Flexible and Generic Data Quality Metadata Exchange

(Practice-oriented Paper)

David Becker
The MITRE Corporation
dbecker@mitre.org

John Jaster
Digital Prospectors Corporation
jwjaster@charter.net

Jereme Kuperman
Illumination Works
jereme.kuperman@ilwllc.com

Abstract: Data quality metadata frequently needs to be exchanged between various parties and tools engaged in the management of data quality (DQ). In this paper we describe a data quality metadata exchange (DQME) Extensible Markup Language (XML) approach that addresses a number of data quality management fundamentals in a flexible and generic manner. The approach provides for the exchange of XML messages covering DQ definitions, business rule evaluations and DQ measurements. Each of these types of DQ metadata exchange operates within the context of a corresponding set of processes and tools. The approach has been developed to support an Enterprise Data Quality Management Service (EDQMS) project being implemented for the United States Air Force (USAF) Operations Support community.

Key Words: Data Quality, Information Quality, Metadata, Exchange Standard, XML

Introduction

Descriptive information about data is called metadata. Metadata that describes different data quality (DQ) dimensions (e.g., the data's accuracy, precision/certainty, completeness/brevity, timeliness, consistency, and lineage/pedigree/provenance) is called DQ metadata. DQ metadata can be developed for any type of data whether it is structured databases, spreadsheets, flat file records, text files, images, audio, graphics, sensor data, log files, error reports, presentations and briefings, web pages and links, etc.

DQ metadata is needed to identify problems with the data that can cause difficulties for systems and users that depend on the data to conduct their work and make decisions. As documented in numerous studies, poor quality data can drive immense costs to an organization, and can decrease trust between partners to a business data exchange[3][6][7][8]. So, there are ample reasons to inform data consumers of the quality of the data they are receiving, and to actively manage that quality so that DQ problems are removed or prevented altogether. This permits cost and trust issues to be reduced or eliminated.

DQ metadata can be generated from many different sources (data profiling tools, DQ measurement tools, application generated error reports and log files, human generated deficiency reports, internal controls and security audits, data sampling and analysis, various forms of data operations management tools, such as extract, transform and load (ETL) systems, manual annotation, community and marketplace voting systems, etc.). These sources all help to reveal information about the quality of the data. DQ metadata can be used for a number of different purposes (data cleansing, data management and operations control, data improvement initiatives, business process reengineering, six-sigma activities, business alerts and notifications, audits and findings for financial reporting, enhancing business intelligence and analytics, improving decision making, etc.).

There are many situations during the generation, processing and presentation of DQ metadata where exchange becomes an important enabling activity. If where DQ metadata is created is different from where it is stored, used and displayed, then the DQ metadata must be packaged for exchange between the interacting functions. This is true for data operations within an organization, and on a larger scale, in Business-to-Business (B2B) information exchanges where trust relationships are paramount [8].

A USAF development team has been working on a project to implement an Enterprise Data Quality Management Service (EDQMS). This work was based on a prior MITRE Corporation Mission Oriented Investigation and Experimentation (MOIE) research project sponsored by the AF Electronic Systems Center (ESC). One key component of EDQMS is a Data Quality Metadata Exchange (DQME) XML approach based upon the principles outlined below. EDQMS and DQME are now being moved into production for the management of data quality across a broad spectrum of IPs and DQ subjects for the AF Ops Support domain. In this paper we discuss the DQME XML approach.

THE ARCHITECTURE OF DATA QUALITY

Any set of information systems through which data flows can be generically characterized [2] as an information manufacturing system consisting of producing applications that generate data called information products (IPs) that are consumed by other applications (the first column in Figure 1).

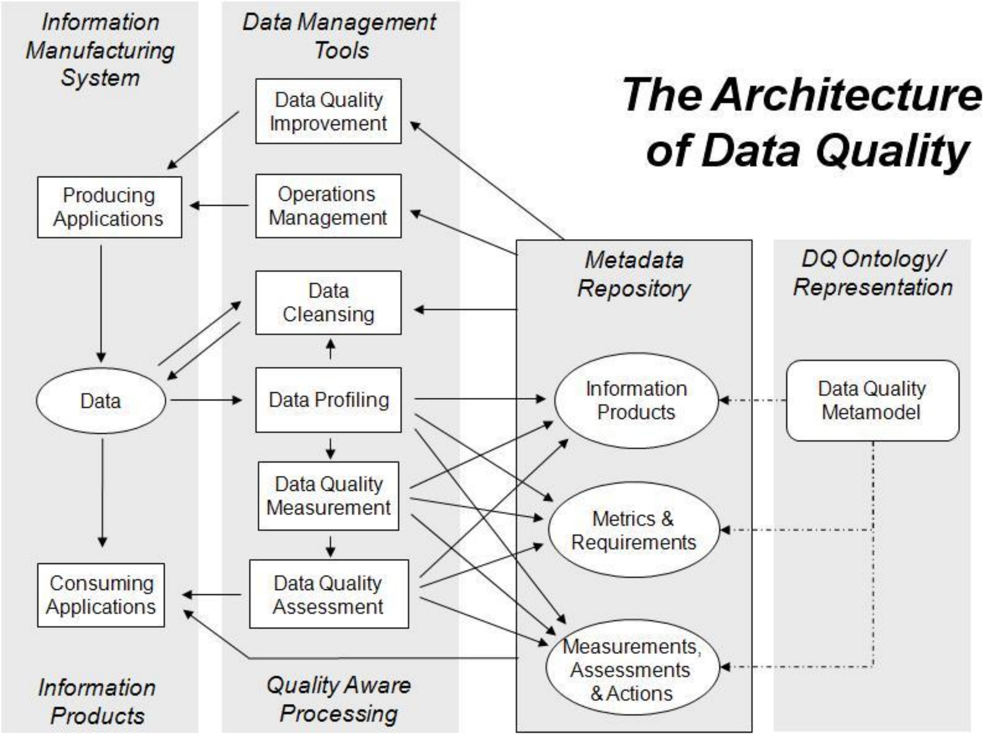


Figure 1 - The Architecture of Data Quality

Usually the management of data quality is accomplished through various data management activities (the bottom four data management tool functions depicted in Figure 1). These are: 1) data profiling to characterize the data and identify where it violates established business rules, 2) DQ measurement to calculate the overall level of the different DQ dimensions for the data based on the business rule violations from profiling, 3) DQ assessment which compares the measured DQ levels to different users' requirements, and 4) data cleansing driven by the previous 3 activities to facilitate loading data into the consuming application. These 4 activities generate large amounts of DQ metadata.

Frequently the DQ metadata generation source is tightly coupled to its immediate use. This is the case in large commercial-off-the-shelf (COTS) suites of data management tools that perform data profiling, measurement, assessment and cleansing used to load data into large enterprise resource planning (ERP) systems or data warehouses and data marts. This is also the case for locally generated error reports and log files which identify business rule violations from business rules embedded in legacy system code. Unfortunately this tight coupling of source and use, while providing immediate convenience and efficiencies to the involved areas, does not lend itself to reuse: e.g., driving adjustments to data operations management that can correct DQ problems, or longer term data quality improvement activities that could prevent DQ problems in the first place. Tightly coupled DQ information is also difficult to obtain or unavailable to users of consuming applications that support transaction processing and decision making. Nor is it typically made available to external B2B parties that are dependent on the data.

