



Technical Specification

ISO/TS 10303-15

Industrial automation systems and integration — Product data representation and exchange —

Part 15: Description methods: SysML XMI to XSD transformation

*Systèmes d'automatisation industrielle et intégration —
Représentation et échange de données de produits —*

*Partie 15: Méthodes de description: Transformation de SysML
XMI en XSD*

**Second edition
2024-07**

iTeh Standards
(<https://standards.iteh.ai>)
Document Preview

[ISO/TS 10303-15:2024](https://standards.iteh.ai/catalog/standards/iso/cbdee33c-f0d7-4bbb-b834-f75baf794e7f/iso-ts-10303-15-2024)

<https://standards.iteh.ai/catalog/standards/iso/cbdee33c-f0d7-4bbb-b834-f75baf794e7f/iso-ts-10303-15-2024>



COPYRIGHT PROTECTED DOCUMENT

© ISO 2024

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
CP 401 • Ch. de Blandonnet 8
CH-1214 Vernier, Geneva
Phone: +41 22 749 01 11
Email: copyright@iso.org
Website: www.iso.org

Published in Switzerland

Contents

	Page
Foreword	v
Introduction	vi
1 Scope	1
2 Normative references	1
3 Terms, definitions, and abbreviated terms	2
3.1 Terms and definitions.....	2
3.1.1 Terms related to SysML constructs.....	2
3.1.2 Terms related to XSD constructs.....	4
3.1.3 Terms related to Schematron rules.....	5
3.2 Abbreviated terms.....	5
4 Structure and components of the XSD	6
4.1 General.....	6
4.2 Presentation conventions.....	6
4.3 Main components of the XSD.....	7
4.4 XSD header.....	7
4.5 Common structural XML element definitions.....	7
4.6 ExternalRefBaseObject.....	9
4.6.1 General.....	9
4.6.2 SubObject and NamedSubObject.....	11
4.7 DataContainer.....	13
4.8 List of application object specifications.....	13
4.8.1 List and definition of the entities as application object specifications.....	13
4.8.2 Object attribute specifications.....	13
4.8.3 Attributes optionality and cardinality.....	14
4.8.4 Base root objects.....	15
4.8.5 Base objects.....	15
4.8.6 Instantiation of a subtype.....	16
4.8.7 Representation of XML identification attribute.....	16
4.8.8 Multilanguage support.....	17
4.8.9 Representation of Date and Time.....	17
4.9 Groups and simple types corresponding to Select Data Types and Enumerations.....	18
4.9.1 Group.....	18
4.9.2 Enumeration.....	18
4.9.3 Simple type.....	18
5 SysML XMI to XSD	18
5.1 General.....	18
5.2 Presentation conventions.....	19
5.3 Common mapping conventions.....	20
5.3.1 Reference to external Canonical XMI (CXMI) files.....	20
5.3.2 Xmi:id, xmi:uuid, and UUID.....	20
5.3.3 Assumed sysml:Block in fragments.....	20
5.3.4 Containment and reference relationships.....	21
5.3.5 Used stereotypes.....	21
5.3.6 Select Data Type and supertype.....	21
5.4 Mapping of the DataContainer.....	21
5.5 Mapping of Entity.....	22
5.5.1 General entity.....	22
5.5.2 Entity attribute ordering.....	23
5.5.3 Mapping of abstract entity.....	23
5.5.4 Mapping of entity with one supertype.....	23
5.5.5 Mapping of entity with more than one supertype.....	24
5.5.6 Mapping of entity with the Generic stereotype.....	26
5.5.7 Mapping of entity with the Enrichment stereotype.....	26

ISO/TS 10303-15:2024(en)

