SS-E2ES-C

Space Science End-To-End Simulator Executive Summary Report

PROJECT REF.:	**SS-E2ES-C-D9**
ISSUE/REV.:	**1.2**
DATE:	**05/07/2018**

Prepared by:	**Anna Gregorio, Federico Dogo, Lucia Soto, Jose Barbosa**	**Date:**	**05/07/2018**
Reviewed by:	**Anna Gregorio, Lucia Soto**	**Date:**	**05/07/2018**
Released by:	**Anna Gregorio**	**Date:**	**05/07/2018**

Table of Contents

1 SS-E2ES Objectives

The objective of this activity was to propose a general infrastructure of an End-to-End Simulators (E2ES) for Space Science missions, namely an E2ES Requirements Baseline (RB), Reference Architecture (RA) and an associated library of software models (Building Blocks, BB) to promote reuse, standardisation and reduction of engineering costs by defining a products/science validation process throughout the lifecycle based on a E2ES. An overview of Space Science missions and instruments was performed to identify a reference architecture and generic BB.

There is no standard approach for an E2ES being used throughout all phases of space science missions. It can be argued that the reason for this is that instrument data processing is often, or even usually, the responsibility of the scientific community rather than ESA. This is in contrast to EO missions where ESA is responsible and a number of E2ES have been developed.
The availability of a standard architecture and library of BB, enabling the development of simulation scenarios without too much effort, could be of great benefit to the instrument teams. Furthermore, use of this architecture means that it is extensible to an end-to-end simulator than could be used in subsequent mission phases, which may not have been the case otherwise.
An E2ES itself consists of a set of software modules simulating the space segment, its data output and the subsequent ground retrieval. The execution of these software modules needs to be orchestrated including in particular invocation and provision of input data. The definition of a set of standardized conventions and requirements, which the modules have to adhere to, allows then the use of a common orchestrating framework.

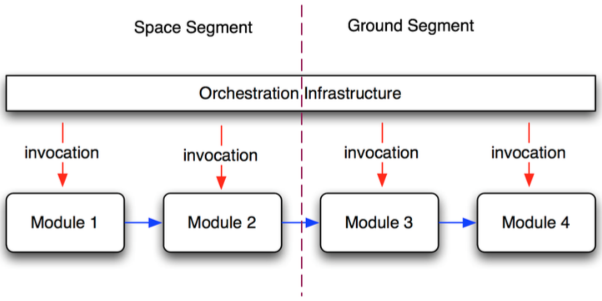

Figure 1-1: E2E Performance Simulator

This chain allows simulating the complete process and flow from a simulated scene (the *truth*) to the computed quantities, to introduce noise and errors, different instrument modeling as well as different data processing algorithms and ultimately to assess and characterize the performance of the whole chain as function of instrument design, data processing algorithm, noise and errors by comparing the simulated truth with the data as retrieved by the simulated ground processing.
In the early phases of a mission the E2E Performance Simulator supports the definition and the verification of the Space Segment requirements; in later phases it is used as an offline Test Data Generator for the Ground Segment and as breadboard for the ground processing.

This work was divided into two main phases, the first one dedicated to the analysis of a general Space Science simulator, SS-E2ES; then this study was applied to a specific demonstration mission identified as ARIEL.

2 SS-E2ES Requirement Baseline - RB

E2ES are built on the basis of technical requirements and of mission and science objectives. The SS-E2ES Requirement Baseline was defined based on previous works and personal experience. The requirements were grouped into main categories, divided with respect to the role they play during the simulator development:

- FUN Functional: operation specifications of the simulator;
- DES Design: design specifications of the simulator;
- INT Interface: portability of simulator's input and output;
- PER Performance: performance of the simulator;

- SIM Simulation Framework: software framework of the simulator;
- OPS Operational: simulator user's capabilities;
- V&V Verification & Validation: simulator checking;
- MOD Module: specific implementation of previous requirements into modules.

The requirements are not all to be met at the same time: as the space mission follows various stages of realisation, so does the related simulator project. Since the E2ES evolves with the science mission along its lifetime, at an early stage requirements can be just partly applicable or not even yet applicable. Three different E2ES stages were defined, according to the mission progress:

1. "Proto simulator" phases A/B1;
2. "Simulator B" phase B2;
3. "Full simulator" phases C/D up to in-Flight.

