S2 MPC

Sen2Cor Configuration and User Manual

Ref. S2-PDGS-MPC-L2A-SUM-V2.8
Issue 2

The power of innovation

This Page is intentionally Blank

Authors Table

	Name	Company	Responsibility	Date	Signature
Written by	U. Mueller-Wilm	TPZV	Project Manager L2A Maintenance	2019-02-05	
Verified by	O Devignot	CS	Quality Manager	2019-02-05	
Approved by	L. Pessiot	CS	Service Manager	2019-02-05	

Change Log

Issue	Date	Reason for change	Pages(s)/Section(s)
1.0	2.July 2012	Created	All
1.1	15 September 2012	Issue after S2PAD Phase 2 CDR. Updated according to ESA comments and discussion on CDR 02/08/2012	All
1.2	1.June 2013	Adaptation of installation procedures according to unification of environments and pre-release of processor for QR	2.4
1.3	31. March 2014	New Section 2.2, updated Installation procedures for Windows, moved information for data IO into new created document [S2-PDGS-MPC-L2A-IODD]	2.2
1.4	27. June 2014	Restructuring of sections 3.1, 3.2 to align the installation after upgrades of Anaconda and GDAL.	3.1.3
2.0	15.May 2015	Integration into Sentinel-2 Toolbox, Version 2.0 Complete Improvement of Installation Procedure: Integration of CONDA Packages for GDAL and GLYMUR Upgrade to PSD V12 Upgrade of JPEG-2000 Readers to OpenJPEG 2.1.0 instead of Jasper Fixes of SPRs according to Release note for Version 2.0	3.1
2.1	10.02.2016	Parallelisation on tile base implemented Upgrade to PSD V13.1	2.5
2.2	13.04.2016	Integration of Look-Up-Tables Automated Aerosol determination Automated Ozone selection New description for DEM selection Various improvements for command line Handling	2.2.2, 2.2.7, 3.3.3

Issue	Date	Reason for change	Pages(s)/Section(s)
2.3	25.11.2016	Corrected command line options according to SIIMPC-686 Updated installation procedure in section 3.1 Added new configuration items in section 3.3.3. Added offset for DEM output	2.2.7, 3.1, 3.3.3
2.4	30.06.2017	Removed old installation procedure, now referring to release note Added copyright notes	3.1,5
2.5	15.11.2017	New section for Sen2Cor 2.5.0 configuration updates	3.2.1
2.6	08.02.2018	New section for Sen2Cor 2.6.0 operation mode	3.3.1
2.6.1	26.02.2018	Updated command line description and removed constraints for Toolbox mode	3.3.1
		Removed section on old PDGS interface as was implemented for Sen2Cor 2.1 due to new Tile mode	N/A
		Added insertions and comments from J. Louis after internal document review	All sections
2.6.2 – 2.6.6	24.07.2018	Improved description on automated aerosol type determination.	2.2.2.2.2
		Updated description of the processing routines for band resampling.	2.4.2 – 2.4.6
		Removed section on parallelisation of tile processing, due to single tile inputs. Changed information on memory requirements due to optimisation.	2.5
		Inserted information for standard SAFE or SAFE compact.	3.3, 3.3.1, 3.3.2
		Added / changed information for raw data selection and resolution	3.3.1, 3.3.2
2.7.0	21.09.2018	Added new command line parameters for database locations	2.4.1
		New parameter for Database Compression factor	2.4.1
		Split image database into two entities and optimized disk usage	2.4.1
		New parameter for Disabling Terrain Correction with DEM	0
2.7.1	11.10.2018	Adding hints for location of databases	2.4.1
2.7.2	14.12.2018	Adding the two new command line parameters for processing_centre and processing_baseline Removed the unused parameter refresh	3.3.2

Issue	Date	Reason for change	Pages(s)/Section(s)
2.8.0	16.01.2019	Added description for multithreading with OpenJPEG and removed description for multi-tile processing. Updated version numbers for reference documents.	2.4.1 2.5.1
	05.02.2019	Changed sections according to RIDs from ESA, differentiating between Toolbox, PDGS and common configuration and command-line options.	3.2, 3.3, 3.4
		Added section 3.2.5	3.2.5

Table of contents

List of Figures

List of Tables

2. Introduction

1.1 Purpose of the document

This document is the Software *Installation and User Manual (SUM)* for the Sentinel-2 Level-2A Prototype Processor, which is labelled ***Sen2Cor*** for ***Sen***tinel ***2*** (atmospheric) ***Cor***rection.

The prototype implementation for the Level 2A processing of Sentinel-2 imagery over land is a combination of state-of-the-art techniques for performing Atmospheric Corrections (AC, including Cirrus clouds and terrain correction), tailored to the Sentinel-2 environment and a Scene Classification (SC) module.

Sen2Cor performs a pre-processing of Level-1C (L1C) Top of Atmosphere (TOA) image data, and applies a scene classification, an atmospheric correction and a subsequent conversion into an ortho-image Level-2A Bottom-Of-Atmosphere (BOA) reflectance product. Outputs are an Aerosol Optical Thickness (AOT) map, a Water Vapour (WV) map and a Scene Classification map together with Quality Indicators data. Details of the products and its contents is provided in [L2A-PDD] of section 1.3.

Level-2A (L2A) products are re-sampled as L1C products with a constant GSD (Ground Sampling Distance) of 10m, 20m and 60 m according to the native resolution of the different spectral bands. If applicable, Level-2A products are provided for each MSI channel at coarser resolution (i.e. 20 m and 60 m) as well.

A large database of look-up tables (LUTs) has been compiled using an atmospheric radiative transfer model based on LibRadtran[1]. The LUTs are generated for a wide variety of atmospheric conditions, solar geometries, and ground elevations and are calculated with a high spectral resolution of 0.6 nm. This database has been subsequently resampled with the Sentinel-2 spectral responses, in order to obtain the sensor-specific functions needed for the atmospheric correction.

1.2 Document structure

The configuration and user manual consists of the following chapters and sections:

Chapter, Section	Description
1	this chapter
2	Introduces the Sen2Cor system. What is the general purpose of the application, how is it structured, what are the processing schemes. It lists the general system architecture modules and functionality and gives a brief overview on its functionality and operation.
3	Introduces the two different environments (runtime and development) of the system in general and the source code distribution.

[1] https://www.libradtran.org

3.1	How to install the processor software and the according runtime environment.
3.2	How to configure the processor software and the according runtime environment.
3.3	How to run the processor software in the according runtime environment.
3.4	Describes the setup configuration and operation of the software development environment, set up on top of Eclipse and PyDev. Contains also a detailed overview on all used third party tools.

With these sections the configuration and user manual enables developers to upgrade and maintain the software and users of the software to operate the system within their specific hardware environment.

What this document will not provide is the scientific background of the application. This is part of the corresponding ATBD [S2-PDGS-MPC-L2A-ATBD], see below. Also the configuration of the processor has been moved into [S2-PDGS-MPC-L2A-IODD]. In order to avoid redundancies and inconsistencies between the different project documents, this content will thus not be repeated here. If it is necessary for the understanding of the operation, this SUM will refer to the according chapters of [S2-PDGS-MPC-L2A-IODD], [S2-PDD] and [S2-PDGS-MPC-L2A-DPM].

1.3 References

Document ID	Description	Version
S2-PDGS-MPC-L2A-PFS	Sentinel-2 MSI – Product Format Specification	14.5
S2-PDGS-MPC-L2A-IODD	Sentinel-2 MSI – Level 2A Prototype Processor Input Output Definition Document	2.8.0
S2-PDGS-MPC-L2A-ATBD	Sentinel-2 MSI - Level 2A Products, Algorithm Theoretical Basis Document	2.1
S2-PDGS-MPC-L2A-DPM	Sentinel-2 MSI – Level 2A Detailed Processing Model	1.0
S2-PDD	GMES Space Component – Sentinel-2 Payload Data Ground Segment (PDGS), Product Definition Document	2.5
S2-PSD	Sentinel-2 Products Specification Document	14.5

3. Functionality and Operation

Sen2Cor is a prototype processor for Sentinel-2 Level 2A product formatting and processing. The processor performs the tasks of atmospheric-, terrain and cirrus correction and a SC of Level 1C input data. Level 2A outputs are: Bottom-Of-Atmosphere (BOA), optionally terrain- and cirrus corrected reflectance images, AOT-, WV-, SC maps and Quality Indicators, including cloud and snow probabilities. The Level 2A Product Formatting performed by the processor follows the specification of the Level 1C User Product. Details are given in [S2-PDGS-MPC-L2A-PFS].

The Sentinel-2 Multi-Spectral Instrument (MSI) consists of 13 spectral bands with three different resolutions (10m, 20m and 60m) as shown in Figure 3-1. The instrument covers a 290 km swath. The Level-1C image product, which serves as the input for the Level-2A processing consists of a series of n tiles, each with a 100 km square. Each tile consists of thirteen compressed JPEG-2000 images, each image representing one single band. The thirteen bands have three different resolutions (10m, 20m and 60m) which lead to different image dimensions of the Level-1C input product. These details are given in [L2A-PDD] and [L2A-IODD].

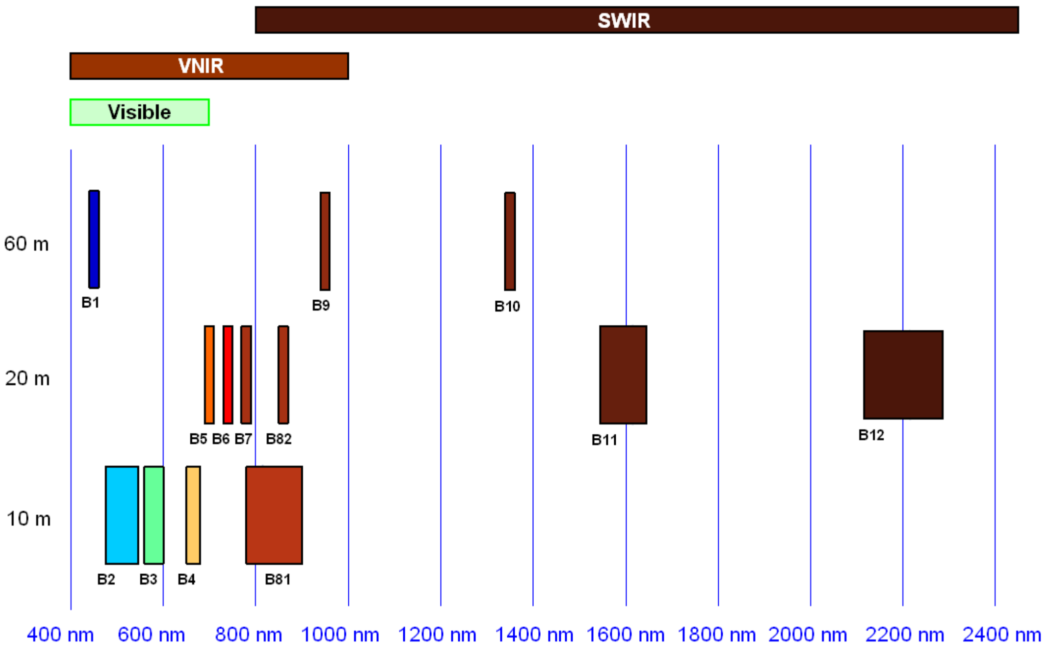

Figure 3-1 – Sentinel-2 Spectral Bands and Resolutions

3.1 Level-2A Processor Architecture

It follows a high level description of the processor architecture:

The Sen2Cor application is designed by the following ten essential modules (classes):

1. **L2A_Process:** the general operator module, which coordinates the interaction between the other modules and creates the skeleton L2A product structure of the metadata.

2. **L2A_ProcessTile:** a single processing module, executing the tasks of scene classification, atmospheric correction and the creation of metadata on tile base.
3. **L2A_SceneClass:** performs the coarse classification of the input images into their different contents like clouds, snow, water, soil etc. and provides statistical analysis.
4. **L2A_AtmCorr:** transforms the input from top of atmosphere (TOA) to bottom of atmosphere (BOA) representation and performs the atmospheric correction of the input.
5. **L2A_Config:** a helper class providing the configuration parameters to all other modules listed above.
6. **L2A_Tables:** a helper class, managing the conversion of the JPEG-2000 based input data to an internal format (and vice versa) and providing a high performance access to the data for the processing modules (see section 4.3). It uses its own private L2A_Config instance.
7. **L2A_Manifest:** a class specialized for the generation of the manifest on product level.
8. **L2A_XmlParser:** a utility class for parsing the metadata and GIPP files on demand.
9. **L2A_Library:** a collection of common tools used by all classes on demand.

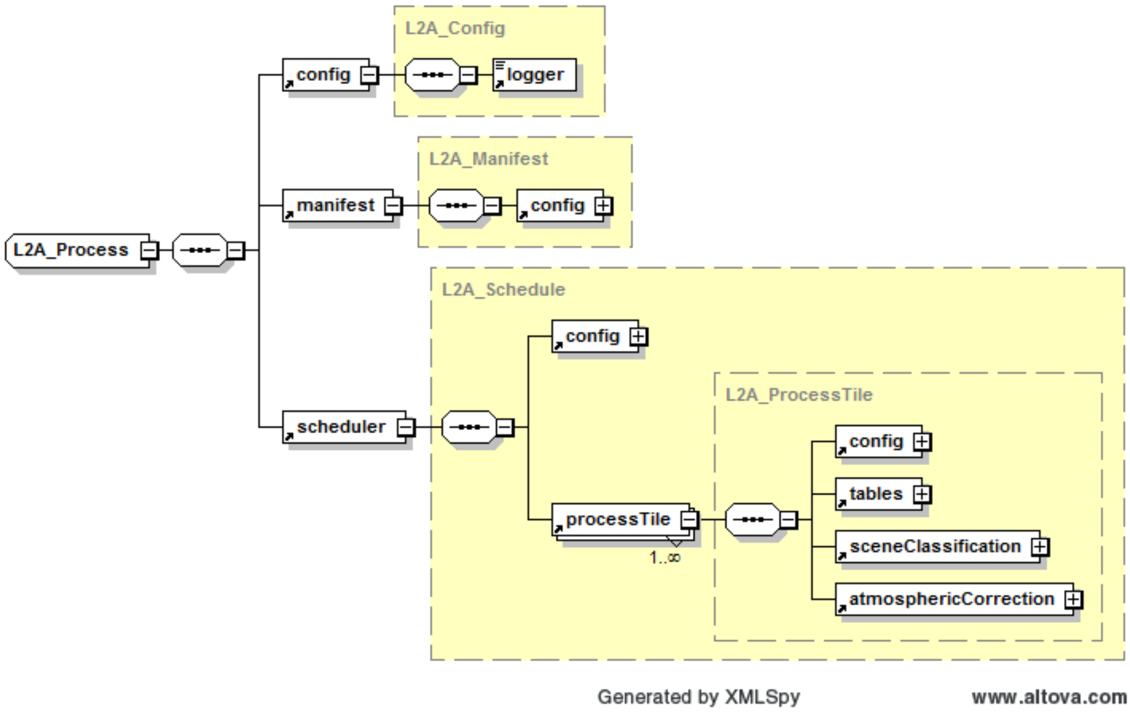

Figure 3-2 – High Level Processor Architecture

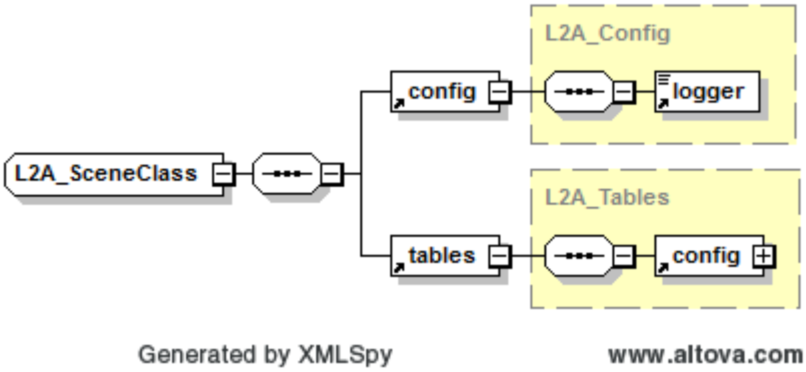

Figure 3-3 – Scene Classification Module

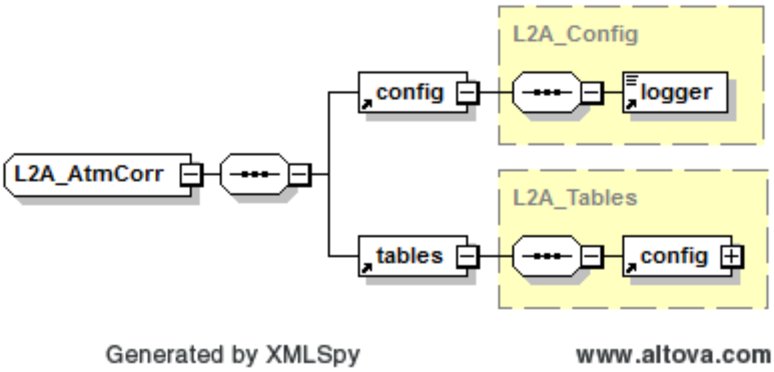

Figure 3-4 – Atmospheric Correction Module

3.2 Workflow

Figure 3-5 below shows the main processing workflow. After reading and processing the input parameter and data the main processing module triggers the creation of an internal temporary database, which is then used by the SC and the AC module to retrieve and to store the data and intermediate products. The processing can act in a loop, dependent on the number of different product resolutions to be generated.

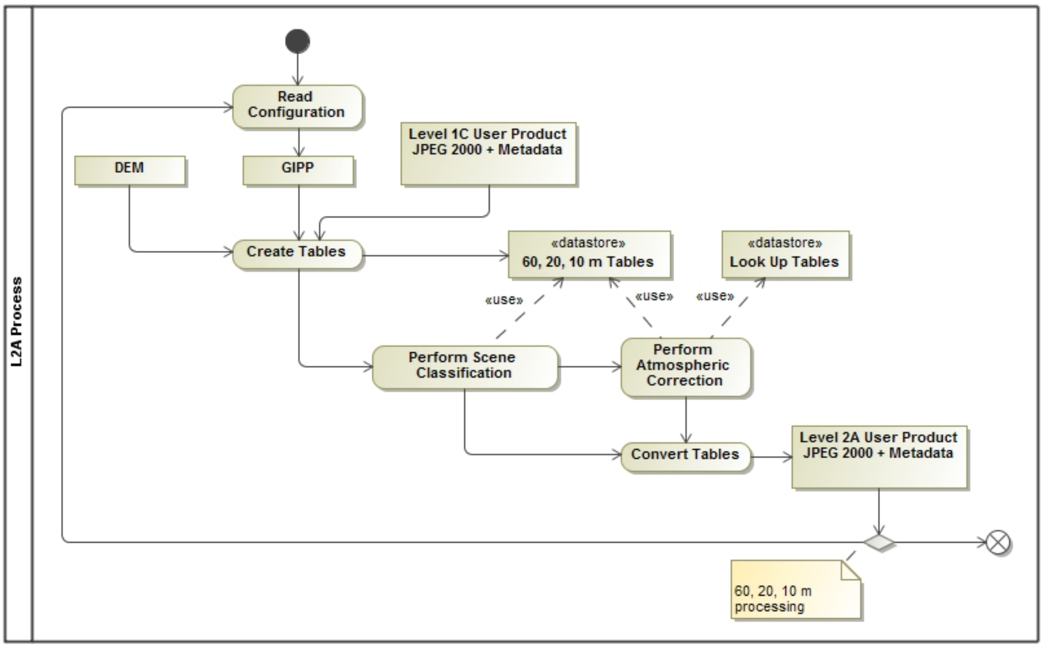

Figure 3-5 – Processing Flow, Overview

3.2.1 Scene Classification (L2A_SceneClass)

The SC algorithm allows to detect clouds, snow and cloud shadows and to generate a classification map, which consists of 4 different classes for clouds (including cirrus), together with six different classifications for shadows, cloud shadows, vegetation, soils / deserts, water and snow. The algorithm is based on a series of threshold tests that use as input top-of-atmosphere reflectance from the Sentinel-2 spectral bands. In addition, thresholds are applied on band ratios and indexes like the Normalized Difference Vegetation - and Snow Index (NDVI, NDSI [3]). For each of these thresholds tests, a level of confidence is associated. At the end of the processing chain a probabilistic cloud mask quality map and a snow mask quality map is produced. The algorithm uses the reflective properties of scene features to establish the presence or absence of clouds in a scene. Cloud screening is applied to the data in order to retrieve accurate atmospheric and surface parameters, either as input for the further processing steps below or for being valuable input for processing steps of higher levels. Figure 3-6 below shows the results of a SC (right side) based on modified AVIRIS test data (left side). Twelve different classifications are provided.

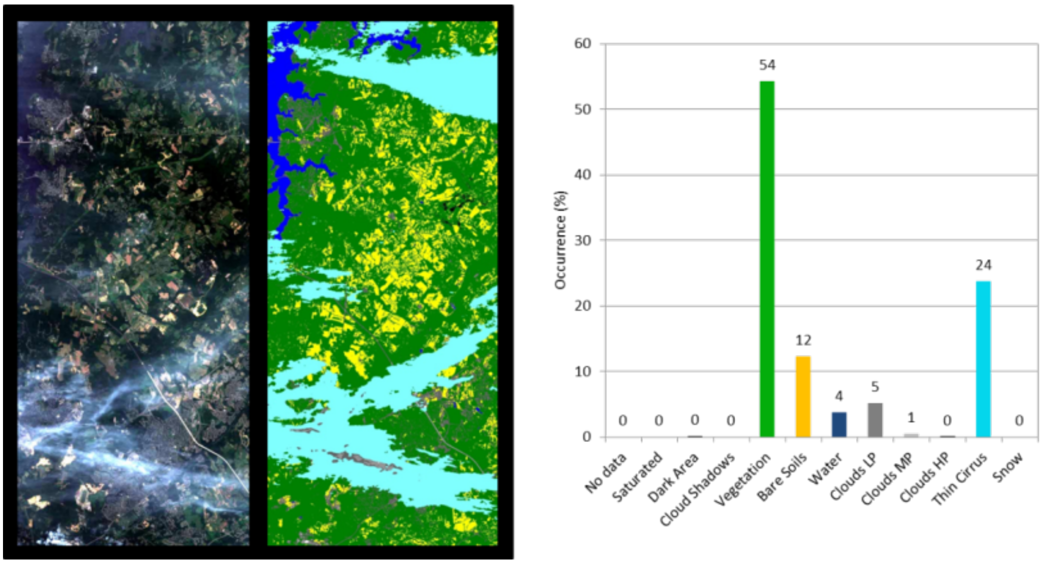

Figure 3-6 – Scene Classification

The generated classification map is specified as follows:

Table 3-1 – Classification Map

Label	Classification
0	NO_DATA
1	SATURATED_OR_DEFECTIVE
2	DARK_AREA_PIXELS
3	CLOUD_SHADOWS
4	VEGETATION
5	NOT_VEGETATED
6	WATER
7	UNCLASSIFIED
8	CLOUD_MEDIUM_PROBABILITY
9	CLOUD_HIGH_PROBABILITY
10	THIN_CIRRUS
11	SNOW

Associated quality indicators on snow and cloud probability are generated from the results. These Quality indicators calculate the probability (0-100%) that the earth surface is obstructed by clouds or optically thick aerosol (ice or snow).

The SC processing consists of six different steps:

1. Snow detection;
2. Cloud detection;
3. Cirrus detection;

4. Cloud shadow detection;
5. Classification map generation.

- The processing is shown in the flow diagram below.

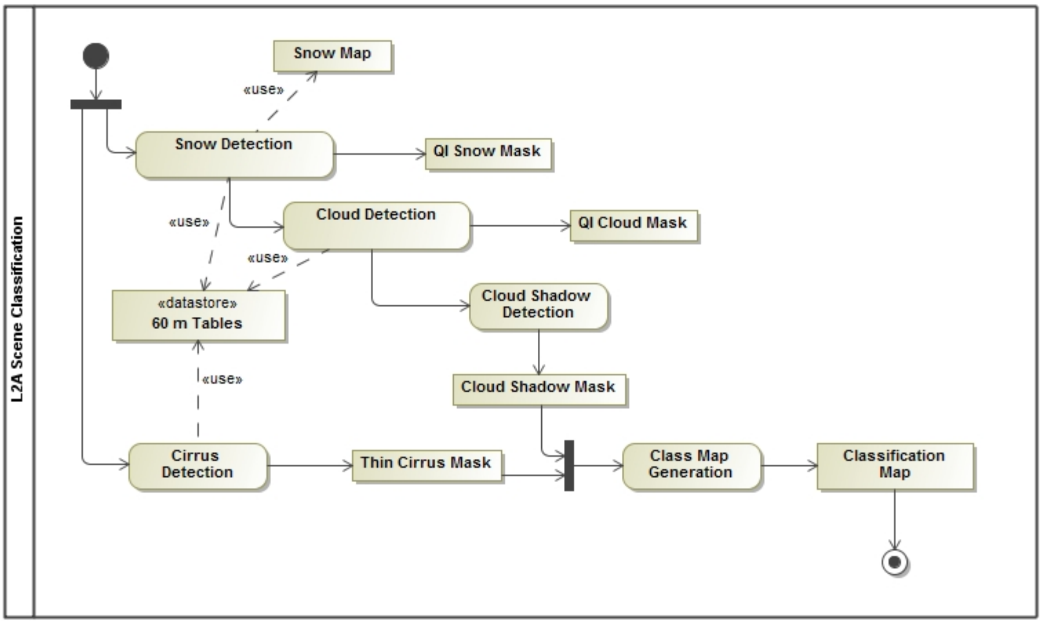

Figure 3-7 – Scene Classification, Processing Flow

Details of this algorithm, especially on the different threshold conditions are given in chapter 3 of [S2-PDGS-MPC-L2A-ATBD] and chapter 2.5 of [L2A-DPM].

3.2.2 Atmospheric Correction (L2A_AtmCorr)

The AC processing consists of a set of four different subtasks, (AOT, WV and terrain retrieval (optional with terrain and cirrus correction, having three different user products as output: AOT and WV tables on pixel level and the BOA corrected reflectance images for all bands measured. Figure 3-8 below shows the processing flow for the atmospheric correction module.

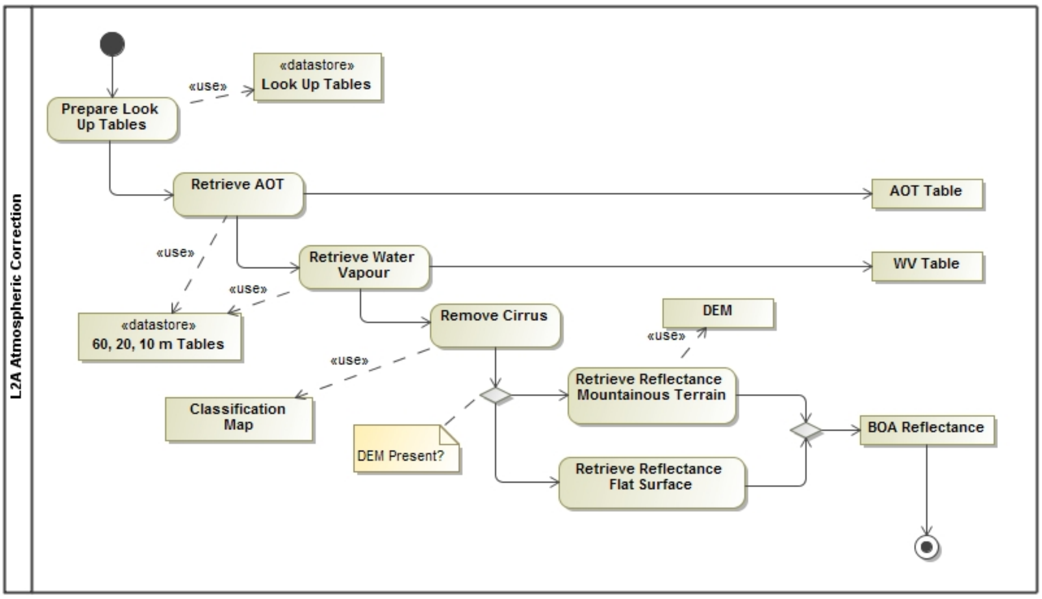

Figure 3-8 – Atmospheric Correction, Processing Flow

3.2.2.1 Look-Up-Tables Generation

The atmospheric model of Sen2Cor (L2A_AtmCorr) is dependent on the calculation of radiative transfer functions for different sensor and solar geometries, ground elevations, and atmospheric parameters. The following list presents a 6-dimensional parameter space and the grid spacing for each parameter. The processor reads the parameter in form of Look-Up-Tables (LUTs) pertaining to this parameter space and interpolates, if required. The LUTs have been generated via LibRadtran, a library for the calculation of solar and thermal radiation in the Earth's atmosphere.

