Faster, Later, Softer: COrDeT – an on-board software reference architecture

Andreas Jung and Jean-Loup Terraillon
European Space Agency
ESTEC, Software Systems Division

2010 Workshop on Spacecraft Flight Software (FSW-10)
8 Dec 2010, Pasadena
Spacecraft on-board software landscape – Observations and concerns

Software **size** of the central computer's ESA missions is increasing...

- **Science satellites**
  - Exosat (launch 1983), RCA1802 – 8K memory, ASM
  - SOHO (launch 1995), 2xMDC281 – 2x64KB, Ada83
  - Rosetta (launch 2004), 2xMA3-1750 – 2x1MB, 170KLoc, Ada83

- **Launcher**
  - Ariane5, 68020, ~200KLoc, Ada

- **Earth Observation**
  - Cryosat-2 (launch 2009), ERC32 – 4MB, ~50KLoc Ada95
  - GOCE (launch 2009), ERC32 – 4MB, ~100KLoc, Ada95
  - Aeolus, ERC32 – 4MB, ~160KLoc

- **ATV (launched 2008)**, ERC32--8MB, 1MLoc (650KLoc code), Ada95

...and their **complexity** is increasing
(see also NASA Study on Flight Software Complexity, 2009).
The **schedule** for the software development is getting tighter:

Nevertheless:

Spacecraft platforms have **similar functionalities**. There are families of spacecrafts (for science, earth observation, ...).

The **platform software is even more similar**... but currently, there are few opportunities to spend effort on advanced functions.
Software Engineering needs: Faster, Later, Softer

FASTER (increase productivity)
- Shorter software development time
- Reduce Verification and Validation effort
- Reduce recurring developments (don’t redevelop recurring software: about 50% of platform SW)
- Increase cost-efficiency (more requirements same cost)
- Quality of the product (at least same quality)

LATER (increase reactivity)
- Mitigate the impact of late requirement definition or change
- Optimize flight maintenance
- Simplification and harmonization of FDIR

SOFTER (increase flexibility)
- Support for various system integration strategies (customer-supplier)
- Industrial policy support
- Role of software suppliers (multi-vendor policy)
- Dissemination activities (concept usable by system engineers)
- Future needs
Why a reference architecture replies to these needs?

**FASTER? → automation of life cycle**, model driven engineering

*yes, but not enough...*

We need also **pre-development of software** for faster configuration, later configuration, softer developments (6 years, 6 months, 6 days...), e.g. missionisation of launchers

*yes, but...*

Predevelopment of what? → Of **building blocks**

Are they Lego? → No, they are flexible (parameterization)

Compose Building Blocks? → Therefore need **interface standardization**

Where are the interfaces? → Therefore a **reference architecture**

Reduce validation? → Composability and compositionability,

*Separation of concerns, correct by construction, component model*
SAVOIR – the umbrella of avionics reference architecture related activities

SAVOIR
Space Avionics Open Interface Architecture

SAVOIR Advisory group:
- Space Agencies (ESA and national)
- Prime industry
- Supplier industry

SW: SAVOIR-FAIRE working group
SW R&D activities

Other SW / HW groups and activities

EA UNCLASSIFIED – Releasable to the Public
ESA R&D studies and activities on software reference architectures:
- COrDeT-1/2 – Component Oriented Development Techniques
- DOMENG – Domain Engineering
- SAVOIR-FAIRE - SW reference architecture working group

Results:
- Result from COrDeT activity: Spacecraft platforms and software have similar functionalities, even across the different families (science, earth observation, ...).
- Therefore: Opportunities to spend more effort on (advanced) functions rather than “re-inventing the wheel” for the common elements → REUSE
How to arrive at and What is a reference architecture?

Reference architecture = mapping of Functional chains & Variability factors on to Software Architectural concepts

Building blocks & Interfaces

Components, Containers, Connectors

Execution platform

Physical architecture

Ground

Functional chain

OBSW

OBC

Sensors
Actuators
Payload

Execution platform

Component A

Connector AB

Component B

Container A

Container B

Onboard Communications H/W (e.g. MIL-STD-1553B, SpaceWire, CAN RS422)

TM/TC

Security

Solid State

Mass Memory

File/Compress/Encrypt

Payloads & Instruments

SSMM

RTU/Intelligent IO

Payload Computer

SOIS

Legacy devices

Standardized devices

Intelligent devices

ADCs / DACs

SOIS

Layers

Sensor and actuators

Digital Sensorbus

Space Linux

Ground OBSW Sensors Actuators Payload

Functional chain

Building blocks & Interfaces

ESA UNCLASSIFIED – Releasable to the Public
Component Based Software Engineering

- **Component model**
- **Computational model**
- **Physical architecture**
- **Execution platform**
- Properties Verification

Components, Containers, Connectors

- "Decorated interface"
- Services for container, services for connector, services for component
The SAVOIR avionics reference architecture (HW + SW)
The SAVOIR avionics reference architecture (HW + SW)

Execution platform

- PUS and MTL services
- Abstract component services
  - OBCP interpreter
  - PUS monitoring
  - Avionics Equipment virtual devices = SOIS DVS
- Component services
  - Context Mgmt
  - On-board time = SOIS TAS
- Connector services
  - Communication services addressing physical distribution across nodes = SOIS MTS
- Container services
  - RTOS
  - BSP

Application BB (mission dependent)

- AOCS
- Plan/ Autonomy Framework
- System mode mgmt
- Power
- OBT Mgmt
- P/L Manager

ABB supported by abstract components:
- Central FDIR
- Satellite Conf and Eqpt Mgmt
- SSMM Mgmt

Software bus

Libraries:
- mathematical, etc.

SOIS Subnetwork layer (1553, CAN, SpW)
(including HDSW)

Container services

Legacy devices

Payload & Instrumentation

Space Linux

Payload Computer
The Function Chain

Onboard Communications H/W (e.g. MIL-STD-1553B, SpaceWire, CAN RS422)

- Legacy devices
- Standardized devices
- Intelligent devices
- ADCs / DACs
- Digital Sensorbus
- SOIS Layers
- RTU/Intelligent IO

Payloads & Instruments
- Space Linux
- Payload Computer

OBSW

Functional chain

Ground

OBC

TM/TC
- Security Unit
- Solid State Mass Memory
- File/Compress/Encrypt

SSMM
- SOIS Layers

TM/TC
- Security Unit

OBSW

Payloads & Instruments

ESA UNCLASSIFIED – Releasable to the Public
Mapping Functional Chain onto SW architectural concept

Ground

Functional chain

Sensors
Actuators
Payload

Components,
Containers, Connectors

Execution platform

OBC

Ground
AOCS
Startracker

TM/TC
Security Unit

SSMM
Solid State Mass Memory
File/Compress/Encrypt

SOIS Layers

Outboard Communications H/W
(e.g. MIL-STD-1553B, SpaceWire, CAN RS422)

Payloads & Instruments
Space Linux
Payload Computer

Standardized devices
Legacy devices
Intelligent devices
Sensor and actuators

RTU/Intelligent IO
ADCs / DACs
Digital Sensorbus
SOIS Layers

Intelligent
devices

Sensor
and actuators

SOIS Layers

Legacy
devices

TM/TC

Payloads & Instruments

OBC

Container A
Component A
Connector AB

Container B
Component B

Ground

ESA UNCLASSIFIED – Releasable to the Public
The SAVOIR (Software) reference architecture

- Onboard Communications H/W
  - (e.g. MIL-STD-1553B, SpaceWire, CAN RS422)

- Execution platform
  - OBC Hardware
    - CPU/NGnP
    - RAM
    - DSP
    - EEPROM
    - Boot
    - PROM

- Software bus
  - Security Unit
  - Solid State Mass Memory
  - File/Compress/Encrypt
  - SOIS Layers

- Libraries: mathematical, etc.

- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)

- Application BB (mission dependent)
  - AOCS
  - Thermal
  - Power

- System mode mgnt
  - OBT Mgmt
  - PL Manager

- PUS and MTL services
  - OBCP interpreter
  - Context Mgmt
  - On-board time
  - SOIS TAS

- Abstract component services

- Connector services
  - Communication services addressing physical distribution across nodes
  - SOIS MTS

- Container services
  - RTOS

- Container services
  - Abstract component services
  - PUS and MTL services

- Application BB (mission dependent)
  - Central FDIR
  - SSMM Mgmt

- Central components:
  - Satellite Conf and Event Mgmt

- Legacy devices
  - Standardized devices

- Intelligent devices
  - ADCs/DACs
  - Digital Sensorbus

- SOIS Layers

- Payloads & Instruments
  - Space Linux
  - Payload Computer

- SOIS Subnetwork layer (1553, CAN, SpW)
Mapping of functional chain on to the SW architecture

Execution platform
- OBC Hardware
  - CPU/NGmP
  - RAM
  - DSP
  - EEPROM
  - Boot PROM
- OBT Timer
- SGM
- HW watchdog
- OB Timer
- SSMM Mgmt
- OBT Mgmt
- PL Manager
- Mil-1553
- SpaceWire
- CAN
- RS422
- Solid State Mass Memory
- File/Compress/Encrypt
- SOIS Layers
- Libraries mathematical, etc.
- Part of OBC Hardware
- OBC supported by abstract components
- Control DFB
- Central DFB
- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)
- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)
- Plan/Autonomy Framework
- System mode mgmt
- Power
- Ground
- AOCS
- Startracker
- Intelligent devices
- Sensor and actuators
- ADCs / DACs
- Digital Sensorbus
- SOIS Layers
- Payloads & Instruments
- Space Linux
- Payload Computer
- thermo
- power
- satellite
- Computer
- mission
- dependent
- Plan
- Autonomy
- Framework
- System
- mode
- Power
- Ground
- AOCS
- Startracker
- TM/TC
- Security Unit
- SSMM
- Solid State Mass Memory
- File/Compress/Encrypt
- SOIS Layers
- Libraries mathematical, etc.
- Part of OBC Hardware
- OBC supported by abstract components
- Control DFB
- Central DFB
- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)
- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)
- Plan/Autonomy Framework
- System mode mgmt
- Power
- Ground
- AOCS
- Startracker
- Intelligent devices
- Sensor and actuators
- ADCs / DACs
- Digital Sensorbus
- SOIS Layers
- Payloads & Instruments
- Space Linux
- Payload Computer
- thermo
- power
- satellite
- Computer
- mission
- dependent
- Plan
- Autonomy
- Framework
- System
- mode
- Power
- Ground
- AOCS
- Startracker
- TM/TC
- Security Unit
- SSMM
- Solid State Mass Memory
- File/Compress/Encrypt
- SOIS Layers
- Libraries mathematical, etc.
- Part of OBC Hardware
- OBC supported by abstract components
- Control DFB
- Central DFB
- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)
- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)
- Plan/Autonomy Framework
- System mode mgmt
- Power
- Ground
- AOCS
- Startracker
- Intelligent devices
- Sensor and actuators
- ADCs / DACs
- Digital Sensorbus
- SOIS Layers
- Payloads & Instruments
- Space Linux
- Payload Computer
- thermo
- power
- satellite
- Computer
- mission
- dependent
- Plan
- Autonomy
- Framework
- System
- mode
- Power
- Ground
- AOCS
- Startracker
- TM/TC
- Security Unit
- SSMM
- Solid State Mass Memory
- File/Compress/Encrypt
- SOIS Layers
- Libraries mathematical, etc.
- Part of OBC Hardware
- OBC supported by abstract components
- Control DFB
- Central DFB
- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)
- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)
- Plan/Autonomy Framework
- System mode mgmt
- Power
- Ground
- AOCS
- Startrack
How is a Block reusable?

1. Architecturally reusable
   → ensured by interface standards and component model
      (composability, compositionality)

2. Functional reusable
   → Domain engineering
   → Variability factors
      (characterize the domain of reuse)
Mapping variability factors

Legend:

- Mission
- System operational concept
- Avionics
- Monitoring & Control I/F
- Network
- Processor module

Execution platform

- PUS and MTL services
- PUS monitoring
- Avionics Equipment virtual devices = SOIS DVx

OBC Hardware

- LIB libraries: mathematical, etc.
- SOIS Subnetwork layer (1553, CAN, SpW) (including HDSW)

Payloads & Instruments

- Space Linux
- Payload Computers

SSMM

- Solid State Mass Memory
- File/Compress/Encrypt
- SOIS Layers

TM/TC

- Security Unit
- Application BB (mission dependent)
- To be confirmed if ABB:
  - Central FDIR
  - SSMM Mgmt
- System mode mgmt
- OBT Mgmt
- PL Manager

Software bus

Legend:

- Onboard Communications H/W
  (e.g., MIL-STD-1553B, SpaceWire, CAN RS422)

Outboard Communications H/W

- RTOS
- Context Mgmt
- On-board time = SOIS TAS
- Communication services addressing physical distribution across nodes = SOIS MTS

Container services

- Connector services
- Abstract component services
- Container services

PUS specific

- Component services
- Abstract component services
- PUS and MTL services

Payloads & Instruments

- Intelligent devices
- Sensor and actuators

- ADCs / DACs
- Digital Sensorbus

- SOIS Layers

- Payloads & Instruments
- Space Linux
- Payload Computers

- Intellignet devices
- Sensor and actuators

- ADCs / DACs
- Digital Sensorbus

- SOIS Layers
Mapping variability factors

Legend:
- Mission
- System operational concept
- Avionics
- Monitoring & Control I/F
- Network
- Processor module
Based on the **SAVOIR definition**, a building block:

1. Has a *clear, open, well-defined, specified, documented* function and interfaces
2. Is *worth developing*, i.e. utilization is envisaged at least for the bulk of the ESA missions
3. *Meets* defined performance, operation and other *requirements*
4. Is *self-contained* so as to be compatible with utilization at higher integration levels, e.g. board, equipment, subsystem
5. *Composability and Compositionality* of its properties shall be guaranteed
6. Has a *TRL and quality level* which can be assessed
7. Is applicable in an *envelope* of well defined physical and software environment
8. Results from a process that can be *repeated with guarantees*
9. Is *designed for reuse* by different users, in different projects (it may be configurable depending on the variability factors)
10. Can be made *available off-the-shelf, under defined conditions*
Complementary activities:
1st: Time and space partitioning

Integration of TSP in the software reference architecture

Extraction completed successfully.
On-board control procedures (OBCP) interpreter is part of the execution platform.
Open points for the software reference architecture

- **Hierarchical** components

- **Architectural decisions:**
  - Fault Detection Isolation Recovery
  - Monitoring, On Board Control Procedure interpreter: common mechanisms in several components

- A **new validation process** also reusable; validate functional and non functional separately; reuse tests suites

- **Methods and tools:**
Summary

COrDeT – OBSW reference architecture:
- Software architectural concept
  → Component based software engineering
- Functional chains & variability factors
  → (building) block & interfaces

Reference architecture = mapping of Functional chains & Variability factors on to Software Architectural concepts

COrDeT - OBSW ref arch | Andreas JUNG and Jean-Loup TERRAILLON | 2010 Workshop on FSW | 8.Dec 2010 | TEC-SW | Slide 23

ESA UNCLASSIFIED – Releasable to the Public
THANK YOU

Andreas Jung and Jean-Loup Terraillon
European Space Agency
Andreas.Jung@esa.int
Jean-Loup.Terraillon@esa.int