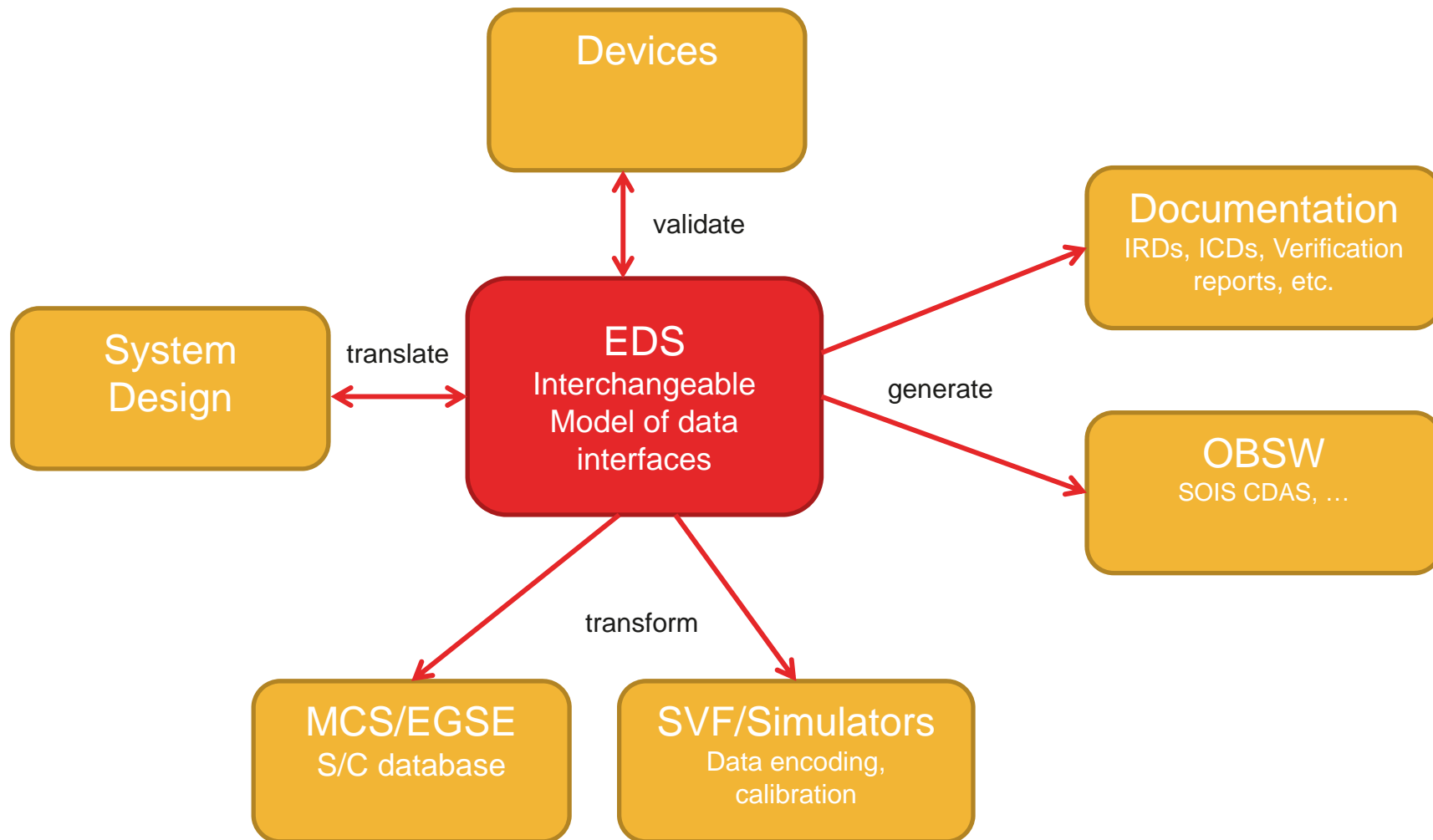


EDS PnP Final presentation

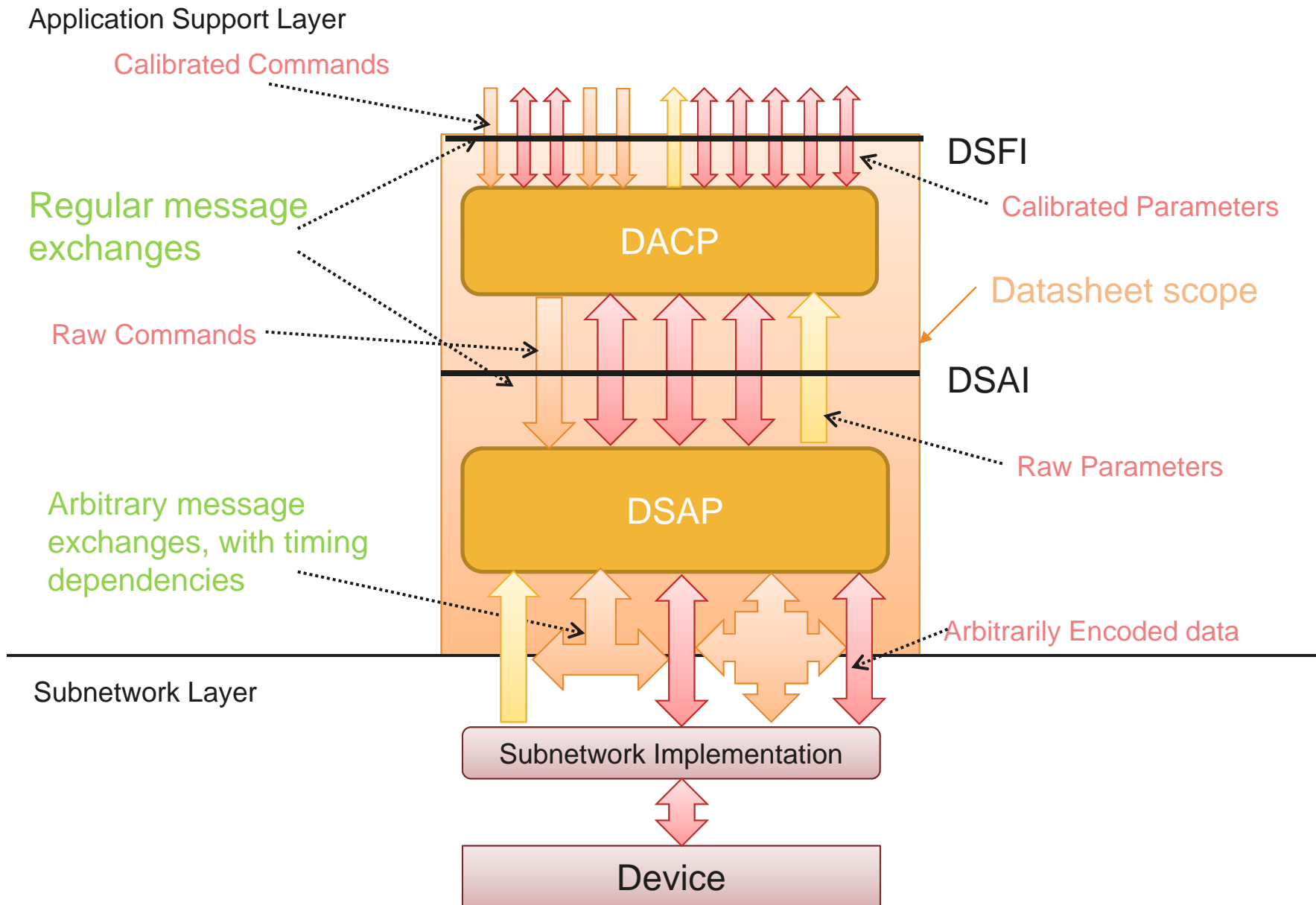
Deploying Plug and Play Avionics (PTRP)