To address this situation, the DQ metadata must be decoupled from the source and exposed. In this way the information can be made available for any consumer to access. And furthermore, to ensure maximum availability, the exposure of the metadata must span variable time periods during which the different potential consumers might need access. These availability objectives are most readily accomplished by storing the DQ metadata in a persistent data store. A database for storing metadata is typically called a metadata repository (MDR). A metadata repository is simply a standard database that contains metadata. So, a database for storing DQ metadata is called a DQ metadata repository (the third column in Figure 1). All of the DQ metadata will be stored in the MDR.

The Basic Data Quality Metamodel

The data model for the MDR must be flexible and generic enough to accommodate the large variety of components involved in the generation, processing, presentation and use of the DQ metadata. The structure of the metadata in the MDR is defined by a conceptual model of the metadata [2], i.e., a metamodel (see Figure 2).

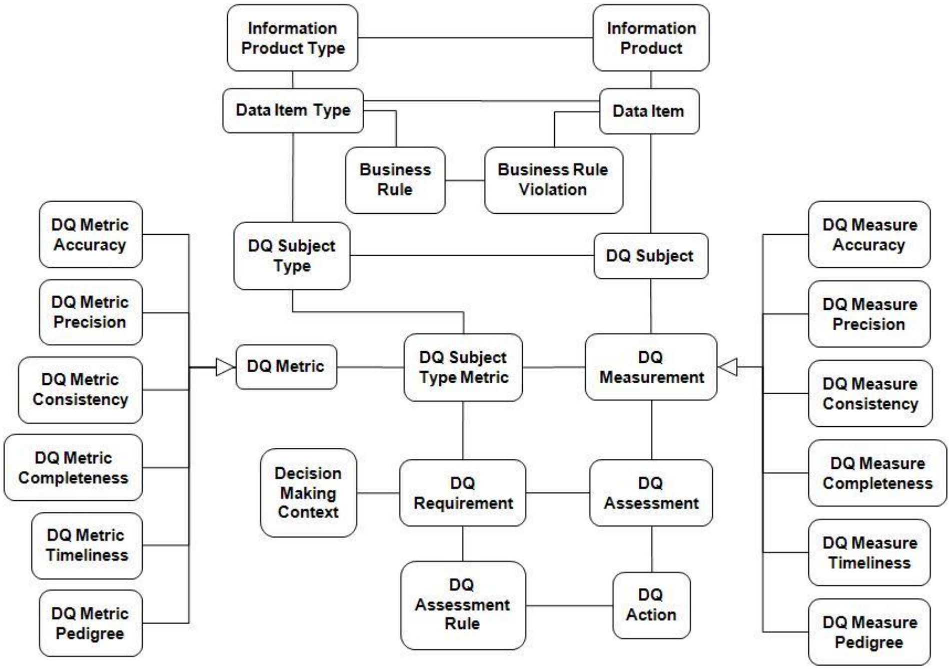


Figure 2 – The Basic DQ Metamodel

The data model represents the flow of the architecture as follows:

- Profiling – An instance of an IP of a specific type will be broken down into its component data items. Business rules applicable to a given type of data item will be applied to identify any business rule violations. The data items and business rule violations will be aggregated into instances of DQ subjects. A DQ subject is any collection of data items that constitutes a data topic of interest to a consumer, and for which quality measurements and assessments will be made.
- Measurement – Individual DQ metrics representing the different DQ dimensions of accuracy, precision, consistency, completeness, timeliness and pedigree will be defined for each particular type of DQ subject. The DQ metrics will be formulas for the percentages of the business rule violations profiled for a given DQ subject. Each instance of a DQ subject will have a DQ measurement calculated for each DQ metric.
- Assessment – Each individual usage context (user or consuming application) has its own DQ requirements or thresholds for acceptable levels of quality for each defined DQ subject's metrics. A DQ assessment is generated by comparing a DQ requirement against a DQ measurement. Various actions are then taken as directed by the threshold ranges in the DQ assessment rules.

For example, USAF aircraft maintenance data is processed through a system called REMIS (Reliability & Maintainability Information System). Information products in the form of flat files containing status updates from a maintenance system in the field called G081 are sent to REMIS on a daily basis. One of the data items from G081 is the status transaction record. As the data is loaded into REMIS, profiling DQ checks are performed against specific data items (e.g., a business rule states that the status record must contain a valid aircraft designator). The total number of business rule successes or failures and the total number of instances checked are part of the formula for calculating a specific DQ metric for a dimension (in this case, for the accuracy dimension, with a percentage as the unit of measure, the metric is the number of records containing valid aircraft designators divided by the total number of records). Different accuracy calculations for different business rules can be combined to produce an aggregate accuracy measurement. Assessment rules are specified for ranges of the measurement levels (say, between 1 and 0.84), to which action rules designate a specific response (say, send an email, post an alert, or generate a report).

Data Quality Metadata Exchange

As indicated in Figure 1 a large number of lines representing DQ metadata exchanges move between the MDR and the other architectural components. Each of the Data Management Tool components will either be generating or consuming DQ metadata as they go about their tasks. Information Manufacturing System components can also utilize DQ metadata. Thus, there are many situations during the generation, processing and presentation of DQ metadata where exchange becomes an important enabling activity. Because there are so many of these exchange situations, because there are so many different parties to these exchanges, and because there are so many different types of data for which DQ metadata can be collected and used, but because the structure of the DQ metadata itself can be well defined, it becomes desirable to establish a formal approach for the exchange of DQ metadata. Such an exchange approach would be employed when there are a number of different parties involved in a frequent and repetitive exchange of information. In this way the exchange can be optimized so that the maximum number of parties can participate in the exchange with a minimum amount of effort.

Exchanges of data between systems are best accomplished through message protocols with the data contained inside or attached to the messages. Most modern exchange approaches are based on the HTTP protocols and the Extensible Markup Language (XML). So, these would be the natural standards to use for the definition of the protocols, format and syntax of the messages involved in the exchange of DQ metadata. The content in the DQ messages must be consistent with the structured representation of the DQ metadata in the DQ metadata repository

Data Quality Fundamentals

A finding of the MITRE MOIE was that that a number of key fundamentals had to be addressed for any data quality management solution to be considered truly flexible and generic. For example:

1. There are many different dimensions to DQ, and the solution must accommodate as many as possible, or at least be configurable to accept new dimensions.
2. The way that data is packaged for efficient operational flow is usually very different from how it is ultimately viewed and used by its consumers (and how it is structured to answer questions about its quality).
3. At a basic level all DQ measurements are driven by business rule violations. The tools for identifying business rule violations come in a dazzling array of shapes and sizes, from small validation scripts, to data error detection code embedded in legacy systems, to standard DBMS quality checking capabilities, to large COTS DQ packages, to manual validation checks.
4. DQ measurement is different than DQ assessment. DQ measurement is intrinsic to the data, while DQ assessment is intrinsic to the user and their requirements. DQ measurement is objective while DQ assessment is subjective.
5. The same data can be used by a number of quite different users, all of whom can have dramatically different DQ requirements. What is good enough DQ for one person might not be adequate for another person.

