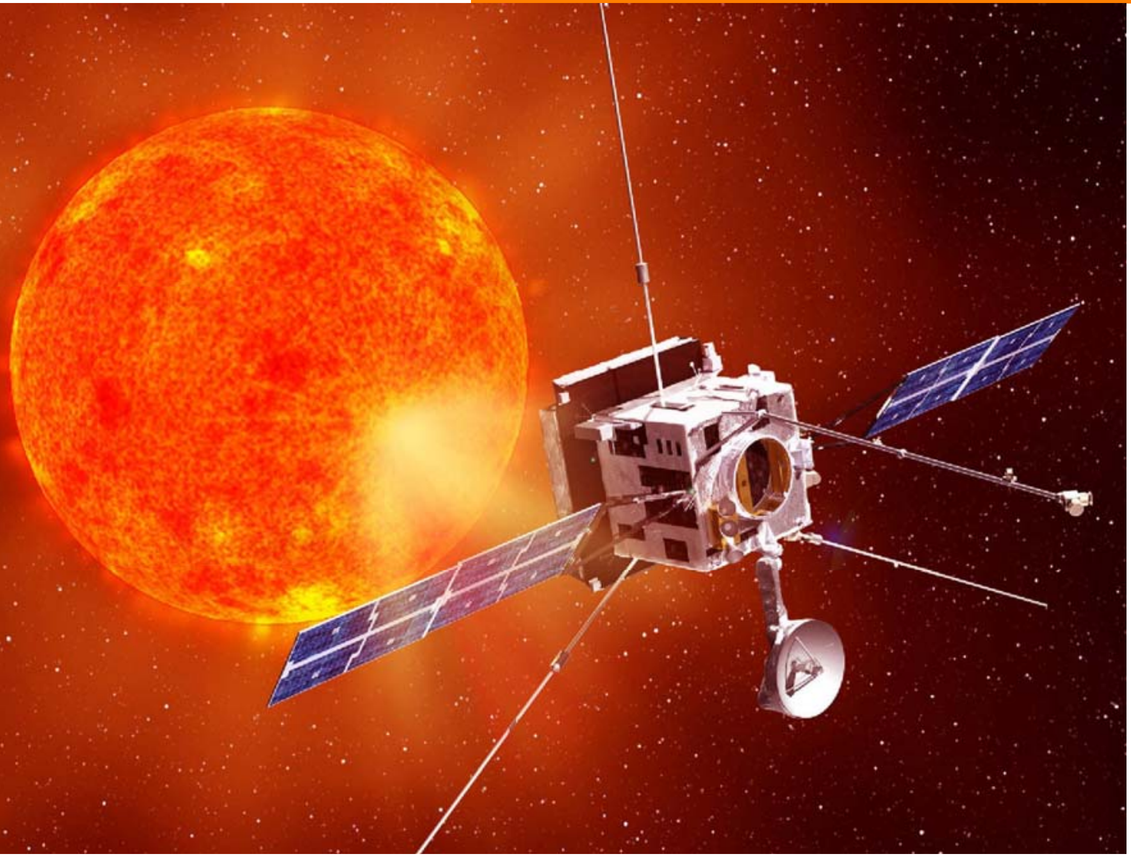
End-to-End Mission Performance Simulators for Space Science missions, a Reference Architecture

SESP 2017



SS-E2ES Team

José Barbosa[1], Jose Julio Ramos[2], Steven Guest[3], Chris Pearson[3], Jose Lopez Moreno[4]

[1]DEIMOS Engenharia
[2]DEIMOS UK
[3]RAL Space
[4]Instituto de Astrofisica de Andalucia

Objectives and Approach to Activity

Objectives

- Define generic user requirements for a SS E2ES to support scientific assessment of remote sensing missions in phase 0/A
- Define a Reference Architecture for such an SS E2E mission simulator by:
  - Categorizing past, current and planned ESA SS missions, in terms of mission application, observation requirements and techniques, instruments and products. Analysis of commonalities to derive most used modules and models.
  - Identifying the building blocks (BB: algorithms, models, functions), required to model the mission elements, in particular the BB that can be generalized for the various mission categories, and defining a template for their description and interfaces
- Apply the reference architecture to the design of two demo missions (one for astrophysics and one for planetary applications, e.g. Euclid and ExoMars)
- Evaluate the reference architecture concept and based on this evaluation, make recommendations for future activities

Similar previous exercise: ARCHEO

Important Directions

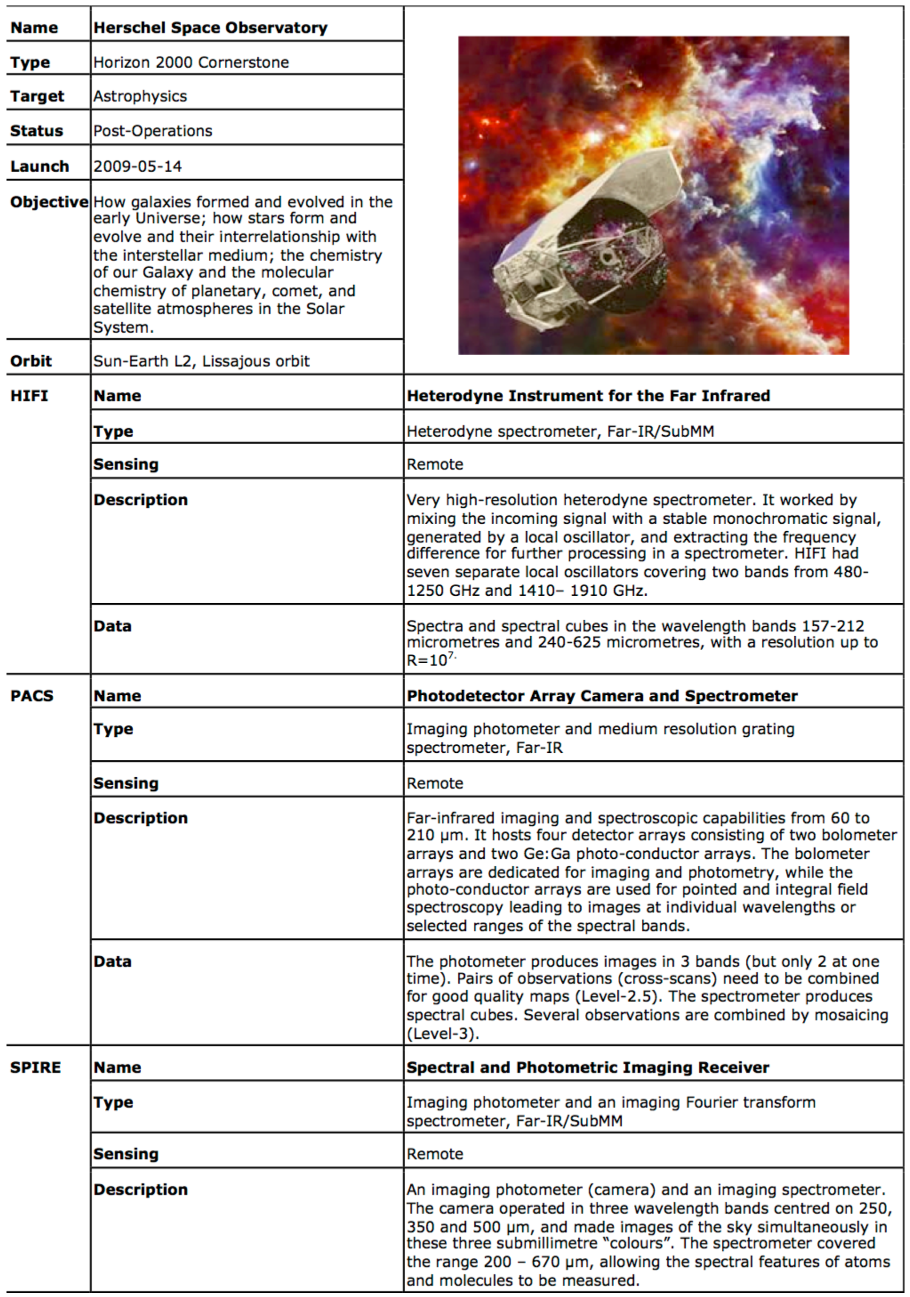
- The set of documents and directives produced was made to be read across several projects for several Space Science Missions. This means that:
  - We tried to take into account our audience as much as possible
  - Respect what is already a convention (e.g., L0.5, FITS and PDS)
  - Documents are not just a collection of tables and bullet lists (it was difficult to avoid)
- We can provide the RA model as an editable basis
- Requirements are a set of guidelines and/or a checklist, not a strict definition of what the E2E simulator should be

- **Important note:** this report is about the activities of one of the consortiums, led by DME. There was a parallel study, led by GMV and reported in the SESP 2015. An activity to merge the outcomes of both activities will start soon.

Categorization and Commonalities TNs

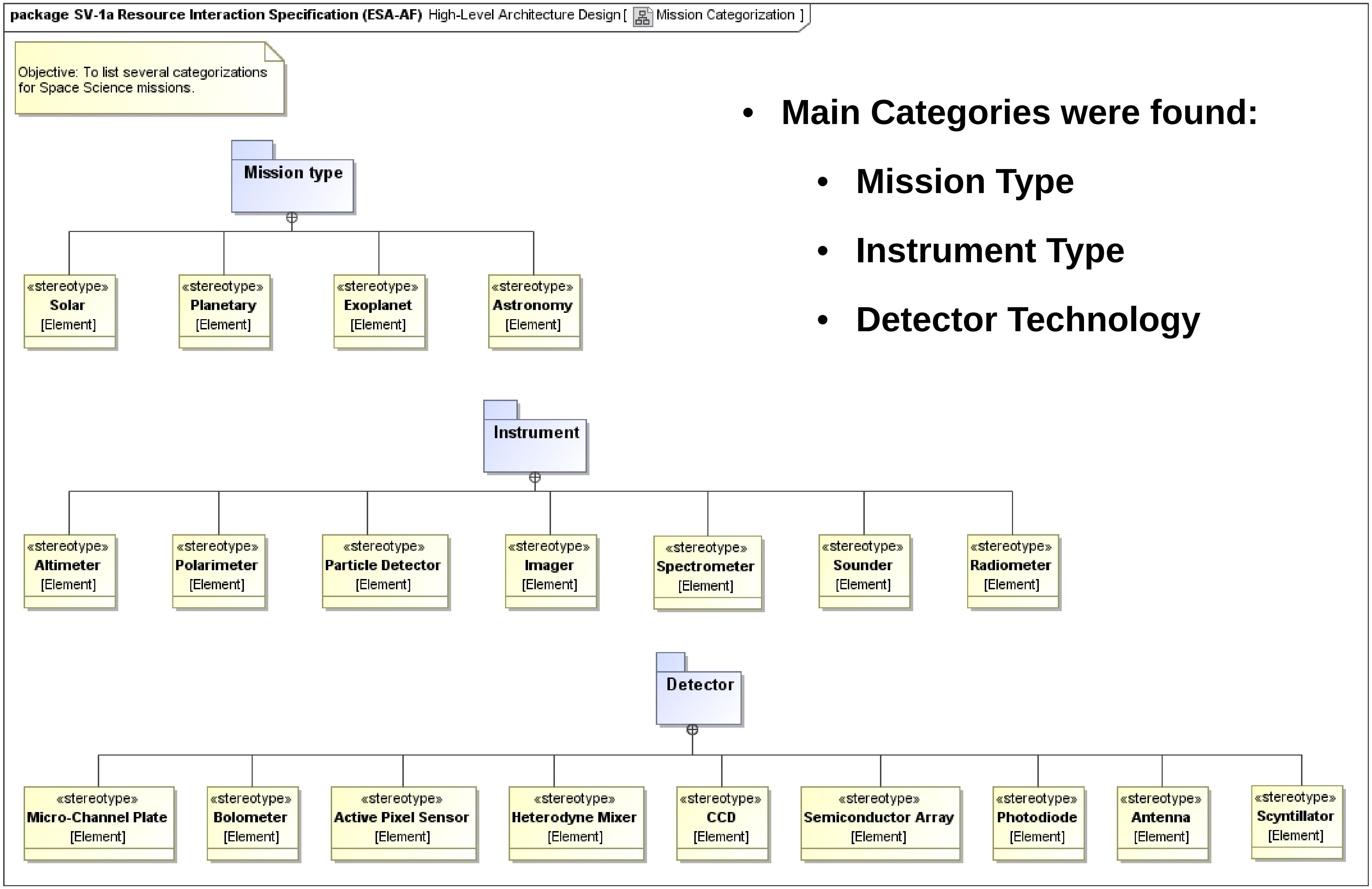
Mission Categorization TN

- 26 missions were analysed in detail and added to a SS Missions Database
- Main mission categories were defined:
  - Mission Type
  - Attitude
  - Orbit
  - Instrument
  - Instrument Type
  - Instrument Class
  - Waveband Class
  - Detector Class
  - Detector Type
  - Detector Composition
  - Product Format
- Several queries possible



Name	**Herschel Space Observatory**	
Type	Horizon 2000 Cornerstone	
Target	Astrophysics	
Status	Post-Operations	
Launch	2009-05-14	
Objective	How galaxies formed and evolved in the early Universe; how stars form and evolve and their interrelationship with the interstellar medium; the chemistry of our Galaxy and the molecular chemistry of planetary, comet, and satellite atmospheres in the Solar System.	
Orbit	Sun-Earth L2, Lissajous orbit	
HIFI	**Name**	**Heterodyne Instrument for the Far Infrared**
	Type	Heterodyne spectrometer, Far-IR/SubMM
	Sensing	Remote
	Description	Very high-resolution heterodyne spectrometer. It worked by mixing the incoming signal with a stable monochromatic signal, generated by a local oscillator, and extracting the frequency difference for further processing in a spectrometer. HIFI had seven separate local oscillators covering two bands from 480-1250 GHz and 1410– 1910 GHz.
	Data	Spectra and spectral cubes in the wavelength bands 157-212 micrometres and 240-625 micrometres, with a resolution up to R=10^{7}.
PACS	**Name**	**Photodetector Array Camera and Spectrometer**
	Type	Imaging photometer and medium resolution grating spectrometer, Far-IR
	Sensing	Remote
	Description	Far-infrared imaging and spectroscopic capabilities from 60 to 210 μm. It hosts four detector arrays consisting of two bolometer arrays and two Ge:Ga photo-conductor arrays. The bolometer arrays are dedicated for imaging and photometry, while the photo-conductor arrays are used for pointed and integral field spectroscopy leading to images at individual wavelengths or selected ranges of the spectral bands.
	Data	The photometer produces images in 3 bands (but only 2 at one time). Pairs of observations (cross-scans) need to be combined for good quality maps (Level-2.5). The spectrometer produces spectral cubes. Several observations are combined by mosaicing (Level-3).
SPIRE	**Name**	**Spectral and Photometric Imaging Receiver**
	Type	Imaging photometer and an imaging Fourier transform spectrometer, Far-IR/SubMM
	Sensing	Remote
	Description	An imaging photometer (camera) and an imaging spectrometer. The camera operated in three wavelength bands centred on 250, 350 and 500 μm, and made images of the sky simultaneously in these three submillimetre "colours". The spectrometer covered the range 200 – 670 μm, allowing the spectral features of atoms and molecules to be measured.

Mission Categorization TN



package SV-1a Resource Interaction Specification (ESA-AF) High-Level Architecture Design [Mission Categorization]
Objective: To list several categorizations for Space Science missions.
Mission type
«stereotype» Solar [Element]
«stereotype» Planetary [Element]
«stereotype» Exoplanet [Element]
«stereotype» Astronomy [Element]
Instrument
«stereotype» Altimeter [Element]
«stereotype» Polarimeter [Element]
«stereotype» Particle Detector [Element]
«stereotype» Imager [Element]
«stereotype» Spectrometer [Element]
«stereotype» Sounder [Element]
«stereotype» Radiometer [Element]
Detector
«stereotype» Micro-Channel Plate [Element]
«stereotype» Bolometer [Element]
«stereotype» Active Pixel Sensor [Element]
«stereotype» Heterodyne Mixer [Element]
«stereotype» CCD [Element]
«stereotype» Semiconductor Array [Element]
«stereotype» Photodiode [Element]
«stereotype» Antenna [Element]
«stereotype» Scyntillator [Element]

- **Main Categories were found:**
  - **Mission Type**
  - **Instrument Type**
  - **Detector Technology**

Mission Commonalities TN

- Commonalities analysis divided by Processing Level. The underlying idea is that commonalities allow us to define more generic Building Blocks inside each Processing Level.
- Processing Levels are the same as the ones recommended for the EO Reference Architecture study. This is intended as it simplifies the nomenclature and understanding of the recommendations:
  - **Geometric Module** (Orbit and Attitude computation commonalities studied)
  - **Scene Generator Module** (7 commonality classes found)
  - **Forward Model Module** (new module, responsible for physical scene simulation)
  - **Instrument Module** (most commonalities found in detector simulation)
  - **Data Processing Modules** (Levels 0, 0.5, 1 and 2 studied)
  - **Performance Evaluation Module**
  - **Utilities**

Recommended Data Product Levels

- Needed to define “standard” levels as there is no standard in SS.
- Generic Data Levels proposed, based on a “typical” astronomy mission:

SS-E2ES Proposed Level	NASA level	Summary	Description
Telemetry	Packet Data	Raw telemetry	Telemetry packets. May be packaged into a file format e.g. ESOC's DDS.
Level 0	Level-0	Raw data	Raw data reformatted from telemetry into a more generally useful format.
Level 0.5	Level 1-A	Raw data in physical units	Data converted into physical units e.g. volts. Often this process will be reversible, but may not be, at least not without loss of accuracy.
Level 1	Level 1-B	Calibrated data	Data calibrated to remove instrumental effects. (This is the goal; in practice some effects may not be removable before the next level).
Level 2	Level 1-C	Science data	Fully calibrated and scientifically useful data products, e.g. images, spectra, spectral cubes.
Level 3	Level 2	Derived data	Further science data products extracted from, or by combining, the Level-2 data. Examples are mosaicked maps, line lists, point source lists.

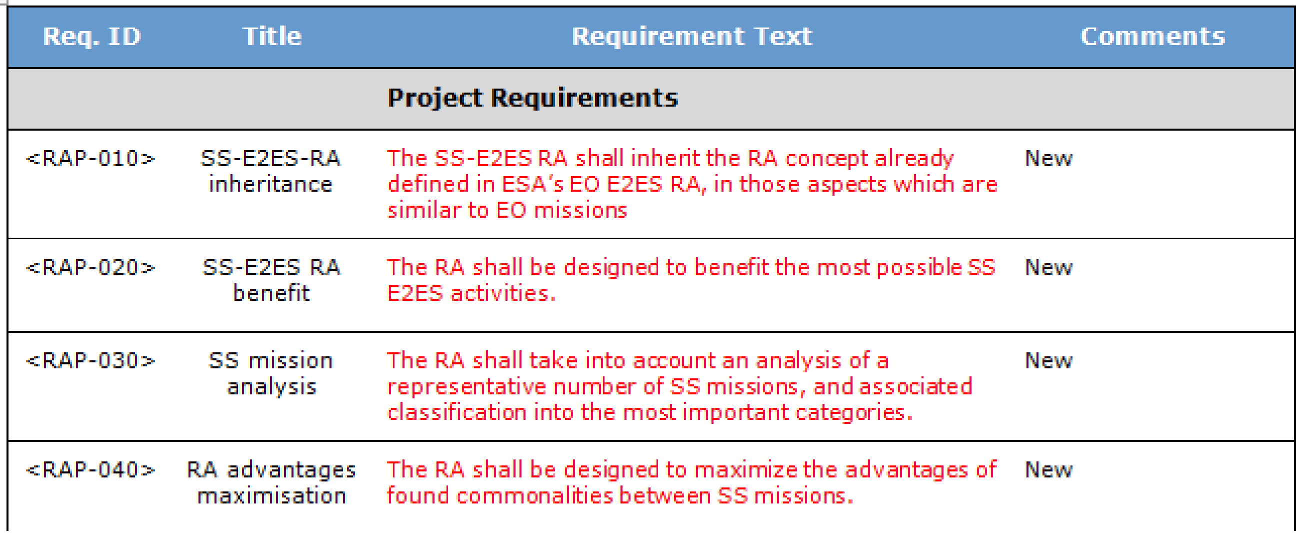
Space Science E2E Mission Requirements Definition

Final Requirement Definition

- Guidelines the team followed for requirement definition:
  - Generalisation of requirements in SoW
  - Analysis of mission and instrument commonalities
  - Experience in E2ES development
  - Practical philosophy: common practices, usefulness and future re-usability

- Example table:
  - Reqs. have code and names

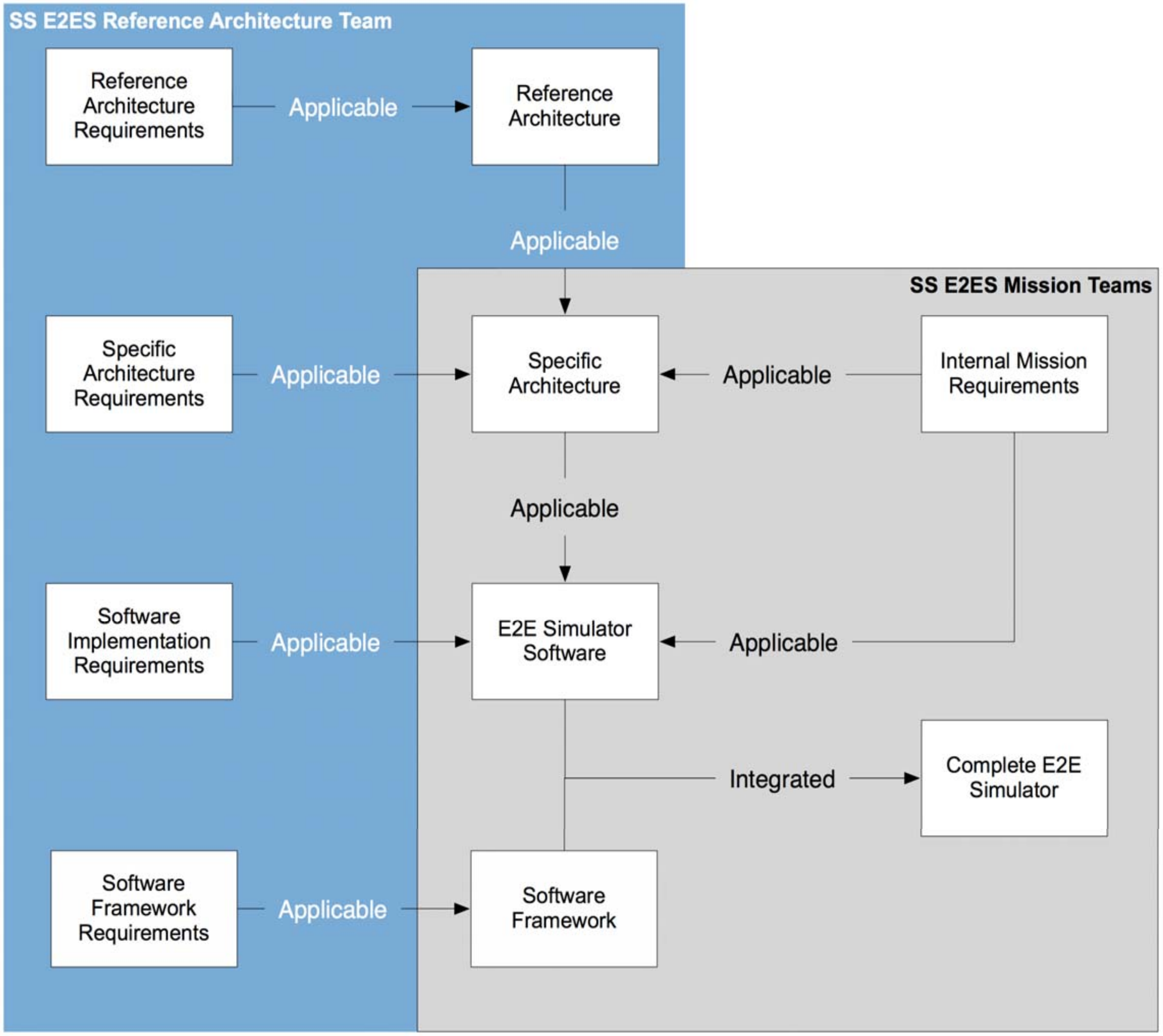
Table 2-3: Reference Architecture Requirements



Req. ID	Title	Requirement Text	Comments
		Project Requirements	
<RAP-010>	SS-E2ES-RA inheritance	The SS-E2ES RA shall inherit the RA concept already defined in ESA's EO E2ES RA, in those aspects which are similar to EO missions	New
<RAP-020>	SS-E2ES RA benefit	The RA shall be designed to benefit the most possible SS E2ES activities.	New
<RAP-030>	SS mission analysis	The RA shall take into account an analysis of a representative number of SS missions, and associated classification into the most important categories.	New
<RAP-040>	RA advantages maximisation	The RA shall be designed to maximize the advantages of found commonalities between SS missions.	New

Requirements Applicability

Requirement classification within this activity was done taking into account how it is applicable to SS E2ES development:



Final Requirement List

- **Reference Architecture Requirements** classification:
  - Project requirements: related to the overall objectives and needs of this study: 7 reqs
  - Functional requirements: related to the functionalities that the simulator obtained applying the RA should have:18 reqs
  - Design requirements: related to the RA layout:13 reqs
  - Interface requirements: most generic interfaces: 2 reqs
- **Specific Architecture Requirements**
  - Functional: 6 reqs, Design: 6 reqs, Interface: 6 reqs
- **Software Framework Requirements**
  - General: 9 reqs, Interface: 3 reqs, Control: 15 reqs, Data storage and visualization 4 reqs
- **Implementation requirements**
  - Functional: 4 reqs, Design: 11 reqs, Interface: 10 reqs, Performance: 7 reqs

Reference Architecture Modelling Approach

Reference Architecture work

- Objective: Define a Reference Architecture for the development of simulators for SS missions.
- Including:
  - Following a stablished methodology, minimizing risks
  - Defining a shared language to communicate and collaborate
  - Covering all –other- aspects of an architecture interesting to stakeholders
  - Providing a set of supporting material (docs and digital companions)
- Approach
  - Choose and follow a Enterprise Architecture Framework for defining the RA
  - Collection of user requirements and practices on the Space-domain
  - Extend, expand and complete RA viewpoints
  - Provide an unique RA model repository and usage tutorial
- Challenges of new work
  - Avoid overhead, misdirection and over-specialization.
  - THINK ON FUTURE USERS!

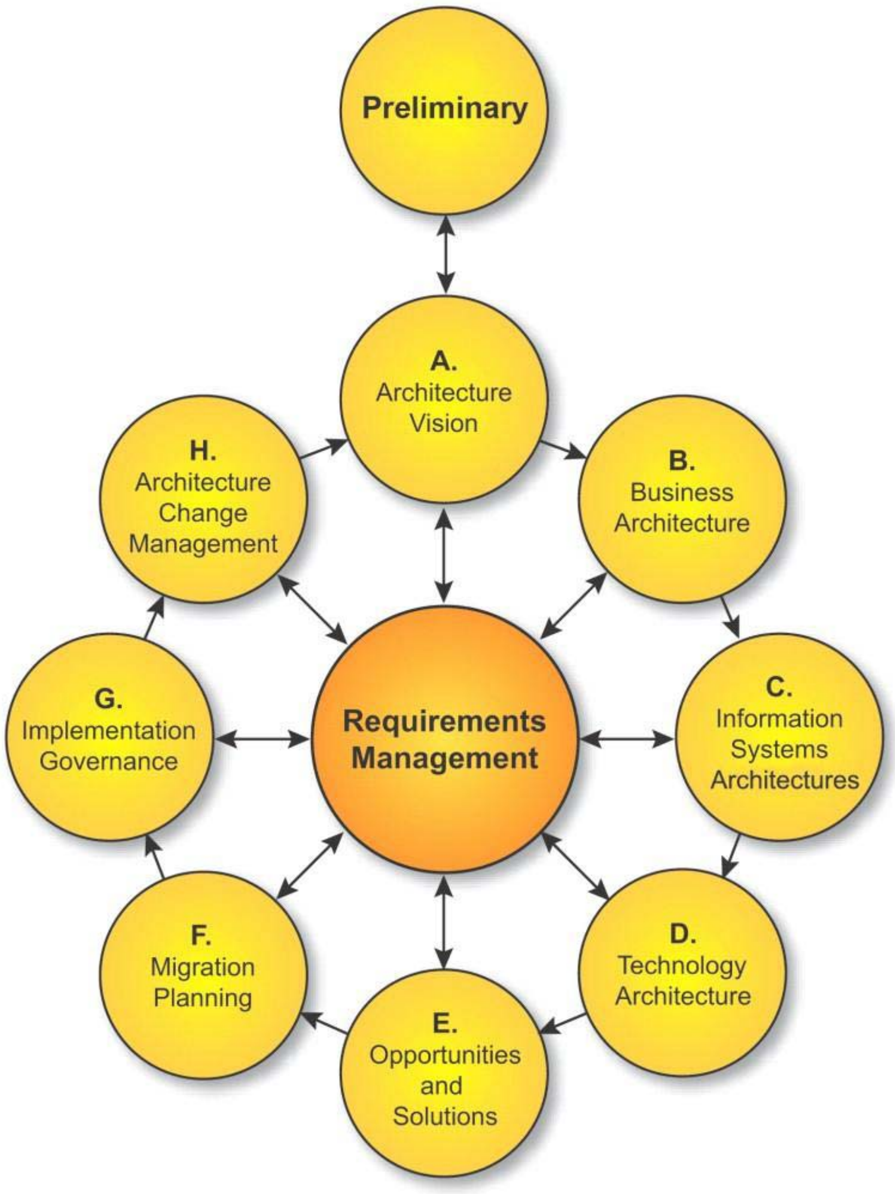
System Engineering approach

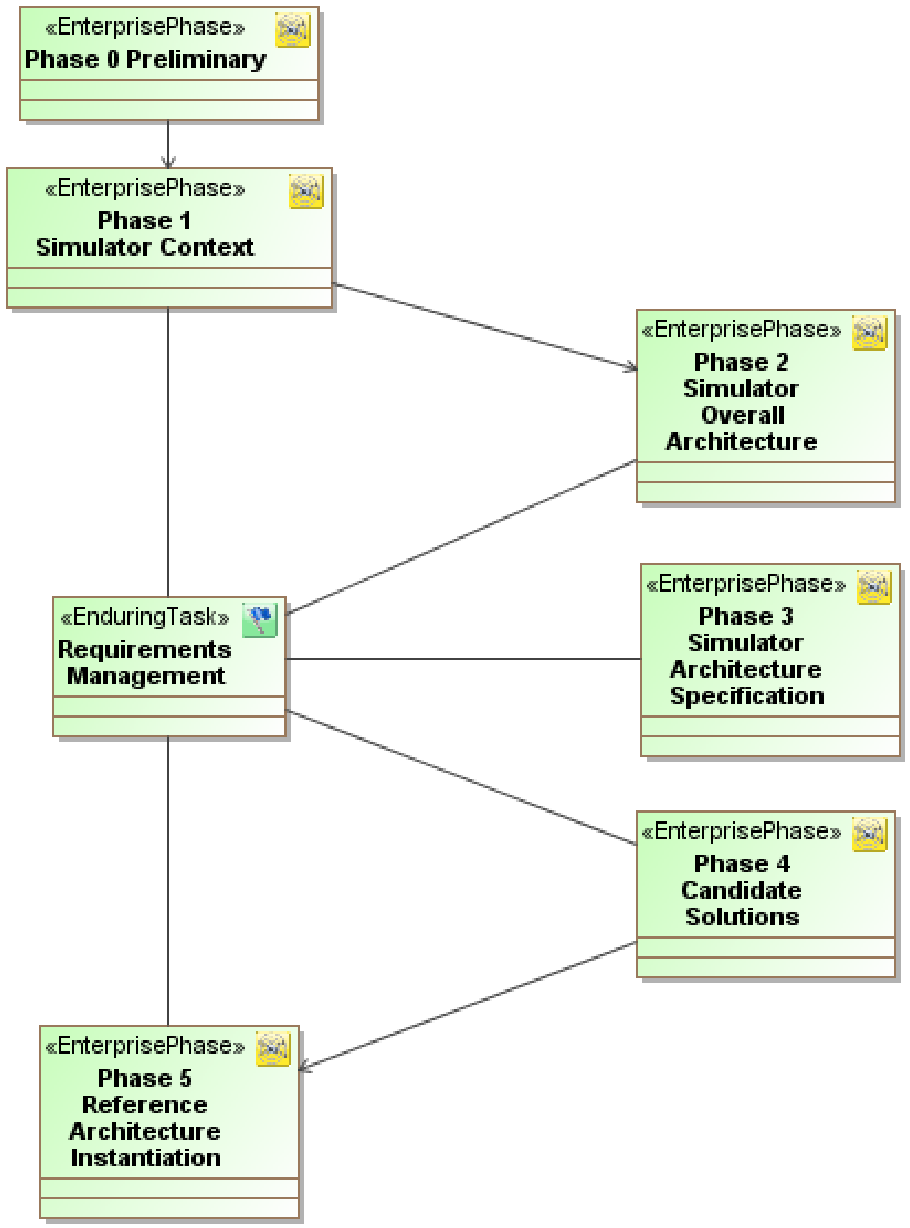
NOT A SOFTWARE FRAMEWORK!!

Phase 0 - Enterprise Architecture Framework selection

- Evaluation
  - Made a survey of existing frameworks: TOGAF, DoDAF, MODAF, Zachman Framework and ESAAF.
- Decision
  - Preferred solution: ESAAF
  - Backup solution: TOGAF
- ESA provided the MagicDraw ESAAF plugin with additional resources.

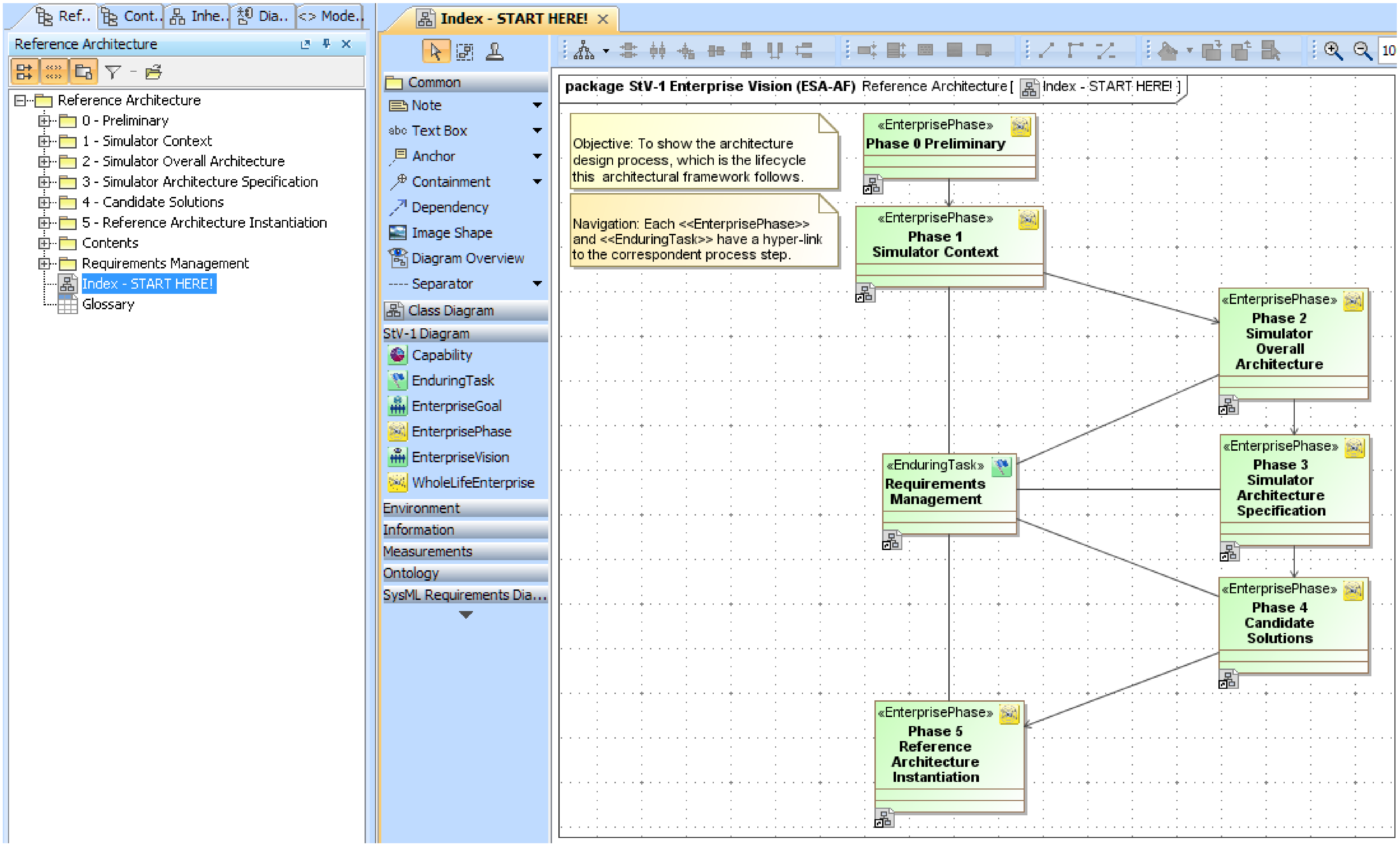
Phase 0 - Adaptation to SS-E2ES


Preliminary
A. Architecture Vision
B. Business Architecture
C. Information Systems Architectures
D. Technology Architecture
E. Opportunities and Solutions
F. Migration Planning
G. Implementation Governance
H. Architecture Change Management
Requirements Management


«EnterprisePhase» Phase 0 Preliminary
«EnterprisePhase» Phase 1 Simulator Context
«EnterprisePhase» Phase 2 Simulator Overall Architecture
«EnduringTask» Requirements Management
«EnterprisePhase» Phase 3 Simulator Architecture Specification
«EnterprisePhase» Phase 4 Candidate Solutions
«EnterprisePhase» Phase 5 Reference Architecture Instantiation

Reference Architecture

START HERE!



Ref.. Cont.. Inhe.. Dia.. Mode..

Reference Architecture

- Reference Architecture
  - 0 - Preliminary
  - 1 - Simulator Context
  - 2 - Simulator Overall Architecture
  - 3 - Simulator Architecture Specification
  - 4 - Candidate Solutions
  - 5 - Reference Architecture Instantiation
  - Contents
  - Requirements Management
  - Index - START HERE!
  - Glossary

Index - START HERE!

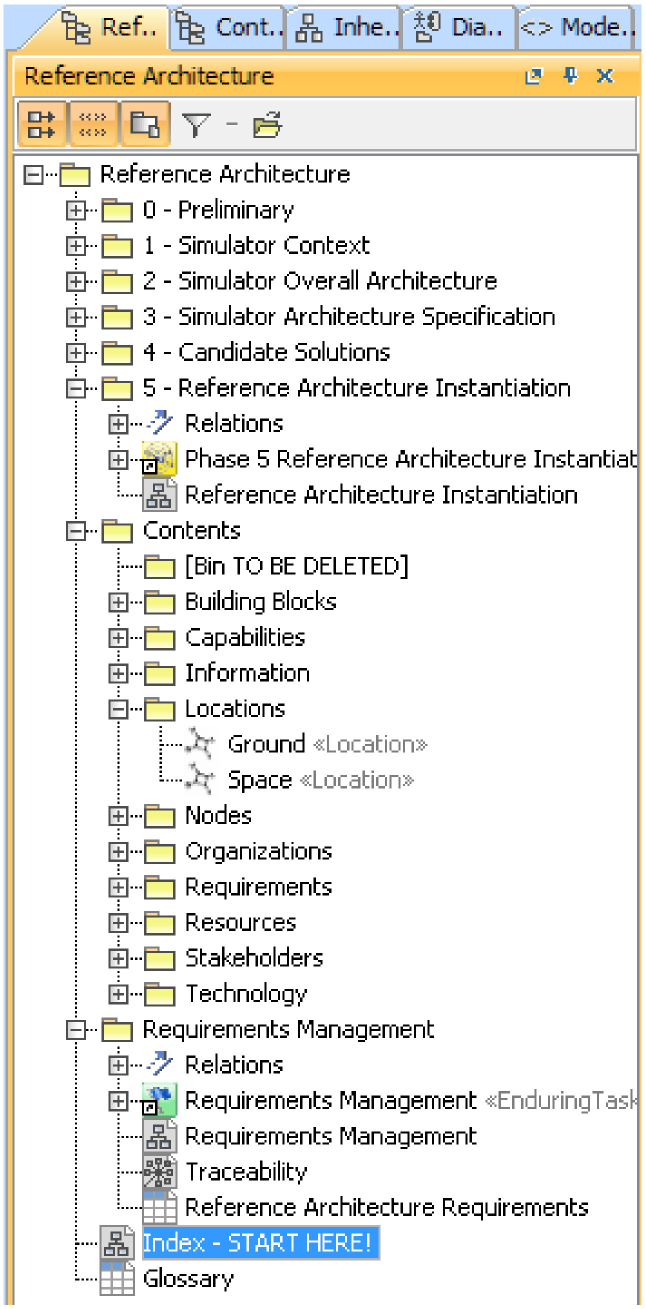
package StV-1 Enterprise Vision (ESA-AF) Reference Architecture [Index - START HERE!]

Objective: To show the architecture design process, which is the lifecycle this architectural framework follows.

Navigation: Each <<EnterprisePhase>> and <<EnduringTask>> have a hyper-link to the correspondent process step.

«EnterprisePhase» Phase 0 Preliminary
«EnterprisePhase» Phase 1 Simulator Context
«EnterprisePhase» Phase 2 Simulator Overall Architecture
«EnterprisePhase» Phase 3 Simulator Architecture Specification
«EnduringTask» Requirements Management
«EnterprisePhase» Phase 4 Candidate Solutions
«EnterprisePhase» Phase 5 Reference Architecture Instantiation

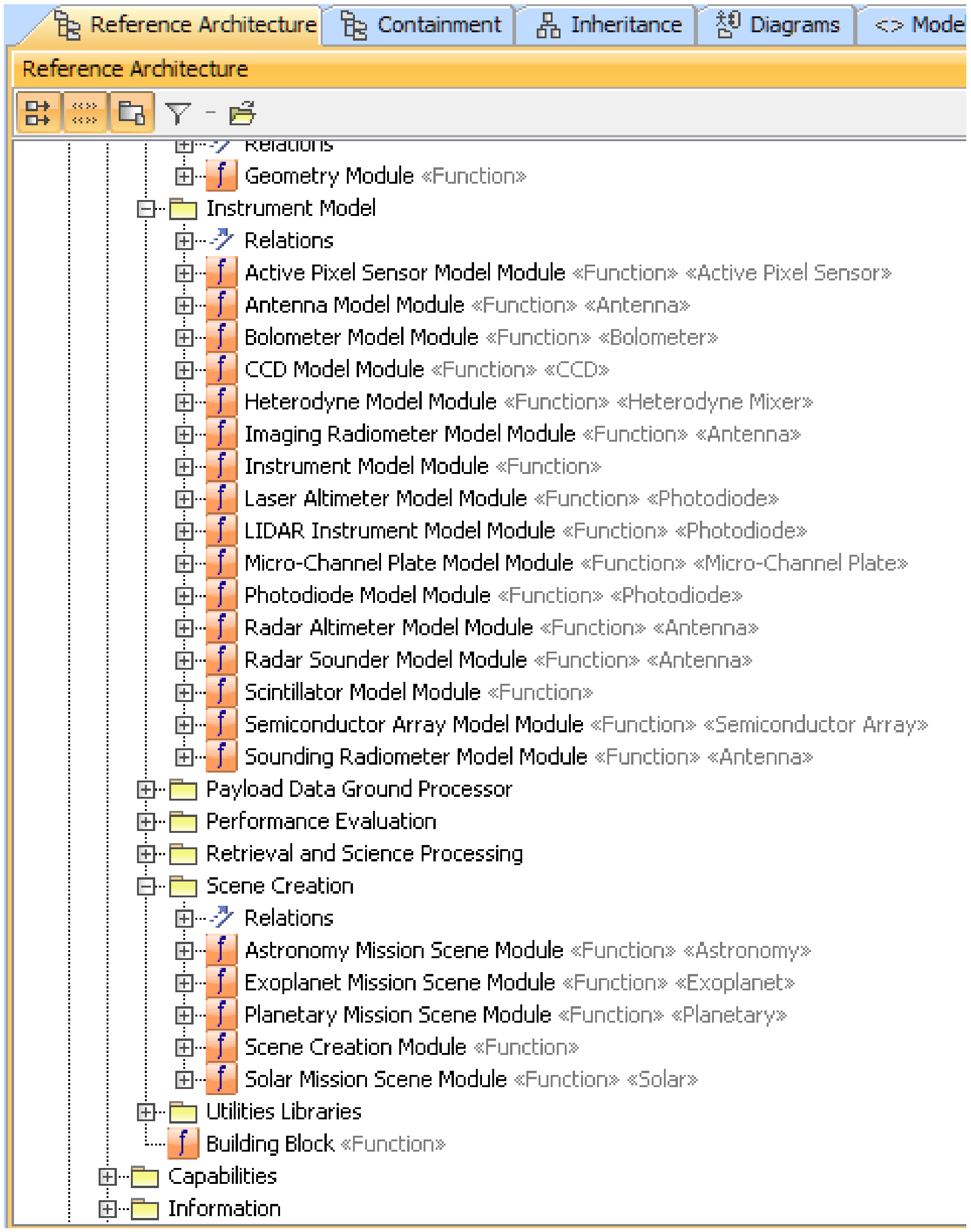
Model structure and navigation



Ref.. | Cont.. | Inhe.. | Dia.. | <> Mode..

Reference Architecture

- Reference Architecture
  - 0 - Preliminary
  - 1 - Simulator Context
  - 2 - Simulator Overall Architecture
  - 3 - Simulator Architecture Specification
  - 4 - Candidate Solutions
  - 5 - Reference Architecture Instantiation
    - Relations
    - Phase 5 Reference Architecture Instantiat
    - Reference Architecture Instantiation
  - Contents
    - [Bin TO BE DELETED]
    - Building Blocks
    - Capabilities
    - Information
    - Locations
      - Ground «Location»
      - Space «Location»
    - Nodes
    - Organizations
    - Requirements
    - Resources
    - Stakeholders
    - Technology
  - Requirements Management
    - Relations
    - Requirements Management «EnduringTask
    - Requirements Management
    - Traceability
    - Reference Architecture Requirements
  - Index - START HERE!
  - Glossary



Reference Architecture | Containment | Inheritance | Diagrams | <> Mode

Reference Architecture

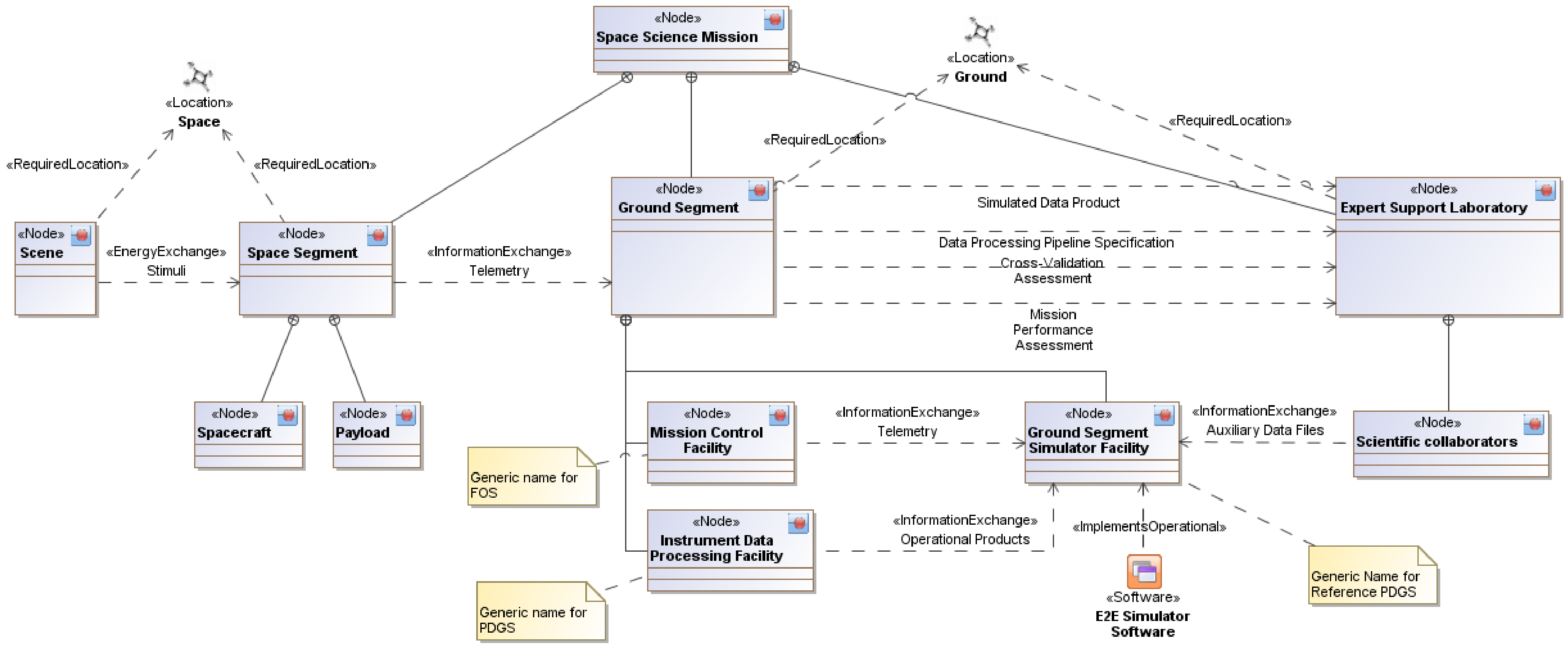
- Relations
- Geometry Module «Function»
- Instrument Model
  - Relations
  - Active Pixel Sensor Model Module «Function» «Active Pixel Sensor»
  - Antenna Model Module «Function» «Antenna»
  - Bolometer Model Module «Function» «Bolometer»
  - CCD Model Module «Function» «CCD»
  - Heterodyne Model Module «Function» «Heterodyne Mixer»
  - Imaging Radiometer Model Module «Function» «Antenna»
  - Instrument Model Module «Function»
  - Laser Altimeter Model Module «Function» «Photodiode»
  - LIDAR Instrument Model Module «Function» «Photodiode»
  - Micro-Channel Plate Model Module «Function» «Micro-Channel Plate»
  - Photodiode Model Module «Function» «Photodiode»
  - Radar Altimeter Model Module «Function» «Antenna»
  - Radar Sounder Model Module «Function» «Antenna»
  - Scintillator Model Module «Function»
  - Semiconductor Array Model Module «Function» «Semiconductor Array»
  - Sounding Radiometer Model Module «Function» «Antenna»
- Payload Data Ground Processor
- Performance Evaluation
- Retrieval and Science Processing
- Scene Creation
  - Relations
  - Astronomy Mission Scene Module «Function» «Astronomy»
  - Exoplanet Mission Scene Module «Function» «Exoplanet»
  - Planetary Mission Scene Module «Function» «Planetary»
  - Scene Creation Module «Function»
  - Solar Mission Scene Module «Function» «Solar»
- Utilities Libraries
- Building Block «Function»

Capabilities

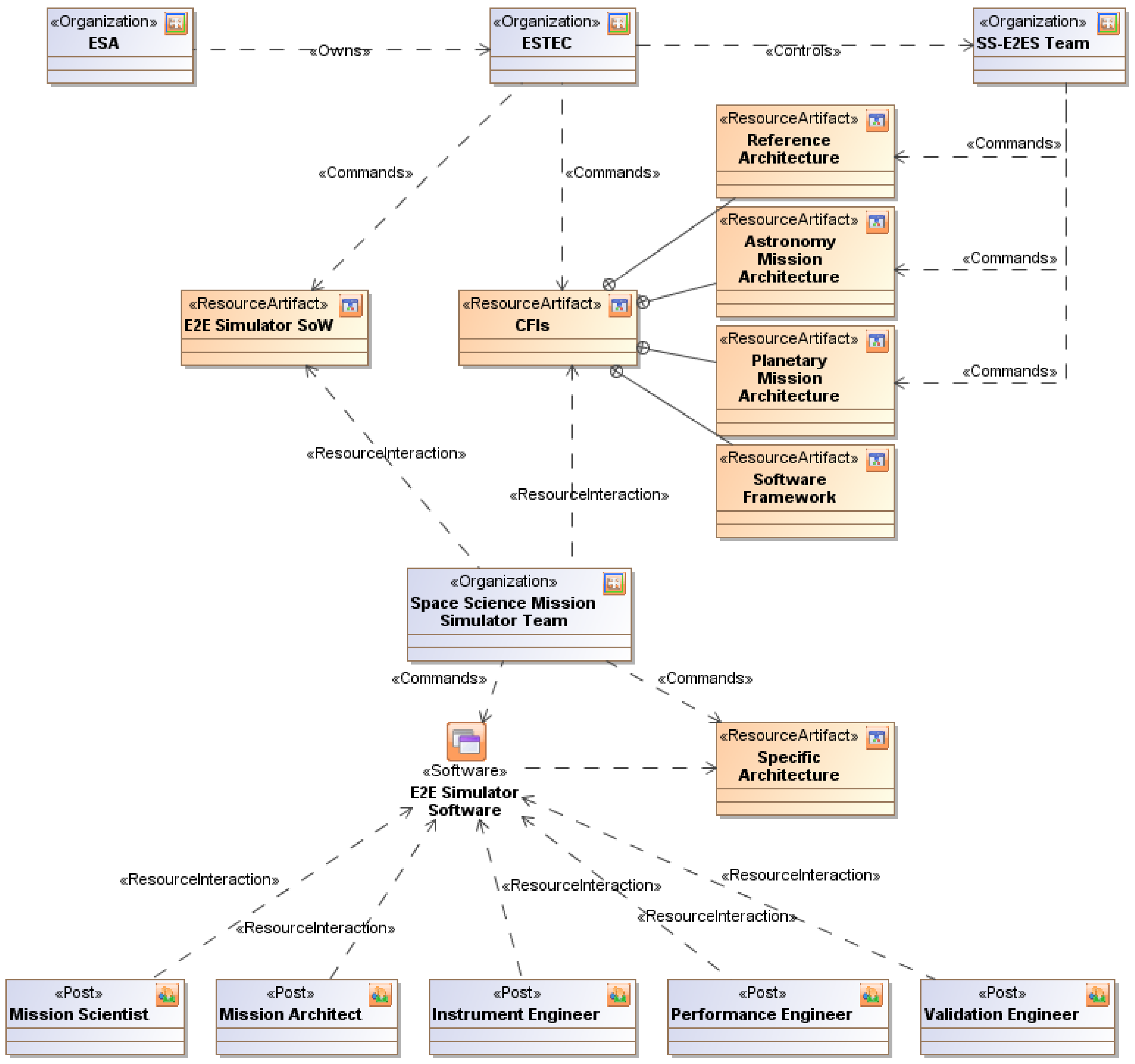
Information

Phase 1 – Simulator Context

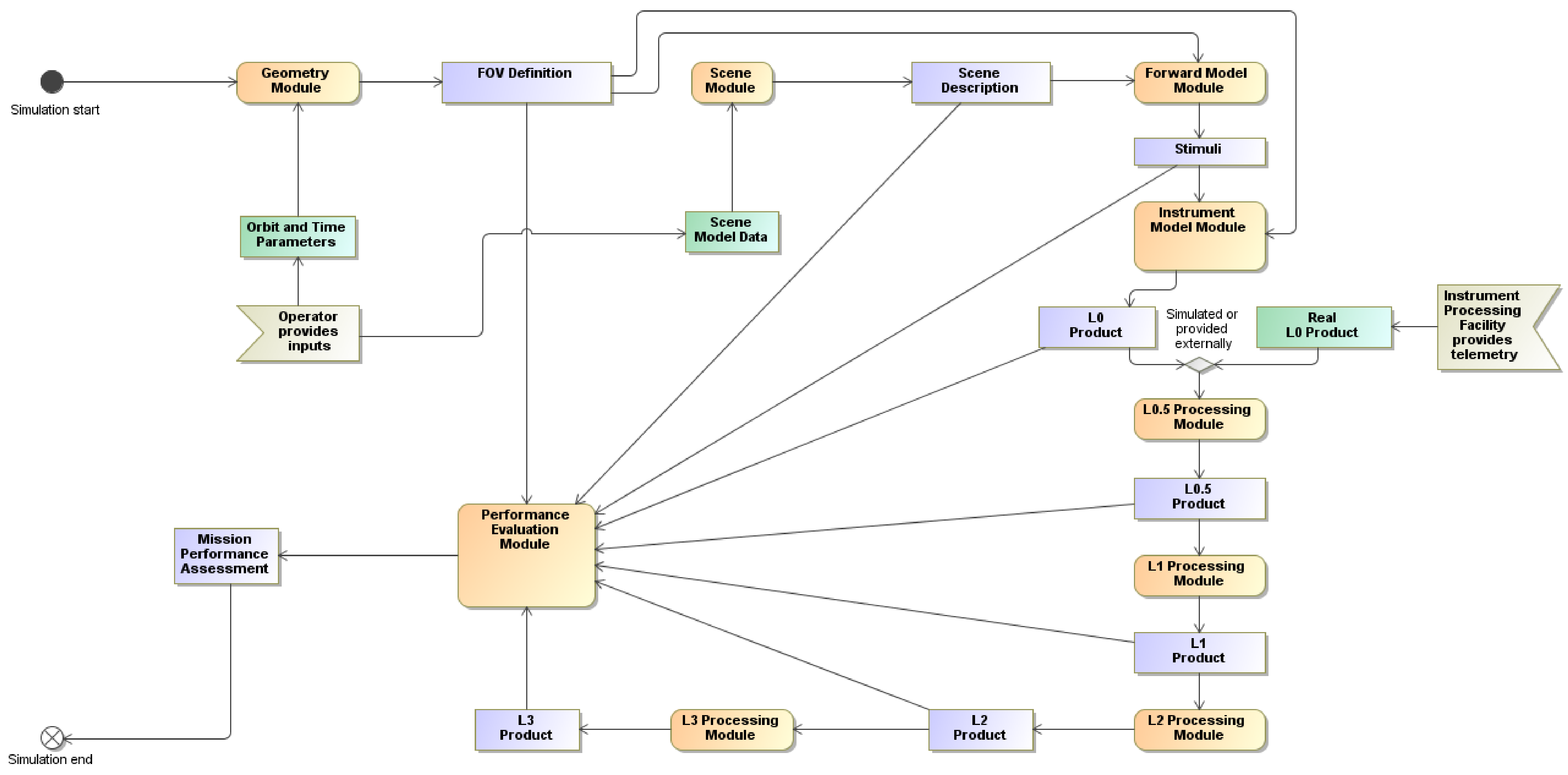
This diagram is an opportunity to be creative in the presentation


«Node» Space Science Mission
«Location» Space
«Location» Ground
«RequiredLocation»
«RequiredLocation»
«RequiredLocation»
«RequiredLocation»
«Node» Scene
«EnergyExchange» Stimuli
«Node» Space Segment
«InformationExchange» Telemetry
«Node» Ground Segment
«Node» Expert Support Laboratory
Simulated Data Product
Data Processing Pipeline Specification
Cross-Validation Assessment
Mission Performance Assessment
«Node» Spacecraft
«Node» Payload
«Node» Mission Control Facility
«InformationExchange» Telemetry
«Node» Ground Segment Simulator Facility
«InformationExchange» Auxiliary Data Files
«Node» Scientific collaborators
Generic name for FOS
«Node» Instrument Data Processing Facility
«InformationExchange» Operational Products
«ImplementsOperational»
«Software» E2E Simulator Software
Generic Name for Reference PDGS
Generic name for PDGS

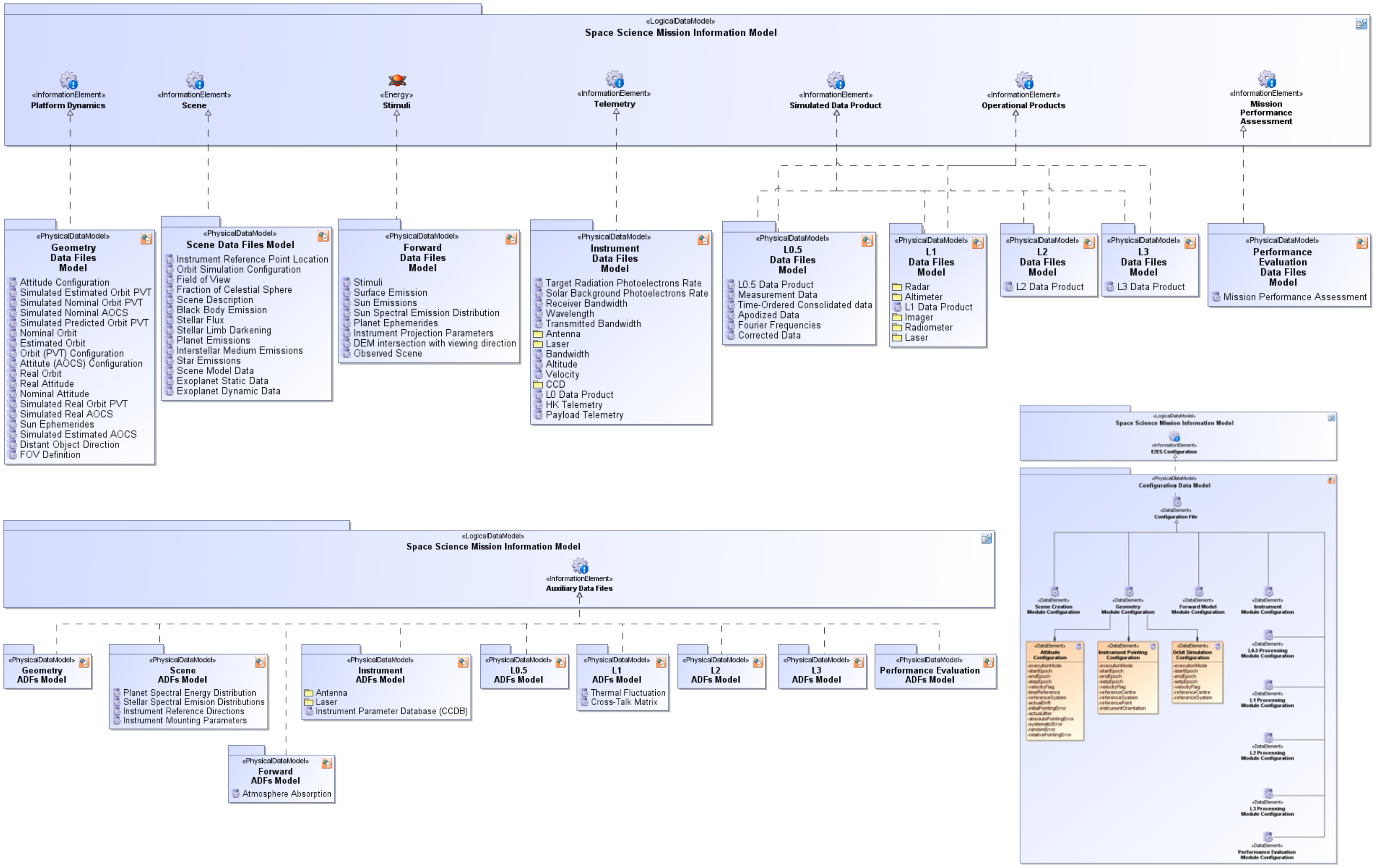
Phase 1 – Simulator Context


«Organization» ESA
«Owns»
«Organization» ESTEC
«Controls»
«Organization» SS-E2ES Team
«ResourceArtifact» Reference Architecture
«Commands»
«Commands»
«Commands»
«ResourceArtifact» Astronomy Mission Architecture
«Commands»
«ResourceArtifact» E2E Simulator SoW
«ResourceArtifact» CFIs
«ResourceArtifact» Planetary Mission Architecture
«Commands»
«ResourceInteraction»
«ResourceInteraction»
«ResourceArtifact» Software Framework
«Organization» Space Science Mission Simulator Team
«Commands»
«Commands»
«Software» E2E Simulator Software
«ResourceArtifact» Specific Architecture
«ResourceInteraction»
«ResourceInteraction»
«ResourceInteraction»
«ResourceInteraction»
«ResourceInteraction»
«Post» Mission Scientist
«Post» Mission Architect
«Post» Instrument Engineer
«Post» Performance Engineer
«Post» Validation Engineer

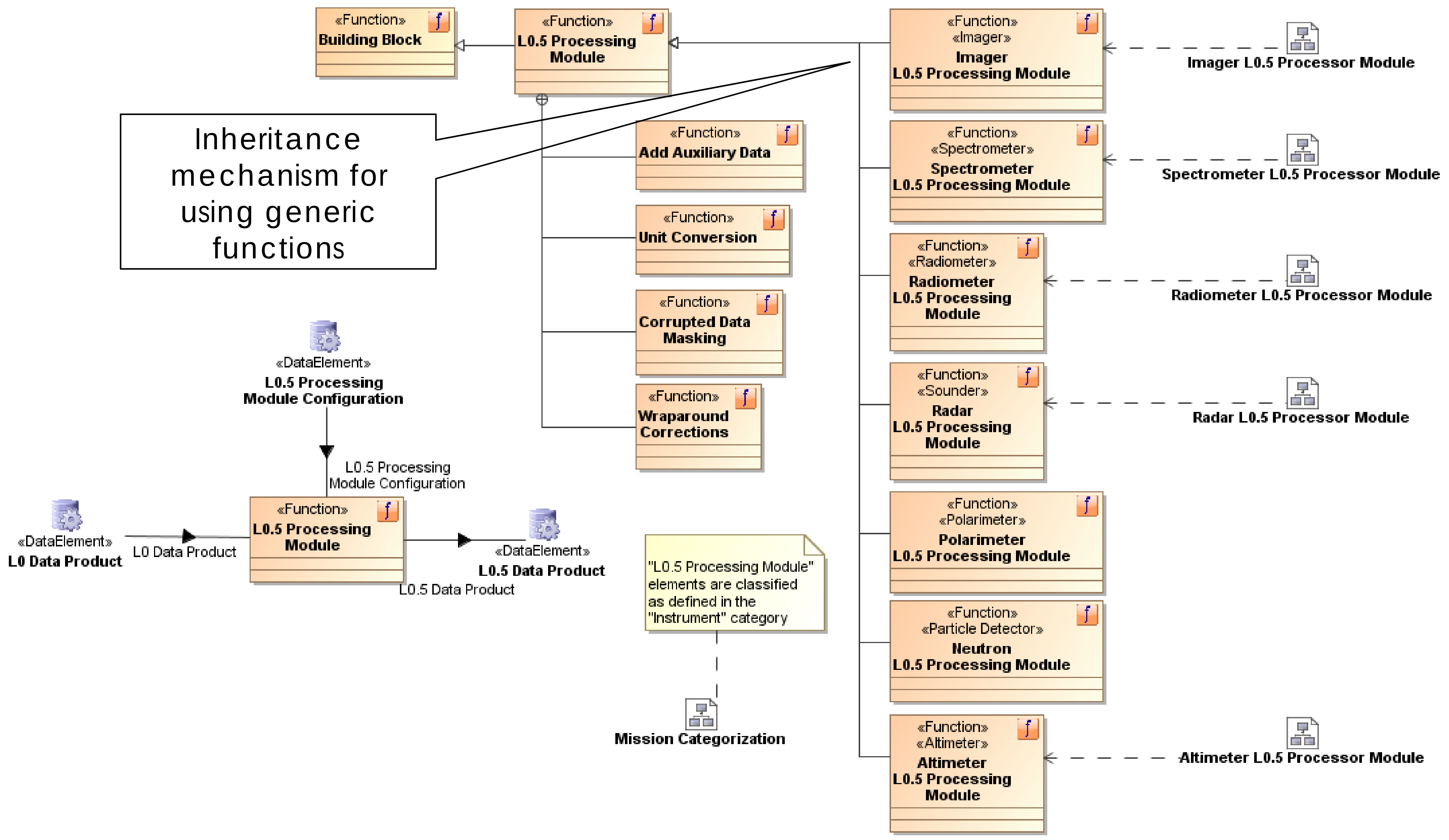
Phase 2 – Simulator Overall Architecture



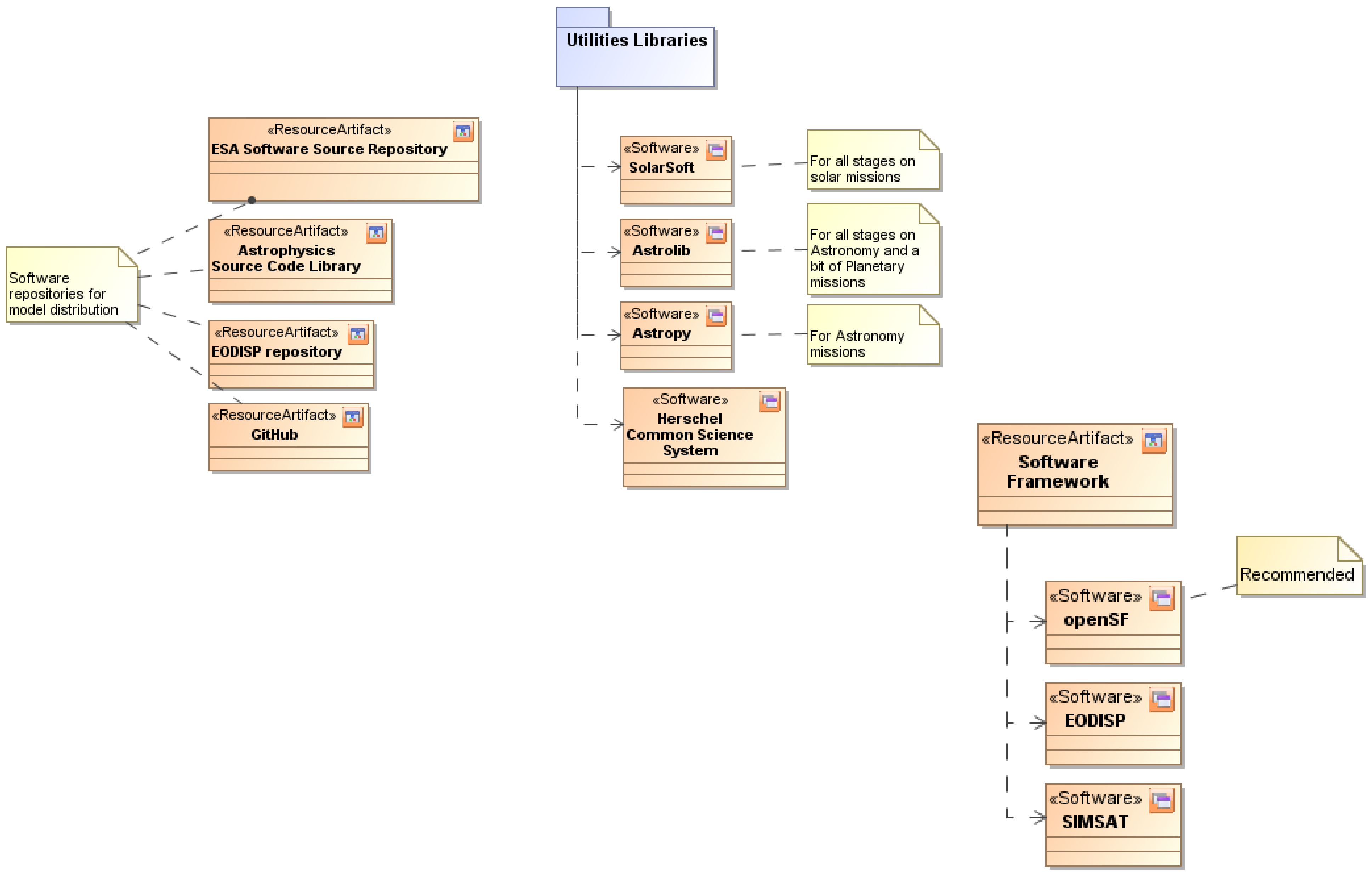
Phase 3 – Simulator Architecture Specification. Data architecture



Phase 3 – Simulator Architecture Specification. Building Blocks design


«Function»
Building Block
«Function»
L0.5 Processing Module
«Function»
Add Auxiliary Data
«Function»
Unit Conversion
«Function»
Corrupted Data Masking
«Function»
Wraparound Corrections
«Function» «Imager»
Imager L0.5 Processing Module
«Function» «Spectrometer»
Spectrometer L0.5 Processing Module
«Function» «Radiometer»
Radiometer L0.5 Processing Module
«Function» «Sounder»
Radar L0.5 Processing Module
«Function» «Polarimeter»
Polarimeter L0.5 Processing Module
«Function» «Particle Detector»
Neutron L0.5 Processing Module
«Function» «Altimeter»
Altimeter L0.5 Processing Module
Imager L0.5 Processor Module
Spectrometer L0.5 Processor Module
Radiometer L0.5 Processor Module
Radar L0.5 Processor Module
Altimeter L0.5 Processor Module
Inheritance mechanism for using generic functions
«DataElement»
L0.5 Processing Module Configuration
L0.5 Processing Module Configuration
«DataElement»
L0 Data Product
L0 Data Product
«Function»
L0.5 Processing Module
«DataElement»
L0.5 Data Product
L0.5 Data Product
"L0.5 Processing Module" elements are classified as defined in the "Instrument" category
Mission Categorization

Phase 4 – Candidate Solutions



Utilities Libraries
«ResourceArtifact» ESA Software Source Repository
«ResourceArtifact» Astrophysics Source Code Library
«ResourceArtifact» EODISP repository
«ResourceArtifact» GitHub
Software repositories for model distribution
«Software» SolarSoft
For all stages on solar missions
«Software» Astrolib
For all stages on Astronomy and a bit of Planetary missions
«Software» Astropy
For Astronomy missions
«Software» Herschel Common Science System
«ResourceArtifact» Software Framework
«Software» openSF
Recommended
«Software» EODISP
«Software» SIMSAT

Phase 5 – Reference Architecture Instantiation

- Defined an implementation strategy for the users:
  - Follow the RA model as “tutorial”
  - Steps:
    - A. Set up your simulator context: SS mission context, stakeholders, system capabilities and constraints, etc.
    - B. Set up your overall architecture: simulation models, data and data flow, target architectures by mission category, etc.
    - C. Set up your architecture specification: describe and categorize elements using provided tools
    - D. Describe your technological solution.

- Users will now have an architectural design ready to be detailed and implemented.
  - Requirements Baseline and Technical Specification
  - Cookbook provided to guide first steps

SS E2E Simulator Building Blocks Definition

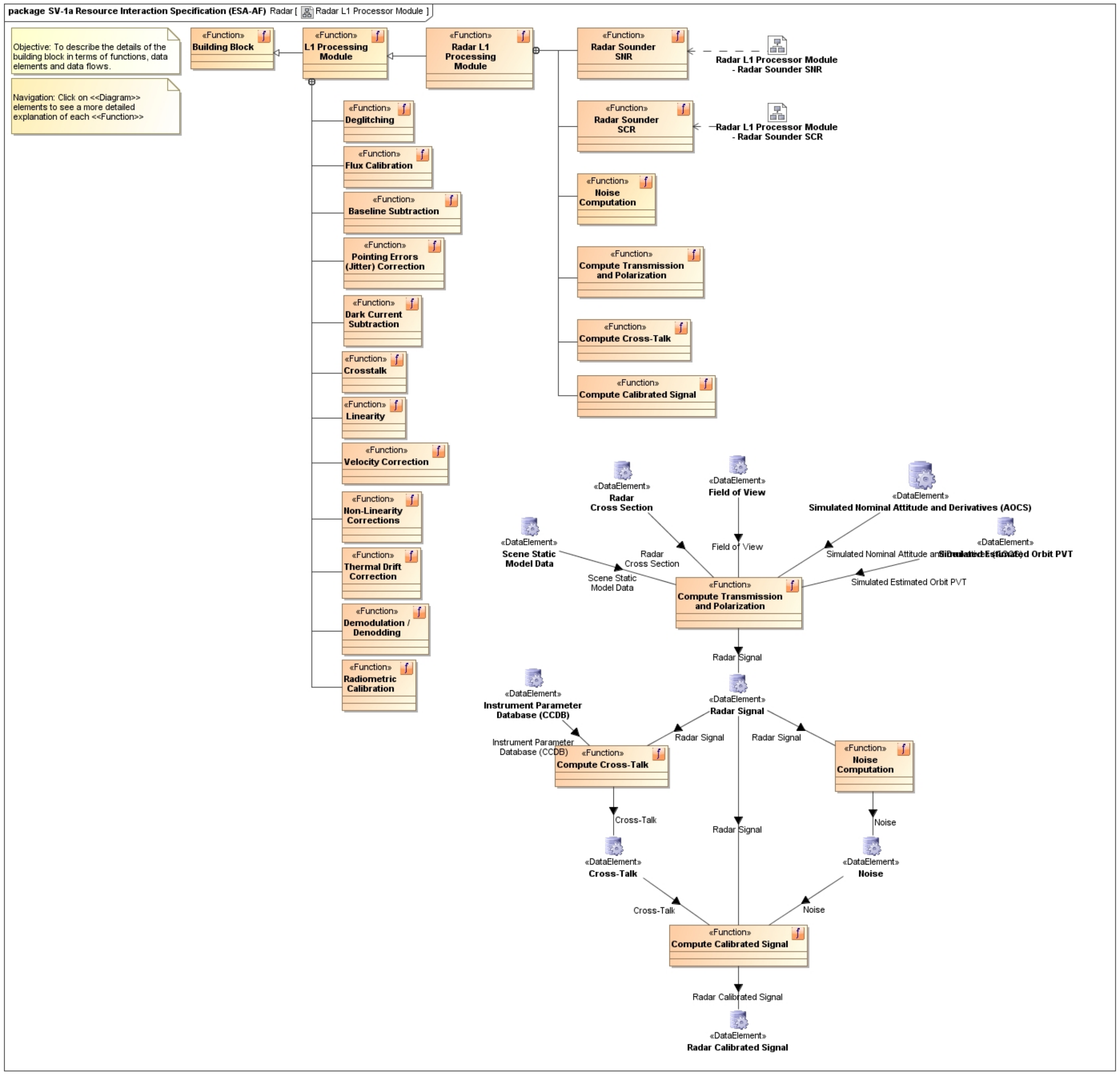
Approach to Building Blocks Definition

- Uses directly the outcome of the Commonalities TN and the structure of the Reference Architecture
- Populates the Reference Architecture with all the Processing Modules found in Space Science documentation
- Largest effort was in analyzing a very extensive bibliography and in trying to harmonize the definition of the Building Blocks
  - In level of detail
  - In template for the definition – here the existence of a clear rule list in the RA already eased the process

Model Structure

1. **Spacecraft Geometry** blocks, also responsible for instrument mounting geometry
2. **Scene Creation** blocks, responsible for simulating the physical processes for each instrument
3. **Forward Model** blocks, these include the computation of the Field-of-View (FOV) and the dynamic simulation of the scene as observed by the instrument
4. **Instrument Model** blocks, responsible for instrument response simulation
5. **Level 0 to 0.5 Processing** blocks, for telemetry and raw data handling
6. **Level 0.5 to 1 Processing** blocks, for calibration and data corrections
7. **Level 1 to 2 Processing** blocks, for scientific observables generation for each instrument
8. **Level 2 to 3 Processing** blocks, for further science extraction from combination of L1 products and instrument outputs
9. **Performance Evaluation**, defining the most common Data Analysis methods
10. **Utilities**, basic libraries and functions (interpolations, linear algebra, etc.)

Building Block Diagrams



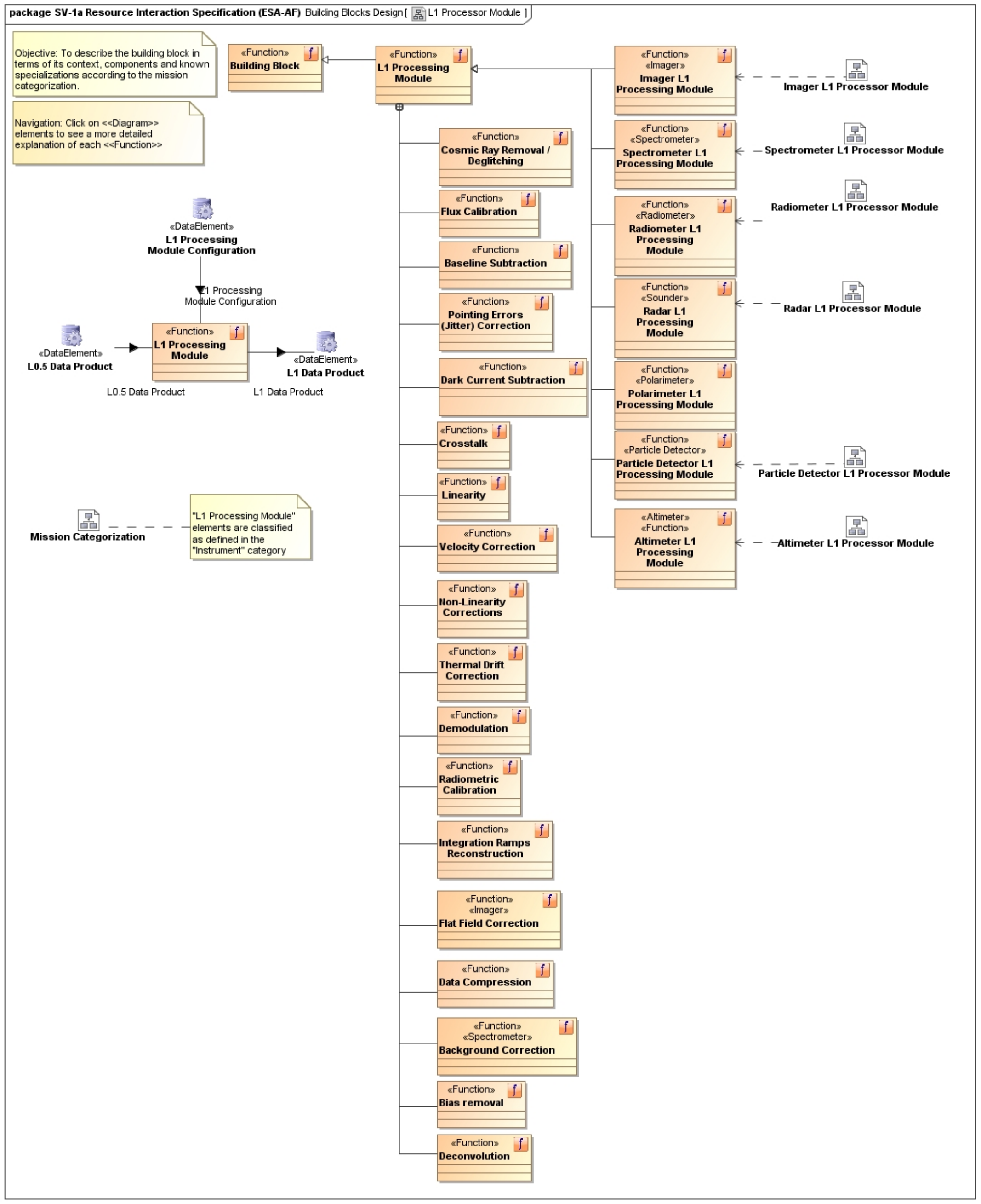
package SV-1a Resource Interaction Specification (ESA-AF) Radar [Radar L1 Processor Module]
Objective: To describe the details of the building block in terms of functions, data elements and data flows.
Navigation: Click on <<Diagram>> elements to see a more detailed explanation of each <<Function>>
«Function» Building Block
«Function» L1 Processing Module
«Function» Radar L1 Processing Module
«Function» Radar Sounder SNR
Radar L1 Processor Module - Radar Sounder SNR
«Function» Radar Sounder SCR
Radar L1 Processor Module - Radar Sounder SCR
«Function» Noise Computation
«Function» Compute Transmission and Polarization
«Function» Compute Cross-Talk
«Function» Compute Calibrated Signal
«Function» Deglitching
«Function» Flux Calibration
«Function» Baseline Subtraction
«Function» Pointing Errors (Jitter) Correction
«Function» Dark Current Subtraction
«Function» Crosstalk
«Function» Linearity
«Function» Velocity Correction
«Function» Non-Linearity Corrections
«Function» Thermal Drift Correction
«Function» Demodulation / Denodding
«Function» Radiometric Calibration
«DataElement» Radar Cross Section
«DataElement» Field of View
«DataElement» Simulated Nominal Attitude and Derivatives (AOCS)
«DataElement» Scene Static Model Data
«DataElement» Simulated Estimated Orbit PVT
Field of View
Radar Cross Section
Simulated Nominal Attitude and Derivatives (AOCS)
Scene Static Model Data
Simulated Estimated Orbit PVT
«Function» Compute Transmission and Polarization
Radar Signal
«DataElement» Instrument Parameter Database (CCDB)
«DataElement» Radar Signal
Instrument Parameter Database (CCDB)
Radar Signal
Radar Signal
«Function» Compute Cross-Talk
«Function» Noise Computation
Cross-Talk
Radar Signal
Noise
«DataElement» Cross-Talk
«DataElement» Noise
Cross-Talk
Noise
«Function» Compute Calibrated Signal
Radar Calibrated Signal
«DataElement» Radar Calibrated Signal

- Component Description Based on ESA-AF standard
- Integrated into ESA-AF based SS-E2ES Reference Architecture
- All Components and Data Types available in electronic version of RA

Example: L0.5 to L1 Processing Model

L0.5 to L1 Data Processing Model has been completed for almost all Instrument Classes.

It is the one with the largest number of Building Blocks, especially for Imagers and Spectrometers, the most prevalent categories in Space Science mission.



package SV-1a Resource Interaction Specification (ESA-AF) Building Blocks Design [L1 Processor Module]
Objective: To describe the building block in terms of its context, components and known specializations according to the mission categorization.
Navigation: Click on <<Diagram>> elements to see a more detailed explanation of each <<Function>>
«Function» Building Block
«Function» L1 Processing Module
«DataElement» L1 Processing Module Configuration
L1 Processing Module Configuration
«DataElement» L0.5 Data Product
«Function» L1 Processing Module
«DataElement» L1 Data Product
L0.5 Data Product
L1 Data Product
Mission Categorization
"L1 Processing Module" elements are classified as defined in the "Instrument" category
«Function» Cosmic Ray Removal / Deglitching
«Function» Flux Calibration
«Function» Baseline Subtraction
«Function» Pointing Errors (Jitter) Correction
«Function» Dark Current Subtraction
«Function» Crosstalk
«Function» Linearity
«Function» Velocity Correction
«Function» Non-Linearity Corrections
«Function» Thermal Drift Correction
«Function» Demodulation
«Function» Radiometric Calibration
«Function» Integration Ramps Reconstruction
«Function» «Imager» Flat Field Correction
«Function» Data Compression
«Function» «Spectrometer» Background Correction
«Function» Bias removal
«Function» Deconvolution
«Function» «Imager» Imager L1 Processing Module
«Function» «Spectrometer» Spectrometer L1 Processing Module
«Function» «Radiometer» Radiometer L1 Processing Module
«Function» «Sounder» Radar L1 Processing Module
«Function» «Polarimeter» Polarimeter L1 Processing Module
«Function» «Particle Detector» Particle Detector L1 Processing Module
«Altimeter» «Function» Altimeter L1 Processing Module
Imager L1 Processor Module
Spectrometer L1 Processor Module
Radiometer L1 Processor Module
Radar L1 Processor Module
Particle Detector L1 Processor Module
Altimeter L1 Processor Module

Most Generic Building Blocks

After the exercise was complete, it was interesting to find out which were the most common Building Blocks:

Processing Level	Building Block
Geometry Simulation	Orbit Simulator
	Attitude Simulator
	Instrument Pointing Simulator
Scene Creation	Generic Blocks
Forward Model	Scene Interaction Geometry
	Stimuli Generation
Instrument Model	Optics Building Block
	L0 Formatter
L0 to L0.5 Processing	Unpack Telemetry
	Decompression
	Sorting
	Repackaging
	Add Auxiliary Data
	Unit Conversion
	Time Correction/Conversion
	Masking
	Data Extraction & Quality Control
	Measurement Pre-Processing
	Time Domain Integration

Most Generic Building Blocks

Processing Level	Building Block	
L0.5 to L1 Processing	Cosmic Ray Removal / Deglitching	
	Integration Ramps Reconstruction	
	Flux Calibration	
	Baseline Subtraction	
	Pointing Errors (Jitter) Compensation	
	Dark Current Subtraction	
	Crosstalk	
	Linearity	
	Velocity Correction	
	Non-Linearity Correction	
	Thermal Drift Corrections	
	Demodulation	
	Radiometric Calibration	
	Decompression	
L1 to L2 Processing	Projection	
	Mapmaking	
	Weighted Averaging	
	Regridding	
	Denoising	
	Deconvolution	
	Pixel Flagging	
L2 to L3 Processing	Source Extraction	
	Mosaicking	Resampling
		Co-Registration
		Tie Point Selection
		Tie Point Matching
		WCS Projection
		Deregistration Estimation Model
		Pixel Classification

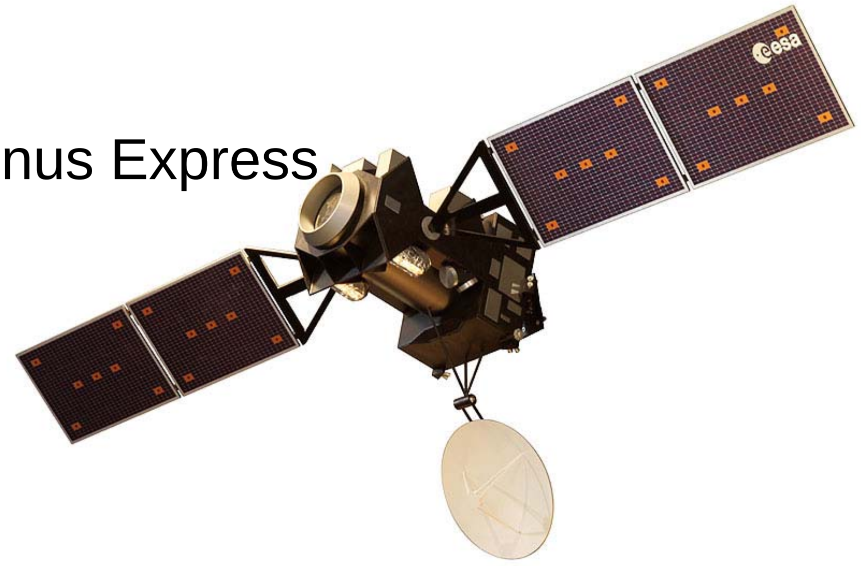
RA Application to ExoMars and Solar Orbiter

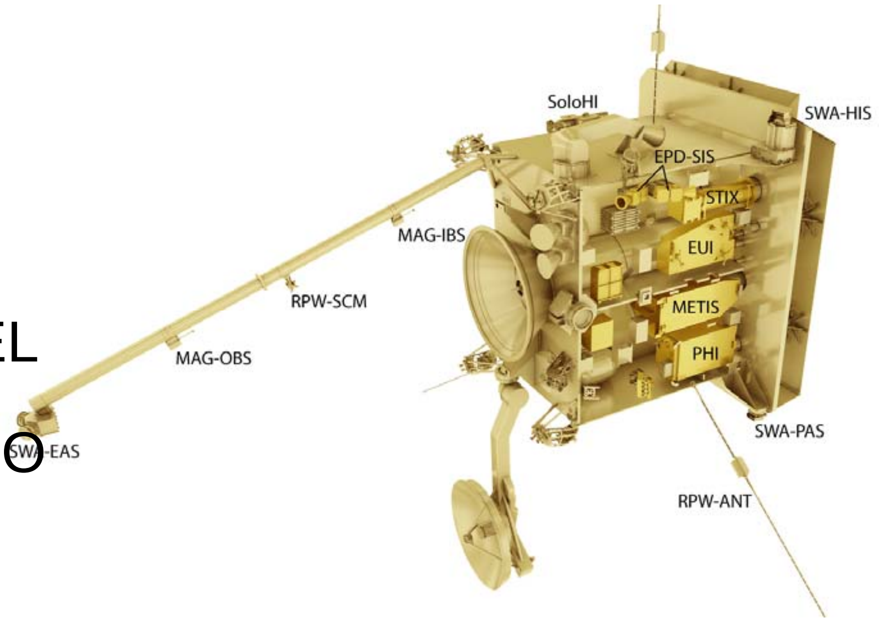
Approach to Application of RA

- Initial exercise of Mission Selection analysed 3 missions
  - Two multi-instrument missions selected, ExoMars and Solar Orbiter
- The Requirements were defined for each mission – 90% were re-used from the RA
- Reference Architecture was pruned, adapted and, in rare cases, expanded to properly define both Specific Architectures
- Main effort was spent in trying to use as many Generic Blocks as possible, in a meaningful way, not in defining very specialized blocks
- The few new Generic Blocks found in the exercise were re-imported into the Generic Building Blocks Document
  - Example are the BBs for Particle Detectors coming from the ExoMars FREND instrument or Data Compression from Solar Orbiter pipelines

Selected Missions

- **ExoMars** (instrument heritage):
  - NOMAD SO/LNO: SOIR from Venus Express
  - ACS TIR: Mars Express PFS
  - CaSSIS: OSIRIS from Rosetta
- **Solar Orbiter** (instrument heritage):
  - PHI: SUNRISE/IMaX
  - EUI: EUVI from STEREO
  - STIX: RHESSI
  - METIS: SCORE from HERSCHEL
  - SoloHI: HI1 and HI2 from STEREO
  - SPICE: CDS from SUMER



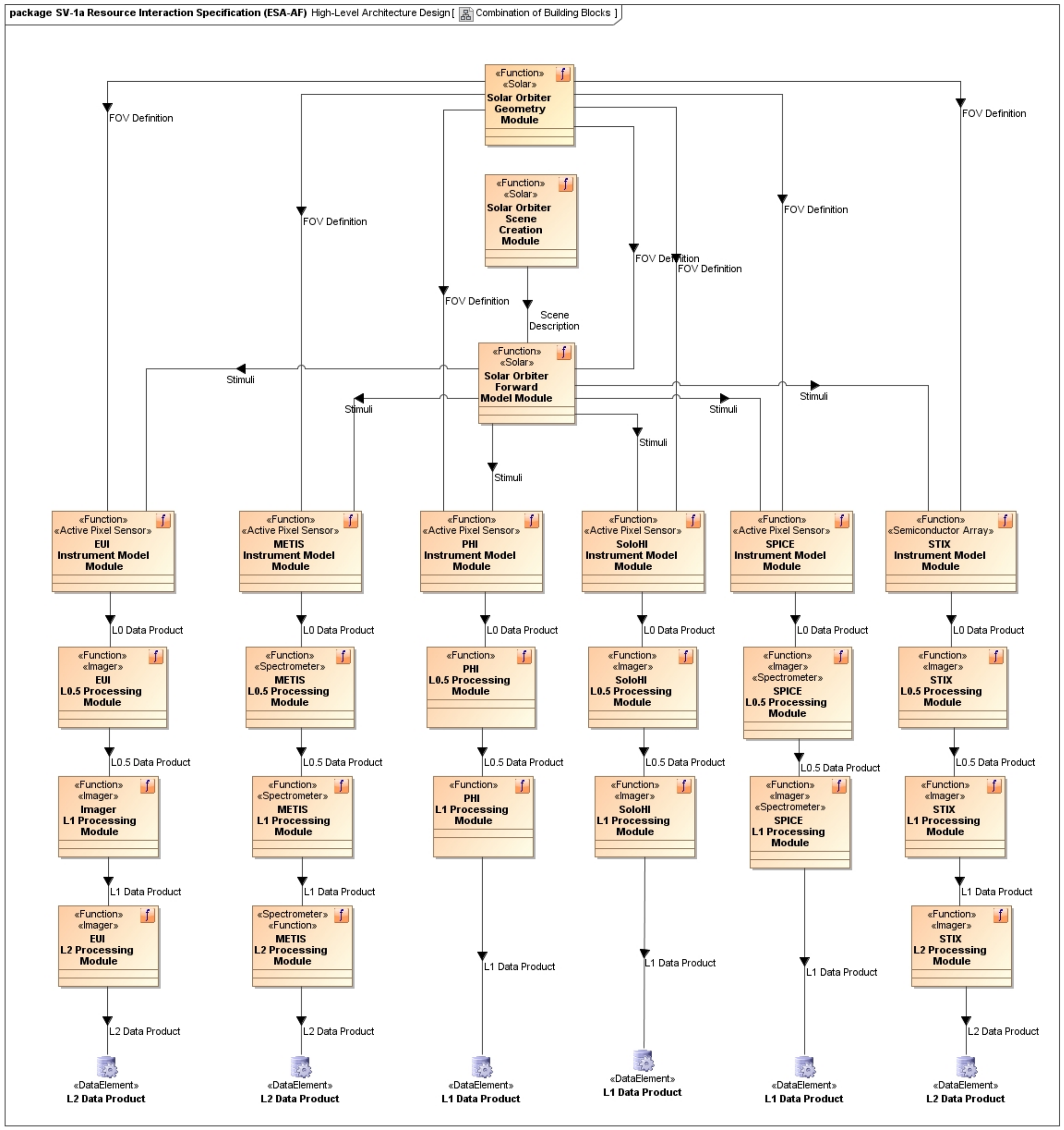


Solar Orbiter overall Pipeline

No combination of Pipelines at the level of routine processing (co-alignemet exercises will not be processed in the Pipeline).

Only some Instruments have L3 BBs (in our definition)

PHI is almost entirely processed onboard but the BBs are still



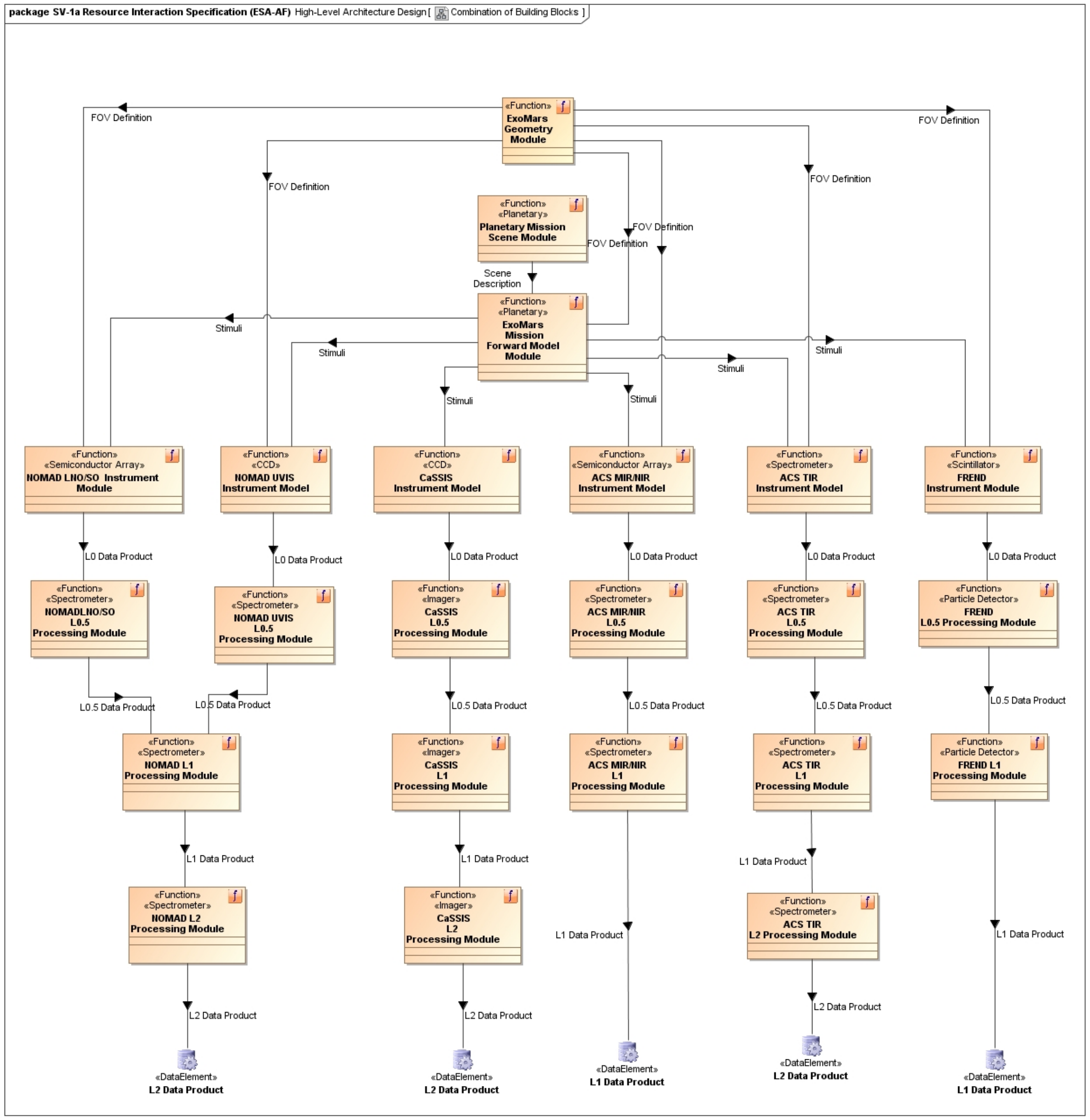
package SV-1a Resource Interaction Specification (ESA-AF) High-Level Architecture Design [Combination of Building Blocks]
«Function» «Solar» Solar Orbiter Geometry Module
«Function» «Solar» Solar Orbiter Scene Creation Module
«Function» «Solar» Solar Orbiter Forward Model Module
FOV Definition
Scene Description
Stimuli
«Function» «Active Pixel Sensor» EUI Instrument Model Module
«Function» «Active Pixel Sensor» METIS Instrument Model Module
«Function» «Active Pixel Sensor» PHI Instrument Model Module
«Function» «Active Pixel Sensor» SoloHI Instrument Model Module
«Function» «Active Pixel Sensor» SPICE Instrument Model Module
«Function» «Semiconductor Array» STIX Instrument Model Module
L0 Data Product
«Function» «Imager» EUI L0.5 Processing Module
«Function» «Spectrometer» METIS L0.5 Processing Module
«Function» PHI L0.5 Processing Module
«Function» «Imager» SoloHI L0.5 Processing Module
«Function» «Imager» «Spectrometer» SPICE L0.5 Processing Module
«Function» «Imager» STIX L0.5 Processing Module
L0.5 Data Product
«Function» «Imager» Imager L1 Processing Module
«Function» «Spectrometer» METIS L1 Processing Module
«Function» PHI L1 Processing Module
«Function» «Imager» SoloHI L1 Processing Module
«Function» «Imager» «Spectrometer» SPICE L1 Processing Module
«Function» «Imager» STIX L1 Processing Module
L1 Data Product
«Function» «Imager» EUI L2 Processing Module
«Spectrometer» «Function» METIS L2 Processing Module
«Function» «Imager» STIX L2 Processing Module
L2 Data Product
«DataElement» L2 Data Product
«DataElement» L2 Data Product
«DataElement» L1 Data Product
«DataElement» L1 Data Product
«DataElement» L1 Data Product
«DataElement» L2 Data Product

ExoMars overall Pipeline

NOMAD LNO/SO detectors have different sensors and should be separated at L0.5 but then the Pipelines are merged.

ACS MIR/NIR are processed differently from TIR (which is a Fourier Spectrometer)

FREND is a Particle Detector and has provided information back to the generic BB definition

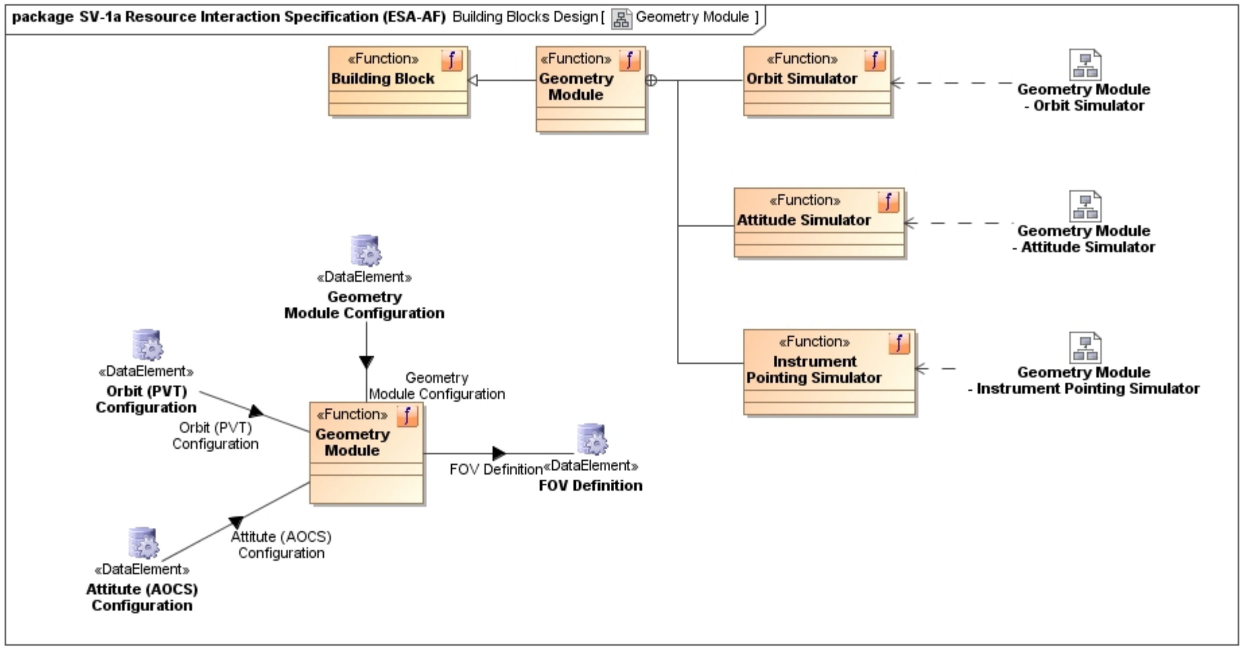


package SV-1a Resource Interaction Specification (ESA-AF) High-Level Architecture Design [Combination of Building Blocks]
«Function» ExoMars Geometry Module
FOV Definition
«Function» «Planetary» Planetary Mission Scene Module
Scene Description
«Function» «Planetary» ExoMars Mission Forward Model Module
Stimuli
«Function» «Semiconductor Array» NOMAD LNO/SO Instrument Module
«Function» «CCD» NOMAD UVIS Instrument Model
«Function» «CCD» CaSSIS Instrument Model
«Function» «Semiconductor Array» ACS MIR/NIR Instrument Model
«Function» «Spectrometer» ACS TIR Instrument Model
«Function» «Scintillator» FREND Instrument Module
L0 Data Product
«Function» «Spectrometer» NOMADLNO/SO L0.5 Processing Module
«Function» «Spectrometer» NOMAD UVIS L0.5 Processing Module
«Function» «Imager» CaSSIS L0.5 Processing Module
«Function» «Spectrometer» ACS MIR/NIR L0.5 Processing Module
«Function» «Spectrometer» ACS TIR L0.5 Processing Module
«Function» «Particle Detector» FREND L0.5 Processing Module
L0.5 Data Product
«Function» «Spectrometer» NOMAD L1 Processing Module
«Function» «Imager» CaSSIS L1 Processing Module
«Function» «Spectrometer» ACS MIR/NIR L1 Processing Module
«Function» «Spectrometer» ACS TIR L1 Processing Module
«Function» «Particle Detector» FREND L1 Processing Module
L1 Data Product
«Function» «Spectrometer» NOMAD L2 Processing Module
«Function» «Imager» CaSSIS L2 Processing Module
«Function» «Spectrometer» ACS TIR L2 Processing Module
L2 Data Product
«DataElement» L2 Data Product
«DataElement» L2 Data Product
«DataElement» L1 Data Product
«DataElement» L2 Data Product
«DataElement» L1 Data Product

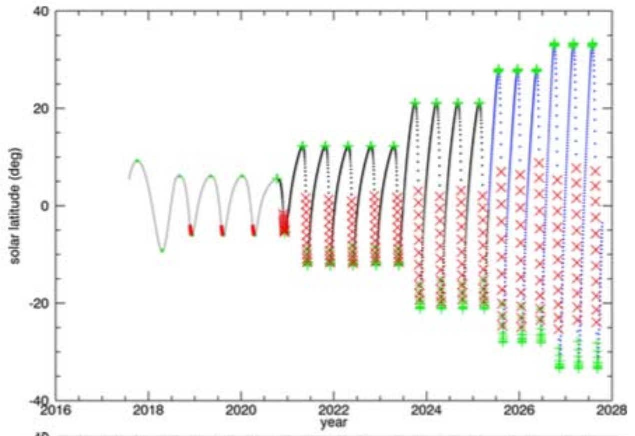
Spacecraft Geometry Module

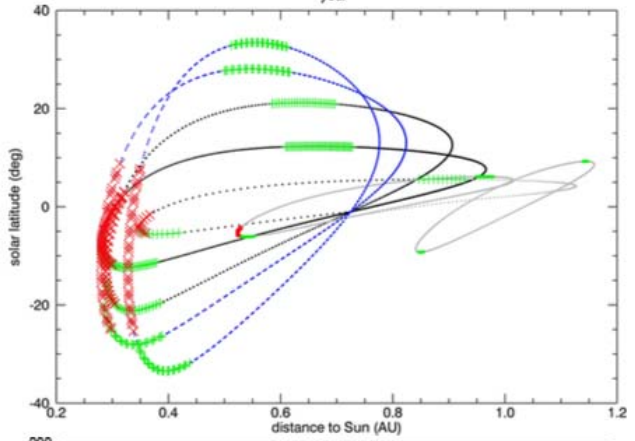
Very generic module, used equally for all the instruments in any of the the two missions and very similar between missions.

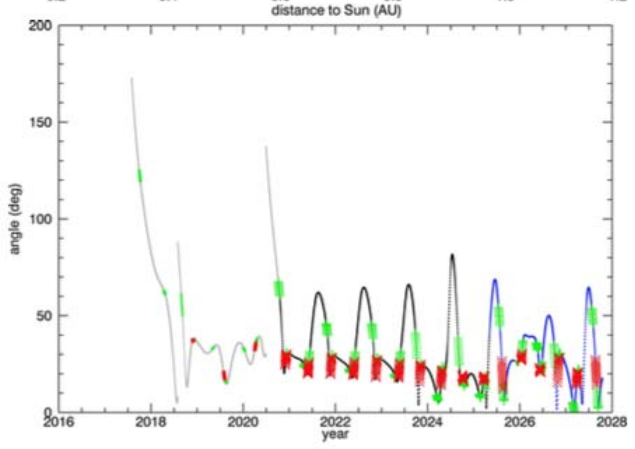
The Mission configuration is what drives the computations (right side is the Solar Orbiter orbital phases) but the BB itself is generic:



package SV-1a Resource Interaction Specification (ESA-AF) Building Blocks Design [Geometry Module]
«Function» Building Block
«Function» Geometry Module
«Function» Orbit Simulator
Geometry Module - Orbit Simulator
«Function» Attitude Simulator
Geometry Module - Attitude Simulator
«Function» Instrument Pointing Simulator
Geometry Module - Instrument Pointing Simulator
«DataElement» Geometry Module Configuration
Geometry Module Configuration
«DataElement» Orbit (PVT) Configuration
Orbit (PVT) Configuration
«Function» Geometry Module
FOV Definition
«DataElement» FOV Definition
Attitute (AOCS) Configuration
«DataElement» Attitute (AOCS) Configuration

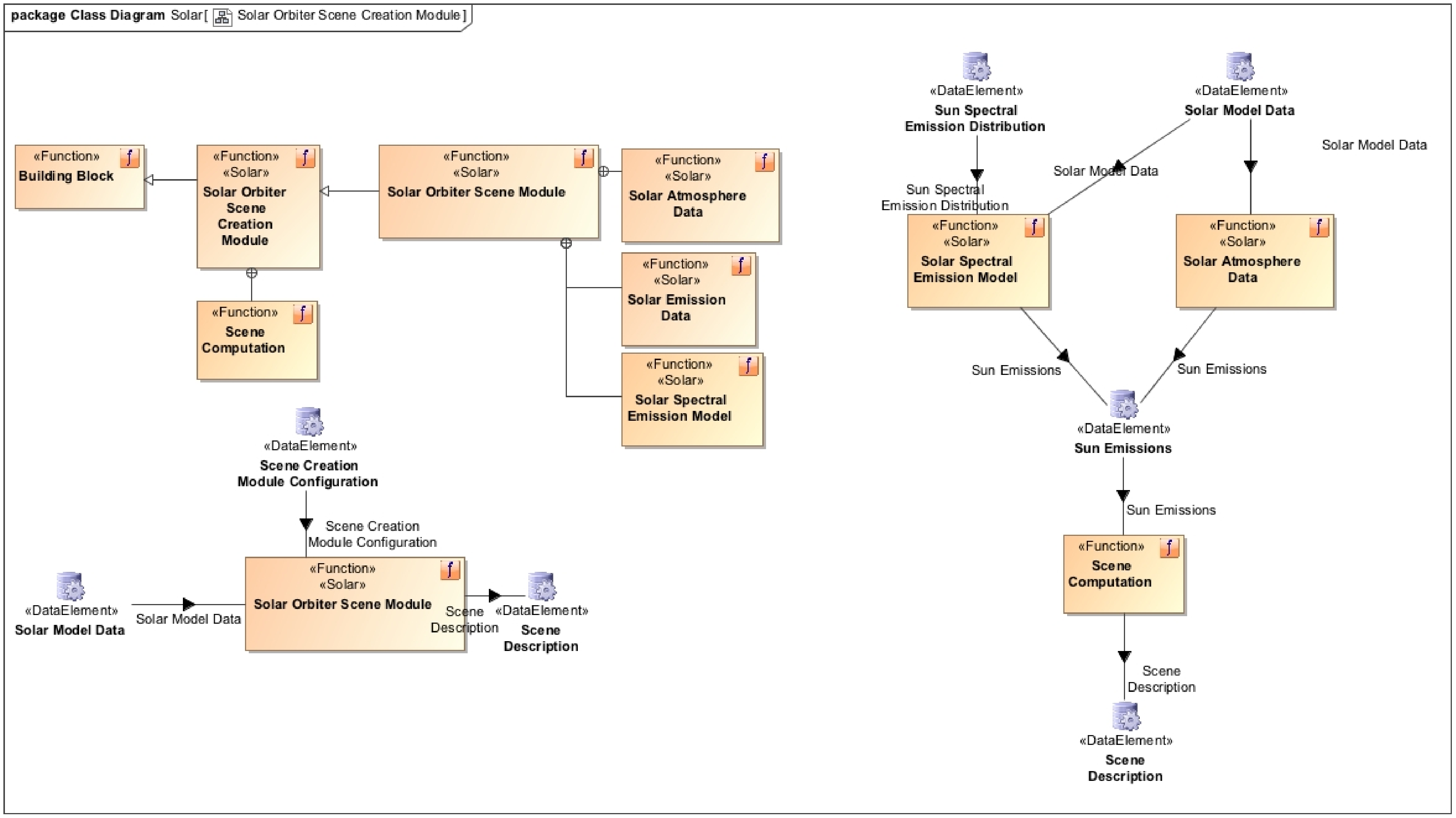






SO Scene Creation Module

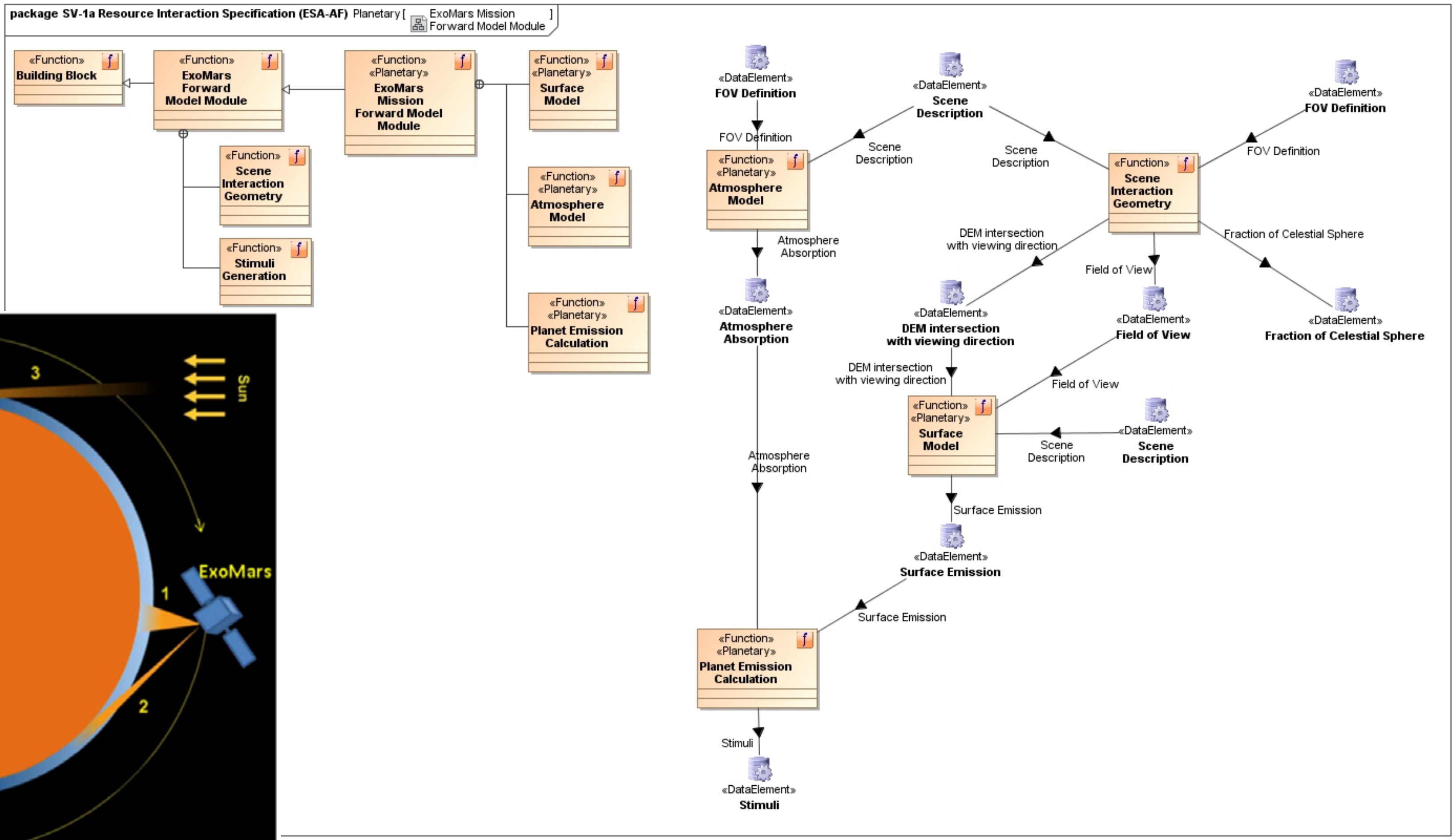
For **Solar Orbiter**, which is imaging a distant target, most of the physics simulations can be done in the Scene Creation Module:



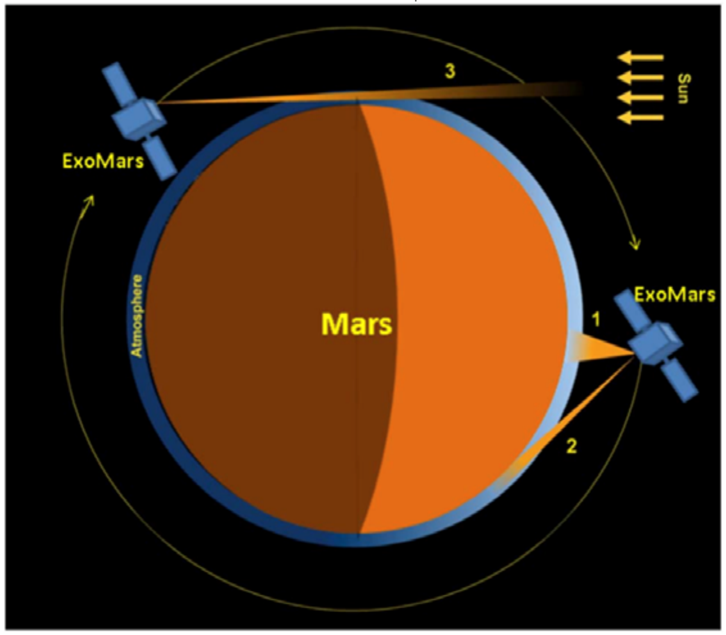
package Class Diagram Solar[Solar Orbiter Scene Creation Module]
«Function» Building Block
«Function» «Solar» Solar Orbiter Scene Creation Module
«Function» «Solar» Solar Orbiter Scene Module
«Function» «Solar» Solar Atmosphere Data
«Function» «Solar» Solar Emission Data
«Function» «Solar» Solar Spectral Emission Model
«Function» Scene Computation
«DataElement» Scene Creation Module Configuration
Scene Creation Module Configuration
«Function» «Solar» Solar Orbiter Scene Module
«DataElement» Solar Model Data
Solar Model Data
Scene Description
«DataElement» Scene Description
«DataElement» Sun Spectral Emission Distribution
«DataElement» Solar Model Data
Solar Model Data
Solar Model Data
Sun Spectral Emission Distribution
«Function» «Solar» Solar Spectral Emission Model
«Function» «Solar» Solar Atmosphere Data
Sun Emissions
Sun Emissions
«DataElement» Sun Emissions
Sun Emissions
«Function» Scene Computation
Scene Description
«DataElement» Scene Description

ExoMars Forward Model

For **ExoMars**, observing a close target, the details of the physics simulations depend on the observation geometry (atmospheric paths, surface illumination, DEM intersection with viewing angles) and so they have to be simulated mostly in the Forward Model

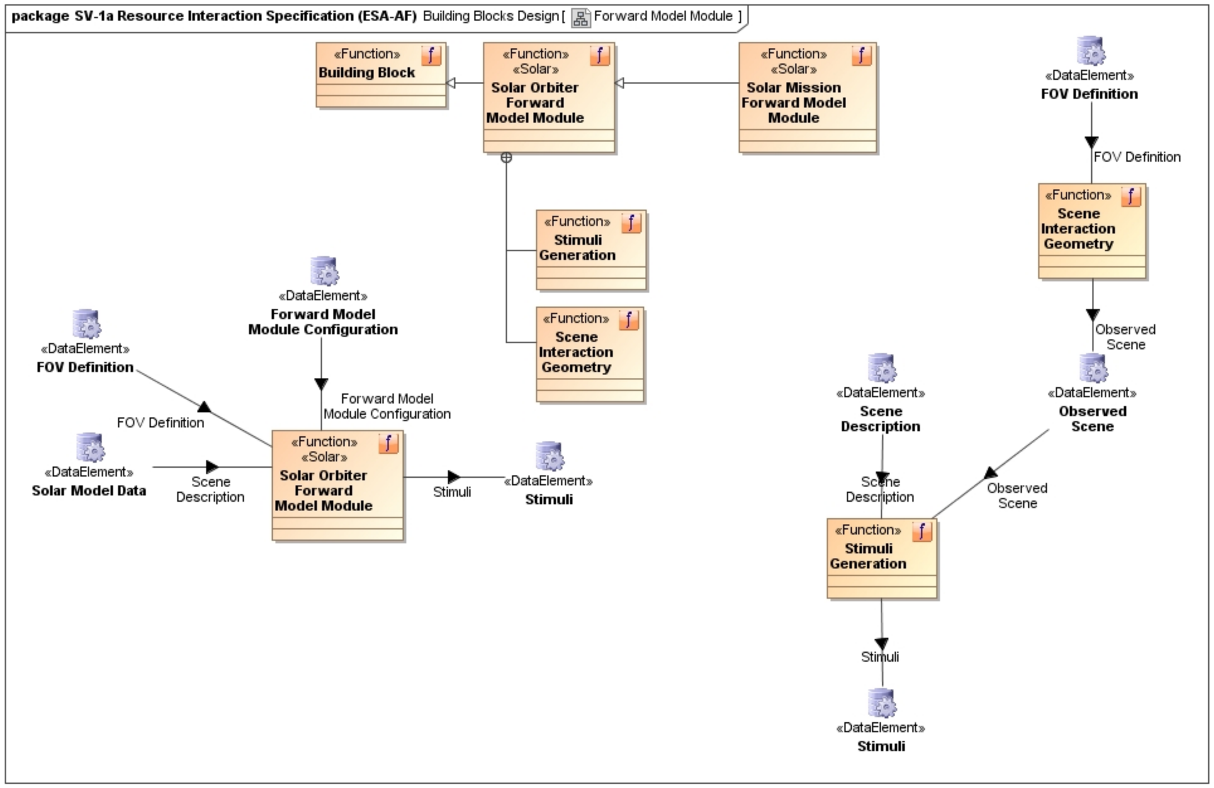


package SV-1a Resource Interaction Specification (ESA-AF) Planetary [ExoMars Mission Forward Model Module]
«Function» Building Block
«Function» ExoMars Forward Model Module
«Function» «Planetary» ExoMars Mission Forward Model Module
«Function» «Planetary» Surface Model
«Function» Scene Interaction Geometry
«Function» «Planetary» Atmosphere Model
«Function» Stimuli Generation
«Function» «Planetary» Planet Emission Calculation
«DataElement» FOV Definition
FOV Definition
«DataElement» Scene Description
Scene Description
Scene Description
«DataElement» FOV Definition
FOV Definition
«Function» «Planetary» Atmosphere Model
«Function» Scene Interaction Geometry
Atmosphere Absorption
DEM intersection with viewing direction
Fraction of Celestial Sphere
Field of View
«DataElement» Atmosphere Absorption
«DataElement» DEM intersection with viewing direction
«DataElement» Field of View
«DataElement» Fraction of Celestial Sphere
DEM intersection with viewing direction
Field of View
«Function» «Planetary» Surface Model
Scene Description
«DataElement» Scene Description
Atmosphere Absorption
Surface Emission
«DataElement» Surface Emission
Surface Emission
«Function» «Planetary» Planet Emission Calculation
Stimuli
«DataElement» Stimuli

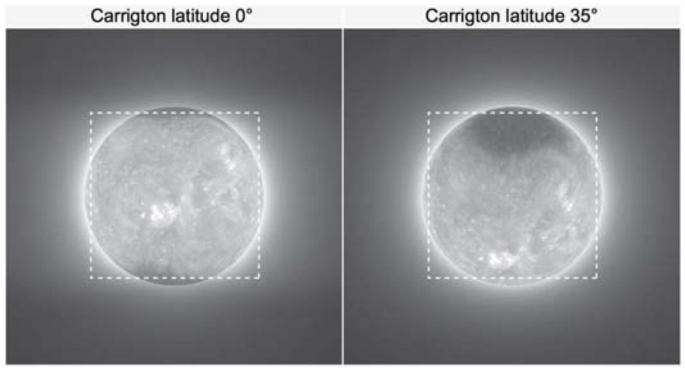


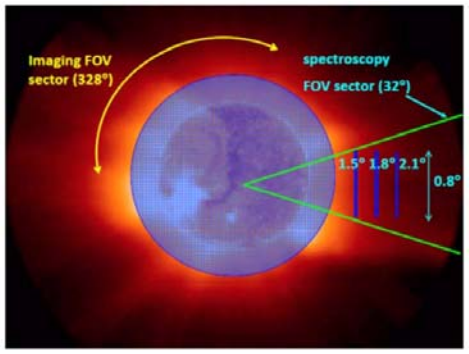
SO Forward Model

In **Solar Orbiter**, the Forward Model, while completely generic, has to take into account the very different FOVs of each instrument, which are defined as configuration parameters in our architecture:

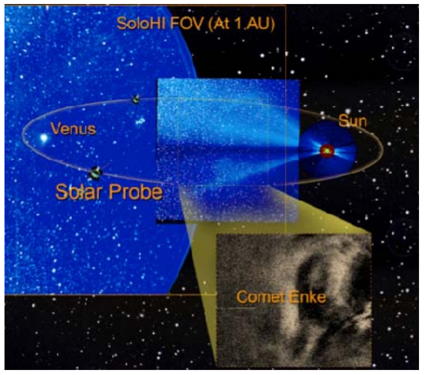

package SV-1a Resource Interaction Specification (ESA-AF) Building Blocks Design [Forward Model Module]
«Function» Building Block
«Function» «Solar» Solar Orbiter Forward Model Module
«Function» «Solar» Solar Mission Forward Model Module
«DataElement» FOV Definition
FOV Definition
«Function» Scene Interaction Geometry
Observed Scene
«Function» Stimuli Generation
«Function» Scene Interaction Geometry
«DataElement» Forward Model Module Configuration
Forward Model Module Configuration
«DataElement» FOV Definition
FOV Definition
«DataElement» Solar Model Data
Scene Description
«Function» «Solar» Solar Orbiter Forward Model Module
Stimuli
«DataElement» Stimuli
«DataElement» Scene Description
«DataElement» Observed Scene
Scene Description
Observed Scene
«Function» Stimuli Generation
Stimuli
«DataElement» Stimuli

PHI FSI





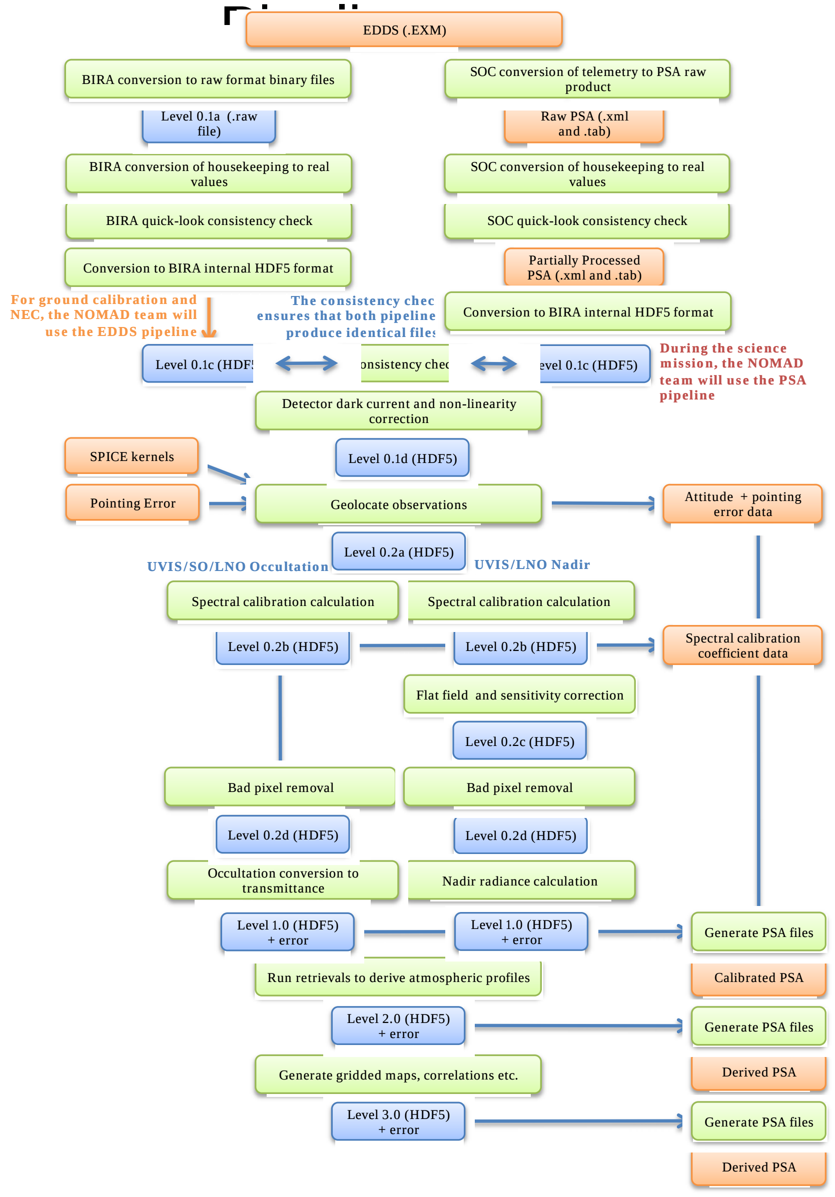
METIS



SoloHI

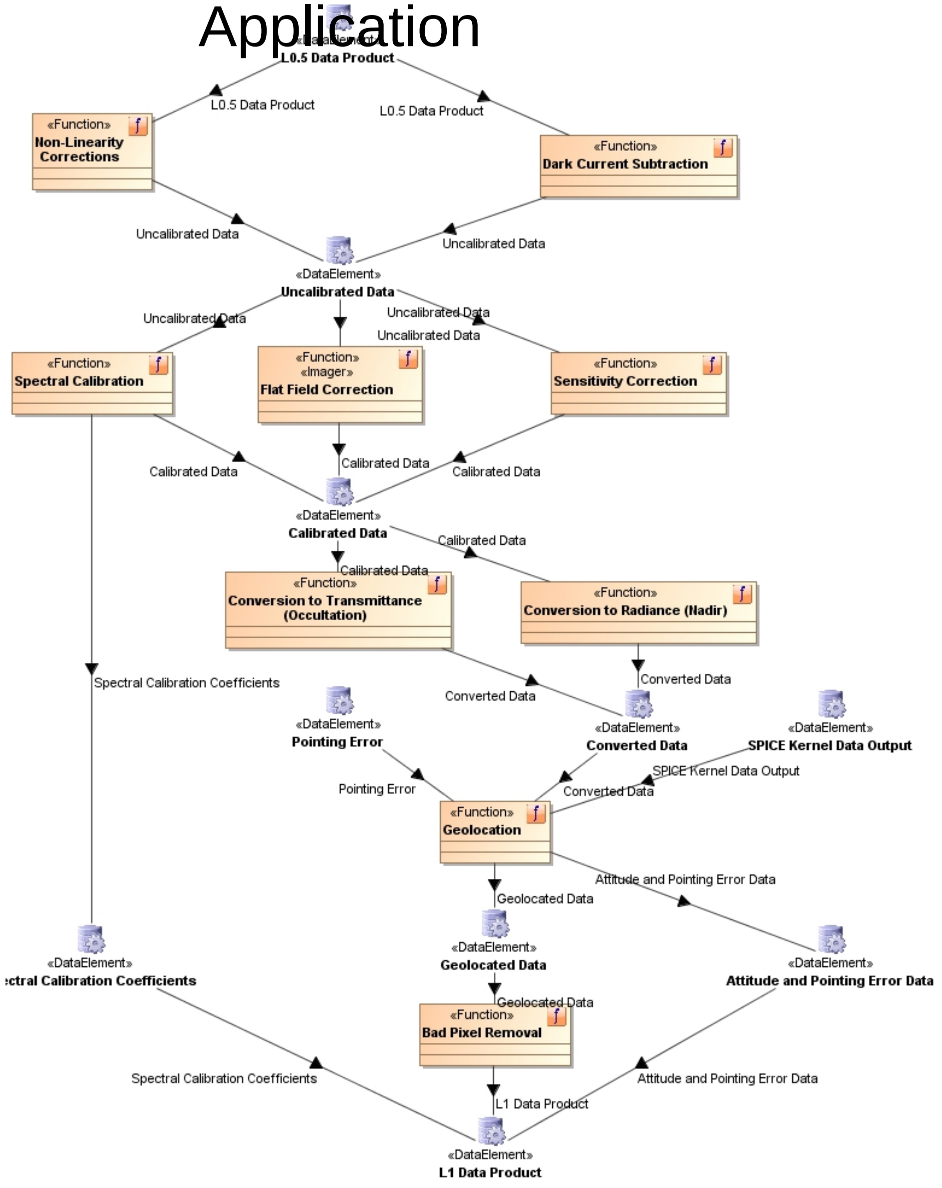
Example for NOMAD Pipeline

Original Pipeli



EDDS (.EXM)
BIRA conversion to raw format binary files
SOC conversion of telemetry to PSA raw product
Level 0.1a (.raw file)
Raw PSA (.xml and .tab)
BIRA conversion of housekeeping to real values
SOC conversion of housekeeping to real values
BIRA quick-look consistency check
SOC quick-look consistency check
Conversion to BIRA internal HDF5 format
Partially Processed PSA (.xml and .tab)
For ground calibration and NEC, the NOMAD team will use the EDDS pipeline
The consistency check ensures that both pipelines produce identical files
Conversion to BIRA internal HDF5 format
Level 0.1c (HDF5)
Consistency check
Level 0.1c (HDF5)
During the science mission, the NOMAD team will use the PSA pipeline
Detector dark current and non-linearity correction
SPICE kernels
Level 0.1d (HDF5)
Pointing Error
Geolocate observations
Attitude + pointing error data
Level 0.2a (HDF5)
UVIS/SO/LNO Occultation
UVIS/LNO Nadir
Spectral calibration calculation
Spectral calibration calculation
Level 0.2b (HDF5)
Level 0.2b (HDF5)
Spectral calibration coefficient data
Flat field and sensitivity correction
Level 0.2c (HDF5)
Bad pixel removal
Bad pixel removal
Level 0.2d (HDF5)
Level 0.2d (HDF5)
Occultation conversion to transmittance
Nadir radiance calculation
Level 1.0 (HDF5) + error
Level 1.0 (HDF5) + error
Generate PSA files
Run retrievals to derive atmospheric profiles
Calibrated PSA
Level 2.0 (HDF5) + error
Generate PSA files
Generate gridded maps, correlations etc.
Derived PSA
Level 3.0 (HDF5) + error
Generate PSA files
Derived PSA

After RA Application

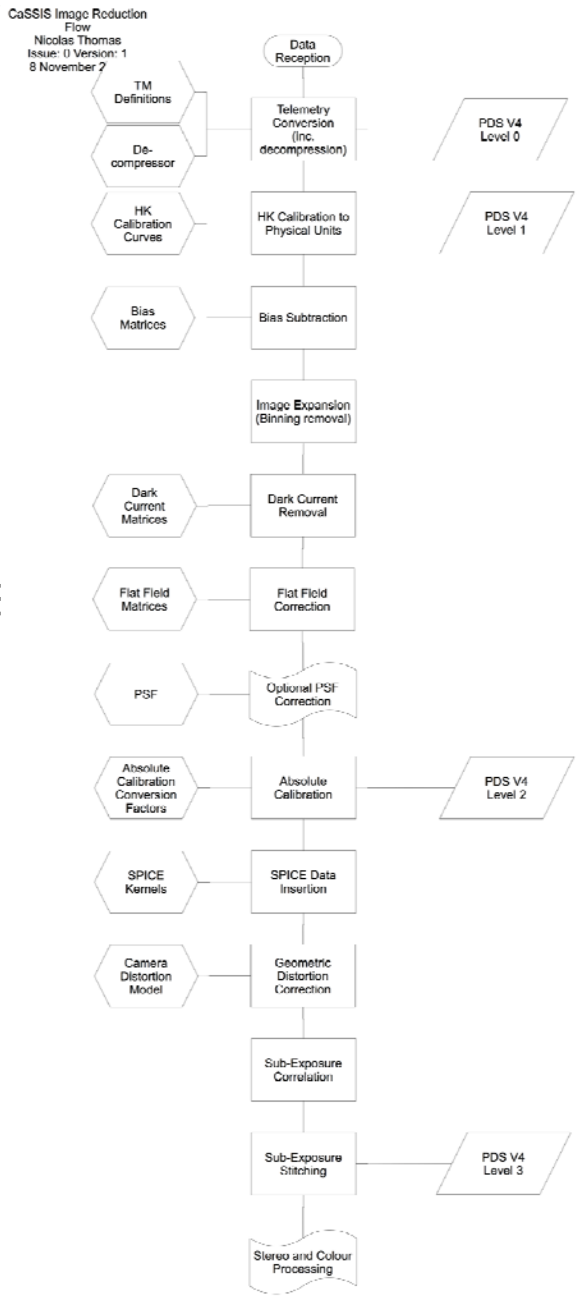


«DataElement» L0.5 Data Product
L0.5 Data Product
L0.5 Data Product
«Function» Non-Linearity Corrections
«Function» Dark Current Subtraction
Uncalibrated Data
Uncalibrated Data
«DataElement» Uncalibrated Data
Uncalibrated Data
Uncalibrated Data
Uncalibrated Data
«Function» Spectral Calibration
«Function» «Imager» Flat Field Correction
«Function» Sensitivity Correction
Calibrated Data
Calibrated Data
Calibrated Data
«DataElement» Calibrated Data
Calibrated Data
Calibrated Data
«Function» Conversion to Transmittance (Occultation)
«Function» Conversion to Radiance (Nadir)
Spectral Calibration Coefficients
Converted Data
Converted Data
«DataElement» Pointing Error
«DataElement» Converted Data
«DataElement» SPICE Kernel Data Output
SPICE Kernel Data Output
Pointing Error
Converted Data
«Function» Geolocation
Attitude and Pointing Error Data
Geolocated Data
«DataElement» Spectral Calibration Coefficients
«DataElement» Geolocated Data
«DataElement» Attitude and Pointing Error Data
Geolocated Data
«Function» Bad Pixel Removal
Spectral Calibration Coefficients
Attitude and Pointing Error Data
L1 Data Product
«DataElement» L1 Data Product

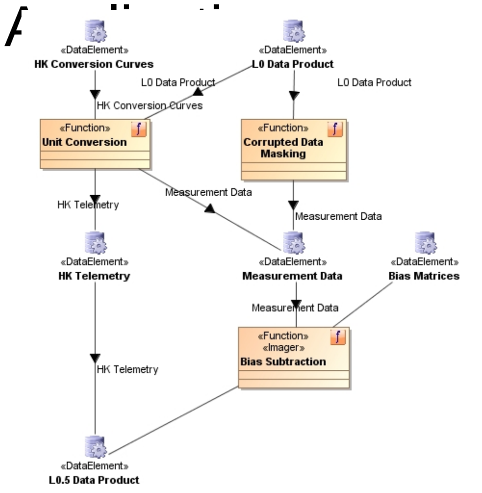
Example for CaSSIS Pipeline

Origina

Pipeline

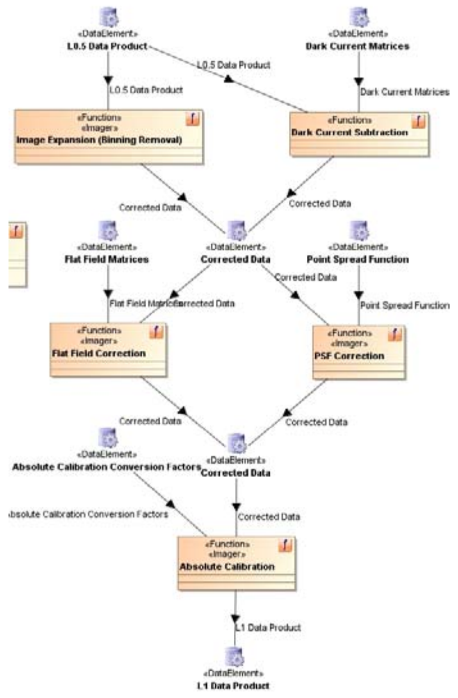


After RA



«DataElement» HK Conversion Curves
«DataElement» L0 Data Product
L0 Data Product
L0 Data Product
HK Conversion Curves
«Function» Unit Conversion
«Function» Corrupted Data Masking
Measurement Data
HK Telemetry
Measurement Data
«DataElement» HK Telemetry
«DataElement» Measurement Data
«DataElement» Bias Matrices
Measurement Data
«Function» «Imager» Bias Subtraction
HK Telemetry
«DataElement» L0.5 Data Product

L0 – L0.5



«DataElement» L0.5 Data Product
«DataElement» Dark Current Matrices
L0.5 Data Product
L0.5 Data Product
Dark Current Matrices
«Function» «Imager» Image Expansion (Binning Removal)
«Function» Dark Current Subtraction
Corrected Data
Corrected Data
«DataElement» Flat Field Matrices
«DataElement» Corrected Data
«DataElement» Point Spread Function
Corrected Data
Flat Field Matrices
Corrected Data
Point Spread Function
«Function» «Imager» Flat Field Correction
«Function» «Imager» PSF Correction
Corrected Data
Corrected Data
«DataElement» Absolute Calibration Conversion Factors
«DataElement» Corrected Data
Absolute Calibration Conversion Factors
Corrected Data
«Function» «Imager» Absolute Calibration
L1 Data Product
«DataElement» L1 Data Product

L0.5 – L1

Conclusions

- **High degree of commonality**: During the application of the RA, it was clear that many of the Building Blocks defined in the previous stages of the contract were very common, especially the Orbit/Geometry simulation and the L0.5 Processing Modules.
- **RAs defined with mostly generic Building Blocks**: In both cases we try to specify as much as possible the Pipelines using generic blocks. This meant that sometimes the only thing common between block of the same name across instruments is the concept of the BB and not its implementation, which can be completely different. The idea here was to find also the places where the interfaces can be standardized.
- **Many specific Blocks defined**: Nevertheless, in many cases a full specification was done (as is the case for SPICE or FREND). The objective in this case was to evaluate in-depth how much effort did the RA existence really save.

Reference Architecture Concept Evaluation and Roadmap

Evaluation of RA Application and Roadmap

- First defined a set of criteria to analyse the RA advantages
  - Tailored the ones defined in the previous ARCHEO contract
- Analysed the different history of E2E Simulators in EO and SS
- Evaluated quantitatively the effort saved when an RA is available and listed its many advantages
- Proposed many ideas for future activities
- Tried to make a strong case for E2ES in SS, suggesting also a least-resistance path

Identified Benefits

- Principally in the areas of standardisation and reuse:
  - **Terminology**. Different missions and instruments may use different terms for the same thing, or even worse, the same term for different things. Providing a reference architecture will help to promote standard terminology.
  - **Requirements**. There are a set of requirements that will be applicable to all mission simulators. The reference architecture will reduce the effort to identify them and help to avoid missing important ones.
  - **Design**. The same fundamental design can be applied to all missions. Software architectural design is difficult and a simulator, at least in the early stages, may well be implemented by scientists rather than professional software engineers. A solid and proven design will be of great benefit here.
  - **Interfaces**. The interfaces between simulation stages would be defined by the RA. The format and structure of the exchanged files would be also provided, meaning no time would be needed to design them.
  - **Implementation**. Some modules may have a ready-made implementation and be ready to use by appropriately setting their configuration parameters to tailor their behaviour to the mission. While it must be understood that there is always likely to be some tailoring needed for mission specifics, reuse of standard building blocks and libraries has great potential to stimulate productivity and significantly reduce the cost of development.

Proposed Approach

- In phase 0/A, develop a simulator using the RA. This is where the benefit of a standard architecture and reusable building blocks is clearest.

- In phase B/C, continue to develop and maintain the simulator. At this point it should be brought in line with good software engineering practices (if not already). The data processing retrieval modules become the initial code for the DP pipeline. We then have a ready-made framework in which to evolve the operational pipeline. Generic algorithms start to be replaced by specific ones tuned to mission and instrument knowledge.

- In phase D/E, continue to refine the algorithms with operation knowledge. The RA remains valid. The pipeline code is *branched* for operational deployment and development can continue using the simulator environment. Environment is also used in pipeline validation.

Conclusions

Project Outcome

- The project has **structured a very sizeable quantity of information** and provided some **unexpected outputs**:
  - **Space Science mission database**, organized by several different categories
  - **Full Reference Architecture Model** in a standardized Architectural Framework, **editable** and suitable for **online reference** (we also produced a **Cookbook** for users)
  - A **full set of Software Requirements** that can be re-used for almost **any SS project**
- The application of the RA to the two selected missions has allowed us to draw **several new conclusions** and make a **strong case for the usage of E2ES in Space Science** and, at the same time, reduce substantially the barriers for its design and implementation

Main Conclusions

- E2E simulators improve the science return by
  - Optimising scientific performance by providing early efficient analysis capabilities
  - Saving time and effort on pipeline development and testing

- Benefits of an E2E simulator Reference Architecture:
  - Clearest in phase 0/A.
  - Standardisation of terminology and design.
  - Reusable Building Blocks.
  - Saves further development time (up to 50% in our case study)

End of Presentation

Thank you!

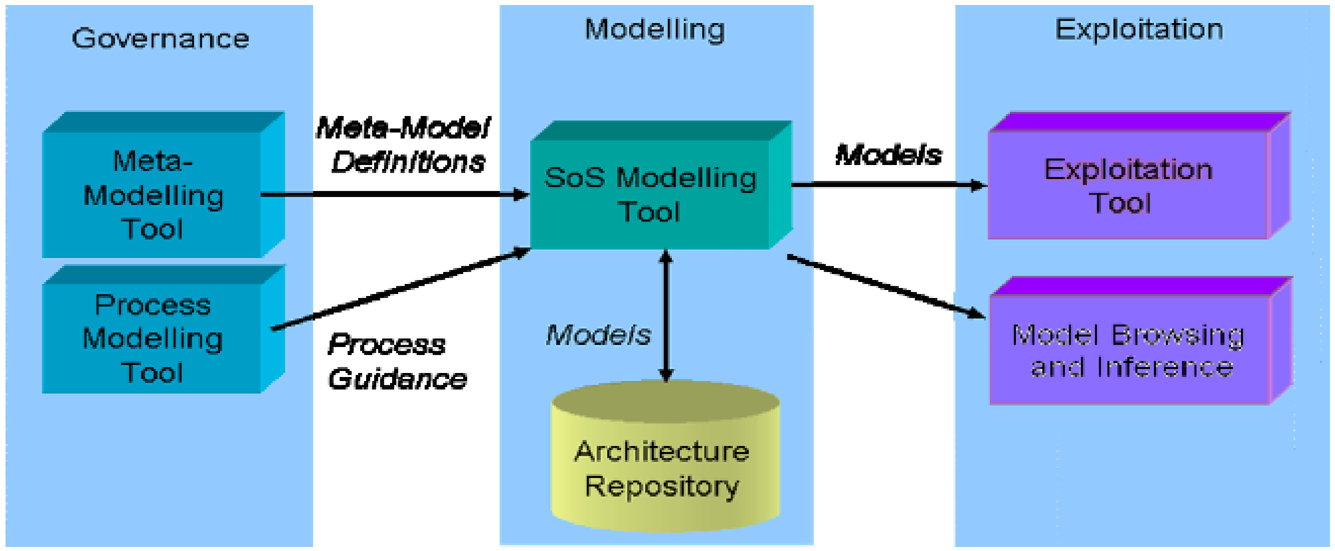
Support Slides

Requirement Classes

- **Reference Architecture Requirements**:
  - Applicable only to RA design in this activity
- **Specific Architecture Requirements**
  - Architecture of each SS-E2ES designed based on the RA
  - "Instructions manual" on how to apply the RA to each mission
- **Software Framework Requirements**
  - Requirements on framework used to support the simulator execution
  - Should be passed also to Development Teams if they wish to select another SF
- **Architecture Implementation Requirements**
  - Requirements on transition from architecture to implementation (paper to code)

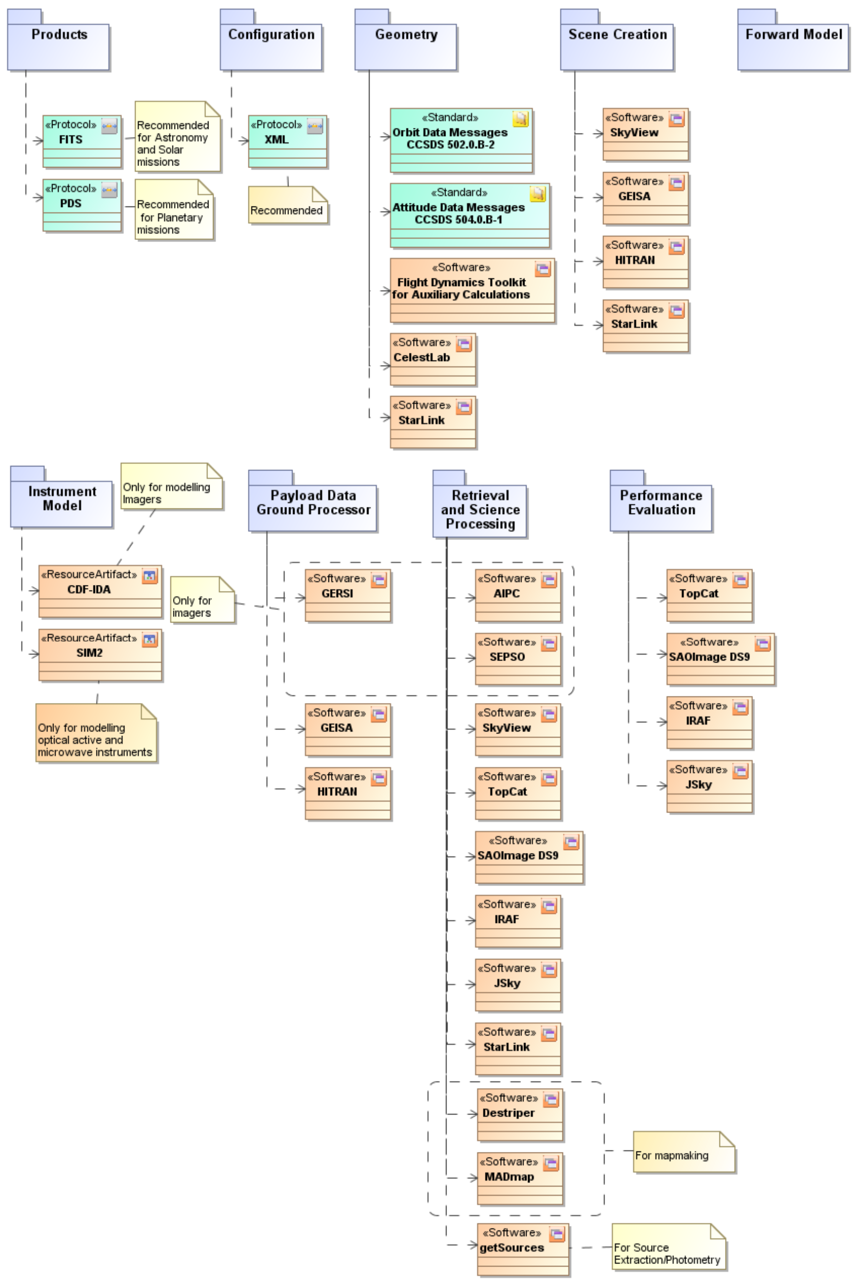
Phase 0 - Adaptation to SS-E2ES

- Combination of ESAAF and TOGAF
- Governance:
  - meta-modelling using operational, system and technology views only
  - TOGAF process
- Modelling: repository and software infrastructure (plugin and Magic Draw tool)
- Exploitation: reporting, visualization, target models

