Data Quality Assessment

Understanding and Generating Data Quality Results with Ab Initio

Overview

Data Quality Assessment can be used to perform assessments of data files or implement data quality systems in enterprise production data flows.

DQA provides an environment for:

- Detecting and correcting data quality issues
- Computing data quality metrics
- Monitoring the data quality of enterprise applications

Features and Benefits of DQA

Business validation rules can be created, tested, and implemented in the DQA without requiring programming experience.

- Reduces development time to implement data quality tests
- Reduces errors translating rules to code
- Validation rules are easier to maintain and update

Users can load data, test for validation errors, and import reports for other viewers to review.

Course Objectives

On successful completion of this course, you will be able to:

- Locate Dataset Data Quality Results in Metadata Hub
- View Data Quality in context of a dataset
- Understand relationship of Metrics and Issue Codes
- View Data Quality History
- View Rules from Metadata Hub (if available)
- Write Data Quality Configuration
  - Create field level validations
  - Create dataset level validations
  - Create and use a Lookup
  - Run and Import Results from Express>It

Course Objectives

Additionally, we may cover additional topics as desired.

These require access to a command-line and GDE:

- Run and Import to Metadata Hub from command-line
- Configuring a new private project for use with Data Quality
- Optional Data Quality Parameters
- Customize Metrics and Issue Codes
- Customize Alerts
- Post-processing results
- Installation of Data Quality Environment

Overview

Metadata Hub and Express>It

What is Data Quality?

Ab Initio defines metric scores for a particular *logical* dataset.

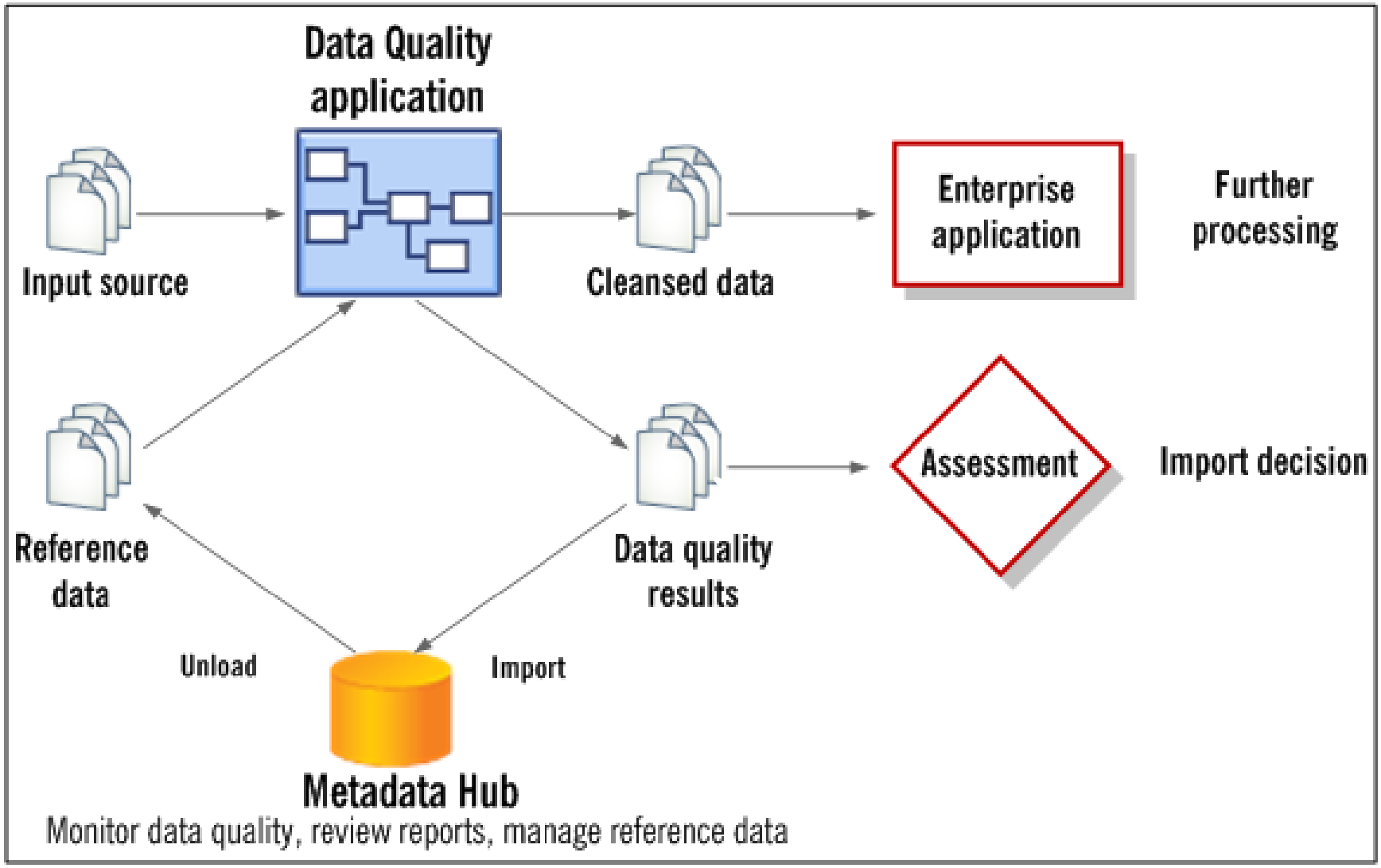
A *logical* dataset has a single format over time and is processed in the same way. The contents (the *physical* dataset) will change.

Each snapshot of a dataset's quality will receive metric scores, which will be stored for a historic view.

Each metric score is calculated based on results of individual rules written specifically for each dataset.

Rules determine invalidity of individual field value or dataset.

Data Quality Assessment



Which Environment?

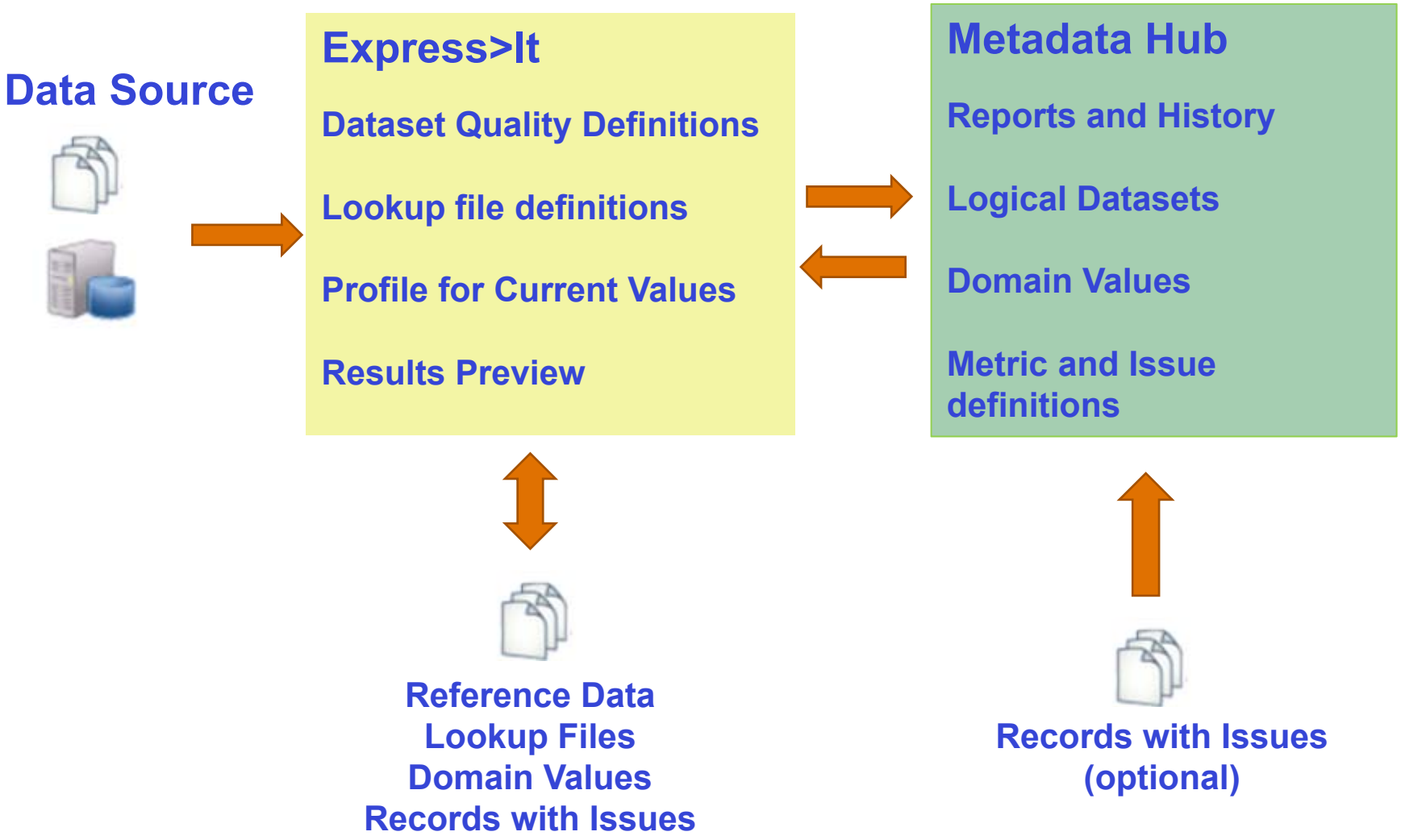
Express>It is used to configure rules for particular logical datasets.

- **address1 must have a value; address2 can be blank**
- **customer_id must be unique**
- **country_id must be in the list of approved countries**

Metadata Hub is used to view results and configure data quality metadata.

- **Where was poor quality data detected?**
- **How frequently were fields invalid due to incomplete data?**
- **How has Data Quality for a particular dataset changed over time?**
- **Is score low enough to trigger alert?**

Data Quality Assessment



Data Quality Issue Codes

Each validation test will generate an Issue Code, also known as an Error Code.

An issue code will be generated for each validation error – there can be none or many for each field or record.

Examples of Issue Codes are:

AB_ERR_IS_BLANK

AB_ERR_VALUE_TOO_LARGE

Each issue code is associated with a metric.

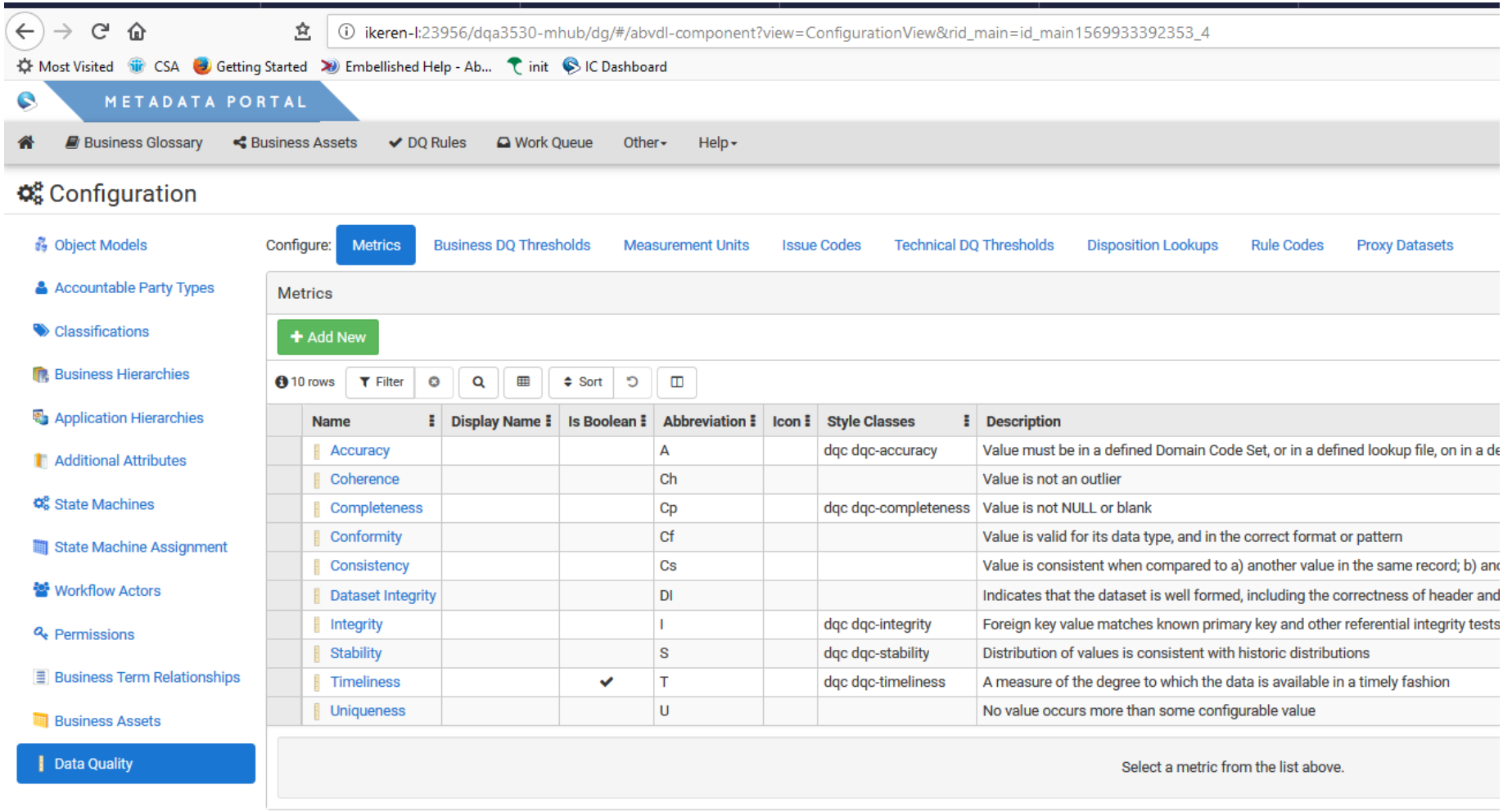
Data Quality Metrics

Metrics are calculated from groups of issues.

DQE pre-defines the following metrics:

- **Accuracy** Value must be in a specified list of values
- **Completeness** Value is not NULL or blank
- **Conformity** Value is valid for its data type, and in the correct format or pattern
- **Consistency** Value is consistent when compared to a) another value in the same record; b) another value in a different dataset or c) the previous version of the same dataset
- **Integrity** Foreign key value matches known primary key and other referential integrity tests
- **Stability** Distribution of values is consistent with historic distributions

Metadata Hub Demonstration



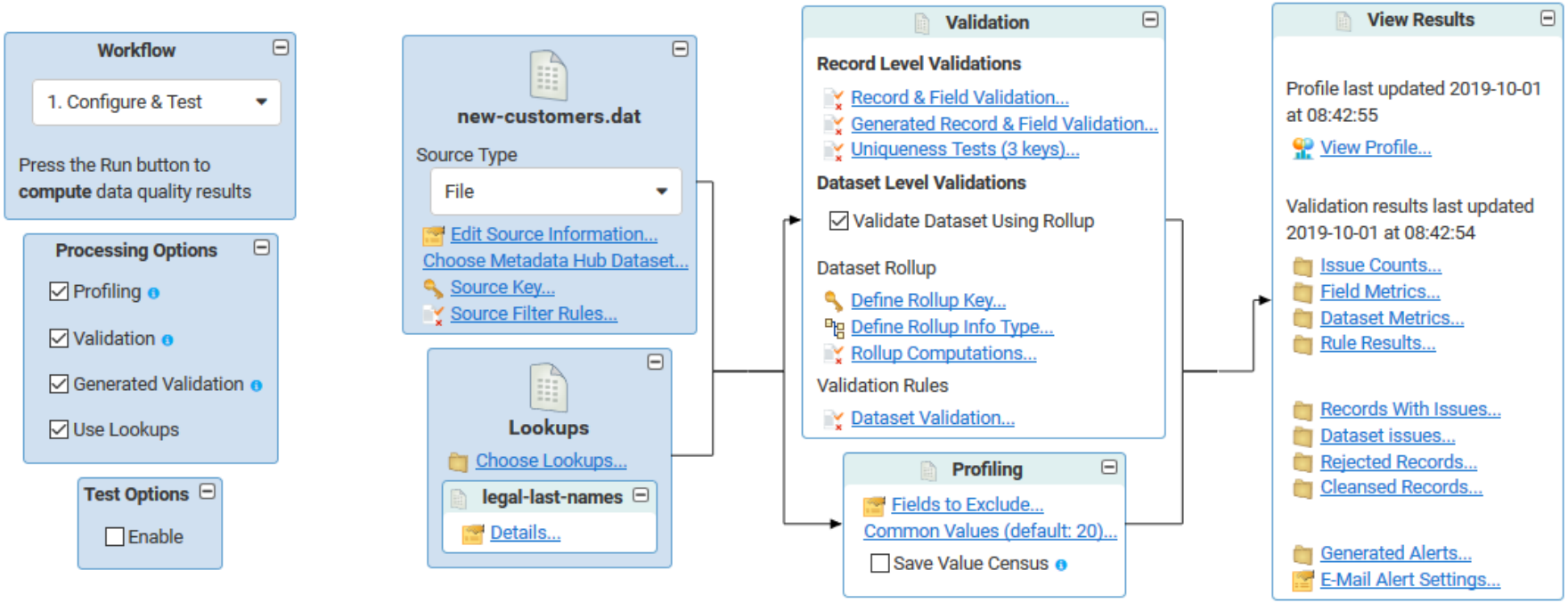
Data Quality Template

Adding a Dataset, Profiling, Validation Rule with Sheet View, Issue Codes, Testing & Importing

Data Quality Template

Data Quality in Express>It

The Data Quality Assessment uses Express>it to configure validation tests. It may live side-by-side with other Express>It applications.



Data Quality Template

Express>It Terminology

Templates define an interface for configuring a particular application. They are provided as part of solutions or written by technical developers.

Configurations are used to run an application using a particular set of values. You provide a name, template and project location when creating a new configuration.

Save and Publish saves a copy of the configuration to the EME, where others can retrieve and view it. This should only be done when the configuration is in a runnable state.

Configurations can exist in any number of projects that include the shipped Data Quality project.

Data Quality Template

Data Quality Templates

The Data Quality Environment provides five templates.

Data Quality Creates a dataset-specific data quality application.

Unload Metadata Hub Creates local copy of reference data maintained in Metadata Hub.

Create Lookup Create lookups for use in a Data Quality Configuration

Create Dataset Modify or join physical data sources to correspond to a logical dataset in the Metadata Hub.

Data Quality Group Create a collection of multiple configurations that are run as a batch process.

Join Analysis and Create Database Configuration templates are included for backwards compatibility.

Data Quality Template

Performs data quality analysis of a single dataset.

Each configuration describes:

- **Data Source**
- **Format of Data**
- **Validation Tests**

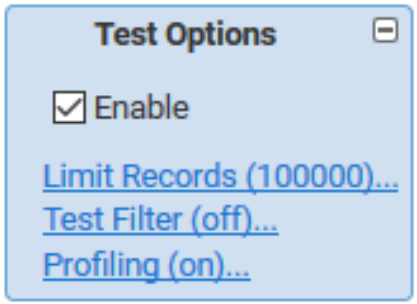
Data Quality Template

Limit Number of Records

Test Options are used to limit the number of records processed during testing.

At least one full test must be performed before you can generate a final report.

Make sure you use these during the class.



Filter… is used if certain records, such as headers and footers, should be excluded from the final report.

Data Quality Template

Data Sources

File — source data file that you can select in the filesystem

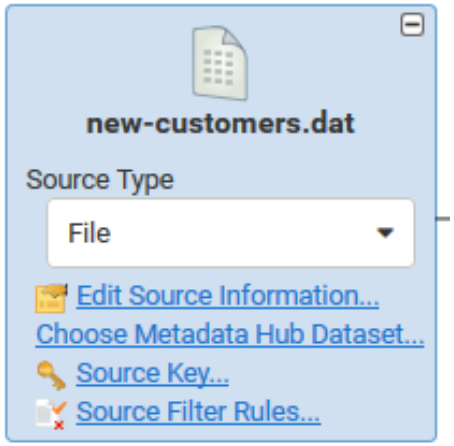
Database — database table or an SQL SELECT statement

EME dataset — previously created EME dataset

Excel — Excel spreadsheet

Graph — input pset for a graph that creates the data source

Hadoop — a Hadoop file or Hive table



Different data sources require different information to fully describe data

Data Quality Template

Record Format

You will usually have existing files to describe your data sources.

- Linked will always read from the current file and will incorporate any changes made outside the DQE to describe format changes.
- Embedded will save a copy of the current format. You will need to update the format, if there is a change in the file.

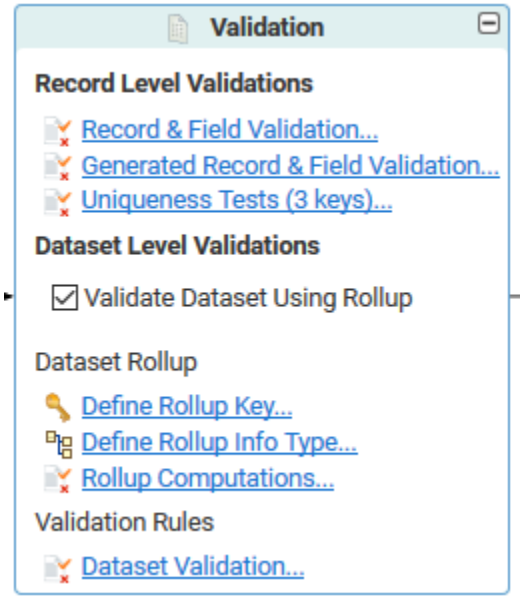
Data Quality Template

Validations

Validations are performed separately at the Record & Field level and at the dataset level.

Record and Field Validations are most common.

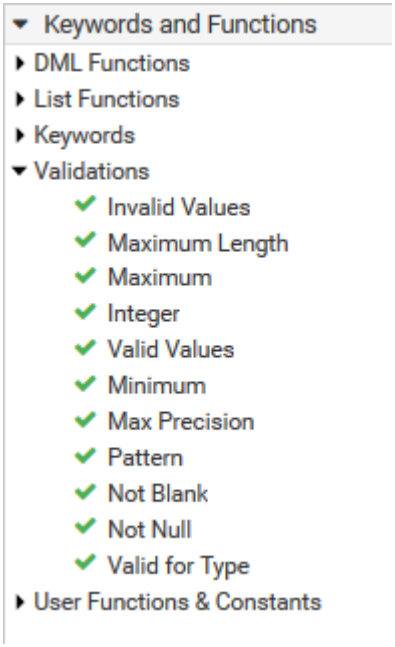
Dataset Validations must be explicitly enabled.



Clicking on these links will open the Ruleset Editor

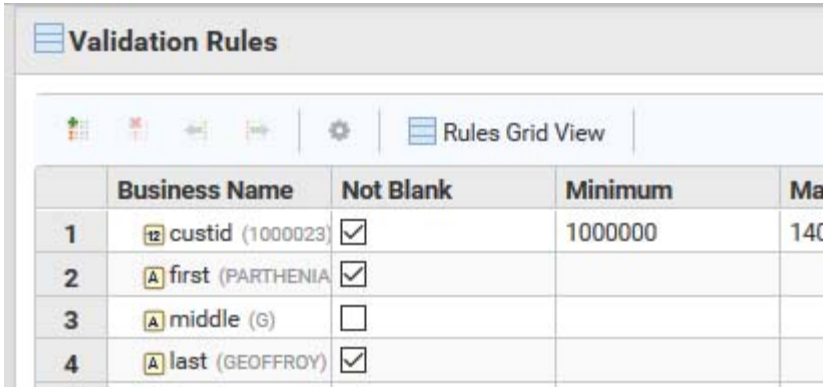
Data Quality Template

Specify Validations in the Ruleset Editor



In sheet view, each test is on a single field.

Double-click on a test in "Validations" to add a test and configure for each field.



Check or provide values per field and test.

An empty cell will not trigger a test.

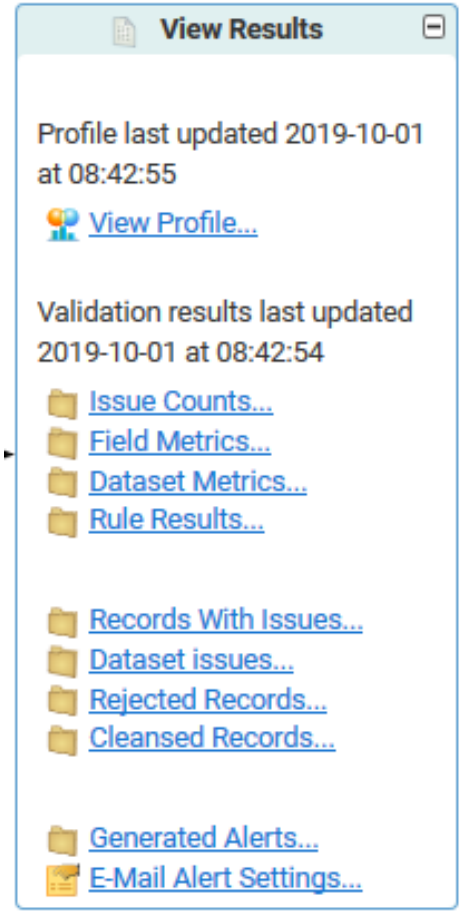
Data Quality Template

Running a Test

To view complete results, you need to Run **the configuration.**

The various links allow you to view the results in different ways before loading to Metadata Hub.

For large datasets, only a sample selection of values may be shown



Data Quality Template

Group Exercise: First Data Quality

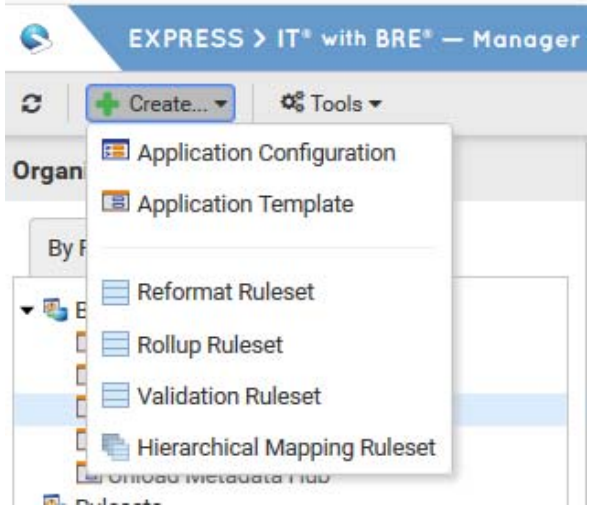
For the first exercise, we will walk through the process of creating a simple Data Quality configuration.

- **Create a new Data Quality configuration**
- **Specify the stores.dat file**
- **Add a sheet view validation to check for any blank values**
- **Run the configuration and look at the results**

Data Quality Template

Group Exercise: New Configuration

Create an Application Configuration

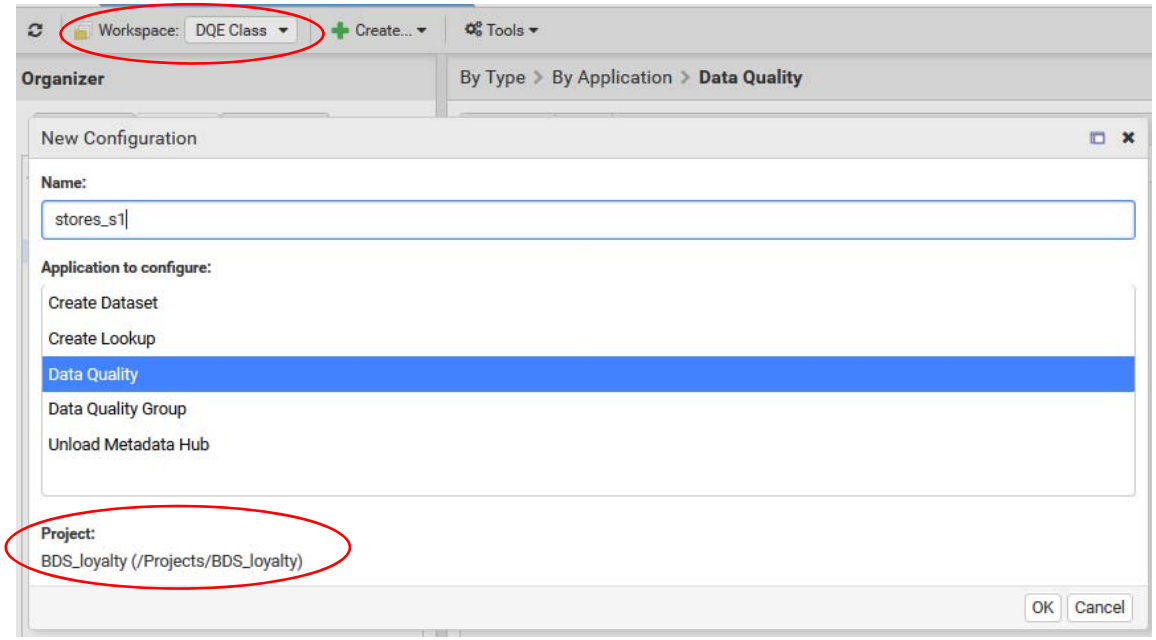


Name: ***stores_<your student number>***

Workspace: DQE Class

Application: ***Data Quality***

Project: ***BDS_loyalty***



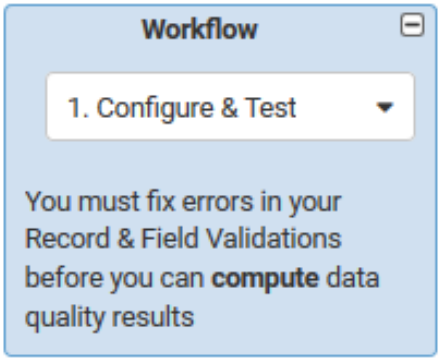
Data Quality Template

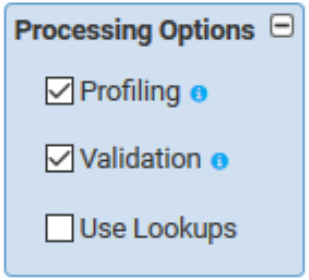
Group Exercise: Main Workflow

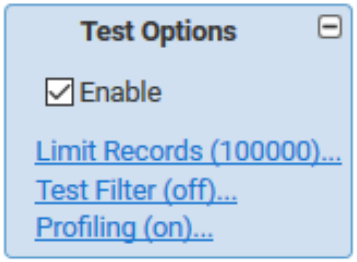
Workflow should be "Configure and Test"

Processing Options should have "Profiling" and "Validation" checked.

Test Options should be enabled, with default settings.







Data Quality Template

Group Exercise: First Data Quality

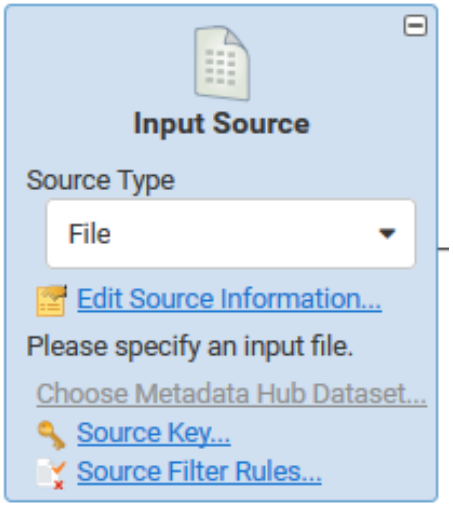
Choose a File and click Edit Source Information

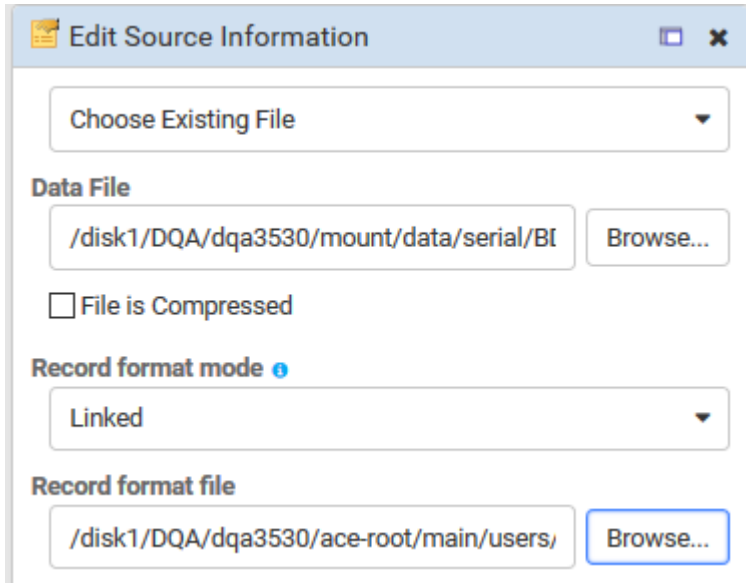
Choose “stores.dat” from lookups

Choose “Linked” for the Record Format mode

Choose “stores.dml” for the Record Format

Click “View” to see the file.

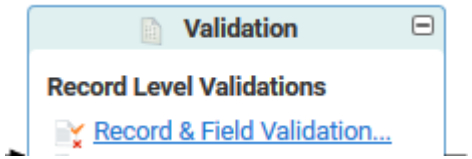




Data Quality Template

Group Exercise: Validations

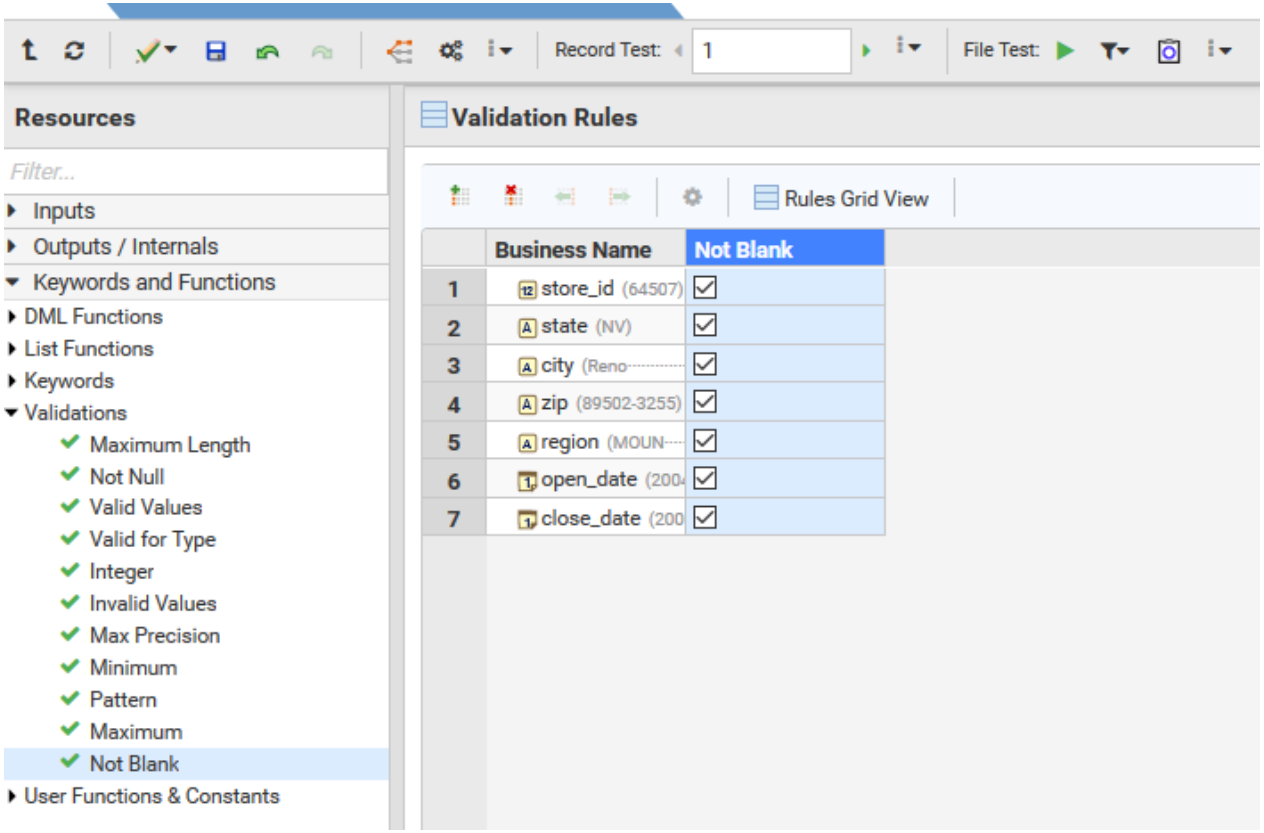
Click on “Record and Field Validations”



Add the “Not Blank” validation

Check all fields

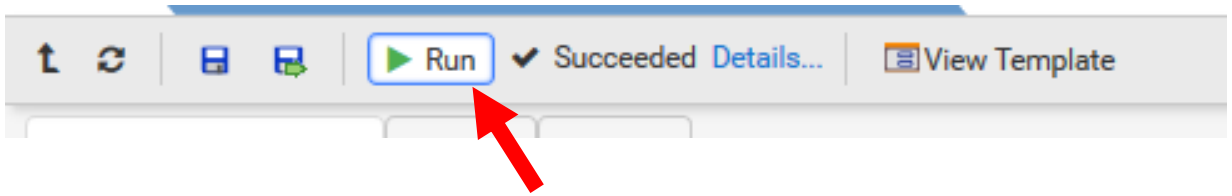
(Ctrl+space)

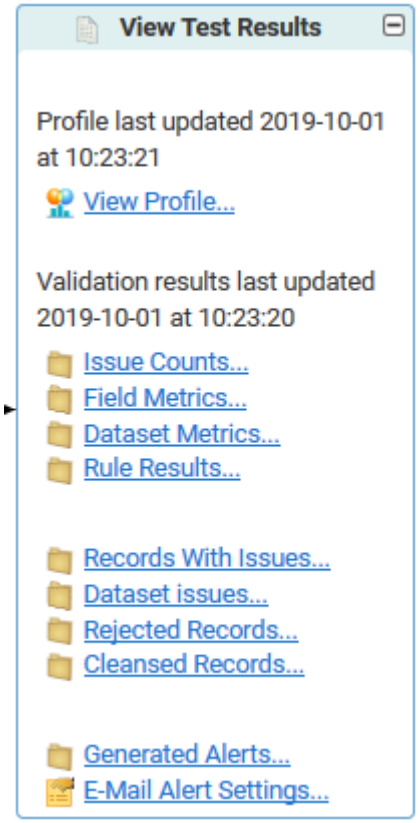


Data Quality Template

Group Exercise: Run the Configuration

Run the Configuration





Data Quality Template

Group Exercise: Summary

Create a new Data Quality Application Configuration named ex1_<your student number> in the BDS_loyalty project.

