RFC: command tool interface for handling attribute name contains ‘/’ character(s)

Jonathan Kim

The ‘/’ character is used as object delimiter, so it’s not allowed to be used as part of an object name. However ‘/’ character can be used as part of an attribute name.

The h5dump tool causes a problem when a user try to specify attribute name which contains ‘/’ character(s) along with the object path in a single argument.

This RFC explains more details and suggestions.

# Background

This topic was discussed for h5diff previously and is revisited due to the h5dump issue; “[HDFFV-7523](http://jira.hdfgroup.uiuc.edu/browse/HDFFV-7523) h5dump fails to open an attribute when using the -a option to specify an attribute with a slash in the name “.

In the h5dump issue, h5dump tool has an option “-a” to let user specify an attribute name to display only the attribute information. However the option requires specifying attribute name along with the full path to an object, and also requires ‘/’ character as a delimiter between object path and attribute.

So h5dump causes a problem, when a user specifies an attribute name which contains ‘/’ character with “-a” option, because h5dump treats the string which comes after the last ‘/’ in the full path as an attribute name since the name became only a portion.

Basically h5dump code was written with an assumption that an attribute name should not contain ‘/’ character as part of the name, but didn’t fully enforce to disallow the character also for the attribute name. As of the result, the assumption became faulty and causes the issue.

From development and maintenance point of view, it would be most effective if the ‘/’ character could be disallowed to be used for a part of attribute name, same as an object name. However it may be too late to enforce it at this point.

This topic had been discussed previously for the h5diff’s option interface for the excluding attribute feature.

Even the issue is originated from the h5dump, it’s appropriate to think from general tool’s view point.

# Suggestions for general tool’s interface

To allow ‘/’ as part of an attribute name, it would be necessary to specify attribute name separate from object path.

## Using two options

## **-O “**<full path to an object>**” -a “**<only attribute name>**”**

If a tool needs an object, a user can use the *–O “<full path to an object>”* option. If a tool needs an attribute, a user can use *“–a”* option along with *“-O”* option.

**Example**:

* h5dump –O “/g1/g2/dset1” -a “speed/hour/high” hdf5file.h5
* h5ls –O “/g1/g2/dset1” --attribute=“speed/hour/high” hdf5file.h5

**Long option names**:

* “-O” or “--object-path”
* “-a” or “—attribute” ( --attribute is used for h5dump and h5stat at this point )

## Using one option with two separate pair of double quotes

## **-O “**<full path to an object>**” “**<only attribute name>**”**

This method was discussed for h5diff’s excluding attribute option.

**Example**:

* h5dump –O “/g1/g2/dset1” “speed/hour/high” hdf5file.h5
* h5ls –O “/g1/g2/dset1” “speed/hour/high” hdf5file.h5

# Suggestion for the h5dump specific case

This is for a case that we’d rather address the issue only for the h5dump and go along with the current h5dump’s interface. For reminder, the h5dump issue is “[HDFFV-7523](http://jira.hdfgroup.uiuc.edu/browse/HDFFV-7523) h5dump fails to open an attribute when using the “-a” option to specify an attribute with a slash in the name”.

The h5dump currently has three options to display each object.

* “-d” for displaying a dataset
* “-g” for displaying a group
* “-t” for displaying a named data type

Currently “-a” requires to specifying attribute name with full path to an object.

Suggestion would be allowing “-a” only for an attribute name when used with one of the object options. If “-a” is used alone, stay same.

**Example**:

* h5dump –d “/g1/g2/dset1” -a “speed/hour/high” hdf5file.h5

: display only the attribute “speed/hour/high” of the dataset “dset1”

* h5dump –g “/g1/g2” -a “speed/hour/mid” hdf5file.h5

: display only the attribute “speed/hour/mid” of the group “g2”

* h5dump -a “/g1/g2/attr1” hdf5file.h5

: display only the attribute “attr1” of the group “/g1/g2”

# Considered ideas, but not feasible

List ideas, but turned out to be not feasible after investigation.

## Make tool parser smarter

It may sound possible to make tool’s parser smarter by checking the each object & attribute existence in given path divided by ‘/’ characters by parsing backwardly. However there are cases that it’s impossible to distinguish for the correct attribute, which makes this idea not feasible.

Example situation

Assume there is a HDF5 file like below:

/ ( root group has an attribute “a/b/c”)

|-- “a” ( the group “a” has an attribute “b/c” )

| |-- “b” ( the group “b” has an attribute “c” )

| |-- “c” (dataset “c” )

|

* Root group’s has an attribute: “a/b/c”
* Group “/a” has an attribute “b/c”
* Group “/a/b” has an attribute “c”

Issue

In this example, “/a/b/c” could mean various as shown below. Parser cannot determine which attribute is what user means.

* The attribute “a/b/c” of the root group
* The attribute “b/c” of the group “/a”
* The attribute “c” of the group “/a/b”

# Revision History

|  |  |
| --- | --- |
| *September 21, 2011:* | Version 1 circulated for initial feedback (Peter & Frank) |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |