Feedback of "Special Values in HDF5" RFC

Kent Yang

Jan. 30th, 2009

I. Background

HDF5 supports a single default or user-defined fill value during dataset creation. However, there are cases where a user may wish to specify more than one "special" value to describe non-standard data. Based on a request from the 2008 NASA HDF annual briefing, we wrote a RFC and sent out to NASA ESDIS and DAAC developers and hdf-forum mailing list. The RFC can be found at http://www.hdfgroup.org/pubs/rfcs/RFC_Special_Values_in_HDF5.pdf.

To facilitate the discussions below, two most relevant sections 4.3.3 and 4.3.4 in the RFC are quoted here.

4.3.3 Parallel Special Values Dataset

Another idea, brought to our attention by a user, would be to write the special values data outside the original dataset by defining a new "parallel" dataset which contains all special value information. This new dataset would have the same rank and dimension as the original dataset, but only contain data for special values. The special values could be defined in the parallel dataset using either of the implementations above. An attribute could be added to the original dataset with an object pointer that "links" to the special values dataset. This design has a few advantages. First of all, the parallel dataset is no longer constrained to using the datatype of the original dataset. Thus, we don't need to worry about reserving a potentially large number of values as "special" in the original data. Also, we have the opportunity to use a more space-efficient datatype, such as an enumeration or a bit-mask. Another advantage is that many datasets can refer to the same parallel dataset. For example, if we have datasets that describe the temperature, pressure, and elevation over the same area, then we may define special values (such as "water-filled" and "ice-covered") that are relevant and shared by each dataset. This design requires additional physical space equal to the size of the original dataset. In most situations, this overhead would be greatly reduced by compression inside HDF5.

4.3.4 Attribute Triplet

Similar to the previous specification, "Dataset Attribute" (4.3.2), we can use an attribute, but use it to store all of special values information. This includes not only the definition for each special value but also the region information as well. Each special value definition would be a triplet of a string (for a name or description), a value (using any defined datatype), and a dataset region reference that contains the selection of elements in the dataset that the value applies to. The dataset would contain an array of these appropriately-defined compound datatype to store the triplet. This particular implementation allows for non-mutual exclusion for special values. A user could even overlay special value regions where real data samples

exist. This implementation would also benefit greatly if the HDF5 library adds support for simple operations on dataset region selections (i.e. union, intersect, complement).

II. Summary of the Feedback

We received six replies. Two of them were from NASA Aura developers. Since the original request also comes from a NASA Aura developer, so we have totally three responses from NASA. They are all Aura developers and represent OMI, HIRDLS and MLS instrument teams.

All replies suggest *not adding the support inside HDF5 libraries, either low-level or high-level.*

Most replies prefer the Parallel Special Values Dataset approach in section 4.3.3 due to its simplicity. Some people think the compression can help reduce the additional physical space when using a parallel dataset to store special values. Some people point out those solutions proposed in 4.3.3 and 4.3.4 can address different use cases.

One NASA developer strongly opposes the support of special values in HDF5. He suggests we shouldn't do anything with this.

Table 1: Summaries of feedback from NASA ESDIS users

Aura Developers	Opinions
MLS	Don't do anything; simply ignore this topic
HIRDLS	Make it simple; prefers "Parallel Special Values Dataset"
OMI	Propose "Parallel Special Values Dataset"

III. Proposal

Based on the feedback, I propose that we provide two examples to demonstrate how to handle special values in an HDF5 application. The purpose is to provide references for those applications that need to handle special values in an HDF5 application. One example is to follow the solution addressed in section 4.3.3 and another example is to follow the solution addressed in section 4.3.4. We will post these examples at our website, probably under FAQ or an appropriate category.

Here are some requirements of these examples:

1. A simple paragraph to describe the issue and the proposed solution
2. One well-commented simple C program
3. One well-commented simple Fortran 90 program