ESDIS
EARTH SCIENCE DATA AND INFORMATION SYSTEM

Cloud Optimized HDF5 Files: Current Status

Aleksandar Jelenak
NASA EED-3 / HDF Group

2024 ESIP Summer Meeting

NASA

Glossary

HDF5: Hierarchical Data Format Version 5

netCDF-4: Network Common Data Form version 4

COH5: Cloud optimized HDF5

S3: Simple Storage Service

EOSDIS: NASA Earth Observing System Data and Information System

MB: megabyte (10^6 bytes)

kB: kilobyte (10^3 bytes)

MiB: mebibyte (2^{20} bytes)

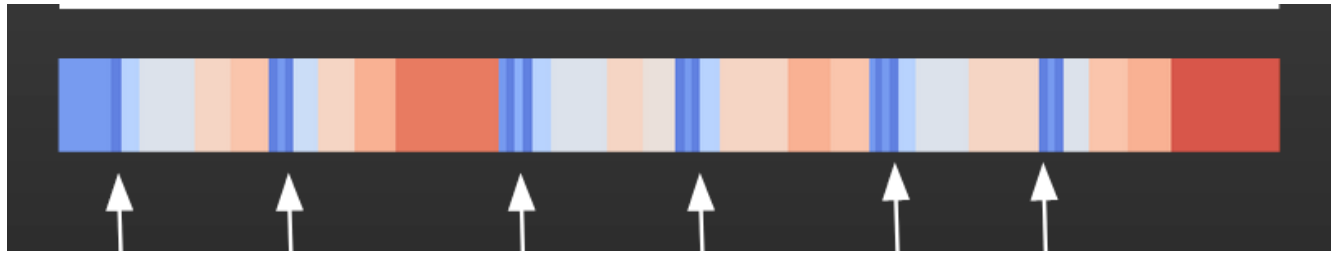
kiB: kibibyte (2^{10} bytes)

LIDAR: laser imaging, detection, and ranging

URI: uniform resource identifier

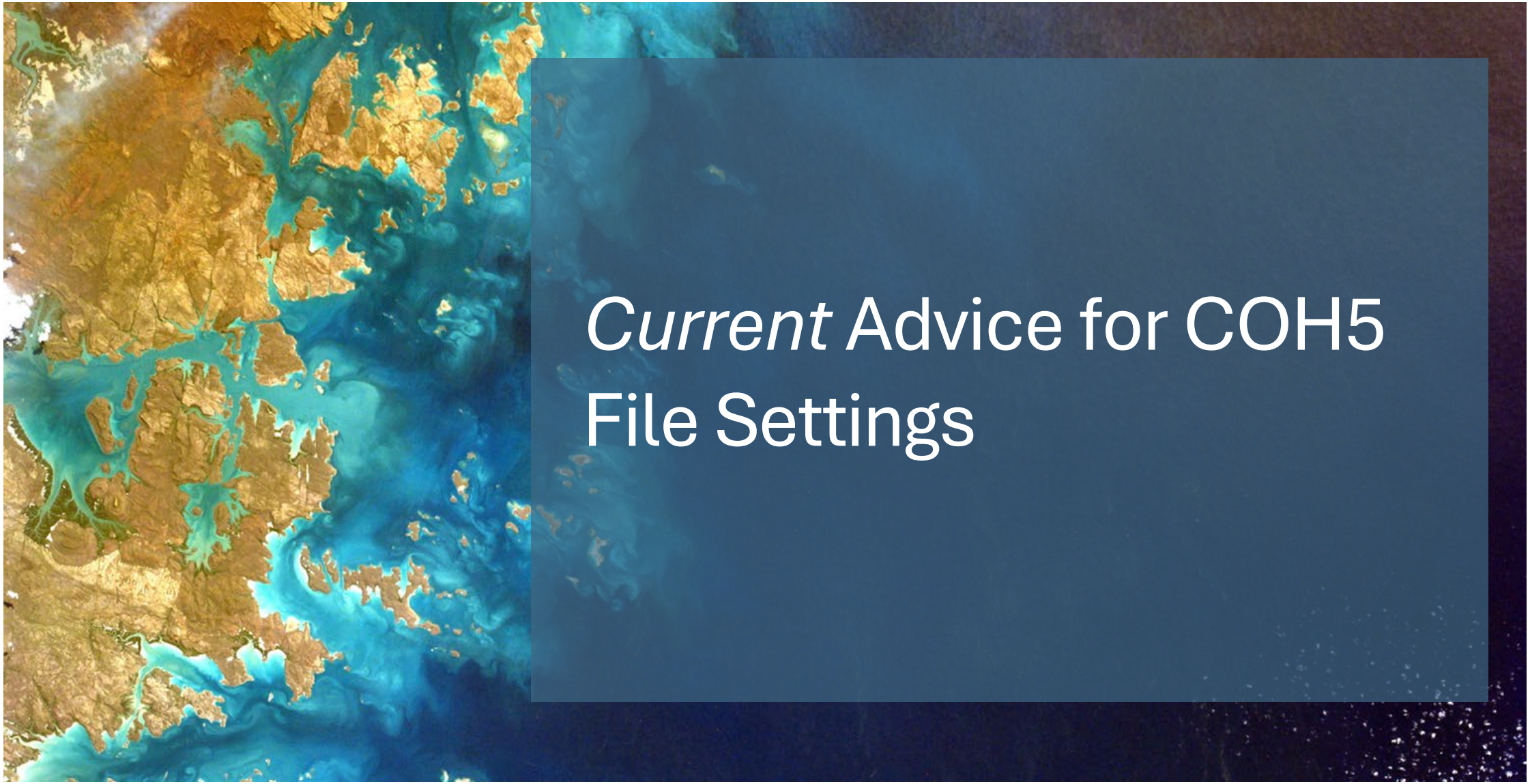
What are cloud optimized HDF5 files?

- Valid HDF5 files. **Not** a new file format or convention.
- Larger dataset chunk sizes.
- Internal file metadata consolidated into bigger contiguous blocks.



From “HDF at the Speed of Zarr” by Luis Lopez, NASA NSIDC.

- Total number of required S3 requests is significantly reduced which directly improves performance.
- For detailed information, see my 2023 ESIP Summer talk.



Current Advice for COH5 File Settings

Larger dataset chunk sizes

- Term clarification:
  - chunk *shape* = number of array elements in each chunk dimension
  - chunk *size* = number of bytes
    (number of array elements in a chunk multiplied by byte size of one array element)
- Chunk size is prior to any filtering (compression, etc.) applied.
- Not enough testing so far:
  - EOSDIS granules with larger dataset chunks are rare.
  - *h5repack* tool is not easy to use for large rechunking jobs.
- Larger chunks = less of them = less internal file metadata.

Consolidation of internal metadata

- Three different consolidation methods (see the YouTube video on slide #3).
- Practically only one of them tested: files created with the *paged aggregation file space management strategy*. (Easier to pronounce: *paged files*.)
- An HDF5 file is divided into pages. Page size set at file creation.
- Each page holds either internal metadata or data (chunks).

Paged file: pros and cons

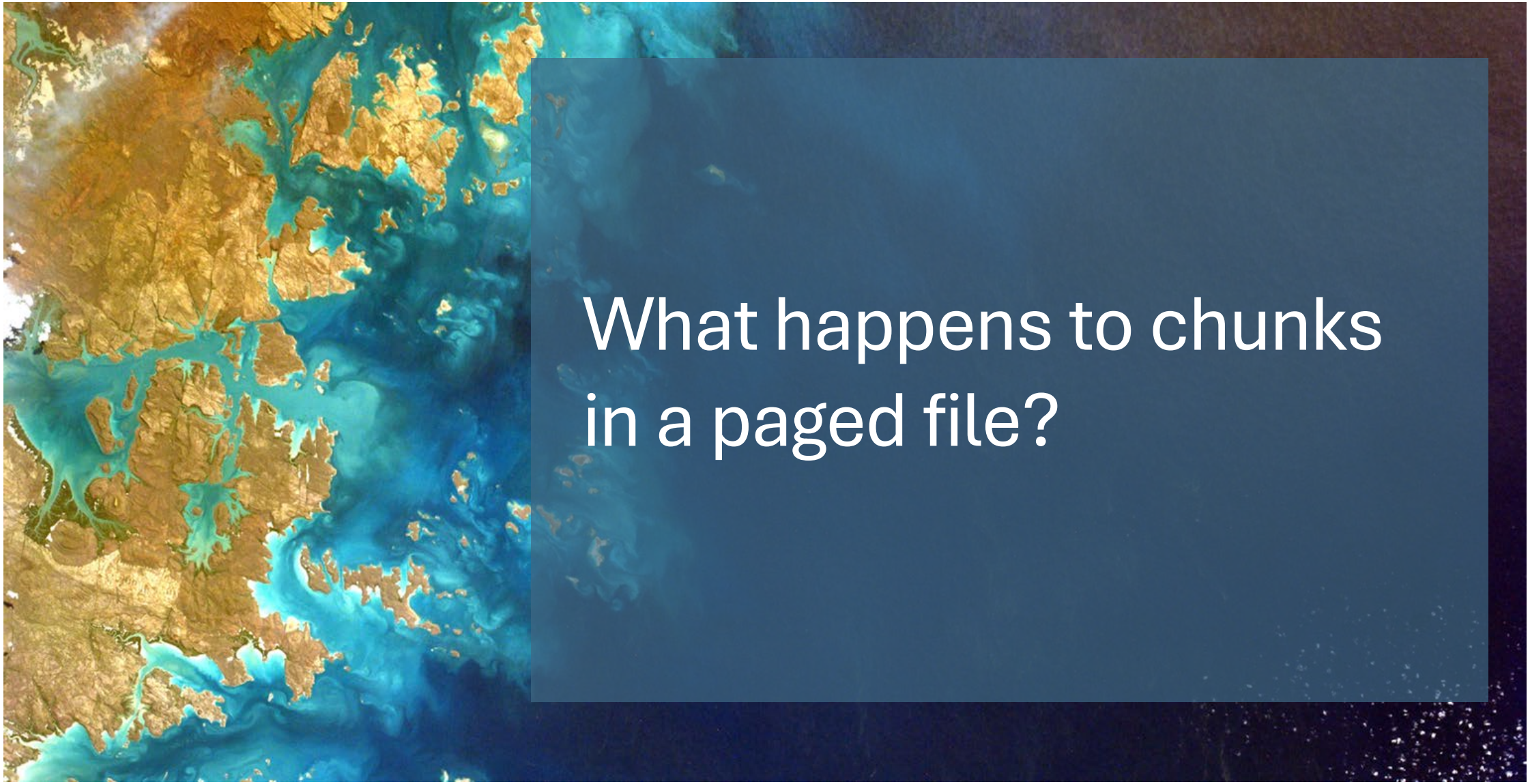
- HDF5 library reads entire pages which yields its best cloud performance.
- It also has a special cache for these pages, called *page buffer*. Its size must be set prior to opening a file.
- One file page can have more than one chunk = less overall S3 requests.
- Paged files tend to have larger size compared to their non-paged version which is caused by extra unused space in each page.
  - Think of a file page as a box filled with different sized objects.

Current Advice: Chunks

- Chunk size needs to account for speed of applying filters (e.g., decompression) when chunks are read.
- NASA satellite data predominantly compressed with the *zlib* (a.k.a., *gzip*, *deflate*) method.
- Need to explore other compression methods for optimal speed vs. compression ratio.
- **Smaller compressed chunks fill file pages better.**
- Suggested chunk sizes: 100k(i)B to 2M(i)B.

Current Advice: Paged files

- Tested file pages of 4, 8, and 16 MiB sizes.
- 8 MiB file page produced slightly better performance, with tolerable (<5%) file size increase.
- Majority of tested files had their internal metadata in one 8MiB file page.
- Don’t worry about unused space in that one file page for internal metadata.
- Majority of datasets in the tested files were stored in a single file page.
- Consider a minimum of four chunks per file page when choosing a dataset’s chunk size.
- **If writing data to a paged file in more than one open-close session, enable re-use of free space in the file when creating it.**
  - Otherwise, the file may end up much larger than needed.
  - *h5repack* can produce a defragmented version of the file.


What happens to chunks
in a paged file?

Example: GEDI Level 2A granule

- Global Ecosystem Dynamics Investigation (GEDI) instrument is on the International Space Station.
- A full-waveform LIDAR system for high-resolution observations of forests' vertical structure.
- Example granule:
  - 1,518,338,048 bytes
  - 136 contiguous datasets
  - 4,184 chunked datasets compressed with the *zlib* filter
- Repacked into a paged file version with 8MiB file page size.
- No chunk was "hurt" (i.e., rechunked) during repacking.

Chunk sizes

#	Chunk size in bytes	# chunked datasets	% of total chunk. datasets	cusum % of total chunk. datasets
0	0e+00 ≤ # < 1e+01	1,152	27.53	27.53
1	1e+01 ≤ # < 1e+03	0	0.0	27.53
2	1e+03 ≤ # < 1e+04	16	0.38	27.92
3	1e+04 ≤ # < 1e+05	72	1.72	29.64
4	1e+05 ≤ # < 1e+06	2,320	55.45	85.09
5	1e+06 ≤ # < 1e+07	432	10.33	95.41
6	1e+07 ≤ # < inf	192	4.59	100.0

Number of stored dataset chunks

#	Chunks stored	# chunked datasets	% of total chunk. datasets	cusum % of total chunk. datasets
0	No chunks	0	0.0	0.0
1	1 chunk	1,152	27.53	27.53
2	2-9 chunks	2,944	70.36	97.9
3	10-99 chunks	88	2.1	100.0
4	100-999 chunks	0	0.0	100.0
5	1000-9999 chunks	0	0.0	100.0
6	10,000-99,999 chunks	0	0.0	100.0
7	100,000 or more chunks	0	0.0	100.0

Dataset chunk spread across file pages

#	# of file pages holding all chunks	# chunked datasets	% of total chunk. datasets	cusum % of total chunk. datasets
0	1 page	3,562	85.13	85.13
1	2 pages	591	14.13	99.26
2	3 pages	6	0.14	99.4
3	4 pages	9	0.22	99.62
4	5 pages	8	0.19	99.81
5	6 - 9 pages	8	0.19	100.0
6	10 - 14 pages	0	0.0	100.0
7	15 - 19 pages	0	0.0	100.0
8	20 - 24 pages	0	0.0	100.0
9	25 - 29 pages	0	0.0	100.0
10	30 or more pages	0	0.0	100.0

Extra file pages compared to dataset total size

#	# file pages anomaly	# chunked datasets	% of total chunk. datasets	cusum % of total chunk. datasets
0	No extra file pages	3,562	85.13	85.13
1	1 extra file page	591	14.13	99.26
2	2 extra file pages	6	0.14	99.4
3	3 extra file pages	9	0.22	99.62
4	4 extra file pages	8	0.19	99.81
5	5 or more extra file pages	8	0.19	100.0

Dataset cache size for all chunks?

#	Chunk cache size	# chunked datasets	% of total chunk. datasets	cusum % of total chunk. datasets
0	1 MiB	3,232	77.25	77.25
1	4 MiB	328	7.84	85.09
2	8 MiB	0	0.0	85.09
3	16 MiB	432	10.33	95.41
4	> 16 MiB	192	4.59	100.0

ESDIS
EARTH SCIENCE DATA AND INFORMATION SYSTEM

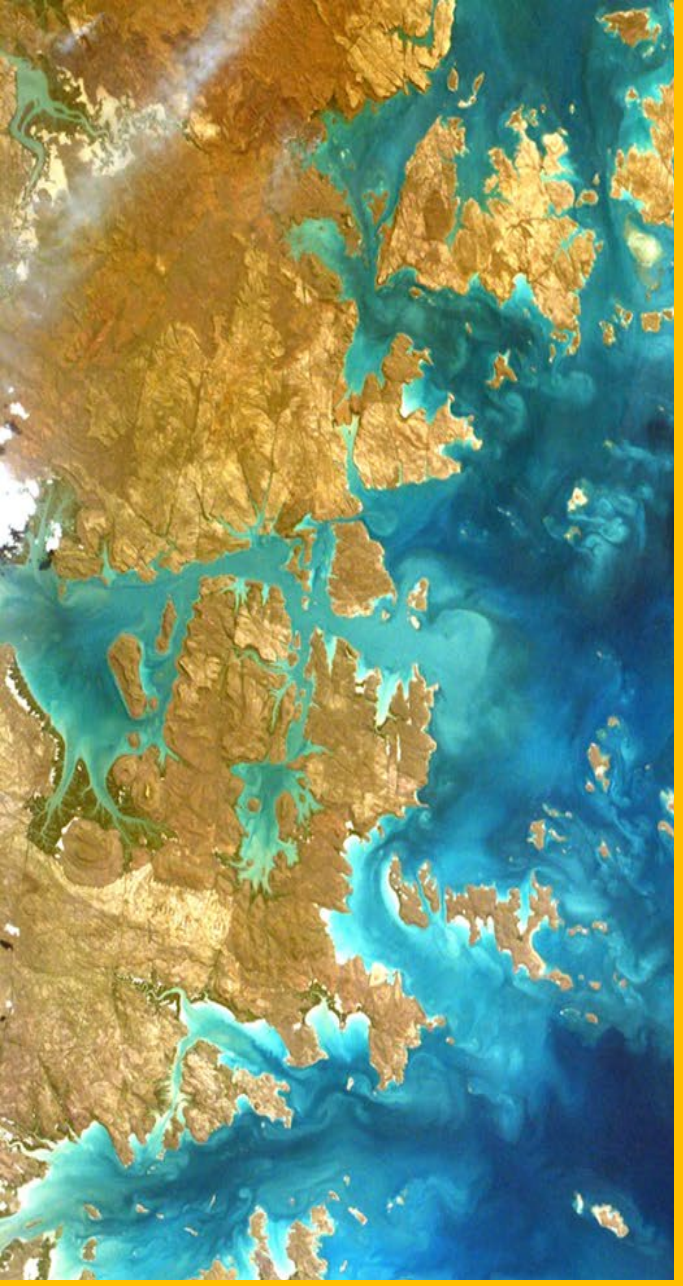
HDF5 Library Improvements for Cloud Data Access

HDF5 library

- Applies to version 1.14.4 only.
- Released in May 2024.
- All other maintenance releases of the library – 1.8.*, 1.10.*, and 1.12.* – are deprecated now.
- Native method for S3 data access: Read-Only S3 (ROS3) virtual file driver (VFD).
  - Not always available – build dependent.
  - Conda Forge *hdf5* package has it but **not** *h5py* from PyPI.
- For Python users: *fsspec* via *h5py*.
  - *fsspec* connected with the library using its virtual file layer API.
  - Lacks communication of important information from the library.

Notable improvements

- ROS3 caches first 16 MiB of the file on open.
- Set-and-forget page buffer size. Opening non-paged files will not cause an error.
- Fixed chunk file location info to account for file's user block size.
- Fixed an h5repack bug for datasets with variable-length data. Important when repacking netCDF-4 string variables.
- **Next release**: Build with *zlib-ng*. This is a newer open-source implementation of the standard *zlib* compression library and ~2x faster.
- **Next release**: *h5repack*, *h5ls*, *h5dump*, and *h5stat* new command-line option for page buffer size. This will enable **much** improved performance for cloud optimized files in S3.
- **Next release**: ROS3 will support relevant AWS environment variables.
- **Next release**: Support for S3 object URIs (`s3://bucket-name/object-name`).



This work was supported by NASA/GSFC under Raytheon Company contract number 80GSFC21CA001

ESDIS
EARTH SCIENCE DATA AND INFORMATION SYSTEM

NASA