
**Reference Manual
for the
NPP Granule Aggregation and Packaging Utility (nagg)**

Version 1.6.2

July 2016



Copyright Notice and License Terms for HDF NPP Aggregation Tool (nagg)

HDF NPP Aggregation Tool (nagg)
Copyright 2011-2016 by The HDF Group.

All rights reserved.

Redistribution and use in source and binary forms, with or without modification, are permitted for any purpose (including commercial purposes) provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions, and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions, and the following disclaimer in the documentation and/or materials provided with the distribution.
3. In addition, redistributions of modified forms of the source or binary code must carry prominent notices stating that the original code was changed and the date of the change.
4. All publications or advertising materials mentioning features or use of this software are asked, but not required, to acknowledge that it was developed by The HDF Group and credit the contributors.
5. Neither the name of The HDF Group nor the name of any Contributor may be used to endorse or promote products derived from this software without specific prior written permission from The HDF Group or the Contributor, respectively.

DISCLAIMER:

THIS SOFTWARE IS PROVIDED BY THE HDF GROUP "AS IS" WITH NO WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED. In no event shall The HDF Group or the Contributors be liable for any damages suffered by the users arising out of the use of this software, even if advised of the possibility of such damage.

The software was developed with support from Riverside Technology, Inc. and The National Environmental Satellite, Data and Information Service (NESDIS) under Prime Contract No. DG133E07CQ0055.

Tool Name: nagg

Syntax:

```
nagg -h | --version

nagg [--version] [-n N] [-t list | -l example] [-g criterion] [-S]
[-A seconds] [-d directory] [-O origin] [-D domain] [--debug]
[--onefile] [--nofill] [--filter=filter] [--layout=layout]
[--minimum=size] INPUT
```

Purpose:

To manipulate granules stored in NPP files into more easily processed NPP files.

Description:

The command-line utility `nagg` is the utility for grouping NPP granules into aggregate and package files.

Aggregates include time contiguous and gap-filled series of the same type of granules together in the same file. Packages include different, but related type granule aggregates with the same temporal and spatial extent together in the same file.

Input parameter `INPUT` is a list of one or more NPP files with the granules to aggregate and/or package.

Output of `nagg` is a file or set of files named according to the NPP file naming convention. Output files are aligned with the bucket boundary. For more information on bucket boundaries, see *JPSS Common Data Format Control Book – External Volume I*, pp. 131-133. Fill granules are generated when there are no data granules available in the input files for "slots" in a bucket. Leading and trailing fill granules are not generated for the first and last output files respectively. Therefore, first and last output files may contain fewer granules than requested.

The default behavior of `nagg` is to aggregate corresponding geolocation granules. `nagg` will find these corresponding granules if they are available in the input files listed or stored in external geolocation files. The geolocation granules in input files can be processed whether packaged with data product granules or existing in separate files. For external geolocation files the names of the file(s) can be found from the value(s) of an `N_GEO_Ref` attribute in the input file. The names of the output geolocation file(s) correspond temporally to those of the primary product.

`nagg` will fail if no input geolocation granule is available. The aggregating of geolocation granules can be overridden with the `-g no` command-line option which directs `nagg` to not use or look for geolocation input granules or files and to not produce geolocation output granules or files. For certain geolocation products associated with a designated geolocation product that has an equivalent geolocation product `nagg` will process either terrain corrected or not terrain corrected geolocation granules as designated with the `-g <product>` option (See RFC: Nagg extension for terrain corrected geolocation).

Geolocation file names can be approximate such that only the beginning parts up to the creation date are matched. Suppose the sensor data file defines its geolocation file as:

GMTCO_npp_d20100906_t0701368_e0703013_b00004_c20111024161933653314_noaa_ops.h5. nagg will accept any file matching the following pattern and will use the one with the latest creation timestamp: GMTCO_npp_d20100906_t0701368_e0703013_b00004_c*.h5. The `-g strict` command-line option can be used to tell nagg that the external geolocation file name must exactly match the `N_GEO_Ref` value.

nagg is non-destructive. It will not overwrite existing files.

Limits Defined:

NAGG_Granule_info_max	20000	Maximum number of granules managed
<i>Note: this limit may now be overridden in nagg 1.6.2 at run time by setting the environment variable NAGG_MAX_GRANULES to the desired maximum. The limit may be removed entirely in a future release.</i>		
NAGG_Product_list_max	42	Maximum number of products requested
NAGG_outputfiles_max	42	Maximum number of output files

See “Appendix 2” in the *NPP Aggregation Tool Components* document for a complete list of size definitions in the nagg tool.

Other Considerations:

- nagg can de-aggregate files by setting `-n 1` and operating on aggregate files.
- nagg can aggregate files by setting `-n N` and operating on a directory of single granule files.
- nagg can re-aggregate files by setting `-n N` to a number different from the number of granules in the input files.

Command-Line Options and Parameters:

`-h, --help`
Use to display the command-line syntax and to list valid granule types and packaging groupings.

`--version`
Displays the version information.

`-t list, --type=list`
list specifies a comma separated list of NPP record type mnemonics. Unless `-s` is specified, the granule types will be packaged together. Types must be compatible to be packaged together. Use `-h` to list valid package groupings. If `-t` is not specified, `-g <product>` must be given explicitly or through `-l example file` to aggregate only the geolocation product granules.

`-n N, --number=N`
N is the number of granules of each product in each aggregate file and must be greater than zero. If neither this option nor `-l example file` is specified, the default value of 1 (single granule files) will be used.

`-g criterion, --geolocation=criterion`
criterion is the criterion used for searching the geolocation granules. Valid values for *criterion* are the following:

no | 0
 yes | 1
 strict | 2
 <product>

Aggregate product files without geolocation input or output
 Allow approximate matching of geolocation input file names (default)
 Require exact matching of geolocation input file names

- With no data product specified: aggregate geolocation product <product> only.
- With one or more data products: aggregate geolocation product <product> with data products. This is provided to allow selection of terrain corrected or not terrain corrected geolocation for products associated with a designated geolocation product that has an equivalent geolocation product (See RFC: Nagg extension for terrain corrected geolocation)

-l *example*, --like=*example*

Package like the *example* file in number of granules, type list, and geoproduct only, without geoproduct, or with unpackaged geoproduct. *example* must be an existing NPP file. Options on the command line override the number of granules, type list, and geolocation options from the *example* file.

-A *seconds*

seconds is the number of seconds in each aggregate file. The aggregation's size is a fixed number of granules per file equal to seconds divided by the period of the granule rounded up to the next integer number of granules. The formula for this is

$$\text{NumGranules/File} = \text{seconds/GranulePeriod}$$

where

NumGranules/File is the number of granules in the aggregated file,

seconds is the number of seconds that are used to determine the size of the aggregated file, and

GranulePeriod is the number of seconds worth of data held by each granule.

If, for example, *seconds* is set to 60 and GranulePeriod is 31.997 seconds per granule, then the following steps show the calculation:

Calculation	Comments
NumGranules/File = 60 seconds/31.997seconds/granule	This is the setting of this parameter divided by the granule period.
NumGranules/File = 1.875 granules	This is the actual value of the calculation.
NumGranules/File = 2 granules	This is the value of the calculation rounded up to the next integer.

- `-S, --simple`
Simple aggregates are produced. Each type is packaged separately. All granule types including geolocation products are packaged in one file.
- `-d directory, --directory=directory`
Directory where the output files should be placed. The default is the current working directory. The tool will fail if the directory does not exist.
- `-O origin`
The origin identifier is a four-character string in the output file name. Only the first four characters will be used. If fewer than four characters are specified, the origin will be pre-pended with the character "X". The default is "XXXX".
- `-D domain`
The domain identifier is a three-character string in the output file name. Only the first three characters will be used. If fewer than three characters are specified, the domain will be pre-pended with the character "X". The default is "XXX".
- `--debug`
In debug mode, all of the granules in the input files including those not specified by the `-t` list are read in, all of the granules are displayed, and the execution is ended without generating the normal output files. Note that this option is intended for tool debugging. Files or output generated should not be used for production purposes: the behavior of debug mode may change from version to version.
- `--onefile`
Produce one aggregation for each product containing all of the granules from the input files.
- `--nofill`
Do not create fill granules for missing granules, neither in an aggregation nor to match packaged products.
- `--filter=filter`
Filter Type. Use to specify a filter to be used. This option can be used more than once on a command line. Filters will be applied in the order they are listed on the command line.

The value of *filter* is a string of the following format:

list_of_objects : *name_of_filter*[=*filter_parameters*]

where

list_of_objects is a comma separated list of object names to which the filter(s) will be applied. If no object names are specified, the filter will be applied to all objects.

name_of_filter specifies the filter and can be one of the following values:

<i>name_of_filter</i>	Comments
GZIP	The HDF5 gzip filter (gzip compression)

<i>name_of_filter</i>	Comments
SZIP	The HDF5 Szip filter (Szip compression)
SHUF	The HDF5 shuffle filter
FLET	The HDF5 checksum filter
NBIT	The HDF5 N-bit filter
SOFF	The HDF5 scale/offset filter
NONE	To remove any filter(s)

filter_parameters specifies optional filter settings. Possible settings are listed below. See the HDF5 User Guide at http://www.hdfgroup.org/HDF5/doc/UG/10_Datasets.html Section 5.6 for detailed information about HDF5 built-in filters.

<i>name_of_filter</i>	<i>filter_parameter</i>	Usage	Comments
GZIP	<i>deflation_level</i>	GZIP=#	# is a number from 1-9.
SZIP	<i>pixels_per_block,coding_method</i>	SZIP=#, XX	# is an even number from 2-32, and XX is either EC or NN*.
SHUF	None		There are no parameters for this filter.
FLET	None		There are no parameters for this filter.
NBIT	None		There are no parameters for this filter.
SOFF	<i>scale_factor,scale_type</i>	SOFF=#,YY	# is an integer, and YY is either IN or DS**.
NONE	None		There are no parameters for this filter.

*Coding methods:
 EC: Entropy coding
 NN: Nearest neighbor
 **Scale types
 IN: Integer
 DS: Floating point
 D-scaling

--layout=*layout*

Layout Type. *layout* specifies the type of layout that will be used and will be a string of the following format:

list_of_objects : *layout_type*[=*layout_parameters*]

where

list_of_objects is a comma separated list of object names to which the layout will be applied. If no object names are specified, the layout will be applied to all objects.

layout_type specifies the type of layout that will be applied and can be one of the following values:

Value	Comments
CHUNK	The chunk layout
COMPA	The compact layout
CONTI	The contiguous layout

layout_parameters will be present only with the `CHUNK` layout and specifies the chunk size of each dimension in the following format with no intervening spaces:

dim_1 × *dim_2* × ... *dim_n*

Note: If no compression or storage layout property is specified, the properties of the datasets in the input files will be preserved.

`--minimum=size`

Size Threshold. Apply global filter(s) and layout(s) only to objects whose size in bytes is equal to or greater than *size*. *size* must be an integer greater than zero (0).

Default: If no size is specified, a threshold of 1024 bytes is assumed.

Thresholds are not applied for COMPACT layout nor for Datasets specified by name.

INPUT

INPUT is a list of one or more NPP files and should always be the final option on the command line.

Exit Status:

0 Succeeded.
>0 An error occurred.
