TECHNICAL SPECIFICATION

ISO/IEC TS 19763-13

First edition
2016-12-01

Information technology — Metamodel framework for interoperability (MFI) —

Part 13: Metamodel for form design registration

Technologies de l'information — Cadre du métamodèle pour l'interopérabilité (MFI) —

Partie 13: Métamodèle pour l'enregistrement de la conception des formulaires

Reference number
ISO/IEC TS 19763-13:2016(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

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.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

The committee responsible for this document is ISO/IEC JTC 1, *Information Technology*, Subcommittee SC 32, *Data management and interchange*.

A list of all parts in the ISO/IEC 19763 series can be found on the ISO website.

Introduction

There is an increasing demand for systems to interoperate by exchanging data, and for data to be reused outside of the original context of its collection. For data exchange or reuses to be meaningful, the business information requirements that are met by the data stored in these systems must be understood so that suitable data exchange mechanisms can be developed and interpretation of the data is reliable.

Not only does this require a clear understanding of the meaning of the data, it also frequently requires the coordination of data capture. Where data input is manual, the definitive source of data semantics is the design of the data entry form. Indeed if we do not understand the encoding of knowledge in the database schema or we suspect some anomaly in the data captured, we inspect the original form and the context of its use. Furthermore, if we wish to gather interoperable data, it is frequently necessary to harmonize aspects of form design before information systems are developed and data is captured. However, there is no abstract, universal metamodel for form designs that supports the registration and comparison or harmonization of form designs and faithful implementation of these designs in information systems. This is the intent of this document.

The Oxford English dictionary defines a form as “a formulary document with blanks for the insertion of particulars”. Other ISO definitions of a form include ISO 5127, “document (printed or otherwise produced), with pre-designated spaces for the recording of specific information”, and ISO 9241-143, “structured display of fields and other user-interface elements that the user reads, fills in, selects entries for (e.g. through check boxes or radio buttons) or modifies”. While we recognize these definitions, none precisely matches the needs of this document. Thus, we will define a form as a structured collection of spaces, suitable instructions and rules that support the collection of specific information that may be subsequently compared and processed in a routine fashion. A form design is thus a description of a particular form such that it may be rendered in any suitable information system, and the metamodel for registration of form designs contained within this document describes the attributes that are necessary to represent the semantics and syntax of form designs.

Given a standard metamodel for the registration of form designs, ISO/IEC 19763 Metamodel framework for interoperability (MFI) and ISO/IEC 11179 Metamodel for metadata registries provide important facilities for the creation and annotation of form designs. ISO/IEC 19763 supports the registration of form designs and section elements as models and model elements, provides facilities to record associations between the components of two or more form design, particularly derivation, specialization, extension and reuse, and allows the association of form designs with the data models that are used to store data captured by their instances. ISO/IEC 11179 provides classes and types that support the identification, naming, registration and administration of form designs and supporting documents, and provides a model either for an associated, standardized question bank or a rich source of question-level metadata attributes with which to explain the meaning of individual data items. When used together, the International Standards can support the rapid design and reuse of form designs, wrap and hide the complexity of semantic annotation from subject matter experts, and provide a ready reference of associations and transformations for users seeking to collect and use interoperable data.

This document does not supplant or replace computer languages such as XForms, Windows Forms, Adobe Forms or relevant parts of HTML, which describe how a form design is implemented, and is deliberately devoid of domain or content specific semantics to ensure wide applicability. However, given the universal applicability of forms, it should be of no surprise that elements of the model can be recognized in many forms standards. Some of these have been mapped to this document in Annex A to Annex E.

Forms may be printed on paper, or encoded in electronic format. Electronic forms may be rendered natively in standard formats such as HTML, XForms or PDF, or propriety ones such as Windows forms, Cocoa or Java Swing. They may also be implemented in a common survey framework such as Survey Monkey or Lime Survey. Despite this diversity, it is eminently possible to create forms in different formats that support the same comparisons and downstream processing *provided the spaces and instructions share the same semantic intent*. Such a collection of forms could be said to share the same *design*. A model that is adequate to record these *form designs* is the subject of this document.

Information technology — Metamodel framework for interoperability (MFI) —

Part 13: Metamodel for form design registration

1 Scope

The primary purpose of the ISO/IEC 19763 series is to specify a metamodel framework for interoperability. This document specifies a metamodel for registering form designs.

This document provides a metamodel to describe the structure and semantics of an implemented form devoid of any specific, domain semantics, e.g. in healthcare, social science, e-government and e-business, or representation format so that data may be faithfully exchanged between systems and system components, and associations expressed between sets of form designs whose data may be compared, joined or composed for analysis.

