Getting the most out of Reporting on Collaborative Lifecycle Management Data with JRS and RPE



RPE Labs

February 2016 edition

NOTICES

This information was developed for products and services offered in the USA.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing IBM Corporation North Castle Drive, MD-NC119 Armonk, NY 10504-1785 United States of America

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

TRADEMARKS

IBM, the IBM logo, and ibm.com are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Cell Broadband Engine is a trademark of Sony Computer Entertainment, Inc. in the United States, other countries, or both and is used under license therefrom.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

IT Infrastructure Library is a Registered Trade Mark of AXELOS Limited.

ITIL is a Registered Trade Mark of AXELOS Limited.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

Linear Tape-Open, LTO, the LTO Logo, Ultrium, and the Ultrium logo are trademarks of HP, IBM Corp. and Quantum in the U.S. and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

© Copyright International Business Machines Corporation 2015.

This document may not be reproduced in whole or in part without the prior written permission of IBM.

US Government Users Restricted Rights - Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Table of contents

/erview /ercises	
Exercise 1: Exploring the Rational Publishing Engine Desktop Interface	4
RPE Launcher	4
RPE Document Studio	10
Exercise 2: Create a requirements document	14
Create a new Document Template	14
Add a data source for the DNG Modules	14
Add elements to print the module name and description	16
Save the template	18
Configure, run and save the report	19
Add elements to iterate requirement	21
Add requirement details	22
Result	28
Exercise 3: Creating a traceability document with RPE	29
Prepare to start	29
Add test data	29
Configure, run and save the report	31
Adding the traceability information	32
Final test	37
Exercise 4: Creating a traceability document with RPE and JRS	38

Prepare to start	38		
Create the query	38		
Export the query and use it with RPE	43		
Modify the template	44		
Exercise 5: Formatting a document			
Prepare to start	45		
Add headers and footers to the document	45		
Use a style sheet with the report	49		
Using document regions	50		
Using variables to customize the document	51		
Adding a table of contents	53		
Exercise 6: Create a test plan document from RQM	54		
Before you start	54		
Add the data source	54		
Details	58		
Exercise 7: Create a sprint document from RTC	62		
Before you start	62		
Add the data source	62		
Make the filter customizable by the report user	69		
Exercise 8: Create a traceability document with details using RPE and JRS	72		
Before you start	72		
Add the work item details not present in JRS	72		
Exercise 9: Exploring the RPE Web UI	78		
Getting started with RPENG	78		

Αŗ	ppendix A Useful knowledge	88
	Run the report	86
	Creating the RPENG assets	83
	Exercise 10: Create and run a custom report in the RPE Web UI	83

Overview

IBM® Rational® Publishing Engine automates document generation from Rational solutions and select third-party tools. You can use Rational Publishing Engine to automate the generation of documents for ad hoc use, formal reviews, contractual obligations or regulatory compliance. Built-in capabilities extract data from a range of data sources to help reduce manual work and risk of errors.

Document publishing automation improves quality while reducing cost and risk

Rational Publishing Engine 2.0.1 Release Highlights

This latest release continues the work started with 2.0 with a focus on usability and reuse:

- 1. Improved collaboration through document sharing
- 2. Simplified installation
- 3. Group artifacts
- 4. Core capabilities update
- 5. Output fixes and improvements
- 6. Stabilize the new components

Rational Publishing Engine 2.0 Release Highlights

This major release is focused on improving usability and reuse through an all new Web UI to improve every one's experience producing high fidelity project documents

- 1. No coding knowledge or additional configuration required
- 2. Built-in library support
- 3. Report designers can quickly create reports from existing or upload new templates.
- 4. Find the assets that you want to work with by tagging, pagination, sorting or searching.

Exercises

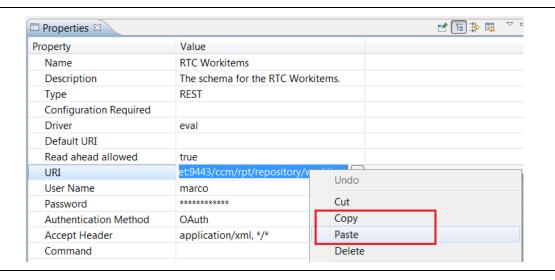
You can choose the exercises to complete based on your interests.

- 1. Exercise 1 introduces you to the RPE interface. It is a good exercise to complete if you have never used RPE before. If you are familiar with the interface, you can choose to skip this first exercise and proceed to exercise 2.
- 2. <u>Exercise 2</u> is a simple exercise for building a RPE template from a DOORS Next Generation (DNG) requirements module.
- 3. Exercise 3 introduces the concepts necessary for building traceability with RPE and the Reportable REST APIs of CLM.
- 4. Exercise 4 explores the novel way in which traceability documents can be created using RPE and Jazz Reporting Service (JRS).
- 5. Exercise 5 demonstrates some of RPE's formatting capabilities including master pages, stylesheets and document regions.
- 6. Exercise 6 builds a test document from a RQM Test Plan
- 7. Exercise 7 creates the work item list from an RTC Sprint
- 8. Exercise 8 continues the work started in exercise 6 and uses the Reportable REST APIs to add details to the backbone provided by JRS
- 9. Exercise 9 introduces the new RPE Web UI interface
- 10. Exercise 10 shows how to create and run RPE reports in the RPE Web UI

A document containing the URLs and scripts for the exercises is available in c:\label{c:labfiles} c:\label{c:label} c:\label{c:labfiles} c:\label{c:labfiles} c:\label{c:labfiles

Noteworthy

A defect in the RPE Launcher perspective in Document Studio prevents **Ctrl+C** and **Ctrl+V** keys from functioning. To copy/paste you need to use the context menu (right click).



Exercise 1: Exploring the Rational Publishing Engine Desktop Interface

Rational Publishing Engine's main artifacts are:

- Document template (*.dta) a reusable artifact
- Document specification/Report (*.dsx) connects one or more templates with data and additional formatting provided through Word stylesheets.

Note: The terms Document specification and Report are equivalent and used interchangeably in RPE.

The desktop part of Rational Publishing Engine is comprised of two parts:

- RPE Launcher create and run document specifications
- RPE Document Studio design and test document templates

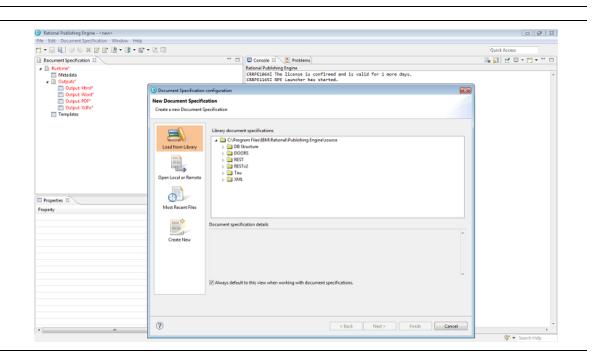
RPE Launcher

Open RPE Launcher

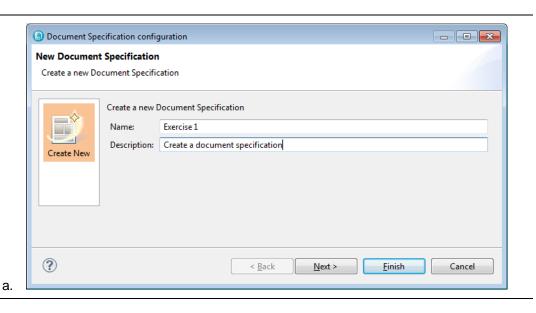
Start RPE Launcher using the Desktop icon or the Start Menu.



As RPE Launcher opens the Publish wizard starts automatically to help run existing reports or create new ones.

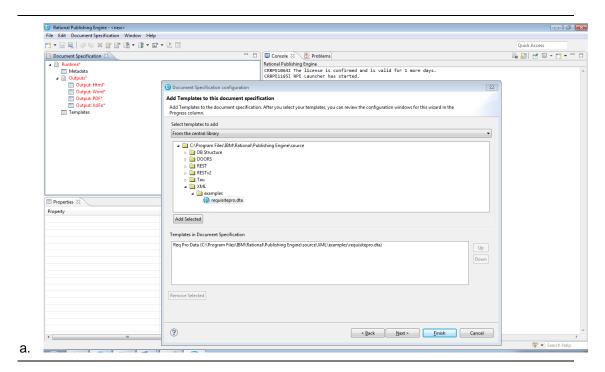


1. Use the **Create new** option and optionally provide a name and description for your report.

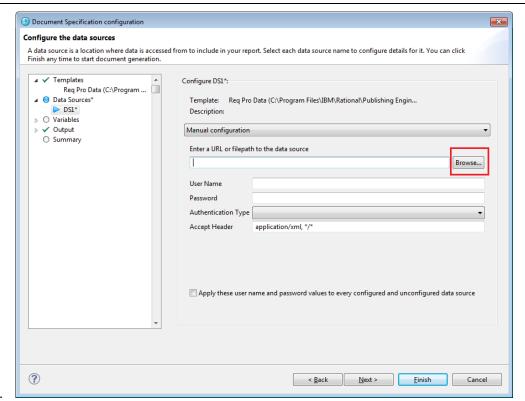


© Copyright IBM Corp. 2016 5

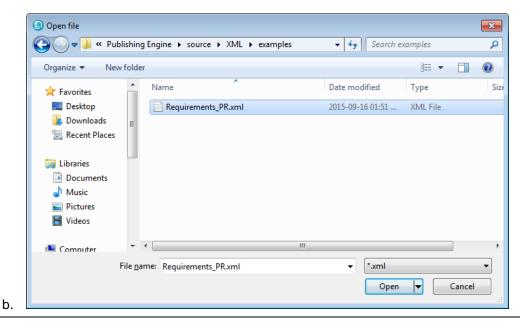
2. Click **Next** and from the list of predefined templates select requisitepro.dta from XML/examples. Click **Add selected** to add the template to the report. When you have at least 1 template added to the report the Next button becomes enabled.



- 3. In the next screen click Yes, configure and/or generate the report now.
- 4. In the next screen you must select the data to be used in the report and the output formats. For the data part click **Browse**. Navigate to the RPE installation folder and select RequirementsPR.xml.

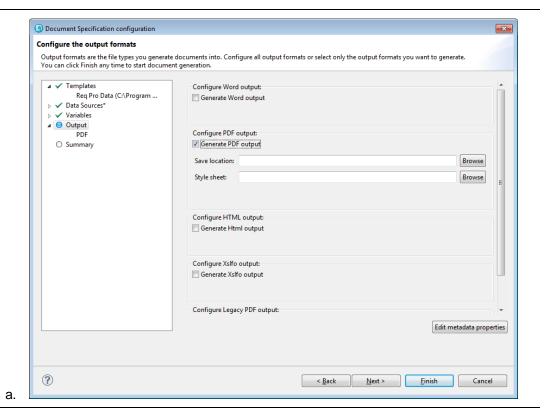


a.

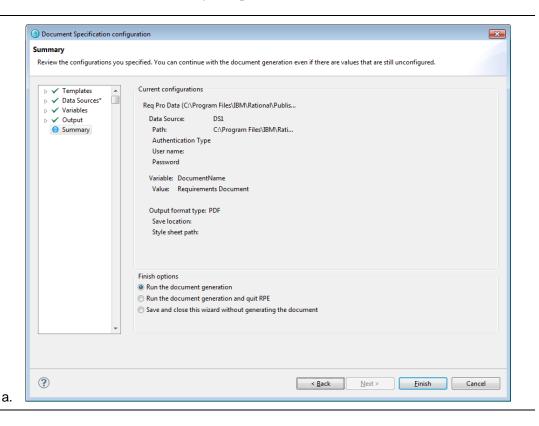


7

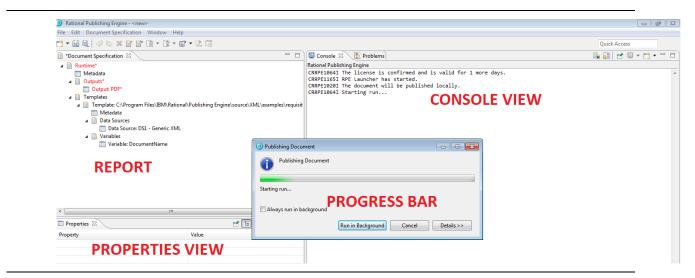
5. Select the output node and clear all options except the PDF one.



6. The last screen of the wizard summarizes your options.



7. Click **Finish** and the document generation will start. The **progress bar** helps you track what happens and the **console view** provides additional details on progress. You can explore the report and see the individual properties of the document outputs or data sources in the **properties view**.



