

BS EN 16603-70-31:2015



BSI Standards Publication

Space engineering — Ground systems and operations — Monitoring and control data definition

bsi.

...making excellence a habit.
AVA Documents™
Document Sharing Hub

National foreword

This British Standard is the UK implementation of EN 16603-70-31:2015.

The UK participation in its preparation was entrusted to Technical Committee ACE/68, Space systems and operations.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2015.
Published by BSI Standards Limited 2015

ISBN 978 0 580 86758 3

ICS 49.140

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 January 2015.

Amendments/corrigenda issued since publication

Date	Text affected
------	---------------

English version

Space engineering - Ground systems and operations - Monitoring and control data definition

Ingénierie spatiale - Systèmes sol et opérations - Définition
des données de commande et contrôle

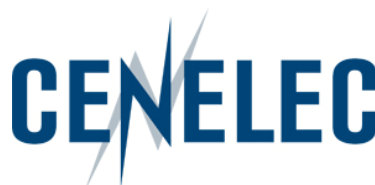
Raumfahrttechnik - Bodensysteme und Bodenbetrieb -
Überwachung und Definitionen zu Steuerdaten

This European Standard was approved by CEN on 23 November 2014.

CEN and CENELEC members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CEN and CENELEC member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN and CENELEC member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

CEN and CENELEC members are the national standards bodies and national electrotechnical committees of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.



CEN-CENELEC Management Centre:
Avenue Marnix 17, B-1000 Brussels

Table of contents

Foreword	8
Introduction.....	8
1 Scope.....	10
2 Normative references	11
3 Terms, definitions and abbreviated terms.....	12
3.1 Terms from other standards.....	12
3.2 Terms specific to the present standard	12
3.3 Abbreviated terms.....	13
4 Background and context.....	15
4.1 The space system model	15
4.2 Monitoring and control view of the SSM.....	16
4.2.1 Introduction	16
4.2.2 Standard SSM definitions.....	16
4.2.3 Product-specific SSM tailoring	18
5 Conventions.....	20
5.1 Data definition.....	20
5.2 Railroad diagrams.....	21
5.3 Case sensitivity.....	22
5.4 Names	22
5.5 Data types	24
5.5.1 General	24
5.5.2 Simple Data Types.....	25
5.5.3 Complex Data Types.....	37
6 Monitoring and control data requirements.....	38
6.1 Data exchange	38
6.2 Specification of complex data types.....	39
6.2.1 General	39
6.2.2 Activity call	39

6.2.3	Expression	40
6.2.4	Interpretation function	40
6.2.5	Procedure	41
6.2.6	Synthetic parameter	41
6.2.7	Value set	42
6.3	Product data	42
6.3.1	Introduction	42
6.3.2	Product configuration data	43
6.4	Data population	47
6.5	System element data	49
6.5.1	Introduction	49
6.5.2	System element generic data	50
6.5.3	System data	51
6.5.4	Application process data	53
6.5.5	Service data	55
6.5.6	MAP data	93
6.5.7	VC data	93
6.5.8	Functions	94
6.5.9	Memory data	95
6.5.10	Memory sub-block data	95
6.5.11	Store data	96
6.5.12	CPDU data	97
6.5.13	On/Off device data	97
6.5.14	Register load device data	98
6.5.15	Sensor data	98
6.6	Reporting data	99
6.6.1	Introduction	99
6.6.2	General	99
6.6.3	Parameters	101
6.6.4	Compound parameters	104
6.6.5	Synthetic reporting data	106
6.6.6	Checking data	106
6.7	Activities	111
6.7.1	General	111
6.7.2	Activity argument value set	114
6.7.3	Activity execution data	114
6.7.4	Telecommands	118

6.7.5	Procedures	123
6.8	Events	125
Annex A (informative) PUS service tailoring.....		126
A.1	Introduction.....	126
A.2	Telecommand verification service.....	127
A.2.1	The PUS service model	127
A.2.2	Service tailoring data.....	128
A.2.3	Service requests and reports	129
A.3	Device command distribution service.....	131
A.3.1	The PUS service model	131
A.3.2	Service tailoring data.....	133
A.3.3	Service requests and reports	134
A.4	Housekeeping and diagnostic data reporting service	134
A.4.1	The PUS service model	134
A.4.2	Service tailoring data.....	136
A.4.3	Service requests and reports	142
A.5	Parameter statistics reporting service	148
A.5.1	The PUS service model	148
A.5.2	Service tailoring data.....	148
A.5.3	Service requests and reports	150
A.6	Event reporting service	151
A.6.1	The PUS service model	151
A.6.2	Service tailoring data.....	152
A.6.3	Service requests and reports	152
A.7	Memory management service.....	153
A.7.1	The PUS service model	153
A.7.2	Service tailoring data.....	154
A.7.3	Service requests and reports	156
A.8	Function management service	158
A.8.1	The PUS service model	158
A.8.2	Service tailoring data.....	158
A.8.3	Service requests and reports	159
A.9	Time management service.....	159
A.9.1	The PUS service model	159
A.9.2	Service tailoring data.....	160
A.9.3	Service requests and reports	160
A.10	On-board operations scheduling service	161

A.10.1	The PUS service model	161
A.10.2	Service tailoring data.....	162
A.10.3	Service requests and reports	164
A.11	On-board monitoring service.....	169
A.11.1	The PUS service model	169
A.11.2	Service tailoring data.....	169
A.11.3	Service requests and reports	173
A.12	Large data transfer service	177
A.12.1	The PUS service model	177
A.12.2	Service tailoring data.....	177
A.12.3	Service requests and reports	180
A.13	Packet forwarding control service	182
A.13.1	The PUS service model	182
A.13.2	Service tailoring data.....	182
A.13.3	Service requests and reports	184
A.14	On-board storage and retrieval service	187
A.14.1	The PUS service model	187
A.14.2	Service tailoring data.....	188
A.14.3	Service requests and reports	192
A.15	Test service	195
A.15.1	The PUS service model	195
A.15.2	Service tailoring data.....	195
A.15.3	Service requests and reports	196
A.16	On-board operations procedure service.....	196
A.16.1	The PUS service model	196
A.16.2	Service tailoring data.....	196
A.16.3	Service requests and reports	197
A.17	Event/action service.....	200
A.17.1	The PUS service model	200
A.17.2	Service tailoring data.....	200
A.17.3	Service requests and reports	202
Annex B (informative)	Data type definitions	204
B.1	Conventions.....	204
B.2	Comments	205
B.3	Data types and data items	205
B.3.1	Definitions	205
B.3.2	EBNF Representation	206

B.4	Activity Call	214
B.5	EXPL - Expression Language	214
B.5.1	Definitions	214
B.5.2	EBNF Representation	216
B.6	IFL - Interpretation Function Language	218
B.6.1	Definition	218
B.6.2	EBNF Representation	220
B.7	SPEL - Synthetic Parameter Expression Language	222
B.7.1	Definitions	222
B.7.2	Bit-manipulation functions	227
B.7.3	EBNF Representation	228
B.8	PLUTO – Procedure Language	232
B.9	VAL – Value Language	233
B.9.1	Definition	233
B.9.2	EBNF Representation	234
Bibliography.....		236

Figures

Figure 4-1: Example product delivery system element hierarchy	16
Figure 4-2: Monitoring and control knowledge associated with a system element	18
Figure 5-1: Example railroad diagram.....	21
Figure A-1 : Diagram convention for PUS packet structures	127
Figure A-2 : Tailoring choices for the telecommand verification service	129
Figure A-3 : Tailoring choices for the device command distribution service	133
Figure A-4 : Tailoring choices for the housekeeping and diagnostic data reporting service (View 1).....	137
Figure A-5 Tailoring choices for the parameter statistics reporting service	149
Figure A-6 : Tailoring choices for the event reporting service	152
Figure A-7 : Tailoring choices for the memory management service (View 1).....	154
Figure A-8 : Tailoring choices for the function management service	158
Figure A-9 : Tailoring choices for the time management service.....	160
Figure A-10 : Tailoring choices for the on-board operations scheduling service (View 1)....	162
Figure A-11 : Tailoring choices for the on-board monitoring service (View 1)	170
Figure A-12 : Tailoring choices for the large data transfer service (View 1)	178
Figure A-13 : Tailoring choices for the packet forwarding control service (View 1).....	183
Figure A-14 : Tailoring choices for the on-board storage and retrieval service (View 1).....	189
Figure A-15 : Tailoring choices for the on-board operations procedure service	197

Figure A-16 Tailoring choices for the event/action service.....	201
---	-----

Foreword

This document (EN 16603-70-31:2015) has been prepared by Technical Committee CEN/CLC/TC 5 “Space”, the secretariat of which is held by DIN.

This standard (EN 16603-70-31:2015) originates from ECSS-E-ST-70-31C.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by July 2015, and conflicting national standards shall be withdrawn at the latest by July 2015.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

This document has been prepared under a mandate given to CEN by the European Commission and the European Free Trade Association.

This document has been developed to cover specifically space systems and has therefore precedence over any EN covering the same scope but with a wider domain of applicability (e.g. : aerospace).

According to the CEN-CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

Introduction

As described in ECSS-S-ST-00 and ECSS-E-ST-10, the development of a space system is an incremental task involving different entities, who can participate as customer or supplier at different levels of space system integration.

Documentation and data of different types is exchanged between supplier and customer. The purpose of this Standard is to define the data to be provided by the supplier to the customer in order to be able to monitor and control the product delivered. Formally, this data is part of the user manual for the corresponding element of the space system (see ECSS-E-ST-70).

1

Scope

This Standard defines the monitoring and control data that a supplier delivers together with a product in order to allow a customer to perform space system integration, testing and mission operations.

The requirements in this Standard are defined in terms of *what* data is provided by the supplier to the customer. *How* this data is provided (e.g. using spreadsheet data or XML) is outside of scope.

The Standard assumes that missions conform to the following ECSS standards:

- ECSS-E-ST-50 and ECSS-E-ST-70;
- ECSS-E-ST-70-41;
- ECSS-E-ST-70-32.

This standard may be tailored for the specific characteristics and constraints of a space project in conformance with ECSS-S-ST-00.

2

Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this ECSS Standard. For dated references, subsequent amendments to, or revision of any of these publications do not apply. However, parties to agreements based on this ECSS Standard are encouraged to investigate the possibility of applying the more recent editions of the normative documents indicated below. For undated references, the latest edition of the publication referred to applies.

EN reference	Reference in text	Title
EN 16603-50	ECSS-E-ST-50	Space engineering - Communications
EN 16603-70	ECSS-E-ST-70	Space engineering - Ground systems and operations -
EN 16603-70-01	ECSS-E-ST-70-01	Space engineering - On board control procedures
EN 16603-70-11	ECSS-E-ST-70-11	Space engineering - Space segment operability
EN 16603-70-32	ECSS-E-ST-70-32	Space engineering - Test and operations procedure language
EN 16603-70-41	ECSS-E-ST-70-41	Space engineering - Telemetry and telecommand packet utilization

Terms, definitions and abbreviated terms

3.1 Terms from other standards

For the purpose of this Standard, the terms and definitions from ECSS-S-ST-00-01 and ECSS-E-ST-70 apply, in particular for the following terms:

anomaly

assembly

availability

contingency procedure

emergency

mission

procedure

space system

subsystem

system

test

validation

verification

3.2 Terms specific to the present standard

3.2.1 activity

space system monitoring and control function

3.2.2 compound parameter

record comprised of any sequence of **reporting data**, arrays of **reporting data** and sub-records that are interpreted together

NOTE E.g.: an anomaly report generated by the space segment comprising an anomaly report ID and a set of associated **parameters**.

3.2.3 event

occurrence of a condition or set of conditions that can arise during the course of a test session or mission phase

3.2.4 parameter

lowest level of elementary information that has a meaning for monitoring the space system

3.2.5 reporting data

data used for assessing the functioning of the space system

NOTE Reporting data can consist of a **parameter** (a simple type) or a **compound parameter** (a complex type).

3.2.6 resource

stock or supply, either depletable or shareable in nature, that can be drawn upon, or provided by, an element of the space system during operation

3.2.7 space system model

representation of the space system in terms of its decomposition into **system elements**, the **activities** that can be performed on these **system elements**, the **reporting data** that reflects the state of these **system elements** and the **events** that can be raised and handled for the control of these **system elements**, **activities** or **reporting data**

3.2.8 synthetic parameter

reporting data generated within the monitoring and control system by means of an expression which may use other **reporting data** and constants as input

3.2.9 system element

representation within the **space system model** of a functional element of the space system

3.3 Abbreviated terms

For the purpose of this standard, the abbreviated terms of ECSS-S-ST-00-01 and the following apply:

Abbreviation	Meaning
AD	acceptance of data
AOCS	attitude and orbit control subsystem
APID	application process identifier
BD	bypass of data
CDMU	command and data management unit
CI	configuration item
COTS	commercial off-the-shelf

CPDU	command pulse distribution unit
CRC	cyclic redundancy check
EBNF	extended Backus-Naur form
EM	engineering model
EMCS	EGSE or mission control system
EXPL	expression language
FID	function identifier
FM	flight model
FTP	file transfer protocol
ID	identifier
IFL	interpretation function language
MAP	multiplexed access point
OBSM	on-board software maintenance
OBT	on-board time
PC	parameter code
PFC	parameter format code
PTC	parameter type code
PLUTO	procedure language for users in test and operations
PUS	packet utilization standard
RID	report identifier
SAU	smallest addressable unit
SE	system element
SID	structure identifier
SPEL	synthetic parameter expression language
SSM	space system model
STM	structural model
TDM	time-division multiplexed
UTC	universal time coordinated
VAL	value definition language
VC	virtual channel

4

Background and context

4.1 The space system model

Throughout the space system lifecycle, suppliers deliver “Products” to customers. A product consists of hardware component, software component, or both, together with associated documentation containing all the knowledge required by the customer during the complete lifecycle of the product (design, development, integration, testing and operation) and across all domains of expertise (quality, management and engineering).

NOTE A prime contractor can deliver a product (e.g. documentation) to a subcontractor. In this context, the prime contractor is the supplier and the subcontractor is the customer.

To facilitate the sharing and reuse of the product knowledge by the customer, the documentation shall be organised according to a formal structure.

For this purpose, this Standard introduces the concept of a space system model (SSM) that is structured so as to be able to capture the space system knowledge. This structure reflects the structure of the space system itself. The SSM is hierarchically broken down into system elements (SE) mirroring the functional breakdown of the space system. A system element is a data structure whose properties are the means to capture the space system knowledge.

System elements correspond to the elements of the space system resulting from the functional decomposition. From the highest level downwards, these are progressively: system, subsystem, set, equipment or software product, assembly, part (hardware) or module (software).

An example of the SSM corresponding to a product delivery (an attitude and orbit control subsystem) is shown in Figure 4-1.

