A Reference Architecture for Earth Observation End-to-End Mission Performance Simulators

Cristina de Negueruela
Project Manager, Engineering and Studies (Space Systems)
Email: cnegueruela@gmv.es

INTRODUCTION

INNOVATING SOLUTIONS

INTRODUCTION

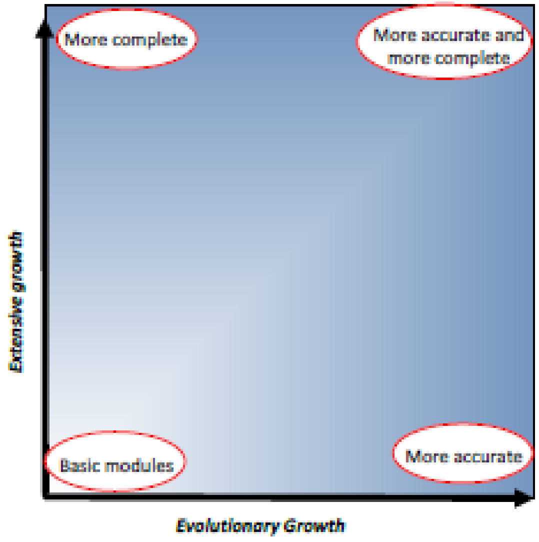
- End-to-end (E2E) Mission Performance Simulators are used in Earth Observation (EO) as a tool to assist during mission development and assess performance
- ESA is currently starting the development of these simulators during the mission feasibility studies,
  - If mission is approved, simulator will evolve into support tool for detailed design definition, preparation and validation of operations, data processing and higher-level mission products generation.
- However, the evolution of design and processing algorithms may require modifying or replacing components of the original simulator -> complex and costly reengineering process.
- ESA has promoted several activities in order to reduce this reengineering process, such as for example a simulation framework (OpenSF) able to support the development of the simulator throughout the mission life cycle.
- The **ARCHEO-E2E** activity, presented here, is framed into this context, and it has the main objective of **defining a Reference Architecture for Earth Observation end-to-end mission simulators**.

PURPOSE OF END-TO-END MISSION PERFORMANCE SIMULATORS

- End-to-end (E2E) Mission Performance Simulators are a useful tool to:
  - Support consolidation of technical requirements and evaluate system implementation options.
  - Assist development of retrieval algorithms at different data levels
  - Assess mission performance
  - Assess fulfillment of requirements by the mission
- In particular, end-to-end simulators (E2ES):
  - Enable generation of simulated Level-1 and Level-2 output data
  - Support assessment of end-to-end performance of mission on the basis of Level-1 and Level-2 products simulated for selected test scenarios
  - Support assessment of impact of individual error sources on output of an ideal system, both separately and simultaneously
  - Support assessment of performance of retrieval algorithms and their associated assumptions

EVOLUTION OF END-TO-END SIMULATORS

- Usually, first release of simulator is developed as a prototype tool to support initial performance assessment of the mission in Phase A.
- For the E2ES to evolve and support detailed mission design during later phases, its architecture has to allow growth along two possible directions:
  - Extensive growth, to include more effects and achieve a more complete simulation.
  - Evolutionary growth, to achieve more accuracy in the simulator.



- Therefore, the idea is to define a Reference Architecture that contains the basic modules for the E2ES, while providing the required flexibility to support both extensive and evolutionary growth.

THE GOAL OF ARCHEO-E2E

- **Defining a Reference Architecture for Earth Observation end-to-end mission simulators with the objective of promoting reuse in the development of mission performance simulators.**
- This goal is accomplished by:
  - Categorising past/current/ planned Earth Observation missions to identify main elements affecting mission performance and impacting simulator architecture.
  - Identifying architecture elements required to model mission depending on type of mission/instrument, and proposing a generic **Reference Architecture** that could be adapted for the different mission particularities.
  - Describing the architecture elements, in particular those that can be generalized for the various mission categories.
  - Evaluating the Reference Architecture by comparing the development of an end-to-end simulator using this new concept vs. ad-hoc simulator development.
  - Defining a roadmap to reach an operational concept for the development of end-to-end simulators based on the presented Reference Architecture.

MISSION AND INSTRUMENT SURVEY AND CLASSIFICATION

INNOVATING SOLUTIONS

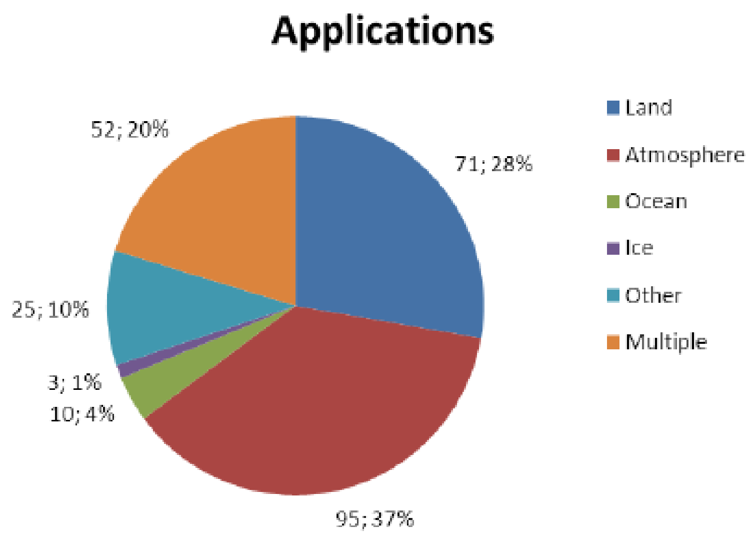
INTRODUCTION

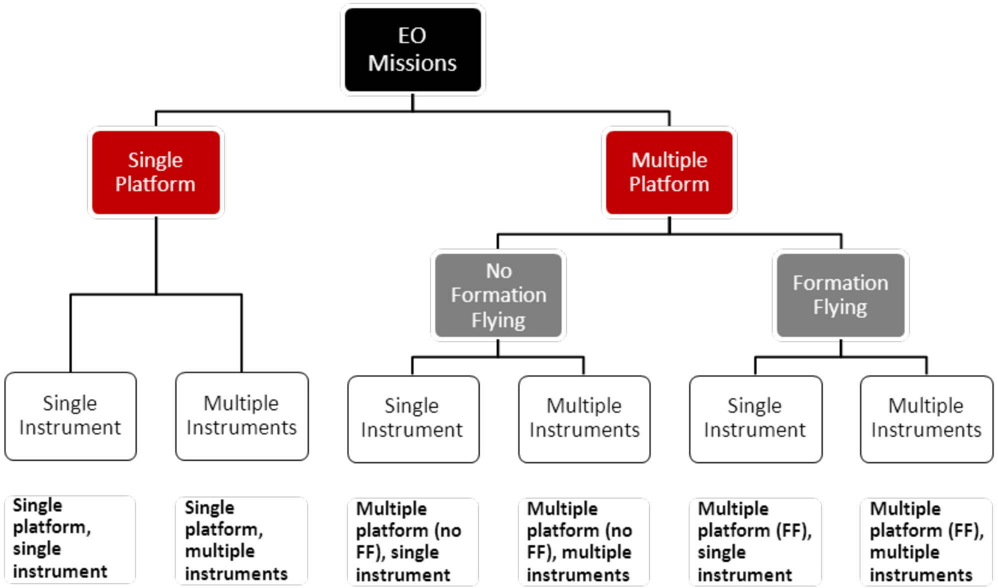
- Overview of EO Missions and identification of classification criteria that have the bigger impact over the E2ES architecture.
- Overview of EO Instruments and identification of classification criteria that have the bigger impact over the E2ES architecture.
- Instruments, while being part of mission, have such a big impact over the reference architecture that have been considered as a separate criteria.
- However, the high-level reference architecture is meant to be as generic as possible being valid for different types of missions and instruments.
- Analysis of commonalities
  - Between missions belonging to different categories and/or for different types of instruments.
  - Between missions from the same category
  - Between instruments from the same category