2 Normative references

There are no normative references in this document.

3 Terms, definitions and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 19763-1, ISO/IEC 19763-10, ISO/IEC 11179-3 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

— IEC Electropedia: available at http://www.electropedia.org/

— ISO Online browsing platform: available at http://www.iso.org/obp

3.1.1
attachment
digital object that is required as a *response* (3.1.15) to a *question* (3.1.14) on a *form* (3.1.9)

Note 1 to entry: Used to indicate that the response to a question includes a file on an accessible file-system that will be loaded when the form transaction is complete.

3.1.2
combinator
operator that joins two *constraints* (3.1.6) (to make a binary constraint) returning a result based upon both

EXAMPLE Conjunction, disjunction, implication.

3.1.3
compliance rule
<form template> specification for some aspect of a *form design* (3.1.10) that shall be satisfied for that design to be a correct implementation of a *form template* (3.1.11)

3.1.4
completed form
form (3.1.9) for which *responses* (3.1.15) have been completed as required according to its *instructions* (3.1.12) and *rules* (3.1.16)

3.1.5
consequence
expression (3.1.7) that sets or specifies some property of an element of a *form design* (3.1.10) when its related *constraint* (3.1.6) evaluates to true

3.1.6
constraint
<form registration> *expression* (3.1.7) about *form design* (3.1.10) elements that evaluates to a Boolean value

3.1.7
expression
statement that evaluates to a string or numeric value

3.1.8
field
space on a *form* (3.1.9) for the recording of a *response* (3.1.15)

3.1.9
form
document or human interface comprising a structured collection of *fields* (3.1.8), suitable *instructions* (3.1.12) and *rules* (3.1.16) that support the collection of specific information that may be subsequently compared and processed in a routine fashion

3.1.10
form design
specification for the creation of equivalent *forms* (3.1.9) in different languages, applications and media

3.1.11
form template
partial *form design* (3.1.10) that establishes a pattern for the creation of other form designs

Note 1 to entry: A form template will often have empty or incomplete form sections with instructions describing what kind of questions are required to create a completed design.

3.1.12
instruction
sentence that directs a person in some aspect of the completion or submission of a *form* (3.1.9)

3.1.13
owl:sameAs
property of the Web Ontology Language that indicates that individuals in an OWL DL ontology refer to the same thing, or in OWL Full to additionally indicate that two classes are equal

Note 1 to entry: See http://www.w3.org/TR/owl-ref/#sameAs-def.

3.1.14
question
sentence worded or expressed so as to elicit information from a person

3.1.15
response
information elicited from a person by a *question* (3.1.14)

3.1.16
rule
principle guiding the behaviour of some aspect of a *form* (3.1.9)

3.1.17
section
subcomponent of a *form* (3.1.9) whose contained *questions* (3.1.14), *instructions* (3.1.12) and *rules* (3.1.16) share a common purpose, meaning or context

3.1.18
skos:related
semantic relation asserting that the object of the labelled relationship is related to the subject

Note 1 to entry: See http://www.w3.org/TR/skos-reference.

3.2 Abbreviated terms

MFI Core and mapping

ISO/IEC 19763-10, Information technology — Metamodel framework for interoperability (MFI) — Part 10: Core model and basic mapping

MDR Metamodel

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

MFI Form design registration

Information technology — Metamodel framework for interoperability (MFI) — Part 13: Metamodel for form design registration

4 Conformance

4.1 General

An implementation claiming conformance with this document shall support the metamodel specified in Clause 5, depending on a degree of conformance as described below.

4.2 Degrees of conformance

4.2.1 General

The distinction between “strictly conforming” and “conforming” implementations is necessary to address the simultaneous needs for interoperability and extensions. This document describes specifications that promote interoperability. Extensions are motivated by needs of users, vendors, institutions and industries, but are not specified by this document.

A strictly conforming implementation may be limited in usefulness but is maximally interoperable with respect to this document. A conforming implementation may be more useful, but may be less interoperable with respect to this document.

4.2.2 Strictly conforming implementation

A strictly conforming implementation

a) shall support the metamodel specified in Clause 5, and

b) shall not support any extensions to the metamodel specified in Clause 5.

4.2.3 Conforming implementation

A conforming implementation

a) shall support the metamodel specified in Clause 5, and

b) may support extensions to the metamodel specified in Clause 5 that are consistent with the metamodel and the MDR mapping package in Clause 5.