Enable Test Options with default options

Configure an Input Source to be the file lookups/stores.dat

Use "Linked" for the Record Format mode and choose "stores.dml"

Test the source configuration by clicking "View"

Click on "Record and Field Validations"

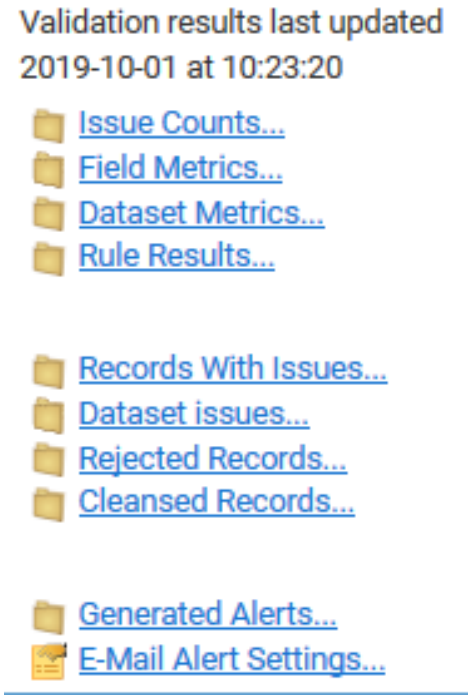
Double-click on the "Not Blank" validation and check all fields

Run the Configuration using the "Run" button on the toolbar

Click refresh to update the results

Data Quality Template

Group Exercise: Viewing Results



How many fields are empty?

How many records have at least one empty field?

How many different issue codes are generated?

Which metrics will be calculated for the dataset?

Is this a meaningful test for all the fields? Does it give a good view of overall data quality?

Data Quality Template

Testing Methods

Run Entire Configuration

These files are viewable by anyone who opens the configuration and will persist until the next time you run. They contain results that are ultimately loaded to the Metadata Hub.

Configure and Test Run Button produces a full set of results

Rules Only

Each set of rules can be tested independently within the editor. These results are temporary and rerun each time.

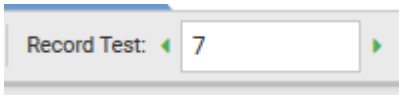
Record Testing results of running rules on a single record.

File Testing results of running rules across the selected dataset.

Data Quality Template

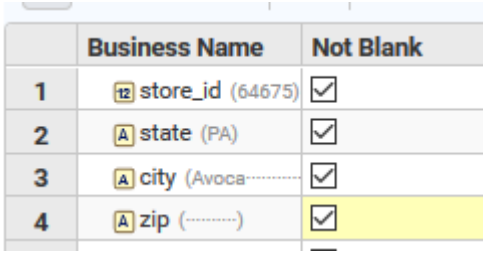
Record Testing

The Reference Data and the Rule Grid display results for the current record.



The “current” record can be changed:

Triggered tests will be highlighted in yellow



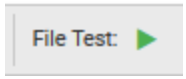
You can control when record tests run – by default they run whenever you make a change.

Why would you want to turn Record Testing off?

Data Quality Template

File Testing

File Testing runs the entire dataset, but only applies the validation logic from the current ruleset.

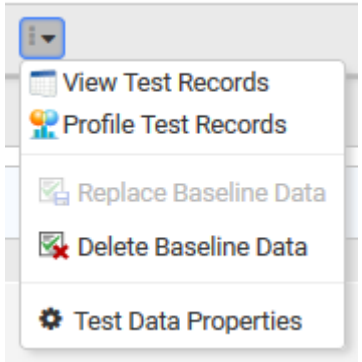
In order to run, click 

from the Rule editor.

You can apply filters for the test 

Once run you can view the results and the summary 

- Results will display the values by input record in the output format.
- You can directly filter by Issue Code or Field Name
- Using the Filter limits testing to records with a particular characteristic.



Exercise 1: Loyalty Data Quality

Create a new Data Quality configuration in the DQE.

This configuration should be named: loyalty_<*studentnumber*>

Add a file dataset loyalty.dat and loyalty.dml

Make sure the workflow is set to Configure and Test

Click on "Record and Field Validations"

Check if any fields contain blank or null values

Run the Configuration

Exercise 2: File Testing

Use the configuration that you just created.

Click Record and Field Validations

Run a File Test

Is there a 1-to-1 relationship between an output record and input record (that is, exactly 1 output for 1 input record)? Why?

Is there a validation you can remove that will change that?

Remove the extra tests.

Run again, do you see the output records change? Why?

Data Quality Template

Exercise 3: Min & Max Values

Continue using the same configuration.

Set the minimum value for est_income at 0

Set a maximum value for annual_purchases at 10,000

Verify your results using File Testing.

What is the first record that fails the test for est_income?

What is the first record that fails the test for total purchases?

Run the full configuration

How many records fail the test for est_income?

How many fail for annual_purchases?

Which metrics are impacted by the new tests?

Understanding the Data

Find statistics and other details about the dataset

Data Quality Template

Exploring the Data

Once the Data Quality template is initially configured, it provides an opportunity to explore the underlying data before importing the results to the Metadata Hub.

This gives you the opportunity to:

- **Confirm the results of specified tests.**
- **Explore data for other validations**

To view the records with data quality issues in Metadata Hub, your administrator needs to install Records With Issues in your environment.

Data Quality Template

Understanding the Data

View Data

- Displays the actual data, record-by-record
- Allows sorting and searching

Profile Results

- Displays field-by-field
- Summarizes and highlights interesting values

Data Quality Template

Profile Results

Data Profiling performs tests automatically that can help to understand the data and write better validations.

- **Range of values**
- **Unique fields**
- **Common and Uncommon**
- **Patterns**
- **Invalid for type**
- **Diversity of Data Values**

The Data Profiler report in the Metadata Hub and under View Results will include any explicit validation tests under "Validity Reasons" on each field.

Data Quality Template

Record Format

The Data Profiler validates fields against the provided format. This is similar to the "Valid for Type" test.

Type

- **Text field: string**
- **A string representation of a number: decimal**
- **Binary numbers: real, integer, packed decimal**
- **Dates: date, datetime with format mask**

Length

Character Set

Nullable

Data Quality Template

Null Value Validation

If the underlying date type is nullable, values that meet that definition will be categorized as NULL rather than valid or invalid.

Which of these values are valid for the following types? “”, “898”, “ABC”

not-nullable string?

not-nullable decimal?

nullable string?

nullable decimal?

Data Quality Template

Field Value Diversity

A unique value occurs *once and only once*

A distinct value is a particular value that occurs *one or more* times within the dataset.

Consider a list of one million customers:

- *There will be one million unique Social Security Numbers.*
- *There will be 50 distinct state names.*

Consider the field n:

- *What are the distinct values?*
- *What are the unique values?*

n
7
8
1
3
0
1
2
1
0
0

Data Quality Template

Patterns in Data Profiler Results

Patterns provide exact abstractions of the data:

Data Profiler reports include common patterns in the data:

- Digits: 9
- Letters: A (or a for case-sensitive test)
- For any other characters: the character itself

Actual Value	Pattern
$1,900.00	$9,999.99
Cf2001-50268	AA9999-99999
John Smith	AAAA AAAAA
Case-sensitive	Aaaa-aaaaaaaaa

Data Quality Template

Examining Records with View Data

In supported contexts, a magnifying glass will open up a View Data dialog to look at subsets of records.

For example, you can look at the full context for an unusually common or uncommon value.

Magnifying Glass

Pennsylvania	3,751	4.18%	issue: AB_ERR_NOT_IN_DOMAIN (M"StateName")

Data profiler view data

Fetch 100 | Filter... | Showing 100 records

	statename	sex	birthday	phonenum	email
1	Pennsylvania	F	1932-07-05	878-766-3855	parthenia_
38	Pennsylvania	M	1930-06-29	717-882-6690	eroque8484
76	Pennsylvania	F	1929-11-04	610-145-2959	jaymie_kot
90	Pennsylvania	F	1948-02-26	724-911-7021	glendajoyn
116	Pennsylvania	M	9999-12-31	835-757-2801	dfortenber
144	Pennsylvania	F	1972-06-18	717-746-6511	arthur_ree
157	Pennsylvania	M	1911-07-21	267-863-1475	jstickney@
220	Pennsylvania	F	9999-12-31	835-748-4138	penney_juv
229	Pennsylvania	F	1977-11-09	814-712-3398	monica_bla
263	Pennsylvania	F	1943-08-28	570-268-2309	roxie.huck
299	Pennsylvania	F	1931-10-14	835-646-0277	tdefeo4092

Navigating Profile Results

Profile Results have many views to allow you to find the results you need.

- **Summary of information by full record type**
- **Highlight of "Interesting" fields based on characteristics**
- **Detailed information about a particular field**

Each view is useful for different types of information -- you will rarely use all views for all fields in a record.

Data Quality Template

Exercise 4: Understanding the Profile

Open your existing configuration.

Investigate the Profile results and find examples of the following:

- **A value invalid for type.**
- **A field containing a value in all records.**
- **A field with a small number of values.**
- **A field with a consistent pattern describing the data.**
- **A field invalid based on one of the rules you've written.**

Are there any issues with state?

Is there an obvious key field? Are there any quality issues?

More Sheetview Validation Tests

Valid for Pattern, Type, and Length

Data Quality Template

Refining tests

Based on the results of the last exercise, we can further refine our understanding of Data Quality.

What are the tests we need for the following fields?

- **home_store**
- **status**
- **Own_or_rent**
- **id**
- **zip**

Data Quality Template

Syntax for Patterns in Sheet View

String patterns are simple and efficient

- Use Data Profile output patterns as a base
- Separate multiple patterns with “,”

Case-insensitive string patterns **S”AA99”,S”AAAAA-9AA”**

- Type **S** and a quoted pattern

Case-sensitive string patterns **s”AAaa”**

- Type **s** and a quoted pattern

Data Quality Template

Syntax for Patterns

Regular Expressions are supported, but are much less efficient.

- Type **r** and a quoted DML regular expression
- Use only when string expressions are not sufficient

Field must start with upper-case letter, followed by 1-3 numbers

S”A999”, S”A99”, S”A9” NOT r”[A-Z][0-9]{3}”’

Field must start with upper-case letter and 3 digits, with no other restrictions

r”\A\d{3}.*”

Data Quality Template

Validation by Type

How can we check that the data in a field matches the defined data type?

Checking the box Valid for Type will validate as the type defined in record format.

What sorts of values would be problematic? Do we know of any fields that this will cause problems for based on profiling results?

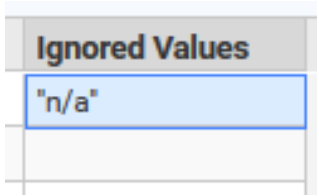
Data Quality Template

Special Columns Related to Type

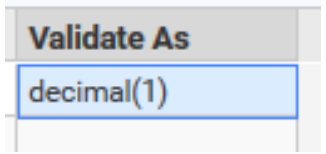
click to add “special” columns

Ignored Values Values to be excluded from validation. For example “n/a” in an otherwise numeric field.

These will usually be a “Most Common Value” in the profile results.



Validate as type Validation using dml type specified. For example, dml specifies a decimal(“|”) field, but the field needs to be validated as decimal(1)



Data Quality Template

Exercise 5: Valid for Type

Which of the following would produce a validation error if you used Valid for Type? *Hover over fields to see their data type*

- state = New York
- state = D.C.
- id = 007
- since= May 07, 2001
- home_store = BLANK
- since= 05/07/2001
- since = 2015/02/30
- own_or_rent = L

Data Quality Template

Valid Values

What if we want to limit the actual acceptable values in a field beyond patterns and type?

Data Quality Template

Provide a list of valid values

Double click on the edit pen, when available, to open pop-up editing window.

Each value should be entered on a separate line, exactly as it appears in the data.

Data Quality Template

Group Exercise: Planning

Using the Profile Results, identify two fields that can be validated using valid values.

What characteristic are we looking for?

Where do we get the list of values to validate against?

Based on the profile results, are there other validity tests to perform on these fields?

Data Quality Template

Exercise 6: More Validations

Look at the profile results and add additional validations

- **Add validation for the two fields that have a list of valid values**
- **Add validation for two fields using a pattern**
- **Revise empty value tests to give reasonable results.**

If you validate each field for type, will any generate invalid values?

- **Do any fields need to have a "Special Value" added?**

Test the ruleset to make sure the validations work as expected.

Data Quality Template

Exercise 7: Data Quality Testing

In your configuration, we added validations for the following fields, based on the class discussion and profiling results.

- **status**
- **own_or_rent**
- **id**
- **zip**
- **home_store**

Run the Data Quality configuration and look at the results. Where do you see the results of the changes?

Data Quality Template

Review Questions

What are the outputs for your validation test?

What has changed as you have added additional tests? Where can you see the changes?

What is the easiest place to see a summary of data values?

What is the easiest place to see how many times a particular error was found in the data?

If your input data are perfect without any data quality problems, what would you expect to see in the file test result?

Issue Codes and Metrics

Express>It and Metadata Hub

Data Quality Template

Issue Codes and Metrics

Before we write more complicated rules, we should look more closely at what we are producing.

Sheet View provides default values for Issue Codes, but we can specify our own or different values, if we want.

Why would we want to?

Data Quality Template

Metrics

Ultimately, we are producing metrics – the default metrics are:

- **Accuracy** Value must be in a specified list of values
- **Completeness** Value is not NULL or blank
- **Conformity** Value is valid for its data type, and in the correct format or pattern
- **Consistency** Value is consistent when compared to a) another value in the same record; b) another value in a different dataset or c) the previous version of the same dataset
- **Integrity** Foreign key value matches known primary key and other referential integrity tests
- **Stability** Distribution of values is consistent with historic distributions

Data Quality Template

Data Quality Metrics

Issue Codes from Field Level validations are combined to calculate metrics.

Each metric represents a certain aspect of the data quality.

These metrics are assigned a score on a scale from 0 – 100.

- A score of 100 means that 100% of the records processed had no data quality problems (failed to meet any Error code criteria);

- A score of 90 means 90% of the records processed had no data quality problems and implies that 10% of the records have data quality problems.

Metric scores are calculated at field and dataset level.

Data Quality Template

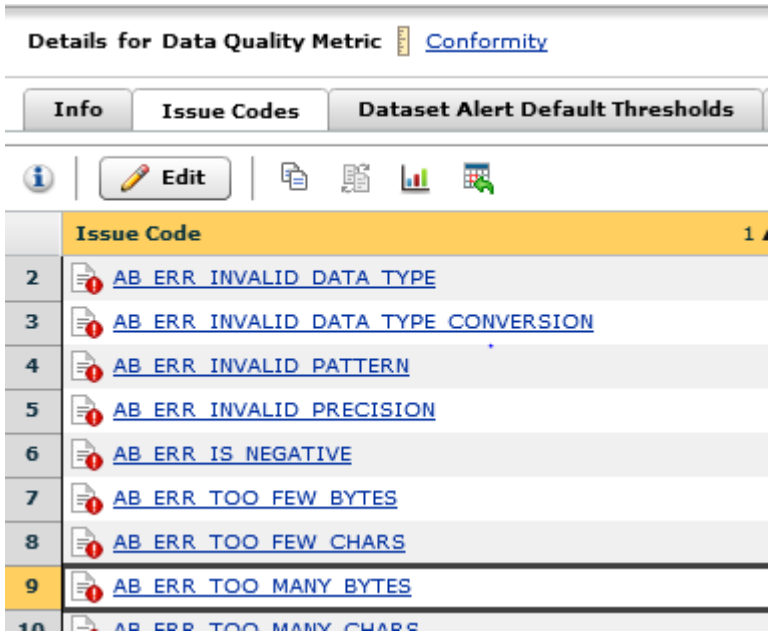
How Metrics Are Calculated

Example 1:

Name Completeness is 40% (2/5)

Income Completeness is 60% (3/5)

Dataset Completeness is 20% (1/5)



Name	Income
IS_BLANK	
IS_BLANK	IS_BLANK
	IS_NULL
IS_NULL	

Name	Income
	IS_BLANK
IS_BLANK	IS_BLANK
TOO_MANY_CHARS	TOO_SMALL

Data Quality Template

Issue Codes

Metrics consist of sets of Issue Codes.

Each Validation Rule is associated with an Issue Code.

