HDF5 1.8.0

Version	HDF5 1.8.0
Release Date	2008-02-12
Download	/ftp/HDF5/releases/hdf5-1.8/hdf5-1.8.0/
Release Notes	Release Notes

Release Notes:

```
HDF5 version 1.8.0 released on Tue Feb 12 20:41:19 CST 2008
================================================================================


INTRODUCTION
============


This document describes the differences between the HDF5-1.6.x release series
and HDF5 1.8.0, and contains information on the platforms tested and known
problems in HDF5-1.8.0.  For more details, see the HISTORY-1_0-1_8_0_rc3.txt file in
the
release_docs/ directory of the HDF5 source.

Links to the HDF5 1.8.0 source code, documentation, and additional materials
can be found on the HDF5 web page at:

     http://www.hdfgroup.org/products/hdf5/

The HDF5 1.8.0 release can be obtained from:

     http://www.hdfgroup.org/HDF5/release/obtain5.html

User documentation for 1.8.0 can be accessed directly at this location:

     http://www.hdfgroup.org/HDF5/doc/

New features in 1.8.0, including brief general descriptions of some new
and modified APIs, are described in the "What's New in 1.8.0?" document:

     http://www.hdfgroup.org/HDF5/doc/ADGuide/WhatsNew180.html

All new and modified APIs are listed in detail in the "HDF5 Software Changes
from Release to Release" document, in the section "Release 1.8.0 (current
release) versus Release 1.6.x":

     http://www.hdfgroup.org/HDF5/doc/ADGuide/Changes.html

If you have any questions or comments, please send them to the HDF Help Desk:

     help@hdfgroup.org


CONTENTS
========
```

```
- New Features
- Removed Feature
- Support for new platforms and languages
- Bug Fixes since HDF5-1.6.0
- Platforms Tested
- Supported Configuration Features Summary
- Known Problems
```

New Features
============

    HDF5 Release 1.8.0 is a major release with many changes and new features.

    New format and interface features discussed in the "What's New in
    HDF5 1.8.0" document include the following:

        Enhanced group object management
        Enhanced attribute management and more efficient meta data handling
        Expanded datatype features
        Creation order tracking and indexing
        Improved meta data caching and cache control
        UTF-8 encoding
        New I/O filters: n-bit and scale+offset compression
        New link (H5L) and object (H5O) interfaces and features
        External and user-defined links
        New high-level APIs:
            HDF5 Packet Table (H5PT) and HDF5 Dimension Scale (H5DS)
        C++ and Fortran interfaces for older high-level APIs:
            H5Lite (H5LT), H5Image (H5IM), and H5Table (H5TB)
        New and improved tools
        And more...

        http://hdfgroup.org/HDF5/doc/ADGuide/WhatsNew180.html

    New APIs associated with these features, other interface changes
    (e.g., ENUM and struct definitions), and new library configuration flags
    are listed in the "Release 1.8.0 (current release) versus Release 1.6.x"
    section of "HDF5 Software Changes from Release to Release."

        http://hdfgroup.org/HDF5/doc/ADGuide/Changes.html

Compatibility

    Many HDF5 users and user communities have existing applications that
    they may wish to port to Release 1.8.0. Alternatively, some users may
    wish to take advantage of Release 1.8.0's improved performance without
    having to port such applications. To facilitate managing application
    compatibility and porting applications from release to release, the HDF
    Team has implemented the following features:
        Individually-configurable macros that selectively map common
            interface names to the old and new interfaces
        Library configuration options to configure the macro mappings

