SAVOIR: Reusing specifications to improve the way we deliver avionics

Jean-Loup Terraillon

SAVOIR: Reusing specifications to improve the way we deliver avionics

SAVOIR Advisory Group
represented by
Jean-Loup Terraillon
European Space Agency
Savoir@esa.int

Keywords
Savoir, space avionics, product line, reference specifications

Abstract
SAVOIR has taken inspiration from AUTOSAR, although the underlying industrial business model is different. The space community is smaller, the production is based on a few spacecraft per year, and there are industrial policy constraints. Still, there is a need to streamline the production of avionics and improve competitiveness of European industry. Reference architectures, reference specifications and standard interfaces between building blocks are an efficient mean to achieve the goal. Reusing specification is expected to allow reusing products.

What You Must KNOW About SAVOIR...

What is SAVOIR

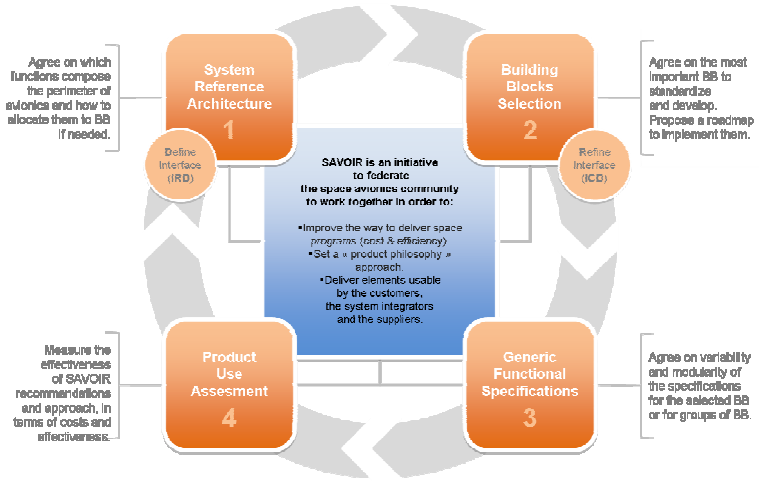
SAVOIR (***S**pace **A**vionics **O**pen **I**nterface a**R**chitecture)* is an initiative to federate the space avionics community and to work together in order to improve the way that the European Space community builds avionics subsystems.

The objectives are:
- to reduce the schedule and risk and thus cost of the avionics procurement and development, while preparing for the future,
- to improve competitiveness of avionics suppliers,
- to identify the main avionics functions and to standardise the interfaces between them such that building blocks may be developed and reused across projects
- to influence standardisation processes by standardising at the right level in order to obtain equipment interchangeability (the topology remains specific to a project).
- to define the governance model to be used for the products, generic specifications, interface definition of the elements being produced under the SAVOIR initiative.

The process is intended to be applied as part of the Agencies ITTs, and throughout the subsequent procurements and development process. A particular goal is to have Savoir outputs exploited in future projects and relevant products as part of European supplier's portfolios.

The primary outputs of Savoir are:
- reference avionics architecture for spacecraft platform hardware and software,
- a set of standard avionics external and internal interface specifications,

- the definition of building blocks composing the architecture,
- the functional specification of selected building blocks comprising the architecture,
- the implementation of selected building blocks at the right TRL level,
- process definition and assessment.



SAVOIR is based on the definition of an avionics reference architecture (1) allowing to define building blocks (2) suitable for a domain of reuse. For selected building blocks, generic functional specification is produced, that includes the necessary variabilities to cover the domain of reuse (3). Industry develops and reuse the building blocks. The process is measured (4) and restarted in order to refine the architecture or the generic specification according to the building block success.

Figure 1: The SAVOIR wheel

The expected benefits of Savoir are:
- for customers, streamline the procurement process of spacecraft avionics,
- for system integrators, facilitate the integration of the spacecraft avionics ,
- for suppliers, prepare the technical conditions for a more efficient product line organization.

SAVOIR supports:
- the space avionics customers and system architects,
- the system integrators,
- the avionics and technology suppliers,
- the standardization bodies.

It is a tool for the industrial policy and R&D plan makers.

Example of SAVOIR output are:
- inputs to the harmonization process
- definition of reference architectures in avionics and software
- initiation of new standards where required
- functional reference specifications of the computer and data concentrator

SAVOIR is coordinated by the Savoir Advisory Group including representative of ESA, CNES, DLR, Astrium, Thales, OHB, RUAG, Selex Galileo, Terma.

How SAVOIR works

Standardisation already exists in space, in the frame of the European Cooperation for Space Standardisation (ECSS, www.ecss.nl) organization and the international standards group Consultative Committee for Space Data Standards (CCSDS). Bus protocols are being standardized (MilStd1553, Can, SpaceWire) as well as communication services (SOIS) and space link protocols. Reference architectures, functional interface and specifications are not yet covered. The SAVOIR outputs are intended to be proposed to ECSS as particular type of document, between standard and handbook, related to products.

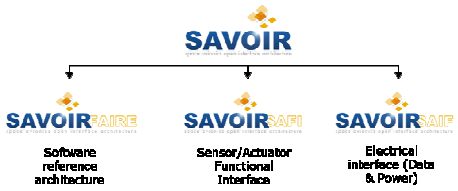
The Savoir ***work plan*** is supported by R&D activities defined in the ESA harmonized R&D plan. This covers multiple domains such as avionics, data handling, payload data processing, microelectronic, control, and software. They address in particular:
- consolidation of the reference architecture (for data handling, AOCS, software)
- modeling of the architecture (e.g. in AADL),
- elaboration of reference specification (for Computers, Remote Terminal Units, software execution platform and functional chains, software and hardware communication)
- onboard communications services and the use of electronic datasheets for interface definition
- interface standardization (for bus protocols, sensor/actuators functional interface, payload interface, space ground interface)
- prototyping of some building blocks.

The SAVOIR ***portfolio*** is organized according to the technology levels: basic technologies, components, modules, equipment, systems. Basic technologies and components are considered as enabling technologies. Reference specifications apply more to modules and equipments. The building blocks are provided with either only a reference specification, or with an implementation at various Technology Readiness Levels.

The ***Savoir Advisory Group*** is supported by sub working groups:

- *SAVOIR-S/A-I/F*
This working group addresses the *electrical interface* of the sensor actuators used for the attitude control and the guidance of the spacecraft.
It has concluded on the selection of the Spacewire and the 1553 as bus standards, and has recommended the standardization of a RS422 interface protocol, which has then been defined in subsequent R&D activities.

- *SAVOIR-SAFI*
This working group is in charge of the standardization of the *functional interface* with sensor and actuators. It intends to prepare the use of interface mechanisms based on the use of Electronic Data Sheets as defined in CCSDS SOIS standards



- *SAVOIR-FAIRE*
This working group is in charge of the on-board *software* reference architecture. It has proposed an architecture (named COrDeT) based on the segregation of the application software (independent from the execution context and expressed with components: Platform independent Model) and the execution platform (providing services to components such as real-time scheduling, communication and specific services).

The SAVOIR Output

<u>The Avionics Reference Architecture</u> (hardware, software, communication)

The considered mission domains (domain of reuse) includes:
- Science and Earth Observation missions with up to 12 years duration to LEO, GEO, Lagrange points, Interplanetary space
- Telecom missions with up to 15 years lifetime
- Commercial earth observation missions

For these missions, an Avionics Reference Architecture has been first defined at high level.

