TECHNICAL REPORT

ISO/IEC TR 20943-3

First edition
2004-03-01

Information technology — Procedures for achieving metadata registry content consistency —

Part 3: Value domains

Technologies de l'information — Procédures pour réaliser la consistance du contenu de l'enregistrement des métadonnées —

Partie 3: Domaines de valeur

Reference number
ISO/IEC TR 20943-3:2004(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

In exceptional circumstances, the joint technical committee may propose the publication of a Technical Report of one of the following types:

— type 1, when the required support cannot be obtained for the publication of an International Standard, despite repeated efforts;

— type 2, when the subject is still under technical development or where for any other reason there is the future but not immediate possibility of an agreement on an International Standard;

— type 3, when the joint technical committee has collected data of a different kind from that which is normally published as an International Standard (“state of the art”, for example).

Technical Reports of types 1 and 2 are subject to review within three years of publication, to decide whether they can be transformed into International Standards. Technical Reports of type 3 do not necessarily have to be reviewed until the data they provide are considered to be no longer valid or useful.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC TR 20943-2, which is a Technical Report of type 3, was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 32, *Data management and interchange.*

ISO/IEC 20943 consists of the following parts, under the general title *Information technology — Procedures for achieving metadata registry content consistency*:

— *Part 1: Data elements* [Technical Report]

— *Part 3: Value domains* [Technical Report]

The following parts are under preparation:

— *Part 2: XML structured data*

— *Part 4: Overview*

Introduction

The exchange of metadata between metadata registries based on ISO/IEC 11179, *Information technology — Metadata registries* (all parts), depends not only on registry software that conforms to the standard, but also on metadata contents that are comparable between registries. While the standard has provisions for data specification and registration, there are pragmatic issues pertaining to populating the registries with content. Based on the experiences of organizations that are implementing the standard, technical reports to explore content issues will help current and future users.

Metadata registries can be used to register data elements, value domains, other objects, and associated attributes for many kinds of organizational data resource collections. Metadata registries can store information describing value domains used to specify the allowed values of a data element, the codes in a standard list, and classification schemes.

This technical report is based on ISO/IEC 11179-3:2003 of the six-part ISO/IEC 11179 International Standard that describes the organization of a registry for managing the semantics of data. The standard specifies the structure of a registry in the form of a conceptual model. The conceptual model is not intended to be a logical or physical data model for a computer system.

ISO/IEC 11179-3:2003, models a value domain and an associated conceptual domain. Conceptualization and articulation of rules and relationships are needed in the creation of conceptual domains and value domains. Reuse of value domains should be enabled and regularized. *Elementarily equivalent domains* have a relationship between their values that needs to be captured in a metadata registry. Some *conceptually equivalent domains* have relationships between their values, too. These also need to be captured. This Technical Report describes how this can be accomplished.

While metadata registries can be used for storing information about a variety of metadata items, this Technical Report addresses only value domains, conceptual domains, and their associated attributes and relationships. The goal of this paper is to ensure that there is a common understanding of the content of the value domain attributes so that metadata can be shared between registries, despite their differences.

Information technology — Procedures for achieving metadata registry content consistency —

Part 3: Value domains

1 Scope

1.1 Background

An ISO/IEC 11179 metadata registry (MDR) is a tool for the management of shareable data; a comprehensive, authoritative source of reference information about data. It supports the standardization and harmonization processes by recording and disseminating descriptions of data, which facilitates data sharing among organizations and users. It provides links to documents that refer to specific data elements, value domains, and classification schemes and to information systems where those objects are used. When used in conjunction with a database, the registry enables users to understand any information obtained from the database better.

A registry does not contain data itself. It contains the metadata that is necessary to clearly describe, inventory, analyse, and classify data. It provides an understanding of the meaning, representation, and identification of units of data. This International Standard identifies the information elements that need to be available for determining the meaning of data to be shared between systems.

1.2 Purpose

The purpose of this Technical Report is to describe a set of procedures for the consistent registration of value domains and their attributes in a registry. This Technical Report is not a data entry manual, but a user's guide for conceptualizing a value domain and its components for the purpose of consistently establishing good quality metadata. An organization may adapt and/or add to these procedures as necessary.

1.3 Limits of this Technical Report

The scope of this Technical Report is limited to value domains, conceptual domains, and their associated attributes and relationships. Examples are used throughout the TR to illustrate the concepts described.

1.4 Registration approach — value domains and data elements

There is a choice when registering value domains in an MDR. Some Registration Authorities treat these sets as value domains, and others treat them as data elements. For the purposes of this Technical Report, the choice will always be to treat the sets as value domains unless explicitly stated. This choice is made to help illustrate the way to register many different kinds of value domains.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 11179-1, *Information technology — Metadata registries (MDR) — Part 1: Framework for the specification and standardization of data elements*

ISO/IEC 11179-2, *Information technology — Specification and standardization of data elements — Part 2: Classification for data elements*

ISO/IEC 11179-3, *Information technology — Metadata registries — Part 3: Registry metamodel and basic attributes*

ISO/IEC 11179-4, *Information technology — Metadata registries (MDR) — Part 4: Rules and guidelines for the formulation of data definitions*

ISO/IEC 11179-5, *Information technology — Specification and standardization of data elements — Part 5: Naming and identification principles for data elements*

ISO/IEC 11179-6, *Information technology — Metadata registries (MDR) — Part 6: Registration of data elements*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 11179 and the following apply.

3.1
conceptually equivalent domains
value domains that represent the same conceptual domain

3.2
elementarily equivalent domains
domains that are elementarily equivalent if there exists a one-to-one correspondence between their permissible values such that given any pair of corresponding permissible values their value meanings are equal

NOTE 1 See Example in 4.2.5.

NOTE 2 All elementarily equivalent domains are conceptually equivalent. This follows from the fact that elementarily equivalent domains have the same set of value meanings, therefore they represent the same conceptual domain.

NOTE 3 Elementarily equivalence is an equivalence relation on the set of all enumerated value domains. So, any number of enumerated value domains may be elementarily equivalent to each other. See Examples in 5.4.1.

4 Understanding value domains

4.1 Introduction

This section is devoted to describing several things about value domains:

1) Some general principles about value domains

2) The structure or relationships that exist in some value domains

3) Code sets as value domains

4) Classification schemes as value domains

5) The relationship of data types to value domains

6) Use of units of measure

7) The importance of dimensionality

8) Classifying value domains

Examples are used throughout to illustrate the ideas. See Annex A for a detailed model (from ISO/IEC 11179-3) illustrating the relationships among all the constructs described herein.

4.2 General principles

4.2.1 Introduction

A *Value Domain* is a set of permissible values. A *Permissible Value* is a combination of some value and the meaning for that value. The associated meaning is called the *Value Meaning*. A permissible value is represented in this Technical Report as an ordered pair delimited by angle brackets as follows: <*value, value meaning*>. A value domain is the set of valid values for one or more data elements. It is used for validation of data in information systems and in data exchange. It is also an integral part of the metadata needed to describe a data element. In particular, a value domain is a guide to the content, form, and structure of the data represented by a data element.

Value domains come in two main types: enumerated and non-enumerated. An *Enumerated Value Domain* is a value domain where all the permissible values are listed explicitly. Examples of types of enumerated value domains include code sets, standard classifications, and categorizations. A *Non-enumerated Value Domain* is a value domain where the permissible values are expressed using a rule, called a *Non-enumerated Value Domain Description*. Thus, the permissible values are listed implicitly. This rule specifies precisely which values belong to the value domain and which do not. Examples of types of non-enumerated value domains include intervals of numbers, character strings, and bit maps.

A *Conceptual Domain* is a set of value meanings. It is a concept for which the extension is a collection of value domains. Conceptual domains, too, come in two main types: enumerated and non-enumerated. The value meanings for an *Enumerated Conceptual Domain* are listed explicitly. This type of conceptual domain corresponds to the enumerated type for value domains. The value meanings for a *Non-enumerated Conceptual Domain* are expressed using a rule, called a *Non-enumerated Conceptual Domain Description*. Thus, the value meanings are listed implicitly. This rule describes the meaning of permissible values in a non-enumerated value domain. This type of conceptual domain corresponds to the non-enumerated type for value domains.

Every value domain represents two kinds of concepts: data element concept (indirectly) and conceptual domain (directly). The *Data Element Concept* is the concept associated with a data element. The value domain is the representation for the data element, and, therefore, indirectly represents the data element concept, too. However, the value domain is directly associated with a conceptual domain, so represents that concept, independent of any data element.

An example will help to illustrate the distinctions in the discussion, which is shown below:

EXAMPLE

Data element name:	Sex of employee – code
Data element concept name:	Sex of employee
Data element concept definition:	The sex of the employee of an organization.
Conceptual domain name:	Human sex categories
Conceptual domain definition:	Enumerations of human sexes.
Value domain name:	Human sex codes (1)
Value domain definition:	Codes for the human sexes.
Permissible values:	<1, Male>
	<2, Female>
	<0, Unknown>

The codes used in the value domain above are taken from ISO/IEC 5218. Using standardized codes ensures interoperability between metadata registries and application systems. However, in general, the choice of codes for a value domain may be arbitrary. In this case, the MDR is the source for obtaining the values and their meanings for a non-standard value domain.

Several points about value domains need to be made here.

4.2.2 Choice of codes

The choice of codes used in the value domain above is arbitrary. Another code set might work just as well, but the set is a different value domain. Which value domain to use is determined by the needs of the application and the organization. The following example is another code set for human sex codes:

EXAMPLE

Value domain name:	Human sex codes (2)
Value domain definition:	Codes for the human sexes.
Permissible values:	<M, Male>
	<F, Female>
	<U, Unknown>

4.2.3 Number of permissible values

The number of permissible values (3 in our example) may also be different. We might want a code for representing hermaphrodites or a code for representing transsexuals. Each time new permissible values are added or subtracted, a new value domain, or value domain version, is created. Determining whether a change to a value domain merits the creation of a new value domain or just a new version of an existing value domain is up to the individual registration authority. The following example shows an expanded value domain considered as a new one, not a version of an old one, as in the example in 4.2.2.

EXAMPLE

Value domain name:	Human sex codes (3)
Value domain definition:	Codes for the human sexes.
Permissible values:	<M, Male>
	<F, Female>
	<H, Hermaphrodite>
	<T, Transsexual>
	<U, Unknown>

4.2.4 Conceptual domain hierarchies

All the value domains for human sex codes can be viewed as being conceptually equivalent. There is no requirement that each of the value meanings in a conceptual domain be associated with a value. However, some Registration Authorities may decide that to adequately differentiate the concept, for example, of using five categories of human sex codes instead of three, separate conceptual domains must be created. See example below. At the highest level, all the value domains (examples in 4.2.1, 4.2.2, and 4.2.3) represent the idea of categories of human sexes. So, the super-ordinate conceptual domain captures the concept represented by a class of value domains (e.g., human sex codes) needed within a registry. The subordinate conceptual domains provide the enumeration of value meanings to be mapped to the corresponding value domains.

EXAMPLE

Super-ordinate conceptual domain	(non-enumerated)
Conceptual domain name:	Human sex categories
Conceptual domain definition:	Categorizations of human sexes.
Subordinate conceptual domain (1)	(enumerated)
Conceptual domain name:	Human sex categories: 3 values
Conceptual domain definition:	Enumerations of human sexes with 3 categories.
Subordinate conceptual domain (2)	(enumerated)
Conceptual domain name:	Human sex categories: 5 values
Conceptual domain definition:	Enumerations of human sexes with 5 categories

4.2.5 Sharing value meanings across permissible values

The value meaning is used to link equivalent permissible values across conceptually equivalent domains. In elementarily equivalent domains, each value meaning links equivalent codes between a unique pair of permissible values, one from each value domain, as the following example illustrates:

EXAMPLE

Conceptual domain name:	Human sex categories
Conceptual domain definition:	Enumerations of human sexes.
Value domain names:	Human sex codes (1) (See Example, 4.2.1)
	Human sex codes (2) (See Example, 4.2.2)
Value domain definition:	Codes for the human sexes.

A one-to-one correspondence that preserves value meanings between these two enumerated value domains is defined as follows:

HSC(1)	|→	HSC(2)
<1, Male>	↔	<M, Male>
<2, Female>	↔	<F, Female>
<0, Unknown>	↔	<U, Unknown>

Each pair of corresponding permissible values has the same value meaning. So, these two enumerated value domains are elementarily equivalent and, therefore, conceptually equivalent.

Each permissible value in one of the two value domains listed above shares its value meaning with that of a permissible value in the other value domain. So, through the use of value meanings, equivalence of values across value domains is achievable, e.g., the values 1 and M mean Male or the values 2 and F mean Female. These two value domains are elementarily equivalent domains.

4.2.6 Sharing value domains across data elements

Sex of employee (the idea that employees are classified or characterized by sex) and *sex of student* (the idea that students are classified or characterized by sex) are different data element concepts, but they could use the same value domain to represent them. So, a value domain (e.g., Human Sex Codes (1)) may be associated with many data element concepts, and, therefore, data elements.

4.2.7 Associating value domains with concepts (data element concepts and conceptual domains)

A data element concept is associated with different value domains as needed to describe similar, but different, data elements, and those value domains are conceptualized by the same conceptual domain (e.g., Human Sex Codes (1), Human Sex Codes (2), Human Sex Codes (3) in the examples). However, the converse is not true: two value domains under the same conceptual domain do not need to be associated with the same data element concept. The following two examples (1 and 2) are of this type:

EXAMPLE 1

Conceptual domain name:	Human sex categories
Conceptual domain definition:	Enumerations of human sexes.
Value domain name:	Human sex codes (1) (See Example in 4.2.1)
Value domain definition:	Codes for the human sexes.
Data element concept name:	Sex of employee
Data element concept definition:	The biological sex of the employee of an organization.

EXAMPLE 2

Conceptual domain name:	Human sex categories
Conceptual domain definition:	Enumerations of human sexes.
Value domain name:	Human sex codes (2) (See Example in 4.2.2)
Value domain definition:	Codes for the human sexes.
Data element concept name:	Sex of student
Data element concept definition:	The biological sex of the student of an educational institution.

4.2.8 Value domains not associated with data elements

Value domains do not have to be associated with a data element concept at all. They can be managed independently, such as code sets sometimes are. For instance, the maintenance agency, which is the authority for maintaining the code values for a standard code set, might make the code set freely available even though it published no data using that value domain.

4.2.9 Contrasting conceptual domains and data element concepts

There are two kinds of semantics associated with data: symbolic and contextual. Symbolic semantics refers to the meaning of symbols, i.e. values. The conceptual domain captures this kind of semantics, as it is a set of value meanings.

Contextual semantics addresses the interrogatives (who, what, when, where, why, and how) as they relate to data. Fundamentally, data represent some observation of a property of a member of some set of objects, an object class. The term observation usually implies some human action, but mechanical or electrical instruments can make or record observations, too.

