Electronic Data Sheets (EDS) - Reboot

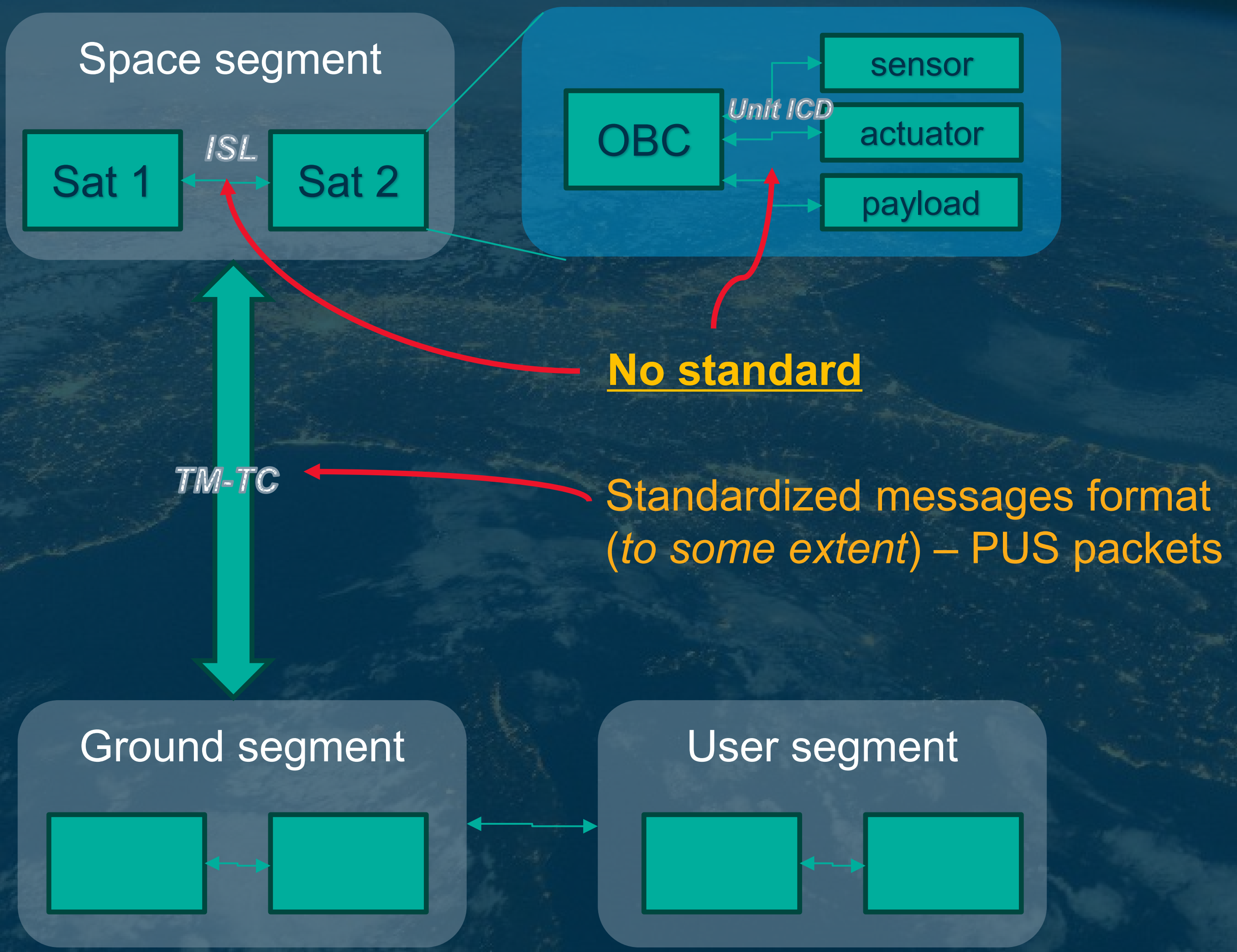
Maxime Perrotin

21/10/2024

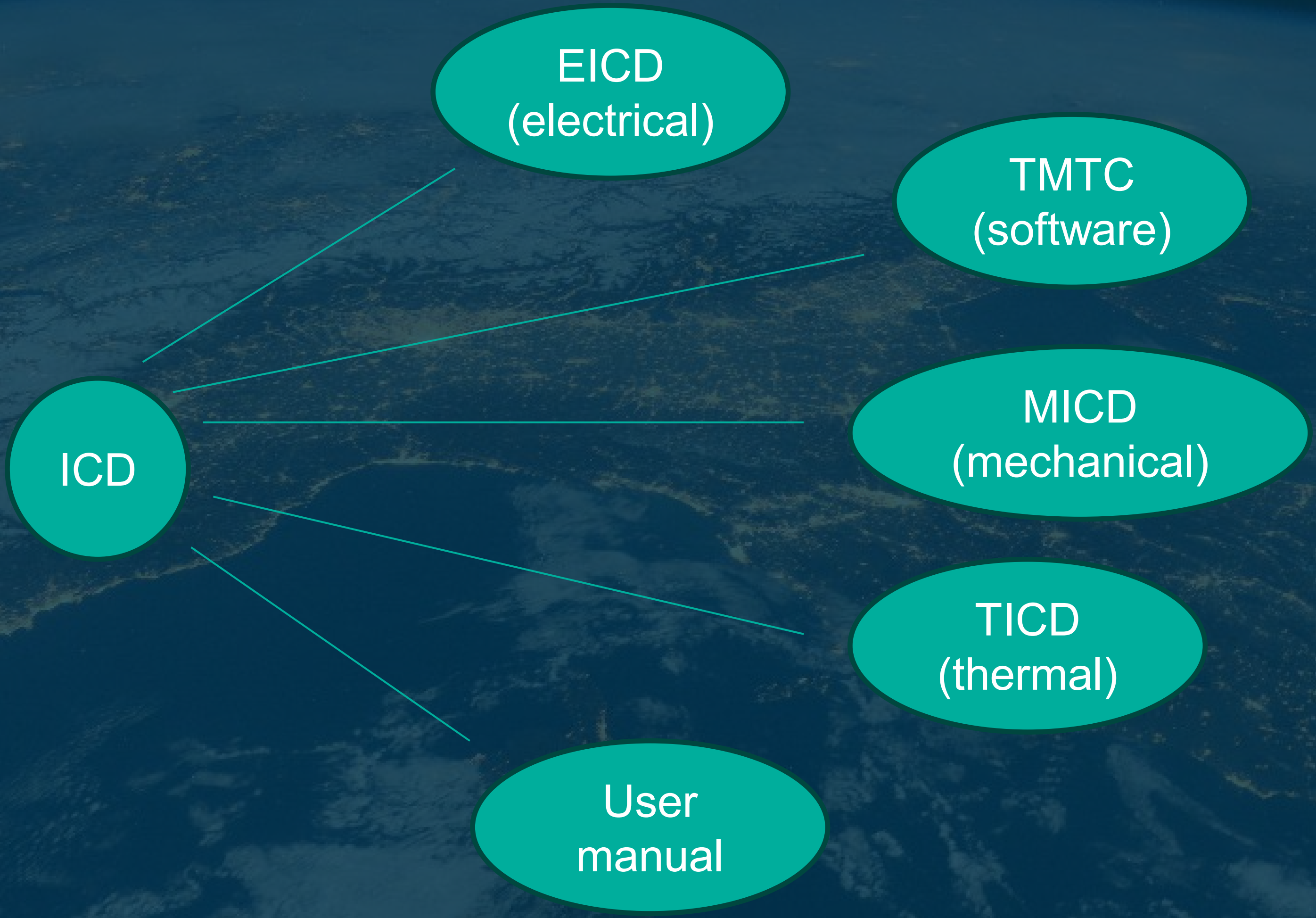
Electronic Data Sheet working group Status

1. What are EDS and who needs them ?
2. How is the problem addressed now ?
3. What is the working group planning to achieve?
4. Tooling, Use Cases and conclusion

EDS : Interface management in space systems



Interface management is multi-disciplinary



From ICD to EDS: what is the scope?

- An EDS is an electronic form of *parts of an* ICD
- To define the **scope** of EDS (machine processable data) we need to answer these two questions:
  - What information is (or *shall be*) contained in an ICD ?
  - Who are the users of the ICD (Use Cases) ?

ICD content – example (reaction wheel – not a PUS terminal)

- Connector (*half duplex RS485 for data)*
- Configuration (*115.2 kbps, 8n1)*
- Command frequency (*up to 8 Hz with reduced performance, 2Hz with full performance)*
- Protocol (*Each command is acknowledged by the device)*
- Redundancy (*The RW can be commanded by 2 hot redundant busses)*
- Restrictions (*User must enforce a delay of 100 us between a response and the next command)*
- Frame control characters (*Start/End marker, Escaping character, offending byte is XOR-ed with 0x20)*
- Byte stuffing (*counter and checksum computed prior to escaping)*
- CRC (*CCITT16)*
- Byte ordering for numbers (*big-endian)*
- Modes and constraints on the modes
- Frame structure (*headers)* and complete packet formats to the bit level
- Description of all error codes from telemetry packets

ICD content

But…

None of this is formalized in any ECSS standard

Actual ICDs are hand-written

- Sometimes it is complex and ambiguous

Table 5-1: Transmission frame

Frame	STX	ADR	ABS	COUNT		MESSAGE - BLOCK (MB)				CS		ETX
No. of bytes	1	1	1	COUNT (H)	COUNT (L)	M_1	M_2	. . .	M_{count}	CS (H)	CS (L)	1
Description	Start of Transmission (Startbyte) 0x7E	Address	Sender	Number of bytes in message block		Command and data bytes (MSB first LSB next)				Checksum CRC16 of STX to last byte of MB		End of Transmission (Endbyte) 0x0D

Command	Command ID Main	Sub	Parameter	Description	Response telegram
SET_MODE1_OMEGA *(not available in boot loader)*	0x6D	0x6A	OmegaSetPoint (signed long) resolution 0.000117186 rpm → e.g. *5120000* = *600 rpm*	This is a ***simple control mode*** for regulating rotation speed by using a controller with PI characteristic → maintaining rotation speed with fixed offset to set point	Section 5.3
SET_MODE3_OMEGA *(not available in boot loader)*	0x6D	0x6D	*Parameter 1:* OmegaSetPoint (signed long) resolution 0.000117186 rpm → e.g. *5120000* = *600 rpm*	This is a ***high precision* control mode** for regulating rotation speed by using a state controller → maintaining rotation speed with state estimation	Section 5.3
SET_MODE3_OMEGA_RES *(not available in boot loader)*	0x6D	0x77		Like SET_MODE3_OMEGA and additionally: current displacement will be set to null	Section 5.3
Extension of SET_MODE3_OMEGA SET_MODE3_OMEGA_RES	0x6D 0x6D	0x6D 0x77	*Parameter 2:* DeOmegaSetPoint (signed long) resolution 0.0009375 rpm/s → e.g. *96000* = *90 rpm/s*	Both are extended by second parameter acceleration. Hence the set rate will be reached by using the set acceleration. Notice: The sign of the acceleration is non-relevant and follows from the set rate. Notice: The length of command is 10 instead of 6.	Section 5.3
SET_MODE3_DEOMEGA *(not available in boot loader)*	0x6D	0x69	DeOmegaSetPoint (signed long)	This is a ***high precision* control mode** for regulating acceleration by using a state regulator → constant	Section 5.3

Same command ID (discriminant), hack in the description field

The case of PUS terminals

- An apparent standard representation of packets in chapter 8

packet primary header							packet data field	
packet version number	packet ID			packet sequence control		packet data length	packet secondary header	user data field
	packet type	secondary header flag	application process ID	sequence flags	packet sequence count or packet name			
3 bits	1 bit	1 bit	11 bits	2 bits	14 bits	16 bits	variable	variable
2 octets				2 octets		2 octets	1 to 65536 octets	

repeated N1 times

repeated N2 times

N1	CPDU ID	N2	output line ID	reserved	duration exponential value
unsigned integer	enumerated	unsigned integer	enumerated (12 bits)	bit-string (1 bit)	unsigned integer (3 bits)

optional

TC packet PUS version number	acknowledgement flags	message type ID		source ID	spare
		service type ID	message subtype ID		
enumerated (4 bits)	enumerated (4 bits)	enumerated (8 bits)	enumerated (8 bits)	enumerated (16 bits)	fixed-size bit-string

optional

NOTE The spare field is used to constrain the length of the telecommand packet secondary header to an integral number of words. Its optional presence of is driven by requirement 7.4.4.1g.

PUS scope: not an ICD

- The PUS does not contain requirements for an actual ICD
- Packet descriptions are generic, and some fields are tailored out
- No machine processable standard format for TM and TC used in ICDs (only a document)

- PUS even introduces confusion and lacks important requirements:
    - Bit naming is a requirement (bit 0 is not 2^0 : it is the bit on the left !)
    - Endianness is not specified

Interface Management in ECSS E-10-24C

The missing piece in the puzzle?

Defines the expected content of ICDs (including DRDs)

But the current version lacks detailed requirements


ECSS-E-ST-10-24C
1 June 2015

EUROPEAN COOPERATION
ECSS
FOR SPACE STANDARDIZATION

Space engineering

Interface management

ECSS Secretariat
ESA-ESTEC
Requirements & Standards Division
Noordwijk, The Netherlands

ECSS-E-10-24C Improvement: our work in the EDS WG

- Scope : EICD and TM-TC ICDs
- Ensure that the ECSS standard is complete and unambiguous

E CSS

ECSS-E-ST-10-24C Rev.1 DIR1
19 March 2024

E-ICD Status: Done (led by Ferdinando Tonicello)

public review closed

5.8 EICD requirements

5.8.1 Space segment equipment

a. The equipment EICD shall contain the detailed circuit diagram of the space segment equipment including component values or provide the reference to the applicable document containing such information.

b. The equipment EICD shall contain the detailed schematics of all electrical interfaces including the part types, the parts references and the part values.

NOTE Detailed schematics are also called electrical diagrams.

c. The equipment EICD shall include the relevant grounding diagram or provide the reference to the applicable document containing such information.

d. The equipment EICD shall include the grounding implementation for each electrical interface type.

e. The equipment EICD shall include the lists of all electrical Interfaces of the equipment classified per signal type, electrical connector list and types (including supplier) and the equipment electrical connectors pin-to pin table.

NOTE It is necessary to produce such lists and tables respecting a specific format for supporting the harness design (electrical interconnections with other equipment) at a higher level.

Ongoing work in the EDS working group

TM-TC ICD

5.9 TM and TC ICD requirements

TM and TC requirements can be found in ECSS-E-ST-40 (Annex E, ICD DRD) and in ECSS-E-ST-70-41.

no

5.10 TICD requirements

Requirements for the TICD are contained in the DRD Annex D of ECSS-E-ST-31.

5.11 MICD requirements

At the moment there are no detailed requirements for mechanical ICD available in ECSS.

Use cases: who needs ICDs?

…and who needs **electronic** ICDs ?

Example: Software engineers need to write unit device drivers and need to process

- Hardware access
- Unit startup
- Communication via packets
- Event and reconfiguration scheme

- Once we have a detailed specification for an ICD and use cases, we can decide what parts require an electronic representation and what format to use

Next Steps

- Work on use cases and create ICDs that comply to the new requirements
  - Candidates : ADHA (Advanced Data Handling Architecture), APA (Advanced Power Architecture)
- Determine which pieces of the ICDs really need an EDS form
- Work on tooling and concrete notations
  - Goal: generate ICDs and EDS that are compliant to ECSS standards by construction
  - Don’t reinvent the wheel, reuse languages and tools
    - SOIS EDS
    - ASN.1
    - SDL

Electronic Data Sheet working group Status

1. What are EDS and who needs them ?
2. How is the problem addressed now ?
3. What is the working group planning to achieve?
4. Tooling, Use Cases and conclusion