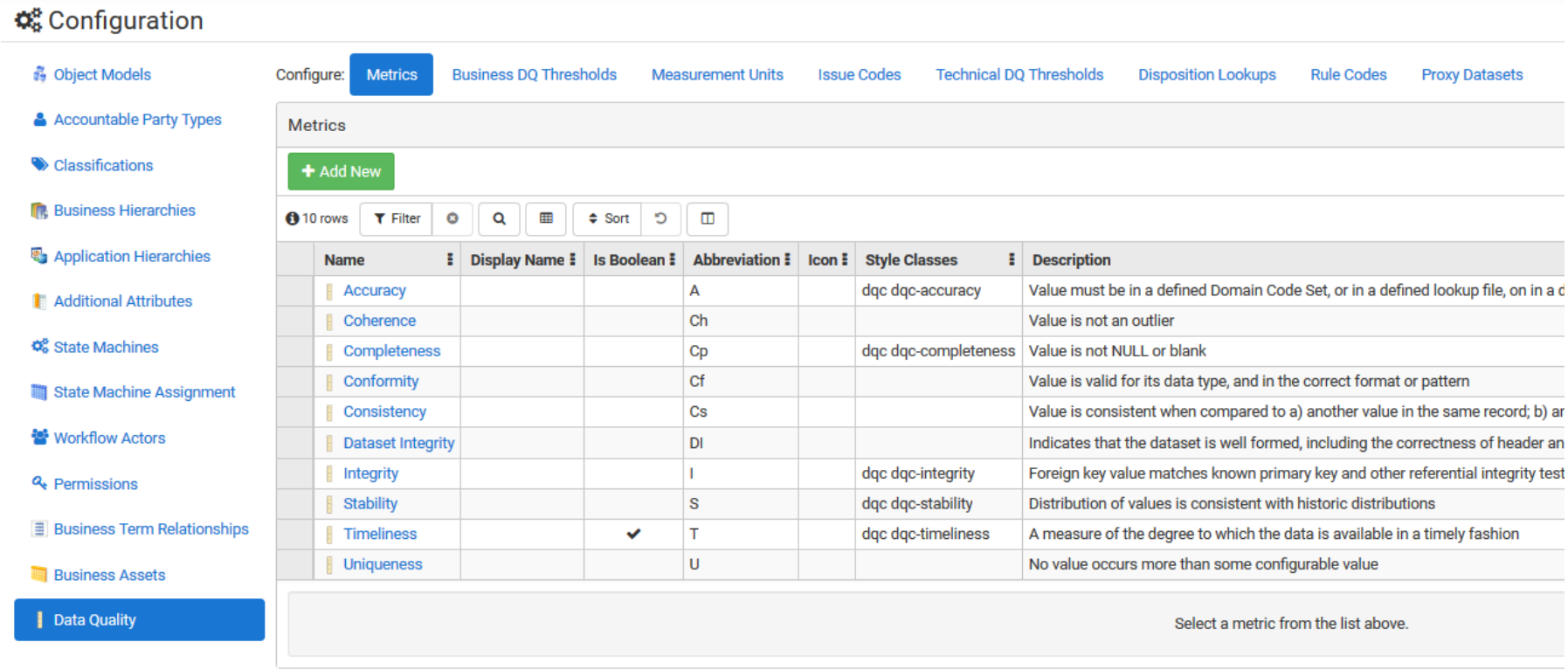
For a rule to be valid, it must be associated with an Issue Code in the Metadata Hub.

This information is maintained in the Metadata Hub and unloaded by Express>It.

Data Quality Template

Issue Codes & Metrics in Metadata Hub

Metrics and Issue Codes are found under the Data Quality Topic in the configuration page



Data Quality Template

Validation Functions

Each Sheet View validation function is assigned an Issue Code, documented in the pop-up or in the Help.

Functions	Issue code
Integer	AB_ERR_INVALID_DATA_TYPE
Invalid Values	AB_ERR_INVALID_DOMAIN
Max Precision	AB_ERR_INVALID_PRECISION
Maximum	AB_ERR_VALUE_TOO_LARGE
Maximum Length	AB_ERR_TOO_MANY_BYTES or AB_ERR_TOO_MANY_CHARS
Minimum	AB_ERR_VALUE_TOO_SMALL
Not Blank	AB_ERR_IS_BLANK
Not Null	AB_ERR_IS_NULL
Pattern	AB_ERR_INVALID_PATTERN
Valid Values	AB_ERR_NOT_IN_DOMAIN
Valid for Type	AB_ERR_INVALID_DATA_TYPE

Data Quality Template

Overriding Issue Code

What if we want to associate negative values in est_income with AB_ERR_IS_NEGATIVE instead of AB_ERR_VALUE_TOO_SMALL?

What will this change? How can we find out?

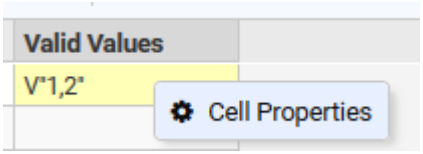
Data Quality Template

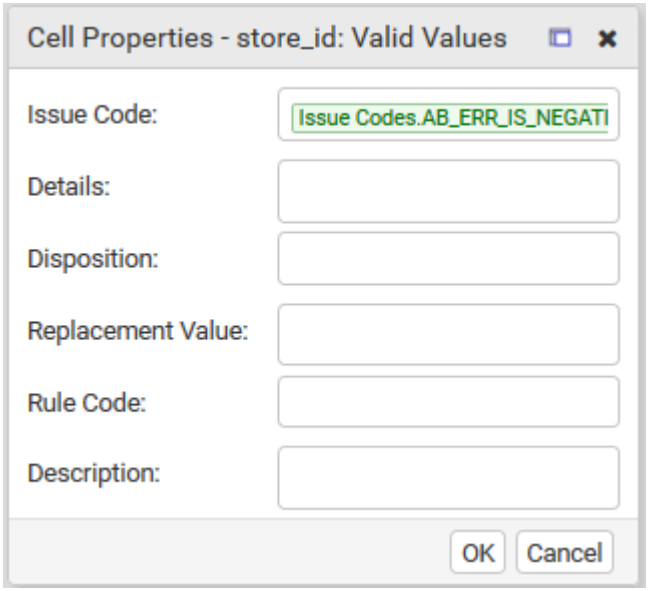
Overriding Validation Rule Results

Validation Rule Properties allow you to provide override values for the results of each validation rule These are the values reported in the results.

Issue Code must be one of the Issue Codes already in the Metadata Hub.

Right click on a cell to get to the Cell Properties





Data Quality Template

Exercise 8: Overriding Issue Code

Use an override to use AB_ERR_IS_NEGATIVE to describe incomes below 0.

How many records will this impact?

Rerun the configuration.

What changes do you see in the metrics?

Importing into the Metadata Hub

Data Quality Reports

Data Quality Template

Importing into the Metadata Hub

Running from Express>It will save the last run of metrics and profile results to disk. But, only the most recent set is saved.

The Metadata Hub is used to store historical results and make the results viewable by a wider audience.

The "Review and Import" workflow mode is used to do this.

What sort of results might we want to see?

How have we specified how results should be grouped?

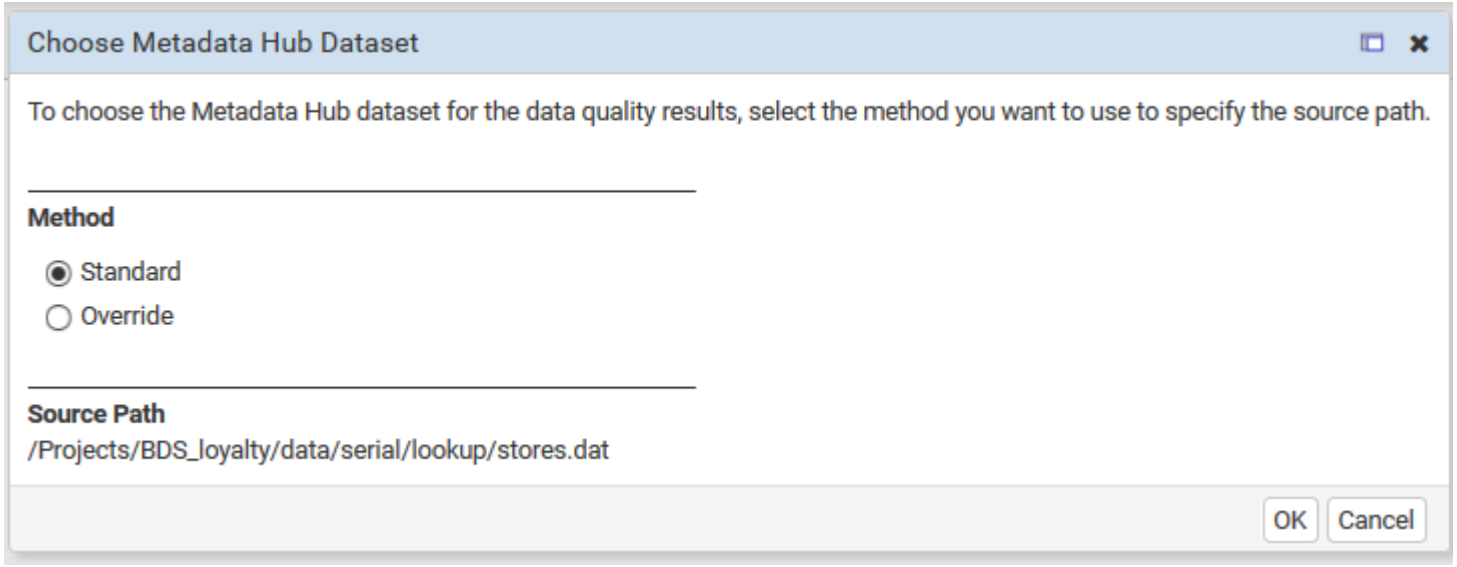
Data Quality Template

Metadata Hub Datasets

All Data Quality results are associated with a logical dataset.

Metadata Hub datasets can be created in a number of ways.

The Data Quality template uses the Data Source's technical repository path as the default location, but you can override it.



Data Quality Template

Metadata Hub Datasets

A few choices your administrators may make:

- **Only allow results to be imported if the dataset already exists in the Metadata Hub**
- **Use a path other than the default path for the datasets**

We generally recommend requiring that the datasets already exist – this guarantees that the datasets have proper lineage and that accidental datasets are not created.

Data Quality Template

Metadata Hub Dataset Override

Overrides are used to align the Data Quality results with already existing metadata in the Metadata Hub.

Why wouldn't you accept the default?

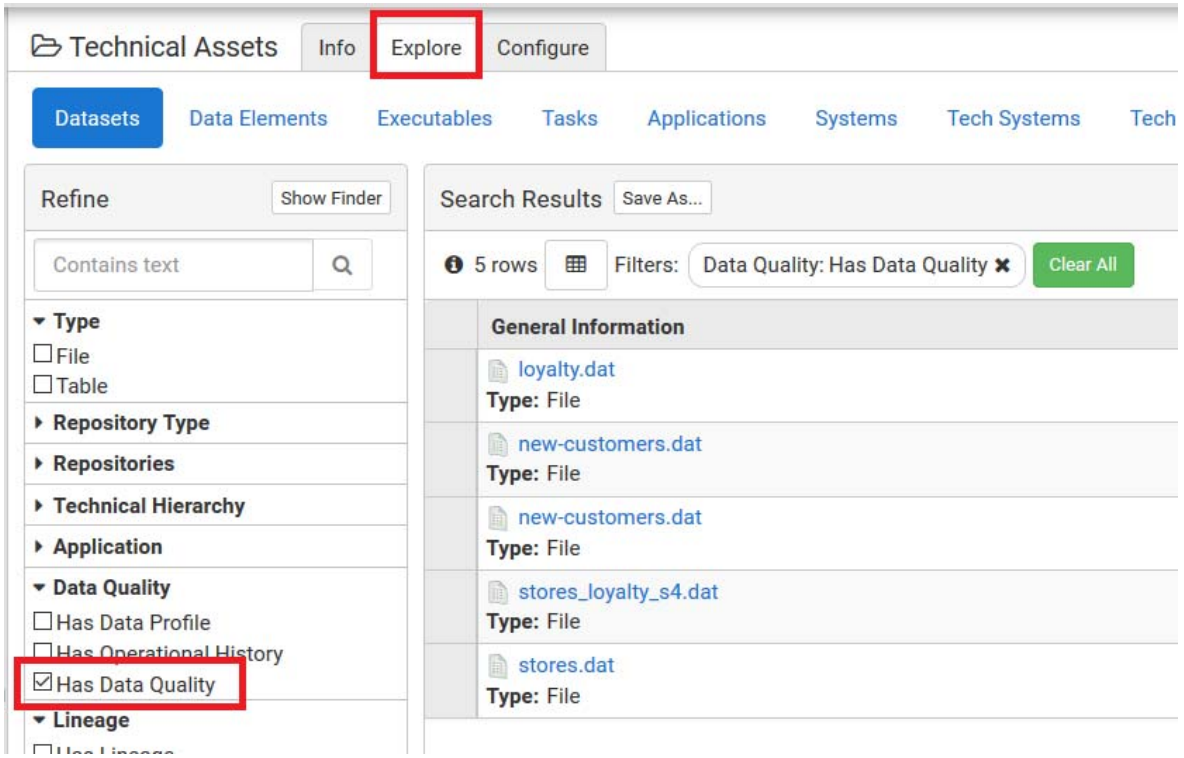
- **datestamped file to be loaded against "generic" value**
- **SQL Query with multiple tables**
- **Metadata Hub dataset does not exist and want to create in specific path**

Data Quality Template

Metadata Hub Datasets

Data Quality Reports are most often viewed by Metadata Hub dataset.

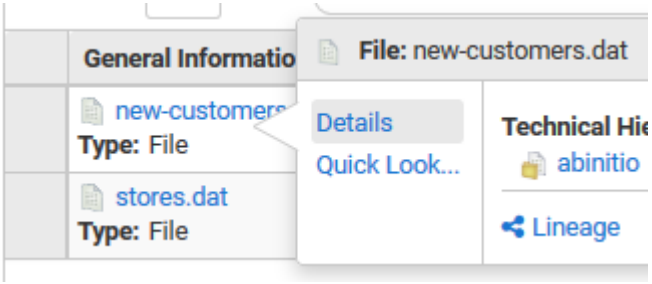
To view available reports, go to 'Technical Assets' 'Explore' tab and select "Has Data Quality"



Data Quality Template

Metadata Hub Datasets

Choose the dataset, click Details and choose Data Quality



File new-customers.dat

Info
Dataset Profile
Dependencies
Data Quality
Versions
Permissions
Lineage
Tags Edit
No tags assigned

Browse by: Metrics Rules Issues

Measured Date: 10/1/2019 Compare Against: Nothing to compare

22 rows Filter Sort

Field Name	Business Term	Accuracy	Completeness	Conformity	Consistency
DATASET		94.51	69	98.29	88.1
credit_card				100	
active_flag		99.998		100	
middle			69	100	
phonenum				99.995	
email				98.83	
state		99.997	100	99.998	
first			99.994	99.998	
monthly_payment		99.52		100	
birthday		99.963	99.998	99.997	
street			100	100	
card_expiration				99.998	99.50

Data Quality Template

Exercise 9: Importing Results

We will each import to our own logical dataset for the purposes of the class, e.g. stores_loyalty_s<student number>.dat

1. **Open the Metadata Hub**
2. **View the report: Has Data Quality**
3. **Which datasets currently have reports? Does yours?**

Exercise 9: Importing Results

Override the dataset to have your student number, e.g.
/Projects/BDS_loyalty/data/serial/main/stores_loyalty_s9.dat

Disable Test Options and rerun your configuration.

Verify the results are as you expect.

Change the Workflow to Review and Import.

Run to Import.

wait until complete – this could take some time.

Find and verify that your results are loaded. Where and how will you find them?

Data Quality Template

Review questions

What happens when you click on the Run button?

in the “Configure & Test” workflow?

in “Review & Import” workflow?

What happens when you click on Save and Publish?

Why do we have to re-run the configuration after disabling the test options?

What happened if you forgot to override the dataset name?

Rules View

More complex validation logic

Data Quality Template

More Complexity

There is an additional requirement from our retention department.

We need to ensure that customers are continuing to make enough value in purchases to meet the designated status and that we aren't failing to give benefits to customers who have made recent increases to their purchasing. We also want to flag Bronze customers showing more than 16k in purchases for potential fraud.

Our status chart is:

Status	Purchases
Gold	>5000
Silver	>3000
Bronze	>1000

What does this look like logically?

Can we do this in sheet view?

Data Quality Template

Ruleset Validation vs Sheet View

Sheet view is quick and easy, but not enough for certain tests.

Rulesets provides greater flexibility and ability to customize validation tests:

- **use values from multiple fields**
- **calculate a value and use it in multiple rules**
- **Implement complex conditional logic**

Rules require more manual effort and may not be as efficient -- use Sheet view when possible!

Data Quality Template

Rules View

Validation rules always produce the same output record type.

Rows ▾ Columns ▾ Cells ▾ Rule Options ▾

	Triggers		Outputs						Rule Case Comment
	Multi-fire Rule								
	in.state (NV)	in.open_date (2004-09-28)	Issue Code ()	Details ()	Field Value ()	Disposition ()	Replacement Value ()	Rule Code ()	
1	"WA"	<"2002-09-09"	AB_ERR_VALUE_TOO_SMALL		state				

In this example, we are testing two values:

state, open_date

We are providing output values ONLY for the required fields.

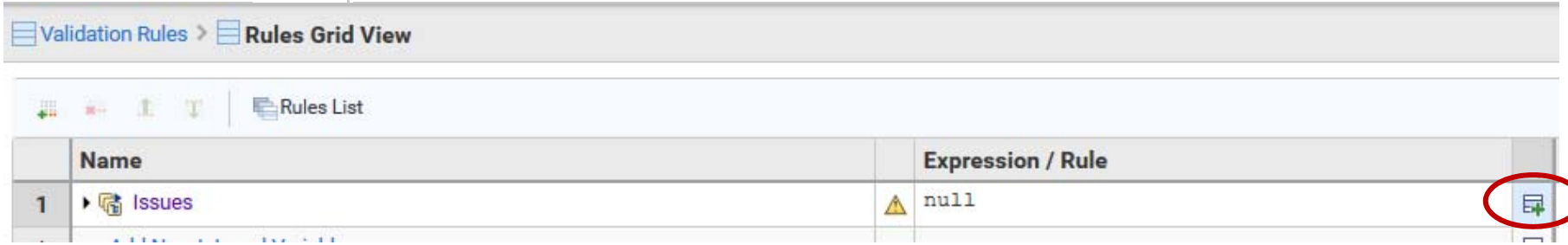
- **Issue Code from the list of available codes**
- **Field Value *single field to report issue against***

Data Quality Template

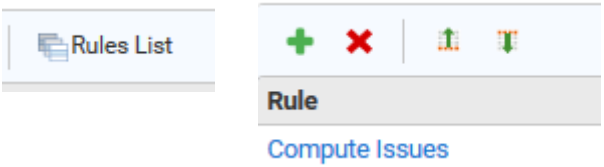
Creating A New Validation Rule

A ruleset can have one or more rules.

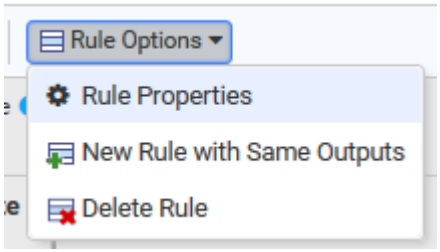
Use “ ” button to add a validation rule.



Subsequent validation rules can be created in the Rule List view



Rule Properties (in Rule Options) inside each rule allows you to provide a name



Data Quality Template

Rules

Rules are written in a tabular format.

Each value to be tested is in its own *trigger* column.

Each output value has its own *output* column.

	Triggers		Outputs					Rule Case Comment
	in.state (NV)	in.open_date (2004-09-28)	Issue Code	Details	Field Value	Disposition	Replacement Value	Rule Code	
1	"WA"	<"2002-09-09"	AB_ERR_VALUE_TOO_SMALL		state				

Each validation test is considered a *case* and is in a single row.

You can add as many triggers, outputs and cases as you need.

Data Quality Template

Conditional Logic in Rules

Each trigger can have multiple comparison cases.

Each case can have more than one trigger.

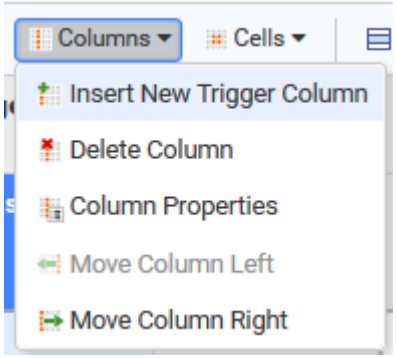
if	salary	< 10000	set	output	to	TOO_LOW
if	salary	>5000000	set	output	to	TOO_HIGH
	trigger	comparison		output		expression

The trigger and the output are the same in each case – only the comparison test and the output expressions change.

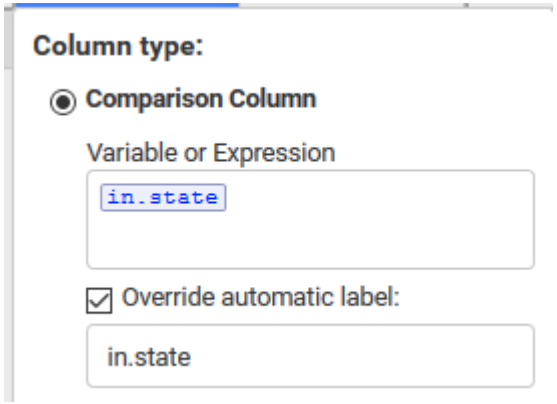
Data Quality Template

Triggers

Double-click an existing trigger to change an existing value, or add as many as you need.



Comparison columns have one value that **is used for all cases in the rule.**



Expression Columns allow you to reformat existing variables.

Data Quality Template

Comparison Cells

Each cell is compared to the trigger column value and determined to be "true" or "false"

- **Simple or compound values: 123, "UK" or "US"**
- **Named constants: Male**
- **Comparison operators: >=100**
- **Built-in functions: is_blank**
- **Complex expressions: < (Price – Discount) / 2**
- **Keywords (choose by right-clicking): *any* *(same)***

For the case to fire and assign output values, all cells must evaluate to true.

Data Quality Template

DML Functions

Each dml function has a *return type* and *arguments* which are described in the help.

date_year

Returns the year of a specified date.

Syntax

long **date_year(**date or datetime *in_date***)**

Argument	Description
in_date	A value of type date or datetime.

For more information on calling functions, see "Function call expressions" and "Calling a function with named arguments".

Details

DML represents dates internally as integer values specifying days relative to January 1, 1900. You can use date and integer values interchangeably. By default, the returned value is in 4-digit year (YYYY) format.

The function returns NULL if *in_date* is NULL.

Example

The following function call:

```
date_year((date("YYYYMMDD"))"19980518")
```

Returns the year associated with the input value **"19980518"** (May 18, 1998):

```
1998
```

Data Quality Template

User-Defined Functions

These are viewable in the resource pane alongside the built-in functions.

An administrator can create user defined functions that can be made available to DQA users

These can be DML functions used within graphs or special functions for use in DQA.

It is possible to write custom sheet view functions using a specific markup style.

Data Quality Template

Adding a New Rule

- **Create rule**
- **Decide on trigger columns**
- **Populate comparison cells**
- **Select an Issue Code**
- **Choose which value is the Field Value – if multiple input values are used, choose the one to report the quality issue against.**

Data Quality Template

Group Exercise: Suspicious Bronze

We also want to flag Bronze customers showing more than 16k in purchases for potential fraud.

- **What are the trigger columns?**
- **What are the values we are comparing?**
- **Which value should we use for the Field Value?**
- **What issue code should we use?**

Data Quality Template

Group Exercise: Suspicious Bronze

We also want to flag Bronze customers showing more than 16k in purchases for potential fraud.

1. **Create a New Rule, named “Fraud Detection”**
2. **Add the necessary trigger columns**
3. **Add the necessary values to the comparison cell**
4. **Populate the Issue Code, Details and Field Value fields**
5. **Perform a File Test to see how many records are found**

Data Quality Template

Exercise 10: Discussion

We need to ensure that customers are continuing to make enough value in purchases to meet the designated status and that we aren't failing to give benefits to customers who have made recent increases to their purchasing.

Status	Purchases
Gold	>5000
Silver	>3000
Bronze	>1000

How many trigger columns will we need?

How many cases?

Which field should go in the field_value?

Anything else we need to know before starting?

What issue codes make sense to use?

Data Quality Template

Exercise 10: Status by Purchases

Add trigger columns for status and annual_purchases

Populate comparison cells appropriately

For output, use an appropriate description and issue code.

The guidelines for each status are below – be sure to write rules that evaluate to true for unexpected values.

Status	Purchases
Gold	>5000
Silver	>3000
Bronze	>1000

Test your rules.

Discussion: Additional Criteria

The business unit wants to do further validation of "loyal" customers and will require that they have valid contact information and aren't signing up for a one-time discount.

They have asked that any status be considered invalid if the customer:

- **Has an empty street_address or zip**
- **Has been a customer for less than 30 days.**

Data Quality Template

Date Processing

The current date can be returned with the function: `today()`

We can get the difference between two dates in days by subtracting the earlier date from today:

```
today() - since
```

Look at the DML Help for other date processing functions.

Data Quality Template

Using Expressions

- **Trigger Column** `today()-since`

  The result of the expression will be tested.

  This is easy to read, but doesn't allow easy independent testing

- **Expression Cell** `today()-since > 30`

  A fully formed expression can be tested. This is useful if we need just one value to be compared, but we still can't test it independently

What if we want to use the value, but also test that it itself is valid?

Data Quality Template

Internal Variables

To calculate the value only once, and use multiple times, we can use an Internal Variable.

- **added in Rules View**
- **can be any simple type**
- **do not appear in file testing output**
- **are assigned an expression or single value once per record.**

Data Quality Template

Exercise 11: Further Status Validation

Use an Internal Variable to store Days in Program

Add a validation to verify that the status is valid according to the logic:

- **Has a populated street_address and zip**
- **Has been a customer for at least 30 days**
- **Choose a reasonable Issue Code.**

In addition, we want to be sure that the "since" field is not newer than today and not greater than 10 years ago.

Test and Import when you are happy with the results.

Data Quality Template

Discussion: Sheet vs. Rules View

Which of these tests cannot be done in sheet view?

- **date_A and date_B must be at least one year apart**
- **zipcode can contain only digits**
- **field_Z must be present in table Z**
- **field_X must contain unique values**
- **field_Y must contain digits or “n/a”**

Enhancement with Multiple Datasets

Using Lookups

Data Quality Template

Additional Data Sources

What if we want to validate the store numbers? The state field? The purchases?

Common validity tests using other data sources are:

- **Values are valid only if they have a corresponding record in another dataset.**
- **Values are valid if they are in a list of approved values**
- **Values are NOT valid if they are in a list of values**
- **Value needed for a calculation in a complex rule**

Data Quality Template

Additional Data Sources

Data Quality Assessment allows the use of data from two different types of external sources – domains and lookups.

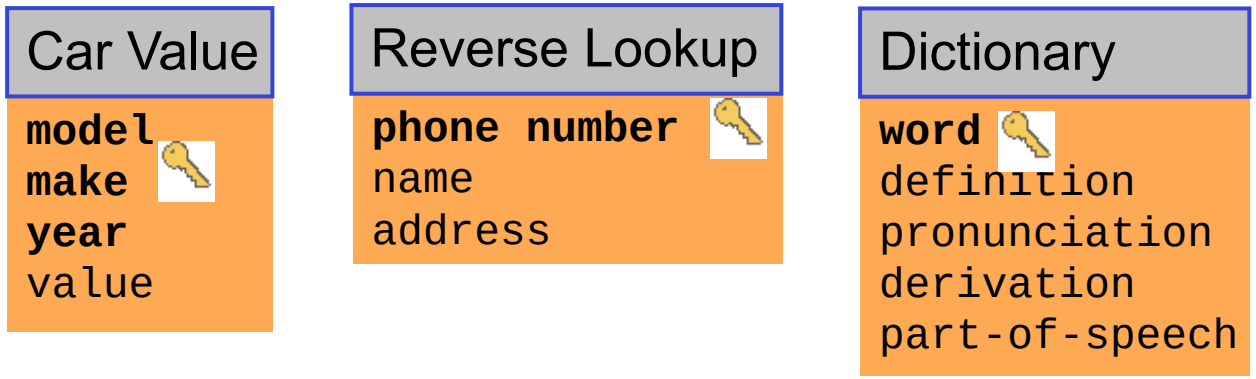
Domains are lists of values of a common type maintained in the Metadata Hub. All domains are unloaded from the Metadata Hub and made available as reference data.

Lookups are a special type of dataset defined within the Data Quality Assessment itself. They are made available using the Create Lookup template or by the administrator.

What is a Lookup?

A lookup is a dataset that contains records that can be matched with records in your incoming dataset.

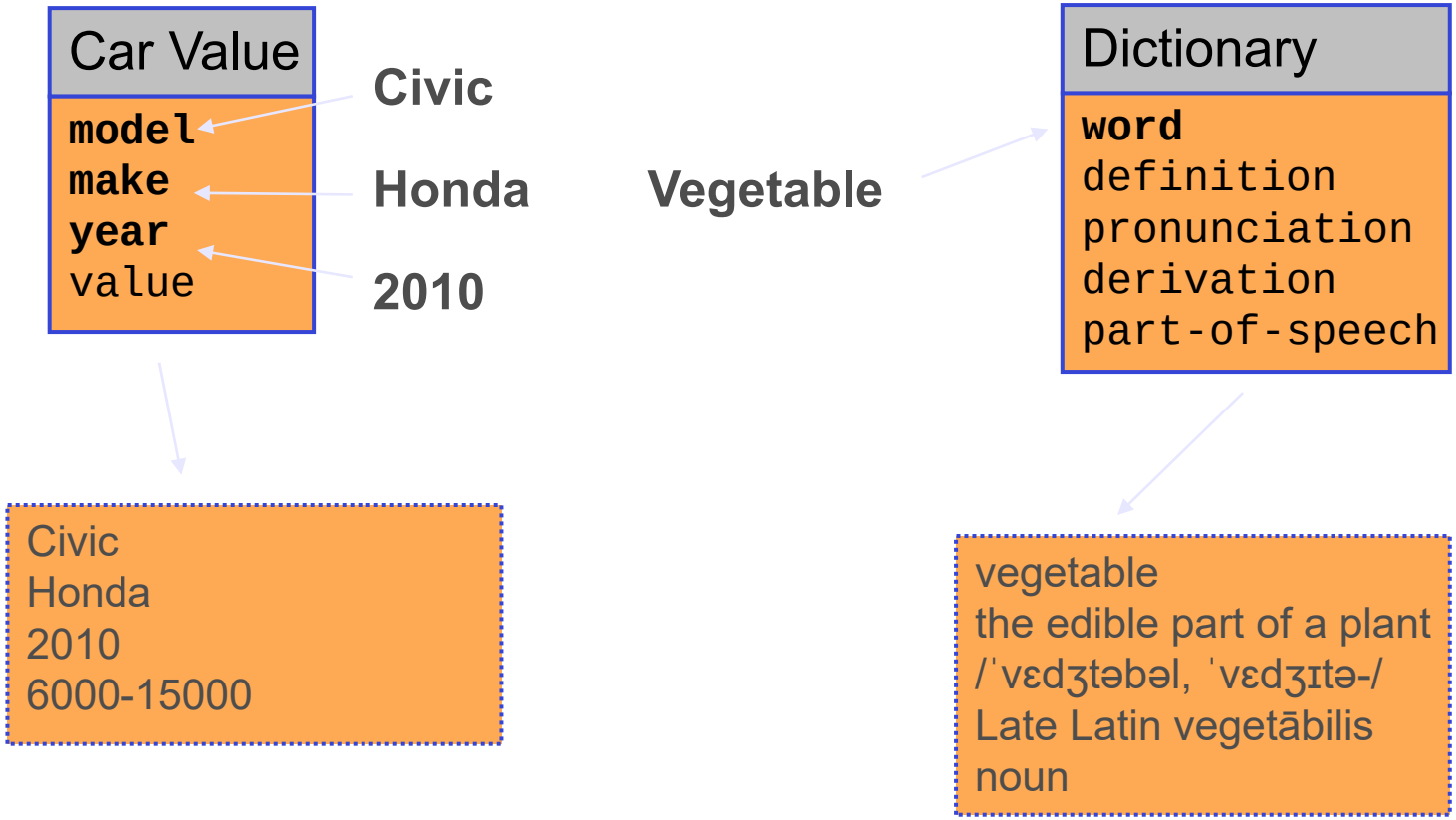
Each lookup has a particular set of fields (a key) that can be used to find the matching record.



When you use a lookup, you must provide a value for each field making up the key – the matching record will be returned.

Data Quality Template

Using a Lookup



To define a lookup, you need the name of the lookup, the data source and the field to match (the key).

Data Quality Template

Create Lookup Template

All Data Quality configurations in the same project have access to all the lookups.

Each configuration creates one lookup in the same project the configuration belongs to.

The template allows you to make changes to an existing dataset to make it suitable for use as a lookup.

Data Quality Template

Configure a Create Lookup Application

The name of the configuration will be used to identify and specify the lookup in a Data Quality configuration.

You are required to provide:

- **Source Type and Source Information**
- **Lookup Key**

Optionally, you can:

- **Provide a different format for the lookup with fewer or more fields**
- **Provide mapping rules for new fields, or changes to existing ones**
- **Remove duplicate records as defined by a key**
- **Filter the source records to contain only a subset of records**

Data Quality Template

Exercise 12: Create a Lookup

We want to verify that each home_store specified in the loyalty dataset matches an existing store.

To do this we will need to create a lookup containing information about the stores that can be used in our Data Quality configuration.

Data Quality Template

Exercise 12: Create a Lookup

Create a new configuration using the Create Lookup template.

Make sure to select the "BDS_loyalty" project.

Name the configuration stores_lookup_s*N*

Choose "File" for the Input Source.

In Source Information, use *stores.dat* with the embedded Record Format from *stores.dml*

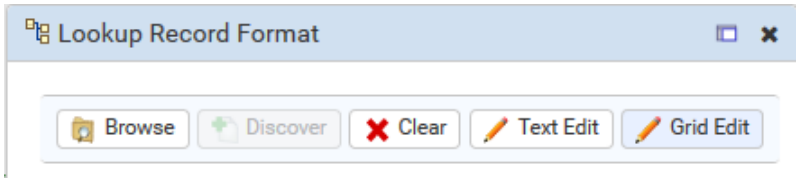
Data Quality Template

Exercise 12: Create a Lookup

For the lookup record format, create a new record format with an *8 character string* field called *store_id.*

Click on Lookup Record Format

Click on 'Clear' and then 'Grid Edit'



Say OK to Structured type

Click on "+" sign to add new field.

Provide the name and specify a fixed width string of 8 characters

Field

- **Basics**
  - Field Name: store_id
  - Type: string (Fixed width)
  - Size: 8
- **Options**
  - Default Value: *none*
  - Nullable: No
  - Hidden: No
  - Technical Comment: *none*

Data Quality Template

Exercise 12: Create a Lookup (cont)

Specify the lookup key to be *store_id.*

Make a mapping rule to map the correct field in the lookup to *store_id.*

Make sure there are no duplicates in the lookup.

Run and review the results.

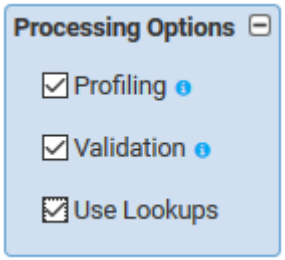
Then click on Save and Publish

Data Quality Template

Using Lookups with Data Quality

Once a Create Lookup is saved, it will be available in any Data Quality configuration in a shared project.

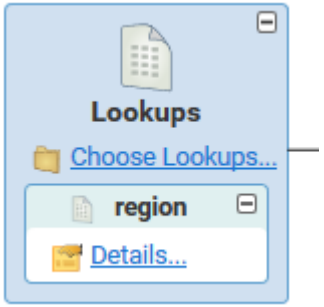
In Data Quality configuration, check the Use Lookups processing option.



Choose from the defined lookups

Choosing Multiple lookups is allowed

Selected lookup details are listed in the Lookups box.

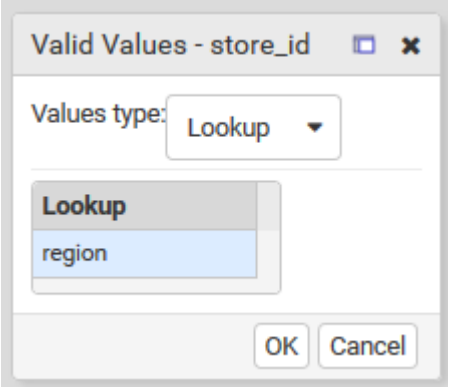


Data Quality Template

Using Lookup in Sheet View

Inside your Validation Ruleset, selected lookups will appear under Reference Data

In Sheet View, lookups are mostly used with the *Valid Value* function



Data Quality Template

Exercise 13: Use lookup in a Sheet View

Open your existing Data Quality configuration

Configure it to use the created lookup

In validation sheet view, add a new validation to test that the value in home_store field refers to an existing store.

