

**Draft Recommendation for
Space Data System Standards**

**SIMPLE SCHEDULE
FORMAT
SPECIFICATION**

DRAFT RECOMMENDED STANDARD

CCSDS 902.1-R-2

RED BOOK
May 2017



CCSDS

The Consultative Committee for Space Data Systems

**Draft Recommendation for
Space Data System Standards**

**SIMPLE SCHEDULE
FORMAT
SPECIFICATION**

DRAFT RECOMMENDED STANDARD

CCSDS 902.1-R-2

RED BOOK

May 2017

AUTHORITY

Issue:	Red Book, Issue 2
Date:	May 2017
Location:	Not Applicable

(WHEN THIS RECOMMENDED STANDARD IS FINALIZED, IT WILL CONTAIN THE FOLLOWING STATEMENT OF AUTHORITY:)

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 e-mail address below.

This document is published and maintained by:

CCSDS Secretariat
National Aeronautics and Space Administration
Washington, DC, USA
E-mail: secretariat@mailman.ccsds.org

STATEMENT OF INTENT

(WHEN THIS RECOMMENDED STANDARD IS FINALIZED, IT WILL CONTAIN THE FOLLOWING 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 **Recommended Standards** and are not considered binding on any Agency.

This **Recommended Standard** is issued by, and represents the consensus of, the CCSDS members. Endorsement of this **Recommendation** is entirely voluntary. Endorsement, however, indicates the following understandings:

- o Whenever a member establishes a CCSDS-related **standard**, this **standard** will be in accord with the relevant **Recommended Standard**. Establishing such a **standard** does not preclude other provisions which a member may develop.
- o Whenever a member establishes a CCSDS-related **standard**, that member will provide other CCSDS members with the following information:
 - The **standard** itself.
 - The anticipated date of initial operational capability.
 - The anticipated duration of operational service.
- o Specific service arrangements shall be made via memoranda of agreement. Neither this **Recommended Standard** nor any ensuing **standard** is a substitute for a memorandum of agreement.

No later than five years from its date of issuance, this **Recommended Standard** 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 Standard** is issued, existing CCSDS-related member standards and implementations are not negated or deemed to be non-CCSDS compatible. It is the responsibility of each member to determine when such standards or implementations are to be modified. Each member is, however, strongly encouraged to direct planning for its new standards and implementations towards the later version of the Recommended Standard.

FOREWORD

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This Recommended Standard 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 e-mail 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 Federal Science Policy Office (BFSP0)/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.
- Chinese 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.
- 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.
- 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.

PREFACE

This document is a draft CCSDS Recommended Standard. Its 'Red Book' status indicates that the CCSDS believes the document to be technically mature and has released it for formal review by appropriate technical organizations. As such, its technical contents are not stable, and several iterations of it may occur in response to comments received during the review process.

Implementers are cautioned **not** to fabricate any final equipment in accordance with this document's technical content.

Recipients of this draft are invited to submit, with their comments, notification of any relevant patent rights of which they are aware and to provide supporting documentation.

DOCUMENT CONTROL

Document	Title	Date	Status
CCSDS 902.1-R-1	Simple Schedule Format Specification, Draft Recommended Standard, Issue 1	September 2014	Original draft issue, superseded
CCSDS 902.1-R-2	Simple Schedule Format Specification, Draft Recommended Standard, Issue 2	May 2017	Current draft: incorporates changes in response to review comments on the original draft issue as well as subsequent working group discussions (note).

NOTE – Changes from the original draft issue are too numerous to permit markup.

CONTENTS

<u>Section</u>	<u>Page</u>
1 INTRODUCTION	1-1
1.1 PURPOSE AND SCOPE.....	1-1
1.2 APPLICABILITY.....	1-2
1.3 RATIONALE.....	1-3
1.4 DOCUMENT STRUCTURE.....	1-5
1.5 DEFINITIONS.....	1-5
1.6 NOMENCLATURE.....	1-6
1.7 CONVENTIONS—THE UNIFIED MODELING LANGUAGE.....	1-6
1.8 REFERENCES.....	1-7
2 OVERVIEW	2-1
2.1 GENERAL.....	2-1
2.2 SIMPLE SCHEDULE.....	2-1
2.3 MAPPING TO W3C XML SCHEMA.....	2-1
3 SIMPLE SCHEDULE	3-1
3.1 GENERAL.....	3-1
3.2 SIMPLE SCHEDULE CONTENT/STRUCTURE.....	3-1
3.3 PERMITTED COMBINATIONS OF PARAMETERS SIMPLESCHEDULEHEADER—STATUS, SCHEDULEDPACKAGE—USER, SERVICEINFO—SERVICETYPE, AND SCHEDULEDACTIVITY— ACTIVITYSTATUS.....	3-14
3.4 ORDERING OF SCHEDULED PACKAGES.....	3-15
3.5 USAGE OF SIMPLE SCHEDULE FOR SPECIFYING APERTURE AVAILABLE FREE TIME.....	3-17
3.6 XML SCHEMAS FOR THE SIMPLE SCHEDULE.....	3-19
ANNEX A IMPLEMENTATION CONFORMANCE STATEMENT (ICS) PROFORMA (NORMATIVE)	A-1
ANNEX B SERVICE MANAGEMENT INFORMATION ENTITY (NORMATIVE)	B-1
ANNEX C SERVICE MANAGEMENT ABSTRACT PARAMETER CLASS (NORMATIVE)	C-1
ANNEX D SECURITY, SANA, AND PATENT CONSIDERATIONS (INFORMATIVE)	D-1
ANNEX E XML SCHEMA FOR THE SERVICE MANAGEMENT INFORMATION ENTITY HEADER (INFORMATIVE)	E-1

CONTENTS

<u>Section</u>	<u>Page</u>
ANNEX F XML SCHEMA FOR THE SERVICE MANAGEMENT ABSTRACT PARAMETER CLASS (INFORMATIVE)	F-1
ANNEX G XML SCHEMA FOR THE SIMPLE SCHEDULE FORMAT (INFORMATIVE)	G-1
ANNEX H ABBREVIATIONS AND ACRONYMS (INFORMATIVE)	H-1
ANNEX I INFORMATIVE REFERENCES (INFORMATIVE)	I-1
ANNEX J FREQUENCY BAND DEFINITIONS (INFORMATIVE)	J-1

Figure

1-1 Simple Schedule Format in the Context of Space Communication Cross Support Service Management.....	1-2
3-1 Simple Schedule Class Diagram.....	3-1
3-2 Example of Ordering of Scheduled Packages in Simple Schedule	3-16
3-3 Schematic of Schedule to Illustrate Use of Simple Schedule for Aperture Free Time	3-18
B-1 Service Management Information Entity Class Diagram.....	B-1
C-1 Abstract Parameter Class Diagram.....	C-1

Table

3-1 Class SimpleSchedule Parameters.....	3-2
3-2 Class SimpleScheduleHeader Parameters	3-4
3-3 Class SimpleScheduleHeader Use of Class SrvMgtHeader Parameters	3-5
3-4 Class ScheduledPackage Parameters.....	3-6
3-5 Class ScheduledPkgAssn Parameters	3-8
3-6 Class ServicePackageXRef Parameters.....	3-9
3-7 Class ScheduledActivity Parameters	3-10
3-8 Class ServiceInfo Parameters	3-13
3-9 Permitted Combinations of Parameters SimpleScheduleHeader—status, ScheduledPackage—user, ServiceInfo—serviceType and ScheduledActivity— activityStatus.....	3-15
A-1 Class AbstractParameter Instantiated for Class SimpleScheduleHeader	A-12
A-2 Class AbstractParameter Instantiated for Class ScheduledPackage.....	A-12
A-3 Class AbstractParameter Instantiated for Class ScheduledPkgAssn.....	A-13
A-4 Class AbstractParameter Instantiated for Class ScheduledActivity	A-13
A-5 Class AbstractParameter Instantiated for Class ServiceInfo	A-13
B-1 Class SrvMgtInfoEntity Parameters	B-2
B-2 Class SrvMgtHeader Parameters	B-3
C-1 Class AbstractParameter Parameters	C-2
C-2 Class BooleanParameter Parameters	C-2

CONTENTS

<u>Table</u>	<u>Page</u>
C-3 Class UnsignedIntParameter Parameters	C-3
C-4 Class IntParameter Parameters	C-3
C-5 Class FloatParameter Parameters	C-4
C-6 Class DoubleParameter Parameters	C-4
C-7 Class StringParameter Parameters	C-5
C-8 Class ExtendedParameter Parameters.....	C-6
C-9 Class DurationParameter Parameters	C-6
C-10 Class TimeParameterA Parameters	C-7
C-11 Class TimeParameterB Parameters.....	C-7

1 INTRODUCTION

1.1 PURPOSE AND SCOPE

1.1.1 PURPOSE

This Simple Schedule Format Recommended Standard specifies a standard format for use in transferring scheduling information related to apertures at ground stations and/or relay satellites between space agencies and commercial or governmental spacecraft operators. Such exchanges are used in:

- a) mission design, e.g., in investigating the feasibility of a mission with respect to its uplink/downlink requirements and the availability of suitable apertures;
- b) mission planning, e.g., to ensure that there are sufficient aperture resources available to carry out the planned operations;
- c) mission operations, e.g., to
 - 1) inform the spacecraft operator of the actual allocation of apertures to their mission;
 - 2) inform the spacecraft operator of any unused times of the apertures;
 - 3) return the results of a query about apertures.

1.1.2 SCOPE

The scope of this Recommended Standard is limited to the exchange of Simple Schedule information. The contents of the simple schedule format described here were derived from the analysis of the standard schedule formats used by a number of space agencies/parts of space agencies. This analysis led to the identification of a 'core' content that was (almost) universally present in all schedules analyzed. It was, however, also identified that there was a need to permit some extension to the core content to cover some use cases. Consequently an extension mechanism has been provided that can be used to add further information in the eventuality that the core content is insufficient for the required purpose.

Definition of the extended content for any specific case is outside the scope of this Recommended Standard and should be specified in an Interface Control Document (ICD) agreed upon among the involved parties.

While this Recommended Standard has been developed within the scope of the Space Communication Cross Support Service Management (SCCS SM) (references [I1] and [I2]) activity, it is intended that the Simple Schedule Format can be used to exchange schedule information where no other use of SCCS SM is made. In view of this, schedule data that is only of relevance when used in the scope of SCCS SM has been made optional. Thus there is no limiting dependency on Service Management in the Simple Schedule Format specification.

Figure 1-1 puts the Simple Schedule Format Specification into context with the various standards that together form the Space Communication Cross Support Service Management.

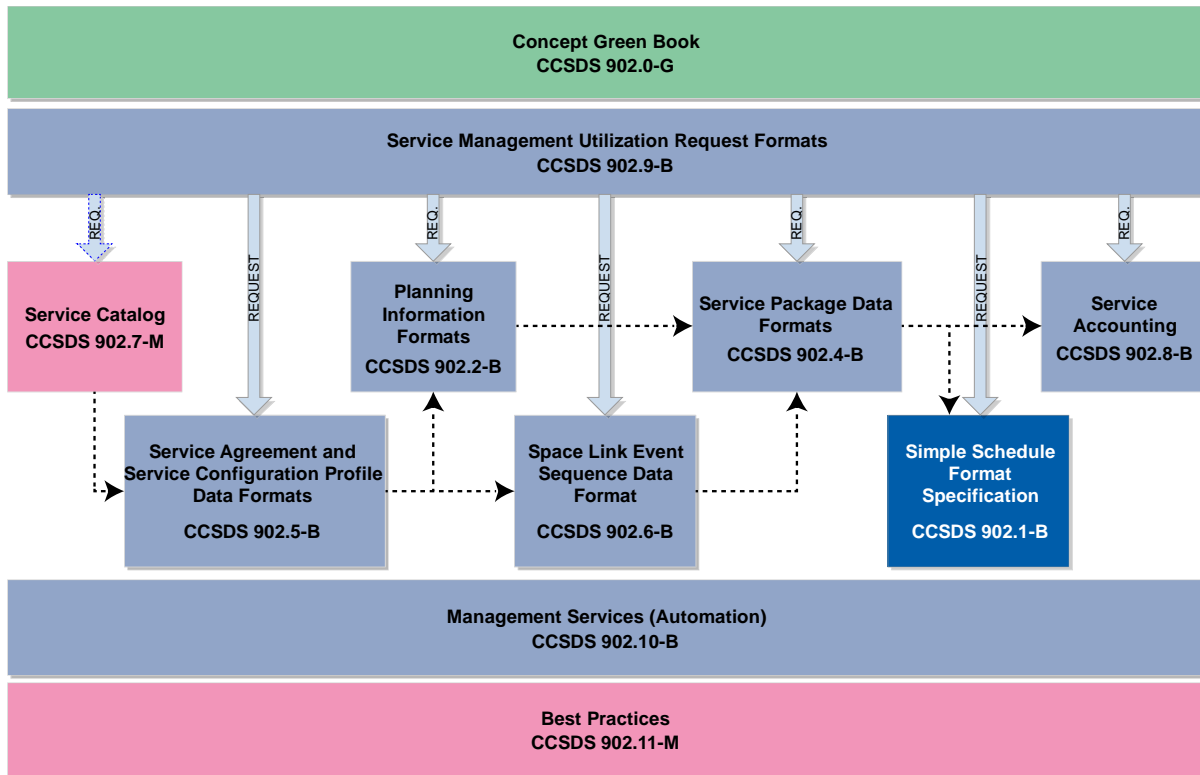


Figure 1-1: Simple Schedule Format in the Context of Space Communication Cross Support Service Management

1.2 APPLICABILITY

This Recommended Standard is applicable only to the schedule format and content, but not to its transmission. With respect to the transmission of the Simple Schedule between agencies and operators there are two scenarios:

- a) The first involves exchange of the Simple Schedule within the context of SCCS SM.
- b) The second involves exchange of the Simple Schedule outside the scope of SCCS SM. Here the mechanism by which the Simple Schedule is transmitted is outside of the scope of this document and should be specified in an ICD agreed by the parties involved.

1.3 RATIONALE

1.3.1 GENERAL

The primary goal of CCSDS is to increase the level of interoperability among agencies. This Recommended Standard furthers that goal by establishing the means to exchange schedule information where most cross support activity occurs: between the tracking stations or ground data handling systems of various agencies and the mission-specific components of a mission ground system.

The use cases described in the following subsections were considered in deriving this Recommended Standard.

1.3.2 USE CASES

1.3.2.1 Mission Design

During the design phase of a mission it is often desirable to be able to obtain some idea of aperture availability so that the feasibility of obtaining the required uplink and downlink bandwidth can be assessed. Typically, during the mission design phase, the information required relates to periods several years in the future. Currently there is no standardized way for obtaining this information from different agencies.

While this Recommended Standard does not address the mechanism by which the required schedule information can be requested (this possibly being covered in the negotiations involved in establishing the service agreement), it provides a standard format in which the possible availability of apertures can be returned.

