

A Proposal for automated testing of NPOESS/JPSS files with the HDF5 Augmentation Tool

Larry Knox
Elena Pourmal

Judging by the identifiers for the files available from the GRAVITE server, it has now hosted more than 50,000,000 different files, a majority of them HDF5 files. The files are currently added (and possibly deleted) at a rate on the order of 2000 files per hour. The HDF5 augmentation tool for NPOESS/JPSS product files has been tested manually with some three dozen of these files, identifying mismatches between the XML file and the HDF5 file in 15-20% of the files. In order to find as many different types of these mismatches in as many different categories of NPOESS files as soon as possible we propose to run the tool as part of the HDF5 continuous backward/forward compatibility tests of NPOESS files from the GRAVITE server. A beta version of the tool is available from <ftp://ftp.hdfgroup.uiuc.edu/pub/outgoing/NPOESS/h5augjpss>. See the [h5augjpss-design.pdf](#) and [src/README.txt](#) files for descriptions of the tool and its purpose.

1 Introduction

Running the HDF5 augmentation tool on files that have been downloaded from the GRAVITE server has shown that the NPOESS/JPSS HDF5 files on the server are sometimes missing one or more datasets referenced in the corresponding XML file. The XML file provides information for adding dimension scales and other metadata to the datasets in the HDF5 files from the server to give scientific meaning to the data when it is opened with netCDF-4. The `-l 2` option for the augmentation tool parses the XML file corresponding to the HDF5 file and for each “field” adds the dimension scales and other attributes to the datasets in the HDF5 file that have the same name as the field. Mismatches between the two files may result in missing attributes or aborting the process.

The HDF Group NPOESS backward/forward compatibility tests continually download files from the GRAVITE server and test for files with backward/forward compatibility problems. The HDF5 augmentation tool testing can be added for the same files. Since the augmentation tool testing is a shorter process, the number of files downloaded can be increased and the backward forward compatibility tests can be run on a subset of the downloaded files.

The specific h5augjpss tests will exercise the four `-l` or `--level` command line options. These options do the following:

1. Make the HDF5 file readable by netCDF-4.
2. Add dimension scales and other metadata to increase the scientific meaning of the data.

3. Add geo-location datasets from a second HDF5 file to add Latitude and Longitude (and Height) coordinates for each relevant (?) data element in the NPOESS/JPSS file.
4. Make the HDF5 file readable for tools that do not support group structures.

Running these tests will report the following:

- mismatches between the contents of the XML and HDF5 files
- HDF5 files for which no corresponding XML file is available
- files with an .h5 extension that are not HDF5 files
- any other warnings or errors generated by the tool

2 Reasons for automated testing of files with h5augjpss

There are several reasons to automate testing of these files with h5augjpss. One is to identify as many of these mismatches as possible so they can be addressed before the system is put into production. Another is to validate certain assumptions that have been made about the structure of the NPOESS HDF5 files or to identify exceptions. Running these tests manually will not sample a sufficient number of the files that are being added to the server, and automated testing will probably increase the types and variations of files tested.

3 Test Procedure for each file

a. Test that option -l 1 makes the HDF5 file readable by netCDF-4, using the ncdump utility as a reader.

The netCDF-4 utility ncdump fails to open NPOESS files, and instead outputs this message: "NetCDF: Bad type ID", indicating the presence of objects not understood by netCDF-4. After running the command "h5augjpss -l 1 <filename>", "ncdump -h <filename>" should display a list of the objects in the file.

b. Test that option -l 2 adds attribute information according to the XML file that matches the group in the /ALL_Data group.

The h5augjpss tool checks that the field names in the XML file match the dataset names in the HDF5 file and that dataset types and sizes match the information in the XML file. This step should add a number of attributes and remove all occurrences of "phony_dim" by adding dimension scales. The tool will report discrepancies as it creates attributes from the information in the XML file. The mismatches between the XML file and the HDF5 file will be identified by this option.

c. Test that option -l 3 copies geo-location datasets from HDF5 file specified by N_GEO_Ref.

The file specified by the N_GEO_Ref attribute in the HDF5 file should exist and the command should add “Height”, “Latitude” and “Longitude” datasets to the HDF5 file. Absence of the specified file, failure to add the datasets, or warnings or errors generated by running the command should all be reported.

d. Test that option -l 4 hides /All_Data group structure.

Option -l 4 should create links to the datasets beneath the /All_Data group, then hide the /All_Data group and its subgroup. Failure to create the links or to remove the group should be reported along with any errors or warnings.

e. Reporting

The name of each file tested should be logged in a file followed by any mismatches, warnings, or errors. The log file should periodically be posted to a location available to interested parties. A summary of the log entries could be emailed at the same time.

4 Recommendation

It would be best to implement this testing as soon as possible. Doing so will potentially allow The HDF Group to find and correct defects in the tool that will not be found otherwise and will provide more information to JPSS regarding incorrect or missing XML and geo-location files corresponding to available HDF5 files before the system is put in production.