Data Quality Engine (DQE)

Dartec Systems

22/06/2016

Dartec Systems

Overview

Why use a DQE for Data Quality, Cleansing and Profiling?

- It's free (if you have Enterprise SQL Server)
- Need to assess quality of data as it is imported
- Need to assess quality of data as it is processed
- Need to resolve/correct data quality issues at both import and processing.
- Produce higher quality cleansed data
- React to data quality issues
- Highlight data quality issues with data supplier
- Monitor data quality over time
- Profile datasets
- Apply minor, simple transformation
- Central Data Quality and Cleansing rules

DQE – Key Features

- Centralise data quality rules in single repository
- Easy and flexible rule configuration
- Profile, validate and cleanse data based on pre-defined quality rules
- Rules can be implemented at multiple points of ETL flow.
- Define rules once and implement many times
- Audit history of all applied rules
- Audit history of changes to all rules
- Audit history of all cleansed data
- Extension execution auditing, making maintenance simplier

DQE – Key Benefits

- Highly configurable - maximum flexibility
- Easily add, amend, delete quality rules – minimal training
- Uses SQL Server Master Data Services – minimal training
- Capture results of past validation and cleansing tests – monitor data quality over time, quickly identify data quality issues
- Resolve data quality at source – prevent issues to downstream services
- Validate data with external and internal reference sources – checks incoming data against standard and master lists.
- User Security Model – only allow access to authorised users.

Rule Types

- Value Correction Rules
- Harmonisation Rules
- Expression Rules
- Reference Rules
- Profiling Rules
- Transformation Rules

Value Correction Rules

Rules that replace identified values with preferred values

- **Text to Number**
  - Twelve to 12
  - Medium to M
- **Text to Text**
  - ‘Devon’ to ‘Devon and Cornwall’
- **Null to Zero**
  - Null to 0

Dartec Systems

Expression Rules

Uses SQL query against an evaluation column to flag specific rows

Examples

- Is Null
- = ‘’
- LEN(Name) <> '8' and Name != ''
- cast (Budget as int) > 50000000
- cast (StartAge as int) < cast (EndAge as int)

Dartec Systems

Reference Rules

Check incoming values are consistent with master or controlled lists

- **Compare Value to External Table**
  - Reference List contains: OrganisationCode = 123456
    Evaluation Dataset contains: OrganisationCode = 123456
    Result = MATCH

- **Compare Values to Internal Look Up List**
  - Reference List contains: Gender values (Male; Female; Other)
    Evaluation Dataset contains: Gender values (M; F)
    Result = No Match

Harmonisation Rules

- **To Upper**
    - Status to STATUS
- **To Lower**
    - Status to status
- **Remove Spaces**
    - 0121 323 2345 to 01213232345
- **Set Blanks As Null**
    - ‘ ’ to NULL

Harmonisation Rules - cont

- **Set Null to Default Value**
  - NULL to Not Applicable
- **Remove Specified Character**
  - £12300 to 12300
- **Replace Value**
  - 12 High St to 12 High Street
- **Special Operation**
  - Add appropriate example

Harmonisation Rules - Date Format

- **Confirm UK Date Format ddmmyyyy**
  - 12/03/2016 – Pass
  - 31/11/2016 – Fail
  - 02/13/2016 – Fail
  - 29/02/2015 – Fail
  - 29/02/2016 – Pass

- **Confirm US Date Format mmddyyyy**
  - 05/19/2016 – Pass
  - 19/05/2016 – Fail

N.B. Rule Check Output can used as Input to next Rule Check

Dartec Systems

Harmonisation Rules – Date Format

Example

Open Date = 12/11/2014

Close Date = 23/05/2016

Rule Check Process

1. Open Date - Confirm UK Date Format ddmmyyyy - Pass
2. Open Date – Transform to date type of IntegerDateTime – 20141112

