

WebSphere Lab Jam

Business Process Management

WebSphere Operational Decision Management (Rules)

Lab Exercises



Catalog Number

Contents

LAB 1	RULES BUSINESS TRACK	13
	1.1 MANAGE THE BUSINESS RULES	13
	1.2 VALIDATE THE RULES CHANGES	24
	1.3 DEPLOY THE RULES	31
	1.4 AUTHOR IN RULE SOLUTIONS FOR OFFICE.....	36
	SUMMARY	41
	1.5 MONITORING THE RULES (OPTIONAL).....	43
APPENDIX A.	NOTICES	46
APPENDIX B.	TRADEMARKS AND COPYRIGHTS	48

THIS PAGE INTENTIONALLY LEFT BLANK

Overview

Business agility through synchronized business and IT cycles

The ability to deal with change in operational systems is directly related to the decisions that they are able to make. Every transaction, order, customer interaction or process is dependent on decisions, which are in turn dependent on particular internal or external requirements and situational contexts. Every change therefore affects decisions, many of which are handled automatically within business systems.

Business agility depends on responsive, intelligent decision automation. With WebSphere Operational Decision Management, you manage decisions separately from business applications, with more flexibility and responsiveness to the changing needs your business.

When decision management is separate from application code, business experts can define and manage the business logic, reducing the amount of time and effort required to update that business logic in production systems, and increasing the ability of an organization to respond to changes in the business environment.

WebSphere Operational Decision Management includes two main components, on both distributed and z/OS platforms:

- **Decision Server** for managing decisions separately from applications.

New IT architectures like Service Oriented Architecture (SOA) allow loose coupling of the applications or processes. Business logic can be encapsulated as decision services, which the applications and processes can call. Decision services enable the operation of Business Process Management (BPM) solutions with decision management.

- **Decision Center** for putting decision management in the hands of those who drive the business.

Business users can manage decisions directly with limited dependence on the IT department. The degree of dependence can range from limited review by business users of the business logic implemented by developers to complete control over the specification, creation, testing, and deployment of the business logic by business users.

From business policy to business rules

Business policies are statements that are used to make decisions such as pricing for insurance or loan underwriting, eligibility approvals for social or health services, or product recommendations for online purchases. Business policies are typically found inside application code, in the form of if-then statements, although they may also be stored elsewhere for documentation purposes, such as in procedural manuals and other documents.

A business policy can be expressed as several business rules. Here is an example of the kind of business policy that might be familiar:

`Customers who spend a lot of money in a single transaction need an upgrade.`

The process of capturing rules consists of formalizing the vocabulary required to express the policy as a conceptual object model and representing the logic of the business policy as if-then statements.

After the vocabulary has been created, the above business policy can be implemented with the following business rule:

```
if
    the customer's category is Gold
    and the value of the customer's shopping cart is more than $1500
then
    change the customer's category to Platinum
```

When a business policy also has an IT policy or security policy embedded in it, you can combine Business Rule Management with additional capabilities to handle the business policy aspects. For example, *customers who spend a lot of money should be routed to a preferential service* or *customers who spend a lot of money require additional security on their transactions*.

In the form of business rules, the business logic can be packaged and called from the application code as a business rule application. Therefore, changes to the business policy do not require changes to the application or process code.

From event pattern detection to event rules

An event is an electronic signal indicating that a change in the state of the business has occurred. Orchestrating business events so that the right applications run at the right time for the right purpose is a challenge that can be particularly difficult with the large variety of business systems currently running in the enterprise. A wide range of technologies is also employed, ranging from batch processing applications to client/server, and to browser-based intranet and internet applications. Orchestrating the processing of the events that occur in these systems, as well as the events that occur manually, might potentially mean major system redesign and many months of modifications, tests, and deployment.

A rapid business systems orchestration requires an alternative approach to complex and expensive redesign and redeployment of existing systems is required. The solution is to implement a Business Event Processing layer that sits across existing systems, takes advantage of functions already developed in those systems, and manages the complex interactions (business events) that can occur between those systems. This layer of architecture is known as Business Event Processing. The Business Event Processing layer communicates significant events in one business system to other systems that require the information to respond to the critical event.

In large organizations, tens of millions of events occur everyday, but not all events or event occurrences are of equal importance. Providing insight requires the ability to determine when a pattern of seemingly unrelated events from one or more sources has occurred and then to coordinate the execution of the responses to that pattern of events.

Business Event Processing is the ability to sense when a business event or event pattern has occurred (or not occurred), indicating an actionable business situation, and to coordinate the right response or action, at the right time.

Event rules help detect, and respond to, event patterns among like or related events, missing events and aggregate events. Event rules also relate the pattern detection to a context and apply a dimension of time to the pattern. So, for example, the following logic can be created:

```
if events A and B occur and event C does not occur in <time frame>, then do
actions X, Y and Z after time frame
```

For example, on a retail web site, if a customer adds a book to his or her shopping basket (event A) and views the delivery information page (event B) but does not complete the purchase at the online checkout (event C) within one week, then send an E-mail to this customer (action X). After one more day has passed, update the customer favorites database with the book details (action Y), and send a message to the Sales department to tell them this customer did not complete the purchase (action Z).

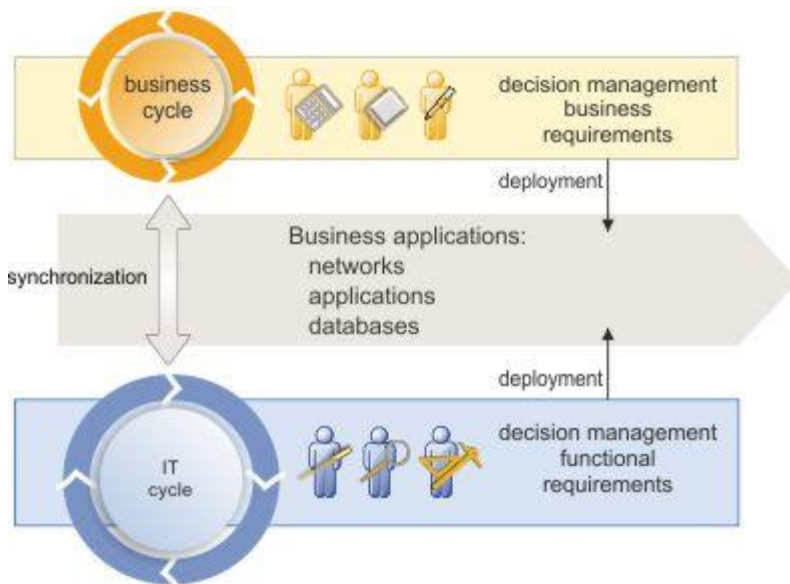
By using predefined logic that describes how business systems are to interact, the event runtime can notify those systems in real time so that they take the appropriate action.

Synchronized business and IT cycles

WebSphere Operational Decision Management provides an environment for designing, developing, and deploying business rule applications and event applications. The IT cycle consists of developing and maintaining this infrastructure. After the infrastructure is set up, distributed business teams can start collaborating through a Web-based environment to create and maintain business rules and events.

The life cycles of decision management and application development can evolve in parallel. Decisions can evolve as required by the business context, without putting an extra load on the development of the application. Each time the application evolves, the decision management environment synchronizes with the application.

With this separation, decisions and application architecture can be managed asynchronously. For example, application developers can develop a new application version in response to changing application infrastructure and additional core business requirements. At the same time, policy managers can work on new decisions delivered in response to an evolving market, changing regulatory environment, or new patterns of events.



In addition to working on different time lines, developers and business users also expect to work with different tools, reflecting their different skill sets and views of the application.

For example, developers are accustomed to the Java world. They use source-code management systems to work simultaneously on separate copies of a project without interfering with each other.

Business users, on the other hand, do not concern themselves with the details of application development, but are interested in testing and managing decisions. Therefore, they need tools that can help them author, organize, and search for rules in the context of the overall policy.

With developers and business users working in their own environments at their own pace, the work of these two groups must be synchronized and merged.

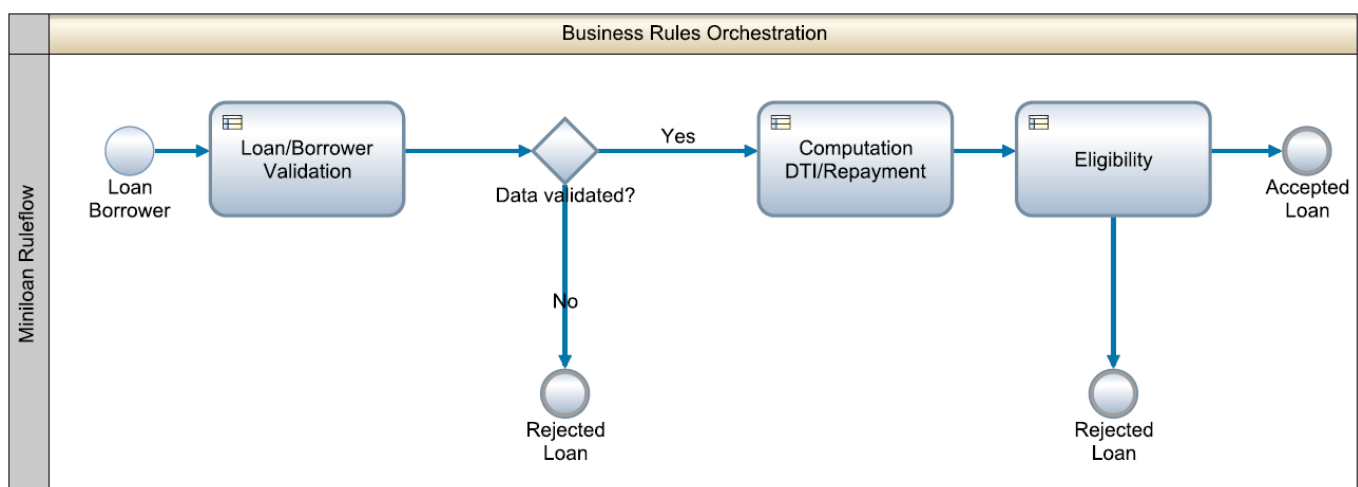
Finally, both developers and business users require access to a rule execution environment to deploy rules to enable testing, validation, and rollout to production of new and changed business logic.