MISSION CATEGORIZATION

Mission categorization has been done according to those parameters that have the biggest impact over the E2ES architecture, e.g.:

- Number of satellites composing the mission
- Number of instruments on-board the spacecraft
- Observation geometry/scanning method
- Scientific objective of the mission
- Orbit characteristics

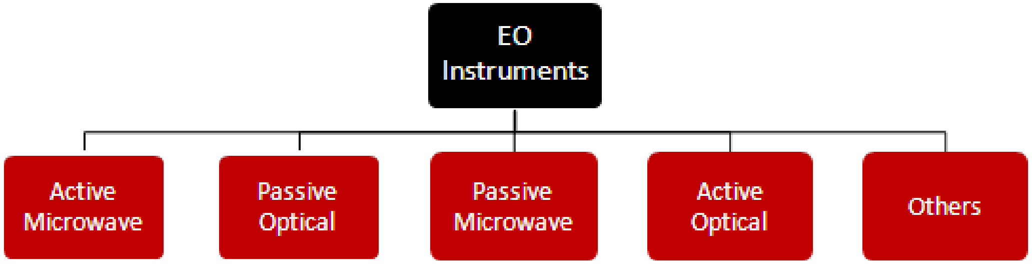




INSTRUMENT CATEGORIZATION

Instrument categorization has been decoupled from mission categorization since instrument type has a big impact at lower level in the E2ES architecture.

- Region of the spectrum at which the measurements are taken
- Passive vs. active instruments
- Target of the measurement
- Type of retrieval products
- Calibration method
- Scanning geometry



CONCLUSIONS

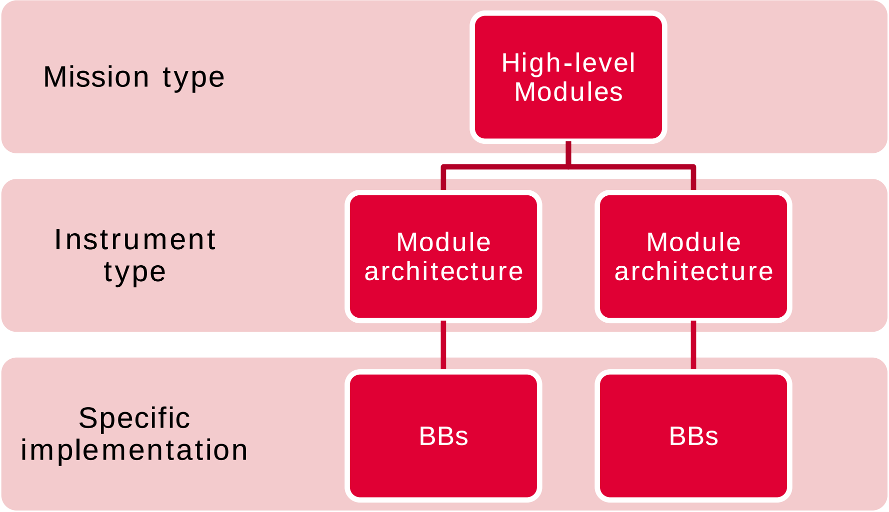
- The most important criteria affecting the reference architecture from the point of view of the mission have been identified.
- The different categories for different instruments will impact on the reference architecture at a second level.

E2ES REFERENCE ARCHITECTURE

INNOVATING SOLUTIONS

REFERENCE ARCHITECTURE CONCEPT

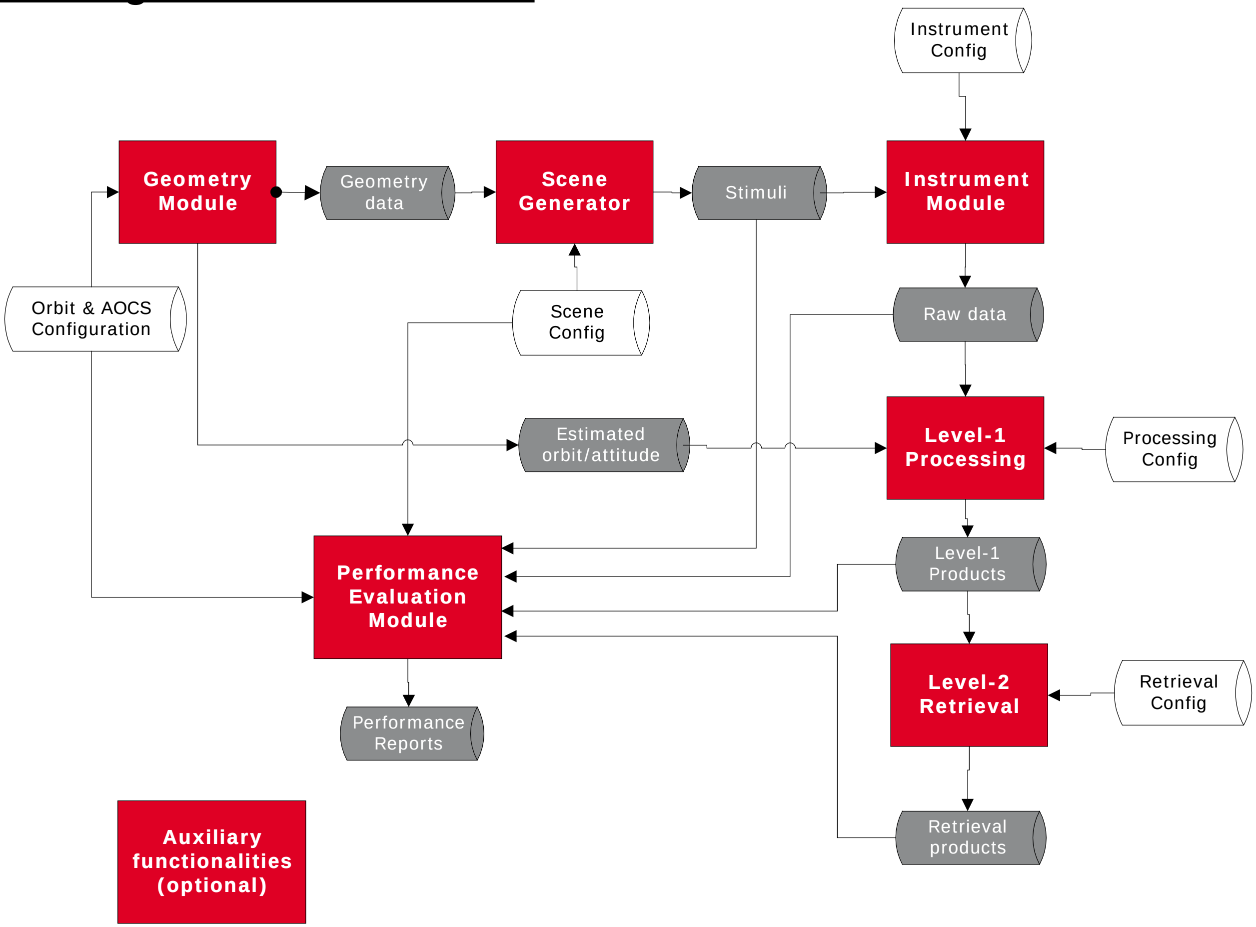
- The reference architecture defines a series of six high-level modules and the interfaces among them that are common to all type of missions and instruments.
- Although the reference architecture is generic, it is flexible to be adapted for the different mission particularities.



- Depending on the type of instrument to be simulated each of the six high-level modules will have an internal architecture broken down in building blocks.
- Different implementations of the same building blocks account for missions parameters, evolution of algorithms, etc.
- Some of the high-level modules and lower-level building blocks will be generic across missions and instruments.