Table 3-2 – Parameter space for atmospheric correction

Parameter	Range	Increment / grid points
Solar zenith angle	0 -70°	10°
Sensor view angle	0 -10°	10°
Relative azimuth angle	0 -180°	30°(180°= backscatter)
Ground elevation	0 -2.5 km	0.5 km
Visibility	5 -120 km	5, 7, 10, 15, 23, 40, 80, 120 km
Water vapour, summer	0.4 -5.5 cm	0.4, 1.0, 2.0, 2.9, 4.0, 5.0 cm
Water vapour, winter	0.2 -1.5 cm	0.2, 0.4, 0.8, 1.1 cm

Starting with Sen2Cor release 2.2.0 the user can select between four atmospheric models: a set of 24 LUTs has been integrated to cover most of atmospheric conditions on Earth for the Sentinel-2 mission. In the Sen2Cor context, a set of LUTs is composed by 6 or 4 LUT files depending on the total water vapour columns content of the atmosphere. Different LUTs are calculated for the mid-latitude summer and mid-latitude winter atmospheres, with 6 different (sea level) ozone contents, a rural and a maritime aerosol, 6 or 4 different sea level water vapour columns. For each supported water vapour level, the ground-to- space water vapour column depending on elevation according to the atmosphere temperature / humidity vertical profile is provided.

Sen2Cor LUTs are calculated for:

- 2 different types of aerosols (rural and maritime)
- 2 different types of atmospheres (mid latitude summer and mid latitude winter)
- 6 different types of ozone concentrations (depending on summer or winter case)
- 6 or 4 different amounts of water vapour column (depending on summer or winter)

3.2.2.2 User configuration

The LUT selection is configurable via the user configuration file (L2A_GIPP.xml) located inside the cfg folder of the directory where the $SEN2COR_HOME environment variable points to. In the Look_Up_Tables selection of the configuration file, three entries: Aerosol_Type, Mid_Latitude and Ozone_Content can be set. The water vapour columns are set internally.

Default processing via configuration is the rural (continental) aerosol type with mid latitude summer and an ozone concentration of 331 Dobson Units.

Please refer to [S2-PDGS-MPC-L2A-IODD] for details.

3.2.2.2.1 Setting of Automated Ozone Input

If the Ozone_Content is set to '0' by the user, it will be determined automatically by the processor. In that case, the measured ozone concentration is read from the L1C input product (located in the AUX_DATA folder of each tile) and the LUT with the best fit for the measured ozone concentration is used. Other parameters possible are referenced in the configuration file.

3.2.2.2.2 Setting Aerosol type (RURAL, MARITIME, AUTO)

If the Aerosol_Type is set to 'AUTO' by the user, it will be determined automatically by the processor. In that case, the processor will process two (aerosol only) test trials before the final processing of the atmospheric correction takes place.

The scene path radiance in the blue and red region is calculated as the total, minus the reflected radiance, using the average values obtained for the dark reference pixels. After calculation of the scene path radiance, the ratio of the path radiance for the blue channel scene compared to the red channel scene is compared to the corresponding ratio of the existing aerosols (RURAL, MARITIME) from the look up tables.

The aerosol type for which the double ratio (dp) is closest to 1 is the best approximation for the scene and will be selected and used in all subsequent measures for the corresponding tile.

As the derived aerosol type is constant per scene in the Sen2Cor model, the question arises whether the automatic selection is the best choice, especially if one considers neighbouring scenes. In this case, the aerosol types could switch, leading to steps in the surface reflectance at the image borders. From this point of view a pre-selected aerosol type (e.g. rural-continental) might be the better choice in practice (see S2-PDGS-MPC-L2A-ATBD).

3.2.2.2.3 Setting Atmosphere (AUTO, SUMMER, WINTER)

If the Mid_Latitude is set to 'AUTO' by the user, it will be determined automatically by the processor. In that case the processor will select WINTER or SUMMER atmosphere profile based on the acquisition date and geographic location of the tile.

3.2.3 Aerosol Optical Thickness

AOT retrieval provides a measure for the visual transparency of the atmosphere. It is derived using the DDV (Dense Dark Vegetation) algorithm [5], using the short wave infrared (SWIR) band 12 and correlates its reflectance with bands 4 (red) and 2 (blue). The algorithm requires that the scene contains reference areas of known reflectance behaviour, preferably Dark Dense Vegetation (DDV) and/or dark soil and water bodies.

The algorithm starts with a user-defined visibility (default: 40 km) as input. If the scene contains no dark vegetation or soil pixels, the surface reflectance threshold of band 12 will be successively iterated in order to include medium brightness reference pixels in the sample. If the scene contains no reference and no water pixels the scene is processed with the start visibility instead. The algorithm delivers an AOT map as shown in Figure 3-9 below.

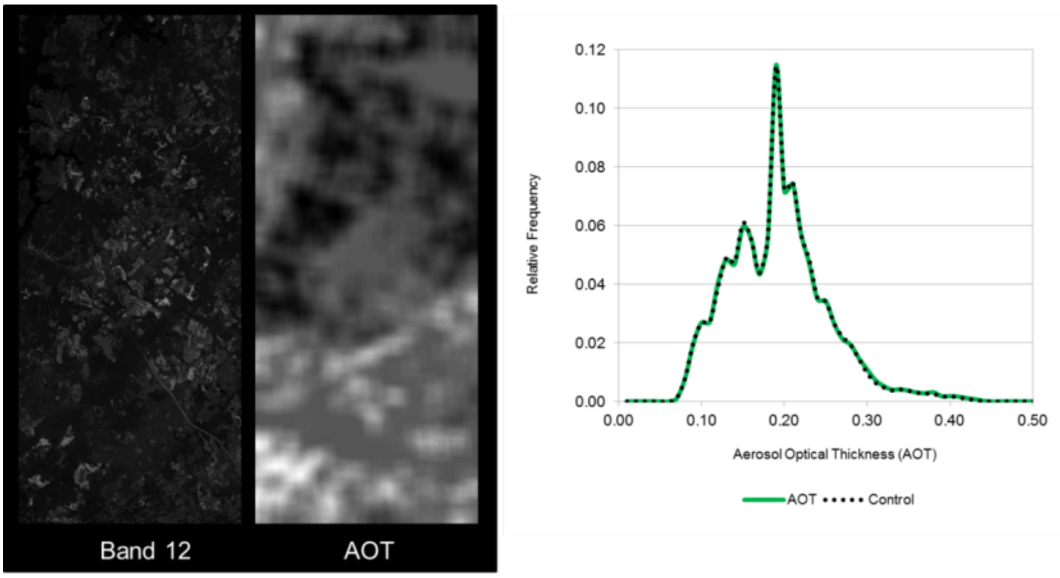

Figure 3-9 – AOT Retrieval using Band 12

3.2.4 Water Vapour Retrieval

WV retrieval over land is performed with the Atmospheric Pre-corrected Differential Absorption algorithm (APDA, [6]) which is applied to the two Sentinel-2 bands B8a, and B9 (Figure 3-10). Band 8a is the reference channel in an atmospheric window region. Band B9 is the measurement channel in the absorption region. The absorption depth is evaluated by calculating the radiance for an atmosphere with no WV, assuming that the surface reflectance for the measurement channel is the same as for the reference channel. The absorption depth is then a measure of the WV column content.

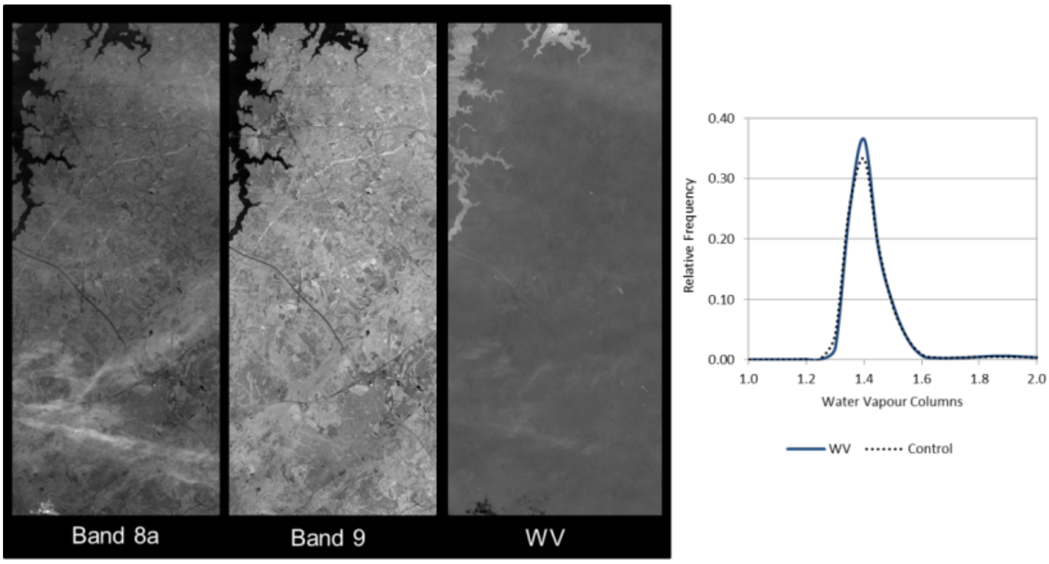

Figure 3-10 – WV Retrieval using Bands 8a and 9

3.2.5 Cirrus Correction

The Cirrus Correction algorithm uses the sentinel (cirrus) channel 10. Thin cirrus clouds affect the visible, near- and shortwave infrared spectral regions. They are partially transparent and thus difficult to detect with broad-band multispectral sensors, especially over spatially inhomogeneous land areas.

WV, in contrast, dominates in the lower troposphere of 0-5 km. A narrow spectral band in a spectral region of very strong WV absorption (Band 10) will thus absorb the ground reflected signal, but will receive the scattered cirrus signal.

Cirrus reflectance of band 10 can therefore be correlated with other bands in the VNIR and SWIR region and the cirrus contribution can thus be removed from the radiance signal to obtain a cirrus-corrected scene. This is shown in Figure 3-11 below as a qualitative result.

Very good results can be achieved locally for a great visual impact however it is not activated by default because on a global scale the results of cirrus correction cannot be always optimal. The choice is let to the user to activate it. Please note that cirrus correction is only applied during 20 m and 60 m processing. It is not applied at 10 m.

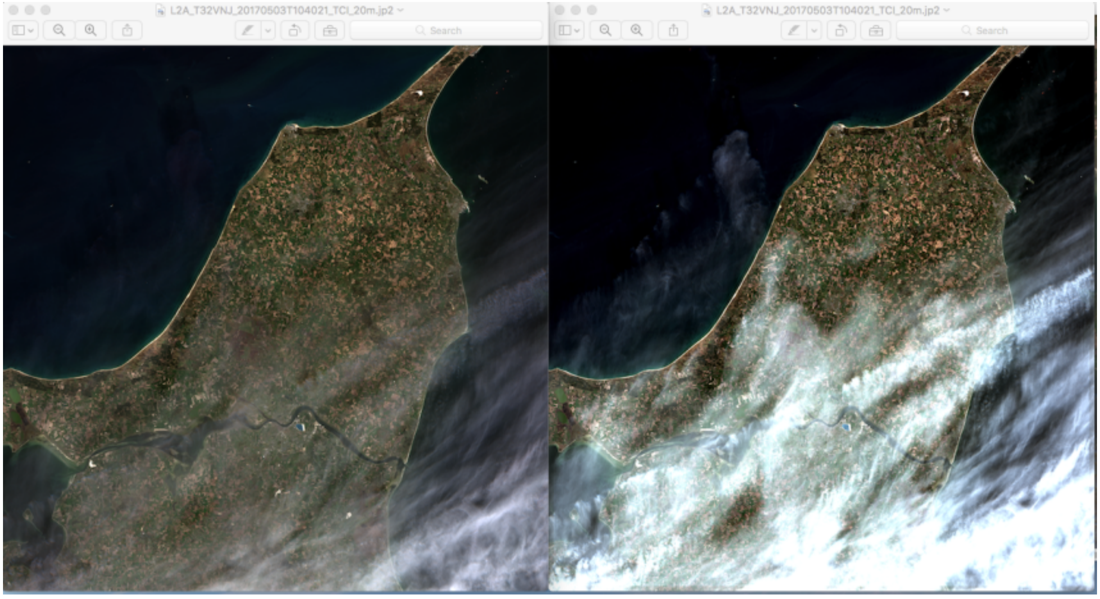

Figure 3-11 – Cirrus Correction over north Denmark, Bands 2-4 with Band 10. Left: L2A Output cirrus corrected, right: without correction.

3.2.6 Surface Reflectance Retrieval

Surface Reflectance retrieval is performed for each sequential Band B1 – B12. Figure 2-9 below shows the Level 1C input data and the corresponding Level 2A output after atmospheric correction from a scene of La Paz, retrieved on 28.03.2016.

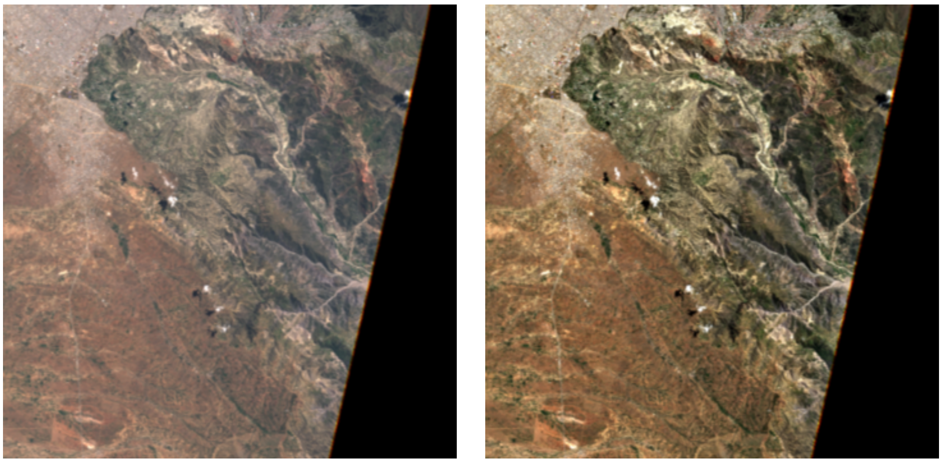

Figure 3-12 – Left: Level 1C Input, Bands 2-4; right: Level 2A Output, Bands 2-4, RGB composite images, scene from La Paz on

3.2.7 Usage of Digital Elevation Maps

Since the release of Sen2Cor 2.2, Digital Elevation Maps can be used for two purposes:

1. Improvement of the scene classification: previous versions of the processor had a tendency of false classification of water pixels inside of cloud borders and the correct discrimination between topographic and cloud shadow pixels. This was improved by taking the height information of an (optional) digital elevation map (DEM) as an additional input. To use this feature it is necessary to activate the reading of an appropriate DEM as is explained below.
2. Improvement of the terrain correction for rugged terrains: the algorithm for rugged terrain requires the existence of an appropriate Digital Elevation Map (DEM) and a set of derived products like slope, aspect and terrain shadow maps. The retrieval of the water vapour map also includes this terrain elevation.

Sen2Cor is currently prepared to make use of two different DEM's. The first one is the **90m SRTM Digital Elevation Database from CGIAR-CSI**. Before you decide to make use of this database, please read carefully the Disclaimer given on:

http://www.cgiar-csi.org/data/srtm-90m-digital-elevation-database-v4-1

The second supported format is the commercial 90m DTED-1 Format from **PlanetDEM** (used for L1C processing in Sentinel-2 ground segment):

http://www.planetobserver.com/products/planetdem/planetdem-90/

The Planet DEM is not downloadable for free and must be purchased. The processor expects all the appropriate DEM files with the form: ***eXXX_nYY.dt1*** in the specified local folder and uses them, if available.

The usage of DEM is not set by default. It has to be set manually. **The user can specify a DEM in the L2A_GIPP.xml configuration file** by setting the parameter for <DEM_Directory> to a relative path, instead of NONE. The DEM shall be located in the specified folder in the Sen2Cor home directory.

Example: <DEM_Directory>dem/srtm</DEM_Directory>

If no appropriate DEM is found in this folder, the processor can try to retrieve DEM files from CGIAR-CSI: the processor will start automatically to download the DEM's from the database that has to be referenced by:

<DEM_Reference>

http://data_public:GDdci@data.cgiar-csi.org/srtm/tiles/GeoTIFF/

</DEM_Reference>

If an already downloaded DEM can be found in the local folder referenced by DEM_Directory, the download will not take place and the local archive is used instead. If the download fails and no local DEM can be found, the processor continues with a flat surface calculation.

The geo-coordinates and angular information are obtained from the Level-1C metadata. The area of interest is created and fitted to the input images according to the information retrieved from the metadata and the conversion is performed using the GDAL libraries from OSGEO.

The algorithm reformats, resamples and assembles the DEMs based on the geo locations obtained either from the Level-1C metadata or from the JPEG-2000 images. Finally, it creates the according slope, aspect and a hill shadow maps using the GDAL dem tools.

The retrieval of the spectral reflectance cube consists of the following steps:

1. iterations for terrain reflectance;
2. (optional) empirical BRDF correction;
3. adjacency correction;
4. spherical albedo correction.

Figure 3-13 below shows a comparison of the Level 2A output without usage of a DEM and the corresponding Level 2A output when a 90 m CGIAR DEM is applied before terrain correction. The same scene as for Figure 3-12 is shown.

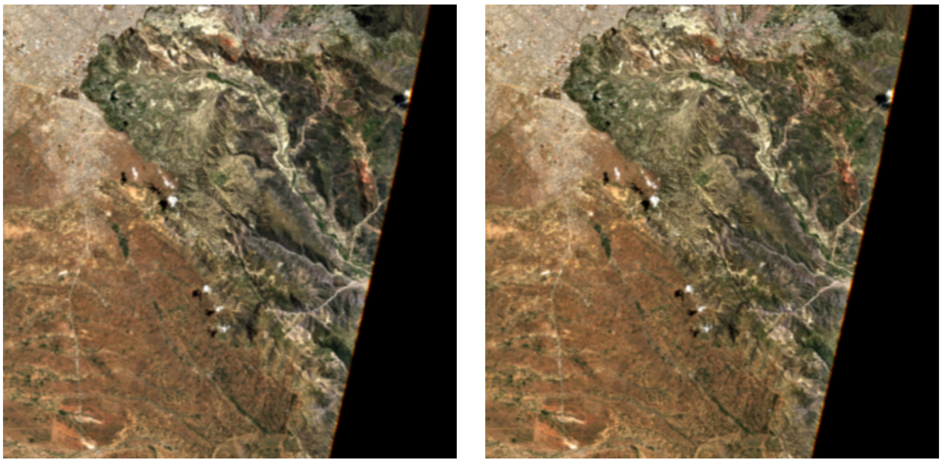

Figure 3-13 – Terrain correction; Left: Level 1C Input, Bands 2-4, RGB composite image, middle: DEM, reshaped to according scene, right: Level 2A Output, Bands 2-4, RGB composite image.

Since Sen2Cor V.2.3.0 (PFS 14.2) an offset of 10.000 is added to the DN output Value of the DEM array in order to support negative heights, as the OpenJPEG format only supports unsigned integer values. In practice, the highest elevation value on Earth: ~8,848 m and lowest elevation value on dry land: ~ -418m are thus encoded as 18848 and 9582 DN.

New since Sen2Cor 2.7.0: to decouple the DEM processing between scene classification, AOT and terrain correction a new configuration parameter has been inserted in the L2A_GIPP.xml. Setting <DEM_Terrain_Correction>to 'FALSE' will disable the terrain iteration for the terrain corrections, whereas scene classification, AOT (and WVP) are still calculated using the DEM information.

3.3 Product Formatting

The Level 2A Product format is closely related to the Level 1C Top-of-Atmosphere (TOA) reflectance product, which serves as an input to the processor. It consists of 13 JPEG-2000 images, associated to the 13 Sentinel-2 spectral bands at three different spatial resolutions with a ground sampling distance of 10, 20, and 60m.

The generated Level 2A BOA reflectance output images are resampled and generated with an equal spatial resolution for all bands, based on three user selectable resolutions of 10, 20 and / or 60m.

The tile level contains different components, based on the resolution selected by the user:

- a 10 m resolution product contains spectral bands 2, 3, 4 and 8 and an AOT map resampled from 20m
- a 20 m product contains band 2 – 7, the bands 8a, 11 and 12 and an AOT and WV map
- a 60m product contains all components of the 20m product and additionally the 60m bands 1 and 9
- Additionally, starting with Sen2Cor 2.3.0 a True Colour composite Image (TCI) using Bands 2-4 with the same resolution, as the monochrome bands will be stored in parallel to the reflectance output images.
- The Cirrus band 10 will be omitted in the Level 2A output, as it does not contain surface information. Figure 3-14 shows the Level 2A user product on tile level.

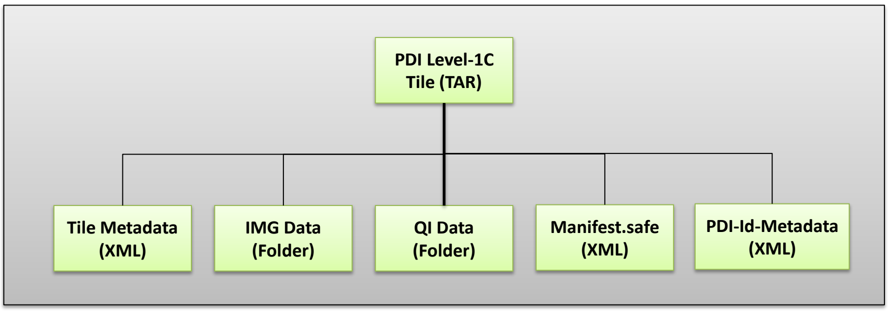

Figure 3-14 – Filing structure of Level-1C product on tile level

The complete specifications for all inputs are provided in [S2-PDGS-MPC-L2A-PFS] and [S2-PDGS-MPC-L2A-IODD]. Level-2A Inputs are derived from the Level-1C data schemes as described in [S2-PFS].

3.4 Memory Requirements

The atmospheric correction processing for 10m resolutions uses a huge amount of memory due to the 10.000 x 10.000 pixel for each image. Multiple images must be kept at certain intervals completely in memory for performing correlations. Thus, for processing at 10m resolution, as a general rule of thumb, a resource of **4 GB of memory** should be available at least.

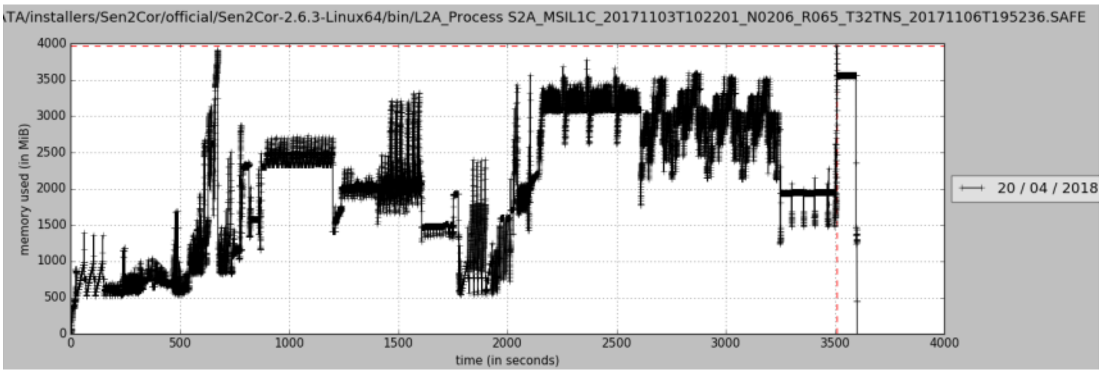

Figure 2-15 Memory Consumption during processing

3.5 Interface changes:

Configuration: the L2A_GIPP.xml has been rearranged. Only user specific configuration parameters are presented. Expert configuration parameters have been moved into 2 configuration files, which are located under the site-package directory of the Sen2Cor application.

L2A_CAL_SC_GIPP.xml: contains the calibration parameter for the scene classification, which should only be changed by expert users. Thus, this file is no member of the user configuration.

L2A_CAL_AC_GIPP.xml: contains the calibration parameter for the atmospheric correction, which should only be changed by expert users. Thus, this file is no member of the user configuration.

Starting with Version 2.3.0 the following four changes in the configuration have been added.

Configuration Item	Description	Value
Scaling_Limiter	Limits the scaling of the path radiance for the blue channel to +/-10%.	true (default)/ false
Scaling_Disabler	Disables the scaling of the path radiance for the blue channel.	true (default) / false
Rho_Retrieval_Step2	Disables the execution of step 2 of the	true (default) / false

As these configuration items are of experimental character they are located in the expert configuration file L2A_CAL_AC_GIPP.xml and should not be changed by standard users.

In the user configuration file L2A_GIPP.xml (located in the cfg directory of the directory where $SEN2COR_HOME points to) the default value for the visibility has been changed from 23 to 40 km.

Configuration Item	**Description**	**Value**
Visibility	Sets the visibility default value	40.0 (default)

3.5.1 Other Changes:

As new L2A user product will always be created with the current timestamp in UTC for the creation time.

- A selected resolution of 60 m will only process the 60 m resolution.
- A selected resolution of 20 m will only process the 20 m resolution.
- A selected resolution of 10 m will process the 20 m resolution first, as it is the base for the 10 m product.
- If the option <Downsample_20_to_60> in the L2A_GIPP.xml is set to TRUE, a 60 m product will be generated via a down sampling of the 20 m product. This is configured by default and can be changed in the L2A_GIPP.xml.

3.6 Logging (Logger)

The module L2A_Config keeps an instance of a logger object. Each of the processing modules writes its own diagnosis and status messages into a common log-file, located in the HTML folder at the top of the User product.

The log level can be configured in the GIPP (L2A_GIPP.xml) as:

`DEBUG, INFO, WARNING, ERROR, CRITICAL, NOTSET`

The log-level is hierarchical: if e.g. set to ***DEBUG***, all higher messages such as ***INFO***, ***WARNING***, ***ERROR*** or ***CRITICAL*** will be displayed as well. Setting the log-level to ***ERROR*** displays only ***ERROR*** and ***CRITICAL*** messages.

Beneath the message itself, the logger displays the system time stamp when the message was generated, the log-level, the module which has generated the message and the function and code line of the module which has generated the message.

Since Version 2.6.0 the formatter of the logging output has been aligned to the Level 1C processor (only for PDGS mode).

4. Configuration and Installation

4.1 Installation and Setup

Installation and Setup of the runtime environment is specific on the actual releases of Sen2Cor and is described in the according release note under Section 4.

4.1.1 ESACCI-LC package for SC module

The ESACCI-LC for Sen2Cor data package is prepared for users of Sen2Cor version ≥ 2.5 who want to benefit from the last improvements of Sen2Cor Cloud Screening and Classification module. This auxiliary data information is used in Sen2Cor to improve the accuracy of Sen2Cor classification over water, urban and bare areas and also to have a better handling of false detection of snow pixels.

Please note that a Digital Elevation Model (DEM) is a pre-requisite for using ESACCI_LC information in the Scene Classification algorithm. Please refer to section 3.2.7 to use an external DEM. In the case SRTM DEM is used, latitudes higher > 60 deg N (and lower < 56 deg S) are not covered by the SRTM DEM, therefore no ESACCI_LC information will be used for these latitudes. Standard Scene Classification algorithm will then be applied.

Users of Sen2Cor version ≥ 2.5 should download this ESACCI-LC for Sen2Cor data package (ESACCI-LC-L4-ALL-FOR-SEN2COR.zip) from this location: ***http://maps.elie.ucl.ac.be/CCI/viewer/download.php***

This zip file shall then be extracted at this location of Sen2Cor installation: '***$SEN2COR_BIN/aux_data/.***' It contains two files and one directory:

- ***ESACCI-LC-L4-LCCS-Map-300m-P1Y-2015-v2.0.7.tif***
- ***ESACCI-LC-L4-WB-Map-150m-P13Y-2000-v4.0.tif***
- ***ESACCI-LC-L4-Snow-Cond-500m-P13Y7D-2000-2012-v2.0***

Example on Ubuntu (Linux) installation:

$ ls Sen2Cor-02.08.00-Linux64/lib/python2.7/site-packages/sen2cor/aux_data

ESACCI-LC-L4-LCCS-Map-300m-P1Y-2015-v2.0.7.tif

ESACCI-LC-L4-Snow-Cond-500m-P13Y7D-2000-2012-v2.0

ESACCI-LC-L4-WB-Map-150m-P13Y-2000-v4.0.tif

Example on Windows7 installation:

>dir Sen2Cor-02.08.00-Linux64/lib/python2.7/site-packages/sen2cor/aux_data

ESACCI-LC-L4-LCCS-Map-300m-P1Y-2015-v2.0.7.tif

ESACCI-LC-L4-Snow-Cond-500m-P13Y7D-2000-2012-v2.0

ESACCI-LC-L4-WB-Map-150m-P13Y-2000-v4.0.tif

Note: Please note that it is possible to use symbolic links in this aux_data folder if you prefer to copy those auxiliary files to another data folder. (unix command: ln –s, windows command: mklink)

4.2 Product Processing

Due to the three different resolutions of the Level-1C input images, conversion routines will serve for an up- and down-sampling of parts of the images into the appropriate resolution as well as the transfer from JPEG-2000 into a different internal format (see below). This has been implemented using the OpenJPEG library, which will be installed during the Sen2Cor setup. Due to the relative huge size of the image data (especially for the high-resolution 10 bands B02 – B04 and B08) data in- and output has been shown as rather time and memory consuming.

Since Sen2Cor 2.6.0, the routines for generating user products of different resolutions have been decoupled and generally improved. Sen2Cor 2.8.0, uses OpenJPEG, version 2.3 for the reading of the L1C input images. This allows the usage of multithreading to speed up the import. The configuration is set by default to AUTO in the L2A_GIPP.xml configuration file (see [S2-PDGS-MPC-L2A-IODD]), and Section 4.3.1 below.

Sen2Cor will now read in all required bands only once; at begin of the processing in their original resolution. For each subsequent processing, the bands will be internally resampled and written to the target resolution.

An intermediate data conversion was selected which converts the JPEG-2000 images and the necessary metadata (like the DEM) into an internal format based on HDF5. HDF5 is a data model, library, and file format for storing and managing data. It is especially designed for flexible and efficient I/O and for high volume and complex data. HDF5 is portable and extensible, allowing applications to evolve in their use and platforms to be supported.

The interfacing with the internal data format is implemented via PyTables. PyTables is a package for managing hierarchical datasets, designed to efficiently cope with extremely large amounts of data. It is built on top of the HDF5 library, the Python language and the NumPy package. PyTables and HDF5 are described in section 4.5.1.6

Operations with previous Sen2Cor Versions (below 2.7) have shown, that the HDF5 database could grow up to more than 5 GB for a full tile, using DEMs as additional auxiliary data. To downscale the consumption of disk space the following modifications have been implemented with 2.7.0:

- There is now a persistent image database containing only the L1C source files in their default resolutions. This database will be first removed after a full processing of all resolutions. Additionally, a temporary database is filled with the resampled and generated products, which will be removed after each processed resolution in order to keep the space of the databases to the lowest limit possible.
- In the first processing stage, the image database will load only the bands needed for the 20 m processing (except 10 m Band 8). After the 20m processing is performed, all bands (except the three 10m bands 1-3) are removed and Band 8 will be loaded. Thus, the database will keep a size below 1 GB during the whole processing.
- The temporary database will import all auxiliary data (like the DEMs and ESACCI images). It will additionally store the resampled bands from the image database and the processed products from the 20 m Scene Classification (Classification Map) and Atmospheric correction (AOT/Visibility and Water Vapor). After the 20 m processing is performed, only these products will be up-sampled to 10 m and copied into a new

temporary database. The old database with all products no longer needed will then be removed.

- In the 10 m processing the temporary database keeps the AOT/Visibility and Water vapor and imports the new resampled auxiliary data, needed for the 10m processing.
- The down-sampling routine to 60 m has also improved in that way that the output products will be directly converted to their final format, so that no further storage of these down-sampled products in the database is required.
- Additionally, all bands no longer needed in the processing chain are removed from the temporary database, to free space for other intermediate products.

With this, the following measures have been obtained by operation of a full sized L1C tile with configured DEM, processing for 20m and 10m and a down sampling of the 20m product to 60m.

Resolution	Image DB	Temp DB
20 m processing	739 MB	1.53 GB
Down sampling to 60	No changes in DB size	No changes in DB size
10 m processing	867 MB	1.963 GB

4.2.1 The 60 m Product Processing

The 60m product processing uses the three 60m bands B01, B09 and B10 and a 60m DEM as inputs. The three 10m bands B02-B04 will be internally down-sampled to 60m. The same is true for the 20m bands B05 – B8A and B11, B12. The 10m band B08 is not used for the 60m processing.

Beneath the twelve optical channels (the cirrus channel B10 is excluded as it does not represent surface information) the 60m product processing generates:

1. a Scene Classification map (SCL);
2. quality indicators for presence of snow and clouds (in percentage);
3. an Atmospheric Optical Thickness (AOT):
4. a Water Vapour map (WVP);
5. a preview image, covering the three visible bands 2-4 within an JPEG-2 compressed image, at 320m resolution.

4.2.2 The 20 m Product Processing

The 20m product processing uses the 20m bands B05 – B8A and B11, B12 and a 20m DEM as inputs. The three 60m bands B01, B09 and B10 will internally be up-sampled to 20m, the three 10m bands B02-B04 will be internally down-sampled to 20m. The 10m band B08 is not used for the 20m processing.

The 20m product processing covers nine optical channels a 20m AOT and a WV and a Visibility Index file corresponding to the AOT. The three resampled 60m bands B01, B09 and B10 are omitted in the output in order to avoid spectral artefacts due to mixed signatures and resampling.

4.2.3 The 10 m Product Processing

Inputs for the 10m product processing are the four 10m bands B02-B04 and B08, an optional 10m DEM an up-sampled Scene Classification and a Visibility Index map, both up-sampled from 20m. As the WV influence is very small, only the scene-average WV needs to be used for the surface reflectance retrieval. The 10m product processing covers thus only four optical channels. The other channels, not used for the calculation will be omitted. Always, a 20 m resolution needs to be performed first, as the 10m processing depends on these input data.

4.2.4 Post-Processing

The OpenJPEG Library is used for generating the final L2A-Product, transferring the internal HDF5 based tables back into the JPEG-2000 format. It keeps for all generated products the final (resampled) resolution.

The output product will be placed in the folder addressed via the option "--output_dir". If this option is omitted, the output product will be placed in the same folder as the Level-1C input product.

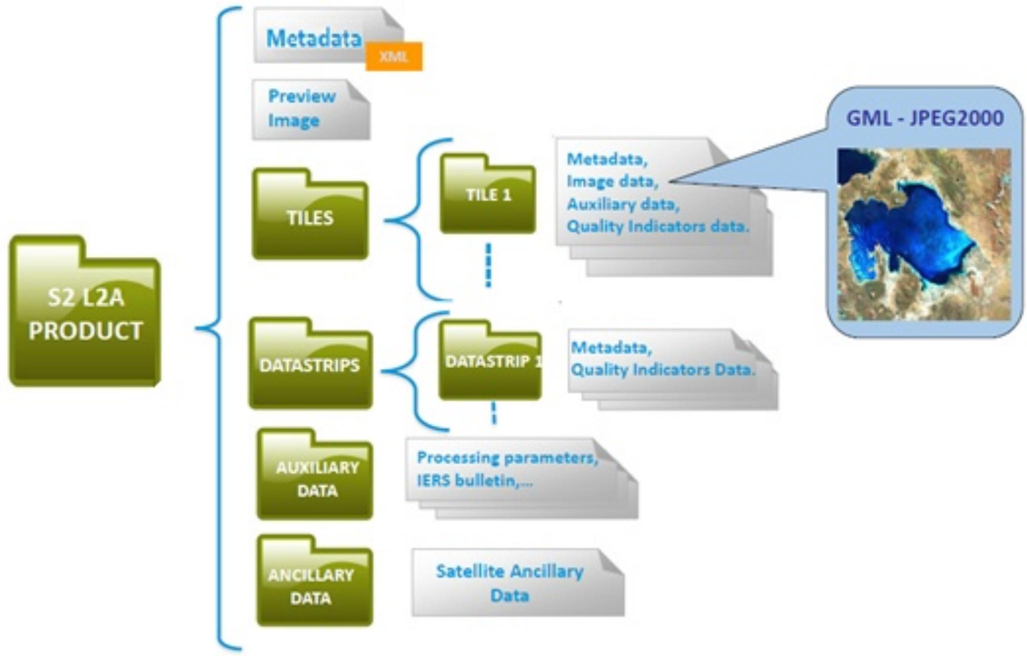

Figure 4-1 – Level 2A product, physical format Configuration (L2A_Config)

4.2.5 Differences between DHUS and Toolbox generated L2A User Products

The HTML folder is not present in the Toolbox generated L2A user product.

The manifest is generated differently for L2A products on the Data Hub Server in comparison to the Toolbox generated user product.

Both issues will be aligned in future releases.

4.3 Configuration

4.3.1 Common Configuration

It is important to know, that the default configuration of the L2A algorithm provided with Sen2Cor version 02.08.00 is the same configuration as for the operational L2A processing in Sentinel-2 Ground Segment (S2PDGS) with L2A processing baseline 02.12. Therefore, no further tuning should be required on user side to get L2A "standard" products as generated by S2PDGS. It is important to highlight that in order to generate products compatible with the L2A products generated by the S2PDGS it is necessary to set the usage of a Digital Elevation Model and have the activation of ESA-CCI.

The only differences between "Toolbox" and "standard" L2A products should either come from the difference in the DEM used (e.g. no DEM, SRTM of PlanetDEM) or from the JPEG2000 encoding, OpenJPEG in Toolbox and Kakadu in "standard" products.

The so-called Ground Image Processing Parameter (GIPP) are configured in an XML file named L2A_GIPP.xml (see example in the appendix). The following new Settings have been implemented since release of Sen2Cor 2.5.5:

- Sen2Cor 2.8.0, uses OpenJPEG, version 2.3 for the reading of the L1C input images. This allows the usage of multithreading to speed up the import. The configuration is set by default to AUTO, which detects the amount of usable threads by calling cpu_count(). If the user does not want this feature or want to set the amount of threads individually, the parameter <Nr_Threads> can be changed between values of 1 (single thread processing) up to 8. Figure 4-2 below shows the speed improvement for reading on a 2 Core Intel I5 platform with 8 GB of RAM between OpenJPEG 2.1 (left) and (new) OpenJPEG 2.3 with one, two and four threads applied.

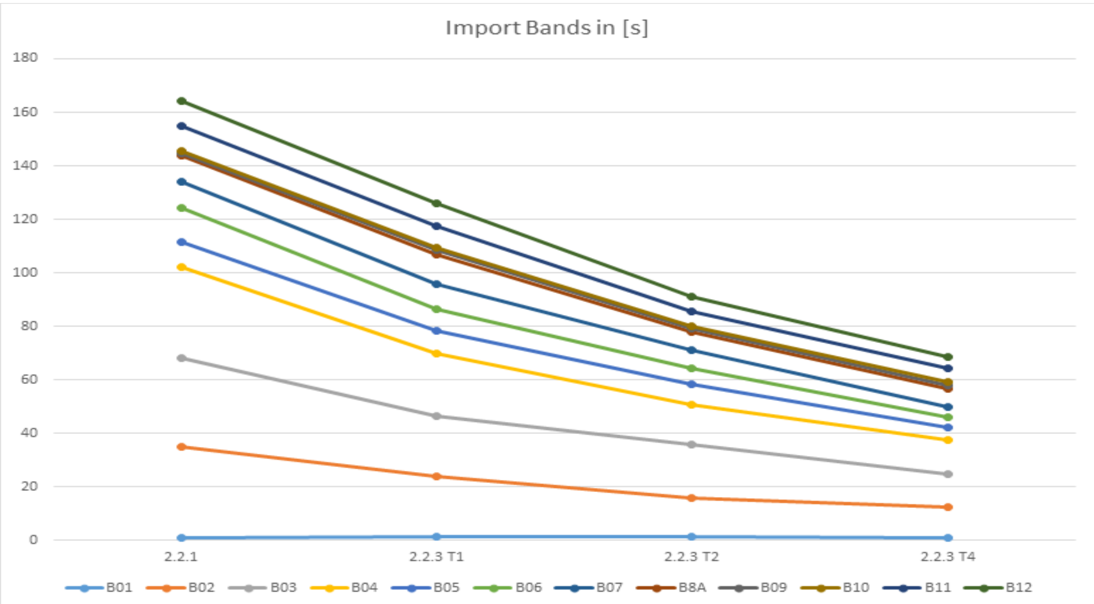

Figure 4-2 – Performance improvement using OpenJPEG 2.3 and multithreading

- Starting with Version 2.6.5 only a 20m and a 10m processing will take place by default. The user however can configure an optional down sampling to a 60m resolution, after a 20m processing has been performed. This is configured via the entry <Downsample_20_to_60> in the L2A_GIPP configuration file.
- Three options are provided in the L2A_GIPP.xml file in order to activate or deactivate the generation of three optional output products:

```
<Generate_DEM_Output>FALSE</Generate_DEM_Output>
FALSE: no DEM output, TRUE: store DEM in the AUX data directory
<Generate_TCI_Output>TRUE</Generate_TCI_Output>
FALSE: no TCI output, TRUE: store TCI in the IMAGE data directory
<Generate_DDV_Output>FALSE</Generate_DDV_Output>2
FALSE: no DDV output, TRUE: store DDV in the QI_DATA directory
```

- The Ozone selection in for the Atmospheric correction is now based on value instead of a single character: the input of a letter for selection of an ozone value (plus profile) in the L2A_GIPP.xml input has been replaced with a direct input of an ozone value. The algorithm chooses the ozone column closest to the ozone value as specified as input. For the Ozone content settings the following options are available:
  - 0: to get the best approximation from the L1C metadata (this is the smallest difference between the metadata and column DU).
  - Otherwise, the parameter can be set to a fixed value of:
  - 250, 290, 331 (standard MS), 370, 410, 450 for midlatitude summer (MS) atmosphere, or:
  - 250, 290, 330, 377 (standard MW), 420, 460 for midlatitude winter (MW) atmosphere.

  This can be configured in the L2A_GIPP.xml:

```
<Ozone_Content>0</Ozone_Content>
<!-- The atmospheric temperature profile and ozone content in Dobson Unit
(DU) -->
```

All other parameters are inherited from previous Sen2Cor Versions are described in the corresponding [S2-PDGS-MPC-L2A-IODD].

---

2 The DDV is not an official supported product and is only of interest for expert testing purposes of the algorithm for Aerosol Optical Thickness.

4.3.2 Toolbox Related Configuration

The Configuration file L2A_GIPP.xml is located in the <cfg> subdirectory of the processor (see example in the appendix). Its parameters, together with all input, output and other auxiliary data are listed in the corresponding [S2-PDGS-MPC-L2A-IODD] document and thus not repeated here.

Located under ***$SEN2COR_HOME/cfg*** the GIPP file of the application can be configured by the user for individual purposes. If a different configuration shall be used, '***$SEN2COR_HOME***' directory can be reconfigured to a different directory:

export SEN2COR_HOME' = <directory of your choice>

This allows the operation with multiple configuration settings.

There is now also a configurable compression factor available in the User configuration file L2A_GIPP.xml (<Database_Compression_Factor>, see [S2-PDGS-MPC-L2A-IODD] for details). A compression factor of 1, allows a reduction of the databases to about 2/3 of their original size in the non-compressed mode. For TOOLBOX users the proposed default setting is no compression (compression factor of 0) to get a better speed performance. If disk space is an issue, the user can increase this compression factor.

4.3.3 PDGS Related Configuration

The location of the configuration file L2A_GIPP.xml is specified via command line. Its parameters, together with all input, output and other auxiliary data are listed in the corresponding [S2-PDGS-MPC-L2A-IODD] document and thus not repeated here.

The Level-1C input data is expected to be present in a folder structure at product level, as specified in section 3.1 of [L2A-PDD]. The location of the Level-1C input product can be specified via the command line argument.

There is now also a configurable compression factor available in the User configuration file L2A_GIPP.xml (<Database_Compression_Factor>, see [S2-PDGS-MPC-L2A-IODD] for details). It is recommended to use a compression factor of 1, which allows a reduction of the databases to about 2/3 of their original size in the non-compressed mode.

4.4 Operation

The processor can be operated in three different ways:

- either as a purely command line driven application;
- or from the Sentinel-2 Toolbox;
- or as a command line driven application inside the PDGS.

Important! Due to the PDGS extensions Sen2Cor Version 2.8.0 does no longer support PSD Versions below 14.2, e.g. Multi-tile L1C products.

4.4.1 Toolbox Related Command Line Options

Calling the script L2A_Process with the option '-h' via command-line displays the following menu:

Command Line Parameters
("Toolbox" mode using L1C product as input_dir)

```
L2A_Process --help
usage: L2A_Process.py [-h] [--resolution {10,20,60}]
                      [--output_dir OUTPUT_DIR]
                      [--processing_centre PROCESSING_CENTRE]
                      [--processing_baseline PROCESSING_BASELINE]
                      [--tif] [--sc_only] [--cr_only]
                      [--GIP_L2A GIP_L2A] [--GIP_L2A_SC GIP_L2A_SC]
                      [--GIP_L2A_AC GIP_L2A_AC] input_dir

positional arguments:
  input_dir             Directory of Level-1C input

optional arguments:
  -h, --help            show this help message and exit
  --resolution {10,20,60}
                        Target resolution, can be 10, 20 or 60m. If omitted,
                        only 20 and 10m resolutions will be processed
  --output_dir OUTPUT_DIR
                        Output directory (a full path) for Level 2A product. If not set,
                        The product will be located in the same directory as the input
                        Product
  --processing_centre
                        Processing centre as regular expression, 4 Characters or '_',
                        like "SGS_".
  --processing_baseline
                        Processing baseline in the format "dd.dd", where d=[0:9], like
                        "08.15".
  --tif                 Export raw images in TIFF format instead of JPEG-2000
  --sc_only             Performs only the scene classification at 60 or 20m
                        resolution
  --cr_only             Performs only the creation of the L2A product tree, no
                        processing
  --GIP_L2A GIP_L2A     Select the user GIPP (a filename with full path)
  --GIP_L2A_SC GIP_L2A_SC
                        Select the scene classification GIPP (a filename with full path)
  --GIP_L2A_AC GIP_L2A_AC
                        Select the atmospheric correction GIPP (a filename with full path)
  --GIP_L2A_PB GIP_L2A_PB
                        Select the processing baseline GIPP (a filename with full path)
```

Usage:

The <input_dir> argument is **mandatory** in Toolbox mode and can be either a relative or an absolute pathname. Input is expected to be a L1C product directory in compact SAFE format. Output will also be formatted as compact SAFE.

Older versions (L1C products with Processing Baseline <= N0204) having the standard SAFE format are no longer supported.

If a relative pathname is given, it is expected that the user is calling Sen2Cor from inside a parent directory. Sen2cor will expand the absolute pathname for that directory.

The pathname shall point to a L1C user product.

Sen2cor will use the L1C user product identifier for generating a subsequent L2A product. For this purpose, the L1C source directory must start with an identifier like **'S2[A|B]_*L1C*'** which is the standard, if you download a L1C user product from the Sentinel 2 data hub. The generated product will get the identifier **'S2[A|B]_*L2A*'**, and the current timestamp in UTC format. Everything else will be inherited from the L1C source.

The output directory for the L2A data selected by the command line, giving an absolute path for the target directory. By default, the processor will create the output product in the same directory where the L1C user product is created, but replacing “L1C” with “L2A” and updates the timestamp for the generation time.

On L1C User Product Level:

Absolute path:

```
L2A_Process
</the_L1C_product_directory>/<the_L1C_product_in_short_naming_convention>
```

Relative Path (the command must be called from inside the user product directory):

```
L2A_Process <the_L1C_product_in_short_naming_convention>
```

4.4.2 PDGS Related Command Line Options

The following command line features are available in the so called PDGS mode:

- Support of Datastrip generation mode with L1C input Datastrip;
- Support of Tile processing mode with L1C input Tile;
- L1C Datastrip and Tiles can have either standard SAFE or SAFE compact format as input;
- L2A Datastrip and Tile are formatted in standard SAFE format for output.

Calling the script L2A_Process with the option ‘-h’ via command line displays the following menu:

Command Line Parameters ("PDGS" mode processing L1C Datastrip and L1C Tile in sequence)

```
L2A_Process --help
usage: L2A_Process.py [-h] [--mode MODE] [--resolution {10,20,60}]
                      [--datastrip DATASTRIP] [--tile TILE]
                      [--output_dir OUTPUT_DIR] [--work_dir WORK_DIR]
                      [--img_database_dir IMG_DATABASE_DIR]
                      [--res_database_dir RES_DATABASE_DIR]
                      [--processing_centre PROCESSING_CENTRE]
                      [--archiving_centre ARCHIVING_CENTRE]
                      [--processing_baseline PROCESSING_BASELINE] [--raw]
                      [--tif] [--sc_only] [--cr_only] [--debug]
                      [--GIP_L2A GIP_L2A] [--GIP_L2A_SC GIP_L2A_SC]
                      [--GIP_L2A_AC GIP_L2A_AC] [--GIP_L2A_PB GIP_L2A_PB]
                      input_dir

Sentinel-2 Level 2A Processor (Sen2Cor). Version: 2.8.0, created: 2019.01.20, supporting
Level-1C product version 14.2 - 14.5.

positional arguments:
  input_dir             Directory of Level-1C input

optional arguments:
  -h, --help            show this help message and exit
  --mode MODE           Mode: generate_datastrip, process_tile
  --resolution {10,20,60}
                        Target resolution, can be 10, 20 or 60m. If omitted,
                        only 20 and 10m resolutions will be processed
  --datastrip DATASTRIP
                        Datastrip folder
  --tile TILE           Tile folder
  --output_dir OUTPUT_DIR
                        Output directory
  --work_dir WORK_DIR   Work directory
  --img_database_dir IMG_DATABASE_DIR
                        Database directory for L1C input images
  --res_database_dir RES_DATABASE_DIR
                        Database directory for results and temporary products
  --processing_centre PROCESSING_CENTRE
                        Processing centre as regex: ^[A-Z_]{4}$, e.g "SGS_"
  --archiving_centre ARCHIVING_CENTRE
                        Archiving centre as regex: ^[A-Z_]{4}$, e.g. "SGS_"
  --processing_baseline PROCESSING_BASELINE
                        Processing baseline in the format: "dd.dd", where
                        d=[0:9]
  --raw                 Export raw images in rawl format with ENVI hdr
  --tif                 Export raw images in TIFF format instead of JPEG-2000
  --sc_only             Performs only the scene classification at 60 or 20m
                        resolution
  --cr_only             Performs only the creation of the L2A product tree, no
                        processing
  --debug               Performs in debug mode
  --GIP_L2A GIP_L2A     Select the user GIPP
  --GIP_L2A_SC GIP_L2A_SC
                        Select the scene classification GIPP
  --GIP_L2A_AC GIP_L2A_AC
                        Select the atmospheric correction GIPP
  --GIP_L2A_PB GIP_L2A_PB
                        Select the processing baseline GIPP
```

PDGS mode for GENERATE_DATASTRIP

Example for command line:

```
L2A_Process.py
        --mode=generate_datastrip
        --datastrip=<DATASTRIP_EXAMPLE_DIRECTORY>/
S2A_OPER_MSI_L1C_DS_SGS__20171129T040150_S20171129T023320_N02.06
        --work_dir=<WORK_EXAMPLE_DIRECTORY> /temp
        --output_dir=<OUTPUT_EXAMPLE_DIRECTORY>/output
        --processing_centre=EDRS
        --archiving_centre=SGS_
        --GIP_L2A_PB=<GIP_EXAMPLE_DIRECTORY>/L2A_PB_GIPP.xml
```

PDGS mode for PROCESS_TILE

Example for command line:

```
L2A_Process.py
        --mode=process_tile
        --datastrip=<DATASTRIP_EXAMPLE_DIRECTORY>
      /DS_SGS__20180130T143750_20171129T023320
        --tile=<TILE_EXAMPLE_DIRECTORY>
      /S2A_OPER_MSI_L1C_TL_SGS__20171129T040150_A012727_T50PRA_N02.06
        --work_dir=<WORK_EXAMPLE_DIRECTORY>
        --output_dir=<OUTPUT_EXAMPLE_DIRECTORY>
        --GIP_L2A_PB=<GIP_EXAMPLE_DIRECTORY>/L2A_PB_GIPP.xml
        --img_database_dir=<DB_EXAMPLE_DIRECTORY_1>
        --res_database_dir=<DB_EXAMPLE_DIRECTORY_2>
        --raw
```

- A new command line option allows specifying the two locations of the image and the temporary database. Two different user selectable locations are available in order to select between the separate usage of a ramdisk and a normal disk space. The configuration is performed via command line (see section 4.3). By default, the databases will be created in the working directory given via command line[3].
- A new GIPP 'GIP_PROBA2' GIPP file of the format below can be provided as input to Sen2Cor as an additional optional command line argument (see paragraph 4.3.1 together with the list of other GIPPs) in order to define the processing baseline identifier of the generated L2A products. Sen2Cor is able to read this new input and use the corresponding value of the processing baseline filed in the relevant metadata and file/folder names of the L2A output product structure. In case the GIPP is not given

[3] Please note (!): the 10m processing uses some parts of the outputs from the 20m processing (Scene Classification, Aerosol Optical Thickness, Water Vapour and Visibility). In addition, both processing steps use the same bands located in the image database. Thus, although it is possible to address two different locations for the image and the temporary database, it is very important that the location for the individual database keeps the same for the two processing steps and that both databases are not removed externally. The processor itself will keep care for the proper housekeeping of the two databases.

as input, Sen2Cor uses the processing baseline identifier from the input Level-1C (in the L1C Datastrip metadata file).

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<GS2_PROCESSING_BASELINE_PARAMETERS
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
      xsi:noNamespaceSchemaLocation="L2A_PB_GIPP.xsd">
   <DATA>
      <Idp_Sc_List workflow="l2a">
      <Idp_Sc_Name>Sen2Cor</Idp_Sc_Name>
       <Idp_Sc_Version>02.07.00</Idp_Sc_Version>
      </Idp_Sc_List>
      <Baseline_Version>02.07</Baseline_Version>
   </DATA>
</GS2_PROCESSING_BASELINE_PARAMETERS>
```

4.4.3 Common Command Line Options

--resolution is the target resolution for the product to be processed:

- If resolution is omitted, all resolutions will be generated (providing that the option **Downsample_20_to_60** be selected in the L2A_GIPP.xml).
- If the user instead explicitly wants to process a 60m resolution, this can be performed via the option --resolution=60 in the command line. In the same way a 20 or 10m resolution processing can be selected.
- The selection of a 10 m resolution requires the generation of a 20 m product. This will be processed first, if the 10 m option is chosen.

4.4.4 Integration into the Sentinel-2 Toolbox

A detailed instruction the installation of Sen2Cor under the umbrella of the Sentinel-2 Toolbox is given under:

https://www.youtube.com/channel/UCPnL3aynCQxTOjPttxMiS3Q

4.5 The Software Development Environment

This chapter describes all necessary steps to install, configure and operate the necessary tools in order to extend or maintain the prototype processor software. The content of this chapter is normally not needed by normal users of the application.

4.5.1 Requirements and Third Party Software

The Sen2Cor development distribution contains an archive of essential open source tools, which provide the development environment for the processor. The following table specifies a comprehensive overview on all required items, including the runtime environment. The IDL runtime environment is only needed in that case, that a 1:1 testing is required with the initial ATCOR code, as this is written in IDL. The original ATCOR IDL code itself is not part of the development distribution and can only be requested at ESA by maintainers of the S2PAD project.

Table 4-1 - Third Party products for the SDE

Operating System			
Product	Name	Version	Source
Linux (alternatives)	CentOS RHEL Ubuntu	>= 6.0 6.0 12.04.2 LTS	http://www.centos.org/ http://www.de.redhat.com/products/rhel/desktop/ http://www.ubuntu.com/download/server
Software Development Environment			
Product	Name	Version	Source
IDE	Eclipse Classic	>= 3.7.1	http://www.eclipse.org/downloads/packages/
Python IDE for Eclipse	PyDev	>= 2.5.0	http://pydev.org/
XML Editor for Eclipse	Rinzo	1.4.0	http://editorxml.sourceforge.net/
Operational Tools and Libraries (see run_time_environment)			
Product	Name	Version	Source
Python Distribution Package (including Python 7.2, NumPy + SciPy, PyTables, Cython, Distutils and more …	Anaconda	>= 4.0.0	https://store.continuum.io

4.5.1.1 Eclipse, PyDev and Rinzo

Eclipse Classic is used and distributed as the standard development environment for the processor. Together with PyDev, it provides a complete development environment, including editor, syntax checker, debugger and runtime environment for python. Rinzo is a plugin for the syntax check of XML files, which are created and used as the standard I/O data format for the processor. For installation, it is referred to the installation procedures of the different tools as can be found in the references given in Table 4-1 above.

4.5.1.2 Anaconda

The Sen2Cor application uses a huge amount of python libraries, where "numpy" and "scipy" are only the most prominent ones. One requirement of this project was, to keep the installation overhead as low as possible. Thus, it was decided to use a free python distribution. We decided for Anaconda after several tests with different Python distributions. Anaconda is a completely free enterprise-ready Python distribution for large-scale data processing, predictive analytics, and scientific computing. Anaconda comes with its own installers, which facilitates the setup of the development environment considerably. Anaconda is able to support other operating systems (Mac OS and Windows) and thus avoiding the common problem of version mismatches between the different operating systems.

4.5.1.3 Cython

Cython is a python compatible language, which enables to convert native python code into C, so that the code can be compiled into a shared library and can be executed without exposing the readable python code. This technique is used here in order to protect the licensed ATCOR algorithm in the runtime distribution from being read and analysed. Cython is an integrated part of the Anaconda distribution described above.

4.5.1.4 Distutils

The Distutils are the Python standard for producing distribution packages. It is used here for generating both, the development and the runtime installation package. Distutils is an integrated part of the Anaconda distribution described above.

4.5.1.5 GDAL

GDAL is a translator library for raster geospatial data formats that is released under an X/MIT style Open Source license by the Open Source Geospatial Foundation. As a library, it presents a single abstract data model to the calling application for all supported formats. It also comes with a variety of useful commandline utilities for data translation and processing, which is used here (among others) for preparing the Digital Elevation Map.

4.5.1.6 PyTables

PyTables is a package for managing hierarchical datasets and designed to efficiently and easily cope with large amounts of data.

PyTables is built on top of the HDF5 library, using the Python language and the NumPy package. It features an object-oriented interface that, combined with C extensions for the performance-critical parts of the code (generated using Cython), makes it a fast, yet extremely easy to use tool for interactively browse, process and search very large amounts of data. One important feature of PyTables is that it optimizes memory and disk resources so that data takes much less space (specially if on-flight compression is used) than other solutions such as relational or object oriented databases. PyTables is an integrated part of the Anaconda distribution described above.

4.5.2 Installation

The software development distribution (SDD) is currently only provided for Linux. Baseline is CentOS 6.0, resp. RHEL 6.0. It is in principle possible to setup the complete SDE for other Linux Distributions or other operating systems such as Mac OSX or Windows. However, the SDD is not tested extensively on these platforms, and thus not recommended. The SDD is provided in form of the original archives, to be found in the project subdirectory labelled 'development_environment'.

For the installation of the development environment two different installation scenarios are possible:

Global installation (default): Eclipse will be installed within a common directory like e.g*. /opt* on the target system. For this installation root access on the target system is needed.

Local installation (optional): Eclipse will be installed under a certain user account (e.g. /home/s2l2app on the target system. For this installation a corresponding user account must exist and the installation shall be performed under that account.

Although Eclipse and PyDev are part of the distribution packages, it is recommended to install these third party tools from the original software providers, following the corresponding installation steps as given in the following links:

- Eclipse: http://wiki.eclipse.org/Eclipse/Installation#Install_a_JVM
- PyDev: https://wiki.appcelerator.org/display/tis/PyDev+Install

4.5.3 Configuration

4.5.3.1 Configure Python

In order to select Anaconda as the Python Interpreter of your choice, within Eclipse, first click on */Project/Properties/PyDev – Interpreter*, second, click on ***Configure and Interpreter not listed.***

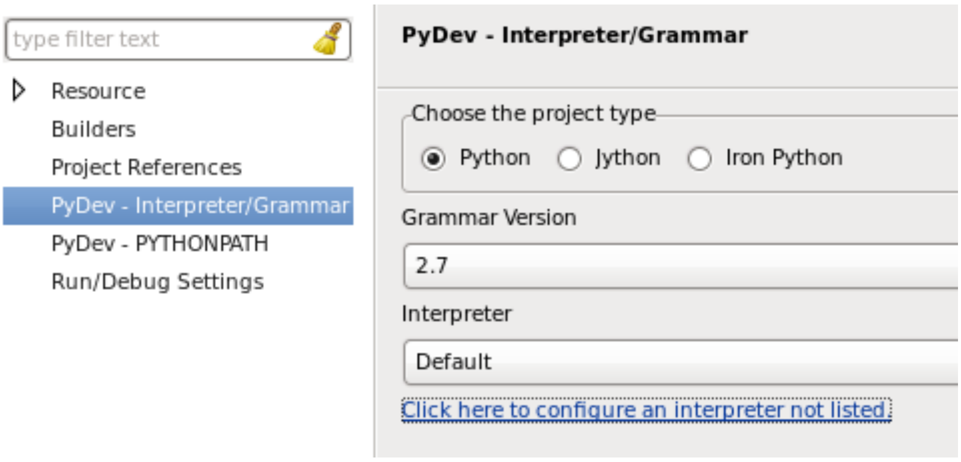

Figure 4-3 – Configuration of Python interpreter within Eclipse I

Then click on ***New*** and select the Python Interpreter in the folder where Anaconda was installed. An example installation is shown below:

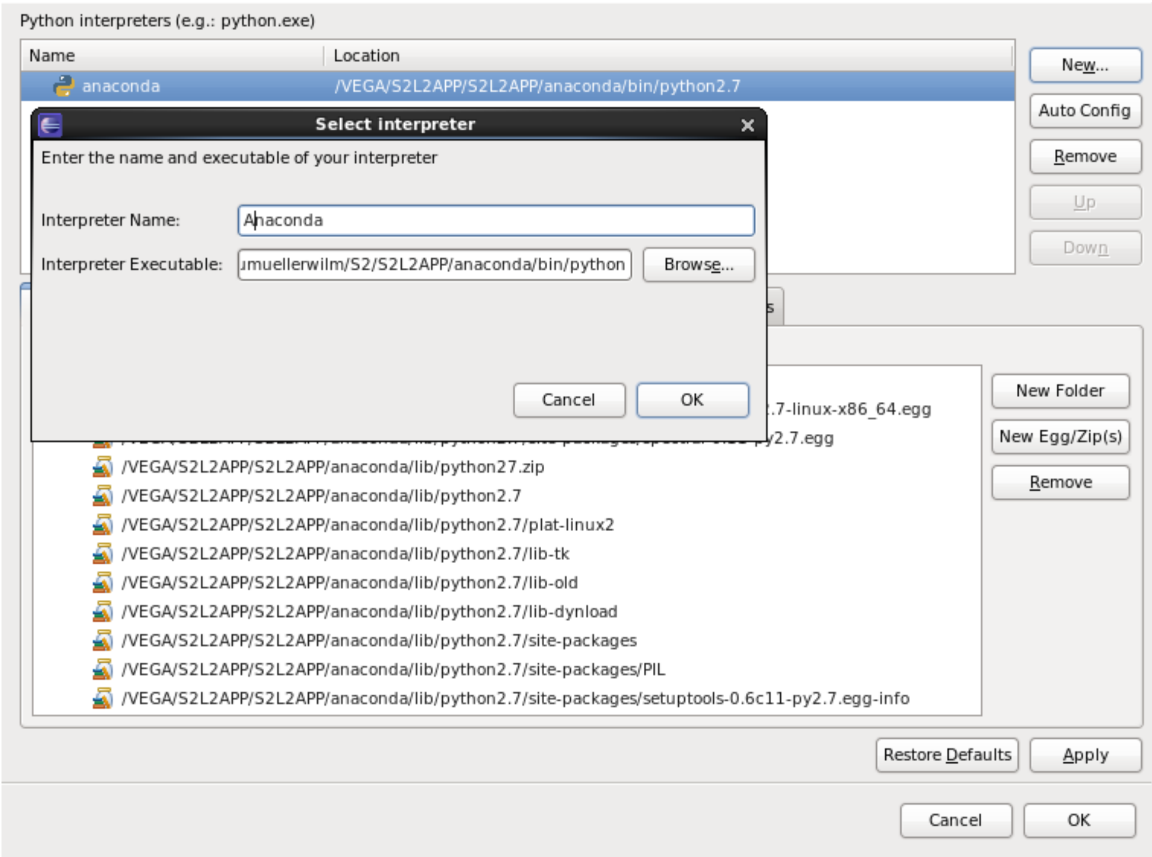

Figure 4-4 – Configuration of Python interpreter within Eclipse II

4.5.3.2 Environment Settings

An environment variable is necessary to be set up in order to obtain a full-configured system:

$SEN2COR_HOME: this environment variable named is used to configure the root directory of the Sen2Cor application package. Example: if the Sen2Cor application package is installed *under /home/sen2cor*, *$SEN2COR_HOME* shall be assigned to:

export SEN2COR_HOME = /home/sen2cor

The settings depend on your local setup.

This allows the operation with multiple configuration settings or the usage of the processor within different environments.

This environment variable can be configured inside of the Eclipse workspace via */Project/Properties/PyDev – Interpreter*, as shown below:

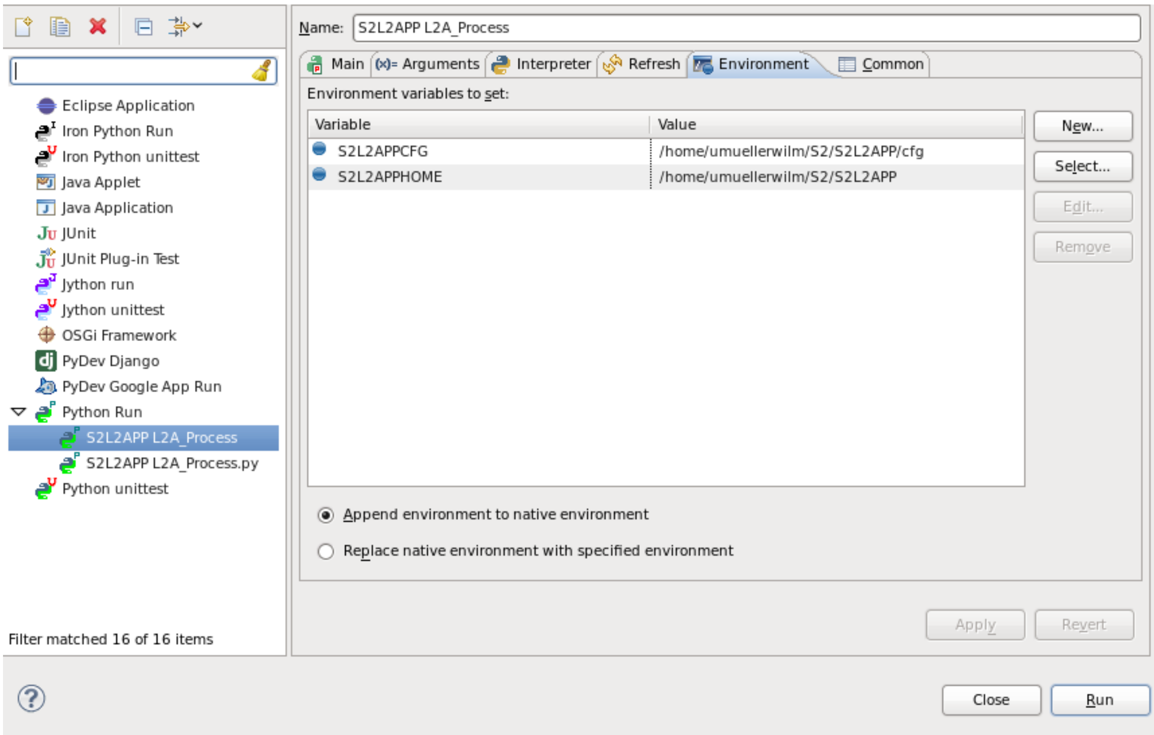

Figure 4-5 – Environment setting via Eclipse

Details on the RTE configuration are given in section 2.4.

4.5.4 Operation

4.5.4.1 Running the processor within Eclipse

Within Eclipse, create a new project using the root of your project (*$SEN2COR_HOME*)

Within Eclipse perform a refresh. The directory will show your project.

The main() operation of the Sen2Cor is located in the L2A_Process module. For testing purposes within Eclipse the whole application can be operated in two different ways:

1. Runtime Mode: */Run/Run*
2. Debug Mode*: /Run/Debug*

In order to simulate a scenario for testing purposes a command line can be configured within Eclipse via:

/Run/Run Configurations

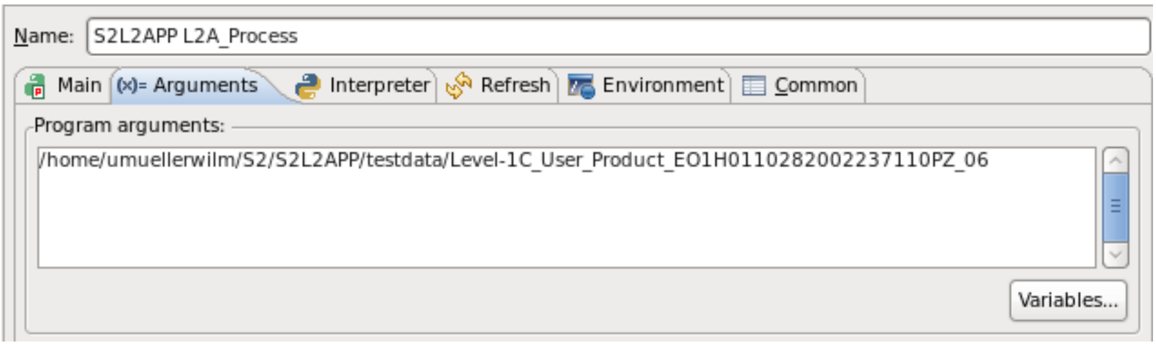

Figure 4-6 – SCD command line arguments

All other operational issues are equivalent with the scenario described for section 1.1.1.

4.5.4.2 Generating a Source Distribution

From the Eclipse workspace or via command line shell execute:

python setup.py s2l2app_SCD.py

This will generate an archive of the form:

s2l2app_SCD_<VERSION>.tar.gz

which then can be used for redistribution. The output will be generated in the distribution folder of the SDE.

Please be aware that the source redistribution is only allowed to be delivered to ESA itself or such ESA contractors, being in charge for the development or maintenance of the processor. In case of doubts, contact the corresponding Technical Officer of the project for clarification.

4.5.4.3 Generating a Build Distribution

The source code of the original ATCOR software is protected by a license agreement (see appendix). As a consequence, the Python source code of the processor core algorithm cannot be distributed to the end user. Thus, Cython is used as an intermediate development step. Cython is is able to generate C source code from python code. The generated C code can then be compiled platform specific on each target platform and can finally be redistributed in a non-human readable shared object or dynamic link library. Following target platforms are currently tested and will be supported:

Linux x86, 64 bit (CENTOS 6.0, alternatively RHEL 6.0)

MacOSX, x86, 64 (Mountain Lion)

Windows, 64 bit (Tested on Windows 7 Enterprise)

Generation of a build distribution therefore is performed in two steps:

Compilation of python source file into a target specific runtime library. This is done via the command *python setup.py build_ext* and must be executed on the target platform for which the distribution package shall be created.

Finally, the platform specific library together with all provided sources, test data, configuration and documentation is archived into platform specific distribution packages, which are named:

Sen2Cor_RTD-x.y.z-CentOS-x86_64.tar.gz

Sen2Cor_RTD-x.y.z-MacOSX-x86_64.tar.gz

Sen2Cor_RTD-x.y.z-Windows-x86_64.tar.gz

The generated packages can then be burned and delivered via a distribution medium.

5. References

1. Richter, R., Wang, X., Bachmann, M. and Schlaepfer, D. (2011). Correction of cirrus effects in Sentinel-2 type of imagery. *International Journal of Remote Sensing,* ***32****,* 2931-2941.
2. Louis, J., A. Charantonis, A. and Berthelot, B (2010). Cloud Detection for Sentinel-2. *Proceedings of ESA Living Planet Symposium.*
3. Salomonson V.V., I. Appel I. (2004). Estimating fractional snow cover from MODIS using the normalized difference snow index. *Remote Sensing of Environment* **89**, 351–360.
4. Vane, G., Green, R. O., Chrien, T. G., Enmark, H. T., Hansen, E. G., and Porter, W. M. (1993). The airborne visible / infrared imaging spectrometer (AVIRIS). *Remote Sens. Environ.* **44**, 127-143.
5. Kaufman, Y., et al. (1997). The MODIS 2.1 µm channel – correlation with visible reflectance for use in remote sensing of aerosol. *IEEE TGRS,* **35,** 1286 – 1298.
6. Schläpfer, D. et al. (1998). Atmospheric precorrected differential absorption technique to retrieve columnar water vapour. *Remote Sens. Environ.,* **65**, 353-366.

6. Copyright Notes

"Derivative Works" shall mean any work, whether in Source or Object form, that is based on (or derived from) the Work and for which the editorial revisions, annotations, elaborations, or other modifications represent, as a whole, an original work of authorship. For the purposes of this License, Derivative Works shall not include works that remain separable from, or merely link (or bind by name) to the interfaces of, the Work and Derivative Works thereof.

"Contribution" shall mean any work of authorship, including the original version of the Work and any modifications or additions to that Work or Derivative Works thereof, that is intentionally submitted to Licensor for inclusion in the Work by the copyright owner or by an individual or Legal Entity authorized to submit on behalf of the copyright owner. For the purposes of this definition, "submitted" means any form of electronic, verbal, or written communication sent to the Licensor or its representatives, including but not limited to communication on electronic mailing lists, source code control systems, and issue tracking systems that are managed by, or on behalf of, the Licensor for the purpose of discussing and improving the Work, but excluding communication that is conspicuously marked or otherwise designated in writing by the copyright owner as "Not a Contribution."

"Contributor" shall mean Licensor and any individual or Legal Entity on behalf of whom a Contribution has been received by Licensor and subsequently incorporated within the Work.

2. Grant of Copyright License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form.

3. Grant of Patent License. Subject to the terms and conditions of this License, each Contributor hereby grants to You a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable (except as stated in this section) patent license to make, have made, use, offer to sell, sell, import, and otherwise transfer the Work, where such license applies only to those patent claims licensable by such Contributor that are necessarily infringed by their Contribution(s) alone or by combination of their Contribution(s) with the Work to which such Contribution(s) was submitted. If You

institute patent litigation against any entity (including a cross-claim or counterclaim in a lawsuit) alleging that the Work or a Contribution incorporated within the Work constitutes direct or contributory patent infringement, then any patent licenses granted to You under this License for that Work shall terminate as of the date such litigation is filed.

4. Redistribution. You may reproduce and distribute copies of the Work or Derivative Works thereof in any medium, with or without modifications, and in Source or Object form, provided that You meet the following conditions:

(a) You must give any other recipients of the Work or Derivative Works a copy of this License; and

(b) You must cause any modified files to carry prominent notices stating that You changed the files; and

(c) You must retain, in the Source form of any Derivative Works that You distribute, all copyright, patent, trademark, and attribution notices from the Source form of the Work, excluding those notices that do not pertain to any part of the Derivative Works; and

(d) If the Work includes a "NOTICE" text file as part of its distribution, then any Derivative Works that You distribute must include a readable copy of the attribution notices contained within such NOTICE file, excluding those notices that do not pertain to any part of the Derivative Works, in at least one of the following places: within a NOTICE text file distributed as part of the Derivative Works; within the Source form or documentation, if provided along with the Derivative Works; or, within a display generated by the Derivative Works, if and wherever such third-party notices normally appear. The contents of the NOTICE file are for informational purposes only and do not modify the License. You may add Your own attribution notices within Derivative Works that You distribute, alongside or as an addendum to the NOTICE text from the Work, provided that such additional attribution notices cannot be construed as modifying the License.

You may add Your own copyright statement to Your modifications and may provide additional or different license terms and conditions for use, reproduction, or distribution of Your modifications, or for any such Derivative Works as a whole, provided Your use, reproduction, and distribution of the Work otherwise complies with the conditions stated in this License.

5. Submission of Contributions. Unless You explicitly state otherwise, any Contribution intentionally submitted for inclusion in the Work by You to the Licensor shall be under the terms and conditions of this License, without any additional terms or conditions. Notwithstanding the above, nothing herein shall supersede or modify the terms of any separate license agreement you may have executed with Licensor regarding such Contributions.

6. Trademarks. This License does not grant permission to use the trade names, trademarks, service marks, or product names of the Licensor, except as required for reasonable and customary use in describing the origin of the Work and reproducing the content of the NOTICE file.

7. Disclaimer of Warranty. Unless required by applicable law or agreed to in writing, Licensor provides the Work (and each Contributor provides its Contributions) on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied, including, without limitation, any warranties or conditions of TITLE, NON-INFRINGEMENT, MERCHANTABILITY, or FITNESS FOR A PARTICULAR PURPOSE. You are solely responsible for determining the appropriateness of using or redistributing the Work and assume any risks associated with Your exercise of permissions under this License.

8. Limitation of Liability. In no event and under no legal theory, whether in tort (including negligence), contract, or otherwise, unless required by applicable law (such as deliberate and grossly negligent acts) or agreed to in writing, shall any Contributor be liable to You for damages, including any direct, indirect, special, incidental, or consequential damages of any character arising as a result of this License or out of the use or inability to use the Work (including but not limited to damages for loss of goodwill, work stoppage, computer failure or malfunction, or any and all other commercial damages or losses), even if such Contributor has been advised of the possibility of such damages.

9. Accepting Warranty or Additional Liability. While redistributing

ATCOR License Agreement

The program "ATCOR" (hereinafer called "program") is proprietary of the German Aerospace center (Deutsches Zentrum Für Luft- und Raumfahrt e.V., DLR) and is protected by the German Copyright Act. The program is made avaialable to customer only for the purpose of use for the ESA Sentinel-2-project, the use of ATCOR for other projects is expressively excluded. With acceptance of these terms of a license the customer receives - without the right to grant sublicenses - a non exclusive, non transferable right to use the program for Sentinel-2 project. This right comprises the right to reprocess the source-code and to translate the source-code into a different programming language (hereinafter called "derived ATCOR")

Definition of the terms ATCOR sources and ATCOR binaries:
The term ATCOR sources specify the following three items: the source code of ATCOR or the source code or Python byte code of the derived ATCOR program.

The term ATCOR binaries specify a non-readable version of the derived ATCOR program. This can be either a binary executable program or an object library which is compiled from the derived ATCOR sources

The ATCOR sources may only be used for own purposes of the customer related to the Sentinel-2 project. The transfer of the ATCOR sources to third parties other than ESA and its ESA selected contractors, or the use of ATCOR sources for other projects than Sentinel-2 is subject to the prior written consent of DLR. No restrictions shall be given in the use of ATCOR binaries.

Notice of Third Party Software Licenses

This package contains open source software binaries and or python code from external projects. These are available on an "as is" basis and subject to their individual license agreements. This package is subject to their individual licenses as well as the Sen2Cor code license. Sen2cor development team reserves the right to change which third party tools will be provided in any version of this project. Changes to license in individual packages will be updated in this file.

Sen2cor windows package contains re-distributable, run-time, shared-library files from the Intel(TM) Math Kernel Library ("MKL binaries"). You are specifically authorized to use the MKL binaries. Use and redistribution of the MKL binaries are subject to the licensing terms located at https://software.intel.com/en-us/license/intel-simplified-software-license. However, GNU/Linux and Apple Mac OSX platforms are free from intel MKL binaries.

This software includes code from Python(R) project
Copyright (c) 2001, 2002, 2003, 2004, 2005, 2006 Python Software Foundation;
http://www.python.org/
Python(R) is a registered trademark of the Python Software Foundation.
Python is released under PSF License:
https://docs.python.org/2/license.html

This software includes binaries from musl libc
Copyright © 2005-2014 Rich Felker, et al.
http://www.musl-libc.org/
Musl is released under standard MIT License:
http://git.musl-libc.org/cgit/musl/tree/COPYRIGHT

This software includes code from GDAL.
Copyright (c) 2000-2016, Frank Warmerdam
http://www.gdal.org/
GDAL is released under X11/MIT License:
https://trac.osgeo.org/gdal/wiki/FAQGeneral#WhatlicensedoesGDALOGRuse

This software includes code from numpy
Copyright (c) 2005, NumPy Developers
http://www.numpy.org/
Numpy is released under BSD-New license:
https://docs.scipy.org/doc/numpy-1.10.0/license.html

This software includes code from scipy

http://www.scipy.org/
Scipy is released under BSD-New license:
https://www.scipy.org/scipylib/license.html

This software includes code from pytables

http://www.pytables.org/
pytables is released under BSD 3-clause "New" or "Revised" License
https://github.com/PyTables/PyTables/blob/2.2-pro/LICENSE.txt

This software includes code from hdf5

http://www.hdfgroup.org/
License for hdf5 is given here:
https://support.hdfgroup.org/products/licenses.html

This software includes code from numexpr.

https://github.com/pydata/numexpr
numexpr is released under MIT License:
https://github.com/pydata/numexpr/blob/master/LICENSE.txt

This software includes code from lxml.

http://lxml.de/
lxml is released under BSD License
https://github.com/lxml/lxml/blob/master/doc/licenses/BSD.txt

This software includes code from psutil.

https://github.com/giampaolo/psutil
psutil is distributed under BSD license
https://github.com/giampaolo/psutil/blob/master/LICENSE

Files:

lib/python2.7/site-packages/sen2cor/L2A_Config.py

lib/python2.7/site-packages/sen2cor/L2A_Library.py
lib/python2.7/site-packages/sen2cor/L2A_Logger.py
lib/python2.7/site-packages/sen2cor/L2A_Manifest.py
lib/python2.7/site-packages/sen2cor/L2A_Process.py
lib/python2.7/site-packages/sen2cor/L2A_ProcessTile.py
lib/python2.7/site-packages/sen2cor/L2A_SceneClass.py
lib/python2.7/site-packages/sen2cor/L2A_Schedule.py
lib/python2.7/site-packages/sen2cor/L2A_Tables.py
lib/python2.7/site-packages/sen2cor/L2A_XmlParser.py
lib/python2.7/site-packages/sen2cor/__init__.py

Copyright: Copyright (C) 2017 European Space Agency (ESA).
All Rights reserved
License: Apache 2.0

Files : lib/python2.7/site-packages/sen2cor/aux_data/
Copyright : Copyright (C) 2017 European Space Agency (ESA).
All Rights reserved
License : Apache 2.0

Files : lib/python2.7/site-packages/sen2cor/cfg/*
Copyright : Copyright (C) 2017 European Space Agency (ESA).
All Rights reserved
License : Apache 2.0

Files : lib/python2.7/site-packages/sen2cor/lib_S2A/*
Copyright : Copyright (C) 2017 European Space Agency (ESA).
All Rights reserved
License : Apache 2.0

Files : lib/python2.7/site-packages/sen2cor/lib_S2B/*
Copyright : Copyright (C) 2017 European Space Agency (ESA).
All Rights reserved
License : Apache 2.0

See copyright/Sen2cor for full text of apache 2.0 license

Files: lib/python2.7/site-packages/sen2cor/L2A_AtmCorr.so
Copyright: German Aerospace center (Deutsches Zentrum Für Luft- und Raumfahrt e.V., DLR)
License: see copyright/ATCOR which is the license granted to ESA by DLR

This software contains binaries from lapack project.

http://www.netlib.org/

Lapack is released under modified BSD license:

http://www.netlib.org/lapack/LICENSE.txt

This package contains binaries from libpng project

http://www.libpng.org/

libpng is distributed under libpng license

Copyright and license are listed here:

http://www.libpng.org/pub/png/src/libpng-LICENSE.txt

This package contains binaries from zlib project

https://www.zlib.net/

zlib is released under zlib license:

https://www.zlib.net/zlib_license.html

This package contains binaries from curl project

https://curl.haxx.se

Curl is released under MIT/X derivative license:

https://curl.haxx.se/docs/copyright.html

This package contains binaries from jpeg-turbo project

http://www.libjpeg-turbo.org

libjpeg-turbo is covered by three compatible BSD-style open source licenses:

https://github.com/libjpeg-turbo/libjpeg-turbo/blob/master/LICENSE.md

This package contains binaries from libxslt

http://xmlsoft.org

xslt is released under MIT license:

http://xmlsoft.org/XSLT/intro.html

This package contains binaries from libexslt

http://xmlsoft.org

exslt is released under MIT license:

http://xmlsoft.org/XSLT/EXSLT/index.html

This package contains binaries from libxml2
http://xmlsoft.org
libxml2 is released under MIT license

This package contains binaries from libtiff
Copyright (c) 1988-1997 Sam Leffler
Copyright (c) 1991-1997 Silicon Graphics, Inc.
http://www.libtiff.org/
libtiff is released under BSD-like licence.

This package contains binaries from proj4
Copyright (c) 2000, Frank Warmerdam
proj4 is released under MIT licence:
http://proj4.org/license.html

This package contains binaries from libexpat
Copyright (c) 1998-2000 Thai Open Source Software Center Ltd and Clark Cooper
Copyright (c) 2001-2017 Expat maintainers
https://libexpat.github.io/
libexpat is released under MIT licence.
https://github.com/libexpat/libexpat/blob/master/expat/COPYING

This package contains binaries from Freetype2
Copyright (c) 2000, Frank Warmerdam
Freetype2 is released under FTL licence:
http://git.savannah.gnu.org/cgit/freetype/freetype2.git/tree/docs/FTL.TXT

This package contains binaries from libgeotiff project
libgeotiff is released under public domain or X style licence:
https://svn.osgeo.org/metacrs/geotiff/trunk/libgeotiff/LICENSE

This package contains binaries from SQLite project
https://sqlite.org/
SQLite is released under public domain
https://sqlite.org/copyright.html

This package contains binaries from OpenJPEG project
http://www.openjpeg.org/
OpenJPEG is released under 2-clause BSD license
https://github.com/uclouvain/openjpeg/blob/master/LICENSE

This package contains binaries from GNU Readline project

https://cnswww.cns.cwru.edu/php/chet/readline/rltop.html
Readline is released under GNU GPL v3 License.
http://www.gnu.org/licenses/gpl.html

This package contains binaries from lcms2 project
http://www.littlecms.com/
lcms2 is released under MIT license

This package contains binaries from libffi project
https://sourceware.org/libffi/
libffi is released under MIT license
https://github.com/libffi/libffi/blob/master/LICENSE

This package contains binaries from ncurses project
Copyright (c) 2011-2014, 2015 by Thomas E. Dickey
https://www.gnu.org/software/ncurses/
ncurses is distributed under MIT license
http://invisible-island.net/ncurses/ncurses-license.html

This package contains binaries from libssh2 project
http://www.libssh2.org/
ncurses is distributed under BSD license
https://libssh2.org/license.html

This package contains binaries from libressl project
http://www.libressl.org/
libressl is distributed under a mix of licenses
https://github.com/libressl/libressl/blob/master/src/LICENSE

This package contains binaries from gnupg project
https://www.gnupg.org/
gnupg is distributed under a GPL license

This package contains binaries from libbz2 project
Copyright (c) Julian Seward, jseward@bzip.org
http://www.bzip.org/
libbz2 is distributed under BSD license

This package contains binaries from BLAS project
http://www.netlib.org/blas/
BLAS is distributed under public domain

This package contains binaries from OpenBLAS project
http://www.openblas.net/
OpenBLAS is distributed under BSD License

This package contains binaries from GDBM project
http://www.gnu.org.ua/software/gdbm/
GDBM is released under GPL v3 License

This software includes binaries from GCC project
Copyright (C) 2009 Free Software Foundation, Inc.
libstdc++ has GCC Runtime Library Exception:
https://gcc.gnu.org/onlinedocs/libstdc++/manual/license.html
The libgcc library is licensed under the GNU GPL plus the GCC Runtime Library Exception