Red Hat Process Automation Manager 7.0

Migrating from Red Hat JBoss BPM Suite 6.4 to Red Hat Process Automation Manager 7.0

Last Updated: 2019-11-12

Red Hat Process Automation Manager 7.0 Migrating from Red Hat JBoss BPM Suite 6.4 to Red Hat Process Automation Manager 7.0

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

Legal Notice

Abstract

This document describes how to migrate project data and server environments from Red Hat JBoss BPM Suite 6.4 to Red Hat Process Automation Manager 7.0.

Table of Contents

PREFACE

As a system administrator or business rules developer, you can migrate your existing project data and server environment in Red Hat JBoss BPM Suite 6.4 to Red Hat Process Automation Manager 7.0. Red Hat provides migration tools wherever possible to facilitate migration, but in some cases, manual migration or asset re-creation is required.

Prerequisites

- Red Hat JBoss BPM Suite 6.4 is installed and contains artifacts that you want to migrate to Red Hat Process Automation Manager 7.0.

- Red Hat Process Automation Manager 7.0 is installed on a separate instance from Red Hat JBoss BPM Suite 6.4 (on Red Hat JBoss EAP or other instance). For installation options, see *Planning a Red Hat Process Automation Manager installation* .

CHAPTER 1. MIGRATION OVERVIEW

If you use Red Hat JBoss BPM Suite version 6.4 and install Red Hat Process Automation Manager version 7.0, you need to migrate the applications that you created and possibly your server environment to the new product. Red Hat provides migration and upgrade tools wherever possible to facilitate migration, but in some cases, manual migration or asset re-creation is required.

The following are the main changes in Red Hat Process Automation Manager 7.0 that affect product migration:

- **New repository structure for Business Central project data:**Business assets that you created in Business Central with Red Hat JBoss BPM Suite 6.4 must be migrated using the Business Central migration tool provided with this release to accommodate a new data structure. Project data is restructured in Red Hat Process Automation Manager 7.0 so that each space (previously organizational unit) contains repositories that correspond to individual projects, instead of multiple projects in a single repository as in Red Hat JBoss BPM Suite 6.4. This improved structure means that you do not need to create and manage repositories in the new Business Central and can focus on developing Business assets.

- **No execution server capabilities in Business Central:**In Red Hat Process Automation Manager 7.0, Business Central no longer has execution server capabilities nor any API endpoints related to the execution server. Instead, the Process Server performs all execution and runtime functions and Business Central is used for application development and management only. Therefore, any Business Central environments that are currently used as an execution server must be properly migrated to the new Business Central and Process Server environments to retain live tasks and process instances in Red Hat Process Automation Manager 7.0. If any applications interact with the Business Central execution server through the remote Java API or through direct REST API calls, these API references must be redirected to the Process Server API.

- **Upgraded database schema:** Red Hat Process Automation Manager 7.0 contains updates to the database schema for optimal performance with the new Business Central and Process Server. The Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 must be updated using one of the upgrade scripts provided with this release to accommodate the change.

The following migration paths are available, depending on your Red Hat JBoss BPM Suite 6.4 environment:

- Chapter 3, *Migrating Business Central as design environment only*
- Chapter 4, *Migrating Business Central as design environment and execution server*
- Chapter 5, *Migrating Java project data*

IMPORTANT

If you are using a version of Red Hat JBoss BPM Suite older than 6.4, migrate your projects to version 6.4 before migrating to 7.0. For migration instructions, see the Red Hat JBoss BPM Suite 6.4 Migration Guide

For more information about updates and new features in this release, see the *Release notes for Red Hat Process Automation Manager 7.0*.

CHAPTER 2. FUNCTIONALITY OF MIGRATED ASSETS IN BUSINESS CENTRAL

Several asset designers have been improved in Business Central for Red Hat Process Automation Manager 7.0. As a result of this ongoing effort to improve Business Central, some previously supported asset features and functions have changed or cannot currently be migrated. Before migrating your project data, consider the following assets with different feature support in Red Hat Process Automation Manager 7.0:

Forms

Forms are migrated to Red Hat Process Automation Manager 7.0 by a forms migration tool that is part of the Business Central migration tool. Migrated forms are managed in a new forms designer that currently does not support all features of the previous forms designer in Red Hat JBoss BPM Suite 6.4. The migration tool and the forms designer will be incrementally improved in subsequent releases to better support currently deprecated features. Therefore, avoid deleting any migrated forms that lose functionality in migration because they may be supported in the future.
The following limitations and considerations apply to form migration:

- Existing forms in Red Hat JBoss BPM Suite 6.4 must be correct. If a conflict arises during migration, some forms might not be migrated.
- Forms related to business processes (forms that end with a **taskform** suffix) no longer support fields with nested bindings (such as client/name) caused by data object variables on the process. Any forms with nested bindings will result in additional SubForms during migration.
- Forms that are not related to business processes (forms that do not end with a **taskform** suffix) must have only one **DataHolder** of type **DataObject** or **ClassName**.
- The new forms designer does not support different bindings for input and output of processes. The same binding must be assigned to the input and output of processes.
- The new forms designer does not support dynamic functions, such as field calculations.

Business processes

Business processes migrated to Red Hat Process Automation Manager 7.0 by the Business Central migration tool are managed in the same process designer as Red Hat JBoss BPM Suite 6.4. Any new business processes created in Red Hat Process Automation Manager 7.0 are managed by default in the new process designer. The new designer does not support all features of the previous process designer, but will be incrementally improved in subsequent releases to better support currently deprecated features. If you encounter features not supported in the new process designer as you create new processes, you can switch to the previous designer for more advanced capabilities until support is restored in future releases.
For more information about creating and managing processes in Red Hat Process Automation Manager 7.0, see *Getting started with business processes*.

Pages (previously Dashboards)

Dashboards in Red Hat JBoss BPM Suite 6.4 cannot be migrated to Red Hat Process Automation Manager 7.0. Any existing dashboards must be re-created as Pages in Red Hat Process Automation Manager 7.0. For information about creating and managing Pages, see *Building custom dashboard widgets*.

CHAPTER 3. MIGRATING BUSINESS CENTRAL AS DESIGN ENVIRONMENT ONLY

If you use Business Central to design business assets but do not use Business Central as an execution server with live tasks and process instances in Red Hat JBoss BPM Suite 6.4., then follow the steps in this section to migrate to Red Hat Process Automation Manager 7.0.

If you use Business Central as both a design environment and an execution server with live tasks and process instances, follow the migration steps in Chapter 4, *Migrating Business Central as design environment and execution server*. A separate migration path is required to transfer execution server functionality to the Process Server for Red Hat Process Automation Manager 7.0, in addition to Business Central data.

Prerequisites

- Red Hat Process Automation Manager 7.0 is installed on a separate instance from Red Hat JBoss BPM Suite 6.4 (on Red Hat JBoss EAP or other instance). For installation options, see *Planning a Red Hat Process Automation Manager installation* .

- All Business Central data for Red Hat JBoss BPM Suite 6.4, including a database used with it (if applicable), has been fully backed up. This is a precaution in case of problems during migration.

3.1. USE THE BUSINESS CENTRAL MIGRATION TOOL

Red Hat Process Automation Manager 7.0 contains a modified data structure and other feature changes that do not support a direct data migration from Red Hat JBoss BPM Suite 6.4. Therefore, Red Hat provides a migration tool that enables you to migrate project data and configurations from Red Hat JBoss BPM Suite 6.4 to Red Hat Process Automation Manager 7.0. You can run the migration tool on the **.niogit** directory of your Business Central distribution or on an external Git repository. The migration tool prepares your Business Central data for the new data structure and features supported in Red Hat Process Automation Manager 7.0.

3.1.1. Using the migration tool on a .niogit directory

The **.niogit** directory of your Business Central distribution contains all Business Central data. You can use the Business Central migration tool on your **.niogit** directory to prepare project data and system configurations for migration to Red Hat Process Automation Manager 7.0.

Prerequisites

- Business Central is not running for either Red Hat JBoss BPM Suite 6.4 or Red Hat Process Automation Manager 7.0.

- Java 8 is installed and is available on the path where the project migration tool will be installed.

Procedure

1. Locate the **bin/.niogit** directory in your current Business Central distribution for Red Hat JBoss BPM Suite 6.4 and locate the same directory in your new Red Hat Process Automation Manager 7.0 installation.
   Example:

   ```
   $ ~/$JBOSS_HOME/bin/.niogit
   ```

```
$ ~/$RHPAM_HOME/bin/.niogit
```

2. In your command terminal, copy the **.niogit** directory from your current Business Central distribution for Red Hat JBoss BPM Suite 6.4 to the **bin** folder of the new Red Hat Process Automation Manager 7.0 installation directory:

   ```
   cp -r /$JBOSS_HOME/bin/.niogit  /$RHPAM_HOME/bin/
   ```

   The copied **.niogit** folder is the directory that the migration tool will prepare for Red Hat Process Automation Manager 7.0. The original **.niogit** folder remains in your Red Hat JBoss BPM Suite 6.4 installation and will not be modified by the migration tool in this procedure.

3. In the Red Hat Process Automation Manager 7.0 **bin** directory, delete the **.index** directory. You must delete this folder because Red Hat Process Automation Manager 7.0 comes with certain updates that are not compatible with Red Hat JBoss BPM Suite 6.4. The **bin** directory will be re-indexed after the migration when you start Red Hat Process Automation Manager 7.0.

4. Note the path to the copied **.niogit** directory for Red Hat Process Automation Manager 7.0. The path will be required when you run the migration tool.

5. If you have specific **dependencies**, **repositories**, or **pluginRepositories** that you want to add, remove, or update as part of the project migration, create a **pom-migration.json** file containing these elements. When you run the migration tool, the tool will prompt you if you want to include this file in the migration.
   The **pom-migration.json** file requires the following sections:

   - **"dependencies":[]**
   - **"repositories-add":[]**
   - **"repositories-remove":[]**
   - **"repositories-update-urls":[]**
   - **"pluginRepositories-add":[]**
   - **"pluginRepositories-remove":[]**
   - **"pluginRepositories-update-urls":[]**

   If any of these elements do not apply to your project, include them as empty sections to prevent parsing errors.

   Example **pom-migration.json** file:

   ```
   {
     "dependencies":[

       {"groupId":"junit", "artifactId":"junit", "version":"4.12", "scope":"test"},
       {"groupId":"com.thoughtworks.xstream", "artifactId":"xstream", "version":"4.12",
   "scope":"test"}

     ],

     "repositories-add":[
       {
   ```

```
      "id":"jboss-public-repository-group",
      "name":"JBoss Public Repository Group",
      "url":"http://repository.jboss.org/nexus/content/groups/public/",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":true,
      "snapshotUpdatePolicy":"never"
     },
     {
      "id":"kie-internal-staging-repository-group",
      "name":"KIE Internal Staging Repositories",
      "url":"https://origin-repository.jboss.org/nexus/content/groups/kie-internal-group",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":false,
      "snapshotUpdatePolicy":"never"
     }
    ],

    "repositories-remove":[
     {
      "id":"productization-repository",
   "url":"http://download.lab.bos.redhat.com/brewroot/repos/jb-ip-6.1-build/latest/maven/"
     }
    ],

    "repositories-update-urls":[
     {
      "id":"guvnor-m2-repo", "url":"http://localhost:8080/business-central/maven3/"
     }
    ],

    "pluginRepositories-add":[

     {
      "id":"jboss-public-repository-group",
      "name":"JBoss Public Repository Group",
      "url":"http://repository.jboss.org/nexus/content/groups/public/",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":true,
      "snapshotUpdatePolicy":"never"
     },
     {
      "id":"kie-internal-staging-repository-group",
      "name":"KIE Internal Staging Repositories",
      "url":"https://origin-repository.jboss.org/nexus/content/groups/kie-internal-group",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":false,
      "snapshotUpdatePolicy":"never"
     }

    ],
  "pluginRepositories-remove":[],
  "pluginRepositories-update-urls":[]
}
```

Example **pom-migration.json** file with **dependencies** only:

```
{
  "dependencies":[

    {"groupId":"junit", "artifactId":"junit", "version":"4.12", "scope":"test"},
    {"groupId":"com.thoughtworks.xstream", "artifactId":"xstream", "version":"4.12",
"scope":"test"}

  ],
  "repositories-add":[],
  "repositories-remove":[],
  "repositories-update-urls":[],
  "pluginRepositories-add":[],
  "pluginRepositories-remove":[],
  "pluginRepositories-update-urls":[]
}
```

6. Navigate to the Software Downloads page in the Red Hat Customer Portal (login required), and select the product and version from the drop-down options:

   - **Product:** Process Automation Manager
   - **Version:** 7.0

7. Download **Red Hat Process Automation Manager 7.0.0 Add-Ons**and extract the downloaded **rhpam-7.0.0-add-ons.zip** file to a temporary directory.

8. In the extracted **rhpam-7.0.0-add-ons** folder, extract the **rhpam-7.0-migration-tool.zip** sub-folder. The migration tool is in the **bin** directory.

9. In your command terminal, navigate to the temporary directory where you extracted the **rhpam-7.0-migration-tool** folder and run the migration tool. The **$RHPAM_NIOGIT_DIR** portion is the path to the **.niogit** directory that you previously copied to the Red Hat Process Automation Manager 7.0 installation.
   On Linux or UNIX-based systems:

   ```
   $ cd $INSTALL_DIR/rhpam-7.0-migration-tool/bin
   $ ./migration-tool.sh -t $RHPAM_NIOGIT_DIR
   ```

   On Windows:

   ```
   $ cd $INSTALL_DIR\rhpam-7.0-migration-tool\bin
   $ migration-tool.bat -t $RHPAM_NIOGIT_DIR
   ```

   In the command prompt that appears, the following options are displayed:

   - **Project structure migration**: Migrates the Red Hat JBoss BPM Suite 6.4 project repository structure to the new project-oriented structure used in Red Hat Process Automation Manager 7.0.

- **System configuration directory structure migration**: Migrates the **system.git** repository structure used in Red Hat JBoss BPM Suite 6.4 to the new structure used in Red Hat Process Automation Manager 7.0. This migration option requires the project structure migration to be executed first.
- **Forms migration**: Migrates forms created in the Red Hat JBoss BPM Suite 6.4 forms designer to the new forms designer. This migration option requires the project structure migration and system configuration directory structure migration to be executed first.
- **POMs migration:** Updates **pom.xml** files with dependencies required for Red Hat Process Automation Manager 7.0. This migration option requires the project structure migration, system configuration directory structure migration, and forms migration to be executed first.
- **All:** Runs all migration options in sequence.
- **Exit:** Exits the migration tool.

10. Select the option to run **All** migrations in sequence.

    NOTE

    If you prefer to run one migration option at a time, select and run the first individual migration option. After the tool runs, re-run the Business Central migration tool and select the next individual migration option in the sequence.

11. Enter **yes** each time you are prompted to run a specific migration option.
    For the POMs migration option, if you want to include a path to an external **pom-migration.json** file that you created previously, enter **yes** when prompted and enter the path.
12. After the tool finishes running, enter the option to **Exit** the migration tool.
    The **.niogit** directory structure is now compatible with Business Central in Red Hat Process Automation Manager 7.0. Project directories are in separate repositories and all other related configurations have been migrated. You can navigate to the new **.niogit** directory to inspect the restructured contents.

3.1.2. Using the migration tool on an external Git repository

If you store Business Central project data in a Git repository outside of your **.niogit** directory, you can also use the Business Central migration tool on the external repository to prepare project data for Red Hat Process Automation Manager 7.0. Only files in project directories in the Git repository will be restructured by the migration tool.

Prerequisite

Java 8 is installed and is available on the path where the project migration tool will be installed.

Procedure

1. In a local directory, create a clone of the Git repository containing the projects to be migrated (if the repository has not been cloned already).

```
$ cd $REPO_DIR/
$ git clone $GIT_REPO_URL
```

2. Note the path to the cloned Git repository. The path will be required when you run the migration tool.

3. Create an output directory where a copy of the newly migrated repository or repositories will be placed after the migration tool runs. You can also use an existing directory as an output location. Note the path to this output directory. The migration tool operates on a copy of the cloned repository and will prompt you for the new output directory path when you run the tool.

4. If you have specific **dependencies**, **repositories**, or **pluginRepositories** that you want to add, remove, or update as part of the project migration, create a **pom-migration.json** file containing these elements. When you run the migration tool, the tool will prompt you if you want to include this file in the migration.
   The **pom-migration.json** file requires the following sections:

   - **"dependencies":[]**
   - **"repositories-add":[]**
   - **"repositories-remove":[]**
   - **"repositories-update-urls":[]**
   - **"pluginRepositories-add":[]**
   - **"pluginRepositories-remove":[]**
   - **"pluginRepositories-update-urls":[]**

   If any of these elements do not apply to your project, include them as empty sections to prevent parsing errors.

   Example **pom-migration.json** file:

```
{
  "dependencies":[

    {"groupId":"junit", "artifactId":"junit", "version":"4.12", "scope":"test"},
    {"groupId":"com.thoughtworks.xstream", "artifactId":"xstream", "version":"4.12",
"scope":"test"}

  ],

  "repositories-add":[
    {
      "id":"jboss-public-repository-group",
      "name":"JBoss Public Repository Group",
      "url":"http://repository.jboss.org/nexus/content/groups/public/",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":true,
      "snapshotUpdatePolicy":"never"
    },
    {
      "id":"kie-internal-staging-repository-group",
      "name":"KIE Internal Staging Repositories",
      "url":"https://origin-repository.jboss.org/nexus/content/groups/kie-internal-group",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":false,
      "snapshotUpdatePolicy":"never"
    }
  ],

  "repositories-remove":[
    {
      "id":"productization-repository",
"url":"http://download.lab.bos.redhat.com/brewroot/repos/jb-ip-6.1-build/latest/maven/"
    }
  ],

  "repositories-update-urls":[
    {
      "id":"guvnor-m2-repo", "url":"http://localhost:8080/business-central/maven3/"
    }
  ],

  "pluginRepositories-add":[

    {
      "id":"jboss-public-repository-group",
      "name":"JBoss Public Repository Group",
      "url":"http://repository.jboss.org/nexus/content/groups/public/",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":true,
      "snapshotUpdatePolicy":"never"
    },
    {
      "id":"kie-internal-staging-repository-group",
      "name":"KIE Internal Staging Repositories",
      "url":"https://origin-repository.jboss.org/nexus/content/groups/kie-internal-group",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":false,
      "snapshotUpdatePolicy":"never"
    }

  ],
  "pluginRepositories-remove":[],
  "pluginRepositories-update-urls":[]
}
```

Example **pom-migration.json** file with **dependencies** only:

```
{
  "dependencies":[

    {"groupId":"junit", "artifactId":"junit", "version":"4.12", "scope":"test"},
    {"groupId":"com.thoughtworks.xstream", "artifactId":"xstream", "version":"4.12",
"scope":"test"}

  ],
  "repositories-add":[],
  "repositories-remove":[],
  "repositories-update-urls":[],
  "pluginRepositories-add":[],
  "pluginRepositories-remove":[],
  "pluginRepositories-update-urls":[]
}
```

5. Navigate to the Software Downloads page in the Red Hat Customer Portal (login required), and select the product and version from the drop-down options:

   - **Product:** Process Automation Manager

   - **Version:** 7.0

6. Download **Red Hat Process Automation Manager 7.0.0 Add-Ons**and extract the downloaded **rhpam-7.0.0-add-ons.zip** file to a temporary directory.

7. In the extracted **rhpam-7.0.0-add-ons** folder, extract the **rhpam-7.0-migration-tool.zip** sub-folder. The migration tool is in the **bin** directory.

