ISO Data Quality Standards for Master Data

Chapter · December 2015

DOI: 10.1016/B978-0-12-800537-8.00011-9

2 authors, including:

John Talburt
University of Arkansas at Little Rock
164 PUBLICATIONS **700** CITATIONS

SEE PROFILE

CHAPTER

ISO Data Quality Standards for Master Data 11

BACKGROUND

In 2009, the International Organization for Standardization (ISO) approved a set of standards for data quality as it relates to the exchange of master data between organizations and systems. These are primarily defined in the ISO 8000-110, -120, -130, -140, and the ISO 22745-10, -30, and -40 standards. Although these standards were originally inspired by the business of replacement parts cataloging, the standards potentially have a much broader application. The ISO 8000 standards are high-level requirements that do not prescribe any specific syntax or semantics. On the other hand, the ISO 22745 standards are for a specific implementation of the ISO 8000 standards in extensible markup language (XML) and are aimed primarily at parts cataloging and industrial suppliers.

The Electronic Commerce Code Management Association (ECCMA) formed in 1999 has largely guided the development of the ISO 8000 standards. The American National Standards Institute (ANSI) is the U.S. representative to ISO, and ECCMA is the ANSI accredited administrator of the U.S. Technical Advisory Group (TAG) to the ISO Technical Committee (TC) 184 and its subcommittees SC4 and SC5 that deal with industrial data and data interoperability, respectively. Under the leadership of Peter Benson, ECCMA continues to be active in developing new data quality standards for master data and providing ISO 8000 training and certification. This chapter will focus primarily on the ISO 8000-110 standard because it describes the overall framework and guidelines to which specific implementations of the ISO 8000 standard such as ISO 22745 must comply (ANSI, 2009).

DATA QUALITY VERSUS INFORMATION QUALITY

Before getting too deeply into the ISO 8000 standard, it might be helpful to review the basic principles of information and data quality. Even though there is almost universal acceptance that data and information are separate concepts, the same cannot be said for the terms *data quality* and *information quality*. Many noted experts use *data quality* and *information quality* interchangeably with the general definition of "fitness for use (purpose)" (Juran, 1989).

Entity Information Life Cycle for Big Data. http://dx.doi.org/10.1016/B978-0-12-800537-8.00011-9

The discussion of the ISO 8000 standard is an occasion where it can be helpful to separate the concepts of data quality and information quality. In 2010, the International Association for Information and Data Quality (IAIDQ) undertook an extensive survey of information professionals to elicit their opinions on the knowledge and skills required to be successful as an information quality professional (Yonke, Walenta, & Talburt, 2012). The survey was part of the development process for the Information Quality Certified Professional (IQCP[sm]) credential (IAIDQ, 2014). The knowledge and skill descriptions gathered from the survey were analyzed and subsequently summarized into six categories, referred to as the IAIDQ Domains of Information Quality. The domains are

1. Information Quality Value and Business Impact
2. Information Quality Strategy and Governance
3. Information Quality Environment and Culture
4. Information Quality Measurement and Improvement
5. Sustaining Information Quality
6. Information Architecture Quality

An important principle emerging from the study was that information quality is primarily about helping organizations maximize the value of their information assets. The survey results provided strong support for the principle that information quality is a business function rather than an information technology responsibility. The first three domains that cover value, business impact, strategy, governance, environment, and culture are primarily business and management issues.

Even though successful MDM depends upon understanding many technical issues, many of which are presented in this book, its adoption is, or should be, a business-driven decision. Furthermore, MDM is generally viewed as a key component of data governance programs, which are themselves seen as essential for organizations to be competitive in an information-based economy.

Only the last three IAIDQ domains deal with measurement, improvement, monitoring, and data architecture, the activities comprising what can be generally defined as data quality. As ISO defines quality as meeting requirements, data quality can be defined as the degree to which data conform to data specifications (data requirements). Information quality includes data quality, but in the larger context it also addresses the business issue of creating value from information and how to manage information as product. Information quality requires information producers to understand the needs of information consumers. Once those are understood, they can be translated into the data specifications that underpin and define data quality.