The DQ metamodel addresses these fundamentals and enables different organizations to capture and exchange all relevant information in a very efficient way. This model represents the semantics of DQ management, i.e., the core set of entities and relationships that should be constructed and populated to properly support the fundamentals of DQ processing.

The different entities in the metamodel can be organized into six major areas: 1) IPs and DQ subjects, 2) DQ metrics, 3) DQ measurements, 4) DQ requirements, 5) DQ assessments, and 6) DQ actions. All of these areas of DQ metadata must be represented in the DQME approach.

Another key idea implemented as part of this metamodel is the separation of the entities into two sets. The first set represents abstract concepts called types (the left side of Figure 2). These are generally definitional in nature and would be specified during set up or configuration of the system for a specific group interested in managing the quality of their data. The second set of entities represents actual instances of the types (the right side of Figure 2). These are operational in nature and will be derived from their corresponding types. The operational tables will be populated and used as data is processed through the information manufacturing system during production.

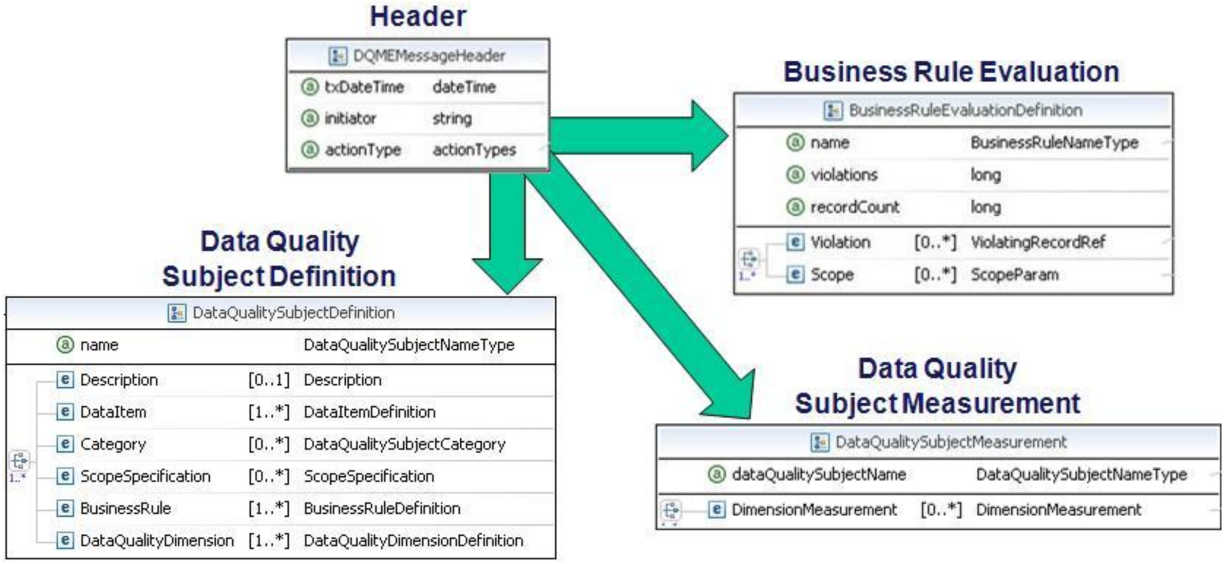
All of these entities and ideas helped to direct the development of and are directly incorporated into the DQME approach.

RELATED WORK

Some efforts have been undertaken to provide metadata interchange models that can accommodate DQ metadata. Because data quality is a type of metadata, a primary criterion for evaluation of these efforts is whether they have addressed the task of defining a separate structure for DQ metadata, and how well they model the fundamentals discussed above needed to achieve goals of generality, flexibility and ease of use. Typically the DQ component is too specific, or too general, and/or is buried in the larger metadata exchange specification. For example, Eurostat has developed the Statistical Data and Metadata Exchange (SDMX) initiative [4] which can support the publishing of metadata regarding DQ. But the DQ metadata must conform to their definition of DQ, and the approach does not address the fundamentals. Other approaches are flexible and generic but only support part of the fundamentals and the full exchange solution. For example, IP-XML [8] focuses primarily on the IP data flow components, and does not support DQ subjects, business rules, metrics, requirements and assessments; nor the operational vs. definitional aspects of the solution requirement. Maydanchik's data quality assessment meta data model [5] provides for excellent representation of errors, rules and scores, but does not provide for exchange.

DQME MESSAGES

DQME-based exchanges will be composed of a series of messages. There are three basic types of messages, all having a common header as depicted in Figure 3:


Header
Business Rule Evaluation
Data Quality Subject Definition
Data Quality Subject Measurement

DQMEMessageHeader	
txDateTime	dateTime
initiator	string
actionType	actionTypes

BusinessRuleEvaluationDefinition		
name		BusinessRuleNameType
violations		long
recordCount		long
Violation	[0..*]	ViolatingRecordRef
Scope	[0..*]	ScopeParam

DataQualitySubjectDefinition		
name		DataQualitySubjectNameType
Description	[0..1]	Description
DataItem	[1..*]	DataItemDefinition
Category	[0..*]	DataQualitySubjectCategory
ScopeSpecification	[0..*]	ScopeSpecification
BusinessRule	[1..*]	BusinessRuleDefinition
DataQualityDimension	[1..*]	DataQualityDimensionDefinition

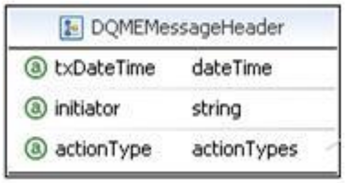
DataQualitySubjectMeasurement		
dataQualitySubjectName		DataQualitySubjectNameType
DimensionMeasurement	[0..*]	DimensionMeasurement

Figure 3 – Types of DQ Messages

The individual message types are derived from the message header so as to contain the specific detail type information as depicted in Figure 3 required for the appropriate exchange situation. One of the message types focuses on the definitional aspects of the DQ metadata exchange, while the other two message types focus on the operational aspects.

DQME Message Header

Each of the messages will start out with a common header **DQMEMessageHeader** (Figure 4). Each of the message types is an extension of this base. The message header provides basic tracking attributes. There is a transaction date and time which provides the effective date and time associated with the data. The initiator attribute designates the person or system that initiated the transaction. An action type attribute supplies an enumerated value (either create, update or delete) used to represent how the data may be applied to the DQ MDR.



DQMEMessageHeader	
txDateTime	dateTime
initiator	string
actionType	actionTypes

Figure 4 - DQME Message Header

Data Quality Subject Definition

A data quality subject is any set of data for which quality measurements and assessments will be made. DQ subjects are different than IPs. This is because collections of data in whose quality consumers have an interest are usually different than the way data is packaged for purposes of efficient operations. DQ subjects will typically be built up from the IPs' component data items. So, the IPs need to be decomposed into their component data items, and the data items will then be used to build up the data objects.