Re-run the configuration and check whether the output contains the error code for the home_store field.

Lookups with Default Keys

Often, the key value is the same every time a lookup is used in the ruleset.

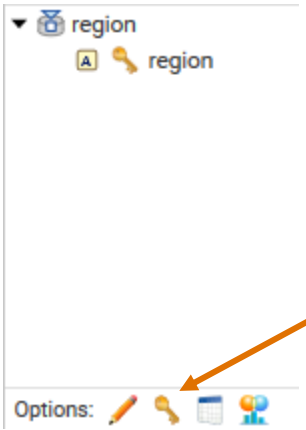
A Lookup variable can be referenced like any other if a default key is provided.

If field names and types are the same, this will be automatic

Data Quality Template

Default Keys in Ruleset Editor

If the field names between the lookup and source file do not match, you can specify a default key – it is most often a single field, but can also be an expression.



click to edit default keys

default key expression for lookup

Validation Rules > Properties > Reference Data > region > Default Key Values

Default key values: *region*

	Field	Key Modifier	Expression	Computed Value
1	region		in.region	<null>

If needed, this value can be overridden.

Data Quality Template

Exercise 14

Add a validation to test that the designated home store is still open. The Valid Values test only checks that it exists.

What change needs to be made to the Create Lookup?

What value needs to be tested?

What does it need to be compared against?

Should this be added in Sheet View or Rule View?

Implement and test the rule.

Run the full configuration.

Data Quality Template

Create Dataset Template

This template is similar to Create Lookup, but allows a dataset to be preprocessed.

Data Quality Template

Review Questions

1. When do you use lookups?
2. What is an example of a key field?
3. What is required when you configure a lookup?

Reference Data from the Metadata Hub

Create and use Lookups

Data Quality Template

Domain Code Set Values

Domain Code Sets document the range of values for a data element in the Metadata Hub.

Looking in the Metadata Hub, you can see that some data elements are associated with a “Domain” – many will not be.

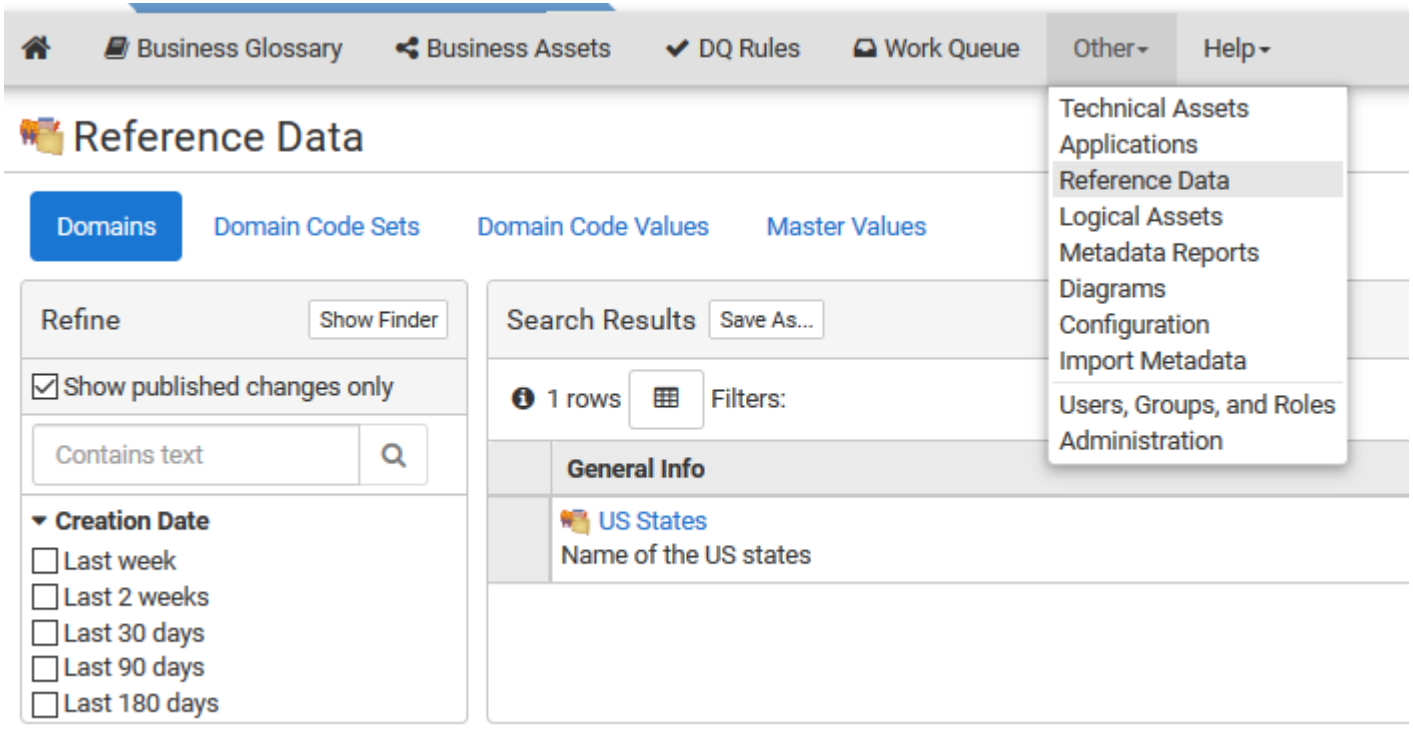
They are maintained independently of the Data Quality Environment and can be useful sources for certain types of validation.

Data Quality Template

Domains

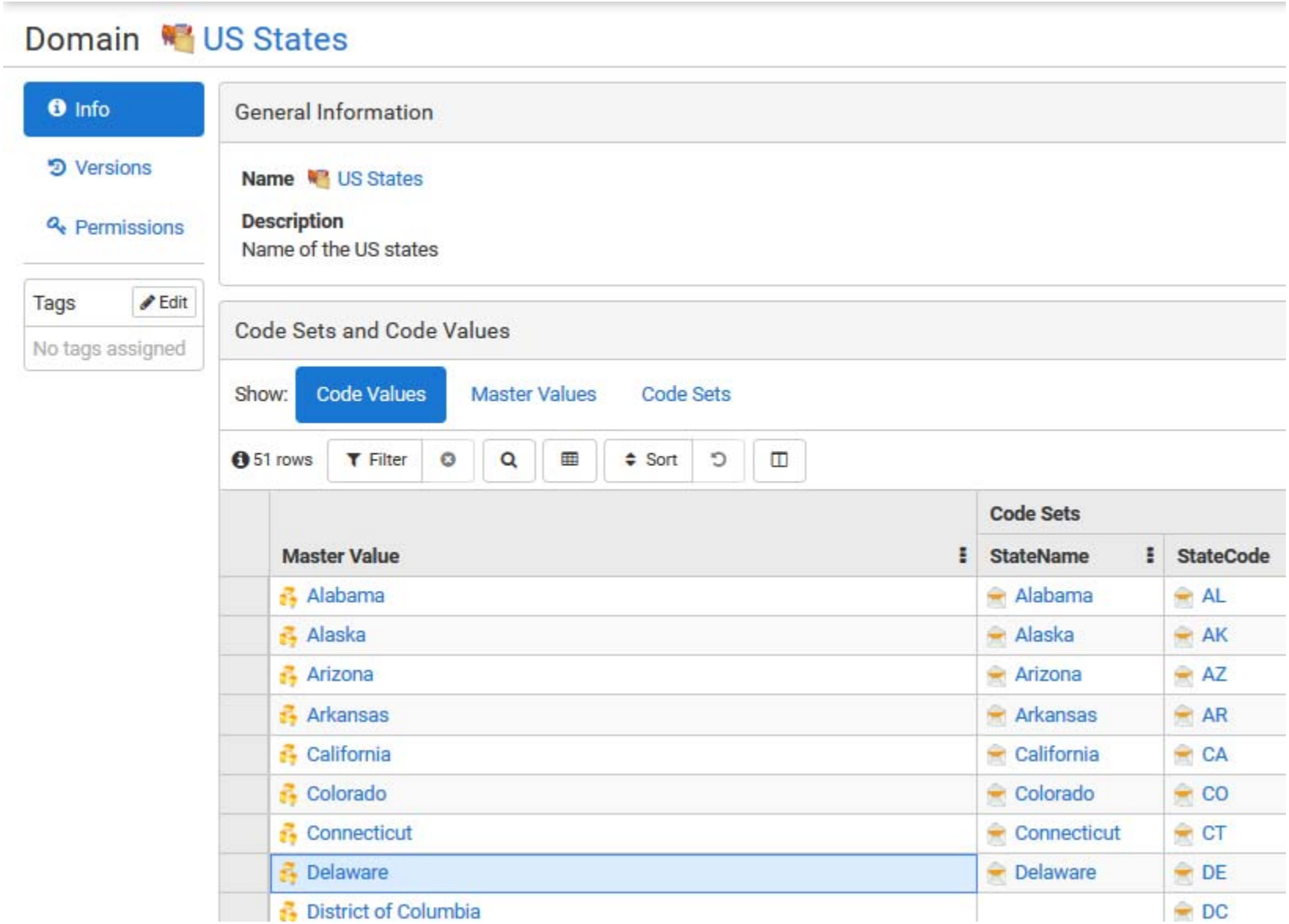
Domains are stored and created in the Metadata Hub to associated coded values with their logical definition.

Master Values contain logical names – like the full state name, or the word associated with a particular value.



Data Quality Template

Domains



Data Quality Template

Code Set

Each Code Set consists of a single list of Code Values associated with the logical Master Values for a particular domain.

Domain Value	Code Set Alpha	Code Set Numeric
Owner	O	1
Renter	R	2

For Data Quality, we only need the codeset values – those in the actual data. But, it is useful to know that "O" and "1" are equivalent in a logical context.

Data Quality Template

Code Set

Domain Code Set StateCode

- Info
- Versions
- Permissions

Tags Edit

No tags assigned

General Information

Name StateCode

Description

Domain US States

Data Type String

Required

Max Length 2

Decimal Precision

Domain Code Values

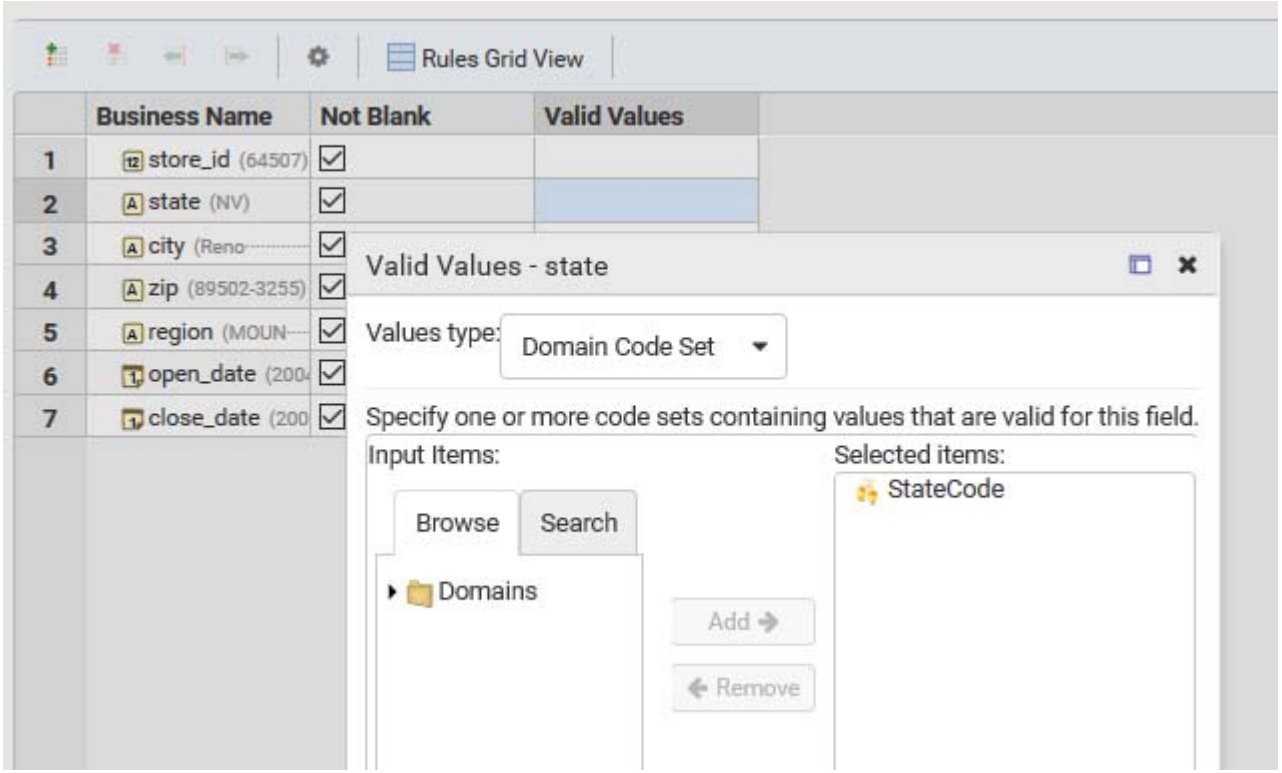
51 rows | Edit | Filter | Sort

Value	MasterValue	Active Date	Expiry Date	Is Default	Is Primary
AK	Alaska	1/1/1801	12/12/2999		
AL	Alabama	1/1/1801	12/12/2999		
AR	Arkansas	1/1/1801	12/12/2999		
AZ	Arizona	1/1/1801	12/12/2999		
CA	California	1/1/1801	12/12/2999		
CO	Colorado	1/1/1801	12/12/2999		
CT	Connecticut	1/1/1801	12/12/2999		
DC	District of Columbia	1/1/1801	12/12/2999		

Domains in Sheet View

Domain Code Sets can be used to specify lists of Valid or Invalid values in Sheet View.

The Domain Code Set does not need to be associated with the field in the Metadata Hub, although it probably "should" be, that is a separate process.



Exercise 15: Domain Value Sets

Open the Metadata Hub

Are any of the fields in the Loyalty Dataset associated with a domain?

Click on Domains to see which Domain Value Sets exist.

Go back to Express>It and use Sheet View to add an appropriate validation using a Domain Value Set.

Test and validate the change.

Data Quality Template

Exercise: Domain Review

Are there any invalid states?

Would this work if we wanted to make sure we only had customers in states where we had stores?

Are there any other fields that might make sense to validate with Domains?

When should we use domains as opposed to Lookups?

Dataset Level validation

Data Quality Template

Validation at Dataset Level

How would we validate field values across records?

- **A field must contain unique values**
- **A field must contain the same value in all records**
- **The range of dates must be no more than 30 days**

All previous tests have been within the context of a single record.

Data Quality Template

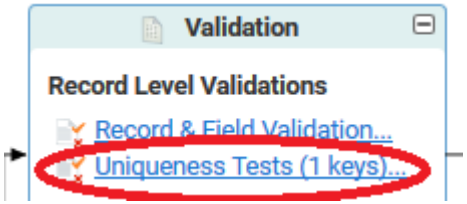
Uniqueness/Cardinality Test

This looks at the whole dataset, but generates AB_ERR_DUPLICATE issues for individual fields.

Designate the value to be tested as a key – a single field, or a combination of fields.

By default, "uniqueness" – once and only once – is tested.

The cardinality option allows values to be duplicated a specified number of times – for example, you may want to a value to occur twice.



Data Quality Template

Exercise 16: Uniqueness

Add a uniqueness test for id.

Run the test. Did you find any invalid duplicates?

Are there any other fields that might be validated this way?
What do profile results show?

Data Quality Template

Validation at Dataset Level

The Dataset Integrity metric looks at the quality of the entire dataset processed in a single run. The criteria can be more complex than simply a duplicate value.

Rather than looking at values within a single record, you can look at all values for a field across the entire dataset.

Dataset metrics will invalidate an entire dataset, rather than being a percentage based on number of valid/invalid records or fields.

Data Quality Template

Dataset Level Metrics

Dataset level metrics **measure whether the distribution of values is consistent with historic distributions.**

- Dataset Integrity
- Stability

Its error codes are calculated at the dataset level.

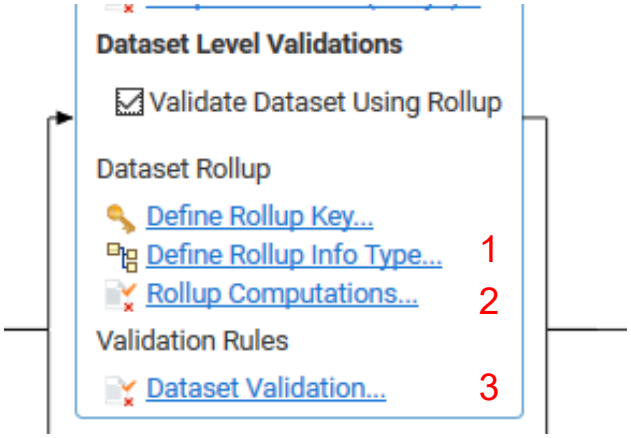
Each metric is either 0 or 1 – not the % of field level metrics.

Data Quality Template

Aggregating Values

Adding a Dataset Level Validation is a 3-step (minimum) process.

1. **Define Rollup Info Type to hold the calculated values**
2. **Specify the Rollup Computations needed to calculate the values.**
3. **Use the results to perform the actual Dataset Validation**

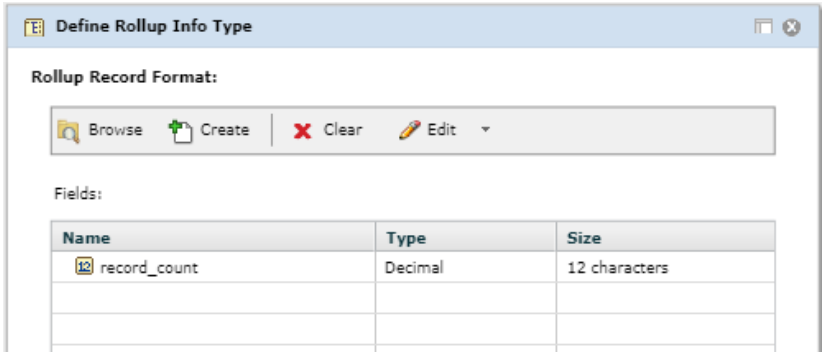


Data Quality Template

Rollup Info Type

Each value to be collected needs to be stored in its own field.

Count is provided, click “Edit” to add additional fields.