1.3.2.2 Mission Planning

In the mission planning use case the provider agency provides the user agency with a schedule that gives the user agency the information it requires to plan its spacecraft operations with respect to the availability of uplink/downlink resources. The means by which the provider agency determines the availability for the user agency can be either:

- a) derived from a request from the user agency;

NOTE – The contents and format of such a request are outside the scope of this Recommended Standard (although this may be addressed elsewhere in the SCCS SM Recommended Standard at a later date).

- b) derived from what is specified in the service agreement between the agencies.

The planning process will typically go through a number of cycles, starting months before the actual activities are expected to take place and being finalized a few days before execution. During these planning cycles the status of the schedule can change with the initial

exchanges being of a ‘provisional’ schedule that is likely to change. The final schedule provided is typically considered as ‘operational’ and can be expected to provide the actual schedule that will be executed, subject to unexpected events such as equipment failure or spacecraft emergencies.

1.3.2.3 Dissemination of Provider Cross Support Service System (CSSS)¹ Schedules

It is desirable that a standardized approach be available for disseminating Provider CSSS schedules, both between agencies and also within agencies. This Recommended Standard aims to provide that. There are, however, differences in policies between agencies with regard to the contents of distributed schedules. Some agencies provide complete schedules to all users, containing information regarding all spacecraft supported by the agency (and possibly unallocated time as well). At the other extreme, other agencies provide only details regarding a spacecraft’s allocation to the operators of that mission.

In view of this, this Recommended Standard has been designed to be flexible enough to be able to cope with the different policies of the various agencies with respect to the information they wish to provide in disseminating their schedules.

1.3.2.4 Dissemination of Unallocated Time Schedules

Some agencies publish a list of unallocated time of the apertures in their network. This information can be used by missions to see if it is possible to request additional coverage by using resources during periods when they would otherwise be unused.

Although not addressing the method by which the usage of unallocated resources can be requested, this Recommended Standard can be used to publish a Provider CSSS unallocated time schedule. It can also be used to inform a mission of the result of a request to use an unallocated resource.

1.3.2.5 Response to Schedule Query

The final use case of the Simple Schedule Format is to return schedule information resulting from a query about resource availability. While the contents and format of such a query are outside the scope of this Recommended Standard, the Simple Schedule Format provides a standard mechanism by which the information can be returned to the requestor. The ability to submit queries and the contents of these will depend on the capabilities and policies of the agency, but, for example, providing the agencies policy permits, it could be used by a mission to find unallocated aperture times which they could then request the use of.

¹ A Provider Cross Support Service System is often colloquially referred to as a Network.

1.4 DOCUMENT STRUCTURE

This document is organized as follows:

- a) Section 1 provides the purpose, scope, applicability, and rationale of this Recommended Standard 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 will have 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 CCSDS-recommended Simple Schedule Format.
- c) Section 3 provides details about the structure and content of the Simple Schedule Format and specifies the normative files that define the XML schemas that govern the content of the XML instance documents that carry Simple Schedules.
- d) Annex A provides the normative Implementation Conformance Statement (ICS) proforma.
- e) Annex B defines the standard Service Management information entity.
- f) Annex C defines the Service Management parameter class.
- g) Annex D discusses security, SANA, and patent considerations.
- h) Annex E provides an informative listing of the normative XML schema that realizes the classes that comprise the abstract Service Management Information.
- i) Annex F provides an informative listing of the normative XML schema that realizes the class that comprises the abstract Service Management Parameter.
- j) Annex G provides an informative listing of the normative XML schema that realizes the classes that comprise the Simple Schedule Format.
- k) Annex H contains a list of Acronyms applicable to the Simple Schedule Format.
- l) Annex I is a list of informative references.
- m) Annex J contains the definition of the frequency bands used in the Recommended Standard.

1.5 DEFINITIONS

For the purposes of this document, the following definitions apply:

- a) the word ‘agencies’ may also be construed as meaning ‘satellite operators’ or ‘satellite service providers’;
- b) the notation ‘n/a’ signifies ‘not applicable’.

1.6 NOMENCLATURE

1.6.1 NORMATIVE TEXT

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

- 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—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 [4]). An overview of UML diagramming conventions can be found in annex E of reference [1].

Within the document use is made only of class diagrams. A UML class diagram describes the structure of a message, its parts, and how those parts interrelate. A UML class, represented in the diagram as a box, represents a data set. Class diagram conventions include composition, generalization, multiplicity, and constraints. Enumeration notation is also used but only when it is involved in a composition constraint.

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] *Space Communication Cross Support—Service Management—Service Specification*. Issue 1. Recommendation for Space Data System Standards (Blue Book), CCSDS 910.11-B-1. Washington, D.C.: CCSDS, August 2009.
- [2] “CCSDS-910.11-B-1_XML_schemas.”
http://public.ccsds.org/publications/archive/CCSDS-910.11-B-1_XML_schemas.zip.
- [3] *Time Code Formats*. Issue 4. Recommendation for Space Data System Standards (Blue Book), CCSDS 301.0-B-4. Washington, D.C.: CCSDS, November 2010.
- [4] *Unified Modeling Language (UML)*. Version 2.4.1. Needham, Massachusetts: Object Management Group, August 2011.
- [5] *CCSDS SANA Registry Management Policy*. Issue 1. CCSDS Record (Yellow Book), CCSDS 313.1-Y-1. Washington, D.C.: CCSDS, May 2016.

2 OVERVIEW

2.1 GENERAL

This section provides a high-level overview of the CCSDS-recommended Simple Schedule Format, which is designed to facilitate standardized exchanges of aperture schedule information between space agencies.

2.2 SIMPLE SCHEDULE

A Simple Schedule file is XML formatted. The format of the Simple Schedule file is suitable for automated interaction and/or (by means of a suitable XML viewer) human interaction.

Data in the Simple Schedule is either mandatory, in which case suitable values must be present, or optional, in which case values may be present or not. In addition it is possible to extend the contents of the Simple Schedule 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 Standard includes the specification of a mapping to World Wide Web Consortium (W3C) extensible Markup Language (XML) schema. The normative mapping of this Recommended Standard to XML W3C schemas is a virtual annex to this Recommended Standard and is contained in a stand-alone set of schema files (reference [2]).

NOTE – The XML schema has been elaborated on the basis of the mapping guidelines described in reference [I1].

3 SIMPLE SCHEDULE

3.1 GENERAL

The Simple Schedule shall consist of digital data exchanged in the form of a file containing an XML instance document.

NOTE – Subsection 3.6 identifies the files that contain the normative Simple Schedule XML schemas. The naming conventions for the files containing the Simple Schedule XML instance documents are outside of the scope of this recommended standard.

3.2 SIMPLE SCHEDULE CONTENT/STRUCTURE

3.2.1 OVERVIEW

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

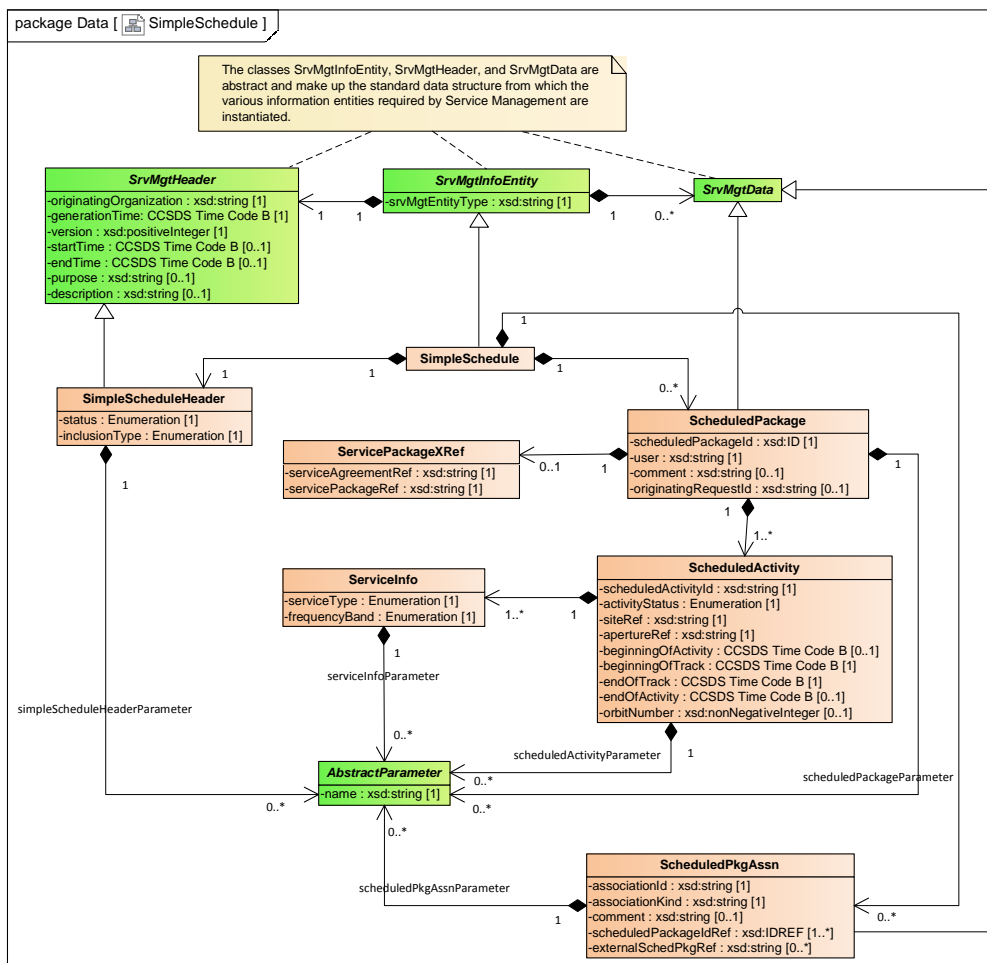


Figure 3-1: Simple Schedule Class Diagram

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

NOTES

- 1 In the following tables, data types specified as ‘xsd:nnn’ are XML schema types. Thus, for example, ‘xsd:string’ indicates that the type is an XML schema string.
- 2 In general the length of strings in this Recommended Standard is unbounded; i.e., no length is specified (in practice there is obviously the constraint of the maximum string length supported). However, in a number of cases, in the various SANA registries used, a length for string values is specified. For these cases the effective restriction on string length is the value specified in the SANA registry definition.

3.2.2 CLASS SimpleSchedule

3.2.2.1 The SimpleSchedule class is mandatory and shall constitute the Simple Schedule information entity.

NOTE – The SimpleSchedule class is a specialization of class SrvMgtInfoEntity described in annex B, and the generic description of the parameters is given in there.

3.2.2.2 The use of the parameters in the SrvMgtInfoEntity in the scope of the Simple Schedule shall be as specified in table 3-1.

Table 3-1: Class SimpleSchedule Parameters

Parameter	Description	Data Type	Data Units
srvMgtEntityType	(See table B-1.)	xsd:string For the Simple Schedule the only permitted value for this parameter is the string ‘SIMPLE_SCHEDULE’.	(See table B-1.)

3.2.3 CLASS SimpleScheduleHeader

3.2.3.1 The SimpleScheduleHeader class is mandatory and shall constitute the ‘header’ of the Simple Schedule.

NOTE – The SimpleScheduleHeader class is a specialization of class SrvMgtHeader described in annex B, and the generic description of the parameters is given there.

3.2.3.2 Each instance of the SimpleScheduleHeader class may have zero or more instances of parameter classes derived from the abstract AbstractParameter class described in annex C. Usage of any additional parameters so specified is not within the scope of this document and should be specified in an ICD between by the relevant parties.

3.2.3.3 The SimpleScheduleHeader class, in addition to the parameters in the abstract SrvMgtHeader class, described in annex B, shall contain parameters as specified in table 3-2.

3.2.3.4 The parameter `inclusionType` of the SimpleScheduleHeader shall be used to specify the type of inclusion rule that has been used to select the ScheduledPackages (see 3.2.4) that are contained in the schedule.

3.2.3.5 Two types of inclusion rules are supported and are discussed below; the following shall apply to both types:

- a) The start time of a ScheduledPackage (t_{ssp}) is defined to be the earliest `beginningOfActivity` or `beginningOfTrack` time of any ScheduledActivity (see 3.2.7) contained within that ScheduledPackage.
- b) The end time of a ScheduledPackage (t_{esp}) is defined to be the latest `endOfActivity` or `endOfTrack` time of any ScheduledActivity contained within that ScheduledPackage.

3.2.3.6 In the case of inclusion type OVERLAP_INCLUSION where the `startTime` and `endTime` specified in the SimpleScheduleHeader are respectively T_s and T_e , ScheduledPackages shall be included in the schedule where:

$$t_{esp} > T_s \text{ and } t_{ssp} < T_e$$

3.2.3.7 In the case of inclusion type START_INCLUSION, in the event that one or more ScheduledPackage are associated via a ScheduledPkgAssn (see 3.2.5), the start time of the associated ScheduledPackages (t_{sasp}) is defined to be the earliest ScheduledPackage time (t_{ssp}) of the associated ScheduledPackages; thus where the `startTime` and `endTime` specified in the SimpleScheduleHeader are respectively T_s and T_e , ScheduledPackages shall be included in the schedule where:

$$t_{ssp} \geq T_s \text{ and } t_{ssp} < T_e \text{ (ScheduledPackage is not associated with other ScheduledPackages)}$$

or

$$t_{sasp} \geq T_s \text{ and } t_{sasp} < T_e \text{ (ScheduledPackage associated with other ScheduledPackages).}$$

NOTES

- 1 It is implicit in the above definition of the inclusion type `START_INCLUSION` that if the criteria $t_{sasp} \geq T_s$ and $t_{sasp} < T_e$ are satisfied for a set of associated `ScheduledPackages` then all of the associated `ScheduledPackages` will appear in the schedule even though individual `ServicePackages` in the association may not satisfy the $t_{ssp} \geq T_s$ and $t_{ssp} < T_e$ criteria.
- 2 Similarly, if the criteria $t_{sasp} \geq T_s$ and $t_{sasp} < T_e$ is not satisfied for a set of associated `ScheduledPackages`, then none of the associated `ScheduledPackages` will appear in the schedule even though individual `ServicePackages` in the association may satisfy the $t_{ssp} \geq T_s$ and $t_{ssp} < T_e$ criteria.

Table 3-2: Class `SimpleScheduleHeader` Parameters

Parameter	Description	Data Type	Data Units
<code>status</code>	The status of the information entity.	Enumeration The permitted values are: – <code>TEST</code> —The schedule has been generated for test purposes only. – <code>PROVISIONAL</code> —The schedule contained is provisional and may still be subject to change. – <code>OPERATIONAL</code> —This is an operational schedule. – <code>OTHER</code> —This message has been generated for other purposes. In this case the optional parameter <code>purpose</code> can be used to further specify the purpose of the message.	n/a
<code>inclusionType</code>	The inclusion type rule that has been applied to the contained simple schedule.	Enumeration The permitted values are – <code>OVERLAP_INCLUSION</code> (see 3.2.3.6 for a description). – <code>START_INCLUSION</code> (see 3.2.3.7 for a description).	n/a

NOTE – In the context of the SimpleScheduleHeader class the usage of the SrvMgtHeader parameters in table 3-3 (as described in table B-2) should be noted.