The object class describes who is being observed, the property is the distinguishing or describing feature of an object that is observed, and the instrument (including humans) is how the observation is made. Both the object class and property are concepts, and they form the basic contextual meaning associated with data. Without them, data have no reason to be. The data element concept captures this meaning.

Details about the instrument (how) and why observations were made (the experimental design) are out of scope for the MDR. The time (when) an observation was made and where it was made are specific to individual observations. This case level metadata is often found in a data record in companion data elements in a database.

4.2.10 Non-enumerated value domains

The examples provided so far are for enumerated value domains. Non-enumerated value domains are used to represent and constrain data through a rule rather than through an enumerated list of permissible values. The following example is of a non-enumerated value domain and associated data element and data element concept:

EXAMPLE	
Conceptual domain name:	Industry descriptions
Conceptual domain definition:	Text describing an industry.
Non-enumerated conceptual domain description:	Limited length text describing an industry
Value domain name:	Textual English industry descriptions
Value domain definition:	Textual descriptions of an industry.
Non-enumerated value domain description:	English text up to 60 characters
Data element name:	Industry description for person's job - text
Data element concept name:	Industry description for person's job
Data element concept definition:	The description of the industry within which a person works.

4.2.11 Value domains with enumerated and non-enumerated components

It is possible, although rare, for a domain to have enumerated and non-enumerated components. This situation may occur when values fall within a certain range, have a minimum (or maximum) value, and discrete values below the minimum (or above the maximum) are used for special cases. The following example (1) will illustrate this:

EXAMPLE 1	
Data element name:	Volume of household monthly water usage - gallons
Data element concept name:	Volume of household monthly water usage
Data element concept definition:	Volume of water used by a household each month.
Conceptual domain name:	Volumes (liquid)
Conceptual domain definition:	Measures of liquid volume with additional special values.
Non-enumerated conceptual domain description:	Liquid volume measures
Value domain name:	Volume in gallons (US, liquid) x 1000
Value domain definition:	Liquid volume in thousands of US gallons.
Non-enumerated value domain description:	Integers greater than or equal to 0, and special values -1 and -2
Permissible values:	<-1, Not reported>
	<-2, Not measurable (e.g., use of well or stream)>

Another situation where a value domain has both enumerated and non-enumerated parts is illustrated in the following example (2). Here, special meaning is attached to a finite number of values in a range.

EXAMPLE 2	
Data element name:	Likelihood of voting in the next election, on a scale of 0 to 1
Data element concept name:	Likelihood of voting in the next election
Data element concept definition:	A measure of how likely the person will vote in the next election.
Conceptual domain name:	Rating scales
Conceptual domain definition:	A measure of effort, attitude, preference, difficulty, etc.
Non-enumerated conceptual domain description:	Rating scale expressed numerically
Value domain name:	Preference scale, 0 to 1
Value domain definition:	Preferences measured on a scale of 0 to 1.
Non-enumerated value domain description:	All real numbers between 0 and 1.
Permissible values	<0, Definitely not>
	<0.5, No preference>
	<1, Definitely>

NOTE This example requires some further explanation. The data may be obtained by reading a mark made by a respondent on a scale printed on a form. The value is interpolated through finding the ratio of the distance from the mark to the "0" end divided by the length of the entire scale. In addition, the use of both enumerated and non-enumerated qualities is fundamental to this value domain. It is possible to represent the same data element concept by reversing the preference scale, so that “0” means “Will participate” and “1” means “ Will not participate”. Therefore, value meanings associated with specific permissible values are required to describe these value domains fully.

4.2.12 Semantic restriction of use of value domains

The permissible values, in particular their value meanings, of the value domains used in the examples so far restrict the number of data elements they may be linked to. Human Sex Codes (1) is an enumerated value domain that can only be used as the valid values of a data element that represents the sex of (some) living things, because the permissible values describe sex categories. Likewise, the non-enumerated value domain in 4.2.11, example 2 (Preference Scale, 0 to 1), can only be used as the valid values of a data element that represents preferences, because the value meanings for some of the permissible values describe levels of preference. These two cases show that some value domains, by their construction, are limited in use by their semantics. They can represent only some kinds of data element concepts.

Enumerated value domains, in particular, seem especially linked to the data elements for which they are the valid values, and all value domains are limited in some regard. For instance, the value domain in the following example would not be used to represent the data element *Sex of Employee - Code*. However, the example also shows that some value domains are applicable in many situations:

EXAMPLE

Data element name:	Person voted in last election, yes or no
Data element concept name:	Person voted in last election
Data element concept definition:	Indication whether person voted in last election.
Conceptual domain name:	Yes or No representations
Conceptual domain definition:	Values for Yes and No.
Value domain name:	Yes or No codes
Value domain definition:	Codes for yes and no responses.
Permissible values	<0, No> <1, Yes>

4.2.13 Rapidly changing enumerated value domains (UPC example)

All the enumerated value domains discussed so far are essentially fixed. They are designed with a fixed set of permissible values in mind, although they might change occasionally. If an enumerated value domain is required to have more permissible values, then a new value domain is created with the additional permissible values (see Examples in 4.2.1, 4.2.2, and 4.2.3).

Some enumerated value domains change (and grow) rapidly. An example is the Universal Product Code (UPC). The UPC is a 12-digit code placed on products that uniquely identifies them and their manufacturers. The code is expressed numerically with 12 digits and pictorially in the form of 55 alternating dark and light lines. These lines have 4 standard widths, each set of 4 consecutive lines determines a digit. There are 3 lines at the start, 5 in the middle after the first 6 digits, and 3 at the end. The first 6 digits identify the manufacturer, the next 5 digits identify the item, and the last digit is a check-digit. Retailers use the codes to manage inventory, set prices, and speed point-of-sale. Manufacturers apply to join the UPC system by registering with the Uniform Code Council (UCC) and paying a membership fee. Then, the UPC Coordinator for the manufacturer manages item identifiers for all its products, retiring numbers when products are no longer offered, and making sure all products are uniquely identified.

It may not make sense to manage a rapidly changing value domain in a MDR. The many changes would require the formation, registration, and administration of too many value domains. Alternatives are to manage snapshots of the value domain, updated periodically, or to maintain references or pointers to the value domain in the registry. Of course, the decision as to how to manage a rapidly changing value domain is up to the Registration Authority.

Alternatively, one could manage a rapidly changing enumerated value domain as a non-enumerated value domain. This has the advantage of maintaining all the metadata in the registry. There are significant disadvantages, however, which the UPC example illustrates. A full description of a valid UPC code exists, and this description fulfills the criteria for a non-enumerated value domain description. However, there is no way to know whether a particular UPC code is in use, and there are obvious advantages to knowing this kind of information. Maintaining the rapidly changing value domain as an enumerated value domain, as difficult as that may be, is the only way to know which codes are in current use.

4.3 Structure in value domains

4.3.1 Introduction

Often, value domains have relationships between them. These can occur in many ways. There are relationships between value domains and between the individual values in (possibly) different value domains. Any time there is a relationship between individual values in different value domains, there is a relationship between the value domains, too.

Value domains and permissible values can have relationships between them. Conceptual domains can be related to each other, too, so the shared conceptual domain for two related value domains may not be the conceptual domain for the individual value domains. A conceptual domain does not have to be represented by any value domains in a MDR.

There are two main types of relationships that need to be described: relationships between value domains, and relationships between values in one or more value domains. Both kinds of relationships are illustrated in the following real examples detailed in the next sub-clauses.

