CSC Data Processors Re-engineering

Software Verification and Validation Report for ASGARD

Reference: CSGF-CSC-DPR-VV_ASGARD

Issue: 1.1 - Date: 02/05/2024

Document Status Log

	Name	Function	Date	Signature
Prepared by:	ASGARD Team		02/05/2024	
Verified by:	Jonathan TAGLIONE	Quality Manager	02/05/2024	Signature numérique de Jonathan TAGLIONE Date : 2024.05.02 14:50:11 +02'00'
Approved by:	Carine QUANG	Project Manager	02/05/2024	Signature numérique de Carine QUANG Date : 2024.05.02 14:55:33 +02'00'

Contents

List of Figures

Table of Tables

1. Document Scope and Applicability

1.1 Context

ASGARD will be the component in charge of geometry computation in re-engineered Sentinel processors. It has to be validated against legacy processors. It is a common part of the re-engineering between the different Sentinel instruments, so the same guidelines apply: it shall be Python library, open-source, efficiently using Numpy, scalable with Dask.

1.2 Scope

This document covers the numerical and performance validation of ASGARD library. It covers the ASGARD version 3 (V3) validation, and addresses the processing levels L0 and L1 for the following instruments:

- ✓ Sentinel 1 SAR (will be available in V4)
- ✓ Sentinel 2 MSI
- ✓ Sentinel 3 OLCI
- ✓ Sentinel 3 SLSTR
- ✓ Sentinel 3 SRAL (will be available in V5)
- ✓ Sentinel 3 MWR (will be available in V5)

For each sensor, two implementations will be available:

- ✓ A “legacy-based” implementation, using either EOCFI (for S1 and S3) or SXGeo (for S2 MSI)
- ✓ A “refactored” implementation, using refactored low-level models. Most models are pure-Python, but some of them rely on Orekit (wrapped in Python using JCC and GraalVM)

This document provides validation results for both implementations, as the legacy-based implementation can give results closer to the original IPF.

1.3 Applicable and Reference Documents

1.3.1 Applicable documents

Table 1 – Applicable Documents

Reference	Title	ID	Version	Date
[AD-ATBD-S3L1_OLCI]	Sentinel-3 Phase B2/C/D/E1 OLCI Level 0, Level 1B Algorithm Theoretical Basis Document	S3-ACR-TN-007	5.0	10/12/2014
[AD-ATBD-S3L1_SLSTR]	SLSTR: Algorithm Theoretical Basis Definition - Document for Level 1 Observables	S3-SL-RAL-TN-032	8.0	01/04/2021
[AD-ATBD-S3L1_SYN]	Level 1 Algorithms Theoretical Baseline Document - Part 2: Optical Products	ESIG_SY-24_S3-DD-TAF-SY-00620_ATBD_L1C_I11	11	17/02/2015

Reference	Title	ID	Version	Date
[AD-ATBD-S2L1]	Sentinel-2-Level-1-Algorithm-Theoretical-Basis-Document-ATBD_v1.1	S2-PDGS-MPC_ATBD-L1_V1.1	1.1	31/01/2023
[AD-ATBD-S1L1]	Sentinel-1 Level 1 Detailed Algorithm Definition	SEN-TN-52-7445	2.5	17/11/2022

1.3.2 Reference documents

Table 2 – Reference Documents

Reference	Title	ID	Version	Date
[RD-EOCFI-Note-052]	SENTINEL-1 IPF: ATTITUDE QUATERNIONS USAGE	EOCFI-NOTE-052	2.1	14/10/2014

1.4 Definitions and abbreviations

1.4.1 Abbreviations

The general abbreviations used throughout the CSC DPR project are defined in [RD-GL]. More specifically, the following abbreviations are used in this document.

Table 3 – Abbreviations

Abbreviation	Description
EO	Earth Observation
DEM	Digital Elevation Model
DPM	Detail Processing Manual
DS	Datastrip
IDP-SC	Independent Software Component
IERS	International Earth Rotation and Reference Systems
IPF	Instrument Processing Facility
L0c	Level-0 consolidated

Abbreviation	Description
L0u	Level-0 unconsolidated
L1	Level-1
MJD	Mean Julian Day
MTD	Metadata
RAC	Radiometric Calibration
S1	Sentinel-1
S2	Sentinel-2
S3	Sentinel-3
TDS	Test Data Set

1.4.2 Definitions

Table 4 – Descriptions

Term	Description
Datastrip	Part of an acquisition with all its information (metadata)
Direct/Inverse Location	✓ a direct location is the computation of ground coordinates from sensor coordinates. ✓ an inverse location is the computation of sensor coordinates from ground coordinates.
Refining	Process to refine satellite pointing to correct small vibrations for example

2. VALIDATION STRATEGY

2.1 ASGARD description

ASGARD is a geometry toolbox designed to perform all sensor geometry computations needed by Sentinels L0 and L1 processors. The features offered by this Python library come as a substitute for EOCFI and S2Geo software, used in legacy processors. The library has two API levels:

- ✓ A high-level API: the user gives input parameters to initialize the geometry of a Sentinel product, and he is able to compute direct/inverse location, footprint, as well as Sun and viewing angles.
- ✓ A low-level API: the user can compose different models that operate a specific task (time conversions, datation, instrument line-of-sight, orbit and attitude estimation, line-of-sight propagation, ...)

Please note that the low-level API is mostly implemented for the “refactored” sensor implementations. Only the time conversion and Earth coordinate conversion models have two flavours: one using EOCFI, and the other using Orekit wrappers.

2.2 Validation scope and objectives

The validation purpose is to verify that future re-implementation of Sentinel processors using ASGARD can meet the performance of legacy processors, in term of geometric precision and runtime execution.

At design phase, an analysis of geometry functions used in legacy processors was performed. The goal was to map the geometry functions (supplied by EOCFI or S2Geo) to ASGARD classes and functions. There is not a direct relationship between them, because of architecture choices for ASGARD. For the sake of durability, ASGARD is not a plain rewrite of EOCFI API in Python.

For instance, some sequence of calls to EOCFI functions (orbit init, attitude computation, xp_target_inter) are mapped to high-level functions like AbstractProduct.direct_loc(). But the low-level API from ASGARD gives an access to smaller steps in this sequence.

That being said, the validation strategy for ASGARD is the following:

- ✓ Focus on high-level functions in ASGARD: they are designed to emulate blocs of geometry computations found in legacy processors. The results from legacy processors will serve as baseline.
- ✓ Some low-level models (such as TimeReference and EarthBody) can be validated against their EOCFI counterparts (ExplorerTimeReference and ExplorerEarthBody) because they have a similar API, and very little added logic compared to EOCFI COTS.

The main challenges of this validation strategy are:

- ✓ To configure ASGARD and legacy processors so that they operate on the same data context. As the legacy processor has a rigid interface, the solution is to derive the ASGARD data context from the processor job order file.
- ✓ To access intermediate data generated by legacy processor. Some geometric computations are not accessible as such in the output L1 product.

The choice of input data and auxiliary files is a central task in the validation process. Some attention is required in particular for the DEM. As the re-engineered processors will use a DEM in ZARR format, we propose the following strategy to validate the impact of DEM format:

- ✓ Generate the reference data using the Legacy processor (using legacy DEM),
- ✓ Run ASGARD “legacy-based” implementation, with Legacy DEM and compare results with references,

- ✓ Run ASGARD “refactored” implementation with Legacy DEM and compare with previous (if “refactored” implementation supports the legacy DEM),
- ✓ Run ASGARD “refactored” implementation with Zarr DEM and compare with previous results.

Each comparison will be done by acceptance thresholds: if the difference of the value observed between the 2 versions is under the defined threshold, the test is passed, and the value is considered as equivalent.

2.3 Verification metrics and requirements

The basics of the validation is to compare results between two runs. Metrics shall then be defined with acceptance thresholds for each one. Compared metrics will be:

- ✓ Time: Time manipulation/conversion (S3 only)
- ✓ Position/velocity: Ephemeris value in ECI/ECF coordinates (S3 only)
- ✓ Ground location: Results of direct location/footprint
- ✓ Image location: Result of inverse location
- ✓ Incidence angles: Azimuth, zenith
- ✓ Sun angles: Azimuth, zenith
- ✓ Footprints
- ✓ Processing features (processing time/RAM)

Each metrics required a dedicated comparison tool, with dedicated thresholds. Tools are common to all and are described in §13.1 but thresholds specific to each Satellite/implementation/configuration. Following sections define those thresholds and their reasons/justifications.

2.3.1 Sentinel-1 thresholds

The following thresholds will be used to validate the accuracy of Sentinel 1 implementation.

Type	Metric	Requirement	Comment
Time	Absolute difference	10^-8 s	Accuracy of range time (fast time) needs to better than range sampling time (~40 ns).
Orbit position	Norm of difference	10^-2 m	
Orbit velocity	Norm of difference	10^-4 m/s	
Ground location	Planimetric distance error at iso elevation	2 m	The lowest resolution in S1 products is 5x5m.
	Altitude error	1 m	

Image location	Difference along each coordinate RMSE	10^-3 pixel	
Incidence angles	Angular distance between pointing directions	10^-4 deg	
Viewing angles	Angular distance between pointing directions	10^-4 deg	
Roll/pitch/yaw angles	Difference element-wise	10^-3 deg	

2.3.2 Sentinel-2 thresholds

Thresholds cannot be the same between the Legacy-base implementation and the Refactored one. Indeed, the Legacy-based is expected to have results very close to the reference as it uses the exact same code with the exact same DEM. Only the loading and configuration are done by ASGARD, which shall have a very tiny impact. On the other hand, the Refactored implementation can have differences higher but still acceptable as the code has been reimplemented and the programming language is not even the same (Python versus java). And finally, using Legacy DEM versus using Zarr DEM can also increase the differences. Hence, 3 tables are presented bellow for validation of ASGARD applied to Sentinel-2:

- ✓ Legacy-based using Legacy DEM: Table 5,
- ✓ Refactored using Legacy DEM: Table 6,
- ✓ Refactored using Zarr DEM: Table 7.

Thresholds are not sensor dependant (and more especially not band dependant) as the precision of the method does not depend on it.

Table 5 – Sentinel-2 metrics and thresholds for Legacy-based implementation

Type	Metric	Threshold	Comment
Ground location	✓ Planimetric distance error at iso elevation ✓ Altitude error	✓ 10-6m ✓ 10-6m	Results are expected to be almost the same as the legacy code is embedded and the same DEM is used.
Image location	Difference along each coordinate	10-2 pixel	Results are expected to be almost the same as the legacy code is embedded and the same DEM is used.
Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	10-3 degree	Results are expected to be almost the same as the legacy code is embedded and the same DEM is used.
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	10-3 degree	Results are expected to be almost the same as the legacy code is embedded and the same DEM is used.

Type	Metric	Threshold	Comment
			Factor 10 below the precision of the model
Footprints	✓ Surface ratio ✓ Distance max between points of footprint to the other one	✓ 0.95 (TBC) ✓ 1 pixel (TBC)	Footprint are supposed to be very close, however, it depends on the sampling used: ✓ The surface ratio is expected to be very close to 1, ✓ Footprint are expected to be very close together. (TBC how the Legacy-based is computing the footprint, and both sampling)
Processing			For now, no threshold is defined, comparison is only given as information

Table 6 – Sentinel-2 metrics and thresholds for Refactored implementation using Legacy DEM

Type	Metric	Threshold	Comment
Ground location	✓ Planimetric distance error at iso elevation ✓ Altitude error	✓ 10-4m ✓ 10-4m (TBC)	Results are expected to be a little different as the code is refactored, however, the same DEM is used so differences shall not be too high.
Image location	Difference along each coordinate	10-1 pixel (TBC)	Results are expected to be a little different as the code is refactored, however, the same DEM is used so differences shall not be too high.
Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	10-3 degree (TBC)	Results are expected to be a little different as the code is refactored, however, the same DEM is used so differences shall not be too high.
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	10-3 degree (TBC)	Results are expected to be a little different as the code is refactored, however, the same DEM is used so differences shall not be too high. Factor 10 below the precision of the model
Footprints	Surface ratio and distance max between points of footprint to the other one	✓ 0.95 (TBC) ✓ 1 pixel (TBC)	Footprint are supposed to be very close, however, it depends on the sampling used: ✓ The surface ratio is expected to be very close to 1, ✓ Footprint are expected to be very close together. (TBC how the Legacy-based is computing the footprint, and both sampling)
Processing			For now, no threshold is defined, comparison is only given as information

Table 7 – Sentinel-2 metrics and thresholds for Refactored implementation using Zarr DEM

Type	Metric	Threshold	Comment
Ground location	✓ Planimetric distance error at iso elevation ✓ Altitude error	TBD	For future versions.
Image location	Difference along each coordinate	TBD	For future versions.
Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	TBD	For future versions.
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	TBD	For future versions.
Footprints	Surface ratio and distance max between points of footprint to the other one	TBD	For future versions.
Processing		TBD	For future versions.

2.3.3 Sentinel-3 thresholds

For Sentinel-3, the thresholds depend on the instrument because of its own spatial resolution for ground location.

Moreover, for the instrument SLSTR, there are two different geometry views: 1km and 0.5km which lead to specific thresholds and location computation on different grid like tie point grid and quasi-cartesian grid which also leads to specific thresholds.

2.3.3.1 Sentinel-3 OLCI legacy based

The thresholds presented in the table below are the one for the Legacy-Based implementation for OLCI instrument (same DEM as legacy processor).

Table 8 – Sentinel-3 OLCI legacy based metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	10 m	The spatial resolution of OLCI Full Resolution (FR) is 300m and 1.2km for OLCI Reduced Resolution (RR). As we are in legacy-based context the results are expected to be almost the same as legacy processor.
	Altitude error	10 m	
Incidence angles	Error between two pointing directions. Each direction expressed by (phi, theta)	10^-3 deg	The legacy-based context aims at considering this low threshold
Sun angles	Error between two pointing directions. Each direction expressed by (phi, theta)	10^-4 deg	The legacy-based context aims at considering this low threshold

2.3.3.2 Sentinel-3 OLCI refactored

The thresholds presented in the table below are the one for the refactored implementation for OLCI instrument (different DEM as legacy processor).

Table 9 – Sentinel-3 OLCI refactored metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	75 m	The spatial resolution of OLCI Full Resolution (FR) is 300m and 1.2km for OLCI Reduced Resolution (RR). As we are in legacy-based context the results are

Type	Metric	Requirement	Comment
			expected to be almost the same as legacy processor.
	Altitude error	10 m	
Incidence angles	Error between two pointing directions. Each direction expressed by (phi, theta)	10^-3 deg	Same threshold as legacy-based context
Sun angles	Error between two pointing directions. Each direction expressed by (phi, theta)	10^-4 deg	Same threshold as legacy-based context

2.3.3.3 Sentinel-3 SLSTR legacy based

The thresholds presented in the table below are the one for the Legacy-Based implementation for SLSTR instrument (same DEM as legacy processor).

Table 10 – Sentinel-3 SLSTR legacy based metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation for grids (with geometry)	500 m 1 km	Depends on the geometry and the grid where the direct location is computed. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Planimetric distance error at iso elevation for quasi cartesian grid	16 km	Independent of any geometry. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Altitude error	1 m	

Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-3 deg	The legacy-based context aims at considering this low threshold
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-4 deg	The legacy-based context aims at considering this low threshold

2.3.3.4 Sentinel-3 SLSTR refactored

The thresholds presented in the table below are the one for the refactored implementation for SLSTR instrument (different DEM as legacy processor).

Table 11 – Sentinel-3 SLSTR refactored metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation for grids (with geometry)	500 m 1 km	Depends on the geometry and the grid where the direct location is computed. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Planimetric distance error at iso elevation for quasi cartesian grid	16 km	Independent of any geometry. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Altitude error	1 m	
Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-3 deg	
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-4 deg	

2.3.3.5 Sentinel-3 MWR legacy based

The thresholds presented in the table below are the one for the Legacy-Based implementation for MWR instrument.

Table 12 - Sentinel-3 MWR legacy based metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	15.5 m	The spatial resolution of MWR Full Resolution (FR) is 20 km. As we are in legacy-based context the results are expected to be almost the same as legacy processor.

2.3.3.6 Sentinel-3 MWR refactored

The thresholds presented in the table below are the one for the refactored implementation for MWR instrument.

Table 13 - Sentinel-3 MWR refactored metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	15.5 m	The spatial resolution of MWR Full Resolution (FR) is 20 km.

2.3.4 Low level functions thresholds

Table 14 - Low level functions thresholds

Type	Metric	Requirement	Comment
Time	Absolute difference	10^-9 JJ	= 8.6 10^-5 s = 86 µseconds Formula used for comparison: | *computed – expected* | <= *(10^-8 + 10^-9 * | expected |)*
	Exact value		The computed time is exactly the expected time
Longitude / Latitude	Absolute difference	10^-9 deg	Difference between computed and expected
Altitude	Absolute difference	10^-4 m	Difference between computed and expected

Geodetic distance	Absolute difference	3 m	Difference between computed and expected
Azimuth	Absolute difference	10^-6 deg	Difference between computed and expected
Position	Absolute difference	11 m	For transformation from EF to J2000: Norm of the difference between computed and expected position
	Relative difference	10^-4	For transformation from EF to J2000: Norm of (the difference between computed and expected position)/expected position
	Absolute difference	10^-8 m	For transformation from EF to J2000 and back to EF: Norm of (the difference between computed and original position)/original position
Velocity	Absolute difference	5.10^-2 m/s	For transformation from EF to J2000: Norm of the difference between computed and expected velocity
	Relative difference	10^-4	For transformation from EF to J2000: Norm of (the difference between computed and expected velocity)/expected velocity
	Absolute difference	10^-11 m/s	For transformation from EF to J2000 and back to EF: Norm of (the difference between computed and original velocity)/original velocity
Sun position	Absolute difference	2.10^6 m	Norm of the difference between computed and expected position
	Relative difference	5e-5	Norm of (the difference between computed and expected position)/expected position
Sun velocity	Absolute difference	200 m/s	Norm of the difference between computed and expected velocity
	Relative difference	2e-4	Norm of (the difference between computed and original velocity)/original velocity

3. SENTINEL 2 MSI

3.1 Legacy processor

S2GEO is the S2 IPF (Sentinel-2 Image processing facility) component to support geometrical processing needs, such as:

- ✓ Initializing datation and viewing directions from metadata and parameter files,
- ✓ Converting masks and images from one viewing model to another
- ✓ Computing physical attributes such as incidence and solar angles, locating pixels/lines on ground ...

The geometry computation is done using S2GEO main dependency: Rugged, an opensource java library strictly focused on direct and inverse location computation, and Orekit for all what concerns transformations between coordinate systems (inertial/terrestrial) and orbital data. The mission-specific library S2GEO implements the Sentinel-2 viewing model by managing S2 file formats and conventions; it acts as the layer between Rugged and the IPF components.

This library is called by command line (by S2 IPF IDP-SC) with an input file containing all information needed for processing (call S2GEO_Input_Interface.xml). Information could be:

- ✓ Links to external data (as DEM_SRTM, DEM_GEOID, IERS file for example),
- ✓ Metadata and other information required to initiate the model,
- ✓ Configuration parameters (algorithm to apply, detector or band to process, and so on).

IDP-SC also put arguments in the command line to limit heap space and to set the number of DEM Tile in cache. For V3, the default configuration (without arguments in command line) of S2GEO is used.

It is to be noted that, historically, the refining part of the Sentinel-2 processing is done by IDP-SC (components/modules) that are based on other libraries, and another one that S2GEO for geometric part, but its purpose is limited to compute direct and inverse locations (that are perfectly done by RUGGED).

3.2 Validation scope

Main roles of S2GEO are:

- ✓ Computation of direct locations and inverse locations:
  - ➢ On several DEM configurations (with or without DEM, at altitude constant, and so on)
  - ➢ On several Captor configuration (L0 simplified model, L0 model, L1 model geometrically un-refined, L1 model geometrically refined, perfect sensor),
- ✓ Computation of Viewing angles:
  - ➢ On several Captor configuration (L0 simplified model, L0 model, L1 model geometrically un-refined, L1 model geometrically refined, perfect captor),
- ✓ Computation of Sun angles:
  - ➢ On several Captor configuration (L0 simplified model, L0 model, L1 model geometrically un-refined, L1 model geometrically refined, perfect captor),
- ✓ The computation of footprint can be resumed at direct location from corners (of the datastrip, of a detector, of a chunk or other), however ASGARD is able to compute detector footprints for Sentinel-2.

The purpose of ASGARD for Sentinel-2 is then to be able to do the same and the validation strategy will concentrate on validating the above quoted roles, anticipating as much as possible specific cases that could occur.

As detailed in §1.2, ASGARD has two implementations, with different behaviours:

- ✓ Legacy-based:
  - ➢ This version calls the legacy processors, meaning for S2, SXGEO,
  - ➢ This version uses the Legacy DEM (in legacy format) and not the Zarr DEM,
  - ➢ The behaviour is then expected to be very close to S2GEO results, and issues based on specific cases are expected to behave as the legacy one, so no specific cases shall be applied.
- ✓ Refactored:
  - ➢ Internal functions are recoded, using pyrugged and Orekit,
  - ➢ It can use the DEM in format as in legacy one, but can also use the Zarr format (recommended),

As explained in §3.1, the legacy processor can be called with restriction on heap space and number of DEM tile in cache. Default behaviour of S2GEO was used to generated references for the V3, default behaviour of SXGEO of ASGARD legacy-based is also used for tests, without forcing parameters.

It is to be noted that the legacy DEM format is much more complex (with one grid file per square degree, of different resolution depending on latitude, and with missing DEM Tiles over water area). The behaviour of ASGARD will then not be tested for all very particular cases of the Legacy DEM, for the refactored implementation as the DEM in Zarr is the new reference that shall be used. However, some specific cases shall still be tested in refactored implementation regarding the Zarr DEM:

- ✓ Robustness to antemeridian,
- ✓ Robustness to meridian 0,
- ✓ Robustness to the equator,
- ✓ High latitude behaviour.

3.3 Validation methodology

3.3.1 Basics

The strategy is the same described in §2.2, summarized in:

- ✓ Comparing [Legacy-based using Legacy DEM] versus [Legacy],
- ✓ Comparing [Refactored using Legacy DEM] versus [Legacy],
- ✓ Comparing [Refactored using Zarr DEM] versus [Refactored using Legacy DEM].

This step-by-step comparison allowing to discard DEM differences as sources of overall differences.

Description of libraries, hardware, DEM used is available in Appendix/13.3.1-Sentinel-2 context configuration.

In the following sections, description of specific cases links to Sentinel-2 and specifics tests are described.

3.3.2 Specific cases

Specific cases can be split in 2 main parts:

- ✓ Linked to the DEM (§3.3.2.1),
- ✓ Linked to degraded cases of the input data (§3.3.2.2).

3.3.2.1 DEM

The Legacy DEM contains some specific cases that could be difficult to handle:

- ✓ Antemeridian,
- ✓ Transition at meridian 0,
- ✓ Transition at the equator,
- ✓ Transition between switch of resolution (in latitude, at 50°/60°/70°/75°/80°/85°, to keep a resolution in meter more or less equivalent over the world). In high latitude, tiles are smaller, but more shall be loaded to have the same area covered,
- ✓ Transition between non existing DEM Tiles (completely over water with all values at 0) and existing ones covering land.

The Zarr DEM contains some specific cases on which ASGARD robustness shall be tested:

- ✓ Antemeridian,
- ✓ Transition at meridian 0,
- ✓ Transition at the equator,
- ✓ As the DEM has a constant resolution in Lat/Long degrees over the world, high latitudes are oversampled:
    - ➢ RAM consumption and processing time shall be monitored carefully on those area,
    - ➢ Results shall be analysed to see if difference occurs between high and lower latitude (more or less precise)

As detailed in §3.2, only the Zarr DEM specific cases will be tested.

3.3.2.2 Degraded Cases

For now, 2 degraded cases are identified:

- ✓ The satellite is not pointing to the earth: ASGARD shall return a clear Exception/error message saying that it is not possible to reach the earth from the satellite or to reach the satellite from the earth,
- ✓ Input data defining the sensor are corrupted in some way (missing GPS part, and so on). (ASGARD Behaviour TBC)

3.3.3 Tests and configuration tested

As already presented, not all specificities shall be tested with all configurations. What will be tested is summarized in this section.

Implementation specificities:

- ✓ Legacy-based:
    - ➢ DEM shall not be an issue as the Legacy processor is embedded, so special cases of the DEM shall be handled the same way,

- Degraded cases shall not be an issue as the Legacy processor is embedded, so they shall be handled the same way,
- For future versions (superior at V3), if needed, specific calls to limit heap space and DEM tiles in cache could be studied.

✓ Refactored, using Legacy DEM:
- DEM special cases are not mandatory as the Legacy DEM is not the final recommendation. This version is specific to compare embedded nominal cases versus code reengineered, with the same configuration and especially the same DEM to have equivalent results regardless of issue coming from DEM.
- Refining
- Perfect Sensor (will be tested in future version)

✓ Refactored, using Zarr DEM, all tests shall be done (will be tested in future version):
- Zarr DEM special cases:
  - Antemeridian,
  - Meridian 0,
  - Equator,
  - High latitudes,
- Degraded cases:
  - Pointing issue,
  - Corrupted data,
- Refining
- Perfect Sensor (future evolution if needed, not in V4)

For the 3 steps above several cases shall be launched to cover all need of the life of the Sentinel-2 data. Cases are summarized in the table below. If not specified, SRTM means DEM SRTM 90.

Table 15 - Configuration list

Model	L0u	L0c/L1A	L1B	L1C	L1C Not Refined
Direct location	✓ GLOBE	✓ SRTM ✓ Constant Altitude	✓ SRTM ✓ Constant Altitude	✓ SRTM ✓ SRTM 30	✓ SRTM
Inverse Location	✓ GLOBE	✓ SRTM ✓ Constant Altitude	✓ SRTM ✓ Constant Altitude	✓ SRTM ✓ SRTM 30	✓ SRTM
Inverse Location grids			✓ Constant Altitude	✓ SRTM ✓ SRTM 30	✓ SRTM
Sun angles	✓ GLOBE	✓ SRTM ✓ Constant Altitude (TBC)	✓ SRTM ✓ Constant Altitude (TBC)	✓ SRTM	✓ SRTM
Viewing angles	✓ GLOBE	✓ SRTM ✓ Constant Altitude (TBC)	✓ SRTM ✓ Constant Altitude (TBC)	✓ SRTM	✓ SRTM

Model	L0u	L0c/L1A	L1B	L1C	L1C Not Refined
Detector footprint	✓ GLOBE	✓ SRTM	✓ SRTM		

For L0c and L1B, a perfect sensor column might be tested if really required, this might be added in further version.

3.3.4 Metrics

Metrics and thresholds used for validation and comparison are described in §2.3.

3.4 Validation test datasets

For the validation, several Test Data Sets (TDS) were selected to cover cases listed in section 3.2 (Validation scope).
They are summarised in the following table, but each TDS is more described in following subsections.

Table 16 – Sentinel-2 Test Datasets Description

TDS	Length	Hemisphere	Satellite	Levels treated	Refining	Particularities
TDS 1	25s	North	S2A	L0c->L1C	Refined	✓ Small Island
TDS 2	123s	South	S2B	L0c->L1C	Refined	✓ Ante meridian
TDS 3	747s	North	S2A	L1B->L1C	Not refined	✓ Meridien 0
TDS 4	563s	North (and South)	S2B	L1B->L1C	Refined	✓ Equateur
TDS 5	527s	North	S2A	L0u->L1C	Not refined	✓ High Latitude ✓ L0u
TDS 6						✓ Degraded pointing
TDS 7						✓ Degraded/Corrupted data
TDS 8						✓ Antemeridian ✓ Footprint in several parts
TDS9						✓ Moon pointing

For each TDS the following computation are done:

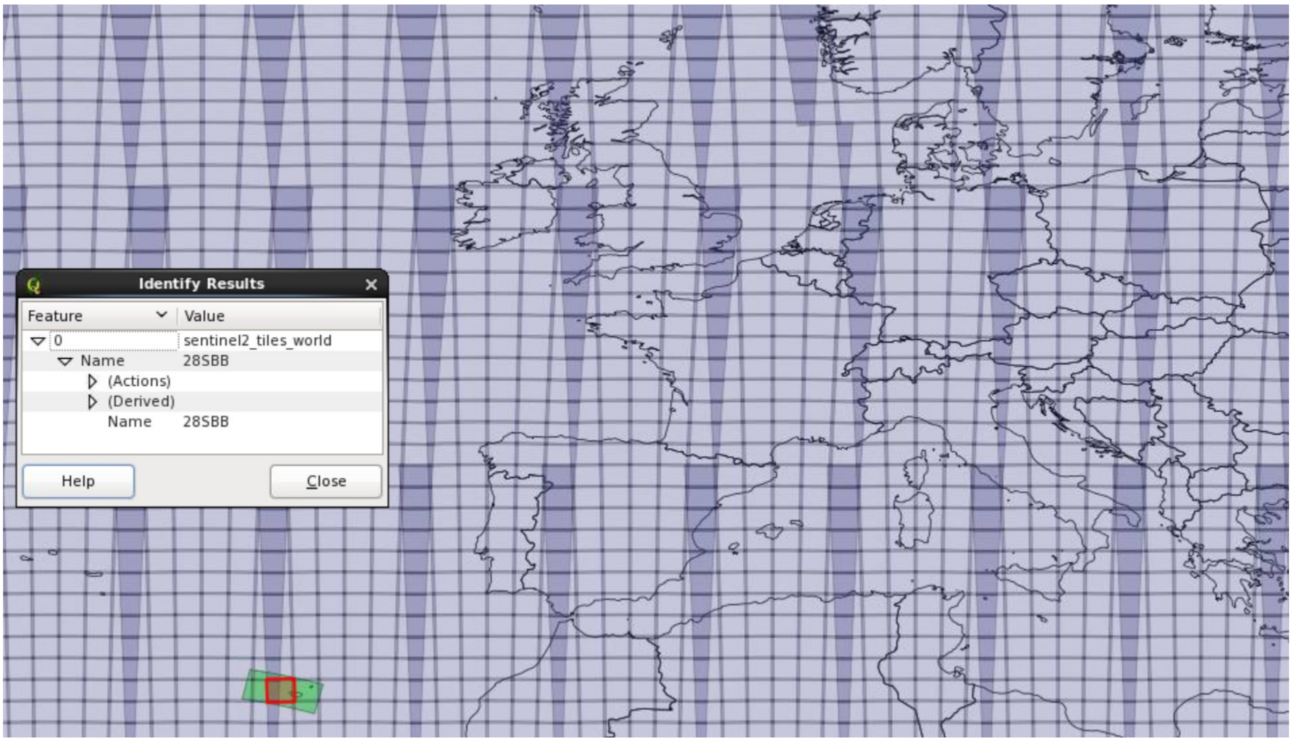
- For each configuration applicable (see §13.3.2.2 for configurations), a loop over all sensors (couple detector/band) is done. For each sensor, 9 points (corners, and middle cross) of the sensor are defined and for each point:
  - Direct location is done and compared to references,
  - Inverse locations are done on the ground coordinates found and then compared to the initial point,
  - Viewing angles of the ground points are computed and compared to references,
  - Sun angles of the ground points are computed and compared to references.

  If reference is available, then, for each sensor, a footprint is computed and compared to the referenced one.
- Also, a dedicated L1C Tile has been chosen to perform an inverse location grid on it and to compare both output grids (using references grids).

3.4.1 S2_MSI_TDS1 - Small Island

This datastrip is over a small island (Madeira). Its characteristics are:

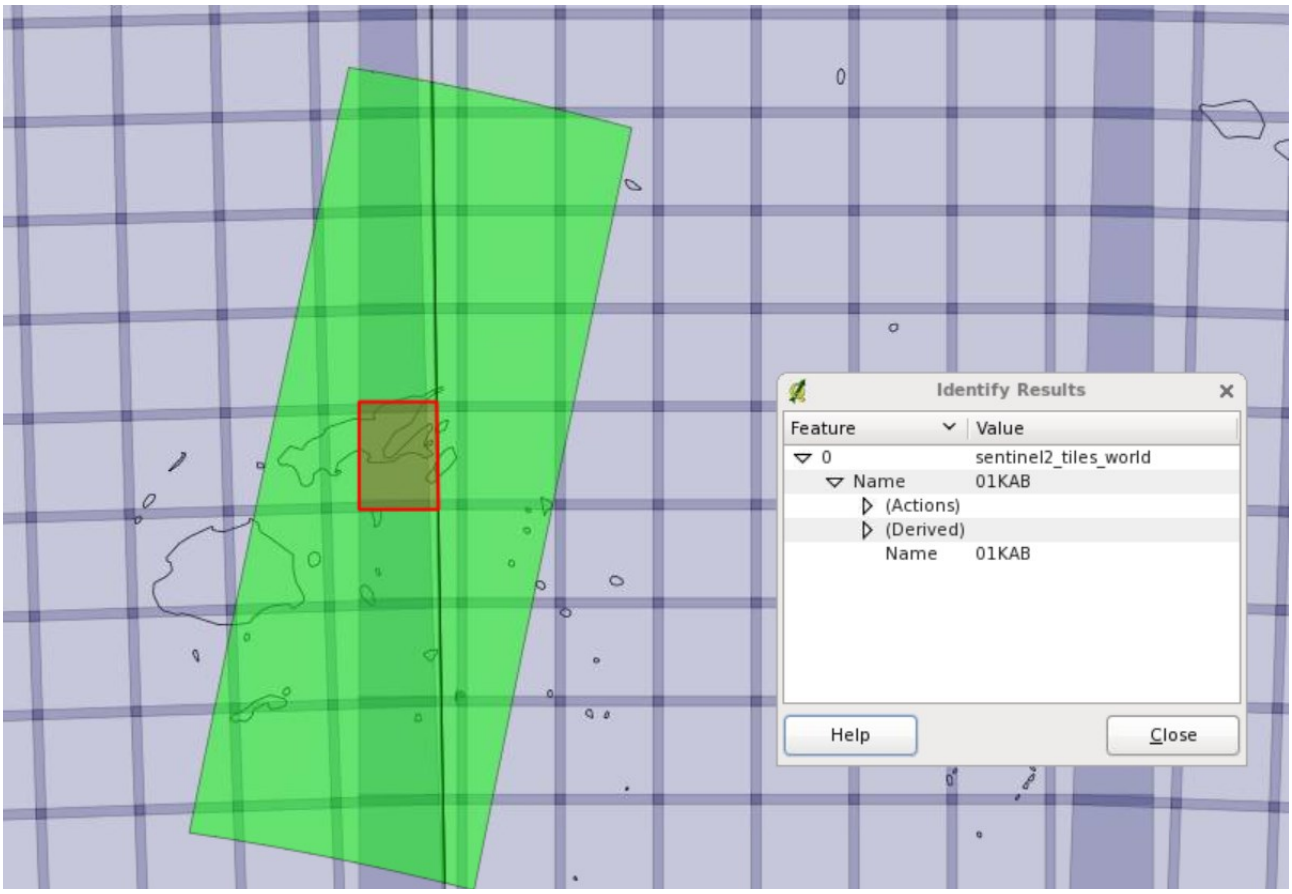
- Datastrip sensing start: 2020-08-16T12:02:20.458Z
- Datastrip sensing time: 25s
- Satellite: Sentinel-2A
- Levels covered: L0c -> L1C
- Chosen L1C tile:
  - 28SBB
  - Both over land and over water
- Reference generation:
  - S2Geo versions used: 06.02.98 (based on 06.02.00 with additional logs)
  - Directory in CS Group Netapp: /DATA_Netapp/aburie/validation_output_ASGARD/S2PDGS-TC-IPF-V2-ORCH-N-016_CR136_Madeira_AllTiles_Batch2/TaskTable_20231207T092205
  - Reference location on ASGARD S3: s3://geolib-input/S2MSIdataset/S2MSI_TDS1



3.4.2 S2_MSI_TDS2 - Antemeridian TDS

This datastrip is crossing the antemeridian and is still over a small island. Its characteristics are:

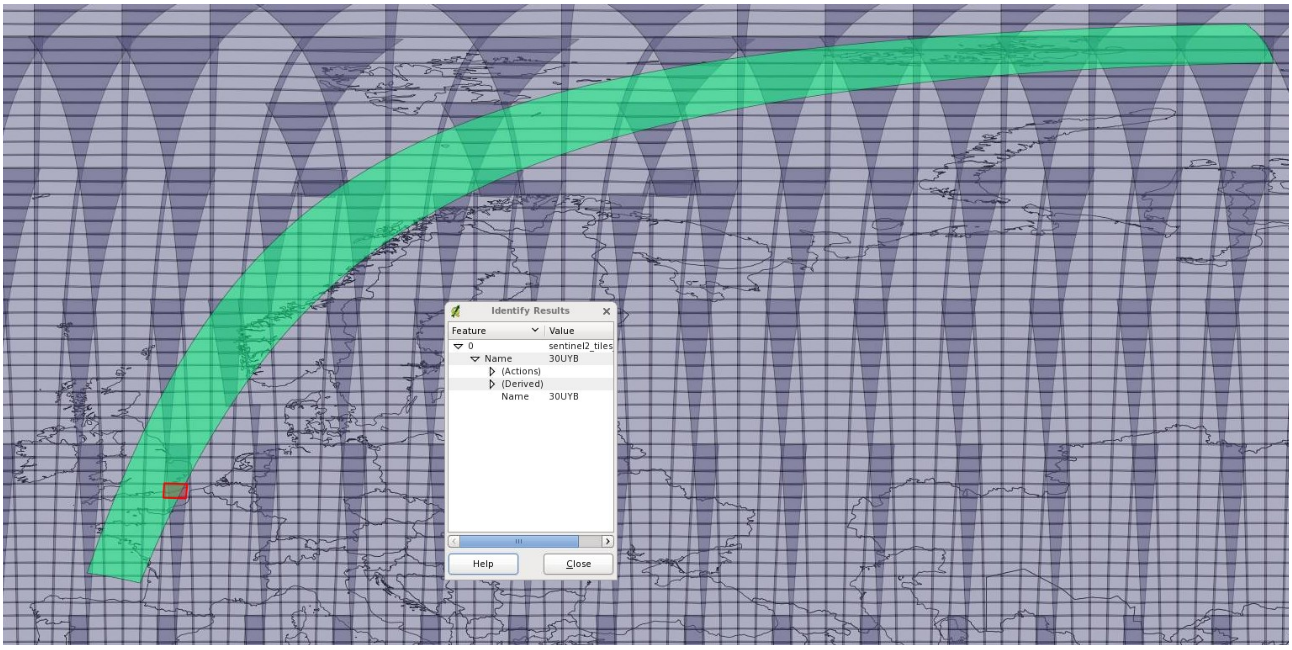
- ✓ Datastrip sensing start: 2019-09-06T22:19:41.462Z
- ✓ Datastrip sensing time: 123s
- ✓ Satellite: Sentinel-2B
- ✓ Levels covered: L0c -> L1C
- ✓ Chosen L1C tile:
    - ➢ 01KAB
    - ➢ Crossing the antemeridian but with a significant land cover
- ✓ Reference generation:
    - ➢ S2Geo versions used: 06.02.98 (based on 06.02.00 with additional logs)
    - ➢ Directory in CS Group Netapp: /DATA_Netapp/aburie/validation_output_ASGARD/S2PDGS-TC-IPF-V2-ORCH-N-016_DEM30_TDS9-R029_6.3/TaskTable_20231204T115727
    - ➢ Reference location on ASGARD S3: s3://geolib-input/S2MSIdataset/S2MSI_TDS2


Identify Results
Feature
Value
0
sentinel2_tiles_world
Name
01KAB
(Actions)
(Derived)
Name
01KAB
Help
Close

3.4.3 S2_MSI_TDS3 - Meridian 0 TDS

This datastrip is crossing meridian 0 at UK level. Its characteristics are:

- ✓ Datastrip sensing start: 2019-09-09T11:06:18.462Z
- ✓ Datastrip sensing time: 747s
- ✓ Satellite: Sentinel-2A
- ✓ Levels covered: L1B and L1C inverse location grids
- ✓ Chosen L1C tile:
  - ➢ 30UYB
  - ➢ Crossing the meridian 0, for DEM handling validation
- ✓ Reference generation:
  - ➢ S2Geo versions used: 06.02.09 (based on 06.02.00 with additional logs and generation of inverse location grids)
  - ➢ Directory in CS Group Netapp: /DATA_Netapp/aburie/validation_output_ASGARD/S2PDGS-TC-IPF-V2-ORCH-N-016_DEM30_TDS1.1-R137/TaskTable_20240122T102823/GEN_ORTHO_TOA_ASGARD
  - ➢ Reference location on ASGARD S3: s3://geolib-input/S2MSIdataset/S2MSI_TDS3

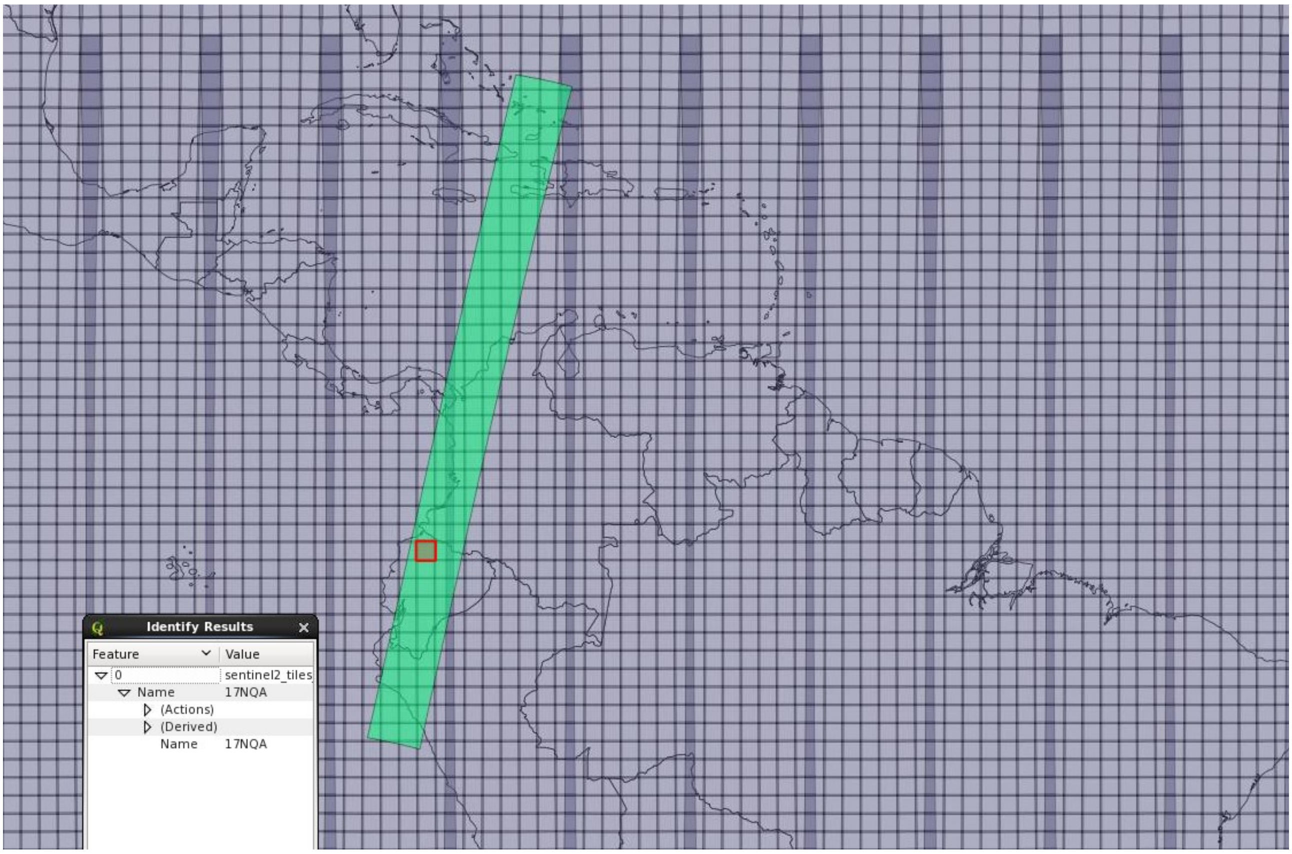


3.4.4 S2_MSI_TDS4 - Equator TDS

This datastrip is crossing the equator over the Equator country. Its characteristics are:

- ✓ Datastrip sensing start: 2024-01-16T15:36:24.459Z
- ✓ Datastrip sensing time: 563s
- ✓ Satellite: Sentinel-2B
- ✓ Levels covered: L1B and L1C inverse location grids
- ✓ Chosen L1C tile:
  - ➢ 17NQA

    - ➢ Crossing the equator, for DEM handling validation
- ✓ Reference generation:
    - ➢ S2Geo versions used: 06.02.97 (based on 06.02.00 with additional logs and generation of inverse location grids)
    - ➢ Directory in CS Group Netapp: /DATA_Netapp/aburie/validation_output_ASGARD/S2PDGS-TC-IPF-V2-ORCH-N-017_L0U_ASGARD_EQUATOR/TaskTable_20240119T165448/
    - ➢ Reference location on ASGARD S3: s3://geolib-input/S2MSIdataset/S2MSI_TDS4

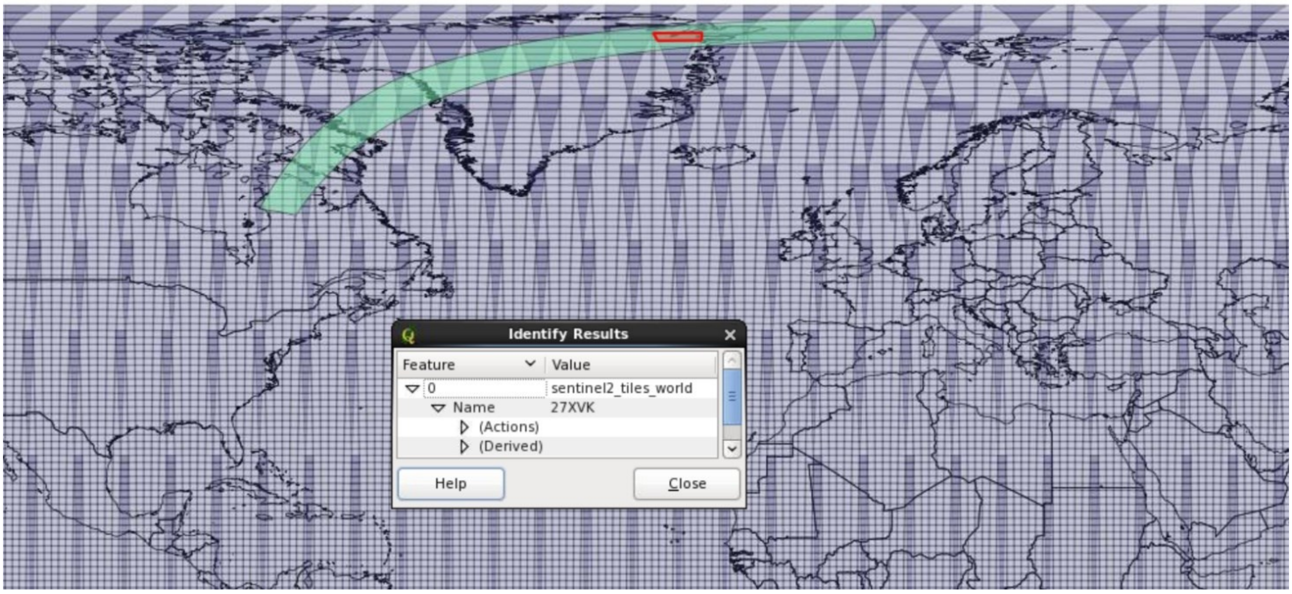


3.4.5 S2_MSI_TDS5 - High latitude

This datastrip is crossing the antemeridian and is still over a small island. Its characteristics are:

- ✓ Datastrip sensing start: 2023-05-04T16:18:28.456Z
- ✓ Datastrip sensing time: 527s
- ✓ Satellite: Sentinel-2A
- ✓ Levels covered: L0c -> L1C
- ✓ Chosen L1C tile:
    - ➢ 27XVK
    - ➢ Over land and in very high latitudes, will be good for performances linked to the differences of resolution of the DEM
- ✓ Reference generation:
    - ➢ S2Geo versions used: 06.02.98 (based on 06.02.00 with additional logs)

- Directory in CS Group Netapp: /DATA_Netapp/aburie/validation_output_ASGARD/S2PDGS-TC-IPF-V2-ORCH-N-017_L0U_JUPITER_S2A_DT35/TaskTable_20231214T091221
- Reference location on ASGARD S3: s3://geolib-input/S2MSIdataset/S2MSI_TDS5



3.4.6 S2_MSI_TDS6 - Degraded pointing (earth not intersected)

TDS for future versions. (TBD)

3.4.7 S2_MSI_TDS7 - Degraded data

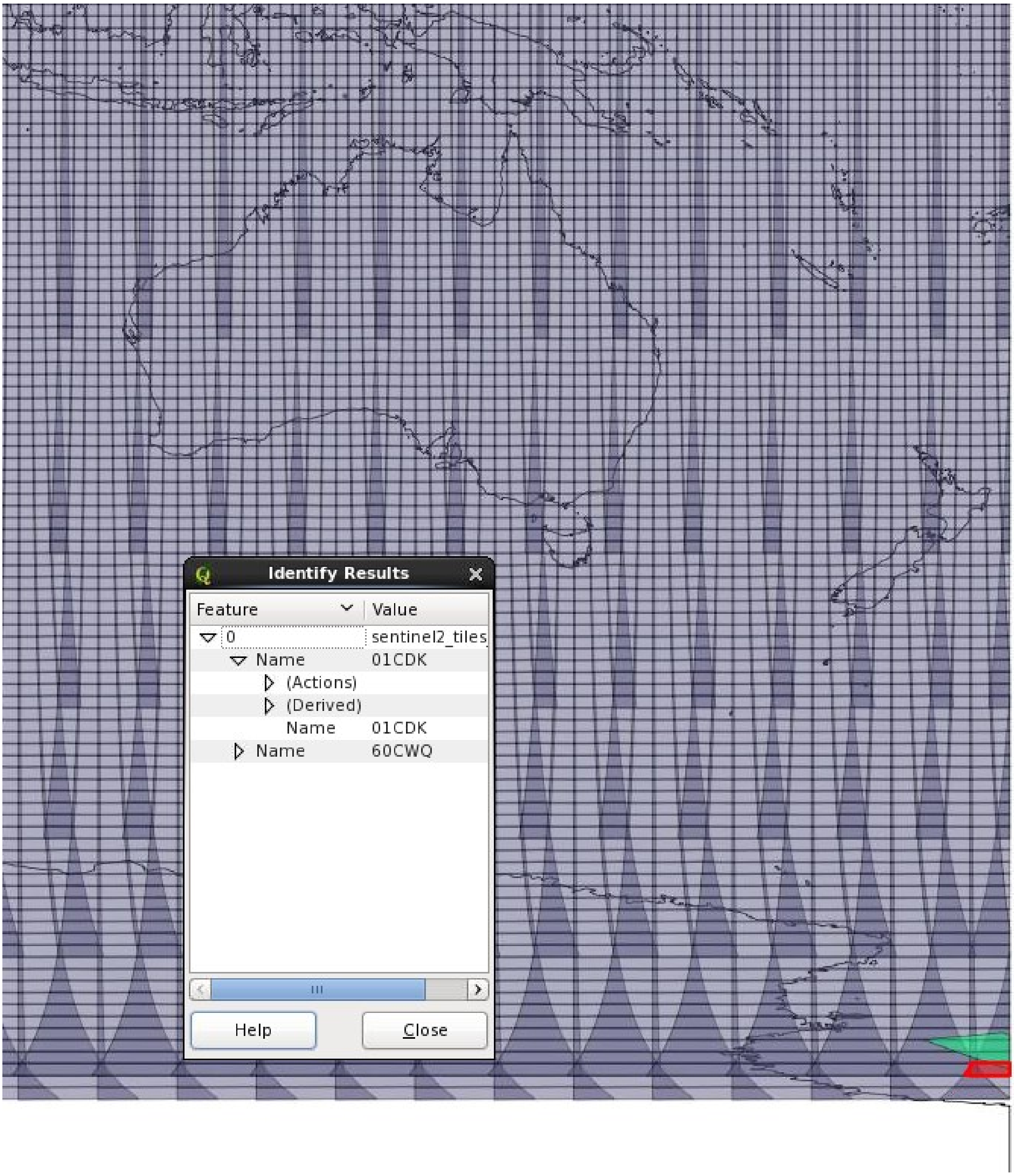
TDS for future versions. (TBD)

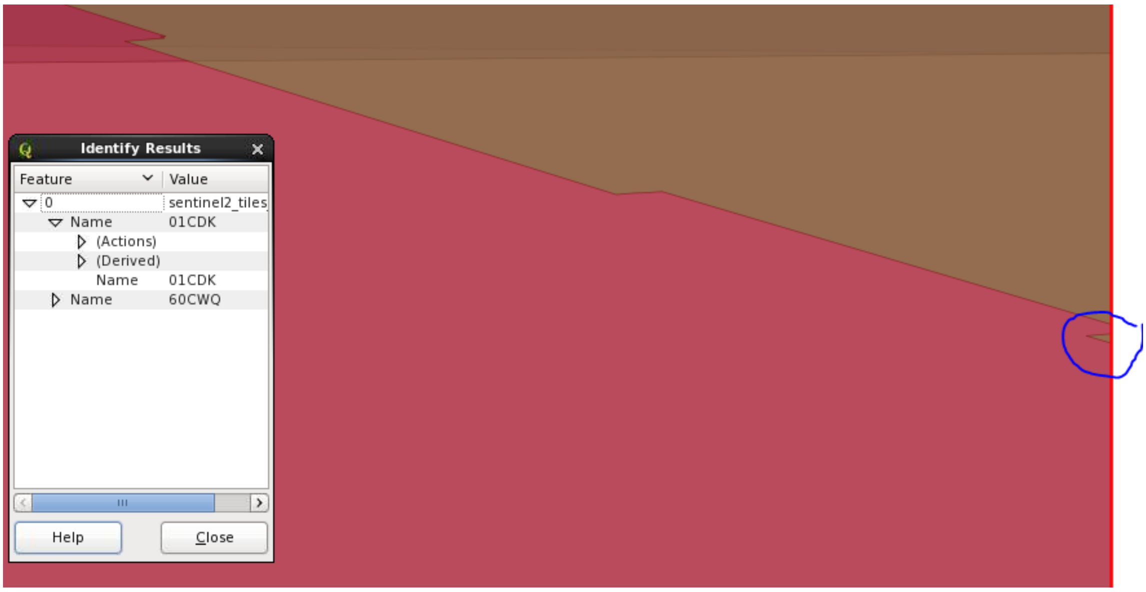
3.4.8 S2_MSI_TDS8 - Footprint in several parts

This datastrip is crossing the antemeridian and its footprint is in several parts due to antemeridian and detector crenelation:

- ✓ Datastrip sensing start: 2023-01-28T18:27:46.462Z
- ✓ Datastrip sensing time: 25s
- ✓ Satellite: Sentinel-2A
- ✓ Levels covered: TODO
- ✓ Chosen L1C tile:
  - 60CWQ
  - TODO
- ✓ Reference generation:
  - S2Geo versions used: 06.02.97 97 (based on 06.02.00 with additional logs and generation of inverse location grids)

- Directory in CS Group Netapp: /DATA_Netapp/aburie/validation_output_ASGARD/S2PDGS-TC-IPF-V2-ORCH-N-007_L0U_ASGARD_ANTE_Multi-Footprint/TaskTable_20240122T152228/
- Reference location on ASGARD S3: TBD





3.4.9 S2_MSI_TDS9 - Moon pointing

TBD that this is required for ASGARD. If confirmed, TDS for future versions.

3.5 Prerequisites

With V3.3, ASGARD has been split into 2 repositories: asgard and asgard-legacy. Asgard-legacy is now an extension of asgard to allow using the legacy-based implementation. Tests will be done using this repository, as both refactored and legacy-based implementation shall be tested in parallel.

For the Sentinel-2 validation you need:

- ✓ One Asgard environment with:
  - ➢ The Asgard installation and dependencies. Either whl file shall be taken, or all gits and dependencies, as described in https://gitlab.eopf.copernicus.eu/-/snippets/7
  - ➢ Additional modules from the Asgard-legacy source code (https://gitlab.eopf.copernicus.eu/geolib/asgard-legacy):
    - `tests/test_sentinel2_msi.py` (unit test with hard-coded data paths to s3)
    - `validation/scripts/validate_sentinel2_msi.py` (used to override the unit test with custom validation data paths in V3)
- ✓ (Optional) To rerun references, one environment with "S2Geo for Asgard" installed.
  - ➢ This refers to the latest S2Geo version with specific modifications to also print the sun and incidence angle values when calling the direct location from the command line.
  - ➢ Please contact the CS Group S2-IPF team from DPR project to obtain this working environment.
- ✓ To know how to launch ASGARD with adapted options.

You can find in the following sub-sections, the way to:

- ✓ Configure your environment to launch ASGARD,

- ✓ Launch ASGARD to generate reference coordinates that shall be used with the Legacy processor to generate the reference for each TDS,
- ✓ Launch the Legacy processor to generate the references per TDS, if not already done,
- ✓ Run ASGARD validation script versus Legacy results.

3.5.1 ASGARD environment configuration

After download of ASGARD and the whl files, the environment of validation shall be configured following those steps:

```
# if outside the Docker:
tmpd=$(mktemp -d)
python -m venv $tmpd
source $tmpd/bin/activate

# Go in the folder where you have asgard git(but not inside the asgard git folder) and the whl files at disposal
cd […]

# Continue the installation
pip install orekit-jcc*.whl sxgeo-*.whl pyRugged-*.whl asgard*.whl # asgard one, not asgard-legacy one
export EE_DIR=$(realpath ./EOCFI)
git config --global --add safe.directory "$(realpath ./asgard-legacy)"
pip install --verbose -e "./asgard-legacy[dev]"

# You may need some additional modules that are missing from the asgard binary installation
pip install pytest rasterio shapely termcolor dill


# Add the validate_sentinel2_msi.py and test_sentinel2_msi.py modules to your PYTHONPATH:
export PATH=$PATH:/sbin/
export PYTHONPATH="$(realpath ./asgard-legacy)/validation/scripts:$(realpath ./asgard-legacy)/tests/:$PYTHONPATH"
export ASGARD_DATA="$(realpath ./asgard-legacy/tests/resources/“

# cython path might be needed, if so, add it (example for python 3.10):
export PYTHONPATH="/build/lib.linux-x86_64-cpython-310:$PYTHONPATH"
```

3.5.2 (Optional) Generation of the references

For S2MSI_TDS from 1 to 5 (ASGARD V4), references have already been generated and are included and linked in the official TDS stored on S3 storage. Hence, those steps are not needed.

3.5.2.1 (Optional) Direct locations, angles and footprint references

3.5.2.1.1 (Optional) Generation of the reference script

S2GEO, the Legacy processor, has been improved to:

- ✓ Take a file with several pixel coordinates as input: it allows computing all actions on all pixel at once, avoiding calling several time S2GEO which takes time to initialize,
- ✓ Log information about direct locations, sun angles and viewing angles computations when one direct location of a pixel is asked.

The reference script is a bash script that will call S2GEO from the command line to calculate:

- ✓ direct locations,
- ✓ sun angles,
- ✓ incidence angles,
- ✓ processing time.

The script then saves results in a txt file that will serve as reference data for Asgard.

We use Asgard to generate this reference script. Asgard will:

- ✓ Use its S2Geo legacy loader to read the S2Geo interface file from the test dataset (containing references to all required inputs),
- ✓ Read each sensor pixel footprint,
- ✓ Calculate several pixel coordinates distributed around and inside each sensor footprint (9 points per detector/band),
- ✓ Generate the bash script that calls S2Geo and saves the reference data.

Optional: if Legacy references are not available, to compute them:

To generate the reference script, run these lines in your Asgard environment terminal previously initialized:

```
# Show the validate_sentinel2_msi.py module help, it should display:
# validate_sentinel2_msi.py [-h] (-s PATH | -r PATH) [-d PATH] [-l] [-m SECONDS]
INTERFACE_PATH ...
python3 -m validate_sentinel2_msi -h

# Run the test_sentinel2_msi.py module on your S2Geo validation data to generate the
reference script
python3 -m validate_sentinel2_msi \
    /path/to/input/S2GEO_Input_interface.xml \
    -s /path/to/output/reference_script.sh

# Your reference script should be written under the provided path.
# A reference text file should be written above the provided path (just the extension is
changed).
```

3.5.2.1.2 (Optional) Legacy validation data generation using the reference script

Optional: if Legacy references are not available, to compute them (as previous step):
The reference script will generate the reference data.

The idea is then to:
- ✓ Take an already existing context,
- ✓ Clean the S2GEO_Input_interface of it to remove the computation asked in it,
- ✓ Use this S2GEO_Input_interface as reference to generate the reference script (previous section)
- ✓ Use this S2GEO_Input_interface with the reference script and the modified S2GEO to generate the reference data.

To do so, switch to your S2GEO environment terminal and run these lines:

```
# cd to the dir that contains both:
# ./target/s2geo-core-*-jar-with-dependencies.jar
# ./resources/orekit-data
cd /path/to/s2geo

# You can run a sample S2Geo command to make sure everything is in place
java -jar ./target/s2geo-core-*-jar-with-dependencies.jar \
    -i /path/to/input/S2GEO_Input_interface.xml \
    -o directLoc -f reference_script.txt

# It should print (among others) the direct location, and the sun and incidence angles
for this sensor and pixel e.g.
# B01, D01 : {x: 124.40240837985563 , y: 55.09380019968725 , z: 658.0226189383338}
# B01, D01 sun angles : {azimuth: 166.31192260306517 , zenith: 53.93841258478091}
# B01, D01 incidence angles : {azimuth: 113.66387073669358 , zenith: 11.97386027845401}

# Now you can run the reference script generated by Asgard in the previous step
chmod u+x /path/to/reference_script.sh
/path/to/reference_script.sh

# Each result is printed in the terminal saved automatically in a txt file by the script.
# There is one result per sensor * 9 points.
# The results are saved as reference data in the same directory than the reference
script,
# in /path/to/output/s2geo_reference.txt
```

3.5.2.2 (Optional) Inverse location grids reference

Optional: if Legacy references are not available, to compute them, several steps shall be done:

- ✓ Take a context that already run and especially the IDP-SC GEN_ORTHO_TOA
- ✓ Modify a S2GEO_Input_Interface.xml (make a _Ref.xml copy first if needed) to:
  - ➢ Compute TOA for all bands
  - ➢ Remove the IncidenceAngles computation part
  - ➢ Leave the L1B masks projection part untouched
  - ➢ Modified the file (ate the very end of the file) to be the one to use
- ✓ Run S2Geo modified on this file and save the log (it will generate a log with all points of the grids, the Lat/Long/Alt grid and the corresponding pixels in sensors) for all bands:

```
# cd to the dir that contains both:
# ./target/s2geo-core-*-jar-with-dependencies.jar
# ./resources/orekit-data
cd /path/to/s2geo

# You can then run the following S2Geo command
java -jar ./target/s2geo-core-*-jar-with-dependencies.jar \
    [XXX]/S2PDGS-TC-IPF-V2-ORCH-N-[XXX]/TaskTable_[XXX]/GEN_ORTHO_TOA_ASGARD/tmp_[XXX]/tmp_GEN_ORTHO_TOA[XXX]/B[NN]/S2GEO_Input_interface_Ref.xml \
>                                                    [XXX]/S2PDGS-TC-IPF-V2-ORCH-N-[XXX]/TaskTable_[XXX]/GEN_ORTHO_TOA_ASGARD/tmp_[XXX]/tmp_GEN_ORTHO_TOA[XXX]/B[NN]/reference_grid_log.txt

# It should print (among others) the grids points for each band (Lat/Long/Alt grid and the corresponding pixels in sensors)
#B10 09 -179.7495793844438 -17.256260909099627 50.393068650338826 230.34580716797396 6649.552533387571
#B09 09 -179.7495793844438 -17.256260909099627 50.393068650338826 231.2905126474651 6647.61279750875
#B01 09 -179.7479074998096 -17.256283789015313 50.394899406918974 234.0805088882749 6646.713554189338
#B10 09 -179.7479074998096 -17.256283789015313 50.394899406918974 233.22618268785087 6648.92751224732
```

Then you can:

- ✓ copy the `reference_grid_log.txt` into a `reference_grid.txt` and remove all the line that differs from the print grid points (*all first lines*, then search for "*20.0m*", "*60.0m*", "*Toa*", "*dump*" to find locations where there are other lines that the grid points)
- ✓ Launch the modified S2GEO to recompute the grid and compute processing time of it, with:
  - ➢ The S2GEO_Input_Interface prepared as for direct_location (cf §3.5.2), **not the one used to generate the grid**,
  - ➢ The grid logs (cleaned)
- ✓ Store the log

```
# cd to the dir that contains both:
# ./target/s2geo-core-*-jar-with-dependencies.jar
# ./resources/orekit-data
cd /path/to/s2geo

# You can then run the following S2Geo command
java -jar ./target/s2geo-core-*-jar-with-dependencies.jar \
    -i /path/to/input/S2GEO_Input_interface.xml \
    -o inverseLoc -f reference_grid.txt \
    > [path_to]/s2geo_reference_grid_INVLOC_Full.txt
```

S2GEO has been modified to be allowed to treat the inverse location of all lines of the file. It will relog its new results (which shall be identical) and will compute the processing time.

The file is processed twice by S2GEO:

- ✓ the first time, it is processed, and each result is logged in the console,
- ✓ the second time, only the processing is done, without logging, hence, 2 processing time are available at the end, one with the logging (higher) and one without, which is closer to what will be done in use.

The reference can then be used by ASGARD for its validation (detailed in next section).

3.5.3 Run ASGARD with legacy references

With V4, pytests have been written, with data pushed on ASGARD S3 remote storage. To launch manually on other or with different configuration, please refer to the previous method, described in Annexe: §13.3.2.2-Run ASGARD with legacy reference.

The example command line to be derived for all TDS and levels is:

```
python3 -m pytest asgard-legacy/tests/test_legacy_sentinel2_msi.py -s -v -k
test_sentinel2_msi[<TDS_Name>_<Level>_<Altitude>_<OPTION_><Implementation>]
```

With:

- ✓ <TDS_Name>: S2MSI_TDS<N> with N from 1 to 5 for V4.
- ✓ <Level>:
  - ➢ L0u: L0u simplified sensor configuration
  - ➢ L0c: L0 sensor configuration
  - ➢ L1B: L1B sensor configuration (with refining parameters if available)
  - ➢ L1C: L1C sensor configuration (with refining parameters if available)
- ✓ <Altitude>:
  - ➢ DEM: Using DEM (GEOID + [GLOBE for L0u; SRTM for other levels])
  - ➢ CONST: Using GEOID + constant altitude (3000m). In this case, the footprint will not be computed
- ✓ <Option_>:
  - ➢ “” <= empty: Compute the direct locations, inverse locations, angles for 9 points for all sensors
  - ➢ “INVLOC_”: Compute only inverse locations grids comparison for all sensors
- ✓ <Implementation>:
  - ➢ LEGACY: call the Legacy based implementation of ASGARD, with Legacy DEM

- REFACTORED: call the Refactored implementation of ASGARD twice, on time using the Legacy DEM, one time using the DEM converted in zarr format. If <Altitude> is set at CONST, two runs will still be launched, one with Legacy GEOID, one with GEOID converted in Zarr ; the altitude will then be constant (and is set at 3000m)

Examples:

- To call the Legacy-based implementation, using the S2MSI_TDS1 on operations that are not the Inverse Location grids:

```
python3 -m pytest asgard-legacy/tests/test_legacy_sentinel2_msi.py -s -v -k
test_sentinel2_msi[S2MSI_TDS1_L0c_DEM_LEGACY]
```

- To call the Refactored implementation, using the S2MSI_TDS5 and doing only the Inverse location grids:

```
python3 -m pytest asgard-legacy/tests/test_legacy_sentinel2_msi.py -s -v -k
test_sentinel2_msi[S2MSI_TDS5_L1C_DEM_INVLOC_REFACTORED]
```

Section 3.7 lists all possible tests per TDS for ASGARD V4.

3.6 How to read output results of tests

Logs are in 2 main parts:

- A looping phase over detectors and bands, computing and logging for each:
  - Direct location from the reference points (9 points per detector/band),
  - Inverse location from the ground points found,
  - Sun angles at ground point found,
  - Viewing angles at ground points found.

  During this looping phase, results are compared to References and a warning log appear if differences are above thresholds.

  Time is also measured per computation.
- A summary part logging:
  - what is above threshold,
  - where,
  - and processing time summary.

3.6.1 Looping phase

Here is an example of the looping phase:

```
Sensor 'B11/D07' #90/156
    /!\ Value superior for B11/D07 in planar error: [0.129 0.129 0.129 0.047 0.047 0.047 0.018 0.018 0.018]
Sensor 'B12/D07' #91/156
    /!\ Value superior for B12/D07 in planar error: [0.129 0.129 0.129 0.047 0.047 0.047 0.018 0.018 0.018]
Sensor 'B01/D08' #92/156
    /!\ Value superior for B01/D08 in planar error: [0.132 0.132 0.132 0.043 0.043 0.043 0.036 0.036 0.036]
Sensor 'B02/D08' #93/156
    /!\ Value superior for B02/D08 in planar error: [0.129 0.129 0.129 0.048 0.048 0.048 0.013 0.013 0.013]
    /!\ Value superior for B02/D08 in altitude shift: [-0. -0. -0. -0. -0. -0.  0. -0.  0.]
Sensor 'B03/D08' #94/156
    /!\ Value superior for B03/D08 in planar error: [0.129 0.129 0.129 0.048 0.048 0.048 0.013 0.013 0.013]
    /!\ Value superior for B03/D08 in altitude shift: [-0. -0. -0. -0. -0. -0.  0. -0.  0.]
Sensor 'B04/D08' #95/156
    /!\ Value superior for B04/D08 in planar error: [0.129 0.129 0.129 0.048 0.048 0.048 0.013 0.013 0.013]
    /!\ Value superior for B04/D08 in altitude shift: [-0. -0. -0. -0. -0. -0.  0. -0. -0.]
Sensor 'B05/D08' #96/156
    /!\ Value superior for B05/D08 in planar error: [0.13  0.13  0.13  0.047 0.047 0.047 0.018 0.018 0.018]
Sensor 'B06/D08' #97/156
    /!\ Value superior for B06/D08 in planar error: [0.13  0.13  0.13  0.047 0.047 0.047 0.018 0.018 0.018]
Sensor 'B07/D08' #98/156
    /!\ Value superior for B07/D08 in planar error: [0.13  0.13  0.13  0.047 0.047 0.047 0.018 0.018 0.018]
    /!\ Value superior for B07/D08 for sun angles: [3.912e-06 3.912e-06 3.912e-06 1.708e-06 0.000e+00 1.207e-06 0.000e+00
       nan 8.538e-07]
Sensor 'B08/D08' #99/156
    /!\ Value superior for B08/D08 in planar error: [0.129 0.129 0.129 0.048 0.048 0.048 0.013 0.013 0.013]
    /!\ Value superior for B08/D08 in altitude shift: [-0. -0. -0. -0. -0. -0.  0. -0.  0.]
    /!\ Value superior for B08/D08 for sun angles: [4.005e-06 3.818e-06 3.818e-06 1.708e-06 1.207e-06 1.207e-06       nan
 0.000e+00 8.538e-07]
Sensor 'B8A/D08' #100/156
    /!\ Value superior for B8A/D08 in planar error: [0.13  0.13  0.13  0.047 0.047 0.047 0.018 0.018 0.018]
Sensor 'B09/D08' #101/156
    /!\ Value superior for B09/D08 in planar error: [0.132 0.132 0.132 0.043 0.043 0.043 0.036 0.036 0.036]
```

The loop over detector/band is clearly visible in black after the ‘Sensor’ log.

Then for several sensor (example B01/D08) the planar error is above threshold for at least one of the 9 points asked. The log provides the differences for the 9 points.

In this log, we can also see that for the Sensor, the altitude shift is above threshold and for others, the Sun angles also (with nan values which are considered as above thresholds).

3.6.2 Summary part

The summary part is divided in 3 parts:

- ✓ The Differences summary is as follows:

```
Max Differences from reference:
  - direct loc:
      - planar dist (m): 0.136 (0.133)
      - altitude (m): 0.00204 (0.00014)
  - inverse loc (pixel): 6e-11 (3.77e-17)
  - sun angles (angular diff (deg)): nan (nan)
  - incidence angles (angular diff (deg)): 0.000395 (0.000345)
  - footprint (ratio): None(None)
  - footprint (max diff in pixel): None(None)
```

  It put for all metrics the maximum distance found and in parenthesis, the CE95 value.

- ✓ The list of sensors that differ from reference:

```
List Sensor that differs from reference:
  - direct loc:
      - planar dist: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
'B11/D12' 'B12/D12']
      - altitude: ['B02/D03' 'B04/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B09/D03' 'B10/D03'
'B11/D03' 'B01/D04' 'B02/D04' 'B05/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
'B10/D04' 'B11/D04' 'B12/D04' 'B05/D05' 'B04/D06' 'B05/D06' 'B10/D06'
'B11/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B08/D07'
'B09/D07' 'B10/D07' 'B02/D08' 'B03/D08' 'B04/D08' 'B08/D08' 'B10/D08'
'B01/D09' 'B02/D09' 'B03/D09' 'B06/D09' 'B07/D09' 'B08/D09' 'B8A/D09'
'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10' 'B08/D10'
'B8A/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11' 'B04/D11' 'B05/D11'
'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B10/D11' 'B11/D11' 'B12/D11'
'B05/D12' 'B06/D12' 'B07/D12' 'B10/D12' 'B11/D12']
  - inverse loc: []
  - sun angles: ['B8A/D02' 'B06/D04' 'B05/D05' 'B06/D05' 'B04/D06' 'B07/D08' 'B08/D08'
'B05/D09' 'B04/D11']
  - incidence angles: []
  - footprint: []
```

For each all metrics, if at least one value of the 9 points is above thresholds, the sensor is stored in the list and log in this part.

- ✓ A summary of the processing times:

```
Execution time for:
  - direct loc: 4.64s
  - inverse loc: 118s
  - sun angles: 0.24s
  - incidence angles: 0.312s

Reference execution time for:
  - direct loc + sun + viewing angles:  3036 ms.
```

The reference processing time is coming from the Legacy reference.txt file (Logs of the Legacy)

If the REFACTORED option is chosen, 2 run are computed, one using the Legacy DEM, and one using the Zarr DEM. All logs are them duplicated, and new ones are also available:

- ✓ « Differences versus ref »: ASGARD refactored using DEM Legacy, versus the references,
- ✓ « Differences zarr versus ref »: ASGARD refactored using DEM in zarr, versus the references,
- ✓ « Differences versus zarr »: ASGARD refactored using DEM Legacy, versus ASGARD refactored using DEM in zarr.

Those comparisons allow to understand if differences to reference come from ASGARD refactored implementation or from the DEM (or Zarr DEM handling).

3.7 Sentinel-2 validation tests list

Possible configurations for already implemented tests in ASGARD V4 are reported in the table below. Please refer to §3.5.3, to know how to launch them.

Table 17 – Possible configuration for S2MSI_TDS

Test ref	Comment
S2MSI_TDS1	✓ S2MSI_TDS1_L0c_DEM_LEGACY ✓ S2MSI_TDS1_L0c_DEM_REFACTORED ✓ S2MSI_TDS1_L0c_CONST_LEGACY ✓ S2MSI_TDS1_L0c_CONST_REFACTORED ✓ S2MSI_TDS1_L1B_DEM_LEGACY ✓ S2MSI_TDS1_L1B_DEM_REFACTORED ✓ S2MSI_TDS1_L1C_DEM_LEGACY ✓ S2MSI_TDS1_L1C_DEM_REFACTORED ✓ S2MSI_TDS1_L1C_DEM_INVLOC_LEGACY ✓ S2MSI_TDS1_L1C_DEM_INVLOC_REFACTORED
S2MSI_TDS2	✓ S2MSI_TDS2_L0c_DEM_LEGACY ✓ S2MSI_TDS2_L0c_DEM_REFACTORED ✓ S2MSI_TDS2_L0c_CONST_LEGACY ✓ S2MSI_TDS2_L0c_CONST_REFACTORED ✓ S2MSI_TDS2_L1B_DEM_LEGACY ✓ S2MSI_TDS2_L1B_DEM_REFACTORED ✓ S2MSI_TDS2_L1C_DEM_LEGACY ✓ S2MSI_TDS2_L1C_DEM_REFACTORED ✓ S2MSI_TDS2_L1C_DEM_INVLOC_LEGACY ✓ S2MSI_TDS2_L1C_DEM_INVLOC_REFACTORED
S2MSI_TDS3	✓ S2MSI_TDS3_L1B_DEM_LEGACY ✓ S2MSI_TDS3_L1B_DEM_REFACTORED ✓ S2MSI_TDS3_L1C_DEM_INVLOC_LEGACY ✓ S2MSI_TDS3_L1C_DEM_INVLOC_REFACTORED
S2MSI_TDS4	✓ S2MSI_TDS4_L1B_DEM_LEGACY ✓ S2MSI_TDS4_L1B_DEM_REFACTORED ✓ S2MSI_TDS4_L1C_DEM_INVLOC_LEGACY ✓ S2MSI_TDS4_L1C_DEM_INVLOC_REFACTORED
S2MSI_TDS5	✓ S2MSI_TDS5_L0u_DEM_LEGACY ✓ S2MSI_TDS5_L0u_DEM_REFACTORED ✓ S2MSI_TDS5_L0c_DEM_LEGACY ✓ S2MSI_TDS5_L0c_DEM_REFACTORED ✓ S2MSI_TDS5_L0c_CONST_LEGACY ✓ S2MSI_TDS5_L0c_CONST_REFACTORED ✓ S2MSI_TDS5_L1B_DEM_LEGACY ✓ S2MSI_TDS5_L1B_DEM_REFACTORED

Test ref	Comment
	✓ S2MSI_TDS5_L1C_DEM_LEGACY ✓ S2MSI_TDS5_L1C_DEM_REFACTORED ✓ S2MSI_TDS5_L1C_DEM_INVLOC_LEGACY ✓ S2MSI_TDS5_L1C_DEM_INVLOC_REFACTORED
S2MSI_TDS6	Not yet available
S2MSI_TDS7	Not yet available
S2MSI_TDS8	Not yet available

3.8 Features not tested

For now, the inverse location grids functionality is not yet tested as it is supposed to be only linked to performances. Indeed, it is first required to have precise inverse location on points before testing it on a large scale with specific cases. It will be tested in further versions, as all degraded cases linked to the refactored implementation using the Zarr DEM (as expressed in sections above).

3.9 Quality tests results

For this version, only the Legacy-based implementation and the Refactor implementation using Legacy DEM is tested. As described in §3, no particular cases are tested in this case, however, to cover all sensor configurations, 2 TDS must be used:

- ✓ S2MSI_TDS1 as small acquisition:
  - ➢ Without L0u model
  - ➢ with refined L1B and L1C model
- ✓ S2MSI_TDS5 as long acquisition:
  - ➢ With L0u model
  - ➢ With L1B and L1C model not refined

3.9.1 S2MSI_TDS1

3.9.1.1 [S2MSI_TDS1_L0c_DEM-010]

[S2MSI_TDS1_L0c_DEM-010]
Summary: *This test aims at assessing that for L0c sensor model using DEM:* ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,*

[S2MSI_TDS1_L0c_DEM-010]

- *Detector footprint generation.*

✓ *There is no large impact on processing time due to the new implementations.*

<u>Actors:</u>

CS GROUP

<u>Expected Results:</u>

Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.

<u>Pass/Fail criteria:</u>

One large increase of time is observed, or another metrics is higher than a defined threshold.

<u>Test Data:</u>

S2MSI_TDS1: Small island

<u>Preconditions:</u>

Reference has been generated with the Legacy processor

#:	Step actions:	Expected Results:	Results:

1	**Run the Legacy-based implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy/tests/test_se
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L0c_DEM_LE`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001,`

		✓ Verify that all metrics except perf are under	**Cf 3.9.1.1.1 - Step 1.2:**

<table>
<tr><th colspan="4">[S2MSI_TDS1_L0c_DEM-010]</th></tr>
<tr>
<td rowspan="2"></td>
<td rowspan="2">ntinel2_msi.py -s -v -k
test_sentinel2_msi[
S2MSI_TDS1_L0c_DEM_LEGACY]

Which represents test configuration:
✓ L0c_DEM_Legacy
✓ on S2MSI_TDS1
✓ with footprint ref
✓ with --legacy</td>
<td>thresholds as described in the logs (cf §3.6 for logs understandings)</td>
<td>Max Differences from reference:
  - direct loc:
    - planar dist (m): 7.37e-09 (5.65e-09)
    - altitude (m): 4.52e-09 (2.42e-13)
  - sun angles (angular diff (deg)): 8.54e-07 (0)
  - incidence angles (angular diff (deg)): 0.0004 (0.000349)
  - inverse loc (pixel): 9.95e-09 (7.1e-10)
  - footprint:
    - ratio: 0.967 (0.967)
    - max dist (pixel): 1.06 (1.06)
✓ Direct loc: OK
✓ Inverse Loc: OK
✓ Sun angles: OK
✓ Incidence angels: OK
✓ Footprint: Not OK but close (~1pixel)
=> PASSED with limitations</td>
</tr>
<tr>
<td>✓ Verify that performances are logged and save results for dedicated performance analysis</td>
<td>✓ Time is monitored and will be analysed in §3.10:
Execution time for:
  - direct loc: 73.9s
  - inverse loc: 1.8s
  - sun_angles: 0.126s
  - incidence_angles: 0.0677s
  - footprint: 13.8s
Reference execution time for:
  - direct loc + sun + viewing angles: 2890 ms.</td>
</tr>
<tr>
<td><u>2</u></td>
<td>Run the Refactored implementation:
Run command:</td>
<td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td>
<td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L0c_DEM_RE

Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001,</td>
</tr>
</table>

<table>
<tr><th colspan="4">[S2MSI_TDS1_L0c_DEM-010]</th></tr>
<tr><td rowspan="2"></td><td rowspan="2">✓ python3 -m pytest asgard-legacy/tests/test_sentinel2_msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS1_L0c_DEM_REFACTORED]

Which represents test configuration:
✓ L0c_DEM_Legacy
✓ on S2MSI_TDS1
✓ with footprint ref
✓ without --legacy</td><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.1.1.2 - Step 2.2:
✓ Legacy DEM
➢ Direct loc: OK with limitation => #204
➢ Inverse Loc: OK
➢ Sun angles: OK
➢ Incidence angels: OK
➢ Footprint: OK, closer than the Legacy-based implementation, which point to an issue of configuration for the legacy-based implementation
✓ Copernicus Zarr DEM not tested => #197
➢ Similar results with GETAS.zarr than using Legacy DEM</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
➢ Using Legacy DEM:
<pre>Execution time for:
    - direct loc: 9.05s
    - inverse loc: 29.4s
    - sun_angles: 0.216s
    - incidence_angles: 0.33s
    - footprint: 283s
Reference execution time for:
    - direct loc + sun + viewing angles: 2890 ms.</pre>➢ Using Zarr DEM:
<pre>Execution time for:
    - direct loc: 30.3s
    - footprint: 276s
Reference execution time for:
    - direct loc + sun + viewing angles: 2890 ms.</pre></td></tr>
</table>

3.9.1.1.1 Step 1.2

Here is the global summary:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 7.37e-09 (5.65e-09)
        - altitude (m): 4.52e-09 (2.42e-13)
    - sun angles (angular diff (deg)): 8.54e-07 (0)
    - incidence angles (angular diff (deg)): 0.0004 (0.000349)
    - inverse loc (pixel): 9.95e-09 (7.1e-10)
    - footprint:
        - ratio: 0.967 (0.967)
        - max dist (pixel): 1.06 (1.06)

List Sensor that differs from reference:
    - footprint: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 73.9s
    - inverse loc: 1.8s
    - sun_angles: 0.126s
    - incidence_angles: 0.0677s
    - footprint: 13.8s
Reference execution time for:
    - direct loc + sun + viewing angles:  2890 ms.

Test run to the end
PASSED
```

Results are good, except for the footprint max dist, which is very close to 1, so acceptable.

3.9.1.1.2 Step 2.2

Here are the global summaries:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 0.394 (0.227)
        - altitude (m): 0.116 (9.83e-06)
    - sun angles (angular diff (deg)): 3.82e-06 (2.26e-06)
    - incidence angles (angular diff (deg)): 0.000401 (0.000349)
    - inverse loc (pixel): 2.82e-06 (3.12e-13)
    - footprint:
        - ratio: 0.967 (0.967)
        - max dist (pixel): 0.0169 (0.0123)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
        - altitude (m): ['B01/D08' 'B02/D08' 'B03/D08' 'B05/D08' 'B06/D08' 'B07/D08' 'B08/D08'
 'B09/D08' 'B11/D08' 'B01/D09' 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09'
 'B06/D09' 'B07/D09' 'B08/D09' 'B09/D09' 'B11/D09' 'B01/D10' 'B03/D10'
 'B05/D10' 'B06/D10' 'B07/D10' 'B09/D10'])


Execution time for:
    - direct loc: 9.05s
    - inverse loc: 29.4s
    - sun_angles: 0.216s
    - incidence_angles: 0.33s
    - footprint: 283s
Reference execution time for:
    - direct loc + sun + viewing angles:  2890 ms.
```

```
Differences zarr versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 30.5 (4.44)
        - altitude (m): 243 (0.565)
    - sun angles (angular diff (deg)): 0.000317 (4.21e-05)
    - incidence angles (angular diff (deg)): 0.000443 (0.000349)
    - inverse loc (pixel): 1.39e-10 (1.21e-14)
    - footprint:
        - ratio: 0.967 (0.967)
        - max dist (pixel): 0.0189 (0.014)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])


Execution time for:
    - direct loc: 30.3s
    - footprint: 276s
Reference execution time for:
    - direct loc + sun + viewing angles:  2890 ms.
```

Using the Legacy DEM, only the “planar dist” is above threshold, with a maximum at 0.394m and a CE95 at 0.227m. Test can then be considered as PASSED with limitations as the quality is far under the Sentinel-2 pixel resolutions, and impact is considered as neglectable.

Regarding processing using Zarr DEM, due to issue #197, the GETAS in zarr is for now used, then results cannot be really compared.

3.9.1.2 [S2MSI_TDS1_L0c_CONST-010]

[S2MSI_TDS1_L0c_CONST-010]
Summary:

[S2MSI_TDS1_L0c_CONST-010]
This test aims at assessing that for L0c sensor model at constant altitude: ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* <u>Actors:</u> *CS GROUP* <u>Expected Results:</u> *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* <u>Pass/Fail criteria:</u> *One large increase of time is observed, or another metrics is higher than a defined threshold.* <u>Test Data:</u> *TDS1: Small island*
<u>Preconditions:</u> Reference has been generated with the Legacy processor

[S2MSI_TDS1_L0c_CONST-010]			
#:	Step actions:	Expected Results:	Results:

1	**Run the Legacy-based implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS1_L0c_ CONST_LEGACY**]
	
	Which represents test configuration:
	✓ L0c_CONST
	✓ on S2MSI_TDS1
	✓ without footprint ref
	✓ with --legacy
	✓ with constant altitude set at 3000
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2
		✓ Verify that the altitude is set at 3000m
			✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L0c_CONST_`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001,`
			`Constant altitude set at: 3000.0m.`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	**Cf 3.9.1.2.1 - Step 1.2:**
			✓ **FAILED:**
			➢ For “planar dist” and “altitude” => #193 raised
			`Differences versus ref`
			
			`Max Differences from reference:`
			`- direct loc:`
			`- planar dist (m): 8.76 (7.9)`
			`- altitude (m): 50.1 (46.8)`
			`- sun angles (angular diff (deg)): 8.7e-05 (7.88e-05)`
			`- incidence angles (angular diff (deg)): 0.000391 (0.000348)`
			`- inverse loc (pixel): 9.7e-09 (7.98e-10)`

		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ Time is monitored and will be analysed in §3.10:
			`Execution time for:`
			`- direct loc: 5.14s`
			`- inverse loc: 0.356s`
			`- sun_angles: 0.116s`
			`- incidence_angles: 0.0652s`
			`Reference execution time for:`
			`- direct loc + sun + viewing angles: 2633 ms.`

2	**Run the Refactored implementation:**	✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**

[S2MSI_TDS1_L0c_CONST-010]			

	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS1_L0c_ CONST_REFACTO RED**]
	
	Which represents test configuration:
	✓ L0c_CONST
	✓ on S2MSI_TDS1
	✓ without footprint ref
	✓ without –legacy
	✓ with constant altitude set at 3000
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L0c_CONST_`
			
			`Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001,`
			`Constant altitude set at: 3000.0m.`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	**Cf 3.9.1.2.2 - Step 2.2:**
			✓ Using DEM Legacy:
			➢ **NOT OK**
			`Differences versus ref`
			`Max Differences from reference:`
			`- direct loc:`
			`- planar dist (m): 8.86 (7.93)`
			`- altitude (m): 50.1 (46.8)`
			`- sun angles (angular diff (deg)): 8.87e-05 (7.91e-05)`
			`- incidence angles (angular diff (deg)): 0.000401 (0.000345)`
			`- inverse loc (pixel): 1.19e+07 (5.8e+06)`
			⇨ #193
			✓ **Copernicus Zarr DEM not tested** => #197
			➢ Results almost identical to DEM Legacy:
			`Differences versus zarr`
			`Max Differences from reference:`
			`- direct loc:`
			`- planar dist (m): 0 (0)`
			`- altitude (m): 0 (0)`
			`- sun angles (angular diff (deg)): 1.21e-06 (8.54e-07)`
			`- incidence angles (angular diff (deg)): 8.54e-07 (8.54e-07)`
			`- inverse loc (pixel): 1.19e+07 (5.8e+06)`

		✓ Verify that performances are logged and save	✓ Time is monitored and will be analysed in §3.10: ➢ Using Legacy DEM:

[S2MSI_TDS1_L0c_CONST-010]			
		results for dedicated performance analysis	Execution time for: - direct loc: 1.33s - inverse loc: 36.6s - sun_angles: 0.214s - incidence_angles: 0.239s Reference execution time for: - direct loc + sun + viewing angles: 2633 ms. ➢ Zarr GETAS => #197 Execution time for: - direct loc: 29.6s Reference execution time for: - direct loc + sun + viewing angles: 2633 ms.

3.9.1.2.1 Step 1.2

Here is the global summary:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 8.76 (7.9)
        - altitude (m): 50.1 (46.8)
    - sun angles (angular diff (deg)): 8.7e-05 (7.88e-05)
    - incidence angles (angular diff (deg)): 0.000391 (0.000348)
    - inverse loc (pixel): 9.7e-09 (7.98e-10)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
```

```
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])


Execution time for:
    - direct loc: 5.14s
    - inverse loc: 0.356s
    - sun_angles: 0.116s
    - incidence_angles: 0.0652s
Reference execution time for:
    - direct loc + sun + viewing angles:  2633 ms.

Test run to the end
PASSED
```

Thresholds for direct location are crossed, it seems to be linked to GEOID altitude.

3.9.1.2.2 Step 2.2

Here is the global summary (only results using DEM Legacy are presented as Zarr ones present similar results and not official zarr is used => #197):

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 8.86 (7.93)
        - altitude (m): 50.1 (46.8)
    - sun angles (angular diff (deg)): 8.87e-05 (7.91e-05)
    - incidence angles (angular diff (deg)): 0.000401 (0.000345)
    - inverse loc (pixel): 1.19e+07 (5.8e+06)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
```

```
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
    - inverse loc: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 1.33s
    - inverse loc: 36.6s
    - sun_angles: 0.214s
    - incidence_angles: 0.239s
Reference execution time for:
    - direct loc + sun + viewing angles:  2633 ms.
```

As in previous step, direct locations present some slight differences, even in altitude, which tends to also point to an issue handling GEOID.

Ticket #193 has been opened.

3.9.1.3 [S2MSI_TDS1_L1B_DEM-010]

[S2MSI_TDS1_L1B_DEM-010]
Summary: *This test aims at assessing that for L1B sensor model using DEM:* ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data:

<table>
<tr><th colspan="4">[S2MSI_TDS1_L1B_DEM-010]</th></tr>
<tr><td colspan="4">TDS1: Small island</td></tr>
<tr><td colspan="4"><u>Preconditions:</u>
Reference has been generated with the Legacy processor</td></tr>
<tr><td><u>#:</u></td><td><u>Step actions:</u></td><td><u>Expected Results:</u></td><td><u>Results:</u></td></tr>
<tr><td rowspan="3"><u>1</u></td><td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS1_L1B_ DEM_LEGACY]

Which represents test configuration:
✓ L1B_DEM_Legacy
✓ on S2MSI_TDS1
✓ with footprint ref
✓ with --legacy</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
<code>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L1B_DEM_LE</code>

<code>Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001,</code></td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.1.3.1 - Step 1.2:
✓ Footprint: Not OK but close (~1pixel)
<pre>Max Differences from reference:
  - direct loc:
     - planar dist (m): 7.8e-09 (5.63e-09)
     - altitude (m): 3.72e-09 (2.48e-13)
  - sun angles (angular diff (deg)): 8.54e-07 (0)
  - incidence angles (angular diff (deg)): 0.000398 (0.000349)
  - inverse loc (pixel): 9.97e-09 (6.25e-10)
  - footprint:
     - ratio: 0.967 (0.967)
     - max dist (pixel): 1.06 (1.06)</pre>=> PASSED with limitations</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:</td></tr>
</table>

[S2MSI_TDS1_L1B_DEM-010]			

			`Execution time for:`
			`    - direct loc: 76.7s`
			`    - inverse loc: 2.13s`
			`    - sun_angles: 0.128s`
			`    - incidence_angles: 0.069s`
			`    - footprint: 14.2s`
			`Reference execution time for:`
			`    - direct loc + sun + viewing angles:  3942 ms.`

2	**Run the Refactored implementation:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS1_L1B_ DEM_REFACTORE D]**
	
	Which represents test configuration:
	✓ S2MSI_TDS1
	✓ with footprint ref
	✓ without --legacy
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L1B_DEM_RE`
			
			`Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001,`
			`Read reference operation text file /DATA/TDS/S2MSI_TDS1/L1B_DEM_Legacy_s2geo_reference.txt`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	**Cf 3.9.1.3.2 - Step 2.2:**
			✓ Using Legacy DEM: **FAILED**
			➢ Direct loc: **NOK** => #200
			➢ Footprint: OK with limitations.
			`Differences versus ref`
			
			`Max Differences from reference:`
			`    - direct loc:`
			`        - planar dist (m): 15 (13.9)`
			`        - altitude (m): 5.82 (0.000361)`
			`    - sun angles (angular diff (deg)): 0.000139 (0.000128)`
			`    - incidence angles (angular diff (deg)): 0.00114 (0.00106)`
			`    - inverse loc (pixel): 2.82e-06 (3.12e-13)`
			`    - footprint:`
			`        - ratio: 0.967 (0.967)`
			`        - max dist (pixel): 1.48 (1.24)`
			✓ **Copernicus Zarr DEM not tested** => #197
			➢ Similar results with GETAS.zarr than using Legacy DEM

		✓ Verify that performances are logged and save	✓ Time is monitored and will be analysed in §3.10: ➢ Using Legacy DEM:

[S2MSI_TDS1_L1B_DEM-010]			

		results for dedicated performance analysis	`Execution time for:`
			`- direct loc: 9.31s`
			`- inverse loc: 29.9s`
			`- sun_angles: 0.208s`
			`- incidence_angles: 0.246s`
			`- footprint: 291s`
			`Reference execution time for:`
			`- direct loc + sun + viewing angles: 3942 ms.`
			➢ Using Zarr DEM
			`Execution time for:`
			`- direct loc: 31.3s`
			`- footprint: 285s`
			`Reference execution time for:`
			`- direct loc + sun + viewing angles: 3942 ms.`

3.9.1.3.1 Step 1.2

Here is the global summary:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 7.8e-09 (5.63e-09)
        - altitude (m): 3.72e-09 (2.48e-13)
    - sun angles (angular diff (deg)): 8.54e-07 (0)
    - incidence angles (angular diff (deg)): 0.000398 (0.000349)
    - inverse loc (pixel): 9.97e-09 (6.25e-10)
    - footprint:
        - ratio: 0.967 (0.967)
        - max dist (pixel): 1.06 (1.06)

List Sensor that differs from reference:
    - footprint: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 76.7s
    - inverse loc: 2.13s
    - sun_angles: 0.128s
    - incidence_angles: 0.069s
    - footprint: 14.2s
Reference execution time for:
    - direct loc + sun + viewing angles:  3942 ms.

Test run to the end
PASSED
```

Results are good, except for the footprint max dist, which is very close to 1, so acceptable.

3.9.1.3.2 Step 2.2

Here is the global summary (only results using DEM Legacy are presented as Zarr ones present similar results and not official zarr is used => #197):

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 15 (13.9)
        - altitude (m): 5.82 (0.000361)
    - sun angles (angular diff (deg)): 0.000139 (0.000128)
    - incidence angles (angular diff (deg)): 0.00114 (0.00106)
    - inverse loc (pixel): 2.82e-06 (3.12e-13)
    - footprint:
        - ratio: 0.967 (0.967)
        - max dist (pixel): 1.48 (1.24)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
        - altitude (m): ['B01/D08' 'B02/D08' 'B03/D08' 'B08/D08' 'B09/D08' 'B12/D08' 'B02/D09'
 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B08/D09' 'B10/D09'])
    - incidence angles: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B02/D02'
 'B03/D02' 'B04/D02' 'B08/D02']
    - footprint: ['B02/D01' 'B03/D01' 'B04/D01' 'B08/D01' 'B02/D02' 'B03/D02' 'B04/D02'
 'B08/D02' 'B02/D03' 'B03/D03' 'B04/D03' 'B08/D03' 'B02/D04' 'B03/D04'
 'B04/D04' 'B08/D04']


Execution time for:
    - direct loc: 9.31s
    - inverse loc: 29.9s
    - sun_angles: 0.208s
    - incidence_angles: 0.246s
    - footprint: 291s
Reference execution time for:
    - direct loc + sun + viewing angles:  3942 ms.
```

Direct locations provide results above threshold that are not neglectable (but not so high). It might be due to refining parameters not well taken into account => #200

3.9.1.4 [S2MSI_TDS1_L1C_DEM-010]

[S2MSI_TDS1_L1C_DEM-010]
Summary:

[S2MSI_TDS1_L1C_DEM-010]
This test aims at assessing that for L1C sensor model using DEM: ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *TDS1: Small island*
Preconditions: Reference has been generated with the Legacy processor

<table>
<tr><th colspan="4">[S2MSI_TDS1_L1C_DEM-010]</th></tr>
<tr><th>#:</th><th>Step actions:</th><th>Expected Results:</th><th>Results:</th></tr>
<tr><td rowspan="3">1</td><td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS1_L1C_ DEM_LEGACY]

Which represents test configuration:
✓ L1C_DEM_Legacy
✓ on S2MSI_TDS1
✓ with footprint ref
✓ with --legacy</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L1C_DEM_LEGACY]
Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>✓ PASSED:
Max Differences from reference:
- direct loc:
- planar dist (m): 7.29e-09 (5.46e-09)
- altitude (m): 3.06e-13 (1.21e-13)
- sun angles (angular diff (deg)): 8.54e-07 (0)
- incidence angles (angular diff (deg)): 0.000393 (0.000349)
- inverse loc (pixel): 9.97e-09 (5.6e-10)

List Sensor that differs from reference:

Test run to the end
PASSED</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
Execution time for:
- direct loc: 28.3s
- inverse loc: 1.36s
- sun_angles: 0.122s
- incidence_angles: 0.0717s
Reference execution time for:
- direct loc + sun + viewing angles: 3136 ms.</td></tr>
</table>

<table>
<tr><th colspan="4">[S2MSI_TDS1_L1C_DEM-010]</th></tr>
<tr>
<td rowspan="3">2</td>
<td rowspan="3">Run the Refactored implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS1_L1C_ DEM_REFACTORE D]

Which represents test configuration:
✓ on S2MSI_TDS1
✓ with footprint ref
✓ without --legacy</td>
<td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td>
<td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L1C_DEM_REFACTORED]

Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td>
</tr>
<tr>
<td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td>
<td>Cf 3.9.1.4.1 - Step 2.2:
✓ Using Legacy DEM: FAILED
➢ Direct locations far from reference => #200
✓ Differences between Zarr and Legacy DEM are small, which is encouraging even if the GETAS was used instead of Copernicus Zarr DEM => #197

Differences versus zarr

Max Differences from reference:
- direct loc:
- planar dist (m): 0.115 (0.06)
- altitude (m): 0.533 (0.364)
- sun angles (angular diff (deg)): 1.21e-06 (8.54e-07)
- incidence angles (angular diff (deg)): 1.21e-06 (8.54e-07)
- inverse loc (pixel): 2.12e-12 (6.64e-19)</td>
</tr>
<tr>
<td>✓ Verify that performances are logged and save results for dedicated performance analysis</td>
<td>✓ Time is monitored and will be analysed in §3.10:
➢ Using Legacy DEM:

Execution time for:
- direct loc: 5.5s
- inverse loc: 35.1s
- sun_angles: 0.208s
- incidence_angles: 0.246s
Reference execution time for:
- direct loc + sun + viewing angles: 3136 ms.