Table 3-3: Class SimpleScheduleHeader Use of Class SrvMgtHeader Parameters

Parameter	Description	Data Type	Data Units
startTime	Mandatory parameter for the Simple Schedule. The start time of the simple schedule contained.	CCSDS ASCII Time Code B (reference [3])	UTC
endTime	Mandatory parameter for the Simple Schedule. The end time of the simple schedule contained.	CCSDS ASCII Time Code B (reference [3])	UTC

NOTE – Permitted combinations of parameters SimpleScheduleHeader—status, ScheduledPackage—user, ServiceInfo—serviceType, and ScheduledActivity—activityStatus are specified in 3.3.

3.2.4 CLASS ScheduledPackage

3.2.4.1 The ScheduledPackage class is optional and may be used to specify the information for service packages that have been scheduled in the time interval bounded by the SimpleSchedule parameters startTime and endTime.

NOTE – The ScheduledPackage class is a specialization of class SrvMgtData described in annex B. The abstract SrvMgtData class contains zero parameters.

3.2.4.2 There shall be zero or more instances of the ScheduledPackage class for each instance of the SimpleSchedule class.

3.2.4.3 There shall be one or more instances of the ScheduledActivityClass for each instance of the ScheduledPackage Class.

3.2.4.4 There shall be zero or one instances of the ServicePackageXRef class for each instance of the ScheduledPackage Class.

3.2.4.5 Each instance of the ScheduledPackage class may have zero or more instances of parameter classes derived from the abstract AbstractParameter class described in annex C. Usage of any additional parameters so specified is not within the scope of this document and should be specified in an ICD between the relevant parties.

3.2.4.6 The ScheduledPackage class shall extend the SrvMgtData class with the parameters specified in table 3-6.

3.2.4.7 The scheduledPackageID must be unique for every scheduled package in a schedule.

3.2.4.8 The user shall be one of

- a) a spacecraft (the typical case);
- b) PROVIDER-CSSS (indicating that the time is allocated to the Provider CSSS for tests, maintenance, upgrades, etc.); or
- c) UNALLOCATED (indicating that the aperture is unallocated during the specified time range).

NOTE – The case for zero instances of the ScheduledPackage is where a Simple Schedule has been returned as a result of a query. In this case it is possible that no ScheduledPackages are scheduled in the time range specified. As noted in 1.3.2.5 the contents and format of such a query are outside the scope of this Recommended Standard.

Table 3-4: Class ScheduledPackage Parameters

Parameter	Description	Data Type	Data Units
scheduledPackageId	An identifier that is unique for every ScheduledPackage in a schedule. NOTE – If a schedule is regenerated, then the scheduledPackageID for a particular Scheduled Package may change.	xsd:ID	n/a
user	The user of the scheduled package. This will typically be a spacecraft name, as specified in SANA, but may also be 'PROVIDER-CSSS' and 'UNALLOCATED', with the meanings indicated in the adjacent column.	xsd:string Permitted values are registered in SANA (see D2.3 for further information). The following strings are also permitted values with the meaning indicated: – UNALLOCATED—The time is unallocated. – PROVIDER-CSSS—The time is allocated to the Provider CSSS (e.g., for tests, maintenance, upgrades, etc.).	n/a
comment	Optional parameter: may be used for the provision of ad-hoc information.	xsd:string Exact use of this may be specified in an ICD between relevant parties.	n/a
originatingRequestID	Optional parameter: if present must echo the identifier from the request that caused the scheduled package to be generated.	xsd:string	n/a

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

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

NOTE – Permitted combinations of parameters `SimpleScheduleHeader—status`, `ScheduledPackage—user`, `ServiceInfo—serviceType`, and `ScheduledActivity—activityStatus` are specified in 3.3.

3.2.5 CLASS ScheduledPkgAssn

3.2.5.1 The `ScheduledPkgAssn` class is optional and may be used to specify associations between scheduled packages.

NOTES

- 1 This may be useful in the case where there may be dependencies between scheduled packages on two or more antennas (e.g., Delta-Differential One-way Range [Delta-DOR] measurements), or where there are two or more spacecraft using the same aperture at the same time (Multiple Spacecraft per Aperture [MSPA]).
- 2 The `ScheduledPkgAssn` class is a specialization of class `SrvMgtData` described in annex B. The abstract `SrvMgtData` class contains zero parameters.

3.2.5.2 There shall be zero or more instances of the `ScheduledPkgAssn` class for each instance of the `SimpleSchedule` class.

3.2.5.3 Each instance of the `ScheduledPkgAssn` class may have zero or more instances of parameter classes derived from the abstract `AbstractParameter` class described in annex C. Usage of any additional parameters so specified is not within the scope of this document and should be specified in an ICD between the relevant parties.

3.2.5.4 The `ScheduledPkgAssn` class shall extend the `SrvMgtData` class with the parameters specified in table 3-5.

Table 3-5: Class ScheduledPkgAssn Parameters

Parameter	Description	Data Type	Data Units
associationId	An identifier that uniquely identifies the association. This is assigned by the Provider CSSS. It should be noted that if a schedule is regenerated then the associationId for a particular scheduled package association may change	xsd:string	n/a
associationKind	The type of the association.	xsd:string Permitted values are registered in SANA (see D2.6 for further information).	n/a
comment	Optional parameter: may be used for the provision of ad hoc information.	xsd:string Exact use of this may be specified in an ICD between relevant parties.	n/a
scheduledPackageIdRef	A list of the scheduledPackageIds (see table 3-4) of the scheduled packages that are associated.	Array of xsd:IDREF	n/a
externalSchedPkgIdRef	Optional parameter: may be used to indicate association to scheduled packages that are not contained in the current file.	Array of xsd:string	n/a

3.2.5.5 The xsd:IDREF type shall be used for an attribute that references an ID:

- a) all attributes of type xsd:IDREF must reference an xsd:ID in the same XML document;
- b) as with ID, an xsd:IDREF value must be an NCName.

3.2.5.6 For a ScheduledPkgAssn to be valid it must refer to a least one scheduled package that is in the same xml document (file).

3.2.5.7 If it is required in a ScheduledPkgAssn to refer to scheduled packages that are not contained in the same xml document (e.g., not contained in the time range covered by the simple schedule contained in the file or, in the case for Delta-DOR, for example, from another Provider CSSS), then the appropriate reference(s) shall be placed in the externalSchedPkgIdRef parameter.

3.2.6 CLASS ServicePackageXRef

3.2.6.1 The ServicePackageXRef class is optional and may be used to map the contents of the Simple Schedule to the appropriate SCCS SM service packages.

3.2.6.2 There shall be zero or one instance of the ServicePackageXRef class for each instance of the ScheduledPackage class.

3.2.6.3 The ServicePackageXRef class shall include the parameters specified in table 3-6.

NOTE – The contents of the ServicePackageXRef class are relevant only when the Simple Schedule is used in the context of SCCS SM.

Table 3-6: Class ServicePackageXRef Parameters

Parameter	Description	Data Type	Data Units
serviceAgreementRef	The Service Agreement (as per CCSDS 910.11-B-1, reference [1]; see also reference [I2] for additional information) to which the following servicePackageRef applies.	xsd:string	n/a
servicePackageRef	Cross-reference to the service package (as per CCSDS 910.11-B-1, reference [1]; see also reference [I2] for additional information), where the aperture allocation is contained in the event that the aperture allocation was obtained via SCCS SM. NOTE – Several ScheduledActivity instances may have the same servicePackageRef since one service package can contain a number of aperture allocations.	xsd:string	n/a

3.2.7 CLASS ScheduledActivity

3.2.7.1 The ScheduledActivity class shall contain the details of the scheduled activity and shall include parameters specified in table 3-7.

3.2.7.2 The ScheduledActivity class shall include two time windows,

- a) the interval within which the aperture is allocated (optional);
- b) the interval within the first window within which the spacecraft is actually being tracked by the aperture.

3.2.7.3 There shall be at least one instance of the ScheduledActivity class for each instance of the ScheduledPackage class.

3.2.7.4 There shall be one or more instances of the ServiceInfo class for each instance of the ScheduledActivity Class.

3.2.7.5 Each instance of the ScheduledActivity class may have zero or more instances of parameter classes derived from the abstract AbstractParameter class described in annex C. Usage of any additional parameters so specified is not within the scope of this document and should be specified in an ICD between by the relevant parties.

Table 3-7: Class ScheduledActivity Parameters

Parameter	Description	Data Type	Data Units
scheduledActivityID	Identifier that is unique for every ScheduledActivity in a schedule. It should be noted that if a schedule is regenerated then the scheduledActivityID for a particular Scheduled Activity may change.	xsd:string	n/a
activityStatus	Used to indicate how firm the commitment to the activity is.	Enumeration The permitted values are: <ul style="list-style-type: none"> – COMMITTED—barring unforeseen circumstances the activity will be supported. – TENTATIVE—the activity is currently scheduled but may still be subject to change. In addition to the above, one of the following two values must be used when the ScheduledPackage—user parameter is UNALLOCATED and the ServiceInfo—serviceType is UNUSED. <ul style="list-style-type: none"> – AVAILABLE indicates that the aperture could be scheduled during the UNUSED time. – UNAVAILABLE indicates that the aperture is completely unavailable during the UNUSED time. 	n/a
siteRef	The site to which the scheduled activity applies.	xsd:string Permitted values are registered in SANA (see D2.4 for further information).	n/a

Parameter	Description	Data Type	Data Units
apertureRef	Identifier specifying which aperture is to be used. The apertureRef must uniquely identify the aperture at a particular site (i.e., ground station/ relay satellite) that is to be used.	xsd:string Permitted values are registered in SANA (see D2.4 for further information).	n/a
beginningOfActivity	Optional parameter: must be present if endOfActivity parameter is present. Time at which the allocation of the site/aperture starts. This is assumed to allow for all necessary setup activities, etc. Where available this time should be supplied subject to agency policy. If it is not possible to specify the actual value this parameter shall be omitted.	CCSDS ASCII Time Code B (reference [3])	UTC
beginningOfTrack	Time at which tracking the spacecraft is scheduled to start.	CCSDS ASCII Time Code B (reference [3])	UTC
endOfTrack	Time at which tracking the spacecraft is scheduled to end.	CCSDS ASCII Time Code B (reference [3])	UTC
endOfActivity	Optional parameter: must be present if beginningOfActivity parameter is present. Time at which the allocation of the site/aperture ends. This is assumed to allow for all necessary teardown activities, etc. Where available this time should be supplied subject to agency policy. If it is not possible to specify the actual value this parameter shall be omitted.	CCSDS ASCII Time Code B (reference [3])	UTC
orbitNumber	Optional parameter. The Orbit Number on which the beginningOfTrack occurs. If this is not relevant (e.g., deep space missions) or not available, this parameter shall be omitted.	xsd:nonNegativeInteger	n/a

NOTE – Permitted combinations of parameters SimpleScheduleHeader—status, ScheduledPackage—user, ServiceInfo—serviceType, and ScheduledActivity—activityStatus are specified in 3.3.

3.2.8 CLASS ServiceInfo

3.2.8.1 The ServiceInfo class shall be used to provide information about the type of operations to be carried out during a pass.

3.2.8.2 The ServiceInfo class shall use the parameters specified in table 3-8 to specify the type of service and frequency band to be used.

3.2.8.3 There shall be at least one instance of the ServiceInfo class for each instance of the ScheduledActivity class.

NOTE – Multiple instances of the ServiceInfo Class are required where more than one service is provided during a pass; e.g.,

- a) Telemetry and telecommanding services are provided;
- b) Telemetry is downlinked on both X- and S-band simultaneously.

3.2.8.4 Each instance of the ServiceInfo class may have zero or more instances of parameter classes derived from the abstract AbstractParameter class described in annex C. Usage of any additional parameters so specified is not within the scope of this document and should be specified in an ICD between by the relevant parties.

Table 3-8: Class ServiceInfo Parameters

Parameter	Description	Data Type	Data Units
serviceType	<p>The type of service that will be carried out during an activity.</p> <p>It should be noted that:</p> <ul style="list-style-type: none"> – APA-AZ/EL indicates Antenna Pointing Angles, Azimuth/Elevation; – APA-X/Y indicates Antenna Pointing Angles, X/Y; – VLBI indicates Very-Long-Baseline Interferometry. 	<p>Enumeration</p> <p>The permitted values are:</p> <ul style="list-style-type: none"> – APA-AZ/EL – APA-X/Y – DELTADOR – DOPPLER – OFFLINE-TM-RECORDING – OFFLINE-TM-PROVISION – RF-ONLY – RANGING – RESERVED – TBD – TELECOMMAND – TELEMETRY – TEST – UNUSED – VLBI 	n/a
frequencyBand	<p>The frequency band that will be used by the service. If the frequency band is not relevant the value N/A (not applicable) shall be used. (See annex J for a definition of the frequencies each band refers to.)</p>	<p>Enumeration</p> <p>The permitted values are:</p> <ul style="list-style-type: none"> – VHF – UHF – C-BAND – KA-BAND – KU-BAND – L-BAND – S-BAND – V-BAND – X-BAND – OPTICAL – N/A – ALL 	n/a

3.2.8.5 For the case of unused time it is possible (but not mandatory) to specify a frequency band. To specify the unused time on a particular band of an Aperture the following shall be specified:

- a) ScheduledPackage—user parameter set to UNALLOCATED;
- b) ScheduledActivity—siteRef parameter set as appropriate;
- c) ScheduledActivity—apertureRef parameter set as appropriate;
- d) ServiceInfo—serviceType parameter set to UNUSED;

- e) ServiceInfo—frequencyBand set to either
 - 1) the appropriate frequency band as contained in table 3-8above; or
 - 2) the value ALL if all frequency band supported by the aperture are UNUSED.

NOTE – Permitted combinations of parameters SimpleScheduleHeader—status, ScheduledPackage—user, ServiceInfo—serviceType, and ScheduledActivity—activityStatus are specified in 3.3.

3.2.9 CLASS AbstractParameter (ABSTRACT)

The AbstractParameter is an abstract class and is fully defined in annex C.

3.3 PERMITTED COMBINATIONS OF PARAMETERS SimpleScheduleHeader—status, ScheduledPackage—user, ServiceInfo—serviceType, AND ScheduledActivity—activityStatus

3.3.1 The permitted combinations of the following parameters shall be those specified in table 3-9:

- a) SimpleScheduleHeader—status;
- b) ScheduledPackage—user;
- c) ServiceInfo—serviceType;
- d) ScheduledActivity—activityStatus.

Table 3-9: Permitted Combinations of Parameters SimpleScheduleHeader—status, ScheduledPackage—user, ServiceInfo—serviceType and ScheduledActivity—activityStatus

SimpleScheduleHeader— status value	ScheduledPackage— user value	ServiceInfo— serviceType value	ScheduledActivity— activityStatus value
TEST	Any permitted value	Any permitted value	Any permitted value
PROVISIONAL	Spacecraft—Permitted values registered in SANA	Any permitted value	TENTATIVE
	UNALLOCATED	UNUSED	AVAILABLE UNAVAILABLE
	PROVIDER-CSSS	Any permitted value	TENTATIVE
OPERATIONAL	Spacecraft—Permitted values registered in SANA	Any permitted value	TENTATIVE COMMITTED
	UNALLOCATED	UNUSED	AVAILABLE UNAVAILABLE
	PROVIDER-CSSS	Any permitted value	TENTATIVE COMMITTED
OTHER	Any permitted value, although this may be further constrained in an ICD between relevant parties	Any permitted value, although this may be further constrained in an ICD between relevant parties	Any permitted value, although this may be further constrained in an ICD between relevant parties

3.4 ORDERING OF SCHEDULED PACKAGES

3.4.1 The ordering of ScheduledPackages in a file containing a Simple Schedule shall be by increasing order of the earliest beginningOfTrack times of all the ScheduledActivities contained in a ScheduledPackage.

3.4.2 In the event that two or more ScheduledPackages contain ScheduledActivities that have the same earliest beginningOfTrack times, the user value of ScheduledPackage shall be used as a secondary key in increasing alphabetic order.

NOTE – Figure 3-2 is an example of how the order of ScheduledPackages should appear. For clarity not all parameters of ScheduledActivities are shown.

```

ScheduledPackage
  scheduledPackageID    = Package-A
  user                  = Mission-A
  ScheduledActivity
    scheduledActivityID = Activity-A1
    beginningOfTrack    = 2020-123T12:00:00.000Z
    .
    .
  ScheduledActivity
    scheduledActivityID = Activity-A2
    beginningOfTrack    = 2020-123T12:30:00.000Z
    .
    .
ScheduledPackage
  scheduledPackageID    = Package-B
  user                  = Mission-B
  ScheduledActivity
    scheduledActivityID = Activity-B1
    beginningOfActivity  = 2020-123T12:15:00.000Z
    beginningOfTrack    = 2020-123T12:30:00.000Z
    .
    .
ScheduledPackage
  scheduledPackageID    = Package-C
  user                  = Mission-C
  ScheduledActivity
    scheduledActivityID = Activity-C1
    beginningOfActivity  = 2020-123T12:00:00.000Z
    beginningOfTrack    = 2020-123T12:30:00.000Z
    .
    .
  ScheduledActivity
    scheduledActivityID = Activity-C2
    beginningOfActivity  = 2020-123T12:10:00.000Z
    beginningOfTrack    = 2020-123T12:40:00.000Z
    .
    .
ScheduledPackage
  scheduledPackageID    = Package-D
  user                  = Mission-A
  ScheduledActivity
    scheduledActivityID = Activity-A3
    beginningOfTrack    = 2020-123T13:00:00.000Z
    .
    .
  ScheduledActivity
    scheduledActivityID = Activity-A4
    beginningOfTrack    = 2020-123T13:30:00.000Z
    .
    .

```

Figure 3-2: Example of Ordering of Scheduled Packages in Simple Schedule

3.5 USAGE OF SIMPLE SCHEDULE FOR SPECIFYING APERTURE AVAILABLE FREE TIME

3.5.1 DISCUSSION

Aperture Free Time, for the purposes of the Simple Schedule, is defined as those time periods outside of scheduled activities; following on from this,

- Aperture Available Free Time is aperture free time which is available for scheduling.
- Aperture Unavailable Free Time is aperture free time which is not available for scheduling.

The objective of using the simple schedule to disseminate Aperture Available Free Time is to provide a standard mechanism by which, subject to the Provider-CSSS policies, missions can be informed of time slots on apertures which are currently unused and are available for use. A detailed description of how Aperture Available Free Time can be specified is contained in 3.5.2.

It is possible for the PROVIDER-CSSS to disseminate a combined schedule that contains scheduled aperture time, i.e., time allocated to specific users, as well as Aperture Available Free Time.

The Aperture Available Free Time is specified as a UTC time range. It is not orbit- or spacecraft-specific. It should be noted that the Aperture Available Free Time may not all be available for tracking of a spacecraft; e.g., time may be required at the start and end of the time interval for station configuration, teardown, etc., and/or antenna slewing in the case of relay satellites.

Figure 3-3 indicates, diagrammatically, what is meant by Aperture Available Free Time. It should be noted that in this diagram it is assumed that all Aperture Free Time is available for use.

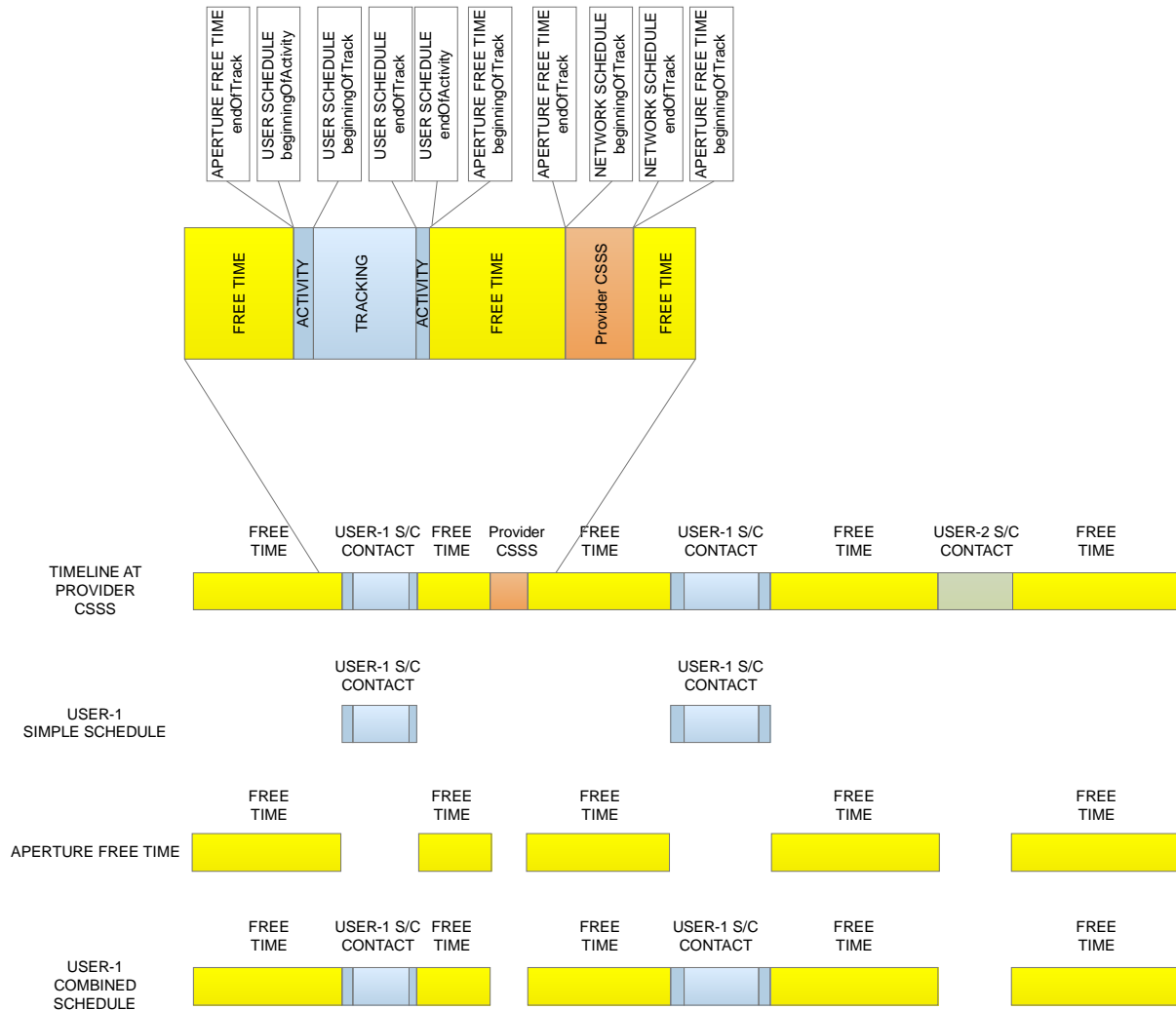


Figure 3-3: Schematic of Schedule to Illustrate Use of Simple Schedule for Aperture Free Time

3.5.2 SPECIFICATION OF APERTURE AVAILABLE FREE TIME

3.5.2.1 As of this issue, Aperture Available Free Time is not user-/spacecraft-specific.

3.5.2.2 When specifying Aperture Available Free Time in the Simple Schedule Format:

- a) the ScheduledPackage—user parameter shall be specified as ‘UNALLOCATED’;
- b) the ScheduledActivity—activityStatus parameter shall be specified as ‘AVAILABLE’;

- c) the ScheduledActivity—beginningOfTrack parameter shall be set to the UTC time at which the Aperture Available Free Time starts;
- d) the ScheduledActivity—endOfTrack parameter shall be set to the UTC time at which the Aperture Available Free Time ends;
- e) The ScheduledActivity—beginningOfActivity and endOfActivity parameters shall be omitted;
- f) the ServiceInfo—serviceType parameter shall be set to ‘UNUSED’;
- g) the ServiceInfo frequencyBand parameter shall be set to either
 - 1) ‘ALL’ indicating that all frequency bands supported by the aperture are available; or
 - 2) a specific frequency band (see table 3-8 for allowed values) indicating that the aperture is free to support that frequency band only.

3.6 XML SCHEMAS FOR THE SIMPLE SCHEDULE

Each Simple Schedule Information Entity shall conform to the XML schemas specified in the files ‘SchemaCsmSimpleSchedule-v1_0_0.xsd’, ‘SchemaCsmSmInfoEntityHeader-v1_0_0.xsd’, and ‘SchemaCsmCcsdsTimecodes-v1_0_0.xsd’. The normative source of these files shall be the SANA SCCS-SM Information Entity XML Schemas registry:

http://sanaregistry.org/r/service_management_xml_schemas/

NOTE – Annex G provides an informative description of the Simple Schedule XML schemas.

ANNEX A

IMPLEMENTATION CONFORMANCE STATEMENT (ICS) PROFORMA

(NORMATIVE)

A1 INTRODUCTION

A1.1 OVERVIEW

This annex provides the Implementation Conformance Statement (ICS) Requirements List (RL) for an implementation of the *Simple Schedule Format Specification* (CCSDS 902.1-R-2). The ICS for an implementation is generated by completing the RL in accordance with the instructions below. An implementation shall satisfy the mandatory conformance requirements referenced in the RL.

The RL in this annex is blank. An implementation's completed RL is called the ICS. The ICS states which capabilities and options have been implemented. The following can use the ICS:

- the implementer, as a checklist to reduce the risk of failure to conform to the standard through oversight;
- a supplier or potential acquirer of the implementation, as a detailed indication of the capabilities of the implementation, stated relative to the common basis for understanding provided by the standard ICS proforma;
- a user or potential user of the implementation, as a basis for initially checking the possibility of interworking with another implementation (it should be noted that, while interworking can never be guaranteed, failure to interwork can often be predicted from incompatible ICSes);
- a tester, as the basis for selecting appropriate tests against which to assess the claim for conformance of the implementation.

A1.2 ABBREVIATIONS AND CONVENTIONS

A1.2.1 General

The RL consists of information in tabular form. The status of features is indicated using the abbreviations and conventions described below.

A1.2.2 Item Column

The item column contains sequential numbers for items in the table.

A1.2.3 Feature Column

The feature column contains a brief descriptive name for a feature. It implicitly means ‘Is this feature supported by the implementation?’

NOTE – The features itemized in the RL are elements of the Simple Schedule Format. Therefore support for a mandatory feature indicates that a generated file will include that feature, and support for an optional feature indicates that generated files can include that feature.

A1.2.4 Class Column/Parameters

The Class/Parameters column contains, where applicable, the Simple Schedule Format class associated with the feature.

A1.2.5 Reference Column

The reference column indicates the relevant subsection or table in the *Simple Schedule Format Specification* (CCSDS 902.1-R-2) (this document).

A1.2.6 Status Column

The status column uses the following notations:

M mandatory.

O optional.

It should be noted that a parameter may be marked as M(andatory) while the class that contains it is marked O(ptional). This should be interpreted to mean that while the class is optional if it is present then the parameter must be present.

A1.2.7 Support Column Symbols

The support column is to be used by the implementer to state whether a feature is supported by entering Y, N, or N/A, indicating:

Y Yes, supported by the implementation.

N No, not supported by the implementation.

N/A Not applicable.

A1.3 INSTRUCTIONS FOR COMPLETING THE RL

An implementer shows the extent of compliance to the Recommended Standard by completing the RL; that is, the state of compliance with all mandatory requirements and the options supported are shown. The resulting completed RL is called an ICS. The implementer shall complete the RL by entering appropriate responses in the support or values supported column, using the notation described in A1.2. If a conditional requirement is inapplicable, N/A should be used. If a mandatory requirement is not satisfied, exception information must be supplied by entering a reference X_i , where i is a unique identifier, to an accompanying rationale for the noncompliance.

A2 ICS PROFORMA FOR SIMPLE SCHEDULE FORMAT

A2.1 GENERAL INFORMATION

A2.1.1 Identification of ICS

Date of Statement (DD/MM/YYYY)	
ICS serial number	
System Conformance statement cross-reference	

A2.1.2 Identification of Implementation Under Test (IUT)

Implementation name	
Implementation version	
Special Configuration	
Other Information	

A2.1.3 Identification of Supplier

Supplier	
Contact Point for Queries	
Implementation Name(s) and Versions	
Other Information necessary for full identification, e.g., names(s) and version(s) for machines and/or operating systems;	

A2.1.4 Document Version

CCSDS 902.1-R-2, Issue 2	
Have any exceptions been required? NOTE – A YES answer means that the implementation does not conform to the Recommended Standard. Non-supported mandatory capabilities are to be identified in the ICS, with an explanation of why the implementation is non-conforming	Yes _____ No _____

A2.1.5 Requirements List**A2.1.5.1 Class SimpleSchedule****A2.1.5.1.1 General**

Item	Description	Ref.	Status	Support
1.	SimpleSchedule	3.2.2	M	

A2.1.5.1.2 Class SimpleSchedule Parameters

Item	Parameter	Ref.	Status	Item Support or Values Supported
1.1.	srvMgtEntityType	Table 3-1	M	

A2.1.5.2 Class SimpleScheduleHeader**A2.1.5.2.1 General**

Item	Description	Ref.	Status	Support
2.	SimpleScheduleHeader	3.2.3	M	

A2.1.5.2.2 Class SimpleScheduleHeader Parameters

Item	Parameter	Ref.	Status	Item Support or Values Supported
2.1.	status	Table 3-2	M	
2.2.	inclusionType	Table 3-2	M	
2.3.	originatingOrganization	Table B-2	M	
2.4.	generationTime	Table B-2	M	
2.5.	version	Table B-2	M	
2.6.	startTime	Table 3-3	M	
2.7.	endTime	Table 3-3	M	
2.8.	purpose	Table B-2	O	
2.9.	description	Table B-2	O	

A2.1.5.3 Class ScheduledPackage**A2.1.5.3.1 General**

Item	Description	Ref.	Status	Support
3.	ScheduledPackage	3.2.4	O	

A2.1.5.3.2 Class ScheduledPackage Parameters

Item	Parameter	Ref.	Status	Item Support or Values Supported
3.1.	scheduledPackageID	Table 3-4	M	
3.2.	user	Table 3-4	M	
3.3.	comment	Table 3-4	O	
3.4.	originatingRequestID	Table 3-4	O	

A2.1.5.4 Class ScheduledPkgAssn**A2.1.5.4.1 General**

Item	Description	Ref.	Status	Support
4.	ScheduledPkgAssn	3.2.5	O	

A2.1.5.4.2 Class ScheduledPkgAssn Parameters

Item	Parameter	Ref.	Status	Item Support or Values Supported
4.1.	associationId	Table 3-5	M	
4.2.	associationKind	Table 3-5	M	
4.3.	comment	Table 3-5	O	
4.4.	scheduledPackageIdRef	Table 3-5	M	
4.5.	externalSchedPkgIdRef	Table 3-5	O	

A2.1.5.5 Class ServicePackageXRef

A2.1.5.5.1 General

Item	Description	Ref.	Status	Support
5.	ServicePackageXRef	3-9	O	

A2.1.5.5.2 Class ServicePackageXRef Parameters

Item	Parameter	Ref.	Status	Item Support or Values Supported
5.1.	serviceAgreementRef	Table 3-6	M	
5.2.	servicePackageRef	Table 3-6	M	

A2.1.5.6 Class ScheduledActivity

A2.1.5.6.1 General

Item	Description	Ref.	Status	Support
6.	ScheduledActivity	3.2.7	C1	

C1: If a ScheduledPackage class is contained in the Simple Schedule then there must be at least one ScheduledActivity for each ScheduledPackage.

A2.1.5.6.2 Class ScheduledActivity Parameters

Item	Parameter	Ref.	Status	Item Support or Values Supported
6.1.	scheduledActivityID	Table 3-7	M	
6.2.	activityStatus	Table 3-7	M	
6.3.	siteRef	Table 3-7	M	
6.4.	apertureRef	Table 3-7	M	
6.5.	beginningOfActivity	Table 3-7	O	
6.6.	beginningOfTrack	Table 3-7	M	
6.7.	endOfTrack	Table 3-7	M	
6.8.	endOfActivity	Table 3-7	O	
6.9.	orbitNumber	Table 3-7	O	

A2.1.5.7 Class ServiceInfo**A2.1.5.7.1 General**

Item	Class	Ref.	Status	Support
7.	ServiceInfo	3.2.8	C2	

C2: If a ScheduledActivity class is contained in the Simple Schedule then there must be at least one ServiceInfo for each ScheduledActivity.

A2.1.5.7.2 Class ServiceInfo Parameters

Item	Parameter	Ref.	Status	Item Support or Values Supported
7.1.	serviceType	Table 3-8	M	
7.2.	frequencyBand	Table 3-8	M	

A2.1.5.8 Class AbstractParameter**A2.1.5.8.1 General**

Item	Description	Ref.	Status	Support
8.	AbstractParameter	C2.2	O	

A2.1.5.8.2 Class AbstractParameter Parameters

Item	Parameter	Ref.	Status	Item Support or Values Supported
8.1.	name	Table C-1	M	

A2.1.5.9 Class BooleanParameter**A2.1.5.9.1 General**

Item	Description	Ref.	Status	Support
9.	BooleanParameter	C2.3	O	

A2.1.5.9.2 Class BooleanParameter Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
9.1.	value	Table C-2	M	

A2.1.5.10 Class UnsignedIntParameter**A2.1.5.10.1 General**

Item	Description	Ref.	Status	Support
10.	UnsignedIntParameter	C2.4	O	

A2.1.5.10.2 Class UnsignedIntParameter Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
10.1.	value	Table C-3	M	

A2.1.5.11 Class IntParameter**A2.1.5.11.1 General**

Item	Description	Ref.	Status	Support
11.	IntParameter	C2.5	O	

A2.1.5.11.2 Class IntParameter Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
11.1.	value	Table C-4	M	

A2.1.5.12 Class FloatParameter**A2.1.5.12.1 General**

Item	Description	Ref.	Status	Support
12.	FloatParameter	C2.6	O	

A2.1.5.12.2 Class FloatParameter Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
12.1.	value	Table C-5	M	

A2.1.5.13 Class DoubleParameter**A2.1.5.13.1 General**

Item	Description	Ref.	Status	Support
13.	DoubleParameter	C2.7	O	

A2.1.5.13.2 Class DoubleParameter Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
13.1.	value	Table C-6	M	

A2.1.5.14 Class StringParameter**A2.1.5.14.1 General**

Item	Description	Ref.	Status	Support
14.	StringParameter	C2.8	O	

A2.1.5.14.2 Class StringParameter Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
14.1.	value	Table C-7	M	

A2.1.5.15 Class ExtendedParameter**A2.1.5.15.1 General**

Item	Description	Ref.	Status	Support
15.	ExtendedParameter	C2.9	O	

A2.1.5.15.2 Class ExtendedParameter Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
15.1.	value	Table C-8	M	

A2.1.5.16 Class DurationParameter**A2.1.5.16.1 General**

Item	Description	Ref.	Status	Support
16.	DurationParameter	C2.10	O	

A2.1.5.16.2 Class DurationParameter Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
16.1.	value	Table C-9	M	

A2.1.5.17 Class TimeParameterA**A2.1.5.17.1 General**

Item	Description	Ref.	Status	Support
17.	TimeParameterA	C2.11	O	

A2.1.5.17.2 Class TimeParameterA Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
17.1.	value	Table C-10	M	

A2.1.5.18 Class TimeParameterB**A2.1.5.18.1 General**

Item	Description	Ref.	Status	Support
18.	TimeParameterB	C2.12	O	

A2.1.5.18.2 Class TimeParameterB Parameters

As per A2.1.5.8.1 with the addition of the following:

Item	Parameter	Ref.	Status	Item Support or Values Supported
18.1.	value	Table C-11	M	

A2.1.5.19 Additional Parameters defined by using Classes instantiated from Class AbstractParameter

If additional parameters for classes SimpleScheduleHeader, ScheduledPackage, ScheduledPkgAssn, ScheduledActivity or ServiceInfo are specified by using one of the classes instantiated from class AbstractParameter (i.e., classes BooleanParameter, UnsignedIntParameter, IntParameter, FloatParameter, DoubleParameter, StringParameter, ExtendedParameter, DurationParameter, TimeParameterA or TimeParameterB) the additional parameters and permitted values for these should be specified in tables A-1–A-5.

In the following tables the columns have the following use:

- Name This column is used to specify the names of the additional parameter(s).
- Description This column is used to enter the description of what the additional parameter(s) is/are.
- Type This column is used to specify what type the parameter is, e.g., String, Integer, Unsigned Integer, Float, CCSDS ASCII Time Code B, etc.
- Permitted Values This column is used to specify the values that are permitted for the additional parameter(s).
- Data units This column is used to specify the Data Unit of the additional parameter(s), e.g., Seconds, Hertz, Volts, UTC, etc.

Table A-1: Class AbstractParameter Instantiated for Class SimpleScheduleHeader

Name	Description	Type	Permitted Values	Data units

Table A-2: Class AbstractParameter Instantiated for Class ScheduledPackage

Name	Description	Type	Permitted Values	Data units

Table A-3: Class AbstractParameter Instantiated for Class ScheduledPkgAssn

Name	Description	Type	Permitted Values	Data units

Table A-4: Class AbstractParameter Instantiated for Class ScheduledActivity

Name	Description	Type	Permitted Values	Data units

Table A-5: Class AbstractParameter Instantiated for Class ServiceInfo

Name	Description	Type	Permitted Values	Data units

ANNEX B

SERVICE MANAGEMENT INFORMATION ENTITY

(NORMATIVE)

B1 GENERAL

The Service Management Information Entity forms the basis for defining information entities required by service management.

B2 SERVICE MANAGEMENT INFORMATION ENTITY
CONTENT/STRUCTURE

B2.1 OVERVIEW

Figure B-1 shows the UML Class diagram for the Service Management Information Entity. For clarity, abstract classes are highlighted in green.

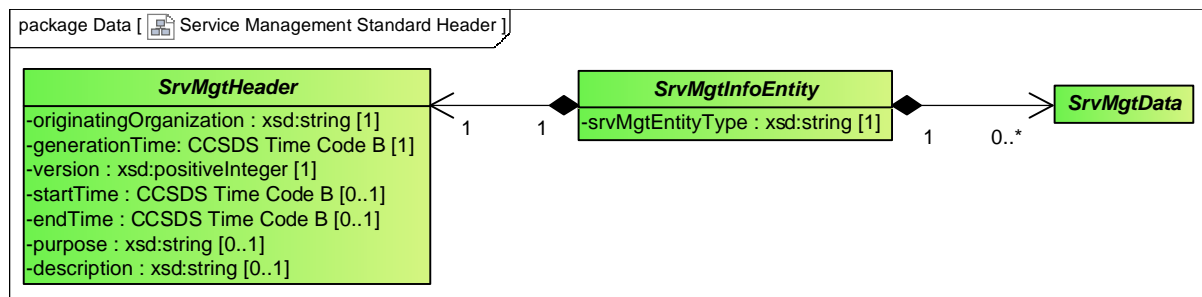


Figure B-1: Service Management Information Entity Class Diagram

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

B2.2 CLASS SrvMgtInfoEntity (ABSTRACT)

B2.2.1 The SrvMgtInfoEntity is an abstract class which is used to instantiate the various classes of information entity required by service management.

B2.2.2 The SrvMgtInfoEntity class parameters specified in table B-1 shall be used to identify the information entity type.

Table B-1: Class SrvMgtInfoEntity Parameters

Parameter	Description	Data Type	Data Units
srvMgtEntityType	The type of service management information entity contained.	xsd:string Permitted values are registered in SANA (see annex C). The following list identifies the currently foreseen values; the list may be extended in the future: <ul style="list-style-type: none"> – ACCOUNTING_REPORT – COMMUNICATIONS_GEOMETRY – CONFIGURATION_PROFILE – COST_ESTIMATE – INFORMATION_REQUEST – PLANNING_INFORMATION_REQUEST – PLANNING_INFORMATION – RADIO_FREQUENCY_INTERFERENCE – REPORT_REQUEST – RESOURCE_AVAILABILITY_INFORMATION – RESOURCE_CONFLICTS – SCHEDULE_OF_SERVICES – SERVICE_AGREEMENT – SERVICE_PACKAGE – SERVICE_PACKAGE – EXECUTION_EVENT_NOTIFICATION – SERVICE_PACKAGE_REQUEST – SPACELINK_EVENT_SEQUENCE – SUBMISSION_REQUEST – SUSTAINABLE_DATA_RATE_VOLUME_ESTIMATE 	n/a

B2.3 CLASS SrvMgtHeader (ABSTRACT)

B2.3.1 SrvMgtHeader is an abstract class which is used to instantiate the header required by service management.

B2.3.2 The SrvMgtHeader class parameters specified in table B-2 shall be used to identify the originating organization, status of the schedule, status, version, the time at which the information entity was generated, the start time to which the information entity applies, and the end time to which it applies. Optional parameters allow for further definition of the purpose of the message and a description of its contents.

Table B-2: Class SrvMgtHeader Parameters

Parameter	Description	Data Type	Data Units
originatingOrganization	The organization that generated the information entity.	xsd:string Permitted values are registered in SANA (see D2.2 for further information).	n/a
generationTime	Time at which the information entity was generated.	CCSDS ASCII Time Code B (reference [3])	UTC
version	The version of the information entity. This increments every time an information entity of the same <code>srvMgtEntityType</code> , status, and time range is generated (i.e., has the same <code>startTime</code> and <code>endTime</code>). NOTE – The version may increment by 1 every time but is not constrained to do so. The only constraint is that each version number is greater than the previous.	xsd:positiveInteger	n/a
startTime	Optional parameter: The start time to which the information entity applies. NOTE – For certain Service Management Information Entity types use of this parameter is mandatory.	CCSDS ASCII Time Code B (reference [3])	UTC
endTime	Optional parameter: The end time to which the information entity applies. NOTE – For certain Service Management Information Entity types use of this parameter is mandatory.	CCSDS ASCII Time Code B (reference [3])	UTC
purpose	Optional parameter: May be used to further specify the purpose of the information entity in the event that the status specified is ‘OTHER’.	xsd:string Exact use of this may be specified in an ICD between relevant parties.	n/a
description	Optional parameter: May be used to describe the contents of the information entity, e.g., any filter criteria used in the generation of the information entity, etc.	xsd:string Exact use of this may be specified in an ICD between relevant parties.	n/a

B2.4 CLASS SrvMgtData (ABSTRACT)

B2.4.1 SrvMgtData is an abstract class which is used to instantiate the various classes of data required by service management.

B2.4.2 The SrvMgtData class contains no parameters.

B3 XML SCHEMAS FOR THE SERVICE MANAGEMENT INFORMATION ENTITY

Each Service Management Information Entity shall conform to the XML schemas specified in the files ‘SchemaCsmSmInfoEntityHeader-v1_0_0.xsd’ and ‘SchemaCsmCcsdsTimecodes-v1_0_0.xsd’. The normative source of these files shall be the SANA SCCS-SM Information Entity XML Schemas registry:

http://sanaregistry.org/r/service_management_xml_schemas/

NOTE – Annex E provides an informative description of the Service Management Information Entity XML schemas.

ANNEX C

SERVICE MANAGEMENT ABSTRACT PARAMETER CLASS

(NORMATIVE)

C1 OVERVIEW

The Abstract parameter class forms the basis for defining parameters required by service management.

C2 ABSTRACT PARAMETER CONTENT/STRUCTURE

C2.1 OVERVIEW

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

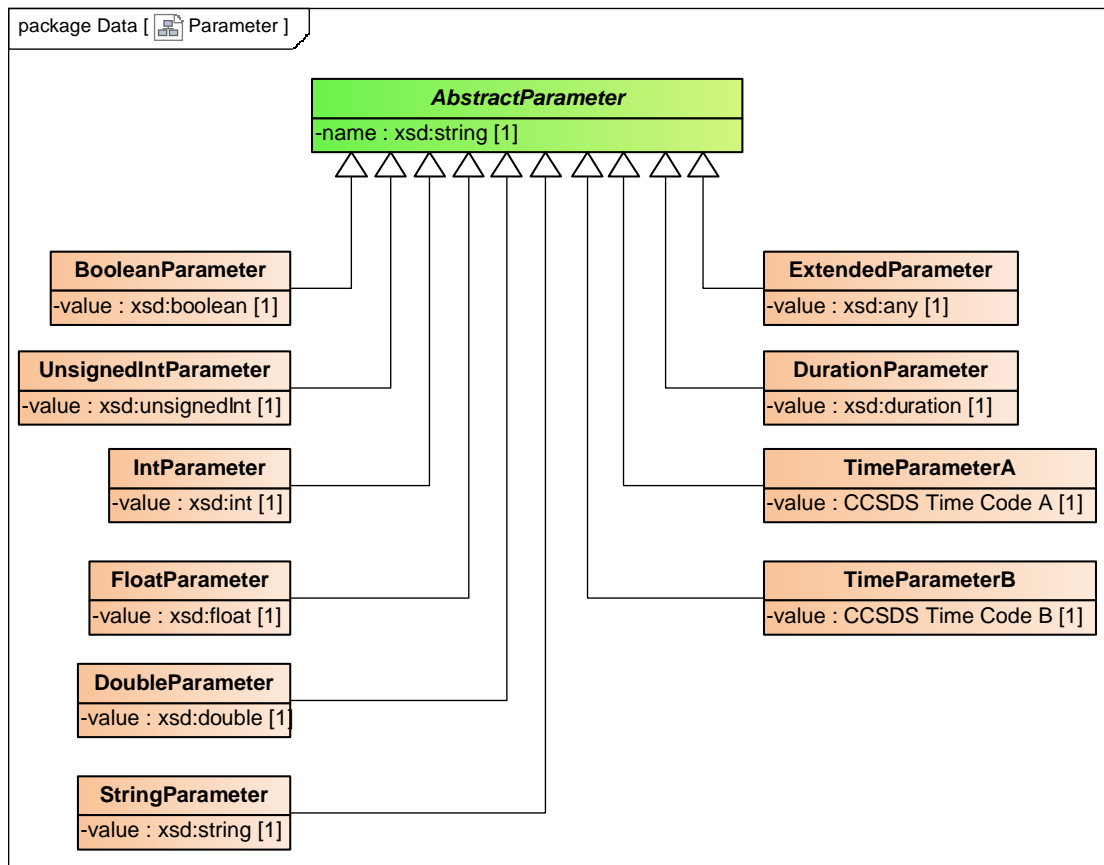


Figure C-1: Abstract Parameter Class Diagram

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

C2.2 CLASS AbstractParameter (ABSTRACT)

C2.2.1 The AbstractParameter is an abstract class which can be used to instantiate various types of parameters.

C2.2.2 The AbstractParameter class parameters specified in table C-1 shall be used to identify the name of the parameter.

Table C-1: Class AbstractParameter Parameters

Parameter	Description	Data Type	Data Units
name	The name of the parameter.	String	n/a

C2.3 CLASS BooleanParameter

C2.3.1 The BooleanParameter is a class which can be used to instantiate a parameter with type xsd:boolean.

NOTE – The BooleanParameter class is a specialization of class AbstractParameter described in C2.2.

C2.3.2 The BooleanParameter class parameters specified in table C-2 shall be used to specify the value of the parameter.

NOTE – The parameters for the BooleanParameter class are as per table C-1 with the addition of table C-2.

Table C-2: Class BooleanParameter Parameters

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

C2.4 CLASS UnsignedIntParameter

C2.4.1 The UnsignedIntParameter is a class which can be used to instantiate a parameter with type xsd:unsigned integer.

NOTE – The UnsignedIntParameter class is a specialization of class AbstractParameter described in C2.2.

C2.4.2 The UnsignedIntParameter class parameters specified in table C-3 shall be used to specify the value of the parameter.

NOTE – The parameters for the UnsignedIntParameter class are as per table C-1 with the addition of table C-3.

Table C-3: Class UnsignedIntParameter Parameters

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

C2.5 CLASS IntParameter

C2.5.1 The IntParameter is a class which can be used to instantiate a parameter with type xsd:int.

NOTE – The IntParameter class is a specialization of class AbstractParameter described in C2.2.

C2.5.2 The IntParameter class parameters specified in table C-4 shall be used to specify the value of the parameter.

NOTE – The parameters for the IntParameter class are as per table C-1 with the addition of table C-4.

Table C-4: Class IntParameter Parameters

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

C2.6 CLASS FloatParameter

C2.6.1 The FloatParameter is a class which can be used to instantiate a parameter with type xsd:float.

NOTE – The FloatParameter class is a specialization of class AbstractParameter described in C2.2.

C2.6.2 The FloatParameter class parameters specified in table C-5 shall be used to specify the value of the parameter.

NOTE – The parameters for the FloatParameter class are as per table C-1 with the addition of table C-5.

Table C-5: 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

C2.7 CLASS DoubleParameter

C2.7.1 The DoubleParameter is a class which can be used to instantiate a parameter with type xsd:double.

NOTE – The DoubleParameter class is a specialization of class AbstractParameter described in C2.2.

C2.7.2 The DoubleParameter class parameters specified in table C-6 shall be used to specify the value of the parameter.

NOTE – The parameters for the DoubleParameter class are as per table C-1 with the addition of table C-6.

Table C-6: 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

C2.8 CLASS StringParameter

C2.8.1 The StringParameter is a class which can be used to instantiate a parameter with type `xsd:string`.

NOTE – The StringParameter class is a specialization of class AbstractParameter described in C2.2.

C2.8.2 The StringParameter class parameters specified in table C-7 shall be used to specify the value of the parameter.

NOTE – The parameters for the StringParameter class are as per table C-1 with the addition of table C-7.

Table C-7: Class StringParameter Parameters

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

C2.9 CLASS ExtendedParameter

C2.9.1 The ExtendedParameter is a class which 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 C2.2.

C2.9.2 Use of 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 D.

C2.9.3 The ExtendedParameter class parameters specified in table C-8 shall be used to specify the value of the parameter.

NOTE – The parameters for the ExtendedParameter class are as per table C-1 with the addition of table C-8.

Table C-8: Class ExtendedParameter Parameters

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

C2.10 CLASS DurationParameter

C2.10.1 The DurationParameter is a class which can be used to instantiate a parameter with type xsd:duration.

NOTE – The DurationParameter class is a specialization of class AbstractParameter described in C2.2.

C2.10.2 The DurationParameter class parameters specified in table C-9 shall be used to specify the value of the parameter.

NOTE – The parameters for the DurationParameter class are as per table C-1 with the addition of table C-9.

Table C-9: Class DurationParameter Parameters

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

C2.11 CLASS TimeParameterA

C2.11.1 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 C2.2.

C2.11.2 The TimeParameterA class parameters specified in table C-10 shall be used to specify the value of the parameter.

NOTE – The parameters for the TimeParameterA class are as per table C-1 with the addition of table C-10.

Table C-10: Class TimeParameterA Parameters

Parameter	Description	Data Type	Data Units
value	The value of the parameter.	Time in CCSDS Time Code A (reference [3])	UTC

C2.12 CLASS TimeParameterB

C2.12.1 The TimeParameterB is a class which 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 C2.2.

C2.12.2 The TimeParameterB class parameters specified in table C-11 shall be used to specify the value of the parameter.

NOTE – The parameters for the TimeParameterB class are as per table C-1 with the addition of table C-11.

Table C-11: Class TimeParameterB Parameters

Parameter	Description	Data Type	Data Units
Value	The value of the parameter.	Time in CCSDS Time Code B (reference [3])	UTC

C3 XML SCHEMAS FOR THE SERVICE MANAGEMENT INFORMATION ENTITY

Each Service Management Information Entity shall conform to the XML schemas specified in the files ‘SchemaCsmCommonTypes-v1_0_0.xsd’ and ‘SchemaCsmCcsdsTimecodes-v1_0_0.xsd’. The normative source of these files shall be the SANA SCCS-SM Information Entity XML Schemas registry:

http://sanaregistry.org/r/service_management_xml_schemas/

NOTE – Annex G provides an informative description of the Simple Schedule XML schemas.

ANNEX D

SECURITY, SANA, AND PATENT CONSIDERATIONS

(INFORMATIVE)

D1 SECURITY CONSIDERATIONS

D1.1 OVERVIEW

This annex presents the results of an analysis of security considerations applied to the technologies specified in this Recommended Standard.

D1.2 CONSEQUENCES OF NOT APPLYING SECURITY TO THE TECHNOLOGY

The consequences of not applying security to the systems and networks on which this Recommended Standard is implemented could include potential loss, corruption, and theft of data. Since it is possible to utilize these messages in preparing and disseminating schedules relating to the availability of communications and tracking resources for spacecraft, the consequences of not applying security to the systems and networks on which this Recommended Standard is implemented could include compromise or loss of the mission if malicious tampering of a particularly severe nature occurs.

D1.3 POTENTIAL THREATS AND ATTACK SCENARIOS

Potential threats or attack scenarios include, but are not limited to, (a) unauthorized access to the programs/processes that generate and interpret the messages, and (b) unauthorized access to the messages during transmission between exchange partners. Protection from unauthorized access during transmission is especially important if the mission utilizes open ground networks such as the Internet to provide ground station connectivity for the exchange of data formatted in compliance with this Recommended Standard. It is strongly recommended that potential threats or attack scenarios applicable to the systems and networks on which this Recommended Standard is implemented be addressed by the management of those systems and networks and the utilization of adequate authentication, suitable protocols, and secured interfaces for the exchange of this information.

D1.4 SECURITY CONCERNS RELATED TO THIS RECOMMENDED STANDARD

D1.4.1 Data Privacy

Privacy of data formatted in compliance with the specifications of this Recommended Standard should be assured by the systems and networks on which this Recommended Standard is implemented.

D1.4.2 Data Integrity

Integrity of data formatted in compliance with the specifications of this Recommended Standard should be assured by the systems and networks on which this Recommended Standard is implemented.

D1.4.3 Authentication of Communicating Entities

Authentication of communicating entities involved in the transport of data which complies with the specifications of this Recommended Standard should be provided by the systems and networks on which this Recommended Standard is implemented.

D1.4.4 Data Transfer between Communicating Entities

The transfer of data formatted in compliance with this Recommended Standard between communicating entities should be accomplished via secure mechanisms approved by the Information Technology Security functionaries of exchange participants.

D1.4.5 Control of Access to Resources

Control of access to resources should be managed by the systems upon which provider formatting and recipient processing are performed.

D1.4.6 Auditing of Resource Usage

Auditing of resource usage should be handled by the management of systems and networks on which this Recommended Standard is implemented.

D1.5 UNAUTHORIZED ACCESS

Unauthorized access to the programs/processes that generate and interpret the messages should be prohibited in order to minimize potential threats and attack scenarios.

D1.6 DATA SECURITY IMPLEMENTATION SPECIFICS

Specific information-security interoperability provisions that apply between agencies and other independent users involved in an exchange of data formatted in compliance with this Recommended Standard should be specified in an ICD.

D2 SANA CONSIDERATIONS

D2.1 GENERAL

The recommendations of this document rely on the SANA registries described below. New assignments in these registries, in conformance with the policies identified, will be available at the SANA registry Web site: <http://sanaregistry.org>. Therefore the reader shall look at the SANA Web site for all the assignments contained in these registries.

D2.2 REGISTRY FOR `originatingOrganization`

The values for `originatingOrganization` (see table B-2) are those listed in the ‘Name’ field of the CCSDS Organizations registry. The CCSDS Organization’s registry is located at

<http://sanaregistry.org/r/organizations/>

The CCSDS Organizations registry also list roles that organizations fulfill. Organizations that publish schedules have the ‘SchedulePublisher’ role assigned (this may be addition to other roles). The policy that governs updates to the Organization and Service Provider Types registries is that defined in subsection 3.3.2.3 of reference [5].

D2.3 REGISTRY FOR `user`

The values for the ‘`user`’ parameter (see table 3-4) are those listed in the ‘Spacecraft Abbreviation’ field of the CCSDS Spacecraft Identifiers registry. The Spacecraft Identifiers registry is located at

<http://sanaregistry.org/r/spacecraftid/>

The policy that governs updates to this registry is that defined in subsection 3.3.2.1 of reference [5].

In addition to the values for the ‘`user`’ parameter contained in the CCSDS Spacecraft Identifiers registry, the following values need to be recognized by implementations (see table 3-4):

<code>user</code>	Description
UNALLOCATED	Indicates that the time is unallocated.
PROVIDER-CSSS	Indicates that the time is allocated for the Provider CSSS.

D2.4 REGISTRY FOR `siteRef` AND `apertureRef`

The values for the ‘`siteRef`’ parameter (see table 3-7) of this Recommended Standard are those of the Site Name Abbreviation field of the Service Sites and Apertures registry. The Service Sites and Apertures registry is located at²

http://sanaregistry.org/r/service_sites_apertures/

Each Site Name record contains at least one Aperture Name field. It may contain as many fields as needed to list as many apertures found at one site as needed. The Aperture Name and Aperture Name Abbreviations are unique with respect to any other Aperture Name or Aperture Name Abbreviations for the particular Site Name.

The values for the `apertureRef` parameter (see table 3-7) of this Recommended Standard are those of the Aperture Name Abbreviation field.

It should be noted that this Recommended Standard specifies strings of 16 characters or less for both the `siteRef` and `apertureRef` parameters.

The policy that governs updates to this registry is that defined in subsection 3.3.2.4 of reference [5].

D2.5 REGISTRY FOR `srvMgtEntityType`

The `srvMgtEntityType` (`SrvMgtInfoEntity` class—see table B-1) values are registered and maintained in SANA; the registry is located at

http://sanaregistry.org/r/service_management_entity_types/

Per reference [5], updates of this type of registry is performed by CCSDS.

D2.6 REGISTRY FOR `associationKinds`

The `associationKinds` (`ScheduledPackageAssn` class—see table 3-5) values are registered and maintained in SANA; the registry is located at

http://sanaregistry.org/r/association_kinds/

Per reference [5], updates of this type of registry is performed by CCSDS.

² The URL for this Registry may not be finalized prior to final publication of this document; at time of release of this current draft it can be found at http://beta.sanaregistry.org/r/service_sites_apertures/.

D2.7 REGISTRY FOR XML SCHEMA

The XML Schema for this recommended standards is registered with SANA and is located at

http://sanaregistry.org/r/service_management_xml_schemas/

Per reference [5], updates of this type of registry are performed by CCSDS.

D2.8 REGISTRY FOR EXTENDED PARAMETER

The extended parameter requires an additional schema or schema files for validation (see C2.9). The content of such schema are agency specific and are outside the scope of this Recommended Standard. However, to assist with inter-agency inter-operations, the agency specific schema are registered.

The information registered is:

- a) Agency – the agency that is registering the extended parameter;

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

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

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

- c) Extends – the name and version of the CCSDS CSSM Schema that the agency-specific schema is extending:
 - for extensions relative to the schema defined by this Recommended Standard, the CSSM Schema name is ‘SimpleSchedule’ and the version number is ‘1.0.0’;
 - multiple version numbers are supplied if the extension is compatible with and applies to multiple versions of the SimpleSchedule schema;
- d) Description – a brief description of the type of extension being afforded by the agency specific schema;
- e) Applicable Start Date – the date when an agency recognizes and processes its particular extended parameter conforming to the registered schema;
- f) 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 implement of the agency specific extended parameter.

- a) Schema file or files – the file(s) that contain the XML schema for the agency specific extended parameter.

Updates to this registry are performed by making a request of the SANA Operator.

The extended parameter registry is located at

http://sanaregistry.org/r/agency_specific_schemas/

D2.9 USE OF UNREGISTERED VALUES

Only values that have been registered should be used for the `originatingOrganization`, `user` and `apertureRef` parameters. Unregistered values for the `originatingOrganization`, `user`, and `apertureRef` 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.

D3 PATENT CONSIDERATIONS

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

ANNEX E

XML SCHEMA FOR THE SERVICE MANAGEMENT

INFORMATION ENTITY HEADER

(INFORMATIVE)

E1 PURPOSE

This annex provides an informative listing of the normative XML schema that realizes the classes that comprise the abstract Service Management Information Entity as defined in annex B.

E2 SCHEMA ORGANIZATION AND PACKAGING

The normative Service Management Information Entity Header schema types and global elements are contained in the file ‘SchemaCssmSmInfoEntityHeader-v1_0_0.xsd’.

The SM Information Entity Header types and global elements are registered in the ‘urn:ccsds:schema:csm:1.0.0’ name space.

The Simple Schedule schema includes the following schemas;

SchemaCssmCcsdsTimecodes-v1_0_0.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 SchemaCssmSmInfoEntityHeader-v1_0_0.xsd and SchemaCssmCcsdsTimecodes-v1_0_0.xsd files is the SANA SCCS-SM Information Entity XML Schemas registry:

http://sanaregistry.org/r/service_management_xml_schemas/

E3 SCHEMA LISTING

The following is a listing of the XML schema for the Service Management Information Entity.

```
<?xml version="1.0" encoding="UTF-8"?>
<xsd:schema xmlns="urn:ccsds:schema:cissm:1.0.0"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="urn:ccsds:schema:cissm:1.0.0" elementFormDefault="qualified"
attributeFormDefault="unqualified" version="1.0.0">
  <xsd:include schemaLocation="SchemaCissmCcsdsTimecodes-v1_0_0.xsd"/>
  <!--
*****
***** -->
  <!--** Service Management Information Entity Header Types **-->
  <xsd:complexType name="SrvMgtInfoEntityType">
    <xsd:sequence>
      <xsd:element ref="abstractSrvMgtEntityType">
        <xsd:annotation>
          <xsd:documentation>substitution group head
element</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element ref="srvMgtHeader">
        <xsd:annotation>
          <xsd:documentation>substitution group head
element</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
      <xsd:element ref="srvMgtData" minOccurs="0" maxOccurs="unbounded">
        <xsd:annotation>
          <xsd:documentation>substitution group head
element</xsd:documentation>
        </xsd:annotation>
      </xsd:element>
    </xsd:sequence>
  </xsd:complexType>
  <xsd:element name="srvMgtHeader" type="SrvMgtHeaderType" abstract="true"/>
  <xsd:complexType name="SrvMgtHeaderType">
    <xsd:attribute name="originatingOrganization" type="xsd:string"
use="required"/>
    <xsd:attribute name="generationTime" type="CcsdsAsciiTimeCodeBType"
use="required"/>
    <xsd:attribute name="version" type="xsd:positiveInteger" use="required"/>
    <xsd:attribute name="startTime" type="CcsdsAsciiTimeCodeBType"/>
    <xsd:attribute name="endTime" type="CcsdsAsciiTimeCodeBType"/>
    <xsd:attribute name="purpose" type="xsd:string"/>
    <xsd:attribute name="description" type="xsd:string"/>
  </xsd:complexType>
  <xsd:element name="srvMgtData" type="SrvMgtDataType" abstract="true"/>
  <xsd:complexType name="SrvMgtDataType" abstract="true"/>
  <xsd:simpleType name="SrvMgtEntityType">
    <xsd:restriction base="xsd:string"/>
  </xsd:simpleType>
  <xsd:element name="abstractSrvMgtEntityType" type="SrvMgtEntityType"
abstract="true">
    <xsd:annotation>
      <xsd:documentation>The srvMgtEntityType element is abstract. It must be
substituted in any concrete, Information Entity-specific derivation of the
```

SrvMgtInfoEntityType. That concrete substitution element must have its value fixed to the single string value specified for that Information Entity type in the SANA registry.</xsd:documentation>
</xsd:annotation>
</xsd:element>
</xsd:schema>

ANNEX F

XML SCHEMA FOR THE SERVICE MANAGEMENT

ABSTRACT PARAMETER CLASS

(INFORMATIVE)

F1 PURPOSE

This annex provides an informative listing of the normative XML schema that realizes the classes that comprise the Service Management Abstract Parameter as defined in annex C.

F2 SCHEMA ORGANIZATION AND PACKAGING

The normative Service Management Abstract Parameter schema types and global elements are contained in the file 'SchemaCsmCommonTypes-v1_0_0.xsd'.

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

The Service Management Abstract Parameter schema the following schemas;

SchemaCsmCcsdsTimecodes-v1_0_0.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 SchemaCsmCommonTypes-v1_0_0.xsd and SchemaCsmCcsdsTimecodes-v1_0_0.xsd files is the SANA SCCS-SM Information Entity XML Schemas registry:

http://sanaregistry.org/r/service_management_xml_schemas/

F3 SCHEMA LISTING

The following is a listing of the XML schema for the Service Management Abstract Parameter class.

```

<xsd:schema xmlns="urn:ccsds:schema:cssm:1.0.0"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="urn:ccsds:schema:cssm:1.0.0" elementFormDefault="qualified"
attributeFormDefault="unqualified" version="1.0.0">

  <!--
  ****
  **** -->
  <xsd:include schemaLocation="SchemaCcssmCcsdsTimecodes-v1_0_0.xsd" />
  <!-- Create an element and type for the abstract parameter -->
  <xsd:element name="AbstractParameter" type="AbstractParameterType"
abstract="true" />
  <xsd:complexType name="AbstractParameterType">
    <xsd:attribute name="name" type="xsd:string" />
  </xsd:complexType>
  <xsd:element name="booleanParameter" type="BooleanParameterType"
substitutionGroup="AbstractParameter" />
  <xsd:complexType name="BooleanParameterType">
    <xsd:complexContent>
      <xsd:extension base="AbstractParameterType">
        <xsd:attribute name="value" type="xsd:boolean" />
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="unsignedIntParameter" type="UnsignedIntParameterType"
substitutionGroup="AbstractParameter" />
  <xsd:complexType name="UnsignedIntParameterType">
    <xsd:complexContent>
      <xsd:extension base="AbstractParameterType">
        <xsd:attribute name="value" type="xsd:unsignedInt" />
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="intParameter" type="IntParameterType"
substitutionGroup="AbstractParameter" />
  <xsd:complexType name="IntParameterType">
    <xsd:complexContent>
      <xsd:extension base="AbstractParameterType">
        <xsd:attribute name="value" type="xsd:int" />
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="floatParameter" type="FloatParameterType"
substitutionGroup="AbstractParameter" />
  <xsd:complexType name="FloatParameterType">
    <xsd:complexContent>
      <xsd:extension base="AbstractParameterType">
        <xsd:attribute name="value" type="xsd:float" />
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="doubleParameter" type="DoubleParameterType"
substitutionGroup="AbstractParameter" />

```

```

<xsd:complexType name="DoubleParameterType">
  <xsd:complexContent>
    <xsd:extension base="AbstractParameterType">
      <xsd:attribute name="value" type="xsd:double"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="stringParameter" type="StringParameterType"
substitutionGroup="AbstractParameter"/>
<xsd:complexType name="StringParameterType">
  <xsd:complexContent>
    <xsd:extension base="AbstractParameterType">
      <xsd:attribute name="value" type="xsd:string"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="extendedParameter" type="ExtendedParameterType"
substitutionGroup="AbstractParameter"/>
<xsd:complexType name="ExtendedParameterType">
  <xsd:complexContent>
    <xsd:extension base="AbstractParameterType">
      <xsd:sequence>
        <xsd:any minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="durationParameter" type="DurationParameterType"
substitutionGroup="AbstractParameter"/>
<xsd:complexType name="DurationParameterType">
  <xsd:complexContent>
    <xsd:extension base="AbstractParameterType">
      <xsd:attribute name="value" type="xsd:duration"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="timeParameterA" type="TimeParameterAType"
substitutionGroup="AbstractParameter"/>
<xsd:complexType name="TimeParameterAType">
  <xsd:complexContent>
    <xsd:extension base="AbstractParameterType">
      <xsd:attribute name="value" type="CcsdsAsciiTimeCodeAType"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:element name="timeParameterB" type="TimeParameterBType"
substitutionGroup="AbstractParameter"/>
<xsd:complexType name="TimeParameterBType">
  <xsd:complexContent>
    <xsd:extension base="AbstractParameterType">
      <xsd:attribute name="value" type="CcsdsAsciiTimeCodeBType"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
</xsd:schema>

```

ANNEX G

XML SCHEMA FOR THE SIMPLE SCHEDULE FORMAT

(INFORMATIVE)

G1 PURPOSE

This annex provides an informative listing of the normative XML schema that realizes the classes that comprise the Simple Schedule Format as defined in section 3.

G2 SCHEMA ORGANIZATION AND PACKAGING

The XML schemas for the Simple Schedule are contained in the file 'SchemaCsmSimpleSchedule-v1_0_0.xsd'. The root element of the SchemaCsmSimpleSchedule-v1_0_0 schemas is the `simpleSchedule` element.

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

The Simple Schedule schema includes the following schemas:

- a) SchemaCsmSmInfoEntityHeader-v1_0_0.xsd. Types and global elements in this schema are registered in the 'urn:ccsds:schema:csm:1.0.0' name space.
- b) SchemaCsmCommonTypes-v1_0_0.xsd. Types and global elements in this schema are registered in the 'urn:ccsds:schema:csm:1.0.0' name space.
- c) SchemaCsmCcsdsTimecodes-v1_0_0.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 SchemaCsmSimpleSchedule-v1_0_0.xsd, SchemaCsmSmInfoEntityHeader-v1_0_0.xsd, SchemaCsmCommonTypes-v1_0_0.xsd and SchemaCsmCcsdsTimecodes-v1_0_0.xsd files is the SANA SCCS-SM Information Entity XML Schemas registry:

http://sanaregistry.org/r/service_management_xml_schemas/

G3 SCHEMA

The following is a listing of the XML schema for the Simple Schedule format.

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<xsd:schema xmlns="urn:ccsds:schema:cssm:1.0.0"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"
targetNamespace="urn:ccsds:schema:cssm:1.0.0" elementFormDefault="qualified"
attributeFormDefault="unqualified" version="1.0.0">
  <xsd:include schemaLocation="SchemaCcssmCcsdsTimecodes-v1_0_0.xsd"/>
  <xsd:include schemaLocation="SchemaCcssmSmInfoEntityHeader-v1_0_0.xsd"/>
  <xsd:include schemaLocation="SchemaCcssmCommonTypes-v1_0_0.xsd"/>
  <!--
*****
***** -->
  <!--** Simple Schedule types and elements ** -->
  <xsd:element name="simpleSchedule" type="SimpleScheduleType"/>
  <xsd:complexType name="SimpleScheduleType">
    <xsd:complexContent>
      <xsd:extension base="SrvMgtInfoEntityType"/>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="srvMgtEntityType"
substitutionGroup="abstractSrvMgtEntityType" fixed="SIMPLE_SCHEDULE">
    <xsd:simpleType>
      <xsd:restriction base="SrvMgtEntityType"/>
    </xsd:simpleType>
  </xsd:element>
  <xsd:simpleType name="Status">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="TEST"/>
      <xsd:enumeration value="PROVISIONAL"/>
      <xsd:enumeration value="OPERATIONAL"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:simpleType name="InclusionType">
    <xsd:restriction base="xsd:string">
      <xsd:enumeration value="OVERLAP_INCLUSION"/>
      <xsd:enumeration value="START_INCLUSION"/>
    </xsd:restriction>
  </xsd:simpleType>
  <xsd:element name="simpleScheduleHeader" type="SimpleScheduleHeaderType"
substitutionGroup="srvMgtHeader"/>
  <xsd:complexType name="SimpleScheduleHeaderType">
    <xsd:complexContent>
      <xsd:extension base="SrvMgtHeaderType">
        <xsd:sequence>
          <!-- allow a list of parameters derived from AbstractParameter -->
          <xsd:element ref="AbstractParameter" minOccurs="0"
maxOccurs="unbounded"/>
        </xsd:sequence>
        <xsd:attribute name="status" type="Status" use="required"/>
        <xsd:attribute name="inclusionType" type="InclusionType"
use="required"/>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>
  <xsd:element name="scheduledPackage" type="ScheduledPackageType"
substitutionGroup="srvMgtData"/>

```

```

<xsd:complexType name="ScheduledPackageType">
  <xsd:complexContent>
    <xsd:extension base="SrvMgtDataType">
      <xsd:sequence>
        <xsd:element name="servicePackageXRef"
type="ServicePackageXRefType" minOccurs="0"/>
        <!-- allow a list of parameters derived from AbstractParameter -->
        <xsd:element ref="AbstractParameter" minOccurs="0"
maxOccurs="unbounded"/>
        <xsd:element name="scheduledActivity" type="ScheduledActivityType"
maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="scheduledPackageId" type="xsd:ID"
use="required"/>
      <xsd:attribute name="user" type="xsd:string" use="required"/>
      <xsd:attribute name="originatingRequestId" type="xsd:string"/>
      <xsd:attribute name="comment" type="xsd:string"/>
    </xsd:extension>
  </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="ScheduledActivityType">
  <xsd:sequence>
    <xsd:element name="serviceInfo" type="ServiceInfoType"
maxOccurs="unbounded"/>
    <!-- allow a list of parameters derived from AbstractParameter -->
    <xsd:element ref="AbstractParameter" minOccurs="0"
maxOccurs="unbounded"/>
  </xsd:sequence>
  <xsd:attribute name="scheduledActivityId" type="xsd:string"
use="required"/>
  <xsd:attribute name="activityStatus" use="required">
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="COMMITTED"/>
        <xsd:enumeration value="TENTATIVE"/>
        <xsd:enumeration value="AVAILABLE"/>
        <xsd:enumeration value="UNAVAILABLE"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:attribute>
  <xsd:attribute name="siteRef" type="xsd:string" use="required">
    <xsd:annotation>
      <xsd:documentation>SANA-registered value</xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="apertureRef" type="xsd:string" use="required">
    <xsd:annotation>
      <xsd:documentation>SANA-registered value</xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
  <xsd:attribute name="beginningOfActivity" type="CcsdsAsciiTimeCodeBType"/>
  <xsd:attribute name="beginningOfTrack" type="CcsdsAsciiTimeCodeBType"
use="required"/>
  <xsd:attribute name="endOfTrack" type="CcsdsAsciiTimeCodeBType"
use="required"/>
  <xsd:attribute name="endOfActivity" type="CcsdsAsciiTimeCodeBType"/>
  <xsd:attribute name="orbitNumber" type="xsd:nonNegativeInteger"/>
</xsd:complexType>
<xsd:complexType name="ServicePackageXRefType">

```

```

    <xsd:attribute name="serviceAgreementRef" type="xsd:string"
use="required"/>
    <xsd:attribute name="servicePackageRef" type="xsd:string" use="required"/>
</xsd:complexType>
<xsd:element name="scheduledPkgAssn" substitutionGroup="srvMgtData">
    <xsd:complexType>
        <xsd:complexContent>
            <xsd:extension base="ScheduledPkgAssnType"/>
        </xsd:complexContent>
    </xsd:complexType>
</xsd:element>
<xsd:complexType name="ScheduledPkgAssnType">
    <xsd:complexContent>
        <xsd:extension base="SrvMgtDataType">
            <xsd:sequence>
                <xsd:element name="scheduledPackageIdRef"
type="ScheduledPackageIdRefType" minOccurs="1" maxOccurs="unbounded"/>
                <xsd:element name="externalSchedPkgIdRef"
type="ExternalSchedPkgIdRefType" minOccurs="0" maxOccurs="unbounded"/>
                <!-- allow a list of parameters derived from AbstractParameter -->
                <xsd:element ref="AbstractParameter" minOccurs="0"
maxOccurs="unbounded"/>
            </xsd:sequence>
            <xsd:attribute name="associationId" type="xsd:string"
use="required"/>
            <xsd:attribute name="associationKind" type="xsd:string"
use="required">
                <xsd:annotation>
                    <xsd:documentation>SANA-registered value</xsd:documentation>
                </xsd:annotation>
            </xsd:attribute>
            <xsd:attribute name="comment" type="xsd:string"/>
        </xsd:extension>
    </xsd:complexContent>
</xsd:complexType>
<xsd:complexType name="ScheduledPackageIdRefType">
    <xsd:attribute name="ref" type="xsd:IDREF"/>
</xsd:complexType>
<xsd:complexType name="ExternalSchedPkgIdRefType">
    <xsd:attribute name="ref" type="xsd:string"/>
</xsd:complexType>
<xsd:simpleType name="ServiceType">
    <xsd:restriction base="xsd:string">
        <xsd:enumeration value="APA-AZ/EL"/>
        <xsd:enumeration value="APA-X/Y"/>
        <xsd:enumeration value="DELTADOR"/>
        <xsd:enumeration value="DOPPLER"/>
        <xsd:enumeration value="OFFLINE-TM-RECORDING"/>
        <xsd:enumeration value="OFFLINE-TM-PROVISION"/>
        <xsd:enumeration value="RF-ONLY"/>
        <xsd:enumeration value="RANGING"/>
        <xsd:enumeration value="RESERVED"/>
        <xsd:enumeration value="TBD"/>
        <xsd:enumeration value="TELECOMMAND"/>
        <xsd:enumeration value="TELEMETRY"/>
        <xsd:enumeration value="TEST"/>
        <xsd:enumeration value="UNUSED"/>
        <xsd:enumeration value="VLBI"/>
    </xsd:restriction>

```

```

</xsd:simpleType>
<xsd:simpleType name="FrequencyBandType">
  <xsd:restriction base="xsd:string">
    <xsd:enumeration value="VHF" />
    <xsd:enumeration value="UHF" />
    <xsd:enumeration value="C-BAND" />
    <xsd:enumeration value="Ka-BAND" />
    <xsd:enumeration value="Ku-BAND" />
    <xsd:enumeration value="L-BAND" />
    <xsd:enumeration value="S-BAND" />
    <xsd:enumeration value="V-BAND" />
    <xsd:enumeration value="X-BAND" />
    <xsd:enumeration value="OPTICAL" />
    <xsd:enumeration value="N/A" />
    <xsd:enumeration value="ALL" />
  </xsd:restriction>
</xsd:simpleType>
<xsd:complexType name="ServiceInfoType">
  <xsd:sequence>
    <!-- allow a list of parameters derived from AbstractParameter -->
    <xsd:element ref="AbstractParameter" minOccurs="0"
maxOccurs="unbounded" />
  </xsd:sequence>
  <xsd:attribute name="serviceType" type="ServiceType" use="required" />
  <xsd:attribute name="frequencyBand" type="FrequencyBandType"
use="required" />
</xsd:complexType>
</xsd:schema>

```

G4 EXAMPLES

G4.1 OVERVIEW

The following subsections give examples of using the simple schedule format.

G4.2 EXAMPLE OF SIMPLE SCHEDULE WITHOUT EXTENSIONS

The following is a listing of a simple schedule which does not use extensions.

```
<?xml version="1.0" encoding="UTF-8"?>
<simpleSchedule xmlns="urn:ccsds:schema:cssm:1.0.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xmi="http://www.omg.org/XMI"
  xsi:schemaLocation="urn:ccsds:schema:cssm:1.0.0
SchemaCssmSimpleSchedule-v1_0_0_b1.xsd" >
  <srvMgtEntityType>SIMPLE_SCHEDULE</srvMgtEntityType>
  <simpleScheduleHeader originatingOrganization="RKPNI"
    generationTime="2016-307T13:46:48.247Z"
    status="TEST"
    inclusionType="OVERLAP_INCLUSION"
    version="1"
    startTime="2016-134T00:00:00.000Z"
    endTime="2016-164T00:00:00.000Z"
    purpose="Purpose of the schedule."
    description="Schedule description."/>
  <scheduledPackage scheduledPackageId="EVENT-2015.190.11.41.47.243348-821429"
    user="GAIA"
    originatingRequestId ="GAIA_Dummy_Request_001">
    <servicePackageXRef
serviceAgreementRef="GAIA_Routine_1.daily_TTC.nominal_definition.GAIA"
    servicePackageRef="EVENT-2015.190.11.41.47.243348-
821429"/>
    <scheduledActivity scheduledActivityId="EVENT-2015.190.11.41.47.243348-
821429"
      activityStatus="COMMITTED"
      siteRef="MLG"
      apertureRef="MLG"
      beginningOfActivity="2016-134T02:47:52.000Z"
      beginningOfTrack="2016-134T03:32:52.000Z"
      endOfTrack="2016-134T10:10:29.000Z"
      endOfActivity="2016-134T10:25:29.000Z">
      <serviceInfo serviceType="TELEMETRY"
        frequencyBand="X-BAND" >
      </serviceInfo>
    </scheduledActivity>
  </scheduledPackage>
  <scheduledPackage scheduledPackageId="EVENT-2015.190.10.48.15.165576-808510"
    user="LIPF"
    originatingRequestId ="LIPF_Dummy_Request_001">
    <servicePackageXRef
serviceAgreementRef="Lisa_Pathfinder_1.daily_TTC.nominal_definition.LIPF"
    servicePackageRef="EVENT-2015.190.10.48.15.165576-
808510"/>
    <scheduledActivity scheduledActivityId="EVENT-2015.190.10.48.15.165576-
808510"
      activityStatus="COMMITTED"
```



```

        siteRef="CEB"
        apertureRef="CEB"
        beginningOfActivity="2016-134T05:15:00.000Z"
        beginningOfTrack="2016-134T06:00:00.000Z"
        endOfTrack="2016-134T14:00:00.000Z"
        endOfActivity="2016-134T14:15:00.000Z">
    <serviceInfo serviceType="TELEMETRY"
        frequencyBand="X-BAND" >
    </serviceInfo>
</scheduledActivity>
</scheduledPackage>
<scheduledPackage scheduledPackageId="EVENT-2015.190.13.59.48.510482-846702"
    user="ROS"
    comment="This is an example of an optional comment for a
scheduled package..."
    originatingRequestId ="ROS_Dummy_Request_001">
    <servicePackageXRef
serviceAgreementRef="ROSETTA_Routine_2.Continuous_TTC.nominal_definition.ROS"
        servicePackageRef="EVENT-2015.190.13.59.48.510482-
846702"/>
    <scheduledActivity scheduledActivityId="EVENT-2015.190.13.59.48.510482-
846702"
        activityStatus="COMMITTED"
        siteRef="NNO"
        apertureRef="NNO"
        beginningOfActivity="2016-134T06:36:08.000Z"
        beginningOfTrack="2016-134T07:21:08.000Z"
        endOfTrack="2016-134T16:05:48.000Z"
        endOfActivity="2016-134T16:20:48.000Z">
    <serviceInfo serviceType="TELEMETRY"
        frequencyBand="X-BAND" >
    </serviceInfo>
</scheduledActivity>
</scheduledPackage>
<scheduledPackage scheduledPackageId="TEST-SC1_Scheduled_Package_001"
    user="TEST-SC1"
    originatingRequestId ="TEST-SC1_Dummy_Request_001">
    <servicePackageXRef serviceAgreementRef="TEST-SC1_Service_Agreement"
        servicePackageRef="TEST-SC1_Service_Package_001"/>
    <scheduledActivity scheduledActivityId="TEST-SC1_Schedule_Activity_001"
        activityStatus="TENTATIVE"
        siteRef="NNO"
        apertureRef="NNO"
        beginningOfActivity="2016-136T16:00:00.000Z"
        beginningOfTrack="2016-136T17:16:30.000Z"
        endOfTrack="2016-136T22:00:00.000Z"
        endOfActivity="2016-136T22:30:00.000Z">
    <serviceInfo serviceType="TELEMETRY"
        frequencyBand="X-BAND" >
    </serviceInfo>
    <serviceInfo serviceType="TELECOMMAND"
        frequencyBand="X-BAND" >
    </serviceInfo>
</scheduledActivity>
</scheduledPackage>
<scheduledPackage scheduledPackageId="TEST-SC2_Scheduled_Package_001"
    user="TEST-SC2"
    originatingRequestId ="TEST-SC2_Dummy_Request_001">
    <servicePackageXRef serviceAgreementRef="TEST-SC2_Service_Agreement"

```

```

        servicePackageRef="TEST-SC2_Service_Package_001" />
<scheduledActivity scheduledActivityId="TEST-SC2_Schedule_Activity_001"
    activityStatus="TENTATIVE"
    siteRef="NNO"
    apertureRef="NNO"
    beginningOfActivity="2016-136T17:00:00.000Z"
    beginningOfTrack="2016-136T17:30:00.000Z"
    endOfTrack="2016-136T23:00:00.000Z"
    endOfActivity="2016-136T23:30:00.000Z">
    <serviceInfo serviceType="TELEMETRY"
        frequencyBand="X-BAND" >
    </serviceInfo>
</scheduledActivity>
</scheduledPackage>
<scheduledPkgAssn associationId = "Association_TEST-
SC1_Schedule_Package_001_and_TEST-SC2_Schedule_Package_001"
    associationKind = "MSPA"
    comment = "MSPA pass on New Norcia for TEST-SC1 and TEST-SC2"
>
    <scheduledPackageIdRef ref="TEST-SC1_Scheduled_Package_001" />
    <scheduledPackageIdRef ref="TEST-SC2_Scheduled_Package_001" />
    <externalSchedPkgIdRef
ref="This_is_an_example_of_a_reference_to_an_external_Scheduled_Package_i.e._One_
that_is_not_contained_in_the_same_XML_file" />
</scheduledPkgAssn>
</simpleSchedule>

```

G4.3 EXAMPLE OF SIMPLE SCHEDULE WITH EXTENSIONS

The following is a listing of a simple schedule which uses extensions. Examples of the use of the extended and string additional parameters are given.

```
<?xml version="1.0" encoding="UTF-8"?>
<simpleSchedule xmlns="urn:ccsds:schema:cssm:1.0.0"
  xmlns:fn="http://www.w3.org/2005/02/xpath-functions"

  xmlns:emsmodel="platform:/resource/esa.egos.ems.portal.model/schemas/Ems.xsd"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xmi="http://www.omg.org/XMI"
  xmlns:ems="http://esa.egos.ems"
  xsi:schemaLocation="urn:ccsds:schema:cssm:1.0.0
  SchemaCssmSimpleSchedule-v1_0_0_b1.xsd http://esa.egos.ems EmsExternal.xsd">
  <srvMgtEntityType>SIMPLE_SCHEDULE</srvMgtEntityType>
  <simpleScheduleHeader originatingOrganization="RKPNI"
    generationTime="2016-307T13:46:48.247Z"
    status="TEST"
    inclusionType="OVERLAP_INCLUSION"
    version="1"
    startTime="2016-134T00:00:00.000Z"
    endTime="2016-164T00:00:00.000Z"
    purpose="Purpose of the schedule..."
    description="Schedule description..."/>
  <scheduledPackage scheduledPackageId="EVENT-2015.190.11.41.47.243348-821429"
    user="GAIA"
    comment="ESOC MCP">
    <servicePackageXRef
  serviceAgreementRef="GAIA_Routine_1.daily_TTC.nominal_definition.GAIA"
    servicePackageRef="EVENT-2015.190.11.41.47.243348-821429"/>
    <extendedParameter name="configurationProfiles">
      <ems:configuration_profile default="true"
        requirement="TM_requirement"
        service="telemetryReception">
        <configuration_profile_name
  xmlns="">default_setup</configuration_profile_name>
        <parameter xmlns="" identifier="Throw_event" type="enum">
          NO </parameter>
        </ems:configuration_profile>
      </extendedParameter>
      <stringParameter value="This is an example of an additional string
  parameter in a schedule package..."/>
    <scheduledActivity scheduledActivityId="EVENT-2015.190.11.41.47.243348-
  821429"
      activityStatus="COMMITTED"
      siteRef="MLG"
      apertureRef="MLG"
      beginningOfActivity="2016-134T02:47:52.000Z"
      beginningOfTrack="2016-134T03:32:52.000Z"
      endOfTrack="2016-134T10:10:29.000Z"
      endOfActivity="2016-134T10:25:29.000Z">
      <serviceInfo serviceType="TELEMETRY"
        frequencyBand="X-BAND" />
    </scheduledActivity>
  </scheduledPackage>

</simpleSchedule>
```

ANNEX H

ABBREVIATIONS AND ACRONYMS

(INFORMATIVE)

ASCII	American Standard Code for Information Interchange
CCSDS	Consultative Committee on Space Data Systems
CSSM	Cross Support Service Management
CSSS	Cross Support Service System
Delta-DOR	delta-differential one-way range
ICD	interface control document
ICS	implementation conformance statement
IUT	implementation under test
MSPA	multiple spacecraft per aperture
NCName	non-colonized name
OMG	Object Management Group
RL	requirements list
SANA	Space Assigned Numbers Authority
SCCS SM	Space Communication Cross Support Service Management
TBD	to be decided
UML	Unified Modeling Language
UTC	Coordinated Universal Time
VLBI	very-long-baseline interferometry
W3C	World Wide Web Consortium
XML	eXtensible Markup Language

ANNEX I

INFORMATIVE REFERENCES

(INFORMATIVE)

- [11] *Space Communication Cross Support—Service Management—Operations Concept*. Issue 1. Report Concerning Space Data System Standards (Green Book), CCSDS 910.14-G-1. Washington, D.C.: CCSDS, May 2011.
- [12] *Extensible Space Communication Cross Support—Service Management—Concept*. Report Concerning Space Data System Standards (Green Book). CCSDS 902.0-G-1, Washington, D.C.: CCSDS, September 2014.

ANNEX J**FREQUENCY BAND DEFINITIONS****(INFORMATIVE)**

The frequency band definitions below are for informative purposes only; the actual details of the specific frequencies used for uplink, downlink, etc., will be specified in the service agreement. The following definitions, which do not distinguish between uplink or downlink or specify sub-bands, are thus sufficient for the purposes of this Recommended Standard.

Band	Range (MHz)	Comment
VHF	30–300	
UHF	300–1000	
C-Band	3400–6725	
Ka-Band	18000–35000	
Ku-Band	10700–14500	
L-Band	1215–1850	
S-Band	2025–2400	
V-Band	37500–50200	
X-Band	7025–8500	
Optical ³		

³ This is essentially a placeholder. It may need further refinement as optical communications standards mature.