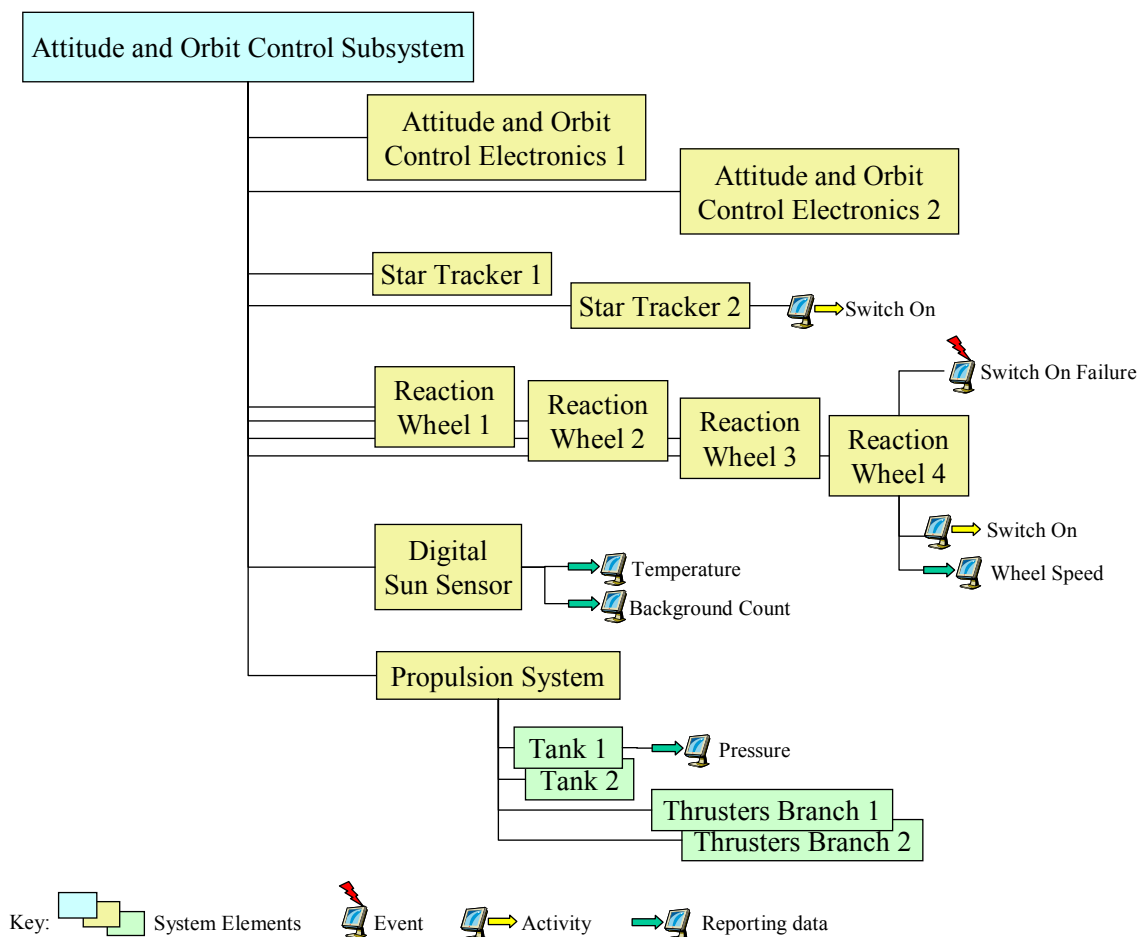


Figure 4-1: Example product delivery system element hierarchy

4.2 Monitoring and control view of the SSM

4.2.1 Introduction

According to the utilisation of the product, actors in different domains will require different “domain-specific views” of the SSM, where a view is the corresponding SE hierarchy (or hierarchies) and the sub-set of SE characteristics relevant to the domain.

The SSM consists of:

- “entity” types e.g. system element is an entity type.
- “value” types e.g. APID is a value type.

4.2.2 Standard SSM definitions

The SSM types that are relevant for monitoring and control purposes are system elements and their associated activities, reporting data, events and constituent system elements.

- An activity is a space system monitoring and control function implemented within the EGSE or mission control system (EMCS). An activity can be implemented as a telecommand either to the space segment or to the ground segment, a procedure, an operating system command (e.g. a printer request, sending an email, transferring a file using FTP) or any other command type that is specific to a given implementation of the space system (e.g. a command to a special check-out system (SCOE) or to a ground station using a proprietary protocol).
- Reporting data is information that a system element provides, irrespective of how this information is used. Reporting data can comprise measurements which reflect the state of the associated system element or an output product whose purpose is to be used by another system element (e.g. manoeuvre parameters provided by the flight dynamics system). Reporting data comprises parameters and compound parameters. A parameter is the lowest level of elementary information that has a meaning for monitoring and control of the space system. A compound parameter is a record comprised of any sequence of parameters, arrays of parameters and sub-records (see also ECSS-E-ST-70-41). For example, a complete telemetry packet, or part thereof, may be represented as a compound parameter. The parameters within a compound parameter are normally interpreted together (e.g. when interpreting the contents of an anomaly report). Reporting data can have different representations depending on its life cycle within the space system (e.g. an on-board measurement has a raw value in telemetry and an engineering value when presented on a ground segment display).
- An event is an occurrence of a condition or group of conditions of operational significance. Events are widely used within the space system to trigger the execution of functions (e.g. acquisition of signal can initiate telemetry processing tasks at the ground station). Users can define mission-specific events, associated with a system element, for example for use within procedures.

In turn, activities, reporting data and events may have their own value and entity types e.g. an activity has a name (value type) and arguments (entity type).

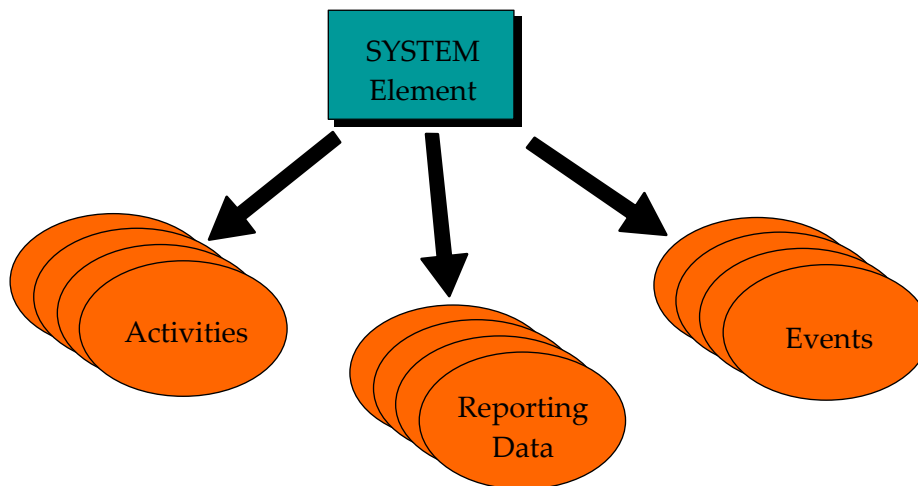


Figure 4-2: Monitoring and control knowledge associated with a system element

Each actor in the overall space system development life cycle has its own view of the SSM hierarchy, which corresponds to its needs and which is comprised of:

- one or more root system elements (a root system element corresponds to a node of the overall SSM hierarchy), and
- one or more corresponding system element hierarchies i.e. for each root system element, a view of the decomposition of this system element which is limited in scope to its level of interest.

NOTE E.g.: a payload manufacturer only knows about the hierarchy of system elements that correspond to its payload and its EGSE.

The concepts of system elements and their characteristics introduced above enable a complete, high-level description of the space system model independent of any specific space system configuration and, as such, can be reused at any level of integration, test and mission operations. For example, the name and description of an activity always remain the same, although it may be implemented as a simulated command during testing and a procedure during mission operations.

4.2.3 Product-specific SSM tailoring

A tailoring process is required to define the SSM that corresponds to a given product. This implies selecting the applicable subset of the entity and value types identified in this Standard and adding product-specific additional types.

NOTE 1 E.g.: adding types corresponding to a 1553 bus-specific protocol.

NOTE 2 E.g.: tailoring of standard PUS (packet utilization standard, ECSS-E-ST-70-41) services for a mission and adding mission-specific services and extensions to the standard PUS services.

NOTE 3 E.g.: Accommodating an existing proprietary protocol for monitoring and controlling an antenna front-end controller.

5

Conventions

5.1 Data definition

This Standard defines the SSM data model (consisting of entity types and value types) that a supplier uses to provide the monitoring and control data for a delivered product.

The SSM data model is specified as a set of requirements in a tabular form. Each table corresponds to an entity type which is uniquely identified by the name of the table.

Tables are comprised of three columns (see Table 5-1 for an example):

- Each row corresponds to a value type.
- The first column identifies a value type by means of a name that is unique within the context of the current table.
- The second column contains the definition of the type i.e. a description and any applicable constraints.
- The third column identifies the data type of the type.

Where the arity of a value type (or a set of value types) is greater than one, the first column also specifies the corresponding lower-level entity type i.e. an ordered list ("Ordered list of...") or an unordered list ("List of...").

NOTE 1 Where a condition is expressed for the provision of a given type, the corollary is that the type is NOT provided if the condition is not TRUE.

NOTE 2 Enumerated values and keywords are shown in bold in the tables.

Table 5-1 Example data requirement table

Name		Definition	Data Type
Type		The type of array. The set of values is {"fixed", "variable"}.	Enumerated
Minimum Number of Repetitions		The minimum number of repetitions of the array component	Unsigned Integer
Maximum Number of Repetitions		The maximum number of repetitions of the array component	Unsigned Integer
Ordered list of	Reporting Data	The reporting data that comprises the component	Reporting Data Reference

Within this Standard, a reference to the SSM indicates both the data model as defined by the data requirements and the corresponding mission data provided by a supplier to a customer. The term “SSM object” is used to refer to any object of the populated database that is uniquely identified by a name e.g. a system element, a reporting data, an activity or an event.

5.2 Railroad diagrams

A graphical convention is used to define the syntax of language-based data type constructs within the remainder of this Standard. This convention is known as a “railroad” diagram, so-called because of the similarity to a railway network. An example railroad diagram is shown in Figure 5-1. The elements of this graphical convention are as follows:

- the name at the top of the diagram is that of the item being defined, also referred to as a "non-terminal symbol";
- a non-terminal symbol is a combination of zero or more non-terminal symbols and "terminal symbols", where a terminal symbol is a sequence of one or more characters forming an irreducible element (i.e. it cannot be decomposed further);
- non-terminal symbols are enclosed in rectangles;
- terminal symbols are enclosed in round-cornered boxes (in the limit, a circle);
- the main line corresponds to mandatory elements of the data type or construct;
- branch lines correspond to optional elements;
- “return” lines correspond to optional repetitions of one or more elements.

Integer Constant =

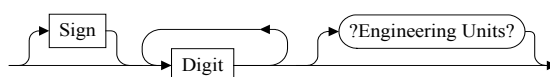
**Figure 5-1: Example railroad diagram**

Figure 5-1 defines that an integer constant starts with an optional sign followed by one or more digits, followed by optional engineering units.