3 SS-E2ES Missions, Instruments and Building Blocks

We also defined a unique categorisation of Missions, Instruments and Building Blocks to support the application of the Requirements Baseline and Reference Architecture. Space science is a broad field that includes an enormous variety of mission types, targets, instrument types and different sensor technologies.

From a wide possible range, the following main categories were selected to group Space Science missions in categories relevant to an E2ES definition: **Mission type, Instrument and Detector type**.

The most important classification of Space science missions, which will shape the form of the simulator, is the type of mission. There are many types of mission, because each science mission is particular and developed ad-hoc. Four global types of Space missions have been identified:

- **Solar Science:** the field that studies the Sun.
- **Planetary Science:** the field that studies the planets of the Solar System.
- **Astronomy:** an extremely wide field that includes the study of celestial bodies and phenomena outside the Solar system**.**
- **Astrophysics**: a branch of astronomy that studies the physical laws, the properties and dynamic processes of celestial bodies and of the Universe, and their evolution.

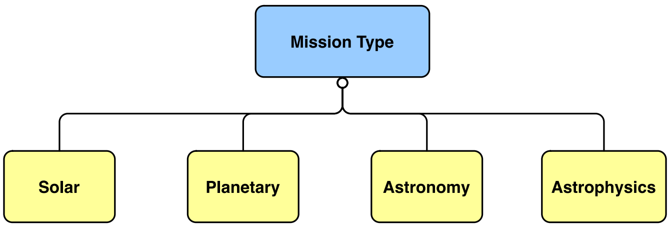

Figure 3-1: Mission type categorisation

The instrument is the payload of the satellite, where the desired information is captured and recorded. Science missions typically carry more than one type of instruments in order to capture various sources of information. The technology associated to each instrument is disparate, and it is one of the main drivers of an E2ES. Different instruments on board of the same platform share parts of the E2ES chain, notably the trajectory and platform orientation, and might share cross-calibration and processing, but the instrument model and processing are mostly instrument-specific.

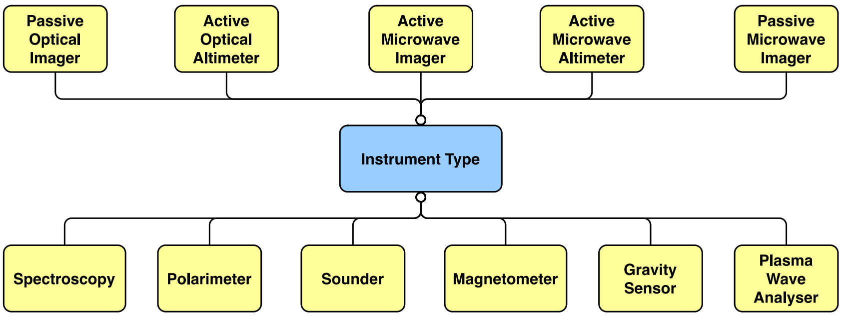

Figure 3-2: Instrument type categorisation

The detector is the sensing element that transforms a given physical variable to voltage, which is digitalised and saved in the on-board memory. The detector is intrinsically linked to the instrument, for instance the most common type of detectors for passive optical instruments (cameras) are Charged Couple Devices (CCD). Nevertheless, the category has been separated from the Instrument as there are cases where there are several detector techniques for a given instrument.

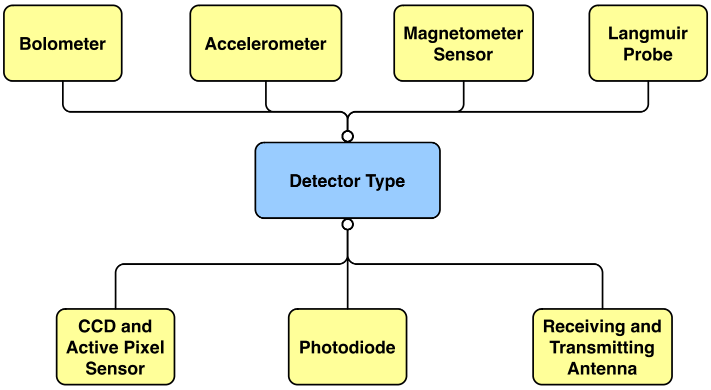

Figure 3-3: Detector type categorisations

4 SS-E2ES Reference Architecture - RA

The proposed RA was based on the generalization and restructuring of several existing architectures from previous Space Missions. It is divided into Main Architecture Modules, which are common for all Missions, and by Building Blocks, which can be generic (e.g. Orbit Simulator blocks) or, in most cases, specific to the mission, instrument or detectors to be modelled.

