



ONE - Workshop

DQ Advanced - Component Rules

Prepared for: v15.4.x

Prepared by: Ataccama

Dated: November, 2024

Contents of the Document

Introduction	3
Tasks	3
1. Creating a component Rule	3
1.1. Creating a validation component	3
1.2. Creating a DQ Rule using a Validation Component	10
Conclusion	13

Introduction

In this workshop you will learn how to use the ONE Desktop capacity to create more complex Data quality rules to be used in the ONE Web application.

Tasks

Sometimes, you need to create a logic for a DQ rule that is too complex to create using the existing options in the web application. In this case, you may consider the ONE Desktop application and use its whole set of features. The rules created this way are called **component rules**.

- Installing the ONE Desktop application and connecting to the ONE Web Application is a prerequisite for the Task 1.1.



*The connection and integration with the ONE Desktop application were already covered in the previous workshop focusing on the **ONE Desktop Integration**.*

1. Creating a component Rule

In this workshop, we will create a **component rule** to validate **credit card numbers**, their **length**, and check if they **contain only digits**.



This scenario has been selected for training purposes. However, it can be completely configured in Ataccama ONE using the functionalities of the web application (Rules Implementation Rule Logic Advanced expressions)

Creating a component rules has two parts:

- Creating a validation component
- Creating a DQ rule using the component in it

1.1. Creating a validation component

- › Navigate to the **Data Quality** section - **Components**> **Validation Components** and click on **Create** in the upper right corner.



*In the Data Quality>Components, you can also find **Transformation** and **Postprocessing** components.*

Data Quality

- Rules
- Components**
 - Validation Components
 - Transformation Components
 - Post Processing Components
- Transformation Plans
- DQ Firewalls
- Monitoring Projects
- Reconciliation Projects
- Lookup Items

Components ? Create ⋮

Published Unpublished All

Search: Type here to search full-text for Components

State: ▼

Standard view ▼ Hidden columns ▼

<input type="checkbox"/>	Name	Component description	Logic description	State
<input type="checkbox"/>	customers_na Excel Report			Implemented
<input type="checkbox"/>	Excel Export			Implemented

- › Fill in its **Name** (e.g., “<prefix>_Credit Card Validation”).
- › Select the **Type** to be a **Validation Component** and **Save** the new component

Create Component ? Save

General information

Name *
KS_Credit Card Validation

Component description

Type *
Validation component

After saving you will be forwarded to the **Overview** tab of the component, where you can see the defined input attributes and description.

- › Switch to the **Implementation** tab.
- › Create a new **Input Attribute** and rename it if you want (e.g., '**VALUE**'). Keep the default **String** data type.
- › You can provide a **Logic Description** – that is useful if the implementation of the rule logic in ONE Desktop is supposed to be done by somebody else, a developer for instance; this way, the developer will get an idea about the purpose of the rule.
- › Add **Explanations**

Components
XY_Credit Card Validation

Overview History Implementation Occurrence

Input Attributes 1

Abc String VALUE

+ Add Term

+ Add input

Implementation State

Ready for implementation

Description

Logic Description

A rule to validate credit card numbers, their length, and check if they contain only digits.

Explanations ⓘ

Explanation INVALID

Explanation VALID

+ Add

Predefined results with explanation ⓘ

Fallback OTHER

- › Change the **Workflow State** to **Ready for implementation** and **publish** it.

