EO E2E2 Reference Architecture

Prepared by: ARCHEO-E2E Teams at:
GMV
Aresys
Universidad de Valencia
Universitat Politècnica de Catalunya
ESA

Approved by: Cristina de Negueruela Alemán

X

Authorized by: Cristina de Negueruela Alemán

X

ARCHEO-E2E

DOCUMENT STATUS SHEET

Version	Date	Pages	Changes
1.0	29/03/2012	185	First version released for PCR.
2.0 DRAFT	20/07/2012	187	Interim version for ESA's review. It includes comments received at PCR and a reorganization of the information presented to improve clarity.
2.0	19/09/2012	202	Updated version released for ICR. It includes comments received after the interim version delivered to ESA and updates in the different sections after a consolidation of the commonalities.
3.0	14/01/2013	213	Updated version released for FCR. It includes comments received at ICR.
3.1	27/02/2013	216	Updated version after FCR RIDs and comments.
3.2	26-03-2018	all	Updated to align with latest discussion with ESA on reference architecture and nomenclature in particular the Geometry module

TABLE OF CONTENTS

LIST OF TABLES AND FIGURES

1. INTRODUCTION

1.1. PURPOSE

This document presents the definition of a generic Reference Architecture to be used for the development of E2E EO Mission Performance Simulators, as well as a review of existing simulation frameworks and repositories.

The document is organized as follows:

- Section 1 is this introduction.
- Section 2 includes the applicable and reference documents.
- Section 3 presents the rationale behind the definition of a Reference Architecture for E2E mission performance simulators as well as the approach followed for this definition.
- Section 4 presents the Reference Architecture itself together with the specific considerations to be taken into account in the reference architecture for special types of missions. This section also describes the methodology used to describe this Reference Architecture.
- Section 5 presents the requirements for the development of an E2E mission performance simulator based on the concept of Reference Architecture.
- Section 6 presents an analysis of generic frameworks and module repositories, together with the implications of the analysed frameworks and repositories in the definition of the Reference Architecture.
- Section 7 presents the requirements for a framework and a repository of EO modules supporting the development of new end-to-end simulators based on the proposed Reference Architecture.
- Section 8 presents the high-level modules of the reference architecture, identifying those elements that are common across different instrument categories.
- Sections 9 to 12 include the specific considerations to be taken into account in the reference architecture, depending on the type of instrument being simulated. For each type of instrument, a baseline architecture is proposed together with the specific details and possible implementations of each of the high-level modules.
- Section 13 includes an additional work performed by UPC upon ESA's request to define the reference architecture for GNSS-R instruments.

While this document offer a thorough review of the proposed Reference Architecture and the details of the implementation for particular types of instruments, the different Sections may be aimed at different publics. Sections 3 and 4 are generic and should be of interest to all publics. Sections 5 to 7 are of more interest to those defining the work to be performed for an E2E mission performance simulator. From Section 8 onwards the contents are more specific to the developers of the simulator itself, and have all information for a particular type of instrument consistently together.

1.2. SCOPE

This document includes the results of Task 2 of the ARCHEO-E2E project, performed under ESA contract 4000104547/111NL/AF. This project is carried out by a consortium led by GMV and including the following members:

- GMV, responsible for management, system-level activities and active microwave instruments.
- Aresys, responsible for active microwave instrument.
- Universidad de Valencia, responsible for passive optical instruments.
- Universitat Politècnica de Catalunya, responsible for passive microwave and GNSS-R instruments.

In 2018, this document was updated by ESA to align with the latest discussions on architecture and openSF nomenclature.

1.3. ACRONYMS

Acronyms used in this document and needing a definition are included in the following table:

Table 1-1 Acronyms

Acronym	Definition
A/D	Analog-to-Digital
AOT	Aerosol Optical Thickness
BB	Building Block
BOA	Bottom Of Atmosphere
CCD	Charge Coupled Devices
DEM	Digital Elevation Model
DN	Digital Numbers
DOS	Dark Object Subtraction
E2ES	End-to-End Performance Simulator
ELM	Empirical Line Method
EO	Earth Observation
ESA	European Space Agency
FOV	Field Of View
FTS	Fourier Transform Spectrometers
GCP	Ground Control Points
GM	Geometry Module
GSD	Ground Sample Distance
IR	Infrared
IRF	Instrument Response Function
IM	Instrument Module
L1PM	Level-1 Processing Module
L2RM	Level 2 Retrieval Module
LMC	Length Modulated Cell
MIR	Mid Infrared
MODTRAN	MODerate resolution atmospheric TRANsmission
MTF	Modulation Transfer Function
NESR	Noise Equivalent Signal Radiance
NESZ	Noise Equivalent Sigma Zero
NIR	Near Infrared
PEM	Performance Evaluation Module
PMC	Pressure Modulated Cell
PRF	Pulse Repetition Frequency
PSF	Point Spread Function
QWIR	Quantum-Well Infrared detector
RTM	Radiative Transfer Model
SAM	Spectral Angle Mapper
SAR	Synthetic Aperture Radar
SNR	Signal-to-Noise Ratio
SVF	Sky Viewing Factors
SWIR	Short Wave Infrared
SWL	Sampling Window Length
SWST	Sampling Window Starting Time
TIR	Thermal Infrared
TOA	Top Of Atmosphere
TOC	Top Of Canopy
VIS	Visible
VNIR	Visible Near Infrared

2. REFERENCES

2.1. APPLICABLE DOCUMENTS

The following documents, of the exact issue shown, form part of this document to the extent specified herein. Applicable documents are those referenced in the Contract or approved by the Approval Authority. They are referenced in this document in the form [AD.X]:

Table 2-1 Applicable documents

Ref.	Title	Code	Version	Date
[AD.1]	ARCHEO-E2E Statement of Work	TEC-xxx/09-290/xx	1.0	21/01/2011

2.2. REFERENCE DOCUMENTS

The following documents, although not part of this document, amplify or clarify its contents. Reference documents are those not applicable and referenced within this document. They are referenced in this document in the form [RD.X]:

Table 2-2 Reference documents

Ref.	Title	Code	Ver.	Date
[RD. 1]	Statement of Work for the BIOMASS End-to-End Mission Performance Simulator	EOP-SFP/2009-05-1390	1.0	April 2010
[RD. 2]	Statement of Work for the CoReH2O End-to-End Mission Performance Simulator	EOP-SFP/2009-06/1397	1.0	April 2010
[RD. 3]	Statement of Work for the PREMIER End-to-End Mission Performance Simulator	EOP-SFP/2010-02-1450	1.2	April 2010
[RD. 4]	OpenSF Interface Control Document	OpenSF-DMS-ICD-001	3.0	26-05-2015
[RD. 5]	OpenSF Architectural Design Document	openSF-DMS-ADD-001	2.2	15-01-2014
[RD. 6]	OpenSF System User Manual	OPENSF-DMS -SUM-001	3.12	15-12-2017
[RD. 7]	SoW "Feasibility of Open Source Software Repository (OSSR)"	AO/1-6301/09/NL/AF	1.3	4th Oct. 2009
[RD. 8]	OSSR Software Requirements Specification	OSSRF-UNI-SRS-11D4	1.2	14th Oct. 2011
[RD. 9]	OSSR Software Design Document	OSSRF-UNI-SDD-11D5	1.1	20th Oct. 2011
[RD. 10]	EODiSP http://www.eodisp.org			
[RD. 11]	Scagliola M., Giudici D., D'Aria D., Acarreta J., Del Pozo E., "Snow Retrieval From Dual Band SAR Data: Preliminary COREH2O E2ES Results Analysis", presented at ARSI 2011, Advanced RF Sensors and Remote Sensing Instruments, Workshop on, ESA-ESTEC 2011			Sept. 2011
[RD. 12]	López-Dekker P., De Zan F., Börner T., Younis M., Papathanassiou K., Guardabrazo T., Bourlon V., Ramongassie S., Taveneau N., Ulander L., Murdin D., Rogers N., Quegan S., Franco R., "BIOMASS END-TO-END MISSION SIMULATION AND PERFORMANCE ASSESSMENT" ", presented at ARSI 2011, Advanced RF Sensors and Remote Sensing Instruments, Workshop on, ESA-ESTEC 2011			Sept. 2011
[RD. 13]	A. Monti Guarnieri, C. Prati, "ScanSAR focusing and interferometry" in IEEE Trans. on Geosci. and Remote Sens., Vol. 34, No. 4, pp. 1029			Jul. 1996
[RD. 14]	A.A. Kokhanovsky, W. von Hoyningen-Huene and J.P. Burrow, *Determination of the cloud fraction in the SCIAMACHY ground scene using MERIS spectral measurements*, International Journal of Remote Sensing, Vol. 30, No. 23, December 2009			December 2009
[RD. 15]	Luis Guanter, Karl Segl and Hermann Kaufmann, *Simulation of Optical Remote-Sensing Scenes with Application to EnMAP Hyperspectral Mission*, IEEE Transactions on Geoscience and Remote Sensing, Vol. 47 No. 7, July 2009			July 2009

Ref.	Title	Code	Ver.	Date
[RD. 16]	M. Parente, T. Clarke, A.J. Brown and J.L. Bishop, *End-to-End Simulation and Analytical Model of Remote-Sensing Systems: application to CRISM*, IEEE Transactions on Geoscience and Remote Sensing, Vol. 48, No. 11, November 2010			November 2010
[RD. 17]	*SkyHelios Official webpage*, http://www.urbanclimate.net/skyhelios/			
[RD. 18]	T. Poglio et al., *OSIrIS: a physically based simulation tool to improve training in thermal infrared remote sensing over urban areas at high spatial resolution*, Remote Sensing of Enviornment, Vol. 104 (2006)			2006
[RD. 19]	S.P.I.Rou: un outil de Synthèse de Paysages en InfraRouge par modélisation des échanges à *la surface*, International Symposium Optronics & Defense, Paris, December 1996			December 1996
[RD. 20]	Sanders, J.S. Johnson et al., *Ground target infrared signature modeling with the Multi-Servide Electro-optic Signature (MuSES) code*, Proceedings of SPIE- The international Society for Optical Engineering, Vol. 4029 (2000)			2000
[RD. 21]	J.P. Gastellu-Etchegorry et al., *Modeling Radiative Transfer in Heterogeneous 3-D Vegetation Canopies*, Remote Sensing of Environment, Vol. 58 (1996)			1996
[RD. 22]	J.P. Gastellu-Etchegorry, *3D modeling of satellite spectral images, radiation budget and energy budget of urban landscapes*, Meteorology and Atmospheric Physics, Vol. 102 (2008)			2008
[RD. 23]	C. Mattar, J.A. Sobrino et al., *Evaluación del modelo DART en el infrarojo térmico*, Teledetección: Agua y desarrollo sostenible. XIII Congreso de la Asociación Española de Teledetección, September 2009			September 2009
[RD. 24]	P. Gege, *The water colour simulator WASI: A software tool for forward an inverse modeling of optical in-situ spectra*, Computers & Geosciences Vol. 30 (2004)			2004
[RD. 25]	T. Kajiyama et al., *High-Performance Ocean Color Monter Carlo Simulation in the Geo-info Project*, Lecture Notes in Computer Science, Vol. 6068 (2010)			2010
[RD. 26]	T. Kajiyama et al., *Performance prediction of ocean color Monter Carlo simulations using multi-layer perceptron networks*, Procedia Computer Science, Vol. 4 (2010)			2010
[RD. 27]	Zhonghai Jin et al., *Introduction to an Online Coupled Ocean-Atmosphere Radiative Transfer Model*, http://snowdog.larc.nasa.gov/jin/rtnote.html			
[RD. 28]	J. Cheng et al., *Comparison of Radiative Transfer Models for Simulating Snow Surface Thermal Infrared Emissivity*, IEEE Selected Topics in Applied Earth Observations and Remote Sensing, Vol. 3, September 2010			September 2010
[RD. 29]	W. Verhoef, *Improved modeling of multiple scattering in leaf canopies: The model SAIL++*, Proceeding of the First Symposium on Recent Advances in Quantitative Remote Sensing, September 2002			September 2002
[RD. 30]	A. Kuusk, *A two-layer camopy reflectance model*, Journal of Quantitative Spectroscopy and Radiative Transfer, Vol. 71 (2001)			2001
[RD. 31]	W. Qin and Y. Xiang, *An analytical model for bidirectional reflectance factor of multicomponent vegetation canopies*, Science in China (Series C), Vol. 40 (1997)			1997
[RD. 32]	S.G. Leblanc and J.M. Chen, *A windows graphic user interface (GUI) for the five-scale model for fast BRDF simulations*, Remote Sensing Reviews, Vol. 19 (2001)			2001
[RD. 33]	P.R.J. North, *Three-dimensional forest light interaction model using a Monter Carlo method*, IEEE Transactions on Geoscience and Remote Sensing, Vol. 34 (1996)	Unival Rd.20		1996
[RD. 34]	W. Verhoef and H. Bach, *Simulation of hyperspectral and directional radiance images using coupled biophysical and atmospheric radiative transfer models*, Remote Sensing of Environment, Vol. 87 (2003)			2003
[RD. 35]	J. Helbert et al., *Hyemalis: simulateur 'images de paysages tridimensionnels complexes*, Revue Française de Photogrammétric et de Télédétection, Vol. 173/174 (2003)			2003

Ref.	Title	Code	Ver.	Date
[RD. 36]	W. Qin and S.A.W. Gerstl, *3-D scene modeling of semidesert vegetation cover and its radiation regime*, Remote Sensing of Environment, Vol. 74 (2000)			2000
[RD. 37]	Y. Govaerts and M.M. Verstraete, *Raytran: A Monte Carlo ray tracing model to compute light scattering in threedimensional heterogeneous media*, IEEE Transactions on Geoscience and Remote Sensing, Vol. 36 (1998)			1998
[RD. 38]	*Radiation transfer Model Intercomparison (RAMI)* official webpage, http://rami-benchmark.jrc.ec.europa.eu			
[RD. 39]	Z. Wang et al., *A Global Infrared Image Synthesis Model for Large-Scale Complex Urban Scene*, International Journal of Infrared and Millimeter Waves, Vol. 22, No. 8, August 2001			August 2001
[RD. 40]	S. Doz et al., *Simulation of Urban Optical Images from High Spectral and Spatial Resolution multi-angular Airborne Acquisitions*, Geoscience and Remote Sensing Symposium (IGARSS), 2010			2010
[RD. 41]	*On Modelling and Rendering Ocean Scenes*, Jean-Christophe Gonzato and Bertran Le Saëc			
[RD. 42]	R. Sundberg et al., *Improved full spectrum cloudy scene simulation*, Proceedings of SPIE, Vol. 7170 (2008)			2008
[RD. 43]	J.-P. Lagouarde et al., *Modelling daytime thermal infrared directional anisotropy over Toulouse city center*, Remote Sensing of Environment, Vol. 114 (2010)			2010
[RD. 44]	*SOLENE – Software for simulating of sunshine, lightning and thermal radiation*, http://www.cerma.archi.fr/CERMA/expertise/solene/soleneanglais/view?searchterm=solene			
[RD. 45]	H. Fujisada, *ASTER Level-1 Data Processing*, IEEE Transactions on Geoscience and Remote Sensing, Vol. 36, No. 4 (1998)			1998
[RD. 46]	T. Storch et al., *Processing chain for the future hyperspectral mission EnMap*, 6th EARSeL Imaging Spectroscopy SIG Workshop, Tel Aviv (Israel), March 2009			March 2009
[RD. 47]	G. Lichtenberg et al., *SCIAMACHY Level 1 data: calibration concept and in-flight calibration*, Atmospheric Chemistry and Physics, Vol. 6 (2006)			2006
[RD. 48]	Yvan Dutil et al., *ACE-FTS Level 0 to 1 Data Processing*, Proceedings of SPIE, Vol. 4814 (2002)			2002
[RD. 49]	G. Lichtenberg, *SCIAMACHY channel 1-5 Memory Effect I: Key data implementation and in-flight measurements*, Technical Report	SRON-SCIA-PhE-RP-11	2	
[RD. 50]	Luis Gomez-Chova et al., *Correction of systematic spatial noise in push-broom hyperspectral sensors: application to CHRIS/PROBA images*, Applied Optics, Vol. 47, No. 28 (2008)			2008
[RD. 51]	L.G. Tilstra et al., *Verification of SCIAMACHY's Polarization Correction over the Sahara desert*, Proc. of Envisat Validation Workshop, Frascati (Italy), December 2002			December 2002
[RD. 52]	Akihiko Kuze et al., *Vicarious Calibration of the GOSAT Sensors Using the Railroad Valley Desert Playa*, IEEE Transactions on Geoscience and Remote Sensing, Vol. 49, No. 5, May 2011			May 2011
[RD. 53]	Helen Worden et al., *TES Level 1 Algorithms: Interferogram Processing, Geolocation, Radiometric, and Spectral Calibration*, IEEE Transactions on Geoscience and Remote Sensing, Vol. 44, No. 5, May 2006			May 2006
[RD. 54]	Robert Knuteson et al., *Level 0-1 Algorithm Description for the Geosyncrhonous Imaging Fourier Transform Spectrometer*, Hyperspectral Imaging and Sounding of the Environment, OSA Technical Digest series (CD) (Optical Society of America, 2007)			2007
[RD. 55]	Konstantin V. Khlopenkov and Alexander P. Trishchenko, *Implementation and Evaluation of Concurrent Gradient Search Method for Reprojection of MODIS Level 1B Imagery*, IEEE Transactions on Geoscience and Remote Sensing, Vol. 46, No. 7, July 2008			July 2008
[RD. 56]	Abdelatif Hassini et al., *Active Fire Monitoring with Level 1.5 MSG Satellite Images*, American Journal of Applied Sciences, Vol. 6 (2009)			2009

Ref.	Title	Code	Ver.	Date
[RD. 57]	Piet Stammes et al., *Verification and Validation of the GOME-2 Level-1 Cloud Product*, Proceedings of the 1st EPS/MetOp RAO Workshop, ESRIN, Frascati (Italy), May 2006			May 2006
[RD. 58]	*Tropospheric Emission Monitoring Internet Services' (TEMIS) FRESCO algorithm for retrieval of Cloud Fraction & Pressure*, http://www.temis.nl/fresco/			
[RD. 59]	André Morel, *Optical Modeling for the Upper Ocean in Relation to its Biogenous Matter Content*, Journal of Geophysics research, Vol. 93, No. C9, September 1988			September 1988
[RD. 60]	V. Billat and L. Wald, *Use of a Simulator for the Study of the Sensitivity of the Signal Sensed by the MERIS Spectrometer*, Geosciences and Remote Sensing Symposium (IGARSS), 1995			1995
[RD. 61]	Ankie J.M. Piters et al., *The Remote Sensing of Tropospheric Composition from Space – Chapter 7: Data Quality and Validationof Satellite Measurements of Tropospheric Composition*, Springer Verlag, ISBN-978-3-642-14790-6, February 2011	ISBN-978-3-642-14790-6		February 2011
[RD. 62]	*Committee on Earth Observation Satellites (CEOS)*, http://www.ceos.org/			
[RD. 63]	F. Montagner, *End-to-end simulations of the MERIS instrument*, Proceedings of Oceanology International, Brigthon (United Kingdom), March 1994			March 1994
[RD. 64]	S. Devadiga, *MODIS Land Product Quality Assessment*, 610.2/614.5 Branch Meeting, March 2011			March 2011
[RD. 65]	*Resourcesat-1 Data Quality Evaluation System*, S.S. Paluse et al., Data Quality Evaluation Division, RESIPA, ISRO. Ahmedabad (India), April 2007			April 2007
[RD. 66]	*EO Portal webpage*, http://www.eoportal.org/			
[RD. 67]	*Terra Mission – EO Portal webpage*, https://directory.eoportal.org/web/eoportal/satellite-missions/t/terra			
[RD. 68]	Daniel S. Tkacik et al., *Atmospheric correction for MASTER image data using localized modelled and observed meteorology and trace gases*, Remote Sensing Letters, Vol. 3, No. 3, May 2011			May 2011
[RD. 69]	L. Guanter, L. Alonso and J. Moreno, *Atmospheric Correction of CHRIS/PROBA Data Acquired in the SPARC Campaign*, Proc. of the 2nd CHRIS/Proba Workshop, ESA/ESRIN, Frascaty (Italy), April 2004			April 2004
[RD. 70]	Ji Zhou et al., *Atmospheric correction of PROBA/CHRIS data in an urban environment*, International Journal of Remote Sensing, 32:9			
[RD. 71]	Naveen R. Shahi et al., *Atmospheric correction for sea surface temperature retrieval from single thermal channel Radiometer data onboard Kalpana satellite*, J. Earth Sys. Sci. Vol. 120, No. 3, June 2011			June 2011
[RD. 72]	P.S. Chavez Jr., *An improved dark-object subtraction technique for atmospheric scattering correction of multi-spectral data*, Remote Sensing of Environment, Vol. 24 (1988)			1988
[RD. 73]	P.S. Chavez Jr., *Image-based atmospheric correction: revised and improved*, Photogrammetric Engineering and Remote Sensing, Vol 62 (1996)			1996
[RD. 74]	Zhao Xiang et al., *Improvement of the Dark Object Method in Atmospheric Correction of hyperspectral remotely sensed data*, Science in China Series D: Earth Sciences, Vol.51, No. 3			
[RD. 75]	Geoffrey M. Smith & Edward J. Milton, *The use of the empirical line method to calibrate remotely sensed data to reflectance*, International Journal of Remote Sensing, 20:13 (1999)			1999
[RD. 76]	S. Adler-Golden et al. *FLAASH, a MODTRAN4 Atmospheric Correction Package for Hyperspectral Data Retrieval and Simulations*, (2008), ftp://popo.jpl.nasa.gov/pub/docs/workshops/98_docs/2.pdf			2008
[RD. 77]	Luis Gómez-Chova et al., *New Cloud Detection Algorithm for Multispectral and Hyperspectral Images: Application to ENVISAT/MERIS and PROBA/CHRIS Sensors,* (2006)			2006

Ref.	Title	Code	Ver.	Date
[RD. 78]	Luis Gómez-Chova et al., *Cloud Probability Mask for PROBA/CHRIS Hyperspectral Images* 4th ESA CHRIS Proba Workshop, ESA-ESRIN, September 2006			September 2006
[RD. 79]	Luis Gómez-Chova et al., *Cloud-Screening Algorithm for ENVISAT/MERIS Multispectral Images*, IEEE Transactions on Geoscience and Remote Sensing, Vol. 45, No. 12, December 2007			December 2007
[RD. 80]	G. Girouard et al., *Validated Spectral Angle Mapper Algorithm for Geological Mapping Comparative Study between Quickbird and Landsat-TM*			
[RD. 81]	Roberta H. Yuhas et al., *Discrimination among semi-arid landscape endmembers using the spectral angle mapper (SAM) algorithm*, Summaries of the 4th JPL Airborne Earth Science Workshop, JPL Publication 92-41			
[RD. 82]	Peg Shippert, *Introduction to Hyperspectral Image Analysis*, Online Journal of Space Comunication (2003)			2003
[RD. 83]	H.G. Lewis, *A generalized confusion matrix for assessing area estimates from remotely sensed data*, Int. J. Remote Sensing, Vol. 22, No. 16 (2002)			2002
[RD. 84]	*Glossary of Remote Sensing Terms – Canada Center for Remote Sensing*, http://cct.rncan.gc.ca/glossary/index_e.php?id=3124			
[RD. 85]	*ESA's BEAM Toolbox webpage*, http://www.brockmann-consult.de/cms/web/beam/welcome			
[RD. 86]	Jochem Verrelst, *BEAM practical lecture: Applications for Ocean Mapping*, Laboratorio de Procesado de Imágenes (Universidad de Valencia), May 2005			May 2005
[RD. 87]	Ismail Avcibas, *Image quality statistics and their use in steganalysis and compression (PhD thesis)*, Bogazici University (2001)			2001
[RD. 88]	Ismail Avcibas, *Statistical evaluation of image quality measures*, Journal of Electronic Imagin 11(2), April 2002			April 2002
[RD. 89]	K. Segl, L. Guanter and H. Kaufmann, *Simulation-based case studies for optimization of sensor design and image-based parameter retrieval using the EnMap Scene simulator*, Proc. Hyperspctral 2010 Workshop, Frascati (Italy), March 2010			March 2010
[RD. 90]	*ONTAR's MODTRAN official webpage*, http://ontar.com/Software/ProductDetails.aspx?item=modtran			
[RD. 91]	*The Earth Observation Handbook - Climate Change Special Edition: Updated with 2010 data*, CEOS 2010. www.eohandbook.com			2010
[RD. 92]	*USGS Lunar Calibration webpage*, http://www.moon-cal.org/			
[RD. 93]	Luis Guanter, Luis Alonso and José Moreno, *A Method for Surface Reflectance Retrieval From PROBA/CHRIS Data Over Land: Application to ESA SPARC Campaigns*, IEEE Transactions on Geosciences and Remote Sensing, Vol. 43, No. 12, December 2005			December 2005
[RD. 94]	L. Guanter, M. Del Carmen González-Sanpedro & J. Moreno (2007): *A method for the atmospheric correction of ENVISAT/MERIS data over land targets*, International Journal of Remote Sensing, 28:3-4, 709-728			2007
[RD. 95]	Camps, A.; Corbella, I.; Vall-llossera, M.; Duffo, N.; Marcos, F.; Martinez-Fadrique, F.; Greiner, M.; "The SMOS end-to-end performance simulator: description and scientific applications," 2003 IEEE International Geoscience and Remote Sensing Symposium, IGARSS '03. Proceedings, Vol.1, pp. 13- 15, Toulouse, France, 21-25 July 2003			July 2003
[RD. 96]	Recommendation ITU-R P.372-10 (10/2009), "Radio noise", pp. 16-23, http://www.itu.int/dms_pubrec/itu-r/rec/p/R-REC-P.372-10-200910-I!!PDF-E.pdf			October 2009
[RD. 97]	Microwave Propagation Toolbox for Planetary Atmospheres, http://www.sat.ltu.se/projects/esa_planetary/ implementing the "ARTS, the atmospheric radiative transfer simulator," Version 2, by Eriksson, P., S. A. Buehler, C. P. Davis, C. Emde, and O. Lemke (2011), J. Quant. Spectrosc. Radiat. Transfer, doi:10.1016/j.jqsrt.2011.03.001			2011

Ref.	Title	Code	Ver.	Date
[RD. 98]	Ruf C.S., "Detection of calibration drifts in spaceborne microwave radiometers using a vicarious cold reference," IEEE Transactions on Geoscience and Remote Sensing, Vol. 38(1), pp. 44-52, 2000			2000
[RD. 99]	Chukhlantsev, A., "Microwave Radiometry of Vegetation Canopies," ed. Springer, 2006, XIV.			2006
[RD. 100]	Smith, D.F., Weber, B.J., and Gasiewski, A.J., "An Anisotropic Ocean Surface Emissivity Model Based on WindSat Polarimetric Brightness Observations," Microwave Radiometry and Remote Sensing of the Environment, 2008. MICRORAD 2008, Doi: 10.1109/MICRAD.2008.4579491			2008
[RD. 101]	Array Systems Computing Inc., Algorithm Theoretical Basis Document (ATBD) for the SMOS Level 2 Soil Moisture Processor Development Continuation Project, ESA No.: SO-TN-ARR-L2PP-0037, Issue: 3.3, June 16, 2010			June 2010
[RD. 102]	J. Tenerelli, "Impact of Celestial Sky Noise on SMOS Brightness temperatures," http://www.boost-technologies.com/jtenerelli/smos/galactic/smos_galactic.pdf, October 6, 2010			October 2010
[RD. 103]	GISM GLOBAL IONOSPHERIC SCINTILLATION MODEL: http://www.itu.int/oth/R0A04000019/en			
[RD. 104]	Le Vine, D.M. Abraham, S. "The effect of the ionosphere on remote sensing of sea surface salinity from space: absorption and emission at L band," IEEE Transactions on Geoscience and Remote Sensing, Vol. 40(4), pp. 771 – 782, April 2002			April 2002
[RD. 105]	International Reference Ionosphere IRI-2011 Fortran source code: http://nssdcftp.gsfc.nasa.gov/models/ionospheric/iri/			2011
[RD. 106]	International Geomagnetic Reference Field: http://www.ngdc.noaa.gov/IAGA/vmod/igrf.html			
[RD. 107]	Schiavon, G., Ferrazzoli, P., Solimini, D., de Maagt P., and Poiares Baptista I.P.V., "A global high-resolution microwave emission model for the Earth," Radio Science, Volume 33, Number 3, Pages 753-766, May-June 1998			June 1998
[RD. 108]	Community Microwave Emission Modelling Platform (CMEM): http://www.ecmwf.int/research/ESA_projects/SMOS/cmem/cmem_index.html			
[RD. 109]	J. Randa et al., "NIST Technical Note 1551: Recommended Terminology for Microwave Radiometry," http://www.nist.gov/pml/electromagnetics/rf_electronics/upload/NIST-TN-1551.pdf, August 2008			August 2008
[RD. 110]	Camps, A.; Tarongí, J.M., "Microwave Radiometer Resolution Optimization Using Variable Observation Times," *Remote Sens.* 2010, Vol. *2*, pp. 1826-1843, http://www.mdpi.com/2072-4292/2/7/1826/			2010
[RD. 111]	Camps, A., "Noise wave analysis of Dicke and noise injection radiometers: Complete S parameter analysis and effect of temperature gradients," *Radio Science*, 2010, Vol. 45, RS5004, doi:10.1029/2009RS004198.			2010
[RD. 112]	[RD.19] Bettenhausen, M., Gaiser, P., "Lessons from WindSat Cal. / Val. Activities," NOAA CDR Workshop, 2010-03-23, http://www.star.nesdis.noaa.gov/star/documents/meetings/CDR2010/talks/DayTwo/Bettenhausen_M.pdf			March 2010
[RD. 113]	DMSP SSM/I-SSMIS Daily Polar Gridded Brightness Temperatures, http://nsidc.org/data/docs/daac/nsidc0001_ssmi_tbs.gd.html			
[RD. 114]	Martin-Neira, M.; Corbella, I.; Torres, F.; Cabot, F.; Closa, J.; Kainulainen, J.; Castro, R.; Barbosa, J.; Gutierrez, A.; Martin-Porqueras, F.; Oliva, R.; Anterrieu, E.; Brown, M.; McMulan, K.; , "Smos payload performance assessment," Geoscience and Remote Sensing Symposium (IGARSS), 2010 IEEE International , vol., no., pp.3150-3153, 25-30 July 2010; http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5649704&isnumber=5648802			July 2010
[RD. 115]	Camps, A., Bará, J., Corbella, I., and Torres, F., "The Processing of Hexagonally Sampled Signals with Standard Rectangular Techniques: Application to 2–D Large Aperture Synthesis Interferometric Radiometers," IEEE Transactions of Geoscience and Remote Sensing, Vol. 35, N. 1, January 1997, pp. 183-190.			January 1997

Ref.	Title	Code	Ver.	Date
[RD. 116]	Young and Vaughan, 2005, "Extinction" Algorithm Theoretical Basis Document, version 2.10			2005
[RD. 117]	ADM-Aeolus assessment report	ESA SP-1311		2008
[RD. 118]	WALES assessment report	ESA SP-1279(3)		2004
[RD. 119]	A-SCOPE Assessment report	ESA-SP 1313-1		2008
[RD. 120]	Berthier, Danovan, van Zadelhoff, ATLAS (ATLID algorithms and Level 2 aspects) ATBD (2011)			2011
[RD. 121]	S. Gleason, S. Hodgart, Yiping Sun, C. Gommenginger, S. Mackin, M. Adjra, and. Unwin, "Detection and Processing of bistatically reflected GPS signals from low Earth orbit for the purpose of ocean remote sensing", *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 43, no. 6, pp. 1229–1241, 2005.	Gnss-r 1		2005
[RD. 122]	S. Gleason, "Remote sensing of ocean, ice and land surfaces using bistatically scattered GNSS signals from low earth orbit", *Ph. Thesis from University of Surrey*, 2006.			2006
[RD. 123]	V. U. Zavorotny and A. G. Voronovich, "Scattering of GPS signals from the ocean with wind remote sensing application", *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 38, no. 2, pp. 951–964, 2000.			2000
[RD. 124]	D. Masters, P. Axelrad and S. Katzberg, "Initial results of land-reflected GPS bistatic radar measurements in SMEX02", *Remote Sensing of Environment*, vol. 92, no. 4, pp. 507–520, 9 2004.			2004
[RD. 125]	S. Gleason, "Towards Sea Ice Remote Sensing with Space Detected GPS Signals: Demonstration of Technical Feasibility and Initial Consistency Check Using Low Resolution Sea Ice Information", *Remote Sensing*, vol. 2, no. 8, pp. 2017–2039, 8 2010.			2010
[RD. 126]	A. Komjathy, J. Maslanik, V. U. Zavorotny, P. Axelrad and S. J. Katzberg, "Sea ice remote sensing using surface reflected GPS signals", in *Geoscience and Remote Sensing Symposium, 2000. Proceedings. IGARSS 2000. IEEE 2000 International*, 2000, vol. 7, pp. 2855–2857 vol.7.			2000
[RD. 127]	P. Ferre-Lillo, N. Rodriguez-Alvarez, X. Bosch-Lluis, E. Valencia, J. F. Marchan-Hernandez and I. Camps, "Delay-Doppler Maps study over ocean, land and ice from space", in *Geoscience and Remote Sensing Symposium,2009 IEEE International,IGARSS 2009*, 2009, vol. 2, p. II–722–II–725.			2009
[RD. 128]	E. F. Fbra, E. Cardellach, O. Nogues-Correig, S. Oliveras, S. Ribo, J.C. Arco, A. Rius, M. Belmonte-Rivas, M. Semmling, S. Pettinato, R. Zasso, and S. D'Addio, "GNSS-R for studies of the crysophere", *presented at the Workshop on GNSS-Reflectometry*, 2010, Barcelona.			2010
[RD. 129]	M. Martín-Neira, S. D'Addio, C. Buck, N. Floury and R. Prieto-Cerdeira, "The PARIS Ocean Altimeter In-Orbit Demonstrator", *IEEE Transactions on Geoscience and Remote Sensing*, vol. 49, no. 6, pp. 2209–2237, 6 2011.			2011
[RD. 130]	ESA generic E2E Simulator Interface Control Document	PE-ID-ESA-GS-464	1.2.3	21-08-2017
[RD. 131]	Generic E2E Performance Simulator and L1/L2 Processor Requirements	PE-TN-ESA-GS-402	1.1	14-06-2016

2.3. PROJECT DOCUMENTS

The following documents are produced in the frame of this activity. They are referenced in this document in the form [PD.X]:

Table 2-3 Project documents

Ref.	Title	Code	Version	Date
[PD.1]	EO Missions and Elements Categorization	ARCHEO-E2E-TN-001	2.0	19/09/2012
[PD.2]	EO E2ES Reference Architecture	ARCHEO-E2E-TN-002	3.0	14/01/2013
[PD.3]	Generic Building Blocks Technical Specification	ARCHEO-E2E-TN-003	3.0	14/01/2013
[PD.4]	EO E2E Common Semantics and Dictionary	ARCHEO-E2E-TN-004	1.0	29/03/2012

Ref.	Title	Code	Version	Date
[PD.5]	Reference Architecture Evaluation Methods and Criteria	ARCHEO-E2E-TN-005	2.0	14/01/2013
[PD.6]	Design Development Process using a Reference Architecture	ARCHEO-E2E-TN-006	1.0	14/01/2013
[PD.7]	Reference Architecture Concept Evaluation	ARCHEO-E2E-TN-007	1.0	14/01/2013
[PD.8]	Reference Architecture Roadmap	ARCHEO-E2E-TN-008	1.0	14/01/2013

3. RATIONALE AND APPROACH

This section presents the rationale behind the definition of a Reference Architecture for E2E mission performance simulators as well as the approach followed for this definition.

End-to-end mission performance simulators for Earth Observation missions are a useful tool to assess the mission performance and support the consolidation of the technical requirements and conceptual design, as well as to allow end-users assessing the fulfillment of requirements by the mission. ESA is currently starting the development of these end-to-end simulators during the mission feasibility studies, so that if the mission is approved, the simulator will evolve into a support tool for the detailed design definition, preparation and validation of operations, data processing and higher-level mission products generation.

The purpose of these end-to-end simulators, as outlined in [RD. 1], [RD. 2] and [RD. 3], is to help in the assessment of different system implementation options, the development of retrieval algorithms at different data levels and the detailed design as well as the scientific preparation of the mission. In particular:

- Enable the generation of simulated Level-1 and Level-2 output data.
- Support the assessment of the end-to-end performance of the mission on the basis of Level-1 and Level-2 products simulated for selected test scenarios.
- Support the assessment of the impact of individual error sources on the output of an ideal system, both separately and simultaneously.
- Support the assessment of the performance of the retrieval algorithms and of their associated assumptions.

Usually, the first release of the simulator is developed as a prototype tool to support the initial performance assessment of the mission at Phase A. It is expected that the simulator will evolve to support the detailed mission design during the development phases. Therefore, the E2ES architecture has to allow a growth of the simulator along two possible directions: an extensive growth and an evolutionary growth, in order to include more effects and to achieve more accuracy in the simulator, respectively. The proposed E2ES architecture has to be able to follow the evolution of the mission along the different Phases, assuming that an increasing accuracy is required. As a consequence, the proposed architecture has to be able to include more complex modules that contain more accurate models for the effect to be taken into account. Thus, as it is shown in Figure 3-1, the basic idea is to define as first an essential architecture that contains the basic modules for the E2ES and providing the required flexibility to the architecture to grow in both the extensive and evolutionary sense in order to include more modules as well as different modules that implements different models.

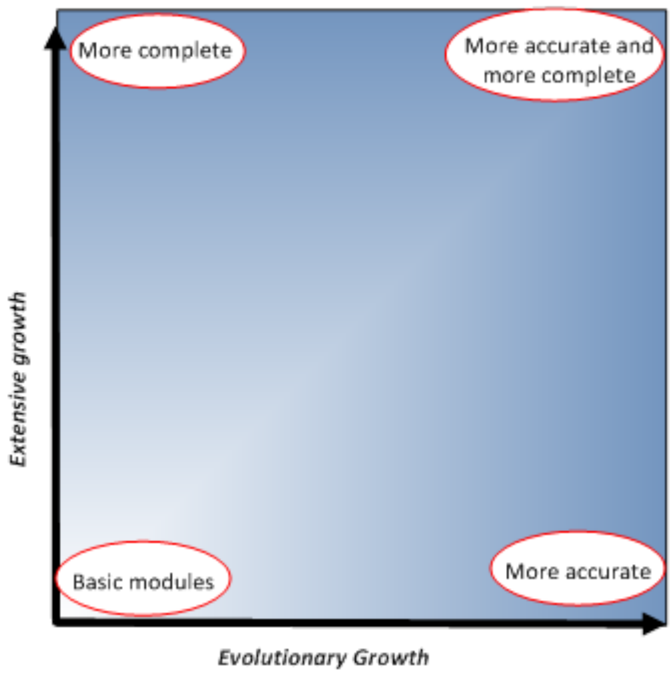


Figure 3-1: Possible growth of E2ES architecture

It is worth underlining that the feasibility of the growth of the E2ES has to be supported by a proper definition of the interfaces among different modules. In fact, the evolutionary growth is possible assuming that different modules, that implement different models, use the same interfaces so that they can be plugged in the simulator without affecting the other modules. Similarly, the extensive growth of the E2ES has to be supported by the additional modules, which are not comprised in the basic set, by assuming that they do not modify the interface between the basic modules even if they are inserted in the chain modifying the data passed to a module to another one. Some examples about this approach will be given in the following.

However, at this stage, the evolution of the design and the processing algorithms may require modifying or even replacing the components of the original simulator, what usually translates into a complex and costly reengineering process. The concept of a Reference Architecture for Earth Observation end-to-end mission performance simulators has the goal of minimizing the tasks related to the architectural design of the simulator by promoting reuse in the development. The scope of this reuse and its implications are dealt with throughout this document.

Thus, the main objective of this project is to define a Reference Architecture applicable for E2E simulators of EO missions in phases A/B1. The objective of the E2ES is to reproduce all the significant processes and steps that impact the mission performance and get simulated final data products able to quantify the mission performance of a given scenario and under certain system assumptions.

The initial steps in the definition of the Reference Architecture have been carried out and presented in [PD.1], by categorizing past, current and planned Earth Observation missions to identify the main elements affecting mission performance and having an impact over the simulator architecture. Then, the present document identifies the architecture elements required to model the mission depending on the type of mission and instrument, and proposes a generic Reference Architecture that could be adapted for the different mission particularities.

The methodology followed for deriving the Reference Architecture presented in this document has followed a bottom-up approach. For each of the main categories of instruments identified, a baseline architecture has been derived following the indications provided in Section 4. Then, the proposed architectures have been analysed with the goal of exploiting the commonalities among them. It has to be noted that, while there may be different options for the implementation of the end-to-end simulators of the different instruments, the priority has been to follow those solutions that may exploit commonalities with other types of instruments.

4. REFERENCE ARCHITECTURE FOR E2E SIMULATORS

This section presents the Reference Architecture itself together with the specific considerations to be taken into account in the reference architecture for special types of missions. This section also describes the methodology used to describe this Reference Architecture.

The reference architecture should cope with all the categories of EO missions and instruments identified in [PD.1]. Moreover, the characteristics of a framework supporting future simulator developments (based on the proposed Reference Architecture), shall be also taking into account, and in particular the OpenSF framework shall be considered the reference framework in which the new E2E EO simulators will be developed. This fact hasn't conditioned the definition of the Reference Architecture, but not considering at all the OpenSF characteristics would not be positive.

These two conditions, the application of the reference architecture to all types of missions/instruments and the use of OpenSF framework, support the decision of defining certain elements at high level that would be present in all simulators, independently of the category of the mission and the type of instrument. In the reference architecture concept these high-level elements are called modules, they cannot be considered exactly as building blocks. They could be identified as the simulator Stages (or even more, simulator Modules) of the OpenSF framework. Section 6.4 below shows the possible implications of the simulation framework over the E2E simulator itself.

Regarding the definition of the high-level reference architecture, one of the premises has been to keep it as simple as possible, defining very few variations with respect to the nominal solution, if possible. This allows having more coherence between the different simulators that will be implemented based on the architecture, being their reuse for other missions favored, even if they are quite different.

Thus, the approach has been to define very few high-level architectures, depending on the type of mission (multi-instrument, multi-platform…). Then, the type of instrument has impact on the second layer, when analysing the building blocks and internal architecture of the different high-level modules.

The main premises of the reference architecture, to be dealt with in detail in coming sections, are:

- 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 mission parameters, evolution of algorithms throughout the different mission phases, etc.
- Some of the high-level modules and lower-level building blocks will be generic across missions and instruments.

This concept is illustrated in Figure 4-1

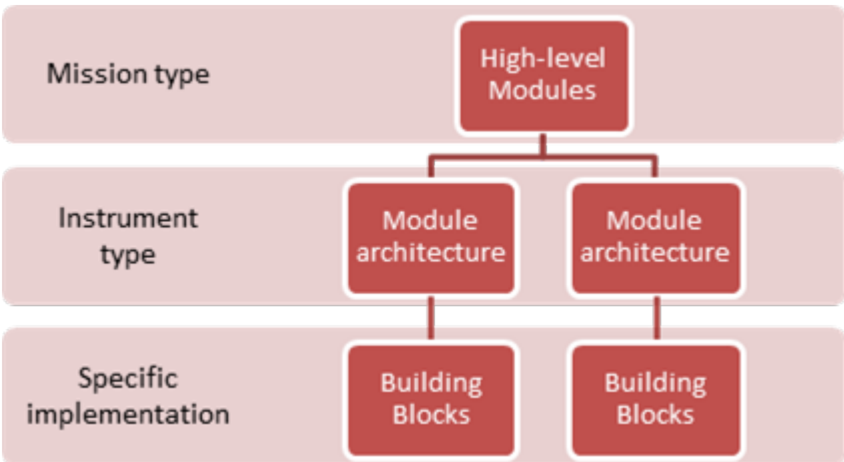


Figure 4-1: The Reference Architecture Concept

Including the high-level modules and the lower-level building blocks, the E2ES can be decomposed in three main elements:

- the Modules (or Building Blocks), that are software objects that implement the chosen models;
- the Data, that are the input/output information for the models and are exchanged among then different Modules;
- the Configuration, which has to be defined by the user depending on the simulation to be run, that can be divided in
  - Configuration parameters, that are used to configure the Modules in order to process the data under the desired conditions (i.e. instrument characteristics, data sampling, ecc.);
  - Activation flags, which are used to enable/disable the execution of a subset of models or to select the algorithm to be adopted when the E2ES is run. These activation flags can also be used to select a particular implementation of the building block when it is shared by different types of instruments.

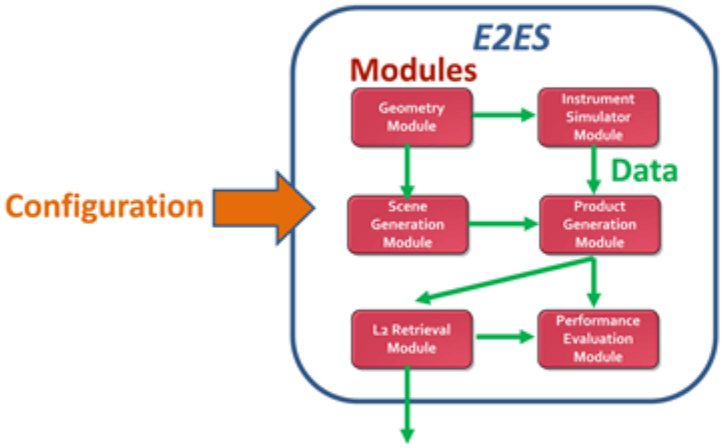


Figure 4-2: Schematic view of the elements in the E2ES

Finally, please note that the following nomenclature applies:

- Module: Top-level building blocks and higher-level elements in the Reference Architecture, associated to a specific functionality. The top-level modules identified in the frame of this study are:
  - Geometry module
  - Scene generator module
  - Instrument module
  - Level-1 processing module
  - Level-2 retrieval module
  - Performance evaluation module
- Building Block: Functional element in the Reference Architecture, which may be composed of lower-level building blocks.
- Block: Generic name used for the higher-level building blocks inside a Module.

4.1. HIGH-LEVEL ELEMENTS OF THE REFERENCE ARCHITECTURE

From the categorization of missions and analysis of commonalities performed in [PD.1], and also taking into account the experience of the project team in the design and implementation of E2E simulators, there are several simulation elements that are candidate to build the higher level of the reference architecture, as shown in Figure 4-3.

Figure 4-3: High-Level Modules in the Reference Architecture

The following are the high-level modules of the proposed reference architecture:

- **Geometry Module.** In charge of simulating the SC orbit and attitude, as well as the generation of the observation geometry of each instrument.
- **Scene Generator Module.** In charge of simulating the scene to be observed (terrain, ocean or atmosphere) and all environmental effects (radiative transfer models, atmosphere simulation, illumination conditions...) to be considered for the correct generation of the stimuli to be entered to the instrument model.
- **Instrument Module.** In charge of simulating the sensor behaviour, having different outputs depending on the type of instrument.
- **Level-1 Processing Module.** In charge of the generation of level-1 products, from level-1a to level-1c.
- **Level-2 Retrieval Module.** In charge of performing the retrieval of the geophysical parameters that are objective of the mission/instrument. Depending on the mission and on its definition of the products, this module would generate level-2 data or products at higher level of processing.
- **Performance Evaluation Module.** In charge of performing the needed analysis of the simulator outputs to evaluate the performances of the mission. It could be run at different points of the simulation chain.

Generally, the interactions between the sensor, the observed scene, its environment, and the data processing algorithms are much too complicated to be described by simple relations between the input and the output quantities of the entire system. All relevant parts have to be understood as modules of one interdependent system in order to estimate the sensitivity of the results to different input parameters and disturbing factors.

Although the data flows between these high-level modules of the reference architecture and even their order of execution could vary depending on the type of mission and instrument at which they are applied to, Figure 4-4 shows the typical generic data flow that is considered as the main Reference Architecture. In addition, Table 4-1 shows the high-level interfaces between the high-level modules. These interfaces are detailed in the specific sections devoted to each of the high-level modules further down in the document.

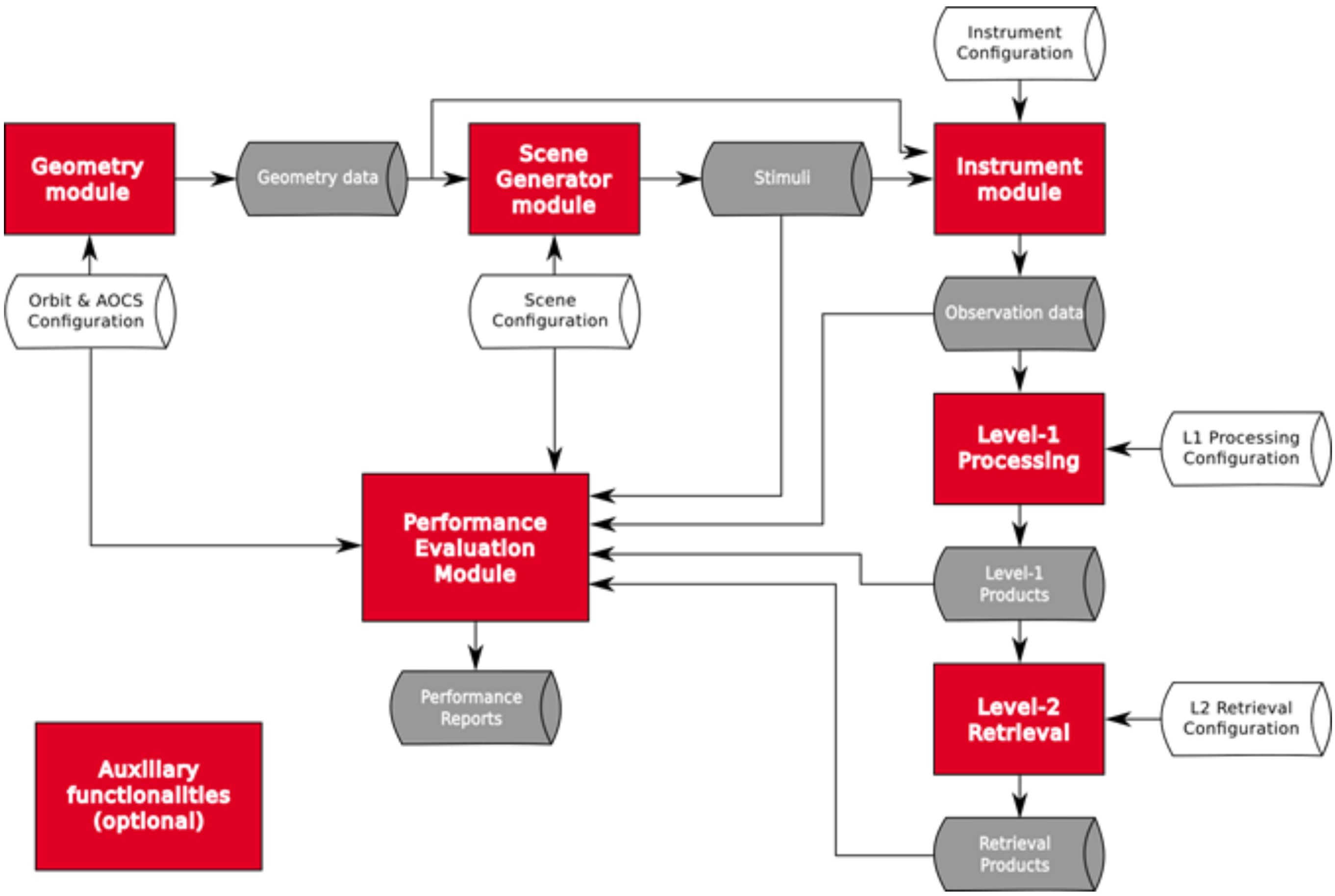


Figure 4-4: Generic data flow at the highest level of the reference architecture.

Table 4-1 High-level interfaces in the Reference Architecture

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 -(Phase A only) Estimated orbit/attitude
Level-1 Processing	Generates level-1 products, from level-1a to level-1c.	-Processing configuration	-Raw data -(Phase A only) Estimated orbit/attitude	-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

However, another possibility that could be implemented as part of the Reference Architecture concept is the one shown in Figure 4-5, in which there is a different interaction between Geometry and Scene

Generator Modules. Both figures are quite schematic, putting together in the same interface the configuration and auxiliary input files of the modules and doing the same with the output data.

In section 4.2 the different options of high-level reference architecture are analyzed in more detail, fulfilling the specific requirements coming from the mission categories that have been established in [PD.1].

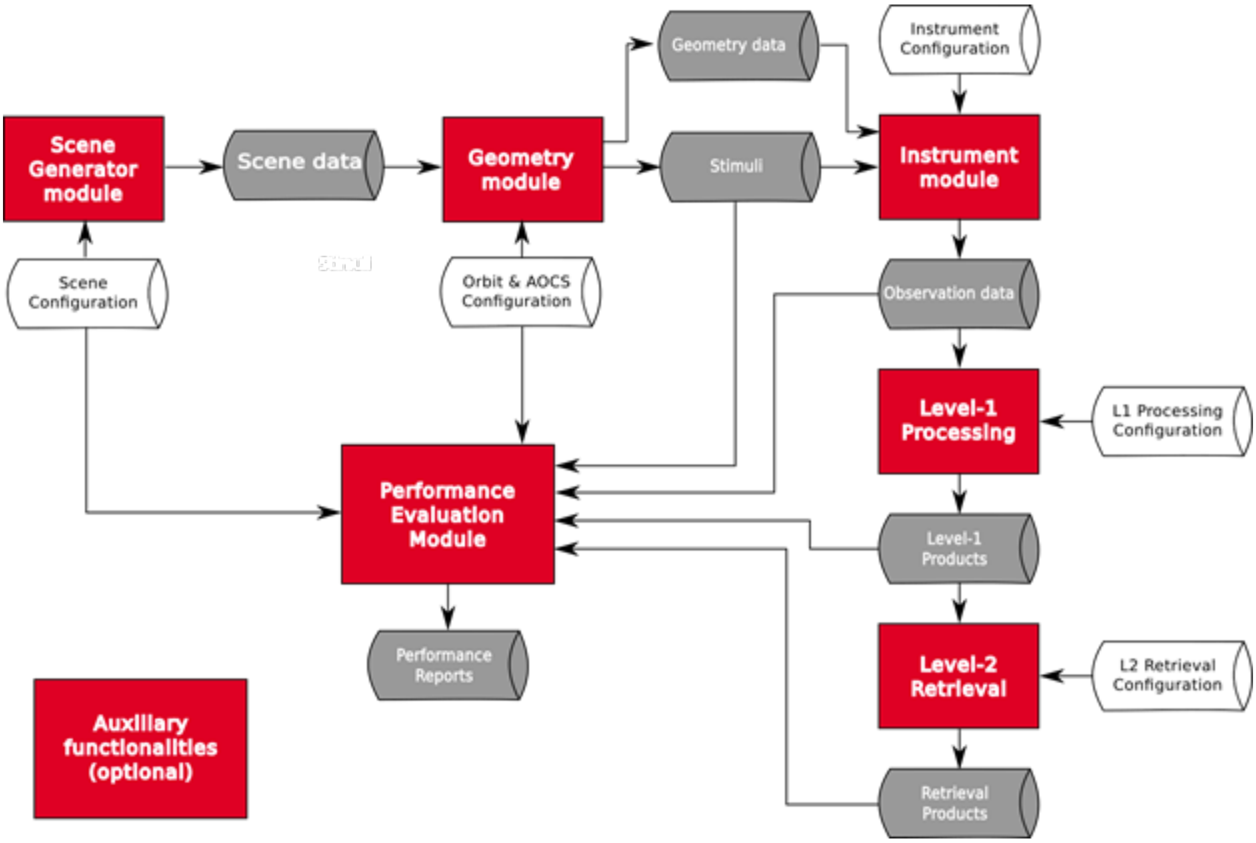


Figure 4-5: Variation of the generic data flow at the highest level of the reference architecture (Scene Generator can be run at first stage)

Although the focus of this activity are the E2E mission performance simulators for Phase A, the generic architecture presented in Figure 4-4 would also be applicable to simulators during more advanced mission phases. This is important because the E2E simulator is also intensively used in later phases (C/D) of the project to support Ground Segment activities e.g. to provide test data to verify ground processor development and interfaces.

Since these activities require that the E2E simulator generates data formatted as it will be on the real interface between satellite system and ground system (i.e. Space Source Packets), the Reference Architecture presented in Figure 4-6 foresees an additional module called "Onboard Data Generation". This module takes outputs from the Geometry Module and the Instrument Module to generate raw data in the form of auxiliary packets (e.g. orbit, attitude…), image packets and transfer frames. This module will generally not be present in the Phase A/B simulator.

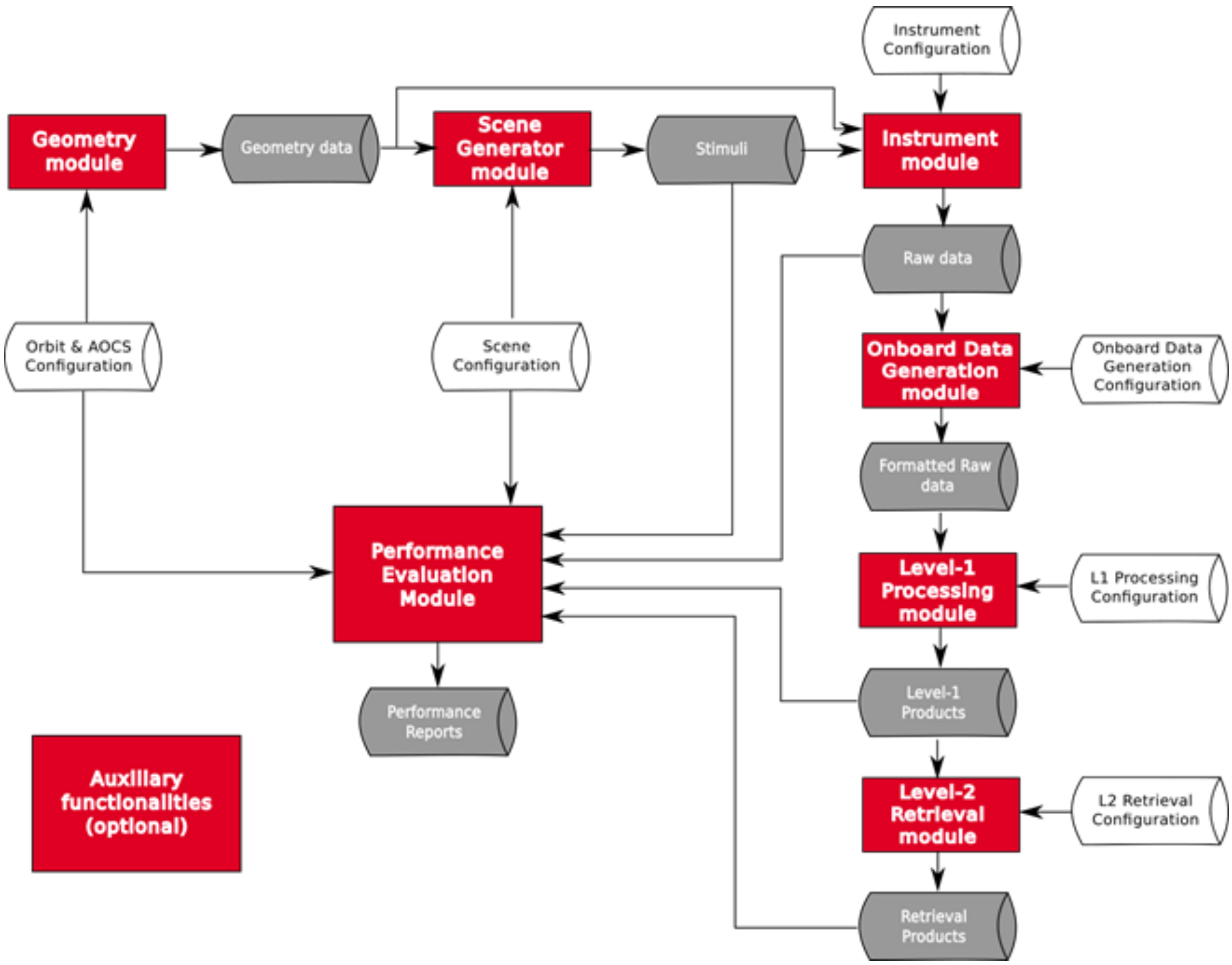


Figure 4-6: Generic data flow at the highest level of the reference architecture (including onboard data generation).

Finally, it has to be noted that the interfaces shown above are those between the high-level modules. There also exists the possibility of having inputs from files or Look-up Tables instead of using actual computations, mainly with the purpose of saving computation time. This can be contemplated at different points in the E2ES and it is dealt with in the appropriate sections of the high-level modules. For example, see Sections 8.2, 10.6.2 or 11.4.2.

4.2. 3REFERENCE ARCHITECTURE FOR SPECIFIC MISSION CATEGORIES

The most relevant mission categories identified in [PD.1] are:

- Categorization of Earth Observation Missions by number of satellites and number of instruments in the mission.
- Categorization of Earth Observation Missions by number of satellites and possible co-registration of measurements.
- Categorization of Earth Observation Missions based on scientific goals

From those categories, some important aspects to be addressed in the definition of the reference architecture have appeared:

- Implications of simulating a mission with one satellite and multiple instruments.

- Implications of simulating a mission with more than one satellite, flying in formation.
- Implications of simulating a mission/instrument which products are combined with other products from the same mission or from different mission

Moreover, there are other aspects whose impact on the architecture should be investigated, like:

- Time driven simulation versus data driven simulation.
- Use of real data from other missions (as input scenes in the Scene Generator or to be ingested at some point by the Retrieval Module)

The following sections contain the analysis of the impact in the reference architecture of the above mentioned aspects.

4.2.1. REFERENCE ARCHITECTURE FOR MULTIPLE INSTRUMENTS

In the case of a single spacecraft carrying several instruments, it is necessary to evaluate if secondary payloads shall be included in the E2E simulator. Anyway, if more than one instrument shall be included in the E2E simulator, the reference architecture shall be tailored with the aim of including several simulation chains.

The following remarks can be done:

- All chains will share most of the Geometry Module functionalities, apart from what refers to the specific instrument observation geometry (AOCS/Instrument Coupling Module).
- It is possible that part or even all the Scene Generator Module is common for both instruments.
- The Instrument and Level-1 Processing Modules will be different for the two payloads
- It could be also possible to have commonalities in the Level-2 Retrieval (to be analyzed in more detail in section 4.2.3).

Two examples of data flows for the E2E simulation of these types of missions are shown in Figure 4-7 and Figure 4-8.

Figure 4-7 assumes that the stimuli for both instruments can be fed to the corresponding Instrument Module using only one Scene Generator, and then sharing the Level-2 Retrieval Module. This module will process indistinctly the Level-1 products generated for the 2 instruments, generating at the end of the chain the same retrieval products. We consider that this specific architecture is quite realistic, because, if the two instruments can share the same Scene Generator Module, it would mean that both look to similar biophysical parameters. Therefore, the Level-2 products would be also quite similar when closing the E2E loop.

On the other hand, Figure 4-8 shows the opposite behavior. If the two instruments cannot share the Scene Generator it indicates that they probably look for quite distinct biophysical parameters (maybe also different scientific area of application) and then the Level-2 Retrieval would be necessarily different. In this case it should be considered the option of defining two different simulators for the mission, one for each instrument, replicating identical Geometry Module, except from the AOCS/Instrument Coupling building block. The reference architecture shown in Figure 4-4 would then be applicable to each separate E2E simulator.

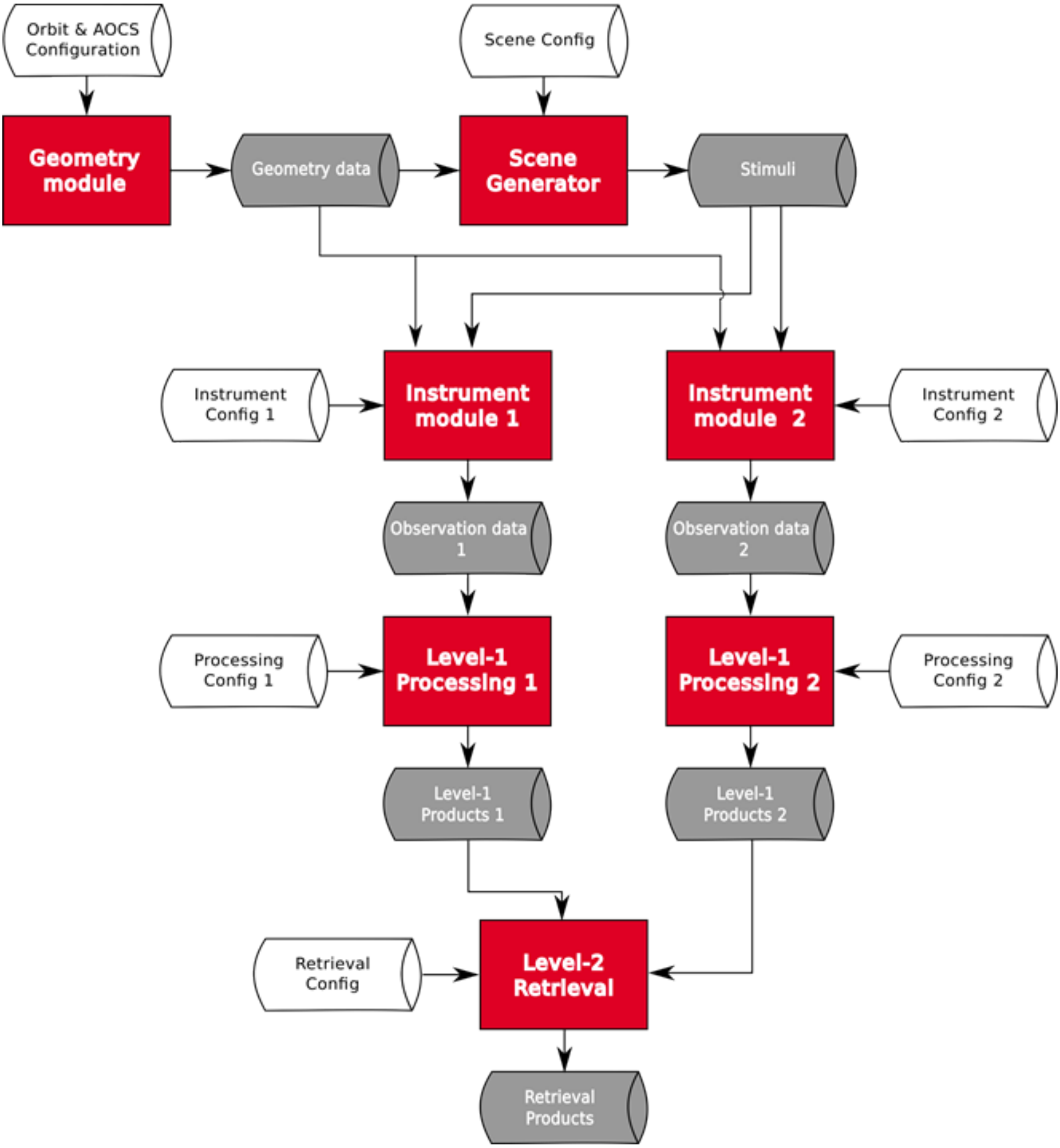


Figure 4-7: Example of generic data flow at the highest level of the reference architecture (single platform / multiple instruments, sharing scene generation and retrieval)

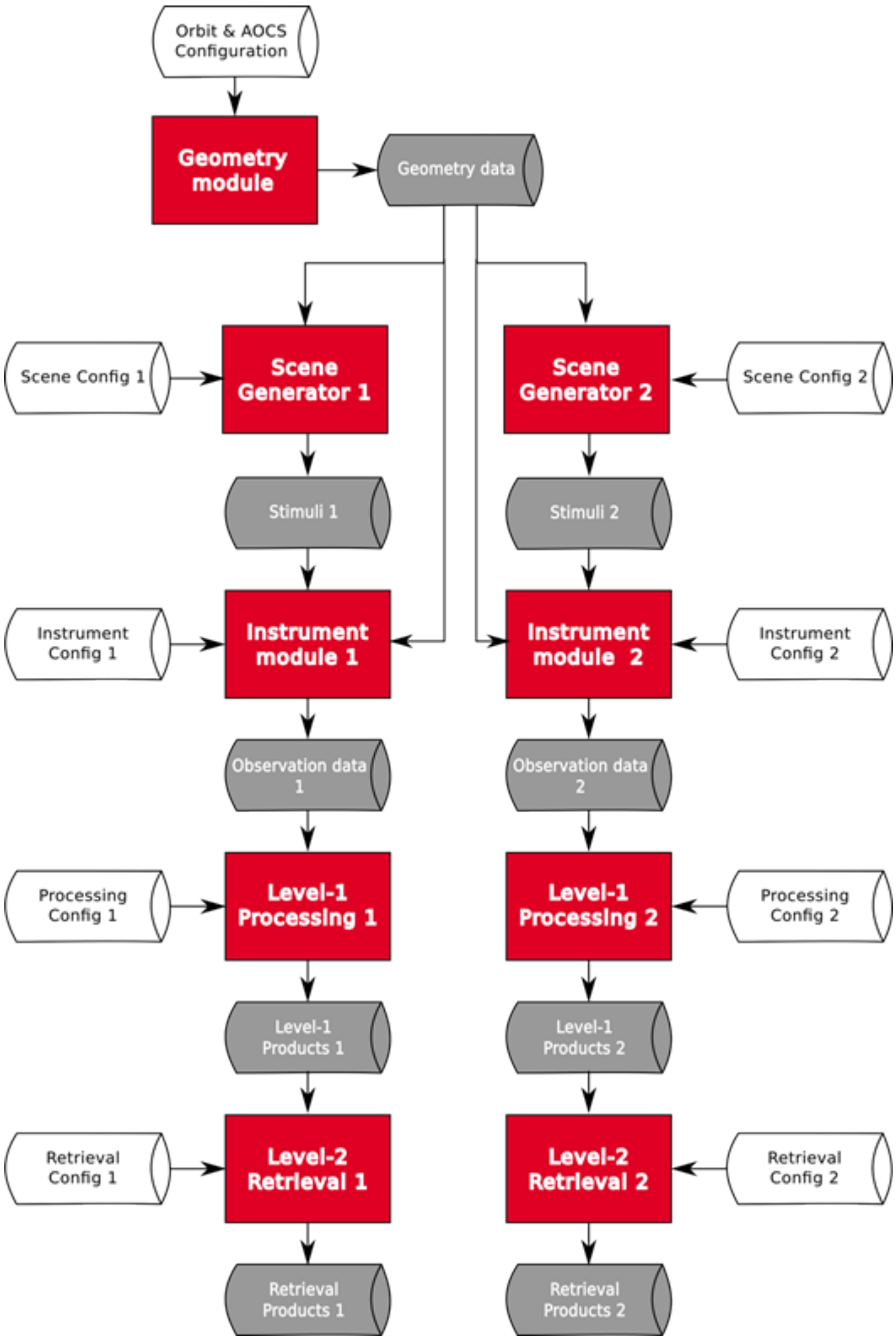


Figure 4-8: Example of generic data flow at the highest level of the reference architecture (single platform / multiple instruments, sharing only geometry simulation)

4.2.2. REFERENCE ARCHITECTURE FOR MULTIPLE SPACECRAFT

Missions like Sentinel-1 or Sentinel-2 (considering only primary payload), would fall in this category, being two identical spacecraft placed at the same orbit, but phased 180 degrees. For this type of mission, in what respects to an E2E simulator of the observing system, it can be considered as the simple case of single platform / single instrument (Figure 4-4). It has been assumed that no combination of the instrument data coming from the two satellites is performed at any point of the processing or retrieval chain (at least for simulation purposes).

In the case of having strict formation flying requirements, the Geometry Module will deal with the generation of two SC orbit and attitude profiles, most likely involving specific building blocks to simulate the formation flying. Depending on the characteristics of each formation flying mission, the high-level reference architecture would present very different simulation chains. Some options might be:

- Both instruments are identical, but fed with slightly different geometry and/or target scene. In this case, the nominal reference architecture should be applicable (Figure 4-4). Only the interfaces would slightly vary, because the data flow between modules will correspond to the data coming from two or more spacecraft instead of from only one.
- Different instruments in each spacecraft, but pointing to the same area with different geometry (Metop and PREMIER). It may happen that the formation is performed with a spacecraft from a different mission. In that case, as the instrument simulation from the other mission will not be incorporated to the E2E, the problem is then how to combine the products from it. If the formation is performed with an spacecraft of the same mission, it shall be evaluated at which point of the processing the data coming from the two spacecraft is put together. This would be probably the most complicated case to be handled from an architectural point of view. Anyway, the issue of combining data from different instruments is analysed in more detail in section 4.2.3.

4.2.3. REFERENCE ARCHITECTURE FOR COMBINATION OF PRODUCTS

This specific question has been already raised in the former sections. Independently of the formation flying condition, the problem can be simplified in two scenarios:

- Combination of products coming from a different instrument of the same mission.
- Combination of products coming from a different instrument of a different mission.

Our assumption is that, up to level-1, the data coming from different instruments is not going to be mixed. Therefore, the Level-2 Retrieval Module would be the one responsible to deal with this data combination in order to produce integrated retrieval products.

Based on that assumption, if the data are provided by another instrument from the same mission, the architecture proposed in Figure 4-7 would be valid. If the data come from other mission, they will be considered as external outputs, as they are not produced in the frame of the E2E Simulator. In this case, there will be no impact on the architecture, as the Level-2 Retrieval Module will internally absorb the impact of considering these data. Only an additional interface should be taken into account in that module. The reference architecture shown in Figure 4-4 would be valid.

4.2.4. TIME DRIVEN VS. DATA DRIVEN SIMULATIONS

All the reference architectures and data flows that have been defined in the former sections consider the implicit assumption of data-driven simulations. This means that each module generates at once all the data that it is needed from it, being then executed the following module in the chain.

But there are use cases in which the simulations are driven by time steps, e.g. to simulate variable operation and calibration profile orbits of data or to provide a flexible test data generator for use within the ground segment. The complete state of the system is then generated for each single time step, being this step propagated into the following time steps. It would happen that the time-driven simulation could be applicable in part of the simulation chain (for instance, up to Instrument Module), but a data-driven simulation should be performed in the rest of the chain (Modules simulating the ground processing).

From the point of view of the high-level reference architecture, the elements and interfaces between them are perfectly applicable in both time-driven and data-driven simulations. The implications to be taken into account by the simulation framework are:

- Supporting a time-driven simulation requires more advanced use of an orchestrating framework and building awareness of the simulation time into the module.
- openSF 3.x includes an invocation mode for modules supporting time-driven orchestration out-of-the-box, while older versions did not.

In summary, for simplicity the present document focusses on data-driven simulations.

4.2.5. USE OF REAL DATA FROM OTHER MISSIONS

The use of real data from other mission could be foreseen at two stages of the E2E simulation:

- At the Scene Generation, feeding the simulation with reference scenes that should be treated to generate the adequate instrument stimuli.
- At the Level-2 Retrieval Module to be combined with the simulated products in order to provide retrieval products.

On both cases, the impact of these external data is found only at the level of interfaces (in what respects to the high-level architecture) and no changes in the reference architecture or in the main data flow are foreseen. But it shall be remarked that the use of real scenes as input to the Scene Generator may add some inconsistencies in the E2E simulation chain. The real images would be typically defined at level-1c or level-2, so they will be biased by the geometrical conditions and real biophysical parameters of the areas being observed while acquiring that real image. Reproducing in the E2E simulator exactly the same conditions will be almost impossible, so part of the E2E evaluations could then be invalid (retrieval products compared to biophysical parameters or even the level-1c products compared to the corresponding instrument stimuli).

Our recommendation is to avoid the use of real data in the Scene Generator, or at least to carefully address the implications of their use case by case. The nominal approach would be to consider synthetic images using biophysical maps, probably externally generated.

4.3. SUMMARY AND CONCLUSIONS ABOUT THE REFERENCE ARCHITECTURE

Based on the objective of reducing as much as possible the number of high-level reference architectures, only three variations of high-level architecture have been selected:

- Nominal reference architecture (Figure 4-4or Figure 4-9). It is proposed that the Geometry module will be always run in first place within the simulation chain. It would be valid for the following categories:
  - Single instrument missions, independently on the number of spacecraft and formation flying conditions.
  - Multiple instrument missions, if each instrument is simulated in a different E2ES.
- Multiple instrument, identical biophysical parameter to be analysed (Figure 4-7 or Figure 4-10). The modules of Scene Generation and Retrieval are identical for both instruments.
- Multiple instrument, different biophysical parameter to be analysed (Figure 4-8 or Figure 4-11). In this case, only the Geometry Module is shared between the two instruments.

It is important to remark that the Performance Evaluation Module has been removed from Figure 4-10 and Figure 4-11 for the sake of simplicity. However, it is taken into account in the reference architecture also in those cases, with similar interfaces to those shown in Figure 4-9.

The Reference Architecture also allows the use of external files (e.g. in the form of LUTs) to be used instead of actual computations. This doesn't have a big impact over the general Reference Architecture, and it is dealt with at module/Building Block level.

The following sections (from section 9. to 12.) contain the particularities of the reference architecture applied to the four instrument categories defined in , but having the current high-level approach as baseline. Those sections get into the details of each instrument class with much more granularity, being the baseline reference architecture the one outlined in this paragraph.

It has been also added a devoted annex to this document (section 13.) with the implications of having an E2E simulator for a mission such as GNSS-R. This type of missions is actually a bistatic radar that uses sources of opportunity coming from the navigation satellites constellation. As this mission is quite specific and difficult to include in a category, a separated reference architecture definition has been described for it by UPC.

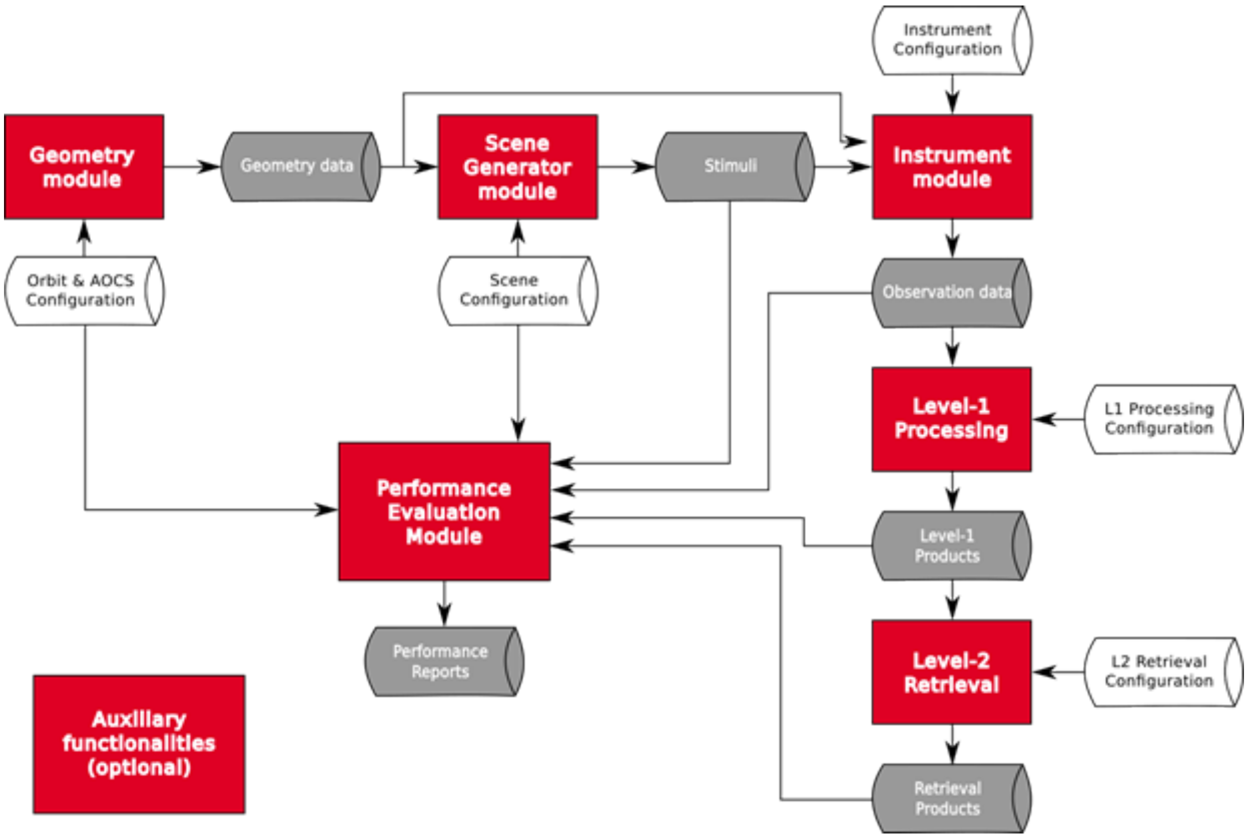


Figure 4-9: Example of generic data flow at the highest level of the reference architecture (Scene Generator needs detailed observation geometry data for its execution)

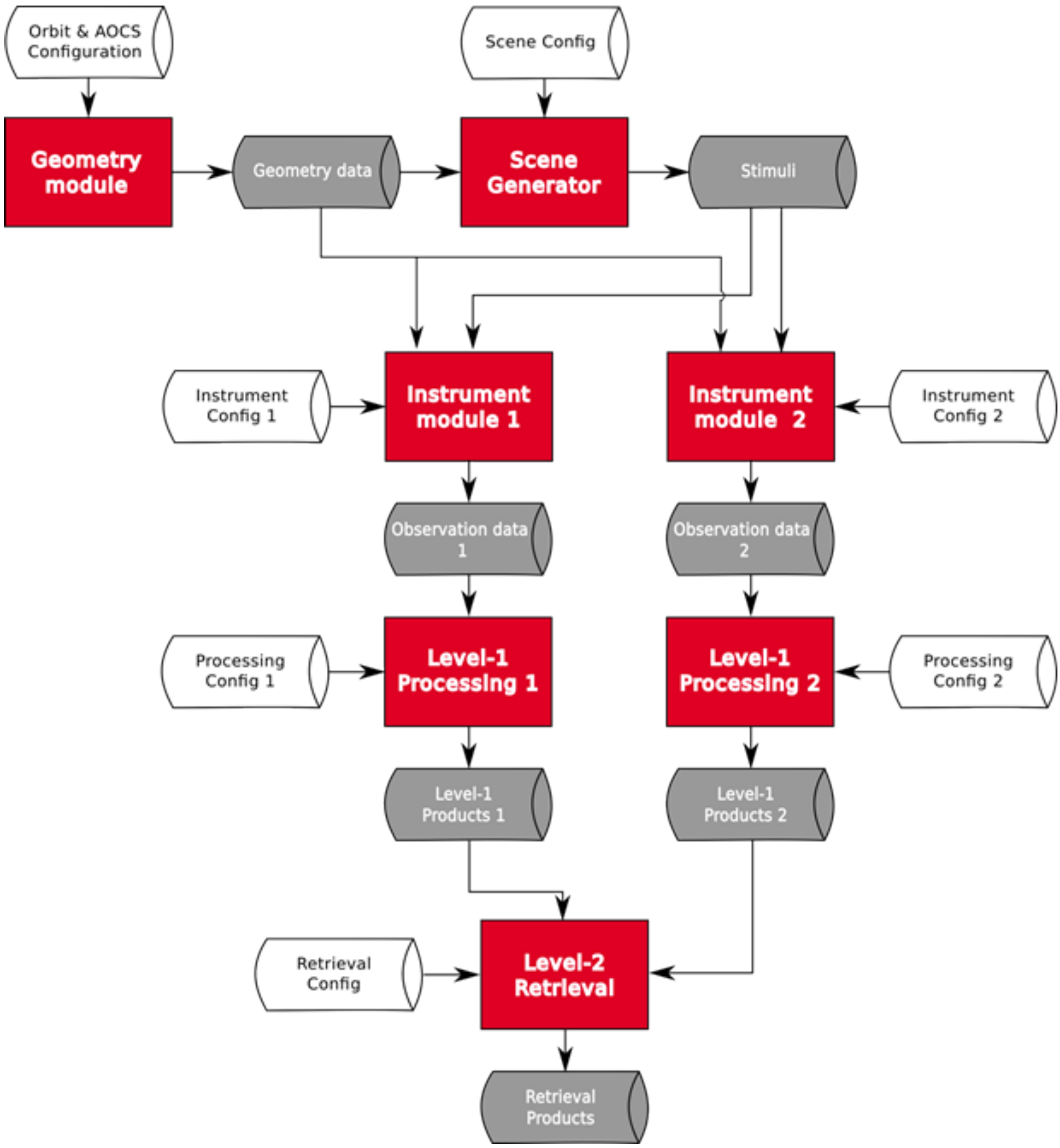


Figure 4-10: Example of generic data flow at the highest level of the reference architecture (single platform / multiple instruments, sharing scene generation and retrieval)

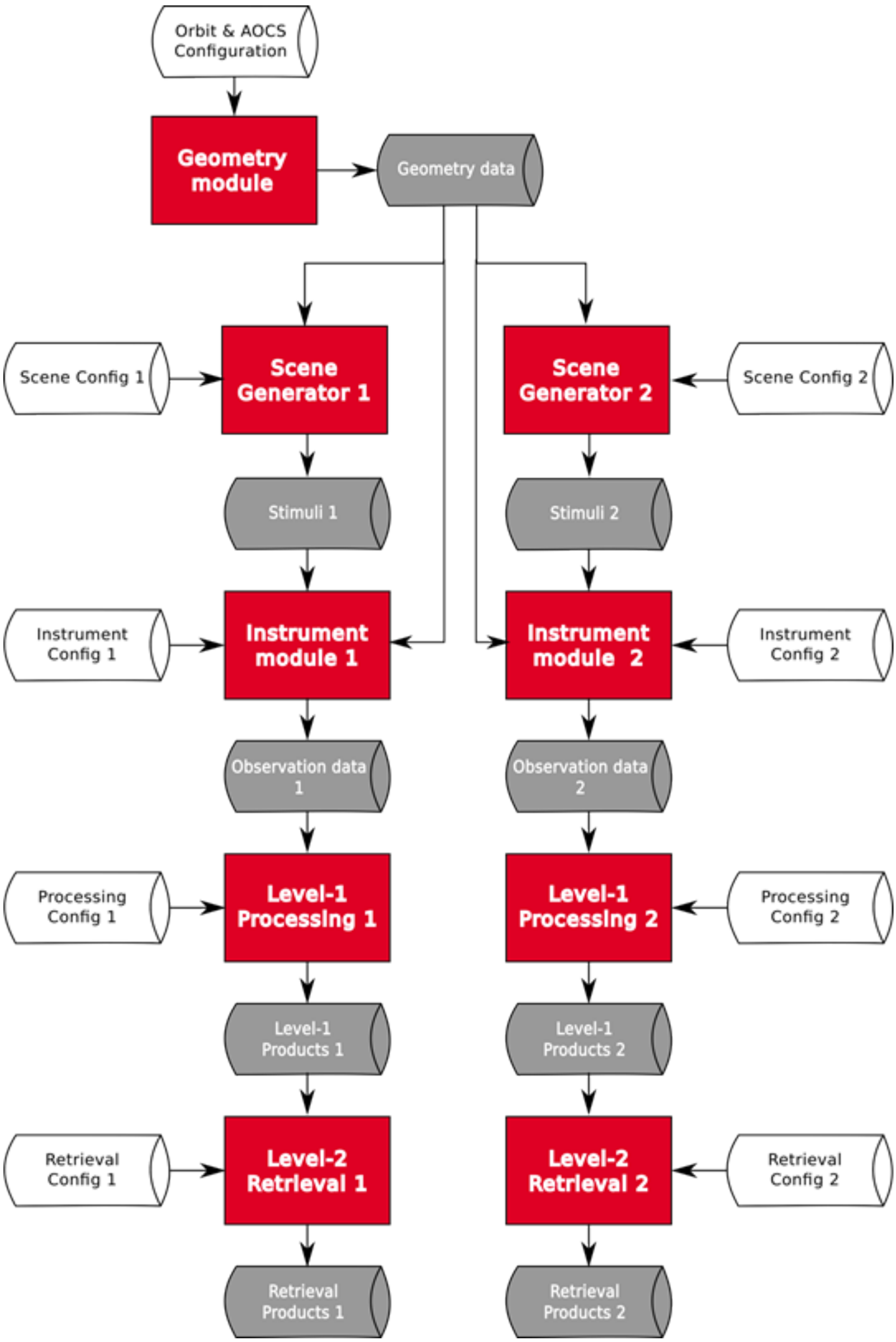


Figure 4-11: Example of generic data flow at the highest level of the reference architecture (single platform / multiple instruments, sharing only geometry simulation)

4.4. METHODOLOGY FOR DESCRIBING THE ARCHITECTURE

Figure 4-12 shows the concept behind the definition of the architecture of the end-to-end simulators. The main elements in the architecture are the building blocks, which are defined by their interfaces. The simulator will be composed of a series of top-level building blocks, also known as modules. At this level the interaction between modules will be highly dependent on the framework supporting the development of the simulator. For example, with the current infrastructure provided by OpenSF, this interaction would be provided by means of input and output files. Each of the main modules of the simulator is composed by a series of building blocks, which may at the same time be composed by additional building blocks, and so on. The level of granularity will be identified during the course of the activity and it will depend on the specific requirements to model each simulator element and on how feasible is their generalisation.