4.3.2 International Standard Industrial Classification (ISIC)

The **International Standard Industrial Classification** (ISIC, rev. 3), maintained by the United Nations Statistics Division[1]), is a four level hierarchy used to classify industries for economic analyses within countries and internationally (see also 4.5). The four levels have names and they represent levels of detail in the hierarchy. The names of the levels and the number of items in each level are

Level 1:	Tabulation Categories	17 items
Level 2:	Divisions	60 items
Level 3:	Groups	159 items
Level 4:	Classes	259 items

Each item at each of the first three levels has one or more items of detail at the next lower level. Here is an example:

EXAMPLE 1

Hierarchy:	International Standard Industrial Classification
Tabulation Category:	G - Wholesale and retail trade; repair of motor vehicles, motorcycles and personal and household goods
Division:	51 - Wholesale trade and commission trade, except of motor vehicles and motorcycles
Group:	513 - Wholesale of household goods
Class:	5131 - Wholesale of textiles, clothing and footwear

Four levels are shown within the wholesale and retail trade tabulation category. There are several choices at each level and explanatory notes to help the user understand the concepts.

The example indicates that the ISIC is really composed of four related value domains, one for each level. They are related for two reasons: 1) each domain (level) is a categorization of industries; 2) the values in each domain are a generalization of the items in the level below it, or the items in each domain are a specialization

1) ISIC is on the Web at the URL http://esa.un.org/unsd/cr/registry/regcst.asp?Cl=2&Lg=1.

of the item in the level above it. In the example above, for instance, item 51 generalizes item 513 (and items 511, 512, 514, 515, & 519). The items in the Group level are specializations of the item (51) in the Division level.

The following example shows the conceptual domains and value domains necessary to register ISIC:

EXAMPLE 2

Conceptual domain (general) name:	Industrial Classification Systems
Conceptual domain definition:	Nested levels of codes representing categories of industries.
Conceptual domain name:	Industrial classification systems, Level 1
Conceptual domain definition:	First level codes representing categories of industries.
Conceptual domain relationship:	specialization of {Industrial Classification Systems}
Value domain name:	Tabulation Category of ISIC
Value domain definition:	Codes for the tabulation (first) level of ISIC.
Permissible values:	<A, Agriculture, hunting and forestry>
	<B, Fishing>
	…
	<Q, Extra-territorial organizations and bodies>
Conceptual domain name:	Industrial classification systems, Level 2
Conceptual domain definition:	Second level codes representing categories of industries.
Conceptual domain relationship:	specialization of {Industrial Classification Systems}
Value domain name:	Division, ISIC
Value domain definition:	Codes for the division (second) level of ISIC.
Permissible values:	<01, Agriculture, hunting and related service activities>
	<02, Forestry, logging and related service activities>
	<05, Fishing, operation of fish hatcheries and fish farms; service activities incidental to fishing>
	…
	<99, Extra-territorial organizations and bodies>
Conceptual domain name:	Industrial classification systems, Level 3
Conceptual domain definition:	Third level codes representing categories of industries.
Conceptual domain relationship:	specialization of {Industrial Classification Systems}
Value domain name:	Group, ISIC
Value domain definition:	Codes for the group (third) level of ISIC.
Permissible values:	<011, Growing of crops; market gardening; horticulture>
	…
	<020, Forestry, logging and related service activities>
	…
	<050, Fishing, operation of fish hatcheries and fish farms; service activities incidental to fishing>
	…
	<990, Extra-territorial organizations and bodies>
Conceptual domain name:	Industrial classification systems, Level 4
Conceptual domain definition:	Fourth level codes representing categories of industries.
Conceptual domain relationship:	specialization of {Industrial Classification Systems}
Value domain name:	Class, ISIC
Value domain definition:	Codes for the class (fourth) level of ISIC.

Permissible values:	<0111, Growing of cereals and other crops n.e.c.> <0112, Growing of vegetables, horticultural specialties and nursery products> … <0200, Forestry, logging and related service activities> … <0500, Fishing, operation of fish hatcheries and fish farms; service activities incidental to fishing> … <9900, Extra-territorial organizations and bodies>

In this example, the relationships between the value domains are maintained through the relationship of the value domain to its conceptual domain and the fact that each conceptual domain is related to the same generalized conceptual domain. The relationships between the values of different domains are maintained through the codes (values) themselves. The number of characters in the code determines the level. The left most character is the first level code, the two left-most characters are the second level code, and so on.

The ISO/IEC 11179 metamodel for metadata registries does not support other types of relationships between values in two value domains. If the registration authority determines that this is necessary for the function of the registry, the necessary extensions to the metamodel need to be added.

4.3.3 Logical Observation Identifiers Names and Codes (LOINC)

Another example of a set of domains requiring relationships is the LOINC[2]. The Logical Observation Identifiers Names and Codes (LOINC) provides a standard set of universal names and codes for identifying individual laboratory results, clinical observations, and diagnostic study observations in the health care arena.

The LOINC has a basic structure defined in the *LOINC Users Guide* as follows:

The fully specified name of a test result or clinical observation has five or six main parts including:

- the name of the component or analyte measured (e.g. glucose, ibuprofin)
- the property observed (e.g. substance concentration, mass, volume); the timing of the measurement (e.g. is it over time or momentary)
- the type of sample (e.g. urine, serum), the scale of measurement (e.g., qualitative vs. quantitative)
- and where relevant, the method of the measurement (e.g., radioimmune assay, immune blot)

These can be described formally with the following syntax:

<Analyte/component>:<kind of property of observation or measurement>:<time aspect>:<system (sample)>:<scale>:<method>

The colon character, “:”, is part of the code and used as a separator.

The first part of the name can be further divided up into three subparts, separated by carats (^). The first subpart can contain up to three levels of increasing taxonomic specification, separated by dots (.). The third and fourth parts of the name (time aspect and system/sample) can also be modified by a second subpart, separated from the first by a carat. In the case of time aspect, the modifier can indicate that the observation is one selected on the basis of the named criterion (maximum, minimum, mean, etc.); in the case of system, the modifier identifies the origin of the specimen if not the patient (e.g., blood donor, fetus, blood product unit).

2) LOINC is on the Web at the URL http://www.regenstrief.org/loinc/loinc.htm.

Providing details for the LOINC is not instructive. The full description of LOINC is very complex and is beyond the scope for this Technical Report. From the short description above, it is clear that multiple code sets and levels are used to specify a LOINC record. A conceptual domain relationship is necessary to capture all the relationships across the various value domains. The following example illustrates how one begins registering the LOINC structured value domains. The time and scale dimensions are left out.

EXAMPLE

Conceptual domain (general) name:	Logical Observation Identifiers Names and Codes (LOINC)
Conceptual domain definition:	Inter-related sets of codes for identifying individual laboratory results, clinical observations, and diagnostic study observations in the health care arena.
Conceptual domain name:	Analyte
Conceptual domain definition:	The designation of the component or analyte measured.
Conceptual domain relationship:	specialization of {LOINC}
Value domain name:	Analyte
Value domain definition:	The name of the component or analyte measured.
Permissible values:	See LOINC web site
Conceptual domain name:	Property
Conceptual domain definition:	The designation of the property observed or measured.
Conceptual domain relationship:	specialization of {LOINC}
Value domain name:	Property
Value domain definition:	The name of the property observed or measured.
Permissible values:	See LOINC web site
Conceptual domain name:	Sample type
Conceptual domain definition:	The designation of the type of sample (e.g. urine, serum).
Conceptual domain relationship:	specialization of {LOINC}
Value domain name:	Sample type
Value domain definition:	The name of the type of sample (e.g. urine, serum).
Permissible values:	See LOINC web site
Conceptual domain name:	Method
Conceptual domain definition:	The designation of the method of the measurement.
Conceptual domain relationship:	specialization of {LOINC}
Value domain name:	Method
Value domain definition:	The name of the method of the measurement (e.g., radioimmune assay, immune blot)..
Permissible values:	See LOINC web site

4.4 Code sets as value domains

Code sets are value domains, because each code is a value and the meaning for each code is a value meaning. As in the sex codes example, using codes to designate well-known meanings is a useful technique. It started in the information technology community when people needed to store data about a characteristic of an object of interest (e.g., households, patients, factories, etc.) and use the fewest number of bytes when doing it. Using codes makes programming easier, because fixed length data is easier to handle than variable length. Also, it is easier to maintain codes in a database than it is descriptive words.