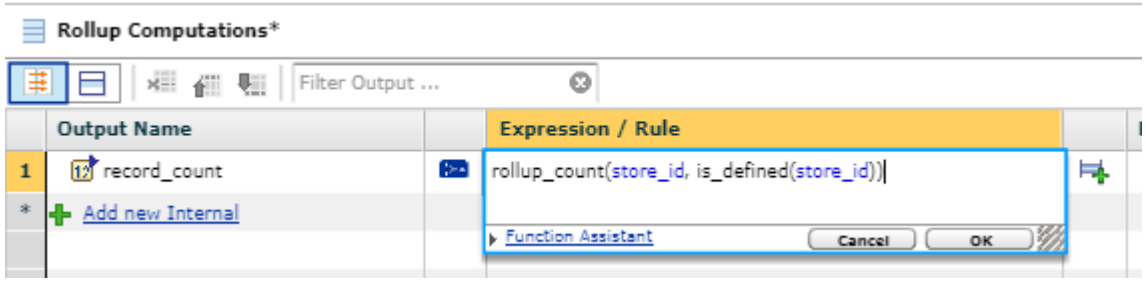
In most cases you will use one of the following:

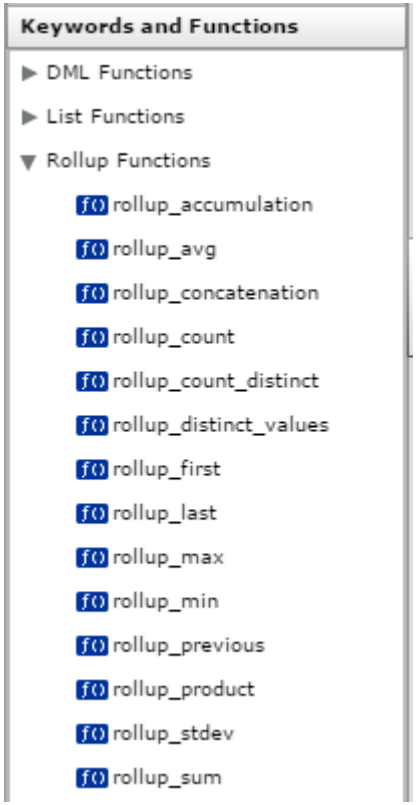
- **Decimal use for all numeric values unless otherwise known**
- **Date/Datetime provide format mask from pull down**
- **String use for everything else**

Data Quality Template

Rollup Computations

A ruleset used to populate the Rollup Info Type.





Output must be assigned using Rollup Functions.

Nearly all take two arguments (variable, condition)

- ***variable*** **is the field whose value is examined**
- ***condition*** **is optional, and allows you to include only some values in the calculation.**

Data Quality Template

Rollup Computations

Rollup functions are easiest to explain with a simple example.

Type	Value	Name
fruit	1	apple
fruit	1	pear
veg	2	lettuce
veg	4	kale

rollup_count_distinct(Type) will return 2

rollup_count_distinct(Name, Type==“fruit”) will return 2

rollup_count_distinct(Name) will return 4

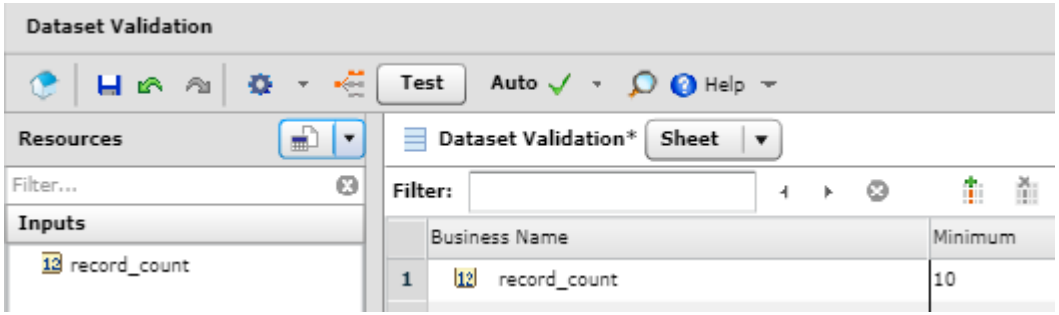
You might have both of these tests in a single record. You just need to have one Rollup Info Type field for each result

Data Quality Template

Dataset Validation

Dataset Validations work just like Record/Field Validations except you only have access to the values in Rollup Info Type.

You can use either Sheet or Rules View.



You will need to think carefully about which fields you need to test – or go back and forth between the various computations.

Data Quality Template

Group Exercise: Dataset level validation

What if we wanted to declare the dataset invalid if it contains more than 3 values (O,U,R) in own_or_rent?

What values need to be in the Rollup Info Type?

How would we calculate those values?

What does the validation look like?

Does this dataset pass?

How could we test it?

Exercise 17: Create a Dataset Level Validation

Add a validation such that the oldest date in since should be no earlier than 1990.

Hint: date_year() returns the year of any date or datetime field

- **Open the existing configuration**
- **Check Dataset Level Validations**
- **Specify the Rollup Info Type to be a date field named oldest_date**
- **Use a rollup calculation to assign it the earliest date. Be sure to check for bad dates using the rollup function condition.**
- **Configure the dataset level validation as required using AB_ERR_INVALID_IN_AGGREGATE**
- **Run the configuration and review the output**
- **When you are satisfied, import to the Metadata Hub**

Group Discussion: Dataset Validation

- **What is the difference between adding a field level validation for a minimum value and adding a dataset validation? Could you add both?**

- **If a value is tested for uniqueness, but three records have the same value for the field, how many errors are generated? Where will you see them?**

- **What is an example of something that could invalidate an entire dataset? Where would it be reported?**

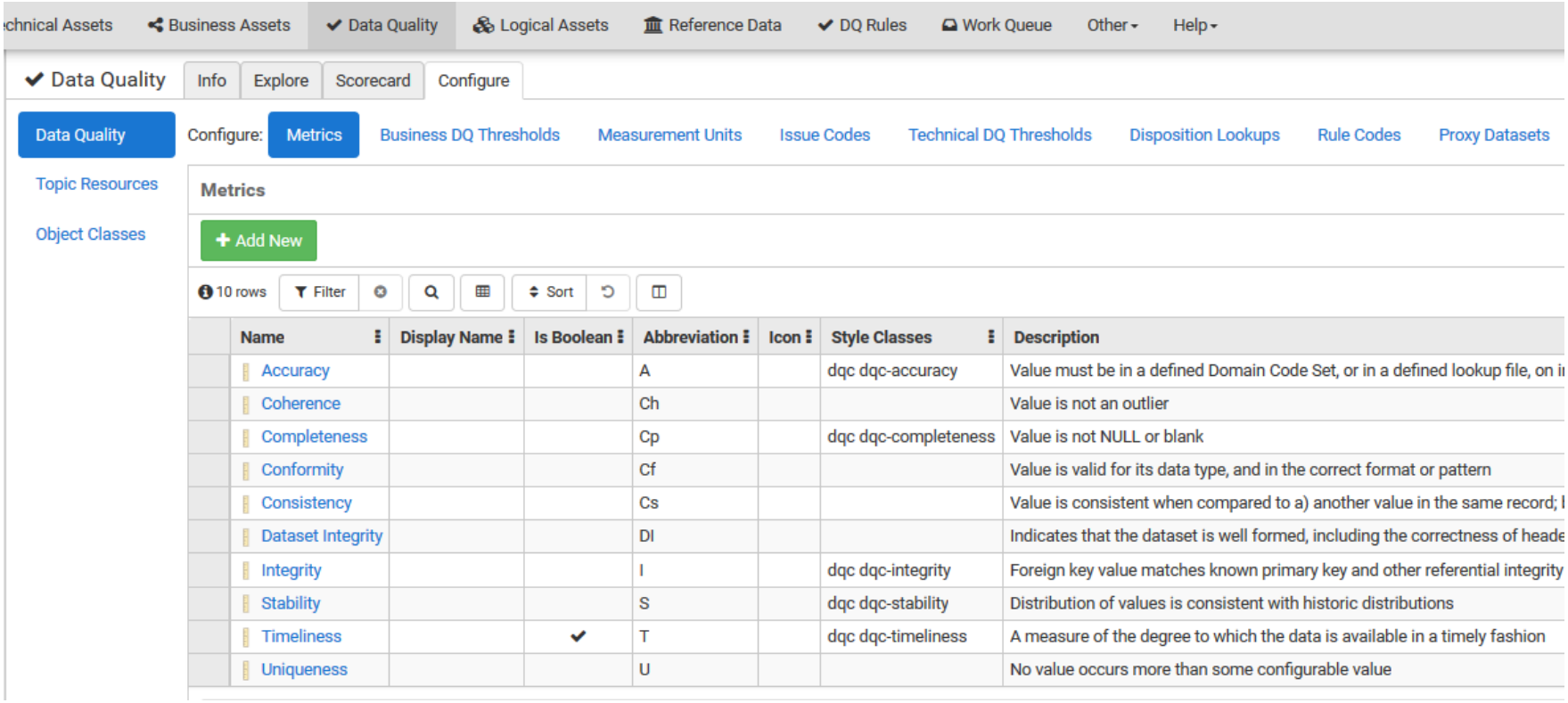
Data Quality in the Metadata Hub

Data Quality Reports are automatically generated in Metadata Hub

Data Quality Template

Data Quality Topics

To configure reference data go to the configuration page either under Other>Configuration or by loading the optional Data Quality toolbar



Name	Display Name	Is Boolean	Abbreviation	Icon	Style Classes	Description
Accuracy			A		dqc dqc-accuracy	Value must be in a defined Domain Code Set, or in a defined lookup file, on i
Coherence			Ch			Value is not an outlier
Completeness			Cp		dqc dqc-completeness	Value is not NULL or blank
Conformity			Cf			Value is valid for its data type, and in the correct format or pattern
Consistency			Cs			Value is consistent when compared to a) another value in the same record;
Dataset Integrity			DI			Indicates that the dataset is well formed, including the correctness of heade
Integrity			I		dqc dqc-integrity	Foreign key value matches known primary key and other referential integrity
Stability			S		dqc dqc-stability	Distribution of values is consistent with historic distributions
Timeliness		✔	T		dqc dqc-timeliness	A measure of the degree to which the data is available in a timely fashion
Uniqueness			U			No value occurs more than some configurable value

Data Quality Template

Navigating Metadata Hub

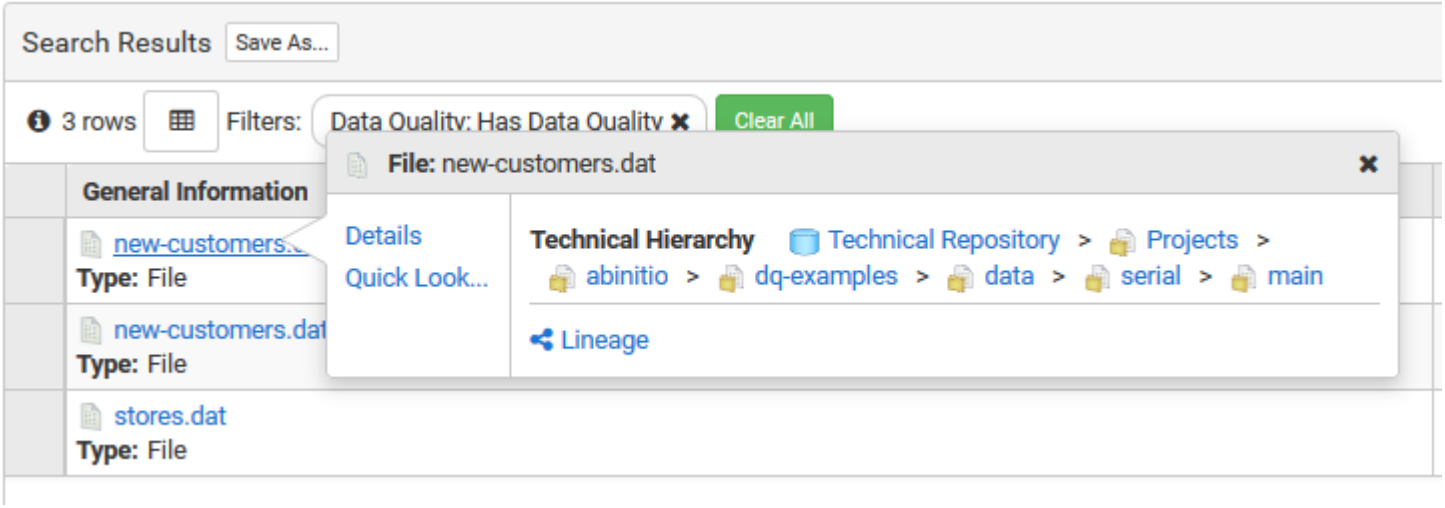
Clicking on link in Metadata Hub will open an *info bubble*

Use info bubble to navigate in Metadata Hub

Use Home button to quickly return to the main page

Details- will open a new tab

Quick Look-will open a bubble



Data Quality Template

What have we already used?

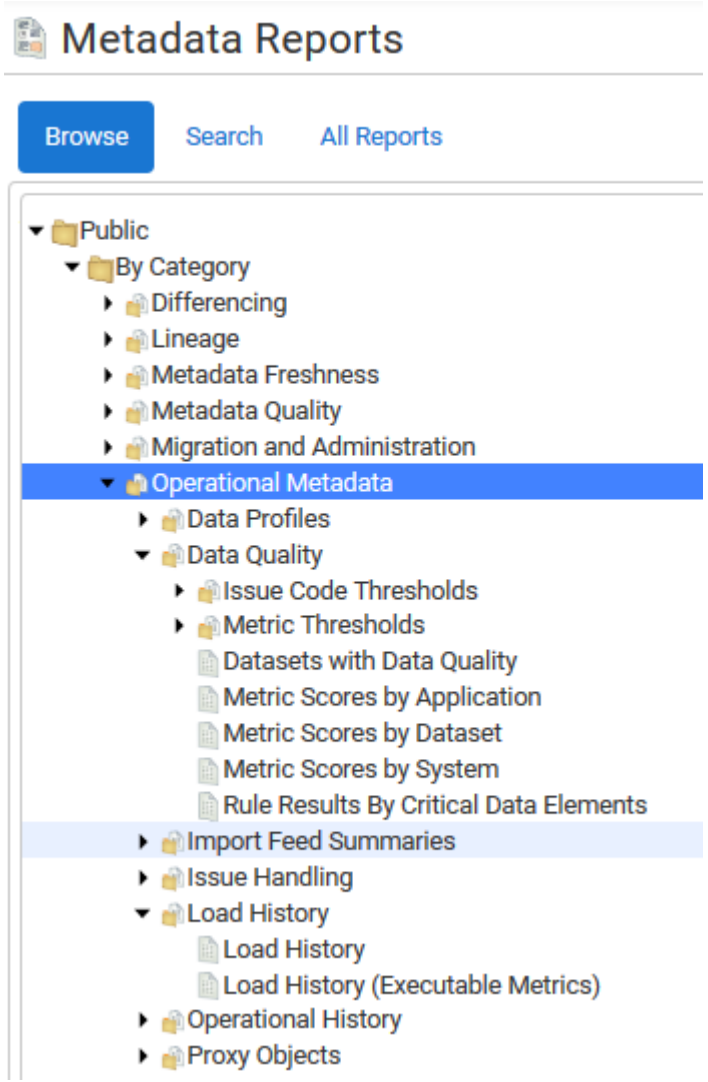
We have already been using the Metadata Hub as we write rules. Which do we already know how to view?

Where can we find?

- Domains?
- Issue Codes?
- Metrics?
- Data Quality Reports?

Data Quality Legacy reports

To get to the Data Quality legacy report go to Other > Metadata Reports > Legacy Reports
Look under Operational Metadata

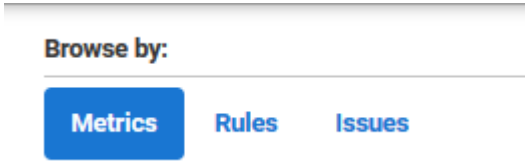


Data Quality Template

Dataset Quality Report

You can view results by

- **Metrics**
- **Rules**
- **Issues**



For Metrics you can view historical trends

You can view Profile Results on the "Dataset Profile" page of the Dataset

Using an optional feature called "Records with Issues" allows you to drill down further and see actual data.

Data Quality Template

Exercise 18: Data Quality Report

Find your student dataset and view the Data Quality Report.

How many different runs are represented?

What information is viewable here that you do not see in Express>It?

What role does Metadata Hub play in the Data Quality solution? Why can't you just use Express>It?

Data Quality Template

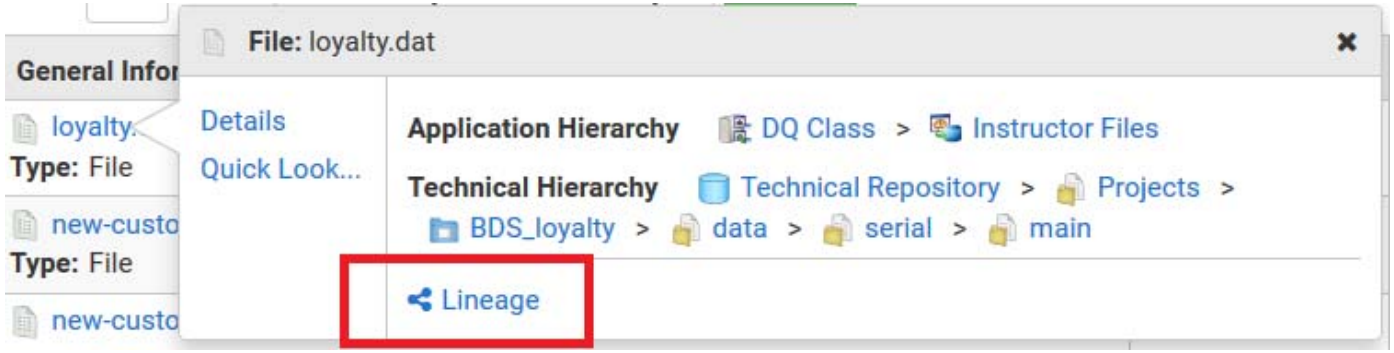
Dataset Lineage

Lineage is a feature of the Metadata Hub you can use to find information about other datasets that use this dataset.

Why is this useful in the context of Data Quality?

To view lineage, choose “lineage” from a Dataset info-bubble.

This requires configuration outside the scope of Data Quality, but is available in most Metadata Hub environments



Data Quality Template

Exercise 19: Data Quality Report

In the Metadata Hub, look at the lineage for your stores_loyalty dataset (e.g. stores_loyalty_s1.dat)

- **Looking at the history, does this dataset have any issues at the Dataset Level? Do they indicate a systemic problem or a single event?**
- **Are there any fields whose quality can be improved? What is the source of those field's values?**
- **Are any other datasets impacted by the "bad" values?**

Data Quality Template

Exercise 20: Metadata Hub Data Quality

- **Which field(s) are responsible for the low accuracy metric?**
- **Which field(s) are responsible for the low completeness metric?**
- **Which dataset(s) did the fields come from?**
- **Do we know if they came from the same record?**

Exercise 21: Looking for results

Where would you find…

- **how many datasets have Data Quality reports?**
- **what is the dataset with the lowest Completeness metric?**
- **what metric is impacted by a particular Issue Code?**
- **when was the last time a report was loaded for a dataset?**
- **which domains are available to use?**

Modifying Reference Data In Metadata Hub

Data Quality Reports, Datasets, Domain Value Sets, Issue Codes

Data Quality Template

System of Record

The Metadata Hub serves as the system of record for a variety of information related to Data Quality.

- **Datasets (metadata only -- no data is stored)**
- **Data Quality Reports**
- **Issue Codes**
- **Domain Value Sets**
- **Metrics**

To make changes, you need more privileges than to simply view the reports.

Data Quality Template

Changeset/Approval Process

As a system of record, there is a three-step process when changes are made.

1. Make the changes.
2. Submit a changeset for approval
3. Approve the changeset to make the changes viewable by all

The DQE automatically submits and approves changesets for the Data Quality Reports.

The ability to make changes and the ability to approve changes are given to users separately by assigning roles.

Data Quality Template

Error Codes/Issue Codes

Issue Codes are maintained in the Metadata Hub and unloaded into the Express>It environment before use.

To add an Issue Code, you need to know:

- **Name of issue code (all caps, no spaces)**
- **Display Name (usually the same as Name)**
- **Description (long text)**
- **Error Class (the type of error)**
- **Disposition Type**

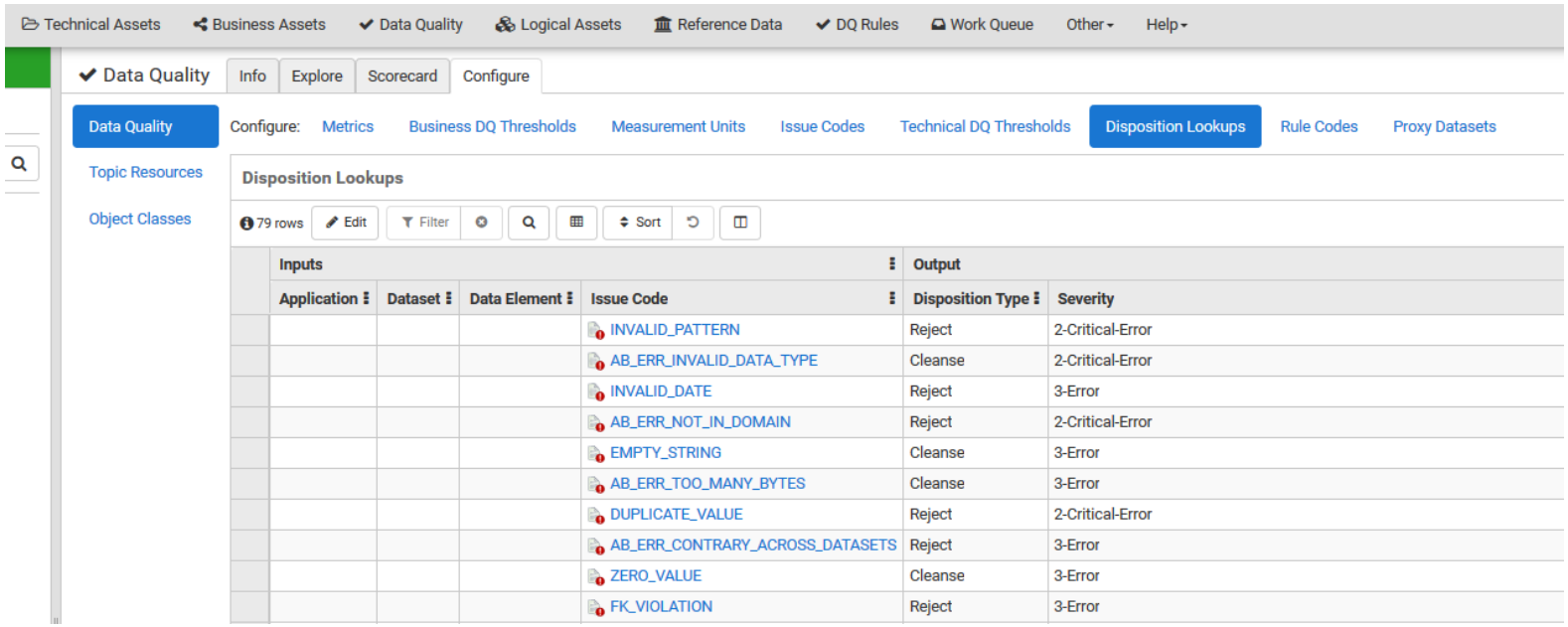
Data Quality Template

Dispositions and Severity

When we use an Issue Code, we always have the opportunity to override Disposition and Severity.

These are optional.

The default values are found on the "Disposition Lookups" tab.



Inputs				Output	
Application	Dataset	Data Element	Issue Code	Disposition Type	Severity
			INVALID_PATTERN	Reject	2-Critical-Error
			AB_ERR_INVALID_DATA_TYPE	Cleanse	2-Critical-Error
			INVALID_DATE	Reject	3-Error
			AB_ERR_NOT_IN_DOMAIN	Reject	2-Critical-Error
			EMPTY_STRING	Cleanse	3-Error
			AB_ERR_TOO_MANY_BYTES	Cleanse	3-Error
			DUPLICATE_VALUE	Reject	2-Critical-Error
			AB_ERR_CONTRARY_ACROSS_DATASETS	Reject	3-Error
			ZERO_VALUE	Cleanse	3-Error
			FK_VIOLATION	Reject	3-Error

Data Quality Template

Disposition

Disposition is used to keep track of how a particular type of error might need to be taken care of when data with that type of error is processed.

This is not used to perform any action on the data itself – but is used when reports are produced.

The standard dispositions are:

- **Cleanse**
- **Log**
- **Reject**
- **Flag**
- **Abort**

Data Quality Template

Alert and Alert Thresholds

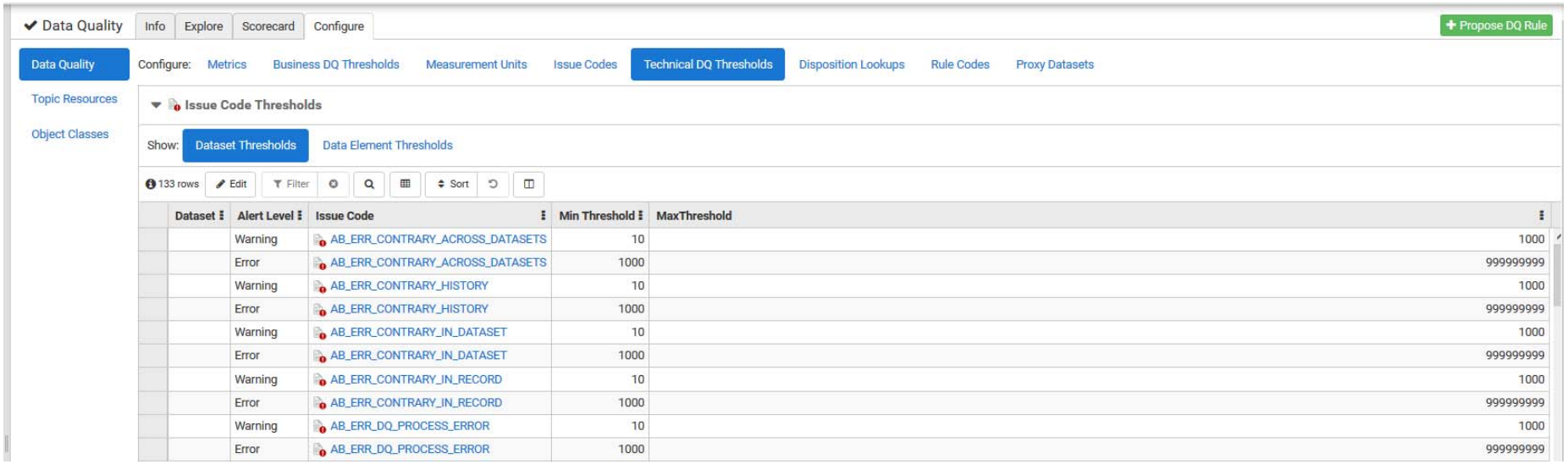
Each alert has a separate min and max threshold for Error and Warning.

96.9
99.3
82
93

Displayed on the report using different colors.

Each alert can be associated with metrics and/or error codes.

The thresholds can be overridden at the dataset or field level.



Dataset	Alert Level	Issue Code	Min Threshold	MaxThreshold
	Warning	AB_ERR_CONTRARY_ACROSS_DATASETS	10	1000
	Error	AB_ERR_CONTRARY_ACROSS_DATASETS	1000	999999999
	Warning	AB_ERR_CONTRARY_HISTORY	10	1000
	Error	AB_ERR_CONTRARY_HISTORY	1000	999999999
	Warning	AB_ERR_CONTRARY_IN_DATASET	10	1000
	Error	AB_ERR_CONTRARY_IN_DATASET	1000	999999999
	Warning	AB_ERR_CONTRARY_IN_RECORD	10	1000
	Error	AB_ERR_CONTRARY_IN_RECORD	1000	999999999
	Warning	AB_ERR_DQ_PROCESS_ERROR	10	1000
	Error	AB_ERR_DQ_PROCESS_ERROR	1000	999999999

Data Quality Template

Domains

Domains are part of the general Metadata associated with a dataset.

They may be maintained and used outside of the Data Quality Environment, so be sure to understand YOUR environment before making changes.

Domains can be added in two different ways.

- **Manually**
- **Using an Importer**

Class Discussion: Domains

When to use domains vs. regular lookups vs. manual coding?

Reusability

Use domains and lookups when code list is used in more than one place

Volatility

Domains are mostly for slow changing data since the updating it in DQA requires a separate import and unload process. Data changes frequently should be in regular lookups.

Natural of the data:

Domain code sets are typically metadata, while regular lookups mostly contain data.

Size of the data:

Domains are mostly for small to midsized data. Regular lookup can be quite large.

Final Exercise

Putting it all together

Data Quality Template

Transactions

There is a new requirement to perform Data Quality analysis on the individual transactions received from the stores.

The data file to use is transactions.dat.

It is described by transactions.dml.

The known requirements for Data Quality are on the next page, but you may need to use the data itself to understand the full implementation details.

Data Quality Template

Final Exercise

1. Every field, except for loyalty_no, must be populated
2. payment_method should be one of the known types
3. store_no should be a valid store
4. loyalty_no must match an existing customer
5. The combination of store_no, register_no, timestamp and loyalty_no should be unique.
6. Total price, discount and quantity must be greater than 0
7. Credit card (CD) payment should not be used if the total price is below $1.5
8. No transaction should be from a closed store (at the day of transaction)
9. Total number of records in the file should be between 200-500K

Implement and test these measures of Data Quality.

Administrivia

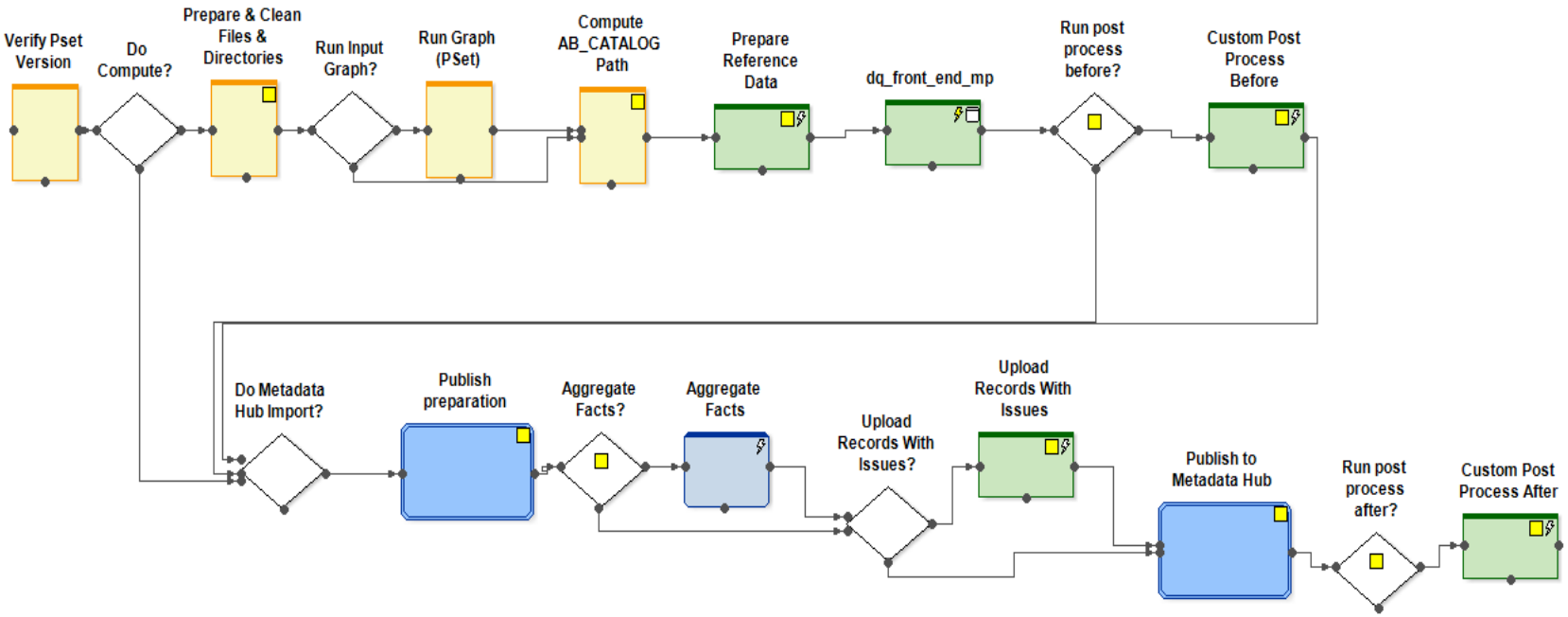
Running on the command-line, Data Quality Project

Data Quality Template

Compute and Import Plan

Each Data Quality run is executed by an Ab Initio plan. A Data Quality Configuration creates psets to drive this plan and some of the plan tasks.

During Compute and Test, only the top-row of tasks are run.



Data Quality Template

Running Data Quality in Production

The Express>It interface is used for testing and configuring jobs. This is usually done in a Development environment.

Once configured, these jobs need to be run automatically against Production data.

```
air sandbox run compute_and_import_config_name.pset
```

When run on the command-line, it will automatically compute results and load them into the Metadata Hub.

You can provide additional parameters to control the behavior.

Data Quality Template

Running Data Quality in Production

ALLOW_DS_CREATION	(true) allow creation of a Metadata Hub dataset.
ALLOW_DS_UPDATE	(true) allow update of Metadata Hub dataset
AUTO_UPDATE_RULESET_DML_CONSTANTS	(true) updates of constants for issue codes, rule codes, and dispositions in validation rulesets.
COLLECTION_TIME	timestamp of data quality test results and profiles.
CUSTOM_POST_PROCESS	custom graph for post-processing.
CUSTOM_POST_PROCESS_WHEN_TO_RUN	order of the execution of a custom post-processing graph (before or after the import of results to the Metadata Hub).
DATASET_RPATH	repository path for an EME input dataset.
DATASET_SOURCEPATH	source path of the associated Metadata Hub dataset
DO_COMPUTE	(true) compute data quality results.
DO_MHUB_IMPORT	(true) import of results to the Metadata Hub.
DO_PROFILER	(true) profile data
DO_TR_DATASET_IMPORT	(true) update create dataset to the technical repository.
FILE_IS_COMPRESSED	(false) compression of input dataset
INPUT_FILE_PATH	input file path
INPUT_RECORDS_LIMIT	number of records used for computations to the value you specify.
LOGGED_RECORDS_LIMIT	number of log records.
OUTPUT_SERIAL_DIRECTORY	output directory for the files produced by while running the configuration.
WRITE_CLEAN_RECORDS	create an output file that contains only clean records.

Data Quality Template

Data Quality Project Configuration

The Data Quality Project is a standard Public Project which contains all of the underlying graphs and plans needed to process and generate Data Quality results.

- Each environment should have a single copy of this project
- Any project containing datasets to be analyzed should include this project.
- The values of the Data Quality project parameters should never be changed in the project parameters – but should be overridden. Why?

Profile Results and PII

Data Quality Reports and Profiles never include full context records, but they do contain values.

Certain types of PII need more protection than the Metadata Hub can provide.

You can

- Configure a filter to remove or mask values from certain fields and provide the value in AIDQ_CONFIG_PROFILE_FILTER_PATH
- Set DO_PROFILER to 0 and avoid profiling altogether

Data Quality Template

Purging Results

```
mh-admin datastore purge datastore-name
        [-do-purge]
        [-sourcepath sourcepath]
        [-start-time datetime -end-time datetime]
        [-dq] [-dp] [-opds] [-opexe] [-all]
        [-purge-metrics-with-proxies-only]
        [-verbose]
```

Data Quality Template

Postprocessing Graphs

You can specify custom processing to happen before or after results are loaded to the Metadata Hub.

These are used to send email, generate particular reports or make changes to the report before results are imported.

Graphs are run as part of the plan and have access to runtime parameters that provide details of a particular run.

Data Quality Template

Promotion

The standard lifecycle is used for promoting Data Quality projects. The DQE Express>It interface is used only for Development.

Development

- **Validate and write rules against samples of real data**
- **Tag and Promote**

QA

- **Run psets on pre-production datasets**
- **Inspect results in Metadata Hub**

Production

- **Scheduled runs of compute_and_import psets**
- **Full data volumes; results imported to Metadata Hub**

Data Quality Template

Maintaining Metadata

All of the Metadata objects: Issue Codes, Metrics, Domains, etc. can be edited by hand in the Metadata Hub.

They can also be imported from external sources such as a spreadsheet.

Before starting to customize these objects, it is worth understanding where your system of record will be and how you will make sure the values stay synchronized.

Contacting Ab Initio Support

Problems, Questions,

Feedback or Concerns?

support@abinitio.com

+1 781 301 2100

+44 0 870 850 8700