How to Use Rollbase as an Application

for

Managing Parameterised Rules

in Corticon

Mike Parish

1/28/2014

The Use Case

In order to put rule writing into the hands of the business users there are a couple of approaches:

1. Give them a copy of Corticon Studio and allow them to modify the rules – this assumes that they are skilled in rule modeling using Corticon. It also does not provide a way to control what parts of a rule model an individual is allowed to change.
2. Give them a simplified interface which limits and controls what part of which rules they can change – typically this interface allows a business user to change the values or thresholds that are used in the underlying Corticon rule model. This is referred to as "Parameterized Rules"

In this document we'll explore approach 2 – Parameterized Rules.

There are several options for the Parameterized Rules approach:

1. We could simply allow the business users to go to the database holding the parameters and change them. The danger of course is that there are no controls over what values they might enter.
2. We could create a java application that manages the "rules" or "parameters" – but this would involve a significant amount of development effort.
3. We could use Rollbase Rapid Application Development to create a parameter management system. In this approach we can then use Corticon to help validate the parameters that are entered by the user.

In this document we'll show how to accomplish option #3.

The Rollbase Solution

CorticonParameters +

All CorticonParameters | New CorticonParameter | All CorticonParameters | Filter

Select | More actions...

Action	CorticonParameter	Classification	Lower	Upper	Units
Edit | Del	Lathosterol	High	160.00	99,999.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Lathosterol	Borderline	90.00	160.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Lathosterol	Optimal	0.00	90.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Desmosterol	Borderline	90.00	120.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Desmosterol	Optimal	0.00	90.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Desmosterol	High	120.00	99,999.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Cholestanol	Borderline	140.00	215.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Cholestanol	High	215.00	999.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Cholestanol	Optimal	0.00	140.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Campesterol	High	250.00	999.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Campesterol	Optimal	0.00	150.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Campesterol	Borderline	150.00	260.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Beta-Sitosterol	High	180.00	9,999.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Beta-Sitosterol	Optimal	0.00	100.00	µmol x 100/mmol of Total Cholesterol
Edit | Del	Beta-Sitosterol	Borderline	100.00	180.00	µmol x 100/mmol of Total Cholesterol

Detailed Edit Screen

CorticonParameters +

CorticonParameter: Lathosterol

Save | Save & New | Cancel

CorticonParameter Information

CorticonParameter: Lathosterol
Classification: High
Lower: 160.00
Upper: 99,999.00
Units: µmol x 100/mmol of Tota
Comment:

Workflow Information

Workflow Status: Created

Organization

Location:
Department:
Function:

Save | Save & New | Cancel

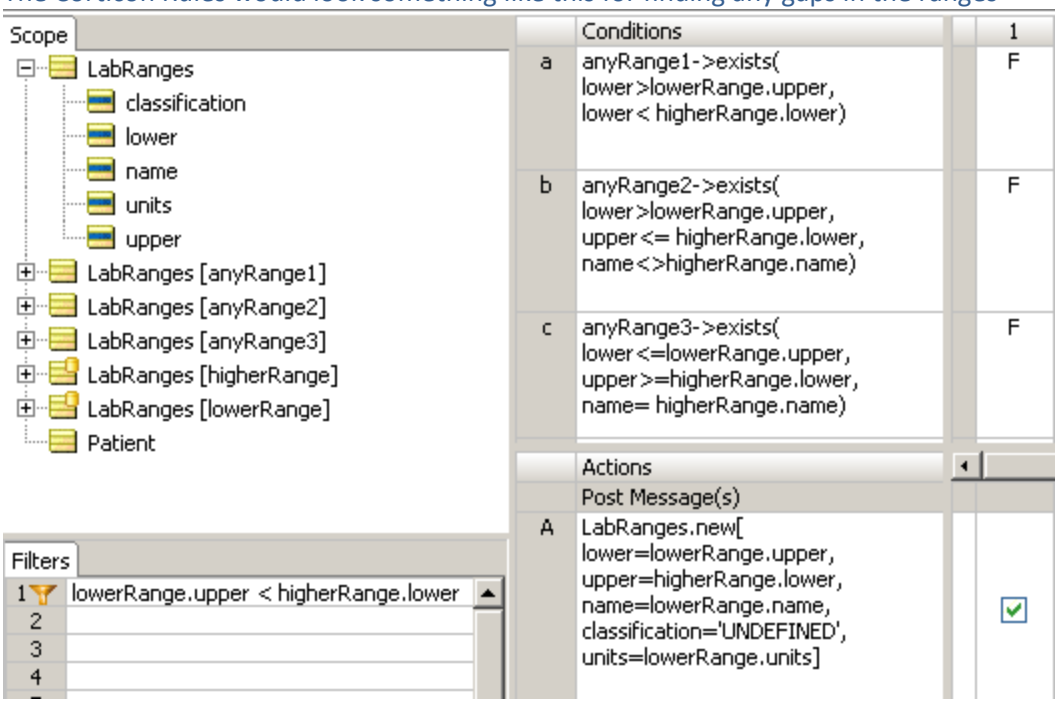
Using Corticon to Validate the Parameter Values