    Two related documents accompany this release:
        "API Compatibility Macros in HDF5" discusses the specifics of the
        new individually-configurable macros and library configuration
        options.

```
              http://hdfgroup.org/HDF5/doc/RM/APICompatMacros.html

              "New Features in HDF5 Release 1.8.0 and Backward/Forward Format
              Compatibility Issues" discusses each new feature with regard to
              its impact on format compatibility.
              http://hdfgroup.org/HDF5/doc/ADGuide/CompatFormat180.html

Referenced documents
--------------------
     http://hdfgroup.org/HDF5/doc/ADGuide/WhatsNew180.html
          "What's New in HDF5 1.8.0"

     http://hdfgroup.org/HDF5/doc/ADGuide/Changes.html
          The "Release 1.8.0 (current release) versus Release 1.6.x "
          section in "HDF5 Software Changes from Release to Release"

     http://hdfgroup.org/HDF5/doc/RM/APICompatMacros.html
          "API Compatibility Macros in HDF5"

     http://hdfgroup.org/HDF5/doc/ADGuide/CompatFormat180.html
          "New Features in HDF5 Release 1.8.0 and Backward/Forward Format
          Compatibility Issues"


Removed Feature
===============
The stream virtual file driver (H5FD_STREAM) have been removed in this
release. This affects the functions H5Pset_fapl_stream and H5Pget_fapl_stream
and the constant H5FD_STREAM.

This virtual file driver will be available at
http://hdf5-addons.origo.ethz.ch/. Note that at the time of this release,
the transition is still in progress; the necessary integration tools may
not be available when HDF5 Release 1.8.0 first comes out.


Support for New Platforms, Languages, and Compilers
===================================================
     - Support for Open VMS 7.3 was added.


Bug Fixes since HDF5-1.6.0
==========================
     This release contains numerous bug fixes. For details, see the
     "Changes from 1.6.0 to 1.8.0-rc3" section of the HISTORY.txt file for
     this release.


Platforms Tested
================
The following platforms and compilers have been tested for for this release.

     AIX 5.2 (32/64 bit)          xlc 8.0.0.11
                                  xlC 8.0
                                  xlf 10.01.0000.0
                                  mpcc_r 6.0.0.8
                                  mpxlf_r 8.1.1.7

     FreeBSD 6.2-STABLE i386      gcc 3.4.6 [FreeBSD] 20060305
```

```
(duty)                          g++ 3.4.6 [FreeBSD] 20060305
                                gcc 4.2.1 20080123
                                g++ 4.2.1 20080123
                                gfortran 4.2.1 20070620


FreeBSD 6.2-STABLE amd64        gcc 3.4.6 [FreeBSD] 20060305
(liberty)                       g++ 3.4.6 [FreeBSD] 20060305
                                gcc 4.2.1 20080123
                                g++ 4.2.1 20080123
                                gfortran 4.2.1 20080123


IRIX64 6.5 (64 & n32)           MIPSpro cc 7.4.4m
                                F90 MIPSpro 7.4.4m
                                C++ MIPSpro cc 7.4.4m


Linux 2.6.9 (RHEL4)             Intel 10.0 compilers
(abe.ncsa.uiuc.edu)


Linux 2.4.21-47                 gcc 3.2.3 20030502
(osage)


Linux 2.6.9-42.0.10            gcc 3.4.6 20060404
(kagiso)                        PGI 7.0-7 (pgcc, pgf90, pgCC)
                                Intel 9.1 (icc, ifort, icpc)


Linux 2.6.16.27 x86_64 AMD      gcc 4.1.0 (SuSE Linux), g++ 4.1.0,
(smirom)                            g95 (GCC 4.0.3)
                                PGI 6.2-5 (pgcc, pgf90, pgCC)
                                Intel 9.1 (icc, iort, icpc)


Linux 2.6.5-7.252.1-rtgfx #1    Intel(R) C++ Version 9.0
     SMP ia64                   Intel(R) Fortran Itanium(R) Version 9.0
(cobalt)                        SGI MPI


SunOS 5.8 32,46                 Sun WorkShop 6 update 2 C 5.3
 (Solaris 2.8)                  Sun WorkShop 6 update 2 Fortran 95 6.2
                                Sun WorkShop 6 update 2 C++ 5.3


SunOS 5.10                      cc: Sun C 5.8
(linew)                         f90: Sun Fortran 95 8.2
                                CC: Sun C++ 5.8


Xeon Linux 2.4.21-32.0.1.ELsmp-perfctr-lustre
(tungsten)                      gcc 3.2.2 20030222
                                Intel(R) C++ Version 9.0
                                Intel(R) Fortran Compiler Version 9.0


IA-64 Linux 2.4.21.SuSE_292.til1 ia64
(NCSA tg-login)                 gcc 3.2.2
                                Intel(R) C++ Version 8.1
                                Intel(R) Fortran Compiler Version 8.1
                                mpich-gm-1.2.5..10-intel-r2


Windows XP                      Visual Studio .NET
                                Visual Studio 2005 w/ Intel Fortran 9.1
                                Cygwin(native gcc compiler and g95)
                                MinGW(native gcc compiler and g95)


Windows XP x64                  Visual Studio 2005 w/ Intel Fortran 9.1
```

```
      Windows Vista                   Visual Studio 2005


      MAC OS  10.4 (Intel)            gcc i686-apple-darwin8-gcc-4.0.1 (GCC) 4.0.1
                                      G95 (GCC 4.0.3 (g95 0.91!) Nov 21 2006)


      Alpha Open VMS 7.3              Compaq C V6.5-001-48BCD
                                      HP Fortran V7.6-3276
                                      Compaq C++ V6.5-004
```

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
              <blank> = testing incomplete on this feature or platform
```

Platform	C	F90 parallel	F90 parallel	C++	zlib	SZIP
SunOS5.8 64-bit	n	y	n	y	y	y
SunOS5.8 32-bit	n	y	n	y	y	y
SunOS5.10 64-bit	y(1)	y	n	y	y	y
SunOS5.10 32-bit	y(1)	y	n	y	y	y
IRIX64_6.5 64-bit	n	y	y	y	y	y
IRIX64_6.5 32-bit	n	n	n	n	y	y
AIX-5.2 32-bit	y	y	y	y	y	y
AIX-5.2 64-bit	y	y	y	y	y	y
Windows XP	n	y(15)	n(15)	y	y	y
Windows XP x64	n	y(15)	n(15)	y	y	y
Windows Vista	n	n	n	y	y	y
Mac OS X 10.4 PowerPC	n		n			
Mac OS X 10.4 Intel	n	y	n	y	y	y
FreeBSD 4.11	n	n	n	y	y	y
RedHat EL3 W (3)	y(1a)	y(10)	y(1a)	y	y	y
RedHat EL3 W Intel (3)	n	y	n	y	y	y
RedHat EL3 W PGI (3)	n	y	n	y	y	y
SuSe x86_64 gcc (3,12)	y(1a)	y(11)	n	y	y	y
SuSe x86_64 Int (3,12)	n	y(13)	n	y	y	n
SuSe x86_64 PGI (3,12)	n	y(8)	n	y	y	y
Linux 2.4 Xeon C Lustre Intel (3,6)	n	y	n	y	y	n
Linux 2.6 SuSE ia64 C Intel (3,7)	y	y	y	y	y	n
Linux 2.6 SGI Altix ia64 Intel (3)	y	y	y	y	y	y
Alpha OpenVMS 7.3.2	n	y	n	y	n	n

Platform	Shared C libs	Shared F90 libs	Shared C++ libs	static- exec	Thread- safe
Solaris2.8 64-bit	y	y	y	x	y
Solaris2.8 32-bit	y	y	y	x	y
Solaris2.10 64-bit	y			x	y

```
Solaris2.10 32-bit       y                           x         y
IRIX64_6.5 64-bit        y        y        n         y         y
IRIX64_6.5 32-bit        y        dna      y         y         y
AIX-5.2 & 5.3 32-bit     n        n        n         y         n
AIX-5.2 & 5.3 64-bit     n        n        n         y         n
Windows XP               y        y(15)    y         y         y
Windows XP x64           y        y(15)    y         y         y
Windows Vista            y        n        n         y         y
Mac OS X 10.3            y                           y         n
FreeBSD 4.11             y        n        y         y         y
RedHat EL3  W      (3) y          y(10)    y         y         y
RedHat EL3  W Intel (3) y         y        y         y         n
RedHat EL3  W PGI   (3) y         y        y         y         n
SuSe x86_64 W GNU (3,12) y        y        y         y         y
SuSe x86_64 W Int (3,12) y        y        y         y(14)     n
SuSe x86_64 W PGI (3,12) y        y        y         y(14)     n
Linux 2.4 Xeon C
    Lustre Intel    (6) y         y        y         y         n
Linux 2.4 SuSE
    ia64 C Intel    (7) y         y        y         y         n
Linux 2.4 SGI Altix
    ia64 Intel          y                           y         n
Alpha OpenVMS 7.3.2      n        n        n         y         n


    Notes: (1)  Using mpich 1.2.6.
           (1a) Using mpich2 1.0.6.
           (2)  Using mpt and mpich 1.2.6.
           (3)  Linux 2.6 with GNU, Intel, and PGI compilers, as indicated.
                  W or C indicates workstation or cluster, respectively.

           (6)  Linux 2.4.21-32.0.1.  Xeon cluster with ELsmp_perfctr_lustre
                  and Intel compilers
           (7)  Linux 2.4.21, SuSE_292.till.  Ia64 cluster with Intel compilers
           (8)  pgf90
           (9)  With Compaq Visual Fortran 6.6c compiler.
           (10) With PGI and Absoft compilers.
           (11) PGI and Intel compilers for both C and Fortran
           (12) AMD Opteron x86_64
           (13) ifort
           (14) Yes with C and Fortran, but not with C++
       (15) Using Visual Studio 2005 or Cygwin
           (16) Not tested for this release.
    Compiler versions for each platform are listed in the preceding
    "Platforms Tested" table.


Known Problems
==============
* We have discovered two problems when running collective IO parallel HDF5
  tests with chunking storage on the ChaMPIon MPI compiler on tungsten, a
  Linux cluster at NCSA.

  Under some complex selection cases:
  1) MPI_Get_element returns the wrong value.
  2) MPI_Type_struct also generates the wrong derived datatype and corrupt
     data may be generated.
  These issues arise only when turning on collective IO with chunking storage
  with some complex selections. We have not found these problems on other
  MPI-IO compilers. If you encounter these problems, you may use independent
```

IO instead.

    To avoid this behavior, change the following line in your code
        H5Pset_dxpl_mpio(xfer_plist, H5FD_MPIO_COLLECTIVE);

    to
        H5Pset_dxpl_mpio(xfer_plist, H5FD_MPIO_INDEPENDENT);

    KY - 2007/08/24

* For SNL, spirit/liberty/thunderbird: The serial tests pass but parallel
  tests failed with MPI-IO file locking message. AKC - 2007/6/25

* On Intel 64 Linux cluster (RH 4, Linux 2.6.9) with Intel 10.0 compilers,
  use -mp -O1 compilation flags to build the libraries. A higher level of
  optimization causes failures in several HDF5 library tests.

* For SNL, Red Storm: Only parallel HDF5 is supported. The serial tests pass
  when run against the parallel library; the parallel tests also pass, but
  with lots of non-fatal error messages.

* For LLNL, uP: both serial and parallel tests pass.
  Zeus: Serial tests pass but parallel tests fail with a known problem in MPI.
  ubgl: Serial tests pass but parallel tests fail.

* On SUN 5.10 C++, testing fails in the "Testing Shared Datatypes with
  Attributes" test.

* Configuring with --enable-debug=all produces compiler errors on most
  platforms: Users who want to run HDF5 in debug mode should use
  --enable-debug rather than --enable-debug=all to enable debugging
  information on most modules.

* On Mac OS 10.4, test/dt_arith.c has some errors in conversion from long
  double to (unsigned) long long and from (unsigned) long long to long double.

* On Altix SGI with Intel 9.0, testmeta.c would not compile with -O3
  optimization flag.

* On VAX, the Scaleoffset filter is not supported. The filter cannot be
  applied to HDF5 data generated on VAX. The Scaleoffset filter only supports
  the IEEE standard for floating-point data.

* On Cray X1, a lone colon on the command line of h5dump --xml (as in
  the testh5dumpxml.sh script) is misinterpereted by the operating system
  and causes an error.

* On mpich 1.2.5 and 1.2.6, if more than two processes contribute no IO and
  the application asks to do collective IO, we have found that when using 4
  processors, a simple collective write will sometimes be hung. This can be
  verified with t_mpi test under testpar.

* On IRIX6.5, when the C compiler version is greater than 7.4, complicated
  MPI derived datatype code will work. However, the user should increase
  the value of the MPI_TYPE_MAX environment variable to some appropriate value
  to use collective irregular selection code. For example, the current
  parallel HDF5 test needs to raise MPI_TYPE_MAX to 200,000 to pass the test.

* A dataset created or rewritten with a v1.6.3 library or after cannot be read

with the v1.6.2 library or before when the Fletcher32 EDC filter is enabled.
There was a bug in the calculating code of the Fletcher32 checksum in the
library before v1.6.3; the checksum value was not consistent between big-
endian and little-endian systems. This bug was fixed in Release 1.6.3.
However, after fixing the bug, the checksum value was no longer the same as
before on little-endian system. Library releases after 1.6.4 can still read
datasets created or rewritten with an HDF5 library of v1.6.2 or before.
SLU - 2005/6/30

* For version 6 (6.02 and 6.04) of the Portland Group compiler on the AMD
  Opteron processor, there is a bug in the compiler for optimization(-O2).
  The library failed in several tests, all related to the MULTI driver.
  The problem has been reported to the vendor.

* On IBM AIX systems, parallel HDF5 mode will fail some tests with error
  messages like "INFO: 0031-XXX ...". This is from the command `poe'.
  Set the environment variable MP_INFOLEVEL to 0 to minimize the messages
  and run the tests again.

  The tests may fail with messages like "The socket name is already in use",
  but HDF5 does not use sockets. This failure is due to problems with the
  poe command trying to set up the debug socket. To resolve this problem,
  check to see whether there are many old /tmp/s.pedb.* files staying around.
  These are sockets used by the poe command and left behind due to failed
  commands. First, ask your system administrator to clean them out.
  Lastly, request IBM to provide a means to run poe without the debug socket.

* 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 do not use this option on these platforms
  during configuration.

* With the gcc 2.95.2 compiler, HDF5 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 ./dsets tests fail on 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.

* Not all platforms behave correctly with Szip's shared libraries. Szip is
  disabled in these cases, and a message is relayed at configure time. Static
  libraries should be working on all systems that support Szip and should be
  used when shared libraries are unavailable.

  There is also a configure error on Altix machines that incorrectly reports
  when a version of Szip without an encoder is being used.

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

* Information about building with PGI and Intel compilers is available in
  the INSTALL file sections 4.7 and 4.8.

* On at least one system, SDSC DataStar, the scheduler (in this case
  LoadLeveler) sends job status updates to standard error when you run
  any executable that was compiled with the parallel compilers.

  This causes problems when running "make check" on parallel builds, as
  many of the tool tests function by saving the output from test runs,
  and comparing it to an exemplar.

  The best solution is to reconfigure the target system so it no longer
  inserts the extra text. However, this may not be practical.

  In such cases, one solution is to "setenv HDF5_Make_Ignore yes" prior to
  the configure and build. This will cause "make check" to continue after
  detecting errors in the tool tests. However, in the case of SDSC DataStar,
  it also leaves you with some 150 "failed" tests to examine by hand.

  A second solution is to write a script to run serial tests and filter
  out the text added by the scheduler. A sample script used on SDSC
  DataStar is given below, but you will probably have to customize it
  for your installation.

  Observe that the basic idea is to insert the script as the first item
  on the command line which executes the the test. The script then
  executes the test and filters out the offending text before passing
  it on.

        #!/bin/csh

        set STDOUT_FILE=~/bin/serial_filter.stdout
        set STDERR_FILE=~/bin/serial_filter.stderr

        rm -f $STDOUT_FILE $STDERR_FILE

        ($* > $STDOUT_FILE) >& $STDERR_FILE

        set RETURN_VALUE=$status

        cat $STDOUT_FILE

        tail +3 $STDERR_FILE

        exit $RETURN_VALUE

  You get the HDF5 make files and test scipts to execute your filter script
  by setting the environment variable "RUNSERIAL" to the full path of the
  script prior to running configure for parallel builds. Remember to
  "unsetenv RUNSERIAL" before running configure for a serial build.

  Note that the RUNSERIAL environment variable exists so that we can
  can prefix serial runs as necessary on the target system. On DataStar,

no prefix is necessary. However on an MPICH system, the prefix might
have to be set to something like "/usr/local/mpi/bin/mpirun -np 1" to
get the serial tests to run at all.

In such cases, you will have to include the regular prefix in your
filter script.

* H5Ocopy() does not copy reg_ref attributes correctly when shared-message
  is turn on. The value of the reference in the destination attriubte is

wrong. This H5Ocopy problem will affect the h5copy tool.