

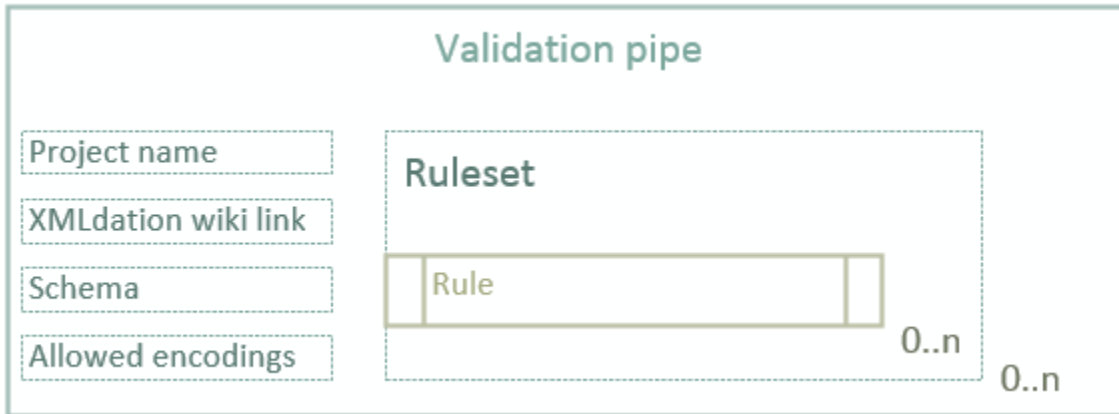
**Purpose**

Rules are used to execute constraints against XML file and may be written by a user of myXML. These rules return a boolean value, either true or false.

Rules themselves may have been defined by a third party and published in a Message Implementation Guide (MIG). Rules within MIG are often referred as "business rules".

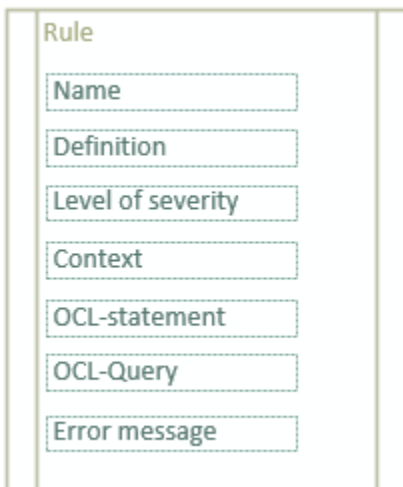
In XMLdation service, collection of several rules is called a ruleset, which forms a validation pipe. XML files may then be inserted to a pipe in [validation process](#).

Below image shows the different parts required in a validation pipe.



**Building blocks of a rule**

Rule contains multiple parts



The core parts of rule are context, OCL-statement, OCL-query and error message. Core parts have their individual wiki entry.

When writing rules, it is crucial to first understand the [concept of context](#). In short, context is set to be complex or simpleType defined in the relevant schema, and OCL-rule is ran whenever the type defined in the context is found within the XML-file.

Restriction defined in the [OCL-statement](#) returns a boolean value. Whenever false is returned, XMLdation

service will return an error message to a pre-defined location within the XML-file (described with OCL-query). Whenever true is returned, the restriction passes and XML file is considered valid against that rule.

The returned error message is written by the creator of the rule. It is good to have in mind that this message, apart from rule logic itself, is the only visible item in validation result for the end user, so writing error message in a clear way, explaining why rule did not pass and describing how to fix the issue is encouraged.

### Examples from myXML

Screenshot of a ruleset containing two rules, viewed in myXML's rules view is depicted below:

The screenshot shows the 'Validation Rule Management' section of the myXML interface. At the top, there are navigation tabs: 'Project Dashboard', 'Validation Rule Management', 'Test File Generation', and 'Settings'. Below these are buttons for 'Test' and 'Production'. A 'Create ruleset' dropdown and a 'Filter rulesets...' search box are also visible. The main area displays a ruleset named 'Example ruleset' with an 'Add new rule' button. Below this is a table listing the rules.

|   | M | A | Rule name | Description | Act                                 | Level | Context     | C   | N   |   |
|---|---|---|-----------|-------------|-------------------------------------|-------|-------------|-----|-----|---|
| ▶ | 0 | 0 | Test rule | null        | <input checked="" type="checkbox"/> | Error | HeaderType1 | sel | Val | <input type="button" value="edit"/> <input type="button" value="delete"/> |