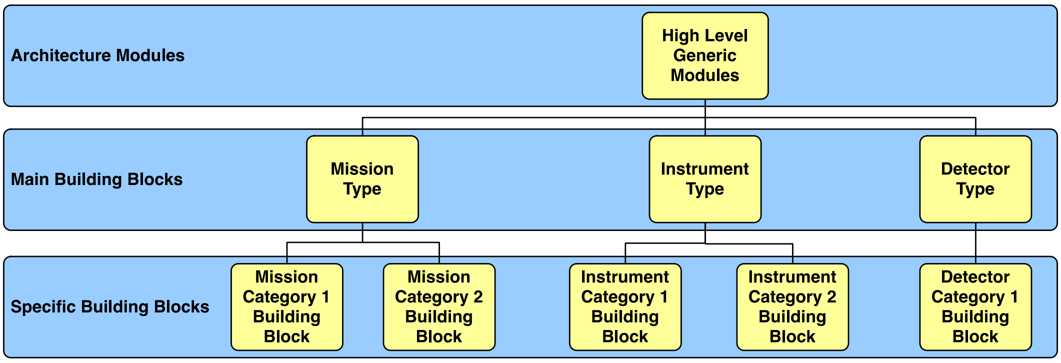

Figure 4-1: The Reference Architecture Concept

For this activity, we chose to adapt ESA-AF, an Architectural Framework developed by ESA, to our particular needs. ESA-AF was designed to support specifically the development of Space Missions software and we will be using its standards and notations.

The implementation strategy for any Space Science mission is to follow a number of steps, checking that the mission needs could be accommodated into the provided architecture, substituting the provided component names for ones closer to the specific domain and adding the missing components. The summarized steps are:

1. Set up the simulator context in accordance with the RB .

a) Define the Space Science mission context.
b) Identify the stakeholders and list their objectives and concerns.
c) Check if the intended high-level capabilities taken from the RB are supported by the provided RA.
d) Plan the simulator capability phasing.

2. Set up the Simulator Overall Architecture.

a) Match the provided target architecture of BB, data and data flows with the different simulation stages.
b) The generic main modules are:
   - The **Observation Timeline** provides the instrument pointing as a function of mission time.
   - **Geometry Module** generates a Field of View Definition from mission orbital status and platform position and pointing, defined in a scenario.
   - The **Scene Module** generates a real or synthetic “Scene Description” from the Scene Model Data taken as external input.
   - The **Geometry Intersection and Forward Module** generates the “stimuli” which will be perceived by the instruments taking as inputs the FOV Definition and the Scene Description.
   - The **Instrument Module** simulates the Instrument response to the “Stimuli” and “FOV definition” coming from the previous modules.
   - The **Platform Module**, that simulates the platform itself and its components.
   - **On-Board Processing Module** uses the Instrument Model data and produces raw data.
   - The **Data Processing Modules** convert the raw product into final science products, at the end of the processing chain.
   - The **Performance Assessment Module** closes the loop, comparing the initial scene with the retrieved scene from the simulator.

3. Specify the detailed Simulator Architecture.

a) Describe the building blocks using the RA model.
b) Define the building blocks of the simulation modules using the provided model.
c) Define the data structures: simulated data products, ADFs and configuration files.
d) Each BB shall be associated with a set of configuration parameters.
e) A consolidation work has to be performed on the parameters, in order to ensure homogeneity.

4. Describe your technology architecture

a) Define the configuration and implementation options of the software framework used for the simulations, either the one provided by the client or a different one.
b) Define the format standards used for all data products.

Once the SS team has followed these steps, the architectural design for their simulator should be finished and ready to be implemented in subsequent development stages.

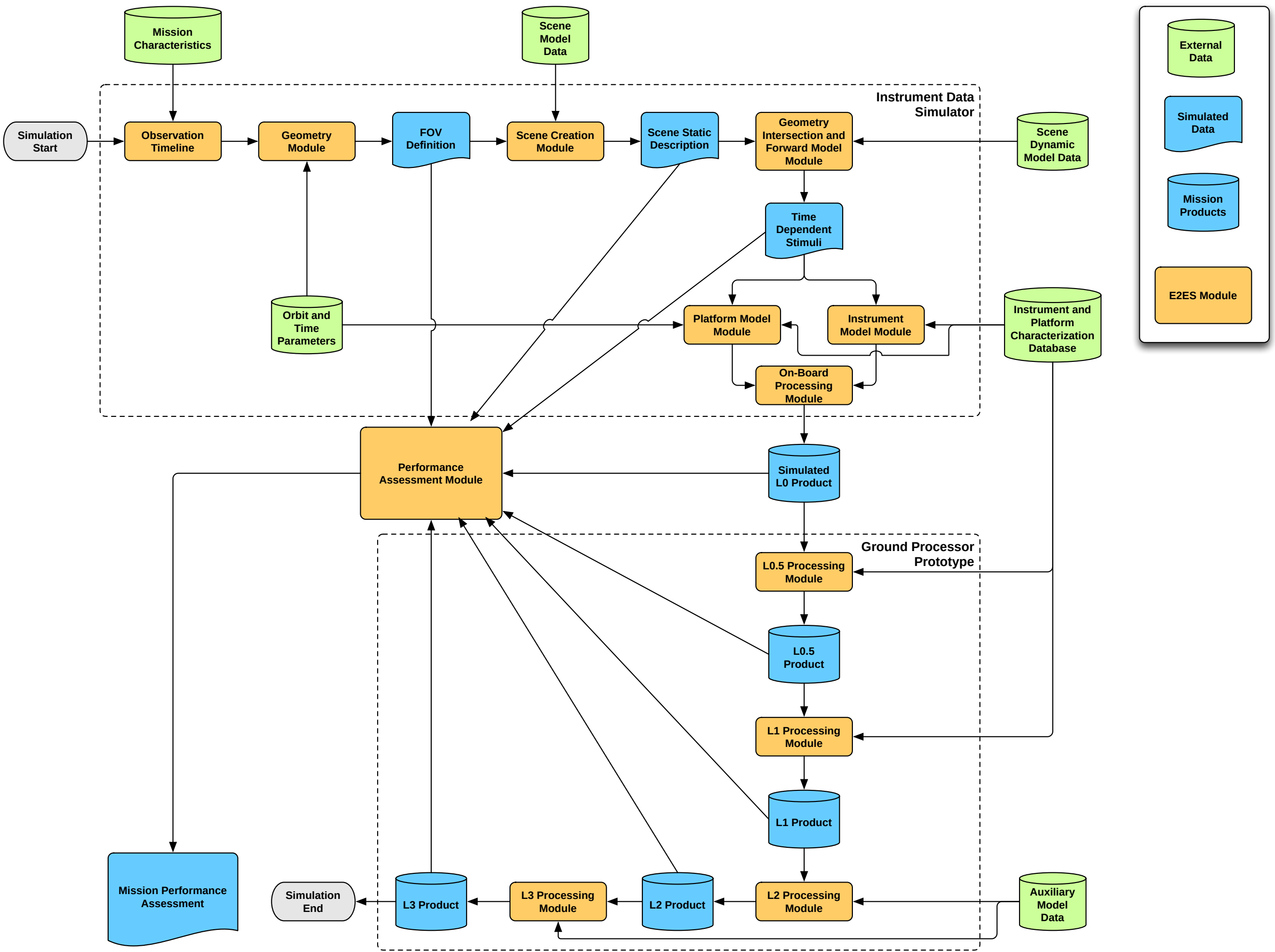

Figure 4-2: End-to-end mission simulation chain - Loop layout

5 ARIEL E2ES Requirements and High Level Architecture

5.1 ARIEL E2ES Requirements

ARIEL E2ES identified requirements are derived from SS-E2ES RB and by analysing the ARIEL mission and science requirements. For each requirement, applicability to Prototype Simulator version, applicable to the actual ARIEL mission phase, and the Full Simulator was stated. In accordance with what is presented in the ARIEL Performance Analysis Report, the Prototype simulator will focus on three target cases:

1) The faintest star to be observed by ARIEL - target GJ1214.
2) The brightest star to be observed by ARIEL - target HD219134.
3) An intermediate target at K magnitude of 6.3 which represents the boundary condition for a 'bright' target, target HD209458.