➢ Using Zarr DEM:</td>
</tr>
</table>

[S2MSI_TDS1_L1C_DEM-010]			
			Execution time for: - direct loc: 29.5s Reference execution time for: - direct loc + sun + viewing angles: 3136 ms.

3.9.1.4.1 Step 2.2

Here are the summaries:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 15 (13.8)
        - altitude (m): 0.000301 (0.000232)
    - sun angles (angular diff (deg)): 0.000139 (0.000125)
    - incidence angles (angular diff (deg)): 0.00114 (0.00105)
    - inverse loc (pixel): 2.12e-12 (6.64e-19)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
    - incidence angles: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B02/D02'
 'B03/D02' 'B04/D02' 'B08/D02']


Execution time for:
    - direct loc: 5.5s
    - inverse loc: 35.1s
    - sun_angles: 0.208s
    - incidence_angles: 0.246s
Reference execution time for:
    - direct loc + sun + viewing angles:  3136 ms.
```

```
Differences zarr versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 14.9 (13.7)
        - altitude (m): 0.575 (0.486)
    - sun angles (angular diff (deg)): 0.000138 (0.000125)
    - incidence angles (angular diff (deg)): 0.00114 (0.00105)
    - inverse loc (pixel): 2.12e-12 (6.64e-19)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
```

```
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
    - incidence angles: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B02/D02'
 'B03/D02' 'B04/D02' 'B08/D02']


Execution time for:
    - direct loc: 29.5s
Reference execution time for:
    - direct loc + sun + viewing angles:  3136 ms.
```

Direct locations provide results above threshold that are not neglectable (but not so high). It might be due to refining parameters not well taken into account => #200

Differences between Zarr and Legacy DEM are small, which is encouraging even if the GETAS was used instead of Copernicus Zarr DEM.

3.9.1.5 [S2MSI_TDS1_L1C_DEM_INVLOC-010]

[S2MSI_TDS1_L1C_DEM_INVLOC-010]
Summary: *This test aims at assessing that for L1C sensor model using DEM:* ✓ *ASGARD gives equivalent results for inverse location grids,* ✓ *There is no large impact on processing time due to the new implementations.*

[S2MSI_TDS1_L1C_DEM_INVLOC-010]

Actors:

CS GROUP

Expected Results:

Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.

Pass/Fail criteria:

One large increase of time is observed, or another metrics is higher than a defined threshold.

Test Data:

TDS1: Small island

Preconditions:

Reference has been generated with the Legacy processor

#:	Step actions:	Expected Results:	Results:

1	**Run the Legacy-based implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L1C_DEM_INVLOC_LEGACY]`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}`

		✓ Verify that all metrics except perf are under thresholds as described in the logs	✓ **OK:**

<table>
<tr><th colspan="4">[S2MSI_TDS1_L1C_DEM_INVLOC-010]</th></tr>
<tr><td rowspan="2"></td><td rowspan="2">test_sentinel2_msi[
S2MSI_TDS1_L1C_DEM_INVOLOC_LEGACY]</td><td>(cf §3.6 for logs understandings)</td><td>Differences versus ref

Max Differences from reference:
- inverse loc (pixel): 1.38e-08 (6.06e-18)

List Sensor that differs from reference:

Execution time for:
- inverse_loc: 2.88e+03s
Reference execution time for:
- inverse loc: 1001964 ms.

Test run to the end
PASSED</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
Execution time for:
- inverse_loc: 2.88e+03s
Reference execution time for:
- inverse loc: 1001964 ms.</td></tr>
<tr><td rowspan="2"><u>2</u></td><td rowspan="2">Run the Refactored implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L1C_DEM_INVLOC_REFACTORED]

Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.1.5.1 - Step 2.2:
✓ FAILED:
➢ Inverse locations a little far from reference => <u>#203</u>
➢ Test not ending => <u>#208</u>
✓ Copernicus Zarr DEM not tested => <u>#197</u></td></tr>
</table>

[S2MSI_TDS1_L1C_DEM_INVLOC-010]			
	S2MSI_TDS1_L1C_DEM_INVLOC_REFACTORED]		➢ Similar results with GETAS.zarr than using Legacy DEM
		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ Test not ending, no time available

3.9.1.5.1 Step 2.2

Here is the start of the looping log:

```
Sensor 'B12/D03' #39/156
    Number of inverse locations: 24055
    Compute inverse loc grid.
      time ~ 98.40799434296787s
      ! Value superior for B12/D03 in inverse_loc error: [0.193 0.193 0.193 0.193 0.194 0.193 0.193 0.193 0.194 0.194] [...]
    Compute inverse loc grid with Zarr.
      time ~ 94.37622237950563s
      ! Value superior for B12/D03 in inverse_loc error: [0.193 0.193 0.192 0.192 0.193 0.193 0.193 0.193 0.193 0.193] [...]
    Compute differences of previous steps.
Sensor 'B01/D04' #40/156
    Number of inverse locations: 318125
    Compute inverse loc grid.
      time ~ 1356.8905342957005s
      ! Value superior for B01/D04 in inverse_loc error: [0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023] [...]
    Compute inverse loc grid with Zarr.
      time ~ 1339.333953739144s
      ! Value superior for B01/D04 in inverse_loc error: [0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023 0.023] [...]
    Compute differences of previous steps.
Sensor 'B02/D04' #41/156
    Number of inverse locations: 5068940
    Compute inverse loc grid.
  ! AttributError(type(exp)=<class 'AttributeError'>): inverse loc 'NoneType' object has no attribute 'shiftedBy'
      time ~ 38225.2270217957s
FAILED

================================================================================================= FAILURES ============================
_______________________________________________________________________ test_sentinel2_msi[S2MSI_TDS1_L1C_DEM_INVLOC_REFACTORED] ____
```

[…]

```
>       return np.square((x_first - x_second) ** 2 + (y_first - y_second) ** 2)
E       ValueError: operands could not be broadcast together with shapes (1013788,) (63625,)

asgard-legacy/tests/helpers/compare.py:316: ValueError
=============================================================================================== warnings summary ================================================
tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L1C_DEM_INVLOC_REFACTORED]
tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS1_L1C_DEM_INVLOC_REFACTORED]
  /usr/local/lib/python3.11/site-packages/asgard/products/sentinel2/msi.py:697: DeprecationWarning: Support of native DEM will be removed, use Zarr versions
    self.propagation_model = PropagationModel(**model_kwargs)

-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
```

2 issues can be observed:

- ✓ Inverse locations are a little far from references => #203
- ✓ ValueError from a NoneType returned => #208

3.9.2 S2MSI_TDS2

3.9.2.1 [S2MSI_TDS2_L0c_DEM-010]

[S2MSI_TDS2_L0c_DEM-010]
Summary: *This test aims at assessing that for L0c sensor model using DEM:* ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.*

[S2MSI_TDS2_L0c_DEM-010]
Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *S2MSI_TDS2: Antemeridian*
Preconditions: Reference has been generated with the Legacy processor

#:	Step actions:	Expected Results:	Results:

1	**Run the Legacy-based implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L0c_DEM_LE`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001,`

		✓ Verify that all metrics except perf are under thresholds as described in the logs	**Cf 3.9.2.1.1 - Step 1.2:** ✓ **FAILED => #198**

[S2MSI_TDS2_L0c_DEM-010]			
	S2MSI_TDS2_L0c_DEM_LEGACY] Which represents test configuration: ✓ L0c_DEM_Legacy ✓ on S2MSI_TDS2 ✓ with footprint ref ✓ with --legacy	(cf §3.6 for logs understandings)	
		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ Test with errors, not representative.

2	**Run the Refactored implementation:**
	Run command:
	✓ python3 -m pytest asgard-legacy/tests/test_sentinel2_msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS2_L0c_DEM_REFACTORED**]
	
	Which represents test configuration:
	✓ L0c_DEM_Legacy
	✓ on S2MSI_TDS2
	✓ with footprint ref
	✓ without --legacy
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L0c_DEM_REFACTORED]`
			`Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	**Cf 3.9.2.1.2 - Step 2.2:** ✓ Using Legacy DEM: **FAILED** ➢ Direct locations a little far from reference => #204 ➢ Shapely error on footprints => #198 ➢ Test is failing due to RAM issue => #199 ✓ **Copernicus Zarr DEM not tested** => #197 ➢ Similar results with GETAS.zarr than using Legacy DEM
		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ **Test not ended.**

3.9.2.1.1 Step 1.2

Here is the global summary:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 5.55e-09 (2.81e-09)
        - altitude (m): 6.32e-10 (7.71e-14)
    - sun angles (angular diff (deg)): 1.21e-06 (8.54e-07)
    - incidence angles (angular diff (deg)): 0.000394 (0.000352)
    - inverse loc (pixel): 8.5e-08 (4.97e-11)
    - footprint:
        - ratio: 0.994 (0.993)
        - max dist (pixel): 1.05 (1.05)

List Sensor that differs from reference:
    - footprint: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D12' 'B02/D12' 'B03/D12'
 'B04/D12' 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12'
 'B10/D12' 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 109s
    - inverse loc: 7.44s
    - sun_angles: 0.173s
    - incidence_angles: 0.068s
    - footprint: 428s
Reference execution time for:
    - direct loc + sun + viewing angles:  32376 ms.

  Error list:
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side l
r if the input geometry is invalid.
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side l
```

Results are good, except for the footprint max dist, which is very close to 1, so acceptable. However, detector 4 to 11 (included) raised errors for all detectors:

```
Sensor 'B12/D03' #39/156
  Check versus reference
      ! Value superior for B12/D03 for footprint: [9.93338778e-01 1.05192262e+00 1.05192262e+00 3.33918494e-04]
Sensor 'B01/D04' #40/156
  Check versus reference
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at 179.73988145092235 -13.93510921402007. This can occu
r if the input geometry is invalid.
Sensor 'B02/D04' #41/156
  Check versus reference
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at 179.74072324717599 -13.935782405466142. This can occ
ur if the input geometry is invalid.
Sensor 'B03/D04' #42/156
```

[…]

```
cur if the input geometry is invalid.
Sensor 'B11/D11' #142/156
  Check versus reference
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at -179.72605338460841 -19.714648199032887. This can oc
cur if the input geometry is invalid.
Sensor 'B12/D11' #143/156
  Check versus reference
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at -179.72641043503873 -19.715000549281648. This can oc
cur if the input geometry is invalid.
Sensor 'B01/D12' #144/156
  Check versus reference
      ! Value superior for B01/D12 for footprint: [0.99332535 1.05167026 1.05167026 0.        ]
Sensor 'B02/D12' #145/156
  Check versus reference
      ! Value superior for B02/D12 for footprint: [9.93528249e-01 1.05171161e+00 1.05171161e+00 2.20719381e-05]
Sensor 'B03/D12' #146/156
  Check versus reference
      ! Value superior for B03/D12 for footprint: [0.99342203 1.05225931 1.05225931 0.        ]
Sensor 'B04/D12' #147/156
```

Shapely error => #198

3.9.2.1.2 Step 2.2

Here is the end of the looping log:

```
  Check versus reference
      ! Value superior for B07/D10 in planar error: [0.054 0.054 0.054 0.108 0.109 0.109 0.249 0.25  0.252]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at -179.71295209153186 -18.623293854985175. This can occur if the input geometry is invalid.
  Check zarr versus reference
      ! Value superior for B07/D10 in planar error: [0.062 0.094 0.144 0.157 0.162 0.165 0.165 0.139 0.117]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: Input geom 0 is invalid: Invalid Coordinate at nan nan
Sensor 'B08/D10' #125/156
  Check versus reference
      ! Value superior for B08/D10 in planar error: [0.042 0.042 0.042 0.03  0.031 0.031 0.041 0.043 0.044]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at 179.74222596046974 -19.866236331819383. This can occur if the input geometry is invalid.
  Check zarr versus reference
      ! Value superior for B08/D10 in planar error: [0.101 0.138 0.191 0.072 0.077 0.08  0.108 0.134 0.158]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: Input geom 0 is invalid: Invalid Coordinate at nan nan
Sensor 'B8A/D10' #126/156
  Check versus reference
      ! Value superior for B8A/D10 in planar error: [0.017 0.015 0.013 0.029 0.03  0.032 0.06  0.061 0.062]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at 179.74291980062657 -19.867568878565674. This can occur if the input geometry is invalid.
  Check zarr versus reference
      ! Value superior for B8A/D10 in planar error: [0.102 0.14  0.193 0.073 0.077 0.081 0.096 0.121 0.144]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: Input geom 0 is invalid: Invalid Coordinate at nan nan
Sensor 'B09/D10' #127/156
Killed
[asgard  2024-03-28 18:17:10  /home/aburie]
```

In conclusion, 3 issues can be observed:

- ✓ Direct locations a little far from reference => #204
- ✓ Shapely error on footprints => #198
- ✓ Test is failing due to RAM issue as the RAM is going up and up between loops over sensors => #199

3.9.2.2 [S2MSI_TDS2_L0c_CONST-010]

Test is failing due to RAM issue as the RAM is going up and up between loops over sensors => #199 [S2MSI_TDS2_L0c_CONST-010]

Summary:
 	
This test aims at assessing that for L0c sensor model at constant altitude:
 	
✓ *ASGARD gives equivalent results for*
    ➢ *Direct locations,*
    ➢ *Inverse locations,*

Test is failing due to RAM issue as the RAM is going up and up between loops over sensors => #199 [S2MSI_TDS2_L0c_CONST-010]			
➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* <u>Actors:</u> *CS GROUP* <u>Expected Results:</u> *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* <u>Pass/Fail criteria:</u> *One large increase of time is observed, or another metrics is higher than a defined threshold.* <u>Test Data:</u> *TDS2: Antemeridian*			
<u>Preconditions:</u> Reference has been generated with the Legacy processor			
<u>#:</u>	<u>Step actions:</u>	<u>Expected Results:</u>	<u>Results:</u>
<u>1</u>		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**

Test is failing due to RAM issue as the RAM is going up and up between loops over sensors => #199 [S2MSI_TDS2_L0c_CONST-010]			

	Run the Legacy-based implementation with Legacy DEM:
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS2_L0c_ CONST_LEGACY**]
	
	Which represents test configuration:
	✓ L0c_CONST
	✓ on S2MSI_TDS2
	✓ without footprint ref
	✓ with --legacy
	✓ with constant altitude set at 3000
		✓ Verify that the altitude is set at 3000m	`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L0c_CONST_`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001`
			`Constant altitude set at: 3000.0m.`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	**Cf 3.9.2.2.1 -Step 1.2:**
			✓ **FAILED =>** #193 raised
			`Max Differences from reference:`
			`- direct loc:`
			`- planar dist (m): 11.8 (9.87)`
			`- altitude (m): 56.2 (55.1)`
			`- sun angles (angular diff (deg)): 0.000108 (9.08e-05)`
			`- incidence angles (angular diff (deg)): 0.000395 (0.000354)`
			`- inverse loc (pixel): 8.49e-08 (1.43e-11)`

		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ Time is monitored and will be analysed in §3.10:
			`Execution time for:`
			`- direct loc: 10.9s`
			`- inverse loc: 0.688s`
			`- sun_angles: 0.166s`
			`- incidence_angles: 0.0691s`
			`Reference execution time for:`
			`- direct loc + sun + viewing angles: 2324 ms.`

2	**Run the Refactored implementation:**
	Run command:
	✓ python3 -m pytest asgard-legacy
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L0c_CONST_`
			`Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001,`
			`Constant altitude set at: 3000.0m.`

<table>
<tr><td colspan="4">Test is failing due to RAM issue as the RAM is going up and up between loops over sensors => #199
[S2MSI_TDS2_L0c_CONST-010]</td></tr>
<tr><td rowspan="2"></td><td rowspan="2">/tests/test_sentinel2_msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS2_L0c_CONST_REFACTORED]

Which represents test configuration:
✓ L0c_CONST
✓ on S2MSI_TDS2
✓ without footprint ref
✓ without –legacy
✓ with constant altitude set at 3000</td><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.2.2.2-Step 2.2:
✓ Using Legacy DEM: FAILED
➢ Altitude handling => #193
➢ Inverse location issues => #177
✓ Copernicus Zarr DEM not tested => #197
➢ Similar results with GETAS.zarr than using Legacy DEM</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
➢ Using Legacy DEM
<pre>Execution time for:
    - direct loc: 1.33s
    - inverse loc: 81.3s
    - sun_angles: 0.239s
    - incidence_angles: 0.283s
Reference execution time for:
    - direct loc + sun + viewing angles: 2324 ms.</pre>➢ Using GETAS.zarr
<pre>Execution time for:
    - direct loc: 37.7s
Reference execution time for:
    - direct loc + sun + viewing angles: 2324 ms.</pre></td></tr>
</table>

3.9.2.2.1 Step 1.2

Here is the global summary:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 11.8 (9.87)
        - altitude (m): 56.2 (55.1)
    - sun angles (angular diff (deg)): 0.000108 (9.08e-05)
    - incidence angles (angular diff (deg)): 0.000395 (0.000354)
    - inverse loc (pixel): 8.49e-08 (1.43e-11)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
```

```
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])


Execution time for:
    - direct loc: 10.9s
    - inverse loc: 0.688s
    - sun_angles: 0.166s
    - incidence_angles: 0.0691s
Reference execution time for:
    - direct loc + sun + viewing angles:  2324 ms.

Test run to the end
PASSED
```

All tests passed without error; hence the problem seems to really be on DEM handling for the rest of this TDS. However, thresholds for direct location are crossed, it seems to be linked to GEOID altitude. #193 raised

3.9.2.2.2 Step 2.2

Here are the summaries:

```
Max Differences from reference:
    - direct loc:
        - planar dist (m): 11.9 (9.93)
        - altitude (m): 56.2 (55.1)
    - sun angles (angular diff (deg)): 0.000108 (9.15e-05)
    - incidence angles (angular diff (deg)): 0.000412 (0.000356)
    - inverse loc (pixel): 2.29e+26 (5.81e+06)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
```

```
    - inverse loc: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 1.33s
    - inverse loc: 81.3s
    - sun_angles: 0.239s
    - incidence_angles: 0.283s
Reference execution time for:
    - direct loc + sun + viewing angles:  2324 ms.
```

```
Differences zarr versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 11.9 (9.93)
        - altitude (m): 56.2 (55.1)
    - sun angles (angular diff (deg)): 0.000108 (9.15e-05)
    - incidence angles (angular diff (deg)): 0.000412 (0.000356)
    - inverse loc (pixel): 2.29e+26 (5.8e+06)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
    - inverse loc: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 37.7s
Reference execution time for:
    - direct loc + sun + viewing angles:  2324 ms.
```

Here we can see that the direct locations are impacted the same way that the Legacy-base implementation, probably not taking into account the GEOID. #193

However, the inverse locations are far from expected results! #177

3.9.2.3 [S2MSI_TDS2_L1B_DEM-010]

[S2MSI_TDS2_L1B_DEM-010]
Summary:

[S2MSI_TDS2_L1B_DEM-010]
This test aims at assessing that for L1B sensor model using DEM: ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *TDS2: Antemeridian*
Preconditions: Reference has been generated with the Legacy processor

[S2MSI_TDS2_L1B_DEM-010]			
#:	Step actions:	Expected Results:	Results:

1	**Run the Legacy-based implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS2_L1B_ DEM_LEGACY**]
	
	Which represents test configuration:
	✓ L1B_DEM_Legacy
	✓ on S2MSI_TDS2
	✓ with footprint ref
	✓ with --legacy
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1B_DEM_LEGACY]`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	**Cf 3.9.2.3.1 - Step 1.2:** ✓ **FAILED:** ➢ Footprints over thresholds but very close ➢ Shapely error on footprints => #198
		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ Test with errors, not representative.

2	**Run the Refactored implementation:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS2_L1B_**
		Using DEM Legacy part:
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2
		✓ Verify that all metrics except perf are under thresholds as described in the logs
			✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1B_DEM_RE`
			`Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001,`
			
			Cf 3.9.2.3.2 - Step 2.2:
			✓ Using Legacy DEM: **FAILED**
			➢ Shapely error on footprints => #198
			➢ Test is failing due to RAM => #199

[S2MSI_TDS2_L1B_DEM-010]			
	DEM_REFACTORED] Which represents test configuration: ✓ S2MSI_TDS2 ✓ with footprint ref ✓ without --legacy	(cf §3.6 for logs understandings) ✓ Verify that performances are logged and save results for dedicated performance analysis	✓ **Copernicus Zarr DEM not tested** => #197 ➢ Similar results with GETAS.zarr than using Legacy DEM
		Using Zarr Part: ✓ Verify that thresholds are the ones from the metrics table in §2.3.2 ✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings) ✓ Verify that performances are logged and save results for dedicated performance analysis	✓ **Test not ending.**

3.9.2.3.1 Step 1.2

Here is the global summary:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 5.81e-09 (2.76e-09)
        - altitude (m): 4.79e-10 (7.71e-14)
    - sun angles (angular diff (deg)): 8.54e-07 (8.54e-07)
    - incidence angles (angular diff (deg)): 0.000394 (0.000352)
    - inverse loc (pixel): 8.5e-08 (4.94e-11)
    - footprint:
        - ratio: 0.994 (0.993)
        - max dist (pixel): 1.05 (1.05)

List Sensor that differs from reference:
    - footprint: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D12' 'B02/D12' 'B03/D12'
 'B04/D12' 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12'
 'B10/D12' 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 106s
    - inverse loc: 8.27s
    - sun_angles: 0.127s
    - incidence_angles: 0.0678s
    - footprint: 417s
Reference execution time for:
    - direct loc + sun + viewing angles:  5194 ms.

  Error list:
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side lo
```

Results are good, except for the footprint max dist, which is very close to 1, so acceptable. However, detector 4 to 11 (included) raised errors for all detectors:

```
Sensor 'B11/D03' #38/156
  Check versus reference
      ! Value superior for B11/D03 for footprint: [9.93385122e-01 1.05315976e+00 1.05315976e+00 7.36425797e-05]
Sensor 'B12/D03' #39/156
  Check versus reference
      ! Value superior for B12/D03 for footprint: [9.93338759e-01 1.05192867e+00 1.05192867e+00 3.33918351e-04]
Sensor 'B01/D04' #40/156
  Check versus reference
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at 179.88337560430898 -13.316281279675071. This can occ
ur if the input geometry is invalid.
Sensor 'B02/D04' #41/156
  Check versus reference
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at 179.88416637505142 -13.317749098561004. This can occ
ur if the input geometry is invalid.
Sensor 'B03/D04' #42/156
  Check versus reference
```

[…]

```
cur if the input geometry is invalid.
Sensor 'B11/D11' #142/156
  Check versus reference
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at -179.72606846716425 -19.714620626595167. This can oc
cur if the input geometry is invalid.
Sensor 'B12/D11' #143/156
  Check versus reference
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at -179.72642625076605 -19.714973071095081. This can oc
cur if the input geometry is invalid.
Sensor 'B01/D12' #144/156
  Check versus reference
      ! Value superior for B01/D12 for footprint: [0.99332534 1.05167648 1.05167648 0.        ]
Sensor 'B02/D12' #145/156
  Check versus reference
```

Issues observed:

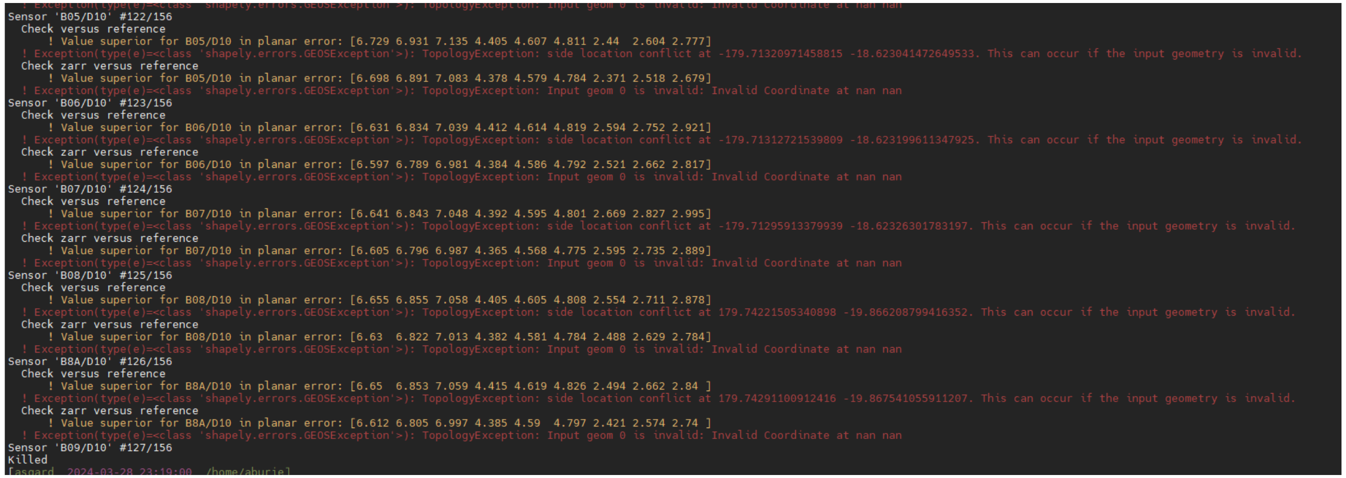
- ✓ Footprint a little far from reference
- ✓ Shapely error for footprints => #198
- ✓ Test is failing due to RAM => #199

3.9.2.3.2 Step 2.2

Here is the start of the looping log:

```
Read reference operation text file /DATA/TDS/S2MSI_TDS2/L1B_DEM_Legacy_s2geo_reference.txt
Sensor 'B01/D01' #1/156
  Check versus reference
      ! Value superior for B01/D01 in planar error: [3.469 3.652 3.836 1.996 2.052 2.027 3.5   3.36  3.233]
      ! Value superior for B01/D01 in altitude error: [0.    0.    0.    0.    0.    0.599 0.    0.    0.   ]
  Check zarr versus reference
      ! Value superior for B01/D01 in planar error: [ 3.602  3.749  3.908  2.184  2.237 10.238  3.659  3.497  3.331]
  Check versus zarr
      ! Value superior for B01/D01 in planar error: [0.217 0.162 0.122 0.19  0.189 8.273 0.218 0.182 0.127]
      ! Value superior for B01/D01 in altitude error: [ 1.031  0.845  0.71   0.904  0.988 48.     1.037  0.954  0.737]
Sensor 'B02/D01' #2/156
  Check versus reference
      ! Value superior for B02/D01 in planar error: [3.277 3.473 3.668 1.737 1.812 1.813 3.046 2.894 2.756]
      ! Value superior for B02/D01 in altitude error: [ 0.     0.    -0.     0.     0.     0.641  0.     0.     0.   ]
  Check zarr versus reference
      ! Value superior for B02/D01 in planar error: [3.372 3.54  3.717 1.918 1.986 5.208 3.206 3.034 2.858]
  Check versus zarr
      ! Value superior for B02/D01 in planar error: [0.214 0.16  0.122 0.188 0.187 3.554 0.214 0.179 0.126]
      ! Value superior for B02/D01 in altitude error: [ 1.027  0.844  0.715  0.903  0.987 20.875  1.028  0.947  0.738]
Sensor 'B03/D01' #3/156
  Check versus reference
      ! Value superior for B03/D01 in planar error: [3.279 3.47  3.661 1.877 1.95  1.935 2.911 2.747 2.594]
      ! Value superior for B03/D01 in altitude error: [0.    0.    0.    0.    0.    0.696 0.    0.    0.   ]
  Check zarr versus reference
      ! Value superior for B03/D01 in planar error: [3.384 3.545 3.716 2.06  2.126 7.674 3.055 2.873 2.686]
  Check versus zarr
      ! Value superior for B03/D01 in planar error: [0.215 0.16  0.122 0.188 0.187 5.896 0.215 0.18  0.126]
      ! Value superior for B03/D01 in altitude error: [ 1.028  0.845  0.715  0.903  0.987 34.585  1.033  0.952  0.739]
Sensor 'B04/D01' #4/156
  Check versus reference
      ! Value superior for B04/D01 in planar error: [3.285 3.471 3.658 1.923 1.984 2.035 3.153 3.009 2.877]
      ! Value superior for B04/D01 in altitude error: [0.    0.    0.    0.    0.    0.189 0.    0.    0.   ]
  Check zarr versus reference
      ! Value superior for B04/D01 in planar error: [3.407 3.558 3.723 2.109 2.165 9.306 3.314 3.149 2.978]
  Check versus zarr
      ! Value superior for B04/D01 in planar error: [0.215 0.161 0.122 0.189 0.187 7.377 0.216 0.181 0.126]
```

But some tests are passing, but are still above thresholds, and overall, the test fails:



```
Sensor 'B05/D10' #122/156
  Check versus reference
      ! Value superior for B05/D10 in planar error: [6.729 6.931 7.135 4.405 4.607 4.811 2.44  2.604 2.777]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at -179.71320971458815 -18.623041472649533. This can occur if the input geometry is invalid.
  Check zarr versus reference
      ! Value superior for B05/D10 in planar error: [6.698 6.891 7.083 4.378 4.579 4.784 2.371 2.518 2.679]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: Input geom 0 is invalid: Invalid Coordinate at nan nan
Sensor 'B06/D10' #123/156
  Check versus reference
      ! Value superior for B06/D10 in planar error: [6.631 6.834 7.039 4.412 4.614 4.819 2.594 2.752 2.921]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at -179.71312721539809 -18.623199611347925. This can occur if the input geometry is invalid.
  Check zarr versus reference
      ! Value superior for B06/D10 in planar error: [6.597 6.789 6.981 4.384 4.586 4.792 2.521 2.662 2.817]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: Input geom 0 is invalid: Invalid Coordinate at nan nan
Sensor 'B07/D10' #124/156
  Check versus reference
      ! Value superior for B07/D10 in planar error: [6.641 6.843 7.048 4.392 4.595 4.801 2.669 2.827 2.995]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at -179.71295913379939 -18.62326301783197. This can occur if the input geometry is invalid.
  Check zarr versus reference
      ! Value superior for B07/D10 in planar error: [6.605 6.796 6.987 4.365 4.568 4.775 2.595 2.735 2.889]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: Input geom 0 is invalid: Invalid Coordinate at nan nan
Sensor 'B08/D10' #125/156
  Check versus reference
      ! Value superior for B08/D10 in planar error: [6.655 6.855 7.058 4.405 4.605 4.808 2.554 2.711 2.878]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at 179.74221505340898 -19.866208799416352. This can occur if the input geometry is invalid.
  Check zarr versus reference
      ! Value superior for B08/D10 in planar error: [6.63  6.822 7.013 4.382 4.581 4.784 2.488 2.629 2.784]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: Input geom 0 is invalid: Invalid Coordinate at nan nan
Sensor 'B8A/D10' #126/156
  Check versus reference
      ! Value superior for B8A/D10 in planar error: [6.65  6.853 7.059 4.415 4.619 4.826 2.494 2.662 2.84 ]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: side location conflict at 179.74291100912416 -19.867541055911207. This can occur if the input geometry is invalid.
  Check zarr versus reference
      ! Value superior for B8A/D10 in planar error: [6.612 6.805 6.997 4.385 4.59  4.797 2.421 2.574 2.74 ]
  ! Exception(type(e)=<class 'shapely.errors.GEOSException'>): TopologyException: Input geom 0 is invalid: Invalid Coordinate at nan nan
Sensor 'B09/D10' #127/156
Killed
```

Several issues:

- ✓ Shapely error on footprints => #198
- ✓ Test is failing due to RAM issue as the RAM is going up and up between loops over sensors => #199

3.9.2.4 [S2MSI_TDS2_L1C_DEM-010]

[S2MSI_TDS2_L1C_DEM-010]
Summary: *This test aims at assessing that for L1C sensor model using DEM:*

[S2MSI_TDS2_L1C_DEM-010]			
✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *TDS2: Antemeridian*			
Preconditions: Reference has been generated with the Legacy processor			
#:	Step actions:	Expected Results:	Results:

<table>
<tr><th colspan="4">[S2MSI_TDS2_L1C_DEM-010]</th></tr>
<tr>
<td rowspan="3">1</td>
<td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS2_L1C_ DEM_LEGACY]

Which represents test configuration:
✓ L1C_DEM_Legacy
✓ on S2MSI_TDS2
✓ with footprint ref
✓ with --legacy</td>
<td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td>
<td>✓ OK:
<code>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1C_DEM_LEGACY]</code>
<code>Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</code></td>
</tr>
<tr>
<td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td>
<td>✓ PASSED:
<pre>Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 6.3e-09 (2.64e-09)
        - altitude (m): 3.47e-10 (9.95e-14)
    - sun angles (angular diff (deg)): 1.21e-06 (8.54e-07)
    - incidence angles (angular diff (deg)): 0.000395 (0.000349)
    - inverse loc (pixel): 8.72e-09 (6.2e-11)

List Sensor that differs from reference:

Execution time for:
    - direct loc: 95.7s
    - inverse loc: 7.66s
    - sun_angles: 0.139s
    - incidence_angles: 0.067s
Reference execution time for:
    - direct loc + sun + viewing angles: 5283 ms.

Test run to the end
PASSED</pre></td>
</tr>
<tr>
<td>✓ Verify that performances are logged and save results for dedicated performance analysis</td>
<td>✓ Time is monitored and will be analysed in §3.10:</td>
</tr>
</table>

[S2MSI_TDS2_L1C_DEM-010]			

			`Execution time for:`
			`    - direct loc: 95.7s`
			`    - inverse loc: 7.66s`
			`    - sun_angles: 0.139s`
			`    - incidence_angles: 0.067s`
			`Reference execution time for:`
			`    - direct loc + sun + viewing angles:  5283 ms.`

2	**Run the Refactored implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS2_L1C_ DEM_REFACTORE D**]
	
	Which represents test configuration:
	✓ on S2MSI_TDS2
	✓ with footprint ref
	✓ without --legacy
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1C_DEM_REFACTORED]`
			`Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	**Cf 3.9.2.4.1 - Step 2.2:** ✓ Using Legacy DEM: **FAILED** ➢ Direct locations give results above thresholds => #200 ➢ Test fails before end => #201 ✓ **Copernicus Zarr DEM not tested** => #197 ➢ Similar results with GETAS.zarr than using Legacy DEM
		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ **Test not ending**

3.9.2.4.1 Step 2.2

Here is the start of the looping log:

```
Sensor 'B01/D01' #1/156
  Check versus reference
      ! Value superior for B01/D01 in planar error: [3.469 3.652 3.836 2.057 2.147 2.144 4.37  4.184 4.004]
      ! Value superior for B01/D01 in altitude error: [ 0.     0.     0.     0.232 -0.337 -0.336  0.     0.     0.   ]
  Check zarr versus reference
      ! Value superior for B01/D01 in planar error: [7.631 6.887 6.156 5.829 5.884 1.064 9.601 8.756 7.75 ]
      ! Value superior for B01/D01 in altitude error: [43.671 43.748 43.692 37.353 41.51   5.969 52.338 52.957 52.525]
  Check versus zarr
      ! Value superior for B01/D01 in planar error: [ 9.174  8.362  7.537  7.795  7.994  1.087 10.98  10.106  9.043]
Sensor 'B02/D01' #2/156
  Check versus reference
      ! Value superior for B02/D01 in planar error: [3.277 3.473 3.668 1.64  1.681 1.748 4.25  4.056 3.869]
      ! Value superior for B02/D01 in altitude error: [ 0.     0.    -0.     0.409  0.169 -0.112  0.     0.     0.   ]
  Check zarr versus reference
      ! Value superior for B02/D01 in planar error: [ 8.298  7.59   6.901  4.856 13.347  2.186  9.262  8.378  7.341]
      ! Value superior for B02/D01 in altitude error: [43.672 43.74  43.67  31.455 79.542 22.876 52.171 52.782 52.434]
  Check versus zarr
      ! Value superior for B02/D01 in planar error: [ 9.112  8.283  7.44   6.476 15.027  3.915 10.874  9.985  8.921]
Sensor 'B03/D01' #3/156
  Check versus reference
      ! Value superior for B03/D01 in planar error: [3.279 3.47  3.661 2.008 1.866 1.939 4.281 4.089 3.904]
      ! Value superior for B03/D01 in altitude error: [ 0.     0.     0.    -0.188  0.444  0.091  0.     0.     0.   ]
  Check zarr versus reference
      ! Value superior for B03/D01 in planar error: [ 8.122  7.402  6.7   26.165  5.374  4.546  9.319  8.441  7.412]
      ! Value superior for B03/D01 in altitude error: [ 43.672  43.742  43.675 134.658  38.611  38.116  52.209  52.813  52.459]
  Check versus zarr
      ! Value superior for B03/D01 in planar error: [ 9.116  8.291  7.45  28.14   7.232  6.483 10.887  9.997  8.935]
Sensor 'B04/D01' #4/156
  Check versus reference
      ! Value superior for B04/D01 in planar error: [3.285 3.471 3.658 2.18  2.123 2.043 4.311 4.12  3.937]
      ! Value superior for B04/D01 in altitude error: [ 0.     0.     0.    -0.193  0.012  0.526  0.     0.     0.   ]
  Check zarr versus reference
      ! Value superior for B04/D01 in planar error: [ 7.841  7.112  6.399 18.091  1.793 21.225  9.379  8.508  7.484]
      ! Value superior for B04/D01 in altitude error: [ 43.671  43.742  43.677 -76.483   1.758 136.748  52.246  52.85   52.481]
  Check versus zarr
      ! Value superior for B04/D01 in planar error: [ 9.126  8.302  7.464 15.936  0.331 23.268 10.904 10.017  8.954]
      ! Value superior for B04/D01 in altitude error: [ -43.671  -43.742  -43.677   76.29    -1.747 -136.222  -52.246  -52.85
  -52.481]
Sensor 'B05/D01' #5/156
  Check versus reference
      ! Value superior for B05/D01 in planar error: [3.424 3.615 3.806 2.075 2.162 1.978 4.336 4.146 3.964]
      ! Value superior for B05/D01 in altitude error: [ 0.     0.     0.     0.058 -0.468  0.61   0.     0.     0.   ]
  Check zarr versus reference
      ! Value superior for B05/D01 in planar error: [ 7.964  7.239  6.53  15.251  1.24   0.637  9.405  8.539  7.518]
```

[…]

```
Sensor 'B05/D07' #83/156
  Check versus reference
      ! Value superior for B05/D07 in planar error: [5.482 5.673 5.865 3.003 3.159 3.322 2.697 2.624 2.567]
  Check zarr versus reference
      ! Value superior for B05/D07 in planar error: [4.588 5.068 5.653 2.34  3.217 4.109 3.217 3.893 4.668]
      ! Value superior for B05/D07 in altitude error: [44.654 45.619 45.49  49.228 49.69  49.333 51.75  52.625 52.666]
  Check versus zarr
      ! Value superior for B05/D07 in planar error: [0.927 1.234 1.862 1.003 1.332 2.013 1.03  1.399 2.146]
Sensor 'B06/D07' #84/156
FAILED
```

```
>           raise ValueError(msg_err)
E           ValueError: Input X contains NaN.
E           PolynomialFeatures does not accept missing values encoded as NaN natively. For supervised learning, you might want to consider sklearn.ensemble.HistGradientBoostingClassifier and Regressor whic
h accept missing values encoded as NaNs natively. Alternatively, it is possible to preprocess the data, for instance by using an imputer transformer in a pipeline or drop samples with missing values. See h
ttps://scikit-learn.org/stable/modules/impute.html You can find a list of all estimators that handle NaN values at the following page: https://scikit-learn.org/stable/modules/impute.html#estimators-that-ha
ndle-nan-values

/usr/local/lib/python3.11/site-packages/sklearn/utils/validation.py:175: ValueError
------------------------------------------------------------------------------------------------ Captured log call ------------------------------------------------------------------------------------------------
WARNING  asgard.models.dem:dem.py:114 Low longitude coverage: 359.833333
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.86732381810572 and longitude 178.3333110067144 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.867879114928495 and longitude 178.33317619732668 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.87181960700765 and longitude 178.3333295523452 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.8724347678593 and longitude 178.33332955235164 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.87181960700765 and longitude 178.3333295523452 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.8724347678593 and longitude 178.33332955235164 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.87181960700765 and longitude 178.3333295523452 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.8724347678593 and longitude 178.33332955235164 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.87181960700765 and longitude 178.3333295523452 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -17.8724347678593 and longitude 178.33332955235164 does not contain required point neighborhood
WARNING  pyrugged:duvenhage_algorithm.py:477 the tile selected for latitude -22.500082786916302 and longitude 178.47988336127844 does not contain required point neighborhood
================================================================================================ warnings summary ================================================================================================
tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1C_DEM_REFACTORED]
tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1C_DEM_REFACTORED]
  /usr/local/lib/python3.11/site-packages/asgard/products/sentinel2/msi.py:697: DeprecationWarning: Support of native DEM will be removed, use Zarr versions
    self.propagation_model = PropagationModel(**model_kwargs)

tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1C_DEM_REFACTORED]
  /usr/local/lib/python3.11/site-packages/asgard/core/product.py:285: RuntimeWarning: invalid value encountered in cast
    output = coordinates.astype(int)

-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
============================================================================================= short test summary info =============================================================================================
FAILED asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1C_DEM_REFACTORED] - ValueError: Input X contains NaN.
========================================================================== 1 failed, 40 deselected, 3 warnings in 160.98s (0:02:40) ==========================================================================
```

2 Issues:

- ✓ Direct locations give results above thresholds => [#200]
- ✓ Test fails before end: [#201]

3.9.2.5 [S2MSI_TDS2_L1C_DEM_INVLOC-010]

[S2MSI_TDS2_L1C_DEM_INVLOC-010]			
Summary: *This test aims at assessing that for L1C sensor model using DEM:* ✓ *ASGARD gives equivalent results for inverse location grids,* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *TDS2: Antemeridian*			
Preconditions: Reference has been generated with the Legacy processor			
#:	Step actions:	Expected Results:	Results:

[S2MSI_TDS2_L1C_DEM_INVLOC-010]			

1	**Run the Legacy-based implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS2_L1C_ DEM_INVOLOC_LE GACY**]
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1C_DEM_INVLOC_LEGACY]`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}`
			`Read reference operation text file /DATA/TDS/S2MSI_TDS2/s2geo_reference_grid_INVLOC_Full.pkl`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	✓ **PASSED**
			`Differences versus ref`
			
			`Max Differences from reference:`
			`- inverse loc (pixel): 1.49e-08 (5.41e-20)`
			
			`List Sensor that differs from reference:`
			
			`Execution time for:`
			`- inverse_loc: 1.96e+03s`
			`Reference execution time for:`
			`- inverse loc: 745257 ms.`
			
			`Test run to the end`
			`PASSED`

		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ Time is monitored and will be analysed in §3.10:
			`Execution time for:`
			`- inverse_loc: 1.96e+03s`
			`Reference execution time for:`
			`- inverse loc: 745257 ms.`

2	**Run the Refactored implementation with Legacy DEM:**	✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS2_L1C_DEM_INVLOC_REFACTORED]`
			`Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}`

[S2MSI_TDS2_L1C_DEM_INVLOC-010]			
	Run command: ✓ python3 -m pytest asgard-legacy /tests/test_sentinel2_msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS2_L1C_DEM_INVLOC_REFACTORED**]	✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	**Cf 3.9.2.5.1 - Step 2.2:** ✓ Using Legacy DEM: **FAILED** ➢ Inverse locations are a little far from reference => #203 ➢ Processing ends in error => #208 ✓ **Copernicus Zarr DEM not tested** => #197 ➢ Similar results with GETAS.zarr than using Legacy DEM
		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ Processing in error, no time available.

3.9.2.5.1 Step 2.2

Here is the start of the looping log:

```
Sensor 'B12/D03' #39/156
Sensor 'B01/D04' #40/156
    Number of inverse locations: 151440
    Compute inverse loc grid.
      time ~ 2296.955801066011s
      ! Value superior for B01/D04 in inverse_loc error: [0.019 0.019 0.019 0.019 0.02  0.02  0.02  0.02  0.02  0.02 ] [...]
    Compute inverse loc grid with Zarr.
      time ~ 2176.8833612389863s
      ! Value superior for B01/D04 in inverse_loc error: [0.021 0.021 0.021 0.021 0.022 0.022 0.022 0.022 0.022 0.022] [...]
    Compute differences of previous steps.
Sensor 'B02/D04' #41/156
    Number of inverse locations: 2406805
    Compute inverse loc grid.
      time ~ 35364.88267797232s
      ! Value superior for B02/D04 in inverse_loc error: [0.232 0.232 0.232 0.231 0.231 0.23  0.23  0.23  0.23  0.23 ] [...]
    Compute inverse loc grid with Zarr.
  ! AttributError(type(exp)=<class 'AttributeError'>): inverse loc 'NoneType' object has no attribute 'getX'
      time ~ 7688.088803958148s
FAILED
```

```
asgard-legacy/tests/test_legacy_sentinel2_msi.py:1078: in inverse_loc_grid_test_sentinel2_msi
    error_invloc_zarr = planar_captor_error(np.array(ref_data[sensor]["inverse_loc"])[:, 3:],
```

```
>       return np.square((x_first - x_second) ** 2 + (y_first - y_second) ** 2)
E       ValueError: operands could not be broadcast together with shapes (481361,) (30288,)

asgard-legacy/tests/helpers/compare.py:316: ValueError
----------------------------------------------------------------------------------------- Captured log call ---
WARNING  asgard.models.dem:dem.py:114 Low longitude coverage: 359.833333
```

Issues:

- ✓ Inverse locations are a little far from reference => #203
- ✓ A NoneType is return, but might be different from TDS2 issue on INVLOC_REFACTORED, hence new issue created => #208

3.9.3 S2MSI_TDS3

3.9.3.1 [S2MSI_TDS3_L1B_DEM-010]

[S2MSI_TDS3_L1B_DEM-010]
Summary: *This test aims at assessing that for L1B sensor model using DEM:* ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.*

<table>
<tr><th colspan="4">[S2MSI_TDS3_L1B_DEM-010]</th></tr>
<tr><td colspan="4"><u>Test Data:</u>

TDS3: Meridian 0</td></tr>
<tr><td colspan="4"><u>Preconditions:</u>
Reference has been generated with the Legacy processor</td></tr>
<tr><td><u>#:</u></td><td><u>Step actions:</u></td><td><u>Expected Results:</u></td><td><u>Results:</u></td></tr>
<tr><td rowspan="3"><u>1</u></td><td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS3_L1B_ DEM_LEGACY]

Which represents test configuration:
✓ L1B_DEM_Legacy
✓ on S2MSI_TDS3
✓ with footprint ref
✓ with --legacy</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
<code>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS3_L1B_DEM_LE</code>

<code>Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001,</code></td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.3.1.1 - Step 1.2:
✓ FAILED:
➢ Planar distance too high => #207
➢ Footprint very very far => #207
<code>Differences versus ref</code>

<code>Max Differences from reference:</code>
<code>- direct loc:</code>
<code>- planar dist (m): 30.7 (0.144)</code>
<code>- altitude (m): 0.0405 (3.16e-06)</code>
<code>- sun angles (angular diff (deg)): 0.00058 (2.41e-06)</code>
<code>- incidence angles (angular diff (deg)): 0.000512 (0.000353)</code>
<code>- inverse loc (pixel): 6.19e-09 (1.21e-10)</code>
<code>- footprint:</code>
<code>- ratio: 0.963 (0.963)</code>
<code>- max dist (pixel): 7.89e+03 (7.81e+03)</code></td></tr>
<tr><td>✓ Verify that performances are logged and save</td><td>✓ Time is monitored and will be analysed in §3.10:</td></tr>
</table>

<table>
<tr><th colspan="4">[S2MSI_TDS3_L1B_DEM-010]</th></tr>
<tr><td></td><td></td><td>results for dedicated performance analysis</td><td>Execution time for:
- direct loc: 111s
- inverse loc: 88.9s
- sun_angles: 0.144s
- incidence_angles: 0.0705s
- footprint: 2.6e+03s
Reference execution time for:
- direct loc + sun + viewing angles: 6865 ms.</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Run the Refactored implementation:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS3_L1B_</td><td rowspan="2">✓ Verify that thresholds are the ones from the metrics table in §2.3.2

✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS3_L1B_DEM_REFACTORED]
Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td></tr>
<tr><td>Cf 3.9.3.1.2 - Step 2.2:
✓ Using Legacy DEM: FAILED
➢ Direct locations give results above thresholds => #200
➢ RAM issues => #199
✓ Copernicus Zarr DEM not tested => #197
➢ Similar results with GETAS.zarr than using Legacy DEM</td></tr>
</table>

[S2MSI_TDS3_L1B_DEM-010]			
	DEM_REFACTORED] Which represents test configuration: ✓ S2MSI_TDS3 ✓ with footprint ref ✓ without --legacy	✓ Verify that performances are logged and save results for dedicated performance analysis	✓ Test not ending, hence no results available

3.9.3.1.1 Step 1.2

Here is the global summary:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 30.7 (0.144)
        - altitude (m): 0.0405 (3.16e-06)
    - sun angles (angular diff (deg)): 0.00058 (2.41e-06)
    - incidence angles (angular diff (deg)): 0.000512 (0.000353)
    - inverse loc (pixel): 6.19e-09 (1.21e-10)
    - footprint:
        - ratio: 0.963 (0.963)
        - max dist (pixel): 7.89e+03 (7.81e+03)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
        - altitude (m): ['B02/D01' 'B04/D01' 'B07/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B04/D02'
 'B06/D02' 'B08/D02' 'B09/D02' 'B10/D02' 'B01/D03' 'B03/D03' 'B06/D03'
 'B07/D03' 'B8A/D03' 'B09/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B03/D04'
 'B09/D04' 'B11/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B06/D05' 'B8A/D05'
 'B09/D05' 'B11/D05' 'B12/D05' 'B02/D06' 'B03/D06' 'B04/D06' 'B07/D06'
 'B08/D06' 'B8A/D06' 'B09/D06' 'B11/D06' 'B02/D07' 'B03/D07' 'B05/D07'
 'B06/D07' 'B8A/D07' 'B11/D07' 'B12/D07' 'B01/D08' 'B02/D08' 'B03/D08'
 'B04/D08' 'B07/D08' 'B08/D08' 'B8A/D08' 'B09/D08' 'B11/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B06/D09' 'B08/D09' 'B8A/D09' 'B10/D09'
 'B11/D09' 'B12/D09' 'B01/D10' 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10'
 'B07/D10' 'B8A/D10' 'B09/D10' 'B10/D10' 'B11/D10' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B07/D11' 'B10/D11' 'B11/D11' 'B01/D12' 'B03/D12'
 'B04/D12' 'B05/D12' 'B06/D12' 'B08/D12' 'B10/D12' 'B11/D12'])
```

```
    - footprint: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 111s
    - inverse loc: 88.9s
    - sun_angles: 0.144s
    - incidence_angles: 0.0705s
    - footprint: 2.6e+03s
Reference execution time for:
    - direct loc + sun + viewing angles:  6865 ms.

