



Red Hat Decision Manager 7.3

テストシナリオを使用したデシジョンサービスの テスト

ガイド

Red Hat Decision Manager 7.3 テストシナリオを使用したデシジョンサービスのテスト

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/Testing_a_decision_service_using_test_scenarios.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[®] 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 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 7.3 で、テストシナリオを使用してデシジョンサービスをテストする方法を説明します。

目次

はじめに	3
第1章 テストシナリオ	4
第2章 データオブジェクト	5
2.1. データオブジェクトの作成	5
第3章 BUSINESS CENTRAL でのテストシナリオデザイナーの使用	7
3.1. テストシナリオデザイナー	7
3.1.1. テストシナリオデザイナーのレイアウト	7
3.1.2. データオブジェクトのインポート	7
3.1.3. テストシナリオのインポート	8
3.2. テストシナリオデザイナーを使用した基本操作	8
3.2.1. テストシナリオの保存	9
3.2.2. テストシナリオの削除	9
3.2.3. テストシナリオの名前変更	9
3.2.4. テストシナリオのコピー	10
3.2.5. テストシナリオのダウンロード	10
3.2.6. テストシナリオのバージョン間の切り替え	11
3.2.7. アラートパネルの表示/非表示	11
3.3. コンテキストメニューのオプション	11
3.4. テストシナリオテンプレート	12
3.4.1. ルールベースシナリオのテストシナリオテンプレートの作成	13
3.4.2. エイリアスのコンセプト (ルールベースのシナリオのみ)	14
3.5. DMN ベーステストシナリオのテストテンプレート	14
3.5.1. DMN ベースのテストシナリオのテストシナリオテンプレート作成	14
3.6. テストシナリオの定義	14
3.7. テストシナリオデザイナーでのリストおよびマッピングコレクション	15
3.8. 式の構文	15
3.8.1. ルールベースシナリオの式構文	15
3.8.2. DMN ベースのテストシナリオでの式の構文	16
3.9. テストシナリオの実行	17
3.10. ローカルでのテストシナリオの実行	17
3.11. MORTGAGES サンプルプロジェクトを使用したテストシナリオの作成	18
第4章 BUSINESS CENTRAL でのテストシナリオ (レガシー) の使用	22
4.1. テストシナリオの作成および実行	22
4.1.1. テストシナリオへの GIVEN ファクトの追加	24
4.1.2. テストシナリオへの EXPECT 結果の追加	26
第5章 次のステップ	28
付録A バージョン情報	29

はじめに

ビジネス分析者またはビジネスルールの開発者は、Business Central でテストシナリオを使用して、プロジェクトをデプロイする前にデシジョンサービスをテストできます。DMN ベースおよびルールベースのデシジョンサービスが適切に、想定通りに機能することをテストできます。またデシジョンサービスは、プロジェクトの開発時にいつでもテストできます。

前提条件

- デシジョンサービスのチームおよびプロジェクトが Business Central に作成されています。詳細は、[デシジョンサービスのスタートガイド](#) を参照してください。
- ルールベースのデシジョンサービスに、ビジネスルールおよび関連するデータオブジェクトが定義されている。詳細は、[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](#) を参照してください。
- DMN ベースのデシジョンサービスに、DMN デシジョンロジックとその関連のカスタムデータタイプが定義されている。詳細は、[DMN モデルを使用したデシジョンサービスの作成](#) を参照してください。



注記

テストシナリオでは、ビジネスルールを設定するように定義したデータをテストできるため、ビジネスルールを先に定義しておくことは、テストシナリオにおける技術的な前提条件ではありません。ただし、先にルールを作成しておくこと、テストシナリオでルール全体をテストすることができ、かつ意図するデシジョンサービスにシナリオがより近づくため便利です。DMN ベースの場合には、テストシナリオを使用することで、DMN デシジョンロジックとその関連のカスタムデータタイプがデシジョンサービス用に定義されます。

第1章 テストシナリオ

Red Hat Decision Manager のテストシナリオでは、ビジネスルールを実稼働環境にデプロイする前に、(ルールベースのテストシナリオの場合) ビジネスルールの機能とデータの妥当性、および (DMN ベースのテストシナリオの場合) DMN モデルを検証できます。このテストシナリオでは、プロジェクトのデータを使用して、指定した条件と、定義した1つ以上のビジネスルールで想定される結果を設定できます。シナリオを実行する際は、想定した結果と、ルールのインスタンスから実際に得られた結果を比較します。想定される結果が実際の結果と一致すると、テストは成功します。想定された結果が実際の結果と一致しないと、テストは失敗します。

テストシナリオデザイナーでは、**.scsim** ファイルからすべてのシナリオを一度に実行できますが、レガシーのテストシナリオデザイナーでは、一度に1つずつ、またはグループ単位での実行になります。グループで実行する場合は、1つのパッケージに含まれるすべてのシナリオが対象になります。テストシナリオは独立しているため、別のシナリオに影響を及ぼしたり修正したりすることはありません。テストシナリオは、Business Central のプロジェクト開発時にいつでも実行できます。テストシナリオを実行するために、デシジョンサービスをコンパイルまたはデプロイする必要はありません。

別のパッケージからのデータオブジェクトは、テストシナリオと同じプロジェクトパッケージにインポートできます。同じパッケージに含まれるアセットはデフォルトでインポートされます。必要なデータオブジェクトとテストシナリオを作成したら、テストシナリオデザイナーの **Data Objects** タブを使用して、必要なデータオブジェクトがすべてリストされていることを検証するか、**アイテムを追加**して既存のデータオブジェクトをインポートします。

第2章 データオブジェクト

データオブジェクトは、作成するルールアセットの設定要素です。データオブジェクトは、プロジェクトで指定したパッケージに Java オブジェクトとして実装されているカスタムのデータタイプです。たとえば、データフィールド **Name**、**Address**、および **DateOfBirth** を使用して **Person** オブジェクトを作成し、ローン申し込みルールに詳細な個人情報を指定できます。このカスタムのデータ型は、アセットとデシジョンサービスがどのデータに基づいているかを指定します。

2.1. データオブジェクトの作成

次の手順は、データオブジェクトの作成の一般的な概要です。特定のビジネスアセットに固有のものではありません。

手順

1. Business Central で、**Menu** → **Design** → **Projects** に移動して、プロジェクト名をクリックします。
2. **Add Asset** → **Data Object** をクリックします。
3. 一意の **データオブジェクト** 名を入力し、**パッケージ** を選択します。これにより、その他のルールアセットでもデータオブジェクトを利用できるようになります。同じパッケージに、同じ名前のデータオブジェクトを複数作成することはできません。指定の DRL ファイルで、どのパッケージからでもデータオブジェクトをインポートできます。

