Red Hat JBoss Data Virtualization 6.4

Migration Guide

This guide is for installation teams
This guide is for installation teams

Red Hat Customer Content Services
Abstract

If you are running an older version of Red Hat JBoss Data Virtualization and you want to upgrade to the latest version, there are some tasks you must perform manually. Firstly, you have to back up your repository to mitigate against the risk of data loss. Secondly, there are some discrepancies between an instance of Red Hat JBoss Data Virtualization that has been patched to the latest version and a completely fresh installation. To resolve these differences, you must perform some manual configuration tasks. This book guides you through these tasks.
# Table of Contents

**CHAPTER 1. MIGRATING YOUR DATA**
- 1.1. BACK UP YOUR REPOSITORY .................................................. 3
- 1.2. RESTORE YOUR REPOSITORY .................................................. 4

**CHAPTER 2. CONFIGURING YOUR SYSTEM**
- 2.1. REQUIRED MIGRATION TASKS ............................................... 5
- 2.2. OTHER CHANGES TO RED HAT JBOSS DATA VIRTUALIZATION ............. 7

**APPENDIX A. REVISION HISTORY** .................................................. 8
CHAPTER 1. MIGRATING YOUR DATA

Use Red Hat JBoss Data Virtualization’s backup and restore functionality to migrate the data stored in your hierarchical database. Each backup you create will contain all of the content for all of the workspaces in a single repository. Therefore, you will need to back up and restore each of your repositories one by one.

The backup and restore functionality is provided by the org.modeshape.jcr.api.RepositoryManager interface:

```java
public interface RepositoryManager {
    ...
    Problems backupRepository( File backupDirectory ) throws RepositoryException;
    ...
    Problems restoreRepository( File backupDirectory ) throws RepositoryException;
    ...
}
```

Run this code in an Apache Jackrabbit (JCR) session in any workspace to access the repository manager:

```java
javax.jcr.Repository repository = ...
javax.jcr.Credentials credentials = ...
String workspaceName = ...
javax.jcr.Session session = repository.login(credentials,workspaceName);
org.modeshape.jcr.api.Session msSession =
    (org.modeshape.jcr.api.Session)session;
org.modeshape.jcr.api.RepositoryManager repoMgr =
    ((org.modeshape.jcr.api.Session)session).getWorkspace().getRepositoryManager();
```

1.1. BACK UP YOUR REPOSITORY

Use the `backupRepository` method provided by the org.modeshape.jcr.api.RepositoryManager interface to back up your repository. All of the workspaces that existed when the back up was initiated will be saved. If you call the backup method on a repository that is in use, all of the changes made while the backup process is underway are also saved.

NOTE

The method blocks other calls until the backup is completed, so you must invoke the method asynchronously to avoid disruptions.

Each backup is stored on the file system in a series of `Gzip` archives. A sub-directory for storing large binaries is also created.
To back up your repository, run this code in your Apache JCR session:

```java
org.modeshape.jcr.api.RepositoryManager repoMgr = ...  
java.io.File backupDirectory = ...  
Problems problems = repoMgr.backupRepository(backupDirectory);  
if ( problems.hasProblems() ) {  
    System.out.println("Problems restoring the repository:");  
    // Report the problems (we'll just print them out) ...  
    for ( Problem problem : problems ) {  
        System.out.println(problem);  
    }
} else {  
    System.out.println("The backup was successful");
}
```

**NOTE**

You must initiate each operation. There is no way to automatically schedule a backup.

### 1.2. RESTORE YOUR REPOSITORY

Once you have a complete backup, you can restore it after upgrading your system by using the `restoreRepository` method provided by the `org.modeshape.jcr.api.RepositoryManager` interface.

Run this code in your Apache JCR session:

```java
org.modeshape.jcr.api.RepositoryManager repoMgr = ...  
java.io.File backupDirectory = ...  
Problems problems = repoMgr.restoreRepository(backupDirectory);  
if ( problems.hasProblems() ) {  
    System.out.println("Problems backing up the repository:");  
    // Report the problems (we'll just print them out) ...  
    for ( Problem problem : problems ) {  
        System.out.println(problem);  
    }
} else {  
    System.out.println("The restoration was successful");
}
```

**WARNING**

Do not delete your old data and backups before verifying that your migration is a success.
CHAPTER 2. CONFIGURING YOUR SYSTEM

2.1. REQUIRED MIGRATION TASKS

If you are upgrading Red Hat JBoss Data Virtualization from version 6.3 to 6.4, you must make some configuration changes. This is because the installer does not manually update these files:

1. Edit the [EAP_HOME]/domain/configuration/domain.xml file to change the Salesforce resource adapter in the ha profile. (You do not need to change the standalone configuration):

   `<resource-adapter id="salesforce-34">
   <module slot="main" id="org.jboss.teiid.resource-adapter.salesforce"/>
   </resource-adapter>`

   Change the 'salesforce' part of the resource adapter id to 'salesforce-34'. The resulting resource adapter will look like this:

   `<resource-adapter id="salesforce-34">
   <module slot="main" id="org.jboss.teiid.resource-adapter.salesforce-34"/>
   </resource-adapter>`

2. Edit the [EAP_HOME]/standalone/configuration/standalone.xml, [EAP_HOME]/standalone/configuration/standalone-ha.xml and [EAP_HOME]/standalone/configuration/standalone-full-ha.xml files and the ha and full-ha profiles of the [EAP_HOME]/domain/configuration/domain.xml file and add these settings and save:

   `<translator name="couchbase" module="org.jboss.teiid.translator.couchbase"/>
   <translator name="amazon-s3" module="org.jboss.teiid.translator.amazon.s3"/>`

3. Edit the [EAP_HOME]/standalone/configuration/standalone.xml, [EAP_HOME]/standalone/configuration/standalone-ha.xml and [EAP_HOME]/standalone/configuration/standalone-full-ha.xml files and the ha and full-ha profiles of the [EAP_HOME]/domain/configuration/domain.xml file, add this resource adapter, and save:

   `<resource-adapter id="couchbase">
   <module slot="main" id="org.jboss.teiid.resource-adapter.couchbase"/>
   </resource-adapter>`

4. Edit the [EAP_HOME]/standalone/configuration/standalone.xml, [EAP_HOME]/standalone/configuration/standalone-ha.xml and [EAP_HOME]/standalone/configuration/standalone-full-ha.xml files and delete this section:

   `<local-cache name="resultset-repl" batching="true">
   <locking isolation="READ_COMMITTED"/>`
2.1.1. Migration steps for the Data Services Builder

If you are upgrading from Red Hat JBoss Data Virtualization 6.3 to 6.4, the Data Services Builder will not work initially. You must deploy it manually (this step is only required if you are going to use the Data Services Builder and should only be performed on a non-production system):

1. Create a keystore:
   
   ```
   keytool -genkeypair -keystore /path/ks.jks -storepass pass11 -keypass pass11 -alias mykey -dname CN=demo,OU=demo,O=demo,L=demo,ST=demo,C=demo -keyalg RSA -keysize 2048
   ```

2. Start the server in admin mode.

3. Create the HTTPS web connector for the profile you are using on the Red Hat JBoss Enterprise Application Platform:
   
   ```
   /subsystem=web/connector=HTTPS:add(name=HTTPS,protocol=HTTP/1.1,scheme=https,socket-binding=https,secure=true)
   ```

4. Apply the secure socket layer to the certificate:
   
   ```
   /subsystem=web/connector=HTTPS/configuration=ssl:add(name=https,key-alias=mykey,password=pass11,certificate-key-file=/path/ks.jks)
   ```

5. Execute the CLI scripts for the profile you intend to use in order to deploy the Data Services Builder:

   **Standalone Profile**
   
   - [EAP_HOME]/cli-scripts/teiid-standalone-install-ds-builder-war.cli
   - [EAP_HOME]/cli-scripts/teiid-standalone-install-vdb-builder-war.cli

   **Domain Profile**
   
   - [EAP_HOME]/cli-scripts/teiid-domain-install-ds-builder-war.cli
2.2. OTHER CHANGES TO RED HAT JBOSS DATA VIRTUALIZATION

**Week Function**

Red Hat JBoss Data Virtualization now consistently returns the ISO 8601 standard for `WEEK` statements. Previously, `locale` was used for many databases. This change can potentially break functionality for users of databases such as H2, Microsoft SQL Server, and Sybase. Avoid using `locale` with these databases.

**ibm.jdk Dependency is Missing From the org.jboss.security.negotiation Module**

The `ibm.jdk` dependency is not present in the `org.jboss.security.negotiation` module, nor is it in the patched version in the `.overlay` directory. You must copy this dependency into the directory every time you patch the Red Hat JBoss Enterprise Application Platform.

**UDF Functions Definition Model**

The UDF Function Definitions model is no longer supported. To avoid compatibility problems, rebuild any models you created with it.
APPENDIX A. REVISION HISTORY

Revision 6.4.0-43
Updates for 6.4.
Fri Nov 10 2017
David Le Sage