NOTE The “?” is a special-sequence-symbol which indicates the start and end of a special sequence (see ISO/IEC 14977); in this case a standalone syntax for Engineering Units.

5.3 Case sensitivity

All words used in expressing the data requirements in this Standard are case-insensitive.

5.4 Names

As a result of the distributed approach employed for space system development, the uniqueness of names assigned by a given supplier cannot be guaranteed at a higher-level of integration. For this reason, this Standard introduces the concepts of relative names and reference paths.

A relative name is a name which is unique within a local context. When the context changes and that uniqueness is compromised, the relative name is rendered unique by attaching a reference path.

A relative name is an identifier comprised of one or more identifier words separated by spaces. An identifier word is a sequence of letters (in upper or lower case) and digits.

NOTE E.g.: Star Tracker

A reference path is also a name comprised of one or more identifier words separated by spaces. A reference path itself can be comprised of a relative name, unique within a lower-level context, and a reference path rendering it unique. The keyword "**of**" is used to separate the left-hand part of the name (i.e. the relative name) from the right-hand part of the name (i.e. the reference path).

NOTE E.g.: X Thruster of Branch 1

For entities that are associated with a parent entity (e.g. reporting data are associated with system elements, activity arguments are associated with activities) the uniqueness of the name can be achieved in two ways: either by assigning a name that is unique at the level of the child entity (e.g. a reporting data) or by using the parent entity in a reference path.

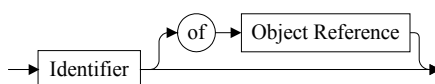
NOTE 1 E.g.: XGyro temperature

NOTE 2 E.g.: temperature of XGyro

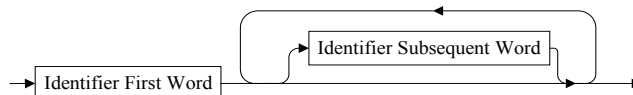
There are no constraints in the use of the keyword "**of**" in naming entities, except the need to maintain the uniqueness of names with a given product. This implies, for example, that “Temperature of XGyro” can either represent an absolute name of a reporting data or a relative name of a reporting data using “of XGyro” as the reference path.

The syntax for defining names is the following:

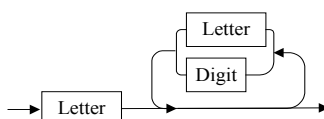
Name =



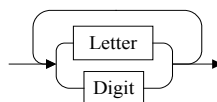
Identifier =



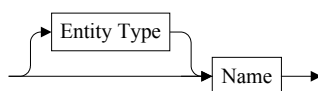
Identifier First Word =



Identifier Subsequent Word =



Object Reference =



where:

- Letter is an upper-case or lower-case letter of the alphabet.
- Digit is one of the decimal characters from 0 to 9.
- Identifiers are case-insensitive.
- Entity Type includes one of “system element”, “activity”, “reporting data”, “parameter” or “event”. The naming convention defined in this clause 5.4 is also used within this Standard to uniquely identify entity types, value types and data types within the SSM data model. This facilitates direct references from supplied data to elements of the SSM data model, e.g. reference to a data type when declaring variables in a test procedure script. For this purpose, Entity Type also includes any entity type referenced within the SSM data model.

Examples of SSM data model names are:

- “System Element” which uniquely identifies an entity type within the data model.
- “Severity of Event” which uniquely identifies the value type “Severity” of the entity type “Event”.
- “Data Type of current parameter” which uniquely identifies the data type of the parameter in whose context it is defined.

5.5 Data types

5.5.1 General

The data types used within this Standard are classified as “simple” or “complex”.

- The “simple” types include:
 - the conventional types "Boolean", "integer", "unsigned integer", "real", "bit string", "octet string", "character string", "absolute time" and "relative time";
 - the types for packet fields identified in ECSS-E-ST-70-41 (the PUS) by the parameter code "PC" comprising the parameter type code "PTC" and the parameter format code "PFC". These PUS types correspond to conventional types, however their representation and scope is constrained by the associated PFC definition (e.g. the type PTC 1 PFC 0 corresponds to Boolean, with the characteristic that TRUE = 1 and FALSE = 0);
 - the enumerated types either defined within this Standard and identified by the keyword “Enumerated”, or user-defined using the “user-defined enumerated type” construct specified in clause 5.5.2.3. An enumerated type is defined by a set of values that can either be:
 - “absolute” i.e. this Standard identifies all values. Such enumerated sets are indicated with “The set of values is {“a”, “b”, “c”}”, or
 - “extensible” i.e. a system compliant with this Standard can add implementation-specific values. Such enumerated sets are indicated with “The set of values includes {“a”, “b”, “c”}”, or
 - “mission-specific” i.e. the enumerated set is comprised of mission-specific data;
 - an enumerated set comprising all populated values (or a subset of these values) of a given data item. Since a data item corresponds to a value type of an SSM object, these types are called “value type sets”;
 - a data type which is inherited from a data item which is itself a data type. These types are called “value type data types”.
- A “complex” type is identified by the name of a language and is defined by the language grammar. Data items of complex type are assigned a character string value whose syntax complies with the language grammar.

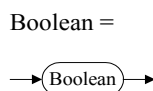
5.5.2 Simple Data Types

5.5.2.1 Conventional data types

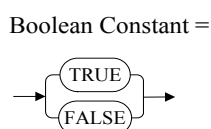
5.5.2.1.1 Boolean

This type comprises the set of truth values, which can be involved in logical operations.

The syntax used for referring to the Boolean type is the following:



The syntax for defining constants of type Boolean is the following:



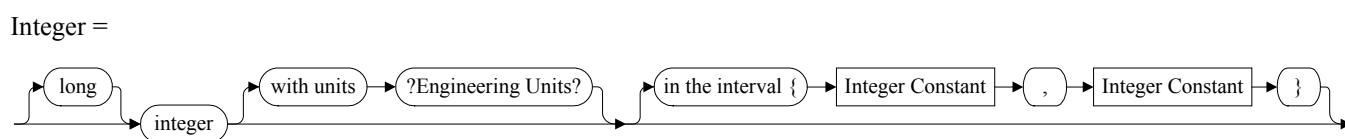
5.5.2.1.2 Integer

This type comprises a subset of whole numbers which can be involved in arithmetical, relational and comparative expressions.

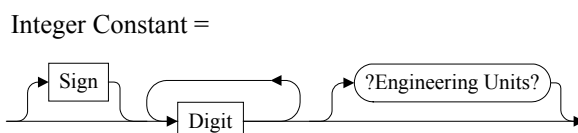
The minimum and maximum values are mission-dependent.

The type can have associated engineering units. The engineering units supported by the ECSS-E-ST-70 series of standards are specified in Annex B of ECSS-E-ST-70-32

The syntax used for referring to the integer type is the following:



The syntax for defining constants of type integer is the following:



where:

- Sign is one of the character symbols “+” or “-”.
- Digit is one of the decimal characters from 0 to 9 and ?Engineering Units? is one of the engineering units defined in Annex B of ECSS-E-ST-70-32.

When providing data of type “integer with associated engineering units”, the value is expressed together with its corresponding units.

NOTE E.g.: 23, 2056, 5 **A** (i.e. 5 amps), 65536 **B** (i.e. 65536 bytes)

5.5.2.1.3 Unsigned Integer

This type comprises a subset of whole positive numbers which can be involved in arithmetical, relational and comparative expressions.

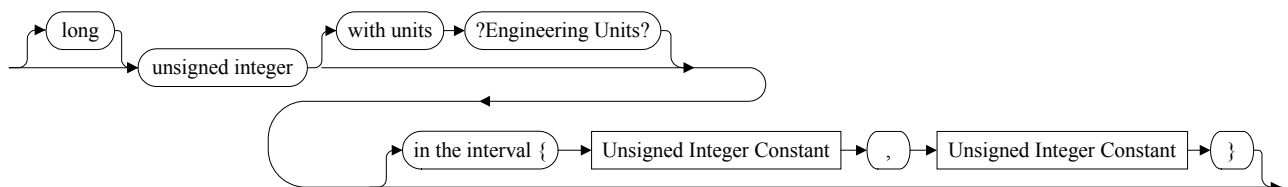
The minimum value is 0 and the maximum value is mission-dependent.

The type can have associated engineering units.

When providing data of type unsigned integer, the value can be given in decimal or in hexadecimal.

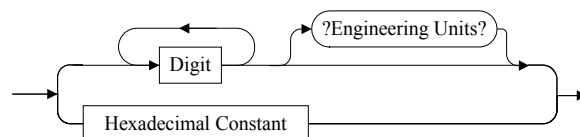
The syntax used for referring to the unsigned integer type is the following:

Unsigned Integer =



The syntax for defining constants of type unsigned integer is the following:

Unsigned Integer Constant =



where:

- Digit is one of the decimal characters from 0 to 9.
- Hexadecimal Constant is a Hexadecimal Symbol followed by one or more Hexadecimal Digits.
- Hexadecimal Symbol is the character pair symbol "0x".
- Hexadecimal Digit is a decimal character from 0 to 9 or a letter from A to F.

NOTE E.g.: 2056, **0x2056**, **0xFFFF**

When providing data of type “unsigned integer with associated engineering units”, the value is expressed together with its corresponding units.

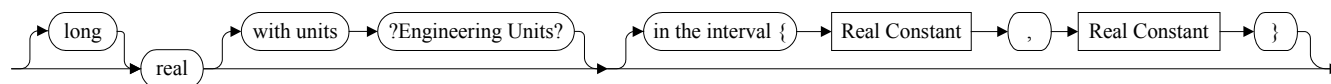
5.5.2.1.4 Real

This type comprises a set of numbers that can be represented with a floating point notation and which can be involved in arithmetical, relational and comparative expressions.

The minimum and the maximum values and the precision are mission-dependent.

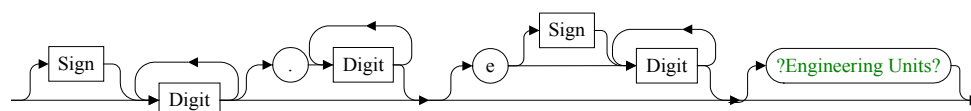
The syntax used for referring to the real type is the following:

Real =



The syntax for defining constants of type real is the following:

Real Constant =



NOTE E.g.: 123, 56e12, 0.1, 23E6

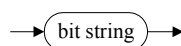
When providing data of type real with associated engineering units, the value is expressed together with its corresponding units.

5.5.2.1.5 Bit String

This type comprises a sequence of binary characters i.e. "1"s and "0"s.

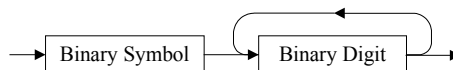
The syntax used for referring to the bit string type is the following:

Bit String =



The syntax for defining constants of type bit string is the following:

Bit String Constant =



where:

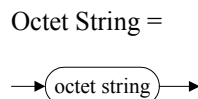
- Binary Symbol is the character pair symbol "0b".
- Binary Digit is a binary character i.e. 0 or 1.

NOTE E.g.: 0b11001

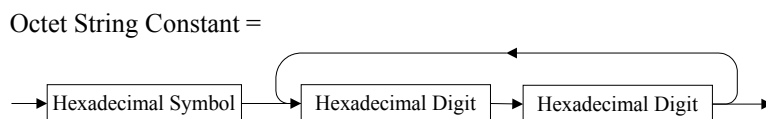
5.5.2.1.6 Octet String

This type comprises a sequence of octets, each octet being an ordered sequence of eight bits.

The syntax used for referring to the octet string type is the following:



The syntax for defining constants of type octet string is the following:



where:

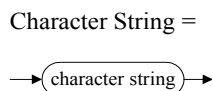
- Hexadecimal Symbol is the character pair symbol "0x".
- Hexadecimal Digit is a decimal character from 0 to 9 or a letter from A to F.

NOTE E.g.: 0x12AB

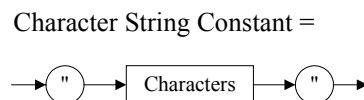
5.5.2.1.7 Character String

This type comprises any visible character string composed of allowable characters. Character strings can be involved in comparative expressions.

The syntax used for referring to the character string type is the following:



The syntax for defining constants of type character string is the following:



where Characters is any sequence of letters or digits or one of the following characters:

space ! " # \$ % & ' () * + , - . / :
; < = > ? @ [\] ^ _ ` { | } ~

NOTE 1 E.g.: "This is a string."

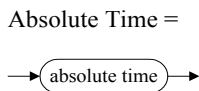
NOTE 2 In order to enter a double-quote character (") or a reverse solidus (backslash) character (\), a reverse

solidus is used as an “escape” character i.e. (\") or (\\).

5.5.2.1.8 Absolute Time

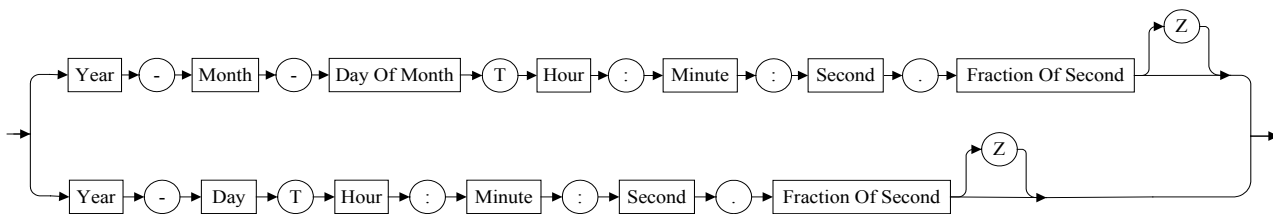
This type comprises a set of values that represents an absolute date which can be involved in arithmetical, relational and comparative expressions.

The syntax used for referring to the absolute time type is the following:



The syntax for defining constants of type absolute time conforms to the following rule:

Absolute Time Constant =



- Month/day of month calendar variation

Year-Month-Day Of MonthTHour:Minute:Second.Fraction Of Second(Z)

where:

- Year = four digits with a value in the range 0001-9999;
- Month = two digits with a value in the range 01-12;
- Day Of Month = two digits with a value in the range 01-28, 01-29, 01-30, or 01-31;
- "T" = calendar-time separator;
- Hour = two digits with a value in the range 00-23;
- Minute = two digits with a value in the range 00-59;
- Second = two digits with a value in the range 00-59 (00-58 or 00-60 during leap seconds);
- Fraction Of Second = one to n digits. To obtain a given precision, the appropriate number of digits to the right of the period is used;
- "Z" = optional terminator.

NOTE E.g.: 2001-08-18T21:07:43.137468Z

- Year/day of year calendar variation

Year-DayTHour:Minute:Second.Fraction Of Second(Z)

where:

- Year, "T", Hour, Minute, Second, Fraction Of Second, "Z": are as defined in a. above;
- Day = three digits with a value in the range 001-365 or 001-366.

NOTE 1 E.g.: 2001-033T13:21:32.226

NOTE 2 Leading zeros are included to make up the specified number of digits.

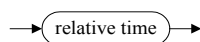
NOTE 3 Elements shown in brackets are optional.

5.5.2.1.9 Relative Time

This type comprises a set of values that represents an interval of time which can be involved in arithmetical, relational and comparative expressions.

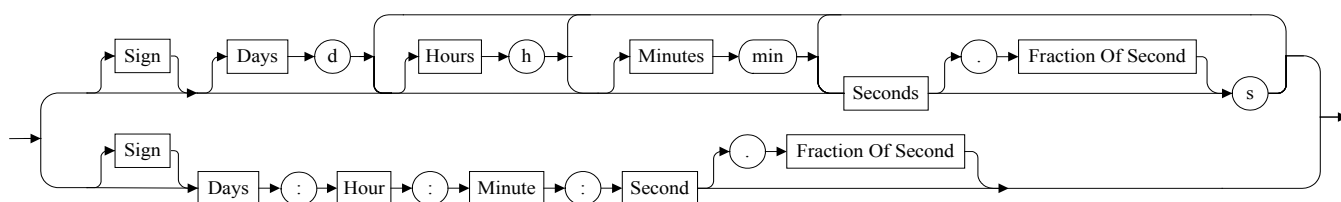
The syntax used for referring to the relative time type is the following:

Relative Time =



The syntax for defining constants of type relative time conforms to the following rule:

Relative Time Constant =



where:

- Days, Hours, Minutes and Seconds are unsigned integers;
- Hour, Minute, Second and Fraction Of Second are as defined in 5.5.2.1.8 above.

NOTE 1 E.g.: 30 h 10 min

NOTE 2 E.g.: 200 s

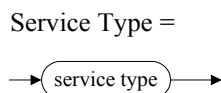
NOTE 3 E.g.: -0:00:10:00

5.5.2.2 PUS Data Types

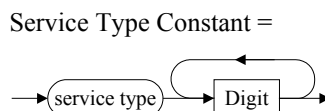
5.5.2.2.1 Service data types

PUS packets defined within ECSS-E-ST-70-41 are associated with services and each packet is assigned a "Service Type" and a "Service Subtype".

The syntax used to refer to the service type data type is the following:

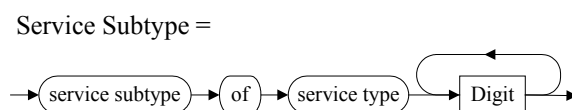


The syntax for defining constants of type service type is:

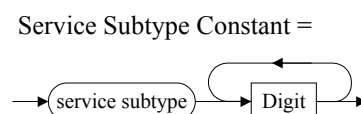


The standard service types are defined in ECSS-E-ST-70-41 and lie in the range 0 to 127. Mission-specific service types can be defined and lie in the range 128 to 255.

The syntax used to refer to the service subtype data type is the following:



The syntax for defining constants of type service subtype is:



The standard service subtypes for a given service are defined in ECSS-E-ST-70-41 and lie in the range 0 to 127. Mission-specific service subtypes can be defined for standard services and lie in the range 128 to 255 or in the range 0 to 255 for mission-specific services.

5.5.2.2.2 Parameter data types

Data within PUS packets are encoded according to a “Parameter Code (PC)” which is a combination of a “Parameter Type Code (PTC)” and a “Parameter Format Code (PFC)”. The PTC identifies the data type (e.g. Boolean, Integer). The PFC identifies the encoding format and hence constrains the values that the data may take.

The standard PUS data types are defined within ECSS-E-ST-70-41 and are as shown in Table 5-2:

Table 5-2 PUS data types

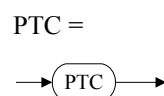
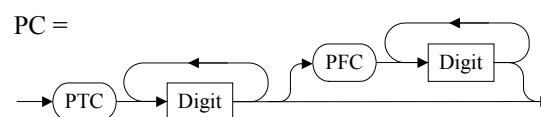
PTC	Data Type	PFC
PTC 1	Boolean	PFC 0
PTC 2	Enumerated	PFC 1 to PFC 16, PFC 24, PFC 32
PTC 3	Unsigned integer	PFC 0 to PFC 16
PTC 4	Signed integer	PFC 0 to PFC 16
PTC 5	Real	PFC 1 to PFC 4
PTC 6	Bit-string	PFC 0 to PFC any
PTC 7	Octet-string	PFC 0 to PFC any
PTC 8	Character-string	PFC 0 to PFC any
PTC 9	Absolute time	PFC 0 to PFC 18
PTC 10	Relative time	PFC 0 to PFC 16
PTC 11	Deduced	PFC 0

NOTE 1 Within ECSS-E-ST-70-41, the PFC is used to define the size of the encoded value in a packet. In this Standard, the PFC is used to express the possible range of values. For example, PTC 3 PFC 0 implies an integer value between 0 and 15. PTC 8 PFC 12 implies a fixed-length character string of 12 characters length.

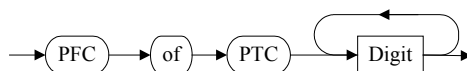
NOTE 2 When a Boolean value is used within a telecommand or a telemetry packet, the raw value 0 is used to express a false value, 1 is used to express a truth value.

Within this Standard, three derived data types are used, depending on whether the data type corresponds to the complete Parameter Code or to only a sub-set of the Parameter Code i.e. the PTC alone or the PFC alone.

The syntax used for referring to these data types is one of the following:



PFC =



The syntax for defining constants of type PC depends on the corresponding PTC and PFC values:

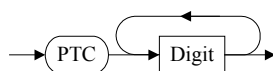
- PTC 1, PTC 2 and PTC 3 constants are unsigned integers as specified in clause 5.5.2.1.3.
- PTC 4 constants are integers as specified in clause 5.5.2.1.2.
- PTC 5 constants are reals as specified in clause 5.5.2.1.4.
- PTC 6 constants are Binary Constants consisting of a Binary Symbol (the character pair symbol "0b") followed by one or more Binary Digits (0 or 1).
- PTC 7 constants are hexadecimal constants as specified in clause 5.5.2.1.3.
- PTC 8 constants are character strings as specified in clause 5.5.2.1.7.
- PTC 9 constants are absolute times as specified in clause 5.5.2.1.8.
- PTC 10 constants are relative times as specified in clause 5.5.2.1.9.

In each case, if the PFC is specified, the range of possible values is constrained in accordance with ECSS-E-ST-70-41.

NOTE PTC 11 is not used in this Standard because all parameters of deduced type are instantiated to one of the other PUS data types identified above.

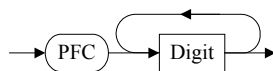
The syntax for defining constants of type PTC is:

PTC Constant =



The syntax for defining constants of type PFC is:

PFC Constant =



5.5.2.3 User-defined enumerated types

5.5.2.3.1

As introduced in clause 5.5.1, enumerated sets can be specified by end-users. These enumerated sets are simple data types that can be used when specifying, for example, activity arguments (see clause 6.7.1.2) and parameters (see clause 6.6.3.1).

For each user-defined enumerated type, the data specified in Table 5-3 shall be provided.

Table 5-3 User defined enumerated type data

Name		Definition	Data Type
Name		The name of the enumerated set	Name
List of	Value	A value of the enumerated set	Character String

5.5.2.3.2

- a. When the data items associated with these types correspond to parameters whose values are sampled within packets, the raw representation of a given value in the packet can differ from the textual representation given in Table 5-3.
- b. In the case specified in 5.5.2.3.2a the correspondence between the textual representation and its raw representation shall be defined for each element of the set, as follows.
 1. When the enumerated set is used for decoding values within telemetry packets which have a one-to-one correspondence between their raw and textual representations or for encoding telecommand arguments within telecommand packets, the data in Table 5-4 shall be provided for each element of the set.

Table 5-4 Raw correspondence data

Name	Definition	Data Type
Raw Value	The raw value of the set element.	PTC 2

NOTE E.g.: the Battery Charge Status enumerated set is comprised of a list of 3 elements, each one having a raw and textual representation as follows:

Value	Raw Value
"trickle charge"	0
"full charge"	1
"discharge"	2

2. When the enumerated set is used for decoding values within telemetry packets or for encoding telecommand arguments within telecommand packets and where there is a many-to-one correspondence between raw and textual representations:
 - (a) For each user-defined enumerated type with a many-to-one correspondence between raw and textual representations, the additional data specified in Table 5-5 shall be provided.

Table 5-5 Raw data type data

Name	Definition	Data Type
PTC	The PTC of the raw data type. The set of values is {PTC 3, PTC 4, PTC 5}	PTC

- (b) For each element of the set, the data specified in Table 5-6 shall be provided.

Table 5-6 Raw correspondence data

Name	Definition	Data Type
List of Raw Value Intervals	Minimum Raw Value	The raw value of the low boundary of the raw value interval of the set element
	Maximum Raw Value	The raw value of the upper boundary of the raw interval of the set element

NOTE E.g.: the position readout of a telescope filter wheel could be interpreted as follows:

Value	Minimum Raw Value	Maximum Raw Value	Default Raw Value
"non-operational"	0	4	0
	13	15	
"red"	5	6	5
"green"	7	8	7
"blue"	9	10	9
"UV"	11	12	11

3. When an enumerated set is used for encoding telecommand arguments within telecommand packets and where there is a many-to-one correspondence between raw and textual representations, the additional data specified in Table 5-7 shall be provided:

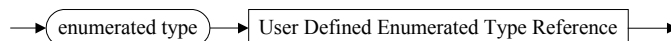
Table 5-7 Telecommand argument encoding data

Name	Definition	Data Type
Default Raw Value	The default value to be used for the argument encoding	PTC of Raw Data Type

5.5.2.3.3

The syntax used for referring to user-defined enumerated types is the following:

User Defined Enumerated Type =

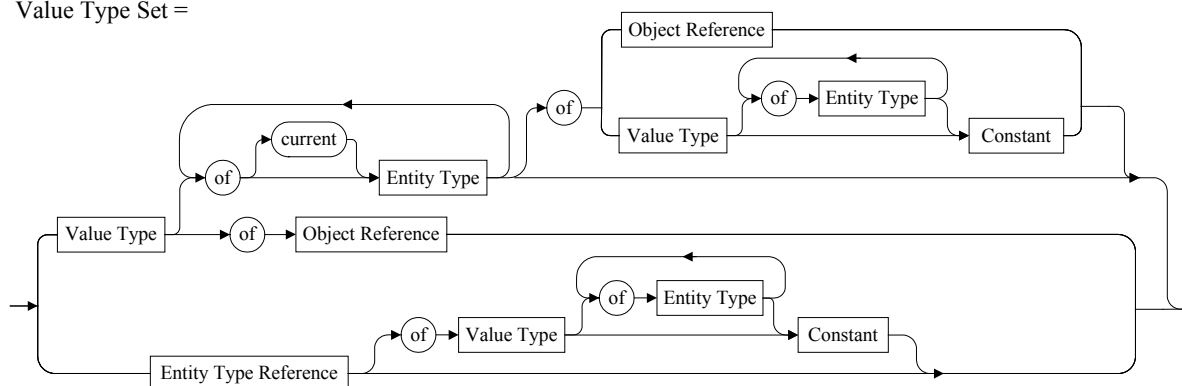


where User Defined Enumerated Type Reference is a name of a user-defined enumerated type as specified in 5.5.2.3.1.

