
h5cube Documentation

Release 0.3

Brian Skinn

26 November 2016

1	Usage	3
2	API	5
3	Gaussian CUBE File Format	9
4	h5cube File Specification	11
5	Dependencies	13
6	References	15
7	TODO List	17
8	Indices and tables	19
	Python Module Index	21

[Purposes for this... lossless or intelligently lossy compression. But, since HDF5, also serves as a proposal of sorts for an HDF5 file specification for compressed CUBE data.]

Only scalar volumetric data currently supported (one value per spatial point)

[Link GitHub](#)

[Link PyPI](#)

Note: Where used herein, the key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” are to be interpreted as described in [RFC 2119](#).

Contents:

Usage

Pending

Focus on command line usage, probably

Probably worth mentioning aspects of API usage too, though.

API intro. Two functions, three constants classes. API functions work directly on files on disk; no options for in-memory-only (de)compression as yet **Functions**

`h5cube.cube_to_h5(cubepath, *, ...)`
[Docstring]

`h5cube.h5_to_cube(h5path, *, delsrc=False, prec=5)`
[Docstring]

Less error/syntax checking here since presumably the data was parsed for validity when the .h5cube file was created.

Constants Classes

class `h5cube.H5`

Names of .h5cube datasets and informational constants

Array shapes and types are those of the data stored in .h5cube files.

COMMENT1 = 'COMMENT1'
First comment line

COMMENT2 = 'COMMENT2'
Second comment line

DSET_IDS = 'DSET_IDS'
If *NATOMS* is negative, a length-*NUM_DSETS* vector of *int*. Else, an empty *float np.array*

GEOM = 'GEOM'
System geometry. 3N x 5 *float*. Each line contains the atomic number [0], atomic charge [1] (typically – always? – equal to the atomic number), and position [2 :] of each atom

LOGDATA = 'LOGDATA'
Base-10 logarithms of the absolute values of the volumetric data points (0.0 for zero values). Dimension is identical to that of *SIGNS*.

NATOMS = 'NATOMS'
Number of atoms. Scalar *int*. Negative if dataset ID values are present.

NUM_DSETS = 'NUM_DSETS'
If *NATOMS* is negative, contains a scalar *int* with the length of *DSET_IDS*.

ORIGIN = 'ORIGIN'
Geometry origin vector. Length-3 *float*.

SIGNS = 'SIGNS'
Algebraic signs of the volumetric data points (0.0 for zero values). For non-orbital CUBE files, *XAXIS[0]*

`x YAXIS[0] x ZAXIS[0]` array of `float`. For orbital CUBE_s, `XAXIS[0] x YAXIS[0] x ZAXIS[0]`
`x NUM_DSETS` array of `float`.

VAL_NOT_ORBFILE = 0

Value to be stored in `NUM_DSETS` when compressing non-orbital CUBE files.

XAXIS = 'XAXIS'

x-axis voxel dimension [0] and vector [1:]. Length-4 `float`.

YAXIS = 'YAXIS'

y-axis voxel dimension [0] and vector [1:]. Length-4 `float`.

ZAXIS = 'ZAXIS'

z-axis voxel dimension [0] and vector [1:]. Length-4 `float`.

class h5cube.DEF

Default values for (de)compression option parameters.

No default is given for the *signed* parameter to `cube_to_h5()` since it is irrelevant when `THRESH` is `False`.

Similarly, since *minmax* and *isofactor* are irrelevant when `THRESH` is `False`, their default `None` values are specified directly in the `cube_to_h5()` signature.

CLIPZERO = False

Default is to clip small values to the thresholding limit with the smallest magnitude, as opposed to coercing to zero.

Todo

Add link to User's Guide explanation of value vs zero clipping

COMP = 9

Default gzip compression level.

DEL = False

Default is not to delete source file in both compression and decompression.

PREC = 5

Default output precision in decompressed CUBE file.

THRESH = False

Default is not to threshold the CUBE data when compressing.

TRUNC = 5

Default volumetric data truncation length during compression.

Todo

Add link to User's Guide explanation of truncation when done.

class h5cube.EXIT

Exit codes for command line invocations.

CMDLINE = 2

Exit with invalid command line parameters

FILEREAD = 4

Exit with error due to failed file read

FILEWRITE = 8

Exit with error due to failed file write

GENERIC = 1

Catch-all exit with error

OK = 0

Exit success

Gaussian CUBE File Format

Layout of CUBE format

h5cube File Specification

h5py dataset names, types, and array sizes (where relevant) for the stored data in h5cube files

Dependencies

h5py and numpy required for core functionality

Once implemented, tqdm optional for status bars

References

Refs list...

TODO List

Todo

Add link to User's Guide explanation of value vs zero clipping

(The original entry is located in `/home/docs/checkouts/readthedocs.org/user_builds/h5cube/checkouts/v.doc/h5cube/h5cube.py:docstring of h5cube.DEF.CLIPZERO`, line 4.)

Todo

Add link to User's Guide explanation of truncation when done.

(The original entry is located in `/home/docs/checkouts/readthedocs.org/user_builds/h5cube/checkouts/v.doc/h5cube/h5cube.py:docstring of h5cube.DEF.TRUNC`, line 3.)

Indices and tables

- `genindex`
- `modindex`
- `search`

h

h5cube, [5](#)

C

CLIPZERO (h5cube.DEF attribute), 6
CMDLINE (h5cube.EXIT attribute), 6
COMMENT1 (h5cube.H5 attribute), 5
COMMENT2 (h5cube.H5 attribute), 5
COMP (h5cube.DEF attribute), 6
cube_to_h5() (in module h5cube), 5

D

DEF (class in h5cube), 6
DEL (h5cube.DEF attribute), 6
DSET_IDS (h5cube.H5 attribute), 5

E

EXIT (class in h5cube), 6

F

FILEREAD (h5cube.EXIT attribute), 6
FILEWRITE (h5cube.EXIT attribute), 6

G

GENERIC (h5cube.EXIT attribute), 7
GEOM (h5cube.H5 attribute), 5

H

H5 (class in h5cube), 5
h5_to_cube() (in module h5cube), 5
h5cube (module), 5

L

LOGDATA (h5cube.H5 attribute), 5

N

NATOMS (h5cube.H5 attribute), 5
NUM_DSETS (h5cube.H5 attribute), 5

O

OK (h5cube.EXIT attribute), 7
ORIGIN (h5cube.H5 attribute), 5

P

PREC (h5cube.DEF attribute), 6

R

RFC
RFC 2119, 1

S

SIGNS (h5cube.H5 attribute), 5

T

THRESH (h5cube.DEF attribute), 6
TRUNC (h5cube.DEF attribute), 6

V

VAL_NOT_ORBFILE (h5cube.H5 attribute), 6

X

XAXIS (h5cube.H5 attribute), 6

Y

YAXIS (h5cube.H5 attribute), 6

Z

ZAXIS (h5cube.H5 attribute), 6