Standardisation of PF/PL Interfaces and Deployment of common Building Blocks

ADCSS 2014

Noordwijk, 29.10.2014

R.Roques,
Airbus Defence and Space
Earth Observation, Navigation & Science

Mission types vs SAVOIR ASRA P/L interface classes

➔ SpaceWire P/L network

Airbus DS Missions	ASRA class		
	Direct (PF RIU)	PLIU	PLMU (ICU)
Commercial optical EO		SPOT6/7/…	
Commercial Radar EO			TandemX, Future
Institutional operational EO		Sentinel-2, SEOSAT	Sentinel-1, Sentinel-5P, MetOp SG (9)
Institutional exploratory EO	SWARM	Aeolus	EarthCare (4), Merlin
Science	GAIA		LISA PF
Science / EO (secondary payloads)	SEOSAT		
Deep Space			BEPI (9), Solar Orbiter (10)

➔ Efficiency of PLMU (ICU HW+SW) development and integration is key for Primes

Presentation contents

In on-board data handling and software area:

- Some lessons learned from payload development projects
- Which Platform/Payload Interface functions should be standardised / made common ?
- Which level of standard building block to deploy to support standardisation ?
- MetOp SG as an example
- Future enabling technologies

Some lessons learned from P/L development projects

At Airbus DS, regarding Platform/Payload Interface and
ICU standardisation /commonality initiatives in ESA projects :

- Start initiatives early to reach really mature specs at ICU ITTs
- Spacecraft Prime to be active. Instrument contractors to master their end-to-end process. But just relying on bottom up commonality is not enough.
- ICU software is complex. Special attention to be paid when it is developed by a 2nd tier supplier. Prime to set up risk mitigation strategies.
- Common building blocks to come with the relevant validation information to avoid re-validation by all users and hence, waste of effort
- Harmonise database structure, naming. Avoid, as far as possible, data base tool conversions between stakeholders
- Flexibility for instrument mode definition is needed. Mode rationalisation to be limited.

Standardisation potential of ICU functions

Function	Possible common cross-mission solution (at least at S/C product line lvl)	Possible common solution to all instruments of a mission	Instrument specific
PF I/F discrete links (SOL, On, Off…)	✓	✓	
Platform Interface data link	✓	✓	
Boot, Service Mode	✓	✓	
Time reference (S9)	✓	✓	
Instrument internal sync. signals		✓(envelope)	✓
Instrument internal 2nd tier data links		✓(envelope)	
Thermal control: operations/control laws Therm/heaters i/f		? (envelope)	✓
Operational Modes		? (basics)	
PUS librairies:			
S1 : TC S3 : Housekeeping S5 : Event reporting S6 : Memory management S12 : O/B Parameter monitoring S17 : test S19 : Event / action S148 : O/B Macro-procedures	✓	✓	
S140 : Parameter mgt		?	✓
S2 : Device cmd distib S142 : Functional monitoring			✓
Instrument mission TM	?	✓	
Instrument DB structure, naming	?	✓	

Standard functions may go
beyond ICU HW + HW dependent SW and
encompass application layer services

Levels of possible standard ICU building blocks

Within a project :

1. Key interoperability concepts detailed in applicable doc.
   - Technical definition of functions, links, protocols, modes…
2. Common ICU
   - Configurable equipment family with, typically, a core section, a variable I/O section and software
3. Common ICU processor module
   - incl. hardware dependent software
4. Common On-board data management software
   - « execution platform », incl. PUS services
   - Software development environment
5. Common ICU processor module design
   - Distribution to ICU contractors (schematics, layout, VHDL…)

Criteria for standard building blocks strategy selection

- Benefits
  - Spacecraft flight operations simplification
  - Spacecraft AIT/ground operations simplification
  - Engineering effort simplification at Prime
  - Instrument development risk reduction
  - Instrument development cost reduction

- Other aspects
  - Adequation/flexibility wrt geo distribution
  - Compatibility with usual ICU supplier industrial capabilities
  - Built-in flexibility to cope with Instrument design evolutions during development phase
  - Capability to define an « envelope » without overdesign

Comparison of commonality strategies

	1	2	3	4	5
	Key interop. Concepts	Common ICU	Common Proc. Module	Common Core Software	Common ICU PM HW design
Simpler Spacecraft flight ops	+++	+++	O	+++	O
Simpler Spacecraft AIT/ground ops	+	+++	O	+++	O
Simpler Engineering effort at Prime	+	+++	++	+++	+
Lower Instrument development risk	O	+++	++	++	+
Lower Instrument development cost	O	+++	+	++	O
Shorter Instrument delivery schedule	O	+	++	++	O
Adequation, flexibility wrt geo distr.	+++	-	+	+	+++
Adequation with supplier delivery capabilities	+++	-	++	++	+++
Flexibility wrt Instrument design changes	+++	-	+++	+++	+++
Envelope solution without overdesign	+++	-	++	++	++

- Deployment of a common processor module with relevant core software (3+4) provides benefits to Prime and Instrument Contractors

MetOp SG Instruments as an example

Instrument	ICI	MWI	MWS	SCA	RO	3MI
Satellite	B	B	A	B	A&B	A
ICU Hardware						
Processor	☐	☐	☐	☐	X	☐
Power & heaters	☐	☐	X	☐	X	☐
Standard I/O	☐	☐	X	☐	X	☐
Specific I/O & pwr			X		X	X
Specific Proc.				X	X	
ICU Software						
Cmd & Ctl application	X	X	X	X	X	X
Science data proc.						X
Execution Platform (incl. PUS)	☐	☐	☐	☐	X	☐
Low Level (HW dependent)	☐	☐	☐	☐	X	☐
Possible commonality approach						
(1) Interface & Ops doc level	☐	☐	☐	☐	☐	☐
(2) ICU level	☐	☐		☐	X	☐
(3) PM HW module + HW dep SW	☐	☐	☐	☐	X	☐
(4) PIM core Data Mgt software	☐	☐	☐	☐	X	☐

common function possible

specific function, no commonality

- Deployment of a common PIM can be offered to 5 out of 6 instruments
- Common ICU Equipment can be considered for up to 4 instruments

Common Platform Interface Module (PIM) – 1/2

Microprocessor	LEON 2/3 approx. 60 MHz
Volatile memory	typ. 512 MB SDRAM
Non volatile	> 8 MB FLASH or EEPROM
Internal SpaceWire	2 links on backplane
External SpaceWire	2 links on external connectors
Other data links	2 UART @ 1 Mbps
Sync	Configurable Sync generator
Mass	< 1kg (single PCB)
Power	7 W typ.
Implementation	typ.: 1 SoC + 1 FPGA + upscreened memory components + i/f

PIM « package » for MetOp SG

- PIM hardware module
  - Platform/Instrument electrical/data link interfaces
  - Provides processing environment for ICU Application software
  - A set of Instrument internal 2nd tier data links and sync lines
- PIM software
  - Boot and service SW (boot, HW dependent SW, service mode)
  - Execution platform for Application software (PUS service, RT OS) in operational mode
  - (Simple) tools for Application SW development and validation

Boot & Service Mode

209	Execute BIT	209	Report ASW death report
209	BIT TM	209	Death report
209	Load SW image	209	Execute w arm reset
209	Report anomaly during SW image loading	209	Stay in boot and service mode
209	Start ASW		

Application Software execution platform

1	Telecommand Verification	1, 2, 7, 8
3	Housekeeping and Diagnostic Data reporting	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 25, 26, 129, 130, 131, 136, 138 139
5	Event Reporting	1, 2, 3, 4, 5, 6, 133, 134
6	Memory management	2, 5, 6, 9, 10, 128
9	Time Management	132, 135
12	On-Board Parameter Monitoring	1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12
17	Test	1, 2, 3, 4
19	Event & Action	1, 2, 3, 4, 5, 6, 7, 130, 131
148	On-Board Macro Procedure	2, 3, 4, 8, 9, 10, 11, 128, 129, 130, 131, 132, 133, 140, 141, 142

Common Platform Interface Module (PIM) – 2/2

What makes PIM-based commonality (even more) efficient

- A PIM form factor that fits all Instrument needs
  - Or conversely, the use by all Instruments of a common form factor
- Application Software execution platform (incl. PUS library) commonality between ICUs and central OBC software
  - To benefit from past/on-going PUS developments and be able to use wide range of PUS subservices without incurring a high cost at Instrument level
  - To minimise Instrument/platform incompatibility risks during development

Promising technologies

A more compact Platform Interface Module

- Replace the current framed module (« unit slice ») by a single compact « component » gathering micro-processor SoC, FPGA, memory
  - ➔ Deployment on instruments with stringent mass/power/volume requirements, e.g. deep space exploration
- A single processing function to replace the PIM of several instruments
  - ➔ Time/Space Partitioning as enabling technology
  - ➔ Mass/Power/Volume benefits while maintaining development/validation independence between platform and payload and between payloads
  - ➔ SW/SW Interfaces to be standardised

Conclusion

- Commonality based on
  Processor Module + Boot/Service SW + Execution Platform SW
  is an efficient lever to increase the efficiency of spacecraft development which involve ICU based Instruments
- Payload developers now able to focus on the functions and performances specific to their instrument
- Prime to play an active role to promote the commonality, including possible synergy with central platform software
- Future technologies (HW and/or SW) such as TSP and very compact electronics may bring additional benefits regarding mass, volume, power