The definition of a data quality subject is specified in a **DataQualitySubjectDefinitionMessage**. This message is defined as an XML schema definition (XSD) complex type called **DataQualitySubjectDefinition** (Figure 5). It provides the name of the data quality subject, and contains the data items, business rules, and dimensions that compose it. It also provides any categories to which the DQ subject belongs, and any scopes that qualify or subset the DQ subject.

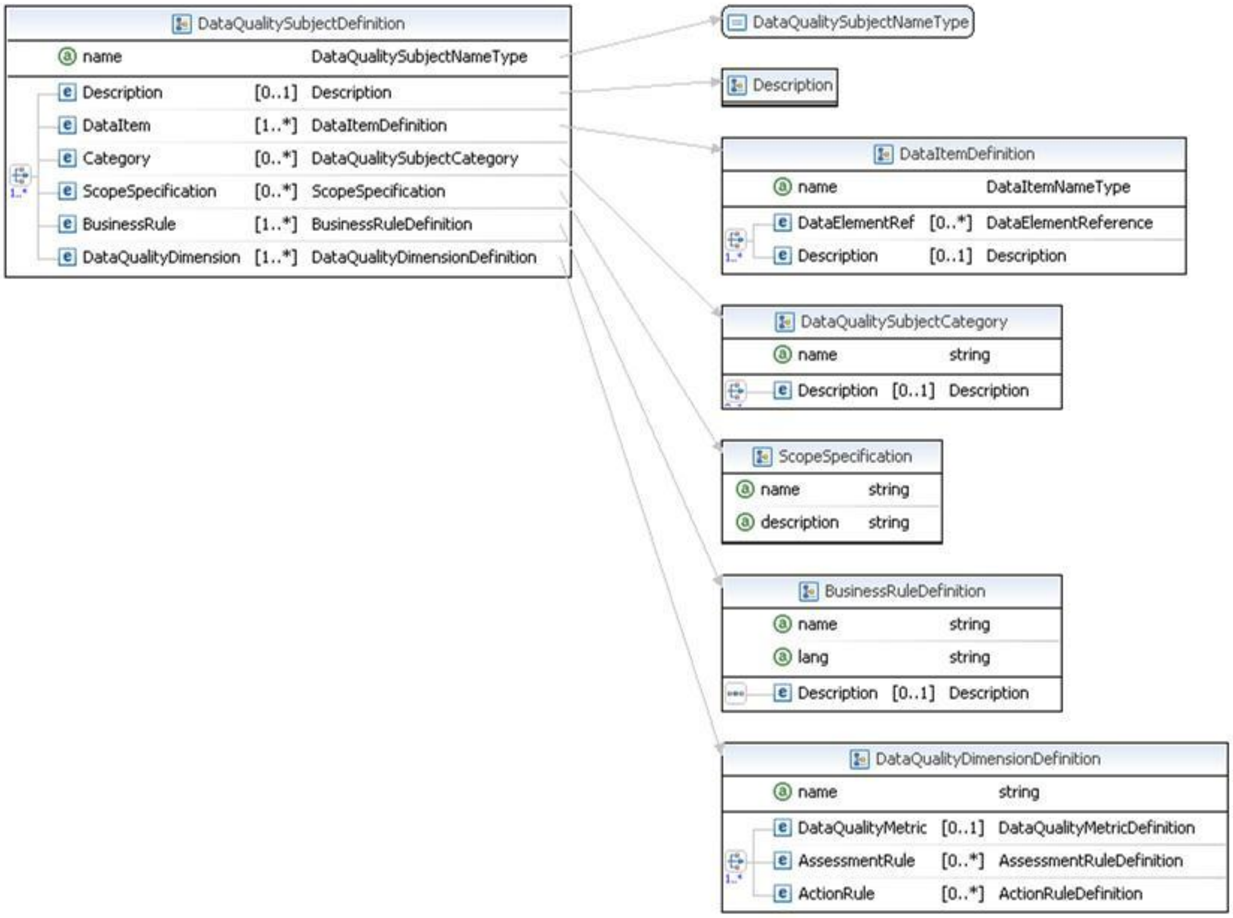

DataQualitySubjectDefinition
name DataQualitySubjectNameType
Description [0..1] Description
DataItem [1..*] DataItemDefinition
Category [0..*] DataQualitySubjectCategory
ScopeSpecification [0..*] ScopeSpecification
BusinessRule [1..*] BusinessRuleDefinition
DataQualityDimension [1..*] DataQualityDimensionDefinition
1..*
DataQualitySubjectNameType
Description
DataItemDefinition
name DataItemNameType
DataElementRef [0..*] DataElementReference
Description [0..1] Description
1..*
DataQualitySubjectCategory
name string
Description [0..1] Description
ScopeSpecification
name string
description string
BusinessRuleDefinition
name string
lang string
Description [0..1] Description
DataQualityDimensionDefinition
name string
DataQualityMetric [0..1] DataQualityMetricDefinition
AssessmentRule [0..*] AssessmentRuleDefinition
ActionRule [0..*] ActionRuleDefinition
1..*

Figure 5 - Data Quality Subject Definition

The DQ subject definition associates the following elements with the DQ subject. These elements can be presented in any order.

- Data Item – A data item specifies the name of the data item, and (optionally) contains references to the source data elements that are used to compose it. An element reference can identify the source system, database, table, and column that contain it. In this way the data quality subject can be tied back to its source.
- Category – A category is a descriptive grouping for data quality subjects. This is valuable for

situations where there are multiple data quality subjects that must be grouped together to cover a particular topical area. A data quality subject can also specify (and belong to) multiple categories in order to address situations where the topical areas overlap.

- Scope – A scope is a subset of a data quality subject. Scopes provide a way to qualify the specific instance measurements of a data quality subject and provide the basis for drill down. Multiple scopes (and thus multiples ways to subset or filter the data) can be defined for a given data quality subject.
- Business Rule – A business rule is a Boolean (true/false or passed/failed) expression describing the expected values of the outcome of a business rule for a data quality subject. This definition provides basic information upon which data quality metrics are evaluated. The business rule definition specifies its name (used in data quality metric definition expressions) and language (natural like English or artificial like SQL). The text it contains is the actual business rule expression or description of the business rule.
- Dimension – A DQ dimension is a named area of data quality that is of interest to some consumer of the DQ metadata. The six dimensions the AF has chosen to focus on at this time are: accuracy, precision, completeness, timeliness, consistency, and lineage (others are can be defined as needed).

The structure of a Data Quality dimension definition is depicted in Figure 6. The data quality dimension definition will be composed of a name, a data quality metric, and a series of assessment rules and action rules.

