

|  |
| --- |
| SOIS XML EDS PROTOTYPING TEST PlAN & REPORT |

DRAFT CCSDS Record

CCSDS 876.1-Y-1 DRAFT #1

Draft Yellow Book

October 2018

FOREWORD

This document represents a test plan and test report for the prototype testing conducted in support of advancing the SOIS EDS XML Specification to CCSDS Blue Book status.

Through the process of normal evolution, it is expected that expansion, deletion, or modification of this document may occur. This document is therefore subject to CCSDS document management and change control procedures, which are defined in the *Procedures Manual for the Consultative Committee for Space Data Systems*. 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 addressed to the CCSDS Secretariat at the 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.
* British National Space Centre (BNSC)/United Kingdom.
* 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 e.V. (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.

Observer Agencies

* Austrian Space Agency (ASA)/Austria.
* Belgian Federal Science Policy Office (BFSPO)/Belgium.
* Central Research Institute of Machine Building (TsNIIMash)/Russian Federation.
* Centro Tecnico Aeroespacial (CTA)/Brazil.
* 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.
* European Organization for the Exploitation of Meteorological Satellites (EUMETSAT)/Europe.
* European Telecommunications Satellite Organization (EUTELSAT)/Europe.
* Hellenic National Space Committee (HNSC)/Greece.
* Indian Space Research Organization (ISRO)/India.
* Institute of Space Research (IKI)/Russian Federation.
* KFKI Research Institute for Particle & Nuclear Physics (KFKI)/Hungary.
* Korea Aerospace Research Institute (KARI)/Korea.
* MIKOMTEK: CSIR (CSIR)/Republic of South Africa.
* Ministry of Communications (MOC)/Israel.
* National Institute of Information and Communications Technology (NICT)/Japan.
* National Oceanic and Atmospheric Administration (NOAA)/USA.
* National Space Organization (NSPO)/Chinese Taipei.
* Naval Center for Space Technology (NCST)/USA.
* Space and Upper Atmosphere Research Commission (SUPARCO)/Pakistan.
* Swedish Space Corporation (SSC)/Sweden.
* United States Geological Survey (USGS)/USA.

DOCUMENT CONTROL

|  |  |  |  |
| --- | --- | --- | --- |
| **Document** | **Title and Issue** | **Date** | **Status** |
| CCSDS 876.1-Y-1 | SOIS XML EDS Prototyping Test Plan and Report, Draft CCSDS Record, Issue 1, Draft #1 | October 2018 | Initially distributed draft |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

CONTENTS

Section Page

[DOCUMENT CONTROL iii](#_Toc532394933)

[CONTENTS iv](#_Toc532394934)

[1 Introduction 1-1](#_Toc532394935)

[1.1 PURPOSE 1-1](#_Toc532394936)

[1.2 SCOPE 1-1](#_Toc532394937)

[1.3 APPLICABILITY 1-1](#_Toc532394938)

[1.4 RATIONALE 1-1](#_Toc532394939)

[1.5 Background 1-1](#_Toc532394940)

[1.6 DOCUMENT STRUCTURE 1-3](#_Toc532394941)

[1.7 References 1-3](#_Toc532394942)

[1.8 ACRONYMS 1-3](#_Toc532394943)

[2 SUMMARY CONCLUSION/RECOMMENDATION 2-5](#_Toc532394944)

[3 SEDS/XML PROTOTYPE TEST PHILOSOPHY 3-1](#_Toc532394945)

[3.1 BASIC TEST PLAN 3-1](#_Toc532394946)

[3.2 TEST REPORTS 3-2](#_Toc532394947)

[4 TEST PLAN DETAILS 4-4](#_Toc532394948)

[4.1 Test 0 4-4](#_Toc532394949)

[4.1.1 Background 4-4](#_Toc532394950)

[4.1.2 Methodology and naming conventions 4-4](#_Toc532394951)

[4.1.3 NASA TEST SCRIPT AND RESULTS 4-4](#_Toc532394952)

[4.1.4 ESA TEST RESULTS 4-5](#_Toc532394953)

[4.2 Test 1 4-5](#_Toc532394954)

[4.2.1 Background 4-5](#_Toc532394955)

[4.2.2 Methodology and naming conventions 4-5](#_Toc532394956)

[4.2.3 NASA TEST SCRIPT AND RESULTS 4-6](#_Toc532394957)

[4.2.4 ESA TEST RESULTS 4-6](#_Toc532394958)

[4.3 TEST 2 4-6](#_Toc532394959)

[4.3.1 Background 4-6](#_Toc532394960)

[4.3.2 Methodology and naming conventions 4-7](#_Toc532394961)

[4.3.3 NASA cFS TEST SCRIPT AND RESULTS 4-7](#_Toc532394962)

[4.3.4 ESA TEST RESULTS 4-7](#_Toc532394963)

[4.4 TEST 3 4-7](#_Toc532394964)

[4.4.1 Background 4-7](#_Toc532394965)

[4.4.2 Methodology and naming conventions 4-8](#_Toc532394966)

[4.4.3 NASA CFS TEST SCRIPT AND RESULTS 4-8](#_Toc532394967)

[4.4.4 ESA TEST RESULTS 4-8](#_Toc532394968)

[4.5 TEST 4 4-8](#_Toc532394969)

[4.5.1 Background 4-8](#_Toc532394970)

[4.5.2 Methodology and naming conventions 4-9](#_Toc532394971)

[4.5.3 NASA TEST SCRIPT AND RESULTS 4-9](#_Toc532394972)

[4.5.4 ESA TEST SCRIPT AND RESULTS 4-9](#_Toc532394973)

[5 NASA PROTOTYPE TEST report 5-10](#_Toc532394974)

[5.1 TESt 0 NASA seds validation TEST results 5-10](#_Toc532394975)

[5.2 TESt 1 NASA seds validation TEST results 5-11](#_Toc532394976)

[5.3 TESt 2 NASA seds validation TEST results 5-12](#_Toc532394977)

[5.4 TESt 3 NASA seds validation TEST results 5-15](#_Toc532394978)

[5.5 TESt 4 NASA seds validation TEST results 5-16](#_Toc532394979)

[6 ESA PROTOTYPE TEST report 6-18](#_Toc532394980)

[6.1 TESt 0 ESA seds validation TEST results 6-18](#_Toc532394981)

[6.2 TESt 1 ESA seds validation TEST results 6-18](#_Toc532394982)

[6.3 TESt 2 ESA seds validation TEST results 6-18](#_Toc532394983)

[6.4 TESt 3 ESA seds validation TEST results 6-19](#_Toc532394984)

[6.5 TESt 4 ESA seds validation TEST results 6-20](#_Toc532394985)

# Introduction

## PURPOSE

This document presents the test plan and test report for interoperability tests conducted to support of advancement of the red CCSDS book titled*, Spacecraft Onboard Interface Service XML Specification for Electronic Data Sheets, CCSDS 876.0-R-3 (Hereafter “SEDS”),* to blue book status. This report will be submitted to the CCSDS Secretariat, CCSDS Engineering Steering Group (CESG) and CCSDS Management Council (CMC), along with results of the Agency Reviews of the SEDS document. Subsequently a formal request will be submitted to the CESG/CMC for progression of the SEDS book to CCSDS Blue Book status.

## SCOPE

The document scope is the interoperability test plan and test results to show interoperability for the *SEDS.* The interoperability test plan is based upon two different agencies’ prototype tests from known valid SEDS, and a report of the test results that show correct interpretations from the two different implementations. Before any SEDS is used in an interoperability test, it is validated against the core schema. Validation is also part of the interoperability testing (Test 0).

## APPLICABILITY

This document applies to the prototype tests conducted for the final edition of the SEDS, *CCSDS 876.0-R-3,* October 2018*.*

## RATIONALE

This test plan/report is required by the *Procedures Manual for the Consultative Committee for Space Data Systems* [1], which states that “at least two independent and interoperable prototypes or implementations must have been developed and demonstrated in an operationally relevant environment, either real or simulated.” The document outlines the SOIS Application Support Working Group’s (APP WG) approach to meet this requirement for the SEDS.

## Background

The SEDSis part of the technical program of the CCSDS SOIS APP WG. It is a XML schema that has been proposed to allow the capture of the relevant information about software components and device interfaces. This should capture the relevant aspects not just to enable an efficient exchange of information (easing maintainability, enforcing consistency, etc.), but also enabling the development process of related software to be supported by the use of model-based software engineering techniques, Figure 1.

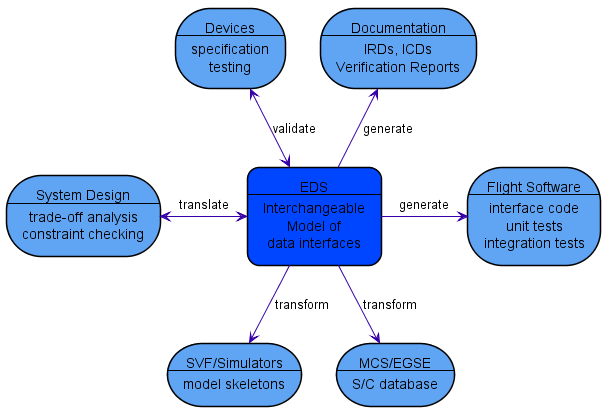


Figure 1 SOIS EDS Concept

In the course of the mission lifecycle, different parts of the overall system (which includes both space and ground) will need to represent or interact with an onboard device, including:

* Tools used for the design and validation of the device itself.
* System design and analysis tools modelling a system using the device, for example to check bus bandwidth and schedulability.
* Tools used for the design and implementation of the Flight Software (FSW) which executes on an onboard computer and is in charge of communication with, and autonomous operation of, the device.
* Mission Control and Electrical Ground Support Equipment (EGSE) systems, in cases where the device contributes to the definition of some portion of the spacecraft Telemetry (TM) and/or the telecommand (TC) definition.
* Software Validation Facilities and Operational Simulators that model device behavior in order to validate the interaction of the FSW and the device.
* Tools that generate portions of the system documentation.

This wide range of usages means that no one tool could plausibly meet them all; hence the need for an intermediate, platform independent format. Consequently, the **SEDS** takes the form of an XML schema designed for tool interchange, i.e. exchanging device data between two software systems. The interoperability tests in this report use tools for design and validation of the device itself as well as tools that generate portions of the system document. As a key attribute of the SEDS is extensibility, a test, to add a new unit of measure is also included (Test 1). The extension is defined in the *seds-extension-sematics.xsd*.

## DOCUMENT STRUCTURE

Section 1 (this section) presents introductory material.

Section 2 presents a summary of the prototype testing and the recommendation of the SOIS APP WG based on that testing.

Section 3 presents the overall test philosophy for the SEDS prototyping.

Section 4 presents the test plan the SEDS prototyping.

Section 5 presents the test results from NASA/GSFC prototype testing.

Section 6 presents the test results from ESA/ESTEC prototype testing.

## References

The following documents are referenced in this document. At the time of publication, the editions indicated were valid. The CCSDS Secretariat maintains a register of currently valid CCSDS documents.

[1] *Procedures Manual for the Consultative Committee for Space Data Systems*. CCSDS A00.0-Y-9. Yellow Book. Issue 9. Washington, D.C.: CCSDS, November 2003.

[2] *Spacecraft Onboard Interface Services - XML Specification For Electronic Data Sheets*, CCSDS 876.0-R-3. Red Book. Issue 0. Washington, D.C.: CCSDS, August 2018.

[3] *Spacecraft Onboard Interface Services* - Specification for Dictionary of Terms for Electronic Data Sheets, CCSDS 876.1-R-0, March 2015

## ACRONYMS

APP Application Support (working group)

ASF Advanced Systems Format

CCSDS Consultative Committee for Space Data Systems

CESG CCSDS Engineering Steering Group

cFS core Flight System (NASA developed open source software framework)

CMC CCSDS Management Council

CWE Common Working Environment

ECSS European Cooperation for Space Standards

EDS Electronics Data Sheet

EGSE Electrical Ground Support Equipment

ESA European Space Agency

ESTEC European Space Agency Technology Center

FSW Flight Software

GSFC Goddard Space Flight Center

ICS Implementation Conformance Statement

MAS Memory Access Service

NASA National Aeronautics and Space Administration

PDU Protocol Data Unit

PS Packet Service

RL Requirements List

SAP Service Access Point

SciSys a pan-European computer software and services company

SEDS Spacecraft Onboard Interface Services Electronics Data Sheet

SOIS Spacecraft Onboard Interface Services

TC Telecommands

TM Telemetry

WG Working Group

XInclude Generic mechanism for merging [XML](https://en.wikipedia.org/wiki/XML) documents

XML Extensible Markup Language

# SUMMARY CONCLUSION/RECOMMENDATION

Five tests were used to demonstrate the interoperability of a SEDS description for two different software frameworks’, i.e., demonstrate the ability of two different software frameworks’ to validate and accurately interpret common SEDS descriptions’ with the same results. These software frameworks’ include one prototype from NASA (cFS) and one prototype from ESA (SciSys). Additionally the SEDS schema extensibility for custom semantics as well as complete coverage for all service features defined in the SEDS red book [2] Annex A (ICS), are demonstrated between the two different software frameworks.

The first test, Test 0, validates the SEDS test files’ against the core SEDS schema, *seds.xsd*. Two different validation tools are used for this test to show interoperability.

The second test, Test 1, demonstrates the ability to extend the schema for custom semantics [3]. Two different validation tools are used for this test to show interoperability.

The third test, Test 2, uses a real spacecraft device’s SEDS description, Jena Star Tracker, to interpret recorded data from the actual device.

The fourth interoperability test, Test 3, uses a synthetic device with synthetic data to test those SEDS schema features that are not being tested by the real device in Test 2.

The fifth test, Test 4, demonstrates the ability of two different XML rendering tools to interpret the SEDS behavioral descriptions (state-machines and activities) with the same results. Test 4 uses visual inspection, instead of automated software tools to perform comparisons of data, whereas Test 0 thru Test 3 use a combination of visual inspection and automated software tools to perform comparison of data.

All tests show that the two different prototype implementations have the same results for the input data as interpreted by the SEDS, hence interoperability is demonstrated.

Based upon the successful test results of the SEDS interoperability tests, it is recommended to promote *Spacecraft Onboard Interface Services - XML Specification For Electronic Data Sheets*, CCSDS 876.0-R-3, red book to blue book status.

# SEDS/XML PROTOTYPE TEST PHILOSOPHY

## BASIC TEST PLAN

The SEDS prototype testing will validate SEDS data sheets’ against the base schema, *seds.xsd*, and exercise each of the items in the Requirements List (RL) of the Implementation Conformance Statement (ICS), documented in *CCSDS 876.0-R-3*, Annex A [2]. Some of these ICE service features are necessary to unambiguously describe data formats (telemetry and commands) and some features are used to specify behavioral information (state-machines) and executable statements (activities).

The test plan for Test 0 is to demonstrate that two independent tools can validate a group of SEDS against the SEDS schema, *seds.xsd*. A group of 32 SEDS was generated to validate every element, child element and enumerated value possible for the SEDS (*seds.xsd)*. This test uses a blank *seds-extension-semantics.xsd* file so that many of the 32 SEDS files fail to validate as expected.

The test plan for Test 1 is to demonstrate that two independent tools can validate a group of 32 SEDS including the addition of a new semantic terms against an extended SEDS schema *per reference [3]*. The extension of the semantics is performed by adding a new term and new enumerated terms for existing types to the *seds-extension-semantics.xsd* file. In this case all test validate because for Test 1 unlike Test 0 the *seds-extension-semantics.xsd* defines the new term and the new enumerated values of existing types.

The test plan for Test 2 demonstrates that two independent software frameworks can use a common device, Jena Star Tracker in this case, and both interpret the data correctly. The input data for this test is generated by using recorded data from a Jena Star Tracker. A Jena Star Tracker’s data bus traffic is captured and post-processed according to the SEDS. This information is captured by a reference to SOIS services’ at the SUBNET Protocol Convergence level, in this case for the Packet Service (PS) and Memory Access Service (MAS), which specify how to map Milbus data bus Protocol Data Units (PDUs) to packets at the SOIS Service Access Point (SAP) interfaces. The underlying SEDS SUBNET Protocol Convergence level specification referenced in the SEDS for this particular case is the European Cooperation for Space Standardization (ECSS) document, *ECSS-E-50-13C*, which specifies how to generate packets from Milbus PDUs. A single software tool is used in this case to map the data bus PDUs to packets per the *ECSS-E-50-13C* specification. The resulting packets from this tool are then decoded using the Jena Star Tracker SEDS and used at the SOIS PS and MAS SAP interface. This decoded data is then re-encoded and checked for correctness against the original packet raw hex file (Bin file) that was used to decode the data. The self-check is done by comparing the two files (original and re-encoded) to make sure they are identical. This is done by both prototype implementations used for the interoperability test. In addition to this check, both implementations decoded data from the original packet raw hex file is then cross-check by visual comparison to ensure correctness. This is done manually because there may be slight differences in precision between the implementations, such as in floating point representations, but the values may still be correct.

The test plan for Test 3, is similar to Test 2, except a synthetic device is created to complete the coverage of SEDS schema data type definitions that was not used by the Jena Star Tracker SEDS. Obviously in this case the input data is created because there is no real device. The value that Test 2 has over Test 3 is that the device and data are real and not synthetic. Otherwise the test flows between Test 2 and Test 3 are identical. The synthetic input data is decoded using the *ccsds\_sois\_sampletypes.xml* SEDS file, re-encoded and compared to the original synthetic input data to make sure they are identical. In addition to this check, both implementations decoded data from the original packet raw hex file are cross-check by visual comparison to ensure correctness. This is done manually because there may be slight differences in precision between the implementations, such as floating point representations but the values may still be correct.

Test 4 verifies the behavioral aspects of a device that are a necessary part of a device specification to use the device. How this information is used by an implementer may vary. Some implementers may auto-generate software code from the behavioral description and others may use the human readable output and manually code the behavior. Since there are not two implementations that auto-generate code from the behavioral description, the method selected to show interoperability is to show that the interpretation from two different rendering tools is the same. Two different prototypes, one by ESA (SciSys) and one by NASA, are used to generate state-machine and activity diagrams. The basic flow is as follows: a SEDS is transformed to a file format that the respective rendering tool can interpret, and then rendered into state-machine and activity diagrams. These output diagrams are then visually compared to assess correctness between the two different prototypes.

All the tests together verify all the elements, child elements and enumerated values in the SEDS schema, *seds.xsd*. This also covers (by default) all the service features in the RL of the SEDS ICS (Annex A) [2].

This is the basic test plan for the SEDS interoperability tests.

A note about validation: before a SEDS may be validated by commonly available XML tools, it may require pre-processing to fill in mission specific profile information or references to other data sheets. An example is the CCSDS Space Packet’s Application ID that cannot be known by the device for its use in a unknown system. These placeholder fields are denoted by ${define}in the SEDS. Although package files may also be complete, these incomplete SEDS are always referred to as package files and must be run through a pre-processor to fill in the actual values and/or references before validation.

## TEST REPORTS

The test report will consist of:

1. Input files for the test.
2. Output files from the test (results)
3. Comparison of files necessary to show the results of the tests (if applicable).
4. Comments regarding test results.

# TEST PLAN DETAILS

The five different tests, Test 0 through Test 4, are detailed in the following sections. The files used for the tests and the results are all stored on the SOIS CWE working group site, [*releases/ccsds\_sois\_seds\_interoperability*](https://github.com/nasa/SOIS-CCSDS-XML)*.zip*. For permission to this site, please contact the, SOIS Area Director.

## Test 0

The following SOIS CWE directory contains artifacts from the interoperability testing between the available SEDS implementations from NASA cFS and ESA for Test 0: *releases/SOIS-CCSDS-XML/seds/interoperability/test\_0.*

### Background

The data for Test 0 was synthetically generated to capture every possible SEDS schema element, child element along with any possible enumerated values along with a term and some enumerated types not originally defined. This data file was created to validate a group of 32 SEDS against the *seds.xsd* and its included extensions. The SEDS that are validated contain extended values that are not defined in the Test 0 version of the *seds-extension-semantics.xsd*. Therefore there are expected validation errors for those terms not included in the *seds-extension-semantics.xsd*.

For this testing exercise, the following was considered:

1. The interoperability test is designed to show the independent validation of the SEDS schema.
2. Intentional errors are introduced into the test by using a *seds-extension-semantics.xsd* that is blank and therefore it is missing defined terms in the SEDS being validated.
3. All SEDS elements, child elements and enumerated values are validated.

### 4.1.2 Methodology and naming conventions

For this test the 32 SEDS files’ are being validated against the *seds.xsd* and it’s referenced XML files. The package files that are included into the *seds.xsd*, include the core semantics (*seds-core-semantics.xsd*), and the extended semantics *(seds-extension-semantics.xsd*). These files may be found at ..test\_0/input/schema. The files being validated may be found at ..test\_0/input/eds. There is also a script executable to run the test for each prototype and a resulting output log file. These files may be found in the locations defined in the following sections.

### 4.1.3 NASA TEST SCRIPT AND RESULTS

The ../*test\_0/nasa/scripts* directory contains a script, validate\_xml.sh, that validates the 32 XML files that represent all the elements, child elements and enumerated types defined in the SEDS red book against the *seds.xsd* and its package files.

The ../*test\_0//nasa/logs* directory contains a log file, validate\_xmlint.txt, of the results of the validation test. For this test there are many XML files that fail to validate as expected because the *seds-extension-semantics.xsd* is blank.

### 4.1.4 ESA TEST RESULTS

The ../*test\_0/esa/logs* directory contains a log file of the results of the validation test. For this test there are many XML files that fail to validate as expected because the *seds-extension-semantics.xsd* is blank.

## 4.2 Test 1

The following SOIS CWE directory contains artifacts from the interoperability testing between the available EDS implementations from NASA cFS and ESA for Test 1: *releases/SOIS-CCSDS-XML/seds/interoperability/test\_1.*

### 4.2.1 Background

The data for Test 1 is the same file as Test 0 except a new term and the new enumerated types for existing types was included in the *seds-extension-semantics.xsd* per reference [3] that is referenced through the Xinclude syntax in the *seds.xsd*. The change was made to add a new unit of measure for Warp Factor, the Cochrane, and extended enumerated values to the existing schema to demonstrate the ability to easily extend the semantics of the SEDS. The change was created to demonstrate the process of updating the semantics. The new *seds-extension-semantics.xsd* file is located in *releases/SOIS-CCSDS-XML/seds/interoperability/test\_1/input/schema/*. The other files for Test 1 may be found at releases/../test\_1/ under the appropriate directory just as for Test 0. The only other difference is in the output file in the log directory. See below sections for details.

For this testing exercise, the following was considered:

1. The interoperability test is designed to show the ability to extend the semantics of the SEDS schema.
2. All SEDS elements, child elements and enumerated values are validated, including a new term and new enumerated values for existing types.

### 4.2.2 Methodology and naming conventions

For this test a group of 32 SEDS files are being validated again (from Test 0) with changes to the *seds-extension-semantics.xsd* file for the new unit type and new enumerated values of existing defined types. There are several package files that are included into the *seds.xsd*, one for the core semantics, *seds-core-semantics.xsd*, and one for the extended semantics, *seds-extension-semantics.xsd*. There is also a script executable to run the test.

### 4.2.3 NASA TEST SCRIPT AND RESULTS

The ../*test\_1/nasa/scripts* directory contains a script, *validate\_xml.sh*, that validates the 32 files that represent all the elements, child elements and enumerated types defined in the SEDS red book against the *seds.xsd* and its package files.

The ../*test\_1/nasa/logs* directory contains a log file, *validate\_xmlint.txt*, of the results of the validation test. In this case unlike the Test 0 case, all tests validate.

### 4.2.4 ESA TEST RESULTS

The ../***test\_1/esa/log*** directory contains a log file of the results of the validation test. In this case unlike the Test 0 case, all tests validate.

## 4.3 TEST 2

The following SOIS CWE directory contains artifacts from the interoperability testing between the available EDS implementations from NASA cFS and ESA for Test 2: *releases/SOIS-CCSDS-XML/seds/interoperability/test\_2.*

### 4.3.1 Background

The original data was captured from an actual Jena Star Tracker device by using a “sniffer” on the Milbus between the device and the host machine.

This "raw" capture data set from the sniffer contains ECSS-encoded link layer data which is then further encoded in an ASF file for storage by the capture tool. The original raw (ASF+ECSS) capture data file is available in *../seds/interoperability/test\_2/input/data/*. These files are named based upon their packet sequence in the recorded data from the “sniffed” device and only a small subset was used. This directory contains six files that represent all recorded Jena Star Tracker data types and two examples of each type. This test covers elements in SEDS document sections 3.3 through 3.13 but not all elements described in these sections. The next test, Test 3, is designed to have complete coverage for SEDS document sections 3.3 through 3.13.

For this testing exercise, the following was considered:

1. The interoperability test should be performed starting at the Packet Service layer. Therefore, the additional storage and link layer encapsulation layers (ASF+ECSS) are outside the scope of SOIS EDS interoperability testing.
2. Many of the recorded frames are quite similar. In order to reduce the test size to a more practical level, a small subsample is taken containing at least one of each unique message type captured and two examples of each type.

### 4.3.2 Methodology and naming conventions

The reconstituted transfer frames are stored in releases/../test\_2/input/data  subdirectory. This reflects the chosen sub-sample which was then pre-processed in an external tool to decode the ASF framing added by the sniffing tool as well as the ECSS link layer framing from the original bus data transfer cycles. Each logical message was then saved in a separate file with a .bin extension along with a 4-digit decimal serial number reflecting the sequence from the original capture. For example, the file:

***capturedata/jena-0345.bin***

contains the 345th logical binary message from the raw capture data.

Four distinct message types were observed in the capture data. Two samples of each message were randomly selected for a total of eight messages for this test exercise. The packets from the captured data are processed (decoded) using the Jena Star Tracker SEDS. The decoded form is then re-encoded and autonomously compared. In addition, the log files of the decoded data is visually compared with the ESA decoded data from the same input files to verify correct operation by a second method.

### 4.3.3 NASA cFS TEST SCRIPT AND RESULTS

The ../*test\_2/nasa/scripts* directory contains a script that works with the NASA cFS EDS implementation to decode the contents of the binary files.

The ../*test\_2/nasa/logs* directory contains a log file of the test execution showing the complete name=value contents of the decoded messages.

### 4.3.4 ESA TEST RESULTS

The ../***test\_2/esa/logs*** directory contains a log file of the test execution showing the complete name=value contents of the decoded messages.

## 4.4 TEST 3

The following SOIS CWE directory contains artifacts from the interoperability testing between the available SEDS implementations from NASA cFS and ESA for Test 3: *releases/SOIS-CCSDS-XML/seds/interoperability/test\_3.*

### 4.4.1 Background

The original data was created to fully cover defined elements in SEDS document sections 3.3 through 3.13.

For this testing exercise, the following was considered:

1. The interoperability test should be performed at the Application layer. Therefore, the SUBNET layer in this particular test case is outside the scope of SOIS EDS interoperability testing.
2. The 15 hex files each represent a type of data defined in the SEDS.

### 4.4.2 Methodology and naming conventions

Each type of data being tested is named with the type of data being tested in the file name, e.g., TESTDATA\_ALL\_BOOLEANS.bin., TEST\_DATA\_ALL\_FLOATS.bin, etc. There are 15 files in all, and are located in the *../seds/interoperability/test\_3/input/data/*. The data being tested with the sample SEDS, which is located at *../seds/interoperability/test\_3/input/eds/*, is first processed to obtain the equivalent format as Test 2, raw form. This data is then decoded using the sample SEDS to obtain the decoded data. The decoded data is then re-encoded and autonomously compared with the raw data. In addition, the log files of the decoded data is visually compared with the ESA decoded data from the same input files to verify correct operation by a second method.

### 4.4.3 NASA CFS TEST SCRIPT AND RESULTS

The *test\_3/nasa/scripts* directory contains a script that works with the NASA cFS SEDS implementation to decode the contents of the binary files.

The *test\_3/nasa/logs* directory contains a log file of the test execution showing the complete name=value contents of the decoded messages.

### 4.4.4 ESA TEST RESULTS

The ***test\_3/esa/logs*** directory contains a log file of the test execution showing the complete name=value contents of the decoded messages.

## 4.5 TEST 4

The following SOIS CWE directory contains artifacts from the interoperability testing between the available SEDS implementations from NASA cFS and ESA for Test 4: *releases/SOIS-CCSDS-XML/seds/interoperability/test\_4/.*

### 4.5.1 Background

This data is used for the visualization of activity and state-machine diagrams, Test 4. It is a combination of SEDS from a real device (Jena Star Tracker), and synthetic data to complete coverage of the activity and state-machine portion of elements of the SEDS. Test 4 tests are meant to test SEDS document sections 3.15 (Activities), 3.16 (State-Machines), 3.14 (component implementation – specifies behavior), 4.5 (Primitive Associations), 4.6 (State-Machine Operation) and 4.7 (Encoding and Decoding).

