

## Recommendation for Space Data System Practices

# ABSTRACT EVENT DEFINITION

**RECOMMENDED PRACTICE**

**CCSDS 902.13-M-1**

**MAGENTA BOOK**  
February 2021

**Recommendation for Space Data System Practices**

**ABSTRACT EVENT  
DEFINITION**

**RECOMMENDED PRACTICE**

**CCSDS 902.13-M-1**

**MAGENTA BOOK**

**February 2021**

## AUTHORITY

Issue:	Recommended Practice, Issue 1
Date:	February 2021
Location:	Washington, DC, USA

This document has been approved for publication by the Management Council of the Consultative Committee for Space Data Systems (CCSDS) and represents the consensus technical agreement of the participating CCSDS Member Agencies. The procedure for review and authorization of CCSDS documents is detailed in *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4), and the record of Agency participation in the authorization of this document can be obtained from the CCSDS Secretariat at the email address below.

This document is published and maintained by:

CCSDS Secretariat  
National Aeronautics and Space Administration  
Washington, DC, USA  
Email: [secretariat@mailman.ccsds.org](mailto:secretariat@mailman.ccsds.org)

## STATEMENT OF INTENT

The Consultative Committee for Space Data Systems (CCSDS) is an organization officially established by the management of its members. The Committee meets periodically to address data systems problems that are common to all participants, and to formulate sound technical solutions to these problems. Inasmuch as participation in the CCSDS is completely voluntary, the results of Committee actions are termed **Recommendations** and are not in themselves considered binding on any Agency.

CCSDS Recommendations take two forms: **Recommended Standards** that are prescriptive and are the formal vehicles by which CCSDS Agencies create the standards that specify how elements of their space mission support infrastructure shall operate and interoperate with others; and **Recommended Practices** that are more descriptive in nature and are intended to provide general guidance about how to approach a particular problem associated with space mission support. This **Recommended Practice** is issued by, and represents the consensus of, the CCSDS members. Endorsement of this **Recommended Practice** is entirely voluntary and does not imply a commitment by any Agency or organization to implement its recommendations in a prescriptive sense.

No later than five years from its date of issuance, this **Recommended Practice** will be reviewed by the CCSDS to determine whether it should: (1) remain in effect without change; (2) be changed to reflect the impact of new technologies, new requirements, or new directions; or (3) be retired or canceled.

In those instances when a new version of a **Recommended Practice** is issued, existing CCSDS-related member Practices and implementations are not negated or deemed to be non-CCSDS compatible. It is the responsibility of each member to determine when such Practices or implementations are to be modified. Each member is, however, strongly encouraged to direct planning for its new Practices and implementations towards the later version of the Recommended Practice.

## FOREWORD

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Recommended Practice is therefore subject to CCSDS document management and change control procedures, which are defined in the *Organization and Processes for the Consultative Committee for Space Data Systems* (CCSDS A02.1-Y-4). Current versions of CCSDS documents are maintained at the CCSDS Web site:

<http://www.ccsds.org/>

Questions relating to the contents or status of this document should be sent to the CCSDS Secretariat at the email address indicated on page i.

At time of publication, the active Member and Observer Agencies of the CCSDS were:

Member Agencies

- Agenzia Spaziale Italiana (ASI)/Italy.
- Canadian Space Agency (CSA)/Canada.
- Centre National d’Etudes Spatiales (CNES)/France.
- China National Space Administration (CNSA)/People’s Republic of China.
- Deutsches Zentrum für Luft- und Raumfahrt (DLR)/Germany.
- European Space Agency (ESA)/Europe.
- Federal Space Agency (FSA)/Russian Federation.
- Instituto Nacional de Pesquisas Espaciais (INPE)/Brazil.
- Japan Aerospace Exploration Agency (JAXA)/Japan.
- National Aeronautics and Space Administration (NASA)/USA.
- UK Space Agency/United Kingdom.

Observer Agencies

- Austrian Space Agency (ASA)/Austria.
- Belgian Science Policy Office (BELSPO)/Belgium.
- Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
- China Satellite Launch and Tracking Control General, Beijing Institute of Tracking and Telecommunications Technology (CLTC/BITTT)/China.
- Chinese Academy of Sciences (CAS)/China.
- China Academy of Space Technology (CAST)/China.
- Commonwealth Scientific and Industrial Research Organization (CSIRO)/Australia.
- Danish National Space Center (DNSC)/Denmark.
- Departamento de Ciência e Tecnologia Aeroespacial (DCTA)/Brazil.
- Electronics and Telecommunications Research Institute (ETRI)/Korea.
- European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
- European Telecommunications Satellite Organization (EUTELSAT)/Europe.
- Geo-Informatics and Space Technology Development Agency (GISTDA)/Thailand.
- Hellenic National Space Committee (HNSC)/Greece.
- Hellenic Space Agency (HSA)/Greece.
- Indian Space Research Organization (ISRO)/India.
- Institute of Space Research (IKI)/Russian Federation.
- Korea Aerospace Research Institute (KARI)/Korea.
- Ministry of Communications (MOC)/Israel.
- Mohammed Bin Rashid Space Centre (MBRSC)/United Arab Emirates.
- National Institute of Information and Communications Technology (NICT)/Japan.
- National Oceanic and Atmospheric Administration (NOAA)/USA.
- National Space Agency of the Republic of Kazakhstan (NSARK)/Kazakhstan.
- National Space Organization (NSPO)/Chinese Taipei.
- Naval Center for Space Technology (NCST)/USA.
- Netherlands Space Office (NSO)/The Netherlands.
- Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
- Scientific and Technological Research Council of Turkey (TUBITAK)/Turkey.
- South African National Space Agency (SANSA)/Republic of South Africa.
- Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
- Swedish Space Corporation (SSC)/Sweden.
- Swiss Space Office (SSO)/Switzerland.
- United States Geological Survey (USGS)/USA.

## DOCUMENT CONTROL

<b>Document</b>	<b>Title</b>	<b>Date</b>	<b>Status</b>
CCSDS 902.13-M-1	Abstract Event Definition, Recommended Practice, Issue 1	February 2021	Original issue

## CONTENTS

<u>Section</u>	<u>Page</u>
<b>1 INTRODUCTION</b> .....	<b>1-1</b>
1.1 PURPOSE AND SCOPE.....	1-1
1.2 APPLICABILITY.....	1-1
1.3 RATIONALE.....	1-2
1.4 DOCUMENT STRUCTURE.....	1-2
1.5 DEFINITIONS.....	1-2
1.6 NOMENCLATURE.....	1-2
1.7 CONVENTIONS—.....	1-3
1.8 REFERENCES.....	1-3
<b>2 OVERVIEW</b> .....	<b>2-1</b>
2.1 GENERAL.....	2-1
2.2 DATA ENTITIES.....	2-1
2.3 MAPPING TO W3C XML SCHEMA.....	2-1
<b>3 ABSTRACT EVENT DATA ENTITIES</b> .....	<b>3-1</b>
3.1 OVERVIEW.....	3-1
3.2 ABSTRACT EVENT DATA ENTITY.....	3-2
<b>ANNEX A SECURITY, SANA, AND PATENT CONSIDERATIONS (INFORMATIVE)</b> .....	<b>A-1</b>
<b>ANNEX B XML SCHEMA ORGANIZATION AND PACKAGING FOR THE ABSTRACT EVENT CLASSES (INFORMATIVE)</b> .....	<b>B-1</b>
<b>ANNEX C INFORMATIVE REFERENCES (INFORMATIVE)</b> .....	<b>C-1</b>
<b>ANNEX D ABBREVIATIONS AND ACRONYMS (INFORMATIVE)</b> .....	<b>D-1</b>

### Figure

1-1 Abstract Event Class in the Context of Space Communication Cross Support Service Management.....	1-1
3-1 Abstract Event Class Diagram.....	3-1

### Table

3-1 Class AbstractEvent Parameters.....	3-2
3-2 Class AbstractParameter Parameters.....	3-4
3-3 Class BooleanParameter Parameters.....	3-4

**CONTENTS (continued)**

<u>Table</u>	<u>Page</u>
3-4 Class UnsignedIntParameter Parameters .....	3-5
3-5 Class IntParameter Parameters .....	3-5
3-6 Class FloatParameter Parameters .....	3-6
3-7 Class DoubleParameter Parameters .....	3-6
3-8 Class StringParameter Parameters .....	3-7
3-9 Class ExtendedParameter Parameters.....	3-7
3-10 Class DurationUnIntParameter Parameters .....	3-8
3-11 Class DurationDblParameter Parameters .....	3-8
3-12 Class TimeParameterA Parameters .....	3-8
3-13 Class TimeParameterB Parameters.....	3-9
3-14 Class AbstractEventTime Parameters.....	3-9
3-15 Class CcsdsAsciiTimeCodeA Parameters .....	3-10
3-16 Class CcsdsAsciiTimeCodeB Parameters .....	3-10
3-17 Class RelativeEventTime Parameters.....	3-11
3-18 Class RelativeToTimeA Parameters.....	3-12
3-19 Class RelativeToTimeB Parameters .....	3-12

