

# Using Compression in HDF5

HDF5 allows compression filters to be applied to a dataset to minimize the amount of space it consumes. There are both pre-defined compression filters (ZLIB and SZIP) and user-defined filters that can be applied.

To apply a compression filter to a dataset it must be created with a chunked dataset storage layout. This can be done by modifying the dataset creation property list. For a brief introduction to using compression in HDF5 see the following [HDF5 Tutorial](#) topics that discuss property lists, the storage layout and creating a compressed dataset:

## Property Lists

[Dataset Storage Layout: Contiguous, Chunked, Compact](#)  
[Creating a Compressed Dataset](#)

For tuning I/O performance with chunked and compressed datasets see: [TechNote-HDF5-ImprovingCompressionPerformance \(docx\)](#)  
For diagnosing situations where a dataset may not be compressed as well as expected, see: [HDF5 Compression Troubleshooting](#)

Below are a variety of links to more information regarding compression:

- [What kind of compression methods does HDF5 support ?](#)
- [HDF5 User's Guide - Old, Chapter 5, HDF5 Datasets, section 5.6 Using HDF5 Filters](#)
- [Third-Party Compression Filters](#)
- [Open Source SZIP Compression \(AEC\)](#) (from the [German Climate Computing Center](#))
- [SZIP Compression in HDF Products](#) (replaced by AEC as of HDF5-1.10.7)

## Third-Party Compression Filters

HDF5 provides two pre-defined filters for compression (ZLIB and SZIP). All other compression methods must be registered with HDF5 before they can be used. Before formally using a third-party filter in an application, it should be registered with The HDF Group, as well. For information on how to register a user-specified compression method with The HDF Group, as well as information on compression methods that are already registered, see: [Registered Third-Party Filters \(Compression\)](#)

## Example Code for Implementing BZIP2 Compression in HDF5

The [bz\\_example.tar.gz](#) file contains an example of implementing the BZIP2 filter to enable BZIP2 compression in HDF5. (This example is based on PyTables code that uses BZIP2 compression.). Download and uncompress this file as follows:

```
gzip -cd bz_example.tar.gz | tar xvf -
```

To compile the example, you will need to install the HDF5 library and use the h5cc compile script found in the bin/ directory of the HDF5 installation. For information on obtaining HDF5, see: [Obtain the Latest HDF5 Software](#)

For information on h5cc, see: [Compiling Your HDF5 Application](#)

Please note that tools like h5dump that display information in an HDF5 file will not be able to display data that is compressed with BZIP2 compression, since BZIP2 is not implemented in HDF5.

However, as of HDF5-1.8.11, a new HDF5 feature will enable the h5dump tool to determine that the data is compressed with an external compression filter such as BZIP2, and will automatically load the appropriate library and display the uncompressed data.

The [bz\\_example](#) example code can be used for modifying the HDF5 source to "include" BZIP2 as one of the "internal" filters. For information on how to do this, see how ZLIB (the deflate filter) is implemented in the HDF5 source code. Specifically look at these files:

[H5Z.c](#), [H5Zdeflate.c](#) and [H5Pocpl.c](#)