Final Presentation

VERICOCOS STUDY

VERIFICATION OF COMPUTER-CONTROLLED SYSTEMS

7th December 2016

Speakers:
- **Elena Alaña (GMV)**
- **Richard Melvin (SCISYS)**

CONTENTS

- Introduction and objectives
- Study activities:
  - Space user needs
  - Survey of modelling languages and tools
  - Tool development framework
  - Training material
- Conclusions and future work

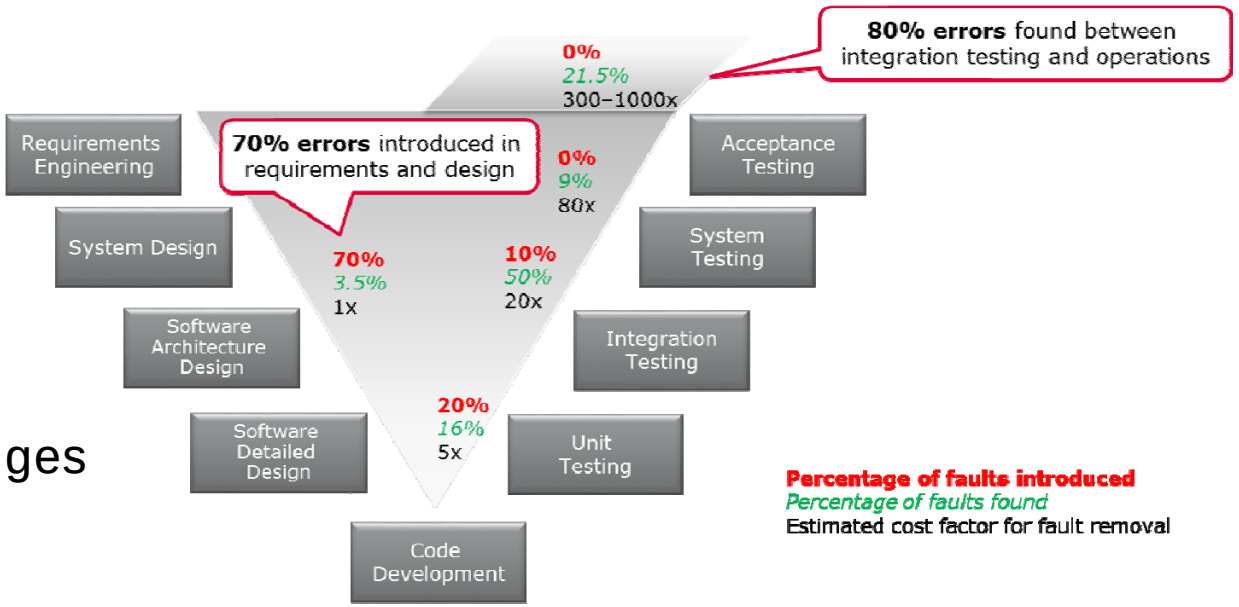
VERICOCOS - Final Presentation

INTRODUCTION AND OBJECTIVES

INNOVATING SOLUTIONS

MOTIVATION BEHIND VERICOCOS

- Where are faults introduced?
  - Most errors are introduced during requirements and design phases
- Source of Requirements errors:
  - Bad requirements quality
    - Ambiguous
    - Incomplete
    - Incoherent
    - Inconsistent
    - Erroneous
  - Requirements changes
  - Design errors



MOTIVATION BEHIND VERICOCOS

- Good requirements management key factor for spacecraft development
- The most problematic sources of errors can be avoided or partially solved by improving the quality of the input specifications
- Requirements formats:
  - Textual requirements
  - Pseudo-code
  - Sequence diagrams
  - State machines

Behavioural modelling provides a more complete specification of requirements

VERICOCOS GOALS

1. Confirm the applicability and subsequently open the door to the generalized use of **state machines** to the specification, design, verification and implementation of on-board software

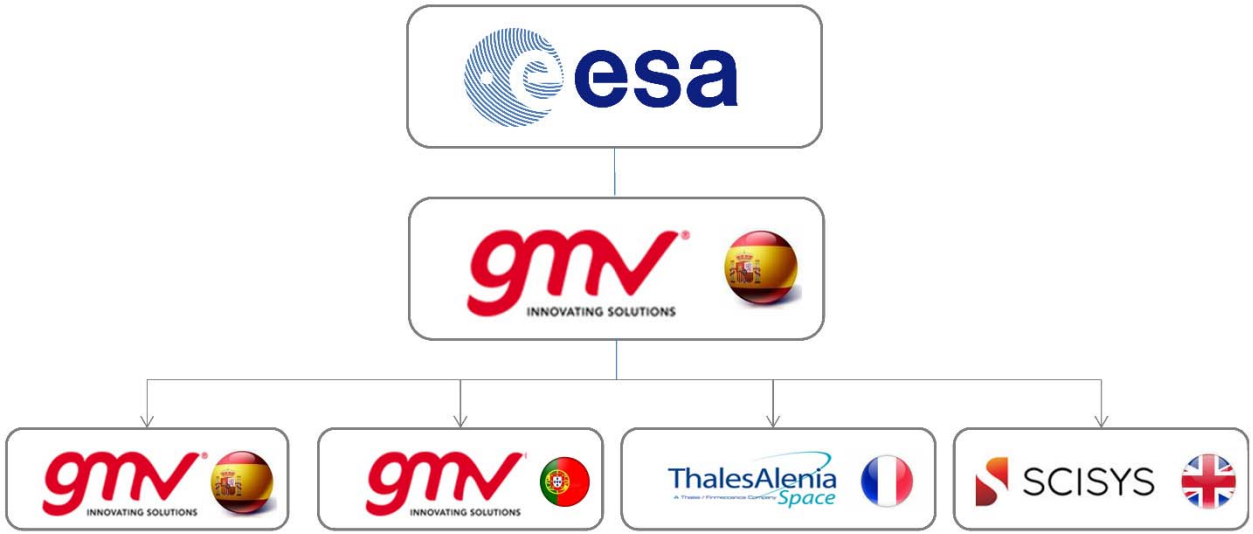
2. Apply tools on a **space-representative case study**, and extend the tools if necessary to fit existing processes

3. Further develop and test behavioural and data modelling tools to support the implementation of **electronic data sheets**

4. Provide **training material** with the result of the study to disseminate the acquired knowledge and know-how on the topic of behavioural modelling

CONSORTIUM

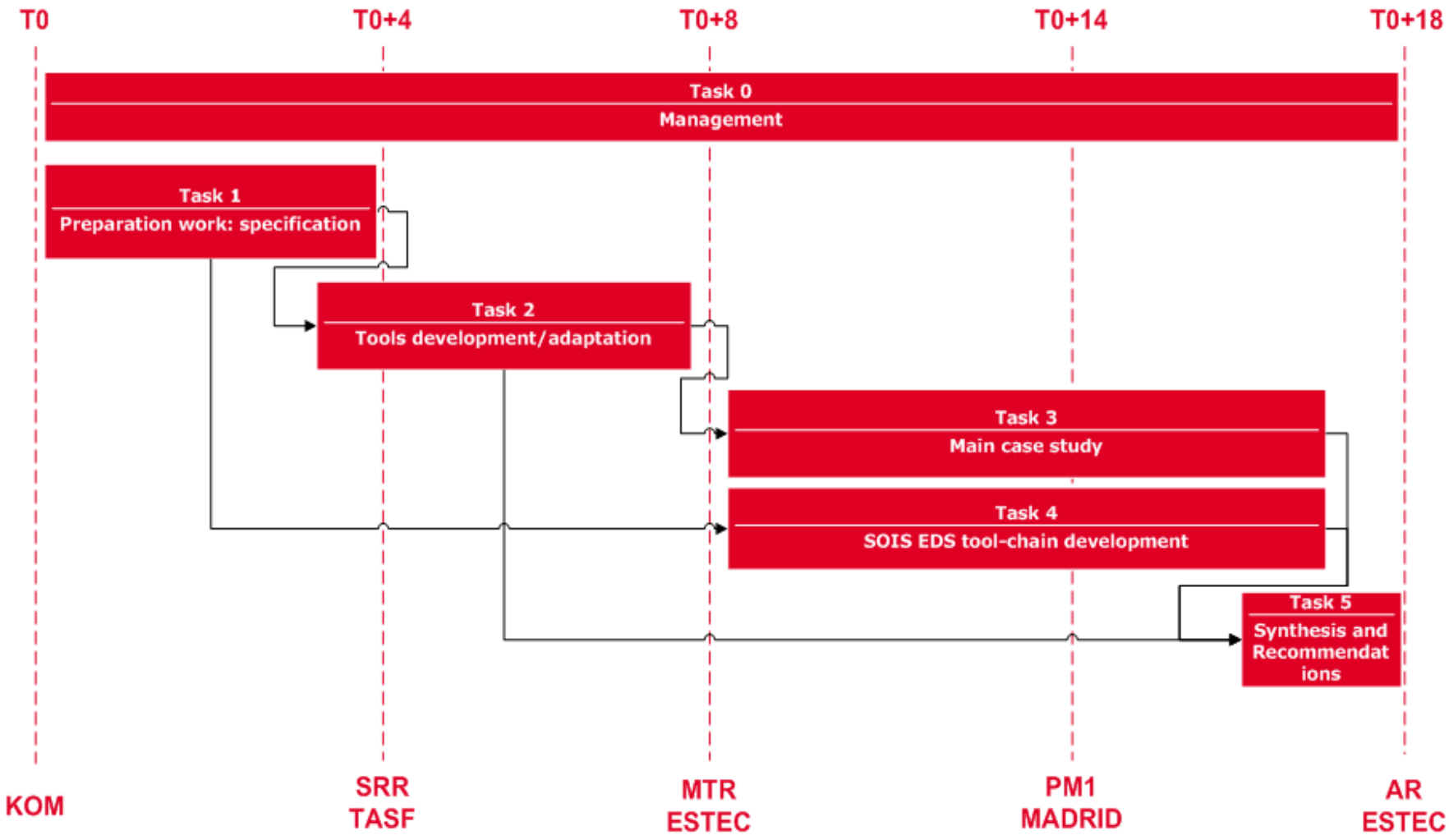
- ESA TO: Maxime Perrotin
- Consortium:
  - GMV Aerospace and Defence, S.A.U. (Prime Contractor)
  - GMV Skysoft S.A. Portugal
  - Thales Alenia Space – France
  - SCISYS UK Ltd



TECHNICAL SHARING

COMPANY	TECHNICAL RESPONSIBILITIES
GMV-ESP	▪ Project management ▪ Technical coordination and responsible for: o The **design of the tool architecture** o The implementation and analysis of the **case study** o **Dissemination** activities o Project **global assessment** and synthesis of the results
TASF	▪ Analysis of behavioural modelling on ESA operational projects ▪ Analysis of **modelling languages and tools** for behavioural modelling ▪ Preliminary **specification of the case study** ▪ **Global assessment** as System Integrator
GMV-POR	▪ **VERICOCOS tool chain**: o Tool development and integration o Verification o Generation of documentation and guidelines to use the tool in the case study
SCISYS	▪ **SOIS EDS tool chain**: o Analysis of the ESA internal SOIS EDS study o Specification and extension of the ESA SOIS EDS tool chain o Validation of the EDS concept on a real test case

WORK FLOW



VERICOCOS - Final Presentation

STUDY ACTIVITIES

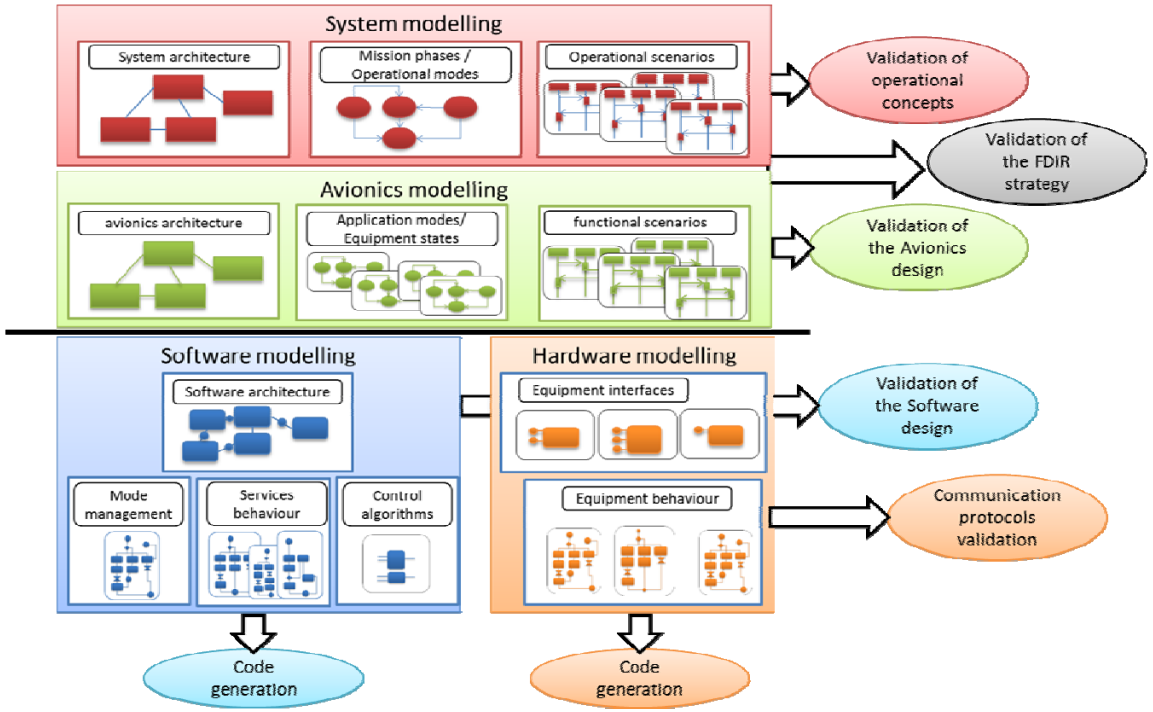
INNOVATING SOLUTIONS

STUDY ACTIVITIES

1. Specification of **Space User Needs** for behavioural modelling
2. Survey of modelling **Languages and Tools**
3. Implementation of a **Tool Development Framework**
   - VERICOCOS Toolchain
   - SOIS EDS Toolchain
4. Elaboration of **Training Material**

SPACE USER NEEDS (1/2)

- Evaluation of state machines for developing embedded SW:
  - Improves the requirements quality (correctness, completeness, coherency)
  - Benefits are not the same at each development phase



SPACE USER NEEDS (2/2)

- Specification of **user needs for behavioural modelling**
  - Modelling levels:
    - System modelling level
    - Avionics modelling level
    - Software modelling level
    - Hardware/Equipment modelling level
  - Technology agnostic
  - Examples:

ID	User Need	Level
[MLR.2]	The VERICOCOS toolchain shall allow to model the system mission phases.	System
[VOR.3]	The VERICOCOS toolchain shall allow to associate a state machine to the software components (implementation) having a behaviour designed with state machine.	Software
[CGR.1]	The VERICOCOS toolchain shall allow to generate code from state machines and procedure models associated to software components in the software detailed design model.	Software

STUDY ACTIVITIES

1. Specification of **Space User Needs** for behavioural modelling
2. Survey of modelling **Languages and Tools**
3. Implementation of a **Tool Development Framework**
   - VERICOCOS Toolchain
   - SOIS EDS Toolchain
4. Elaboration of **Training Material**

SURVEY LANGUAGES AND TOOLS (1/2)

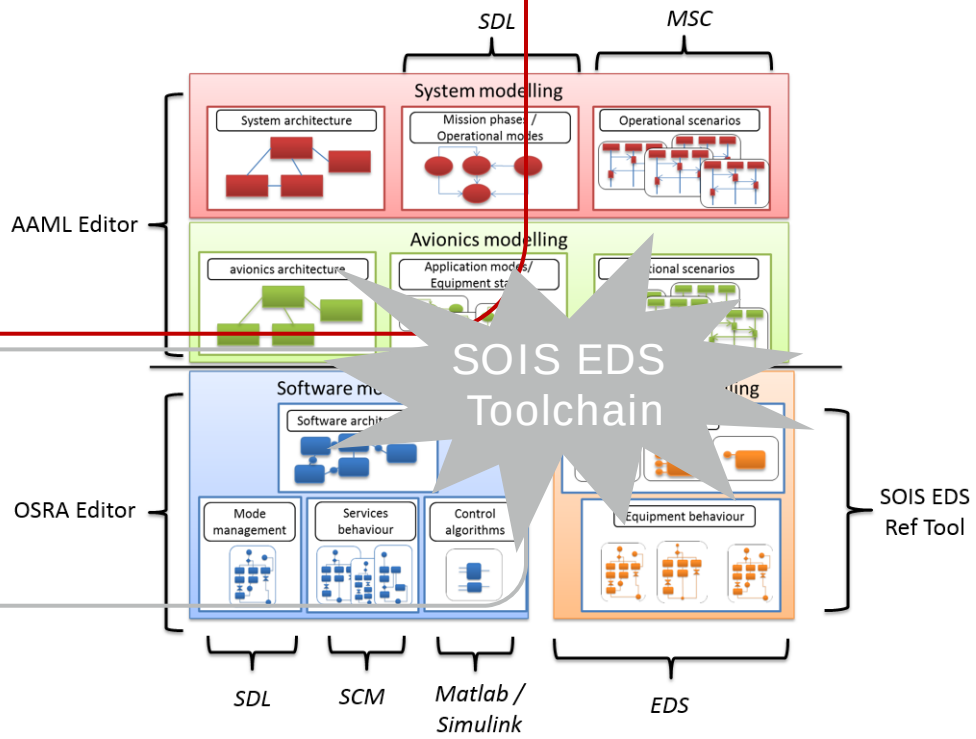
- **Goal**:
  - Find languages and tools suitable for most of the space user needs
- Survey of **languages and tools** that can cover those needs
  - Languages:
    - Focused on SDL
    - Background concepts of other languages: UML, AADL, SysML, etc.
    - Rated with respect to the modelling needs:
      - Compliance with respect to System/SW/HW modelling needs, aligned on a standard, non-proprietary language, graphical notation, etc.
  - Tools:
    - Existing modelling tools are described such as TASTE, OpenGEODE, AAML Editor and Capella Editor
    - Rated according to the tool needs:
      - Maturity level, maintainability and long-term support, user manual quality, scalability, separation of concerns, etc.

SURVEY LANGUAGES AND TOOLS (2/2)

- Selection:
  - System and Avionics modelling:
    - Languages: AAML and SDL
    - Tools: AAML Editor, MSC Editor and OpenGEODE
  - Software modelling:
    - Languages: SCM and SDL
    - Tools: OSRA Editor and OpenGEODE
  - Hardware modelling:
    - EDS with TASTE integration for code generation

VERICOCOS Toolchain

SOIS EDS Toolchain

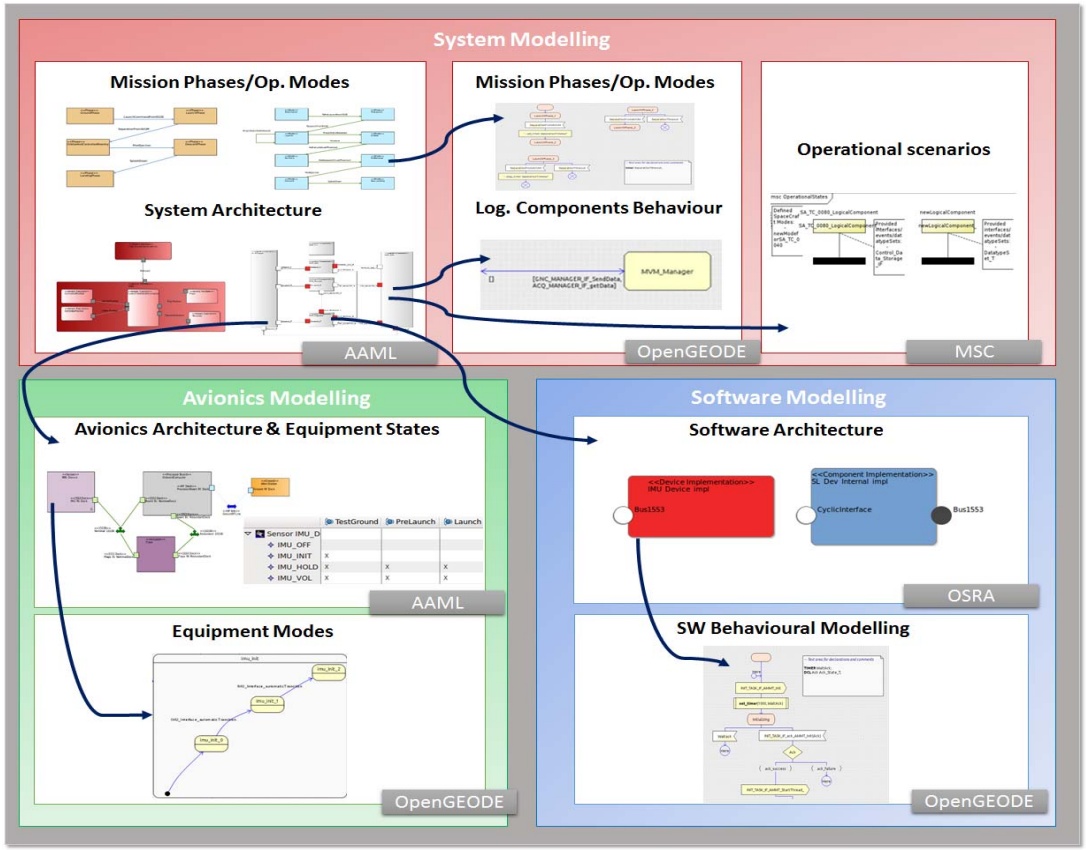


STUDY ACTIVITIES

1. Specification of **Space User Needs** for behavioural modelling
2. Survey of modelling **Languages and Tools**
3. Implementation of a **Tool Development Framework**
   - VERICOCOS Toolchain
   - SOIS EDS Toolchain
4. Elaboration of **Training Material**

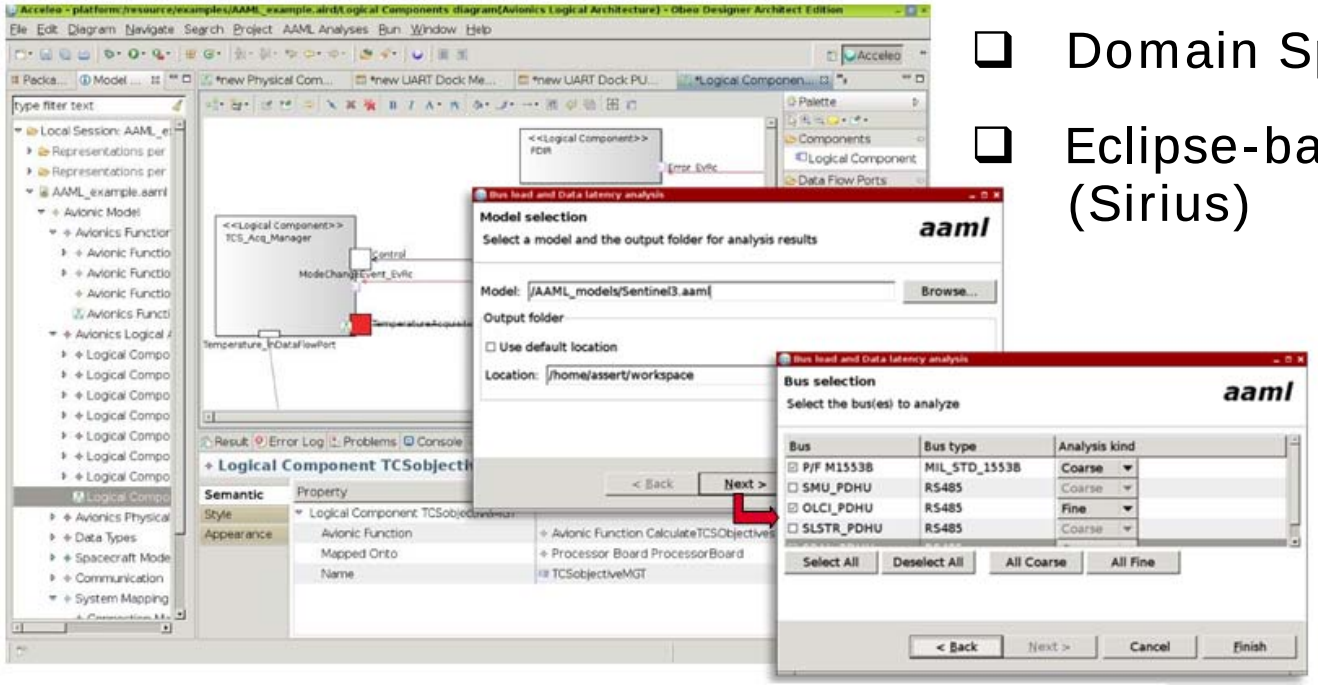
VERICOCOS TOOLCHAIN

- Modelling levels:
  - System & Avionics
  - Software
- Based on:
  - Model-Based Development
  - Modelling views



SYSTEM AND AVIONICS MODELLING

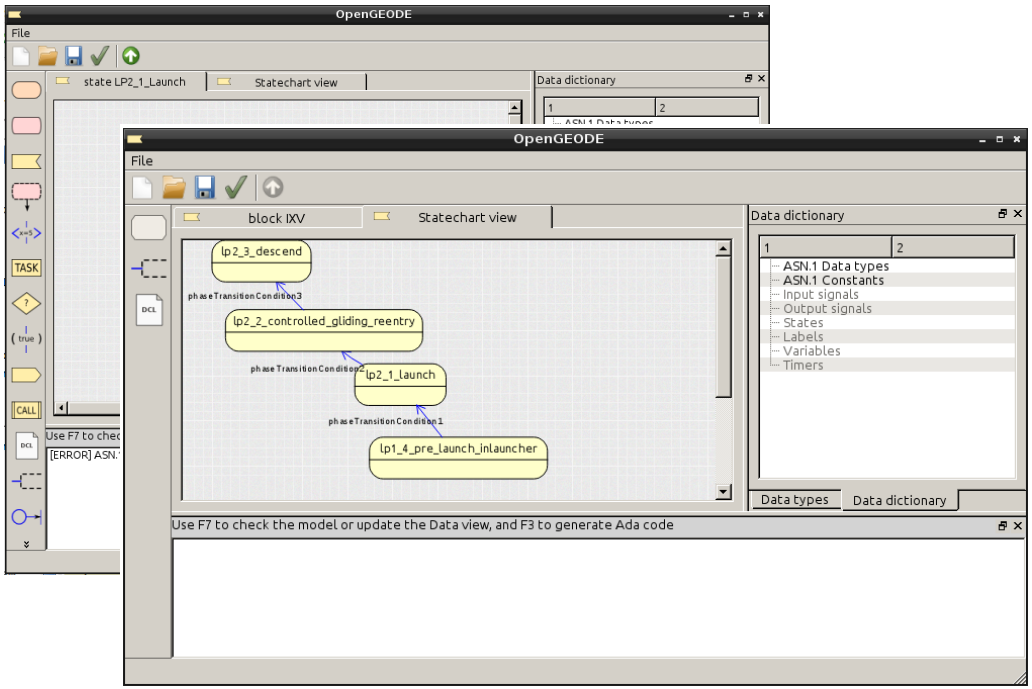
- The **AAML Toolset (Avionics Architecture Modelling Language)** aims at advancing the avionics engineering practices towards a model-based approach
- Coarse- and fine-grained specification of the avionics architecture and non-functional properties



- Domain Specific Language
- Eclipse-based toolset (Sirius)

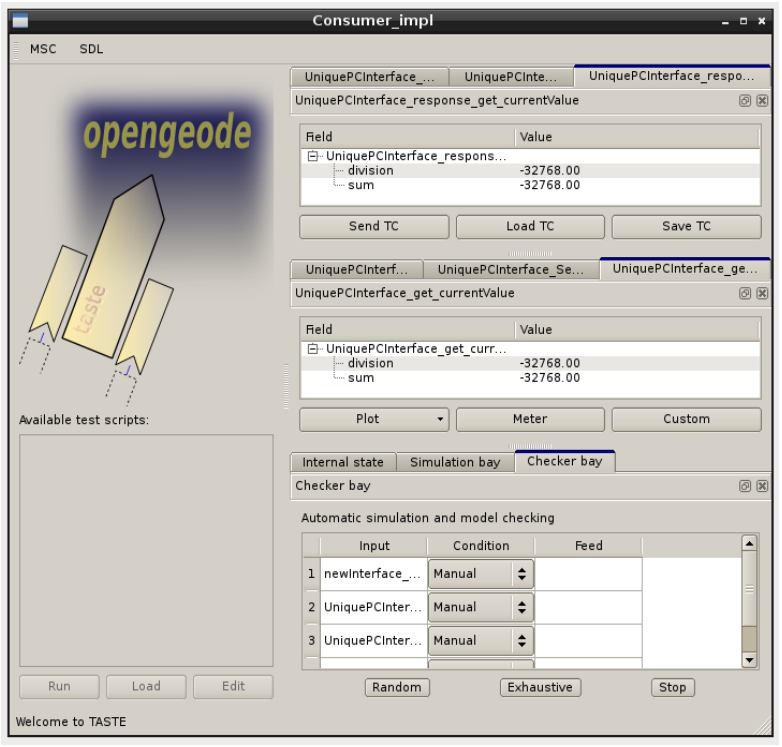
OPENGEODE TOOL

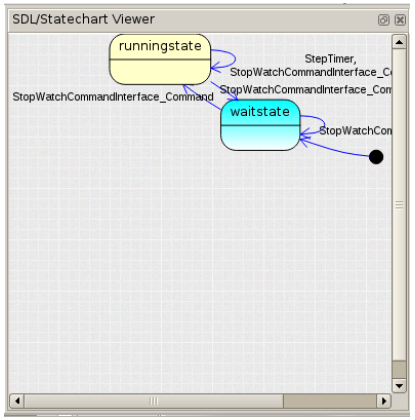
- OpenGEODE is an SDL editor (Specification and Description Language) part of ESA **TASTE (The ASSERT Set of Tools for Engineering)**
- SDL is a mature ITU standard, formal behaviour specification of complex systems through state machines
  - Behavioural modelling through rich state machines
  - Textual and graphical notations
- OpenGEODE implements a safe SDL subset for OBSW development

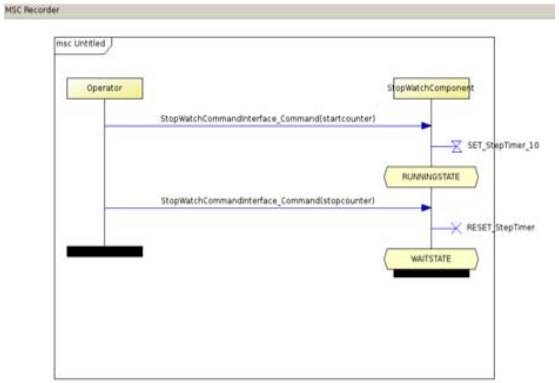


OPENGEODE FEATURES

- SDL Editor
  - State machine viewer
- SDL Checker
  - Correctness & safety checks
- Animation and simulation
  - Inject signals
  - Debugging capabilities
  - Store simulated behaviour as sequence diagram (MSC)
- Code generation
  - C and Spark Ada

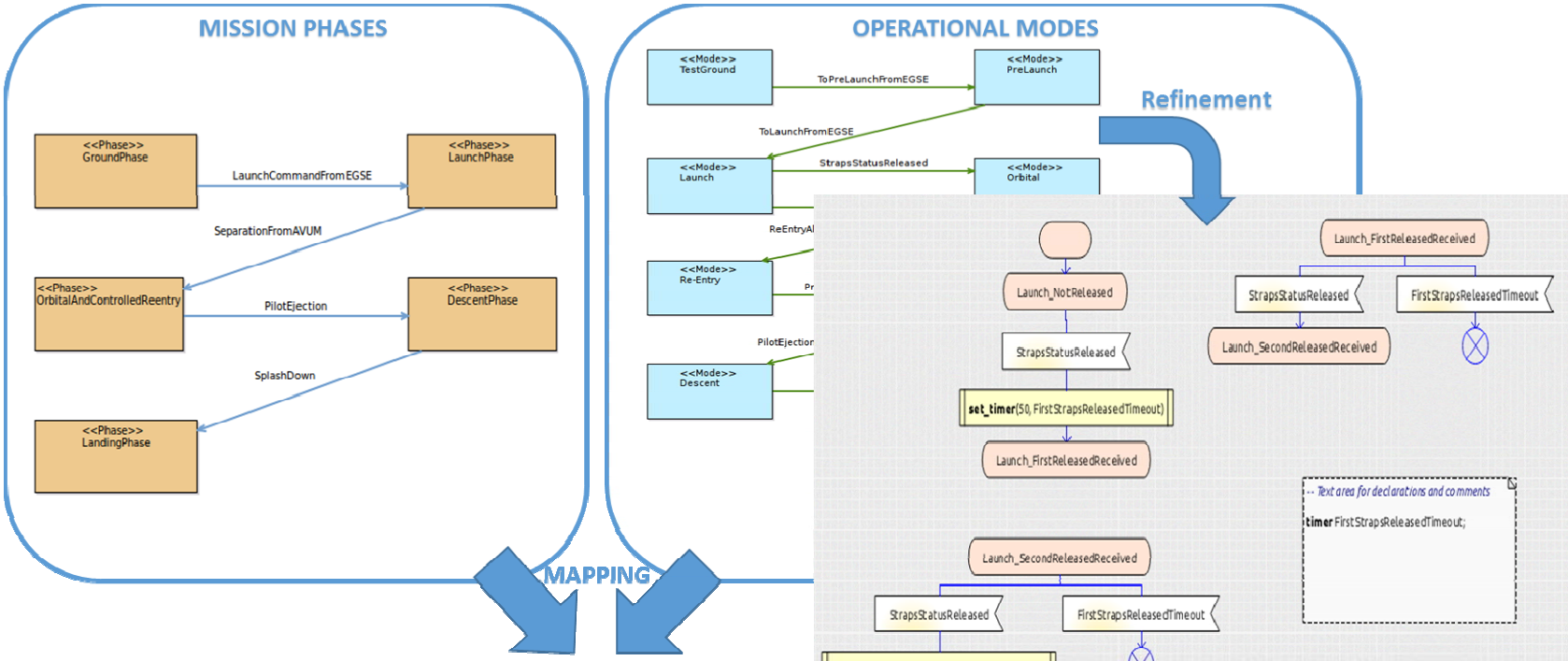






UPDATES AT SYSTEM & AVIONICS LEVEL

- Formalization of **mission phases** and **operational modes:**

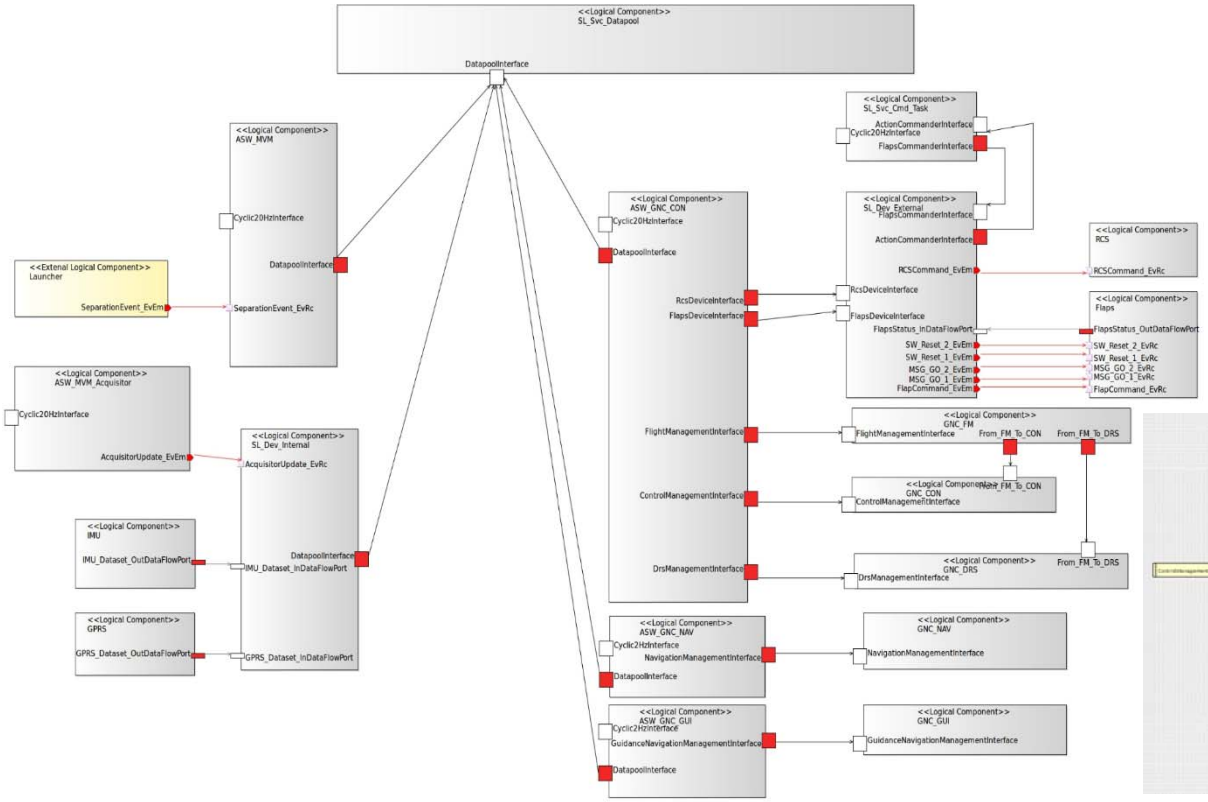


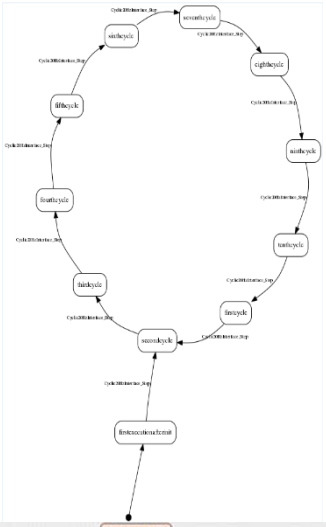
IXV mapping: Mission Phases - SpacecraftModes Mapping Table

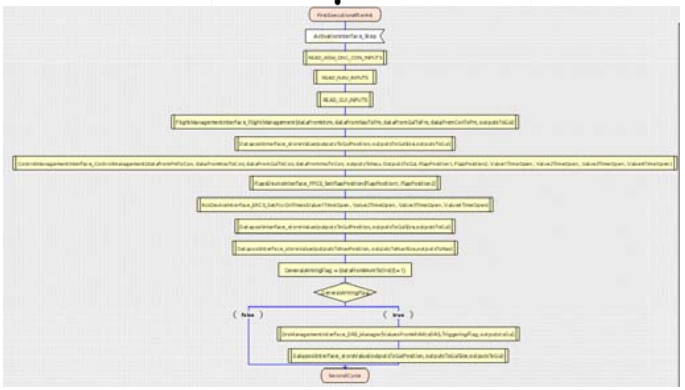
	SpaceCraft Modes
GroundPhase	Mode TestGround, Mode PreLaunch
LaunchPhase	Mode Launch
OrbitalAndControlledReentry	Mode Orbital, Mode Re-Entry, Mode PreRelease
DescentPhase	Mode Descent
LandingPhase	Mode Sea-Landing

UPDATES AT SYSTEM & AVIONICS LEVEL

- Logical architecture
  - Dynamic behaviour of logical components (OpenGEODE)
    - Functionality subset used in the mission



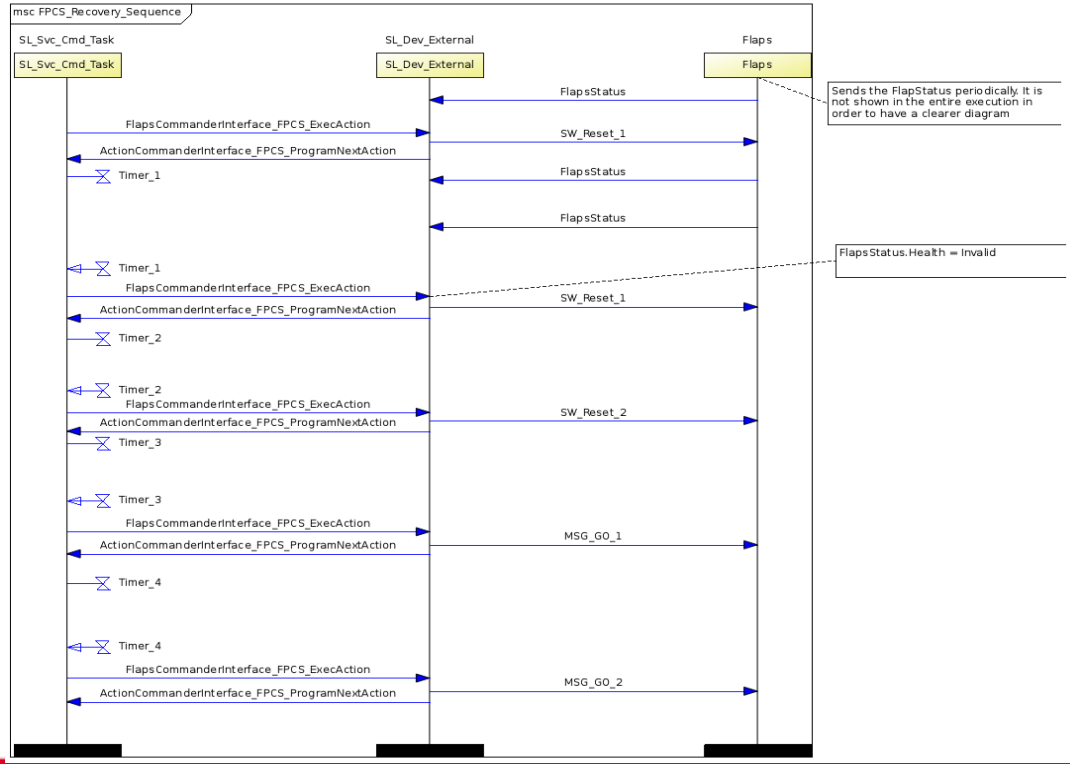




UPDATES AT SYSTEM & AVIONICS LEVEL

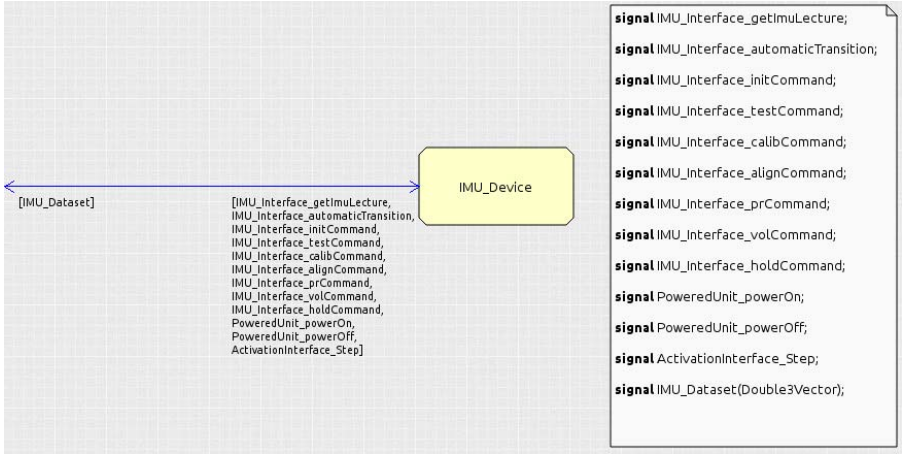
❑ Logical architecture

- Inter-component scenarios (MSC Editor):
  - Operational scenarios (nominal mission sequences)
  - FDIR scenarios (fault detection and recovery sequences)



UPDATES AT SYSTEM & AVIONICS LEVEL

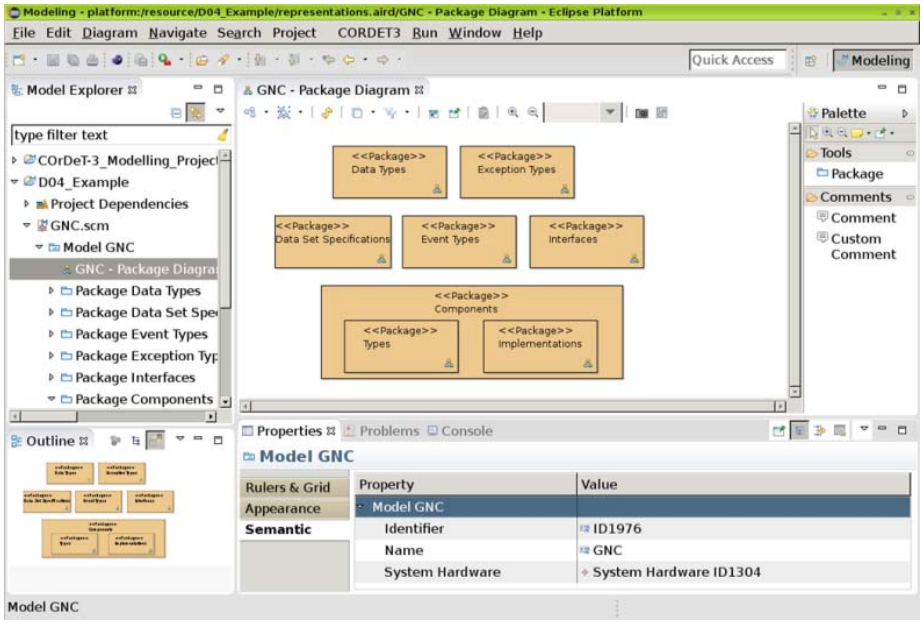
- **Physical architecture**:
  - Behaviour of physical devices
    - Complete dynamic aspects of the device
  - Definition configuration
    - Definition of device states
    - Can be mapped to operational modes



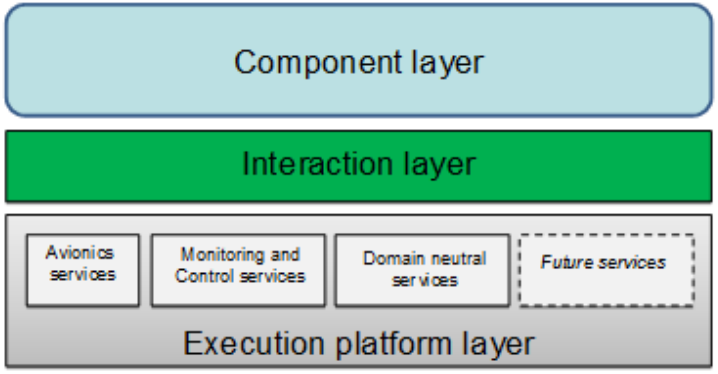
	TestGround	PreLaunch	Launch	Orbital	Re-Entry	PreRelease	Descent	Sea-Landing
Sensor IMU_Device								
IMU_INIT		X	X					
IMU_HOLD		X	X					
IMU_ALIGN		X	X					
IMU_OFF	X							
IMU_VOL				X	X	X	X	
IMU_CALIB								
IMU_PR								
IMU_TEST	X							
Actuator Flaps								
ON		X	X	X	X	X	X	
OFF	X							
FAILED				X	X	X	X	

SOFTWARE MODELLING

- The ESA **OSRA Editor (On-board Software Reference Architecture)** assists the software architect and suppliers to design, develop, integrate, and generate the final executable
- Methodological approach for developing Space Applications (See OSRA Specification and Rational for further details)

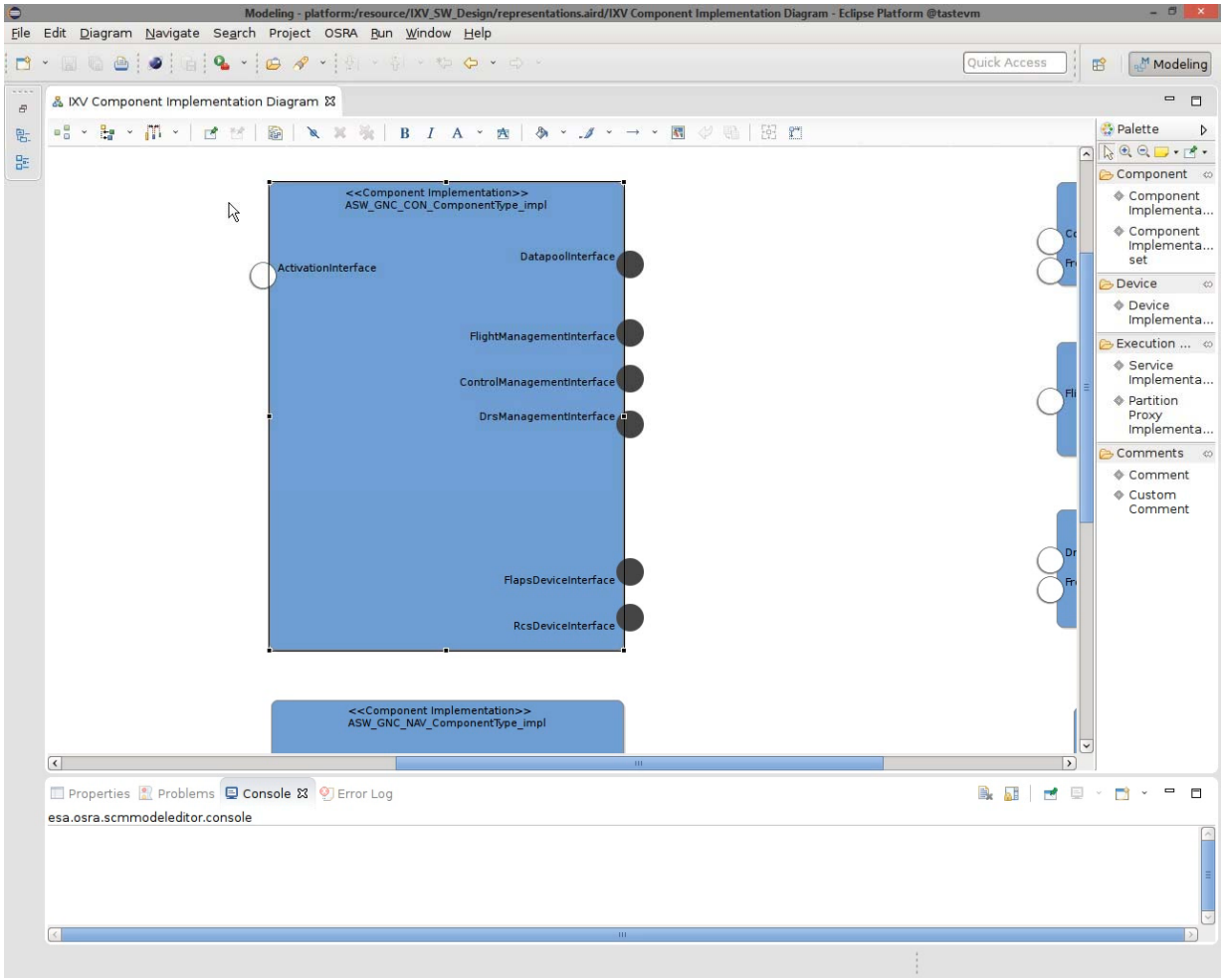


- Domain Specific Language (Space Component Model)
- Eclipse-based toolset (Sirius)

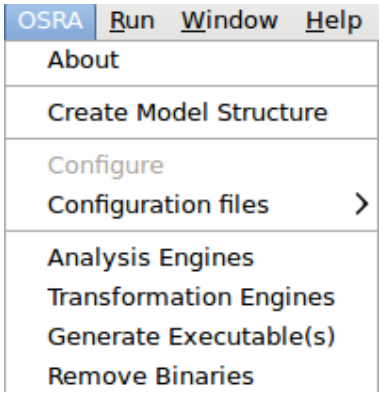


UPDATES AT SOFTARE LEVEL

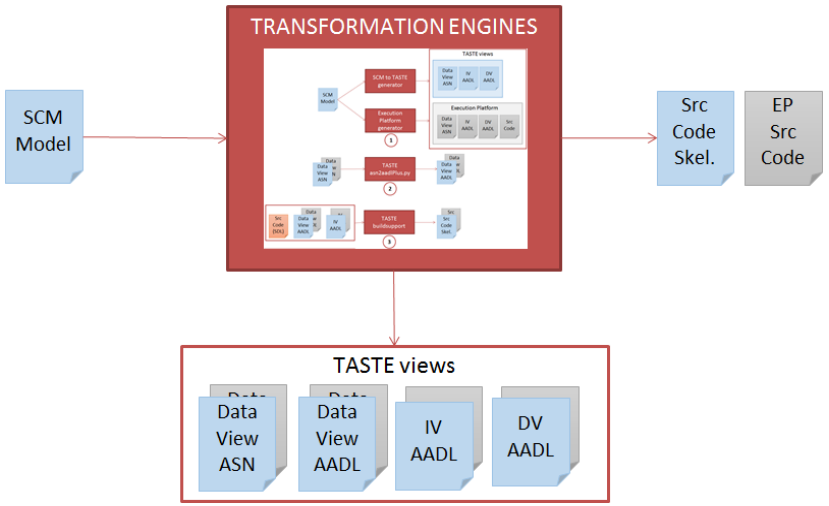
- Specification of the behaviour of component implementations
  - SDL language
  - OpenGEODE tool
- All communication entities defined in OSRA for the component, are available in the SDL diagram
- Datatypes translated into ASN.1 notation
- Code generation from SDL state machines
- Document generation



CODE GENERATION

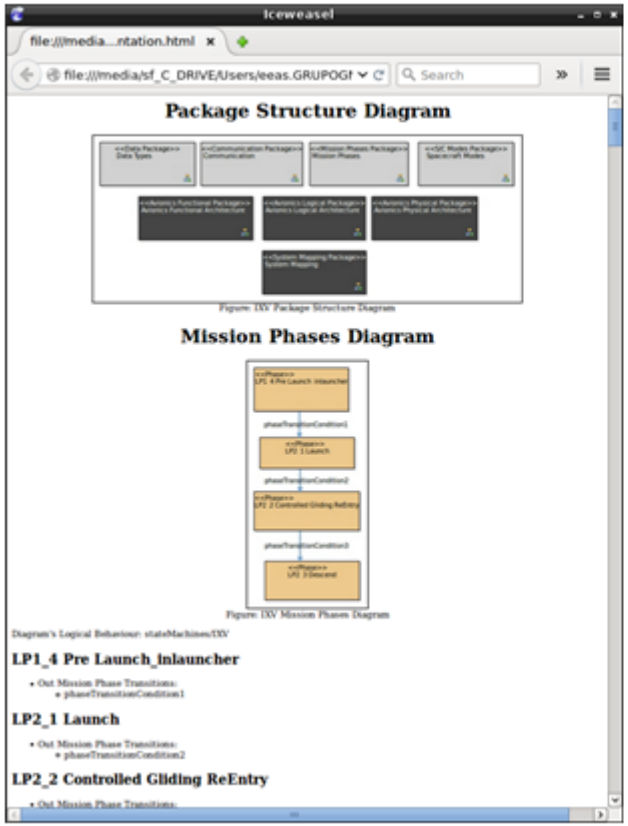


- Code generation from the OSRA menu
  - Transformation Engines": generates the source code skeletons of the OBSW
  - "Generate Executables" produces the OBSW binary(ies)
- The programming language is specified in the Components Implementation:
  - Ada
  - C
  - SDL
    - Ada language skeletons
    - Skeletons automatically filled in with the source code of the associated state machines following this process

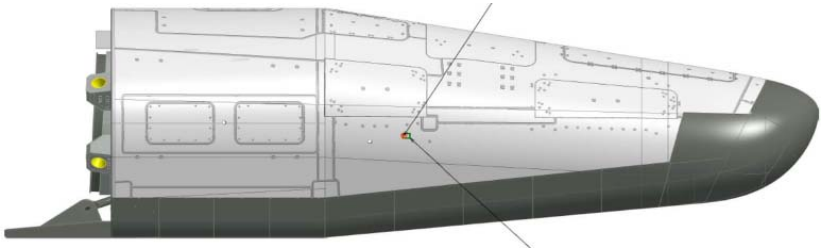


DOCUMENT GENERATION

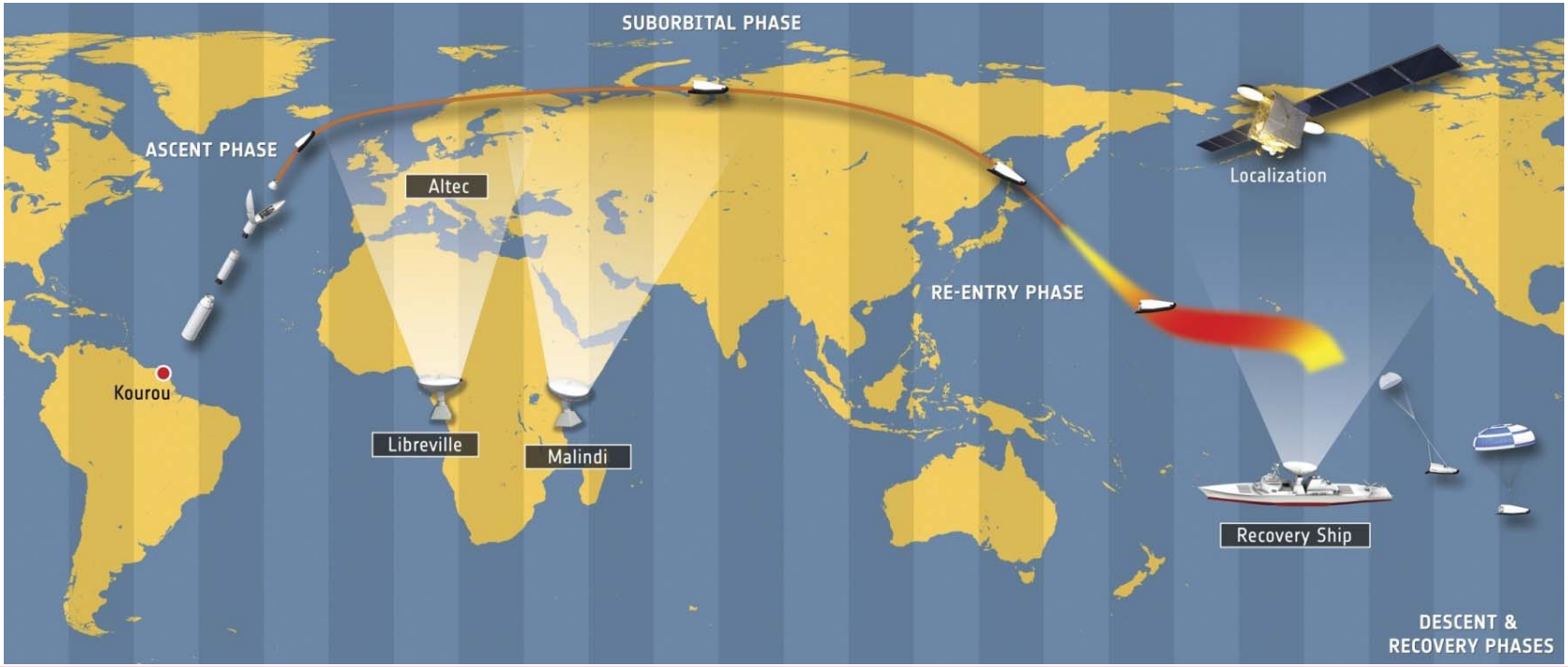
- Proof-of-concept
- AAML and OSRA generates HTML, with all diagrams and tables
- To be tailored to specific documentation needs



INTERMEDIATE EXPERIMENTAL VEHICLE

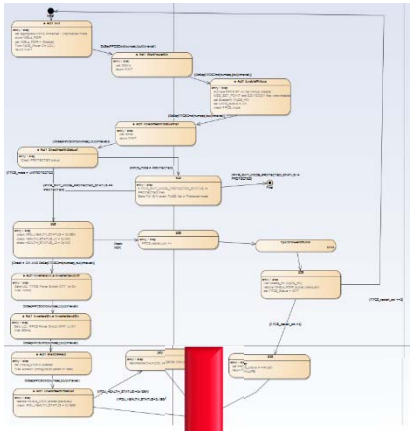


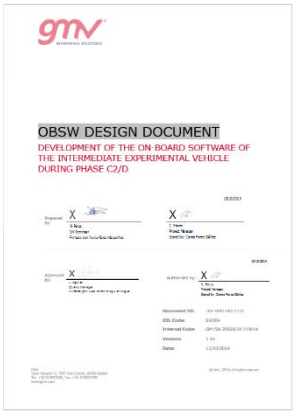
- Space vehicle to experiment on **atmospheric re-entry**
- Duration: around 100 min
- Fully-automated sub-orbital flight
- No pilot, passengers, nor TC
- TM monitored from ground, and recorded in vehicle
- Phases: Ascent, Orbital, Re-entry, Descent, Sea-landing



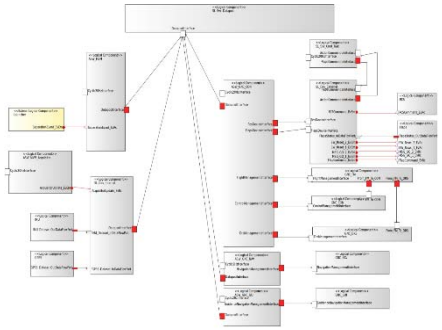
IXV CASE STUDY

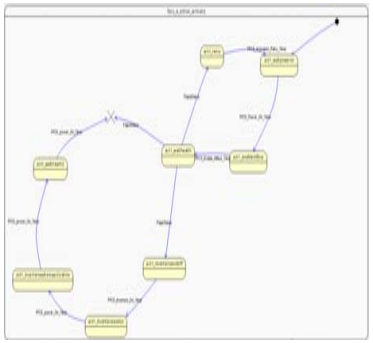
IXV DATA

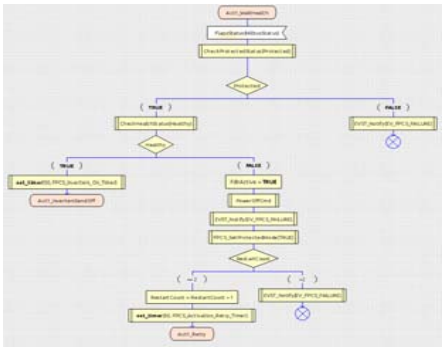




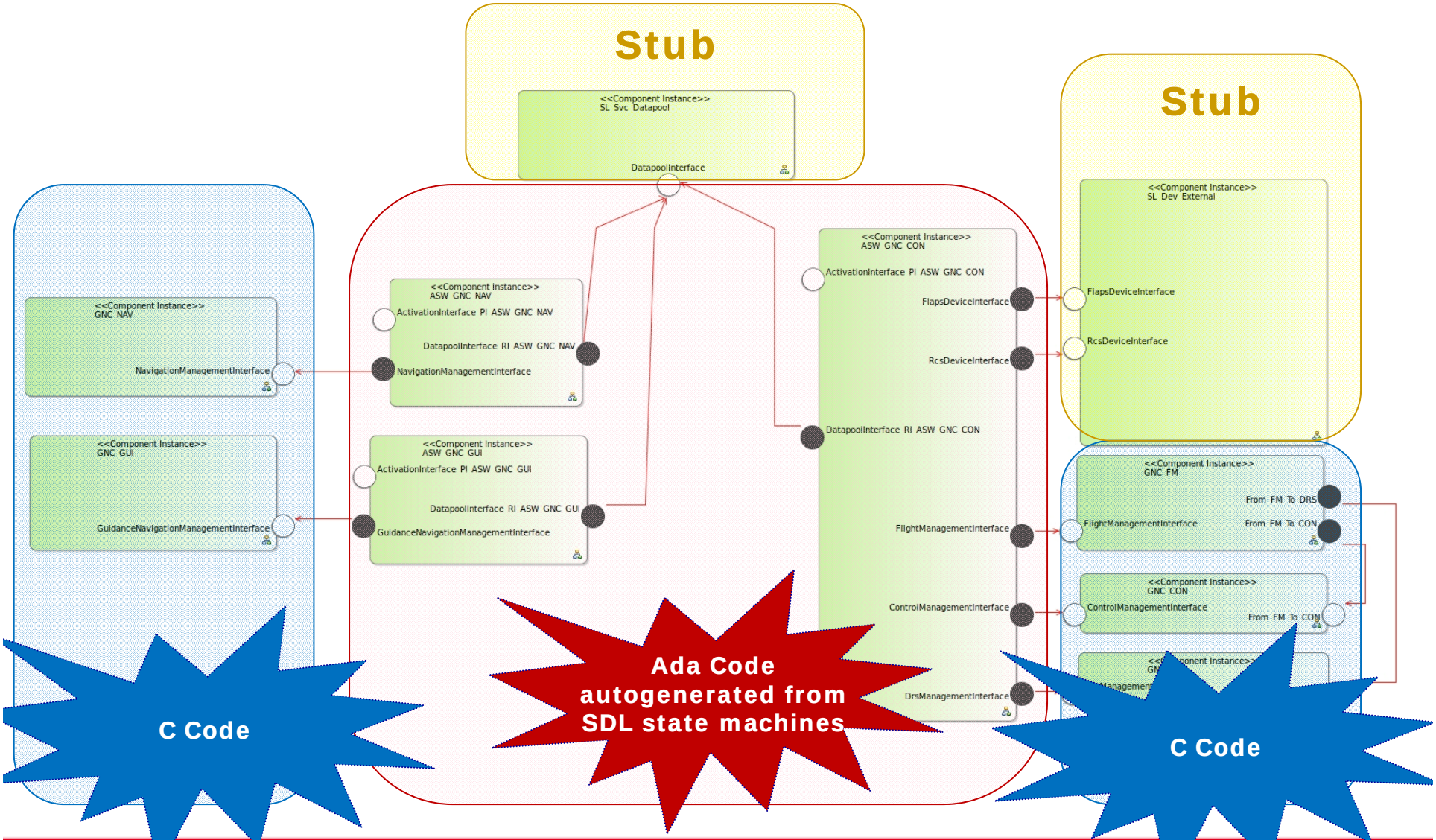
VERICOCOS MODEL





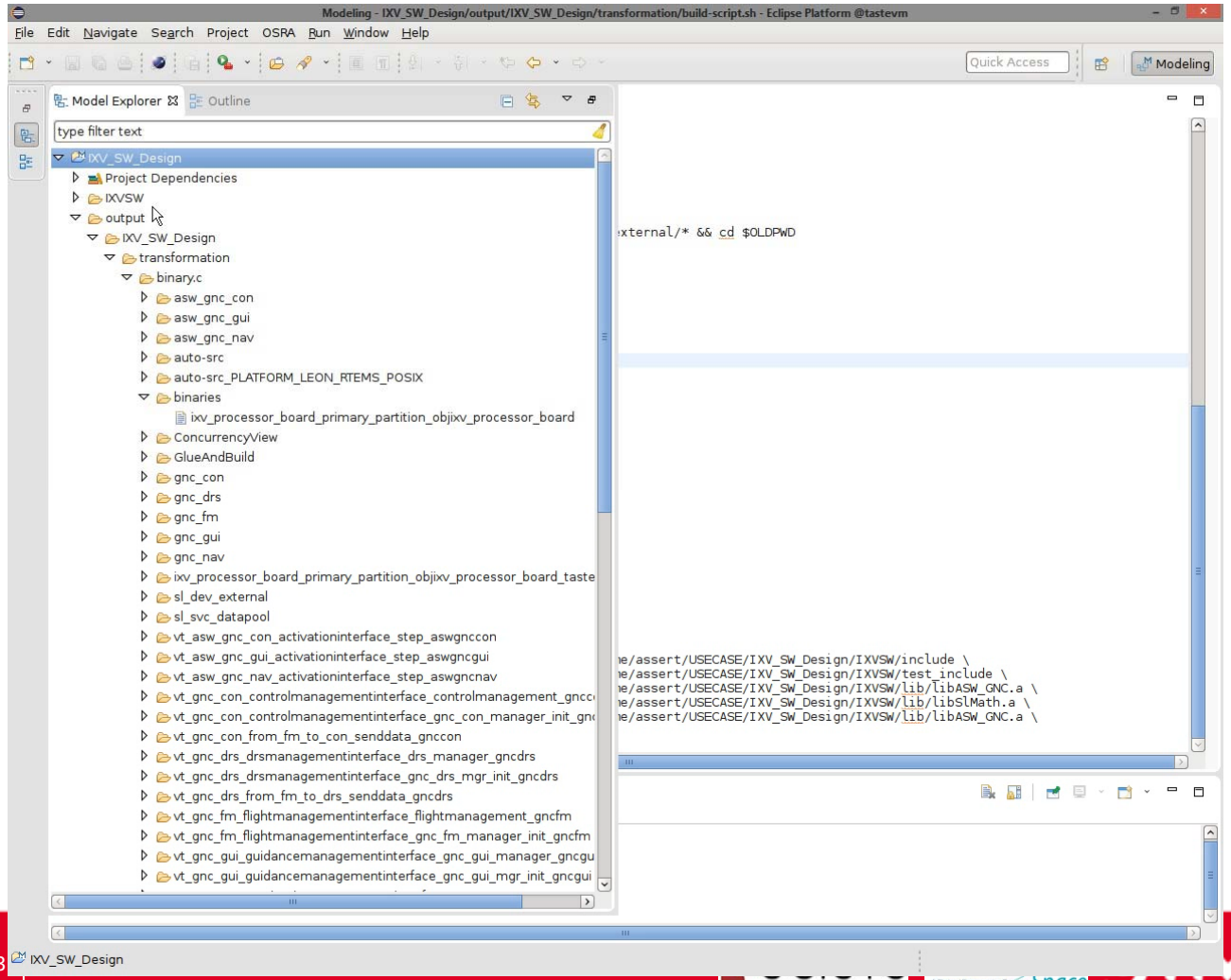


INTEGRATION WITH REAL CODE



GENERATION OF EXECUTABLE

- Configuration of the TASTE Processor:
  - IXV_Processor_Board : PROCESSOR ocarina_processors_leon::leon.rtems_posix;
- TASTE used for:
  - Code skeletons generation
  - Executable generation



RESULTS

- **Unambiguous and precise semantics** due to formal languages ✓
  - Early error detection when creating SW from scratch
  - Designer is forced to think about all cases, including error sequences
- Design maintained at **model level**, including dynamic behaviour ✓
  - Mappings ensure consistency
  - Document and code generation
- **Early verification and validation** at design phase ✓
  - Static correctness checks
  - Animate state machines
  - Possibility to improve analysis engines with behavioural information not explored in the case study
- **Process in line with life cycle of space projects** ✓
- Tools must evolve to be applicable to other space projects ⚠

STUDY ACTIVITIES

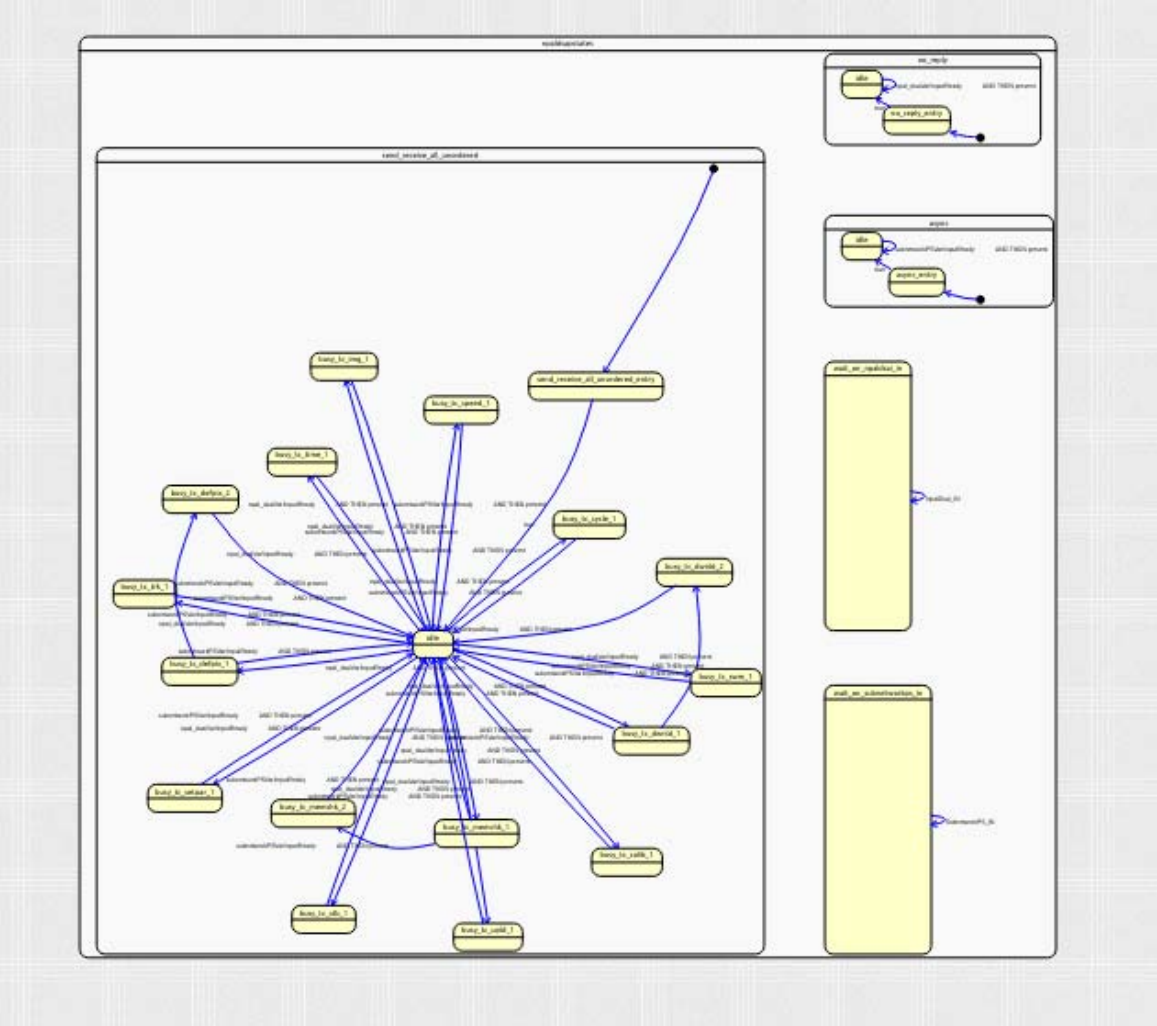
1. Specification of **Space User Needs** for behavioural modelling
2. Survey of modelling **Languages and Tools**
3. Implementation of a **Tool Development Framework**
   - VERICOCOS Toolchain
   - SOIS EDS Toolchain
4. Elaboration of **Training Material**

SOIS EDS TOOLCHAIN (1)

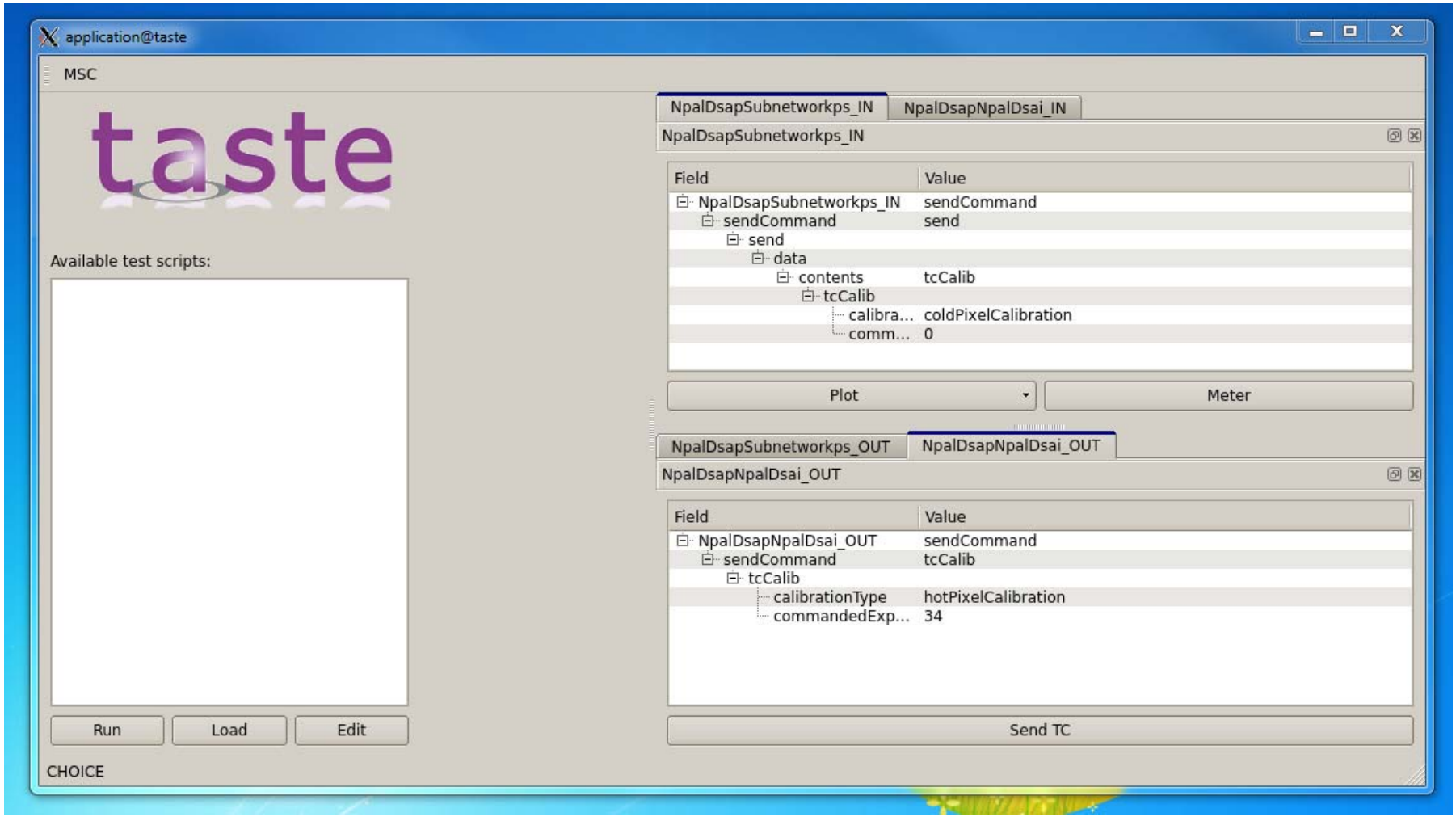
Presenter: SCISYS

- USE of EDS as an input to behavioural modelling
  - Typical case: hardware exists, you want to monitor and control it
    - Other direction not currently addressed
  - ESA's TASTE used as behavioural modelling tool
  - Automatic generation of TASTE model from an EDS
    - ASN.1/ACN data types
    - SDL/PR behavioural model
    - AADL system structure (for a testbench)

TRANSLATED EDS IN OPENGEODE



TASTE TEST BENCH FROM AN EDS



SOIS EDS TOOLCHAIN (2)

- Interoperability Test Data Set for EDS Reference Tooling
  - Necessary step in CCSDS standardisation process
  - All-Pairs generation of artificial test data from schema
  - 20,000 lines of xml
- Update Reference Tooling to fully handle this test data set
- Feed ITDS into TASTE translator
  - TASTE was improved as a result of this activity

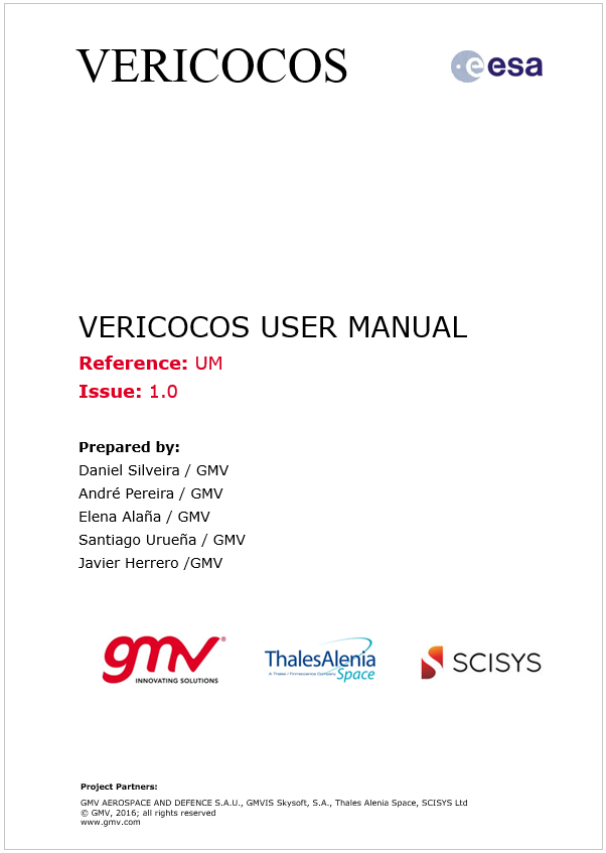
STUDY ACTIVITIES

1. Specification of **Space User Needs** for behavioural modelling
2. Survey of modelling **Languages and Tools**
3. Implementation of a **Tool Development Framework**
   - VERICOCOS Toolchain
   - SOIS EDS Toolchain
4. Elaboration of **Training Material**

TRAINING MATERIAL

- Supports the dissemination of the project results and, ultimately, foster the use of state machines and behavioural modelling in on-board software specification, design, development and validation

  - **VERICOCOS User Manual**
    - Integrated User manual for System, Avionics & Software levels
    - Step-by-Step descriptions


VERICOCOS

VERICOCOS USER MANUAL
Reference: UM
Issue: 1.0

Prepared by:
Daniel Silveira / GMV
André Pereira / GMV
Elena Alaña / GMV
Santiago Urueña / GMV
Javier Herrero /GMV

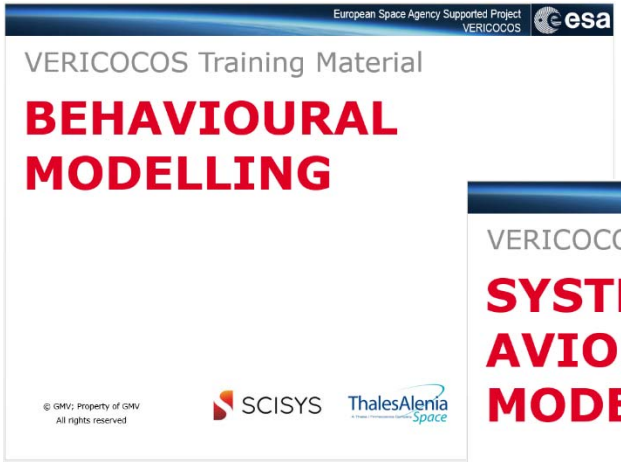
Project Partners:
GMV AEROSPACE AND DEFENCE S.A.U., GMVIS Skysoft, S.A., Thales Alenia Space, SCISYS Ltd

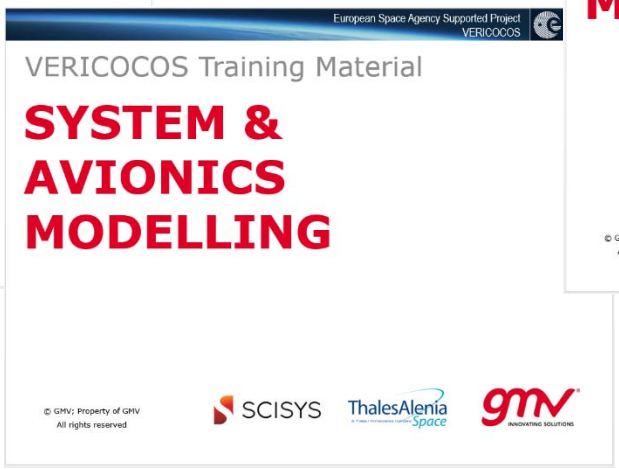
www.gmv.com

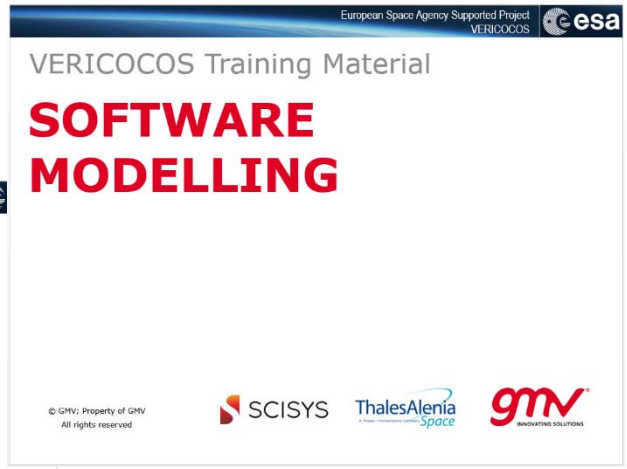
TRAINING MATERIAL

- **Presentations**

Title	Audience	Duration	Comments
Behavioural Modelling Overview	Managers/Engineers	30 min.	Technology independent
System & Avionics Modelling	Engineers	2 hours	Self-training (AAML/OpenGEODE)
Software Modelling	Engineers	2 hours	Self-training (OSRA/OpenGEODE)







TRAINING MATERIAL

- **Website**

  http://vericocos.gmv.com/

  - Public information only
  - Private documentation shall be requested to Maxime Perrotin
  - The website will be updated with new publications

VERICOCOS - Final Presentation

CONCLUSIONS AND FUTURE WORK

INNOVATING SOLUTIONS

BENEFITS

- Model Simulation
- Model Checking
- Information interchange
- Abstraction
- Avoid ambiguities
- Document generation
- Code generation
- SVF
- System tests

ASSESSMENT

- **Achievements**: 
  - VERICOCOS outcomes matches the expectation of the VERICOCOS project
  - The study demonstrates the benefits of behavioural modelling
- **Improvements**: 
  - The adoption of this new approach requires money and time
  - It does not only depend on the technical quality of the toolchain but also on the project management decisions
- **Recommendations**:
  - Dissemination, evolutions of VERICOCOS toolchain, consolidation of EDS process, etc.

FUTURE LINES OF WORK

- Dissemination of the VERICOCOS results
- Evolutions of the VERICOCOS Tool Framework
  - Potential evolutions have been identified:
    - **ID**: Identifier of the open point/future work
    - **Item**: Element (e.g. tool) involved in the open issue
    - **Description**: Brief summary of the capability to be implemented
    - **Priority**: Low/Medium/High
    - **Work-plan** (effort needed): short-term, mid-term or long-term activity
- Example:

ID	Item	Description	Priority	Work-plan
1.	MSC Editor	There is no check in charge of verifying if the operations are used between the correct components.	High	Mid-term

Thank you

VERICOCOS consortium
http://vericocos.gmv.com/

INNOVATING SOLUTIONS