依存します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.http.port**: このデモンは、HTTP デモンが有効な場合にポート番号としてこのプロパティを使用します。これは、HTTP 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTP は、継続してサブレットコンテナに依存します。デフォルト値は **8080** です。

### HTTPS 経由の Git

次のプロパティを使用して、HTTPS 経由で Git リポジトリにアクセスできるように設定します。

- **org.uberfire.nio.git.proxy.ssh.over.https**: SSH が HTTPS プロキシを使用するかどうかを指定します。デフォルト値は **false** です。

- **https.proxyHost**: HTTPS プロキシのホスト名。デフォルト値は **null** です。
- **https.proxyPort**: HTTPS プロキシのホストポート (整数値)。デフォルト値は **null** です。
- **https.proxyUser**: HTTPS プロキシ名を定義します。
- **https.proxyPassword**: HTTPS プロキシのユーザーパスワードを定義します。
- **user.dir**: ユーザーディレクトリーの場所。
- **org.uberfire.nio.git.https.enabled**: HTTPS デーモンを有効または無効にします。デフォルト値は **false** です。
- **org.uberfire.nio.git.https.host**: このデーモンは、HTTPS デーモンが有効な場合にホストの識別子としてこのプロパティーを使用します。これは、HTTPS 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTPS は、継続してサブレットコンテナに依存します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.https.hostname**: このデーモンは、HTTPS デーモンが有効な場合にホスト名の識別子としてこのプロパティーを使用します。これは、HTTPS 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTPS は、継続してサブレットコンテナに依存します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.https.port**: このデーモンは、HTTPS デーモンが有効な場合にポート番号としてこのプロパティーを使用します。これは、HTTPS 経由で Git リポジトリにアクセスする方法を表示するときに使用する参考属性です。HTTPS は、継続してサブレットコンテナに依存します。デフォルト値は **8080** です。

## JGit

- **org.uberfire.nio.jgit.cache.instances**: JGit キャッシュサイズを定義します。
- **org.uberfire.nio.jgit.cache.overflow.cleanup.size**: JGit キャッシュオーバーフローのクリーンアップサイズを定義します。
- **org.uberfire.nio.jgit.remove.eldest.iterations**: 最も古い JGit の反復を削除するかどうかを定義します。
- **org.uberfire.nio.jgit.cache.evict.threshold.duration**: JGit 退避のしきい値の期間を定義します。
- **org.uberfire.nio.jgit.cache.evict.threshold.time.unit**: JGit 退避のしきい値の時間単位を定義します。

## Git デーモン

次のプロパティーを使用して、Git デーモンを有効にして設定します。

- **org.uberfire.nio.git.daemon.enabled**: Git デーモンを有効または無効にします。デフォルト値は **true** です。
- **org.uberfire.nio.git.daemon.host**: Git デーモンが有効な場合は、このプロパティーをローカルホストの識別子として使用します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.daemon.hostname**: Git デーモンが有効な場合は、このプロパティーをローカルホスト名の識別子として使用します。デフォルト値は **localhost** です。

- **org.uberfire.nio.git.daemon.port**: Git デーモンが有効な場合は、このプロパティをポート番号として使用します。デフォルト値は **9418** です。
- **org.uberfire.nio.git.http.sslVerify**: Git リポジトリを確認する SSL 証明書を有効または無効にします。デフォルト値は **true** です。



#### 注記

デフォルトポートまたは割り当てられたポートが既に使用されている場合は、別のポートが自動的に選択されます。ポートが利用可能であることを確認し、詳細についてはログをチェックします。

## Git SSH

次のプロパティを使用して、Git SSH デーモンを有効にして設定します。

- **org.uberfire.nio.git.ssh.enabled**: SSH デーモンを有効または無効にします。デフォルト値は **true** です。
- **org.uberfire.nio.git.ssh.host**: SSH デーモンが有効な場合は、このプロパティをローカルホスト識別子として使用します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.ssh.hostname**: SSH デーモンが有効な場合は、このプロパティをローカルホスト名の識別子として使用します。デフォルト値は **localhost** です。
- **org.uberfire.nio.git.ssh.port**: SSH デーモンが有効な場合は、このプロパティをポート番号として使用します。デフォルト値は **8001** です。



#### 注記

デフォルトポートまたは割り当てられたポートが既に使用されている場合は、別のポートが自動的に選択されます。ポートが利用可能であることを確認し、詳細についてはログをチェックします。

- **org.uberfire.nio.git.ssh.cert.dir**: ローカルの証明書が保存される **.security** ディレクトリーの場所。デフォルトは作業ディレクトリーです。
- **org.uberfire.nio.git.ssh.idle.timeout**: SSH のアイドルタイムアウトを設定します。
- **org.uberfire.nio.git.ssh.passphrase**: SCP スタイルの URL を持つ Git リポジトリのクローンを作成する場合に、オペレーティングシステムの公開キーストアにアクセスするためのパスフレーズ。たとえば、**git@github.com:user/repository.git** です。
- **org.uberfire.nio.git.ssh.algorithm**: SSH で使用されるアルゴリズム。デフォルト値は **RSA** です。
- **org.uberfire.nio.git.gc.limit**: GC の制限を設定します。
- **org.uberfire.nio.git.ssh.ciphers**: コンマ区切りの暗号化の文字列。利用可能な暗号化は **aes128-ctr**、**aes192-ctr**、**aes256-ctr**、**arcfour128**、**arcfour256**、**aes192-cbc**、**aes256-cbc** です。このプロパティを使用しない場合は、すべての暗号化が読み込まれます。
- **org.uberfire.nio.git.ssh.macs**: コンマ区切りのメッセージ認証コード (MAC) の文字列。利用可能な MAC は **hmac-md5**、**hmac-md5-96**、**hmac-sha1**、**hmac-sha1-96**、**hmac-sha2-256**、**hmac-sha2-512** です。このプロパティを使用しない場合は、すべての MAC が読み込まれます。



## 注記

RSA、または DSA 以外のアルゴリズムを使う場合は、Bouncy Castle JCE ライブラリーを使用するようにアプリケーションサーバーを設定します。

## KIE Server ノードおよび Decision Manager コントローラー

以下のプロパティを使用して Decision Manager コントローラーと KIE Server ノードの接続を設定します。

- **org.kie.server.controller:** URL は Decision Manager コントローラーへの接続に使用されます。たとえば、**ws://localhost:8080/decision-central/websocket/controller** です。
- **org.kie.server.user:** Decision Manager コントローラーから KIE Server ノードに接続する時に使用するユーザー名。このプロパティが必要なのは、この Business Central システムを Decision Manager コントローラーとして使用する場合があります。
- **org.kie.server.pwd:** Decision Manager コントローラーから KIE Server ノードに接続する時に使用するパスワード。このプロパティが必要なのは、この Business Central システムを Decision Manager コントローラーとして使用する場合があります。

## Maven など

以下のプロパティを使用して、Maven などの機能を設定します。

- **kie.maven.offline.force:** Maven のオフライン動作を強制します。true に設定すると、オンラインの依存関係解決が無効になります。デフォルト値は **false** です。



## 注記

このプロパティは、Business Central にのみ使用してください。他のコンポーネントとランタイム環境を共有する場合は、設定を分離して、Business Central にだけ適用してください。

- **org.uberfire.gzip.enable:** **GzipFilter** 圧縮フィルターで Gzip の圧縮を有効にするか、または無効にします。デフォルト値は **true** です。
- **org.kie.workbench.profile:** Business Central プロファイルを選択します。許容値は、**FULL** または **PLANNER\_AND\_RULES** です。プリフィックス **FULL\_** で、プロファイルを設定し、管理者設定にこのプロファイルの設定が表示されないようにします。デフォルト値は **FULL** です。
- **org.appformer.m2repo.url:** Business Central は依存関係を検索する時に、Maven リポジトリのデフォルトの場所を使用します。デフォルト値は、**http://localhost:8080/business-central/maven2** など、Business Central 内の Maven リポジトリを参照します。このプロパティは、Business Central が起動する前に設定してください。デフォルト値は、内部の **m2** リポジトリへのファイルパスです。
- **appformer.ssh.keystore:** クラス名を指定して Business Central で使用する、カスタムの SSH キーストアを定義します。このプロパティが指定されていない場合はデフォルトの SSH キーストアを使用します。
- **appformer.ssh.keys.storage.folder:** このプロパティは、デフォルトの SSH キーストアを使用する場合にユーザーの SSH 公開鍵の保存フォルダーを定義します。このプロパティを指定しないと、この公開鍵は Business Central の **.security** フォルダーに保存されます。

- **appformer.experimental.features:** 実験的機能のフレームワークを有効にします。デフォルト値は **false** です。
- **org.kie.demo:** GitHub 外部へのデモアプリケーションのクローン作成を有効にします。
- **org.uberfire.metadata.index.dir:** Lucene の **.index** ディレクトリーが保存される場所。デフォルトは作業ディレクトリーです。
- **org.uberfire ldap.regex.role\_mapper:** LDAP プリンシパル名をアプリケーションのロール名にマッピングするのに使用する regex パターン。プリンシパルの値とロール名が一致する場合は、アプリケーションのロール名が変数ロールに置き換えられるため、変数ロールはパターンの一部でなければならない点に注意してください。
- **org.uberfire.sys.repo.monitor.disabled:** 設定モニターを無効にします。無効にした場合の影響を正しく理解していない場合は、無効にしないでください。デフォルト値は **false** です。
- **org.uberfire.secure.key:** パスワードの暗号化で使用するパスワード。デフォルト値は **org.uberfire.admin** です。
- **org.uberfire.secure.alg:** パスワードの暗号化で使用する暗号化アルゴリズム。デフォルト値は **PBEWithMD5AndDES** です。
- **org.uberfire.domain:** uberfire が使用するセキュリティドメイン名。デフォルト値は **ApplicationRealm** です。
- **org.guvnor.m2repo.dir:** Maven リポジトリーディレクトリーが保存される場所。デフォルト値は **<working-directory>/repositories/kie** です。
- **org.guvnor.project.gav.check.disabled:** グループ ID、アーティファクト ID、およびバージョン (GAV) のチェックを無効にします。デフォルト値は **false** です。
- **org.kie.build.disable-project-explorer:** Project Explorer で選択したプロジェクトの自動ビルドを無効にします。デフォルト値は **false** です。
- **org.kie.builder.cache.size:** プロジェクトビルダーのキャッシュサイズを定義します。デフォルト値は **20** です。
- **org.kie.library.assets\_per\_page:** プロジェクト画面のページごとのアセット数をカスタマイズできます。デフォルト値は **15** です。
- **org.kie.verification.disable-dtable-realtime-verification:** デシジョンテーブルのリアルタイム確認および検証を無効にします。デフォルト値は **false** です。

## Decision Manager コントローラー

以下のプロパティを使用して、Decision Manager コントローラーへの接続方法を設定します。

- **org.kie.workbench.controller:** Decision Manager コントローラーとの接続に使用する URL。たとえば、**ws://localhost:8080/kie-server-controller/websocket/controller** です。
- **org.kie.workbench.controller.user:** Decision Manager コントローラーのユーザー名。デフォルト値は **kieserver** です。
- **org.kie.workbench.controller.pwd:** Decision Manager コントローラーのパスワード。デフォルト値は **kieserver1!** です。

- **org.kie.workbench.controller.token**: Decision Manager コントローラーとの接続に使用するトークン文字列。

## Java Cryptography Extension KeyStore (JCEKS)

JCEKS を設定するには、以下のプロパティを使用します。

- **kie.keystore.keyStoreURL**: Java Cryptography Extension KeyStore (JCEKS) の読み込みに使用する URL。たとえば、**file:///home/kie/keystores/keystore.jceks** です。
- **kie.keystore.keyStorePwd**: JCEKS に使用するパスワード。
- **kie.keystore.key.ctrl.alias**: デフォルトの REST Decision Manager コントローラーに使用するキーのエイリアス。
- **kie.keystore.key.ctrl.pwd**: デフォルトの REST Decision Manager コントローラーのエイリアスのパスワード。

## レンダリング

以下のプロパティを使用して、Business Central と KIE Server のレンダリングフォームを切り替えます。

- **org.jbpm.wb.forms.renderer.ext**: Business Central と KIE Server のフォームのレンダリングを切り替えます。デフォルトでは、フォームのレンダリングは Business Central が行います。デフォルト値は **false** です。
- **org.jbpm.wb.forms.renderer.name**: Business Central と KIE Server のレンダリングフォームを切り替えることができます。デフォルト値は **workbench** です。

## パート V. IBM WEBSPHERE APPLICATION SERVER への KIE SERVER のインストールおよび設定

システム管理者は、Red Hat KIE Server に IBM WebSphere Application Server を設定し、IBM サーバーインスタンスに KIE Server をインストールできます。

### 前提条件

- IBM WebSphere Application Server インスタンスのバージョン 9.0 以降がインストールされている。詳細なインストール手順は [IBM WebSphere Application Server 製品ページ](#) を参照してください、
- WebSphere 統合ソリューションコンソール (通常は <http://<HOST>:9060/ibm/console>) へのアクセスがある。



## 第42章 KIE SERVER

KIE Server は、Red Hat Decision Manager のルールおよびその他のアーティファクトが保存され、実行されるサーバーです。KIE Server はスタンドアロンで組み込みのコンポーネントで、REST、Java Message Service (JMS)、または Java クライアントサイドアプリケーションで利用可能なインターフェイス、およびソルバーを介した Red Hat Business Optimizer 機能を使用して、ルールのインスタンスを作成して実行することができます。

Web でデプロイ可能な WAR ファイルとして作成することで、KIE Server は Web コンテナであればどこにでもデプロイできます。KIE Server の現在のバージョンには、Red Hat Decision Manager および Red Hat Process Automation Manager の両方に対するデフォルトの拡張機能が含まれます。

KIE Server はメモリー消費が最小限でフットプリントが小さいため、クラウドインスタンスに簡単にデプロイできます。このサーバーの各インスタンスでは、複数のコンテナを開いてインスタンスを作成できるため、並行して複数のルールサービスを実行できます。

KIE Server は、Oracle WebLogic Server、IBM WebSphere Application Server などのアプリケーションサーバーと統合して、Red Hat Decision Manager のアプリケーション管理を合理化できます。

## 第43章 IBM WEBSPHERE APPLICATION SERVER

IBM WebSphere Application Server は、Java ベースの Web アプリケーションをホストし、Java EE 認定ランタイム環境を提供する、柔軟性がある安全な Web アプリケーションです。IBM WebSphere 9.0 は Java SE 8 に対応しており、Java EE 7 に完全に準拠しています。

## 第44章 IBM WEBSPPHERE APPLICATION SERVER のインストールおよび実行

KIE Server に対応する多くの設定を適用するために、IBM WebSphere Application Server をインストールして実行する必要があります。本セクションは、IBM WebSphere をインストールして起動する方法を説明します。

インストールに関する最新の詳細説明は、[IBM Knowledge Center](#) を参照してください。

### 手順

1. IBM Installation Manager バージョン 1.8.5 以降を [IBM Installation Manager and Packaging Utility download links](#) ページからダウンロードします。IBM WebSphere のインストールには IBM Installation Manager が必要です。
2. ダウンロードしたアーカイブを展開し、作成されたディレクトリーで、root 権限で以下のコマンドを実行します。

```
sudo ./install
```

IBM Installation Manager が開きます。

3. **File** → **Preferences** の順に移動して、**Add Repository** をクリックします。
4. **Add Repository** ウィンドウに、IBM WebSphere 9.0 のリポジトリ URL を入力します。すべてのリポジトリ URL は、IBM Knowledge Center の [Online product repositories for IBM WebSphere Application Server offerings](#) ページで確認できます。
5. 端末で、インストール時に指定した IBM WebSphere Application Server ディレクトリーの場所に移動します。
6. **/bin** ディレクトリーに変更し、以下の例のようなコマンドを実行して、IBM WebSphere のプロファイル、ユーザー名、およびパスワードを作成します。プロファイルは、ランタイム環境を定義します。プロファイルには、ランタイム環境でサーバーが処理し、変更できるファイルがすべて含まれます。このユーザーはログインに必要になります。

```
sudo ./manageprofiles.sh -create -profileName testprofile -profilePath /profiles/testprofile -adminUserName websphere -adminPassword password123
```

7. 端末で、作成したプロファイルの **bin** ディレクトリー (例: **/profiles/testprofile/bin**) に移動し、以下のコマンドを実行して IBM WebSphere Application Server インスタンスを起動します。

```
sudo ./startServer.sh <SERVER_NAME>
```

**<SERVER\_NAME>** は、WebSphere 統合ソリューションコンソールの **Servers** → **Server Types** → **IBM WebSphere Application Servers** で定義した IBM WebSphere Application Server の名前に置き換えます。

8. Web ブラウザーで、以下の URL を開きます。

```
http://<HOST>:9060/ibm/console
```

**<HOST>** は、ターゲットサーバーの名前または IP アドレスです。

たとえば、システムで起動している IBM WebSphere のローカルインスタンスに対して WebSphere 統合ソリューションコンソールを起動する場合は、以下の URL を Web ブラウザーに入力します。

`http://localhost:9060/ibm/console`

9. WebSphere 統合ソリューションコンソールのログインページが表示されたら、管理者の認証情報を入力します。

## 第45章 KIE SERVER への IBM WEBSPPHERE APPLICATION SERVER の設定

IBM WebSphere Application Server に KIE Server をデプロイする前に、システムプロパティ、セキュリティ設定、JMS 要件、その他の IBM WebSphere プロパティを設定する必要があります。この設定により、KIE Server との最適な統合が促進されます。

### 前提条件

- IBM WebSphere Application Server をインストールして実行している。
- WebSphere 統合ソリューションコンソールにログインしている。

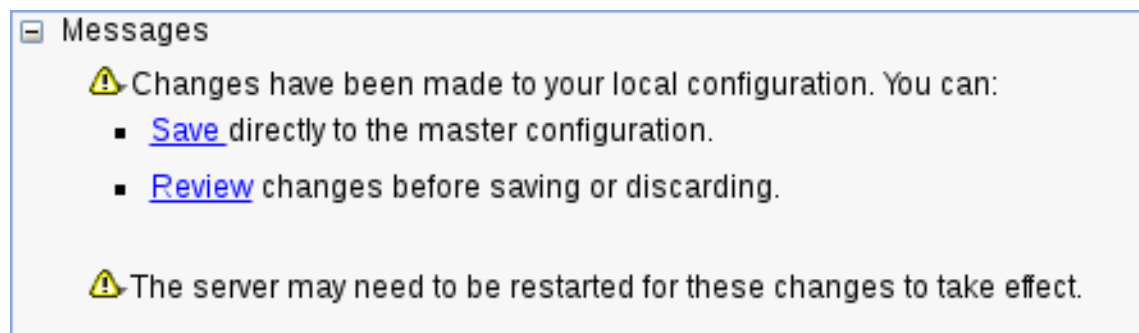
### 45.1. 管理セキュリティの有効化

WebSphere 統合ソリューションコンソールで管理セキュリティを有効にして、ユーザーおよびグループを作成するのに必要な権限を取得します。

#### 手順

1. WebSphere 統合ソリューションコンソールで、**Security** → **Global Security** の順にクリックし、**Enable Application Security** オプションが選択されていることを確認します。選択していても、サーバーレベルで上書きされている可能性があります。
2. **Security Configuration Wizard** をクリックし、**Next** をクリックします。
3. ユーザー情報を含むリポジトリを選択します。たとえば、ローカル設定で **Federated repositories** を選択します。
4. **Next** をクリックします。
5. **Primary administrative user name** および **Password** を入力します。
6. **Next** をクリックし、**Finish** をクリックします。
7. **Messages** ウィンドウで **Save** をクリックして、プライマリー設定への変更を保存します。

図45.1 セキュリティー変更の保存



8. 端末で、インストール時に指定した IBM WebSphere Application Server の **/bin** ディレクトリーの場所に移動し、以下のコマンドを実行して IBM WebSphere を停止して再起動し、セキュリティの変更を適用します。

```
sudo ./stopServer.sh <SERVER_NAME>
```

```
sudo ./startServer.sh <SERVER_NAME>
```

**<SERVER\_NAME>** は、WebSphere 統合ソリューションコンソールの **Servers → Server Types → IBM WebSphere Application Servers** で定義した IBM WebSphere Application Server の名前に置き換えます。

## 45.2. JAVA MESSAGE SERVICE (JMS) の設定

Java Message Service (JMS) は、KIE Server が、Oracle WebLogic Server、IBM WebSphere Application Server などのアプリケーションサーバーとメッセージを交換するために使用する Java API です。KIE Server を経由して JMS メッセージを送受信するようにアプリケーションサーバーを設定し、2 台のサーバー間で連携されるようにします。

### 45.2.1. サービスバスを作成し、IBM WebSphere Application Server を追加します。

JMS を使用するためには、サービスバスを作成し、IBM WebSphere Application Server をメンバーとして追加します。

#### 手順

1. WebSphere 統合ソリューションコンソールで、**Service Integration → Buses → New** の順に移動します。
2. 新しいバス名を入力し、**Bus Security** オプションの選択を解除します。
3. **Next** をクリックしてから **Finish** をクリックし、サービスバスを作成します。
4. 作成しておいたサービスバスを選択します。
5. **Topology** で **Bus Members → Add** を選択します。
6. **Add a New Bus Member** ウィザードを使用し、IBM WebSphere Application Server と、永続性に関するメッセージストアのタイプを選択します。メッセージストアのプロパティを指定することもできます。
7. **Finish** をクリックし、バスメンバーを追加します。

### 45.2.2. JMS 接続ファクトリーの作成

KIE Server でメッセージングを有効にするには、メッセージの送受信に JMS 接続ファクトリーをいくつか作成する必要があります。

#### 前提条件

- IBM WebSphere Application Server にサービスバスを作成している。

#### 手順

1. WebSphere 統合ソリューションコンソールで、**Resources → JMS → Connection Factories** の順に移動します。
2. 正しいスコープを選択し、**New** をクリックします。
3. **Default Messaging Provider** オプションを選択し、**OK** をクリックします。

- 以下に示す必須の接続ファクトリーに、接続ファクトリー名 (例: **KIE.SERVER.REQUEST**) および JNDI 名 (例: **jms/cf/KIE.SERVER.REQUEST**) を入力して、**Bus Name** ドロップダウンリストからサービスバス名を選択します。残りのオプションはデフォルト値のままにします。
- Apply**、**Save** の順にクリックして プライマリー設定への変更を保存し、必要な各ファクトリーに上述の手順を繰り返します。

#### 45.2.2.1. KIE Server の JMS 接続ファクトリー

以下の表は、KIE Server で JMS メッセージングを有効にするために必要な Java Message Service (JMS) 接続ファクトリーになります。

表45.1 KIE Server に必要な JMS 接続ファクトリー

名前	デフォルト値	用途
<b>KIE.SERVER.REQUEST</b>	<b>jms/cf/KIE.SERVER.REQUEST</b>	全要求を KIE Server へ送信
<b>KIE.SERVER.RESPONSE</b>	<b>jms/cf/KIE.SERVER.RESPONSE</b>	KIE Server が生成する応答をすべて受信

#### 45.2.3. JMS キューの作成

JMS キューは、ポイントツーポイントメッセージング (point-to-point messaging) の宛先エンドポイントになります。KIE Server で JMS メッセージングを有効にするには JMS キューをいくつか作成する必要があります。

##### 前提条件

- IBM WebSphere Application Server にサービスバスを作成している。

##### 手順

- WebSphere 統合ソリューションコンソールで、**Resources** → **JMS** → **Queues** の順に移動します。
- 正しいスコープを選択し、**New** をクリックします。
- Default Messaging Provider** オプションを選択し、**OK** をクリックします。
- 以下に示す必須のキューに、キューの名前 (例: **KIE.SERVER.REQUEST**) および JNDI 名 (例: **jms/KIE.SERVER.REQUEST**) を入力して、**Bus Name** ドロップダウンリストからサービスバスを選択します。
- Queue Name** ドロップダウンリストで **Create Service Integration Bus Destination** を選択して一意の識別子を入力し、事前に作成したバスメンバーを選択します。
- Apply**、**Save** の順にクリックして プライマリー設定への変更を保存し、必要な各キューに上述の手順を繰り返します。

#### 45.2.3.1. KIE Server 向けの JMS キュー

以下の表は、KIE Server で JMS メッセージングを有効にするために必要な JMS キューになります。

以下の表は、KIE Server で JMS メッセージングを有効にするために必要な Java Message Service (JMS) キューです。

表45.2 KIE Server に必要な JMS キュー

名前	デフォルト値	用途
<b>KIE.SERVER.REQUEST</b>	<b>jms/KIE.SERVER.REQUEST</b>	全要求を KIE Server へ送信
<b>KIE.SERVER.RESPONSE</b>	<b>jms/KIE.SERVER.RESPONSE</b>	KIE Server が生成する応答をすべて受信

#### 45.2.4. JMS アクティベーション指定の作成

キューと、JMS を有効にするメッセージ駆動型 Bean との間にブリッジを行うには、JMS アクティベーション指定が必要です。

##### 前提条件

- IBM WebSphere Application Server にサービスバスを作成している。
- JMS キューを作成している。

##### 手順

1. WebSphere 統合ソリューションコンソールで、**Resources → JMS → Activation Specifications** の順に移動します。
2. 正しいスコープを選択し、**New** をクリックします。
3. **Default Messaging Provider** オプションを選択し、**OK** をクリックします。
4. 以下に示す必須のアクティベーション指定に、アクティベーション指定名 (例: **KIE.SERVER.REQUEST**) および JNDI 名 (例: **jms/activation/KIE.SERVER.REQUEST**) を入力して、**Bus Name** ドロップダウンリストからサービスバス名を選択します。
5. **Destination Type** ドロップダウンリストで **Queue** を選択し、**Destination lookup** に対応するキューの名前 (例: **jms/KIE.SERVER.REQUEST**) を入力します。
6. **Apply**、**Save** の順にクリックして プライマリー設定への変更を保存し、必要な各アクティベーション指定に上述の手順を繰り返します。

##### 45.2.4.1. KIE Server の JMS アクティベーション仕様

以下の表は、KIE Server で JMS メッセージングを有効にするために必要な Java Message Service (JMS) アクティベーション仕様になります。

表45.3 KIE Server に必要な JMS アクティベーション仕様



名前	デフォルト値	用途
<b>KIE.SERVER.REQUEST</b>	<b>jms/activation/KIE.SERVER.REQUEST</b>	全要求を KIE Server へ送信
<b>KIE.SERVER.RESPONSE</b>	<b>jms/activation/KIE.SERVER.RESPONSE</b>	KIE Server が生成する応答をすべて受信

### 45.3. IBM WEBSHERE APPLICATION SERVER へのシステムプロパティの設定

KIE Server をデプロイする前に、IBM WebSphere Application Server に、本セクションに記載するシステムプロパティを設定します。

#### 手順

1. WebSphere 統合ソリューションコンソールで、**Servers → Server Types → IBM WebSphere Application Servers** の順にクリックします。
2. アプリケーションサーバーの一覧で、KIE Server をデプロイするサーバーを選択します。
3. **Server Infrastructure** で **Java and Process Management → Process Definition** をクリックします。

図45.2 WebSphere 設定ページ

Application servers

[Application servers](#) > **server1**

Use this page to configure an application server. An application server is a server that provides services required to run enterprise applications.

**Runtime** Configuration

**General Properties**

Name  
server1

Node name  
dhcp-4-116Node01

☐ Run in development mode

☒ Parallel start

☐ Start components as needed

Access to internal server classes  
Allow ▼

**Server-specific Application Settings**

ClassLoader policy  
Multiple ▼

Class loading mode  
Classes loaded with parent class loader first ▼

Apply OK Reset Cancel

**Container Settings**

- Session management
- SIP Container Settings
- Web Container Settings
- Portlet Container Settings
- EJB Container Settings
- Container Services
- Business Process Services

**Applications**

- Installed applications

**Server messaging**

- Messaging engines
- Messaging engine inbound transports
- WebSphere MQ link inbound transports
- SIB service

**Server Infrastructure**

- Java and Process Management
  - Class loader
  - Process definition**
  - Process execution
- Administration
  - Java SDKs

4. Additional Properties で Java Virtual Machine をクリックします。

図45.3 プロセスの定義設定ページ

Application servers

[Application servers](#) > [server1](#) > Process definition

Use this page to configure a process definition. A process definition defines the command line information necessary to start or initialize a process.

Configuration

**General Properties**

Executable name

Executable arguments

Start command

Start command arguments

Stop command

Stop command arguments

Working directory

Executable target type  
JAVA\_CLASS

Executable target

Apply OK Reset Cancel

**Additional Properties**

- Java Virtual Machine
- Environment Entries
- Process execution
- Process Logs
- Logging and tracing

これにより、IBM WebSphere を起動するのに使用する JVM の設定プロパティーが開きます。

- Initial heap size および Maximum heap size の両方を **2048** に設定し、**Apply** をクリックして Java Virtual Machine (JVM) メモリーサイズを増やします。KIE Server はこの値でテストされています。JVM メモリーサイズを増やさないと、KIE Server のデプロイ時に IBM WebSphere Application Server がフリーズするかエラーが発生します。
- Additional Properties** で **Custom Properties** をクリックします。
- New** → **Custom JVM Properties** の順にクリックし、以下のプロパティーを IBM WebSphere に追加します。

表45.4 KIE Server のシステムプロパティー

名前	値	説明
<b>kie.server.jms.queues.response</b>	<b>jms/KIE.SERVER.RESPONSE</b>	KIE Server が使用する応答用の JMS キューの JNDI 名。
<b>org.kie.server.domain</b>	<b>WSLogin</b>	JMS の使用時にユーザーの認証に使用される JAAS <b>LoginContext</b> ドメイン。

名前	値	説明
<b>org.jbpm.server.ext.disabled</b>	<b>true</b>	RHDM でサポートされていない Business Central の機能を無効にします。設定されていない場合、KIE Server は機能しますが、起動時にエラーメッセージを表示します。
<b>org.jbpm.ui.server.ext.disabled</b>	<b>true</b>	RHDM でサポートされていない Business Central の機能を無効にします。設定されていない場合、KIE Server は機能しますが、起動時にエラーメッセージを表示します。
<b>org.jbpm.case.server.ext.disabled</b>	<b>true</b>	RHDM でサポートされていない Business Central の機能を無効にします。設定されていない場合、KIE Server は機能しますが、起動時にエラーメッセージを表示します。
<b>org.jboss.logging.provider</b>	<b>jdk</b>	このプロパティは、 <b>CA SiteMinder TAI (SMTAI)</b> が環境にインストールされている場合にのみ必要になります。このプロパティを使用すると、Dashbuilder でログを記録するために、Hibernate が <b>log4j</b> ではなく、 <b>JDK</b> を強制的に使用しようとします。 <b>CA SiteMinder TAI (SMTAI)</b> には古いバージョンの <b>log4j</b> が含まれており、これにより競合が生じます。

8. **保存** をクリックして、プライマリー設定への変更を保存します。

## 45.4. IBM WEBSPHERE APPLICATION SERVER の停止および再起動

IBM WebSphere Application Server に必要なシステムプロパティをすべて設定したあと、IBM サーバーを停止および再起動して、設定が適用されていることを確認します。

### 手順

端末で、インストール時に指定した IBM WebSphere Application Server の **/bin** ディレクトリーの場所に移動し、以下のコマンドを実行して IBM WebSphere を停止して再起動し、設定変更を適用します。

■

```
sudo ./stopServer.sh <SERVER_NAME>
```

```
sudo ./startServer.sh <SERVER_NAME>
```

**<SERVER\_NAME>** は、WebSphere 統合ソリューションコンソールの **Servers → Server Types → IBM WebSphere Application Servers** で定義した IBM WebSphere Application Server の名前に置き換えます。

## 第46章 IBM WEBSPHERE APPLICATION SERVER への KIE SERVER のインストールおよび実行

IBM WebSphere Application Server に必要なシステムプロパティをすべて設定したら、IBM WebSphere に KIE Server をインストールして、Red Hat Decision Manager アプリケーション管理を合理化します。

### 前提条件

- [45章 KIE Server への IBM WebSphere Application Server の設定](#) の説明どおりに IBM WebSphere Application Server インスタンスを設定している。

### 手順

1. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product:** Decision Manager
  - **バージョン:** 7.10
2. Red Hat Decision Manager 7.10.0 KIE Server for All Supported EE7 Containers をダウンロードします。
3. **rhdm-7.10.0-kie-server-ee7.zip** アーカイブを一時ディレクトリーに展開します。以下の例では、この名前を **TEMP\_DIR** とします。
4. **kie-server.war** ディレクトリーを再パッケージ化します。
  - a. **TEMP\_DIR/rhdm-7.10.0-kie-server-ee7/kie-server.war** ディレクトリーに移動します。
  - b. **TEMP\_DIR/rhdm-7.10.0-kie-server-ee7/kie-server.war** ディレクトリーのコンテンツを選択し、**kie-server.zip** ファイルを作成します。
  - c. **kie-server.zip** の名前を **kie-server.war** に変更します。このファイルを使用して、KIE Server をデプロイします。
  - d. 必要に応じて、新しい **kie-server.war** ファイルを、デプロイしやすい便利な場所にコピーします。
5. WebSphere 統合ソリューションコンソールで、**Applications → Application Types → WebSphere Enterprise Applications** の順に移動します。
6. **InstCII** をクリックします。
7. 再パッケージ化した **kie-server.war** ファイルに移動して選択し、アップロードします。
8. **Fast Path** を選択し、**Next** をクリックします。  
**Install New Application** ウィザードが開きます。
9. **Application Name** を **kie-server** に変更し、**Next** をクリックします。
10. 要件に合わせて KIE Server モジュールをサーバーにマッピングし、**Next** をクリックします。
11. **Bind Listeners for Message-Driven Beans** の場合は、両 Bean に **Activation Specification** を選択し、**Target Resource JNDI Name** フィールドに **jms/activation/KIE.SERVER.REQUEST**

を入力し、**KIE.SERVER.REQUEST** 接続ファクトリーの **jms/cf/KIE.SERVER.REQUEST** JNDI 名を入力します。

12. **Map Virtual Hosts for Web Modules** セクションでは、デフォルト値をそのままにして、**Next** をクリックします。
13. コンテキストのルートをも **kie-server** に設定します。
14. **Metadata for Modules** セクションで、デフォルト値をそのままにして **Next** をクリックします。
15. **Finish** をクリックして KIE Server をインストールし、**Save** をクリックしてプライマリー設定に変更を保存します。

## 46.1. KIE SERVER のグループおよびロールの作成

KIE Server をインストールしたら、**kie-server** グループと、ユーザーを作成します。

### 前提条件

- KIE Server が IBM WebSphere Application Server インスタンスにインストールされている。

### 手順

1. WebSphere 統合ソリューションコンソールで、**Users and Groups** → **Manage Groups** の順にクリックします。
2. **Manage Groups** 画面で **Create** をクリックします。
3. **Create a Group** 画面の **Group name** ボックスに **kie-server** と入力し、**Create** をクリックします。
4. **kie-server** グループに追加するユーザーを作成するには、**Users and Groups** → **Manage Users** の順にクリックします。
5. **Create a User** セクションで、必要な情報を入力します。
6. **Group Membership** をクリックします。
7. **Group Membership** 画面で **kie-server** をクリックして **Mapped To** に移動し、**Close** をクリックします。
8. **Create a User** 画面で **Create** をクリックします。

## 46.2. KIE SERVER のグループおよびロールのマッピング

KIE Server をインストールしたら、WebSphere 統合ソリューションコンソールで **kie-server** ロールを **kie-server** グループにマッピングして、KIE Server を実行します。

### 前提条件

- KIE Server が IBM WebSphere Application Server インスタンスにインストールされている。
- IBM WebSphere Application Server に、1人以上のユーザーが追加されている **kie-server** グループがある。

## 手順

1. WebSphere 統合ソリューションコンソールで、**Applications → Application Types → WebSphere Enterprise Applications** に移動し、新たにインストールした **kie-server** アプリケーションを選択します。
2. **Detail Properties** で **Security Role to User/Group Mapping** をクリックします。
3. **kie-server** ロールを選択し、**Map Groups** をクリックして、**kie-server** グループを検索します。
4. **kie-server** グループを、**Available** 一覧から **Selected** 一覧に移動し、**OK** をクリックします。このマッピングにより、IBM WebSphere Application Server の **kie-server** グループのユーザーに、KIE Server へのアクセスが付与されます。
5. **Save** をクリックしてマッピングを完了します。

## 46.3. KIE SERVER のクラ出力ディニングの設定

KIE Server をインストールしたら、親クラスを最後に読み込むように、クラ出力ディニングを設定する必要があります。

## 手順

1. **Applications → Application Types → WebSphere Enterprise Applications** に移動し、**kie-server** をクリックします。
2. 左側の **Detail Properties** の見出しにある **Class Loading and Update Detection** をクリックします。
3. プロパティーで、**Class Loader Order** を **Classes loaded with local class loader first (parent last)** に、**WAR Class Loader Policy** を **Single class loader for application** に変更します。
4. 変更をプライマリー設定に保存します。

## 46.4. インストールの検証

KIE Server をインストールして KIE Server グループマッピングを定義したら、サーバーが実行していることを確認します。

## 前提条件

- KIE Server が IBM WebSphere Application Server インスタンスにインストールされている。
- ヘッドレス Decision Manager コントローラーに必要なシステムプロパティーをすべて設定している。
- IBM WebSphere Application Server に KIE Server グループマッピングを定義している。

## 手順

1. サーバーが実行していることを確認するには、以下のタスクのいずれかを実行します。
  - KIE Server の URL (**http://<HOST>:<PORT>/kie-server**) に移動します。



- **http://<HOST>:<PORT>/kie-server/services/rest/server** に **GET** 要求を送信し、KIE Server REST API が応答するかどうかを確認します。  
上記の例では、以下のプレースホルダーを置き換えてください。
  - **<HOST>** は、KIE Server ホストの ID または名前に置き換えます (例: **localhost** または **192.7.8.9**)。
  - **<PORT>** は、KIE Server ホストのポート番号に置き換えます (例: **9060**)。
- 2. KIE Server が実行していない場合は、IBM WebSphere Application Server インスタンスを停止して再起動し、KIE Server の URL または API に再度アクセスしてみてください。

## 第47章 IBM WEBSPHERE APPLICATION SERVER へのヘッドレス DECISION MANAGER コントローラーのインストールおよび実行

KIE Server REST API または Java Client API を使用して KIE Server に接続するには、IBM WebSphere Application Server にヘッドレス Decision Manager コントローラーをインストールします。ヘッドレス Decision Manager コントローラーは、一元化された方法で KIE Server 設定を管理するため、このコントローラーを使用してコンテナの作成および維持を行い、他のサーバーレベルのタスクを実行することができます。



### 注記

実稼働環境で最適な結果を得るには、KIE Server とヘッドレス Decision Manager コントローラーを別のサーバーにインストールします。開発環境の場合は、KIE Server とヘッドレス Decision Manager コントローラーを同じサーバーにインストールします。

### 前提条件

- [45章 KIE Server への IBM WebSphere Application Server の設定](#) の説明どおりに IBM WebSphere Application Server インスタンスを設定している。
- KIE Server が IBM WebSphere Application Server インスタンスにインストールされている。
- インストールを完了するのに必要なユーザーパーミッションがある。

### 手順

1. Red Hat カスタマーポータルでの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product:** Decision Manager
  - **バージョン:** 7.10
2. Red Hat Decision Manager 7.10.0 Add-Ons をダウンロードします。
3. ダウンロードした **rhdm-7.10.0-add-ons.zip** ファイルを一時ディレクトリーに展開します。
4. **rhdm-7.10.0-add-ons.zip** ファイルを一時ディレクトリーに展開します。以下の例では、この名前を **TEMP\_DIR** とします。
5. **rhdm-7.10.0-add-ons/rhdm-7.10.0-controller-ee7.zip** ファイルを展開します。
6. **controller.war** ディレクトリーを再パッケージ化します。
  - a. **TEMP\_DIR/rhdm-7.10.0-add-ons/rhdm-7.10.0-controller-ee7/controller.war** ディレクトリーに移動します。
  - b. **TEMP\_DIR/rhdm-7.10.0-add-ons/rhdm-7.10.0-controller-ee7/controller.war** ディレクトリーのコンテンツを選択して、**controller.zip** ファイルを作成します。
  - c. **controller.zip** の名前を **controller.war** に変更します。このファイルを使用して、ヘッドレス Decision Manager コントローラーをデプロイします。

- d. 必要に応じて、新しい **controller.war** ファイルを、デプロイしやすい場所にコピーします。
7. WebSphere 統合ソリューションコンソールで、**Applications → Application Types → WebSphere Enterprise Applications** の順に移動します。
8. **Install** をクリックします。
9. 再パッケージ化した **controller.war** ファイルに移動し、選択してアップロードします。
10. **Fast Path** を選択し、**Next** をクリックします。  
**Install New Application** ウィザードが開きます。
11. **Application Name** を **controller** に変更し、**Next** をクリックします。
12. 要件に合わせて ヘッドレス Decision Manager コントローラーモジュールをサーバーにマッピングし、**Next** をクリックします。
13. **Bind Listeners for Message-Driven Beans** の場合は、両 Bean に **Activation Specification** を選択し、**Target Resource JNDI Name** フィールドに **jms/activation/KIE.SERVER.REQUEST** を入力し、**KIE.SERVER.REQUEST** 接続ファクトリーの **jms/cf/KIE.SERVER.REQUEST** JNDI 名を入力します。
14. **Map Virtual Hosts for Web Modules** セクションでは、デフォルト値をそのままにして、**Next** をクリックします。
15. コンテキストルートを **controller** に設定します。
16. **Metadata for Modules** セクションで、デフォルト値をそのままにして **Next** をクリックします。
17. **Finish** をクリックしてヘッドレス Decision Manager コントローラーをインストールし、**Save** をクリックしてプライマリー設定への変更を保存します。

## 47.1. ヘッドレス DECISION MANAGER コントローラーへのクラ出力ディニングの設定

ヘッドレス Decision Manager コントローラーをインストールしたら、親クラスを最後に読み込むように、KIE Server クラ出力ディニングを設定する必要があります。

### 手順

1. **Applications → Application Types → WebSphere Enterprise Applications** に移動し、**kie-server** をクリックします。
2. 左側の **Detail Properties** の見出しにある **Class Loading and Update Detection** をクリックします。
3. プロパティーで、**Class Loader Order** を **Classes loaded with local class loader first (parent last)** に、**WAR Class Loader Policy** を **Single class loader for application** に変更します。
4. 変更をプライマリー設定に保存します。

## 47.2. ヘッドレス DECISION MANAGER コントローラーのシステムプロパティーの設定

ヘッドレス Decision Manager コントローラーをインストールしたら、本セクションで紹介するシステムプロパティをアプリケーションサーバーに設定して、ヘッドレス Decision Manager コントローラーとの適切な接続を有効にします。



## 注記

実稼働環境で最適な結果を得るには、KIE Server とヘッドレス Decision Manager コントローラーを別のサーバーにインストールします。開発環境の場合は、KIE Server とヘッドレス Decision Manager コントローラーを同じサーバーにインストールします。いずれの場合も、ヘッドレス Decision Manager コントローラーがインストールされているすべてのアプリケーションサーバーでこのプロパティを変更します。

## 前提条件

- KIE Server およびヘッドレス Decision Manager コントローラーが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る。

## 手順

1. ヘッドレス Decision Manager コントローラーがインストールされているアプリケーションサーバーインスタンスに、以下の JVM プロパティ値を指定します。

表47.1 ヘッドレス Decision Manager コントローラーに必要なプロパティ

名前	要件
<b>org.kie.server.user</b>	<b>kie-server</b> ロールが割り当てられているユーザー
<b>org.kie.server.pwd</b>	<b>org.kie.server.user</b> プロパティに指定したユーザーのパスワード

2. KIE Server がインストールされているアプリケーションサーバーインスタンスに、以下の JVM プロパティ値を指定します。

表47.2 ヘッドレス Decision Manager コントローラーがインストールされている場合に KIE Server に必要なプロパティ

名前	要件
<b>org.kie.server.controller.user</b>	<b>kie-server</b> ロールが割り当てられているユーザー
<b>org.kie.server.controller.pwd</b>	<b>org.kie.server.controller.user</b> プロパティに指定したユーザーのパスワード
<b>org.kie.server.id</b>	KIE Server インストールの ID または名前 ( <b>rhdm700-decision-server-1</b> など)
<b>org.kie.server.location</b>	KIE Server の URL ( <b>http://&lt;HOST&gt;:&lt;PORT&gt;/kie-server/services/rest/server</b> )

名前	要件
<b>org.kie.server.controller</b>	ヘッドレス Decision Manager コントローラーの URL ( <a href="http://&lt;HOST&gt;:&lt;PORT&gt;/controller/rest/controller">http://&lt;HOST&gt;:&lt;PORT&gt;/controller/rest/controller</a> )

**<HOST>** は、KIE Server ホストの ID または名前に置き換えます (例: **localhost** または **192.7.8.9**)。

**<PORT>** は、KIE Server ホストのポート番号に置き換えます (例: **7001**)。

### 47.3. ヘッドレス DECISION MANAGER コントローラーのグループおよびロールのマッピング

ヘッドレス Decision Manager コントローラーをインストールしたら、WebSphere 統合ソリューションコンソールに **kie-server** グループを作成し、**kie-server** ロールを **kie-server** グループにマッピングします。

#### 前提条件

- ヘッドレス Decision Manager コントローラーが IBM WebSphere Application Server インスタンスにインストールされている。

#### 手順

1. WebSphere 統合ソリューションコンソールで、[KIE Server グループおよびロールの作成](#)の記載通りに、**kie-server** グループと、このグループのユーザーを作成します。
2. **Applications** → **Application Types** → **WebSphere Enterprise Applications** に移動し、新たにインストールしたヘッドレス Decision Manager コントローラーを選択します。
3. **Detail Properties** で **Security Role to User/Group Mapping** をクリックします。
4. **kie-server** ロールを選択し、**Map Groups** をクリックして、**kie-server** グループを検索します。
5. **kie-server** グループを、**Available** 一覧から **Selected** 一覧に移動し、**OK** をクリックします。  
このマッピングにより、事前に作成した管理者ユーザーに Decision Manager へのアクセスが付与されます。
6. **Save** をクリックしてマッピングを完了します。

### 47.4. インストールの検証

ヘッドレス Decision Manager コントローラーをインストールし、必要なシステムプロパティとロール要件をアプリケーションサーバーに定義したら、ヘッドレス Decision Manager コントローラーが正しく動作していることを確認します。

#### 前提条件

- KIE Server およびヘッドレス Decision Manager コントローラーが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る。

- アプリケーションサーバーで、ヘッドレス Decision Manager コントローラーに必要なシステムプロパティおよびロール要件をすべて設定している。

## 手順

端末で以下のコマンドを実行して、ヘッドレス Decision Manager コントローラーが動作していることを確認します。

```
curl -X GET "http://<HOST>:<PORT>/controller/rest/controller/management/servers" -H "accept: application/xml" -u '<CONTROLLER>:<CONTROLLER_PWD>'
```

<HOST> は、KIE Server ホストの ID または名前に置き換えます (例: **localhost** または **192.7.8.9**)。

<PORT> は、KIE Server ホストのポート番号に置き換えます (例: **7001**)。

<CONTROLLER> および <CONTROLLER\_PWD> は、本セクションで作成したユーザーの認証情報です。

このコマンドにより、KIE Server インスタンスに関する情報が返されます。



### 注記

別の方法として、KIE Server Java API Client を使用してヘッドレス Decision Manager コントローラーにアクセスすることもできます。

ヘッドレス Decision Manager コントローラーが実行していない場合は、アプリケーションサーバーインスタンスを停止して再起動し、ヘッドレス Decision Manager コントローラーの URL または API へ再度アクセスしてみてください。

## 第48章 IBM WEBSPPHERE APPLICATION SERVER での埋め込みデシジョンエンジンの設定

デシジョンエンジンとは、デシジョンとビジネスプロセスを実行できるようにする軽量のルールエンジンです。デシジョンエンジンは、Red Hat Decision Manager アプリケーションに含めるか、OpenShift、Kubernetes、および Docker 経由でサービスとしてデプロイできます。また、API を使用して Red Hat Decision Manager アプリケーションに埋め込むか、または contexts and dependency injection (CDI) サービスセットの一部として、デシジョンエンジンを埋め込むことができます。

Red Hat Decision Manager アプリケーションで、埋め込まれたエンジンを使用する予定の場合には、Red Hat Business Automation 部品表 (BOM) ファイルをプロジェクトの **pom.xml** ファイルに追加して、プロジェクトに、Maven の依存関係を追加する必要があります。Red Hat Business Automation BOM は、Red Hat Decision Manager に適用されます。Red Hat Business Automation BOM (Bill of Materials) についての詳細情報は、[What is the mapping between Red Hat Decision Manager and the Maven library version?](#) を参照してください。

### 手順

1. Red Hat Business Automation BOM を **pom.xml** ファイルで宣言します。

```
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>com.redhat.ba</groupId>
      <artifactId>ba-platform-bom</artifactId>
      <version>7.10.0.redhat-00004</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
<dependencies>
  <!-- Your dependencies -->
</dependencies>
```

2. **<dependencies>** タグでお使いのプロジェクトに必要な依存関係を宣言します。製品の BOM をプロジェクトにインポートしたら、ユーザー向け製品依存関係のバージョンが定義されるため、**<dependency>** 要素のサブ要素 **<version>** を指定する必要はありません。ただし、**<dependency>** 要素を使用して、プロジェクトで使用する依存関係を宣言する必要があります。
- 標準的な Red Hat Decision Manager プロジェクトでは、使用する機能に応じて、以下の依存関係を宣言します。

### 埋め込みデシジョンエンジン依存関係

```
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-compiler</artifactId>
</dependency>

<!-- Dependency for persistence support. -->
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-persistence-jpa</artifactId>
```

```
</dependency>
```

```
<!-- Dependencies for decision tables, templates, and scorecards.
For other assets, declare org.drools:business-central-models-* dependencies. -->
```

```
<dependency>
```

```
  <groupId>org.drools</groupId>
```

```
  <artifactId>drools-decisiontables</artifactId>
```

```
</dependency>
```

```
<dependency>
```

```
  <groupId>org.drools</groupId>
```

```
  <artifactId>drools-templates</artifactId>
```

```
</dependency>
```

```
<dependency>
```

```
  <groupId>org.drools</groupId>
```

```
  <artifactId>drools-scorecards</artifactId>
```

```
</dependency>
```

```
<!-- Dependency for loading KJARs from a Maven repository using KieScanner. -->
```

```
<dependency>
```

```
  <groupId>org.kie</groupId>
```

```
  <artifactId>kie-ci</artifactId>
```

```
</dependency>
```

- KIE Server を使用するには、以下の依存関係を宣言します。

#### クライアントアプリケーション KIE Server の依存関係

```
<dependency>
```

```
  <groupId>org.kie.server</groupId>
```

```
  <artifactId>kie-server-client</artifactId>
```

```
</dependency>
```

- Red Hat Decision Manager にリモートクライアントを作成するには、以下の依存関係を宣言します。

#### クライアントの依存関係

```
<dependency>
```

```
  <groupId>org.uberfire</groupId>
```

```
  <artifactId>uberfire-rest-client</artifactId>
```

```
</dependency>
```

- ルール、プロセス定義など、アセットを含む JAR ファイルを作成する場合は、お使いの Maven プロジェクトのパッケージングの種類を **kjar** と指定し、**org.kie:kie-maven-plugin** を使用して、**<project>** 要素に置かれた **kjar** パッケージングタイプを処理します。以下の例の **\${kie.version}** は、[What is the mapping between Red Hat Decision Manager and the Maven library version?](#) に記載されている Maven ライブラリーのバージョンです。

```
<packaging>kjar</packaging>
```

```
<build>
```

```
  <plugins>
```

```
    <plugin>
```

```
      <groupId>org.kie</groupId>
```

```
      <artifactId>kie-maven-plugin</artifactId>
```

```
      <version>${kie.version}</version>
```



```

    <extensions>true</extensions>
  </plugin>
</plugins>
</build>

```

- プロジェクトで永続サポートのあるデシジョンエンジンを使用する場合は、Red Hat Business Automation BOM ファイルから **version.org.hibernate-4ee7** プロパティをコピーして、**pom.xml** ファイルの **dependencyManagement** セクションに以下のハイバーネートの依存関係を宣言する必要があります。

### 永続性のあるデシジョンエンジンにおけるハイバーネートの依存関係

```

<!-- hibernate dependencies -->
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.hibernate</groupId>
      <artifactId>hibernate-entitymanager</artifactId>
      <version>${version.org.hibernate-4ee7}</version>
    </dependency>

    <dependency>
      <groupId>org.hibernate</groupId>
      <artifactId>hibernate-core</artifactId>
      <version>${version.org.hibernate-4ee7}</version>
    </dependency>
  </dependencies>
</dependencyManagement>

```

## 第49章 キーストアを使用したパスワードセキュリティの確保

キーストアを使用して、Business Central と KIE Server の間の通信に使用するパスワードを暗号化できます。コントローラーと KIE Server のパスワードを暗号化する必要があります。Business Central と KIE Server を別のアプリケーションサーバーにデプロイする場合は、いずれのアプリケーションサーバーもキーストアを使用する必要があります。

Java Cryptography Extension KeyStore (JCEKS) は、対称鍵をサポートするため、キーストアには JCEKS を使用してください。



### 注記

KIE Server が JCEKS で設定されていない場合、KIE Server のパスワードはシステムプロパティにプレーンテキスト形式で保存されます。

### 前提条件

- KIE Server が IBM WebSphere Application Server にインストールされている。
- 「[KIE Server のグループおよびロールの作成](#)」の記載通りに、**kie-server** ロールを割り当てた KIE Server ユーザーが作成されている。
- Java 8 以降がインストールされている。

### 手順

1. JCEKS キーストアを作成します。
2. プロンプトが表示されたら、作成した KIE Server ユーザーのパスワードを入力します。
3. 以下の表に記載のシステムプロパティを設定します。

表49.1 KIE Server JCEKS を読み込む時に使用するシステムプロパティ

システムプロパティ	プレースホルダー	説明
<b>kie.keystore.keyStoreURL</b>	<b>&lt;KEYSTORE_URL&gt;</b>	使用する JCEKS の URL (例: <b>file:///home/kie/keystores/keystore.jceks</b> )
<b>kie.keystore.keyStorePwd</b>	<b>&lt;KEYSTORE_PWD&gt;</b>	JCEKS のパスワード
<b>kie.keystore.key.server.alias</b>	<b>&lt;KEY_SERVER_ALIAS&gt;</b>	パスワードの保存先となる REST サービスのキーのエイリアス
<b>kie.keystore.key.server.pwd</b>	<b>&lt;KEY_SERVER_PWD&gt;</b>	保存したパスワードを使用する REST サービスのエイリアスのパスワード
<b>kie.keystore.key.ctrl.alias</b>	<b>&lt;KEY_CONTROL_ALIAS&gt;</b>	パスワードの保存先のデフォルトの REST Process Automation Controller のキーのエイリアス

システムプロパティ	プレースホルダー	説明
<b>kie.keystore.key.ctrl.pwd</b>	<b>&lt;KEY_CONTROL_PWD&gt;</b>	保存したパスワードを使用する、デフォルトの REST Process Automation Controller のエイリアスのパスワード

4. KIE Server を起動して、設定を検証します。

## 第50章 次のステップ

- [デシジョンサービスのスタートガイド](#)
- [ガイド付きデシジョンテーブルを使用したデシジョンサービスの作成](#)

## パート VI. ORACLE WEBLOGIC SERVER への KIE SERVER のインストールおよび設定

システム管理者は、Red Hat KIE Server に Oracle WebLogic Server を設定し、Oracle サーバーインスタンスに KIE Server をインストールできます。

### 前提条件

- Oracle WebLogic Server インスタンスのバージョン 12.2.1.3.0 以降がインストールされている。詳細なインストール手順は [Oracle WebLogic Server 製品ページ](#) を参照してください。
- Oracle WebLogic Server 管理コンソール (通常は **http://<HOST>:7001/console**) へのアクセスがある。

## 第51章 KIE SERVER

KIE Server は、Red Hat Decision Manager のルールおよびその他のアーティファクトが保存され、実行されるサーバーです。KIE Server はスタンドアロンで組み込みのコンポーネントで、REST、Java Message Service (JMS)、または Java クライアントサイドアプリケーションで利用可能なインターフェイス、およびソルバーを介した Red Hat Business Optimizer 機能を使用して、ルールのインスタンスを作成して実行することができます。

Web でデプロイ可能な WAR ファイルとして作成することで、KIE Server は Web コンテナであればどこにでもデプロイできます。KIE Server の現在のバージョンには、Red Hat Decision Manager および Red Hat Process Automation Manager の両方に対するデフォルトの拡張機能が含まれます。

KIE Server はメモリー消費が最小限でフットプリントが小さいため、クラウドインスタンスに簡単にデプロイできます。このサーバーの各インスタンスでは、複数のコンテナを開いてインスタンスを作成できるため、並行して複数のルールサービスを実行できます。

KIE Server は、Oracle WebLogic Server、IBM WebSphere Application Server などのアプリケーションサーバーと統合して、Red Hat Decision Manager のアプリケーション管理を合理化できます。

## 第52章 ORACLE WEBLOGIC SERVER

Oracle WebLogic Server は、分散型 Java アプリケーションを作成するために API の標準セットを提供する Java EE アプリケーションサーバーで、データベース、メッセージングサービス、外部のエンタープライズシステムへの接続など、さまざまなサービスにアクセスできます。ユーザークライアントは、Web ブラウザークライアントまたは Java クライアントを使用してこれらのアプリケーションにアクセスします。

## 第53章 ORACLE WEBLOGIC SERVER のインストールおよび実行

KIE Server に対応する多くの設定を適用するために、Oracle WebLogic Server をインストールして実行する必要があります。本セクションは、スタンドアロンの Oracle WebLogic Server ドメインに Oracle WebLogic Server をインストールして起動する方法を説明します。

最新の詳細情報は [Oracle WebLogic Server 製品ページ](#) を参照してください。



### 注記

起動するサーバーと同じリスナーポートを使用する Oracle WebLogic Server インスタンスを実行している場合は、新たにサーバーを起動する前に、稼働しているサーバーを停止する必要があります。

### 手順

1. Oracle WebLogic Server の [ダウンロードページ](#) から、Oracle WebLogic Server 12.2.1.3.0 以降をダウンロードします。
2. ターゲットシステムにサインインし、認定済み JDK がシステムにインストールされていることを確認します。インストーラーには認定済み JDK が必要です。システム要件は、[Oracle Fusion Middleware Systems Requirements and Specifications](#) を参照してください。JDK のダウンロードは、[Planning an Installation of Oracle Fusion Middleware](#) の About JDK Requirements for an Oracle Fusion Middleware Installation のセクションを参照してください。
3. インストールプログラムをダウンロードしたディレクトリーに移動します。
4. 以下の例のように、システムの JDK ディレクトリーから **java -jar** を実行して、インストールプログラムを起動します。  
UNIX ベースのオペレーティングシステムの場合は、以下のコマンドを入力します。

```
/home/Oracle/jdk/jdk1.8.0_131/bin/java -jar fmw_12.2.1.3.0_wls_generic.jar
```

Windows オペレーティングシステムの場合は、以下のコマンドを入力します。

```
C:\Program Files\Java\jdk1.8.0_131\bin\java -jar fmw_12.2.1.3.0_wls_generic.jar
```

この例の JDK の場所は、実際にシステムに保存されている JDK の場所に置き換えます。

5. インストールウィザードプロンプトに従い、インストールを完了します。
6. インストールが完了したら、**WLS\_HOME/user\_projects/<DOMAIN\_NAME>** に移動します。**<DOMAIN\_NAME>** はドメインディレクトリーに置き換えます。以下の例では、**mydomain** はドメインディレクトリーです。

```
WLS\user_projects\mydomain
```

7. 以下のいずれかのコマンドを実行して、Oracle WebLogic Server を起動します。  
UNIX ベースのオペレーティングシステムの場合は、以下のコマンドを入力します。

```
startWebLogic.sh
```

Windows オペレーティングシステムの場合は、以下のコマンドを入力します。

■



```
startWebLogic.cmd
```

起動スクリプトが多数のメッセージを表示しますが、最後に以下のようなメッセージを表示します。

```
<Dec 8, 2017 3:50:42 PM PDT> <Notice> <WebLogicServer> <000360> <Server started in  
RUNNING mode>
```

8. Web ブラウザーで、以下の URL を開きます。

```
http://<HOST>:<PORT>/console
```

このコマンドでは、以下のプレースホルダーを置き換えてください。

- **<HOST>** は、ホストサーバーのシステム名または IP アドレスに置き換えます。
- **<PORT>** は、ホストサーバーが要求をリッスンしているポートの番号 (デフォルトでは 7001) に置き換えます。

たとえば、システムで起動している Oracle WebLogic Server のローカルインスタンスに対して管理コンソールを起動するには、以下の URL をブラウザに入力します。

```
http://localhost:7001/console/
```

SSL (secure socket layer) を使用して管理コンソールを起動したら、**http** の後に **s** を追加して、**https://<HOST>:<PORT>/console** にする必要があります。

9. WebLogic 管理コンソールのログインページが表示されたら、管理者の認証情報を入力します。

## 第54章 KIE SERVER への ORACLE WEBLOGIC SERVER の設定

Oracle WebLogic Server に KIE Server をデプロイする前に、システムプロパティ、セキュリティ設定、JMS 要件、その他の Oracle WebLogic Server プロパティを設定する必要があります。この設定により、KIE Server との最適な統合が促進されます。

### 前提条件

- Oracle WebLogic Server をインストールし、実行している。
- WebLogic 管理コンソールにログインしている。

### 54.1. KIE SERVER のグループおよびユーザーの設定

WebLogic 管理コンソールでユーザーを **kie-server** グループに割り当て、Oracle WebLogic Server で、コンテナ管理の認証メカニズムを有効にします。

#### 手順

1. WebLogic 管理コンソールで **Security Realms** に移動します。
2. 必要なセキュリティレルムを選択するか、**New** をクリックして新しいセキュリティレルムを作成します。
3. **Users and Groups** → **Groups** → **New** の順に移動して、**kie-server** グループを作成します。
4. **Users** → **New** の順に移動し、新しいユーザーを作成します。
5. **server-user** などのユーザーとそのパスワードを入力し、**OK** をクリックします。



#### 重要

選択したユーザー名が、ロールまたはグループの名前と競合しないようにします。たとえば、**kie-server** という名前のロールが存在する場合は、**kie-server** という名前のユーザーを作成しないでください。

6. 新たに作成したユーザーをクリックして、**Groups** タブに戻ります。
7. 選択ツールを使用して、**kie-server** グループを **Available** フィールドから **Chosen** フィールドに移動し、**Save** をクリックします。

### 54.2. JAVA MESSAGE SERVICE (JMS) の設定

Java Message Service (JMS) は、KIE Server が、Oracle WebLogic Server、IBM WebSphere Application Server などのアプリケーションサーバーとメッセージを交換するために使用する Java API です。KIE Server を経由して JMS メッセージを送受信するようにアプリケーションサーバーを設定し、2 台のサーバー間で連携されるようにします。

#### 54.2.1. JMS サーバーの作成

KIE Server および Oracle WebLogic Server で JMS を使用する JMS サーバーを作成します。

#### 手順

1. WebLogic 管理コンソールで、**Services → Messaging → JMS Servers** の順に移動します。
2. **New** をクリックして、JMS サーバーを新たに作成します。
3. JMS サーバーの名前を入力して、**Next** をクリックします。
4. KIE Server デプロイメントに使用するターゲットサーバーを選択します。
5. **Finish** をクリックします。

### 54.2.2. JMS モジュールの作成

接続ファクトリー、キューなどの JMS リソースを格納する JMS モジュールを作成する必要があります。

#### 前提条件

- JMS サーバーを作成している。

#### 手順

1. WebLogic 管理コンソールで、**Services → Messaging → JMS Modules** の順に移動します。
2. **New** をクリックしてモジュールを作成します。
3. モジュール名を入力し、**Next** をクリックします。
4. KIE Server デプロイメントに使用するターゲットサーバーを選択し、**Finish** をクリックします。
5. 新たに作成したモジュール名をクリックし、**Subdeployments** をクリックします。
6. **New** をクリックして、モジュールのサブデプロイメントを作成します。
7. サブデプロイメントの名前を入力し、**Next** をクリックします。
8. チェックボックスを選択して、事前に作成した JMS サーバーを選択します。
9. **Finish** をクリックして、サブデプロイメントの設定を完了します。