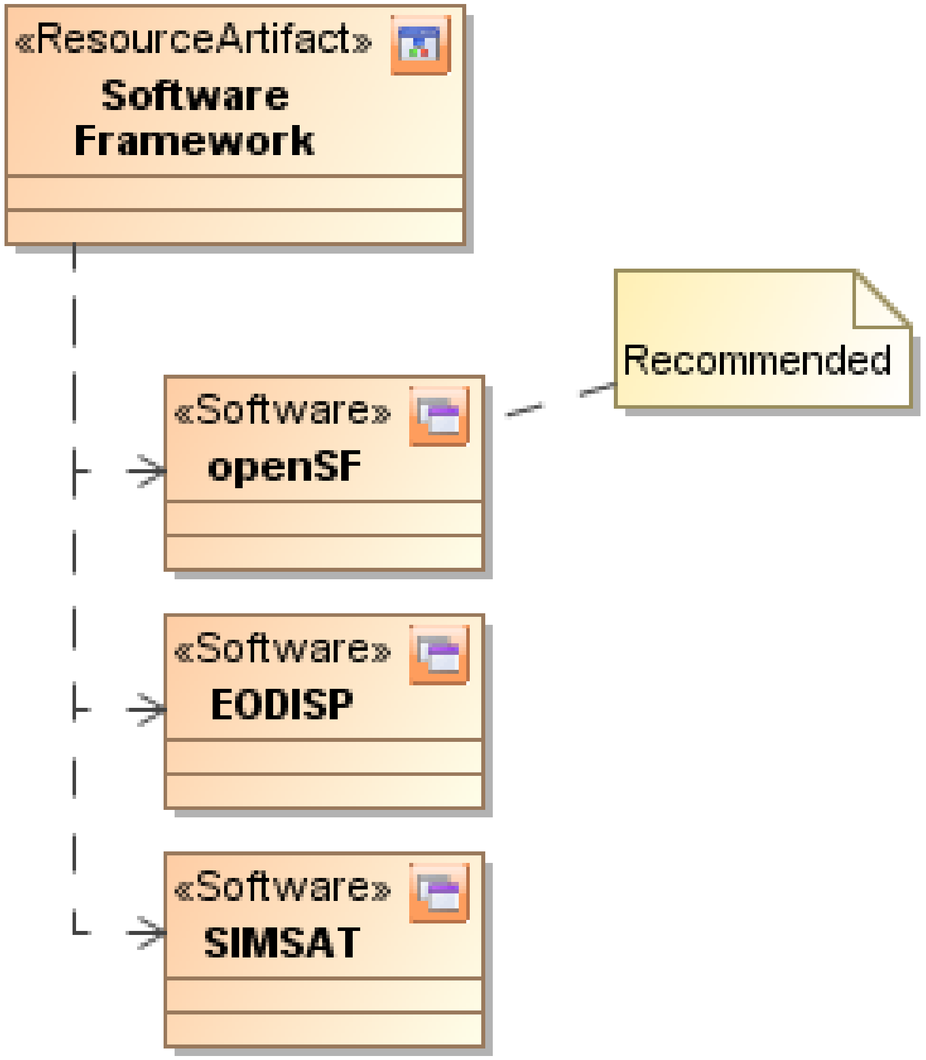


Expressed using a formal language, able to be "solved" into other formats, including source code (Java, C/C++), Matlab/Simulink and Mathematica models

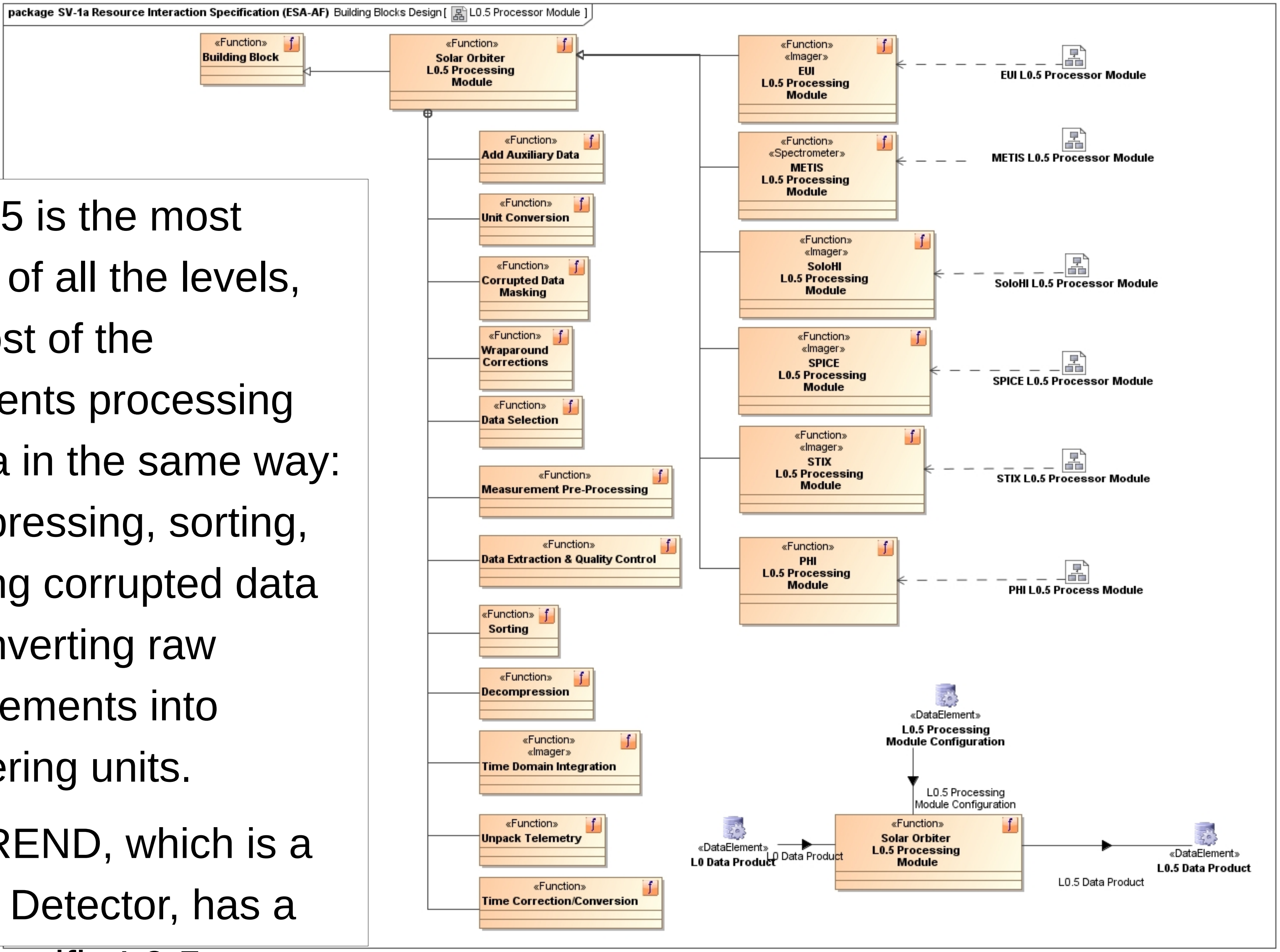
Phase 4 – Candidate Solutions



Phase 4 – Candidate Solutions



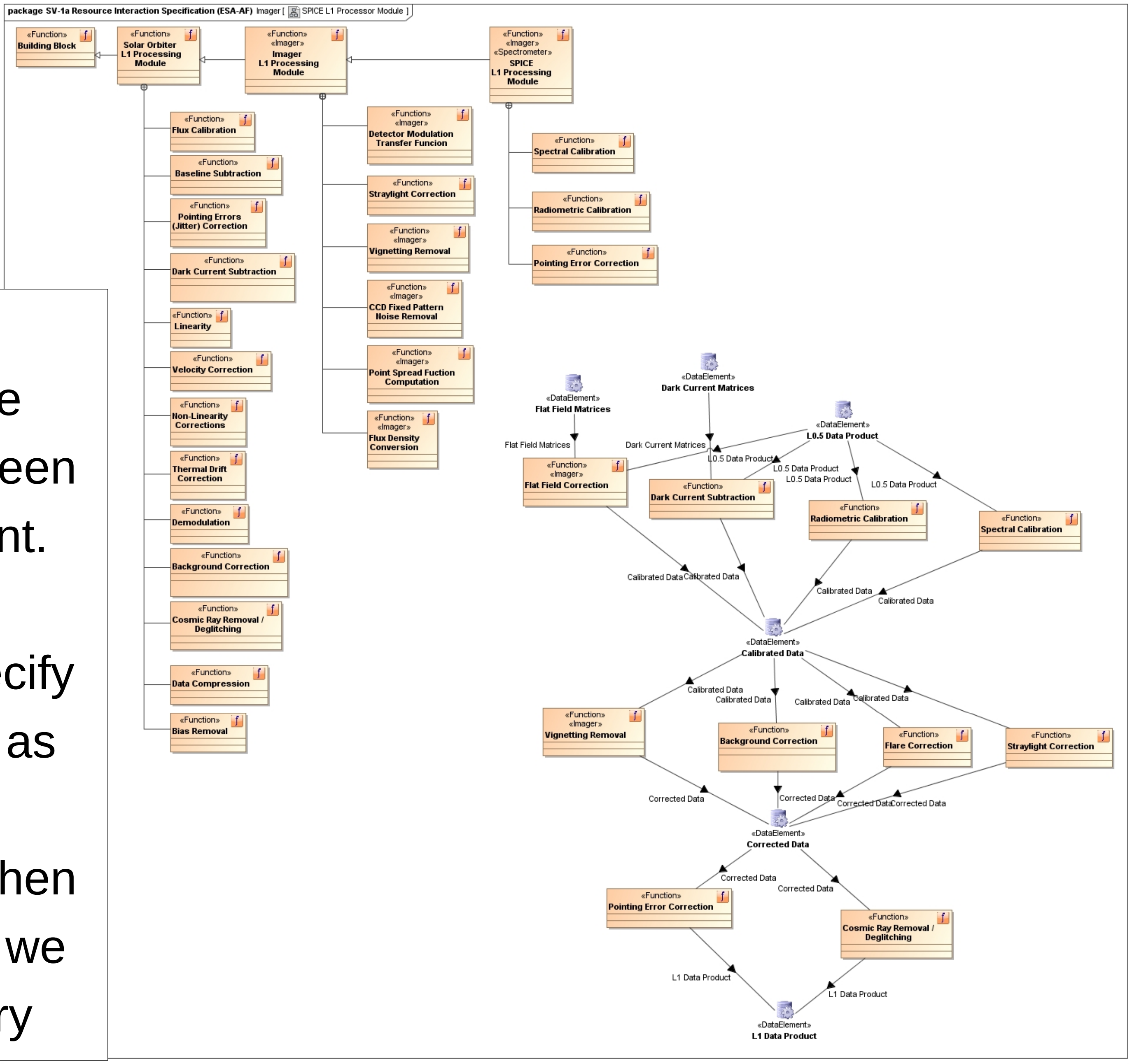
L0.5 Processing Module


package SV-1a Resource Interaction Specification (ESA-AF) Building Blocks Design [L0.5 Processor Module]
«Function» Building Block
«Function» Solar Orbiter L0.5 Processing Module
«Function» «Imager» EUI L0.5 Processing Module
EUI L0.5 Processor Module
«Function» «Spectrometer» METIS L0.5 Processing Module
METIS L0.5 Processor Module
«Function» «Imager» SoloHI L0.5 Processing Module
SoloHI L0.5 Processor Module
«Function» «Imager» SPICE L0.5 Processing Module
SPICE L0.5 Processor Module
«Function» «Imager» STIX L0.5 Processing Module
STIX L0.5 Processor Module
«Function» PHI L0.5 Processing Module
PHI L0.5 Process Module
«Function» Add Auxiliary Data
«Function» Unit Conversion
«Function» Corrupted Data Masking
«Function» Wraparound Corrections
«Function» Data Selection
«Function» Measurement Pre-Processing
«Function» Data Extraction & Quality Control
«Function» Sorting
«Function» Decompression
«Function» «Imager» Time Domain Integration
«Function» Unpack Telemetry
«Function» Time Correction/Conversion
«DataElement» L0.5 Processing Module Configuration
L0.5 Processing Module Configuration
«DataElement» L0 Data Product
L0 Data Product
«Function» Solar Orbiter L0.5 Processing Module
L0.5 Data Product
«DataElement» L0.5 Data Product

The L0.5 is the most generic of all the levels, with most of the instruments processing the data in the same way: decompressing, sorting, removing corrupted data and converting raw measurements into engineering units.

Only FREND, which is a Particle Detector, has a more specific L0.5.

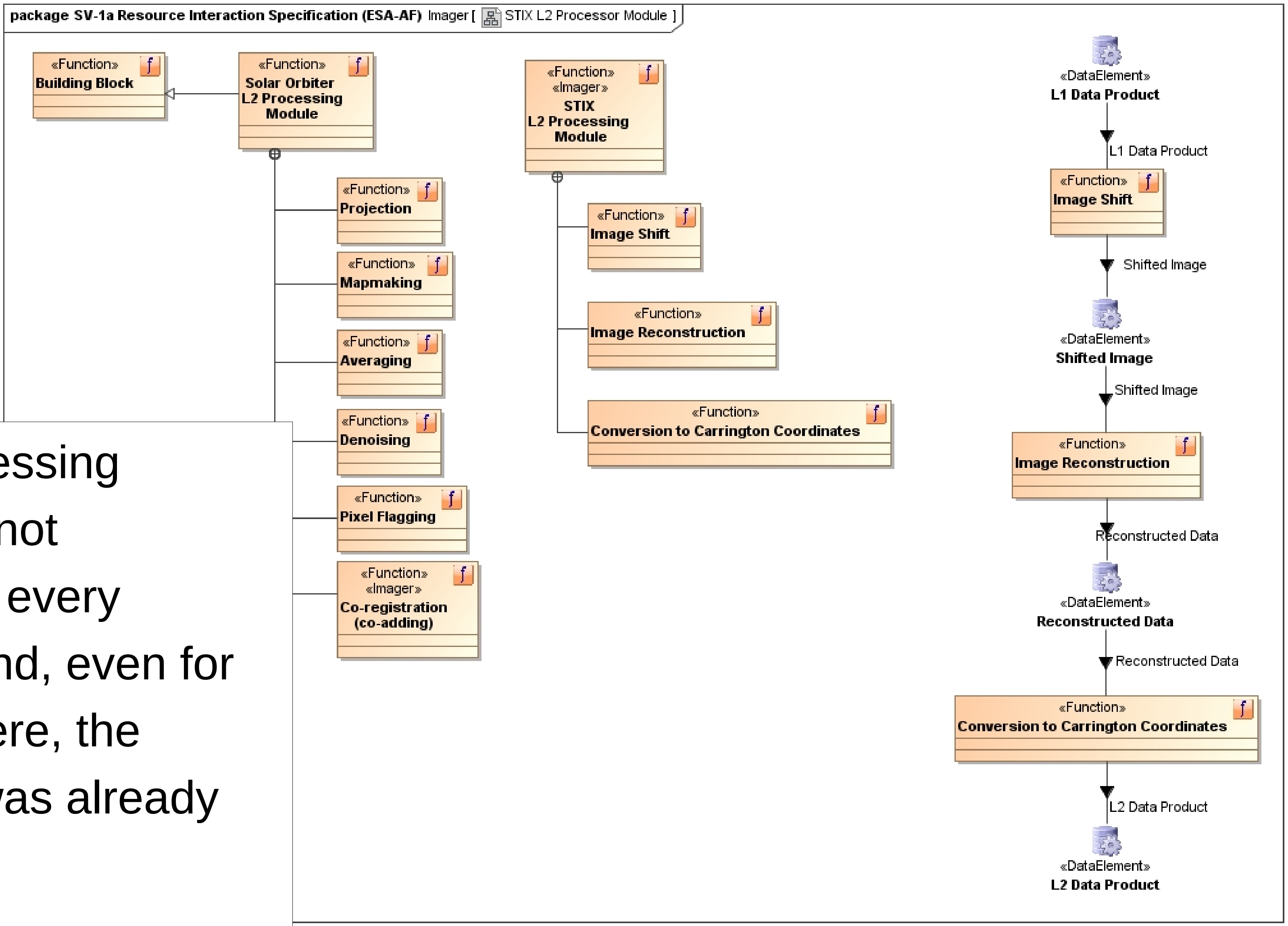
L1 Processing Module



package SV-1a Resource Interaction Specification (ESA-AF) Imager [SPICE L1 Processor Module]
«Function» Building Block
«Function» Solar Orbiter L1 Processing Module
«Function» «Imager» Imager L1 Processing Module
«Function» «Imager» «Spectrometer» SPICE L1 Processing Module
«Function» Flux Calibration
«Function» Baseline Subtraction
«Function» Pointing Errors (Jitter) Correction
«Function» Dark Current Subtraction
«Function» Linearity
«Function» Velocity Correction
«Function» Non-Linearity Corrections
«Function» Thermal Drift Correction
«Function» Demodulation
«Function» Background Correction
«Function» Cosmic Ray Removal / Deglitching
«Function» Data Compression
«Function» Bias Removal
«Function» «Imager» Detector Modulation Transfer Funcion
«Function» Straylight Correction
«Function» «Imager» Vignetting Removal
«Function» «Imager» CCD Fixed Pattern Noise Removal
«Function» «Imager» Point Spread Fuction Computation
«Function» «Imager» Flux Density Conversion
«Function» Spectral Calibration
«Function» Radiometric Calibration
«Function» Pointing Error Correction
«DataElement» Flat Field Matrices
«DataElement» Dark Current Matrices
«DataElement» L0.5 Data Product
Flat Field Matrices
Dark Current Matrices
L0.5 Data Product
L0.5 Data Product
L0.5 Data Product
L0.5 Data Product
«Function» «Imager» Flat Field Correction
«Function» Dark Current Subtraction
«Function» Radiometric Calibration
«Function» Spectral Calibration
Calibrated Data
Calibrated Data
Calibrated Data
Calibrated Data
«DataElement» Calibrated Data
Calibrated Data
Calibrated Data
Calibrated Data
Calibrated Data
«Function» «Imager» Vignetting Removal
«Function» Background Correction
«Function» Flare Correction
«Function» Straylight Correction
Corrected Data
Corrected Data
Corrected Data
Corrected Data
«DataElement» Corrected Data
Corrected Data
Corrected Data
«Function» Pointing Error Correction
«Function» Cosmic Ray Removal / Deglitching
L1 Data Product
L1 Data Product
«DataElement» L1 Data Product

The L1 Processing Module is where all the main differences between instruments are present. We modelled many in detail, and tried to specify the pipelines by using as many generic Building Blocks as possible. When that was not possible, we specified the necessary specific Building Blocks.

L2 Processing Module


package SV-1a Resource Interaction Specification (ESA-AF) Imager [STIX L2 Processor Module]
«Function» Building Block
«Function» Solar Orbiter L2 Processing Module
«Function» Projection
«Function» Mapmaking
«Function» Averaging
«Function» Denoising
«Function» Pixel Flagging
«Function» «Imager» Co-registration (co-adding)
«Function» «Imager» STIX L2 Processing Module
«Function» Image Shift
«Function» Image Reconstruction
«Function» Conversion to Carrington Coordinates
«DataElement» L1 Data Product
L1 Data Product
«Function» Image Shift
Shifted Image
«DataElement» Shifted Image
Shifted Image
«Function» Image Reconstruction
Reconstructed Data
«DataElement» Reconstructed Data
Reconstructed Data
«Function» Conversion to Carrington Coordinates
L2 Data Product
«DataElement» L2 Data Product

The L2 Processing Module was not applicable to every instrument and, even for those that were, the processing was already too specific.

For Imagers, the Mapmaking modules were the most generic.

Building Block Definitions

Model Sub-Structure

- Inside each of the main sections, Building Blocks are further subdivided
  - From **Spacecraft Geometry** Module to the **Scene Creation** Module, no sub-division is necessary
  - **Forward Models** are now organized by **Mission Type**:
    - Astronomy, Planetary, Solar and Exoplanet
  - The **Instrument Models** are organized by **Detector Type**:
    - Micro-Channel Plate, Bolometer, APS, Heterodyne Mixer, CCD, Semiconductor Array, Photodiode, Antenna, Scintillator
  - For **Processing Modules**, the Building Blocks are organized by **Detector Type**:
    - Altimeter, Polarimeter, Particle Detector, Imager, Spectrometer, Sounder and Radiometer

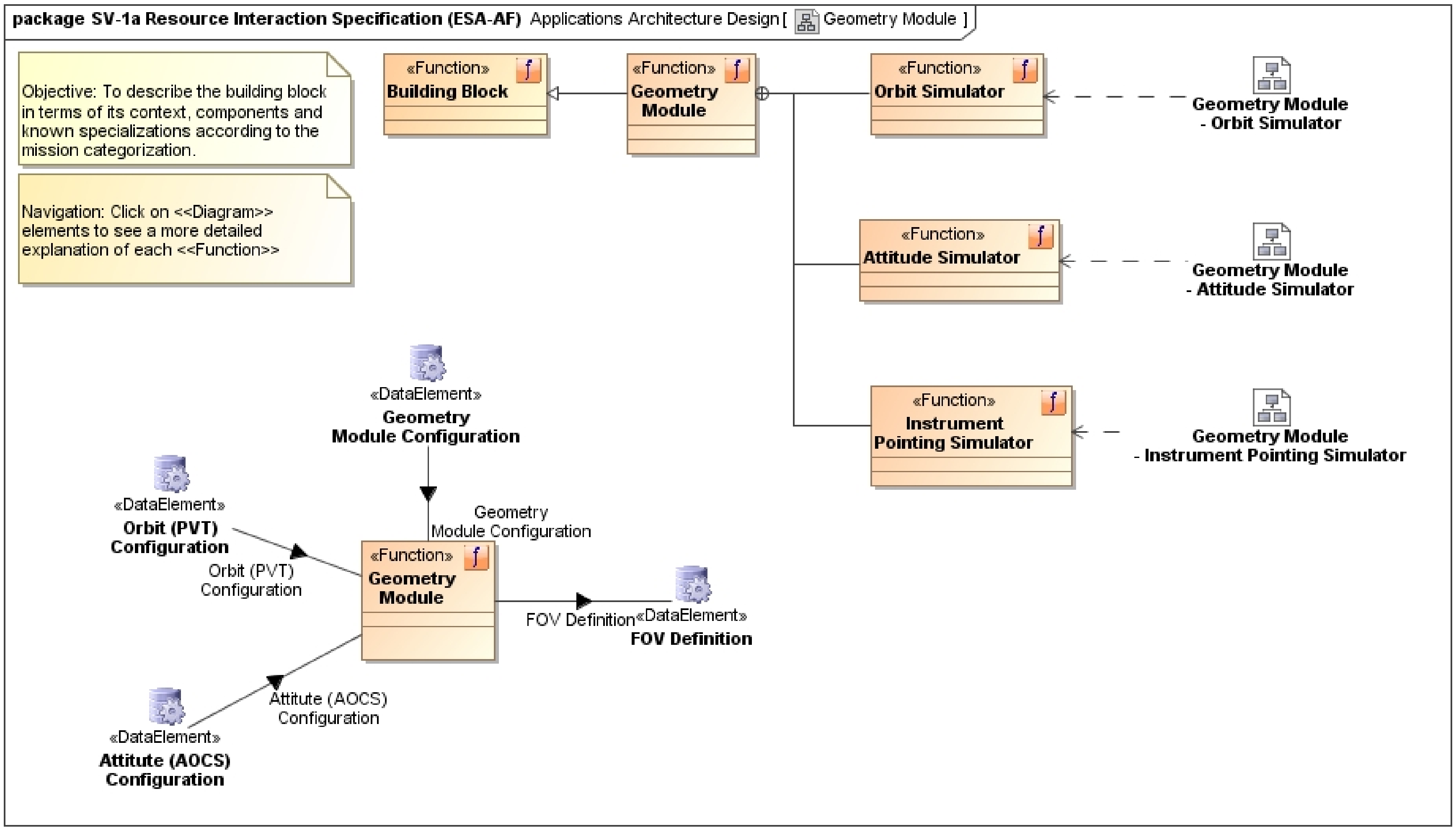
Building Block Description

- Every Building Block has the same description sections
  - **Functionality**, with a description of the Building Block algorithms
  - **Interfaces**, with a description of the main Building Block inputs and outputs
  - **Building Block**, with a standard diagram describing the Building Block contents and interfaces

Geometry Module

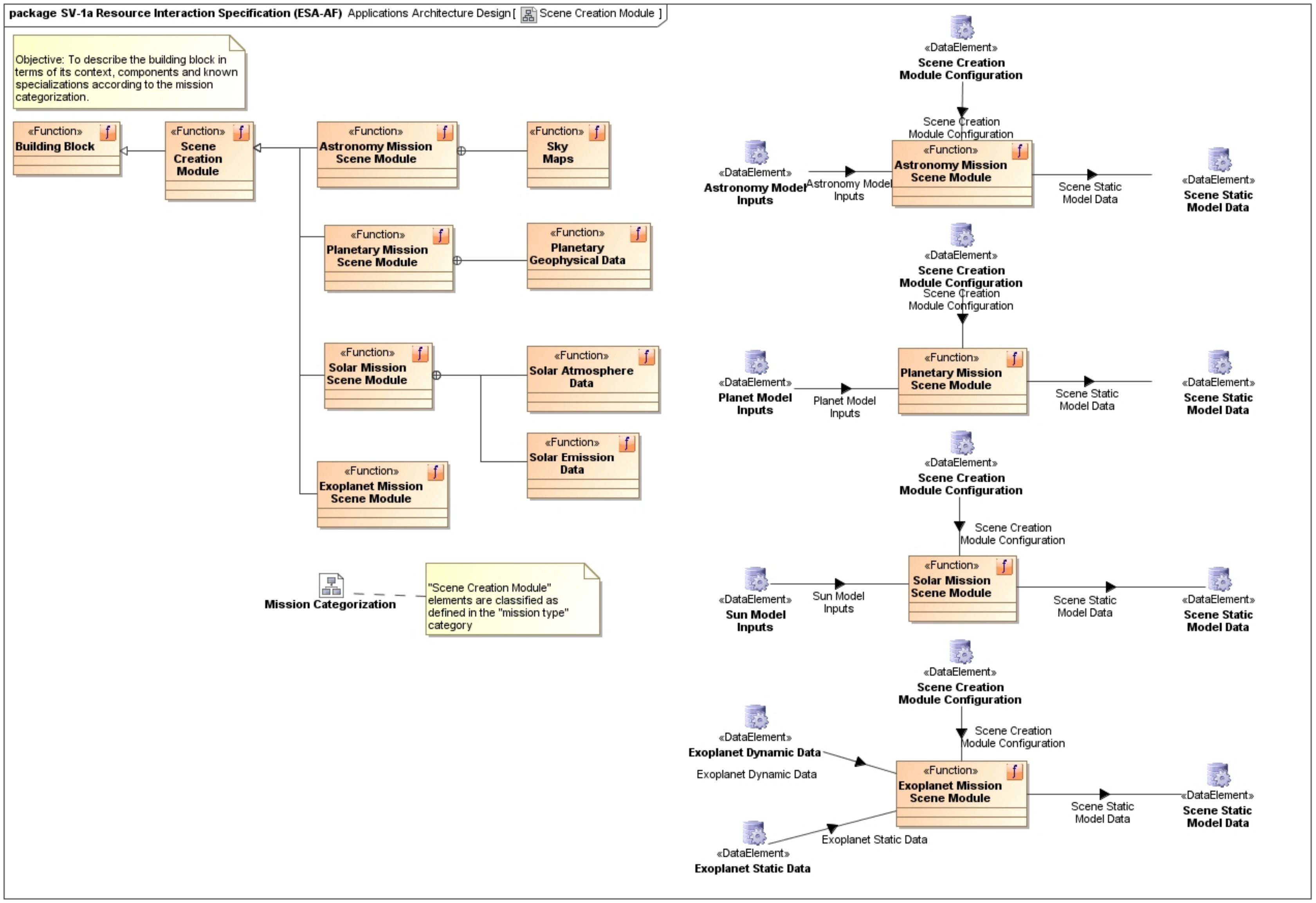
Spacecraft Geometry Module is very generic and applicable to every mission.

Included BBs and sub-diagrams for **Orbit Simulator**, **Attitude Simulator** and **Instrument Pointing Simulator**


package SV-1a Resource Interaction Specification (ESA-AF) Applications Architecture Design [Geometry Module]
Objective: To describe the building block in terms of its context, components and known specializations according to the mission categorization.
Navigation: Click on <<Diagram>> elements to see a more detailed explanation of each <<Function>>
«Function» Building Block
«Function» Geometry Module
«Function» Orbit Simulator
«Function» Attitude Simulator
«Function» Instrument Pointing Simulator
Geometry Module - Orbit Simulator
Geometry Module - Attitude Simulator
Geometry Module - Instrument Pointing Simulator
«DataElement» Geometry Module Configuration
Geometry Module Configuration
«DataElement» Orbit (PVT) Configuration
Orbit (PVT) Configuration
«Function» Geometry Module
FOV Definition
«DataElement» FOV Definition
Attitute (AOCS) Configuration
«DataElement» Attitute (AOCS) Configuration

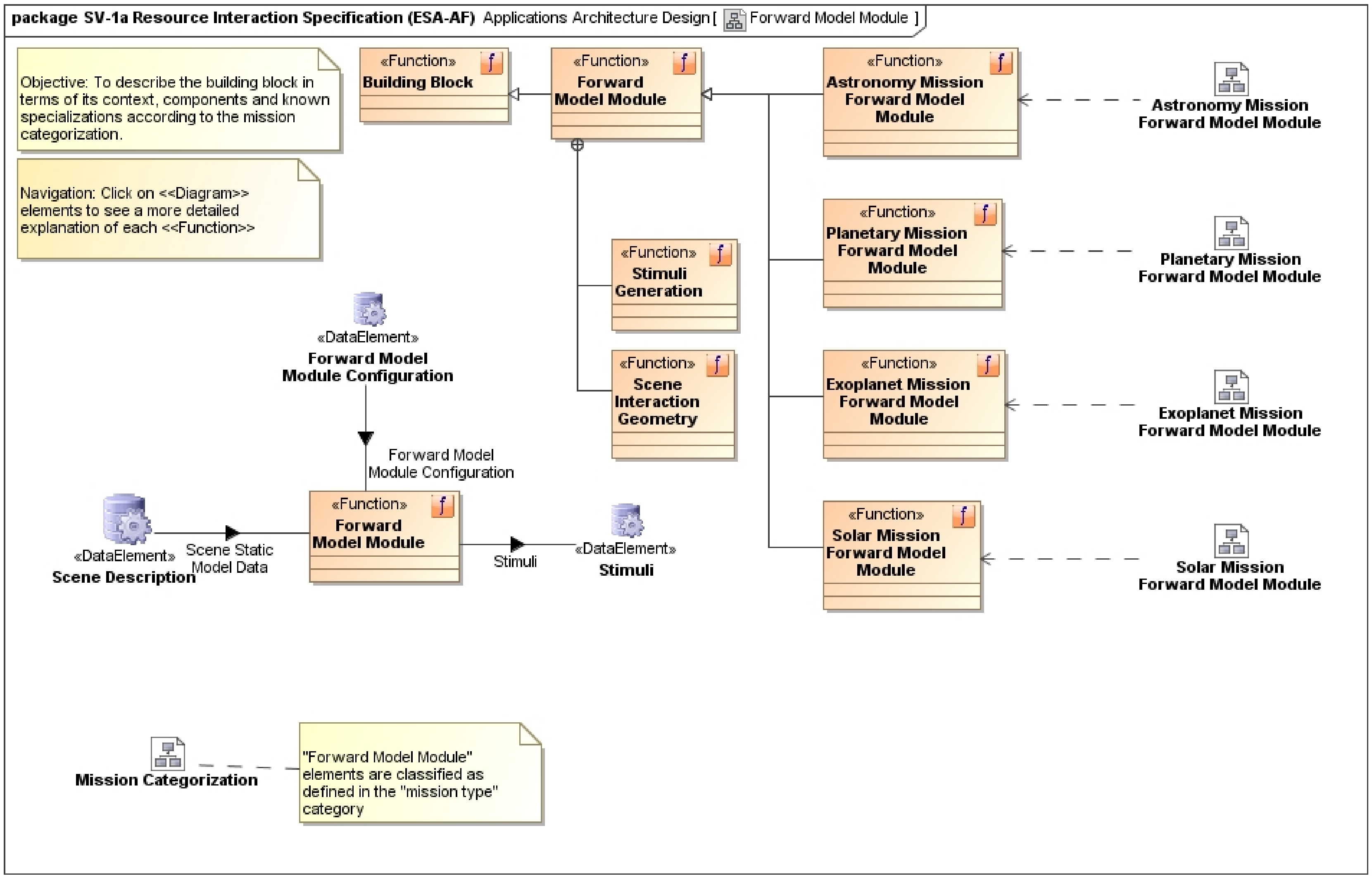
Scene Creation Module

Scene Creation Module section is done for 4 types of Missions


package SV-1a Resource Interaction Specification (ESA-AF) Applications Architecture Design [Scene Creation Module]
Objective: To describe the building block in terms of its context, components and known specializations according to the mission categorization.
«Function» Building Block
«Function» Scene Creation Module
«Function» Astronomy Mission Scene Module
«Function» Sky Maps
«Function» Planetary Mission Scene Module
«Function» Planetary Geophysical Data
«Function» Solar Mission Scene Module
«Function» Solar Atmosphere Data
«Function» Solar Emission Data
«Function» Exoplanet Mission Scene Module
Mission Categorization
"Scene Creation Module" elements are classified as defined in the "mission type" category
«DataElement» Scene Creation Module Configuration
Scene Creation Module Configuration
«DataElement» Astronomy Model Inputs
Astronomy Model Inputs
«Function» Astronomy Mission Scene Module
Scene Static Model Data
«DataElement» Scene Static Model Data
«DataElement» Scene Creation Module Configuration
Scene Creation Module Configuration
«DataElement» Planet Model Inputs
Planet Model Inputs
«Function» Planetary Mission Scene Module
Scene Static Model Data
«DataElement» Scene Static Model Data
«DataElement» Scene Creation Module Configuration
Scene Creation Module Configuration
«DataElement» Sun Model Inputs
Sun Model Inputs
«Function» Solar Mission Scene Module
Scene Static Model Data
«DataElement» Scene Static Model Data
«DataElement» Scene Creation Module Configuration
Scene Creation Module Configuration
«DataElement» Exoplanet Dynamic Data
Exoplanet Dynamic Data
«Function» Exoplanet Mission Scene Module
Scene Static Model Data
«DataElement» Scene Static Model Data
Exoplanet Static Data
«DataElement» Exoplanet Static Data

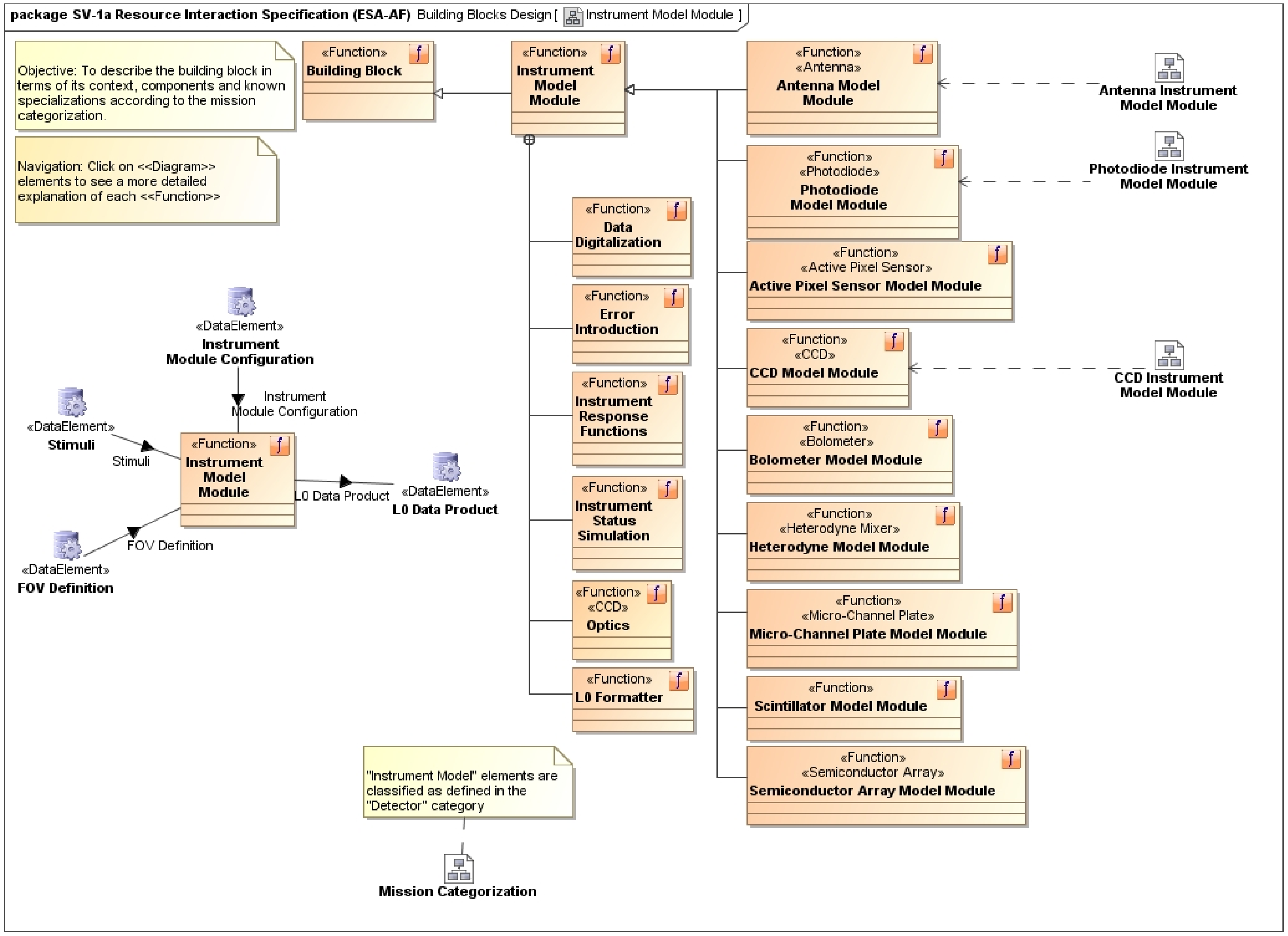
Forward Model

Forward Model Module is also completed for the same 4 Mission Types (with Mission Specific BBs and sub-diagrams).


package SV-1a Resource Interaction Specification (ESA-AF) Applications Architecture Design [Forward Model Module]
Objective: To describe the building block in terms of its context, components and known specializations according to the mission categorization.
Navigation: Click on <<Diagram>> elements to see a more detailed explanation of each <<Function>>
«Function» Building Block
«Function» Forward Model Module
«Function» Astronomy Mission Forward Model Module
Astronomy Mission Forward Model Module
«Function» Planetary Mission Forward Model Module
Planetary Mission Forward Model Module
«Function» Exoplanet Mission Forward Model Module
Exoplanet Mission Forward Model Module
«Function» Solar Mission Forward Model Module
Solar Mission Forward Model Module
«Function» Stimuli Generation
«Function» Scene Interaction Geometry
«DataElement» Forward Model Module Configuration
Forward Model Module Configuration
«DataElement» Scene Description
Scene Static Model Data
«Function» Forward Model Module
Stimuli
«DataElement» Stimuli
Mission Categorization
"Forward Model Module" elements are classified as defined in the "mission type" category

Instrument Model

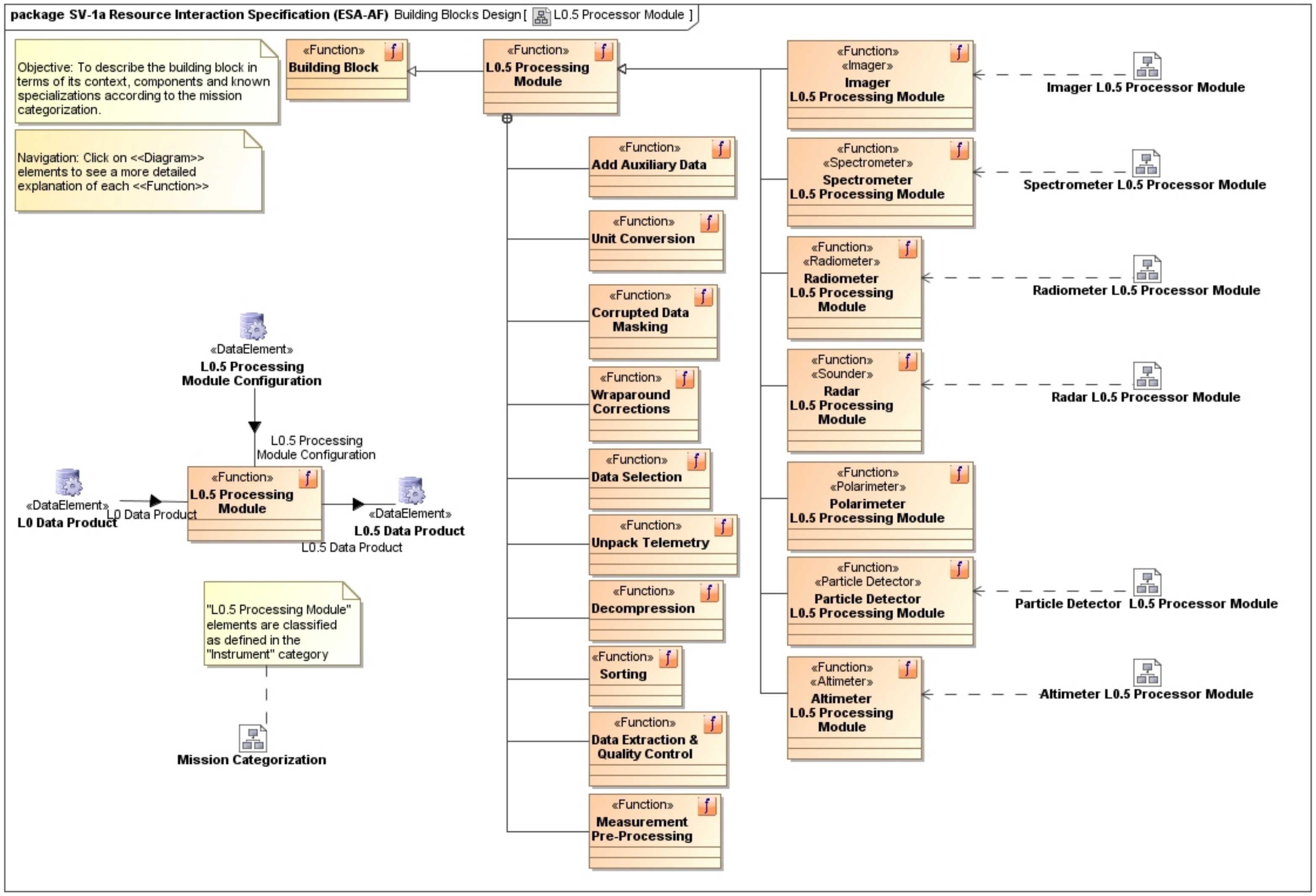
Instrument Model with 3 Sensor Types fleshed out. The sensor simulation can be very specific and we did not attempt to go further.



package SV-1a Resource Interaction Specification (ESA-AF) Building Blocks Design [Instrument Model Module]
Objective: To describe the building block in terms of its context, components and known specializations according to the mission categorization.
Navigation: Click on <<Diagram>> elements to see a more detailed explanation of each <<Function>>
«Function» Building Block
«Function» Instrument Model Module
«Function» «Antenna» Antenna Model Module
Antenna Instrument Model Module
«Function» «Photodiode» Photodiode Model Module
Photodiode Instrument Model Module
«Function» «Active Pixel Sensor» Active Pixel Sensor Model Module
«Function» «CCD» CCD Model Module
CCD Instrument Model Module
«Function» «Bolometer» Bolometer Model Module
«Function» «Heterodyne Mixer» Heterodyne Model Module
«Function» «Micro-Channel Plate» Micro-Channel Plate Model Module
«Function» Scintillator Model Module
«Function» «Semiconductor Array» Semiconductor Array Model Module
«Function» Data Digitalization
«Function» Error Introduction
«Function» Instrument Response Functions
«Function» Instrument Status Simulation
«Function» «CCD» Optics
«Function» L0 Formatter
«DataElement» Instrument Module Configuration
Instrument Module Configuration
«DataElement» Stimuli
Stimuli
«Function» Instrument Model Module
L0 Data Product
«DataElement» L0 Data Product
FOV Definition
«DataElement» FOV Definition
"Instrument Model" elements are classified as defined in the "Detector" category
Mission Categorization

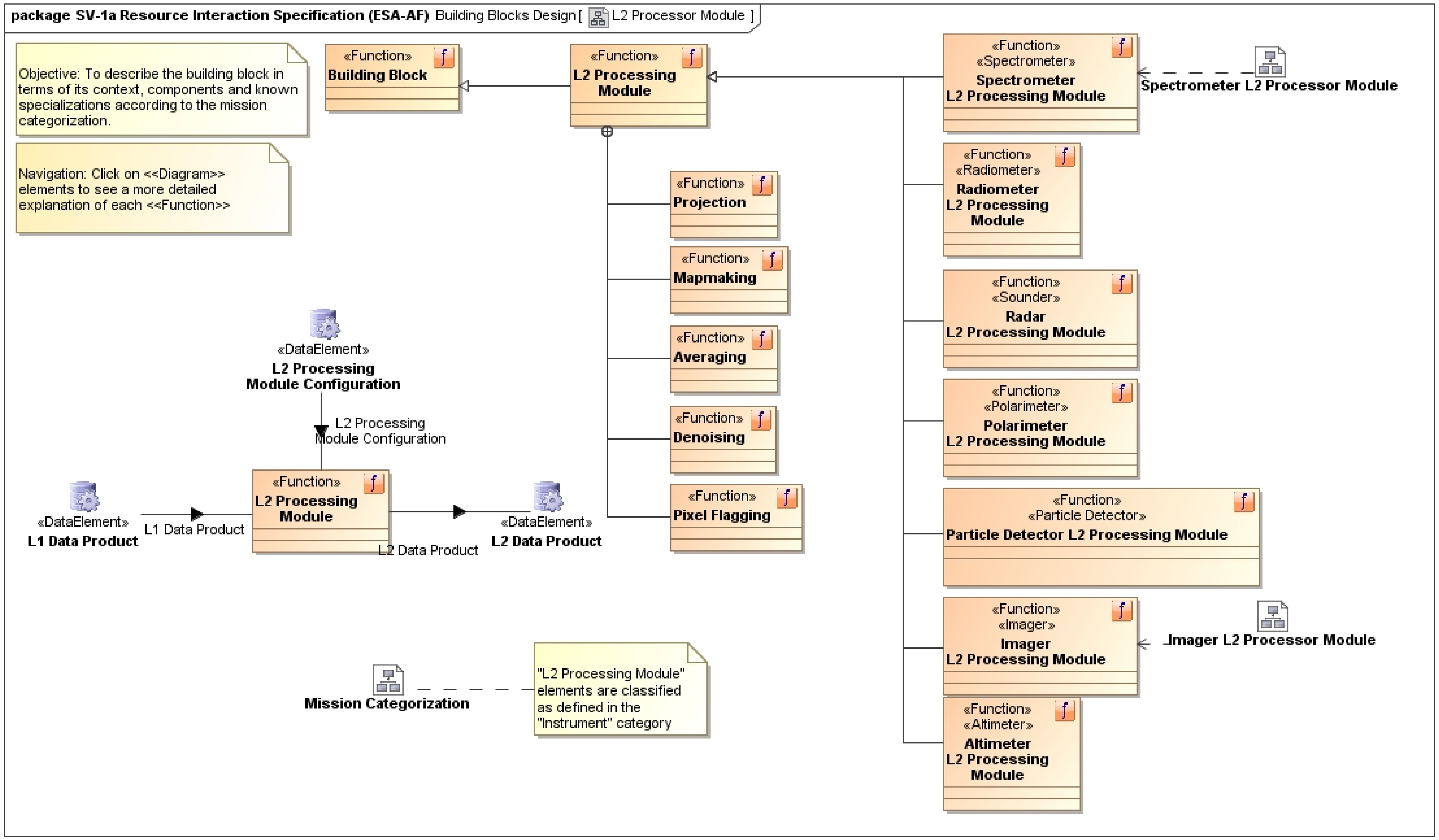
L0 to L0.5 Processing Model

L0 to L0.5 Data Processing Model has been defined for all Instrument Classes except Polarimeters. It is the most generic Processing


package SV-1a Resource Interaction Specification (ESA-AF) Building Blocks Design [L0.5 Processor Module]
Objective: To describe the building block in terms of its context, components and known specializations according to the mission categorization.
Navigation: Click on <<Diagram>> elements to see a more detailed explanation of each <<Function>>
«Function» Building Block
«Function» L0.5 Processing Module
«Function» Add Auxiliary Data
«Function» Unit Conversion
«Function» Corrupted Data Masking
«Function» Wraparound Corrections
«Function» Data Selection
«Function» Unpack Telemetry
«Function» Decompression
«Function» Sorting
«Function» Data Extraction & Quality Control
«Function» Measurement Pre-Processing
«Function» «Imager» Imager L0.5 Processing Module
«Function» «Spectrometer» Spectrometer L0.5 Processing Module
«Function» «Radiometer» Radiometer L0.5 Processing Module
«Function» «Sounder» Radar L0.5 Processing Module
«Function» «Polarimeter» Polarimeter L0.5 Processing Module
«Function» «Particle Detector» Particle Detector L0.5 Processing Module
«Function» «Altimeter» Altimeter L0.5 Processing Module
Imager L0.5 Processor Module
Spectrometer L0.5 Processor Module
Radiometer L0.5 Processor Module
Radar L0.5 Processor Module
Particle Detector L0.5 Processor Module
Altimeter L0.5 Processor Module
«DataElement» L0.5 Processing Module Configuration
L0.5 Processing Module Configuration
«DataElement» L0 Data Product
L0 Data Product
«Function» L0.5 Processing Module
«DataElement» L0.5 Data Product
L0.5 Data Product
"L0.5 Processing Module" elements are classified as defined in the "Instrument" category
Mission Categorization

L1 to L2 Processing Model

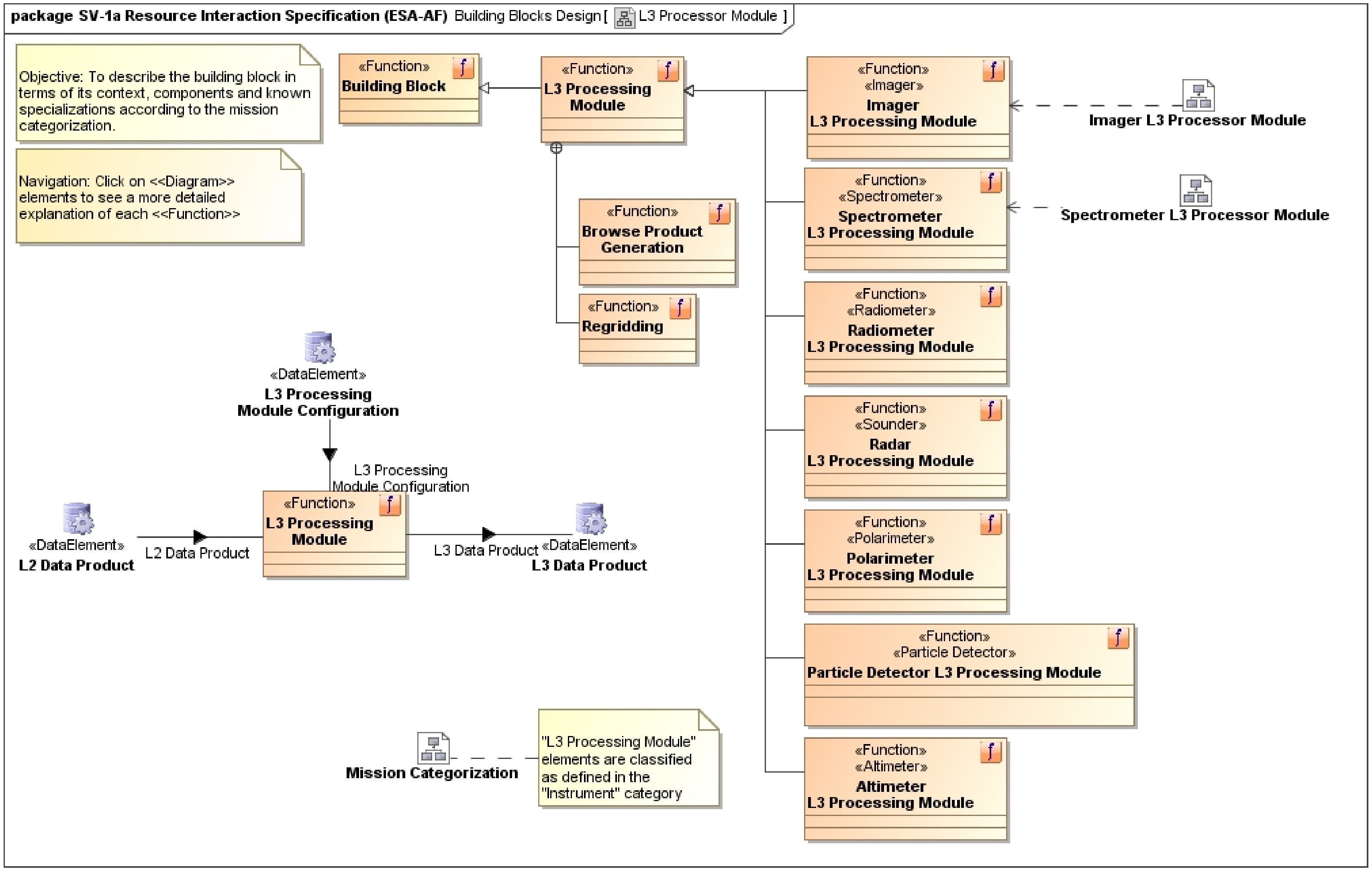
L1 to L2 Data Processing Model has been filled out for the classes that have the most generic functions. Not all classes were filled.



package SV-1a Resource Interaction Specification (ESA-AF) Building Blocks Design [L2 Processor Module]
Objective: To describe the building block in terms of its context, components and known specializations according to the mission categorization.
Navigation: Click on <<Diagram>> elements to see a more detailed explanation of each <<Function>>
«Function» Building Block
«Function» L2 Processing Module
«Function» «Spectrometer» Spectrometer L2 Processing Module
Spectrometer L2 Processor Module
«Function» «Radiometer» Radiometer L2 Processing Module
«Function» «Sounder» Radar L2 Processing Module
«Function» «Polarimeter» Polarimeter L2 Processing Module
«Function» «Particle Detector» Particle Detector L2 Processing Module
«Function» «Imager» Imager L2 Processing Module
Imager L2 Processor Module
«Function» «Altimeter» Altimeter L2 Processing Module
«Function» Projection
«Function» Mapmaking
«Function» Averaging
«Function» Denoising
«Function» Pixel Flagging
«DataElement» L2 Processing Module Configuration
L2 Processing Module Configuration
«DataElement» L1 Data Product
L1 Data Product
«Function» L2 Processing Module
L2 Data Product
«DataElement» L2 Data Product
Mission Categorization
"L2 Processing Module" elements are classified as defined in the "Instrument" category

Status: L2 to L3 Processing Model

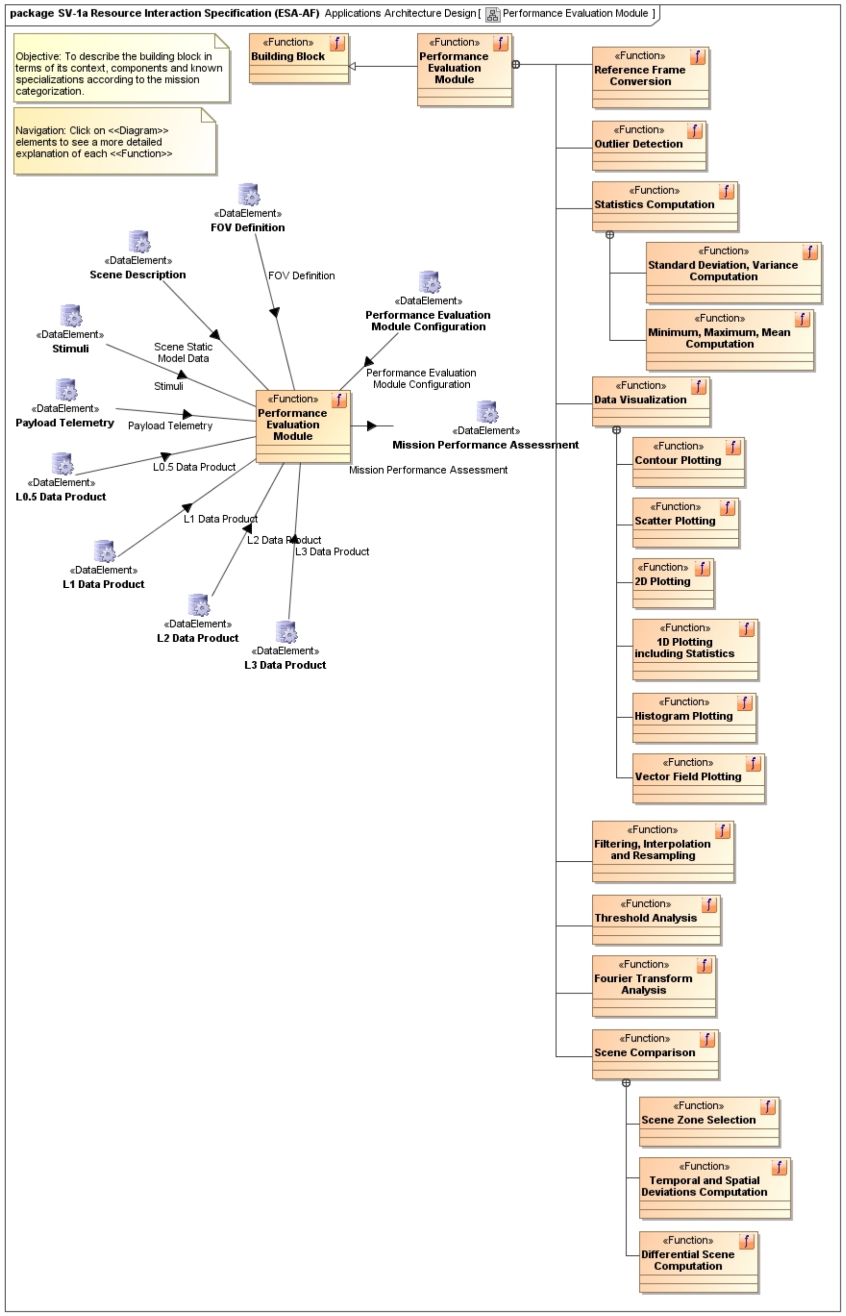
L2 to L3 Data Processing Model has been filled out for the most representative classes. Not all classes were filled. No application in selected n



package SV-1a Resource Interaction Specification (ESA-AF) Building Blocks Design [L3 Processor Module]
Objective: To describe the building block in terms of its context, components and known specializations according to the mission categorization.
Navigation: Click on <<Diagram>> elements to see a more detailed explanation of each <<Function>>
«Function» Building Block
«Function» L3 Processing Module
«Function» Browse Product Generation
«Function» Regridding
«Function» «Imager» Imager L3 Processing Module
Imager L3 Processor Module
«Function» «Spectrometer» Spectrometer L3 Processing Module
Spectrometer L3 Processor Module
«Function» «Radiometer» Radiometer L3 Processing Module
«Function» «Sounder» Radar L3 Processing Module
«Function» «Polarimeter» Polarimeter L3 Processing Module
«Function» «Particle Detector» Particle Detector L3 Processing Module
«Function» «Altimeter» Altimeter L3 Processing Module
«DataElement» L3 Processing Module Configuration
L3 Processing Module Configuration
«DataElement» L2 Data Product
L2 Data Product
«Function» L3 Processing Module
L3 Data Product
«DataElement» L3 Data Product
Mission Categorization
"L3 Processing Module" elements are classified as defined in the "Instrument" category

Status: Performance Evaluation Module

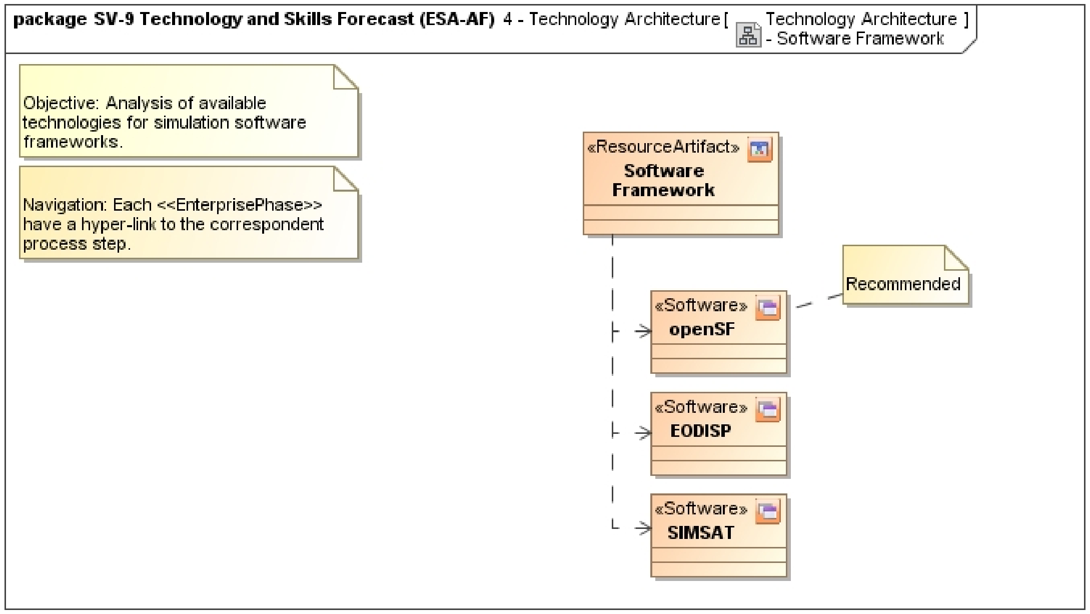
The **Performance Evaluation Module** has been filled out with several Building Blocks. Also here it is not possible to recommend a series of processing steps as these will vary depending on the Evaluation tasks performed.



package SV-1a Resource Interaction Specification (ESA-AF) Applications Architecture Design [Performance Evaluation Module]
Objective: To describe the building block in terms of its context, components and known specializations according to the mission categorization.
Navigation: Click on <<Diagram>> elements to see a more detailed explanation of each <<Function>>
«Function» Building Block
«Function» Performance Evaluation Module
«Function» Reference Frame Conversion
«Function» Outlier Detection
«Function» Statistics Computation
«Function» Standard Deviation, Variance Computation
«Function» Minimum, Maximum, Mean Computation
«Function» Data Visualization
«Function» Contour Plotting
«Function» Scatter Plotting
«Function» 2D Plotting
«Function» 1D Plotting including Statistics
«Function» Histogram Plotting
«Function» Vector Field Plotting
«Function» Filtering, Interpolation and Resampling
«Function» Threshold Analysis
«Function» Fourier Transform Analysis
«Function» Scene Comparison
«Function» Scene Zone Selection
«Function» Temporal and Spatial Deviations Computation
«Function» Differential Scene Computation
«DataElement» FOV Definition
«DataElement» Scene Description
«DataElement» Stimuli
«DataElement» Payload Telemetry
«DataElement» L0.5 Data Product
«DataElement» L1 Data Product
«DataElement» L2 Data Product
«DataElement» L3 Data Product
«DataElement» Performance Evaluation Module Configuration
«DataElement» Mission Performance Assessment
FOV Definition
Scene Static Model Data
Stimuli
Payload Telemetry
L0.5 Data Product
L1 Data Product
L2 Data Product
L3 Data Product
Performance Evaluation Module Configuration
Mission Performance Assessment

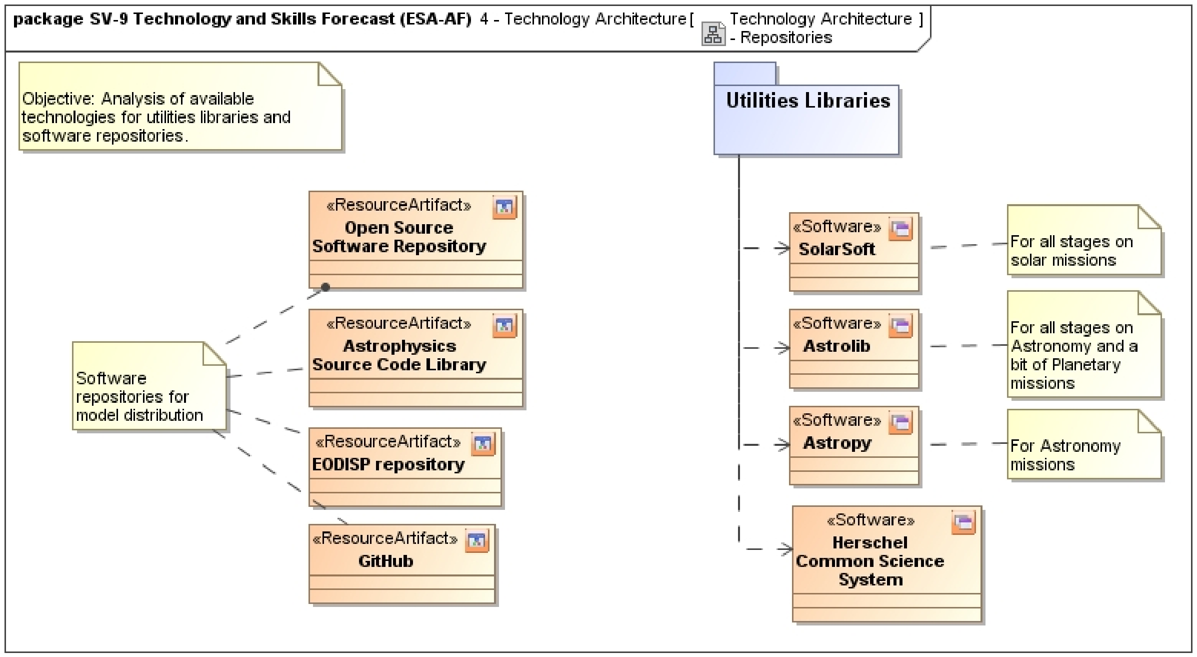
Status: Utilities Module

Beyond the Processing Libraries survey (see Slide 17), the **Utilities Module** has been filled out with the Software Framework and Repositories surveys.



package SV-9 Technology and Skills Forecast (ESA-AF) 4 - Technology Architecture [Technology Architecture - Software Framework]
Objective: Analysis of available technologies for simulation software frameworks.
Navigation: Each <<EnterprisePhase>> have a hyper-link to the correspondent process step.
«ResourceArtifact» Software Framework
«Software» openSF
«Software» EODISP
«Software» SIMSAT
Recommended

Software Frameworks



package SV-9 Technology and Skills Forecast (ESA-AF) 4 - Technology Architecture [Technology Architecture - Repositories]
Objective: Analysis of available technologies for utilities libraries and software repositories.
«ResourceArtifact» Open Source Software Repository
«ResourceArtifact» Astrophysics Source Code Library
«ResourceArtifact» EODISP repository
«ResourceArtifact» GitHub
Software repositories for model distribution
Utilities Libraries
«Software» SolarSoft
«Software» Astrolib
«Software» Astropy
«Software» Herschel Common Science System
For all stages on solar missions
For all stages on Astronomy and a bit of Planetary missions
For Astronomy missions

Repositories

Building Block TN

- Very large phase space – a systematic attempt done, the most important classes are described
- Difficult to organize in an intuitive way – structure was revised many times but the usage of ESA-AF has eased the usage of a common language for the whole document
- Variable levels of detail in bibliography make it difficult to have an homogeneous description – a large effort was spent trying to have a standard description of the existing modules
- Description must be generic, not all inputs and outputs can be described – but still very difficult to decide where to stop detailing

Mission Analysis TN

RA Application Evaluation

- Programmatic criteria (applicable to this activity):
  - **Management and coordination**: time needed to follow the development of the SS-E2E simulator and coordination of activities
  - **Requirements definition**: time needed to define the requirements specification of the simulator
  - **Architecture and interfaces definition**: Time required to define the architecture of the simulator and its interfaces
  - **Modules defintion, development and validation**: Time needed to define, develop and validate the simulator modules
  - **Simulator integration**: Time needed to integrate the complete simulator
  - **Simulator verification and validation**: Time needed to verify and validate the simulator.
  - **Maintenance**: Time devoted to simulator maintenance

RA Application Evaluation

- Technical criteria (applicable to a specific/real SS-E2ES implementation):
  - **Modularity**: substituting one module or building block for another implementation
  - **Evolution capability** for use in later phases: evolving the simulator for later phases of the missions.
  - **Execution performance**: efficiency of the execution of the E2E simulation, etc
  - **Propagation of errors**: efficiency of detecting and isolating a failure in the simulator
  - **Parameter consistency checking**: ensuring coherence of the interfaces and minimizing out-of-range input parameters

Analyzed Missions - ExoMars

Category	ExoMars TGO
Mission Type	Planetary with 9 months transit time
Instrument Type	**NOMAD**: Semiconductor Array (IR) and CCD (UV) **ACS**: Semiconductor Array with two different substrates (HgCdTe, PbCdSe) **CaSSIS**: CCD **FREND**: Scintillator + He-3 Counter
Detector Type	**NOMAD**: 2x Imaging Spectrometer **ACS**: 2x Echelle Spectrometer + Dual Band Fourier Spectrometer **CaSSIS**: Imager **FREND**: Neutron
Waveband	**NOMAD**: IR (2.2-4.3 μm) and VIS/UV (0.2-0.65 μm) **ACS-NIR**: NIR (0.7-1.7 μm) **ACS-MIR**: MIR (2.3-4.6 μm) **ACS-TIR**: TIR and FIR (1.7-17 μm and 1.7-4 μm) **CaSSIS**: Four band filters: pan-chromatic (centred at 650 nm), blue-green (475 nm), IR (950 nm) and NIR (850 nm) **FREND**: Detector 1: ^{3}He counter for epithermal neutrons (0.4 eV-500 keV) Detector 2: stylbene scintillator crystal for fast neutrons (0.5-10 MeV)

Analyzed Missions - Euclid

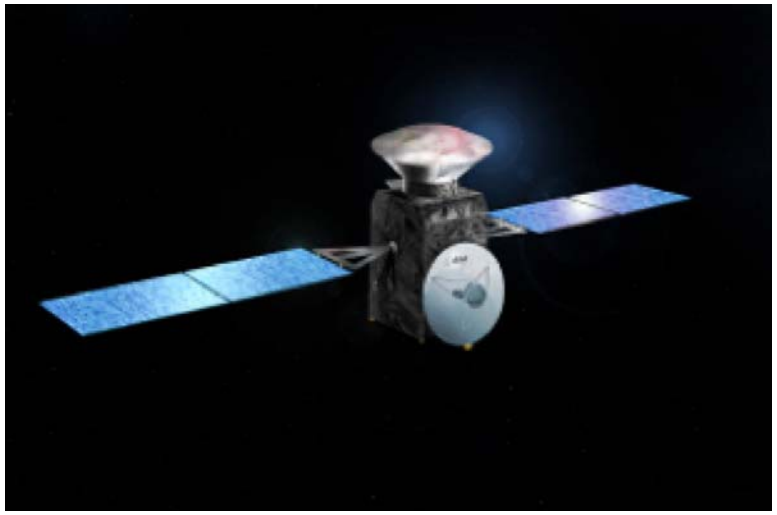
Category	Euclid
Mission Type	Astronomy
Instrument Type	**VIS**: CCD **NISP**: CCD
Detector Type	**VIS**: Imager **NISP**: Imager & Photometer
Waveband	**VIS**: VIS (0.55 µm to 0.9 µm) **NISP**: NIR 3 broad band filters (Y, J, H) on a wheel, covering the band from 1.0 to 2.0 µm 4 grisms on a wheel to read redshift data, which is in the range 0.7-2.0 µm

Analyzed Missions – Solar Orbiter

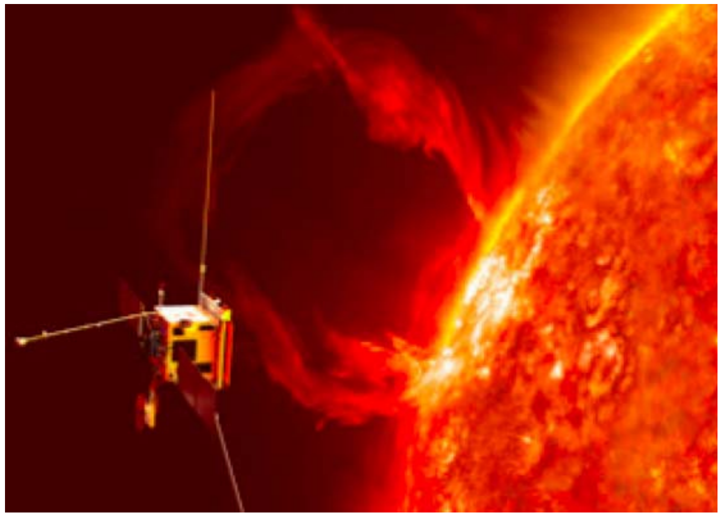
Category	Solar Orbiter (remote only)
Mission Type	Solar with 2 years transit time
Detector Type	**EUI:** APS **METIS:** APS **PHI:** APS **SoloHI:** APS **SPICE:** APS **STIX:** Collimator
Instrument Type	**EUI:** Imager **METIS:** Imaging Spectrometer /Coronagraph **PHI:** Imager & Polarimeter **SoloHI:** Imager **SPICE:** Imaging Spectrometer **STIX:** Imaging X-Ray Spectrometer
Waveband	**EUI:** High Resolution Imagers (HRI): two sensors, one centred at Lyman-ɑ and the other at 17.4 nm (extreme UV). Full-Sun Imager: a single sensor with two interchangeable filters, at 17.4 and 30.4 nm (extreme UV). **METIS:** Broad-band (visible at 500-600 nm) and narrow-band (UV at 121.6 nm and EUV at 30.4 nm) **PHI:** Visibile **SoloHI:** Visible **SPICE:** Extreme UV, 70.2-79.2 nm, 97.2-105.0 nm and 48.5-52.5 nm **STIX:** X-ray

Selected Missions

Name	**ExoMars Trace Gas Orbiter (TGO)**
Type	Exploration
Target	Planetary
Status	Implementation
Launch	Planned in 2016
Objective	Gaining a better understanding of methane and other atmospheric gases that are present in small concentrations (less than 1% of the Martian atmosphere) but nevertheless could be evidence for possible biological or geological activity.
Orbit	Mars, circular ~400km altitude



Name	**Solar Orbiter**
Type	Cosmic Vision M1
Target	Solar Physics
Status	Implementation
Launch	Planned in 2017
Objective	Investigation of heliosphere dynamics, including the generation of solar wind and its connection with the solar dynamo.
Orbit	Sun, elliptic orbit with perihelion near 0.28 AU and inclination of ~30º



Advantages of Each Mission

- **ExoMars TGO**
  - Measuring properties of atmosphere, surface and sub-surface planetary features.
  - Different implementation of Spectrometers
  - Has an Imager with CCD sensors
  - Has a Particle Detector
- **Solar Orbiter Remote Sensors**
  - Six remote-sensing instruments, including Imagers, Coronagraph and Polarimeter
  - Many similarities with astronomy instruments. FITS format data.
  - APS sensors instead of CCDs
  - RAL involvement with the SPICE instrument.

Overall Conclusions

- A total 5 different Instrument Types and 9 Detector Types in current selection
  - Detection of electromagnetic radiation (visible and infrared on both Solar Orbiter and ExoMars, additionally UV in the case of ExoMars's NOMAD instrument), particles (Fine Resolution Epithermal Neutron Detector)
  - Performing imaging, spectrometry, coronography, photometry and polarimetry
- Measurements depend on the spacecraft's position (ExoMars's Mars mapping) and orbital position (Solar Orbiter observation modes)
- Single-instrument results and examples of data from multiple instruments being combined
- Possibility of extending the RA of the ExoMars TGO to include the Lander instrument suite and Solar Orbiter RA to include the in-situ instruments in future iterations

Presentation of Software Requirements Documents

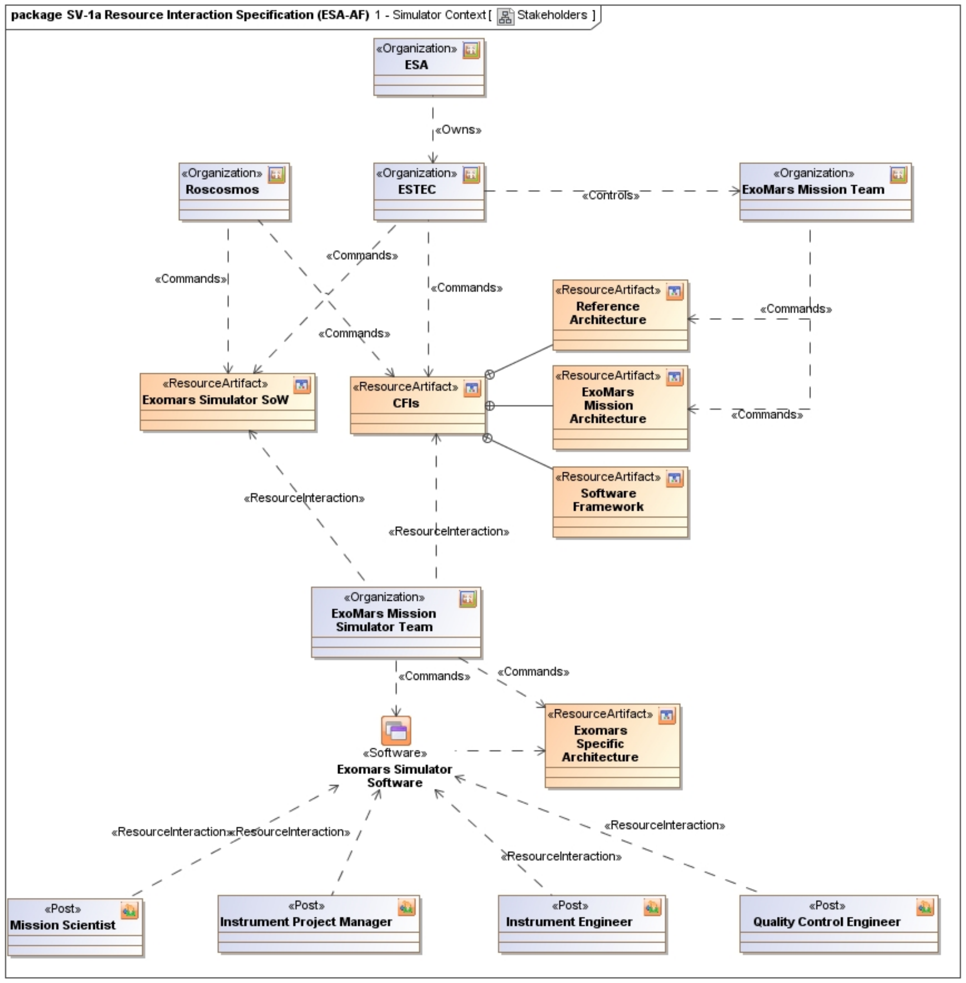
Generic Requirements Application

- The Generic Requirements were applied to each of the missions, taking into account their specificities. Requirements were divided into 4 categories:
  - **General Architecture**: These requirements include the definition of each mission General Architecture, its objectives and main tasks.
  - **Specific Architecture**: This is the Specific Architecture of each mission End-to-End Simulator, to be designed based on theReference Architecture. This set of requirements is an "instruction manual" on how to apply the Reference Architecture to ExoMars needs.
  - **Software Framework**: This set of requirements is transmitted to the E2ES Team in case they wish to evaluate several Software Frameworks.
  - **Software Implementation**: These are the requirements that direct the E2ES teams on making the transition from Architecture to Implementation (or paper to code).

Generic Requirements Application

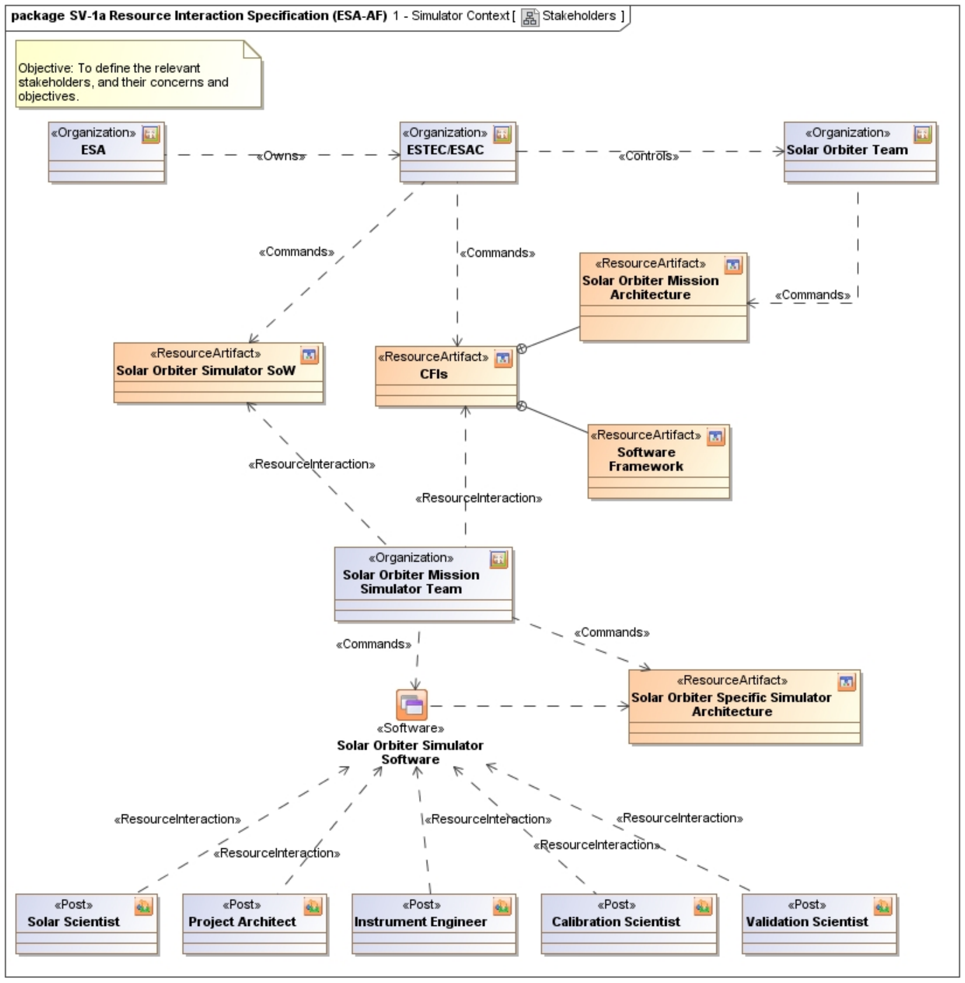
- But also, the RA initial definitions, such as Mission Context and Stakeholders were tailored for each of the missions:

ExoMars



package SV-1a Resource Interaction Specification (ESA-AF) 1 - Simulator Context [Stakeholders]
«Organization» ESA
«Owns»
«Organization» Roscosmos
«Organization» ESTEC
«Controls»
«Organization» ExoMars Mission Team
«Commands»
«ResourceArtifact» Reference Architecture
«ResourceArtifact» Exomars Simulator SoW
«ResourceArtifact» CFIs
«ResourceArtifact» ExoMars Mission Architecture
«ResourceArtifact» Software Framework
«ResourceInteraction»
«Organization» ExoMars Mission Simulator Team
«ResourceArtifact» Exomars Specific Architecture
«Software» Exomars Simulator Software
«Post» Mission Scientist
«Post» Instrument Project Manager
«Post» Instrument Engineer
«Post» Quality Control Engineer

Solar Orbiter



package SV-1a Resource Interaction Specification (ESA-AF) 1 - Simulator Context [Stakeholders]
Objective: To define the relevant stakeholders, and their concerns and objectives.
«Organization» ESA
«Owns»
«Organization» ESTEC/ESAC
«Controls»
«Organization» Solar Orbiter Team
«Commands»
«ResourceArtifact» Solar Orbiter Mission Architecture
«ResourceArtifact» Solar Orbiter Simulator SoW
«ResourceArtifact» CFIs
«ResourceArtifact» Software Framework
«ResourceInteraction»
«Organization» Solar Orbiter Mission Simulator Team
«ResourceArtifact» Solar Orbiter Specific Simulator Architecture
«Software» Solar Orbiter Simulator Software
«Post» Solar Scientist
«Post» Project Architect
«Post» Instrument Engineer
«Post» Calibration Scientist
«Post» Validation Scientist

Conclusion

- Both Software Requirement Specifications share a lot of content and most of the Generic requirements are directly re-usable
- Differences between the final Requirements lists should mostly come from the Mission Specific requirements, which we have not tried to cover in this exercise
- The list of generic requirements is extremely valuable because it provides the teams a checklist of what to make sure is specified
- The addition of context and stakeholder templates is also very valuable in the sense that it helps team identify how to organize the work groups, who should be heard in which stage and how the information should flow between the whole team

Evaluation Criteria 1/4

Category	Criteria	Relative effort w/wo Reference Architecture	Comments
Programmatic Criteria	Management and coordination: time needed to follow the development of the SS-E2E simulator and coordination of activities	**0.7**	It is expected that the time needed for this will be reduced as a consequence of following a standard procedure as recommended here.
	Requirements definition: time needed to define the requirements specification of the simulator	**0.4**	General requirements are already defined in [RD.6]. Only mission-specific requirements are needed in addition.
	Architecture and interfaces definition: Time required to define the architecture of the simulator and its interfaces	**0.5**	The RA specifies an architectural design pattern as a standard way to design an end-to-end simulator. As with all effective design patterns, there are several advantages. 1. The design work is already mostly done and there is only a need for mission-specific tailoring. 2. Using a proven design significantly reduces risk and its associated costs of overruns and even project failure. 3. All stakeholders (users, architects, developers etc.) become familiar with a standard design, terms and definitions. This helps prevent miscommunication and improves efficiency in subsequent similar projects. Standardised interfaces further reduce the definition cost, and moreover help to promote software reuse.
	Modules defintion, development and validation: Time needed to define, develop and validate the simulator modules	**0.6**	This partly follows on from the previous point. Following a standard architecture and interfaces should by itself reduce not only definition time, but also helps with development and verification/validation due to the modular design minimising coupling and promoting testability. A further pay-off is to a large extent dependent on the success of an exercise to identify generic and reusable building block implementations. We have established that there is good deal of commonality across all modules in terms of steps to be applied. In some cases it can be seen that generic implementations are possible, but it remains to be seen exactly how widely that can be applied. It is a tantalising thought to consider chaining together generic building blocks parameterised by the mission and instrument details in simple scripts.

Evaluation Criteria 2/4

Category	Criteria	Relative effort w/wo Reference Architecture	Comments
Programmatic Criteria	Simulator integration: Time needed to integrate the complete simulator	**0.6**	This is expected to be significantly simplified due to the standardisation of modules and interfaces. It may also be possible to construct a standard framework for automated integration testing. Savings could be made here using continuous integration whether the RA is used or not.
	Simulator verification and validation: Time needed to verify and validate the simulator scientifically and functionally. In terms of parameter checking, it will consist in three main activities: 1. Syntactic checks 2. Units checking 3. Semantic (physical meaning) checks	**0.85**	Verification has overlap with integration in checking that the overall simulator will run, helped by the standard architecture and interfaces. Generic and verified building block implementations with good unit tests would provide enormous benefit. It is harder to make a case for validation. Unit and semantic checks of parameters would certainly help, but ultimately scientific validation can only be done in the general case, at least at this time, by trained scientists inspecting the data. In certain cases, there may be more benefit. For example, if the point of the simulation is to determine whether a target number of stars can be identified at the end of the retrieval modules over noise and confusion, then the performance module could compare the number of stars output from the point source extraction module with the number injected into the scene creation. Potentially it could also check photometry. Note that this scenario requires a Level-3 module.
	Maintenance: Time devoted to simulator maintenance	**0.7**	Several factors already mentioned would contribute to reduced maintenance effort: ● Standard design. ● Standard interfaces. ● Pre-verified standard building blocks. ● Automated integration tests.

Evaluation Criteria 3/4

Category	Criteria	Relative effort w/wo Reference Architecture	Comments
Technical)Criteria)	Execution*performance:**efficiency*of*the* execution*of*the*E2E*simulation,*etc*	**0.8)**	The*main*concern*here*is*to*do*with*using*files*as* interfaces.*The*reference*architecture*defines*file*as* interfaces*between*modules;*however*there*is*nothing*to* stop*them*being*using*at*sub?modular*levels,*in*fact*it*is* implicit*from*the*requirements.*If*these*file*interfaces*are* used*at*too*low*a*level*of*building*blocks,*or*if*data*rates* are*very*high,*it*is*possible*that*poor*performance*could* render*a*simulator*quite*unusable.*There*are*ways*to* mitigate*this*risk,*but*they*all*have*potential*drawbacks.* 1. Don't*define*low?level*building*blocks.*But*this*could* reduce*the*potential*for*code*reuse.* 2. Define*APIs.*This*has*issues*with*interfaces*across*several* programming*languages*and*portability.* 3. Implement*file*I/O*in*a*very*efficient*way.*This*might* require*a*very*deep*understanding*of*how*the*file* format*works,*or*even*a*new*implementation*of*the* format*libraries.* The*standard*modules*are*run*in*a*set*sequence,*so*there* seems*to*be*little*if*any*potential*for*parallelisation*at*a* high?level.*However*some*building*blocks*could*potentially* be*highly*parallel*and*this*could*be*a*criterion*for*their* identification.*
	Propagation*of*errors:*efficiency*of*detecting* and*isolating*a*failure*in*the*simulator*	**0.5)**	The*standard*architecture*should*help*in*determining*the* location*of*errors.* Modularity*is*again*an*advantage,*allowing*for*standardized* unit*testing*and*for*quickly*being*able*to*isolate*a*BB*output* and*assess*its*correctness*against*other*implementations*or* a*reference.* Moreover,*since*the*module*order*is*fixed,*this*adds*the* potential*to*be*run*by*some*execution*framework*which* can*report*and*log*errors*in*a*standard*and*coherent*way.* The*performance*analysis*module*can*detect*deviation* from*expected*results*and*report*accordingly.*

Evaluation Criteria 4/4

Category	Criteria	Relative effort w/wo Reference Architecture	Comments
Technical Criteria	Modularity: substituting one module or building block for another implementation	**0.3**	The whole reference architecture concept is strongly modular in its nature. The high-level modules are well-defined in their scope, with the caveat that some building blocks potentially apply at different data processing levels depending on the instrument. It should also be straightforward to change one building block for another with a different algorithm or implementation.
	Evolution capability for use in later phases: evolving the simulator for later phases of the missions.	**0.3**	As stated above, the inherent modularity allows building blocks to be substituted or upgraded over time without invalidating the overall architecture. The instrument and retrieval modules will require a good deal of evolution as the instrument is built and characterised, but there is no reason to think that this will impact on the overall system architecture. The scene and geometry modules are likely to be more stable. The impact on the science return of different orbits or attitude performance could in many cases be simulated by changing the input parameters rather than the algorithms. The reference architecture allows for real data to be injected into the data processing chain.

Roadmap Proposals

Roadmap: in-situ

- Expand the RA to include in-situ instruments.
- Mostly it is still applicable, but there are some different elements:
  - Formation flying (Cluster, Double Star).
  - Planetary and minor body rovers and landers.
  - CDF data format.
- This is in fact needed for complete coverage of our chosen missions ExoMars and Solar Orbiter.

Roadmap: EO

- Harmonise the SS and EO architectures.
- The ESA-AF model we have developed can be extended to EO.

Roadmap: Libraries

- Survey of libraries to look for reusable building blocks and supporting infrastructure.
- This is a big job to do properly:
  - There are a lot of libraries in a variety of languages.
  - Identify optimal algorithms for classes of instrument, data sizes etc.
  - Implementations might be “not quite generic” or otherwise inadequate.
  - Recommendation could be to develop reusable building blocks from scratch.

Roadmap: Data Products

- Add support for:
  - CDF (in-situ solar and STP)
  - netCDF (EO)
  - Others? (RA should allow for it)
- Analyse benefit of a “data model” independent of a specific data format.
  - Conversion would be an import/export operation.
  - Complicated by formats having different structure e.g. hierarchical versus flatter FITS format.

Roadmap: Tools & Frameworks

- While creating the RA and the website and while reviewing its contents, the team discussed possible expansions and improvements to the RA as a tool for users:
    - The model could be dynamic and able to provide a generic RA according to user inputs:
        - Enter mission name, type, orbit type, attitude type etc
        - Add instrument
        - Enter instrument name, waveband, type, detector type
    - The set of forms designed for the Proposal could be re-used as templates for inputs of a web application
    - Model-Based Engineering: generate scaffolding code out of the Information Systems Architecture (data and applications) to a target language of choice (C++, Java, Python, etc.)

Roadmap: Tools & Frameworks

- Automated Integration Test framework
  - May follow on from standardisation of modules and interfaces.
- Categorization Database
  - Database was created for categorization exercise.
  - It could be expanded and maintained.
  - A web-based front end could be added for querying.

Roadmap: TM packets

- The RA deliberately starts with Level-0 data products.
- Insertion of TM packets into the simulation allows for fuller end-to-end GS testing.
  - But requires standard TM description. XTCE?
- TM packet generation module.
  - Inverse problem of above.
  - Simulator would be capable of generating TM packets.
  - Useful for end-to-end GS testing.

Roadmap: Use it!

- The fundamental test of whether a reference architecture is correct and useful in a practical sense is to apply it to a real-world project.
- The goal is to go beyond a design and produce an actual working simulator.
- In the first place this probably should be for a relatively small and non-complex mission.
- This would likely result in further refinement of the architecture.