# 1 INTRODUCTION

## 1.1 PURPOSE AND SCOPE

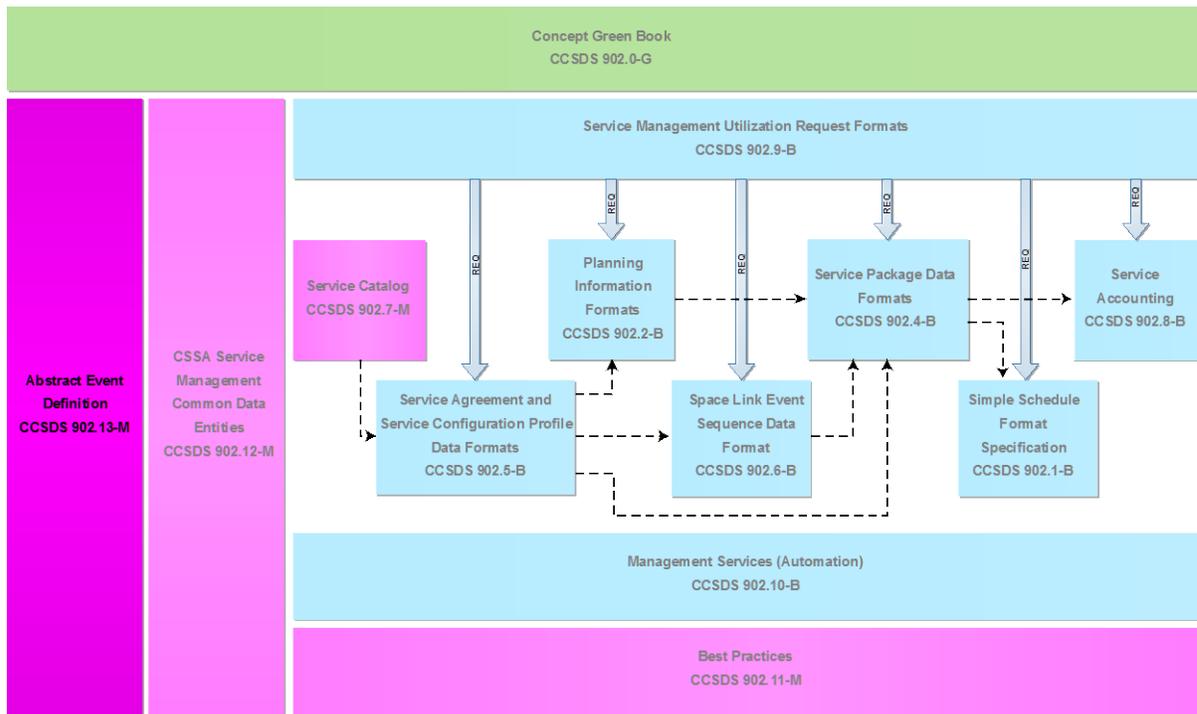
### 1.1.1 PURPOSE

The purpose of this Recommended Practice is to define an event in the abstract that is the basis for consistent tailoring for use by other CCSDS specifications.

### 1.1.2 SCOPE

The scope of this book is of the Abstract Event class.

Figure 1-1 puts the Abstract Event class Specification into context with the various standards that together form the Space Communication Cross Support Service Management.



**Figure 1-1: Abstract Event Class in the Context of Space Communication Cross Support Service Management**

## 1.2 APPLICABILITY

Whilst the Abstract Event class has been defined in the scope of the Service Management working group, it has been discussed with other working groups, principally Navigation, and so is expected to be applicable to other specifications in addition to those for Cross Support Service Management.

### 1.3 RATIONALE

The rationale for this document is to provide a standard definition of the event data structure.

### 1.4 DOCUMENT STRUCTURE

This document is organized as follows:

- a) Section 1 provides the purpose, scope, applicability, and rationale of this Recommended Practice and identifies the conventions and references used throughout the document. This section also describes how this document is organized. A brief description is provided for each section and annex so that the reader has an idea of where information can be found in the document. It also identifies terminology that is used in this document but is defined elsewhere.
- b) Section 2 provides a brief overview of the Abstract Event data entities.
- c) Section 3 provides details of the Abstract Event entities.
- d) Annex A discusses security, SANA, and patent considerations.
- e) Annex B provides an informative description of XML schema organization and packaging for the Abstract Event.
- f) Annex C is a list of informative references.
- g) Annex D contains a list of Acronyms applicable to the Abstract Event Definition.

### 1.5 DEFINITIONS

For the purposes of this document, the following definition applies:

**agency:** A satellite operator or satellite service provider.

### 1.6 NOMENCLATURE

#### 1.6.1 NORMATIVE TEXT

The following conventions apply for the normative specifications in this Recommended Practice:

- a) the words ‘shall’ and ‘must’ imply a binding and verifiable specification;
- b) the word ‘should’ implies an optional, but desirable, specification;
- c) the word ‘may’ implies an optional specification;
- d) the words ‘is’, ‘are’, and ‘will’ imply statements of fact.

NOTE – These conventions do not imply constraints on diction in text that is clearly informative in nature.

## 1.6.2 INFORMATIVE TEXT

In the normative sections of this document, informative text is set off from the normative specifications, either in notes or under one of the following subsection headings:

- Overview;
- Background;
- Rationale;
- Discussion.

## 1.7 CONVENTIONS

### 1.7.1 THE UNIFIED MODELING LANGUAGE

The Unified Modeling Language (UML) diagrams used in the specification (including class diagrams, package diagrams, sequence diagrams, and activity diagrams) follow the notation, semantics, and conventions imposed by the Version 2.4.1 UML specification of the Object Management Group (OMG) (reference [2]).

### 1.7.2 XML SCHEMA DATATYPES

Many of the datatype definitions in this Recommended Practice are the same as definitions defined by XML Schema, Part 2 Datatypes (see reference [4]). Wherever an XML Schema datatype is indicated in this Recommended Practice, it is prefixed with the string “xsd:”.

## 1.8 REFERENCES

The following publications contain provisions which, through reference in this text, constitute provisions of this document. At the time of publication, the editions indicated were valid. All publications are subject to revision, and users of this document are encouraged to investigate the possibility of applying the most recent editions of the publications indicated below. The CCSDS Secretariat maintains a register of currently valid CCSDS publications.

- [1] *Time Code Formats*. Issue 4. Recommendation for Space Data System Standards (Blue Book), CCSDS 301.0-B-4. Washington, D.C.: CCSDS, November 2010.
- [2] *Unified Modeling Language (UML)*. Version 2.4.1. Needham, Massachusetts: Object Management Group, August 2011.
- [3] *Space Assigned Numbers Authority (SANA)—Role, Responsibilities, Policies, and Procedures*. Issue 3. CCSDS Record (Yellow Book), CCSDS 313.0-Y-3. Washington, D.C.: CCSDS, October 2020.
- [4] David Peterson, et al., eds. *W3C XML Schema Definition Language (XSD) 1.1 Part 2: Datatypes*. Version 1.1. W3C Recommendation.

## **2 OVERVIEW**

### **2.1 GENERAL**

This section provides a high-level overview of the Abstract Event.

### **2.2 DATA ENTITIES**

The Abstract Event data entities are defined via a UML class diagram, which is then mapped to a W3C XML Schema.

Data is either mandatory, in which case suitable values must be present, or optional, in which case values may be either present or not. Additionally, it is possible to extend the contents of the data entities by defining additional parameters. The content of any additional parameters so defined is outside the scope of this document and should be documented in an ICD agreed upon by the involved parties.

### **2.3 MAPPING TO W3C XML SCHEMA**

This Recommended Practice includes the specification of a mapping to World Wide Web Consortium (W3C) eXtensible Markup Language (XML) schema. The normative mapping of this Recommended Practice to XML W3C schemas is a virtual annex to this Recommended Practice and is contained in a stand-alone set of schema files.

NOTE – The XML schema has been elaborated on the basis of the mapping guidelines described in reference [C1].

### 3 ABSTRACT EVENT DATA ENTITIES

#### 3.1 OVERVIEW

##### 3.1.1 GENERAL

The abstract event data entity is defined in 3.2.

##### 3.1.2 ABSTRACT EVENT CONTENT/STRUCTURE

The AbstractEvent class forms the basis for defining Event.

Figure 3-1 shows the UML Class diagram for the Abstract Event Class. For clarity, abstract classes are highlighted in green.

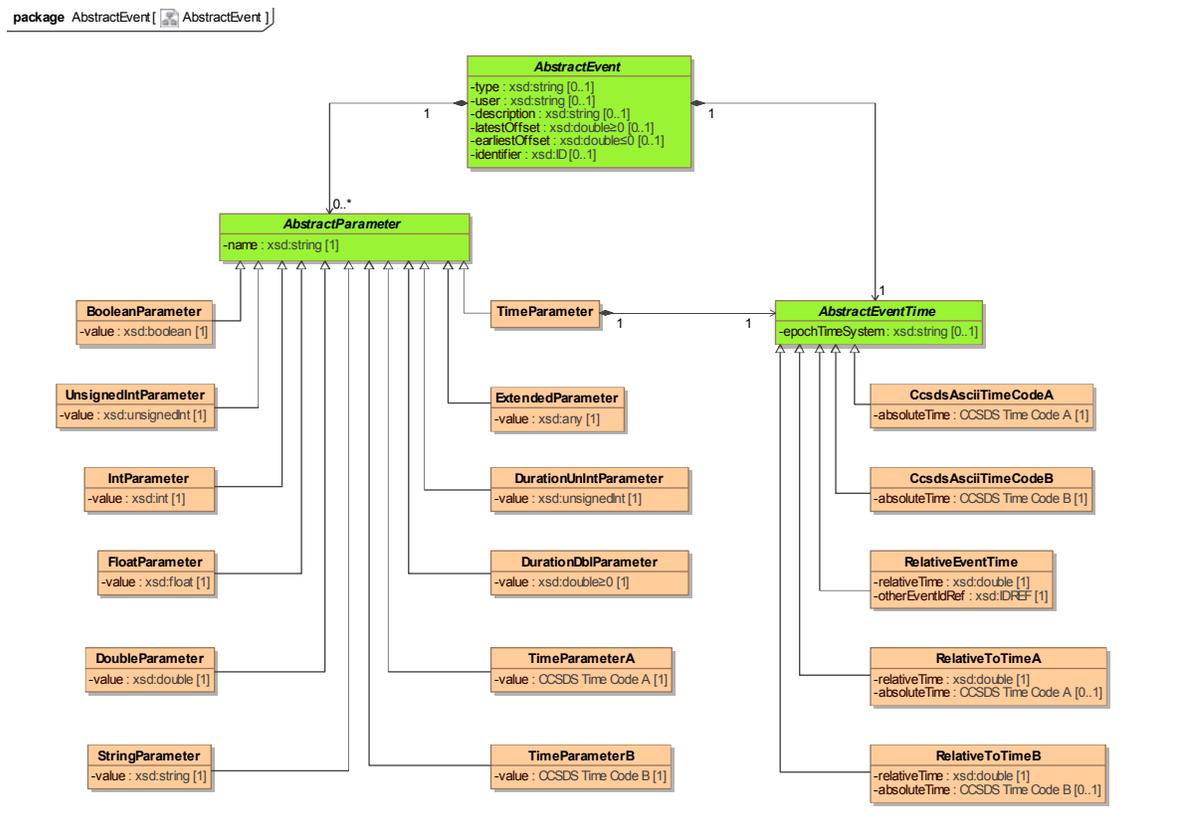


Figure 3-1: Abstract Event Class Diagram

The attributes of each class are described further in the following subsections and tables.

## 3.2 ABSTRACT EVENT DATA ENTITY

### 3.2.1 CLASS ABSTRACTEVENT (ABSTRACT)

**3.2.1.1** The AbstractEvent class is an abstract class that is used to instantiate various events.

**3.2.1.2** There shall be one and only one instantiation of a class specialized from the AbstractEventTime class for each instantiation of the AbstractEvent class.

**3.2.1.3** The AbstractEvent class shall contain the parameters specified in table 3-1.

**Table 3-1: Class AbstractEvent Parameters**

Parameter	Description	Data Type	Data Units
type	Optional Parameter. This is used to specify the type of the event.  NOTE – Recommendations that make use of the Abstract Event Definition shall make this parameter mandatory unless a waiver is granted by CESG. (Also see SANA Considerations, A2, concerning the registry for this parameter.)	xsd:string	n/a
user	Optional Parameter. This can be used to specify the user to which the event is relevant.	xsd:string	n/a
description	Optional Parameter. This can be used to specify a description of the event.	xsd:string	n/a
latestOffset	Optional Parameter. This can be used to express an offset from the expected time of the event to the latest time at which the event can occur.  This can be used to express any uncertainty about when exactly the event occurs.	xsd:double $\geq 0$	Seconds

Parameter	Description	Data Type	Data Units
earliestOffset	Optional Parameter. This can be used to express an offset from the expected time of the event to the earliest time at which the event can occur.  This can be used to express any uncertainty about when exactly the event occurs.	xsd:double ≤0	Seconds
identifier	Optional Parameter. An identifier that is unique for every occurrence of a particular event in a particular file.  NOTE – This Recommended Practice does not impose any requirement on persistence of identifiers. Should it be required, this shall be specified in the document that adopts the Abstract Event Definition.	xsd:ID	n/a

NOTE – The xsd:ID type is used for an attribute that uniquely identifies an element in an XML document. An xsd:ID value is an NCName. This means that it starts with a letter or underscore and can only contain letters, digits, underscores, hyphens, and periods. xsd:ID carries several additional constraints:

- xsd:ID values are unique within an XML instance, regardless of the attribute's name or its element name.
- A complex type cannot include more than one attribute of type xsd:ID, or any type derived from xsd:ID.
- xsd:ID attributes cannot have default or fixed values specified.

### 3.2.2 CLASS AbstractParameter (ABSTRACT)

3.2.2.1 The AbstractParameter is an abstract class that can be used to specialize various types of parameters.

3.2.2.2 The AbstractParameter class shall contain parameters as specified in table 3-2.

**Table 3-2: Class AbstractParameter Parameters**

Parameter	Description	Data Type	Data Units
name	The name of the parameter.	xsd:string	n/a

### 3.2.3 CLASS BooleanParameter

**3.2.3.1** The BooleanParameter is a class that can be used to instantiate a parameter with type xsd:boolean.

NOTE – The BooleanParameter class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.3.2** The BooleanParameter class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-3.

**Table 3-3: Class BooleanParameter Parameters**

Parameter	Description	Data Type	Data Units
value	The value of the parameter.	xsd:boolean	n/a

### 3.2.4 CLASS UnsignedIntParameter

**3.2.4.1** The UnsignedIntParameter is a class that can be used to instantiate a parameter with type xsd:unsignedInt.

NOTE – The UnsignedIntParameter class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.4.2** The UnsignedIntParameter class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-4.

**Table 3-4: Class UnsignedIntParameter Parameters**

Parameter	Description	Data Type	Data Units
value	The value of the parameter.	xsd:unsignedInt	n/a

### 3.2.5 CLASS IntParameter

**3.2.5.1** The IntParameter is a class that can be used to instantiate a parameter with type xsd:int.

NOTE – The IntParameter class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.5.2** The IntParameter class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-5.

**Table 3-5: Class IntParameter Parameters**

Parameter	Description	Data Type	Data Units
value	The value of the parameter.	xsd:int	n/a

### 3.2.6 CLASS FloatParameter

**3.2.6.1** The FloatParameter is a class that can be used to instantiate a parameter with type xsd:float.

NOTE – The FloatParameter class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.6.2** The FloatParameter class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-6.

**Table 3-6: Class FloatParameter Parameters**

Parameter	Description	Data Type	Data Units
value	The value of the parameter.	xsd:float (single-precision 32-bit floating-point numbers)	n/a

**3.2.7 CLASS DoubleParameter**

**3.2.7.1** The DoubleParameter is a class that can be used to instantiate a parameter with type xsd:double.

NOTE – The DoubleParameter class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.7.2** The DoubleParameter class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-7.

**Table 3-7: Class DoubleParameter Parameters**

Parameter	Description	Data Type	Data Units
value	The value of the parameter.	xsd:double (double-precision 64-bit floating-point numbers)	n/a

**3.2.8 CLASS StringParameter**

**3.2.8.1** The StringParameter is a class that can be used to instantiate a parameter with type xsd:string.

NOTE – The StringParameter class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.8.2** The StringParameter class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-8.

**Table 3-8: Class StringParameter Parameters**

Parameter	Description	Data Type	Data Units
value	The value of the parameter.	xsd:string	n/a

### 3.2.9 CLASS ExtendedParameter

**3.2.9.1** The ExtendedParameter is a class that can be used to instantiate a parameter with any required type. This can be used to define parameter types not specified above, including complex types.

NOTE – The ExtendedParameter class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.9.2** Use of a parameter derived from the ExtendedParameter class requires an extension to the Service management XML schema such that this contains the appropriate definition of the type. The extension should be registered in SANA as indicated in annex A.

**3.2.9.3** The ExtendedParameter class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-9.

**Table 3-9: Class ExtendedParameter Parameters**

Parameter	Description	Data Type	Data Units
value	The value of the parameter.	xsd:any	n/a

### 3.2.10 CLASS DurationUnIntParameter

**3.2.10.1** The DurationUnIntParameter is a class that can be used to instantiate a duration expressed as an integer number of seconds  $\geq 0$ .

NOTE – The DurationUnIntParameter class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.10.2** The DurationUnIntParameter class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-10.

**Table 3-10: Class DurationUnIntParameter Parameters**

Parameter	Description	Data Type	Data Units
Value	The value of the parameter.	xsd:unsignedInt	Seconds

**3.2.11 CLASS DurationDbIParameter**

**3.2.11.1** The DurationDbIParameter is a class that can be used to instantiate a duration expressed as a real number of seconds  $\geq 0.0$ .

NOTE – The DurationDbIParameter class is a specialization of class AbstractParameter described in 3.2.2 above,

**3.2.11.2** The DurationDbIParameter class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-11.

**Table 3-11: Class DurationDbIParameter Parameters**

Parameter	Description	Data Type	Data Units
Value	The value of the parameter.	xsd:double $\geq 0$	Seconds

**3.2.12 CLASS TimeParameterA**

**3.2.12.1** The TimeParameterA is a class which can be used to instantiate a parameter with type CCSDS Time Code A.

NOTE – The TimeParameterA class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.12.2** The TimeParameterA class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-12.

**Table 3-12: Class TimeParameterA Parameters**

Parameter	Description	Data Type	Data Units
value	The value of the parameter.	xsd: string restricted to holding the time in CCSDS ASCII Time Code A format (reference [1])	UTC

### 3.2.13 CLASS TimeParameterB

**3.2.13.1** The TimeParameterB is a class that can be used to instantiate a parameter with type CCSDS Time Code B.

NOTE – The TimeParameterB class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.13.2** The TimeParameterB class, in addition to the parameters specified in table 3-2, shall contain parameters as specified in table 3-13.

**Table 3-13: Class TimeParameterB Parameters**

Parameter	Description	Data Type	Data Units
Value	The value of the parameter.	xsd: string restricted to holding the time in CCSDS ASCII Time Code B format (reference [1])	UTC

### 3.2.14 CLASS ABSTRACTEVENTTIME (ABSTRACT)

**3.2.14.1** The AbstractEventTime class shall be used to specialize the timing parameter for an Event.

**3.2.14.2** There shall be one and only one instantiation of a class specialized from the AbstractEventTime class for each instantiation of the AbstractEvent class.

**3.2.14.3** The AbstractEventTime class shall contain the parameters as specified in table 3-14.

**Table 3-14: Class AbstractEventTime Parameters**

Parameter	Parameter	Data Type	Data Units
epochTimeSystem	Optional Parameter. This parameter may be used to specify the Epoch Time System to which the time is relative.	xsd:string  Permitted values are registered in SANA (see annex A for further information).	n/a

**3.2.14.4** If no epochTimeSystem is specified, then additional metadata in the XML document shall specify the epoch time system to be applied to the time. How this is specified is outside the scope of this document.

**3.2.15 CLASS CcsdsAsciiTimeCodeA**

**3.2.15.1** The CcsdsAsciiTimeCodeA class shall be used to permit the instantiation of absolute timing information for an Event in which the format of the absolute time is expressed in CCSDS ASCII Time Code A format.

NOTE – The CcsdsAsciiTimeCodeA class is a specialization of the class AbstractEventTime class described in 3.2.2 above.

**3.2.15.2** The CcsdsAsciiTimeCodeA class, in addition to the parameters specified in table 3-14, shall contain the parameters as specified in table 3-15.

**Table 3-15: Class CcsdsAsciiTimeCodeA Parameters**

Parameter	Parameter	Data Type	Data Units
absoluteTime	The absolute time at which the event occurs.	xsd: string restricted to holding the time in CCSDS ASCII Time Code A format (reference [1])	n/a

**3.2.16 CLASS CcsdsAsciiTimeCodeB**

**3.2.16.1** The CcsdsAsciiTimeCodeB class shall be used to permit the instantiation of absolute timing information for an Event in which the format of the absolute time is expressed in CCSDS ASCII Time Code B format.

NOTE – The CcsdsAsciiTimeCodeB class is a specialization of the class AbstractEventTime class described in 3.2.2 above.

**3.2.16.2** The CcsdsAsciiTimeCodeB class, in addition to the parameters specified in table 3-14, shall contain the parameters as specified in table 3-16.

**Table 3-16: Class CcsdsAsciiTimeCodeB Parameters**

Parameter	Parameter	Data Type	Data Units
absoluteTime	The absolute time at which the event occurs.	xsd: string restricted to holding the time in CCSDS ASCII Time Code B format (reference [1])	UTC

### 3.2.17 CLASS RelativeEventTime

**3.2.17.1** The RelativeEventTime class shall be used to permit the specification of relative timing information for an Event to another event in the same file.

NOTE – The RelativeEventTime class is a specialization of the class AbstractEventTime class described in 3.2.2 above.

**3.2.17.2** The RelativeEventTime class, in addition to the parameters specified in table 3-14, shall contain the parameters as specified in table 3-17.

**Table 3-17: Class RelativeEventTime Parameters**

Parameter	Parameter	Data Type	Data Units
relativeTime	The relative time at which the event occurs.	xsd:double	seconds
otherEventIdRef	The identifier (see table 3-1) of the event to which the time of this event is relative.	xsd:IDREF	n/a

NOTE – The xsd:IDREF type is used for an attribute that references an ID. All attributes of type xsd:IDREF reference an xsd:ID in the same XML document. Like ID, an xsd:IDREF value will be an NCName.

### 3.2.18 CLASS RelativeToTimeA

**3.2.18.1** The RelativeToTimeA class shall be used to permit the specification of relative timing information for an Event to an absolute time expressed in CCSDS ASCII Time Code A format.

NOTE – RelativeToTimeA class is a specialization of the class AbstractEventTime class described in 3.2.2 above.

**3.2.18.2** The RelativeToTimeA class, in addition to the parameters specified in table 3-14, shall contain the parameters as specified in table 3-18.

**Table 3-18: Class RelativeToTimeA Parameters**

Parameter	Parameter	Data Type	Data Units
relativeTime	The relative time at which the event occurs.	xsd:double	Seconds
absoluteTime	Optional Parameter. This may be used to specify the absolute time to which the event time is relative.	xsd: string restricted to holding the time in CCSDS ASCII Time Code A format (reference [1])	n/a

**3.2.18.3** If no absoluteTime parameter is specified, then additional metadata in the XML document shall specify the absolute time to be applied. How this is specified is outside the scope of this document.

**3.2.19 CLASS RelativeToTimeB**

**3.2.19.1** The RelativeToTimeB class shall be used to permit the specification of relative timing information for an Event to an absolute time expressed in CCSDS ASCII Time Code A format.

NOTE – RelativeToTimeB class is a specialization of the class AbstractEventTime class described in 3.2.2 above.

**3.2.19.2** The RelativeToTimeB class, in addition to the parameters specified in table 3-14, shall contain the parameters as specified in table 3-19.

**Table 3-19: Class RelativeToTimeB Parameters**

Parameter	Parameter	Data Type	Data Units
relativeTime	The relative time at which the event occurs.	xsd:double	Seconds
absoluteTime	Optional Parameter. This may be used to specify the absolute time to which the event time is relative.	xsd: string restricted to holding the time in CCSDS ASCII Time Code B format (reference [1])	n/a

**3.2.19.3** If no absoluteTime parameter is specified, then additional metadata in the XML document shall specify the absolute time to be applied. How this is specified is outside the scope of this document.

### **3.2.20 CLASS TimeParameter**

**3.2.20.1** The TimeParameter is a class that can be used to instantiate time parameters derived from the AbstractEventTime class (see 3.2.14).

NOTE – The TimeParameter class is a specialization of class AbstractParameter described in 3.2.2.

**3.2.20.2** Each instance of the TimeParameter class shall contain one and only one instance of a time parameter specialized from the AbstractEventTime class (see 3.2.14).

**3.2.20.3** The TimeParameter class shall contain the parameters as specified in table 3-14.

NOTE – The parameters for the TimeParameter class are specified in table 3-14. There are no additional parameters.

## ANNEX A

### SECURITY, SANA, AND PATENT CONSIDERATIONS

#### (INFORMATIVE)

##### A1 SECURITY CONSIDERATIONS

Security considerations are expected to be addressed in the specifications that reference the data structures contained in this document.

##### A2 SANA CONSIDERATIONS

###### A2.1 GENERAL

The recommendations of this document rely on the SANA registries described below. Assignments in these registries, in conformance with the policies identified, will be available at the SANA registry Web site: <https://sanaregistry.org>.

###### A2.2 REGISTRY FOR TYPE PARAMETER

The type of events that a using recommendations defines are registered with SANA in an event types registry.

The information registered is:

- a) the name given to the event type;
- b) a description of the event type;
- c) the CCSDS document that is making use of the AED, providing the event name and description.

As this is a CCSDS global registry, updates are in accordance with reference [3], subsection 3.11 (b), that is, under the authority of CESG or a body delegated by CESG.

The event types registry is located at

[https://sanaregistry.org/r/event\\_types](https://sanaregistry.org/r/event_types)

### A2.3 REGISTRY FOR EXTENDED PARAMETER

Parameters derived from the extended parameter class require an additional schema or schema files for validation (see 3.2.9). The contents of such schemas are agency specific and outside the scope of this Recommended Practice. However, to assist with inter-agency inter-operations, the agency-specific schemas are registered.

The information registered is:

- a) Status – the status of the extended parameter;
- b) Organization – the agency that is registering the extended parameter;

NOTE – This should be the same as the agency name in the SANA Organizations registry.

- c) Point of Contact – the responsible party of the agency registering the particular agency schema;

NOTE – This should be one of the names found in the SANA Contacts registry.

- d) Schema – the name and version of the CCSDS Schema that the agency-specific schema is extending:

#### NOTES

- 1 For extensions relative to the schemas defined by this Recommended Practice, the CSSM Schema names and version number can be found in B2.
- 2 Multiple version numbers are supplied if the extension is compatible with and applies to multiple versions of the SchemaCsmAbstractEvent schema;
- e) Name – the name of the parameter derived from the ExtendedParameter class.
- f) Description – a brief description of the type of extension being afforded by the agency-specific schema;
- g) Applicable Start Date – the date when an agency recognizes and processes its particular extended parameter conforming to the registered schema;
- h) Applicable End Date – the date when an agency will no longer recognize and process its particular extended parameter conforming to the registered schema.

NOTE – This can have the value of ‘indefinite’ if the agency has no plans for ending implementation of the agency-specific extended parameter.

- i) References – specification of any references that may be relevant to the extended parameter. May be left empty.
- j) Extended Schema File(s) – the file(s) that contain the XML schema for the agency-specific extended parameter.

In accordance with reference [3], section 3.11 (c), updates to this registry are at the discretion of CCSDS member agencies or registered organizations, via the registered agency or organization representative. The extended parameter registry is located at

[https://sanaregistry.org/r/agency\\_specific\\_schemas/](https://sanaregistry.org/r/agency_specific_schemas/)

#### **A2.4 REGISTRY FOR `epochTimeSystem`**

The values for the ‘`epochTimeSystem`’ parameter (`AbstractEventTime` class—see table 3-14) are those listed in the ‘Name’ field of the CCSDS Time Systems registry. The CCSDS Time Systems registry is located at

[https://sanaregistry.org/r/time\\_systems/](https://sanaregistry.org/r/time_systems/)

In accordance with reference [3], section 3.18, updates to this registry are at the discretion of a CCSDS Expert Group.

#### **A2.5 USE OF UNREGISTERED VALUES**

Only values that have been registered should be used for the `epochTimeSystem` and `type` parameters. Unregistered values for the `epochTimeSystem` and `type` parameters may be used. If unregistered values are used they should be prefixed with the string ‘UNR::’.

#### **NOTES**

- 1 ‘UNR::’ indicates an unregistered value;
- 2 this helps eliminate potential confusion in a multi-agency cross support context;
- 3 use of unregistered values is not recommended and should be avoided if possible.

#### **A3 PATENT CONSIDERATIONS**

No patent rights are known to adhere to any of the specifications of the Recommended Practice.

## ANNEX B

# XML SCHEMA ORGANIZATION AND PACKAGING FOR THE ABSTRACT EVENT CLASSES

### (INFORMATIVE)

#### B1 PURPOSE

This annex provides an informative description of XML schema organization and packaging for the Abstract Event classes as defined in 3.1.2.

#### B2 SCHEMA ORGANIZATION AND PACKAGING

The normative Abstract Event schema types and global elements are contained in the file '902x13m1-AbsEvnt.xsd'.

The Abstract Event types and global elements are registered in the 'urn:ccsds:schema:csm:1.0.0' name space.

The Abstract Parameter schema includes the following schemas;

- a) 902x13m1-AbsEvnt-AbsParm.xsd
  - Types and global elements in this schema are registered in the 'urn:ccsds:schema:csm:1.0.0' name space.
- 902x13m1-AbsEvnt-TimCde.xsd
  - Types and global elements in this schema are registered in the 'urn:ccsds:schema:csm:1.0.0' name space.

The source of the following schema files,

- 902x13m1-AbsEvnt.xsd,
- 902x13m1-AbsEvnt-AbsParm.xsd, and
- 902x13m1-AbsEvnt-TimCde.xsd.

is the SANA SCCS-SM Information Entity XML Schemas registry:

[https://sanaregistry.org/r/service\\_management\\_xml\\_schemas](https://sanaregistry.org/r/service_management_xml_schemas)

**ANNEX C**

**INFORMATIVE REFERENCES**

**(INFORMATIVE)**

- [C1] *Space Communication Cross Support—Service Management—Operations Concept*. Issue 1-S. Report Concerning Space Data System Standards (Historical), CCSDS 910.14-G-1-S. Washington, D.C.: CCSDS, (May 2011) June 2017.

**ANNEX D****ABBREVIATIONS AND ACRONYMS****(INFORMATIVE)**

<u>Term</u>	<u>Meaning</u>
ASCII	American Standard Code for Information Interchange
CCSDS	Consultative Committee for Space Data Systems
CESG	CCSDS Engineering Steering Group
CSSM	Cross Support Service Management
ICD	interface control document
n/a	not applicable
NCName	non-colonized name
OMG	Object Management Group
SANA	Space Assigned Numbers Authority
SCCS-SM	Space Communication Cross Support Service Management
UML	Unified Modeling Language
UTC	Coordinated Universal Time
W3C	World Wide Web Consortium
XML	eXtensible Markup Language