A special mention is deserved for the ARIEL scientific top-level requirements, which represent the Figures of Merit (FoM) of the E2ES, at least for the Prototype simulator:

a) The spectral resolving power;
b) The signal-to-noise ratio and noise requirements;
c) The photometric stability;
d) Calibration: the spectrometer absolute photometric calibration;
e) Calibration: the spectrometer absolute wavelength calibration.

5.2 High Level Architecture

The ARIEL E2ES RA was defined starting from the SS-E2ES RA and the ARIEL E2ES Requirements. This definition is divided into the following components:

- **High-Level Architecture design** - Logical analysis of high-level modules (building blocks) for the end-to-end simulator.
- **Data Specification** - Logical analysis of data and data flows between systems structures, including products and model configuration parameters.
- **Building Blocks Architecture design** - Logical analysis of system structures, in this case end-to-end simulator building blocks, and definition of models on different granularity levels for each structure.

In the Architecture proposed, the Observation Timeline, the Geometry Module, as well as the Scene Module, are common to all the instruments.

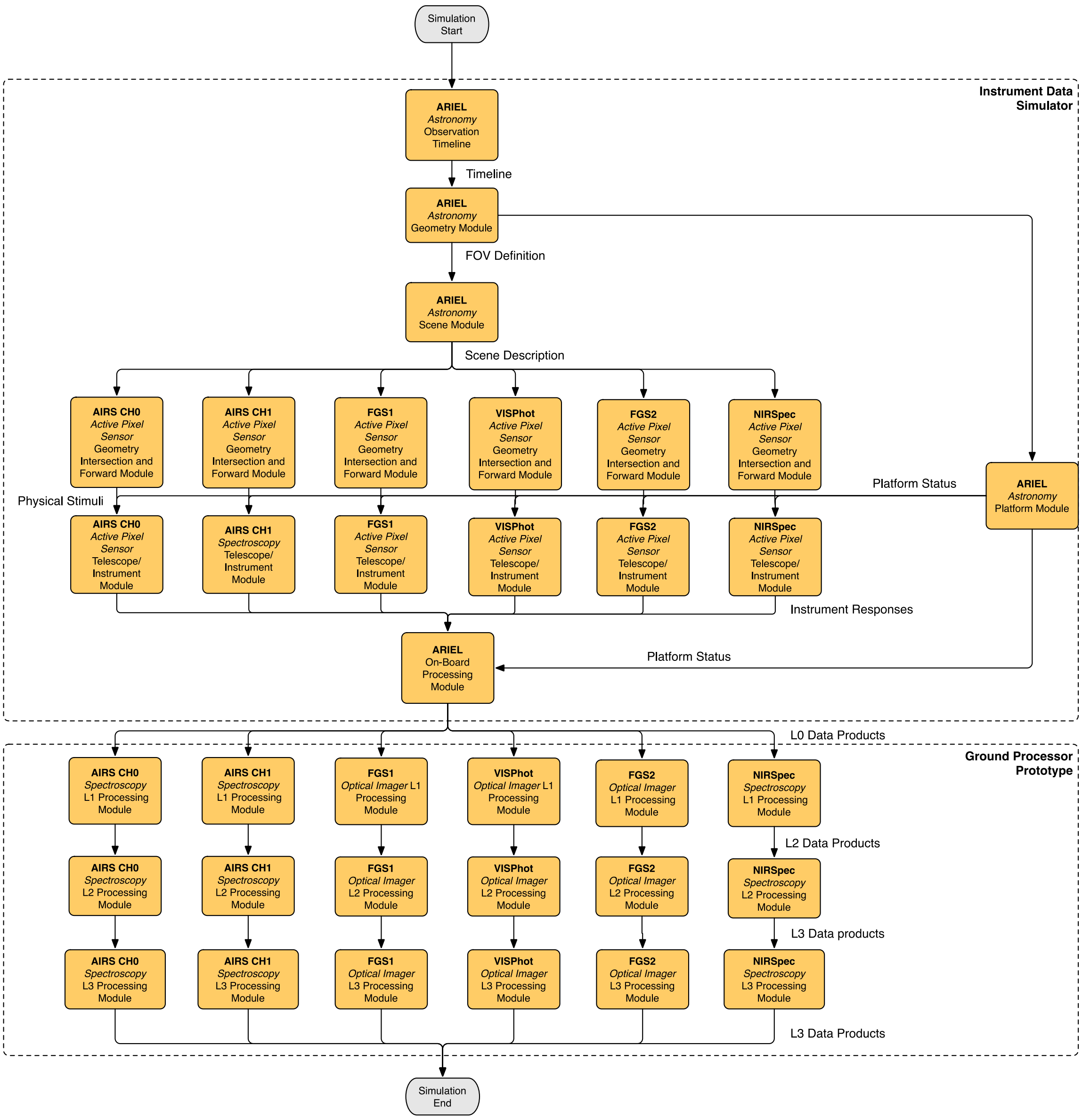

Figure 5-1: ARIEL High-Level Architecture

While this generalization is certainly advisable (all the instruments share the same platform and look at the same scene), the generalization of the physical simulations is no longer possible since there are different characteristics for each detector that are interesting to keep in separate modules. So, starting from the Geometry Intersection & Forward Module, different simulation chains (and processing pipelines) are proposed for each of ARIEL's six detectors. This first separation is done to take into account the different optical paths and the different detector characteristics (wavelength, bandwidth, etc.). Then, of course, the Instrument Modules must also be separated. However, all detectors again share the Platform and On-Board Processing Modules.

The Data Processing pipelines themselves, although sharing much functionality for each of the two types of detectors (Spectrometer and Photometers), are also separated by detector. This proposal can be revisited at a later time as more information about the algorithms for the ARIEL mission become available.
One last note must be made about the Fine Guidance System. Although it is used to feedback and tune the instrument pointing, this feedback is a feature not initially considered in the generic SS-E2ES RA. Since this can certainly be modeled and included in a further iteration of the ARIEL E2ES Architecture (in the Platform Module), it was decided to interpret the FGS channels as simple photometers for the time being.

