

SMPTE STANDARD SMPTE

Register Interchange Format



Page 1 of 25 pages

Table of Contents	Page
Foreword	2
Intellectual Property	2
1 Introduction (Informative)	3
2 Scope	3
3 Conformance Notation	3
4 Normative References	4
5 Definition of Acronyms and Terms.....	4
5.1 Acronyms and Terms.....	4
6 Generic Class Model.....	5
6.1 Overview (Informative).....	5
6.2 Scalar Data Types.....	6
6.3 Compound Data Types	7
6.4 Class UL Definitions.....	8
6.5 RIF Class Definitions.....	9
6.6 RIF Base Class	9
7 XML Encoding Rules.....	14
7.1 Namespaces	14
7.2 Scalar Data Types.....	15
7.3 Compound Data Types	16
7.4 References.....	16
7.5 XML Document Structure.....	17
8 Rules for Extending the Data Model to Support Specific Registers	18
8.1 Namespaces	18
8.2 Scalar Data Types.....	18
8.3 Compound Data Types	18
8.4 References.....	18
Annex A Base XML Schema Definition (Normative).....	19
Annex B Class Model and XML Schema Extension Example for SMPTE 395M (Informative).....	20
B.1 Class Model Extension.....	20
B.2 RIF-Schema Extension	22
Annex C Class Model and XML Schema Extension Example for SMPTE 335M (Informative).....	23
C.1 Class Model Extension.....	23
C.2 RIF-Schema Extension	23
Annex D Bibliography (Informative)	25

Foreword

SMPTE (the Society of Motion Picture and Television Engineers) is an internationally-recognized standards developing organization. Headquartered and incorporated in the United States of America, SMPTE has members in over 80 countries on six continents. SMPTE's Engineering Documents, including Standards, Recommended Practices, and Engineering Guidelines, are prepared by SMPTE's Technology Committees. Participation in these Committees is open to all with a bona fide interest in their work. SMPTE cooperates closely with other standards-developing organizations, including ISO, IEC and ITU.

SMPTE Engineering Documents are drafted in accordance with the rules given in Part XIII of its Administrative Practices.

SMPTE ST 2045 was prepared by Technology Committee 30MR.

Intellectual Property

At the time of publication no notice had been received by SMPTE claiming patent rights essential to the implementation of this Standard. However, attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. SMPTE shall not be held responsible for identifying any or all such patent rights.

1 Introduction (Informative)

This standard defines a common class model and XML encoding for the UL-based registers defined by SMPTE. The corresponding defining documents of these registers need to specify rules about how to extend the common class model to support the specific registers requirements.

At the time of writing this document the defining documents of the UL-based registers are under revision. Future releases of the RIF document family will further specify the encodings rules for each SMPTE UL-based register for the RIF representation according to this standard.

Requirements for register entries, both leaves and nodes, are contained in the defining document for these registers. This standard defines the common administrative and identification metadata required to manage the contents of the UL registers.

This standard starts with an overview of the generic common class model. Sections 6.2 and 6.3 define the scalar data types and the compound data types respectively. Section 6.4 defines the class identifiers, followed by Section 6.5 where all RIF classes are specified including their properties. The specification how to encode the contents of registers as XML documents is given in Section 7. The rules defining how to extend the data model to support specific registers can be found in Section 8. Annex A provides a base XML schema definition. In Annex B and Annex C informative examples are given which illustrate the support for SMPTE 395M and SMPTE 335M.

2 Scope

This standard specifies a generic data model to represent the contents of the UL-based SMPTE registers. Examples of such registers are SMPTE 335M, SMPTE 395M or SMPTE 400M.

This generic data model also provides elements to hold administrative information used for the maintenance of the UL-based SMPTE registers. It includes rules how to extend the data model to represent the contents of specific UL-based SMPTE registers.

This standard also specifies how to encode the contents of individual registers as XML documents.

This standard does not define the relationships between the different registers. The data model is limited to defining the attributes that can be used by different SMPTE registers.

3 Conformance Notation

Normative text is text that describes elements of the design that are indispensable or contains the conformance language keywords: "shall", "should", or "may". Informative text is text that is potentially helpful to the user, but not indispensable, and can be removed, changed, or added editorially without affecting interoperability. Informative text does not contain any conformance keywords.

All text in this document is, by default, normative, except: the Introduction, any section explicitly labeled as "Informative" or individual paragraphs that start with "Note:".

The keywords "shall" and "shall not" indicate requirements strictly to be followed in order to conform to the document and from which no deviation is permitted.

The keywords, "should" and "should not" indicate that, among several possibilities, one is recommended as particularly suitable, without mentioning or excluding others; or that a certain course of action is preferred but not necessarily required; or that (in the negative form) a certain possibility or course of action is deprecated but not prohibited.

The keywords "may" and "need not" indicate courses of action permissible within the limits of the document.

The keyword "reserved" indicates a provision that is not defined at this time, shall not be used, and may be defined in the future. The keyword "forbidden" indicates "reserved" and in addition indicates that the provision will never be defined in the future.

A conformant implementation according to this document is one that includes all mandatory provisions ("shall") and, if implemented, all recommended provisions ("should") as described. A conformant implementation need not implement optional provisions ("may") and need not implement them as described.

Unless otherwise specified the order of precedence of the types of normative information in this document shall be as follows. Normative prose shall be the authoritative definition. Tables shall be next, followed by formal languages, then figures, and then any other language forms.

4 Normative References

The following standards contain provisions which, through reference in this text, constitute provisions of this recommended practice. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this recommended practice are encouraged to investigate the possibility of applying the most recent edition of the standards indicated below.

IETF RFC 3061, A URN Namespace of Object Identifiers

IETF RFC 4122, A Universally Unique Identifier (UUID) URN Namespace

ISO/IEC 10646:2003, Information technology – Universal Multiple-Octet Coded Character Set (UCS)

SMPTE 298M-2009, Universal Labels for Unique Identification of Digital Data

SMPTE 335M-2001, Metadata Element Dictionary Structure

SMPTE 336M-2007, Data Encoding Protocol Using Key-Length-Value

SMPTE 395M-2003, Metadata Groups Registry

SMPTE 400M-2004, SMPTE Labels Structure

SMPTE 2029-2009, Uniform Resource Names for SMPTE Resources

XML, Extensible Markup Language (XML) 1.0, W3C Recommendation

Namespaces in XML, W3C Recommendation

XML Schema Part 1: Structures, W3C Recommendation

XML Schema Part 2: Datatypes, W3C Recommendation

5 Definition of Acronyms and Terms

Acronyms, terms and data types used in the RIF specification are defined in this section.

5.1 Acronyms and Terms

Abstract Class: A Class used within the definition of an inheritance hierarchy that cannot be instantiated as an Object. Only non-abstract Subclasses of an Abstract Class can be instantiated as Objects.

Aggregation: A form of Association that defines a whole-part relationship between a whole Object and a part Object or multiple part Objects. See Composition.

Class: A category of Objects with defined properties and defined behavior.

Composition: A stronger form of Aggregation in which parts may only belong to one whole and may only exist as part of the whole. Composition is used to combine simple Objects into more complex ones.

KLV: Key Length Value. The binary encoding protocol defined in SMPTE 336M.

Object: An instance of a Class.

Reference: A relationship between Objects (e.g. StrongRef).

RIF: Register Interchange Format.

RIF Base XML-Schema: The core part of the RIF XML-Schema which is provided in Annex A. This XML-Schema has to be extended in order to support specific UL-based SMPTE registers.

Set: A grouping of KLV items defined in SMPTE336M.

StrongRef: Strong Reference. A Composition relationship between Objects implemented as an aggregation by reference between Objects using identifiers (e.g. the value of an InstanceUID). Strong References are a one to one relationship. They are typed. This means that the definition of the aggregation identifies the kinds of Objects (encoded using KLV or XML encoding) which may be the target of the reference.

Subclass: A Class that is directly or indirectly derived through Inheritance from another Class. The other Class is also called a Superclass of the Subclass.

UL: SMPTE Universal Label (see SMPTE 298M).

UL Item Designator: The last 8 bytes of a SMPTE Universal Label. They uniquely identify an entry in a SMPTE register. The Item Designator is a reference to the SMPTE label inside the register that is identified by the UL Designator. SMPTE registers relevant to this specification are SMPTE 335M, SMPTE 400M and SMPTE 395M.

UL Designator: The first 8 bytes of a SMPTE Universal Label. They identify the register, its minor and major versions, and may also convey other information that is used by the KLV encoding protocol. The register is the normative reference for the values of these 8 bytes.

UUID: Universally Unique Identifier according to IETF RFC 4122.

6 Generic Class Model

6.1 Overview (Informative)

SMPTE 336M defines the elements that are needed to exactly define entries in all Universal Label-based SMPTE registers. The upper part of the UML class diagram in Figure 1 illustrates this common base which includes the definition of the abstract classes "RifBaseClass", "AdministrationBaseClass", "RegisterAdministrationBaseClass", "EntryAdministrationBaseClass", "Entry", "Leaf" and CompoundEntryElementBaseClass. Also it includes the definition of the concrete classes "Register", "Node" and ApplicationInformation.

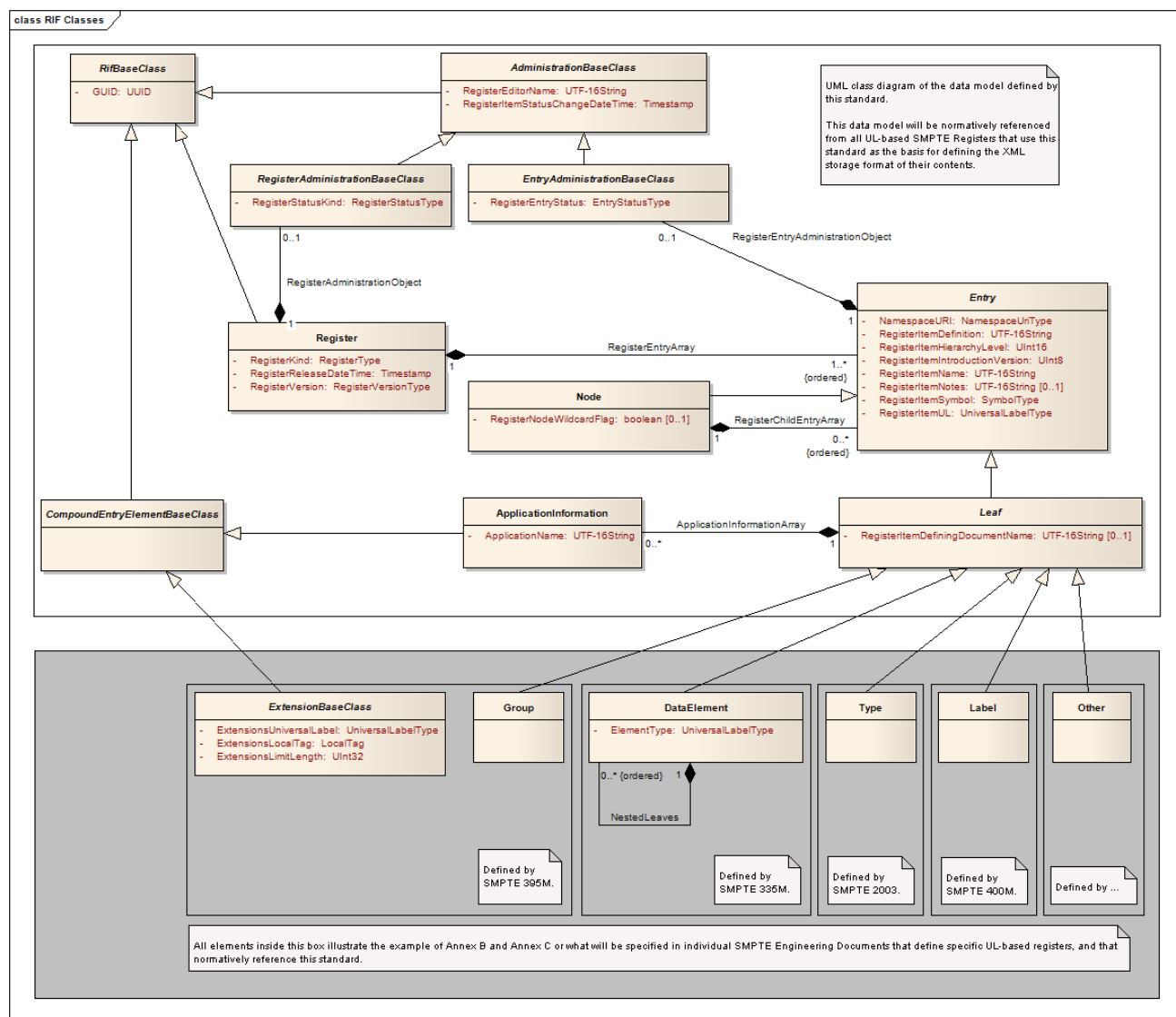


Figure 1 – UML class diagram of the common class model

Each individual Register extends the abstract base class “Leaf” to represent its own, register-specific information. This is illustrated in the lower part of Figure 1.

6.2 Scalar Data Types

This section defines the scalar data types used by elements of the RIF classes. The corresponding XML encodings of the type values are defined in Section 7.2.

Table 1 – Scalar Data Types

Data Type	Definition
<i>Boolean</i>	<i>A logical 'true' or 'false'.</i>
<i>Byte</i>	<i>A unit of information storage.</i>
<i>EntryStatusType</i>	<i>Enumerated value. The approval status of an entry.</i>
<i>Int16</i>	<i>Signed 16-bit integer. Note: An integer in the range -32768 to 32767.</i>
<i>LocalTag</i>	<i>Local Tag as defined in SMPTE 336M.</i>
<i>NamespaceURIType</i>	<i>String type that conforms to XML language syntax.</i>
<i>RegisterStatusType</i>	<i>Enumerated value. The published status of a register.</i>
<i>RegisterType</i>	<i>Enumerated value. The type of a register.</i>
<i>StrongRef</i>	<i>Strong Reference. A Composition relationship between Objects implemented as an aggregation by reference between Objects using UUIDs as identifiers for sub-classes of RifBaseClass and using UL for sub-classes of Entry. Strong References are a one to one relationship. They are typed. This means that the definition of the aggregation identifies the kinds of Objects which may be the target of the reference.</i>
<i>SymbolType</i>	<i>A symbol is a name that conforms to computer language syntax restrictions, and it is intended for use in computer languages such as the Extensible Markup Language (XML). To enable the use of symbols in a wide range of computer languages, a symbol shall be a string composed only of the characters A-Z, a-z, 0-9, and _, and it shall begin with an alpha character (A-Z, a-z) or an underscore (_).</i>
<i>UInt8</i>	<i>Unsigned 8-bit integer. Note: An integer in the range 0 to 255.</i>
<i>UInt16</i>	<i>Unsigned 16-bit integer. Note: An integer in the range 0 to 65535.</i>
<i>UInt32</i>	<i>Unsigned 32-bit integer. Note: An integer in the range 0 to 4294967295.</i>
<i>UnitsOfMeasureType</i>	<i>Enumerated value. A list of the units of measure in order to ensure consistency. E.g. "meters", "seconds", "degree Celsius".</i>
<i>UniversalLabel</i>	<i>SMPTE Universal Label as defined in SMPTE 298M.</i>
<i>UTF-16String</i>	<i>A variable-length string that is able to hold any valid sequence of Unicode symbols. Note: The encoding is defined in ISO/IEC 10646:2003, Annex Q. RFC-2781 also contains a description of the encoding.</i>
<i>UUID</i>	<i>Universally unique identifier according to IETF RFC 4122.</i>

6.3 Compound Data Types

This section defines the compound data types used by elements of the RIF classes. The corresponding XML encodings of the type values are defined in Section 7.3.

Table 2 – Compound Data Types

Data Type	Definition
Array	A compound type comprising multiple individual elements where the elements are ordered and the type is defined. The order of the individual Array elements in the Array is significant. An Array is identified as an Array of Type, where Type is the defined type of the elements of the Array.
Batch	A compound type comprising multiple individual elements where the elements are unordered and unique and the type is defined. The order of the individual Batch elements in a Batch is not significant. RIF encoders shall not place duplicate elements into a Batch. RIF decoders shall be able to successfully decode RIF instances which do contain Batches with repeated Batch element values. A Batch is identified as a Batch of Type, where Type is the defined type of the elements of the Batch.
Timestamp	A time and date item according to the Gregorian calendar comprising Year: [Int16], Month: [UInt8], Day: [UInt8], Hour: [UInt8], Minute: [UInt8], Second: [UInt8] and mSec/4: [UInt8]. A value of '0' for every field identifies a timestamp value of 'unknown'. This value should not be used unless unavoidable. Timestamp values shall be specified according to UTC time.
RegisterVersionType	A compound data type comprising the major version number [UInt8] and the minor version number [UInt8] of the register.

6.4 Class UL Definitions

Class IDs shall be Universal Labels. All RIF classes shall be registered in the SMPTE Groups register. The following tables specify the Universal Label values for the classes that are defined in this document.

Table 3 – Common Universal Label for all RIF classes

Byte No.	Description	Value (hex)	Meaning
1	Object Identifier	06h	
2	Label size	0Eh	
3	Designator	2Bh	ISO, ORG
4	Designator	34h	SMPTE
5	Register Category Designator	02h	Sets & packs
6	Register Designator:	7fh	SMPTE 395M-defined default value for undefined KLV codings.
7	Structure Designator	01h	Set/Pack Dictionary
8	Version Number	01h	Register Version 1
9	Item Designator	0Ch	Compound
10	Item Designator	01h	Data Models
11	Item Designator	01h	Dictionary Representation
12	Application	01h	Register Interchange Format (RIF)
13	Structure version	01h	RIF Version 1 compatible classes
14	Set Kind (1)	xxh	RIF class Definition (see)
15	Set Kind (2)	yyh	RIF class Definition (see)
16	Reserved	00h	Reserved