4.2.4 Implementation Conformance Statement (ICS)

An implementation claiming conformance with this document shall include an Implementation Conformance Statement stating

a) whether it is a strictly conforming implementation or a conforming implementation (see 4.2.2), and

b) what extensions are supported if it is a conforming implementation (see 4.2.3).

Conformance statements for systems that implement this document shall additionally describe the languages used to convey Rules, and the relationship types available for the Mapping_Relation class.

5 Structure of MFI form design registration

5.1 Overview of MFI form design registration

Figure 1 shows the metamodel for the registration of form designs.

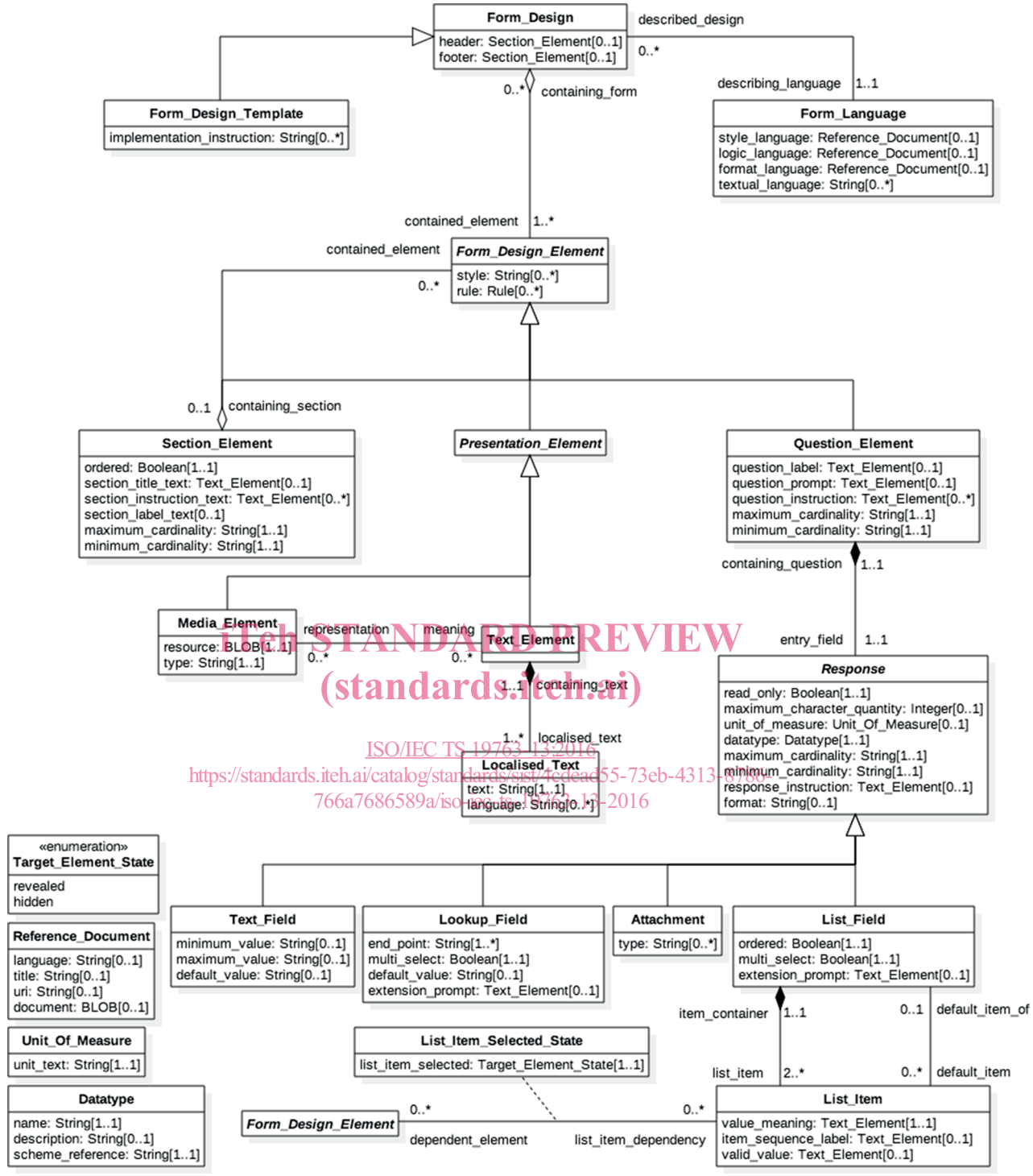

Figure 1 — Form design metamodel

Forms have questions and sections that are constrained or unavailable for completion dependent upon the answers given to earlier questions. Figure 2 is a model for the rule language used to describe such dependencies between form elements: textual expressions in this language are used to complete the *rule* attribute of the **Form_Design_Element** class.

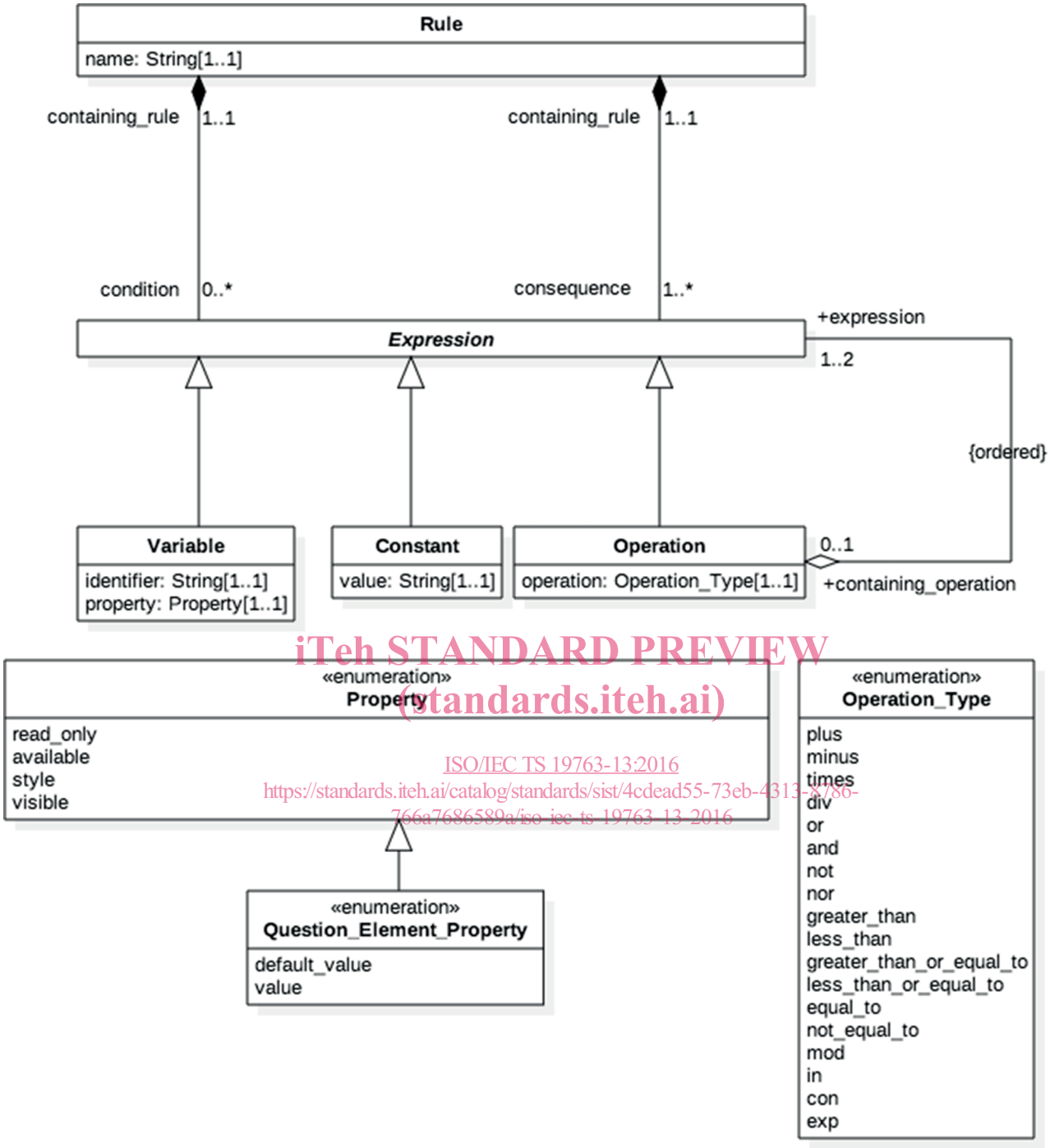

Figure 2 — Rule

The metamodel for information model registration comprises the following metaclasses:

Attachment_Field
Constant
Datatype
Expression
Form_Design
Form_Design_Element
Form_Design_Template
Form_Design_Language
List_Field
List_Item
List_Item_Selected_State
Localised_Text
Lookup_Field
Media_Element
Operation
Presentation_Element
Question_Element
Response
Reference_Document
Rule
Section_Element
Text_Element
Text_Field
Variable
Unit_of_Measure

The purpose and use of the metamodel is described in detail in Annex A. Detailed specifications of the metaclasses are provided in Annex B.

5.2 Relationship of metaclasses to the MDR Metamodel

As explained in ISO/IEC 19763-10, instances of the metaclasses defined in this subclause may be extended by the types defined in the MDR Metamodel as follows.

— **Form_Design** may be extended as an **Identified_Item**, **Designatable_Item**, **Registered_Item**, **Administered_Item** and **Classifiable_Item**.

— **Form_Design_Element** may be extended as an **Identified_Item**, **Designatable_Item** and **Classifiable_Item**.

— Any instance of a **Form_Design_Element** may be mapped to an instance of a **Concept**.

— Any instance of a **Question_Element** may be mapped to an instance of a **Data_Element**.

— **List_Item** may be extended as an **Identified_Item**; any instance of which may be mapped to a **Concept** and/or **Permissible_Value**.

— **Rule** may be extended as an **Identified_Item** and **Designatable_Item**.

5.3 Details provided in each metaclass definition

For each metaclass, the following details are shown:

— a definition that describes the role or significance of instances of the metaclass;

— the name of its immediate supertype;

— any alternative names (synonyms or aliases) for the metaclass;

— a list of attributes;

— a list of references.

For each attribute, the following details are shown:

a) the name of the attribute; where the attribute is one that is provided by the type defined in the MDR metamodel by which when instances of the metaclass are extended, the name is italicized;

b) the datatype for values of the attribute;

c) the multiplicity of the attribute;

d) a description that describes the role or significance of values of the attribute.

For each reference, the following details are shown:

— the name of the reference; this is the role name that describes the role played by the referenced metaclass with respect to the association identified by this reference;

— the name of the referenced metaclass;

— the multiplicity of the reference;

— a description that describes the role or significance of the instance, or instances, of the referenced metaclass with respect to an instance of this metaclass;

— the name of the reference in the referenced metaclass that provides the inverse definition for the association;

— an indication as to whether this metaclass is responsible for the maintenance of the association, i.e. the precedence of the metaclass with respect to the association.

5.4 Basic Types and Enumerations in MFI form design registration

5.4.1 General

Basic Types specify common datatypes for use in the metaclasses. A datatype is a set of distinct values, characterized by properties of those values and by operations on those values (see ISO/IEC 11404). The datatypes used in the specification of the **metaclasses** (see 5.5) are restricted to Boolean, Integer, Date, Value, Sign, Postal_Address, String, Natural_Range, Datetime, String, Notation and Phone_Number (MDR Metamodel 6.2.1 Overview of Basic Types). The types used in the metaclasses are based on this core set of types, with a single addition of the type Binary Large Object (BLOB), and any compliant implementation of a metadata registry should include an implementation of the semantics specified in these core types.

NOTE These datatypes are used in specification of the metaclass attributes themselves, and are not intended to constrain the datatypes that may be used in specifying Response datatypes.

Enumerations specify the list of value for use with metaclass attributes.

For each enumeration, the following details are shown:

— the name of the referenced enumeration;

— a description of the enumeration;

— the datatype of the values in the enumeration;

— the name of each value in the enumeration;

— a description of the semantics of each enumeration value;

— the name of the metaclass where this enumeration is used;

— the name of the attribute where this enumeration is used.

5.4.2 Property

Property is an enumeration of values listing properties of a **Presentation_Element**, **Section_Element**, **Question_Element** or a **List_Item** that may be addressed by a **Rule** (see Figure 2).

Datatype

String

Value	Description
read_only	Indicates that the **Form_Design_Element** read_only property is to be tested or set as part of an **Expression** in a **Rule**
available	Indicates that the **Form_Design_Element** available property is to be tested or set as part of an **Expression** in a **Rule** The state of the available property may also be set by a **List_Item** that has a dependent_element association with the respective **Form_Design_Element**.
style	Indicates that a **Form_Design_Element** style property is to be tested or set as part of an **Expression** in a **Rule**
visible	Indicates that a **Form_Design_Element's** visibility to the user interacting with the form is to be tested or set as part of an **Expression** in a **Rule**

5.4.3 Question_Element_Property

Question_Element_Property is an enumeration of values listing additional properties of a **Question_Element** that may be addressed in a **Rule** (see Figure 2).

Supertype

Property

Datatype

String

Value	Description
default_value	Indicates that the **Question_Element** default_value property is to be tested or set as part of an **Expression** in a **Rule**
value	Indicates that the **Question_Element** value property is to be tested or set as part of an **Expression** in a **Rule**

5.4.4 Target_Element_State

Target_Element_State is an enumeration of values listing the possible states that a dependent **Form_Design_Element** may take when a **List_Item** is selected.

Datatype

String