### 54.2.3. JMS 接続ファクトリーの作成

KIE Server でメッセージングを有効にするには、メッセージの送受信に JMS 接続ファクトリーをいくつか作成する必要があります。

#### 前提条件

- JMS サーバーを作成している。
- JMS モジュールを作成している。

#### 手順

1. WebLogic 管理コンソールで、**Services → Messaging → JMS Modules** の順に移動して、JMS モジュールの一覧を確認します。

2. 事前に作成したモジュールを選択し、**New** をクリックして新しい JMS リソースを作成します。
3. **Connection Factory** を選択し、**Next** をクリックします。
4. 以下の表に記載されている必須の各接続ファクトリーに、接続ファクトリー名 (例: **KIE.SERVER.REQUEST**) および JNDI 名 (例: **jms/cf/KIE.SERVER.REQUEST**) を入力し、**Next** をクリックします。接続ファクトリーが、JMS モジュールにデフォルトで割り当てるサーバーを自動的に選択します。

表54.1 KIE Server に必要な JMS 接続ファクトリー

名前	デフォルト値	用途
<b>KIE.SERVER.REQUEST</b>	<b>jms/cf/KIE.SERVER.REQUEST</b>	全要求を KIE Server へ送信
<b>KIE.SERVER.RESPONSE</b>	<b>jms/cf/KIE.SERVER.RESPONSE</b>	KIE Server が生成する応答をすべて受信

5. **Finish** をクリックして接続ファクトリーを追加し、必要な各ファクトリーに対して上記手順を繰り返します。

#### 54.2.4. JMS キューの作成

JMS キューは、ポイントツーポイントメッセージング (point-to-point messaging) の宛先エンドポイントになります。KIE Server で JMS メッセージングを有効にするには JMS キューをいくつか作成する必要があります。

##### 前提条件

- JMS サーバーを作成している。
- JMS モジュールを作成している。

##### 手順

1. WebLogic 管理コンソールで、**Services → Messaging → JMS Modules** の順に移動して、JMS モジュールの一覧を確認します。
2. 事前に作成したモジュールを選択し、**New** をクリックして新規 JMS リソースを作成します。
3. **Queue** を選択し、**Next** をクリックします。
4. 以下の表に示す必須の各キューに、キューの名前 (例: **KIE.SERVER.REQUEST**) および JNDI 名 (例: **jms/KIE.SERVER.REQUEST**) を入力して、**Next** をクリックします。

表54.2 KIE Server に必要な JMS キュー

名前	デフォルト値	用途
<b>KIE.SERVER.REQUEST</b>	<b>jms/KIE.SERVER.REQUEST</b>	全要求を KIE Server へ送信

名前	デフォルト値	用途
<b>KIE.SERVER.RESPONSE</b>	<b>jms/KIE.SERVER.RESPONSE</b>	KIE Server が生成する応答をすべて受信

5. JMS サーバーに接続する JMS モジュールのサブデプロイメントを選択します。
6. **Finish** をクリックしてキューを追加し、必要な各キューに上記手順を繰り返します。

### 54.3. ORACLE WEBLOGIC SERVER へのシステムプロパティの設定

KIE Server をデプロイする前に、Oracle WebLogic Server に、本セクションに記載するシステムプロパティを設定します。

#### 手順

1. Java Virtual Machine (JVM) のメモリーサイズを増やすには、以下のシステムプロパティを設定します。

```
USER_MEM_ARGS=-Xms512m -Xmx1024m
```

JVM メモリーサイズを増やさないと、KIE Server のデプロイ時に Oracle WebLogic Server がフリーズするか、エラーが発生します。

2. Oracle WebLogic Server インスタンスに、以下の KIE Server システムプロパティを指定します。

表54.3 KIE Server のシステムプロパティ

名前	値	説明
<b>kie.server.jms.queues.response</b>	<b>jms/KIE.SERVER.RESPONSE</b>	KIE Server が使用する応答用の JMS キューの JNDI 名。
<b>org.kie.server.domain</b>	<b>OracleDefaultLoginConfiguration</b>	JMS の使用時にユーザーの認証に使用される JAAS <b>LoginContext</b> ドメイン。
<b>org.jbpm.server.ext.disabled</b>	<b>true</b>	RHDM でサポートされていない Business Central の機能を無効にします。設定されていない場合、KIE Server は機能しますが、起動時にエラーメッセージを表示します。

名前	値	説明
<b>org.jbpm.ui.server.ext.disabled</b>	<b>true</b>	RHDM でサポートされていない Business Central の機能を無効にします。設定されていない場合、KIE Server は機能しますが、起動時にエラーメッセージを表示します。
<b>org.jbpm.case.server.ext.disabled</b>	<b>true</b>	RHDM でサポートされていない Business Central の機能を無効にします。設定されていない場合、KIE Server は機能しますが、起動時にエラーメッセージを表示します。

3. **JAVA\_OPTIONS** 環境編集に同じプロパティ値を設定します。

```

JAVA_OPTIONS="-Dkie.server.jms.queues.response=jms/KIE.SERVER.RESPONSE
-Dorg.kie.server.domain=OracleDefaultLoginConfiguration
-Dorg.jbpm.server.ext.disabled=true
-Dorg.jbpm.ui.server.ext.disabled=true
-Dorg.jbpm.case.server.ext.disabled=true"

```

## 54.4. ORACLE WEBLOGIC SERVER の停止および再起動

Oracle WebLogic Server で必要なシステムプロパティをすべて設定したら、Oracle サーバーを停止して再起動し、設定が適用されていることを確認します。

### 手順

1. WebLogic 管理コンソールで、**Change Center** → **Lock & Edit** の順に移動します。
2. **Domain Structure** で **Environment** → **Servers** → **Control** の順にクリックします。
3. 停止するサーバーを選択して、**Shutdown** をクリックします。
4. **When Work Completes** を選択してサーバーを正常に停止するか、**Force Shutdown Now** を選択して継続中のタスクの完了を待たずに直ちにサーバーを停止します。
5. **Server Life Cycle Assistant** ペインで **Yes** をクリックして、シャットダウンを完了します。
6. シャットダウンが完了したら、端末でドメインディレクトリーに移動し、**WLS\_HOME/user\_projects/<DOMAIN\_NAME>** に移動します。以下に例を示します。

```
WLS\user_projects\mydomain
```

7. 以下のいずれかのコマンドを実行して Oracle WebLogic Server を再起動し、新しい設定を適用します。

UNIX ベースのシステムの場合:

```
startWebLogic.sh
```

Windows オペレーティングシステムの場合:

```
startWebLogic.cmd
```

8. Web ブラウザーで管理コンソール (例: **<http://localhost:7001/console/>**) を開き、認証情報を使用してログインします。

## 第55章 ORACLE WEBLOGIC SERVER への KIE SERVER のインストール

Oracle WebLogic Server に必要なシステムプロパティをすべて設定したら、Oracle WebLogic Server に KIE Server をインストールして、Red Hat Decision Manager アプリケーション管理を合理化します。

### 前提条件

- [54章 KIE Server への Oracle WebLogic Server の設定](#) の説明どおりに、Oracle WebLogic Server インスタンスが設定されている。

### 手順

1. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product:** Decision Manager
  - **バージョン:** 7.10
2. Red Hat Decision Manager 7.10.0 KIE Server for All Supported EE7 Containers をダウンロードします。
3. ダウンロードした **rhdm-7.10.0-kie-server-ee7.zip** ファイルを一時ディレクトリーに展開します。
4. WebLogic 管理コンソールで **Deployments** に移動し、既存のアプリケーションをすべて表示します。
5. **Install** をクリックします。
6. **rhdm-7.10.0-kie-server-ee7.zip** ファイルをダウンロードして展開した一時ディレクトリーに移動し、**rhdm-7.10.0-kie-server-ee7/kie-server.war** に移動します。
7. **kie-server.war** ファイルを選択し、**Next** をクリックして続行します。
8. ターゲットとするスタイルとして **Install this deployment as an application** を選択し、**Next** をクリックします。
9. アプリケーション名を **kie-server** に設定し、セキュリティーモデルを **DD Only** に設定します。残りのオプションはデフォルトのままにし、**Next** をクリックして続行します。
10. **Additional Configuration** セクションで **No, I will review the configuration later** を選択し、**Finish** をクリックします。

### 55.1. ORACLE WEBLOGIC SERVER への KIE SERVER インストールの確認

Oracle WebLogic Server に KIE Server をインストールしたら、インストールが成功していることを確認します。

### 前提条件

- [54章 KIE Server への Oracle WebLogic Server の設定](#) の説明どおりに、Oracle WebLogic Server インスタンスが設定されている。



- [55章 Oracle WebLogic Server への KIE Server のインストール](#) の記載通りに KIE Server がインストールされている。

## 手順

1. Web ブラウザーに KIE Server の URL (**http://<HOST>:<PORT>/kie-server/services/rest/server**) を入力します。
2. KIE Server が実行していることを確認します。  
KIE Server が実行していない場合は、Oracle WebLogic Server インスタンスを停止して再起動し、KIE Server の URL に再度アクセスしてみてください。

## 第56章 ORACLE WEBLOGIC SERVER へのヘッドレス DECISION MANAGER コントローラーのインストールおよび実行

KIE Server REST API または Java Client API を使用して KIE Server に接続するには、Oracle WebLogic Server にヘッドレス Decision Manager コントローラーをインストールします。ヘッドレス Decision Manager コントローラーは、一元化された方法で KIE Server 設定を管理するため、このコントローラーを使用してコンテナの作成および維持を行い、他のサーバーレベルのタスクを実行することができます。

### 前提条件

- [54章 KIE Server への Oracle WebLogic Server の設定](#) の説明どおりに、Oracle WebLogic Server インスタンスが設定されている。
- KIE Server が Oracle WebLogic Server インスタンスにインストールされている。
- インストールを完了するのに必要なユーザーパーミッションがある。

### 手順

1. Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから製品およびバージョンを選択します。
  - **Product:** Decision Manager
  - **バージョン:** 7.10
2. Red Hat Decision Manager 7.10.0 Add-Ons をダウンロードします。
3. ダウンロードした **rhdm-7.10.0-add-ons.zip** ファイルを一時ディレクトリーに展開します。
4. WebLogic 管理コンソールで、**Security Realms** → **Users and Groups** の順に移動します。
5. 事前に作成した **kie-server** グループに移動して、ヘッドレス Decision Manager コントローラーのユーザー (**controller** など) と、そのパスワードを作成したら、**OK** をクリックします。グループとユーザーの作成方法は「[KIE Server のグループおよびユーザーの設定](#)」を参照してください。
6. **Deployments** に移動して、既存のアプリケーションをすべて表示します。
7. **Install** をクリックします。
8. **rhdm-7.10.0-add-ons.zip** ファイルをダウンロードして展開した一時ディレクトリーに移動し、**rhdm-7.10.0-add-ons/rhdm-7.10.0-controller-ee7/controller.war** に移動します。
9. **controller.war** ファイルを選択し、**Next** をクリックして続行します。
10. ターゲットとするスタイルとして **Install this deployment as an application** を選択し、**Next** をクリックします。
11. アプリケーション名を **controller** のままにし、セキュリティーモデルを **DD Only** に設定します。残りのオプションはデフォルトのままにし、**Next** をクリックして続行します。
12. **Additional Configuration** セクションで **No, I will review the configuration later** を選択し、**Finish** をクリックします。

## 56.1. ヘッドレス DECISION MANAGER コントローラーのシステムプロパティーの設定

ヘッドレス Decision Manager コントローラーをインストールしたら、本セクションで紹介するシステムプロパティーをアプリケーションサーバーに設定して、ヘッドレス Decision Manager コントローラーとの適切な接続を有効にします。



### 注記

実稼働環境で最適な結果を得るには、KIE Server とヘッドレス Decision Manager コントローラーを別のサーバーにインストールします。開発環境の場合は、KIE Server とヘッドレス Decision Manager コントローラーを同じサーバーにインストールします。いずれの場合も、ヘッドレス Decision Manager コントローラーがインストールされているすべてのアプリケーションサーバーでこのプロパティーを変更します。

### 前提条件

- KIE Server およびヘッドレス Decision Manager コントローラーが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る。

### 手順

1. ヘッドレス Decision Manager コントローラーがインストールされているアプリケーションサーバーインスタンスに、以下の JVM プロパティー値を指定します。

表56.1 ヘッドレス Decision Manager コントローラーに必要なプロパティー

名前	要件
<b>org.kie.server.user</b>	<b>kie-server</b> ロールが割り当てられているユーザー
<b>org.kie.server.pwd</b>	<b>org.kie.server.user</b> プロパティーに指定したユーザーのパスワード

2. KIE Server がインストールされているアプリケーションサーバーインスタンスに、以下の JVM プロパティー値を指定します。

表56.2 ヘッドレス Decision Manager コントローラーがインストールされている場合に KIE Server に必要なプロパティー

名前	要件
<b>org.kie.server.controller.user</b>	<b>kie-server</b> ロールが割り当てられているユーザー
<b>org.kie.server.controller.pwd</b>	<b>org.kie.server.controller.user</b> プロパティーに指定したユーザーのパスワード
<b>org.kie.server.id</b>	KIE Server インストールの ID または名前 ( <b>rhdm700-decision-server-1</b> など)

