HDF5 1.8.15

Version	HDF5 1.8.15
Release Date	2015-05-04
Download	/ftp/HDF5/releases/hdf5-1.8/hdf5-1.8.15/
Release Notes	Release Notes

Release Notes:

```
HDF5 version 1.8.15 released on 2015-05-04
================================================================================

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

This document describes the differences between HDF5-1.8.14 and
HDF5-1.8.15, and contains information on the platforms tested and
known problems in HDF5-1.8.15.

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

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

The HDF5 release can be obtained from:

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

User documentation for HDF5 can be accessed directly at this location:

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

All new and modified APIs are listed in detail in the "HDF5 Software Changes
from Release to Release" document at this location:

     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
- Support for New Platforms, Languages, and Compilers
- Bug Fixes since HDF5-1.8.14
- Supported Platforms
- Supported Configuration Features Summary
- More Tested Platforms
- Known Problems


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

    Configuration
    -------------
    - CMake

      Improvements made to the CMake build system.

      The default options were changed to align with the Autotools configure
      defaults. CMake configure files now support components when packaged
      with CPack. Windows CPack supports WiX packaging, and will look for
      WiX and NSIS in the standard locations.

      The CMake minimum has been changed to 3.1.

      (ADB - 2015/04/01 HDFFV-8074, 8968, 9006)

    - cmakehdf5 for Cmake building.
      Added configure options to support the building of Fortran or CXX API,
      to enable/disable testings. Use "cmakehdf5 --help" for details.
      (AKC - 2014/12/09 HDFFV-8932)

    - Building Shared and Parallel Made Explicit

      When --enable-parallel is specified, configure used to disable
      shared by default.

      Removed the restriction for building shared when parallel is
      enabled. --disable-shared has to be used explicitly if user
      wishes to disable shared libraries.

      (MSC - 2015/02/19 HDFFV-9069)

    - Inferring Parallel Compilers

      configure was trying to infer if a compiler is a parallel
      compiler with MPI support and enable parallel even if the user
      did not explicitly enable parallel. This should not happen.

      Disabled inferring parallel compilers to enable parallel HDF5
      build. --enable-parallel has to be used explicitly to build
      parallel HDF5 regardless of the compiler type being used.

      (MSC - 2015/02/19 HDFFV-9068)

    - Large File Support Configuration Option

      Removed the option to enable or disable large file support. It will
      always be enabled.

      (MSC - 2015/02/19 HDFFV-9097)

    - Removed Configuration Feature

      When configure detected that the CodeWarrior compiler was being used it
```

would define a symbol that caused a test in test/tfile.c to be skipped due to a broken CodeWarrior open() command.

Since this only masks the problem instead of fixing it and we don't support CodeWarrior anyway, this functionality was removed.

(DER - 2015/02/21, HDFFV-9080)

- VMS Build/Test Files Have Been Removed

  HDF5 no longer supports VMS, and the files were getting out of date. Since we have no access to a VMS machine, there is no way for us to maintain them.

  A Subversion tag was created at:

      https://svn.hdfgroup.uiuc.edu/tags/vms_last_support_1_8

  immediately before removing the files.

  (DER - 2015-02-26, HDFFV-9147)

- Removal of --with-default-vfd configure Option

  In theory, this option was intended to allow setting a default VFD that would be used by the library. In practice, the feature only accepted the POSIX (SEC2) VFD (already the default) and the stdio VFD (a demo VFD not intended for production use). The inability to pass key VFD parameters at configure time limits the full implementation of this feature, so it was retired.

  (DER - 2015-02-26, HDFFV-9081)

- Direct VFD configure Behavior

  The configure options for Linux now allow the Direct VFD to build without passing additional compiler options/defines like _GNU_SOURCE. Passing --enable-direct-vfd is now all that is needed to enable the feature.

  The Direct VFD is now disabled by default since it is intended for specialized audiences. It was previously enabled by default, but the configure script did not set correct POSIX levels, etc. making this a moot point.

  Note that the Direct VFD can only be configured on Linux when the O_DIRECT flag to open()/create() and posix_memalign() function are available. This is unchanged from previous behavior.

  (DER - 2015-02-26, HDFFV-9057, 7567, 9088, 7566)

- _POSIX_C_SOURCE, _GNU_SOURCE, and _BSD_SOURCE No Longer Exported to h5cc and Other Compiler Wrappers

  The _POSIX_C_SOURCE, _GNU_SOURCE, and _BSD_SOURCE definitions are not required for using API functions and may conflict with user code requirements.

  (DER - 2015-03-08, HDFFV-9152)

```
- Removed the --enable-filters Option from configure

  This option allowed the user to disable selected internal filters,
  presumably to make the library smaller. It has been removed since
  it saved little space (the internal filters are small with respect
  to the overall library size) and was not generally extendible to
  the library at large due to the large number of #ifdefs that would
  be required.

  Note that this features applied to internal filters such as shuffle
  and n-bit and not external filters like gzip or Szip. Those are still
  enabled or disabled via their own configure options.

  (DER - 2015-03-08, HDFFV-9086)

- Removed Obsolete Time Functionality from configure and the C Library

  The library contained some residual functionality from obsolete
  time zone handling code. This has been removed, and the configure
  checks for the time functions have been cleaned up.

  * Lumped all the time functionality together in configure.ac.
    This was previously more spread out due to Solaris issues
    with the ordering of certain checks.

  * Removed processing that handles __tm_gmtoff members of struct
    tm. (libc-4)

  * Removed BSDgettimeofday(). (IRIX 5.3)

  * Removed timezone struct handling in gettimeofday() (considered
    harmful).

  Note that the HDF5 Library stores timestamps in a platform-independent
  manner, so old files can still be read. This only affects converting
  system time to HDF5 timestamps.

  The library currently uses the tm_gmtoff member of the tm struct
  (preferred, if available) or the timezone global variable to
  construct HDF5 timestamps.

  (DER - 2015-03-09, HDFFV-9083 and 9085)

- Added -D_DEFAULT_SOURCE to CPPFLAGS on Linux Systems

  This is the replacement for -D_BSD_SOURCE in versions of glibc since 2.19.
  Since both are defined, it should work for all versions of glibc. Defining
  both suppresses the warning about defining _BSD_SOURCE.

  (NAF - 2015-04-02, HDFFV-9079)

Library
-------
- Added Memory Allocation Functions that Use the Library's Allocator

  HDF5 filters may need to allocate or resize the buffer that is passed
  to them from the library. If the filter has been compiled separately
  from the library, it and the library may use different memory
```

  allocation libraries for the (re)allocation and free calls. This can cause heap corruption and crashes. This is particularly a problem on Windows since each C run-time library is implemented as a separate shared library, but can also show up on POSIX systems when debug or high-performance allocation libraries are in use.

  Two new functions (H5allocate_memory() and H5resize_memory()) were added to the HDF5 C library. These functions have the same semantics as malloc/calloc and realloc, respectively. Their primary purpose is to allow filter authors to allocate or resize memory using the same memory allocation library as the HDF5 library. Filter authors are highly encouraged to use these new functions in place of malloc, calloc, and realloc. They should also use the H5free_memory() call when freeing memory.

  Note that the filters provided with the library (zlib, szip, etc.) do not experience the problems that these new functions are intended to fix. This work only applies to third-party filters that are compiled separately from the library.

  (DER - 2015-04-01, HDFFV-9100)

- H5Pset_istore_k and H5Pset_sym_k

  These two functions didn't check the value of the input parameter "ik". When 2*ik exceeded 2 bytes of storage, data was lost in the file; for example, some chunks would be overwritten.

  Added validation of "ik" to not exceed the max v1 btree entries (2 bytes) to these two routines.

  (VC - 2015-03-24, HDFFV-9173)

- Added Functions to Control the Value of H5PL_no_plugin_g without Using an Environment Variable

  Sometimes it is necessary for an application to disable the use of dynamically loaded plugin libraries without requiring the library to be built with plugin support disabled or to set an environment variable to disable plugin support globally.

  Two new functions (H5PLset_loading_state() and H5PLget_loading_state()) were added to the HDF5 C Library. These functions require a parameter that indicates which type of dynamically loaded plugin is enabled or disabled.

  (ADB - 2015-03-17, HDFFV-8520)

Parallel Library

- MPI_Finalize and HDF5 Library Shutdown

  Calling HDF5 routines after MPI_Finalize has been closed should not be done, since those routines might call MPI functions that would not be possible to do after finalizing the MPI library.

  Attached an attribute destroy callback to MPI_COMM_SELF that shuts down the HDF5 library when MPI_COMM_SELF is destroyed, in other words, on MPI_Finalize. This should fix several issues

```
      that users see when they forget to close HDF5 objects before
      calling MPI_Finalize().

      (MSC - 2015/02/25, HDFFV-883)

    Tools
    -----
    - None

    High-Level APIs
    ---------------
    - None

    Fortran API
    -----------
    - Added Global Variables

      These new global variables are equivalent to the C definitions
      without the '_F':

      H5G_UDLINK_F
      H5G_SAME_LOC_F
      H5O_TYPE_UNKNOWN_F
      H5O_TYPE_GROUP_F
      H5O_TYPE_DATASET_F
      H5O_NAMED_DATATYPE_F
      H5O_TYPE_NTYPES_F

      (MSB - 2015/02/03, HDFFV-9040)


    C++ API
    -------
    - New Wrappers for C Functions H5P[s/g]et_libver_bounds

      Wrappers were added to class H5::FileAccPropList for the
      C Functions H5Pget_libver_bounds and H5Pset_libver_bounds.

      (BMR, 2015/04/06, Part of HDFFV-9167)

    - New Wrappers to Get the Object Header's Version

      The following wrappers are added to class H5::CommonFG
      Returns the object header version of an object in a file or group,
      given the object's name.

      unsigned childObjVersion(const char* objname) const;
      unsigned childObjVersion(const H5std_string& objname) const;

      (BMR, 2015/04/06)

    - New DataType Constructor

      Added a DataType constructor that takes a PredType object, and this
      constructor will invoke H5Tcopy to generate another datatype id
      from a predefined datatype.

      (BMR, 2015/04/06)
```

```
Support for New Platforms, Languages, and Compilers
===================================================
    - Support for Linux 3.10.0-123.20.1.el7 added (LK - 2015/04/01)
    - Support for Mac OS X Yosemite 10.10 added (AKC - 2015/03/04, HDFFV-9007)
    - Support for AIX 6.1 added and AIX 5.3 is retired. (AKC - 2015/01/09)

Bug Fixes since HDF5-1.8.14
===========================

    Configuration
    -------------
    - Make uninstall generated "test: argument expected".
      The error is due to $EXAMPLETOPDIR is used without setting a value first.

      Fixed by assign it with the proper value.

      (AKC - 2015/04/29, HDFFV-9298)

    - Windows Installer Incorrect Display of PATH Environment Variable

      In the Windows installer, the dialog box where the user can elect to
      add the product's bin path to the %PATH% environment variable displayed
      an incorrect path. This path was missing the C:\Program Files part
      and used the POSIX file separator '/' before the bin (<path>/bin,
      instead of <path>\bin).

      The dialog box text was changed to simply say that the product's bin
      path would be added instead of explicitly displaying the path.
      This is in line with most installers. The reason for not fixing the
      displayed path instead is that it is difficult to pass the correct
      path from CPack to the NSIS installer for display.

      Note that this was never a code issue - it was just a display
      problem. The installer always did the right thing when updating the
      environment variable.

      (DER - 2014/11/14, HDFFV-9016)

    Library
    -------
    - Incorrect Usage of List in CMake COMPILE_DEFINITIONS set_property

      The CMake command set_property with COMPILE_DEFINITIONS property
      needs a quoted semi-colon separated list of values. CMake will
      transform the list to a series of -D{value} for the compile.

      (ADB - 2014/12/09, HDFV-9041)

    - Fixed Compile Errors on Windows w/ Visual Studio and CMake When
      UNICODE is Defined

      The HDF5 Library could not be built on Windows with Visual Studio when
      UNICODE was defined. This was due to the incorrect use of the TEXT()
      macro and some Win32 API functions that take TCHAR parameters. The faulty
      code was a part of the filter plugin functionality. This was a
      compile-time error that only affected users who build HDF5 from source
      and define UNICODE, usually when HDF5 is being built as a part of a
      larger product. There were no run-time effects.
```

These errors caused no problems when UNICODE was not defined. HDF5 is normally not built with UNICODE defined and the binaries were unaffected.

The fix was to remove the TEXT() macro and explicitly use the 'A' form of the Win32 API calls, which expect char strings instead of wchar_t strings.

Note that HDF5 currently does not support Unicode file paths on Windows.

(DER - 2015/02/22, HDFFV-8927)

- Addition of Error Tracing Functionality to Several C API Calls

  A bug in a text processing script caused API calls that return a pointer to not receive error tracing macros/functionality.

  The bug has been corrected and error tracing functionality has been added to the affected API calls. These functions will now correctly print trace information when library errors are encountered.

  (DER - 2015/02/26, HDFFV-9141)

- H5Rdereference Now Checks for HADDR_UNDEF or Uninitialized References

  When passed HADDR_UNDEF or uninitialized references, the previous behavior of H5Rdereference was to continue to process the reference as a valid address.

  H5Rdereference was changed to return immediately (with an error message) if the references are HADDR_UNDEF or uninitialized.

  (MSB - 2015/3/10, HDFFV-7959)

- Fixed Bugs in H5Sextent_copy

  H5Sextent_copy would not free the previous extent, resulting in a memory leak. Also, H5Sextent_copy would not update the number of elements selected if the selection was "all", causing various problems. These issues have been fixed.