8. In your command terminal, navigate to the temporary directory where you extracted the **rhpam-7.0-migration-tool** folder and run the migration tool. The **$GIT_REPO_PATH** portion is the path to the cloned Git repository.
   On Linux or UNIX-based systems:

   ```
   $ cd $INSTALL_DIR/rhpam-7.0-migration-tool/bin
   $ ./migration-tool.sh -t $GIT_REPO_PATH
   ```

   On Windows:

   ```
   $ cd $INSTALL_DIR\rhpam-7.0-migration-tool\bin
   $ migration-tool.bat -t $GIT_REPO_PATH
   ```

9. In the command prompt that appears, enter the path to the output directory where the migrated copy of the repository will be placed. The migration tool operates on a copy of the cloned repository and will place the new repository or repositories in the output location that you specify.
   After you enter the output location, the migration tool prepares the repository copy and restructures all project directories to be compatible with Red Hat Process Automation Manager 7.0.

   In the command prompt, the following additional migration options are displayed:

   - **Forms migration**: Migrates forms created in the Red Hat JBoss BPM Suite 6.4 forms designer to the new forms designer.

   - **POMs migration:**Updates **pom.xml** files with dependencies required for Red Hat Process Automation Manager 7.0. This migration option requires the forms migration to be executed first.

   - **All:** Runs all remaining migration options in sequence.

   - **Exit:** Exits the migration tool.

> NOTE
>
> The **Project structure migration** option is not displayed because that option was run automatically when you entered the output location to initiate the migration tool.

10. Select the option to run **ALL** migrations in sequence.

    > NOTE
    >
    > If you prefer to run one migration option at a time, select and run the first individual migration option. After the tool runs, re-run the Business Central migration tool and select the next individual migration option in the sequence.

11. Enter **yes** each time you are prompted to run a specific migration option.
    For the POMs migration option, if you want to include a path to an external **pom-migration.json** file that you created previously, enter **yes** when prompted and enter the path.

12. After the tool finishes running, enter the option to **Exit** the migration tool.
    The project directories in the specified output location are now separated into individual repositories compatible with Business Central in Red Hat Process Automation Manager 7.0. The new project repositories are bare repositories with no working directory, and therefore do not show any content files. You can clone each repository to create non-bare repositories and inspect directory contents.

13. Log in to Business Central for Red Hat Process Automation Manager 7.0.

14. For each project, create or select the space for the project and click **Import Project**.

15. Enter the **Repository URL** for the newly structured project repository. This URL may be the local file path to the output location if you are importing directly from the workstation where you ran the migration tool, or a GitHub URL or other file hosting location where you have uploaded the repository.
    Example: Local file location

    ```
    file:///$OUTPUT_DIR/loan-application.git
    ```

    Example: GitHub repository location

    ```
    https://github.com/$REPO/loan-application.git
    ```

    > NOTE
    >
    > If you use Git **hooks** with your project data, you may need to update your **hooks** scripts to accommodate the migration.

3.2. UPGRADE YOUR DATABASE

Red Hat Process Automation Manager 7.0 contains updates to the database schema for optimal performance with the new Business Central and Process Server. You must update the Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 using one of the upgrade scripts provided with this release to accommodate the change.

Prerequisites

- The **Red Hat Process Automation Manager 7.0.0 Add-Ons**distribution has been downloaded from the Red Hat Customer Portal and extracted to a temporary directory.
- The database has been backed up, as a precaution.

Procedure

1. Navigate to the extracted **rhpam-7.0.0-add-ons** folder in your temporary directory and extract the **rhpam-7.0-migration-tool.zip** sub-folder (if not extracted already).
2. Navigate to the extracted **rhpam-7.0-migration-tool** folder and open the **upgrade-scripts** folder. This folder contains database upgrade scripts that are required for applying changes to the database model made in Red Hat Process Automation Manager 7.0.
3. Log on to the Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 and run the relevant Red Hat Process Automation Manager 7.0 upgrade script, according to the supported database type. This prepares the database for Red Hat Process Automation Manager 7.0.
4. If you use Quartz as your scheduling service, additionally run the **quartz-upgrade.sql** script provided in the relevant database folder within the **upgrade-scripts** folder. This script updates your Quartz version to 2.2.3 in compliance with Red Hat Process Automation Manager 7.0 requirements.

3.3. UPDATE API REFERENCES TO BUSINESS CENTRAL KNOWLEDGE STORE

The Knowledge Store REST API has been deprecated in Red Hat Process Automation Manager 7.0, but certain endpoints are still supported so that you can continue to manage Business Central resources. If any of your applications interact with Business Central spaces (previously organizational units), repositories, and projects through Knowledge Store REST API calls (**/business-central/rest/**), you must update these API references according to the new endpoints supported in Red Hat Process Automation Manager 7.0.

Procedure

Wherever applicable in your application code, replace any legacy REST API calls to the Business Central Knowledge Store in the format **http://<SERVER>:<PORT>/business-central/rest/<ENDPOINT>** with the new corresponding Knowledge Store REST API calls listed in Section 3.3.1, “Supported calls to Knowledge Store REST API”.

As a result of the restructured content in Red Hat Process Automation Manager 7.0, the endpoints may differ between the legacy Business Central REST API calls and the corresponding REST API calls in Red Hat Process Automation Manager 7.0. Be sure to use the new endpoints to replace your legacy Business Central REST API calls. If you cannot find a corresponding REST API call in the list to replace a legacy call, then that call is no longer supported in Red Hat Process Automation Manager 7.0.

The following example is an update to a Knowledge Store REST API call for spaces.

Legacy Knowledge Store REST API call:

```
http://localhost:8080/business-central/rest/organizational-units
```

New Knowledge Store REST API call:

```
http://localhost:8080/business-central/rest/spaces
```

3.3.1. Supported calls to Knowledge Store REST API

The following table lists the Knowledge Store REST API calls supported in Red Hat Process Automation Manager 7.0. The base URL for each listed endpoint is **http://<SERVER>:<PORT>/business-central/rest/**.

Example call URL for spaces:

```
http://localhost:8080/business-central/rest/spaces
```

Table 3.1. Knowledge Store REST API operations

Category	Endpoint	HTTP method	Description
Job calls	**/jobs/*JOB_ID***	GET	Returns a job status
	/jobs/*JOB_ID*	DELETE	Removes a job
Project calls	**/projects**	GET	Returns a list of all projects
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*	GET	Returns information about a specific project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*	POST	Creates a project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*	DELETE	Removes a specified project
	/spaces/*SPACE_NAME*/projects	GET	Returns a list of projects in a given space
	/spaces/*SPACE_NAME*/git/clone	POST	Clones a project into the space from a given Git address
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*	DELETE	Removes a specific project from a given space
Space calls	**/spaces**	GET	Returns a list of all spaces
	/spaces/*SPACE_NAME*	GET	Returns information about a specific space
	/spaces	POST	Creates a space
	/spaces/*SPACE_NAME*	DELETE	Removes a specified space

Category	Endpoint	HTTP method	Description
Maven calls	**/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*/maven/compile**	POST	Compiles a project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*/maven/test	POST	Tests a project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*/maven/install	POST	Installs a project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*/maven/deploy	POST	Deploys a project

3.4. VERIFY THE MIGRATION IN BUSINESS CENTRAL

After you use the Business Central migration tool to migrate project data and you upgrade your Red Hat JBoss BPM Suite 6.4 database, verify that the migration was successful in the new Business Central for Red Hat Process Automation Manager 7.0.

Prerequisites

- Project data from Red Hat JBoss BPM Suite 6.4 has been migrated using the Business Central migration tool.
- The database configured for Red Hat JBoss BPM Suite has been upgraded.

Procedure

1. Start Red Hat Process Automation Manager 7.0 depending on your installation.
   Example: Installation with Red Hat JBoss EAP

   ```
   $ ~/$EAP_HOME/bin/standalone.sh -c standalone-full.xml
   ```

2. Log in to Business Central with your credentials and navigate to **Menu → Design → Projects**.
3. Verify that all projects have been migrated and select each project to verify the migrated project assets.
4. For each project, click **Build** and then **Deploy** in the upper-right corner of the project window to validate the migrated project data and the configured Process Server.

3.4.1. Troubleshooting Business Central migration problems

If you encounter problems with your Business Central migration to Red Hat Process Automation Manager 7.0, review the following troubleshooting suggestions:

- If any project data is missing from Business Central, ensure that the **.niogit** directory for Red Hat Process Automation Manager 7.0 contains the restructured data and is in the correct installation location for Red Hat Process Automation Manager 7.0 (for example, **~/$EAP_HOME/bin/**).

- If projects fail to build, open the project **pom.xml** file and remove the **http://<SERVER>:<PORT>/business-central/maven2/** repository, if present. The use of this repository can prevent projects from building properly in Red Hat Process Automation Manager 7.0. If this repository contains project data that you want to preserve, create a new repository for the data and add the repository as a dependency in the **pom.xml** file.
  Example repository to be removed from **pom.xml** file:

  ```
  <repository>
    <id>guvnor-m2-repo</id>
    <name>Guvnor M2 Repo</name>
    <url>http://localhost:8080/business-central/maven2/</url>
  </repository>
  ```

- If projects fail to deploy to Process Server, review your Process Server installation and configuration. For more information, see *Planning a Red Hat Process Automation Manager installation*.

- If you encounter other migration problems, review Chapter 6, *Other migration considerations*.

- If you cannot resolve migration problems, complete the migration process again on a new Red Hat Process Automation Manager 7.0 installation.

CHAPTER 4. MIGRATING BUSINESS CENTRAL AS DESIGN ENVIRONMENT AND EXECUTION SERVER

In Red Hat Process Automation Manager 7.0, Business Central no longer has execution server capabilities nor any API endpoints related to the execution server. Instead, the Process Server performs all execution and runtime functions and Business Central is used for application development and management only. Therefore, if you use Business Central to design business assets and to manage live tasks and process instances through the built-in execution server, then follow the steps in this section to migrate to Red Hat Process Automation Manager 7.0.