Some have likened the difference between data and information quality to the classic engineering dilemma: Am I building the thing right? Or am I building the right thing? Building an information product the right way is about conforming to the data specifications for the product, which is the definition of data quality. Building the right thing is creating an information product that provides value to its users, the definition of information quality.

To be successful, data quality and information quality must work hand-in-hand. An organization should constantly align its data quality activities with its information quality activities. Data cleansing, assessment, and monitoring (DQ) should only be undertaken as a solution to a business need (IQ). Conversely, data quality assessments (DQ) can often uncover data problems whose solutions directly impact increased cost, lost revenue, operational risk, compliance default, and many other business issues (IQ).

RELEVANCE TO MDM

The ISO 8000 standards are relevant to MDM for several reasons. The most obvious is the standard is specifically for master data. In particular, the accuracy of ER that supports MDM has a strong dependence on understanding of reference identity attributes and the alignment of identity attributes across different sources.

But beyond that, there is general agreement that data standards are important for data quality but are hard to establish (Redman, 1996). The problem of establishing standards for data is probably best solved first through master data, because there is broad agreement about the things that require master data, at least at the industry level.

As organizations move to exploit Big Data, the need for metadata that describes data in a source becomes even more acute as data are created at an amazing rate of speed. Instead of simply automating manual processes, organizations are now creating new ways of collecting data highly dependent on instrumentation. Without information about provenance, completeness, accuracy, and other data characteristics, it is nearly impossible to understand what the data even represents.

GOALS AND SCOPE OF THE ISO 8000-110 STANDARD

Despite widespread adoption of the ISO 8000-110 standard in certain industries, it is not well understood in many information quality and MDM circles. Even though it is a generally applicable data quality standard for master data, its application has been mainly limited to replacement parts cataloging for the military and petroleum industry. For example, the North Atlantic Treaty Organization (NATO) now requires all parts suppliers to comply with the ISO 22745 version of the ISO 8000-110 standard.

UNAMBIGUOUS AND PORTABLE DATA

The ISO 8000-110 standard has two major goals. The first is to remove as much ambiguity as possible from the exchange of master data. The second is to make master data portable from system-to-system.

The goal of being unambiguous is addressed in the standard through extensive use of semantic encoding. Semantic encoding is the replacement of natural language terms and descriptions with unique identifiers that reference clear and unambiguous data dictionary entries. The standard requires the referenced data dictionary to be easily accessible by both the transmitting and receiving parties.

THE SCOPE OF ISO 8000-110

It is important to understand what the ISO 8000-110 standard covers, and also what it does not cover. At a high level, the ISO 8000-110 is a standard for representing data definitions and specifications for

- Master data in the form of characteristic data that are
- Exchanged between organizations and systems, and that
- Conform to the data specifications that can be validated by computer software.

The latter point about conformance to data requirements puts the ISO 8000-110 standard squarely in the realm of data quality. To place ISO 8000 into the vocabulary of this book, characteristics (also called properties) correspond to the identity attributes of the entities under management. Master data in the form of characteristic data are essentially the entity references described in Chapter 1.

Logically they comprise a set of attribute-value pairs where the attribute name is a characteristic or property of the entity, and the value is a specific instance of the characteristic or property for a particular entity. In a physical implementation the attribute-value pairs for an entity could be represented as a row in a spreadsheet, a record in a file, or XML document. For example, a characteristic of an electric motor might be its operating voltage with a value of 110 volts/AC. Other characteristics might be power consumption in watts or its type of mounting.

Perhaps the most common misunderstanding about the ISO 8000 standards is that they somehow establish certain levels of data quality such as 80% completeness or 95% accuracy for particular master data domains and characteristics. This is not at all the case. Instead, ISO 8000 describes a standard for embedding references to data definitions and data specifications into the master data exchanged between two organizations in such a way that the organizations can automatically validate that the referenced specifications have been met.

The automatic verification of conformance to specifications is an important aspect and innovation of the standard. Although several ISO standards address quality, such as the ISO 9000 family of standards for quality management systems, ISO 8000 specifically requires that conformance to the specifications must be verifiable by a computer (software) rather than by manual audits.