3. Close Date - Confirm UK Date Format ddmmyyyy - Pass
4. Close Date – Transform to date type of IntegerDateTime – 20160523

5. Check Open Date < Close Date – (20141212 < 20160523) - Pass

Harmonisation Rules – Format Mask

- **Get Alphanumeric Mask Pattern (Run 1st)**
  - Run against Values

12345	–	NNNNN
1234567	–	NNNNNNN
1234E6	-	NNNAN
12346	-	NNNNN

- **Get Column Value Distribution Profile (Run 2nd)**
  - Run against above results for Values

NNNNN – 2 rows

NNNNNN – 1 Row

NNNAN – 1 Row

Dartec Systems

Harmonisation Rules – Format Mask

- **Get Alphanumeric Mask Pattern (Run 1st)**
  - Run against Values

12345	–	NNNNN
1234567	–	NNNNNNN
1234E6	–	NNNAN
12346	–	NNNNN

- **Run Expression Rule (Run 2nd)**
  - Run against above results for Values

    input:- where ! = 'NNNNN'
    output:- 2nd and 3rd rows are flagged

Dartec Systems

Profiling Rules

- **Max and Min Value Profile**
  - Input - 12, 1, 77, 3, 88, 1, 3, 34, 24
    Output - Min = 1, Max = 88
- **Table Row Count**
  - Total Row Count = 54, 236

Profiling Rules - Enhanced Dup. Check

- **Check for Duplicates (based on primary key defined in DQEntity)**
  - Input - 12345, 12346, 12347, 12345, 12348, 12345

    Output

      12345 – 1

      12345 – 2

      12345 – 2

      12346 – Null

      12347 – Null

      12348 - Null
- **Check for Duplicates results in two duplicate outputs (1 & 2). 1 is provisionally flagged as the value to retain whilst 2 denotes a provisional attempt to flag duplicates to be deleted.**

Transformation Rules

- **Varchar to Integer**
- **Integer to Varchar**
- **Varchar to Varchar**

- **Varchar (UK) to DateTime**
  - 25/12/16 to 25/12/2016
- **Varchar (US) to DateTime**
  - 12/25/16 to 12/25/2016
- **Varchar (UK) to IntegerDateTime**
  - 28/09/2015 to 20150928
- **Varchar (US) to IntegerDateTime**
  - 09/28/2015 to 20150928

Rule Sequencing - configurable

To achieve more complex checks or transformation, rules can be executed in a pre-defined sequence. This is useful if you need to test the quality of a value before transforming it (for example).

- **1st Tranche = Rule 1**
- **2nd Tranche = Rule 2**

Example

Input = CountyName = Corn wall

Rule 1 = Remove Spaces

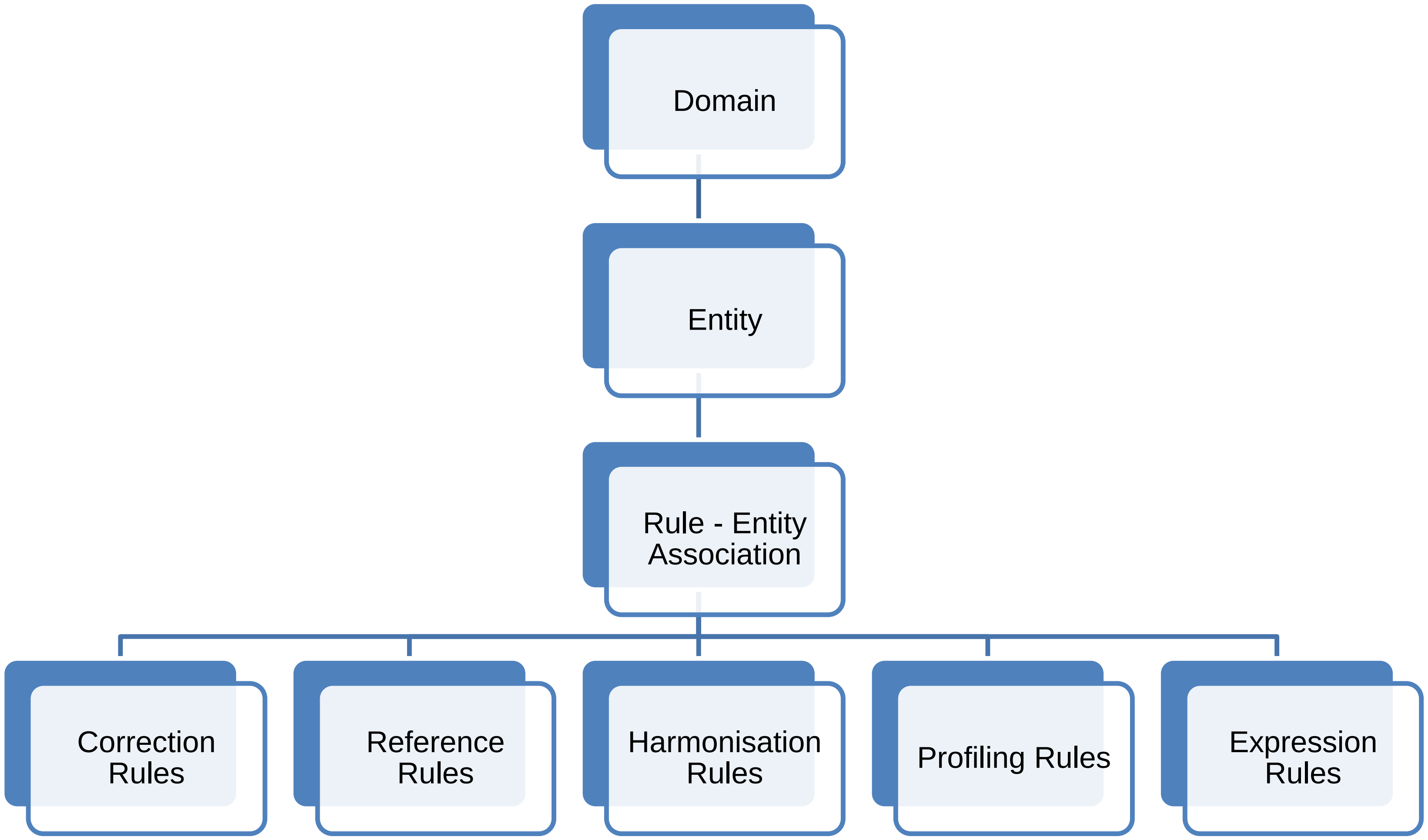
Output = CountyNameRemoveSpaces = Cornwall

Rule 2 = Set ‘Cornwall’ to ‘Devon and Cornwall’

