RFC: Default EPSILON values for comparing floating point data

Albert Cheng (acheng@hdfgroup.org)
Peter Cao (xcao@hdfgroup.org)
Pedro Vicente (pvn@hdfgroup.org)
Neil Fortner (nfortne2@hdfgroup.org)

This RFC discusses how to set the default EPSILON values for comparing floating point data. The h5diff tool currently uses H5DIFF_FLT_EPSILON (0.00001 or 10E-6) and H5DIFF_DBL_EPSILON (10E-10) as default EPSILON values. Users reported problems with the current default setting because of the precision. This RFC proposes to change the current default.

1 H5diff floating point data comparison

1.1 How h5diff determines if two floating point type data are different

To determine if two floating point values, *float1* and *float2*, are different, one cannot use the simple comparison of (*float1* == *float2*) to decide if they are equal. Two floating point values can be slightly different but the difference is not numerically significant. Therefore h5diff use the following comparison to determine if two floating point data are different.

(ABS((float1- float2) / float2) < H5DIFF_FLT_EPSILON), where H5DIFF_FLT_EPSILON is hardcoded to 0.00001. The 0.00001, same as 10E-6, is the maximum permitted FLT_EPSILON value as defined in the C standard .

1.2 Problems using this fixed value

Since this fixed value of 10E-6 is the maximum limit and is larger than most system defined FLT_EPSILON values of 1.19209E-07 (see simple program in appendix 1), this means h5diff will determine two floating point data as the same when their relative difference is larger than the system FLT_EPSILON but smaller than the set H5DIFF_FLT_EPSILON value. That results in a false negative as “incorrectly reporting there is no significant difference”.

For the double type the value used is H5DIFF_DBL_EPSILON defined as .000000001, same as 10E-10.

2 Possible changes to h5diff

There are three possible options:

a) Stay with the current H5DIFF_FLT_EPSILON and H5DIFF_DBL_EPSILON
b) Use the system defined values (FLT_EPSILON and DBL_EPSILON)
c) Use a constants that is closer to the system defined values
d) Use strict equality as default

2.1 Stay with the current set H5DIFF_FLT_EPSILON value

2.1.1 Pros:

h5diff will maintain previous behavior, therefore users will not see new changes. H5diff will report the same differences in all platforms even if they may have different EPSILON values. One may view this as a platform independent behavior.

2.1.2 Cons:

H5diff will continue showing false negatives. Users may consider this behavior as an incorrect implementation.

2.2 Use the system defined values (FLT_EPSILON and DBL_EPSILON)

h5diff will use the system FLT_EPSILON values for comparisons when they are defined. If not, will use the current value of 10E-6. (See sample code in Appendix 2).

2.2.1 Pros:

H5diff reports differences according to the best of the system ability. H5diff will not report false negatives, at least not due to the above problem. Users who insist on the previous behavior may run h5diff with the "-p 10.0E-6" option.

2.2.2 Cons:

H5diff will show results different from before. This may upset some users. On the other hand, will a user insist on the previous false negatives once the reasons of the change are explained?

2.3 Use a constant that is closer to the system defined value

Do a complete survey of the system defined value in systems we have access and change the set H5DIFF_FLT_EPSILON to a value closer to all these system defined values.

2.3.1 Pros:

H5diff will report the same differences in all platforms even if they may have different EPSILON values. One may view this as a platform independent behavior. (See cons below.)

2.3.2 Cons:

H5diff will continue showing false negatives wherever the new H5DIFF_FLT_EPSILON is larger than the system EPSILON values.

H5diff will show false positives in a new platform which has EPSILON values that are larger than the new H5DIFF_FLT_EPSILON but less than the C standard maximum of 10E-6.

Users may consider this behavior as an incorrect implementation.

2.4 Use strict equality as default

Daniel Kahn and Christopher Lynnes proposed to use strict equality as default. Daniel Kahn pointed out "*However, the constant you have used is essentially arbitrary and you are guessing that it will be the one the user needs. I view the guessing game as hopeless. Strict equality will be most the most conservative test, if users aren't happy about it they will be forced to think about what tolerance is best for their problem. It has always been the user's responsibility to choose the tolerance for the problem at hand, by making this explicit the HDF Group confirms its neutrality in this*." The main point is that the tool should not guess what users need and should leave it to the users.

2.4.1 Pros:

The comparison is simple and straightforward, and has better performance.

2.4.2 Cons:

As pointed out at section 1.1, h5diff may report two numbers are different even though they are the same due to machine precision using strict equality.

3 Conclusions

Based users' input and the technical discussions of the HDF5 developer's meeting, we made a conclusion as the following:

- Use strict equality as default
- Use "--use-system-epsilon" for system EPSILON
    - Use "|a-b| > EPSILON" for comparison
    - If the system epsilon is not defined, use the value below:
        - FLT_EPSILON = 1.19209E-07 for float
        - DBL_EPSILON = 2.22045E-16 for double
- Use "-p" or "-d" for whatever user's choice of epsilon
- Use "-p 0" or "-d 0" for strict equality (same as default)
- The "--use-system-epsilon", "-d", and "-p" options are mutually exclusive. Using them together will cause error
- Document all options and the reasons

4 Documentation (h5diff –h)

4.1 Current:

```
   -d D, --delta=D         Print difference when greater than limit D
   -p R, --relative=R      Print difference when greater than relative limit R


  D - is a positive number. Compare criteria is |a - b| > D
  R - is a positive number. Compare criteria is |(b-a)/a| > R
```

4.2 Proposed:

```
   -d D, --delta=D         Print difference if (|a-b| > D), D is a positive
number.
   -p R, --relative=R      Print difference if (|(a-b)/b| > R), R is a positive
number.
   --use-system-epsilon    Print difference if (|a-b| > EPSILON), EPSILON is
                           a system epsilon value.
                           If the system epsilon is not defined, the below
                           one of the following predefined values will be used:
                             FLT_EPSILON = 1.19209E-07 for floating-point type
                             DBL_EPSILON = 2.22045E-16 for double percision type
```

Revision History

June 12, 2009:	Version 1 circulated for comment within The HDF Group.
September 10, 2009:	Version 2 added changes based on the comments from NASA users and the discussion at The HDF Group.
May 11, 2011:	Version 2This revision added clarification about options are exclusive with each other.