Test run to the end
PASSED
```

Results are NOK because some footprint are very far and some direct location too (30.7m) even if the altitude is OK. It might be due to the same root cause, hence only one issue is opened for now => #207

3.9.3.1.2 Step 2.2

Here is the start of the looping log:

```
Read reference operation text file /DATA/TDS/S2MSI_TDS3/L1B_DEM_Legacy_s2geo_reference.txt
Sensor 'B01/D01' #1/156
  Check versus reference
      ! Value superior for B01/D01 in planar error: [0.024 0.024 0.024 0.007 0.007 0.007 0.025 0.025 0.025]
      ! Value superior for B01/D01 for footprint: [9.58291463e-01 1.26900955e+03 1.26900955e+03 1.03950842e-02]
  Check zarr versus reference
      ! Value superior for B01/D01 in planar error: [0.401 0.323 0.412 0.119 0.108 0.093 0.08  0.071 0.077]
      ! Value superior for B01/D01 in altitude error: [1.971 1.761 2.46  0.529 0.525 0.499 0.256 0.235 0.297]
      ! Value superior for B01/D01 for footprint: [9.58287888e-01 1.26901279e+03 1.26901279e+03 0.00000000e+00]
  Check versus zarr
      ! Value superior for B01/D01 in planar error: [0.417 0.34  0.428 0.112 0.101 0.087 0.054 0.045 0.052]
Sensor 'B02/D01' #2/156
  Check versus reference
      ! Value superior for B02/D01 in planar error: [0.081 0.08  0.079 0.22  0.219 0.217 0.033 0.033 0.032]
      ! Value superior for B02/D01 for footprint: [9.58250866e-01 7.60485337e+03 7.60485337e+03 2.07738255e-02]
  Check zarr versus reference
      ! Value superior for B02/D01 in planar error: [0.444 0.366 0.45  0.332 0.319 0.303 0.083 0.073 0.079]
      ! Value superior for B02/D01 in altitude error: [1.976 1.764 2.46  0.532 0.527 0.499 0.257 0.234 0.297]
      ! Value superior for B02/D01 for footprint: [9.58247288e-01 7.60486878e+03 7.60486878e+03 0.00000000e+00]
  Check versus zarr
      ! Value superior for B02/D01 in planar error: [0.416 0.337 0.423 0.112 0.101 0.086 0.054 0.045 0.051]
Sensor 'B03/D01' #3/156
  Check versus reference
      ! Value superior for B03/D01 in planar error: [0.092 0.091 0.091 0.279 0.277 0.275 0.068 0.068 0.067]
      ! Value superior for B03/D01 for footprint: [9.58300572e-01 7.60807932e+03 7.60807932e+03 0.00000000e+00]
  Check zarr versus reference
      ! Value superior for B03/D01 in planar error: [0.326 0.248 0.335 0.17  0.178 0.191 0.048 0.048 0.048]
      ! Value superior for B03/D01 in altitude error: [1.975 1.763 2.46  0.532 0.527 0.499 0.257 0.235 0.297]
      ! Value superior for B03/D01 for footprint: [9.58297672e-01 7.60809567e+03 7.60809567e+03 0.00000000e+00]
  Check versus zarr
      ! Value superior for B03/D01 in planar error: [0.416 0.337 0.424 0.112 0.101 0.086 0.054 0.045 0.051]
Sensor 'B04/D01' #4/156
  Check versus reference
      ! Value superior for B04/D01 in planar error: [0.209 0.207 0.206 0.195 0.193 0.191 0.187 0.185 0.183]
      ! Value superior for B04/D01 for footprint: [9.58277123e-01 7.61013150e+03 7.61013150e+03 2.37194832e-03]
  Check zarr versus reference
      ! Value superior for B04/D01 in planar error: [0.209 0.132 0.22  0.131 0.132 0.135 0.236 0.226 0.231]
      ! Value superior for B04/D01 in altitude error: [1.974 1.762 2.459 0.531 0.527 0.499 0.257 0.235 0.297]
      ! Value superior for B04/D01 for footprint: [9.58273885e-01 7.61014868e+03 7.61014868e+03 0.00000000e+00]
  Check versus zarr
      ! Value superior for B04/D01 in planar error: [0.416 0.337 0.424 0.112 0.101 0.086 0.054 0.045 0.051]
Sensor 'B05/D01' #5/156
```

Direct locations give results above thresholds => #200

Footprints are far from each other, and overall, the test fails:

```
Sensor 'B8A/D02' #22/156
Killed
```

Fail is due to RAM issue as the RAM is going up and up between loops over sensors => #199

```
Sensor 'B8A/D01' #9/156
asgard  8%  49,79 GB / 125,79 GB
```

```
Sensor 'B11/D01' #12/156
asgard  9%  68,21 GB / 125,79 GB
```

```
Sensor 'B12/D01' #13/156
asgard  7%  71,86 GB / 125,79 GB
```

3.9.3.2 [S2MSI_TDS3_L1C_DEM_INVLOC-010]

[S2MSI_TDS3_L1C_DEM_INVLOC-010]
Summary: *This test aims at assessing that for L1C sensor model using DEM:* ✓ *ASGARD gives equivalent results for inverse location grids,* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *TDS2: Meridian 0*

<table>
<tr><th colspan="4">[S2MSI_TDS3_L1C_DEM_INVLOC-010]</th></tr>
<tr><td colspan="4"><u>Preconditions:</u>
Reference has been generated with the Legacy processor</td></tr>
<tr><td><u>#:</u></td><td><u>Step actions:</u></td><td><u>Expected Results:</u></td><td><u>Results:</u></td></tr>
<tr><td rowspan="3">1</td><td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS3_L1C_ DEM_INVLOC_LEG ACY]</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
<code>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS3_L1C_DEM_INVLOC_LEGACY]</code>
<code>Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</code></td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>✓ OK:
<code>Differences versus ref</code>

<code>Max Differences from reference:</code>
<code>   - inverse loc (pixel): 6.62e-09 (9.36e-19)</code>

<code>List Sensor that differs from reference:</code>

<code>Execution time for:</code>
<code>   - inverse_loc: 682s</code>
<code>Reference execution time for:</code>
<code>   - inverse loc: 192371 ms.</code>

<code>Test run to the end</code>
<code>PASSED</code></td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
<code>Execution time for:</code>
<code>   - inverse_loc: 682s</code>
<code>Reference execution time for:</code>
<code>   - inverse loc: 192371 ms.</code></td></tr>
</table>

<table>
<tr><th colspan="4">[S2MSI_TDS3_L1C_DEM_INVLOC-010]</th></tr>
<tr><td rowspan="3">2</td><td rowspan="3">Run the Refactored implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS3_L1C_ DEM_INVLOC_REF ACTORED]</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
<code>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS3_L1C_DEM_INVLOC_REFACTORED]</code>
<code>Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</code></td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.3.2.1 - Step 2.2:
✓ Using Legacy DEM: FAILED
➢ Inverse locations are a little far from references => #203
➢ Test not ending => #202
✓ Copernicus Zarr DEM not tested => #197
➢ Similar results with GETAS.zarr than using Legacy DEM</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Test not ending</td></tr>
</table>

3.9.3.2.1 Step 2.2

Here is the end of the looping log:

```
Sensor 'B01/D09' #105/156
    Number of inverse locations: 90390
    Compute inverse loc grid.
      time ~ 1284.0049109430984s
      ! Value superior for B01/D09 in inverse_loc error: [0.017 0.017 0.017 0.017 0.017 0.017 0.017 0.017 0.017 0.017] [...]
    Compute inverse loc grid with Zarr.
      time ~ 1315.0858951518312s
      ! Value superior for B01/D09 in inverse_loc error: [0.017 0.018 0.018 0.018 0.017 0.017 0.018 0.017 0.017 0.017] [...]
    Compute differences of previous steps.
Sensor 'B02/D09' #106/156
    Number of inverse locations: 1425325
    Compute inverse loc grid.
      time ~ 21139.9568294498522s
      ! Value superior for B02/D09 in inverse_loc error: [0.102 0.102 0.102 0.102 0.102 0.101 0.101 0.101 0.101 0.101] [...]
    Compute inverse loc grid with Zarr.
      time ~ 20819.35085535515s
      ! Value superior for B02/D09 in inverse_loc error: [0.114 0.115 0.117 0.119 0.121 0.122 0.124 0.125 0.127 0.128] [...]
    Compute differences of previous steps.
      ! Value superior for B02/D09 in inverse_loc error: [0. 0. 0. 0. 0. 0. 0. 0. 0. 0.] [...]
Sensor 'B03/D09' #107/156
    Number of inverse locations: 1423045
    Compute inverse loc grid.
Fatal Python error: Segmentation fault
```

```
Extension modules: numpy.core._multiarray_umath, numpy.core._multiarray_tests, numpy.linalg._umath_linalg, numpy.fft._pocketfft_internal, numpy.random._common, numpy.random.bit_generator, numpy.random._bou
nded_integers, numpy.random._mt19937, numpy.random.mtrand, numpy.random._philox, numpy.random._pcg64, numpy.random._sfc64, numpy.random._generator, sxgeo._sxgeo, shapely.lib, shapely._geos, shapely._geomet
ry_helpers, sklearn.__check_build._check_build, scipy._lib._ccallback_c, scipy.sparse._sparsetools, _csparsetools, scipy.sparse._csparsetools, scipy.linalg._fblas, scipy.linalg._flapack, scipy.linalg.cytho
n_lapack, scipy.linalg._cythonized_array_utils, scipy.linalg._solve_toeplitz, scipy.linalg._flinalg, scipy.linalg._decomp_lu_cython, scipy.linalg._matfuncs_sqrtm_triu, scipy.linalg.cython_blas, scipy.linal
g._matfuncs_expm, scipy.linalg._decomp_update, scipy.sparse.linalg._dsolve._superlu, scipy.sparse.linalg._eigen.arpack._arpack, scipy.sparse.csgraph._tools, scipy.sparse.csgraph._shortest_path, scipy.spars
e.csgraph._traversal, scipy.sparse.csgraph._min_spanning_tree, scipy.sparse.csgraph._flow, scipy.sparse.csgraph._matching, scipy.sparse.csgraph._reordering, psutil._psutil_linux, psutil._psutil_posix, scip
y.spatial._ckdtree, scipy._lib.messagestream, scipy.spatial._qhull, scipy.spatial._voronoi, scipy.spatial._distance_wrap, scipy.spatial._hausdorff, scipy.special._ufuncs_cxx, scipy.special._ufuncs, scipy.s
pecial._specfun, scipy.special._comb, scipy.special._ellip_harm_2, scipy.spatial.transform._rotation, scipy.ndimage._nd_image, _ni_label, scipy.ndimage._ni_label, scipy.optimize._minpack2, scipy.optimize._
group_columns, scipy.optimize._trlib._trlib, scipy.optimize._lbfgsb, _moduleTNC, scipy.optimize._moduleTNC, scipy.optimize._cobyla, scipy.optimize._slsqp, scipy.optimize._minpack, scipy.optimize._lsq.given
s_elimination, scipy.optimize._zeros, scipy.optimize._highs.cython.src._highs_wrapper, scipy.optimize._highs._highs_wrapper, scipy.optimize._highs.cython.src._highs_constants, scipy.optimize._highs._highs_
constants, scipy.linalg._interpolative, scipy.optimize._bglu_dense, scipy.optimize._lsap, scipy.optimize._direct, scipy.integrate._odepack, scipy.integrate._quadpack, scipy.integrate._vode, scipy.integrate
._dop, scipy.integrate._lsoda, scipy.special.cython_special, scipy.stats._stats, scipy.stats.beta_ufunc, scipy.stats._boost.beta_ufunc, scipy.stats.binom_ufunc, scipy.stats._boost.binom_ufunc, scipy.stats.
nbinom_ufunc, scipy.stats._boost.nbinom_ufunc, scipy.stats.hypergeom_ufunc, scipy.stats._boost.hypergeom_ufunc, scipy.stats.ncf_ufunc, scipy.stats._boost.ncf_ufunc, scipy.stats.ncx2_ufunc, scipy.stats._boo
st.ncx2_ufunc, scipy.stats.nct_ufunc, scipy.stats._boost.nct_ufunc, scipy.stats.skewnorm_ufunc, scipy.stats._boost.skewnorm_ufunc, scipy.stats.invgauss_ufunc, scipy.stats._boost.invgauss_ufunc, scipy.inter
polate._fitpack, scipy.interpolate.dfitpack, scipy.interpolate._bspl, scipy.interpolate._ppoly, scipy.interpolate.interpnd, scipy.interpolate._rbfinterp_pythran, scipy.interpolate._rgi_cython, scipy.stats.
_biasedurn, scipy.stats._levy_stable.levyst, scipy.stats._stats_pythran, scipy._lib._uarray._uarray, scipy.stats._ansari_swilk_statistics, scipy.stats._sobol, scipy.stats._qmc_cy, scipy.stats._mvn, scipy.s
tats._rcont.rcont, scipy.stats._unuran.unuran_wrapper, sklearn.utils._isfinite, sklearn.utils.murmurhash, sklearn.utils._openmp_helpers, sklearn.utils._random, sklearn.utils._seq_dataset, sklearn.utils.spa
rsefuncs_fast, sklearn.metrics.cluster._expected_mutual_info_fast, sklearn.preprocessing._csr_polynomial_expansion, sklearn.preprocessing._target_encoder_fast, sklearn.metrics._dist_metrics, sklearn.metric
s._pairwise_distances_reduction._datasets_pair, sklearn.utils._cython_blas, sklearn.metrics._pairwise_distances_reduction._base, sklearn.metrics._pairwise_distances_reduction._middle_term_computer, sklearn
.utils._heap, sklearn.utils._sorting, sklearn.metrics._pairwise_distances_reduction._argkmin, sklearn.metrics._pairwise_distances_reduction._argkmin_classmode, sklearn.utils._vector_sentinel, sklearn.metri
cs._pairwise_distances_reduction._radius_neighbors, sklearn.metrics._pairwise_distances_reduction._radius_neighbors_classmode, sklearn.metrics._pairwise_fast, sklearn.linear_model._cd_fast, sklearn._loss._
loss, sklearn.utils.arrayfuncs, sklearn.svm._liblinear, sklearn.svm._libsvm, sklearn.svm._libsvm_sparse, sklearn.utils._weight_vector, sklearn.linear_model._sgd_fast, sklearn.linear_model._sag_fast, lxml._
elementpath, lxml.etree, lxml.objectify, cftime._cftime, netCDF4._netCDF4, asgard.wrappers.dfdl.s3_navatt, asgard.wrappers.dfdl.s3_slstr_slt, asgard.wrappers.dfdl.s3_timestamp_annotation, pyrugged.utils.ma
th_utils, numcodecs.compat_ext, numcodecs.blosc, numcodecs.zstd, numcodecs.lz4, numcodecs._shuffle, msgpack._cmsgpack, numcodecs.jenkins, numcodecs.vlen, numcodecs.fletcher32 (total: 169)
Segmentation fault (core dumped)
```

Issues observed:

- ✓ Inverse locations are a little far from reference => #203
- ✓ Segmentation fault => #202

3.9.4 S2MSI_TDS4

3.9.4.1 [S2MSI_TDS4_L1B_DEM-010]

[S2MSI_TDS4_L1B_DEM-010]
<u>Summary:</u> *This test aims at assessing that for L1B sensor model using DEM:* ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* <u>Actors:</u> *CS GROUP* <u>Expected Results:</u> *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* <u>Pass/Fail criteria:</u> *One large increase of time is observed, or another metrics is higher than a defined threshold.*

<table>
<tr><th colspan="4">[S2MSI_TDS4_L1B_DEM-010]</th></tr>
<tr><td colspan="4"><u>Test Data:</u>

TDS4: Equator</td></tr>
<tr><td colspan="4"><u>Preconditions:</u>
Reference has been generated with the Legacy processor</td></tr>
<tr><td><u>#:</u></td><td><u>Step actions:</u></td><td><u>Expected Results:</u></td><td><u>Results:</u></td></tr>
<tr><td rowspan="3"><u>1</u></td><td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS4_L1B_ DEM_LEGACY]

Which represents test configuration:
✓ L1B_DEM_Legacy
✓ on S2MSI_TDS4
✓ with footprint ref
✓ with --legacy</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
<code>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS4_L1B_DEM_LEGACY]</code>
<code>Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</code></td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf3.9.4.1.1 - Step 1.2:
✓ PASSED</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
<pre>Execution time for:
    - direct loc: 170s
    - inverse loc: 4.1s
    - sun_angles: 0.149s
    - incidence_angles: 0.0702s
    - footprint: 3.89e+03s
Reference execution time for:
    - direct loc + sun + viewing angles: 13567 ms.</pre></td></tr>
<tr><td><u>2</u></td><td>Run the Refactored implementation:
Run command:</td><td>Using DEM Legacy part:</td><td>✓ OK:
<code>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS4_L1B_DEM_REFACTORED]</code>
<code>Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</code></td></tr>
</table>

[S2MSI_TDS4_L1B_DEM-010]			
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS4_L1B_DEM_REFACTORED**] Which represents test configuration: ✓ S2MSI_TDS4 ✓ with footprint ref ✓ without --legacy	✓ Verify that thresholds are the ones from the metrics table in §2.3.2 ✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings) ✓ Verify that performances are logged and save results for dedicated performance analysis	**Cf 3.9.4.1.2 - Step 2.2:** ✓ Using Legacy DEM: **FAILED** ➢ Direct locations give results above thresholds => #200 ➢ Test is crashing due to too high RAM consumption => #199 ✓ **Copernicus Zarr DEM not tested** => #197 ➢ Similar results with GETAS.zarr than using Legacy DEM
			✓ Test not ending, hence no result available

3.9.4.1.1 Step 1.2

Here is the global summary:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 9.53e-09 (4.03e-09)
        - altitude (m): 2.15e-09 (1.28e-10)
    - sun angles (angular diff (deg)): 1.21e-06 (8.54e-07)
    - incidence angles (angular diff (deg)): 0.000416 (0.000354)
    - inverse loc (pixel): 9.4e-09 (2.11e-10)
    - footprint:
        - ratio: 0.998 (0.998)
        - max dist (pixel): 1.06 (1.05)

List Sensor that differs from reference:
    - footprint: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 170s
    - inverse loc: 4.1s
    - sun_angles: 0.149s
    - incidence_angles: 0.0702s
    - footprint: 3.89e+03s
Reference execution time for:
    - direct loc + sun + viewing angles:  13567 ms.

Test run to the end
PASSED
```

Results are good, except for the footprint max dist, which is very close to 1, so acceptable

3.9.4.1.2 Step 2.2

Here is the start of the looping log:

```
Read reference operation text file /DATA/TDS/S2MSI_TDS4/L1B_DEM_Legacy_s2geo_reference.txt
Sensor 'B01/D01' #1/156
  Check versus reference
      ! Value superior for B01/D01 in planar error: [ 2.57   2.764  2.955  6.309  6.714  7.291 11.115 11.258 11.404]
      ! Value superior for B01/D01 in altitude error: [-0.    -0.    -0.     2.015  0.268 -3.276 -0.    -0.     0.   ]
  Check zarr versus reference
      ! Value superior for B01/D01 in planar error: [ 2.606  2.774  2.979 21.281 13.282  9.02  11.262 11.289 11.477]
      ! Value superior for B01/D01 in altitude error: [ -0.856  -0.273  -0.611 118.27   86.496  68.153  -0.964  -0.221  -0.576]
  Check versus zarr
      ! Value superior for B01/D01 in planar error: [ 0.181  0.053  0.106 24.511 16.554 12.384  0.203  0.042  0.1  ]
      ! Value superior for B01/D01 in altitude error: [   0.856    0.273    0.611 -116.255  -86.228  -71.429    0.964    0.221
    0.576]
Sensor 'B02/D01' #2/156
  Check versus reference
      ! Value superior for B02/D01 in planar error: [ 2.62   2.812  3.003  6.714  6.685  7.075 10.819 10.974 11.132]
  Check zarr versus reference
      ! Value superior for B02/D01 in planar error: [ 2.624  2.813  3.007 20.85  13.002  5.857 10.954 11.    11.192]
      ! Value superior for B02/D01 in altitude error: [ -0.857  -0.275  -0.614 113.016  80.411  26.374  -1.043  -0.222  -0.58 ]
  Check versus zarr
      ! Value superior for B02/D01 in planar error: [ 0.18   0.052  0.105 24.075 15.352  4.989  0.218  0.042  0.099]
      ! Value superior for B02/D01 in altitude error: [   0.857    0.275    0.614 -115.103  -80.819  -29.18     1.043    0.222
    0.58 ]
Sensor 'B03/D01' #3/156
  Check versus reference
      ! Value superior for B03/D01 in planar error: [ 2.567  2.761  2.953  6.435  6.611  6.994 10.909 11.06  11.213]
  Check zarr versus reference
      ! Value superior for B03/D01 in planar error: [ 2.584  2.765  2.964 24.622 13.424  7.274 11.051 11.087 11.277]
      ! Value superior for B03/D01 in altitude error: [ -0.856  -0.274  -0.614 131.537  83.065  47.799  -1.037  -0.222  -0.579]
  Check versus zarr
      ! Value superior for B03/D01 in planar error: [ 0.18   0.052  0.105 27.66  15.916  8.678  0.217  0.042  0.099]
      ! Value superior for B03/D01 in altitude error: [   0.856    0.274    0.614 -132.151  -83.697  -50.684    1.037    0.222
    0.579]
Sensor 'B04/D01' #4/156
  Check versus reference
      ! Value superior for B04/D01 in planar error: [ 2.541  2.734  2.925  6.542  6.812  7.263 10.942 11.09  11.241]
      ! Value superior for B04/D01 in altitude error: [-0.    -0.    -0.     0.075 -0.754 -3.554 -0.    -0.     0.   ]
  Check zarr versus reference
      ! Value superior for B04/D01 in planar error: [ 2.554  2.737  2.935 23.867 13.571  8.465 11.086 11.118 11.308]
      ! Value superior for B04/D01 in altitude error: [ -0.857  -0.275  -0.613 129.315  85.688  60.118  -1.013  -0.222  -0.579]
  Check versus zarr
      ! Value superior for B04/D01 in planar error: [ 0.18   0.052  0.106 27.086 16.466 10.926  0.212  0.042  0.099]
      ! Value superior for B04/D01 in altitude error: [   0.857    0.275    0.613 -129.24   -86.442  -63.672    1.013    0.222
    0.579]
```

Direct locations give results above thresholds => #200

Moreover, the test is not ending as the RAM consumption goes over machine limits (=> #199):


```
Sensor 'B11/D01' #12/156
Killed
```

3.9.4.2 [S2MSI_TDS4_L1C_DEM_INVLOC-010]

[S2MSI_TDS4_L1C_DEM_INVLOC-010]			
Summary: *This test aims at assessing that for L1C sensor model using DEM:* ✓ *ASGARD gives equivalent results for inverse location grids,* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *TDS4: Equator*			
Preconditions: Reference has been generated with the Legacy processor			
#:	Step actions:	Expected Results:	Results:

<table>
<tr><th colspan="4">[S2MSI_TDS4_L1C_DEM_INVLOC-010]</th></tr>
<tr><td rowspan="3">1</td><td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS4_L1C_ DEM_INVOLOC_LE GACY]</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS4_L1C_DEM_INVLOC_LEGACY]
Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>✓ PASSED:
Differences versus ref

Max Differences from reference:
- inverse loc (pixel): 1.2e-08 (1.16e-23)

List Sensor that differs from reference:

Execution time for:
- inverse_loc: 2.05e+03s
Reference execution time for:
- inverse loc: 750686 ms.

Test run to the end
PASSED</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
Execution time for:
- inverse_loc: 2.05e+03s
Reference execution time for:
- inverse loc: 750686 ms.</td></tr>
<tr><td>2</td><td>Run the Refactored implementation with Legacy DEM:</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS4_L1C_DEM_INVLOC_REFACTORED]
Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td></tr>
</table>

[S2MSI_TDS4_L1C_DEM_INVLOC-010]			
	Run command: ✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS4_L1C_DEM_INVLOC_REFACTORED**]	✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	**Cf 3.9.4.2.1 - Step 2.2:** ✓ Using Legacy DEM: **FAILED** ➢ Inverse locations are a little far from references => #203 ➢ Fail due to NoneType returned by inverse locations => #208 ✓ **Copernicus Zarr DEM not tested** => #197 ➢ Similar results with GETAS.zarr than using Legacy DEM
		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ **Test not ending, but very very long**

3.9.4.2.1 Step 2.2

Here is the start of the looping log:

```
Sensor 'B01/D01' #1/156
    Number of inverse locations: 910
    Compute inverse loc grid.
      time ~ 22.091823641210794s
      ! Value superior for B01/D01 in inverse_loc error: [0.026 0.026 0.026 0.026 0.026 0.026 0.026 0.026 0.027 0.026] [...]
    Compute inverse loc grid with Zarr.
      time ~ 15.606625439599156s
      ! Value superior for B01/D01 in inverse_loc error: [0.043 0.046 0.045 0.043 0.039 0.034 0.034 0.034 0.035 0.045] [...]
    Compute differences of previous steps.
Sensor 'B02/D01' #2/156
    Number of inverse locations: 11080
    Compute inverse loc grid.
      time ~ 169.38983298186213s
      ! Value superior for B02/D01 in inverse_loc error: [0.946 0.966 0.965 0.965 0.946 0.946 0.909 0.928 0.928 0.91 ] [...]
    Compute inverse loc grid with Zarr.
      time ~ 69.3097899435088s
      ! Value superior for B02/D01 in inverse_loc error: [ 8.719  9.912 10.829 11.671 12.111 12.479 12.664 13.128 12.62  11.717] [...]
    Compute differences of previous steps.
      ! Value superior for B02/D01 in inverse_loc error: [1.663 2.066 2.407 2.731 2.934 3.081 3.235 3.395 3.184 2.849] [...]
Sensor 'B03/D01' #3/156
    Number of inverse locations: 11135
    Compute inverse loc grid.
      time ~ 170.51733480766416s
      ! Value superior for B03/D01 in inverse_loc error: [0.991 0.993 0.994 0.995 0.999 1.001 1.002 1.003 1.002 1.004] [...]
    Compute inverse loc grid with Zarr.
      time ~ 69.99445575289428s
      ! Value superior for B03/D01 in inverse_loc error: [ 9.814 11.077 12.109 13.058 13.652 14.074 14.505 14.945 14.377 13.456] [...]
    Compute differences of previous steps.
      ! Value superior for B03/D01 in inverse_loc error: [1.69  2.093 2.441 2.775 2.975 3.126 3.281 3.442 3.23  2.886] [...]
Sensor 'B04/D01' #4/156
    Number of inverse locations: 11665
    Compute inverse loc grid.
      time ~ 182.0668577933684s
      ! Value superior for B04/D01 in inverse_loc error: [0.959 0.961 0.961 0.962 0.962 0.963 0.963 0.964 0.986 0.985] [...]
    Compute inverse loc grid with Zarr.
      time ~ 73.76144424639642s
      ! Value superior for B04/D01 in inverse_loc error: [ 9.835 11.099 12.13  13.076 13.655 14.072 14.499 14.937 14.457 13.513] [...]
    Compute differences of previous steps.
      ! Value superior for B04/D01 in inverse_loc error: [1.705 2.109 2.461 2.788 2.996 3.147 3.304 3.467 3.251 2.909] [...]
Sensor 'B05/D01' #5/156
    Number of inverse locations: 3235
    Compute inverse loc grid.
      time ~ 57.37956755142659s
      ! Value superior for B05/D01 in inverse_loc error: [0.174 0.174 0.174 0.174 0.174 0.175 0.175 0.175 0.175 0.175] [...]
    Compute inverse loc grid with Zarr.
      time ~ 21.767384129576385s
      ! Value superior for B05/D01 in inverse_loc error: [0.716 0.824 0.893 0.933 0.917 0.85  0.803 0.696 0.599 0.516] [...]
    Compute differences of previous steps.
      ! Value superior for B05/D01 in inverse_loc error: [0.057 0.081 0.096 0.106 0.103 0.085 0.074 0.052 0.034 0.022] [...]
Sensor 'B06/D01' #6/156
```

[…]

```
Sensor 'B01/D02' #14/156
    Number of inverse locations: 232320
    Compute inverse loc grid.
      time ~ 3234.0757736833766s
      ! Value superior for B01/D02 in inverse_loc error: [0.026 0.026 0.026 0.026 0.026 0.026 0.026 0.026 0.026 0.026] [...]
    Compute inverse loc grid with Zarr.
      time ~ 3312.557463068515s
      ! Value superior for B01/D02 in inverse_loc error: [0.028 0.029 0.029 0.028 0.027 0.026 0.026 0.026 0.026 0.027] [...]
    Compute differences of previous steps.
Sensor 'B02/D02' #15/156
    Number of inverse locations: 3719045
    Compute inverse loc grid.
  ! AttributeError(type(exp)=<class 'AttributeError'>): inverse loc 'NoneType' object has no attribute 'shiftedBy'
      time ~ 42880.526597723365s
FAILED
```

```
>       return np.square((x_first - x_second) ** 2 + (y_first - y_second) ** 2)
E       ValueError: operands could not be broadcast together with shapes (743809,) (46464,)

asgard-legacy/tests/helpers/compare.py:316: ValueError
------------------------------------------------------------------------------------------ Captured log call ------------------------------------------------------------------------------------------
WARNING  asgard.models.dem:dem.py:114 Low longitude coverage: 359.833333
========================================================================================== warnings summary ==========================================================================================
tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS4_L1C_DEM_INVLOC_REFACTORED]
tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS4_L1C_DEM_INVLOC_REFACTORED]
```

2 issues observed:

- ✓ Inverse locations are a little far from reference => #203
- ✓ A NoneType is return, but might be different from TDS2 issue on INVLOC_REFACTORED, hence new issue created => #208

3.9.5 S2MSI_TDS5

3.9.5.1 [S2MSI_TDS5_L0u_DEM-010]

[S2MSI_TDS5_L0u_DEM-010]
Summary:

[S2MSI_TDS5_L0u_DEM-010]
This test aims at assessing that for L0c sensor model of S2MSI_TDS5 using DEM: ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* <u>Actors:</u> *CS GROUP* <u>Expected Results:</u> *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* <u>Pass/Fail criteria:</u> *One large increase of time is observed, or another metrics is higher than a defined threshold.* <u>Test Data:</u> *S2MSI_TDS5: High Latitude*
<u>Preconditions:</u> Reference has been generated with the Legacy processor

[S2MSI_TDS5_L0u_DEM-010]			
#:	Step actions:	Expected Results:	Results:

1	**Run the Legacy-based implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[**S2MSI_TDS5_L0u_ DEM_LEGACY**]
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L0u_DEM_LEGACY]`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	✓ **PASSED:**
			`Differences versus ref`
			
			`Max Differences from reference:`
			`    - direct loc:`
			`        - planar dist (m): 6.46e-09 (4.82e-09)`
			`        - altitude (m): 3.1e-10 (3e-11)`
			`    - sun angles (angular diff (deg)): 1.21e-06 (8.54e-07)`
			`    - incidence angles (angular diff (deg)): 0.000393 (0.000346)`
			`    - inverse loc (pixel): 1e-08 (2.82e-09)`
			
			`List Sensor that differs from reference:`
			
			`Execution time for:`
			`    - direct loc: 33.5s`
			`    - inverse loc: 2.18s`
			`    - sun_angles: 0.151s`
			`    - incidence_angles: 0.0666s`
			`Reference execution time for:`
			`    - direct loc + sun + viewing angles:  3036 ms.`
			
			`Test run to the end`
			`PASSED`

		✓ Verify that performances are	✓ Time is monitored and will be analysed in §3.10:

<table>
<tr><th colspan="4">[S2MSI_TDS5_L0u_DEM-010]</th></tr>
<tr><td></td><td></td><td>logged and save results for dedicated performance analysis</td><td><pre>Execution time for:
    - direct loc: 33.5s
    - inverse loc: 2.18s
    - sun_angles: 0.151s
    - incidence_angles: 0.0666s
Reference execution time for:
    - direct loc + sun + viewing angles: 3036 ms.</pre></td></tr>
<tr><td rowspan="3">2</td><td rowspan="3">Run the Refactored implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS5_L0u_ DEM_REFACTORE D]</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:<pre>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L0u_DEM_REFACTORED]

Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</pre></td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.5.1.1 - Step 2.2:
✓ Using Legacy DEM: PASSED with limitations
➢ Direct locations a little far from reference => #204
✓ Copernicus GLOBE DEM in Zarr not tested</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
➢ Using Legacy DEM<pre>Execution time for:
    - direct loc: 3.28s
    - inverse loc: 48.5s
    - sun_angles: 0.272s
    - incidence_angles: 0.262s
Reference execution time for:
    - direct loc + sun + viewing angles: 3036 ms.</pre>➢ Using Zarr DEM</td></tr>
</table>

[S2MSI_TDS5_L0u_DEM-010]			
			Execution time for: - direct loc: 4.13s - inverse loc: 52.5s - sun_angles: 0.164s - incidence_angles: 0.266s Reference execution time for: - direct loc + sun + viewing angles: 3036 ms.

3.9.5.1.1 Step 2.2

Here are global summaries:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 0.138 (0.133)
        - altitude (m): 0.00191 (0.000141)
    - sun angles (angular diff (deg)): 5.53e-06 (4.83e-06)
    - incidence angles (angular diff (deg)): 0.000395 (0.000344)
    - inverse loc (pixel): 6.01e-11 (3.7e-17)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
        - altitude (m): ['B08/D03' 'B11/D04'])


Execution time for:
    - direct loc: 3.28s
    - inverse loc: 48.5s
    - sun_angles: 0.272s
    - incidence_angles: 0.262s
Reference execution time for:
    - direct loc + sun + viewing angles:  3036 ms.
```

```
Differences zarr versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 14.2 (3.91)
        - altitude (m): 117 (15.1)
    - sun angles (angular diff (deg)): 0.000216 (5.86e-05)
    - incidence angles (angular diff (deg)): 0.000395 (0.000346)
    - inverse loc (pixel): 1.89e-10 (3.96e-16)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
```

```
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])


Execution time for:
    - direct loc: 4.13s
    - inverse loc: 52.5s
    - sun_angles: 0.164s
    - incidence_angles: 0.266s
Reference execution time for:
    - direct loc + sun + viewing angles:  3036 ms.
```

Direct locations a little far from reference => #204 and zarr results cannot be analysed as GETAS DEM is used for now instead of a the Copernicus GLOBE DEM converted in Zarr.

3.9.5.2 [S2MSI_TDS5_L0c_DEM-010]

[S2MSI_TDS5_L0c_DEM-010]
Summary: *This test aims at assessing that for L0c sensor model of S2MSI_TDS5 using DEM:* ✓ *ASGARD gives equivalent results for*

[S2MSI_TDS5_L0c_DEM-010]			
➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *S2MSI_TDS5: High Latitude*			
Preconditions: Reference has been generated with the Legacy processor			
#:	Step actions:	Expected Results:	Results:
1		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**

<table>
<tr><th colspan="4">[S2MSI_TDS5_L0c_DEM-010]</th></tr>
<tr><td rowspan="2"></td><td rowspan="2">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS5_L0c_ DEM_LEGACY]</td><td></td><td>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L0c_DEM_LEGACY]

Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>✓ PASSED with limitation: (Only footprint is impacted, wit 3 pixel of diff)
Differences versus ref

Max Differences from reference:
- direct loc:
- planar dist (m): 8.47e-09 (5.05e-09)
- altitude (m): 1.54e-09 (2.61e-10)
- sun angles (angular diff (deg)): 1.21e-06 (8.54e-07)
- incidence angles (angular diff (deg)): 0.000395 (0.000347)
- inverse loc (pixel): 7.79e-09 (1.4e-10)
- footprint:
- ratio: 0.998 (0.998)
- max dist (pixel): 3.47 (3.35)

List Sensor that differs from reference:
- footprint: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
'B11/D12' 'B12/D12']

Execution time for:
- direct loc: 117s
- inverse loc: 41.9s
- sun_angles: 0.139s
- incidence_angles: 0.0667s
- footprint: 3.39e+03s
Reference execution time for:
- direct loc + sun + viewing angles: 12502 ms.

Test run to the end
PASSED</td></tr>
</table>

<table>
<tr><th colspan="4">[S2MSI_TDS5_L0c_DEM-010]</th></tr>
<tr><td></td><td></td><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
<pre>Execution time for:
    - direct loc: 117s
    - inverse loc: 41.9s
    - sun_angles: 0.139s
    - incidence_angles: 0.0667s
    - footprint: 3.39e+03s
Reference execution time for:
    - direct loc + sun + viewing angles: 12502 ms.</pre></td></tr>
<tr><td rowspan="3"><u>2</u></td><td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS5_L0c_ DEM_REFACTORE D]</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
<pre>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L0c_DEM_REFACTORED]

Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</pre></td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.5.2.1 - Step 2.2:
✓ Using Legacy DEM:
➢ Direct locations are a little far from reference => <u>#204</u>
➢ Too high RAM consumption => <u>#199</u>
✓ Copernicus Zarr DEM not tested => <u>#197</u>
➢ Similar results with GETAS.zarr than using Legacy DEM</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Full process is not available as test was killed but here what can be seen for subset of tests (will be analysed in §3.10):
<pre>[asgard 2024-03-25 14:27:11 /home/aburie]
root$ python3 -m pytest asgard-legacy/tests/te</pre><pre>Sensor 'B11/D01' #12/156
Killed
[asgard 2024-03-25 15:53:26</pre></td></tr>
</table>

3.9.5.2.1 Step 2.2

Here is the end of the looping log with pyrugged issues:

```
Sensor 'B08/D01' #8/156
  Check versus reference
      ! Value superior for B08/D01 in planar error: [0.147 0.146 0.145 0.051 0.049 0.048 0.308 0.305 0.303]
  Check zarr versus reference
      ! Value superior for B08/D01 in planar error: [0.243 0.034 0.048 0.185 0.225 0.09  0.305 0.223 0.328]
      ! Value superior for B08/D01 in altitude error: [ 1.849  0.935  1.116  1.02   1.339  0.681  0.012  0.432 -0.147]
  Check versus zarr
      ! Value superior for B08/D01 in planar error: [0.389 0.179 0.192 0.214 0.256 0.117 0.003 0.083 0.025]
      ! Value superior for B08/D01 in altitude error: [-1.849 -0.935 -1.116 -1.02  -1.339 -0.681 -0.012 -0.432  0.147]
Sensor 'B8A/D01' #9/156
  Check versus reference
      ! Value superior for B8A/D01 in planar error: [0.115 0.115 0.114 0.035 0.035 0.034 0.232 0.231 0.229]
  Check zarr versus reference
      ! Value superior for B8A/D01 in planar error: [0.29  0.088 0.099 0.183 0.225 0.085 0.236 0.311 0.205]
      ! Value superior for B8A/D01 in altitude error: [ 1.847  0.935  1.117  1.028  1.344  0.684  0.017  0.435 -0.145]
  Check versus zarr
      ! Value superior for B8A/D01 in planar error: [0.391 0.18  0.194 0.217 0.259 0.119 0.004 0.084 0.025]
      ! Value superior for B8A/D01 in altitude error: [-1.847 -0.935 -1.117 -1.028 -1.344 -0.684 -0.017 -0.435  0.145]
Sensor 'B09/D01' #10/156
  Check versus reference
      ! Value superior for B09/D01 in planar error: [0.126 0.125 0.124 0.266 0.264 0.263 0.067 0.067 0.066]
  Check zarr versus reference
      ! Value superior for B09/D01 in planar error: [0.266 0.057 0.073 0.482 0.52  0.379 0.064 0.019 0.092]
      ! Value superior for B09/D01 in altitude error: [ 1.847  0.935  1.116  1.031  1.346  0.683  0.017  0.435 -0.147]
  Check versus zarr
      ! Value superior for B09/D01 in planar error: [0.392 0.181 0.195 0.219 0.26  0.119 0.004 0.084 0.026]
      ! Value superior for B09/D01 in altitude error: [-1.847 -0.935 -1.116 -1.031 -1.346 -0.683 -0.017 -0.435  0.147]
Sensor 'B10/D01' #11/156
  Check versus reference
      ! Value superior for B10/D01 in planar error: [0.105 0.104 0.103 0.075 0.075 0.074 0.297 0.295 0.293]
  Check zarr versus reference
      ! Value superior for B10/D01 in planar error: [0.335 0.141 0.151 0.143 0.184 0.046 0.303 0.38  0.27 ]
      ! Value superior for B10/D01 in altitude error: [ 1.847  0.937  1.118  1.036  1.354  0.695  0.025  0.444 -0.136]
  Check versus zarr
      ! Value superior for B10/D01 in planar error: [0.388 0.179 0.192 0.218 0.259 0.12  0.005 0.085 0.024]
      ! Value superior for B10/D01 in altitude error: [-1.847 -0.937 -1.118 -1.036 -1.354 -0.695 -0.025 -0.444  0.136]
Sensor 'B11/D01' #12/156
Killed
```

Issues observed:

- ✓ Direct locations are a little far from reference => #204
- ✓ Process killed due to too high RAM consumption (over machine limit) => #199

Zarr results can be due to GETAS instead of Copernicus DEM in Zarr => #197

3.9.5.3 [S2MSI_TDS5_L0c_CONST-010]

[S2MSI_TDS5_L0c_CONST-010]
Summary: *This test aims at assessing that for L0c sensor model of S2MSI_TDS5 at constant altitude:* ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data:

<table>
<tr><th colspan="4">[S2MSI_TDS5_L0c_CONST-010]</th></tr>
<tr><td colspan="4">S2MSI_TDS5: High Latitude</td></tr>
<tr><td colspan="4"><u>Preconditions:</u>
Reference has been generated with the Legacy processor</td></tr>
<tr><td><u>#:</u></td><td><u>Step actions:</u></td><td><u>Expected Results:</u></td><td><u>Results:</u></td></tr>
<tr><td rowspan="3"><u>1</u></td><td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS5_L0c_ CONST_LEGACY]</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2
✓ Verify that the altitude is set at 3000m</td><td>✓ OK:
<pre>asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L0c_CONST_LEGACY]

Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}
Constant altitude set at: 3000.0m.</pre></td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>3.9.5.3.1 - Step 1.2
✓ FAILED: (direct locations => #193)
<pre>Max Differences from reference:
    - direct loc:
        - planar dist (m): 8.74 (7.08)
        - altitude (m): 38.3 (36.9)
    - sun angles (angular diff (deg)): 0.000135 (0.000111)
    - incidence angles (angular diff (deg)): 0.000394 (0.000349)
    - inverse loc (pixel): 5.8e-09 (1.66e-10)</pre></td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
<pre>Execution time for:
    - direct loc: 30.7s
    - inverse loc: 0.861s
    - sun_angles: 0.171s
    - incidence_angles: 0.0632s
Reference execution time for:
    - direct loc + sun + viewing angles: 2626 ms.

Test run to the end
PASSED</pre></td></tr>
</table>

<table>
<tr><th colspan="4">[S2MSI_TDS5_L0c_CONST-010]</th></tr>
<tr>
<td rowspan="3">2</td>
<td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS5_L0c_ CONST_REFACTO RED]</td>
<td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2
✓ Verify that the altitude is set at 3000m</td>
<td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L0c_CONST_REFACTORED]

Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}
Constant altitude set at: 3000.0m.</td>
</tr>
<tr>
<td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td>
<td>Cf 3.9.5.3.2 - Step 2.2
FAILED: direct locations far from references => #193
✓ Using Legacy DEM
Differences versus ref

Max Differences from reference:
- direct loc:
- planar dist (m): 9 (7.12)
- altitude (m): 38.3 (36.9)
- sun angles (angular diff (deg)): 0.000139 (0.00011)
- incidence angles (angular diff (deg)): 0.000406 (0.000349)
- inverse loc (pixel): 1.3e+07 (5.69e+06)
✓ Using Zarr DEM (But no DEM used, so identical results):
Differences zarr versus ref

Max Differences from reference:
- direct loc:
- planar dist (m): 9 (7.12)
- altitude (m): 38.3 (36.9)
- sun angles (angular diff (deg)): 0.000139 (0.00011)
- incidence angles (angular diff (deg)): 0.000406 (0.000349)
- inverse loc (pixel): 1.25e+07 (5.72e+06)</td>
</tr>
<tr>
<td>✓ Verify that performances are logged and save results for dedicated performance analysis</td>
<td>✓ Time is monitored and will be analysed in §3.10:
➢ Using Legacy DEM:</td>
</tr>
</table>

[S2MSI_TDS5_L0c_CONST-010]			
			Execution time for: - direct loc: 1.38s - inverse loc: 319s - sun_angles: 0.229s - incidence_angles: 0.251s Reference execution time for: - direct loc + sun + viewing angles: 2626 ms. ➢ Using Zarr DEM: Execution time for: - direct loc: 49.4s Reference execution time for: - direct loc + sun + viewing angles: 2626 ms.

3.9.5.3.1 Step 1.2

Here is the summary log:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 8.74 (7.08)
        - altitude (m): 38.3 (36.9)
    - sun angles (angular diff (deg)): 0.000135 (0.000111)
    - incidence angles (angular diff (deg)): 0.000394 (0.000349)
    - inverse loc (pixel): 5.8e-09 (1.66e-10)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
```

```
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])


Execution time for:
    - direct loc: 30.7s
    - inverse loc: 0.861s
    - sun_angles: 0.171s
    - incidence_angles: 0.0632s
Reference execution time for:
    - direct loc + sun + viewing angles:  2626 ms.

Test run to the end
PASSED
```

Direct locations are far from references => #193

3.9.5.3.2 Step 2.2

Here are the global summaries:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 9 (7.12)
        - altitude (m): 38.3 (36.9)
    - sun angles (angular diff (deg)): 0.000139 (0.00011)
    - incidence angles (angular diff (deg)): 0.000406 (0.000349)
    - inverse loc (pixel): 1.3e+07 (5.69e+06)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
```

```
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
    - inverse loc: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 1.38s
    - inverse loc: 319s
    - sun_angles: 0.229s
    - incidence_angles: 0.251s
Reference execution time for:
    - direct loc + sun + viewing angles:  2626 ms.
```

```
Differences zarr versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 9 (7.12)
        - altitude (m): 38.3 (36.9)
    - sun angles (angular diff (deg)): 0.000139 (0.00011)
    - incidence angles (angular diff (deg)): 0.000406 (0.000349)
    - inverse loc (pixel): 1.25e+07 (5.72e+06)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
```

```
        - altitude (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
    - inverse loc: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12']


Execution time for:
    - direct loc: 49.4s
Reference execution time for:
    - direct loc + sun + viewing angles:  2626 ms.