If you use Business Central as a design environment only, with no execution server functionality, follow the migration steps in Chapter 3, *Migrating Business Central as design environment only* . A separate migration path is required when Business Central is not used as an execution server.

Prerequisites

- Red Hat Process Automation Manager 7.0 is installed on a separate instance from Red Hat JBoss BPM Suite 6.4 (on Red Hat JBoss EAP or other instance). For installation options, see *Planning a Red Hat Process Automation Manager installation* .
- All Business Central data for Red Hat JBoss BPM Suite 6.4, including a database used with it (if applicable), has been fully backed up. This is a precaution in case of problems during migration.

4.1. PREPARE BUSINESS CENTRAL FOR MIGRATION

Before you can migrate project data from Red Hat JBoss BPM Suite 6.4, you must add to the new Business Central a server configuration and deployment units (KIE containers) corresponding to Red Hat JBoss BPM Suite 6.4 process deployments.

Procedure

1. Start Red Hat JBoss BPM Suite according to your installation (for example, **$JBOSS_HOME/bin/standalone.sh**) and log in to Business Central.
2. Navigate to **Deploy → Process Deployments** and note the **Deployment** identifier and other details for your deployments. This information will be required when you prepare the new Business Central in Red Hat Process Automation Manager 7.0 for migration.
3. Close Business Central and stop Red Hat JBoss BPM Suite 6.4 according to your installation.
4. Start Red Hat Process Automation Manager 7.0 according to your installation (for example, **$RHPAM_HOME/bin/standalone.sh -c standalone-full.xml**) and log in to the new Business Central. This distribution of Business Central does not yet contain your project data or live instances, but needs to be prepared for the execution server migration.
5. Navigate to **Menu → Deploy → Execution Servers → New Server Configuration**and define a new server configuration (for example, **myserver**). This server configuration will be used by the new Process Server to manage the deployments currently on the Business Central execution server for Red Hat JBoss BPM Suite 6.4.
6. Select the new server configuration, click **Add Deployment Unit**, and add a deployment unit (KIE container) for each of the deployments on the execution server. Name each deployment unit with the exact deployment identifier from Red Hat JBoss BPM Suite 6.4 (for example, **org.jbpm:HR:1.5**).

> IMPORTANT
>
> Each deployment unit name in the server configuration must match exactly the deployment identifier for the deployment on the execution server. If they do not match, the deployed instances will not be migrated properly to the new Process Server.

7. Close Business Central and stop both Red Hat JBoss BPM Suite 6.4 (if not stopped already) and Red Hat Process Automation Manager 7.0 server instances based on how you started them.
   Example:

   ```
   $ ~/$EAP_HOME/bin/jboss-cli.sh --connect command=:shutdown
   ```

4.2. USE THE BUSINESS CENTRAL MIGRATION TOOL

Red Hat Process Automation Manager 7.0 contains a modified data structure and other feature changes that do not support a direct data migration from Red Hat JBoss BPM Suite 6.4. Therefore, Red Hat provides a migration tool that enables you to migrate project data and configurations from Red Hat JBoss BPM Suite 6.4 to Red Hat Process Automation Manager 7.0. You can run the migration tool on the **.niogit** directory of your Business Central distribution or on an external Git repository. The migration tool prepares your Business Central data for the new data structure and features supported in Red Hat Process Automation Manager 7.0.

4.2.1. Using the migration tool on a .niogit directory

The **.niogit** directory of your Business Central distribution contains all Business Central data. You can use the Business Central migration tool on your **.niogit** directory to prepare project data and system configurations for migration to Red Hat Process Automation Manager 7.0.

Prerequisites

- Business Central is not running for either Red Hat JBoss BPM Suite 6.4 or Red Hat Process Automation Manager 7.0.
- Java 8 is installed and is available on the path where the project migration tool will be installed.

Procedure

1. Locate the **bin/.niogit** directory in your current Business Central distribution for Red Hat JBoss BPM Suite 6.4 and locate the same directory in your new Red Hat Process Automation Manager 7.0 installation.
   Example:

   ```
   $ ~/$JBOSS_HOME/bin/.niogit
   ```

   ```
   $ ~/$RHPAM_HOME/bin/.niogit
   ```

2. In your command terminal, copy the **.niogit** directory from your current Business Central distribution for Red Hat JBoss BPM Suite 6.4 to the **bin** folder of the new Red Hat Process Automation Manager 7.0 installation directory:

   ```
   cp -r /$JBOSS_HOME/bin/.niogit  /$RHPAM_HOME/bin/
   ```

The copied **.niogit** folder is the directory that the migration tool will prepare for Red Hat Process Automation Manager 7.0. The original **.niogit** folder remains in your Red Hat JBoss BPM Suite 6.4 installation and will not be modified by the migration tool in this procedure.

3. In the Red Hat Process Automation Manager 7.0 **bin** directory, delete the **.index** directory. You must delete this folder because Red Hat Process Automation Manager 7.0 comes with certain updates that are not compatible with Red Hat JBoss BPM Suite 6.4. The **bin** directory will be re-indexed after the migration when you start Red Hat Process Automation Manager 7.0.

4. Note the path to the copied **.niogit** directory for Red Hat Process Automation Manager 7.0. The path will be required when you run the migration tool.

5. If you have specific **dependencies**, **repositories**, or **pluginRepositories** that you want to add, remove, or update as part of the project migration, create a **pom-migration.json** file containing these elements. When you run the migration tool, the tool will prompt you if you want to include this file in the migration.
   The **pom-migration.json** file requires the following sections:

   - **"dependencies":[]**
   - **"repositories-add":[]**
   - **"repositories-remove":[]**
   - **"repositories-update-urls":[]**
   - **"pluginRepositories-add":[]**
   - **"pluginRepositories-remove":[]**
   - **"pluginRepositories-update-urls":[]**

   If any of these elements do not apply to your project, include them as empty sections to prevent parsing errors.

   Example **pom-migration.json** file:

```
{
  "dependencies":[

    {"groupId":"junit", "artifactId":"junit", "version":"4.12", "scope":"test"},
    {"groupId":"com.thoughtworks.xstream", "artifactId":"xstream", "version":"4.12",
"scope":"test"}

  ],

  "repositories-add":[
    {
      "id":"jboss-public-repository-group",
      "name":"JBoss Public Repository Group",
      "url":"http://repository.jboss.org/nexus/content/groups/public/",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":true,
      "snapshotUpdatePolicy":"never"
    },
    {
      "id":"kie-internal-staging-repository-group",
      "name":"KIE Internal Staging Repositories",
      "url":"https://origin-repository.jboss.org/nexus/content/groups/kie-internal-group",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":false,
      "snapshotUpdatePolicy":"never"
    }
  ],

  "repositories-remove":[
    {
      "id":"productization-repository",
"url":"http://download.lab.bos.redhat.com/brewroot/repos/jb-ip-6.1-build/latest/maven/"
    }
  ],

  "repositories-update-urls":[
    {
      "id":"guvnor-m2-repo", "url":"http://localhost:8080/business-central/maven3/"
    }
  ],

  "pluginRepositories-add":[

    {
      "id":"jboss-public-repository-group",
      "name":"JBoss Public Repository Group",
      "url":"http://repository.jboss.org/nexus/content/groups/public/",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":true,
      "snapshotUpdatePolicy":"never"
    },
    {
      "id":"kie-internal-staging-repository-group",
      "name":"KIE Internal Staging Repositories",
      "url":"https://origin-repository.jboss.org/nexus/content/groups/kie-internal-group",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":false,
      "snapshotUpdatePolicy":"never"
    }

  ],
  "pluginRepositories-remove":[],
  "pluginRepositories-update-urls":[]
}
```

Example **pom-migration.json** file with **dependencies** only:

```
{
  "dependencies":[

    {"groupId":"junit", "artifactId":"junit", "version":"4.12", "scope":"test"},
    {"groupId":"com.thoughtworks.xstream", "artifactId":"xstream", "version":"4.12",
"scope":"test"}

  ],
  "repositories-add":[],
  "repositories-remove":[],
  "repositories-update-urls":[],
  "pluginRepositories-add":[],
  "pluginRepositories-remove":[],
  "pluginRepositories-update-urls":[]
}
```

6. Navigate to the Software Downloads page in the Red Hat Customer Portal (login required), and select the product and version from the drop-down options:

    - **Product:** Process Automation Manager

    - **Version:** 7.0

7. Download **Red Hat Process Automation Manager 7.0.0 Add-Ons**and extract the downloaded **rhpam-7.0.0-add-ons.zip** file to a temporary directory.

8. In the extracted **rhpam-7.0.0-add-ons** folder, extract the **rhpam-7.0-migration-tool.zip** sub-folder. The migration tool is in the **bin** directory.

9. In your command terminal, navigate to the temporary directory where you extracted the **rhpam-7.0-migration-tool** folder and run the migration tool. The **$RHPAM_NIOGIT_DIR** portion is the path to the **.niogit** directory that you previously copied to the Red Hat Process Automation Manager 7.0 installation.
   On Linux or UNIX-based systems:

   ```
   $ cd $INSTALL_DIR/rhpam-7.0-migration-tool/bin
   $ ./migration-tool.sh -t $RHPAM_NIOGIT_DIR
   ```

   On Windows:

   ```
   $ cd $INSTALL_DIR\rhpam-7.0-migration-tool\bin
   $ migration-tool.bat -t $RHPAM_NIOGIT_DIR
   ```

   In the command prompt that appears, the following options are displayed:

    - **Project structure migration**: Migrates the Red Hat JBoss BPM Suite 6.4 project repository structure to the new project-oriented structure used in Red Hat Process Automation Manager 7.0.

    - **System configuration directory structure migration**: Migrates the **system.git** repository structure used in Red Hat JBoss BPM Suite 6.4 to the new structure used in Red Hat Process Automation Manager 7.0. This migration option requires the project structure migration to be executed first.

    - **Forms migration**: Migrates forms created in the Red Hat JBoss BPM Suite 6.4 forms designer to the new forms designer. This migration option requires the project structure migration and system configuration directory structure migration to be executed first.

    - **POMs migration:** Updates **pom.xml** files with dependencies required for Red Hat Process Automation Manager 7.0. This migration option requires the project structure migration, system configuration directory structure migration, and forms migration to be executed first.

- **All:** Runs all migration options in sequence.
- **Exit:** Exits the migration tool.

10. Select the option to run **All** migrations in sequence.

    > NOTE
    >
    > If you prefer to run one migration option at a time, select and run the first individual migration option. After the tool runs, re-run the Business Central migration tool and select the next individual migration option in the sequence.

11. Enter **yes** each time you are prompted to run a specific migration option.
    For the POMs migration option, if you want to include a path to an external **pom-migration.json** file that you created previously, enter **yes** when prompted and enter the path.

12. After the tool finishes running, enter the option to **Exit** the migration tool.
    The **.niogit** directory structure is now compatible with Business Central in Red Hat Process Automation Manager 7.0. Project directories are in separate repositories and all other related configurations have been migrated. You can navigate to the new **.niogit** directory to inspect the restructured contents.

4.2.2. Using the migration tool on an external Git repository

If you store Business Central project data in a Git repository outside of your **.niogit** directory, you can also use the Business Central migration tool on the external repository to prepare project data for Red Hat Process Automation Manager 7.0. Only files in project directories in the Git repository will be restructured by the migration tool.

Prerequisite

Java 8 is installed and is available on the path where the project migration tool will be installed.

Procedure

1. In a local directory, create a clone of the Git repository containing the projects to be migrated (if the repository has not been cloned already).

    ```
    $ cd $REPO_DIR/
    $ git clone $GIT_REPO_URL
    ```

2. Note the path to the cloned Git repository. The path will be required when you run the migration tool.

3. Create an output directory where a copy of the newly migrated repository or repositories will be placed after the migration tool runs. You can also use an existing directory as an output location. Note the path to this output directory. The migration tool operates on a copy of the cloned repository and will prompt you for the new output directory path when you run the tool.

4. If you have specific **dependencies**, **repositories**, or **pluginRepositories** that you want to add, remove, or update as part of the project migration, create a **pom-migration.json** file containing these elements. When you run the migration tool, the tool will prompt you if you want to include this file in the migration.

The **pom-migration.json** file requires the following sections:

- **"dependencies":[]**
- **"repositories-add":[]**
- **"repositories-remove":[]**
- **"repositories-update-urls":[]**
- **"pluginRepositories-add":[]**
- **"pluginRepositories-remove":[]**
- **"pluginRepositories-update-urls":[]**

If any of these elements do not apply to your project, include them as empty sections to prevent parsing errors.

Example **pom-migration.json** file:

```
{
  "dependencies":[

    {"groupId":"junit", "artifactId":"junit", "version":"4.12", "scope":"test"},
    {"groupId":"com.thoughtworks.xstream", "artifactId":"xstream", "version":"4.12",
"scope":"test"}

  ],

  "repositories-add":[
    {
      "id":"jboss-public-repository-group",
      "name":"JBoss Public Repository Group",
      "url":"http://repository.jboss.org/nexus/content/groups/public/",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":true,
      "snapshotUpdatePolicy":"never"
    },
    {
      "id":"kie-internal-staging-repository-group",
      "name":"KIE Internal Staging Repositories",
      "url":"https://origin-repository.jboss.org/nexus/content/groups/kie-internal-group",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":false,
      "snapshotUpdatePolicy":"never"
    }
  ],

  "repositories-remove":[
    {
      "id":"productization-repository",
"url":"http://download.lab.bos.redhat.com/brewroot/repos/jb-ip-6.1-build/latest/maven/"
    }
  ],

  "repositories-update-urls":[
    {
      "id":"guvnor-m2-repo", "url":"http://localhost:8080/business-central/maven3/"
    }
  ],

  "pluginRepositories-add":[

    {
      "id":"jboss-public-repository-group",
      "name":"JBoss Public Repository Group",
      "url":"http://repository.jboss.org/nexus/content/groups/public/",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":true,
      "snapshotUpdatePolicy":"never"
    },
    {
      "id":"kie-internal-staging-repository-group",
      "name":"KIE Internal Staging Repositories",
      "url":"https://origin-repository.jboss.org/nexus/content/groups/kie-internal-group",
      "releasesEnabled":true,
      "releasesUpdatePolicy":"never",
      "snapshotEnabled":false,
      "snapshotUpdatePolicy":"never"
    }

  ],
  "pluginRepositories-remove":[],
  "pluginRepositories-update-urls":[]
}
```

Example **pom-migration.json** file with **dependencies** only:

```
{
  "dependencies":[

    {"groupId":"junit", "artifactId":"junit", "version":"4.12", "scope":"test"},
    {"groupId":"com.thoughtworks.xstream", "artifactId":"xstream", "version":"4.12",
"scope":"test"}

  ],
  "repositories-add":[],
  "repositories-remove":[],
  "repositories-update-urls":[],
  "pluginRepositories-add":[],
  "pluginRepositories-remove":[],
  "pluginRepositories-update-urls":[]
}
```

5. Navigate to the Software Downloads page in the Red Hat Customer Portal (login required), and select the product and version from the drop-down options:

   - **Product:** Process Automation Manager

- **Version:** 7.0

6. Download **Red Hat Process Automation Manager 7.0.0 Add-Ons**and extract the downloaded **rhpam-7.0.0-add-ons.zip** file to a temporary directory.

7. In the extracted **rhpam-7.0.0-add-ons** folder, extract the **rhpam-7.0-migration-tool.zip** sub-folder. The migration tool is in the **bin** directory.

8. In your command terminal, navigate to the temporary directory where you extracted the **rhpam-7.0-migration-tool** folder and run the migration tool. The **$GIT_REPO_PATH** portion is the path to the cloned Git repository.
   On Linux or UNIX-based systems:

   ```
   $ cd $INSTALL_DIR/rhpam-7.0-migration-tool/bin
   $ ./migration-tool.sh -t $GIT_REPO_PATH
   ```

   On Windows:

   ```
   $ cd $INSTALL_DIR\rhpam-7.0-migration-tool\bin
   $ migration-tool.bat -t $GIT_REPO_PATH
   ```

9. In the command prompt that appears, enter the path to the output directory where the migrated copy of the repository will be placed. The migration tool operates on a copy of the cloned repository and will place the new repository or repositories in the output location that you specify.
   After you enter the output location, the migration tool prepares the repository copy and restructures all project directories to be compatible with Red Hat Process Automation Manager 7.0.

   In the command prompt, the following additional migration options are displayed:

   - **Forms migration**: Migrates forms created in the Red Hat JBoss BPM Suite 6.4 forms designer to the new forms designer.

   - **POMs migration:**Updates **pom.xml** files with dependencies required for Red Hat Process Automation Manager 7.0. This migration option requires the forms migration to be executed first.

   - **All:** Runs all remaining migration options in sequence.

   - **Exit:** Exits the migration tool.

   NOTE

   The **Project structure migration** option is not displayed because that option was run automatically when you entered the output location to initiate the migration tool.

10. Select the option to run **ALL** migrations in sequence.

NOTE

If you prefer to run one migration option at a time, select and run the first individual migration option. After the tool runs, re-run the Business Central migration tool and select the next individual migration option in the sequence.

11. Enter **yes** each time you are prompted to run a specific migration option.
    For the POMs migration option, if you want to include a path to an external **pom-migration.json** file that you created previously, enter **yes** when prompted and enter the path.

12. After the tool finishes running, enter the option to **Exit** the migration tool.
    The project directories in the specified output location are now separated into individual repositories compatible with Business Central in Red Hat Process Automation Manager 7.0. The new project repositories are bare repositories with no working directory, and therefore do not show any content files. You can clone each repository to create non-bare repositories and inspect directory contents.

13. Log in to Business Central for Red Hat Process Automation Manager 7.0.

14. For each project, create or select the space for the project and click **Import Project**.

15. Enter the **Repository URL** for the newly structured project repository. This URL may be the local file path to the output location if you are importing directly from the workstation where you ran the migration tool, or a GitHub URL or other file hosting location where you have uploaded the repository.
    Example: Local file location

    ```
    file:///$OUTPUT_DIR/loan-application.git
    ```

    Example: GitHub repository location

    ```
    https://github.com/$REPO/loan-application.git
    ```

NOTE

If you use Git **hooks** with your project data, you may need to update your **hooks** scripts to accommodate the migration.

4.3. UPGRADE YOUR DATABASE

Red Hat Process Automation Manager 7.0 contains updates to the database schema for optimal performance with the new Business Central and Process Server. You must update the Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 using one of the upgrade scripts provided with this release to accommodate the change.

Prerequisites

- The **Red Hat Process Automation Manager 7.0.0 Add-Ons**distribution has been downloaded from the Red Hat Customer Portal and extracted to a temporary directory.

- The database has been backed up, as a precaution.

Procedure

1. Navigate to the extracted **rhpam-7.0.0-add-ons** folder in your temporary directory and extract the **rhpam-7.0-migration-tool.zip** sub-folder (if not extracted already).

2. Navigate to the extracted **rhpam-7.0-migration-tool** folder and open the **upgrade-scripts** folder. This folder contains database upgrade scripts that are required for applying changes to the database model made in Red Hat Process Automation Manager 7.0.

3. Log on to the Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 and run the relevant Red Hat Process Automation Manager 7.0 upgrade script, according to the supported database type. This prepares the database for Red Hat Process Automation Manager 7.0.

4. If you use Quartz as your scheduling service, additionally run the **quartz-upgrade.sql** script provided in the relevant database folder within the **upgrade-scripts** folder. This script updates your Quartz version to 2.2.3 in compliance with Red Hat Process Automation Manager 7.0 requirements.

4.4. REPLACE API REFERENCES TO BUSINESS CENTRAL EXECUTION SERVER

In Red Hat Process Automation Manager 7.0, Business Central no longer has execution server capabilities nor any API endpoints related to the execution server. Instead, the Process Server performs all execution and runtime functions and Business Central is used only for designing and managing business assets. Therefore, if any of your applications interact with the Business Central execution server through the remote Java API (**kie-remote-client**) or through direct REST API calls (**/business-central/rest/**), you must update these API references to redirect to Process Server.

Prerequisite

The **Red Hat Process Automation Manager 7.0.0 Add-Ons** distribution has been downloaded from the Red Hat Customer Portal and extracted to a temporary directory.

Procedure

1. In your application code for remote Java clients, replace any instances of the following legacy package and module for Business Central with the new package and module for Process Server. This replaces references to the remote Java API with references to Process Server.
   Legacy Business Central package and module:

   ```
   org.kie.remote.client
   kie-remote-client
   ```

   New Process Server package and module:

   ```
   org.kie.server.client
   kie-server-client
   ```

2. Rewrite any Java client application code that currently interacts with the legacy Process Server remote client so that it interacts instead with the relevant Process Server client. When you download the **Red Hat Process Automation Manager 7.0.0 Add-Ons** distribution from the Red Hat Customer Portal, the client services are found in **~/rhpam-7.0.0-add-ons/rhpam-7.0-process-engine/kie-server-client-7.7.0.Final-redhat-7/org/kie/server/client**.
   For more information about Process Server clients, see Section 4.4.1, “Available Process Server clients”.

The following example illustrates code updates from the legacy **kie-remote-client** to the new **kie-server-client**.

Legacy code using **kie-remote-client**:

```
RuntimeEngine engine = RemoteRuntimeEngineFactory.newRestBuilder()
   .addUrl(new URL("http://localhost:8080/business-central"))
   .addUserName("john").addPassword("password")
   .addDeploymentId("org.jbpm:Evaluation:1.0")
    .build();
  KieSession ksession = engine.getKieSession();
  TaskService taskService = engine.getTaskService();

  // start a new process instance
  Map<String, Object> params = new HashMap<String, Object>();
  params.put("employee", "john");
  params.put("reason", "Yearly performance evaluation");
  ProcessInstance processInstance =  ksession.startProcess("evaluation", params);
  System.out.println("Start Evaluation process " + processInstance.getId());

  // complete Self Evaluation
  List<TaskSummary> tasks = taskService.getTasksAssignedAsPotentialOwner("john", "en-
UK");
  TaskSummary task = tasks.get(0);
  System.out.println("'john' completing task " + task.getName() + ": " + task.getDescription());
  taskService.start(task.getId(), "john");
  Map<String, Object> results = new HashMap<String, Object>();
  results.put("performance", "exceeding");
  taskService.complete(task.getId(), "john", results);
```

New code using **kie-server-client**:

```
KieServicesConfiguration config =  KieServicesFactory.newRestConfiguration(
   "http://localhost:8080/kie-server/services/rest/server", "john", "password");
  KieServicesClient client = KieServicesFactory.newKieServicesClient(config);
  ProcessServicesClient processServices =
client.getServicesClient(ProcessServicesClient.class);
  UserTaskServicesClient taskServices =
client.getServicesClient(UserTaskServicesClient.class);

  // start a new process instance
  Map<String, Object> params = new HashMap<String, Object>();
  params.put("employee", "john");
  params.put("reason", "Yearly performance evaluation");
  Long processInstanceId = processServices.startProcess("evaluation_1.0.0-SNAPSHOT",
"evaluation", params);
  System.out.println("Start Evaluation process " + processInstanceId);

  // complete Self Evaluation
  List<TaskSummary> tasks = taskServices.findTasksAssignedAsPotentialOwner("john", 0,
10);
  TaskSummary task = tasks.get(0);
  System.out.println("'john' completing task " + task.getName() + ": " + task.getDescription());
  taskServices.startTask("evaluation_1.0.0-SNAPSHOT", task.getId(), "john");
```

```
Map<String, Object> results = new HashMap<String, Object>();
results.put("performance", "10");
taskServices.completeTask("evaluation_1.0.0-SNAPSHOT", task.getId(), "john", results);
```

3. Wherever applicable in your application code, replace any legacy REST API calls to the Business Central execution server in the format **http://<SERVER>:<PORT>/business-central/rest/<ENDPOINT>** with the new corresponding REST API calls to Process Server in the format **http://<SERVER>:<PORT>/kie-server/services/rest/server/<ENDPOINT>**.
   The endpoints may differ between the legacy Business Central REST API calls and the corresponding Process Server REST API calls. To view the new endpoints for the corresponding Process Server API calls, start your Process Server instance and go to **http://<SERVER>:<PORT>/kie-server/docs** in a web browser (for example, http://localhost:8080/kie-server/docs). Use the endpoints in this page to replace your legacy Business Central REST API calls.

   ### BUSINESS CENTRAL CONTROLLER AND KNOWLEDGE STORE REST API

   As you update all API references to the Business Central execution server, note that the Business Central Controller and the Knowledge Store REST API also use the base URL **http://<SERVER>:<PORT>/business-central/rest/**. However, these endpoints are handled separately from execution server API calls:

   - **Business Central Controller:** Do not update any Business Central Controller endpoints, which are in the format **http://<SERVER>:<PORT>/business-central/rest/controller**. The Business Central Controller is not affected by the API changes in Red Hat Process Automation Manager 7.0.
   - **Knowledge Store REST API:** Do not update Knowledge Store REST API endpoints in the same way as execution server endpoints. Instead, update any Knowledge Store REST API endpoints (in the format **http://<SERVER>:<PORT>/business-central/rest/<ENDPOINT>**) according to the Knowledge Store endpoints supported in Red Hat Process Automation Manager 7.0. For more information about updating Knowledge Store endpoints, see Section 4.5, "Update API references to Business Central Knowledge Store" .

   The following example is an update to a REST API call for tasks.

   Legacy Business Central REST API call:

   ```
   http://localhost:8080/business-central/rest/tasks
   ```

   New Process Server REST API call (from **http://<SERVER>:<PORT>/kie-server/docs**):

   ```
   http://localhost:8080/kie-server/services/rest/server/queries/tasks/instances
   ```

   ### NOTE

   If Process Server is deployed in a separate Red Hat JBoss EAP or other instance or has been moved to a new location, the server and port locations may differ from Business Central.

4.4.1. Available Process Server clients

The following are some of the Java client services available in the **org.kie.server.client** package. You can use these services to configure your Java applications to interact with the relevant Process Server Java client APIs. When you download the **Red Hat Process Automation Manager 7.0.0 Add-Ons** distribution from the Red Hat Customer Portal , the client services are found in **~/rhpam-7.0.0-add-ons/rhpam-7.0-process-engine/kie-server-client-7.7.0.Final-redhat-7/org/kie/server/client**.

- **JobServicesClient**: Used to schedule, cancel, re-queue, and get job requests.
- **ProcessServicesClient**: Used to start, signal, and abort processes or work items.
- **QueryServicesClient**: Used to query processes, process nodes, and process variables.
- **RuleServicesClient**: Used to send commands to the server to perform rule-related operations (for example, insert objects into the working memory or execute rules).
- **SolverServicesClient**: Used to perform all Red Hat Business Optimizer operations, such as getting the solver state and the best solution, or disposing of a solver.
- **UserTaskServicesClient**: Used to perform all user-task operations (such as to start, claim, or cancel a task) and query tasks by specified field (such as process instances ID or user).
- **UIServicesClient**: Used to get String representation of forms (XML or JSON) and of the process image (SVG).
- **ProcessAdminServicesClient**: Provides an interface for operations with process instances (found in **~/org/kie/server/client/admin**).
- **UserTaskAdminServicesClient**: Provides an interface for operations with user tasks (found in **~/org/kie/server/client/admin**).

The **getServicesClient** method provides access to any of these clients:

```
RuleServicesClient rulesClient = kieServicesClient.getServicesClient(RuleServicesClient.class);
```

4.5. UPDATE API REFERENCES TO BUSINESS CENTRAL KNOWLEDGE STORE

The Knowledge Store REST API has been deprecated in Red Hat Process Automation Manager 7.0, but certain endpoints are still supported so that you can continue to manage Business Central resources. If any of your applications interact with Business Central spaces (previously organizational units), repositories, and projects through Knowledge Store REST API calls (**/business-central/rest/**), you must update these API references according to the new endpoints supported in Red Hat Process Automation Manager 7.0.

Procedure

Wherever applicable in your application code, replace any legacy REST API calls to the Business Central Knowledge Store in the format **http://<SERVER>:<PORT>/business-central/rest/<ENDPOINT>** with the new corresponding Knowledge Store REST API calls listed in Section 4.5.1, “Supported calls to Knowledge Store REST API”.

As a result of the restructured content in Red Hat Process Automation Manager 7.0, the endpoints may differ between the legacy Business Central REST API calls and the corresponding REST API calls in Red Hat Process Automation Manager 7.0. Be sure to use the new endpoints to replace your legacy Business Central REST API calls. If you cannot find a corresponding REST API call in the list to replace a legacy call, then that call is no longer supported in Red Hat Process Automation Manager 7.0.

The following example is an update to a Knowledge Store REST API call for spaces.

Legacy Knowledge Store REST API call:

```
http://localhost:8080/business-central/rest/organizational-units
```

New Knowledge Store REST API call:

```
http://localhost:8080/business-central/rest/spaces
```

4.5.1. Supported calls to Knowledge Store REST API

The following table lists the Knowledge Store REST API calls supported in Red Hat Process Automation Manager 7.0. The base URL for each listed endpoint is **http://<SERVER>:<PORT>/business-central/rest/**.

Example call URL for spaces:

```
http://localhost:8080/business-central/rest/spaces
```

Table 4.1. Knowledge Store REST API operations

Category	Endpoint	HTTP method	Description
Job calls	**/jobs/*JOB_ID***	GET	Returns a job status
	/jobs/*JOB_ID*	DELETE	Removes a job
Project calls	**/projects**	GET	Returns a list of all projects
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*	GET	Returns information about a specific project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*	POST	Creates a project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*	DELETE	Removes a specified project
	/spaces/*SPACE_NAME*/projects	GET	Returns a list of projects in a given space
	/spaces/*SPACE_NAME*/git/clone	POST	Clones a project into the space from a given Git address
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*	DELETE	Removes a specific project from a given space
Space calls	**/spaces**	GET	Returns a list of all spaces

Category	Endpoint	HTTP method	Description
	/spaces/*SPACE_NAME*	GET	Returns information about a specific space
	/spaces	POST	Creates a space
	/spaces/*SPACE_NAME*	DELETE	Removes a specified space
Maven calls	**/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*/maven/compile**	POST	Compiles a project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*/maven/test	POST	Tests a project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*/maven/install	POST	Installs a project
	/spaces/*SPACE_NAME*/projects/*PROJECT_NAME*/maven/deploy	POST	Deploys a project

4.6. CONFIGURE AND START PROCESS SERVER

After you have migrated all Business Central data and have upgraded the relevant database, configure and start the new Process Server to transfer the execution server functionality and apply the migrated changes.

Prerequisites

- Project data from Red Hat JBoss BPM Suite 6.4 has been migrated using the Business Central migration tool.
- The Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 has been upgraded.
- All API references to Business Central execution server in application code have been redirected to the Process Server.

Procedure

Navigate to the Red Hat Process Automation Manager 7.0 **bin** directory and start the new Process Server with the following properties. Adjust the specific properties according to your environment.

```
$ ~/EAP_HOME/bin/standalone.sh --server-config=standalone-full.xml 1
-Dorg.kie.server.id=myserver 2
-Dorg.kie.server.user=process_server_username 3
-Dorg.kie.server.pwd=process_server_password 4
-Dorg.kie.server.controller=http://localhost:8080/business-central/rest/controller 5
-Dorg.kie.server.controller.user=controller_username 6
```

```
-Dorg.kie.server.controller.pwd=controller_password 7
-Dorg.kie.server.location=http://localhost:8080/kie-server/services/rest/server 8
-Dorg.kie.server.persistence.dialect=org.hibernate.dialect.PostgreSQLDialect 9
-Dorg.kie.server.persistence.ds=java:jboss/datasources/psjbpmDS 10
```

1 Start command with **standalone-full.xml** server profile

2 Server ID that must match the server configuration name defined in Business Central

3 User name to connect with Process Server from the Process Automation Manager controller

4 Password to connect with Process Server from the Process Automation Manager controller

5 Process Automation Manager controller location, Business Central URL with **/rest/controller** suffix

6 User name to connect to the Process Automation Manager controller REST API

7 Password to connect to the Process Automation Manager controller REST API

8 Process Server location (on the same instance as Business Central in this example)

9 Hibernate dialect to be used

10 JNDI name of the data source used for your previous Red Hat JBoss BPM Suite database

NOTE

If Business Central and Process Server are installed on separate application server instances (Red Hat JBoss EAP or other), use a separate port for the Process Server location to avoid port conflicts with Business Central. If a separate Process Server port has not already been configured, you can add a port offset and adjust the Process Server port value accordingly in the Process Server properties.

Example:

```
-Djboss.socket.binding.port-offset=150
-Dorg.kie.server.location=http://localhost:8230/kie-server/services/rest/server
```

If the Business Central port is 8080, as in this example, then the Process Server port, with a defined offset of 150, is 8230.

Process Server connects to the new Business Central and collects the list of deployment units (KIE containers) to be deployed.

4.7. VERIFY THE MIGRATION IN BUSINESS CENTRAL

After you have completed all data migration and have configured and started the new Process Server, verify that the migration was successful in the new Business Central for Red Hat Process Automation Manager 7.0.

Prerequisites

- Project data from Red Hat JBoss BPM Suite 6.4 has been migrated using the Business Central migration tool.
- The Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 has been upgraded.
- All API references to Business Central execution server in application code have been redirected to the Process Server.
- The new Process Server has been configured and started.

Procedure

1. After Process Server server starts, log in to Business Central for Red Hat Process Automation Manager 7.0 and navigate to **Menu → Deploy → Execution Servers**
2. Select the server configuration that you created previously and verify that the server is now listed under **Remote Servers** and that each associated **Deployment Unit** is validated.
3. Navigate to **Menu → Manage → Process Instances**and verify that all live process instances have been migrated.
4. Navigate to **Menu → Manage → Tasks**and verify that all live tasks have been migrated.
5. Navigate to **Menu → Design → Projects**, verify that all projects have been migrated, and select each project to verify the migrated project assets.
6. For each project, click **Build** and then **Deploy** in the upper-right corner of the project window to validate the migrated project data and the configured Process Server.

4.7.1. Troubleshooting Business Central migration problems

If you encounter problems with your Business Central migration to Red Hat Process Automation Manager 7.0, review the following troubleshooting suggestions:

- If any project data is missing from Business Central, ensure that the **.niogit** directory for Red Hat Process Automation Manager 7.0 contains the restructured data and is in the correct installation location for Red Hat Process Automation Manager 7.0 (for example, **~/$EAP_HOME/bin/**).
- If projects fail to build, open the project **pom.xml** file and remove the **http://<SERVER>:<PORT>/business-central/maven2/** repository, if present. The use of this repository can prevent projects from building properly in Red Hat Process Automation Manager 7.0. If this repository contains project data that you want to preserve, create a new repository for the data and add the repository as a dependency in the **pom.xml** file.
  Example repository to be removed from **pom.xml** file:

```
<repository>
  <id>guvnor-m2-repo</id>
  <name>Guvnor M2 Repo</name>
  <url>http://localhost:8080/business-central/maven2/</url>
</repository>
```

- If projects fail to deploy to Process Server, review your Process Server installation and configuration. For installation information, see *Planning a Red Hat Process Automation Manager installation*. For configuration information, see Section 4.6, “Configure and start Process

Server”.

- If you encounter other migration problems, review Chapter 6, *Other migration considerations* .
- If you cannot resolve migration problems, complete the migration process again on a new Red Hat Process Automation Manager 7.0 installation.

CHAPTER 5. MIGRATING JAVA PROJECT DATA

Projects that you developed in Java code in Red Hat JBoss BPM Suite 6.4 (for example, in Eclipse) also require modification for Red Hat Process Automation Manager 7.0. You must update the dependencies in the **pom.xml** file for each project, upgrade the database configured for the project, and rebuild the project. To migrate your Java client applications, you also must update the dependencies in the **pom.xml** file for each project. If your application uses embedded Red Hat JBoss BPM Suite engines (jBPM, Drools, OptaPlanner), this change also updates the engines. If the application calls the Process Server, the API client library is updated.

5.1. UPDATE JAVA PROJECT DEPENDENCIES

You must update several dependencies in your Java projects to prepare them for migration to Red Hat Process Automation Manager 7.0.

Prerequisite

The Maven repository for Red Hat Process Automation Manager 7.0 has been downloaded from the Red Hat Customer Portal and made available to the local Maven installation (as a local or remote repository).

Procedure

1. Open the **pom.xml** file of the project and remove the **<version>** tag from any dependencies under the following groups:

    - **org.kie**
    - **org.drools**
    - **org.jbpm**
    - **org.optaplanner**

2. Add the following dependency under the **<dependencies>** tag within the **<dependencyManagement>** section:

    ```
    <dependency>
      <groupId>org.jboss.bom</groupId>
      <artifactId>rhpam-platform-bom</artifactId>
      <version>${rhpam.version}</version>
      <scope>import</scope>
      <type>pom</type>
    </dependency>
    ```

    The **${rhpam.version}** is the version of the **org.jboss.bom:rhpam-platform-bom** artifact in the Maven repository for Red Hat Process Automation Manager 7.0. You can view the version of the artifact by entering the repository and navigating to the **maven-repository/org/jboss/bom/rhpam/rhpam-platform-bom** directory. The name of the only sub-directory in this directory is the version. For example: **7.0.0.GA-redhat-1**.

3. If your code uses any Drools CDI annotations (**@KReleaseId** , **@KContainer**, **@KBase**, **@KSession**), also add the following dependency:

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

```
  <artifactId>drools-cdi</artifactId>
</dependency>
```

This dependency is necessary because the CDI extension for processing the annotations is now defined using a separate module. Without this dependency, the annotations do not work.

4. Replace the legacy **jbpm-workitems** dependency (if applicable) with one or more of the new dependencies. The new dependencies that you choose depend on the work items that your code uses.
   Legacy **jbpm-workitems** dependency:

   ```
   <dependency>
     <groupId>org.jbpm</groupId>
     <artifactId>jbpm-workitems</artifactId>
   </dependency>
   ```

   New **jbpm-workitems** dependency options:

   ```
   <dependency>
     <groupId>org.jbpm</groupId>
     <artifactId>jbpm-workitems-core</artifactId>
   </dependency>
   <dependency>
     <groupId>org.jbpm</groupId>
     <artifactId>jbpm-workitems-bpmn2</artifactId>
   </dependency>
   <dependency>
     <groupId>org.jbpm</groupId>
     <artifactId>jbpm-workitems-email</artifactId>
   </dependency>
   <dependency>
     <groupId>org.jbpm</groupId>
     <artifactId>jbpm-workitems-jms</artifactId>
   </dependency>
   <dependency>
     <groupId>org.jbpm</groupId>
     <artifactId>jbpm-workitems-rest</artifactId>
   </dependency>
   <dependency>
     <groupId>org.jbpm</groupId>
     <artifactId>jbpm-workitems-webservice</artifactId>
   </dependency>
   ```

5. Replace the legacy **org.guvnor** dependency (if applicable) with the new **org.uberfire** dependency.
   Legacy **org.guvnor** dependency:

   ```
   <dependency>
     <groupId>org.guvnor</groupId>
     <artifactId>guvnor-rest-client</artifactId>
   </dependency>
   ```

   New **org.uberfire** dependency:

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

6. Replace the legacy **org.kie.remote** dependency (if applicable) with the new **org.kie.server** dependency.
   Legacy **org.kie.remote** dependency:

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

   New **org.kie.server** dependency:

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

   This dependency is necessary because Business Central no longer has execution server capabilities nor API endpoints related to the execution server.

7. Update the version of the project artifact and save the **pom.xml** file.

5.2. UPGRADE YOUR DATABASE

Red Hat Process Automation Manager 7.0 contains updates to the database schema for optimal performance with the new Business Central and Process Server. You must update the Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 using one of the upgrade scripts provided with this release to accommodate the change.

Prerequisites

- The **Red Hat Process Automation Manager 7.0.0 Add-Ons**distribution has been downloaded from the Red Hat Customer Portal and extracted to a temporary directory.
- The database has been backed up, as a precaution.

Procedure

1. Navigate to the extracted **rhpam-7.0.0-add-ons** folder in your temporary directory and extract the **rhpam-7.0-migration-tool.zip** sub-folder (if not extracted already).
2. Navigate to the extracted **rhpam-7.0-migration-tool** folder and open the **upgrade-scripts** folder. This folder contains database upgrade scripts that are required for applying changes to the database model made in Red Hat Process Automation Manager 7.0.
3. Log on to the Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 and run the relevant Red Hat Process Automation Manager 7.0 upgrade script, according to the supported database type. This prepares the database for Red Hat Process Automation Manager 7.0.

4. If you use Quartz as your scheduling service, additionally run the **quartz-upgrade.sql** script provided in the relevant database folder within the **upgrade-scripts** folder. This script updates your Quartz version to 2.2.3 in compliance with Red Hat Process Automation Manager 7.0 requirements.

5.3. VERIFY THE JAVA PROJECT MIGRATION

After you have migrated all Java project data and have upgraded the relevant database, rebuild the Java project to verify successful migration.

Prerequisites

- Java project dependencies have been updated.
- The Red Hat JBoss BPM Suite 6.4 database to be used with Red Hat Process Automation Manager 7.0 has been upgraded.
- Maven 3.x or later is installed.
- Java 8 or later is installed.

Procedure

1. Rebuild each Java project using a regular Maven build.

   ```
   $ mvn clean install
   ```

2. Review all project data in each build to confirm successful migration.
   If you encounter build errors, verify that the project dependencies have been updated correctly. To review other product changes that may have impacted your Java project, see Chapter 6, *Other migration considerations*.

CHAPTER 6. OTHER MIGRATION CONSIDERATIONS

Red Hat Process Automation Manager 7.0 contains changes to APIs, rule logic, and Red Hat Business Optimizer features that you should note during your migration. These changes are generally backward compatible between product versions, but in some cases, migration effort is required to resolve build or migration errors.

As part of your migration to Red Hat Process Automation Manager 7.0, review these changes and address any inconsistencies or errors in your projects that arise.

6.1. RED HAT BUSINESS OPTIMIZER CHANGES IN RED HAT PROCESS AUTOMATION MANAGER 7.0

Red Hat Business Optimizer is an embeddable planning engine in Red Hat Process Automation Manager that optimizes planning problems. Red Hat Business Optimizer is based on the community OptaPlanner project that is regularly updated and in some cases requires code changes for the latest Red Hat Business Optimizer features. For an overview of the latest OptaPlanner changes and migration requirements, see the OptaPlanner upgrade recipe archive . OptaPlanner upgrade information for versions 7.0 through 7.7, inclusive, is relevant for upgrading from Red Hat JBoss BPM Suite 6.4 to Red Hat Process Automation Manager 7.0.

In Red Hat Process Automation Manager 7.0, certain Red Hat Business Optimizer configurations in Business Central must be updated to accommodate recent OptaPlanner changes.

6.1.1. Updating Red Hat Business Optimizer asset configurations in Business Central

If you have any solver configuration assets (**.solver.xml** files) or solution-related data objects in Business Central, you must make certain updates to these assets in Red Hat Process Automation Manager 7.0 to accommodate recent Red Hat Business Optimizer changes.

Prerequisite

Business Central data has been migrated from Red Hat JBoss BPM Suite 6.4 to Red Hat Process Automation Manager 7.0. For example, see Chapter 3, *Migrating Business Central as design environment only*.

Procedure

1. Log in to Business Central for Red Hat Process Automation Manager 7.0.
2. Go to **Menu → Design → Projects** and select the project name.
3. Open any solver configuration asset (**.solver.xml** file), if present.
4. In the solver configuration designer, click **Save** without making any changes. This step is required to generate the new code for solver configuration assets in Red Hat Process Automation Manager 7.0. Do this step with any other solver configuration assets.
5. Under **Data Objects** in the **Project Explorer** (left menu), open any data object (**.java** file) configured as a **Planning Solution**, if applicable.
   To verify if this setting is selected for this data object, click the OptaPlanner icon on the right side of the data objects designer. If **Planning Solution** is not selected, then this procedure does not apply.

6. Under **general properties** in the data objects designer, set the **Superclass** drop-down option to **Nothing selected**. This setting is no longer required by Red Hat Business Optimizer in Red Hat Process Automation Manager 7.0.

7. On the right side of the data objects designer, click the OptaPlanner icon to reveal the **Planner Settings**. The **Planning Solution** option should be selected (if not, then this procedure does not apply).

8. Select **No selected**, then re-select **Planning Solution** and specify the **Solution Score Type**. This step is required to generate the new code for planning solutions in Red Hat Process Automation Manager 7.0.

9. Click **Save** in the data objects designer and make the same changes to any other data objects configured as a **Planning Solution**.

6.2. API CHANGES IN RED HAT PROCESS AUTOMATION MANAGER 7.0

Many of the API changes from Red Hat JBoss BPM Suite 6.4 to Red Hat Process Automation Manager 7.0 are backward compatible and do not affect migration. However, the following table lists class changes in APIs that may not be compatible in all cases.

If you encounter build or migration errors related to these changes, review the full List of API changes in the Red Hat Customer Portal and update your code as needed.

Table 6.1. API changes in Red Hat Process Automation Manager 7.0 (abbreviated)

API	Description	Changed classes
Drools core (**drools-core**)	The process engine	● **org.drools.core.command.*** ● **org.drools.core.common.***
KIE API (**kie-api**)	The main API for all projects from KIE Group	● **org.kie.api.task.*** ● **org.kie.api.executor.*** ● **org.kie.api.concurrent.*** ● **org.kie.api.builder.*** ● **org.kie.api.command.*** ● **org.kie.api.runtime.***

API	Description	Changed classes
KIE server API (**kie-server-api**)	The general Process Server API	- **org.kie.server.api.commands.*** - **org.kie.server.api.marshalling.*** - **org.kie.server.api.model.*** - **org.kie.server.api.rest.RestURI** (constants have slightly changed, omitting leading /)
KIE server client API (**kie-server-client**)	The API for Process Server client	- **org.kie.server.client.SolverServicesClient** - **org.kie.server.client.UIServicesClient** - **org.kie.server.client.admin.ProcessAdminServicesClient** - **org.kie.server.client.ProcessServicesClient** - **org.kie.server.client.QueryServicesClient** - **org.kie.server.client.JobServicesClient** - **org.kie.server.client.UserTaskServicesClient** - **org.kie.server.client.KieServicesClient** - **org.kie.server.client.KieServicesConfiguration**
KIE server controller API (**kie-server-controller-api**)	The API for the headless Process Automation Manager controller	- **org.kie.server.controller.api.service.***
KIE server controller REST API (**kie-server-controller-rest**)	The REST API for headless Process Automation Manager controller	- **org.kie.server.controller.rest.RestSpecManagementServiceImpl** - **org.kie.server.controller.rest.RestKieServerControllerImpl**

6.3. LOGIC CHANGES IN RED HAT PROCESS AUTOMATION MANAGER 7.0

Note the following logic changes in Red Hat Process Automation Manager 7.0:

- In Red Hat JBoss BPM Suite, when a rule executes the sum function in an accumulate pattern, the result always returns as a double data type regardless of the data type of the inputs. Red Hat Process Automation Manager 7.0 preserves the data type of the inputs on which the sum is executed. This enhancement provides a more accurate result from the sum function. In the following example, the result type of the accumulate function is **Long** instead of **double**:

  ```
  Long(...) from accumulate(..., sum($p.getLongWeight()))
  ```

  If the rules in your Red Hat JBoss BPM Suite 6.4 project include the sum function in an accumulate pattern, search for these functions and review them. These functions return a double data type in Red Hat JBoss BPM Suite 6.4, but will return the data type of the input values in Red Hat Process Automation Manager 7.0.

- When no fact matches the accumulate pattern, **min** and **max** accumulate functions in Red Hat Process Automation Manager 7.0 do not return **+/-Integer.MAX_VALUE**, but return **null**. Therefore, unlike in Red Hat JBoss BPM Suite 6.4, the accumulate in the rule is not matched and the rule does not fire in Red Hat Process Automation Manager 7.0.

- Business processes that contain business rule tasks with an **implementation=Java** configuration will not be compiled in Red Hat Process Automation Manager 7.0 due to stricter validation requirements. To resolve compilation errors related to this restriction, set the implementation configuration to **implementation=##unspecified** or remove the **implementation** attribute.

CHAPTER 7. NEXT STEPS

- *Getting started with decision services*
- *Designing a decision service using guided decision tables*
- *Getting started with business processes*

APPENDIX A. VERSIONING INFORMATION

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