
rasmlib Documentation

Release 0.0.0

Joe Hamman

July 29, 2016

1	RASM	1
2	Purpose of rasmlib	3
3	Documentation	5

RASM

The **Regional Arctic System Model (RASM)** is a high resolution, regional, coupled atmosphere - land - sea ice - ocean model that

- Weather Research and Forecasting (WRF) atmospheric model
- Variable Infiltration Capacity (VIC) hydrology model
- RVIC streamflow routing model
- Parallel Ocean Program (POP) model
- Los Alamos Sea Ice model (CICE)

Purpose of rasmlib

rasmlib is a collection of post-processing, analysis, and plotting tools used to support the development of RASM. Many of the scripts and functions in this package were developed specifically for RASM and are likely not applicable to other projects.

3.1 Installing

3.1.1 Dependencies:

- Python 2.6 or later
- [netCDF4](#) : python/numpy interface for NetCDF4

3.1.2 analysis module dependencies:

- [numpy](#) : Fundamental package needed for scientific computing with Python.
- [scipy](#): Open-source software for mathematics, science, and engineering.
- [pandas](#) : Flexible and powerful data analysis / manipulation library for Python.
- [xray](#) : N-D labeled arrays and datasets in Python.
- matplotlib [basemap](#) : The matplotlib basemap toolkit is a library for plotting 2D data on maps in Python.
- [seaborn](#) : Statistical data visualization using matplotlib.

3.1.3 post_process module dependencies:

- [nco](#) : Command line utility that manipulates and analyzes data stored in netCDF-accessible formats, including DAP, HDF4, and HDF5.
- [nco-bindings](#) : Python bindings to nco comamnd line utility.

Before you install *rasmlib*, be sure you have the required dependencies (python and netCDF4) installed. The easiest way to do so is to use the [Anaconda](#) python distribution.

To install *rasmlib*, use *setuptools*:

```
python setup.py install
```

3.2 Post Processing

After each CESM / RASM simulation, it is necessary to do a certain amount of post processing on the output files before comprehensive analysis can be performed. This page provides a brief explanation of how a few of the RASM model components are post processed and the tools used from the *rasmlib* package.

3.2.1 Tools

rasmlib_post_process: a command line script that performs batch post processing on model output. It can be run using multiple cores using Python *multiprocessing* module. See the *Scripts* page for information on usage of this script.

3.2.2 Models

VIC

Current VIC output requires two steps of post processing. Step 1: The time variable and filename timestamp must be corrected. Step 2: Model output is concatenated to 3 levels of model output:

- monthly mean diurnal cycles: requires hourly model output timestep.
- daily timeseries: requires daily or subdaily model output timestep.
- monthly timeseries: performed for any model output timestep

RVIC, CPL, WRF

Output from these models is post processed in the same way as for the **VIC** model except there is no need to adjust the time variables or timesteps.

3.3 Analysis

Typical analysis of RASM model output is supported by *rasmlib* through the *analysis* module. This module includes functions and classes that assist with statistical analysis and plotting.

3.4 API

This page provides an auto-generated summary of *rasmlib*'s API.

3.4.1 Modules

<code>analysis</code>
<code>post_processing</code>

3.4.2 calendar

Functions:

<code>next_month</code>

Continued on next page

Table 3.2 – continued from previous page

prev_month
next_day
prev_day
leap_year
get_dpm

Attributes:

HOURSPERDAY
SECSPERHOUR
MINSPERHOUR
SECSPERMINUTE
MINSPERDAY
SECSPERDAY
MONTHSPERYEAR
dpm
seasons

3.4.3 io

Functions:

read_config
config_type
make_tarfile
get_data_files_namelist
get_variables_namelist
get_datasets
read_domain
get_time_units

3.4.4 utils

Functions:

argsort
clean_dir
clean_file
make_directories
multiple_replace
custom_strptime
custom_strftime
partition
chunks

3.5 Post Process API

This page provides an auto-generated summary of rasmlib's post processing API.

3.5.1 post_processing.share

Classes:

`Histfile`

Attributes:

`host`

`MACH_OPTS`

`NCOFORMAT`

3.5.2 post_processing.means

Functions:

`monthly_mean_diurnal_cycle`

`daily_mean_timeseries`

`cat_chunks`

`monthly_mean_timeseries`

3.5.3 post_processing.adjust_timestamp

Functions:

`adjust_timestamp`

3.6 Analysis API

This page provides an auto-generated summary of rasmlib's analysis API.

3.6.1 analysis.climatology

Functions:

`season_mean`

`annual_mean`

3.6.2 analysis.plotting

Classes:

Bmap

Functions:

make_bmap
sub_plot_pcolor
cmap_discretize
plot4_seasons

Attributes:

projections
default_map

3.7 Scripts

rasmlib includes two command line scripts:

1. **rasm_post_process**: this script runs a set of utilities to concatenate and adjust RASM output files. Its implementation is fairly generic and has been used to post process output from the VIC, RVIC, ATM, and CPL components.
 2. **rasm_analysis**: this script is currently underdevelopment and is intended to provide a very flexible interface to doing batch analysis on RASM output.
-

3.7.1 rasm_post_process

usage: `rasm_post_process [-h] [--output_preset OUTPUT_PRESET] [--processed_dir PROCESSED_DIR] [-np NUMOFPROC] config_file short_term_archive`

Generic RASM history file post processing script

positional arguments: `config_file` Input configuration file `short_term_archive` Case short term archive directory

optional arguments:

-h, --help show this help message and exit

--output_preset OUTPUT_PRESET Input configuration preset

--processed_dir PROCESSED_DIR Input configuration file (default=\$WORKDIR/processed/\$RUN/\$COMPONENT)

-np NUMOFPROC, --numofproc NUMOFPROC Number of processors used to run job

3.7.2 rasm_analysis

usage: `rasm_analysis [-h] config`

positional arguments: `config` Input Configuration File

optional arguments:

-h, --help show this help message and exit

3.8 Modules

rasmlib is organized into two modules:

- *analysis*
- *post_processing*

3.9 Command Line Utilities:

rasmlib includes two main command line utilities:

- *rasm_analysis*
- *rasm_post_process*

3.10 License

rasmlib is available under the GNU GPL v3.0 license.