- ESA Contract No. **4000113033 /15/NL/LF**
- 250k Euro
- 20 Months - actual
- Prime: **SCISYS UK** Ltd
- Subs:
 - Airbus Defence and Space Ltd.
 - Thales Alenia Space UK Ltd.
 - Jena Optronik GmbH
- Objectives:
 - Improve understanding and take-up of **electronic data sheets**
 - Demonstrate the process, tools and outputs related to the **production** of EDS.
 - European vendors delivering products to a LSI
 - Authoring of validated electronic data sheets
 - Refinement of **reference tooling**.

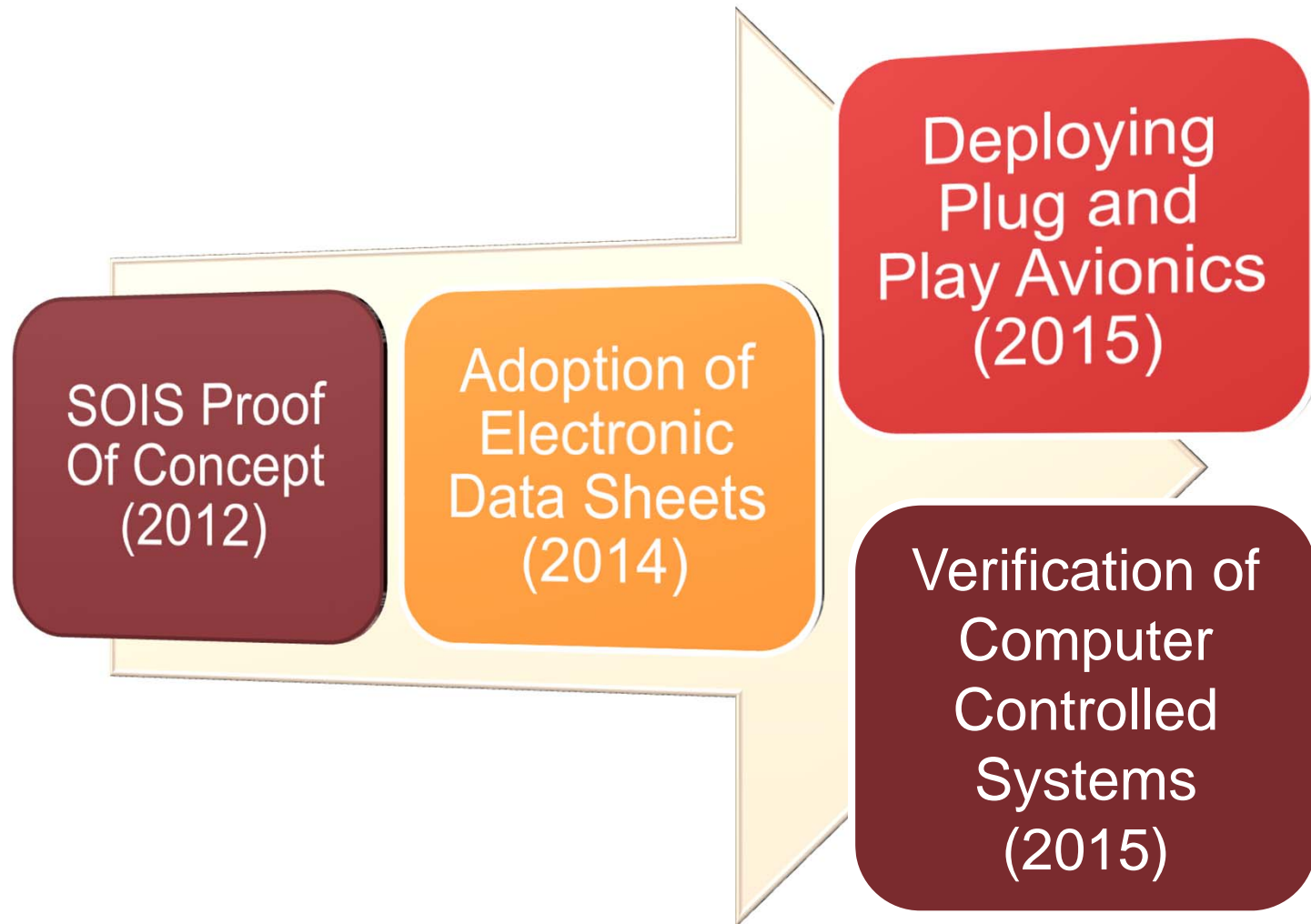
SOIS EDS: Common Interchange Format



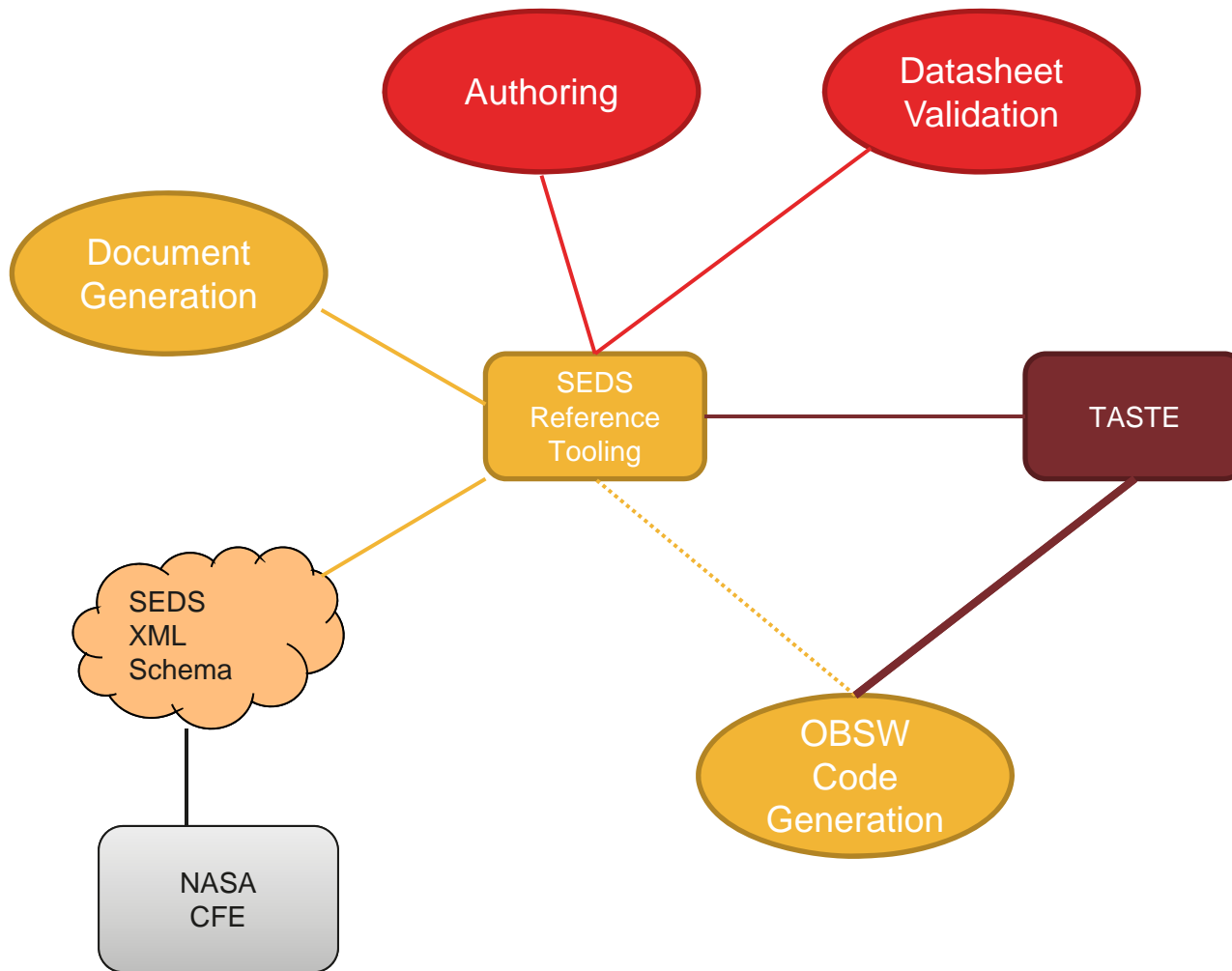
EDS IN CCSDS Reference Architecture



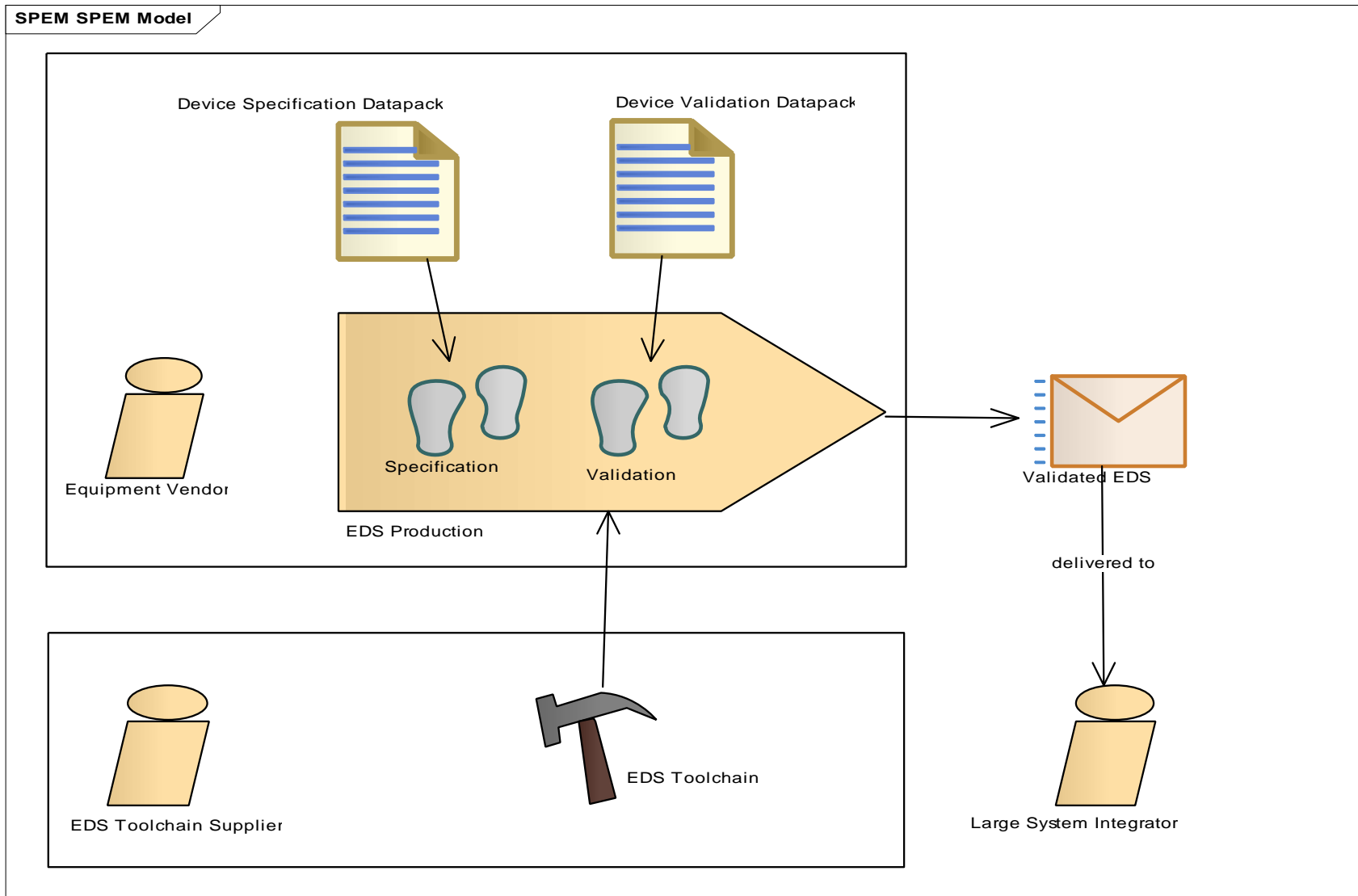
ESA Study Projects



Tooling Status



Deploying Plug And Play Avionics: Scope



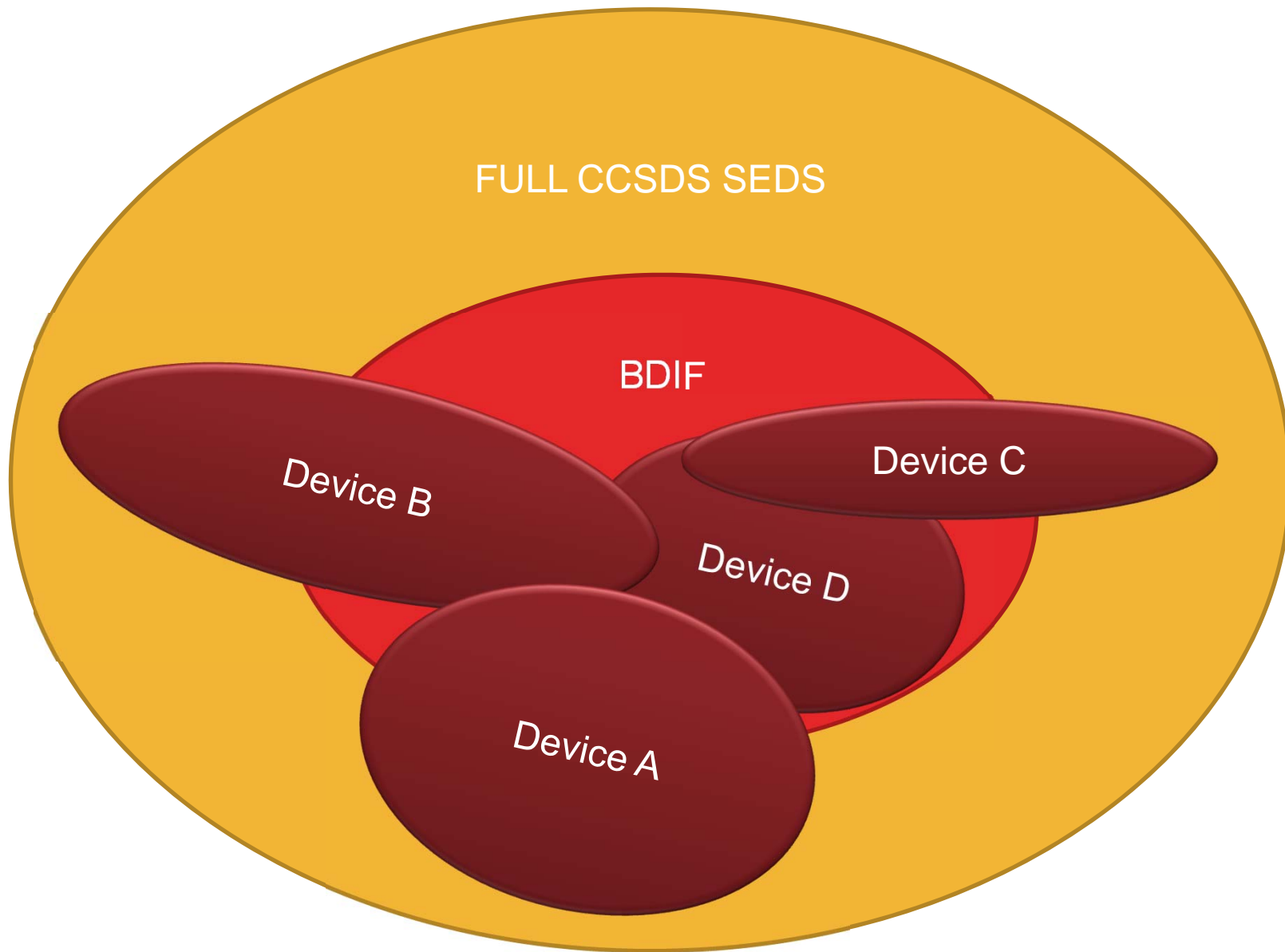
Interviews with Vendors and Primes

- Vendors
 - Jena Optronik
 - TAS UK
- LSI (aka Prime)
 - Airbus
- Topics covered:
 - Feasibility
 - IP considerations
 - Preferred Input Format
 - How to Verify
 - Concerns

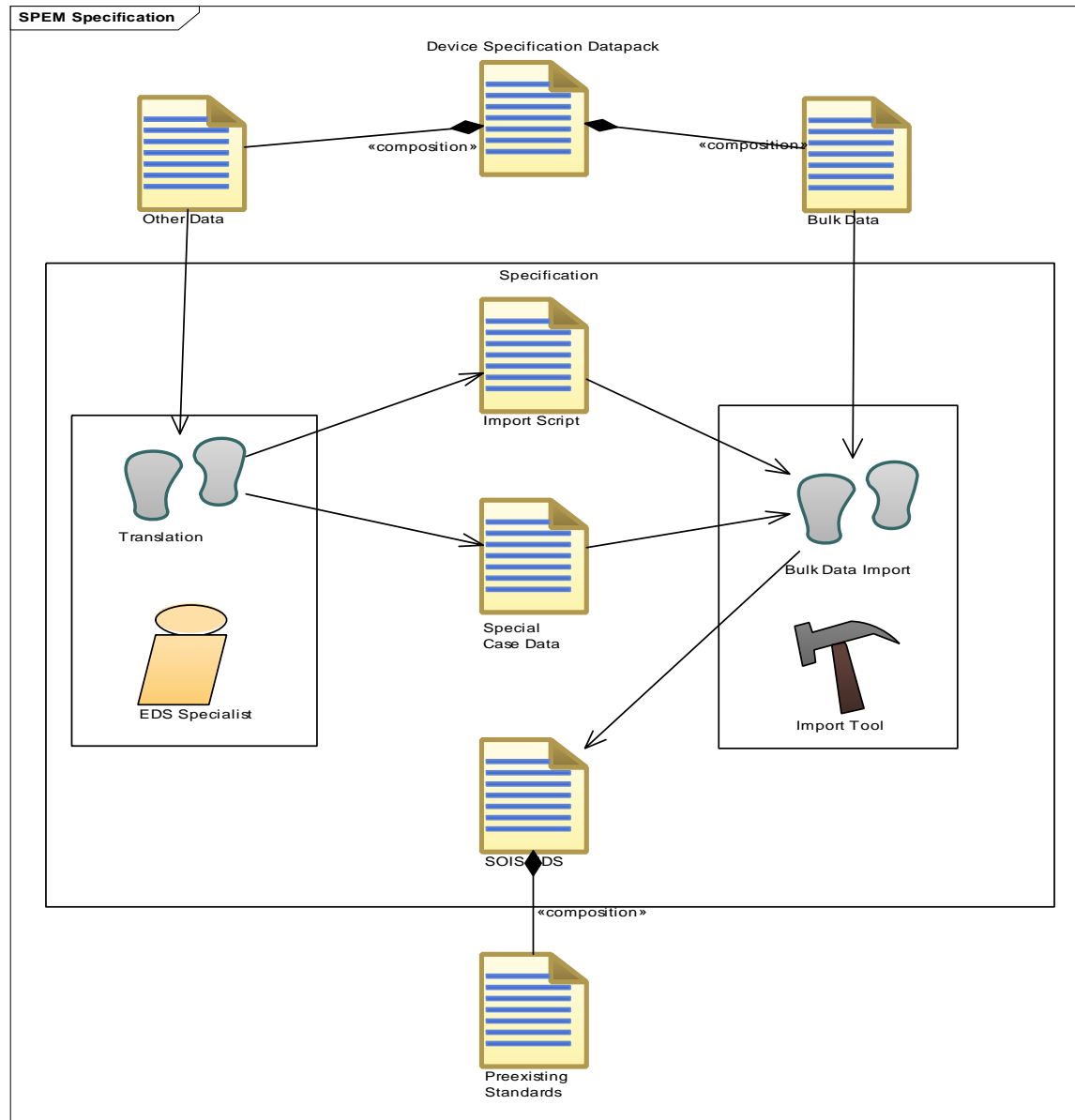
Vendor Concerns

- Do not want to give away detailed implementation information (e.g. full models).
- Want to define information once, and deliver it many times
- Need to be able to handle all the special cases their device requires
- Do not want to be concerned with all the complexity other manufacturer's devices may require

Complexity of Standards and Devices



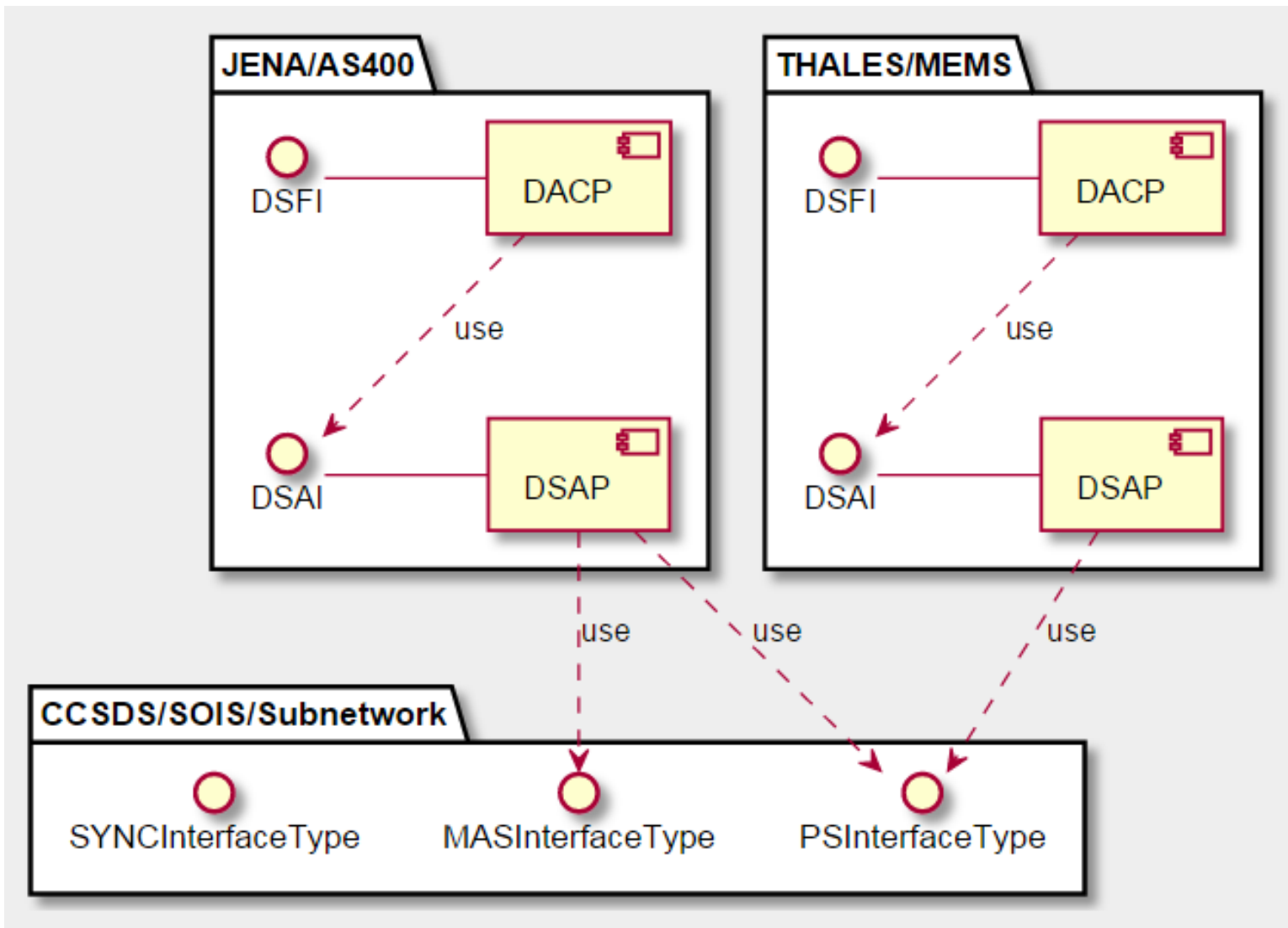
Bulk Data Import



Authoring Real-world devices

- TAS-UK MEMS Rate Sensor
 - Small device, so no bulk data
 - Datasheet created by writing an import script based on PDF ICD.
- Jena AS400 Star Tracker
 - Bulk data automatically translated from a custom XML schema to the tooling format
 - Special cases handled by an import script based on PDF ICD.