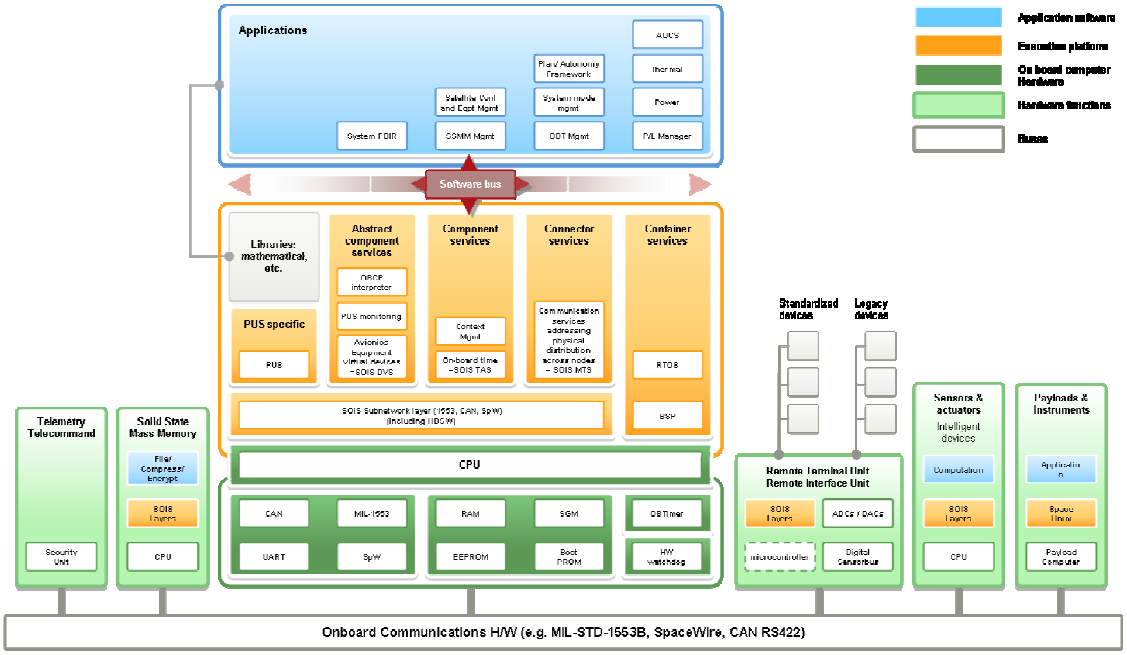


Figure 2: The SAVOIR Avionics Reference Architecture

- The on-board communication links are the backbone of the avionics architecture. They allow connecting together the main electronic functions of the spacecraft. Generally they are split into several serial links to segregate the data flows and to propose an adapted link to the various data rate exchanges (low-speed communication links and high-speed communication links).
- The TM/TC function is in charge of the RF link with the ground for both directions. It generally includes the TM/TC transceivers, the antennas and the RF harness. Security functions are functionally attached to this function for TC authentication and TM
- The Mass Memory is the storage mean for high volume of data: Housekeeping

Telemetry (HKTM) and science data. It provides the capability to manage data files and to compress science data.
- The On-Board Computer includes the processor and its software. The Computer software, as well as any avionics software, is systematically split into an infrastructure part (for communication and real-time) and an applicative part.
- The RTU/RIU, is in charge to connect the dumb units (sensors and actuators) to the main communication bus, to collect discrete data from the spacecraft (thermistors, status…) and to dispatch discrete commands to the spacecraft.
- The set of AOCS sensors and actuators is selected regarding the orbit characteristics, the spacecraft size, the pointing accuracy / agility required by the mission and the reliability constraints.
- The payloads and instruments are seen in this document as an external function which requires data handling interfaces and processing resources to manage it.

The reference architecture implements a set of requirements: space ground communication, on-board communication, dependability, fault management and autonomy requirements, on-board function and performance, design constraints, operability requirements. These requirements include the variability associated to the domain of reuse expressed as parameters (number of command per second, data throughput, availability, accuracy, protocols, etc).

The hardware and communication functional architecture

From the avionics reference architecture, the hardware and communication *functional* architecture is defined [ASRA].

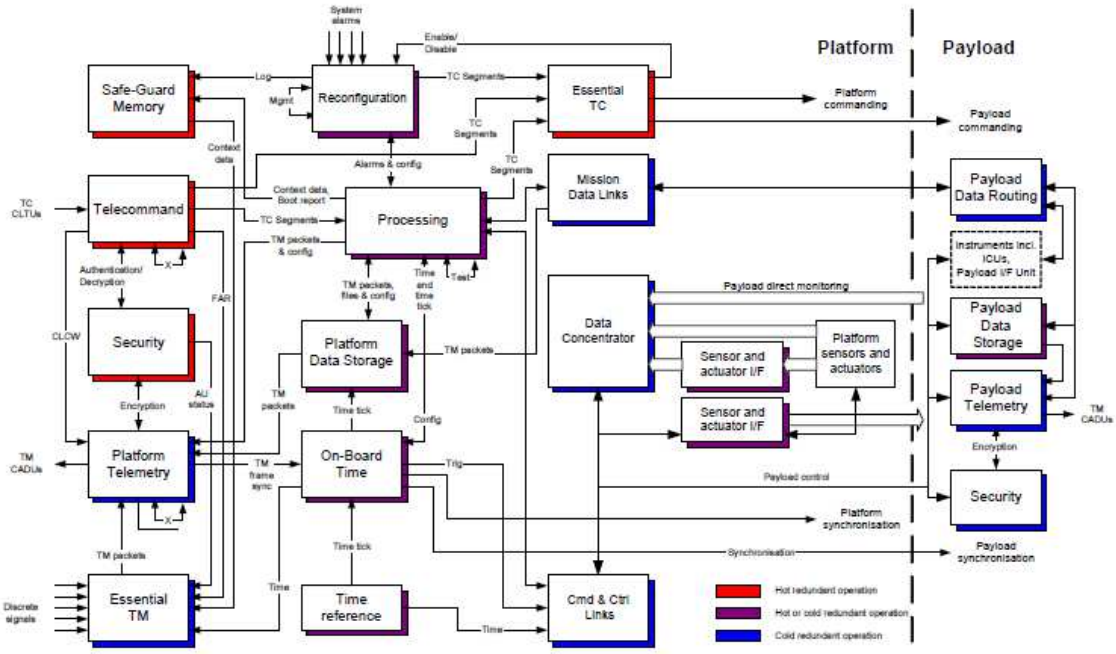


Figure 3: The SAVOIR hardware and communication functional architecture

The hardware and communication functions can be grouped into various *physical* hardware equipments according to the technology of various Suppliers, or according to the evolution of technology of a given Supplier. For example, the functional

architecture is robust to a physical bus change, to the addition of data concentrators, and even to the relocation of function processing through software mobility (e.g. from sensor to computer).

Each function can then be associated to its generic specification with its variability parameters. The functions are grouped to define physical equipments. The trend in space platform avionics is to converge towards two major equipments: the On-Board Computer (OBC) and a set of intelligent bus bridges and data concentrators called Remote Terminal Units (RTU) or Remote Interface Units (RIU).

SAVOIR is currently working on the production of these two generic specification. They will refer to the existing or future interface standards, in particular the avionics bus protocols such as the MilStd1553, the SpaceWire, the CAN bus, the serial link, which will be available in ECSS.

The software architecture

Motivation: Faster, Later, Softer

The OBSW life cycle must be organised in consistency with the system life cycle that imposes the definition of functional increments. It must in particular:
- allow for a *faster* software development in the context of a reduced schedule, i.e. the ability to release in a short time a new version of the OBSW product but also ultimately the complete and validated product,
- be compatible with a *later* definition or changes of some of its requirements, typically for mission specificities like the system FDIR, system mission management or adaptation to hardware changes,
- cope with the various system integration strategies, i.e. be flexible (*softer*) enough to allow for early release for integration needs, or to cope with late hardware availability

From the above given programmatic stakes, the slogan "*Faster, Later, Softer*" [ERTSS10] represents a summary of three stakes for the on-board software life cycle. Those stakes are included and defined as user needs.

User needs

In order to guide the work on software architecture, some user needs have been collected, such as to get a shorter software development time, to reduce recurring costs, to keep or increase the quality of the product, to increase cost-efficiency, to reduce the Verification and Validation effort, to mitigate the impact of late requirement definition or change, to support various system integration strategies, and to increase the role of software suppliers.

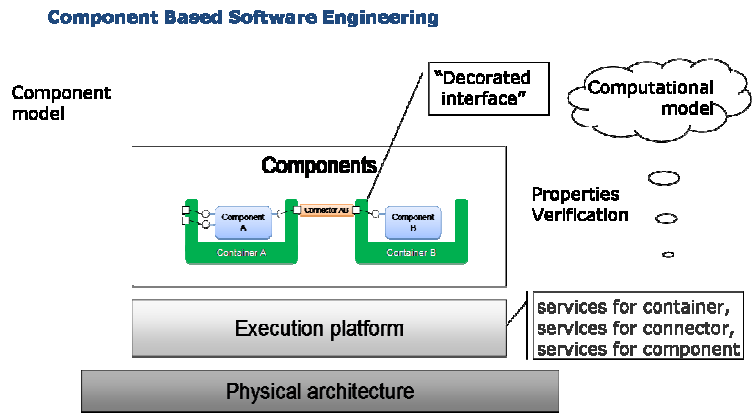
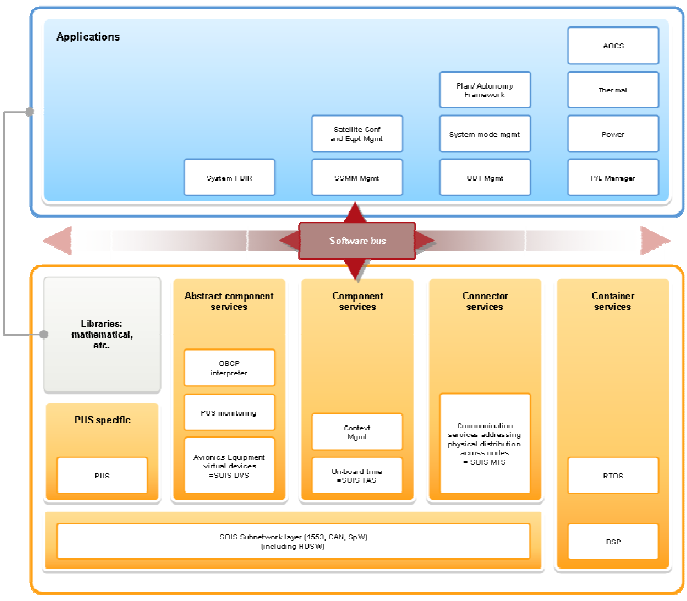


Figure 4: Component Model

Software proposed solution

The various technical activities supporting the Savoir-Faire working group have produced the On Board Software Reference Architecture called "COrDeT" [COrDeT].

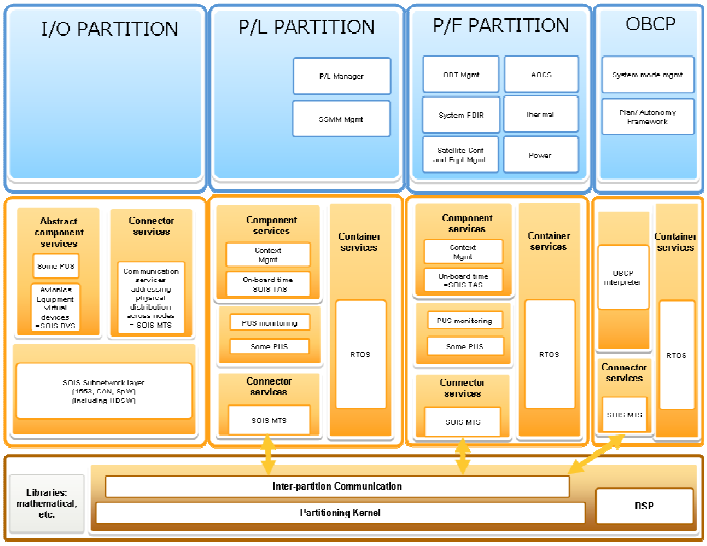
Figure 5: The COrDeT on-board software ref. architecture



The definition of the software reference architecture relies on the notion of *component model*. Application software is implemented with components that don't have a particular knowledge of their execution environment. Components are executed through services (real-time, communication, etc) offered by an *execution platform* that deploys the components on the physical architecture of choice.

The mapping of the components on the execution platform is done by a set of design rules materialized in a thin *interaction layer* on top of the execution platform. This layer can be generated automatically with an automated set of tools called *software bus*.

The attributes allocated to components allow to verify properties of the software by automated verification of design. This notion, called "*correct by construction*" allows to replace part of the testing by earlier verification in order to benefit the schedule.



The application software implements the platform functional chains by use of components. The architecture separates clearly the informatics issues (such as communication, real-time, software mechanisms) from the functional sub system needs (AOCS, Thermal, Power, etc). This *separation of concerns* allows to decouple the system and the low level software issues and give flexibility and reactivity to the software life cycle.

Figure 6: COrDeT and TSP

An example of flexibility is the introduction of the Time and Space Partitioning concept in the Execution Platform, which can be done by adding a partitioning kernel and moving some building blocks in the appropriate partitions [Figure 6].

The COrDeT generic specification

In compliance with SAVOIR objectives, there is a set of generic specification for the on-board software, which includes:

- the *on board software reference architecture specification,* including about 50 technical needs on the architectural process, spacecraft database, execution platform, methods and tools, development process, integration, and V&V.
- the *execution platform interface specification,* which describes 10 services, including 3 from the CCSDS SOIS (Message Transfer Service, Device Access Service, Command and Data Access Service), and task service, system management service, commanding service, reporting service, monitoring service, automation service, archiving service. Each service is described with his parameters and the description of the primitives offered, with function, semantics, when it is generated, and its effect on receipt. The functional specification of the services is under development.
- the *component model needs.* However, it is questioned if a Component model specification needs be developed. It will depends on the industrial selection of one or several components models and their level of consistency.
- the *tools needs*, about 150 technical needs on tools have been produced, and Obeo designer has been evaluated for the Space Component Model.

The S/A Functional Interface

The goal of the SAVOIR-SAFI working group is to define standardized functional interface for classes of sensor actuators, in order to achieve the following objectives:

- define as early as possible the AOCS, aiming at using the same functional interface at model level and at on-board software level
- reuse the functional interface in AOCS functional simulator all along the development up to the on-board software: the functional interface is automatically put in a specific on-board software envelop
- improve schedule software by minimizing changes of S/A interface when the actual equipment is selected
- streamline the development of the on-board software element that links the control loop to the specific equipment access layers: manual ad hoc glue software is replaced by configurable architectural blocs
- in the S/A interface, separate the functional aspects from the format of data that may be handled by a configuration mechanism.
- expose a more common interface to the users

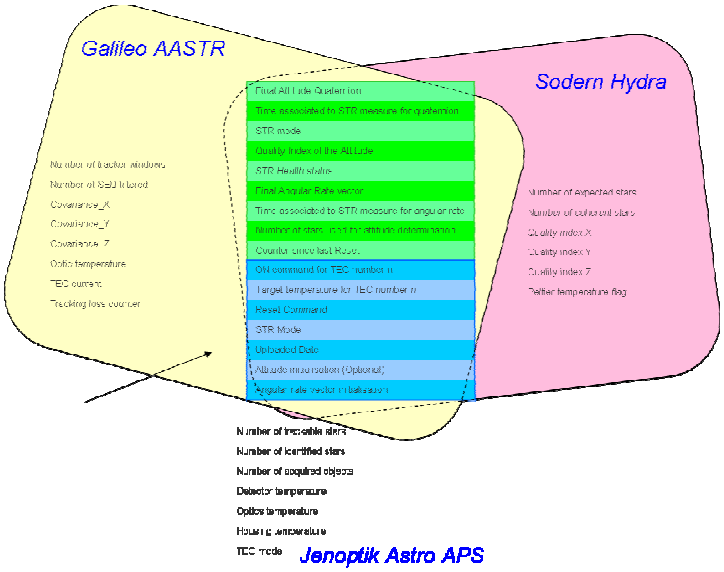


Figure 7: an example of common star tracker interface

Having a standardized S/A functional interface contributes to facilitate the interoperability of different equipments brand on the same platform.

Various software mechanisms implemented in the execution platform or as component level design patterns, allows to decouple the impact of the equipment on the overall software, to minimize the change, and to automate the configuration of the on-board software. The currently envisaged mechanisms are the Device Access Service and the Device Virtualisation Service proposed by the CCSDS SOIS. Their automation would be supported by the Electronic Data Sheet concept. SAVOIR is used as a laboratory to help refining the related CCSDS SOIS standard, by providing real needs of functional interface behaviour, in order to make sure that the SOIS services will reply to the needs.

Business Model

The particular organisation and mission of the European Space Agency does not intend to impose or define business models from pure technological considerations. The industrial policies are elaborated with industry through various mechanisms for the institutional missions, and are combined by Industry with the particular needs of commercial programs.

Still, technology can enable industrial policies, and favour the emergence of product lines. SAVOIR allows to bring stakeholder together to define the conditions that will maximize product reuse. A clear message is that the specifications of similar products should be similar. This imposes, amongst the customer chain, a discipline that is favoured by the existence of generic specification and standard interface. Even if only this goal is achieved, the benefit for avionics development will be there.

Examples of identified use cases are:
- common development by two spacecraft integrators of the same platform concept for future telecom spacecrafts, in order to allow them to instantiate the concept, following their own commercialisation and industrial policy, while procuring interoperable equipments from various sources.
- accommodation of ESA geographical return constraints by allowing the procurement of interoperable sensor/actuators in different countries, or allowing the distribution of software developments amongst several suppliers.
- definition of common ways to operate spacecraft platforms (in order to save on ground segment)

The suppliers business model cannot be demonstrated, quantified, ensured. However, it is sure that the future needs will call for more sophisticated payloads, or more complex systems of systems. The platform will become an "infrastructure", although it must deliver appropriate and accurate services to its payload (e.g. pointing accuracy, life time, availability). The competitiveness of European industry must focus on flexible platform production, in order to reduce the time to market and invest more on payloads.

Summary of the SAVOIR approach and results

SAVOIR is a cooperation of avionics stakeholder to improve the way avionics is delivered. It aims at producing reference architecture, generic specification, and interface standard in view of favouring the emergence of product lines among suppliers. An avionics reference architecture is available

On the hardware side, SAVOIR has produced a functional reference architecture, and is working on the generic specification of the On Board Computer and the Remote Terminal unit
On the software side, SAVOIR has produced the COrDeT on-board software reference architecture, based on a component based development. It allows to produce software faster, to introduce changes later, and to adapt, in a softer way, various industrial constraints. Specifications of the reference architecture and the execution platform are under production, whereas the functional chains generic specification will be started shortly.
On the sensor actuator side, SAVOIR will propose a standard functional interface aiming at improving the interoperability of devices of same class on the same platform.

The remaining challenges are the verification of the completeness of the generic specification, their adequation to the actual needs (domain of reuse), the setup of the appropriate organisation to deploy them, and the measurement and improvement of the SAVOIR process.

Acknowledgment

In addition to the software group mentioned in [ERTSS10], the author acknowledge the contributions of the SAVOIR working group including the SAG (Thierry Duhamel, Astrium - Bernard Bruenjes, OHB System - Carsten Jorgensen, TERMA - Franco Boldrini, Selex Galileo - Jacques Busseuil, Thales Alenia Space - Paola van Troostenberghe, Cnes - Torbjörn Hult, RUAG - Thomas Wolf, DLR - Alain Benoit, Philippe Armbruster, Kjeld Hjortnaes, Juan Miro, ESA) and supporting contributors (Chris Taylor, Giorgio Magistrati, Guillermo Ortega, Roger Jansson, Stephen Airey, Fabio van Hattem, ESA - Bernard Alison, Thales - Remi ROQUES, Astrium, and it is not possible to name them all...)

References

[ASRA] *Avionics System Reference Architecture - SAVOIR Functional Reference Architecture* - RUAG, Astrium, Thales, OHB for SAVOIR - TEC-SW/11-477/JLT - v2 - 23/10/11

[ERTSS10] *Faster, Later, Softer: COrDeT, a reference on-board software architecture for spacecrafts* - J.L. Terraillon, A. Jung, European Space Agency, ERTSS2010

[COrDeT] *Definition Of A Reference On Board Software Architecture Basic Constituents* - Cordet Astrium (Astrium, SciSys), Cordet TAS (TAS, UPD, P&P SW) and ESA; with contribution to this document by Intecs, GMV and SoftWcare - TEC-SWE/09-248/AJ - v1r2 - 03/07/09

[SAVOIR-FAIRE] *SAVOIR-FAIRE - On Board Software Reference Architecture* - Andreas Jung, Marco Panunzio and Jean-Loup Terraillon supported by the Savoir-Faire working group - TEC-SWE/09-289/AJ - v1r0 - 10/06/10