Cookie-cutter User's Guide

Fabrice Ducos
ICARE
Data Management and Processing Center

Université des Sciences et Technologies de Lille
59655, Lille
France
+33 (0)3 20 33 59 79
fabrice.ducos@icare.univ-lille1.fr

1. Introduction

The Cookie-cutter project is born from the need to compare data from various moving satellites to those from the SEVIRI instrument over geostationnary satellite MSG. The goal was to obtain, for each acquisition day of a given moving satellite, mosaics of SEVIRI slot extractions (the *cookies*) in best temporal coincidence with the moving platform, in order to allow direct comparisons of data.

Cookie-cutter is a try to achieve this goal. Given a set of some user-specified informations about the moving satellite orbit (period, inclination, node longitudes and hours), the software will generate files in HDF format containing all SEVIRI channels, where only the best coincident data (7'30 at worst, half of a slot duration) are kept, on a one-file-per-day basis. In order to facilitate data exploration, an image file (browse) is produced with each HDF data file.

This document will guide you through the simple steps of installing, configuring and running cookie-cutter on your workstation.

METEOSAT-8 extraction over PARASOL track, on 1st june 2005.

2. Installation

First, make sure that the last release of HDF (ftp://ftp.ncsa.uiuc.edu/HDF/HDF/HDF_Current/src) (cautious: not HDF5) library is installed on your system. cookie-cutter should be built with at least release 4.2r1 of HDF library.

In order to build and install properly the HDF Library on your system, please do the following operations:

```
$ tar xzf HDF4.2r1.tar.gz
$ cd HDF4.2r1
$ ./configure
$ make
$ make install
```

At this step a new sub-directory should have appeared: NewHDF (at least with the HDF4.2r1 release)
You must be super-user from this point:

```
$ su
```

(you'll be prompted here for the super-user's password)

```
# mv NewHDF /usr/local/hdf4.2r1
# cd /usr/local
# ln -s hdf4.2r1 hdf
# cd lib ; ln -s ../hdf/lib hdf ; cd ..
# cd include ; ln -s ../hdf/include hdf ; cd ..
# exit
```

Of course you're free to install the library where you want to, but the current release of cookie-cutter
expects to find these soft links at build-time. If you don't want to (or can't) create them, you'll have to
modify HDF_INC and HDF_LIB values in the following `Makefiles` of cookie-cutter (see below):

```
[user@hostname cookie-cutter]$ pwd
/path/to/cookie-cutter
[user@hostname cookie-cutter]$ find . -name Makefile
./src/seviri_utils/Makefile
./src/Makefile
./src/geostat/Makefile
./src/xrit/Makefile
./src/sds2ppm/Makefile
./src/Hdf/Makefile
./Makefile
```

Of course this is quite awkward and messy. A future release of cookie-cutter may test and use HDF_INC
and HDF_LIB environment variables shared between all the `Makefiles` to build the software.

Get the latest release of **cookie-cutter** on the icare web site
(http://www.icare.univ-lille1.fr/projets/cookie-cutter/releases/cookie-cutter.tar.gz).

Go to the root directory of cookie-cutter and build the project with **make**:

```
[user@hostname cookie-cutter]$ pwd
/path/to/cookie-cutter
[user@hostname cookie-cutter]$ make
```

The project should be built. If the build process fails, check that you correctly installed the HDF Library (and configured cookie-cutter' Makefiles if HDF is installed in a non-standard emplacement). See above for details.

3. Configuration

cookie-cutter needs the variable COOKIE_CUTTER_ROOT to be defined in its execution environment. In a Bourne or Korne derived shell (sh, ksh, bash, zsh), define it as follows:

```
[user@hostname cookie-cutter]$ COOKIE_CUTTER_ROOT=/path/to/cookie-cutter
[user@hostname cookie-cutter]$ export COOKIE_CUTTER_ROOT
```

Be sure to record the environment variable into a persistent configuration file: `/etc/profile`, `.profile`, `.bash_profile` ... depending on your installation and shell. To test your configuration, execute the **cookie-cutter** command at your shell prompt:

```
[user@hostname cookie-cutter]$ pwd
/path/to/cookie-cutter/bin
[user@hostname cookie-cutter]$ ./cookie-cutter
```

If **cookie-cutter** complains without printing its usage on the terminal, it's probably the COOKIE_CUTTER_ROOT variable which is not correctly set.

4. Usage

cookie-cutter [-i *orbital_inclination*] [-T *orbital_period*] [-w *extraction_width*] *ascending_node_file input_directory output_file*

Option arguments

orbital_inclination

> The orbital inclination of the heliosynchronous satellite ; must be given in degrees (90° corresponds to a polar orbit). Default : 98.2° (A-Train value)

orbital_period

> The orbital period of the heliosynchronous satellite ; must be given in minutes. Default : 98.88 minutes (A-Train value)

`extraction_width`

> The width of extraction of SEVIRI data, around the heliosynchronous satellite track, in degrees of longitude at the equator. Default : 10°

Mandatory arguments

`ascending_node_file`

> This is a three-columns formatted text file, containing the dates, hours and longitudes of the sun-synchronous ascending orbit nodes. For example, here is a file containing these infos for the satellite PARASOL, the 1st of june 2005:

```
2005-06-01 01:07:50     184.1
2005-06-01 02:46:43     159.38
2005-06-01 04:25:36     134.66
2005-06-01 06:04:29     109.94
2005-06-01 07:43:23     85.22
2005-06-01 09:22:16     60.5
2005-06-01 11:01:09     35.78
2005-06-01 12:40:02     11.06
2005-06-01 14:18:55     346.34
2005-06-01 15:57:48     321.62
2005-06-01 17:36:41     296.9
2005-06-01 19:15:34     272.18
2005-06-01 20:54:27     247.46
2005-06-01 22:33:20     222.74
```

> Comments (beginning with #) and empty lines are currently forbidden in this text file, but might be supported in a future release.

`input_directory`

> The directory containing the XRIT input files, in the *SATMOS* archive format (see Appendix A). For example:

```
[user@hostname SATMOS-20050601]$ pwd
/path/to/data/seviri/SATMOS-20050601
[user@hostname SATMOS-20050601]$ ls
H-000-MSG1__-MSG1_____-IR016_____-CYCLE_____-200506010000
H-000-MSG1__-MSG1_____-IR016_____-CYCLE_____-200506010015
H-000-MSG1__-MSG1_____-IR016_____-CYCLE_____-200506010030
H-000-MSG1__-MSG1_____-IR016_____-CYCLE_____-200506010045
H-000-MSG1__-MSG1_____-IR016_____-CYCLE_____-200506010100
H-000-MSG1__-MSG1_____-IR016_____-CYCLE_____-200506010115
H-000-MSG1__-MSG1_____-IR016_____-CYCLE_____-200506010130
H-000-MSG1__-MSG1_____-IR016_____-CYCLE_____-200506010145
H-000-MSG1__-MSG1_____-IR016_____-CYCLE_____-200506010200
[...]
```

> Note that these are not the original files diffused by EUMETSAT, but files derived from them by grouping all the segments of the same channel. We'll call it the SATMOS archive format and provide a tool to convert the EUMETSAT archive format into the SATMOS archive format. See the appendix for more details.

output_file

> The name under which the output product will be saved. In fact two files will be created: an HDF file and a PPM file (an image file). Extensions .hdf and .ppm will be added automatically. A path may be specified but only the parent directory of the output file will be created if necessary. Otherwise an error will occur (for example if the parent of the parent doesn't exist). This behavior may change in the future, don't rely on it. You should always ensure that the parent directory already exists.

5. Some practice

It's time to do some practice. Here is an example of how to use cookie-cutter

Before going on, you should have built and configured the package.

1) Look for an EUMETSAT archive directory (with compressed files). It should contain files like these:

```
[user@hostname EUMETSAT-ARCHIVE]$ ls
[...]
/path/to/seviri_data/H-000-MSG1__-MSG1_____-_____-PRO_____-200506011200-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000001___-200506011200-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000002___-200506011200-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000003___-200506011200-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000004___-200506011200-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000005___-200506011200-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000006___-200506011200-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000007___-200506011200-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000008___-200506011200-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-_____-EPI_____-200506011200-C_
[...]
```

2) **catslot** one day of the directory content into another directory (see appendix A for a brief description of the command **catslot** and the SATMOS archive format):

```
[user@hostname cookie-cutter]$ catslot EUMETSAT-ARCHIVE SATMOS-20050601 20050601
[user@hostname cookie-cutter]$ cd SATMOS-20050601
[user@hostname SATMOS-20050601]$ ls
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506010000
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506010015
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506010030
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506010045
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506010100
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506010115
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506010130
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506010145
```

```
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506010200
[...]
```

Of course, the previous steps are in no way necessary if your MSG data were provided by the *SATMOS* archive center.

3) get (or produce) a text file containing node times and longitudes for this day. We'll call it `nodes_20050601.txt`:

```
2005-06-01 01:07:50     184.1
2005-06-01 02:46:43     159.38
2005-06-01 04:25:36     134.66
2005-06-01 06:04:29     109.94
2005-06-01 07:43:23     85.22
2005-06-01 09:22:16     60.5
2005-06-01 11:01:09     35.78
2005-06-01 12:40:02     11.06
2005-06-01 14:18:55     346.34
2005-06-01 15:57:48     321.62
2005-06-01 17:36:41     296.9
2005-06-01 19:15:34     272.18
2005-06-01 20:54:27     247.46
2005-06-01 22:33:20     222.74
```

4) Find the following orbital parameters of the chosen sun-synchronous orbit: inclination (we'll take 98.2° in this example) and orbital period (we'll take 98.88 minutes).

5) Choose a width of extraction (in degrees of longitude at the equator). We'll take 10°.

6) Run the command:

```
[user@hostname cookie-cutter]$ cookie-cutter -i 98.2 -T 98.88 -w 10 \
      nodes_20050601.txt SATMOS-20050601 output_file
```

The last argument is the name of the file which will be created by **cookie-cutter**. In fact, two files will be produced: `output_file.hdf` and `output_file.ppm`. The first one is a standard HDF file and the other one is an image in ppm format.

A. EUMETSAT and SATMOS archive formats, and how to convert them

EUMETSAT provides MSG1/SEVIRI data in XRIT format. Data are available each 15 minutes, in 8 segments for low-resolution channels and 24 segments for the high-resolution channel.

For example, here is all the files associated to the SEVIRI channel IR_016, for the slot 11:45 on 22th june 2005. PRO is for 'PROLOGUE', EPI is for 'EPILOGUE'. Prologue and Epilogue files are common to all the channels.

```
[user@hostname EUMETSAT-ARCHIVE]$
[...]
/path/to/seviri_data/H-000-MSG1__-MSG1_____-_____-PRO_____-200506221145-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000001___-200506221145-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000002___-200506221145-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000003___-200506221145-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000004___-200506221145-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000005___-200506221145-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000006___-200506221145-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000007___-200506221145-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-IR_016___-000008___-200506221145-C_
/path/to/seviri_data/H-000-MSG1__-MSG1_____-_____-EPI_____-200506221145-C_
[...]
```

Now imagine that there are 11 SEVIRI low-resolution channels and 1 high-resolution channel. This gives 114 files by 15 minutes, or about 11000 files by day.

cookie-cutter can't operate on separate segments. All the segments of the same slot (same hour) must be grouped together, with the Prologue and Epilogue files, before any processing by **cookie-cutter**. The operation is quite simple. You'll have to decompress all the files with **xRITDecompress** (a command provided with EUMETSAT), then to **cat** them in the order :

prologue
image segments
epilogue

The output must be given the following name (in the present case):

```
H-000-MSG1__-MSG1_____-IR016____-CYCLE____-200506221145
```

The number of files is reduced by nearly 8. This is the *SATMOS archive format*, from the *SATMOS* archive center at Lannion. Be aware that the underscore in the original channel name (IR_016) is still present, but has been moved *after* the name (IR016_).

The cookie-cutter suite provides a small command **catslot** which does all this nasty job for you. Suppose you've got EUMETSAT data files in a directory EUMETSAT-ARCHIVE. To have them processed by **cookie-cutter**, you'll need to convert them into the *SATMOS* archive format, in a directory, let's say, of name SATMOS-YYYYMMDD:

```
[user@hostname cookie-cutter]$ catslot EUMETSAT-ARCHIVE SATMOS-20050622 20050622
[user@hostname cookie-cutter]$ catslot EUMETSAT-ARCHIVE SATMOS-20050622 20050622 1145
[user@hostname cookie-cutter]$ catslot EUMETSAT-ARCHIVE SATMOS-20050622 20050622 1145 1900
```

In the first command, all files from 22th june 2005 in EUMETSAT-ARCHIVE will be concatened into SATMOS files in SATMOS-20050622. The two other commands show how to convert only some slots (you may specify one slot, or a range of slots)

cookie-cutter is ready to process all the files in SATMOS-20050622.

Warning

In order to work, **catslot** supposes that **xRITDecompress** is available in your PATH. You won't find **xRITDecompress** in the cookie-cutter suite, you'll have to get it from EUMETSAT.