SDHDF: Single Dish Hierarchical Data Format

A modern file format for spectral line and continuum data from single dish radio telescopes

Lawrence Toomey
DRAFT v1.9 (2020-02-03)

Table of Contents

1 Introduction

Single Dish Hierarchical Data Format (SDHDF) is a new file format for radio astronomy spectral line and continuum data from single dish telescopes, based on the Hierarchical Data Format (HDF - https://www.hdfgroup.org/).
With ever increasing instantaneous bandwidth and higher data volumes output from new receivers both existing and planned, there is a need for agreement on a new data format capable of meeting future requirements. The SDHDF file format was therefore specified and designed for the following reasons:

- The data model is based on HDF, a well-defined format with long term support
- It is capable of storing comprehensive data and meta-data products in a hierarchical structure
- It is modern, extensible and portable
- The format can be parsed by modern and conventional computer languages, as well as by open source HDF command line tools and graphical user interfaces

2 SDHDF Overview

SDHDF format can be thought of as a tree-like structure of ‘group’ and ‘dataset’ binary objects, containing the raw data, observation metadata, and time, frequency and polarisation information.

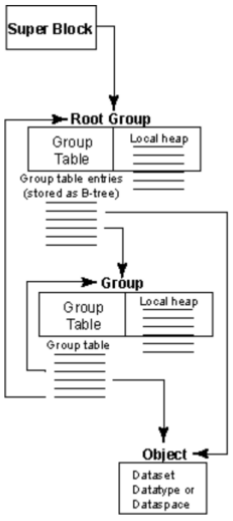


Figure 1: Relationships among the HDF5 root group, other groups, and objects

The format conforms to the HDF definition (HDF5 v3.0 https://portal.hdfgroup.org/display/HDF5/File+Format+Specification) at the time of writing. Figure 1 shows the group-object relationship.

A single output SDHDF file is a combination of many input data and metadata products from a given observation. Input data streams and associated metadata are captured for both astronomy and calibration data products, in addition to general observation metadata.

3 The SDHDF Definition

A collection of Python classes in a configuration module defining the data and metadata structure, descriptions, units and values, are implemented to form a template SDHDF file for a specific observation. The template structure is then populated by the raw data and metadata to form the final product.

The configuration module may be adapted to suit a particular set of backend parameters for a particular telescope.

Full details and structure of the SDHDF definition can be found in Appendix A.

4 Viewing File Contents

The HDF group provide basic command-line tools that can be installed on Unix/Linux systems. For example:

- h5ls
- h5dump

There are also GUIs available, for example:

- h5pyViewer (https://pypi.org/project/h5pyViewer/)
- HDFView (https://www.hdfgroup.org/downloads/hdfview/)
- Panoply (https://www.giss.nasa.gov/tools/panoply/)

Pythonic interfaces and modules are also available, for example:

- h5py (https://www.h5py.org/)

4.1 Command-line HDF tools

In this section we will describe how to access the contents of an SDHDF file with command-line HDF tools such as h5ls and h5dump.

4.1.1 Example: using h5ls and h5dump

In this example, the structure of the top level of an SDHDF file can be viewed with:

```
% h5ls uwl_191208_055418.hdf
```

Where the output may be similar to:

```
beam_0                   Group
config                   Group
metadata                 Group
```

A detailed look at the structure can be viewed with:

```
% h5dump -n uwl_191208_055418.hdf
```

Where the output may be similar to:

```
HDF5 "uwl_191208_055418.hdf" {
FILE_CONTENTS {
 group      /
 group      /beam_0
 group      /beam_0/band_SB7
 group      /beam_0/band_SB7/astronomy_data
 dataset    /beam_0/band_SB7/astronomy_data/data
 dataset    /beam_0/band_SB7/astronomy_data/frequency
 group      /beam_0/band_SB7/calibrator_data
 dataset    /beam_0/band_SB7/calibrator_data/cal_data_off
 dataset    /beam_0/band_SB7/calibrator_data/cal_data_on
 dataset    /beam_0/band_SB7/calibrator_data/cal_frequency
 group      /beam_0/band_SB7/metadata
 dataset    /beam_0/band_SB7/metadata/cal_obs_params
 dataset    /beam_0/band_SB7/metadata/obs_params
 group      /beam_0/band_SB8
 group      /beam_0/band_SB8/astronomy_data
 dataset    /beam_0/band_SB8/astronomy_data/data
 dataset    /beam_0/band_SB8/astronomy_data/frequency
 group      /beam_0/band_SB8/calibrator_data
 dataset    /beam_0/band_SB8/calibrator_data/cal_data_off
 dataset    /beam_0/band_SB8/calibrator_data/cal_data_on
 dataset    /beam_0/band_SB8/calibrator_data/cal_frequency
 group      /beam_0/band_SB8/metadata
 dataset    /beam_0/band_SB8/metadata/cal_obs_params
 dataset    /beam_0/band_SB8/metadata/obs_params
 group      /beam_0/metadata
 dataset    /beam_0/metadata/band_params
 dataset    /beam_0/metadata/cal_band_params
 group      /config
 dataset    /config/backend_config
 dataset    /config/cal_backend_config
 group      /metadata
 dataset    /metadata/beam_params
 dataset    /metadata/history
 dataset    /metadata/primary_header
 dataset    /metadata/software_versions
 }
}
```

An observation can have one or many ‘beam’ groups (one for a single pixel feed). A beam group may have many ‘band’ groups depending on the receiver frequency range configuration. A ‘band’ group contains astronomy data, calibrator data and associated metadata.

The following command shows the content of the astronomy data group:

```
% h5ls uwl_191208_055418.hdf/beam_0/band_SB7/astronomy_data
```

Where the output may be similar to:

```
data                     Dataset {182, 1, 4, 262144, 1}
frequency                Dataset {262144}
```

The dimensions of the data arrays are shown in the curly brackets.

4.2 Graphical user interfaces

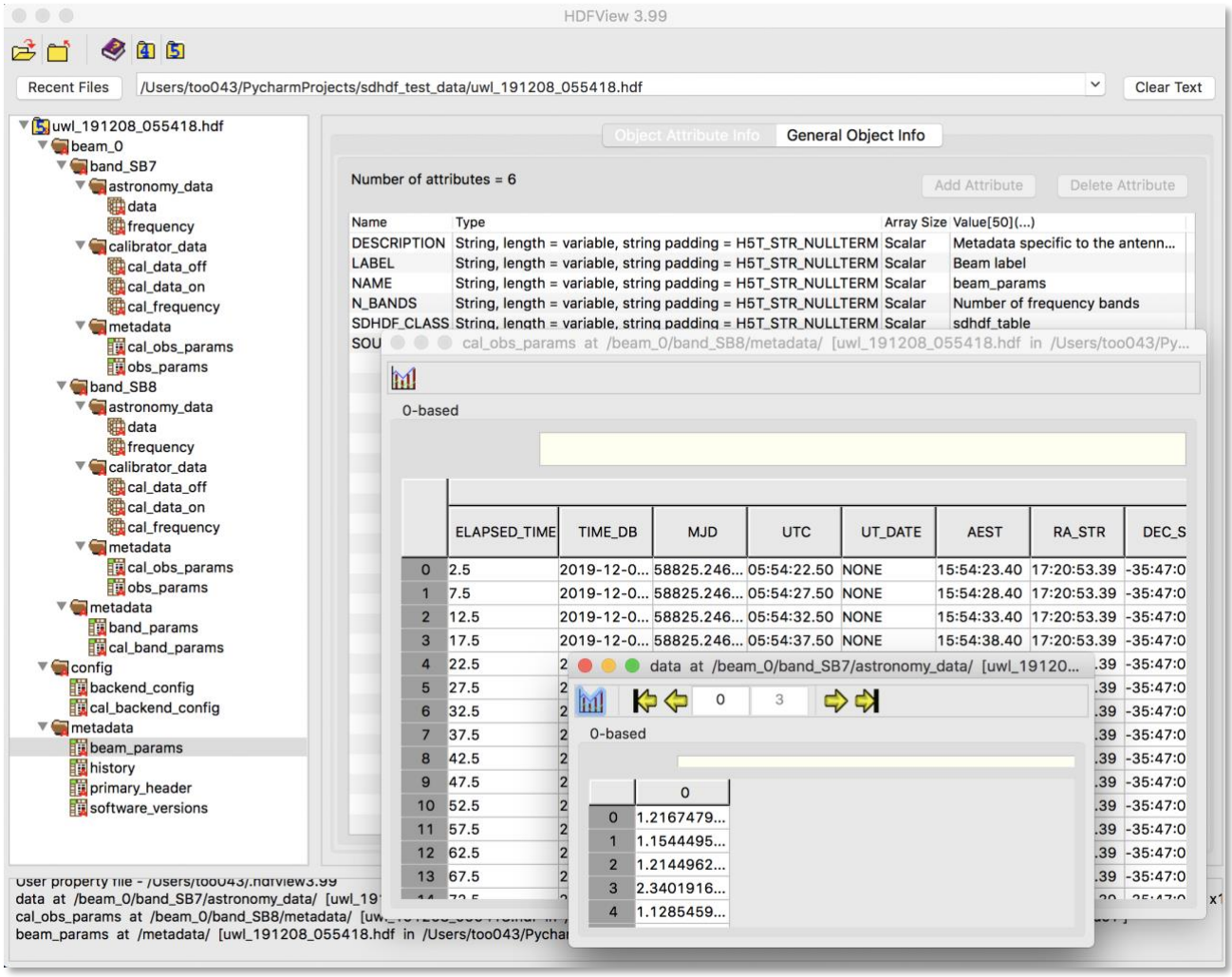
In this section we show how to view SDHDF file contents using open source graphical user interfaces.

4.2.1 Example 1: HDFView

In this example, we demonstrate how HDFView can be used to display available data and meta-data in an SDHDF file. Load the file with:

```
% hdfview uwl_191208_055418.hdf
```

Click on an HDF object in the side pane to display the HDF attributes and meta-data, and double-click to display the data.



4.3 Python interfaces

H5py is a powerful pythonic interface to view and manipulate HDF files, and is the module used to write the SDHDF data products.
Additional python modules are available for viewing data and metadata, and can be found in the following git repository:

https://bitbucket.csiro.au/projects/CPDA/repos/sdhdf_tools/browse

4.3.1 Viewing metadata

A basic implementation of this interface can be found in the read_sdhdf_header.py module that describes the file contents, for example:

```
% python read_sdhdf_header.py --filename uwl_191208_055418.hdf
```

Where the output may be similar to:

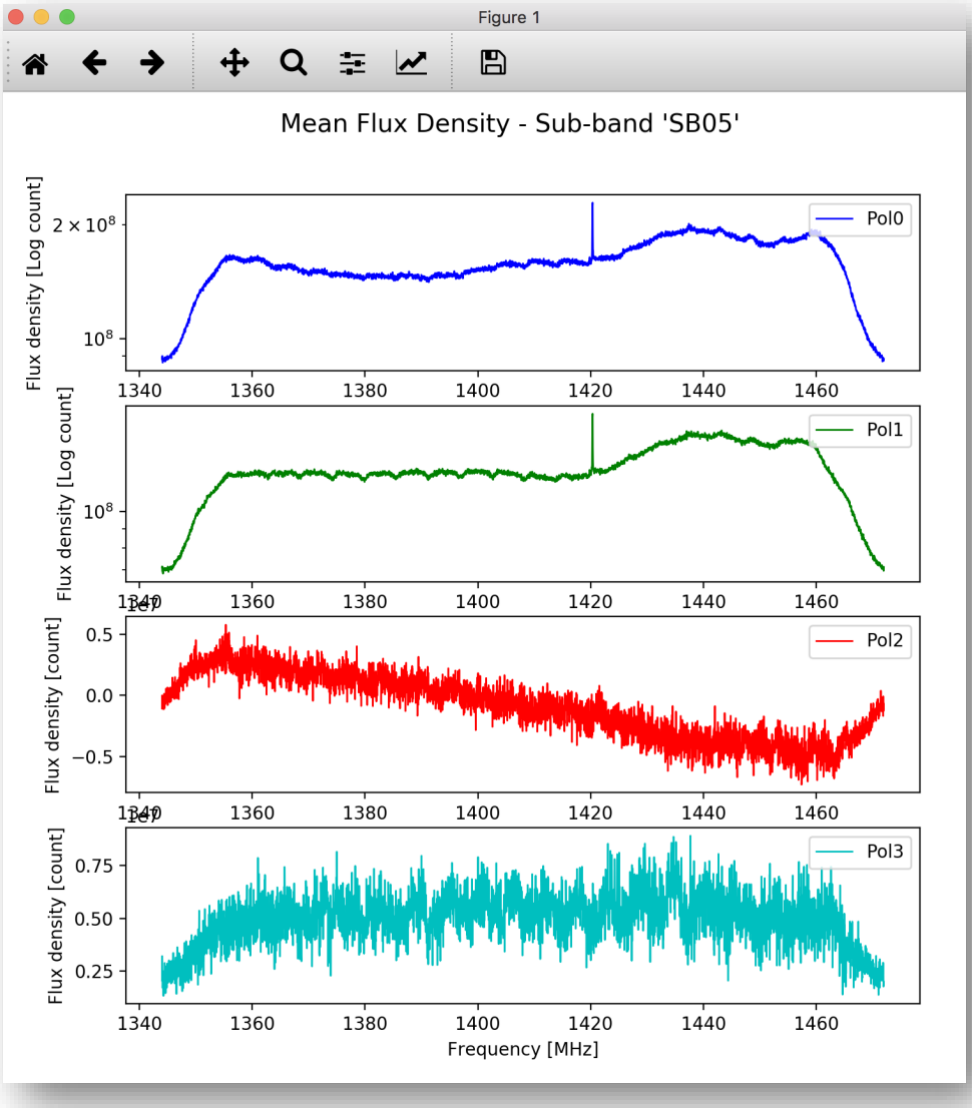
```
Displaying primary header for file:
uwl_190531_092815.hdf

               -- Key --           -- Value --
0                   DATE  2020-02-03-10:25:11
1              HDR_DEFN                 SDHDF
2      HDR_DEFN_VERSION                   1.9
3           FILE_FORMAT                   HDF
4   FILE_FORMAT_VERSION                   5.0
5         SCHED_BLOCK_ID                  6183
6              CAL_MODE                    ON
7            INSTRUMENT                Medusa
8              OBSERVER                gre469
9                   PID                  P737
10            RECEIVER                   UWL
11           TELESCOPE                 Parkes
12           UTC_START  2019-12-08-05:54:18
13             N_BEAMS                     1
```

4.3.2 Plotting spectra

The plot_sdhdf.py module can display the spectrum for individual sub-bands. Example usage is:

```
% python plot_sdhdf.py --filename uwl_190531_092815.hdf
```

Where the output may be similar to:



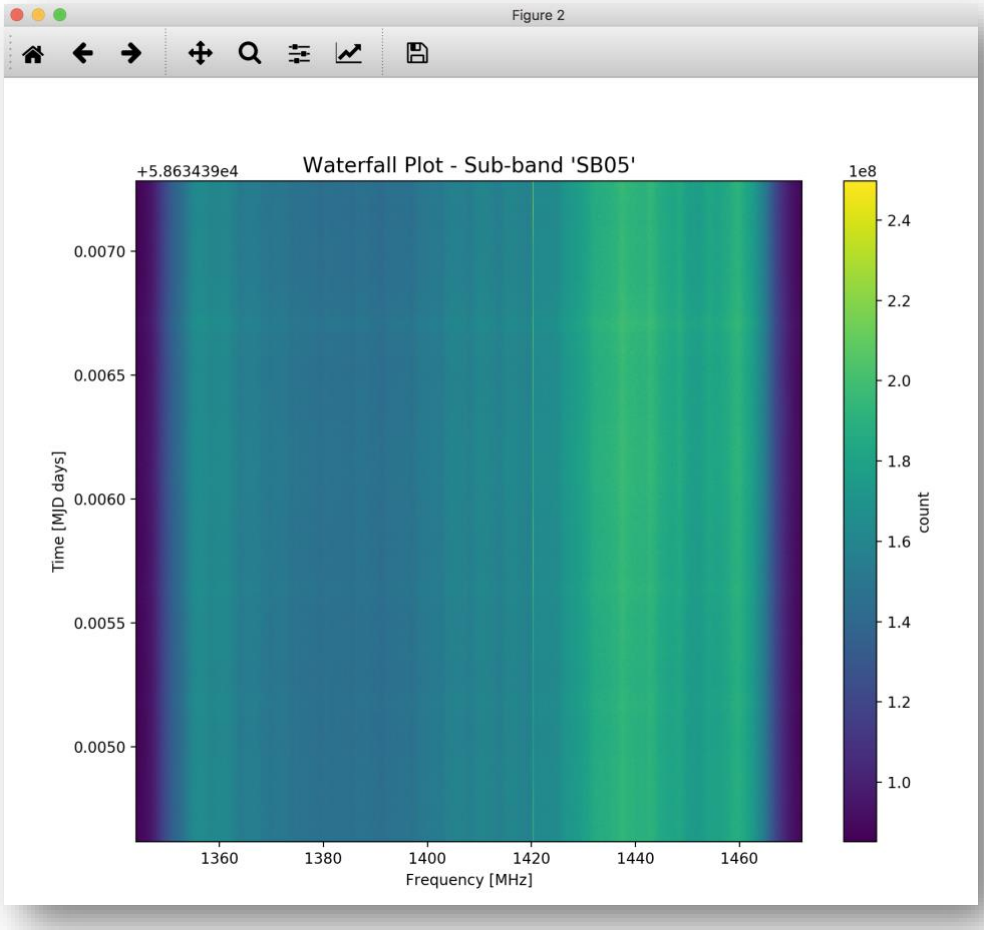


Figure 2
Waterfall Plot - Sub-band 'SB05'
+5.863439e4
Time [MJD days]
0.0070
0.0065
0.0060
0.0055
0.0050
1360
1380
1400
1420
1440
1460
Frequency [MHz]
1e8
2.4
2.2
2.0
1.8
1.6
1.4
1.2
1.0
count

Appendix A: The SDHDF definition (v1.9)

A1. SDHDF overview

A1.1 SDHDF Definition Overview

SDHDF Definition Version: 1.9
Author: Lawrence Toomey
Copyright: CSIRO 2020

A1.2 SDHDF File Overview

HDF_Object_Name	HDF_Object_Type	Description
uwl_191208_055418.hdf	File	SDHDF format file

A1.3 SDHDF Structure Overview

HDF_Object_Name	HDF_Object_Type	Description
/beam_0	Group	SDHDF group containing data products specific to the antenna beam
/beam_0/band_SB7	Group	SDHDF group containing data products specific to the frequency band
/beam_0/band_SB7/astronomy_data	Group	SDHDF group containing observation data
/beam_0/band_SB7/calibrator_data	Group	SDHDF group containing observation data (calibrator)
/beam_0/band_SB7/metadata	Group	SDHDF group containing observation metadata
/beam_0/band_SB8	Group	SDHDF group containing data products specific to the frequency band
/beam_0/band_SB8/astronomy_data	Group	SDHDF group containing observation data
/beam_0/band_SB8/calibrator_data	Group	SDHDF group containing observation data (calibrator)
/beam_0/band_SB8/metadata	Group	SDHDF group containing observation metadata
/beam_0/metadata	Group	SDHDF group containing observation metadata
/beam_0/metadata/band_params	Dataset	Metadata specific to the frequency bands of the antenna beam
/beam_0/metadata/cal_band_params	Dataset	Metadata specific to the frequency bands of the antenna beam

HDF_Object_Name	HDF_Object_Type	Description
/config	Group	SDHDF group containing configuration parameters as defined at the time of the observation
/config/backend_config	Dataset	Astronomy backend configuration
/config/cal_backend_config	Dataset	Astronomy backend configuration (calibration)

HDF_Object_Name	HDF_Object_Type	Description
/metadata	Group	SDHDF group containing observation metadata
/metadata/beam_params	Dataset	Metadata specific to the antenna beam
/metadata/history	Dataset	Metadata specific to the processing history of the file
/metadata/primary_header	Dataset	General observation metadata
/metadata/software_versions	Dataset	Metadata specific to software packages used for creating or processing the file