Adoption of Electronic Data Sheets and Device Virtualisation for onboard devices

Final Presentation days

Estec May 2014

Chris Taylor – TEC-ED

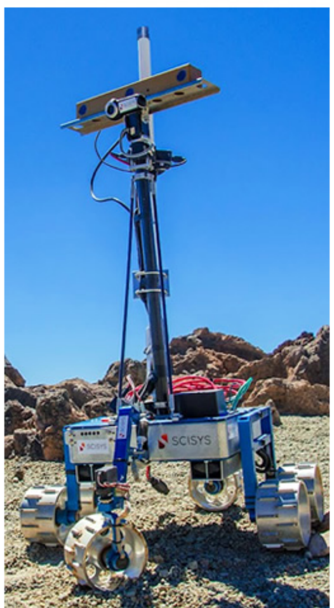
Context

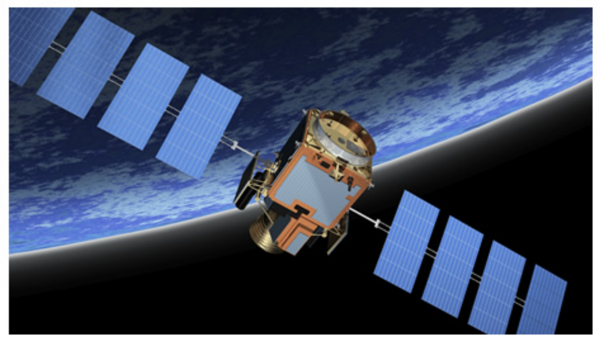
- The standardisation of the flight avionics infrastructure is progressing on many different fronts:
  - The Savoir WG has developed a reference avionics architecture identifying individual building blocks
  - The CCSDS SOIS working group has made significant progress in standardising the Avionics communications services with many standards already under publication
  - The ECSS has developed a number of Datalink protocols and device interface standards to be used for avionics interconnection
- The primary objective of the activity was to demonstrate the use of electronic datasheets in the above context to supplement existing paper ICDs, and to employ the results to interfacing onboard equipment
- TRP Contract with SciSys - 200k 12 Month TRP activity

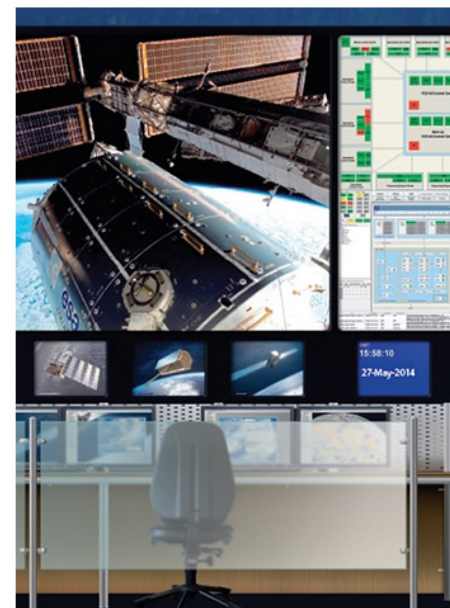
Adoption of Electronic Data Sheets and Device Virtualisation for Onboard Devices

Stuart Fowell

22 May 2014







Overview

- Consortium and Project Objectives
- Electronic Data Sheet Requirements
- XML Technologies Survey
- Overview of SEDS XML Schema
- SEDS Demonstrator
- Recommendations for Future Work
- Conclusions

SCISYS

TRP Project and Consortium

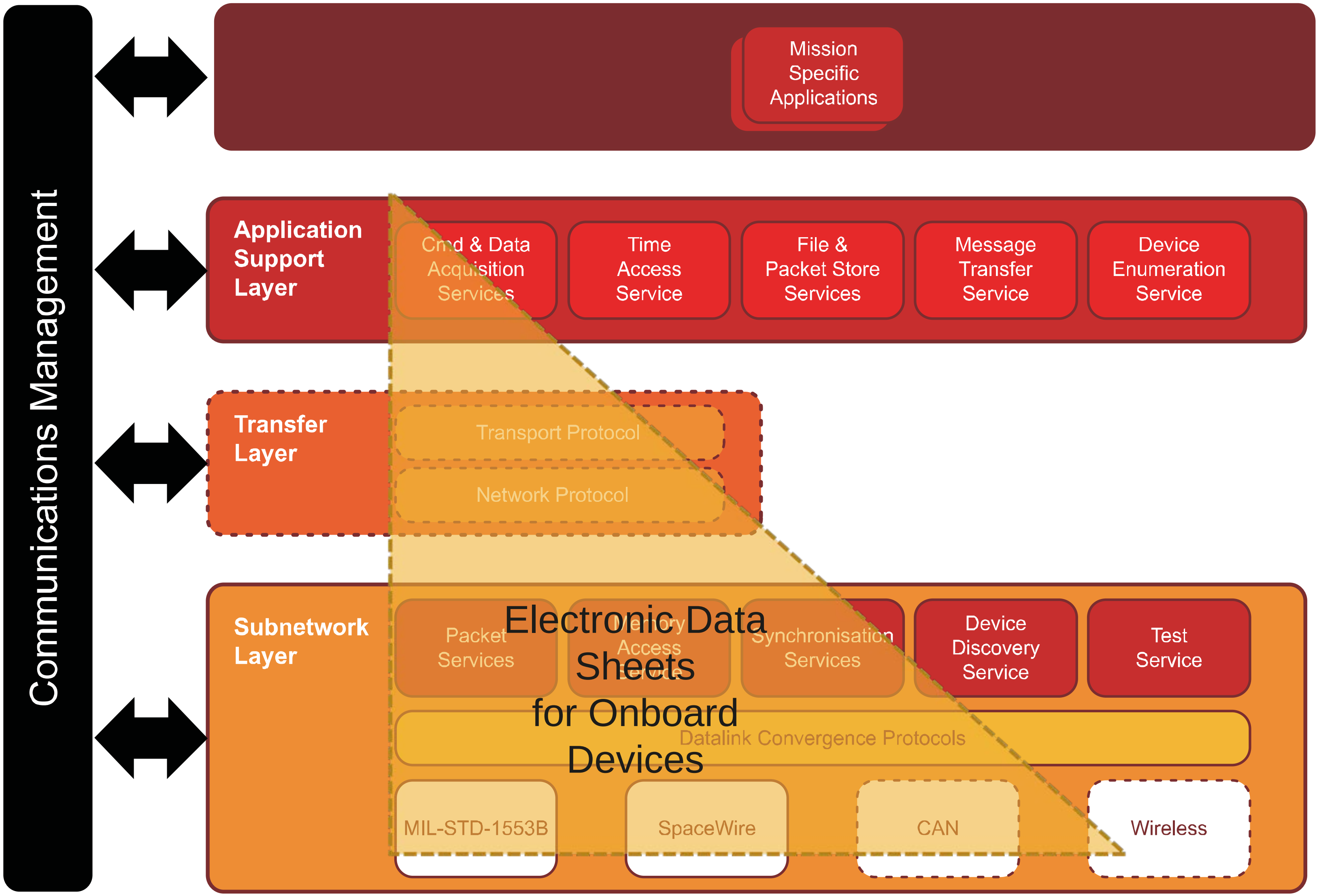
- Project
  - » Follow-on from SOIS Proof of Concept TRP study
  - » SCISYS supported by Astrium (F) (reqs & ICDs) and TAS-F (reqs, ICDs & schema review)
  - » 15 month TRP study, kicked on September 2012 completing January 2014
- Objectives
  - » EDS Use Cases Capture and resulting Requirements
  - » Definition of EDS XML Schema and Specification
  - » Test with defining EDS from real-world ICDs
  - » Proof of Concept demonstration of code and ICD generation from EDS
- Outputs
  - » EDS XML Schema & draft CCSDS SOIS standard
    - › In cooperation with CCSDS SOIS WG and SAVOIR-SAFI WG
  - » Example Functional Interfaces and EDS for selected real-world devices
    - › Hydra Star Tracker, FOG Gyro, NPAL Camera
    - › Use of draft Common Dictionary of Terms from AFRL
    - › Using SAVOIR-SAFI generic Functional Interfaces, where possible
  - » Proof of Concept Demonstration on RASTA
    - › EDS-generation toolkit
    - › SOIS and ICD documentation auto-generation
    - › Demonstration of OBSW using auto-generated SOIS to interface to simulated devices
      - Based on SOIS Proof of Concept software

Electronic Data Sheets replacing Device ICDs

- Function Interface information for a device is today typically provided within an ICD
  - Paper document
    - Different formats from different organisations
    - With potentially different levels of information provided
  - Requires extensive testing for inconsistencies with implemented device
  - Requires manual translation to:
    - OBSW development
    - Spacecraft databases
    - Simulators
    - Mission Control System databases
    - Others?

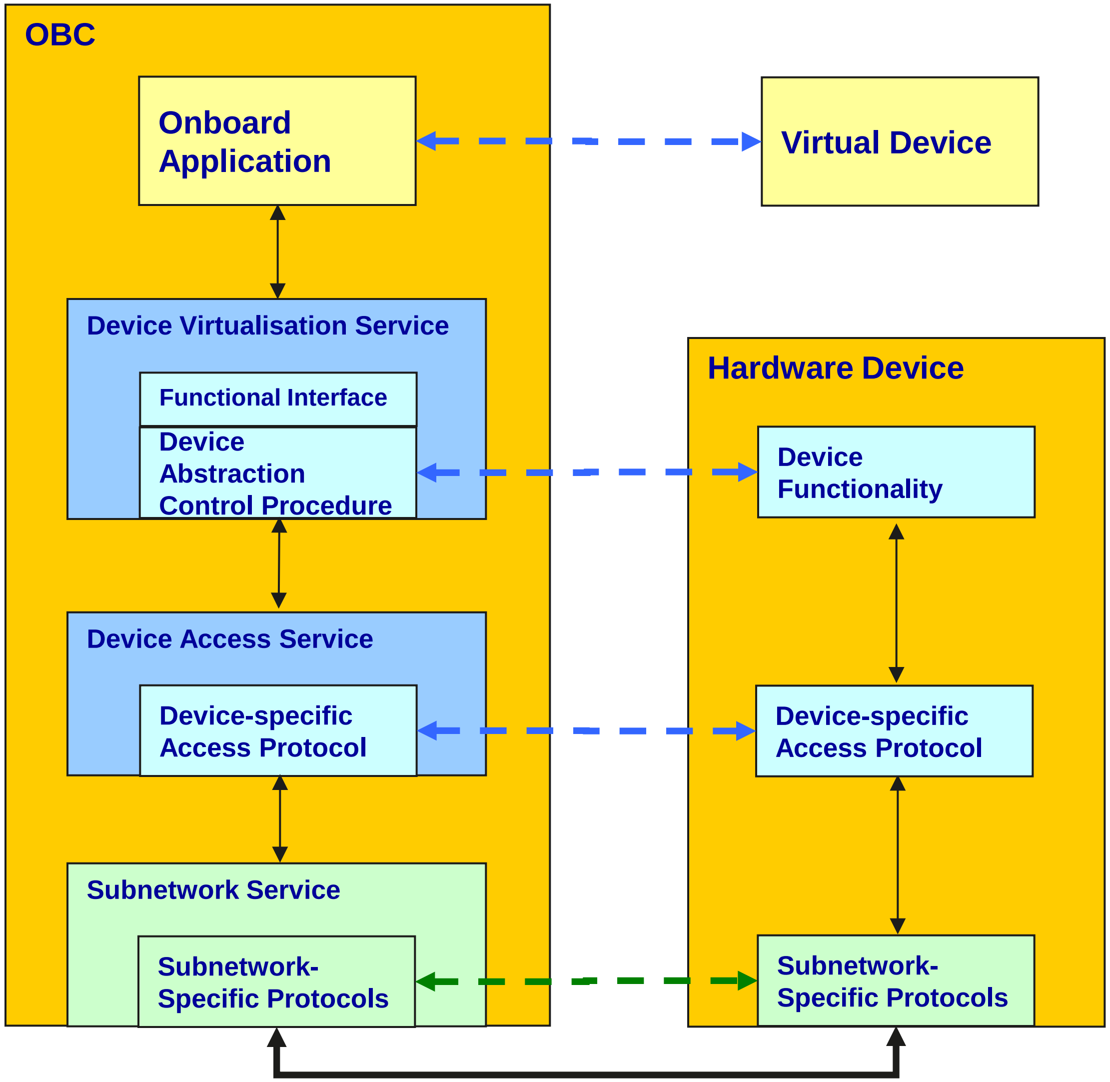
- Define Electronic Data Sheets to replace ICDs
  - Capture electronically all information
  - Include associated semantic meaning
  - Allows for checking that information is consistent and complete
  - Allows for automatic transformation into OBSW, test harnesses, databases, ICDs, etc.

CCSDS SOIS Reference Communications Architecture



SCISYS

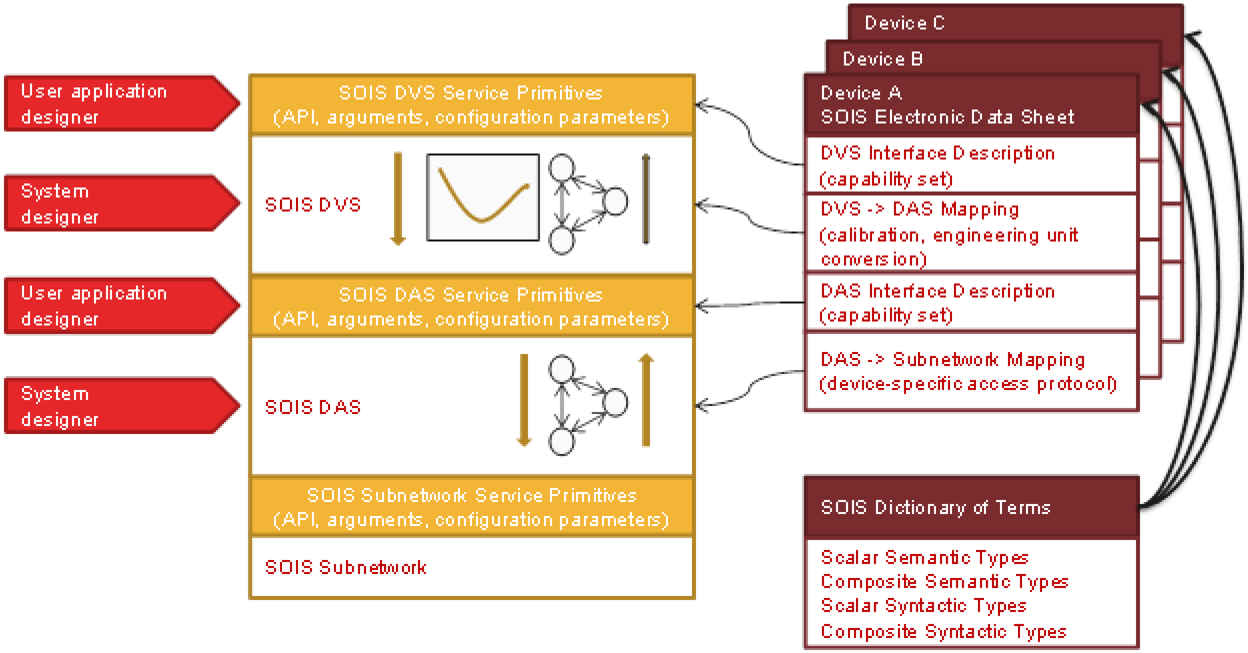
SOIS Command and Data Acquisition Services



- Generic Functional Interface
  - Functionality common to a device type
- Device Abstraction Control Procedure
  - How the Functional Interface is mapped onto the device-specific access protocols
  - Type conversions, operations, state-machine
- Device-specific Access Protocol
  - How to command and acquire raw data for specific devices using subnetwork-specific protocols, e.g. packet structures
  - State machine
- Subnetwork-specific Protocol
  - How to transfer data to/from device across subnetwork
  - QoS: ack, retransmit, priority etc.

SCISYS

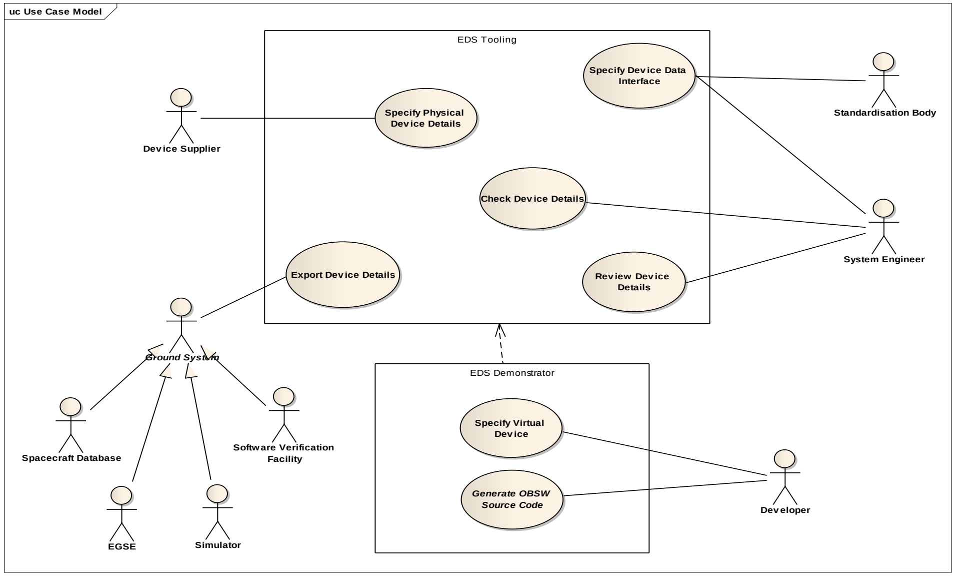
SOIS and the Usage of Electronic Data Sheets



EDS required to describe:

- DVS and DAS identifiers for all types/classes of devices;
- DVS interface and its semantics;
- DAS interface (if being used then the semantics of DAS has to be known by the DAS user);
- Interfacing between DVS, DAS and SOIS subnetwork services;
- Different mapping between DVS and DAS, and between DAS and subnetwork services.
- Conversion functions of “raw” data to engineering units, calibration curves, etc.
- Protocols in order to communicate with the device; meaning messages exchanged between DVS/DAS and the device using SOIS subnetwork services

Use Cases



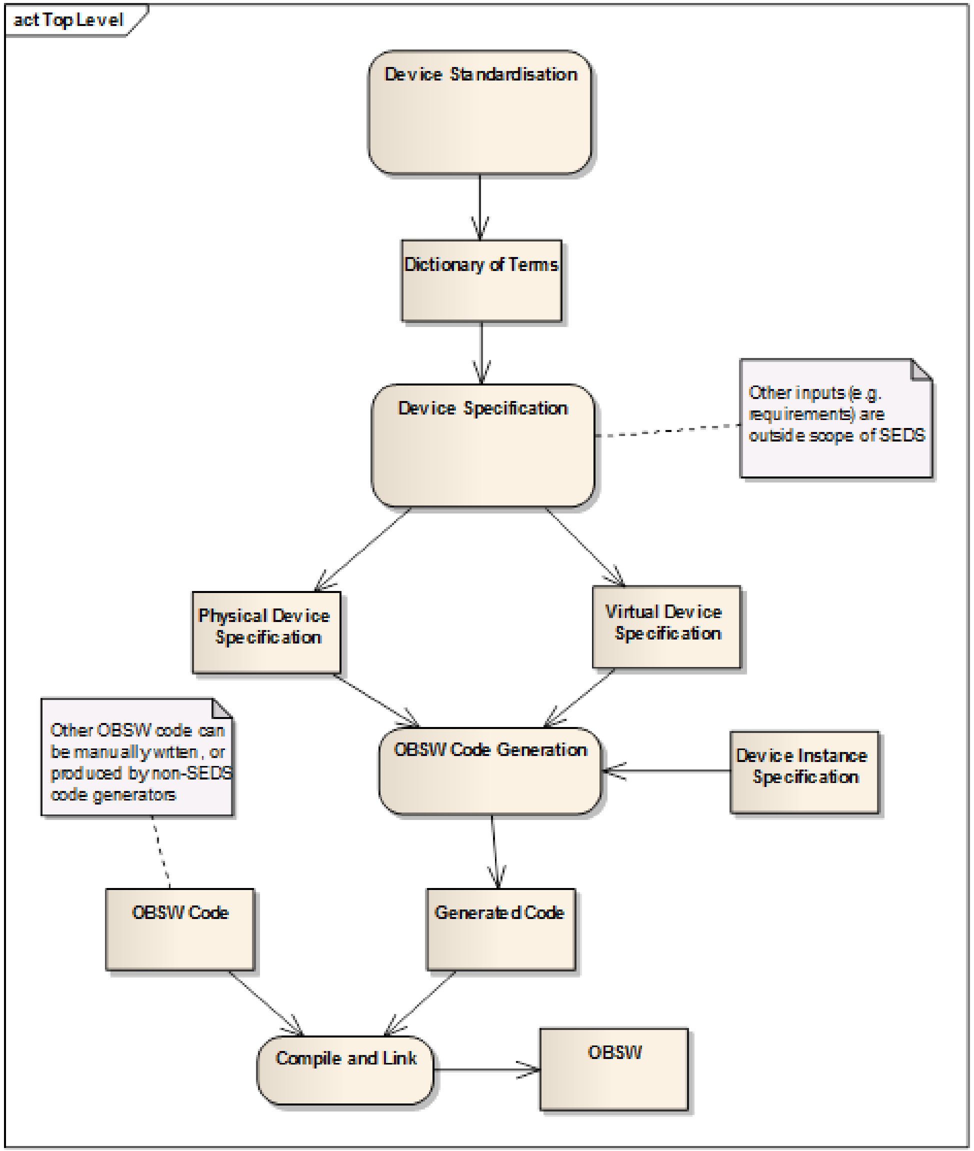
Use Cases to be addressed in project:

- Specify Device Functional Interface. A standardisation body or system engineer defines data interface to device in abstract terms.
- Specify Physical Device Details. The equipment supplier prepares the XML document describing the device.
- Specify Virtual Device Details. An OBSW developer specifies the mapping between the functional interface of the virtual device and that of underlying physical one.
- Check Device Details. A system engineer applies automated checks on the supplied information for correctness, completeness and consistency.
- Review Device Details. A system engineer reviews the device information in a static, readable form comparable to the current paper equipment ICDs.
- Export Device Details. Non-resource-constrained ground segment applications can simply read and process exported SEDS data.
- Generate OBSW Source Code. For resource-constrained onboard applications, OBSW developer will generate code from SEDS data to be integrated into OBSW

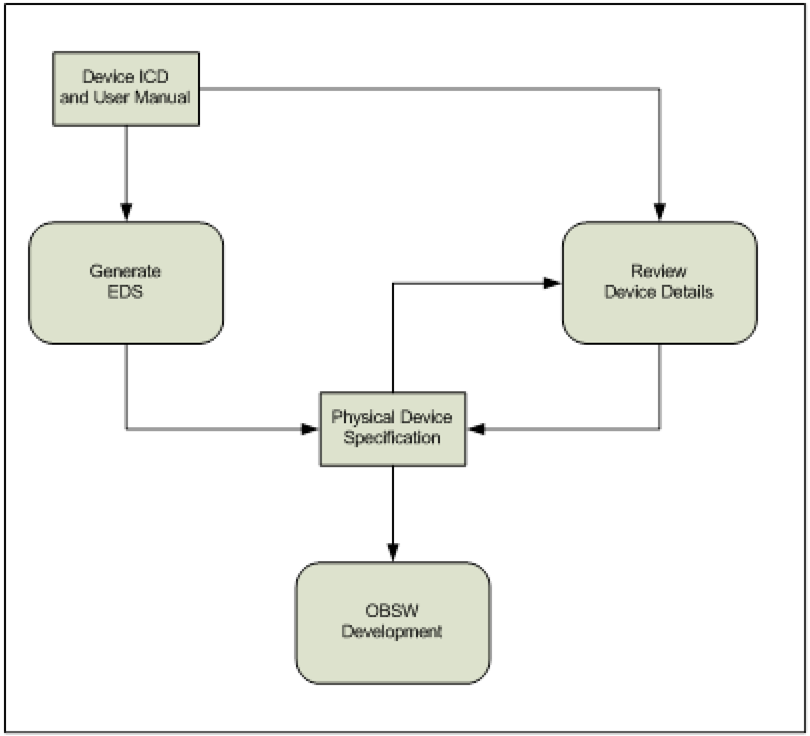
Additional Identified Use Cases out of scope of project:

- Import device model into Onboard Software Reference Architecture (OSRA).
- Exchange format with MCS Data Models, S/C Dbs and Sys. Eng. Dbs, containing device interface info for system, simulators, AIT and ops engineering data
- Extension to Specify Virtual Device Details to support Device Modes.
- Support for RS422 and discrete I/O devices

General-Purpose Device & OBSW Dev. Process Model

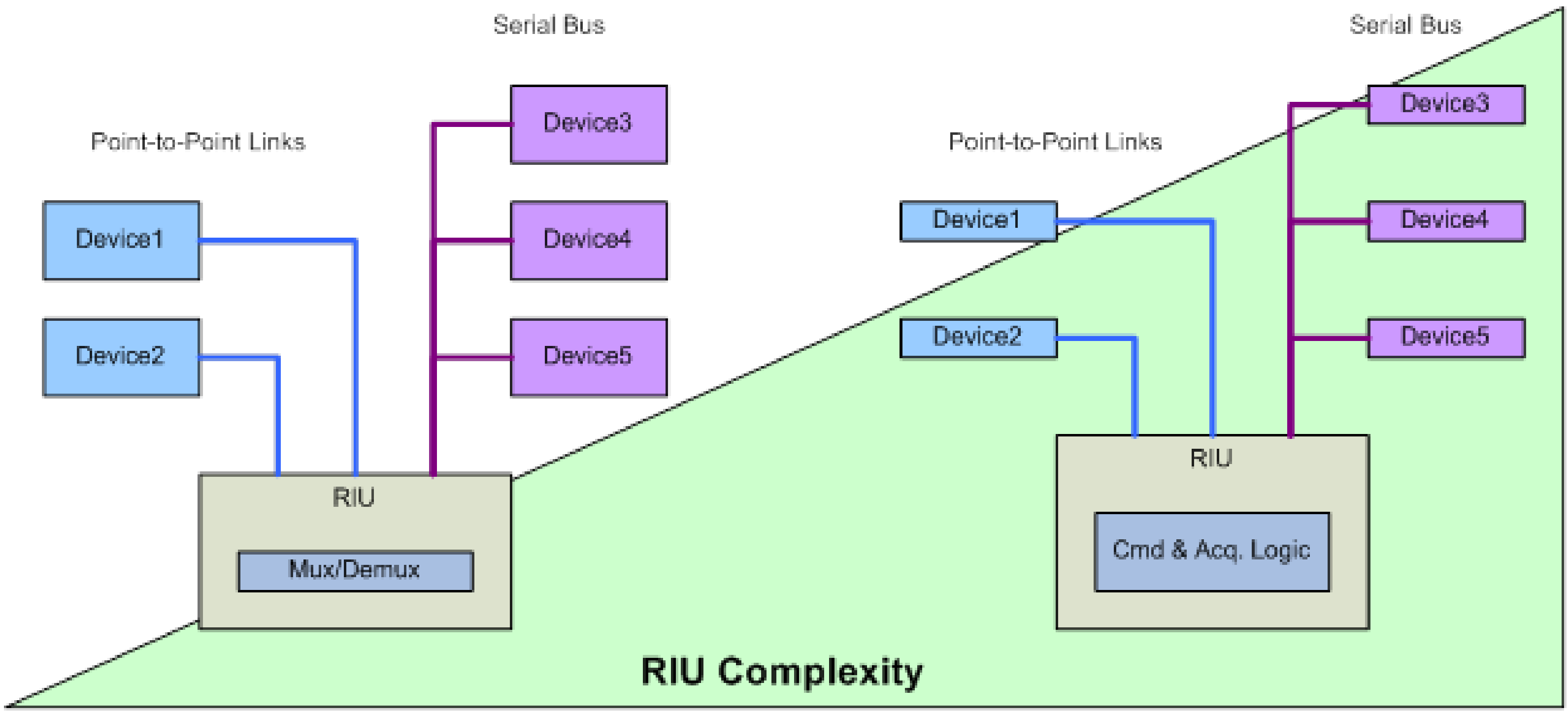


Devices Supplied without a SEDS



- During early adoption of SEDS, likely H/W manufacturers will be unable to provide SEDS for a device
  - » Traditional ICD and user manual supplied by the hardware manufacturer
- So team different from H/W manufacturer will create SEDS for the device
  - » May use electronic information such as TM/TC spreadsheets or device design models as inputs for bespoke tools that perform, maybe only partial, automatic generation of SEDS for device.
- Resultant SEDS validated by H/W manufacturer
  - » Perhaps using automatically generated documentation
- With appropriate toolchain, H/W manufacturer can maintaining SEDS for device through lifecycle in mission development.

Remote Terminal Units



- Tunnelling RIU: PM manages each device separately and is aware of the access protocol and data formats for each device
  - » RIU is just mux. and uses a mux. protocol to access each device at the subnetwork level using an RTU Channel selector. Each device still has its own DAP but is encapsulated into RIU mux. protocol.
  - » Each device described by separate EDS. RTU has separate EDS for its specific functionality. EDS for device accessed via RIU constructed from device & RIU EDSs, and RIU access channel
  - » Each RIU-accessed Device EDS would typically be mission specific (unless the same combination of RIU, device, and RIU access channel are used upon multiple missions), while the original device EDS would be generic.
- Smart RIU: RIU autonomously accesses devices and implements DAP. The PM is only aware of RIU DAP
  - » Smart RIU manages device access itself, so DAP only between the RIU and devices
  - » Device commanding and data accesses are provided using Value IDs mapping to the individual devices.
  - » Smart RIU likely built specifically for each mission, as dependent upon DAPs of different devices connected for each mission.
  - » As with Tunnelling RIU, each device described by a separate EDS. Specific EDS for Smart RIU

Selection of Equipment ICDs

- Set of sample AOCS equipment ICDs was taken and:
  - Assessed what would be required to represent the ICD's content
  - Confirmed that the ICDs contain sufficient information to support code generation of DAS and DVS
  - Selected which ICDs will be used for the SEDS Demonstrator
- Ideally equipment would be identified that covered the following classifications:
  - Simple/dumb and complex/intelligent equipment.
  - Actuators, sensors and RTUs.
  - SpaceWire and MIL-STD-1553B subnetwork types.

- 3 physical device ICDs were determined to be suitable for use in SEDS Demonstrator
  - FOG Astrix 120 Gyro
  - Sodern Hydra Star Tracker
  - NPAL Camera
- Covered categories of SpaceWire/MIL-STD-1553B, and simple/complex sensor, not actuator
- Together total 50 commands and 32 sets of telemetry data, sufficient size to provide realistically sized data sheets

- Generic functional interfaces specified by SAVOIR-SAFI for reaction wheels and star trackers, for use in data sheet of Hydra star tracker
  - Gyro generic functional interface not complete

XML Technology Survey

- Aim to reuse existing technologies where good match for EDS requirements
  - Focussing upon supporting XML syntax
- Survey Aims and Approach
  - Identify existing XML schemas that may be used
  - Reasons for looking for reuse:
    - Existing schema likely to be more mature and better explored than could be easily achieved in short timescale for development of new schema
    - May give compatibility with existing devices by permitting device information captured according to existing schema to be used directly in a SOIS context
    - May be possible to leverage existing software tools for creating, parsing and processing data represented according to existing schema
  - Types of re-use:
    - Syntactic re-use; where syntax and semantics defined by existing schema are re-used for SEDS schema
    - Semantic re-use; syntax of SEDS schema differ to that of source technology but underlying semantics are the same
  - Also possible to re-use subset of or extend existing schema
- Assessment Criteria
  - Physical Phenomena; Use of standard units (S.I.) and wider ontology for describing semantics
  - High-Level Functions
    - Interface description based on attributes and operations, representation independent data types, mathematical expressions, conditional stateless expressions, stateful expressions
  - Communications Functions
    - Interface description based on acquisition and commanding, machine representative data types, packing/unpacking data types, conditional stateless expressions, stateful expressions, timing information, CRCs/checksums/etc
  - I/O Mechanisms; subnetwork-specific configuration
  - Representations; XML schema
  - Documentation; textual and diagrammatic documentation
  - Additional Information; stable/community support/likely future viability, applicability to space domain

Technologies Assessed

Device oriented technologies:

- IEEE 1451
- ❖ AIAA SPA xTEDS
- CIA CANopen
- ✓ OGC SensorML
- OGC TransducerML
- ✓ XidML

- ✓ [candidate for re-use]
- ❖ [candidate for partial re-use]

Supporting technologies:

- ❖ OMG XMI
- ✓ OMG QUDV
- ISO/IEC/ITU ASN.1
- ✓ OMG/CCSDS XTCE
- ✓ CCSDS EAST
- ✓ CCSDS DEDSL
- CCSDS MAL
- ✓ W3C MathML
- ❖ ITU SDL
- W3C XEXPR

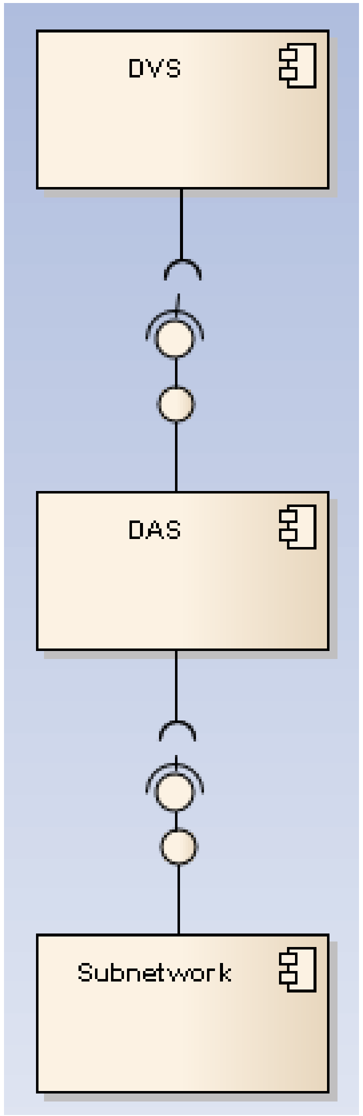
Chosen Approach to constructing SEDS XML Schema

- Bottom-Up Approach
  - » Build suitable framework around direct re-use of supporting technologies
    - › XTCE, EAST with XML syntax, DEDSL, QUDV, MathML
  - » Also considered for at least design purposes
    - › XidML, XMI for UML state machines and activity diagrams for protocol descriptions
  - » Less likely to be fragile to incomplete or incorrect requirements
- Top-Down Approach
  - » Select single promising device-oriented technology upon which to base SEDS schema
    - › SensorML and XTCE
  - » Difficulty is that challenges of adapting existing device-oriented standard for use in SOIS likely to be concentrated in the detail, such as the limitations of expression in particular cases, the underlying types and conceptual assumptions
  - » If requirements necessary and complete, can select technology with confidence
- Drawing on experience of XTCE, bottom-up approach was selected
  - » Existing schema selected for capture of semantics: RDF/OWL with QUDV

Overview of SEDS XML Schema

- Start with familiar technologies for the industry
  - Re-use from XTCE: design principles and data types
  - Re-use from UML: state machine semantics
  - Data sheet construction semantics close to those for UML class diagrams
  - Re-use RDF/OWL for semantics specification (ontology)
- Design schema so that re-use of definitions is easy
  - Interfaces and data types (e.g. for standardisation)
  - Semantic terms
  - Protocols
  - Algorithms/procedures
- Base schema on a minimal set of constructs used for everything

High-Level View: An EDS is Component-Based



- DVS and DAS are components
- Components are instances of component types
- A component type has
  - Provided interfaces (one or more)
  - Required interfaces (one or more)
  - An implementation (one)
- The subnetwork is effectively a component with provided interfaces only
  - No implementation in data sheet as standardised
  - In data sheet just another component, but understood by tooling
- Interfaces are instances of an interface type
  - Interface types could be standardised
- Interface types use data types
  - Also could be standardised
- EDS is strongly typed

SCISYS

Type Systems: Data, Interface and Component Types

- Data types are based on a few basic types
  - Scalar types e.g. Integer, Float, Boolean, String
  - Arrays (including variable length with upper bound)
  - Containers (like a variant record)
- Data types may optionally specify an encoding (serialisation) and semantics
- Data type inheritance is supported, container inheritance can be conditional

- Interface type defines an interface in terms of
  - Parameters (i.e. attributes)
    - Can be read-only
    - Can be marked as "async" where values are "pushed" rather than "pulled"
  - Commands (i.e. operations)
    - Can have zero or more arguments with in/out/inout modes
- Interface types can be generic (on data types)
- Interface type inheritance is supported

- Component types define
  - Component required and provided interface in terms of interface types
  - Implementation in terms of state machines and activities
- Component types can be generic
- Component type inheritance is not supported

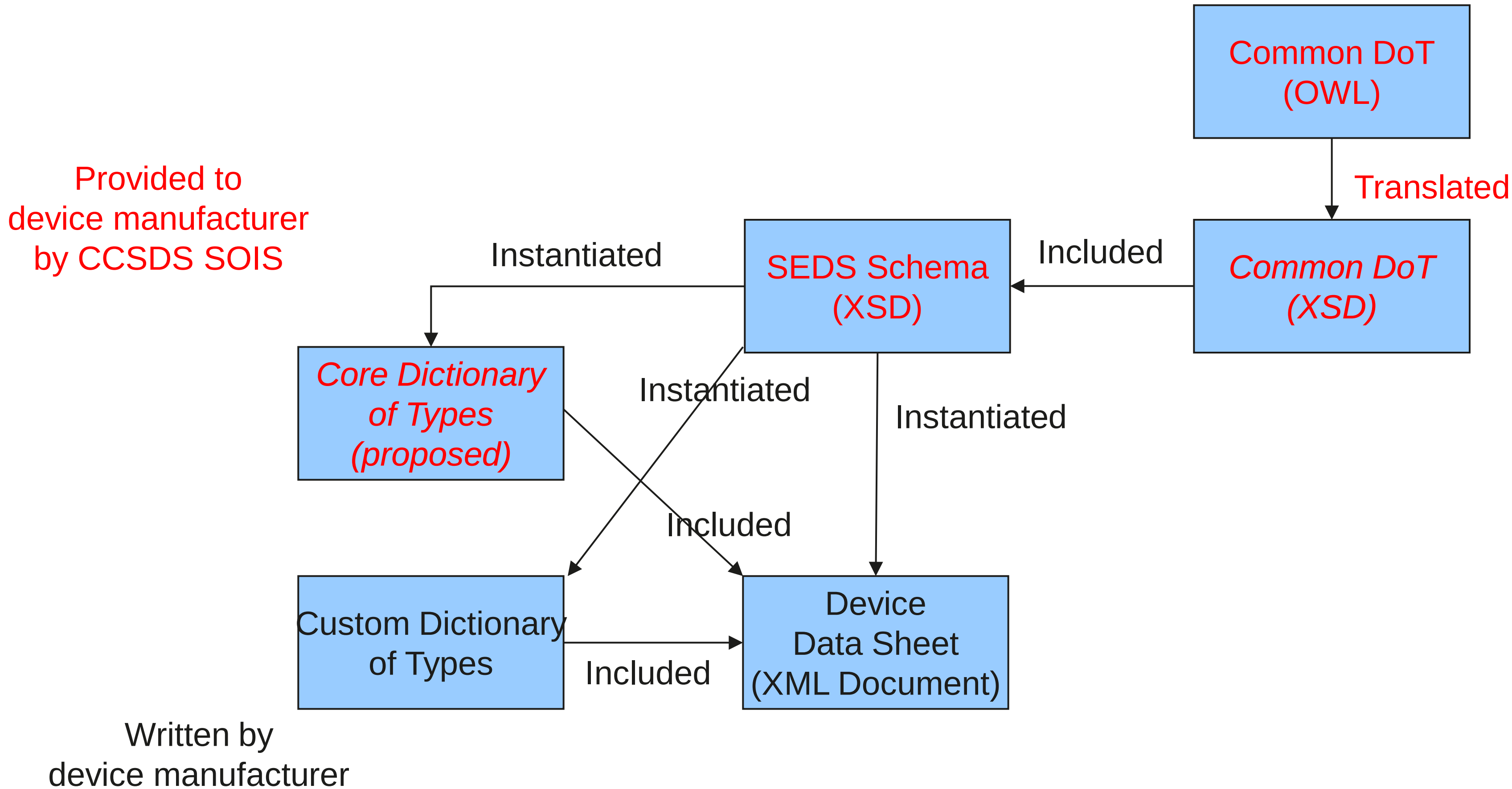
Component Implementation (Behaviour) Specification

- The “implementation” of a component type specifies its behaviour
- Stateful description based on
  - State machines for event-driven state transition description
  - Events are the exchange of service primitives as described by SOIS
  - Activities to describe stateless procedural tasks for states
- Stateless description is a mapping e.g. a calibration
  - Same semantics as a single activity
- Multiple state machines describe parallelism supported by the device
- Protocol description is straightforward
  - Container types with encoding information describe packets
  - State machines describe exchange patterns
- Can implement a component using sub-components
  - Supports protocol re-use, assisted by use of generics

Other Schema Features

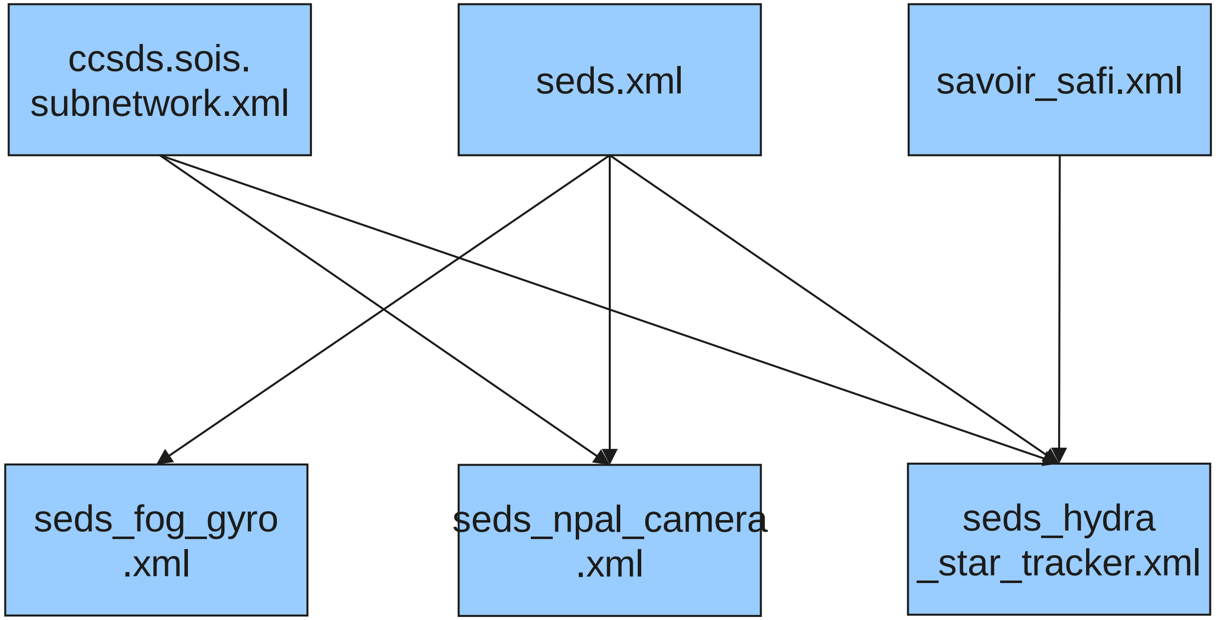
- All data sheet elements are organised into namespaces
  - Better support for standardisation and reuse
- Documentation facilities
  - Support for documenting all aspects of device interface and behaviour throughout data sheet (uses XHTML)
- Subnetwork constraints
  - Permissible device addresses, link speeds etc.
    - Only SpaceWire addressed in SEDS Schema but other subnetworks can be added, e.g. MIL-STD-1553B
- Non-functional data specification
  - Device data specified as typed constants with semantic tags
  - Can structure data using semantically tagged categories
- Custom ontology support

Standards, Ontologies, Schemas and Data Sheets



Normative elements indicated in red

Device Data Sheets for the SEDS Demonstrator



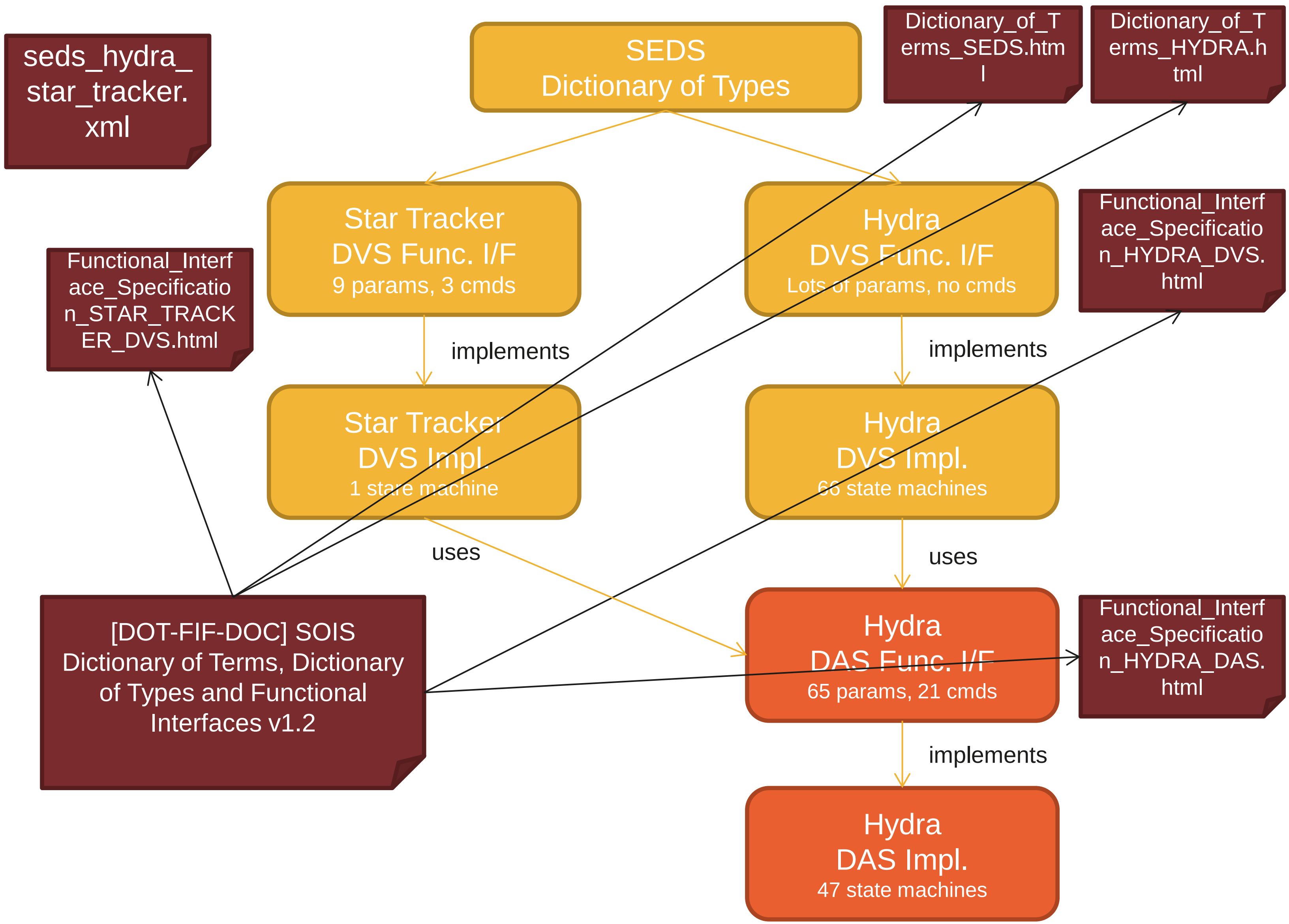
- seds.xml. Contains the Core Dictionary of Types, for use in device data sheets.
- ccsds.sois.subnetwork.xml. Defines the functional interfaces for the SOIS Subnetwork Packet and Memory Access Services, for reference in device data sheets.
- savoir_safi.xml. Defines the generic functional interfaces for Star Trackers and Reaction Wheels, as defined by the SAVOIR-SAFI working group.
- seds.fog_gyro.xml. Data sheet for the FOG Gyro, including seds.xml and ccsds.sois.subnetwork.xml.
- seds_hydra_star_tracker.xml. Data sheet for the Hydra Star Tracker, including seds.xml, ccsds.sois.subnetwork.xml and savoir_safi.xml.
- seds_npal_camera.xml. Data sheet for the NPAL Camera, including seds.xml and ccsds.sois.subnetwork.xml

Hydra Star Tracker Data Sheet

- PDF ICD + Excel Spreadsheet
  - 2,173,015 bytes = 2.07Mi bytes
  - 92 pages
  - 25 TCs, 9 TM Packets, 6425 Parameters
- Data Sheet seds_hydra_star_tracker.xml
  - 24,484 XML input lines, 1,761,787 bytes = 1.68Mi bytes
    - 1st Draft was 29,397 XML input lines, 2,018,033 bytes = 1971Ki bytes = 1.9Mi bytes
  - Lots of parameter type definitions!
  - Interfaces
    - Device-specific Hydra DAS Interface
      - 65 parameters, 21 commands
    - Device-specific Hydra DVS Interface
      - Lots of parameters, no commands
  - Hydra DAS (DAP) component
    - Provides Device-specific Hydra DAS Interface
    - Uses Subnetwork MAS Interface
    - Lots of parameter type definitions!
    - 108 parameters
    - 86 activities
    - 66 state machines
      - NO_REPLY: 1 state, 22 transitions
      - 65 x READ_VALUE: 2 states, 2 transitions (to/from IDLE)
  - Hydra DVS (DACP) component
    - Provides Device-specific Hydra DVS Interface
    - Uses Device-specific Hydra DAS Interface
    - Simple mapping from Hydra DAS interface
  - SAVOIR-SAFI Hydra DVS (DACP) component
    - Provides SAVOIR-SAFI ATTITUDE, ANGULAR_RATE and HOUSEKEEPING DVS Interfaces
    - Uses Device-specific Hydra DAS Interface
    - 6 parameter type definitions
    - 7 parameters
    - 6 activities
    - 1 state machine
      - 4 states (IDLE, RESET, UPLOAD_DATE, SET_STR_MODE), each of which has two transitions (to/from IDLE)

SCISYS

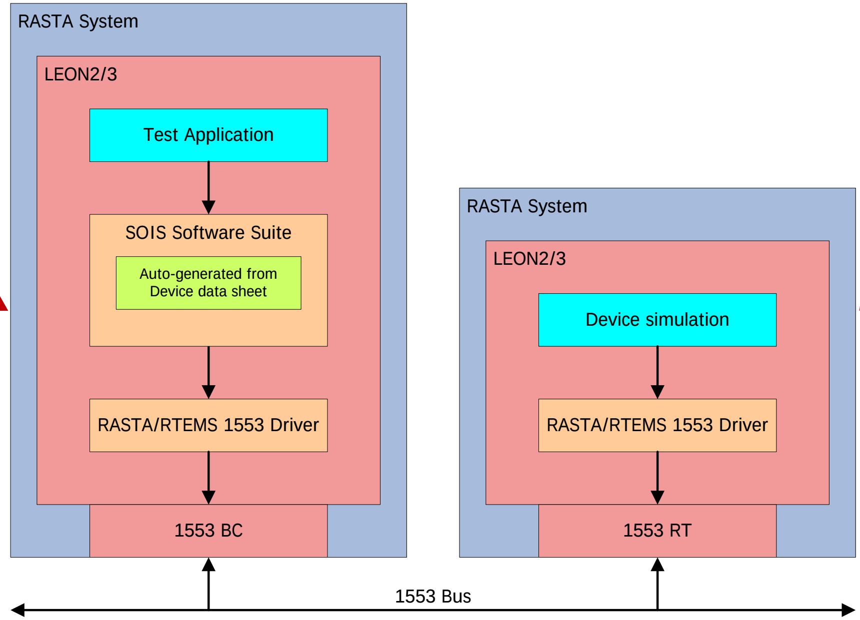
Contents of a Device Data Sheet: Hydra Star Tracker



SCISYS

Demonstrator – Auto-generation of OBSW from EDS

RASTA representing OBC
SOIS Software Suite from SOIS PoC project containing *auto-generated code from device data sheets*
E.g.
- Interfaces
- State machines



RASTA representing simulated Hydra Star Tracker
Hand-coded from paper ICD

- Demonstrate that device data sheets can be used to automatically generate OBSW
- Proof of concept code generator toolset for SOIS Software Suite framework from SOIS Proof of Concept project
- Test Applications call DVS or DAS to command and acquire data from e.g. simulated Hydra Star Tracker
- Successful demonstration

SCISYS

Demonstrator – Toolset

- Objective of toolset is to validate that code and documents can be autogenerated from device data sheets and so can only be considered functional prototypes

- *Codegen*: DAS and DVS code generator
  - Takes a datasheet and generates corresponding set of interfaces and implementations
- *Docgen*: DoT and ICD document generator
  - Takes a datasheet and generates corresponding Dictionary of Terms and ICD documents in HTML

- Support libraries
  - *scisys.eds.seds*: API for reading and validating datasheets
  - *scisys.eds.seds.generator*: API for creating or importing datasheets

- Device Data Sheet generation
  - XML tools still available but better to use tools, due to necessary complexity of schema and device datasheets, e.g. spreadsheet to autogenerate data sheet
  - *NpalCameraTest*: test case that uses generator API with hard-coded data to specifiy NPAL DAS and DVS
  - *StarTrackerTest*: test case that uses generator API and data read from a spreadsheet to specifiy HYDRA DAS and DVS
  - *GyroTest*: test case that uses generator API to specifiy FOG DAS and DVS

Demonstrator – Results

Builds	Text (bytes)	Data (bytes)	Bss (bytes)	Total (bytes)
NPAL Initiator	639,712	40,788	173,766	854,264
Hydra Initiator	779,200	40,804	176,996	997,000
FOG Initiator	593,072	40,788	152,965	786,824
NPAL Target	540,128	12,756	183,732	736,616
Hydra Target	172,128	35,044	12,724	219,896
FOG Target	172,144	35,028	12,164	219,336

- SOIS Software Suite contains software implementation of RMAP & bespoke PTP
- No optimisation of RTEMS and C library functionality included
- Objectives did not include any improvements to automatically-generated code sizes

Technical and Process Issues Identified

- Balance readability of data sheets with sufficient complexity to capture all sensible patterns
  - » XML is unreadable in all but most simple cases
  - » Most simple cases are too simple to test schema is sufficiently rich enough
  - » Viewer, editor support tools required (beyond standard XML)
- Need to explore multiple use cases to iron out issues
  - » Coverage of all device classes and sufficient examples
  - » Test interfacing to real rather than simulated devices
  - » Different uses, different processes
- Handling access to legacy devices
  - » Little or no implementation of ECSS 1553 services by devices
  - » NPAL camera doesn’t use SpW protocol IDs
- Not enough standardisation of SpW protocols yet
  - » Need e.g. SpW-D & defined protocol stack

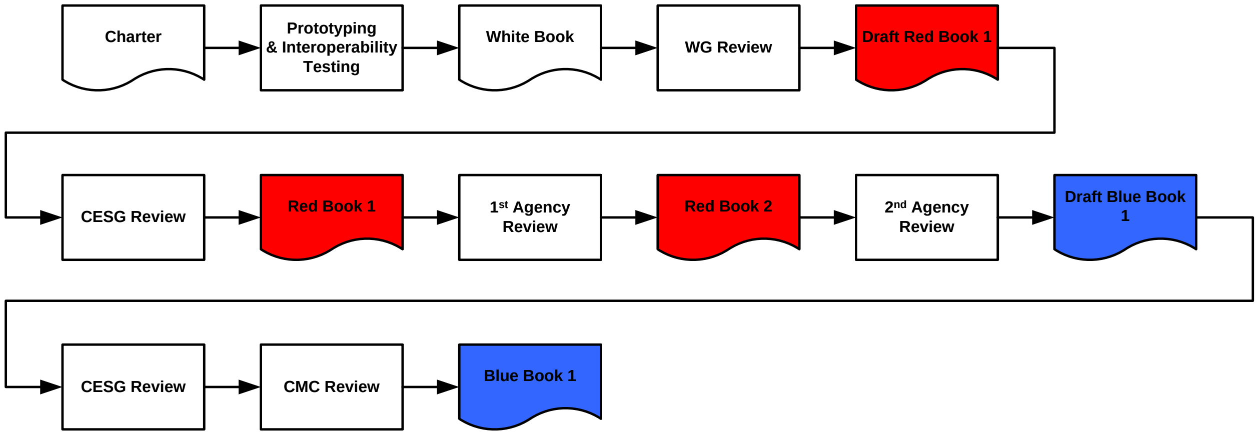
SCISYS

Recommended Future Directions for SEDS Demonstrator

- Further testing of SEDS Demonstrator within current scope (i.e. with existing device ICDs)
  - Incrementing sequence count for commands sent from NPAL Camera
  - Timeout transitions for acknowledgement failure for NPAL Camera
- Extend scope of SEDS Demonstrator (i.e. beyond existing device ICDs) by adding:
  - More types of mathematical expressions, Variable-length arrays, One’s- complement encoding, MIL-STD-1553B float format, etc.
  - Synchronisation and other SOIS Subnetwork Services
  - Additional device ICDs
- Improvements to SEDS Tooling
  - Systematic coverage testing of SEDS features and options.
  - Negative testing to ensure always produces appropriate error message on invalid input.
  - Develop a domain-specific language (DSL)-based interactive editor with full context help and validation.
  - Improvements to formatting and contents of generated documentation.
- Changes to SOIS Proof of Concept software
  - Extend and test MAS mapping to MIL-STD-1553B and update ECSS 1553 Services
- Sizing and Performance
  - Further measurements of sizing and run-time performance of auto-generated software and SOIS Software Suite
  - Stress testing under high load.
  - Change Hydra Star Tracker data sheet to make more use of arrays
  - Define and implement sizing and performance optimisations
- Minor SEDS XML schema improvements
  - Specify dimension and scaling for units
  - Take semantic constraints and formalise them as either additional syntax, a constraint language, or with reference to a standardised tooling

Standardisation of EDS XML Schema by CCSDS

Supported by
- ESA
- UKSA
- NASA:
  - GSFC, JSC, SUMO, AIAA



- CCSDS Standards
  - » CCSDS 876.0 XML Specification for Electronic Data Sheets for Onboard Devices
    - › Blue Book, EDS XML schema
  - » CCSDS 876.1 Common Dictionaries of Terms & Types for Onboard Devices
    - › Blue Book, OWL ontology, Common Dictionary of Terms XML schema, Common Dictionary of Types XML document
  - » Items in Red managed and online access provided by CCSDS SANA
- Electronic Data Sheets informational report (Green Book)
  - » Overview of structure and expected usage of Electronic Data Sheets
    - › More detailed that general SOIS Informational Report Issue 2
- Standardisation of device class-specific generic Functional Interfaces
  - » In EDS format
  - » Extensible set of device classes, perhaps derived from SAVOIR-SAFI work
  - » *Which standards organisation should own this?*

SCISYS

Conclusions

The Adoption of Electronic Data Sheets and Device Virtualisation study has successfully:

- Defined use cases for using electronic data sheets to specify the functional interfaces of off-the-shelf and custom devices
- Defined, and refined through practical use, a prototype XML schema for SOIS electronic data sheets
- Produced data sheets for three real AOCS devices using the SEDS XML schema
- Produced proof of concept tooling to automatically generate ICD and SOIS DVS and DAS OBSW from the data sheets
- Demonstrated the use of the automatically generated OBSW to allow an onboard test application to command and acquire data from simulations of the three devices using the RASTA Test Facility across SpaceWire and MIL-STD-1553B buses
- Produced a candidate draft CCSDS standard and XML schema for SEDS

Thank you

Any questions?

EDS Schema Standardisation