6 SS and ARIEL-E2ES Building Blocks Technical Specification - TS

The technical specifications of identified building blocks that compose the modules for each category for a generic SS-E2ES were provided, as well as identifying the commonalities. Based on the ARIEL E2ES RB and RA, ARIEL E2ES Building Blocks were selected and further specified to conform to ARIEL needs.
It is important to mention that the BB were defined based only on publicly available ARIEL documentation and therefore were a best guess. The objective was to exemplify how the BB would be defined and articulated in the context of the application of the RA to an ARIEL E2ES and not to accurately describe all the details of the ARIEL mission data simulation and processing.
The Building Blocks defined were:

Processing Level	**Category**	**Building Block**
Observation Timeline Module	Generic	Instrument Scheduling Block
		Instrument Scan Law
Spacecraft Geometry Module	Generic	Orbit Simulator
		Attitude Simulator
		Instrument Pointing Simulator
		Field of View calculator
		Perturbations Block
Scene Creation	Exoplanet Mission	Sky Map (Astroscene Module)
		Image Assembly Engine
		Exoplanet Astroscene
		Black Body Emissions Calculator
		Planetary Spectral Emission Module
		Exoplanet Model
		Exoplanet Orbital Model
		Stellar Flux Calculator
		Stellar Limb Darkening Calculator
Geometry Intersection and Forward Module	Generic	Scene Interaction Geometry
		Stimuli Generation
Instrument Module	Generic	Optics Building Block
	Active Pixel Sensor	Generic Blocks
Platform Module	Generic	Propulsion Subsystem Block
		Power Subsystem Block
		Communications Subsystem Block
		Structure Subsystem Block
		Thermal Subsystem Block
		Telemetry and Command Subsystem Block
On-board Processing Module	Generic	Data Processing
		Data Formatting - Compression & Telemetry Block
		Integration Block
		L0 Formatter Block
L0 to L1 Processing	Generic	Unpack Telemetry
		Decompression
		Sorting
		Add Auxiliary Data
		Unit Conversion
		Time Correction/Conversion
		Masking
		Data Extraction & Quality Control
		Measurement Pre-Processing
		Time Domain Integration
	Spectrometers	Basic corrections for Spectrometers Sub-block
L1 to L2 Processing	Generic	Cosmic Ray Removal / Deglitching
		Flux Calibration
		Pointing Errors (Jitter) Compensation
		Dark Current Subtraction
		Crosstalk
		Linearity
		Velocity Correction
		Non-Linearity Correction

		Thermal Drift Corrections
	Imagers	Detector Modulation Transfer Function
		Flat Field Correction
		Vignetting Removal
		CCD Fixed Pattern Noise Removal
		Point Spread Function (PSF) Calculation
		Straylight Correction
	Spectrometers	Phase Correction
		Telescope Emission Calculation
		Transient Correction
		Spectral Calibration
L2 to L3 Processing	Imagers	Long Term / Persistent Transient Correction
	Spectrometers	Spectral Rebinning
Performance Assessment Module	Generic	Generic Blocks

7 The E2ES Evaluation

The quality of the SS-E2ES design was assessed taking into account some points, such as:

- Efficiency of the design process;
- Potential reuse of building blocks between different simulator of SS missions, and from EO missions;
- Comparison of the proposed development concept with previous developed simulators;
- Efficiency of development and validation process;
- Quality of results.

In particular, an estimation of the Reference Requirements Baseline and Architecture quality was performed by means of the following evaluation criteria, with rank given in terms of 'Relative effort w/wo Reference Architecture' (i.e., 0.7 means the design saves 30% of effort):

Programmatic criteria	
Criteria	*Relative effort*
Management and coordination: time needed to follow the development of the SS-E2E simulator and coordination of activities	0.7
Requirements definition: time needed to define simulator requirements	0.4
Architecture and interfaces definition: Time required to define the architecture of the simulator and its interfaces	0.5
Modules definition, development and validation: Time needed to define, develop and validate the simulator modules	0.6
Simulator integration: Time needed to integrate the complete simulator	0.6
Simulator verification and validation: Time needed to verify and validate the simulator scientifically and functionally	0.8
Maintenance: Time devoted to simulator maintenance	0.7
Overall Simulator Development	**0.6**

Technical criteria	
Criteria	*Relative effort*
Execution performance: efficiency of the execution of the E2E simulation, etc.	0.8
Propagation of errors: efficiency of detecting and isolating a failure in the simulator	0.5
Modularity: substituting one module or building block for another implementation	0.3
Evolution capability for use in later phases: evolving the simulator for later phases	0.3
Overall Validation	**0.5**

As a result, there are several advantages gained by the application of a reference E2ES, mostly in standardisation and reuse:

- Standardisation of terminology. Different missions and instruments may use different terms for the same thing, or even worse, the same term for different things. It is hoped that providing a reference architecture will help to promote standard terminology.
- Standardisation of requirements. There is a set of requirements that will be applicable to all mission simulators. The reference architecture will reduce the effort to identify them and to avoid missing important ones.
- Standardisation of 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. Moreover, skills acquired will be transferrable to other missions as the design remains familiar.
- Standardisation of 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.
- Standardisation of implementations. 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. Some modules may not be implemented in the operational RA, but having their design ready, plus some building blocks available for reuse providing part of its functionality, and supporting libraries that significantly ease the implementation, would greatly reduce the effort to implement them. 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.

All of these advantages are very apparent when the Reference Architecture is applied to a real mission End-to-End simulator design process.

Finally, probably the most important point, the implementation of an E2ES at a very early mission stage, that can be quickly and efficiently implemented, will provide a solid support to demonstrate the key technical and scientific aspects of the mission.

8 Roadmap

In order to support the definition of the SS-E2ES through the RA, the accurate management of the Building Blocks (BB) is a best practice. Indeed, it would be desirable to put effort in improving the specification of the identified generic BBs, mainly because it will allow reducing the number of BBs to be developed from scratch; moreover, an adequate reuse would reduce also the recurrent cost of detailed design of the specific SS-E2E simulators. For this purpose, the implementation of a model library with generic components must be driven by a serious analysis of the possible reuse, genericity and priority of the different elements.
For the roadmap steps and priority definition, the criteria we took into account are the following:

- The degree of genericity of the component and elements which can be common to many missions and already developed and available.
- Which parts of the processing are common to all E2ES: Geometry module, On-board and data processing, parts of the PAM.
- Which are the two most common types of instruments of Science Remote Sensing instruments: passive optical imagers and spectroscopy.
- The characteristics of the coming Space Science missions, which may drive the needs in the first step to certain types of missions.
- The availability of similar models from Earth Observation missions.