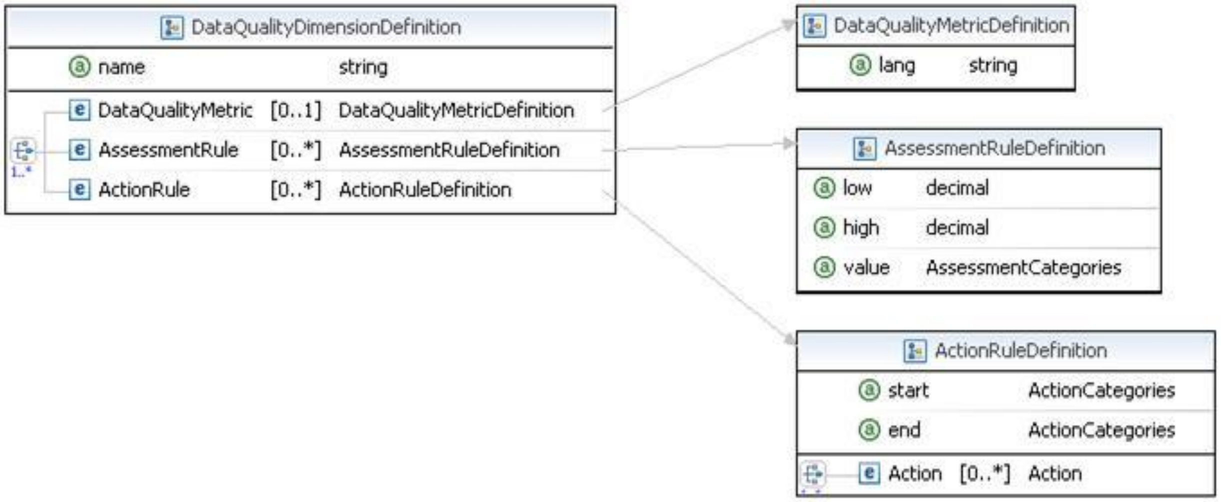

DataQualityDimensionDefinition
name string
DataQualityMetric [0..1] DataQualityMetricDefinition
AssessmentRule [0..*] AssessmentRuleDefinition
ActionRule [0..*] ActionRuleDefinition
1..*
DataQualityMetricDefinition
lang string
AssessmentRuleDefinition
low decimal
high decimal
value AssessmentCategories
ActionRuleDefinition
start ActionCategories
end ActionCategories
Action [0..*] Action

Figure 6 - Data Quality Dimension Definition

- Data Quality Metric – A data quality metric defines specifically how the value of the dimension will be calculated. It specifies the name and language of the metric. The data quality metric definition language or "dqmdl" that we have developed to express the computation is described below. Other languages could also be implemented. The content of the element contains the dqmdl expression that will be evaluated to compute the value of the metric.
- Assessment Rule – An assessment rule defines how to assess the value of the measurements for the dimension. It represents the data quality requirements. It specifies the range of values (high and low ranges) for a data quality measurement that will be assigned to a specific assessment category. The possible assessment categories are “normal”, “warning” and “critical”.
- Action Rule – An action rule specifies the actions to take based on the assessed value of the dimension. It is triggered upon the transition of the assessed value of a data quality measurement

from an initial (start) state to a final (end) state. These states can be either “normal”, “warning”, “critical”, “any”, or “undefined”. The default for both start and end states is “any”. The action rule will also contain a series of action elements. Multiple actions can be specified to occur when an action rule is triggered.

An action element (see Figure 7) specifies in an attribute string the type of action to be taken. The names and values of any number of parameters required for the action can be specified as separate elements within the action. These actions could be anything that might be previously arranged for processing by the system receiving the DQ metadata, and could include actions such as notifications by email, raising of automated alerts and alarms, or generation of reports.

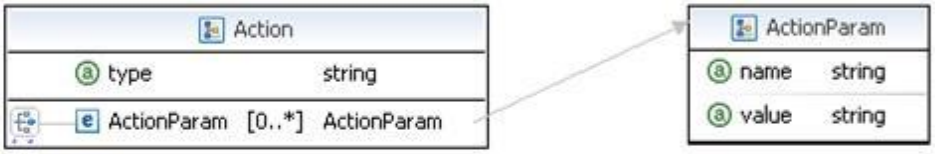


Figure 7 - Actions

Example:

The following example of a data quality subject definition message is drawn from the aircraft maintenance system introduced earlier whose data will be processed through a series of business rules, violations of which will be used to construct the DQ measurements and assessments. In it we present examples of all of the different elements described above.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<DataQualitySubjectDefinitionMessage actionType="create" initiator="USAF Maintenance Data Panel"
           txDateTime="2009-01-15T14:12:15.381-05:00"
           xsi:schemaLocation="http://www.mitre.org/dqme dqme.xsd"
           xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
           xmlns="http://www.mitre.org/dqme">
      <DataQualitySubjectDefinition name="G081_TO_REMIS_AIRCRAFT_STATUS_COMPARISON">
           <Description>Compares G081 Aircraft Status data sent To REMIS and what REMIS has
                      successfully processed and stored</Description>
           <DataItem name="G081_Status_Transactions">
                 <Description>Record structure for G0998018</Description>
                 <DataElementRef column="SERIAL_NUMBER" table="v_afks_remis_8018"
                            database="cams_fm_ar" datasource="G081"/>
                 <DataElementRef column="EQUIPMENT_DESIGNATOR" table="v_afks_remis_8018"
                            database="cams_fm_ar" datasource="G081"/>
                 . . .
           </DataItem>
           <DataItem name="REMIS_Asset_type">
                 <Description>Required fields to validate against REMIS for an
                            aircraft</Description>
                 <DataElementRef column="equip_desig_ID" table="asset_type" database="remis_ev"
                            datasource="REMIS"/>
                 <DataElementRef column="type_equip_ID" table="asset_type" database="remis_ev"
                            datasource="REMIS"/>
                 . . .
           </DataItem>
           . . .
           <BusinessRule lang="english" name="Valid_Aircrafts">Validate G081 Equipment Designator
                      exists in REMIS</BusinessRule>
           <BusinessRule lang="english" name="Valid_Tail_Numbers">Validate G081 Serial Number
                      exists in REMIS</BusinessRule>
           . . .
           <DataQualityDimension name="precision">
                 <DataQualityMetric lang="dqmdl">1-(Compare_Sum_Status_Hours.failed /
                            Status_Records.total)</DataQualityMetric>
                 <AssessmentRule value="normal" high="1" low="0.97"/>
                 <AssessmentRule value="warning" high="0.96" low="0.93"/>
                 <AssessmentRule value="critical" high="0.92" low="0"/>
                 <ActionRule start="any" end="critical">
                      <Action type="email">
                            <ActionParam name="email_address" value="joe.user@wpafb.af.mil" />
                            <ActionParam name="subject"     value="G081 TO REMIS AIRCRAFT STATUS
                                       COMPARISON is critical" />
                            <ActionParam name="body">
                                  <![CDATA[The precision dimension for Data Quality Subject,"G081
                                       TO REMIS AIRCRAFT STATUS COMPARISON", is outside of
                                       acceptable bounds.]]>
                            </ActionParam>
                      </Action>
                 </ActionRule>
           </DataQualityDimension>
           <DataQualityDimension name="timeliness">
                 <DataQualityMetric lang="dqmdl">1-(Late_Reported_Status.failed /
                            Status_Records.total)</DataQualityMetric>
                 <AssessmentRule value="normal" high="1" low="0.8"/>
                 <AssessmentRule value="warning" high="0.79" low="0.63"/>
                 <AssessmentRule value="critical" high="0.62" low="0"/>
                 <ActionRule start="normal" end="warning">
                      <Action type="alert">
                            <ActionParam name="title" value="G081 TO REMIS AIRCRAFT STATUS
                                       COMPARISON has changed" />
                            <ActionParam name="message"     value="The timeliness dimension has
                                       changed from Normal to Warning." />
                            <ActionParam name="urgent" value="Y" />
                      </Action>
                 </ActionRule>
           </DataQualityDimension>
           <DataQualityDimension name="accuracy">
                 <DataQualityMetric lang="dqmdl">1 - ((Valid_Aircrafts.failed *1.05 +
                            Valid_Tail_Numbers.failed *1.02  + Valid_Commands.failed +
                            Valid_Work_Unit_Codes.failed + Valid_Purpose_Codes.failed +
                            Valid_Purpose_Code_Commands.failed + Valid_Organizations.failed +
                            Valid_Work_Centers.failed + Valid_Status_Start_Stop_Dates.failed)/
                            Status_Records.total)</DataQualityMetric>
                 <AssessmentRule value="normal" high="1" low="0.84"/>
                 <AssessmentRule value="warning" high="0.83" low="0.71"/>
                 <AssessmentRule value="critical" high="0.70" low="0"/>
                 <ActionRule start="warning" end="critical">
                      <Action type="report">
                            <ActionParam name="report_name" value="G081/REMIS Status Detail" />
                            <ActionParam name="report_type" value="Cognos ReportNet" />
                            <ActionParam name="environment" value="production" />
                            <ActionParam name="schedule_time" value="now()" />
                      </Action>
                 </ActionRule>
           </DataQualityDimension>
           <ScopeSpecification name ="POSSESSING_COMMAND"/>
           <ScopeSpecification name ="POSSESSING_ORGAN"/>
           <ScopeSpecification name ="EQUIPMENT_DESIGNATOR"/>
           <Category name="Maintenance"/>
      </DataQualitySubjectDefinition>
</DataQualitySubjectDefinitionMessage>
```

Data Quality Metric Definition Language (DQMDL):

Utilized in the “DataQualityMetric” elements above is a very simple expression language we have called DQMDL. It supports four operands and four operators. The current set of operators is: “+”, “-”, “*”, and “/”. It also supports parentheses for grouping: “(“ and “)”. Standard operator precedence applies: ”(“, “)”, then “*”, “/”, then “+”, “-”.

The four types of operands are:

- <business rule name>.passed

- <business rule name>.failed
- <business rule name>.total
- any floating point constant

Business rule names must match a business rule defined in the DQ subject definition, and only consist of the following characters: “a-zA-Z0-9_”

Business Rule Evaluation Set

A business rule evaluation set is a collection of results from the processing of a set of business rules against a specific set of data. The information is captured and represented in a manner consistent with the Data Quality Subject Definition presented above, meaning that DQ subject names and business rule names must match for the evaluations to be properly tied back to their respective definitions.

The definition of a business rule evaluation set (see Figure 8) is specified in a **BusinessRuleEvaluationSetMessage**. This message is defined as an XML schema definition (XSD) complex type called **BusinessRuleEvaluationSet**. It provides in an attribute the name of the data quality subject to which it applies, and includes a set of the business rule evaluations and optionally detailed information about the violating records.

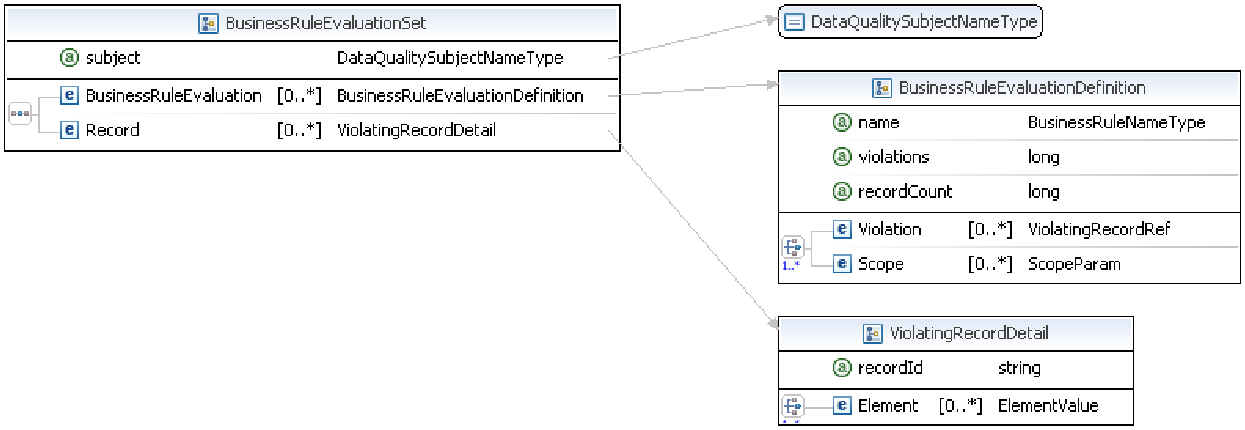

BusinessRuleEvaluationSet
subject
DataQualitySubjectNameType
BusinessRuleEvaluation [0..*] BusinessRuleEvaluationDefinition
Record [0..*] ViolatingRecordDetail
DataQualitySubjectNameType
BusinessRuleEvaluationDefinition
name
BusinessRuleNameType
violations
long
recordCount
long
Violation [0..*] ViolatingRecordRef
Scope [0..*] ScopeParam
1..*
ViolatingRecordDetail
recordId
string
Element [0..*] ElementValue

Figure 8 - Business Rule Evaluation Set

The business rule evaluation set associates the following evaluation elements with a given DQ subject. These elements must be presented in the specified sequence with all of the evaluations of the business rules optionally followed by the details any of violating record that have been provided for drill down.

Business Rule Evaluation – A business rule evaluation provides the details of the business rule being evaluated. It includes as attributes the name of the business rule (which must match the name in the DQ subject definition), the number of records violating the business rule, and the total number of records against which the business rule was evaluated. The business rule evaluation also provides references to any violating records that have been provided, and any scopes and detailed scope values for which subsetting of the evaluations may be desired.

Violating Record Detail – The violating record detail provides the detailed record information of a violation that may be desired for drill down purposes. These violating records may be the full set of violations, or simply a sampling of the violating records. Each record has an identifier that should match up with a violation in a business rule evaluation. A record can match up with more than one violation. The record detail will consist of an element for each relevant field, along with the actual value present in the field.

Example:

The following example of a business rule evaluation set message is also drawn from the same aircraft maintenance system that was used in the example above.

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<BusinessRuleEvaluationSetMessage actionType="create" initiator="WPAFB Data Services Informatica
      Profiler Adapter" txDateTime="2009-01-06T00:46:44.000Z"
      xsi:schemaLocation="http://www.mitre.org/dqme dqme.xsd"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.mitre.org/dqme">
      <BusinessRuleEvaluationSet subject="G081_TO_REMIS_AIRCRAFT_STATUS_COMPARISON">
            <BusinessRuleEvaluation recordCount="3" violations="0" name="Valid_Commands">
                  <Scope value=" WC130J" name="EQUIPMENT_DESIGNATOR"/>
                  <Scope value="MTC" name="POSSESSING_COMMAND"/>
                  <Scope value="0000WRACE" name="POSSESSING_ORGAN"/>
            </BusinessRuleEvaluation>
            <BusinessRuleEvaluation recordCount="483" violations="0" name="Valid_Commands">
                  <Scope value=" WC130J" name="EQUIPMENT_DESIGNATOR"/>
                  <Scope value="AFR" name="POSSESSING_COMMAND"/>
                  <Scope value="0053WERSQ" name="POSSESSING_ORGAN"/>
            </BusinessRuleEvaluation>
            . . .
            <BusinessRuleEvaluation recordCount="22" violations="22" name="Valid_Organizations">
                  <Scope value="  C130J" name="EQUIPMENT_DESIGNATOR"/>
                  <Scope value="MTC" name="POSSESSING_COMMAND"/>
                  <Scope value="0000WRALC" name="POSSESSING_ORGAN"/>
                  <Violation recordId="rec44242"/>
                  <Violation recordId="rec44322"/>
                  <Violation recordId="rec44889"/>
                  <Violation recordId="rec45518"/>
                  <Violation recordId="rec45545"/>
            </BusinessRuleEvaluation>
            . . .
            <Record recordId="rec34014">
                  <Element value="9510000228" name="EDA_SEQUENCE_NUM"/>
                  <Element value="9980" name="DATA_PRO_CENTER_NUM"/>
                  <Element value="25" name="PROG_LINE_NUM"/>
                  <Element value="1" name="GANG_NUM"/>
                  <Element name="REMOTE_ID"/>
                  <Element name="CMD_CODE"/>
                  . . .
            </Record>
      </BusinessRuleEvaluationSet>
</BusinessRuleEvaluationSetMessage>
```