REFERENCE ARCHITECTURE OVERVIEW

From mission categorization, analysis of commonalities and experience of project team, the reference architecture has been designed around the following six high-level modules:



HIGH-LEVEL MODULES

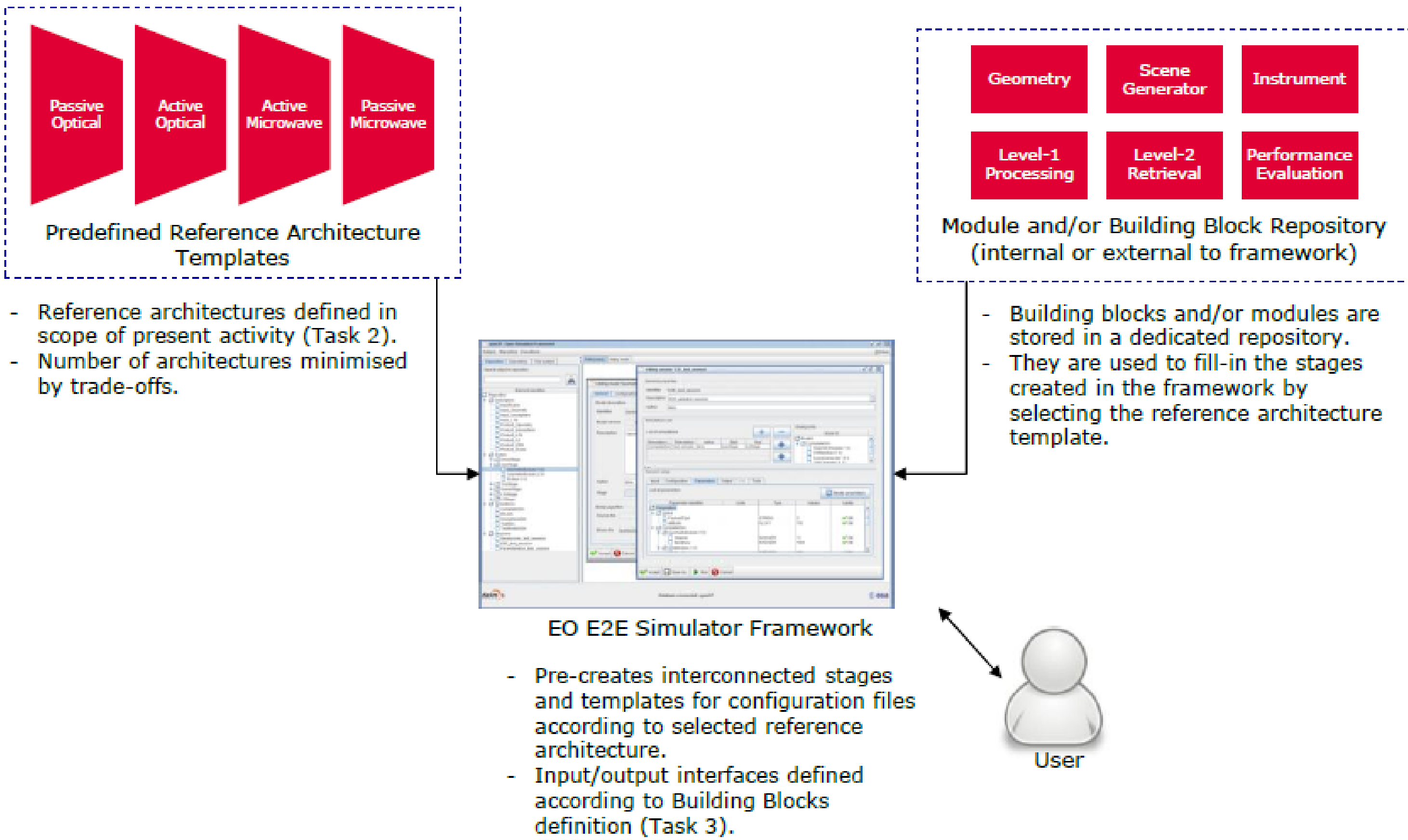
Module	Purpose	Configuration	Inputs	Outputs
Geometry	Simulates SC orbit & attitude & observation geometry of each instrument	-Orbit & AOCS configuration	N/A	-Geometry data
Scene Generator	Simulates scene to be observed and environmental effects needed for generation of stimuli to enter instrument model.	-Scene configuration	-Geometry data	-Stimuli
Instrument	Simulates sensor behavior, having different outputs depending on type of instrument.	-Instrument configuration	-Stimuli	-Raw data
Level-1 Processing	Generates level-1 products, from level-1a to level-1c.	-Processing configuration	-Raw data	-Level-1 products
Level-2 Retrieval	Performs retrieval of geophysical parameters objective of the mission/instrument.	-Retrieval configuration	-Level-1 products	-Retrieval products
Performance Evaluation	Performs analysis of simulator outputs to evaluate mission performances. It could be run at different points of the simulation chain.	-Orbit & AOCS configuration -Scene configuration	-Stimuli -Raw data -Level-1 products -Retrieval products	-Performance reports