Differences versus zarr
```

Direct location are a little far from references => #193

And zarr results are identical, as no DEM is used, due to constant altitude set.

3.9.5.4 [S2MSI_TDS5_L1B_DEM-010]

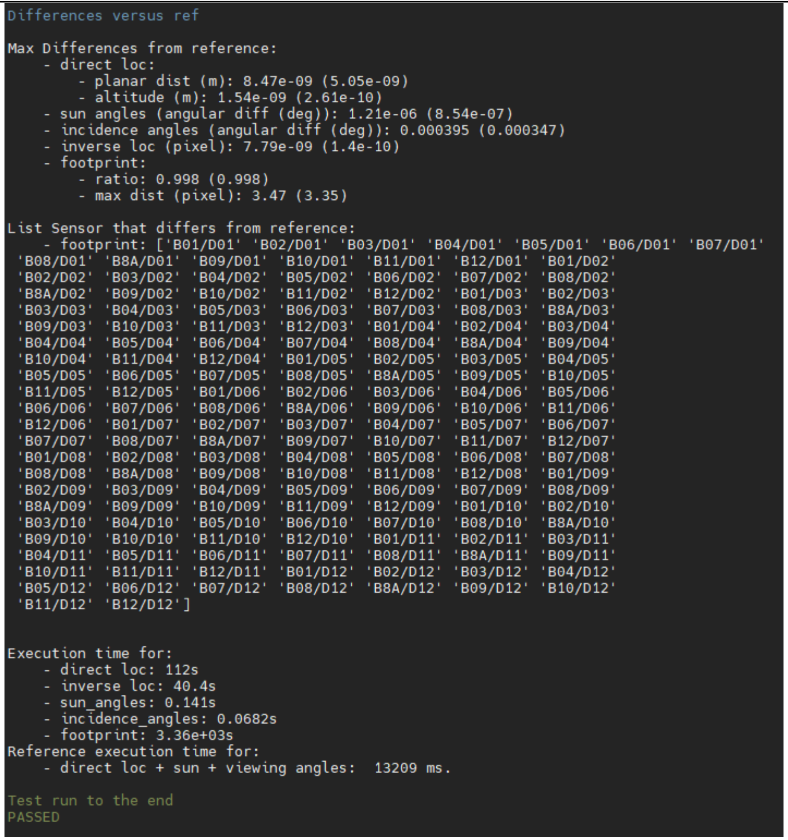
[S2MSI_TDS5_L1B_DEM-010]
Summary:

[S2MSI_TDS5_L1B_DEM-010]
This test aims at assessing that for L1B sensor model of S2MSI_TDS5 using DEM: ✓ *ASGARD gives equivalent results for* ➢ *Direct locations,* ➢ *Inverse locations,* ➢ *Sun angles computation,* ➢ *Viewing angles computation,* ➢ *Detector footprint generation.* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *S2MSI_TDS5: High Latitude*
Preconditions: Reference has been generated with the Legacy processor

[S2MSI_TDS5_L1B_DEM-010]			
#:	Step actions:	Expected Results:	Results:

1	**Run the Legacy-based implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L1B_DEM_LEGACY]`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	✓ **PASSED with limitations** as footprint is a little far from reference:

<table>
<tr><th colspan="4">[S2MSI_TDS5_L1B_DEM-010]</th></tr>
<tr><td></td><td rowspan="2">S2MSI_TDS5_L1B_
DEM_LEGACY]</td><td></td><td>
Differences versus ref

Max Differences from reference:
- direct loc:
- planar dist (m): 8.47e-09 (5.05e-09)
- altitude (m): 1.54e-09 (2.61e-10)
- sun angles (angular diff (deg)): 1.21e-06 (8.54e-07)
- incidence angles (angular diff (deg)): 0.000395 (0.000347)
- inverse loc (pixel): 7.79e-09 (1.4e-10)
- footprint:
- ratio: 0.998 (0.998)
- max dist (pixel): 3.47 (3.35)

List Sensor that differs from reference:
- footprint: ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
'B11/D12' 'B12/D12']

Execution time for:
- direct loc: 112s
- inverse loc: 40.4s
- sun_angles: 0.141s
- incidence_angles: 0.0682s
- footprint: 3.36e+03s
Reference execution time for:
- direct loc + sun + viewing angles: 13209 ms.

Test run to the end
PASSED</td></tr>
<tr><td></td><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:</td></tr>
</table>

<table>
<tr><th colspan="4">[S2MSI_TDS5_L1B_DEM-010]</th></tr>
<tr><td></td><td></td><td></td><td>Execution time for:
- direct loc: 112s
- inverse loc: 40.4s
- sun_angles: 0.141s
- incidence_angles: 0.0682s
- footprint: 3.36e+03s
Reference execution time for:
- direct loc + sun + viewing angles: 13209 ms.</td></tr>
<tr><td rowspan="3">2</td><td rowspan="3">Run the Legacy-based implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS5_L1B_ DEM_REFACTORE D]</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L1B_DEM_REFACTORED]
Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.5.4.1 - Step 2.2:
✓ Using Legacy DEM: FAILED
➢ Direct locations a little far from reference => #204
➢ Too high RAM consumption => #199
✓ Copernicus Zarr DEM not tested => #197
➢ Similar results with GETAS.zarr than using Legacy DEM</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10, however, only limited results are available as launched did not go up to the end:
[asgard 2024-03-25 11:21:33 /home/aburie]
root$ python3 -m pytest asgard-legacy/tests/te
==
platform linux -- Python 3.11.7, pytest-8.0.2,
Sensor 'B11/D01' #12/156
Killed
[asgard 2024-03-25 14:16:55 /home/aburie]</td></tr>
</table>

3.9.5.4.1 Step 2.2

Here is the start of the looping log:

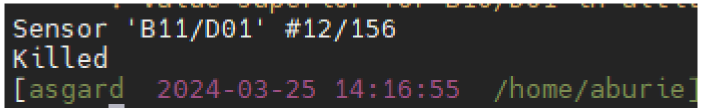
```
Read reference operation text file /DATA/TDS/S2MSI_TDS5/L1B_DEM_Legacy_s2geo_reference.txt
Sensor 'B01/D01' #1/156
  Check versus reference
      ! Value superior for B01/D01 in planar error: [0.091 0.09  0.09  0.035 0.034 0.034 0.176 0.174 0.173]
  Check zarr versus reference
      ! Value superior for B01/D01 in planar error: [0.3   0.09  0.105 0.197 0.238 0.1   0.172 0.09  0.199]
      ! Value superior for B01/D01 in altitude error: [ 1.847  0.934  1.116  1.031  1.347  0.684  0.017  0.435 -0.146]
  Check versus zarr
      ! Value superior for B01/D01 in planar error: [0.391 0.18  0.194 0.218 0.26  0.119 0.004 0.084 0.026]
      ! Value superior for B01/D01 in altitude error: [-1.847 -0.934 -1.116 -1.031 -1.347 -0.684 -0.017 -0.435  0.146]
Sensor 'B02/D01' #2/156
  Check versus reference
      ! Value superior for B02/D01 in planar error: [0.027 0.027 0.027 0.012 0.012 0.012 0.081 0.08  0.079]
  Check zarr versus reference
      ! Value superior for B02/D01 in planar error: [0.411 0.201 0.214 0.204 0.246 0.108 0.084 0.162 0.055]
      ! Value superior for B02/D01 in altitude error: [ 1.85   0.935  1.116  1.018  1.338  0.681  0.012  0.432 -0.147]
  Check versus zarr
      ! Value superior for B02/D01 in planar error: [0.389 0.179 0.192 0.214 0.256 0.117 0.002 0.083 0.025]
      ! Value superior for B02/D01 in altitude error: [-1.85  -0.935 -1.116 -1.018 -1.338 -0.681 -0.012 -0.432  0.147]
Sensor 'B03/D01' #3/156
  Check versus reference
      ! Value superior for B03/D01 in planar error: [0.012 0.013 0.013 0.258 0.256 0.254 0.047 0.047 0.047]
  Check zarr versus reference
      ! Value superior for B03/D01 in planar error: [0.39  0.18  0.194 0.472 0.511 0.371 0.05  0.13  0.022]
      ! Value superior for B03/D01 in altitude error: [ 1.849  0.935  1.116  1.021  1.34   0.681  0.013  0.433 -0.146]
  Check versus zarr
      ! Value superior for B03/D01 in planar error: [0.389 0.179 0.192 0.215 0.256 0.117 0.003 0.083 0.025]
      ! Value superior for B03/D01 in altitude error: [-1.849 -0.935 -1.116 -1.021 -1.34  -0.681 -0.013 -0.433  0.146]
Sensor 'B04/D01' #4/156
  Check versus reference
      ! Value superior for B04/D01 in planar error: [0.032 0.032 0.032 0.342 0.339 0.337 0.026 0.026 0.026]
  Check zarr versus reference
      ! Value superior for B04/D01 in planar error: [0.42  0.209 0.223 0.127 0.083 0.219 0.029 0.108 0.011]
      ! Value superior for B04/D01 in altitude error: [ 1.848  0.935  1.116  1.023  1.341  0.682  0.014  0.434 -0.146]
  Check versus zarr
      ! Value superior for B04/D01 in planar error: [0.389 0.179 0.193 0.216 0.257 0.118 0.003 0.083 0.025]
      ! Value superior for B04/D01 in altitude error: [-1.848 -0.935 -1.116 -1.023 -1.341 -0.682 -0.014 -0.434  0.146]
Sensor 'B05/D01' #5/156
  Check versus reference
      ! Value superior for B05/D01 in planar error: [0.027 0.027 0.027 0.001 0.001 0.001 0.326 0.324 0.322]
  Check zarr versus reference
      ! Value superior for B05/D01 in planar error: [0.367 0.157 0.171 0.217 0.258 0.119 0.323 0.244 0.346]
```

[…]

```
  Check versus zarr
      ! Value superior for B8A/D01 in planar error: [0.391 0.18  0.194 0.217 0.259 0.119 0.004 0.084 0.025]
      ! Value superior for B8A/D01 in altitude error: [-1.847 -0.935 -1.117 -1.028 -1.344 -0.684 -0.017 -0.435  0.145]
Sensor 'B09/D01' #10/156
  Check versus reference
      ! Value superior for B09/D01 in planar error: [0.126 0.125 0.124 0.266 0.264 0.263 0.067 0.067 0.066]
  Check zarr versus reference
      ! Value superior for B09/D01 in planar error: [0.266 0.057 0.073 0.482 0.52  0.379 0.064 0.019 0.092]
      ! Value superior for B09/D01 in altitude error: [ 1.847  0.935  1.116  1.031  1.346  0.683  0.017  0.435 -0.147]
  Check versus zarr
      ! Value superior for B09/D01 in planar error: [0.392 0.181 0.195 0.219 0.26  0.119 0.004 0.084 0.026]
      ! Value superior for B09/D01 in altitude error: [-1.847 -0.935 -1.116 -1.031 -1.346 -0.683 -0.017 -0.435  0.147]
Sensor 'B10/D01' #11/156
  Check versus reference
      ! Value superior for B10/D01 in planar error: [0.105 0.104 0.103 0.075 0.075 0.074 0.297 0.295 0.293]
  Check zarr versus reference
      ! Value superior for B10/D01 in planar error: [0.335 0.141 0.151 0.143 0.184 0.046 0.303 0.38  0.27 ]
      ! Value superior for B10/D01 in altitude error: [ 1.847  0.937  1.118  1.036  1.354  0.695  0.025  0.444 -0.136]
  Check versus zarr
      ! Value superior for B10/D01 in planar error: [0.388 0.179 0.192 0.218 0.259 0.12  0.005 0.085 0.024]
      ! Value superior for B10/D01 in altitude error: [-1.847 -0.937 -1.118 -1.036 -1.354 -0.695 -0.025 -0.444  0.136]
Sensor 'B11/D01' #12/156
Killed
```

Issues observed:

- ✓ Direct locations a little far from reference => #204
- ✓ Process is killed due to too high RAM consumption (over machine limit) => #199



3.9.5.5 [S2MSI_TDS5_L1C_DEM-010]

[S2MSI_TDS5_L1C_DEM-010]

<u>Summary:</u>

This test aims at assessing that for L1C sensor model of S2MSI_TDS5 using DEM:

- ✓ *ASGARD gives equivalent results for*
  - ➢ *Direct locations,*
  - ➢ *Inverse locations,*
  - ➢ *Sun angles computation,*
  - ➢ *Viewing angles computation,*
  - ➢ *Detector footprint generation.*
- ✓ *There is no large impact on processing time due to the new implementations.*

<u>Actors:</u>

CS GROUP

<u>Expected Results:</u>

[S2MSI_TDS5_L1C_DEM-010]			
Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds. <u>Pass/Fail criteria:</u> *One large increase of time is observed, or another metrics is higher than a defined threshold.* <u>Test Data:</u> *S2MSI_TDS5: High Latitude*			
<u>Preconditions:</u> Reference has been generated with the Legacy processor			
<u>#:</u>	<u>Step actions:</u>	<u>Expected Results:</u>	<u>Results:</u>

<u>1</u>	**Run the Legacy-based implementation with Legacy DEM:**
	Run command:
	✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[
		✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:**
			`asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L1C_DEM_LEGACY]`
			`Thresholds = {'ground': 1e-06, 'altitude': 1e-06, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}`

		✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)	✓ **PASSED:**

<table>
<tr><th colspan="4">[S2MSI_TDS5_L1C_DEM-010]</th></tr>
<tr><td rowspan="2"></td><td rowspan="2">S2MSI_TDS5_L1C_DEM_LEGACY]</td><td></td><td>Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 9.26e-09 (7.23e-09)
        - altitude (m): 1.78e-09 (1.83e-10)
    - sun angles (angular diff (deg)): 1.21e-06 (8.54e-07)
    - incidence angles (angular diff (deg)): 0.000393 (0.000345)
    - inverse loc (pixel): 3.29e-09 (7.19e-12)

List Sensor that differs from reference:

Execution time for:
    - direct loc: 79.2s
    - inverse loc: 16.5s
    - sun_angles: 0.119s
    - incidence_angles: 0.154s
Reference execution time for:
    - direct loc + sun + viewing angles: 25424 ms.

Test run to the end
PASSED</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
Execution time for:
    - direct loc: 79.2s
    - inverse loc: 16.5s
    - sun_angles: 0.119s
    - incidence_angles: 0.154s
Reference execution time for:
    - direct loc + sun + viewing angles: 25424 ms.</td></tr>
<tr><td>2</td><td>Run the Legacy-based implementation with Legacy DEM:</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L1C_DEM_REFACTORED]

Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td></tr>
</table>

<table>
<tr><th colspan="4">[S2MSI_TDS5_L1C_DEM-010]</th></tr>
<tr><td rowspan="2"></td><td rowspan="2">Run command:
✓ python3 -m pytest asgard-legacy /tests/test_sentinel2 _msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS5_L1C_DEM_REFACTORED]</td><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>Cf 3.9.5.5.1 - Step 2.2:
✓ Using Legacy DEM: PASSED with limitations
➢ Direct locations a little far from reference => #204
✓ Copernicus Zarr DEM not tested => #197
➢ Similar results with GETAS.zarr than using Legacy DEM</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
➢ Legacy DEM
<pre>Execution time for:
    - direct loc: 27.3s
    - inverse loc: 122s
    - sun_angles: 0.228s
    - incidence_angles: 0.235s
Reference execution time for:
    - direct loc + sun + viewing angles: 25424 ms.</pre>➢ Zarr DEM
<pre>Execution time for:
    - direct loc: 57.8s
Reference execution time for:
    - direct loc + sun + viewing angles: 25424 ms.</pre></td></tr>
</table>

3.9.5.5.1 Step 2.2

Here are the global summaries:

```
Differences versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 0.248 (0.159)
        - altitude (m): 0.00257 (0.000203)
    - sun angles (angular diff (deg)): 5.73e-06 (3.62e-06)
    - incidence angles (angular diff (deg)): 0.00041 (0.000346)
    - inverse loc (pixel): 1.07e-06 (4.61e-12)

List Sensor that differs from reference:
    - direct loc:
        - planar dist (m): ['B01/D01' 'B02/D01' 'B03/D01' 'B04/D01' 'B05/D01' 'B06/D01' 'B07/D01'
 'B08/D01' 'B8A/D01' 'B09/D01' 'B10/D01' 'B11/D01' 'B12/D01' 'B01/D02'
 'B02/D02' 'B03/D02' 'B04/D02' 'B05/D02' 'B06/D02' 'B07/D02' 'B08/D02'
 'B8A/D02' 'B09/D02' 'B10/D02' 'B11/D02' 'B12/D02' 'B01/D03' 'B02/D03'
 'B03/D03' 'B04/D03' 'B05/D03' 'B06/D03' 'B07/D03' 'B08/D03' 'B8A/D03'
 'B09/D03' 'B10/D03' 'B11/D03' 'B12/D03' 'B01/D04' 'B02/D04' 'B03/D04'
 'B04/D04' 'B05/D04' 'B06/D04' 'B07/D04' 'B08/D04' 'B8A/D04' 'B09/D04'
 'B10/D04' 'B11/D04' 'B12/D04' 'B01/D05' 'B02/D05' 'B03/D05' 'B04/D05'
 'B05/D05' 'B06/D05' 'B07/D05' 'B08/D05' 'B8A/D05' 'B09/D05' 'B10/D05'
 'B11/D05' 'B12/D05' 'B01/D06' 'B02/D06' 'B03/D06' 'B04/D06' 'B05/D06'
 'B06/D06' 'B07/D06' 'B08/D06' 'B8A/D06' 'B09/D06' 'B10/D06' 'B11/D06'
 'B12/D06' 'B01/D07' 'B02/D07' 'B03/D07' 'B04/D07' 'B05/D07' 'B06/D07'
 'B07/D07' 'B08/D07' 'B8A/D07' 'B09/D07' 'B10/D07' 'B11/D07' 'B12/D07'
 'B01/D08' 'B02/D08' 'B03/D08' 'B04/D08' 'B05/D08' 'B06/D08' 'B07/D08'
 'B08/D08' 'B8A/D08' 'B09/D08' 'B10/D08' 'B11/D08' 'B12/D08' 'B01/D09'
 'B02/D09' 'B03/D09' 'B04/D09' 'B05/D09' 'B06/D09' 'B07/D09' 'B08/D09'
 'B8A/D09' 'B09/D09' 'B10/D09' 'B11/D09' 'B12/D09' 'B01/D10' 'B02/D10'
 'B03/D10' 'B04/D10' 'B05/D10' 'B06/D10' 'B07/D10' 'B08/D10' 'B8A/D10'
 'B09/D10' 'B10/D10' 'B11/D10' 'B12/D10' 'B01/D11' 'B02/D11' 'B03/D11'
 'B04/D11' 'B05/D11' 'B06/D11' 'B07/D11' 'B08/D11' 'B8A/D11' 'B09/D11'
 'B10/D11' 'B11/D11' 'B12/D11' 'B01/D12' 'B02/D12' 'B03/D12' 'B04/D12'
 'B05/D12' 'B06/D12' 'B07/D12' 'B08/D12' 'B8A/D12' 'B09/D12' 'B10/D12'
 'B11/D12' 'B12/D12'])
        - altitude (m): ['B10/D01' 'B06/D06' 'B10/D06' 'B01/D07' 'B06/D07' 'B10/D07' 'B02/D08'
 'B07/D08' 'B06/D09' 'B03/D11' 'B09/D11' 'B10/D11' 'B07/D12' 'B11/D12'])


Execution time for:
    - direct loc: 27.3s
    - inverse loc: 122s
    - sun_angles: 0.228s
    - incidence_angles: 0.235s
Reference execution time for:
    - direct loc + sun + viewing angles:  25424 ms.
```

```
Differences zarr versus ref

Max Differences from reference:
    - direct loc:
        - planar dist (m): 12 (6.7)
        - altitude (m): 36.3 (35.3)
    - sun angles (angular diff (deg)): 0.000169 (8.69e-05)
    - incidence angles (angular diff (deg)): 0.000407 (0.000349)
    - inverse loc (pixel): 6.87e+15 (2.48e-16)
```

Direct locations a little far from reference => #204 and zarr results cannot be analysed as GETAS DEM is used for now #197

3.9.5.6 [S2MSI_TDS5_L1C_DEM_INVLOC-010]

[S2MSI_TDS5_L1C_DEM_INVLOC-010]
Summary:

[S2MSI_TDS5_L1C_DEM_INVLOC-010]

This test aims at assessing that for L1C sensor model using DEM:

- ✓ *ASGARD gives equivalent results for inverse location grids,*
- ✓ *There is no large impact on processing time due to the new implementations.*

Actors:

CS GROUP

Expected Results:

Comparison between metrics (in meters, pixels, degrees, seconds) between both versions are under the defined thresholds.

Pass/Fail criteria:

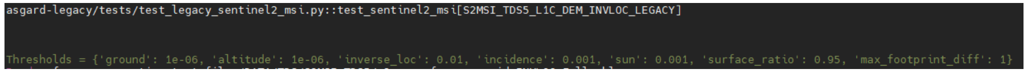
One large increase of time is observed, or another metrics is higher than a defined threshold.

Test Data:

TDS4: Equator

Preconditions:

Reference has been generated with the Legacy processor

#:	Step actions:	Expected Results:	Results:
1	**Run the Legacy-based implementation with Legacy DEM:**	✓ Verify that thresholds are the ones from the metrics table in §2.3.2	✓ **OK:** 

<table>
<tr><th colspan="4">[S2MSI_TDS5_L1C_DEM_INVLOC-010]</th></tr>
<tr><td rowspan="2"></td><td rowspan="2">Run command:
✓ python3 -m pytest asgard/tests/test_sentinel2_msi.py -s -v -k test_sentinel2_msi[S2MSI_TDS5_L1C_DEM_INVOLOC_LEGACY]</td><td>✓ Verify that all metrics except perf are under thresholds as described in the logs (cf §3.6 for logs understandings)</td><td>✓ PASSED:
Differences versus ref

Max Differences from reference:
    - inverse loc (pixel): 1.1e-08 (3.78e-19)

List Sensor that differs from reference:

Execution time for:
    - inverse_loc: 1.75e+03s
Reference execution time for:
    - inverse loc: 455733 ms.

Test run to the end
PASSED</td></tr>
<tr><td>✓ Verify that performances are logged and save results for dedicated performance analysis</td><td>✓ Time is monitored and will be analysed in §3.10:
Execution time for:
    - inverse_loc: 1.75e+03s
Reference execution time for:
    - inverse loc: 455733 ms.</td></tr>
<tr><td rowspan="2">2</td><td rowspan="2">Run the Refactored implementation with Legacy DEM:
Run command:
✓ python3 -m pytest asgard/tests/test_sentinel2_msi.py -s -v -</td><td>✓ Verify that thresholds are the ones from the metrics table in §2.3.2</td><td>✓ OK:
asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L1C_DEM_INVLOC_REFACTORED]

Thresholds = {'ground': 1e-06, 'altitude': 0.001, 'inverse_loc': 0.01, 'incidence': 0.001, 'sun': 0.001, 'surface_ratio': 0.95, 'max_footprint_diff': 1}</td></tr>
<tr><td>✓ Verify that all metrics except perf are under thresholds as described in the logs</td><td>Cf 3.9.5.6.1 - Step 2.2:
➢ Using Legacy DEM: FAILED
➢ Inverse locations are a little far from references => #203</td></tr>
</table>

[S2MSI_TDS5_L1C_DEM_INVLOC-010]			
	k test_sentinel2_msi[**S2MSI_TDS5_L1C_DEM_INVLOC_REFACTORED**]	(cf §3.6 for logs understandings)	➢ ValueError from a NoneType returned => #208 ✓ **Copernicus Zarr DEM not tested** => #197 ➢ Similar results with GETAS.zarr than using Legacy DEM
		✓ Verify that performances are logged and save results for dedicated performance analysis	✓ **Test not ending, but very very long**

3.9.5.6.1 Step 2.2

Here is the end of the looping log:

```
    Compute differences of previous steps.
Sensor 'B10/D03' #37/156
    Number of inverse locations: 2435
    Compute inverse loc grid.
      time ~ 53.327571398578584s
      ! Value superior for B10/D03 in inverse_loc error: [0.017 0.017 0.017 0.017 0.017 0.017 0.017 0.017 0.017 0.017] [...]
    Compute inverse loc grid with Zarr.
      time ~ 19.38646038994193s
      ! Value superior for B10/D03 in inverse_loc error: [0.032 0.033 0.035 0.034 0.034 0.033 0.033 0.032 0.032 0.031] [...]
    Compute differences of previous steps.
Sensor 'B11/D03' #38/156
    Number of inverse locations: 9385
    Compute inverse loc grid.
      time ~ 104.46375614311546s
      ! Value superior for B11/D03 in inverse_loc error: [0.05 0.05 0.05 0.05 0.05 0.05 0.05 0.05 0.05 0.05] [...]
    Compute inverse loc grid with Zarr.
      time ~ 73.65554749500006s
      ! Value superior for B11/D03 in inverse_loc error: [0.694 0.706 0.737 0.793 0.838 0.836 0.803 0.778 0.786 0.783] [...]
    Compute differences of previous steps.
      ! Value superior for B11/D03 in inverse_loc error: [0.216 0.222 0.241 0.27  0.29  0.289 0.272 0.259 0.263 0.261] [...]
Sensor 'B12/D03' #39/156
    Number of inverse locations: 9460
    Compute inverse loc grid.
      time ~ 162.80934090074152s
      ! Value superior for B12/D03 in inverse_loc error: [0.05  0.05  0.049 0.05  0.05  0.05  0.05  0.05  0.05  0.05 ] [...]
    Compute inverse loc grid with Zarr.
      time ~ 75.7511259894818s
      ! Value superior for B12/D03 in inverse_loc error: [0.757 0.769 0.804 0.868 0.921 0.921 0.889 0.855 0.859 0.853] [...]
    Compute differences of previous steps.
      ! Value superior for B12/D03 in inverse_loc error: [0.227 0.234 0.256 0.284 0.305 0.304 0.285 0.271 0.275 0.274] [...]
Sensor 'B01/D04' #40/156
    Number of inverse locations: 170310
    Compute inverse loc grid.
      time ~ 3160.242086957209s
      ! Value superior for B01/D04 in inverse_loc error: [0.017 0.017 0.017 0.017 0.017 0.017 0.017 0.017 0.017 0.017] [...]
    Compute inverse loc grid with Zarr.
      time ~ 2734.969460115768s
      ! Value superior for B01/D04 in inverse_loc error: [0.015 0.015 0.015 0.015 0.015 0.015 0.014 0.014 0.014 0.014] [...]
    Compute differences of previous steps.
      ! Value superior for B01/D04 in inverse_loc error: [0.001 0.001 0.001 0.    0.    0.    0.    0.    0.    0.   ] [...]
Sensor 'B02/D04' #41/156
    Number of inverse locations: 2703775
    Compute inverse loc grid.
  ! AttributeError(type(exp)=<class 'AttributeError'>): inverse loc 'NoneType' object has no attribute 'toArray'
      time ~ 44996.83504175674s
FAILED
```

```
>       return np.square((x_first - x_second) ** 2 + (y_first - y_second) ** 2)
E       ValueError: operands could not be broadcast together with shapes (540755,) (34062,)

asgard-legacy/tests/helpers/compare.py:316: ValueError
============================================ warnings summary ============================================
tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L1C_DEM_INVLOC_REFACTORED]
tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L1C_DEM_INVLOC_REFACTORED]
  /usr/local/lib/python3.11/site-packages/asgard/products/sentinel2/msi.py:697: DeprecationWarning: Support of native DEM will be removed, use Zarr versions
    self.propagation_model = PropagationModel(**model_kwargs)

-- Docs: https://docs.pytest.org/en/stable/how-to/capture-warnings.html
========================================== short test summary info =======================================
FAILED asgard-legacy/tests/test_legacy_sentinel2_msi.py::test_sentinel2_msi[S2MSI_TDS5_L1C_DEM_INVLOC_REFACTORED] - ValueError: operands could not be broadcast together with shapes (540755,) (34062,)
================================ 1 failed, 40 deselected, 2 warnings in 55124.47s (15:18:44) ================================
```

Issues observed:

- ✓ Inverse locations are a little far from references => [#203](#203)

- ✓ ValueError from a NoneType returned => #208

3.10 Performances tests results

3.10.1 [S2MSI_PERF]

[S2MSI_PERF]
Summary: *This test aims at assessing that:* ✓ *ASGARD gives equivalent results for Direct/Inverse locations sun/viewing angles computation,* ✓ *There is no large impact on processing time due to the new implementations.* Actors: *CS GROUP* Expected Results: *Comparison between metrics (in meters, degrees, seconds) between both versions are under the defined thresholds.* Pass/Fail criteria: *One large increase of time is observed, or another metrics is higher than a defined threshold.* Test Data: *S2MSI_TDS1,*

[S2MSI_PERF]			
S2MSI_TDS5			
Preconditions: Reference has been generated with the Legacy processor			
#:	Step actions:	Expected Results:	Results:
1	Collect all data from launched tests on: ✓ S2MSI_TDS1 ✓ S2MSI_TDS2 ✓ S2MSI_TDS3 ✓ S2MSI_TDS4 ✓ S2MSI_TDS5	✓ Compare results of the processing times	**Cf 3.10.1.1-Performance tables:** ✓ Todo (fill tables below and summarize them here)
		✓ Compare results of the RAM consumption	✓ **RAM is around 15Go for LEGACY-based implementation, versus very high (higher than 125Go for some TDS) for the REFACTORED implementation !!**

3.10.1.1 Performance tables

For now, only the processing time was looked at. In the following tables, ASGARD-LB will be used for ASGARD legacy-based implementation and ASGARD-R for ASGARD refactored implementation.

3.10.1.1.1 Direct location, sun angles and viewing angles

Direct locations, sun angles and viewing angles are regrouped in a single table as:

- ✓ For now, the legacy processing time is available as a group,
- ✓ They are similar computations with expected processing time low compared to inverse locations or detector footprint generation.

Table 18 – Processing time of direct locations, sun and viewing angles computation: sum of all (s)

TDS	Level	ASGARD Legacy-based	ASGARD Refactored DEM Legacy	ASGARD Refactored DEM Zarr	Reference
TDS1	L0c_DEM	**74,0937**	9,596	Not Available (issue on validation script corrected now)	2,89
	L0c_CONST	5,3212	1,783	Not Available (issue on validation script corrected now)	2,633
	L1B_DEM	**76,897**	9,764	Not Available (issue on validation script corrected now)	3,942
	L1C_DEM	**28,4937**	5,954	Not Available (issue on validation script corrected now)	3,136
TDS2	L0c_DEM	Not Available	Not Available	Not Available	Not Available
	L0c_CONST	**11,1351**	1,852	Not Available (issue on validation script corrected now)	2,324
	L1B_DEM	Not Available	Not Available	Not Available	Not Available
	L1C_DEM	**95,906**	Not Available	Not Available	5,283
TDS3	L1B_DEM	**111,2145**	Not Available	Not Available	6,865
TDS4	L1B_DEM	**170,2192**	Not Available	Not Available	13,567
TDS5	L0u_DEM	**33,7176**	4.178	4,56	3,036

TDS	Level	ASGARD Legacy-based	ASGARD Refactored DEM Legacy	ASGARD Refactored DEM Zarr	Reference
	L0c_DEM	117,2057	Not Available	Not Available	12,502
	L0c_CONST	30,9342	1,86	Not Available (issue on validation script corrected now)	2,626
	L1B_DEM	112,2092	Not Available	Not Available	13,209
	L1C_DEM	79,473	27,763	Not Available (issue on validation script corrected now)	25,424

Processing time has not been optimized yet; hence, we can see that the overlay is not neglectable, and can go higher that x10 but only for Legacy-Based. It shall be analysed after optimisation.

3.10.1.1.2 Inverse locations

Table 19 – Processing time of inverse locations (s)

TDS	Level	Inverse Location ASGARD-LB	Inverse Location ASGARD-R DEM Legacy	Inverse Location ASGARD-R DEM Zarr	Inverse Location Reference
TDS 1	L0c_DEM	1,8	29,4	Not Available (issue on validation script corrected now)	Not available
	L0c_CONST	0,356	36,6	Not Available (issue on validation script corrected now)	Not available

TDS	Level	Inverse Location ASGARD-LB	Inverse Location ASGARD-R DEM Legacy	Inverse Location ASGARD-R DEM Zarr	Inverse Location Reference
	L1B_DEM	2,13	29,9	Not Available (issue on validation script corrected now)	Not available
	L1C_DEM	1,36	35,1	Not Available (issue on validation script corrected now)	Not available
TDS2	L0c_DEM	Not available	Not available	Not available	Not available
	L0c_CONST	0,688	81,3	N Not Available (issue on validation script corrected now)	Not available
	L1B_DEM	Not available	Not available	Not available	Not available
	L1C_DEM	7,66	Not available	Not available	Not available
TDS3	L1B_DEM	88,9	Not available	Not available	Not available
TDS4	L1B_DEM	4,1	Not available	Not available	Not available
TDS 5	L0u_DEM	2,18	48.5	52.5	Not available
	L0c_DEM	41,9	Not available	Not available	Not available
	L0c_CONST	0,861	319	Not Available (issue on validation script corrected now)	Not available
	L1B_DEM	40,4	Not available	Not available	Not available

TDS	Level	Inverse Location ASGARD-LB	Inverse Location ASGARD-R DEM Legacy	Inverse Location ASGARD-R DEM Zarr	Inverse Location Reference
	L1C_DEM	16,5	112	Not Available (issue on validation script corrected now)	Not available

Inverse locations algorithm calls multiple direct locations. So, their processing time is partially linked to direct locations processing time. And for now, direct location processing is larger than reference, as explained in previous step. But when we look at inverse location grids:

Table 20 – Processing time of inverse locations grids (s)

TDS	Level	Refining	Inverse Location ASGARD-LB	Inverse Location ASGARD-R Legacy DEM	Inverse Location ASGARD-R Zarr DEM	Inverse Location Reference
TDS 1	L1C	REFINED	2 880		Done with GETAS	1 002
TDS 2	L1C	REFINED	1 960	35 365 for B02/D04	Done with GETAS	745
TDS 3	L1C	NOT_REFINED	682	20 514 for B02/D09	Done with GETAS	192
TDS 4	L1C	REFINED	2 050	42 880 for only B02/D02	Done with GETAS	751
TDS 5	L1C	NOT_REFINED	1 750	28 800 for only B02/D03	Done with GETAS	456

We can see that the processing time is very far from reference and is unacceptable yet:

- ✓ Legacy-based: There is a non-neglectable impact which can come from:
  - ➢ Overlay due to interfacing between python and java,
  - ➢ Processor are processed all bands at once in the Legacy reference, with optimisations, and only sensor per sensor in ASGARD, which implies a lack of one level of optimisation.
- ✓ Refactored: Not optimised yet, but for now it is unacceptable, and it won't be possible to put it in operation like that. It might only be due to direct location, but improvements are foreseen in future versions (on direct locations and other parts). Final assessment will be done in future version.

However, it seems that from observed uncomplete results of using GETAS DEM in Zarr take the same processing time that using the Legacy_DEM with the Refactored implementation. It is not expected to have a better processing time with Copernicus DEM in Zarr instead of GETAS one.

3.10.1.1.3 Detector footprint

Table 21 – Processing time of detector footprints computations (s)

TDS	Level	Detector footprint ASGARD-LB	Detector footprint ASGARD-R Legacy DEM	Detector footprint ASGARD-R Zarr DEM	Detector footprint Reference
TDS 1	L0c_DEM	13.8	283	276	Not available
	L1B_DEM	14.2	291	285	Not available
TDS 2	L0c_DEM	Not available	Not available	Not available	Not available
	L1B_DEM	Not available	Not available	Not available	Not available
TDS 3	L1B_DEM	2600	Not available	Not available	Not available
TDS 4	L1B_DEM	3890	Not available	Not available	Not available
TDS 5	L0c_DEM	3390	Not available	Not available	Not available
	L1B_DEM	3360	Not available	Not available	Not available

S2MSI_TDS1 is a small TDS with a limited location on DEM and so a limited number of DEM tiles crossed.

S2MSI_TDS 3 is a long “nominal” case, which takes a non-neglectable time.

S2MSI_TDS 4 and 5 are long datastrips in high latitude, which means that they crosse a lot of DEM tiles, which might be the root cause of a longer processing time even if they are shorter than S2MSI_TDS3 in acquisition time.

Processing time for this part is high, however, references are not available. It should be deeper analysed when final implementation is ready with Zarr DEM used and optimisation done both on processing time (on direct locations used during footprint generation) and on RAM consumption.

4. Sentinel 3 OLCI legacy based

4.1 Legacy processors

Table below presents all processors that must be supported by ASGARD because geometric functionalities are involved in.

Table 22 –Sentinel-3 legacy function link

Mission	Level	Instrument	Processor	Processor Version	IPF Processing Baseline3	Comments
S3	L1	OLCI	S3_OL1	6.17	3.23	
S3	L1	OLCI	S3_OL1_RAC	6.15	3.23	

4.2 ASGARD implementations using EOCFI

ASGARD comes with an implementation of several products and models relying on EOCFI for OLCI S3 product (legacy-based implementation under **asgard-legacy** project).

It will become deprecated once Rugged/Orekit based implementation will support those sensors (Refactored implementation, see § *5 Sentinel 3 OLCI refactored*).

In order to compare results from legacy processor based on EOCFI and ASGARD results, the legacy processors have been launched with a JobOrder containing all necessary files (ADF, L0 product, orbit file, attitude file) in input to compute the level 1 product of each sensor. Then we manage to load and use the same inputs in ASGARD to compute direct location, sun angles, incidence angles and compare those results with associated Breakpoints generated by the legacy processor. The comparison is then based and acceptance thresholds which are detailed in § *2.3.32.3.3.1 Sentinel-3 OLCI legacy based.*

Only the Legacy-based implementation is tested here.

4.3 Validation scope

Validation of geometric processor covers the following ICPs modules in which ASGARD is involved:

- ✓ For Earth Observation mode EO:
  - ➢ Data extraction and quality checks (O1-DE)
  - ➢ Geo-referencing (O1-GR)
  - ➢ Spatial re-sampling (O1-SRS)
- ✓ For radiometric calibration mode RAC:
  - ➢ Data extraction and quality checks (OC-DE)
  - ➢ Acquisition geometry (OC-GE)

4.4 Validation tests datasets

4.4.1 S3_OLCI_TDS1.1 - EO mode

The sensing time of processor legacy is defined at the beginning of the JobOrder:

- ✓ Start: 2022-05-13T03:50:40.00000
- ✓ Stop: 2022-05-13T00:37:04.000000

Table 23 – Input dataset S3_OLCI_TDS1.1

Files name	Type
S3A_AX___FRO_AX_20220510T000000_20220520T000000_20220513T065145___________________EUM_O_AL_001.SEN3	Orbit
S3A_TM_0_NAT____20220513T001317_20220513T015537_20220513T021511_6139_085_159______PS1_O_AL_004.SEN3	Navatt
S3A_OL_0_EFR____20220513T003504_20220513T003704_20220513T021703_0119_085_159______PS1_O_NR_002.SEN3	L0 product
S3A_AX___OSF_AX_20160216T192404_99991231T235959_20220330T090651___________________EUM_O_AL_001.SEN3	Orbit
S3__AX___MF1_AX_20220512T180000_20220513T060000_20220512T175950___________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20220513T000000_20220513T120000_20220512T181114___________________ECW_O_NR_001.SEN3	ADF
S3A_OL_1_INS_AX_20201030T120000_20991231T235959_20220505T120000___________________MPC_O_AL_009.SEN3	ADF
S3A_OL_1_CAL_AX_20220930T000000_20991231T235959_20221110T120000___________________MPC_O_AL_027.SEN3	ADF
S3A_OL_1_PRG_AX_20160425T095210_20991231T235959_20210309T120000___________________MPC_O_AL_004.SEN3	ADF
S3A_OL_1_CLUTAX_20160425T095210_20991231T235959_20160525T120000___________________MPC_O_AL_003.SEN3	ADF
S3__AX___DEM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___LWM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___OOM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___CLM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF

Files name	Type
S3__AX___TRM_AX_20000101T000000_20991231T235959_20151214T120000___ ________________MPC_O_AL_001.SEN3	ADF
Orbit.xml attitude.xml	Orbit and attitude breakpoints
BP_O1-DE_3-1.nc BP_O1-DE_4-1.nc BP_O1-GR_4-1.nc	Time and geolocation breakpoints

4.4.2 S3_OLCI_TDS1.2 - RAC mode

The sensing time of processor legacy is defined at the beginning of the JobOrder:

- ✓ Start: 2022-05-11T23:45:25.000000
- ✓ Stop: 2022-05-11T23:47:19.000000

Table 24 – Input dataset S3_OLCI_TDS1.2

Files name	Type
S3A_OL_0_CR0____20220511T234525_20220511T234719_20220512T005839_0113_085_144______PS1_O_NR_002.SEN3	L0 product
S3A_TM_0_NAT____20220513T001317_20220513T015537_20220513T021511_6139_085_159______PS1_O_AL_004.SEN3	Navatt
S3A_AX___FRO_AX_20220510T000000_20220520T000000_20220513T065145___________________EUM_O_AL_001.SEN3	Orbit
S3A_AX___OSF_AX_20160216T192404_99991231T235959_20220330T090651___________________EUM_O_AL_001	Orbit
S3A_OL_1_INS_AX_20201030T120000_20991231T235959_20220505T120000___________________MPC_O_AL_009	ADF
S3A_OL_1_CAL_AX_20230620T000000_20991231T235959_20230616T120000___________________MPC_O_AL_028	ADF
S3A_OL_1_PRG_AX_20160425T095210_20991231T235959_20210309T120000___________________MPC_O_AL_004	ADF
Orbit.xml Attitude.xml	Orbit and attitude breakpoint
BP_OC-DE_3-1.nc BP_OC-DE_4-1.nc BP_OC-GE_4-1.nc BP_OC-GE_5-1.nc	Time and geolocation breakpoints

4.5 Validation tests list

Those legacy steps referring to DPM sections are covered by ASGARD legacy based validations scripts located in tests/validations/ (see § *4.3 Validation scope*)

Table 25 – S3 OLCI L1 validation tests list

Processors	Validation script	Test Name
S3_OL1	tests/validations/test_sentinel3_olci_validation.py: ✓ test_direct_loc_all_cam ✓ test_sun_angles_all_cam ✓ test_incidence_angles_all_cam	[S3OLCI_TDS1_EO_LEGACY] - Geo referencing
S3_OL1_RAC	tests/validations/test_sentinel3_olci_rac_validation.py: ✓ test_val_instrument_to_sun	[S3OLCI_TDS1_RAC_LEGACY] - Acquisition Geometry

4.6 Features not tested

For this version, the inverse location functions are not tested as they are based on iterative inversion of direct location.

4.7 Quality tests results

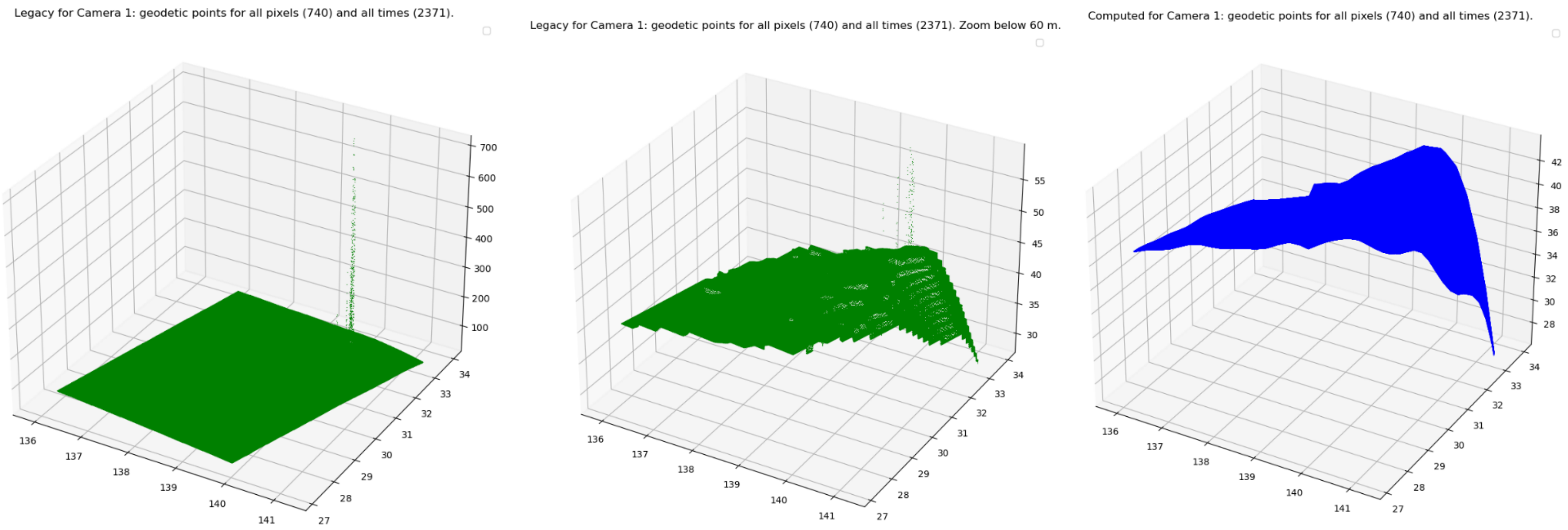
4.7.1 [S3OLCI_TDS1_EO_LEGACY] - Geo referencing

[S3OLCI_TDS1_EO_LEGACY] - Geo referencing
Summary: *This test aims at testing the geometric functions involved in OLCI legacy based geo-referencing step of the Earth Observation (EO) mode step.* Actors: *CS GROUP* Expected Results: ✓ Verify *that difference between geodetic points computed and geodetic points from legacy is below threshold as defined in § 2.3.3.1* Sentinel-3 OLCI legacy based ✓ *Verify that difference between sun angles/incidence angles computed and sun angles/incidence angles from legacy is below threshold which is defined by angles requirement in § 2.3.3.1* Sentinel-3 OLCI legacy based Pass/Fail criteria: ✓ *For direct location, between the computed and legacy geodetic coordinates:* ➢ *No planimetric distance errors at iso altitude are above requirement,* ➢ *No altitude differences are above requirement.* ✓ *For Sun angles (azimuth phi, zenith theta) between computed and legacy* ➢ *No direction errors are above requirement.*

[S3OLCI_TDS1_EO_LEGACY] - Geo referencing			
✓ *For incidence angles (azimuth phi, zenith theta)) between computed and legacy* ➢ *No direction errors are above requirement.* Test Data: *S3_OLCI_TDS1.1 (§ 4.4.1)*			
Preconditions: Reference has been generated with the Legacy processor S3_OL1			
#:	Step actions:	Expected Results:	Results:
1	In ***tests/validations/test_sentinel3_olci_validation.py*** run: ***test_direct_loc_all_cam***	In comparison with legacy results for Earth Observation mode: ✓ no planimetric error at iso altitude, ✓ no altitude difference. between the legacy geodetic and the computed geodetic coordinates are above requirement as expressed in § 2.3.3.1, for each camera, each pixel, and each date.	Number of geodetic points above requirement (upon 740 pixels * 2731 dates = 2020940 geodetic points): ✓ Planimetric errors at iso altitude: ➢ For camera 1: 766687 (error max = 647m) ➢ For camera 2: 19452 (error max = 111m) ➢ For camera 3: 0 (error max = 7.5m) ➢ For camera 4: 0 (error max = 5.6m) ➢ For camera 5: 0 (error max = 9.7m) ✓ Altitude differences: ➢ For camera 1: 655 (error max = 674 m) ➢ For camera 2: 1073 (error max= 274m) ➢ For camera 3: 0 (error max = 3m) ➢ For camera 4: 0 (error max = 3.8m) ➢ For camera 5: 0 (error max = 3.6m)

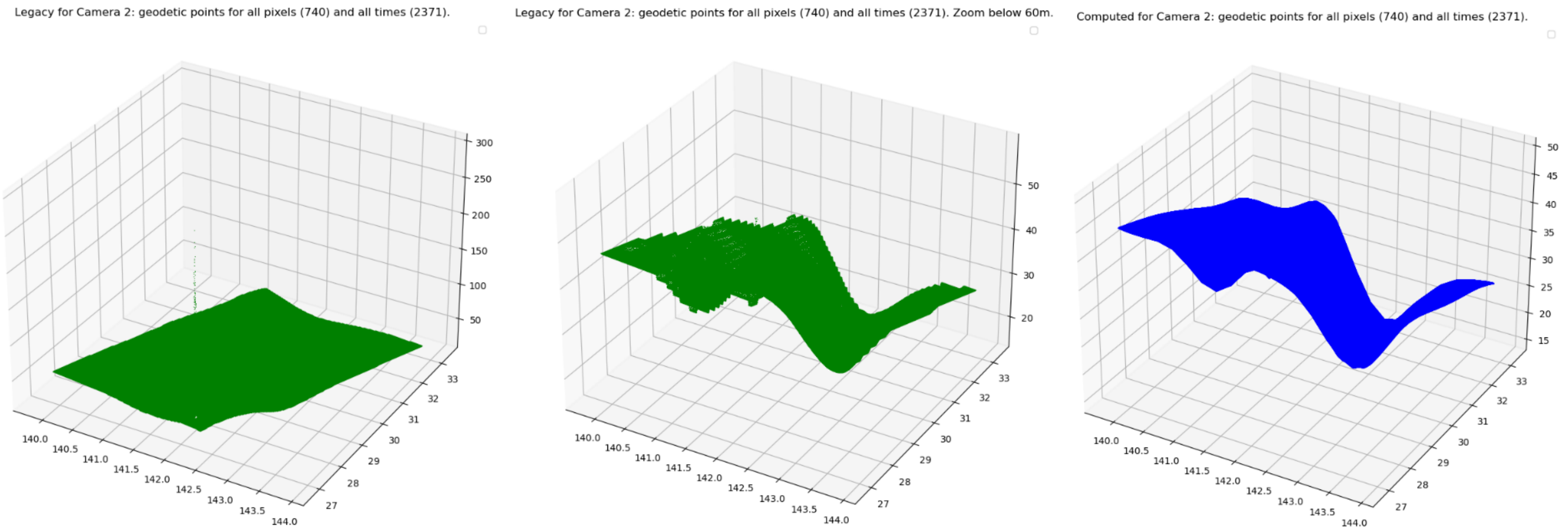
[S3OLCI_TDS1_EO_LEGACY] - Geo referencing			
			However, for camera 1 and 2, the issue is not on ASGARD but on legacy, **legacy results show aberrant geodetic points** as shown in Figure 1 & Figure 2 below. ➔ The test can then be considered as **PASSED**
2	In ***tests/validations/test_sentinel3_olci_validation.py*** run: ***test_sun_angles_all_cam***	✓ In comparison with legacy results for Earth Observation mode, for Sun angles in OLCI frame (phiSf (azimuth angle), thetaSf (zenith angle)), no direction errors between computed and legacy are above requirement as expressed in § 2.3.3.1 for each camera, each pixel, and each date.	Number of direction errors of sun angles above requirement (upon 740 pixels * 2731 dates = 2020940 angles): ✓ For camera 1: 0 (error max = 1.2e-6 deg) ✓ For camera 2: 0 (error max = 1.2e-6 deg) ✓ For camera 3: 0 (error max = 1.2e-6 deg) ✓ For camera 4: 0 (error max = 1.2e-6 deg) ✓ For camera 5: 0 (error max = 1.2e-6 deg) ➔ **PASSED**
3	In ***tests/validations/test_sentinel3_olci_validation.py*** run: ***test_incidence_angles_all_cam***	✓ In comparison with legacy results for Earth Observation mode, for incidence angles in OLCI frame (phiSv (azimuth angle), thetaSv (zenith angle)), no direction errors between computed and legacy are above requirement as expressed in § 2.3.3.1 for each camera, each pixel, and each date.	Number of direction errors of incidence angles above requirement (upon 740 pixels * 2731 dates = 2020940 angles): ✓ For camera 1: 0 (error max = 5.1e-4 deg) ✓ For camera 2: 0 (error max = 4.4e-4 deg) ✓ For camera 3: 0 (error max = 9.5e-5 deg) ✓ For camera 4: 0 (error max = 7.3e-6 deg) ✓ For camera 5: 0 (error max = 1.6e-4 deg) ➔ **PASSED**

Figure 1: S3OLCI_TDS_EO Camera 1 issue analysis



Legacy Legacy zoom below 60m ASGARD Legacy-based

Figure 2: S3OLCI_TDS_EO Camera 2 issue analysis



Legacy | Legacy zoom below 60m | ASGARD Legacy-based

4.7.2 [S3OLCI_TDS1_RAC_LEGACY] - Acquisition Geometry

[S3OLCI_TDS1_RAC_LEGACY] - Acquisition Geometry
Summary: *This test aims at testing the geometric functions involved in OLCI legacy based Acquisition Geometry step in the Radiometric Calibration (RAC) mode.*

[S3OLCI_TDS1_RAC_LEGACY] - Acquisition Geometry

Actors:

CS GROUP

Expected Results:

- ✓ *Verify that difference between sun angles computed and sun angles from legacy is below threshold which is defined by angles requirement in § 2.3.3.1* Sentinel-3 OLCI legacy based

Pass/Fail criteria:

For Instrument to sun angles (azimuth phi, zenith theta) between computed and legacy:

- ✓ *No direction errors are above requirement.*

Test Data:

S3_OLCI_TDS1.2 (§ 4.4.2)

Preconditions:

Reference has been generated with the Legacy processor S3_OL1_RAC

#:	Step actions:	Expected Results:	Results:
1	In ***tests/validations/test_sentinel3_olci_rac_validation.py*** Run	✓ In comparison with legacy results for RAC mode, for Sun angles in OLCI frame (phiSf (azimuth angle), thetaSf (incidence = zenith angle)), no direction errors between computed and legacy are above requirement as expressed in OLCI metrics (§ 2.3.3.1) for each camera, each pixel and each date.	Direction errors of sun angles above requirement (upon 536 directions): ✓ 0 (error max = 1.2e-06) ➔ **PASSED**

[S3OLCI_TDS1_RAC_LEGACY] - Acquisition Geometry			
	test_val_instrument_to_sun		

5. Sentinel 3 OLCI refactored

5.1 ASGARD implementations refactored

As seen in §*4 Sentinel 3 OLCI legacy based*, the OLCI code implementation was relying on models and products on EOCFI.

This was a first step in the development of ASGARD before the refactoring of the code.

The refactored implementation of ASGARD is based on Rugged/Orekit libraries through the ***pyrugged*** project (refactored implementation under **asgard** project).

In order to compare results from legacy processor based on EOCFI and ASGARD results, the legacy processors have been launched with a JobOrder containing all necessary files (ADF, L0 product, orbit file, attitude file) in input to compute the level 1 product of each sensor. Then we manage to load and use the same inputs in ASGARD to compute direct location, sun angles, incidence angles and compare those results with associated Breakpoints generated by the legacy processor. The comparison is then based and acceptance thresholds which are detailed in § 2.3.3.2 Sentinel-3 OLCI refactored.

Note that for the input datasets the only difference with OLCI legacy-based is the use of a DEM zarr which is going to be the new used format.

5.2 Validation scope

The validation scope is the same as the one for the § *4 Sentinel 3 OLCI legacy based* (see § *4.3* Validation scope).

5.3 Validation tests datasets

5.3.1 S3_OLCI_TDS1.1 - EO mode

The sensing time of processor legacy is defined at the beginning of the JobOrder:

- ✓ Start: 2022-05-13T03:50:40.00000
- ✓ Stop: 2022-05-13T00:37:04.000000

Table 26 – Input dataset S3_OLCI_TDS1.1

Files name	Type
S3A_AX___FRO_AX_20220510T000000_20220520T000000_20220513T065145___________________EUM_O_AL_001.SEN3	Orbit
S3A_TM_0_NAT____20220513T001317_20220513T015537_20220513T021511_6139_085_159______PS1_O_AL_004.SEN3	Navatt

Files name	Type
S3A_OL_0_EFR____20220513T003504_20220513T003704_20220513T021703_0119_085_159______PS1_O_NR_002.SEN3	L0 product
S3A_AX___OSF_AX_20160216T192404_99991231T235959_20220330T090651__________________EUM_O_AL_001.SEN3	Orbit
S3__AX___MF1_AX_20220512T180000_20220513T060000_20220512T175950__________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20220513T000000_20220513T120000_20220512T181114__________________ECW_O_NR_001.SEN3	ADF
S3A_OL_1_INS_AX_20201030T120000_20991231T235959_20220505T120000__________________MPC_O_AL_009.SEN3	ADF
S3A_OL_1_CAL_AX_20220930T000000_20991231T235959_20221110T120000__________________MPC_O_AL_027.SEN3	ADF
S3A_OL_1_PRG_AX_20160425T095210_20991231T235959_20210309T120000__________________MPC_O_AL_004.SEN3	ADF
S3A_OL_1_CLUTAX_20160425T095210_20991231T235959_20160525T120000__________________MPC_O_AL_003.SEN3	ADF
DEM : S0__ADF_GETAS_20000101T000000_21000101T000000_20230428T185052.zarr	ADF
S3__AX___LWM_AX_20000101T000000_20991231T235959_20151214T120000__________________MPC_O_AL_001.SEN3	ADF
S3__AX___OOM_AX_20000101T000000_20991231T235959_20151214T120000__________________MPC_O_AL_001.SEN3	ADF
S3__AX___CLM_AX_20000101T000000_20991231T235959_20151214T120000__________________MPC_O_AL_001.SEN3	ADF
S3__AX___TRM_AX_20000101T000000_20991231T235959_20151214T120000__________________MPC_O_AL_001.SEN3	ADF
Orbit.xml attitude.xml	Orbit and attitude breakpoints
BP_O1-DE_3-1.nc BP_O1-DE_4-1.nc BP_O1-GR_4-1.nc	Time and geolocation breakpoints

5.3.2 S3_OLCI_TDS1.2 - RAC mode

The sensing time of processor legacy is defined at the beginning of the JobOrder:

- ✓ Start: 2022-05-11T23:45:25.000000
- ✓ Stop: 2022-05-11T23:47:19.000000

Table 27 – Input dataset S3_OLCI_TDS1.2

Files name	Type
S3A_OL_0_CR0____20220511T234525_20220511T234719_20220512T00583 9_0113_085_144______PS1_O_NR_002.SEN3	L0 product
S3A_TM_0_NAT____20220513T001317_20220513T015537_20220513T02151 1_6139_085_159______PS1_O_AL_004.SEN3	Navatt
S3A_AX___FRO_AX_20220510T000000_20220520T000000_20220513T0651 45___________________EUM_O_AL_001.SEN3	Orbit
S3A_AX___OSF_AX_20160216T192404_99991231T235959_20220330T09065 1___________________EUM_O_AL_001	Orbit
S3A_OL_1_INS_AX_20201030T120000_20991231T235959_20220505T12000 0___________________MPC_O_AL_009	ADF
S3A_OL_1_CAL_AX_20230620T000000_20991231T235959_20230616T12000 0___________________MPC_O_AL_028	ADF
S3A_OL_1_PRG_AX_20160425T095210_20991231T235959_20210309T1200 00___________________MPC_O_AL_004	ADF
Orbit.xml Attitude.xml	Orbit and attitude breakpoint
BP_OC-DE_3-1.nc BP_OC-DE_4-1.nc BP_OC-GE_4-1.nc BP_OC-GE_5-1.nc	Time and geolocation breakpoints

5.4 Validation tests list

Those legacy steps referring to DPM sections are covered by ASGARD refactored validations scripts located in tests/validations/ (see §*5.2 Validation scope*)

Table 28 – S3 OLCI L1 validation tests list

Processors	Validation script	Test Name
S3_OL1	tests/validations/test_sentinel3_olci_validation_v2.py: ✓ test_direct_loc_all_cam_v2 ✓ test_sun_angles_all_cam ✓ test_incidence_angles_all_cam	[S3OLCI_TDS1_EO] - Geo referencing
S3_OL1_RAC	tests/validations/test_sentinel3_olci_rac_validation_v2.py: ✓ test_val_instrument_to_sun	[S3OLCI_TDS1_RAC] - Acquisition Geometry

5.5 Features not tested

For this version, the inverse location functions are not tested as they are based on iterative inversion of direct location.

5.6 Quality tests results

5.6.1 [S3OLCI_TDS1_EO] - Geo referencing

<table>
<tr><th colspan="4">[S3OLCI_TDS1_EO] - Geo referencing</th></tr>
<tr><td colspan="4"><u>Summary:</u>
This test aims at testing the geometric functions involved in OLCI refactored geo-referencing step of the Earth Observation (EO) mode step.

<u>Actors:</u>

CS GROUP

<u>Expected Results:</u>

✓ Verify that difference between geodetic points computed and geodetic points from legacy is below threshold as defined in § 2.3.3.2 Sentinel-3 OLCI refactored
✓ Verify that difference between sun angles/incidence angles computed and sun angles/incidence angles from legacy is below threshold which is defined by angles requirement in § 2.3.3.2 Sentinel-3 OLCI refactored

<u>Pass/Fail criteria:</u>

✓ For direct location, between the computed and legacy geodetic coordinates:
➢ No planimetric distance errors at iso altitude are above requirement,
➢ No altitude differences are above requirement.
✓ For Sun angles (azimuth phi, zenith theta) between computed and legacy
➢ No direction errors are above requirement.
✓ For incidence angles (azimuth phi, zenith theta)) between computed and legacy
➢ No direction errors are above requirement.

<u>Test Data:</u>

S3_OLCI_TDS1.1 (§ 5.3.1)</td></tr>
<tr><td colspan="4"><u>Preconditions:</u>
Reference has been generated with the Legacy processor S3_OL1</td></tr>
<tr><td><u>#:</u></td><td><u>Step actions:</u></td><td><u>Expected Results:</u></td><td><u>Results:</u></td></tr>
<tr><td><u>1</u></td><td>In
tests/validations/test_sentinel3_olci_validation_v2.py

run:</td><td>In comparison with legacy results for Earth Observation mode:
✓ no planimetric error at iso altitude,
✓ no altitude difference.
between the legacy geodetic and the computed geodetic coordinates are above requirement as expressed in §</td><td>Number of geodetic points above requirement (upon 740 pixels * 2731 dates = 2020940 geodetic points):
✓ Planimetric errors at iso altitude:
➢ For camera 1: 119378 (error max = 590 m)
➢ For camera 2: 854 (error max = 164 m)</td></tr>
</table>

[S3OLCI_TDS1_EO] - Geo referencing			
	test_direct_loc_all_cam_v2	2.3.3.2 for each camera, each pixel, and each date.	➢ For camera 3: 0 (error max = 70 m) ➢ For camera 4: 0 (error max = 68 m) ➢ For camera 5: 0 (error max = 72 m) ✓ Altitude differences: ➢ For camera 1: 655 (error max = 674 m) ➢ For camera 2: 1073 (error max= 274 m) ➢ For camera 3: 0 (error max = 3 m) ➢ For camera 4: 0 (error max = 3.8 m) ➢ For camera 5: 0 (error max = 3.6 m) **However**, for camera 1 and 2, the issue is not on ASGARD but on legacy, **legacy results show aberrant geodetic points** as shown in Figure 1 & Figure 2 in § 4.7.1. ➔ The test can then be considered as **PASSED**
2	In ***tests/validations/ test_sentinel3_olci_validation_v2.py*** run: ***test_sun_angles_all_cam***	✓ In comparison with legacy results for Earth Observation mode, for Sun angles in OLCI frame (phiSf (azimuth angle), thetaSf (zenith angle)), no direction errors between computed and legacy are above requirement as expressed in § 2.3.3.2 for each camera, each pixel, and each date.	Number of direction errors of sun angles above requirement (upon 740 pixels * 2731 dates = 2020940 angles): Using Orekit Sun position Computation: ✓ For camera 1: 2020940 (error max = 3.9e-4 deg) ✓ For camera 2: 2020940 (error max = 3.9e-4 deg) ✓ For camera 3: 2020940 (error max = 3.9e-4 deg) ✓ For camera 4: 2020940 (error max = 3.9e-4deg) ✓ For camera 5: 2020940 (error max = 3.9e-4 deg) Using EOCFI Sun position computation: ✓ For camera 1: 0 (error max = 1.2e-6 deg)

[S3OLCI_TDS1_EO] - Geo referencing			
			✓ For camera 2: 0 (error max = 1.2e-6 deg) ✓ For camera 3: 0 (error max = 1.2e-6 deg) ✓ For camera 4: 0 (error max = 1.2e-6 deg) ✓ For camera 5: 0 (error max = 1.2e-6 deg) ➔ The test can then be considered as **PASSED**
Comments: The error made on sun angles is about 3.87 10^{-4}, this is due to the fact that Orekit is using another method of sun position computation compared to EOCFI. By using EOCFI sun position computation, there are no errors on sun angles.			
3	In ***tests/validations/ test_sentinel3_olci_validation_v2.py*** run: ***test_incidence_angles_all_cam***	✓ In comparison with legacy results for Earth Observation mode, for incidence angles in OLCI frame (phiSv (azimuth angle), thetaSv (zenith angle)), no direction errors between computed and legacy are above requirement as expressed in § 2.3.3.2Sentinel-3 OLCI refactored for each camera, each pixel and each date.	Number of direction errors of incidence angles above requirement (upon 740 pixels * 2731 dates = 2020940 angles): ✓ For camera 1: 0 (error max = 0.000507 deg) ✓ For camera 2: 0 (error max = 0.00044 deg) ✓ For camera 3: 0 (error max = 9.52e-5 deg) ✓ For camera 4: 0 (error max = 7.29e-6 deg) ✓ For camera 5: 0 (error max = 0.00016 deg) **PASSED**
Comments: The error made on incidence angles is about 10^{-2}, this seems to be due to the fact that Orekit uses geodetic coordinates whereas EOCFI uses cartesian coordinates for topocentric frame definition			

5.6.2 [S3OLCI_TDS1_RAC] - Acquisition Geometry

[S3OLCI_TDS1_RAC] - Acquisition Geometry
Summary:

<table>
<tr><th colspan="4">[S3OLCI_TDS1_RAC] - Acquisition Geometry</th></tr>
<tr><td colspan="4">This test aims at testing the geometric functions involved in OLCI refactored Acquisition Geometry step in the Radiometric Calibration (RAC) mode.

<u>Actors:</u>

CS GROUP

<u>Expected Results:</u>

✓ Verify that difference between sun angles computed and sun angles from legacy is below threshold which is defined by angles requirement in § 02.3.3.2 Sentinel-3 OLCI refactored.

<u>Pass/Fail criteria:</u>

For Instrument to sun angles (azimuth phi, zenith theta) between computed and legacy:

✓ No direction errors are above requirement.

<u>Test Data:</u>

S3_OLCI_TDS1.2 (§ 5.3.2)</td></tr>
<tr><td colspan="4"><u>Preconditions:</u>
Reference has been generated with the Legacy processor S3_OL1_RAC</td></tr>
<tr><td><u>#:</u></td><td><u>Step actions:</u></td><td><u>Expected Results:</u></td><td><u>Results:</u></td></tr>
<tr><td><u>1</u></td><td>In tests/validations/test_sentinel3_olci_rac_validation_v2.py

Run
test_val_instrument_to_sun</td><td>✓ In comparison with legacy results for RAC mode, for Sun angles in OLCI frame (thetaSf (incidence = zenith angle), phiSf (azimuth angle)), no direction errors between computed and legacy are above requirement as expressed in OLCI metrics (§ 2.3.3.2) for each camera, each pixel and each date.</td><td>Direction errors of sun angles above requirement (upon 536 directions):
✓ 536 (error max = 0.0028521 °)

The test pass using sun position from EOCFI
➔ The test can then be considered as PASSED</td></tr>
<tr><td colspan="4"><u>Comments:</u>
As for <u>S3OLCI_TDS1_EO</u>, the sun position is computed with Orekit (which is more precise than EOCFI), the error is in average equals to 0.0028509 ° but using sun position from EOCFI the test pass.</td></tr>
</table>

6. SENTINEL 3 SLSTR LEGACY BASED

6.1 Legacy processors

Table below presents all processors that must be supported by ASGARD because geometric functionalities are involved in.

Table 29 –Sentinel-3 legacy function link

Mission	Level	Instrument	Processor	Processor Version	IPF Processing Baseline3	Comments
S3	L1	SLSTR	S3_SL1	6.21	3.23	

6.2 ASGARD implementations using EOCFI

ASGARD comes with an implementation of several products and models relying on EOCFI for SLSTR S3 product (legacy-based implementation under **asgard-legacy** project).

It will become deprecated once Rugged/Orekit based implementation will support those sensors (Refactored implementation, see §*7 SENTINEL 3 SLSTR refactored*).

In order to compare results from legacy processor based on EOCFI and ASGARD legacy-based results, the legacy processors have been launched with a JobOrder containing all necessary files (ADF, L0 product, orbit file, attitude file) in input to compute the level 1 product of each sensor. Then we manage to load and use the same inputs in ASGARD to compute direct location, sun angles, incidence angles and compare those results with associated Breakpoints generated by the legacy processor. The comparison is then based and acceptance thresholds which are detailed in § *2.3.3 Sentinel-3 thresholds*

Only the Legacy-based implementation is tested here.

6.3 Validation scope

Validation of geometric processor covers the following ICPs modules in which ASGARD is involved (in L1a processing):

- ✓ For Geo-localisation (S1-L1A_5)
- ✓ Position and attitude interpolation (S1-L1A_5_2)
- ✓ Instrument grid (S1-L1A_5_3)
- ✓ Tie point grid (S1-L1A_5_3)
- ✓ Quasi Cartesian Grid (S1-L1A_5_1 and S1-L1A_5_4)

Those legacy steps are referring to Detailed Processing Manual (DPM) sections which refers to geometric functions (which are pointed out in parenthesis in the above list).

6.4 Validation tests dataset

6.4.1 S3_SLSTR_TDS1

Table 30 – Input dataset S3_SLSTR_TDS1

Input files in JobOrder SL1	Type
S3A_SL_0_SLT____20221108T222346_20221108T222846_20221108T232245_0299_092_029______PS1_O_NR_004.SEN3 S3A_SL_0_SLT____20221108T222846_20221108T223346_20221108T232246_0299_092_029______PS1_O_NR_004.SEN3 S3A_SL_0_SLT____20221108T223346_20221108T223846_20221108T232244_0299_092_029______PS1_O_NR_004.SEN3	L0 product
S3A_TM_0_NAT____20221108T194311_20221108T212346_20221108T213932_6034_092_028______PS1_O_AL_004.SEN3 S3A_TM_0_NAT____20221108T212347_20221108T230512_20221108T232106_6084_092_029______PS1_O_AL_004.SEN3	Navatt
S3A_SL_1_VSC_AX_20221108T210522_20500101T000000_20221108T232410___________________PS1_O_NN____.SEN3	ADF
S3A_AX___FRO_AX_20221105T000000_20221115T000000_20221108T065251___________________EUM_O_AL_001.SEN3	Orbit
S3A_AX___OSF_AX_20160216T192404_99991231T235959_20220330T090651___________________EUM_O_AL_001.SEN3	Orbit
S3__AX___LWM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___DEM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___OOM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___CLM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___TRM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___MF1_AX_20221107T180000_20221108T060000_20221107T060347___________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20221108T000000_20221108T120000_20221107T060345___________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20221108T060000_20221108T180000_20221107T180655___________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20221108T120000_20221109T000000_20221107T180653___________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20221108T180000_20221109T060000_20221108T060316___________________ECW_O_NR_001.SEN3	ADF

Input files in JobOrder SL1	Type
S3__AX___MF1_AX_20221109T000000_20221109T120000_20221108T060314_ __________________ECW_O_NR_001.SEN3	
S3A_SL_1_PCP_AX_20160216T000000_20991231T235959_20211011T120000_ __________________MPC_O_AL_013.SEN3	ADF
S3A_SL_1_ANC_AX_20160216T000000_20991231T235959_20190912T120000_ __________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_N_F1AX_20160216T000000_20991231T235959_20170324T120000__ _________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_F2AX_20160216T000000_20991231T235959_20170324T120000__ _________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_S7AX_20160216T000000_20991231T235959_20170324T120000__ _________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_S8AX_20160216T000000_20991231T235959_20170324T120000__ _________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_S9AX_20160216T000000_20991231T235959_20170324T120000__ _________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_O_F1AX_20160216T000000_20991231T235959_20170324T120000__ _________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_O_F2AX_20160216T000000_20991231T235959_20170324T120000__ _________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_O_S7AX_20160216T000000_20991231T235959_20170324T120000_ __________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_O_S8AX_20160216T000000_20991231T235959_20170324T120000_ __________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_O_S9AX_20160216T000000_20991231T235959_20170324T120000_ __________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_S1AX_20160216T000000_20991231T235959_20180202T120000__ _________________MPC_O_AL_009.SEN3	ADF
S3A_SL_1_N_S2AX_20160216T000000_20991231T235959_20180202T120000__ _________________MPC_O_AL_009.SEN3	ADF
S3A_SL_1_N_S3AX_20160216T000000_20991231T235959_20180202T120000__ _________________MPC_O_AL_009.SEN3	ADF
S3A_SL_1_O_S1AX_20160418T094050_20991231T235959_20180202T120000_ __________________MPC_O_AL_011.SEN3	ADF
S3A_SL_1_O_S2AX_20160418T094050_20991231T235959_20180202T120000_ __________________MPC_O_AL_011.SEN3	ADF
S3A_SL_1_O_S3AX_20160418T094050_20991231T235959_20180202T120000_ __________________MPC_O_AL_011.SEN3	ADF
S3A_SL_1_NAS4AX_20160216T000000_20991231T235959_20180202T120000_ __________________MPC_O_AL_010.SEN3	ADF

Input files in JobOrder SL1	Type
S3A_SL_1_NAS5AX_20160216T000000_20991231T235959_20180202T120000_ _________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_NAS6AX_20160216T000000_20991231T235959_20180202T120000_ _________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_NBS4AX_20160216T000000_20991231T235959_20180202T120000_ _________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_NBS5AX_20160216T000000_20991231T235959_20180202T120000_ _________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_NBS6AX_20160216T000000_20991231T235959_20180202T120000_ _________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_OAS4AX_20160418T094050_20991231T235959_20180202T120000_ _________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OAS5AX_20160418T094050_20991231T235959_20180202T120000_ _________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OAS6AX_20160418T094050_20991231T235959_20180202T120000_ _________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OBS4AX_20160418T094050_20991231T235959_20180202T120000_ _________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OBS5AX_20160418T094050_20991231T235959_20180202T120000_ _________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OBS6AX_20160418T094050_20991231T235959_20180202T120000_ _________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_VIC_AX_20160216T000000_20991231T235959_20161012T120000__ ________________MPC_O_AL_004.SEN3	ADF
S3A_SL_1_GEO_AX_20160216T000000_20991231T235959_20190912T120000_ _________________MPC_O_AL_008.SEN3	ADF
S3A_SL_1_GEC_AX_20190101T000000_20991231T235959_20191010T120000_ _________________MPC_O_AL_009.SEN3	ADF
S3A_SL_1_CLO_AX_20160216T000000_20991231T235959_20180202T120000_ _________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_IRE_AX_20160216T000000_20991231T235959_20180202T120000__ ________________MPC_O_AL_001.SEN3	ADF
S3__SL_2_LSTBAX_20000101T000000_20991231T235959_20151214T120000__ ________________MPC_O_AL_001.SEN3	ADF
S3A_SL_1_LCC_AX_20160216T000000_20991231T235959_20201015T120000_ _________________MPC_O_AL_002.SEN3	ADF
S3A_SL_1_CDP_AX_20160216T000000_20991231T235959_20180202T120000_ _________________MPC_O_AL_001.SEN3	ADF
S3A_SL_1_CLP_AX_20160216T000000_20991231T235959_20180202T120000__ ________________MPC_O_AL_001.SEN3	ADF

Input files in JobOrder SL1	Type
S3A_SL_1_ADJ_AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_001.SEN3	ADF
S3A_SL_1_RTT_AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_001.SEN3	ADF
S3A_SL_1_ESSTAX_20160216T000000_20991231T235959_20180202T120000____________________MPC_O_AL_001.SEN3	ADF
attitude_scratch_file.xml orbit_scratch_file.xml	Orbit and attitude breakpoint
BP_S1_L1A_5_4_1_geo_with_outliers_NAD_1KM.nc, BP_S1_L1A_5_4_1_geo_with_outliers_NAD_A.nc, BP_S1_L1A_5_4_1_geo_with_outliers_NAD_B.nc, BP_S1_L1A_5_4_1_geo_with_outliers_NAD_F1.nc, BP_S1_L1A_5_4_1_geo_with_outliers_OBL_1KM.nc, BP_S1_L1A_5_4_1_geo_with_outliers_OBL_A.nc, BP_S1_L1A_5_4_1_geo_with_outliers_OBL_B.nc, BP_S1_L1A_5_4_1_geo_with_outliers_OBL_F1.nc	Instrument grid geolocation breakpoints
BP_S1_L1A_5_4_1_TPix_NAD_1KM.nc, BP_S1_L1A_5_4_1_TPix_NAD_A.nc, BP_S1_L1A_5_4_1_TPix_NAD_B.nc, BP_S1_L1A_5_4_1_TPix_OBL_1KM.nc, BP_S1_L1A_5_4_1_TPIx_OBL_A.nc, BP_S1_L1A_5_4_1_TPix_OBL_B.nc	Tie point grid geolocation and angles breakpoints
BP_S1_L1A_5_1_3_1.nc	Quasi-cartesian grid geolocation breakpoints

6.5 Validation tests list

The following ASGARD legacy based validations scripts located in tests/validations, cover the geometric functions specified in §6.3 -Validation scope.

Table 31 – S3 SLSTR L1 validation tests list

Processor	Validation script	Test Name
S3_SL1	tests/validations/test_sentinel3_slstr_validation.py:	
	✓ test_slstr_direct_loc_nad_1km_val ✓ test_slstr_direct_loc_all_val	✓ [S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)
	✓ test_slstr_tp_grid_val ✓ test_slstr_tp_angles_val	✓ [S3SLSTR_TDS1_TPG_LEGACY] - Tie Point Grid (S1-L1A_5_3)
	✓ test_slstr_qc_grid_loc_val	✓ [S3SLSTR_TDS1_QCG_LEGACY] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)

6.6 Features not tested

For this version, the inverse location functions are not tested as they are based on iterative inversion of direct location.

6.7 Quality tests results

6.7.1 [S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)

[S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)
Summary: *This test aims at testing the geometric functions involved in SLSTR legacy based geolocation on instrument grid.* Actors: *CS GROUP* Expected Results: ✓ *Verify that difference between geodetic points computed and geodetic points from legacy is below threshold which correspond to specific spatial resolution of each SLSTR geometry as defined in § 2.3.3.3* Sentinel-3 SLSTR legacy based Pass/Fail criteria: ✓ *For direct location, between the computed and legacy geodetic coordinates:* ➢ *No planimetric distance errors at iso altitude are above requirement,* ➢ *No altitude differences are above requirement.* Test Data: *S3_SLSTR_TDS1 (§ 6.4.1)*
Preconditions: Reference has been generated with the Legacy processor S3_SL1

[S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)			
#:	Step actions:	Expected Results:	Results:
1	Compare with legacy results of ground location and altitude for Nadir 1km grid first detector (NAD/1KM/0) with whole coordinates array, 902 x 1199 In ***tests/validations/ test_sentinel3_slstr_validation.py*** Run: ***test_slstr_direct_loc_nad_1km_val***	✓ Check that planetary distance difference between computed and legacy is below tolerance as expressed in SLSTR metric (*§ 2.3.3.3*) of 1km for 99.73% of points	✓ **PASSED**
		✓ Check that altitude difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§2.3.3.3*) for 99.73% of point	✓ **PASSED**
2	Compare with legacy results of ground location and altitude for different geometry with a subset of coordinates array (10%) In ***tests/validations/ test_sentinel3_slstr_validation.py*** Run: ***test_slstr_direct_loc_all_val*** with option: *--log-cli-level WARNING*	✓ Check that planetary distance difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.3*) for 99.73% of points	✓ NAD/1KM/0 **PASSED** ✓ NAD/1KM/1 **PASSED** ✓ OBL/1KM/0 **FAILED for 0.78% of points** ✓ OBL/1KM/1 **FAILED for 0.54% of points** ✓ NAD/1KM_F1/0 **FAILED for 25.55% of points** ✓ NAD/1KM_F1/1 **FAILED for 25.46% of points** ✓ OBL/1KM_F1/0 **PASSED** ✓ OBL/1KM_F1/1 **PASSED** ✓ NAD/05KM_A/0 **FAILED for 4.71% of points** ✓ NAD/05KM_A/1 **FAILED for 4.69% of points** ✓ NAD/05KM_A/2 **FAILED for 4.69% of points** ✓ NAD/05KM_A/3 **FAILED for 4.68% of points** ✓ NAD/05KM_B/0 **FAILED for 4.67% of points** ✓ NAD/05KM_B/1 **FAILED for 4.66% of points**

[S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)			
			✓ NAD/05KM_B/2 **FAILED for 4.63% of points** ✓ NAD/05KM_B/3 **FAILED for 4.62% of points**
			✓ OBL/05KM_A/0 **FAILED for 80.35% of points** ✓ OBL/05KM_A/1 **FAILED for 80.20% of points** ✓ OBL/05KM_A/2 **FAILED for 80.11% of points** ✓ OBL/05KM_A/3 **FAILED for 79.00% of points**
			✓ OBL/05KM_B/0 **FAILED for 80.38% of points** ✓ OBL/05KM_B/1 **FAILED for 80.23% of points** ✓ OBL/05KM_B/2 **FAILED for 80.07% of points** ✓ OBL/05KM_B/3 **FAILED for 79.87% of points**
		✓ Check that altitude difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.3)* for 99.73% of points	✓ NAD/1KM/0 **PASSED** ✓ NAD/1KM/1 **PASSED**
			✓ OBL/1KM/0 **PASSED** ✓ OBL/1KM/1 **PASSED**
			✓ NAD/1KM_F1/0 **PASSED** ✓ NAD/1KM_F1/1 **PASSED**
			✓ OBL/1KM_F1/0 **PASSED** ✓ OBL/1KM_F1/1 **PASSED**
			✓ NAD/05KM_A/0 **PASSED** ✓ NAD/05KM_A/1 **PASSED** ✓ NAD/05KM_A/2 **PASSED** ✓ NAD/05KM_A/3 **PASSED**
			✓ NAD/05KM_B/0 **PASSED** ✓ NAD/05KM_B/1 **PASSED** ✓ NAD/05KM_B/2 **PASSED**

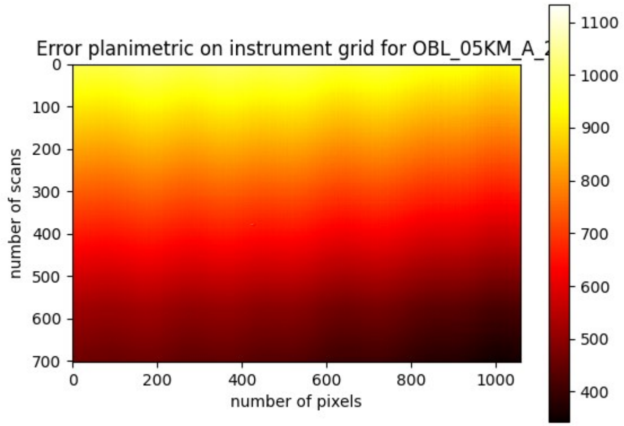
[S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)			
			✓ NAD/05KM_B/3 **PASSED**
			✓ OBL/05KM_A/0 **PASSED** ✓ OBL/05KM_A/1 **PASSED** ✓ OBL/05KM_A/2 **PASSED** ✓ OBL/05KM_A/3 **PASSED**
			✓ OBL/05KM_B/0 **PASSED** ✓ OBL/05KM_B/1 **PASSED** ✓ OBL/05KM_B/2 **PASSED** ✓ OBL/05KM_B/3 **PASSED**

Comments:

From the last version of this validation plan, we change the subset of coordinates for direct location, instead of taking the last 10% of values we have taken a value out of ten which is more relevant.

The error is made on 05KM group, and it seems that there is a problem with location using oblique view. This issue is investigated using source code ICF using EOCFI 4.23.

Here is an example of planimetric error for oblique view OBL/05KM_A/2 (to be compared to the refactored code which shows the same error pattern in §*7.6.1 [S3SLSTR_TDS1_IG] - Instrument Grid (S1-L1A_5_3)*):

[S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)


6.7.2 [S3SLSTR_TDS1_TPG_LEGACY] - Tie Point Grid (S1-L1A_5_3)

[S3SLSTR_TDS1_TPG_LEGACY] - Tie Point Grid (S1-L1A_5_3)
Summary: *This test aims at testing the geometric functions involved in SLSTR legacy based geolocation, sun angles, incidence angles computation on tie point grid (TPix).* Actors:

[S3SLSTR_TDS1_TPG_LEGACY] - Tie Point Grid (S1-L1A_5_3)
CS GROUP Expected Results: ✓ *Verify that difference between cartesian coordinates computed and cartesian coordinates points from legacy is below threshold which correspond to specific spatial resolution of each SLSTR geometry as defined in § 2.3.3.3* Sentinel-3 SLSTR legacy based ✓ *Verify that difference between sun angles/incidence angles computed and sun angles/incidence angles from legacy is below threshold which is defined by angles requirement in § 2.3.3.3* Sentinel-3 SLSTR legacy based Pass/Fail criteria: ✓ *For direct location, between the computed and legacy geodetic coordinates:* ➢ *No planimetric distance errors at iso altitude are above requirement,* ➢ *No altitude differences are above requirement.* ✓ *For Sun angles (azimuth phi, zenith theta) between computed and legacy:* ➢ *No direction errors are above requirement.* ✓ *For incidence angles (azimuth phi, zenith theta)) between computed and legacy* ➢ *No direction errors are above requirement.* Test Data: *S3_SLSTR_TDS1 (§6.4.1)*
Preconditions: Reference has been generated with the Legacy processor S3_SL1

#:	Step actions:	Expected Results:	Results:
1	Compare with legacy results of ground location and altitude for different	✓ Check that distance (norm computation) difference between computed and legacy is	✓ NAD/1KM/0 **FAILED for 4.61% of points** ✓ NAD/1KM/1 **FAILED for 4.72% of points**

[S3SLSTR_TDS1_TPG_LEGACY] - Tie Point Grid (S1-L1A_5_3)			
	geometry with whole coordinates array, 902x1199. In ***tests/validations/ test_sentinel3_slstr_validation.py*** Run: ***test_slstr_tp_grid_val*** with option: *--log-cli-level WARNING*	below tolerance as expressed in SLSTR metrics (*§ 2.3.3.3*) for 99.73% of points	✓ OBL/1KM/0 **FAILED for 46.31% of points** ✓ OBL/1KM/1 **FAILED for 46.08% of points** ✓ NAD/05KM_A/0 **FAILED for 17.78% of points** ✓ NAD/05KM_A/1 **FAILED for 17.60% of points** ✓ NAD/05KM_A/2 **FAILED for 17.72% of points** ✓ NAD/05KM_A/3 **FAILED for 18.24% of points** ✓ NAD/05KM_B/0 **FAILED for 17.68% of points** ✓ NAD/05KM_B/1 **FAILED for 17.65% of points** ✓ NAD/05KM_B/2 **FAILED for 17.71% of points** ✓ NAD/05KM_B/3 **FAILED for 18.17% of points** ✓ OBL/05KM_A/0 **FAILED for 98.24% of points** ✓ OBL/05KM_A/1 **FAILED for 98.19% of points** ✓ OBL/05KM_A/2 **FAILED for 98.24% of points** ✓ OBL/05KM_A/3 **FAILED for 98.17% of points** ✓ OBL/05KM_B/0 **FAILED for 98.23% of points** ✓ OBL/05KM_B/1 **FAILED for 98.24% of points** ✓ OBL/05KM_B/2 **FAILED for 98.24% of points** ✓ OBL/05KM_B/3 **FAILED for 98.24% of points**

Comments:

The reason of failing tests for direct loc on tie point grid needs to be investigated.

The direct_loc results for each tie point is expressed in longitude and latitude and the legacy results of geolocation on tie point grid is expressed as (X, Y) coordinates on quasi cartesian grid (and not expressed in longitude and latitude).

In order to validate properly direct location expressed in ground coordinates, we built this quasi-cartesian grid in ASGARD and computed (X, Y) coordinates on this grid from the longitude/latitude results but for now

- ➔ 95% of value errors is below tolerance for NAD/1KM,
- ➔ 85% for NAD/05KM

[S3SLSTR_TDS1_TPG_LEGACY] - Tie Point Grid (S1-L1A_5_3)			
➔ and only 1% for all oblique geometry. Hence more investigation needs to be done for oblique view using the version of the ICF code which is using EOCFI 4.23.			
2	Compare with legacy results of sun angles/incidence angles for different geometry with whole coordinates array of tie point grid of size (115x101 for nadir view and 141x40 for oblique view) In ***tests/validations/ test_sentinel3_slstr_validation.py*** Run: ***test_slstr_tp_angles_val*** with option: *--log-cli-level WARNING*	✓ Check that no sun angles difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.3*) for 99.73% of points. Sun angles is defined as: solar_phi, solar_theta	✓ NAD/1KM/0 **FAILED for 100% of points** ✓ NAD/1KM/1 **FAILED for 100% of points**
			✓ OBL/1KM/0 **FAILED for 100% of points** ✓ OBL/1KM/1 **FAILED for 100% of points**
			✓ NAD/05KM_A/0 **FAILED for 100% of points** ✓ NAD/05KM_A/1 **FAILED for 100% of points** ✓ NAD/05KM_A/2 **FAILED for 100% of points** ✓ NAD/05KM_A/3 **FAILED for 100% of points**
			✓ NAD/05KM_B/0 **FAILED for 100% of points** ✓ NAD/05KM_B/1 **FAILED for 100% of points** ✓ NAD/05KM_B/2 **FAILED for 100% of points** ✓ NAD/05KM_B/3 **FAILED for 100% of points**
			✓ OBL/05KM_A/0 **FAILED for 100% of points** ✓ OBL/05KM_A/1 **FAILED for 100% of points** ✓ OBL/05KM_A/2 **FAILED for 100% of points** ✓ OBL/05KM_A/3 **FAILED for 100% of points**
			✓ OBL/05KM_B/0 **FAILED for 100% of points** ✓ OBL/05KM_B/1 **FAILED for 100% of points** ✓ OBL/05KM_B/2 **FAILED for 100% of points** ✓ OBL/05KM_B/3 **FAILED for 100% of points**
		✓ Check that no incidence angles difference between computed and legacy is below	✓ NAD/1KM/0 **FAILED for 100% of points** ✓ NAD/1KM/1 **FAILED for 100% of points**

[S3SLSTR_TDS1_TPG_LEGACY] - Tie Point Grid (S1-L1A_5_3)			
		tolerance as expressed in SLSTR metrics (§ *2.3.3.3*) for 99.73% of points. Incidence angle is defined as: viewing_phi, viewing_theta	✓ OBL/1KM/0 **FAILED for 100% of points** ✓ OBL/1KM/1 **FAILED for 100% of points**
			✓ NAD/05KM_A/0 **FAILED for 100% of points** ✓ NAD/05KM_A/1 **FAILED for 100% of points** ✓ NAD/05KM_A/2 **FAILED for 100% of points** ✓ NAD/05KM_A/3 **FAILED for 100% of points**
			✓ NAD/05KM_B/0 **FAILED for 100% of points** ✓ NAD/05KM_B/1 **FAILED for 100% of points** ✓ NAD/05KM_B/2 **FAILED for 100% of points** ✓ NAD/05KM_B/3 **FAILED for 100% of points**
			✓ OBL/05KM_A/0 **FAILED for 100% of points** ✓ OBL/05KM_A/1 **FAILED for 100% of points** ✓ OBL/05KM_A/2 **FAILED for 100% of points** ✓ OBL/05KM_A/3 **FAILED for 100% of points**
			✓ OBL/05KM_B/0 **FAILED for 100% of points** ✓ OBL/05KM_B/1 **FAILED for 100% of points** ✓ OBL/05KM_B/2 **FAILED for 100% of points** ✓ OBL/05KM_B/3 **FAILED for 100% of points**

Comments:

The error on sun/incidence angle computation is around 10^{-3} ° so it is just above tolerance, this is due to the propagation error made on direct location. Indeed, the sun/incidence angles computation takes in input geodetic coordinates, as we do not have the legacy geodetic coordinates, we could not only check the error on sun/incidence angles computation, but we must use the geodetic coordinates computed with direct localisation function which already have an error on it. When the direct location will be fixed, the sun/incidence angle computation would pass because it involves the same code as OLCI sun/incidence angles computation.

6.7.3 [S3SLSTR_TDS1_QCG_LEGACY] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)

[S3SLSTR_TDS1_QCG_LEGACY] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)			
Summary: *This test aims at testing the geometric functions involved in SLSTR legacy based geolocation on quasi cartesian grid (QC).* Actors: *CS GROUP* Expected Results: *Verify that difference between cartesian coordinates computed and cartesian coordinates from legacy is below threshold which correspond to specific spatial resolution as defined in § 0* ✓ Sentinel-3 thresholds, *notice that this quasi-cartesian grid is independent of the geometry.* Pass/Fail criteria: ✓ *For direct location, between the computed and legacy geodetic coordinates:* ✓ *No planimetric distance errors at iso altitude are above requirement,* ✓ *No altitude differences are above requirement.* Test Data: *S3_SLSTR_TDS1 (§6.4.1)*			
Preconditions: Reference has been generated with the Legacy processor S3_SL1			
#:	Step actions:	Expected Results:	Results:

[S3SLSTR_TDS1_QCG_LEGACY] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)			
1	Compare with legacy results of ground location for Nadir 1km grid first detector (NAD/1KM/0) with whole coordinates array, 115x130. In ***tests/validations/ test_sentinel3_slstr_validation.py*** Run: ***test_slstr_qc_grid_loc_val***	✓ Check that planetary distance difference between computed and legacy is below tolerance as expressed in SLSTR metrics (§ *2.3.3.3*) for 99.73% of points	✓ **PASSED**

7. SENTINEL 3 SLSTR REFACTORED

7.1 ASGARD implementation refactored

As seen in §6-*Sentinel 3 SLSTR legacy based*, the SLSTR code implementation was relying on models and products on EOCFI. This was a first step in the development of ASGARD before the refactoring of the code.

The refactored implementation of ASGARD is based on Rugged/Orekit libraries through the ***pyrugged*** project (refactored implementation under **asgard** project).

In order to compare results from legacy processor based on EOCFI and ASGARD refactored results, the legacy processors have been launched with a JobOrder containing all necessary files (ADF, L0 product, orbit file, attitude file) in input to compute the level 1 product of each sensor. Then we manage to load and use the same inputs in ASGARD to compute direct location, sun angles, incidence angles and compare those results with associated Breakpoints generated by the legacy processor. The comparison is then based and acceptance thresholds which are detailed in § *2.3.3.4 Sentinel-3 SLSTR refactored*

The thresholds presented in the table below are the one for the refactored implementation for SLSTR instrument (different DEM as legacy processor).

Table 11 – Sentinel-3 SLSTR refactored metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation for grids (with geometry)	500 m 1 km	Depends on the geometry and the grid where the direct location is computed. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.

	Planimetric distance error at iso elevation for quasi cartesian grid	16 km	Independent of any geometry. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Altitude error	1 m	
Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-3 deg	
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-4 deg	

7.1.1.1 Sentinel-3 MWR legacy based

The thresholds presented in the table below are the one for the Legacy-Based implementation for MWR instrument.

Table 12 - Sentinel-3 MWR legacy based metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	15.5 m	The spatial resolution of MWR Full Resolution (FR) is 20 km. As we are in legacy-based context the results are expected to be almost the same as legacy processor.

Sentinel-3 MWR.

Note that for the input datasets the only difference with SLSTR legacy-based is the use of a DEM zarr which is going to be the new used format.

7.2 Validation scope

The validation scope is the same as the one for the Sentinel 3 SLSTR legacy based (see §6.3-Validation scope).

7.3 Validation tests dataset

7.3.1 S3_SLSTR_TDS1

Table 32 – Input dataset S3_SLSTR_TDS1

Input files in JobOrder SL1	Type
S3A_SL_0_SLT____20221108T222346_20221108T222846_20221108T232245_0299_092_029______PS1_O_NR_004.SEN3 S3A_SL_0_SLT____20221108T222846_20221108T223346_20221108T232246_0299_092_029______PS1_O_NR_004.SEN3 S3A_SL_0_SLT____20221108T223346_20221108T223846_20221108T232244_0299_092_029______PS1_O_NR_004.SEN3	L0 product
S3A_TM_0_NAT____20221108T194311_20221108T212346_20221108T213932_6034_092_028______PS1_O_AL_004.SEN3 S3A_TM_0_NAT____20221108T212347_20221108T230512_20221108T232106_6084_092_029______PS1_O_AL_004.SEN3	Navatt

Input files in JobOrder SL1	Type
S3A_SL_1_VSC_AX_20221108T210522_20500101T000000_20221108T232410___________________PS1_O_NN____.SEN3	ADF
S3A_AX___FRO_AX_20221105T000000_20221115T000000_20221108T065251___________________EUM_O_AL_001.SEN3	Orbit
S3A_AX___OSF_AX_20160216T192404_99991231T235959_20220330T090651___________________EUM_O_AL_001.SEN3	Orbit
S3__AX___LWM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
DEM : S0__ADF_GETAS_20000101T000000_21000101T000000_20230428T185052.zarr	ADF
S3__AX___OOM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___CLM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___TRM_AX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3__AX___MF1_AX_20221107T180000_20221108T060000_20221107T060347___________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20221108T000000_20221108T120000_20221107T060345___________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20221108T060000_20221108T180000_20221107T180655___________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20221108T120000_20221109T000000_20221107T180653___________________ECW_O_NR_001.SEN3 S3__AX___MF1_AX_20221108T180000_20221109T060000_20221108T060316___________________ECW_O_NR_001.SEN3	ADF

Input files in JobOrder SL1	Type
S3__AX___MF1_AX_20221109T000000_20221109T120000_20221108T060314___________________ECW_O_NR_001.SEN3	
S3A_SL_1_PCP_AX_20160216T000000_20991231T235959_20211011T120000___________________MPC_O_AL_013.SEN3	ADF
S3A_SL_1_ANC_AX_20160216T000000_20991231T235959_20190912T120000___________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_N_F1AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_F2AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_S7AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_S8AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_S9AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_O_F1AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_O_F2AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_O_S7AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_O_S8AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF

Input files in JobOrder SL1	Type
S3A_SL_1_O_S9AX_20160216T000000_20991231T235959_20170324T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_N_S1AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_009.SEN3	ADF
S3A_SL_1_N_S2AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_009.SEN3	ADF
S3A_SL_1_N_S3AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_009.SEN3	ADF
S3A_SL_1_O_S1AX_20160418T094050_20991231T235959_20180202T120000___________________MPC_O_AL_011.SEN3	ADF
S3A_SL_1_O_S2AX_20160418T094050_20991231T235959_20180202T120000___________________MPC_O_AL_011.SEN3	ADF
S3A_SL_1_O_S3AX_20160418T094050_20991231T235959_20180202T120000___________________MPC_O_AL_011.SEN3	ADF
S3A_SL_1_NAS4AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_NAS5AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_NAS6AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_NBS4AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_NBS5AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_010.SEN3	ADF

Input files in JobOrder SL1	Type
S3A_SL_1_NBS6AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_010.SEN3	ADF
S3A_SL_1_OAS4AX_20160418T094050_20991231T235959_20180202T120000___________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OAS5AX_20160418T094050_20991231T235959_20180202T120000___________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OAS6AX_20160418T094050_20991231T235959_20180202T120000___________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OBS4AX_20160418T094050_20991231T235959_20180202T120000___________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OBS5AX_20160418T094050_20991231T235959_20180202T120000___________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_OBS6AX_20160418T094050_20991231T235959_20180202T120000___________________MPC_O_AL_012.SEN3	ADF
S3A_SL_1_VIC_AX_20160216T000000_20991231T235959_20161012T120000___________________MPC_O_AL_004.SEN3	ADF
S3A_SL_1_GEO_AX_20160216T000000_20991231T235959_20190912T120000___________________MPC_O_AL_008.SEN3	ADF
S3A_SL_1_GEC_AX_20190101T000000_20991231T235959_20191010T120000___________________MPC_O_AL_009.SEN3	ADF
S3A_SL_1_CLO_AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_006.SEN3	ADF
S3A_SL_1_IRE_AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_001.SEN3	ADF

Input files in JobOrder SL1	Type
S3__SL_2_LSTBAX_20000101T000000_20991231T235959_20151214T120000___________________MPC_O_AL_001.SEN3	ADF
S3A_SL_1_LCC_AX_20160216T000000_20991231T235959_20201015T120000___________________MPC_O_AL_002.SEN3	ADF
S3A_SL_1_CDP_AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_001.SEN3	ADF
S3A_SL_1_CLP_AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_001.SEN3	ADF
S3A_SL_1_ADJ_AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_001.SEN3	ADF
S3A_SL_1_RTT_AX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_001.SEN3	ADF
S3A_SL_1_ESSTAX_20160216T000000_20991231T235959_20180202T120000___________________MPC_O_AL_001.SEN3	ADF
attitude_scratch_file.xml orbit_scratch_file.xml	Orbit and attitude breakpoint
BP_S1_L1A_5_4_1_geo_with_outliers_NAD_1KM.nc, BP_S1_L1A_5_4_1_geo_with_outliers_NAD_A.nc, BP_S1_L1A_5_4_1_geo_with_outliers_NAD_B.nc, BP_S1_L1A_5_4_1_geo_with_outliers_NAD_F1.nc, BP_S1_L1A_5_4_1_geo_with_outliers_OBL_1KM.nc, BP_S1_L1A_5_4_1_geo_with_outliers_OBL_A.nc, BP_S1_L1A_5_4_1_geo_with_outliers_OBL_B.nc, BP_S1_L1A_5_4_1_geo_with_outliers_OBL_F1.nc	Instrument grid geolocation breakpoints
BP_S1_L1A_5_4_1_TPix_NAD_1KM.nc, BP_S1_L1A_5_4_1_TPix_NAD_A.nc, BP_S1_L1A_5_4_1_TPix_NAD_B.nc, BP_S1_L1A_5_4_1_TPix_OBL_1KM.nc,	Tie point grid geolocation and angles breakpoints

Input files in JobOrder SL1	Type
BP_S1_L1A_5_4_1_TPIx_OBL_A.nc, BP_S1_L1A_5_4_1_TPix_OBL_B.nc	
BP_S1_L1A_5_1_3_1.nc	Quasi-cartesian grid geolocation breakpoints

7.4 Validation tests lists

The following ASGARD legacy based validations scripts located in tests/validations, cover the geometric functions specified in § *7.2 Validation scope.*

Table 33 – S3 SLSTR L1 validation tests list

Processor	Validation script	Test Name
S3_SL1	tests/validations/ test_sentinel3_slstr_validation_v2.py:	
	✓ test_slstr_direct_loc_all_val	✓ [S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)
	✓ test_slstr_tp_grid_val ✓ test_slstr_tp_angles_val	✓ [S3SLSTR_TDS1_TPG_LEGACY] - Tie Point Grid (S1-L1A_5_3)
	✓ test_slstr_qc_grid_loc_val	✓ [S3SLSTR_TDS1_QCG_LEGACY] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)

7.5 Features not tested

For this version, the inverse location functions are not tested as they are based on iterative inversion of direct location.

7.6 Quality tests results

7.6.1 [S3SLSTR_TDS1_IG] - Instrument Grid (S1-L1A_5_3)

[S3SLSTR_TDS1_IG] - Instrument Grid (S1-L1A_5_3)
Summary:

[S3SLSTR_TDS1_IG] - Instrument Grid (S1-L1A_5_3)
This test aims at testing the geometric functions involved in SLSTR refactored geolocation on instrument grid.

Actors:

CS GROUP

Expected Results:

7.6.1.1 *Verify that difference between geodetic points computed and geodetic points from legacy is below threshold which correspond to specific spatial resolution of each SLSTR geometry, as defined in § 2.3.3.4* Sentinel-3 SLSTR refactored

The thresholds presented in the table below are the one for the refactored implementation for SLSTR instrument (different DEM as legacy processor).

Table 11 – Sentinel-3 SLSTR refactored metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation for grids (with geometry)	500 m 1 km	Depends on the geometry and the grid where the direct location is computed. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Planimetric distance error at iso elevation for quasi cartesian grid	16 km	Independent of any geometry. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.

	Altitude error	1 m	
Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-3 deg	
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-4 deg	

7.6.1.2 Sentinel-3 MWR legacy based

The thresholds presented in the table below are the one for the Legacy-Based implementation for MWR instrument.

Table 12 - Sentinel-3 MWR legacy based metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	15.5 m	The spatial resolution of MWR Full Resolution (FR) is 20 km. As we are in legacy-based context the results are expected to be almost the same as legacy processor.

- ✓ Sentinel-3 MWR

Pass/Fail criteria:

[S3SLSTR_TDS1_IG] - Instrument Grid (S1-L1A_5_3)			
✓ *For direct location, between the computed and legacy geodetic coordinates:* ➢ *No planimetric distance errors at iso altitude are above requirement,* ➢ *No altitude differences are above requirement.* Test Data: S3_SLSTR_TDS1 *(§7.3.1)*			
Preconditions: Reference has been generated with the Legacy processor S3_SL1			
#:	Step actions:	Expected Results:	Results:
1	Compare with legacy results of ground location and altitude for different geometry with a subset of whole coordinates array 902 x 1199 In ***tests/validations/test_sentinel3_slstr_validation_v2.py***: Run: ***test_slstr_direct_loc_all_val*** with option: *--log-cli-level WARNING*	✓ Check that planetary distance difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.4*) for 99.73% of points upon 91 x 120	✓ NAD/1KM/0 **PASSED** ✓ NAD/1KM/1 **PASSED** ✓ OBL/1KM/0 **FAILED for 0.94% of points** ✓ OBL/1KM/1 **FAILED for 0.67% of points** ✓ NAD/1KM_F1/0 **FAILED for 25.58% of points** ✓ NAD/1KM_F1/1 **FAILED for 25.51% of points** ✓ OBL/1KM_F1/0 **PASSED** ✓ OBL/1KM_F1/1 **PASSED**
		✓ Check that planetary distance difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.4*) for 99.73% of points upon 91 x 240	✓ NAD/05KM_A/0 **FAILED for 4.71% of points** ✓ NAD/05KM_A/1 **FAILED for 4.69% of points** ✓ NAD/05KM_A/2 **FAILED for 4.69% of points** ✓ NAD/05KM_A/3 **FAILED for 4.67% of points** ✓ NAD/05KM_B/0 **FAILED for 4.67% of points** ✓ NAD/05KM_B/1 **FAILED for 4.66% of points** ✓ NAD/05KM_B/2 **FAILED for 4.63% of points** ✓ NAD/05KM_B/3 **FAILED for 4.62% of points**

[S3SLSTR_TDS1_IG] - Instrument Grid (S1-L1A_5_3)			
			✓ OBL/05KM_A/0 **FAILED for 80.54% of points** ✓ OBL/05KM_A/1 **FAILED for 80.35% of points** ✓ OBL/05KM_A/2 **FAILED for 80.23% of points** ✓ OBL/05KM_A/3 **FAILED for 80.01% of points**
			✓ OBL/05KM_B/0 **FAILED for 80.54% of points** ✓ OBL/05KM_B/1 **FAILED for 80.32% of points** ✓ OBL/05KM_B/2 **FAILED for 80.19% of points** ✓ OBL/05KM_B/3 **FAILED for 80.01% of points**
		✓ Check that altitude difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.4*) for 99.73% of points upon 91 x 120 for 1KM or 91 x 240 for 0.5KM	✓ NAD/1KM/0 **PASSED** ✓ NAD/1KM/1 **PASSED**
			✓ OBL/1KM/0 **PASSED** ✓ OBL/1KM/1 **PASSED**
			✓ NAD/1KM_F1/0 **PASSED** ✓ NAD/1KM_F1/1 **PASSED**
			✓ OBL/1KM_F1/0 **PASSED** ✓ OBL/1KM_F1/1 **PASSED**
			✓ NAD/05KM_A/0 **PASSED** ✓ NAD/05KM_A/1 **PASSED** ✓ NAD/05KM_A/2 **PASSED** ✓ NAD/05KM_A/3 **PASSED**
			✓ NAD/05KM_B/0 **PASSED** ✓ NAD/05KM_B/1 **PASSED** ✓ NAD/05KM_B/2 **PASSED** ✓ NAD/05KM_B/3 **PASSED**
			✓ OBL/05KM_A/0 **PASSED**

[S3SLSTR_TDS1_IG] - Instrument Grid (S1-L1A_5_3)			
			✓ OBL/05KM_A/1 **PASSED** ✓ OBL/05KM_A/2 **PASSED** ✓ OBL/05KM_A/3 **PASSED**
			✓ OBL/05KM_B/0 **PASSED** ✓ OBL/05KM_B/1 **PASSED** ✓ OBL/05KM_B/2 **PASSED** ✓ OBL/05KM_B/3 **PASSED**

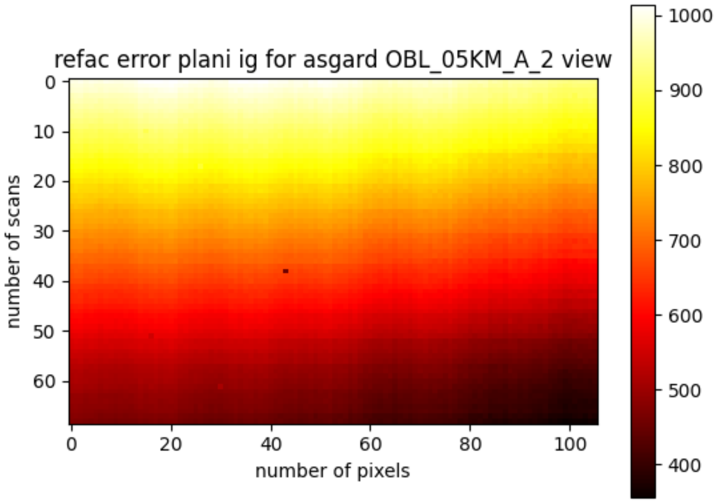
Comments:

From the last version of this validation plan, we change the subset of coordinates for direct location, instead of taking the last 10% of values we have taken a value out of ten which is more relevant.

The error is made on 05KM group, and it seems that there is a problem with location using oblique view. This has been investigated using source code ICF, but it is a version using EOCFI 4.17, while we are using EOCFI 4.23 for the tests here.

We need to take time investigating the source code ICF which corresponds to our context.

Here is an example of planimetric error for oblique view OBL/05KM_A/2 (to be compared to the legacy code which shows the same error pattern in §*6.7.1 [S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)*):

[S3SLSTR_TDS1_IG] - Instrument Grid (S1-L1A_5_3)


7.6.2 [S3SLSTR_TDS1_TPG] - Tie Point Grid (S1-L1A_5_3)

[S3SLSTR_TDS1_TPG] - Tie Point Grid (S1-L1A_5_3)
Summary:

[S3SLSTR_TDS1_TPG] - Tie Point Grid (S1-L1A_5_3)

This test aims at testing the geometric functions involved in SLSTR refactored geolocation, sun angles, incidence angles computation on tie point grid (TPix).

Actors:

CS GROUP

Expected Results:

7.6.2.1 *Verify that difference between cartesian coordinates computed and cartesian coordinates points from legacy is below threshold which correspond to specific spatial resolution of each SLSTR geometry as defined in § 2.3.3.4*Sentinel-3 SLSTR refactored

The thresholds presented in the table below are the one for the refactored implementation for SLSTR instrument (different DEM as legacy processor).

Table 11 – Sentinel-3 SLSTR refactored metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation for grids (with geometry)	500 m 1 km	Depends on the geometry and the grid where the direct location is computed. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Planimetric distance error at iso elevation for quasi cartesian grid	16 km	Independent of any geometry. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.

	Altitude error	1 m	
Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-3 deg	
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-4 deg	

7.6.2.2 Sentinel-3 MWR legacy based

The thresholds presented in the table below are the one for the Legacy-Based implementation for MWR instrument.

Table 12 - Sentinel-3 MWR legacy based metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	15.5 m	The spatial resolution of MWR Full Resolution (FR) is 20 km. As we are in legacy-based context the results are expected to be almost the same as legacy processor.

✓ Sentinel-3 MWR

7.6.2.3 *Verify that difference between sun angles/incidence angles computed and sun angles/incidence angles from legacy is below threshold which is defined by angles requirement in § 2.3.3.4* **Sentinel-3 SLSTR refactored**

The thresholds presented in the table below are the one for the refactored implementation for SLSTR instrument (different DEM as legacy processor).

Table 11 – Sentinel-3 SLSTR refactored metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation for grids (with geometry)	500 m 1 km	Depends on the geometry and the grid where the direct location is computed. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Planimetric distance error at iso elevation for quasi cartesian grid	16 km	Independent of any geometry. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Altitude error	1 m	
Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-3 deg	
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-4 deg	

[S3SLSTR_TDS1_TPG] - Tie Point Grid (S1-L1A_5_3)

7.6.2.4 Sentinel-3 MWR legacy based

The thresholds presented in the table below are the one for the Legacy-Based implementation for MWR instrument.

Table 12 - Sentinel-3 MWR legacy based metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	15.5 m	The spatial resolution of MWR Full Resolution (FR) is 20 km. As we are in legacy-based context the results are expected to be almost the same as legacy processor.

✓ Sentinel-3 MWR

Pass/Fail criteria:

- ✓ *For direct location, between the computed and legacy geodetic coordinates:*
  - ➢ *No planimetric distance errors at iso altitude are above requirement,*
  - ➢ *No altitude differences are above requirement.*
- ✓ *For Sun angles (azimuth phi, zenith theta) between computed and legacy:*
  - ➢ *No direction errors are above requirement.*
- ✓ *For incidence angles (azimuth phi, zenith theta)) between computed and legacy*
  - ➢ *No direction errors are above requirement.*

[S3SLSTR_TDS1_TPG] - Tie Point Grid (S1-L1A_5_3)			
Test Data: S3_SLSTR_TDS1 (§7.3.1)			
Preconditions: Reference has been generated with the Legacy processor S3_SL1			
#:	Step actions:	Expected Results:	Results:
1	Compare with legacy results of ground location and altitude for different geometry, with a subset of whole coordinates array of tie point grid of size 115 x 101 for nadir view and 141 x 40 for oblique view In ***tests/validations/test_sentinel3_slstr_validation_v2.py*** Run: ***test_slstr_tp_grid_val*** with option: *--log-cli-level WARNING*	✓ Check that distance (norm computation) difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.4*) for 99.73% of points upon 23 x 21	✓ NAD/1KM/0 **FAILED for 6.00% of points** ✓ NAD/1KM/1 **FAILED for 6.42% of points** ✓ OBL/1KM/0 **FAILED for 46.55% of points** ✓ OBL/1KM/1 **FAILED for 47.41% of points**
		✓ Check that distance (norm computation) difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.4*) for 99.73% of points upon 23 x 21	✓ NAD/05KM_A/0 **FAILED for 20.29% of points** ✓ NAD/05KM_A/1 **FAILED for 19.05% of points** ✓ NAD/05KM_A/2 **FAILED for 19.67% of points** ✓ NAD/05KM_A/3 **FAILED for 19.88% of points** ✓ NAD/05KM_B/0 **FAILED for 20.08% of points** ✓ NAD/05KM_B/1 **FAILED for 19.05% of points** ✓ NAD/05KM_B/2 **FAILED for 19.67% of points** ✓ NAD/05KM_B/3 **FAILED for 19.88% of points**

[S3SLSTR_TDS1_TPG] - Tie Point Grid (S1-L1A_5_3)			
			✓ OBL/05KM_A/0 **FAILED for 99.14% of points** ✓ OBL/05KM_A/1 **FAILED for 99.14% of points** ✓ OBL/05KM_A/2 **FAILED for 99.14% of points** ✓ OBL/05KM_A/3 **FAILED for 99.14% of points**
			✓ OBL/05KM_B/0 **FAILED for 99.14% of points** ✓ OBL/05KM_B/1 **FAILED for 99.14% of points** ✓ OBL/05KM_B/2 **FAILED for 99.14% of points** ✓ OBL/05KM_B/3 **FAILED for 99.14% of points**
Comments: The reason of failing tests for direct loc on tie point grid needs to be investigated. The direct_loc results for each tie point is expressed in longitude and latitude and the legacy results of geolocation on tie point grid is expressed as (X, Y) coordinates on quasi cartesian grid (and not expressed in longitude and latitude). In order to validate properly direct location expressed in ground coordinates, we built this quasi-cartesian grid in ASGARD and computed (X, Y) coordinates on this grid from the longitude/latitude results but for now: ➔ 94% of value errors is below tolerance for NAD/1KM, ➔ 80% for NAD/05KM ➔ and-less than 1% for all oblique geometry. Hence more investigation needs to be done for oblique view using the new version of the ICF code which is using EOCFI 4.23.			
2	Compare with legacy results of sun angles/incidence angles for different geometry, with a subset of whole coordinates array of tie point grid of size 115 x 101 for nadir view and 141 x 40 for oblique view	✓ Check that no sun angles difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.4*) for 99.73% of points upon 23 x 21 for nadir view and upon 29 x 8 for oblique view	✓ NAD/1KM/0 **FAILED for 100% of points** ✓ NAD/1KM/1 **FAILED for 100% of points**
			✓ OBL/1KM/0 **FAILED for 100% of points** ✓ OBL/1KM/1 **FAILED for 100% of points**
			✓ NAD/05KM_A/0 **FAILED for 100% of points** ✓ NAD/05KM_A/1 **FAILED for 100% of points** ✓ NAD/05KM_A/2 **FAILED for 100% of points**

[S3SLSTR_TDS1_TPG] - Tie Point Grid (S1-L1A_5_3)			
	In ***tests/validations/test_sentinel3_slstr_validation_v2.py*** Run: ***test_slstr_tp_angles_val*** with option: *--log-cli-level WARNING*	Sun angles is defined as: solar_phi, solar_theta	✓ NAD/05KM_A/3 **FAILED for 100% of points**
			✓ NAD/05KM_B/0 **FAILED for 100% of points** ✓ NAD/05KM_B/1 **FAILED for 100% of points** ✓ NAD/05KM_B/2 **FAILED for 100% of points** ✓ NAD/05KM_B/3 **FAILED for 100% of points**
			✓ OBL/05KM_A/0 **FAILED for 100% of points** ✓ OBL/05KM_A/1 **FAILED for 100% of points** ✓ OBL/05KM_A/2 **FAILED for 100% of points** ✓ OBL/05KM_A/3 **FAILED for 100% of points**
			✓ OBL/05KM_B/0 **FAILED for 100% of points** ✓ OBL/05KM_B/1 **FAILED for 100% of points** ✓ OBL/05KM_B/2 **FAILED for 100% of points** ✓ OBL/05KM_B/3 **FAILED for 100% of points**
		✓ Check that no incidence angles difference between computed and legacy is below tolerance as expressed in SLSTR metrics (*§ 2.3.3.4*) for 99.73% of points upon 23 x 21 for nadir view and upon 29 x 8 for oblique view Incidence angle is defined as: viewing_phi, viewing_theta	✓ NAD/1KM/0 **FAILED for 100% of points** ✓ NAD/1KM/1 **FAILED for 100% of points**
			✓ OBL/1KM/0 **FAILED for 100% of points** ✓ OBL/1KM/1 **FAILED for 100% of points**
			✓ NAD/05KM_A/0 **FAILED for 100% of points** ✓ NAD/05KM_A/1 **FAILED for 100% of points** ✓ NAD/05KM_A/2 **FAILED for 100% of points** ✓ NAD/05KM_A/3 **FAILED for 100% of points**
			✓ NAD/05KM_B/0 **FAILED for 100% of points** ✓ NAD/05KM_B/1 **FAILED for 100% of points** ✓ NAD/05KM_B/2 **FAILED for 100% of points** ✓ NAD/05KM_B/3 **FAILED for 100% of points**

[S3SLSTR_TDS1_TPG] - Tie Point Grid (S1-L1A_5_3)			
			✓ OBL/05KM_A/0 **FAILED for 100% of points** ✓ OBL/05KM_A/1 **FAILED for 100% of points** ✓ OBL/05KM_A/2 **FAILED for 100% of points** ✓ OBL/05KM_A/3 **FAILED for 100% of points**
			✓ OBL/05KM_B/0 **FAILED for 100% of points** ✓ OBL/05KM_B/1 **FAILED for 100% of points** ✓ OBL/05KM_B/2 **FAILED for 100% of points** ✓ OBL/05KM_B/3 **FAILED for 100% of points**

Comments:

The error on sun/incidence angle computation is around 10^{-3} ° so it is just above tolerance, this is due to the propagation error made on direct location.

Indeed, the sun/incidence angles computation takes in input geodetic coordinates, as we do not have the legacy geodetic coordinates, we could not only check the error on sun/incidence angles computation, but we must use the geodetic coordinates computed with direct location function which already have an error on it. When the direct location will be fixed the sun/incidence angle computation would pass because it involves the same code as OLCI sun/incidence angles computation.

7.6.3 [S3SLSTR_TDS1_QCG] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)

[S3SLSTR_TDS1_QCG] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)
Summary: *This test aims at testing the geometric functions involved in SLSTR refactored geolocation on quasi cartesian grid (QC).*

[S3SLSTR_TDS1_QCG] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)

Actors:

CS GROUP

Expected Results:

7.6.3.1 *Verify that difference between cartesian coordinates computed and cartesian coordinates from legacy is below threshold which correspond to specific spatial resolution as defined in §2.3.3.4* Sentinel-3 SLSTR refactored

The thresholds presented in the table below are the one for the refactored implementation for SLSTR instrument (different DEM as legacy processor).

Table 11 – Sentinel-3 SLSTR refactored metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation for grids (with geometry)	500 m 1 km	Depends on the geometry and the grid where the direct location is computed. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Planimetric distance error at iso elevation for quasi cartesian grid	16 km	Independent of any geometry. Notice that for all the geometries concerned the altitude threshold is fixed at 1m.
	Altitude error	1 m	

Incidence angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-3 deg	
Sun angles	Error between two pointing directions Each direction expressed by (phi, theta)	10^-4 deg	

7.6.3.2 Sentinel-3 MWR legacy based

The thresholds presented in the table below are the one for the Legacy-Based implementation for MWR instrument.

Table 12 - Sentinel-3 MWR legacy based metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	15.5 m	The spatial resolution of MWR Full Resolution (FR) is 20 km. As we are in legacy-based context the results are expected to be almost the same as legacy processor.

- ✓ Sentinel-3 MWR, *notice that this quasi-cartesian grid is independent of the geometry.*

Pass/Fail criteria:

- ✓ *For direct location, between the computed and legacy geodetic coordinates:*
  - ➢ *No planimetric distance errors at iso altitude are above requirement,*
  - ➢ *No altitude differences are above requirement.*

[S3SLSTR_TDS1_QCG] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)			
Test Data: S3_SLSTR_TDS1 *(§ 7.3.1)*			
Preconditions: Reference has been generated with the Legacy processor S3_SL1			
#:	Step actions:	Expected Results:	Results:
1	Compare with legacy results of ground location for Nadir 1km grid first detector (NAD/1KM/0) with whole coordinates array, 115 x 130 In ***tests/validations/test_sentinel3_slstr_validation_v2.py*** Run: ***test_slstr_qc_grid_loc_val*** with option: *--log-cli-level WARNING*	✓ Check that planetary distance difference between computed and legacy is below tolerance (*§ 2.3.3.4)* for 99.73% of points upon 115 x 130	✓ **FAILED for 100% of points**
Comment: For the same reason as above the error must be investigated.			

8. SENTINEL 3 MWR LEGACY BASED

8.1 Legacy processors

Table below presents all processors that must be supported by ASGARD because geometric functionalities are involved in.

Table 34 –Sentinel-3 legacy function link

Mission	Level	Instrument	Processor	Processor Version	IPF Processing Baseline3	Comments
S3	L1	MWR	S3_MW1	6.15	3.21	

8.2 ASGARD implementations using EOCFI

ASGARD comes with an implementation of several products and models relying on EOCFI for MWR S3 product (legacy-based implementation under **asgard-legacy** project).

It will become deprecated once Rugged/Orekit based implementation will support those sensors (Refactored implementation, see *Sentinel 3 MWR refactored* section).

In order to compare results from legacy processor based on EOCFI and ASGARD results, the legacy processors have been launched with a JobOrder containing all necessary files (ADF, L0 product, orbit file, attitude file) in input to compute the level 1 product of each sensor. Then we manage to load and use the same inputs in ASGARD to compute direct location, sun angles, incidence angles and compare those results with associated Breakpoints generated by the legacy processor. The comparison is then based and acceptance thresholds which are detailed in § *2.3.3.5.*

Only the Legacy-based implementation is tested here.

8.3 Validation scope

8.4 Validation tests datasets

The sensing time of processor legacy is defined at the beginning of the JobOrder:

- ✓ Start: 2022-11-01T06.02.06.000000
- ✓ Stop: 2022-11-01T07.43.39.000000

Table 35 - Input dataset S3_MWR

Files name	Type
S3A_MW_0_MWR____20221101T060206_20221101T074339_20221101T080003_6092_091_305______PS1_O_NR_004.SEN3	L0 product
S3A_TM_0_NAT____20221101T041952_20221101T060206_20221101T061759_6133_091_304______PS1_O_AL_004.SEN3	Navatt
S3A_TM_0_NAT____20221101T060207_20221101T074339_20221101T075933_6092_091_305______PS1_O_AL_004.SEN3	Navatt
S3A_TM_0_NAT____20221101T074340_20221101T092440_20221101T094037_6059_091_306______PS1_O_AL_004.SEN3	Navatt
S3A_AX___OSF_AX_20160216T192404_99991231T235959_20220330T090651___________________EUM_O_AL_001.SEN3	Orbit
S3A_AX___FRO_AX_20221030T000000_20221109T000000_20221102T065450___________________EUM_O_AL_001.SEN3	ADF
S3A_MW_1_SLC_AX_20160216T000000_20991231T235959_20190621T120000___________________MPC_O_AL_003.SEN3	ADF
S3A_MW___CHDNAX_20160216T000000_20991231T235959_20210929T120000___________________MPC_O_AL_005.SEN3	ADF
S3A_MW___CHDRAX_20160216T000000_20991231T235959_20170908T120000___________________MPC_O_AL_004.SEN3	ADF
S3A_MW___STD_AX_20160216T000000_20991231T235959_20220111T120000___________________MPC_O_AL_003.SEN3	ADF
3A_MW_1_NIR_AX_20000101T000000_20230417T020641_20230427T023158___________________LN3_R_AL____.SEN3	ADF
S3A_MW_1_DNB_AX_20000101T000000_20230417T020641_20230427T023158___________________LN3_R_AL____.SEN3	ADF
measurement.nc	Time and geolocation breakpoints

8.5 Validation tests list

Those legacy steps referring to DPM sections are covered by ASGARD legacy based validations scripts located in tests/validations.

Table 36 - S3 MWR validation test list

Processors	Validation script	Test Name
S3_MW1	tests/validations/test_sentinel3_mwr_validation.py: ✓ test_direct_loc_mwr_val	[S3MWR_TDS_LEGACY] - Geo referencing

8.6 Features not tested

For this version, the inverse location functions are not tested as they are based on iterative inversion of direct location.

8.7 Quality tests results

8.7.1 [S3MWR_TDS_LEGACY] - Geo referencing

[S3OLCI_TDS1_EO_LEGACY] - Geo referencing

Summary:

This test aims at testing the geometric functions involved in MWR legacy based geo-referencing.

Actors:

CS GROUP

Expected Results:

- ✓ Verify *that difference between geodetic points computed and geodetic points from legacy is below threshold as* defined *§ 2.3.3.5 Sentinel-3 MWR legacy based*

The thresholds presented in the table below are the one for the Legacy-Based implementation for MWR instrument.

Table 12 - Sentinel-3 MWR legacy based metrics and thresholds

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	15.5 m	The spatial resolution of MWR Full Resolution (FR) is 20 km. As we are in legacy-based context the results are expected to be almost the same as legacy processor.

- ✓ Sentinel-3 MWR

[S3OLCI_TDS1_EO_LEGACY] - Geo referencing			
Pass/Fail criteria: ✓ *For direct location, between the computed and legacy geodetic coordinates:* ➢ *No planimetric distance errors at iso altitude are above requirement,* ➢ *No altitude differences are above requirement.* Test Data: *S3_MWR_TDS (§ 8.4)*			
Preconditions: Reference has been generated with the Legacy processor S3_MW1			
#:	Step actions:	Expected Results:	Results:
1	In ***tests/validations/test_sentinel3_mwr_validation.py*** run: ***test_direct_loc_mwr_val***	✓ Check that planimetric error at iso altitude between computed and legacy is below tolerance as expressed in MWR metrics (*§ 2.3.3.5 Sentinel-3 MWR legacy based* The thresholds presented in the table below are the one for the Legacy-Based implementation for MWR instrument.	For camera 1: ✓ **PASSED** For camera 2: ✓ **PASSED**

[S3OLCI_TDS1_EO_LEGACY] - Geo referencing			
		Table 12 - Sentinel-3 MWR legacy based metrics and thresholds	

Type	Metric	Requirement	Comment
Ground location	Planimetric distance error at iso elevation	15.5 m	The spatial resolution of MWR Full Resolution (FR) is 20 km. As we are in legacy-based context the results are expected to be almost the same as legacy processor.

✓ Sentinel-3 MWR) for 99.73% of points

9. SENTINEL 3 MWR REFACTORED

9.1 ASGARD implementations refactored

As seen in *§ 8 Sentinel 3 MWR legacy based*, the MWR code implementation was relying on models and products on EOCFI.

This was a first step in the development of ASGARD before the refactoring of the code.

The refactored implementation of ASGARD is based on Rugged/Orekit libraries through the ***pyrugged*** project (refactored implementation under **asgard** project).

In order to compare results from legacy processor based on EOCFI and ASGARD results, the legacy processors have been launched with a JobOrder containing all necessary files (ADF, L0 product, orbit file, attitude file) in input to compute the level 1 product of each sensor. Then we manage to load and use the same inputs in ASGARD to compute direct location and compare those results with associated outputs generated by the legacy processor. The comparison is then based and acceptance thresholds which are detailed in *§ 2.3.3.6 Sentinel-3 MWR refactored*.

9.2 Validation scope

9.3 Validation tests datasets

The sensing time of processor legacy is defined at the beginning of the JobOrder:

- ✓ Start: 2022-11-01T06.02.06.000000
- ✓ Stop: 2022-11-01T07.43.39.000000

Table 37 - Input dataset S3_MWR

Files name	Type
S3A_MW_0_MWR____20221101T060206_20221101T074339_20221101T080003_6092_091_305______PS1_O_NR_004.SEN3	L0 product
S3A_TM_0_NAT____20221101T041952_20221101T060206_20221101T061759_6133_091_304______PS1_O_AL_004.SEN3	Navatt
S3A_TM_0_NAT____20221101T060207_20221101T074339_20221101T075933_6092_091_305______PS1_O_AL_004.SEN3	Navatt
S3A_TM_0_NAT____20221101T074340_20221101T092440_20221101T094037_6059_091_306______PS1_O_AL_004.SEN3	Navatt

S3A_AX___OSF_AX_20160216T192404_99991231T235959_20220330T090651___________________EUM_O_AL_001.SEN3	Orbit
S3A_AX___FRO_AX_20221030T000000_20221109T000000_20221102T065450___________________EUM_O_AL_001.SEN3	ADF
S3A_MW_1_SLC_AX_20160216T000000_20991231T235959_20190621T120000___________________MPC_O_AL_003.SEN3	ADF
S3A_MW___CHDNAX_20160216T000000_20991231T235959_20210929T120000___________________MPC_O_AL_005.SEN3	ADF
S3A_MW___CHDRAX_20160216T000000_20991231T235959_20170908T120000___________________MPC_O_AL_004.SEN3	ADF
S3A_MW___STD_AX_20160216T000000_20991231T235959_20220111T120000___________________MPC_O_AL_003.SEN3	ADF
3A_MW_1_NIR_AX_20000101T000000_20230417T020641_20230427T023158___________________LN3_R_AL____.SEN3	ADF
S3A_MW_1_DNB_AX_20000101T000000_20230417T020641_20230427T023158___________________LN3_R_AL____.SEN3	ADF
measurement.nc	Time and geolocation breakpoints

9.4 Validation tests list

Those legacy steps referring to DPM sections are covered by ASGARD refactored validations scripts located in tests/validations.

Table 38 - S3 MWR validation test list

Processors	Validation script	Test Name
S3_MW1	tests/validations/test_sentinel3_mwr_validation.py: ✓ test_direct_loc_mwr_val	[S3MWR_TDS] - Geo referencing

9.5 Features not tested

For this version, the inverse location functions are not tested as they are based on iterative inversion of direct location.

9.6 Quality tests results

9.6.1 [S3MWR_TDS] - Geo referencing

[S3OLCI_TDS1_EO_LEGACY] - Geo referencing
Summary: *This test aims at testing the geometric functions involved in MWR refactored geo-referencing.* Actors: *CS GROUP* Expected Results: ✓ Verify *that difference between geodetic points computed and geodetic points from legacy is below threshold as* defined in *§ 2.3.3.6 Sentinel-3 MWR refactored* Pass/Fail criteria: ✓ *For direct location, between the computed and legacy geodetic coordinates:* ➢ *No planimetric distance errors at iso altitude are above requirement,* ➢ *No altitude differences are above requirement.* Test Data: *S3_MWR_TDS (§ 9.4)*
Preconditions:

[S3OLCI_TDS1_EO_LEGACY] - Geo referencing			
Reference has been generated with the Legacy processor S3_MW1			
#:	Step actions:	Expected Results:	Results:
1	In ***tests/validations/test_sentinel3_mwr_validation.py*** run: ***test_direct_loc_mwr_val***	✓ Check that planimetric error at iso altitude between computed and legacy is below tolerance as expressed in MWR metrics (*§ 2.3.3.6 Sentinel-3 MWR refactored*) for 99.73% of points	For camera 1: ✓ **PASSED** For camera 2: ✓ **PASSED**

10. SENTINEL 1 SAR

10.1 Legacy processor

We will be using the S1_L12 processor 3.7.1 to generate the reference outputs for validation.

10.2 Validation scope

The S1 validation will focus on the following computations:

- ✓ Direct location on ground
- ✓ Incidence angles
- ✓ Viewing angles
- ✓ Roll/pitch/yaw angles
- ✓ Terrain height estimation

The following outputs can't be validated using legacy processor, as they are not exposed:

- ✓ Projected ground velocity (appear in section 9.10 of AD-ATBD-S1)

The inverse location is not tested because the generic inversion algorithm may not be compatible with the presence of burst in EW and IW products.

10.3 Validation methodology

The methodology will be rather straightforward: we will use the output L1 products generated by the legacy processor and compare some of its output to data generated by ASGARD. We will cover different types of acquisition modes (SM, IW, EW, WV).

We use the full geolocation grid supplied in SLC products. From this grid, we setup tests to check:

- ✓ The direct location on ground at constant altitude. We use the line, sample and altitude given in the geolocation grid, and we compare the output geodetic points.
- ✓ We also check the acquisition time returned by direct_loc() against the azimuth time.
- ✓ The slant range localisation method: we give as input the range, azimuth and elevation from the geolocation grid and compare the output geodetic points.
- ✓ The incidence angles
- ✓ The viewing angles
- ✓ The elevation difference between the GETASSE DEM and the geolocation grid

We use the terrain height table given in SLC products to check the results of terrain height estimation.

We use the azimuth time and roll angles found in the antenna pattern table to check the results of roll/pitch/yaw angles. Note that only the roll angle can be checked.

Here is the detailed list of validation tests for both S1SARProduct implementations (pyRugged and EOCFI flavors).

Validation script	Implementation tested	Validation test name	Python test name
asgard/ tests/ validations/ test_sentinel1_sar_validation.py	PyRugged based	[S1SAR_GEO]	test_direct_location
		[S1SAR_RANGE]	test_slant_range_localisation
		[S1SAR_INCI]	test_incidence_angles
		[S1SAR_VIEW]	test_viewing_angles
		[S1SAR_HEIGHT]	test_terrain_height
		[S1SAR_ZD]	test_zero_doppler_to_attitude
asgard-legacy/ tests/ validations/ test_sentinel1_sar_validation.py	EOCFI based	[S1_EE_GEO]	test_direct_location
		[S1_EE_RANGE]	test_slant_range_localisation
		[S1_EE_INCI]	test_incidence_angles
		[S1_EE_VIEW]	test_viewing_angles
		[S1_EE_HEIGHT]	test_terrain_height
		[S1_EE_ZD]	test_zero_doppler_to_attitude

10.4 Validation test datasets

The following test datasets have been used.

Dataset	Product type	Product name (SLC)
S1_TDS1	EW	S1A_EW_SLC__1SDH_20221111T114659_20221111T114758_045846_057C1E_CCDA.SAFE
S1_TDS2	IW	S1A_IW_SLC__1SDV_20230110T220000_20230110T220027_046727_059A05_62E1.SAFE
S1_TDS3	S4	S1A_S4_SLC__1SDH_20220801T234916_20220801T234940_044366_054B70_88A1.SAFE
S1_TDS4	WM	S1A_WV_SLC__1SSV_20220801T200456_20220801T203530_044363_054B5E_510C.SAFE

10.5 Quality tests results

Here are the results obtained (in term of tolerance threshold) the different test cases.

10.5.1 S1SAR_GEO and S1_EE_GEO

10.5.1.1 Planar difference

We give the tolerance threshold (in meters) such that all planar differences obtained through the entire geolocation grid (for each swath) are inferior.

Test dataset	S1SAR_GEO	S1_EE_GEO
TDS1	0.1	0.1
TDS2	0.1	0.1
TDS3	0.1	0.1
TDS4	0.6	0.7

10.5.1.2 Azimuth time difference

We give the tolerance threshold (in seconds) such that all azimuth time differences obtained through the entire geolocation grid (for each swath) are inferior.

Test dataset	S1SAR_GEO	S1_EE_GEO
TDS1	10^{-4}	10^{-4}
TDS2	10^{-4}	10^{-4}
TDS3	10^{-4}	10^{-4}
TDS4	10^{-4}	10^{-4}

10.5.2 S1SAR_RANGE and S1_EE_RANGE

We show the thresholds (in meters) so that all planar differences obtained on the entire geolocation grid (for all swaths) are inferior.

Test dataset	S1SAR_GEO	S1_EE_GEO
TDS1	0.02	0.02
TDS2	0.02	0.02
TDS3	0.02	0.02

TDS4	0.02	0.02

10.5.3 S1SAR_INCI and S1_EE_INCI

We show the thresholds (in degrees) so that the differences obtained on the incidence angle through the entire geolocation grid (for all swaths) are inferior.

Test dataset	S1SAR_GEO	S1_EE_GEO
TDS1	10^{-4}	10^{-4}
TDS2	10^{-4}	10^{-4}
TDS3	10^{-4}	10^{-4}
TDS4	10^{-4}	10^{-4}

10.5.4 S1SAR_VIEW and S1_EE_VIEW

We show the thresholds (in degrees) so that the differences obtained on the viewing angle through the entire geolocation grid (for all swaths) are inferior.

Test dataset	S1SAR_GEO	S1_EE_GEO
TDS1	10^{-4}	10^{-4}
TDS2	10^{-4}	10^{-4}
TDS3	10^{-4}	10^{-4}
TDS4	10^{-4}	10^{-4}

10.5.5 S1SAR_HEIGHT and S1_EE_HEIGHT

We show the mean and standard deviation of the elevation error (in meters) between the terrain height table from SLC product and the one computed in ASGARD. These results are not conclusive, we were not able to retrieve the same parameters to obtain similar results.

Test dataset	S1SAR_GEO	S1_EE_GEO
TDS1	EW3 mean=-7.31 , std=72.34 EW1 mean=-21.14 , std=39.23 EW4 mean=-2.10 , std=125.28 EW5 mean=-11.43 , std=66.74 EW2 mean=-10.30 , std=84.02	EW3 mean=-4.71 , std=72.52 EW1 mean=-21.70 , std=38.76 EW4 mean=-1.14 , std=124.26 EW5 mean=-10.30 , std=66.36 EW2 mean=-6.77 , std=80.18
TDS2	IW2 mean=12.76 , std=3.99 IW1 mean=10.69 , std=3.97 IW3 mean=13.75 , std=2.41	IW2 mean=12.77 , std=3.99 IW1 mean=10.70 , std=3.97 IW3 mean=13.75 , std=2.42

TDS3	S4 mean=-23.07 , std=9.98	S4 mean=-23.26 , std=10.42
TDS4	WV2/074 mean=-0.67 , std=0.47 WV1/079 mean=-7.88 , std=0.17	WV2/074 mean=-0.67 , std=0.47 WV1/079 mean=-7.88 , std=0.16

10.5.6 S1SAR_ZD and S1_EE_ZD

We show the thresholds (in degrees) so that the differences obtained on the roll angle through the entire antenna pattern (for all swaths) are inferior.

Test dataset	S1SAR_GEO	S1_EE_GEO
TDS1	$2x10^{-3}$	$2x10^{-3}$
TDS2	$2x10^{-3}$	$2x10^{-3}$
TDS3	$2x10^{-3}$	$2x10^{-3}$
TDS4	$2x10^{-3}$	$2x10^{-3}$

10.6 Conclusions for S1

Overall, the two implementations (pyRugged and EOCFI based) have similar performance.

The direct location routine has an acceptable accuracy of the order 0.1m. Given that the slant range location routine (using directly the azimuth time as input) has better accuracy (2cm), we conclude that the azimuth time estimation is degrading the results of direct location.

The incidence and viewing angles have good accuracy (10^{-4} °), they mostly depend on orbit position estimates. This angular error corresponds to a satellite position error of ~1.2m.

The computation of terrain height as described in section 4.4 of AD-ATBD-S1 is far from reference. We need to explore several cases:

- ✓ The input DEM is different (unlikely, we use the GETASSE).
- ✓ The terrain height table in the SLC annotations doesn't really correspond to the output of the terrain height function. The sampling in azimuth is different anyway.
- ✓ The estimation of sampling points (longitude/latitude) is not correct. However, the planar accuracy obtained in direct location seems to good enough to avoid any impact on terrain height function.
- ✓ The parameters used in this function are different. This is the most likely explanation.

The computation of roll/pitch/yaw angles has good accuracy, but only the roll angle had reference values.

The computation of projected ground velocity was not validated due to missing reference data.

11. LOW-LEVEL FUNCTIONS

11.1 EOCFI versus Orekit

Some low-level functions are similar in EOCFI and Orekit libraries.

As these low-level functions are used everywhere, it is necessary to be sure that they give the same results to avoid errors propagation among calling functions.

11.2 Validation scope

The following low-level components are used at each step of the computations: time conversions, Earth related coordinates and frames conversions, Sun position and angles.

- ✓ `asgard.models.time.TimeReference`: validated against ExplorerTimeReference
  - ➢ Timescale conversion: TAI, UTC, UT1 and GPS
  - ➢ Leap second query
  - ➢ ASCII to processing / Processing to ASCII
  - ➢ CUC to processing
  - ➢ Transport to processing
- ✓ `asgard.models.body.EarthBody`: validated against ExplorerEarthBody
  - ➢ Cartesian to geodetic
  - ➢ Geodetic to cartesian
  - ➢ Reference frame change (EF, EME2000, …)
  - ➢ Geodetic distance
  - ➢ Sun position

11.3 Validation test datasets

The **default epoch** used in the project is ***2000-01-01 00:00:00***.

The **Julian Dates** (named **JD** here, not to be confused with “real” Julian Dates) are expressed with respect to this epoch.

11.3.1 Time Reference dataset

Table 34 – TimeReference dataset

#	Test case	Data	Specific input files	Comments
1	Timescale conversion	Same date expressed in different time scale: ✓ TAI = 8360.890787037037 JD ✓ UTC = 8360.890358796296 JD	For ASGARD: ✓ sxgeo/src/main/resources/ orekit-data/ UTC-TAI.history (from SXGEO wheel)	

		✓ UT1 = 8360.890358796296 JD ✓ GPS = 8360.890567129629 JD	For legacy: ✓ tests/resources/ 207_BULLETIN_B207.txt	
2	Leap second query	Dates: ✓ Standard test between ➢ Start 4745.0 JD ➢ End 6570.0 JD (dates expressed in processing format) ✓ Limit test case around leap: ➢ 2015-06-30T23:59:59 UTC ➢ 2015-07-01T00:00:00 UTC	For ASGARD: ✓ sxgeo/src/main/resources/ orekit-data/ UTC-TAI.history (from SXGEO wheel) For legacy: ✓ tests/resources/ 207_BULLETIN_B207.txt ✓ tests/resources/ bulletinb-348.txt	
3	ASCII to processing / Processing to ASCII	Date expressed in processing format in TAI scale to ASCII formats: ✓ TAI = 8360.890787037037 JD Dates expressed in ASCII formats to processing format in TAI: ✓ STD format = 2022-11-21_21:22:44 ✓ CCSDSA MICROSEC format = 2022-11-21T21:22:44.000000 Date expressed in processing format in UTC scale to ASCII STD format: ✓ UTC = 8338 JD Date with real offset to STD format ✓ Real offset of 2022-10-30_00:00:30 UTC From processing in different scale to STD format, where offset not detected: ✓ TAI = 8338.00034722 JD ✓ UTC = 8338.00034722 JD Conversion on the fly from processing to STD format, from TAI to UTC scale ✓ TAI = 8338.00042824 JD	For ASGARD: ✓ sxgeo/src/main/resources/ orekit-data/ UTC-TAI.history (from SXGEO wheel) For legacy: ✓ tests/resources/ 207_BULLETIN_B207.txt	

		✓ TAI = 8338.00077546 JD Dates expressed in STD format around leap seconds: ✓ 2016-12-31_23:59:59 ✓ 2016-12-31_23:59:60 ✓ 2017-01-01_00:00:00 Date expressed in processing format in UTC scale to ASCII STD format around leap second not detected: ✓ UTC = 6210.0 JD		
4	CUC to processing	Dates in GPS time (CUC format): ✓ [1336436015, 0] ✓ [1336436015, 1000] ✓ [1336436015, 1000000] ✓ [1336436015, 10000000] ✓ [1336522415, 0]	For ASGARD: ✓ sxgeo/src/main/resources/ orekit-data/ UTC-TAI.history (from SXGEO wheel) For legacy: ✓ tests/resources/ 207_BULLETIN_B207.txt	
5	Transport to processing	Dates in transport format (days / seconds / microsec): ✓ [8320, 2122, 71836] ✓ [8320, 2122, 115836] ✓ [8320, 2122, 159835] ✓ [8320, 2122, 203835]	For ASGARD: ✓ sxgeo/src/main/resources/ orekit-data/ UTC-TAI.history (from SXGEO wheel) For legacy: ✓ tests/resources/ 207_BULLETIN_B207.txt	

11.3.2 Earth Body dataset

Table 39 – EarthBody dataset

#	Test case	Data	Specific input files	Comments
1	Cartesian to geodetic / Geodetic to cartesian	Geodetic position (°, °, m): ✓ [-40.0 for ASGARD / 320 for legacy, 25.0, 62.0], ✓ [45.0, -12.0, 45.0], ✓ [175.0, 45.0, 12.0], ✓ [64.0, -86.0, 150.0]	None	
2	Geodetic distance	First points longitude, latitude (°, °): ✓ [10.0, 0.0],	None	

		✓ [10.0, 50.0], ✓ [10.0, 50.0], ✓ [10.0, 45.0], ✓ [10.0, 30.0], ✓ [4.17, 10.0] Second points longitude, latitude (°, °): ✓ [20.0, 0.0], ✓ [20.0, 50.0], ✓ [40.0, 50.0], ✓ [20.0, 55.0], ✓ [10.0, 40.0], ✓ [4.1705, 10.001] All points at same altitude = 30 m		
3	Reference frame change EF -> EME2000	Time TAI (JD): ✓ 8338.06258101852, ✓ 8338.06292824074, ✓ 8338.063275462962, ✓ 8338.063622685186, ✓ 8338.063969907407, Position in EF (m): ✓ [4221826.162, -5719877.893, 1024795.802] ✓ [4161165.040, -5720733.673, 1242927.811] ✓ [4096499.892, -5715798.296, 1459852.080] ✓ [4027918.272, -5705058.933, 1675357.777] ✓ [3955511.592, -5688508.798, 1889235.445] Velocity in EF (m/s) ✓ [-1954.363827, -0124.852632, 7288.843309] ✓ [-2089.246655, 67.905300, 7252.112699] ✓ [-2221.276716, 261.191149, 7208.333772] ✓ [-2350.323386, 454.804057, 7157.548875] ✓ [-2476.259870, 648.542280, 7099.807314]	None	

4	Sun position	Time GPS (JD) ✓ 8338.06423611 ✓ 8338.06523611 ✓ 8338.06623611 ✓ 8338.06723611 ✓ 8338.06823611 ✓ 8338.06923611 ✓ 8338.07023611 ✓ 8338.07123611 ✓ 8338.07223611 ✓ 8338.07323611	For ASGARD: ✓ tests/resources/ 207_BULLETIN_B207.txt For legacy: ✓ tests/resources/ S3/FRO/S3A_OPER_MPL_ORBRES_20221030T000000_20221109T000000_0001.EOF	

11.4 Validation tests lists

11.4.1 Time reference tests list

Table 40 - Time Reference tests list

ASGARD Validation script (under asgard project)	Legacy validation script (under asgard-legacy project)	Test Name
tests/test_models_time.py: ✓ test_ time_reference_convert	tests/tests_models_explorer.py: ✓ test_time_reference_convert	[LLF_TSC] – Time Scale Conversion
tests/test_models_time.py: ✓ test_time_reference_leap_second	tests/tests_models_explorer.py: ✓ test_time_reference_leap_second	[LLF_LSQ] - Leap Second Query
tests/test_models_time.py: ✓ test_time_reference_ascii ✓ test_time_reference_ascii_leap	tests/tests_models_explorer.py: ✓ test_time_reference_ascii ✓ test_time_reference_ascii_leap	[LLF_ATOP_PTOA] - ASCII To Processing / Processing To ASCII
tests/test_models_time.py: ✓ test_time_reference_cuc	tests/tests_models_explorer.py: ✓ test_time_reference_cuc	[LLF_CUCTOP] - CUC to processing
tests/test_models_time.py: ✓ test_time_reference_transport	tests/tests_models_explorer.py: ✓ test_time_reference_transport	[LLF_TTOP] - Transport to processing

11.4.2 Earth Body tests list

Table 41 - Earth Body tests list

ASGARD Validation script (under asgard project)	Legacy validation script (under asgard-legacy project	Test Name
tests/test_models_body.py: ✓ test_earth_body_convert	tests/test_models_explorer.py: ✓ test_explorer_earth_body_convert	[LLF_CTOG_GTOC] - Cartesian to geodetic / Geodetic to cartesian
tests/test_models_body.py: ✓ test_geodetic_distance	tests/test_models_explorer.py: ✓ test_explorer_earth_body _geodetic_distance	[LLF_GEODIS] - Geodetic Distance
tests/test_models_body.py: ✓ test_earth_body_change_reference _frame	tests/test_models_explorer.py: ✓ test_explorer_earth_body_change _reference_frame	[LLF_REFFC] - Reference frame change (EF, EME2000, …)
tests/test_models_body.py: ✓ test_earth_body_sun_pv	tests/test_models_explorer.py: ✓ test_explorer_earth_body_sun_pv	[LLF_SUNPOS] – Sun position

11.5 Quality tests results

11.5.1 [LLF_TSC] – Time Scale Conversion

[LLF_TSC] – Time Scale Conversion
Summary: *This test aims at testing the conversion between* TAI, UTC, UT1 and GPS time scales, to verify that ASGARD gives the same results as legacy. Actors: *CS GROUP* Expected Results: *Verify, both for ASGARD and legacy, that difference between converted timescales and expected timescales is below threshold as expressed in §2.3.4* Low level functions thresholds Pass/Fail criteria: *For conversion between time scales:* ✓ *No differences between converted and expected time scale are above requirement,* Test Data: See *Test Case #1* *in §11.3.1 Time Reference dataset*
Preconditions:

[LLF_TSC] – Time Scale Conversion			
#:	Step actions:	Expected Results:	Results:
1	For ASGARD, in ***tests/test_models_time.py*** Run: ***test_time_reference_convert***	For each timescale conversion data, the difference between the converted timescale and the expected timescale is below threshold	Conversions: ✓ TAI -> UTC: **PASSED** ✓ TAI -> UT1: **PASSED** ✓ TAI -> GPS: **PASSED** ✓ UTC -> TAI: **PASSED** ✓ UTC -> UT1: **PASSED** ✓ UTC -> GPS: **PASSED** ✓ GPS -> TAI: **PASSED** ✓ GPS -> UTC: **PASSED** ✓ GPS -> UT1: **PASSED**
2	For legacy, in **tests/tests_models_explorer.py** *Run:* ***test_time_reference_convert***	For each timescale conversion data, the difference between the converted timescale and the expected timescale is below threshold	Conversions: ✓ TAI -> UTC: **PASSED** ✓ TAI -> UT1: **PASSED** ✓ TAI -> GPS: **PASSED** ✓ UTC -> TAI: **PASSED** ✓ UTC -> UT1: **PASSED** ✓ UTC -> GPS: **PASSED** ✓ GPS -> TAI: **PASSED** ✓ GPS -> UTC: **PASSED** ✓ GPS -> UT1: **PASSED**

11.5.2 [LLF_LSQ] - Leap Second Query

[LLF_LSQ] - Leap Second Query			
Summary: *This test aims at testing the detection of leap second, to verify that ASGARD gives the same results as legacy.* Actors: *CS GROUP* Expected Results: *Verify that leap seconds are correctly computed between a start and end time, both for ASGARD and legacy* Pass/Fail criteria: *The list of leap seconds equals expected leap seconds.* Test Data: *See Test Case #2 §11.3.1 Time Reference dataset*			
Preconditions:			
#:	Step actions:	Expected Results:	Results:
1	For ASGARD, In ***tests/test_models_time.py*** Run ***test_time_reference_leap_second***	✓ For standard dates, expressed in processing format, ✓ For dates at limit condition for leap seconds,	Leap seconds between: ✓ 4745.0 and 6570.0 PASSED ✓ 2015-06-30T23:59:59 and 2015-07-01T00:00:00 PASSED

[LLF_LSQ] - Leap Second Query			
		the list of leap seconds equals expected leap seconds	
2	For legacy, In ***tests/tests_models_explorer.py*** Run ***test_time_reference_leap_second***	✓ For standard dates, expressed in processing format, ✓ For dates at limit condition for leap seconds, the list of leap seconds equals expected leap seconds	Leap seconds between: ✓ 4745.0 and 6570.0 **PASSED** ✓ 2015-06-30_23:59:59 and 2015-07-01_00:00:00 **PASSED**

11.5.3 [LLF_ATOP_PTOA] - ASCII To Processing / Processing To ASCII

[LLF_ATOP_PTOA] - ASCII To Processing / Processing To ASCII
Summary: *This test aims at testing conversions between ASCII format and processing format, with different time scales, without or with leap second, to verify that ASGARD gives the same results as legacy* Actors: *CS GROUP* Expected Results: *Verify that, both for ASGARD and* legacy: ✓ *dates are exactly converted to the expected dates either in ASCII format or processing format*

[LLF_ATOP_PTOA] - ASCII To Processing / Processing To ASCII			
Pass/Fail criteria: *For conversion from processing to ASCII format and ASCII to processing format:* ✓ *Dates are exactly equals to expected,* Test Data: *See Test Case #3 in §11.3.1 Time Reference dataset*			
Preconditions:			
#:	Step actions:	Expected Results:	Results:
1	For ASGARD, In ***tests/test_models_time.py*** Run ***test_time_reference_ascii***	✓ Conversion from processing format in TAI scale to ASCII formats ✓ Conversion from ASCII formats to processing format in TAI scale ✓ Conversion from processing format in UTC scale to ASCII format ✓ Conversion from date with real offset to STD format ✓ Conversion from processing in different scale to STD format, where offset not detected ✓ Conversion on the fly from processing to STD format, from TAI to UTC scale	Processing format in TAI scale to ASCII formats ✓ **STD** **PASSED** ✓ **STD_MICROSEC** **PASSED** ✓ **COMPACT** **PASSED** ✓ **CCSDSA_MICROSEC** **PASSED** ASCII format to processing format in TAI scale ✓ **STD** **PASSED** ✓ **CCSDSA_MICROSEC** **PASSED** Processing format in UTC scale to ASCII format ✓ **STD** **PASSED** Date with real offset

[LLF_ATOP_PTOA] - ASCII To Processing / Processing To ASCII			
		are exactly converted to the expected date either in ASCII format or processing format.	✓ **STD** **PASSED** From processing in different scale to STD format, where offset not detected: ✓ **TAI** **PASSED** ✓ **UTC** **PASSED** On the fly from processing to STD format, from TAI to UTC scale: ✓ **TAI** 8338.00042824 **PASSED** ✓ **TAI** 8338.00077546 **PASSED**
2	For legacy, In ***tests/tests_models_explorer.py*** Run ***test_time_reference_ascii***	✓ Conversion from processing format in TAI scale to ASCII formats ✓ Conversion from ASCII formats to processing format in TAI scale ✓ Conversion from processing format in UTC scale to ASCII format ✓ Conversion from date with real offset to STD format ✓ Conversion from processing in different scale to STD format, where offset not detected ✓ Conversion on the fly from processing to STD format, from TAI to UTC scale	Processing format in TAI scale to ASCII formats ✓ **STD** **PASSED** ✓ **STD_MICROSEC** **PASSED** ✓ **COMPACT** **PASSED** ✓ **CCSDSA_MICROSEC** **PASSED** ASCII format to processing format in TAI scale ✓ **STD** **PASSED** ✓ **CCSDSA_MICROSEC** **PASSED** Processing format in UTC scale to ASCII format ✓ **STD** **PASSED** Date with real offset ✓ **STD** **PASSED**

[LLF_ATOP_PTOA] - ASCII To Processing / Processing To ASCII			
		are exactly converted to the expected date either in ASCII format or processing format.	From processing in different scale to STD format, where offset not detected: ✓ **TAI** PASSED ✓ **UTC** PASSED On the fly from processing to STD format, from TAI to UTC scale: ✓ **TAI** 8338.00042824 PASSED ✓ **TAI** 8338.00077546 PASSED
3	For ASGARD, In ***tests/test_models_time.py*** Run ***test_time_reference_ascii_leap***	✓ Conversion of dates, around leap seconds, expressed in STD format to processing format, ✓ Conversion of date expressed in processing format in UTC scale: ➢ to ASCII STD format around leap second not detected, ➢ converted in TAI, then to ASCII STD in UTC detected, are exactly converted to the expected date.	Dates expressed in STD format with leap seconds: ✓ 2016-12-31_23:59:59 PASSED ✓ 2016-12-31_23:59:60 PASSED ✓ 2017-01-01_00:00:00 PASSED Date expressed in processing format in UTC scale: ✓ 6210.0 UTC: leap second not detected PASSED ✓ 6210.0 UTC -> TAI: leap second detected PASSED
4	For legacy, In ***tests/tests_models_explorer.py*** Run ***test_time_reference_ascii***	✓ Conversion of dates, around leap seconds, expressed in STD format to processing format, ✓ Conversion of date expressed in processing format in UTC scale: ➢ to ASCII STD format around leap second not detected,	Dates expressed in STD format with leap seconds: ✓ 2016-12-31_23:59:59 PASSED ✓ 2016-12-31_23:59:60 PASSED ✓ 2017-01-01_00:00:00 PASSED Date expressed in processing format in UTC scale: ✓ 6210.0 UTC: leap second not detected PASSED

[LLF_ATOP_PTOA] - ASCII To Processing / Processing To ASCII			
		➢ converted in TAI, then to ASCII STD in UTC detected, are exactly converted to the expected date.	✓ 6210.0 UTC -> TAI: leap second detected **PASSED**

11.5.4 [LLF_CUCTOP] - CUC to processing

[LLF_CUCTOP] - CUC to processing
Summary: *This test aims at testing GPS dates, in CUC format, conversion to processing format, to verify that ASGARD gives the same results as legacy.* Actors: *CS GROUP* Expected Results: *Verify, both for ASGARD and legacy, that difference between converted date and expected date is below threshold as expressed in §2.3.4* Low level functions thresholds Pass/Fail criteria: *For each conversion of dates:* ✓ *No differences between converted and expected date are above requirement.*

[LLF_CUCTOP] - CUC to processing			
Test Data: *See Test Case #4 in §11.3.1 Time Reference dataset*			
Preconditions:			
#:	Step actions:	Expected Results:	Results:
1	For ASGARD, In ***tests/test_models_time.py*** Run ***test_time_reference_cuc***	For each GPS date, in CUC format, when converted to the processing format, the difference between the converted date and the expected date is below threshold	Conversion of: ✓ [1336436015, 0] **PASSED** ✓ [1336436015, 1000] **PASSED** ✓ [1336436015, 1000000] **PASSED** ✓ [1336436015, 10000000] **PASSED** ✓ [1336522415, 0] **PASSED**
2	For legacy, In ***tests/tests_models_explorer.py*** Run ***test_time_reference_cuc***	For each GPS date, in CUC format, when converted to the processing format, the difference between the converted date and the expected date is below threshold	Conversion of: ✓ [1336436015, 0] **PASSED** ✓ [1336436015, 1000] **PASSED** ✓ [1336436015, 1000000] **PASSED** ✓ [1336436015, 10000000] **PASSED** ✓ [1336522415, 0] **PASSED**

11.5.5 [LLF_TTOP] - Transport to processing

[LLF_TTOP] - Transport to processing
Summary:

[LLF_TTOP] - Transport to processing			
This test aims at testing dates, in transport format (days/seconds/microsec), conversion to processing format, to verify that ASGARD gives the same results as legacy. Actors: *CS GROUP* Expected Results: *Verify, both for ASGARD and legacy, that difference between converted date and expected date is below threshold as expressed in §2.3.4* Low level functions thresholds Pass/Fail criteria: *For each conversion of dates:* ✓ *No differences between converted and expected date are above requirement.* Test Data: *See Test Case #5 in §11.3.1 Time Reference dataset*			
Preconditions:			
#:	Step actions:	Expected Results:	Results:
1	For ASGARD, In ***tests/test_models_time.py*** Run ***test_time_reference_transport***	For each transport date format (days/seconds/microsec), when converted to the processing format, the difference between the	Conversion of: ✓ [8320, 2122, 71836] **PASSED** ✓ [8320, 2122, 115836] **PASSED** ✓ [8320, 2122, 159835] **PASSED**

[LLF_TTOP] - Transport to processing			
		converted date and the expected date is below threshold	✓ [8320, 2122, 203835] **PASSED**
2	For legacy, In ***tests/tests_models_explorer.py*** Run ***test_time_reference_transport***	For each transport date format (days/seconds/microsec), when converted to the processing format, the difference between the converted date and the expected date is below threshold	Conversion of: ✓ [8320, 2122, 71836] **PASSED** ✓ [8320, 2122, 115836] **PASSED** ✓ [8320, 2122, 159835] **PASSED** ✓ [8320, 2122, 203835] **PASSED**

11.5.6 [LLF_CTOG_GTOC] - Cartesian to geodetic / Geodetic to cartesian

[LLF_CTOG_GTOC] - Cartesian to geodetic / Geodetic to cartesian
Summary: *This test aims at testing conversion between geodetic and cartesian points, to verify that ASGARD gives the same results as legacy.* Actors: *CS GROUP* Expected Results: *Verify, both for ASGARD and legacy, that* ✓ *difference between converted geodetic point and initial geodetic point,*

[LLF_CTOG_GTOC] - Cartesian to geodetic / Geodetic to cartesian			
✓ *the norm of the difference of converted cartesian point and expected cartesian point,* *is below threshold as expressed in in §2.3.4* Low level functions thresholds Pass/Fail criteria: *For each conversion:* ✓ *No differences between converted and expected geodetic point are above requirement.* ✓ *No norms of the difference between converted and expected cartesian point are above requirement.* Test Data: *See TestCase #1 in §11.3.2 Earth Body dataset*			
Preconditions:			
#:	Step actions:	Expected Results:	Results:
1	For ASGARD, In ***tests/ test_models_body.py*** Run ***test_earth_body_convert***	For each geodetic position, when converted to cartesian and then back to geodetic, the difference between the final geodetic point and the initial geodetic point is below threshold. For each geodetic position, when converted to cartesian, the norm of the difference between the converted and the expected cartesian point is below threshold.	Conversion of geodetic positions to cartesian and back to geodetic: ✓ [-40.0, 25.0, 62.0] **PASSED** ✓ [45.0, -12.0, 45.0] **PASSED** ✓ [175.0, 45.0, 12.0] **PASSED** ✓ [64.0, -86.0, 150.0] **PASSED** Conversion of geodetic positions to cartesian: ✓ [-40.0, 25.0, 62.0] **PASSED** ✓ [45.0, -12.0, 45.0] **PASSED** ✓ [175.0, 45.0, 12.0] **PASSED** ✓ [64.0, -86.0, 150.0] **PASSED**

[LLF_CTOG_GTOC] - Cartesian to geodetic / Geodetic to cartesian			
2	For legacy, In ***tests/tests_models_explorer.py*** Run ***test_explorer_earth_body_convert***	For each geodetic position, when converted to cartesian and then back to geodetic, the difference between the final geodetic point and the initial geodetic point is below threshold. For each geodetic position, when converted to cartesian, the norm of the difference between the converted and the expected cartesian point is below threshold.	Conversion of geodetic positions to cartesian and back to geodetic: ✓ [320, 25.0, 62.0] **PASSED** ✓ [45.0, -12.0, 45.0] **PASSED** ✓ [175.0, 45.0, 12.0] **PASSED** ✓ [64.0, -86.0, 150.0] **PASSED** Conversion of geodetic positions to cartesian: ✓ [-40.0, 25.0, 62.0] **PASSED** ✓ [45.0, -12.0, 45.0] **PASSED** ✓ [175.0, 45.0, 12.0] **PASSED** ✓ [64.0, -86.0, 150.0] **PASSED**

11.5.7 [LLF_GEODIS] - Geodetic Distance

[LLF_GEODIS] - Geodetic Distance
Summary: *This test aims at compute the geodetic distance between a first and a second geodetic points, the relative azimuth between the first and the second geodetic points, and the relative azimuth between the second and the first geodetic points, to verify that ASGARD gives the same results as legacy.* Actors:

<table>
<tr><th colspan="4">[LLF_GEODIS] - Geodetic Distance</th></tr>
<tr><td colspan="4">CS GROUP

<u>Expected Results:</u>

Verify for ASGARD
✓ that difference between computed geodetic distance and legacy geodetic distance is below threshold
✓ that difference between computed azimuth and legacy azimuth is below threshold
as expressed in §2.3.4 Low level functions thresholds

Verify for legacy
✓ that difference between computed geodetic distance and reference geodetic distance is below threshold
✓ that difference between computed azimuth and reference azimuth is below threshold
as expressed in §2.3.4 Low level functions thresholds

<u>Pass/Fail criteria:</u>

For each pair of geodetic points:
✓ No differences between computed and expected distance are above requirement.
✓ No differences between computed and expected azimuth are above requirement.

<u>Test Data:</u>

See <u>TestCase #2</u> in §11.3.2 Earth Body dataset</td></tr>
<tr><td colspan="4"><u>Preconditions:</u></td></tr>
<tr><td>#:</td><td><u>Step actions:</u></td><td><u>Expected Results:</u></td><td><u>Results:</u></td></tr>
</table>

[LLF_GEODIS] - Geodetic Distance			
1	For ASGARD, In ***tests/ test_models_body.py*** Run ***test_geodetic_distance***	For each geodetic distance between first and second point, the difference between the computed distance and the legacy distance is below threshold. For each relative azimuth between first and second point, the difference between the computed azimuth and the legacy azimuth is below threshold. For each relative azimuth between second and first point, the difference between the computed azimuth and the legacy azimuth is below threshold.	Geodetic distance between: ✓ [10.0, 0.0] and [20.0, 0.0] **PASSED** ✓ [10.0, 50.0] and [20.0, 50.0] **PASSED** ✓ [10.0, 50.0] and [40.0, 50.0] **PASSED** ✓ [10.0, 45.0] and [20.0, 55.0] **PASSED** ✓ [10.0, 30.0] and [10.0, 40.0] **PASSED** ✓ [4.17, 10.0] and [4.1705, 10.001] **PASSED** Azimuth between first and second points: ✓ [10.0, 0.0] and [20.0, 0.0] **PASSED** ✓ [10.0, 50.0] and [20.0, 50.0] **PASSED** ✓ [10.0, 50.0] and [40.0, 50.0] **PASSED** ✓ [10.0, 45.0] and [20.0, 55.0] **PASSED** ✓ [10.0, 30.0] and [10.0, 40.0] **PASSED** ✓ [4.17, 10.0] and [4.1705, 10.001] **PASSED** Azimuth between second and first points: ✓ [20.0, 0.0] and [10.0, 0.0] **PASSED** ✓ [20.0, 50.0] and [10.0, 50.0] **PASSED** ✓ [40.0, 50.0] and [10.0, 50.0] **PASSED** ✓ [20.0, 55.0] and [10.0, 45.0] **PASSED** ✓ [10.0, 40.0] and [10.0, 30.0] **PASSED** ✓ [4.1705, 10.001] and [4.17, 10.0] **PASSED**
2	For legacy, In ***tests/tests_models_explorer.py***	For each geodetic distance between first and second point, the difference between the computed	Geodetic distance between: ✓ [10.0, 0.0] and [20.0, 0.0] **PASSED** ✓ [10.0, 50.0] and [20.0, 50.0] **PASSED**

[LLF_GEODIS] - Geodetic Distance			
	Run ***test_explorer_earth_body_geodetic_distance***	distance and the reference distance is below threshold. For each relative azimuth between first and second point, the difference between the computed azimuth and the reference azimuth is below threshold. For each relative azimuth between second and first point, the difference between the computed azimuth and the reference azimuth is below threshold.	✓ [10.0, 50.0] and [40.0, 50.0] **PASSED** ✓ [10.0, 45.0] and [20.0, 55.0] **PASSED** ✓ [10.0, 30.0] and [10.0, 40.0] **PASSED** ✓ [4.17, 10.0] and [4.1705, 10.001] **PASSED** Azimuth between first and second points: ✓ [10.0, 0.0] and [20.0, 0.0] **PASSED** ✓ [10.0, 50.0] and [20.0, 50.0] **PASSED** ✓ [10.0, 50.0] and [40.0, 50.0] **PASSED** ✓ [10.0, 45.0] and [20.0, 55.0] **PASSED** ✓ [10.0, 30.0] and [10.0, 40.0] **PASSED** ✓ [4.17, 10.0] and [4.1705, 10.001] **PASSED** Azimuth between second and first points: ✓ [20.0, 0.0] and [10.0, 0.0] **PASSED** ✓ [20.0, 50.0] and [10.0, 50.0] **PASSED** ✓ [40.0, 50.0] and [10.0, 50.0] **PASSED** ✓ [20.0, 55.0] and [10.0, 45.0] **PASSED** ✓ [10.0, 40.0] and [10.0, 30.0] **PASSED** ✓ [4.1705, 10.001] and [4.17, 10.0] **PASSED**

11.5.8 [LLF_REFFC] - Reference frame change (EF, EME2000, …)

[LLF_REFFC] - Reference frame change (EF, EME2000, …)
<u>Summary:</u> *This test aims at testing conversion of (position, velocity) in EF frame to J2000 frame and back to EF frame, to verify that ASGARD gives the same results as legacy.* <u>Actors:</u> *CS GROUP* <u>Expected Results:</u> *Verify, both for ASGARD and legacy, that the norm of the difference for position and velocity* ✓ *between computed in J2000 frame and expected in J2000 frame, in absolute and relative value, is below threshold* ✓ *between computed in EF frame and original in EF frame, in absolute value, is below threshold* *as expressed in §2.3.4* Low level functions thresholds <u>Pass/Fail criteria:</u> *For each conversion:* ✓ *No norms of absolute or relative differences between converted, in J2000 frame, and expected, in J2000 frame, position and velocity are above requirement.* ✓ *No norms of absolute differences between converted, in EF frame, and original, in EF frame, position and velocity are above requirement.* <u>Test Data:</u>

[LLF_REFFC] - Reference frame change (EF, EME2000, …)			
See [TestCase #3](#) in §11.3.2 Earth Body dataset			
Preconditions:			
#:	Step actions:	Expected Results:	Results:
1	For ASGARD, In ***tests/ test_models_body.py*** Run ***test_earth_body_change_reference _frame***	For each (time, position, velocity) in EF frame, converted in J2000 frame, the norm of the difference between the computed (position, velocity)$_{J2000}$ and the reference (position, velocity)$_{J2000}$ is below threshold: ✓ in absolute value ✓ in relative value	Conversion of: ✓ (8338.06258..., [4221826...], [-1954...]) **PASSED** ✓ (8338.0629..., [4161165...], [-2089...]) **PASSED** ✓ (8338.0632..., [4096499...], [-2221...]) **PASSED** ✓ (8338.0636..., [4027918...], [-2350...]) **PASSED** ✓ (8338.06396..., [3955511...], [-2476...]) **PASSED**
2	For legacy, In ***tests/tests_models_explorer.py*** Run ***test_explorer_earth_body_change _reference_frame***	For each (time, position, velocity) in EF frame, converted in J2000 frame, the norm of the difference between the computed (position, velocity)$_{J2000}$ and the reference (position, velocity)$_{J2000}$ is below threshold: ✓ in absolute value ✓ in relative value	Conversion of: ✓ (8338.06258..., [4221826...], [-1954...]) **PASSED** ✓ (8338.0629..., [4161165...], [-2089...]) **PASSED** ✓ (8338.0632..., [4096499...], [-2221...]) **PASSED** ✓ (8338.0636..., [4027918...], [-2350...]) **PASSED** ✓ (8338.06396..., [3955511...], [-2476...]) **PASSED**
3	For ASGARD, In ***tests/ test_models_body.py*** Run ***test_earth_body_change_reference _frame***	For each (time, position, velocity) in EF frame, converted in J2000 frame and back to EF frame, the norm of the difference between the computed (position, velocity)$_{EF}$ and the original (position, velocity)$_{EF}$ is below threshold in absolute value	Conversion of: ✓ (8338.06258..., [4221826...], [-1954...]) **PASSED** ✓ (8338.0629..., [4161165...], [-2089...]) **PASSED** ✓ (8338.0632..., [4096499...], [-2221...]) **PASSED** ✓ (8338.0636..., [4027918...], [-2350...]) **PASSED** ✓ (8338.06396..., [3955511...], [-2476...]) **PASSED**

[LLF_REFFC] - Reference frame change (EF, EME2000, …)			
4	For legacy, In ***tests/tests_models_explorer.py*** Run ***test_explorer_earth_body_change_reference_frame***	For each (time, position, velocity) in EF frame, converted in J2000 frame and back to EF frame, the norm of the difference between the computed (position, velocity)$_{EF}$ and the original (position, velocity)$_{EF}$ is below threshold in absolute value	Conversion of: ✓ (8338.06258..., [4221826...], [-1954...]) **PASSED** ✓ (8338.0629..., [4161165...], [-2089...]) **PASSED** ✓ (8338.0632..., [4096499...], [-2221...]) **PASSED** ✓ (8338.0636..., [4027918...], [-2350...]) **PASSED** ✓ (8338.06396..., [3955511...], [-2476...]) **PASSED**

11.5.9 [LLF_SUNPOS] – Sun position

[LLF_SUNPOS] – Sun position
Summary: *This test aims at computing Sun position and velocity, to verify that ASGARD gives the same results as legacy.* Actors: *CS GROUP* Expected Results: *Verify, both for ASGARD and legacy, that the norm of the difference for position and for velocity of the Sun* ✓ *between computed and expected, in absolute and relative value, in EF frame is below threshold* *as expressed in § 2.3.4* Low level functions thresholds

[LLF_SUNPOS] – Sun position			
Pass/Fail criteria: *For each date of computation for the Sun:* ✓ *No norms of the absolute difference between computed and expected position and velocity are above requirement* ✓ *No norms of the relative difference between computed and expected position and velocity are above requirement* Test Data: *See TestCase #4 in §11.3.2 Earth Body dataset*			
Preconditions:			
#:	Step actions:	Expected Results:	Results:
1	For ASGARD, In ***tests/ test_models_body.py*** Run ***test_earth_body_sun_pv***	For each date, the difference between the norm of the difference of the computed Sun position/velocity and the legacy Sun position/velocity is below threshold. For each date, the difference between the norm of (the difference of the computed Sun position/velocity and the legacy Sun position/velocity)/legacy Sun position/velocity is below threshold.	Compute absolute difference of the Sun position and velocity at JD (timescale GPS) ✓ 8338.06423611 **PASSED** ✓ 8338.06523611 **PASSED** ✓ 8338.06623611 **PASSED** ✓ 8338.0672361 **PASSED** ✓ 8338.06823611 **PASSED** ✓ 8338.06923611 **PASSED** ✓ 8338.07023611 **PASSED** ✓ 8338.07123611 **PASSED** ✓ 8338.07223611 **PASSED** ✓ 8338.07323611 **PASSED**

[LLF_SUNPOS] – Sun position			
			Compute relative difference of the Sun position and velocity at JD (timescale GPS) ✓ 8338.06423611 **PASSED** ✓ 8338.06523611 **PASSED** ✓ 8338.06623611 **PASSED** ✓ 8338.0672361 **PASSED** ✓ 8338.06823611 **PASSED** ✓ 8338.06923611 **PASSED** ✓ 8338.07023611 **PASSED** ✓ 8338.07123611 **PASSED** ✓ 8338.07223611 **PASSED** ✓ 8338.07323611 **PASSED**
2	For legacy, In ***tests/tests_models_explorer.py*** Run ***test_explorer_earth_body_sun_pv***	For each date, the difference between the norm of the difference of the computed Sun position/velocity and the expected Sun position/velocity is below threshold. For each date, the difference between the norm of (the difference of the computed Sun position/velocity and the expected Sun position/velocity)/ expected Sun position/velocity is below threshold.	Compute absolute difference of the Sun position and velocity at JD (timescale GPS) ✓ 8338.06423611 **PASSED** ✓ 8338.06523611 **PASSED** ✓ 8338.06623611 **PASSED** ✓ 8338.0672361 **PASSED** ✓ 8338.06823611 **PASSED** ✓ 8338.06923611 **PASSED** ✓ 8338.07023611 **PASSED** ✓ 8338.07123611 **PASSED** ✓ 8338.07223611 **PASSED** ✓ 8338.07323611 **PASSED** Compute relative difference of the Sun position and velocity at JD (timescale GPS)

[LLF_SUNPOS] – Sun position			
			✓ 8338.06423611 **PASSED** ✓ 8338.06523611 **PASSED** ✓ 8338.06623611 **PASSED** ✓ 8338.0672361 **PASSED** ✓ 8338.06823611 **PASSED** ✓ 8338.06923611 **PASSED** ✓ 8338.07023611 **PASSED** ✓ 8338.07123611 **PASSED** ✓ 8338.07223611 **PASSED** ✓ 8338.07323611 **PASSED**

12. VALIDATION CONCLUSIONS

12.1 Validation tests lists results

All validation of S2MSI using zarr have been done with GETAS in zarr due to issue with Copernicus DEM in Zarr => #197

Table 42 –Validation tests list

Family	Test	Status
Quality	[S2MSI_TDS1_L0c_DEM-010]	PASSED with limitations => #204
Quality	[S2MSI_TDS1_L0c_CONST-010]	FAILED => #193
Quality	[S2MSI_TDS1_L1B_DEM-010]	FAILED => #200
Quality	[S2MSI_TDS1_L1C_DEM-010]	FAILED => #200
Quality	[S2MSI_TDS1_L1C_DEM_INVLOC-010]	FAILED => #203, #208
Quality	[S2MSI_TDS2_L0c_DEM-010]	FAILED => #204, #198, #199
Quality	Test is failing due to RAM issue as the RAM is going up and up between loops over sensors => #199 [S2MSI_TDS2_L0c_CONST-010]	FAILED => #193, #177
Quality	[S2MSI_TDS2_L1B_DEM-010]	FAILED => #198, #199
Quality	[S2MSI_TDS2_L1C_DEM-010]	FAILED => #200, #201
Quality	[S2MSI_TDS2_L1C_DEM_INVLOC-010]	FAILED => #203, #208
Quality	[S2MSI_TDS3_L1B_DEM-010]	FAILED => #207, #200, #199
Quality	[S2MSI_TDS3_L1C_DEM_INVLOC-010]	FAILED => #202, #203
Quality	[S2MSI_TDS4_L1B_DEM-010]	FAILED => #200, #199
Quality	[S2MSI_TDS4_L1C_DEM_INVLOC-010]	FAILED => #203, #208
Quality	[S2MSI_TDS5_L0u_DEM-010]	PASSED with limitations => #204
Quality	[S2MSI_TDS5_L0c_DEM-010]	FAILED => #204, #199
Quality	[S2MSI_TDS5_L0c_CONST-010]	FAILED => #193
Quality	[S2MSI_TDS5_L1B_DEM-010]	FAILED => #204, #199
Quality	[S2MSI_TDS5_L1C_DEM-010]	PASSED with limitations => #204
Quality	[S2MSI_TDS5_L1C_DEM_INVLOC-010]	FAILED => #203, #208
Perf	[S2MSI_PERF]	Still to be optimized

Family	Test	Status
Quality	[S3OLCI_TDS1_EO_LEGACY] - Geo referencing	PASSED (see explanations)
Quality	[S3OLCI_TDS1_RAC_LEGACY] - Acquisition Geometry	PASSED
Quality	[S3OLCI_TDS1_EO] - Geo referencing	PASSED (see explanations)
Quality	[S3OLCI_TDS1_RAC] - Acquisition Geometry	PASSED
Quality	[S3SLSTR_TDS1_IG_LEGACY] - Instrument Grid (S1-L1A_5_3)	PASSED for some geometry tests case FAILED => #186
Quality	[S3SLSTR_TDS1_TPG_LEGACY] - Tie Point Grid (S1-L1A_5_3)	FAILED => #186
Quality	[S3SLSTR_TDS1_QCG_LEGACY] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)	PASSED
Quality	[S3SLSTR_TDS1_IG] - Instrument Grid (S1-L1A_5_3)	PASSED for some geometry tests case FAILED => #186
Quality	[S3SLSTR_TDS1_TPG] - Tie Point Grid (S1-L1A_5_3)	FAILED => #186
Quality	[S3SLSTR_TDS1_QCG] - Quasi Cartesian Grid (S1-L1A_5_1 & S1-L1A_5_4)	FAILED => #186
Quality	[LLF_TSC] – Time Scale Conversion	PASSED
Quality	[LLF_LSQ] - Leap Second Query	PASSED
Quality	[LLF_ATOP_PTOA] - ASCII To Processing / Processing To ASCII	PASSED
Quality	[LLF_CUCTOP] - CUC to processing	PASSED
Quality	[LLF_TTOP] - Transport to processing	PASSED
Quality	[LLF_CTOG_GTOC] - Cartesian to geodetic / Geodetic to cartesian	PASSED
Quality	[LLF_GEODIS] - Geodetic Distance	PASSED
Quality	[LLF_REFFC] - Reference frame change (EF, EME2000, …)	PASSED
Quality	[LLF_SUNPOS] – Sun position	PASSED

12.2 List of validated functions

12.2.1 S2MSI

Even if a lot of tests are presented as FAILED, results are globally very good at CE95. Some specific points give strange results and must be investigated but ASGARD is overall performant. Metrics might be over thresholds, but all functionalities give far enough good results to be used for Sentinel-2 for now, except:

- ✓ Inverse location at constant altitude,
- ✓ Refining for L1B and L1C models,
- ✓ Special cases and especially for inverse location grids.

Finally, footprint generation is very slow but seems to give exploitable, even good in most cases, results.

Regarding performances, ASGARD V4 was not deeply tested, and RAM was not strongly assessed for this version but what have been observed is:

- ✓ Large processing time overlay (but optimisation not done yet)
- ✓ Large RAM consumption for footprint generation (issue created to investigate it).

However, regarding inverse location grids,

- ✓ Legacy-based: There is a non-neglectable impact which can come from:
  - ➢ Overlay due to interfacing between python and java,
  - ➢ Processor are processed all bands at once in the Legacy reference, with optimisations, and only sensor per sensor in ASGARD, which implies a lack of one level of optimisation.
- ✓ Refactored: Not optimised yet, but for now it is unacceptable, and it won't be possible to put it in operation like that.

12.2.2 S3OLCI & S3SLSTR Legacy based

Main geometric functions for Sentinel-3 instruments (OLCI in Earth Observation mode, OLCI Radiometric mode and SLSTR) **have been validated with ASGARD legacy-based implementation**.

For OLCI all geolocations, sun angles and incidence angles have been validated with a very low threshold error.

For SLSTR, from the last version of this validation plan, we change the subset of coordinates for direct location, instead of taking the last 10% of values, we have taken a value out of ten which is more relevant.

The main functionality of the instrument validated concern the NADIR geolocation on instrument grid for the nadir geometries, which passed with an error inferior to the associated spatial resolution. It seems that there is a problem with location using oblique geometries. This has been investigated using source code ICF, but it is a version using EOCFI 4.17, while we are using EOCFI 4.23 for the tests here. With the appropriate version of the source code, we will continue our investigations.

As concern geolocations on tie point grid, it still need investigation with the appropriate EOCFI source code as for instrument grid.

As concern sun angles and incidence angles computation on tie point grid, all the problems seem related to direct location computation, as it propagates to angles computation, so the problems could be resolved once the direct location is solved.

The geolocation on quasi cartesian grid is validated.

12.2.3 S3OLCI & S3SLSTR refactored

Main geometric functions for Sentinel-3 instruments (OLCI in Earth Observation mode, OLCI Radiometric mode and SLSTR) **have been validated with ASGARD refactored implementation**.

For OLCI all geolocations have been validated with a very low threshold error.

As concern the sun angles, the differences are due to the fact that Orekit is using another method of sun position computation compared to EOCFI. By using EOCFI computation of sun position, there are no errors on sun angles.

For SLSTR, the main functionality of the instrument validated concern the NADIR geolocation on instrument grid for the nadir geometries which passed with an error inferior to the associated spatial resolution. It seems that there is a problem with location using oblique geometries. This has been investigated using source code ICF, but it is a version using EOCFI 4.17, while we are using EOCFI 4.23 for the tests here. With the appropriate version of the source code, we will continue our investigations.

As concern geolocations on tie point grid, it still need investigation with the appropriate EOCFI source code as for instrument grid.

Regarding sun angles and incidence angles computation on tie point grid, all the problems seem related to direct location computation, as it propagates to angles computation, so the problems could be resolved once the direct location is solved.

The geolocation on quasi cartesian grid have the same problems due to direct location as for tie point grid.

12.2.4 Low level functions

All the low-level functions have been validated.

12.2.5 General conclusions

Processing time might not be totally optimized yet and RAM consumption has not been deeply assessed. It must be analysed in future version, but especially when Zarr DEM usage will be possible.

But, overall, ASGARD V4 can be used as a geometry library to assess feasibilities or making some prototype using it. Results are globally good and can be used normally, except for several issues that has been raised and that will be investigated for future versions (see § 12.3 Anomalies).

12.3 Anomalies

The following tickets were opened to following troubleshooting points highlighted in validation of V3:

- ✓ #153: Fix the GenericPlatformModel for MSI (ASGARD V3)
- ✓ #173: SLSTR computation of X, Y coordinates on QC grid from tie point grid geolocation
- ✓ #174: improves accuracy of sun/incidence angles from tie point grid geolocation on SLSTR
- ✓ #175: Direct locations are far from reference for S2MSI with ASGARD (V3) Legacy-based
- ✓ #176: Inverse locations are far from reference for S2MSI (ASGARD V3)
- ✓ #177: Inverse locations are really far from reference for S2MSi with constant altitude set (ASGARD V3)
- ✓ #178: NullPointerException when calling ASGARD (V3) Legacy-based on S2MSI_TDS5

Regarding issues about V4, only 177 is still present, but all others are new issues:

- ✓ #177: Inverse locations are really far from reference for S2MSI with constant altitude set (ASGARD V3)
- ✓ #186: SLSTR legacy-based validation imprecision for direct_loc on tie point grid for 05KM group
- ✓ #193: Wrong altitude when direct location at constant altitude over DEM_Legacy geoid
- ✓ #197: Support Copernicus DEM in ZARR format
- ✓ #198: Bug with footprint on dataset TDS2

- ✓ #199: S2MSI: RAM consumption important with REFACTORED code for L1B and L1C product when computing footprint
- ✓ #200: S2MSI: Diff in plani of 5-15 m for L1B and L1C level
- ✓ #201: S2MSI validation: ValueError: Input X contains Nan
- ✓ #202: S2MSI: Fatal Python error: Segmentation fault
- ✓ #203: => Closed in favour of 204 as inverse location algorithm calls direct locations.
- ✓ #204: S2MSI: Direct location a little far from references
- ✓ #207: S2MSI LEGACY-BASED direct location far from reference
- ✓ #208: S2MSI inverse location returning NonType

Issues are visible at: https://gitlab.eopf.copernicus.eu/geolib/asgard/

13. APPENDIX

13.1 Tools

Several tools have been implemented to make senseful comparison on different metrics. There are summarized in this section.

13.1.1 Planar captor distance

Compute Euclidian distance (error) between two sets of coordinates in row/col.

This function (*planar_captor_error*) is available in file tests/helpers/compare.py of ASGARD.

13.1.2 Height error

Compute altimetric error between two sets of geodetic coordinates.

This function (*height_error*) is available in file tests/helpers/compare.py of ASGARD.

13.1.3 Planar distance

Compute planimetric error between two sets of geodetic coordinates. The altitude used comes from the first set.

This function (*planar_error*) is available in file tests/helpers/compare.py of ASGARD.

13.1.4 Pointing direction comparison

Pointing error in degrees between two pointing directions (az_1, ze_1) and (az_2, ze_2) expressed as (azimuth, zenith):

$$\begin{cases} x_1 = \cos(az_1)\sin(ze_1) \\ y_1 = \sin(az_1)\sin(ze_1) \\ z_1 = \cos(ze_1) \end{cases}$$

$$\begin{cases} x_2 = \cos(az_2)\sin(ze_2) \\ y_2 = \sin(az_2)\sin(ze_2) \\ z_2 = \cos(ze_2) \end{cases}$$

$$d = \mathrm{acos}\,(x_1x_2 + y_1y_2 + z_1z_2)$$

This function (*pointing_error_azi_zen*) is available in file tests/helpers/compare.py of ASGARD.

13.1.5 Footprint comparison

In its vector shape, the footprint is a list of geolocations defining the perimeter of the footprint surface. The geolocations are not directly comparable, considering possible differences in the sampling used to distribute the geolocated points along the perimeter. The tool for footprint comparison shall use some surface measurements. A good criterium is to use the intersection between both footprints.

$$relativeDiff = \frac{surface(Footprint \cap FootprintRef)}{surface(Footprint \cup FootprintRef)}$$

On large surfaces, this criterium may not be sensitive enough on some points of the perimeter which largely derive from the reference footprint. A criterium on maximum distance (*dist*) between footprints will be added.

$$dist1 = maxi\ point\ in\ Footprint\big(dist(i, FootprintRef)\big)$$
$$dist2 = maxi\ point\ in\ FootprintRef\big(dist(i, Footprint)\big)$$
$$dist = \max(dist1, dist2)$$

The footprint handling has been implemented using python library shapely.

The distance tool used is the one described in § 13.1.2 with an altitude of 0.

This function (*footprint_comparison*) is available in file tests/helpers/compare.py of ASGARD.

13.2 Orbit position influence on incidence angle precision

Let define an orbit position vector (cartesian coordinates)

$$X = [5467124.62065836, 2666943.161324, 3810732.08810585]$$

Let define ground coordinates position

$$G = [136.96767074, 33.9208555, 36.01727374]$$

We can compute the norm

$$|X| = 7178002.263224866$$

Let define Y so that

$$|X| - |Y| = 1$$

i.e. $Y = \frac{|X| - 1}{|X|} \cdot X = [5467124.56977274, 2666943.13650125, 3810732.05263721]$

We compute pointing error in degrees between those two pointing directions from G position:

pointing_error $([\phi_X, \theta_X], [\phi_Y, \theta_Y]) \approx 10^{-6\circ}$

13.3 Additional information on runs

13.3.1 Sentinel-2 context configuration

13.3.1.1 Platforms

- ✓ Legacy run:
  - ➢ CPU: Intel(R) Xeon(R) CPU E5-2640 0 @ 2.50GHz
  - ➢ RAM: 64.520
  - ➢ 24 threads max (only on used)
- ✓ ASGARD run:
  - ➢ CPU: Intel(R) Xeon(R) Gold 5218 CPU @ 2.30GHz
  - ➢ RAM: 125Gi
  - ➢ 16 threads max (only one used)

13.3.1.2 Libraries used

Libraries as whl files used for S2MSI validation are:

- ✓ asgard-4.0-cp311-cp311-linux_x86_64.whl
- ✓ pyRugged-1.0.4.post11-cp311-cp311-linux_x86_64.whl
- ✓ sxgeo-0.1.11-cp311-cp311-linux_x86_64.whl

Based version of S2GEO modified to add logs is version 6.2.0.

13.3.1.3 DEM Used

DEM used are:

- ✓ Legacy:
  - ➢ Copernicus DEM_GEOID
  - ➢ Copernicus DEM_SRTM90
  - ➢ Copernicus DEM_GLOBE
- ✓ Zarr (only GETAS used finally):
  - ➢ S0__ADF_DEM90_20000101T000000_21000101T000000_20221028T100105.zarr
  - ➢ S0__ADF_GEOI8_20000101T000000_21000101T000000_20231207T105055.zarr
  - ➢ S0__ADF_GETAS_20000101T000000_21000101T000000_20230428T185052.zarr

13.3.2 Previous version launches

13.3.2.1 Sentinel-2 context configuration

Libraries as whl files used for S2MSI validation are:

- pyRugged-1.0.4.post5-cp310-cp310-linux_x86_64.whl
- sxgeo-0.1.9-cp310-cp310-linux_x86_64.whl

Based version of S2GEO modified to add logs is version 6.2.0.

13.3.2.2 Run ASGARD with legacy reference

Back to your Asgard environment terminal, run the `validate_sentinel2_msi.py` module again on your test data, but this time to run the Asgard processing and compare results with the reference data generated in the previous step:

```
# Validate the Asgard legacy processing (that uses the Java Rugged and SXGeo components):
python3 -m validate_sentinel2_msi \
    /path/to/input/S2GEO_Input_interface.xml \
    -r /path/to/s2geo_reference.txt \
    --legacy

# Validate the Asgard native processing, it is the same command line without the --legacy
option:
python3 -m validate_sentinel2_msi \
    /path/to/input/S2GEO_Input_interface.xml \
    -r /path/to/s2geo_reference.txt
```

/!\ DEM is read from the S2GEO_Input_interface.xml files, for now, they contains links to the Legacy DEM. Further validation shall create new S2GEO_Input_interface.xml with links to the Zarr DEM (cf Table 44 below).

Options are:
- -r: path to the reference file generated by Legacy processor
- -f: path to detector footprint folder (per detector/band) generated by Legacy processor (files named as follows DETFOO_DNN_BXX.gml, with NN in [01-12] and XX in [01-12] including 8A)
- --legacy: if configured, call the Legacy-based implementation, if not, call the Refactored implementation.
- -c: if configured, call a constant altitude with the altitude set just after (float)

The example command line to be derived for all TDS and levels is:

```
python3 -m validate_sentinel2_msi \
    /<path_to_TDS>/<TDS_Name>/<Config>_S2GEO_Input_interface.xml \
    -r /path/to/<Config>_s2geo_reference.txt (-f
/<path_to_TDS>/<TDS_Name>/<Config>_FOOTPRINTS) (--legacy) (-c <altitude>)
```

With:
- Path_to_TDS: the path where is located your TDS (will be downloaded from S3 in future versions)
- TDS_Name:

Table 43 – S2 TDS_Name correspondence

Test ref	TDS_Name to put in the command line
S2MSI_TDS1	S2PDGS-TC-IPF-V2-ORCH-N-016_CR136_Madeira_AllTiles_Batch2
S2MSI_TDS2	S2PDGS-TC-IPF-V2-ORCH-N-016_DEM30_TDS9-R029_6.3
S2MSI_TDS3	TBD
S2MSI_TDS4	TBD
S2MSI_TDS5	S2PDGS-TC-IPF-V2-ORCH-N-017_L0U_JUPITER_S2A_DT35
S2MSI_TDS6	TBD
S2MSI_TDS7	TBD
S2MSI_TDS8	TBD

- ✓ Config:

Table 44 – S2 configuration description

Configuration Name	description
L0u_DEM_Legacy	✓ L0u simplified sensor configuration ✓ Using GLOBE Legacy DEM
L0u_DEM_Zarr	✓ L0u simplified sensor configuration ✓ Using GLOBE Zarr DEM (TBC that this is available and to be used) => For future version
L0c_CONST	✓ L0 sensor configuration ✓ Using constant altitude (DEM has no impact but SRTM Legacy is still provided)
L0c_CONST_Zarr	✓ L0 sensor configuration ✓ Using constant altitude (DEM has no impact but Zarr is still provided) => For future version
L0c_DEM_Legacy	✓ L0 sensor configuration ✓ Using SRTM Legacy DEM
L0c_DEM_Zarr	✓ L0 sensor configuration ✓ Using Zarr DEM => For future versions
L1B_DEM_Legacy	✓ L1B sensor configuration ✓ Using SRTM Legacy DEM
L1B_DEM_Zarr	✓ L1B sensor configuration ✓ Using Zarr DEM => For future versions
L1C_DEM_Legacy	✓ L1C sensor configuration ✓ Using SRTM Legacy DEM
L1C_DEM_Zarr	✓ L1C sensor configuration

Configuration Name	description
	✓ Using Zarr DEM => For future versions
L1C_DEM30_Legacy	✓ L1C sensor configuration ✓ Using SRTM30 Legacy DEM => For future versions
L1C_DEM30_Zarr	✓ L1C sensor configuration ✓ Using Zarr DEM30 => For future versions

Example:

To call the Legacy-based implementation (*--legacy*), using the TDS1 (*S2PDGS-TC-IPF-V2-ORCH-N-016_CR136_Madeira_AllTiles_Batch2*) located in (*/home/jovyan/shared/ABE*), with the reference and footprints computation:

```
python3 -m validate_sentinel2_msi /home/jovyan/shared/ABE/S2PDGS-TC-IPF-V2-ORCH-N-
016_CR136_Madeira_AllTiles_Batch2/L0c_DEM_Legacy_S2GEO_Input_interface.xml -r
/home/jovyan/shared/ABE/S2PDGS-TC-IPF-V2-ORCH-N-
016_CR136_Madeira_AllTiles_Batch2/L0c_DEM_Legacy_s2geo_reference.txt -f
/home/jovyan/shared/ABE/S2PDGS-TC-IPF-V2-ORCH-N-
016_CR136_Madeira_AllTiles_Batch2/L0c_DEM_Legacy_FOOTPRINTS --legacy
```

13.3.3 Sentinel-3 context configuration

13.3.3.1 Legacy based

The following libraries versions are to be installed:

- orekit_jcc-11.1-cp310-cp310-linux_x86_64.whl
- asgard-3.2.dev26+g030349e-cp310-cp310-linux_x86_64.whl
- pyRugged-1.0.4.post9-cp310-cp310-linux_x86_64.whl
- sxgeo-0.1.9-cp310-cp310-linux_x86_64.whl

13.3.3.2 Refactored

TBD

Changes

Issue	Date	Change description
1.0	12/22/2023	Issuing of document
1.1	02/05/2024	Validation report of ASGARD V4.1