Code sets are used as value domains in many subject matter areas and standards. In the US health care community, the following quote[3)] describes the use of code sets:

> Under HIPAA [Health Insurance Portability and Accountability Act], a "code set" is any set of codes used for encoding data elements, such as tables of terms, medical concepts, medical diagnosis codes, or medical procedure codes. Medical data code sets used in the health care industry include coding systems for diseases, impairments, other health related problems, and their manifestations; causes of injury, disease, impairment, or other health-related problems; actions taken to prevent, diagnose, treat, or manage diseases, injuries, and impairments; and any substances, equipment, supplies, or other items used to perform these actions. Code sets for medical data are required for data elements in the administrative and financial health care transaction standards adopted under HIPAA for diagnoses, procedures, and drugs.

In the manufacturing standards community (ISO/TC184/SC4), the standard ISO 10303, *Industrial automation systems and integration – Product data representation and exchange* – contains many examples of the use of code sets as value domains for data elements. This example is taken almost verbatim from ISO 10303-41:2000, according to 21.3.19 – **si_prefix**.

EXAMPLE

An si_prefix is the name of a prefix that may be associated with an si_unit. The definitions (value meanings) of SI prefixes are specified in ISO 1000:1992, *SI units and recommendations for the use of their multiples and of certain other units*. Some example permissible values are:

Permissible values:	<exa,	10 to the 18th power>
	<peta,	10 to the 15th power>
	<tera,	10 to the 12th power>
	<pico,	10 to the -12th power>
	<femto,	10 to the -15th power>
	<atto,	10 to the -18th power>

Code sets are often defined in international standards, and some international standards documents contain multiple code sets. For example, in ISO 3166-1, *Country Codes*, the 3 code sets and 4 other enumerated value domains defined in the standard are elementarily equivalent. The meanings for each set of values are not repeated for each individual enumerated value domain.

Code sets are not necessarily enumerated value domains. Codes may be generated by a rule, thus, a code set specified by a rule for determining the codes is a non-enumerated value domain. It is also true that most typical examples of enumerated value domains are code sets. This may lead one to erroneously assume that every enumerated value domain is a code set. The list of first names for baby boys (and the meaningsof those names) is a value domain that is not a code set.

4.5 Classification schemes as value domains

Some code sets have structure built into them. Every code set with structure (as described above in 4.3) is both a classification scheme and value domain. Some people say that every code set, even every enumerated value domain, is a classification scheme. This distinction does not seem to be worth debating, but it is important to realize that many classification schemes (especially the ones that are also code sets) are also value domains.

The US health care community has adopted many international classifications for use as code sets in their work. The following list shows some of the classifications in use:

3) Quote is on the Web at the URL http://aspe.os.dhhs.gov/admnsimp/faqcode.htm.

International Classification of Diseases, 9th Edition, Clinical Modification, (ICD-9-CM), Volumes 1 and 2, for the following conditions:

1) Diseases

2) Injuries

3) Impairments

4) Other health related problems and their manifestations

5) Causes of injury, disease, impairment, or other health-related problems

International Classification of Diseases, 9th Edition, Clinical Modification, (ICD-9-CM), Volume 3 Procedures, for the following procedures or other actions taken for diseases, injuries, and impairments on hospital inpatients reported by hospitals:

1) Prevention

2) Diagnosis

3) Treatment

4) Management

The international statistical community has adopted several classifications as code sets, such as

International Standard Industrial Classification (ISIC) is used by statistical agencies and international statistical organizations to classify business establishments and to report economic statistics. Within organizations that maintain registers of business establishments, it is a code set for the data element that represents the industry category to which each establishment belongs.

Countries and groups of countries have their own standard industrial classifications (SIC), also. The ISIC is a four-tiered hierarchy SIC. In North America (Canada, Mexico, and United States), the **North American Industrial Classification System** (NAICS) was first published in 1997. It is a five-tiered hierarchy SIC of industry classes and codes, too. It is used similarly to the ISIC across countries, but each North American country has added more levels of detail to serve the needs of their own constituents. Other countries have similarly detailed classifications of industries.

Correspondence tables are created to map one SIC to another. They are especially important for mapping new versions of an SIC to the next older version of it. These correspondence tables are used by analysts to maintain time series tables across breaks in the series caused by the SIC updates. Correspondence tables are not limited to use for SICs. They are used for any standard classification or sets of similar classifications (see below). The ISO/IEC 11179 metamodel of a MDR does not contain a correspondence table class. If a registration authority deems it necessary to include correspondence tables in its registry, then local extensions to the metamodel must be made.

In addition to industrial classifications, there are standard occupational classifications (SOC) in use, too. Many countries have adopted SOC systems for use in reporting statistics. The occupations for which companies hire employees are classified, aggregated, and reported through the use of the SOCs. Individual descriptions of jobs by people are also classified using SOC and SIC together.

These examples provided above are in no way exhaustive. They are provided to give the reader an idea of the breadth of the use of classifications across subject matter areas and as code sets for data elements.

4.6 Data types and value domains

4.6.1 Basics

ISO/IEC 11404 defines data types as consisting of a value space (i.e., value domain) and sets of characterizing operations and properties. A value space by itself is not enough to specify a data type.

ISO/IEC 11179 only addresses value domains, however, identifying a data type with a value domain is important for sharing and understanding data. Data types are fundamental to the declaration of variables (data elements) in programming languages and are implicit in the construction of entities and classes in data modeling.

The example shows how the same data, in terms of meaning, can have different data types.

EXAMPLE		
#1		
	Value domain name:	Volume (in whole gallons)
	Value domain definition:	Volume amounts in whole gallons.
	Non-enumerated value domain description:	Non-negative integers
	Data type	Integer
#2		
	Value domain name:	Volume (in whole gallons)
	Value domain definition:	Volume amounts in whole gallons.
	Non-enumerated value domain description:	Non-negative integers
	Data type	Character

Data that belongs to each one of these value domains looks the same on a computer screen or printed on paper, but the data look much different to computer programs. Transferring data belonging to the integer data value domain requires writing a different program to receive it than for data belonging to the character data value domain. The data type does not provide all the information necessary to complete the transfer; some additional machine dependent parameters must be known, but they are out of scope of ISO/IEC 11179.

So, it is at the data transfer level that the data type becomes an issue. The difference is so important for applications to work properly that the value domains in the above example must be considered as different. Then, associating each value domain with the same data element concept requires the formation of separate data elements. In this way, the proper understanding of the data is ensured.

In the ISO/IEC 11179 metamodel for a MDR, *data type* has four attributes: *name*, *description*, *scheme*, and *annotation*. It is sufficient to supply the name of the data type, a description of its derivation (if applicable), the scheme from which the primitive or derivation came (e.g., ISO/IEC 11404), and any other appropriate descriptive notes. This way, the user of the MDR can understand the meaning of the listed data type, but the MDR does not need to support the machinery for deriving data types. This is especially important because there are many schemes for supporting data types, new ones are generated periodically, and it is impractical to build a model to support them all.

4.6.2 Value domains with more than one data type — limitations of value meaning

Sometimes a value domain may have more than one data type associated with it. This means that the value meanings associated with the permissible values are limited, in some sense. In the current MDR specification (ISO/IEC 11179-3) this also means that the value domain must be registered separately for each data type.

Following the ISO/IEC 11404 framework, a value domain is a value space with meaning associated with each value. So, a value domain has more than one data type assigned if the characterizing operations or properties change with each instance. The following example illustrates the idea. The two character alphanumeric country codes and their meanings in ISO 3166-1 are assumed to be the permissible values in each case. The data type of “state” is defined in ISO/IEC 11404.

EXAMPLE		
#1		
	Value domain name:	2-alpha country codes
	Value domain definition:	Two alphanumeric codes for countries of the world.
	Permissible values:	"Alphanumeric 2 character country codes and meanings"
	Data type	State (with additional properties defined by country physical boundary laws)
#2		
	Value domain name:	2-alpha country codes
	Value domain definition:	Two alphanumeric codes for countries of the world.
	Permissible values:	"Alphanumeric 2 character country codes and meanings"
	Data type	State (with additional properties defined by air space laws)

In this example, the value domain and, hence, the permissible values do not change; the properties associated with the permissible values do. Since the permissible values don't change, then the value meanings stay the same. So, the example shows that data type is associated with the use of a value domain in addition to its structure. The use here means that country is interpreted to mean physical boundary on the ground in one case and air space above the ground in the other.

The definitions of the data element concepts that are associated with the value domains are necessary in this example to understand the use distinctions. Air space and ground boundaries are concepts associated with country as a property of some object classes. It means that the data type is really associated with a data element. Further editions of ISO/IEC 11179-3 will clarify this.

4.7 Units of measure

The allowed values of value domains reflect how humans quantify the world. A quantification has a unit of measure associated with it. There are many types of measures, each with many choices for units of measure (see also 4.8), which the following list shows:

Revenues -	Dollars, Euros
Red blood cell counts -	Counts in millions per cubic millimeter
Mass -	Grams
Height -	Inches, Centimeters
Speed -	Miles per hour, Meters per second
Wavelength -	Ångströms
Concentrations -	Parts per million
Sedimentation -	Milliliters per hour

Usually, we encounter the unit of measure when analyzing the data represented by a data element. However, it makes sense from several perspectives to link units of measure with value domains when managing them in a MDR. The metamodel of a MDR described in ISO/IEC 11179-3:2002 links units of measure with value domains. ISO 1000:1992 contains a more detailed description of SI and other kinds of units of measure. See also the *Unified Code for Units of Measure*, by G. Schadow and C. McDonald (1999)[4].

4) *Unified Code for Units of Measure*, by G. Schadow and C. McDonald (1999), is on the Web at the URL http://aurora.rg.iupui.edu/UCUM/.

Consider enumerated value domains with the following example:

EXAMPLE 1

Conceptual domain name:	Categorizations of age ranges
Conceptual domain definition:	Categories representing ranges of ages.
Dimensionality:	Time
Value domain name:	Age categories
Value domain definition:	Categories of age defined by unequal length ranges.
Unit of Measure:	Year
Permissible values:	<1, Ages 0 to 15 years>
	<2, Ages 16 to 29 years>
	<3, Ages 30 to 64 years>
	<4, Ages 65 years and above>

The enumerated value domain above uses a unit of measure, years, to help label the ranges that define the categories that comprise the domain. Notice, the use of the unit of measure contained in the value meanings is fundamental to the understanding of each range. Another enumerated value domain with the same range values but different value meanings is a different domain, because an enumerated value domain contains permissible values – defined to be value/value meaning pairs.

The example shows some enumerated value domains need to have the unit of measure attached to them. There is no need to know any associated data elements to see that the unit of measure is required to confer meaning to the values in the permissible values of this domain. Recall, however, from the examples of sex codes in the sub-clauses of clause 4.2 that units of measure are not applicable for every enumerated value domain.

For non-enumerated value domains, the choice is the same. The units of measure associated with the allowed values of the domain help determine the "value meanings" for the domain. Different units of measure associated with the same range of values determine different value domains. The next example showing two value domains illustrates the point:

EXAMPLE 2

#1

Conceptual domain name:	Volumes
Conceptual domain definition:	Measures of volume.
Non-enumerated conceptual domain description:	Volume measures
Dimensionality:	Volume
Value domain name:	Volume (in whole US gallons
Value domain definition:	Volume amount in whole US gallons.
Non-enumerated value domain description:	Non-negative integers
Unit of Measure:	US gallon

#2

Conceptual domain name:	Volumes
Conceptual domain definition:	Measures of volume.
Non-enumerated conceptual domain description:	Volume measures
Dimensionality:	Volume
Value domain name:	Volume (in whole liters)
Value domain definition:	Volume amount in whole liters.
Non-enumerated value domain description:	Non-negative integers
Unit of Measure:	Liter

The value domains in the examples above are conceptually equivalent and share the same non-enumerated value domain description. However, the same numbers from each domain (with the exception of zero!) do not mean the same thing. One liter is not the same volume as one gallon. For this reason, changing the unit of measure changes the domain.

4.8 Dimensionality

Dimensionality is the name applied to the conceptual part of units of measure. Measures such as meters per second, miles per hour, and furlongs per fortnight are all measures of speed. Measures such as pounds and newtons are measures of force. The members of each set above belong to the same class, or are said to have the same dimensionality.

In the language of mathematics, dimensionality forms an equivalence class on the set of all units of measure. The equivalence between two units of measure is determined by the existence of an invertible transformation of one set of units to the other. This means that two units of measure have the same dimensions if there exists a function that maps values in one unit of measure to values in the other and the inverse of the function maps values in the second units back to values in the first. Recognizing when these functions exist and when they do not is an important aspect of maintaining a registry. The two examples below illustrate this.

EXAMPLE 1: Fahrenheit - Celsius temperature conversions

F° = (9/5) C° + 32 and C° = (5/9) (F°- 32)

The functions defined here are inverse transformations of each other.

EXAMPLE 2: Inches - Centimeters length conversions

In = (2.54)Cm and Cm = In/(2.54)

The functions defined here are inverse transformations of each other.

See *Measurement Units in XML Datatypes*, by F. Olken and J. McCarthy (1999)[5]. Again, see the *Unified Code for Units of Measure*, by G. Schadow and C. McDonald (1999), for related ideas.

It is very important for the registrar and subject matter specialists that manage value domains in a MDR to understand dimensionality. The users of the registry need to know whether data supplied in one set of units can be transformed into another. Units of measure used in conceptually equivalent domains must have the same dimensionality. The next example of two conceptually equivalent non-enumerated value domains will illustrate this:

EXAMPLE 3

Conceptual domain name:	Amounts in monetary units
Conceptual domain definition:	Numbers signifying monetary values (in whole units).
Non-enumerated conceptual domain description:	Money in whole units
Dimensionality:	Monetary units
Value domain name (1):	Amounts in whole US dollars
Value domain definition:	Monetary amount in whole US dollars.
Non-enumerated value domain description (1):	Non-negative integers representing whole dollars.
Unit of Measure:	US dollars
Value domain name (2):	Amounts in whole euros
Value domain definition:	Monetary amount in whole euros.
Non-enumerated value domain description (2):	Non-negative integers representing whole euros.
Unit of Measure:	euros

5) *Measurement Units in XML Datatypes*, by F. Olken and J. McCarthy (1999), is on the Web at the URL http://pueblo.lbl.gov/~olken/mendel/w3c/xml.schema.wg/units/syntax.htm.