Introduction

This section describes the scenario you will work through during this tutorial.

The Miniloan Business Application

The getting started tutorial is based on a web based online loan application. The tutorial objective is to modify the application to support the new business process shown below. On receipt of the Loan application, the borrowers' credit score would be checked and the interest rate identified. This information would normally be provided from existing services within the organization so for the tutorial will be provided as part of the input data. The Debt-To-Income (DTI) and the Yearly Repayments will be calculated thanks to the rules in the step Computation.

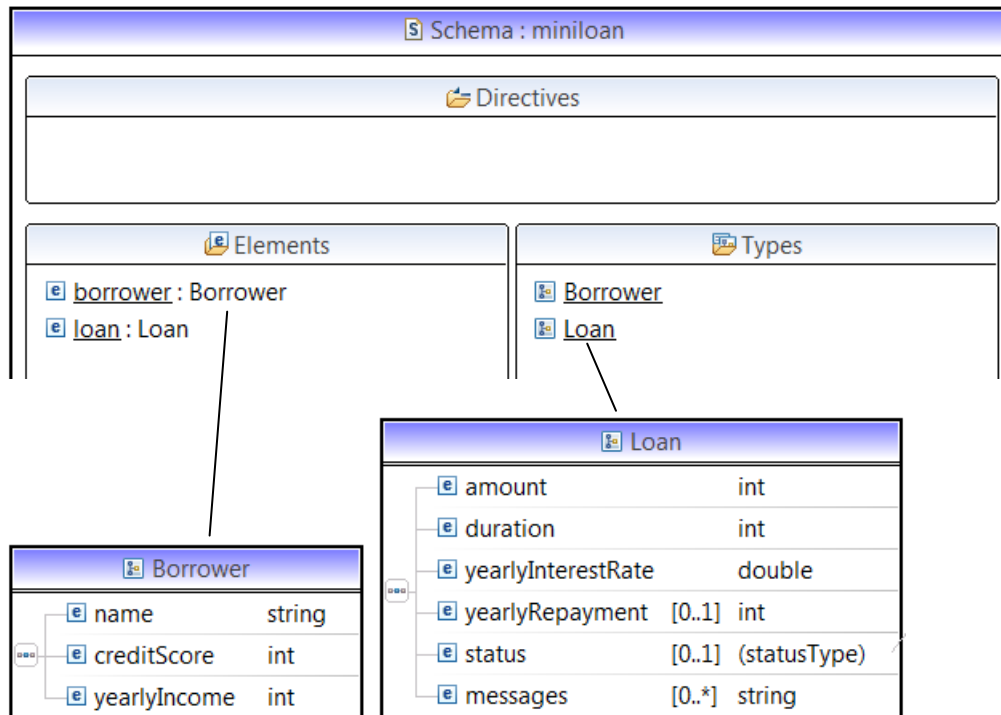


The next stage in the process is to make a decision about whether the loan should be accepted or not. This decision will use business rules to determine the loan risk based on the relative debt to income ratio and the borrowers' credit score. If the loan is not approved, the borrower is informed that the loan has been rejected together with the reasons.

A additional stage could be to use event based decision making to determine if the borrower has already been offered a loan several times but is still continuing to submit new applications. If any particular borrower has already been offered a loan 3 times, another offer should not be made but the borrower should be requested to contact a helpdesk to follow up (and accept) the existing offer.

The Miniloan Information Model

The information available to make the decision is shown below.



The Borrower represents the person applying for the loan. They are identified by a name and have a yearly income which is used to assess their debt to income ratio. In order to make a good decision about the risk of the loan, the borrower's credit score is also required. This would usually be provided from an outside source but for these purposes we will have to supply this information as input.

The Loan represents the loan characteristics and is divided up into three sections.

- The amount and the duration (or repayment period in months) will be provided by the borrower.
- The yearly interest rate will be provided by the application at submission time. It could be modified by the rules but in our example it is fixed.
- The yearly repayment and the loan status will be populated by the rules.

Introducing the Tutorial Environment

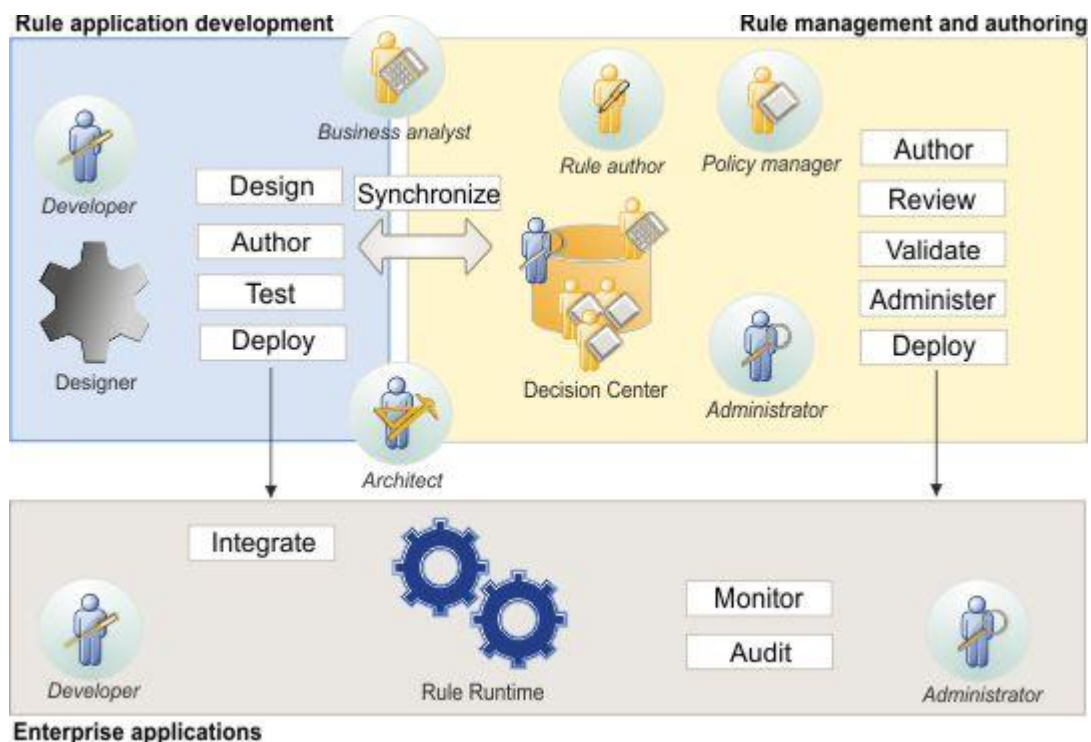
This section introduces the components you will use in the tutorial and the steps you will take.

The WebSphere Operational Decision Management WODM modules

WebSphere Operational Decision Management comprises a set of modules that operate in different environments, but also work together to provide a comprehensive decision management platform.

- **Rule Designer** – is an integrated development environment (IDE) for rule applications that integrates directly into the Eclipse family of integrated development environments, including Eclipse, IBM Rational Application Developer (RAD) and ISM Rational Software Architect (RSA). Rule Designer is used by developers and architects to develop and integrate business rule sets into applications.

- **Rule Execution Server** – is a robust, J2SE and J2EE-compliant execution environment for deploying business rule SOA services to the leading web and application servers such as WebSphere Application Server. The Rule Execution Server includes components for synchronous, asynchronous and web service based invocation of business rules and includes a web administration console. The Rule Execution Server is fully integrated with the Rule Designer and Decision Center to support business rule deployment for both developers and policy managers.
- **Decision Center** – A scalable rule management server and repository with a collaborative web environment for authoring, managing, validating, and deploying business rules. Decision Center provides project governance, including role-based security, history maintenance, and custom metadata. Decision Center provides enhanced collaboration between teams through multi-user access for business users and synchronization between IT and business user environments.
- **Rule Solutions for Office** – is used for offline authoring and sharing of the business rules using standard office tools. Rule authors write rules in Microsoft Word and edit decision tables in Microsoft Excel. They can create mixed rule and non-rule content in a RuleDoc, and retain semantic information together with the actual implementation content of the rules.



Two categories of users are involved in developing and maintaining a decision management solution:

IT users

Architects, developers, and administrators develop and maintain the rule application. Developers work with Rule Designer in Eclipse for design, Java development, and rule project development. Using Decision Validation Services, they can also test rulesets against real or fictitious

scenarios to support and troubleshoot problems found in Decision Center.

Business users

- Business users work with Decision Center to write and maintain business rules, both during application development and after the application is deployed to production.
- Business users can perform end-user testing and simulation in Decision Center. Business analysts can simulate business outcomes, run updated rules against historical data, simulate expected changes in data profiles against existing rules, and analyze aggregate outcomes.
- Policy managers and other business users can use Rule Solutions for Office to author rules, in a familiar environment. RuleDocs are Microsoft Office documents that contain business rules. Business users can publish RuleDocs from Decision Center, edit the RuleDocs in Word or Excel, and update their changes back into Decision Center.
- RuleDocs can be published to any WebDAV server and can be short-lived or managed in a content management system such as SharePoint.

Overview of Tutorial tasks

The tutorial also offers an introduction to both **Decision Server and Decision Center** where you take the role of a business user first to manage the business rules developed in Decision Server from a Web-based environment and then the role of an IT user to discover how the business rule application was built.

The tutorial is divided into three parts using the modules provided with WebSphere Decision Server as shown below.

Students are advised to progress through these tasks in order. Each task is divided up into steps that incrementally build the solution to the scenario described in the next section. Solutions are available at certain steps in the tutorial allowing the student to catch up quickly if mistakes are made in earlier tasks.

Pre-requisites

Start Sample Server and the Miniloan web application

Operational Decision Management provides a sample server to run Decision Center. Before signing in to Decision Center, you must start the sample server.

This tutorial is based on the miniloan-rules rule project that is already deployed to Decision Center.

The scenario on which the tutorial is based features the web-based application of an online lender.

To begin, take a look at the Miniloan application in its initial state.