Data Quality Subject Measurement

A data quality subject measurement contains new measured values for the dimensions of a data quality subject. These are evaluated metrics that have presumably been calculated based on a set of low level business rule violations, in which case the DQ subject measurement is an output of that evaluation activity, and consequently an input to DQ assessment and action processing activities. If business rule evaluations have been provided, and the business rules for a data quality subject have been fully defined, then a separate data quality subject measurement message may not be needed.

However, the DQ subject measurement may also be the output of some other mechanism for establishing the quality levels of a given dimension. In this case, the data quality subject measurement message will be an alternative to the business rule evaluation set message. If this is the case, then there will be no drill down opportunity to see specific business rules and violating records.

The definition of a data quality subject measurement (see Figure 9) is specified in a

DataQualitySubjectMeasurementMessage. This message is defined as an XML schema definition (XSD) complex type called **DataQualitySubjectMeasurement**. It provides in an attribute the name of the data quality subject to which it applies, and includes a set of dimension measurements.

Dimension Measurement – A dimension measurement provides the calculated values or measurements of a data quality dimension. It must include in its attributes the name of the dimension, and the value of the DQ measurement. It can also optionally provide any scopes and detailed scope values for which subsetting of the evaluations may be desired.

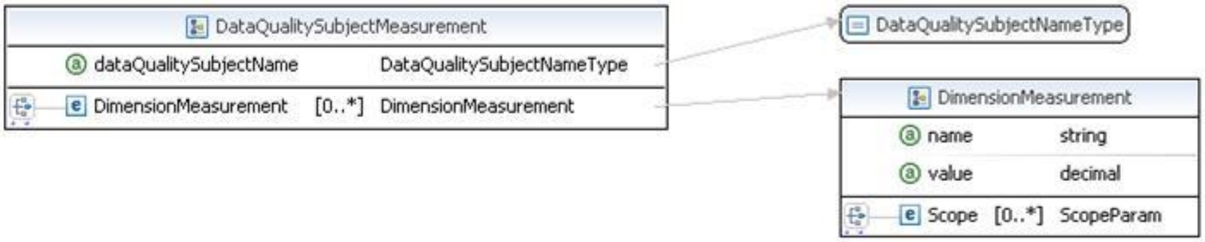


Figure 9 - Data Quality Subject Measurement

THE DQME PROCESSES

There are three process flows that will involve the exchange of DQ metadata. Two are typical process flows one for definition and one for evaluation (Figure 10), and a third atypical process flow for measurements only.

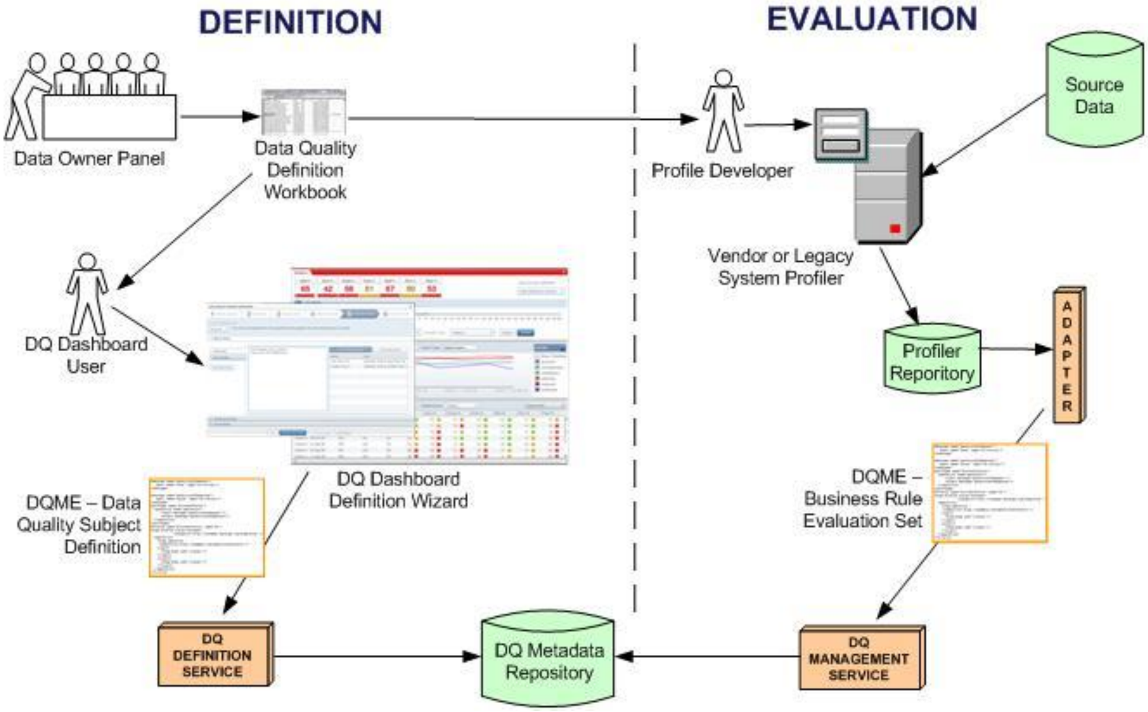


Figure 10 - Typical EDQMS Process Flows

The Definition Process Flow – The Definition Process Flow starts with an entity we call a Data Owner Panel who will be responsible for the key DQ definition activities: defining IPs and DQ subjects, business rules, DQ dimensions and metrics, usage contexts and requirements, and actions. This work can be performed off-line and captured in a DQ definition workbook (Excel spreadsheet). The DQ definition metadata must then be converted into DQME (a Data Quality Subject Definition Message) so that it can be imported and stored in the DQ MDR. For our project, we have developed an online dashboard wizard user interface to facilitate this activity. It could alternatively be accomplished by an adapter that converts the DQ Definition Workbook directly into DQME.

The Evaluation Process Flow – The Evaluation Process Flow starts with an entity we call a Profiler Developer who will be responsible for configuring a specific vendor or legacy system profiler tool to capture its business rule violations using the same DQ definition naming and structure as developed in the Definition process flow. The DQ definition workbook can come in handy for this purpose. Once configured, the profiler tool can then evaluate the business rules against the source data to identify violations which it stores in its custom or proprietary repository. The business rule violation metadata must then be converted to DQME (a Business Rule Evaluation Set Message) so that it can be imported and stored in the DQ MDR. For our project, we used an Informatica profiling tool, and developed an adapter that extracts the information from the Informatica internal repository (an Oracle database), and converts it directly into DQME. But we expect simple DQME adapters can easily be constructed for any profiling tool, or possibly an interface for manually performed measurements such as physical audits.

The Measurement Process Flow – An atypical process flow would be one that employs the Data Quality Subject Measurement DQME Message. In this process, a data organization has already generated its DQ measurements, and simply wants to obtain or provide them to another organization for processing or incorporation into their activities and tools. The providing organization would be responsible for generating the Data Quality Subject Measurement DQME Message, and the receiving organization would be responsible for consuming it using whatever adapters and web services were appropriate. A further assumption is that the measurement exchange would have been preceded by an exchange of a Data Quality Subject Definition Message.

The DQ MDR in our EDQMS is surrounded by a set of web services [1] that manage the inputs and updates to, and extracts from the database. EDQMS also provides a DQ Dashboard that can display the DQ metadata from the MDR in two views:

- Leadership View: that gives the oversight function a high level perspective of current quality levels for categories and DQ subjects with drill down available as desired
- Analysis View: that gives data stewards extended capabilities to apply different filters, examine trend analysis, and conduct drill downs to explore causes in more detail.

Furthermore, EDQMS provides the ability to raise alerts and send email notifications as specific actions to be taken on specified DQ assessments.

DEVELOPMENT & EXTENSIONS OF DQME

DQME was developed over a short 2 month period [3], and has been updated only cosmetically since. The development was performed using the open source Eclipse platform from the Eclipse Foundation. Many of the Figures presented above are drawn directly from interactions with the XML Schema Definition (XSD) design capabilities that are part of the Eclipse suite of tools. The Eclipse tool has proven to be very productive and reliable in our DQME design activities. The DQME XSD uses the following for its XML namespace: "`https://www.mitre.org/dqme`".

There are a number of possible refinements or extensions to DQME under consideration:

- Decision Making Context – clearly identify and represent "decision making contexts" which would be directly associated with assessment rules and action rules

- DQ Impact Evaluation Message - inclusion of a message which would provide for extra type of business rules to calculate the cost and mission impact of a given assessed DQ level
- Assessment Rule Extension – a richer capability for assessment rule specification beyond the current simple 3-level threshold model, including finer grained or continuous assessment categories, weighted assessments for aggregations, and probabilistic or fuzzy assessments.
- Extensions to DQMDL - future operators could include other Excel-type equation operators: e.g., “ceil”, “floor”, “round”, “exponent”, etc.

CONCLUSION & FUTURE WORK

DQME is proving to be a valuable mechanism for optimizing the exchange of DQ metadata, while ensuring that the maximum number of parties can participate in the exchange. The different messages in DQME address all of the basic metadata exchanges that are required to support a flexible and generic approach to managing data quality across the data lifecycle. We are actively engaged in utilizing it to the fullest extent possible for our AF sponsor’s Operations Support domain, particularly in support of several large ERP acquisition efforts that involve a huge amount of data migration and cleansing. This will involve a multitude of data sources and legacy systems.

We believe DQME also has significant merit as a potential formal standard. We have driven down to some key fundamentals regarding data quality that must be addressed. We have developed workable approaches to represent these fundamentals and to allow different organizations to capture and exchange the relevant metadata in a very efficient and effective manner. While there is room for extension and improvement, we believe DQME in its current form could easily serve as a basis for nomination to a formal standards body such as Object Management Group (OMG) as an industry standard.

REFERENCES

[1] Becker, D., “Information Quality & Service Oriented Architecture”, *Proceedings of the MIT 2007 Information Quality Industry Symposium*, July 2007, Cambridge, MA, pp 592-622.

[2] Becker, D., McMullen, W., Hetherington-Young, K., “A Flexible and Generic Data Quality Metamodel”, *Proceedings of the 12th International Conference on Information Quality*, November 2007, Cambridge, MA, pp 50-64.

[3] Fisher, C., Lauria, E., Chengalur-Smith, S., Wang, R., *Introduction to Information Quality*, 2006, MIT Information Quality Program, Cambridge, MA.

[4] Glossioti, M., Farmakis, G., Kassis, K., Liapis, S., Nikoloutsos, E., "A reference architecture for automatic XML data & metadata exchange between public administrations: Eurostat's case study", *eGovernment Interoperability Conference*, October 2007, Paris, France.

[5] Maydanchik, A., *Data Quality – The Accuracy Dimension*, 2007, Technics Publications, LLC, Bradley Beach, NJ.

[6] Olsen, J., *Data Quality Assessment*, 2003, Morgan Kaufmann Publishers, Boston, MA.

[7] Redman, T., *Data Quality – The Field Guide*, 2001, Digital Press, Boston, MA.

[8] Shankaranarayanan, G., Cai, Y., “A Web Services Application for the Data Quality Management in the B2B Networked Environment”, Proceedings *of the 38th Hawaii International Conference on Systems Sciences, 2005*, 2005, 0-7695-2268-8/05, IEEE.

[9] Tannenbaum, A., *Metadata Solutions, Using Metamodels, Repositories, XML, and Enterprise Portals, to Generate Information on Demand*, 2002, Addison Wesley, Boston, MA.

[10] Walmsley, P., *Definitive XML Schema*, 2002, Prentice-Hall, Inc., Upper Saddle River, NJ.

[11] Wang, R. Y., Ziad, M., Lee, Y. W., *Data Quality*, 2001, Kluwer Academic Publishers, Norwell, MA.