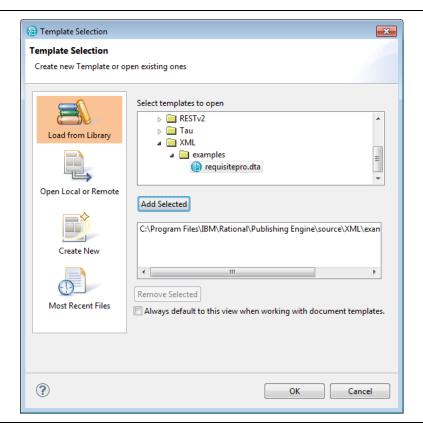
RPE Document Studio

Open Document Studio

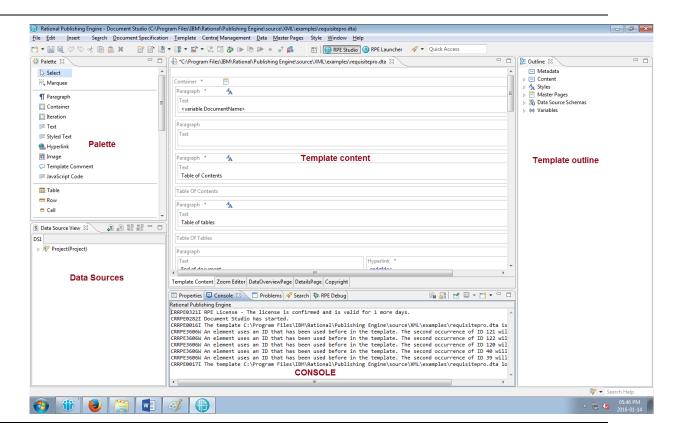
Start Document Studio using the desktop icon or the Start menu.



With the **Load from library** option selected, pick requisitepro.dta template under XML/examples.



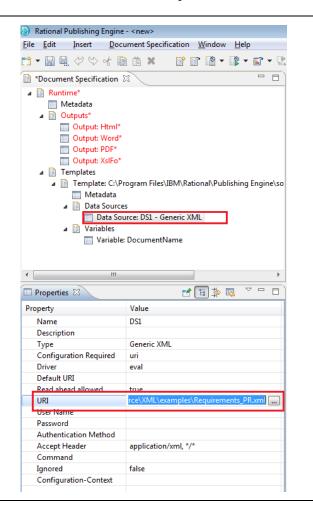
RPE Document Studio opens the template and you can observe its components as well as the content of the template. The components are:



You can test the templates you are building by producing documents directly from Document Studio. You must first switch to the Launcher perspective by using the toolbar button.



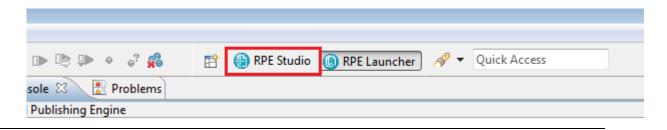
In the Launcher perspective you need to specify the data that the template uses. Click on the DS1 data source in the Document Specification view and select the Requirements_PR.xml file in the URI property.



You can run the document generation using the **Generate Document** button from the toolbar menu.



Once finished you should return to the RPE Studio perspective by using the appropriate toolbar button.



Tip: The Generate Document toolbar menu is available in the Document Studio perspective as well. Once you have configured the test report you do not need to switch back to the Launcher perspective to run the document generation.

Exercise 2: Create a requirements document

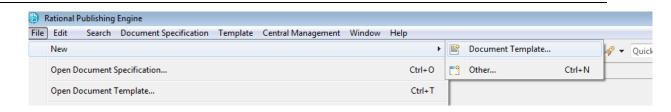
In this exercise you will learn how to create a document from IBM Rational DOORS Next Generation and how to follow links within the same product.

Create a new Document Template

Open Document Studio.

Note: If Document Studio is already running you should close any open templates. If you have opened one of the Out Of the Box templates you should choose to not save the changes.

Create a new template using the **File** menu option in Document Studio.



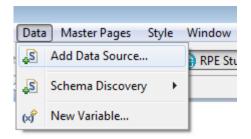
Tip: You should provide a name and a description for your template to document it. The name is also used by RPE the first time the template is saved.

Add a data source for the DNG Modules

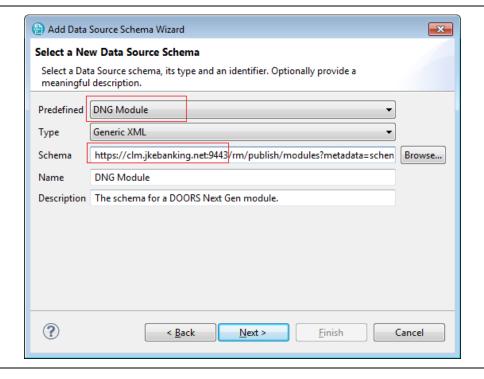
In order to use data in a report you must first "tell" RPE what the data looks like. You do this by adding the data source schema to the template.

Note: RPE uses the schema of the data when building the template. The actual data is used only when the document is generated. This makes the template reusable across multiple data instances.

Each product that supports RPE offers a schema of its data as part of its Reportable API. RPE keeps a list of the most commonly used API for easy access in the **Add Data Source** wizard under the **Data** menu.



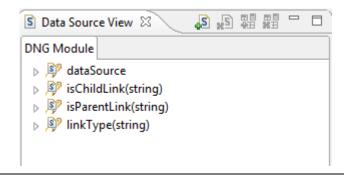
On the second page of the wizard select **DNG Module** from the list of predefined data sources. In the **Schema** field replace https://[host]:[port] with https://clm.jkebanking.net:9443 like this:



When prompted to authenticate use the credentials marco/marco and OAuth for the authentication type.



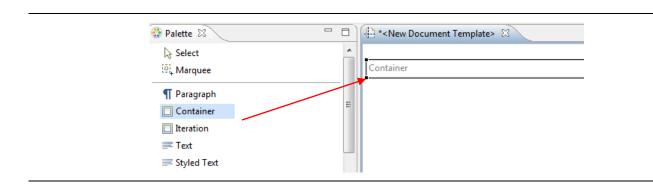
Finish the wizard and the data source is added to your template. You can see the data source in the Data Source view in Document Studio.



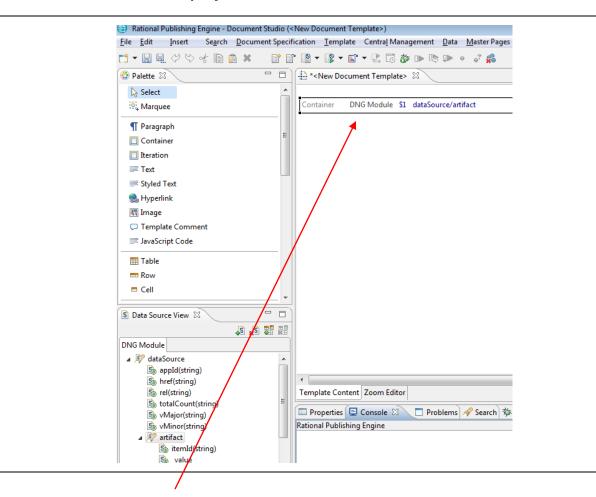
Add elements to print the module name and description

From the palette, select the container element and add it to the template.

Tip: Containers are used to organize your template, especially for holding data queries. Use them as often as needed as they have no impact on the output.



In the data source view expand the **datasource/artifact** node in the tree. Drag the **artifact** node from the data source view onto the container you just created.



Inside the container, create a paragraph and on the paragraph drag the **title** element under **artifact** from the data source view. When prompted by RPE, select that you want the element to be used as value.



Your template should look like this now.



Save the template

Save your template in c:\labfiles\reporting\Exercise 2.dta

Tip: Save often.

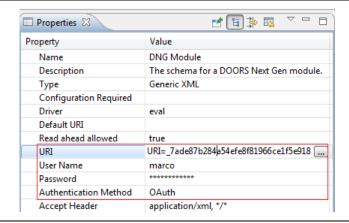
Configure, run and save the report

Configure the test report

The template is ready to be tested and for this you need the data source information.

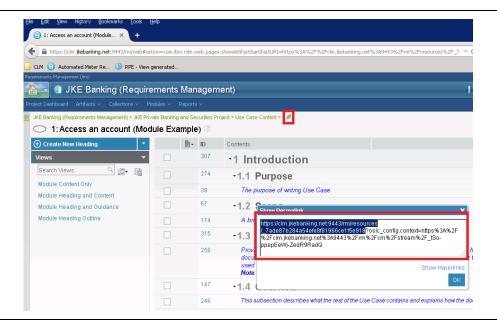
- 1. Save the template.
- 2. Switch to the RPE Launcher perspective.
- 3. Select the DNG Module in the Report tree.
- 4. Introduce the URL of the "Access an account" module in the **URI** field in the Property view. See below for how to obtain the URL.
- 5. Use the same credentials and OAuth method you used when adding the schema to the template.

Tip: Test often especially after adding data queries and properties.



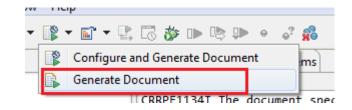
Each point product documents how to obtain the Reportable REST URL of its artifacts. To obtain the URI of a DNG module you need to do the following:

- 1. Log in to DNG
- 2. Open the "Access an account" module example
- 3. Copy the link shown in the pop-up dialog up to the **?oslc_config** part. The URL should look like https://clm.jkebanking.net:9443/rm/resources/_7ade87b284a54efe8f81966ce1f5e918
- 4. Replace **resources** with **publish/test?resourceURI** to obtain a new URL that looks like this: https://clm.jkebanking.net:9443/rm/publish/modules?resourceURI=_7ade87b284a54efe8f81966c e1f5e918



Save and test the report

- 1. Save the report as Exercise 2.dsx in c:\labfiles\reporting
- 2. Perform the document generation using the **Generate document** toolbar button and observe the output.

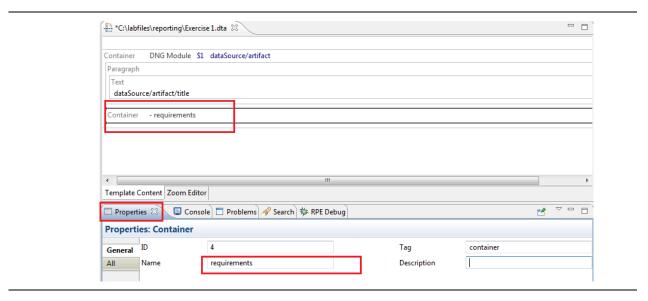


Add elements to iterate requirement

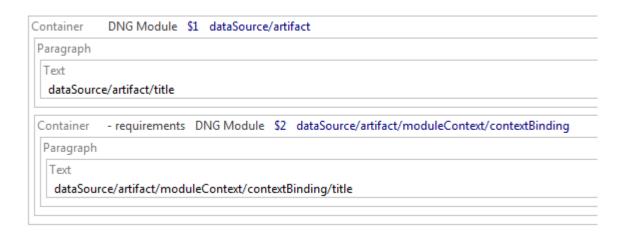
With the module title in the document it's time to add the requirements.

1. Create a new container under the root one and name it **requirements**. Name the container in the properties view under the General section.

Tip: Names are useful as they help document the template.



- 2. Add the **data/source/module/contextBinding** query on the requirements container.
- 3. Add a paragraph inside the container
- 4. Drag **title** from under the **contextBinding** node in the data source view on the newly created paragraph.



Test the report

Run the report again and observe the results. The output document contains now the title of each requirement in addition to the module title.

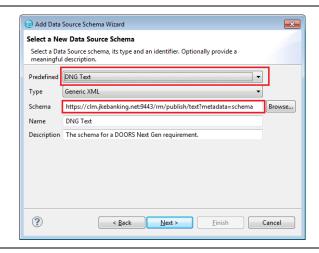
Add requirement details

The requirement information present in the DNG Module data source is usually not enough. Additional details, like the rich text of the requirement, are available at the URLs stored in the module context binding.

Add a data source for requirement details

To access the requirement details you need to first add a new data source of type DNG Text.

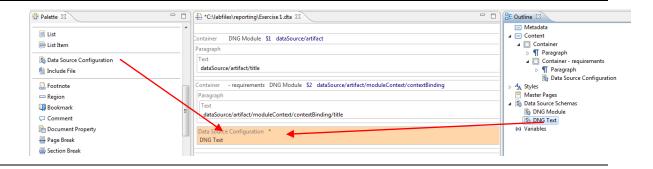
- 1. Open the Add data source wizard
- 2. Select the **DNG Text** predefined data source.
- 3. Modify the URL to point to the clm.jkebanking.net server.



Dynamically configure the DNG Text data source

Unlike the DNG Module data source, the DNG Text one does not have to be configured statically by the report user. Instead this data source will be automatically configured by RPE using information from the DNG Module data source. The automatic configuration is done using the Dynamic Data Source element. The term *DDS* will be used in the rest of the lab to denote a Dynamic Data Source element.

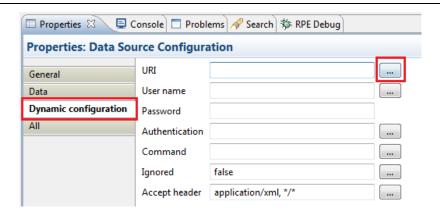
- 1. Add a Dynamic Data Source element from the palette
- 2. Drag the **DNG Text** data source from the outline view on the DDS element.