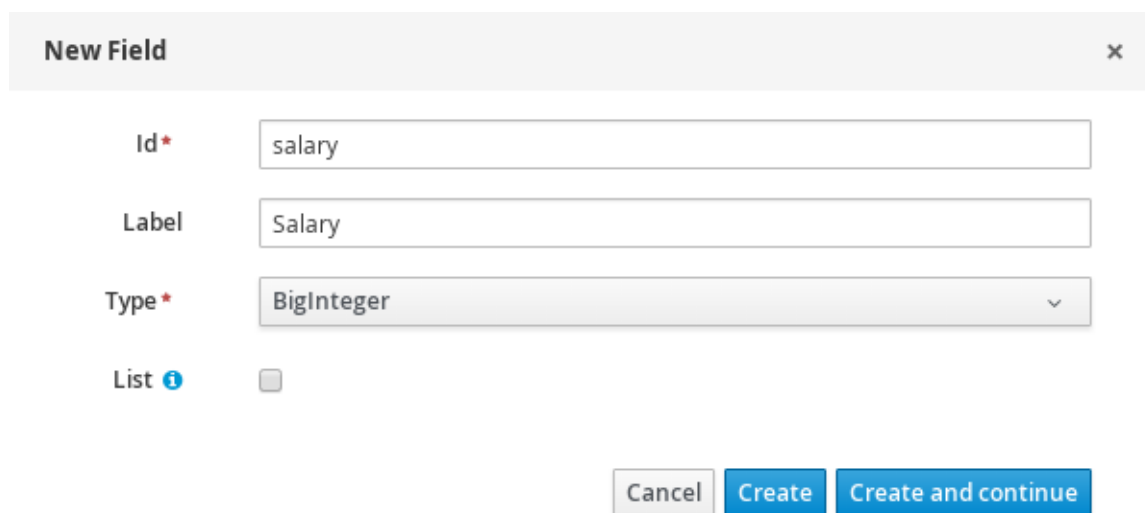


別のパッケージからのデータオブジェクトのインポート

別のパッケージから直接アセットデザイナーに、既存のデータオブジェクトをインポートすることができます。プロジェクトに関連するルールアセットを選択し、アセットデザイナーで **Data Objects** → **New item** に移動して、インポートするオブジェクトを選択します。

4. データオブジェクトを永続化するには、**Persistable** チェックボックスを選択します。永続型データオブジェクトは、JPA 仕様に準じてデータベースに保存できます。デフォルトの JPA は Hibernate です。
5. **OK** をクリックします。
6. データオブジェクトデザイナーで **add field** をクリックして、**Id** 属性、**Label** 属性、および **Type** 属性を使用するオブジェクトにフィールドを追加します。必須属性にはアスタリスク (*) マークが付いています。
 - **Id**: フィールドの一意の ID を入力します。
 - **Label**: (任意) フィールドのラベルを入力します。
 - **Type**: フィールドのデータ型を入力します。
 - **List**: このチェックボックスを選択すると、このフィールドで、指定したタイプのアイテムを複数保持できるようになります。

図2.1 データオブジェクトへのデータフィールドの追加



New Field x

Id* salary

Label Salary

Type* BigInteger

List

Cancel Create Create and continue

7. **Create** をクリックして、新しいフィールドを追加します。**Create and continue** をクリックすると、新しいフィールドが追加され、別のフィールドを引き続き作成できます。



注記

フィールドを編集するには、フィールド行を選択し、画面右側の **general properties** を使用します。

第3章 BUSINESS CENTRAL でのテストシナリオデザイナーの使用

Red Hat Decision Manager には、2つのデザイナーが同梱されており、テストシナリオを作成して実行し、ビジネスルールとデータを検証できます。この2つのデザイナーの相違点は、テストシナリオデザイナーでは DMN ベースと、ルールベース両方のテストシナリオを作成して実行できますが、テストシナリオ (レガシー) デザイナーではルールベースのテストシナリオの作成と実行しかできません。Red Hat では、どちらのデザイナーも実稼働環境での使用をサポートします。本章のセクションを確認すると、テストシナリオデザイナーを使用したテストシナリオの作成と実行方法が理解しやすくなります。



注記

現在、テストシナリオデザイナーは、デフォルトの KIE セッションしか利用できません。

3.1. テストシナリオデザイナー

Red Hat Decision Manager には、DMN モデルテストを完全にサポートする **テストシナリオデザイナー** が新たに同梱されるようになりました。このデザイナーは、DMN ベースのテストシナリオをテストするか、検証するときのユーザー体験を一新し、強化しています。この新しいデザイナーの目的は、作成したルールやナレッジベースが正しく機能していることをテストしやすくするためです。

3.1.1. テストシナリオデザイナーのレイアウト

デザイナーは、テーブル形式のレイアウトを導入し、シナリオテンプレートと関連のテストケースすべてを定義できるようにします。デザイナーレイアウトはヘッダーと個別の行を持つテーブルで設定されています。ヘッダーは、**GIVEN** と **EXPECT** の行、インスタンスの行、対応のフィールドの行の3つで設定されます。ヘッダーは、テストシナリオテンプレートとしても知られており、個別の行はテストシナリオ定義と呼ばれます。

テストシナリオテンプレートまたはヘッダーは以下の2つの部分で設定されます。

- **GIVEN** データオブジェクトおよびそのフィールド: 入力情報を表現します。
- **EXPECT** データオブジェクトおよびそのフィールド: オブジェクトとフィールドを表現します。実際の値が指定の情報をもとにチェックされ、想定の結果を設定するのに、この値を使用します。

テストシナリオの定義は、個別のテストケーステンプレートを表現します。

デザイナーの左パネルから **Project Explorer** にアクセスでき、右パネルからは、**Test Tools** と **Test Report** にアクセスできます。右パネルから **Test Tools** を使用をして、データオブジェクトのマッピング (**Test Editor**) を設定したり、参考として使用するノードが含まれるシート (**Scenario Cheatsheet**) にアクセスしたりできます。

3.1.2. データオブジェクトのインポート

テストシナリオデザイナーは、テストシナリオと同じパッケージ内に配置されている全データオブジェクトを読み込みます。すべてのデータオブジェクトは、デザイナーの **Data Objects** タブから表示できます。読み込んだデータオブジェクトは、**Test Tools** パネルの **Test Editor** タブにも表示されます。

データオブジェクトが変更された場合に (新規データオブジェクトの作成時や、既存のデータオブジェクトの削除時など) デザイナーを終了して、開き直す必要があります。一覧からデータオブジェクトを選択して、フィールドとフィールドタイプを表示します。

テストシナリオとは異なるパッケージに配置されているデータオブジェクトを使用する場合には、先にそのデータオブジェクトをインポートする必要があります。以下の手順に従い、ルールベースのテストシナリオ用にデータオブジェクトをインポートしてください。



注記

DMN ベースのテストシナリオの作成中にデータオブジェクトをインポートできません。DMN ベーステストシナリオは、プロジェクトからのデータオブジェクトを使用せず、DMN ファイルで定義したカスタムのデータタイプを使用します。

手順

1. **Test Scenario** デザイナーの **Project Explorer** パネルに移動します。
2. **Test Scenario** からテストシナリオを選択します。
3. **Data Objects** タブを選択して、**New Item** をクリックします。
4. **Add import** ウィンドウで、ドロップダウンリストからデータオブジェクトを選択します。
5. **Ok** をクリックしてから **Save** をクリックします。
6. **Test Scenario** デザイナーを終了して開きおなし、データオブジェクトリストから新しいデータオブジェクトを表示します。

3.1.3. テストシナリオのインポート

プロジェクトビューの **Asset** タブにある **Import Asset** ボタンを使用して、既存のテストシナリオをインポートできます。

手順

1. Business Central で、**Menu** → **Design** → **Projects** に移動して、プロジェクト名をクリックします。
2. プロジェクトの **Asset** タブから **Import Asset** をクリックします。
3. **Create new Import Asset** ウィンドウで、以下を実行します。
 - インポートセットの名前を入力します。
 - **Package** ドロップダウンリストからパッケージを選択します。
 - **Please select a file to upload** から、**Choose File...** をクリックしてテストシナリオファイルを参照します。
4. ファイルを選択して **Open** をクリックします。
5. **Ok** をクリックすると、レビューデザイナーでテストシナリオが開きます。

3.2. テストシナリオデザイナーを使用した基本操作

テストシナリオ デザイナーを使用して、以下の操作を実行します。

- テストシナリオの保存

- テストシナリオの削除
- テストシナリオの名前変更
- テストシナリオのコピー
- テストシナリオの実行
- テストシナリオのダウンロード
- テストシナリオの異なるバージョン間での切り替え
- アラートの表示または非表示

3.2.1. テストシナリオの保存

テストシナリオは、テストシナリオテンプレートの作成時や、テストシナリオの定義時にいつでも保存できます。

手順

1. 右上の **テストシナリオ デザイナー** をのツールバーから、**Save** をクリックします。
2. **Confirm Save** ウィンドウで、以下を実行します。
 - a. テストシナリオに関するコメントを追加する場合は、**add a comment** をクリックします。
 - b. もう一度 **Save** をクリックします。

テストシナリオが正常に保存されたことを示すメッセージが画面に表示されます。

3.2.2. テストシナリオの削除

テストシナリオ デザイナーを使用して作成した既存のテストシナリオを削除できます。

手順

1. 右上の **テストシナリオ デザイナー** をのツールバーから、**Delete** をクリックします。
2. **Confirm Delete** ウィンドウで、以下を実行します。
 - テストシナリオの削除に関するコメントを追加するには、**add a comment** をクリックします。
 - **Delete** をクリックします。

テストシナリオが正常に削除されたことを示すメッセージが画面に表示されます。

3.2.3. テストシナリオの名前変更

テストシナリオの名前変更は、デザイナーの右上にある **Rename** ボタンを使用します。

手順

1. 右上の **テストシナリオ デザイナー** をのツールバーから、**Rename** をクリックします。

2. **Rename Asset** ウィンドウで、以下を実行します。

- a. **Asset Name** フィールドに名前を入力します。
- b. コメントを追加するには、**add a comment** をクリックします。
- c. **Rename** をクリックします。



注記

Rename をクリックするだけで、テストシナリオファイルの名前が変更されます。

- d. 右上の **テストシナリオ デザイナー** をのツールバーから、**Save** をクリックします。
- e. **Confirm Save** ウィンドウで、以下を実行します。
 - i. コメントを追加するには、**add a comment** をクリックします。
 - ii. もう一度 **Save** をクリックします。
- f. **Save and Rename** をクリックして、同時にシナリオをの保存と、名前変更が可能です。

テストシナリオの名前が正常に変更されたことを示すメッセージが画面に表示されます。

3.2.4. テストシナリオのコピー

右上のツールバーの **Copy** ボタンを使用して、既存のテストシナリオを同じパッケージまたは別のパッケージにコピーできます。

手順

1. 右上の **テストシナリオ デザイナー** をのツールバーから、**Copy** をクリックします。
2. **Make a Copy** ウィンドウで、以下を実行します。
 - a. **New Name** フィールドに名前を入力します。
 - b. テストシナリオをコピーするパッケージを選択します。
 - c. オプションでコメントを追加するには、**add a comment** をクリックします。
 - d. **Make a Copy** をクリックします。

テストシナリオが正常にコピーされたことを示すメッセージが画面に表示されます。

3.2.5. テストシナリオのダウンロード

今後の参考に、またはバックアップとして、ローカルのマシンにテストシナリオのコピーをダウンロードできます。

手順

1. 右上の **テストシナリオ デザイナー** をのツールバーから、**Download** をクリックします。

.scsim ファイルがローカルのマシンにダウンロードされます。

3.2.6. テストシナリオのバージョン間の切り替え

Business Central には、テストシナリオのさまざまなバージョン間で切り替える機能があります。シナリオを保存するたびに、シナリオの新しいバージョンが **Latest Versions** に表示されます。この機能を使用するには、一度はテストシナリオファイルを保存しておく必要があります。

手順

1. 右上の **テストシナリオ デザイナー** をのツールバーから、**Latest Version** をクリックします。ファイルに複数バージョンがある場合は、ファイルの全バージョンが **Latest Version** に表示されます。
2. 作業するバージョンをクリックします。テストシナリオの選択したバージョンが **テストシナリオ デザイナー** で開きます。
3. デザイナーツールバーから **Restore** をクリックします。
4. **Confirm Restore** で以下を実行します。
 - a. コメントを追加するには、**add a comment** をクリックします。
 - b. **Restore** をクリックして確定します。

選択したバージョンが正常にデザイナーに再読み込まれたことを示すメッセージが画面に表示されます。

3.2.7. アラートパネルの表示/非表示

テストシナリオ デザイナーの下部に、テスト結果と **Alert** メッセージを含む、メッセージパネルが表示されます。右上のデザイナーツールバーから **Hide Alerts/View Alerts** をクリックしてレポートパネルを有効化または無効化します。

3.3. コンテキストメニューのオプション

テストシナリオ デザイナーには、コンテキストメニューオプションがあり、行と列の追加、削除、複製など、表での基本操作を実行できます。コンテキストメニューを使用するには、テーブル要素を右クリックする必要があります。メニューオプションは、選択するテーブル要素により異なります。

表3.1 コンテキストメニューのオプション

テーブル要素	セルのラベル	利用可能なコンテキストメニューオプション
Header	# & シナリオの説明	下に行を挿入
	GIVEN & EXPECT	左端に列を挿入、右端に列を挿入、下に行を挿入
	INSTANCE 1, INSTANCE 2 & PROPERTY 1, PROPERTY 2	左に列を挿入、右に列を挿入、列を削除、下に行を挿入
行		左に列を挿入、右に列を挿入、列を削除、下に行を挿入、上に行を挿入、行を複製、行を削除

表3.2 テーブルの操作の説明

テーブルの操作	説明
左端に列を挿入	(ユーザーの選択をもとにテーブルの GIVEN または EXPECT セクションで) 新たに左端に列を挿入します。
右端に列を挿入	(ユーザーの選択をもとにテーブルの GIVEN または EXPECT セクションで) 新たに右端に列を挿入します。
左に列を挿入	選択した列の左に新しい列を挿入します。新しい列は、(ユーザーの選択をもとにテーブルの GIVEN または EXPECT セクションに) 選択した列と同じタイプを挿入します。
右に行を挿入	選択した列の右に新しい列を挿入します。新しい列は、(ユーザーの選択をもとにテーブルの GIVEN または EXPECT セクションに) 選択した列と同じタイプを挿入します。
列を削除	選択した列を削除します。
上に行を挿入	選択した行の上に新しい行を挿入します。
下に行を挿入	選択した行の下に新しい行を挿入します。ヘッダーセルからの呼び出しの場合は、インデックス 1 の行を新たに挿入します。
行を複製	選択した行を複製します。
行を削除	選択した行を削除します。

Insert column right または **Insert column left** コンテキストメニューのオプションは以下の条件で動作が異なります。

- 選択した列にタイプが定義されていない場合には、タイプなしで新しい列が追加されます。
- 選択した列にタイプが定義されていない場合には、親のインスタンスタイプを引き継いだ列または新しい空の列が作成されます。
- アクションがインスタンスヘッダーから実行された場合には、タイプなしの列が新たに作成されます。
- アクションがプロパティのヘッダーから実行される場合には、親のインスタンスタイプを引き継いだ列が新たに作成されます。

3.4. テストシナリオテンプレート

テストシナリオの定義を指定する前に、テストシナリオテンプレートを作成する必要があります。テストシナリオテーブルのヘッダーにより、各シナリオのテンプレートが定義されます。GIVEN と EXPECT の両セクションに、インスタンスタイプとプロパティヘッダーを設定する必要があります。インスタンスヘッダーは、特定のデータオブジェクト (ファクト) にマッピングし、プロパティヘッダーは対応するデータオブジェクトの特定のフィールドにマッピングします。

テストシナリオ デザイナーを使用して、ルールベースと DMN ベース両方のシナリオのテストシナリオテンプレートを作成できます。

3.4.1. ルールベースシナリオのテストシナリオテンプレートの作成

以下の手順に従いルールベースのシナリオのテストシナリオテンプレートを作成してルールとデータを検証します。

手順

1. Business Central で、**Menu → Design → Projects** に移動して、テストシナリオを作成するプロジェクトをクリックします。
2. **Add Asset → Test Scenario** の順にクリックします。
3. **テストシナリオ** 名を入力し、適切な **パッケージ** を選択します。選択するパッケージは、必要なデータオブジェクトとルールアセットが割り当てられている、またはこれから割り当てるパッケージにする必要があります。
4. **Source type** で **RULE** を選択します。
5. **Ok** をクリックして、**テストシナリオ デザイナー** でテストシナリオを作成して開きます。
6. **GIVEN** コラムヘッダーをデータオブジェクトにマッピングするには、以下を実行します。
 - a. **GIVEN** セクションのインスタンスヘッダーをクリックします。
 - b. **Test Editor** タブからデータオブジェクトを選択します。
 - c. **Add** をクリックします。
7. データオブジェクトのプロパティをさらに挿入するには、プロパティヘッダーを右クリックして、必要に応じて、**Insert column right** または **Insert column left** を選択します。
8. データオブジェクトをプロパティセルにマッピングするには、以下を実行します。
 - a. プロパティセルをクリックします。
 - b. **Test Editor** タブからデータオブジェクトフィールドを選択します。
 - c. **Add** をクリックします。
9. **EXPECT** コラムヘッダーをデータオブジェクトにマッピングするには、以下を実行します。
 - a. **EXPECT** セクションのインスタンスヘッダーをクリックします。
 - b. **Test Editor** タブからデータオブジェクトを選択します。
 - c. **Add** をクリックします。
10. データオブジェクトのプロパティをさらに挿入するには、プロパティヘッダーを右クリックして、必要に応じて、**Insert column right** または **Insert column left** を選択します。
11. データオブジェクトをプロパティセルにマッピングするには、以下を実行します。
 - a. プロパティセルをクリックします。
 - b. **Test Editor** タブからデータオブジェクトフィールドを選択します。
 - c. **Add** をクリックします。

必要に応じて、コンテキストメニューを使用して列を追加または削除します。

データオブジェクトとそのプロパティに **GIVEN & EXPECT** 両方の列を作成してマッピングした後に、次のテストシナリオを定義する必要があります。

3.4.2. エイリアスのコンセプト (ルールベースのシナリオのみ)

テストシナリオデザイナーで、データオブジェクトにヘッダーセルがマッピングされると、データオブジェクトは **Test Tools** パネルの **Test Editor** タブから削除されます。エイリアスのコンセプトを使用して、データオブジェクトを別のヘッダーセルに再マッピングできます。エイリアスでは、同じデータオブジェクトの複数インスタンスを指定できます。エイリアスの名前が一意であることを確認してください。こうすることで、データオブジェクトはもう一度、**Test Editor** タブで利用でき再マッピングが可能になります。エイリアスを追加するには、対応するヘッダーセルをダブルクリックして、名前を手動で変更します。プロパティのエイリアスを作成することも可能です。プロパティのエイリアスを作成すると、テーブルで直接使用するプロパティの名前を変更できます。

3.5. DMN ベーステストシナリオのテストテンプレート

Business Central は、DMN ベースの全テストシナリオアセットに対してテンプレートを生成し、この中に、関連の DMN モデルの指定のインプットやデシジョンがすべて含まれます。DMN モデルの入力ノードごとに、**GIVEN** 列が追加され、各デシジョンノードは **EXPECT** 列で表現されます。デフォルトのテンプレートは、必要に応じていつでも変更できます。また、DMN モデルの一部のみをテストするには、生成した列を削除することも、**EXPECT** から **GIVEN** セクションのデシジョンノードを移動することが可能です。

3.5.1. DMN ベースのテストシナリオのテストシナリオテンプレート作成

以下の手順に従い、DMN ベースのシナリオのテストシナリオテンプレートを作成し、DMN モデルを検証します。

手順

1. Business Central で、**Menu** → **Design** → **Projects** に移動して、テストシナリオを作成するプロジェクトをクリックします。
2. **Add Asset** → **Test Scenario** の順にクリックします。
3. テストシナリオ名を入力し、適切な **パッケージ** を選択します。
4. **Source type** で **DMN** を選択します。
5. **Choose DMN asset** オプションを使用して、既存の DMN アセットを選択します。
6. **Ok** をクリックして、テストシナリオデザイナーでテストシナリオを作成して開きます。

テンプレートは自動的に生成され、ニーズに合わせてこのテンプレートを変更できます。

3.6. テストシナリオの定義

テストシナリオテンプレートの作成後、テストシナリオを定義する必要があります。テストシナリオテーブルの行では、個別のテストシナリオを定義します。テストシナリオには、一意のインデックス番号、説明、入力値セット (**Given** の値)、出力値セット (**Expect** の値) を定義します。

前提条件

選択したテストシナリオに、テストシナリオテンプレートが存在することを確認します。

手順


1. テストシナリオ デザイナーでテストシナリオを開きます。
2. テストシナリオの説明を入力して、行のセルに必要な値を入力します。
3. コンテキストメニューを使用して、必要に応じて行を追加または削除します。
セルをダブルクリックしてインラインの編集を開始します。テスト評価で特定のセルを省略するには、そのセルを空白のままにします。

テストシナリオの定義したら、テストを実行できます。

3.7. テストシナリオデザイナーでのリストおよびマッピングコレクション

テストシナリオ デザイナーは、DMN ベースとルールベース両方のテストシナリオのリストおよびマッピングコレクションをサポートします。リストやマッピングのようなコレクションは、**GIVEN** および **EXPECT** の両コラムに、特定のセルの値として定義できます。

手順

1. 列タイプを先に設定します (タイプがリストまたはマッピングのフィールドを使用します)。
2. 列内のセルをダブルクリックして、値を入力します。
3. コレクションエディターのポップアップで、**Add new item** をクリックします。
4. 必要な値を入力して、チェックアイコン  をクリックし、追加した各コレクションアイテムを保存します。
5. **Save** をクリックします。
コレクションからアイテムを削除するには、コレクションのポップアップエディターでゴミ箱のアイコンをクリックします。**Remove** をクリックして、コレクション自体を削除します。

3.8. 式の構文

テストシナリオデザイナーでは、ルールベースおよび DMN ベースのテストシナリオにおいて異なる式言語をサポートします。ルールベースのテストシナリオは、基本的な式言語を、DMN ベースのテストシナリオは FEEL 式言語をサポートします。

3.8.1. ルールベースシナリオの式構文

テストシナリオ デザイナーでは、以下に示すルールベースのテストシナリオの定義式がサポートされます。

表3.3 式構文の説明

Operator	説明
=	値と同等であることを指定します。これは、全列でデフォルトとなっており、GIVEN 列でサポートされる唯一の演算子です。

Operator	説明
$=$, \neq , $\langle \rangle$	値が同等でないことを指定します。この演算子は、他の演算子と組み合わせることができます。
\langle , \rangle , \leq , \geq	比較 (未満、値よりも大きい、以下、以上) を指定します。
[value1, value2, value3]	値の一覧を指定します。1つまたは複数の値が有効な場合は、シナリオの定義が True と評価されます。
expression1; expression2; expression3	式のリストを指定します。すべての式が有効な場合は、シナリオ定義が True と評価されます。



注記

ルールベースのテストシナリオを評価する場合、空のセルは評価からスキップされます。空の文字列を定義するには、 $=$ 、 $[]$ 、または $;$ を使用します。null 値を定義するには、**null** を使用します。

表3.4 式の例

式	説明
-1	実際の値は、-1 と同等です。
$\langle 0$	実際の値は、0 未満です。
$! \rangle 0$	実際の値は、0 以下です。
[-1, 0, 1]	実際の値は、-1、0、または 1 です。
$\langle \rangle [1, -1]$	実際の値は、1 でも -1 でもありません。
$!100; 0$	実際の値は、100 ではなく、0 です。
$! \leq 0; \langle \rangle > 1$	実際の値は、0 未満でも、1 以上でもありません。
$\langle \rangle \leq 0; \geq 1$	実際の値は、0 以下でなく、1 以上です。

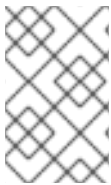
ルールベースのテストシナリオデザイナーの右側にある **Scenario Cheatsheet** タブで、サポートされているコマンドと構文を参照できます。

3.8.2. DMN ベースのテストシナリオでの式の構文

以下のデータタイプは、テストシナリオデザイナーの DMN ベーステストシナリオでサポートされません。

表3.5 DMN ベースのシナリオがサポートするデータタイプ

サポートされるデータタイプ	説明
番号および文字列	文字列は、" John Doe "、" Brno "、または "" のように、引用符で囲む必要があります。
ブール値	true 、 false 、および null 。
日時	例: date("2019-05-13") または time("14:10:00+02:00")
関数	avg 、 max などの組み込み計算機能をサポートします。
コンテキスト	例: {x : 5, y : 3}
範囲とリスト	例: [1 ..10] または [2, 3, 4, 5] です。



注記


DMN ベースのテストシナリオを評価する場合、空のセルは評価からスキップされます。DMN ベースのテストシナリオで空の文字列を定義するには "" を使用し、**null** 値を定義するには **null** を使用します。

DMN ベースのテストシナリオデザイナーの右側にある **Scenario Cheatsheet** タブで、サポートされているコマンドと構文を参照できます。


3.9. テストシナリオの実行

テストシナリオテンプレートを作成してテストシナリオを定義したら、テストを実行してビジネスルールとデータを検証できます。

手順

1. テストシナリオ デザイナーの上部にある **テストの実行**  アイコンをクリックします。
2. **Test Report** パネルでは、デザイナーの右側にポップアップ表示され、テスト結果の概要とシナリオのステータスが表示されます。
3. テストの実行後に、**View Alerts** をクリックして、デザイナーの下部にある **Alerts** パネルを開き、エラーメッセージを確認します。
4. テストに失敗したら、エラーメッセージを確認します。赤いバツマークの付いたメッセージは、テストに失敗したことを示し、緑のチェックマークはテストに成功したことを示します。デザイナーでは、エラーのある行やセルが赤で強調表示されます。
5. 必要な変更を加えて、テストに成功するまでもう一度テストを実行します。

3.10. ローカルでのテストシナリオの実行

Red Hat Decision Manager では 2 種類の方法でテストを実行できます。1つの方法は、テストシナリオデザイナーの上部にある **テストの実行**  アイコンをクリックすることです。もう1つの方法は、コマンドラインを使用してローカルでテストを実行することです。

手順

1. Business Central で、**Menu → Design → Projects** に移動して、プロジェクト名をクリックします。
2. プロジェクトのホームページで **Settings** タブを選択します。
3. **git URL** を選択し、**Clipboard**  をクリックして git url をコピーします。
4. コマンドターミナルを開いて、git プロジェクトのクローンを作成するディレクトリーに移動します。
5. プロジェクトのディレクトリーから、以下のコマンドを実行します。

```
git clone your_git_project_url
```

your_git_project_url は、**git://localhost:9418/MySpace/ProjectTestScenarios** などの適切なデータに置き換えます。

6. プロジェクトのクローンを正しく作成したら、git プロジェクトディレクトリーに移動して、以下のコマンドを実行します。

```
mvn clean test
```

プロジェクトのビルド情報およびテスト結果 (テスト実行回数およびテスト実行が成功したかどうかなど) は、コマンドターミナルに表示されます。エラーがある場合には、修正して、もう一度コマンドを実行します。

3.11. MORTGAGES サンプルプロジェクトを使用したテストシナリオの作成

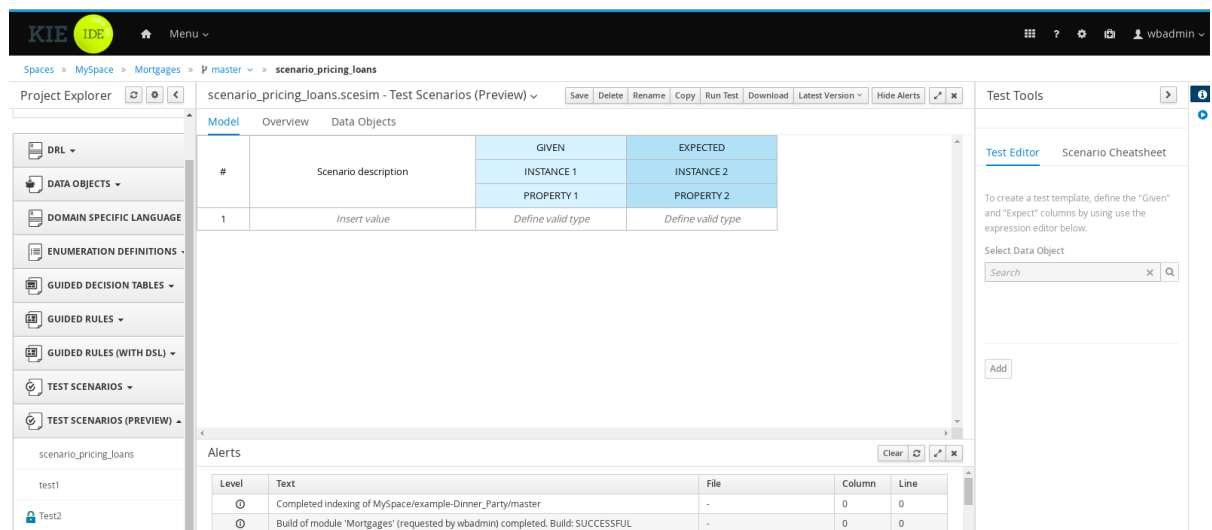
本章では、テストシナリオデザイナーを使用して、Business Central に同梱されているサンプルの **Mortgages** プロジェクトからテストシナリオを作成して実行する方法を説明します。本章のテストシナリオの例は、**Mortgages** プロジェクトからの **Pricing loans** のガイド付きデシジョンテーブルに基づいています。

手順

1. Business Central にログインし、**Menu → Design → Projects** の順にクリックし、**Mortgages** をクリックします。
2. プロジェクトが **Projects** の下にリストされていない場合は、**MySpace** から 3 つのドット () ページの右上隅にあります。
3. **Try Samples → Mortgages → OK** の順にクリックします。アセットのウィンドウが表示されます。
4. **Add Asset → Test Scenario** の順にクリックします。

5. **Test Scenario** の名前として、**scenario_pricing_loans** を入力し、**Package** ドロップダウンリストから、デフォルトの **mortgages.mortgages** パッケージを選択します。
選択するパッケージには、必要なルールアセットがすべて含まれている必要があります。
6. **Ok** をクリックして、テストシナリオ デザイナーでテストシナリオを作成して開きます。
7. **Project Explorer** を展開して以下を確認します。
 - **Applicant**、**Bankruptcy**、**IncomeSource**、および **LoanApplication** データオブジェクトが存在する。
 - **Pricing loans** ガイド付きのデシジョンテーブルが存在する。
 - 新しいテストシナリオが **Test Scenario** に表示されていることを確認する。
8. データオブジェクトがすべて配置されていることを確認してから、テストシナリオデザイナーの **Model** タブに戻り、利用可能なデータオブジェクトのシナリオに、**GIVEN** データと **EXPECT** データを定義します。

図3.1 空のテストシナリオデザイナー



9. **GIVEN** コラムの詳細を定義します。
 - a. **GIVEN** 列ヘッダーにある **INSTANCE 1** という名前のセルをクリックします。
 - b. **Test Tools** → **Test Editor** から **LoanApplication** データオブジェクトを選択します。
 - c. **Add** をクリックします。
10. データオブジェクトのプロパティを作成するには、必要に応じて **Insert column right** または **Insert column left** を選択します。この例では、**GIVEN** 列の下に。プロパティセルをさらに 2 つ作成する必要があります。
11. 最初のプロパティセルをクリックします。
 - a. **Test Tools** → **Test Editor** から **LoanApplication** データオブジェクトを選択して展開します。
 - b. **amount** をクリックして、追加し、プロパティセルにデータオブジェクトフィールドをマッピングします。
12. 2 つ目のプロパティセルをクリックします。

- a. **Test Tools** → **Test Editor** から **LoanApplication** データオブジェクトを選択して展開します。
 - b. **deposit** をクリックしてから **Add** をクリックします。
13. 2つ目のプロパティセルをクリックします。
 - a. **Test Tools** → **Test Editor** から **LoanApplication** データオブジェクトを選択して展開します。
 - b. **lengthYears** をクリックしてから **Add** をクリックします。
14. **LoanApplication** ヘッダーセルを右クリックし、**Insert column right** を選択します。右側に新しい **GIVEN** 列が作成されます。
15. 新しいヘッダーセルをクリックします。
 - a. **Test Tools** → **Test Editor** から **IncomeSource** データオブジェクトを選択します。
 - b. **Add** をクリックして、データオブジェクトをヘッダーセルにマッピングします。
16. **IncomeSource** の下のプロパティセルをクリックします。
 - a. **Test Tools** → **Test Editor** から **IncomeSource** データオブジェクトを選択して展開します。
 - b. **type** をクリックして、**追加** し、プロパティセルにデータオブジェクトフィールドをマッピングします。
GIVEN 列セルがすべて定義されました。
17. 次に **EXPECT** コラムの詳細を定義します。
 - a. **EXPECT** 列ヘッダーにある **INSTANCE 2** という名前のセルをクリックします。
 - b. **Test Tools** → **Test Editor** から **LoanApplication** データオブジェクトを選択します。
 - c. **Add** をクリックします。
18. データオブジェクトのプロパティを作成するには、必要に応じて **Insert column right** または **Insert column left** を選択します。**EXPECT** 列の下に、プロパティセルをさらに2つ作成します。
19. 最初のプロパティセルをクリックします。
 - a. **Test Tools** → **Test Editor** から **LoanApplication** データオブジェクトを選択して展開します。
 - b. **approved** をクリックして、**追加** し、プロパティセルにデータオブジェクトフィールドをマッピングします。
20. 2つ目のプロパティセルをクリックします。
 - a. **Test Tools** → **Test Editor** から **LoanApplication** データオブジェクトを選択して展開します。
 - b. **insuranceCost** をクリックしてから **Add** をクリックします。
21. 2つ目のプロパティセルをクリックします。

- a. **Test Tools** → **Test Editor** から **LoanApplication** データオブジェクトを選択して展開します。
 - b. **approvedRate** をクリックしてから **Add** をクリックします。
22. 次に、テストシナリオを定義するには、1行目に以下のデータを入力します。
- **GIVEN** コラムの値として、**Scenario Description** には **Row 1 test scenario**、**amount** には **150000**、**deposit** には **19000**、**lengthYears** には **30**、**type** には **Asset** を入力します。
 - **EXPECT** コラムの値として、**approved** には **true**、**insuranceCost** には **0** を入力します。
23. 次に2番目の行に、以下のデータを入力します。
- **GIVEN** 列の値として、**Scenario Description** には **Row 2 test scenario**、**amount** には **100002**、**deposit** には **2999**、**lengthYears** には **20**、**type** には **Job** を入力します。
 - **EXPECT** コラムの値として、**approved** には **true**、**insuranceCost** には **10** を入力します。
24. シナリオに対して、**GIVEN**、**EXPECT**、その他のデータをすべて定義したら、テストシナリオデザイナーで **Save** をクリックして、設定した内容を保存します。
25. 右上隅の **Run Test** をクリックして **.scsim** ファイルを実行します。
テスト結果は、**Test Report** パネルに表示されます。**View Alerts** をクリックして、**Alerts** セクションからメッセージを表示します。テストに失敗した場合は、ウィンドウの下部にある **Alerts** セクションのメッセージを参照し、シナリオの全コンポーネントをレビューして修正してから、シナリオが合格するまでシナリオの検証を再度行います。
26. テストシナリオデザイナーで **Save** をクリックして、必要な変更をすべて加えてから、作業を保存します。

第4章 BUSINESS CENTRAL でのテストシナリオ (レガシー) の使用

Red Hat Decision Manager には、テストシナリオデザイナーが2つあります。1つ目は、テストシナリオデザイナー、もう1つは **テストシナリオ (レガシー) デザイナー** です。ビジネスルールとデータのテストにはいずれを使用しても構いません。テストシナリオ デザイナーはデフォルトのデザイナーで、これを使用して DMN ベースのテストシナリオを作成し、実行できます。本章では、**テストシナリオ (レガシー) デザイナー** を使用してテストシナリオを作成し、実行する方法を説明します。テストシナリオ (レガシー) デザイナーは、DRL ファイルとガイド付きのデシジョンテーブルをもとに実行します。

4.1. テストシナリオの作成および実行

ビジネスルールデータをデプロイする前に、Business Central にテストシナリオを作成して、その機能をテストできます。基本的なテストシナリオには、少なくとも以下のデータが必要です。

- 関連するデータオブジェクト
- **GIVEN (指定した) ファクト**
- **EXPECT (想定される) 結果**

テストシナリオでは、このデータを使用し、定義したファクトに基づいて、そのルールインスタンスに対して想定した結果と実際の結果の妥当性を検証できます。**CALL METHOD**と利用可能な **globals** をテストシナリオに追加することもできますが、これは必須ではありません。

手順

1. Business Central で、**Menu → Design → Projects** に移動して、プロジェクト名をクリックします。
2. **Add Asset → テストシナリオ (レガシー)** の順にクリックします。
3. **テストシナリオ** 名を入力し、適切な **パッケージ** を選択します。指定するパッケージは、必要なルールアセットが割り当てられている、またはこれから割り当てるパッケージと同じにする必要があります。データオブジェクトは、パッケージからアセットのデザイナーにインポートできます。
4. **OK** をクリックして、テストシナリオを作成します。
Project Explorer の **Test Scenarios** パネルに、新しいテストシナリオが追加されました。
5. **Data Objects** タブをクリックして、テストするルールに必要なデータオブジェクトがすべてリストされていることを検証します。追加されていない場合は、**New item** をクリックして別のパッケージから必要なデータオブジェクトをインポートするか、パッケージに **データオブジェクトを作成** します。
6. データオブジェクトをすべて配置したら、テストシナリオデザイナーの **Model** タブに戻り、利用可能なデータオブジェクトに基づいたシナリオに、**GIVEN** データと **EXPECT** データを定義します。

図4.1 テストシナリオデザイナー

The screenshot shows the Test Scenario Designer interface with the following components:

- + GIVEN** button
- Block 1: Insert 'Applicant' [a]
 - Field: age: 17
 - Button: 'Applicant' facts
- Block 2: Insert 'LoanApplication' [application]
 - Field: amount: 1
 - Button: 'LoanApplication' facts
- Block 3: Insert 'IncomeSource' [incomeSource]
 - Field: Add a field
 - Button: 'IncomeSource' facts
- + CALL METHOD** button
- Text: Add input data and expectations here.
- + EXPECT** button
- Block 4: LoanApplication 'application' has values:
 - Field: approved: equals
 - Field: false
 - Button: 'application'
- Button: Delete one scenario block above
- Button: More...
- + (globals)** button

GIVEN セクションには、テストする入力ファクトを定義します。たとえば、プロジェクトの **Underage** ルールで、ローン申請者の年齢が 21 歳未満であれば承認しない場合は、テストシナリオの **GIVEN** ファクトで、**Applicant** の **age** に、21 より小さい数字に設定する必要があります。

EXPECT セクションには、**GIVEN** に入力したファクトに基づいて想定される結果を定義します。つまり、入力ファクトを **GIVEN (指定)** すると、その他のファクトが有効であること、またはルール全体が有効であることを **EXPECT (想定)** します。たとえば、このシナリオで、申請者が 21 歳未満の場合に **想定される** 結果は、(申請者の年齢が基準を満たさないため) **LoanApplication** の **approved** が **false** になるか、**Underage** ルール全体が有効になります。

7. **CALL METHOD** と **globals** をテストシナリオに追加することもできます。

- **CALL METHOD**: ルールの実行を開始する際に、別のファクトからメソッドを呼び出します。**CALL METHOD** をクリックし、ファクトを選択し、▶ をクリックして呼び出すメソッドを選択します。(可能な場合は) プロジェクトに対してインポートした Java ライブラリー、または JAR から (ArrayList のメソッドなどの) Java クラスメソッドを呼び出すことができます。
- **globals**: テストシナリオで妥当性を確認するプロジェクトにグローバル変数を追加します。**globals** をクリックして、妥当性を確認する変数を選択し、テストシナリオデザイナーでグローバル名をクリックして、グローバル変数に適用するフィールド値を定義します。グローバル変数が利用できない場合は、Business Central に新しいアセットとして作成する

必要があります。グローバル変数はデシジョンエンジンに表示されますが、ファクトに対するオブジェクトとは異なるオブジェクトの名前です。グローバルのオブジェクトを変更しても、ルールの再評価は行われません。

8. 必要に応じて、テストシナリオデザイナーの下部で **More** をクリックし、同じシナリオファイルに別のデータブロックを追加します。
9. シナリオに対して、**GIVEN**、**EXPECT**、その他のデータをすべて定義したら、テストシナリオデザイナーで **Save** をクリックして、設定した内容を保存します。
10. 右上の **Run scenario** をクリックして、この **.scenario** ファイルを実行します。プロジェクトパッケージに **.scenario** ファイルが複数ある場合にすべて保存している場合は、**Run all scenarios** をクリックして、すべてのシナリオを実行します。**Run scenario** オプションでは、個々の **.scenario** ファイルを保存する必要はありませんが、**Run all scenarios** オプションを使用する場合は、すべての **.scenario** ファイルを保存する必要があります。テストに失敗したら、ウィンドウ下部の **Alerts** メッセージに記載されている問題に対応し、シナリオの全コンポーネントを見直し、エラーが表示されなくなるまで妥当性確認を行います。
11. 変更がすべて終了したら、テストシナリオデザイナーで **Save** をクリックして、設定した内容を保存します。

GIVEN ファクトをテストシナリオに追加する方法は [「テストシナリオへの GIVEN ファクトの追加」](#) を参照してください。

EXPECT 結果をテストシナリオに追加する方法は [「テストシナリオへの EXPECT 結果の追加」](#) を参照してください。

4.1.1. テストシナリオへの GIVEN ファクトの追加

GIVEN セクションには、テストする入力ファクトを定義します。たとえば、プロジェクトの **Underage** ルールで、ローン申請者の年齢が 21 歳未満であれば承認しない場合は、テストシナリオの **GIVEN** ファクトで、**Applicant** の **age** に、21 より小さい数字に設定する必要があります。