  (NAF - 2015/04/02)

Parallel Library

- Fixed a Potential Memory Error

  Fixed a potential memory error when performing parallel I/O on a dataset with a single chunk, and at least one process has nothing to do.

  (NAF - 2015/02/16)

- Parallel Test Problem Fixed

  Fixed problem with parallel tests where they failed beyond a certain number of ranks. All tests should work for any arbitrary number of ranks.

(MSC - 2014/11/06, HDFFV-1027,8962,8963)

- MPE Support

  Enabling MPE was causing HDF5 build to fail. Support for it was dropped at some point in time.

  Fixed problem with enabling MPE. Users should use the community maintained MPE on github (http://git.mpich.org/mpe.git/).

  (MSC - 2015/02/20, HDFFV-9135)

Performance

- None

Tools

- h5repack crashed on enumerated 8-bit type.

  Previous version 1.8.14 introduced an error that caused the reading of enumerated 8-bit type nested in compound type to fail.

  Fixed library code responsible for reading the particular type.
  (AKC - 2015.03/31, HDFFV-8667)

- h52gif crashed non-8bit images.

  h52gif crashed if instructed to convert images other than 8bit images.

  h52gif could handle only 8bit images. Added code to detect non-8bit images and flag them as failure. Update tool document page to reflect the limit.
  (AKC - 2015/03/31, HDFFV-8957)

- perform/benchpar.c retired.

  benchpar.c has not been built for a long time and its original purpose is not needed any more.
  (AKC - 2014/12/19, HDFFV-8156)

- Source perform/ directory moved to tools/perform.
  The perform directory is moved to tools/perform for easier maintenance.
  (AKC - 2014/12/17, HDFFV-9046)

Fortran API

- Fortran Fails with --enable-fortran2003 and Intel 15.x Compilers

  Added BIND(C) to the offending APIs.

  The Fortran Library (--enable-fortran2003) now works using Intel 15.x without the need for any additional compilers flags.

  (MSB - 2015/1/26, HDFFV-9049)

- h5tenum_insert_f Does Not Work with Default 8 Byte Integers

```
      (xlf compiler)

      In the Fortran 90 API, 'value' is no longer cast into the C int type.
      Therefore, if h5tenum_insert_f is passed an 8 byte integer (via -i8)
      then 'value' is written as the same type as the default Fortran
      integer type (which can be 8 bytes).

      A new Fortran 2003 API was added which is more in line with the C
      API and users are strongly encouraged to use the Fortran 2003 API
      instead of the Fortran 90 API.

      SUBROUTINE h5tenum_insert_f(type_id, name, value, hdferr)
      INTEGER(HID_T)  , INTENT(IN) :: type_id
      CHARACTER(LEN=*), INTENT(IN) :: name
      TYPE(C_PTR)     , INTENT(IN) :: value
      INTEGER, INTENT(OUT) :: hdferr

      (MSB - 2015/2/19, HDFFV-8908)

    - Some Fortran APIs Never Returned the Error State

      Some Fortran APIs never returned the error state: they
      would always return a positive number. The APIs include
      the following:

      h5fget_file_image_f
      h5lget_name_by_idx_f
      h5oget_comment_by_name_f

      They were corrected to return a negative number as described in
      the Reference Manual if an error occurred.

      (MSB - 2015/3/19, HDF5-239)

    - Fixed h5pget_class_f

      h5pget_class_f never correlated the class identifier to the property
      list class name as indicated in the HDF5 Reference Manual; it instead
      returned a property list class identifier as an INTEGER. The INTEGER
      needed to be of type INTEGER(HID_T) to be correct.

      The h5pget_class_f API was changed to return an INTEGER(HID_T)
      property list class identifier instead of an INTEGER. This mimics the
      intended behavior of the C API.

      (MSB - 2015/3/16, HDFFV5-9162)

    C++ API
    ------
    - Combined Two H5File::getObjCount Overloaded Methods

      The following two methods

      ssize_t getObjCount(unsigned types) const;
      ssize_t getObjCount() const;

      were combined into one:

      ssize_t getObjCount(unsigned types = H5F_OBJ_ALL) const;
```

(BMR - 2015/04/06)

- Many Warnings Were Removed

  Many warnings such as conversion, unused variables, missing base class initialization, and initializing base classes in wrong order were removed.

  (BMR, 2015/04/06)

- Functionality Deprecation

  The following two constructors of classes AbstractDs, IdComponent, H5Location, and H5Object are no longer appropriate after the data member "id" had been moved from IdComponent to the sub-classes in previous releases.

```
<Classname>(const hid_t h5_id);
<Classname>(const <Classname>& original);
```

  The copy constructors were no-op and removed in 1.8.15. The other constructors will be removed from 1.10 release, and then from 1.8.17 if their removal does not cause any problems.

  (BMR, 2015-04-06)

High-Level APIs:

- Suppress Warnings from Flex/Bison-generated Code

  Warning suppression #pragmas, etc. have been added to H5LTparse.c and H5LTanalyze.c. We have no control over this code since it's created by a generator.

  (DER - 2015/03/08 - HDFFV-9149)

- Changed hdf5_hl.h to Include the HDF5 Main Library "hdf5.h"

  User's no longer need to include both hdf5_hl.h and hdf5.h

  (MSB - 2015/2/14, HDFFV-8685)

- H5PTcreate_fl Does Not Convert to Memory Datatype

  H5PTcreate_fl now converts to the table's native memory datatype to fix the problem of handling BE and LE packet tables.

  (MSB - 2015/2/26 - HDFFV-9042)

- Fix for H5LT Attribute Functions

  H5LT attribute functions fail to create attributes whose name is a substring of an existing attribute.

  H5LT attribute functions can now create attributes whose name is a substring of an existing attribute.

(MSB - 2015/2/24, HDFFV-9132)

Fortran High-Level APIs:

- Internal Library Fix for Missing Argument Declaration

  In Interface block for h5tbmake_table_c, "max_char_size_field_names" is listed as an input, but in the argument definitions it is "INTEGER :: max_char_size". This caused no known problems with the Fortran HL API.

  Fixed missing argument definition.

  (MSB - 2015/2/18, HDFFV-8559)

Testing

- None

Supported Platforms

The following platforms are supported and have been tested for this release. They are built with the configure process unless specified otherwise.

AIX 6.1 (NASA G-ADA)	xlc/xlc_r 10.1.0.5 xlC/xlC_r 10.1.0.5 xlf90/xlf90_r 12.1.0.6
Linux 2.6.18-308.13.1.el5PAE #1 SMP i686 i686 i386 (jam)	GNU C (gcc), Fortran (gfortran), C++ (g++) compilers for 32-bit applications; Version 4.1.2 20080704 (Red Hat 4.1.2-55) Version 4.8.4, 4.9.2 PGI C, Fortran, C++ Compilers for 32-bit applications; Version 14.10-0 Intel(R) C, C++, Fortran Compiler for 32-bit applications; Version 15.0.1.133 (Build 20141023)
Linux 2.6.18-371.6.1.el5 #1 SMP x86_64 GNU/Linux (koala)	GNU C (gcc), Fortran (gfortran), C++ (g++) compilers for 64-bit applications; Version 4.1.2 20080704 (Red Hat 4.1.2-55) Version 4.8.4, 4.9.2 Intel(R) C, C++, Fortran Compilers for applications running on Intel(R) 64; Version 15.0.1.133 Build 20141023
Linux 2.6.32-431.11.2.el6 #1 SMP x86_64 GNU/Linux (platypus)	GNU C (gcc), Fortran (gfortran), C++ (g++) compilers: Version 4.4.7 20120313 Version 4.8.2, Version 4.9.2 PGI C, Fortran, C++ for 64-bit target on x86-64; Version 14.10-0

```
                                  Intel(R) C (icc), C++ (icpc), Fortran (icc)
                                  compilers:
                                     Version 15.0.1.133 Build 20141023

    Linux 3.10.0-123.20.1.el7     GNU C (gcc), Fortran (gfortran), C++ (g++)
    #1 SMP x86_64 GNU/Linux       compilers:
    (moohan)                         Version 4.8.2 20140120 (Red Hat 4.8.2-16)
                                  Intel(R) C Intel(R) 64 Compiler XE for
                                  applications running on Intel(R) 64,
                                     Version 15.0.1.133 Build 20141023

    Linux 2.6.32-431.29.2.el6.ppc64 gcc (GCC) 4.4.7 20120313 (Red Hat 4.4.7-4)
    #1 SMP ppc64 GNU/Linux        g++ (GCC) 4.4.7 20120313 (Red Hat 4.4.7-4)
    (ostrich)                     GNU Fortran (GCC) 4.4.7 20120313 (Red Hat 4.4.7-4)
                                  IBM XL C/C++ V13.1
                                  IBM XL Fortran V15.1

    SunOS 5.11 32- and 64-bit     Sun C 5.12 SunOS_sparc
    (emu)                         Sun Fortran 95 8.6 SunOS_sparc
                                  Sun C++ 5.12 SunOS_sparc

    Windows 7                     Visual Studio 2008  (cmake)
                                  Visual Studio 2010 w/ Intel Fortran 14 (cmake)
                                  Visual Studio 2012 w/ Intel Fortran 15 (cmake)
                                  Visual Studio 2013 w/ Intel Fortran 15 (cmake)
                                  Cygwin(CYGWIN_NT-6.1 1.7.34(0.285/5/3) gcc(4.9.2)
compiler and gfortran)
                                  (cmake and autotools)

    Windows 7 x64                 Visual Studio 2008  (cmake)
                                  Visual Studio 2010 w/ Intel Fortran 14 (cmake)
                                  Visual Studio 2012 w/ Intel Fortran 15 (cmake)
                                  Visual Studio 2013 w/ Intel Fortran 15 (cmake)

    Windows 8.1                   Visual Studio 2012 w/ Intel Fortran 15 (cmake)
                                  Visual Studio 2013 w/ Intel Fortran 15 (cmake)

    Windows 8.1 x64               Visual Studio 2012 w/ Intel Fortran 15 (cmake)
                                  Visual Studio 2013 w/ Intel Fortran 15 (cmake)

    Mac OS X Mt. Lion 10.8.5      Apple clang/clang++ version 5.1 from Xcode 5.1
    64-bit         gfortran GNU Fortran (GCC) 4.8.2
    (swallow/kite)                Intel icc/icpc/ifort version 14.0.2

    Mac OS X Mavericks 10.9.5     Apple clang/clang++ version 6.0 from Xcode 6.1.1
    64-bit         gfortran GNU Fortran (GCC) 4.8.2
    (wren/quail)                  Intel icc/icpc/ifort version 14.0.2

    Mac OS X Yosemite 10.10.2     Apple clang/clang++ version 6.0 from Xcode 6.1.1
    64-bit         gfortran GNU Fortran (GCC) 4.9.2
    (osx1010dev/osx1010test)      Intel icc/icpc/ifort version 15.0.1
```

Tested Configuration Features Summary
=====================================

```
    In the tables below
          y   = tested
          n   = not tested in this release
```

C   = Cluster
W   = Workstation
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 parallel	F90/ F2003	F90 parallel	C++	zlib	SZIP
SunOS 5.11 32-bit	n	y/y	n	y	y	y
SunOS 5.11 64-bit	n	y/y	n	y	y	y
Windows 7	y	y/y	n	y	y	y
Windows 7 x64	y	y/y	n	y	y	y
Windows 7 Cygwin	n	y/y	n	y	y	n
Windows 8.1	n	y/y	n	y	y	y
Windows 8.1 x64	n	y/y	n	y	y	y
Mac OS X Mountain Lion 10.8.5 64-bit	n	y/y	n	y	y	y
Mac OS X Mavericks 10.9.5 64-bit	n	y/y	n	y	y	y
Mac OS X Yosemeti 10.10.2 64-bit	n	y/y	n	y	y	y
AIX 6.1 32- and 64-bit	n	y/n	n	y	y	y
CentOS 5.9 Linux 2.6.18-308 i686 GNU	y	y/y	y	y	y	y
CentOS 5.9 Linux 2.6.18-308 i686 Intel	n	y/y	n	y	y	y
CentOS 5.9 Linux 2.6.18-308 i686 PGI	n	y/y	n	y	y	y
CentOS 5.9 Linux 2.6.18 x86_64 GNU	n	y/y	n	y	y	y
CentOS 5.9 Linux 2.6.18 x86_64 Intel	n	y/y	n	y	y	y
CentOS 6.4 Linux 2.6.32 x86_64 GNU	y	y/y	y	y	y	y
CentOS 6.4 Linux 2.6.32 x86_64 Intel	n	y/y	n	y	y	y
CentOS 6.4 Linux 2.6.32 x86_64 PGI	n	y/y	n	y	y	y
CentOS 7.0 Linux 3.10.0 x86_64 GNU	y	y/y	y	y	y	y
CentOS 7.0 Linux 3.10.0 x86_64 Intel	n	y/y	n	y	y	y
Linux 2.6.32-431.11.2.el6.ppc64	n	y/n	n	y	y	y

