



Purpose

Purpose here is to introduce modification of existing rules and to use XMLdation validator service when testing, as opposed to Studio built in validation.

One of the benefits of using the validation service is that it allows us to edit XML files in the browser, enabling us to test the changes we have made quickly.

Description

Following steps are done here:

1. Modifying validation rule level
2. Modifying validation rule message
3. Compiling and deploying
4. Testing in validator
5. Using edit mode in validator

Instructions

1: Modifying validation rule level

In order to modify a rule it has to be opened.

In this example we can modify the rule created in previous section, Creditor name not allowed. Double clicking the box will open a modal containing all data for the rule.

Creditor ▼	PartyIdentification	1..1	
Name	Max140Text	0..1	▲ Name_NotAllowed
Remittance ▼	RemittanceType	0..1	

In this section we don't have to worry about displayed here. Instead, we'll find the section "severity".

Name_NotAllowed
(Antero's ruleset) Rule XML view

Details Actions

MIG details: ▶

Rule basic details: ▼

Active

Name* Name_NotAllowed

Internal note Optional internal note of the rule

Wiki <http://wiki.xmldata.com/XMLdataon-wiki-page>

Rule location XPath /Message/Transaction/Creditor/Name

Validation definition: ▼

Severity* ▲ Error ▼

Context* TransactionType1

Definition*
1 self.Creditor.Name->size() = 0

Validation message* Element 'Name' is not allowed

Message location
1 self.Creditor.Name

[Customize message locations](#)

Delete Cancel Save

And change it to notice

Validation definition: ▼

Severity* ● Notice ▼

Context* TransactionType1

This means that if the rule is saved now, it will return a notice level message instead of an error level message.

2: Modifying validation rule message

The task we did in phase 1 modified severity of rule. This means that our message wording would be in a

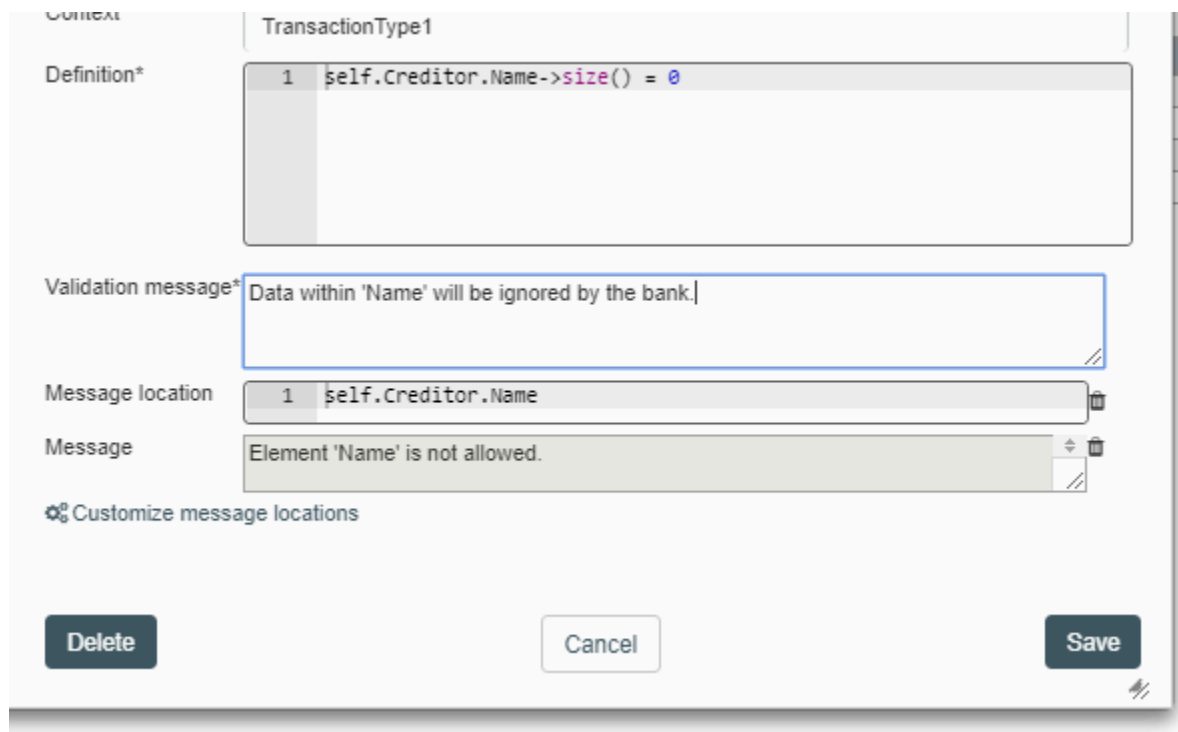
mismatch with the level.

Element 'Name' is not allowed.

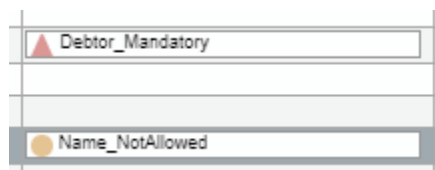
indicates an error, but rule is not an error.

