

SMOS L1 Processor Prototype 6.0.0

Software Release Note

Code : SO-SRN-DME-L1PP-0270
Issue : 1.0
Date : 29/11/12

	Name	Function	Signature
Prepared by	J. Barbosa	Project Engineer	
	A. Gutiérrez	Project Engineer	
	R. Castro	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>

© DEIMOS Engenharia 2012



Code : SO-SRN-DME-L1PP-0270
SMOS L1 Processor Date : 29/11/12
Prototype 6.0.0 Issue : 1.0
Software Release Note Page : ii

This page intentionally left blank

Document Information

Contract Data	Classification
Contract Number: 4000101241/10/I-AM	Internal <input type="checkbox"/>
Contract Issuer: ESA	Public <input checked="" type="checkbox"/>
	Industry <input type="checkbox"/>
	Confidential <input type="checkbox"/>

Internal Distribution		
Name	Unit	Copies

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

Archiving	
Word Processor:	MS Word 2000
File Name:	SO-SRN-DME-L1PP-0270-L1PP-6-0-0-Software-Release-Note.doc
Archive Code:	SO-SRN-DME-L1PP-0270

Document Status Log

Issue	Change description	Date	Approved
1.0	V6.0.0 L1PP Release	2012-11-29	

Table of Contents

1. INTRODUCTION	6
1.1. Purpose and Scope	6
1.2. Acronyms and Abbreviations	6
1.3. Applicable and Reference Documents.....	7
1.3.1. Applicable Documents	7
1.3.2. Reference Documents	8
2. Version Description	9
2.1. Baseline	9
2.2. Changes Implemented	9
2.3. Differences wrt Baseline	10
2.4. Documents Delivered and Changes.....	11
2.5. Implementation of Mandatory Fields	12
2.6. Known limitations and bugs	12
2.7. Future Implementations	12
3. Installation Procedure	13
3.1. Hardware Requirements	13
3.2. Dependencies	13
3.3. Installation Kit Description.....	15
3.4. L1PP Installation Steps	15
3.4.1. Environment Variables.....	16
3.5. Launching L1PP.....	17
4. Compilation Procedure	18



4.1. Dependencies 18
4.2. Compilation 19
5. Annex A: L1PP Install Kit Contents 20
6. Annex B: L1PP Release Kit Contents..... 21
7. Annex C: L1PP 6.0.0 Usage of DPGS ADF files..... 22
8. Annex D: Implementation of mandatory fields in L1C SCIENCE Headers 23

List of Tables

Table 1: Table of Acronyms..... 6
Table 2: Applicable Documents. 7
Table 3: Reference Documents..... 8

1. INTRODUCTION

1.1. Purpose and Scope

This document provides the Software Release Note for the SMOS L1 Processor Prototype v6.0.0 and associated tools. It also details the Installation and Compilation procedures.

1.2. Acronyms and Abbreviations

CAS	Calibration System
CFI	Customer Furnished Item
COTS	Commercial Off-The-Shelf
CRSD	Correlated Noise Injection Calibration File
EE	Earth Explorer
EEFH	Earth Explorer File Handling CFI (ASCII XML library)
GUI	Graphical User Interface
HTML	HyperText Markup Language
LICEF	Light-weight Cost-Effective
L1PP	Level 1 processor prototype
NS	Noise Source
SMOS	Soil Moisture and Ocean Salinity
SPAR	S-Parameters
SVP	Software Validation Plan
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”.

1.3. Applicable and Reference Documents

1.3.1. Applicable Documents

Ref.	Code	Title	Issue
AD.1	SO-SOW-CASA-PLM-0855	Level 1 Processor Prototype Development Phase 3 and Support and Analysis Activities. Statement of Work	1.0
AD.2	EE-MA-DMS-GS-0008	EE XML/Binary CFI File Handling Library	3.7.3 (07/05/10)
AD.3	SO-DS-DME-L1PP-0007	SMOS L1 Processor L0 to L1a Data Processing Model	2.16 (29/11/12)
AD.4	SO-DS-DME-L1PP-0012	SMOS L1 Processor Prototype ADD (obsolete, integrated in DPMs)	1.3
AD.5	BinX	Editkt::BinX 1.2 Developer's Guide	1.3
AD.6	SO-TR-DME-L1PP-0018	SMOS L1 Processor Software Verification and Validation Plan (SVVP)	2.12 (29/11/12)
AD.7	SO-TN-IDR-GS-0005	SMOS L1 DPGS Product Specification	5.24 (06/12/12)
AD.8	SO-DS-DME-L1PP-0008	SMOS L1 Processor L1a to L1b Data Processing Model	2.16 (29/11/12)
AD.9	SO-DS-DME-L1PP-0009	SMOS L1 Processor L1c Data Processing Model	2.11 (29/11/12)
AD.10	SO-DS-DME-L1PP-0011	SMOS L1 Processor Algorithm Theoretical Baseline Definition	2.10 (29/10/10)
AD.11	SO-UM-DME-L1PP-0016	SMOS L1 Processor Prototype User Manual	2.18 7 (29/11/12)
<u>AD.12</u>	<u>SO-SOW-ESA-GS-6647</u>	<u>SMOS Expert Support Laboratories for the period 2010-2014 - ESL Level 1 Calibration and Reconstruction</u>	<u>1.2</u> (10/12/2009)
<u>AD.13</u>	SO-ID-IDR-GS-0016	<u>DPGS Master Interface Control Document</u>	<u>3.6</u> (18/05/2011)

Table 2: Applicable Documents.

1.3.2. Reference Documents

Ref.	Code	Title	Issue
RD₋₁	EE-MA-DMS-GS-0001	Earth Explorer Mission CFI Software MISSION CONVENTIONS DOCUMENT	1.5 (07/05/10)
RD₋₂	PE-TN-ESA-GS-0001	Earth Explorer Ground Segment File Format Standard	1.4 (13/06/03)
RD₋₃	EE-MA-DMS-GS-0002	Earth Explorer Mission CFI Software GENERAL SOFTWARE USER MANUAL	3.7.3 (07/05/10)
RD₋₄	SO-LI-CASA-PLM-0094	Directory of Acronyms and abbreviations	04 (10/11/03)
RD₋₅	SO-TR-DME-L1PP-0271	L1PP v600 Software Validation Report	1.0 (29/11/12)

Table 3: Reference Documents

2. Version Description

2.1. Baseline

This is version 6.0.0 of the SMOS L1 Processor Prototype. The L1PP has been system and scientifically tested. No unit tests were performed, since all functionality is tested with the full L0 to L1c scientific tests.

2.2. Changes Implemented

This is the fourth release of the SMOS L1 Processor Prototype Phase 6, corresponding to the build version 6.0.0. The following changes with respect to the previous release (L1PP Build 5.5.0) are included in this release.

L1PP Version: 6.0.0

CVS Tag: SMOS-P6-REL-V6_0_0-20121128

Creation Date: 2012-11-28

Corrected the following SPRs with respect to the previous release (v 5.5.0):

- SPR 581 - Pixels in Sun tails caused by Sun replicas outside the hexagon are not flagged
- SPR 580 - L1PP crashes when using L1c consolidation
- SPR 579 - TLM_MIRA1A file does not annotate the AUX_PMS in the product header
- SPR 578 - Incorrect end pole consolidation in L1PP
- SPR 577 - Sun tails flag should be expanded to the Sun BT configurable radius
- SPR 576 - Sunlint flagging is not correctly computed in FOV
- SPR 575 - Minor discrepancies in BT ($10E-5$) for the same pixel in Land and Sea products
- SPR 574 - MIR_CSTD1A is not annotated in the header of ANIR L1a Products
- SPR 573 - Wrong identification of end of product in L1a Full Pol processing
- SPR 572 - Missing -3dB radius in APDS and APDL ADFs
- SPR 571 - Calibration sequence duplication when processing overlapping products
- SPR 570 - Missing or incorrect references in L1b and L1c headers
- SPR 569 - Wrong Tsys used in radiometric accuracy computation for T3/T4

- SPR 568 - Annotation of Foreign_Sources_Correction is wrong for TARX1B files
- SPR 567 - Use of Galactic T3 component should be enabled
- SPR 566 - Border flagging in xi-eta is not working correctly

For more details on SPR correction, please refer to [RD.5]. In addition to the SPR correction, the following updates were made:

- Major updates
 - New CRSx1A format for inclusion of new External L1 Calibration data
 - Added External L1 Calibration computation and usage in science data
 - New FTTx format for inclusion of new parameters needed for Baseline Based FTT
 - Added Baseline Based FTT in L1b processing
 - Added new NIR leakage and cross-coupling computation and application, validated with EADS-CASA and HUT
 - Improved Gibbs 0 and Gibbs 1 application according to UPC recommendations
- Minor updates
 - This L1PP Installer includes the latest XML RW API and DPGS schemas available at the time of creation (v05.00.35, Release for L1PP only)

2.3. Differences wrt Baseline

This version of the L1 Prototype (6.0.0) has been delivered together with the latest documentation baseline of L1 Processing (see Section 2.4). The latest instrument and processing baseline is fully implemented with some exceptions:

- On all DPGS products, the quality information (in both headers and datablocks) is set to zero as L1PP v6.0.0 is not computing any quality indicators (except for those already present in the L0 headers which are simply transmitted onwards)
- The following changes are to be understood as L1PP “evolutions”, which are implemented and tested using L1PP framework before issuing formally a recommendation for DPGS implementation (or not). Some of them are already known not to be needed in DPGS, but are reported here for completeness:
 - Configurable option to activate Unoise correction for baselines covered only by a common Noise Source (no recommendation has yet been issued)

- Configurable option to decimate Local Oscillator sequences by configuration (not recommended for DPGS)
- Configurable option to generate and use a Hexagonally expanded G-matrix in extended sources removal (no recommendation has yet been issued)
- Configurable option to use a four point interpolation to compute the Sun BT during the Direct Sun correction.

2.4. Documents Delivered and Changes

The following table provides a list of the documents that delivered for this release and summarizes the changes introduced into the baseline for L1PP v6.0.0.

Code	Changes	Issue
SO-DS-DME-L1PP-0007 DPM L1a	Introduced External L1 Calibration Introduced new RFI Detection technique based on NIR and Tsys values	2.16
SO-DS-DME-L1PP-0008 DPM L1b	Removed generation of FTT, GMAT and JMAT Introduced Baseline Based FTT application Minor adjustments to Gibbs removal	2.16
SO-DS-DME-L1PP-0009 DPM L1c	Updated use of RFI flags inherited from L1b	2.11
SO-UM-DME-L1PP-0016 Software User Manual	Updated for L1PP V6.0.0	2.18
SO-TDD-DME-L1PP-0023 ADF Description, Table Generation Requirement Document	Moved contents from DPM L1b. Added new configuration for Sun Removal and update Baseline Weights file.	3.0
SO-TR-DME-L1PP-0018 Validation Plan	Updated for L1PP V6.0.0	2.12
SO-TDD-DME-L1PP-0272 Test Data Set Description	Test Data Set for v6.0.0	1.0
SO-TR-DME-L1PP-0271 Software Validation Report	Software Validation Report for v6.0.0 delivery	1.0

Code	Changes	Issue
SO-DS-DME-L1PP-0011	No updates	2.10
SO-TN-IDR-GS-0005 L1 Product Specification	Updated CRSx1A products with external L1 calibration. Updated SCx_1A products with RFI flag. Updated FTTx products with information per LICEF. Updated CNFL1P with new configuration parameters	5.24

2.5. Implementation of Mandatory Fields

A table reporting the mandatory fields for Level 1c product headers, can be found in Annex D. This table only reports the status of the L1C product headers. All datablock values are set according to the L1 Product Specification document.

2.6. Known limitations and bugs

As of Phase 5 (L1PP v3.1.0), only the 64bit version of the SW is supported.

There are no Software Problem Reports to be investigated in L1PP v6.0.0 (all closed).

For reporting additional SPRs, please use the Problem report form available at the L1PP project's webpage:

http://www.smos.com.pt/problem_report.html.

2.7. Future Implementations

There are no future deliveries expected for L1PP. This shall be the last official version supported.

3. INSTALLATION PROCEDURE

3.1. Hardware Requirements

L1PP may be run on any x86 architecture (PentiumIV, Xenon, AMD Opteron, ...) 64 bits with LINUX installed. The memory and disk resources needed for executing the prototype depend on the type of algorithms being used. For full functionality, the user needs the following resources:

- ❑ 18,5 GiB of Disk Space – from these 18,5GB, 16GiB are needed for generating the G-Matrix ADF¹. After the generation of the ADF, the L1PP needs around 10GiB of disk space for all the ADFs, libraries and some available space for products generation;
- ❑ Minimum 4GiB RAM - the RAM memory available will be critical when handling the G-Matrix ADF (Level 1B). If the user wants to use the G-Matrix ADF for performing the Foreign Sources Correction in Full Polarisation, a minimum of 10GiB is advisable (16 GiB recommended). If the user has less than 8 GiB of RAM, a swap partition of at least 8 GiB is also recommended as the total amount of memory needed in L1b processing may reach 10GiB for a full pol product half-orbit.
- ❑ From L1PP v5.5.0 onwards there is the possibility to generate and use an expanded G-Matrix. This implies that the user must have 41 GiB of disk space, since the generation of this expanded matrix requires ~38 GiB. The expanded matrix needs a minimum of 19 GiB of RAM memory.

As mentioned before, only the 64bit version of L1PP is maintained, which requires a 64bit HW platform to execute.

3.2. Dependencies

In order to run the SMOS L1PP, 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)
- ❑ BinaryXML File Handler 3.7 (already includes BinX 1.2.6) (<http://www.smos.esa.int/>)
- ❑ xerces 2.8.0 (<http://xml.apache.org/xerces/>)

¹ The ADF occupies around 8GB of disk space. However, since it is not provided with the Install Kit, it must be generated by the prototype (refer to [AD.11] for further details on generating the G-Matrix ADF). For generating it a minimum of 2X8GB of disk space available is needed. After the generation of the ADF, only the 8GB occupied by the ADF are needed.

- XML RW API (ftp://131.176.251.166/smos/software/XML_RW_API/)
- xerces 2.7.0 (<http://xml.apache.org/xerces/>) (needed by XML RW API)
- DOM4J 1.5.2 (<http://www.dom4j.org>)
- Jaxen 1.1 (<http://jaxen.codehaus.org>)
- Swing 1.0
- Log4c 1.2.0 (<http://log4c.sourceforge.net>)
- Java J2SE SDK 1.5.0 (<http://java.sun.com/j2se/1.5.0/download.html>)
- Lapack 3.0 (<http://www.netlib.org/lapack/>)
- Blas 3.0 (version included in Lapack 3.0)
- FFTW 3.1.2 (<http://www.fftw.org/>)

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

Nevertheless, the L1PP Install Kit already contains some of the above dynamic libraries in case the user has not installed them locally. The included libraries are:

- BinaryXML 3.7
- xerces 2.8.0 (<http://xml.apache.org/xerces2-j/>)
- XML RW API (ftp://131.176.251.166/smos/software/XML_RW_API/)
- xerces 2.7.0 (<http://xml.apache.org/xerces/>) (needed by XML RW API)
- Libxml2 2.6.16
- Log4c 1.2.0
- DOM4J 1.5.2
- Jaxen 1.1
- Swing 1.0
- Lapack 3.0
- Blas 3.0 (version included in Lapack 3.0)
- FFTW 3.1.2
- gfortran (version included in gcc 4.3)

Libraries *dom4j.jar*, *jaxen-1.1-beta-8.jar* and *swing-layout-1.0.jar* are provided in *l1pp/lib* directory (no extraction or installation required).

The rest of the libraries may need to be compiled as appropriate, so it may be possible that they need to be installed as root. Please check each installation instructions for this purpose, although for GNU packages it can be done simply by executing *./configure* and then *make*.

3.3. Installation Kit Description

The L1 Processor Prototype is provided as an InstallKit composed of a single *tgz* file. This binary distribution is delivered as *L1PP-6.0.0_Installer.tgz* which after decompressing contains:

- ❑ *InstallKitL1PP_6_0_0.readme*
- ❑ *InstallKitL1PP_6_0_0.tgz*: which contains the libraries and configuration files of the L1 prototype. The package contains inside a pre-configured structure of the L1PP working directory with the Linux 64 bits version and Mac OS X of the L1PP. All required configuration files, product schemas and header templates are also included;
- ❑ *InstallKitL1PP.sh*

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

- ❑ *ADFPackage_6_0_0.tgz*: contains all the Auxiliary Data Files needed by the prototype as well as the corresponding xml schemas and documentation;
- ❑ *TestDataPackage_6_0_0*: contains several test data scenarios (L0 products), each packed independently, corresponding configuration files in order to be used directly by the prototype. In addition, documents with the test scenarios description and with the product format specification are also provided in the same webpage².

3.4. L1PP Installation Steps

In order to install the L1PP, the user shall extract the *tgz* file and execute the install script in the command line:

```
$ tar xzvf L1PP_Installer.tgz
```

```
$ sh InstallKitL1PP.sh $INSTALL_PATH
```

Where *\$INSTALL_PATH* is the directory where the user wants to install the prototype. For example, if we want to create it in */home/smos/tools/*, the command to execute shall be:

```
$ sh InstallKitL1PP.sh /home/smos/tools/
```

² http://www.smos.com.pt/project_data_products.html

and the script will extract and create a new directory inside *tools* called *l1pp-<version>* with all the L1 Prototype contents (Libraries, configuration files, xml schemas, product headers, etc.). The script will also perform an md5 checksum for all the files contained in the distribution. For a complete list of these files and their checksum, please refer to “InstallKit.md5”, provided in the home directory of the distribution.

The installation script shall also update the configuration files being extracted so that the paths referred inside are correctly referenced to the installation directory. This is done automatically by using the script named *scripts/update-config-path.sh*. If at any time there is the need to reuse it with a default configuration file, it can be invoked on its own.

When installing a new test scenario from the provided Test Data Set (found on the L1PP SMOS webpage), the script *scripts/update-config-path.sh* will have to be re-executed for the *configurationFile.xml* and *log4crc* files inside the scenario directory.

After the installation is complete, the directories structure shown in Section 5 has been created automatically by the installation script. Nevertheless, the user must later populate manually the ADF and L0/L1 products that he intends to process into the appropriate directories.

After installing the prototype, the user will need the Auxiliary Data Files contained in *ADFPackage_6_0.tgz*. After extracting the package, a soft link pointing from *<l1pp home>/data/adf* directory to *ADF-Package_6_0/adf* has to be created by the user.

Finally the user will need to fetch the directories *<l1pp home>/data/l1a-in* , *<l1pp home>/data/l1b-in*, *<l1pp home>/data/l1c-in* with the products provided as examples on the test data webpage.

As part of the 6.0.0 delivery, the *dpgs_interface* directory is provided inside the Install Kit. This directory contains the latest status of the DPGS-V3 products and schemas (v05.00.35) at the time of the release.

3.4.1. Environment Variables

For execution of the prototype, a new environment variable named *L1PP_ROOT* must be created by the user, pointing to the *l1pp* directory. For instance, if the *\$INSTALL_PATH* is */home/smos/tools/* , the L1PP will be installed in */home/smos/tools/l1pp-v6.0.0*. Therefore *L1PP_ROOT* variable must be set by the user as */home/smos/tools/l1pp-v6.0.0/* :

```
$ export L1PP_ROOT=/home/smos/tools/l1pp-v6.0.0/
```

Remark: It is important to highlight that the environment variable *\$L1PP_ROOT* shall be always terminated with a slash (“/”) character. For instance if the L1PP is installed in */home/smos/tools/l1pp-v6.0.0* the *L1PP_ROOT* shall be */home/smos/tools/l1pp-v6.0.0/*.

It is also necessary to update the *LD_LIBRARY_PATH* environment variable with the location of the shared libraries, as follows:

```
$ export LD_LIBRARY_PATH=$L1PP_ROOT/lib64: $L1PP_ROOT/external_libs/lib64:$LD_LIBRARY_PATH
```


LD_LIBRARY_PATH shall include the directories where the different libraries are installed. Some of the external libraries are provided with the Installation Kit in the directory *L1PP_ROOT/external_libs/lib64*, while others shall be installed by the user (See Section 3.2). The path for the L1PP internal libraries (*L1PP_ROOT/lib64* – for the 64-bit version) shall also be included in LD_LIBRARY_PATH variable.

Note: If the user changes the location of the libraries, the LD_LIBRARY_PATH shall be updated accordingly. However, the files *dom4j.jar*, *jaxen-1.1-beta-8.jar* and *swing-layout-1.0.jar*, provided in *\$L1PP_ROOT/lib* shall always be kept in this directory.

These two variables are updated if the prototype is executed with the scripts provided in the installation package. If the user wants to use a different execution method, these variables must be defined. In order to avoid overriding the user's local libraries, the L1PP will only use the provided additional libraries in case it cannot find them in the user-defined path.

3.5. Launching L1PP

The prototype may be started in 64bits by running the script *run-l1pp64.sh*, as follows:

```
$ sh ./run-l1pp64.sh
```

This script is provided as part of the installation package and is configured also during installation time. If the user moves the L1PP directory to another location, it shall be required to verify the correctness of the scripts.

The following type of output should be displayed right after execution:

```
Set L1PP_ROOT=/home/smolest/11pp/
Set LD_LIBRARY_PATH=/home/smolest/11pp/lib:/usr/local/lib:/home/smolest/11pp/external_libs/lib
Set XML_RW_API_HOME=/home/smolest/11pp/dpgs_interface/
#####
Starting the L1PP GUI
#####
SMOS Level 1 Processor Prototype v6.0.0
Developed by Deimos Engenharia & Critical Software S.A.
Under contract of EADS CASA Espacio and ESA
```

In order to run in text mode, in which L1PP runs directly from the terminal without any HMI, the following command shall be executed:

```
$ sh ./run-l1pp64.sh -text
```

4. Compilation Procedure

The following sub-sections describe a set of procedures required to compile the L1 Processor Prototype and BinaryXML from the Release Kit.

4.1. Dependencies

In order to compile the BinaryXML library from the Release Kit, the following software must already be installed in the target platform:

- gcc 4.3 (<http://gcc.gnu.org>)
- Earth Explorer CFI 3.7.3 (<http://eop-cfi.esa.int/>)
- Libxml2 2.6.16 or higher
- BinX 1.2.6 (<http://www.edikt.org/binx>)
- xerces 2.8.0 (<http://xml.apache.org/xerces2-j/>) - required by BinX

In order to compile the SMOS L1PP, the following software must already be installed in the target platform:

- GCC 4.3 (<http://gcc.gnu.org>)
- glibc 2.3.4 (<http://gcc.gnu.org>)
- Earth Explorer CFI 3.7.3 (which already includes libxml2 2.6.16) (<http://eop-cfi.esa.int/>)
- BinaryXML File Handler 3.7 (<http://www.smos.esa.int/>)
- FFTW 3.1.2 (<http://www.fftw.org/>)
- BLAS v3.0 (<http://www.netlib.org/blas/>) (included in LAPACK distribution)
- LAPACK v3.0 (<http://www.netlib.org/lapack/>)
- DOM4J 1.5.2 (<http://www.dom4j.org>)
- Jaxen 1.1 (<http://jaxen.codehaus.org>)
- Log4c 1.2.0 (<http://log4c.sourceforge.net>)
- j2sdk1.5.0 (<http://java.sun.com/j2se/1.4.2/download.html>)
- XML RW API (ftp://131.176.251.166/smos/software/XML_RW_API/)

- ❑ xerces 2.7.0 (<http://xml.apache.org/xerces/>) (needed by XML RW API)

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

When all external libraries are installed, it is necessary to create some soft links to the installed libraries:

- ❑ create a link from the Explorer CFI home directory to "/opt/cfi"
- ❑ create a link from the BinX home directory to "/opt/binx"
- ❑ create a link from the xerces-c home directory to "/opt/xerces-c"
- ❑ create a link from the binxml-fh home directory to "/opt/binxml-fh"
- ❑ create a link from the XML RW API home directory to "/opt/xml_rw"

e.g.: An example for creating a link for the BinX library installed in the user's home directory would be:

```
$ ln -s ~/binx1.2 /opt/binx
```

If these links cannot be created, then the Makefile inside the 11pp directory should be updated with the paths to the different libraries by editing the first lines in the file. It should also be verified that the paths to the include and lib files are correct within the Makefile. It should also be verified that it has the permissions to access the libraries.

These instructions for compilation are valid for any OS (Linux, MacOS), as they are the paths where the Makefile expects to find the libraries.

For execution of the prototype, it is also necessary to update the LD_LIBRARY_PATH environment variable with the location of the shared libraries, as follows:

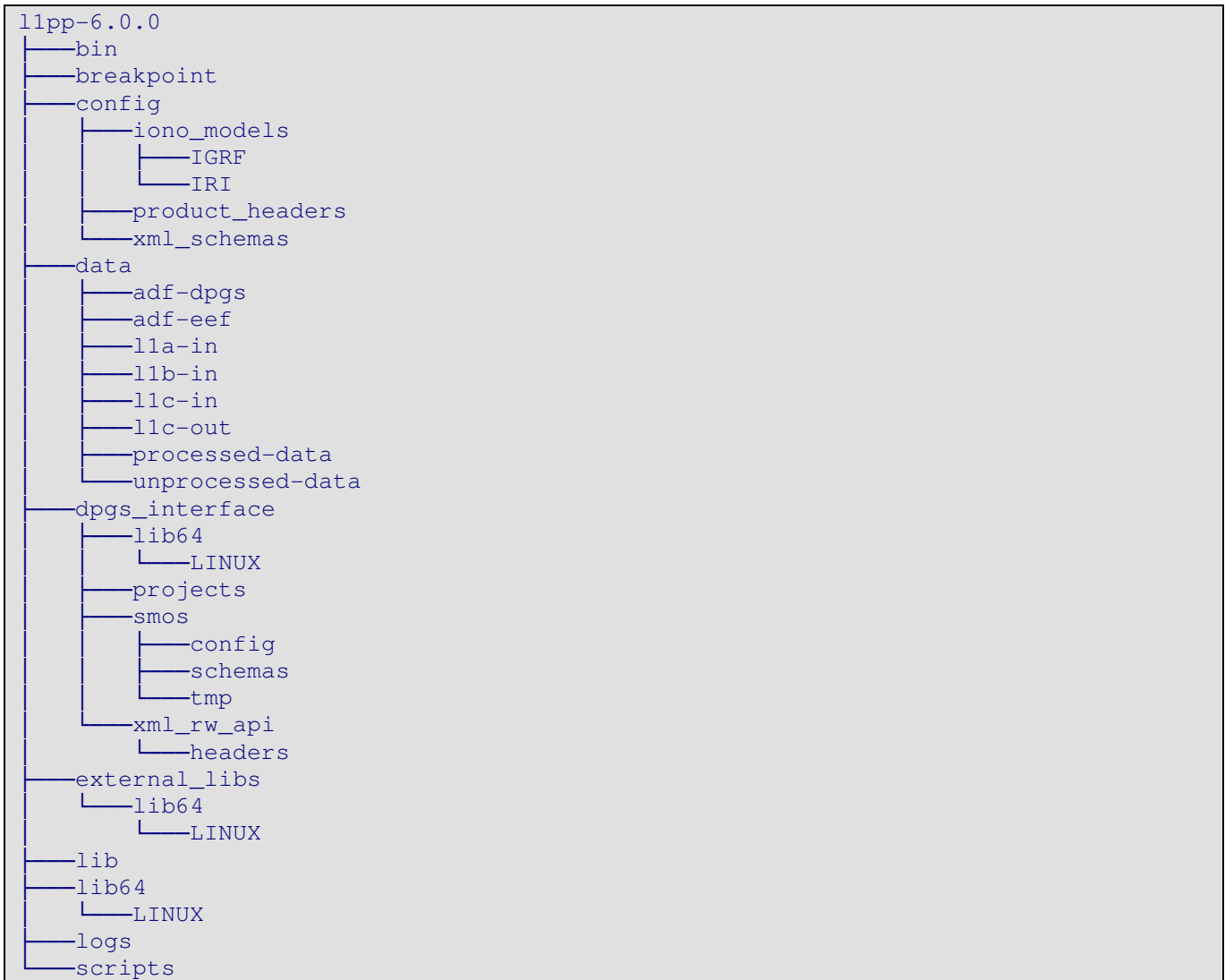
```
$ export LD_LIBRARY_PATH=/opt/xerces-c/lib:/opt/binxml-fh/lib:  
/opt/cfi/aux_tools/libxml/LINUX/lib:$LD_LIBRARY_PATH
```

4.2. Compilation

First extract the source code from the supplied *ReleaseKit* (see Annex B). The command “make clean all” shall initiate the compilation of all libraries and executables. Compilation on 32 or 64bits can be toggled by commenting out the variable ARCH64 inside the Makefile.

5. ANNEX A: L1PP INSTALL KIT CONTENTS

In the figure below, the directory structure is shown. For a full listing of the L1PP files, please consult “Installkit.md5”, provided in the binary distribution home directory (*l1pp-6.0.0*).



6. ANNEX B: L1PP RELEASE KIT CONTENTS

This is the directory structure for the release Kit for the L1PP source release kit. Please note that source code is provided “as is”, as per the Software License Agreement and that no support is available for the usage of its contents (refer to LICENSE and README files inside *l1pp-src-6.0.0* directory).

```
l1pp-src-6.0.0
├── bin
├── breakpoint
├── config
│   ├── iono_models
│   │   ├── IGRF
│   │   └── IRI
│   ├── product_headers
│   └── xml_schemas
├── data
│   ├── adf-dpgs
│   ├── adf-eef
│   ├── l1a-in
│   ├── l1b-in
│   ├── l1c-in
│   ├── l1c-out
│   ├── processed-data
│   └── unprocessed-data
├── dpgs_interface
│   ├── lib
│   ├── lib64
│   ├── smos
│   │   ├── config
│   │   ├── schemas
│   │   └── tmp
│   ├── VEGA
│   │   ├── include
│   │   ├── lib
│   │   ├── lib64
│   │   └── src
│   └── xml_rw_api
│       └── headers
├── include
│   ├── core_components
│   ├── error_correction
│   ├── geolocation
│   ├── image_reconstruction
│   ├── ionospheric_correction
│   └── llp_orchestrator
├── lib
├── lib64
├── logs
├── scripts
└── src
    ├── core_components
    ├── error_correction
    ├── geolocation
    ├── image_reconstruction
    ├── ionospheric_correction
    └── llp_orchestrator
```

7. ANNEX C: L1PP 6.0.0 USAGE OF DPGS ADF FILES

Product	Header	Datablock
APDL /APDS	Not used	Read
BFP		
BWGHT		
BSCAT		
DGG		
FAIL		
GALAXY		
GALNIR		
LCF		
LSMASK		
MASK		
MISP		
MOONT		
NIR		
PATT		
PLM		
PMS		
RFI		
RFILST		
SPAR		
SUNT		
VTEC		

8. ANNEX D: IMPLEMENTATION OF MANDATORY FIELDS IN L1C SCIENCE HEADERS

Item	L2 Usage	Value	Notes
Earth_Explorer_Header	L2 uses this value		Our headers are in line with the specification. The value defined by the schema is the one we use.
File_Name	L2 uses this value		This value is copied from the internal data-structure that holds the actual product name. The Site Instance value is always set to 0 in L1PP.
File_Description			This value is hard-coded as specified in the documentation.
Notes		x	No value is being written, but the schemas' default is 'x'.
Mission	L2 uses this value	SMOS	It is always set to 'SMOS'
File_Class	L2 uses this value		This value is copied from the product filename.
File_Type	L2 uses this value		This value is copied from the product filename.
Validity_Start	L2 uses this value		The value that is used to generate the product name is also used to calculate this value.
Validity_Stop	L2 uses this value		The value that is used to generate the product name is also used to calculate this value.

Item	L2 Usage	Value	Notes
File_Version	L2 uses this value		This value is copied from the product filename.
System		L1PP	Value obtained from L1PP configuration file.
Creator		L1PP	Always set to 'L1PP'
Creator_Version			This value is set to the version of the L1PP. This same version is used to generate the product name.
Creation_Date			This value is generated when writing the header.
Ref_Doc		SO-TN-IDR-GS-0005	Always set to 'SO-TN-IDR-GS-0005'
Acquisition_Station	L2 uses this value		L1PP propagates the values read from the previous levels.
Processing_Centre		DMEP	Value obtained from L1PP configuration file.
Logical_Proc_Centre		DME	Value obtained from L1PP configuration file.
Orbit_Information	L2 uses this value		All <i>Orbit_Information</i> values are copied from previous levels products
Phase	L2 uses this value		This value is read from previous level products (L0) and propagated.

Item	L2 Usage	Value	Notes
Cycle	L2 uses this value		This value is read from previous level products (L0) and propagated.
Rel_Orbit	L2 uses this value		This value is read from previous level products (L0) and propagated.
Abs_Orbit	L2 uses this value		This value is read from previous level products (L0) and propagated.
OSV_TAI/UTC/UT1	L2 uses this value		This value is read from previous level products (L0) and propagated.
Leap_Second	L2 uses this value		This value is read from previous level products (L0) and propagated.
X_/Y_/Z_Position	L2 uses this value		This value is read from previous level products (L0) and propagated.
X_/Y_/Z_Velocity	L2 uses this value		This value is read from previous level products (L0) and propagated.
Vector_Source	L2 uses this value		This value is read from previous level products (L0) and propagated.
Product_Confidence			This value is set according to the specification, however the error counters are not updated during the processing, so unless previous levels have a value different than 0, the Product_Confidence field will always be set to 'NOMINAL'.
SPH_Descriptor			Value hard-coded according to specification.
<i>Time_Info</i>			
Precise_Validity_Start	L2 uses this value		It is computed for each header.

Item	L2 Usage	Value	Notes
Precise_Validity_Stop	L2 uses this value		It is computed for each header.
Abs_Orbit_Start	L2 uses this value		It is computed for each header.
Start_Time_ANX_T	L2 uses this value		It is computed for each header.
Abs_Orbit_Stop	L2 uses this value		It is computed for each header.
Stop_Time_ANX_T	L2 uses this value		It is computed for each header.
UTC_at_ANX	L2 uses this value		It is computed for each header.
Long_at_ANX	L2 uses this value		It is computed for each header.
Ascending_Flag	L2 uses this value		This value is copied from previous levels. If this value is not available the value is set to 'A'
Semiorbit_Start_Time			This value is copied from previous levels. If this value is not available the value is set to 'UTC=2007-01-01T00:00:00.000000'
Semiorbit_Stop_Time			This value is copied from previous levels. If this value is not available the value is set to 'UTC=2007-01-01T00:00:00.000000'
Correlator_Layer			This value is copied from previous levels. If this value is not available the value is set to 'N'
Checksum			The value is correctly computed using the XML_RW methods.
Header_Schema	L2 uses this value		This value is correctly computed using the XML_RW methods.
Datablock_Schema	L2 uses this value		This value is correctly computed using the XML_RW methods.
Header_Size			The value is correctly computed.
Datablock_Size			The value is correctly computed.
HW_Identifier		L1PP	The value is always set to 'L1PP'

Item	L2 Usage	Value	Notes
<i>Quality_Information</i>			
Software_Error_Counter		0 (schema default)	Currently not available
Instrument_Error_Counter		0 (schema default)	Currently not available
ADF_Error_Counter		0 (schema default)	Currently not available
N_Discarded_Scenes		0	This value is not correctly computed, an internal SPR has been opened.
Error_Counter		0	Currently not available
N_Invalid_Blocks		0 (schema default)	Currently not available
N_Missing_Packets		0 (schema default)	Currently not available
N_Missing_Blocks		0 (schema default)	Currently not available
Reference Datasets (ALGORITHM_CONFIG_FILE)			This value is always set to blanks in L1PP, since this file does not exist in L1PP.
Product Location			
Start_Lat	L2 uses this value		This value is correctly computed.
Start_Lon	L2 uses this value		This value is correctly computed.
Stop_Lat	L2 uses this value		This value is correctly computed.
Stop_Lon	L2 uses this value		This value is correctly computed.
Mid_Lat	L2 uses this value		This value is correctly computed.

Item	L2 Usage	Value	Notes
Mid_Lon	L2 uses this value		This value is correctly computed.
Radiometric_Accuracy_Scale	L2 uses this value	50	We use the reference value presented by the reference document. This value is taken from L1PP configuration file.
Pixel_Footprint_Scale	L2 uses this value	100	We use the reference value presented by the reference document. This value is taken from L1PP configuration file.
Percentage_of_Mixed_Pixels			This value is correctly computed.
Apodisation_Window		001	The value is not computed in L1PP, and it is always set to '001' - Blackman Window.
Total_Num_Grid_Points			This value is correctly computed.