Index diagram for both device datasheets



Datasheet: example.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<DataSheet xmlns="http://www.ccsds.org/schema/sois/seds"
  xmlns:xi="http://www.w3.org/2001/XInclude"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://www.ccsds.org/schema/sois/seds seds.xsd">
  <xi:include href="ccsds.sois.subnetwork.xml" xpointer="element(/1/1)"/>
  <Namespace name="Demo">
    <!-- This is the set of all parameter types which are used in the public interfaces to the component types described in
    this namespace -->
    <DataTypeSet>
      <!-- This is the set of all interface types used by component types in this namespace -->
      <DeclaredInterfaceSet>
        <Interface name="DeviceFunctionalInterface" level="functional">
          <Interface name="DeviceAccessInterface" level="access">
            <ParameterSet>
              <CommandSet>
                <!-- Commands issued over the PS interface to the device -->
                <Command name="setMode" mode="sync">
                  <Argument mode="in" name="mode" type="ModeType"/>
                  <Argument mode="out" name="status" type="CommandStatusType"/>
                </Command>
                <Command name="setUserData" mode="sync">
                  <Argument mode="in" name="data" type="CCSDS/SOIS/Subnetwork/Octet"/>
                  <Argument mode="out" name="status" type="CommandStatusType"/>
                </Command>
              </CommandSet>
            </Interface>
          </DeclaredInterfaceSet>
          <!-- Finally here are the actual components -->
          <ComponentSet>
            <Component name="DemoDeviceDACP">
              <Component name="DemoDeviceDSAP">
            </ComponentSet>
          </Namespace>
        </DataSheet>
```

Run Commands:

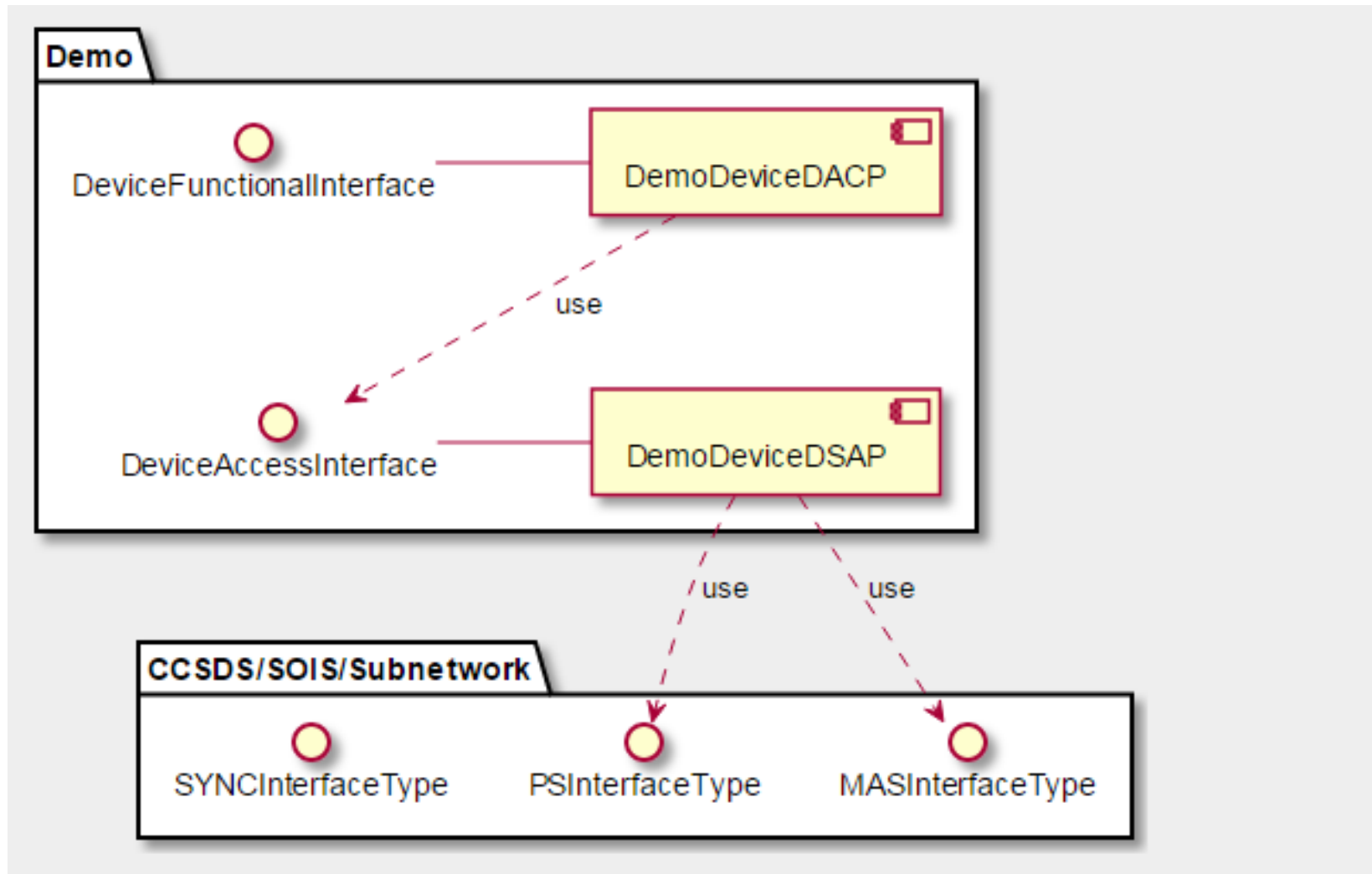
Generate HTML Documentation for example datasheet

```
java -jar SedsTooling.jar docgen  
test-data/xml/example.xml  
--output-dir example_html --level ALL
```

Generate HTML Index and Component Diagram for example datasheet

```
java -jar SedsTooling.jar docgen  
test-data/xml/example.xml  
--output-dir example_html --level INDEX
```

Component Diagram from example.xml



Device Access Interface from example.xml

2.2 Interface: DeviceAccessInterface

Name	Details
Level	ACCESS

Interface Properties

Name	Type	Unit	Ranges	Details	Description
deviceEvent	EventType			READ ONLY ASYNC	
extendedMode	ExtendedModeType		Inherent:[0..255]	READ ONLY SYNC	
extendedStatus	ExtendedStatusType		Inherent:[0..255]	READ ONLY SYNC	
queryCount	QueryCountType		Inherent:[0..255]	READ ONLY SYNC	