Therefore we should change this text to following, for example:

Data within 'Name' will be ignored by the bank.



After clicking save, rule will be saved to Studio and it's level updated in the UI



Not required in this phase, but here it is worth locating the button for rule deletion, which may be useful later on. This button is in the lower left corner of the modal.

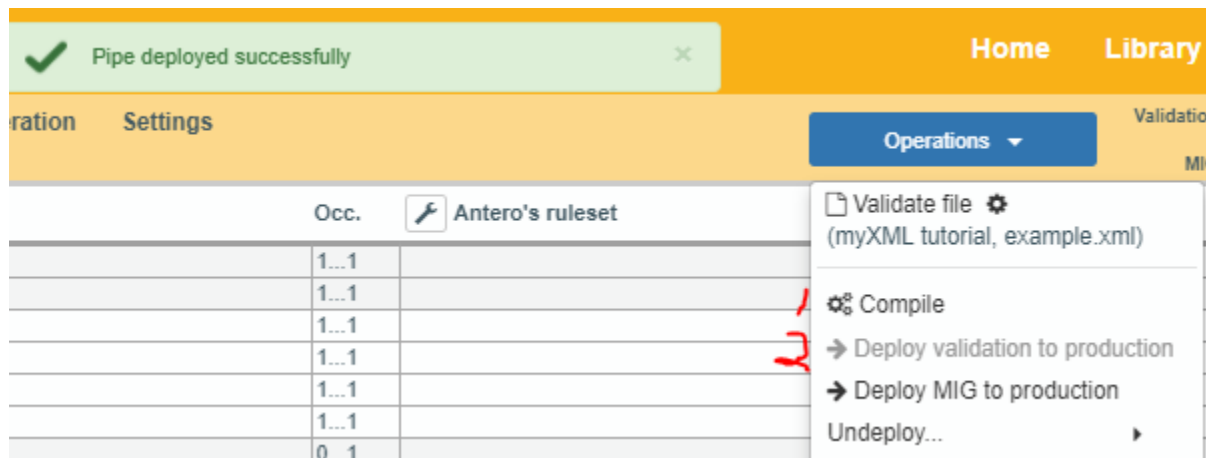
3: Compiling and deploying

At this point, we may test the implementation again via Studio validation. But in addition we can also try out the normal validation accessible for XMLdation validation service users.

In order to activate the changes we have made two prior actions will have to be done:

1. Compiling
2. Deploying

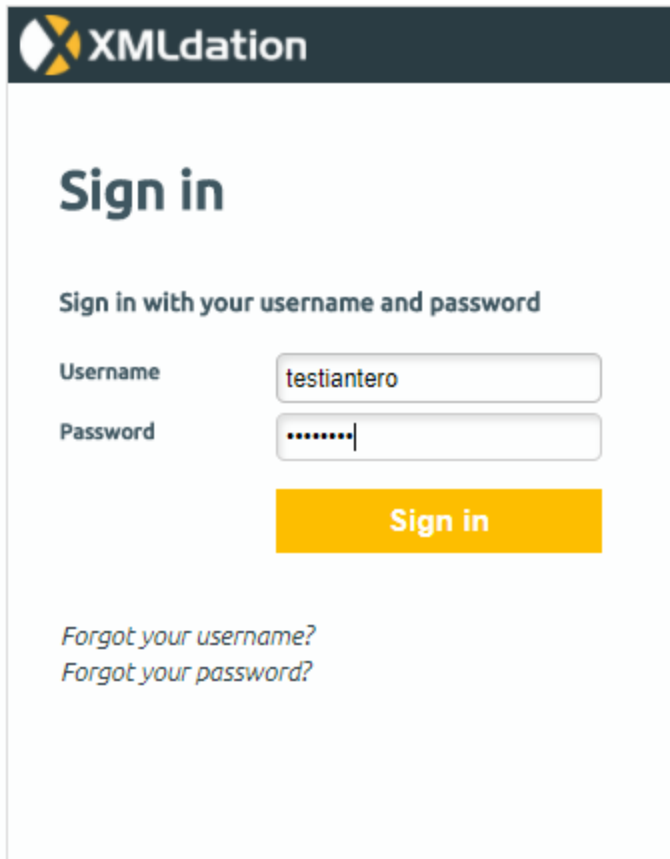
Compile makes sure the rules we have made are valid. Deploying means we transfer the changes to validation service. If deploying is not done, the changes we have done will not be visible from validation service's point of view.



4: Testing in validation service

We can login using the same credentials as used in Studio, via url <https://demo.xmlvalidation.com>.

It's a good idea to open this service to another tab/window so we can return to Studio later.



XMLdation

Sign in

Sign in with your username and password

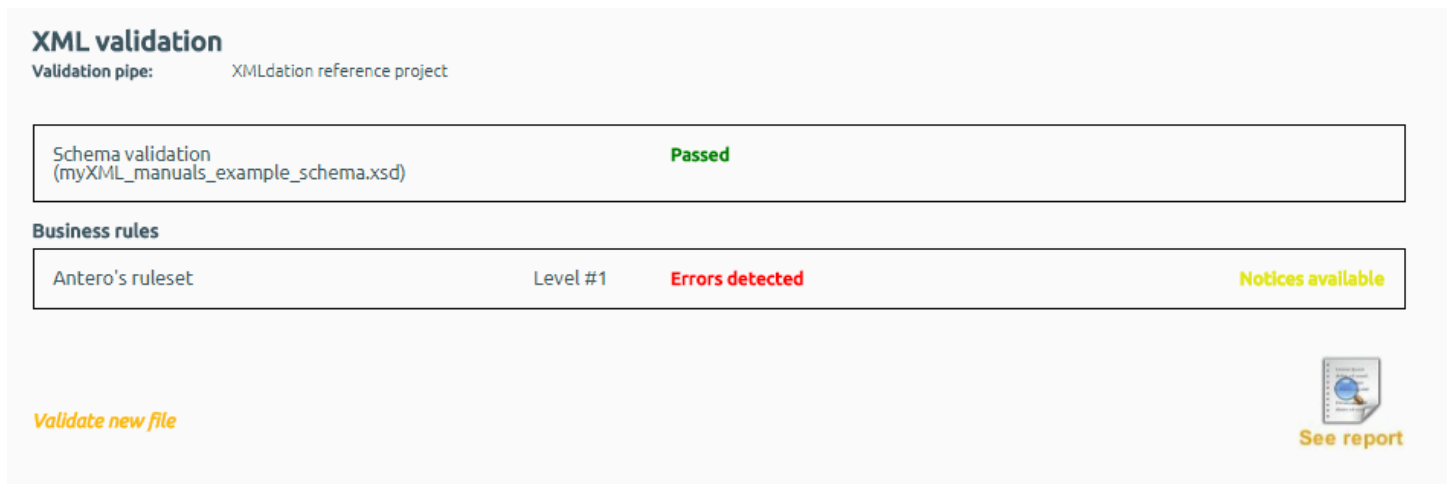
Username:

Password:

Sign in

[Forgot your username?](#)
[Forgot your password?](#)

After login, selecting validation -> selecting browse -> selecting the earlier example file -> clicking validate -> selecting our newly created validation pipe, we should get a validation result




XML validation
 Validation pipe: XMLdation reference project

Schema validation (myXML_manuals_example_schema.xsd)	Passed
---------------------------------------------------------	---------------

Business rules

Antero's ruleset	Level #1	Errors detected	Notices available
------------------	----------	------------------------	--------------------------

[Validate new file](#)

 **See report**

Opening the report will show the XML data. Benefit of this mode is that we are able to use "edit mode", which means that we're able to test the validation rules further.

5: Using edit mode in validator

At this point we should have following result open

XMLdation Validation Report

Report ID: 158938
 Date: 2018-01-05 12:17:08
 Original filename: myXML tutorial, example.xml
 Validation pipe name: (XMLdation myXML training) XMLdation reference project
 Validation result: **Failed**

Report as PDF
 Report as HTML
 Validated file
 Compact report

Edit Mode

```

1: <?xml version="1.0" encoding="UTF-8">
2: <Message xmlns="http://www.XMLdation.com" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
3:   <Header>
4:     <Id>ExampleId11</Id>
5:     <TimeStamp>2015-07-03T12:17:50</TimeStamp>
6:     <ControlSum>6</ControlSum>
7:     <NumberOfTransactions>3</NumberOfTransactions>
8:   </Header>
9:   <Transaction>
10:    <Id>a</Id>
11:    <Amount>2</Amount>
12:    <Debtor>
13:      <Name>Debtor1</Name>
14:    </Debtor>
15:    <Creditor>
16:      <Name>Creditor1</Name>
17:    </Creditor>
18:  </Transaction>
19: </Message>
20: </xml>
    
```

NOTICE Data within 'Name' will be ignored by the bank.
 Notice origin: level #1 - Antero's ruleset
 XPath to the element: //Message[1]/Transaction[1]/Creditor[1]/Name[1]
 Rule name: Name_NotAllowed

In the upper right corner, there's an icon for edit mode. Clicking it will make XML editable. Therefore we can test whether a notice is given when Creditor name is missing from the file

```

10:   <Id>a</Id>
11:   <Amount>2</Amount>
12:   <Debtor>
13:     <Name>Debtor1</Name>
14:   </Debtor>
15:   <Creditor>
16:     <Name>Creditor1</Name>
17:   </Creditor>
18: </Transaction>
19: </Transaction>
    
```

-->

```

12:   <Debtor>
13:     <Name>Debtor1</Name>
14:   </Debtor>
15:   <Creditor>
16:   </Creditor>
17: </Transaction>
18: </Transaction>
19: </Transaction>
    
```

Clicking revalidate will initiate new validation run

Report as PDF
 Report as HTML
 Validated file
 Compact report

Revalidate Exit edit mode

In this validation, we should no longer be getting the notice for Name which is present.

At this point we're able to create, modify and test validation rules. Next we'll change rulesets to be group rulesets, enabling us to work more efficiently in the case when we have multiple different validation projects we should do.