5.5.2.4 Value Type Set

When a data item is of data type “the enumerated set comprising all populated values (or a subset of these values) of another data item”, this data type is of type value type set. The syntax used for referring to a value type set is the following:

Value Type Set =



where:

- Value Type is an Identifier corresponding to a value type of the SSM data model.
- Entity Type is an Identifier corresponding to an entity type of the SSM data model.
- Entity Type Reference is one of {"System Element Reference", "Reporting Data Reference", "Activity Reference", "Event Reference", "Activity Argument Reference"}. The value that a corresponding data item can take is one of the names associated with the specified entity type.

NOTE 1 E.g.: the enumerated set identified by “Name of Output Line of current system element”. This set comprises all output line names defined for the current CPDU as specified in Table 6-69, data item “Name of Output Line”.

NOTE 2 E.g.: the enumerated set identified by “ID of system element of Type “application process””.

This set comprises all application process IDs defined for the mission as specified in Table 6-14, data item "ID".

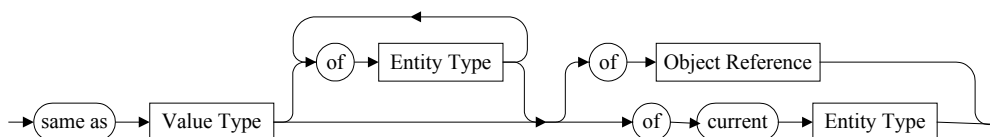
NOTE 3 E.g.: the enumerated set identified by "Name of reporting data of Power Subsystem of Cluster1". This set comprises all names of reporting data as specified in Table 6-73, data item "Name", restricted to those belonging to the Power Subsystem system element of the Cluster 1 system element.

NOTE 4 E.g.: the enumerated set identified by "system element reference of Type "application process"". This set comprises all names of application processes as specified in Table 6-8, data item "Name".

5.5.2.5 Value Type Data Type

When the type of a data item is inherited from a data item which is itself a data type, this type is of type value type data type. The syntax used for referring to a value type data type is the following:

Value Type Data Type =



NOTE E.g.: the data type of the data item Engineering Value of linear interpolation data of reporting data as specified in Table 6-78 is "same as Data Type of current parameter".

5.5.3 Complex Data Types

A complex data type is identified by the name of a language and defined by a grammar. Data items of complex type are assigned a character string value whose syntax complies with the corresponding language grammar. Within this Standard, the following complex data types are used:

- "Activity Call" to define an instantiation of an activity,
- "EXPL" to define expressions,
- "IFL" to define interpretation functions,
- "PLUTO" to define procedures,
- "SPEL" to define synthetic parameters,
- "VAL" to define values.

Monitoring and control data requirements

6.1 Data exchange

- a. This Standard specifies which data shall be delivered by a supplier for the purposes of monitoring and control of the delivered product(s). According to the nature of the product, the product data schema corresponds to:
 - a subset of the data model identified in this Standard, supplemented by
 - product-specific entity and value types.
- b. The product data schema to be used for exchange of data shall be specified.
- c. The product data schema shall be suitable for communication, interpretation and processing by computers.
- d. Formal languages are used within this Standard for the definition of data types and constants. Data delivered by suppliers to customers at any point in the overall space system development and operation lifecycle should comply with these languages. Such compliance is encouraged to optimize the process of sharing and reuse of product data.
- e. If other languages are used, these languages shall be formally specified using the extended Backus-Naur form (EBNF), see ISO/IEC 14977.
- f. The product data schema shall ensure the integrity of the data, i.e. it shall be possible to re-import data, extracted in accordance with the schema, without any loss of information.
- g. The electronic database used to transfer the product data shall be subject to configuration management.
- h. The configuration identification data for the electronic database specified in Table 6-1 shall be provided.

Table 6-1 Electronic database configuration identification data

Name		Definition	Data Type
ID		The Configuration Item (CI) identifier of the electronic database. NOTE: To ensure the system-wide uniqueness of this identifier, a common practice is to prefix the filename with an enterprise identifier.	Enumerated
Version Number		The version number of the electronic database Example: Revision 1 of Version 2	Name
Date		The date of generation	Absolute Time
ID of Supplier		The supplier identifier	Enumerated
List of Digital Files	Name	The name of a digital file within the electronic database	Name
	Type	The type of the digital file. The set of values is product-specific.	Enumerated
	Type of Checksum	The type of the checksum. The set of values is product-specific.	Enumerated
	Value of Checksum	The checksum of the file	Unsigned Integer

- i. Furthermore, the following requirements apply to any constituting digital file.
 - 1. Each digital file shall be uniquely identified.
 - 2. Each digital file shall be compressed using a standard compression tool (e.g. zip, tar).

6.2 Specification of complex data types

6.2.1 General

- a. It shall be possible to include comments at any location in the script of a complex data type.

6.2.2 Activity call

- a. It shall be possible to define the data relating to the instantiation of an activity namely, the activity reference, the values of the activity argument or a reference to a predefined set of argument values and any directives that apply for the instantiation of the activity.

NOTE 1 E.g.: to define corrective actions to be taken in specific circumstances (limit violations, activity failures etc.).

NOTE 2 E.g.: to define arguments of activities (e.g. a PUS service 11 subtype 4 request which inserts telecommand packets in the on-board schedule).

NOTE 3 E.g.: telecommand directives can include for example: the MAP ID to be used; whether the telecommand is to be sent in AD or BD mode; the specification of the telecommand verification packets to be generated in response to this telecommand.

- b. The syntax for defining data of type Activity Call should conform to that specified in Annex A 3.9.28 of ECSS-E-ST-70-32

6.2.3 Expression

- a. It shall be possible to specify data values by means of an expression.

NOTE E.g.: validity or selection conditions for reporting data or pre-conditions and post-conditions for an activity.

- b. Expressions shall be able to operate on constants and space system parameters acquired from the EMCS.
- c. Mathematical, time and string functions shall be supported.
- d. Engineering units shall be supported.
- e. The capability to assign engineering units to constants shall be supported.
- f. The capability to mix compatible units freely within an expression shall be supported
- g. The automatic conversion between different, but compatible, units shall be supported.
- h. The syntax for defining data of type EXPL should conform to that specified in Annex B.5.

6.2.4 Interpretation function

- a. It shall be possible to specify the interpretation function for a reporting data by means of an expression.
- b. Interpretation function expressions shall be able to operate on constants and space system parameters acquired from the EMCS.
- c. Mathematical, time and string functions shall be supported.
- d. Engineering units shall be supported.
- e. The capability to assign engineering units to constants shall be supported.

- f. The capability to mix compatible units freely within an expression shall be supported.
- g. The automatic conversion between different, but compatible, units shall be supported.
- h. The syntax for defining data of type IFL should conform to that specified in Annex B.6.

6.2.5 Procedure

- a. The requirements in ECSS-E-ST-70-32 clause 5 shall apply.

6.2.6 Synthetic parameter

- a. It shall be possible to synthesize reporting data within the EMCS from other reporting data and constants.
- b. It shall be possible to synthesize a parameter or one or more components of a compound parameter.
- c. The evaluation function shall return the engineering value of the synthetic parameter i.e. no subsequent interpretation (e.g. calibration) shall be required.
- d. It shall be possible to specify the condition(s) under which a new value of a synthetic parameter is evaluated (i.e. triggered).
- e. The possible conditions shall include:
 - 1. upon update of any reporting data (usually a “contributing” reporting data i.e. one whose value itself is used in the synthetic parameter evaluation).

NOTE Update, in this context, means the arrival of a new occurrence of the reporting data, not a change of value of the reporting data.

- 2. upon the occurrence of a specified event;
 - 3. any combination of the above.
- f. It shall be possible to define the context in which a synthetic parameter is evaluated i.e. a system element or reporting data context.
- g. The capability shall be provided to perform the following execution flow controls within a synthetic parameter evaluation:
 - 1. simple conditional branching (i.e. if ... then ... else ...);
 - 2. multiple conditional branching where the path taken is dependent on the value of a specified parameter (or local variable, see h below);
- h. Local variables used within a synthetic parameter evaluation shall be declared and their type explicitly stated.
- i. The capability shall be provided to assign a value to a local variable.
- j. Persistent variables shall be supported.

NOTE 1 A persistent variable is one whose value is maintained (memorized) from one evaluation of a synthetic parameter to the next.

NOTE 2 E.g.: a synthetic parameter that uses the delta between the current value and the previous value of a given parameter.

- k. It shall be possible to define an expression when assigning a value to a local variable or to a synthetic parameter.
- l. Synthetic parameter expressions shall be able to operate on constants, space system parameters acquired from the EMCS and local variables.
- m. Mathematical, time and string functions shall be supported.
- n. Engineering units shall be supported.
- o. The capability to assign engineering units to constants shall be supported.
- p. The capability to mix compatible units freely within an expression shall be supported
- q. The automatic conversion between different, but compatible, units shall be supported.
- r. The syntax for defining data of type SPEL should conform to that specified in Annex B.7.

6.2.7 Value set

- a. It shall be possible to specify fixed values or value sets for data items.

NOTE E.g.: default values.

- b. It shall be possible to specify a simple value, a record value or an array value for a single value or a component of a value set.
- c. The syntax for defining data of type VAL should conform to that specified in Annex B.9.

6.3 Product data

6.3.1 Introduction

Product data corresponds to the data that is provided, together with a given product delivery, by a supplier to a customer and comprises:

- Product configuration data;
- Product monitoring and control data (i.e. the SSM).

6.3.2 Product configuration data

6.3.2.1 Overview

According to ECSS-M-ST-40, product configuration data is provided to enable traceability from a product to its defining documentation.

Depending on the configuration management requirements associated with a given product, different types of product configuration data are provided.

6.3.2.2 Requirement

- a. For each product identified within the electronic database, the data specified in Table 6-2 shall be provided.

Table 6-2 Product configuration data

Name	Definition	Data Type
Name of Product	The name of the product which is unique within this delivery	Name
Nature	The nature of the product. The set of values is {" specification ", " realization "}. NOTE: " specification " corresponds to the specification of the product (i.e. a theoretical product); " realization " corresponds to a developed product (i.e. a real product).	Enumerated
Type	The type of product. The set of values includes {" developed hardware CI ", " developed software CI ", " COTS and standard product ", " non CI product "}. 	Enumerated

6.3.2.3

- a. For products of type "**developed hardware CI**", the additional data specified in Table 6-3 shall be provided.

Table 6-3 Developed hardware CI configuration data

Name	Definition	Data Type
ID	The Configuration Item identifier	Enumerated
Part Number	The part number of the product	Character String
Serial Number	The serial number of the product	Character String
Lot Number	The lot number of the product	Character String
Model	The model of the space system to which the product is applicable e.g. EM, STM, FM	Enumerated
ID of Manufacturer	The identifier of the manufacturer (i.e. the original supplier)	Enumerated

6.3.2.4

- a. For products of type “**developed software CI**”, the additional data specified in Table 6-4 shall be provided.

Table 6-4 Developed software CI configuration data

Name	Definition	Data Type
ID	The Configuration Item identifier	Enumerated
Software Identifier	The identifier of the software	Enumerated
Version Number	The version number of the software	Name
Release Date	The date of release of this version of the software	Absolute Time
ID of Manufacturer	The identifier of the manufacturer (i.e. the original supplier)	Enumerated

6.3.2.5

- a. For products of type “**COTS and standard product**”, the additional data specified in Table 6-5 shall be provided.

Table 6-5 COTS and standard product configuration data

Name	Definition	Data Type
Part Number	The part number of the product	Character String
Serial Number	The serial number of the product	Character String
Lot Number	The lot number of the product	Character String
ID of Manufacturer	The identifier of the manufacturer (i.e. the original supplier)	Enumerated

6.3.2.6

- a. For products of type “**non CI product**” (i.e. not defined as a configuration item), the additional data specified in Table 6-6 shall be provided.

Table 6-6 Non CI product configuration data

Name	Definition	Data Type
Part Number	The part number of the product	Character String
Serial Number	The serial number of the product	Character String
Lot Number	The lot number of the product	Character String
ID of Manufacturer	The identifier of the manufacturer (i.e. the original supplier)	Enumerated

6.3.2.7

- a. When one (or more) property of a system element has changed between successive deliveries of the same product or when one (or more) activities, reporting data, events or constituent system elements has been created or deleted, a new version shall be defined for this system element.
- b. When one (or more) property of an activity has changed between successive product deliveries, a new version shall be defined for this activity.
- c. When one (or more) property of a reporting data has changed between successive product deliveries or when one (or more) sub reporting data of a reporting data (i.e. of record type) has been created or deleted, a new version shall be defined for this reporting data.
- d. When one (or more) property of an event has changed between successive product deliveries, a new version shall be defined for this event.'

NOTE The SSM corresponds to one or more elements of the overall space system.

The product configuration data also contains configuration data for each constituent system element, activity, reporting data and event of the SSM according to the previous requirements.

6.3.2.8

- a. For each system element, activity, reporting data and event of the SSM, the data specified in Table 6-7 shall be provided.

Table 6-7 Version data

Data	Definition	Data Type
Number	The version number of the configuration data	Name
Date	The date when this configuration item version was produced	Absolute Time
Originator	The name of the organization or person responsible of having produced this configuration item version	Character String
Reason	The reason(s) for the change, with respect to the previous version	Character String
Status	Information relating to the validation testing status of this configuration item. The set of values includes {" not validated ", " validated by review ", " validated by simulation ", " validated by testing "}.	Enumerated

6.3.2.9

- a. The data specified in Table 6-2, Table 6-3, Table 6-4, Table 6-5 or Table 6-6 shall be provided for each constituent product, according to the product type.
- b. The data specified in Table 6-7 shall be provided for each object (i.e. system elements, reporting data, activities and events) of the SSM associated with each constituent product.

NOTE Products are the result of procurement and integration activities at lower-levels and earlier phases in the overall space system lifecycle.

The objective of these requirements is to allow the traceability of information supplied at these lower-levels (or earlier phases).

6.4 Data population

6.4.1

- a. All monitoring and control data shall be provided using the English language.

6.4.2

- a. All monitoring and control data shall be considered as case-insensitive, with the exception of engineering units.

NOTE Engineering units are defined in Annex B of ECSS-E-ST-70-32.

6.4.3

- a. If the name of an SSM object (i.e. system elements, reporting data, activities or events) is changed during integration of lower-level products, the former name(s) of this object shall be retained within the integrated product for traceability purposes.

6.4.4

- a. The data supplied with the delivered product shall be fully consistent with the product documentation (e.g. ICD, user manual).

6.4.5

- a. Uniqueness of SSM objects names within a given product delivery shall be ensured.

6.4.6

- a. Some elements of the space system are qualified for a maximum number of operational cycles or a maximum operational life. For these elements, the accumulated number of operational cycles or the accumulated operational life shall be provided with each product delivery.

6.4.7

- a. Sufficient information shall be provided by the sub-suppliers such that the data required by the customer is fully populated.

6.4.8

- a. Data specific to a domain (e.g. Software, AIV or Operations) shall be identified as such.

NOTE The principal customers of the monitoring and control data are “Software” (i.e. on-board and ground software development teams), “AIV” and “Operations”.

- b. Data required to monitor and control the space system in all operational and testing modes shall be fully populated.
- c. For the SSM, fully populated shall mean:
 - 1. All system elements that correspond to the delivered product be provided.
 - 2. All monitoring and control characteristics of the system elements are provided. In the case of ECSS-E-ST-70-41 services (and other on-board software), this includes all service configuration data required on ground for monitoring and control of the service.
- d. For reporting data, fully populated means:
 - 1. All reporting data defined within the product documentation shall be provided. This includes all data that can be downlinked, for example, software tables and memory dumps.
 - 2. When more than one reporting data conveys the same information, the redundancy link between them shall be provided.
 - 3. All interpretation functions (required to calibrate raw values) shall be provided together with the conditions under which they are applied.
 - 4. When the validity of a reporting data depends on specific conditions, the validity conditions shall be provided.
 - 5. Ground monitoring checks shall be provided for all reporting data together with the conditions under which they are applied.

NOTE The only exceptions are reporting data that can take any possible value.

- 6. When the value that a reporting data may take depends on the execution of a given activity or the occurrence of a given event, status consistency checks shall be provided accordingly.
 - 7. The corrective actions to be taken whenever a reporting data violates a given check shall be provided.
- e. For activities, fully populated means:
 - 1. All activities defined within the product documentation (e.g. operation procedures contained in the space segment user manual) shall be provided.
 - 2. All atomic activities used to control elements of the space system shall be provided.
 - 3. Arguments of activities shall be provided in engineering values together with (where applicable) the deinterpretation function used to produce the raw values.
 - 4. Pre-conditions and post-conditions for activities shall be provided.