ARCHITECTURE PARTICULARITIES

- Data-driven simulation assumed: each module generates at once all data needed, and following module in chain is executed.
- Modular architecture allows defining partial processing chains, to stop simulation at intermediate points, e.g. for performance evaluation.
- Lower-level architectures of each high-level module are also modular for flexibility and reuse.
- Auxiliary data/input perturbations handled at Building Block level.
- Performance Evaluation Module allows evaluation of performance at different levels, by accepting as inputs those coming from L-1 and L-2.
- The six high level modules can be combined to produce slightly different reference architectures at high level, e.g.:
  - Implications of having multiple instruments in the same spacecraft:
    - Instrument sharing application or retrieval methods/parameters
    - Instruments not sharing application or retrieval methods/parameters

PROPOSED SOLUTION

- Reference Architecture should cope with mission/instruments categories, but take into account framework supporting simulator development.
  - OpenSF considered as reference framework, but without conditioning definition of Reference Architecture.



BUILDING BLOCKS

INNOVATING SOLUTIONS

APPROACH TO DESCRIPTION OF BUILDING BLOCKS

- Each of the six high-level modules has been studied for each instrument type, with an architecture proposed and divided in BBs.
- The definition of these architectures and BBs has taken into account commonalities among different types of instruments.
- BBs granularity determined after identifying elements to be modeled.
- Level of detail in definition of BBs must allow composability (i.e. reuse) of the architectural elements, both at syntactic (engineering) level and at semantic (modeling) level.
  - Syntactic level: requires definition of implementation details such as external interfaces or configuration parameters to ensure engineering compatibility of the modules.
  - Semantic level: requires definition of metadata to ensure that not only models can be syntactically linked, but also that combined results are semantically valid. E.g. units of input/output data.
- This level of detail is achieved by defining a common custom template.
- Template also includes information on how to properly validate correct operation of BB and possible existing libraries to be reused.

BUILDING BLOCKS TEMPLATE (I)

Building Block Description	
General Information	
Building Block Name:	*Write the Building Block name here*
Instrument Type: (*tick all that apply with* ☑) ☐ Generic ☐ Active Microwave ☐ Passive Optical ☐ Passive Microwave ☐ Active Optical	
Module: (*tick applicable module with* ☑) ☐ Geometry Module ☐ Scene Generator Module ☐ Instrument Module ☐ Level-1 Processing Module ☐ Level-2 Retrieval Module ☐ Performance Evaluation Module	
Higher-level Building Blocks: *If applicable, list the higher-level building blocks up to the Module-level.*	
Functional Description: *Include a short functional description of the building block.*	
Scope of Application and Limitations: *Include details of under which conditions is the building block valid (e.g. type of scene, type of instrument, etc.).*	

BUILDING BLOCKS TEMPLATE (II)

Input Parameters and Constraints			
Mission (*include mission-related parameters that impact the Building Block)*			
Name	**Units/Format**	**Description**	
Name	*[Units]/Format*	*Description*	
Instrument (*include instrument-related parameters that impact the Building Block)*			
Name	**Units/Format**	**Description**	
Name	*[Units]/Format*	*Description*	
Interfaces *(include parameters that come from other Building Blocks or from external data sets)*			
Name	**Units/Format**	**Description**	**From**
Name	*[Units]/Format*	*Description*	BB Name / External
Output Parameters and Constraints			
Performance (*include parameters that can be used for performance evaluation of the BB/system*)			
Name	**Units/Format**	**Description**	**To**
Name	*[Units]/Format*	*Description*	BB Name / Performance
Interfaces *(include parameters that go into other Building Blocks)*			
Name	**Units/Format**	**Description**	**To**
Name	*[Units]/Format*	*Description*	BB Name

BUILDING BLOCKS TEMPLATE (III)

Implementation	
Composing Building Blocks:	
If applicable, list the sub-Building Blocks in which the present Building Block is decomposed.	
Name	**Description**
Name	*Description*
Name	*Description*
Name	*Description*
Algorithms:	
Give an overview of the different algorithms that could be used for the implementation of the Building Block, and how different algorithms would make the Building Block valid under certain conditions or with a higher level of performance at the sake of complexity. If applicable, list any limitations in the input/output data derived from the specific algorithm. **Option 1:** **Option 2:**	
Existing Libraries or Implementations:	
Refer to any existing libraries that could be used for the implementation of the Building Block, or to other existing implementations/E2E simulators from where the Building Block could be entirely or partially reused.	
Validation	
Include information on how to validate the correct operation of the Building Block, e.g. - What is expected from the model under certain operational conditions. - Type of test data to be used in the validation. - Preferred method for validation (e.g. comparative or analytical).	

CONCLUSIONS AND NEXT STEPS

gmv® INNOVATING SOLUTIONS

CONCLUSIONS

- A generic Reference Architecture has been identified. However the many cases of missions and instruments make unfeasible a static architecture valid for all purposes.
- The identification of common high-level building blocks with different architectures and broken down in building blocks promotes flexibility and reuse from one E2ES to another.
- The Reference Architecture is flexible enough to incorporate particularities of certain missions (e.g. multiple instruments, sharing retrieval methods…)
- This modular architecture allows defining partial processing chains implying the consecutive execution of some of the modules previously mentioned.
- Lower-level architectures of each high-level module are also modular for flexibility and reuse.

NEXT STEPS

- The Reference Architecture has been defined and the building blocks identified and described by means of a standard template.
- The next step will be to evaluate the Reference Architecture to gain an understanding of the advantages of the Reference Architecture approach with respect to the current approach.
- This evaluation will be done in three different areas:
  - Analyse the process of designing and developing an E2E simulator for a specific EO mission by applying the Reference Architecture.
  - Evaluate the proposed Reference Architecture concept compared to ad-hoc E2E simulators development.
  - Assess the capabilities of existing simulation frameworks and repositories to support the Reference Architecture and propose, if applicable, improvements to both of them.
- Final task in ARCHEO-E2E will be defining a roadmap to reach an operational Reference Architecture, including the identification of priorities in implementing generic building blocks and improvements to existing simulation framework and model repository.

Thank you

The ARCHEO-E2E Team:

Cristina de Negueruela, GMV

Fernando Alemán, GMV

Michele Scagliola, Aresys

Davide Giudici, Aresys

José Moreno, Universidad de Valencia

Jorge Vicent, , Universidad de Valencia

Adriano Camps, Universitat Politècnica de Catalunya

Hyuk Park, Universitat Politècnica de Catalunya

Pierre Flamant, IPSL / Laboratoire de Météorologie Dynamique

Raffaella Franco, ESA/ESTEC

INNOVATING SOLUTIONS