New ⓘ Discard Publish

Components
XY_Credit Card Validation Edit ⓘ

Overview History Implementation Occurrence

Input Attributes 1

Abc String VALUE

+ Add Term

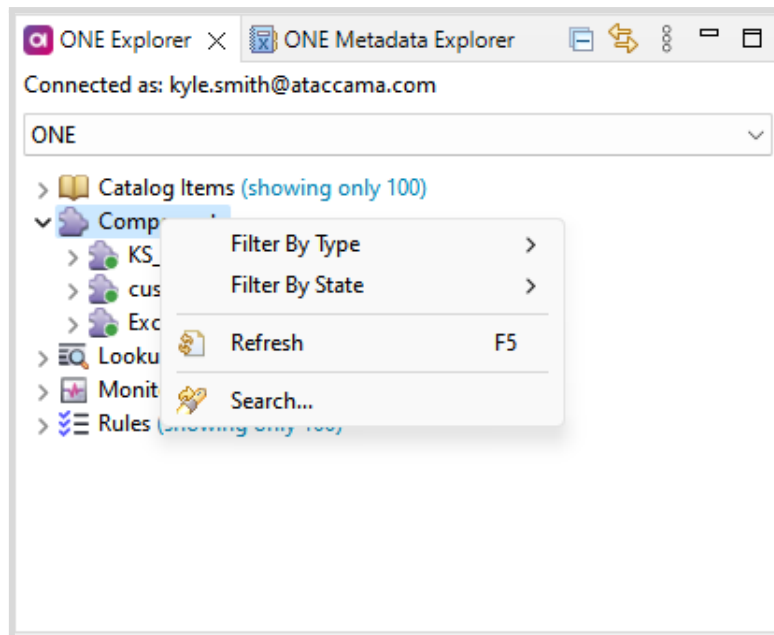
Implementation State

Ready for implementation

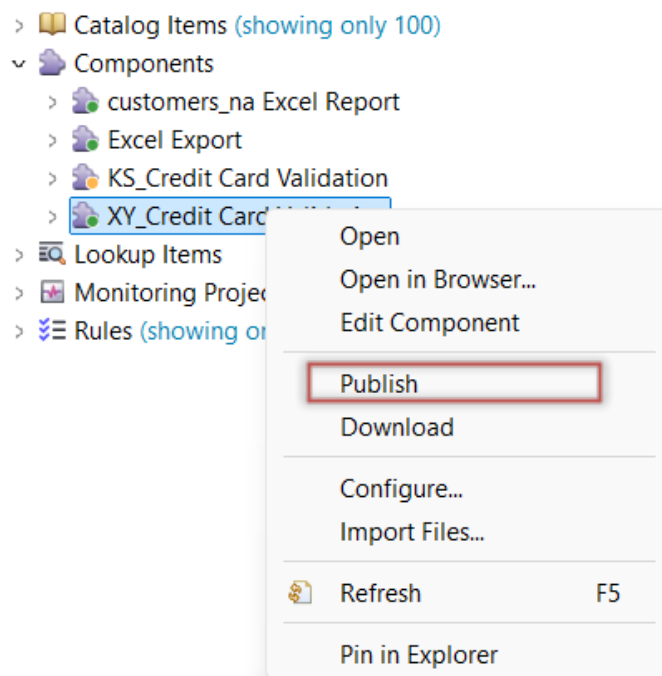
Now the component should be available in the ONE Desktop:

- › Go to your ONE Desktop and connect to the ONE server (this was covered in the **DQ Advanced Workshop – Integration of ONE Desktop**).
- › In the **ONE Explorer** tab, expand the **Components** section.

- ▶ If you cannot see the newly created component, right-click on the item and click on **Refresh**; this will download the latest changes from the server.



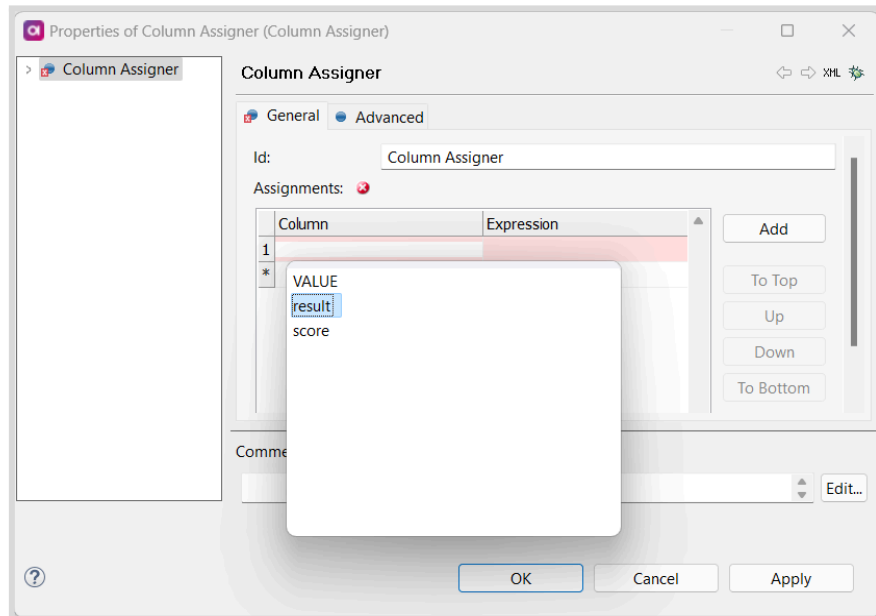
- ▶ Right-click on the component **<prefix>_Credit Card validation** and choose the '**Edit component**' option.



This action will open its configuration in a pre-generated component file where you can develop the logic. Notice the guidelines in the provided comment box. You can see the text you entered for the **Logic Description** you filled in the ONE Web Application and other useful information on how to create the rule logic.

In the next phase, we will construct the logic of the rule itself. We want to check if the credit card number contains only 16 digits (common Credit Card format) without letters and other characters. We will use the **Column Assigner** step to apply this logic via an expression.

- › Place a new **Column Assigner** step between the two already existing Integration steps in the data flow. Specify the **result** column as the target of your logic.

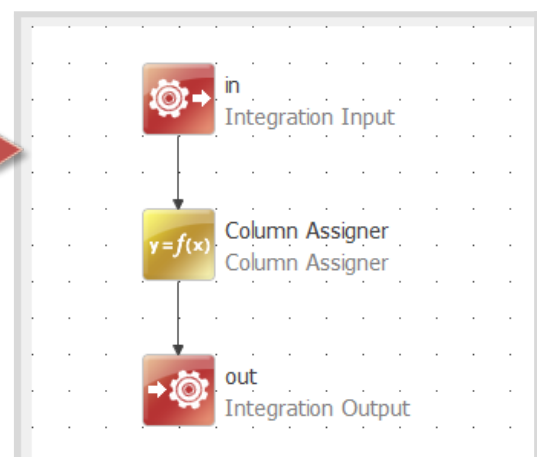
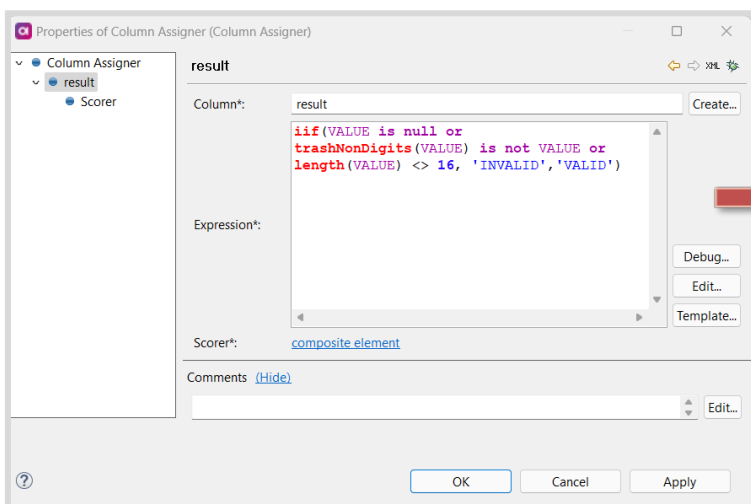


- › Create the expression so that it returns either '**VALID**' or '**INVALID**' strings to the **result** column:

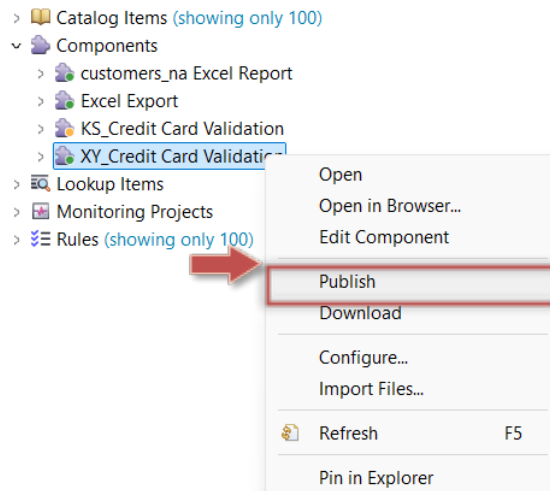
iif (VALUE is null or
trashNonDigits (VALUE) is not VALUE or
length (VALUE) is not 16, 'INVALID','VALID')



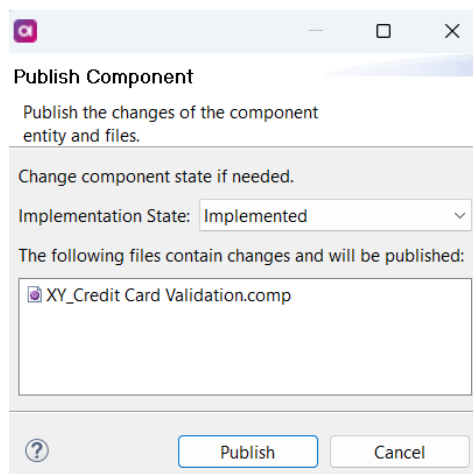
Explanations for INVALID situations might be configured and used in developing the logic if they have been defined in the Component's **Implementation** tab of the ONE Web application.



- › Click **OK** to save the changes in the step.
- › **Save** the complete plan, then right-click on the component item in the **ONE Explorer** tab and select **Publish**.



- › Choose **Implemented** as the component **State**. This will indicate that you are done with the implementation and the component can be used in the ONE Web Application.



- › **Publish** your changes so that they will be updated to the ONE Web Application.

If you check the Components section in the web application and refresh the page, your component should now be in the **Implemented** state there as well.

Data Quality

Rules

Components

Validation ...

Transform...

Post Proc...

Transformati...

</> DQ Firewalls

Monitoring P...

Reconciliatio...

Lookup Items

Components

Published

Unpublished

All

Type here to search full-text for Components

State

Standard view

Hidden columns

	Name	Component description	Logic description	State
<input type="checkbox"/>	customers_na Excel Report			Implemented
<input type="checkbox"/>	Excel Export			Implemented
<input type="checkbox"/>	XY_Credit Card Validation		A rule to validate credit card numbers, their len...	Implemented

1.2. Creating a DQ Rule using a Validation Component

Now when we have the validation component created, it is time to use it in a rule:

- › Create a new DQ rule and call it '**<prefix>_validation Credit Card**'.
- › Select **Validity** as the **Rule Type**.

The screenshot shows the 'Create Rule' interface. On the left is a sidebar with a 'Data Quality' header and a list of icons. The main area is titled 'Create Rule' and includes a 'Save' button. Under the 'General information' section, the 'Name' field is populated with 'XY_Credit Card Validation'. The 'Rule type' dropdown is set to 'Validity'. The 'Description' field contains a rich text editor with the text 'Fix grammar'.

- › Switch to the **Implementation** tab.
- › Make sure the default Input of your rule matches that of your component; you can rename it to that input name if you want.
- › Change the **Logic** of the rule from **Rule** to **Component**.
- › Choose the component you have just created and modified in the previous part - the new **Credit Card validation** component.

New ⓘ
Discard
Publish

Rules
XY_Credit Card Validation
🗨
✓
🔗
?
Edit
⋮

Overview
Implementation
Occurrence
History

☐ Use AI to generate Rule logic and inputs Beta

DQ Evaluation Rule

Validity

Inputs ⓘ

ATTRIBUTES 1

Abc String

VALUE

🗑

▼

+ Add Term

+ Add attribute

Rule Logic

Component

Test Rule

📘 Component rule can't be processed in pushdown.

🔗 XY_Credit Card Validation

Description

B I </> ☰ ☷ H1 H2 🔗

/ ↶ ⚙ Generate ▼

▼ Rule summary

DQ Dimension: Validity

Input attributes, parameters and variables used in conditions

Rule logic

Failed result

Passed result

1. OTHER

- Map the input attribute of the component to that of the Rule and map the explanations as appropriate.

1 Input Attributes Map Component Inputs to Rule Inputs

Component Input

Rule Input

Abc VALUE

→ VALUE

✕

▼

2 Rule Logic 🔗 XY_Credit Card Validation

3 Explanations

Explanation












Result

INVALID Invalid

VALID Valid

OTHER Valid

- Test** your rule by clicking the Test Rule button..

Test		
VALUE		Message
1234567890123456		 VALID
1234567890123456		 INVALID
VISA1234567890123456		 INVALID
123456789012345		 INVALID
MY CREDIT CARD		 INVALID
 New Row		

- › **Publish** the changes. The rule is now ready to be used in DQ evaluation of Catalog Items or monitoring projects.

Conclusion

We have come to the end of this workshop!

We have learned how to use ONE Desktop to create even more complex Data Quality rules.