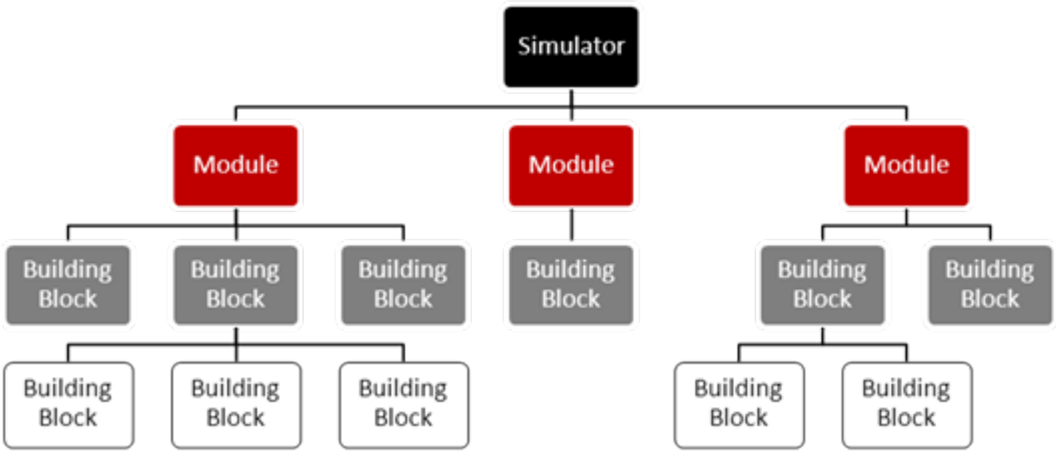


Figure 4-12: Conceptual architecture of the end-to-end simulators

The reference architecture will be described at three different levels, as shown in Figure 4-13. The structural level will describe the software components that make up the system (i.e. modules and building blocks) and the relationships among them. The behavioral level will describe the functionality of the different software components. The interactional level will describe how and when the different components interact with each other.

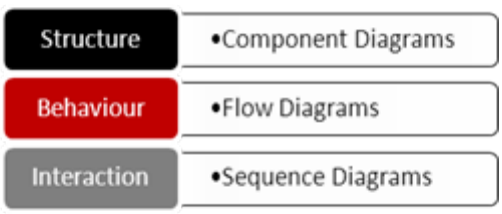


Figure 4-13: Levels of description of the Reference Architecture

It is proposed to use the Unified Modeling Language (UML) for the description of the Reference Architecture. Although the power of UML lies in the description of object-oriented software it does offer a useful and standard set of symbols to document the relationships between software modules in a way that it is not linked to the actual implementation language.

Although describing the architecture at three different layers may result in some redundancy and not strictly necessary.

4.4.1. STRUCTURE

The main elements of description of the structural layer of the Reference Architecture will be UML Component Diagrams, which will be used to describe the different modular parts of the system and the relationships among them. The following criteria will be used for the description:

- All modules and building blocks will be encapsulated as UML components, with one component being formed by a set of lower-level components.

- A component will have a number of interfaces to the rest of components. The inter-relationships shown by these interfaces will be at the level of data exchanged rather than functionalities. This is done in order to map each of the components to a building block, that will be defined by its data input and output definitions. For the sake of simplicity, the component diagram will not show the interfaces among second level components.

  The interfaces between components will be described by making use of external interfaces (to other components at the same level) and using "delegate" connectors to link the external interface to the internal component that will handle the data or realize that behaviour.

The Reference Architecture will be described, at the structural level, with as many Component Diagrams as needed, starting from the high-level modules and down through the different layers of building blocks.

Each Component Diagram will have associated a table describing the different components with the following items:

- Component name
- Functional description
- Inputs
- Outputs

Figure 4-14 and shows an example of a generic component diagram.

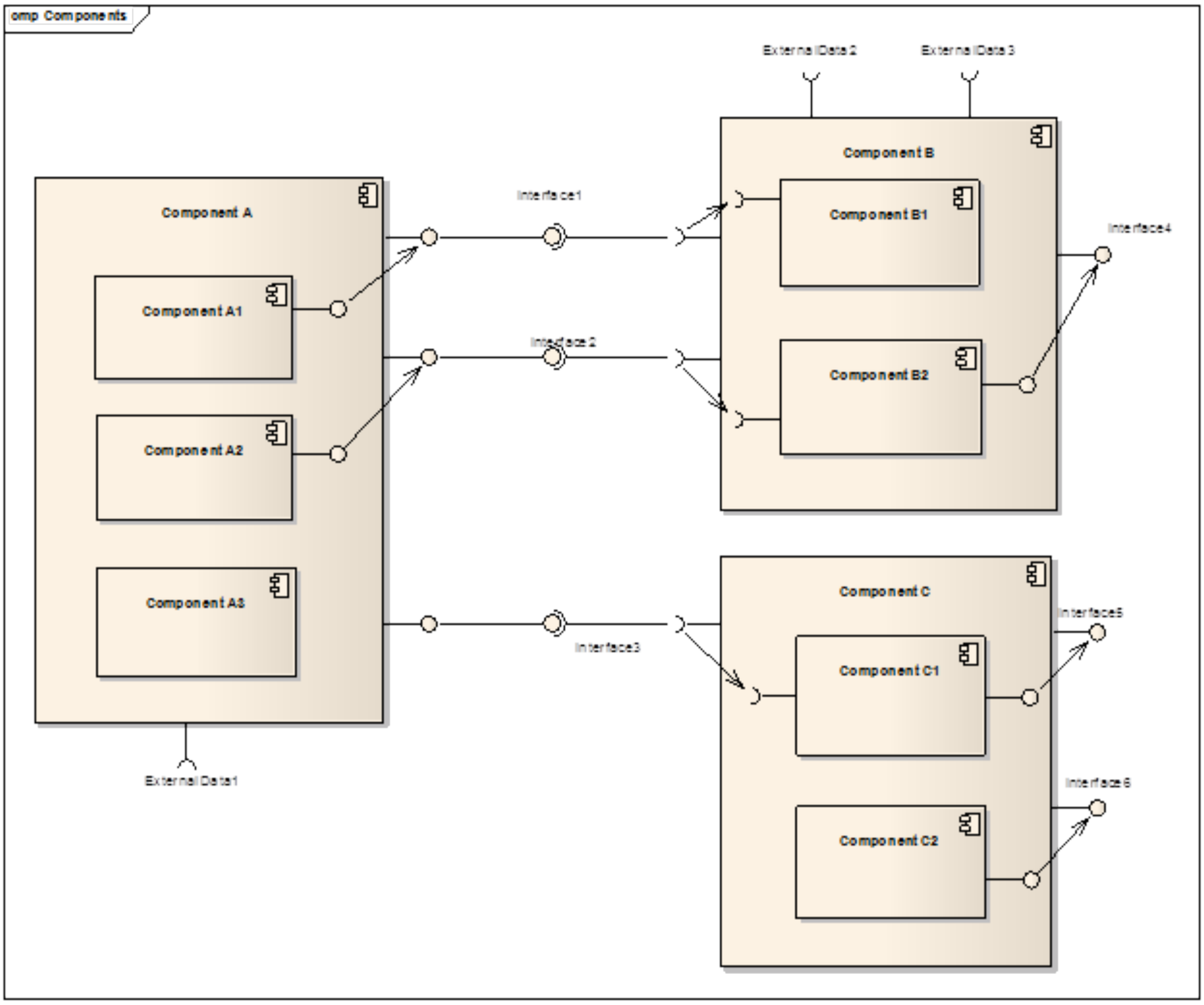


Figure 4-14: Example of component diagram to describe the reference architecture

4.4.2. BEHAVIOUR

The main element of description of the behavioral layer of the reference architecture will be standard data flow diagrams, which will be used to describe the functionality of the different software components. Rather than focusing on the structure and interfaces as the component diagram does, the data flow diagram will show the way data flows from one component to another and how it is stored. The following elements will be used for the description:

- Modules or building blocks.
- Sources, to represent input data (e.g. scientific data or configuration parameters) external to the execution of the module being represented. This will be shown at the top of the diagram.
- Data elements, to represent data that is exchanged between the elements shown in the diagram, usually building blocks in one single module.

Figure 4-15 shows an example of a data flow diagram.

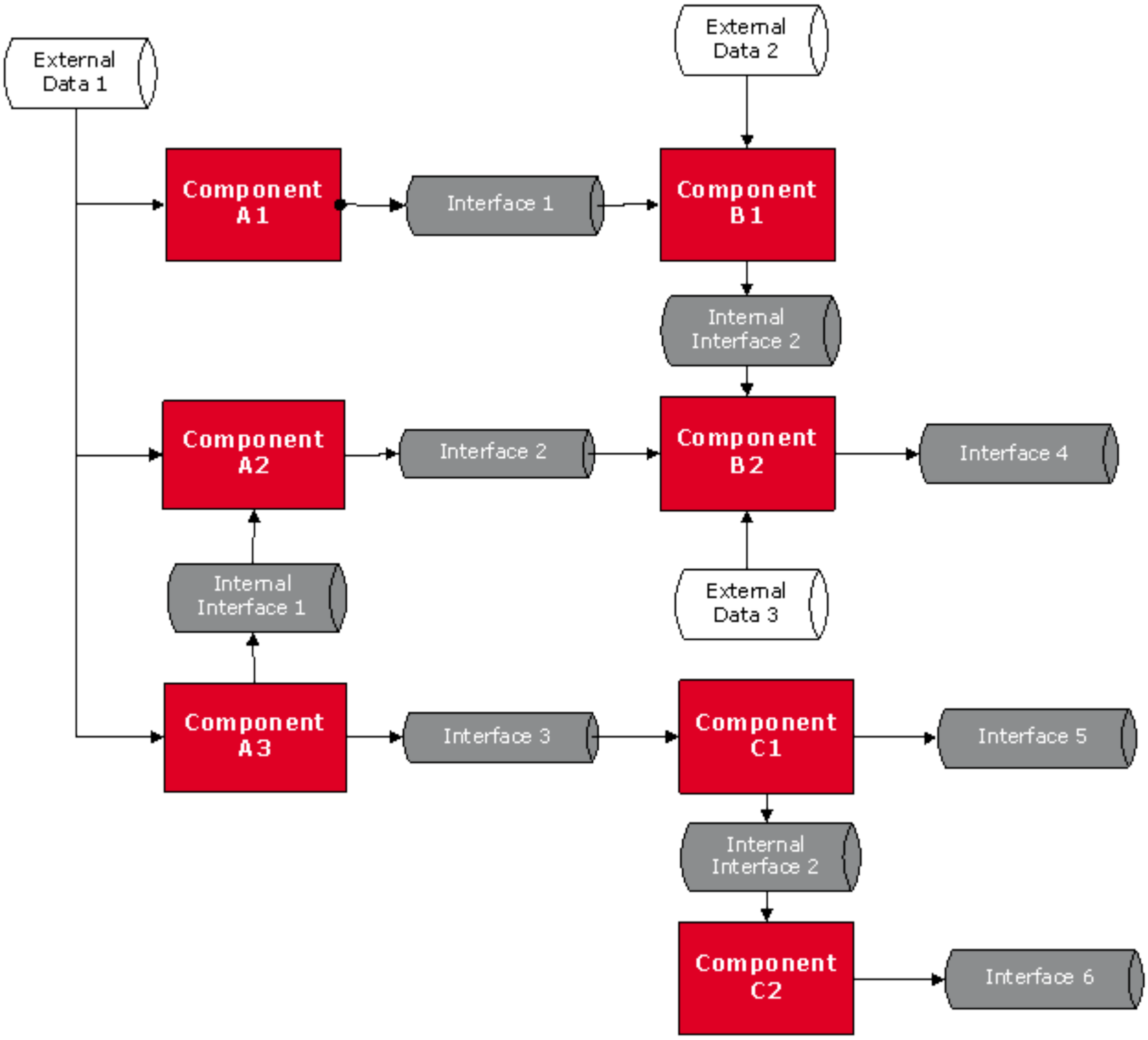


Figure 4-15: Example of flow diagram to describe the reference architecture

4.4.3. INTERACTION

The main elements of description of the interactional layer of the Reference Architecture will be UML Sequence Diagrams, which will be used to describe how and when the different components interact with each other. It is meant to complement the component and the data-flow diagrams by providing information such as the timing of execution of the different modules and whether the modules will be executed in sequence or in parallel.

In principle this last layer of description will not be used as much as the structural and behavioural layers. However it is foreseeable that there will be parts of the architecture where a sequence diagram would shed more information on the actual interactions, for example:

- To understand and derive high-level requirements for the optimum framework architecture, for example in the case of contemplating parallelisation, in which it would be important to visualise the interactions between parallel implementation of a same building block.
- To illustrate other interactions between the building blocks, like loops or iterative processes.

The following criteria will be used for the description:

- The sequence elements in the diagram will correspond to the components identified in the Structural Architecture (i.e. the component diagram).
- The interaction between components will be done through messages, which represent flow of information or transition of control between elements.

Figure 4-16 shows an example of a sequence diagram.

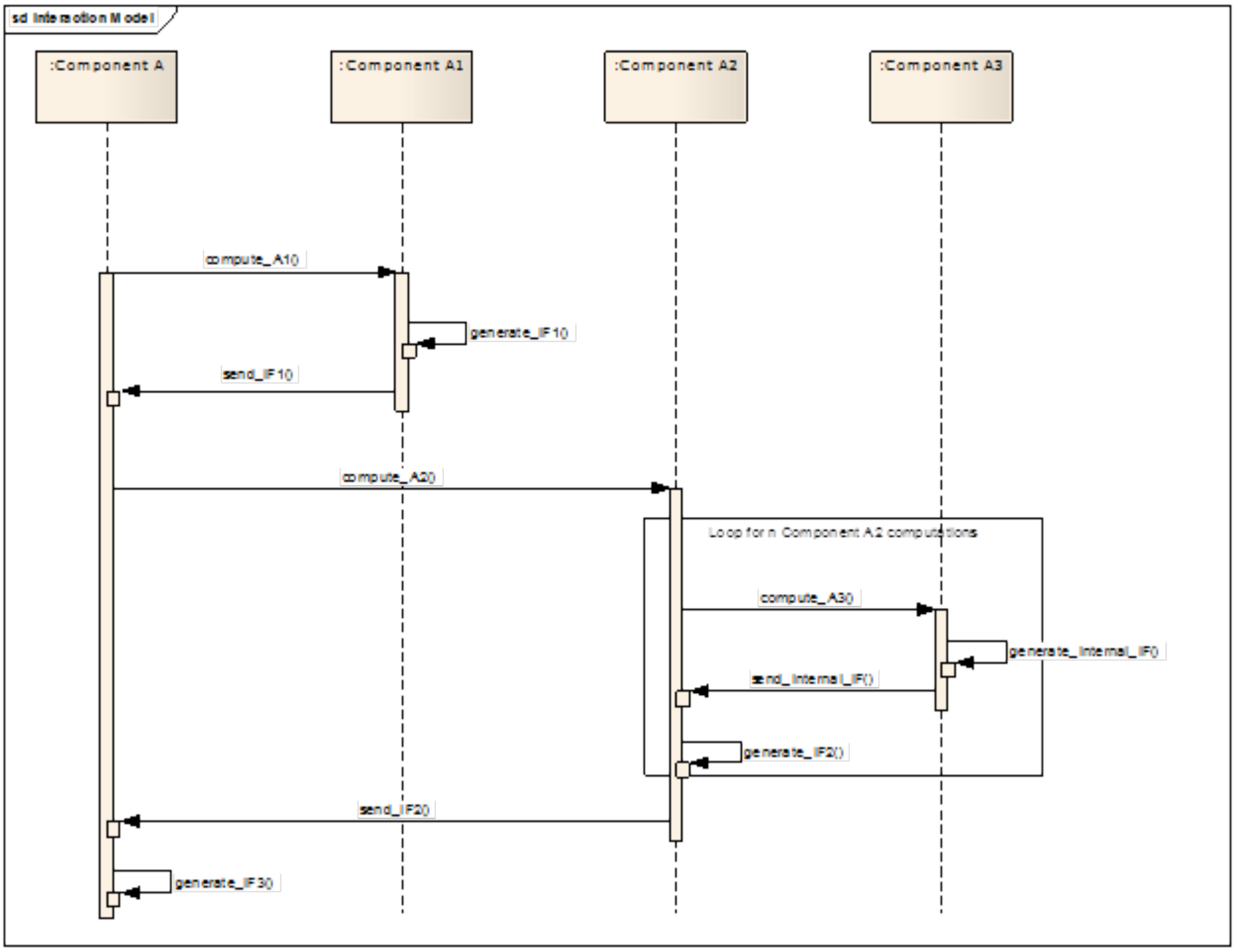


Figure 4-16: Example of sequence diagram to describe the reference architecture

5. REQUIREMENTS FOR AN E2E MISSION PERFORMANCE SIMULATOR

This section presents some generic requirements to be applied when designing an E2ES based on the concept of the reference architecture. While the specific requirements will be very much mission dependent, these requirements address some architectural and interface issues.

The definition of requirements has been done taking into consideration the requirements of the E2E Simulators of the three candidate Earth Explorer phase A missions currently (CoReH2O, PREMIER and BIOMASS), from the ARCHEO-E2E Statement of Work and the first-hand experience of the project team.

Note that the requirements below do not need be specialized to compatible with openSF, as openSF already implements them.

Note also that these requirements apply to E2ES during for Phase A, which is the focus of this activity. They are, however, generic enough to be applicable to more advanced mission phases. When required, comments on applicability to later mission phases have been included. A separate generic template for specialised for requirements applicable to phase C/D exists [RD. 131]; it is planned to merge the requirement in this document with the one contained there.

5.1. GENERAL REQUIREMENTS

<GEN-010>	The E2ES shall provide an end-to-end simulation capability (i.e. from geophysical scenarios to Level 2 data products) allowing the assessment of the fulfillment of the science goals/mission requirements specified in the mission’s requirement baseline.
<GEN-020>	The E2ES shall be able to generate simulated products at level 1and level 2 as well as other TBD intermediate results for the specified user scenarios.
<GEN-030>	The E2ES shall be able to generate all the level 1b and level 2 simulated mission products both error-free and with all or some errors applied.
<GEN-040>	The E2ES shall be able to assess the performance of the retrieval algorithms and their associates assumptions
<GEN-050>	The basic functions needed for the execution of the data exchange mechanisms, data flow control and maintenance of interfaces shall be implemented by means of the specified simulation software framework.
<GEN-060>	The implementation of the E2Es shall make use, as much as possible, of existing EO libraries or developments from other E2ES.
<GEN-070>	The implementation of the E2Es shall be done in a modular way, in such a way that foster reuse of new developments by future E2ES.
<GEN-080>	The required time CPU performance for the simulator shall be specified. This will drive the need for parallelization or not.
<GEN-090>	The calibration modeling requirements shall be specified for both geometric and radiometric requirements.
<GEN-100>	The requirements to support statistical error propagation (e.g. either standard error propagation or Montecarlo-type analyses) shall be specified, if applicable.
<GEN-110>	The requirements for a “sensitivity analysis mode” that allows iteration on any module parameter shall be specified, if applicable.

5.2. ARCHITECTURE REQUIREMENTS

<ARC-010>	The architecture of the E2ES shall be modular and expandable.

<ARC-020>	The simulator shall provide an end-to-end simulation capability by using a functional and modular architecture and interfaces based on the following six high-level modules: - Geometry Module - Scene Generator Module - Instrument Module - Level-1 Processing Module - Level-2 Retrieval Module - Performance Evaluation Module *Note: for Phase B and later phases the simulator shall also include an "Platform simulation and Onboard data generation" Module.*
<ARC-030>	The architecture of the E2E simulator shall have at least one of each of the high-level modules specified by <ARC-020>, and may have more than one of some of them depending on the mission characteristics.
<ARC-040>	In case of multiple-instruments, the E2ES shall define independent simulation chains for each instrument.
<ARC-050>	The Geometry Module shall handle the simulation of the SC orbit and attitude, as well as the generation of the observation geometry the instrument(s).
<ARC-060>	The Scene Generator Module(s) shall handle the simulation of the scene(s) to be observed and all environmental effects to be considered for the correct generation of the stimuli to be entered to the instrument model(s).
<ARC-070>	The Instrument Module(s) shall handle the simulation of the sensor(s) behaviour, having different outputs depending on the type of instrument.
<ARC-080>	The Level-1 Processing Module(s) shall handle the generation of level-1 products.
<ARC-090>	The Level-2 Retrieval Module(s) shall handle the retrieval of the geophysical parameters that are objective of the mission/instrument(s).
<ARC-100>	The Performance Evaluation Module shall handle the needed analysis of the simulator outputs to evaluate the performances of the mission.
<ARC-110>	It shall be possible to execute the Performance Evaluation Module at different points of the simulation chain.
<ARC-120>	The high-level modules specified by <ARC-020> shall be self-contained, to facilitate the exchange of components by modified or more refined elements.
<ARC-130>	The implementation of the high-level modules specified by <ARC-020> shall be modular in the form of interconnected building blocks, to facilitate the exchange of components by modified or more refined elements.
<ARC-140>	The building blocks composing the high-level modules shall be reused, as much as possible, from existing EO repositories or other E2ES.
<ARC-150>	The implementation of the building blocks shall make use, as much as possible, of existing EO libraries or developments from other E2ES.
<ARC-160>	The E2ES shall support the independent execution of any of the high-level modules specified by <ARC-020>, using as input results generated in earlier simulation runs.
<ARC-170>	The E2E simulation shall be data driven, allowing the user to define partial processing chains implying the consecutive execution of some of the high-level modules.
<ARC-180>	The requirements associated to each high-level module shall be specified, including error injection and/or computation.

5.3. INTERFACE REQUIREMENTS

<INT-010>	The interfaces between the high-level modules specified by <ARC-020> must be through one or more input-output data exchange files in a project-specific format (netCDF, TIFF, XML, etc.).
<INT-020>	The E2E simulator shall have a global configuration file that is input to all the high-level modules.
<INT-030>	Each high-level module shall have a dedicated configuration file, to be read in addition to the global configuration file defined in <INT-020>.
<INT-040>	All configuration files in the E2ES shall be in XML format.
<INT-050>	The E2ES shall store all intermediate results of each of the high-level modules.
<INT-060>	The E2ES shall have a configurable option in the global configuration file to eliminate all intermediate results of each of the high-level modules.
<INT-070>	The error injection requirements shall be specified. These could include functional error injection specification, statistical error injection or a combination of both.
<INT-080>	The capability to read data from external files instead of computing the data in the simulator shall be specified, if needed.
<INT-090>	The external auxiliary data that has be read by the E2ES shall de identified and specified.

5.4. PERFORMANCE REQUIREMENTS

<PER-010>	The errors introduced in the simulator chain by the architecture and implementation of the E2ES shall be negligible in comparison with errors introduced by the Geometry Module, the Scene Generator Module, the Instrument Module, the Level-1 Processing Module and the Level-1 Retrieval Module.

6. GENERIC EO SIMULATION FRAMEWORKS AND MODULE REPOSITORIES

This section presents an analysis of generic frameworks and model repositories, together with the implications of the analysed frameworks and repositories in the definition of the Reference Architecture.

The generation of the reference architecture has taken into account, as much as possible, the existing ESA developments for facilitating the reuse of E2E EO simulators (frameworks and model repositories), both orienting the reference architecture to the possible use of those tools but keeping a critical approach to them.

This section presents a review of their main features and offers and critically analyses their suitability to implement the reference architecture.

Several existing tools, model repositories and concepts are mentioned in the section 2 of the SoW, [AD.1], providing a general view of the background activities related to this ITT. But among them, two elements have to be highlighted:

- The OpenSF framework for E2E EO simulators. It is explicitly mentioned as the simulation framework in which the reference architecture shall be evaluated. Its analysis is presented in Section 6.1.
- The Open Source Software Repository. It is also explicitly mentioned, as it has to be evaluated its suitability for being the EO model repository. Its analysis is presented in Section 6.2.
    - Provides possibility to integrate heterogenous and distributed simulation packages (e.g. Matlab, Excel, C/C++, Fortran…).
    - Emphasis made on technology behind simulation framework and the interface standards used (HLA).
    - Provides wrapper for packages not a priori compliant.
    - Doesn't focus on semantics of modules.
    - Doesn't provide generic architecture for use cases in Earth Observation.
    - Limited to data driven simulations.

In addition to these major developments, Section 6.3 includes a review and assessment of EODiSP.

6.1. OPENSF FRAMEWORK

OpenSF is a simulation framework that has been initially developed based on the EarthCare E2E Simulator, so it is completely focused on the simulation of EO missions. The OpenSF documentation (i.e. [RD. 4] [RD. 5] [RD. 6]) provides a detailed description of the OpenSF functionalities and how to use it for the integration of E2E simulations of EO missions. This section summarises the most important features of the OpenSF framework, in terms of their adequacy to support the intended reference architecture definition.

The OpenSF framework allows:

- The integration of the separate modules composing a simulator, by means of defining the inputs and output files of each module and their associated links. These modules can be of heterogeneous nature (i.e. executables or source code in C/C++, Matlab, IDL, Fortran. Python,…)
- The interaction with the user in what respects to the definition of the simulation chain.
- The editing of the configuration parameters through the Parameter Editor.
- Controlling the execution of the E2E simulator (sequential call to the different modules, logs, progress notification…)
- Editing and visualising the simulator outputs by means of external tools associated to certain file extensions.

The integration of the simulator modules and the execution of a simulation chain are possible thanks to the following elements defined by OpenSF framework:

- **Stages:** they are the phases of the simulation chain, following a predefined order of execution. Modules are associated to stages.

- **Descriptors:** set of input/output interfaces of a model. Their main objective is to link the different modules consecutively.
- **Modules:** each executable element that can take part of a simulation. An OpenSF simulation stage may contain one or several modules in it. The Module has its own descriptors associated (input and output interfaces) and two configuration files (one global for the complete simulator, one specific of the module) defined by the XML file whose format is specified in ESA generic ICD [RD. 130]. Figure 6-1 shows the editing panel of a model in OpenSF. OpenSF supports modules implemented in almost any kind of programming language. For the interaction of the model developer with the framework (reading configuration parameters, sending log or progress messages,...) the OpenSF provides a library called OSFI in the following programming languages: FORTRAN 77/90/2003, C/C++, Matlab, Python, and IDL.
- **Simulations:** the set of modules that are run sequentially under the order defined by the stages creates a simulation. One simulation may consist in the selection of only one model of the simulation chain. Simulation are usesd as (optional) templates to define Sessions.
- **Sessions:** it is the instantiation of a E2E simulation chain. It can be defined based on templates (called simulation) or directly concatenating modules.. All information related to a session (execution time and status, logging, configuration/input/output files) is stored once executed by the user. It is possible also to generate scripts allowing the execution in batch mode of a session defined in OpenSF. Figure 6-2 shows the editing panel of a simulation in OpenSF.

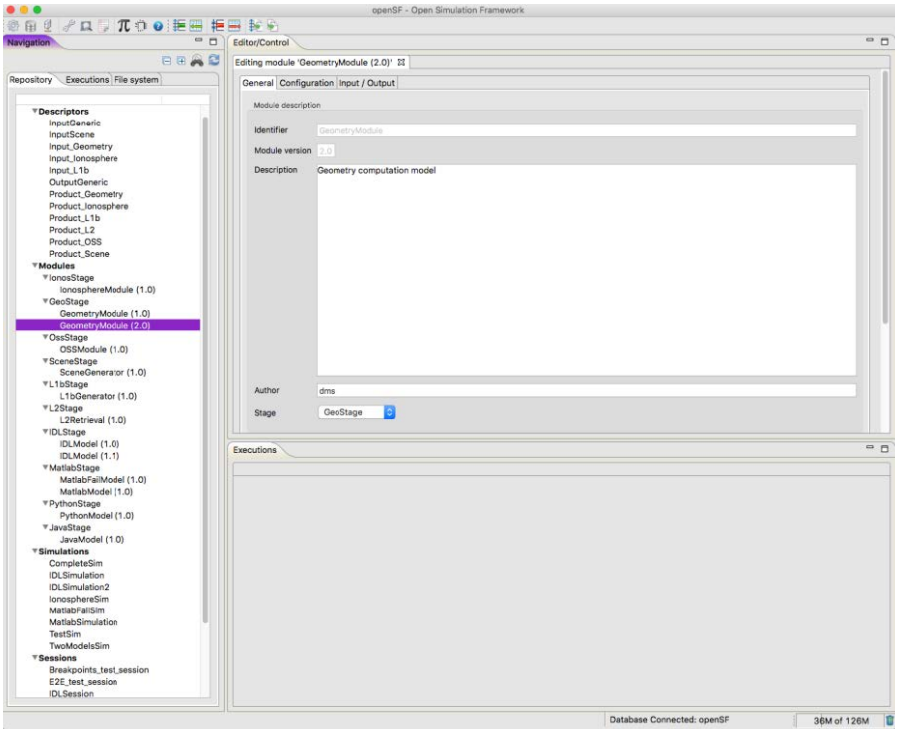


Figure 6-1: Example of OpenSF panel for editing a module

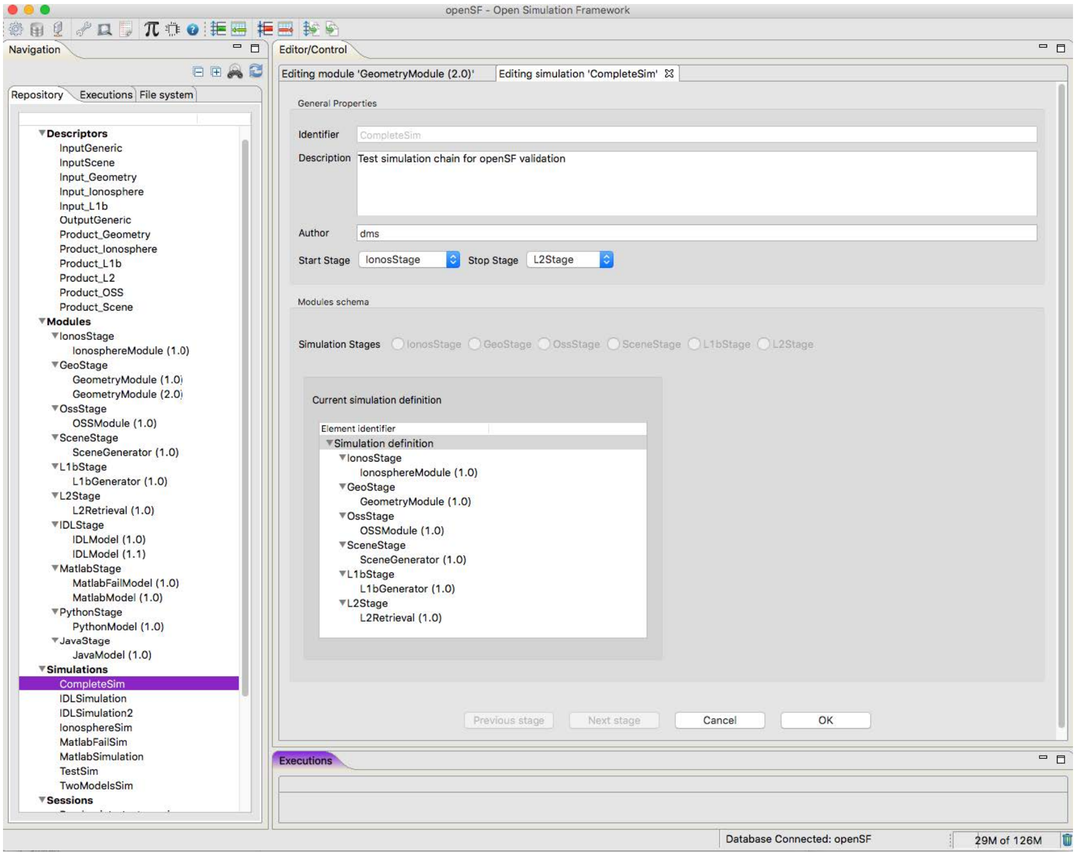


Figure 6-2: Example of OpenSF panel for editing a simulatiom

Therefore, starting from the definition of each single module within the OpenSF framework, it is possible to integrate the components by means of their I/O interfaces and then complete the E2E Simulation chain. Once all the modules are perfectly defined within OpenSF, the infrastructure is responsible for the correct control of each simulation. Any stage of the E2E Simulator can be run separately or together with its adjacent stages, by means of the concept of Simulation. Finally, OpenSF Sessions allow running several executions of different simulation chains at once. It shall be highlighted that the current version of OpenSF does not support parallel processing of different simulations at the same time, so the concept of Session is purely sequential.

As it has been already mentioned, each model has two configuration files associated: one global and one specific. The user can access the contents of both files through a specific tool within the OpenSF environment called Parameter Editor. Apart from displaying the formats and contents of the configuration file, allowing their edition, it also permits the consistency checking of parameters (range, type…) and the definition of rules and constraints between them, even if they are located in different configuration files. A snapshot of the Parameter Editor is shown in Figure 6-3.

With respect to the presentation of results, OpenSF allows assigning an external tool to a determined file extension, in such a way that, if the user double-clicks on the file in the OpenSF MMI, the external tool opens directly that file. For instance, if a model generates a jpeg file as output, an image editor can be assigned to that extension.

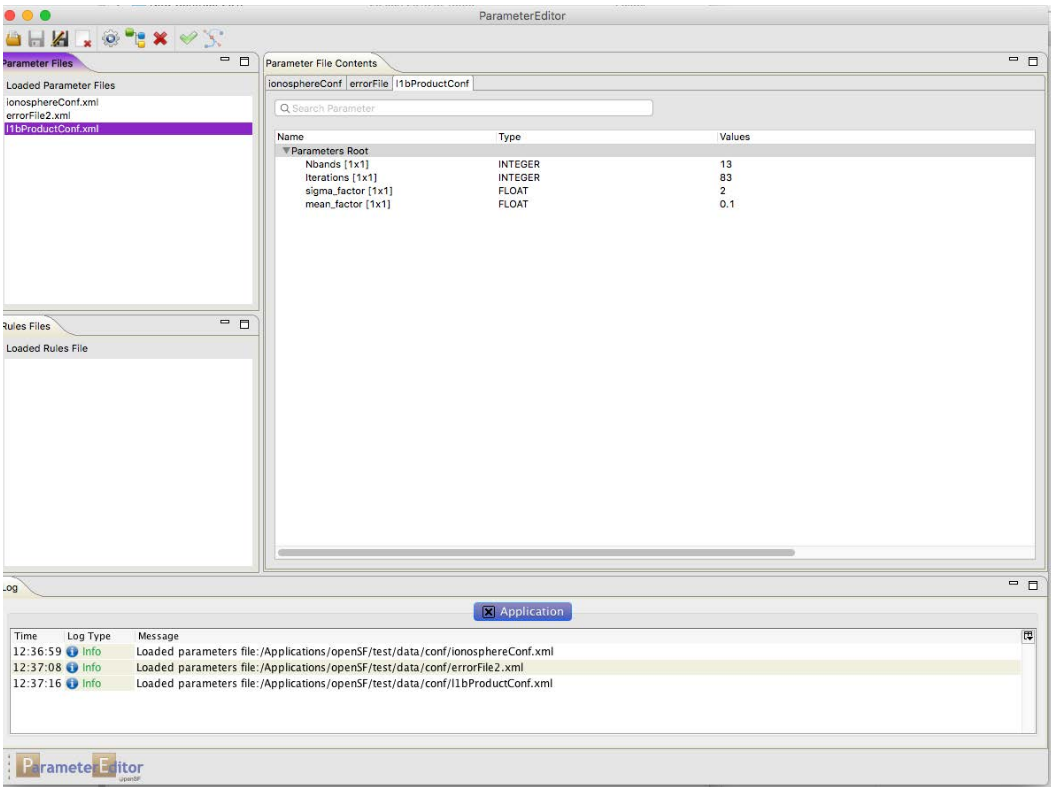


Figure 6-3: Example of editing a XML configuration file with Parameter Editor

The experience of GMV and Aresys in the integration of PREMIER and CoReH2O E2E simulators allows providing a list of advantage and disadvantages associated to the use of the OpenSF framework. The main advantages that can be highlighted are:

- The OpenSF framework assumes efficiently the data flow management of a sequential execution of very heterogeneous modules, coming from different sources.
- It allows the implementation of modules in any programming language following the scheme of command line call of an executable.
- Ensures the coherence of the input/output data when connecting the modules to create a simulation chain.
- Moreover it provides support libraries (OSFI) for interacting with the framework in the most frequent programming languages used normally in EO simulation at the early phases of the mission (Matlab, IDL, FORTRAN77/90/2003, C/C++, Python).

Its main disadvantages would be:

- The interfaces between the modules are based on files. Going down in the level of granularity of the definition of the reference architecture, if numerous sequential building blocks are defined:
  - The simulation could be slowed down due to the frequent access to hard disk.
  - It would be produced a vast amount of data in heavy simulation scenarios.
- Only sequential executions are allowed, i.e. no loops: a module can only be executed after execution of previous module has ended.
- The framework does not offer limited possibility for parallel processing..
- The rigid definition of simulation stages puts heavy work on the simulator integrator. For example, in order to add a new input to a model, keeping the same executable and just adding a new input file, requires redefining the model from scratch.
- It is difficult to incorporate to the framework a module for Performance Evaluation. The rigid definition of simulation stages and model descriptors in OpenSF has obvious advantages, but does not allow integrating a module that could be run at different points of the E2E simulation chain

and that it would need outputs from a variable number of previous simulation executions (comparison of results).

- The concept of auxiliary libraries, which might be helpful in different modules implementing the same low level functionality, is not supported. Common functionality shall be incorporated directly and separately in each module.

This list of disadvantages does not mean, a priory, that OpenSF is unsuitable for the concept or a Reference Architecture. On the contrary, based on GMV's experience and comparing the development of E2E simulators from scratch and based on the OpenSF framework, it does offer a set of unique features that are of value for building an E2E simulator. Section 6. provides an overview of the implications of using OpenSF for building-up an E2E simulator based on the concept of a Reference Architecture.

Section 7. [PD.3]defines a series of requirements for a generic framework supporting the Reference Architecture, and OpenSF will be evaluated against these requirements in future work of the ARCHEO-E2E project.

[PD.3]

6.2. OPEN SOURCE SOFTWARE REPOSITORY

The Open Source Software Repository (OSSR) is an ESA project that was born to address the fact that, when ESA initiates and manages a new open source project, there was no standard infrastructure support for the management and hosting of the project. Moreover, this lack of infrastructure made difficult for ESA and industry to have an overview of existing assets that could be reused or used as a starting point for a new development. [RD. 7]

The objective of this activity was to analyze the requirements, architecture and design of a repository of space open source software (OSS), covering flight software, flight software and engineering tools. It was remarked in the OSSR SoW that the repository shall not exclude hosting other type of software different from open source code, like operational or non-open source SW if identified as relevant.

A priori, it seems that the OSSR would be useful to hold also the EO modules coming from the definition of the reference architecture and generic building blocks. As this ITT is focused in generic building blocks, it is not expected that the resulting SW pieces will contain proprietary information, so in their implementation the BBs could be treated as open source SW elements.

A prototype (proof-of-concept) of the repository has been made available to GMV for its evaluation, together with the relevant project documentation. The rest of this section is based on first-hand evaluation of both the documentation and the prototype. Please note that, in addition to the functionalities more related to a software repository, OSSR also offer a powerful collaborative software development environment which is out of the scope of this activity. Therefore the evaluation of features has not taken these capabilities into account.

The operation of OSSR is based on the concept of projects. Upon the creation of a new project one can create different packages associated to that project. Each of these packages can have different files associated to it (e.g. source code, executables, etc.), and new releases of the complete package are allowed and kept track of. Figure 6-4 shows an example of packages created to emulate the possible high-level modules of the Reference Architecture, where several files have been released under the package "Scene Generator Module".

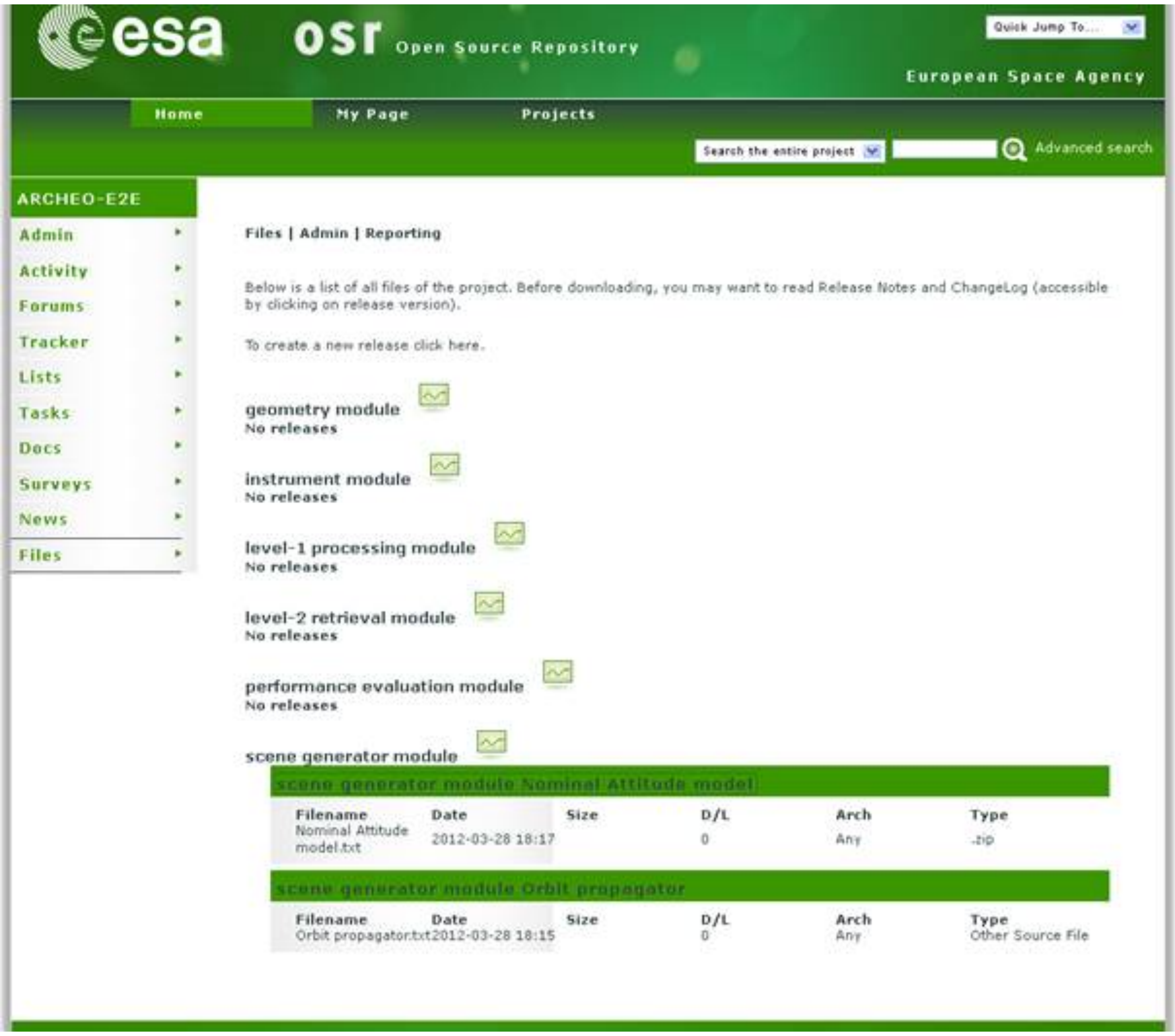


Figure 6-4: Example of packages created on an OSSR project

Upon reviewing the documentation [RD. 9] [RD. 8] and using the OSSR prototype, the following main advantages can be highlighted:

- Offers repository capabilities in a collaborative distributed environment.
- Hosts not only open source code, but also non open source code and other items such as mathematical models.
- Possibilities of adding release notes and associated documentation to a certain file within a package.
- Tracking of different versions and releases for each of the packages uploaded.
- Logs activity of new uploads and version changes, so that it is easy to identify the latest additions to the repository.
- Implements ESA security policies and allows classification of software and documents as Unrestricted and Restricted.
- In addition to these functionalities more related to a software repository, OSSR also offer a powerful collaborative software development environment.

The main disadvantages found from the point of view of the needs to sustain the concept of the Reference Architecture are:

- There is no possibility of creating a hierarchy of packages within a project. This would be desirable in order to have higher-level packages identified as the high-level modules in the reference architecture, and lower-level building blocks depending on these modules.

- There is no possibility of cross-communication between projects. This would be of interest to have central repositories (projects) with all the modules and building blocks of a certain type of instrument, and allocate them to a new project. In addition, it would be desirable that new releases of modules and building blocks in a user project are copied back to the main repository, ensuring a complete global repository and enabling reuse.

The requirements for a repository of EO modules supporting the concept of Reference Architecture have been derived and presented in Section 7. Future work in the ARCHEO-E2E project will check these requirements against OSSR to assess its suitability.

6.3. EODISP

EODiSP is a generic platform to support the development and operation of distributed simulators that was developed under ESA contract. It was initially aimed at supporting the development of EO E2E simulators, but since it provides capabilities for integration of packages of heterogeneous nature it is, in principle, suitable for other types of simulators. [RD. 10]

Its main advantages are:

- Provides possibility to integrate heterogenous and distributed simulation packages (e.g. Matlab, Excel, C/C++, Fortran…).
- Provides wrapper for packages not a priori compliant.
- Implements the HLA standard, the most widely used simulation architecture. It is implemented in a dedicated package so it could be easily reused.

On the other hand, its main disadvantages are:

- Emphasises the technology behind the simulation framework and the interface standards used, rather than usability and applications.
- Doesn't focus on semantics of modules.
- Doesn't provide generic architecture for use cases in Earth Observation.
- It is limited to data driven simulations.

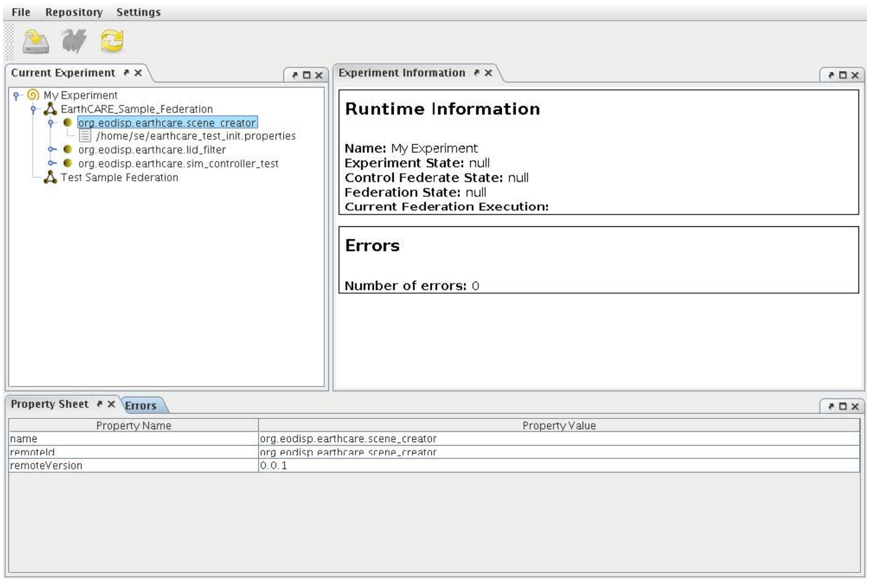


Figure 6-5: Example of EODiSP user interface

It is considered that EODiSP does not offer any significant advantage over OpenSF, which is more suitable towards the development of EO E2E simulators. Therefore the analysis to be done in the ARCHEO-E2E project will focus on OpenSF.

6.4. IMPLICATIONS OF GENERIC EO SIMULATION FRAMEWORKS IN THE REFERENCE ARCHITECTURE

Although the methodology proposed for describing the architecture, as shown in Section 4.4, is generic and non-dependent on the final implementation method or the simulation framework, there are a number of aspects related to the simulation framework that have implications on the way the reference architecture can be interpreted and on the definition of the scope of the building blocks. This section makes a evaluation of this impact and identifies features of the simulation framework that affect the interpretation of the reference architecture and the building blocks.

Based on the currently existing simulation framework and the evaluation of OpenSF done in Section 6.1, Table 6-1 shows three different options in the implementation of en EO simulation framework, always maintaining the common philosophy of having an architecture based on high-level modules that are connected to each other.

Table 6-1 Options for the implementation of a simulation framework

Option	Description	Advantages	Disadvantages
1	High-level modules (i.e. building blocks) with interface via input/output files.	Simple interconnection of the modules. Independent development of each module.	Less flexibility in the implementation. No control over the reusability of the implementation elements. Need to access disk during the simulation. Only sequential execution.
2	High-level modules with input/output file interfaces and building blocks as implementation elements of the high-level modules via coding libraries (e.g. processing functions).	Simple interconnection of the building blocks. Reusability in the implementation of the high-level modules. Medium flexibility in the implementation.	Need to access disk during the simulation. Only sequential execution.
3	High-level modules and building blocks as implementation elements of the high-level modules via integrated editor.	Flexibility on the implementation. Reusability in the implementation of the high-level modules. No need to access disk during the simulation. Non-sequential execution allowed.	Complex simulation framework.

Option 1 is the one currently implemented in OpenSF. The framework works from high-level compiled modules that are interfaced to each other via input/ouput files. This philosophy allows independent development of each module and an easy interconnection given that the interfaces have been previously defined. In this option the high-level modules are effectively the building blocks of the simulator. The main disadvantages are the lack of control over the reusability of functions or sub-modules and, more related to the execution of the simulator, the inability of interaction among modules beyond a purely sequential execution.

A first step to improve the reusability of modules and functions is the definition of building blocks, which is the goal of the present activity and what has been contemplated in the definition of the reference architecture. However there are two different approaches represented by Option 2 and Option 1.

Option 2 is functionally equivalent to Option 1, but there is an external “toolbox” of building blocks defined in the form of libraries that are used to implement the high-level compiled modules to be included in the simulation framework. This approach requires the previous effort of standardising the common functions and modules and ensures consistency among high-level modules using the same functionalities. The main disadvantage in this approach is, like in Option1, the inability of interaction among modules beyond a purely sequential execution.

Finally, Option 3 aims at overcoming the disadvantages of having close modules as the high-level elements in the simulation framework. In this Option the building blocks would be sub-elements in the simulation framework, which would allow building a high-level module by interconnecting building blocks within the same simulation framework. This translates into a much versatile but complex simulation framework, with an environment similar to COTS such as MathWorks' Simulink or National Instrument's LabView. The main advantages are, as in Option 2, the reusability of building blocks, and in addition a complete flexibility in the implementation and execution of high-level modules, not constrained by a sequential operation and allowing interaction among the high-level modules.

- If Option 1 would be chosen, the mentioned high-level elements of the Reference Architecture would correspond to OpenSF stages and the building blocks would be understood as OpenSF modules that are sequentially executed.
- Option 2 goes for a more flexible concept of simulator framework, where the building blocks would be understood as libraries providing functionality that can be used by the developer for the generation of OpenSF Modules. In this case, depending on the level of granularity of the generic building block definitions, the high-level elements could be seen also as OpenSF Modules, apart from being OpenSF Stages.

The premises for the selection of one or another option are to maximize the reuse of current frameworks and exploit OpenSF advantages, while at the same time to propose alternatives for incorporation of the concept of Reference Architecture. The most straightforward solution would be to choose Option 2, with the following features:

- It is functionally equivalent to Option 1.
- It includes an external toolbox or repository of building blocks to be used for implementation of high-level modules.
- There is a significant effort required to standardize common functions and modules and ensure consistency.

The proposed solution is an improved Option 2, as shown in Figure 4-10. This solution overcomes some drawbacks of the current OpenSF implementation by helping the user set a simulator from predefined reference architectures. In this option the framework has the ability of loading pre-defined schemes/templates according to the defined reference architectures. Then, the simulation stages are created with template configuration files to be filled-in by the user according to the specific application.

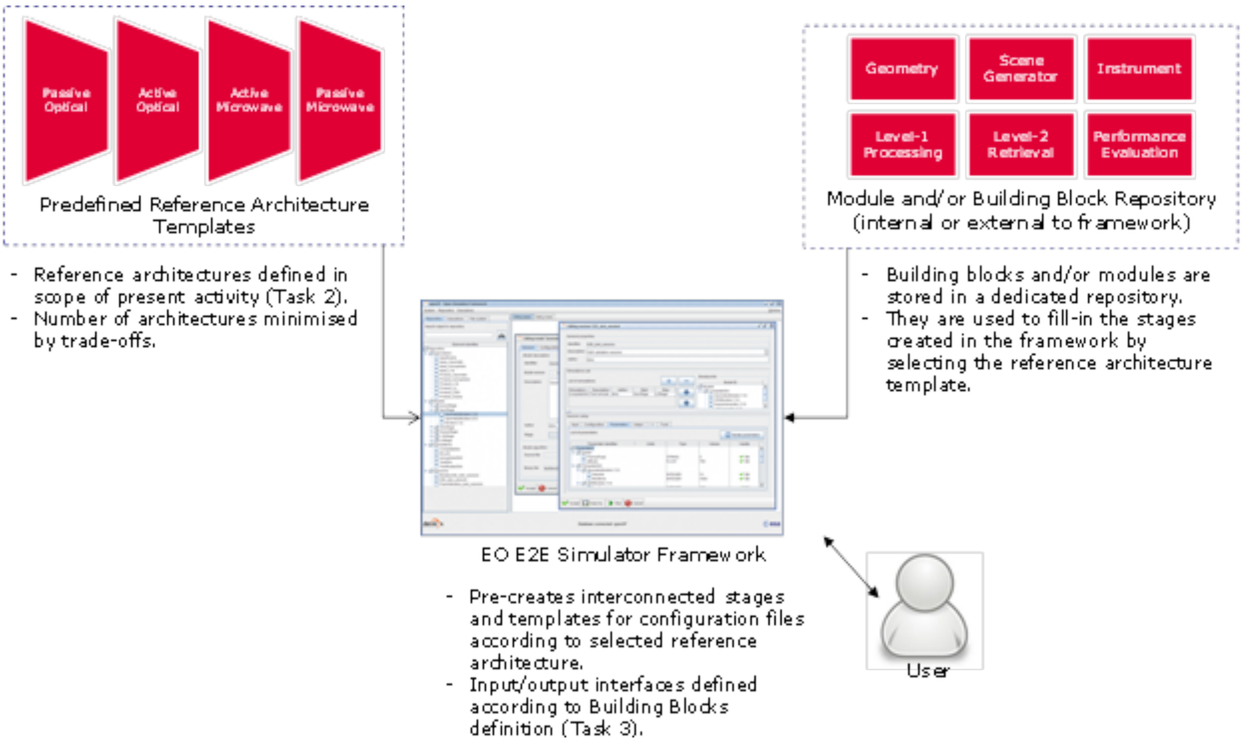


Figure 6-6: Proposed solution for the implementation of the Reference Architecture concept

The steps followed in the process of setting the initial simulator are the following:

- The user selects a predefined reference architecture based on the type of mission, number of instruments and type of instruments.
- The framework has the ability of loading pre-defined schemes/templates according to the reference architectures and the type of instrument selected by the used.
- Simulation stages are created with template configuration files to be filled-in by user according to specific application. Simulation stages may correspond to high-level modules or, preferably, to building blocks. The stages correspond to the different building blocks composing each of the modules. Depending on the type of instrument selected by the user the stages may or not may be present, or a dummy may be created.
- The framework will show different possibilities for the implementation of each stage according to an internal or externally linked repository.

7. REQUIREMENTS FOR A SIMULATION FRAMEWORK AND MODULE REPOSITORY

This section identifies the requirements for a framework and a repository of EO modules supporting the development of new end-to-end simulators based on the proposed Reference Architecture. The definition of requirements has been done taking into consideration the requirements of the E2E Simulators of the three candidate Earth Explorer missions currently under study in phase A (CoReH2O, PREMIER and BIOMASS), from the ARCHEO-E2E Statement of Work and the first-hand experience of the project team.

7.1. REQUIREMENTS FOR A SIMULATION FRAMEWORK

7.1.1. GENERAL REQUIREMENTS

<GEN-010>	The framework shall allow building an end-to-end performance simulator in a modular and expandable manner.
<GEN-020>	The framework shall the simulation of different instrument chains within the same E2E simulator.
<GEN-030>	The framework shall provide generic architectures for each instrument chain depending on the type of instrument to be simulated.
<GEN-040>	The framework shall, upon selecting a generic architecture, set up a template for the different modules in the simulator and the interfaces among them.
<GEN-050>	The framework shall allow modification of the generic architecture template to build-up a customized simulator.
<GEN-060>	The framework shall allow sharing a same module implementation among different instrument chains.
<GEN-070>	The framework shall provide an end-to-end simulation capability by using the functional and modular architecture and interfaces defined in the reference architecture.
<GEN-080>	The framework shall support the implementation of a data-driven simulation.
<GEN-090>	The framework shall support the execution of sensitivity analysis and Montecarlo simulations.
<GEN-100>	The framework shall allow associating an error model and/or a range of variation to any input parameter with the purpose of performing sensitivity analysis and Montecarlo simulations.
<GEN-110>	The framework shall allow the batch execution of all simulations resulting from the variation of the specified input parameters.
<GEN-120>	The framework shall minimize the impact of file-based interfaces between modules when performing sensitivity analysis and Montecarlo simulations.

7.1.2. MODULES AND INTERFACE REQUIREMENTS

<MIT-010>	The framework shall allow including both Customer-Furnished Item (CFI) modules and newly developed ones.
<MIT-020>	The framework shall allow integrating heterogenous simulation packages (e.g. Matlab, C/C++, Fortran…).
<MIT-030>	The framework shall facilitate the exchange of modules by modified or more refined versions without the need to redefine the entire simulation chain.

7.1.3. CONTROL REQUIREMENTS

<CTL-010>	The framework shall allow specifying execution chains composed of several modules in the architecture.
<CTL-020>	The framework shall allow running independently each of the different chains.
<CTL-030>	The framework shall provide the basic functions needed for the execution of the data exchange mechanisms, data flow control and maintenance of interfaces.
<CTL-040>	The framework shall univocally name each simulation run and create an associated scenario.
<CTL-050>	The framework shall allow deleting an entire scenario.
<CTL-060>	The framework shall allow copying a scenario, modify its inputs and settings and launch the new scenario.
<CTL-070>	The framework shall store all data pertaining to a processing run in the associated scenario folder.
<CTL-080>	The framework shall allow controlling the simulation either via command line or via a user interface.
<CTL-090>	The framework shall allow preparing several runs and launch multiple runs in a user defined sequence (batch mode).
<CTL-100>	The framework shall allow stopping or pausing and restarting the execution of a simulation.
<CTL-110>	The framework shall support the independent execution of the modules using as input results generated in earlier simulation runs.
<CTL-120>	The framework shall allow visual editing of XML configuration files associated to each module.
<CTL-130>	The framework shall provide progress information on the execution of each module.
<CTL-140>	The framework shall provide progress information on the execution of each simulation chain.
<CTL-150>	The framework shall detect the issue of an error message by the module being executed and stop the execution of the simulation.

7.1.4. DATA STORAGE AND VISUALIZATION REQUIREMENTS

<DSV-010>	The framework shall allow storing all intermediate results of each of the modules.
<DSV-020>	The framework shall support displaying and storing an event log of each processing run.
<DSV-030>	The framework shall provide visualization capabilities to visualize simulation results.
<DSV-040>	The framework shall allow associating external tools to certain file extensions in order to visualize simulator outputs.

7.2. REQUIREMENTS FOR A MODEL REPOSITORY

7.2.1. GENERAL REQUIREMENTS

<GEN-010>	The repository shall allow hosting source code and libraries of heterogenous nature (e.g. C/C++, Matlab, Fortran…).

<GEN-020>	The repository shall allow three different levels of items to be stored: building blocks, modules and generic libraries.
<GEN-030>	The repository shall track the different versions of each of the items stored.
<GEN-040>	The repository shall tag each of the items stored with the programming language under which the item has been developed.
<GEN-050>	The repository shall tag each of the items with a classification level of Unrestricted, Restricted, or Confidential.
<GEN-060>	The repository shall allow searching for information (e.g. metadata in building blocks).
<GEN-070>	The repository shall allow uploading documentation associated to each of the items stored.
<GEN-080>	The repository shall allow the definition of reference projects that store the items related to a specific category (i.e. generic of specific instruments).
<GEN-090>	The repository shall allow the definition of working projects under which to link to existing items in the reference projects.
<GEN-100>	The repository shall allow the working projects to create new items and make them available in the related reference project.

7.2.2. ITEMS STORAGE REQUIREMENTS

<IST-010>	The repository shall always store building blocks linked to a higher-level building block or to a module.
<IST-020>	The repository shall store the modules either as a collection of links to different building blocks, or as a monolithic implementation.
<IST-030>	The repository shall store the building blocks with additional information associated to them, as described in the template shown in Section 7.2.2
<IST-040>	The repository shall associate additional metadata to each version of the items stored, such as the project or mission where it is used.

7.2.3. ACCESIBILITY REQUIREMENTS

<ACC-010>	The repository shall be remotely accessed by all authorized users.
<ACC-020>	The repository shall implement different levels of access for administrators and authorized users.
<ACC-030>	The repository shall allow uploading new building blocks and uploading updated versions of existing building blocks.
<ACC-040>	The repository shall allow creating new high-level modules for any existing category.
<ACC-050>	The repository shall allow the administrator to create new categories for the high-level modules.

8. REFERENCE ARCHITECTURE MODULES

This section presents the high-level modules of the reference architecture, identifying those elements that are common across different instrument categories.

Once the high-level approach to the reference architecture it is settled, it is convenient to analyse which common architectural elements (building blocks) could be applicable to several/all identified categories of missions and instruments. Will they be completely different the building blocks composing the Geometry Module of an active microwave mission with respect to that of a passive optical mission? And the Scene Generator or the Instrument Module?

In principle it is quite obvious that the Instrument, Level-1 processing and Level-2 Retrieval modules will be highly dependent on the category of mission/instrument, and all the more, in the specific mission/instrument within the same category. However a extensive review of the modules and building blocks proposed for the different categories has allowed deriving some commonalities to reach some common architectures at the level of the high-level modules.

But most of the common elements that can be translated to the Reference Architecture will be found in the Geometry Module, and in less extent, in the Scene Generator Module.

8.1. GEOMETRY MODULE

This high-level component of the architecture will be in most of the cases the first element to be run in an E2E simulation.

- The Geometry Module will present many commonalities between the different types of missions and instruments:
  - At architectural level, being common for most of the missions the building blocks that can be defined.
  - At model level, being the same modules (implementations) valid for different missions (J2 analytic propagator, Yaw Steering, Zero Doppler, attitude error models…).
- Auxiliary functions like coordinate frame conversions or basic geometry libraries could be also considered as part of the Geometry Module (maybe valid also for other high-level modules).

The Geometry Module would be typically integrated by the following building blocks (Figure 8-1):

- Orbit Simulator
- Attitude Simulator
- AOCS/Instrument Coupling (Instrument Geometry)
- Scene Interaction Geometry

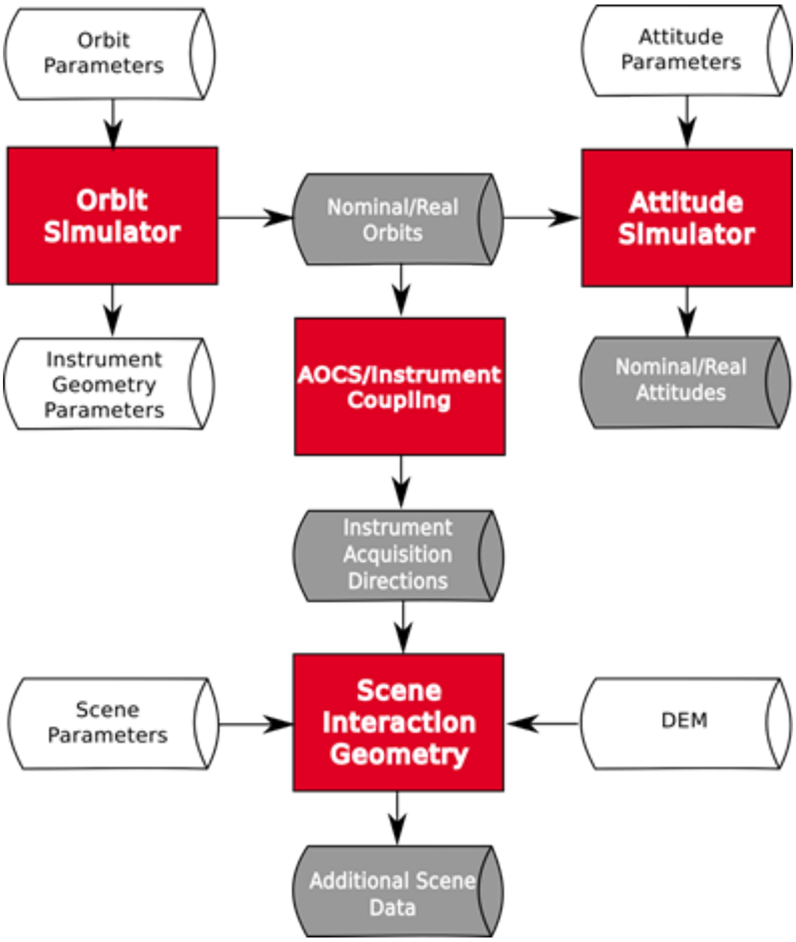


Figure 8-1: Data Flow of the generic Geometry Module

8.1.1. ORBIT SIMULATOR

The orbit simulator building block is in charge of generating nominal and real orbit profiles. This block is common for all kind of missions, including formation flying. For those missions with multiple instruments, only one instance per mission is simulated. The frequency at which the nominal/real orbit is generated depends on the configuration of the orbit simulator (see Table 8-1).

Different propagators can be used to feed the implementation of the Orbit Simulator, including relative dynamics in formation flying. There are many possible implementations of the Orbit Propagator, for example:

- J2 analytic propagation based on typical mean elements.
- TLE propagation.
- Numeric propagation with orbit perturbations (air drag, Earth geopotential, Moon & Sun gravitational effects…)

Each specific implementation will have slightly different configuration parameters, but the same output.

Figure 8-2 shows the generic architecture of the Geometry Module, in which the following building blocks have been identified:

- Nominal Orbit Computation
- Real Orbit Propagation

In addition, a generic "Orbit Propagation" building block is present both in the "Nominal Orbit Computation" and in the "Real Orbit Propagation" building blocks.

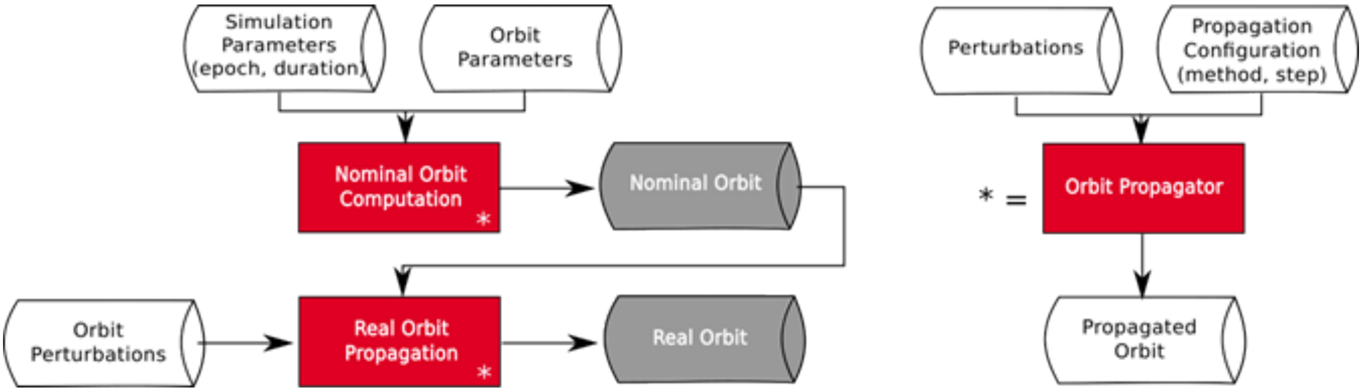


Figure 8-2: Data Flow of the Orbit Simulator Building Block

Table 8-1 shows the input and output files of the Orbit Simulator building block, as well as the main configuration parameters.

Table 8-1 Parameters of the Orbit Simulator Building Block

Input Files	Configuration Parameters	Output Files
N/A	- Epoch (MJD2000) - Simulation duration (seconds) - Orbit time step - Mean Orbital Elements (if J2 analytic propagation) - Two-Line Elements (if TLE propagation) - Osculating Orbital Elements or State Vector (if Numeric propagation) - Air density Model, number of geopotential terms (if Numeric propagation with perturbations)	- Nominal S/C Position and Velocity (if needed, it is the reference orbit, without perturbations) at each time step - Real S/C Position and Velocity during the data acquisition at each time step

8.1.2. ATTITUDE SIMULATOR

The attitude simulator building block is an element common for all kind of missions, and as in the case of the orbit simulator, only one instance for spacecraft is simulated.

There will be different implementation of the attitude simulator building block, depending on the nominal attitude profiles: i.e., yaw steering, Zero-Doppler steering, Limb pointing. It will also be possible to have external inputs for the attitude profiles to account for vibrations or thermal deformations.

Attitude errors shall be simulated to generate the real attitude profile (i.e., platform vibrations, mispointing...) from the nominal profile. The real attitude is generated at the same frequency as the nominal/real orbit.

The Real Attitude Simulator will be based on the application of errors on top of the nominal attitude. Error models will be generic:

- Bias
- Harmonic contributions.
- Spectral envelopes.
- …

Figure 8-3 shows the generic architecture of the Geometry Module, in which the following building blocks have been identified:

- Commanded Attitude Simulator
- Real Attitude Simulator

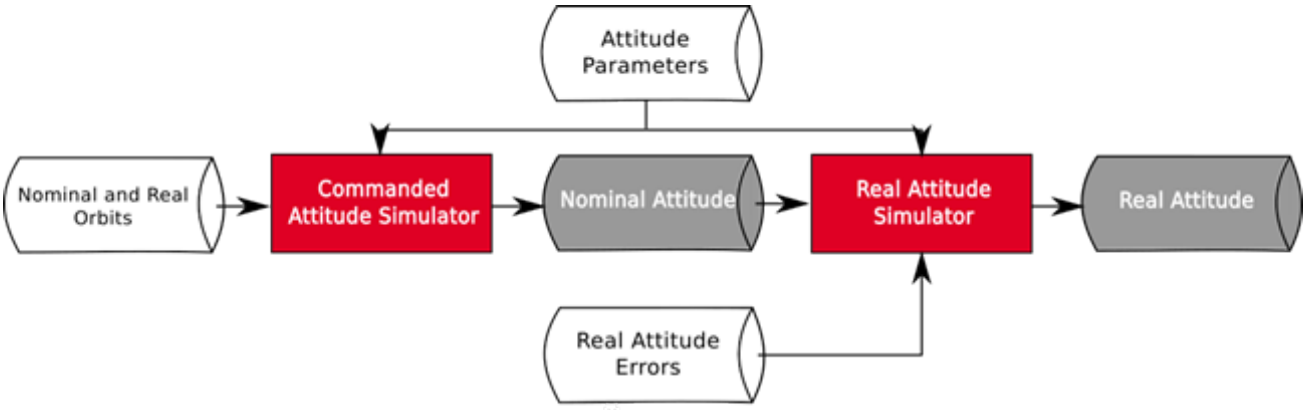


Figure 8-3: Data Flow of the Attitude Simulator Building Block

Table 8-2 shows the input and output files of the Attitude Simulator Building Block, as well as the main configuration parameters.

Table 8-2 Parameters of the Attitude Simulator Building Block

Input Files	Configuration Parameters	Output Files
- Nominal S/C Position and Velocity (if needed, it is the reference orbit, without perturbations) at each time step. - Real S/C Position and Velocity during the data acquisition at each time step.	- Epoch (MJD2000) - Simulation duration (seconds) - Type of attitude mode being commanded - Additional parameters defining the attitude mode (if any, example, altitude of limb point). - Error parameters to define real attitude	- Quaternions expressing the nominal S/C attitude (from spacecraft reference frame to ECEF reference frame) at each time step. - Quaternions expressing the real S/C attitude (from spacecraft reference frame to ECEF reference frame) at each time step.

In case of a mission with multiple instruments the Attitude Simulator may need to simulate not only platform errors but also pointing errors of the individual instruments.

In addition, and for calibration purposes (depending on the specific simulator needs) it may also be needed to provide Sun/Moon geometry data.

8.1.3. AOCS/INSTRUMENT COUPLING

The AOCS/Instrument coupling building block is present in all kind of missions and instruments, and there will be as many instances (with different implementations) as instruments being simulated.

This element simulates the mounting of each instrument in the platform, and also accounts for the observation geometry of the instrument (instrument FOV, focal plane geometry, depending on the type of instrument).

This element will be typically under the responsibility of the instrument experts.

Figure 8-4 shows the generic architecture of this module, in which the following building blocks have been identified:

- LOS Simulator
- Instrument Projection

While the architecture of the AOCS/Instrument Coupling is the same for the different categories of instruments, the actual implementation of the sub building blocks may differ from one instrument to the other. This will be dealt with in the specific sections further down this document.

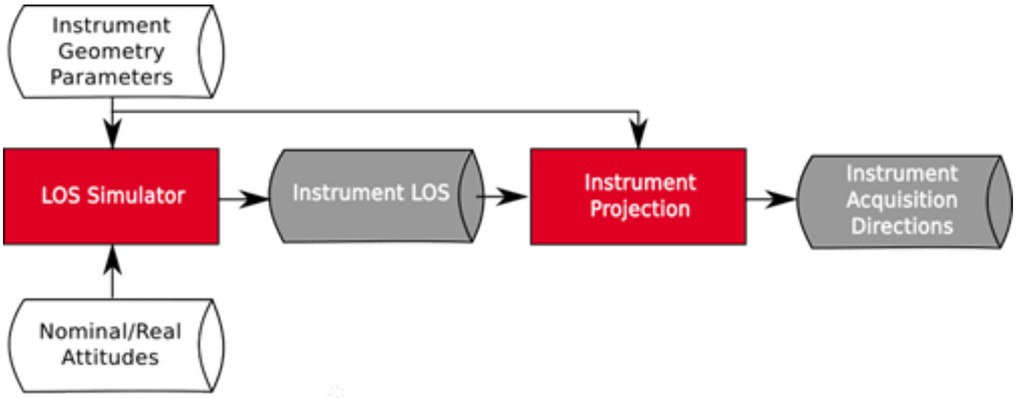


Figure 8-4: Data Flow of the AOCS/Instrument Coupling Building Block

Table 8-3 shows the input and output files of the AOCS/Instrument Coupling Building Block, as well as the main configuration parameters.

Table 8-3 Parameters of the AOCS/Instrument Coupling Building Block

Input Files	Configuration Parameters	Output Files
- Quaternions expressing the real S/C attitude (from spacecraft reference frame to ECEF reference frame) at each time step	- Epoch (MJD2000) - Simulation duration (seconds) - Instrument configuration - Instrument mounting data (rotation matrix from instrument reference frame to S/C reference frame) - Platform/Instrument interface errors.	- Instrument LOS directions (for each element of acquisition) expressed in the reference frame more adequate in each case (S/C reference frame, ECEF, ECI…) at each time step

8.1.4. SCENE INTERACTION GEOMETRY

The scene interaction geometry building block accounts for the generation of the instrument stimuli, which it is often necessary to calculate additional information from the interaction of the viewing geometry with the scene:

- Intersection of viewing directions with terrain (using DEM)
- Shadowing masks
- Observation geometry parameters
- …

Figure 8-5 shows the generic architecture of this module. While the architecture of the Scene Interaction Geometry building block is the same for the different categories of instruments, the actual implementation of the sub building blocks may differ from one instrument to the other. This will be dealt with in the specific sections further down this document.

For example, in the case of Active Microwave Instruments, the implementation of the “Intersection with DEM” building block will have additional functionalities present in the form of additional sub-building blocks to compute the geometric matrices to create correspondence between the geographic coordinates on DEM (e.g. Latitude/Longitude) and the instrument coordinates (e.g., azimuth/slant range). These Building Blocks are named SAR2GEO, GEO2SAR and Range ambiguities).

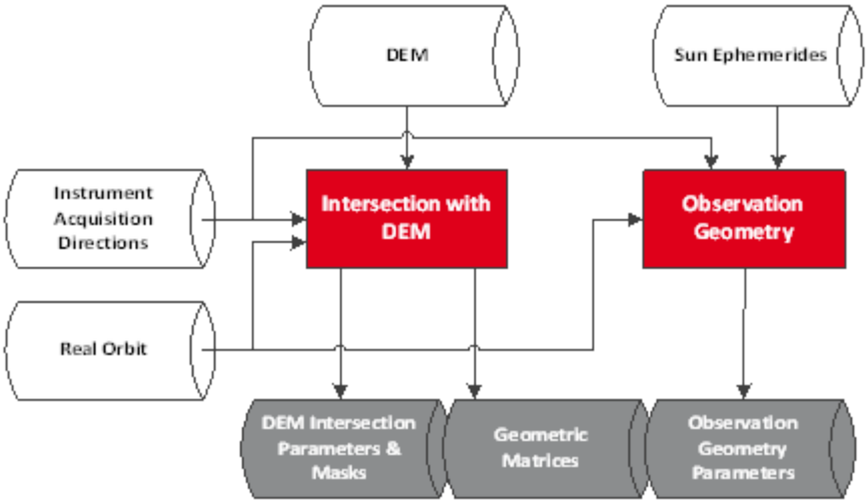


Figure 8-5: Data Flow of the Scene Interaction Geometry Building Block

The "Intersection with DEM" building block provides the line of sight for target geolocation as well as DEM-related parameters that allow modelling the effects of the topography on the generated scene.

- DEM Intersection Parameters & Masks:
  - Intersection of viewing directions with terrain
  - Sky View Factor
  - Terrain View Factor
  - Shadowing Map

In addition, the "Intersection with DEM" building block also provides:

- Geometric matrices for correspondence between geographic coordinates on DEM and instrument coordinates (Active Microwaves only)

The "Observation Geometry" building block provides parameters related to the geometry of the observation:

- Observation Geometry Parameters:
  - Observation Zenith/Azimuth Angle
  - Sun Zenith Angle
  - Relative Observation-Sun Azimuth Angle

Table 8-4shows the input and output files of the Scene Interaction Geometry Building Block, as well as the main configuration parameters.

Table 8-4 Parameters of the Scene Interaction Geometry Building Block

Input Files	Configuration Parameters	Output Files
- Instrument LOS directions (for each element of acquisition) expressed in the reference frame more adequate in each case (S/C reference frame, ECEF, scene reference frame) at each time step	- Epoch (MJD2000) - Simulation duration (seconds) - Scene location parameters	Specific auxiliary data (geometry-related) needed for each element of the scene or for each LOS direction, depending on the specific case: - DEM Intersection Parameters - Observation Geometry Parameters

8.2. SCENE GENERATOR MODULE

The Scene Generator Module simulates the scene to be observed and takes into account all relevant environmental effects in order to generate the stimuli to be used by the instrument model.

There two main blocks that could be considered for the Scene Generator:

- Reference Model
- Environment Simulator

Figure 8-6 shows the generic architecture for the Scene Generator Module.

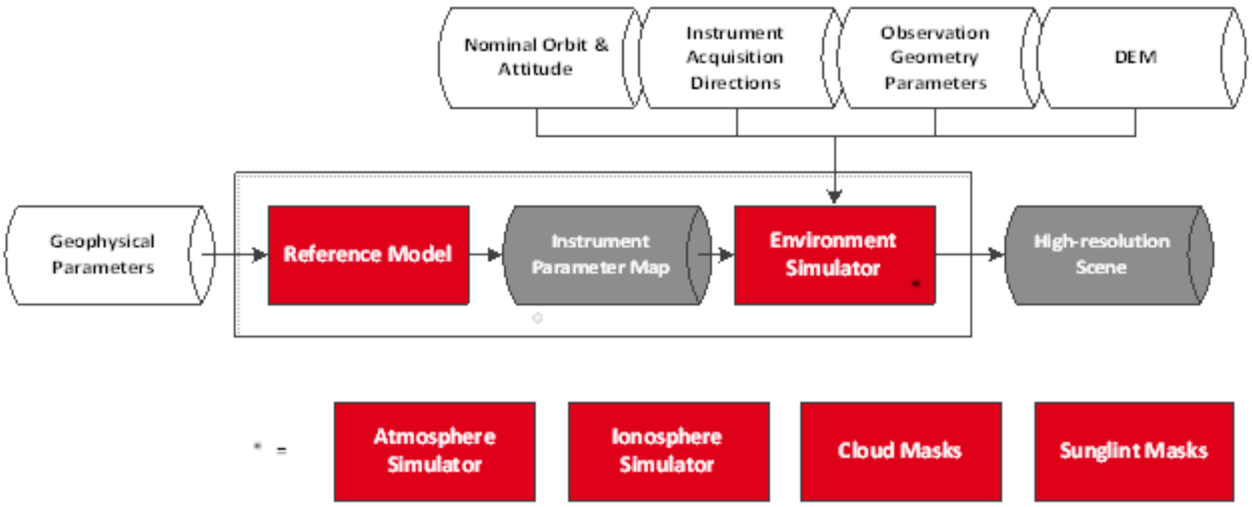


Figure 8-6: Generic architecture of the Scene Generator Module

As described in the following sections, there is room for some commonalities and a general architecture, but mainly the elements of the Scene Generator Module are highly dependent on the category of instrument/mission being considered.

Furthermore, the implementation of the Scene Generator shall allow the user to specify external data, such as existing cloud cover maps or atmospheric properties among others. These inputs are handled at the level of the individual building blocks.

