PEGA MARKLOGIC CONNECTOR

NOVEMBER 2021

Contents

GENERAL INFORMATION

COMPONENT DESCRIPTION

The Pega MarkLogic Connector implements the Pega Repository API for the MarkLogic NoSQL Database.

MarkLogic is an Operational and Transactional Enterprise NoSQL Database that is multi-model, natively storing data as JSON, XML and RDF. Clients that want to use Pega Infinity and MarkLogic can now have case attachments and content be stored in a MarkLogic database.

By adding this component to a Pega application, you can integrate your application with a MarkLogic database using the new Repository interface. You can select MarkLogic as a Repository type when creating a new Repository. By enabling the use of the MarkLogic repository from the Application definition, all attachments and content will automatically be stored in the MarkLogic database.

More on Repository API is available here: https://community.pega.com/knowledgebase/articles/custom-repository-types.

COMPATIBILITY

Pega 8.5.3 or greater

INSTALLATION

INSTALLATION OF THE MARKLOGIC ECM SERVICES

In order to use the Pega MarkLogic Component, you need an existing MarkLogic cluster that has been configured with the corresponding backend Enterprise Case Management (ECM) services to support the Pega MarkLogic Component.

The connector distribution can be downloaded from developer.marklogic.com. It contains a Gradle project that will configure MarkLogic to expose the required REST endpoints and transforms.

In order to deploy this project's application, you'll first need the following software installed:

- Java 8 to 16 (required by Gradle)
- MarkLogic 10.0 (preferably the latest version)

The project uses Gradle but it includes the Gradle wrapper, which means you don't need to install Gradle locally.

The project also uses the Gradle properties plugin to select the corresponding properties file to use to set the host, username, and password for deploying to an instance of MarkLogic.

To deploy the project to MarkLogic, first unzip the distribution archive that you downloaded from developer.marklogic.com.

In the “marklogic-pega-connector-1.0.0” folder, create a gradle properties file for your environment. For example, for the "local" environment, create `./gradle-local.properties` and define the following properties:

- mlHost ("localhost" if you're running MarkLogic locally; otherwise, the host of your remote MarkLogic cluster)
- mlUsername (the username to run the deployment as)
- mlPassword (the password for the user defined by mlUsername)

Use the mlDeploy task to deploy the services. If no "env" is specified, it defaults to "local" (the "-i" flag is included for info-level logging, which can be helpful for debugging if there is an error).

```
./gradlew -i mlDeploy
```

If you want to deploy to an environment other than "local", for example "test", create `./gradle-test.properties` and define the properties as listed above and use the following to deploy:

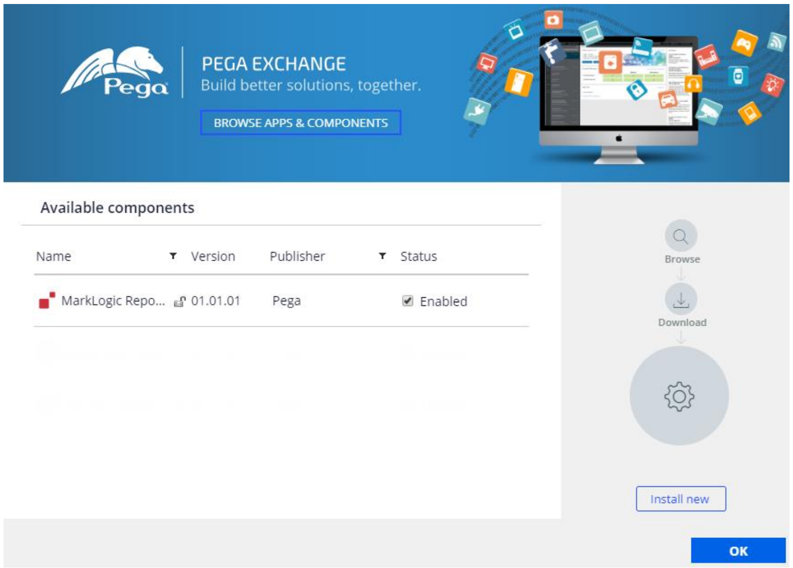
```
./gradlew -i mlDeploy -Penv=test
```

***Note**: For Windows users, replace “./gradlew” with “gradlew.bat” above.*

Once this has been run, you can install the Pega MarkLogic Connector and configure it to point to your instance of MarkLogic.

INSTALLATION OF THE PEGA MARKLOGIC CONNECTOR

The Pega MarkLogic Repository Component is contained in the “marklogic-pega-connector-1.0.0/rulesets” folder that was created when the distribution archive was unzipped above. Open the Application rule form then, under the Enabled components section, click on Manage components button and then click on Install new and browse to the already downloaded component and then click OK.



Available components

Name	Version	Publisher	Status
MarkLogic Repo...	01.01.01	Pega	☑ Enabled

Verify that the Component is listed on the Application rule from and save the Application rule.

CONFIGURATION

The Pega MarkLogic Repository Component provides an implementation of the Repository API.

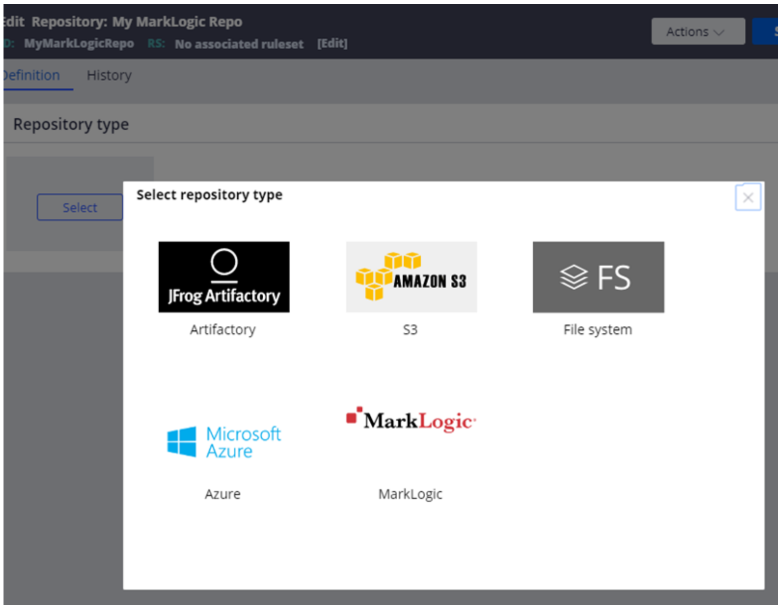
Important: Prior to configuring the Repository instance, make sure the Operator account that will be creating the Repository has the PegaRULES:RepositoryAdministrator role.

To use MarkLogic as the Repository provider:

CONFIGURE THE REPOSITORY INSTANCE

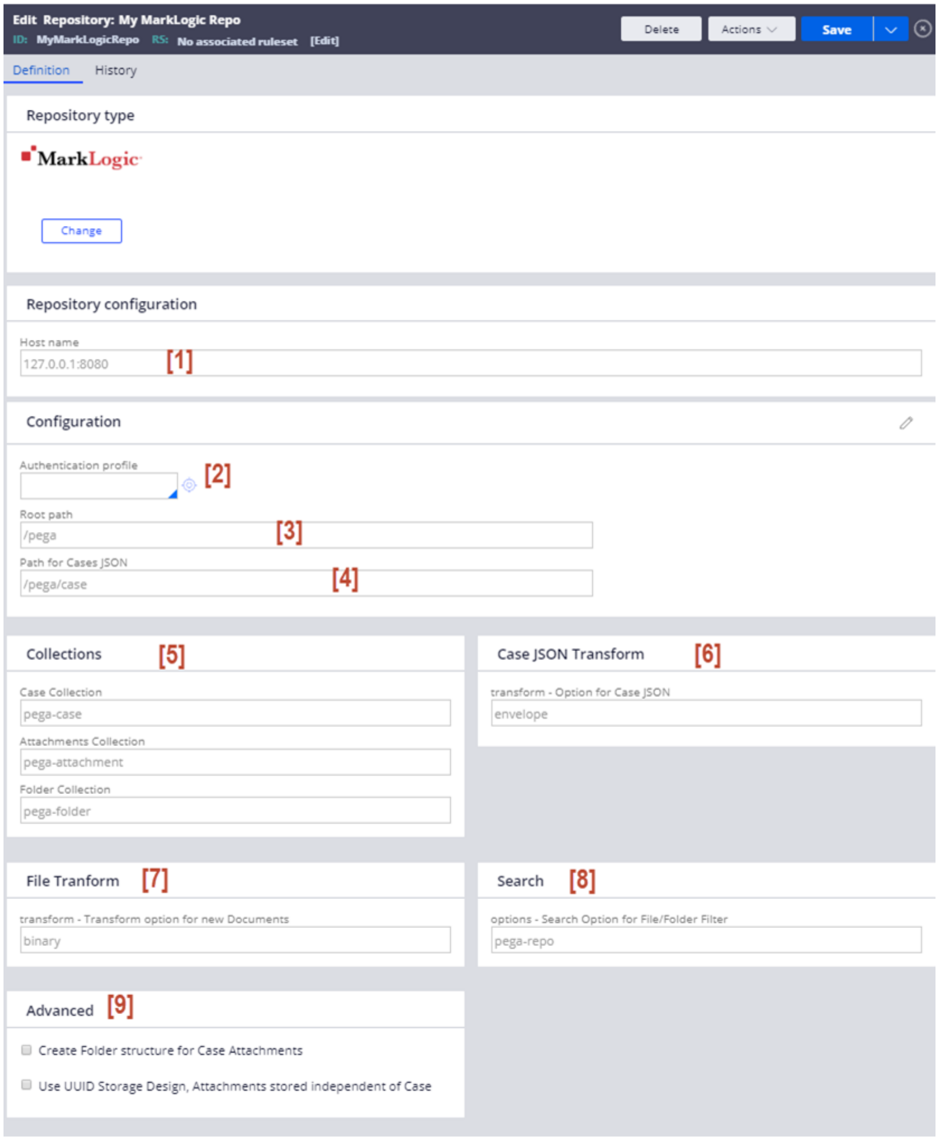
1) Create a New Repository rule (Create->SysAdmin->Repository), provide your Repository Name/Description.

2) On the Repository Type selection screen, you will have a MarkLogic repository. Select it:



(

3. Fill in the Repository Settings:



Edit Repository: My MarkLogic Repo
ID: MyMarkLogicRepo RS: No associated ruleset [Edit]

Repository type

Repository configuration

Host name: 127.0.0.1:8080 [1]

Configuration

Authentication profile: [2]

Root path: /pega [3]

Path for Cases JSON: /pega/case [4]

Collections [5]

Case Collection: pega-case

Attachments Collection: pega-attachment

Folder Collection: pega-folder

Case JSON Transform [6]

transform - Option for Case JSON: envelope

File Tranform [7]

transform - Transform option for new Documents: binary

Search [8]

options - Search Option for File/Folder Filter: pega-repo

Advanced [9]

□ Create Folder structure for Case Attachments

□ Use UUID Storage Design, Attachments stored independent of Case

Fields

[1] **Host Name**: MarkLogic server and port will be used as the root to append the REST services: `HostName/v1/documents`

[2] **Authentication Profile**: Create a new Basic Authentication Profile with the credentials for the MarkLogic account. Make sure ‘Pre-emptive authentication’ is checked.

[3] **Root Path**: Root URI for Pega documents. Used to Test Connectivity

[4] **Path for Cases JSON**: Root URI for where Case JSON documents are placed

[5] **Collections**: The different types of documents from Pega will be tagged by the collections string are defined here.

- Case Collection: Used for Case JSON documents
- Attachment Collection: File attachments to case/pulse
- Folder Collection: Folder objects

[6] **Case JSON Transform**: The REST document transform options for Case JSON documents.

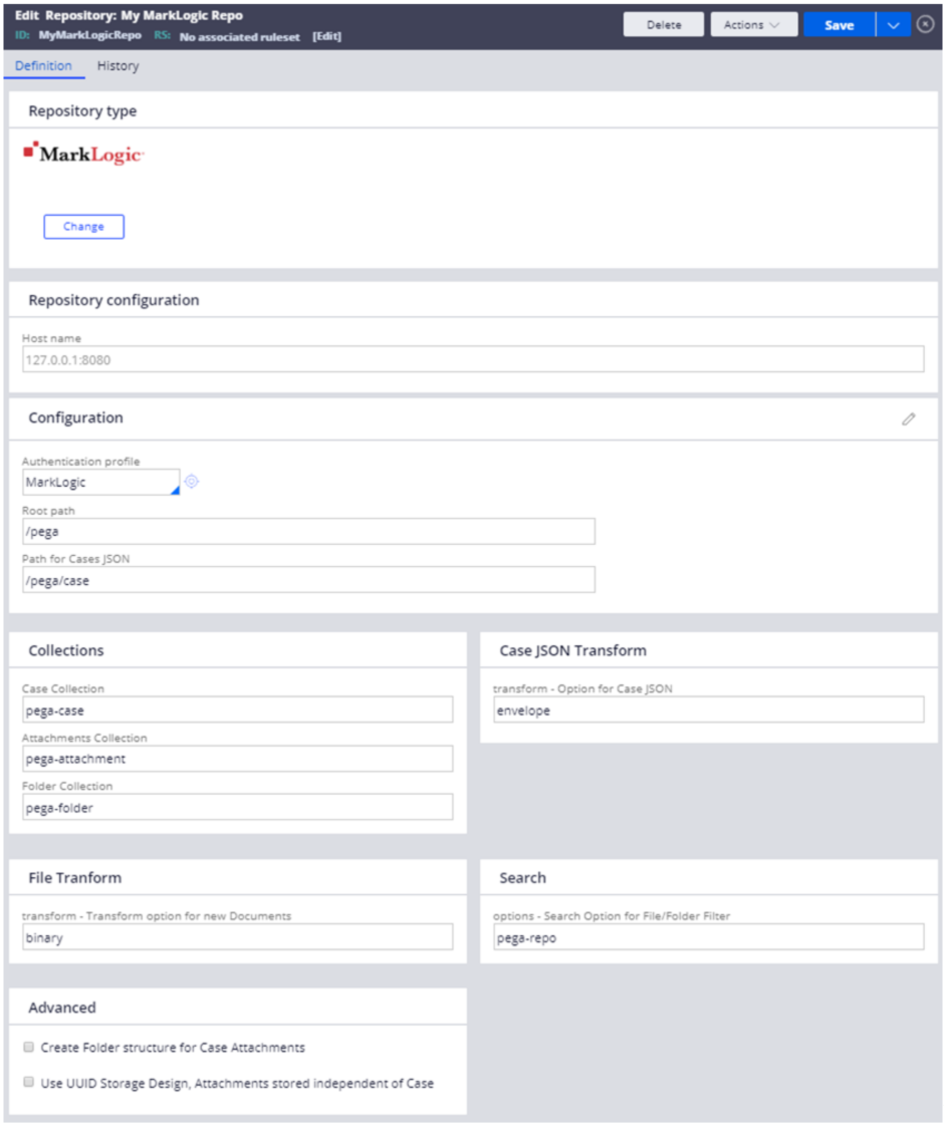
[7] **File Transform**: The REST document transform options for File attachments.

[8] **Search Options**: The pre-configured search option that filters for Attachment/Folder collections.

[9] **Advanced**: Toggle on to create dedicated Case URI paths.

- Example: `/pega/case/B-1/hello123_B-1.txt` rather than `/pega/case/hello123_B-1.txt`

Completed Sample:



Edit Repository: My MarkLogic Repo

ID: MyMarkLogicRepo RS: No associated ruleset [Edit]

Repository type

MarkLogic

Repository configuration

Host name: 127.0.0.1:8080

Configuration

Authentication profile: MarkLogic

Root path: /pega

Path for Cases JSON: /pega/case

Collections

Case Collection: pega-case

Attachments Collection: pega-attachment

Folder Collection: pega-folder

Case JSON Transform

transform - Option for Case JSON: envelope

File Tranform

transform - Transform option for new Documents: binary

Search

options - Search Option for File/Folder Filter: pega-repo

Advanced

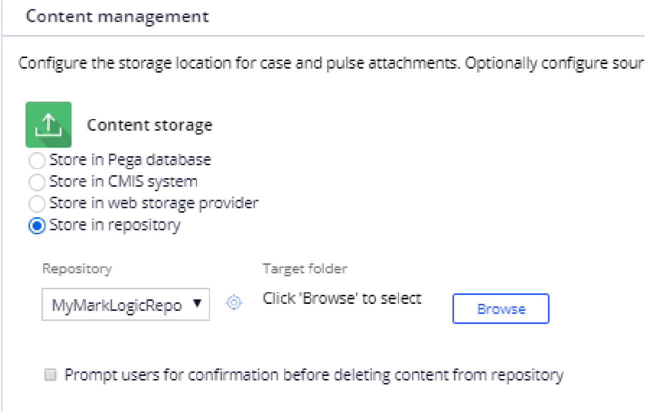
□ Create Folder structure for Case Attachments

□ Use UUID Storage Design, Attachments stored independent of Case

)

UPDATE APPLICATION TO USE THE REPOSITORY

Update the Application Integration & Security tab to toggle on the Content Storage from the new Repository:



Content management

Configure the storage location for case and pulse attachments. Optionally configure sour

Content storage

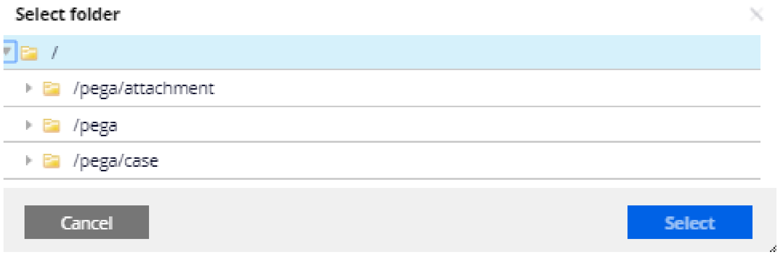
○ Store in Pega database
○ Store in CMIS system
○ Store in web storage provider
● Store in repository

Repository: MyMarkLogicRepo ▼

Target folder: Click 'Browse' to select

□ Prompt users for confirmation before deleting content from repository

Browse the Repository to select the root path where all the Pega content will be stored:



Select folder

- /
- /pega/attachment
- /pega
- /pega/case

FEATURE LIST

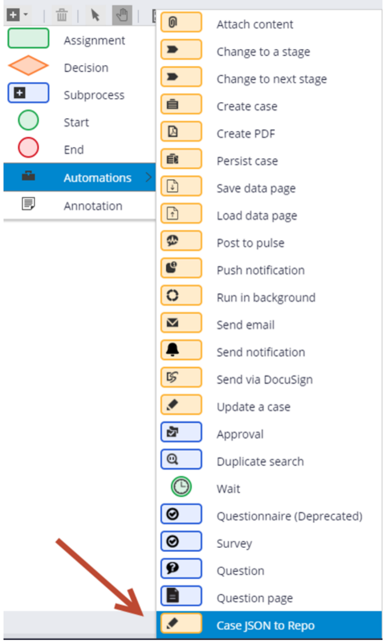
Included Data Pages

Page Name	MarkLogic Endpoint [Methods]	Description
D_MLRepoBasicSearch	v1/search [GET]	Enables Basic Search with options, collection, directory, q, view and pageLength parameters.
D_MLRepoDelete	v1/search [DELETE] v1/documents [DELETE] v1/resources/files [DELETE]	Repository API page. Depending on the advanced toggles, will delete/recursively delete based on path/uri.

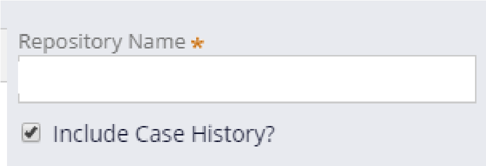
D_MLRepoExists	**v1/documents [GET]** **v1/resources/files [GET]**	**Repository API page. Depending on the advanced toggles, will check for the existence of a file/document based on path/uri.**
D_MLRepoGetFile	v1/documents [GET] v1/resources/files [GET]	Repository API page. Depending on the advanced toggles, will get the file/document based on path/uri.
D_MLRepoIsAuthenticated	v1/documents [GET]	Repository API page. Will ping the repository to check if the AuthenticationProfile is valid.
D_MLRepoListFiles	v1/search [GET]	Repository API page. Will query the repository to get back the files/folders based on an initial folder uri.
D_MLRepoNewCase	v1/documents [PUT]	Will push a JSON document representing a case to the repository.
D_MLRepoNewCaseHistory	v1/documents [PUT]	Will push a JSON document representing a case history to the repository.
D_MLRepoNewFile	v1/documents [PUT] v1/resources/files [PUT]	Repository API page. Will create/push a document to the repository for a specific uri/path.
D_MLRepoNewFolder	v1/documents [PUT]	Repository API page. Will create/push a folder “document” to the repository for a specific uri/path.

SMART SHAPES

The component adds a new Automation Smart Shape to workflow editing:



The Smart Shape takes the following parameters:



Repository Name *

☑ Include Case History?

- ○ **Repository Name**: Name of the Repository Rule
- ○ **Include Case History flag**: To toggle sending the case history

The Smart Shape will call D_MLRepoNewCase and D_MLRepoNewCaseHistory if the flag is set. It will use the current case context when sending the JSON data.

About MarkLogic

For over a decade, organizations around the world have come to rely on MarkLogic to power their innovative information applications. As the world's experts at integrating data from silos, MarkLogic's operational and transactional Enterprise NoSQL database platform empowers our customers to build next generation applications on a unified, 360-degree view of their data. Headquartered in Silicon Valley, MarkLogic has offices throughout the U.S., Europe, Asia, and Australia.

MarkLogic®

999 SKYWAY ROAD, SUITE 200, SAN CARLOS, CA 94070
+1 650 655 2300 · +1 877 992 8885 · www.marklogic.com · sales@marklogic.com