

EARTH OBSERVATION MISSION CFI SOFTWARE

Release Notes –Version 4.15

1 INTRODUCTION

This document describes the changes introduced in this release of the Earth Observation Mission CFI Software.

2 USER SUPPORT

For any question related to the usage of the EOCFI or to report a problem, please contact:

EOCFI Software Support Team

e-mail: cfi@eopp.esa.int

3 RELEASE DESCRIPTION

3.1 Software

The following table lists the released libraries, their version and issue date:

Library Name	Version	Issue Date
File Handling	4.15	20 April 2018
Data Handling	4.15	20 April 2018
Lib	4.15	20 April 2018
Orbit	4.15	20 April 2018
Pointing	4.15	20 April 2018
Visibility	4.15	20 April 2018
EECommon (*)	4.15	20 April 2018

(*) only C++ and JAVA APIs

The core API of the above libraries is written in C and provides an API for C, C++ and JAVA.

The libraries installation packages are available for download at the following URL (registration required):

<http://eop-cfi.esa.int/index.php/mission-cfi-software/eocfi-software/branch-4-x/eocfi-v4x-download>

3.2 Documentation

The following documents are available:

Type	Document Name	Version
General	Mission Conventions Document	4.15
General	General Software User Manual	4.15
C API	Quick Start Guide	4.15
C API	File Handling Software User Manual	4.15
C API	Data Handling Software User Manual	4.15
C API	Lib Software User Manual	4.15
C API	Orbit Software User Manual	4.15
C API	Pointing Software User Manual	4.15
C API	Visibility Software User Manual	4.15

The documentation is available for download (and on-line browsing for C++ and JAVA APIs) at the following URL:

<http://eop-cfi.esa.int/index.php/mission-cfi-software/eocfi-software/branch-4-x/eocfi-v4x-documentation>

More information on the Earth Observation CFI Software can be found at the following URL:

<http://eop-cfi.esa.int/index.php/mission-cfi-software/eocfi-software>

The EO CFI Software file formats are specified in the EO Mission Software File Format Specification document, which is available at the following URL:

http://eop-cfi.esa.int/Repo/PUBLIC/DOCUMENTATION/SYSTEM_SUPPORT_DOCS/PE-ID-ESA-GS-584-1.2-EO_Mission_SW_File_Format_Specs.pdf

Note: In Section 3.2 of EO CFI File Format Specification (Orbit Scenario File), the element <ANX_Longitude_Drift> and its contents are not supported by the latest EOCFI SW version.

3.3 Supported platforms

The following platforms are supported by this release of the CFI

(the following are requirements for the **C API**):

- **LINUX64_LEGACY**

- LINUX 64-bit (Legacy)
- Platform Requirements: x86_64 based PC, Linux Operating System (Kernel version 2.6.x)
- Software Requirements: gcc compiler version 4.5.x, glibc (C Library) version 2.12 (*)

(*) According to gcc documentation, forward compatibility is ensured up to gcc/g++ version 4.9.x.

- **LINUX64**

- LINUX 64-bit
- Platform Requirements: x86_64 based PC, Linux Operating System (Kernel version 4.10.x)
- Software Requirements: gcc compiler version 6.3.x, glibc (C Library) version 2.24 (*)

- **WINDOWS64**

- Microsoft WINDOWS PC (64-bit)
- Platform Requirements: x86_64 based PC, Microsoft Windows 7 Operating Systems.
- Software Requirements: Microsoft Visual C++ Compiler (Visual Studio 2017 Express or Professional edition, 64-bit)

- **MACIN64**

- MACOSX on Intel (64-bit)
- Platform Requirements: x86_64 based Mac Computer, Mac OS X version 10.12.x (Sierra)
- Software Requirements: Xcode 9.2 with clang compiler front end (gcc is an alias for clang)

NOTE for MACIN64 platform:

As of version 5 of Xcode the default compiler is clang (see <http://clang.llvm.org/>). clang is a compiler front end for C and C++ and can build an application linking against the EOCFI C / C++ libraries.

The gcc and g++ program provided within Xcode are aliases for clang.

OpenMP is not supported in clang. Therefore, the `-fopenmp` switch shall not be used. Functions using parallelized computations, e.g. `xp_target_list...` functions will operate in single-threading mode.

The following are additional requirements for the **C++ API** (a C++ compiler is required):

- g++ compiler version 4.5.x for LINUX64_LEGACY (*)
(in MACIN64, g++ is an alias for clang) and g++ compiler version 6.3.x for LINUX64 (*)

- Microsoft Visual C++ Compiler (Visual Studio 2017 Express or Professional edition) for WINDOWS

The following are additional requirements for the **JAVA API** (a JAVA SDK is required):

- Java Standard Edition (SE) version 8 for all platforms

3.4 Installation Packages

The CFI libraries are provided as zip packages:

API	Package Name	MD5 Checksum
C	EOCFI-4.15-CLIB-LINUX64.zip	7d97c8fd30928bde3685a0fc5265a953
C	EOCFI-4.15-CLIB-LINUX64_LEGACY.zip	d6005ff94a0cd9fcd4fbf31eb9bd3bcd
C	EOCFI-4.15-CLIB-MACIN64.zip	3ee56104794be0524941abf54ab3c1ab
C	EOCFI-4.15-CLIB-WINDOWS64.zip	cd1b2a1a3a234acf43f5cd0f613af8c7
C++	EOCFI-4.15-CPPLIB-LINUX64.zip	72ba012fde8cdea210143f4d25cecf4e
C++	EOCFI-4.15-CPPLIB-LINUX64_LEGACY.zip	b826a88311fa55ef8fc3b2f05e19937f
C++	EOCFI-4.15-CPPLIB-MACIN64.zip	127fa6b808bb7a775b4dfcded6b3385a
C++	EOCFI-4.15-CPPLIB-WINDOWS64_DLL.zip	0f15f98152717b6bee35b75f6c8c92d4
C++	EOCFI-4.15-CPPLIB-WINDOWS64_STA.zip	c443f8e2a06687e1124031944e2c3d84
JAVA	EOCFI-4.15-JAVALIB-LINUX64.zip	c03f6b92509ab157f45cdebd2b30d9a3
JAVA	EOCFI-4.15-JAVALIB-LINUX64_LEGACY.zip	b672f8e52606959e801850311e7de6da
JAVA	EOCFI-4.15-JAVALIB-MACIN64.zip	5feb30db0391b0ba2c9dd0c26005637
JAVA	EOCFI-4.15-JAVALIB-WINDOWS64.zip	18c32c8dd4cb05ae03c866d30f0ad637

(*) Dynamic libraries (DLLs)

(**) Static libraries

Information on how to get and use the supported DEM datasets can be found at the following URL:

<http://eop-cfi.esa.int/index.php/mission-cfi-software/eocfi-software/support-files>

Note: the package EOCFI-4.15-JAVALIB-MACIN64.zip was re-packaged after the original release due to a problem related to rpath definitions in the native dynamic libraries.

3.5 Installation Hints

The CFI libraries can be installed by expanding the installation package in any directory.

For specific hints related to the usage of the libraries, please consult:

- Section 6 “CFI LIBRARIES INSTALLATION” of the General SUM;
- Section 6 “LIBRARY USAGE” of each Library User Manual.

The EOCFI libraries make use of the following third-party libraries:

- pthreads (POSIX threads): this library is normally pre-installed in Linux and Mac OS X systems. For Windows, the library is provided in the `cfi_tools` directory within the distribution package. Pthreads is covered by the GNU Lesser General Public License. (see <https://www.gnu.org/licenses/old-licenses/lgpl-3.0.html>).
- libxml2 (see <http://xmlsoft.org/>): for reading and writing XML files.
- libgeotiff (see <http://trac.osgeo.org/geotiff/>), libtiff (see <http://www.libtiff.org/>), libproj (see <http://trac.osgeo.org/proj/>): for reading ASTER GDEM files.

Please refer to Section 6 of the User Manual of each Library for specific usage instructions. Terms and conditions for usage of such libraries are detailed in the text file (included in the distribution package) TERMS_AND_CONDITIONS.TXT.

libxml2, libgeotiff, libtiff and libproj are provided:

- in the C API distribution packages: as separated static libraries (see Section 6 of each User Manual for instruction on how to link them to the application program).
- in the C++ / Java APIs distribution packages: as separated dynamic libraries (see Section 6 of each User Manual for instruction on how to link them to the application program). In the Java API for MAC OS X platform, due to incompatibilities with system libraries, they are instead embedded in the EOCFI libraries.

User applications using the Pointing library need to be built with OpenMP support (adding `-fopenmp` switch in gcc, see Section 6 of the Pointing User Manual).

OpenMP is not supported in clang (Mac OS X) and Visual Studio (Windows), therefore no additional switch is required. Functions will operate in single-threading mode.

The XML validation function and tool in the Data Handling library uses the libxml2 library. For Windows platforms, it is required to link the user application against the `ws2_32.lib`.

4 NEW FEATURES

The following sections describe the new features introduced in this release.

The description refers to the C API. Equivalent features and methods are available in the C++ and JAVA APIs. For further details on the presented features, the user may want to refer to the User Manual of the related library.

- **Deprecated platforms:**
 - LINUX 32-bit (Legacy, Kernel version 2.6x, gcc 4.2.x, glibc 2.7)
 - LINUX 64-bit (Legacy, Kernel version 2.6x, gcc 4.2.x, glibc 2.7)
 - WINDOWS PC (32-bit)

5 SOLVED PROBLEMS

The following Anomalies have been solved:

ANR Id	Description
649	ROF generation with TLE is not allowed when time file is used
711	xd_mxl_validate: segmentation fault when schema is given as an absolute path in the file header
712	xp_get_attitude_data does not (and it shall) raise an error if nominal attitude is not initialised
727	xl_time_processing_to_ascii: Internal Error "Output ascii day is out of range"
730	targetListInter: Outliers with ACE2 DEM

6 KNOWN PROBLEMS

The updated list of known issues that will be resolved in a future release can be found at the following URL:

<http://eop-cfi.esa.int/index.php/mission-cfi-software/eocfi-software/branch-4-x/known-issues-branch-4>