SMOS L1 API v5.5.0 User Manual

Code	**:**	SO-UM-DME-L1PP-0247
Issue	**:**	2.2
Date	**:**	25/11/11

	Name	Function	Signature
Prepared by	A. Gutiérrez	Project Engineer	
Checked by	J. Barbosa	Quality A. Manager	
Approved by	J. Barbosa	Project Manager	

DEIMOS Engenharia
Av. D. João II, Lote 1.17, Torre Zen, 10º
1998-023 Lisboa, PORTUGAL
Tel: +351 21 893 3017
Fax: +351 21 896 9099
E-mail: mailto:deimos@deimos.com.pt

This page intentionally left blank

Document Information

Contract Data	Classification
Contract Number: 4000101241/10/I-AM	**Internal** ☐ **Public** ☐ **Industry** ☐ **Confidential** ☑
Contract Issuer: ESA	

Internal Distribution		
Name	Unit	Copies

External Distribution		
Name	Organisation	Copies
Jean-Claude Debruyn	ESA	1
Steven Delwart	ESA	1

Archiving	
Word Processor:	MS Word 2000
File Name:	SO-UM-DME-L1PP-0247-L1-API-Software-User-Manual_25Nov.doc
Archive Code:	SO-UM-DME-L1PP-0247

Document Status Log

Issue	Change description	Date	Approved
1.0	First version for the Maintenance Phase and L1PP V3.5.0	2010-12-03	
2.0	Second version for L1PP V5.0.0	2011-05-20	
2.2	Third version for L1PP V5.5.0	2011-11-25	

Table of Contents

List of Tables

1. INTRODUCTION

1.1. Purpose and Scope

This purpose of this document is to provide to the user all the information needed for installing and running the SMOS L1 API (L1-API) v5.5.0. This User Manual provides the following information:

- Installation Steps;
- Configuration Procedures;
- Description of the functionalities and L1-API usage;
- Description of L1-API executable tool.

This document was produced in the scope of the project "SMOS Level 1 Processor Prototype Development".

1.1.1. Acronyms and Abbreviations

ADF	Auxiliary Data Files
API	Application Programming Interface
APID	Application program identifier
CFI	Customer Furnished Item
COTS	Commercial Off-The-Shelf
DPM	Data Processing Model
EE	Earth Explorer
EEFH	Earth Explorer File Handling CFI (ASCII XML library)
EM	Engineering Model
FWF	Fringe Wash Function
GUI	Graphical User Interface
HKTM	HouseKeeping Telemetry
HTML	HyperText Markup Language
IGRF	International Geomagnetic Reference Field
IRI	International Reference Ionosphere
ISO	International Organization for Standardization

IVT	Image Validation Test
L1PP	Level 1 processor prototype
LCF	LiCeF (Lightweight and Cost-Effective Front-end)
MIRAS	Microwave Imaging Radiometer with Aperture Synthesis
NIR	Noise Injection Radiometer
NRTP	Near Real Time Processor
OBET	On Board Elapsed Time
PLM	PayLoad Module
PMS	Power Measurement Signal
SEPS	SMOS End-to-end Performance Simulator
SMOS	Soil Moisture and Ocean Salinity
SVP	Software Validation Plan
TBW	To Be Written
UPC	*Universitat Politècnica de Catalunya (*Technical University of Catalonia*) Barcelona Tech.*
XML	Extended Markup Language
XSL	eXtensible Stylesheet Language

Table 1: Table of Acronyms.

For the complete list of acronyms, please refer to the document SO-LI-CASA-PLM-0094 “Directory of Acronyms and abbreviations” [RD.5].