Whenever a change is made to any parameter, Corticon needs to be invoked (via a trigger) to perform some validation checks. Corticon would need to retrieve ALL the records from the Rollbase database corresponding to the parameter name that had been modified. So the template for the data to be passed to Corticon would simply have the name of the parameter:

1. Is the value valid
2. Are there overlaps with other parameter values (ambiguity)
3. Are there gaps in the ranges (incompleteness)

Using Corticon to Find Gaps in the Parameter Ranges

The Corticon Rules would look something like this for finding any gaps in the ranges



Using Corticon to Find Ambiguities in the Parameters

In order to find ambiguities in the ranges the rules might look like this:

Scope

- LabRanges [R1]
  - Filters
  - classification
  - lower
  - name
  - upper
- LabRanges [R2]
  - Filters
  - classification
  - lower
  - name
  - upper

Filters

1	R1 <> R2
2	R1.lower <= R2.lower
3	R1.name = R2.name
4	

	Conditions	1	2	3	4	5	6
a	R1.lower < R2.lower	T	-	T	T	F	F
b	R1.upper <= R2.lower	T	-	F	F	-	-
c	R1.lower >= R2.upper	-	T	T	F	T	F
d	R1.upper > R2.upper	-	T	F	-	F	-
e	R1.classification = R2.classification	F	F	F	F	F	F
f							
g							
h							
i							
j							
k							
l							
m							

	Actions						
	Post Message(s)						
A							
B							
C							
D							
E							
	Overrides						

Rule Statements | Rule Messages | Natural Language | Properties | History

Ref	Post	Alias	Text	Rule Name	R
	Info	R1	OK: {R1.lower}--{R1.upper} is below {R2.lower}--{R2.upper}		
	Info	R1	OK: {R1.lower}--{R1.upper} is above {R2.lower}--{R2.upper}		
3:6	Violation	R1	Range Conflict for {R1.name}: {R1.lower}--{R1.upper} ({R1.classification}) overlaps {R2.lower}--{R2.upper} ({R2.classification})		

Using the Parameters in a Corticon Rule Sheet

Then in the application that actually needs to apply the business rules that depend on these parameters you might see rule sheets like this:

Scope

- Change
- LabRanges [range]
  - Filters
  - classification
  - lower
  - name
  - units
  - upper
- Patient
  - Filters
  - labs (LabResult) [lab]
    - Filters
    - classification
    - name
    - ratioToTC

Filters

1	range.name = lab.name
2	
3	
4	

	Conditions	0	1
a	lab.ratioToTC > range.lower		T
b	lab.ratioToTC< range.upper		T
c			

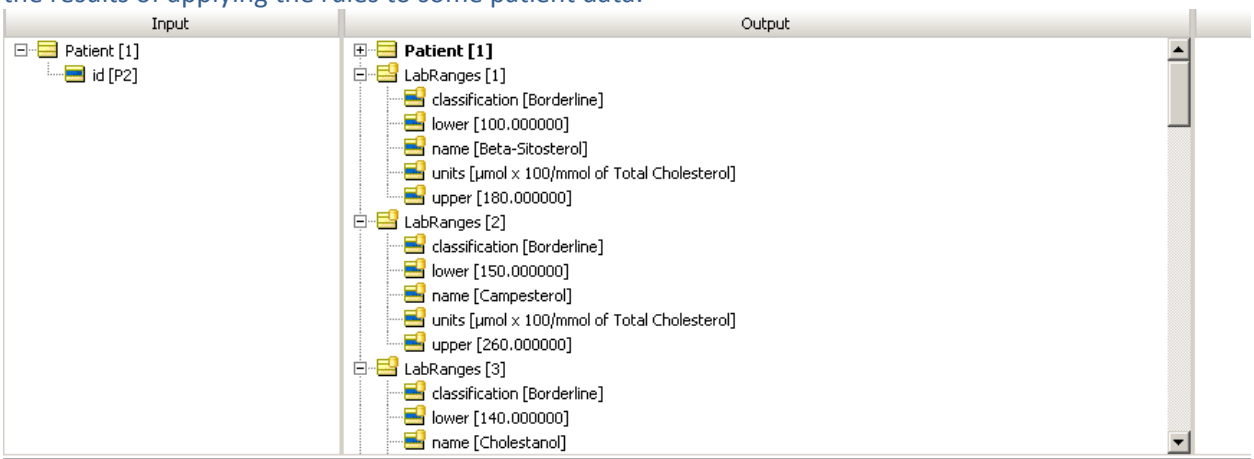
	Actions		
	Post Message(s)	✉	✉
A	lab.classification = range.classification	☐	☑
B	Change.new[who='MEP', what='Read classification ranges from database', why='Illustrate DB access technique', where='Lab Ranges.ers', when='2013/7/30']	☑	☐
C			
D			
E			
F			
G			
H			
I			
J			
K			
	Overrides		

Rule Statements | Rule Messages | Natural Language | Properties | History

Ref		Post	Alias	Text
1		Info	range	Lab value {lab.ratioToTC} for {lab.name} falls in the {range.classification} range. ({range.lower} - {range.upper} {range.units})
0		Info	Patient	NOTE: Patient Lipid lab values and the Classification ranges for the lipids are obtained from the database in this version

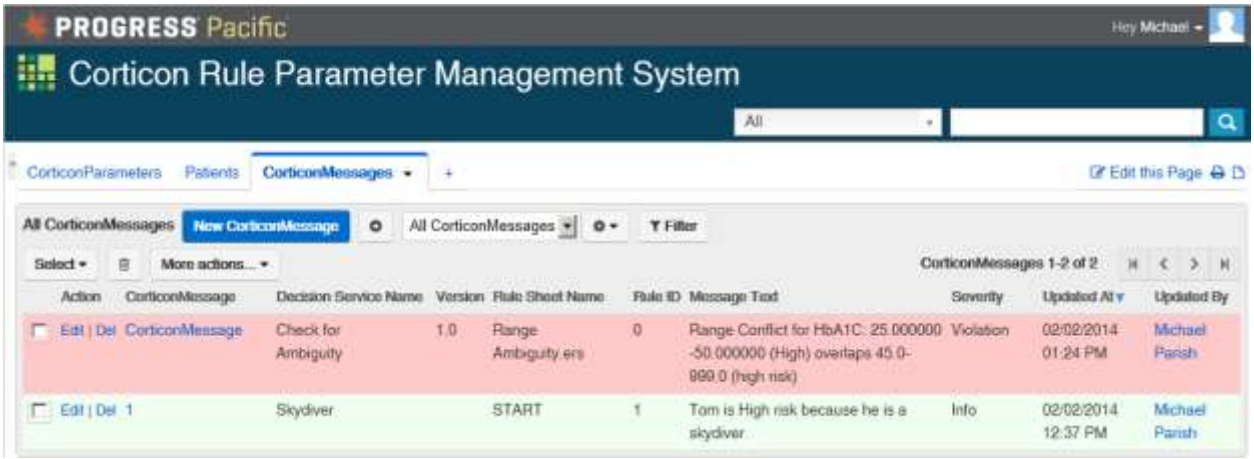
A Test Case

Here's a sample test case showing both the detection of problems with the database parameters and the results of applying the rules to some patient data:

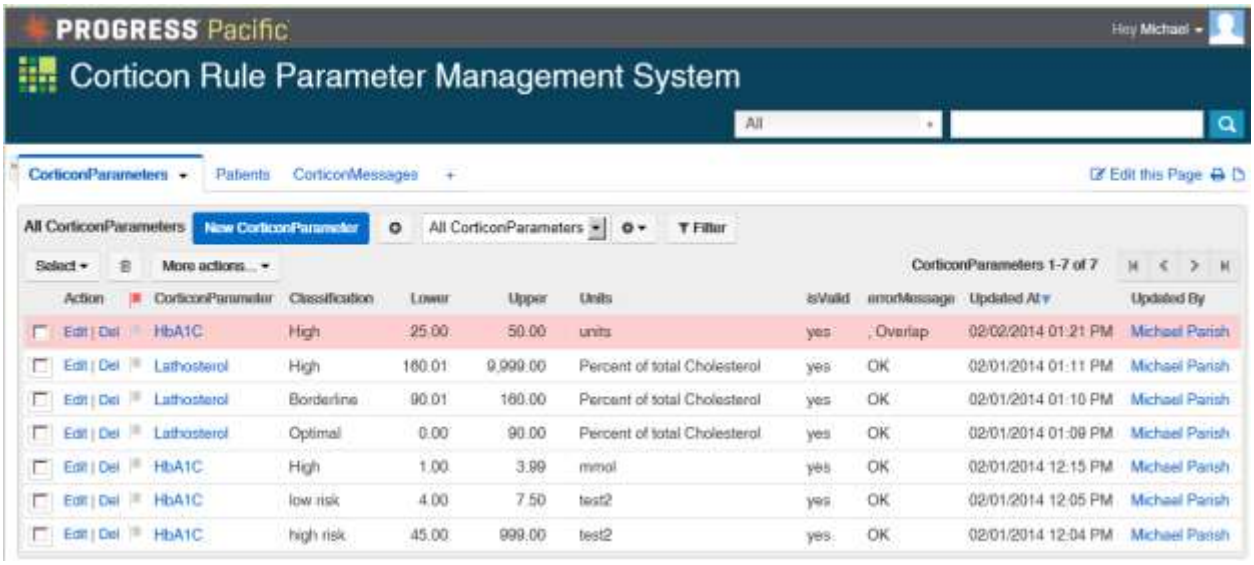


Severity	Message	Entity
Violation	Range Conflict for Campesterol: 150.000000-260.000000 (Borderline) overlaps 250.000000-999.000000 (High)	LabRanges[2]
Info	Lipid ratios calculated	Patient[1]
Info	NOTE: Classification ranges for the lipids are obtained from the database in this version	Patient[1]
Info	Lab value 120.000000 for Lathosterol falls in the Borderline range. (90.000000 - 160.000000 µmol x 100/mmol of Total Cholesterol)	LabRanges[5]
Info	Lab value 80.000000 for Cholestanol falls in the Optimal range. (0.000000 - 140.000000 µmol x 100/mmol of Total Cholesterol)	LabRanges[13]
Info	Lab value 160.000000 for Desmosterol falls in the High range. (120.000000 - 99999.000000 µmol x 100/mmol of Total Cholesterol)	LabRanges[9]
Info	Lab value 154.000000 for Beta-Sitosterol falls in the Borderline range. (100.000000 - 180.000000 µmol x 100/mmol of Total Cholesterol)	LabRanges[1]
Info	Lab value 220.000000 for Campesterol falls in the Borderline range. (150.000000 - 260.000000 µmol x 100/mmol of Total Cholesterol)	LabRanges[2]

Messages



```
CorticonMessage.new[
decisionService='Check for Ambiguity',
ruleSheet='Range Ambiguity.ers',
ruleID = '0',
version='1.0',
severity='Violation',
messageText='Range Conflict for ' + input.name + ': ' +
input.lower.toString  + '-'+ input.upper.toString + ' (' +
input.classification + ') overlaps ' + db.lower.toString + '-' +
db.upper.toString + ' (' + db.classification + ')'
```

