



## Red Hat Decision Manager 7.0

デシジョンサービスのパッケージングおよびデプロイメント



# Red Hat Decision Manager 7.0 デシジョンサービスのパッケージングおよびデプロイメント

---

Red Hat Customer Content Services  
brms-docs@redhat.com

## 法律上の通知

Copyright © 2019 Red Hat, Inc.

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

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

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

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

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

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

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

All other trademarks are the property of their respective owners.

## 概要

本書は、Red Hat Decision Manager 7.0 におけるデシジョンサービスのパッケージングおよびデプロイメントについて説明します。

## 目次

前書き .....	3
第1章 概要 .....	4
1.1. デシジョンサービスのパッケージング	4
1.2. デシジョンサービスのデプロイメント	4
第2章 デシジョンサービスへのグループ ID、アーティファクト ID、およびバージョンの値 (GAV) の設定 ....	6
第3章 DECISION CENTRAL を使用したデシジョンサービスのデプロイメントおよび管理 .....	7
3.1. DECISION CENTRAL へのデシジョンサービスのビルドおよびデプロイメント	7
3.2. DECISION CENTRAL でのデシジョンサービスのデプロイメントの管理	7
3.2.1. コンテナの作成	8
3.2.2. コンテナの管理	8
3.3. DECISION CENTRAL に接続するように DECISION SERVER の設定	8
第4章 外部の MAVEN リポジトリを使用したデシジョンサービスのデプロイメント .....	11
4.1. 外部 MAVEN リポジトリを使用するように DECISION CENTRAL の設定	11
4.2. デシジョンサービスを実行するように DECISION SERVER の設定	11
4.2.1. Maven リポジトリを使用するように Decision Server の設定	12
4.2.2. Decision Server へのサービスのロードおよび起動	12
4.2.3. サービスの停止および Decision Server から削除	13
付録A バージョン情報 .....	14



---

## 前書き

Decision Central を使用してデシジョンサービスを開発したら、使用するためにデプロイする必要があります。デシジョンサービスをパッケージ化し、Decision Server にデプロイする方法が複数あります。

## 第1章 概要

Decision Central を使用して Red Hat Decision Manager でデシジョンサービスを開発できます。デシジョンサービスを実行するには、Decision Server にデプロイする必要があります。

### 1.1. デシジョンサービスのパッケージング

デシジョンサービスは常に **KJAR ファイル** にパッケージ化されます。このファイルは Java JAR ファイルのバージョンです。Decision Server は、その KJAR ファイルからサービスを実行しますが、KJAR ファイルを直接設定する必要はありません。

代わりに、Red Hat Decision Manager は、**Maven リポジトリ** に KJAR ファイルを保存します。Maven は、Java パッケージのライフサイクルおよび依存関係管理を自動化する総合システムです。Maven は、その他のシステムとともに、(KJAR ファイルを含む) JAR ファイルのアップロードやダウンロードを行うリポジトリシステムをサポートします。

Maven では、KJAR ファイルの識別に、名前ではなく、**グループ ID**、**アーティファクト ID**、**バージョン** の 3 つの値を使用します。通常、グループ ID はプロジェクトの識別子で、プロジェクトにおける特定のアーティファクト (デシジョンサービスなど) を識別します。バージョンの値は、デプロイするすべてのバージョンで一貫にする必要があります。

Maven で (KJAR ファイルを含む) アーティファクトを識別するには、この 3 つの値がすべて必要になります。この情報セットはしばしば GAV (Group, Artifact, Version) と呼ばれます。

Decision Central にデシジョンサービスをデプロイする場合は、プロジェクト設定画面に GAV を設定できます。

### 1.2. デシジョンサービスのデプロイメント

デプロイメントプロセスは、インフラストラクチャーのニーズに基づいて変わる可能性があります。

Red Hat Decision Manager の簡単なデプロイメントでは、Decision Central および Decision Server が 1 つずつあります。Decision Central を使用してデシジョンサービスを開発し、Decision Server を管理します。Decision Central にプロジェクトをビルドして、Decision Server に自動的にデプロイします。

自動デプロイメントを有効にするために、Decision Central に、ビルドインの Maven リポジトリが含まれています。Decision Central を使用して、Decision Server を管理し、デシジョンサービスと、ビルドしたバージョンのデプロイ、削除、起動、停止を行うことができます。

複数の Decision Server を同じ Decision Central に接続することもできます。異なる **サーバーテンプレート** に分けることもできます。同じサーバーテンプレートに属するサーバーは、同じデシジョンサービスを実行しますが、別のテンプレートに、異なるデシジョンサービス、または異なるバージョンのデシジョンサービスをデプロイできます。

たとえば、**Test** テンプレートにテストサーバーがあり、**Production** テンプレートにプロダクションサーバーがあるとします。デシジョンサービスを開発する場合は、**Test** テンプレートにデプロイします。そのバージョンのサービスを十分にテストしたら、そのサービスを **Production** テンプレートにデプロイできます。

このとき、サービスの開発を継続するには、プロジェクト設定でバージョンを変更します。これにより、ビルドイン Maven リポジトリで、新しいバージョンと古いバージョンが別のアーティファクトと見なされます。新しいバージョンを **Test** テンプレートにデプロイし、古いバージョンを **Production** テンプレートで実行し続けることができます。



このデプロイメントプロセスは単純ですが、重要な制限があります。中でも、アクセス制限は十分ではなく、開発者がプロジェクトをプロダクション環境に直接デプロイできます。

適切な統合プロセスが必要な場合は、外部の Maven リポジトリ (Nexus など) を使用できます。ビルドインリポジトリではなく、外部リポジトリにデシジョンサービスファイルをプッシュするように、Decision Central を設定することもできます。

Decision Central を使用して、テスト環境の Decision Server にデシジョンサービスをデプロイできますが、このプロセスでは、(ステージングやプロダクションなどの) その他の Decision Server が Decision Central に接続されません。代わりに、デシジョンサービスの KJAR ファイルと、必要な依存関係を Maven リポジトリから取得します。必要に応じて、実装する統合プロセスと一致するリポジトリから KJAR バージョンを処理できます。

OpenShift に Decision Server をデプロイする場合は、Maven リポジトリからデシジョンサービスを自動的にロードして起動するように設定できます。オンプレミスで Decision Server をデプロイする場合は、サーバーを設定する際に Maven リポジトリへのアクセスを設定できます。これにより、サーバーの REST API を使用して、リポジトリからデシジョンサービスをロードし、起動できます。API コールの自動化には、Maven プロジェクトおよび Java クライアントライブラリーが利用できます。

## 第2章 デシジョンサービスへのグループ ID、アーティファクト ID、およびバージョンの値 (GAV) の設定

Decision Central にデシジョンサービスをデプロイする場合は、プロジェクト設定画面に GAV を設定できます。

この値は、Maven リポジトリで、デシジョンサービスの識別子となります。どのデプロイメント方法を選択しても、デシジョンサービスは GAV によって識別されます。これは、GAV を変更しないで新しいバージョンをビルドすると、Maven リポジトリおよび Decision Server にある古いバージョンに置き換えられることを示しています。GAV の値 (通常はバージョン) を変更すると、別のアーティファクトとして見なされ、古いバージョンと共存させることができます。

### 手順

1. メインメニューで **Projects** をクリックします。
2. プロジェクト名をクリックします。  
もしくは、**Add Project** ボタンをクリックして、新しいプロジェクトを作成します。
3. **Settings** ボタンをクリックします。
4. 必要に応じて **Group ID** フィールド、**Artifact ID** フィールド、**Version** フィールドを修正します。デシジョンサービスをすでに開発しており、新しいバージョンを開発中の場合、通常はバージョン番号を変更する必要があります。
5. **Save** ボタンをクリックします。

## 第3章 DECISION CENTRAL を使用したデシジョンサービスのデプロイメントおよび管理

Decision Central を使用して、デシジョンサービスをビルドしてデプロイできます。複数の Decision Server を Decision Central に接続した場合は、web UI を使用してデシジョンサービスを全サーバーにデプロイして管理できます。

Decision Server を異なる **サーバーテンプレート** に分けることもできます。デシジョンサービスが同じ場合は、同じサーバーテンプレートに属するサーバーを自動的に実行しますが、別のテンプレートに、異なるデシジョンサービス、または異なるバージョンのデシジョンサービスをデプロイすることもできます。

そのサーバーの設定ファイルに、各 Decision Server のテンプレート名を設定できます。



### 重要

Decision Central を使用して、Decision Server を別のテンプレートに移動することはできません。サーバーテンプレート名を変更するには、サーバーの設定ファイルを変更する必要があります。

### 3.1. DECISION CENTRAL へのデシジョンサービスのビルドおよびデプロイメント

Decision Central にデシジョンサービスをビルドし、自動的に Decision Server にデプロイできます。

#### 手順

1. メインメニューで **Projects** をクリックします。
2. プロジェクト名をクリックします。
3. **Build & Deploy** ボタンをクリックします。
4. Decision Central に Decision Server を 1 つだけ接続する場合、または接続したすべての Decision Server が同じサーバーテンプレートにある場合は、プロジェクトが自動的にビルドされ、デプロイされます。それ以外の場合は、Decision Central がダイアログを表示します。このダイアログでは、次の値を設定します。
  - **Container ID** および **Container Alias** は、Decision Server でサービスを実行しているコンテナを識別します。通常は、この設定を変更する必要はありません。
  - **Server template** : このプロジェクトをデプロイするサーバーテンプレートを選択します。あとで、再ビルドしなくてもその他のテンプレートにデプロイできます。
  - **Start container?** : このボックスの選択を解除すると、このサービスはサーバーにデプロイされますが起動しません。通常は、このボックスを選択したままにします。

### 3.2. DECISION CENTRAL でのデシジョンサービスのデプロイメントの管理

Decision Central を使用して、1 つ以上の Decision Server でデシジョンサービスのデプロイを管理できます。各 Decision Server に、Decision Central への接続を設定する必要があります。

デシジョンサービスは、常に Decision Server の **コンテナ** で実行します。Decision Server にコンテナを作成して起動できます。プロジェクトをビルドおよびデプロイすると、コンテナが自動的に作成されます。コンテナは、常に指定したサーバーテンプレートに作成されます。

コンテナの起動、停止、および削除を行うこともできます。

### 3.2.1. コンテナの作成

Decision Central でビルドしたデシジョンサービスからコンテナを作成できます。

#### 手順

1. メインメニューで **Execution servers** をクリックします。
2. **Server templates** の下で、テンプレートを選択します。
3. **KIE Containers** の下で、**Add Container** をクリックします。
4. Decision Central にダイアログボックスが表示されます。ダイアログのテーブルで **GAV** を選択し、**GAV** の隣にある **Select** ボタンをクリックします。
5. サービスをすぐに起動する場合は、**Start container?** ボックスを選択します。
6. **Finish** ボタンをクリックします。

#### 結果

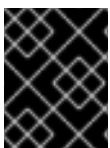
サービスに対してコンテナが作成されました。このコンテナは、このサーバーテンプレートに設定した Decision Server に置かれます。**Start container?** を選択すると、サービスが起動します。

### 3.2.2. コンテナの管理

コンテナを起動すると、そのコンテナのサービスが使用できるようになります。必要に応じてデシジョンサービスでコンテナの起動、停止、または削除を行います。

#### 手順

1. メインメニューで **Execution servers** をクリックします。
2. **Server templates** の下で、テンプレートを選択します。
3. **KIE Containers** の下で、コンテナを選択します。
4. **Start** ボタン、**Stop** ボタン、または **Remove** ボタンをクリックします。



#### 重要

コンテナが実行している場合は削除できません。実行中のコンテナを削除するには、先に停止する必要があります。

## 3.3. DECISION CENTRAL に接続するように DECISION SERVER の設定

既存の Decision Central に接続するように Decision Server を設定できます。サーバーテンプレートの名前も設定できます。このサーバーはサーバーテンプレートの一部となります。テンプレートにデプロイされたデシジョンサービスは、このサーバーにデプロイされます。



## 重要

OpenShift に Decision Server をデプロイする場合は、Decision Central に接続する設定方法について『[Red Hat OpenShift Container Platform への Red Hat Decision Manager のデプロイメント](#)』を参照してください。

## 手順

1. Decision Server をインストールします。インストール方法は『[Red Hat Decision Manager のオンプレミスインストール](#)』を参照してください。
2. Decision Central が使用する Maven リポジトリに接続するための `settings.xml` ファイルを作成します。Decision Central の設定を変更していない場合は、所有しているビルドイン Maven リポジトリを使用します。以下の `settings.xml` の設定を使用します。

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

  <servers>
    <server>
      <id>remote-repo</id>
      <username>user</username>
      <password>pwd</password>
    </server>
  </servers>
  <profiles>
    <profile>
      <id>additional-maven-repos</id>
      <repositories>
        <repository>
          <id>remote-repo</id>
          <url>http://centralhost:centralport/decision-
central/maven2/</url>
        </repository>
      </repositories>
    </profile>
  </profiles>
  <activeProfiles>
    <activeProfile>additional-maven-repos</activeProfile>
  </activeProfiles>
</settings>
```

以下の値を置き換えます。

- **user**: Decision Central にログインするユーザーのユーザー名
- **pwd**: Decision Central にログインするユーザーのパスワード
- **centralhost**: Decision Central のホスト名
- **centralport**: Decision Central のポート  
Decision Central を OpenShift にデプロイした場合は、URL から `decision-central/` を削除します。

既知の場所 (`/opt/maven/settings.xml` など) にファイルを保存します。

3. `<eap_home>/standalone/configuration/standalone.xml` ファイルの `<system-properties>` タグの下に、次のプロパティを設定します。

- **org.kie.server.controller.user**: Decision Central にログオンするアカウントのユーザー名。
- **org.kie.server.controller.pwd**: アカウントのパスワード。
- **org.kie.server.controller**: Decision Central の API に接続する URL。通常は `http://<centralhost>:<centralport>/decision-central/rest/controller` (`<centralhost>` および `<centralport>` はそれぞれ Decision Central のホスト名およびポート) になります。Decision Central を OpenShift にデプロイしている場合は、URL から `decision-central/` を削除します。
- **org.kie.server.location**: Decision Server の API に接続する URL。通常は `http://<serverhost>:<serverport>/kie-server/services/rest/server` (`<serverhost>` および `<serverport>` はそれぞれ Decision Server のホスト名およびポート) になります。
- **org.kie.server.id**: サーバーテンプレートの名前。このサーバーテンプレートが Decision Central に存在しない場合は、Decision Server が Decision Central に接続する際に自動的に作成されます。
- **kie.maven.settings.custom**: Maven リポジトリに接続するための `settings.xml` ファイルへのフルパス。

例:

```
<property name="org.kie.server.controller.user"
value="central_user"/>
<property name="org.kie.server.controller.password"
value="central_password"/>
<property name="org.kie.server.controller"
value="http://central.example.com:8080/decision-
central/rest/controller"/>
<property name="org.kie.server.location"
value="http://kieserver.example.com:8080/kie-
server/services/rest/server"/>
<property name="org.kie.server.id" value="production-servers"/>
<property name="kie.maven.settings.custom"
value="/opt/maven/settings.xml"/>
```

4. Decision Server を起動または再起動します。

## 第4章 外部の MAVEN リポジトリを使用したデシジョンサービスのデプロイメント

外部の Maven リポジトリ (Nexus など) を使用して、統合プロセスを設定できます。ビルドインリポジトリではなく、外部リポジトリにデシジョンサービスをプッシュするように、Decision Central を設定できます。

Decision Central を使用して、テスト環境の Decision Server にデシジョンサービスをデプロイできますが、このプロセスでは、(ステージングやプロダクションなどの) その他の Decision Server が Decision Central に接続されません。代わりに、デシジョンサービスの KJAR ファイルと、必要な依存関係を Maven リポジトリから取得します。必要に応じて、実装する統合プロセスと一致するリポジトリから KJAR バージョンを処理できます。

Decision Server を OpenShift にデプロイする場合は、デシジョンサービスを Maven リポジトリから自動的にロードして起動するように設定できます。

オンプレミスに Decision Server をデプロイする場合は、サーバーに設定するときに Maven リポジトリへのアクセスを設定できます。次に、このサーバーの REST API を使用して、リポジトリからデシジョンサービスをロードして起動できます。API コールの自動化には、Maven プロジェクトおよび Java クライアントライブラリーが利用できます。

### 4.1. 外部 MAVEN リポジトリを使用するように DECISION CENTRAL の設定

ビルドインリポジトリの代わりに外部の Maven リポジトリを使用するように Decision Central を設定できます。この場合、Decision Central は、このリポジトリにビルドの KJAR ファイルをすべてプッシュします。このファイルは、統合プロセスを実装する場合に必要なに応じてリポジトリから処理できます。



#### 重要

OpenShift にデプロイした Decision Central で、外部 Maven リポジトリを使用することはサポートされていません。

#### 手順

1. Decision Central をインストールします。インストール方法は『[Red Hat Decision Manager のオンプレミスインストール](#)』を参照してください。
2. 外部リポジトリへの接続およびアクセスの詳細を含む Maven `settings.xml` ファイルを作成します。`settings.xml` ファイルの詳細は、Maven ドキュメンテーション <https://maven.apache.org/settings.html> を参照してください。
3. `<eap_home>/standalone/configuration/standalone.xml` ファイルの `<system-properties>` タグで、以下のように `kie.maven.settings.custom` プロパティに `settings.xml` ファイルのフルパス名を設定します。

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

4. Decision Central を起動または再起動します。

### 4.2. デシジョンサービスを実行するように DECISION SERVER の設定

Decision Central を使用せずに、デシジョンサービスを自動的にロードして実行するように Decision Server を設定できます。このサービスは Maven リポジトリに存在する必要があります。リポジトリへのアクセス情報と、サービスの GAV 値が必要です。また、後でこのサーバーからサービスを停止して削除することもできます。Decision Server に、同時に複数のサービスを起動することもできます。



### 重要

この手順は、OpenShift にデプロイした Decision Server では行わないでください。Maven リポジトリからデシジョンサービスをロードして実行するように Decision Server を設定する方法は『[Red Hat OpenShift Container Platform への Red Hat Decision Manager のデプロイメント](#)』を参照してください。

## 4.2.1. Maven リポジトリを使用するように Decision Server の設定

Decision Server のデシジョンサービスをロードして起動する前に、Maven リポジトリを使用するようにサーバーを設定する必要があります。

### 手順

1. Decision Server をインストールします。インストール方法は『[Red Hat Decision Manager のオンプレミスインストール](#)』を参照してください。
2. 外部リポジトリへの接続およびアクセスの詳細を含む Maven `settings.xml` ファイルを作成します。`settings.xml` ファイルの詳細は、Maven ドキュメンテーション <https://maven.apache.org/settings.html> を参照してください。
3. `<eap_home>/standalone/configuration/standalone.xml` ファイルの `<system-properties>` タグで、以下のように `kie.maven.settings.custom` プロパティに `settings.xml` ファイルのフルパス名を設定します。

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

4. Decision Server を起動または再起動します。

## 4.2.2. Decision Server へのサービスのロードおよび起動

Maven リポジトリを使用するように Decision Server を設定すると、API コールを使用してサービスをロードできます。このサービスは自動的に起動します。

### 手順

1. Decision Server のコンテナにサービスをロードして起動するには、以下のコマンドを実行して、API リクエストを送信します。

```
$ curl --user "<username>:<password>" -H "Content-Type: application/json" -X PUT -d '{"container-id" : "<containerID>","release-id" : {"group-id" : "<groupID>","artifact-id" : "<artifactID>","version" : "<version>"}}' http://<serverhost>:<serverport>/kie-server/services/rest/server/containers/<containerID>
```

ここでは、



- **<username>** は、Decision Server にログオンして管理を行うアカウントのユーザー名です。
  - **<password>** は、アカウントのパスワードです。
  - **<containerID>** は、コンテナの識別子です。ランダムな識別子を使用することもできますが、コマンドの URL およびデータの両方で同じものを使用する必要があります。
  - **<groupId>**、**<artifactID>**、**version** は GAV 値です。
  - **<serverhost>** は、KIE サーバーのホスト名です。KIE サーバーと同じホストでコマンドを実行する場合は **localhost** になります。
  - **<serverport>** は、KIE サーバーのポート番号です。
- 例:

```
curl --user "rhdmAdmin:password@1" -H "Content-Type: application/json" -X PUT -d '{"container-id" : "kie1", "release-id" : {"group-id" : "org.kie.server.testing", "artifact-id" : "container-crud-tests1", "version" : "2.1.0.GA"}}' http://localhost:39043/kie-server/services/rest/server/containers/kie1
```

### 4.2.3. サービスの停止および Decision Server から削除

API コールを使用して、Decision Server サービスを停止および削除できます。サービスをロードする際に設定したコンテナ識別子が必要です。

#### 手順

1. Decision Server のサービスを使用してコンテナを停止および削除するには、以下のコマンドを実行して API リクエストを送信します。

```
$ curl --user "<username>:<password>" -X DELETE http://<serverhost>:<serverport>/kie-server/services/rest/server/containers/<containerID>
```

ここでは、

- **<username>** は、Decision Server にログオンして管理を行うアカウントのユーザー名です。
  - **<password>** は、アカウントのパスワードです。
  - **<containerID>** は、コンテナの識別子です。
  - **<serverhost>** は、KIE サーバーのホスト名です。KIE サーバーと同じホストでコマンドを実行する場合は **localhost** になります。
  - **<serverport>** は、KIE サーバーのポート番号です。
- 例:

```
curl --user "rhdmAdmin:password@1" -X DELETE http://localhost:39043/kie-server/services/rest/server/containers/kie1
```

## 付録A バージョン情報

Documentation last updated on: Monday, October 1, 2018.