5.5.8	Mapping of entity without supertype and not used by containment.....	27
5.5.9	Mapping of entity without supertype and used by containment.....	27
5.6	Mapping of simple type.....	28
5.7	Mapping of aggregation type.....	31
5.8	Mapping of aggregation of aggregation type.....	32
5.9	Select Data Type.....	33
5.9.1	Mapping of Select Data Type.....	33
5.9.2	Proxy artefact.....	37
5.10	Mapping of enumeration type.....	37
5.11	Mapping of entity attribute.....	38
5.11.1	General.....	38
5.11.2	Mapping of multiplicity and optionality.....	39
5.11.3	Attribute typed as an Entity.....	41
5.11.4	Attribute typed as Select Data Type.....	42
5.11.5	Attribute typed as Enumeration type.....	43
5.11.6	Attribute typed as simple type.....	44
5.11.7	Fixed Value attribute.....	44
5.11.8	Inverse Generic attribute.....	45
5.11.9	Inverse Composite Aggregation attribute.....	46
5.11.10	Redefined attributes.....	48
6	SysML XMI to Schematron.....	48
6.1	General.....	48
6.2	Type of referenced subtype entity.....	49
6.3	Fixed Value attribute.....	50
6.4	Inverse Generic attribute.....	51
6.5	Redefinition.....	52
6.6	Constraint as formal proposition.....	54
6.7	ExternalRefBaseObject.....	55
	Annex A (normative) Information object registration.....	57
	Annex B (informative) SysML XMI to XSD transformation — Illustrative diagrams and files.....	58
	Annex C (informative) EXPRESS / Information modelling constructs and the equivalent SysML modelling constructs.....	61
	Bibliography.....	72

Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

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).

ISO draws attention to the possibility that the implementation of this document may involve the use of (a) patent(s). ISO takes no position concerning the evidence, validity or applicability of any claimed patent rights in respect thereof. As of the date of publication of this document, ISO had not received notice of (a) patent(s) which may be required to implement this document. However, implementers are cautioned that this may not represent the latest information, which may be obtained from the patent database available at www.iso.org/patents. ISO shall not be held responsible for identifying any or all such patent rights.

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

For an explanation of the voluntary nature of standards, 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 www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/TC 184, *Automation systems and integration*, Subcommittee SC 4, *Industrial data*.

This second edition cancels and replaces the first edition (ISO/TS 10303-15:2021), which has been technically revised.

The main changes are as follows:

- replaced XSD reference verification method (key-keyref) with schematron rules (updated scope, terms, definitions, added [Clause 6](#), removed [4.7](#));
- inclusion of external references (added [4.6](#));
- inclusion of generic and enrichment multiple inheritance (added [5.5.6](#) and [5.5.7](#));
- removed common definitions import (removed namespace 'cmn' from all code fragments, removed [4.5](#), replaced [Annex B](#));
- inclusion of transformation illustration (replaced [Annex B](#)).

A list of all parts in the ISO 10303 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

ISO 10303 series of International Standards describe the computer-interpretable representation of product information for the exchange of product data. The objective is to provide a neutral mechanism capable of describing product data throughout the life cycle of a product and independent from any particular system. The nature of this description makes it suitable not only for neutral file exchange, but also as a basis for implementing and sharing product databases and archiving.

This document is part of the description methods series. It specifies a mapping of SysML XMI to the XSD. It supports the STEP (ISO 10303) Extended Architecture, [\[17\]](#)[\[18\]](#)[\[19\]](#) including the related information modelling presented in [Annex C](#). This document specifies the transformation from a STEP data model in SysML XMI to a STEP XSD and a STEP Schematron rule file.

The Object Management Group (OMG) has standardized the XML Metadata Interchange (XMI) specification that integrates the OMG Systems Modeling Language (SysML), the OMG Unified Modeling Language (UML), and the World Wide Web Consortium (W3C) Extensible Markup Language (XML). In addition, XMI and SysML integrates data model constraints, specified in OMG Object Constraint Language (OCL) and in Schematron.

SysML inherits the XMI interchange capability from UML. XMI is a mechanism for the interchange of metadata between UML-based modeling tools. OMG has also standardized an XMI compliant interchange format for the SysML thus specifying a lexical representation of SysML models based on a standardized metamodel of the SysML. In addition, XMI format represents SysML model structural constraints, OCL constraints, and Schematron rules.

The W3C has standardized the XML Schema Definition (XSD). XSD defines shared vocabularies and allows machines to carry out rules made by developers. They provide a means for defining the structure, content, and semantics of XML documents. ISO/IEC 19757 specifies the Schematron language, an XML rule-based validation language, that enables the validation of rules applied to XML documents, that the XSD language does not support.

