SLOVENSKI STANDARD
SIST EN 16603-40-07:2020

01-september-2020

Vesoljska tehnika - Ploščadi za simulacijsko modeliranje

Space engineering - Simulation modelling platform

Raumfahrttechnik - Software-Modellierungs-Platform

Ingénierie Spatiale - Plateforme informatique de modèles de simulation

Ta slovenski standard je istoveten z: **EN 16603-40-07:2020**

ICS:

49.140	Vesoljski sistemi in operacije	Space systems and operations

SIST EN 16603-40-07:2020 **en,fr,de**

EUROPEAN STANDARD

NORME EUROPÉENNE

EUROPÄISCHE NORM

EN 16603-40-07

June 2020

ICS 49.140

English version

Space engineering - Simulation modelling platform

Ingénierie spatiale - Plateforme de modélisation pour simulation

Raumfahrttechnik - Teil 40-07: Modelliersoftware-Platform

This European Standard was approved by CEN on 17 May 2020.

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, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Republic of North Macedonia, Romania, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.

CEN-CENELEC Management Centre:
Rue de la Science 23, B-1040 Brussels

Table of contents

Figures

Tables

European foreword

This document (EN 16603-40-07:2020) has been prepared by Technical Committee CEN-CENELEC/TC 5 "Space", the secretariat of which is held by DIN.

This standard (EN 16603-40-07:2020) originates from ECSS-E-ST-40-07C.

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 December 2020, and conflicting national standards shall be withdrawn at the latest by December 2020.

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 standardization request 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, Serbia, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

Introduction

Space programmes have developed simulation software for a number of years, which are used for a variety of applications including analysis, engineering operations preparation and training. Typically, different departments perform developments of these simulators, running on several different platforms and using different computer languages. A variety of subcontractors are involved in these projects and as a result a wide range of simulation software are often developed. This standard addresses the issues related to portability and reuse of simulation models. It is based on the work performed by ESA in the development of the Simulator Model Portability Standards SMP1 and SMP2 starting from the mid-end of the nineties.

This standard integrates the ECSS-E-ST-40 with additional requirements which are specific to the development of simulation software. The formulation of this standard takes into account:

- The existing ISO 9000 family of documents, and
- The Simulation Model Portability specification version 1.2.

The intended readership of this standard is the simulator software customer and supplier.

1
Scope

ECSS-E-ST-40-07 is a standard based on ECSS-E-ST-40 for the engineering of simulation software.

ECSS-E-ST-40-07 complements ECSS-E-ST-40 in being more specific to simulation software. Simulation software include both Simulation environments and simulation models. The standard enables the effective reuse of simulation models within and between space projects and their stakeholders. In particular, the standard supports model reuse across different simulation environments and exchange between different organizations and missions.

This standard can be used as an additional standard to ECSS-E-ST-40 providing the additional requirements which are specific to simulation software.

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

Applicability

This standard lays down requirements for simulation software including both Simulation environments and simulation models. The requirements cover simulation models' interfaces and simulation environment interfaces for the purpose of model re-use and exchange to allow simulation models to be run in any conformant simulation environment.

A consequence of being compliant to this standard for a model is the *possibility* of being reused in several simulation facilities or even in several projects. However, adherence to this standard does not imply or guarantees model reusability, it is only a precondition. Other characteristics of the model, to be defined outside this standard, such as its functional interfaces and behaviour, its configuration data as well as quality, suitability and performance, etc. are also heavily affecting the potential for a model to be reused. In addition, agreements need to be reached on simulation environments compatibility, model validation status as well as legal issues and export control restrictions.

Therefore, this standard *enables* but does not mandate, impose nor guarantee successful model re-use and exchange.

Model reuse in this standard is meant both at source-code and binary level, with the latter restricted to a fixed platform.

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 16601-00-01	ECSS-S-ST-00-01	ECSS system - Glossary of terms
EN 16603-40	ECSS-E-ST-40	Space engineering - Software general requirements
	[SMP_FILES]	ECSS_SMP_Issue1(2March2020).zip – SMP C++ Header files, SMP XML schemas and SMP Catalogue. (Available from ECSS website)
	https://www.w3.org/TR/xmlschema11-2/	XML schema specification
	http://www.opengroup.org	The UUID specification from Open Group.
	https://www.osgi.org/developer/specifications/	OSGi Specifications

3
Terms, definitions and abbreviated terms

3.1 Terms from other standards

a. For the purpose of this Standard, the terms and definitions from ECSS-S-ST-00-01 and ECSS-E-ST-40 apply.

b. For the purpose of this Standard, the terms and definitions from ECSS-E-ST-70 apply, in particular the following term:

1. mission

3.2 Terms specific to the present standard

In the following list of terms, underlined words are further defined in the same list.

3.2.1 aggregate

relationship between two components implemented by storing their references

NOTE Each component in such a relationship keeps its own lifecycle and it does not dependent on that of other components.

3.2.2 association

relationship between two instances of any data-type, where each instance has its own lifecycle and there is no owner

3.2.3 breakpoint

unambiguous state of a simulation

3.2.4 component

building block of a simulation that can be instantiated and that has a well-defined contract to its environment

3.2.5 composite

component implementing composition

3.2.6 composition

hierarchical relationship where child component is destroyed if the parent component is destroyed

3.2.7 configuration

specification of values for fields of components

3.2.8 constructor

specific operation of a component, bearing the same name of the component, whose purpose is to allocate and build an instance of said component

3.2.9 consumer

component that can receive data in one of its input fields from an output field of another component

3.2.10 container

typed collection of child components

3.2.11 contract

set of interfaces, operations, fields, entry points, event sinks, event sources and all the associated constraints, used to interact with a component

3.2.12 data transfer

copy of value from an output field to an input field

3.2.13 entry point

operation without parameters that does not return a value, which can be added to the scheduler or event manager service

3.2.14 epoch time

absolute time of the simulation

3.2.15 event

see "simulation event"

3.2.16 event manager

component that implements the IEventManager interface

NOTE The IEventManager interface is specified in clause 5.3.4.

3.2.17 event sink

receiver of specific notifications, owned by a component and subscribed via a subscription mechanism

3.2.18 event source

emitter of specific notifications, owned by a component and offering a subscription mechanism

3.2.19 exception

non-recoverable error that can occur when calling into an operation or property

3.2.20 field

feature characterised by a value type and holding a value

3.2.21 input field

field explicitly marked for receiving values as a result of a data transfer

3.2.22 interface

named set of properties and operations

3.2.23 logger

component that implements the ILogger interface

NOTE The ILogger interface is specified in clause 5.3.1.

3.2.24 mission time

relative time measuring elapsed time from a mission specific point in time

3.2.25 model

component that implements the IModel interface

NOTE The IModel interface is specified in clause 5.2.3.2.

3.2.26 model implementation

executable code implementing a model

3.2.27 model instance

occurrence of a model implementation

3.2.28 output field

field explicitly marked for being the source of a value in a data transfer

3.2.29 operation

declaration of a behavioural feature of a component or an interface with the option to define parameters, return value and raised exceptions

3.2.30 package

collection of types, where each one is either a value type or a component

3.2.31 platform

set of subsystems/technologies that provide a coherent set of functionality through APIs and specified usage patterns

3.2.32 primitive type

type that can no longer be de-composed and that is pre-defined by the standard

NOTE The available primitive types are listed in Table 5-1: Primitive Types.