The DDS is used to access the details for each requirement (one at a time) using the requirement URL stored in the module.

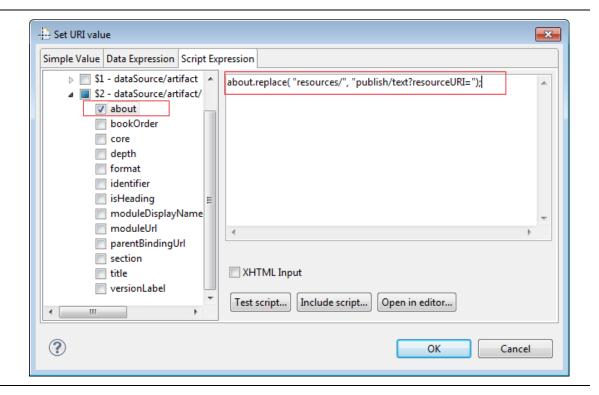
© Copyright IBM Corp. 2016 23

- 1. Select the DDS element
- 2. In the Properties view, click the URI button under the **Dynamic configuration** tab.
- 3. Click the ... button to open the value editor



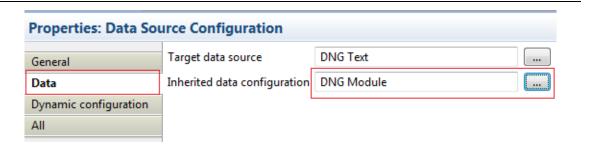
Note: The link for the requirement details is stored in the "about" property of the contextBinding. However that URL cannot be used directly. We need to derive a different URL from it, which is the URL of the Reportable REST representation of the requirement. This manipulation resumes at replacing a particular segment of the URL as shown below.

- 4. In the editor select the **Script Expression** tab and enter the script to convert the URL
 - a. Requirement URL: https://clm.jkebanining.net:9443/rm/resources/_d29b5a3aadb242ccae6595568eced24c
 - b. Reportable REST URL: https://clm.jkebanining.net:9443/rm/publish/text?resourceURI=_d29b5a3aadb242ccae6 595568eced24c



The second part of configuring the DNG Text is the authentication. We could have the report user enter the credentials in the report but that is not needed since the DNG Module credentials can be reused.

- 1. Select the DDS element.
- 2. Click the **Inherited data configuration** button in the **Data** tab from the Properties view.
- 3. Select **DNG Module** from the list.

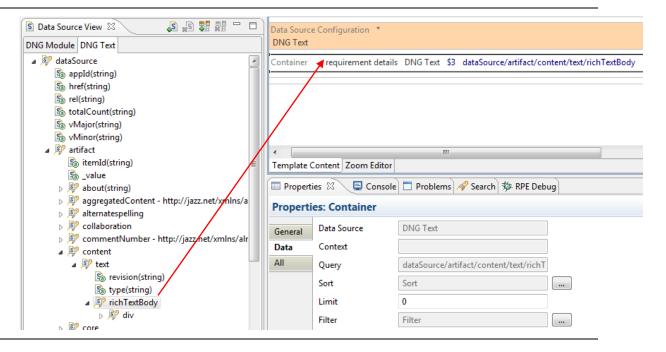


© Copyright IBM Corp. 2016 25

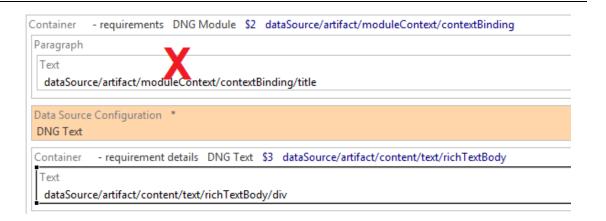
Add elements to print the requirement text

The DNG Text data source is now fully configured and we can use it in the template.

- 1. Add a container named requirement details immediately after the DDS
- 2. Drag on it the dataSource/artifact/content query from the DNG Text data source.

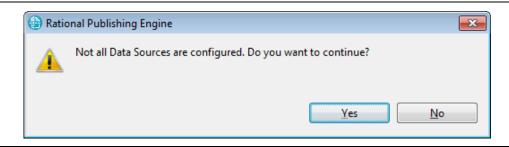


- 3. Add a text element inside the requirement details.
- 4. Drag the **div** property on it and select to use it as a value when prompted by RPE.
- 5. Delete the paragraph containing the **contextBinding/title** as that information is now redundant



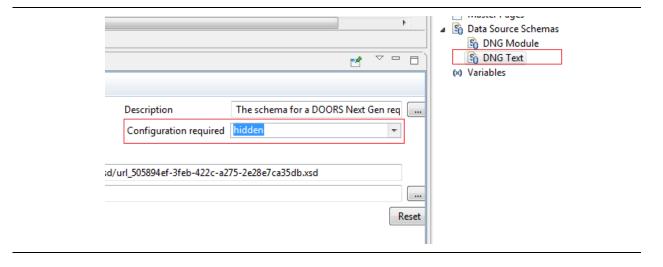
Hide dynamically configured data sources

If you run the report now you will notice a message from RPE warning that not all data sources are configured. The message is raised as there is no configuration information provided for the DNG Text data source in the report.



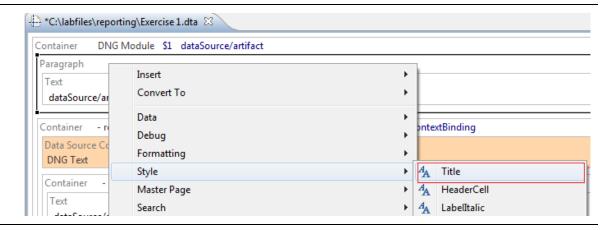
Since the entire configuration is done dynamically in the template you can safely ignore the error when prompted by RPE. The better solution is to hide altogether the data source from the report user.

- 1. Select the **DNG Text** data source in the outline view
- 2. Select **hidden** in the **Configuration required** property.



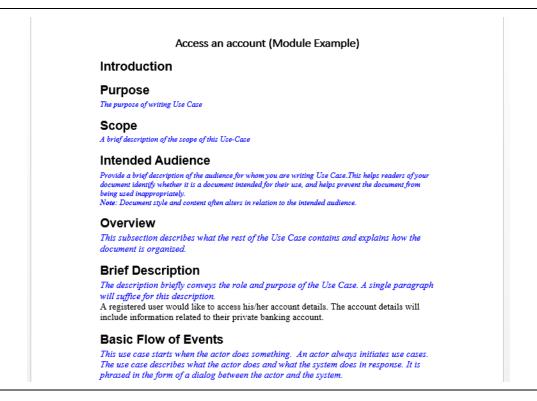
Make the module title stand out in the document

- 1. Right-click in the paragraph holding the module title.
- 2. Select **Title** from the style sub-menu.



Result

Test your report again and observe the results.



Exercise 3: Creating a traceability document with RPE

In this exercise we will create a document listing all the test cases and their associated requirements and work items. The purpose is to get familiar with following traceability links using the OSLC and Reportable REST APIs of the point products.

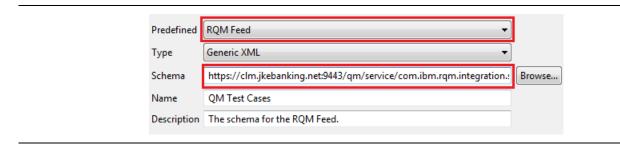
Prepare to start

If Document Studio is not open or you do not have the artifacts from the previous exercises loaded please do the following:

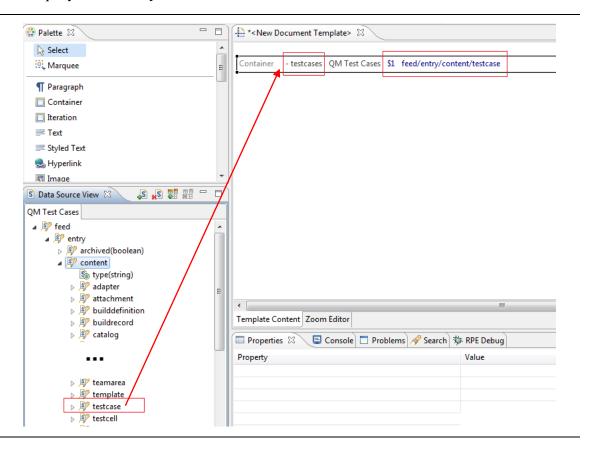
- 1. Open Document Studio.
- 2. Close all templates and documents.
- 3. Create a new document template.
- 4. Log in CLM 6.0.1

Add test data

The document will start with the test data, more precisely the test cases. Add the RQM Feed data source using the **Add Data Source** wizard you have used before. Name the data source **QM Test Cases**.



- 1. Create a new container named **testcases**.
- 2. Set its query to **feed/entry/content/testcase**.



With the query in place you can now print the test case's properties. Do the following:

- 1. Add a paragraph called **test case details**.
- 2. Inside the **test case details** paragraph create 3 text elements:
 - a. Drag the **webid** property from the Data Source View on the first text element and select to use it as value.
 - b. Drag the **title** property on the third text element and select to use it as value.
 - c. Double-click the new text element and set its content to a single white space.



Save the template

Save the template in c:\labfiles\reporting as Exercise 3.dta.

Configure, run and save the report

Configure the report

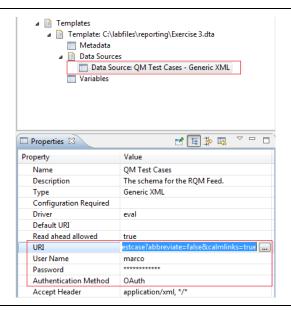
Switch to the RPE Launcher perspective using the toolbar icon. Once in the Launcher perspective select the QM Test Cases data source and introduce the following in the data source:

URI:

https://clm.jkebanking.net:9443/qm/service/com.ibm.rqm.integration.service.IIntegrationService/resources/JKE+Banking+%28Quality+Management%29/testcase?abbreviate=false&calmlinks=true

2. User name: *marco*3. Password: *marco*

4. Authentication method: *OAuth*



Run the report

Run the report using the **Generate Document** button from the toolbar. The output should contain the id and title of all the test cases from the QM JKE Banking project

Adding the traceability information

The QM data contains the links to the related requirements and work items. The next steps are to follow those links and retrieve the information we need for the document.

Adding the requirements information

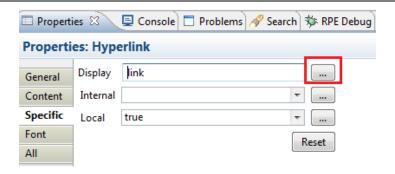
For requirements we will not actually follow the traceability link. We will only print a hyperlink to the DNG Requirement using the requirement's title as the hyperlink text.

Tip: You can easily enhance this and include the actual DNG Requirement text using the same principle used in Exercise 2 when following links from DNG Modules to DNG Requirements.

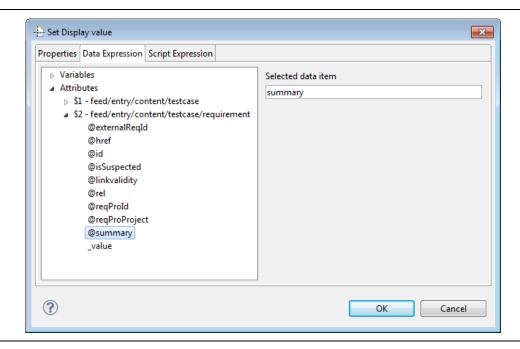
- 1. Under the testcases container create a new container called **linked requirements**.
- 2. Drag feed/entry/content/testcase/requirement on top of the linked requirements container.
- 3. Create a paragraph inside **linked requirements**.
- 4. Create a text element inside the paragraph and set its content to **Requirement:**
- 5. Apply italic formatting to the text using the context menu or pressing Ctrl+I.
- 6. Create a hyperlink inside the paragraph.
- 7. Assign the **internal hyperlink** style to the hyperlink using the context menu or by dragging the mentioned style from the Outline view onto the hyperlink.
- 8. Drag the **href** attribute from the data source view on top of the hyperlink.



9. Select the hyperlink and expand the **Specific** tab in the Properties view.



10. Open the editor using the indicated button and assign the **summary** attribute as the value to be used for the hyperlink's display.



Run the report

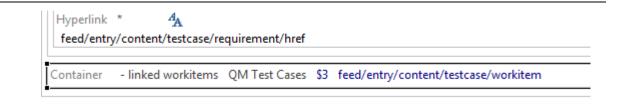
Run the report using the **Generate Document** button from the toolbar.

Adding the work item information

We will follow links to the work items and for each we will print the title, type, state and description.

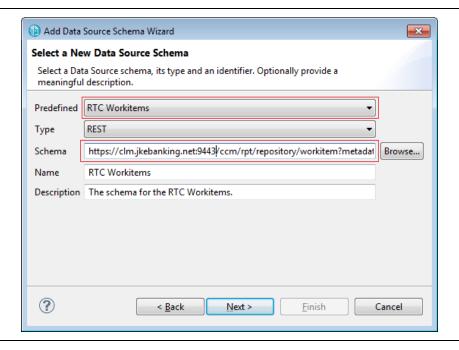
Follow the traceability links to RTC

- 1. Under the testcases container create a new container called **linked workitems**.
- 2. Drag feed/entry/content/testcase/workitem on top of linked workitems.

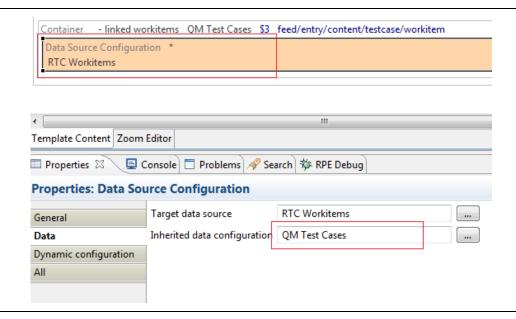


© Copyright IBM Corp. 2016

3. Add a new data source to the report called **RTC Workitems**. Use the appropriate entry from the predefined list in the **Add Data Source** menu. We will use this data source to follow the links to Rational Team Concert (RTC).

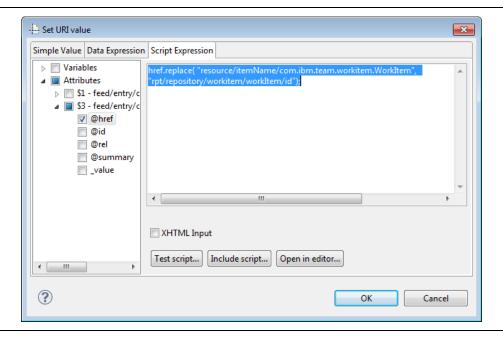


- 4. Create a Dynamic Data Source inside the **linked workitems** query.
- 5. Drag the **RTC Workitems** data source from the outline view on the DDS.
- 6. Select the DDS and go to the **Data** tab.
- 7. Set the value of the **Inherited data configuration** to **QM Test cases**.



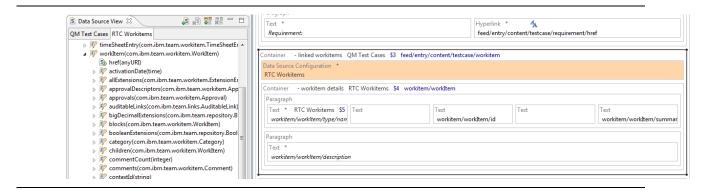
- 8. The URI of the linked work item is in the **href** property but that URL cannot be used directly as it is the OSLC URL of the work item. However we can use that URL to derive the reportable REST URL.
 - a. Select **Dynamic configuration**.
 - b. Click on the ... button of the URI property
 - c. Switch to the script expression tab.
 - d. Drag the @href property from under \$3 in the editing area.
 - e. Set the script expression to be the one printed below.

href.replace("resource/itemName/com.ibm.team.workitem.WorkItem", "rpt/repository/workitem/workItem/id");



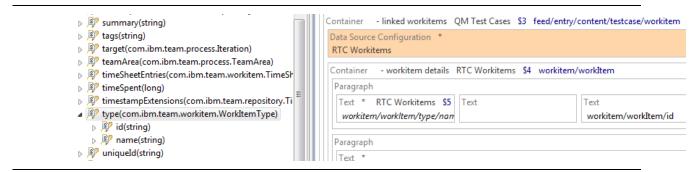
Print the work item information

We want to print the work item type, id, title and description.



© Copyright IBM Corp. 2016

- 1. Create a new container after the DDS but still inside the **linked workitems** container. Name it **workitem details**.
- 2. Drag the **workItem** query from the RTC Workitems data source onto the new container.
- 3. Create a paragraph inside the workitem details container
- 4. Create 4 text elements inside the paragraph
 - a. Drag the **type** element from the RTC Workitems data source on the first text element. RPE will create a query for it on the text element.
 - b. Drag the **name** property of the type element on the **same** text element. RPE will use the property as the value of the text element.



- c. Set the content of the second text element to a single empty space.
- d. Drag the **webid** property of the workitem on the 3rd text element.
- e. Set the content of the 4th text element to a single empty space.
- f. Drag the **summary** property of the workitem on the 5th text element
- 5. Create another paragraph inside workitem details.
- 6. Drag the **description** property of the workitem on top of the new paragraph. Select to use it as a value and RPE will create a new text element with this property set as its value.

Create an empty line between test cases

1. Create a paragraph inside the **testcases** container.

Note: Make sure this paragraph is not inside the **linked workitems** container.

2. Inside the new paragraph, add a text element with the content set to a single white space.

Final test

Running the report from the **Generate document** toolbar icon will produce a document looking like this.

4 Process hard copy requests

Requirement: Requests sent in form of hard copy mail Story 58 Requests sent in form of hard copy mail Requests sent in form of hard copy mail.

5 Process email requests

Requirement: Requests sent in form of email Story 59 Requests sent in form of email Requests sent in form of email.

6 Organization must identify how much money is desired Requirement: Organization must identify how much money is desired Story 60 Organization must identify how much money is desired Organization must identify how much money is desired.

7 Allocate dividends by amount and frequency
Requirement: Allocate dividends by amount and frequency
Story 61 Allocate dividends by amount and frequency
Allocate dividends by amount and frequency to complete template: Found under Business
Recovery Matters Elaborated Stories.

Exercise 4: Creating a traceability document with RPE and JRS

In this exercise we will create a document listing all the test cases and their associated requirements and work items. The purpose is to get familiar with following traceability links using JRS as the data provider for RPE and to compare the ease of use of querying and aggregating traceability data in JRS versus doing the same using the Reportable REST API.

Note: As you will notice in the exercise, JRS cannot provide all the details needed for the document as rich text and images are typically unavailable. So in practice JRS is used to build the backbone of the document (the linked resources) while the Reportable REST API is used as needed to bring in the details.

Prepare to start

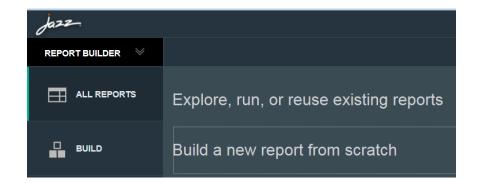
If Document Studio is not open or you do not have the artifacts from the previous exercises loaded please do the following:

- 1. Open Document Studio.
- 2. Close all templates and documents.
- 3. Create a new document template.
- 4. Log in to JRS at https://clm.jkebanking.net:9443/rs.

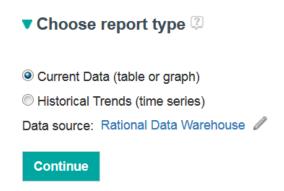
Create the query

In this step of the exercise we build the data for the RPE document using JRS. If you already have a JRS query that contains traceability information you can use it.

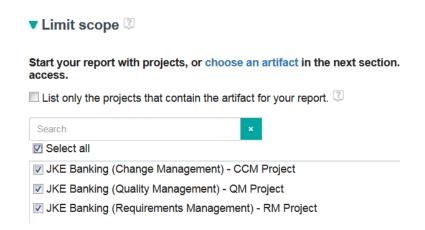
1. Create a new JRS report.



2. Select Current data.

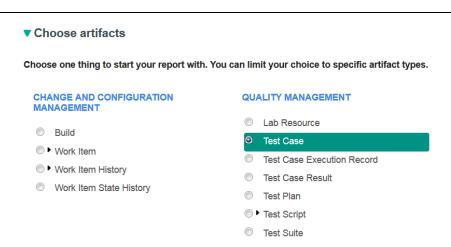


3. Select all projects in the Limit Scope section.

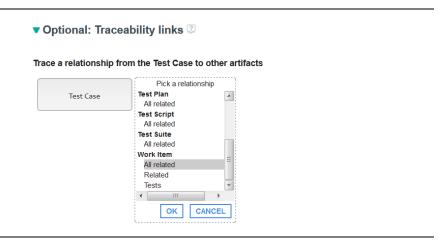


© Copyright IBM Corp. 2016

4. Use **Test Case** as the starting resource.



5. Add the traceability link to work items.



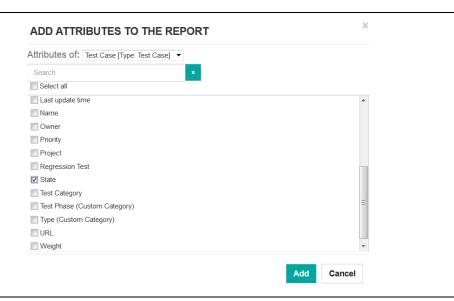
6. Set the traceability link as **Optional** to ensure all test cases are printed, regardless if they have associated work items or not.



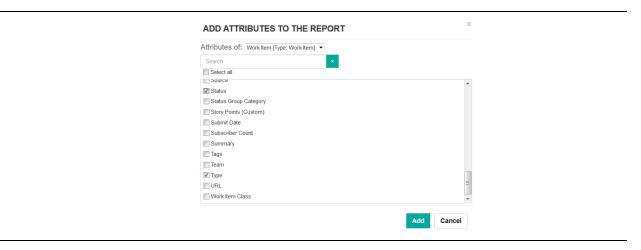


7. Click **Continue** until you are taken to the Format Data page.

- 8. Make the following changes in the selected columns:
 - a. Delete the **Project** field.
 - b. Add the **State** property for the test cases.

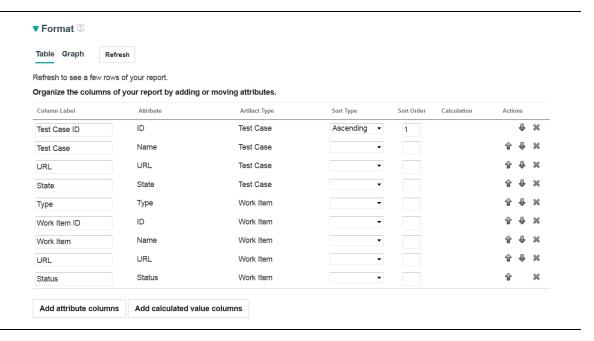


c. Add the **Type** and **Status** properties for Work Items.

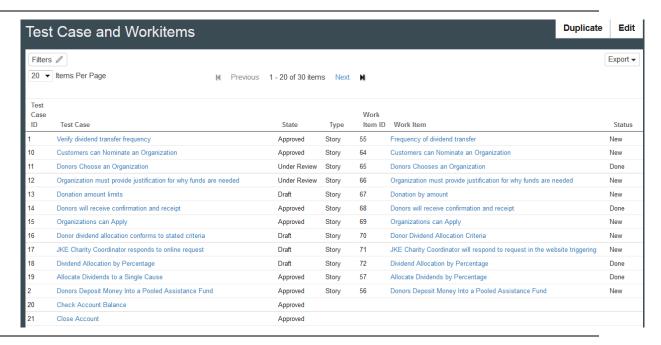


d. Set the Sort Type for the Test Case ID to Ascending to have the report data sorted.

e. Use the up and down arrows to obtain the following attribute list.



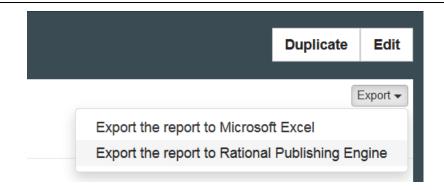
- 9. Click Continue.
- 10. Name your report **Test Case and Workitems** and save it.
- 11. Go to **My reports**, select your newly created report and run it. Observe the results.



Export the query and use it with RPE

Export to RPE

12. From the report page select the **Export to Rational Publishing Engine** option.



13. Download the .dta and .dsx files from the page that JRS opens. Save them in c:\labfiles\reporting using the names given by JRS.

Create a new Rational Publishing Engine document

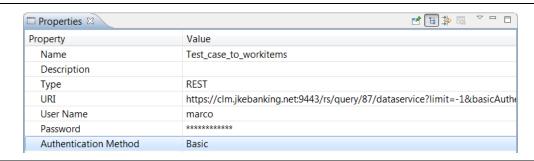
Two files are required: a template (DTA) file and a document specification (DSX) file. The DS

- 1. Download these files into the same folder:
 - o Template (DTA) file
 - o Document specification (DSX) file
- 2. Open the DSX file for your "Test Case and Workitems" report.
- 3. Configure the data source with a user name and password.
- 4. Generate the document.

Run the report in RPE

- 14. In Document Studio open the template:
 - a. Select the option to upgrade the template to RPE Template version 2.
- 15. Switch to the RPE Launcher perspective.
- 16. Open the document specification.

17. Select the data source and in the Properties view provide your credentials while setting the authentication method to **Basic**.

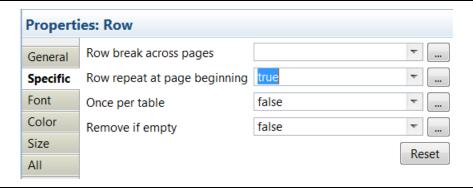


18. Save and run the RPE report.

Modify the template

If you compare this with the process you followed in Exercise 3 the simplification of the data querying is obvious. Additionally by looking at the schema in the .dta file you will notice that the structure is simpler and friendlier for document generation. With the complexity of the data query behind us you can focus on formatting.

- 1. Switch to the Document Studio perspective
- 2. Select the table header row.
- 3. In the Properties view set the **Row repeat at page beginning** to **true**.



- 4. Add color coding to the Test Case state and to the Work Item status as shown in exercise 4.
- 5. Set the Work Item type property to italic.
- 6. Drag the **InternalHyperlink** style from the Outline view to the two Hyperlink elements generated by RPE.

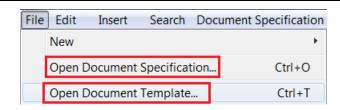
Exercise 5: Formatting a document

In this exercise we will go through some techniques for formatting documents. The template from Exercise 2 is used as the starting point but you can use any of the templates from the previous exercises.

Prepare to start

If Document Studio is not open or you do not have the artifacts from the previous exercises loaded please do the following:

- 1. Open Document Studio.
- 2. Close all templates and documents.
- 3. Open Exercise 2.dta and Exercise 2.dsx.



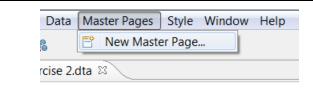
- 4. (optional) Log into CLM 6.0.1
- 5. Test the report to verify everything is running and note the appearance of the document.

Add headers and footers to the document

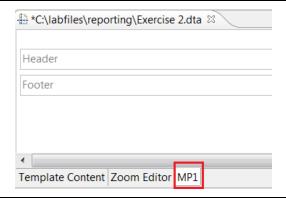
Most formal documents have headers and footers. In RPE the headers and footers are introduced through Master Pages.

Create the master page

1. Create a master page and name it **MP1**.

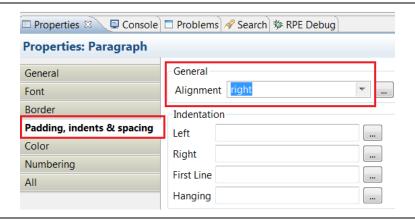


2. Document Studio automatically selects the Master Page tab in the editor.



Create the header

- 3. Create a paragraph in the header.
- 4. Create a text element and set its value to **Requirements Document** (or any other value you want).
- 5. Right-align the paragraph content using the Alignment property in the **Padding, indents & spacing** section of the paragraph's properties.

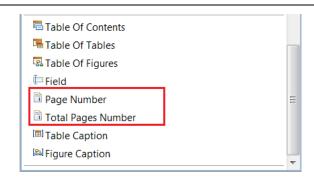


Note: RPE is not a What You See Is What You Get editor (WYSIWYG) so the paragraph display in Document Studio will not change.



Create the footer

- 6. Create a paragraph in the footer.
- 7. From the palette add in this order a **Page Number**, a text element and a **Total Page Number** element



- 8. Set the content of the text element to "/".
- 9. Set the paragraph alignment to **Center**.



Apply the master page

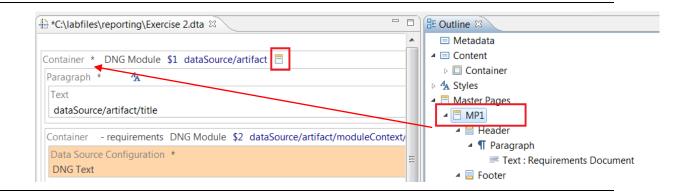
In order for the header and footer to appear in the output document, the master page must be applied in the document template.

10. Switch to the template content by clicking the Template Content tab in the editor.



11. Drag the master page from the Outline view on the root container in the template. Notice the new symbol that gets added to the container element.

© Copyright IBM Corp. 2016

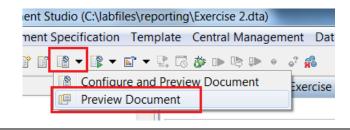


Test

Save the template and run the report now to see the document title and the page numbers.



Tip: Use the preview function for quickly testing changes in your report.



Use a style sheet with the report

The format of a document can be defined in the template through formatting properties but the better option is through external stylesheets. To apply a stylesheet to your report do the following:

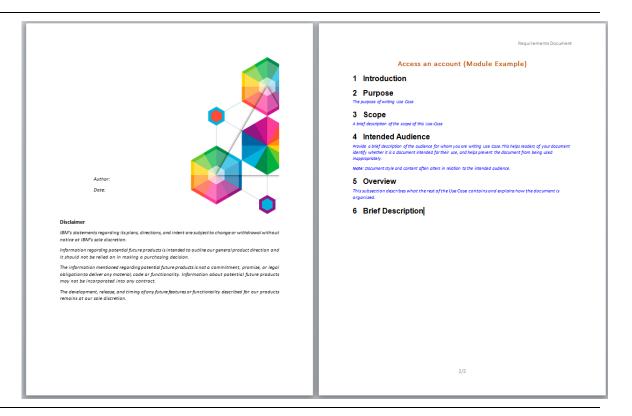
- 1. Switch to the RPE Launcher perspective.
- 2. Select the Word output.
- 3. Modify the stylesheet property and set its value to c:\labfiles\reporting\frontpage.dotx using the ... button.



4. Repeat for the PDF output.

Tip: You can use Microsoft Word documents as stylesheets for both the Word and PDF output of RPE.

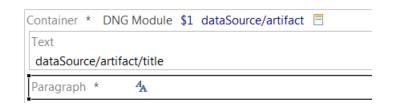
- 5. Run the report and observe the Word output. Notice the changes from your previous run:
 - a. Additional front page content.
 - b. The headings are numbered.
 - c. The title of the document is in orange.



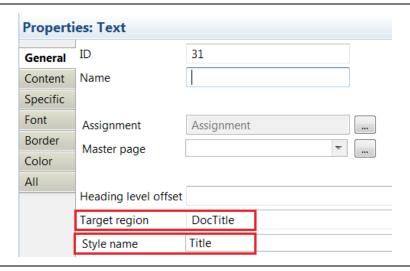
Using document regions

You can direct the output of RPE elements to a specific region in the output using bookmarks and the **target region** property. The front page has a few regions for author, title and date that we can use.

- 1. Select the title text element in the template.
- 2. Drag it outside the paragraph but still in the root container.



- 3. Delete the paragraph.
- 4. Apply the **Title** style to the text element.
- 5. Set the value of the **Target Region** property to **DocTitle**. DocTitle is a bookmark in the frontpage.dotx file.

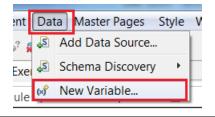


6. Run or preview the report again and observe the new layout. The title is now inserted in the content of the stylesheet.

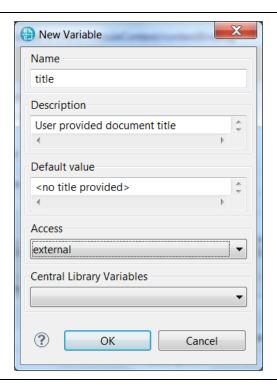
Using variables to customize the document

To make a template truly reusable, it must be possible for parts of it to change depending on external factors. RPE template variables are the main mechanism that adds flexibility to the templates. You can use variables to input data in the document or to take decisions at runtime, such as following links to related resources. In this example we will use a variable to customize the title that appears in the header of the document.

1. Open the **New variable** dialog.



2. Create an external variable named title.



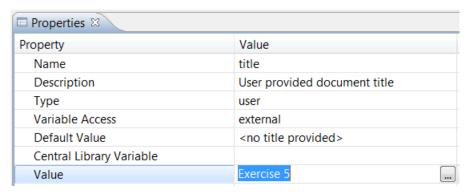
- 3. (Optional) Provide a description to the variable to help users know how to use it.
- 4. (Optional) Provide a default value to the variable.
- 5. Switch the editor to the MP1 master page.
- 6. Drag the variable onto the text element containing the static text.



7. Save your template.

Note: If you do not save the template the new variable will not appear in the report view in the RPE Launcher perspective.

- 8. Switch to the RPE Launcher perspective.
- 9. Select the variable in the report view.
- 10. Provide a value for the title variable.



- 11. Save the report as Exercise 5.dsx in C:\labfiles\reporting.
- 12. Run or preview your report.

Exercise 5

- 1 Introduction
- 2 Purpose

The purpose of writing Use Case

3 Scope

A brief description of the scope of this Use-Case

Adding a table of contents

The final touch is to add a table of contents to the document.

- 1. Add a table of contents element inside the root container, right before the title text element.
- 2. Save the template and run or preview it. RPE automatically populates the table of contents using the DNG headings from the document.

Exercise 6: Create a test plan document from RQM

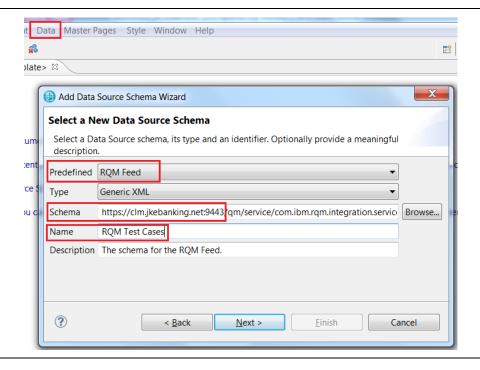
Before you start

- 5. Open Document Studio or if already open close all templates and documents
- 6. Create a new document template
- 7. Log in CLM 6.0.1

Add the data source

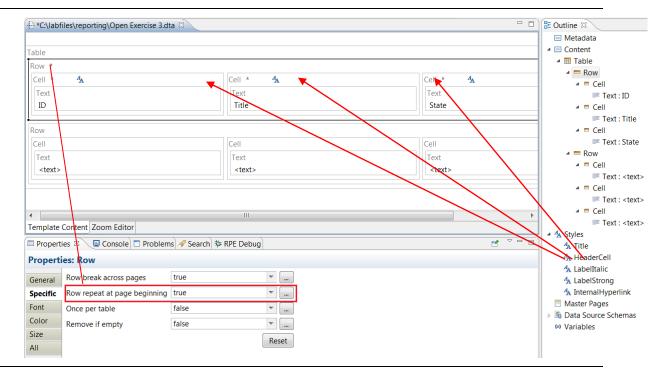
Add a data source for the QM Test cases

- 1. Open the Add Data Source wizard and select RQM Feed in the list of predefined data sources
- 2. In the **Schema** field replace https://clm.jkebanking.net:9443
- 3. Authenticate using the marco/marco credentials and "OAuth"
- 4. Name your data source "RQM Test Cases"



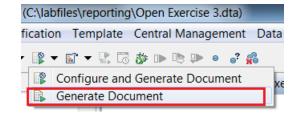
Create the document frame

- 1. Create a table with 2 rows and 2 columns
- 2. Create a text element in each cell
- 3. The content of the first row is the header of the table. Enter "ID", "Title" and "State" in the 3 texts
- 4. Drag the **Header Cell** style from the outline onto the cells from the first row
- 5. Set the **Row Repeat at page beginning** property of the first row to **true** to create a header row



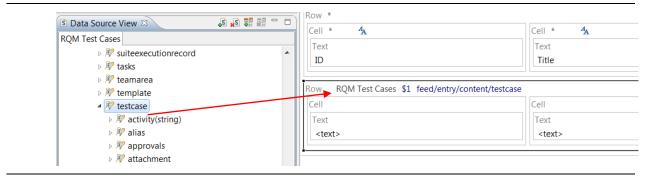
Save and test

- 6. Save the template as Exercise 6.dta in c:\labfiles\reporting
- 7. Run the report using the toolbar icon. Ignore the warning shown by RPE.



Add the test case information

- 8. In the data source view expand the **feed**, **entry** and **content**
- 9. Select the **testcase** node and drag it onto the second row of the table



- 10. Add the properties of the **testcase** in the second row.
 - a. Drag webid on the first cell's text. Select Use as value when prompted by RPE



- b. Drag title on the second cell's text
- c. Drag state on the third cell's text. You will not have the option to use it as a value
- d. Drag the "value" property under "state" onto the text



Note: A text element can have both a query and a value.

11.

RQM Test Case Feed URL

In order to run the report you need to configure the report with the URL of the data. The QM Reportable REST API is documented on https://jazz.net/wiki/bin/view/Main/RqmApi

1. The URL that gives all the test cases in the JKE Banking project is:

https://clm.jkebanking.net:9443/qm/service/com.ibm.rqm.integration.service.IIntegrationService/resources/JKE+Banking+(Quality+Management)/testcase

2. The data returned accessing the above URL does not include the details of the testcases, only their title and href. To get the details you need to add the **abbreviate=false** query parameter to the request. The **calmlinks=true** parameter is required if you need link information.

https://clm.jkebanking.net:9443/qm/service/com.ibm.rqm.integration.service.IIntegrationService/resources/JKE+Banking+(Quality+Management)/testcase?abbreviate=false&calmlinks=true

3. To include only the test cases belonging to a specific test plan a special syntax is required.

https://clm.jkebanking.net:9443/qm/service/com.ibm.rqm.integration.service.IIntegrationService/resources/JKE+Banking+(Quality+Management)/testcase?fields=feed/entry/content/testcase/(*|testplan[@href="<testplanhref>"])

To get the test cases for the **User Acceptance** test plan the URL is:

https://clm.jkebanking.net:9443/qm/service/com.ibm.rqm.integration.service.IIntegrationService/resources/JKE+Banking+(Quality+Management)/testcase?fields=feed/entry/content/testcase/(*|testplan[@href="https://clm.jkebanking.net:9443/qm/service/com.ibm.rqm.integration.service.IIntegrationService/resources/JKE+Banking+(Quality+Management)/testplan/urn:com.ibm.rqm:testplan:2"])

Test the report

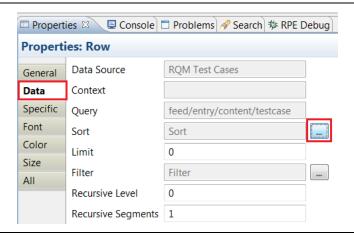
- 1. Switch to the RPE Launcher perspective
- 2. Select the RQM Test Cases data source in the report view
- 3. Enter the URL #3 from the above list in test case in the URI property
- 4. Run the report

Tip: Use the **Preview** function for fast document generating when building and testing the template

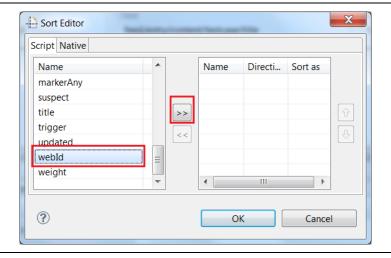
Details

Sort the data

- 1. Activate the properties view
- 2. Select the 2nd row of the table, the one with the test case query on it
- 3. Select the data tab in the properties view and open the sort editor using the "..." button



4. From the left pane select the **WebId** property and add it to the sort list using the ">>" button

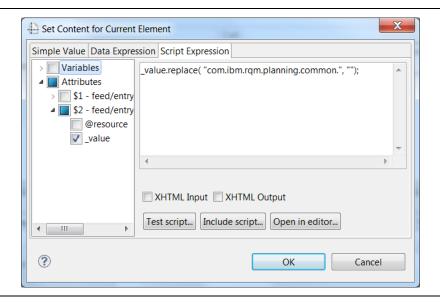


5. Save the template

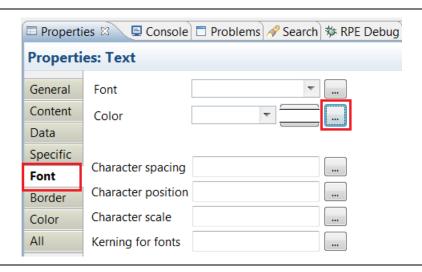
Format state information

With the data in place it is now time to make the document look nice

- 1. Double click the text element holding the state information to edit its content
- 2. Switch to the **Script Expression** tab of the editor
- 3. Drag the _value property from the left hand side tree and add the text as shown in the image below. This will shorten the value for state shown in the document

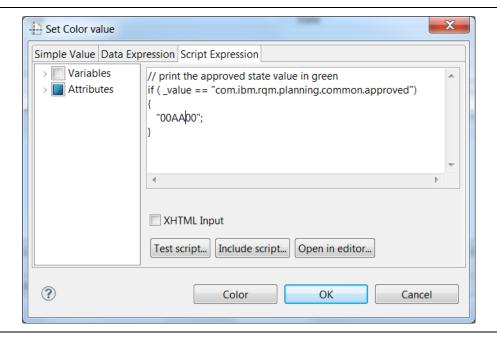


- 4. Close the editor and in the Properties View for the same text element select the **Font** tab
- 5. Open the Color editor using the "..." button



6. In the value editor switch to the **Script Expression** tab

7. Select the **_value** property from the left hand side tree and enter the following code

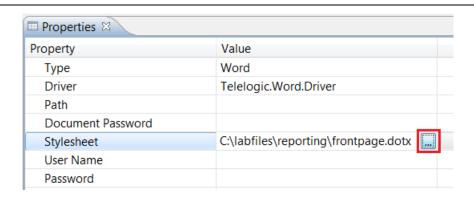


8. Save the template

Apply a stylesheet

The final touch for creating a good looking document is to apply a stylesheet.

- 1. Switch to the RPE Launcher perspective
- 2. Select the **Word** output
- 3. In the **Stylesheet** property select the frontpage.dotx from C:\labfiles\reporting using the "..." button



4. Repeat step 2 and 3 for the **PDF** output

5. Save the document specification

6. Run the document



Disclaimer

IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal withou notice at IBM's sole discretion.

Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.

The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.

The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Test Plan Summary

ID	Title	State
2	Donors Deposit Money Into a Pooled Assistance Fund	approved
26	Consistent display of currency	approved
3	Allocate Dividends by Percentage	new
4	Process hard copy requests	approved
5	Process email requests	approved
7	Allocate dividends by amount and frequency	new
8	Organizations may apply with an initial request	approved
9	Donors Can Choose to Support an Organization	underreviev
10	Customers can Nominate an Organization	approved
11	Donors Choose an Organization	underreviev
13	Donation amount limits	new
14	Donors will receive confirmation and receipt	approved
16	Donor dividend allocation conforms to stated criteria	new
18	Dividend Allocation by Percentage	new
19	Allocate Dividends to a Single Cause	approved

Page 2 of 2

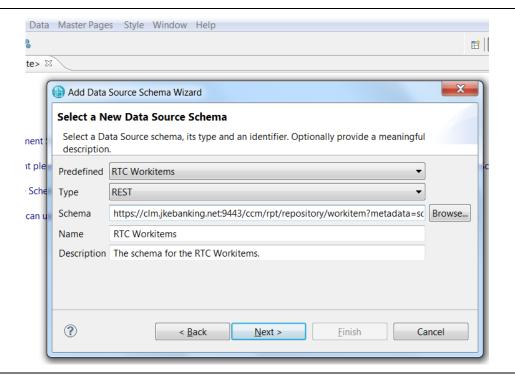
Exercise 7: Create a sprint document from RTC

Before you start

- 1. Open Document Studio or if already open close all templates and documents
- 2. Create a new document template
- 3. Log in CLM 6.0.1

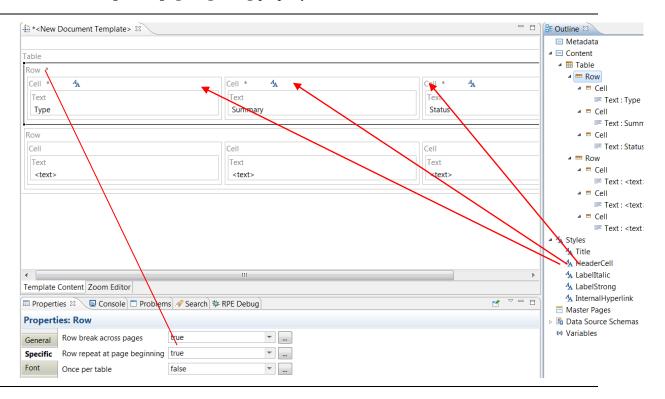
Add the data source

- Open the Add Data Source wizard and select RTC Workitems from the list of predefined data sources
- 2. Replace the [host] and [port] placeholders with "clm.jkebanking.net" and "9443"
- 3. Authenticate using the marco/marco credentials and "OAuth"
- 4. Name the data source **RTC Workitems**



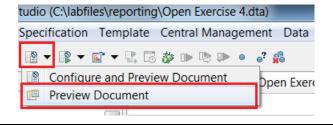
Create the document frame

- 1. Create a table with 2 rows and 2 columns
- 2. Create a text element in each cell
- 3. The content of the first row is the header of the table. Enter **ID**, **Title** and **State** in the 3 texts
- 4. Drag the **Header Cell** style from the outline onto the cells from the first row
- 5. Set the **Row Repeat at page beginning** property of the first row to **true** to create a header row



Save and test

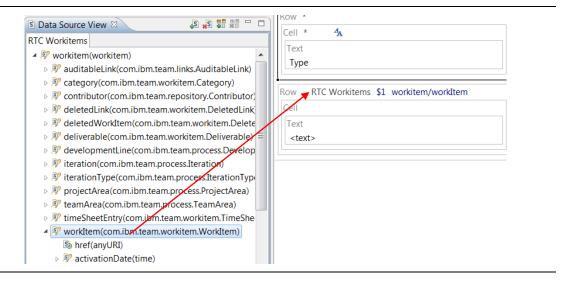
- 6. Save the template as Exercise 7.dta in c:\labfiles\reporting
- 7. Preview the report using the toolbar icon. Ignore the warning shown by RPE.



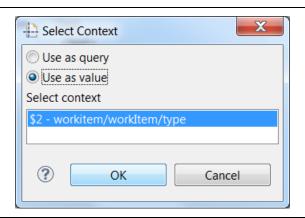
Tip: Use the **Preview** function for fast testing of the document template.

Add the workitem information

- 8. In the Data Source View expand the workitem node and select workiItem
- 9. Drag workitem/workItem onto the second row of the table

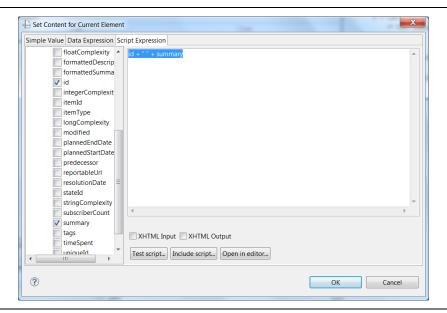


- 10. Add the properties for **workitem** in the second row
 - a. Drag the type element of workItem onto the text in the first cell in the second row
 - b. Drag the name element of type on the same text element. Select Use as value

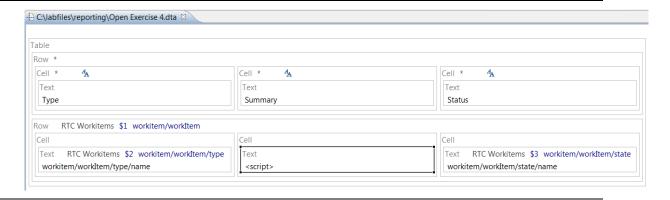


c. Double click the 2nd text cell in the second row and switch to the **Script Expression** tab

d. In the left hand side tree select **id** and **summary** and enter the following code in the expression editor: id + " " + summary



- e. Drag the **state** element of **workItem** onto the text in the third cell in the 2nd row
- f. Drag the **name** element of type on the same text element. Select use as value
- 11. You should have the following in the editor now



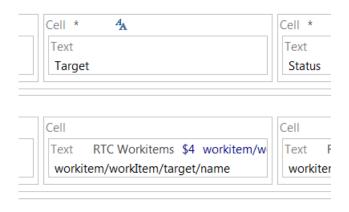
Test the report

- 1. Switch to the RPE Launcher perspective
- 2. Select the RTC Workitems data source in the report view
- 3. Enter RTC's Reportable REST URL in the **URI** property. The URL is https://clm.jkebanking.net:9443/ccm/rpt/repository/workitem
- 4. Use marco/marco for credentials and OAuth for authentication type

5. Save the report as Exercise 7.dsx in c:\labfiles\reporting

Add the target iteration for each work item

- 1. Add a new cell in the first row between cells 2 and 3
- 2. Add a text element with the content **Iteration**
- 3. Apply the **Header cell** style to the cell
- 4. Add a new cell in the second row between cells 2 and 3
- 5. Add a text element in the new cell
- 6. From the data source view drag the **target** element of **workItem** onto the third cell in the 2nd row
- 7. From the data source view drag the **name** element of type on the same text element. Select **Use as value**
- 8. You should have this now



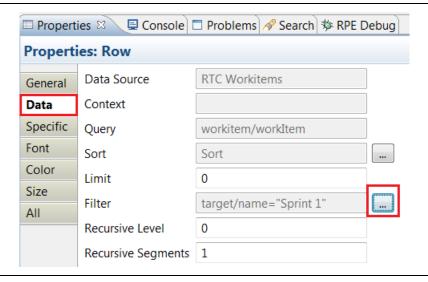
9. Run the report and observe the results

Filter the work items by their target iteration

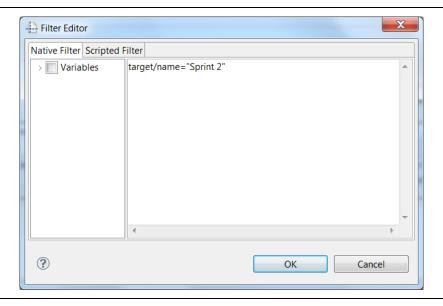
The most efficient way for filtering data is using the capabilities of the data source. This filters the data on the server side resulting in reduced network traffic and processing on the RPE side.