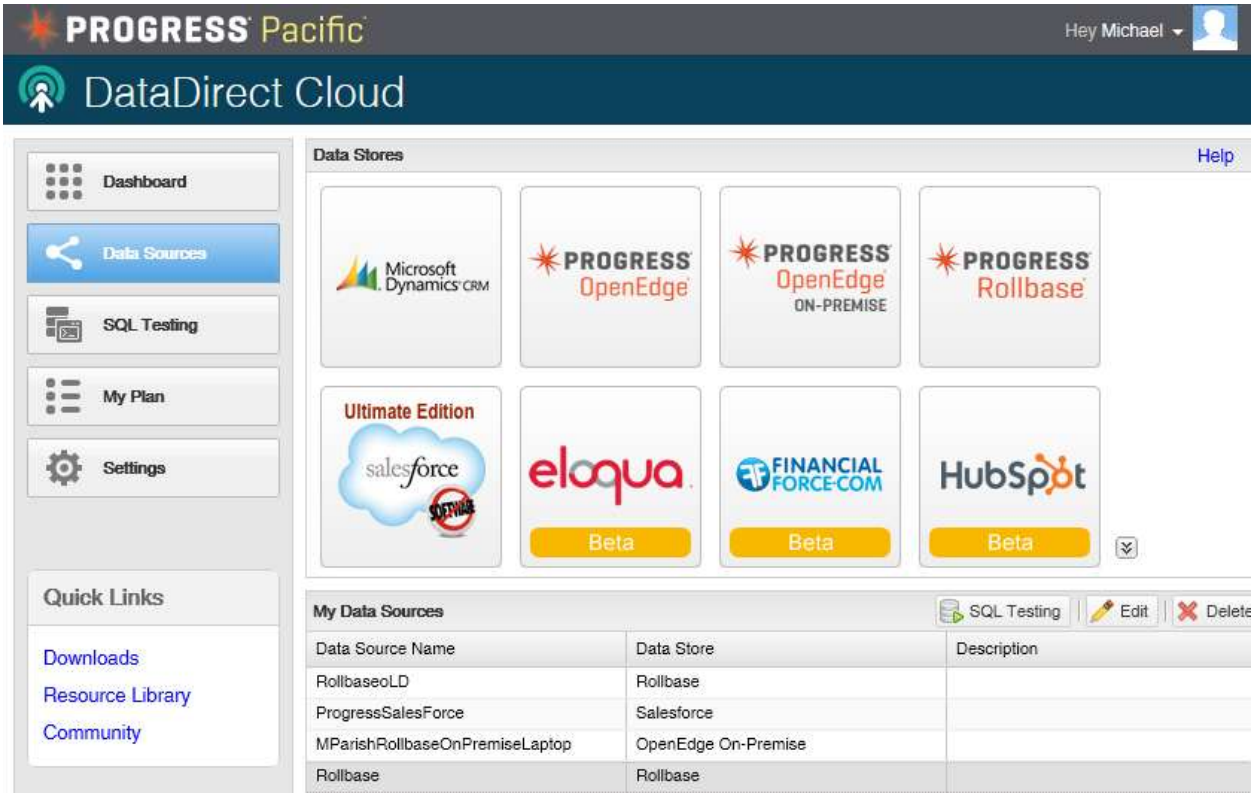


Action	CorticonParameter	Classification	Lower	Upper	Units	isValid	errorMessage	Updated At	Updated By
Edit | Del	HbA1C	High	25.00	50.00	units	yes	, Overlap	02/02/2014 01:21 PM	Michael Parish
Edit | Del	Lathosterol	High	160.01	9,999.00	Percent of total Cholesterol	yes	OK	02/01/2014 01:11 PM	Michael Parish
Edit | Del	Lathosterol	Borderline	90.01	160.00	Percent of total Cholesterol	yes	OK	02/01/2014 01:10 PM	Michael Parish
Edit | Del	Lathosterol	Optimal	0.00	90.00	Percent of total Cholesterol	yes	OK	02/01/2014 01:09 PM	Michael Parish
Edit | Del	HbA1C	High	1.00	3.99	mmol	yes	OK	02/01/2014 12:15 PM	Michael Parish
Edit | Del	HbA1C	low risk	4.00	7.50	test2	yes	OK	02/01/2014 12:05 PM	Michael Parish
Edit | Del	HbA1C	high risk	45.00	999.00	test2	yes	OK	02/01/2014 12:04 PM	Michael Parish

How to Connect Corticon to the Rollbase Database

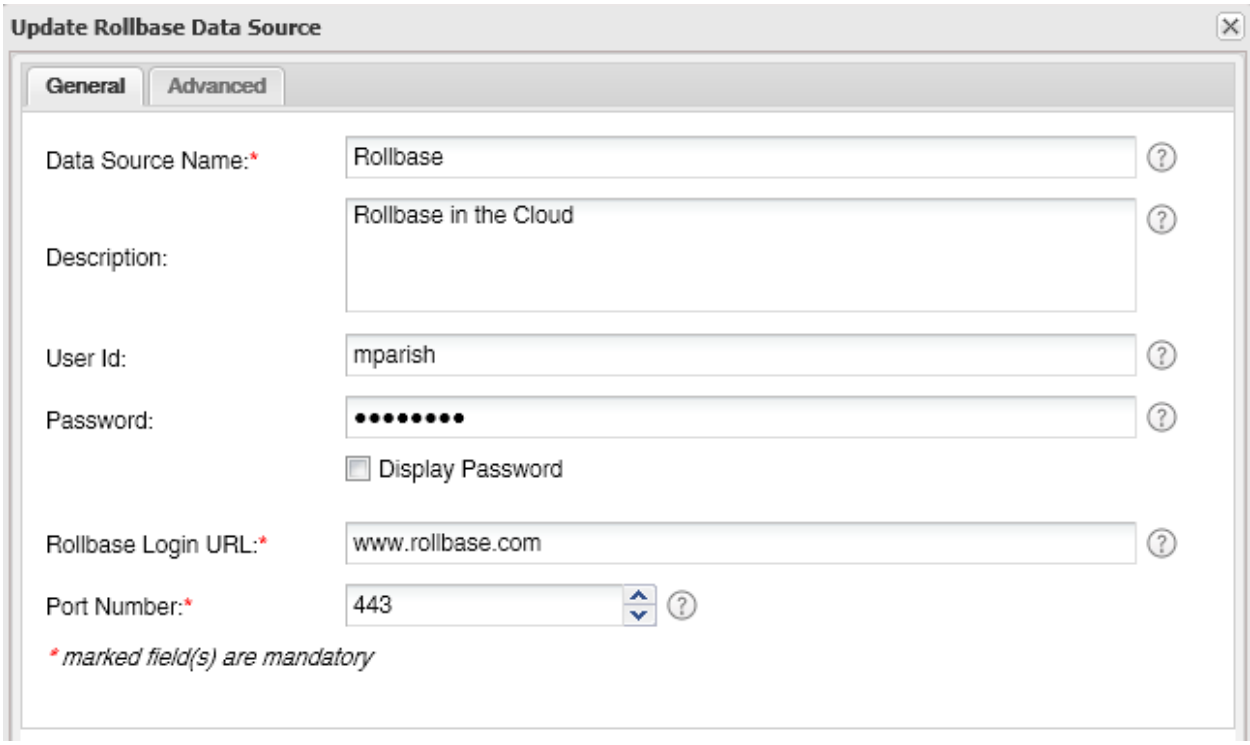
Create a Data Direct Cloud Source that points to your Rollbase instance.

Choose Rollbase as the data source



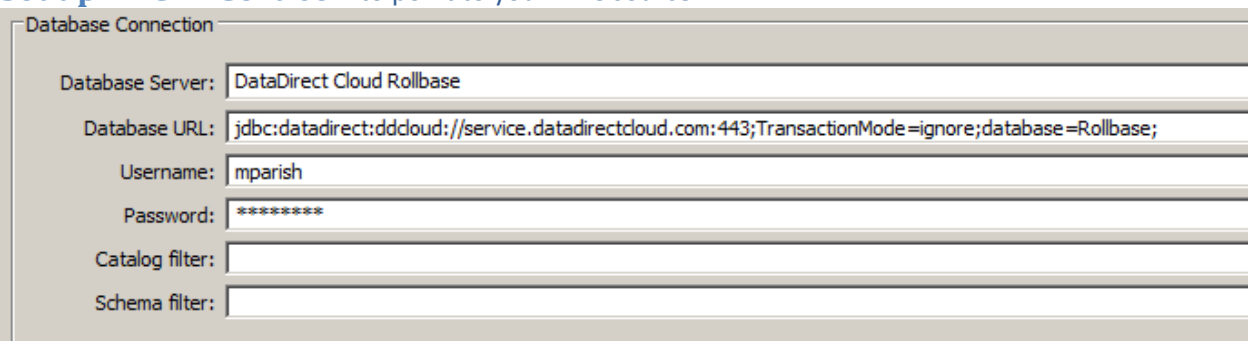
Enter your Rollbase Credentials

On this screen use the user id and password for the selected data source.
For Rollbase it will probably be your Progress id and password