Another important point is the standard applies only to master data in transit between organizations and systems. The standard does not apply to data at rest inside of a system, nor does it provide a way to talk about an MDM system as being ISO 8000-100 compliant.

MOTIVATIONAL EXAMPLE

Consider the following example of how a master data exchange standard could be helpful. Suppose ABC Bank has developed a new financial product it wants to market to its existing customers. However, ABC also wants to target different customers in different ways depending on the customer's characteristics, i.e. through market segmentation. One important characteristic for segmenting its customers is their

level of income. ABC has a customer MDM system, but income level is not one of the identity attributes of the MDM IKB nor is it a business attribute maintained by any of the internal clients of the ABC MDM system. In order to segment its customers, ABC needs to acquire this information from third-party data brokers.

ABC approaches two data brokers, DB1 and DB2, about providing income information. ABC sends both DB1 and DB2 a file of its customers' names and addresses. Both brokers match the ABC file against their MDM systems to append income information to ABC's file. When ABC receives the appended files from DB1 and DB2, it finds that both brokers also provided a separate document describing the formats and definitions of the items in the returned data.

When the marketing team examines the returned files, they have to decode the results. For example, when they look at the record for customer John Doe living at 123 Oak St, they find that broker DB1 has appended a code value of "C" for income level. Looking this up in the documentation provided by DB1 they find that "C" corresponds to an income level between $40,000 and $60,000.

The marketing team finds broker DB2 also recognized the same customer, John Doe, in its system and appended the income code "L4". The documentation from DB2 indicates that "L4" corresponds to an income level between $75,000 and $100,000.

After the analysis of the data returned from the two brokers, the marketing team now has two problems. The first problem is that the two brokers are using different income increments for their income brackets. DB1 reports in increments of $20,000, and DB2 in increments of $25,000. The difference creates a problem of how to assign a single income level code to each ABC customer. It turns out there are many cases where DB1 had income information, but DB2 did not, and conversely, cases where DB2 had information, but DB1 did not. If in these cases they simply use the broker's codes, then the income fields contain two different sets of codes. This is a data quality problem known as an overloaded field. At the same time, because the brackets are of different sizes, it is not clear how to translate the two different sets of data broker codes into a single set of meaningful codes for ABC.

A second, more troubling, problem is the case of John Doe where both brokers reported income, but the levels are entirely different. When the marketing team investigates further, they find DB2 levels are consistently higher than those reported by DB1. In an attempt to determine which broker might have more accurate reporting, the marketing team called both brokers to better understand their data collection process. It was during these conversations that ABC uncovered yet another issue. DB1 explained they were collecting individual income, but DB2 was collecting and reporting household income. In other words, the value reported by DB1 was their estimate of the income level for just John Doe himself. The value reported by DB2 not only included John Doe's income, but also his spouse's income, and possibly other family members. Even though both were reporting "income," each broker had a different definition of what that means.

In the context of ISO 8000-110, the master data are the customers of ABC Bank. The characteristic data are name, address, and income level. DB1 and DB2 are both

using different semantics (definitions) for income level. In addition, both brokers also use different data syntax specifications, i.e. different bracket sizes and different bracket codes.

What ISO 8000-110 provides as a response to this problem might be a service level agreement (SLA) for data brokers supplying this information to ABC Bank. The ABC SLA might require that it will only buy income data from brokers willing to meet the following conditions and specifications:

1. The income level must represent "individual" income as defined by ABC.
2. The income level codes must be the letters "A", "B", "C", "D", and "E".
3. The income level brackets must be in $25,000 increments starting with $0, with the "E" level representing the bracket $100,000 and above.
4. All of the required specifications are published in an online data dictionary available to the data providers.
5. The transmitted file must include metadata encoded in a data specification language understood by both ABC and the data provider, which will allow ABC to automatically validate conformance to these specifications.

FOUR MAJOR COMPONENTS OF THE ISO 8000-110 STANDARD

The ISO 8000-110 standard has four major parts (ANSI, 2009). These are

Part 1 General Requirements
Part 2 Message Syntax Requirements
Part 3 Semantic Encoding Requirements
Part 4 Conformance to Data Specification

PART 1: GENERAL REQUIREMENTS

According to the ISO 8000-110 standard, a master data message must meet six general requirements.