1.2. Applicable and Reference Documents

1.2.1. Applicable Documents

Ref.	Code	Title	Issue	Date
AD.1	SO-SOW-ESA-GS-6647	SMOS Expert Support Laboratories for the period 2010-2014 - ESL Level 1 Calibration and Reconstruction	1.2	10-12-2009
AD.2	EE-MA-DMS-GS-0008	EE XML/Binary CFI File Handling Library User Manual	3.7.3	7-05-2010
AD.3	SO-DS-DME-L1PP-0007	SMOS L1 Processor L0 to L1a Data Processing Model	2.15	25-11-2011

Ref.	Code	Title	Issue	Date
AD.4	SO-DS-DME-L1PP-0008	SMOS L1 Processor L1a to L1b Data Processing Model	2.15	25-11-2011
AD.5	SO-DS-DME-L1PP-0009	SMOS L1 Processor L1b to L1c Data Processing Model	2.10	25-11-2011
AD.6	SO-TN-IDR-GS-0005	SMOS Level 1 and Auxiliary Data Products Specification	5.22	25-11-2011
AD.7	SO-DD-DME-L1PP-0250	SMOS L1-API Detailed Design Definition Document	2.2	25-11-2011

Table 2: Applicable Documents.

1.2.2. Reference Documents

Ref.	Code	Title	Issue	Date
RD.1	EE-MA-DMS-GS-0001	Earth Explorer Mission CFI Software MISSION CONVENTIONS DOCUMENT	3.7.3	7-05-2010
RD.2	PE-TN-ESA-GS-0001	Earth Explorer Ground Segment File Format Standard	1.4	13-06-2003
RD.3	EE-MA-DMS-GS-0002	Earth Explorer Mission CFI Software GENERAL SOFTWARE USER MANUAL	3.7.3	7-05-2010
RD.4	EE-MA-DMS-GS-0008	EXPLORER FILE HANDLING Reference Manual	3.7.3	7-05-2010
RD.5	SO-LI-CASA-PLM-0094	Directory of Acronyms and abbreviations		

Table 3: Reference Documents

2. SMOS L1 API GUIDE

This chapter presents all the information needed by the user in order to understand the objective and the functioning of the SMOS L1-API v5.5.0. The chapter first introduces the L1-API application, presenting summarily its objectives and components. Then the installation, configuration usage and tuning procedures are detailed.

2.1. Objectives

The purpose of the SMOS L1-API is to provide the SMOS NRTP and L1OP processors with a self-contained set of SMOS L1 algorithm processing libraries.

The API provides a fixed interface to these processing libraries, such that the SMOS L1 algorithms are totally encapsulated and can be used from any external application. These applications will need to handle the reading and writing of input/output files as well as the orchestration and execution of the different L1 processing functions. A logging functionality is provided within the L1-API, although it can be overloaded from the external application.

The SMOS L1-API v5.5.0 supports the operational DPGS V3 format, described in [AD.6], although the read/write routines themselves are not part of the API.

In addition, the SMOS L1-API v5.5.0 is fully aligned with the SMOS L1 processing algorithms implemented in L1PP v5.5.0

2.2. Components

2.2.1. Processing Functions

The SMOS L1-API includes different L1 processing functions, responsible for the execution of the following tasks:

- **processL0AncillaryData**: function to generate HTKM telemetry L1a products from L0 products
- **processL0CorrelatedNoiseData**: function to generate Correlated Noise Injection L1a products from L0 products
- **processL0FWFCorrelatedNoiseData**: function to generate FWF Shape L1a products from L0 products
- **processL0UncorrelatedNoiseData**: function to generate Uncorrelated Noise Injection L1a products from L0 products
- **processL0NIRData**: function to generate NIR calibration L1a products from L0 products
- **processL0ScienceDualData**: function to generate Science Dual L1a-b-c products from L0 products

- **processL0ToL1AScienceDualData**: function to generate Science Dual L1a products from L0 products
- **processL1AToL1BScienceDualData**: function to generate Science Dual L1b products from L1a products
- **processL1BToL1CScienceData**: function to generate Science Dual L1c products from L1b products

❑ **processL0ScienceFullData**: function to generate Science Full L1a-b-c products from L0 products

- **processL0ToL1AScienceFullData**: function to generate Science Full L1a products from L0 products
- **processL1AToL1BScienceFullData**: function to generate Science Full L1b products from L1a products
- **processL1BToL1CScienceData**: function to generate Science Full L1c products from L1b products

❑ **processFlatTargetDualData**: function to generate the Flat Target Transformation Dual Pol in-orbit calibration products from External Science L1a Dual Pol products

❑ **processFlatTargetFullData**: function to generate the Flat Target Transformation Full Pol products from External Science L1a Full Pol products

❑ **gMatrixGenerator**: function to generate the G Matrix in-orbit calibration products using FWF Shape L1a products as input

❑ **jMatrixGeneratorMM**: function to generate the Cross-polar J Matrix in-orbit calibration products using FWF Shape L1a products as input

❑ **jMultiMatrixGeneratorMM**: function to generate the Co-polar J Matrix in-orbit calibration products using FWF Shape L1a products as input

For further information on the L1 processing function components, please refer to [AD.7].

2.2.2. SMOS L1-API Core Libraries

The SMOS L1-API is fully contained in a set of dynamic libraries that hold the above mentioned processing functions. This allows the building of executable programs by linking the main program to these libraries.

The physical name of the libraries provided by the SMOS L1-API is the following:

❑ **libNrtpL1a.so**: contains all processing functions from L0 to L1a levels, including all L1a in-orbit calibration processing

- **libNrtpL1b.so**: contains all processing functions from L1a to L1b levels, including all L1b in-orbit calibration processing (i.e. FTT, G Matrix, J Matrix)
- **libNrtpL1c.so**: contains all processing functions from L1b to L1c levels
- **libNrtpCommon.so**: contains a wide set of basic mathematical functions and operators required by the remaining libraries, and which are common to all of them (e.g. matrix multiplication, complex number manipulations…)
- **libNrtpInterfaces.so**: contains a small set of read/write functions for all XML SMOS Auxiliary products, which can be useful if the external application does not want to replicate them.
- **libNrtpProcessor.so**: contains a basic logging functionality, by which error, warning and information messages are printed on the screen. This library can be overloaded by the external application linking to the L1-API, in order to redirect the log messages to any other desired location (e.g. a file, a socket…)

2.3. Installation guide

The following sections describe the steps necessary for installing L1-API: check Hardware requirements, install external libraries, execute the installation procedures and set environment variables.

2.3.1. Hardware Requirements

The SMOS L1-API may be run in a Pentium IV 64 with LINUX installed. The memory and disk resources required will depend on the type of algorithms being used. For full functionality, the user may need the following resources:

- 20 GB of Disk Space – for generating the G and J matrices, which are the biggest files produced by the SMOS in-orbit calibration. Nevertheless, the user should note that the volume rate produced by SMOS Science L1 data each half-orbit may exceed 1GB in full pol mode (L1a+L1b+L1c);
- Minimum 4GB RAM - the RAM memory available will be critical when handling the G-Matrix in-orbit calibration file (Level 1B). If the user wants to use the G-Matrix ADF for performing the Foreign Sources Correction, a minimum of 4GB is advisable for dual pol (8 GB recommended), and a minimum of 10GB is advisable for full pol (16GB recommended). If the user has less than 8 GB of RAM, a swap partition of at least 8 GB is also recommended as the total amount of memory needed in L1b processing may reach 10GB for a full pol product half-orbit. A possibility to reduce the RAM demand in full pol processing is to disable the cross-polarisation correction in the L1 configuration file.

2.3.2. Dependencies

In order to install the SMOS L1-API, it is recommended to have the following software already installed in the target platform:

- gcc 4.3 (http://gcc.gnu.org);
- glibc 2.3.4 (http://gcc.gnu.org);
- gfortran (version included in gcc 4.3);
- xerces 2.7.0 (http://xml.apache.org/xercesc/)
- ATLAS v3.8.2 (http://math-atlas.sourceforge.net/atlas_install/), including LAPACK 3.0 and BLAS (http://www.netlib.org/lapack/);
- LibXML2 v2.6 or higher ((http://xmlsoft.org);

Each product referred above should be installed according to its own instructions in the target platform.

2.3.3. Installation Kit Description

The SMOS L1-API is provided as an InstallKit composed of a single tgz file. This binary distribution is delivered as SMOS-L1-API-5.5.0_Installer.tgz which after decompressing contains:

- A directory (*include*) with all header files defining the interfaces to the processing functions
- A directory (*lib64*) with
  - all the physical dynamic library files containing the SMOS L1 processing functions themselves;
  - a dedicated dynamic library (libVegaRW.so) enabling the use of specific read/write routines developed by VEGA in the frame of the NRTP project (this library is needed by the L1 API Test Tool, as shall be explained in section 3.1)
- A directory (*bin*) with a self-executable test program that uses the L1-API to process L0 data (this program is described in detail in the test tools section)
- A directory (*doc*) with the SMOS L1-API DDD document in html format
- A directory (*data*) with the configuration of specific read/write routines needed by the test tool, and a backup copy of the IRI and IGRF model files (they are identical to the ones provided by L1PP v5.5.0)

In order to run the test tool the user will also need the Auxiliary Data Files and the test data scenarios, which are provided separately:

- All the Auxiliary Data Files needed by the SMOS L1-API are the same as the ones needed for the L1 prototype, so they can be retrieved from the SMOS L1PP webpage;

- *SMOS-L1-API-TestDataPackage_5_0.tgz*: contains several test data scenarios (L0 and L1 products), each packed independently, and the corresponding joborder files in order to be used directly by the test program.

2.3.4. SMOS L1-API Installation Steps

In order to install the L1-API, the user shall extract the *tgz* file:

$ tar xzvf SMOS-L1-API-5.5.0_Installer.tgz

After the installation is complete, the directories structure shown in Section 4 has been created by the installation script.

A user with an external application that wants to use the SMOS L1 Processing functions will only need the L1-API header files (*include*) and the L1-API dynamic libraries (*lib64*) to be able to successfully compile and build his application.

In addition, the user must define a specific environment variable called *NRTP_CONFIG*, pointing to the path where the IGRF and IRI ionospheric model parameters provided in the installation kit (*data/iono_models/*) are located, so that the parameters can be used by the SMOS L1-API. The non-availability of these models' parameters will not stop the L1-API from running; it will simply disable the TEC and IGRF model computation at L1c processing.

Note: The ionospheric model parameters provided with the L1-API are the same as the ones provided with the associated L1PP reference version (v5.5.0). Nevertheless, it is recommended that the user makes sure that these files are shared between applications (API and L1PP) so that no misalignment error is possible in the long term.

2.3.5. Environment Variables

As stated before, the variable *NRTP_CONFIG* is needed for the use of the SMOS L1-API, and has to be defined in the following way, where *L1_API_ROOT* is the home directory of the L1-API installation:

$ export NRTP_CONFIG=$L1_API_ROOT/data/

In addition, the user shall modify the variable LD_LIBRARY_PATH to include the directories where the SMOS L1-API libraries are installed, such as:

$ export LD_LIBRARY_PATH=$L1_API_ROOT/lib64:$LD_LIBRARY_PATH

Note: If the user changes the location of the libraries, the LD_LIBRARY_PATH must be updated accordingly; otherwise a program linking to these libraries may not find them and refuse to run.

3. SMOS L1-API TEST TOOL

As part of the SMOS L1-API activities, a comprehensive demonstration test tool was created as a demonstration of the encapsulation of the API. This test tool is command line executed and is available only for LINUX 64bit architectures.

This test tool (SMOS L1-API Executable) uses not only the L1-API but also the VEGA read/write library developed in the frame of the NRTP contract, the INDRA read/write library, the VEGA BUFR converter and the ECMWF BUFR encoding libraries. This allows the test tool to read and write L0, L1 and BUFR products, whose functionality is not directly supported by the L1-API.

It should be noted that the VEGA BUFR converter and the ECMWF BUFR library are static libraries and as such are already incorporated into the test tool executable file (i.e. they are not contained in the installation kit), but the ECMWF BUFR requires a set of BUFR encoding tables to be available in the system. This is why the ECMWF BUFR library has to be installed in the execution machine as described in the following chapter.

3.1. Dependencies

In order to execute the SMOS L1-API test tool, the following software must be installed in the target platform:

- gcc v4.3 (http://gcc.gnu.org), including the gfortran compiler;
- glibc v2.3.4 (http://gcc.gnu.org);
- XML RW API v04.01.05 (ftp://193.146.123.163/smos/software/XML_RW_API/)
  - In addition, the complete set of SMOS L1 and ADF schemas must be available in the target platform with an environment variable called XML_RW_API_HOME pointing to it.
  - Also, a copy or a soft link to the XML RW API configuration file (*xml_rw_api.usr_conf.xml*) must be available from the location where the test tool is executed (this is a requirement imposed by the XML RW API)
- xerces v2.7.0 (http://xml.apache.org/xercesc/) (needed by XML RW API)
- ATLAS v3.8.2 (http://math-atlas.sourceforge.net/atlas_install/), including LAPACK 3.0 and BLAS (http://www.netlib.org/lapack/);
- LibXML2 v2.6 or higher ((http://xmlsoft.org);
- ECMWF BUFR encoding/decoding library v386 (http://www.ecmwf.int/products/data/software/download/bufr.html)

- The installation target must be */usr/local/lib*, as prompted by default by the BUFR installer program

Each product referred above should be installed according to its own instructions in the target platform.

In addition, as it has been mentioned above, the SMOS L1-API test tool uses some components developed by VEGA in the frame of the NRTP project, namely L1 specific read/write routines and the ECMWF BUFR converter for SMOS L1c products.

A dedicated VEGA library is provided within the *lib64* directory of the installation kit that enables the test tool to use these routines. This VEGA library is controlled by a configuration file provided also in the installation kit (*data/rw/vega_rw_api.usr_conf.xml*), and requires the definition of an environment variable "NRTP_CONFIG" that points to the *data* directory, which should already be available if the user has defined the variables required in section 2.3.5.

3.2. Usage

First of all, the user must take care to define the environment variable LD_LIBRARY_PATH such that it includes all the paths to the dynamic libraries required above, and including as well the L1-API provided dynamic libraries.

The test tool, when executed through the command line without any inputs, will prompt the correct way of setting its parameters.

Syntax: smos-l1-api-test <Action> <Joborder> <Output Path>

Action: <1> Dual Pol Data Processing
<2> Full Pol Data Processing
<3> Correlated Noise Data Processing
<4> Uncorrelated Noise Data Processing
<5> Fringe Washing Function Shape Data Processing
<6> NIR External Calibration Data Processing
Joborder: Full path to the joborder file up to and including the joborder name
Output Path: Full output path where products shall be generated

3.2.1. Joborder structure

The joborder required by the test tool is a simple ASCII text file with a list of input files and their full file system paths. The structure of the joborder is constant, independently of the action selected for the test tool. The list of expected input files in the joborder is the following one:

1. AUX CNFL1P
2. AUX PLM
3. AUX FAIL
4. MPL ORBSCT
5. AUX_BULL_B
6. AUX_PMS
7. AUX_NIR
8. AUX_LCF
9. AUX_SPAR

10. AUX_BWGHT
11. AUX_APDNRT (or alternatively AUX_APDL/AUX_APDS)
12. AUX_BFP
13. AUX_MISP
14. AUX_DGG
15. AUX_NRTMSK (or alternatively AUX_MASK)
16. AUX_RFI
17. AUX_LSMASK
18. AUX_PATT
19. AUX_GALAXY
20. AUX_GALNIR
21. AUX_BSCAT
22. AUX_SUNT
23. AUX_MOONT
24. AUX_RFILST
25. AUX_VTEC (on any type P/R/C)
26. MIR_FTTD (for case 1) or MIR_FTTF (for case 2)
27. MIR_GMATD
28. MIR_JMATD
29. MIR_CRSD1A
29.30. MIR_CSTD1A
30.31. MIR_UAVD1A
31.32. MIR_ANIR1A
32.33. TLM_MIRA0 or TLM_MIRA1A
33.34. MIR_SC_D0 (case 1), MIR_SC_F0 (case 2), MIR_CORD0 (cases 3 and 5), MIR_UNCD0 (case 4), MIR_TARD0 (case 6)

In case a file is not applicable or not available, the keyword *NULL* shall be used to indicate to the test tool that it cannot use this file (please note that in some cases input products are mandatory, and setting *NULL* will not eliminate this dependency). The user should be knowledgeable in SMOS L1 processing, orchestration and dependencies before using this test tool

3.2.2. Example Joborder

```
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_CNFL1P_20110501T000000_20500101T000000_360_001_3.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_PLM____20070101T000000_20500101T000000_300_005_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_FAIL___20070101T000000_20500101T000000_300_002_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_MPL_ORBSCT_20091102T031142_20500101T000000_341_006_3.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_BULL_B_20100202T000000_20100301T235959_100_001_3
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_PMS____20050101T000000_20500101T000000_340_004_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_NIR____20070101T000000_20500101T000000_350_004_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_LCF____20050101T000000_20500101T000000_360_003_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_SPAR___20050101T000000_20500101T000000_340_003_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_BWGHT__20070101T000000_20500101T000000_340_002_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_APDL___20070101T000000_20500101T000000_300_001_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_BFP____20050101T000000_20500101T000000_340_001_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_MISP___20070101T000000_20500101T000000_300_002_0.EEF
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_DGG____20070101T000000_20500101T000000_300_001_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_MASK___20070101T000000_20500101T000000_300_001_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_RFI____20070101T000000_20500101T000000_300_002_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_LSMASK_20070101T000000_20500101T000000_300_002_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_PATT___20050101T000000_20500101T000000_320_003_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_GALAXY_20070101T000000_20500101T000000_300_002_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_GALNIR_20070101T000000_20500101T000000_300_001_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_BSCAT__20070101T000000_20500101T000000_300_001_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_SUNT___20070101T000000_20500101T000000_300_001_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_MOONT__20070101T000000_20500101T000000_300_001_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_AUX_RFILST_20070101T000000_20500101T000000_360_002_0.EEF
NULL
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_MIR_FTTD___20100202T170303_20100202T171723_350_001_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_MIR_GMATD__20100202T144523_20500101T000000_350_001_0
/opt4/smostest/SharedADF-L1PP-5.5.0/SM_TEST_MIR_JMATD__20100202T144523_20500101T000000_350_001_0
test/data/REAL_DATA/DUAL/SM_TEST_MIR_CRSD1A_20091208T044712_20091208T184221_321_002_0
test/data/REAL_DATA/DUAL/SM_TEST_MIR_CSTD1A_20091208T044712_20091208T184221_550_002_0
test/data/REAL_DATA/DUAL/SM_TEST_MIR_UAVD1A_20091208T141710_20091208T160003_321_001_0
test/data/REAL_DATA/DUAL/SM_TEST_MIR_ANIR1A_20100914T021114_20100914T022500_340_001_0
test/data/REAL_DATA/DUAL/SM_OPER_TLM_MIRA0__20091208T183801_20091208T184742_000_001_0
test/data/REAL_DATA/DUAL/SM_OPER_MIR_SC_D0__20091208T183801_20091208T184742_000_001_0
```

In which it can be seen that a VTEC ADF (optional input) is not available and is set to NULL, whereas the remaining inputs are defined and available.

4. L1-API DIRECTORY STRUCTURE

```
l1-api-5.5.0
├──bin
├──data
│   ├──iono_models
│   │   ├──IGRF
│   │   └──IRI
│   └──rw
├──doc
│   └──html
├──include
│   ├──common
│   ├──interfaces
│   ├──l1a_algorithms
│   ├──l1b_algorithms
│   ├──l1c_algorithms
│   └──Processor
└──lib64
```

The directory structure is populated with data from the Install Kit, and its contents are described in section 2.3.3.