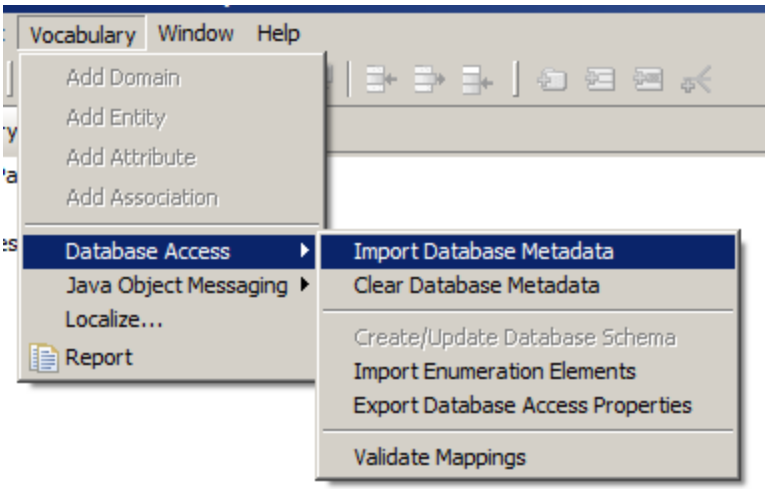
NOTE: When you add business objects to your Rollbase instance you'll need to recreate this data source otherwise it won't pick up the newly added objects.

Set up EDC in Corticon to point to your DDC source:

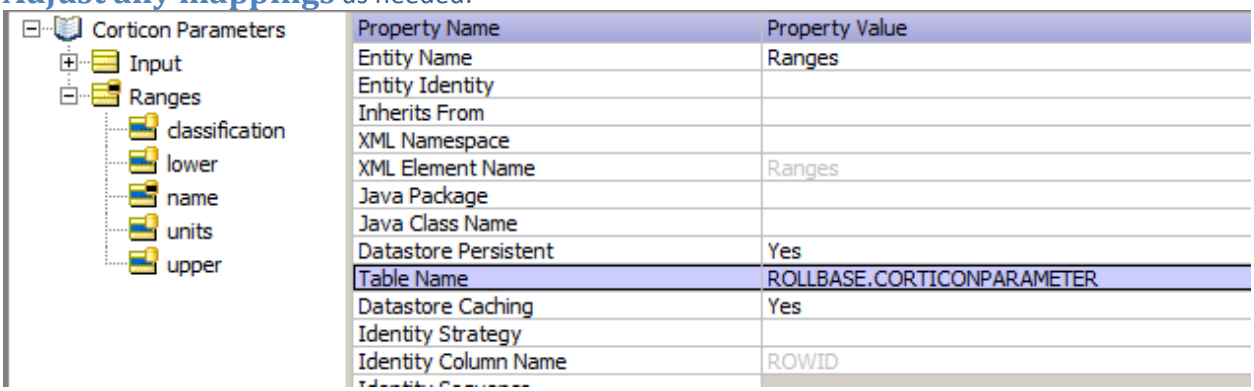


The user id and password is the one you use to connect to DataDirect (probably your Progress id and password)

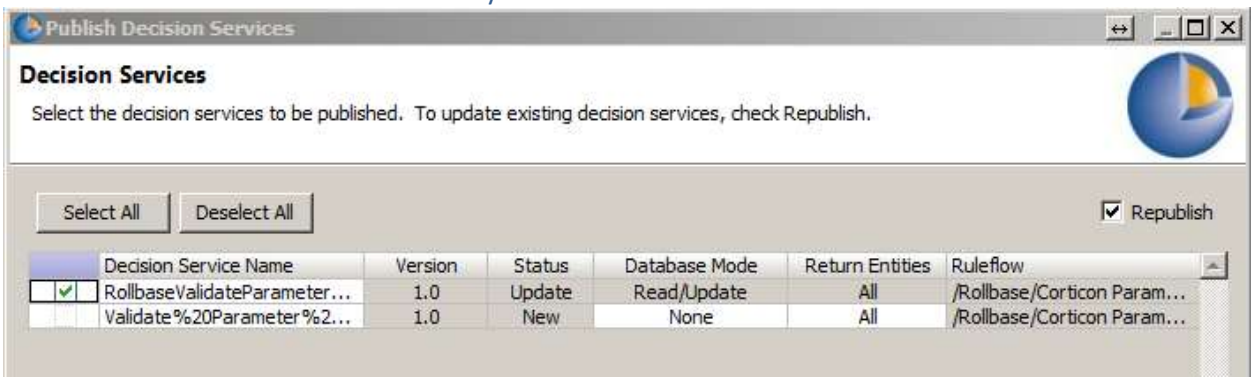
Import the database meta data into Corticon:



Adjust any mappings as needed:



Publish the decision service to your favorite Corticon server.



Make sure to set “read/update” for database access

Note that since the Rollbase instance is up in the cloud it will need to be able to “see” the URL for your Corticon instance. If Corticon is on your laptop then you may have to do port forwarding on your home router so that Rollbase can get to your Corticon server.

Configure the Web Console

In the web console, set up some monitored attributes to you can see what values are being passed into Corticon and what values the rules are producing. This can be very helpful in tracking down problems. A common problem is setting a Corticon value to something other than the contents of a picklist in Rollbase.

Modify the Document Template so the decision service name matches what you used when you deployed the decision service.

CorticonParameters +

Templates | New Mail Template | New Document Template | New Record Template

Action	Template	Type	Format	Created By	Last Updated	Last Updated By
Edit | Del	Corticon_Request_Template For Parameter Validation	Document	text/xml	Michael Parish	01/30/2014	Michael Parish

CorticonParameter: Edit Document Template

Save | Cancel

Define Document Template

Template Name: Corticon_Request_Template For Parameter Validation

Integration Code:

Render as PDF: (for HTML templates only)

Template Helper | String Tokens | Test EVAL[ ] block

CorticonParameter

Upload File: Browse...

View current template

Supported template formats: **DOC, XLS, HTML, RTF, CSV, XML, TXT, PDF**

... or modify text below:

```
<?xml version="1.0" encoding="UTF-8"?>
<SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" xmlns:SOAP-
ENC="http://schemas.xmlsoap.org/soap/encoding/" xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <SOAP-ENV:Body>
    <CorticonRequest xmlns="urn:Corticon" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
decisionServiceName="RollbaseValidateParameterRanges">
      <WorkDocuments>
        <Input id="Input_id_1">
          <name>{!name}</name>
          <lower>{!lower}</lower>
          <classification>{!classification}</classification>
```

If you decide to pass any additional fields to the rules you can add them here.
If appropriate you could add either a version number or an effective date (but not both)

Modify the Request Trigger

Enter the URL for your Corticon Server (You can try mine, but it won't always be running)

CorticonParameter: Edit Trigger

Save Cancel

Deployment Status

☑ This trigger is deployed

General Properties

When should this trigger run? In addition to this timing criteria, triggers can also be invoked from workflow actions of the type "Run Triggers".

Timing ☐ Before Create ☐ Before Update ☐ Before Delete ☐ On Finalize
☑ After Create ☑ After Update ☐ After Delete

Type Send HTTP POST Request

Trigger Name Corticon Request Trigger for Parameter Validation

Integration Name

If you want this trigger to run only when a particular field changes value, select the field that should fire this trigger here. This option will only affect "Before Update" and "After Update" triggers.

On Field Change Any Update

Trigger Properties

Enter the target URL and optionally include HTTP headers for this HTTP POST request

Target URL http://108.185.68.230:8082/axis/services/Corticon

Encoding ◉ XML ○ URL ○ Other

Content Type text/xml; charset=UTF-8

HTTP Header **Header's Value**

SOAPAction urn:Corticon

Modify the Response Trigger

But only if you need to extract additional data from the Corticon return payload

```
 if (tag == 'ns1:errorMessage') { // Corticon attribute name
 rbv_api.setFieldValue('corticonparameter', {!id}, 'errormessage',value);} //value must match picklist
if (tag == 'ns1:isValid') { // Corticon attribute name
 rbv_api.setFieldValue('corticonparameter', {!id}, 'isvalid',value);} //value must match picklist
```