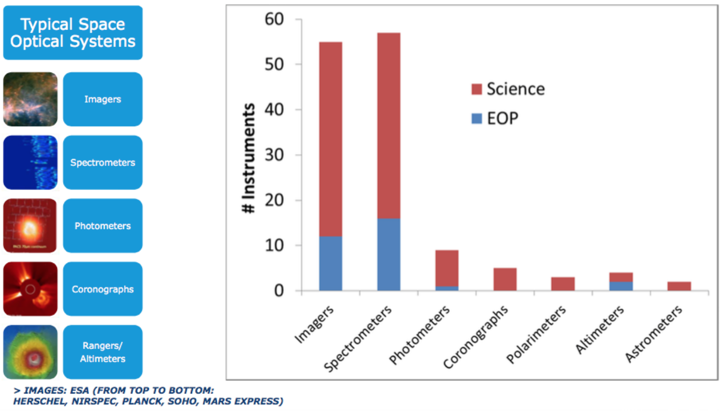

Figure 8-1: Survey of the most common types of instruments in Space Science (Credit: ESA)

Taking into consideration these criteria, the following priority BB were identified:

1. First priority – Geometry Blocks
2. Second priority – Blocks for Passive Optical Instrument and Spectrometry
3. Third priority – Blocks common with EO, part of the Instrument Model, or the calibration. Blocks for the on-board processing processing. The Geometry blocks of Planetary can be inherited from Earth Observation.
4. Fourth priority – Focus on next generation Science missions. The frequency of Science missions is low in comparison with EO, so it would benefit the potential development to have an assurance that it will be used before it gets obsolete.
5. Fifth priority – Follow and complete already existing libraries. Use existing developments as a starting point.

A preliminary estimate of the effort to implement the priority 1 & 2 Building Blocks was attempted. The effort of development of a single block was estimated to be 1.5 months (including detailed design, implementation, V&V, documentation). However, some blocks are containers (assigned 2 days) and some blocks might be reused from other developments, for ex. BIBLOS, and assigned a lower effort. Depending on the options taken during specification, the effort totals between 3.5 and 5 man-years.

Further Roadmap Proposals presented in the Evaluation and Roadmap TN are:

- Harmonisation of SS and EO architectures
- Survey of Libraries
- Data Products: Formats and Data model
- Tools and Frameworks
- Telemetry Packets
- Use the architecture for a real project

9 Overall Study Conclusion

This activity has evaluated the process of developing an E2ES for a Science mission, with the goal of finding the ways where there is margin to improve re-use, lower costs, and promote standardization. There is no doubt that End-to-End Simulators are useful for space science missions in Phase 0/A, and this is where the benefits of a reusable architecture and building blocks are clearest. Moreover, SS-E2ES would improve the science return of the mission by:

- Optimising scientific performance.
- Saving time and money on pipeline development and testing.

The RA can help to deliver both, because it provides a standard design with a solid software engineering base, thus reducing costs and is of great benefit in itself. Additional gains are achieved simply through standardization of terminology.

The provision of a standard set of reusable building block implementations would make SS-E2ES more affordable throughout the space community, being fundamental to follow an incremental approach in the implementation, based on the priorities agreed with all actors.
Furthermore, and as demonstrated, the availability of a Reference Architecture will save significant costs in the definition, detailed design and implementation of an E2ES. The eventual provision of a standard set of reusable building block implementations is more ambitious, but, if it can be achieved, it will surely go a long way towards making such simulators even more affordable throughout the space science community.

A roadmap to reach usable E2E RA for future Space Science missions is defined through the following activities, ordered sequentially in time:

1. Target a representative near future Science mission
2. Select a subset of models to implement
3. Re-use as much software as possible from existing libraries
4. Select the language, the framework, by speaking to the community
5. Identify the on-going Science activities, and the needs in the present and near future.
6. For future Space Science missions in which an E2ES will be developed, the RA documentation resulting from this contract should become an applicable document in the ITTs of those future E2ES. It is up to ESA to determine how mandatory will be to follow the RA and the associated repository of BBs.
7. In the meantime, follow-up activities should be started, to accomplish the detailed design and implementation of the modules identified as high priority.
8. Continue with the rest of building blocks of the RA, those considered with lower priority.

END OF DOCUMENT