To start the web application Miniloan and the Decision Center:

1. Click **Start > All Programs > IBM WebSphere Operational Decision Management V7.5 > Start Samples Server**.
2. Wait until you see the message `BUILD SUCCESSFUL` in the console.

```
[exec]
[exec] wsStartServer:
[exec] [startServer] profileName=WODMSample750 registry=C:\IBM\WebSphere700
17\AppServer\properties\profileRegistry.xml
[exec] [startServer] profileHome=C:\IBM\WebSphere70017\AppServer\profiles\W
ODMSample750
[exec]
[exec] BUILD SUCCESSFUL
[exec] Total time: 1 minute 14 seconds

checkIfIsInstallingDC:
update.decisioncenter.db:
delete.new.installation.files:
BUILD SUCCESSFUL
Total time: 1 minute 22 seconds
Press any key to continue . . .
```

Note: The server might take a few minutes to start.

3. Enter the following URL in a browser:

`http://localhost:9081/miniloan-server`

The Miniloan application is displayed:

4. Make sure the checkbox `Use rules` is selected and click **Validate Loan**.

The results of the validation are:

Borrower Information	Loan Information
Name: Joe	Amount: 500000
Yearly Income: 80000	Duration (months): 240
Credit Score: 600	Yearly Interest Rate : 0.05

Use Rules

Validate Loan

Sorry, the loan has been rejected.

Reasons:

- Debt-to-income ratio is 31%
- Too big Debt-To-Income ratio

Rules Execution Summary	
4 rule(s) fired	1: repayment in rule task: miniloan#computation 2: debt-to-income in rule task: miniloan#computation 3: repayment and score 4 in rule task: miniloan#eligibility 4: minimum income in rule task: miniloan#eligibility

[Execution details](#)

Lab 1 Rules Business Track

In this lab, you will act as a business user, using Decision Center to write, modify, test, and deploy business rules.

You will first use Decision Center to search for rules in the rule repository. Based on the query results, you will edit a rule that determines the minimum credit score required for loan eligibility. You will also modify a decision table and then test the rules by using Decision Validation Services. Finally, you will learn how to deploy rule applications from Decision Center to Rule Execution Server.

Pre-requisite: Start the sample server, as described in Start Sample Server and the Miniloan Web application, if you have closed it.

1.1 Manage the business rules

Business users create and edit business rules in Decision Center. Their work is saved in the central rule repository, which manages rule versioning, rule history, and multi-user access.



This task should take you about 60 minutes to complete.

- Step 1: Explore the rule project
- Step 2: Write and run a query
- Step 3: Modify a rule
- Step 4: Modify a decision table
- Step 5: Review the rule history
- Step 6: Create a smart folder

1.1.1 Explore the rule project

In Decision Center, you navigate through your rules in the business user environment.

To explore the rule project:

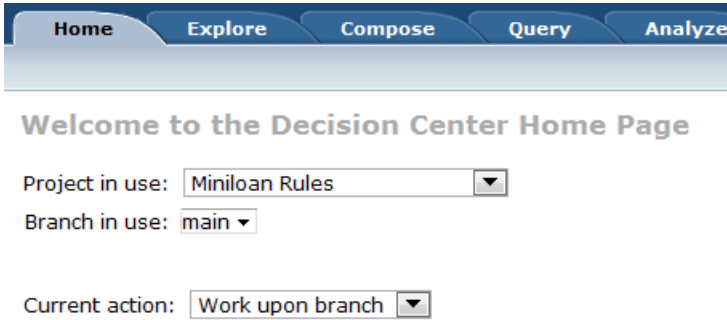
__1. Sign in to Decision Center using the following details:

Username: `rtsUser1`

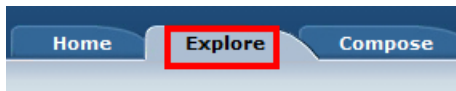
Password: `rtsUser1`

Note: Decision Center can handle different user profiles. Here you sign in as a regular business user.

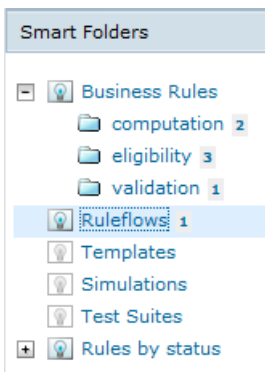
- __2. On the Decision Center **Home** tab, in the **Project in use** field, select Miniloan Rules.



- __3. Click the Explore tab. In the tree view on the left hand, the rules are organized into rule folders, and some additional views such as Test Suites.

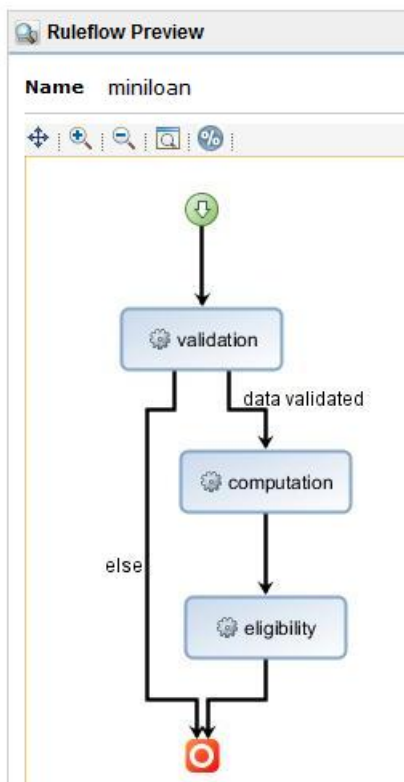


- __4. Under Smart Folders on the left, click the Ruleflows folder.

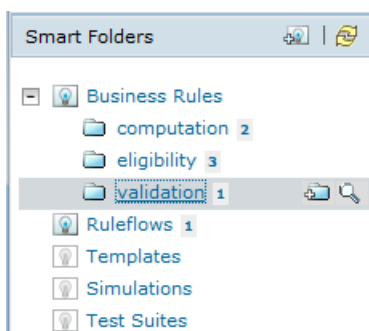



__5. Preview the content of the miniloan ruleflow.

<input checked="" type="checkbox"/> Actions	Name
<input checked="" type="checkbox"/>	miniloan





__6. Under Business Rules on the left, click the validation folder.



- __7. Preview the content of the maximum amount rule by clicking  Preview beside the name of the rule in the table to the right.

Display by 10

	Actions	Name	Status
<input checked="" type="checkbox"/>		 maximum amount	New

Rule Preview

[Edit](#)

Name maximum amount
Status New

if
 the amount of 'the loan' is more than 1000000
then
 reject 'the loan' with the reason: "The loan cannot exceed 1,000,000" ;

- __8. Click the `eligibility` folder.
- __9. Preview the content of the `repayment and score` decision table.

Decision Table Preview

[Edit](#)

Name repayment and score
Status New

if
 the status of 'the loan' is not "Rejected"

	Debt-to-income %		credit score		Loan Status	Reason
	min	max	min	max		
1	0	30	0	200	Rejected	debt-to-income too high compared to credit score
2			200	800	Accepted	
3	30	45	0	400	Rejected	debt-to-income too high compared to credit score
4			400	800	Accepted	
5	45	50	0	600	Rejected	debt-to-income too high compared to credit score
6			600	800	Accepted	
7	Otherwise		0	800	Rejected	debt-to-income too high compared to credit score

1.1.2 Write and run a query

In your business user role, you are asked to change the allowable debt-to-income ratio from 30% to 40% for people who earn less than \$50,000.

Your starting point in Decision Center is to find all the rules in your project that use the yearly income on which the debt-to-income ratio is based.

To write and run a query:

- __1. Make sure you are signed in to Decision Center as `rtsUser1` and that the **Project in use** is set to `Miniloan Rules` in the Home Page.
- __2. Click the **Query** tab, and click **New**.
- __3. In the **Name** field, type `Uses Yearly Income`.
- __4. Click **[such that]** and select `<a rule artifact> uses the value of <a member>`.
- __5. Click `<a rule artifact>`, and select each business rule.
- __6. Click `<a member>`, and build the following statement:

`the yearly income of a borrower`

Your query looks like this:

- __7. Click **Save** and **Run Query**.

The query results show the `debt-to-income` rule and the `minimum income` rule.

- __8. To help you evaluate which rules you must change to implement the new policy, click **Generate Report on Query Results**.
- __9. On the Project Report page, click `Miniloan Rules` to view the report.

An HTML report is displayed, showing details for all the rules returned by your query.

By looking at the report, you see that `debt-to-income` already calculates the debt-to-income ratio, therefore only `minimum income` needs to be changed to implement the policy change.

- __10. Close the HTML report, and click the **Back** button in the Analyze tab, to return to the Query tab.



1.1.3 Modify a rule

Now that you have found the rules that need to be modified, you can change the allowable debt-to-income ratio from 30% to 40%.

To change the allowable debt-to-income ratio:

- ___1. In the query results table, select minimum income and then click **Quick Edit** button beside the rule name:

Display by 10 ▼

<input type="checkbox"/>	Actions	Name	Status	Priority	Last Changed By	Last Changed On
<input type="checkbox"/>		debt-to-income	New		rtsAdmin	12/16/11 11:58 AM
<input type="checkbox"/>		minimum income	New		rtsAdmin	12/16/11 4:08 PM

- ___2. The following panel is launched to edit the content of the rule and its properties:

Rule Editing

Save | Cancel

Name*

Status*

[definitions]

if

the yearly repayment of the loan [±] is more than the yearly income of the borrower * ▼ 0.3 [±] ✕

↕

then

reject the loan with the reason: ▼ Too big Debt-To-Income ratio [±] ✕

↕

[else]

Add a comment to this version

- ___3. In the **if** part of the rule, click the value 0.3 to activate the field, and change the value to 0.4. Press RETURN to incorporate the change.
- ___4. Click the ↕ blue arrow just above the **then** keyword and add the following condition:
and the yearly income of the borrower is less than 50,000
- ___5. Change the **Status** of the rule from New to Defined.
- ___6. In the **Add a comment to this version** section, type the following text:

Increase the debt-to-income from 30% to 40% for salary below 50,000

Rule Editing

Save | Cancel

Name*

Status* Defined ▼

[definitions]

if

the yearly repayment of the loan [±] is more than the yearly income of the borrower * ▼ 0.4 [±] ✗
 and the yearly income of the borrower [±] is less than ▼ 40000 [±] ✗

↴

then

reject the loan with the reason: ▼ Too big Debt-To-Income ratio [±] ✗

↴

[else]

Add a comment to this version


Increase the debt-to-income from 30% to 40% for salary below 50,000

__7. Click **Save**. The new rule details are displayed.

1.1.4 Modify a decision table

You are asked to make your lending criteria less restrictive, so that you reject a loan for someone whose debt-to-income ratio is between 45 and 50 only if the applicant's credit score is less than 500. To do this, you modify the repayment and score decision table.

To modify the decision table and correct any resulting errors:

- __1. Click the **Explore** tab. Under **Business Rules** on the left, click the `eligibility` folder.
- __2. Preview the content of the `repayment` and `score` decision table by clicking  Preview beside the name of the rule in the table to the right.

Display by 10 ▾

<input type="checkbox"/>	Actions	Name	Status	Priority	Last Changed By	Last Changed On
<input type="checkbox"/>		minimum credit score	New		rtsAdmin	12/16/11 7:24 AM
<input type="checkbox"/>		minimum income	New		rtsAdmin	12/16/11 9:00 AM
<input checked="" type="checkbox"/>		repayment and score	New		rtsAdmin	12/16/11 10:05 AM

3 Results

Decision Table Preview

[Edit](#)

Name repayment and score
Status New

	Debt-to-income %		credit score		Loan Status	Reason
	min	max	min	max		
1	0	30	0	200	Rejected	debt-to-income too high compared to credit score
2			200	800	Accepted	
3	30	45	0	400	Rejected	debt-to-income too high compared to credit score
4			400	800	Accepted	
5	45	50	0	600	Rejected	debt-to-income too high compared to credit score
6			600	800	Accepted	
7	Otherwise		0	800	Rejected	debt-to-income too high compared to credit score

__3. Click **Edit**. The following panel is launched to edit the content of the decision table and its properties:

Decision Table Editing

[Save](#) | [Cancel](#)

Name*
Status*

1 - 7 | All

Condition editor


Debt-to-income ratio [±] is at least ▼ 0 [±] and less than ▼ 30 [±] [and/or]

i Use this editor to edit the selected cell.

	Debt-to-income %		credit score		Loan Status	Reason
	min	max	min	max		
1	0	30	0	200	Rejected	debt-to-income too high compared to credit score
2	0	30	200	800	Accepted	
3	30	45	0	400	Rejected	debt-to-income too high compared to credit score
4			400	800	Accepted	
5	45	50	0	600	Rejected	debt-to-income too high compared to credit score
6			600	800	Accepted	
7	Otherwise		0	800	Rejected	debt-to-income too high compared to credit score

__4. Click the cell in the **max** column that contains 600 under **credit score**. Replace 600 with 500, and then press ENTER.

- __5. Notice how a warning is displayed on the column header. This indicates that there is an error. A warning message states that Rows have gap(s).

	Debt-to-income %		credit score 	
	min	max	min	max
1	0	30	0	200
2			200	800
3	30	45	0	400
4			400	800
5	45	50	0	500
6			600	800
7	Otherwise		0	800

- __6. To solve this problem, click the cell in the **min** column under **credit score** containing 600, and replace 600 with 500, and then press ENTER. The problem disappears.
- __7. Change the **Status** of the rule from New to Defined.

Decision Table Editing

Save | Cancel

Name* repayment and score

Status* Defined

1 - 7 | All

Condition editor

the credit score of the borrower [±] is at least ▼ 0 [±] and less than ▼ 500 [±] [and/or]

i Use this editor to edit the selected cell.

	Debt-to-income %		credit score		Loan Status	Reason
	min	max	min	max		
1	0	30	0	200	Rejected	debt-to-income too high compared to credit score
2			200	800	Accepted	
3	30	45	0	400	Rejected	debt-to-income too high compared to credit score
4			400	800	Accepted	
5	45	50	0	500	Rejected	debt-to-income too high compared to credit score
6			500	800	Accepted	
7	Otherwise		0	800	Rejected	debt-to-income too high compared to credit score

- __8. Click **Save**.

The new decision table details are displayed.

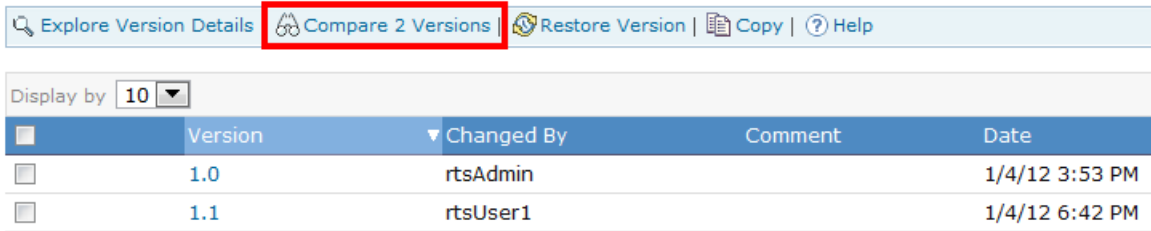
1.1.5 Review the decision table history

Any changes made to a rule are stored in the rule repository, while new versions of a rule are given new version numbers. This allows us to compare differences between rule versions, and restore older versions if desired.

In this exercise we will review the decision table history for `repayment` and `score`. Now you can check the history of the decision table that you modified, and compare the differences between the two versions.

To review the history of the decision table:

- __1. Click **History** in the toolbar. On the History page, you can see the versions of the decision table since its creation.
- __2. Select versions 1.0 and 1.1, and click **Compare 2 Versions** to see the changes in the rule between these two versions.



Each line in the table corresponds to a difference between the versions, in this case a change to the status of the rule, and another to its contents.

- __3. Click on the **Content** link in the second row to display the differences between the old and new values of the rule content.

OLD VALUE							NEW VALUE						
	Debt-to-income %		credit score		Loan Status	Reason		Debt-to-income %		credit score		Loan Status	
	min	max	min	max				min	max				
1	0	30	0	200	Rejected	debt-to-income too high compared to credit score	1	0	30	0	200	Rejected	debt
2			200	800	Accepted		2			200	800	Accepted	
3	30	45	0	400	Rejected	debt-to-income too high compared to credit score	3	30	45	0	400	Rejected	debt
4			400	800	Accepted		4			400	800	Accepted	
5	45	50	0	600	Rejected	debt-to-income too high compared to credit score	5	45	50	0	500	Rejected	debt
6			600	800	Accepted		6			500	800	Accepted	
7	Otherwise		0	800	Rejected	debt-to-income too high compared to credit score	7	Otherwise		0	800	Rejected	debt

Lines that have been modified appear in red.

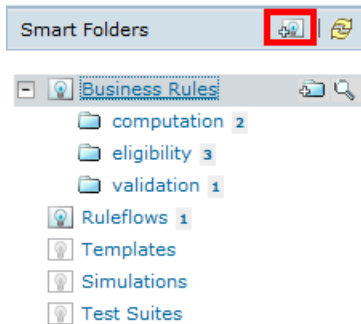
Business users can also use Microsoft Excel to edit rules and work offline with the Rule Solutions for Office.

1.1.6 Create a smart folder

To view all the rules and decision tables contained in the rule project, we will use the Explorer view and create a smart folder.

To create a smart view:

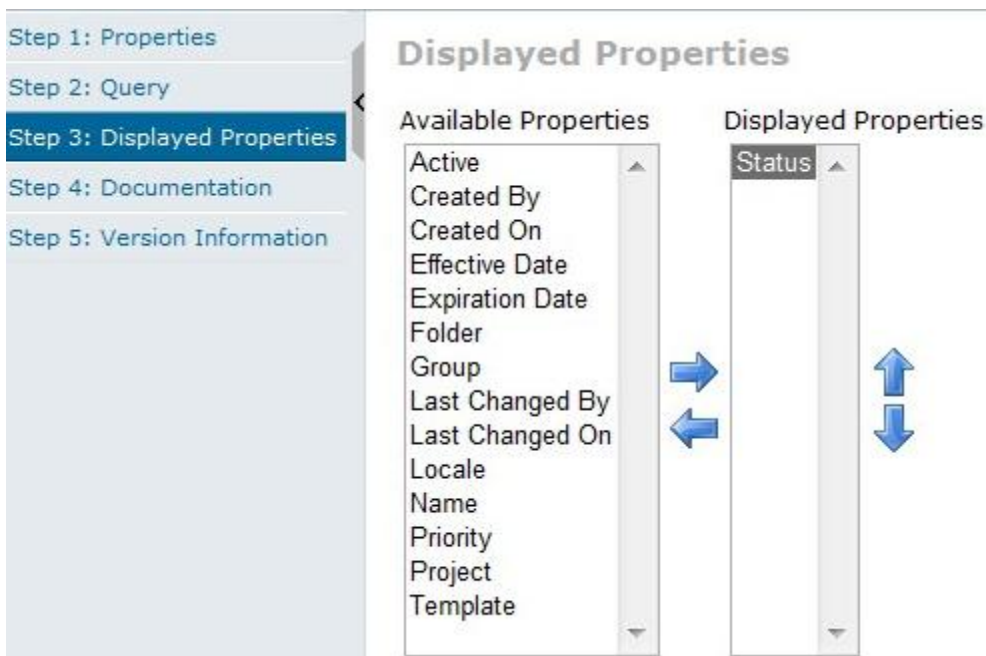
- __1. Click the **Explore** tab.
- __2. In the **Explore** toolbar, click the New button to create a new view.



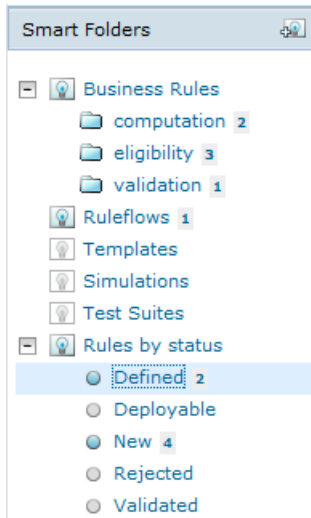
- __3. In the **Step 1: Properties**, type `Rules by status` in the Name field.



- __4. Click on **Step 3: Displayed Properties**, select `Status` in the list of Available Properties, then click on the middle arrow to push it to the Displayed Properties.



__5. Click **Finish** and you will see the new smart folder in the Explore tab.



Now, you can see that both rule and decision table you set with the status **Defined** are listed under Defined with the number (2).

After editing the business logic in Decision Center, you validate the rules to make sure that they behave as expected. You will do this in the next task.

1.2 Validate the rules changes

In this task, you use Decision Validation Services in Decision Center to run tests and check that the rules execute as expected.

Decision Validation Services allows you to test a ruleset against typical execution scenarios to check that the rules execute as expected.

In the previous task, you modified the debt-to-income ratio in the `minimum income` rule and the credit score in the `repayment` and `score` decision table. In this task, you run the scenarios to check that the modified rule behaves as expected.

You assume the role of business user to create a test suite in Decision Center and run it.



This task should take you about 20 minutes to complete.

- Step 1: Create and run a test suite
- Step 2: Modify the Excel scenario file
- Step 3: Edit and run the test suite

1.2.1 Create and run a test suite

Now that you've modified and created rules in Decision Center, the rules should be tested before they are deployed to the production Rule Execution Server environment. The testing and simulation is made possible through Decision Validation Services.

In this exercise you will use Decision Validation Services to test and simulate the rule changes you made in the prior exercises.

To create a test suite:

- __1. Make sure you are signed in to Decision Center as `rtsUser1` and that the **Project in use** is set to `Miniloan Rules` in the Home Page.
- __2. Click the **Compose** tab.
- __3. In the **Compose** tab, select **Test Suite** to the left, and then click **OK**.
- __4. In **Step 1: Properties**, enter `Miniloan Test` as name for the test suite, and click **Next**.
- __5. In **Step 2: Rules Tested**, keep the default option to test all the rules in the project, and click **Next**.
- __6. In **Step 3: Scenarios**, click on the blue hyperlink **Generate template**. A wizard opens.
- __7. Under **Expected Results**, expand the loan and select the `status` attribute as we want to test it. Then, click on **Generate**.

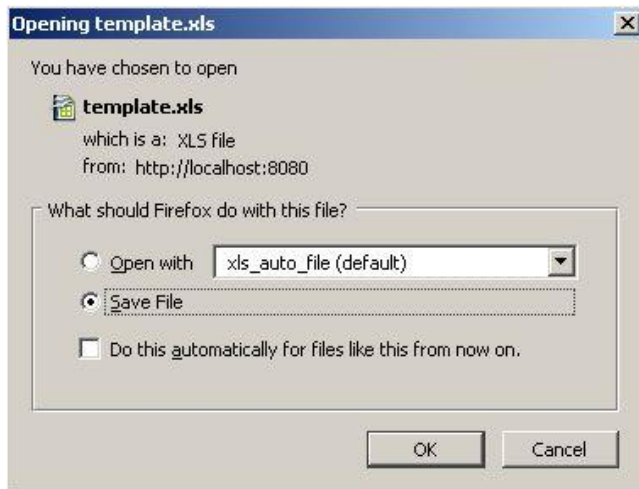
Generate Scenario File Template

Locale
Select the display locale: ▼

Expected Values
Select the columns to include in the Expected Results sheet and in the Expected Execution Details sheet

Select/Unselect All		
<div style="background-color: #e6eef3; padding: 2px;"> Expected Results </div>		
<div style="background-color: #e6eef3; padding: 2px;"> the loan </div>		
<input type="checkbox"/>	amount	<input type="text"/>
<input type="checkbox"/>	duration	<input type="text"/>
<input type="checkbox"/>	messages	<input type="text"/>
<input checked="" type="checkbox"/>	status	<input type="text"/>
<input type="checkbox"/>	yearly interest rate	<input type="text"/>
<input type="checkbox"/>	yearly repayment	<input type="text"/>
<div style="background-color: #e6eef3; padding: 2px;"> Expected Execution Details </div>		
<input type="checkbox"/>	The list of rules fired	<input type="text"/>
<input type="checkbox"/>	The list of executed ruleflow tasks	<input type="text"/>
<input type="checkbox"/>	The duration (in ms) of execution	<input type="text"/>

- __8. Click the generated `template.xls` to save it. The Excel file will be saved by default on the Desktop.



Note: Make sure to save it. Otherwise, opening it will save the excel file in a temporary folder and will open it in Read-Only mode instead.

- __9. Open the saved Excel file. Fill in the scenario file template as follows:

a) **Scenarios** sheet:

Scenario ID	description	the borrower			the loan		
		credit score	name	yearly income	amount	duration	yearly interest rate
1	rejected loan	600	Joe	80000	500000	240	0.05
2	accepted loan	600	Joe	80000	250000	240	0.05

Note: To add rows, copy and paste the first one.

- b) Copy/paste the scenarios ID between the worksheet **Scenarios** and **Expected Results** to make sure the ID are the same between both worksheets. This should look lie this:

Scenario ID	the status of the loan equals
1	Rejected
2	<div style="border: 1px solid black; padding: 2px;"> Accepted Referred Rejected </div>

- __10. Save the file.

- __11. Go back to Decision Center, click on **Back** button.
- __12. In **Step 3: Scenarios**, click **Browse** in the File section, and select `Desktop/template.xls` to upload the scenario file you created. You should see the message: *File Uploaded*

Scenarios

Format: Excel (2003)

Generate template: Generate a scenario file to upload below when finished

File: template.xls (File uploaded)

Browse...

- __13. Click **Finish and Run**.
- __14. On the **Run** page, make sure the check box `The list of rules fired` is selected, and click **Run**.

Run Miniloan Test - Version: 1.0

Server: Rule Server

Report Options:

Name: Miniloan Test - Report

Create an Excel file separate from the report to store the output values

In addition to any tests specified in the scenario file, you can add in the report:

The list of rules fired

The list of executed ruleflow tasks

The duration (in ms) of execution

Cancel Run

Decision Center may take a few seconds to complete this action.

The report opens and shows the results of the tests and the list of rules executed:

Summary

Number of scenarios 2

Scenarios with failures 1

Success Rate 50%

Details

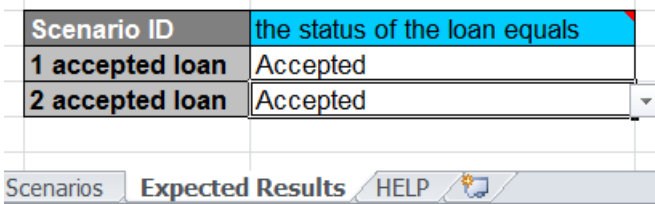
Name	Status												
1 rejected loan	[Failure]												
	<table border="0" style="width: 100%;"> <tr> <td style="width: 30%;">Status Test</td> <td style="width: 30%;">the status of the loan equals</td> <td style="width: 40%;">Message</td> </tr> <tr> <td></td> <td></td> <td>The observed value "Accepted" does not equal the expected value "Rejected"</td> </tr> <tr> <td colspan="3">The list of rules fired</td> </tr> <tr> <td colspan="3">computation.repayment computation.debt-to-income eligibility.repayment and score 4</td> </tr> </table>	Status Test	the status of the loan equals	Message			The observed value "Accepted" does not equal the expected value "Rejected"	The list of rules fired			computation.repayment computation.debt-to-income eligibility.repayment and score 4		
Status Test	the status of the loan equals	Message											
		The observed value "Accepted" does not equal the expected value "Rejected"											
The list of rules fired													
computation.repayment computation.debt-to-income eligibility.repayment and score 4													
2 accepted loan	[Success]												
	<table border="0" style="width: 100%;"> <tr> <td style="width: 30%;">Status Test</td> <td style="width: 30%;">the status of the loan equals</td> <td style="width: 40%;">Message</td> </tr> <tr> <td></td> <td></td> <td>The observed value "Accepted" is the expected value</td> </tr> <tr> <td colspan="3">The list of rules fired</td> </tr> <tr> <td colspan="3">computation.repayment computation.debt-to-income eligibility.repayment and score 2</td> </tr> </table>	Status Test	the status of the loan equals	Message			The observed value "Accepted" is the expected value	The list of rules fired			computation.repayment computation.debt-to-income eligibility.repayment and score 2		
Status Test	the status of the loan equals	Message											
		The observed value "Accepted" is the expected value											
The list of rules fired													
computation.repayment computation.debt-to-income eligibility.repayment and score 2													

Scenario 1 fails because it expects the loan to be rejected. However, the loan is now approved because you have increased the allowable debt-to-income ratio in the `minimum income` rule.

1.2.2 Modify the Excel scenario file


To correct the problem in **Scenario 1**, you edit `miniloan-test.xls` and modify the expected results for **Scenario 1**. You also add a new scenario to verify that the loan is rejected if the debt-to-income ratio is too high.

- __1. Open `Desktop/template.xls`.
- __2. Click the **Expected Results** tab, and change the scenario ID and the expected result for **Scenario 1** to `Accepted`.




- __3. Click the **Scenarios** tab, copy and paste the Scenario 1 row, to create a third scenario and change the yearly income to 40000. You can also update the description as follows:

Scenario ID	description	the borrower			the loan		
		credit score	name	yearly income	amount	duration	yearly interest rate
1	accepted loan	600	Joe	80000	500000	240	0.05
2	accepted loan	600	Joe	80000	250000	240	0.05
3	rejected loan	600	Joe	40000	500000	240	0.05

Scenarios / Expected Results / HELP 

- __4. Click the **Expected Results** tab, add a new row for **Scenario 3**, and enter Rejected as expected result.

Scenario ID	the status of the loan equals
1	accepted loan
2	accepted loan
3	rejected loan

Scenarios / Expected Results / HELP 

- __5. Save the changes and close the file.

1.2.3 Edit and run the test suite

To test the scenarios that you modified, you must edit the test suite and run it again.

- __1. Make sure you are signed in to Decision Center as `rtsUser1` and that the **Project in use** is set to `Miniloan Rules`.
- __2. Click the **Explore** tab, and click **Test Suites** to the left.
- __3. Next to the `Miniloan Test` suite, click **Quick Edit** as follows:

<input checked="" type="checkbox"/> Actions	Name	Last Changed By
<input checked="" type="checkbox"/>   	Miniloan Test	rtsAdmin

__4. In **File** section, click **Choose File** to upload the updated excel scenario.

Test Suite Editing
 Save | Cancel

Name*

By default all rules of the current branch will be tested
 Or you can test:
 rules from a baseline of this branch:
 rules selected from:

For this ruleset, the entry point is:

Format:

[Generate template:](#) Generate a scenario file template for this test suite that

File: *(If you edit this file do not forget to upload it below when finished)*

__5. Select the Desktop/template.xls to upload the scenario file that you modified, and make it available to Decision Center.

__6. Click **Save** and click on **Run** button as follows:

<input checked="" type="checkbox"/>	Actions	Name	Last Changed By
<input checked="" type="checkbox"/>		Miniloan Test	rtsAdmin

__7. On the **Run** page, make sure the check box **The list of rules fired** is selected, and click **Run**.

Run Miniloan Test - Version: 1.3

Server:

Report Options:
 Name

Create an Excel file separate from the report to store the output values

In addition to any tests specified in the scenario file, you can add in the report:

- The list of rules fired
- The list of executed ruleflow tasks
- The duration (in ms) of execution

The report opens, the tests of the three scenarios are successful: the expected results match the execution results.

Details

Name	Status
1 accepted loan	
2 accepted loan	
3 rejected loan	

Status	Test	Message
	the status of the loan equals	The observed value " Rejected " is the expected value

The list of rules fired [computation.repayment](#)
[computation.debt-to-income](#)
[eligibility.repayment and score 7](#)

If you expand the third scenario, you notice that the row 7 of the decision table `repayment and score` was executed. You can click on this hyperlink to open the decision table.

After testing the ruleset in Decision Center, users with the appropriate role can deploy the changes to Rule Execution Server. You will do this in the next task.

1.3 Deploy the rules

In this task, you deploy your ruleset directly from Decision Center to Rule Execution Server.

Users with the correct access rights can deploy rulesets directly from Decision Center.

You can deploy both from Rule Designer and Decision Center. You will create a RuleApp, a container for your ruleset, and deployed the RuleApp to Rule Execution Server so that it could be executed.

In this task you will do the same thing from Decision Center, and then see how the new rules affect the Miniloan application.



This task should take you about 20 minutes to complete.

- Step 1: Deploy the rules
- Step 2: See the effects in the Miniloan application

1.3.1 Deploy the rules

Because some of the rules were already deployed, you will increment the RuleApp version. Later you will see how Rule Execution Server manages these changes.

You will now deploy the rules to the Rule Execution Server. Because some rules may have already been deployed, you will increment the ruleapp version.

To deploy the RuleApp:

__1. If you are not logged to Decision Center, login using the following credentials:

Username: rtsUser1

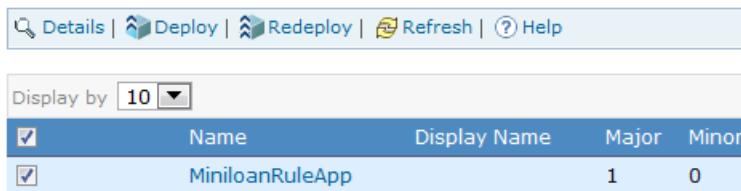
Password: rtsUser1

__2. In the Decision Center Home page, change the **Project in use** to `Miniloan Rules`.

__3. Click the **Project** tab, and then click **Deploy RuleApps**.

__4. Under **Available RuleApps**, select the RuleApp `MiniloanRuleapp`.

Available RuleApps



__5. Click **Deploy** in the toolbar.

__6. On the Deployment Name page, type `ITERATION 1`, and then click **Next**.

This creates a deployment baseline called `ITERATION 1`, that is, a snapshot in Decision Center of the version of each element about to be deployed. A baseline captures the changes that occur in a project over time, and allows you to revert back to one of the rules if required.

__7. On the RuleApp target page, select **Deploy on a Rule Execution Server**, and then click **Next**.

__8. On the Versioning Policy page, keep the default **Increment RuleApp major version**.



This increments the RuleApp version to `2.0` if one ruleapp was already deployed.

__9. Click **Next**.

__10. On the Server List page, select the Rule Server.

__11. Click **Deploy**.

When processing is complete, you see the `Deployment Succeeded` message.

Deployment Succeeded

RuleApp archive deployed MiniloanRuleApp
Versioning Policy Increment RuleApp major version
Rule Execution Server http://localhost:9081/res

Archive content	Operation	Result
/MiniloanRuleApp/1.0	+ Version changed and element added	/MiniloanRuleApp/ 2.0
/MiniloanRules/1.0	+ Version changed and element added	/MiniloanRules/1.0

List of deployment baselines created

Baseline Name	Project	RuleApp	Ruleset
ITERATION 1	Miniloan Rules	MiniloanRuleApp	MiniloanRules

__12. Sign out of Decision Center.

1.3.2 See the effects in the Miniloan application

Finally, you will see how the business policy changes you made are reflected back into the Miniloan application.

To see the effects in Miniloan:

__1. Start Miniloan by entering the following URL in a browser:

`http://localhost:9081/miniloan-server`

- __2. In Miniloan, make sure to check the option **Use Rules** and click **Validate Loan** with the default values.

Miniloan Validation

Borrower Information		Loan Information	
Name:	Joe	Amount:	500000
Yearly Income:	80000	Duration (months):	240
Credit Score:	600	Yearly Interest Rate :	0.05

Use Rules

Validate Loan

Congratulations! The loan is approved.

Reasons:
 Debt-to-income ratio is 31%
 Yearly repayment is \$25,250

Rules Execution Summary	
3 rule(s) fired	1: repayment in rule task: miniloan#computation 2: debt-to-income in rule task: miniloan#computation 3: repayment and score 4 in rule task: miniloan#eligibility

[Execution details](#)

This time, the loan is approved because the policy change for borrowers with an income of 500,000 has been implemented.

- __3. You can use Decision Warehouse to access the execution details for a decision and view the rules that were fired in the Rule Execution Server. To do so, click on the hyperlink `Execution Details`.

- __4. The console of Rule Execution Server will open in a new tab. Login using the following credentials:

Username: resAdmin









Password: resAdmin

- __5. The Home page of the console opens. If you click again on the hyperlink of the result page, then it will open the **Execution Details** as follows:

Execution Details

Decision ID: 795009d7-2d9c-4989-bc77-a5a84abc65d9
Date: 2012-01-04 19:07:55
Executed ruleset path: /MiniloanRuleApp/2.0/MiniloanRules/1.0
Processing Time (ms) 154
Number of rules fired 3
Number of tasks executed 4

Decision Trace

- [-]  Ruleflow Tasks (1)
 - [-]  miniloan (3)
 -  miniloan>validation
 - [-]  miniloan>computation (2)
 -  computation.repayment
 -  computation.debt-to-income
 - [-]  miniloan>eligibility (1)
 -  eligibility.repayment and score 4


Input Parameters

loan :

By expanding the tree view of the Ruleflow Tasks, you can see the path and the rules executed to make the decision.

Each rule is displayed as a hyperlink to allow you to access the rule content. When the rule is actually a decision table, a number is displayed at the end of the rule name to indicate which row was executed.

- __6. Click on the decision table `eligibility.repayment and score 4`. This will open the following screen:

 Table

if
the status of 'the loan' is not "Rejected"

	Debt-to-income %		credit score		Loan Status	Reason
	min	max	min	max		
1			0	200	Rejected	debt-to-income too high compared to credit score
2	0	30	200	800	Accepted	
3			0	400	Rejected	debt-to-income too high compared to credit score
4	30	45	400	800	Accepted	
5			0	600	Rejected	debt-to-income too high compared to credit score
6	45	50	600	800	Accepted	
7	Otherwise		0	800	Rejected	debt-to-income too high compared to credit score

You have now deployed your ruleset to Rule Execution Server and see the effects on the web application while auditing which decision was made.


1.4 Author in Rule Solutions for Office

In this task, you use Rule Solutions for Office to edit rules using the Microsoft Office.

You can edit business rules directly in Decision Center, but you can also use Rule Solutions for Office to edit rules offline with the Microsoft Office tools that you are already familiar with.

In this task, you edit an action rule and a decision table, and then you update the changes in Decision Center.

Note: This task is optional. To do this task, you must have Rule Solutions for Office installed.

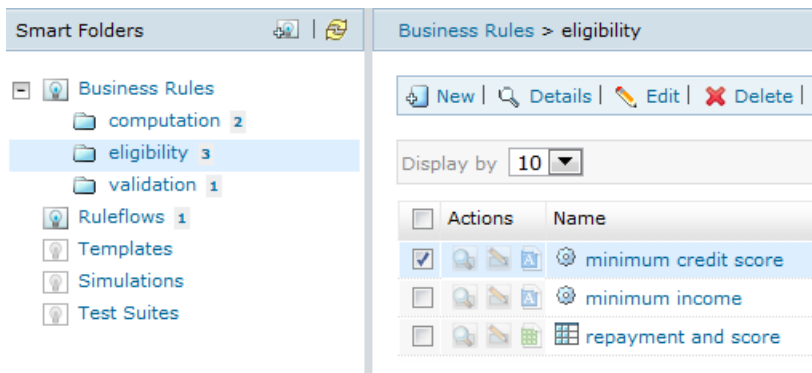
 This task should take you about 20 minutes to complete.

- Step 1: Modify an action rule in RuleDocs
- Step 2: Modify a decision table in a RuleDocs

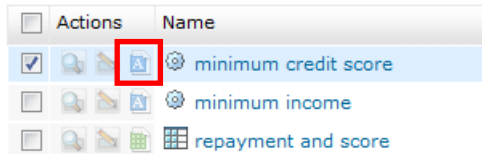
1.4.1 Modify an action rule in RuleDocs

First, you will edit a rule. Because some of the rules were already deployed, you will increment the RuleApp version to 1.1.

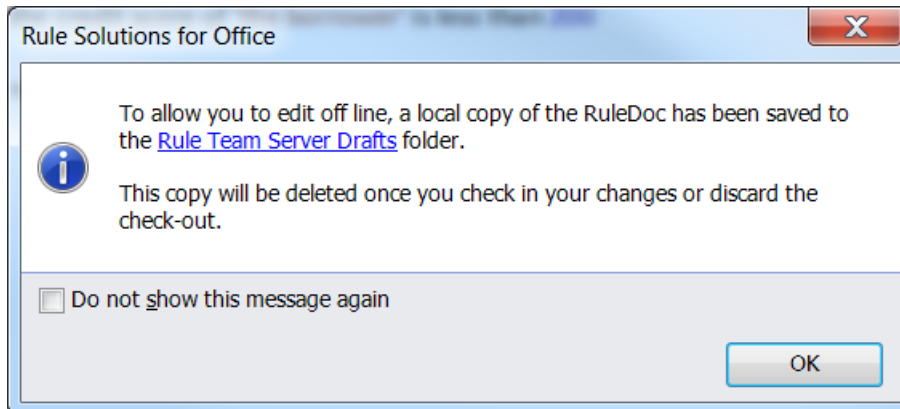
1. In **Smart Folders**, click the `eligibility` folder and, then select the `minimum credit score` rule.



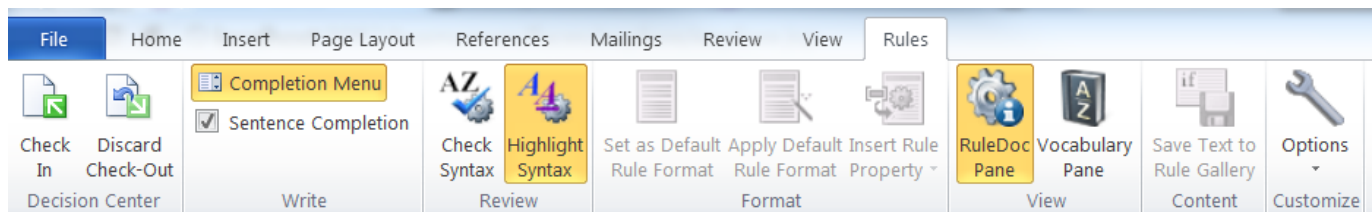
2. Click on the Edit in Rule Solutions for Office icon to open the rule in Word.



- __3. A Word document called `minimum credit score.docx` will be saved temporarily on your machine. Open this document.



- __4. If the RuleDoc pane is not visible, on the Rule tab, in the View group, click **RuleDoc** Pane to display it.



- __5. In the rule, replace both values 200 of the credit score with 300.

minimum credit score

Package: eligibility

if

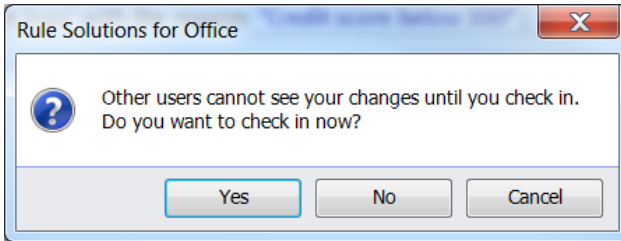
the credit score of 'the borrower' is less than 300

then

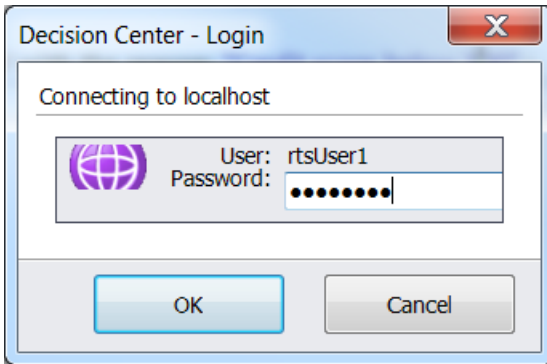
reject 'the loan' with the reason: "Credit score below 300" ;

- __6. On the Rule tab, in the View group, click the **Vocabulary** Pane to display the full business object model.

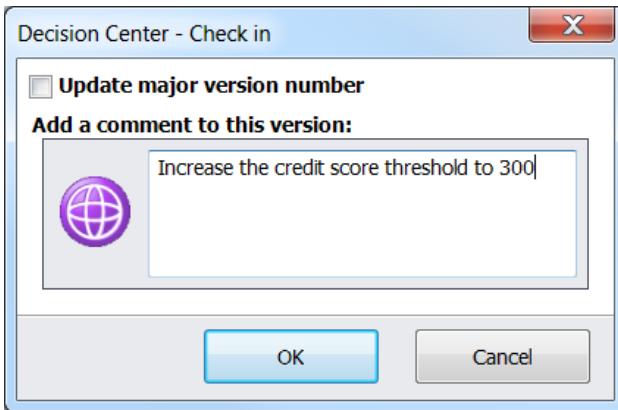
__7. Close the document. Then, answer **Yes** to “Do you want to check in now?”



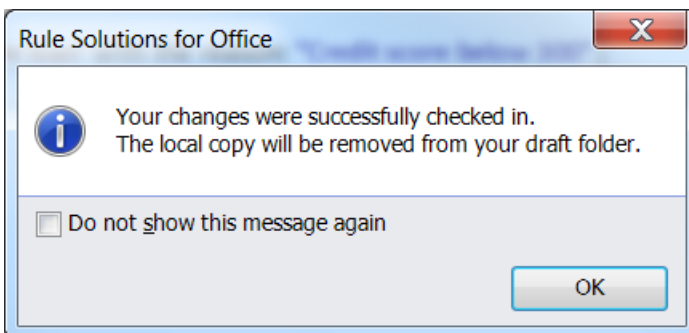
When asked to enter a password, type: `rtsUser1`.



Finally, you can enter a comment to this version as follows:



The temporary copy will be removed from the draft folder.

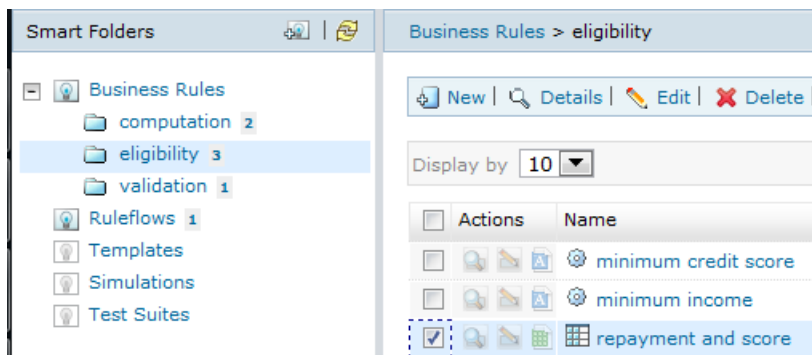


1.4.2 Modify a decision table in RuleDocs

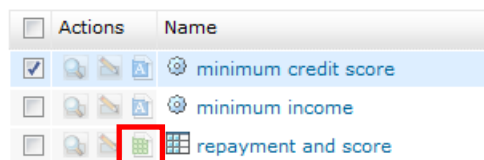
You are asked to make your lending criteria less restrictive, so that you reject a loan for someone whose debt-to-income ratio is between 45 and 50 only if the applicant's credit score is less than 500. To do this, you modify the repayment and score decision table.

To modify a decision table and correct any resulting errors:

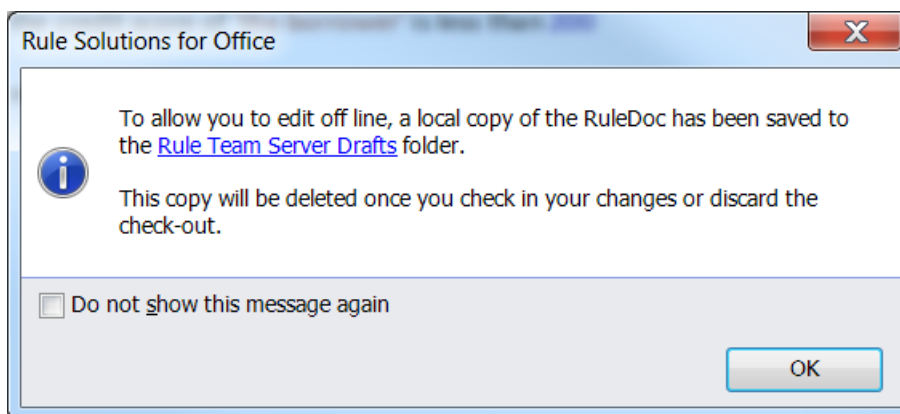
- __1. In **Smart Folders**, click the `eligibility` folder and, then select the decision table `repayment and score` rule.



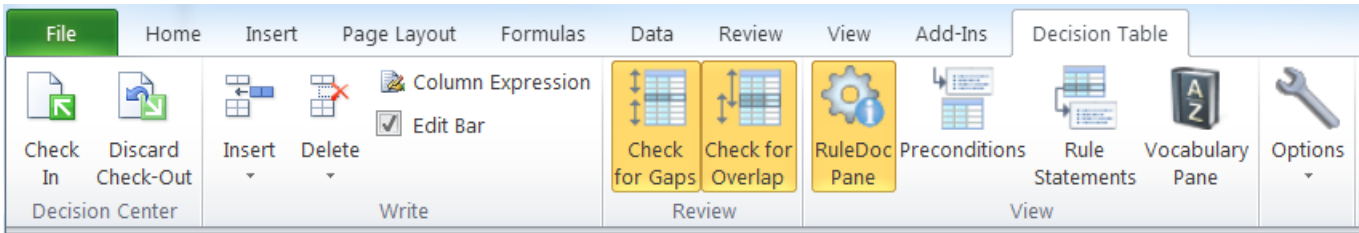
- __2. Click on the Edit in Rule Solutions for Office icon to open the rule in Excel.



- __3. A Word document called `repayment and score.xlsx` will be saved temporarily on your machine. Open this document.



- __4. If the RuleDoc pane is not visible, on the Rule tab, in the View group, click **RuleDoc** Pane to display it.



__5. Under credit score, click the cell in the max column that contains 200. Replace 200 with 300, and then press Enter.

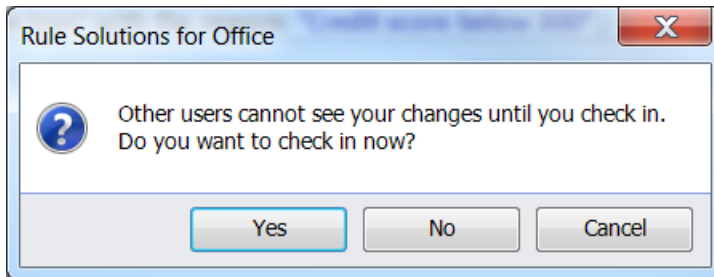
Debt-to-income %		credit score		Loan Status	Reason
min	max	min	max		
0	30	0	300	Rejected	debt-to-income too high compare
		200	800	Accepted	
30	45	0	400	Rejected	debt-to-income too high compare
		400	800	Accepted	
45	50	0	600	Rejected	debt-to-income too high compare
		600	800	Accepted	
Otherwise		0	800	Rejected	debt-to-income too high compare

Notice how some cells have a yellow background. This indicates that there is an error. In the Problem List in the RuleDoc pane, you can see exactly what the error is: Rows have gaps.

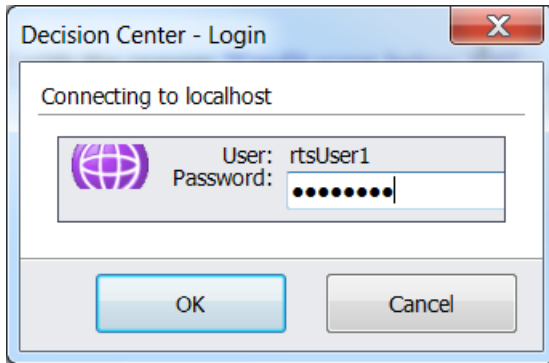
__6. To solve this problem, click the cell in the min column under credit score containing 200, and replace 200 with 300, and then press Enter.

The problem is no longer displayed in the Problem List in the RuleDoc pane.

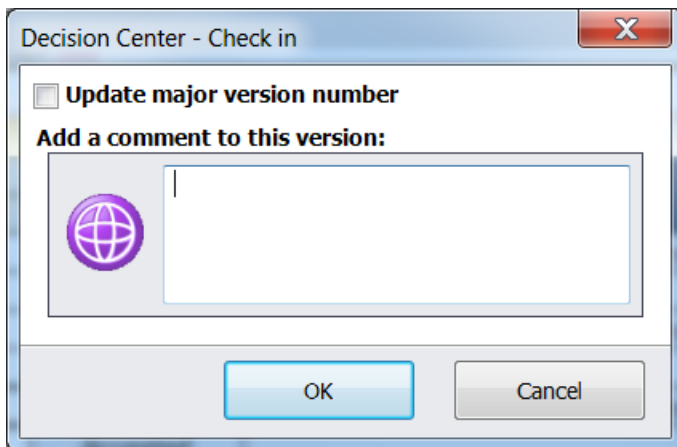
__7. Close the document. Then, answer **Yes** to “Do you want to check in now?”



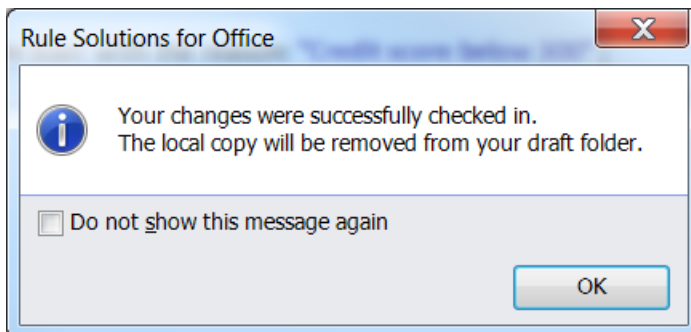
When asked to enter a password, type: `rtsUser1`.



Finally, leave the comment to this version empty and click Ok:



The temporary copy will be removed from the draft folder.



You are now familiar with Rule Solutions for Office to edit rules offline with the Microsoft Office tools.

Summary

In this lab you used Decision Center to perform the following business analyst tasks:

- Created and executed a rule query.

- Edited a rule and compared different rule versions.
- Created a new rule.
- Tested and simulated the rules by leveraging Decision Validation Services.
- Created a new RuleApp and deployed it to Rule Execution Server.

1.5 Monitoring the rules (Optional)

In this task, you use the Decision Warehouse to audit and view stored decisions.

Auditors can analyze the execution performance of your rulesets, and troubleshoot any problematic transaction that may be reported. To identify the problem when a transaction fails, auditors and analysts need to know the business policies that were applied, and the transactional data that was used at execution time.

Decision Warehouse is available from Rule Execution Server and stores ruleset execution traces that can be used for auditing purposes.

In this task, you see how to trace a transaction and view the execution report.



This task should take you about 15 minutes to complete.

- *Step 1: Search for past transactions in Decision Warehouse*
- *Step 2: View the rules fired*

1.5.1 Search for past transactions in Decision Warehouse

When the RuleApp was configured, the `monitoring.enabled` ruleset property was set to monitor the execution of the ruleset in Decision Warehouse. Every transaction you simulate for the miniloan ruleset is now stored and logged in Decision Warehouse.

To search for past transactions:

- __1. Sign in to Rule Execution Server using the following details:

Enter the following URL in a browser:

```
http://localhost:9081/res
```

Username: resAdmin

Password: resAdmin

- __2. Click the **Decision Warehouse** tab.
- __3. On the Search Decisions page, leave the fields blank, and click **Search**.

Decision Warehouse displays the decisions for the transactions that you have executed in the Miniloan application using the latest version of the ruleset. For example, the decision for the transaction executed when testing the web service shows the date and the processing time, and indicates that three rules

were fired.

1 Decision(s) found Display by: 10					
Decision ID	Date	Ruleset Version	Number of rules fired	Decision Trace	Processing Time (ms)
8f02c11a-d5b9-49ec-ab1e-368de19ac83f	2012-01-06 17:39:54	/MiniloanRuleApp/2.0/MiniloanRules/1.0	4	View Decision details	11

1 - 1 out of 1 results

1.5.2 View the rules fired

You can use Decision Warehouse to understand why the loan has been rejected. You check the execution details for a decision and view the rules that were fired.

To view the execution details for a decision:

- ___1. In the table that lists the decisions found, in the Decision Trace column, click the hyperlink **View Decision details** for the decision where one rule was fired. The decision trace details open in a new window.
- ___2. In the Decision Trace section, expand Ruleflow Tasks to see which rules were executed.

Execution Details

Decision ID: 8f02c11a-d5b9-49ec-ab1e-368de19ac83f

Date: 2012-01-06 17:39:54










Executed ruleset path: /MiniloanRuleApp/2.0/MiniloanRules/1.0

Processing Time (ms): 11

Number of rules fired: 4

Number of tasks executed: 4

Decision Trace

- [-]  Ruleflow Tasks (1)
 - [-]  miniloan (3)
 -  miniloan>validation
 - [-]  miniloan>computation (2)
 -  computation.repayment
 -  computation.debt-to-income
 - [-]  miniloan>eligibility (2)
 -  eligibility.repayment and score 1
 -  eligibility.minimum credit score

- ___3. At the bottom, you can also view the input and output parameters as well as the execution output:

Input Parameters

loan :

```

<q1:loan xmlns:q1="miniloan">
  <amount>1000</amount>
  <duration>12</duration>
  <yearlyInterestRate>2.75</yearlyInterestRate>
</q1:loan>

```

borrower :

```

<?xml version="1.0" encoding="UTF-8"?>
  <q1:borrower xmlns:q1="miniloan" xmlns:q2="http://www.ibm.com/rules/decisions"
  <name>Joe</name>
  <creditScore>100</creditScore>
  <yearlyIncome>10000</yearlyIncome>
  </q1:borrower>

```

Output Parameters

loan :

```

<ns0:loan xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:ns0="miniloan">

```

Execution output

Rejected

You have now seen how to discover which rules were fired, and report on the time it took to execute them, as well as the inputs, outputs and execution for each decision.

Appendix A. Notices

This information was developed for products and services offered in the U.S.A.

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
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106-0032, Japan

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 Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites 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.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have

been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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. All references to fictitious companies or individuals are used for illustration purposes only.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs.

Appendix B. Trademarks and copyrights

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

IBM	AIX	CICS	ClearCase	ClearQuest	Cloudscape
Cube Views	DB2	developerWorks	DRDA	IMS	IMS/ESA
Informix	Lotus	Lotus Workflow	MQSeries	OmniFind	ILOG
Rational	Redbooks	Red Brick	RequisitePro	System i	
<i>System z</i>	<i>Tivoli</i>	<i>WebSphere</i>	<i>Workplace</i>	<i>System p</i>	

Adobe, Acrobat, Portable Document Format (PDF), and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, other countries, or both.

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.

Java and all Java-based trademarks and logos are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both. See Java Guidelines

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

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.

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

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

ITIL is a registered trademark and a registered community trademark of the Office of Government Commerce, and is registered in the U.S. Patent and Trademark Office.

IT Infrastructure Library is a registered trademark of the Central Computer and Telecommunications Agency which is now part of the Office of Government Commerce.

Other company, product and service names may be trademarks or service marks of others.



© Copyright IBM Corporation 2011.

The information contained in these materials is provided for informational purposes only, and is provided AS IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, these materials. Nothing contained in these materials is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software. References in these materials to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates. This information is based on current IBM product plans and strategy, which are subject to change by IBM without notice. Product release dates and/or capabilities referenced in these materials may change at any time at IBM's sole discretion based on market opportunities or other factors, and are not intended to be a commitment to future product or feature availability in any way.

IBM, the IBM logo and ibm.com are trademarks or registered trademarks of International Business Machines Corporation in the United States, other countries, or both. If these and other IBM trademarked terms are marked on their first occurrence in this information with a trademark symbol (® or ™), these symbols indicate U.S. registered or common law trademarks owned by IBM at the time this information was published. Such trademarks may also be registered or common law trademarks in other countries. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at ibm.com/legal/copytrade.shtml

Other company, product and service names may be trademarks or service marks of others.



Please Recycle