8.2.1. REFERENCE FORWARD MODEL

The Reference Forward Model allows using as inputs to the simulator scene definitions in the form of bio/geophysical parameters (e.g. radiance profiles) and transforms them into Level-2 reference maps for the Scene Stimuli Generator. The Reference Forward model does not depend on the geometry of the orbit or the characteristics of the instrument, therefore is run independently from them. This process can, however, be very time consuming and, at the same time, one same reference map can be used for the generation of multiple scenes. Therefore the Reference Forward model is to be run only when needed, being up to the Scene Stimuli Generator to check if, for the specified input parameters, a reference map already exists, and if not, run the Reference Forward model. The outputs of this Reference Forward Model are also an output of the complete Scene Generator Module, since they are the "ideal" scenes to which compare the obtained Level-2 data.

For this definition of reference maps, many commonalities can be found between the different types of mission or instruments, mainly those sharing the same scientific target or application (land, ocean, atmosphere…). Even if the geophysical parameters assigned in the biophysical map are different, from the point of view of the E2E simulation there will be functionalities and interfaces that will be common between the missions.

It could be possible, in principle, to use external reference images directly as the Level-2 reference map, however this is rather complicated since a good knowledge of the scene is needed in terms of its acquisition conditions. This knowledge must be available in order to set the simulation to those same observation conditions so that the retrieved Level-2 can be efficiently compared with the reference map without the introduction of artifacts. This makes very complicated the use of external images as reference maps for the scene generator.

The implementation of the reference forward model will be very dependent on the application. For example, when the instrument is employed for atmospheric applications, the reference forward model is defined by means of running an atmospheric radiative transfer model (HITRAN, LOWTRAN, MODTRAN, libRadtran...), that can be shared by various types of missions. Since these computations may be very time consuming, it will also be possible to make use of previously-generated look-up tables.

8.2.2. ENVIROMENT SIMULATOR

The Environment Simulator translates the geolocated scene derived from the Level-2 reference map into the stimuli to be fed to the Instrument module, being reflectivities in the case of microwave instruments or radiances in the case of optical instruments.

Its generic implementation is composed of several lower-level building blocks:

- Atmosphere Simulator
- Ionosphere Simulator
- Cloud Masks
- Sunglint Masks

8.2.2.1. Scene Mapper

The Scene Mapper generates a geolocated scene taking into account the observation geometry and the illumination conditions. In addition, it can add a natural variability, randomly or based on different distributions to the biophysical parameters. Due to its nature, many functionalities in this module can be reused across several missions and instruments, for instance those functions interacting with the map or the management of possible DEMs assigned to the geographical grid.

8.2.2.1.1. Atmosphere Simulator

For optical instruments the environment is simulated by means of running an atmospheric radiative transfer model (HITRAN, LOWTRAN, MODTRAN, libRadtran...), that can be shared by various types of missions for translating the reflectance at Top Of Canopy (TOC), which is the one present in the geolocated scene map, to Top Of Atmosphere (TOA) radiances.

8.2.2.1.2. Ionosphere Simulator

The simulation of the ionosphere offer some commonalities and can be shared by different types of instruments, as it highly affects the measurements of microwave (passive or active) instruments at certain bands (i.e. P-band).

8.3. INSTRUMENT MODULE

The Instrument Module is in charge of simulating the sensor behaviour, having different outputs depending on the type of instrument. There are three main different architectures valid for this module, depending on the type of instrument: optical (with small variations for active/passive), active microwave and passive microwave.

It would be interesting to develop the instrument modules in a way that they can be run from a database of previously processed scenes, so this possibility shall be contemplated in the implementation.

8.3.1. OPTICAL INSTRUMENTS

The architecture of the Instrument Module for optical instruments is based on the identification of four main building blocks: spatial, spectral, radiometric and data pre-processing. These blocks are very much common to both active and passive optical instruments, with particular implementations for each of them as explained in the specific sections of this document. The main difference between the architecture for passive (Figure 8-7) and active (Figure 8-8) instruments is the presence of an additional Laser Transmitter block in the architecture for active optical instruments.

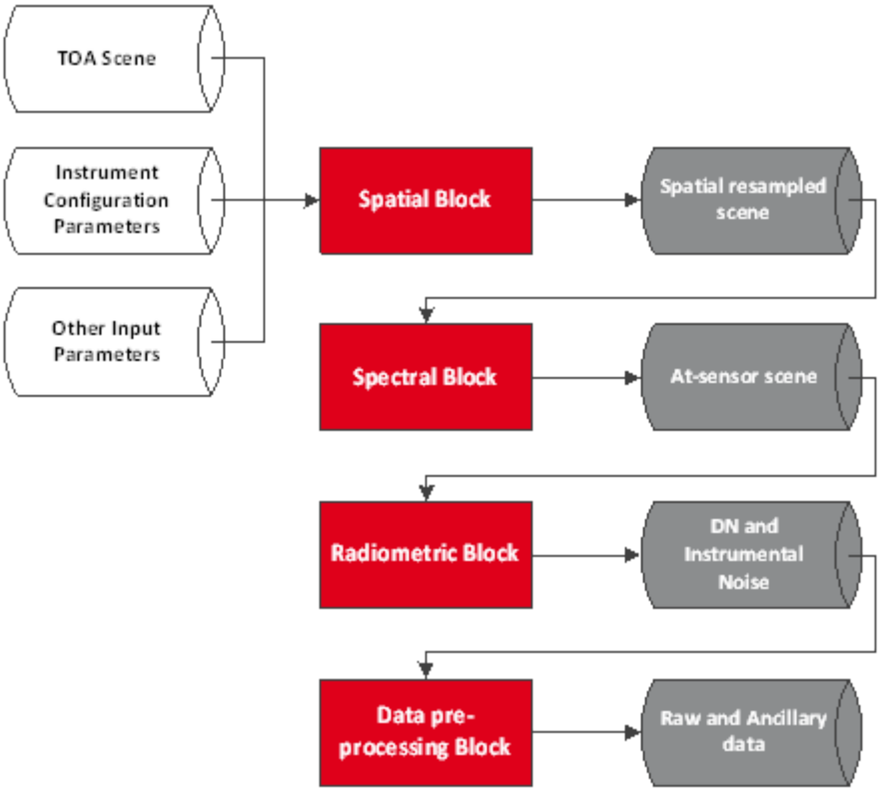


Figure 8-7: Generic architecture of the Instrument Module for Passive Optical Instruments

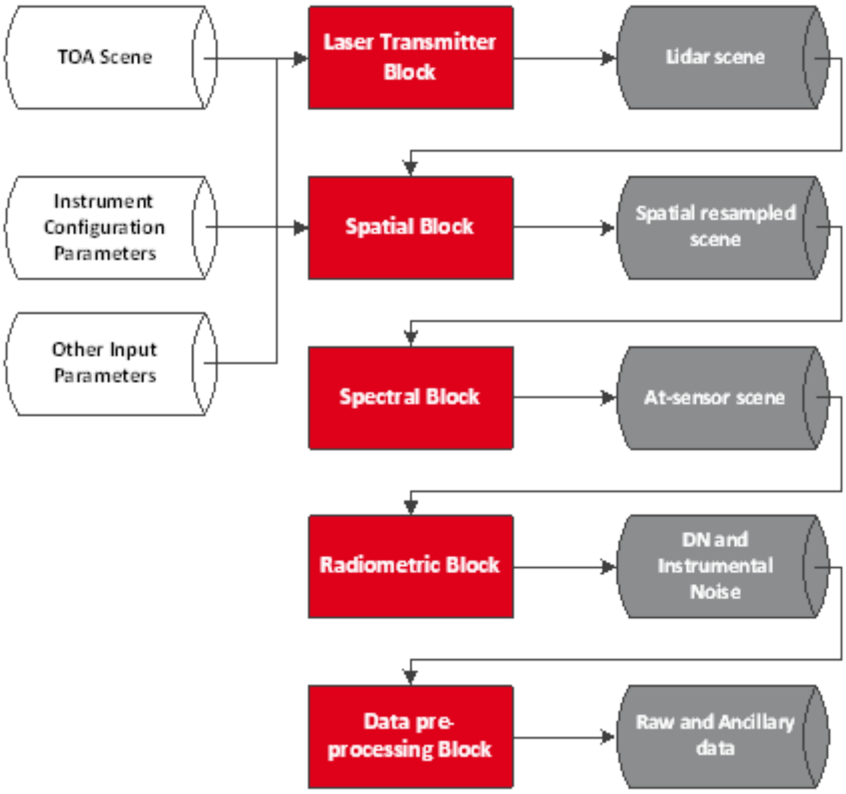


Figure 8-8: Generic architecture of the Instrument Module for Active Optical Instruments

8.3.2. MICROWAVE INSTRUMENTS

The architecture of the Instrument Module for the Active Microwave Instruments has four high-level building blocks, as shown in Figure 9-9. These building blocks are:

- Antenna Block
- Errors & Sensitivity Block
- IRF Generation Block
- Processing Chain Block

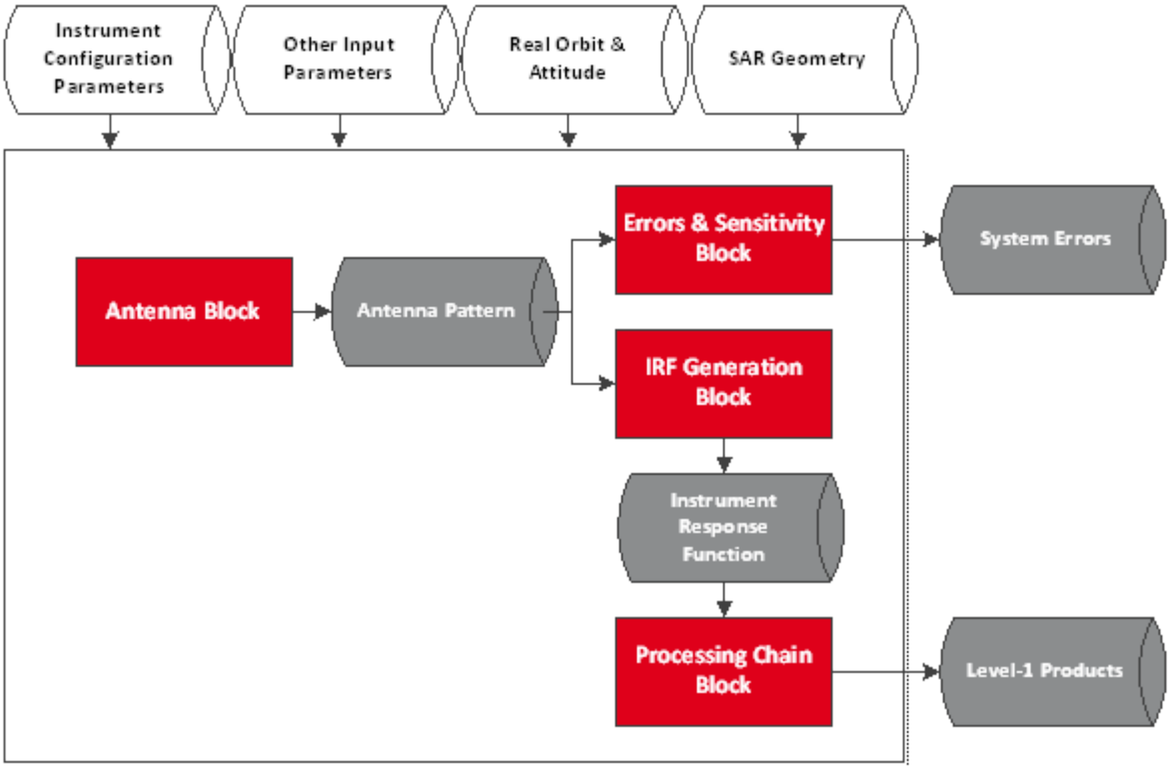


Figure 8-9: Generic architecture of the Instrument Module for Active Optical Instruments

The architecture of the Instrument Module for the Passive Microwave Instruments has three high-level building blocks, as shown in Figure 8-10. These building blocks are:

- Antenna Block
- Receiver Block
- Data pre-processing Block

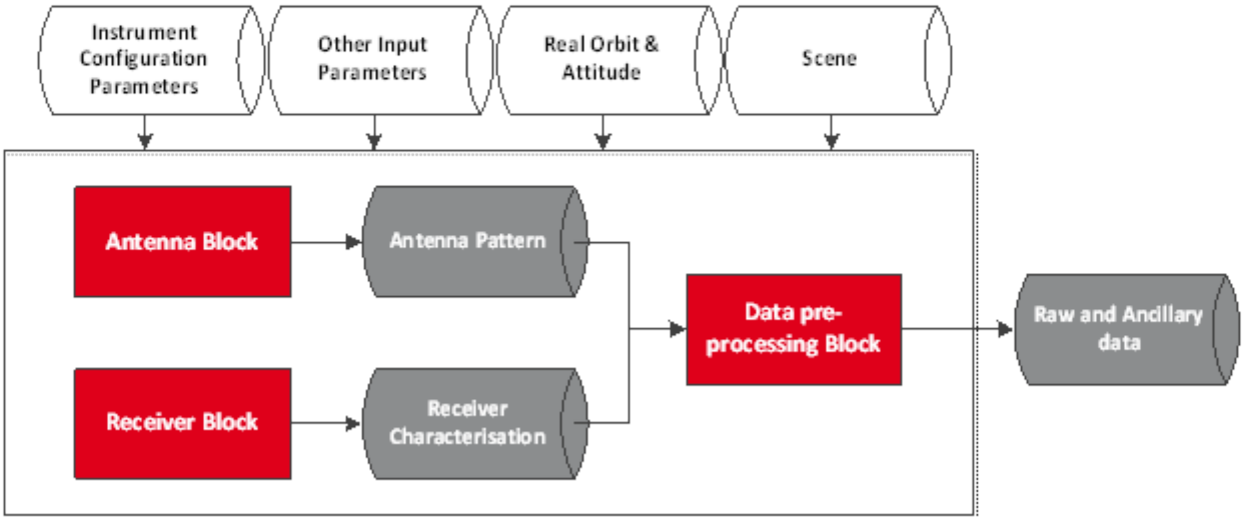


Figure 8-10: Generic architecture of the Instrument Module for Active Optical Instruments

The Antenna Block is common to both the Active and Passive Microwave Instruments. This block, as detailed in Sections 9.1.4 and 11.4.2.1, includes a building block to handle the partial failure of the antenna, mainly used in Active Microwave Instruments. While Passive Microwave Sensors rarely use an array antenna, but a lens antenna or a (rotating) reflector, this building block has been included in the Antenna Block for the sake of commonalities and will just not be used when a Passive Microwave Instrument is being simulated.

Another can be found at a lower level. There is a building block common to both Active and Passive Microwave Instruments, the Receiver Frequency Response Computation, which is contained inside the Errors & Sensitivity Block of the Active Microwaves Instruments and inside the Receiver Block of the Passive Microwave instruments.

8.4. LEVEL-1 PROCESSING MODULE

This module is specific to each type of instrument; however some common building blocks have been identified across different types of instruments. The architectures proposed in each case are presented in the specific sections further down this document.

The commonalities identified within this module are the following:

- Calibration Block present for both passive types of instruments, although with implementation that may differ.
- Determination of Calibration Parameters building block present for both passive types of instruments, although with implementations that may differ.
- Polarization Rotation: Active and Passive Microwaves.

8.5. LEVEL-2 RETRIEVAL MODULE

The retrieval algorithms are very specific to the type of instrument being simulated, and to the mission itself, therefore this module is very specific to each type of instrument. However there are three distinct building blocks that have been identified as common to the different categories of instruments, such as:

- Spatial and Timing Coregistration: performs both space and time coregistration.
- Target Identification
- Atmospheric Correction
- Feature Selection: selects subset of data from L1 and performs the required transformations to be fed to the Retrieval Algorithm block.
- Retrieval Algorithms

Figure 9-11 shows the minimum generic architecture derived for the Retrieval Module. It has to be noted that there could be additional building blocks depending on the instrument type being simulated. More details can be found in the specific sections of each type of instrument.

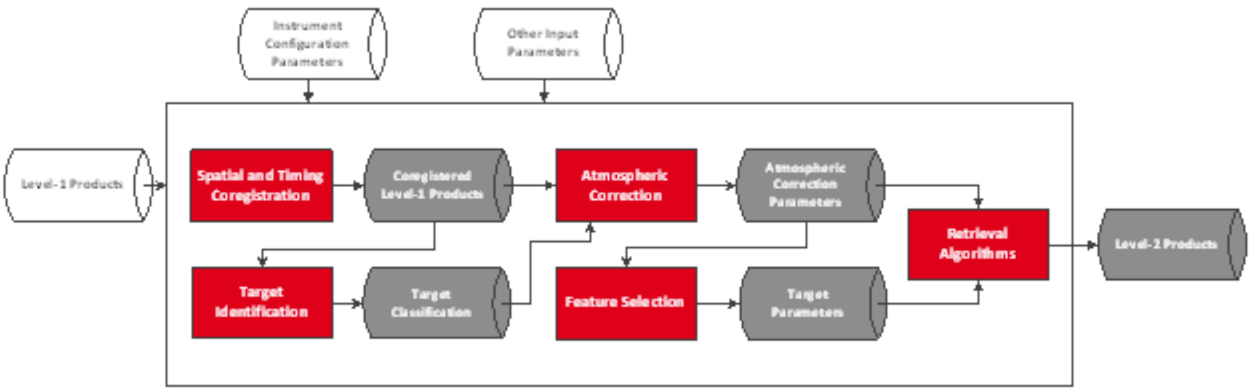


Figure 8-11: Generic architecture of the Level-2 Retrieval Module

It is important to note that the focus of ARCHEO-E2E is to identify commonalities across the different instrument types as identified in [PD.1]. This is one of the reasons why there are few commonalities at Level-2 at a lower level of detail than the proposed generic architecture for the high-level module.

However, when level 2 methods are structured by geophysical domain or application domain, important commonalities could appear which could be the basis for a domain-specific level-2 architecture (compared to the instrument-type specific Level-1 architecture).

For example, for optical instruments, the following can be identified when looking at the applications:

Land Applications

- Computation of **vegetation indices** based on atmospheric corrected data, i.e. surface reflectance, that follow a regression curve with the biophysical parameter to be retrieved. These regression curves must be "calibrated" for different observation sites, i.e. there should be ground measurements correlating the biophysical input parameter with the output vegetation index.
- **Machine Learning algorithms** trained with a large dataset of Level-1 or Level-2 synthetic data correlating the input biophysical parameters with the output TOA radiance or TOC reflectance under different atmospheric, illumination and observation conditions.
- **Reflectance/Radiance Model Inversion** by minimization of cost functions computed between acquired Level-1 (or Level-2) TOA radiance/TOC reflectance and simulated radiance/reflectance based on input biophysical parameters
- **Spectral Unmixing** (linear or non-linear) of TOC reflectance data determining composition of the imaged pixel in terms of an end-member spectral library (e.g. soil reflectance spectral library).

Ocean Applications

- As for Land Applications, computation of **spectral indices** correlated with ocean parameters (e.g. chlorophyll content in water). **Machine learning algorithms** and **Model Inversion** can be also applied in a similar way as for Land Applications.

Atmospheric Applications

- **Differential absorption techniques** (e.g. APDA, CIBR…) relating the coefficient with the amount of trace gas under study in the atmosphere. Calibrated regression curves or minimization of cost functions can be applied for the retrieval of concentration of the trace gas (e.g. Content of Water Vapor, CWV). It works with Level-1 data. This is common to both passive and active optical instruments.
  Differential absorption techniques are indeed conceptually similar to the use of spectral indices, both used in Land and Ocean Applications.
- **Atmospheric Model Inversion** retrieving the aerosol content (i.e. Aerosol Optical Thickness=AOT) based on minimization of cost functions between simulated and acquired Level-1 TOA radiance.
- **Machine Learning algorithms** trained with a large dataset of images with cloud covers. The dataset together with spectral characteristics (e.g. whiteness, pressure, Water Vapor absorption…) can be used for detection of clouds.
- **Threshold value** on the 1380 nm band for detection of cirrus clouds.

With respect to microwave instruments, apart from some generic atmospheric and ionospheric corrections, or swath edge corrections due to antenna/mirror collecting radiation from the sky (shared with the passive optical sensors), there are not many commonalities with other sensors. Actually, in many cases, the retrievals (e.g. wind speed or rain rate, are done only with simultaneous radiometric data acquired by the same sensor in the same platform) using polynomials of the different brightness temperatures at different frequencies and/or polarizations (regardless if the application is land, ocean, cryosphere or atmosphere). Usually up to order 2 at most, and or the 22 GHz channel (water vapor channel). At other bands, where absorption is much stronger, other function (natural logarithms etc) are used to better capture the non-linearities.

There is however one that comes to mind in which there are "commonalities", but actually they are not commonalities, it is just because microwave radiometers go together with another sensor: it is in the radar altimetry case, in which the radiometric data is used to compensate for the atmospheric wet delay.

This information is provided here as a guideline for a possible update of the Level-2 module architecture, but once again, we believe that this module is highly dependent on the application itself and may not be the best to generalize it.

8.6. PERFORMANCE EVALUATION MODULE (PEM)

The Performance Evaluation Module (PEM) will be present in all the E2E Simulators for Earth Observation, as it has a key role in the validation of the complete simulator and in the test campaign aiming at determining the E2E performances of the complete mission. Figure 9-12 shows the architecture of this module, common to all types of instruments.

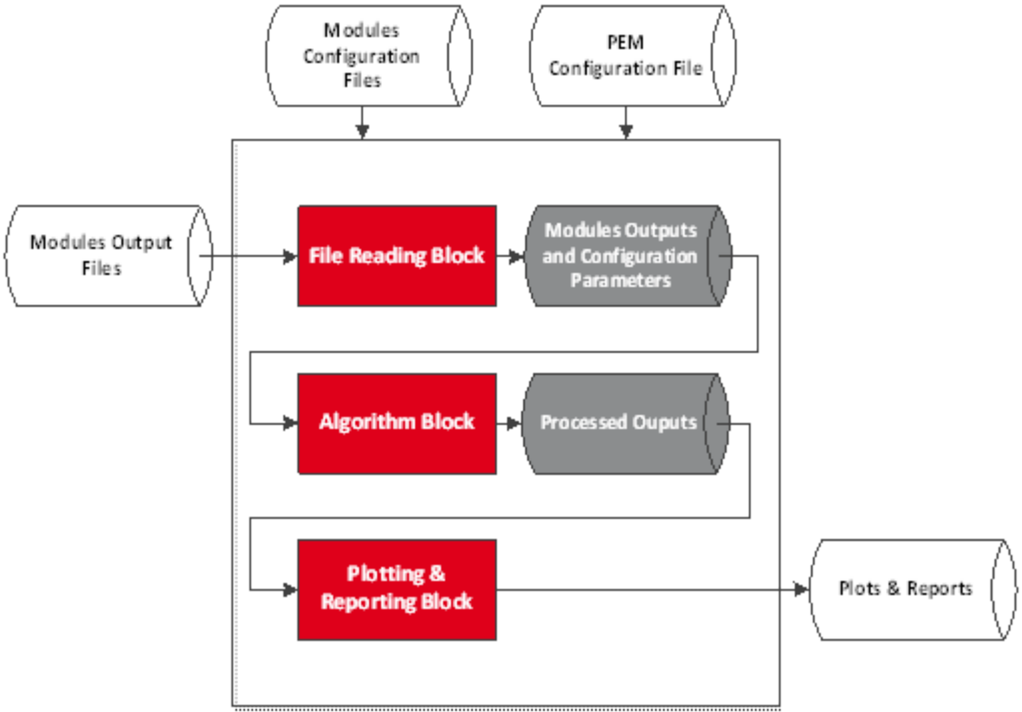


Figure 8-12: Generic architecture of the Performance Evaluation Module

The PEM has three high-level building blocks:

- File Reading Block. This BB reads the input, outputs and configuration files of the different modules in order to plot results, make comparisons, generate statistics… As it has been proposed to use a common format for all the modules and all the simulators, this File Reading Building Block could be easily reused from other simulators. There are two building blocks inside:
  - Configuration File Reader
  - NetCDF File Reader
- Algorithm Block. This BB contains all the algorithms needed to treat the input and output results to make the adequate figures, comparisons and reports. They would include statistical analyses (regressions, RMS calculations…) and also algorithms more specific of the mission (i.e., application of certain transformations to the data to allow comparisons). There are three building blocks inside:
  - Platform Performance Evaluation
  - Instrument Performance Evaluation
  - Level-1 Performance Evaluation
  - Level-2 Performance Evaluation
- Plotting & Reporting Tool/. This BB performs the visualization of the data read by the File Reading BB and probably processed by the Algorithm Library Module.

In addition to the generic architecture for the PEM presented in Figure 8-12, Figure 8-13 presents a modified architecture in which the PEM has an additional building block, the PEM Controller, which handles several simulations at once (i.e. N simulations vs. 1 simulation). This would allow comparing among several different runs in which possibly a single parameter has been modified.

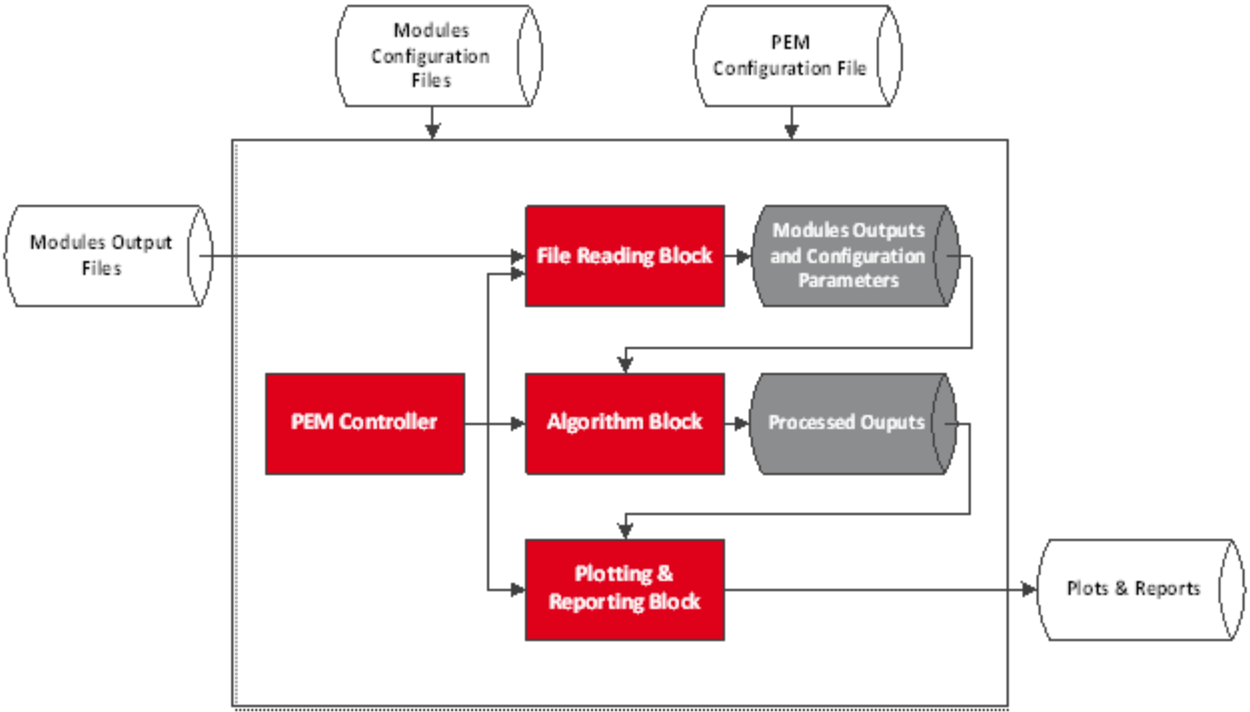


Figure 8-13: Modified generic architecture of the Performance Evaluation Module

9. REFERENCE ARCHITECTURE FOR ACTIVE MICROWAVE INSTRUMENTS

In the active microwave case, the basic elements of the system, that encompass basically data acquisition propagation and processing, can be mapped to the typical top-level architecture for an End-to-End performance simulator as schematically shown in Figure 9-1. *Note that the architecture described here does **not** conform to the recommendations in previous sections of this document*, but rather is based on a legacy approach.

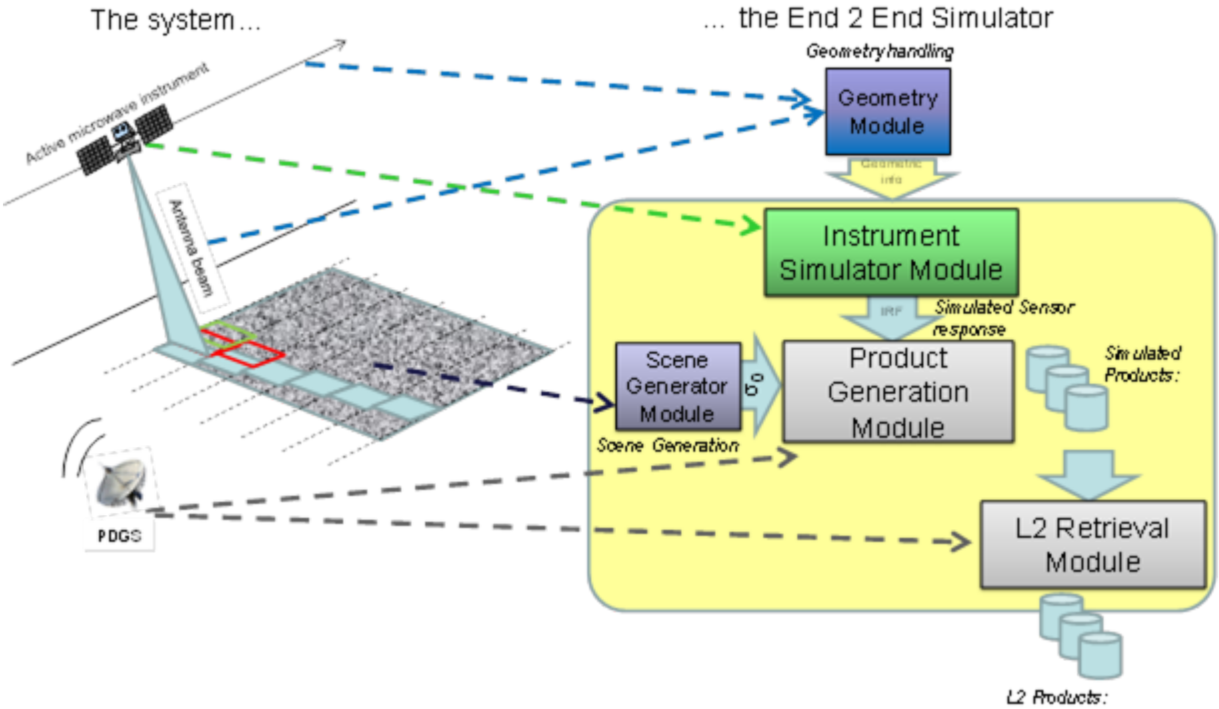


Figure 9-1: Mapping of the elements of an active microwave end-to-end system to the modules of the defined high level architecture.

Two E2ESs for active microwave missions have been already developed in the framework for Phase-A of two of the six Earth Candidate Explorer Core Missions proposed to ESA. Thus, according to the architectures there implemented [RD. 11][RD. 12], the basic architecture has been identified by mapping each element of an active microwave system to the corresponding module in an End-to-End performance simulator, as it can be seen in Figure 9-1. At least conceptually, the first module is the Geometry Module (GM), which is in charge of managing all the geometric information that are related with the orbit of the satellite, the pointing of the instrument and the altimetry of the area on ground that is simulated. The geometric information evaluated from the GM guarantees the spatial synchronization of all the other modules, i.e. the instrument simulator uses as input the pointing information provided by the GM to see towards the same area on ground that is simulated by the scene generator. The Scene Generator Module (SGM), starting from the knowledge of the biophysical characteristics and of the electromagnetic properties of the earth surface, is in charge of providing the backscattering intensity for the area of interest, namely the $\sigma 0$ maps. The Instrument Module (IM) is in charge of modeling the active microwave instrument together with its non-idealities. Thus, it gives as output the impulse response function (IRF) of the system taking into account both the sensor physical characteristics, such as the antenna pattern, and the features of the foreseen active microwave processor. The IM provides also all the parameters that describe the non-idealities and the performance of the active microwave system. The Level-1 Processing Module (L1PM) is the main module of the E2ES in the sense that it is linked to all the other modules: in fact the L1PM combines the backscattering data with the instrument model exploiting the geometric information in order to provide the Level-1 data to the Level-2 Retrieval Module (L2RM).

Eventually, the Level-2 retrieval Module implements the inversion algorithm to estimate the desired parameters related to the area on ground from the Level1 products, which are given as output by the L1PM.

As it has been previously stated, the scheme in Figure 9-1 contains the basic modules only.
As it has been discussed in [PD.1], the active microwave missions can be grouped in three main families:

- Synthetic Aperture Radar (SAR) instruments, which are coherent radar systems that generates high-resolution remote sensing imagery exploiting both the magnitude and the phase of the received echoes over successive pulses.
- Radar altimeters, that basically measure the altitude of the earth surface beneath the spacecraft by evaluating the distance between the instrument and the ground.
- Scatterometers, that basically measure the power backscatter from the surface roughness.

The following discussion on the E2ES architecture for active microwave instruments will be primarily focused on SAR imagery instruments. This choice is motivated by the fact that radar altimeter and scatterometers can be seen as a simplified system with respect to SAR instruments, in particular for what concerns the acquisition geometry. As a consequence, the E2ES architecture for radar altimeters and scatterometers will be presented as a simplification/modification of the E2ES described for SAR imagery missions. Moreover, SAR imagery missions are very rapidly increasing in the past years and several ones are planned in the next decade.

9.1. SYNTHETIC APERTURE RADAR (SAR)

9.1.1. PROPOSED ARCHITECTURE

Throughout this section, the reference architecture of an E2ES for an active microwave mission will be discussed. To describe the interactions among the different modules, in the following the reference architecture will be described exploiting data-flow diagrams, component diagrams and sequence diagrams.

In the framework of the E2ES for active microwave sensor, two possible reference architectures have been identified. The main difference between the two architectures is the execution order of the different modules, which influences also how the acquisition timing and pointing information are shared among the modules.

It is worth recalling that in SAR imagery systems, that represent the main application of active microwave sensors, the imaged area on ground in the across track direction is defined swath and it can be divided in more than one subswaths, depending on the acquisition mode. The interval in the across track direction related to each subswath is a function of the acquisition timing parameters (sampling window length, sampling window start time, rank, PRF) that are strictly related to the instrument to be simulated. Thus the GM, in order to compute all the geometric parameters related to each subswath, has to know the timing parameters together with the orbit and mechanical pointing information.

Two possible approaches, with the corresponding reference architectures, have been identified to make available the timing parameters for the GM when its execution starts, recalling that the all the other modules need the Geometric information to be executed:

1) The acquisition timing parameters are listed in the global configuration file of the E2ES
2) The acquisition timing parameters are passed to the GM as an input data by the IM

The first architecture is named as “GM first” and it is depicted in the data-flow diagram below. It is worth underlining here that in the diagrams depicted throughout this section the interfaces between some modules have been simplified in order to improve the clarity of the diagrams. In this architecture, the GM is the first module in the chain and the timing parameters are assumed to be defined in the global configuration of the E2ES by the user and those parameters are there shared between the IM, since those are parameters strictly related to the instrument, and the GM. The main drawback of this approach is that a relevant number of parameters (sampling window length, sampling window start time, rank, PRF for each subswath to be simulated) has to be written in the global configuration by the user, taking care that these are very important parameters which respect to the instrument is expected to be very sensitive.

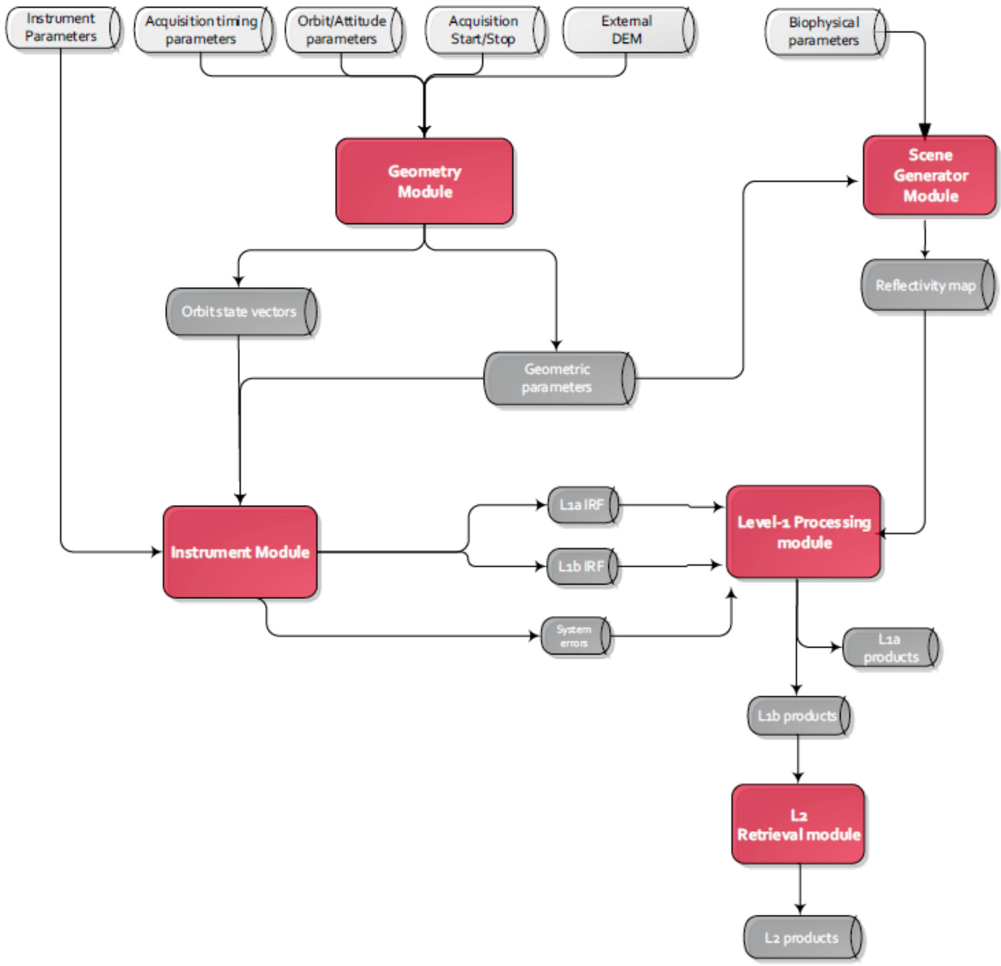


Figure 9-2: Data-flow diagram for the "GM first" reference architecture

The second architecture is named as "IM first" and it is depicted in the flow diagram below. In this architecture, the IM is the first module in the chain and it has to be executed twice. When the IM is called the first time, only the "Timing & pointing set" building block is executed and the timing parameters together with the mechanical pointing information are produced as the only output data. Those data are then read by the GM that is here the second module in the E2ES chain. When the IM is executed the second time, the other building blocks will be executed so that, exploiting the Geometric information, the IRFs and the errors will be computed.

The main advantage of this approach is that the timing parameters are managed at IM-level, as it reasonably expected since they are strictly related with the instrument to be simulated, and they are passed as data to the GM, without including all these values in the global configuration file. The drawback is that a module, namely the IM, has to be executed twice in the chain and this approach can be not very clear in a sequential chain.

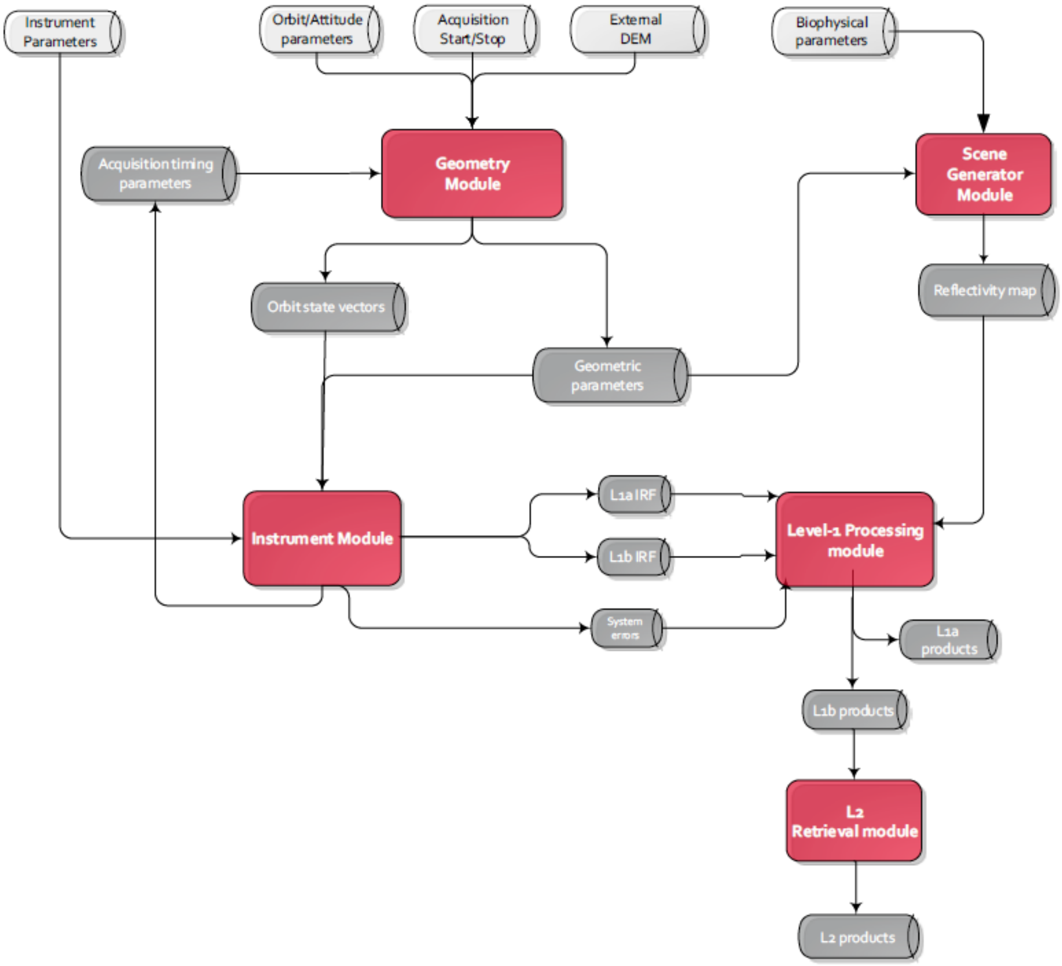


Figure 9-3: Data-flow diagram for the "IM first" reference architecture

The comparison between the component diagrams for reference architecture "GM first" and "IM first" is shown in the following figure. There, the additional interface between the IM and the GM can be noticed for the "IM first" reference architecture.

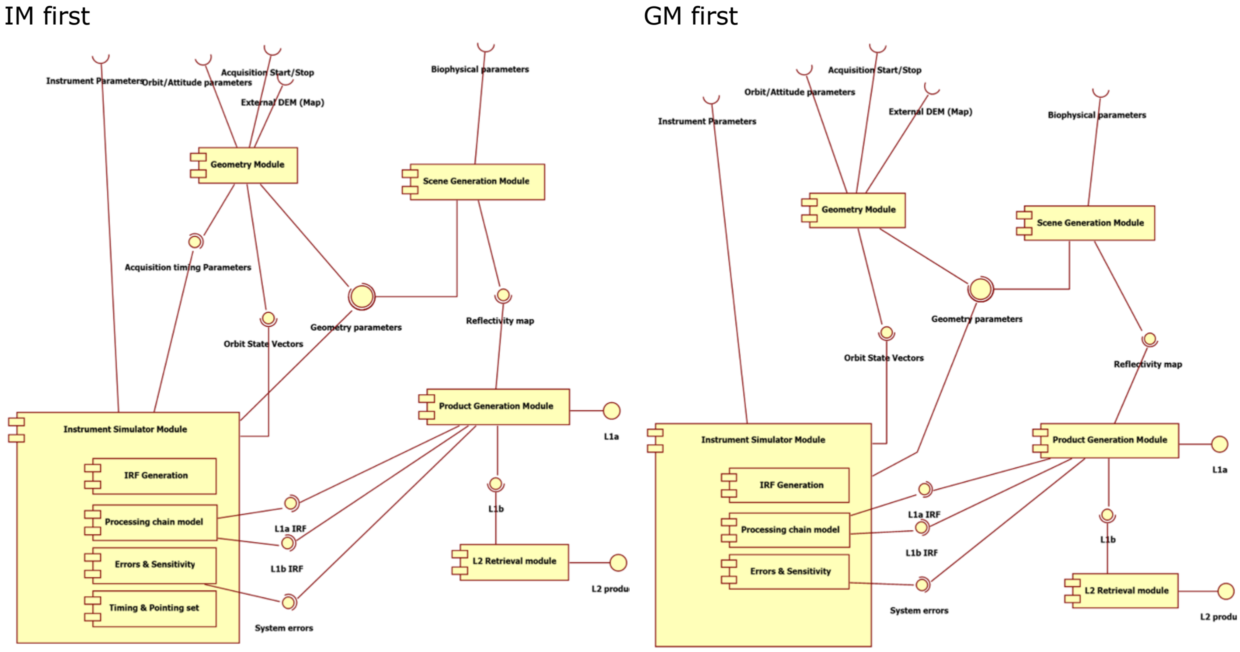


Figure 9-4: Component diagram for the "IM first" reference architecture, on the left, and for the "GM first" reference architecture, on the right.

It is worth underlining that the "IM first" reference architecture allow also to easily solve a problem which is related to the convolution to be performed in the L1PM to create the L1 products. Recalling that the main operation to be performed in the L1PM is the combination of the reflectivity maps with the IRF provided by the IM and that this operation consists of the convolution between those data, the convolution will be performed between two time limited signals. As a consequence, considering as valid output of the convolution only the samples computed when the whole IRF is fully overlapped to the reflectivity data, the data matrix obtained as result of the convolution will be unavoidably smaller than the reflectivity map. Cutting away data in the across track direction can be dangerous in case of acquisition mode that uses more subswaths, since holes with no data can be created between successive subswaths. In order to avoid this problem, in the "IM first" reference architecture the IM, when it is executed the first time, can pass the IRF length to the GM so that the Geometric information related to the area on ground are enlarged of the length of the IRF in each direction. This way, even if the boundary data are cut away when the convolution is computed, the size of each subswath after the convolution will be equal to the expected and the holes between successive subswaths will be avoided.

Other two possible approaches to manage the sharing of information related to the acquisition timing parameters between the IM and the GM have been identified. As a first alternative approach, these parameters could be passed as configuration parameters to both the GM and the IM, but not allowing a non-expert user to access them. In fact, the performance of the instrument are heavily dependent and very sensitive to PRF, SWL, SWST and rank, so that changing them without the needed care and expertise can lead to meaningless simulation results. As a second alternative approach, the Timing&Pointing Set building block could be replaced by a Radar Parameter Generator (RL1PM) Module, as it is foreseen in the BIOMASS E2ES. This BB is aimed at converting the user defined acquisition geometry in terms of swath, number of subswaths, etc. into a set of optimized SAR configuration parameters (PRF, SWL, SWST, rank) that are passed as input to both the GM and the IM. In its basic model, the L1PM can be implemented simply by software that copies the configuration parameters (PRF, SWL, SWST, rank), that are visible only to the RL1PM itself, in the output files to be passed to the GM and the IM. In long term, the L1PM can be made more complex to be able to compute the SAR acquisition parameters starting from high level parameters given by the user such as the swath size, the resolution, etc.

In conclusion, the "GM first" will be considered as the baseline reference architecture in the following in order to share the same architecture with other instrument. In fact, the "GM first" is almost identical to the reference architecture described in Section 4. With respect to the reference architecture depicted in Fig. 4-4, the only difference is that the data flow for Active Microwave foresees that the SGM feeds the L1PM instead of the IM. This is required since the L1PM is in charge of mixing the instrument information that is provided by the IM with the backscatter map that is provided by the SGM.

Moreover, in the framework of the reference architecture definition, to highlight the data flow from the various modules in the E2ES chain towards the Performance Evaluation Module, the flow diagram in the following figure is here presented. There it can observed as the PEM receives as input the IRFs, the L1 products and the L2 products to compare them with the reference data obtained from the IM and the SGM (data to be compared are highlighted using arrows with the same color).

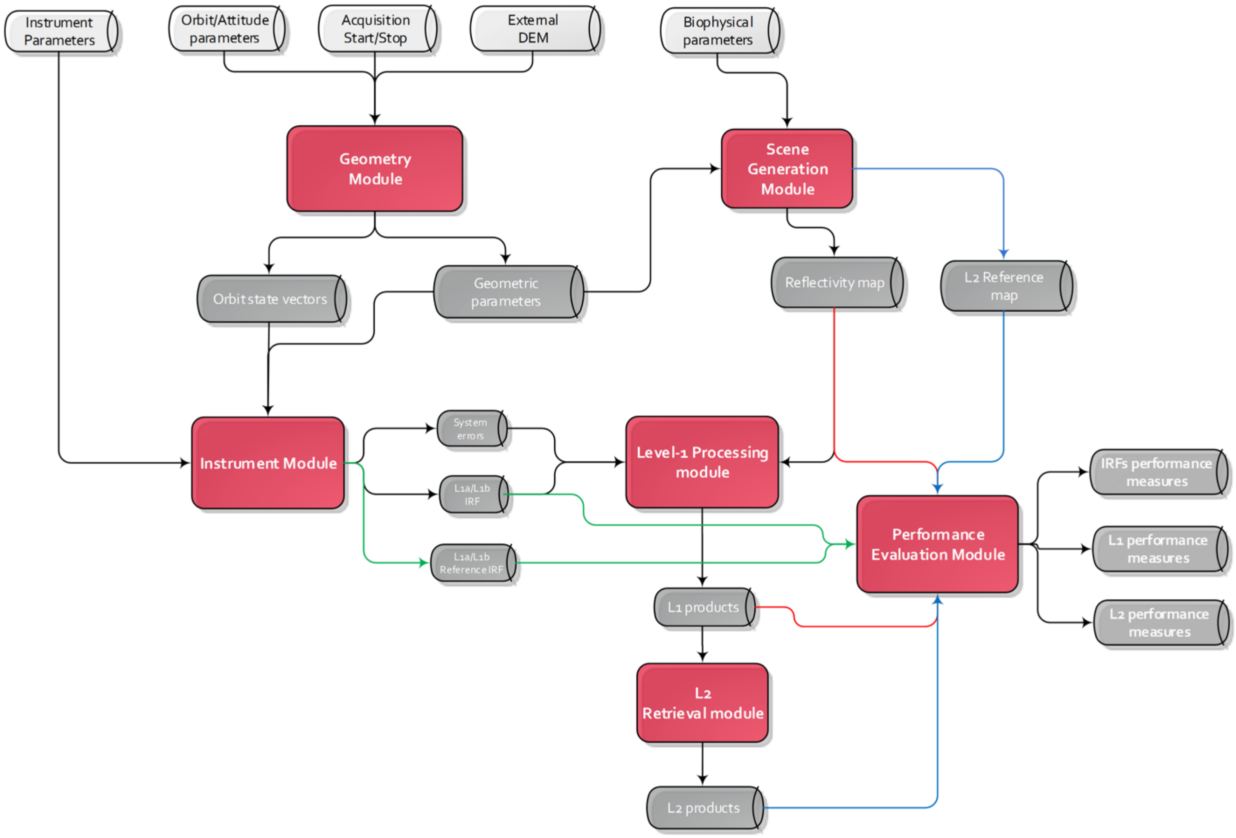


Figure 9-5: Data-flow diagram for the reference architecture including the PEM

Finally, foreseeing the evolution of the E2ES in successive phases of the mission, while the architectures presented above are more appropriate for the preliminary Phases, a different architecture can be considered in case of an E2ES for the later Phases. In fact, in the later Phases, it can be interesting to test the L1 focuser and to have also the possibility to inspect the raw data before the L1 focusing. As a consequence of the different requirements for the later Phases, the data-flow diagram for the reference architecture can be modeled as in the following figure.

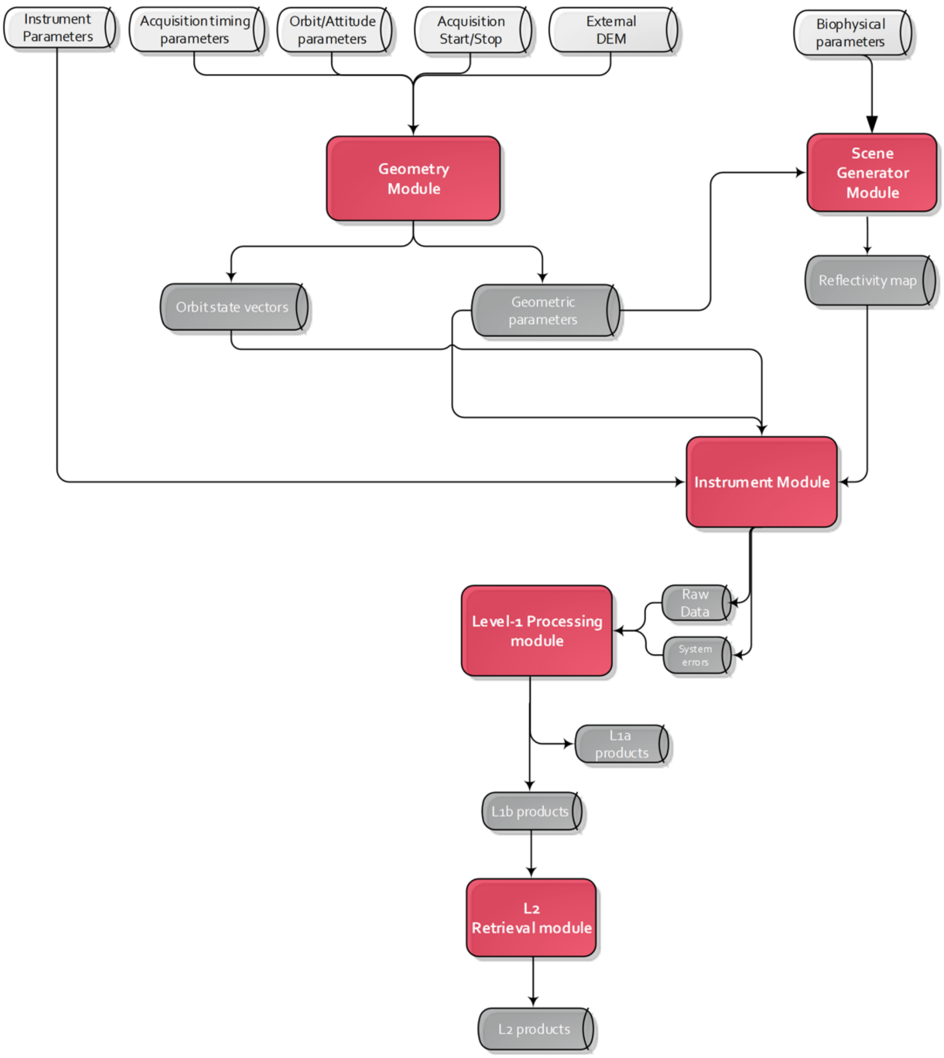


Figure 9-6: Data-flow diagram for the reference architecture for later Phases

In the figure above, it can be noticed that:

1. The SGM passes its output to the IM
2. The IM starting from the Reflectivity Map compute the Raw data
3. The L1PM basically apply the SAR focusing to the Raw data and then it applies the System errors

The modifications to the Modules introduced by the reference architecture described above have been detailed in Section 9.1.8.

9.1.1.1. Extensive and evolutionary growth of the reference architecture

As an example of the extensive and evolutionary growth of the reference architecture for active microwave missions, the example of the atmospheric effects is here discussed. In fact, the atmospheric module has been included in the hierarchical view of the building blocks but it has not been included in the basic set of modules.

In an active microwave mission, the atmospheric effects could be not included in a first phase of study of the system but they have to be taken into account when an increased accuracy for the simulation results is needed. This is due to the fact that the impact of the atmosphere at the frequencies usually adopted for the active microwave instruments can be not so high when a feasibility study is performed. However these considerations are strictly application dependent.

In case that the atmospheric module is introduced in the reference architecture previously defined, the data-flow diagram is updated as in the figure below.

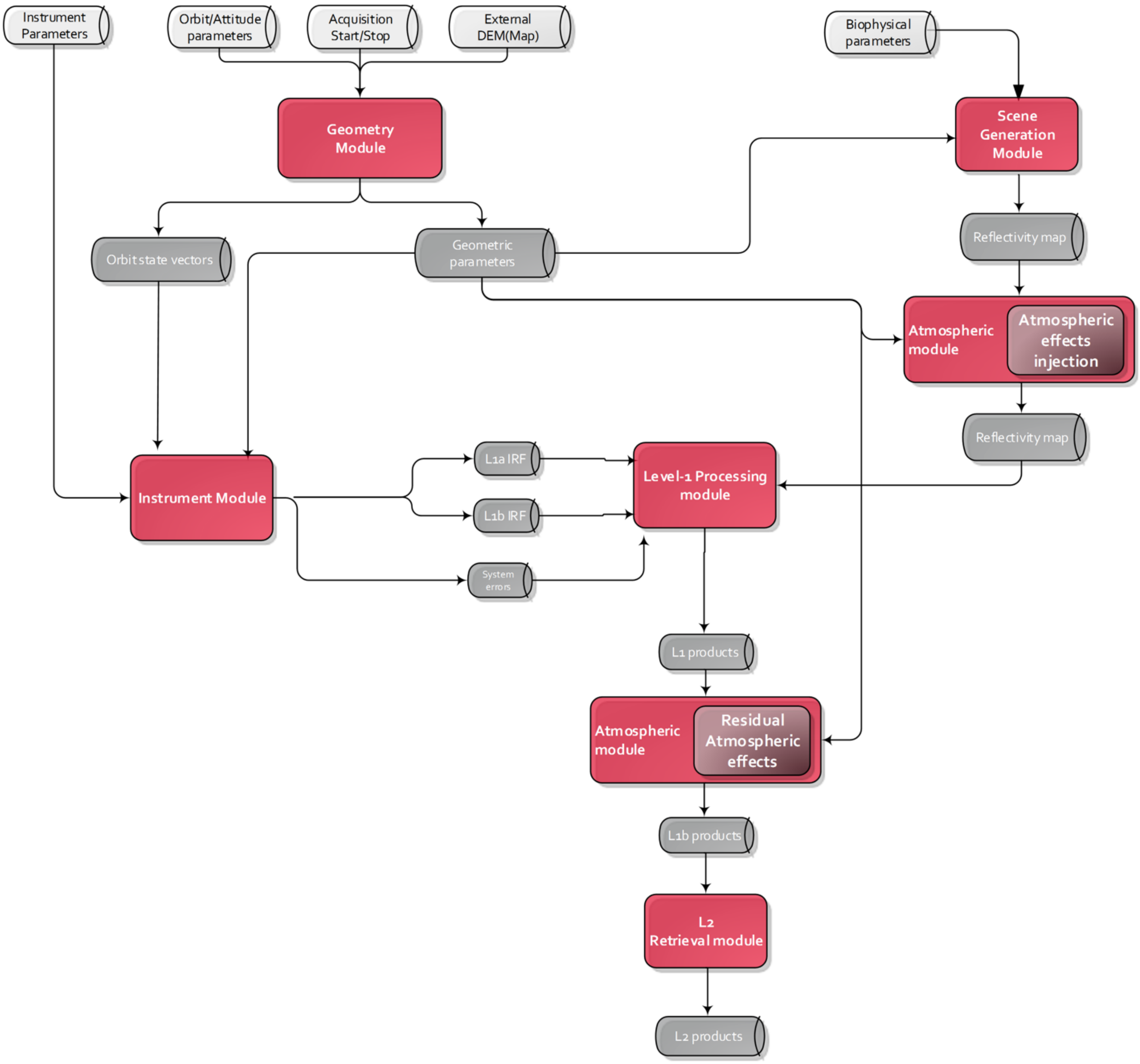


Figure 9-7: Data-flow diagram for the reference architecture including residual atmospheric effects injection

It can be noticed that the atmospheric module has been interposed between the L1PM and the L2RM. The included module is designed to add to the L1 products only the residual atmospheric effects after the atmosphere compensation. The extension of the architecture by adding a module can be done without impacting on the other modules only if the additional module, in this case the atmospheric module, does not change the input interface of the L2RM. In other words, assuming that the L2RM expect the L1 product with a given filename and/or in a given folder, the filename and the path have not to be changed by the atmospheric module.

In conclusion, by adding the residual atmospheric effects as described above, the E2ES moves towards a more complete architecture along the vertical axis in Figure 3-1.

However, in case that a higher level of accuracy is needed in the simulation results, the E2ES architecture can be extended again evolving at the same time by including more accurate models. In fact, instead of injecting to the L1 products the residual atmospheric effects, another possible (and more realistic) approach is to add the atmospheric effects on the reflectivity map and then to perform the atmospheric effects removal from the L1b products. This more accurate and more complete architecture is depicted in the data-flow diagram below. Also in this case, the atmospheric module has to be designed to not change the interface between then basic modules (the SGM and the L1PM for the injection, the L1PM and the L2RM for the removal) in order to be interposed without affecting the other modules.

In conclusion, by adding the injection and the removal of the atmospheric effects as described above, the E2ES moves at the same time towards a more complete architecture along the vertical axis and towards a more accurate architecture in Figure 9-8.

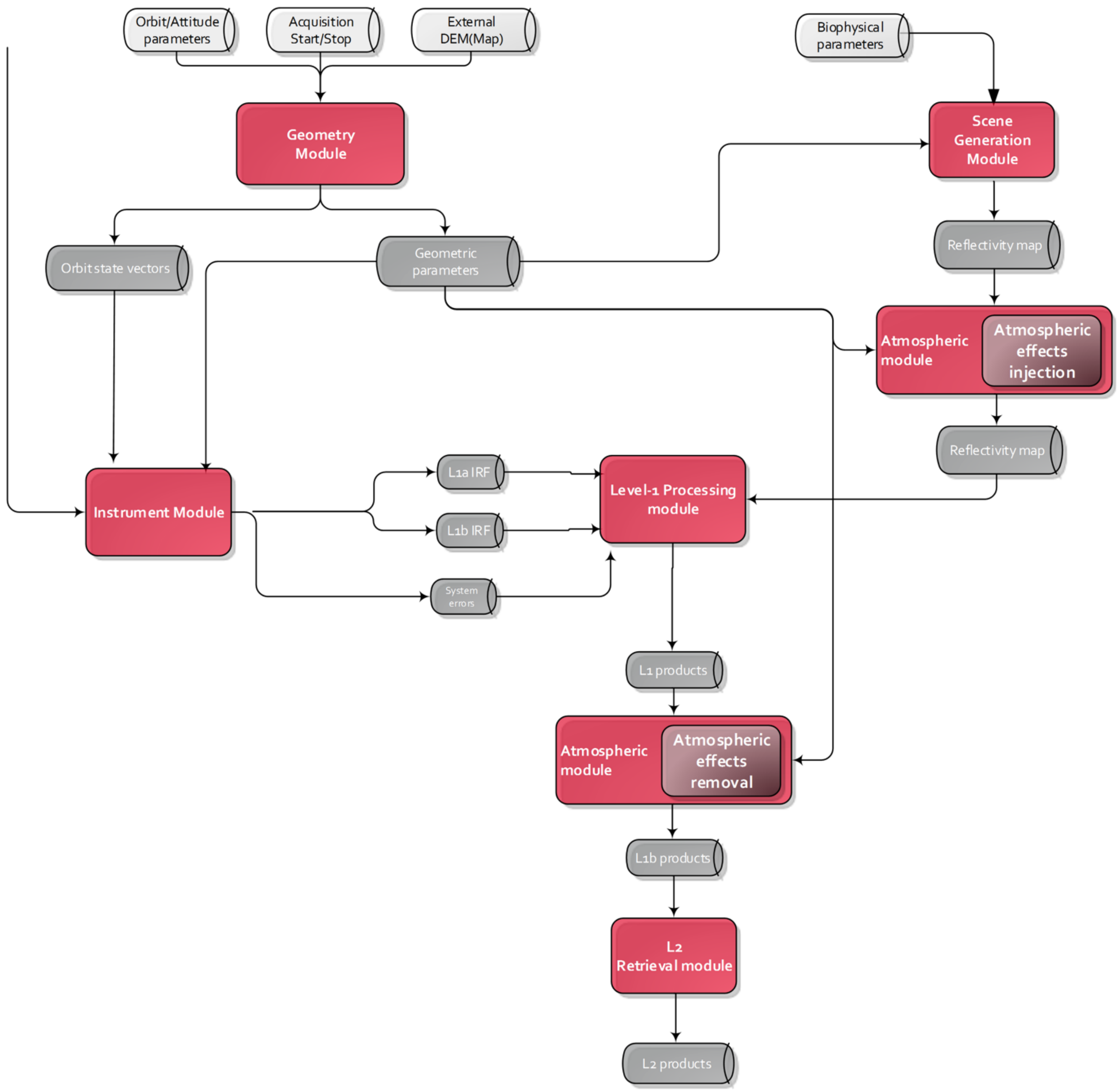


Figure 9-8: Data-flow diagram for the reference architecture including atmospheric effects injection and atmospheric effects removal

The component diagrams for the two possible reference architecture that have been discussed above are depicted in the following figure. There it can be noticed how the interfaces between the modules are not changed by interposing the atmospheric effects modules.

Residual atmospheric effects injection

Atmospheric effects injection and removal

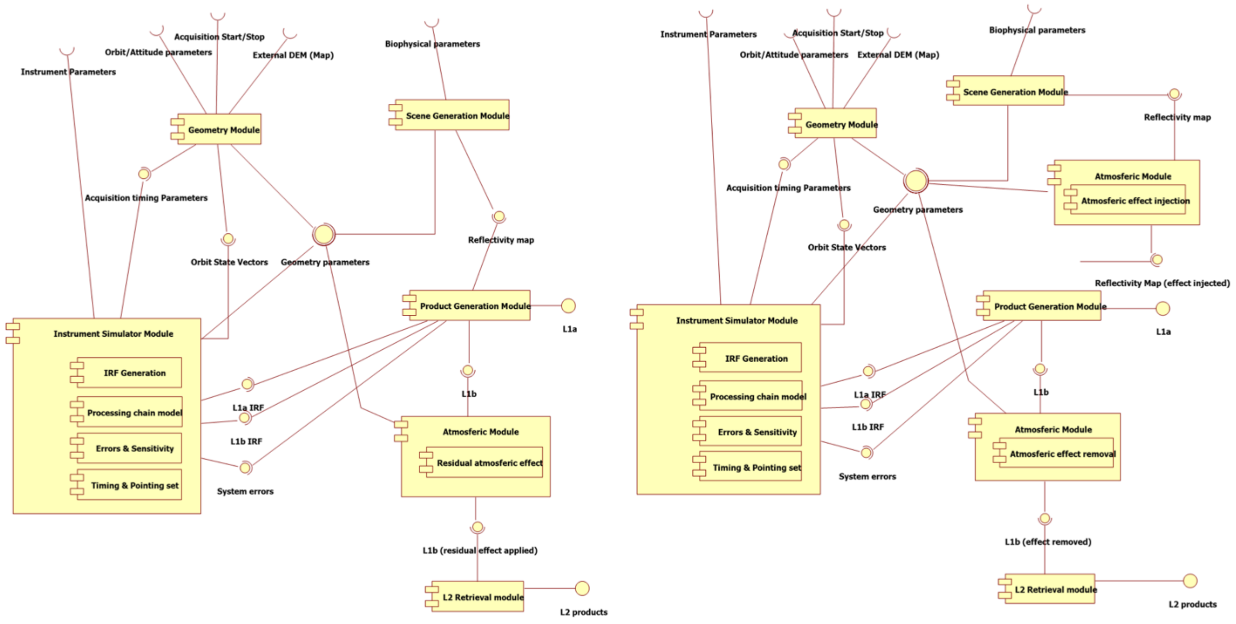


Figure 9-9: Component diagram for the reference architecture including residual atmospheric effects injection, on the left, and for the reference architecture including atmospheric effects injection and atmospheric effects removal, on the right.

The discussions drawn above can be extended within the reference architecture for active microwave E2ES from module-level to the building block-level. In fact, it is worth highlighting that the extensive and evolutionary growth is not limited at module-level: additional and/or more accurate building blocks can be included within a module.

On the other hand, within the classical processing chain for the Level1 data and Level2 data for active microwave missions there are several processing blocks aimed at removing or attenuating the physical effects that affect the quality of the data. As an example, the amplitude shaping of the elevation antenna pattern is removed on the data or, in case of ScanSAR acquisition mode, the scalloping effect is removed exploiting proper algorithms.

Thus, when a removal block is foreseen in the processing chain for the ground processors, in the E2ES block chain two possible approaches can be followed:

1) Residual effect injection
2) Effect injection and effect removal

The choice between these two approaches is reasonably driven by the phase of the mission for which the E2ES has to be designed. In fact, in case that the priority is to have short execution time for the E2ES aiming at verifying the feasibility of the core system as in the first phases of a mission, the residual effect injection could be a better choice. On the other hand, moving towards the latest phases when the accuracy is a fundamental requirement and when it is interesting to test the removal algorithms that will be actually implemented, the second approach is more reasonable. A comparison of the two approaches is presented in the following table.

Table 9-1 Residual effect injection versus effect injection and effect removal

Approach	PROs	CONs
Residual effect injection	• Less computation burden • Simpler architecture	• Lower accuracy • Requires assumptions to model the residual effects
Effect injection and effect removal	• Higher accuracy • The performance of the removal algorithm can be tested • The performance of different removal algorithms can be compared	• Higher computational burden • More complex architecture

It is worth recalling here that the possibility to switch from an approach to the other during different phases of the active microwave mission and, as a consequence, in different stages of the E2ES requires that the effect injection does not modify the interface between the previous and the following block. This design strategy allows interposing the effect injection block without affecting the other blocks. On the other hand, the residual effect injection block and the removal effect block have to share the same interface in order to use alternatively one or the other.

9.1.1.2. Single subswath versus multiple subswath acquisition mode

According to TN-E2E-MC, one of the driving features that determine the simulation architecture definition for active microwave missions is the acquisition mode.

The acquisition modes for active microwave missions and in particular for SAR imagery missions can be grouped in two main categories: the single swath acquisition modes (i.e. Stripmap/Wave, Spotlight) and the multiple subswath acquisition modes (i.e. ScanSAR, TOPSAR).

The single swath acquisition mode assumes that the echoes coming from the whole across track direction are acquired at the same time taking fixed the antenna beam pointing. As a consequence, the whole swath data are processed together up to Level1b product.

The multiple subswath acquisition mode assumes that the acquisition is performed in a burst-by-burst basis and acquiring at the same time the echoes coming from an interval in the across track direction, namely the subswath, so that dynamic elevation pointing is required to switch to the successive subswath. As a consequence, the per-subswath data are processed together and then, in the Level1 processing chain, the data from different subswath are merged to have the full swath Level1b products.

The single swath acquisition mode and the multiple swath acquisition mode are depicted in the following figure.

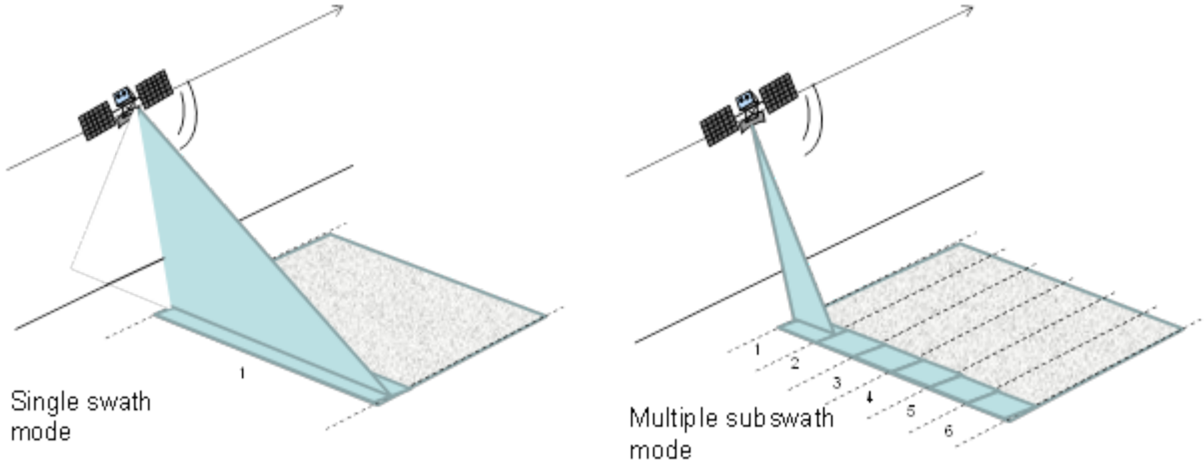


Figure 9-10: Single swath acquisition mode and multiple swath acquisition mode

In the framework of the reference architecture definition for SAR imagery E2ES, from the simulation architecture point of view, a multiple subswath acquisition mode can be seen, up to a certain level, as the repetition for the different subswaths of the same processing defined for a single swath acquisition mode.

In our view, the best approach to manage the repetition of the functionalities is to realize the loop through the subswaths at building block level within each module. The rationale behind this choice is that, within each module, there are some building blocks that are run once in both the single swath and the multiple subswaths cases. Moreover, it is worth noticing that some building blocks will be unavoidably different in the two cases, since they have to implement different models depending on the acquisition mode. An overview of the reuse of the building block for each module is given in the following table in case of single swath acquisition and multiple swath acquisition.

Table 9-2 Reuse of the building blocks in single swath acquisition mode and multiple subswaths acquisition mode

Module	Custom building block for acquisition mode	Reused building block to be run once in single swath and multiple subswaths	Reused building block to be run one time for each subswath
GM		- Orbit Simulator - Attitude Simulator - AOCS/Instrument coupling - SAR2GEO BB	- GEO2SAR BB - Shadowing mask - Range ambiguities
SGM		- L2 Imaging area - Imaging area	- L2 Ambiguous area - Ambiguous area
IM	- IRF generation - Errors & sensitivity	- Antenna	
L1PM	- Swath merging - Scalloping injection	- Projection	- GEO2SAR projection - Shadowing mask application - L1 generation module - Errors injection - Polarization rotation
L2RM		- Coregistration block - Retrieval Algorithms - Atmospheric Correction - Target separation	
PEM		- L1 Performance Evaluation - L2 Performance Evaluation	- Instrument Performance Evaluation

By inspection of the table above, it can be observed that for different acquisition modes a wide range of building blocks can be reused in different missions involving active microwave missions even if different acquisition modes are applied. In fact it can be noticed that, while the building blocks in the IM that are strictly related to the istrument so that it is needed to implement specific models for each acquisition mode, a wide range of building blocks in all the other modules can be reused in E2ES for different acquisition modes.

Some additional details are needed regarding the choice of executing within the GM once the SAR2GEO block and a time for each subswath the GEO2SAR block in case of multiple subswaths acquisition mode.

It is worth recalling here the definition of the output of those building blocks as it has been given before:

- **GEO2SAR:** this building block will compute the matrix that gives the correspondence between SAR coordinates and geographic coordinates and it is defined in SAR coordinates. It is used to project data from geographic coordinates to SAR coordinates. This building block will compute also the other geometric informations (i.e. elevation angle, incidence angle) in SAR coordinates. The output of this BB is a GEO2SAR product for each subswath.
- **SAR2GEO:** this building block will compute the matrix that gives the correspondence between geographic coordinates and the SAR coordinates and it is defined in geographic coordinates. It is used to project data from SAR coordinates to geographic coordinates. This building block will compute also the other geometric informations (i.e. elevation angle, incidence angle) in geographic coordinates. The output of this BB is only one SAR2GEO product for the whole swath.

Thus, in case of multiple subswaths acquisition mode, we will have a GEO2SAR product for each subswaths and one SAR2GEO product for the whole swath. The rationale behind this approach is that, since the SAR domain is acquired subswath-by-subswath, the geometric information is defined on the same basis.

Moreover this approach has also an advantage in terms of simulation chain: when the reflectivity provided by the SGM are projected from geographic coordinates to SAR coordinates, exploiting the GEO2SAR matrices, the reflectivity data related to the current subswath can be automatically selected and projected to SAR coordinates, allowing the subswath-by-subwath processing for building blocks that follow in the chain.

The SAR2GEO matrices are used to project the full swath, after swath merging, and only one product for the full swath is expected. This way, all the geometric data defined in geographic coordinate will be referred to the full swath area on ground. Since the SGM is expected to work in geographic coordinates, as it has been also discussed in section 9.1.1.1, one reflectivity maps will be produced for the full swath. The advantage of this approach is that, after the projection of Level 1 data on geographic coordinates, the reflectivity map computed by the SGM and the GTC projected Level 1 products will be defined on the same geographic grid, so that they can be immediately compared in a pixel-by-pixel basis. The proposed approach has been sketched in the following figure.

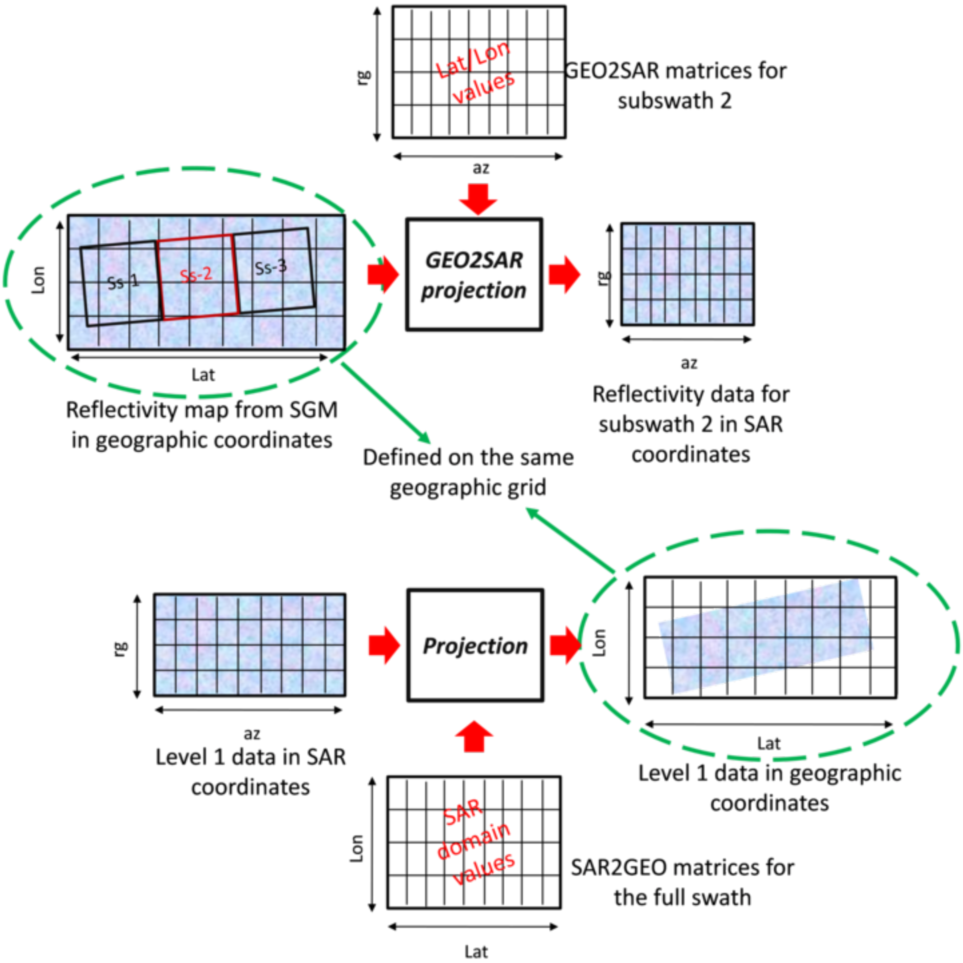


Figure 9-11: Proposed approach for GEO2SAR and SAR2GEO

The wide reuse of building blocks that has been discussed above relies on the assumption that it has to be managed a loop through the subswaths in the execution of the building blocks (or of a chain of building blocks) within the modules. This can be verified by comparison of the sequence diagrams for the single swath acquisition architecture and multiple subswaths acquisition architecture, which are represented in Figure 9-12 and in Figure 9-13, respectively.

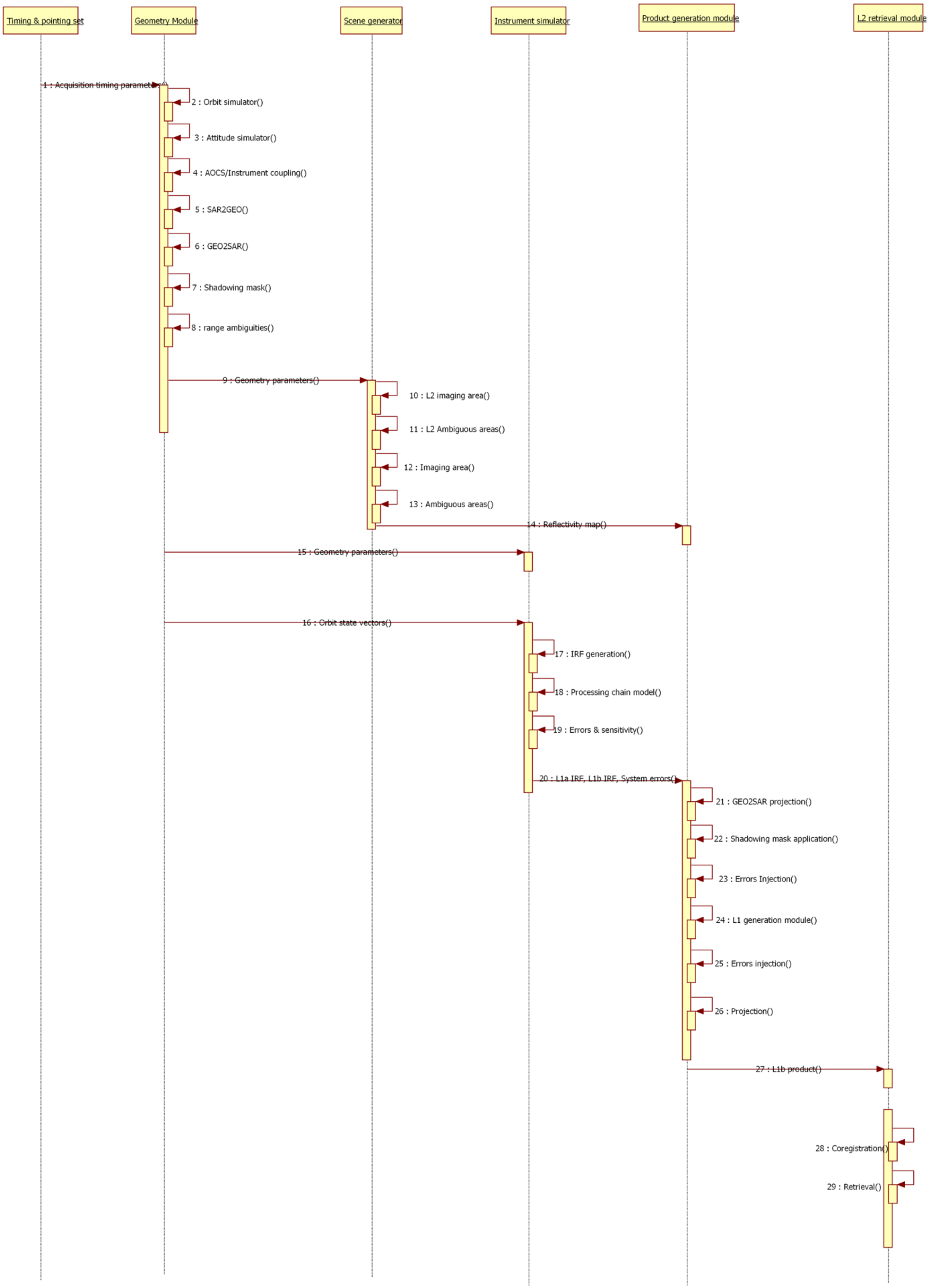


Figure 9-12: Sequence diagram for the reference architecture in case of single swath acquisition

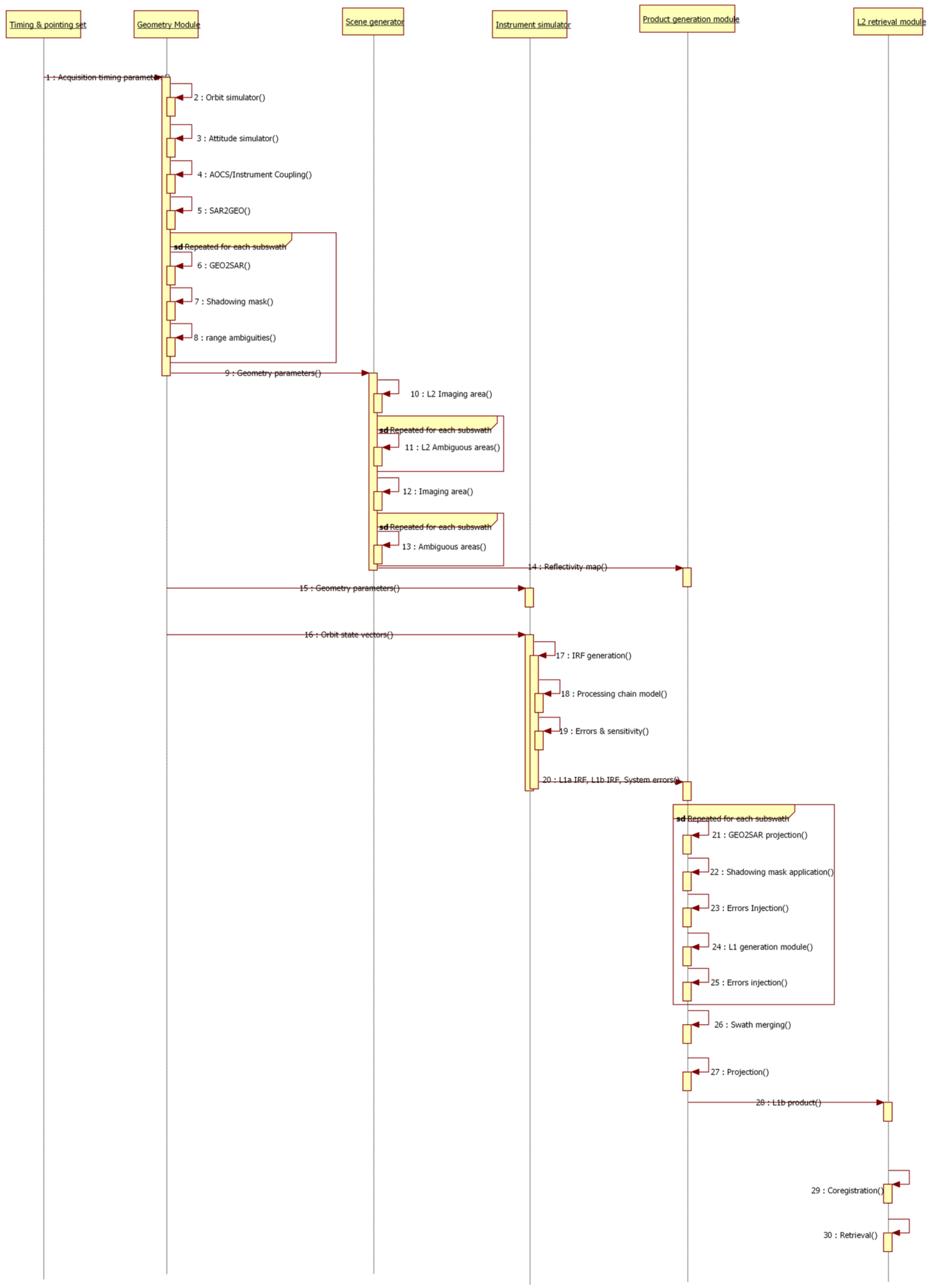


Figure 9-13: Sequence diagram for the reference architecture in case of multiple subswaths acquisition

9.1.1.3. Approach to the Modules description

In this section, as first the hierarchical construction of the reference architecture will be given describing how each module can be decomposed in building blocks. Then, for each module, the input and output interfaces will be briefly presented in order to describe the data flow between the whole architecture. Moreover, the inner architecture of each module will be also presented exploiting a data-flow diagram in order to describe the interaction among the building blocks within the module.