Part 1.a The master data message shall unambiguously state all information necessary for the receiver to determine its meaning
Part 1.b A formal syntax must be specified using a formal notation
Part 1.c Any data specification required by the message shall be in a computer-interpretable language
Part 1.d The message must explicitly indicate both the data specifications it fulfills and the formal syntax (or syntaxes) to which it complies
Part 1.e It must be possible to check the correctness of the master data message against both its formal syntax and its data specifications
Part 1.f The references within the master data message to data dictionary entries must be in the form of unambiguous identifiers conforming to an internationally recognized scheme.

PART 2: SYNTAX OF THE MESSAGE

The requirements for master data message syntax are:

- The message shall contain in its header a reference to the formal syntax to which it complies.
- The reference shall be an unambiguous identifier for the specific version of the formal syntax used to encode the message.
- The formal syntax shall be available to all interested parties.

The first point is somewhat of a conundrum. The standard says the header of the message must point to the definition of the syntax in which the message is encoded. However, the receiver cannot really locate and understand the part of the message that comprises the header without already knowing the syntax of the message. The reason for this is likely the inspiration for the syntax standard in XML. All XML documents should start with the element

<?xml version="1.0"?>

declaring itself as an XML document.

There are some other things about message syntax worth noting. First of all, a compliant message can refer to more than one syntax. Again, this likely goes back to XML because there are many data standards defined as restrictions of XML. In other words, the first or underlying syntax can be XML. Then a second level of syntax is added by restricting the document to only use certain element names that have a predefined meaning.

Some examples where this has been done are the Global Justice XML Data Model (GJXML) for the exchange of information among law enforcement agencies and the eXtensible Business Reporting Language (XBRL) adopted by the Securities and Exchange Commission for financial reporting. Both syntaxes provide a common vocabulary through predefined XML tags (elements) and attributes. Most notably, the ISO 22745 standard, which is an ISO 8000 compliant standard, has a syntax that is a restriction of the XML syntax.

Even though XML and many of its restrictions such as XBRL are open and free, free access is not required by the standard. Nothing prevents an organization from developing its own syntax and charging a fee for its use. However, the message syntax must be formally defined in a formal notation (General Requirement Part1.b.), so it is computer readable.

Another point is that the syntax requirement does not preclude encryption of the message because encryption itself is not a syntax. The encryption of master data messages just adds a second layer of translation. Once a message has been created in the ISO 8000 compliant syntax, it can then be encrypted for transmission. The receiver must first decrypt the message, and then interpret the content according to the ISO 8000 compliant syntax.

From a practical standpoint, the formal syntax can be almost any commonly used computer-readable document or dataset format such as XML, comma

separated values (CSV) files, spreadsheets, files in fixed-length field record format, ISO 22745-40 conformant messages, and ISO 9735 (EDIFACT) conformant messages.

PART 3: SEMANTIC ENCODING

The semantic encoding requirement is just an elaboration of the General Requirement Part 1.f. (The reference within the master data message to data dictionary entries must be in the form of unambiguous identifiers conforming to an internationally recognized scheme). Because the master data message must be in the form of characteristic data, its basic format of the master data message is a collection of property value pairs.

(property1, value1), (property2, value2), …, (propertyN, valueN)

In order to meet the semantic encoding requirement, each property must be represented as an unambiguous identifier that references a data dictionary entry. This means that a typical message in the form

Message: (Name, "John Doe"), (Income, "A"), …

is not compliant. However, a message of the form

Message: (ICTIP.Property.ABC.101, "John Doe"),
(ICTIP.Property.ABC.105, "A"), …

can be compliant.

First of all, the properties are represented as valid, uniform resource identifiers (URI). This complies with the part of General Requirement Part 1.f requiring identifiers to conform to an internationally recognized scheme. The only question remaining is whether these identifiers reference data dictionary entries.

