Constraints for ReqIF 1.0

Constraints for validating ReqIF 1.0 models

C1.1 EnumValue - existence of long name
[C1.1] LongName is not set – LongName of EnumValue must be set

C1.2 AttributeDefinitionBoolean - existence of long name
[C1.2] LongName is not set – LongName of AttributeDefinitionBoolean must be set

C1.3 AttributeDefinitionDate - existence of long name
[C1.3] LongName is not set – LongName of AttributeDefinitionDate must be set

C1.4 AttributeDefinitionEnumeration - existence of long name
[C1.4] LongName is not set – LongName of AttributeDefinitionEnumeration must be set

C1.5 AttributeDefinitionInteger - existence of long name
[C1.5] LongName is not set – LongName of AttributeDefinitionInteger must be set

C1.6 AttributeDefinitionReal - existence of long name
[C1.6] LongName is not set – LongName of AttributeDefinitionReal must be set

C1.7 AttributeDefinitionString - existence of long name
[C1.7] LongName is not set – LongName of AttributeDefinitionString must be set

C1.8 AttributeDefinitionXHTML - existence of long name
[C1.8] LongName is not set – LongName of AttributeDefinitionXHTML must be set

C1.9 RelationGroup - existence of long name
[C1.9] LongName is not set – LongName of RelationGroup must be set

C1.10 DatatypeDefinition - existence of long name
[C1.10] LongName of {0} is not set – LongName of DatatypeDefinition must be set

C2.0 AttributeDefinition - existence of DatatypeDefinitiion
[C2.3] DatatypeDefinition is not set – DatatypeDefinition of AttributeDefinition must be set

C2.1 AttributeDefinitionBoolean - existence of DatatypeDefinitionBoolean
[C2.1] DatatypeDefinition of wrong type – DatatypeDefinition of AttributeDefinition must have correct type

C2.6 AttributeDefinitionReal - existence of DatatypeDefinitionReal
[C2.6] DatatypeDefinition of wrong type – DatatypeDefinition of AttributeDefinition must have correct type

C3.1 DatatypeDefinition - minValue is set
[C3.1] minValue of {0} is not set – minValue of DatatypeDefinitionReal or DatatypeDefinitionInteger is set.

C3.2 DatatypeDefinition - maxValue is set
[C3.2] maxValue of {0} is not set – maxValue of DatatypeDefinitionReal or DatatypeDefinitionInteger is set.

C3.3 DatatypeDefinition - Accuracy is set
[C3.3] accuracy of {0} is not set – Accuracy of DatatypeDefinitionReal is set.

C3.4 DatatypeDefinition - maxLength is set
[C3.4] maxLength of {0} is not set – maxLength of DatatypeDefinitionString is set.

C3.5 DatatypeDefinitionReal - minValue<=maxValue
[C3.5] DatatypeDefinitionReal minValue is greater than maxValue – minValue of DatatypeDefinitionReal should not be greater than maxValue

C3.6 DatatypeDefinitionInteger - minValue<=maxValue
[C3.6] DatatypeDefinitionInteger minValue is greater than maxValue – minValue of DatatypeDefinitionInteger should not be greater than maxValue

AV1.1 AttributeValueReal - between min and max
[AV1.1] AttributeValueReal {0} than {1} – theValue of AttributeValueReal must be between minValue and maxValue

AV1.2 AttributeValueInteger - between min and max
[AV1.2] AttributeValueInteger {0} than {1} – theValue of AttributeValueInteger must be between minValue and maxValue

AV1.3 AttributeValueString - not longer than maxLength
[AV1.3] AttributeValueString is longer than maxLength – theValue of AttributeString must not be longer than maxLength

AV1.4 AttributeValueEnumeration Multiplicity - max one value if not multiValued
[AV1.4] AttributeValueEnumeration is single valued but contains {0} values – AttributeValueEnumeration must not contain more than one Value if its not multiValued

AV1.5 AttributeValueReal - not more precise that accuracy
[AV1.5] AttributeValueReal {0} has {1} fracional digits but accuracy of DatatypeDefinition is {2} –

AttributeValueReal must not have more fractional digits than accuracy

AV1.6 AttributeValueXHTML - referenced objects exist
[AV1.6] AttributeValueXHTML references {0} that can not be found: {1} – All files referenced in an AttributeValueXHTML by a file URI exist

AV1.6a AttributeValueXHTML - referenced objects is file
[AV1.6a] AttributeValueXHTML references {0} non-file object(s) {1} – object should be referenced by a file URI

AV1.7 AttributeValueEnumeration uses valid EnumValue
[AV1.7] AttributeValueEnumeration uses an illegal EnumValue – An AttributeValueEnumeration only references values that are defined by its DatatypeDefinitionEnumeration

C6.1 The XHTML class attribute must not be used
[C6.1] AttributeValueXHTML contains {0} with a class attribute: {1} – The 'class' attribute of the XHTML Core Attribute Collection MUST NOT be used

C6.2 Alternative images and texts must be exported
[C6.2] AttributeValueXHTML missing alternative for data object – For XHTML object elements that refer to an external object that is not an image with the MIME-Type image/png, an alternative image AND an alternative text MUST be provided

C6.3 Only data, type, width and height are allowed as attributes for XHTML <object>
[C6.3] AttributeValueXHTML contains {0} with illegal attributes: {1} – If there is a specific MIME-type for the application that handles the external object, it MUST be stored in the 'type' attribute and no other attributes than 'data', 'type', 'width' and 'height' MUST be used in that case.

C6.4 The location of an external object MUST be specified via the data attribute
[C6.4] AttributeValueXHTML missing data attribute in {0} – The location of an external object MUST be specified via the 'data' attribute.

SCHEMA - reqif complies xsd schema
[XSD] {0} – The input file must comply the xsd schema

EMFSCHEMA - reqif complies xsd schema
{0} – the Reqif must comply the emf-schema

IG_R3 Name of Attributes should always be represented without a ReqIF. prefix
[IG_R3] Name of Attribute {0} has a ReqIF. prefix – According to the Implementaion Guide"Names of user defined attributes should always be represented in ReqIF XML documents and requirements authoring tools without a ReqIF. prefix." [2.3.2 R3] Note: this constraint checks for the prefix "ReqIF." case sensitive

IG_2_10 Conflicting IsEditable for AttributeDefinitionEnumeration
[IG_2_10] DatatypeDefinitionEnumeration {0} is referenced by different AttributeDefinitionEnumeration of which some are editable and some are not editable – Generates a Warning if a DatatypeDefinitionEnumeration is used by more than one AttributeDefinition and one of the is set to isEditable while another one is set to Non-Editable. See Implementaion Guide Chapter 2.10 How to resolve conflicting isEditable values for AttributeDefinitionEnumeration: [...]Concerning AttributeDefinitionEnumeration, the following conflict may occur as a consequence: one AttributeDefinitionEnumeration element (A) is editable a different AttributeDefinitionEnumeration element (B) is not editable both AttributeDefinitionEnumeration elements refer to the same DatatypeDefinitionEnumeration, which defines the enumeration literals. If that conflict occurs, it may not be clear whether the first AttributeDefinitionEnumeration element (A) should be treated as editable or not, as changes to the list of enumeration literals will be visible in the second AttributeDefinitionEnumeration element (B).