Input = CountyNameRemoveSpaces = Cornwall

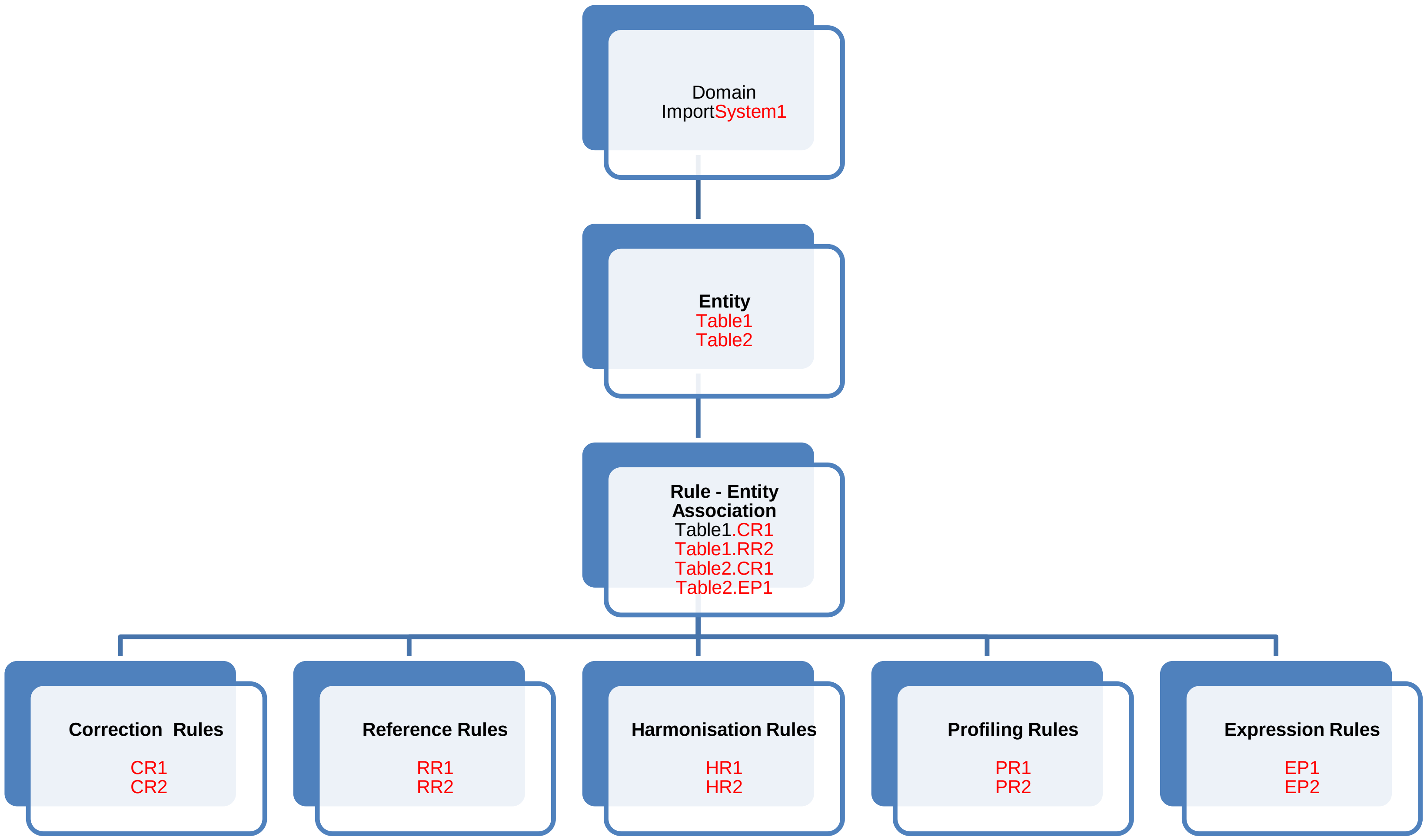
Output = CountyNameCornwallToDevon&Cornwall = Devon and Cornwall

Associating Rules



Dartec Systems

Associating Rules – more detail



Dartec Systems

Data Quality Rules execution activities

Extensive auditing is built into the solution with a number of tables present within the ‘**DQ**’, **‘Audit’** and **‘Reports’** schema:-

- **DQ.DataQualityHistory**
  Holds the summary and top level results from the rule executions
- **DQ.DataQualityRowHistory**
  Holds the row level information captured by each of the data quality rules
- **DQ.DataQualityPrimaryKeyValues**
  Holds the primary keys of all the records logged in the previous table.
- **DQ.RuleExecutionHistory**
  Holds a record of each rule (as a SQL statement) as constructed and applied to the data.
- **[Audit].[RoutineLoad]**

  Core table used to track the execution of all procedures and packages.
- **[Audit].[RoutineError]**

  Captures any technical execution errors detected at runtime
- **[Audit].[RoutineLoadHistory]**

  Created from RoutineLoad and provides a more use friendly view of the load audit data
- **[Reports].[DQSummaryResults]**

  View created to support the reporting

Dartec Systems