

HDF5 1.6.0

| | |
|---------------|--|
| Version | HDF5 1.6.0 |
| Release Date | 2003-07-03 |
| Download | /ftp/HDF5/releases/hdf5-1.6/hdf5-1.6.0.tar.gz |
| Release Notes | Release Notes |

Release Notes:

HDF5 version 1.6.0-pre1 currently under development

=====

INTRODUCTION

This document describes the differences between HDF5-1.4.* and HDF5-1.6.0, and contains information on the platforms tested and known problems in HDF5-1.6.0. For more details check the HISTORY.txt file in the HDF5 source.

The HDF5 documentation can be found on the NCSA ftp server (ftp.ncsa.uiuc.edu) in the directory:

`/HDF/HDF5/docs/`

For more information look at the HDF5 home page at:

`http://hdf.ncsa.uiuc.edu/HDF5/`

If you have any questions or comments, please send them to:

`hdfhelp@ncsa.uiuc.edu`

CONTENTS

- New Features
- Support for new platforms and languages
- Bug Fixes since HDF5-1.4.0
- Platforms Tested
- Known Problems

New Features

=====

Configuration:

The following flags have been added to the configuration script:

- `--enable-hdf5v1_4` Compile the HDF5 v1.4 compatibility interface
- `--enable-filters=all` Turn on all internal I/O filters. One may also specify a comma-separated list of filters or the word no. The default is all internal I/O filters.
- `--with-mpe=DIR` Use MPE instrumentation [default=no]

--with-szlib=DIR Use szlib library for external szlib I/O filter
 [default=no]

Library:

Summary: This release has the following new features that are not available in 1.4.* releases

1. Generic properties to give application more control on I/O pipeline
2. Time allocation and fill value properties
3. New filters: external compression filter szip
 internal shuffling and checksum filters
4. Compact storage layout for datasets
5. Redesigned I/O pipeline for better performance.

For more information see

http://hdf.ncsa.uiuc.edu/HDF5/doc_resource/SZIP/index.html

<http://hdf.ncsa.uiuc.edu/HDF5/doc/ADGuide.html>

<http://hdf.ncsa.uiuc.edu/HDF5/doc/ADGuide/Changes.html>

Complete list of changes:

- * Changed dataset modification time to `_not_` be updated when raw data is written to a dataset. The modification time is only updated when the dataset's metadata is modified. QAK - 2003/06/10
- * Changed `H5Sget_select_bounds` to use `'hssize_t *'` for start and end parameters, instead of `'hsize_t *'`, to better match other parts of the dataspace API. QAK - 2003/06/04
- * Changed raw data I/O to only access each chunk once, improving performance in many situations with hyperslabs and large chunks or chunks with filters. These improvements are currently limited to serial I/O, with similar parallel I/O improvements forthcoming sometime in the future. QAK - 2003/05/07
- * Added parameter to the MPI/POSIX driver. If GPFS is enabled (by modifying the `H5FDmpiposix.c` file to uncomment the `USE_GPFS_HINTS` macro), then this extra parameter will turn GPFS hints on and off during runtime. BW - 2003/05/05
- * Added option to print 1-byte integer datasets as ASCII to `h5dump`. BW - 2003/04/30
- * Added a new utility "h5fc". It can be used to compile easily Fortran programs which use HDF5. It automatically uses the compiler the HDF5 library was built with and links in any libraries HDF5 requires. BW - 2003/04/10
- * Added new dataset creation property list functions for working with I/O filters: `H5Pmodify_filter`, `H5Pget_filter_by_id` and `H5Pall_filters_avail`. Also changed `H5Zregister` to use new method of registering filters with library. QAK - 2003/04/08
- * The first version of szip compression support were implemented. User should have static szlib library installed. Using function `H5Pset_szip` to pass the szip parameters to the HDF5 library. More detailed description of the process will be followed. KY-2003/04/01
- * Added Fletcher32 checksum as a filter in pipeline. It only works in chunked dataset. SLU - 2003/2/11
- * MPICH/MPE instrumentation feature added. Use `--with-mpe[=DIR]` to configure it. AKC - 2003/1/3
- * New functions `H5Gget_num_objs`, `H5Gget_objname_by_idx` and `H5Gget_objtype_by_idx` are added to the library. SLU - 2002/11/25
- * `H5Dget_offset` is added to return the offset of a dataset's data relative

- to the beginning of the file. SLU - 2002/11/7
- * Functions H5Tget_native_type and H5Tis_variable_str are added. The first one reconstructs a datatype based on native memory datatype. The second one checks if a datatype is variable string. SLU - 2002/11/6
 - * Added environment variable "HDF5_DISABLE_VERSION_CHECK", which disables the version checking between the header files and the library linked into an application if set to '1'. This should be used with caution, mismatched headers and library binaries can cause `_serious_` problems. QAK - 2002/10/15
 - * Added new API function to get the name of an object in a file, using an open ID (`hid_t`). QAK - 2002/10/14
 - * Added API functions to return pointer to low-level file handle (`H5Fget_vfd_handle` and `H5FDget_vfd_handle`) and related property list setting functions (`H5Pset_family_offset` and `H5Pset_multi_type`). SLU - 2002/09/30
 - * Changed "H5P[set|get]_space_time" functions to "H5P[set|get]_alloc_time" Unify all symbolic names for these functions to use "alloc time" instead of other names. QAK - 2002/09/13
 - * Added "H5D_SPACE_ALLOC_INCR" setting to `H5D_SPACE_ALLOC_EARLY` and `H5D_SPACE_ALLOC_LATE` for `H5Dset_space_time()`. This allows chunked datasets to be incrementally allocated as in the 1.4.x branch. QAK - 2002/08/27
 - * Compact dataset is added to the library. The data will be stored in the header message of dataset layout. Space allocation time has to be `EARLY`. No hyperslab is supported for parallel collective write. There is no API changes except activating `H5Pset_layout` and `H5Pget_layout` for compact dataset. -SLU, 2002/8/20
 - * Added 'closing' parameter to VFL 'flush' callback function and `H5FDflush`. This allows the library to indicate that the file will be closed immediately following the call to 'flush' and can be used to avoid actions that are duplicated in the VFL 'close' callback function. QAK - 2002/05/20
 - * Added feature to parallel chunk allocation routine to not write fill values to chunks allocated if the user has set the "fill time" to never. This can improve parallel I/O performance for chunked datasets. QAK - 2002/05/17
 - * New functions `H5Glink2` and `H5Gmove2` were added to allow link and move to be in different locations in the same file. The old functions `H5Glink` and `H5Gmove` remain valid. SLU - 2002/04/26
 - * Fill-value's behaviors for contiguous dataset have been redefined. Basically, dataset won't allocate space until it's necessary. Full details are available at http://hdf.ncsa.uiuc.edu/RFC/Fill_Value, at this moment. SLU - 2002/04/11
 - * Added new routine "H5Dfill" to fill a selection with a particular value in memory. QAK - 2002/04/09
 - * Improved performance of "regular" hyperslab I/O when using MPI-IO and the datatype conversion is unnecessary. QAK - 2002/04/02
 - * Improved performance of single hyperslab I/O when datatype conversion is unnecessary. QAK - 2002/04/02
 - * Added new "H5Sget_select_type" API function to determine which type of selection is defined for a dataspace ("all", "none", "hyperslab" or "point"). QAK - 2002/02/07
 - * Added support to read/write portions of chunks directly, if they are uncompressed and too large to cache. This should speed up I/O on chunked datasets for a few more cases. QAK - 2002/01/31
 - * Added `H5Rget_obj_type()` API function, which performs the same functionality as `H5Rget_object_type()`, but requires the reference type as a parameter in order to correctly handle dataset region references. Moved `H5Rget_object_type()` to be only compiled into the library when v1.4 compatibility is enabled.

- * Added a new file access property, file close degree, to control file close behavior. It has four values, H5F_CLOSE_WEAK, H5F_CLOSE_SEMI, H5F_CLOSE_STRONG, and H5F_CLOSE_DEFAULT. Two correspond functions H5Pset_fcclose_degree and H5Pget_fcclose_degree are also provided. Two new functions H5Fget_obj_count and H5Fget_obj_ids are offered to assist this new feature. For full details, please refer to the reference manual under the description of H5Fcreate, H5Fopen, H5Fclose and the functions mentioned above.
- * Removed H5P(get|set)_hyper_cache API function, since the property is no longer used.
- * Improved performance of non-contiguous hyperslabs (built up with several hyperslab selection calls).
- * Improved performance of single, contiguous hyperslabs when reading or writing.
- * As part of the transition to using generic properties everywhere, the parameter of H5Pcreate changed from H5P_class_t to hid_t, as well the return type of H5Pget_class changed from H5P_class_t to hid_t. Further changes are still necessary and will be documented here as they are made.
- * Improved regular hyperslab I/O by about a factor of 6 or so.
- * Modified the Pablo build procedure to permit building of the instrumented library to link either with the Trace libraries as before or with the Pablo Performance Capture Facility.
- * Added new F90 APIs for generic properties, new filters, and time/space allocation properties.
- * C++ API:
 - Added two new member functions: Exception::getFuncName() and Exception::getCFuncName() to provide the name of the member function, where an exception is thrown.
 - IdComponent::operator= becomes a virtual function because DataType, DataSpace, and PropList provide their own implementation. The new operator= functions invoke H5Tcopy, H5Scopy, and H5Pcopy to make a copy of a datatype, dataspace, and property list, respectively.

Parallel Library:

Tools:

- * When the "-S" option for "simple" output is chosen, h5ls now displays modification times of datasets in UTC instead of local time.
QAK - 2003/06/06
- * h5diff to compare two HDF5 files was added
- * h5import to import ascii and binary data to an HDF5 file was added. Old h5import tool in the tools/misc directory was renamed to h5createU8 to reflect its purpose. h5createU8 will be deleted in 1.6.1 release.
- * Two new scripts h5fc and h5c++ were added to compile F90 and C++ HDF5 applications.

Support for new platforms, languages and compilers.

=====

- * Added C++ API support on HPUX11.00. BMR - 2003/03/19
- * Absoft compiler is supported for Fortran HDF5 Library. When building with Absoft compiler, add -DH5_ABSOFT to C compilation flags to get correct names of C functions called by Fortran APIs.

Bug Fixes since HDF5-1.4.0 release

=====

Library

- * Don't attempt to perform collective I/O on chunked datasets with parallel I/O. QAK - 2003/06/05
- * The library now correctly reuses space when objects are deleted in the file. This should be handled correctly for every situation, except datasets with variable-length datatypes are not returning the space they use in the global heap currently. QAK - 2003/04/13
- * Fixed error in B-tree deletion routine which could cause groups to be corrupted when objects are removed from them. QAK - 2003/04/11
- * Fixed error in file space freeing code which could cause metadata to fail to be written to the file. QAK - 2003/04/11
- * -O caused errors in AIX 5.x platforms. Removed it from --enable-production mode. AKC - 2003/03/31
- * Corrected memory/resource leaks in per-thread key information when thread-safe operation was enabled. QAK - 2003/02/07
- * Improved error assertion for nil VL strings, making it fail with error stack instead of just assertion failure. SLU - 2002/12/16
- * Added two new API functions: H5Zunregister & H5Zfilter_avail. QAK - 2002/11/16
- * Add data shuffle filter(source code H5Zshuffle.c), the combination of the shuffling and compression can make data compression better without suffering much encoding and decoding CPU time for many application datasets(especially for floating point data). This adds a new API function: H5Pset_shuffle. KY - 2002/11/13
- * Allow scalar dataspace to be used for parallel I/O. QAK - 2002/11/05
- * New functions H5Gget_comment(modification), H5Aget_storage_size, H5Arename. SLU - 2002/10/29
- * Fixed an assertion of H5S_select_iterate that did not account for scalar type that has no dimension sizes. AKC - 2002/10/15
- * Partially fixed space allocation inefficiencies in the file by improving our algorithms for re-using freed space. QAK - 2002/08/27
- * Fixed data corruption problem which could occur when fill values were written to a contiguously stored dataset in parallel. QAK - 2002/08/27
- * Fixed VL memory leak when data is overwritten. The heap objects holding old data are freed. If the fill value writing time is set to H5D_FILL_TIME_NEVER, the library prohibits user to create VL type dataset. The library free all the heap objects storing VL type if there is nested VL type(a VL type contains another VL type). SLU - 2002/07/10
- * Tweaked a few API functions to use 'size_t' instead of 'unsigned' or 'hsize_t', which may cause errors in some cases.

Configuration

- * Included the both the examples of fortran and c++ "make check-install" testing. This tests the correctness of the h5fc command. AKC - 2003/04/22
- * When using gcc 3.x, we use -std=c99 instead of -ansi for compiling. QAK - 2003/04/11
- * IA64 platform has its own configure setting and use Intel Compilers as the default compilers (were gcc and pgf90 before.) This also eliminated the segmentation fault in the fortran test. The missing reference of "exit" is fixed too. AKC - 2003/04/02

Performance

- * Improved dataset creation time by about 30% (relative to the 1.4.x branch).

Tools

- * Added a -force option to h5redeploy. AKC - 2003/03/04
- * The VL string bug(data and datatype cannot be shown) in h5dump is fixed. -SLU - 2002/11/18
- * Fixed segfault if h5dump was invoked with some options but no file (e.g., h5dump -H). -AKC, 2002/10/15
- * Fixed so that the "-i" flag works correctly with the h5dumper.
- * Fixed segfault when "-v" flag was used with the h5dumper.

Documentation

Platforms Tested

=====

| | |
|--------------------------|---|
| AIX 5.1 (32 and 64-bit) | C for AIX Compiler, Version 6.0.0.4 xlf 8.1.0.3 poe 3.2.0.10 |
| Cray T3E sn6606 2.0.6.08 | Cray Standard C Version 6.6.0.2 Cray Fortran Version 3.6.0.0.2 mpt 2.2.0.0 |
| Cray SV1 sn9617 10.0.1.2 | Cray Standard C Version 6.6.0.2 mpt 2.2.0.0 |
| Cray T90IEEE 10.0.1.01y | Cray Fortran Version 3.6.0.0.2 Cray Standard C Version 6.4.0.2.3 Cray Fortran Version 3.4.0.3 mpt 2.1.0.0 |
| FreeBSD 4.7 | gcc 2.95.4 g++ 2.95.5 |
| HP-UX B.11.00 | HP C HP92453-01 A.11.01.20 HP F90 v2.4 HP ANSI C++ B3910B A.03.13 MPIch 1.2.4 |
| IRIX 6.5 | MIPSpro cc 7.30 |
| IRIX64 6.5 (64 & n32) | MIPSpro cc 7.3.1.3m F90 MIPSpro 7.3.1.3m (64 only) MPIch 1.2.4 |
| Linux 2.4.18 | gcc 2.96, 3.2.2, 3.2.3 g++ 3.2.2, 3.2.3 Intel(R) C++ Version 7.1 Intel(R) Fortran Compiler Version 7.1 PGI compilers (pgcc, pgf90) version 4.0-2 MPIch 1.2.4 |
| OSF1 V5.1 | Compaq C V6.4-014 Compaq Fortran X5.4A-1684 |

| | |
|---------------------------|--|
| | Compaq C V6.3-027 , MPI_64bit_R5 |
| | gcc version 3.0 for C++ |
| SunOS 5.7 | WorkShop Compilers 5.0 98/12/15 C 5.0 |
| (Solaris 2.7) | WorkShop Compilers 5.0 98/12/15 C++ 5.0 |
| | WorkShop Compilers 5.0 98/10/25 |
| | FORTRAN 90 2.0 Patch 107356-04 |
| SunOS 5.8/32 | Sun WorkShop 6 update 2 C 5.3 |
| (Solaris 2.8) | Sun WorkShop 6 update 2 Fortran 90 |
| | Sun WorkShop 6 update 2 C++ 5.3 |
| SunOS 5.8/64 | Sun WorkShop 6 update 2 C 5.3 |
| (Solaris 2.8) | Sun WorkShop 6 update 2 Fortran 90 |
| | Sun WorkShop 6 update 2 C++ 5.3 |
| TFLOPS r1.0.4 v4.3.3 i386 | pgcc Rel 3.1-4i with mpich-1.2.4 with local modifications |
| IA-32 Linux 2.4.9 | gcc 2.96 |
| | Intel(R) C++ Version 7.0 |
| | Intel(R) Fortran Compiler Version 7.0 |
| IA-64 Linux 2.4.16 ia64 | gcc version 2.96 20000731 |
| | Intel(R) C++ Version 7.0 |
| | Intel(R) Fortran Compiler Version 7.0 |
| Windows 2000 (NT5.0) | MSVC++ 6.0 |
| | DEC Visual Fortran 6.0 |
| | Intel C and F90 compilers version 7.1 |
| Code Warrior 8.0 | |
| Windows XP | MSVC++.NET |
| MAC OS X | Darwin 6.5 |
| | gcc and g++ Apple Computer, Inc. GCC version 1161, based on gcc version 3.1 |

Supported Configuration Features Summary

=====

In the tables below

y = tested and supported
n = not supported or not tested in this release
x = not working in this release
dna = does not apply
() = footnote appears below second table

| Platform | C | C parallel | F90 | F90 parallel | C++ | Shared libraries (4) | zlib |
|-------------------|---|---------------|-----|-----------------|-----|-------------------------|------|
| Solaris2.7 64-bit | y | y (1) | y | y (1) | y | y | y |
| Solaris2.7 32-bit | y | y (1) | y | y (1) | y | y | y |
| Solaris2.8 64-bit | y | y (1) | y | y (1) | y | y | y |
| Solaris2.8 32-bit | y | y | y | y (1) | y | y | y |
| IRIX6.5 | y | y (1) | n | n | n | y | y |
| IRIX64_6.5 64-bit | y | y (2) | y | y | y | y | y |
| IRIX64_6.5 32-bit | y | y (2) | n | n | n | y | y |
| HPUX11.00 | y | y (1) | y | y | y | y | y |
| OSF1 v5.1 | y | y | y | y | y | y | y |
| T3E | y | y (5) | y | y (5) | n | n | y |
| SV1 | y | y (5) | y | y (5) | n | n | y |
| T90 IEEE | y | y (5) | y | y (5) | n | n | y |
| TFLOPS | n | y (1) | n | n | n | n | y |
| AIX-5.1 32-bit | y | y | y | y | y | n | y |
| AIX-5.1 64-bit | y | y | y | y | y | n | y |

| | | | | | | | | |
|----------------------|-----|---|-------|---------|---|---|---|---|
| WinXP | (6) | y | n | n | n | y | y | y |
| WinXP Intel | | y | n | n | n | y | y | y |
| Win2000 | | y | n | y | n | y | y | y |
| Win2000 Intel | | y | n | y | n | y | y | y |
| WinNT CW | | y | n | n | n | n | n | y |
| Mac OS X 10.2 | | y | n | n | n | y | y | y |
| FreeBSD | | y | y (1) | n | n | y | y | y |
| Linux 2.4 gcc | (3) | y | y (1) | y (PGI) | n | y | y | y |
| Linux 2.4 Intel | (3) | y | n | y | n | y | n | y |
| Linux 2.4 PGI | (3) | y | n | y | n | y | n | y |
| Linux 2.4 IA32 Intel | | y | n | y | n | y | n | y |
| Linux 2.4 IA64 Intel | | y | n | y | n | y | n | y |

ASCII Table 2 -- for RELEASE.txt

| Platform | static- exec | Thread- safe | SZIP | GASS | STREAM- VFD | High-level APIs | H4/H5 tools (7) |
|----------------------|-----------------|-----------------|------|------|----------------|--------------------|--------------------|
| Solaris2.7 64-bit | x | y | y | n | y | y | n |
| Solaris2.7 32-bit | x | y | y | n | y | y | y |
| Solaris2.8 64-bit | x | y | y | n | y | y | n |
| Solaris2.8 32-bit | x | y | y | n | y | y | y |
| IRIX6.5 | x | n | y | n | y | y | y |
| IRIX64_6.5 64-bit | x | y | y | y | y | y | y |
| IRIX64_6.5 32-bit | x | y | y | y | y | y | y |
| HPUX11.00 | x | n | y | n | y | y | y |
| OSF1 v5.1 | y | n | y | n | y | y | y |
| T3E | y | n | n | n | y | y | y |
| SV1 | y | n | n | n | y | y | y |
| T90 IEEE | y | n | n | n | y | y | n |
| TFLOPS | y | n | n | n | n | y | y |
| AIX-5.1 32-bit | y | n | y | n | y | y | y |
| AIX-5.1 64-bit | y | n | y | n | y | y | y |
| WinXP | (6) | y | n | y | n | n | y |
| WinXP Intel | | y | n | y | n | n | y |
| Win2000 | | y | n | y | n | n | y |
| Win2000 Intel | | y | n | y | n | n | y |
| WinNT CW | | y | n | y | n | n | y |
| Mac OS X 10.2 | | y | n | y | n | y | n |
| FreeBSD | | y | y | y | n | y | y |
| Linux 2.4 gcc | (3) | y | y | y | n | y | y |
| Linux 2.4 Intel | (3) | y | n | y | n | y | n |
| Linux 2.4 PGI | (3) | y | n | y | n | y | n |
| Linux 2.4 IA32 Intel | | y | n | y | n | y | y |
| Linux 2.4 IA64 Intel | | y | n | y | n | y | y |

- Notes: (1) Using mpich 1.2.4.
(2) Using mpt and mpich 1.2.4.
(3) Linux 2.4 with GNU, Intel, and PGI compilers, respectively.
(4) Shared libraries are provided only for the C library, except on Windows where they are provided for C and C++.
(5) Using mpt.
(6) Binaries only; source code for this platform is not being released at this time.
(7) Includes the H4toH5 Library and the h4toh5 and h5toh4 utilities.

Compiler versions for each platform are listed in the preceding "Platforms Tested" table.

Known Problems

=====

- * PGI C++ compiler fails when compiling the C++ library's tests. Therefore, we cannot verify that the C++ library built with the PGI C++ compiler is correct.
- * The h5dump tests may fail to match the expected output on some platforms (e.g. parallel jobs, Windows) where the error messages directed to "stderr" do not appear in the "right order" with output from stdout. This is not an error.
- * The stream-vfd test uses ip port 10007 for testing. If another application is already using that port address, the test will hang indefinitely and has to be terminated by the kill command. To try the test again, change the port address in test/stream_test.c to one not being used in the host.
- * The --enable-static-exec configure flag fails to compile for Solaris platforms. This is due to the fact that not all of the system libraries on Solaris are available in a static format.

The --enable-static-exec configure flag also fails to correctly compile on IBM SP2 platform for the serial mode. The parallel mode works fine with this option.

It is suggested that you don't use this option on these platforms during configuration.

- * With the gcc 2.95.2 compiler, HDF 5 uses the '-ansi' flag during compilation. The ANSI version of the compiler complains about not being able to handle the 'long long' datatype with the warning:

```
warning: ANSI C does not support 'long long'
```

This warning is innocuous and can be safely ignored.

- * The Stream VFD was not tested yet under Windows. It is not supported in the TFLOPS machine.
- * The ./dsets tests failed in the TFLOPS machine if the test program, dsets.c, is compiled with the -O option. The hdf5 library still works correctly with the -O option. The test program works fine if it is compiled with -O1 or -O0. Only -O (same as -O2) causes the test program to fail.
- * Certain platforms give false negatives when testing h5ls:
 - Cray J90 and Cray T90IEEE give errors during testing when displaying some floating-point values. These are benign differences due to the different precision in the values displayed and h5ls appears to be dumping floating-point numbers correctly.
- * Before building HDF5 F90 Library from source on Crays replace H5Aff.f90, H5Dff.f90 and H5Pff.f90 files in the fortran/src subdirectory in the top level directory with the Cray-specific files

from the site:

- * On some platforms that use Intel compilers to build HDF5 fortran library, compilation may fail for fortranlib_test.f90, fflush1.f90 and fflush2.f90 complaining about exit subroutine. Comment out the line
IF (total_error .ne. 0) CALL exit (total_error)

ftp://hdf.ncsa.uiuc.edu/pub/outgoing/hdf5/hdf5-1.6.0/F90_source_for_Crays

- * On IA32 and IA64 systems, if you use a compiler other than GCC (such as Intel's ecc or icc compilers), you will need to modify the generated "libtool" program after configuration is finished. On or around line 104 of the libtool file, there are lines which look like:

```
# How to pass a linker flag through the compiler.  
wl=""
```

change these lines to this:

```
# How to pass a linker flag through the compiler.  
wl="-Wl,"
```

UPDATE: This is now done automatically by the configure script. However, if you still experience a problem, you may want to check this line in the libtool file and make sure that it has the correct value.

* Information about building with PGI and Intel compilers is available in
INSTALL file sections 5.7 and 5.8