status	StatusType			READ ONLY SYNC	
telemetrySet1	TelemetrySet1Type			READ ONLY ASYNC	
telemetrySet2	TelemetrySet2Type			READ ONLY ASYNC	

Interface Parameters for DeviceAccessInterface

Command	Argument	Type	Unit	Range	Description
setMode		SYNC			
	mode	ModeType		[0..9]	IN
	status	CommandStatusType		[0..65535]	OUT
setUserData		SYNC			
	data	Octet		[0..255]	IN
	status	CommandStatusType		[0..65535]	OUT

Interface Commands for DeviceAccessInterface

Subnetwork Interface from example.xml

3.1 Subnetwork Interface: subnetworkPS

SOIS SUBNETWORK PACKET SERVICE from 851.0-M-1

Command	Argument	Type	Unit	Range	Description
receive		ASYNCR	PACKET_RECEIVE.indication primitive only		
	data	TelemetryType			OUT
send		ASYNCR	PACKET_SEND.request primitive only		
	data	TelecommandType			IN

Interface Commands for subnetworkPS

SendData Type	ReceiveData Type
TelecommandType	TelemetryType

Fixed Type Bindings for subnetworkPS

The above table describes how the types in the interface definition are unconditionally used within this component. Each column is a generic type, and each row is a specific type.

3.2.1 PDU Classification Tables

3.2.1.1 Abstract PDU: TelecommandType

type	Telecommand Type
Mode	TelecommandModeType
UserData	TelecommandUserData Type

Abstract PDU classification table for TelecommandType

The above table describes how a value of this abstract type can be classified as a value of a concrete PDU. Each column is a packet field, and each row is a matching concrete one.

3.2.1.2 Abstract PDU: TelemetryType

Common header for TM

type	Telemetry Type
Type1	Telemetry1Type
Type2	Telemetry2Type
Ack	TelemetryAckType
Event	TelemetryEventType

Abstract PDU classification table for TelemetryType

PDU Detail from example.xml

3.2.2.6 PDU: TelemetryEventType

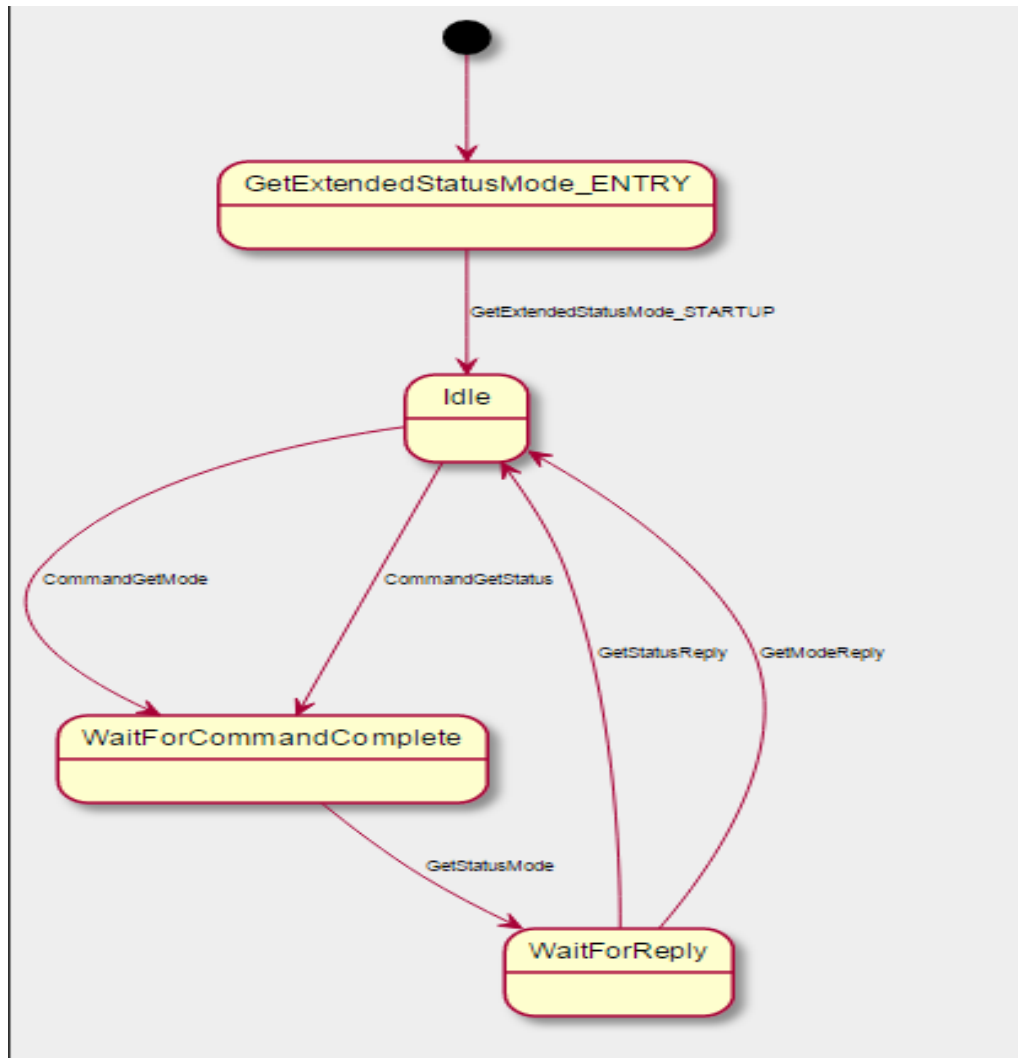
Event packet

Byte Offset	Bit Range	Field Name	Type	Encoding	Fixed Value	Description
0	[0..31]	type	TelemetryTypeEnumType	SIGNED	Event	
4	[0..31]	eventType	EventTypeType	SIGNED		
8	[0..7]	eventData	EventDataTypes	UNSIGNED		

Fixed byte length is 9

PDU Binary Encoding for TelemetryEventType

Behavioral Model Detail



Artificial test data: example.raw

#Telemetry2Type

00 00 00 02 00 00

00 00 00 02 33 44

00 00 00 02 55 aa

#TelemetryEventType

00 00 00 03 00 00 00 00 00

00 00 00 03 00 00 00 01 00

00 00 00 03 00 00 00 02 22

00 00 00 03 00 00 00 00 05

#Illegal TelemetryEventType

00 00 00 03 00 00 00 09 05

00 00 00 03 00 00 00 12 05

00 00 00 03 00 00 00 23 05



Run Command:

Statically validate the datasheet against schema and additional rules from 876x0.