There exists a way to translate dollars to euros and back again, and it is evident that these are conceptually equivalent domains. They both represent the same idea, which is "amounts of money". However, the example also illustrates a problem. Money conversion requires a function that changes almost continuously for some applications, e.g., banking, because exchange rates fluctuate so often. Issues of this sort will require much monitoring if they are important to the function of the registry.

The following is a (partial) list of possible problem areas with dimensionality and the difficulty stewards and users may face in converting units:

1) Transforming monetary units (see above)

2) The difference between transforming temperature at a given time and place and transforming temperature intervals

3) The difference between transforming date and time for a specific event and transforming intervals of dates or times

4) Recognizing when some SI and English (or some other system of) units have the same dimensionality

4.9 Classifying value domains

Value domains are classified in one of two main ways, either through associations with the terms in a classification scheme, or through the use of *Representation Class*. Representation class is a special classification scheme represented in the metamodel for a MDR for classifying value domains. The value domain, its attributes, and its associations with data type and unit of measure make up the representation for a data element. So, the representation class is a classification of representations. It is very simple in structure, essentially a list of key words. These words are designations for concepts describing the class of representation. Here, several examples that appeared before will now have a representation class term included:

EXAMPLE 1

Conceptual domain name:	Categorizations of human sexes
Conceptual domain definition:	Enumeration of concepts of human sexes.
Value domain name:	Human sex codes (1)
Value domain definition:	Codes for the human sexes.
Permissible values:	<1, Male>
	<2, Female>
	<3, Unknown>
Representation class term:	Categories

EXAMPLE 2

Conceptual domain name:	Textual descriptions
Conceptual domain definition:	Text describing an activity or thing
Non-enumerated conceptual domain description:	Text of limited length
Value domain name:	Textual English descriptions
Value domain definition:	Textual description of an activity or thing.
Non-enumerated value domain description:	English text up to 60 characters
Representation class term:	Text

EXAMPLE 3

Conceptual domain name:	Volumes
Conceptual domain definition:	Volumes expressed in different units of measure.
Non-enumerated conceptual domain description:	Volume measures
Dimensionality:	Volume
Value domain name:	Volume (in whole liters)
Value domain definition:	Volume amount in whole liters.
Non-enumerated value domain description:	Non-negative integers
Unit of Measure:	Liters
Representation class term:	Numeric with units of measure

5 Registering value domains

5.1 Introduction

This clause contains a description of the detail needed to register a value domain in a MDR. Two examples are given, one for an enumerated value domain and the other for a non-enumerated value domain. The discussion will focus on the unique requirements for registering value domains. In particular,

— Subclause 5.2 explains the rules for registering conceptual domains and value domains

— Subclause 5.3 explains strategies for registering conceptual domains and value domains

— Subclause 5.4 provides detailed examples based on the rules and strategies

5.2 Rules for registering value domains

In general, the following classes in the model are related to value domains and conceptual domains. They are populated when registering value domains and conceptual domains in a MDR -

Conceptual domain

Conceptual domain relationship (if necessary)

Enumerated conceptual domain or Non-enumerated conceptual domain (possibly both)

If enumerated –

Value meanings

If non-enumerated -

Description

Value domain

Value domain relationship (if necessary)

Enumerated value domain or Non-enumerated value domain (possibly both)

If enumerated –

Value

Permissible value (Associate Value with Value meaning)

If non-enumerated -

Description

Unit of measure (if necessary)

Data type

Representation class

A model in UML notation for the value domain – conceptual domain portion of the metamodel for metadata registries is provided in Annex A. The model shows the classes, with their names, attributes, and relationships. The requirement and the cardinality of each attribute and relationship are provided.

The model contains some basic rules for registering value domains and conceptual domains in a MDR. Some of the rules were stated in the preceding paragraphs and sections. Others are discussed here. In addition, some of the rationale for the rules is lost in the modeling notation.

1) A conceptual domain can exist by itself, i.e., it does need an associated value domain for it to exist; but, a value domain must have an associated conceptual domain.

   Some conceptual domains are not linked directly to value domains. So, the conceptual domain stands alone. Example 2 in 4.3.2 illustrates this.

2) Value meanings must be carefully maintained.

   Value meanings are the means to link different values in different enumerated value domains with the same meaning; the means to track changes to the meaning of a value over time; and the means to build correspondence tables between classifications. Similar concepts are often represented across classifications; mapping these concepts to each other is important for data users who need to compare different data classified to similar but different classification schemes.

3) Enumerated and non-enumerated sub-types of value domains are not exclusive. This means some value domains can have an enumerated part and a non-enumerated part. See Examples 1 and 2 in 4.2.11.

   It is easy to think that the sub-types are exclusive, i.e., either one or the other is true, but not both. Almost all examples are of one type or the other. However, special circumstances sometimes necessitate the creation of unusual value domains.

4) Data type is a very important component and must be carefully documented, as described in 4.6.

   Data type is required for successful data transfers. The problem is that there are many mechanisms for specifying a data type. There are programming languages, database query languages, data transfer languages, abstract data definition languages, and other standards. Each one has its own way of describing and naming its allowed data types. Not all of these specifications allow the same kinds of data types. Data types with the same name in different specifications do not always mean the same thing. Not all specifications have the same primitive data types. The ways of generating new data types from the primitive ones differ across specifications.

   ISO/IEC 11179 does not require the user to use one specification for naming or describing data types. It does require the user to provide the name of the specification and the name or derivation of the data type, so the user can determine the meaning of the data type.

5) Units of measure describe data values and help determine comparability between data from different sources. They are a required part of the documentation for any measurement, either from an experiment or through a statistical process. See 4.7 and 4.8.

Units of measure are grouped into classes by dimensionality. Dimensionality is a concept for units of measure and is associated with the conceptual domain. Units of measure used in conceptually equivalent domains must have the same dimensionality.

6) A value meaning does not have to be associated with a permissible value. A permissible value must have an associated value meaning.

7) A conceptual domain is a set of value meanings. Item 6 (above) implies that the set of value meanings does not have to be in one-to-one correspondence with the permissible values for any of the enumerated value domains representing the conceptual domain. In addition, conceptually equivalent enumerated value domains do not have to be elementarily equivalent. Non-elementarily equivalent, but conceptually equivalent, domains might even have the same number of permissible values.

The issue of how many permissible values conceptually equivalent enumerated value domains must share is open to interpretation. Relating a value domain to a conceptual domain is a subject matter expert's decision. Two reasonable people may differ in their choices. Needs of the organization, needs of internal users, needs of outside users, and needs dictated by standards all have an influence.

The following example shows three conceptually equivalent, but non-elementarily equivalent enumerated value domains; two with the same number of permissible values.

EXAMPLE

Conceptual domain name:	Categorizations of school subjects
Conceptual domain definition:	Enumeration of concepts of school subjects.
Value domain name:	School subject codes (1)
Value domain definition:	Codes for subjects studied in school.
Permissible values:	<0, Other>
	<1, Mathematics>
	<2, Language>
	<3, Foreign Language>
	<4, History>
Value domain name:	School subject codes (2)
Value domain definition:	Codes for subjects studied in school.
Permissible values:	<0, Other>
	<1, Mathematics>
	<2, Language>
	<3, Foreign Language>
	<4, European History>
	<5, Pan-American History>
Value domain name:	School subject codes (3)
Value domain definition:	Codes for subjects studied in school.
Permissible values:	<0, Other>
	<1, Mathematics>
	<2, Language>
	<3, Foreign Language>
	<4, African History>
	<5, Asian History>

5.3 Strategies

Strategies and steps to register value domains can be broken into two large categories, those for enumerated value domains and those for non-enumerated value domains. The main strategy is provided in a series of steps. The steps do not have to be followed in the order given. The given steps are not exhaustive.

For any domain, do the following:

1) Understand what values are allowed. This requires a full understanding of the rule defining the allowed values for a non-enumerated value domain or understanding the structure of values and their meanings for an enumerated value domain. It may not be possible to know all the permissible values of an enumerated value domain at first, but understanding the content, meaning, and form of typical examples is necessary.

2) Determine whether a suitable conceptual domain exists in the registry for the value domain. If not, then a new one must be created and entered into the registry. It may be hard to determine which of several existing conceptual domains is the appropriate one for the given value domain. This is best determined through the work of a team, which includes the registrar and appropriate subject matter experts.

3) If an appropriate conceptual domain already exists, then determine whether the value domain itself already exists. Examine all the conceptually equivalent value domains representing the selected conceptual domain. If the given value domain is new, then it must created and entered into the registry.

4) If the value domain is non-enumerated, then the rule, units of measure, and dimensionality need to be determined.

5) If the value domain is enumerated, then the values and the value meanings of each permissible value must be determined.

6) If the conceptual domain existed previously, then the value meanings for that conceptual domain must be checked against each of the value meanings for the new enumerated value domain. Enter any value meanings that did not previously exist. Mark the permissible values where the value meanings already exist in the registry.

7) Enter the permissible values, using links to previously existing value meanings. There is no need to manage the set of values. Values may be repeated in the registry.

8) Determine if the conceptual domain is related to other conceptual domains. This is especially true when registering a set of value domains such as a classification (see 4.5). The classification is often represented as a hierarchy, and a hierarchy of conceptual domains is created to capture the concepts that each level (value domain) represents.

9) Determine if the value domain is related to other value domains. This is especially true when registering a set of value domains such as a classification (see 4.5). The classification is often represented as a hierarchy, and the relationship between the levels is represented as relationships between the value domains.

10) Currently, the metamodel for metadata registries does not handle relationships between permissible values. Relationships do exist in cases where value meanings are the same across value domains, and the relationship is realized by the links to a common value meaning. Relationships also exist in hierarchical code structures such as a classification (see 4.5), and the relationship is realized through the structure of the value, i.e., code, itself.

5.4 Examples

The following sub-clauses each contain an example illustrating the complete registration of a value domain: one for a set of elementarily equivalent domains; and the other for a non-enumerated value domain. Both examples are based on international standards. The attributes listed in the examples are contained and described in ISO/IEC 11179-3.

5.4.1 Enumerated value domain

The first example is for a linked set of enumerated value domains, contained in the standard ISO 3166-1. This section demonstrates how to register enumerated value domains (in this case, code sets) in a MDR. Four code sets contained in ISO 3166-1 are registered: short English name, 2 character alpha code, 3 character alpha code, and 3 character numeric code. Each permissible value in each code set is linked to the corresponding value meaning that defines those values. These links establish a correspondence across each of the value domains on an element-by-element basis. The set of value meanings is registered along with a conceptual domain.

Registration of Conceptual Domain		
1	**Conceptual Domain**	
	Conceptual Domain Context	ISO 3166-1
	Conceptual Domain Name	Countries of the World
	Conceptual Domain Definition	The primary geopolitical entities of the world.
	Conceptual Domain Type	Enumerated
	CD Identifier/Version Number (DI:VI)	(93-273-8065) 1234:1
	Dimensionality	Codes
2	**Value Meanings**	
	Value Meaning ID	10001
	Value Meaning Description	The primary geopolitical entity known as <Democratic Republic of Afghanistan>
	VM Begin Date	19971001
	VM End Date	(Not applicable)
	Value Meaning ID	10002
	Value Meaning Description	The primary geopolitical entity known as <People's Socialist Republic of Albania>
	VM Begin Date	19971001
	VM End Date	(Not applicable)
	. . . .	
	Value Meaning ID	10220
	Value Meaning Description	The primary geopolitical entity known as <Republic of Zimbabwe>
	VM Begin Date	19971001
	VM End Date	(Not applicable)
5	**Other Metadata Attributes**	
	Origin	ISO 3166-1:1997
	Comment	(Not Applicable)
	Submitting organization	Bureau of Labor Statistics

Registration of Conceptual Domain		
	Data Steward	Dan Gillman
6	**Classification**	
	Keyword	Country, World
	Group	Geopolitical Entities, Country Identifiers
	Object	Country
	Layer of abstraction	Conceptual Domain
7	**Quality Control**	
	Registration Status	Standard
	Administrative Status	Final

Registration of Value Domain #1		
3	**Value Domain**	
	Value Domain Context	ISO 3166-1:1997
	Value Domain (VD) Name	Short English-Language Country Names
	Value Domain Name Context	Registry
	Value Domain Definition	Short English-language names of all countries.
	VD Identifier/Version Number (DI:VI)	(93-273-8065) 9876:1
	Conceptual Domain ID for this Value Domain	1234:1
	Value Domain Type	Enumerated
	Data Type	Character string
	Data Type Scheme	C programming language
	Minimum Characters	4
	Maximum Characters	44
	Format	%s
	Unit of Measure	(Not Applicable)
	Precision	(Not Applicable)
	VD Origin (Enumerated)	ISO 3166-1:1997
	VD Explanatory Comment	(Not applicable)
4	**Permissible Values**	
	Values	Afghanistan
	PV Begin Date	19971001

Registration of Value Domain #1		
	PV End Date	(Not applicable)
	Value Meaning ID	10001
	Values	Albania
	PV Begin Date	19971001
	PV End Date	(Not applicable)
	Value Meaning ID	10002
	. . . .	
	Values	Zimbabwe
	PV Begin Date	19971001
	PV End Date	(Not applicable)
	Value Meaning ID	10220
5	Other Metadata Attributes	
	Origin	ISO 3166-1:1997
	Comment	(Not Applicable)
	Submitting organization	Bureau of Labor Statistics
	Data Steward	Dan Gillman
7	Quality Control	
	Registration Status	Standard
	Administrative Status	Final

Registration of Value Domain #2		
3	Value Domain	
	Value Domain Context	ISO 3166-1:1997
	Value Domain (VD) Name	2 Character Alphanumeric Country Codes
	Value Domain Name Context	Registry
	Value Domain Definition	2 Character Alphanumeric Codes for all countries
	VD Identifier/Version Number (DI:VI)	(93-273-8065) 9877:1
	Conceptual Domain ID for this Value Domain	1234:1
	Value Domain Type	Enumerated
	Data Type	Character string of length 2
	Data Type Scheme	C programming language
	Minimum Characters	(Not Applicable)
	Maximum Characters	(Not Applicable)

Registration of Value Domain #2		
	Format	%2c
	Unit of Measure	(Not Applicable)
	Precision	(Not Applicable)
	VD Origin (Enumerated)	ISO 3166-1:1997
	VD Explanatory Comment	(Not applicable)
4	**Permissible Values**	
	Values	AF
	PV Begin Date	19971001
	PV End Date	(Not applicable)
	Value Meaning ID	10001
	Values	AL
	PV Begin Date	19971001
	PV End Date	(Not applicable)
	Value Meaning ID	10002
	. . . .	
	Values	ZW
	PV Begin Date	19971001
	PV End Date	(Not applicable)
	Value Meaning ID	10220
5	**Other Metadata Attributes**	
	Origin	ISO 3166-1:1997
	Comment	(Not Applicable)
	Submitting organization	Bureau of Labor Statistics
	Data Steward	Dan Gillman
7	**Quality Control**	
	Registration Status	Standard
	Administrative Status	Final