SAVOIR

space avionics open interface architecture

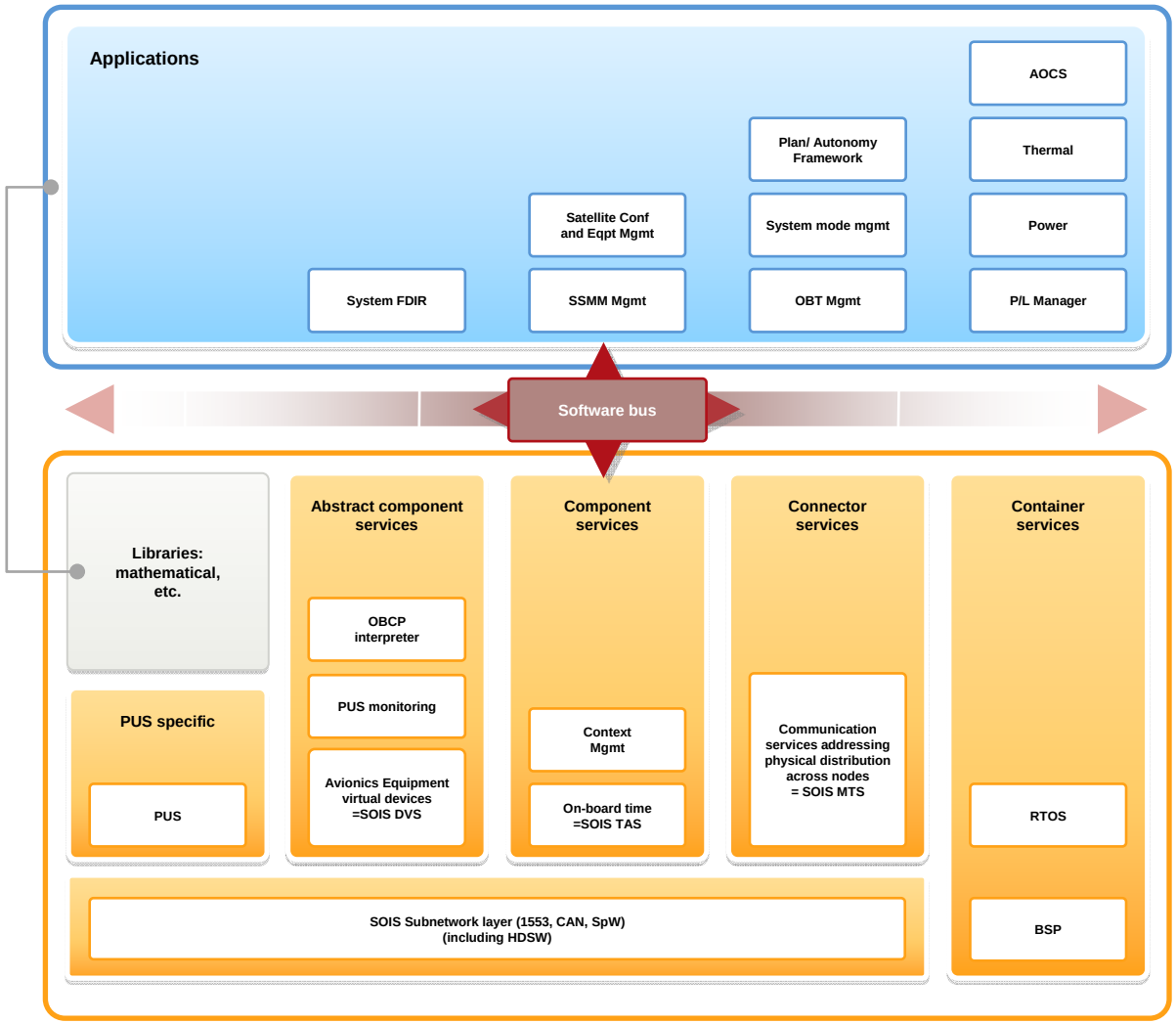
SAVOIR Implementation Strategy
The Software Perspective

J.L. Terraillon
Software Systems Engineering
25/10/2011

Agenda

- Savoir software output
- Generic Specifications
- Prototypes
- Products
- Tool strategy
- Schedule and plans

The software reference architecture



Application level hosts the components of the functional chains

Software bus is a tool that generates the **interaction layer** between both

Execution platform hosts the services needed by the application (mediated through the interaction layer)

The SAVOIR software output

GENERIC SPECIFICATIONS

- Flight Computer Initialisation Sequence (Boot software) generic specification (from within Estec)
- OBSW reference architecture specification, including execution platform and component model (from COrDeT2)
- OBCP generic specification (from GSTP5 EI2 x2)
- TSP, IMA, Security partition kernel specifications
- Generic obsw requirements (from within Estec, draft, includes resources, fault management and functional requirements)
- Functional chains generic specification (from GSTP5EI2 OSRAc x2)
- PUS updated standard interface (planned)
- SOIS standards interface, see status in Chris presentation

PROTOTYPES

- Software bus prototype from TEC laboratory
- COrDeT prototype (planned)
- Multicore system study prototype (see ADCSS11 Thursday)

PRODUCTS

- RTEMS from Portuguese consistent effort

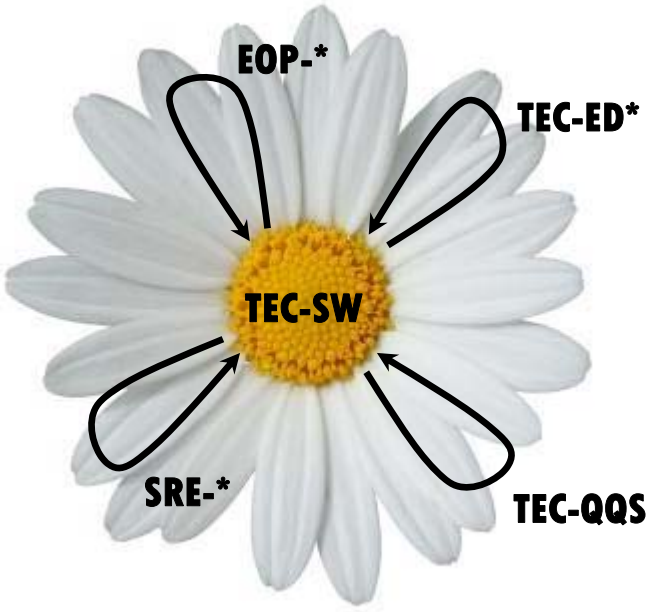
GENERIC SPECIFICATIONS, some examples

Flight Computer Initialisation Sequence generic specification (1/2)

- Integrates the projects' lessons learned
- Is derived from the various existing specifications
- Is applicable to platform or payload computers
- Includes use cases such as nominal sequence, software maintenance in flight and on ground
- Lists a minimum set of requirements
  - functional: nominal sequence, standby, monitor, initialization, self tests
  - non-functional: performance, resources, design, quality
- Each requirement comes with justifications and explanatory notes
- Follows ECSS-E-ST-40C DRD structure for SRS

Flight Computer Initialisation Sequence generic specification (2/2)

- Prepared and reviewed inside Flight Software Systems Section
- Reviewed by a large number of ESA experts coming from:
  - Earth Observation projects
  - Science projects
  - TEC software product assurance
  - TEC data handling



- Review statistics:
  - 18 reviewers
  - 190 RIDs
  - ~100 action items implemented

Contact: Felice.Torelli@esa.int

OBSW reference architecture specification

- 9 High level industrial needs from Savoir-Faire
- On board software reference architecture specification
  - ~50 Technical needs on the architectural process, spacecraft database, execution platform, methods and tools, development process, integration, V&V.
- Execution platform interface specification
  - 10 services, incl.
    3 from CCSDS SOIS (MTS, DAS, CDAS) and Task service, system management service, commanding service, reporting service, monitoring service, automation service, archiving service,
  - Each service described with his parameters and the description of the primitives offered, with function, semantics, when generated, effect on receipt.
  - Functional specification of the services (to be done)
- Component model needs. Component model specification to be developed.
- ~150 technical needs on tools, with an evaluation of Obeo designer for SCM

4.7.6. System Management Service

The System Management service is responsible for context management and the reporting of Execution Platform errors to the Interaction Layer.

4.7.6.1. Parameters

The following parameters are used by the primitives in the Task service.

Parameter Name	Description
ContextAttrIDSet	A set of unique identifiers associated with attributes
ContextAttrValue	A set of values of an attributes
ContextID	A unique identifier associated with a system context
ErrorID	A unique identifier associated with an error type from the service
Status	Status information associated with the completion of an operation

4.7.6.2. Primitives

SaveContext.request

- **Function**

The SaveContext.request shall be used to request a save of the specified context attributes to the named ContextID.

- **Semantics**

The SaveContext.request primitive shall use the following semantics:

SaveContext.request(ContextID, ContextAttrIDSet, ContextAttrValueSet)

- **When Generated**

The SaveContext.request is generated to save a copy of the current values for the set of parameters specified by the attribute set, saving them in the specified context.

- **Effect on Recipt**

Receipt of the SaveContext.request primitive shall cause the service to save the current values for the specified set of attributes to non volatile memory referenced as the named ContextID.

- **Additional Comments**

None

TSP, IMA, Security

IMA-SP:

- Users requirements
- Application service specification (tailoring of ARINC653)

→ reviewed by the IMA-SP consortium

Security:

- Generic security requirements for separation kernels
- Separation kernel protection profile for Xtratum (based on Common Criteria Separation Kernel Protection Profile)

→ to be reviewed by a wider audience

Specs to be revisited after the prototyping activities.

Contact: James.Windsor@esa.int

PROTOTYPES

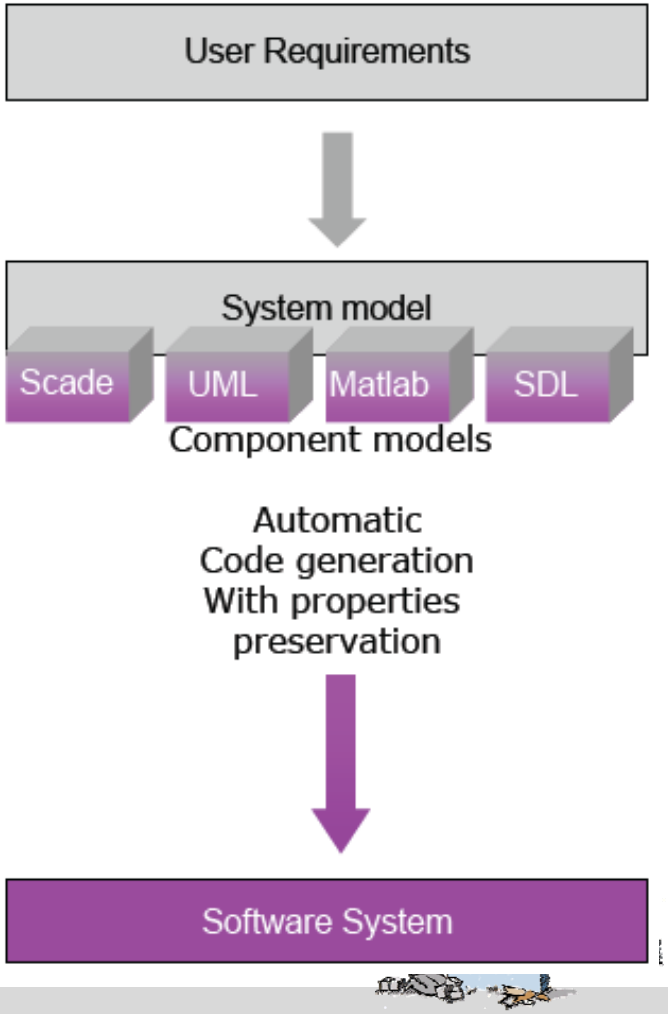
TASTE, a compiler of Models

Source code is replaced by models in various languages :

- Matlab/Simulink,
- State machines (stateflow, SDL),
- Data models (ASN.1)
- Microelectronic models (VHDL)

but can integrate C or Ada.

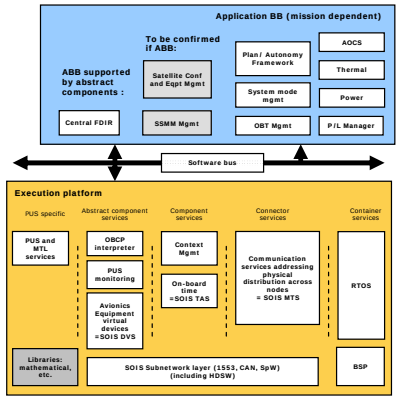
TASTE is the link between the Model Based System-Software Co-engineering and the software implementation

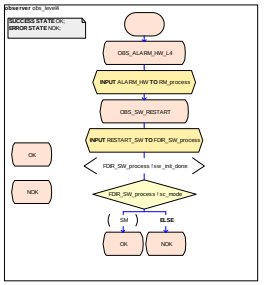


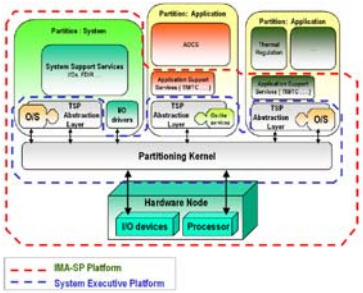
TASTE, the lab prototype of the future

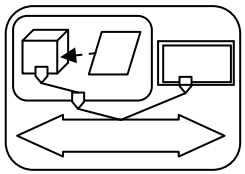
TASTE is the laboratory prototype used to validate most of the innovative R&D concepts:

- On-board software reference architecture (component model)
- Configurable execution platform (single/distributed/TSP/multi-core)
- Software bus mapping the components on the execution platform
- Correct by construction:
  formal verification of the design in order to reduce the test campaign
  (e.g. schedulability analysis, dependability)
- FDIR modelling
- Avionics modelling
- Interface OBSW/Simulator through SMP [ECSS-E40-07]
- HW-SW Co-design









http://www.assert-project.net/-TASTE-

Prototype of the software bus

COrDeT component model

SCM to Taste

TASTE component model

RASTA

Taste execution platform:

-Polyorb (SOIS MTS)

-Edisoft RTEMS

-SOIS subnetwork

taste

XILINX

- Interaction layer generation
- Real-time architecture generation
- Models & interface code generation
- Deployment on hardware
- Hardware software co-design

Taste execution platform

Execution Platform

- SOIS MTS semantic compliant Send/Receive; Partially compliant synchronous query & announcement; Publish/Subscribe not implemented (not needed): [TRL 4]
- Compliant SOIS sub-network layer (SpW) [TRL 4]
    - (Note: includes the component view of drivers)
- No other SOIS services implemented, no PUS, no OBCP yet
- Operating system RTEMS Edisoft [TRL 6]
- Partitioned operating system POK [TRL 3]; Xtratum [TRL 2]

Software bus

- Generation of data model (interface) code, [TRL 4]
- Generation of *interaction layer*, [TRL 4]
- Generation of the real time architecture, incl exec p/f tasks, [TRL 4]
- Inclusion of Simulink and SDL/RTDS models, [TRL 4]
- Connection to schedulability analyzers (MAST, Cheddar) [TRL 3]

Use cases of prototypes

Verify that the generic specification and architecture are feasible and correct, e.g.

- feasibility of the software bus concept
- assesment of a SOIS interface with an actual C API,
- support SAFI for feasibility of SOIS DAS/DVS/EDS
- completeness of component model concept (e.g. data model)

Contribute to the overall Savoir validation approach

- Rapid prototyping of software for RASTA different configuration, processors, partitioning or not, multicore
- Verify functional chains performance such as advanced descent and landing AOCS/GNC on Rasta, or centralised SW architecture for Star Trackers
- Support projects such as Galileo FDIR modeling
- Implement use cases of FDIR for RASTA+

PRODUCTS

RTEMS [TRL6]

Subset of the baseline RTEMS 4.8.0

Quality level:

- Galileo Software Standards (Engineering, Configuration Management, Dependability, Product Assurance, Software Reuse and ISVV) Development Assurance Level – B
- SPEC/SPEC1 evaluation by TUV Rheinland InterTraffic GmbH
- ISVV by CAPTEC - Computer Applied Techniques (http://www.captec.ie/)
- ISVV by SPATIOIT - Soluzioni Informatiche s.a.s (http://www.spazioit.com/)
  - Statement Coverage 100% for C and Assembly Code for LEON 2 and LEON 3 Boards and ERC32, LEON 2 and LEON3 simulators
  - Decision Coverage 100% for C and Assembly Code for LEON 2 and LEON 3 Boards and ERC32, LEON 2 and LEON3 simulators

Users

- smallGEO
- Galileo
- Sentinel2
- IXV
- IMA for Space
- EarthCare

http://rtemscentre.edisoft.pt

Partitioning kernels

3 candidates in IMA-SP: PikeOS, AIR, Xtratum

→ The strategy is to keep a commercial and an open source solution

For open source, need to organize the community of stakeholders

→ Set up the governance of Partitioning kernels (users, maintainer, funding support, steering and change control boards, etc…)

→ Seek R&D support e.g. GSTP

In particular, there is a discussion about bringing Xtratum to a level of a product

- Interest from CNES and Astrium for partitioning

Note: the IMA kernel solutions may be used for security and multicore

Contact: James.Windsor@esa.int

PLANNING AND FUTURE WORK

Software tentative planning

ID	Task Name	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017
21	SOFTWARE	02/01	SOFTWARE									31/12	
22	Execution Platform specification		01/01						31/12				
23	Execution Platform prototype				01/01						31/12		
24	ECSS PUS update					03/01				31/12			
25	PUS on SOIS						02/01			31/12			
26	OBCP specification				01/01				31/12				
27	OBCP prototype						02/01			31/12			
28	OBCP product							01/01			31/12		
29	Boot Software specification				01/01				31/12				
30	Software bus specification	02/01							31/12				
31	Software bus prototype (TASTE)		01/01					31/12					
32	Functional chains specifications					03/01				31/12			
33	Functional chains prototypes						02/01				31/12		
34	IMA for Space specifications			01/01					31/12				
35	IMA for Space prototype				01/01				31/12				
36	IMA for Space executives (PikeOS...				01/01							31/12	
37	Security specifications				01/01				31/12				
38	Security prototype					01/01			31/12				
39	Demonstrator Execution Platform		01/01								31/12		
40	Demonstrator On-Board Software		01/01									31/12	

Savoir[-Faire] Agenda

Savoir-Faire

- Review the technical documentation (COrDeT2, OSRAc, SISTORA, etc)
- Devise on deployment of the generic documents in projects
- Confirm the R&D roadmap
- Advise on SAFI
- Investigate component models and tools
- Partitioning and separation kernels selection

Savoir

- Security implementation strategy to be consolidated, in context of the overall space-ground security
- IMA-SP to be extended towards hardware support (I/O board, time-triggered bus, multicore)

Contact

Feedback: savoir@esa.int