In the definition of the reference architecture, a fundamental step is the definition of the level of granularity at which the modules to simulate active microwave mission should be provided. The diagram represented in Figure 9-14 gives a more detailed view of the blocks and of the functionality that they provide.

In Figure 9-14, the modules are identified by the red boxes and each module has been decomposed in first level of building blocks (gray boxes) on a functional basis. In other words each first level building block is related to a functionality that has to be provided by the module, taking into account that different functionalities can be comprised in a single module, i.e. for the IM the IRF generation and instrument error computation. Some first level building blocks required to be detailed by a second level of building block (white boxes) on a model basis. In fact the idea is to decompose each functional block in some lower level building block that implements a particular model for the effect to be taken into account, as it can be seen as an example for the Error&Sensitivity building block. This approach would allow performing simulations under a different combination of models, each of them related to a different effect to be modeled.

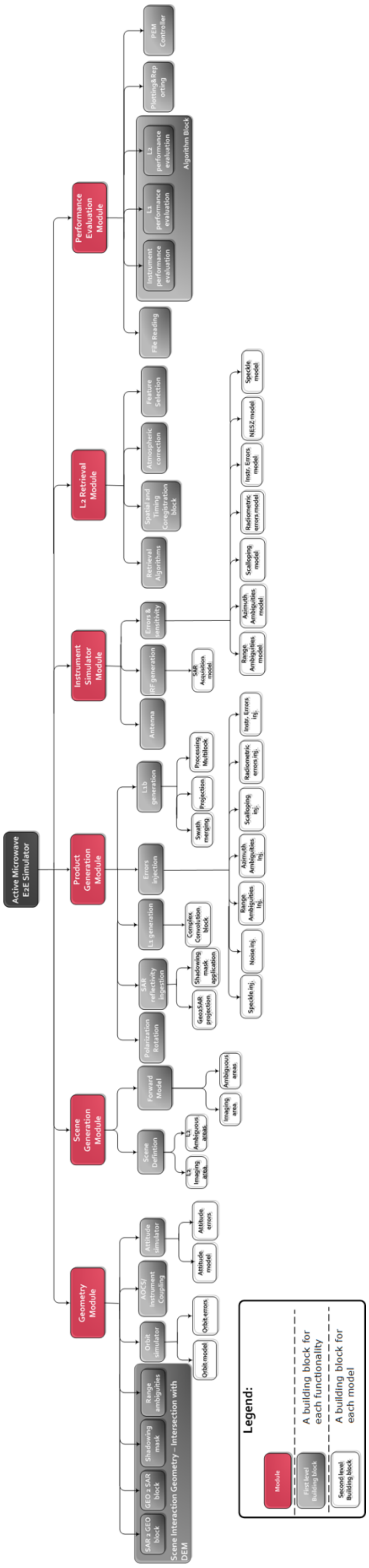


Figure 9-14: Hierarchical view of the active microwave E2ES reference architecture

The diagram in Figure 9-14 shows all the building blocks that can be used in the reference architecture for an active microwave E2ES, highlighting the inner hierarchy of building blocks within each module.

Firstly, it is worth describing the classical five blocks in the basic set configuration (without considering additional blocks going for an extensive growth of the E2ES) that compose the reference architecture for active microwave (i.e. SAR) missions. In the Table below, for each module it is given a brief description of the main functionalitites offered, moreover the needed input data and the produced output data are there listed.

Table 9-3 Description of the modules in the basic set of the reference architecture with input and output interfaces

Module name	Functional Description	Inputs	Outputs
Geometry Module	This block is in charge of the computation of the geometry: handling the orbit and the attitude of the satellite and of computing the correspondences between SAR coordinates (range, azimuth) and geographic coordinates (e.g. lat, lon, height)	- **External DEM:** input received from the external (i.e. from the user) - **Orbit/Attitude parameters:** input received from the external (i.e. from the user) - **Acquisition timing parameters:** receives the SWST,SWL, PRF parameters of the acquisition - **Nominal pointing:** receives the mechanical pointing of the instrument - **Acquisition start/stop:** receives the information about the size and the geographical location of the acquisition	- **GEO2SAR:** this product contains the matrix that gives the correspondence between SAR coordinates and geographic coordinates. It is used to project data from geographic coordinates to SAR coordinates. Other geometric information in SAR coordinates are here contained. - **SAR2GEO:** this product contains the matrix gives the correspondence between geographic coordinates and the SAR coordinates. It is used to project data from SAR coordinates to geographic coordinates. Other geometric information in Lat/Lon coordinates are here contained. - **Instrument acquisition direction** - **Orbit** - **GEOAMB:** incidence angle, off-nadir angle, slant range for the ambiguous zones - **Shadowing mask:** this matrix gives the information of the point on ground that are not under visibility due to the acquisition geometry and the DEM
Scene generator module	This block implements the forward module from the biophysical properties of the terrain to the SAR reflectivity	- **SAR2GEO:** geometric info in Lat/Lon coordinates - **GEOAMB:** geometric info from ambiguous zone in Lat/Lon coordinates - **Biophysical parameters**	- **Reflectivity map:** SAR reflectivity for the non ambiguous zone - **Reflectivity from ambiguous zone:** SAR reflectivity for the ambiguous swaths - **L2 Reference map:** the reference map for the physical measure to be retrieved by the L2RM. This map represents the "true" Level 2 information.
Instrument module	This block represents the modelling of SAR instrument and of the processing chain. Is main objective are: - Providing the instrument impulse response functions - Providing the sensitivity, the ambiguity rejection performances and	- **GEO2SAR geometry:** geometric information in SAR coordinates - **Orbit** - **Instrument acquisition direction** - **GEOAMB:** geometric info from ambiguous zone in SAR coordinates	- **L1a IRF and L1b IRF:** the Impulse Response Functions for Level1-a and Level1-b products - **System errors:** the parameters that model error to be injected on the data caused by acquisition (ambiguities, scalloping, speckle) and by instrument itself (thermal noise, cross talk, channel imbalance)

Module name	Functional Description	Inputs	Outputs
	the instrument non idealities	- **Instrument parameters:** all the parameters that characterize the instrument.	
Level-1 Processing Module	This block handles the generation of the L1a and L1b products through ingestion of the reflectivity map and of the instrument IRF, taking into account the instrument errors and non-idealities.	- **L1a/L1b IRF** from Instrument simulator (IM) - **System errors** from the System errors simulator (IM) - **Reflectivity map and Reflectivity map from ambiguous zone** from the SGM -	- **L1a product** - **L1b products**
L2 Retrieval Module	This block inverts the forward module to retrieve the physical properties of the terrain from the L1 SAR product	- **Auxiliary data:** all the auxiliary information that can be needed to invert the L1 information - **L1 products**	- **L2 products**
Performance Evaluation Module	This block offers the functionalities to evaluate the quality of the products of some modules	- **L1a/L1b IRF** - **L1 products** - **L2 products** - **Reflectivity map** - **L2 Reference map**	- **Instrument performance measures:** they characterize the IRF in the sense of typical measures such as resolution, Peak-to-Side Lobe Ratio (PSLR), ecc. - **L1 performance measures:** they represent the quality of the L1 data, also by comparison with the Reflectivity map to verify the impact of the system. - **L2 performance measures:** they represent the quality of the L2 data, also by comparison with the L2 Reference map to verify the impact of the system.

9.1.2. GEOMETRY MODULE

The Geometry Module (GM) is in charge of the computation of all the geometric information related to both the satellite orbit and the instrument pointing. It is worth to stress that the other modules have to rely on the functionality offered by the Geometry Module for any information related to the acquisition geometry, such as the altitude, the attitude, the relation between Geographic and SAR coordinates, etc.

The GM can be decomposed in the following first level building blocks:

- **Orbit simulator:** this building block is in charge of characterizing the orbit of the satellite through the computation of the Orbit State Vectors. Also possible orbit errors are here taken into account. This block is shared with all the other instruments and its generic description can be found in Section 8.1.
- **Attitude simulator:** this building block is in charge of computing the attitude of the platform. Also possible attitude restitution errors are here taken into account. This block is shared with all the other instruments and its generic description can be found in Section 8.1.
- **AOCS/Instrument coupling:** this building block simulates the mounting of the instrument in the platform, combining the attitude with the antenna to compute the actual pointing direction of the instrument. This block is shared with all the other instruments and its generic description can be found in Section 8.1.

- **Scene Interaction Geometry:** this building block computes all the geometric information related to the interaction of the acquisition geometry with the scene. In case of active microwave instruments, within the block "Intersection with DEM" described in Section 8.1.4, four different specialized lower level building blocks can be identified:

- **SAR2GEO block**: this building block computes all the geometric information (elevation angle, incidence angle, ecc.) in Geographic coordinates. Moreover this block is in charge of computing the matrixes to project the Level 1 data from SAR coordinates (range, azimuth) to Geographic coordinates (Latitude, Longitude). It is worth underlining that within the SAR2GEO product, all the geometric information will computed for a grid in Lat/Lon that identifies the extent of the scene to be simulated in Geographic coordinates.
- **GEO2SAR block:** this building block computes all the geometric information (elevation angle, incidence angle, ecc.) in SAR coordinates. Moreover this block is in charge of computing the matrixes to project the SAR reflectivity map from Geographic coordinates (Latitude, Longitude) to SAR coordinates (range, azimuth). It is worth underlining that within the GEO2SAR product, all the geometric information will computed for a grid in range/azimuth that identifies the extent of the scene to be simulated in SAR coordinates.
- **Shadowing mask:** this building block computes the shadowing mask, that identies the points in SAR coordinates from which there will not be any backscattering due to local altimetry on ground that prevents the Line-Of-Sight visibility from the antenna.
- **Range ambiguities:** this building block computes all the geometric information (elevation angle, incidence angle, ecc.) from the ambiguous swath areas.

In Figure 9-15 the input and output interfaces of the GM are represented.

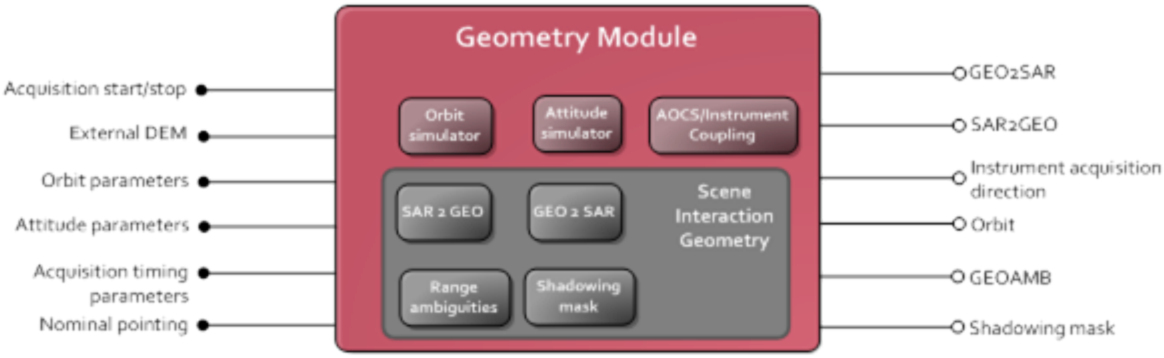


Figure 9-15: Input and output interfaces of the Geometry Module

In Figure 9-16, the data flow diagram of the Geometry Module is depicted. Due to the high resolution of SAR instruments, all the geometric information are computed according to the DEM passed as an input to the Module. As it can be there observed, essentially each building blocks computes its own output data starting from the inputs provided to the GM. Only the Shadowing mask building block takes as input the GEO2SAR product to compute the shadowed points. It is worth noticing that all the inputs data of the GM are parameters or auxiliary data, as the DEM, that have to be provided by the user as a priori information.

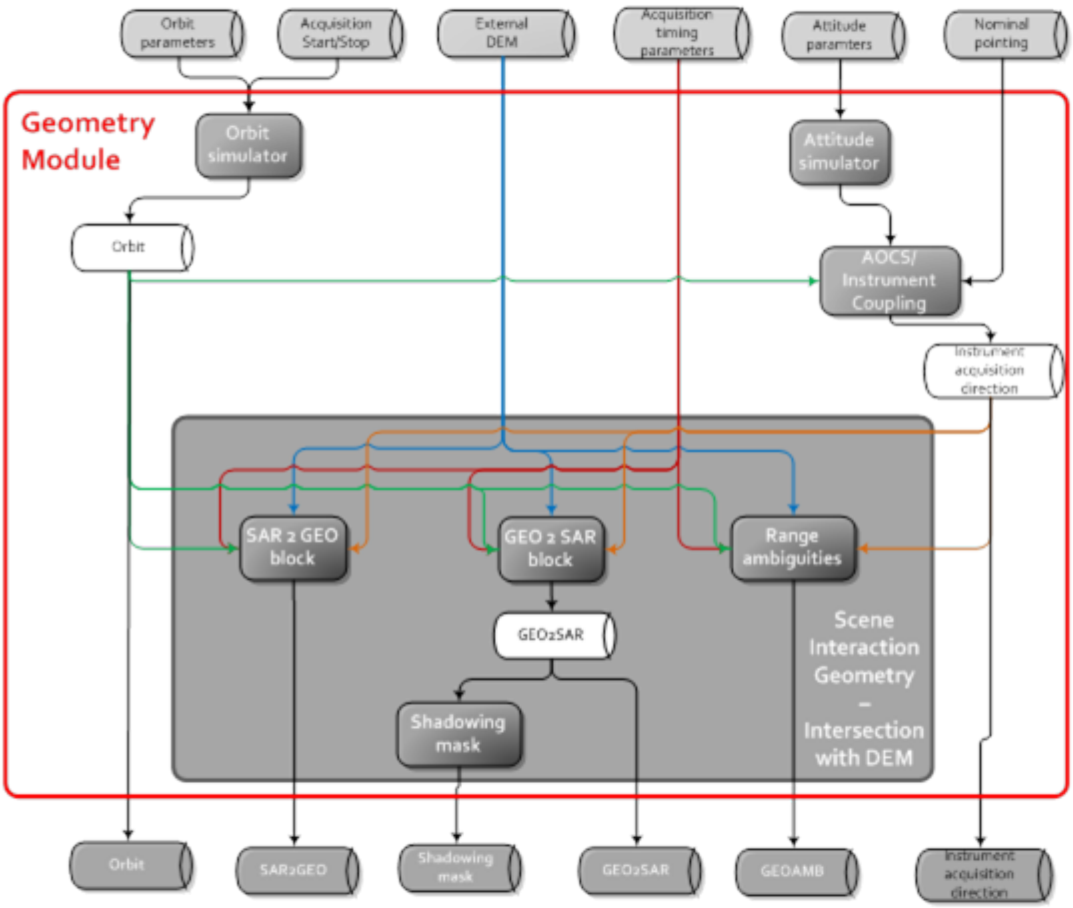


Figure 9-16: Data-flow diagram for Geometry Module

9.1.3. SCENE GENERATOR MODULE

The Scene generator module is in charge of providing the Reflectivity map of the area observed by the instrument on ground depending on the frequency of the signals transmitted by the instrument and on the biophysical characterization of the scene.

The computation of the Reflectivity map from the biophysical parameters can be decomposed in a two-step approach:

1) from the biophysical parameters to the Level 2 reference map
2) from the Level 2 reference map to the Reflectivity map

In fact, in the most general case, the biophysical parameters of the scene are passed as input to the SGM so that, exploiting a Level 2 forward model, the desired Level 2 Reference map is computed. As an example, if the Level 2 information is a vegetation map, the Scene Definition Block will give as output the vegetation map according to the biophysical parameters and the implemented model. Then the Forward Model building block, starting from the L2 Reference map, will compute the Reflectivity map for the desired scene. It is worth noticing that the L2 Reference map will be given also as an output of the SGM since it represents the ideal Level 2 information that should be retrieved by the L2RM. Thus, by comparison of the Level 2 product with the L2 Reference map, the performance of the system in the Level 2 retrieval sense can be directly evaluated.

As it can be seen in Figure 9-17, the SGM can be decomposed in two building blocks, the Scene Definition and the Forward Model. The Scene Definition has to implement the physical model that transforms the biophysical parameters to the Level 2 data. The Forward Model has to implement the model of the interaction of the incident electromagnetic wave with the physical characteristics of the scene, characterized by the Level 2 Reference map.

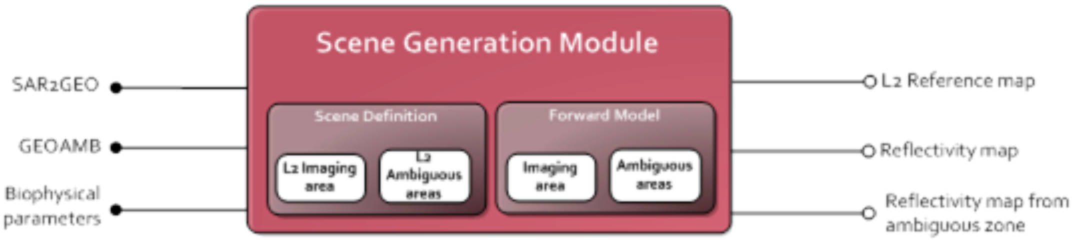


Figure 9-17: Data Input and output interfaces of the Scene generator module

The SGM can be decomposed in the following first and second level building blocks:

- **Scene Definition**
  - **L2 imaging area:** to compute the L2 Reference map in the imaging area
  - **L2 Ambiguous areas:** to compute the L2 Reference map in the ambiguous swaths in the range direction
- **Forward Model**
  - **Imaging area:** to compute the Reflectivity map in the imaging area
  - **Ambiguous areas:** to compute the Reflectivity map in the ambiguous swaths in the range direction

It has been here stressed the decomposition of the BBs in two second level elements, one for the imaging area and one for the ambiguous areas, for two reasons. The former is that different models could be used in the imaging area and in the ambiguous areas since an increased accuracy is required in the imaging area. The latter is that, looking for commonalities across different instrument types, the BBs for the imaging area will be common to all the instruments types while the BBs for ambiguous areas will be typical of the active microwave instruments.

From the figure below, that depicts the data-flow diagram for the SGM, it can be observed that also the Geometrical information is needed by both the BBs in the SGM. In case of the Forward Model building block, this input is needed to evaluate the reflectivity, which is function of the properties of the scene as well as of the incidence angle of incident wave.

Moreover, the geometric data are needed as input to both the L2 Forward model and the Forward Model to select the interval of Geographic coordinates for which the scene has to be modeled. In fact, the geometric data are defined on a grid in Geographic coordinates that coincides with the extent of the scene to be simulated. Thus, the SGM can take the information about the interval of Geographic coordinates for which the scene as to be modeled as well as the sampling steps in both the directions from the geometric data.

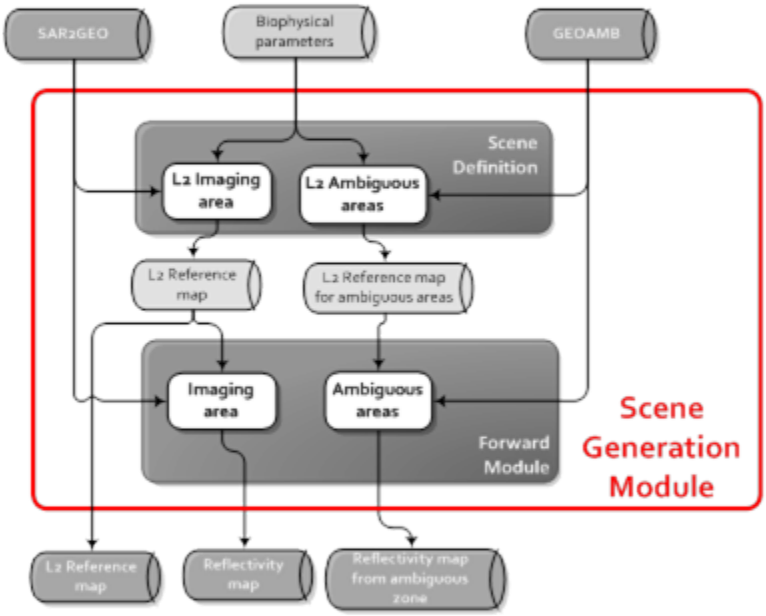


Figure 9-18: Data-flow diagram for the Scene generator module

As an alternative approach, the L2 Reference Map could be provided as an input to the SGM instead of being computed according to a model. This approach assumes that a database exists where the information for the physical parameter that is aimed to be retrieved by the L2RM is stored as maps in Geographic coordinates. The data-flow can be thus represented as in the following figure. In this chain, the functionality offered by the Scene Definition Block is to cut and resample (if needed) the L2 Reference map given as input on the Geographic grid contained in the Geometric data.

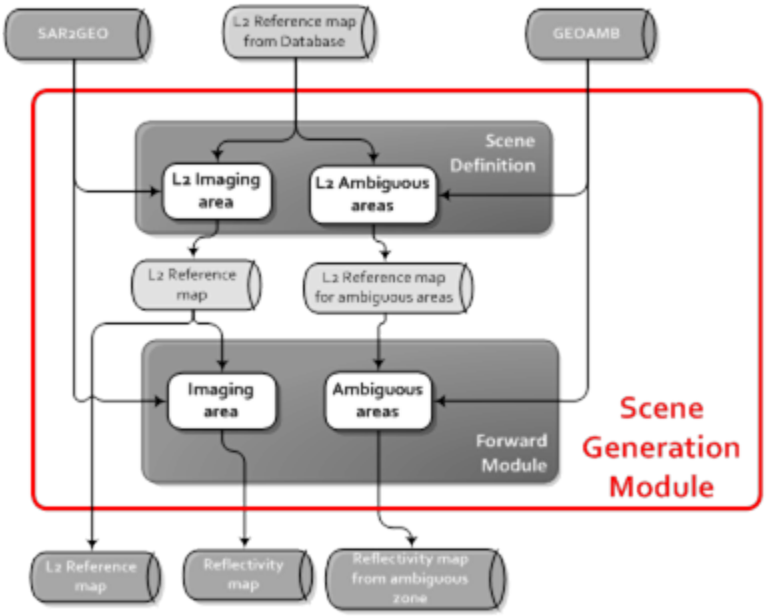


Figure 9-19: Data-flow diagram for the Scene generator module in case of L2 Reference map passed as input

9.1.4. INSTRUMENT MODULE

The Instrument Model is in charge of characterizing the active microwave instrument, taking into account its inaccuracies too, and part of the Level 1 processing chain, since these two parts of the system are strictly related. Moreover, this module contains also a building block that is in charge of providing the acquisition timing parameters to the GM, allowing it to compute the geometric information that depends on the characteristics of the instrument, such as the swath size.

The IM can be decomposed in the following building blocks:

- **Antenna Block:** this BB is in charge of computing the antenna pattern in both elevation and azimuth direction starting from the parameters set in the configuration. Thus, here essentially a model for the antenna usually adopted in active microwave instrument has to be implemented (planar array or reflector antenna). As an optional functionality, the antenna model has to be able to simulate a partial failure of the antenna (i.e. a given number of elements of a planar array antenna are not working). Moreover, this block has also to offer the functionality of taking an antenna pattern as external input, taking care of resampling it, if needed.
- **IRF generation:** this BB is in charge of providing the IRFs along both the slant range direction and the azimuth direction as well as the reference IRFs. The model here implemented will combine the parameters read from the configuration with the knowledge of the acquisition mode to compute the IRF. It is worth underlining that the IRF model should take into account also the distortion of the IRF itself due to errors, such as intra-pulse errors, inter-pulses errors, Doppler centroid errors, ecc. Those errors have an impact on the IRF, modifying it in the sense of typical features such as 3dB-width resolution and PSLR. On the other hand, the L1a/L1b Reference IRFs are computed as error free IRFs in order to provide as output the ideal IRF of the simulated instrument to be used by the Performance Evaluation Module.
- **Errors&Sensitivity:** this BB is in charge of evaluating all the errors related to the physical instrument as well the effects that depends on the SAR acquisition. Several inputs are needed here: geometric information, instrument parameters from the configuration, the antenna patterns. Moreover some models will be unavoidably customized to the acquisition mode. The following errors models will be here implemented:
  - Range ambiguities model
  - Azimuth ambiguities model
  - Scalloping model
  - Radiometric errors model
  - Instrument errors model, that comprises the errors due to the physical instrument such as polarization cross-talk, channel imbalance, etc.
  - NESZ model
  - Speckle model

In the figure below, the BBs composing the IM and the input/output interface are summarized.

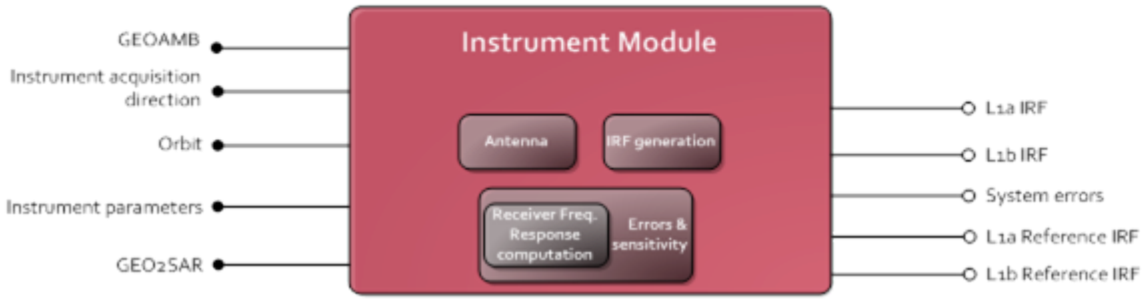


Figure 9-20: Input and output interfaces of the Instrument module

In the figure below, the data-flow diagram for the IM is shown.

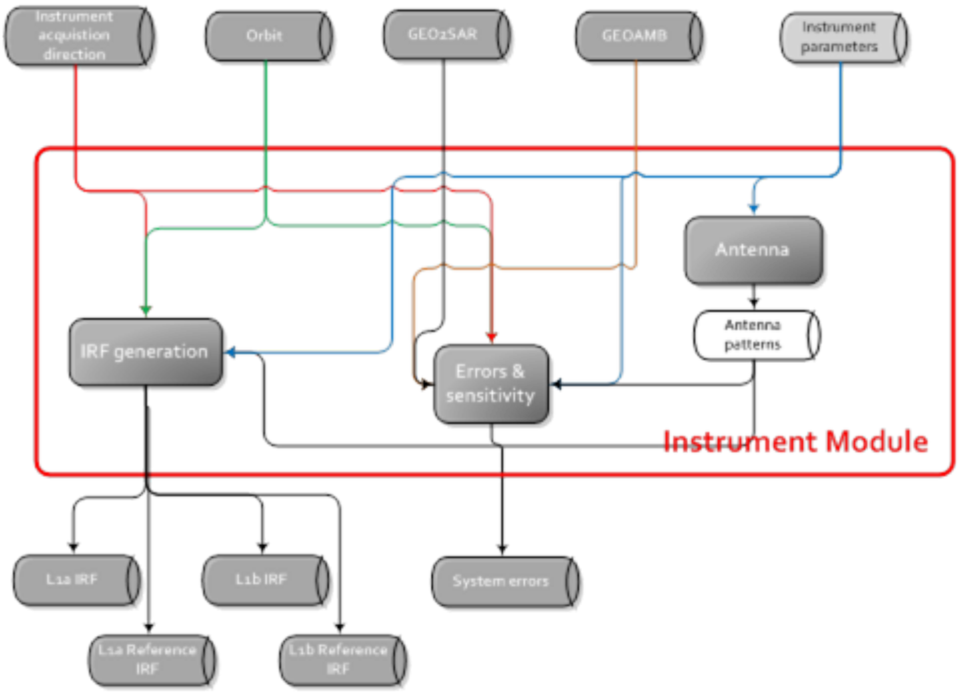


Figure 9-21: Data-flow diagram for the Instrument module

9.1.5. LEVEL-1 PROCESSING MODULE

The Level-1 Processing Module handles the generation of the L1a and L1b products through ingestion of the data provided by all the other modules apart from the Geometry Module and the L2 Retrieval Module which takes as input the Level 1 products. Thus, the L1PM can be thought as a sink where the information from the SGM and the IM are combined to produce the Level 1 products. This can be verified by inspection of the following figure, where the input and output interfaces of the L1PM are represented.

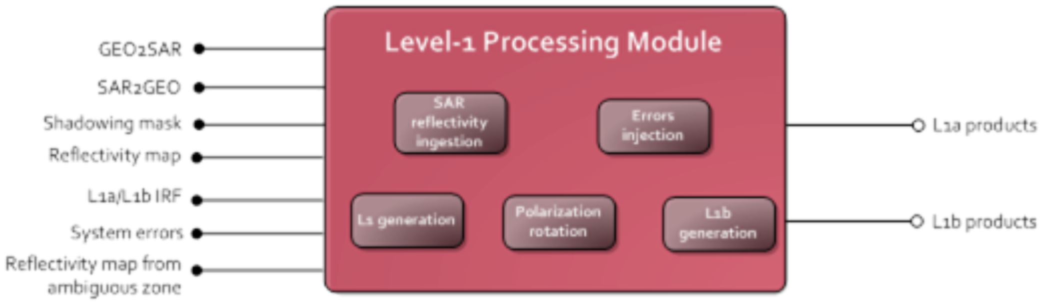


Figure 9-22: Input and output interfaces of the Level-1 Processing Module

As it is shown in the figure above, the L1PM is composed by the following BBs:

- **SAR Reflectivity ingestion:** this block is in charge of ingesting the SAR reflectivity maps from the SGM (for both the imaging area and the ambiguous areas) exploiting the geometric data provided by the GM. Basically, two operations are here performed by two different second level building blocks:
  - GEO2SAR projection that essentially projects the Reflectivity matrix data from Geographic coordinates to SAR coordinates exploiting the GEO2SAR product

  - Shadowing mask application that essentially nullifies the reflectivity contribution from the point in SAR coordinates that are not in visible from the instrument according to the shadowing mask product

- **L1 generation:** this block is in charge of combining the reflectivity data in SAR coordinates with the IRF in order to apply to the reflectivity data the spatial resolution of the instrument. This BB basically performs a 2D complex convolution between the reflectivity data and each IRF, for L1a and L1b respectively.
- **L1b generation:** this block applies to the output of the L1 generation, namely the L1b convolved reflectivity, the remaining processing steps up to the L1b product. Here the following operations are performed:
  - Processing multilooking (only in case that the multilloking has not been taken into account at IRF level by the IM)
  - Swath merging for multiple subswaths acquisition modes
  - Projection from SAR coordinates to Geographic coordinates exploiting the SAR2GEO product provided by the GM
- **Polarization Rotation:** this BB is applied in case of different polarization are acquired by the instrument. In that case, a rotation can be applied to the data from the different polarizations to simulate the effects of inaccuracies of the instrument, such as cross-talk, or the effects of the medium, such as Faraday rotation.
- **Errors injection:** this BB is in charge of applying to the data all the errors that have been modeled by the IM and provided as input to the L1PM. It is worth noticing here that, according to the following figure where the data-flow diagram of the L1PM is sketched, the errors are applied in three points of the chain, depending on the particular effect to be injected. In fact, some effects have to be injected on the reflectivity map in SAR coordinates before the convolution (i.e. channel imbalance), some other effects have to be injected on the L1 convolved reflectivity (i.e. NESZ effect) and in this case, different errors can be injected on the L1a and on the L1b convolved reflectivity.

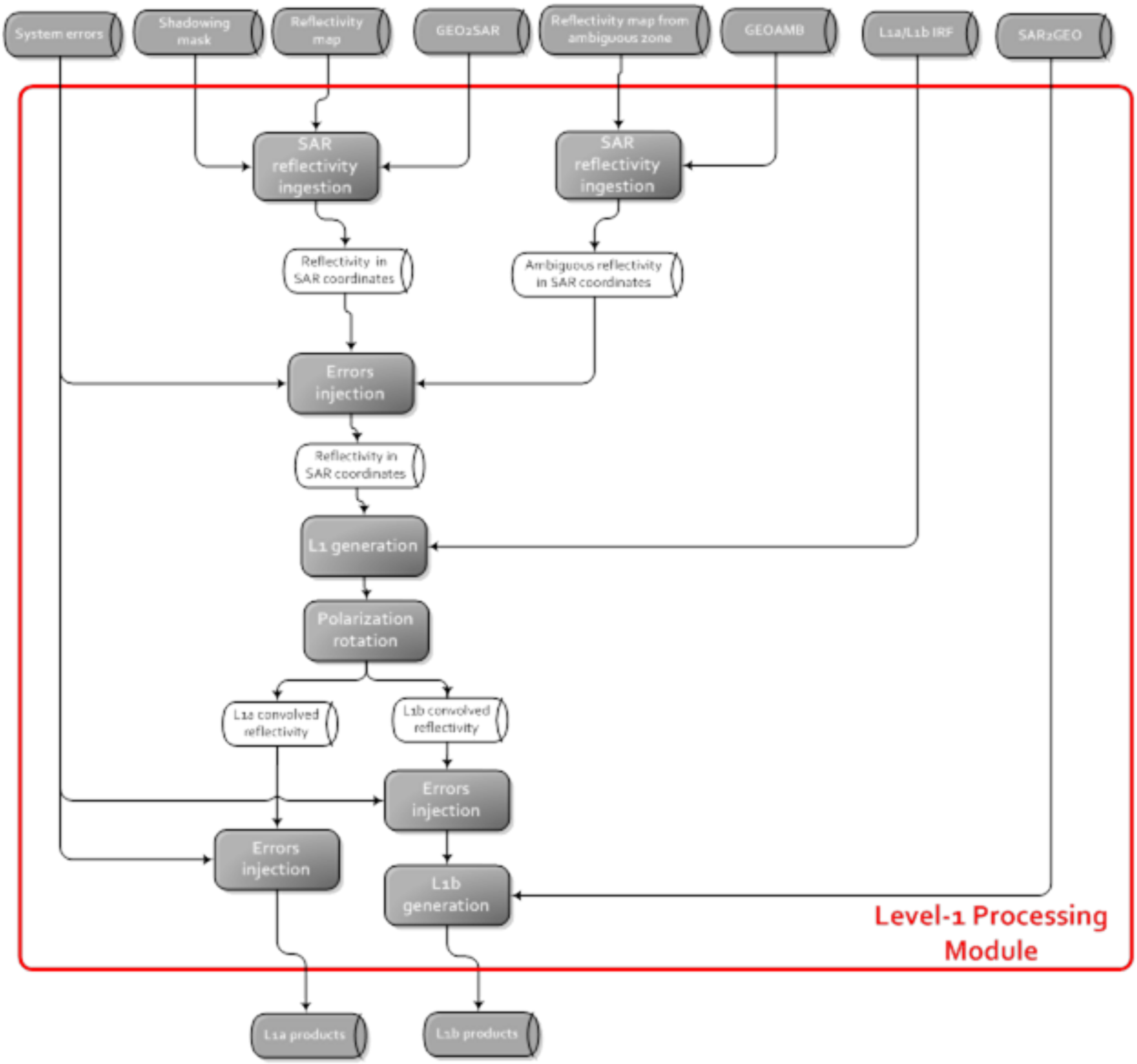


Figure 9-23: Data-flow diagram for the Level-1 Processing Module

9.1.6. LEVEL-2 RETRIEVAL MODULE

The Level 2 Retrieval Module, starting from the L1 products and some other auxiliary information is aimed at implementing the inversion algorithm to compute the Level 2 product, in order to close the end-to-end data flow of the simulator.

Depending on the particular retrieval algorithm, the other information that are needed as input can be foreseen to be divide in geometric information, that can be taken from L1 products, and in auxiliary data, such as maps that furnish additional information about the area on ground to improve the accuracy of the retrieved Level 2 measures.

As it can be seen by inspection of the following figure, where the BBs and the interfaces of the L2RM are described, the L2RM can be decomposed in the following BBs:

- **Atmospheric correction:** this block is in charge of correcting the distortion introduced by the atmosphere on the L1 data.
- **Feature Selection:** this block is in charge of classifying the pixels that belong to different areas in order to apply the retrieval algorithms only to the correct area. As an example, in case of soil moisture retrieval, this block will identify the pixels for land and the pixels for sea in order to apply the retrieval to the land pixels only. The Feature Selection block can exploit both a priori

information (i.e. coverage maps) and extrapolate the classification for each pixel from the L1b data.

- **Spatial and Timing Coregistration block:** the coregistration building block has been foreseen to provide this functionality to all the Level 2 algorithms that requires more than L1 product to estimate the desired physical characteristic, taking into account that those different L1 products can be obtained by multiple passes, different acquisition geometry, different instruments and so on.
- **Retrieval algorithms:** the actual retrieval routine.

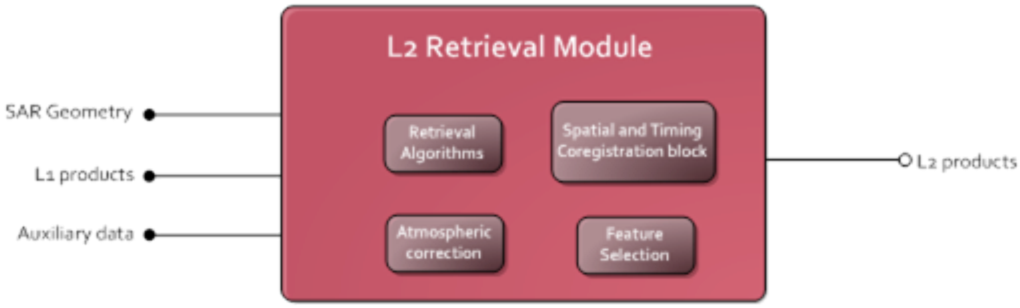


Figure 9-24: Input and output interfaces of the Level 2 Retrieval Module

In the figure below, the data-flow diagram for the L2RM is presented. There it can be observed that the coregistration block basically is in charge of ingesting the L1 products and, after atmospheric correction, of aligning them on a same sampling grid before passing them to the Retrieval block.

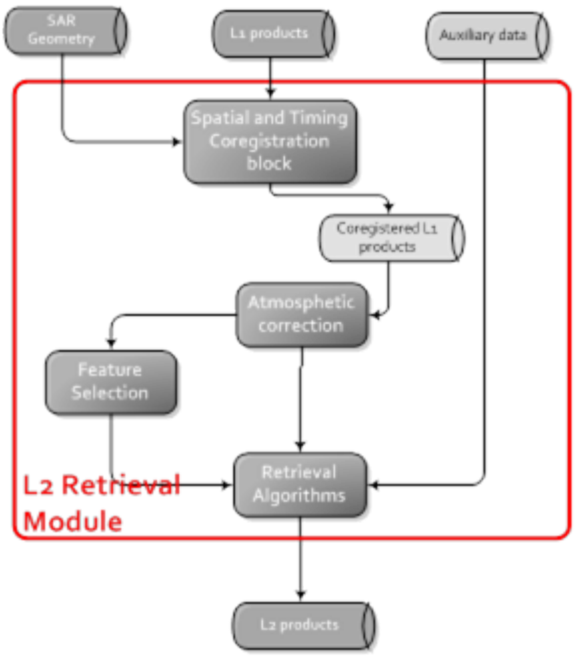


Figure 9-25: Data-flow diagram for the Level 2 Retrieval Module

9.1.7. PERFORMANCE EVALUATION MODULE

The performance evaluation module is aimed at evaluating the quality of the output products given by different modules throughout the E2ES chain. This is a key module in the E2ES since it is in charge of providing the performance measures of the simulated system, recalling that this is the final objective of the whole E2ES.

According to the end-to-end approach from the data-flow point of view, the initial data for the simulation are maps that describe physical quantity that has to be retrieved as Level 2 product. This approach allows then to have available at the same time the "real" physical quantity to be evaluated and its estimate computed by the simulated system, giving the opportunity to directly compare them in order to verify the accuracy of the Level 2 product.

The availability of the reference quantities with respect to compare those computed by the simulated system can be exploited not only for Level 2 data but for Level 1 and IRF too. In fact, the following pairs to be compared can be identified:

- L2 Reference map – L2 product
- Reflectivity map – L1b product
- L1a/L1b Reference IRF – L1a/L1b IRF

Moreover, a block dedicated to Platform Performance Evaluation can be foreseen too, as detailed in Section 8.6. This BB is shared with all the other instrument since it is aimed at evaluating basically the performance of the data provided by the common blocks into the GM.

According to the consideration drawn above, the input/output interface of the PEM can be defined as in the following figure.

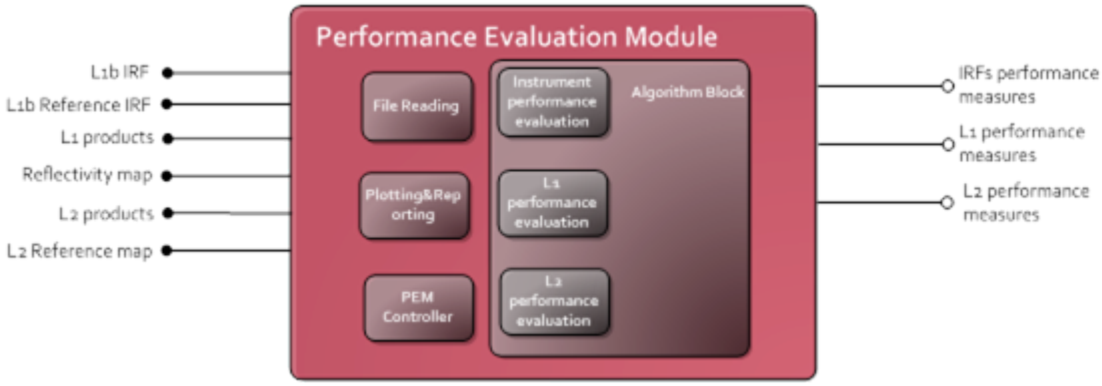


Figure 9-26: Input and output interfaces of the Performance Evaluation Module

As it can be noticed by inspection of the following figure, where the data-flow diagram of the PEM is depicted, apart the generic blocks that are mutuated from Section 8.6, the Algorithm Block can be decomposed in three BBs, one for each product to be analyzed:

- **Instrument performance evaluation:** this BB, taking as input the L1a/L1b Reference IRF and the L1a/L1b IRF, is aimed at comparing the performance measures evaluated on both of them. Hence, the main objective of this BB is to evaluate how the IRF is affected by the errors that have been injected by IM. Typical quality measure will be here evaluated, such as the 3dB-width resolution and the PSLR.
- **L1 performance evaluation:** this BB, taking as input the Reflectivity map and the L1 products, is aimed at evaluating the quality of the L1 product by measuring how much the simulated system (up tp L1PM) modify the reflectivity data provided by the SGM. The performance measures that can be provided as output will be
  - Error map: the pixel-by-pixel difference between the estimated and the reference map
  - Global error measures evaluated over the whole simulated area, such as the mean absolute error, the root-mean-square error and the percentual error.
  - Error measures, such as the mean absolute error, the root-mean-square error and the percentual error, computed with a binning approach to see how the error varies as function of the observed quantity.
- **L2 performance evaluation:** this BB, taking as input the L2 Reference map and the L2 products, is aimed at evaluating the quality of the L2 product by measuring how much the simulated system modify the reference physical data provided by the SGM.

- Error map: the pixel-by-pixel difference between the estimated and the reference map
- Global error measures evaluated over the whole simulated area, such as the mean absolute error, the root-mean-square error and the percentual error.
- Error measures, such as the mean absolute error, the root-mean-square error and the percentual error, computed with a binning approach to see how the error varies as function of the observed quantity.

It is worth noticing that, even if the same error measures are foreseen to be computed for the L1 and L2 product, it has been chosen to have to different BBs, the L1 performance evaluation building block and the L2 performance evaluation building block respectively. This is motivated by the fact that, in case some ad-hoc measures need to be introduced for the L2 product, they can be introduced without affecting the BB for L1 performance evaluation.

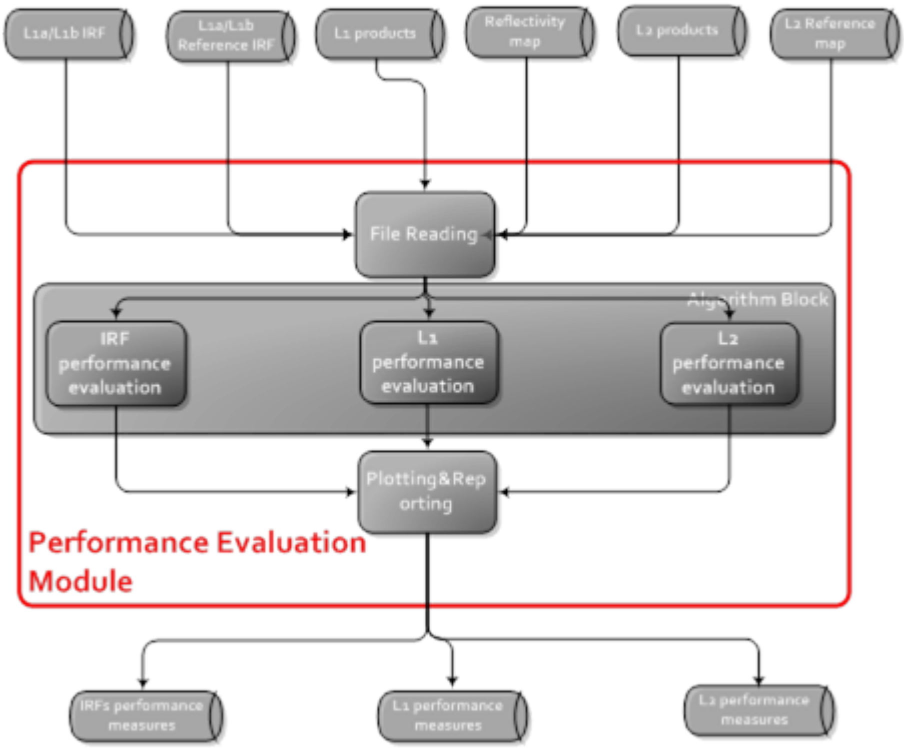


Figure 9-27: Data-flow diagram for the Performance Evaluation Module

In case that several sessions of the E2ES are executed, in order to compare the performance obtained by all the simulations, the PEM controller is introduced. The functionality offered by this BB is to manage the reading of several files, the evaluation of the overall performance and the reporting of the results.

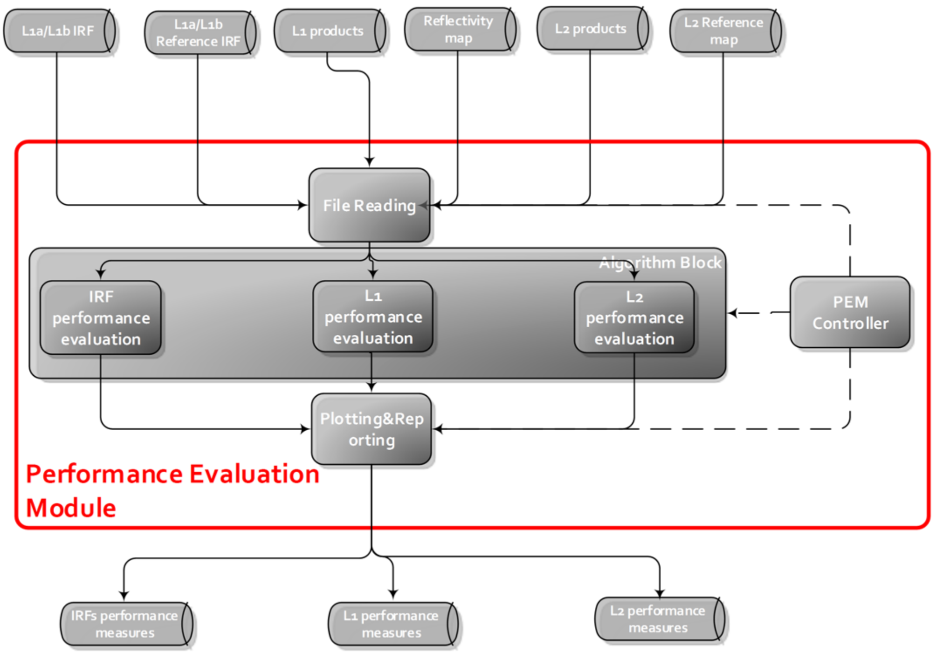


Figure 9-28: Data-flow diagram for the Performance Evaluation Module

9.1.8. REFERENCE ARCHITECTURE FOR LATER PHASES

In this section, as first the hierarchical construction of the reference architecture for later Phases will be given describing how each module can be decomposed in building blocks

In the following figure, the modules granularity is described.

As it can be noticed there, by changing the reference architecture, the modifications are concentrated only in two Modules: the IM and the L1PM. In particular some BBs are moved from the L1PM to the IM (SAR Reflectivity Ingestion and Polarization Rotation) and a new couple of BBs is introduced:

1. The Defocuser in the IM, that is in charge of transforming the reflectivity map in the Raw data for SAR. This functionality can be seen as the deconvolution of the reflectivity map by the two-dimensional chirp that is used in SAR focusing.
2. The L1 Focuser, that is the actual focuser that will be used in the ground Processor.

The advantage of using this architecture is that it allows to test the performance of the focuser that will be used during the operational life of the mission.

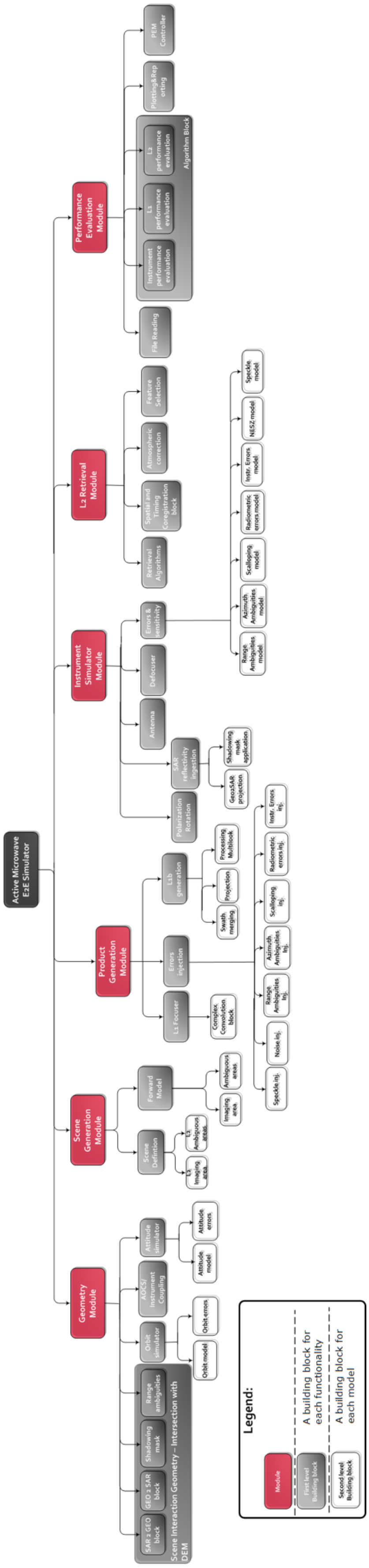


Figure 9-29: Hierarchical view of the active microwave E2ES reference architecture

In the following subsections, the IM and the PGM referred to the reference architecture under discussion will be detailed.

9.1.8.1. Instrument Module

The Instrument Model is in charge of producing the raw data, as they are seen by the instrument, starting from the reflectivity map provided by the SGM.

As a consequence, the Module has to contain BBs to ingest the reflectivity data and BBs to apply to them the effect of the instrument. The IM can be thus decomposed in the following building blocks:

- **Antenna Block:** this BB is in charge of computing the antenna pattern in both elevation and azimuth direction starting from the parameters set in the configuration. This is the same block that is within the IM in the Reference Architecture for Phase A.
- **Polarization Rotation:** this BB is in charge of rotating the polarization in case of multi-pol. acquisition. This is the same block that is within the L1PM in the Reference Architecture for Phase A.
- **SAR reflectivity ingestion:** this block is in charge of ingesting the SAR reflectivity maps from the SGM (for both the imaging area and the ambiguous areas) exploiting the geometric data provided by the GM. This is the same block that is within the L1PM in the Reference Architecture for Phase A.
- **Defocuser:** this block is in charge of applying a deconvolution operation to the ingested reflectivity map by the inverse of the two-dimensional chirp used in SAR focusing. This way, the raw data is obtained.
- **Errors&Sensitivity:** this BB is in charge of evaluating all the errors related to the physical instrument as well the effects that depends on the SAR acquisition. In the figure below, the BBs composing the IM and the input/output interface are summarized. This is the same block that is within the IM in the Reference Architecture for Phase A.

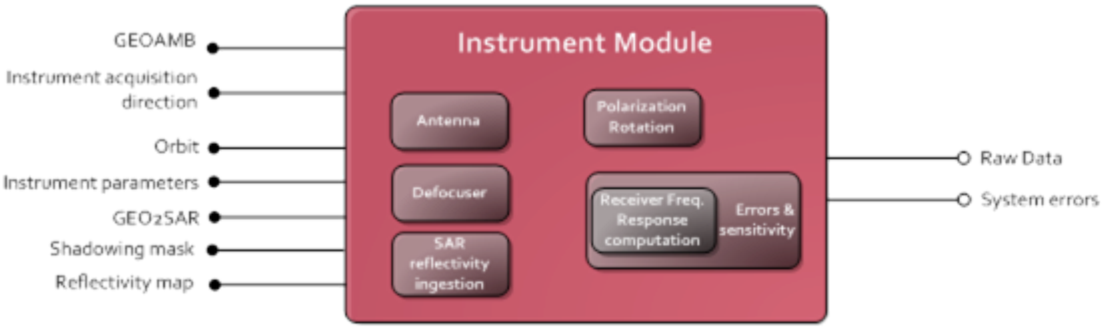


Figure 9-30: Input and output interfaces of the Instrument module

In the figure below, the data-flow diagram for the IM is shown.

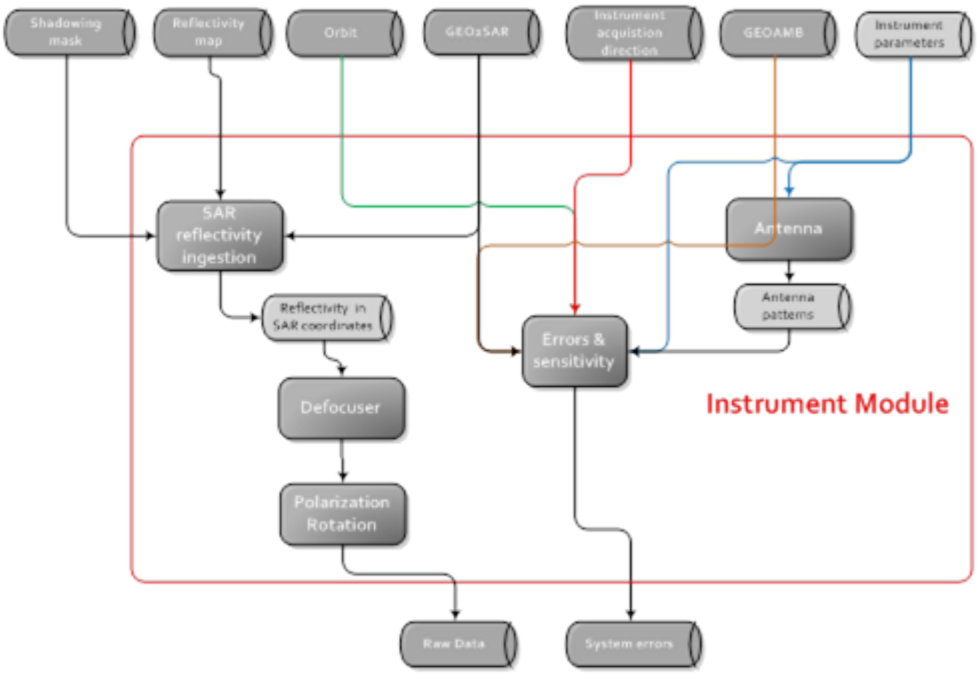


Figure 9-31: Data-flow diagram for the Instrument module

9.1.8.2. Level-1 Processing Module

The Level-1 Processing Module handles the generation of the L1a and L1b products through ingestion of the data provided by all the other modules apart from the Geometric Module and the L2 Retrieval Module which takes as input the Level 1 products. Thus, the L1PM can be thought of as a sink where the information from the SGM and the IM are combined to produce the Level 1 products. This can be verified by inspection of the following figure, where the input and output interfaces of the L1PM are represented.

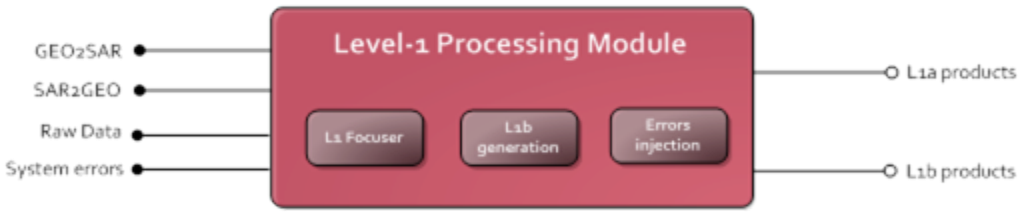


Figure 9-32: Input and output interfaces of the Level-1 Processing Module

As it is shown in the figure above, the L1PM is composed by the following BBs:

- **L1 Focuser:** this block is in charge of focusing the raw data. This block will contain the same SAR focuser that is foreseen to be used in the Ground Processor during the operational life of the mission.
- **L1b generation:** this block applies to the output of the L1 generation, namely the L1b convolved reflectivity, the remaining processing steps up to the L1b product. This is the same block that is within the L1PM in the Reference Architecture for Phase A.
- **Errors injection:** this BB is in charge of applying to the data all the errors that have been modeled by the IM and provided as input to the L1PM. This is the same block that is within the L1PM in the Reference Architecture for Phase A.

In the figure below, the data-flow diagram for the L1PM is shown.

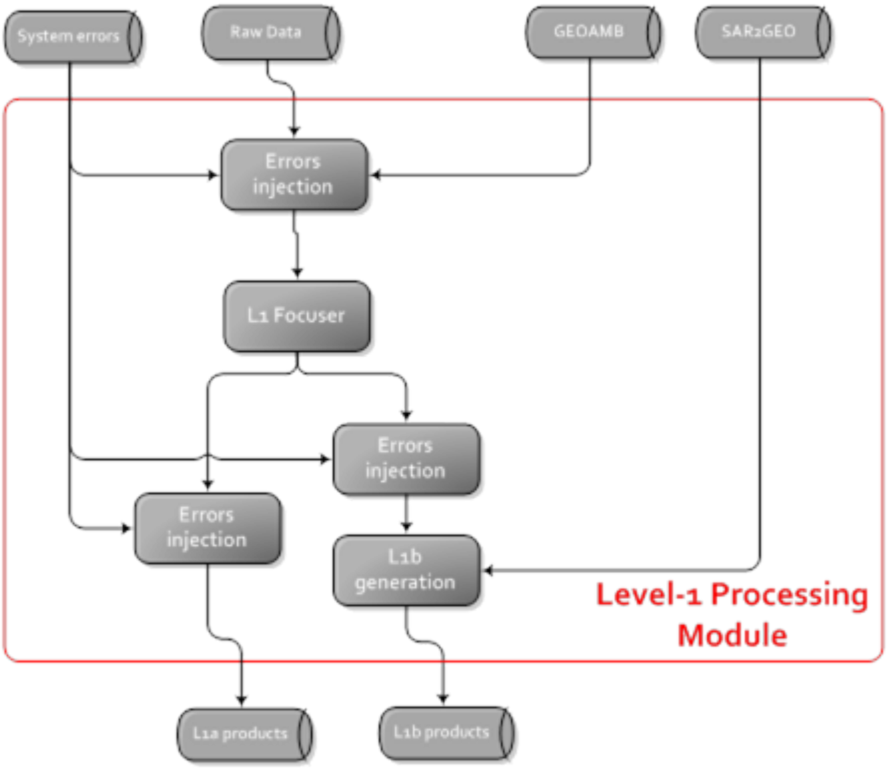


Figure 9-33: Data-flow diagram for the Level-1 Processing Module

9.2. RADAR ALTIMETERS

9.2.1. PROPOSED ARCHITECTURE

The main information retrieved by radar altimeters is the estimate of the distance from a target surface by measuring the satellite-to-target round-trip time of a radar pulse. However, significant additional information can be extracted by the received echoes by analyzing the backscattering intensity.

Starting from the basic radar principle (the altimeter emits a radar wave and analyses the echo), target height can be retrieved as the difference between the satellite's position on orbit with respect to an arbitrary reference surface and the satellite-to-surface range, that is computed by measuring the round-trip time. It is worth underlining here that

- in order to obtain measurements accurate to within a few centimetres over a range of several hundred kilometres, an extremely precise knowledge of the position of the platform is necessary;
- the accurate knowledge of a reference surface is required to allow the L2 retrieval algorithm to verify local variations of the target height.

Besides target height, by looking at the amplitude of the received echo, the backscatter coefficient and the surface roughness can be evaluated.

Taking into account the discussion drawn just above, the E2ES architecture for radar altimeters has to take into account that

- a different acquisition geometry with respect to SAR instruments is needed and a smaller subset of geometric information is necessary;
- the IM has to be customized for radar altimeters and a subset of the error effects has to be considered;
- different auxiliary data are needed, depending on the particular Level 2 physical quantities to be retrieved.

As a consequence, the main differences between the E2ES architecture for SAR imagery missions and the E2ES architecture for radar altimeters are concentrated at module level, while the execution order and the data-flow between the modules are not affected.

In the following sections, each module is discussed in the framework of the E2ES for radar altimeters.

9.2.2. GEOMETRY MODULE

The GM in case of radar altimeter has to take into account that a different acquisition geometry is adopted in those instruments and, in particular, a custom Scene Interaction Geometry building block is needed. In fact, in radar altimeters the following coordinates systems are defined:

- Scene coordinates: height with respect a reference surface (i.e. the ellipsoid or the geoid) and along track position on ground
- Radar coordinates: range and azimuth

so that an update of the Scene Interaction Geometry building block is needed to manage these coordinate systems providing the geometric matrices to perform the projection of the reflectivity data from one coordinate system to the other.

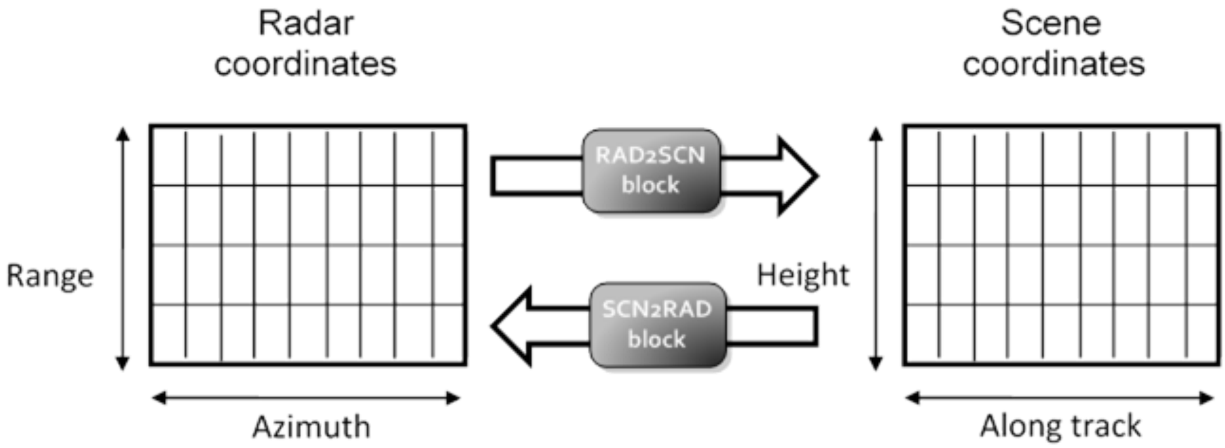


Figure 9-34: Coordinate system for radar altimeters

The GM input/output interfaces can be represented as in the following figure. There it can be noticed that

- all the changes are in the Scene Interaction Geometry building block, and in particular within the lower level building block Interaction with DEM;
- the SAR2GEO block and the GEO2SAR block are replaced by the RAD2SCN block and the SCN2RAD block, respectively. Those blocks provide the same functionalities but for the coordinate systems defined in the framework of radar altimeters;
- the Range ambiguities block and the Shadowing mask block are not needed here;
- the name of the output data has not been changed to not modify the interfaces among modules.

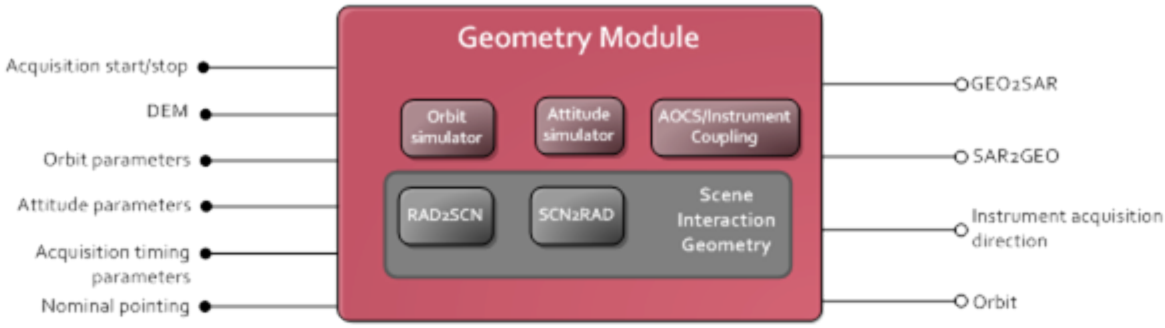


Figure 9-35: Input and output interfaces of the GM for radar altimeters

The data-flow for the GM in case of radar altimeter is then updated as in the following figure.

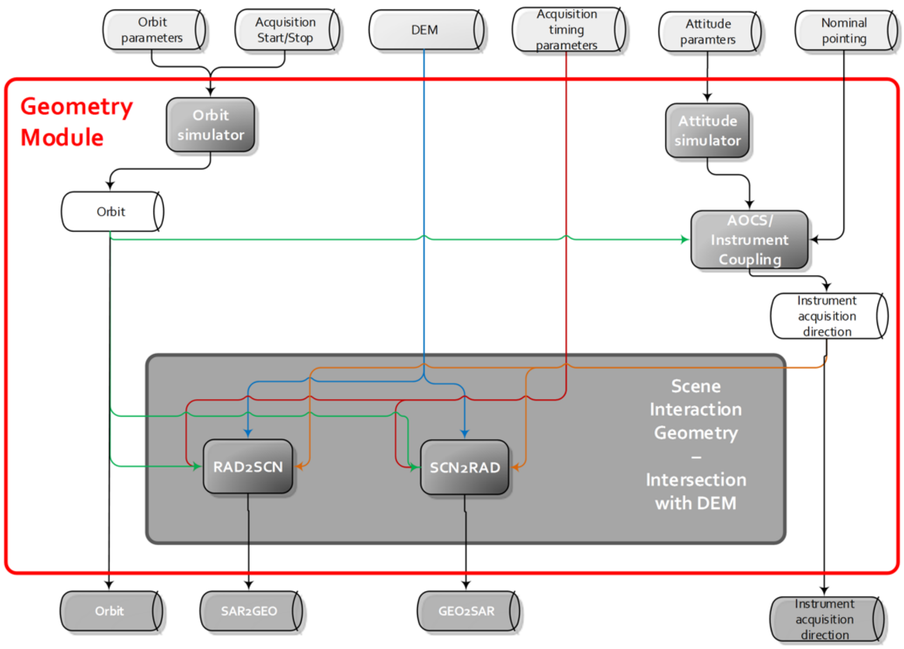


Figure 9-36: Data flow for GM in case of radar altimeters

9.2.3. SCENE GENERATOR MODULE

In the SGM in case of radar altimeters we need to take into account that

- the modules for both the building blocks (Scene Definition and Forward Model) have to be customized for the specific application;
- the second level building blocks for ambiguous areas are not needed since this effect has not to be simulated.

The resulting data-flow diagram for SGM in case of radar altimeters is shown in the following figure.

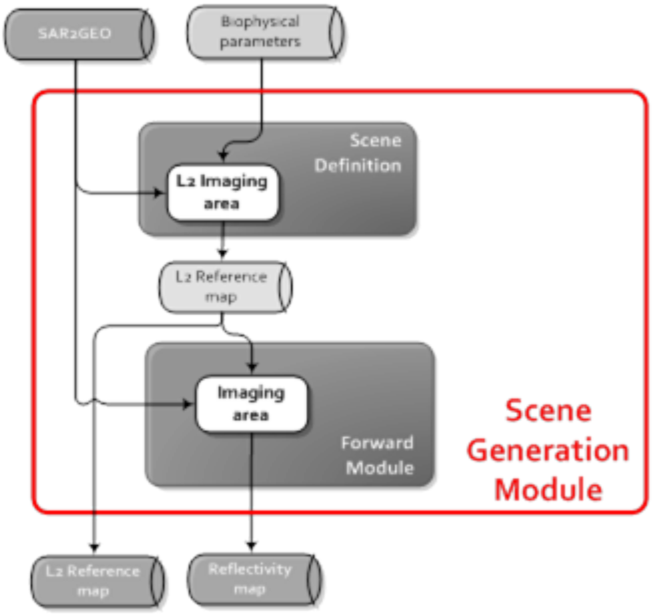


Figure 9-37: Data flow for SGM in case of radar altimeters

9.2.4. INSTRUMENT MODULE

The IM in case of radar altimeters is modified according to the input/output interfaces described in the figure below. As it can be observed there, the main difference is that only one IRF is given as ouput.

Moreover, the modules implemented by the BBs have to be customized for radar instrument and its processing chain, taking into account the different acquisition types (Pulse-limited or Delay-Doppler), according to [PD.1].

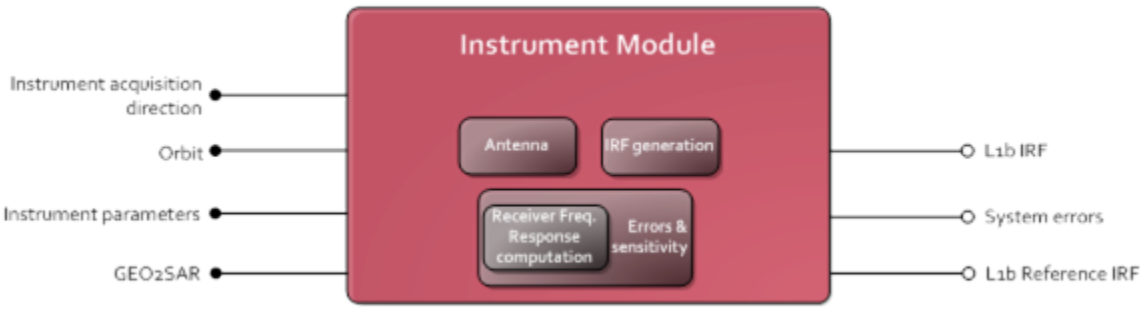


Figure 9-38: Input and output interfaces of the IM for radar altimeters

9.2.5. LEVEL-1 PROCESSING MODULE

The L1PM in case of radar altimeters performs its functionalities starting from the available input data. The data-flow for the L1PM is sketched in the figure below. There it can be observed that the L1PM processing chain for radar altimeters is a simplified version of the processing chain that has been defined in section 9.1.5 for SAR instruments since

- the ambiguities has not to be managed;
- the shadowing mask has not to be managed;
- only the L1b chain is implemented
- no multiple polarization are used in the existing/future altimetric mission

Moreover, the BBs that perform the projection from scene coordinates to radar coordinates and vice versa can be reused from the SAR processing chain, since in both the cases basically a 2D interpolation has to be performed.

Reasonably the L1 generation BB can be reused, since a 2D convolution is implemented.

On the other hand, the modules for the error injection have to be adapted to the different way of working of the instrument.

Another simplification of the L1PM for radar altimeters is that multiple subswath acquisition modes are not to be considered here.

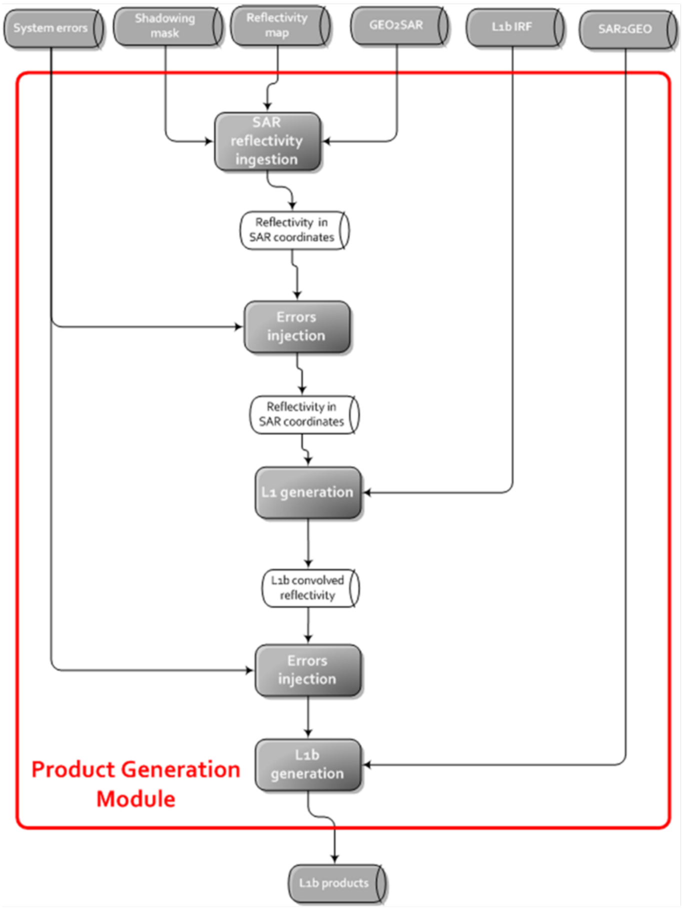


Figure 9-39: Data flow for L1PM in case of radar altimeters

9.2.6. LEVEL-2 RETRIEVAL MODULE

The L2RM in case of radar altimeters is expected to change due to the different retrieval algorithm implemented, since typical physical quantities have to be evaluated. Thus, the same architecture that was described in section 9.1.6 can be here considered, taking into account that the retrieval algorithm can need specific auxiliary data, such as a reference surface and a high precision orbit to retrieve the target height starting from the range information stored in L1 product.

The Spatial and Timing Coregistration block is needed also in case of radar altimeters since, according to [PD.1], some missions carry instruments at different frequencies so that an alignment of the L1 products can be required.

9.2.7. PERFORMANCE EVALUATION MODULE

The PEM in case of radar altimeters has the same interfaces of the SAR case, apart from the IRF, since only L1b IRF is considered here, as it is shown in the following figure.

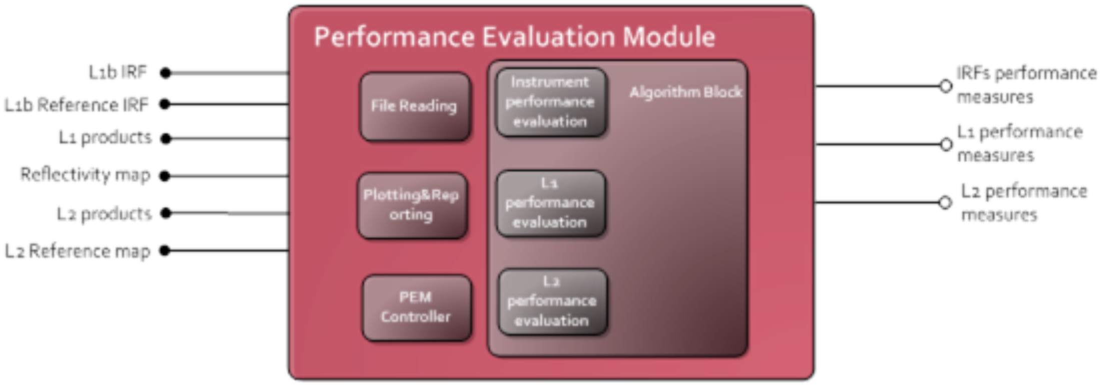


Figure 9-40: Input and output interfaces of the IM for radar altimeters

9.3. SCATTEROMETERS

9.3.1. PROPOSED ARCHITECTURE

The main information retrieved by scatterometers is the estimate of the power backscattered from the surface roughness when it is illuminated by a radar pulse.

Starting from the basic radar principle (the scatterometer emits a radar wave and analyses the echo), several information can be retrieved from the power backscattered. However, the main application of scatterometers is the measurement of ocean surface wind vectors since, at incidence angle higher than 20°, the backscatter coefficient increases with wind speed. Moreover the backscatter depends also on the wind direction relative to the direction of the radar beam. Because the backscatter is symmetric about the mean wind direction, observations at many azimuth angle are needed to resolve the directional ambiguity. As a consequence, different acquisition approaches have been developed to scan the earth surface at different incidence angles (mainly fan beam and rotating pencil beam).

Taking into account the discussion drawn just above, the E2ES architecture for scatterometers has to take into account that

- a different acquisition geometry with respect to SAR instruments is needed and a smaller subset of geometric information is necessary;
- the IM has to be customized for scatterometers and a subset of the error effects has to be considered;
- different auxiliary data are needed, depending on the particular Level 2 physical quantities to be retrieved.

As a consequence, the main differences between the E2ES architecture for SAR imagery missions and the E2ES architecture for scatterometers are concentrated at module level, while the execution order and the data-flow between the modules are not affected.

In the following sections, each module is discussed in the framework of the E2ES for scatterometers.

9.3.2. GEOMETRY MODULE

The GM in case of scatterometer has to take into account that a different acquisition geometry is adopted in those instruments and, in particular, a custom Scene Interaction Geometry building block is needed. In fact, as in the other architectures for active microwave E2ES, this block is mainly devoted to map the scene coordinates to the acquisition scatterometer coordinates.
Thus, in scatterometer the following coordinates systems are defined:

- Scene coordinates: geographic coordinates, i.e Latitude/Longitude
- Radar coordinates: signal-time/Doppler for fan beam acquisition

so that an update of the Scene Interaction Geometry building block is needed to manage these coordinate systems providing the geometric matrices to perform the projection of the reflectivity data from one coordinate system to the other. Reasonably, due to the very low spatial resolution of scatterometers with respect to SAR, high resolution DEM is not needed here.

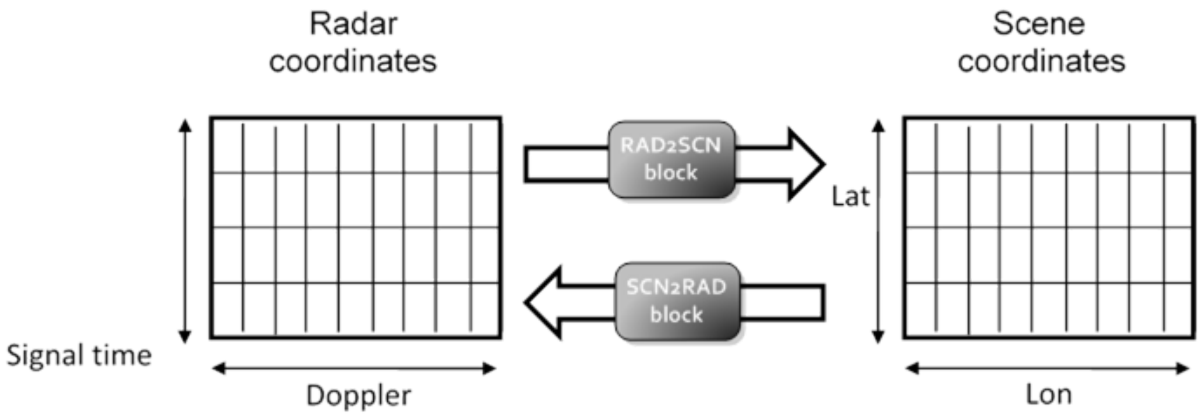


Figure 9-41: Coordinate system for scatterometers

The GM input/output interfaces can be represented as in the following figure. There it can be noticed that

- all the changes are in the Scene Interaction Geometry building block, and in particular within the lower level building block Interaction with DEM;
- the SAR2GEO block and the GEO2SAR block are replaced by the RAD2SCN block and the SCN2RAD block, respectively. Those blocks provide the same functionalities but for the coordinate systems defined in the framework of scatterometers;
- the Range ambiguities block and the Shadowing mask block are not needed here;
- the name of the output data has not been changed to not modify the interfaces among modules.

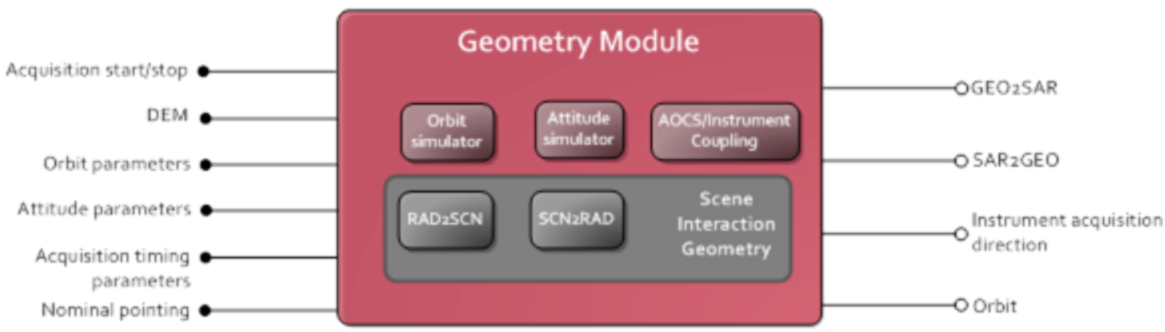


Figure 9-42: Input and output interfaces of the GM for radar altimeters

The data-flow for the GM in case of radar altimeter is then updated as in the following figure.

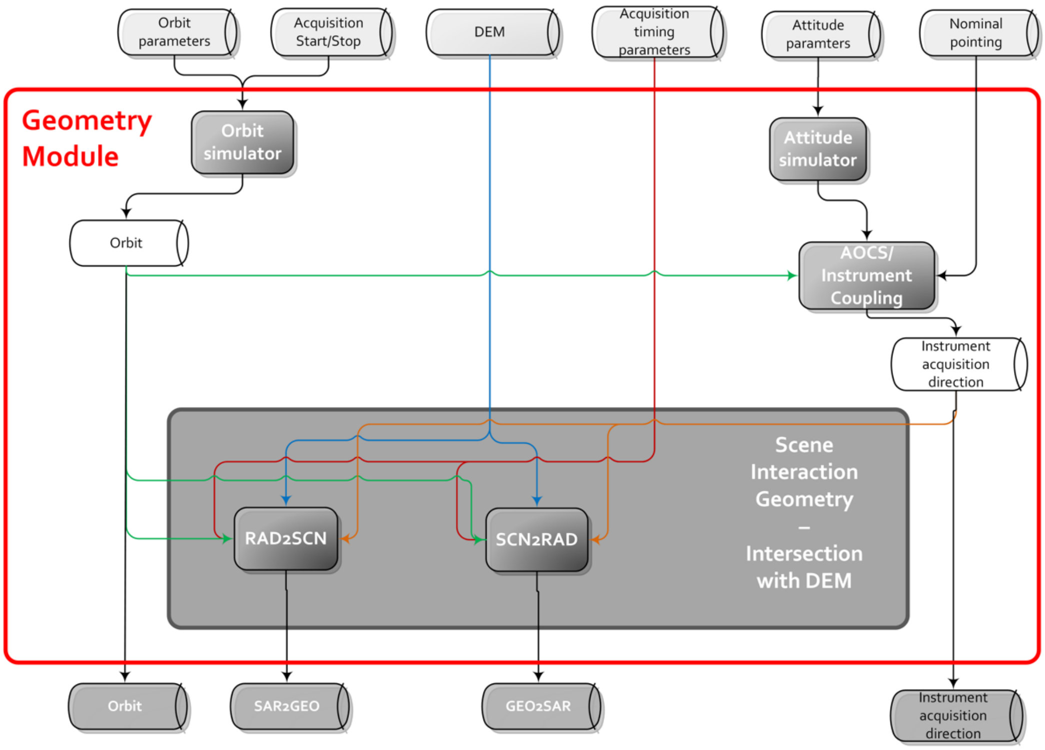


Figure 9-43: Data flow for GM in case of radar altimeters

9.3.3. SCENE GENERATOR MODULE

In the SGM in case of scatterometers we need to take into account that

- the modules for both the building blocks (Scene Definition and Forward Model) have to be customized for the specific application;
- the second level building blocks for ambiguous areas are not needed since this effect has not to be simulated.

The resulting data-flow diagram for SGM in case of scatterometers is shown in the following figure.

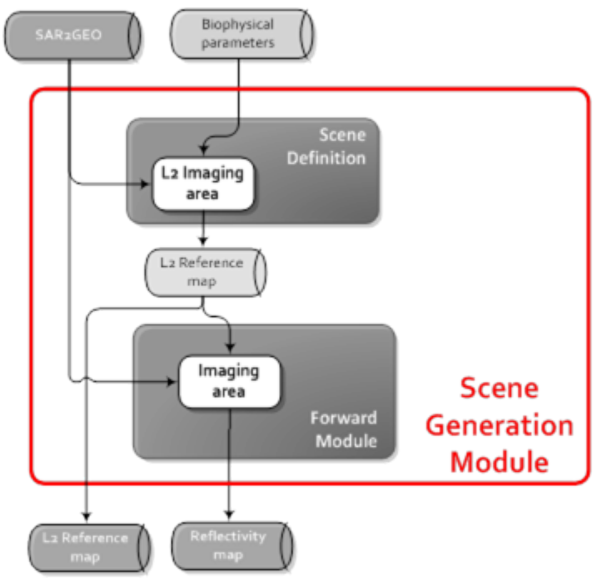


Figure 9-44: Data flow for SGM in case of radar altimeters

9.3.4. INSTRUMENT MODULE

The IM in case of scatterometers is modified according to the input/output interfaces described in the figure below. As it can be observed there, the main difference is that only one IRF is given as ouput.

Moreover, the modules implemented by the BBs have to be customized for radar instrument and its processing chain, taking into account the different acquisition types (fan beam or rotating pencil beam), according to [PD.1].

In case of fan beam acquisition, the IRF is shaped as the narrow antenna beam pattern along the signal time direction and shaped as the Doppler filter response along the Doppler direction. See figure below for an example. Moreover, depending on the number of pulses that are integrated, a loss in resolution has to be taken into account. Due to the flight of satellite, successive pulses are slightly shifted along the signal time direction, so that when they are integrated the resolution along that direction is degraded.

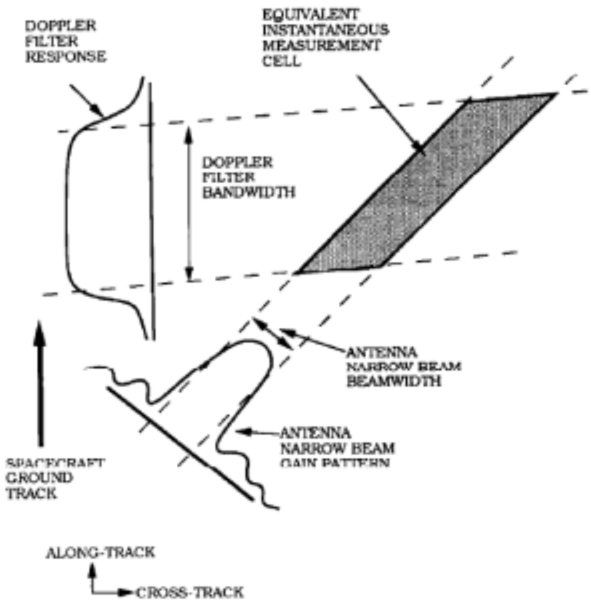


Figure 9-45: IRF determination for scatterometer

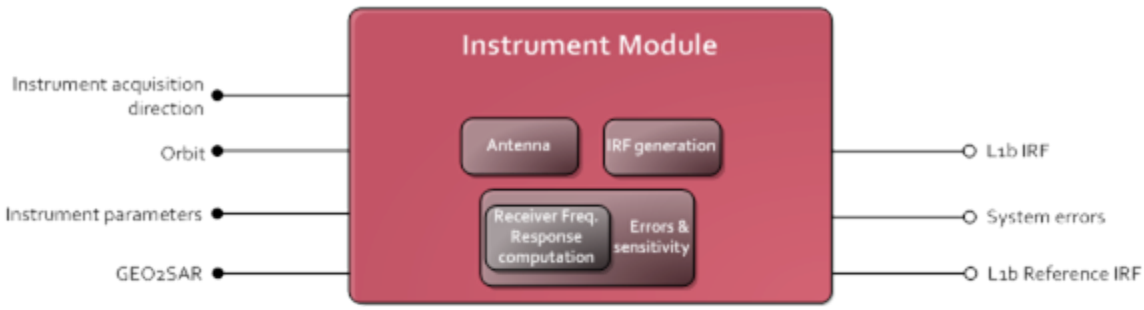


Figure 9-46: Input and output interfaces of the IM for radar altimeters

9.3.5. LEVEL-1 PROCESSING MODULE

The L1PM in case of scatterometers performs its functionalities starting from the available input data. The data-flow for the L1PM is sketched in the figure below. There it can be observed that the L1PM processing chain for scattrometers is a simplified version of the processing chain that has been defined in section 9.1.5 for SAR instruments since

- the ambiguities has not to be managed;
- only the L1b chain is implemented

Moreover, the BBs that perform the projection from scene coordinates to radar coordinates and vice versa can be reused from the SAR processing chain, since in both the cases basically a 2D interpolation has to be performed.

Reasonably the L1 generation BB can be reused, since a 2D convolution is implemented.

On the other hand, the modules for the error injection have to be adapted to the different way of working of the instrument.

It is worth underlining that similarly to SAR, where multiple subswaths have to be managed, the same approach can be reused here to manage the acquisition from the different beams in fan beam acquisition mode. In fact, the same blocks have to be executed for each beam to generate the L1 product related to each fan beam.

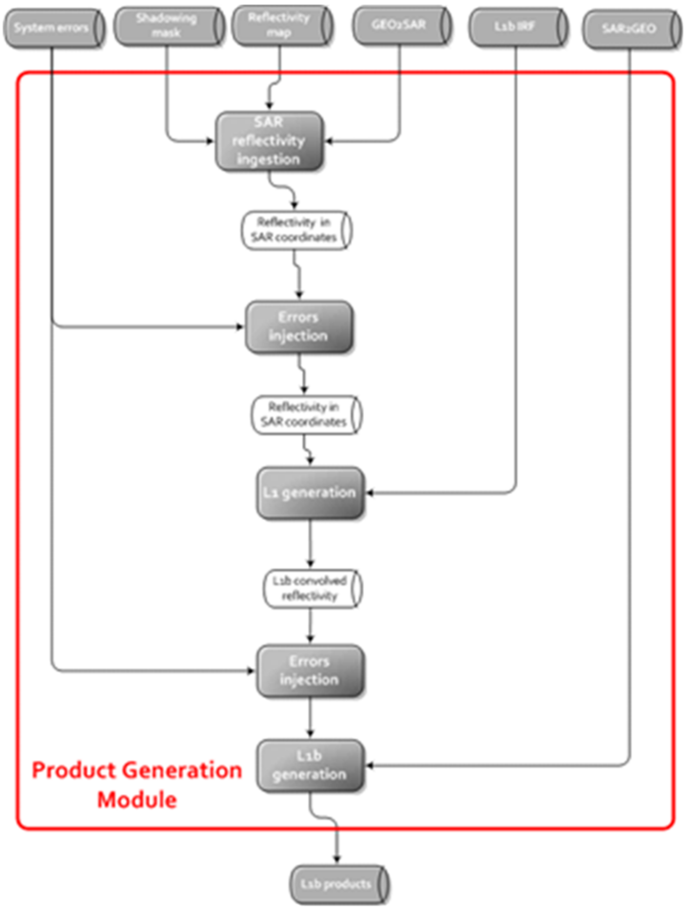


Figure 9-47: Data flow for L1PM in case of radar altimeters

9.3.6. LEVEL-2 RETRIEVAL MODULE

The L2RM in case of scatterometers is expected to change due to the different retrieval algorithm implemented, since typical physical quantities have to be evaluated. Thus, the same architecture that was described in section 9.1.6 can be here considered, taking into account that the retrieval algorithm can need specific auxiliary data, such as a reference surface and a high precision orbit to retrieve the target height starting from the range information stored in L1 product.

The Spatial and Timing Coregistration block is needed also in case of radar altimeters since, according to [PD.1], some missions carry instruments at different frequencies so that an alignment of the L1 products can be required.

9.3.7. PERFORMANCE EVALUATION MODULE

The PEM in case of radar altimeters has the same interfaces of the SAR case, apart from the IRF, since only L1b IRF is considered here, as it is shown in the following figure.

As an example, typical parameters for scatterometers as the normalized standard deviation can be computed here in the Instrument Performance Evaluation block.

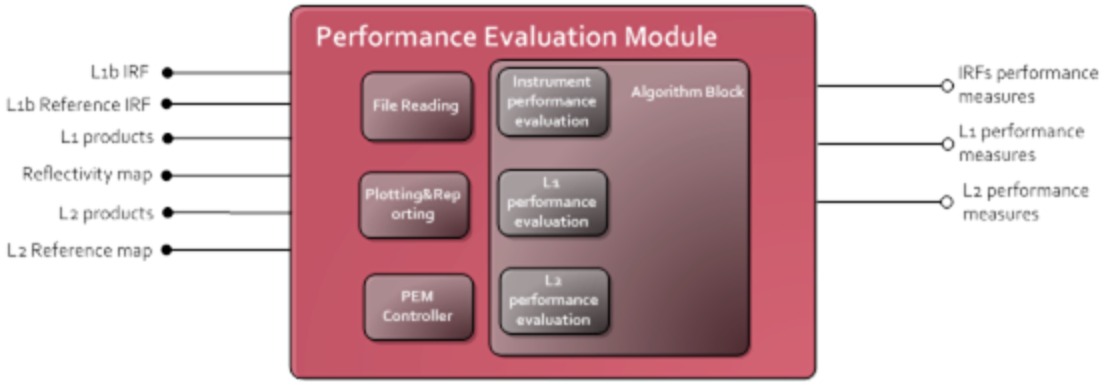


Figure 9-48: Input and output interfaces of the IM for radar altimeters

9.4. MULTIPLE CHAINS ARCHITECTURE

Throughout this section some examples of multiple chains architecture are discussed. It is worth underlining here that multiple chains apply to both the case of SAR instruments and radar altimeters.

According to TN-E2E-MC, the majority of the active microwave instruments have polarimetric capabilities, so that the instrument is able to acquire at the same time images of the same area at up to four linear polarizations.

From the E2ES point of view, parallel simulation chains have to be foreseen so that each polarization has its own chain. It is worth underlining here that the two chains have not to be necessarily implemented at software level by executing twice the modules in the two chains, but they can be also implemented at configuration level setting the module to compute its own outputs for the two polarizations in one execution. This approach configuration-driven requires that the module support this functionality.

According to the considerations drawn above, the GM will be reasonably executed once for all the simulation chains since the acquisition geometry, which relies on the orbit and on the attitude, is not polarization dependent as well as the DEM of the imaged area.

On the other hand, each per-polarization simulation chain will contain

- the SGM, since the interaction between the incident wave and the earth surface depends on the polarization
- the IM, since the instrument can have different IRFs and different errors for each polarization
- the L1PM, since in this module the ground reflectivity is combined with the instrument model

Finally, the L2RM, in the most general case, will use the Level1b products obtained from all the polarizations to compute the Level2 products. As a consequence, the L2RM will be executed once taking as input the Level1b products for each polarization.

The reference architecture discussed above has been represented in the following data-flow diagram in case of two different polarizations. By inspection of the data-flow diagram, it can be noticed that the IM from the first chain provides the acquisition timing parameters to the GM under the hypothesis that the two polarizations shares the same timing parameters. In case that this assumptions was not verified, the GM should be included in each per-polarization chain since the geometric parameters wil be polarization-dependent.

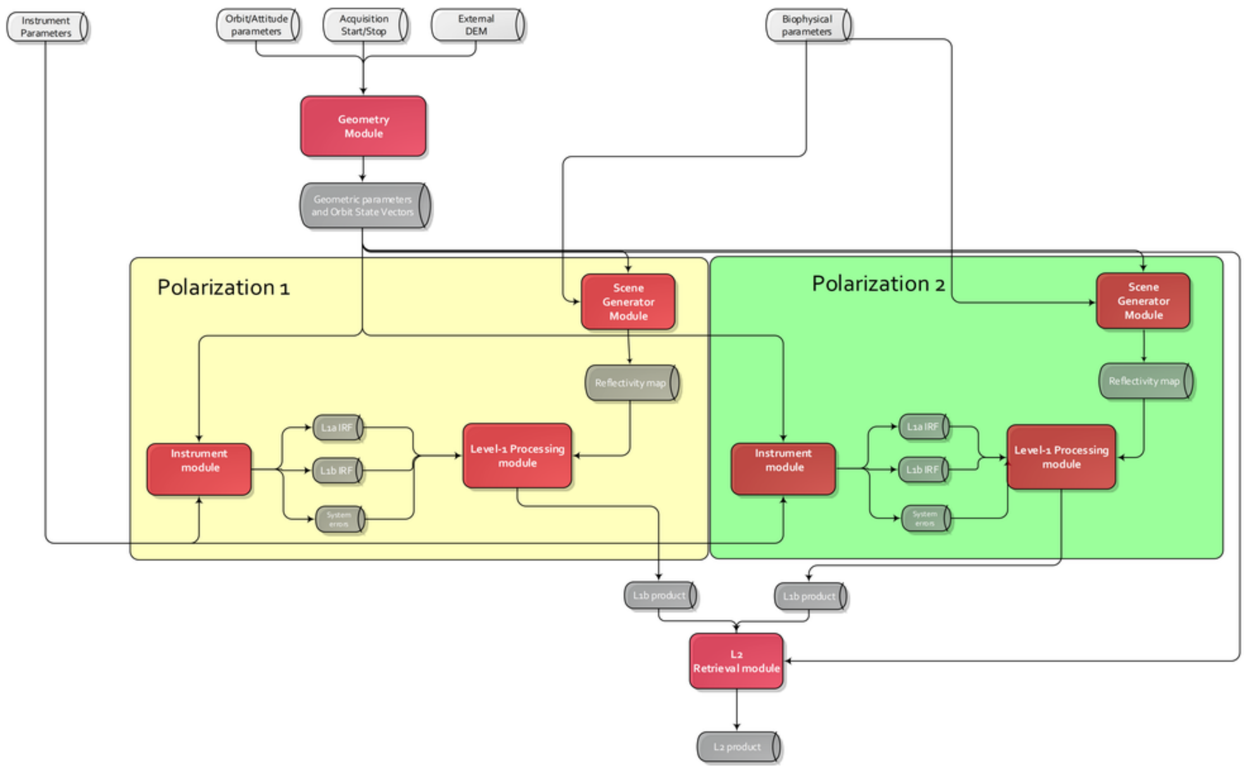


Figure 9-49: Reference architecture for multiple polarization acquisition

Multiple simulation chains are also required when the Level2 Retrieval Module needs Level1 products coming from different passes over the same area on ground to compute the Level 2 products. As an example this happens in multiple-pass interferometric or in multiple-pass PolInSAR applications. Basically, a data-flow for each pass has to be managed by the simulator. In that case, since in different passes the acquisition geometry will be probably different, it is needed to include the GM in the simulation chain related to each pass. This approach allows a more realistic and accurate simulation but requires to run twice the GM, that in turn involves an additional computation burden. The data-flow diagram representing the reference architecture discussed above for multiple-pass Level 2 applications is depicted in the following figure.

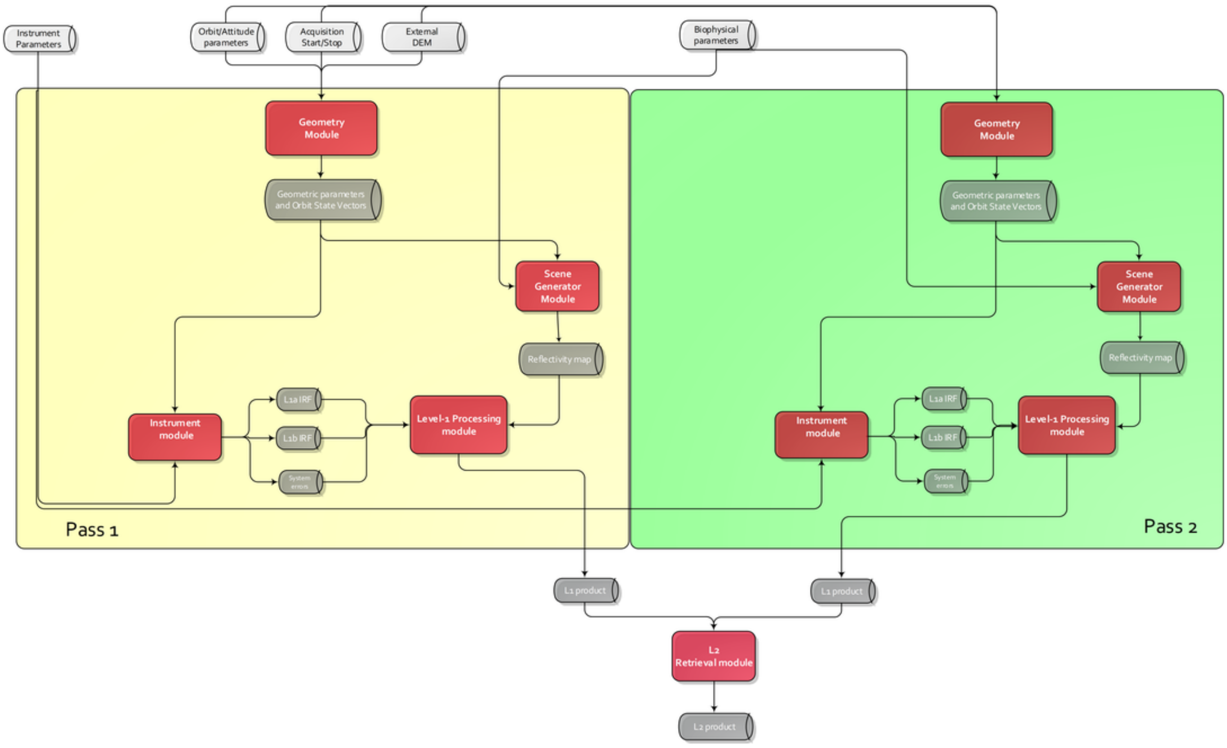


Figure 9-50: Reference architecture for multiple-pass Level 2 applications

A less accurate and with lower computational burden architecture can be introduced also in case of multiple-pass Level 2 applications. In fact, another approach could consists of executing once the GM and the SGM for both the passes at the cost of introducing an additional module that models all the changes that can happen on one Level 1 product before to be passed as input to the Level2 Retrieval Module. Reasonably this additional module, namely "Multilple-pass effects injection", will need the geometric parameters to compute the effects of a different pass on the Level 1 product. As an example, in case of an interferometric application, the Multilple-pass errors injection module should modify properly the phase of the SLC product produced by the L1PM and should introduce a shift in localization information contained in the product. It is worth underlining that the IM and the L1PM have to be executed one time for each pass in order to introduce errors uncorrelated between the two passes (as an example the additive noise realizations added in the two passes have to be uncorrelated). The data-flow diagram representing the simplified reference architecture discussed just above for multiple-pass Level 2 applications is depicted in the following figure.

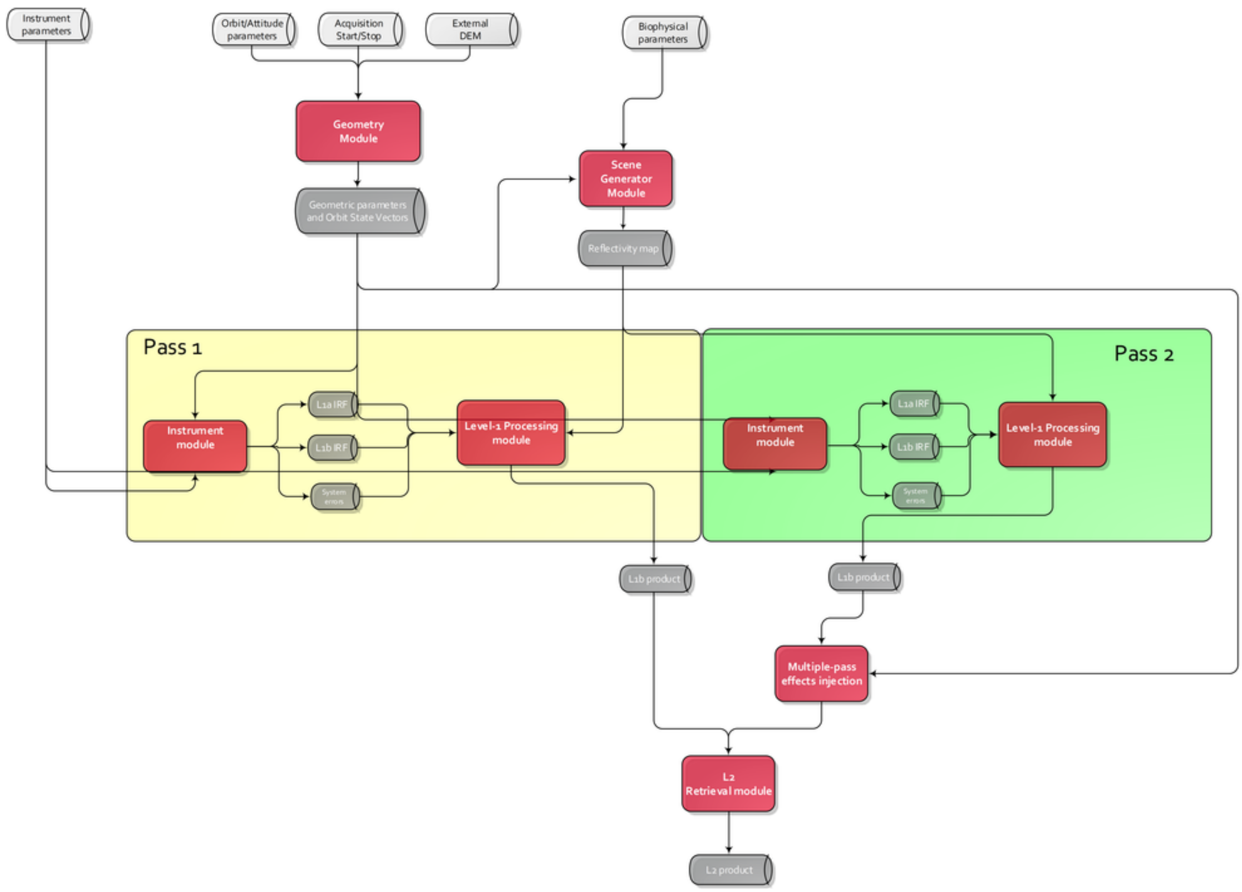


Figure 9-51: Simplified reference architecture for multiple-pass Level 2 applications

10. REFERENCE ARCHITECTURE FOR PASSIVE OPTICAL INSTRUMENTS

10.1. PROPOSED ARCHITECTURE

Passive optical instruments share many commonalities as it was reported in [PD.1]. These commonalities point out to a reference architecture for Passive Optical Instruments E2E Simulator. In this section, this reference architecture will be introduced by connecting the six proposed modules (Geometry, Scene Generator, Instrument, Level-1 Processing, Level-2 Retrieval and Performance Evaluation) one by one showing the data flow and their interaction.

Figure 10-1 shows the full reference architecture that will be explained in the following paragraphs with zoom-in views and highlights to show the working process and data flow.

In the following pages, the simulation data flow will be explained and general information about the functions of each module will be given pressing special attention to the inputs and outputs.

In sections 10.3 to 10.7, the different modules and building blocks of the E2E Mission Simulator for Passive Optical Instrument will be explained separately, determining needed inputs and generated outputs. In addition, the main operations at each building block will be described, proposing different methods, algorithms, modules, functions and software.

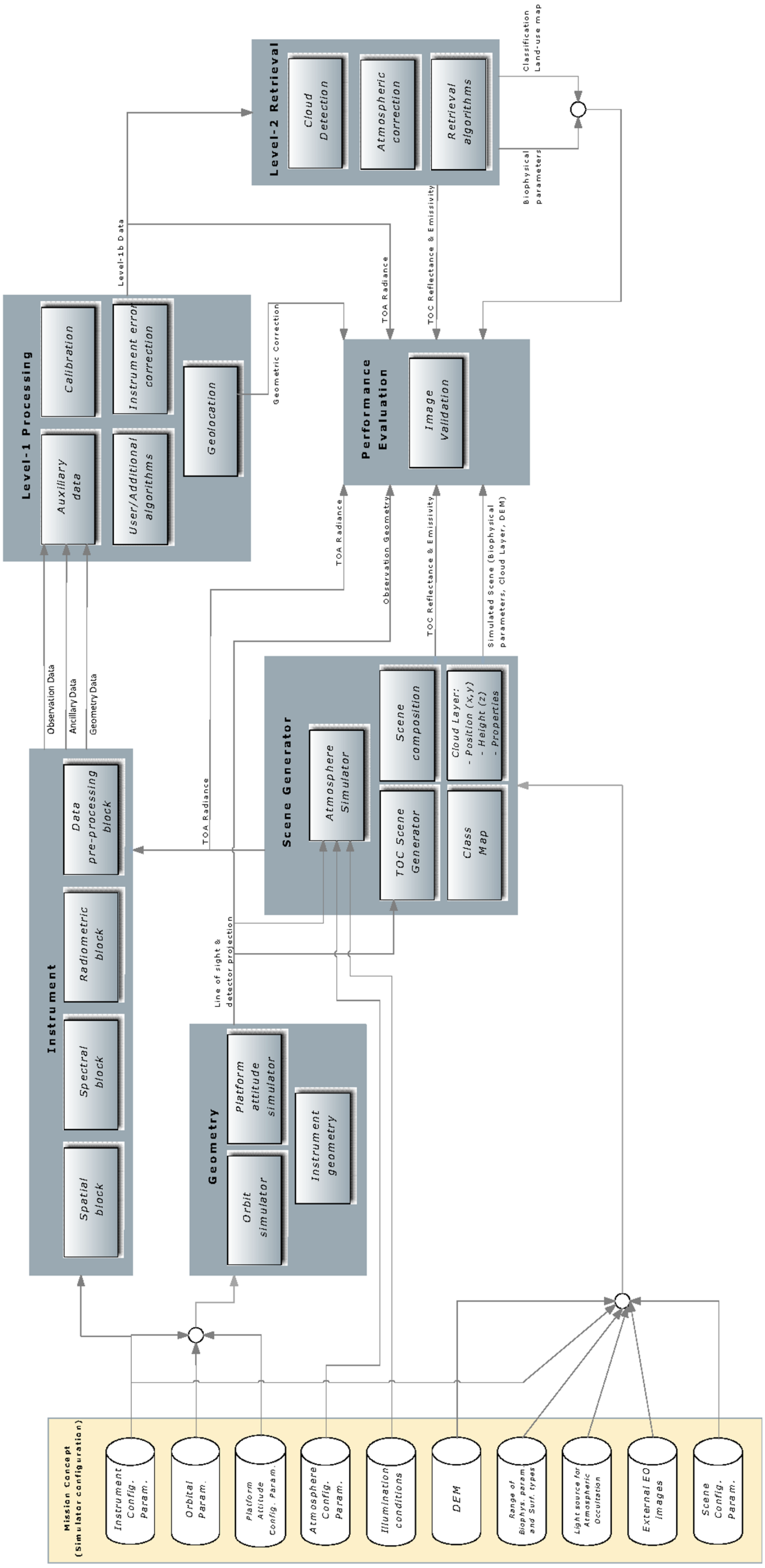


Figure 10-1. Reference architecture for the E2E Mission Simulator for EO Passive Optical missions.

Before starting a mission simulation, configuration parameters and other input data (e.g., EO images, DEM, orbital parameters, atmospheric parameters) for the different blocks should be introduced by the user to define the **mission concept** (yellow block in the diagram).

These parameters are then input (blue lines) in the *Geometry* block and the *Scene Generator* block which will be executed (red border) nearly in parallel (first the *Scene Generation* module will process the input data and parameters to generate the different scene layers, then the *Geometry* module will compute the observation geometry which will be again taken by the *Scene Generator* module to compute the output TOA Radiance).

The outputs of the modules are shown with green lines:Line of sight & Detector projection (observation geometry), TOC Reflectance, Emissivity, Temperature and Fluorescence (only for Earth surface pointing scenes), TOA Radiance and Simulated Scene (cloud cover, DEM and bio/geophysical parameters, including the atmospheric parameters).

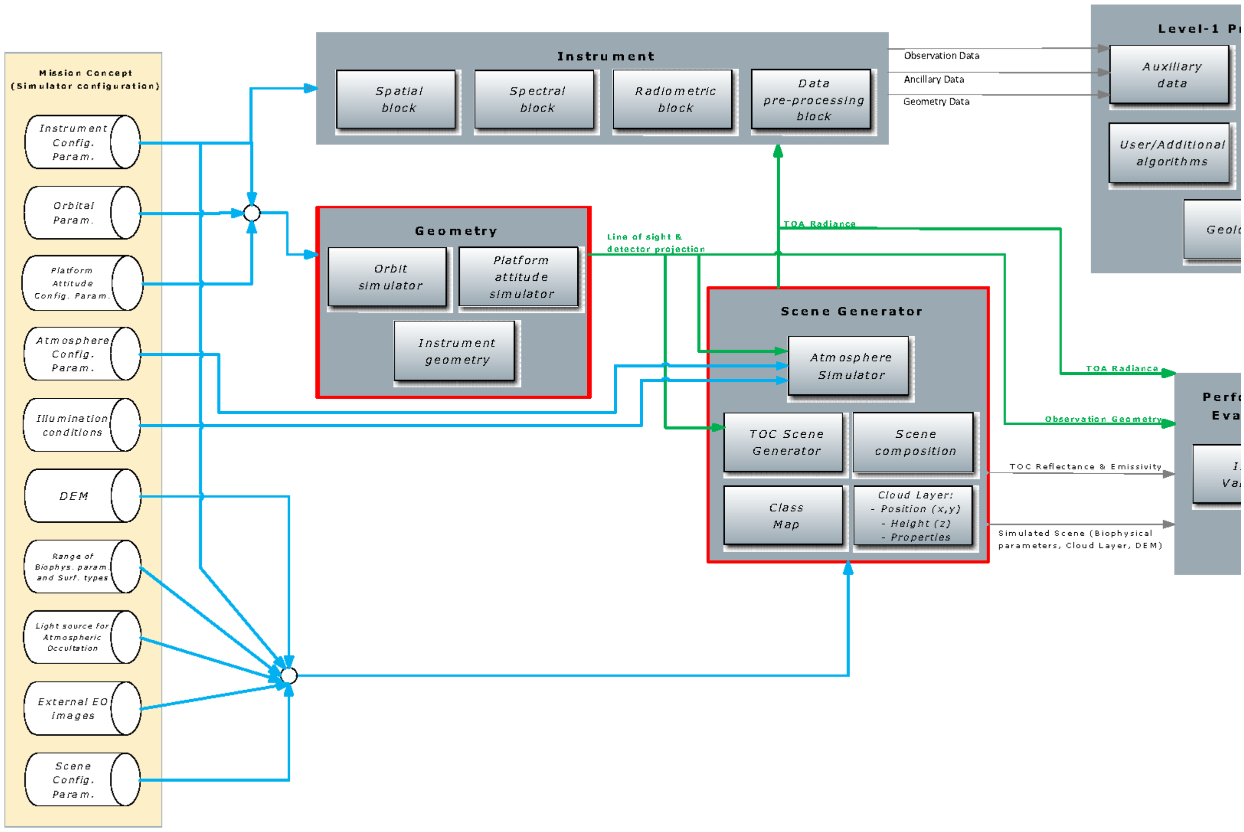


Figure 10-2. First step: Scene generation considering the observation geometry.

The following step in the simulation process encompasses the detection of the TOA Radiance by the instrument. The stimuli (TOA Radiance) and instrument configuration parameters are input into the *Instrument* module in order to generate the output Raw and Ancillary data. These output data will be later used in the *Level-1 Processing* module.

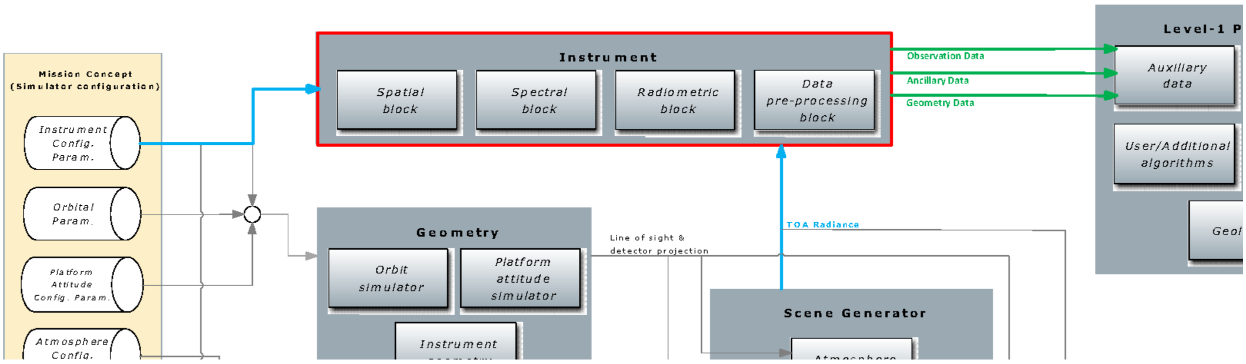


Figure 10-3. Second step: Instrument acquisition of the incoming radiance.

When the instrument provides the raw and ancillary data, the *Level-1 Processing* module will process this information together with the telemetry producing the output level-1 products, which includes the retrieved TOA Radiance that is geolocated, calibrated and error corrected.

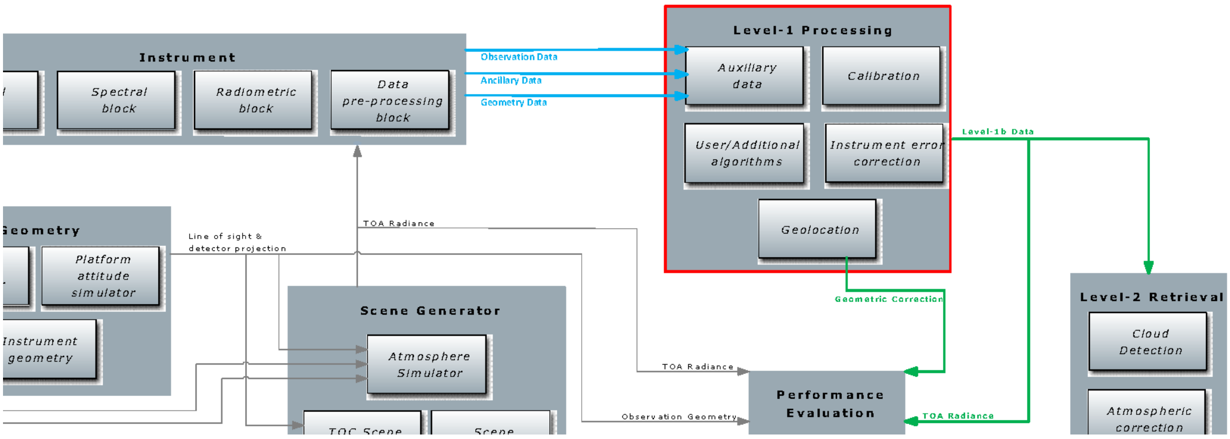


Figure 10-4. Third step: Instrument data processing at level-1 and retrieval of TOA Radiance.

As the first results are being obtained from the simulation process, the *Performance Evaluation* module will be executed, comparing the generated ("truth") with the retrieved Level-1 TOA Radiance. In addition, the geometric correction will also be evaluated by comparing the retrieved geometric correction against the "real" observation geometry from the *Geometry* module.

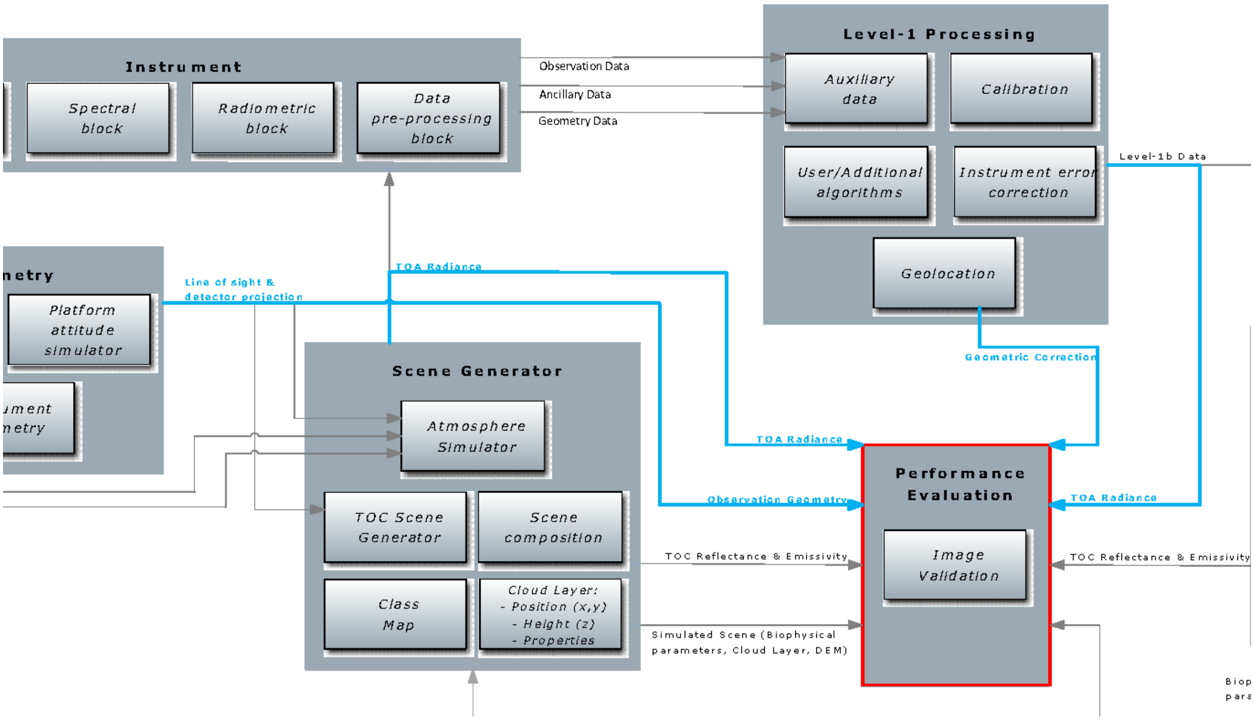


Figure 10-5. Fourth step: Performance evaluation of the TOA Radiance retrieval and geometric correction.

The fifth step performs the level-2 retrieval algorithms and, only for Earth surface pointing instruments, atmospheric correction. The *Level-2 Retrieval* module will use the level-1b data from the Level-1 Processing module and will perform the operations explained in section 10.6 to retrieve the TOC Reflectance, Emissivity, Temperature and Fluorescence as well as the bio/geophysical parameters (including atmospheric parameters or stereoscopic image) and Land-use map.

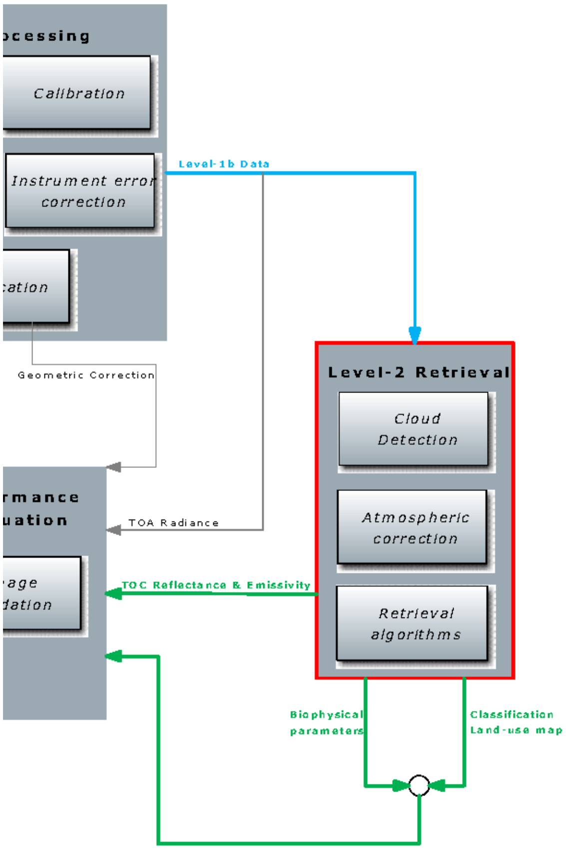


Figure 10-6. Fifth step: Retrieval of TOC image and biophysical parameters.

The output results from the *Level-2 Retrieval* module are finally evaluated in the *Performance Evaluation* module and compared with the equivalent parameters from the generated scene.

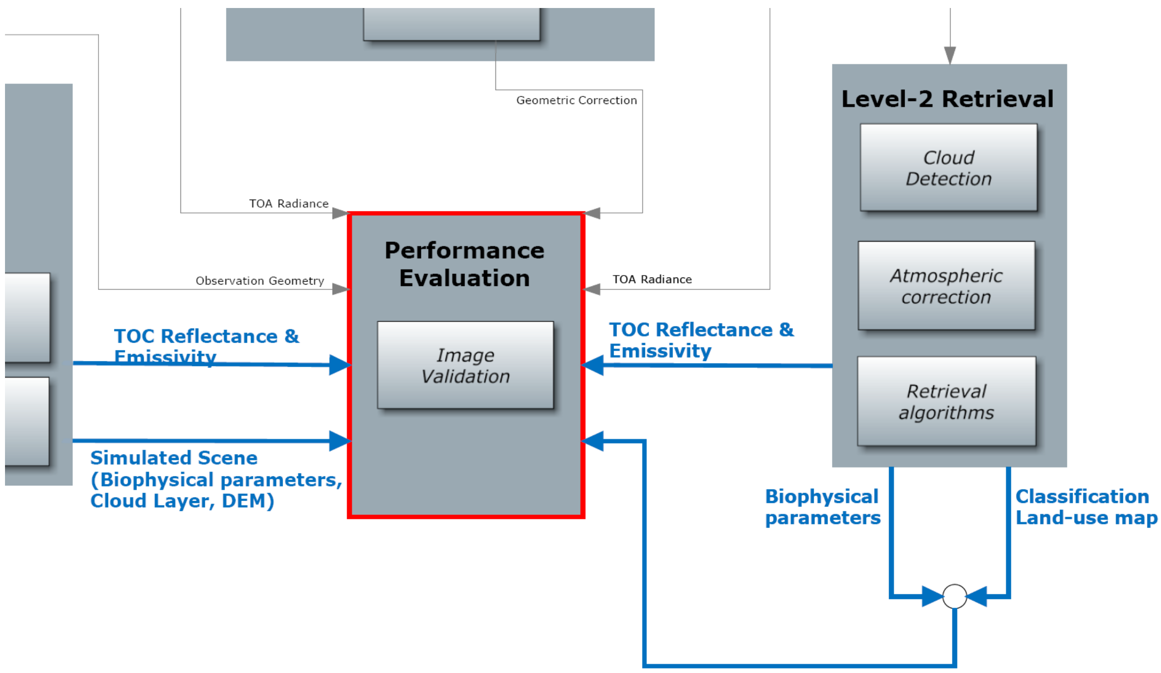


Figure 10-7. Sixth step: Performance evaluation of the level-2 products.

The process explained in the previous figures can be seen summarized in the following interactional diagram where the vertical dimension represents the time line. The operation appearing in the diagram and data flow in each module and building blocks have been summarized with respect to the functions explained in the previous sections.

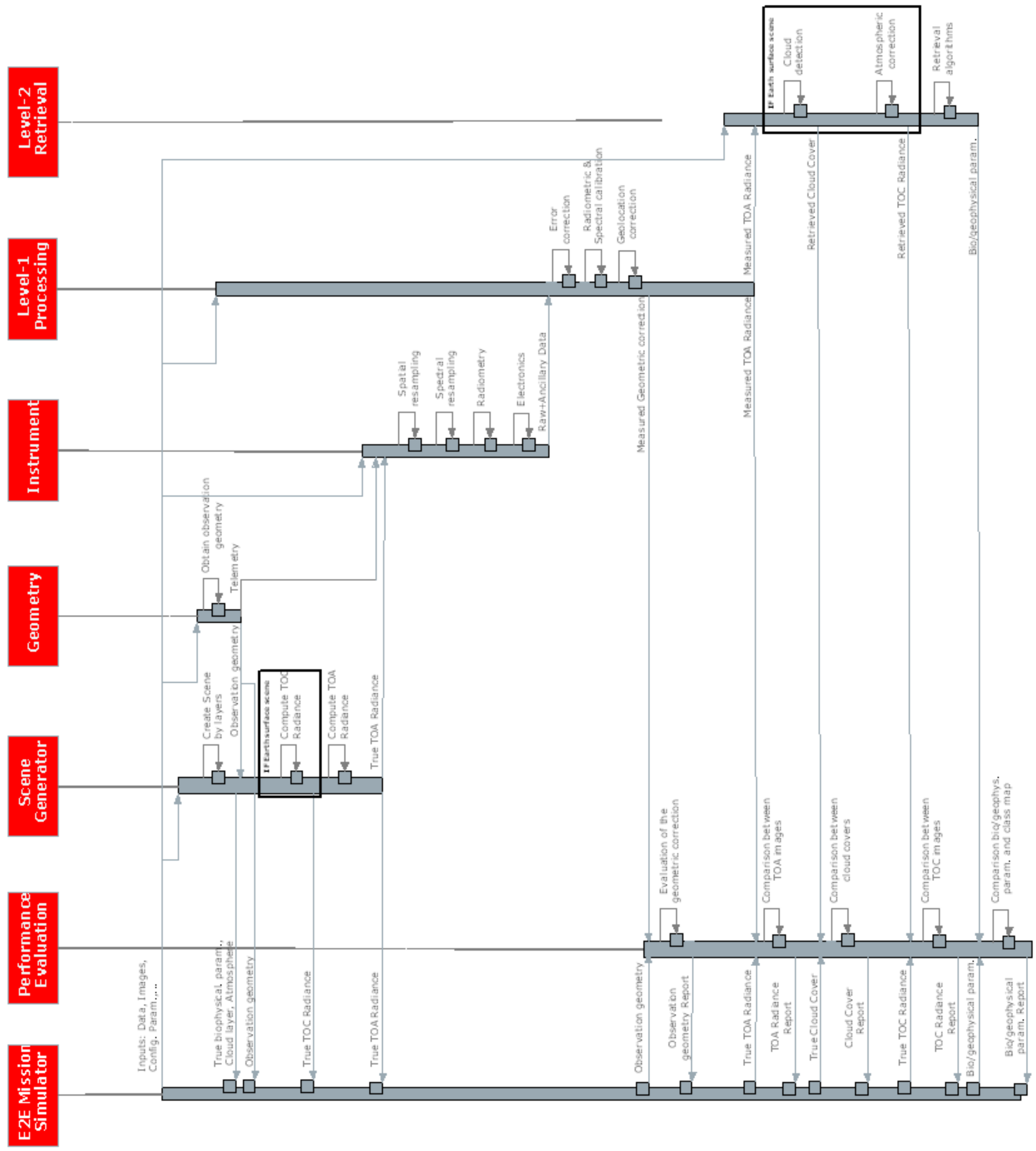


Figure 10-8. Interactional level of the E2E Mission Simulator for Passive Optical Instruments.

10.2. GEOMETRY MODULE

The *Geometry* module should provide with the instrument observation geometry. This observation geometry is given by the satellite orbital position, platform attitude and instrument attitude considering as well instrument characteristics, i.e. projection of the detector array over the surface/limb and other characteristics. According to this, this module should have the following three building blocks:

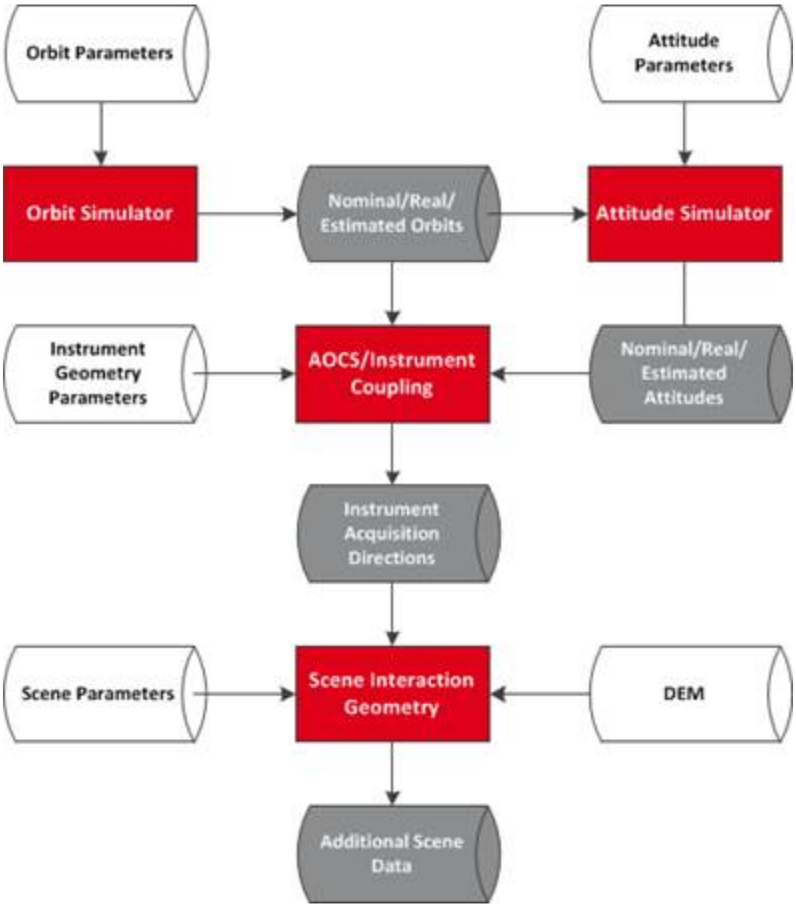


Figure 10-9. Architecture and building blocks of the *Geometry* module.

The *AOCS/Instrument Coupling* block will simulate the instrument attitude generated by instrument's optics and mechanical features (e.g., pointing mirrors, multiple cameras) coupled with the platform attitude. This block, together with the *Orbit simulator* and *Attitude Simulator*, will provide with the instrument acquisition directions which will be used in the *Scene Generator* module to calculate the Top Of Atmosphere (TOA) radiance for each particular pointing direction and Field Of View (FOV). In addition, the Scene Interaction Geometry will provide with additional scene data as illumination conditions after interaction with surface properties (e.g. DEM).

In addition, the *Geometry* module should provide with telemetry data that will be later attached to the instrument raw data in the Instrument Module and used at *Level-1 Processing* module for geolocation.

As inputs, the *Geometry* block would need the complete set of instrument configuration parameters, like instrument type (pushbroom, whiskbroom or frame), FOV of each pixel or detector integration time.

10.3. SCENE GENERATOR MODULE

10.3.1. PURPOSE AND SCENE CLASSIFICATION

The *Scene Generator* Module is one of the main constituents of a mission simulator. It allows defining the bio- and geo-physical/chemical characteristics of a scientific target area, considered to be the *truth* (or reality), that is going to be later observed by different instruments.

The generated scene should be simulated according to aim of the mission and the instrumental characteristics (spectral bands, resolution and Ground Sample Distance (GSD)). It is recommended to generate the scene with a **spectral and spatial resolution at least 10 times higher than the one of the instrument**.

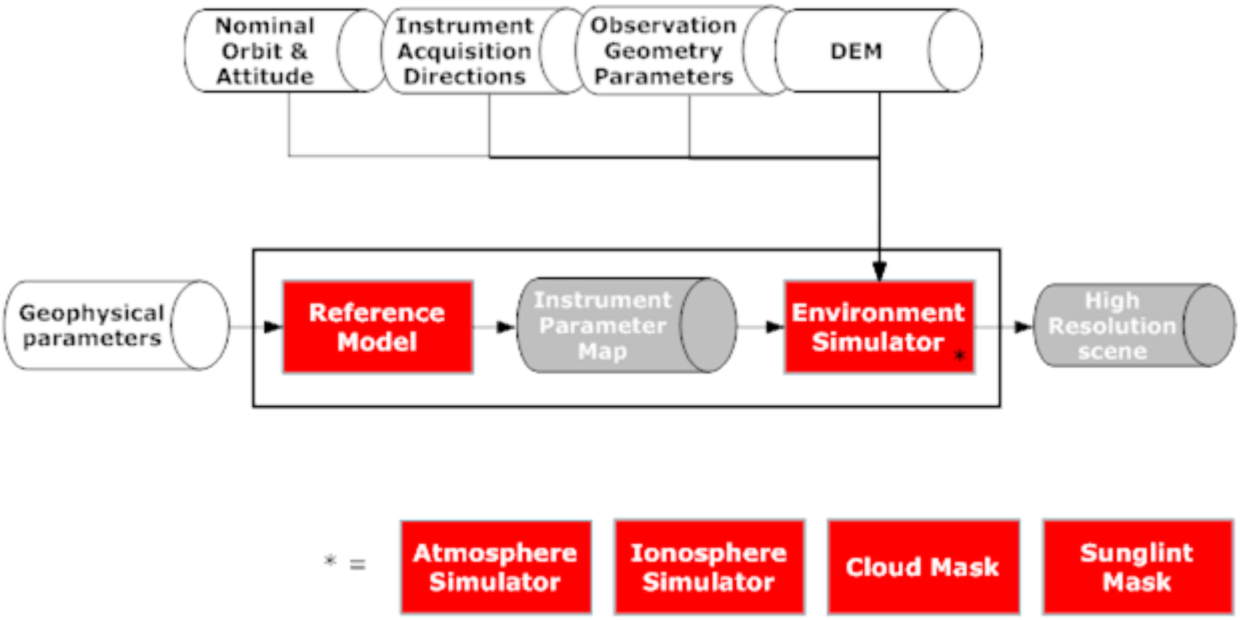
Furthermore, the type of target to be observed, i.e. land, snow/ice, atmosphere, ocean or inland waters, will define certain characteristics of the scene. After the work presented in [PD.1], it has been decided to differentiate between three types of scenes:

- **Earth pointing scenes:** These are all those scenes that focus on pointing towards the Earth's surface, mainly to retrieve characteristics of land (e.g., vegetation, land use, urban distribution/growing, fire detection or snow/ice cover among others), ocean (e.g., surface temperature or ocean colour) or atmosphere (e.g., cloud cover or aerosol distribution). The Earth pointing scenes will take into account surface elevation as well as atmospheric effects to generate TOA radiances.
- **Limb pointing scenes:** In this case, the scene is not generated by a surface but by defining the atmospheric characteristics (e.g., chemical species, water vapour, temperature profiles…). These scenes can also include a reference background spectrum (e.g., Sun, Moon or stars) that, by occultation behind the Earth's atmosphere, allows obtaining the atmospheric properties.
- **Calibration scenes:** Scenes for calibration (radiometric, spectral and dark current) of the instrument detectors.

10.3.2. MODULE DESCRIPTION

According to the above-described different possible scenes of a passive optical mission, the *Scene Generator* Module will consist of two main building block (*Reference Model*, *Environment Simulator*,) as shown in Figure 10-5. The use of some of these blocks will depend on the configuration parameters defining the mission concept. Furthermore, the user should have the possibility to input external, and previously developed, scenarios or part of them such as the distribution of biophysical parameters, cloud cover or atmospheric properties among others.

Figure 10-10. Structure of the building blocks of the Scene Genetator Module.



10.3.2.1. Limb pointing scenes

If a **Limb Pointing Scene** is to be generated, two options are foreseen: The first case computes the atmospheric radiance spectrum with the given satellite attitude pointing towards the limb, the second case modifies a light source spectrum (Sun, Moon or star) from a spectral catalogue after occultation behind the Earth's atmosphere.

This block should take the following input data:

- Observation geometry: The orbit position and instrument attitude will be given by the *Geometry* Module.

- Instrument spectral characteristics: The instrument's spectral range and resolution will be used to select the corresponding values to perform the simulation. It should hereby be kept in mind that the spectral resolution of each spectrum should be at least 10 times higher than the instrument's spectral resolution.
- Scene configuration parameters: Including the solar angles (or date of observation), observer's latitude/longitude, solar irradiance spectrum and atmospheric parameters (e.g., aerosols, temperature profile, ozone profile).
- Light source: This input is only needed in case the limb pointing scene performs the occultation of a light source. This source is then defined by its spectra and position and is provided by the Bottom Of Atmosphere (BOA) radiance. After the analysis presented in [PD.1], three possible reference spectrums (Sun, Moon or stars) taken from an external spectral library or input by the user can be used for the occultation technique.

This sub-block is limited by the spectral resolution of the atmospheric simulator, e.g., MODTRAN simulations are given with a spectral resolution up to 0.1 cm^-1, i.e. between 0.0016 nm at 400 nm and 0.06 nm at 2500 nm.

10.3.2.2. Earth surface pointing scenes

10.3.2.2.1. Scene Generator inputs

When the instrument points towards the **Earth surface**, the external data and configuration parameters are input to the *Environmen Simulator* block together with the atmospheric conditions and Digital Elevation Model (DEM) are input to the *Atmosphere Simulator* block. The DEM and illumination conditions should be taken into account for the correction the radiance from the shadowing effects. Furthermore, the observation geometry from the *Geometry* module is input to the *Environment Simulator*, calculating the surface reflectance, emissivity, temperature and fluorescence in the line of sight.

For the scene generation, the user should be allowed to input an already generated layer. For instance, the user can input a cloud cover layer into the *Environment Simulator* block or a distribution of predefined biophysical parameters into the *Reference Model* block.

10.3.2.2.2. Cloud layer and distribution of biophysical parameters

A *Cloud Maskrsub-* block generates a synthetic cloud layer. One possibility is to retrieve cloud cover from an external Earth Observation (EO) image by applying cloud screening algorithms using different bands, [RD. 14]. The generated cloud layer serves for projecting shadows on the Earth's surface and to mask those surface pixels covered by clouds.

The cloud cover will be defined by its position in the scene (x,y), height (z) and optical properties. It should be stressed that simulation of a realistic cloud cover and its interaction with light is computationally very demanding (e.g. by Monte Carlo simulations). One possibility could be the simulation of the cloud cover as a mask for occultation of surface pixels.

The *Reference Model* block distributes the bio/geo physical parameters on the scene. The class distribution on the scene can be obtained from the analysis of an external EO image, [RD. 15], based on other terrain properties such as altimetry, [RD. 16]. This block also adds a natural variability, randomly or based on different distributions (e.g., radial, sinusoidal…), to the biophysical parameters.

10.3.2.2.3. Obtaintion of surface reflectance/emissivity. Surface Radiative Transfer Models

Biophysical parameter distribution layers are appended in the *Reference Model* block together with the external input DEM

The Reference Model block should calculate the surface reflectance, emissivity, temperature and fluorescence in the line of sight (defined in the *Geometry* module) with *Radiative Transfer Models* (RTM). Table 10-1includes some RTM for surface reflectance, emissivity, temperature and fluorescence calculation.

This block selects the adequate surface RTM depending on the input parameters (e.g., class, spectral range, atmospheric parameters) and illumination conditions. For instance, Skyhelios, a commercial software, [RD. 17], calculates the Sky Viewing Factors (SVF) as function of the DEM and other natural obstacles as those that can be defined by an urban area.

The selection of a surface RTM depends on the observed target defined on the *Class Map* block, e.g., vegetation areas, ocean, ice and snow, urban areas. CEOS' EO Handbook, [RD. 91], proposes to classify the RTMs by 3 different surface types: Land (e.g., urban scenes, vehicles, vegetation, landscape…), Snow/ice and Ocean.

The following table provides a brief summary of the characteristics of some surface RTMs that can serve to generate a synthetic scene. Depending on the spatial resolution of the instrument, the scene should be generated in 1D or 3D accounting or not for shadowing effects. It should be noted that each model is limited by the spectral ranges and spatial resolutions. These characteristics should be taken into account to select among all the existing RTMs depending on the configuration parameters of the scene and the pixel class.

Table 10-1. RTM's characteristics for synthetic scene generation.

Model Name	Spectral Range	1D / 3D	Surface Type	Radiance/ Reflectance	Other characteristic	Ref.
OSIrIS	3-14 □m	3D	Urban Area	Radiance		[RD. 18]
SPIRou	3-12 □m	1D	Landscape	Radiance		[RD. 19]
MuSES	NIR-TIR	3D	Vehicles	Radiance	For active heat sources	[RD. 20]
PRISM	NIR-TIR	3D	Vehicles	Radiance	For active heat sources	
DART	UV-TIR	3D	Any surface	Radiance and reflectance	Ray tracing. Limited in thermal range	[RD. 21] [RD. 22] [RD. 23]
WASI	VIS-TIR	1D	Ocean color	Radiance and reflectance		[RD. 24]
MOX		1D	Ocean color	Radiance		[RD. 25] [RD. 26]
COART	UV-TIR	1D	Ocean color	Radiance		[RD. 27]
Mie theory based snow models	TIR	1D	Snow/Ice	Radiance		[RD. 28]
SAIL++	0.4-2.5 □m @1nm	1D	Vegetation	Reflectance	Dense homogeneous vegetation with random leaf distribution	[RD. 29]
ACRM	0.4-2.4 □m @1nm	1D	Vegetation	Reflectance	Nonlambertian soil BRDF, specular reflection of direct Sun radiation, hot spot effect, two-parameter leaf angle distribution (LAD).	[RD. 30]
MBRF		1D	Vegetation	Reflectance	Multicomponent canopies with non-uniformly vertical distribution of foliage area density, and non-random spatial dispersion	[RD. 31]
5-Scale	0.4-2.4 □m	3D	Vegetation	Reflectance	Non-random spatial dispersion with internal structures and biochemical properties	[RD. 32]
FLIGHT		3D	Vegetation	Reflectance		[RD. 33]
4SAIL2	0.4-2.5 □m @1nm	3D	Vegetation	Reflectance	Non-lambertian soil BRDF and the consideration of vegetation with ground or crown coverage	[RD. 34]
HYEMALIS	0.4-2.5 □m	3D	Landscape	Reflectance		[RD. 35]

Model Name	Spectral Range	1D / 3D	Surface Type	Radiance/ Reflectance	Other characteristic	Ref.
RGM		3D	Vegetation	Reflectance		[RD. 36]
Raytran		3D	Vegetation	Reflectance	Ray tracing	[RD. 37]
SOLENE		3D	Urban Area	Radiance and reflectance		[RD. 43] [RD. 44]

Some of the most commonly used RTMs for the simulation of vegetated scenes are described in the European Commission Radiation transfer Model Intercomparison (RAMI) webpage, [RD. 38]. In RAMI, these models have been compared and their characteristics and limitations are documented in detail.

Other models that simulate urban areas are explained in [RD. 39] (for infrared scenes) and [RD. 40] (for reflectance retrieval).

Ocean scenes can be generated by different methods. Ray-tracing and geometric models such as [RD. 41] are usually applied to create computer animations that can also be used to generate coastal images for EO applications. Water-leaving spectral radiances simulated for clear ocean waters have also been developed, [RD. 59], e.g., in the view of the MERIS mission simulator, [RD. 60].

10.3.2.2.4. Generation of TOA radiance

Surface RTMs in the *Reference Model* block generate, based on the input parameters, the Top of Canopy (TOC) and surface reflectances (dominant in the visible and Near InfraRed (NIR) range), emissivities (dominant in the Mid-InfraRed (MIR) and Thermal InfraRed (TIR) bands), fluorescence (e.g. vegetation fluorescence in the Visible) and temperature. These information is used in the *Environment Simulator* block to compute surface leaving radiance and propagate through atmosphere obtaining the output TOA radiance.

10.3.2.2.5. Earth surface scenes. Summary and outputs

The configuration parameters and other data and information are input to each . The *Reference Model* block creates a surface cover classification map and adds natural variability to the biophysical parameters that are delivered as an output of this block for each scene pixel. All these parameters are introduced in the adequate RTM obtaining the surface reflectances, emissivities, temperatures and fluorescence. The Environment Simulator block will run in parallel a *Cloud Mask* sub-block providing with the cloud cover. These layers are composed together with an external DEM to create the simulated scene.

Due to the large amount of parameters defining the scene (biophysical characteristics, atmospheric parameters, cloud layer…), the generated scene is divided into units with same characteristics. The *Atmosphere Simulator* block computes the TOA radiance based on the output from the *TOC Scene Generator* block together with the DEM and cloud layer. It also considers the given sensor geometry and applies the shadow effects due to cloud cover, terrain altimetry and illuminating conditions.

Scene Generator outputs

One of the outputs from the *Scene Generator* module, the generated scene TOA radiances, serve as an input for the *Instrument* module and the *Performance Evaluation* module for a later performance evaluation of the *Level-1 Processing* module's TOA radiances.

The TOC Reflectance, Emissivity, Temperature and Fluorescence output from the *Scene Generator* module will later be used in the *Performance Evaluation* module to analyze the performance of the level-2 retrieved TOC Reflectance, Emissivity, Temperature and Fluorescence.

Furthermore, the distribution of biophysical parameters will be taken by the *Performance Evaluation* module and will be compared with the biophysical parameters obtained in the *Level-2 Retrieval* module. The same approach will be done with the cloud layer to analyze the performance of the cloud detection algorithms in the *Level-2 Retrieval* module.

10.3.2.3. Calibration scenes

Calibration scenes are a particular case of the *Scene Generator* module. The study presented in [PD.1] showed that calibration scenes are characterized by homogeneous at-instrument radiance with a known spectrum.

All these calibration scenes are defined on the *Reference Model* block (e.g., a homogeneous white background for a solar diffuser scene, deep space background, lunar radiance/irradiance models, [RD. 92]). Vicarious calibration scenes are generated following the same scheme followed in section 10.3.2.2.2.

10.4. INSTRUMENT MODULE

10.4.1. INTRODUCTION

Instruments are defined by their geometric parameters (e.g., pixel size, number of pixels, instrument type, FOV), spectral parameters (spectral bands and response) and instrumental noise parameters (e.g., dead pixels, cross-talk, vertical striping) that depend on their instrument type (pushbroom, whiskbroom or frame) and detector type (e.g., CCD array, InGaAs detectors, HgCdTe detectors, photodiodes, microbolometers, QWIR detectors). Nevertheless, the acquisition of the TOA image implies that the geometric, spectral and instrumental-noise features are applied at different points of the simulation process.

While the *Geometry* module calculates the instrument observation geometry (explained in Section 10.3), the *Instrument* module considers only the spectral and spatial resampling together with all the noise produced in the instrument. In addition, the *Instrument* module transforms the measured TOA radiances to Digital Numbers (DN) and makes the analog-to-digital conversion together with other data pre-processing such as binning or data compression.

The *Instrument* module involves all the necessary calculations with respect to the specific configurations of the considered optical sensor. These sensor-specific settings are organized and tackled into 3 building blocks: *Scene Interpolator*, Radiometric *block,* and *Data pre-processing*.

10.4.2. INSTRUMENT MODULE: DESCRIPTION OF BUILDING BLOCKS

The Instrument module's building blocks are described based on the approach taken for the EnMap mission simulator, [RD. 15] [RD. 91].

- The *Scene Interpolator* block performs the spatial and spectral resampling to the final ground sampling distance and instrument spectral response, taking into account the real instrument Modulation Transfer Function (MTF), spatial non-uniformity sources (spectrometer co-registration, keystone, and telescope smile and distortion), spectral response function and spectral non-uniformity noises.

  This block records the pixel information simulating a flight over a 2D or 3D artificial radiance image scene. This process is characterized by the convolution of the spectral surface information with filter functions along and across orbit representing the sensor specific Point Spread Functions (PSFs). The characteristics of the wavelength depending PSFs are defined by the MTF in the Fourier domain incorporating the optical, detector, vibration, and motion MTF of the satellite.

  The *Scene Interpolator* block affects to the spatial domain of the image, which is correlated with the instrument type (whiskbroom, pushbroom or frame). Conceptually, it can be considered that pushbroom and frame instruments are particular types of whiskbroom instruments in which the acquisition time of a group of pixels (line in pushbroom instrument or 2D array in frame instruments) are taken at the same time instead of different times. With this concept, the spatial block will be common for every instruments type despite the particularities of each instrument will require to apply different spatial non-uniformity aberrations like keystone, smile, distortion and co registration.

31	32	33
21	22	23
11	12	13

Figure 10-11. Scheme of a 3x3 array of pixel areas. In a whiskbroom instrument each pixel area is detected at a different time. On a pushbroom (frame) instrument each line (the whole array) is acquired at the same time.

Due to the different acquisition times in whiskbroom, pushbroom and frame instruments, the *Geometry* module will be in charge of computing the observed surface area by each pixel element.

In addition, the *Scene Interpolator* block performs the spectral resampling from the finer working resolution of the scene generator to the spectral configuration of the sensor. This involves the simulation of the instrument spectral response functions, the spectral resolution and spectral sampling interval, as well as the spectral non-uniformity, given by smile and spectral shift. For sensors covering the entire Visible Near InfraRed (VNIR) – Short Wave InfraRed (SWIR) spectral range, the spectral module should be implemented so that the spectral range can be separated into two regions, each one with a different spectral performance, in order to simulate separate sensors. A key parameter to be analyzed and optimized when designing a space-borne sensor is the spectral curvature or smile effect. Smile absolute magnitude, smile band-to-band variations, and systematic spectral should be considered as sensor-dependent parameters in this block. The resampling of the working resolution TOA radiance to the final spectral configuration should be performed on a per-column basis in order to include spectral smile.

The acquisition of the spectral domain for the observed spatial elements (single pixel, line or frame) changes for each instrument design, spectroradiometer type and spectral mode therefore this block should perform different operations to retrieve the image in the spectral domain. The following commonalities have been found among every particular case and they require for a different approach on the *spectral* building block:

1. In terms of how the spectrum is obtained for each pixel prism and grating spectrometers can be represented with the same function.

   In general, the optical system, which includes grating, prism and/or filters, will guide the incoming light to different detectors (e.g., CCD array, HgCdTe detectors, photodiodes) in the focal planes that will record the spectral bands according to certain instrument parameters (e.g., sampling intervals, spectral resolution, spectral response function). This block will therefore define the spectra acquired by the different detectors in the instrument

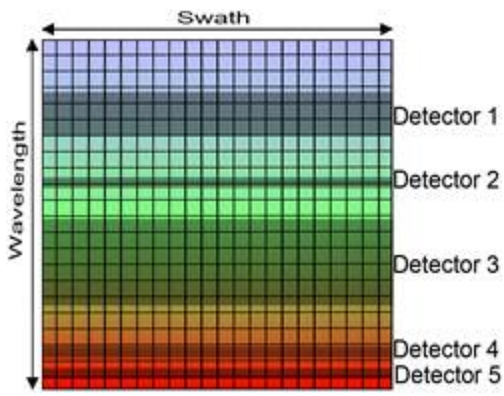


Figure 10-12. Scheme of the spectra diffracted by a pushbroom Prism or Grating Spectrometer and the multispectral or hyper/ultra-spectral bands detected by different detectors.

2. Fourier Transform Spectrometers (FTS) should be considered in a different sub-block as their spectral domain differs from the other spectrometers. For each spatial dimension, an interferogram is obtained for each difference of optical path between the two arms of the FTS so the imaged surface area is slightly different as the satellite advances in the orbit.

3. Two radiometers types will use optical filters to detect the spectral band. Multi-camera and Filter Wheel radiometers can also be considered to be the same for the *spectral* block as the only differences will be the acquisition time at each band and spectral characteristics for the detectors are each band.

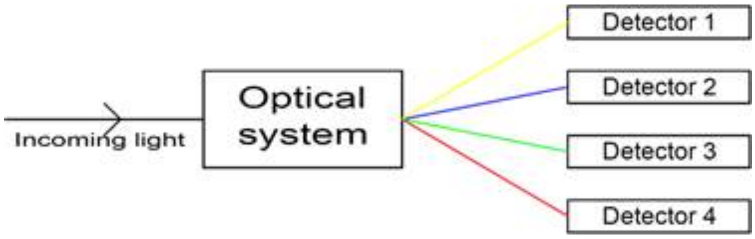


Figure 10-13. Scheme of a multispectral multicamera or filter wheel radiometer. In case of a Filter Wheel radiometer, each detector in the figure is actually the same and the different bands are acquired at a different time.

4. During the work reported in [PD.1], it was detected a particular case of spectrometer, MOPITT, on board of Terra mission. This *Gas Correlation Spectrometer* acquires the spectrum based on the gas correlation technique schematized on the following figure:

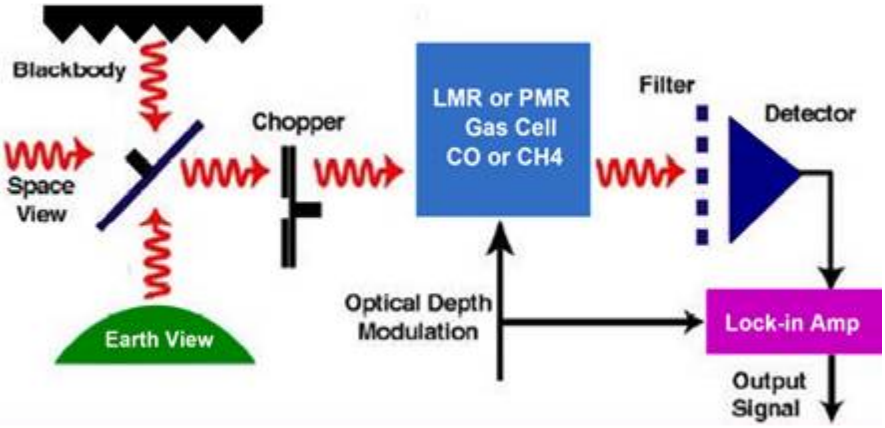


Figure 10-14. Schematic view of the gas correlation concept. Credit: NCAR/University of Toronto.

The radiance passes through a Length Modulated Cell (LMC) or Pressure Modulated Cell (PMC) where the length or pressure is changed. This variation produces a modulation on the cell opacity at the spectral absorption lines of the target gas while the cell opacity at other wavelengths remains constant, [RD. 67].

It is to be discussed if particular instruments like MOPITT should be considered in a different sub-block in the *spectral* building block. A proper and established definition of the inputs/outputs of the *spectral* building block would allow the implementation of new instrument types.

In addition, the *spectral* block should add different spectral aberrations (e.g., spectral smile, spectral shift…) corresponding to each particular type of instrument. These aberrations will modify the spectral acquisition from the ideal case.

- The *radiometric* block can be split in two sub-blocks: The A/D Converter block transforms at-sensor radiance to DN considering various sensor parameters such as integration time or quantum efficiency. The Electronic Noises sub-block adds shot and quantization noise, variable high and low gains, random dark signal and variable non-linear detector responses, and pixel to pixel differences (pixel uniformity). In the absence of information on the instrument electronics, multiplicative Gaussian-distributed noise processes are assumed, whose magnitude is given by the required Signal-to-Noise Ratio (SNR). The simulator should encompass the options to include the following kind of noise: spatially coherent noise (striping), dead pixels (those providing a constant readout) and bad pixels (those providing a wrong readout).

  The architecture of this block will be the same for every sensor but the effects here applied will be adapted according the different instrument and detector characteristics defined as input instrument configuration parameters.

- The *data pre-processing* block transforms the signal captured by the sensor taking into account the Analog-to-Digital (A/D) conversion, and any further on-board processing that might take place, such as binning or data compression.

  The steps taken in this block will be common for every instrument after the definition on their configuration parameters.

Furthermore, the *Instrument* module will provide ancillary data in the same manner that the ground segment would receive from telemetry and system characterization. Only the mandatory ancillary data for the L0-L2 processing, such as orbit and pointing for geometric correction, or sensor temperature for the noise reduction, will be included.

10.4.3. INSTRUMENT MODULE: DATA FLOW

The data flow in this module (see Figure 10-15) is linear so each building block is executed with the output information obtained from the previous block.

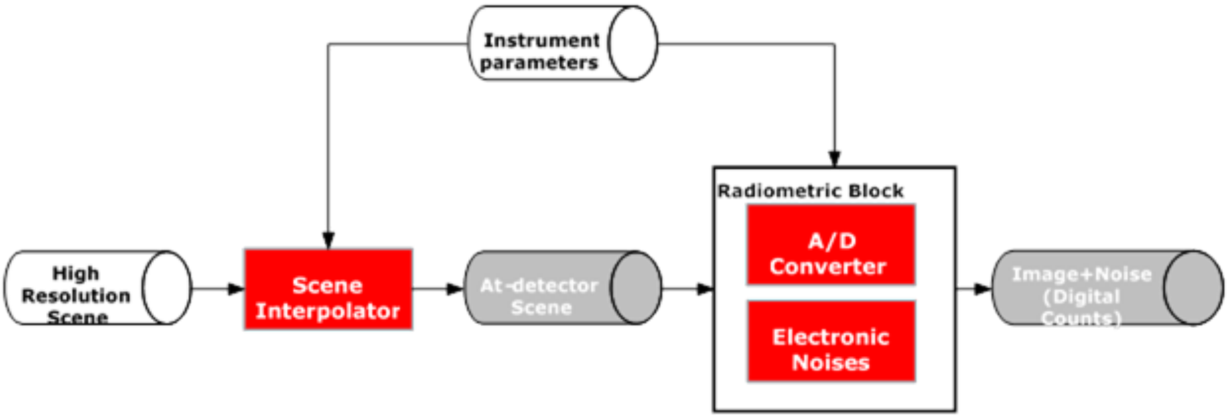


Figure 10-15. Behavioral diagram of the Instrument Module.

10.5. LEVEL-1 PROCESSING MODULE

10.5.1. INTRODUCTION

After the analysis and study presented in [PD.1], it was concluded that the data processing at level-1 can be considered to be similar for every passive optical mission. The raw data produced at the *Instrument* module is processed, retrieving radiometrically calibrated, geolocated, and noise-corrected images.

Furthermore, a cloud cover mask can be obtained and applied as well as image quality identifiers can be attached. In addition, preliminary pixel classification (land, water or clouds) is given for further processing at level-2.

In the following section, the proposal of block description and behavior will be explained showing the limitations and alternatives when possible.

10.5.2. PROPOSED MODULE ARCHITECTURE

The data processing at level-1 has been assumed to be nearly the same for each passive optical mission. Auxiliary data (e.g., instrument telemetry) is attached to the instrument raw data, which is pre-processed by compressing or unpacking it. All this information is then processed at three levels: 1) Calibration, 2) Geolocation and 3) Instrument error correction.

In order to allow further data processing, particular of each mission, it is proposed to consider an extra block, *User/Additional Algorithms*, to run user defined algorithms (e.g., pixel pre-classification as land/water masks, alternative geolocation algorithms developed by the user).

This module and its building blocks are schematized in the following figure:

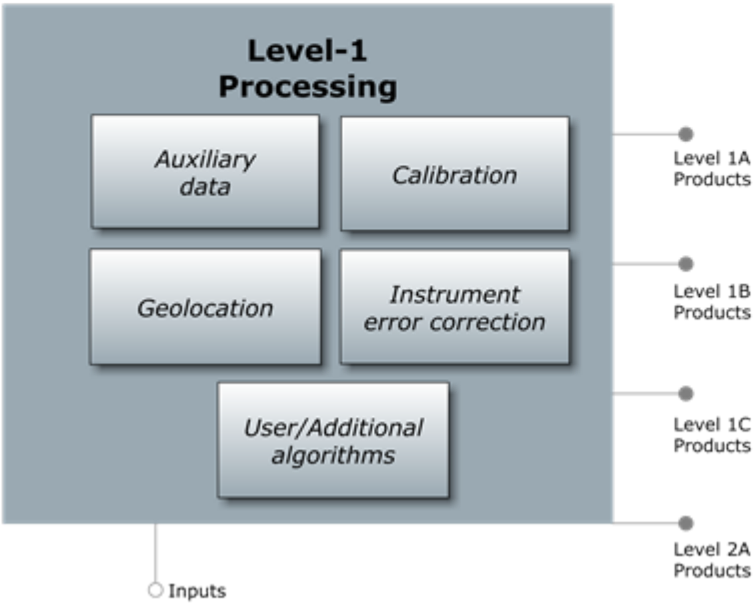


Figure 10-16. Structure of the Level-1 Processing module and its building blocks.

Figure 10-17, [RD. 45], summarizes the end-to-end level-1 processing flow for ASTER instrument onboard of Terra and illustrates the approach considered for the *Level-1 Processing module*.

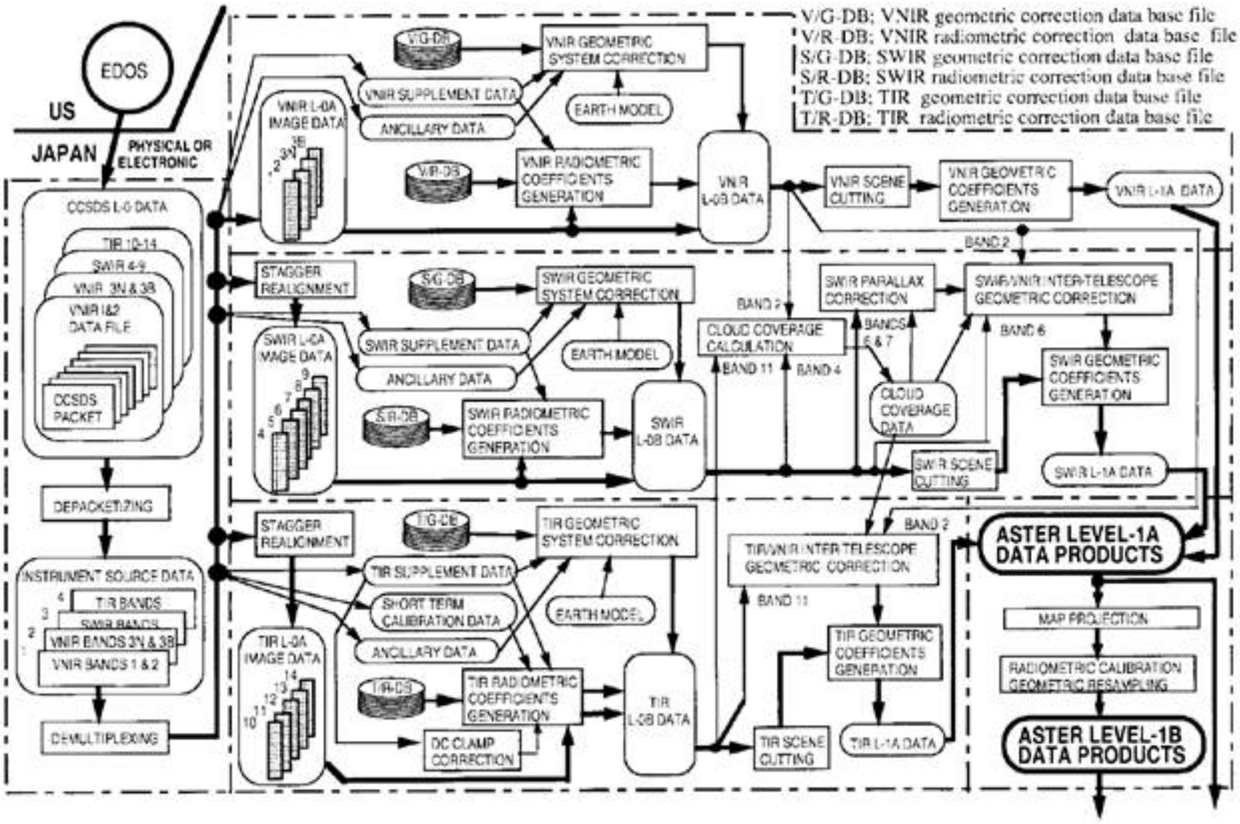


Figure 10-17. End-to-end level-1 processing flow for VNIR, SWIR and TIR channels on ASTER/Terra.

In this particular example, raw data is first compressed in a particular format to be sent to another processing module. Then the data is uncompressed recovering the information at each spectral band.

Each instrument raw data processed together with instrument supplementary data and satellite ancillary data to generate geometric correction and radiometric coefficients generation conforms level-0b data. With the calculated cloud mask, parallax and system geometric corrections, ASTER data at level-1a is produced. The radiometric calibration, map projection and geometric resampling generate level-1b data products.

A similar processing chain is given for DLR's EnMap mission, [RD. 46]:

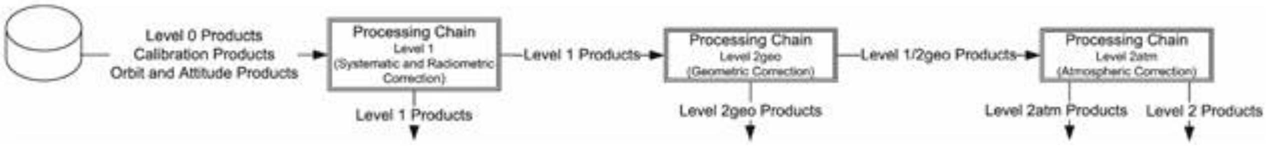


Figure 10-18. Processing chain for level-1 and level-2 products for EnMap mission.

Despite the different product level definition for EnMap mission products respect to European Space Agency's (ESA) standard definition, the Level-1 data processing chain is common at level-1. At level-0, raw data, calibration products and geometric (orbit and attitude) products are collected and the quality of the acquired data is appended. These data is then used to calibrate the raw data in radiance units and correct from instrumental errors and noises (e.g., radiometric non-uniformities, saturated pixels, bad/dead pixels, dark current). Furthermore, a preliminary land/water mask is applied.

In addition, geometric correction obtains ortho-rectified images at level-1b (level 2geo in EnMap product level definition). Orbit and attitude data are considered to obtain Ground Control Points (GCPs) and a DEM used to project the images.

Another example of the level-1 data processing flow, where the data is corrected from instrumental noises, radiometrically calibrated and geolocated, is the concept for SCIAMACHY instrument on board of ENVISAT mission, [RD. 47].

The proposed level-1 behavioral diagram, shown in Figure 10-19, is based on these examples that are generally repeated for every passive optical instrument.

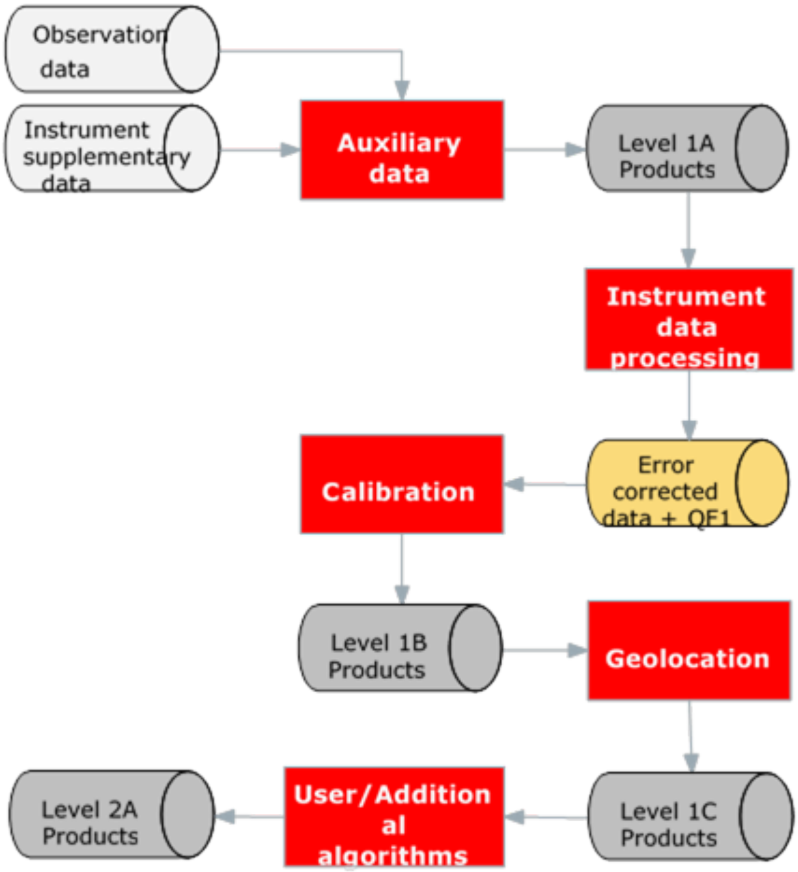


Figure 10-19. Behavioral layer of the Level-1 Processing module.

In addition, it should be considered the possibility to implement an Archiving & Catalog of the output level-1 products for future use.

10.5.3. DATA FLOW AND BUILDING BLOCKS

The instrument raw data together with instrument supplementary data and ancillary data is inputted into the *Auxiliary data* block in which geometric correction and radiometric calibration parameters are calculated and appended to the raw data for further processing defining the **Level-1a products** that will be given as an output of this block. In addition, previous unpackettizing of the data is performed if needed.

10.5.3.1. instrument data processing block

Level-1a products are firstly processed by the *Instrument data processing* block that aims to eliminate or reduce the effects of systematic instrumental spectral and spatial noises. The algorithms here applied greatly depend on the design Phase of the instrument, instrument type and detector type. The following list shows some of the identified systematic and instrumental noise corrections that should be considered on the generic mission simulator:

- Saturated pixel detection.
- Bad and dead pixel detection by applying a mask previously obtained on-ground calibration.
- Non-linearity correction in the spatial and spectral correction of the sensor which can be measured during on-ground callibration.

- Offset dectection and substraction.
- Dark Current detection and substraction. The dark current depends on instrument and sensor characteristics (e.g., analog offset independent of the integration time, leakage current, growth of the ice layer on the detector or background thermal signal for Infrared (IR) channels [RD. 47]).
- Correction from the non-uniformity of the sensor response in the spatial and spectral directions.
- Spectral and spatial stray-light correction.
- Smile correction.
- The socalled "Memory effect" where a previous readout deviates the signal from the linear response, [RD. 49]. On paper [RD. 47], white light source measurements and dark current are taken into account to substract this effect.
- Correction of polarisation, which mainly applied to all grating spectrometers, by determination of the instrument polarization sensitivity and determination of the Stokes vector. Reference [RD. 47] proposes a method of polarization correction (validated in [RD. 51]) based on four basic steps: 1) Determination of polarization reference frame; 2) Determination of instrument polarization sensitivity; 3) Calculation of polarization of atmospheric light and 4) Interpolation of Stokes vector to a full wavelength grid.
- Spatially coherent noise like the vertical stripping typically found in pushbroom sensors, [RD. 50].

All these noises and effects were also previously considered and simulated in the *Instrument* module.

In the *Instrument data processing* block, the user can select the error correction algorithms to be applied among the standard, or previously developed, algorithms from a library. In addition, user-defined algorithms can be included at Phases C and D of the mission development keeping the same simulator architecture.

In addition, in case of FTS, the recovery of the spectra from the interferogram level-1a data is done in the *Instrument data processing* block (see Figure 10-21). Some examples of this spectral recovery from the interferogam are shown on [RD. 48], [RD. 53] and [RD. 54].

The *Instrument data processing* block will also attach a quality flag for the correction algorithms in order to track the goodness of these corrections and perform a further processing of the images.

10.5.3.2. calibration block

With the data corrected from sytematic errors, the *Calibration* block is the following block to be used (see Figure 10-20 and Figure 10-21). This block converts the instrumental data from DN, also called Binary Units, into radiance units.

The calibration process and applied algorithms depend on the instrument type and mission design, nevertheless, as some of the steps are common for every passive optical instrument, the same architecture can be taken at different mission Phases. To illustrate this, different, and complementary, radiometric calibration mechanisms have been found in the literature:

- Prelaunch instrument calibration on-ground as instrument sensor linearisation (e.g., [RD. 54] and [RD. 56]) or spectral calibration for Fourier Transform Spectrometers (e.g., [RD. 53], [RD. 54])
- Vicarious calibration: Radiometric calibration by measures of particular sites on Earth surface (e.g., over desert, [RD. 52]).
- Calibration by pointing the instrument towards a reference light source like the Sun, spectral calibration of ACE-FTS [RD. 48], or Moon.
- On-board calibration using a cavity blackbody and/or deep space measurements, [RD. 53] and [RD. 54].

Two parallel and interrelated calibration processes need to be done to obtain the TOA radiance. One of them consist into a radiometric calibration in which DN are converted into radiance units, the other process consists into a spectral calibration assigning the frequency scale to the spectral points.

In any of these cases, the input *Instrument Supplementary Data* coming from on-ground or in-flight calibration will be applied in different algorithms to restore the spectral radiance values for the image. As for the *Instrument Error Correction* block, the *Calibration* block will also attach quality flags indicating, for example, when the calibration data is missing or when the Noise-Equivalent Signal Radiance (NESR) exceed a preselected threshold, [RD. 53].

A previous simulation must be previously run so the calibration coefficients are determined and stored for their later use in the callibration block.

10.5.3.3. Geolocation block

When the image is radiometrically calibrated, the data is then processed by the *Geolocation* block (see Figure 10-20 and Figure 10-21) which performs all the geometrical corrections. The required input data will come from the *Geometry* module via the Instrument Model, such as platform attitude, pointing vectors, satellite position, and configuration parameters from the instrument design. Among the instrument design parameters, there are all those geometric parameters such as the detector's line-of-sight vector, pointing axes and conversion factors (instrument parameters calibrated on-ground), [RD. 45]. Depending on the mission Phase, these parameters will be specified (Phases C/D) or assumed to a nominal value. Note that these parameters are already attached to the raw data creating the Level-1a products.

The image is subsequently geometrically corrected, [RD. 53], through a coordinate transformation to an Earth Coordinate System, [RD. 55], such as WSG85.

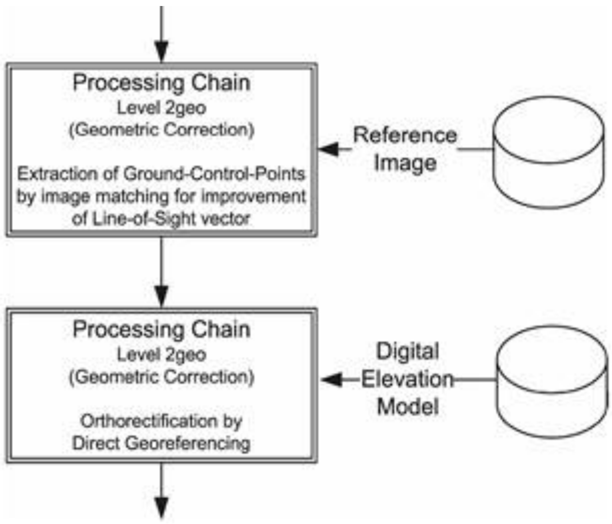


Figure 10-20. Geometric correction process chain for EnMap data. Source: [RD. 46]

10.5.3.4. User/additional algorithms block

The last block, *User/additional algorithms*, accounts for all the user-defined or mission-specific algorithms at level-1 that are not generally used in every instrument/mission. For example, some instruments perform preliminary pixel classification (browsing) to differentiate between Clouds, Ocean and Land.

The Cloud Fraction & Pressure algorithm used by the instruments GOME-2 and SCIAMACHY, [RD. 57] and [RD. 58], will be part of this block as it is considered to be a level-1 product.

Additional user-developed algorithms can be implemented in this block. Alternative geolocation algorithms are an example of algorithms that could be executed in this building block.

10.5.4. SUMMARY: INTERACTIONAL DIAGRAM

The interactional layer of the *Level-1 Processing* module is given in Figure 10-21, which shows the operations and algorithms in the time-flow diagram. Each block will have corresponding quality flags to track the goodness of the data processing. Furthermore, it is proposed that each block has the versatility to include user-defined processing algorithms developed along the different mission design Phases. It is important to define the block inputs and outputs necessary to keep the same simulator architecture when applying new processing algorithms.

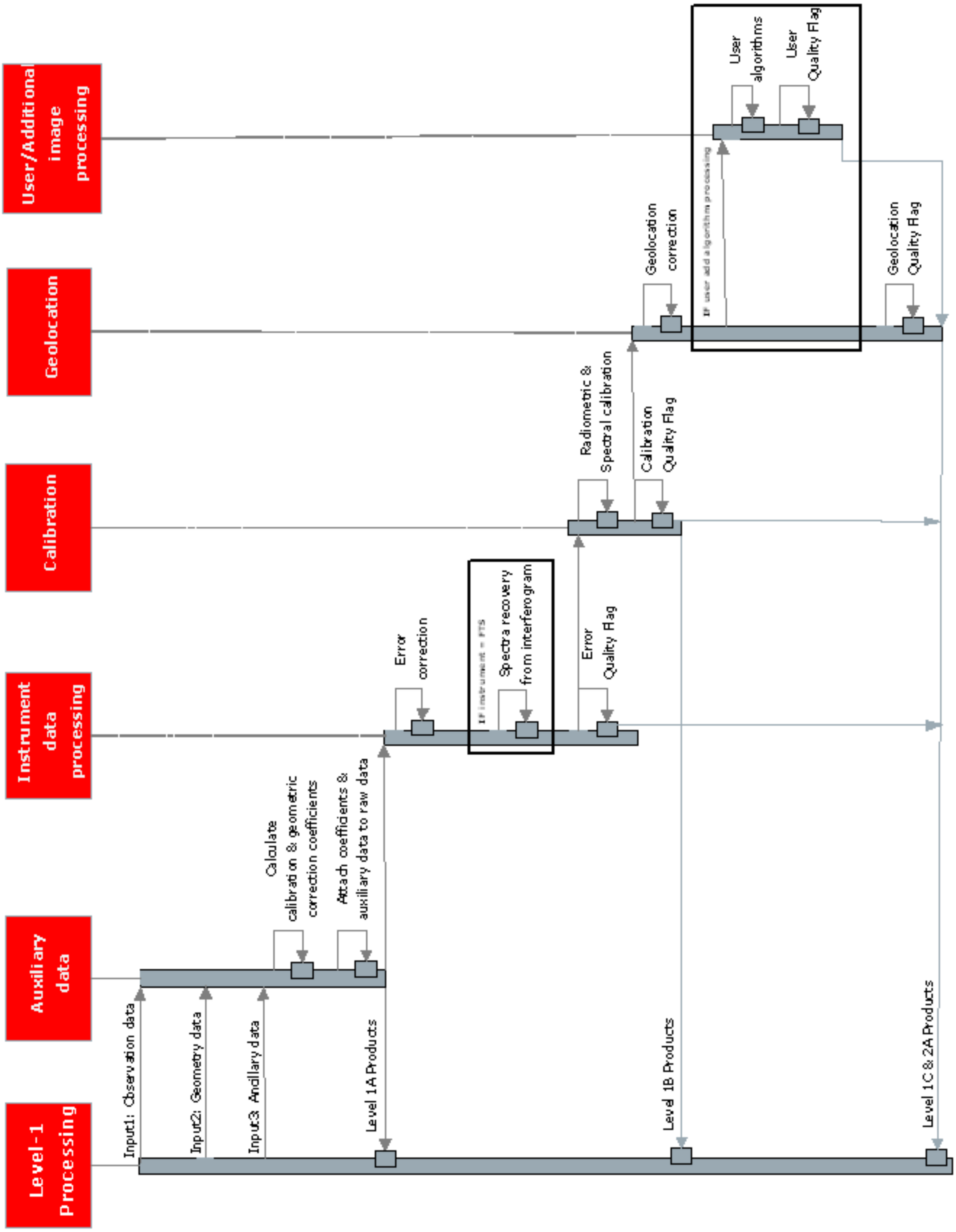


Figure 10-21. Interaction diagram for the Level-1 Processing module.

10.6. LEVEL-2 RETRIEVAL MODULE

10.6.1. INTRODUCTION

The *Level-2 Retrieval* module is in charge of obtaining the bio- and geo-physical parameter. These parameters could change for different missions as their spectral bands and resolution is different from each case and new developed algorithms could be applied to obtain high level products.

The study undertaken in TN1 showed that, while level-1 data products were almost the same for every passive optical mission, level-2 retrieval algorithms depends greatly on the specificities of each mission. Notwithstanding, level-2 products are related with the bio- and geo-physical parameters and thus it is possible differentiate between two main targets, [RD. 91]:

- **Atmospheric** measures generally provide with trace gases information, ozone, temperature & humidity vertical profiles, cloud properties (e.g., cloud coverage, top temperature or pressure), aerosol properties (e.g., absorption coefficients or optical thickness) including also volcanic ash properties or smoke plumes and lightning detection.
- **Surface** imagery generally produces information about vegetation, land/ocean surface temperature, and retrieval of bio- and geo-physical parameters (e.g., chlorophyll, Leaf area Index, snow & ice coverage and thickness, fire spots, fluorescence). Images from the Earth surface are provided with atmospheric correction to retrieve Top Of Canopy (TOC) (or surface) radiances instead of TOA radiances produced at the end of level-1.

The proposal building blocks for the *Level-2 Retrieval* module, their description and behavior and interaction diagrams will be shown and explained in the following section.

10.6.2. LEVEL-2 RETRIEVAL MODULE: DESCRIPTION OF BUILDING BLOCKS

The *Level-2 Retrieval* module structure is given by the following diagram:

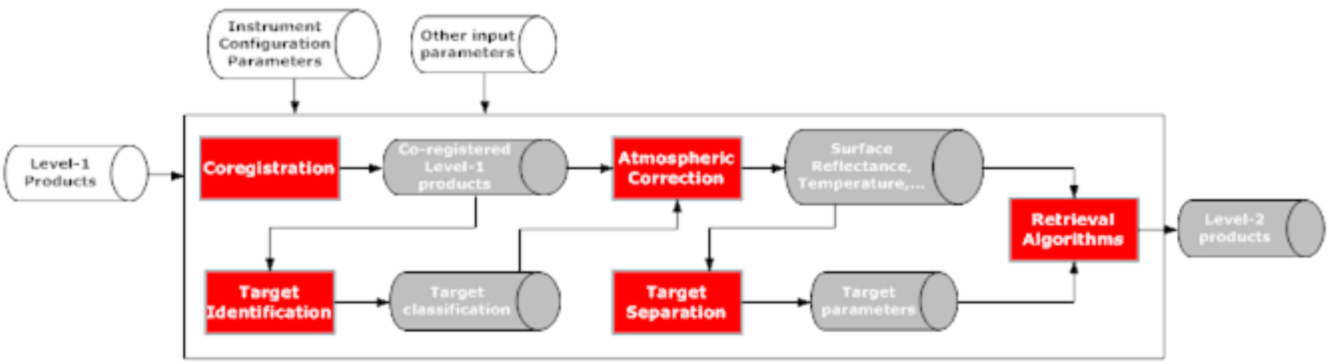


Figure 10-22. Structure of the Level-2 Retrieval module and its building blocks.

Level-1b products from the *Level-1 Processing* module will be taken as input together with configuration parameters defining the atmospheric profile, instrument spectral bands and additional parameters needed for each building block.

A preliminary coregistration of data from different instruments might be needed for further processing at Level-2. Prior to perform the atmospheric correction of the data, target identification should be done for archiving and implementation of a preliminary cloud mask. The Atmospheric Correction building block will determine the atmospheric parameters needed to retrieve surface properties (e.g. reflectance) and the final Retrieval Algorithms building block will provide with Level-2 products according to the target being observed.

For surface pointing instruments, this module should provide as output a classification of the land-use, retrieval of the biophysical parameters and the TOC radiance.

Two main mission targets, being atmosphere and Earth surface, are considered and they will define which building blocks will be used. While Earth surface pointing instruments may be able to apply cloud detection algorithms and atmospheric correction for a better retrieval of the biophysical parameters, the atmospheric measures will be used directly to retrieve the atmospheric parameters.

10.6.2.1. Coregistration block

This block is in charge of merging Level-1 products coming from different instruments/satellites so they can be combined for the retrieval of common Level-2 products. The Level-1 products should be coregistered and resampled to the same spatial resolution so the images can be compared and processed as a single product.

10.6.2.2. Target Identification block

The *Target Identification* block is in charge of making a preclassification of each image pixel. The classification is mainly done for archiving of the image and it generally applies simple classification (land/ocean/sunglint/cloud). Apart from being used as archiving, a preliminary cloud mask will determine those image pixels affected by clouds, process known as cloud screening or masking. Cloud detection algorithms are classified in two main types: 1) binary classification, providing with flags to indicate the presence or absence of clouds and 2) abundance methods, indicating the probability of cloud cover on a pixel.

Binary classification methods rely on the specific properties associated with clouds (e.g., lower temperature, high reflectance, white color) so a flag is added when their values are over a threshold.

Abundance methods study a pixel and its surroundings determining the probability of clouds. Gómez-Chova et al., [RD. 77] [RD. 78] [RD. 79] recently proposed a new cloud detection algorithm adapted for multispectral and hyperspectral images. The method makes use of the information obtained from the instrument measures at TOA (e.g., oxygen and water absorption band, brightness, whiteness) and, if available, from a DEM of the observed surface. The algorithm consists in a preliminary features extraction and clustering, spectral unmixing and final cloud cluster's labeling. The use of this type of algorithms requires the generation of a realistic cloud cover on the *Environment Simulator* building block (see section 10.3.2.2.2).

Depending on how the cloud cover was generated, some of the cloud detection algorithms will be available.

10.6.2.3. Atmospheric Correction block

The *Atmospheric correction* building block transforms the input TOA radiance (Level-1 data) to surface radiance considering a DEM and Aerosol Optical Thickness (AOT) as input and corrects, if desired, the shadowing effects occasioned by the cloud cover. Several methods are explained in the bibliography ranging from precise fine atmospheric corrections to gross and approximate corrections:

- **Atmospheric Radiative Transfer Models** (RTM)

  Atmospheric RTMs like MODTRAN or 6S are used to simulate the effects of the atmosphere and correct from its effect. This method is divided into two stages:

  1. To retrieve the atmospheric parameters in order to quantify the effect of the atmosphere on the measured radiation.

     These parameters can be retrieved from the instrument data itself, e.g., by minimizing a merit function between the measured and simulated TOA radiance or reflectance, [RD. 69]. Alternatively, these parameters can also be obtained from radiosonde measures or using standard atmospheric models, [RD. 68], or by recalibration of the image taking ground measurements of the reflectance, [RD. 70].
  2. Inversion of the atmospheric RTM, with the selected parameters obtained in the first step, derives TOC radiances.

  An example of the implementation of this approach is the FLAASH tool, [RD. 76], available in ENVI and incorporating MODTRAN4 or on L. Guanter method, [RD. 93] [RD. 94] applied for correction of ENVISAT/MERIS data.

- **Image-based empirical methods**

  Image-based empirical methods, generally providing with poorer results due to assumptions about the uniform atmospheric transmission or simplifications on the scattering and adjacency effects, are also used for atmospheric correction of passive optical measures. Some methods commonly used are briefly explained in the following paragraphs:

  - **Dark Object Subtraction (DOS) Method**

This classical method of atmospheric correction is based on the rough approximation that considers the contribution to radiance measured by the darkest pixel of an image comes from the atmosphere irradiation. This assumption is not completely true as neighbour pixels also contribute to the detected radiance. Furthermore, one cannot consider only the darkest pixel in the visible as in the infrared range there is also radiation from the surface.

Improved DOS methods have been developed, [RD. 72] [RD. 73], including multiplicative correction from the effect of the atmospheric transmittance and adding atmospheric scattering, [RD. 70]. Other improved methods make use of Look Up Tables (LUT) (visibility, water vapour and land surface reflectance) previously generated and obtain the surface reflectance from an iterative process based on the DOS method, [RD. 74].

- **Empirical Line Method (ELM)**

  ELM method assumes a linear relation between the TOA radiance and surface reflectance, [RD. 70][RD. 75]. The constants of this linear relation are obtained by measuring two different calibration targets with high spectral contrast.

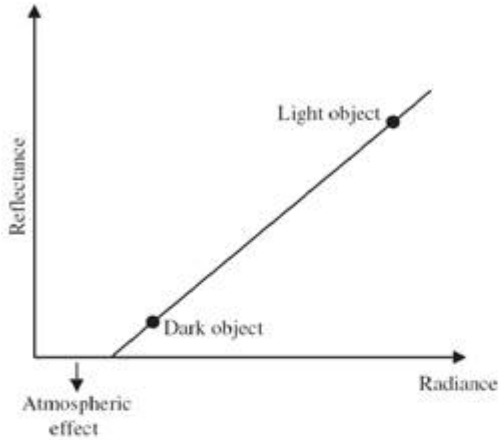


Figure 10-23. ELM atmospheric correction based on calibration by two (dark and light) targets.

Empirical methods are not used nowadays due to their lower accuracy. Nevertheless, Empirical methods have lower computational times and do not require additional knowledge of the scene so they can be used to make preliminary atmospheric correction. The RTM-based atmospheric correction methods are more effective but their computational times will be higher and will require additional configuration parameters.

It is proposed to provide a list of atmospheric correction algorithms (based on RTM or Empirical methods) adapted to the specific instrument bands. In addition, the possibility should be given that the user applies its own developed atmospheric correction algorithms, taken the required inputs (TOA radiance image) and outputs (TOC reflectance, emissivity image) into account. Furthermore, the user is given the possibility to skip this step and not perform any atmospheric correction so the TOA reflectance image is considered to be the TOC reflectance image.

10.6.2.4. Target Separation block

Using surface reflectance/temperatures as input, the Target Separation block will perform a more precise pixel classification so the retrieval algorithms will be selected according to the surface type being imaged.

10.6.2.5. Retrieval Algorithms block

The *Retrieval algorithms* block should include those algorithms obtaining the bio- and geo-physical characteristics (e.g., chlorophyll, surface temperature, ozone profile, fluorescence) from the instrument data..

The complexity of the retrieval algorithms depends primarily on the scientific mission. A library of standard retrieval functions can be implemented (e.g., vegetation indices) but the users should also be given the possibility of adding their own retrieval algorithms.

Retrieval algorithms can be based on methods of different nature. The following items give an idea of some types:

- Relatively simple mathematical relations between two or more spectral bands provide with an estimation of the biophysical parameter of interest.
- The generation of a LUT based on different RTM and the later minimization of a cost function for the difference between the Level-1C product and the values of the LUT is used to derive the biophysical parameters.

ESA has developed a toolbox (BEAM toolbox, [RD. 85] [RD. 86]) for optical satellite data processing between level-1 and level-2. This toolbox consists of several functionalities for image processing and retrieval algorithms applied for different ESA mission including atmospheric correction, retrieval of biophysical parameters and orthorectification (applied in *Level-1 Processing* module).

It is important to note that not all missions include standard data processing to Level-2. Therefore the simulation will run this processing block only if it is defined in the mission concept.

10.7. PERFORMANCE EVALUATION MODULE

10.7.1. INTRODUCTION & PROPOSED MODULE

Every mission simulation results should be evaluated, in terms of its performance and quality, to ease the mission design being able to track the possible sources of errors that affect the final mission objectives, i.e. the retrieval of biophysical parameters and image quality.

The *Performance Evaluation* module aims to answer the following questions about the simulated partial and final (level-2) results:

- Are the retrieved bio- and geo- physical parameters representative of the ground/atmospheric "truth"?
- Which are the bias and accuracies of the retrieved products with respect to the "truth"?
- How well the imager processing and retrieval algorithms resolve spatial structures?
- To what extent error sources, corrections, calibrations and algorithms have an impact on the retrieved products?

Product quality assessment can be done by four main steps as proposed in [RD. 61] and defined by CEOS [RD. 62]:

- **Validation** of the data products, i.e. analyze, by comparison with other data sets (e.g., field measurements, simulated scenes…), if the product accuracy is suitable for later use or analysis.
- **Verification**: Evaluates if the data/product fulfils specific requirements. For example, the verification process checks if the measurements are out-of-bounds, the statistical behaviour or the internal consistency.
- **Calibration**: Obtain the radiometric values of the instrument measures taking into account the temporal degradation of the instrument.
- **Monitoring**: Tracking of specific quality parameters to detect fails in the instrument, data processing algorithms or auxiliary data problems.

The calibration step is implemented internally at *Level-1 Processing* module; therefore the *Performance Evaluation* module will:

- **Validate** the data products through comparison of the retrieved products (i.e. level-1b TOA radiance, level-2 TOC radiance and level-2 retrieved biophysical parameters) with the "truth" data. This will determine the errors produced at the different steps of the product processing chain without tracking the source of the errors.

  The image quality validation process is well summarized in Figure 10-24, [RD. 63], where each pair of equivalent derived-truth values is compared with each other.

- **Assess** the product **quality** (verification and monitoring) respect to an expected performance. This will be done by studying the different simulation results without comparison to external data. The product quality assessment will generate the corresponding quality flags.

The validation process can be executed after the complete simulation, when all the products are delivered or right after the generation of the different products. The latter is considered as more flexible as the user can choose to stop the simulation when an important failure is detected. For example, a big difference when comparing the "real" and obtained level-1 TOA radiance can be indicative of an error on the image error correction algorithms, calibration algorithms or geolocation. The user can then stop the simulation before executing the next simulation step and modify corresponding algorithms.

In addition, quality assessment of data products and processes is performed during the simulation process. This evaluates the mission simulation performance and generates the corresponding quality flags that are attached to the different data products.

The scheme proposed for the *Performance Evaluation* module (see Figure 10-25 and Figure 10-26) is similar to the defined for MODIS/Terra Land Quality Assessment (QA), [RD. 64].

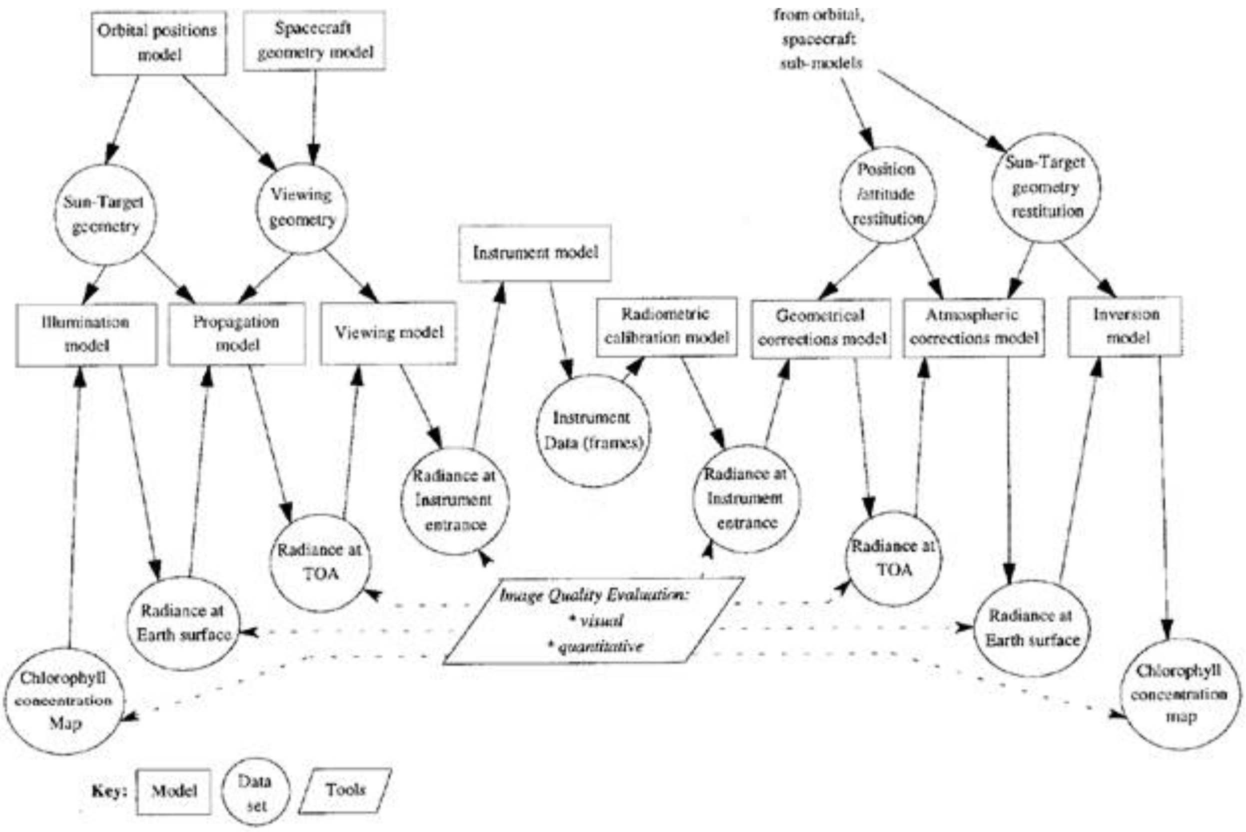


Figure 10-24. End-to-End mission simulator block diagram for Envisat's MERIS instrument.

Quality assessment evaluation is performed at the following entities of the simulation process:

- **Instrument**: Instrumental errors, band registration, overlap between different cameras of the same instrument, incomplete payload data transmission…
- **Platform**: inaccuracy of ancillary data sets, incomplete ephemeris data transmission, uncertainty of platform/instrument attitude…
- **Data processing**: incomplete instrument characterization and calibration knowledge, geolocation uncertainties…

10.7.2. PERFORMANCE EVALUATION: MODULE DESCRIPTION

The *Performance Evaluation* module will be run at different points of the simulation process and will compare images/data generated at different times so the interface should allow storing all generated data. This storing routine can be implemented and executed in each module/block/sub-block which generates the needed results.

The proposed module in Figure 10-25 has one single building block with performance evaluation functions adapted to the input data.

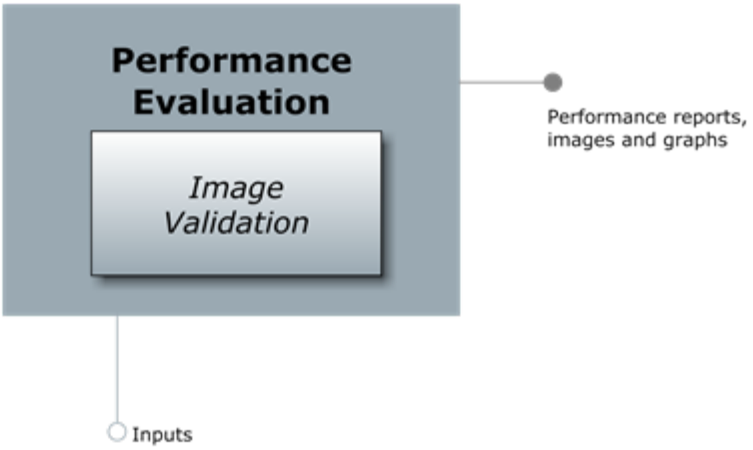


Figure 10-25. Performance evaluation module structure with its two building blocks.

The performance evaluation module will compare the following features:

- **Bio- and geo-physical parameters** defining the scene. These parameters, distributed on the scene, were generated in the *Class Map* sub-block belonging to the *Scene Generator* module. They will be compared with those obtained in the *Retrieval Algorithms* block in the *Level-2 Performance* module.
- **Reflectance, emissivity, temperature and fluorescence at the Earth surface** or Top Of Canopy. These values were obtained after applying the illumination calculations and RTM in the *Scene Generator* module. They will be used to analyze the performance of the simulator with the **level-2 TOC Reflectance emissivity, temperature and fluorescence** image obtained at *Level-2 Retrieval* module once the atmospheric correction is done with the corresponding building block.
- **Radiances at TOA**, also computed by the *Scene Generator* module after the propagation of the TOC radiance and emissivity though the atmosphere. These images will be compared with the **level-1b TOA radiance** image output from the *Level-1 Processing* module.
- Real **Observation geometry** given by the orbit and instrument pointing in the *Geometry* module will be compared with the observation geometry obtained and applied in the *Geolocation* block of the Level-1 Processing module.
- **Cloud layer** generated in the *Cloud Layer* building block of the *Scene Generator* module will be compared with the **cloud mask** retrieved in the *Cloud Detection* block of the *Level-2 Retrieval* module.
- The input **DEM** on the *Scene Generator* module will be compared with the corresponding surface altimetry derived from stereoscopic images from the *Level-2 Retrieval* module.
- The **atmospheric parameters** used in the configuration for the *Atmospheric Effects* block of the *Scene Generator* module will be compared with those retrieved by the *Atmospheric Correction* block or the *Retrieval Algorithms* block on *Level-2 Retrieval* module.

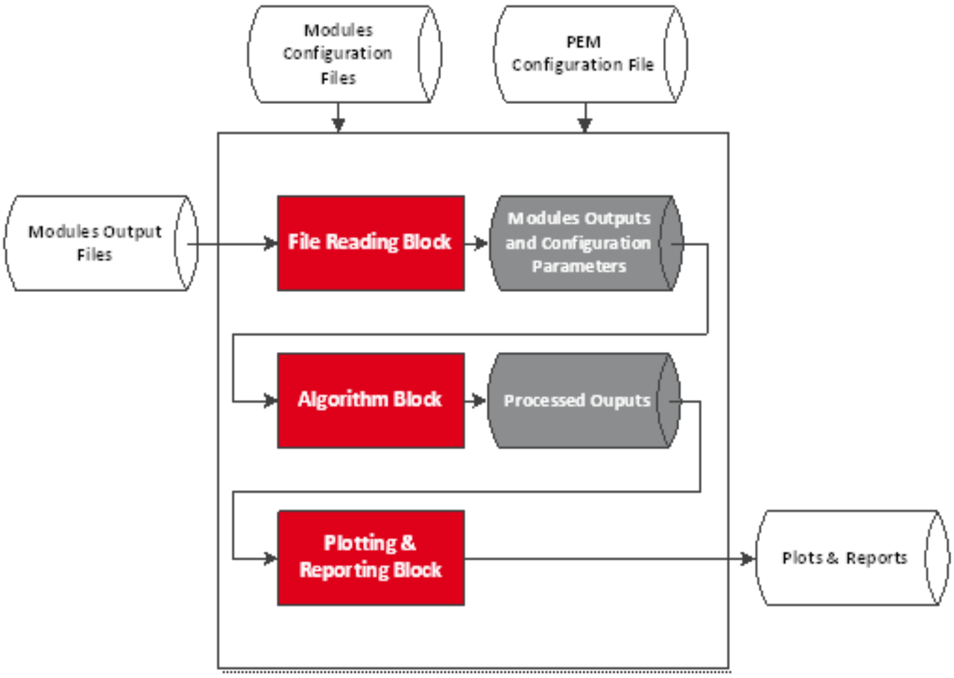


Figure 10-26. Schematic behavior of the *Performance Evaluation* module.

10.7.2.1. Image validation algorithms

The *Image Validation* block will be run each time a simulator product is generated, i.e. when the level-1 TOA radiance, level-2 TOC reflectance/emissivity/temperature/fluorescence and level-2 biophysical products are retrieved. These images/data are then compared visually and analytically with the corresponding image/data from the *Scene Generator* module.

Depending on the evaluated input data, the performance evaluation functions will be different. In the following points some commonly used methods are briefly described:

- **Spectral Angle Mapper (SAM) algorithm**: This method compares two "equivalent" pixels from the truth image and the retrieved image. Each pixel spectrum is represented as a vector so the SAM method calculates the angle between the truth image pixel spectral vector and the retrieved image pixel vector. If the obtained image pixel is the same as the truth image pixel, the angle should be zero. [RD. 80] [RD. 81].

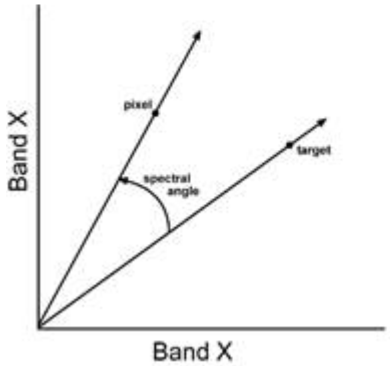


Figure 10-27. Scheme of the SAM algorithm. Source: [RD. 82].

- **Confusion Matrix**: This method, [RD. 83] [RD. 84], compares the truth of the biophysical parameters with those retrieved on the *Level-2 Retrieval* module so the algorithms here implemented can be evaluated in terms of their performance. Two matrix of size m×c (m=number of pixels; c=number of classes/parameters) is created for the target (truth) image, T, and retrieved image, R. The confusion matrix, C, is obtained by the product between the transpose of T and R. The analysis of the confusion matrix allows determining which classes have not been correctly classified as well as those image areas where the classification fails.

Percentages	Classification Data							
Reference Data	Water	Bare Ground	Deciduous Forest	Coniferous Forest	Urban	Row Total	Producer's Accuracy	Errors of Omission
Water	17.3	0.1	0.2	0.1	0.3	18.0	96.1%	3.9%
Bare Ground	0.1	19.7	0.4	0.4	0.8	21.3	92.1%	7.9%
Deciduous Forest	0.1	0.7	15.5	1.1	1.2	18.6	83.3	16.7
Coniferous Forest	0.6	0.2	1.2	13.8	1.1	16.9	81.7%	18.3%
Urban	0.8	1.2	1.4	2.0	19.8	25.2	78.7%	21.3%
Column Total	18.8	21.9	18.6	17.5	23.2	100.0		
User's Accuracy	91.8%	89.9%	83.1%	78.8%	85.6%			
Errors of Commission	8.3%	10.1%	16.9%	21.2%	14.4%			

Figure 10-28. Example of a Confusion matrix where the columns indicate the retrieved classes and each line correspond to the truth classes. Credit: [RD. 84].

- **Scatter plots**: This method compares each pair of parameters from the same pixel in the truth image and retrieved image. The dispersion of the values allows determining and quantify how similar the images and retrieved values are. Scatter plots typically goes along with statistical results (e.g., R^2, RMSE).

  For illustration purposes, a scatter plot is shown in Figure 10-30 comparing the two Figure 10-29 images. The visual analysis shows indeed that the images are not the same and this is confirmed by the dispersion of the values on the scatter plot.

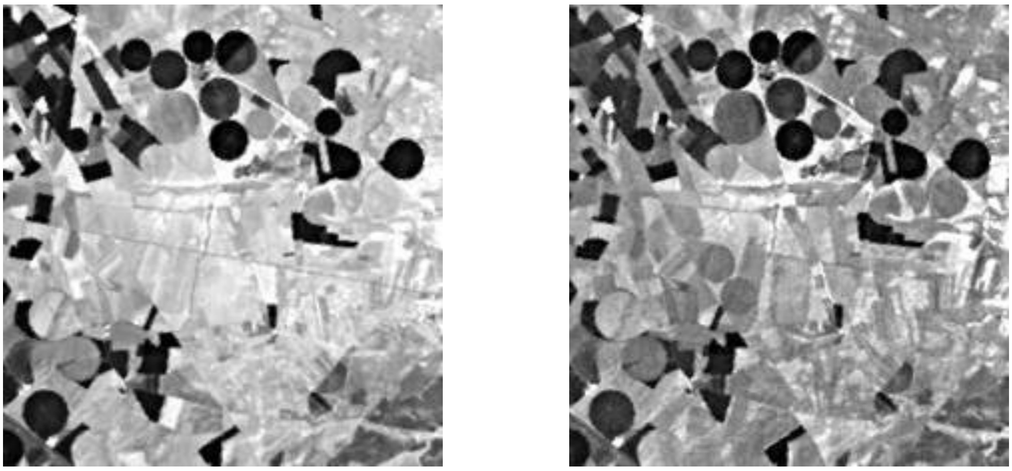


Figure 10-29. Truth distribution of a certain biophysical parameter (left) and retrieved parameters at level-2 (right) of the same area on a grey color scale.

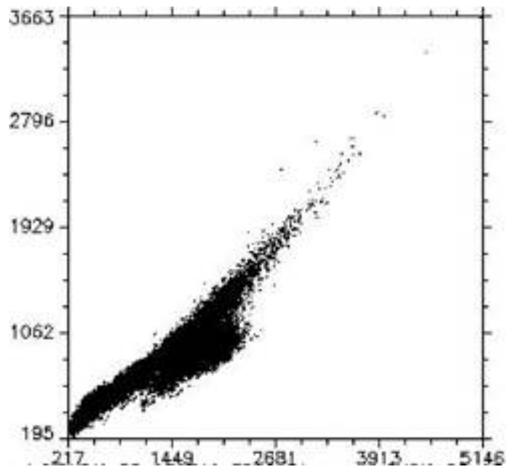


Figure 10-30. Scatter plot obtained from the images in Figure 10-29.

Several other methods can be applied to compare images and evaluate the quality of the retrieval. [RD. 87][RD. 88] list some of the methods (e.g., pixel difference based, correlation-based measures) which can be used in this block.

The user is given the possibility to implement their own functions that accept two images at the same level (TOC radiance, TOA radiance or biophysical parameters layer) and retrieve image quality evaluation parameters.

10.8. CONCLUSIONS

An E2E Mission Simulator architecture for Passive Optical instruments have been proposed in this section. The architecture was composed by six modules (*Geometry*, *Scene Generator*, *Instrument*, *Level-1 Processing*, *Level-2 Retrieval* and *Performance Evaluation*) that were described in the corresponding sections.

Each of these modules were sub-divided in building blocks performing operations such as image processing, data generation, retrieval algorithms or performance evaluation.

Examples of possible algorithms, software and functions that can be considered in the simulator were briefly described. In addition, limitations and constraints at each building block were foreseen.

We can conclude that the main issues for the construction of a generic mission simulator are the definition of inputs/outputs as well as the data flow. The proposed mission architecture in section 6 is considered flexible enough to be used for all passive optical instruments and probably some modules and building blocks can be reused or adapted for active optical and active/passive microwave instruments.

In addition, it is foreseen that the mission architecture can be kept at all mission Phases. The internal module description and its building blocks should be adapted for a mission simulator at more advanced mission Phases.

For future steps, it is recommended to check the consistency and homogeneity of data flow at simulator modules and building blocks for the different instruments (passive optical, active optical and microwave). Furthermore, definition of key parameters and inputs/outputs must be done for consolidation of the given architecture and building blocks.

11. REFERENCE ARCHITECTURE FOR PASSIVE MICROWAVE INSTRUMENTS

11.1. PROPOSED ARCHITECTURE

This section provides the methods to simulate the output of real aperture radiometers[1]. The End-to-End simulator covers the complete simulation of the environmental conditions seen by the microwave radiometer, the full instrument modeling and the determination of the retrieved relevant data measured by the instrument. The comparison with the "real" world conditions previously simulated will allow the evaluation of the instrument performances. If the comparison is performed at L1 (brightness temperatures) the radiometric performance of the instrument is assessed, but if the comparison is performed at L2 (retrieved geophysical parameters) the combined performance of the instrument and the geophysical parameter retrieval algorithms is assessed.

The simulator should cover the implementation of the:

- The **geometry** module including the platform's motion simulation (position and attitude).
- The **scene generator** module that simulates the "real" world sensed by the instrument system including the apparent brightness temperature (Top of the Atmosphere – TOA) reaching the sensor (Earth's emission including atmospheric, ionospheric and sky, cosmic and galactic radiations).
- The **instrument** module containing:
  - The physical model of the antenna,
  - The receiver models (including type of radiometer), and
  - The calibration models, aimed to implement a way to obtain measurements to perform the error correction.The generation of instrument's observables, that is the instrument outputs as a numerical value ("digital counts" or "digital number" or DN in short).
- The **L1-processing** module, including the algorithms to calibrate, rotate the polarization reference frame (of needed), and geo-reference the instrument's raw data and derive from it the brightness temperature maps (in Kelvin) as measured by the instrument which will allow the user to compare them with the real world generated brightness temperature.
- The **L2- retrieval** module, including the algorithms to retrieve the geophysical parameters from the brightness temperatures and then allow the user to compare with the real world geophysical parameters.

[1] Note: synthetic aperture radiometers are deliberately excluded because of their totally different approach, lack of commonalities with the other radiometer types, because so far there is only one synthetic aperture radiometer based mission (ESA's SMOS which a dedicated simulator was developed: SEPS), and finally because there is an on-going ESA-sponsored project to develop a generic E2E Performance Simulator for Synthetic Aperture Radiometers. However, if it is considered necessary to include Synthetic Aperture Radiometers as well, it is proposed to consider them as a special case of a push-broom radiometer with a "synthetic beam" (including negative side lobes) in the (ξ,η) directions as given by the reciprocal grid of the (u,v) grid in which the spatial frequency domain is sampled. In any case, it is understood that the specific calibration procedures are out of the scope of this simulator definition.

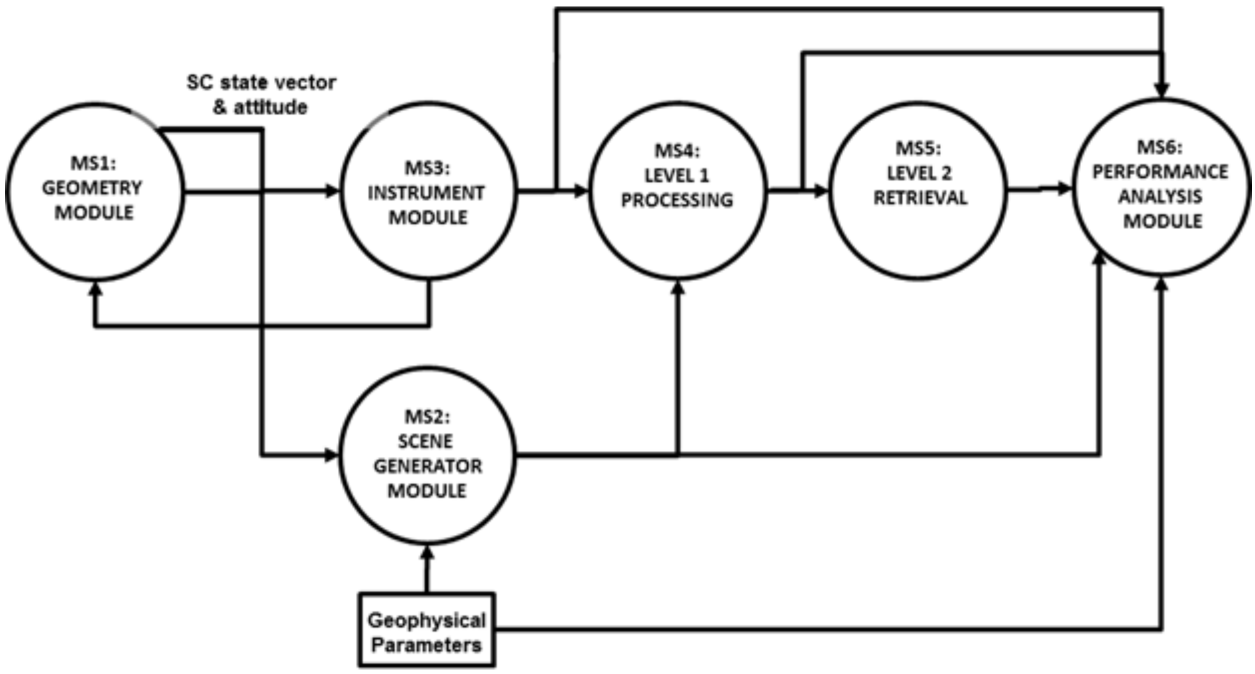


Figure 11-1. Structure and building blocks of the Mission Simulation Modules (MS) and the Performance Evaluation Module (PEM)

By looking at the above listed functions included in the simulator, it is clear the fulfillment with the End-to-End concept: the simulator user is able to define, re-define, or modify any parameter that could influence the mission results.

It is advanced that one of the critical building blocks will be the brightness temperature generator (**MS2** block) for arbitrary frequencies, incidence angle, and polarizations, because, despite the progress is modeling the atmospheric behavior, there is still a long way to go in the modeling of the emission behavior of land and sea surface (including the third and fourth Stokes parameters T3 and T4). This is probably due to the fact that many missions operate at a constant incidence angle ~53°, and most emission models have been developed around this value.

Before starting a mission simulation, it is necessary to define the mission. This includes the definition of input parameters (variables affecting the definition of the instrument), and the definition of control parameters (variables affecting the way the simulation is performed). The **Mission Definition** (**MD**) module must include the parameters related to the satellite, the instrument, the definition of the imaging mode of the instrument (in our case: nadir-looking, push-broom with N beams at $\theta_{i,1\text{-}N}$ incidence angle, conical scan at θ_i incidence angle, cross-track scan of $\pm\theta$ degrees off-nadir…), and finally the definition of the scenario (Figure 11-2).

The output results from the Level-2 Geophysical Parameter Retrieval module are finally evaluated in the Performance Evaluation module (**PEM**) and compared with the equivalent parameters from the generated scene (either TB at L1 or the geophysical parameters themselves at L2).

Table 11-1 is an ordered list of components ordered top-down that describes the hierarchical relationships between the different modules to be an input to the architecture definition.

Table 11-1. Identification of Components of a Passive Microwave E2E Performance Simulator

MD: MISSION DEFINITION
MD1. SATELLITE RELATED DEFINITION
MD2. INSTRUMENT DEFINITION
MD3. IMAGING DEFINITION
MD4. SIMULATION DEFINITION
MD5. SCENARIO DEFINITION

MS: MISSION SIMULATION
MS1: GEOMETRY MODULE

MS2: SCENE GENERATOR MODULE
MS2.1: CLASS MAP
MS2.2: BARE SOIL & VEGETATION COVERED SOIL
MS2.3: FROSTED & NON-FROSTED SNOW-COVERED SOIL
MS2.4: OCEAN
MS2.5: DIRECT SUN & SUN GLINT
MS2.6: DIRECT & REFLECTED SKY AND GALAXY NOISE
MS2.7: ATMOSPHERIC MODEL (up- & down-welling contributions, losses…) MS2.8: IONOSPHERIC MODEL (TEC Map Computation, up- & down-welling contributions, losses…) MS2.9: FARADAY ROTATIONMS2.10: ADD CONTRIBUTIONS BUILDING BLOCKMS2.11: SCENE INTERPOLATOR

MS3: INSTRUMENT MODULE MS3.1: ANTENNA BUILDING BLOCK MS3.1.1: ANTENNA POSITION (STATIC)MS3.1.2: ANTENNA POSITION (DYNAMIC)
MS3.1.3: ANTENNA TEMPERATURE AND OHMIC LOSSES
MS3.1.4: ANTENNA PATTERN COMPUTATIONMS3.2: RECEIVER BUILDING BLOCK
MS3.2.1: ARCHITECTURE DEFINITION
MS3.2.2: FREQUENCY RESPONSE COMPUTATIONMS3.3: OBSERVABLES GENERATOR
MS3.3.1: ANTENNA TEMPERATURE CALCULATOR FOR EXTERNAL TARGETS (SCENE OR VICARIOUS CALIBRATION)MS3.3.2: ANTENNA TEMPERATURE CALCULATOR FOR INTERNAL REFERENCESMS3.3.3: DATA PRE-PROCESSING BLOCK

MS4: LEVEL 1 PROCESSING
MS4.1: INSTRUMENT CALIBRATIONMS4.1.1: DETERMINATION OF INTERNAL CALIBRATION PARAMETERSMS4.1.2: DETERMINATION OF EXTERNAL CALIBRATION PARAMETERSMS4.1.3: ERROR COMPENSATION BUILDING BLOCK
MS4.2.TRANSFORMATION OF REFERENCE FRAME AND GEOLOCATIONMS4.2.1: POLARIZATION ROTATIONMS4.2.2: GEOLOCALIZATIONMS4.2.3: RESAMPLING TO STANDARD GRID

MS5: LEVEL 2 RETRIEVAL – GEOPHYSICAL PARAMETERS RETRIEVAL ALGORITHM
MS5.1: TARGET SEPARATIONMS5.2: LAND/OCEAN RETRIEVAL ALGORTIHMSMS5.3: ATMOSPHERE RETRIEVAL ALGORTIHMSMS5.4: ATMOSPHERIC CORRECTIONS BUILDING BLOCK

PEM: PERFORMANCE EVALUATION MODULE
PEM1: PLATFORM PEM PEM2: INSTRUMENT PEM PEM3: BRIGHTNESS TEMPERATURE MAPS PEMPEM4: GEOPHYSICAL PARAMETERS PEMPEM5: TABULAR AND GRAPHICAL REPRESENTATION

Before entering in the detail of the Mission Simulation modules, first Figure 11-2 to Figure 11-4 present the Mission, Satellite and Instrument Definition modules and their interaction, according to the above table. The Satellite Definition Module (**MD1**, Figure 11-3) includes the definition of orbit, platform, and instrument's attitude definition (where are the beams pointing to at a given instant of time?).
The Instrument's Definition Module (**MD2**, Figure 11-4) include the definition of the antenna(s), receiver(s), analog –to- digital converter (ADC) used to discretize the analog samples of the detected voltages, and the calibration.

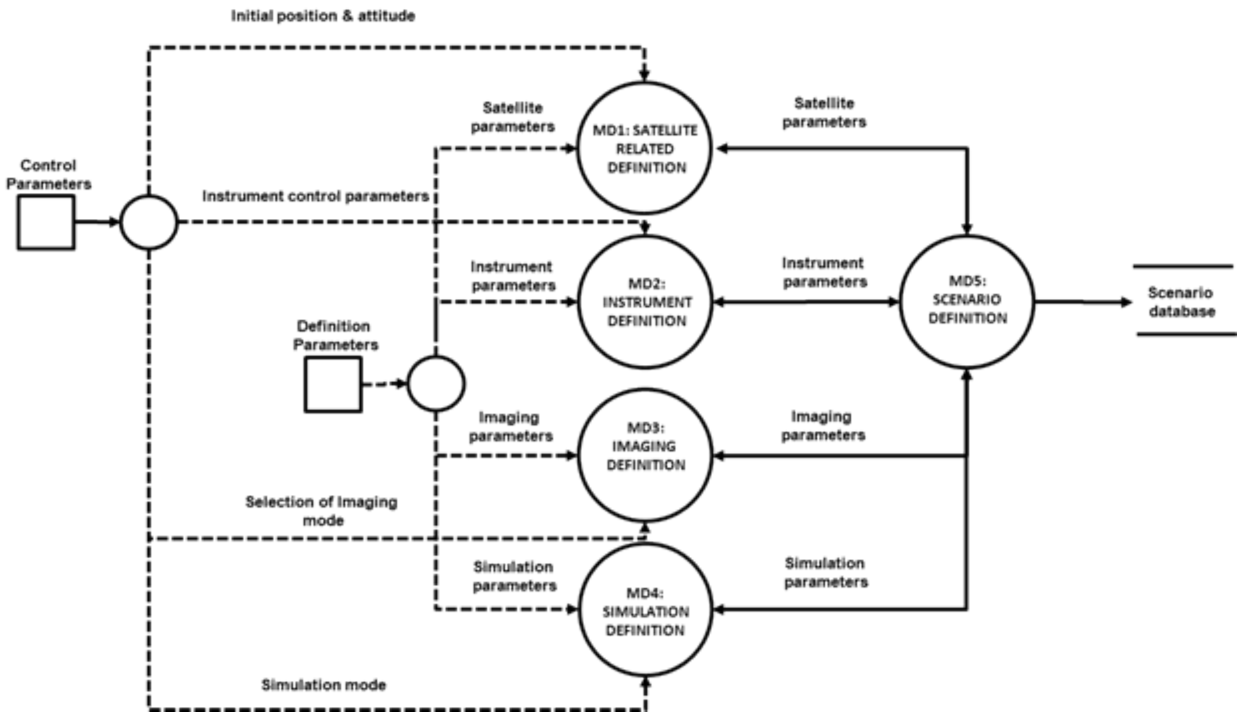


Figure 11-2. Structure and building blocks of the Mission Definition Module (MD)

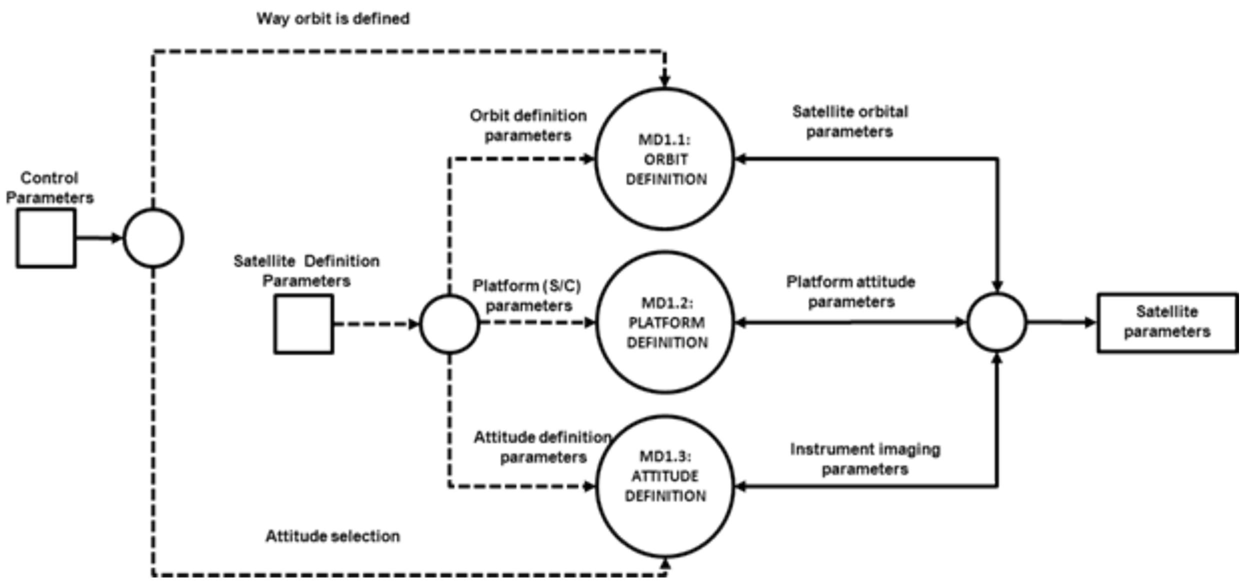


Figure 11-3. Structure and building blocks of the of the Satellite Definition Module (MD1)

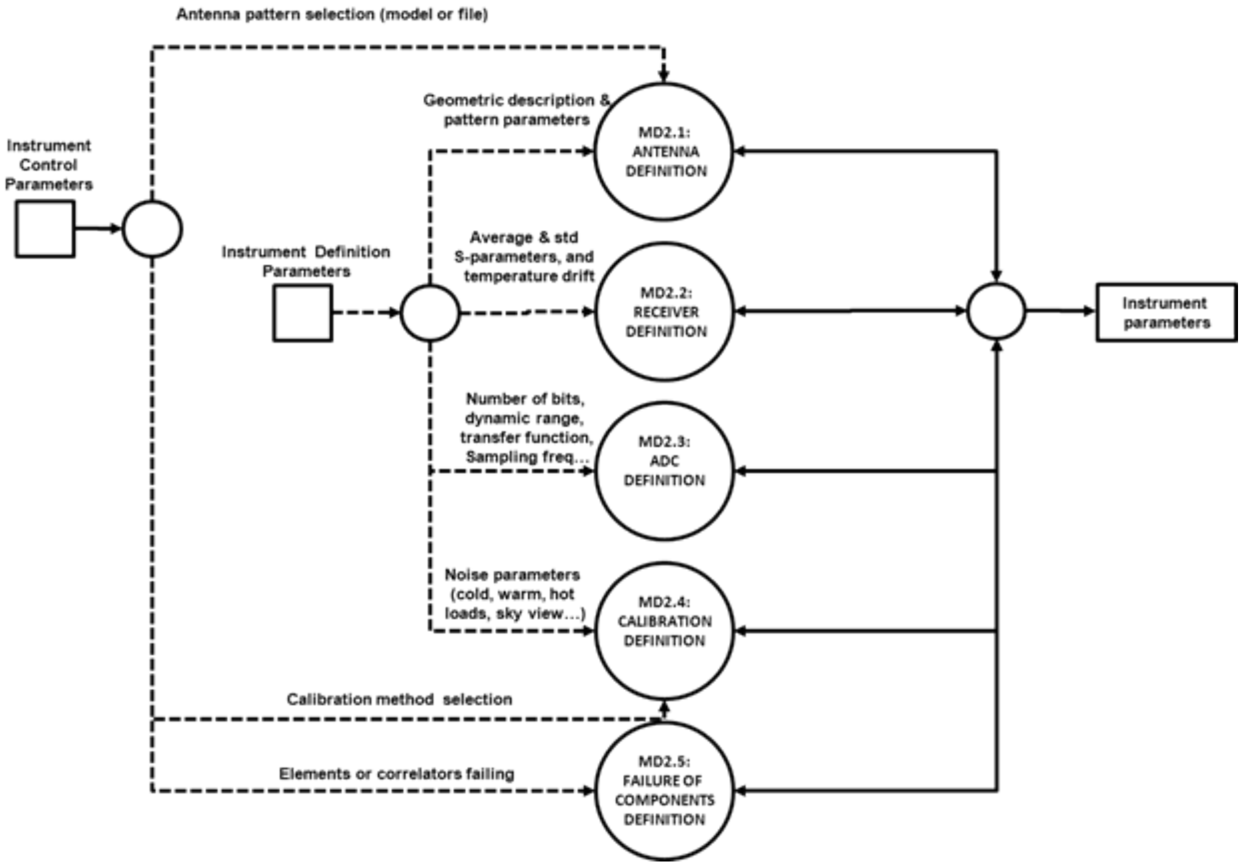


Figure 11-4. Structure and building blocks of the Instrument Definition Module (MD2)

Additionally, the control parameters determine the type of simulation to be performed. Three different types of simulations are proposed:

- **Snapshot mode:** to assess the instrument's response to a single input brightness temperature scene,
- **Monte-Carlo mode**: to assess the instrument response to a series of snap-shots in which the input brightness temperature remains constant, but the instrument parameters are randomly varied either assuming a Gaussian distribution (characterized by its mean and standard deviation), or an arbitrary probability density function (pdf). In this mode, the user shall have the possibility to select which instrument parameters are fixed and which ones are allowed to vary according to its configured law. This way, error budget analysis can be tailored to specific parameters or groups of them, as well as covering the whole instrument if needed. And,
- **Time-evolution mode**: to predict the instrument response to a series of selectable snapshots (typically consecutive snapshots), including the time-dependent variable input brightness temperatures and/or instrument aging and thermal drifts (antennas and receivers physical temperatures, gain and noise, offsets drifts) derived either from a mathematical function or read from an input data file. This mode will be the normal one when trying to simulate the mission.

In the following sections the Mission Simulation (MS) and the Performance Analysis Modules (PAM) will be explained in more detail.

11.2. GEOMETRY MODULE

The Geometry Module (Figure 11-5) provides the instrument observation geometry and depends on the satellite orbital position, the platform attitude. According to this, this module should have the following building blocks: **MS1.1**: Orbit computation (satellite's position), **MS1.2**: Platform computation (platform's attitude), and **MS1.3**: Geometry parameters computation.

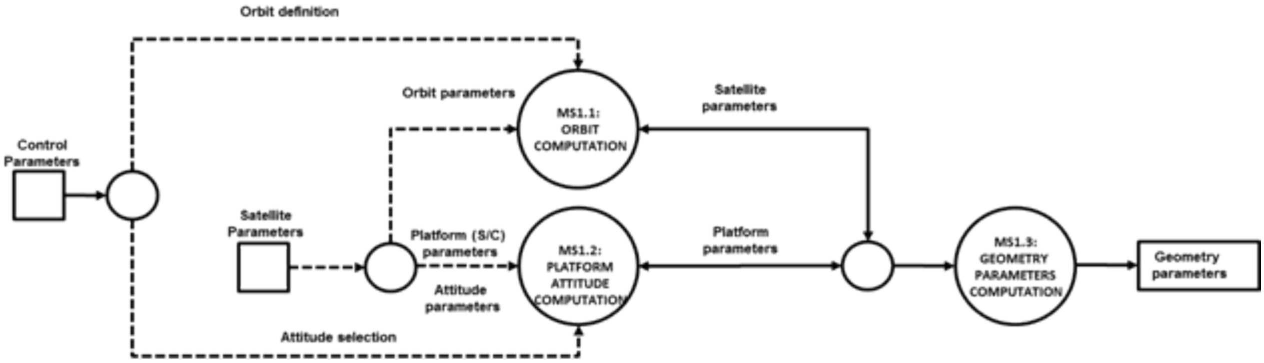


Figure 11-5. Structure and building blocks of the Geometry Module (MS1)

This module should also provide the telemetry data (platform position and attitude and time stamp[2]) to be used for geo-localization purposes in the Level 1 processor.

This module should be common to all EO missions.

11.3. SCENE GENERATOR MODULE

11.3.1. INTRODUCTION

As for the other sensors types, the Scene Generator, in the case of passive microwaves the "Brightness Temperature Module" is one of the major constituents of a mission simulator. The generated scenes should be simulated according to the mission objectives and the instrument characteristics (number of frequency bands, bandwidths, polarizations[3], radiometric resolution and accuracy, spatial resolution…). Although the spatial resolution must be much higher than that instrument to be simulated. Except for synthetic scenes, or local imagery, this is often difficult to achieve, especially for simulators that include the whole Earth, as it is usually the case in passive microwave EO missions. In many cases, the new instrument to be studied has a better spatial, radiometric and/or spectral resolution than the ones of existing instruments, therefore there is no previous data at enough resolution to be used as an input. In order to satisfy the Nyquist sampling theorem, a minimum factor of 2 is required, as it has proven to be enough for the detailed SMOS simulations using the SEPS [RD. 95], since there was no previous L-band radiometric data, and it had to be generated from models using as inputs geophysical data at a very modest resolution.

After the work undertaken in TN1 [PD.1], four different types of scenes have to be considered:

- **Earth pointing scenes:** Following the SEPS example, Earth's surface targets can be classified as: land (bare soil, vegetated soil, snow covered, inland water bodies…), and ocean (roughed sea, iced sea) , and all them will be affected by atmospheric and ionospheric effects. Note: in general atmospheric effects increase with frequency, while ionospheric effects decrease with increasing frequency, and are only noticeable at low microwave frequencies. Depending on the frequency and the required accuracy they could be neglected or not.
- **Sky pointing scenes:** These scenes are used for calibration purposes, either by turning the satellite upside down and making the instrument point to the zenith direction, of by switching the Earth-looking antenna and the "sky-horn", as it is done in the microwave radiometers accompanying the radar altimeters. Sky scenes should take into account the cosmic background (2.725 K), the galactic radiation, and the Sun and the Moon (whenever applicable [RD. 96]).
- **Limb pointing scenes:** These scenes are generated by defining the atmospheric characteristics and can also include a reference background spectrum (e.g., cosmic background, galactic radiation, Sun, or Moon when applicable [RD.3]). For limb sounders, special care must be taken when modelling the atmospheric emission (see for example [RD. 97]).
- **Other scenes:** A way to calibrate small radiometer drifts and/or cross-correlate different instruments consists of comparing the brightness temperature histograms and forcing the

[2] Instrument attitude (e.g. antenna pointing) is computed later in MS3.1.1 and MS3.1.2.

[3] Either vertical (T_v) and/or horizontal (T_h) polarizations, and/or third (T_3) and/or fourth (T_4) Stokes elements.

minimum value to be the same for all. This method is called vicarious calibration [RD. 98]. It can be seen as a particular case of the Earth pointing scenes over regions that are known to provide the coolest places on Earth in term of T_B.

As -for example- in the SEPS, these four different scenarios can actually be computed by the same module, according to the instrument pointing previously defined (Section 2). A fifth category of scenes are user-generated defined that can be used to test the instrument performance, such as homogeneous brightness temperatures, or the geophysical parameters retrieval algorithms.

11.3.2. DESCRIPTION OF BUILDING BLOCKS

According to the above-described different possible scenes of a passive optical mission, the Brightness Temperature Module will consist of the building blocks shown in Figure 11-6. The use of some of these blocks will depend on the configuration parameters defining the mission concept, for example, in a limb sounder, the ocean, land, ice/snow parameters are not needed, and special care must be paid to the atmospheric modeling.

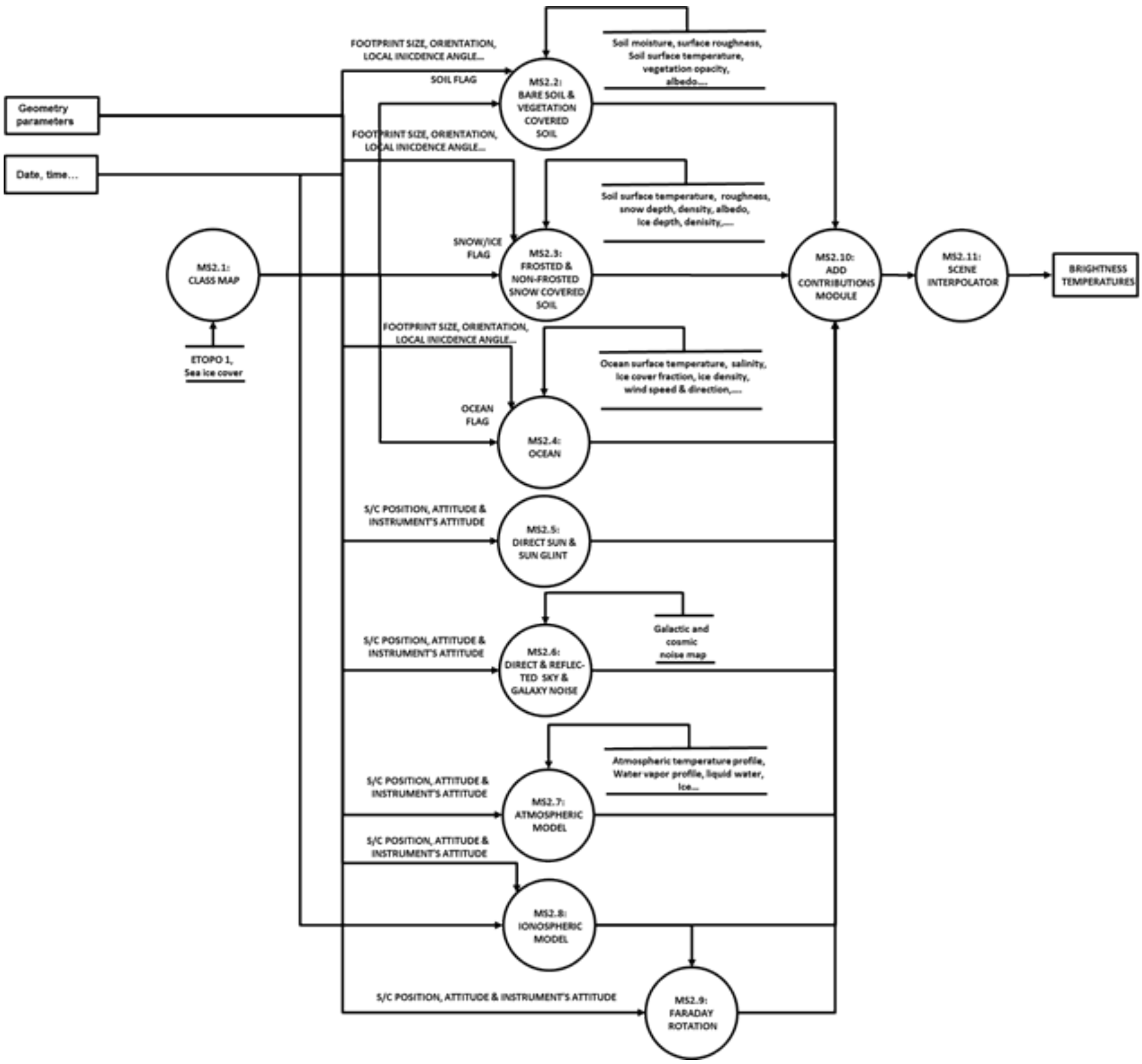


Figure 11-6. Structure and building blocks of the Scene Generator Module (MS2)

The Scene Generator Module starts with the Geometry Parameters that provide the footprint size, orientation, and local incidence angle required to compute the brightness temperature. The second step consists of separating the different surface types by differentiating land from non-frozen sea, and then differentiating bare soil and vegetation-covered soil, frosted and non-frosted snow-covered soil (**MS2.1**).

Land modeling (**MS2.2**, and **MS2.3**) requires a number of parameters, including the surface's temperature, moisture, roughness, topography, vegetation opacity, albedo… [RD. 99]. Since the antenna footprint size of space-borne microwave radiometers is -at least- a few kilometers, pixels are seldom uniform at this scale, and the fraction of the footprint covered by inland water bodies, urban areas, bare and vegetation-covered soils, snow etc. must be a priori known. Coast lines and topography can be obtained at 1 arcsecond resolution from Etopo 1[4]. This Digital Elevation Model (DEM) can also be used to compute the atmospheric contribution depending on the topographic height, and even to compute the effect of the variation of the local incidence angle.

The modeling of the ocean emission (**MS.2.4**) is simpler and it is usually parameterized in terms of the sea surface salinity, temperature, and the wind speed and direction. Thanks to the Coriolis/Windsat mission, polarimetric and anisotropic (including azimuthal variation) emission models have been refined and validated at 10.7, 18 and 37 GHz (see for example [RD. 100]).

In coastal regions, in a nominally ocean footprint[5], the impact of the land must be accounted for by including –at least- the fraction of the footprint covered by land, and if possible, by including the same descriptors as for a land pointing footprint. However, the impact of the sea in a nominally land footprint must also be accounted for by including the fraction of the footprint covered by sea, but fortunately, the variation of the TB of the sea is much more limited than that of the land (see for example [RD. 101]).

Atmospheric modeling (**MS2.8**) is a complex issue, especially at higher frequencies and in the presence of hydrometeors. At present ESA is running a project with the Satellite Atmospheric Science Group (SAT) of Luleå University of Technology, Kiruna, Sweden, to "design and build up a fast and easy-to-use propagation model available to ESA as an in-house tool supporting the definition of future missions for Earth observation" [RD. 97]. This on-going project will end in October 2012, and should be available for the integration in software packages like this one. In the case of a limb pointing scene only the **MS2.8** building block will be required, while in a Earth looking radiometer at frequencies below 10 GHz, the impact is minor, and simplified models can be used.

Sun contributions (direct and scattered, **MS2.5**) require a precise knowledge of the Sun brightness temperature at the frequency of observation, and at the time of observation, since it is very variable with both parameters. The Sun is a strong variable noise source with a noise temperature of about 10^6 K at 50 to 200 MHz and at least 10^4 K at 10 GHz under quiet Sun conditions. Large increases occur when the Sun is disturbed. The brightness temperature of the Moon is almost independent of the frequency above 1 GHz; it varies from about 140 K at new Moon up to 280 K at full Moon [RD. 96].

The galaxy (direct and scattered, **MS2.6**) decreases with frequency as $(f/0.408\ \mathrm{GHz})^{-2.75}$ and it is only significant below 2 GHz. Above 2 GHz, one only needs to consider the Sun and a few very strong non-thermal sources such as Cassiopeia A, Cygnus A and X and the Crab, and the cosmic background [RD. 96]. Since the only band reserved for passive microwave observations below 2 GHz is the 1400-1427 MHz one, the galactic map used in SMOS could be directly used.

Despite the progress in the numerical scattering models, they have failed in predicting the galaxy scattered contribution in SMOS, therefore this remains an open issue [RD. 102].

Ionospheric contributions (**MS.2.8**) mainly produce three effects:

1) a small attenuation of the radiation coming from the Earth, together with a small up- and down-welling contributions,
2) a rotation of the oscillation plane of the wave (Faraday rotation, building block **MS2.9**), and
3) scintillation or rapid variations of phase and amplitude of a signal, which passes through the Ionosphere. They are very pronounced in equatorial regions where they appear every day after sunset and may last a few hours. They are related in particular to the solar activity and the season. The main factors whose dependency has been established are indicated hereafter. At mid latitudes, the scintillations are rather weak, except during conditions of ionospheric storms. The scintillations result in a signal degradation from VHF up to C band [RD. 103].

Other effects such as curvature of the direction of propagation can be neglected in most cases taking into account the poor spatial resolution of microwave radiometers.

Modeling of the ionosphere contributions (1) and (2) above (**MS2.8**, [RD. 104]) requires precise and updated ionospheric total electron content (TEC) maps, and the Earth's geomagnetic field. This

[4] http://ngdc.noaa.gov/mgg/global/global.html

[5] Antenna boresight points to the sea, but the antenna footprint includes land as well.

information can be obtained from the International Reference Ionosphere IRI-2011 model [RD. 105] and the International Geomagnetic Reference Field IGRF-2011 model [RD. 106].

Finally, all these contributions have to be integrated (**MS.2.10**). As described above, the lack of general emission and scattering models over a wide range of frequencies, polarizations is a problem when creating a general model of the Earth's emission.

Very few validated studies have been reported. One of those studies is the one reported by Schiavon et al. [RD. 107] over the frequency range of 5-50 GHz, at vertical and horizontal polarizations only. On the lower frequency range, the CMEM (Community Microwave Emission Modelling platform) [RD. 108] has been calibrated and validated in the range 1-20 GHz. To authors knowledge, there is a gap in this area that will have to be covered in view of the development of this type of generic simulators.

Finally, BT maps are resampled to the instrument grid for easier calculation of the instrument's observables (**MS2.11**).

11.4. INSTRUMENT MODULE

11.4.1. INTRODUCTION

Microwave Radiometers are defined by:

- **Instrument type:** Total Power Radiometer (TPR), Dicke Radiometer (DR), Noise Injection Radiometer (NIR), or any other topology,
- **Geometric parameters**: number of antenna beams at each frequency band and pointing direction (static + dynamic, computed in **MS3.1**, according to the coupling between the platform's AOCS and the instruments), antenna half power beamwidth[6] (related to the footprint size), imaging model, and Field Of View or FOV (related to the swath width). Note: the footprint size, orientation, and FOV are computed in **MS1**, using the information on antenna pointing direction computed in this module.
- **Radiometric parameters**: number of frequency bands, center frequencies and bandwidths (including –if possible- the frequency response), radiometric resolution[7] (random noise associated to the finite integration time, computed as Type-A uncertainty [RD. 109]), radiometric accuracy (systematic errors including linearity errors, computed as Type-B uncertainty [RD. 109]).

The Instrument module (**MS3**) considers the spectral and spatial sampling of the instrument itself, including all the radiometric errors, and transforms the measured TOA radiances into Digital Counts. It is convenient to separate the instrument module in two blocks: the antenna building block (**MS3.1**) and the receiver building block (**MS3.2**), as sketched in Figure 11-7 below. These blocks are explained in detail in the following Section.

[6] It is equivalent to the Instantaneous Field of View or IFOV in optical sensors.

[7] It depends on the antenna temperature, the receiver's noise temperature, the noise bandwidth, and the integration time.

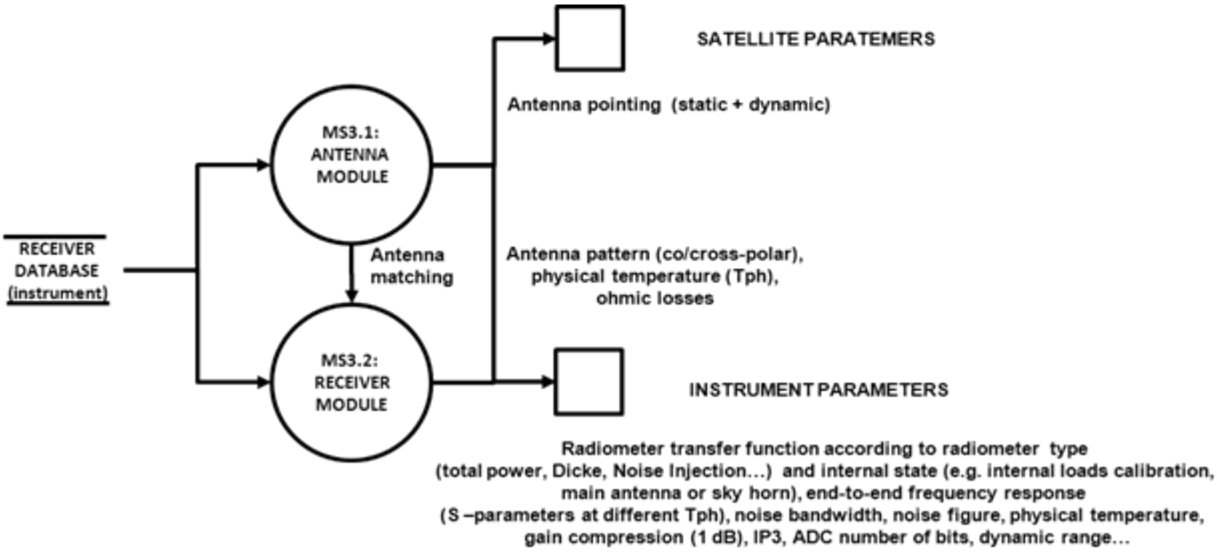


Figure 11-7. Structure and building blocks of the Instrument Module (MS3)

11.4.2. DESCRIPTION OF BUILDING BLOCKS

11.4.2.1. MS3.1: Antenna Building block

The Antenna modeling block (**MS3.1**) uses the satellite-related geometric parameters computed in **MS1.4**, and the antenna pattern model to compute the antenna pointing (static + dynamic). This information is stored in the satellite parameters data set, and reused in **MS1.4** to calculate the antenna footprint, shape, and orientation. Note: this information is required in **MS2** to compute later the antenna temperatures from the TB's.

The **MS3.1** building block is composed of four blocks:

- **MS3.1.1** that computes the static antenna position (antenna boresight with respect to the platform),
- **MS3.1.2** that computes the dynamic antenna position (changes in time of the antenna boresight direction with respect to the platform),
- **MS3.1.3** that computes the antenna ohmic losses and physical temperature. This information is required to evaluate the thermal noise introduced by the antenna losses. The thermal evolution of the antenna will depend on the antenna type, the orbit configuration, and the Sun orientation. It will have to be pre-computed by analysis using specific software, and the outputs can be input to the simulator as a look-up table (LUT) as a function of the latitude of the orbit at different times during the year.

And

- **MS3.1.4** that computes the antenna co- and cross-polar patterns (for each beam). This information is needed to evaluate the correction finite cross-polarization needed in the L1A processor.

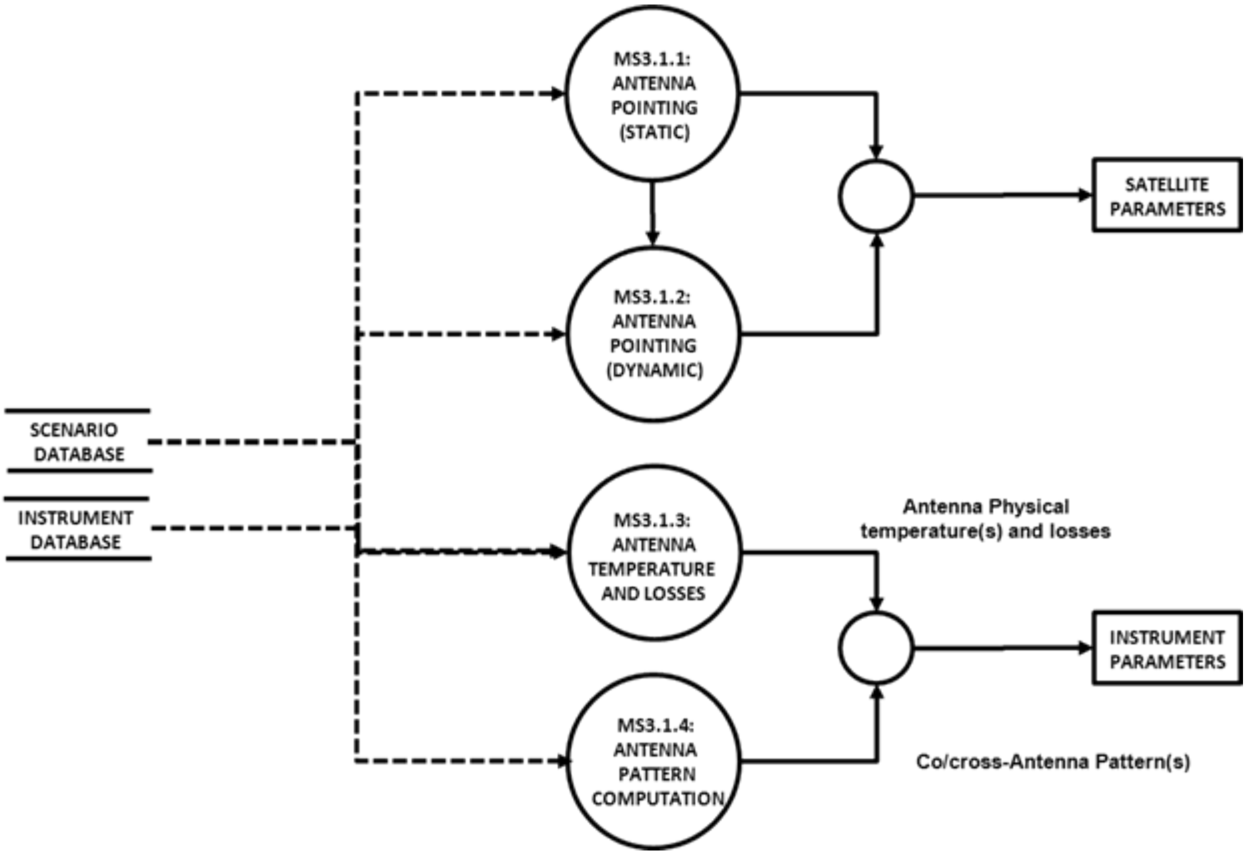


Figure 11-8. Structure and building blocks of the Antenna Building Block (MS3.1)

11.4.2.2. MS3.2: Receiver Building block

The receiver building block models the instrument electronics response, and its impact in the output, in absolute terms and in noise terms. The different building blocks are described below:

- **MS3.2.1** defines the instrument architecture (topology). A detailed hardware modeling of all the subsystems (e.g. amplifiers, filters, mixers, isolators…) is absolutely impractical for a generic simulator like this one. In addition, the number of radiometer types is very large [RD. 110], each on having a different radiometric resolution, and transfer function. For this project it is proposed to use an analytical formula derived from detailed S-parameters analysis as in [RD. 111] of the form:

$$V_0 = a(T_{ph}) + b(T_{ph}) \cdot T_A + c(T_{ph}) \cdot T_{ref} + d(T_{ph}) \cdot T_{sky} + [e_1 + e_2(T_{ph})] \cdot T_{ph} + f(T_{ph}) \cdot T_R + \cdots \quad (1)$$

  By proper selection of the coefficients, eqn. (1) can be adapted to any type of radiometer, including thermal drifts (either in the reference temperature and/or the S-parameters from which the a, b, c… coefficients are derived).

  For example:

  – in an ideal Total Power Radiometer (TPR): b = f = ct., and a, c, d, e1, e2…= 0.

  – in an ideal Dicke Radiometer (DR): $b = -e_1 = ct.$, and $a, c, d, e_2, f \ldots = 0$.

  – in a Noise Injection Radiometer (NIR), the output is zero, and the instrument's observable is the duty cycle (η) of the injected noise, which is proportional to T_A-T_{ph} divided by the $T_{N,\,\mathrm{ON}} - T_{N,\,\mathrm{OFF}}$, the noise injected when the noise source is ON and OFF. Therefore, $b = -e_1 = ct.$, and $a, c, d, e_2, f \ldots = 0$, as well.

  The T_{sky} term in eqn. (1) accounts for example for the finite isolation between the sky horn and the Earth looking channels, when looking to the Earth, or the other way around when the receiver is connected to the sky horn.

In a real radiometer, even small mismatches in the S-parameters transform into mismatches in the coefficients in eqn. (1), for example $b \approx -e_1$, but $b \neq -e_1$ in a Dicke radiometer. See [RD. 111] for more details on these modeling aspects.

- **MS3.2.2** computes the receiver's noise bandwidth, noise figure, gain fluctuations… which will be used later on to compute the radiometric resolution, as a function also of the radiometer's architecture (**MS3.2.1**)

The information on the physical temperature at different points of the receiver, the state of the switches that commute between different states (internal references, sky horn, Earth antenna…), and the actual antenna pointing (including satellite's orbit position and platform's attitude) will be added to the telemetry data to be downloaded for later post-processing.

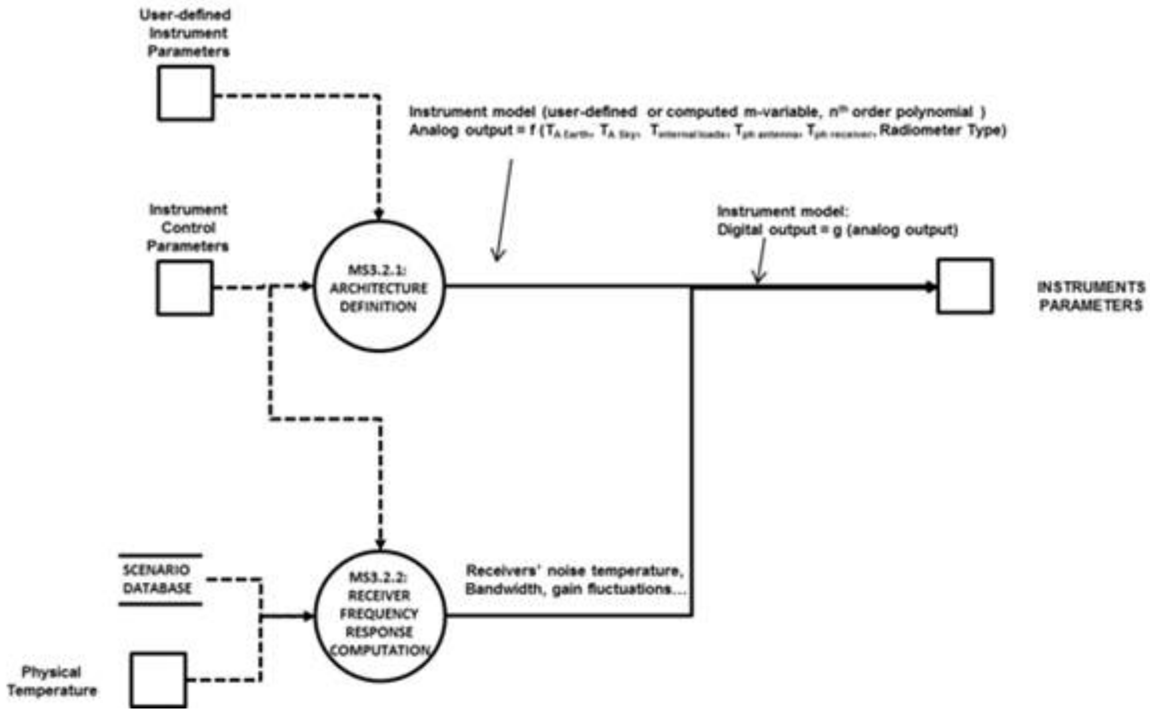


Figure 11-9. Structure and building blocks of the Receiver Block (MS3.2)

11.4.2.3. MS3.3 OBSERVABLES GENERATOR

11.4.2.3.1. Introduction

In the proposed architecture, the Observables Generator is the one that actually computes the raw instrument's response. To do that the outputs of **MS2** (Scene Generator Module) and **MS3.2** (Instrument Building block) are required. The brightness temperature map (from **MS2**) has to be convolved with the antenna pattern pointing in the direction corresponding to the particular time instant when the measurement is being acquired (**MS3.1**). This will provide the antenna temperature that would be measured by an ideal radiometer, without noise, and/or any other systematic errors (**MS3.3**).

In the following section, the block description is explained.

11.4.2.3.2. Description of Building Blocks

The Observables Generator block diagram is composed of three building blocks, as shown in Figure 11-10:

- **MS3.3.1** computes the instrument's output as measured by an ideal noise-free radiometer with the same co- and cross-polar antenna patterns (derived from [RD. 112] by making subscripts 1 equal to 2 and u=v=0):

$$V_{0,XX} \stackrel{\Delta}{=} k_B \cdot B \cdot \frac{1}{\Omega} \iint_{\xi^2+\eta^2\leq 1} \frac{|R_x|^2 T_{xx} + 2\Re e\{R_x C_x^*\} T_{yx} + |C_x|^2 T_{yy}}{\sqrt{1-\xi^2-\eta^2}} d\xi d\eta, \quad (2a)$$

$$V_{0,YY} \stackrel{\Delta}{=} k_B \cdot B \cdot \frac{1}{\Omega} \iint_{\xi^2+\eta^2\leq 1} \frac{|C_y|^2 T_{xx} + 2\Re e\{R_y C_y^*\} T_{yx} + |R_y|^2 T_{yy}}{\sqrt{1-\xi^2-\eta^2}} d\xi d\eta, \quad (2b)$$

$$V_{0,YX} \stackrel{\Delta}{=} k_B \cdot B \cdot \frac{1}{\Omega} \iint_{\xi^2+\eta^2\leq 1} \frac{C_y R_x^* T_{xx} + \left(R_y R_x^* + C_y C_x^*\right) T_{yx} + R_y C_x^* T_{yy}}{\sqrt{1-\xi^2-\eta^2}} d\xi d\eta. \quad (2c)$$

And $V_{0,XY}$ is obtained from (2c) exchanging subscript X and Y. In eqns. (2a)-(2c):

- the terms $V_{0,XX}$, $V_{0,YY}$, $2\Re e\{V_{0,YX}\}$, and $2\Im m\{V_{0,YX}\}$ are the instruments outputs for the "horizontal", "vertical", 3rd and 4th Stokes elements, respectively. Note: the use of X and Y to refer to "horizontal" and "vertical" polarizations comes from the fact that the definitions are in the antenna reference frame (X and Y), instead of the Earth's surface local reference frame (H and V), and there may be misalignments, or in the case of antennas with a wide beam, the polarization is not the same in all the directions within the beam.
- k_B, B and Ω are the Boltzman's constant, the receiver's noise bandwidth, and the antenna pattern solid angle, respectively,
- $R_{X,Y}$ and $C_{X,Y}$ are the co- and cross-polar antenna patterns at X and Y-polarizations, and
- $\xi = sin\theta cos\varphi$, and $\eta = sin\theta sin\varphi$ are the director cosines in the antenna reference frame.

For simplicity, it is proposed to perform the calculations in the director cosines domain for a number of reasons:

- Scanning configurations are easily implemented. For example, a conical scan with an off-nadir angle θ0 describes a circumference of radius $sin\theta_0$. A cross-track zig-zag scan perpendicular to the along-track direction describes a segment from $[-sin\theta_0, +sin\theta_0]$ in ξ, and η=0. A N-beam push-broom radiometer is defined by a set of directions (ξ0,i, η0,i), with i= 1…N.
- The shape of the antenna beam is invariant (unless it is physically rotated), regardless of the direction. For example, if a phased array is used, the antenna beam will be the same in all (ξ,η) directions.
- The inclusion of antenna side lobes is straightforward from eqns. (2a)-(2c).
- The inclusion of the sky, galactic, Sun, and Moon contributions is also straightforward, since they appear automatically in the TB map. Therefore, the same building block can be used for ANY radiometer type and imaging configuration.

- A flag will determine which type of scene is the antenna pointing at: a measurement scene, or an external calibration scene. In this second case, the brightness temperature map is a predefined region used for calibration purposes, for example, a zenith-pointing view of the sky, some particular regions of the Amazon rainforest, Dome C in Antarctica, or some regions of the ocean, known to exhibit the lowest possible TB at a given frequency (used for "vicarious" calibration [RD. 98]).
- **MS3.3.2.** is the Antenna Temperature Calculator (Internal References) building block. It calculates the equivalent noise temperature of the internal references used for calibration purposes (selected by means of a flag):
  - An Active Cold Load (ACL or Cold-FET): used as a cold load (below the physical temperature). They are characterized by the T_{COLD} parameter and the physical temperature.
  - A warm load: a matched load at ambient temperature used typically for calibration or offset cancellation, such as in a Dicke radiometer. Its noise temperature depends on the matching (usually extremely good) and its physical temperature, so that $T_{WARM} \approx T_{ph}$.
  - A hot load: usually an avalanche diode used as a noise source, providing extremely high equivalent noise temperatures (T_{HOT}). They are characterized by the Excess Noise Ratio ($ENR \triangleq 10log_{10}((T_{HOT} - T_{COLD})/T_0)$), and physical temperature.

The "antenna temperature" using internal references will be either T_{COLD}, T_{WARM} or T_{HOT}. Note that this building block is different from MS3.3.1. because it DOES NOT NEED to convolve a BT image with the antenna pattern, it ONLY needs the receiver's frequency response, and the noise temperature and matching of the internal loads.

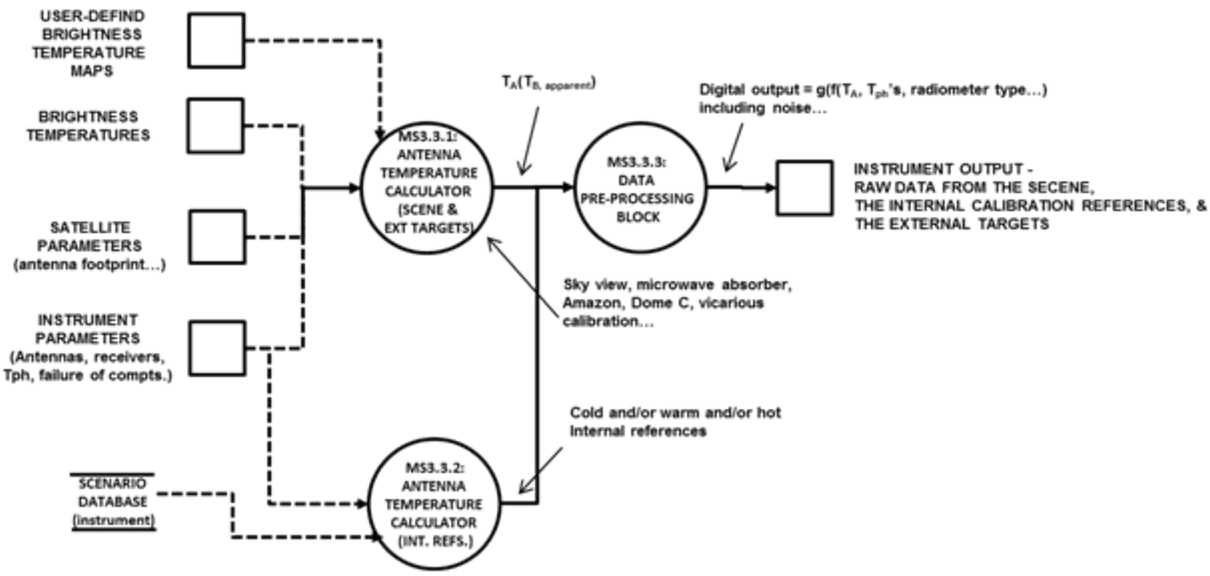


Figure 11-10. Structure and building blocks of the Instruments Observables Generator Block (MS4)

- **MS3.3.3.** computes the instrument's raw data in digital counts, including the radiometer type, all systematic errors and noise, as described in section 11.4.2.2.

11.5. LEVEL-1 PROCESSING MODULE

11.5.1. CALIBRATED INSTRUMENT OBSERVABLES

11.5.1.1. Introduction

Due to the nature of the magnitude being sensed (natural self-emission or scattered radiation from other bodies) the Level-1 data processing is very similar for all passive microwave sensors, and to optical sensors, as well. The raw data produced by the building block **MS3.3.3**: RAW INSTRUMENT OUTPUT is processed with the calibration data derived from on-ground measurements and the radiometric data from external targets and/or internal references (MS3.3.1 and MS3.3.2) to derive calibrated Top of the Atmosphere (TOA) radiometrically calibrated images, that means that ONLY instrumental effects are compensated for. In the following section, the building blocks are described.

The geolocalization of these data is performed in the block MS4.2.: GEOLOCATION that converts calibrated instrument observables into geolocated instrument observables.

11.5.1.2. Description of Building Blocks

The input of the **MS4**: LEVEL 1 PROCESSING is the raw data generated by the building block **MS3.3.3.**: Raw Instrument Output. The process to pass from output "digital counts" to physical quantities "Kelvin" is the instrument calibration. Actually instrument calibration consists of two steps, and internal instrumental error compensation, and an external absolute calibration using know external targets. It has been found that the processing steps are quite similar in all radiometer types, including atmospheric sounders (e.g. AMSU-A, with cross-track scanning), imagers (conical scanning), and limb sounders, except for synthetic aperture radiometers, which would deserve a special separate attention, and will only be briefly introduced. The calibration procedure also shares many commonalities with that of the passive optical sensors.

In principle, the calibration of a microwave radiometer is simple. Assuming a linear response (Figure 11-11), the radiometer's output to two known input antenna temperatures T_{cold} and T_{hot}, allows to compute the slope (sensitivity [counts/K]) and the offset (V_0 for T = 0 K).

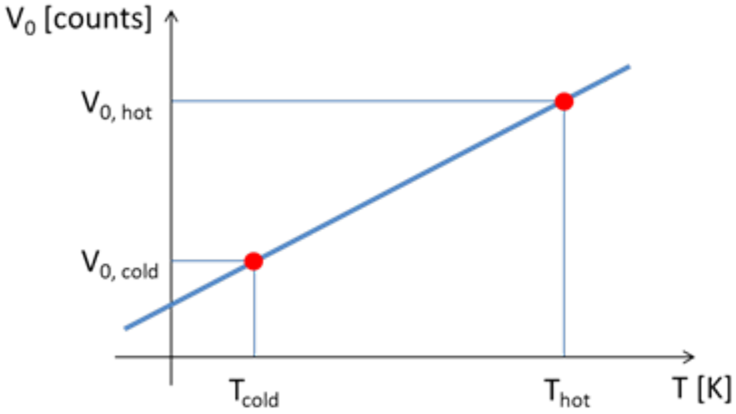


Figure 11-11. Calibration line to be derived from on-board calibration measurements or from pre-launch ground measurements.

However, even though the radiometer is designed to be an extremely linear system, the level of accuracy demanded (< 1 K in many applications) requires to correct even the small non-linearity errors using either pre-launch measurements, or by acquiring -at least- a third independent measurement[8], as sketched in Figure 11-12.

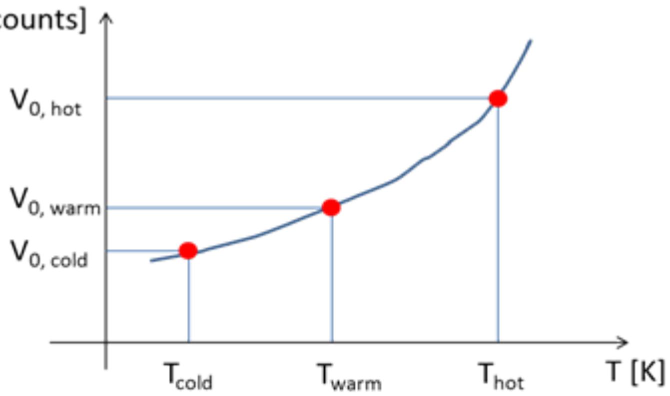


Figure 11-12. Calibration curve to be derived from on-board calibration measurements (T_{cold}, T_{warm} and T_{hot}) or from pre-launch ground measurements.

In order to perform the calibration of each channel (each individual beam at a given frequency band and polarization), the T_{COLD}, T_{HOT} (and eventually T_{WARM}) measurements must be performed. These measurements are computed in blocks **MS3.3.1** using external targets (sky view, microwave absorber, Amazon, Dome C in Antarctica, some ocean regions etc.), and **MS3.3.2.** using internal references (cold and/or warm and/or hot reference temperatures), and are the inputs of the building blocks **MS4.1.1** (determination of the internal calibration parameters), and **MS4.1.2** (determination of the external calibration parameters), to be applied in **MS4.1.3** (error compensation building block), as sketched in Figure 11-13.

[8] Note: with three independent measurements a second order polynomial can be fitted.

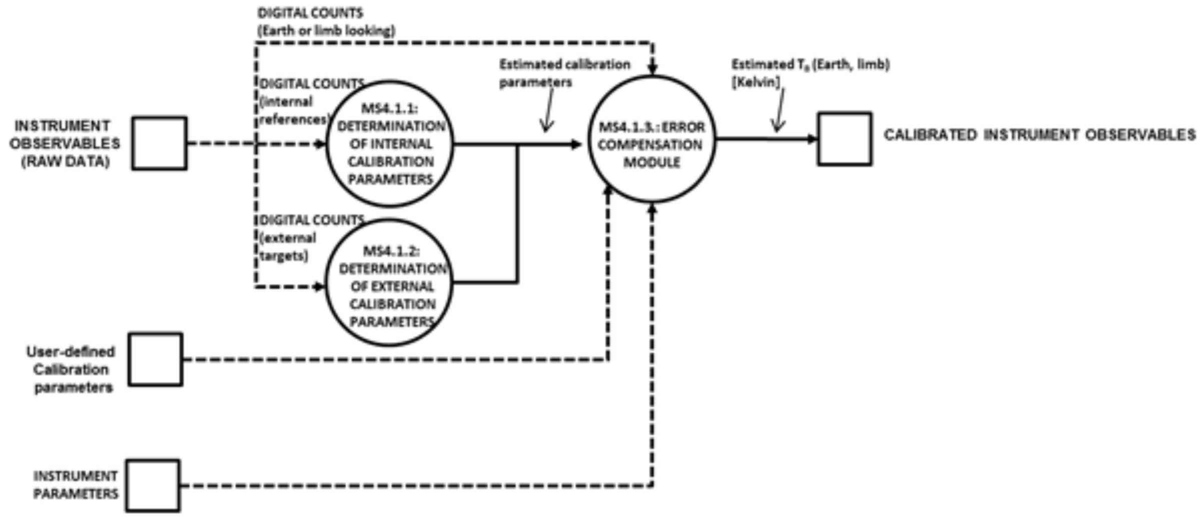


Figure 11-13. Structure and building blocks of the LEVEL 1 Processing Module – Instrument Calibration (MS4.1)

Since the calibration references are not always placed in front of the main antenna, the main antenna is not in the calibration loop, and this complicates the calibration procedure. Some examples:

- the small microwave absorber or the sky mirror placed in front of the antenna feeder, which are typically found in conical scan radiometers, do not allow to calibrate for the ohmic losses of the main reflector, although they are usually small[9]. They also introduce some errors are the swath edges due to diffraction of the sky radiation that require an along-track correction.
- the internal references are placed after a switch located right after the antenna feeder. Therefore, the feeder and the switch losses are not in the calibration loop. The calibrated antenna temperatures must then be referred to the antenna reference plane (in front of the antenna) using auxiliary measurements of on-ground data.
- the sky horn typically used in the radiometers accompanying the radar altimeters are different antennas than the Earth looking ones, and a switch is required to commute between them, therefore, the calibration and the measurement paths are different (see [RD. 111] for an analysis on how the transfer function coefficients are modified from one path to the other one).

After performing the calibration measurements, and before the calibration is actually applied to the raw data, a number of corrections have to be applied:

- Detector non-linearity correction based upon pre-flight characterization of the instrument response, and the actual instrument temperature, or using three measurements as sketched in Figure 11-12.
- Warm bias correction based upon pre-flight measurements of the difference between its effective brightness temperature and its measured temperature. It corrects for contamination by radiation originating from the enclosure of the black body target placed between the main reflector and the antenna feeder, which is influenced by the spacecraft temperature and the Sun or Earth's limb radiance.
- Cold bias correction linearly shifts the brightness temperatures to produce a space-view value consistent with the known radiative background (cosmic background ~2.7 K). It corrects for contamination of the measured background radiation when viewing space by radiation originating from the spacecraft, the Earth limb, the Moon, and -in some bands (e.g. 10.7 GHz)- RFI from geo-stationary satellites [RD. 112].

9

At this point, the error compensation building block (**MS4.1.3**) is applied. From the parameters estimated using the internal references (building block **MS4.1.1**), the antenna temperature $\hat{T}''_A$ (after the antenna) can be estimated as[10]:

$$\hat{T}''_A = a \cdot digital\ counts + b, \tag{3}$$

where a and b are the sensitivity and offset of the calibration line (inverse of that in Fig. 11).

Then, using the antenna ohmic efficiency (η_Ω), estimated from pre-launch measurements or simulations, or from external target measurements (e.g. sky view, building block **MS4.1.2**), the antenna losses correction can be taken into account:

$$\hat{T}'_A = \frac{1}{\eta_\Omega}\hat{T}''_A - \left(\frac{1}{\eta_\Omega} - 1\right)T_{ph}. \tag{4}$$

However, $\hat{T}'_A$ still has to be corrected for along-scan variations (Fig. 14 from [RD. 112]) resulting from changes in the feed-horn field-of-view:

$$\hat{T}_A = \eta_{spill-over}\hat{T}'_A + \left(1 - \eta_{spill-over}\right)T_{sky} + \beta, \tag{5}$$

where $\eta_{spill-over}$ is the spill-over efficiency, T_{sky} is the cosmic background temperature, and β is an offset.

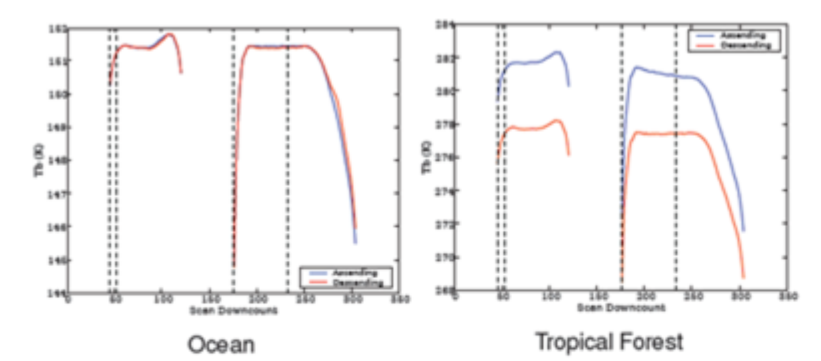
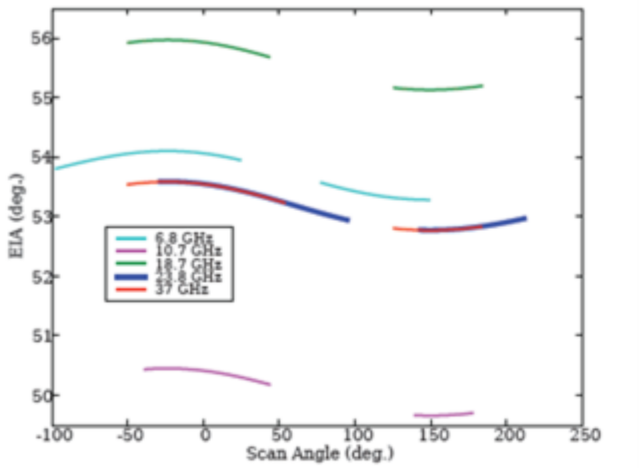


Figure 11-14. Along-scan variations in the 6.8 GHz vertical-polarization channel of Coriolis/Windsat. Note the sharp decrease of the measured antenna temperature at the edges of the swath [RD. 112].

Finally, this correction is complicated by the variation of the Earth Incidence Angle (EIA) along the scan due to satellite pitch/roll offsets (Figure 11-15).



[10] For the sake of simplicity, in these equations a linear relationship has been assumed. If not, a second order polynomial should be used instead.

Figure 11-15. Coriolis/Windsat EIA along scan variations [RD. 112].

Finally, the antenna finite main beam efficiency has to be taken into account to correct for the noise being collected outside the main beam of the antenna. This correction takes into account the finite cross-polarization of the antenna as well. It is performed by deconvolving the antenna temperatures using an approximated formulation, such as the one used in SSM/I[11]:

$$\hat{T}^{p}_{B\,i,j} = \sum_{m=i-1}^{i+1} \sum_{n=j-1}^{j+1} a_{m,n} \cdot \hat{T}^{p}_{A\,m,n} + b_{i,j} \cdot \hat{T}^{p}_{A\,i,j}, \tag{6}$$

Last, but not least, foreign sources (e.g. Sun, Moon, Galaxy plane…) must be corrected for using appropriate models based on the relative position of the external sources with respect to the antenna beam.

User-defined calibration parameters including pre-launch calibration parameters should also foreseen for increased flexibility.

The last step is a quality check. Since all brightness temperatures must be between 50 K and 350 K, if the scan calibration is wrong, and/or if the temperatures are outside this range or are erroneous, they are set to zero and not processed at higher levels.

Finally, in SSM/I, for example, the recovered brigthness temperatures are stored as 2 byte integers, multiplied by 10 (that is, with a 1K/10 resolution).

11.5.2. GEOLOCATION BLOCKINTRODUCTION

When the brightness temperature image is radiometrically calibrated, the data is then processed to perform all the geometrical corrections, including the polarization rotation, geolocalization, and resampling to a pre-defined standard grid in an Earth Coordinate System. The required input data comes from the Geometry module (**MS1**), an includes satellite position, platform attitude, and pointing vectors of the antenna beams. Note that these parameters are already attached as ancillary data to the Level 0 raw data.

Figure 11-16 summarizes the main building blocks, which are described in the following section.

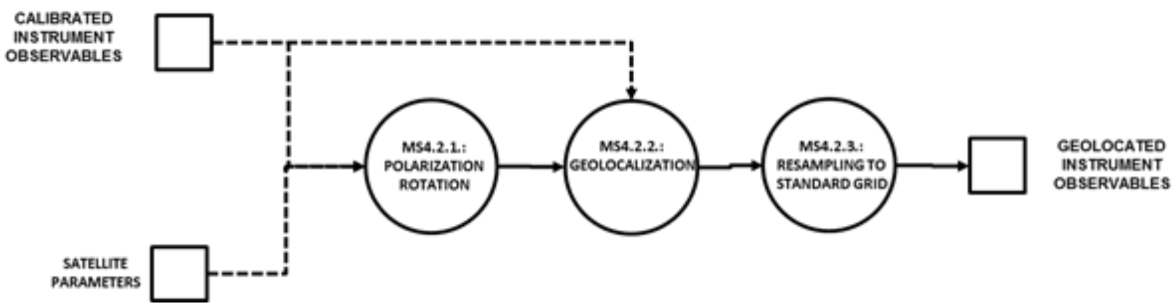


Figure 11-16. Structure and building blocks of the LEVEL 1 Processing Module – Transformation of Reference Frame and Geolocation (MS4.2)

11.5.2.1. Description of Building Blocks

11.5.2.1.1. Polarization rotation

Usually, the first step to transfer the radiometrically calibrated TOA brightness temperatures into geo-located TOA brightness temperatures is the rotation of the polarization reference frame. This converts the X- and Y-polarization brightness temperatures, into H- and V-polarization brightness temperatures, more suitable for later processing in the Level 2 processor. However, in some cases, when this transformation is not feasible (see eqn. (8) below) the Level 2 processor can be designed to operate in

[11] Note: in [RD.20] Wentz uses only the four closest neighbors instead of the eight surrounding neighbors.

the antenna reference frame directly. For this reason a shortcut is drawn from the input parameters up to block **MS4.2.2**: Geolocalization.

The transformation from the antenna reference frame (X- and Y-polarizations) into the local reference frame (H- and V- polarizations) has the form:

$$\begin{bmatrix} \hat{T}_{hh} \\ \hat{T}_{hv} \\ \hat{T}_{vh} \\ \hat{T}_{vv} \end{bmatrix} = \begin{bmatrix} A^2 & -AB & -AB & B^2 \\ AB & A^2 & -B^2 & -AB \\ AB & -B^2 & A^2 & -AB \\ B^2 & AB & AB & A^2 \end{bmatrix} \begin{bmatrix} \hat{T}_{xx} \\ \hat{T}_{xy} \\ \hat{T}_{yx} \\ \hat{T}_{yy} \end{bmatrix}, \tag{7}$$

where $A = \cos\psi$ and $B = \sin\psi$, and ψ is the sum of the Faraday and geometric rotations12: $\psi = \Phi_{geometric} + \Phi_{Faraday}$. For fully-polarimetric radiometers, eqn. (7) can be inverted for any ψ. However, in case of a dual-polarization radiometer only $\hat{T}_{xx}$ and $\hat{T}_{yy}$ are measured and eqn. (7) becomes:

$$\begin{bmatrix} \hat{T}_{hh} \\ \hat{T}_{vv} \end{bmatrix} = \begin{bmatrix} A^2 & B^2 \\ B^2 & A^2 \end{bmatrix} \begin{bmatrix} \hat{T}_{xx} \\ \hat{T}_{yy} \end{bmatrix}, \tag{8}$$

which is singular for $\psi = 45°$, and therefore cannot be inverted.

11.5.2.1.2. Geolocalization

The geolocalization building block (**MS4.2.2**) projects the radiometrically calibrated TOA data (either in the antenna or in the local reference frames) into an Earth Coordinate System. To do that, the satellite position, platform attitude, and instrument pointing directions (the "geometry parameters", **MS1.4,** passed to the Observables Generator and thence the LEVEL1 Processing Module) are used to compute the intersection between the Earth's surface and the antenna boresight direction.

Compared to other EO sensors, microwave radiometers have a very poor spatial resolution. Therefore, it is not surprising to have geolocalization errors of kilometers, on the order of half a footprint. For example, the low frequency channels of SSM/I (with an antenna footprint of 69 x 42 km) have typically geolocalization erros between 20 to 30 km. The final accuracy reaches only ~ 5 km (about 1/8th of a pixel) after correction of the alignment error between SSM/I's boresight and instrument's boresight, and the computation of a 7-day average satellite orbit [RD. 113]. It is hard to believe that the geolocalization errors can be improved better than that such a fraction of the antenna footprint. For this reason, Control Ground Points (CGP) are not used in microwave radiometry for geolocalization purposes.

11.5.2.1.3. Resampling to Discrete Global Grid

The last step is the resampling of the geo-referenced brightness temperatures from the arbitrary coordinates given by the instrument pointing direction, into a standard discrete global grid.

There are a number of approaches with different levels of complexity. Here only two extreme cases are described:

- In SSM/I and SSMIS the grid used is the Equal-Area Scalable Earth-Grid (EASE-Grid). "The methodology for binning the swath brightness temperatures into the 12.5 km and 25 km grid cells has been carefully considered. Previous work (Nimbus-5 ESMR and Nimbus-7 SMMR) used a simple sum and average ("drop-in-the-bucket") approach where the grid cell that contained the center of the observation footprint was given the whole weight of the observation. After reviewing several alternatives to this method of binning, NSIDC concluded that the increase in accuracy obtained with more sophisticated algorithms was not sufficient to warrant increasing the required computer time for binning by a factor of 30 or more; thus, the "drop-in-the-bucket" approach was adopted. The ability to switch to a more sophisticated algorithm in the future has been retained by structuring the archive to accept a fractional number of observations." [RD. 113]

---

[12] Note: the geometric rotation should only be applied in case of a very wide antenna beam, in a case as in SMOS, or when the antenna polarization frame rotates with the direction. Otherwise, only the Faraday rotation must be taken into account, especially at low frequencies (*f* < 10 GHz).

- In SMOS the grid used is the ISEA 4H9 (Icosahedral Snyder Equal Area projection with aperture 4, resolution 9 and shape of cells as hexagon), the geo-localization over the standard regular grid is performed after the image reconstruction process. After the image reconstruction, the Fourier transform components of the brightness temperature image are computed. Then, the exact the brightness temperatures on the standard grid points is recomputed by means of a sort of Discrete Fourier transform. This procedure avoids in principle any interpolation error, at the price of being computationally intensive.

At this point, products can be flagged according to the type of surface being imaged. In some cases, the products are even separated in different files for later processing in separate Level 2 processors, for example, in the case of SMOS, for land and ocean.

11.6. LEVEL-2 RETRIEVAL MODULE

11.6.1. INTRODUCTION

The Level-2 Retrieval module is in charge of obtaining the bio- and geo-physical parameters from the TOA geolocated brightness temperature data over a discrete global grid. The algorithms are very mission specific, however, there are an number of commonalities between them.

The building blocks for the Level-2 Retrieval module are described in the following sections.

11.6.2. DESCRIPTION OF BUILDING BLOCKS

The Level-2 Retrieval module structure is sketched in Figure 11-17. The first block is **MS5.1** in charge of separating the different targets (land, ocean, or atmosphere –over land or over ocean-) from the flag provided in the geolocated and calibrated data products.

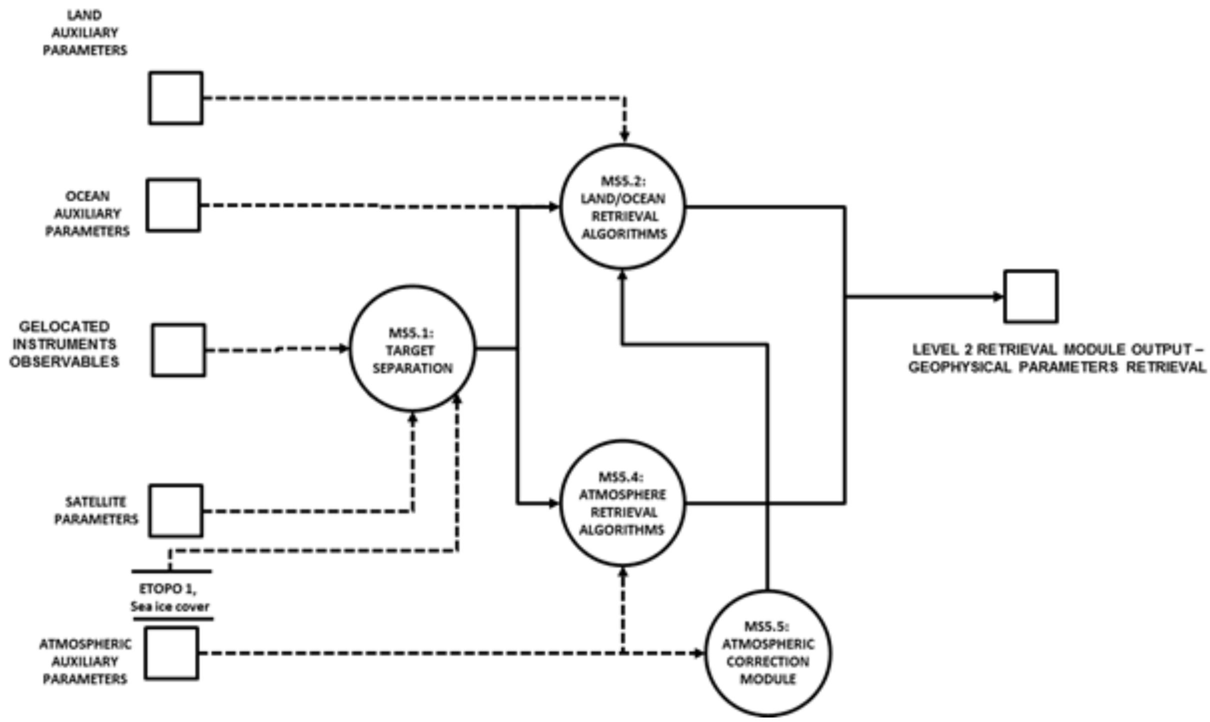


Figure 11-17. Structure and building blocks of the LEVEL 2 Retrieval Module – Gephysical Parameter Retrieval block (MS5)

Then, different retrieval algorithms have to be applied depending on the type of surface being imaged (**MS5.2** for land, **MS5.3** for ocean) and the bio- or geo-physical parameter to be retrieved. Except for the retrieval of atmospheric parameters, the TOA brightness temperatures have to be compensated for the atmospheric effects, and the scattering of external sources.

For the sake of simplicity, to illustrate the problem, a non-scattering atmosphere and a flat sea or land surface will be assumed. The Bottom Of the Atmosphere (BOA) brightness temperature ($T_{B,p}^{BOA}(\theta)$) can be related to the TOA brightness temperature ($T_{B,p}^{TOA}(\theta)$) by:

$$T_{B,p}^{BOA}(\theta) = T_{B,p}^{TOA}(\theta) \cdot L(\theta) - \left\{ \hat{\Gamma}_p(\theta) \cdot \frac{T_{sky}(\theta,\varphi)}{L(\theta)} + T_{up}(\theta) \cdot L(\theta) \right\} - \hat{\Gamma}_p(\theta) \cdot T_{down}(\theta), \quad (9)$$

where $L(\theta)$ is the atmospheric losses, $\hat{\Gamma}_p(\theta)$ is an estimated reflection coefficient of the land/sea surface[13], $T_{sky}(\theta,\varphi)$ is the sky (cosmic + galactic) noise, and $T_{up}(\theta)$ and $T_{down}(\theta)$ are the up-welling and down-welling brightness temperatures of the atmosphere.

In eqn. (9) $L(\theta)$, $T_{up}(\theta)$, and $T_{down}(\theta)$ account for:

- oxygen absorption, which is quite homogeneously distributed, and can be easily accounted for using the surface temperature and pressure,
- water vapor and clouds, which is highly variable in time and space, and can be very important at some frequencies, especially around 22 GHz,
- rain pixels have to be flagged. Rain can be negligible (depending on the application) at the lowest microwave frequencies, but it becomes more important as frequency and rain intensity increase. For intense rains, in the highest frequency channels of most microwave radiometers (~86 GHz or higher) scattering effects are also important.

And $T_{sky}(\theta,\varphi)$ accounts for:

- Sun direct and reflected should be avoided by a proper orbital design and imaging, but sometimes Sun radiation is measured by scattering in the platform, and
- the sky scattering on the Earth's surface is only noticeable at 1.4 GHz and over the sea (which exhibits a larger reflection coefficient). Over land the reflection coefficient is much smaller, and the dynamic range of the brightness temperature to be measured is much large. Above 1.4 GHz, the galactic noise is nearly negligible ([RD. 96]) and only the cosmic background has to be taken into account.

The atmospheric correction building block (**MS5.5**) requires then, the estimation[14] of $L(\theta)$, $T_{up}(\theta)$, and $T_{down}(\theta)$ which relies in radiative transfer models (RTM) using Radiative Transfer Models (RTMs), such as the general one being developed in the frame of [RD. 97]. After the atmospheric corrections, the bio- and geo-physical parameters can be retrieved either using a semi-empirical relationship between the different channels (frequencies and/or polarizations) or using a minimization procedure between the model and the data.
Similarly, the retrieval of atmospheric parameters (**MS5.4**) relies either on the estimation of some variables using a semi-empirical relationship between the different channels (frequencies and/or polarizations), or on the direct inversion of the RTM as in the case of the limb sounders.

11.6.3. SOME EXAMPLES – REAL APERTURE RADIOMETERS

11.6.3.1. Atmospheric applications

11.6.3.1.1. Nadir-looking radiometers and imagers

Two examples are reviewed:

[13] Note that the actual reflection coefficient depends often on the parameters to be retrieved, so either some auxiliary data is used to estimate this term, or an iterative procedure is foreseen to include eqn. (9) completely in the retrieval process.

[14] Actually it may require the estimation of other parameters such as the single scattering albedo, to account for scattering effects at least to first order.

The nadir-looking radiometers used to perform the wet-delay correction in radar altimetry, estimate the cloud water vapour CWV) by means of the following formulas:

$$CWV[kg/m^2] = a_0(\theta) + a_1(\theta)T_{b,lin}(f_1,\theta) + a_2(\theta)T_{b,lin}(f_2,\theta), \quad (10)$$

$$T_{b,lin}(f_i,\theta) = T_{b,0}(f_i) - \left(T'_{eff}(f_i,\theta) - T_{b,0}(f_i,\theta)\right) ln\left(\frac{T_{eff}(f_i,\theta)-T_b(f_i,\theta)}{T_{eff}(f_i,\theta)-T_{b,0}(f_i)}\right), \quad (11)$$

where:

- $T_b(f_i,\theta)$ is the brightness temperature measured by the radiometer,
- $T_{b,0} = 2.7\,K$ is the brightness temperature of the cosmic background,
- $T_{eff}^{(\prime)}(f_i,\theta) = a_2(f_i,\theta)T_s$ are effective atmospheric temperatures, and
- T_s is the surface's temperature

- In conical-scan imagers such as SSM/I, the algorithms to retrieve Liquid Water Content (LWC) are of the form:

$$LWC[kg/m^2] = 1.31.95 - 39.5 \cdot ln(280 - T_{22V}) + 12.49 \cdot ln(280 - T_{37V}), \quad (12)$$

where T_{22V} and T_{37V} are the brightness temperatures at vertical polarization, at the 22 and 37 GHz channels.

As it can be noted from eqns. (10)-(12), the general formulation of the geophysical parameter follows an equation of the form:

$$\text{Parameter} = a_0(\theta) + \sum_{i=1}^{N} a_i(\theta)T_{b,lin}(f_i,\theta). \quad (13)$$

11.6.3.1.2. Atmospheric and Limb Sounders

The radiative transfer equation inverted by both atmospheric and limb sounders has the following form:

$$T_{b,j}(\infty) = \int_0^{\infty} T(z)K_j(z)dz \approx \sum_i T_i K_{j,i}\Delta z_i, \quad (14)$$

where the vector $\bar{T}$ containing estimated temperatures T_i at level i, is derived from:

$$\hat{\bar{T}} = \bar{\bar{S}}_x\bar{\bar{K}}^T\left(\bar{\bar{K}}\bar{\bar{S}}_x\bar{\bar{K}}^T + \bar{\bar{S}}_y\right)^{-1}\left(\bar{T}_b(\infty) - \bar{\bar{K}}\bar{T}_0\right) + \bar{T}_0. \quad (15)$$

In eqn. (15):

- $\bar{\bar{K}}$ is the weighting function matrix, whose elements are $K_{j,i}\cdot\Delta z_i$, being $K_{j,i}$ the weighting function at layer i and frequency j, and Δz_i the thickness of layer i,
- $\bar{\bar{S}}_x$ is the error covariance matrix of the atmospheric temperature profiles,
- $\bar{\bar{S}}_y$ is the error covariance matrix of the measured brightness temperatures,
- $T_{b,j}(\infty)$ is the measured brightness temperature at frequency j, and
- $\bar{T}_0$ is a climatological temperature profile known as "a priori" profile.

11.6.3.2. Ocean, Land and rain rate applications

The following examples correspond to different applications derived from SSM/I data to illustrate commonalities, but apply to other imagers such as AMSR-E, and SSM/I follow-on missions:

11.6.3.2.1. Ocean applications: wind speed retrieval

The 19.5 m height wind speed retrieval algorithm in SSM/I uses the T_{19V}, T_{22V}, and $T_{37V,H}$ information, combined linearly:

$$U_{19.5m}[m/s] = 147.9 + 1.0969 \cdot T_{19V} - 0.4555 \cdot T_{22V} - 1.76 \cdot T_{37V} + 0.786 \cdot T_{37H}. \quad (16)$$

The second to fourth terms in eqn. (16) perform the atmospheric opacity correction, and obviously, the 22 GHz channel information is performing the water vapor correction. The last term T_{37H} is the one that contains the wind speed information, since all other vertical polarization channels have null or very small sensitivity to surface roughness around the 53° incidence angle.

11.6.3.2.2. Land applications: snow depth and snow water equivalent retrieval

The Snow Depth (*SD*) and Snow Water Equivalent (*SWE*) retrieval algorithms in SSM/I include the $T_{19V,H}$ and T_{37V} information, combined in a parameter called the Spectral Polarization Difference (*SPD*) as follows:

$$SPD = (T_{19V} - T_{37V}) + (T_{19V} - T_{19H}), \quad (17)$$

$$SD[cm] = \begin{cases} 0.68 \cdot SPD - 0.67, & \forall data, \\ 0.72 \cdot SPD - 1.24, & T_{max,air} < 0, \end{cases} \quad (18)$$

$$SWE[mm] = \begin{cases} 2.20 \cdot SPD - 7.11, & \forall data, \\ 2.02 \cdot SPD - 7.42, & T_{max,air} < 0, \end{cases} \quad (19)$$

11.6.3.2.3. Atmospheric applications: rain rate over land and ocean

In the following lines a number of algorithms are listed to perform the rain rate estimation both over land and over the ocean. They are included to show the general shape of the equations used in the retrievals.

Over land:

Ferraro's algorithm:

$$\mathrm{SI} = 438.5 - 0.46 \cdot T_{19v} - 1.735 \cdot T_{22v} + 0.00589 \cdot T_{22v}^2 - T_{85v}, \quad (20)$$

$$\text{Rain rate } [mm/h] = -2.71 + 0.362 \cdot \mathrm{SI}. \quad (21)$$

Ferriday's algorithm:

$$\text{Rain rate } [mm/h] = (T_{19v} + T_{22v} - T_{37v} - T_{85v})/7. \quad (22)$$

Over the ocean:

Ferraro's algorithm:

$$\mathrm{SI} = -182.7 + 0.75 \cdot T_{19v} - 2.543 \cdot T_{22v} + 0.00543 \cdot T_{22v}^2 - T_{85v}, \quad (23)$$

$$\text{Rain rate } [mm/h] = -1.05 + 0.149 \cdot \mathrm{SI}. \quad (24)$$

Ferriday's algorithm:

$$\text{Rain rate } [mm/h] = (T_{19v} + T_{37v} - T_{85v} - T_{85H} + 50)/10 \,. \tag{25}$$

Additionally, Bauer's algorithm that has a different formulation that all others is presented:

$$ln(\text{Rain rate}[mm/h]) = 17.45 - 0.1761 \cdot 10^{11} \cdot T_{19V}^{-4} - 0.0057 \cdot T_{22V} + 0.1404 \cdot 10^{6} \cdot T_{19V}^{-2} \quad - 0.1188 \cdot 10^{-6} \cdot \left(T_{19V} + T_{19H}\right)^{4} + 0.1944 \cdot 10^{3} \cdot \left(T_{22V} - T_{85H}\right)^{2} \cdot \left(T_{19V} + T_{19H}\right)^{4} + 0.1944 \cdot 10^{-3} \cdot \left(T_{22V} - T_{85H}\right)^{2} \tag{26}$$

The inspection of eqns. (16) to (26) shows that the general form of the equations is a polynomial on the variables that are the different brightness temperatures measured in different channels (frequencies and polarization) and including all cross-terms:

$$\text{Parameter} = \sum_{m_1=1}^{M} \dots \sum_{m_{B-1}=1}^{M} \sum_{m_B=1}^{M} a_{m_1 \dots m_{B-1}, m_B} T_1^{m_1} \dots T_{B-1}^{m_{B-1}} T_B^{m_B}, \tag{27}$$

This equation includes Bauer's rain rate retrieval algorithm, if we assimilate $ln(\text{Rain rate})$ as the parameter to be retrieved.

However, in MOST retrieval algorithms the polynomial is limited to first order, there are not cross-terms, and only in some cases a second order term on T_{22V} is included to perform a better correction of the water vapor effects. Therefore, the most common retrieval equation has the form of eqn. (28):

$$\text{Parameter} = c + \sum_{m=1}^{M} \sum_{p=\{H,V\}} a_{m,p} T_{b,p}(f_m, \theta) + b \cdot T_{b,V}^2 (f = 22GHz, \theta), \tag{28}$$

11.6.4. SOME EXAMPLES – SYNTHETIC APERTURE RADIOMETERS:

Although synthetic aperture radiometers are not the focus of this report, it is worth to briefly describe the approach used in SMOS, which is totally different from the previous ones. In a LEO instrument such as MIRAS in SMOS, the platform movement allows to have many partially overlapping images of the same area. Therefore, the multi-look and multi-incidence angle capabilities can be exploited to better retrieve the geophysical parameters, or to retrieve more than one even using one single frequency. For example, in SMOS, over land soil moisture and vegetation water content, and over the ocean sea surface salinity and an "effective" wind speed.

In this last case, the retrieval algorithm follows the minimization of the general equation bellow, until some convergence criteria are met:

$$\varepsilon = \sum_{n=1}^{N_{observations}} \left\{ \left(\vec{F}^{model}(\theta_n, \vec{x}) - \vec{F}^{data}(\theta_n, \vec{x}) \right)^T \left(\bar{\bar{C}}_{Full-pol/\,Dual\,pol}^{Earth\,ref\,frame, odd(n)/even(n)} \right) \left(\vec{F}^{model}(\theta_n, \vec{x}) - \vec{F}^{data}(\theta_n, \vec{x}) \right) \right\}, \tag{29}$$

where:

- $\vec{F}^{model/data}(\theta_n, \vec{x})$ are vectors that contain the modeled and measured observables, and its structure depends on the formulation of the retrieval problem:
    - $\vec{F}^{model/data}(\theta_n, \vec{x}) = [T_{hh}(\theta_n, \vec{x}), T_{vv}(\theta_n, \vec{x})]^T$ if the problem is formulated in terms of the brightness temperatures in the Earth's reference frame (H and V polarizations),
    - $\vec{F}^{model/data}(\theta_n, \vec{x}) = \left[T_{xx}(\theta_n, \vec{x}), T_{yy}(\theta_n, \vec{x})\right]^T$ if the problem is formulated in terms of the brightness temperatures in the antenna's reference frame (X and Y polarizations), and
    - $\vec{F}^{model/data}(\theta_n, \vec{x}) = [I(\theta_n, \vec{x})]^T = [T_{xx}(\theta_n, \vec{x}), +(\theta_n, \vec{x})]^T$ if the problem is formulated in terms of the first Stokes parameter.
- $N_{observations}$ is the number of measurements acquired of the same location in a satellite overpass,

- $\vec{x}$ is a vector containing the true geophysical parameters affecting the emissivity (brightness temperatures), and $\hat{\vec{x}}$ containing the geophysical parameters that will be estimated, and
- $\bar{\bar{C}}^{Earth\ ref\ frame, odd(n)/even(n)}_{Full-pol/Dual\ pol}$ is the error covariance matrix that depends on the SMOS operation mode (Full-polarimetric or Dual-polarization) and the reference frame (antenna or Earth-based) and if the snap-shot is even or odd.

For other synthetic aperture radiometers the retrieval algorithm approach will depends very much on the orbit and application, for example in geo-stationary instruments such as the upcoming as GAS and geoSTAR there is no platform movement, but they can be treated as push-broom atmospheric sounders, and the retrieval algorithm (eqn. (15)) can be applied to all pixels (synthetic beams) simultaneously.

11.7. PERFORMANCE EVALUATION MODULE

11.7.1. INTRODUCTION

In every mission simulator, results must be evaluated to assess the suitability of the proposed configuration (platform + instrument + geo-physical parameter retrieval) to satisfy the mission requirements. The Performance Analysis Module (**PEM**) performs this type of studies.

It is foreseen to perform this evaluation in two levels:

- L1B geo-located brightness temperatures:
  - radiometric performance in terms of:
    - the radiometric sensitivity (associated to noise and finite integration time),
    - the radiometric accuracy (rms error for different pixels vs the mean value averaged for all pixels in FOV or region defined by the user),
    - the radiometric bias (correspondence of mean value of retrieved TB image vs. input brightness temperature image within the FOV or region defined by the user),

    The mathematical definitions of the radiometric performance is given in Table 11-2 below.
  - angular/spatial resolution performance in terms of:
    - the angular resolution (width of synthetic beam),
    - the spatial resolution (computing pixels footprints),
- L2 bio- or geo-physical parameters retrieved:
    - rms between the retrieved and the original parameters used as input of the simulation,
    - bias between the retrieved and the original parameters used as input of the simulation.

Table 11-2. Definition of radiometric errors

Random errors (noise due to finite integration time)	Temporal average	Zero	0
	Temporal standard deviation	Radiometric resolution[1]	$\Delta T_{resolution} \approx \sqrt{\frac{\sum_{i=1}^{M}\left(\hat{T}_B(\xi,\eta,t_i) - \left\langle \hat{T}_B(\xi,\eta,t)\right\rangle_t\right)^2}{M-1}}$
Systematic errors	Spatial average	Radiometric bias[2] (scene bias)	$\Delta T_{bias} = \frac{1}{N}\sum_{i=1}^{N}\left(\left\langle \hat{T}_B(\xi_i,\eta_i,t)\right\rangle_t - T_B(\xi_i,\eta_i)\right)$

(instrumental errors)	Spatial standard deviation	Radiometric accuracy[2] (pixel bias)	$\Delta T_{accuracy} \approx \sqrt{\frac{\sum_{i=1}^{N}\left(\left\langle \hat{T}_B(\xi_i,\eta_i,t)\right\rangle_t - T_B(\xi_i,\eta_i,t)\right)^2}{N-1}}$

[1]Computed in MonteCarlo mode with a preferably homogeneous scene,
[2]Computed in snap-shot mode.
Additionally, several MonteCarlo simulations can be computed in each case to better estimate both the radiometric errors.

Additionally, platform and instrument quality assessment has to be performed to analyze the suitability of their behavior (e.g. platform attitude is within predefined limits, instrument's physical temperature and gradients are within pre-specified limits…):

11.7.2. DESCRIPTION OF BUILDING BLOCKS

The Performance Analysis Module (**PEM**) module will compare different outputs of the simulation process (L1B – geolocated brightness temperatures and L2 – retrieved geo-physical parameters) and original brightness temperatures and/or geo-physical parameters to assess the instrument and retrieval algorithms performance.

Figure 18 presents the different building blocks for the performance analysis. **PEM1** presents the platform parameters (including orbit ephemeris, LTAN, attitude – Euler angles etc.), **PEM2** presents the instrument parameters (receivers' physical and noise temperatures, bandwidth, antenna co- and cross-polar patterns…), **PEM3** and **PEM4** are the ones that evaluate the instrument end-to-end performance at L1 (brightness temperature and error maps as a function of the polarization and snap-shot) and L2 (geophysical parameters and error maps), respectively. Finally, **PEM5** presents all the previous results in the form of tables and/or as graphical results.

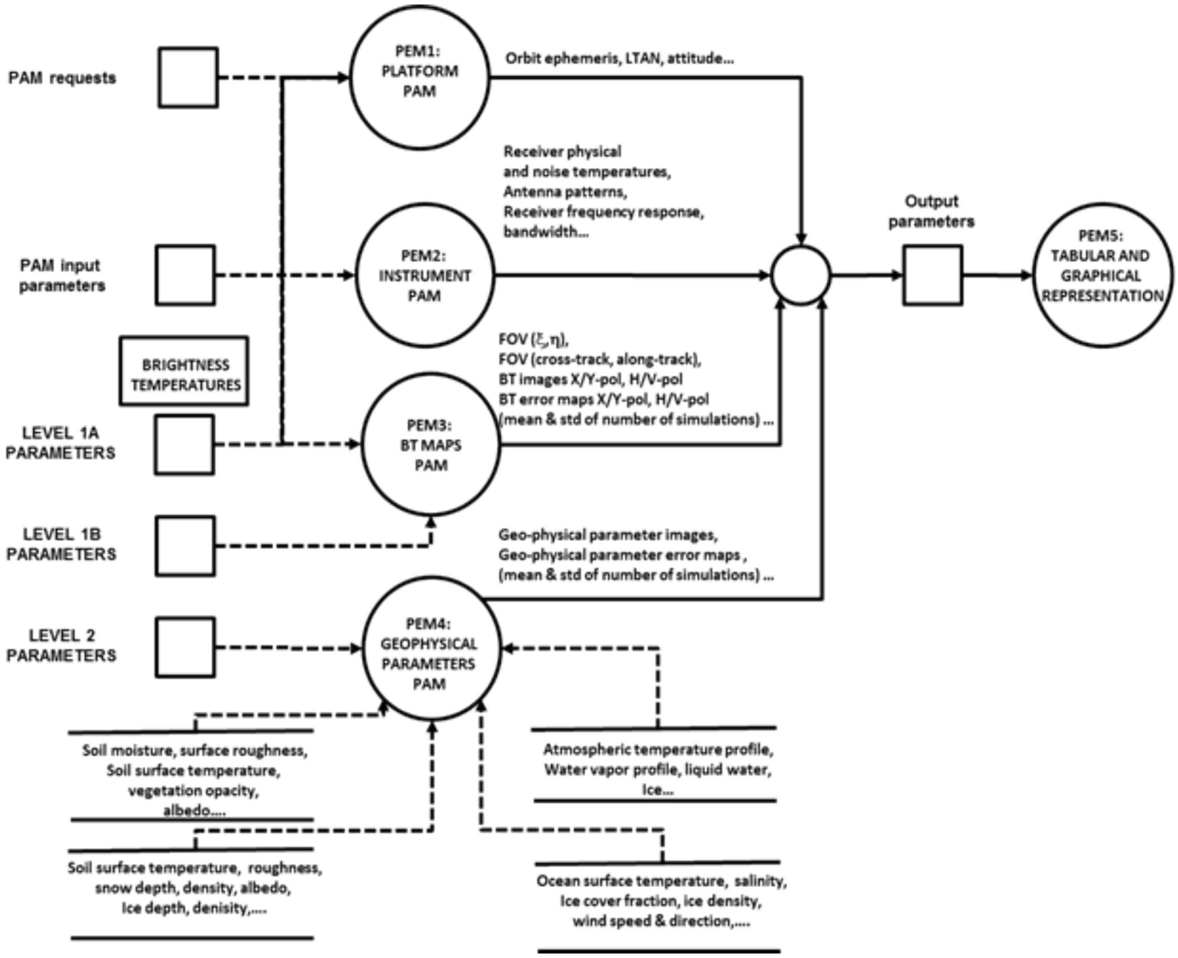


Figure 11-18. Structure and building blocks of the 6th module:

The PERFORMANCE EVALUATION MODULE (PEM)

11.8. CONCLUSIONS

This section has presented a proposal for an E2E Mission Simulator Architecture for Passive Microwave instruments. The architecture is composed of three main blocks: a mission definition module (**MD**), a mission simulation module (**MS**), and a performance evaluation module (**PEM**). At its time, the **MS** module is composed of seven blocks that compute specific parameters of the simulation process: **MS1**: geometry module, **MS2**: scene generator module, **MS3**: instrument module, including antennas and receivers, **MS4**: Level 1 processing, whose outputs are calibrated and geolocated instruments observables, **MS5**: LEVEL 2 Retrieval algorithm (geophysical parameters), and a last module called **PEM**, which stands for Performance and Evaluation Module, and it is in charge of comparing the correspondence between the input scene and the L1 output (instrument's performance only), or the input geophysical parameters (used to compute the input scene) and the L2 output (retrieved geophysical parameter: end-to-end performance).

Each of these modules has been sub-divided in smaller building blocks to perform basic operations.

The proposed architecture is believed to be flexible enough to be used for all passive microwave instruments (real aperture), and even –with some limitations- to synthetic aperture radiometers, if they are defined as push-brooms with as many beams as independent directions exist in the reciprocal grid of the spatial frequencies sampled by the array [RD. 115].

Most probably some modules and building blocks can be shared with other EO sensors, especially **MS1**, which is instrument independent, and to some extent **MS3**, with active microwave sensors, and **MS4**, with passive optical sensors.

12. REFERENCE ARCHITECTURE FOR ACTIVE OPTICAL INSTRUMENTS

12.1. PROPOSED ARCHITECTURE

The Active Optical Instrument end-to-end simulator is built to evaluate the mission performance with respect to the mission specifications. It shall therefore enable a direct comparison in absolute value between the variables of interest introduced as model inputs and the retrieved variables as model outputs. It is fully necessary for the optimization of system performances and the derivation of expected error budget. The Lidar simulator requires the following two sequential chains (see Figure 12-1):

- A forward processing chain to address the so-called direct problem: it is built around the Lidar signals simulator, which starts from the space borne Laser emission with propagation into the atmosphere down to the surface and then light backscattering to the space borne Lidar.
- A backward processing chain to address the inverse problem: it is built around the Lidar signals and data processor, it starts from Lidar signals (level-1a data) generated by the Lidar signals simulator through various Level-1b and Level-2 data with higher levels data if applicable.

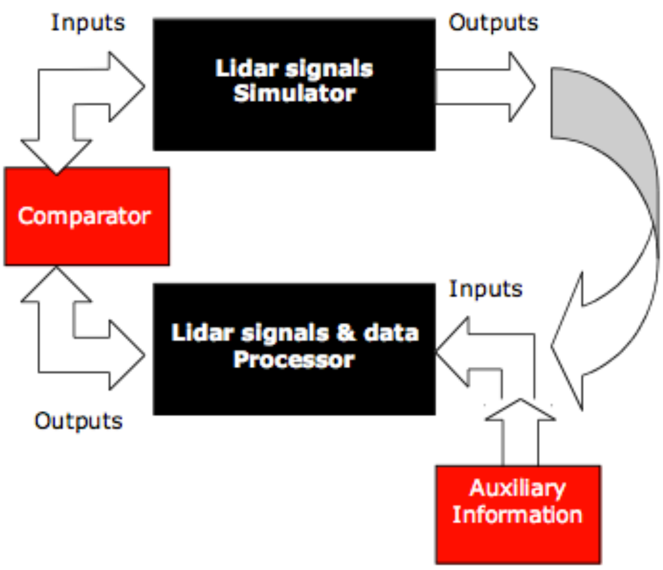


Figure 12-1. Basic schematic of the Lidar mission performance simulator

The proposed data flow can therefore be divided into three parts.

1) The Lidar data generation flow that involves all the modules necessary for the Lidar signal and noises simulation. This module is common to all Lidar missions and techniques
2) The Lidar data processing flow that involves all the modules necessary for the retrieval of various Level products. This module is Lidar technique dependent
3) The verification at all stages in the processing module against the equivalent product in the data generation module, that allows estimating instrument performance wrt retrieval accuracy and so provides feedback on instrument design to achieve the desired precision.

A common architecture for active and passive instruments can be used (referring to Figures 9-14 and 9-19). Accordingly, the architecture for active optical instrument is

- Laser Transmitter Block: generates the pulsed laser signal according to instrument characteristics and convolutes the pulsed laser with impulse response of scene to generate lidar signal. It could include
  - Blocks: Transmitter (Tx) laser, Scene Convolution
  - Inputs: scene data

- Outputs: lidar scene

- Spatial Block: spatial resampling to the final ground sampling distance taking into account instrument MTF and spatial non-uniformity sources. Output is the scene within the footprint of the beam. It is relevant to IPDA technique and ranging and topography applications. Includes:
  - Blocks: Laser pointing/Receiver telescope, Rx/Tx co-alignment
  - Inputs: lidar scene, instrument configuration parameters, other input parameters
  - Outputs: spatial resampled lidar scene

- Spectral Block: spectral resampling from the finer working resolution of the scene generator to the spectral configuration of the sensor.
  - Blocks: Front-end optics
  - Inputs: spectral resampled lidar scene
  - Outputs: at-sensor scene

- Radiometric Block: transforms at-sensor scene to digital counts.
  - Blocks: Lidar Detection (Dx) unit
  - Inputs: at-sensor scene
  - Outputs: digital counts and instrument noises

Data pre-processing Block: transforms the signal captured by the sensor taking into account the Analog-to-Digital (A/D) conversion.

- Blocks: Signal recorder and onboard payload computer
- Inputs: digital counts and instrument noise
- Outputs: raw and ancillary data

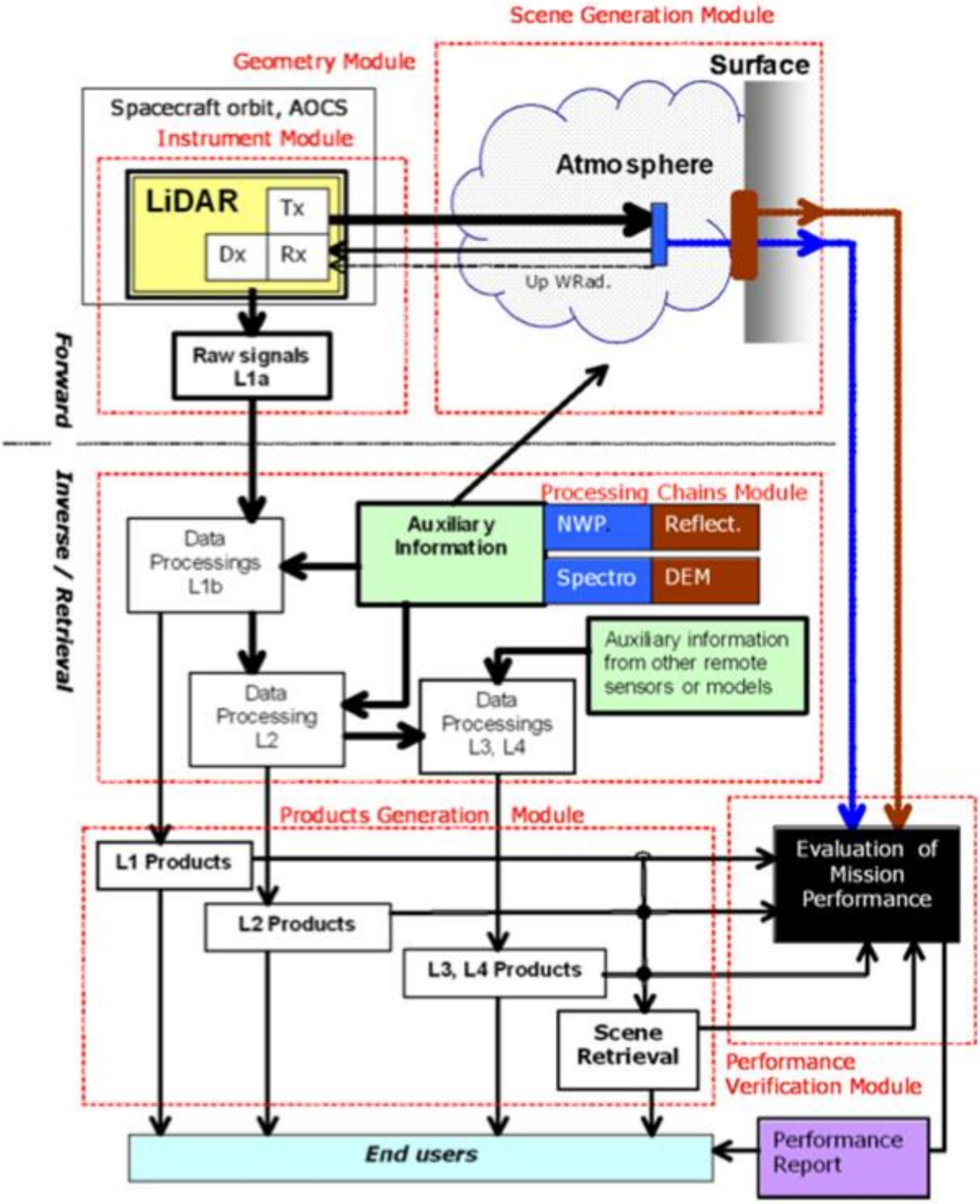


Figure 12-2. Architecture for Lidar E2E simulators

Lidar observations from space are conducted for 1) vertical profiles or column content of key atmospheric variables such as clouds and aerosols optical properties, trace gas densities (H_2O, CO_2, CH_4) and wind, and 2) surface topography and canopy vertical distribution. Usually a lidar mission addresses one main product that drives the overall mission and instrument concept. In addition, the mission can be used to obtain valuable spin-off products.

Several referenced lidar techniques are suitable for atmospheric and surface remote sensing. Cloud and aerosols vertical profiles can be retrieved using simple elastic backscatter lidar technique (lidar CALIOP on CALIPSO mission) or high spectral resolution elastic backscatter lidar technique (lidar ATLID on EarthCARE mission). Range resolved trace gas densities profiles can be retrieved using Differential Absorption Lidar (DiAL) technique (WALES for H_2O). Trace gas densities column can be retrieved using Integrated Path Differential Absorption (IPDA) Lidar technique (A-SCOPE for CO_2, MERLIN for CH_4). In this later case, because the sources and sinks for CO_2 and CH_4 are at the surface, the emphasis for high sensitivity is put on the Lidar weighting function to have a maximum close to the surface. The weighting function is computed using spectroscopic information about the molecule of interest and atmospheric primary variables (surface pressure, temperature, humidity). Horizontal wind velocity can be retrieved using the Doppler frequency shift Lidar technique (lidar ALADIN on ADM-Aeolus). Surface topography and canopy vertical distribution can be retrieved using accurate ranging and geo-location of measurements (ICESat-1 mission).

All the Lidar missions consider data driven simulations and share the same basic architecture as displayed on Figure 12-2. It is a strong communality. Figures 11-1 and Figure 12-2 shows the architecture for a Lidar mission performance simulator displaying the interconnection between the components of the physical Lidar simulator and the high-level modules identified in Section 4.1. Red dashed lines boxes outline the main components of the Reference Architecture for E2E simulator.

After launch, the processing modules can be fed by space borne Lidar data.

If we restrict to Level-2 data, the reference architecture for the Lidar mission simulator is therefore based on the six high-level building blocks as presented before (Section 6.2 and Fig. 6.1):

- Geometry Module and relevant sub modules: orbit simulator, attitude simulator, AOCS/instrument coupling, scene interaction geometry
- Scene generator Module to be coupled with geometry module through the scene interaction geometry sub-module
- Instrument Module to be coupled with scene generator module and geometry module through the AOC/instrument coupling module
- Level-1 Processing Module
- Level-2 Retrieval Module
- Performance Evaluation Module (PEM)

Usually, the scene generator is run at first stage according to Fig. 6.1. The geometry module is common to all missions with various degrees of complexity. Notice that most space borne lidars operate in nadir viewing and sample the atmosphere at random. The constraint sets on orbit selection is mostly due to the revisit time period.

The scene generator module is common to active and passive optical instruments with various degrees of complexity.

The instrument module can be made common to active and passive optical instruments if we consider that the difference between the two bears mostly on the incoming light to generate the useful signals and the use of a laser transmitter for lidar to provide the useful signal instead of solar scattered light or IR emission. This module can be implemented in a modular and generic way so that it can be applied to different Lidar configurations according to the mission objective.

Regarding the Level-1 processing module and Level-2 retrieval module, the idea is to define them as flexible enough to adopt new processing features or algorithm/models updates.

12.1.1. LIDAR SIGNALS SIMULATOR

This part of the simulator is specific to lidar missions but even so it has communalities with all missions i.e. geometry module, and optical passive instrument i.e. scene generator module. The 3 key building blocks of the Lidar signal simulator (see Figure 12-1) is presented on Figure 12-3. It can be re-used. *It is common to active (Lidar) and passive optical instruments.*

We follow the "geometry module first" approach, even if it seems more general to start with the scene generation module. This simulator is appropriate for 1D column and 3D simulations along a full orbit.

The **Inputs** are the following 3 building blocks

- The geometry module (including atmosphere and surface)
- Scene generator module
- Instrument module

The **Outputs** are the

- Lidar useful signal (see TN1)
- Radiometric background (see TN1)
- Various noises at lidar output
- Range resolution along the Lidar LOS (see TN1) and range ambiguity

The outputs of the Lidar signals simulator are the inputs to the Lidar signals and data processor (see Figure 12-1):

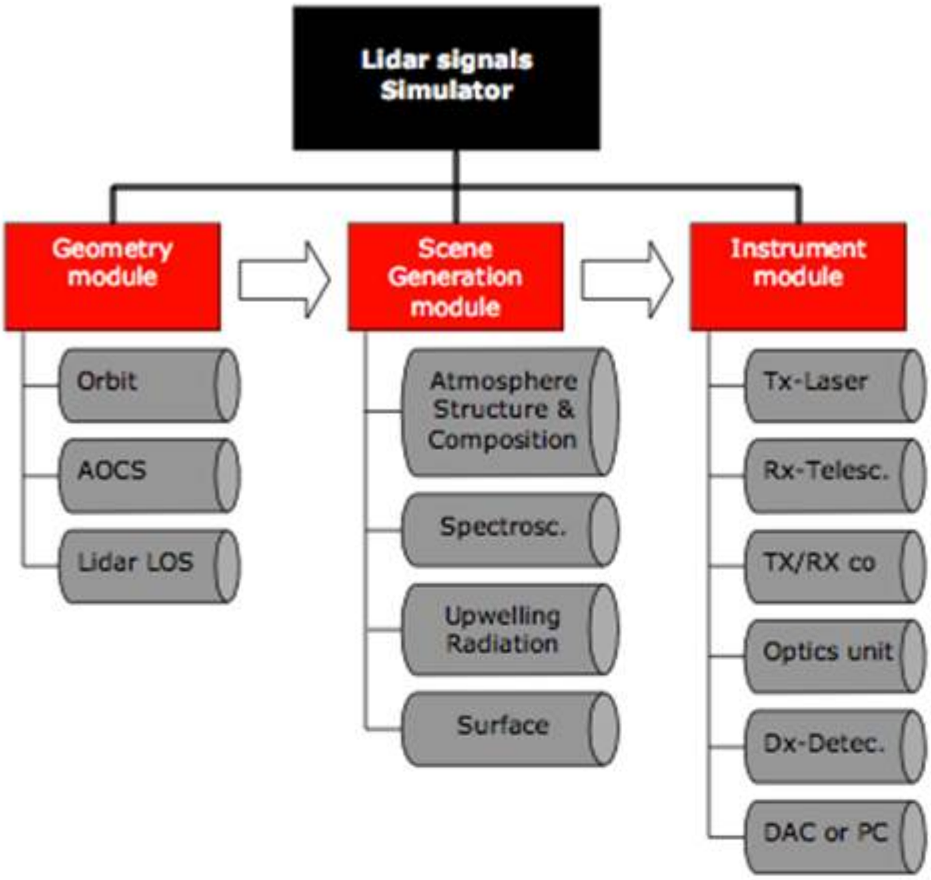


Figure 12-3. Building blocks of the Lidar signals Simulator. This Lidar Simulator Architecture is common to all Lidar missions and it can be easily re-used.

12.1.2. LEVEL-1 AND LEVEL-2 PROCESSORS

The inputs of the Lidar Signals and data Processor are the L1-a data generated by the Lidar signals simulator. The Lidar Signals and data Processor implements the Level-1b processing algorithms on the Level-1a data, and then Level-2 processing algorithms on Level-1b data, and so on to higher levels processing and scene retrieval algorithms as shown on Figure 12-2 and Figure 12-4.

It is important to point out that if the basic architecture is the same for the various Lidar applications. Regarding Level-1 Processing Modules, the atmospheric remote sensing lidar missions can be distributed into two categories: 1) range resolved atmospheric profiling such the return signals are provided by molecules and particles scattering processes, and 2) atmospheric total column such the return signals are provided by surface diffuse reflections. The surface topography missions are in a third category. There is communality for the mission in the first category i.e. range resolved atmospheric applications. The purpose of level-1 processing module is to obtain calibrated attenuated lidar signals. Then, all Level-2 retrieval modules are different for every application according to the lidar techniques (see overview on retrieval methods provided in TN001). In lidar missions, the communalities are on auxiliary information. The present sub-section is based on Algorithm Theoretical Basic Developments (ATBD) made available for public released on internet. Notice that the ATBD resulted of research works, some of them are still under development and not published yet for review by the Lidar community.

Figure 12-4 shows the modules and building blocks of the Lidar signals and data processor for Lidar Missions. The main three modules are:

- Level-1 Processing Module
- Level-2 Retrieval Module
- Level-3 and Level-4 Retrieval Module (out of the scope of this activity)

Each of these modules will have an implementation depending on the type of application, which are, starting from the left to the right:

1) Simple elastic backscatter lidar for surface ranging (ex: GLAS on ICESat),

2) Simple elastic backscatter Lidar for atmosphere profiling (ex: GLAS on ICEsat, CALIOP on CALIPSO),

3) High Spectral Resolution backscatter Lidar for atmospheric profiling (Atlid on EarthCARE),

4) High Spectral Resolution Doppler backscatter Lidar for atmospheric wind profiling (Aladin on ADM-Aeolus),

5) Range resolved Differential Absorption Lidar for trace gas density profiling (WALES),

6) Integrated Path Differential Absorption Lidar for trace gas density column content (A-SCOPE, MERLIN).

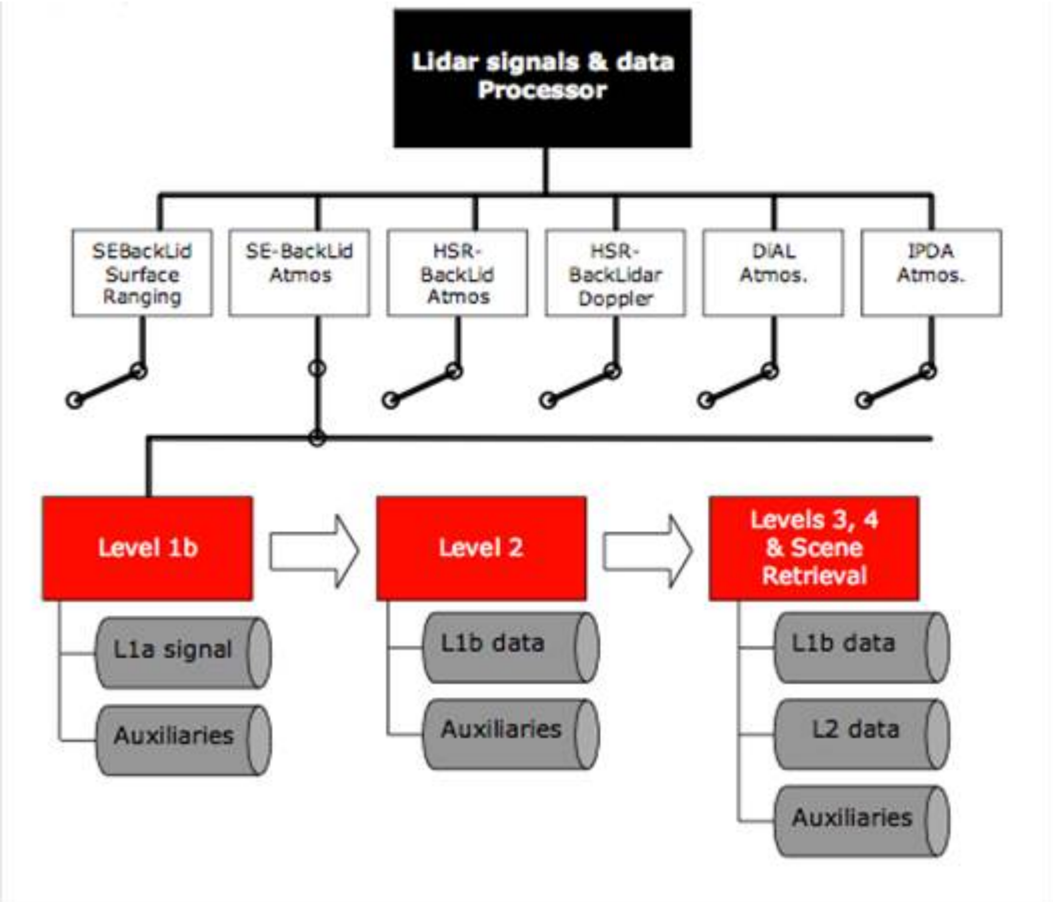


Figure 12-4. Building blocks of the data processor for Lidar Missions.

For clarity, the data products are presented below in Table 12-1

Table 12-1. Definition of Lidar Products

Level	Lidar data	Products	Comment
0	*Time ordered raw data down linked*	*Raw data after restoration of the chronological data sequence for the instrument operating in observational mode*	*Not applicable to Lidar signals simulator* Applicable to down linked real data
1a	Annotated data	Level 0 data with all calibrations computed and appended, but not applied. Annotation from time, satellite position, velocity and pointing	Applicable to Lidar signals simulator and down linked real data
1b	Geo-located and calibrated products	Level 1a data with radiometric calibration applied using auxiliary information when necessary, and processed to get physical quantities relevant to mission products	Applicable to Lidar signals simulator and down linked real data
2	Mission products	Level 1b data processed with auxiliary information	Applicable to Lidar signals simulator and down linked real data

12.2. GEOMETRY MODULE

The data flow of the generic Geometry Module is presented on Fig. 7-1. This block is in charge of the computation of the lidar measurement geometry: handling the orbit and attitude of the satellite, the lidar LOS pointing with respect to local vertical, and correspondences between lidar coordinates (range, azimuth) and geographic coordinates (e.g. latitude, longitude, height).

Building blocks (see Figure 7-1 and Figure 12-3):

- Orbit Simulator (see Figure 7.?). This building block is an input received from the external (i.e. from the user). Most lidar missions are operated on a sun synchronous orbit to provide sufficient electric power to the lidar payload and spacecraft. The main parameter for the lidar signals simulator are the height that determines the range to the surface and repeat cycle that determines the revisit cycle and sampling pattern at Earth surface. A slow orbit height reduction due to atmospheric lag is not of a first importance for a phase A instrument simulator with the exception of ranging and topography application like ICESat-1.
- Attitude Simulator (see Figure 7.?). This building block controls the platform attitude and provides knowledge on platform attitude. It is of primary importance for ranging, HSR, DiAL, IPDA and Doppler frequency shift lidar technique and to a lesser extend to simple elastic backscatter Lidar technique.
- AOCS/instrument coupling (see Figure 7.?). The Lidar mission for atmosphere can live with some cross track pointing that induce no Doppler frequency shift. On the contrary, along track missed pointing are detrimental to Lidar measurements that depend on Doppler frequency shift. It is obvious for a wind Lidar mission like ADM-Aeolus if considering that a fraction of the spacecraft velocity (7 km/sec) may contribute to the LOS component. For a DiAL or an IPDA mission, the wavelengths of the laser emission and the selected wavelength for absorption will not match. It will result in a measurement bias and larger random error because the optical depth will be different from optimal. For a HSR Lidar like EarthCARE the wavelength of the scattered light and the Fabry-Perot filter will not match and it will result in reduced signal for particles and bias on aerosols and clouds optical properties.
- Scene interaction Geometry (see Figure 7.?). Most Lidar missions use a fix near nadir pointing. In most cases, the LOS is off nadir by several degrees to avoid specular reflections from oriented ice particles, still water areas or ice sheets. Lidar LOS fluctuations and uncertainties on AOCS have the same impact on the measurements.
- Acquisition timeline receives the parameters of the acquisition. This function is not of first importance for a phase A instrument simulator.

12.3. SCENE GENERATOR MODULE

The data flow of the generic Geometry Module is presented on Figure 7-2. This block implements the forward module from the physical properties of atmosphere and terrain for generation of lidar signal. One can also refer to section 9.3.2 for module description relevant to passive optical instruments, and especially to section 9.3.2.2 and Figure 9-11.

The peculiar building block of the scene generation module for Lidar is intended to reproduce the actual scene at various degrees of complexity in terms of atmospheric structure and composition, and surface properties. Atmospheric profiling mission does not require accurate surface simulation for it is only used to generate the unwanted radiometric background light. In the case of IPDA technique a refined description of surface properties is necessary to evaluate the Lidar performance. Ranging applications for surface topography and canopy description require highly detailed surface properties. In this later case, this block has the capability of “translating” physical terrain quantities into reflectivity.

Building blocks:

- Atmosphere structure and composition to describe
  - Clouds and aerosols layers,
  - Profiles of trace gases of interest such as CO2 and CH4,
  - Primary meteorological variables such as the temperature profile from the surface to the height of interest for the mission, the surface pressure, the water vapour (H2O) profile, the wind field.

- Spectroscopy. This building block describes the spectroscopy and optical properties of
  - Trace gases (HITRAN and GEISA databases)
  - Clouds and aerosols (ESA Reference Model of the Atmosphere for example)
- Upwelling radiation. This building block describes the background light collected by the Lidar receiver. Usually it is a mean value prescribe for reference scene, season and solar angle. For applications that require a precise knowledge of actual SNR when SNR is limited to ten or several tens and could vary with local conditions, it could be computed along the orbit by radiative transfer computation
- Surfaces
  - Continental and maritime surfaces optical properties such as reflectance (or albedo)
  - Continental surfaces DEM
  - Continental subsurface components such as vegetation, canopy

Inputs:

- Lidar geometry: lidar pointing, range to surface
- Atmospheric composition, primary meteorological variables, surface properties and DEM

Outputs:

- Reference scene i.e. scattering and attenuating properties for atmosphere and surface according to the purpose of the lidar simulation

12.4. INSTRUMENT MODULE

Space borne Lidar makes use of a pulsed Laser emission for range discrimination and profiling. The Lidar optical signal is the convolution of the laser pulse power with the impulse response of the atmosphere and/or the surface (bare soils, canopy, …). The analog Lidar signal is the convolution of the Lidar optical signal with the impulse response of the detection unit. Then the digital Lidar signal is accumulated in a range bin using an analog-to-digital unit. At each steps noises are added to the useful Lidar signal that contains the relevant information.

This block represents the modeling of Lidar instrument and of the processing chain of Lidar signal. It provides the instrument non idealities.

As described in Section 8.3.1, the Instrument Module is composed of the following building blocks:

Building blocks:

- **Laser Transmitter Block:** generates the pulsed laser signal according to instrument characteristics and convolutes the pulsed laser signal with impulse response of scene to generate lidar signal. It includes the following building blocks: Transmitter (Tx) laser, Scene Convolution.
  - Inputs: scene data
  - Outputs: lidar scene
  - Transmitter (Tx) Laser. This building block describes the pulsed Laser characteristics that are important for the Lidar simulator. Various type of laser can be used with various Non Linear Optics processes to generate the wavelength of interest. At this stage for a phase A study, all the characteristics can be listed as input parameters as given below
    - One or multiple laser wavelength(s), laser linewidth(s), single mode emission (SLM) i.e. longitudinal and transverse (Gaussian beam shape) and spectral purity
    - Pulse separation in the case of multiple wavelengths
    - Laser energy per pulse
    - Pulse repetition frequency (depends on pulse energy to result in average power). It sets the horizontal sampling and number of independents available for a given horizontal resolution (i.e. 50 or 100 km) or time according to the platform velocity (7 km/sec)

- Laser pulse shape and duration
- Laser beam divergence, cross section, shape
- Laser polarisation (linear, circular, other), polarisation purity. Moist applications use linear polarisation with a high degree of purity (> 99%)

- **Spatial Block:** spatial resampling to the final ground sampling distance taking into account instrument MTF and spatial non-uniformity sources. Output is the scene within the footprint of the beam. It includes the following building blocks: Laser pointing/Receiver telescope, Rx/Tx co-alignment.
  - Inputs: lidar scene, instrument configuration parameters, other input parameters
  - Outputs: spatial resampled lidar scene
  - Receiver (Rx) Telescope. This building block describes the telescope characteristics. *It is common to active and passive optical instruments*. Various types can be used such as Cassegrainian or off-axis telescope. Off-axis configuration maximizes the collecting area by reducing the blocking effect due to the secondary mirror. For a phase A study, all the telescope characteristics can be listed as input parameters as given below. Primary diameter and collecting surface (A)
    - Primary diameter and collecting surface (A)
    - Field of view (FOV)
    - Collimated or focused at distant range. Most Lidar application use collimated FOV
    - Optical efficiency
  - TX/RX co-alignment. The laser and the telescope can be set side-by-side (in paraxial arrangement) or in coaxial arrangement. In both cases, the axis of emission and axis of observation are parallel or the same. The telescope field of view is larger than the laser beam divergence to accommodate inherent fluctuations in laser beam direction. The Tx/Rx alignment determines the overlap and so the optical efficiency. The telescope FOV is a compromise between full overlap for maximum collected signal and minimal unwanted background light.
- **Spectral Block:** spectral resampling from the finer working resolution of the scene generator to the spectral configuration of the sensor. It includes the following building blocks: Front-end optics.
  - Inputs: spatial resampled lidar scene
  - Outputs: at-sensor scene
  - Front-end optics unit. Starting from the focal plane, this building block describes an optical system to reshape the collected light in a parallel beam to be distributed in various channels using dichroic beam splitters and polarisation elements, and rejection/filtering of unwanted background radiation
    - Dichroic beam splitters for multiple wavelengths
    - Polarisation element to separate parallel and perpendicular polarisations
    - Broadband and narrowband filters to reject background light
- **Radiometric Block:** transforms at-sensor scene to digital counts. It includes the following building blocks: Lidar Detection (Dx) unit.
  - Inputs: at-sensor scene
  - Outputs: digital counts and instrument noise
  - Detection (Dx) unit. This building block describes the photo detector characteristics and front end electronic and circuitry. Various types of photo detectors are used depending on wavelength: Photomultiplier Tube (PMT) in UV and visible spectral range, Avalanche Photodiode in near IR, CCD matrix in UV and visible
    - Quantum yield (<1). It is essential to maximize the Lidar efficiency and the best use of scattered laser photons

- Intrinsic noise (dark current, Johnson noise), NEP
- Excess noise (factor for APD)
- Gain (or amplification)

- **Data pre-processing Block:** transforms the signal captured by the sensor taking into account the Analog-to-Digital (A/D) conversion. It includes the following building blocks: Signal recorder and onboard payload computer
  - Inputs: digital counts and instrument noise
  - Outputs: raw and ancillary data
  - Analogue-to-Digital Converter (ADC) or photon counting unit. This building block describes the conversion of electric signals or number of photons in digital signals for further numerical processing. The ADC technique is implemented when the photo electrons flow after detection is sufficiently large to generate electric current. On the contrary Photon counting technique is used when photo-electron generation is limited o it enables successve distinct single count.
    - ADC main characteristics are the sampling frequency and the digitization capability given in number of bits. Today, there is no practical limitation on both.
    - Photon counting main characteristic is the sampling time that sets the range resolution

12.5. LEVEL-1 PROCESSING MODULE

Active optical instruments have commonality with passive optical instruments, see Figure 9-21 and 9-24. This block handles the generation of the Level-1 products through ingestion of atmospheric scene and of the instrument signal. This module ingests the lidar signal and non idealities coming from the instrument and from the processing chain, to generate a realistic Level-1 Lidar product. The input errors to be injected are generated by the Instrument simulator. Besides the instrument errors, the product generation module is in charge of applying the "blurring" to the ideal Lidar signal.

Inputs:

- Signal from instrument simulator
- Instrument performances from the System errors simulator

Outputs:

- Level-1 products

Building blocks:

Due to the particularities of the Lidar processing the Level-1 product generation modules are mission-specific and no lower-level building blocks have been identified. There are, however, different implementations of this module depending on the specific application, as shown in the next sections.

12.5.1. SIMPLE ELASTIC BACKSCATTER

This is for example, the case of CALIOP and ICESat-1 atmospheric channel. Table 12-7 shows the different Level-1 products and the definition of their basic processing.

Table 12-2. Level-1 Products for Simple Elastic Backscatter

Level	Products
1a	- Level 0 data at 532 nm parallel and perpendicular, and 1064 nm with all calibrations computed and appended, but not applied. - Annotation from time, satellite position, velocity and pointing
1b	- Calibrated Lidar attenuated backscatter profiles at 532 nm parallel - Calibrated Lidar attenuated backscatter profiles at 532 nm perpendicular using calibration made on ground - Auxiliary information: meteorological variables (p, T) from NWP model - Synthetic molecular signals computed using p and T information - Assumption of clean zone free from aerosols and clouds to match synthetic molecular signals in order to determine the instrumental factor

	- Cross calibration of 1064 nm signals on cirrus clouds using signals at 532 nm and assuming the same backscattering coefficients

12.5.2. ELASTIC HIGH SPECTRAL RESOLUTION

This is for example, the case of EarthCare. Please note that research activities are currently ongoing. Table 12-3shows the different Level-1 products and the definition of their basic processing.

Table 12-3. Level-1 Products for Elastic High Spectral Resolution

Level	Products
1a	- Level 0 data at 355 nm with all calibrations computed and appended, but not applied. - Annotation from time, satellite position, velocity and pointing
1b	- Calibrated Lidar attenuated backscatter profiles at 355 nm - Auxiliary information: meteorological variables (p, T) from NWP model - Synthetic molecular signals computed using p and T information - Assumption of clean zone free from aerosols and clouds to match synthetic molecular signals in order to determine the instrumental factor

12.5.3. ELASTIC HSR DOPPLER

This is for example, the case of ADM-AEOLUS. Please note that research activities are currently ongoing.

The measurement principle is presented on Figure 11-6. The left panel displays the Lidar pointing configuration across the orbit track for continuous mode operation. The right panel displays the scattered lidar spectra made of a narrow line spectrum from particles and a broad line spectrum from molecules. The Doppler frequency shift is displays as dashed spectra. The dual Fabry-Perot interferometer analyses the molecular signal rejected by the Fizeau interferometer into two filters A and B. The signals are used to retrieve the particle extinction and backscatter coefficients and the wind velocity component along the Lidar LOS

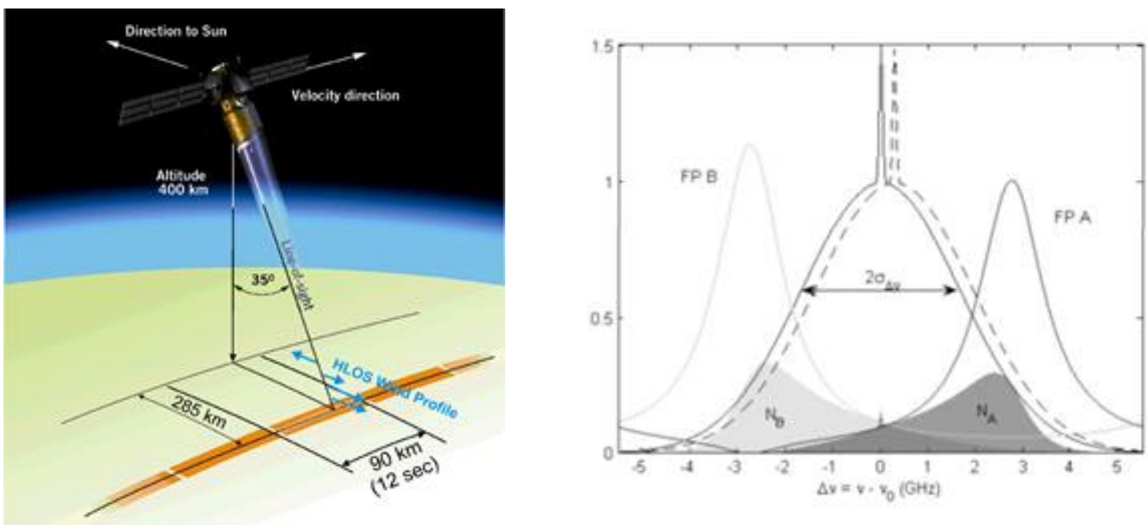


Figure 12-5. Measurement principle of ADM-AEOLUS [RD. 117]

Table 12-4 shows the different Level-1 products and the definition of their basic processing.

Table 12-4. Level-1 Products for Elastic HSR Doppler [RD. 117]

Level	Products

1a	- Level 0 data at 355 nm with all calibrations computed and appended, but not applied. - Annotation from time, satellite position, velocity and pointing
1b	- Signal strength of Rayleigh (molecular) and Mie (particles) profiles in 25 bins (or pre selected range gates) - Scattering ratio from Mie channel - Instrument corrected Horizontal LOS

12.5.4. RANGE RESOLVED DIFFERENTIAL ABSORPTION LIDAR

This is for example, the case of WALES. Table 12-5 shows the different Level-1 products and the definition of their basic processing.

Table 12-5. Level-1 Products for Differential Absorption Lidar [RD. 118]

Level	Products
1a	- Level 0 data at **4 wavelengths** around 935 nm (3 wavelengths for DAODs and one for reference) with all calibrations computed and appended, but not applied. - Annotation from time, satellite position, velocity and pointing
1b	- Calibrated lidar attenuated backscatter profiles at the 4 wavelengths - Auxiliary information: meteorological variables (p, T) from NWP model - Synthetic molecular signals computed using p and T information - Assumption of clean zone free from aerosols and clouds to match synthetic molecular signals in order to determine the instrumental factor

12.5.5. INTEGRATED PATH DIFFERENTIAL ABSORPTION LIDAR

This is for example, the case of A-SCOPE and MERLIN.

The measurement principle is presented on Figure 12-6. It requires 2 wavelengths for absorption and reference as shown on CO_2 line displays on bottom right. The laser pulses are transmitted within 200 µs and it requires a good overlapping of the 2 beams on ground to minimize potential bias.

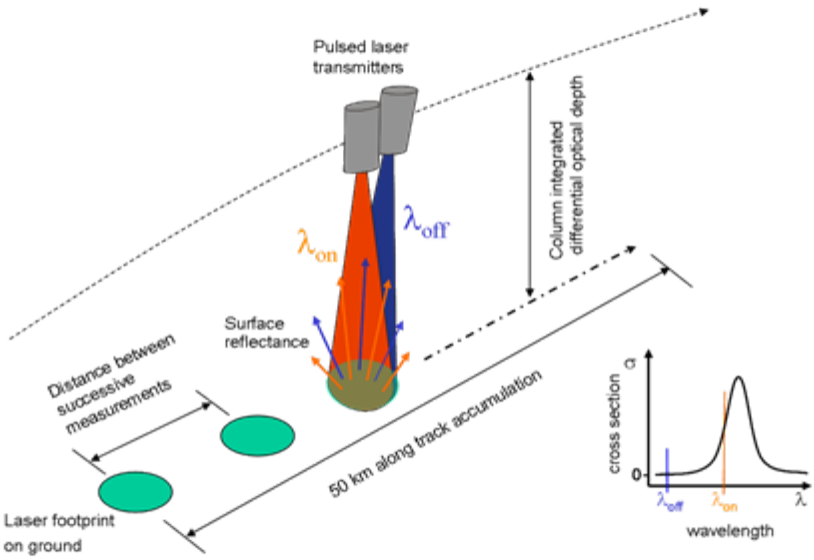


Figure 12-6. Basis principle measurement of IPDA method for CO_2.

Table 12-6 shows the different Level-1 products and the definition of their basic processing.

Table 12-6. Level-1 Products for Integrated Path Differential Absorption Lidar [RD. 119]

Level	Products
1a	Level 0 data at 2 wavelengths used to probe the absorption line of the trace specie of interest (CO2, CH4) and for reference with all calibrations computed and appended, but not applied. Annotation from time, satellite position, velocity and pointing
1b	Differential Absorption Optical Depth (DAOD) using Lidar signals scattered from surface (hard target) Scattering Surface Elevation (SSE) based on Lidar ranging measurements. The dry mixing ratio retrieval needs measurement accuracy better than 10 m

12.6. LEVEL-2 RETRIEVAL MODULE

The Level-2 Module adapts to the generic architecture presented in Section 8.5. This block inverts the forward module to retrieve the atmospheric variables and surface properties of interest from the Level-1 lidar product. Therefore, it implements a procedure that should ideally invert the one performed by the scene generation module to retrieve the physical quantity (Level-2 product).

Building blocks:

Inputs:

- Lidar geometry: lidar slant range
- Level-1 data

Outputs:

- Level-2 product

Building blocks:

The Level-2 Retrieval Algorithms are mission-specific. In addition, given the Level-1 products, different Level-2 retrieval algorithms can be used in order to obtained the same Level-2 product. There are, however, different implementations of the Retrieval Algorithms Block depending on the specific application.

12.6.1. SIMPLE ELASTIC BACKSCATTER

This is for example, the case of CALIOP and ICESat-1 atmospheric channel. Table 12-7 shows the different Level-2 products and the definition of their basic processing.

Table 12-7. Level-2 Products for Simple Elastic Backscatter

Level	Products
2	- Particle extinction coefficient using iterative solution of the Lidar equation and lidar ratio as proposed by Elterman 1966 or Gambling and Bartusek 1972 - Depolarization ratio for particle layers - Scattering ratio - Scene classification using

12.6.2. ELASTIC HIGH SPECTRAL RESOLUTION

This is for example, the case of EarthCare. Please note that research activities are currently ongoing. Table 12-7 shows the different Level-2 products and the definition of their basic processing.

Table 12-8. Level-2 Products for Elastic High Spectral Resolution

Level	Products
2a	- Particle extinction coefficient at 355 nm using Rayleigh channel - Depolarization ratio for particle layers at 355 nm using Mie channel - Scattering ratio at 355 nm

	- Ice water content (IWC)
2b	Synergism between the 3 instruments on the same platform ATLID, CPR and MWI

12.6.3. ELASTIC HSR DOPPLER

This is for example, the case of ADM-AEOLUS. Please note that research activities are currently on-going. Table 12-9 shows the different Level-2 products and the definition of their basic processing.

Table 12-9. Level-2 Products for Elastic HSR Doppler [RD. 117]

Level	Products
2a	- Cloud and aerosols products from Rayleigh and Mie channels and combining the two channels: backscatter, optical depth, extinction-to-backscatter ratio, Classification
2b	- Wind products: Rayleigh and Mie channels HLOS

12.6.4. RANGE RESOLVED DIFFERENTIAL ABSORPTION LIDAR

This is for example, the case of WALES. Table 12-10 shows the different Level-2 products and the definition of their basic processing.

Table 12-10. Level-2 Products for Differential Absorption Lidar [RD. 118]

Level	Products
2	- Water vapor densities at 3 probing wavelengths - Auxiliary information: 1) spectroscopy of the 3 water vapor absorption lines of interest and 2) meteorological variables (p, T) from NWP model - Spin-off products: aerosols backscatter profiles, Cloud tops and bases, PBL height, Surface reflectance

12.6.5. INTEGRATED PATH DIFFERENTIAL ABSORPTION LIDAR

This is for example, the case of A-SCOPE and MERLIN. Table 12-11 shows the different Level-2 products and the definition of their basic processing.

Table 12-11. Level-2 Products for Integrated Path Differential Absorption Lidar [RD. 119]

Level	Products
2	- Weighting function WF(p) as a function of atmospheric pressure p - Dry mixing ratio XCO2 or XCH4 - Spin-off products: Canopy, Surface reflectance, clouds top

12.7. PERFORMANCE EVALUATION MODULE

In each cases, the evaluation performance module (see Figure 12-1 and Figure 12-2) compare the value of the variable of interest that is used as input to the Lidar signals simulator to the output of Lidar signals and data processor. It is a one-to-one direct comparison of profiles or dry mixing ratio. The PEM will follow the generic architecture presented in Section 8.6.

13. CONCLUSIONS

An extensive review of Earth Observation missions and their instruments has allowed deriving a Reference Architecture for end-to-end mission performance simulators. A generic Reference Architecture, based on six high-level modules, has been defined and it suits most mission concepts and instruments. However it does offer the flexibility of being adapted for different mission particularities, for example by adding additional high-level modules, but always among those defined in the Reference Architecture.

The use of this Reference Architecture for the development of new simulators has the potential of reducing the reengineering process associated to the evolution of the simulator throughout the different mission phases. Moreover, the identification of common elements for different types of instruments also enables reuse of the architectural elements across several mission simulators.

In addition, the requirements for an E2E simulator based on the Reference Architecture Concept have been derived, as well as the requirements for a framework and repository of models.

14. ANNEX: REFERENCE ARCHITECTURE FOR GNSS-R INSTRUMENTS

14.1. INTRODUCTION AND SCOPE

Earth observation by means of GNSS-reflection (GNSS-R) is based on the use of reflected signals emitted from navigation satellites. Since it is a very specific application and the development of an E2E simulator falls outside of the generic philosophy of this activity, it had not been considered as a case study. However, following ESA's expression of interest during the ARCHEO-E2E Progress Meeting, the Universitat Politècnica de Catalunya (UPC) has kindly agreed to make a complete assessment of GNSS-R systems and how a reference architecture could be proposed for these systems.

14.2. PROPOSED ARCHITECTURE

The End-to-End simulator covers the complete simulation of the environmental conditions seen by the GNSS-R instrument, the full instrument modeling and the determination of the retrieved relevant data measured by the instrument. The comparison with the "real" world conditions previously simulated will allow the evaluation of the instrument performances.

GNSS-R is a relatively new technology, and the Earth observation using GNSS-R is in start-up state. Currently, GNSS-R technology is still being studied including the observation model, instrument effects, calibration techniques, and geophysical parameter retrieval algorithms. Moreover, there is no heritage space mission of GNSS-R except the feasibility test mission of UK-DMC [RD. 121],[RD. 122]. Therefore, the implementation of GNSS-R E2E simulator is somehow premature. Especially, the lack of a validated model makes the simulation difficult.

In this project, it is proposed to use the mostly used observation model proposed in [RD. 123]:

$$\langle |Y(\tau, f_d)|^2 \rangle = \frac{P_T \lambda^2 T_i^2}{(4\pi)^3} \int_{A_s} \frac{G_T(\vec{\rho}) G_R(\vec{\rho}) |\chi(\Delta\tau, \Delta f_d)|^2}{R_T^2(\vec{\rho}) R_R^2(\vec{\rho})} \sigma^0(\vec{\rho}) d^2\vec{\rho} \tag{1}$$

where P_T is the transmitter power, G_T is the transmitter antenna gain, G_R is the receiver antenna gain, R_T and R_R are the distances between the scattering point and the transmitter and the receiver, respectively; T_i denotes the coherent integration time, λ is the wavelength of the signal, $\chi(\Delta\tau, \Delta f_d)$ stands for the Woodward Ambiguity Function (WAF) (correlation characteristics of GNSS PRN code), and σ^0 is the scattering coefficient discussed in **MS2**. The delay offset and Doppler shift associated the surface point $\vec{\rho}$ are represented by $\tau(\vec{\rho})$ and $f_d(\vec{\rho})$, respectively; $\Delta\tau = \tau - \tau(\vec{\rho})$ and $\Delta f_d = f_d - f_d(\vec{\rho})$.

Although the model (1) has been widely used for GNSS-R observation, it does not include the some effects such as the receiver, thermal noise, etc., which must be taken into account the simulation. Therefore, in this document the reference architecture for GNSS-R instruments contains all the elements affecting the instrument output as many as possible, although their analytical models have not been concretely validated.

The simulator should cover the implementation of the:

- The geometry module including the platform's motion simulation (position and attitude).
- The scene module that simulates the "real" world sensed by the instrument system including the scattering coefficients, the direct / reflected delay (including atmospheric delay), and the Doppler frequency of the observed Earth surface.
- The instrument module containing:
  - The physical model of the antenna,
  - The receiver models (including type of GNSS-R), and
  - The calibration models, aimed to implement a way to obtain measurements to perform the error correction.
- The L0-processor module to generate the instrument's observables, that is the instrument outputs as a numerical value ("digital counts" or "digital number").
- The L1-processor module to generate time/geo-referenced 'Calibrated DDM'
- The L2-processor module, including the retracking, to retrieve the geophysical parameters from the DDM (L1b or L1c data) and then allows the user to compare with the real world geophysical parameters.

These functionalities are graphically represented in Figure 14-1:

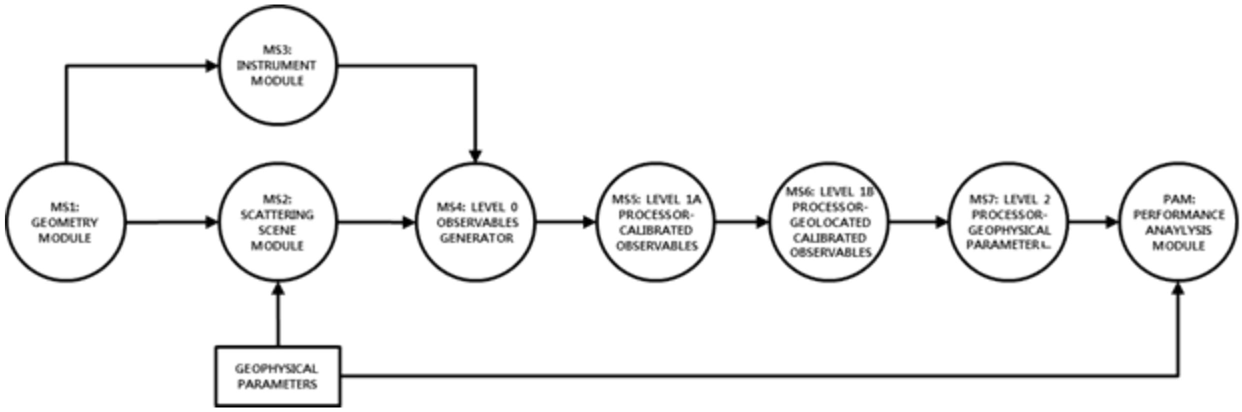


Figure 14-1. Structure and building blocks of the Mission Simulation Modules (MS) and the Performance Evaluation Module (PEM)

As shown functions in Figure 14-1, it is clear the fulfillment with the End-to-End concept: the simulator user is able to define, re-define, or modify any parameter that could influence the mission results.

The notable feature of the GNSS-R simulator is on the Performance Evaluation Module (PEM). The GNSS-R does not image the scattering scene on the spatial domain. The output is DDM on the delay-Doppler shift domain, and it is impossible to performance comparison at L1 (calibrated geo/time referenced DDM) because there is no input DDM (only input scattering scene). Therefore, the PAM should be performed at L2 by comparing the retrieved geophysical parameter (sea surface height, wind speed, etc.) and the input geophysical parameters.

The list below is an ordered list of components ordered top-down that describes the hierarchical relationships between the different modules to be an input to the architecture definition.

Table 1. Identification of Components of a GNSS-R E2E Performance Simulator

- **MD: MISSION DEFINITION**
  - MD1. SATELLITE RELATED DEFINITION
    - MD1.1: ORBIT DEFINITION
    - MD1.2: PLATFORM DEFINITION
    - MD1.3: ATTITUTDE DEFINITION
  - MD2. INSTRUMENT DEFINITION
    - MD2.1: ANTENNA DEFINITION
    - MD2.2: RECEIVER DEFINITION
    - MD2.3: CORRELATOR/ADC DEFINITION
    - MD2.4: CALIBRATION DEFINITION
  - MD3. SIMULATION DEFINITION
  - MD4. SCENARIO DEFINITION
- **MS: MISSION SIMULATION**
  - MS1: GEOMETRY MODULE
    - MS1.1: ORBIT COMPUTATION
    - MS1.2: PLATFORM COMPUTATION
    - MS1.3: ATTITUDE COMPUTATION
    - MS1.4: FOV COMPUTATION
    - MS1.5: GEOMETRY PARAMETERS COMPUATION
  - MS2: SCATTERING SCENE MODULE
    - MS2.1: COASTLINES DEFINITION
    - MS2.2: DISCRIMINATOR
    - MS2.3: LAND
    - MS2.4: OCEAN
    - MS2.5: ICE
    - MS2.6: ATMOSPHERIC CONTRIBUTION
    - MS2.7: ADDING CONTRIBUTION
  - MS3: INSTRUMENT MODULE
    - MS3.1: ANTENNA MODULE
      - MS3.1.1: ARRAY CONFIGURATION
      - MS3.1.2: ANTENNA POINTING (DYNAMIC)
      - MS3.1.3: ANTENNA LOSSES
      - MS3.1.4: ARRAY FACTOR COMPUTATION
    - MS3.2: RECEIVER MODULE
      - MS3.2.1: ARCHITECTURE DEFINITION
      - MS3.2.2: FREQUENCY RESPONSE COMPUTATION
      - MS3.2.3: ADC/CORRELATOR RESPONSE COMPUTATION
  - MS4: LEVEL 0 - OBSERVABLES GENERATOR
    - MS4.1: WAF GENERATOR
    - MS4.2: IDEAL DDM CALCULATOR
    - MS4.3: NOISE GENERATOR
    - MS4.4: LEVEL 0 DATA CALCULATOR
  - MS5: LEVEL 1A PROCESSOR – CALIBRATED INSTRUMENT OBSERVABLES
    - MS5.1: DETERMINATION OF INTERNAL CALIBRATION PARAMETERS
    - MS5.2: DETERMINATION OF EXTERNAL CALIBRATION PARAMETERS
    - MS5.3: ERROR COMPENSATION MODULE
  - MS6: LEVEL 1B PROCESSOR – GEOLOCATED CALIBRATED DDM
    - MS6.1: GEOLOCALIZATION
  - MS7: LEVEL 2 PROCESSOR – GEOPHYSICAL PARAMETERS RETRIEVAL ALGORITHM
    - MS7.1: TARGET SEPARATION
    - MS7.2: LAND RETRIEVAL ALGORTIHMS
    - MS7.3: OCEAN RETRIEVAL ALGORTIHMS
    - MS7.4: ICE RETRIEVAL ALGORTIHMS
    - MS7.5: ATMOSPHERIC CORRECTION
    - MS7.6: RESAMPLING TO STANDAR GRID
- **PEM: PERFORMANCE EVALUATION MODULE**
  - PEM1: PLATFORM PAM
  - PEM2: INSTRUMENT PAM
  - PEM3: GEOPHYSICAL PARAMETERS PAM
  - PEM5: TABULAR AND GRAPHICAL REPRESENTATION

14.3. MD: MISSION DEFINITION

Before starting a mission simulation, it is necessary to define the mission. This includes the definition of the input parameters (variables affecting the definition of the instrument, and environment), and the definition of the control parameters (variables affecting the way the simulation is performed). The **Mission Definition** (**MD**) module must include the parameters related to the satellite, the instrument, the environmental parameters, and finally the definition of the scenario (Figure 14-2).

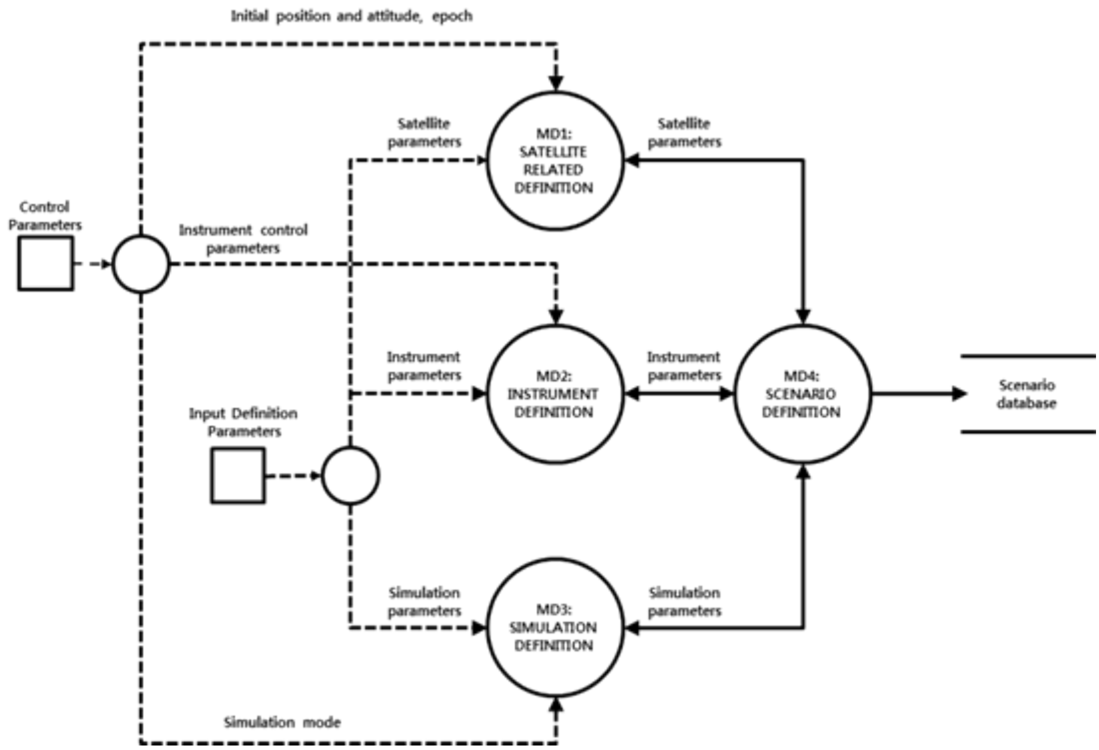


Figure 14-2. Structure and building blocks of the Mission Definition Module (MD)

From Figure 14-2 to Figure 14-4 present the Mission, Satellite and Instrument Definition modules and their interaction, according to the above table. The Satellite Definition Module (**MD1**, Figure 14-3) includes the definition of orbit, platform, and instrument's attitude definition.

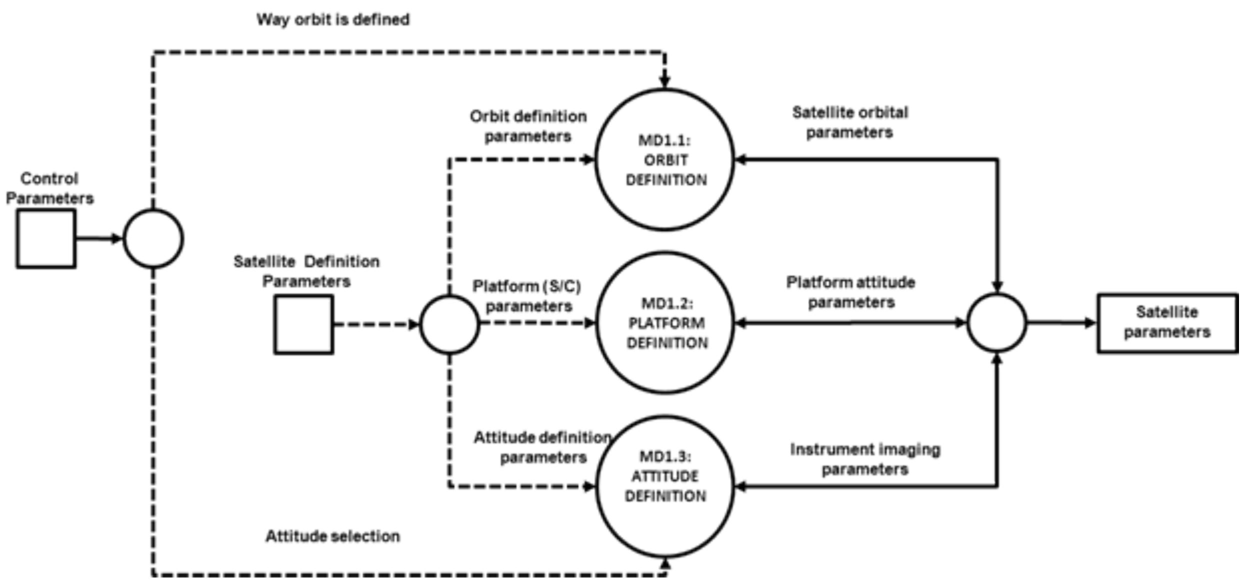


Figure 14-3. Structure and building blocks of the Satellite Definition Module (MD1)

The Instrument's Definition Module (**MD2**, Figure 14-4) includes the definition of the antenna(s), receiver(s), correlator / analog–to-digital converter (ADC) used to discretize the analog samples of the detected voltages, and the calibration.

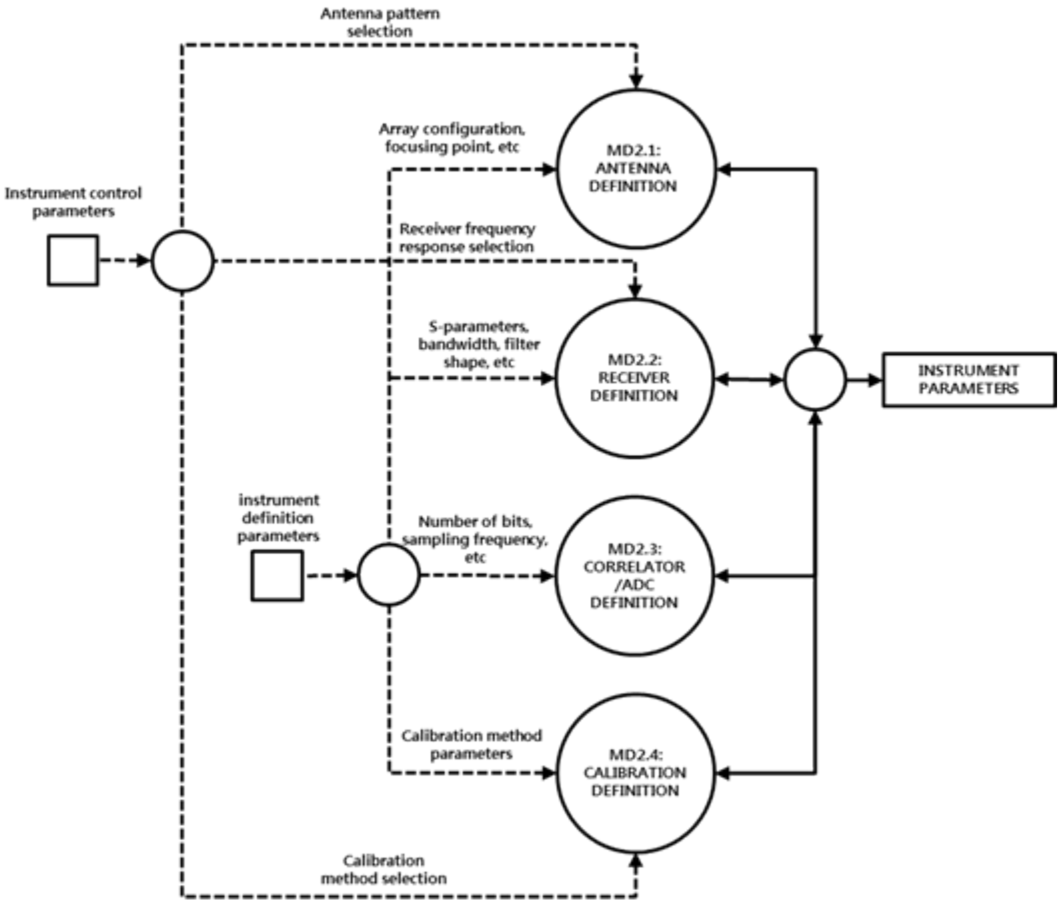


Figure 14-4. Structure and building blocks of the Instrument Definition Module (MD2)

Additionally, the control parameters determine the type of simulation to be performed (MD3). Three different types of simulations are proposed:

- **Snapshot mode**: to assess the instrument's response to a single input scattered scene,
- **Monte-Carlo mode**: to assess the instrument response to a series of snap-shots in which the input scattered scene remains constant, but the instrument parameters are randomly varied either assuming a Gaussian distribution (characterized by its mean and standard deviation), or an arbitrary probability density function (pdf). In this mode, the user shall have the possibility to select which instrument parameters are fixed and which ones are allowed to vary according to its configured law. In this way, the error budget analysis can be tailored to specific parameters or groups of them, as well as covering the whole instrument if needed.
- **Time-evolution mode**: to predict the instrument response to a series of selectable snapshots (typically consecutive snapshots), including the time-dependent variable input scattered scene and observation geometry, and/or instrument aging and thermal drifts (antennas and receivers physical temperatures, gain and noise, offsets drifts) derived either from a mathematical function or read from an input data file. This mode will be the normal one when trying to simulate the mission.

In the following sections the Mission Simulation (MS) and the Performance Analysis Modules (PAM) will be explained in more detail.

14.4. MS1: GEOMETRY MODULE

The Geometry Calculation Module (Figure 14-5) provides the observation geometry which is determined by the receiver (S/C), the transmitter (GNSS), and the scattered points on the Earth. Also it depends on the satellite orbital position, the platform attitude, and the instrument attitude (antenna pointing as a function of time), and the epoch.

A specific situation for GNSS-R is that it requires the satellite states (position and velocity) for GNSS (e.g. GPS, Galileo) satellites as well as the receiver (S/C) satellite. So, in this module, the GNSS satellites state must be calculated according to their orbit and epoch. Moreover, the field of view (FOV) of GNSS-R, so-called glistening zone, is defined differently from other imaging sensor. i.e., it is defined not only by the instrument attitude and antenna pattern, but also by the satellites' positions and velocities.

This module should have the following building blocks: **MS1.1**: Orbit computation (satellites' positions and velocities), **MS1.2**: Platform computation (platform's attitude), **MS1.3**: Attitude computation (instrument's attitude with respect to the platform), and **MS1.4**: FOV computation.

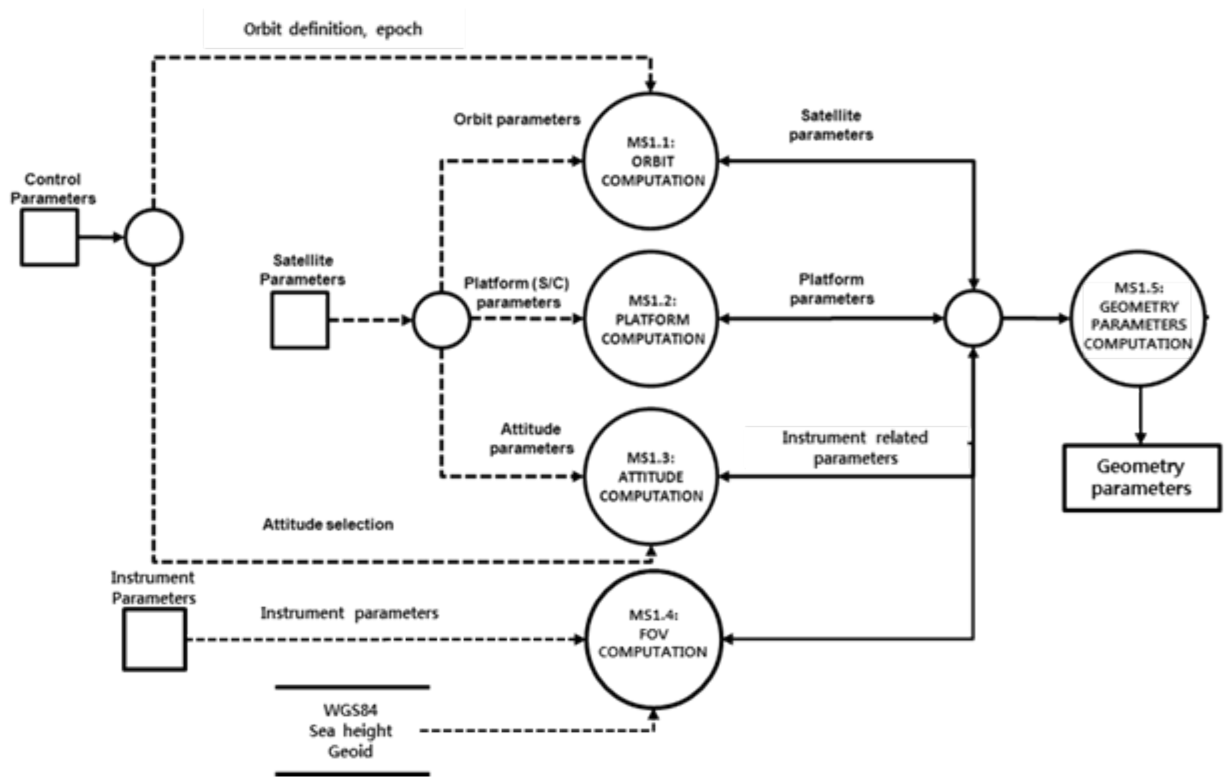


Figure 14-5. Structure and building blocks of the Geometry Calculation Module (MS1)

From the instrument point of view, the FOV of the GNSS-R (glistening zone) is defined by the DDM domain ranges (delay ranges, and Doppler frequency ranges) which are the instrument parameters. **MS1.4**: FOV computation module computes the specular reflection point on the Earth where the incidence angle equals to the reflection angle, and this module finds the outermost iso-delay line which is the FOV boundary.

The information computed in these blocks is then used in the module **MS1.5**: Geometry Parameters Computation the reflected path delay, the Doppler frequency, the incidence and reflected angle of the scattered points on the Earth, and they will be used later by the scattering scene module (**MS2**). Additionally, the platform's and instrument's attitude will be used in the instrument module (**MS3**) to calculate the antenna pattern term of the scattered points. This module should also provide the telemetry data (platform position and attitude, instrument attitude, and time stamp) to be used for geo-localization purposes in the Level 1B processor.

14.5. MS2: SCATTERING SCENE MODULE

14.5.1. INTRODUCTION

As for the other sensors types, the Scene Generator, in the case of GNSS-R the "Scattering Scene Generation Module" is one of the major constituents of a mission simulator. The generated scenes should be simulated according to the mission objectives and the instrument characteristics (GNSS signal frequency, polarizations15, etc.). The spatial resolution must be much higher than that of the instrument to be simulated. Except for synthetic scenes, or local imagery, this is often difficult to achieve, especially for simulators that include the whole Earth, as it is usually the case in passive microwave EO missions. In many cases, the new instrument to be studied has a better spatial, spectral resolution than the ones of existing instruments, therefore there is no previous data at enough resolution to be used as an input. In order to satisfy the Nyquist sampling theorem, a minimum factor of 2 is required; for GNSS-R, the factor of 2 in the spatial area corresponding to the DDM resolution (delay and Doppler bin resolutions).

In the GNSS-R simulation, the sea scattering scene is normally generated using the KA-GO (Kirchoff Approximation using Geometric Optics) [RD. 123]:

$$\sigma^o = \frac{\pi|\Re|^2 q^4}{q_z^4} P\left(\frac{\vec{q_\perp}}{q_z}\right) \qquad (2)$$

where $\sigma^0(lat,lon)$ is the scattering coefficient, $\Re(lat,lon)$ is the Fresnel reflection coefficient, $\overrightarrow{q(lat,lon)}$ is the scattering vector, and $P\left(\frac{\vec{q_\perp}}{q_z}\right)$ is sea surface slope probability density function. In order to calculate the scattering scene, the geophysical data of the scattering point (lat,lon) are required, for example, sea salinity, temperature, surface wind speed and direction.

For the land, the GNSS reflection is usually considered as quasi specular reflection, and the power reflectivity is modeled as [RD. 124]:

$$\Gamma_{rl} = |\Re|^2 \exp(-h\cos^2\theta)$$
$$h = 4k^2\sigma^2 \qquad (3)$$

where h is called a roughness parameter, k is the wavenumber, and σ is standard deviation of the surface.

Additionally, the cryosphere scattering can be modeled. Although the particular model is still being studied, the feasibility of crysophere monitoring using GNSS-R has been conducted [RD. 125]-[RD. 128]. Therefore, the cryosphere scattering scene generation module is employed for simulator.

In the scattering scene module, additional environmental effects are also computed, e.g., the atmospheric effect. The ionosphere and troposphere affect the signal delay and consequently the instrument output DDM. Therefore, these effects should be simulated.

The three category of scenes are user-generated defined that can be used to test the instrument performance, such as the geophysical parameters retrieval algorithms.

14.5.2. DESCRIPTION OF BUILDING BLOCKS

According to the above-described different possible scenes, the Scattering Scene Module will consist of the building blocks shown in Figure 14-6.

[15] Emitted right hand circular polarization changes to the left hand circular polarization after reflection.

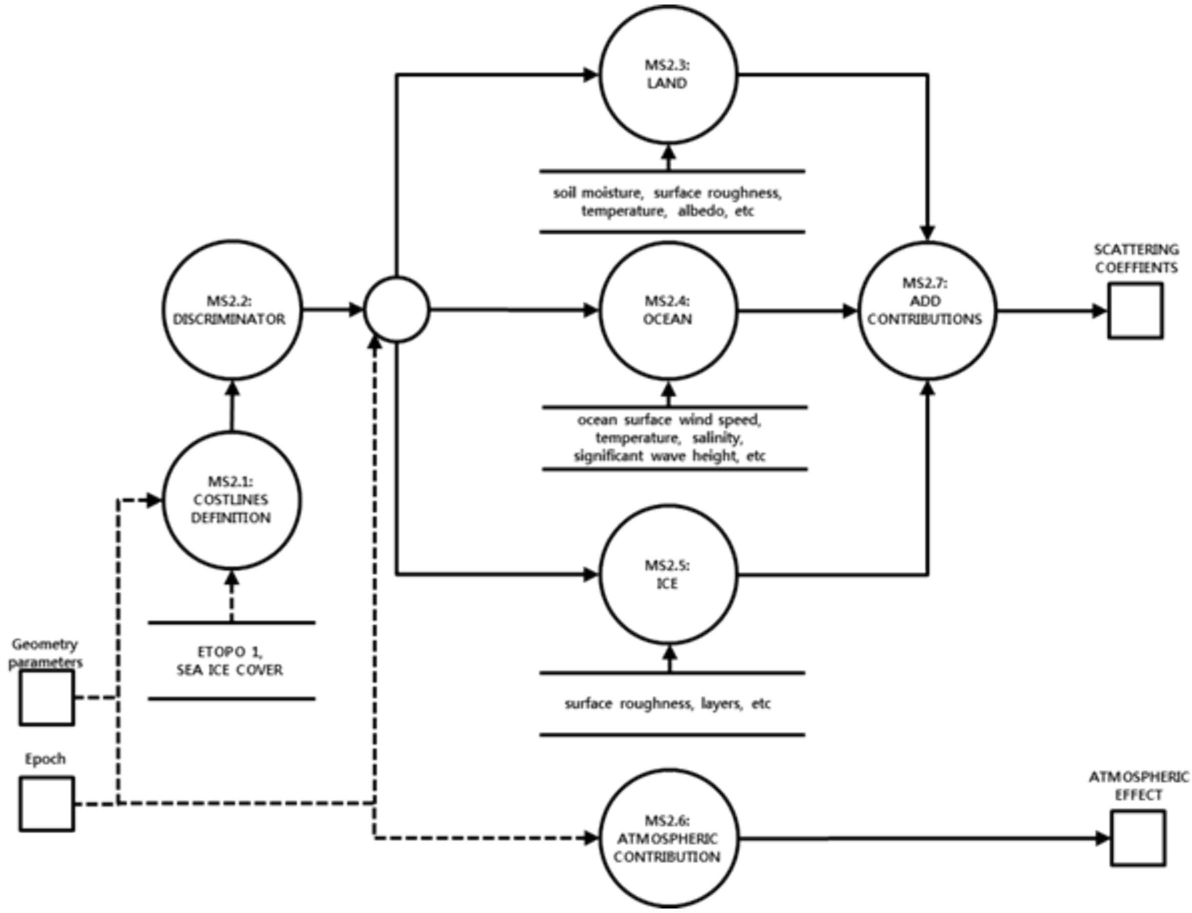


Figure 14-6. Structure and building blocks of the Scattering Scene Module (MS2)

The Scattering Scene Module starts with the Geometry Parameters that provide the location of the FOV (e.g. latitude and longitude of all scattering points), and local incidence /reflection angle required to compute the scattering coefficients. The second step consists of separating the different surface types: land (**MS2.3**), ocean (**MS2.4**), and ice (**MS.2.5**).

Land modeling requires a number of parameters, including the surface's temperature, moisture, roughness, topography, vegetation, albedo, etc. The fraction of the footprint covered by inland water bodies, urban areas, bare and vegetation-covered soils, snow etc. must be a priori known. Coast lines and topography can be obtained. This Digital Elevation Model (DEM) can also be used for the topographic height estimation.

The modeling of the ocean scattering (**MS.2.4**) is relatively well reported, it is usually parameterized in terms of the sea surface salinity, temperature, and the wind speed and direction, as shown in (3).

Additionally, the atmospheric contribution (**MS.2.6**) computes the ionospheric/tropospheric delay and attenuation, which will be applied to L0 observable generator (**MS4**)

14.6. MS3: INSTRUMENT MODULE

14.6.1. INTRODUCTION

GNSS-R instruments are defined by:

- **Instrument type**: Conventional GNSS-R (correlates the reflected signal with the locally generated code replica), and interferometric GNSS-R (correlates the reflected signal with the directly received signal).
- **Antennas:** number of antenna beams, and pointing (focusing) direction (static + dynamic, computed in **MS3**), antenna pattern.

- **Receivers:** number of frequency bands (e.g. GPS L1 and L5), center frequencies and bandwidths (including the frequency response), filter shape, etc.
- **ADC/Correlator**: number of delay and Doppler bins for DDM output, resolution of delay bin (waveform sampling frequency), resolution of Doppler bin, number of bit, etc.

The Instrument module (**MS3**) considers the spectral and spatial sampling of the instrument itself, including all the instrument errors. According to the definition of instrument type, it is executed either for a single chain (only down-looking for the conventional GNSS-R) or for a two chains (up- and down-looking for the interferometric GNSS-R).

It is convenient to separate the instrument module in two blocks: the antenna module (**MS3.1**) and the receiver module including ADC/Correlator (**MS3.2**), as sketched in Figure 14-7 below. These blocks are explained in detail in the following section.

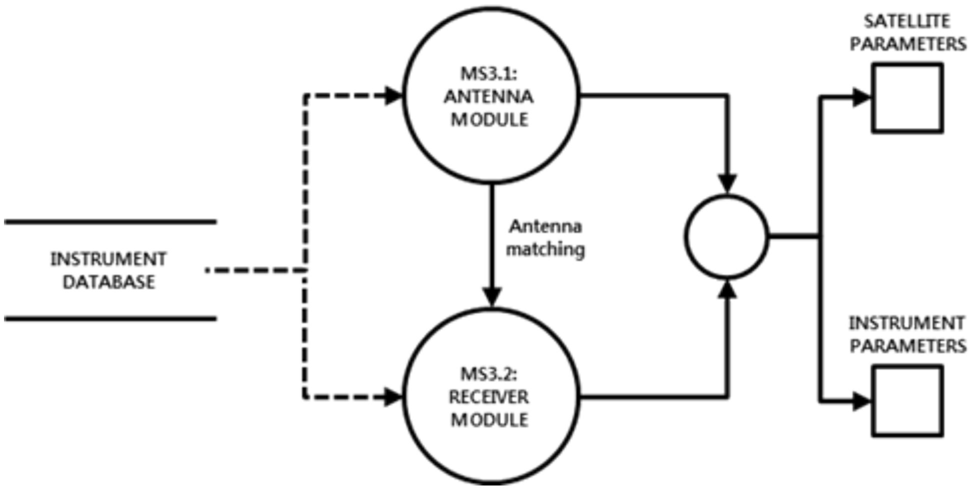


Figure 14-7. Structure and building blocks of the Instrument Module (MS3)

14.6.2. DESCRIPTION OF BUILDING BLOCKS

14.6.2.1. MS3.1: Antenna Module

The Antenna modeling block (**MS3.1**) uses the satellite-related geometric parameters computed in **MS1.5**, e.g. platform and instrument attitudes, and the user-defined instrument parameters. Normally GNSS-R employs the array antenna system in order to increase the antenna gain, generate / steer the multi-beams. Therefore, the antenna module is basically designed for the case of an array. In the case of non-array system, the proper mission definition is required, i.e., the input of single elementary antenna. The antenna module finally computes the beam pattern for all the directions of the scattering points in FOV.

The **MS3.1** module is composed of four blocks:

- **MS3.1.1** that computes the static antenna positions according to the array configuration input.
- **MS3.1.2** that computes the dynamic antenna pointing, i.e., focusing direction or bore sight of array antenna (changes in time with respect to the platform). It computes the focusing direction as many as the number of beams (instrument parameter).
- **MS3.1.3** that computes the antenna losses, e.g., ohmic losses and matching. This information is required to evaluate the thermal noise introduced by the antenna losses.

And

- **MS3.1.4** that computes the array antenna patterns (array factor) for each beam corresponding to the polarization. In the case the conventional GNSS-R, it computes the array factor of only down-looking antenna (LHCP). The other case, the interferometric GNSS-R, it computes both the up-looking (RHCP) and the down-looking (LHCP) antennas. The final outputs of the module are the antenna pattern values at all the scattering points in the FOV, to be used in L0 observables generator (**MS4**)

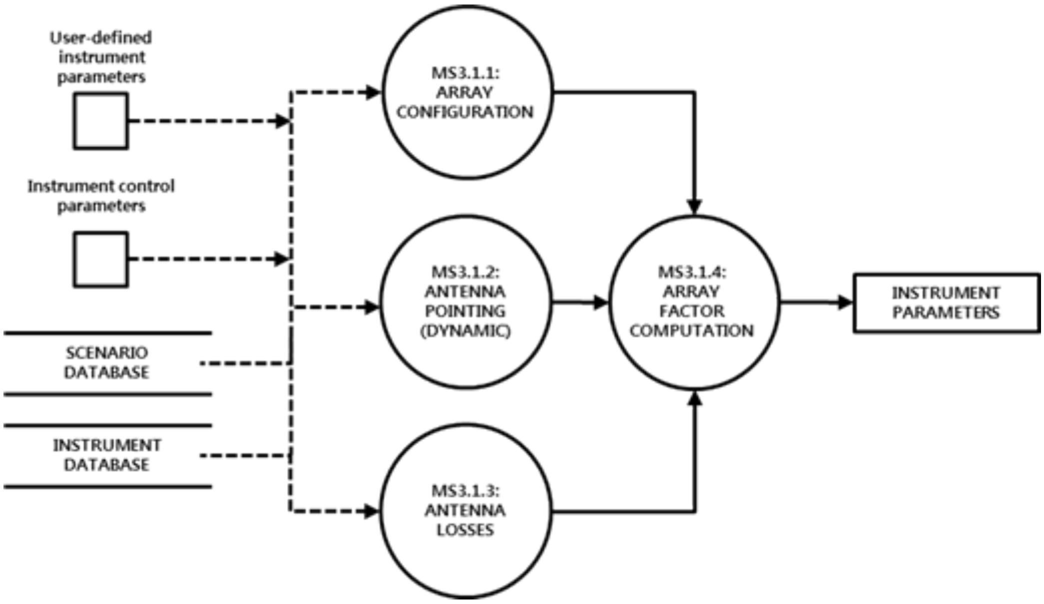


Figure 14-8. Structure and building blocks of the Antenna Module (MS3.1)

14.6.2.2. MS3.2: Receiver Module

The receiver module models the instrument electronics response, and its impact in the output. The different modules are described below:

- **MS3.2.1** defines the instrument architecture (topology). A detailed hardware modeling of all the subsystems (e.g. amplifiers, filters, mixers, isolators…) is absolutely impractical for a generic simulator like this one. For this project it is proposed to include the receiver effect by applying the fringe-washing function and the equivalent system noise temperature, similar to the microwave radiometer system. The noise temperature is basically computed from the noise figure of the receiver components in this module. The other affecting terms are included in the other receiver modules, e.g., noise figure, gain fluctuations. Besides, the ADC and correlator effects are included.
- **MS3.2.2** computes the receiver's frequency response (noise bandwidth, filter shape, etc.) which will be used later on to compute the fringe-washing function, as a function also of the architecture (**MS3.2.1**)
- **MS3.2.3** defines the digital output of the instrument. It includes any digitalization process and the associated errors (correlator bit, quantization noise, clipping, jitter[16]…). This module also calculates the waveform sampling point according to the delay bin definition and tracking method input.

The physical temperature at different points of the receiver, and the actual antenna pointing (including satellite's orbit position and platform's attitude) will be added to the telemetry data to be downloaded for later post-processing.

[16] In a statistical sense sampling jitter will translate in an amplitude decrease.

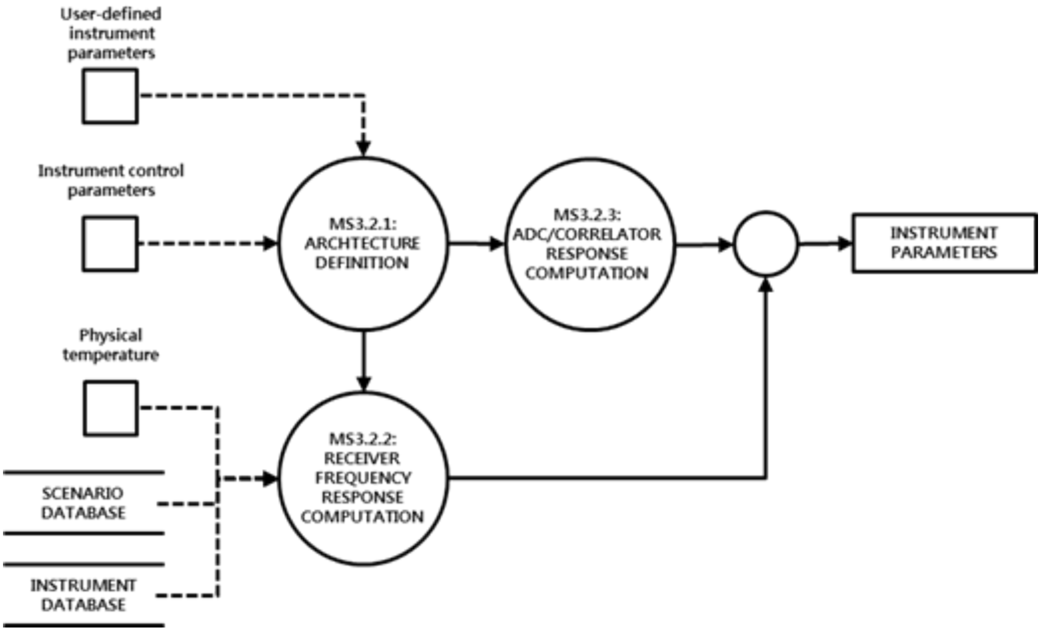


Figure 14-9. Structure and building blocks of the Receiver Module (MS3.2)

14.7. MS4: LEVEL 0 – OBSERVABLES GENERATOR

14.7.1. INTRODUCTION

In the proposed architecture, the Level-0 data processor is the one that actually computes the raw instrument's response; for GNSS-R, raw coherently integrated DDM. To do that the outputs of **MS2** (Scattering Scene) and **MS3** (Instrument Module) are required. The DDM has to be computed using the scattering coefficient scene / atmospheric effect (from **MS2**) and instrument parameters (from MS3), and geometry parameters (from **MS1**).

Based on the basic DDM model (1), the instrument output model (L0) is modeled as follows:

The effect of receiver can be applied as the convolution with the fringe-washing function. The thermal and speckle noise should be applied. The atmospheric (ionosphere and troposphere) effect is applied as an additional delay.

Let $\langle|Y(\tau,f_d)|^2\rangle$ in (1) be the ideal DDM $DDM_{id}(\tau,f_d)$, then the instrument output $DDM_m(\tau,f_d)$ can be expressed as

$$DDM_m(\tau,f_d) = g[DDM_{id}(\tau+\tau_{atm},f_d) * |\tilde{r}(\Delta\tau)|^2 + N(\tau+\tau_{atm},f_d)] \quad (4)$$

where $\tilde{r}(\Delta\tau)$ denotes the fringe-washing function, $N(\tau,f_d)$ denotes the noise component (thermal and speckle noise[17]), τ_{atm} denotes the atmospheric delay. The operators ' $*$ ' denotes the one-dimensional convolution, and ' $g[\cdot]$ ' denotes the operation of the digital count of cross-correlation output. This module performs the computation of (4) step-by-step.

In the following section, the block description is explained.

14.7.2. DESCRIPTION OF BUILDING BLOCKS

The Level 0 – Observables Generator block diagram is composed of four blocks, as shown in Figure 14-10:

17 Note that the speckle noise has multiplicative nature, so it will be proportional to the value of DDM_{id}

- **MS4.1** computes WAF according to the signal and instrument parameters. For example, the case of interferometric GNSS-R using GPS L1 band uses the composite WAF [RD. 129], and the conventional GNSS-R uses the triangle function in delay domain and sinc function in the frequency domain.
- **MS4.2** computes the ideal DDM $DDM_{id}(\tau + \tau_{atm}, f_d)$ using (1).
- **MS4.3** generates thermal and speckle noise. The thermal noise (additive noise) can be computed using the system equivalent noise temperature, receiver's bandwidth, and coherent integration time. The speckle noise can be modelled as multiplicative Rayleigh random distribution, and can be computed using the ideal DDM.
- **MS4.4** computes the instrument digital output by convolving and adding the contribution computed from previous modules. Basically, the output of module is a coherently integrated DDM, i.e., a sample DDM for incoherently averaged DDM. According to the instrument definition about the incoherent average (e.g., on-board or on ground), this module produce the L0 data.

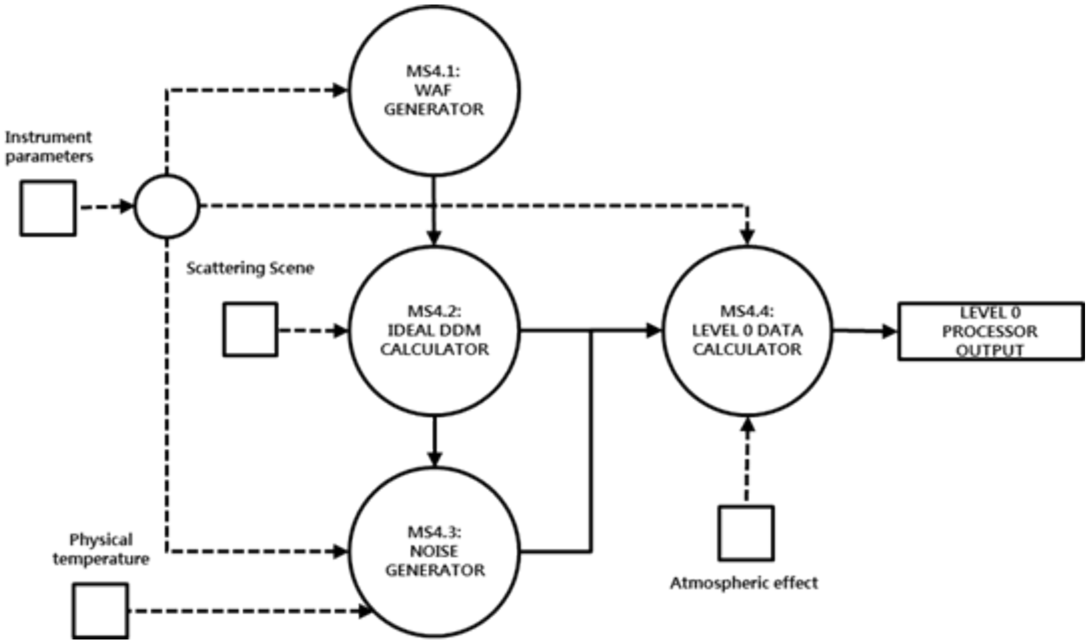


Figure 14-10. Structure and building blocks of the LEVEL 0 – Observables Generator Module (MS4)

14.8. MS5: LEVEL-1A PROCESSOR – CALIBRATED INSTRUMENT OBSERVABLES (AVERAGED DDM)

14.8.1. INTRODUCTION

In this work, the calibration of GNSS-R is defined by the correction of errors and scale change (or normalization) as compared to the (internal and external) reference power.

Because of the immaturity of GNSS-R space mission, there is no defined / validated calibration method compared with external reference source. However, for defining a reference architecture for an EO E2E simulator, it is assumed that the external source calibration is employed.

14.8.2. DESCRIPTION OF BUILDING BLOCKS

From the L0 data, the digital output from internal reference looking is used to determine the calibration parameters. Using this, the digital output of external reference is accompanied with determination of the final calibration parameters. These calibration parameters are used to compensate the error of actual Earth observation digital output. Then, the L1A calibrated instrument observables are produced. In Figure 14-11, the structure of this module is illustrated.

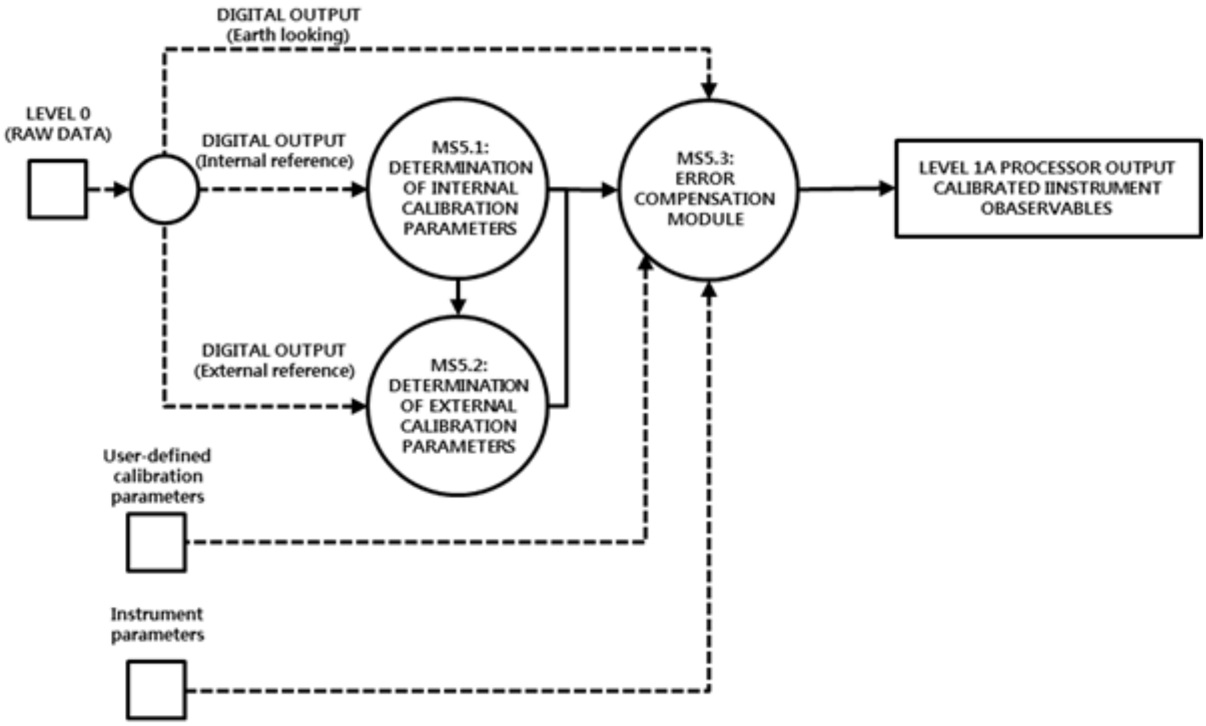


Figure 14-11. Structure and building blocks of the LEVEL 1A Processor – Calibrated Instrument Observables Generator Module (MS5)

14.9. MS6: LEVEL 1 B PROCESSOR – TOA GEOLOCATED INSTRUMENT OBSERVABLES

14.9.1. INTRODUCTION

After the observables are calibrated, the data is then processed by the Level 1B processor which performs all the geometrical corrections, geolocalization, i.e., assign the Earth coordinate to the FOV and the specular reflection point . The required input data comes from the Geometry module (**MS1**), the location of scattering scene in FOV. These parameters are already attached as ancillary data to the Level 0 raw data. Note that the spatial resampling to the regular grid will be performed in the LEVEL2 after the retrieval of the geophysical parameters because the GNSS-R observables DDM is not the direct imaging of the Earth.

Figure 14-12 summarizes the main building blocks, which are described in the following section.

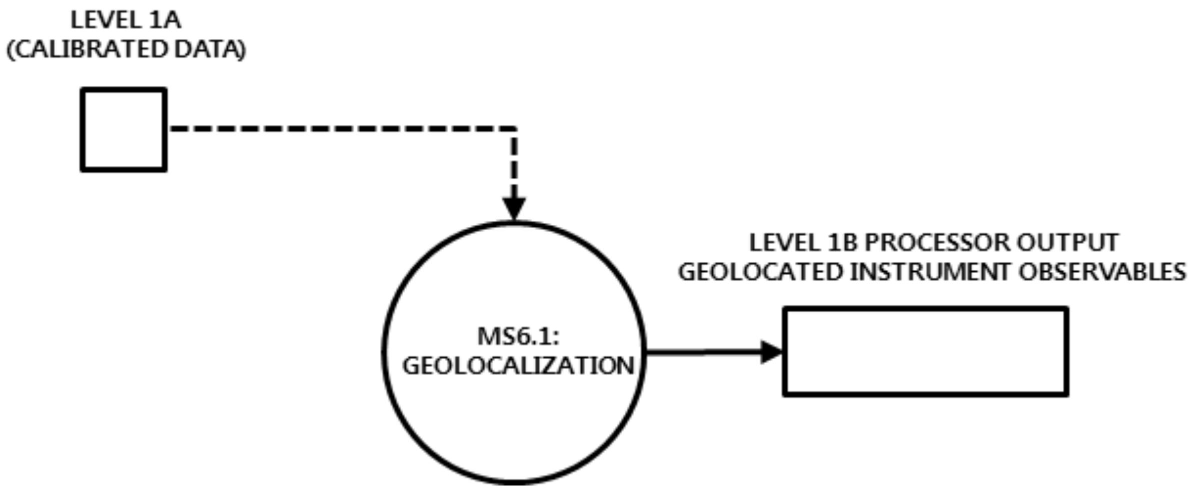


Figure 14-12. Structure and building blocks of the LEVEL 1B Processor – Geolocated Instrument Observables Module (MS6)

14.10. MS7: LEVEL-2 RETRIEVAL MODULE

14.10.1. INTRODUCTION

The Level-2 Retrieval module is in charge of obtaining the geophysical parameters from the geolocated calibrated DDM data over a discrete global grid. The retrieval algorithms are very mission specific, and still being studied. However, this architecture design accommodates them as a building block.

The building blocks for the Level-2 Retrieval module are described in the following sections.

14.10.2. DESCRIPTION OF BUILDING BLOCKS

The Level-2 Retrieval module structure is sketched in Figure 14-13. The first module **MS7.1** is in charge of separating the different targets (land, ocean, and ice) from the flag provided in the Level 1B data products.

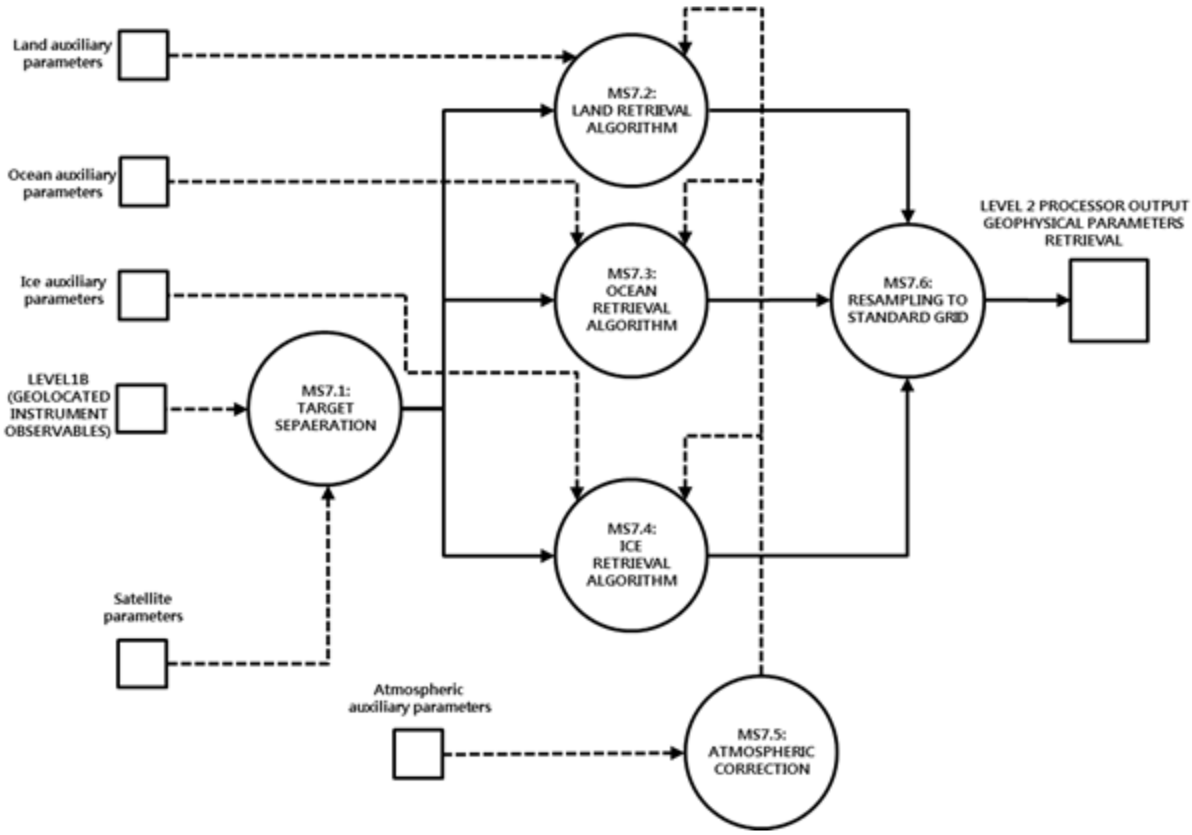


Figure 14-13. Structure and building blocks of the LEVEL 2 Processor – Geophysical Parameter Retrieval Module (MS7)

Then, different retrieval algorithms have to be applied depending on the type of surface being imaged (**MS7.2** for land, **MS7.3** for ocean, and **MS7.4** for ice) and the geo-physical parameter to be retrieved. In retrieving the geophysical parameters, the atmospheric and ionospheric delays are computed in **MS7.5** and compensated.

The retrieved geophysical parameters (e.g., sea height, SWH, soil moisture, ice depth, etc.) are then resampled from the arbitrary coordinates given by the FOV, into a standard discrete global grid in **MS7.6**. Finally, the output is L2 geophysical parameters on standard global grid, which can be assessed by comparing the input in **PAM** module.

14.11. PAM: PERFORMANCE ANALYSIS MODULE

14.11.1. INTRODUCTION

In every mission simulator, results must be evaluated to assess the suitability of the proposed configuration (platform + instrument + geo-physical parameter retrieval) to satisfy the mission requirements. The Performance Analysis Module (**PAM**) performs this type of studies.

Because the GNSS-R is not an imaging sensor, it does not produce the image (such as the brightness temperature map in radiometer or σ^0 in a SAR). In other words, the image input of simulation is the scattering scene from **MS2,** but the output of instrument is a DDM, which is not a one-to-one correspondence to the input scattering scene. Consequently, the PAM is mainly conducted at the level of L2 retrieved geophysical parameters:

- L2 geo-physical parameters retrieved:
    - rms between the retrieved and the original parameters used as input of the simulation,
    - bias between the retrieved and the original parameters used as input of the simulation.

Additionally, the platform and instrument quality assessment has to be performed to analyze the suitability of their behavior (e.g. platform attitude is within predefined limits, accuracy of array antenna beam pointing, instrument's physical temperature and gradients are within pre-specified limits, etc.):

14.11.2. DESCRIPTION OF BUILDING BLOCKS

The Performance Analysis Module (**PAM**) module will compare different outputs of the simulation process (L2 – retrieved geo-physical parameters) and original geo-physical parameters to assess the instrument and retrieval algorithms performance.

Figure 18 presents the different modules for the performance analysis. **PAM1** presents the platform parameters (including orbit ephemeris, LTAN, attitude – Euler angles etc.), **PAM2** presents the instrument parameters (receivers' physical and noise temperatures, bandwidth, antenna patterns, etc.), **PAM3** evaluates the instrument end-to-end performance L2 (geophysical parameters and error maps), respectively. Finally, **PAM4** presents all the previous results in the form of tables and/or as graphical results.

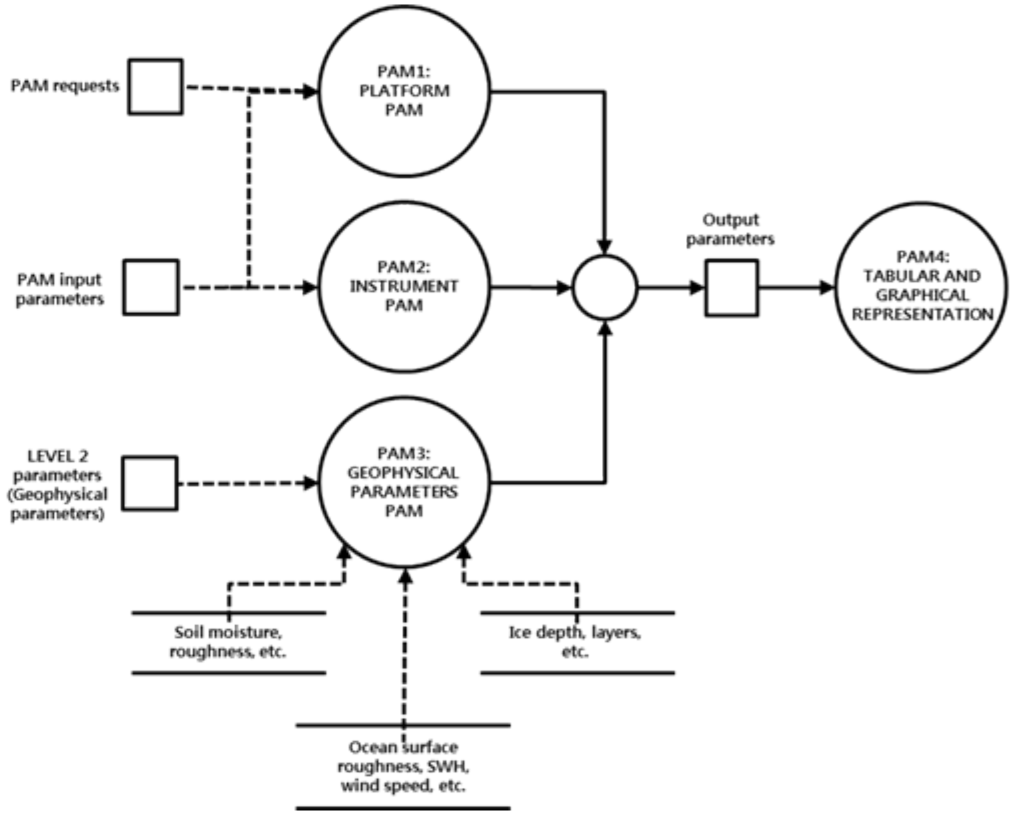


Figure 14-14. Structure and building blocks of the Performance Analysis Module (PAM)

14.12. CONCLUSIONS

This section has presented a proposal for an E2E Mission Simulator Architecture for GNSS-R instruments. The architecture is composed of three main blocks: a mission definition module (**MD**), a mission simulation module (**MS**), and a performance analysis module (**PAM**). At its time, the **MS** module is composed of seven sub-modules that compute specific parameters of the simulation process: **MS1**: geometry module, **MS2**: scattering scene module, **MS3**: instrument module, including antennas and receivers, **MS4**: Level 0 – observables generator, **MS5**: LEVEL 1 A processor – calibrated instruments observables, **MS6**: LEVEL 1B processor – geolocated calibrated data, and finally the **MS7**: LEVEL 2 processor – geophysical parameters retrieval algorithm.

Each of these modules has been sub-divided into smaller building blocks to perform basic operations.

The proposed architecture is believed to be flexible enough to be used for various GNSS-R instruments (e.g., conventional / interferometric GNSS-R, single and array antennas, various GNSS signals).

Most probably some modules and building blocks related to the satellite states and instruments can be shared with other EO sensors.

END OF DOCUMENT

