

EARTH OBSERVATION MISSION CFI SOFTWARE

Release Notes – Version 4.24

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

Visit us at <http://eop-cfi.esa.int/index.php/mission-cfi-software/eocfi-software> for more.

1 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

2 NEW FEATURES & IMPROVEMENTS

| Ref./EOCFI-ANR- | Description |
|-----------------|-------------------------------------------------------------------------------------|
| 776 | Enabled support for pre-computed maximum altitude <i>mini-tiles</i> for Generic DEM |
| 908 | Enabled support for MS Visual Studio 2022 + Windows 10 |
| 979 | Enabled reading OEM <i>Useable Start Time</i> and <i>Useable Stop Time</i> fields |
| 980 | Enable the use of multiple AEM files to initialize Attitude |

3 SOLVED PROBLEMS

| Ref./EOCFI-ANR- | Description |
|-----------------|---------------------------------------------------------------------------------------|
| 882 | Corrected EOCFI usage in MATLAB Technical Note |
| 942 | Corrected memory leaks in <code>xv_stationvistime_compute</code> when using SDF |
| 947 | Added missing information in C++ API documentation |
| 972 | Corrected use of symbolic links for 3 rd party libraries in EOCFI packages |
| 974 | Corrected calculation of semi-major axis and inclination when MLST drift != 0 |
| 975 | Corrected generation of OSF based on input inclination |
| 976 | Corrected <code>xo_orbit_init_def</code> for orbits based on INCLINATION drift mode |
| 983/985 | Corrected multiple typos in error messages |
| 984 | Corrected multiple documentation issues |
| 994 | Corrected the handling of multiple OS line endings when reading TLE files |
| 1000 | Corrected leap second interpretation in IERS bulletin B |
| 1002 | Corrected value of enumerate for IERS bulleting in EOCFI Java |
| 1005 | Corrected the access to model data in EOCFI Java |

4 RELEASE DESCRIPTION

4.1 Software

Earth Observation Mission CFI Software 4.24 is composed of the following libraries:

| Library Name | Version | Issue Date |
|---------------|---------|------------|
| File Handling | 4.24 | 12/12/2022 |
| Data Handling | | |
| Lib | | |
| Orbit | | |
| Pointing | | |
| Visibility | | |
| EECommon (*) | | |

(*) 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>

4.2 Documentation

The following documents are available:

| Type | Document Name | Version |
|---------|------------------------------------|---------|
| General | Mission Conventions Document | 4.24 |
| | General Software User Manual | |
| C API | Quick Start Guide | |
| | File Handling Software User Manual | |
| | Data Handling Software User Manual | |
| | Lib Software User Manual | |
| | Orbit Software User Manual | |
| | Pointing Software User Manual | |
| | Visibility Software User Manual | |

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

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

The Earth Observation Mission CFI Software file formats are specified in the EO Mission Software File Format Specification document, which is available in [Mission SW File format Specification \(ref. PE-ID-ESA-587\)](#).

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.

4.3 Supported platforms

The following platforms are supported by this release of the CFI (the following are requirements for the **C API**):

| Designation | Platform/ Architecture | Minimum Platform Requirements | Software Requirements |
|----------------|---------------------------|----------------------------------------------------------------------|-------------------------------------------------------------------------------------------------|
| LINUX64_LEGACY | Linux 64-bit | x86_64 based PC Linux Operating System (Kernel version 2.6.x) | GCC compiler version 4.5.x glibc (C Library) version 2.12 (*) |
| LINUX64 | Linux 64-bit | x86_64 based PC Linux Operating System (Kernel version 4.10.x) | GCC compiler version 6.3.x glibc (C Library) version 2.24 |
| WINDOWS64 | Windows 64-bit | x86_64 based PC Microsoft Windows 7 | Microsoft Visual C++ Compiler (Visual Studio 2017 Express or Professional edition, 64-bit) |
| WINDOWS64-V10 | Windows 64 bit | x86_64 based PC Microsoft Windows 10 | Microsoft Visual C++ Compiler (Visual Studio 2022 Community or Professional edition, 64-bit) |
| MACIN64 | MacOS/Intel 64-bit | x86_64 based Mac Computer Mac OS X version 10.12.x (Sierra) | Xcode 9.2/Clang compiler frontend |

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

NOTE for MACIN64 distribution:

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 AppleClang. Therefore, the `-fopenmp` compiler option cannot 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

4.4 Distribution Packages

The Earth Observation Mission CFI Software libraries are provided as Zip archives:

| API | Package Name | MD5 Checksum |
|------|---------------------------------------------------|----------------------------------|
| C | EOCFI-4.24-CLIB-LINUX64.zip | 5b2e4b4f407a42094df33c707a6056d1 |
| C | EOCFI-4.24-CLIB-LINUX64_LEGACY.zip | d6bbcf81f1e50303f7097175275b7c26 |
| C | EOCFI-4.24-CLIB-MACIN64.zip | 8020ca4a104c36458708dfb14b563279 |
| C | EOCFI-4.24-CLIB-WINDOWS64.zip | 4ef4186c9e1e0e437c1f83fde1897d31 |
| C | EOCFI-4.24-CLIB-WINDOWS64-V10.zip | 8ed72d24f9d98d4024e1ce9c17d24e61 |
| C++ | EOCFI-4.24-CPPLIB-LINUX64.zip | 75868f385dc5317e95aec370c7d8cdb |
| C++ | EOCFI-4.24-CPPLIB-LINUX64_LEGACY.zip | 212b9ce71c392dae26cec259bd2106c4 |
| C++ | EOCFI-4.24-CPPLIB-MACIN64.zip | 8d1126fe23ecb53a1d7b16a090df41e7 |
| C++ | EOCFI-4.24-CPPLIB-WINDOWS64_DLL.zip (*) (***) | aade4915902616a7ab3cd9b85d90a121 |
| C++ | EOCFI-4.24-CPPLIB-WINDOWS64_STA.zip (**)(***) | 0ec8ee87e4d62d34c5dbb16517936bd7 |
| C++ | EOCFI-4.24-CPPLIB-WINDOWS64_DLL-V10.zip (*) (***) | de6d1cb45c09b05af3b0e57b10bebae2 |
| C++ | EOCFI-4.24-CPPLIB-WINDOWS64_STA-V10.zip (**)(***) | 4445b64feb8e833ea7f6c7e500b63c83 |
| JAVA | EOCFI-4.24-JAVALIB-LINUX64.zip | 6a09cb734ff4ec1d255d083467869071 |
| JAVA | EOCFI-4.24-JAVALIB-LINUX64_LEGACY.zip | d6afc6a30b5478c9bdc14b35e356c50e |
| JAVA | EOCFI-4.24-JAVALIB-MACIN64.zip | 5ae1765d3d86d016156fd06851a97cc7 |
| JAVA | EOCFI-4.24-JAVALIB-WINDOWS64.zip | 086d5f27459d60b6fdd38a15a70f27b7 |
| JAVA | EOCFI-4.24-JAVALIB-WINDOWS64-V10.zip | 5377162af2eec43c77758de08f138d82 |

(*) Dynamic libraries (DLLs)

(**) Static libraries

(***) Debug Package will be made available to interested users (please contact support for more information)

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>

4.5 Installation Hints

To install Earth Observation Mission CFI Software libraries, simply extract the contents of the distribution package in the desired installation directory. More information on how to install and use the libraries can be found on:

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

The Earth Observation Mission CFI Software makes 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.sourceware.org/pthreads-win32/copying.html>).
- libxml2 (see <http://xmlsoft.org/>): for reading and writing XML files.
- libgeotiff (see <https://trac.osgeo.org/geotiff/>)
- libtiff (see <http://www.libtiff.org/>)
- libproj (see <https://proj.org/>): for reading ASTER GDEM files.

Terms and conditions for usage of such libraries are detailed in the text file (included in the distribution package) `TERMS_AND_CONDITIONS.TXT`.

The libraries `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 AppleClang (Mac OS X) and Visual C++ (Windows), therefore no additional switch is required. In these platforms the library 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 with the `ws2_32.lib`.

5 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>