Minimally, a data dictionary entry must have three parts: a unique identifier, a term (name), and a clear definition. Therefore, for the second set of tuples in the above example to be compliant, there must exist a data dictionary entry in the form

Identifier ICTIP.Property.ABC.105
Term Individual_Income_Bracket
Definition Range of individual income given in increments of $25,000 starting at $0 and coded with single letters "A", "B", "C", "D", and "E" where "A" for [0−25,000], "B" for [25,001−50,000], "C" for [50,001−75,000], "D" for [75,001−100,000], and "E" for [100,000 and above].

One weak point of the standard is that it only specifies the properties be "clear and well-defined," a quite subjective statement. However, this weakness can be offset somewhat by the data specification part of the requirement discussed next.

The standard does note that, in order to understand the meaning of a property value, its data type should always be given. The standard allows for the data type of a property to be given in several ways, including:

- Explicitly in the property value, e.g. quotation marks to indicate string values.
- In the data dictionary definition of the property as in the foregoing example.
- Reference to a data dictionary entry for data type.
- Reference to a data specification entry, which includes a data type specification.

Another requirement is the data dictionary must be accessible to the receiver of the message. Here again the standard allows for flexibility in compliance including:

- Providing a downloadable version of the entire dictionary from the Internet (downloadable free of charge).
- Making the data dictionary interactively accessible through an API available through the Internet (usable free of charge), e.g. web services using SOAP.
- Inserting data dictionary entries needed in the message into the same dataset (message) as the property value. If this last option is used, the data dictionary entries must also be defined and supported by the message syntax.

The standard is also quite flexible as to the overall schema for the message itself. It does not specify what the message syntax should be, only that it must have one that is machine readable. Generally there are two approaches, a single-record schema and a multiple-record schema.

In a single-record scheme the actual instance of a master data reference can mirror its logical structure where the reference is a sequence of property-value pairs. Figure 11.1 shows the structure of a single-record message referencing an external data dictionary.

The problem with representing each instance of a master data reference as a set of property-value pairs is when there are multiple instances of master data references in the same message, and all of the references have the same properties, then the message will contain a large amount of redundant data. There is no need to repeat the property reference for each property value in every record.

Figure 11.2 shows a schema in which each property definition is referenced only one time, and each instance comprises only the set of property values listed in the same order as the property definition references.

PART 4: CONFORMANCE TO DATA SPECIFICATIONS

The conformance to data specifications has three requirements:

1. Each master data message shall contain in its header a reference to the data specification or specifications to which the master data message complies.
2. Each reference shall be in the form of an unambiguous identifier for the specific version of the data specification used to encode the master data message.

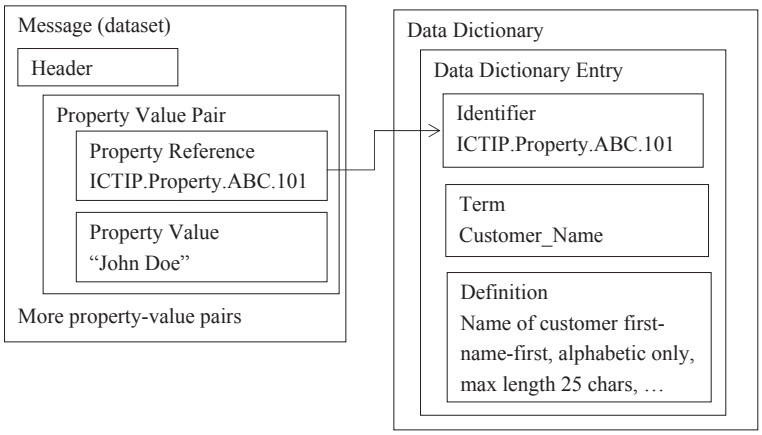

FIGURE 11.1

Single-record message structure.