前提条件

テストシナリオに必要なデータオブジェクトがすべて作成、またはインポートされていて、テストシナリオ (レガシー) デザイナーの **Data Objects** タブにリストされています。

手順


1. テストシナリオ (レガシー) デザイナーで、**GIVEN** をクリックして、利用可能なファクトが含まれる **New input** ウィンドウを表示します。

図4.2 テストシナリオへの GIVEN 入力への追加

リストには以下のオプションが含まれます。表示されるオプションは、テストシナリオデザイナーの **Data Objects** タブで利用可能なデータオブジェクトによって異なります。

- **Insert a new fact:** ファクトを追加して、フィールド値を修正します。 **Fact name** にファクトの変数を入力します。
 - **Modify an existing fact:** 別のファクトが追加される場合に限り表示されます。これを使用して、シナリオの実行間でデシジョンエンジンで変更される前に挿入されたファクトを指定します。
 - **Delete an existing fact:** 別のファクトが追加される場合に限り表示されます。これを使用して、シナリオの実行間でデシジョンエンジンで削除される前に挿入されたファクトを指定します。
 - **Activate rule flow group:** そのグループ内にあるすべてのルールをテストできるように、有効にするルールフローグループを指定します。
2. 目的の入力オプションに対するファクトを選択し、**Add** をクリックします。たとえば、**Insert a new fact:** に **Applicant** を設定し、**Fact name** に対して **a** または **app**、もしくは別の変数を入力します。
 3. テストシナリオデザイナーのファクトをクリックし、修正するフィールドを選択します。

図4.3 ファクトフィールドの修正

4. 編集アイコン () をクリックし、以下のフィールド値を選択します。
 - **Literal value:** 特定のリテラル値を入力するオープンフィールドを作成します。

- **Bound variable:** このフィールドの値を、選択した変数にバインドするファクトに設定します。フィールドタイプが、バインドした変数型に一致する必要があります。
 - **Create new fact:** 新しいファクトを作成し、そのファクトを親ファクトのフィールド値として割り当てます。テストシナリオデザイナーで子ファクトをクリックし、同じようにフィールド値を割り当てるか、別のファクトをネストできます。
5. 続いて、シナリオに別の **GIVEN** 入力データを追加し、テストシナリオデザイナーで **Save** をクリックして、設定した内容を保存します。

4.1.2. テストシナリオへの EXPECT 結果の追加

EXPECT セクションには、**GIVEN** に入力したファクトに基づいて想定される結果を定義します。つまり、入力ファクトを **GIVEN (指定)** すると、その他のファクトが有効であること、またはルール全体が有効であることを **EXPECT (想定)** します。たとえば、このシナリオで、申請者が 21 歳未満の場合に **想定される結果は、(申請者の年齢が基準を満たさないため) LoanApplication の approved が false になるか、Underage ルール全体が有効になります。**

前提条件

テストシナリオに必要なデータオブジェクトがすべて作成、またはインポートされていて、テストシナリオ (レガシー) デザイナーの **Data Objects** タブにリストされています。

手順

1. テストシナリオ (レガシー) デザイナーで、**EXPECT** をクリックして、利用可能なファクトが含まれる **New expectation** ウィンドウを表示します。

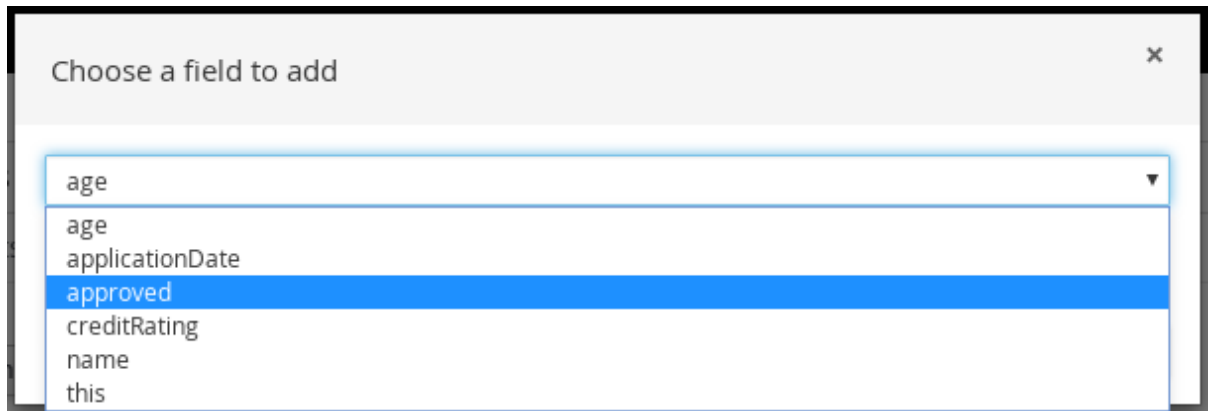
図4.4 テストシナリオへの EXPECT 結果の追加

リストには以下のオプションが含まれます。表示されるオプションは、**GIVEN** セクションのデータや、テストシナリオデザイナーの **Data Objects** タブで利用可能なデータオブジェクトによって異なります。

- **Rule:** プロジェクトに、**GIVEN** に指定した内容に対して有効になることが想定される特定のルールを指定します。ルールの名前を入力するか、ルールリストから選択します。次に、テストシナリオデザイナーで、ルールが有効になるべき回数を指定します。
- **Fact value:** ファクトを選択し、**GIVEN** セクションに定義したファクトに対して有効になることが想定される値を定義します。ファクトは、**GIVEN** の入力に対して事前に定義した **Fact name** でリストされます。
- **Any fact that matches:** **GIVEN** に指定した内容に対して、指定した値を持つファクトが最低 1 つ存在するかどうかの妥当性を確認します。

2. (Fact value: **application** などの) 期待される結果のファクトを選択し、**Add** または **OK** を選択します。
3. テストシナリオデザイナーでファクトをクリックし、追加または修正するフィールドを選択します。

図4.5 ファクトフィールドの修正



4. フィールド値に、**GIVEN** に指定した内容に対して、有効になると想定される値 (**approved | equals | false** など) を設定します。
5. 続いて、シナリオに別の **EXPECT** 入力データを追加し、テストシナリオデザイナーで **Save** をクリックして、設定内容を保存します。
6. シナリオに **GIVEN**、**EXPECT**、その他のデータを定義して保存したら、右上の **Run scenario** をクリックしてこの **.scenario** ファイルを実行するか、(複数の場合は) **Run all scenarios** をクリックして、プロジェクトパッケージに保存したすべての **.scenario** を実行します。 **Run scenario** オプションでは、個々の **.scenario** ファイルを保存する必要はありませんが、**Run all scenarios** オプションを使用する場合は、すべての **.scenario** ファイルを保存する必要があります。
テストに失敗したら、ウィンドウ下部の **Alerts** メッセージに記載されている問題に対応し、シナリオの全コンポーネントを見直し、エラーが表示されなくなるまで妥当性確認を行います。
7. 変更がすべて終了したら、テストシナリオデザイナーで **Save** をクリックして、設定した内容を保存します。

第5章 次のステップ

Red Hat Decision Manager プロジェクトのパッケージ化およびデプロイ

付録A バージョン情報

本書の最終更新日: 2021年11月15日(月)