Platform	Shared C libs	Shared F90 libs	Shared C++ libs	Thread- safe
SunOS 5.11 32-bit	y	y	y	y
SunOS 5.11 64-bit	y	y	y	y
Windows 7	y	y	y	y
Windows 7 x64	y	y	y	y
Windows 7 Cygwin	n	n	n	y
Windows 8.1	y	y	y	y
Windows 8.1 x64	y	y	y	y
Mac OS X Mountain Lion 10.8.5 64-bit	y	n	y	y
Mac OS X Mavericks 10.9.5 64-bit	y	n	y	y
Mac OS X Yosemeti 10.10.2 64-bit	y	n	y	y
AIX 6.1 32- and 64-bit	y	n	n	y
CentOS 5.9 Linux 2.6.18-308 i686 GNU	y	y	y	y
CentOS 5.9 Linux 2.6.18-308 i686 Intel	y	y	y	n
CentOS 5.9 Linux 2.6.18-308 i686 PGI	y	y	y	n
CentOS 5.9 Linux 2.6.18 x86_64 GNU	y	y	y	y
CentOS 5.9 Linux 2.6.18 x86_64 Intel	y	y	y	n
CentOS 6.4 Linux 2.6.32 x86_64 GNU	y	y	y	n
CentOS 6.4 Linux 2.6.32 x86_64 Intel	y	y	y	n
CentOS 6.4 Linux 2.6.32 x86_64 PGI	y	y	y	n
CentOS 7.0 Linux 3.10.0 x86_64 GNU	y	y	y	n
CentOS 7.0 Linux 3.10.0 x86_64 Intel	y	y	y	n
Linux 2.6.32-431.11.2.el6.ppc64	y	y	y	n

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

More Tested Platforms

The following platforms are not supported but have been tested for this release.

```
    Linux 2.6.18-308.13.1.el5PAE  MPICH mpich 3.1.3 compiled with
    #1 SMP i686 i686 i386             gcc 4.9.2 and gfortran 4.9.2
    (jam)                         g95 (GCC 4.0.3 (g95 0.94!)

    Linux 2.6.18-431.11.2.el6     MPICH mpich 3.1.3 compiled with
    #1 SMP x86_64 GNU/Linux           gcc 4.9.2 and gfortran 4.9.2
    (platypus)                    g95 (GCC 4.0.3 (g95 0.94!)

    FreeBSD 8.2-STABLE i386       gcc 4.5.4 [FreeBSD] 20110526
    (loyalty)                     gcc 4.6.1 20110527
                                  g++ 4.6.1 20110527
                                  gfortran 4.6.1 20110527

    FreeBSD 8.2-STABLE amd64      gcc 4.5.4 [FreeBSD] 20110526
    (freedom)                     gcc 4.6.1 20110527
                                  g++ 4.6.1 20110527
                                  gfortran 4.6.1 20110527

    Debian7.5.0 3.2.0-4-amd64 #1 SMP Debian 3.2.51-1 x86_64 GNU/Linux
                                  gcc (Debian 4.7.2-5) 4.7.2
                                  GNU Fortran (Debian 4.7.2-5) 4.7.2
                                  (cmake and autotools)

    Fedora20 3.15.3-200.fc20.x86_64 #1 SMP x86_64 x86_64 x86_64 GNU/Linux
                                  gcc (GCC) 4.8.3 20140624 (Red Hat 4.8.3-1)
                                  GNU Fortran (GCC) 4.8.3 20140624 (Red Hat 4.8.3-1)
                                  (cmake and autotools)

    SUSE 13.1 3.11.10-17-desktop #1 SMP PREEMPT x86_64 x86_64 x86_64 GNU/Linux
                                  gcc (SUSE Linux) 4.8.1
                                  GNU Fortran (SUSE Linux) 4.8.1
                                  (cmake and autotools)

    Ubuntu 14.04 3.13.0-35-generic #62-Ubuntu SMP x86_64 GNU/Linux
                                  gcc (Ubuntu/Linaro 4.9.1-0ubuntu1) 4.9.1
                                  GNU Fortran (Ubuntu/Linaro 4.9.1-0ubuntu1) 4.9.1
                                  (cmake and autotools)

    hopper.nersc.gov              PrgEnv-gnu/5.2.40
                                  gcc (GCC) 4.9.2 20141030 (Cray Inc.)
                                  GNU Fortran (GCC) 4.9.2 20141030 (Cray Inc.)
                                  g++ (GCC) 4.9.2 20141030 (Cray Inc.)
```

Known Problems

* On Windows platforms in debug configurations, the VFD flush1 tests will fail with the split and multi VFD drivers. These tests will display a modal debug dialog which must be answered or wait for the test timeout to expire.
  (ADB - 2014/06/23 - HDFFV-8851)

* CLANG compiler with the options -fcatch-undefined-behavior and -ftrapv catches some undefined behavior in the alignment algorithm of the macro DETECT_I

in H5detect.c (Issue 8147). Since the algorithm is trying to detect the alignment of integers, ideally the flag -fcatch-undefined-behavior shouldn't to be used for H5detect.c. In the future, we can separate flags for H5detect.c from the rest of the library. (SLU - 2013/10/16)

* Make provided by Solaris fails in "make check". Solaris users should use gmake to build and install the HDF5 software. (AKC - 2013/10/08 - HDFFV-8534)

* The C++ and FORTRAN bindings are not currently working on FreeBSD with the native release 8.2 compilers (4.2.1), but are working with gcc 4.6 from the ports (and probably gcc releases after that).
  (QAK - 2012/10/19)

* The following h5dump test case fails in BG/P machines (and potentially other machines that use a command script to launch executables):

  ```
  h5dump --no-compact-subset -d "AHFINDERDIRECT::ah_centroid_t[0] it=0 tl=0"
    tno-subset.h5
  ```

  This is due to the embedded spaces in the dataset name being interpreted by the command script launcher as meta-characters, thus passing three arguments to h5dump's -d flag. The command passes if run by hand, just not via the test script.
  (AKC - 2012/05/03)

* The STDIO VFD does not work on some architectures, possibly due to 32/64 bit or large file issues. The basic STDIO VFD test is known to fail on 64-bit SunOS 5.10 on SPARC when built with -m64 and 32-bit OS X/Darwin 10.7.0. The STDIO VFD test has been disabled while we investigate and a fix should appear in a future release.
  (DER - 2011/10/14 - HDFFV-8235)

* h5diff can report inconsistent results when comparing datasets of enum type that contain invalid values. This is due to how enum types are handled in the library and will be addressed in a future release.
  (DER - 2011/10/14 - HDFFV-7527)

* The links test can fail under the stdio VFD due to some issues with external links. This will be investigated and fixed in a future release.
  (DER - 2011/10/14 - HDFFV-7768)

* After the shared library support was fixed for some bugs, it was discovered that "make prefix=XXX install" no longer works for shared libraries. It still works correctly for static libraries. Therefore, if you want to install the HDF5 shared libraries in a location such as /usr/local/hdf5, you need to specify the location via the --prefix option during configure time. E.g, ./configure --prefix=/usr/local/hdf5 ...
  (AKC - 2011/05/07 - HDFFV-7583)

* The parallel test, t_shapesame, in testpar/, may run for a long time and may be terminated by the alarm signal. If that happens, one can increase the alarm seconds (default is 1200 seconds = 20 minutes) by setting the environment variable, $HDF5_ALARM_SECONDS, to a larger value such as 3600 (60 minutes). Note that the t_shapesame test may fail in some systems (see the "While working on the 1.8.6 release..." problem below). If it does, it will waste more time if $HDF5_ALARM_SECONDS is set to a larger value.
  (AKC - 2011/05/07)

- Shared Fortran libraries are not quite working on AIX. While they are generated when --enable-shared is specified, the fortran and hl/fortran tests fail. the issue. HL and C++ shared libraries should now be working as intended, however.
  (MAM - 2011/04/20)

- While working on the 1.8.6 release of HDF5, a bug was discovered that can occur when reading from a dataset in parallel shortly after it has been written to collectively. The issue was exposed by a new test in the parallel HDF5 test suite, but had existed before that. We believe the problem lies with certain MPI implementations and/or file systems.

  We have provided a pure MPI test program, as well as a standalone HDF5 program, that can be used to determine if this is an issue on your system. They should be run across multiple nodes with a varying number of processes. These programs can be found at:
  http://www.hdfgroup.org/ftp/HDF5/examples/known_problems/
  (NAF - 2011/01/19)

- All the VFL drivers aren't backward compatible. In H5FDpublic.h, the structure H5FD_class_t changed in 1.8. There is new parameter added to get_eoa and set_eoa callback functions. A new callback function get_type_map was added in. The public function H5FDrealloc was taken out in 1.8. The problem only happens when users define their own driver for 1.6 and try to plug in 1.8 library. Because there's only one user complaining about it, we (Elena, Quincey, and I) decided to leave it as it is (see bug report #1279). Quincey will make a plan for 1.10.
  (SLU - 2010/02/02)

- The --enable-static-exec configure flag will only statically link libraries if the static version of that library is present. If only the shared version of a library exists (i.e., most system libraries on Solaris, AIX, and Mac, for example, only have shared versions), the flag should still result in a successful compilation, but note that the installed executables will not be fully static. Thus, the only guarantee on these systems is that the executable is statically linked with just the HDF5 library.
  (MAM - 2009/11/04)

- 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 calculation 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 - 2005/06/30)
```

Compatibility

hdf5: 1.8.14 to 1.8.15 compatibility report

API compatibility report for the hdf5 library between 1.8.14 and 1.8.15 versior

Binary Compatibility | Source Compatibility

Test Info

Library Name	hdf5
Version #1	1.8.14
Version #2	1.8.15
CPU Type	x86
GCC Version	4.1.2
Subject	Binary Compatibility

Test Results

Total Header Files	75
Total Shared Libraries	6
Total Symbols / Types	1776 / 537
Verdict	**Incompatible (3.7%)**

Problem Summary

	Severity	Count
Added Symbols	-	15
Removed Symbols	High	9
Problems with Data Types	High	2
	Medium	0
	Low	0
	High	0

Problems with Symbols	Medium	0
	Low	2
Problems with Constants	Low	20
Other Changes in Symbols	-	5
Other Changes in Constants	-	5

Added Symbols (15)

H5Attribute.h, **libhdf5_cpp.so.10.0.0**
namespace **H5**
f_Attribute_setId (Attribute* *attr*, hid_t *new_id*)

H5CommonFG.h, **libhdf5_cpp.so.10.0.0**
namespace **H5**
CommonFG::childObjVersion (char const* *objname*) **const**

H5DataSet.h, **libhdf5_cpp.so.10.0.0**
namespace **H5**
f_DataSet_setId (DataSet* *dset*, hid_t *new_id*)

H5DataSpace.h, **libhdf5_cpp.so.10.0.0**
namespace **H5**
f_DataSpace_setId (DataSpace* *dspace*, hid_t *new_id*)

H5DataType.h, **libhdf5_cpp.so.10.0.0**
namespace **H5**
DataType::DataType [in-charge] (PredType const& *pred_type*)
DataType::DataType [not-in-charge] (PredType const& *pred_type*)
f_DataType_setId (DataType* *dtype*, hid_t *new_id*)

H5FaccProp.h, **libhdf5_cpp.so.10.0.0**
namespace **H5**
FileAccPropList::getLibverBounds (H5F_libver_t& *libver_low*, H5F_libver_t& *libver_high*) **c**

FileAccPropList::setLibverBounds (H5F_libver_t *libver_low*, H5F_libver_t *libver_high*) **cons**

H5File.h, **libhdf5_cpp.so.10.0.0**
namespace **H5**
H5File::H5File [in-charge] (hid_t *existing_id*)
H5File::H5File [not-in-charge] (hid_t *existing_id*)

H5PLpublic.h, **libhdf5.so.10.0.0**
H5PLget_loading_state (unsigned int* *plugin_type*)
H5PLset_loading_state (unsigned int *plugin_type*)

H5public.h, **libhdf5.so.10.0.0**
H5allocate_memory (size_t *size*, hbool_t *clear*)
H5resize_memory (void* *mem*, size_t *size*)

to the top

Removed Symbols (9)

H5AbstractDs.h, **libhdf5_cpp.so.9.0.0**
namespace **H5**
AbstractDs::AbstractDs [in-charge] (AbstractDs const& *original*)
AbstractDs::AbstractDs [not-in-charge] (AbstractDs const& *original*)

H5File.h, **libhdf5_cpp.so.9.0.0**
namespace **H5**
H5File::getObjCount ( ) **const**

H5IdComponent.h, **libhdf5_cpp.so.9.0.0**
namespace **H5**
IdComponent::IdComponent [in-charge] (IdComponent const& *original*)
IdComponent::IdComponent [not-in-charge] (IdComponent const& *original*)

H5Location.h, **libhdf5_cpp.so.9.0.0**
namespace **H5**
H5Location::H5Location [in-charge] (H5Location const& *original*)
H5Location::H5Location [not-in-charge] (H5Location const& *original*)

H5Object.h, **libhdf5_cpp.so.9.0.0**
namespace **H5**
H5Object::H5Object [in-charge] (H5Object const& *original*)
H5Object::H5Object [not-in-charge] (H5Object const& *original*)

to the top

Problems with Data Types, High Severity (2)

H5AbstractDs.h
namespace **H5**
[+] class **AbstractDs (1)**

H5CommonFG.h
namespace **H5**
[+] class **CommonFG (1)**

to the top

Problems with Symbols, Low Severity (2)

H5AbstractDs.h, **libhdf5_cpp.so.9.0.0**
namespace **H5**
[+] **AbstractDs::AbstractDs** [not-in-charge] (hid_t const *ds_id*) **(1)**
[+] **AbstractDs::AbstractDs** [in-charge] (hid_t const *ds_id*) **(1)**

to the top

Problems with Constants, Low Severity (20)

H5pubconf.h
[+] **H5_FP_TO_ULLONG_ACCURATE**
[+] **H5_FP_TO_ULLONG_RIGHT_MAXIMUM**
[+] **H5_GETTIMEOFDAY_GIVES_TZ**
[+] **H5_HAVE_FILTER_FLETCHER32**
[+] **H5_HAVE_FILTER_NBIT**
[+] **H5_HAVE_FILTER_SCALEOFFSET**
[+] **H5_HAVE_FILTER_SHUFFLE**
[+] **H5_HAVE_STRUCT_TIMEZONE**
[+] **H5_HAVE_STRUCT_TM_TM_ZONE**
[+] **H5_HAVE_TM_ZONE**
[+] **H5_INTEGER_TO_LDOUBLE_ACCURATE**
[+] **H5_LDOUBLE_TO_INTEGER_ACCURATE**
[+] **H5_LDOUBLE_TO_INTEGER_WORKS**
[+] **H5_LDOUBLE_TO_UINT_ACCURATE**
[+] **H5_LLONG_TO_FP_CAST_WORKS**
[+] **H5_ULLONG_TO_FP_CAST_WORKS**
[+] **H5_ULLONG_TO_LDOUBLE_PRECISION**
[+] **H5_ULONG_TO_FLOAT_ACCURATE**
[+] **H5_ULONG_TO_FP_BOTTOM_BIT_ACCURATE**
[+] **H5_VSNPRINTF_WORKS**

to the top

Other Changes in Symbols (5)

H5ArrayType.h, **libhdf5_cpp.so.9.0.0**
namespace **H5**
[+] **ArrayType::ArrayType** [in-charge] ( ) **(1)**
[+] **ArrayType::ArrayType** [not-in-charge] ( ) **(1)**

H5File.h, **libhdf5_cpp.so.9.0.0**
namespace **H5**
[+] **H5File::getObjCount** (unsigned int *types*) **const (1)**

H5VarLenType.h, **libhdf5_cpp.so.9.0.0**
namespace **H5**
[+] **VarLenType::VarLenType** [in-charge] ( ) **(1)**

[+] **VarLenType::VarLenType** [not-in-charge] ( ) **(1)**

to the top

Other Changes in Constants (5)

H5FDpublic.h

[+] **H5FD_FEAT_ALLOCATE_EARLY**
[+] **H5FD_FEAT_HAS_MPI**

H5PLpublic.h

[+] **H5PL_ALL_PLUGIN**
[+] **H5PL_FILTER_PLUGIN**

H5pubconf.h

[+] **H5_HAVE_TIMEZONE**

to the top

Header Files (75)

H5AbstractDs.h
H5ACpublic.h
H5api_adpt.h
H5Apublic.h
H5ArrayType.h
H5AtomType.h
H5Attribute.h
H5Classes.h
H5CommonFG.h
H5CompType.h
H5Cpp.h
H5CppDoc.h
H5Cpublic.h
H5DataSet.h
H5DataSpace.h
H5DataType.h
H5DcreatProp.h
H5DOpublic.h
H5Dpublic.h
H5DSpublic.h
H5DxferProp.h
H5EnumType.h
H5Epubgen.h
H5Epublic.h
H5Exception.h
H5f90i.h
H5f90i_gen.h
H5FaccProp.h

H5FcreatProp.h
H5FDcore.h
H5FDdirect.h
H5FDfamily.h
H5FDlog.h
H5FDmpi.h
H5FDmpio.h
H5FDmulti.h
H5FDpublic.h
H5FDsec2.h
H5FDstdio.h
H5File.h
H5FloatType.h
H5Fpublic.h
H5Gpublic.h
H5Group.h
H5IdComponent.h
H5IMpublic.h
H5Include.h
H5IntType.h
H5Ipublic.h
H5Library.h
H5Location.h
H5Lpublic.h
H5LTpublic.h
H5MMpublic.h
H5Object.h
H5Opublic.h
H5overflow.h
H5PacketTable.h
H5PLextern.h
H5Ppublic.h
H5PredType.h
H5PropList.h
H5PTpublic.h
H5pubconf.h
H5public.h
H5Rpublic.h
H5Spublic.h
H5StrType.h
H5TBpublic.h
H5Tpublic.h
H5VarLenType.h
H5version.h
H5Zpublic.h
hdf5.h
hdf5_hl.h

to the top

Shared Libraries (6)

libhdf5.so.9.0.0
libhdf5_cpp.so.9.0.0
libhdf5_fortran.so.9.0.0
libhdf5_hl.so.9.0.0
libhdf5_hl_cpp.so.9.0.0
libhdf5hl_fortran.so.9.0.0

to the top

Test Info

Library Name	hdf5
Version #1	1.8.14
Version #2	1.8.15
CPU Type	x86
GCC Version	4.1.2
Subject	Source Compatibility

Test Results

Total Header Files	75
Total Shared Libraries	6
Total Symbols / Types	1811 / 540
Verdict	**Incompatible (4.5%)**

Problem Summary

	Severity	Count
Added Symbols	-	17
Removed Symbols	High	9
Problems with Data Types	High	2
	Medium	0
	Low	0
Problems with Symbols	High	0
	Medium	0

	Low	0
Problems with Constants	Low	26
Other Changes in Symbols	-	7
Other Changes in Constants	-	13

Added Symbols (17)

H5Attribute.h

namespace **H5**

Attribute::getSpace ( ) **const**

f_Attribute_setId (Attribute* *attr*, hid_t *new_id*)

H5CommonFG.h

namespace **H5**

CommonFG::childObjVersion (char const* *objname*) **const**

CommonFG::childObjVersion (int const *H5std_string*) **const**

H5DataSet.h

namespace **H5**

f_DataSet_setId (DataSet* *dset*, hid_t *new_id*)

H5DataSpace.h

namespace **H5**

f_DataSpace_setId (DataSpace* *dspace*, hid_t *new_id*)

H5DataType.h

namespace **H5**

DataType::DataType [in-charge] (PredType const& *pred_type*)

DataType::DataType [not-in-charge] (PredType const& *pred_type*)

f_DataType_setId (DataType* *dtype*, hid_t *new_id*)

H5FaccProp.h

namespace **H5**

FileAccPropList::getLibverBounds (H5F_libver_t& *libver_low*, H5F_libver_t& *libver_high*) **c**

FileAccPropList::setLibverBounds (H5F_libver_t *libver_low*, H5F_libver_t *libver_high*) **cons**

H5File.h

namespace **H5**

H5File::H5File [in-charge] (hid_t *existing_id*)

H5File::H5File [not-in-charge] (hid_t *existing_id*)

H5PLpublic.h

H5PLget_loading_state (unsigned int* *plugin_type*)

H5PLset_loading_state (unsigned int *plugin_type*)

H5public.h
H5allocate_memory (size_t *size*, hbool_t *clear*)
H5resize_memory (void* *mem*, size_t *size*)

to the top

Removed Symbols (9)

H5AbstractDs.h
namespace **H5**
AbstractDs::AbstractDs [in-charge] (AbstractDs const& *original*)
AbstractDs::AbstractDs [not-in-charge] (AbstractDs const& *original*)

H5File.h
namespace **H5**
H5File::getObjCount ( ) **const**

H5IdComponent.h
namespace **H5**
IdComponent::IdComponent [in-charge] (IdComponent const& *original*)
IdComponent::IdComponent [not-in-charge] (IdComponent const& *original*)

H5Location.h
namespace **H5**
H5Location::H5Location [in-charge] (H5Location const& *original*)
H5Location::H5Location [not-in-charge] (H5Location const& *original*)

H5Object.h
namespace **H5**
H5Object::H5Object [in-charge] (H5Object const& *original*)
H5Object::H5Object [not-in-charge] (H5Object const& *original*)

to the top

Problems with Data Types, High Severity (2)

H5AbstractDs.h
namespace **H5**
[+] class **AbstractDs (1)**

H5CommonFG.h
namespace **H5**
[+] class **CommonFG (1)**

to the top

Problems with Constants, Low Severity (26)

H5pubconf.h

[+] **H5_DEFAULT_VFD**
[+] **H5_FP_TO_ULLONG_ACCURATE**
[+] **H5_FP_TO_ULLONG_RIGHT_MAXIMUM**
[+] **H5_GETTIMEOFDAY_GIVES_TZ**
[+] **H5_HAVE_FILTER_FLETCHER32**
[+] **H5_HAVE_FILTER_NBIT**
[+] **H5_HAVE_FILTER_SCALEOFFSET**
[+] **H5_HAVE_FILTER_SHUFFLE**
[+] **H5_HAVE_STRUCT_TIMEZONE**
[+] **H5_HAVE_STRUCT_TM_TM_ZONE**
[+] **H5_HAVE_TM_ZONE**
[+] **H5_INTEGER_TO_LDOUBLE_ACCURATE**
[+] **H5_LDOUBLE_TO_INTEGER_ACCURATE**
[+] **H5_LDOUBLE_TO_INTEGER_WORKS**
[+] **H5_LDOUBLE_TO_UINT_ACCURATE**
[+] **H5_LLONG_TO_FP_CAST_WORKS**
[+] **H5_PACKAGE_STRING**
[+] **H5_PACKAGE_VERSION**
[+] **H5_ULLONG_TO_FP_CAST_WORKS**
[+] **H5_ULLONG_TO_LDOUBLE_PRECISION**
[+] **H5_ULONG_TO_FLOAT_ACCURATE**
[+] **H5_ULONG_TO_FP_BOTTOM_BIT_ACCURATE**
[+] **H5_VERSION**
[+] **H5_VSNPRINTF_WORKS**

H5public.h

[+] **H5_VERS_INFO**
[+] **H5_VERS_RELEASE**

to the top

Other Changes in Symbols (7)

H5AbstractDs.h
namespace **H5**

[+] **AbstractDs::AbstractDs** [not-in-charge] (hid_t const *ds_id*) **(1)**
[+] **AbstractDs::AbstractDs** [in-charge] (hid_t const *ds_id*) **(1)**

H5ArrayType.h
namespace **H5**

[+] **ArrayType::ArrayType** [in-charge] ( ) **(1)**
[+] **ArrayType::ArrayType** [not-in-charge] ( ) **(1)**

H5File.h
namespace **H5**

[+] **H5File::getObjCount** (unsigned int *types*) **const (1)**

H5VarLenType.h
namespace **H5**
[+] **VarLenType::VarLenType** [in-charge] ( ) **(1)**
[+] **VarLenType::VarLenType** [not-in-charge] ( ) **(1)**

to the top

Other Changes in Constants (13)

H5api_adpt.h
[+] **H5_DLLCPPVAR**
[+] **H5_HLCPPDLLVAR**
[+] **H5_HLDLLVAR**
[+] **HDF5_HL_F90CSTUBDLLVAR**

H5f90i_gen.h
[+] **c_hsize_t_8**
[+] **c_size_t_4**

H5FDpublic.h
[+] **H5FD_FEAT_ALLOCATE_EARLY**
[+] **H5FD_FEAT_HAS_MPI**

H5Include.h
[+] **H5O_VERSION_1**
[+] **H5O_VERSION_2**

H5PLpublic.h
[+] **H5PL_ALL_PLUGIN**
[+] **H5PL_FILTER_PLUGIN**

H5pubconf.h
[+] **H5_HAVE_TIMEZONE**

to the top

Header Files (75)

H5AbstractDs.h
H5ACpublic.h
H5api_adpt.h
H5Apublic.h
H5ArrayType.h
H5AtomType.h
H5Attribute.h
H5Classes.h
H5CommonFG.h
H5CompType.h
H5Cpp.h

```
H5CppDoc.h
H5Cpublic.h
H5DataSet.h
H5DataSpace.h
H5DataType.h
H5DcreatProp.h
H5DOpublic.h
H5Dpublic.h
H5DSpublic.h
H5DxferProp.h
H5EnumType.h
H5Epubgen.h
H5Epublic.h
H5Exception.h
H5f90i.h
H5f90i_gen.h
H5FaccProp.h
H5FcreatProp.h
H5FDcore.h
H5FDdirect.h
H5FDfamily.h
H5FDlog.h
H5FDmpi.h
H5FDmpio.h
H5FDmulti.h
H5FDpublic.h
H5FDsec2.h
H5FDstdio.h
H5File.h
H5FloatType.h
H5Fpublic.h
H5Gpublic.h
H5Group.h
H5IdComponent.h
H5IMpublic.h
H5Include.h
H5IntType.h
H5Ipublic.h
H5Library.h
H5Location.h
H5Lpublic.h
H5LTpublic.h
H5MMpublic.h
H5Object.h
H5Opublic.h
H5overflow.h
H5PacketTable.h
H5PLextern.h
H5Ppublic.h
H5PredType.h
H5PropList.h
```

H5PTpublic.h
H5pubconf.h
H5public.h
H5Rpublic.h
H5Spublic.h
H5StrType.h
H5TBpublic.h
H5Tpublic.h
H5VarLenType.h
H5version.h
H5Zpublic.h
hdf5.h
hdf5_hl.h

to the top

Shared Libraries (6)

libhdf5.so.9.0.0
libhdf5_cpp.so.9.0.0
libhdf5_fortran.so.9.0.0
libhdf5_hl.so.9.0.0
libhdf5_hl_cpp.so.9.0.0
libhdf5hl_fortran.so.9.0.0

to the top

*Generated on Tue May 5 08:57:06 2015 for **hdf5** by ABI Compliance Checker 1.99.9*
A tool for checking backward compatibility of a C/C++ library API