10. Select the 2nd row of the table, the one with the **workitem/workitems** query

11. In the Properties View select the **Data** tab



12. In the editor introduce the following expression: target/name="Sprint 2"



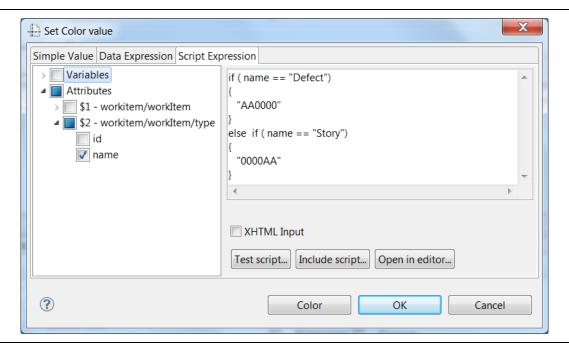
13. Run the report and observe that only workitems targeted for the first sprint will be shown

Color code the state and type information

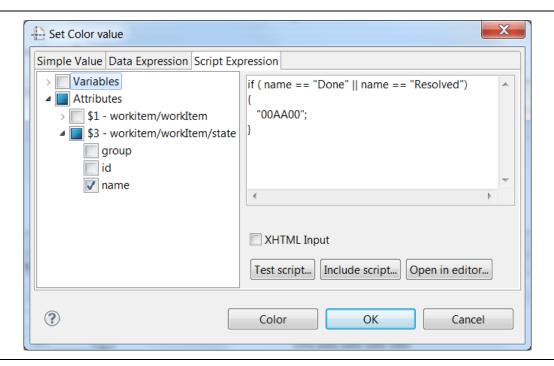
To make the document easier to read we can color code the state and type information.

- 1. Select the **type** text element (first cell in the second row)
- 2. In the properties view switch to the **Font** tab and open the **Color** editor

3. Enter the code printed below to color defects in red and stories in blue



- 4. Save the template
- 5. Select the **state** text element (last cell in the second row)
- 6. In the properties view switch to the **Font** tab and open the **Color** editor
- 7. Enter the code printed below to color resolved items in green



- 8. Save the template
- 9. Run the report and observe the results

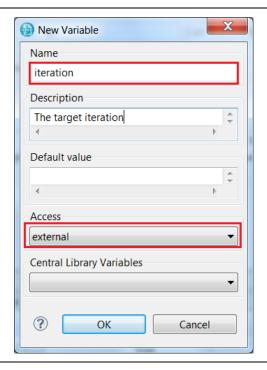
Make the filter customizable by the report user

To make the template more reusable we need to give end users the ability to change the target iteration to print without modifying the template. This can be done using Template variables.

1. From the Data menu create a new **external** variable and name it **iteration**

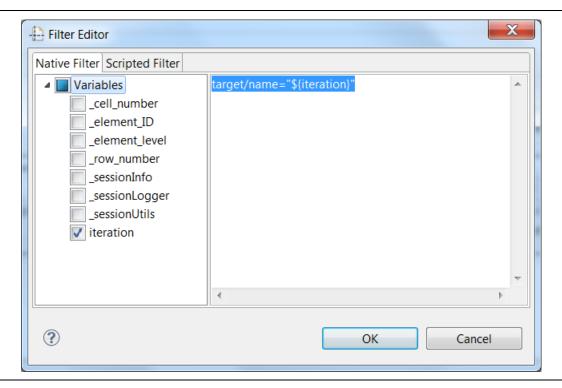
Note: External variables are visible in the RPE Document Specification and users can provide values to them. Internal variables are not visible in the report.

2. (Optional) provide a description and a default value to the variable



- 3. Select the 2nd row of the table, the one with the query
- 4. In the properties view open the **Filter** editor in the **Data** tab
- 5. Select the hardcoded **Sprint 2** text (including the quotes) and delete it

6. From the left hand side tree drag the **iteration** variable at the end of the expression. Surround \${iteration} with quotes. Your filter should look like this:



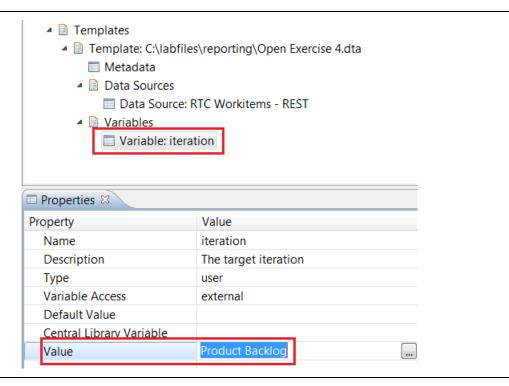
Test the changes

1. Save the template

Note: A variable will not appear in the report view of the RPE Launcher perspective until you save the template after your create it.

- 2. Switch to the RPE Launcher perspective
- 3. Select the variable in the Report View

4. Enter an iteration name in the "value" field, for example Product Backlog



- 5. Run the report and observe that only workitems targeting the Defect Backlog are printed
- 6. Change the variable's value to **Sprint 1** and run the report again. Observe the results.

Exercise 8: Create a traceability document with details using RPE and JRS

In this exercise you will learn how to mix JRS and traditional methods for building traceability documents. This exercise requires you to complete exercise 6

Note: This exercise has as prerequisites exercise 4

The document created in exercise 6 does not contain the Work item text. There are many other properties of the CLM resources that are not available in JRS as they are not required for the JRS reports but they are needed for documents. These details can be obtained by making requests to the Reportable REST API.

Before you start

- 1. Open Document Studio or if already open close all templates and documents
- 2. Load the template and document specification you have created in exercise 4
- 3. Log in CLM 6.0.1

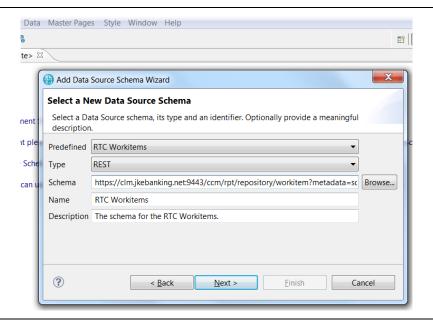
Add the work item details not present in JRS

Connect to the RTC Reportable REST API

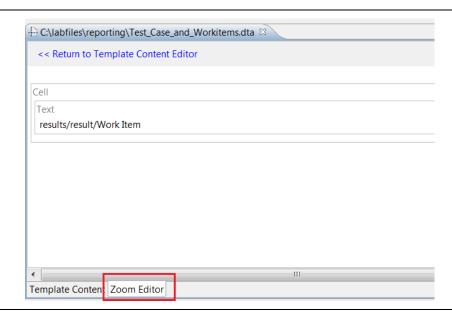
We'll add the work item text in the same cell that currently holds the work item's **summary**.

 Open the Add Data Source wizard and select RTC Workitems in the list of predefined data sources

- 2. Replace the [host] and [port] placeholders with "clm.jkebanking.net" and "9443"
- 3. Authenticate using the marco/marco credentials and "OAuth"

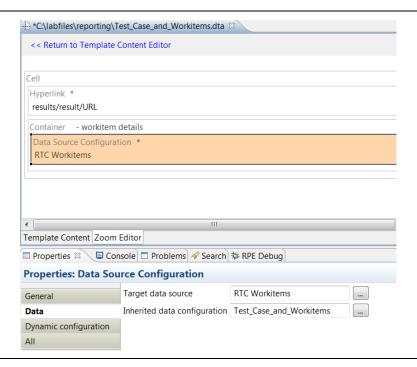


- 4. Select the 6^{th} cell in the 2^{nd} row of the template
- 5. Switch to the **Zoom Editor** to simplify editing. You can switch between full template content and zoom as you like.



Tip: Use the zoom editor every time you have to perform multiple operations in the same template element

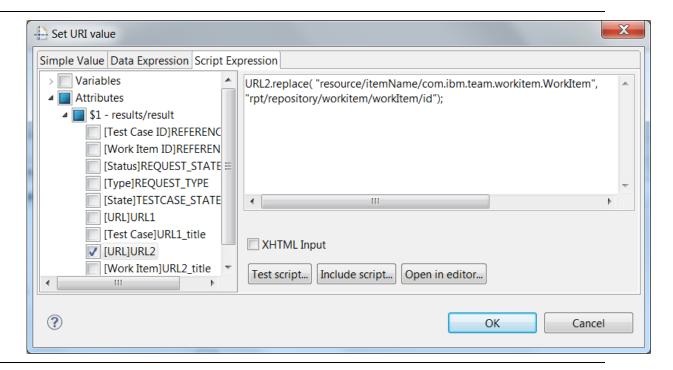
- 6. Create a container in the cell right after the text element
- 7. Using the General section of the properties view name the new container workitem details
- 8. Using the palette create a **Dynamic Data Source**
- 9. From the Outline View drag the **RTC Workitems** data source onto the **DDS** element
- 10. Since **JRS** and **RTC** use the same credentials we can set the DDS element to reuse the of the JTS data source for **RTC Workitems**



11. There is however a difference between the RTC and JRS data source, RTC uses OAuth for authentication while by default the JRS Reportable API uses Basic authentication. We can specify this difference in the **Dynamic configuration** section of the DDS properties. Set **OAuth** for the **Authentication** property

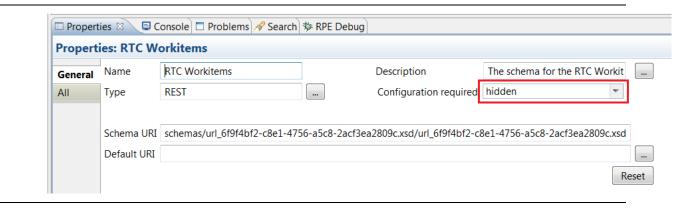


12. The URL of the RTC Workitems data source is built from the HREF stored in JRS



Tip: The RTC Workitems data source is now fully configured with no input required from the user whatsoever. Such data sources are best hidden from the report

13. .With the RTC Workitems data source selected in the Outline View set the **Configuration** required property to hidden in the Properties View

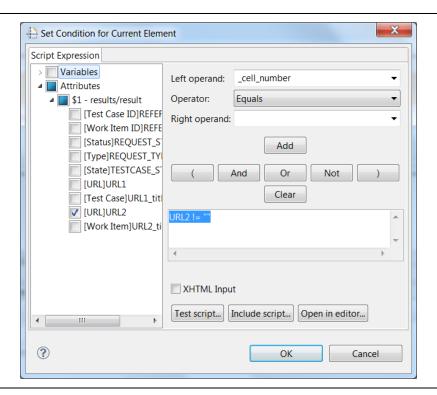


14. Save the template

Conditional execution

The workitem details can be retrieved only if there is a workitem associated with the test case. The hyperlink element is always generated, even when there is no linked test case. This will cause errors at runtime so we need to ensure RPE follows the RTC links only when such a link exists.

- 15. Select the workitem details element
- 16. In the Properties View open the **Condition Editor** from the **General** tab
- 17. Set the condition to be URL2 != ""

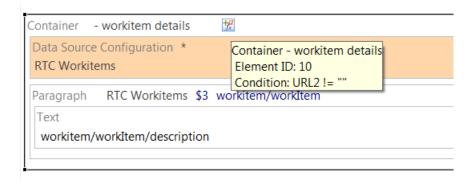


- 18. With the above condition in place the DDS container and its elements will only be evaluated if a workitem associated with the current test case exists
- 19. Save the template

Add the work item details

- 20. Create a paragraph right after the DDS element inside the workitem details container
- 21. From the left hand side tree drag the **workitem/workItem** element of the **RTC Wokitems** data source onto the paragraph
- 22. Create a text element inside the paragraph

23. Drag the **description** element of the **workItem** element onto the text element



24. Save the template

Test the results

1. Without any modifications to the DSX file from exercise 6 run the report again.

Test Case ID	Test Case	State	Type	Work Item ID	Work Item	Status
1	Verify dividend transfer frequency	Approved	Story	55	Frequency of dividend transfer A donor can choose the following frequencies.	New
10	Customers can Nominate an Organization	Approved	Story	64	Customers can Nominate an Organization Customers may nominate an organization for assistance whether a result of a catastrophic event or from some other justification.	New
11	Donors Choose an Organization	Under Review	Story	65	Donors Chooses an Organization Donor identifies a specific organization.	Done
12	Organization must provide justification for why funds are needed	Under Review	Story	66	Organization must provide justification for why funds are needed Organization must provide justification for why funds are needed.	New
13	Donation amount limits	Draft	Story	67	Donation by amount Donation amount cannot exceed donation limit specified in user profile; Donation	New

Exercise 9: Exploring the RPE Web UI

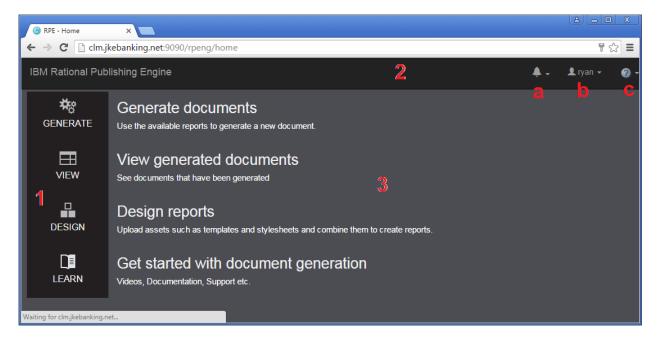
This exercise is meant to introduce you the RPE Web UI.

Getting started with RPENG

- 1. Open the browser and go to http://clm.jkebanking.net:9090/rpeng
- 2. Login using ryan/xyz

Home Page

The home page of RPENG contains 3 sections



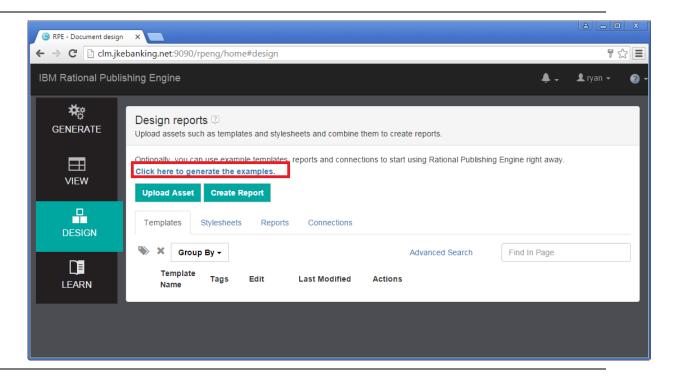
- 1. Navigation pane
- 2. Menu
 - a. Notifications
 - b. User settings and logout
 - c. Help
- 3. Content

Design Page

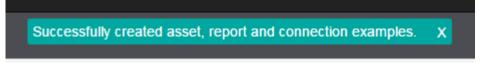
1. Go to the **Design** page

Note: The **Design** page is visible and accessible only for report designers. If you login using susan/xyz who is a report user you will not see be able to access it.

2. Click on the Click here to generate the examples link

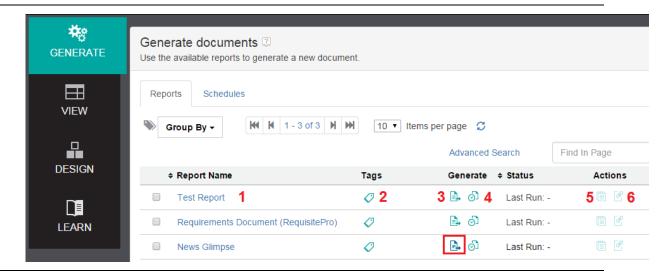


3. Wait for RPE to create the test data



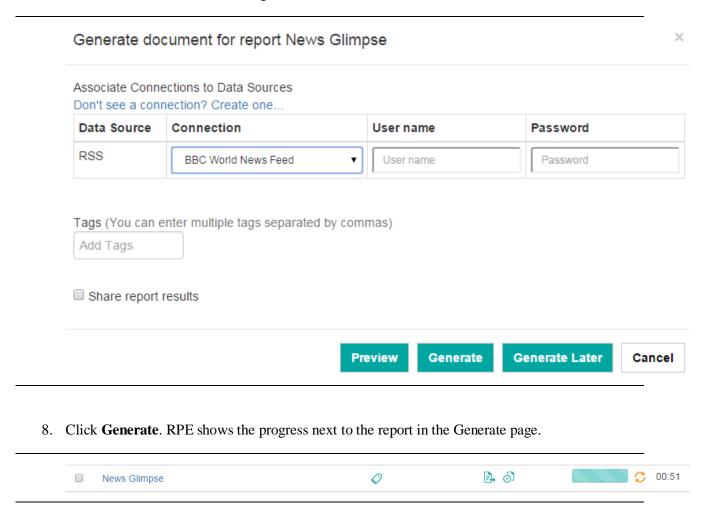
Generate Page

4. Once RPE finishes creating the examples go to the Generate page



- 5. The anatomy of the **Generate** page is
 - 1. The **name** of the reports available to you
 - 2. The **tags** for the report. If you are a report designer you can edit them right here
 - 3. Run Report immediately
 - 4. Schedule report
 - 5. Show the **list of documents** generated for this report
 - 6. Show the URL of the report. This is useful for sharing the document

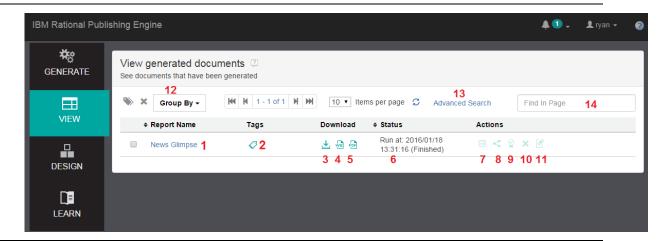
- 6. Start the **New Glimpse** report
- 7. In the Generate document dialog select "BBC World News Feed" for the RSS source.



Note: The time shown for the first execution of a report is always 1 minute. Subsequent executions of the report will show more accurate estimates based on past results.

View Page

9. Once the document generation is complete you can access the results from the View page



- 10. The **View** page gives access to all the report executions on the server that you have access to. The anatomy of the page is:
 - 1) Report Name
 - 2) **Document Tags** you can change them if you are the owner of the document
 - 3) **Download archive** downloads a zip with all the outputs of the report (doc, pdf)
 - 4) Download individual output format Word
 - 5) Download individual output format PDF
 - 6) Date and time of the report execution
 - 7) View the document generation **log**
 - 8) **Share document** with all other users on the server
 - 9) **Promote document** the document will not be deleted by the maintenance jobs of the server
 - 10) **Delete** result
 - 11) Get the link to this **report result**
 - 12) Allows to **group results** by report or ungrouped (default)
 - 13) Search report executions in the entire Database
 - 14) Search report executions in the current page

Exercise 10: Create and run a custom report in the RPE Web UI

This exercise will familiarize you with creating and using assets on the new RPE Web UI.

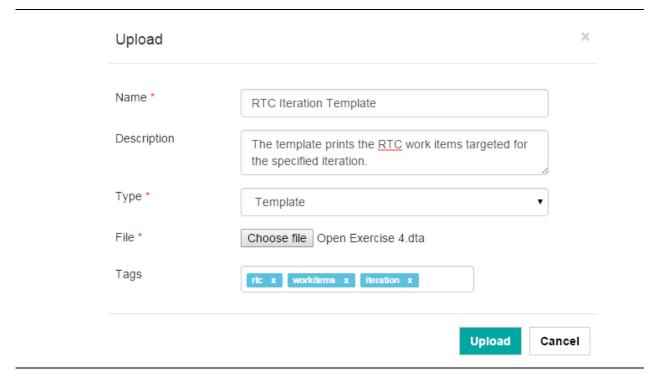
Note: This exercise has as prerequisite one of the exercise from 1 to 5. I will be using the template from exercise 4 and if you skipped that one you can use any of the other exercises.

Creating the RPENG assets

1) Login to RPENG as ryan/xyz

Create the RPENG template resource

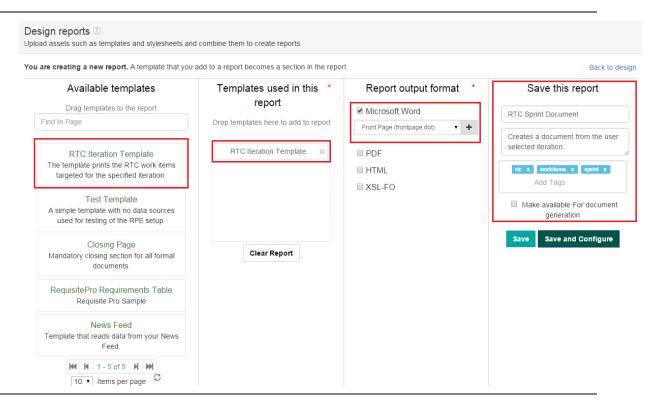
- 2) Go to **Design** page and click **Upload asset**. Make sure the **Templates** tab is selected
 - a. Give a meaningful name to the template
 - b. In the file section select the Exercise 6.dta from c:\labfiles\reporting
 - c. (optional but highly recommended) provide a description and tags



3) Click **Upload** to create the resource

Create the RPENG report

- 4) Click the **Create Report** button from the **Design** page
- 5) Fill the report
 - a. Drag the RTC Iteration Template from the Available Templates list into the Templates used in this report one
 - b. Select the Word output
 - c. (optional) select the **Front Page** stylesheet from the list
 - d. Give a meaningful **name** to the report and provide a suggestive **description** and **tags**



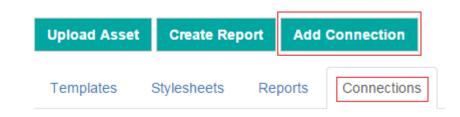
- 6) Click Save
- 7) Click **Back to design** to return to the **Design** page

Note: The **Make available for document generation** is left unchecked for the time being. This allows the template designer to test the templates before the end users gain access to them.

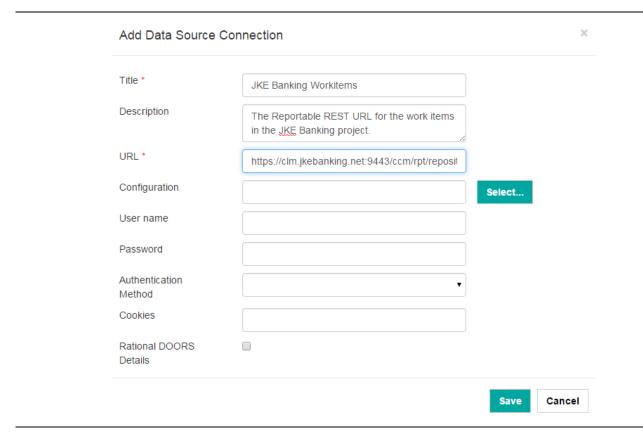
Create a data connection

Unlike the RPE Launcher, RPENG does not require the end user to enter URLs. It is the report designers' responsibility to create "connection" objects which the end user will then select at runtime.

8) In the design page switch to the "Connections" tab and click "Add Connection"



- 9) Enter the connection data
 - a. Provide a meaningful name to the connection alongside with a good description
 - b. In the **URL** field enter the same value you have used in in Exercise 6: https://clm.jkebanking.net:9443/ccm/rpt/repository/workitem



10) Click Save

Run the report

With all the assets in place you are now ready to run the report.

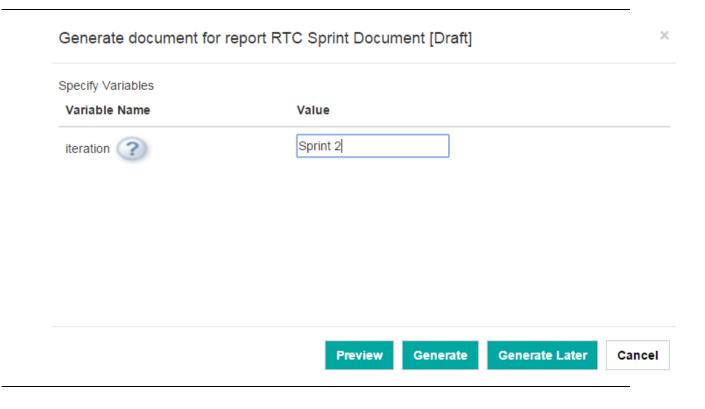
- 1) Go to the **Generate** page
- 2) Click the **Generate Document** button for the new report



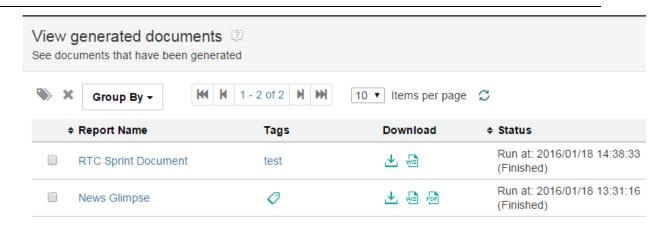
- 3) Configure the data source
 - a. Select the JKE Banking Workitems connection from the list
 - b. Enter your CLM credentials (ex: marco/marco)
 - c. (optional) provide tags for the generated documents

Generate document for report RTC Sprint Document [Draft] Associate Connections to Data Sources Don't see a connection? Create one... Data Source Connection User name **Password** RTC JKE Banking Workitems marco Workitems Tags (You can enter multiple tags separated by commas) test Share report results Next Cancel

- 4) Click **Next** to configure the variables
 - a. Enter **Sprint 2** in the **Iteration** variable



- 5) Click **Generate.** This closes the dialog and starts the process
- 6) When the document generation is complete the report becomes available in the **View** page.



Tip: If the document generation fails you can learn more about the error from the log in the View Page

Appendix A Useful knowledge

- RPE Actual
- IBM Fix Central
- IBM Software RFE Community
- Official RPE page
- RPE DevWorks Forum
- RPE DevWorks Wiki
- RPE Product documentation All versions
- RPE Product documentation 2.0.1
- RPE System Requirements
- RPE System Requirements Generator