```
java -jar SedsTooling.jar validate  
test-classes/xml/example.xml
```

Verify correctness of datasheet against sample data

```
java -jar SedsTooling.jar verify  
test-classes/xml/example.xml  
--data-encoding HEX_BYTE  
--test-data test-classes/test_logs/example.raw  
--output-dir example_html
```

Verification Report: Overview

1. Overview

Verification report for datasheet example generated from test data file(s) test-classes\test_logs\example.raw.

2. Errors And Warnings

Summary of all issues detected during the test session.

Level	Message
INFO	Validating CCSDS/SOIS/Subnetwork/PSInterfaceType
INFO	Validating CCSDS/SOIS/Subnetwork/MASInterfaceType
INFO	Validating CCSDS/SOIS/Subnetwork/SYNCInterfaceType
INFO	Validating CCSDS/SOIS/Subnetwork
INFO	Validating Demo/DeviceFunctionalInterface
INFO	Validating Demo/DeviceAccessInterface
INFO	Validating Demo/DemoDeviceDACP/VirtualInterface
INFO	Validating Demo/DemoDeviceDSAP/DeviceInterface
INFO	Validating Demo

Static Validation Messages

PDU	Level	Message
TelemetryEventType	WARN	Illegal EventTypeType '9' at eventType
	WARN	State machine Demo/DemoDeviceDSAP/Telemetry1 in state Idle rejected message subnetworkPS/receive reply @ PT0.01S
	WARN	State machine Demo/DemoDeviceDSAP/Telemetry2 in state Idle rejected message subnetworkPS/receive reply @ PT0.01S
	WARN	State machine Demo/DemoDeviceDSAP/TC in state Idle rejected message subnetworkPS/receive reply @ PT0.01S
	WARN	State machine Demo/DemoDeviceDSAP/Event in state Idle rejected message subnetworkPS/receive reply @ PT0.01S

Subnetwork Interface: subnetworkPS : PSInterfaceType

Verification Report: Subnetwork Coverage

3. Subnetwork Level PDU Coverage

For each interface in the datasheet, list all defined Protocol Data Units, and record whether or not they were seen during the test session.

PDU	Count	Sample
TelecommandModeType	0	
TelecommandUserData Type	0	
Telemetry1Type	4	Telemetry1Type
Telemetry2Type	3	Telemetry2Type
TelemetryAckType	5	TelemetryAckType
TelemetryEventType	8	TelemetryEventType

Subnetwork Interface: subnetworkPS : PSInterfaceType

PDU	Count	Sample
ExtendedModeUnionType	0	
ExtendedStatusUnionType	0	
StatusType	0	

Subnetwork Interface: subnetworkMAS : MASInterfaceType

Verification Report: Low level packet log

PDU: TelemetryEventType : *Event packet*

PDU TelemetryEventType

Byte Offset	Bit Range	Field Name	Encoding	Description	Value	Raw Value
0	[0..32)	type	SIGNED		Event	00 00 00 03
4	[0..32)	eventType	SIGNED		Event1	00 00 00 00
8	[0..8)	eventData	UNSIGNED		5	05
INFO	State machine Demo/DemoDeviceDSAP/Event starting transition ReceiveEvent from state Idle to state Idle after PT0S					
INFO	State machine Demo/DemoDeviceDSAP/Event completed transition ReceiveEvent: on subnetworkPS/receive [reply] (data:eventTM) [(data instanceof TelemetryEventType)] -> Idle from state Idle to state Idle					
INFO	ACCESS Message Aggregate update DeviceInterface/deviceEvent @ PT0.01S was ignored by all state machines					

PDU: TelemetryEventType : *Event packet*

PDU TelemetryEventType

Byte Offset	Bit Range	Field Name	Encoding	Description	Value	Raw Value
0	[0..32)	type	SIGNED		Event	00 00 00 03
4	[0..32)	eventType	SIGNED			00 00 00 09
8	[0..8)	eventData	UNSIGNED		5	05
WARN	Illegal EventTypeType '9' at eventType					
WARN	State machine Demo/DemoDeviceDSAP/Telemetry1 in state Idle rejected message subnetworkPS/receive reply @ PT0.01S					
WARN	State machine Demo/DemoDeviceDSAP/Telemetry2 in state Idle rejected message subnetworkPS/receive reply @ PT0.01S					
WARN	State machine Demo/DemoDeviceDSAP/TC in state Idle rejected message subnetworkPS/receive reply @ PT0.01S					
WARN	State machine Demo/DemoDeviceDSAP/Event in state Idle rejected message subnetworkPS/receive reply @ PT0.01S					

Verification Report: Behavioral Model Coverage

4. Behavioural Model Coverage

For each state machine defined in the datasheet, list all states and transitions, and record whether or not they were visited during

Component	Machine	State	Count
Demo/DemoDeviceDSAP	Event	Event_ENTRY	1
Demo/DemoDeviceDSAP	Event	Idle	10
Demo/DemoDeviceDSAP	GetExtendedStatusMode	GetExtendedStatusMode_ENTRY	1
Demo/DemoDeviceDSAP	GetExtendedStatusMode	Idle	0
Demo/DemoDeviceDSAP	GetExtendedStatusMode	WaitForCommandComplete	0
Demo/DemoDeviceDSAP	GetExtendedStatusMode	WaitForReply	0
Demo/DemoDeviceDSAP	GetStatus	GetStatus_ENTRY	1
Demo/DemoDeviceDSAP	GetStatus	Idle	0
Demo/DemoDeviceDSAP	GetStatus	WaitForReply	0
Demo/DemoDeviceDSAP	TC	Idle	0
Demo/DemoDeviceDSAP	TC	TC_ENTRY	1
Demo/DemoDeviceDSAP	TC	WaitForReply	0
Demo/DemoDeviceDSAP	Telemetry1	Idle	4
Demo/DemoDeviceDSAP	Telemetry1	Telemetry1_ENTRY	1
Demo/DemoDeviceDSAP	Telemetry2	Idle	3
Demo/DemoDeviceDSAP	Telemetry2	Telemetry2_ENTRY	1
Demo/DemoDeviceDACP	internal__mappings	stateless	5
Demo/DemoDeviceDSAP	internal__mappings	stateless	1

State Coverage Table

Real World Example: Jena Star Tracker

- No device-specific information 'baked into' the tool; everything is in the EDS.
- Datasheet is 7600 lines of xml, covering 80 TC and 32 TM.
- Test data (supplied by manufacturer) is 1Mb of raw binary milbus log.
- Corresponding low-level html packet log is ~40 Mb of html (split into 20 chunks for readability).
- Validation Summary identifies the one error in the file (reported to manufacturer)

Real World Example: Verification Report

PDU	Level	Message
TM_EDB	WARN	Incorrect CRC_16_CCITT '0' (expected bee9) at 2 bytes from end of TM_EDB
	INFO	State machine JENA/AS400/DSAP/ASYNC starting transition RECV_TM_EDB from state Idle to state Idle after PT0.28S
	INFO	State machine JENA/AS400/DSAP/ASYNC completed transition RECV_TM_EDB from state Idle to state Idle
	INFO	ACCESS Message Aggregate update dsai/TM_EDB @ PT22.1S was ignored by all state machines
TM_EV_TIME_SHIFT	INFO	State machine JENA/AS400/DSAP/ERROR starting transition RECV_TM_EV_TIME_SHIFT from state Idle to state ERROR after PT9M31.8S
	WARN	State machine JENA/AS400/DSAP/ERROR completed transition RECV_TM_EV_TIME_SHIFT from state Idle to exit state ERROR
	INFO	State machine JENA/AS400/DSAP/ERROR starting transition ERROR_STARTUP from state ERROR_ENTRY to state Idle after PT0S
	INFO	State machine JENA/AS400/DSAP/ERROR completed transition ERROR_STARTUP from state ERROR_ENTRY to state Idle

Subnetwork Interface: subnetworkPS : PSInterfaceType

Real World Example: Failed packet

453	[0..4)	windowSize[9].width	UNSIGNED	0	0000
	[4..8)	windowSize[9].height	UNSIGNED	0	0000
454	[0..4)	windowSize[10].width	UNSIGNED	0	0000
	[4..8)	windowSize[10].height	UNSIGNED	0	0000
455	[0..4)	windowSize[11].width	UNSIGNED	0	0000
	[4..8)	windowSize[11].height	UNSIGNED	0	0000
456	[0..4)	windowSize[12].width	UNSIGNED	0	0000
	[4..8)	windowSize[12].height	UNSIGNED	0	0000
457	[0..4)	windowSize[13].width	UNSIGNED	0	0000
	[4..8)	windowSize[13].height	UNSIGNED	0	0000
458	[0..4)	windowSize[14].width	UNSIGNED	0	0000
	[4..8)	windowSize[14].height	UNSIGNED	0	0000
459	[0..4)	windowSize[15].width	UNSIGNED	0	0000
	[4..8)	windowSize[15].height	UNSIGNED	0	0000
460	[0..16)	crc	UNSIGNED Packet checksum.	0	00 00
WARN Incorrect CRC_16_CCITT '0' (expected bee9) at 2 bytes from end of TM_EDB					
INFO	State machine JENA/AS400/DSAP/ASYNC starting transition RECV_TM_EDB from state Idle to state Idle after PT0.28S				
INFO	State machine JENA/AS400/DSAP/ASYNC completed transition RECV_TM_EDB: on subnetworkPS/receive [reply] (data:working_tm_edb) [(data instanceof TM_EDB)] -> Idle from state Idle to state Idle				
INFO	ACCESS Message Aggregate update dsai/TM_EDB @ PT22.1S was ignored by all state machines				

Jena Star Tracker in Sentinel Simulator

- S2A simulator uses (an earlier revision of) the AS400 Star Tracker.
- Trace log of running the standard Star Tracker model regression tests is XXMb.
- Corresponding low-level html packet log is ~YY Mb of html (split into ZZ chunks for readability).
- Validation Summary identifies TBD
- SCISYS Test Front End can run tests online and, using integrated copy of EDS tooling, check for violations in real-time.

Recap

- Validation and Verification of Device Interfaces is key to early, and hence cheap, problem detection and remediation.
- Capturing device interfaces formally allows tools to be used to support that activity.
- SOIS EDS Reference Tooling provides a Proof of Concept implementation of such a device testing tool.
- Adding EDS support into existing test tools (e.g. simulators, SVF, EGSE, etc.) would allow additional testing capabilities.