名前	要件
<b>org.kie.server.location</b>	KIE Server の URL ( <b>http://&lt;HOST&gt;:&lt;PORT&gt;/kie-server/services/rest/server</b> )
<b>org.kie.server.controller</b>	ヘッドレス Decision Manager コントローラーの URL ( <b>http://&lt;HOST&gt;:&lt;PORT&gt;/controller/rest/controller</b> )

**<HOST>** は、KIE Server ホストの ID または名前に置き換えます (例: **localhost** または **192.7.8.9**)。

**<PORT>** は、KIE Server ホストのポート番号に置き換えます (例: **7001**)。

## 56.2. インストールの検証

ヘッドレス Decision Manager コントローラーをインストールし、必要なシステムプロパティーとロール要件をアプリケーションサーバーに定義したら、ヘッドレス Decision Manager コントローラーが正しく動作していることを確認します。

### 前提条件

- KIE Server およびヘッドレス Decision Manager コントローラーが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る。
- アプリケーションサーバーで、ヘッドレス Decision Manager コントローラーに必要なシステムプロパティーおよびロール要件をすべて設定している。

### 手順

端末で以下のコマンドを実行して、ヘッドレス Decision Manager コントローラーが動作していることを確認します。

```
curl -X GET "http://<HOST>:<PORT>/controller/rest/controller/management/servers" -H "accept: application/xml" -u '<CONTROLLER>:<CONTROLLER_PWD>'
```

**<HOST>** は、KIE Server ホストの ID または名前に置き換えます (例: **localhost** または **192.7.8.9**)。

**<PORT>** は、KIE Server ホストのポート番号に置き換えます (例: **7001**)。

**<CONTROLLER>** および **<CONTROLLER\_PWD>** は、本セクションで作成したユーザーの認証情報です。

このコマンドにより、KIE Server インスタンスに関する情報が返されます。



### 注記

別の方法として、KIE Server Java API Client を使用してヘッドレス Decision Manager コントローラーにアクセスすることもできます。

ヘッドレス Decision Manager コントローラーが実行していない場合は、アプリケーションサーバーインスタンスを停止して再起動し、ヘッドレス Decision Manager コントローラーの URL または API へ再度アクセスしてみてください。

## 第57章 ORACLE WEBLOGIC SERVER での埋め込みデシジョンエンジンの設定

デシジョンエンジンとは、デシジョンとビジネスプロセスを実行できるようにする軽量のルールエンジンです。デシジョンエンジンは、Red Hat Decision Manager アプリケーションに含めるか、OpenShift、Kubernetes、および Docker 経由でサービスとしてデプロイできます。また、API を使用して Red Hat Decision Manager アプリケーションに埋め込むか、または contexts and dependency injection (CDI) サービスセットの一部として、デシジョンエンジンを埋め込むことができます。

Red Hat Decision Manager アプリケーションで、埋め込まれたエンジンを使用する予定の場合には、Red Hat Business Automation 部品表 (BOM) ファイルをプロジェクトの **pom.xml** ファイルに追加して、プロジェクトに、Maven の依存関係を追加する必要があります。Red Hat Business Automation BOM は、Red Hat Decision Manager に適用されます。Red Hat Business Automation BOM (Bill of Materials) についての詳細情報は、[What is the mapping between Red Hat Decision Manager and the Maven library version?](#) を参照してください。

### 手順

1. Red Hat Business Automation BOM を **pom.xml** ファイルで宣言します。

```
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>com.redhat.ba</groupId>
      <artifactId>ba-platform-bom</artifactId>
      <version>7.10.0.redhat-00004</version>
      <type>pom</type>
      <scope>import</scope>
    </dependency>
  </dependencies>
</dependencyManagement>
<dependencies>
  <!-- Your dependencies -->
</dependencies>
```

2. **<dependencies>** タグでお使いのプロジェクトに必要な依存関係を宣言します。製品の BOM をプロジェクトにインポートしたら、ユーザー向け製品依存関係のバージョンが定義されるため、**<dependency>** 要素のサブ要素 **<version>** を指定する必要はありません。ただし、**<dependency>** 要素を使用して、プロジェクトで使用する依存関係を宣言する必要があります。
  - 標準的な Red Hat Decision Manager プロジェクトでは、使用する機能に応じて、以下の依存関係を宣言します。

### 埋め込みデシジョンエンジン依存関係

```
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-compiler</artifactId>
</dependency>

<!-- Dependency for persistence support. -->
<dependency>
  <groupId>org.drools</groupId>
  <artifactId>drools-persistence-jpa</artifactId>
```

```
</dependency>
```

```
<!-- Dependencies for decision tables, templates, and scorecards.
For other assets, declare org.drools:business-central-models-* dependencies. -->
```

```
<dependency>
```

```
  <groupId>org.drools</groupId>
```

```
  <artifactId>drools-decisiontables</artifactId>
```

```
</dependency>
```

```
<dependency>
```

```
  <groupId>org.drools</groupId>
```

```
  <artifactId>drools-templates</artifactId>
```

```
</dependency>
```

```
<dependency>
```

```
  <groupId>org.drools</groupId>
```

```
  <artifactId>drools-scorecards</artifactId>
```

```
</dependency>
```

```
<!-- Dependency for loading KJARs from a Maven repository using KieScanner. -->
```

```
<dependency>
```

```
  <groupId>org.kie</groupId>
```

```
  <artifactId>kie-ci</artifactId>
```

```
</dependency>
```

- KIE Server を使用するには、以下の依存関係を宣言します。

#### クライアントアプリケーション KIE Server の依存関係

```
<dependency>
```

```
  <groupId>org.kie.server</groupId>
```

```
  <artifactId>kie-server-client</artifactId>
```

```
</dependency>
```

- Red Hat Decision Manager にリモートクライアントを作成するには、以下の依存関係を宣言します。

#### クライアントの依存関係

```
<dependency>
```

```
  <groupId>org.uberfire</groupId>
```

```
  <artifactId>uberfire-rest-client</artifactId>
```

```
</dependency>
```

- ルール、プロセス定義など、アセットを含む JAR ファイルを作成する場合は、お使いの Maven プロジェクトのパッケージングの種類を **kjar** と指定し、**org.kie:kie-maven-plugin** を使用して、**<project>** 要素に置かれた **kjar** パッケージングタイプを処理します。以下の例の **\${kie.version}** は、[What is the mapping between Red Hat Decision Manager and the Maven library version?](#) に記載されている Maven ライブラリーのバージョンです。

```
<packaging>kjar</packaging>
```

```
<build>
```

```
  <plugins>
```

```
    <plugin>
```

```
      <groupId>org.kie</groupId>
```

```
      <artifactId>kie-maven-plugin</artifactId>
```

```
      <version>${kie.version}</version>
```

```

    <extensions>true</extensions>
  </plugin>
</plugins>
</build>

```

- プロジェクトで永続サポートのあるデシジョンエンジンを使用する場合は、Red Hat Business Automation BOM ファイルから **version.org.hibernate-4ee7** プロパティをコピーして、**pom.xml** ファイルの **dependencyManagement** セクションに以下のハイバーネートの依存関係を宣言する必要があります。

### 永続性のあるデシジョンエンジンにおけるハイバーネートの依存関係

```

<!-- hibernate dependencies -->
<dependencyManagement>
  <dependencies>
    <dependency>
      <groupId>org.hibernate</groupId>
      <artifactId>hibernate-entitymanager</artifactId>
      <version>${version.org.hibernate-4ee7}</version>
    </dependency>

    <dependency>
      <groupId>org.hibernate</groupId>
      <artifactId>hibernate-core</artifactId>
      <version>${version.org.hibernate-4ee7}</version>
    </dependency>
  </dependencies>
</dependencyManagement>

```

## 第58章 キーストアを使用したパスワードセキュリティの確保

キーストアを使用して、Business Central と KIE Server の間の通信に使用するパスワードを暗号化できます。コントローラーと KIE Server のパスワードを暗号化する必要があります。Business Central と KIE Server を別のアプリケーションサーバーにデプロイする場合は、いずれのアプリケーションサーバーもキーストアを使用する必要があります。

Java Cryptography Extension KeyStore (JCEKS) は、対称鍵をサポートするため、キーストアには JCEKS を使用してください。JDK インストールに含まれる KeyTool を使用して、新しい JCEKS を作成します。



### 注記

KIE Server が JCEKS で設定されていない場合、KIE Server のパスワードはシステムプロパティにプレーンテキスト形式で保存されます。

### 前提条件

- KIE Server が Oracle WebLogic Server にインストールされている。
- 「[KIE Server のグループおよびユーザーの設定](#)」の記載通りに、**kie-server** ロールを割り当てた KIE Server ユーザーが作成されている。
- Java 8 以降がインストールされている。

### 手順

1. KeyTool を使用して JCEKS を作成するには、Java 8 のホームディレクトリーで以下のコマンドを実行します。

```
$<JAVA_HOME>/bin/keytool -importpassword -keystore <KEYSTORE_PATH> -keypass
<ALIAS_KEY_PASSWORD> -alias <PASSWORD_ALIAS> -storepass
<KEYSTORE_PASSWORD> -storetype JCEKS
```

上記の例では、以下の変数を置き換えてください。

- **<KEYSTORE\_PATH>**: キーストアの保存先のパス
  - **<KEYSTORE\_PASSWORD>**: キーストアのパスワード
  - **<ALIAS\_KEY\_PASSWORD>**: エイリアスで保存した値にアクセスする時に使用するパスワード
  - **<PASSWORD\_ALIAS>**: プロセスへのエントリーに使用するエイリアス
2. プロンプトが表示されたら、作成した KIE Server ユーザーのパスワードを入力します。
  3. 以下の表に記載のシステムプロパティを設定します。

表58.1 KIE Server JCEKS を読み込む時に使用するシステムプロパティ

システムプロパティ	プレースホルダー	説明
-----------	----------	----



システムプロパティ	プレースホルダー	説明
<b>kie.keystore.keyStoreURL</b>	<b>&lt;KEYSTORE_URL&gt;</b>	使用する JCEKS の URL (例: <b>file:///home/kie/keystores/key store.jceks</b> )
<b>kie.keystore.keyStorePwd</b>	<b>&lt;KEYSTORE_PWD&gt;</b>	JCEKS のパスワード
<b>kie.keystore.key.server.alias</b>	<b>&lt;KEY_SERVER_ALIA S&gt;</b>	パスワードの保存先となる REST サービスのキーのエイリアス
<b>kie.keystore.key.server.pwd</b>	<b>&lt;KEY_SERVER_PWD &gt;</b>	保存したパスワードを使用する REST サービスのエイリアスのパ スワード
<b>kie.keystore.key.ctrl.alias</b>	<b>&lt;KEY_CONTROL_AL IAS&gt;</b>	パスワードの保存先のデフォルト の REST Process Automation Controller のキーのエイリアス
<b>kie.keystore.key.ctrl.pwd</b>	<b>&lt;KEY_CONTROL_P WD&gt;</b>	保存したパスワードを使用する、 デフォルトの REST Process Automation Controller のエイリア スのパスワード

4. KIE Server を起動して、設定を検証します。

## 第59章 次のステップ

- [デシジョンサービスのスタートガイド](#)
- [ガイド付きデシジョンテーブルを使用したデシジョンサービスの作成](#)

## パート VII. RED HAT DECISION MANAGER 7.10 の修正およびアップグレード

Red Hat Decision Manager をお使いの場合は、Red Hat カスタマーポータルで公開されている Red Hat Decision Manager への更新を、お使いの Red Hat Decision Manager ディストリビューションに適用して、機能強化および修正を最新に保つことができます。お客様がより簡単に Red Hat Decision Manager および KIE Server に更新を適用できるように、Red Hat では、新しい製品リリース向けの更新ツールおよび製品通知をご用意しています。

### 前提条件

- Red Hat カスタマーポータルアカウント
- Red Hat Decision Manager 7.10 をインストールしている。インストールオプションは、[Red Hat Decision Manager インストールの計画](#) を参照してください。



### 注記

Red Hat Decision Manager 7.9 を使用しており、Red Hat Decision Manager 7.10 にアップグレードする場合は、[Red Hat Decision Manager 7.9 の修正およびアップグレード](#) を参照してください。

## 第60章 RED HAT DECISION MANAGER のバージョン管理

Red Hat Decision Manager のバージョンには、メジャーリリース、マイナーリリース、およびパッチリリースのそれぞれの数字を組み合わせた **メジャーリリース.マイナーリリース.パッチリリース** という書式が使用されています (例: **7.10.1**)。メジャーリリースが **7.x.x**、マイナーリリースが **7.10.x**、パッチリリースが **7.10.1** となります。メジャーリリースでは、大抵の場合はデータ移行が求められますが、マイナーリリースの更新やパッチの更新は、Red Hat Decision Manager リリースアーティファクトで提供されるアップグレードおよび更新ツールで管理されます。

以下は、Red Hat Decision Manager の一般的なリリースの種類になります。

### メジャーリリースの移行

Red Hat Decision Manager のメジャーリリースには、かなりの数の機能強化、セキュリティ更新、バグ修正が含まれており、再設計された機能も含まれている可能性があります。Red Hat JBoss BRMS 6.4.x から Red Hat Decision Manager 7.0 など、メジャーリリースから別のメジャーリリースにアプリケーションを移行すると、データ移行が必要になる場合があります。移行を容易にするために、Red Hat Decision Manager の新しいメジャーバージョンで、自動化した移行ツールがしばしば提供されますが、場合によっては特定のデータおよび設定を手動で処理する必要があります。サポートされる移行パスは、製品のお知らせおよびドキュメントに指定されています。

たとえば、移行の方法については、[Red Hat JBoss BRMS 6.4 から Red Hat Decision Manager 7.0 への移行](#) を参照してください。

### マイナーリリースのアップグレード

Red Hat Decision Manager のマイナーリリースには、機能拡張、セキュリティ更新、バグ修正が含まれます。Red Hat Decision Manager 7.9.x から 7.10 など、アプリケーションを別のマイナーリリースに移行する場合には、データの移行が必要な場合があります。自動更新ツールは、多くの場合、パッチ更新と Red Hat Decision Manager の新しいマイナーバージョンの両方で提供され、Business Central、KIE Server、ヘッドレス Decision Manager コントローラーなど、Red Hat Decision Manager の特定コンポーネントの更新を容易にします。デシジョンエンジンやスタンドアロンの Business Central など、その他の Red Hat Decision Manager アーティファクトは、各マイナーリリースが含まれる新しいアーティファクトとしてリリースされるため、再インストールして更新を適用する必要があります。

新たなマイナーリリースにアップグレードする前に、お使いの Red Hat Decision Manager に最新のパッチ更新を適用してください。

### パッチの更新

Red Hat Decision Manager のパッチの更新には、最新のセキュリティ更新、およびバグ修正が含まれます。予定されているパッチ更新には、製品のマイナーバージョンに対してすでにリリースされているパッチ更新がすべて含まれるため、更新を最新の状態にするために、パッチの更新を1つ1つ適用していく必要はありません。たとえば、Red Hat Decision Manager 7.9.0 または 7.9.1 は Red Hat Decision Manager 7.9.2 に更新できます。ただし、Red Hat Decision Manager で最適なパフォーマンスを得るには、製品の更新が利用可能になったらすぐに適用する必要があります。また、既存製品の通常の更新サイクルとは別に、予定されていないパッチ更新がリリースされる場合があります。これには、特定の問題を修正するために Red Hat グローバルサポートサービス (GSS) によって提供されるセキュリティまたはその他の更新が含まれる場合があります、累積的な更新ではない場合があります。

自動更新ツールは、多くの場合、パッチ更新と Red Hat Decision Manager の新しいマイナーバージョンの両方で提供され、Business Central、KIE Server、ヘッドレス Decision Manager コントローラーなど、Red Hat Decision Manager の特定コンポーネントの更新を容易にします。デシジョンエンジンやスタンドアロンの Business Central など、その他の Red Hat Decision Manager アーティファクトは、各マイナーリリースが含まれる新しいアーティファクトとしてリリースされるため、再インストールして更新を適用する必要があります。

リリース間での移行を最適に保ち、Red Hat Decision Manager ディストリビューションに最新の機能強化および修正が含まれるようにするには、製品リリースや更新が Red Hat カスタマーポータルで利用可能になった時点で、Red Hat Decision Manager に適用します。Red Hat カスタマーポータルで製品の通知を有効にすることも検討してください。

製品通知を有効にする方法は、[61章Red Hat カスタマーポータルでRed Hat Decision Manager の通知の有効化](#)を参照してください。

## 第61章 RED HAT カスタマーポータルで RED HAT DECISION MANAGER の通知の有効化

Red Hat カスタマーポータルで通知を有効にして、Red Hat Decision Manager の更新およびお知らせを受け取ることができます。Red Hat Decision Manager に関して通知されるのは、ドキュメントの更新または追加、製品リリース、およびパッチ更新となります。通知を有効にすれば、Red Hat カスタマーポータルで利用可能になった場合に、Red Hat Decision Manager に製品の更新を適用し、Red Hat Decision Manager ディストリビューションで最新の機能強化と修正を維持するのが簡単になります。

### 前提条件

- Red Hat カスタマーポータルのアカウントがあり、ログインしている。

### 手順

1. Red Hat カスタマーポータルウィンドウの右上隅で、プロフィール名をクリックして **通知** をクリックします。
2. **通知** タブを選択し、**通知の管理** をクリックします。
3. **フォロー** の横にあるドロップダウンメニューから **製品** を選択し、表示されたドロップダウンメニューから **Red Hat Decision Manager** を選択します。
4. **通知設定を保存** をクリックします。

必要に応じて、その他の製品の通知も同じように追加できます。

## 第62章 RED HAT DECISION MANAGER へのパッチ更新およびマイナーリリースアップグレードの適用

自動更新ツールは、多くの場合、パッチ更新と Red Hat Decision Manager の新しいマイナーバージョンの両方で提供され、Business Central、KIE Server、ヘッドレス Decision Manager コントローラーなど、Red Hat Decision Manager の特定コンポーネントの更新を容易にします。デシジョンエンジンやスタンドアロンの Business Central など、その他の Red Hat Decision Manager アーティファクトは、各マイナーリリースが含まれる新しいアーティファクトとしてリリースされるため、再インストールして更新を適用する必要があります。

この自動更新ツールを使用してパッチ更新とマイナーリリースアップグレードの両方を Red Hat Decision Manager 7.10 に適用することができます。バージョン 7.10 から 7.10.1 への更新といった Red Hat Decision Manager のパッチ更新には、最新のセキュリティ更新とバグ修正が含まれます。バージョン 7.9.x から 7.10 へのアップグレードといった Red Hat Decision Manager のマイナーリリースアップグレードには、機能強化、セキュリティ更新、バグ修正が含まれます。



### 注記

Red Hat Decision Manager 更新ツールに含まれるのは、Red Hat Decision Manager の更新のみです。Red Hat JBoss EAP への更新は、Red Hat JBoss EAP パッチ配信を使用して適用する必要があります。詳細は [Red Hat JBoss EAP パッチおよびアップグレードガイド](#) を参照してください。

### 前提条件

- Red Hat Decision Manager インスタンスおよび KIE Server インスタンスが稼働していない。Red Hat Decision Manager または KIE Server のインスタンスを実行している間は更新を適用しないでください。

### 手順

- Red Hat カスタマーポータルの [Software Downloads](#) ページに移動し (ログインが必要)、ドロップダウンオプションから選択およびバージョンを選択します。  
バージョン 7.9.x から 7.10 などのように Red Hat Decision Manager のマイナーリリースにアップグレードする場合は、お使いの Red Hat Decision Manager に最新のパッチ更新を適用してから、以下の手順に従って、新たなマイナーリリースにアップグレードしてください。
- Patches** をクリックし、**Red Hat Decision Manager [VERSION] Update Tool** をダウンロードし、ダウンロードした **rhdm-\$VERSION-update.zip** ファイルを一時ディレクトリーに展開します。  
この更新ツールは、Business Central、KIE Server、ヘッドレス Decision Manager コントローラーなど、Red Hat Decision Manager の一定のコンポーネントの更新を自動化します。このツールを使用して最初に更新を適用し、Red Hat Decision Manager ディストリビューションに関連するその他の更新、または新しいリリースアーティファクトをインストールします。
- 更新ツールがファイルを更新しないようにするには、展開した **rhdm-\$VERSION-update** ディレクトリーに移動し、**blacklist.txt** ファイルを開き、更新しないファイルの相対パスを追加します。  
ファイルが **blacklist.txt** ファイルの一覧に追加されていると、更新スクリプトは、そのファイルを新しいバージョンに置き換えずにそのまま残し、新しいバージョンのファイルに **.new** 接尾辞を付けて追加します。ブロックファイルが配布されなくなると、更新ツールは、**.removed** 接尾辞の付いた、空のマーカーファイルを作成します。次に、これらの新しいファイルを手動で保持、マージ、または削除することを選択できます。

**blacklist.txt** ファイルで除外されるファイルの例:

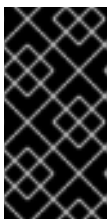
```
WEB-INF/web.xml // Custom file
styles/base.css // Obsolete custom file kept for record
```

更新後の、ブロックされたファイルディレクトリー内のコンテンツ:

```
$ ls WEB-INF
web.xml web.xml.new
```

```
$ ls styles
base.css base.css.removed
```

4. コマンドの端末で、**rhdm-\$VERSION-update.zip** ファイルから展開した一時ファイルに移動し、以下の形式で **apply-updates** スクリプトを実行します。



### 重要

更新を適用する前に、Red Hat Decision Manager インスタンスおよび KIE Server インスタンスが実行していないことを確認します。Red Hat Decision Manager または KIE Server のインスタンスを実行している間は更新を適用しないでください。

Linux システムまたは Unix ベースのシステムの場合:

```
$ ./apply-updates.sh $DISTRO_PATH $DISTRO_TYPE
```

Windows の場合:

```
$ .\apply-updates.bat $DISTRO_PATH $DISTRO_TYPE
```

**\$DISTRO\_PATH** の部分は、関連するディストリビューションディレクトリーへのパスで、**\$DISTRO\_TYPE** の部分は、更新しているディストリビューションの種類となります。

Red Hat Decision Manager 更新ツールでは、以下のディストリビューションの種類がサポートされます。

- **rhdm-decision-central-eap7-deployable**: Business Central を更新します (**decision-central.war**)。
- **rhdm-kie-server-ee8**: KIE Server を更新します (**kie-server.war**)。
- **rhdm-kie-server-jws**: Red Hat JBoss Web Server で KIE Server を更新します (**kie-server.war**)。
- **rhdm-controller-ee7**: ヘッドレス Decision Manager コントローラー (**controller.war**) を更新します。
- **rhdm-controller-jws**: Red Hat JBoss Web Server でヘッドレスの Decision Manager コントローラーを更新します (**controller.war**)。  
Red Hat JBoss EAP で、Red Hat Decision Manager の完全ディストリビューションに対する Business Central および KIE Server への更新の例:



```
./apply-updates.sh ~EAP_HOME/standalone/deployments/decision-central.war rhdm-  
decision-central-eap7-deployable
```

```
./apply-updates.sh ~EAP_HOME/standalone/deployments/kie-server.war rhdm-kie-  
server-ee8
```

ヘッドレス Decision Manager コントローラーへの更新例 (使用している場合):

```
./apply-updates.sh ~EAP_HOME/standalone/deployments/controller.war rhdm-controller-  
ee7
```

更新スクリプトは、展開した **rhdm-\$VERSION-update** ディレクトリーに、指定したディストリビューションのコピーを含む **backup** ディレクトリーを作成してから、更新を行います。

- 更新ツールが完了したら、Red Hat カスタマーポータルで、更新ツールをダウンロードした **Software Downloads** ページに戻り、Red Hat Decision Manager ディストリビューションに関するその他の更新または新しいリリースアーティファクトをインストールします。  
デシジョンエンジンまたはその他のアドオンに関する **.jar** など、Red Hat Decision Manager ディストリビューションにすでに存在しているファイルについては、ファイルの既存のバージョンを、Red Hat カスタマーポータルから取得した新しいバージョンと取り替えます。
- エアギャップ環境など、スタンドアロンの **Red Hat Decision Manager 7.10.0 Maven Repository** アーティファクト (**rhdm-7.10.0-maven-repository.zip**) を使用している場合は、**Red Hat Decision Manager 7.10.x Maven Repository** をダウンロードし、ダウンロードした **rhdm-7.10.x-maven-repository.zip** ファイルを既存の **~/maven-repository** ディレクトリーに展開して、関連するコンテンツを更新します。

Maven リポジトリの更新例:

```
$ unzip -o rhdm-7.10.x-maven-repository.zip 'rhba-7.10.1.GA-maven-repository/maven-  
repository/*' -d /tmp/rhbaMavenRepoUpdate
```

```
$ mv /tmp/rhbaMavenRepoUpdate/rhba-7.10.0.GA-maven-repository/maven-repository/  
$REPO_PATH/
```



## 注記

更新が完了したら **/tmp/rhbaMavenRepoUpdate** ディレクトリーを削除してください。

- 関連する更新をすべて適用したら、Red Hat Decision Manager および KIE Server を起動して、Business Central にログインします。
- Business Central 内のすべてのプロジェクトデータが存在して正確であることを確認し、Business Central ウィンドウの右上隅でプロファイル名をクリックし、**About** をクリックして、更新した製品バージョン番号を確認します。  
Business Central でエラーが発生したり、データが不足していることが通知されたら、**rhdm-\$VERSION-update** ディレクトリーの **backup** ディレクトリーにコンテンツを復元し、更新ツールへの変更を戻します。Red Hat カスタマーポータルで Red Hat Decision Manager の以前のバージョンから、関連するリリースアーティファクトを再インストールできます。以前のディストリビューションを復元したら、更新を再実行してください。

## 第63章 関連情報

- [Red Hat Decision Manager インストールの計画](#)
- [Red Hat Decision Manager 7.9 の修正およびアップグレード](#)
- [Red Hat Decision Manager 7.10 のリリースノート](#)

## 付録C バージョン情報

本書の最終更新日：2022年3月8日（火）

## 付録D お問い合わせ先

Red Hat Decision Manager ドキュメントチーム: [brms-docs@redhat.com](mailto:brms-docs@redhat.com)