This document specifies a description method of ISO 10303, which defines the transformation of SysML constructs to the XSD constructs. In addition, this document specifies the transformation of SysML constructs to Schematron constructs for the purpose of representing the SysML model constraints represented in XMI as Schematron rules.

Because the XMI standard (ISO/IEC 19509) specifies the XML representation of SysML metamodel constructs, standardizing the binding of SysML constructs into XSD and Schematron constructs supports the representation of SysML models as XML schemas and Schematron rules.

The specified mapping is a one-way transformation from SysML information model represented in XMI into an XML schema and Schematron rules. These limitations make the mapping unsuitable for the transformation of arbitrary SysML models to XSD.

A detailed knowledge of the W3C XML and XSD languages, and the OMG Systems Modelling language is useful.

The main components of this document are:

- the structure, conventions, and concepts of the XSD;
- the specification of the transformation from SysML XMI to XSD for each STEP element modelled in SysML;
- the specification of the transformation from SysML XMI to Schematron for each STEP constraint modelled in SysML.

Industrial automation systems and integration — Product data representation and exchange —

Part 15:

Description methods: SysML XMI to XSD transformation

1 Scope

This document specifies the transformation of SysML (ISO/IEC 19514) constructs to XSD constructs and Schematron (ISO/IEC 19757) constructs for the purpose of representing the SysML model represented in XMI (ISO/IEC 19509) as XML^[13] schemas and Schematron rules.

The specified mapping is a one-way transformation from SysML information model represented in XMI into an XML schema and Schematron rules. These limitations make the mapping unsuitable for the transformation of arbitrary SysML models.

The following are within the scope of this document:

- the specification of the structure, components, and conventions of the XSD for the ISO 10303 series XML implementation method;
- the transformation of SysML metamodel constructs represented in XMI to XSD constructs for the purpose of representing SysML information models as XML schemas;
- the transformation of SysML constructs to Schematron constructs for the purpose of representing the SysML model constraints represented in XMI as Schematron rules.

The following are outside the scope of this document:

- the transformation of SysML metamodel constructs into XSD constructs that are not used in the STEP Extended Architecture;
- the transformation of SysML metamodel constructs into XSD constructs for other purposes than representing SysML constructs as STEP concepts;
- codes and scripts to transform SysML XMI to XSD schema and to Schematron rules;
- process, codes, and scripts to apply Schematron rules on XML documents with a resulting human-readable report.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements 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 10303-2, *Industrial automation systems and integration — Product data representation and exchange — Part 2: Vocabulary*

3 Terms, definitions, and abbreviated terms

3.1 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 10303-2 and the following apply.

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

- ISO Online browsing platform: available at <https://www.iso.org/obp>
- IEC Electropedia: available at <https://www.electropedia.org/>

3.1.1 Terms related to SysML constructs

3.1.1.1

Canonical XMI CXMI

specific constrained format of XMI that minimizes variability and provides more predictable identification and ordering

Note 1 to entry: A Canonical XMI file is itself a valid XMI file.

Note 2 to entry: The full definition is provided in ISO/IEC 19509:2014, Annex B.

3.1.1.2

association

classification of a set of tuples representing links between typed model elements

Note 1 to entry: The full definition is provided in ISO/IEC 19505-1:2012, 11.5.

3.1.1.3

auxiliary

stereotype applied to an abstract block that has no properties

Note 1 to entry: The full definition is provided in ISO/IEC 19505-1:2012, Clause 22.

3.1.1.4

Block

modular construct used for defining an entity

Note 1 to entry: A Block is used for defining Application activity model concepts, Application Data Planning objects, Application Domain Model Business Objects, Core model objects and ARM in SysML Entities. They can include: reference, part, and value properties; constraints. They can be specializations of other Blocks.

Note 2 to entry: The full definition is provided in OMG Systems Modeling Language ISO/IEC 19514:2017, Clause 8.

3.1.1.5

composite aggregation

responsibility for the existence of a composed object

Note 1 to entry: If a composite object is deleted, all of its part instances that are objects are deleted with it.

Note 2 to entry: The full definition is provided in ISO/IEC 19505-1:2012, 11.4.1.

3.1.1.6

directed association

association between a collection of source model elements and a collection of target model elements that is said to be directed from the source elements to the target elements

Note 1 to entry: the full definition is provided in ISO/IEC 19505-1:2012, 7.2.3.3.

**3.1.1.7
enumeration**

Value Type whose values are enumerated

Note 1 to entry: the full definition is provided in ISO/IEC 19505-1:2012, 10.2.3.3.

**3.1.1.8
enumeration literal**

named value for an enumeration

Note 1 to entry: The full definition is provided in ISO/IEC 19505-1:2012, 10.2.3.3.

**3.1.1.9
data type**

type whose instances are identified only by their value

Note 1 to entry: The full definition is provided in ISO/IEC 19505-1:2012, 10.2.3.1.

**3.1.1.10
generalization**

directed relationship between a more general supertype and a more specific subtype

Note 1 to entry: Each generalization relates a specific Classifier to a more general Classifier. Given a Classifier, the transitive closure of its general Classifiers is often called its generalizations, and the transitive closure of its specific Classifiers is called its specializations. The immediate generalizations are also called the Classifier's subtype, and where the Classifier is a Class, its supertype.

Note 2 to entry: The full definition is provided in ISO/IEC 19505-1:2012, C.1.1.

**3.1.1.11
primitive type**

definition of a predefined DataType, without any substructure

Note 1 to entry: The full definition is provided in ISO/IEC 19505-1:2012, Clause 21.

**3.1.1.12
part property**

property that specifies a part with strong ownership and coincidental lifetime of its containing Block

Note 1 to entry: It describes a local usage or a role of the typing Block in the context of the containing Block. Every Part Property has Composite Aggregation and is typed by a Block.

Note 2 to entry: The full definition is provided in ISO/IEC 19514:2017, 8.3.2.3.

**3.1.1.13
reference property**

property that specifies a reference of its containing Block to another Block

Note 1 to entry: The full definition is provided in ISO/IEC 19514:2017, 8.3.2.3.

**3.1.1.14
stereotype**

limited kind of metaclass that cannot be used by itself but must always be used in conjunction with one of the metaclasses it extends

Note 1 to entry: The full definition is provided in ISO/IEC 19505-1:2012, 12.3.3.4.

**3.1.1.15
value property**

property of a block that is typed with a ValueType

Note 1 to entry: The full definition is provided in ISO/IEC 19514:2017, 8.3.2.3.

3.1.1.16

Value Type

stereotype of UML Data Type that is used to define types of values that may be used to express information but cannot be identified as the target of any reference

Note 1 to entry: The full definition is provided in ISO/IEC 19514:2017, 8.3.2.14.

3.1.2 Terms related to XSD constructs

3.1.2.1

XML Schema Definition

XSD

schema with a set of components such as type definitions and element declarations specified by the XML Schema Definition Language

Note 1 to entry: The purpose of an XSD is to define and describe a class of XML documents by using schema components to constrain and document the meaning, usage, and relationships of their constituent parts: DataTypes, elements and their content and attributes and their values.

[SOURCE: World Wide Web Consortium's XML Schema Definition Language (W3C XSD)]^[14]

3.1.2.2

Schema Definition Language

language for XML schemas

[SOURCE: World Wide Web Consortium's XML Schema Definition Language (W3C XSD)]^[14]

3.1.2.3

complex type

<XSD> set of attribute definitions and content type for an element in an XML schema

Note 1 to entry: A `xsd:complexType` provides the definition for an XML Element. It specifies which element and attributes are permitted and the rules regarding where they can appear and how many times they can appear. They can be used in-place within an element definition or named and defined globally.

Note 2 to entry: The full definition is provided in World Wide Web Consortium's XML Schema Definition Language. ^[14]

3.1.2.4

attribute type

<XSD> name, type, and occurrence for a property in an XML schema

Note 1 to entry: An attribute provides extra information within an element. Attributes have name and type properties. An Attribute can appear 0 or 1 times within a given element in the XML document. Attributes are either optional or mandatory (by default they are optional). The "use" property in the XSD definition is used to specify if the attribute is optional or mandatory. An attribute is specified within a `xsd:complexType`, the type information for the attribute comes from a `xsd:simpleType` (either defined inline or via a reference to a built in or user defined `xsd:simpleType` definition). The Type information describes the data the attribute can contain in the XML document, such as string, integer, date. Attributes can also be specified globally and then referenced.

Note 2 to entry: The full definition is provided in World Wide Web Consortium's XML Schema Definition Language. ^[14]

3.1.2.5

extension

<XSD> complex type that is inherited

Note 1 to entry: It is possible to take an existing `<xsd:complexType>` and extend it using `<xsd:extension>` and the "base" attribute. The introduced construct `<xsd:extension>` indicates that an existing type is extended and specifies a new type. The construct `<xsd:complexContent>` must be used to as container for the extension.

Note 2 to entry: The full definition is provided in World Wide Web Consortium's XML Schema Definition Language. ^[14]

3.1.2.6

group

<XSD> reusable collection of elements and attributes in an XML schema

Note 1 to entry: Elements and Attributes can be grouped together using <xsd:group> and <xsd:attributeGroup>. These groups can then be referred to elsewhere within the schema. Groups have a unique name and are defined as children of the <xsd:schema> element. When a group is referred to, it is as if its contents have been copied into the location from which it is referenced.

Note 2 to entry: The full definition is provided in World Wide Web Consortium's XML Schema Definition Language. [14]

3.1.2.7

namespace

<XSD> scope for named elements in an XML file

Note 1 to entry: Namespaces are a mechanism for partitioning schemas. XSD standard defines a structure XSD schemas by breaking them into multiple files. These child schemas can be included into a parent schema. Breaking schemas into multiple files can have several advantages: re-usable definitions can be used across several projects; definitions are easier to read and version because the schema is in smaller units that are simpler to manage.

Note 2 to entry: The full definition is provided in World Wide Web Consortium's XML Schema Definition Language. [14]

3.1.2.8

restriction

<XSD> definition of acceptable values for elements in an XML schema

Note 1 to entry: The usage of extensions, mixed contents, namespaces, groups, provides the capability to restrict the definition of a type.

Note 2 to entry: The full definition is provided in World Wide Web Consortium's XML Schema Definition Language. [14]

3.1.2.9

simple type

<XSD> element type with text-only attributes in an XML schema

3.1.3 Terms related to Schematron rules

3.1.3.1

Schematron

language for making assertions about the presence or absence of patterns in XML trees

Note 1 to entry: The full definition and specification is provided in ISO/IEC 19757-3.

3.1.3.2

Schematron rule

XML assertions to validate the presence or absence of patterns in XML trees

Note 1 to entry: Rules can be associated with "plain English" (or any language) validation error messages, allowing translation of numeric Schematron error codes into meaningful user error messages.

3.2 Abbreviated terms

For the purposes of this document, the following abbreviated terms apply:

CXMI	Canonical XMI
ID	IDentifier
OCL	Object Constraint Language
OMG	Object Management Group
STEP	STandard for the Exchange of Product model data
SCH	Schematron
SysML	Systems Modeling Language
UML	Unified Modeling Language
UoS	Unit of Serialization
UUID	Universal Unique Identifier
URI	Uniform Resource Identifier
XMI	XML Meta-data Interchange
XML	eXtensible Markup Language
XSD	XML Schema Definition

4 Structure and components of the XSD

4.1 General

This clause describes the structure and components of the resulting XSD. The resulting XSD shall conform to World Wide Web Consortium's XML Schema Definition Language (W3C XSD). [\[13\]](#)

This document shall be unambiguously identified in an open information system by the code defined in [Annex A](#).

[Clause 5](#) provides the mapping specification of each XSD construct mentioned in [Clause 4](#).

[Clause 6](#) provides the mapping specification from XMI to Schematron (SCH) rules. The XSD file is intended to be used with the Schematron rule file to ensure that STEP XML data based on the XSD are fully compliant to the SysML model. Schematron rules cover verifications not supported by the XSD or not suitable to be done via XSD language, such as model structural rules and formal propositions.

4.2 Presentation conventions

For ease of identification, the fragments of XSD are presented in boxes.

EXAMPLE 1 XSD fragment presented in a box:

XSD extract

The items significant to support the explanations are formatted using bold text effect to aid identification of the items in the XSD fragment.

EXAMPLE 2 Usage of bold text effect to support the explanation.

An XML **attribute** is contained in the XML element:

```
<Element attribute="...">
...
```

4.3 Main components of the XSD

The resulting XSD shall be composed of the following main components:

- the XSD header, located at the beginning of the file;
- common structural XML element definitions;
- the specification of the DataContainer;
- the list of application object specifications in XSD;
- the groups and simple types corresponding to Select Data Types and Enumerations.

Each of these components are described in [4.4](#) to [4.9](#).

4.4 XSD header

The header of the XSD (xsd:schema) defines:

- the namespaces of the XSD schema. Regarding the namespace conventions, the namespace prefixes are used throughout this document to refer to the namespaces identified by the corresponding URI. The prefixes and associated URIs are the following:
- xmlns:targetNamespace: https://standards.iso.org/iso/ts/10303/-4442/-ed-1/tech/xml-schema/domain_model,
- xmlns: https://standards.iso.org/iso/ts/10303/-4442/-ed-1/tech/xml-schema/domain_model,
- xsd: <https://www.w3.org/2001/XMLSchema>,
- xsi: <https://www.w3.org/2001/XMLSchema-instance>;
- resulting XML schema version, that is defined by an ISO number and a date.

A valid resulting XSD header is provided in the following XSD fragment:

```
XSD:

<xsd:schema xmlns="https://standards.iso.org/iso/ts/10303/-4442/ed-4/tech/xml-schema/domain_model"
            xmlns:xsd="https://www.w3.org/2001/XMLSchema"
            targetNamespace="https://standards.iso.org/iso/ts/10303/-4442/ed-4/tech/xml-schema/domain_model"
            version="N11164;2023-06-28">
```

4.5 Common structural XML element definitions

The header of the XSD shall be followed by a Unit of Serialization (UoS) by declaring an xsd:element named UoS. The complex type (xsd:ComplexType) Uos specifies that an XML shall include a UoS object that includes a header (described below) and one or more DataContainers (see [4.7](#)).

The mandatory header element contains administrative information that characterizes the content of the data package. The header elements are described in ISO 10303-28:2007, 5.2, and are as follows:

- Name: human readable identifier for the XML resource;
- TimeStamp: date and time when the XML resource was created;

ISO/TS 10303-15:2024(en)

- Author: identifies the person or group of persons who created the XML resource;
- Organization: identifies the organization that created, or is responsible for, the XML resource;
- PreprocessorVersion: identifies the software system that created the XML resource itself, including platform and version identifiers;

NOTE 1 The PreprocessorVersion identifies the system used to produce the XML resource. It can be distinct from the software system that created or captured the original information.

- OriginatingSystem: identifies the software system that created or captured the information contained in the XML resource, including platform and version identifiers;
- Authorization: specifies the release authorization for the XML resource and the signatory, where appropriate;

NOTE 2 The Authorization can be distinct from the authorizations for various information units contained within the document.

- Documentation: free text field for information;
- Uuid5namespace: string field in the “Header” for the UUID5 name space typed as UUID (see 4.6 for more information).

The following XSD extract is the XSD specification of the Header ComplexType:

```
XSD
<xsd:complexType name="Header">
  <xsd:annotation>
    <xsd:documentation/>
  </xsd:annotation>
  <xsd:sequence>
    <xsd:element name="Name" type="xsd:string" minOccurs="0"/>
    <xsd:element name="TimeStamp" type="xsd:dateTime" minOccurs="0"/>
    <xsd:element name="Author" type="NameAndAddress" minOccurs="0"/>
    <xsd:element name="Organization" type="NameAndAddress" minOccurs="0"/>
    <xsd:element name="PreprocessorVersion" type="xsd:string" minOccurs="0"/>
    <xsd:element name="OriginatingSystem" type="xsd:string" minOccurs="0"/>
    <xsd:element name="Authorization" type="xsd:string" minOccurs="0"/>
    <xsd:element name="Documentation" type="xsd:string" minOccurs="0"
maxOccurs="unbounded"/>
    <xsd:element name="uuid5namespace" type="UUID" minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>
```

The UoS XSD also defines the following structural features:

- BaseObject: this is the generic object from which all entities are extended. This element type is abstract. This object specifies that all XML elements shall have the XML attribute “uid” typed by the standardized xsd:ID type and may have an optional uuid XML attribute. In XML dataset ruled by this XSD, its element shall have unique “uid” XML attributes. The BaseObject specification shall be as follows: