

RFC: Writing a Userblock for NPP files with nagg

Larry Knox
Elena Pourmal

This document describes a new nagg feature which will write NPP metadata elements in a userblock in nagg output files, according to the specification in the *JPSS Common Data Format Control Book – External – Volume V*, Section 3.2.

Copyright 2013 by The HDF Group.

All rights reserved.

For more information about The HDF Group, see www.hdfgroup.org.

Table of Contents

1. Introduction	4
2. Userblock Requirements.....	5
3. Implementation Considerations	6
3.1. Top Level Elements	6
3.2. Data_Product Sub-elements	6
3.3. nagg Structures for Userblock Element Values.....	7
4. Unaddressed Issue	8
5. Additional Utility Applications	9
6. Summary and Recommendations.....	10
7. Revision History	11
8. Appendix 1 - XML Userblock Schema.....	12
9. Appendix 2 - Userblock Example.....	14

1. Introduction

nagg (http://www.hdfgroup.org/projects/jpss/nagg_index.html) is a tool for aggregating JPSS data products from existing files into new files with a different number of granules per file or different combinations of compatible products. JPSS data files have a userblock that contains XML elements which are a subset of the metadata elements in the file, but can be accessed without HDF5 tools or libraries. These elements are attributes of the root group, the `/Data_Products/<Collection Short Name>` groups, or the `/Data_Products/<Collection Short Name>/<Collection Short Name>_Aggr` datasets in the file. The userblock for the new file should contain these elements with the appropriate values for the products and aggregations in the new file. This RFC enumerates the currently required elements of this userblock and the locations of the corresponding attributes for obtaining the element values.

While the userblock element names match the names of the corresponding attributes, the location of those attributes is not given in the userblock, and the values of the aggregation's attributes are determined as the file is written. Although the current JPSS specification for the userblock appears to be static, nagg should read the userblocks of the input files to check for unexpected new elements or user additions and issue a warning if any are found, as they will not be written in the output file.

2. Userblock Requirements

The XML schema for JPSS data product files of type SDR, TDR, EDR, ARP, GEO, and IP is specified in the “Metadata” section of *JPSS Common Data Format Control Book – External Volume V* and is included in Appendix 1 (see page 12).

The userblock will contain exactly one of each of the first four elements and one or more of the `Data_Product` elements to match the number of JPSS data products included in the file. While nagg does not currently support aggregation of RDR data products, note for future reference that the userblock for RDRs differs only by the absence of the `N_GEO_Ref` element.

Each `Data_Product` element will contain exactly one of the following elements:

- `N_Collection_Short_Name`
- `Instrument_Short_Name`
- `N_Dataset_Type_Tag`
- `N_Processing_Domain`
- `AggregateBeginningDate`
- `AggregateBeginningOrbitNumber`
- `AggregateBeginningTime`
- `AggregateEndingDate`
- `AggregateEndingOrbitNumber`
- `AggregateEndingTime`
- `AggregateBeginningGranuleID`
- `AggregateEndingGranuleID`

An example of a userblock for a file with two products is included in Appendix 2 (see page 14).

3. Implementation Considerations

nagg copies the values of the attributes corresponding to the userblock elements from the input file to the output file in the course of processing the files. It should be most efficient to read and save the values for the userblock at that time or to replace the copy by reading and writing them to the attribute and the userblock string at the time they are copied from the input granule. The required size of the userblock can be calculated before creating the file according to the element names and types and the number of products to be packaged in the output file. Element values can be saved for writing after the product granules are processed for an output file.

A list of expected elements will be useful for checking the input files' userblocks. If more flexibility is required in the future, this list could be put in a configuration file to be read at runtime. Entries would need an HDF5 path for the corresponding attribute. This is not part of the work currently proposed.

3.1. Top Level Elements

The top level elements are described in this section.

`Mission_Name` and `Platform_Short_Name` are root group attributes as well as userblock elements. Their values should be the same for all input and output files.

The `N_GEO_Ref` element for the output file userblock will have the same value as the `/N_GEO_Ref` attribute. If the geolocation is packaged, the `/N_GEO_Ref` attribute will not be created. The userblock element will be present, but empty ("`<N_GEO_Ref></ N_GEO_Ref>`").

The value of the `Number_of_Data_Products` element will be determined by the number of data products packaged in each output file (including the geolocation product).

One `Data_Product` element is written to the userblock for each product included in the file. The `Data_Product` element has sub-elements that are described in the section below.

3.2. Data_Product Sub-elements

The `Data_Product` sub-elements are described in this section.

The first four elements correspond to a subset of the attributes of the `/Data_Products/<Collection Short Name>` group which use the same names:

`N_Collection_Short_Name`
`Instrument_Short_Name`

```
N_Dataset_Type_Tag
N_Processing_Domain
```

These values are copied by nagg from the first input file to the output file.

The remaining eight elements correspond to a subset of attributes of the
/Data_Products/<Collection Short Name>/<Collection Short Name>_Aggr dataset:

```
AggregateBeginningDate
AggregateBeginningOrbitNumber
AggregateBeginningTime
AggregateEndingDate
AggregateEndingOrbitNumber
AggregateEndingTime
AggregateBeginningGranuleID
AggregateEndingGranuleID
```

The values for these attributes are copied by nagg from the first and last granules in the aggregation. They can also be read and written to the userblock.

3.3. nagg Structures for Userblock Element Values

The nagg structures to be added will be similar to these. Character arrays can be replaced by char *s where information is obtained from persistent granules.

```
struct user_block_t {
    char mission_name[];
    char platform_name[];
    char *geofile_name;
    unsigned int number_data_products;
    struct data_product_info_t *data_products[];
}

struct data_product_info_t {
    char product_name [];
    char instrument_name[];
    char dataset_type_tag[];
    char processing_domain[];
    char beginning_date[];
    uint64_t beginning_orbit;
    char beginning_time[];
    char ending_date[];
    uint64_t ending_orbit;
    char ending_time[];
    char beginning_granule_id[];
    char ending_granule_id[];
}
```

4. Unaddressed Issue

The proposed change to the nagg application will not address situations where input files have different values for `/N_Processing_Domain`. The value of the root group attribute for the output file is copied from the root group attribute in the input file for the first granule to be aggregated. If later granules in the same aggregation are from a file with a different value for `/N_Processing_Domain`, that value will be ignored for the current aggregation. There is no provision for this situation, which applies to several other attributes in the CDFCB specification.

5. Additional Utility Applications

If the proposed change to nagg is implemented, it would be good if two other utility programs were also developed.

A test program should be written to read the XML tags in the userblock and compare the values to those found by reading the corresponding attributes of the root group, the product groups, and the `<product>_aggr` datasets. This program would alert users to any unexpected attributes.

A utility to output the userblock with carriage returns and indentations as displayed in the appendix would also be useful. This program would produce text that might be manipulated with standard tools such as GREP to compare the input and output attributes and alert users to anomalies.

6. Summary and Recommendations

nagg currently produces files without the information expected in the userblock. The values can be determined when nagg is creating new files, and for most nagg operations the majority of the values in the userblock must be determined when creating the output file, as they will be different from the values in the input files. The userblock should be added with the implementation suggested by the information in the “Implementation Considerations” section on page 6.

7. Revision History

<i>September 20, 2013:</i>	Version 1 available for internal review.
<i>October 22, 2013</i>	Version 2 committed to subversion repository.
<i>October 25, 2013</i>	Version 3. Edited text.

8. Appendix 1 - XML Userblock Schema

The XML userblock schema shown below is for HDF5 SDR, TDR, EDR, ARP, GEO, and IP products. This schema can be found in *JPSS CDFCB – External – Volume V*, Section 3.2.1.

```
<?xml version="1.0"?>
<xsd:schema xmlns:xsd=http://www.w3.org/2001/XMLSchema targetNamespace=http://
"NPOESS SDR, TDR, EDR, ARP, and IP XML User Block" xmlns="http://NPOESS
SDR, TDR, EDR, ARP, and IP XML User Block">
<!-- == NPP/NPOESS HDF5 XML User Block XML Schema == -->
<!-- == NPP/NPOESS HDF5 User Block - Primary Element== -->
<xsd:element name="HDF_UserBlock" type="HDF_UserBlockType" />
<!-- == Annotation for this schema == -->
<xsd:annotation>
<xsd:documentation xml:lang="en">
NPP/NPOESS HDF5 User Block XML Schema
</xsd:documentation>
</xsd:annotation>
<!-- == NPP/NPOESS HDF5 User Block == -->
<xsd:complexType name="HDF_UserBlockType">
<xsd:sequence>
<xsd:element name="Mission_Name" type="xsd:string" minOccurs="1"
maxOccurs="1" />
<xsd:element name="Platform_Short_Name" type="xsd:string"
minOccurs="1" maxOccurs="1" />
<xsd:element name="N_GEO_Ref" type="xsd:string"
minOccurs="0" maxOccurs="1" />
<xsd:element name="Number_Of_Data_Products" type="xsd:integer"
minOccurs="1" maxOccurs="1" />
<xsd:element name="Data_Product" type="Data_ProductType"
minOccurs="1" maxOccurs="unbounded" />
</xsd:sequence>
</xsd:complexType>
<!-- == Data Product Type == -->
<xsd:complexType name="Data_ProductType">
<xsd:sequence>
<xsd:element name="N_Collection_Short_Name" type="xsd:string"
minOccurs="1" maxOccurs="1" />
<xsd:element name="Instrument_Short_Name" type="xsd:string"
minOccurs="1" maxOccurs="1" />
<xsd:element name="N_Dataset_Type_Tag" type="xsd:string"
minOccurs="1" maxOccurs="1" />
<xsd:element name="N_Processing_Domain" type="xsd:string"
minOccurs="1" maxOccurs="1" />
<xsd:element name="AggregateBeginningDate" type="xsd:string" minOccurs="1"
maxOccurs="1" />
<xsd:element name="AggregateBeginningOrbitNumber"
type="xsd:integer" minOccurs="1" maxOccurs="1" />
<xsd:element name="AggregateBeginningTime" type="xsd:string"
minOccurs="1" maxOccurs="1" />
<xsd:element name="AggregateEndingDate" type="xsd:string"
minOccurs="1" maxOccurs="1" />
<xsd:element name="AggregateEndingOrbitNumber" type="xsd:integer"
minOccurs="1" maxOccurs="1" />
<xsd:element name="AggregateEndingTime" type="xsd:string"
minOccurs="1" maxOccurs="1" />
<xsd:element name="AggregateBeginningGranuleID" type="xsd:string"
minOccurs="1" maxOccurs="1" />

```

```
<xsd:element name="AggregateEndingGranuleID" type="xsd:string"
minOccurs="1" maxOccurs="1" />
</xsd:sequence>
</xsd:complexType>
</xsd:schema>
```

9. Appendix 2 - Userblock Example

Here is a userblock example for a packaged file with four GMODO granules and four SVM01 granules:

```
<HDF_UserBlock><Mission_Name>NPP</Mission_Name><Platform_Short_Name>NPP</Platform_Short_Name><N_GEO_Ref></N_GEO_Ref><Number_Of_Data_Products>2</Number_Of_Data_Products><Data_Product><N_Collection_Short_Name>VIIRS-M1-SDR</N_Collection_Short_Name><Instrument_Short_Name>VIIRS</Instrument_Short_Name><N_Dataset_Type_Tag>SDR</N_Dataset_Type_Tag><N_Processing_Domain>ops</N_Processing_Domain><AggregateBeginningDate>20120813</AggregateBeginningDate><AggregateBeginningOrbitNumber>4115</AggregateBeginningOrbitNumber><AggregateBeginningTime>094839.337354Z</AggregateBeginningTime><AggregateEndingDate>20120813</AggregateEndingDate><AggregateEndingOrbitNumber>4115</AggregateEndingOrbitNumber><AggregateEndingTime>095419.727577Z</AggregateEndingTime><AggregateBeginningGranuleID>NPP000255233200</AggregateBeginningGranuleID><AggregateEndingGranuleID>NPP000255235761</AggregateEndingGranuleID></Data_Product><Data_Product><N_Collection_Short_Name>VIIRS-MOD-GEO</N_Collection_Short_Name><Instrument_Short_Name>VIIRS</Instrument_Short_Name><N_Dataset_Type_Tag>GEO</N_Dataset_Type_Tag><N_Processing_Domain>ops</N_Processing_Domain><AggregateBeginningDate>20120813</AggregateBeginningDate><AggregateBeginningOrbitNumber>4115</AggregateBeginningOrbitNumber><AggregateBeginningTime>094839.337354Z</AggregateBeginningTime><AggregateEndingDate>20120813</AggregateEndingDate><AggregateEndingOrbitNumber>4115</AggregateEndingOrbitNumber><AggregateEndingTime>095419.727577Z</AggregateEndingTime><AggregateBeginningGranuleID>NPP000255233200</AggregateBeginningGranuleID><AggregateEndingGranuleID>NPP000255235761</AggregateEndingGranuleID></Data_Product></HDF_UserBlock>
```

The same string with carriage returns between fields, indented for nesting is repeated below, color coded **yellow** for top level elements, **blue** for data product 1 sub-elements, and **olive** for data product 2 sub-elements.

```
<HDF_UserBlock>
<!--Root Group Attributes -->
  <Mission_Name>NPP</Mission_Name>
  <Platform_Short_Name>NPP</Platform_Short_Name>
  <N_GEO_Ref></N_GEO_Ref>
  <Number_Of_Data_Products>2</Number_Of_Data_Products>
  <Data_Product>
    <N_Collection_Short_Name>VIIRS-M1-SDR</N_Collection_Short_Name>
    <Instrument_Short_Name>VIIRS</Instrument_Short_Name>
    <N_Dataset_Type_Tag>SDR</N_Dataset_Type_Tag>
    <N_Processing_Domain>ops</N_Processing_Domain>
    <AggregateBeginningDate>20120813</AggregateBeginningDate>
    <AggregateBeginningOrbitNumber>4115</AggregateBeginningOrbitNumber>
    <AggregateBeginningTime>094839.337354Z</AggregateBeginningTime>
    <AggregateEndingDate>20120813</AggregateEndingDate>
    <AggregateEndingOrbitNumber>4115</AggregateEndingOrbitNumber>
    <AggregateEndingTime>095419.727577Z</AggregateEndingTime>
```

No matching attribute

Product group attributes

Aggregate dataset attributes

```
<AggregateBeginningGranuleID>NPP000255233200</AggregateBeginningGranuleID>
<AggregateEndingGranuleID>NPP000255235761</AggregateEndingGranuleID>
</Data_Product>
<Data_Product>
  <N_Collection_Short_Name>VIIRS-MOD-GEO</N_Collection_Short_Name>
  <Instrument_Short_Name>VIIRS</Instrument_Short_Name>
  <N_Dataset_Type_Tag>GEO</N_Dataset_Type_Tag>
  <N_Processing_Domain>ops</N_Processing_Domain>
  <AggregateBeginningDate>20120813</AggregateBeginningDate>
  <AggregateBeginningOrbitNumber>4115</AggregateBeginningOrbitNumber>
  <AggregateBeginningTime>094839.337354Z</AggregateBeginningTime>
  <AggregateEndingDate>20120813</AggregateEndingDate>
  <AggregateEndingOrbitNumber>4115</AggregateEndingOrbitNumber>
  <AggregateEndingTime>095419.727577Z</AggregateEndingTime>
  <AggregateBeginningGranuleID>NPP000255233200</AggregateBeginningGranuleID>
  <AggregateEndingGranuleID>NPP000255235761</AggregateEndingGranuleID>
</Data_Product>
</HDF_UserBlock>
```

Product group attributes

Aggregate dataset attributes

The top level elements, those in yellow in the sample above, are described in the table below.

Table 1. Top level elements

Element	Comments
Mission_Name	Matches an attribute of the root group. Should be the same for all input and output files.
Platform_Short_Name	Matches an attribute of the root group. Should be the same for all input and output files.
N_GEO_Ref	Matches a root group attribute that is present only for unpackaged files. This will be an empty field in the userblock for files with packaged geolocation.
Number of Data_Products	The number of data products packaged in the file.
Data_Product	There is one of these for each product in the output file. The sub-elements that appear in this element are listed in the table below.

The sub-elements of Data_Product, those in blue and olive in the sample above, are listed in the below.

Table 2. Data_Product sub-elements

Sub-element	Comments
N_Collection_Short_Name	Matches an attribute of the Data_Products child group named with the value of the N_Collection_Short_Name field.
Instrument_Short_Name	Matches an attribute of the Data_Products child group named with the value of the N_Collection_Short_Name field.
N_Dataset_Type_Tag	Matches an attribute of the Data_Products child group named with the value of the N_Collection_Short_Name field.
N_Processing_Domain	Matches an attribute of the Data_Products child group named with the value of the N_Collection_Short_Name field.
AggregateBeginningDate	Matches an attribute of the Aggregate dataset of the Data_Products child group named with the value of the N_Collection_Short_Name field.
AggregateBeginningOrbitNumber	Matches an attribute of the Aggregate dataset of the Data_Products child group named with the value of the N_Collection_Short_Name field.
AggregateBeginningTime	Matches an attribute of the Aggregate dataset of the Data_Products child group named with the value of the N_Collection_Short_Name field.
AggregateEndingDate	Matches an attribute of the Aggregate dataset of the Data_Products child group named with the value of the N_Collection_Short_Name field.
AggregateEndingOrbitNumber	Matches an attribute of the Aggregate dataset of

Table 2. Data_Product sub-elements

Sub-element	Comments
	the Data_Products child group named with the value of the N_Collection_Short_Name field.
AggregateEndingTime	Matches an attribute of the Aggregate dataset of the Data_Products child group named with the value of the N_Collection_Short_Name field.
AggregateBeginningGranuleID	Matches an attribute of the Aggregate dataset of the Data_Products child group named with the value of the N_Collection_Short_Name field.
AggregateEndingGranuleID	Matches an attribute of the Aggregate dataset of the Data_Products child group named with the value of the N_Collection_Short_Name field.