Australia's National
Science Agency

SDHDF: Single Dish Hierarchical Data Format

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

Lawrence Toomey

V2.0 (June 22nd 2021)

Table of Contents

1 Introduction

Single Dish Hierarchical Data Format[1] (SDHDF) is a new file format for radio astronomy spectral line and continuum data from single dish telescopes, based on the Hierarchical Data Format[2] (HDF).

With ever increasing instantaneous bandwidth and higher data volumes output from new receivers both existing, such as the Ultra-Wide-Bandwidth Low Frequency (UWL, Hobbs et al. 2020) receiver at Parkes and planned (for example the cryo-PAF receiver at Parkes) , there is a need for agreement on a new data format capable of meeting future requirements.

The use of HDF as a file format for Parkes spectral line data was first introduced specifically for the HI-Pulsar 'HIPSR' digital backend (Price et al. 2016) of the 21cm Multibeam receiver, as a replacement to the more traditional format amongst the astronomical fraternity, FITS (Pence et al. 2010).

Consequently we explored the use of HDF for all spectral line data products from the UWL, converging on the SDHDF model as the formal convention due to the following:

- The data model is based on HDF version 5 (HDF5), a well-defined format
- It is capable of storing comprehensive data and metadata products in a nested hierarchical structure suitable for telescopes with multiple beams
- It is extensible and portable, with long-term development support
- The format can be parsed by modern and conventional computer languages, as well as by open source tools and graphical user interfaces (GUIs)
- Data products are suitable for ingest into CSIRO's Australia Telescope Online Archive[3] (ATOA)

In this document, we introduce the SDHDF format and definition in Section 2, describe a suite of tools for interrogating the format in Section 3, and present SDHDF data reduction tools in Section 4.

The full detailed definition is laid out in the formal definition document (location TBD).

[1] The formal 'SDHDF' definition is in Toomey et. al 2020 (in prep.)

[2] https://www.hdfgroup.org/

[3] https://atoa.atnf.csiro.au/

2 SDHDF Overview

The 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, and conforms to the HDF definition (HDF5 v3.0[4]) at the time of writing.

An SDHDF file can be a combination of many input data streams, associated metadata for both astronomy and calibration data products, and observation metadata, but may also be a product of post-processing from any number of other SDHDF data and metadata products structured according to the definition - such is the flexible nature of the format.

The data and metadata structure, descriptions, units and values, are implemented in a template SDHDF file for a specific observation. The template structure is then populated by the raw data and metadata to form the final data product.

The metadata are stored in HDF 'attributes', small objects directly describing the dataset or group object they are attached to.

The configuration may be adapted to suit a particular set of parameters for a specific part of the hardware and software chains. This flexibility allows uptake by multiple radio astronomy institutions, whilst the implementation of the template ensures that all data products adhere strictly to the file format definition, thereby enhancing data provenance.

Full details and structure of the SDHDF definition can be found in Appendix A, with version history detailed in Appendix B.

3 Viewing File Contents

The HDF group provides basic command-line tools that can be installed on Unix/Linux systems, such as *h5ls* and *h5dump*.

There are also open source GUIs available, for example *h5pyViewer*[5]*, HDFView*[6], *Panoply*[7] *and Silx*[8].

Pythonic interfaces and modules are also available, such as *h5py*[9].

---

[4] https://portal.hdfgroup.org/display/HDF5/File+Format+Specification

[5] https://pypi.org/project/h5pyViewer/

[6] https://www.hdfgroup.org/downloads/hdfview/

[7] https://www.giss.nasa.gov/tools/panoply/

[8] http://www.silx.org/doc/silx/0.7.0/index.html

[9] https://www.h5py.org/

3.1 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*.

3.1.1 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}
```

In the above example, the dimensions of the data arrays are shown in the curly brackets.

3.2 Graphical user interfaces

In this section we show how to view SDHDF file contents using open source GUIs.

3.2.1 Example: 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 (see Figure 1).

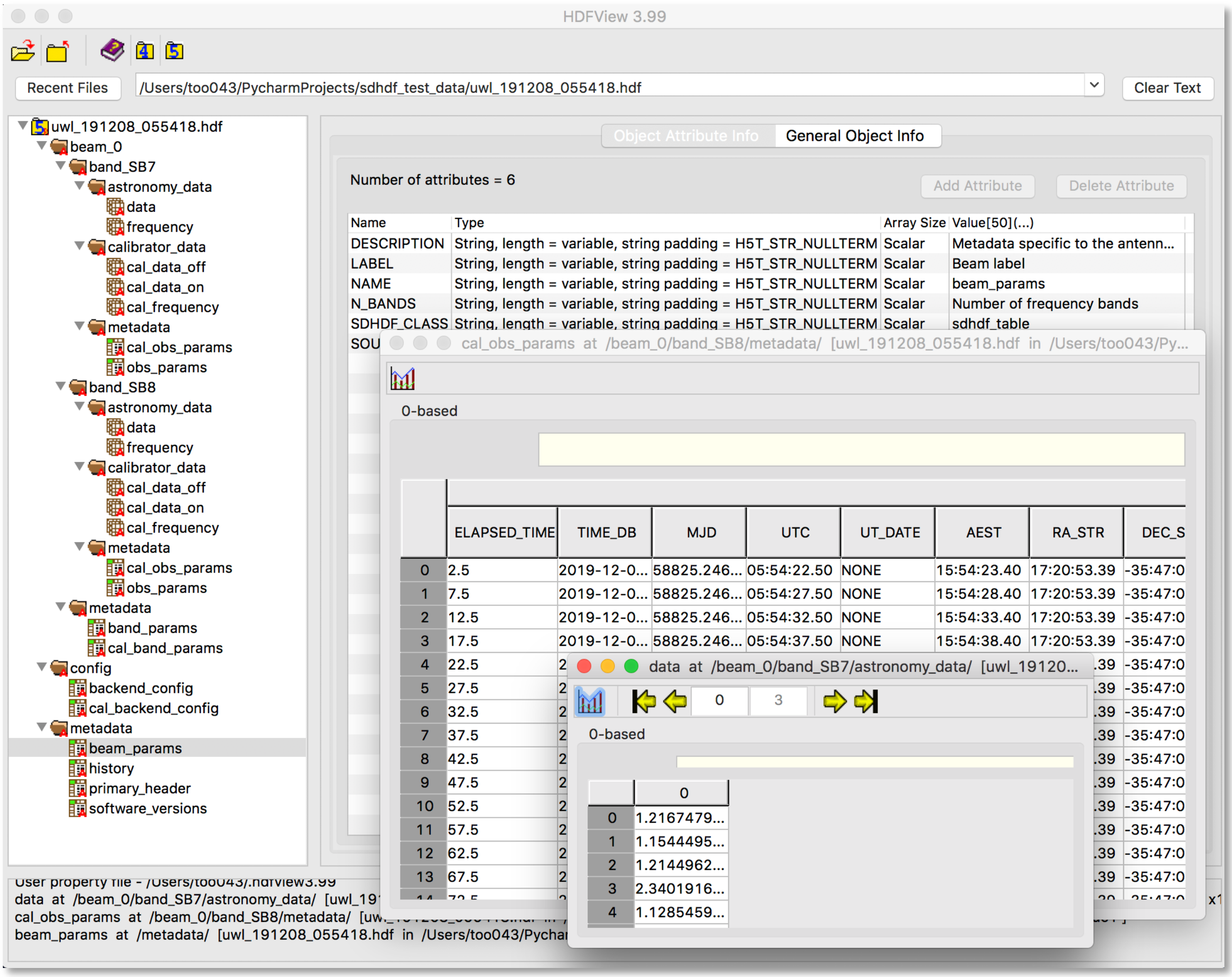

Figure 1: HDFView display panels

3.3 Python interfaces

H5py is a powerful python 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 SDHDF tools repository[10].

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

3.3.1 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
```

3.3.2 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 those in Figure 2.

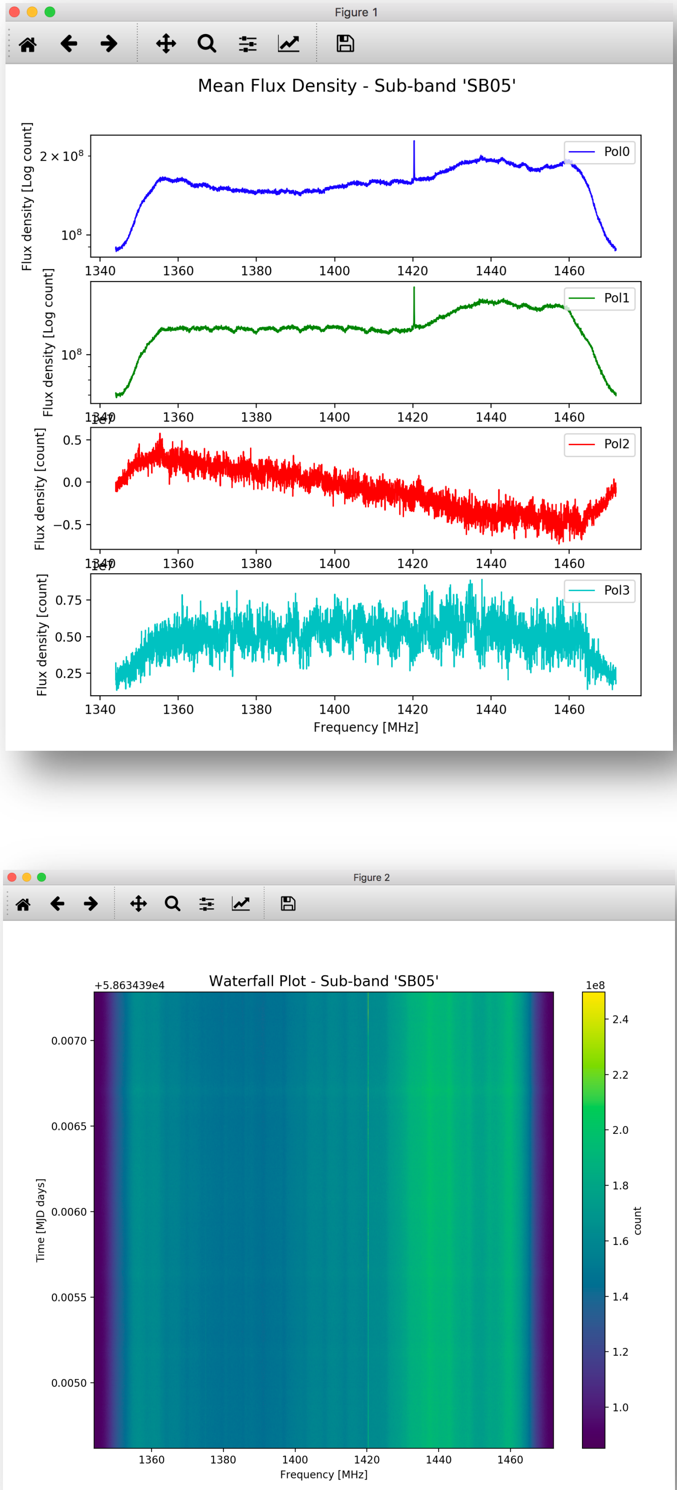

Figure 2: A plot of the spectrum (top) and waterfall data from an example SDHDF file

4 Processing an SDHDF File

A suite of routines '*sdhdfProc*', written in C, for the interrogation, processing and calibration of SDHDF data are also available from the SDHDF tools repository[11]. The suite comes with substantial documentation on the usage of *sdhdfProc* and examples of use cases.

In Figure 3, an example from the documentation, the *sdhdf_plotSpectrum* routine is used to plot a region of the spectrum around an OH maser, and then convert the frequency axis into velocity at the Local Standard of Rest using *sdhdf_modify*.

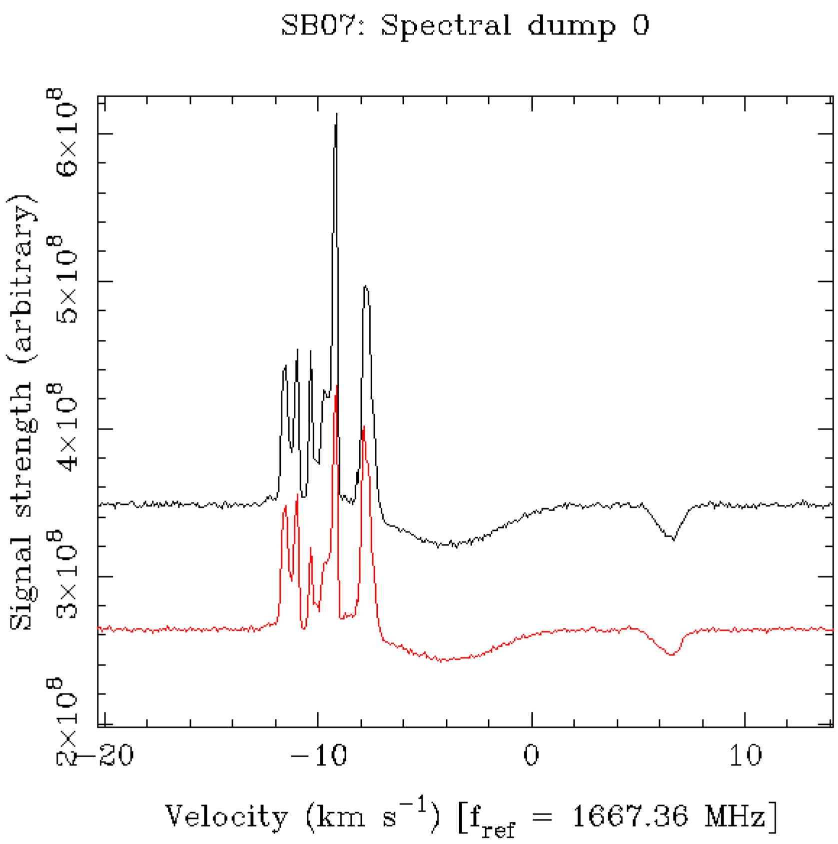

Figure 3: An OH maser at LSR formed by the *sdhdfProc* package

For further information on *sdhdfProc*, please contact george.hobbs@csiro.au.

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

5 Conclusion

In this document we have presented a new file format, SDHDF, for spectral line radio astronomy data, a format flexible enough to handle current and high data volumes expected from an era of large telescopes with single pixel or multi-beam receivers.

The Parkes radio telescope is the first to use the format natively for spectral line and continuum observations, since late 2018, and we encourage other institutions to adopt this as their primary format because of its flexible nested structure and comprehensive metadata capture.

The SDHDF definition will be formerly published in a refereed journal shortly, and we have provided details in this document on how to access SDHDF data reduction packages.

References

Hobbs, G., et al., 2020. *An ultra-wide bandwidth (704 to 4032 MHz) receiver for the Parkes radio telescope*. Publications of the Astronomical Society of Australia, Volume 37, article id. e012

Pence, W.D., et al., 2010. *Definition of the Flexible Image Transport System (FITS), version 3.0*. A&A 524, 42.

Price, D., et al., 2016. *HIPSR: A Digital Signal Processor for the Parkes 21-cm Multibeam Receiver*

https://arxiv.org/abs/1702.00443

As Australia's national science agency and innovation catalyst, CSIRO is solving the greatest challenges through innovative science and technology.

CSIRO. Unlocking a better future for everyone.

Contact us

1300 363 400
+61 3 9545 2176
csiroenquiries@csiro.au
www.csiro.au

For further information

CSIRO Space & Astronomy
Lawrence Toomey
+61 0 0000 0000
lawrence.toomey@csiro.au
csiro.au/cass