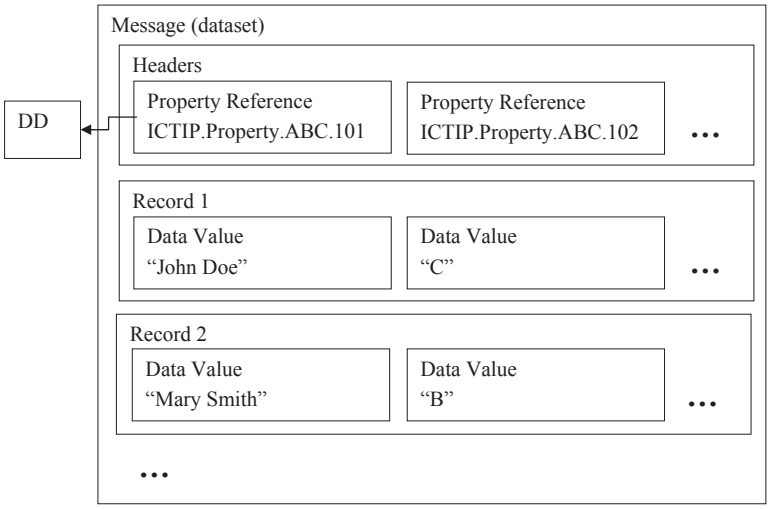

FIGURE 11.2

Multiple-record schema.

3. All referenced data specifications shall be available to all interested parties. If the master data is offered to the public, then all referenced data specifications shall be publicly available. The data specifications should be available at a reasonable cost.

Notice the similarity between this requirement and the semantic encoding requirement. In the semantic encoding requirement, the identifier points to a data dictionary entry that has the three parts — identifier, term, and definition. In the

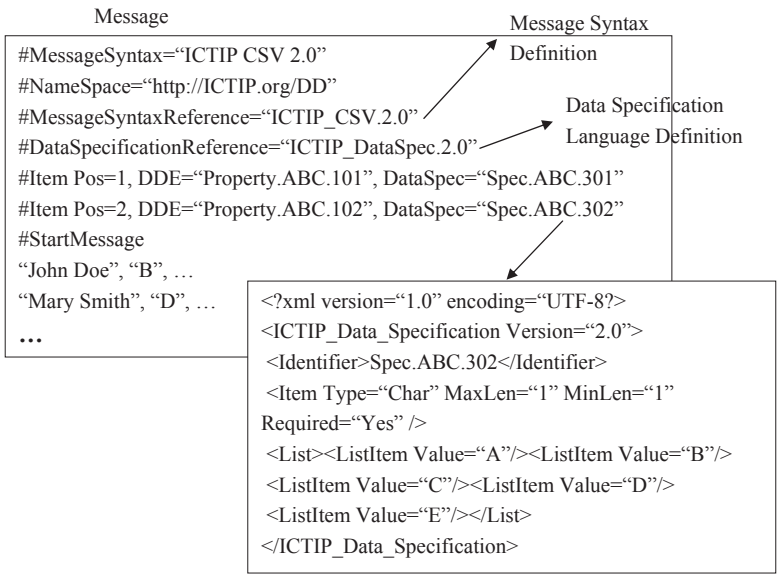

FIGURE 11.3

Message referencing a data specification.

conformance to data specification, the identifier points to a data specification. The main difference is that the definition of the property in the data dictionary is for human interpretation, but a data specification is for machine interpretation.

A data specification is basically a formalization of the data dictionary definition so as to make it actionable. Take as an example the data dictionary entry for Individual_Income_Bracket given earlier. Note the property definition laid out several specifications for this value. For example, it stated the values must be single alphabetic characters "A" through "E" with specific semantics. However, the data dictionary entry is for human reference; its purpose is to assure the sender and receiver have a common understanding of the data elements being transmitted. On the other hand, an ISO 8000 data specification is a set of machine interpretable instructions that will allow the receiver to verify by software whether data values are actually in compliance with the specification.

Take, as an example, the message and supporting references shown in Figure 11.3. The syntax of the message is that the text lines beginning with the "#" character comprise the header (metadata) section of the message. The first line identifies the message syntax. The second line gives the address of a web service where items referenced in the message can be found. The third line is a reference to the location where the complete and formal definition of the message syntax can be found. (Note that neither ICTIP_CSV.2.0 nor ICTIP_DataSpec.2.0 are actual formally defined syntaxes, although they could be. The sections of the message shown in the example are only intended to illustrate the concepts.)