For this testing exercise, the following was considered:

1. The interoperability tests are performed at the Application layer only for Test 4.
2. Visualization is the method of validation.

### 4.5.2 Methodology and naming conventions

The SEDS files used to test the elements for Test 4 are collected from real devices and examples and their names reflect their source, e.g., Jena\_star\_tracker.xml, example.xml and expressions.xml.

### 4.5.3 NASA TEST SCRIPT AND RESULTS

The ../*test4/nasa/scripts* directory contains a NASA written script that works with the SEDS descriptions for activities and state-machines to render the elements in a pictorial format for visual inspection.

The ../*test4/nasa/logs* directory contains a rendered files of the test execution showing the pictorial rendering for visual inspection.

### 4.5.4 ESA TEST SCRIPT AND RESULTS

The ../*test4/esa/scripts* directory contains a ESA (SciSys) written script that works with the SEDS descriptions for activities and state-machines to render the elements in a pictorial format for visual inspection.

The ../*test4/esa/logs* directory contains a rendered files of the test execution showing the pictorial rendering for visual inspection.

# NASA PROTOTYPE TEST report

The following sections show one example from each of the test runs.

## TESt 0 NASA seds validation TEST results

The report may be found in the *releases/SOIS-CCSDS-XML/seds/interoperability/test\_0/nasa/logs/.* The report shows the results of the validation, and if any errors were encountered. There were many errors as expected because the new term and the new enumerated values were not included in the *seds-extension-semantics.xsd* file. A portion of the test log is shown as follows:

*input/eds/ccsds\_sois\_sampletypes.xml validates*

*input/eds/encodingAndPrecision\_package.xml:9: element FloatDataType: Schemas validity error : Element '{http://www.ccsds.org/schema/sois/seds}FloatDataType', attribute 'unit': 'cochrane' is not a valid value of the union type '{http://www.ccsds.org/schema/sois/seds}Unit'.*

*input/eds/encodingAndPrecision\_package.xml:9: element FloatDataType: Schemas validity error : Element '{http://www.ccsds.org/schema/sois/seds}FloatDataType', attribute 'quantityKind': 'subspaceDistortionQK' is not a valid value of the union type '{http://www.ccsds.org/schema/sois/seds}QuantityKind'.*

*input/eds/encodingAndPrecision\_package.xml:10: element FloatDataEncoding: Schemas validity error : Element '{http://www.ccsds.org/schema/sois/seds}FloatDataEncoding', attribute 'encodingAndPrecision': 'anotherEandP' is not a valid value of the union type '{http://www.ccsds.org/schema/sois/seds}FloatEncodingAndPrecisionType'.*

*input/eds/encodingAndPrecision\_package.xml fails to validate*

*input/eds/errorControl\_package.xml:17: element ErrorControlEntry: Schemas validity error : Element '{http://www.ccsds.org/schema/sois/seds}ErrorControlEntry', attribute 'errorControlType': 'ALWAYS\_ZERO' is not a valid value of the union type '{http://www.ccsds.org/schema/sois/seds}ErrorControlType'.*

*input/eds/errorControl\_package.xml fails to validate*

*input/eds/integerEncoding\_package.xml:10: element IntegerDataEncoding: Schemas validity error : Element '{http://www.ccsds.org/schema/sois/seds}IntegerDataEncoding', attribute 'encoding': 'unary' is not a valid value of the union type '{http://www.ccsds.org/schema/sois/seds}IntegerEncodingType'.*

*input/eds/integerEncoding\_package.xml fails to validate*

*input/eds/interface\_package.xml:42: element Operator: Schemas validity error : Element '{http://www.ccsds.org/schema/sois/seds}Operator', attribute 'operator': 'noOp' is not a valid value of the union type '{http://www.ccsds.org/schema/sois/seds}MathOperatorsType'.*

*input/eds/interface\_package.xml fails to validate*

*input/eds/mathOperator\_package.xml:31: element Operator: Schemas validity error : Element '{http://www.ccsds.org/schema/sois/seds}Operator', attribute 'operator': 'noOp' is not a valid value of the union type '{http://www.ccsds.org/schema/sois/seds}MathOperatorsType'.*

*input/eds/mathOperator\_package.xml fails to validate*

*input/eds/ns2.xml validates*

*input/eds/ns3.xml validates*

*input/eds/ns5.xml validates*

*input/eds/ns7.xml validates*

## TESt 1 NASA seds validation TEST results

The report may be found in the *releases/SOIS-CCSDS-XML/seds/interoperability/test\_1/nasa/logs/.* The report shows the results of the validation, and if any errors were encountered. There were no errors, unlike Test 0 because the Test 1 version of the *seds-extension-semantics.xsd* contains the new unit term and the extended enumerated values of existing types. The test log is shown as follows:

*input/eds/ccsds\_sois\_sampletypes.xml validates*

*input/eds/encodingAndPrecision\_package.xml validates*

*input/eds/errorControl\_package.xml validates*

*input/eds/integerEncoding\_package.xml validates*

*input/eds/interface\_package.xml validates*

*input/eds/mathOperator\_package.xml validates*

*input/eds/ns2.xml validates*

*input/eds/ns3.xml validates*

*input/eds/ns5.xml validates*

*input/eds/ns7.xml validates*

*input/eds/statistical\_package.xml validates*

*input/eds/stringEncoding\_package.xml validates*

*input/eds/valid\_activities\_test.xml validates*

*input/eds/valid\_components\_test.xml validates*

*input/eds/valid\_composite\_datatypes\_test.xml validates*

*input/eds/valid\_datasheet\_test.xml validates*

*input/eds/valid\_datatypes\_test.xml validates*

*input/eds/valid\_device1.xml validates*

*input/eds/valid\_device2.xml validates*

*input/eds/valid\_device3.xml validates*

*input/eds/valid\_device4.xml validates*

*input/eds/valid\_device5.xml validates*

*input/eds/valid\_device6.xml validates*

*input/eds/valid\_implementation\_test.xml validates*

*input/eds/valid\_interfaces\_test.xml validates*

*input/eds/valid\_meta\_data\_test.xml validates*

*input/eds/valid\_namespaces\_test.xml validates*

*input/eds/valid\_ranges\_test.xml validates*

*input/eds/valid\_scalar\_datatypes\_test.xml validates*

*input/eds/valid\_state\_machines\_test.xml validates*

*input/eds/valid\_type\_instance\_test.xml validates*

*input/eds/warpdrive\_package.xml validates*

## TESt 2 NASA seds validation TEST results

The following file was generated using the raw packets generated from areal Jena Star Tracker and decoded using the Jena Star Tracker SEDS by the NASA cFS tools. The results were then re-encoded and autonomously compared to the original raw packet files to generate a go/no-go test. The data from this file is also compared to the ESA (SciSys) implementation visually. For more examples please see the *releases/SOIS-CCSDS-XML/seds/interoperability/test\_2/nasa/logs/.*

Executing LUA: nasacfs/scripts/jena\_test.lua

START JENA STAR TRACKER EDS DECODE TEST SCRIPT

capturedata/jena-0002.bin TEST CASE

Loaded 62 bytes from capturedata/jena-0002.bin

Reference type = CCSDS/SPACELINK/PACKET/PRIMARY\_HEADER

Decoded type = JENA/AS400/TM\_ADB

packetVersionNumber ==> 0

packetType ==> TM

secondaryHeaderFlag ==> true

apid ==> 662

sequenceFlags ==> UNSEGMENTED

packetSequenceCount ==> 1187

packetDataLength ==> 113

pusVersionNumber ==> 1

serviceType ==> 3

serviceSubType ==> 25

destinationId ==> 0

time: <CONTAINER>

coarse ==> 706836847

fine ==> 7500032

timeQuality ==> 21

sid ==> 11

qv1 ==> 928427

qv2 ==> -55531

qv3 ==> 437651932

qs ==> 980500600

rateX ==> -30

rateY ==> 13

rateZ ==> 55

centerOfIntegrationTimeStamp: <CONTAINER>

coarse ==> 2761081

fine ==> 13475124

headIdentification ==> 56

velocityX ==> 0

velocityY ==> 0

velocityZ ==> 0

attitudeQuality ==> 7

adbSpare01 ==> 0

isAberrationCorrected ==> false

rateQuality ==> 2

isValidRate ==> true

attitudeQualityIndex ==> 232

crc ==> 38752

Reencoded to 62 bytes

-----

capturedata/jena-0182.bin TEST CASE

Loaded 88 bytes from capturedata/jena-0182.bin

Reference type = CCSDS/SPACELINK/PACKET/PRIMARY\_HEADER

Decoded type = JENA/AS400/TM\_EV\_TIME\_SHIFT

packetVersionNumber ==> 0

packetType ==> TM

secondaryHeaderFlag ==> true

apid ==> 663

sequenceFlags ==> UNSEGMENTED

packetSequenceCount ==> 33

packetDataLength ==> 73

pusVersionNumber ==> 1

serviceType ==> 5

serviceSubType ==> 3

destinationId ==> 0

time: <CONTAINER>

coarse ==> 706839647

fine ==> 370432

timeQuality ==> 21

rid ==> 33664

evTimeShiftDeviation ==> 127

evTimeShiftSyncSrc ==> 1

crc ==> 38967

Reencoded to 26 bytes

Decoded type = JENA/AS400/TM\_ADB

packetVersionNumber ==> 0

packetType ==> TM

secondaryHeaderFlag ==> true

apid ==> 662

sequenceFlags ==> UNSEGMENTED

packetSequenceCount ==> 1367

packetDataLength ==> 113

pusVersionNumber ==> 1

serviceType ==> 3

serviceSubType ==> 25

destinationId ==> 0

time: <CONTAINER>

coarse ==> 706839791

fine ==> 7886592

timeQuality ==> 21

sid ==> 11

qv1 ==> 920973

qv2 ==> -50686

qv3 ==> 437686721

qs ==> 980485078

rateX ==> -7

rateY ==> 37

rateZ ==> 195

centerOfIntegrationTimeStamp: <CONTAINER>

coarse ==> 2761092

fine ==> 5085845

headIdentification ==> 56

velocityX ==> 0

velocityY ==> 0

velocityZ ==> 0

attitudeQuality ==> 7

adbSpare01 ==> 0

isAberrationCorrected ==> false

rateQuality ==> 2

isValidRate ==> true

attitudeQualityIndex ==> 235

crc ==> 60603

Reencoded to 62 bytes

## TESt 3 NASA seds validation TEST results

The following file was generated using synthetic data generated and encoded using an examples type SEDS by the NASA cFS tools. The result was then de-encoded and autonomously compared to the original de-coded source files to generate a go/no-go test. The data from this file is also compared to the ESA (SciSys) implementation visually. For more examples please see the *releases/SOIS-CCSDS-XML/seds/interoperability/test\_3/nasa/logs/.*

Executing LUA: scripts/decode\_test.lua

START CCSDS/SOIS EDS DECODE TEST SCRIPT

ALL\_BOOLEANS TEST CASE

Loaded 1 bytes from data/TESTDATA\_ALL\_BOOLEANS.bin

Reference type = CCSDS/SOIS/SAMPLETYPES/ALL\_BOOLEANS

Decoded type = CCSDS/SOIS/SAMPLETYPES/ALL\_BOOLEANS

normal ==> true

invert ==> true

-----

ALL\_UNSIGNED\_INTS TEST CASE

Loaded 26 bytes from data/TESTDATA\_ALL\_UNSIGNED\_INTS.bin

Reference type = CCSDS/SOIS/SAMPLETYPES/ALL\_UNSIGNED\_INTS

Decoded type = CCSDS/SOIS/SAMPLETYPES/ALL\_UNSIGNED\_INTS

u8el ==> 1

u8eb ==> 2

u16el ==> 3

u16eb ==> 4

u32el ==> 5

u32eb ==> 6

u7 ==> 7

u11 ==> 8

u21 ==> 9

u36 ==> 10

fixed5\_u21 ==> 5

-----

ALL\_SIGNED\_INTS TEST CASE

Loaded 39 bytes from data/TESTDATA\_ALL\_SIGNED\_INTS.bin

Reference type = CCSDS/SOIS/SAMPLETYPES/ALL\_SIGNED\_INTS

Decoded type = CCSDS/SOIS/SAMPLETYPES/ALL\_SIGNED\_INTS

s8el ==> -1

s16el ==> -2

s32el ==> -3

s6eb ==> -4

s15eb ==> -5

s27eb ==> -6

oc20eb ==> -7

sm40el ==> -8

bcd40eb ==> 123

bcd56el ==> 456

fixedminus4 ==> -4

fixedplus10 ==> 10

-----

## TESt 4 NASA seds validation TEST results

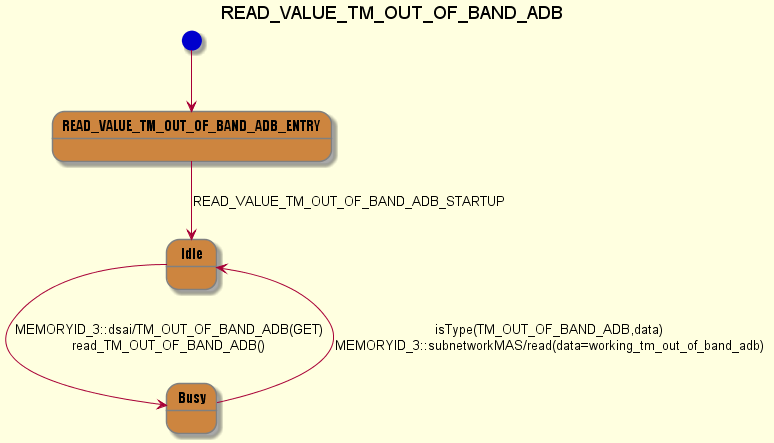
The following diagram, Figure 2, is an example of a state machine renderied using the NASA tool-chain. For more examples please see *releases/SOIS-CCSDS-XML/seds/interoperability/test\_4/nasa/logs/.*

Figure 2. NASA SEDS State-Machine Rendering

# ESA PROTOTYPE TEST report

The following sections show one example from each of the test runs.

## TESt 0 ESA seds validation TEST results

The report may be found in the *releases/SOIS-CCSDS-XML/seds/interoperability/test\_0/esa/logs/. The report shows the results of the validation, and if any errors were encountered. There were expected errors, but they validate correct interpretation of the SEDS.*

## TESt 1 ESA seds validation TEST results

The report may be found in the *releases/SOIS-CCSDS-XML/seds/interoperability/test\_1/esa/logs/. The report shows the results of the validation, and if any errors were encountered. There were no errors.*

## TESt 2 ESA seds validation TEST results

The following file was generated using the raw packets generated from a real Jena Star Tracker and decoded using the Jena Star Tracker SEDS by the ESA (SciSys) tools. The results were then re-encoded and autonomously compared to the original raw packet files to generate a go/no-go test. The data from this file is also compared to the NASA (cFS) implementation visually. For more examples please see the *releases/SOIS-CCSDS-XML/seds/interoperability/test\_2/esa/logs/.*

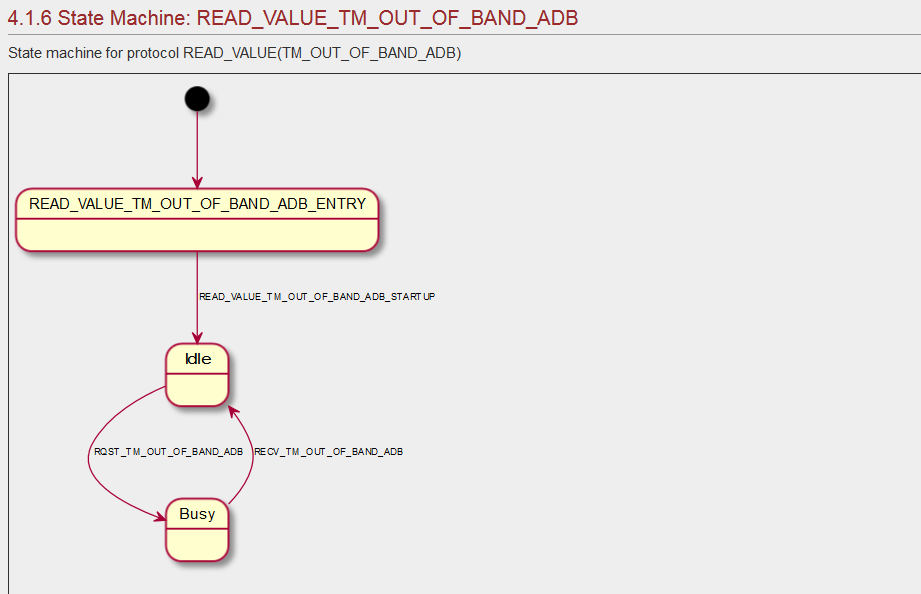


## TESt 3 ESA seds validation TEST results

The following file was generated using synthetic data generated and encoded using an examples type SEDS by the ESA (SciSys) tools. The result was then de-encoded and autonomously compared to the original de-coded source files to generate a go/no-go test. The data from this file is also compared to the NASA (cFS) implementation visually. For more examples please see the *releases/SOIS-CCSDS-XML/seds/interoperability/test\_3/esa/logs/.* 

## TESt 4 ESA seds validation TEST results

The following diagram, Figure 3, is an example of a state machine renderied using the ESA tool-chain. For more examples please see the *releases/SOIS-CCSDS-XML/seds/interoperability/test\_4/esa/logs/.*



|  |  |
| --- | --- |
|  | Figure 3. ESA SEDS State-Machine Rendering |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |