ObsCore Builder

Nathan Barloy
nathan.barloy@astro.unistra.fr
Gilles Landais
gilles.landais@astro.unistra.fr

Abstract

The catalogs in the CDS VizieR service need to have an ObsCore table associated to them to be easily referenced. But the quantity of new catalogs we receive is huge compared to the number of staff we have on it.

That is why we are creating a tool to automatically generate those ObsCore tables from the FITS files of the catalogs

This tool is for now a Python library that will extract a maximum of information from the headers of the FITS files.

Once this is done, it is possible to curate the information with external services.

Furthermore, since a non-negligible number a FITS files don't follow the standard but make their own, we added a way to custom the way information will be extracted with a specification file we call a Mapping file.

For now this tool is aimed to help the ingestion process in the CDS VizieR service, but if more persons are interested, we could try to make an open version of this.

Reason

The number of catalogs we keep receiving for VizieR at the CDS is always growing.

A lot of them are not integrated in our ObsCore database (called assocdata), so it can be difficult to access them from searches. We have the ODAS project (Open Data for AStronomy) at CDS, and we want to have more catalogs indexed in the ObsCore database.

The problem is, the process is not easy, and our staff is already overworked, there is always more catalogs that are submitted to VizieR.

We want to try to automate this process at best, to unburden the load of work of our staff

The scope could be expanded if it works well : trying to have an open version of this, that other people will be able to use. But for now, we are really focused to have a tool to help our process.

Purpose

The ObsCore Builder library is a Python library that gives tools to create an ObsCore table from a collection of FITS files.

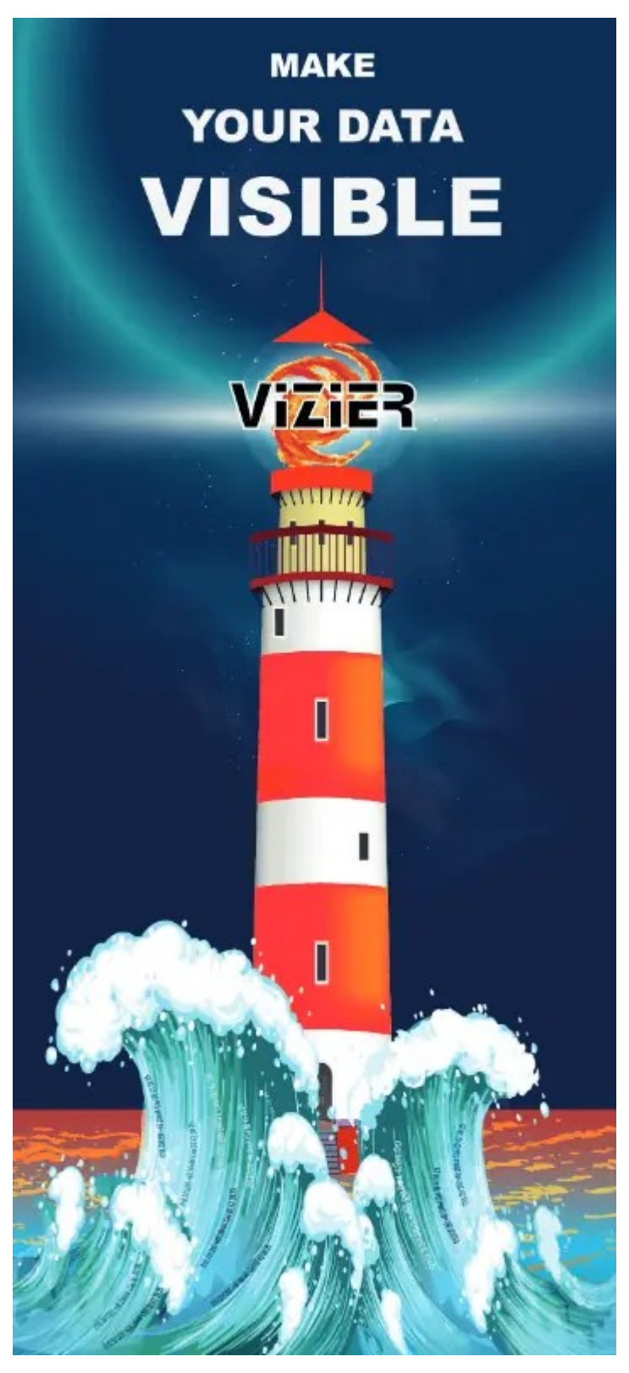
We already have a tool for this called Saada, but it is not really maintained and has some conception flaws making it always slower and bigger.

This library can not only extract information from the FITS files, but it can also be given complementary information from the user in the form of a Mapping file. Using this file, the library can also apply those complementary information to an already existing ObsCore table.

Furthermore, a Python library is very versatile and could be used in a lot of cases. For example, making a CLI would not be very difficult, and creating a back-end service would be very useful to create a new submission flow for new catalogs.

The ODAS project

The CDS-ODAS project (Open Data for AStronomy) is a 3-year project with the purpose of making the VizieR data more FAIR (Findable, Accessible, Interoperable, Reusable). We already follow the IVOA standards, but not enough of our data is easily accessible via some ObsCore information. The main focus will logically be to have a tool allowing for more data to be added in our ObsCore database, and to create a better workflow for data submission.



How does it work ?

Requirements

This Python library uses Python 3.11. It is heavily based on the AstroPy library.

Extraction

The ObsCore extraction from a FITS collection is the main feature of this library. Since the catalogs we receive can be very large, we have to be as efficient as possible when extracting the information. That is why only the headers of the files will be opened.

The extraction process is composed of 3 main steps :

1. Extract information from the FITS keywords. For some ObsCore fields, we just use a list of common FITS keywords that can contain the useful information. Those lists of common FITS keywords are in a configuration file, and can be modified/improved if needed.
2. Solve the WCS. Those are often the most important information of the ObsCore (s_ra and s_dec for an image, em_min and em_max for a spectrum).
3. Clean the results. We have to make sure that each value we got can be converted in the right format for ObsCore.

Note : There is often a dilemma about how much non-standard information we should accept and understand. The general solution we use is that we can support non-standard keywords and notations as long as it doesn't conflict with the FITS standards.

Mapping file

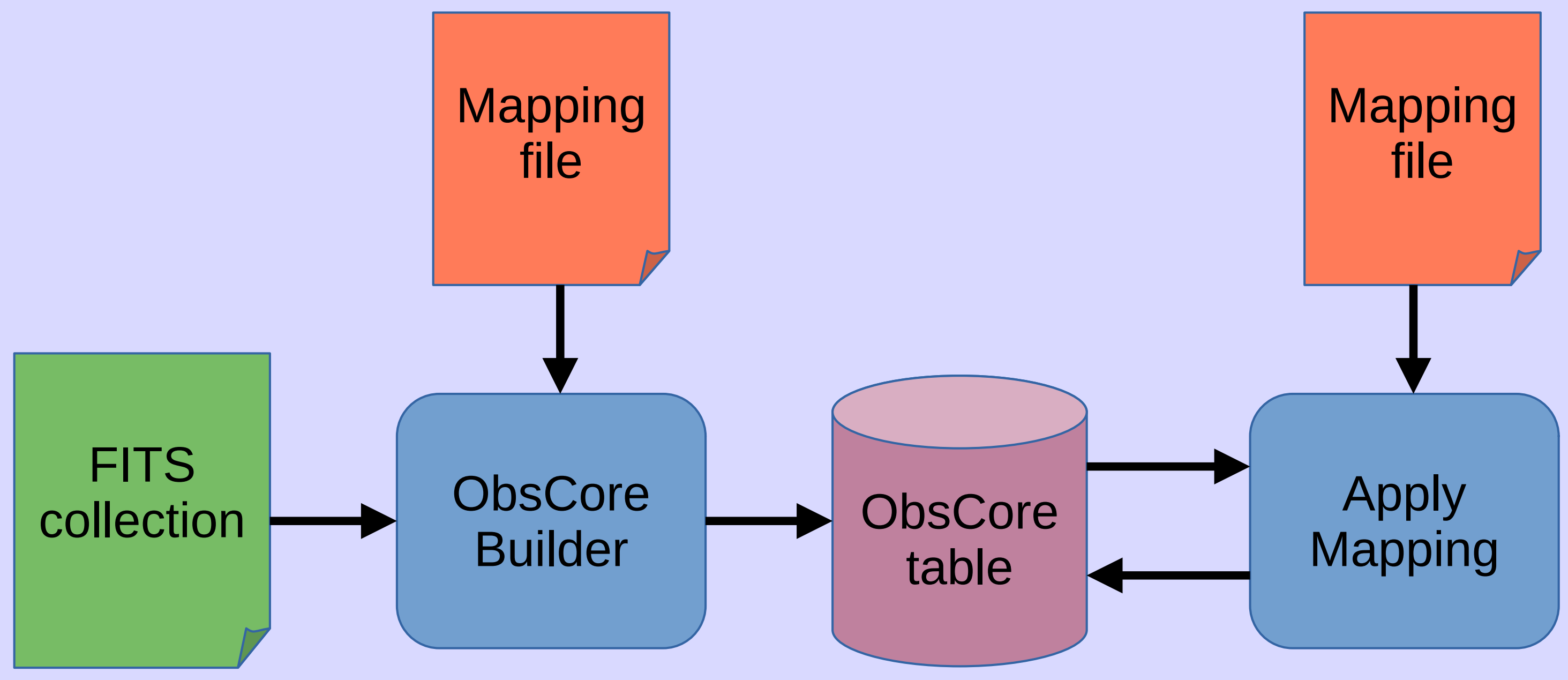
Sometimes, we know that the FITS files will follow their own rules, and have their own keywords. That is why it is useful to have a way for the user to indicate more specific information:

- Default values
- Missing FITS keywords
- Define your own functions
- …

Our Mapping file format is a way to achieve that.

Usage

There is 2 functions in this library. The first on extract an ObsCore table from a FITS collection and an optional Mapping file. The second one applies a Mapping file to an already existing ObsCore table to modify it.



Links and references

Astropy

CDS

VizieR Assocdata

ADASS – Görlitz, 9 Nov to 13 Nov 2025 – CDS – Observatoire astronomique de Strasbourg