Similarly, the fourth line is a reference to the location where the complete and formal definition of the data specification language can be found. The overall structure of the message follows the multiple-record scheme where each record in the message has two properties defined by the next two "Item" lines. Line five references Property.ABC.101, which is the Customer_Name data dictionary entry. The sixth line references the Individual_Income_Bracket data dictionary entry.

In the message schema shown in Figure 11.3, a property can have more than one identifier. The first identifier is the required reference to a data dictionary entry that defines the property. The second identifier is a reference to a data specification. For example, the Individual_Income_Bracket property references the data specification with the identifier "Spec.ABC.302". The content of the data specification is shown in the inset.

The data specification language used in this example is a (made-up) restricted set of XML elements. Although the formal semantics of the language are not shown here, they can be inferred from the names of the structure and names of the tags. The first part of the specification is given by the <Item> tag. It defines the data type as character, gives the requirement that the value must be present in every record, and the requirement that its length must be exactly one character. The second part of the specification is given by the <List> tag. The <List> tag encloses a set of <ListItem> tags that define the allowable property values "A", "B", "C", "D", and "E".

As long as the receiver has an interpreter for the data specification language, any incoming message with property value specifications written in that language can be verified for compliance with the specification. In this example, the receiver's software could verify all values of the second property (individual income bracket) are present and the value is one of the five character values in the list.

One interesting aspect of the ISO 8000 standard is data specifications are not required. The standard only requires the message contains a reference to the definition of a data specification language "if" data specifications are given. If the message is sent without data specifications other than the property definitions, then the message can still be ISO 8000-110 compliant. This means there are two levels of ISO 8000 compliance, simple and strong.

SIMPLE AND STRONG COMPLIANCE WITH ISO 8000-110

In order for an organization's master data messages to be in compliance with the ISO 8000-110 standard, certain things are mandatory. The primary requirement is the existence of a freely available data dictionary in which every property used in the message is defined. Furthermore, the identifiers must be unique and follow an international standard. The data dictionary entries must have at least an identifier, a term, and a clear definition.

Next, the organization must either define or adopt a formal syntax for its messages. Several existing standards, for example ISO 22745, ISO 13584, and EDIFACT, define complete message syntaxes. Of these, ISO 22745 is the most flexible, and it is probably the best candidate for introducing ISO 8000 compliance into other MDM domains besides parts cataloging.

Compliance with the syntax and semantic encoding parts of the ISO 8000-110 standard is sufficient for overall compliance because the data specifications are optional. A data specification syntax and language only need to be referenced in the message if the message includes data specifications. Simple compliance to ISO 8000-110 does not include data specifications.

However, simple compliance fails to realize the full power of the standard which lies in the automated verification of the message against data specifications. In order to have strong compliance the organization must either define or adopt a formal data specification language, and insert data specifications in its master data messages.

ISO 22745 INDUSTRIAL SYSTEMS AND INTEGRATION

The ISO 22745 standard defines a specific implementation of ISO 8000-110 that includes both a message syntax and a data specification syntax based on a restriction of XML. ISO 22745 was primarily designed to support parts cataloging. It has been adopted by the North Atlantic Treaty Organization (NATO) for its supply chain management. There are also some commercial software solutions that implement the ISO 22745 standard including the PiLog® Data Quality Solution (PiLog, 2014).

Different parts of the ISO 22745 standard define the components necessary to meet the ISO 8000-110 standard. These include

- ISO 22745-10:2010(E) – Open Technical Dictionary (OTD) to meet data dictionary requirements of ISO 8000-110.
- ISO 22745-30:2009(E) – Identification Guide (IG) to meet the machine readable data specification language of ISO 8000.
- ISO 22745-40:2010(E) – Master data representation to meet the message syntax of ISO 8000.

The documentation of these standards is much lengthier and much more detailed than for the basic ISO 8000 framework. This is because these standards must formally define the data dictionary syntax, the message syntax, and the data specification syntax and logic.

BEYOND ISO 8000-110

After the initial development of the ISO 8000-110 standard, several smaller parts were added. The new parts of the standard try to address data quality dimensions difficult to put into a data specification language. Take, as an example, the property Individual_Income_Bracket, defined earlier. It is easy to see how a machine-readable data specification could be designed that would verify that the values for this property are only the characters “A” through “E” such as the specification instructions illustrated in Figure 11.3.

However, it is less clear how to verify that when the value “A” is used, the actual income of the customer is between $0 and $25,000, as required by the definition. This would require in the field verification and represent a measure of

accuracy. Accuracy cannot be verified by applying a rule to the property value. Accuracy requires verification against the primary source, or against other records verified against the primary source, so-called "golden records." In short, data specifications are a powerful way to perform data validation, but they will not work for all data requirements.

PART 120: PROVENANCE

In its broadest sense, data provenance is the history of a data item from the time of its creation to the present. Provenance is a term commonly used in the art world, something art and auction galleries must verify, especially for high-value, historical works. Here provenance is being able to account for and scrupulously document all of the owners from the original artist to the present day without any gaps. The same concept is used in the court system where the chain of custody must be established for any physical evidence before it can be introduced at trial.

The ISO 8000-120 standard does not go quite that far. It basically looks at the most recent owner of the data. At the data element level, the standard requires the specification of two things:

1. When the data was extracted from the database.
2. The owner of the database.

PART 130: ACCURACY

As stated earlier, accuracy is a measure of how closely data represent the state of the real world at any given time. Usually, but not always, it means the current state of the world. For example, accuracy for customer address is usually understood to be the current address of the customer. However, previous addresses can be historically correct, i.e. correct at the time, but are no longer current. Because it relies on verification, rather than rule validation, accuracy is perhaps the most difficult of all the data quality dimensions to measure.

The ISO 8000 approach to accuracy is as follows. At the data elements level:

- The organization claiming the accuracy must be identified.
- The accuracy can either be covered by a warranty or be asserted.
  - If covered by warranty, the place where the warranty statement and terms can be found must be provided
  - If the accuracy is asserted, the location where a description of the assertion that explains why the data are believed to accurate must be provided.

PART 140: COMPLETENESS

Completeness is a data quality measure of the amount of data provided in proportion to the amount of data possible. Completeness can be measured at several levels. The two most common are at the population or data set level and the depth of

completeness at the record level. For example, in a customer MDM system, population completeness would be the proportion of all customers of the company who are actually represented in the system and under management. On the other hand, at the depth or record level the question might be for the 20 attributes collected for each customer and what proportion of these values are actually present.

It is important to note here that a missing value is not necessarily a null value. Certainly a null value is a missing value, but a value can also be missing because it is an empty string, a blank value, or a placeholder value.

The ISO 8000 approach to completeness is simply to require that the organization claiming completeness must be identified.

CONCLUDING REMARKS

Whether ISO 8000 compliance produces value for an organization will depend on several factors. One of these factors is the position of the organization with respect to being primarily a data consumer or a data provider. The ISO 8000 was born from a consumer perspective, i.e. the need for a data consumer to have a systematic way of requisitioning equivalent replacement parts from multiple suppliers.

The lack of clear understanding about the syntax, semantics, and specifications of datasets exchanged between data providers and data consumers is a common problem in many industries and one of the primary motivations for MDM in the first place. Many organizations receive data from numerous sources and spend inordinate amounts of time and effort to rationalize the sources into the same format and semantics. If the data are references to real-world entities, then there is the further process of entity resolution.

Just as in the example of the ABC Bank at the beginning of this chapter, publishing data definitions and specifications, requiring data suppliers to conform to those definitions and specifications, and being able to verify their compliance automatically, could result in enormous cost savings and increased productivity. Common standards for these functions also helps to solve the problem of how to structure service level agreements that govern data quality requirements for data acquired from third parties (Caballero et al., 2014).

One consideration in adopting the ISO standards is how much initial time and effort it will take to create the definitions and specifications. A second is whether the organization has enough influence or authority to require that their suppliers conform to their data quality specifications.

From the data producer viewpoint, compliance to ISO 8000 can make the data products of an organization easier to use and understand. This could in turn create a competitive advantage and increased market share. Consumers of the information will know how to interpret the data and understand exactly what is in it.

This page intentionally left blank