## CONTENTS:

1. Documentation ... 3
2. Contributing ... 5
3. License ... 7
4. Credits ... 9
5. Indices and tables ... 139

Index ... 141
Finch Documentation, Release 0.5.3

Finch (the bird)  *Finch is a bird . . .*

A Web Processing Service for Climate Indicators.
Learn more about Finch in its official documentation at https://finch.readthedocs.io.
Submit bug reports, questions and feature requests at https://github.com/bird-house/finch/issues
You can find information about contributing in our Developer Guide.

Please use bumpversion to release a new version.
Free software: Apache Software License 2.0
This package was created with Cookiecutter and the bird-house/cookiecutter-birdhouse project template.

4.1 Installation

- **Install from Conda**
- **Install from GitHub**
- **Start Finch PyWPS service**
- **Run Finch as Docker container**
- **Use Ansible to deploy Finch on your System**

### 4.1.1 Install from Conda

**Warning:** TODO: Prepare Conda package.

### 4.1.2 Install from GitHub

Check out code from the Finch GitHub repo and start the installation:

```
$ git clone https://github.com/bird-house/finch.git
$ cd finch

Create Conda environment named `finch`:

```
$ conda env create -f environment.yml
$ source activate finch
```

Install Finch app:

```
$ pip install -e .
```

For development you can use this command:
After successful installation you can start the service using the finch command-line.

```
$ finch --help # show help
$ finch start # start service with default configuration
```

OR

```
$ finch start --daemon # start service as daemon
```

loading configuration
forked process id: 42

The deployed WPS service is by default available on:

**Note:** Remember the process ID (PID) so you can stop the service with `kill PID`.

You can find which process uses a given port using the following command (here for port 5000):

```
$ netstat -nlp | grep :5000
```

Check the log files for errors:

```
$ tail -f pywps.log
```

... or do it the lazy way

You can also use the Makefile to start and stop the service:

```
$ make start
$ make status
$ tail -f pywps.log
$ make stop
```

### 4.1.4 Run Finch as Docker container

You can also run Finch as a Docker container.

**Warning:** TODO: Describe Docker container support.
4.1.5 Use Ansible to deploy Finch on your System

Use the Ansible playbook for PyWPS to deploy Finch on your system.

4.2 Configuration

4.2.1 Command-line options

You can overwrite the default PyWPS configuration by using command-line options. See the Finch help which options are available:

```
$ finch start --help
--hostname HOSTNAME hostname in PyWPS configuration.
--port PORT port in PyWPS configuration.
```

Start service with different hostname and port:

```
$ finch start --hostname localhost --port 5001
```

4.2.2 Use a custom configuration file

You can overwrite the default PyWPS configuration by providing your own PyWPS configuration file (just modify the options you want to change). Use one of the existing sample-* .cfg files as example and copy them to etc/custom.cfg.

For example change the hostname (demo.org) and logging level:

```
$ cd finch
$ vim etc/custom.cfg
$ cat etc/custom.cfg
[server]
url = http://demo.org:5000/wps
outputurl = http://demo.org:5000/outputs

[logging]
level = DEBUG
```

Start the service with your custom configuration:

```
# start the service with this configuration
$ finch start -c etc/custom.cfg
```

4.3 Notebooks

These notebooks demonstrates a few features of the Finch server.

If you’re unfamiliar with notebooks, note that typing TAB after an object will display a drop-down menu of the object’s attributes and methods, and that you need to hit CTRL-Enter to run a cell. You can also type ? after a function or method to display the corresponding help message.
4.3.1 Finch usage

Finch is a WPS server for climate indicators, but also has a few utilities to facilitate data handling. To get started, first instantiate the client. Here, the client will try to connect to a local or remote finch instance, depending on whether the environment variable WPS_URL is defined.

```python
import os
import xarray as xr
from birdy import WPSClient

# Disable Dataset.__repr_html__ and DataArray.__repr_html__, enabled by default in xarray 0.15.1
xr.set_options(display_style='text')

pavics_url = 'https://pavics.ouranos.ca/twitcher/ows/proxy/finch/wps'
url = os.environ.get('WPS_URL', pavics_url)
verify_ssl = True if 'DISABLE_VERIFY_SSL' not in os.environ else False
wps = WPSClient(url, verify=verify_ssl)
```

The list of available processes can be displayed using the magic `?` command (`wps?`). Similarly, help about any individual process is available using `?` or the `help` command.

```python
help(wps.frost_days)
```

Help on method frost_days in module birdy.client.base:

frost_days(tasmin=None, freq='YS') method of birdy.client.base.WPSClient instance

Number of days where daily minimum temperatures are below 0.

Parameters
----------

  NetCDF Files or archive (tar/zip) containing netCDF files.
- freq: string
  Resampling frequency

Returns
-------

- output_netcdf: ComplexData:mimetype:`application/x-netcdf`
  The indicator values computed on the original input grid.
- output_log: ComplexData:mimetype:`text/plain`
  Collected logs during process run.
- ref: ComplexData:mimetype:`application/metalink+xml; version=4.0`
  Metalink file storing all references to output files.

To actually compute an indicator, we need to specify the path to the netCDF file used as input for the calculation of the indicator. To compute `frost_days`, we need a time series of daily minimum temperature. Here we’ll use a small test file. Note that here we’re using an OPeNDAP link, but it could also be an url to a netCDF file, or the path to a local file on disk. We then simply call the indicator. The response is an object that can poll the server to inquire about the status of the process. This object can use two modes: - synchronous: it will wait for the server’s response before returning; or - asynchronous: it will return immediately, but without the actual output from the process.

Here, since we’re applying the process on a small test file, we’re using the default synchronous mode. For long computations, use the asynchronous mode to avoid time-out errors. The asynchronous mode is activated by setting the `progress` attribute of the WPS client to True.
The `get` method returns a `NamedTuple` object with all the WPS outputs, either as references to files or actual content. To copy the file to the local disk, you can use the `getOutput` method, but birdy’s `get` function has a more user-friendly solution. Setting the `asobj` argument to `True` will directly download all the output files and return outputs as Python objects.
4.3.2 Computing indicators on DAP subsets

In a typical programming environment, the standard way to compute an indicator for a given location would be to extract the time series at the given location, then run the computation on this subset. When interacting with a remote server, things are a bit more complicated. One option would be to first call a subsetting process to extract the data at the desired location, then run the climate indicator process on that subsetted file. The other option showcased here is to pass a DAP url that encodes the subsetting operation.

This tutorial shows how to get the index for the desired location and pass them as a DAP link to a Finch indicator process.

```python
[1]: import os
import xarray as xr
from birdy import WPSClient

# Disable Dataset.__repr_html_ and DataArray._repr_html_, enabled by default in xarray 0.15.1
xr.set_options(display_style='text')

# Link to file storing precipitation
pr = "https://pavics.ouranos.ca/twitcher/ows/proxy/thredds/dodsC/birdhouse/testdata/flyingpigeon/cmip3/pr.sresa2.miub.echo_g.run1.atm.da.nc"

# Open connection to Finch WPS server
pavics_url = 'https://pavics.ouranos.ca/twitcher/ows/proxy/finch/wps'
url = os.environ.get('WPS_URL', pavics_url)
wps = WPSClient(url)

[2]: # Open remote dataset and extract location indices
ds = xr.open_dataset(pr)
ds

[2]: <xarray.Dataset>
Dimensions:   (bnds: 2, lat: 6, lon: 7, time: 7200)
Coordinates:
* lat       (lat) float64  42.68  46.39  50.1  53.81  57.52  61.23
* lon       (lon) float64  281.2  285.0  288.8  292.5  296.2  300.0  303.8
* time      (time) object  2046-01-01 12:00:00 ... 2065-12-30 12:00:00
Dimensions without coordinates: bnds
Data variables:
 lat_bnds   (lat, bnds) float64 ...
 lon_bnds   (lon, bnds) float64 ...
 time_bnds  (time, bnds) object ...
 pr          (time, lat, lon) float32 ...
Attributes:
 comment: Spinup: restart files from end of experiment 20C3M (corre...
title: MIUB model output prepared for IPCC Fourth Assessment SR...
cmor_version: 0.96
institution: MIUB (University of Bonn, Germany)
source: ECHO-G(1999): atmosphere: ECHAM4 (T30L19) with partial se...
contact: Stephanie Legutke (legutke@dkrz.de), Seung-Ki Min(s...
references: ECHAM4: E. Roecker et al., 1996, The atmospheric general...
experiment_id: SRES A2 experiment
realization: 1
directory: /ipcc/sresa2/atm/da/
table_id: Table A2 (17 November 2004)
calendar: 360_day
project_id: IPCC Fourth Assessment
```
(continues on next page)
Conventions: CF-1.0
id: pcmdi.ipcc4.miub_echo_g.sresa2.run1.atm.da
history: Mon Aug 1 11:42:37 2011: ncks -4 -L 7 -d lat,42.0,64.0 ...
NCO: 4.0.9

# Use the `remap_label_indexers` function to convert coordinates to positional indexes.
```python
import datetime as dt
coords = dict(lat=45, lon=290)
index, _ = xr.core.coordinates.remap_label_indexers(ds, coords, method="nearest")
```

# The `nearest` method cannot be used with slices, so we do another call for the time period.
```python
ti, _ = xr.core.coordinates.remap_label_indexers(ds, dict(time=slice("2060-01-01", "2064-12-30")))
```

# Merge the spatial and temporal indices
```python
index.update(ti)
```

```python
{'lat': 1, 'lon': 2, 'time': slice(5040, 6840, None)}
```

Subsetting URLs

The subset syntax consists in a ? followed by comma separated list of variable names, each followed by a slice [start, step, stop] for each dimension. So for example, to get the very first time step of the precipitation time series over the entire grid, we’d write

```
<url>?pr[0:1:0][0:1:5][0:1:6]
```

Note that this uses a 0-based indexing system, so [0:1:1] is a slice including the first and second elements.

```python
xr.open_dataset(pr+"?pr[0:1:0][0:1:5][0:1:6]"
```

<array.Dataset>
Dimensions: (lat: 6, lon: 7, time: 1)
Dimensions without coordinates: lat, lon, time
Data variables:
  pr (time, lat, lon) float32 ...
Attributes:
  comment: Spinup: restart files from end of experiment 20C3M (corre...
  title: MIUB model output prepared for IPCC Fourth Assessment SR...
  cmor_version: 0.96
  institution: MIUB (University of Bonn, Bonn, Germany)
  source: ECHO-G(1999): atmosphere: ECHAM4 (T30L19) with partial se...
  contact: Stephanie Legutke (legutke@dkrz.de), Seung-Ki Min(skmin@u...
  references: ECHAM4: E. Roeckner et al., 1996, The atmospheric general...
  experiment_id: SRES A2 experiment
  realization: 1
directory: /ipcc/sresa2/atm/da/
table_id: Table A2 (17 November 2004)
calendar: 360_day
project_id: IPCC Fourth Assessment
Conventions: CF-1.0
id: pcmdi.ipcc4.miub_echo_g.sresa2.run1.atm.da
history: Mon Aug 1 11:42:37 2011: ncks -4 -L 7 -d lat,42.0,64.0 ...
NCO: 4.0.9

4.3. Notebooks 15
Note that the returned array has no time, lat or lon variables. We only requested the pr variable, not these other coordinate variables. To remedy the situation, we add these coordinate variables to the request.

```python
[5]: xr.open_dataset(pr+'?pr[0:1:0][0:1:5][0:1:6],time[0:1:0],lat,lon")

[5]: <xarray.Dataset>
Dimensions: (lat: 6, lon: 7, time: 1)
Coordinates:
  * lat (lat) float64 42.68 46.39 50.1 53.81 57.52 61.23
  * lon (lon) float64 281.2 285.0 288.8 292.5 296.2 300.0 303.8
  * time (time) object 2046-01-01 12:00:00
Data variables:
  pr (time, lat, lon) float32 ...
Attributes:
  comment: Spinup: restart files from end of experiment 20C3M (corre...
  title: MIUB model output prepared for IPCC Fourth Assessment SR...
  cmor_version: 0.96
  institution: MIUB (University of Bonn, Bonn, Germany)
  source: ECHO-G(1999): atmosphere: ECHAM4 (T30L19) with partial se...
  contact: Stephanie Legutke (legutke@dkrz.de), Seung-Ki Min(skmin@u...
  references: ECHAM4: E. Roeckner et al., 1996, The atmospheric general...
  experiment_id: SRES A2 experiment
  realization: 1
  directory: /ipcc/sresa2/atm/da/
  table_id: Table A2 (17 November 2004)
  calendar: 360_day
  Conventions: CF-1.0
  id: pcmdi.ipcc4.miub_echo_g.sresa2.run1.atm.da
  history: Mon Aug 1 11:42:37 2011: ncks -4 -L 7 -d lat,42.0,64.0 -d ...
  NCO: 4.0.9
```

Now let’s go back to our original index and convert it into a DAP subset URL.

```python
[6]: xr.open_dataset(pr+'?pr[5040:1:6840][1:1:1][2:1:2],lat[1:1:1],lon[2:1:2],time[5040:1:6839])

[6]: <xarray.Dataset>
Dimensions: (lat: 1, lon: 1, time: 1800, time_1: 1801)
Coordinates:
  * lat (lat) float64 46.39
  * lon (lon) float64 288.8
  * time (time) object 2060-01-01 12:00:00 ... 2064-12-30 12:00:00
Dimensions without coordinates: time_1
Data variables:
  pr (time_1, lat, lon) float32 ...
Attributes:
  comment: Spinup: restart files from end of experiment 20C3M (corre...
  title: MIUB model output prepared for IPCC Fourth Assessment SR...
  cmor_version: 0.96
  institution: MIUB (University of Bonn, Bonn, Germany)
  source: ECHO-G(1999): atmosphere: ECHAM4 (T30L19) with partial se...
  contact: Stephanie Legutke (legutke@dkrz.de), Seung-Ki Min(skmin@u...
  references: ECHAM4: E. Roeckner et al., 1996, The atmospheric general...
  experiment_id: SRES A2 experiment
  realization: 1
  directory: /ipcc/sresa2/atm/da/
  table_id: Table A2 (17 November 2004)
  calendar: 360_day
```

(continues on next page)
```python
[7]: index
[7]: {'lat': 1, 'lon': 2, 'time': slice(5040, 6840, None)}

[8]: def dap_slice(index):
    """Convert python index dictionary into DAP subset index dictionary."""
    dap = {}
    for key, val in index.items():
        if isinstance(val, slice):
            dap[key] = f"[{val.start}::{val.step or 1}:{val.stop - 1}]"
        elif isinstance(val, int):
            dap[key] = f"[{val}:{val}]"
    return dap

def dap_subset(da, index):
    """Return DAP subset URL."""
    s = dap_slice(index)
    vs = [da, ] + list(da.coords.values())
    url = "?" + ",".join({x.name + ".join([s[dim] for dim in x.dims]) for x in vs})
    return url

sub = dap_subset(ds.pr, index)
print(sub)


[9]: xr.open_dataset(pr + sub)

<xarray.Dataset>
  Dimensions: (lat: 1, lon: 1, time: 1800)
  Coordinates:
  * lat (lat) float64 46.39
  * lon (lon) float64 288.8
  * time (time) object 2060-01-01 12:00:00 ... 2064-12-30 12:00:00
  Data variables:
  pr (time, lat, lon) float32 ...
  Attributes:
  comment:  Spinup: restart files from end of experiment 20C3M (corre... 
  title:  MIUB model output prepared for IPCC Fourth Assessment SR...
  cmor_version: 0.96
  institution:  MIUB (University of Bonn, Bonn, Germany)
  source:  ECHO-G(1999): atmosphere: ECHAM4 (T30L19) with partial se...
  contact:  Stephanie Legutke (legutke@dkrz.de), Seung-Ki Min(skmin@u...
  references:  ECHAM4: E. Roeckner et al., 1996, The atmospheric general...
  experiment_id:  SRES A2 experiment
  realization:  1
  directory:  /ipcc/sresa2/atm/da/
  table_id:  Table A2 (17 November 2004)
  calendar:  360_day
```

(continues on next page)

4.3. Notebooks  17
Using the subset URL in a WPS process

Now this subset url can be used as a normal netCDF link in WPS processes. Here, let’s compute the average precipitation during wet days (sdii) over our subset. As expected, the output is only computed for the five years we requested on a single grid point closest to the coordinates we chose.

```python
[10]: resp = wps.sdii(pr + sub)
    out = resp.get(asobj=True)
    out.output_netcdf.sdii
```

```python
[10]: <xarray.DataArray 'sdii' (time: 5, lat: 1, lon: 1)>
    array([[[4.984402]],
           [[5.412616]],
           [[5.488322]],
           [[5.164405]],
           [[4.69054 ]]]
    Coordinates:
    * time (time) object 2060-01-01 00:00:00 ... 2064-01-01 00:00:00
    * lat (lat) float64 46.39
    * lon (lon) float64 288.8
    Attributes:
    units: mm/day
    cell_methods: time: mean (interval: 30 minutes)
    history: pr=max(0,pr) applied to raw data;\n        [2020-04-05 10:27:31] ...
    standard_name: lwe_thickness_of_precipitation_amount
    long_name: Average precipitation during wet days (sdii)
    description: Annual simple daily intensity index (sdii) : annual avera...
```

4.4 Developer Guide

- Building the docs
- Running tests
- Run tests the lazy way
- Prepare a release
- Bump a new version
Warning: To create new processes look at examples in Emu.

4.4.1 Building the docs

First install dependencies for the documentation:

```
$ make develop
```

Run the Sphinx docs generator:

```
$ make docs
```

4.4.2 Running tests

Run tests using pytest.

First activate the finch Conda environment and install pytest.

```
$ source activate finch
$ pip install -r requirements_dev.txt  # if not already installed
OR
$ make develop
```

Run quick tests (skip slow and online):

```
$ pytest -m 'not slow and not online'
```

Run all tests:

```
$ pytest
```

Check pep8:

```
$ flake8
```

4.4.3 Run tests the lazy way

Do the same as above using the Makefile.

```
$ make test
$ make test-all
$ make lint
```
4.4.4 Prepare a release

Update the Conda specification file to build identical environments on a specific OS.

**Note:** You should run this on your target OS, in our case Linux.

```
$ conda env create -f environment.yml
$ source activate finch
$ make clean
$ make install
$ conda list -n finch --explicit > spec-file.txt
```

4.4.5 Bump a new version

Make a new version of Finch in the following steps:

- Make sure everything is commit to GitHub.
- Update CHANGES.rst with the next version.
- **Dry Run:** `bumpversion --dry-run --verbose --new-version 0.8.1 patch`
- **Do it:** `bumpversion --new-version 0.8.1 patch`
- ... or `bumpversion --new-version 0.9.0 minor`
- Push it: `git push`
- **Push tag:** `git push --tags`

See the bumpversion documentation for details.

4.5 Processes

- *xclim Indicators*
- *Ensemble Processes*
- *Other Processes*

4.5.1 xclim Indicators

```python
class finch.processes.xclim.cdd_Indicator_Process
cdd unidecode.unidecode (v0.1)
```

**Parameters**

- **pr** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
- **thresh** (*string, optional*) – Threshold
- **freq** (*string, optional*) – Resampling frequency
Returns

- **output_netcdf** (*application/x-netcdf*) – The indicator values computed on the original input grid.
- **output_log** (*text/plain*) – Collected logs during process run.
- **ref** (*application/metalink+xml; version=4.0*) – Metalink file storing all references to output files.

```python
class finch.processes.xclim.cold_spell_days_Indicator_Process
cold_spell_days unidecode.unidecode (v0.1)
```

**Parameters**

- **tas** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
- **thresh** (*string, optional*) – Threshold
- **window** (*integer, optional*) – Window
- **freq** (*string, optional*) – Resampling frequency

Returns

- **output_netcdf** (*application/x-netcdf*) – The indicator values computed on the original input grid.
- **output_log** (*text/plain*) – Collected logs during process run.
- **ref** (*application/metalink+xml; version=4.0*) – Metalink file storing all references to output files.

```python
class finch.processes.xclim.cold_spell_duration_index_Indicator_Process
cold_spell_duration_index unidecode.unidecode (v0.1)
```

**Parameters**

- **tasmin** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
- **tn10** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
- **window** (*integer, optional*) – Window
- **freq** (*string, optional*) – Resampling frequency

Returns

- **output_netcdf** (*application/x-netcdf*) – The indicator values computed on the original input grid.
- **output_log** (*text/plain*) – Collected logs during process run.
- **ref** (*application/metalink+xml; version=4.0*) – Metalink file storing all references to output files.

```python
class finch.processes.xclim.consecutive_frost_days_Indicator_Process
consecutive_frost_days unidecode.unidecode (v0.1)
```

**Parameters**

- **tasmin** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
• **freq** *(string, optional)* – Resampling frequency

Returns

• **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.

• **output_log** *(text/plain)* – Collected logs during process run.

• **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

```python
class finch.processes.xclim.cooling_degree_days_Indicator_Process
cooling_degree_days unidecode.unidecode (v0.1)
```

Parameters

• **tas** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*

• **thresh** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

Returns

• **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.

• **output_log** *(text/plain)* – Collected logs during process run.

• **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

```python
class finch.processes.xclim.cwd_Indicator_Process
cwd unidecode.unidecode (v0.1)
```

Parameters

• **pr** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*

• **thresh** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

Returns

• **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.

• **output_log** *(text/plain)* – Collected logs during process run.

• **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

```python
class finch.processes.xclim.dlyfrzthw_Indicator_Process
dlyfrzthw unidecode.unidecode (v0.1)
```

Parameters

• **tasmax** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*

• **tasmin** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
• `thresh_tasmax` *(string, optional)* – Threshold
• `thresh_tasmin` *(string, optional)* – Threshold
• `freq` *(string, optional)* – Resampling frequency

Returns

• `output_netcdf` *(application/x-netcdf)* – The indicator values computed on the original input grid.
• `output_log` *(text/plain)* – Collected logs during process run.
• `ref` *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

class `finch.processes.xclim.dtr_Indicator_Process`

dtr unidecode.unidecode (v0.1)

Parameters

• `tasmax` *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
• `tasmin` *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
• `freq` *(string, optional)* – Resampling frequency

Returns

• `output_netcdf` *(application/x-netcdf)* – The indicator values computed on the original input grid.
• `output_log` *(text/plain)* – Collected logs during process run.
• `ref` *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

class `finch.processes.xclim.dtrvar_Indicator_Process`
dtrvar unidecode.unidecode (v0.1)

Parameters

• `tasmax` *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
• `tasmin` *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
• `freq` *(string, optional)* – Resampling frequency

Returns

• `output_netcdf` *(application/x-netcdf)* – The indicator values computed on the original input grid.
• `output_log` *(text/plain)* – Collected logs during process run.
• `ref` *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

class `finch.processes.xclim.etr_Indicator_Process`
etr unidecode.unidecode (v0.1)

Parameters
• \texttt{tasmax} (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• \texttt{tasmin} (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• \texttt{freq(string, optional)} – Resampling frequency

Returns

• \texttt{output_netcdf} (application/x-netcdf) – The indicator values computed on the original input grid.

• \texttt{output_log(text/plain)} – Collected logs during process run.

• \texttt{ref(application/metalink+xml; version=4.0)} – Metalink file storing all references to output files.

\texttt{class} \texttt{finch.processes.xclim.\texttt{freshet\_start\_Indicator\_Process}}

\texttt{freshet\_start} unidecode.unidecode (v0.1)

Parameters

• \texttt{tas(application/x-netcdf, application/x-ogc-dods)} – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• \texttt{thresh(string, optional)} – Threshold

• \texttt{window(integer, optional)} – Window

• \texttt{freq(string, optional)} – Resampling frequency

Returns

• \texttt{output_netcdf} (application/x-netcdf) – The indicator values computed on the original input grid.

• \texttt{output_log(text/plain)} – Collected logs during process run.

• \texttt{ref(application/metalink+xml; version=4.0)} – Metalink file storing all references to output files.

\texttt{class} \texttt{finch.processes.xclim.\texttt{frost\_days\_Indicator\_Process}}

\texttt{frost\_days} unidecode.unidecode (v0.1)

Parameters

• \texttt{tasmin} (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• \texttt{freq(string, optional)} – Resampling frequency

Returns

• \texttt{output_netcdf} (application/x-netcdf) – The indicator values computed on the original input grid.

• \texttt{output_log(text/plain)} – Collected logs during process run.

• \texttt{ref(application/metalink+xml; version=4.0)} – Metalink file storing all references to output files.

\texttt{class} \texttt{finch.processes.xclim.\texttt{growing\_degree\_days\_Indicator\_Process}}

\texttt{growing\_degree\_days} unidecode.unidecode (v0.1)

Parameters
• **tas**(application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **thresh**(string, optional) – Threshold

• **freq**(string, optional) – Resampling frequency

**Returns**

• **output_netcdf**(application/x-netcdf) – The indicator values computed on the original input grid.

• **output_log**(text/plain) – Collected logs during process run.

• **ref**(application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

```python
class finch.processes.xclim.heat_wave_frequency_Indicator_Process
heat_wave_frequency unidecode.unidecode (v0.1)
```

**Parameters**

• **tasmin**(application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **tasmax**(application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **thresh_tasmin**(string, optional) – Threshold

• **thresh_tasmax**(string, optional) – Threshold

• **window**(integer, optional) – Window

• **freq**(string, optional) – Resampling frequency

**Returns**

• **output_netcdf**(application/x-netcdf) – The indicator values computed on the original input grid.

• **output_log**(text/plain) – Collected logs during process run.

• **ref**(application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

```python
class finch.processes.xclim.heat_wave_index_Indicator_Process
heat_wave_index unidecode.unidecode (v0.1)
```

**Parameters**

• **tasmax**(application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **thresh**(string, optional) – Threshold

• **window**(integer, optional) – Window

• **freq**(string, optional) – Resampling frequency

**Returns**

• **output_netcdf**(application/x-netcdf) – The indicator values computed on the original input grid.

• **output_log**(text/plain) – Collected logs during process run.
• ref (application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.heat_wave_max_length_Indicator_Process

heat_wave_max_length unidecode.unidecode (v0.1)

Parameters

• tasmin (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• tasmax (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• thresh_tasmin (string, optional) – Threshold
• thresh_tasmax (string, optional) – Threshold
• window (integer, optional) – Window
• freq (string, optional) – Resampling frequency

Returns

• output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.
• output_log (text/plain) – Collected logs during process run.
• ref (application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.heat_wave_total_length_Indicator_Process

heat_wave_total_length unidecode.unidecode (v0.1)

Parameters

• tasmin (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• tasmax (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• thresh_tasmin (string, optional) – Threshold
• thresh_tasmax (string, optional) – Threshold
• window (integer, optional) – Window
• freq (string, optional) – Resampling frequency

Returns

• output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.
• output_log (text/plain) – Collected logs during process run.
• ref (application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.heating_degree_days_Indicator_Process

heating_degree_days unidecode.unidecode (v0.1)

Parameters
• **tas** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*

**thresh** *(string, optional)* – Threshold

**freq** *(string, optional)* – Resampling frequency

**Returns**

• **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.

• **output_log** *(text/plain)* – Collected logs during process run.

• **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

```python
class finch.processes.xclim.ice_days_Indicator_Process

```  

**Parameters**

• **tasmax** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*

• **freq** *(string, optional)* – Resampling frequency

**Returns**

• **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.

• **output_log** *(text/plain)* – Collected logs during process run.

• **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

```python
class finch.processes.xclim.liquidprcptot_Indicator_Process

```  

**Parameters**

• **pr** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*

• **tas** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*

• **freq** *(string, optional)* – Resampling frequency

**Returns**

• **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.

• **output_log** *(text/plain)* – Collected logs during process run.

• **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

```python
class finch.processes.xclim.max_n_day_precipitation_amount_Indicator_Process

```  

**Parameters**

• **pr** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
• window (integer, optional) – Window
• freq (string, optional) – Resampling frequency

Returns
• output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.
• output_log (text/plain) – Collected logs during process run.
• ref (application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.prcptot_Indicator_Process
prcptot unidecode.unidecode (v0.1)

Parameters
• pr (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• tas (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• freq (string, optional) – Resampling frequency

Returns
• output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.
• output_log (text/plain) – Collected logs during process run.
• ref (application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.rain_frzgr_Indicator_Process
rain_frzgr unidecode.unidecode (v0.1)

Parameters
• pr (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• tas (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• thresh (string, optional) – Threshold
• freq (string, optional) – Resampling frequency

Returns
• output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.
• output_log (text/plain) – Collected logs during process run.
• ref (application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.rx1day_Indicator_Process
rx1day unidecode.unidecode (v0.1)

Parameters
• **pr** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **freq**(*string, optional*) – Resampling frequency

**Returns**

• **output_netcdf** (*application/x-netcdf*) – The indicator values computed on the original input grid.

• **output_log** (*text/plain*) – Collected logs during process run.

• **ref** (*application/metalink+xml; version=4.0*) – Metalink file storing all references to output files.

```python
class finch.processes.xclim.sdii_Indicator_Process
    sdii unidecode.unidecode (v0.1)

Parameters

• **pr** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **thresh** (*string, optional*) – Threshold

• **freq** (*string, optional*) – Resampling frequency

**Returns**

• **output_netcdf** (*application/x-netcdf*) – The indicator values computed on the original input grid.

• **output_log** (*text/plain*) – Collected logs during process run.

• **ref** (*application/metalink+xml; version=4.0*) – Metalink file storing all references to output files.

```python
class finch.processes.xclim.solidprcptot_Indicator_Process
    solidprcptot unidecode.unidecode (v0.1)

Parameters

• **pr** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **tas** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **freq** (*string, optional*) – Resampling frequency

**Returns**

• **output_netcdf** (*application/x-netcdf*) – The indicator values computed on the original input grid.

• **output_log** (*text/plain*) – Collected logs during process run.

• **ref** (*application/metalink+xml; version=4.0*) – Metalink file storing all references to output files.

```python
class finch.processes.xclim.tg10p_Indicator_Process
    tg10p unidecode.unidecode (v0.1)

Parameters

• **tas** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• **t10** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **freq** (*string, optional*) – Resampling frequency

Returns

• **output_netcdf** (*application/x-netcdf*) – The indicator values computed on the original input grid.

• **output_log** (*text/plain*) – Collected logs during process run.

• **ref** (*application/metalink+xml; version=4.0*) – Metalink file storing all references to output files.

**class** finch.processes.xclim.tg90p_Indicator_Process

tg90p unidecode.unidecode (v0.1)

Parameters

• **tas** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **t90** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **freq** (*string, optional*) – Resampling frequency

Returns

• **output_netcdf** (*application/x-netcdf*) – The indicator values computed on the original input grid.

• **output_log** (*text/plain*) – Collected logs during process run.

• **ref** (*application/metalink+xml; version=4.0*) – Metalink file storing all references to output files.

**class** finch.processes.xclim.tg_Indicator_Process

tg unidecode.unidecode (v0.1)

Parameters

• **tasmin** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **tasmax** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

Returns

• **output_netcdf** (*application/x-netcdf*) – The indicator values computed on the original input grid.

• **output_log** (*text/plain*) – Collected logs during process run.

• **ref** (*application/metalink+xml; version=4.0*) – Metalink file storing all references to output files.

**class** finch.processes.xclim.tg_mean_Indicator_Process

tg_mean unidecode.unidecode (v0.1)

Parameters

• **tas** (*application/x-netcdf, application/x-ogc-dods*) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
• **freq**(string, optional) – Resampling frequency

Returns

• **output_netcdf**(application/x-netcdf) – The indicator values computed on the original input grid.

• **output_log**(text/plain) – Collected logs during process run.

• **ref**(application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.tn10p_Indicator_Process
tn10p unidecode.unidecode (v0.1)

Parameters

• **tasmin**(application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **t10**(application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **freq**(string, optional) – Resampling frequency

Returns

• **output_netcdf**(application/x-netcdf) – The indicator values computed on the original input grid.

• **output_log**(text/plain) – Collected logs during process run.

• **ref**(application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.tn90p_Indicator_Process
tn90p unidecode.unidecode (v0.1)

Parameters

• **tasmin**(application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **t90**(application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **freq**(string, optional) – Resampling frequency

Returns

• **output_netcdf**(application/x-netcdf) – The indicator values computed on the original input grid.

• **output_log**(text/plain) – Collected logs during process run.

• **ref**(application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.tn_days_below_Indicator_Process
tn_days_below unidecode.unidecode (v0.1)

Parameters

• **tasmin**(application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• **thresh**(string, optional) – Threshold

4.5. Processes
• freq(string, optional) – Resampling frequency

Returns

• output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.

• output_log (text/plain) – Collected logs during process run.

• ref(application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class Finch Documentation, Release 0.5.3

finch.processes.xclim.tn_max_Indicator_Process

tn_max unidecode.unidecode (v0.1)

Parameters

• tasmin (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• freq(string, optional) – Resampling frequency

Returns

• output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.

• output_log (text/plain) – Collected logs during process run.

• ref(application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class Finch Documentation, Release 0.5.3

finch.processes.xclim.tn_mean_Indicator_Process

tn_mean unidecode.unidecode (v0.1)

Parameters

• tasmin (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• freq(string, optional) – Resampling frequency

Returns

• output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.

• output_log (text/plain) – Collected logs during process run.

• ref(application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class Finch Documentation, Release 0.5.3

finch.processes.xclim.tn_min_Indicator_Process

tn_min unidecode.unidecode (v0.1)

Parameters

• tasmin (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)

• freq(string, optional) – Resampling frequency

Returns

• output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.
- **output_log** *(text/plain)* – Collected logs during process run.
- **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

```python
class finch.processes.xclim.tropical_nights_Indicator_Process
tropical_nights unidecode.unidecode (v0.1)

Parameters

- **tasmin** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
- **thresh** *(string, optional)* – Threshold
- **freq** *(string, optional)* – Resampling frequency

Returns

- **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.
- **output_log** *(text/plain)* – Collected logs during process run.
- **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.
```

```python
class finch.processes.xclim.tx10p_Indicator_Process
tx10p unidecode.unidecode (v0.1)

Parameters

- **tasmax** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
- **t10** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
- **freq** *(string, optional)* – Resampling frequency

Returns

- **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.
- **output_log** *(text/plain)* – Collected logs during process run.
- **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.
```

```python
class finch.processes.xclim.tx90p_Indicator_Process
tx90p unidecode.unidecode (v0.1)

Parameters

- **tasmax** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
- **t90** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
- **freq** *(string, optional)* – Resampling frequency

Returns

- **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.
• **output** _log_ *(text/plain)* – Collected logs during process run.

• **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.

```python
class finch.processes.xclim.tx_days_above_Indicator_Process
tx_days_above unidecode.unidecode (v0.1)

    Parameters

    • **tasmax** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*
    
    • **thresh** *(string, optional)* – Threshold
    
    • **freq** *(string, optional)* – Resampling frequency

    Returns

    • **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.

    • **output_log** *(text/plain)* – Collected logs during process run.

    • **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.
```

```python
class finch.processes.xclim.tx_max_Indicator_Process
tx_max unidecode.unidecode (v0.1)

    Parameters

    • **tasmax** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*

    • **freq** *(string, optional)* – Resampling frequency

    Returns

    • **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.

    • **output_log** *(text/plain)* – Collected logs during process run.

    • **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.
```

```python
class finch.processes.xclim.tx_mean_Indicator_Process
tx_mean unidecode.unidecode (v0.1)

    Parameters

    • **tasmax** *(application/x-netcdf, application/x-ogc-dods)* – NetCDF Files or archive (tar/zip) containing netCDF files. *(Info)*

    • **freq** *(string, optional)* – Resampling frequency

    Returns

    • **output_netcdf** *(application/x-netcdf)* – The indicator values computed on the original input grid.

    • **output_log** *(text/plain)* – Collected logs during process run.

    • **ref** *(application/metalink+xml; version=4.0)* – Metalink file storing all references to output files.
```
class finch.processes.xclim.tx_min_Indicator_Process

   tx_min unidecode.unidecode (v0.1)

   Parameters

   • tasmx (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
   • freq (string, optional) – Resampling frequency

   Returns

   • output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.
   • output_log (text/plain) – Collected logs during process run.
   • ref (application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.tx_tn_days_above_Indicator_Process

   tx_tn_days_above unidecode.unidecode (v0.1)

   Parameters

   • tasmin (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
   • tasmax (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
   • thresh_tasmin (string, optional) – Threshold
   • thresh_tasmax (string, optional) – Threshold
   • freq (string, optional) – Resampling frequency

   Returns

   • output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.
   • output_log (text/plain) – Collected logs during process run.
   • ref (application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.wetdays_Indicator_Process

   wetdays unidecode.unidecode (v0.1)

   Parameters

   • pr (application/x-netcdf, application/x-ogc-dods) – NetCDF Files or archive (tar/zip) containing netCDF files. (Info)
   • thresh (string, optional) – Threshold
   • freq (string, optional) – Resampling frequency

   Returns

   • output_netcdf (application/x-netcdf) – The indicator values computed on the original input grid.
   • output_log (text/plain) – Collected logs during process run.
• ref(application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

4.5.2 Ensemble Processes

class finch.processes.xclim.cdd_Ensemble_Bbox_Process
    ensemble_bbox_cdd unidecode.unidecode (v0.1)
    Parameters
    • lat0(float) – Minimum latitude.
    • lat1(float) – Maximum latitude.
    • lon0(float) – Minimum longitude.
    • lon1(float) – Maximum longitude.
    • start_date(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
    • end_date(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
    • ensemble_percentiles(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
    • dataset_name({"bccaqv2"}, optional) – Name of the dataset from which to get netcdf files for inputs.
    • rcp({"rcp26", "rcp45", "rcp85"}) – Representative Concentration Pathway (RCP)
    • models({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
    • thresh(string, optional) – Threshold
    • freq(string, optional) – Resampling frequency
    • output_format({'netcdf', 'csv'}, optional) – Choose in which format you want to recieve the result

Returns
    • output(application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
    • output_log(text/plain) – Collected logs during process run.

class finch.processes.xclim.cdd_Ensemble_GridPoint_Process
    ensemble_grid_point_cdd unidecode.unidecode (v0.1)
    Parameters
    • lat(string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
    • lon(string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (*{'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** (*string, optional*) – Threshold

• **freq** (*string, optional*) – Resampling frequency

• **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to receive the result

Returns

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

```python
class finch.processes.xclim.cdd_Ensemble_Polygon_Process
ensemble_polygon_cdd unidecode.unidecode (v0.1)
```

Parameters

• **shape** (*application/vnd.geo+json*) – Polygon contour, as a geojson string.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (*{'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• **thresh** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *(('netcdf', 'csv'), optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.cold_spell_days_Ensemble_Bbox_Process
ensemble_bbox_cold_spell_days unidecode.unidecode (v0.1)
```

**Parameters**

• **lat0** *(float)* – Minimum latitude.

• **lat1** *(float)* – Maximum latitude.

• **lon0** *(float)* – Minimum longitude.

• **lon1** *(float)* – Maximum longitude.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *(('bccaqv2'), optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *(('rcp26', 'rcp45', 'rcp85'))* – Representative Concentration Pathway (RCP)

• **models** *(('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...), optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** *(string, optional)* – Threshold

• **window** *(integer, optional)* – Window

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *(('netcdf', 'csv'), optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.
class finch.processes.xclim.cold_spell_days_English_GridPoint_Process
ensemble_grid_point_cold_spell_days unidecode.unidecode (v0.1)

Parameters

- **lat (string)** – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon (string)** – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name ({'bccaqv2'}, optional)** – Name of the dataset from which to get netcdf files for inputs.
- **rcp ({'rcp26', 'rcp45', 'rcp85'})** – Representative Concentration Pathway (RCP)
- **models ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional)** – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh (string, optional)** – Threshold
- **window (integer, optional)** – Window
- **freq (string, optional)** – Resampling frequency
- **output_format ({'netcdf', 'csv'}, optional)** – Choose in which format you want to recieve the result

Returns

- **output (application/x-netcdf, application/zip)** – The format depends on the output_format input parameter.
- **output_log (text/plain)** – Collected logs during process run.

class finch.processes.xclim.cold_spell_days_English_Polygon_Process
ensemble_polygon_cold_spell_days unidecode.unidecode (v0.1)

Parameters

- **shape (application/vnd.geo+json)** – Polygon contour, as a geojson string.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name (bccaqv2, optional)** – Name of the dataset from which to get netcdf files for inputs.

• **rcp (rcp26, rcp45, rcp85)** – Representative Concentration Pathway (RCP)

• **models (24MODELS, PCIC12, BNU-ESM, CCSM4, CESM1-CAM5, CNRM-CM5, CSIRO-Mk3-6-0, CanESM2, FGOALS-g2, GFDL-CM3, ..), optional)** – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh (string, optional)** – Threshold

• **window (integer, optional)** – Window

• **freq (string, optional)** – Resampling frequency

• **output_format (netcdf, csv, optional)** – Choose in which format you want to recieve the result

**Returns**

• **output (application/x-netcdf, application/zip)** – The format depends on the ‘output_format’ input parameter.

• **output_log (text/plain)** – Collected logs during process run.

**class finch.processes.xclim.cold_spell_duration_index.Ensemble_Bbox_Process**

**ensemble_bbox_cold_spell_duration_index**

**unidecode.unidecode (v0.1)**

**Parameters**

• **lat0 (float)** – Minimum latitude.

• **lat1 (float)** – Maximum latitude.

• **lon0 (float)** – Minimum longitude.

• **lon1 (float)** – Maximum longitude.

• **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name (bccaqv2, optional)** – Name of the dataset from which to get netcdf files for inputs.

• **rcp (rcp26, rcp45, rcp85)** – Representative Concentration Pathway (RCP)

• **models (24MODELS, PCIC12, BNU-ESM, CCSM4, CESM1-CAM5, CNRM-CM5, CSIRO-Mk3-6-0, CanESM2, FGOALS-g2, GFDL-CM3, ..), optional)** – When calculating the ensemble, include only these models. By default, all 24 models are used.
• **window**(integer, *optional*) – Window
• **freq**(string, *optional*) – Resampling frequency
• **output_format**({'netcdf', 'csv'}, *optional*) – Choose in which format you want to receive the result

**Returns**
• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.
• **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.cold_spell_duration_index_Ensemble_GridPoint_Process
ensemble_grid_point_cold_spell_duration_index

**Parameters**
• **lat**(string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **lon**(string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **start_date**(string, *optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date**(string, *optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **ensemble_percentiles**(string, *optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• **dataset_name**({'bccaqv2'}, *optional*) – Name of the dataset from which to get netcdf files for inputs.
• **rcp** ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)
• **models** ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, *optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• **window**(integer, *optional*) – Window
• **freq**(string, *optional*) – Resampling frequency
• **output_format**({'netcdf', 'csv'}, *optional*) – Choose in which format you want to receive the result

**Returns**
• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.
• **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.cold_spell_duration_index_Ensemble_Polygon_Process
ensemble_polygon_cold_spell_duration_index

**Parameters**
• **lat**(string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **lon**(string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **start_date**(string, *optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date**(string, *optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **ensemble_percentiles**(string, *optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• **dataset_name**({'bccaqv2'}, *optional*) – Name of the dataset from which to get netcdf files for inputs.
• **rcp** ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)
• **models** ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, *optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• **window**(integer, *optional*) – Window
• **freq**(string, *optional*) – Resampling frequency
• **output_format**({'netcdf', 'csv'}, *optional*) – Choose in which format you want to receive the result

**Returns**
• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.
• **output_log** *(text/plain)* – Collected logs during process run.
• shape (application/vnd.geo+json) – Polygon contour, as a geojson string.

• start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• dataset_name ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• rcp ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• models ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• window (integer, optional) – Window

• freq (string, optional) – Resampling frequency

• output_format ({'netcdf', 'csv'}, optional) – Choose in which format you want to recieve the result

Returns

• output (application/x-netcdf, application/zip) – The format depends on the 'output_format' input parameter.

• output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.consecutive_frost_days_Ensemble_Bbox_Process

ensemble_bbox_consecutive_frost_days unidecode.unidecode (v0.1)

Parameters

• lat0 (float) – Minimum latitude.

• lat1 (float) – Maximum latitude.

• lon0 (float) – Minimum longitude.

• lon1 (float) – Maximum longitude.

• start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• dataset_name ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.
• `rcp` ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• `models` ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• `freq`(string, optional) – Resampling frequency

• `output_format` ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

Returns

• `output` (`application/x-netcdf`, `application/zip`) – The format depends on the ‘output_format’ input parameter.

• `output_log` (text/plain) – Collected logs during process run.

class finch.processes.xclim.consecutive_frost_days.Ensemble_GridPoint_Process

ensemble_grid_point_consecutive_frost_days unidecode.unidecode (v0.1)

Parameters

• `lat`(string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• `lon`(string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• `start_date`(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• `end_date`(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• `ensemble_percentiles`(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• `dataset_name` ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• `rcp` ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• `models` ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• `freq`(string, optional) – Resampling frequency

• `output_format` ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

Returns

• `output` (`application/x-netcdf`, `application/zip`) – The format depends on the ‘output_format’ input parameter.

• `output_log` (text/plain) – Collected logs during process run.

4.5. Processes 43
class finch.processes.xclim.consecutive_frost_days_Ensemble_Polygon_Process
ensemble_polygon_consecutive_frost_days unidecode.unidecode (v0.1)

Parameters

- **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
- **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
- **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq** *(string, optional)* – Resampling frequency
- **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

- **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.
- **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.cooling_degree_days_Ensemble_Bbox_Process
ensemble_bbox_cooling_degree_days unidecode.unidecode (v0.1)

Parameters

- **lat0** *(float)* – Minimum latitude.
- **lat1** *(float)* – Maximum latitude.
- **lon0** *(float)* – Minimum longitude.
- **lon1** *(float)* – Maximum longitude.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• dataset_name({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• rcp({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• models({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• thresh(string, optional) – Threshold

• freq(string, optional) – Resampling frequency

• output_format({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

Returns

• output(application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• output_log(text/plain) – Collected logs during process run.

class finch.processes.xclim.cooling_degree_days_Ensemble_GridPoint_Process
ensemble_grid_point_cooling_degree_days unidecode.unidecode (v0.1)

Parameters

• lat(string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• lon(string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• start_date(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• end_date(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• ensemble_percentiles(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• dataset_name({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• rcp({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• models({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• thresh(string, optional) – Threshold

• freq(string, optional) – Resampling frequency

• output_format({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result
Returns

- **output**(application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- **output_log**(text/plain) – Collected logs during process run.

class finch.processes.xclim.cooling_degree_days_Ensemble_Polygon_Process

**ensemble_polygon_cooling_degree_days**

unidecode.unidecode (v0.1)

Parameters

- **shape**(application/vnd.geo+json) – Polygon contour, as a geojson string.
- **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name**({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.
- **rcp**({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)
- **models**({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh**(string, optional) – Threshold
- **freq**(string, optional) – Resampling frequency
- **output_format**({'netcdf', 'csv'}, optional) – Choose in which format you want to recieve the result

Returns

- **output**(application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- **output_log**(text/plain) – Collected logs during process run.

class finch.processes.xclim.cwd_Ensemble_Bbox_Process

**ensemble_bbox_cwd**

unidecode.unidecode (v0.1)

Parameters

- **lat0**(float) – Minimum latitude.
- **lat1**(float) – Maximum latitude.
- **lon0**(float) – Minimum longitude.
- **lon1**(float) – Maximum longitude.
- **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** `finch.processes.xclim.cwd_Ensemble_GridPoint_Process`  
**ensemble_grid_point.cwd** `unidecode.unidecode (v0.1)`  

**Parameters**

• **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *(({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *(({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *(({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
• **thresh**(string, *optional*) – Threshold

• **freq**(string, *optional*) – Resampling frequency

• **output_format** (\{'netcdf', 'csv'\}, *optional*) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

```bash
class finch.processes.xclim.cwd_Ensemble_Polygon_Process
class_polygoncwd
```

**Parameters**

• **shape**(application/vnd.geo+json) – Polygon contour, as a geojson string.

• **start_date**(string, *optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date**(string, *optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles**(string, *optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (\{'bccaqv2'\}, *optional*) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (\{'rcp26', 'rcp45', 'rcp85'\}) – Representative Concentration Pathway (RCP)

• **models** (\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...\}, *optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh**(string, *optional*) – Threshold

• **freq**(string, *optional*) – Resampling frequency

• **output_format** (\{'netcdf', 'csv'\}, *optional*) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

```bash
class finch.processes.xclim.dlyfrzthw_Ensemble_Bbox_Process
class_bboxdlyfrzthw
```

**Parameters**

• **lat0**(float) – Minimum latitude.

• **lat1**(float) – Maximum latitude.
• **lon0** (*float*) – Minimum longitude.
• **lon1** (*float*) – Maximum longitude.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (*{'rcp26', 'rcp45', 'rcp85'}*) – Representative Concentration Pathway (RCP)

• **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh_tasmax** (*string, optional*) – Threshold

• **thresh_tasmin** (*string, optional*) – Threshold

• **freq** (*string, optional*) – Resampling frequency

• **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to recieve the result

Returns

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

```python
class finch.processes.xclim.dlyfrzthw_Ensemble_GridPoint_Process
ensemble_grid_point_dlyfrzthw unidecode.unidecode (v0.1)
```

Parameters

• **lat** (*string*) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** (*string*) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• `dataset_name` ("bccaqv2", optional) – Name of the dataset from which to get netcdf files for inputs.

• `rcp` ("rcp26", 'rcp45', 'rcp85')) – Representative Concentration Pathway (RCP)

• `models` ("24MODELS", 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', .), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• `thresh_tasmax` (string, optional) – Threshold

• `thresh_tasmin` (string, optional) – Threshold

• `freq` (string, optional) – Resampling frequency

• `output_format` ("netcdf", 'csv'), optional) – Choose in which format you want to receive the result

Returns

• `output` (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• `output_log` (text/plain) – Collected logs during process run.

class finch.processes.xclim.dlyfrzthw_Ensemble_Polygon_Process
    ensemble_polygon_dlyfrzthw unidecode.unidecode (v0.1)

Parameters

• `shape` (application/vnd.geo+json) – Polygon contour, as a geojson string.

• `start_date` (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• `end_date` (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• `ensemble_percentiles` (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• `dataset_name` ("bccaqv2"), optional) – Name of the dataset from which to get netcdf files for inputs.

• `rcp` ("rcp26", 'rcp45', 'rcp85')) – Representative Concentration Pathway (RCP)

• `models` ("24MODELS", 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', .), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• `thresh_tasmax` (string, optional) – Threshold

• `thresh_tasmin` (string, optional) – Threshold

• `freq` (string, optional) – Resampling frequency

• `output_format` ("netcdf", 'csv'), optional) – Choose in which format you want to receive the result
Returns

- **output** *(application/x-netcdf, application/zip)* – The format depends on the 'output_format' input parameter.
- **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.dtr_Ensemble_Bbox_Process
ensemble_bbox_dtr
```

**Parameters**

- **lat0** *(float)* – Minimum latitude.
- **lat1** *(float)* – Maximum latitude.
- **lon0** *(float)* – Minimum longitude.
- **lon1** *(float)* – Maximum longitude.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
- **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
- **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq** *(string, optional)* – Resampling frequency
- **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

- **output** *(application/x-netcdf, application/zip)* – The format depends on the 'output_format' input parameter.
- **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.dtr_Ensemble_GridPoint_Process
ensemble_grid_point_dtr
```

**Parameters**

- **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

---

**4.5. Processes**
• **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• **models** ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (string, optional) – Resampling frequency

• **output_format** ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

Returns

• **output** (application/x-netcdf, application/zip) – The format depends on the 'output_format' input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.dtr_Ensemble_Polygon_Process
ensemble_polygon_dtr unidecode.unidecode (v0.1)

Parameters

• **shape** (application/vnd.geo+json) – Polygon contour, as a geojson string.

• **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• **models** ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (string, optional) – Resampling frequency
• **output_format** ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.dtrvar_Ensemble_Bbox_Process
def ensemble_bbox_dtrvar(self, unidecode.unidecode(ref, v0.1))

**Parameters**

• **lat0** (float) – Minimum latitude.

• **lat1** (float) – Maximum latitude.

• **lon0** (float) – Minimum longitude.

• **lon1** (float) – Maximum longitude.

• **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• **models** ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (string, optional) – Resampling frequency

• **output_format** ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.dtrvar_Ensemble_GridPoint_Process
def ensemble_grid_point_dtrvar(self, unidecode.unidecode(ref, v0.1))

**Parameters**

• **lat** (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **lon (string)** – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name (string, optional)** – Name of the dataset from which to get netcdf files for inputs.

• **rcp (['rcp26', 'rcp45', 'rcp85'])** – Representative Concentration Pathway (RCP)

• **models**
  {{'24MODELS', 'PCIC12', 'BNNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional} – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq (string, optional)** – Resampling frequency

• **output_format (['netcdf', 'csv'], optional)** – Choose in which format you want to receive the result

Returns

• **output (application/x-netcdf, application/zip)** – The format depends on the 'output_format' input parameter.

• **output_log (text/plain)** – Collected logs during process run.

class finch.processes.xclim.dtrvar_Ensemble_Polygon_Process

**ensemble_polygon_dtrvar** unidecode.unidecode (v0.1)

Parameters

• **shape (application/vnd.geo+json)** – Polygon contour, as a geojson string.

• **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name (['bc cavq2'], optional)** – Name of the dataset from which to get netcdf files for inputs.

• **rcp (['rcp26', 'rcp45', 'rcp85'])** – Representative Concentration Pathway (RCP)

• **models**
  {{'24MODELS', 'PCIC12', 'BNNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional} – When calculating the ensemble, include only these models. By default, all 24 models are used.
When calculating the ensemble, include only these models. By default, all 24 models are used.

- `freq` (string, optional) – Resampling frequency
- `output_format` (['netcdf', 'csv'], optional) – Choose in which format you want to recieve the result

Returns

- `output` (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- `output_log` (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.etr_Ensemble_Bbox_Process
ensemble_bbox_etr unidecode.unidecode (v0.1)

Parameters

- `lat0` (float) – Minimum latitude.
- `lat1` (float) – Maximum latitude.
- `lon0` (float) – Minimum longitude.
- `lon1` (float) – Maximum longitude.
- `start_date` (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- `end_date` (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- `ensemble_percentiles` (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- `dataset_name` (['bccaqv2'], optional) – Name of the dataset from which to get netcdf files for inputs.
- `rcp` (['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)
- `models` (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- `freq` (string, optional) – Resampling frequency
- `output_format` (['netcdf', 'csv'], optional) – Choose in which format you want to recieve the result

Returns

- `output` (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- `output_log` (text/plain) – Collected logs during process run.
```
```
• lat (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• lon (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• dataset_name ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• rcp ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• models ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• freq (string, optional) – Resampling frequency

• output_format ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

Returns

• output (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.etr_Ensemble_Polygon_Process

ensemble_polygon_etr unidecode.unidecode (v0.1)

Parameters

• shape (application/vnd.geo+json) – Polygon contour, as a geojson string.

• start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• dataset_name ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• rcp ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)
• **models** (('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq**(string, optional) – Resampling frequency

• **output_format** (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.freshet_start_Ensemble_Bbox_Process
ensemble_bbox_freshet_start unidecode.unidecode (v0.1)
```

Parameters

• **lat0**(float) – Minimum latitude.

• **lat1**(float) – Maximum latitude.

• **lon0**(float) – Minimum longitude.

• **lon1**(float) – Maximum longitude.

• **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name**(['bccaqv2'], optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp**(['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)

• **models** (('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh**(string, optional) – Threshold

• **window**(integer, optional) – Window

• **freq**(string, optional) – Resampling frequency

• **output_format** (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
• **output_log (text/plain)** – Collected logs during process run.

```python
class finch.processes.xclim.freshet_start_Ensemble_GridPoint_Process
ensemble_grid_point_freshet_start unidecode.unidecode (v0.1)
```

**Parameters**

- **lat (string)** – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon (string)** – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name (bccaqv2, optional)** – Name of the dataset from which to get netcdf files for inputs.
- **rcp (rcp26, rcp45, rcp85) – Representative Concentration Pathway (RCP)**
- **models (24MODELS, PCIC12, BNU-ESM, CCSM4, CESM1-CAM5, CNRM-CM5, CSIRO-Mk3-6-0, CanESM2, FGOALS-g2, GFDL-CM3) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh (string, optional)** – Threshold
- **window (integer, optional)** – Window
- **freq (string, optional)** – Resampling frequency
- **output_format (netcdf, csv) – Choose in which format you want to receive the result**

**Returns**

- **output (application/x-netcdf, application/zip)** – The format depends on the 'output_format' input parameter.
- **output_log (text/plain)** – Collected logs during process run.

```python
class finch.processes.xclim.freshet_start_Ensemble_Polygon_Process
ensemble_polygon_freshet_start unidecode.unidecode (v0.1)
```

**Parameters**

- **shape (application/vnd.geo+json)** – Polygon contour, as a geojson string.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** *(string, optional)* – Threshold

• **window** *(integer, optional)* – Window

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

• **output** *(application/x-netcdf, application/zip)* – The format depends on the `output_format` input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.frost_days_Ensemble_Bbox_Process

`ensemble_bbox_frost_days` unidecode.unidecode (v0.1)

Parameters

• **lat0** *(float)* – Minimum latitude.

• **lat1** *(float)* – Maximum latitude.

• **lon0** *(float)* – Minimum longitude.

• **lon1** *(float)* – Maximum longitude.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
**freq (string, optional)** – Resampling frequency

**output_format (['netcdf', 'csv'], optional)** – Choose in which format you want to receive the result

**Returns**

- **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- **output_log** (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.frost_days_Ensemble_GridPoint_Process
ensemble_grid_point_frost_days unidecode.unidecode (v0.1)
```

**Parameters**

- **lat (string)** – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon (string)** – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name (['bccaqv2'], optional)** – Name of the dataset from which to get netcdf files for inputs.
- **rcp (['rcp26', 'rcp45', 'rcp85'])** – Representative Concentration Pathway (RCP)
- **models (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...], optional)** – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq (string, optional)** – Resampling frequency
- **output_format (['netcdf', 'csv'], optional)** – Choose in which format you want to receive the result

**Returns**

- **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- **output_log** (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.frost_days_Ensemble_Polygon_Process
ensemble_polygon_frost_days unidecode.unidecode (v0.1)
```

**Parameters**

- **shape (application/vnd.geo+json)** – Polygon contour, as a geojson string.
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *(({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

• **output** *(application/x-netcdf, application/zip)* – The format depends on the `output_format` input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** *finch.processes.xclim.growing_degree_days_Ensemble_Bbox_Process*

**ensemble_bbox_growing_degree_days** *unidecode.unidecode (v0.1)*

**Parameters**

• **lat0** *(float)* – Minimum latitude.

• **lat1** *(float)* – Maximum latitude.

• **lon0** *(float)* – Minimum longitude.

• **lon1** *(float)* – Maximum longitude.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
models (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

thresh (string, optional) – Threshold

freq (string, optional) – Resampling frequency

output_format (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

output (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.growing_degree_days_Ensemble_GridPoint_Process

ensemble_grid_point_growing_degree_days unidecode.unidecode (v0.1)

Parameters

lat (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

lon (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

dataset_name (['bccaqv2'], optional) – Name of the dataset from which to get netcdf files for inputs.

rcp (['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)

models (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

thresh (string, optional) – Threshold

freq (string, optional) – Resampling frequency

output_format (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

output (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

output_log (text/plain) – Collected logs during process run.
class finch.processes.xclim.growing_degree_days_Ensemble_Polygon_Process
ensemble_polygon_growing_degree_days unidecode.unidecode (v0.1)

Parameters

- **shape** (*application/vnd.geo+json*) – Polygon contour, as a geojson string.
- **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** (({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.
- **rcp** (({'rcp26', 'rcp45', 'rcp85'})) – Representative Concentration Pathway (RCP)
- **models** (({''24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh** (*string, optional*) – Threshold
- **freq** (*string, optional*) – Resampling frequency
- **output_format** (({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

Returns

- **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.
- **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.heat_wave_frequency_Ensemble_Bbox_Process
ensemble_bbox_heat_wave_frequency unidecode.unidecode (v0.1)

Parameters

- **lat0** (*float*) – Minimum latitude.
- **lat1** (*float*) – Maximum latitude.
- **lon0** (*float*) – Minimum longitude.
- **lon1** (*float*) – Maximum longitude.
- **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

dataset_name ("bccaqv2", optional) – Name of the dataset from which to get netcdf files for inputs.

crp ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)

models ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

thresh_tasmin (string, optional) – Threshold

thresh_tasmax (string, optional) – Threshold

window (integer, optional) – Window

freq (string, optional) – Resampling frequency

output_format ("netcdf", "csv", optional) – Choose in which format you want to receive the result

Returns

output (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.heat_wave_frequency_Ensemble_GridPoint_Process

ensemble_grid_point_heat_wave_frequency

Parameters

lat (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

lon (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

dataset_name ("bccaqv2", optional) – Name of the dataset from which to get netcdf files for inputs.

crp ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)

models ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
Finch Documentation, Release 0.5.3

• **thresh_tasmin** *(string, optional)* – Threshold

• **thresh_tasmax** *(string, optional)* – Threshold

• **window** *(integer, optional)* – Window

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *(['netcdf', 'csv'], optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.heat_wave_frequency_Ensemble_Polygon_Process

ensemble_polygon_heat_wave_frequency
```

**Parameters**

• **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *(['bccaqv2'], optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *(['rcp26', 'rcp45', 'rcp85'])* – Representative Concentration Pathway (RCP)

• **models** *(['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...], optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh_tasmin** *(string, optional)* – Threshold

• **thresh_tasmax** *(string, optional)* – Threshold

• **window** *(integer, optional)* – Window

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *(['netcdf', 'csv'], optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.
class finch.processes.xclim.heat_wave_index_Ensemble_Bbox_Process

ensemble_bbox_heat_wave_index unidecode.unidecode (v0.1)

Parameters

- **lat0** (float) – Minimum latitude.
- **lat1** (float) – Maximum latitude.
- **lon0** (float) – Minimum longitude.
- **lon1** (float) – Maximum longitude.
- **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** ( {'bccaqv2'}, optional ) – Name of the dataset from which to get netcdf files for inputs.
- **rcp** ( {'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)
- **models** ( {'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ... }, optional ) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh** (string, optional) – Threshold
- **window** (integer, optional) – Window
- **freq** (string, optional) – Resampling frequency
- **output_format** ( {'netcdf', 'csv'}, optional ) – Choose in which format you want to receive the result

Returns

- **output** ( application/x-netcdf, application/zip ) – The format depends on the ‘output_format’ input parameter.
- **output_log** ( text/plain ) – Collected logs during process run.

class finch.processes.xclim.heat_wave_index_Ensemble_GridPoint_Process

ensemble_grid_point_heat_wave_index unidecode.unidecode (v0.1)

Parameters

- **lat** (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon** (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name**(‘bccaqv2’, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp**(‘rcp26’, ’rcp45’, ’rcp85’) – Representative Concentration Pathway (RCP)

• **models**(‘24MODELS’, ’PCIC12’, ’BNU-ESM’, ’CCSM4’, ’CESM1-CAM5’, ’CNRM-CM5’, ’CSIRO-Mk3-6-0’, ’CanESM2’, ’FGOALS-g2’, ’GFDL-CM3’,..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh**(string, optional) – Threshold

• **window**(integer, optional) – Window

• **freq**(string, optional) – Resampling frequency

• **output_format**(‘netcdf’, ’csv’), optional) – Choose in which format you want to receive the result

Returns

• **output**(application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log**(text/plain) – Collected logs during process run.

class finch.processes.xclim.heat_wave_index.Ensemble_Polygon_Process

ensemble_polygon_heat_wave_index unidecode.unidecode (v0.1)

Parameters

• **shape**(application/vnd.geo+json) – Polygon contour, as a geojson string.

• **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name**(‘bccaqv2’, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp**(‘rcp26’, ’rcp45’, ’rcp85’) – Representative Concentration Pathway (RCP)

• **models**(‘24MODELS’, ’PCIC12’, ’BNU-ESM’, ’CCSM4’, ’CESM1-CAM5’, ’CNRM-CM5’, ’CSIRO-Mk3-6-0’, ’CanESM2’, ’FGOALS-g2’, ’GFDL-CM3’,..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh**(string, optional) – Threshold
• `window` (integer, optional) – Window
• `freq` (string, optional) – Resampling frequency
• `output_format` (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

• `output` (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
• `output_log` (text/plain) – Collected logs during process run.

class finch.processes.xcim.heat_wave_max_length_Ensemble_Bbox_Process
  ensemble_bbox_heat_wave_max_length unidecode.unidecode (v0.1)

Parameters

• `lat0` (float) – Minimum latitude.
• `lat1` (float) – Maximum latitude.
• `lon0` (float) – Minimum longitude.
• `lon1` (float) – Maximum longitude.
• `start_date` (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• `end_date` (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• `ensemble_percentiles` (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• `dataset_name` (['bccaqv2'], optional) – Name of the dataset from which to get netcdf files for inputs.
• `rcp` (['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)
• `models` (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• `thresh_tasmin` (string, optional) – Threshold
• `thresh_tasmax` (string, optional) – Threshold
• `window` (integer, optional) – Window
• `freq` (string, optional) – Resampling frequency
• `output_format` (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

• `output` (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
• `output_log` (text/plain) – Collected logs during process run.
class finch.processes.xclim.heat_wave_max_length_Ensemble_GridPoint_Process
ensemble_grid_point_heat_wave_max_length unidecode.unidecode (v0.1)

Parameters

- **lat (string)** – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon (string)** – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name ("bccaqv2"), optional** – Name of the dataset from which to get netcdf files for inputs.
- **rcp ("rcp26", "rcp45", "rcp85")** – Representative Concentration Pathway (RCP)
- **models ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional** – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh_tasmin (string, optional)** – Threshold
- **thresh_tasmax (string, optional)** – Threshold
- **window (integer, optional)** – Window
- **freq (string, optional)** – Resampling frequency
- **output_format ("netcdf", "csv"), optional** – Choose in which format you want to receive the result

Returns

- **output (application/x-netcdf, application/zip)** – The format depends on the ‘output_format’ input parameter.
- **output_log (text/plain)** – Collected logs during process run.

class finch.processes.xclim.heat_wave_max_length_Ensemble_Polygon_Process
ensemble_polygon_heat_wave_max_length unidecode.unidecode (v0.1)

Parameters

- **shape (application/vnd.geo+json)** – Polygon contour, as a geojson string.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *(\{'bccaqv2\}', optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *(\{'rcp26', 'rcp45', 'rcp85'\})* – Representative Concentration Pathway (RCP)

• **models** *(\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..\}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh_tasmin** *(string, optional)* – Threshold

• **thresh_tasmax** *(string, optional)* – Threshold

• **window** *(integer, optional)* – Window

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *(\{'netcdf', 'csv'\}, optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** **finch.processes.xclim.heat_wave_total_length_Embelle_Ensemble_Bbox_Process**

**ensemble_bbox_heat_wave_total_length** unidecode.unidecode (v0.1)

**Parameters**

• **lat0** *(float)* – Minimum latitude.

• **lat1** *(float)* – Maximum latitude.

• **lon0** *(float)* – Minimum longitude.

• **lon1** *(float)* – Maximum longitude.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *(\{'bccaqv2\}', optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *(\{'rcp26', 'rcp45', 'rcp85'\})* – Representative Concentration Pathway (RCP)

• **models** *(\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..\}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
'FGOALS-g2', 'GFDL-CM3', ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

- **thresh_tasmin** *(string, optional) – Threshold*
- **thresh_tasmax** *(string, optional) – Threshold*
- **window** *(integer, optional) – Window*
- **freq** *(string, optional) – Resampling frequency*
- **output_format** *({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result*

**Returns**

- **output** *(application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.*
- **output_log** *(text/plain) – Collected logs during process run.*

```python
class finch.processes.xclim.heat_wave_total_length_Ensemble_GridPoint_Process
ensemble_grid_point_heat_wave_total_length unidecode.unidecode (v0.1)
```

**Parameters**

- **lat** *(string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.*
- **lon** *(string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.*
- **start_date** *(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.*
- **end_date** *(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.*
- **ensemble_percentiles** *(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.*
- **dataset_name** *({'bcacq2'}, optional) – Name of the dataset from which to get netcdf files for inputs.*
- **rcp** *({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)*
- **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.*
- **thresh_tasmin** *(string, optional) – Threshold*
- **thresh_tasmax** *(string, optional) – Threshold*
- **window** *(integer, optional) – Window*
- **freq** *(string, optional) – Resampling frequency*
- **output_format** *({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result*
• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.heat_wave_total_length_Ensemble_Polygon_Process

**ensemble_polygon_heat_wave_total_length** unidecode.unidecode (v0.1)

**Parameters**

• **shape** (*application/vnd.geo+json*) – Polygon contour, as a geojson string.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (collected logs during process run.

• **models** ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh_tasmin** (*string, optional*) – Threshold

• **thresh_tasmax** (*string, optional*) – Threshold

• **window** (*integer, optional*) – Window

• **freq** (*string, optional*) – Resampling frequency

• **output_format** ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.heating_degree_days_Ensemble_Bbox_Process

**ensemble_bbox_heating_degree_days** unidecode.unidecode (v0.1)

**Parameters**

• **lat0** (*float*) – Minimum latitude.

• **lat1** (*float*) – Maximum latitude.

• **lon0** (*float*) – Minimum longitude.

• **lon1** (*float*) – Maximum longitude.
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *(('bccaqv2'), optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *(('rcp26', 'rcp45', 'rcp85'))* – Representative Concentration Pathway (RCP)

• **thresh** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *(('netcdf', 'csv'), optional)* – Choose in which format you want to receive the result.

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** `finch.processes.xclim.heating_degree_days_Ensemble_GridPoint_Process`  
`ensemble_grid_point_heating_degree_days` `unidecode.unidecode (v0.1)`

**Parameters**

• **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *(('bccaqv2'), optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *(('rcp26', 'rcp45', 'rcp85'))* – Representative Concentration Pathway (RCP)
• **models**
  ('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4',
  'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2',
  'FGOALS-g2', 'GFDL-CM3', ..), optional) – When calculating the
  ensemble, include only these models. By default, all 24 models are used.

• **thresh**(string, optional) – Threshold

• **freq**(string, optional) – Resampling frequency

• **output_format**
  ('netcdf', 'csv', optional) – Choose in which format
  you want to recieve the result

**Returns**

• **output**
  (application/x-netcdf, application/zip) – The format depends on
  the ‘output_format’ input parameter.

• **output_log**
  (text/plain) – Collected logs during process run.

class finch.processes.xclim.heating_degree_days_Ensemble_Polygon_Process

ensemble_polygon_heating_degree_days

unidecode.unidecode (v0.1)

**Parameters**

• **shape**
  (application/vnd.geo+json) – Polygon contour, as a geojson string.

• **start_date**
  (string, optional) – Initial date for temporal subsetting. Can be ex-
  pressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to
  first day in file.

• **end_date**
  (string, optional) – Final date for temporal subsetting. Can be ex-
  pressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to
  last day in file.

• **ensemble_percentiles**
  (string, optional) – Ensemble percentiles to calcu-
  late for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name**
  ('bccaqv2'), optional) – Name of the dataset from which to
  get netcdf files for inputs.

• **rcp**
  ('rcp26', 'rcp45', 'rcp85') – Representative Concentration Pathway
  (RCP)

• **models**
  ('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4',
  'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2',
  'FGOALS-g2', 'GFDL-CM3', ..), optional) – When calculating the
  ensemble, include only these models. By default, all 24 models are used.

• **thresh**(string, optional) – Threshold

• **freq**(string, optional) – Resampling frequency

• **output_format**
  ('netcdf', 'csv', optional) – Choose in which format
  you want to recieve the result

**Returns**

• **output**
  (application/x-netcdf, application/zip) – The format depends on
  the ‘output_format’ input parameter.

• **output_log**
  (text/plain) – Collected logs during process run.

class finch.processes.xclim.ice_days_Ensemble_Bbox_Process

ensemble_bbox_ice_days

unidecode.unidecode (v0.1)
Parameters

- **lat0 (float)** – Minimum latitude.
- **lat1 (float)** – Maximum latitude.
- **lon0 (float)** – Minimum longitude.
- **lon1 (float)** – Maximum longitude.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name ({'bccaqv2'}, optional)** – Name of the dataset from which to get netcdf files for inputs.
- **rcp ({'rcp26', 'rcp45', 'rcp85'})** – Representative Concentration Pathway (RCP)
- **models ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional)** – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq (string, optional)** – Resampling frequency
- **output_format ({'netcdf', 'csv'}, optional)** – Choose in which format you want to recieve the result

Returns

- **output (application/x-netcdf, application/zip)** – The format depends on the ‘output_format’ input parameter.
- **output_log (text/plain)** – Collected logs during process run.

```python
class finch.processes.xclim.ice_days_Ensemble_GridPoint_Process
  ensemble_grid_point_ice_days
```

Parameters

- **lat (string)** – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon (string)** – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• **dataset_name** ('bccaqv2', optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ('rcp26', 'rcp45', 'rcp85') – Representative Concentration Pathway (RCP)

• **models** ('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (string, optional) – Resampling frequency

• **output_format** ('netcdf', 'csv'), optional) – Choose in which format you want to receive the result

Returns

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.ice_days_Ensemble_Polygon_Process

ensemble_polygon_ice_days

Parameters

• **shape** (application/vnd.geo+json) – Polygon contour, as a geojson string.

• **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ('bccaqv2', optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ('rcp26', 'rcp45', 'rcp85') – Representative Concentration Pathway (RCP)

• **models** ('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (string, optional) – Resampling frequency

• **output_format** ('netcdf', 'csv'), optional) – Choose in which format you want to receive the result

Returns

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.
class finch.processes.xclim.liquidprcptot_Ensemble_Bbox_Process

ensemble_bbox_liquidprcptot unidecode.unidecode (v0.1)

Parameters

- **lat0** *(float)* – Minimum latitude.
- **lat1** *(float)* – Maximum latitude.
- **lon0** *(float)* – Minimum longitude.
- **lon1** *(float)* – Maximum longitude.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
- **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
- **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq** *(string, optional)* – Resampling frequency
- **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

- **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.
- **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.liquidprcptot_Ensemble_GridPoint_Process

ensemble_grid_point_liquidprcptot unidecode.unidecode (v0.1)

Parameters

- **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netCDF files for inputs.

• **rcp** (*{'rcp26', 'rcp45', 'rcp85'}*) – Representative Concentration Pathway (RCP)

• **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (*string, optional*) – Resampling frequency

• **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to receive the result

Returns

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.liquidprcptot_Ensemble_Polygon_Process

```
ensemble_polygon_liquidprcptot unidecode.unidecode (v0.1)
```

Parameters

• **shape** (*application/vnd.geo+json*) – Polygon contour, as a GeoJSON string.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netCDF files for inputs.

• **rcp** (*{'rcp26', 'rcp45', 'rcp85'}*) – Representative Concentration Pathway (RCP)

• **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (*string, optional*) – Resampling frequency

• **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to receive the result

Returns

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.
output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.max_n_day_precipitation_amount_Ensemble_Bbox_Process
ensemble_bbox_max_n_day_precipitation_amount unidecode.unidecode (v0.1)

Parameters

• lat0 (float) – Minimum latitude.
• lat1 (float) – Maximum latitude.
• lon0 (float) – Minimum longitude.
• lon1 (float) – Maximum longitude.
• start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• dataset_name ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.
• rcp ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)
  • models ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', '...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• window (integer, optional) – Window
• freq (string, optional) – Resampling frequency
• output_format ({'netcdf', 'csv'}, optional) – Choose in which format you want to recieve the result

Returns

• output (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
• output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.max_n_day_precipitation_amount_Ensemble_GridPoint_Process
ensemble_grid_point_max_n_day_precipitation_amount unidecode.unidecode (v0.1)

Parameters

• lat (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• lon (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date** (*string*, *optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string*, *optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ("bccaqv2", *optional*) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)

• **models** ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), *optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **window** (*integer*, *optional*) – Window

• **freq** (*string*, *optional*) – Resampling frequency

• **output_format** ("netcdf", "csv"); *optional*) – Choose in which format you want to recieve the result

**Returns**

• **output** (*application/x-netcdf*, *application/zip*) – The format depends on the 'output_format' input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

**class** `finch.processes.xclim.max_n_day_precipitation_amount_Ensemble_Polygon_Process`  
`ensemble_polygon_max_n_day_precipitation_amount`  
`unidecode.unidecode (v0.1)`

**Parameters**

• **shape** (*application/vnd.geo+json*) – Polygon contour, as a geojson string.

• **start_date** (*string*, *optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string*, *optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string*, *optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ("bccaqv2", *optional*) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)

• **models** ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), *optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **window** (*integer*, *optional*) – Window

• **freq** (*string*, *optional*) – Resampling frequency
• **output_format** (\{'netcdf', 'csv'\}, optional) – Choose in which format you want to receive the result

**Returns**

- **output** (application/x-netcdf, application/zip) – The format depends on the 'output_format' input parameter.
- **output_log** (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.prcptot_Ensemble_Bbox_Process
ensemble_bbox_prcptot
```

**Parameters**

- **lat0** (float) – Minimum latitude.
- **lat1** (float) – Maximum latitude.
- **lon0** (float) – Minimum longitude.
- **lon1** (float) – Maximum longitude.
- **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** (\{'bccaqv2'\}, optional) – Name of the dataset from which to get netcdf files for inputs.
- **rcp** (\{'rcp26', 'rcp45', 'rcp85'\}) – Representative Concentration Pathway (RCP)
- **models** (\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..\}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq** (string, optional) – Resampling frequency
- **output_format** (\{'netcdf', 'csv'\}, optional) – Choose in which format you want to receive the result

**Returns**

- **output** (application/x-netcdf, application/zip) – The format depends on the 'output_format' input parameter.
- **output_log** (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.prcptot_Ensemble_GridPoint_Process
ensemble_grid_point_prcptot
```

**Parameters**

- **lat** (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

• **output** *(application/x-netcdf, application/zip)* – The format depends on the 'output_format' input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.prcptot_Ensemble_Polygon_Process
ensemble_polygon_prcptot unidecode.unidecode (v0.1)

Parameters

• **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
```

82 Chapter 4. Credits
When calculating the ensemble, include only these models. By default, all 24 models are used.

- **freq** (*string*, *optional*) – Resampling frequency
- **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to receive the result

Returns

- **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.
- **output_log** (*text/plain*) – Collected logs during process run.

### class **finch.processes.xclim.rain_frzgr_Ensemble_Bbox_Process**

#### ensemble_bbox_rain_frzgr

**class** unidecode.unidecode (v0.1)

**Returns**

- **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.
- **output_log** (*text/plain*) – Collected logs during process run.

### Parameters

- **lat0** (*float*) – Minimum latitude.
- **lat1** (*float*) – Maximum latitude.
- **lon0** (*float*) – Minimum longitude.
- **lon1** (*float*) – Maximum longitude.
- **start_date** (*string*, *optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** (*string*, *optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** (*string*, *optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netcdf files for inputs.
- **rcp** (*{'rcp26', 'rcp45', 'rcp85'}*) – Representative Concentration Pathway (RCP)
- **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh** (*string*, *optional*) – Threshold
- **freq** (*string*, *optional*) – Resampling frequency
- **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to receive the result

### Returns

- **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.
- **output_log** (*text/plain*) – Collected logs during process run.

### 4.5. Processes
Parameters

- **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
- **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
- **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh** *(string, optional)* – Threshold
- **freq** *(string, optional)* – Resampling frequency
- **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

- **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.
- **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.rain_frzgr_Ensemble_Polygon_Process

**ensemble_polygon_rain_frzgr** unidecode.unidecode (v0.1)

Parameters

- **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
• rcp (['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)
• models (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• thresh (string, optional) – Threshold
• freq (string, optional) – Resampling frequency
• output_format (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns
• output (application/x-netcdf, application/zip) – The format depends on the 'output_format' input parameter.
• output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.rx1day_Ensemble_Bbox_Process
ensemble_bbox_rx1day unidecode.unidecode (v0.1)

Parameters
• lat0 (float) – Minimum latitude.
• lat1 (float) – Maximum latitude.
• lon0 (float) – Minimum longitude.
• lon1 (float) – Maximum longitude.
• start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• dataset_name (['bccaqv2'], optional) – Name of the dataset from which to get netcdf files for inputs.
• rcp (['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)
• models (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• freq (string, optional) – Resampling frequency
• output_format (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns
• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.rx1day_Ensemble_GridPoint_Process
    ensemble_grid_point_rx1day unidecode.unidecode (v0.1)

    Parameters

    • **lat**(string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

    • **lon**(string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

    • **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

    • **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

    • **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

    • **dataset_name**({'bccaqc2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

    • **rcp**({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

    • **models**
        ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

    • **freq**(string, optional) – Resampling frequency

    • **output_format**({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

Returns

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.rx1day_Ensemble_Polygon_Process
    ensemble_polygon_rx1day unidecode.unidecode (v0.1)

    Parameters

    • **shape**(application/vnd.geo+json) – Polygon contour, as a geojson string.

    • **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

    • **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

- **dataset_name** *(('bccaqv2'), optional)* – Name of the dataset from which to get netcdf files for inputs.

- **rcp** *(('rcp26', 'rcp45', 'rcp85'))* – Representative Concentration Pathway (RCP)

- **models** *((24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..), optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

- **freq** *(string, optional)* – Resampling frequency

- **output_format** *(('netcdf', 'csv'), optional)* – Choose in which format you want to receive the result

**Returns**

- **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

- **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.sdii_Ensemble_Bbox_Process

ensemble_bbox_sdii unidecode.unidecode (v0.1)
```

**Parameters**

- **lat0** *(float)* – Minimum latitude.

- **lat1** *(float)* – Maximum latitude.

- **lon0** *(float)* – Minimum longitude.

- **lon1** *(float)* – Maximum longitude.

- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

- **dataset_name** *(('bccaqv2'), optional)* – Name of the dataset from which to get netcdf files for inputs.

- **rcp** *(('rcp26', 'rcp45', 'rcp85'))* – Representative Concentration Pathway (RCP)

- **models** *((24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..), optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

- **thresh** *(string, optional)* – Threshold

- **freq** *(string, optional)* – Resampling frequency

4.5. Processes
• **output_format** ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

**Returns**

• **output** ([application/x-netcdf, application/zip]) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.sdii_Ensemble_GridPoint_Process
ensemble_grid_point_sdii
```  

Parameters

• **lat** (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• **models** ({{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional}) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** (string, optional) – Threshold

• **freq** (string, optional) – Resampling frequency

• **output_format** ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

**Returns**

• **output** ([application/x-netcdf, application/zip]) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.sdii_Ensemble_Polygon_Process
ensemble_polygon_sdii
```  

Parameters

• **shape** (application/vnd.geo+json) – Polygon contour, as a geojson string.
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** `finch.processes.xclim.solidprcptot_Ensemble_Bbox_Process`  
`ensemble_bbox_solidprcptot` `unidecode.unidecode (v0.1)`

**Parameters**

• **lat0** *(float)* – Minimum latitude.

• **lat1** *(float)* – Maximum latitude.

• **lon0** *(float)* – Minimum longitude.

• **lon1** *(float)* – Maximum longitude.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
• **models** (\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (string, optional) – Resampling frequency

• **output_format** (\{'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.solidprcptot_Ensemble_GridPoint_Process
ensemble_grid_point_solidprcptot

Parameters

• **lat** (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (\{'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (\{'rcp26', 'rcp45', 'rcp85'})) – Representative Concentration Pathway (RCP)

• **models** (\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (string, optional) – Resampling frequency

• **output_format** (\{'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.
```
Parameters

- **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
- **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
- **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq** *(string, optional)* – Resampling frequency
- **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

- **output** *(application/x-netcdf, application/zip)* – The format depends on the 'output_format' input parameter.
- **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.tg10p_Ensemble_Bbox_Process
ensemble_bbox_tg10p unidecode.unidecode (v0.1)
```

Parameters

- **lat0** *(float)* – Minimum latitude.
- **lat1** *(float)* – Maximum latitude.
- **lon0** *(float)* – Minimum longitude.
- **lon1** *(float)* – Maximum longitude.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
- `rcp` ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)
- `models` ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ...) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- `freq` (string, optional) – Resampling frequency
- `output_format` ("netcdf", "csv", optional) – Choose in which format you want to receive the result

Returns

- `output` (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- `output_log` (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.tg10p_Ensemble_GridPoint_Process
    ensemble_grid_point_tg10p
```

Parameters

- `lat` (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- `lon` (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- `start_date` (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- `end_date` (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- `ensemble_percentiles` (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- `dataset_name` ("bccaqv2", optional) – Name of the dataset from which to get netcdf files for inputs.
- `rcp` ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)
- `models` ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ...) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- `freq` (string, optional) – Resampling frequency
- `output_format` ("netcdf", "csv", optional) – Choose in which format you want to receive the result

Returns

- `output` (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- `output_log` (text/plain) – Collected logs during process run.
class finch.processes.xclim.tg10p_Ensemble_Polygon_Process
ensemble_polygon_tg10p

Parameters

- **shape** (application/vnd.geo+json) – Polygon contour, as a geojson string.
- **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** ("bccaqv2", optional) – Name of the dataset from which to get netcdf files for inputs.
- **rcp** ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)
- **models** ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq** (string, optional) – Resampling frequency
- **output_format** ("netcdf", "csv", optional) – Choose in which format you want to recieve the result

Returns

- **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.tg90p_Ensemble_Bbox_Process
ensemble_bbox_tg90p

Parameters

- **lat0** (float) – Minimum latitude.
- **lat1** (float) – Maximum latitude.
- **lon0** (float) – Minimum longitude.
- **lon1** (float) – Maximum longitude.
- **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• `dataset_name` (bccaqv2, optional) – Name of the dataset from which to get netcdf files for inputs.
• `rcp` (rcp26, rcp45, rcp85) – Representative Concentration Pathway (RCP)
• `models` (24MODELS, PCIC12, BNU-ESM, CCSM4, CESM1-CAM5, CNRM-CM5, CSIRO-Mk3-6-0, CanESM2, FGOALS-g2, GFDL-CM3, ...) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• `freq` (string, optional) – Resampling frequency
• `output_format` (netcdf, csv, optional) – Choose in which format you want to receive the result

Returns
• `output` (application/x-netcdf, application/zip) – The format depends on the `output_format` input parameter.
• `output_log` (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.tg90p_Ensemble_GridPoint_Process
ensemble_grid_point_tg90p
```

Parameters
• `lat` (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• `lon` (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• `start_date` (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• `end_date` (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• `ensemble_percentiles` (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• `dataset_name` (bccaqv2, optional) – Name of the dataset from which to get netcdf files for inputs.
• `rcp` (rcp26, rcp45, rcp85) – Representative Concentration Pathway (RCP)
• `models` (24MODELS, PCIC12, BNU-ESM, CCSM4, CESM1-CAM5, CNRM-CM5, CSIRO-Mk3-6-0, CanESM2, FGOALS-g2, GFDL-CM3, ...) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• `freq` (string, optional) – Resampling frequency
• `output_format` (netcdf, csv, optional) – Choose in which format you want to receive the result

Returns
• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.tg90p_Ensemble_Polygon_Process

ensemble_polygon_tg90p unidecode.unidecode (v0.1)

Parameters

• **shape**(application/vnd.geo+json) – Polygon contour, as a geojson string.

• **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name**({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp**({"rcp26", "rcp45", "rcp85"}) – Representative Concentration Pathway (RCP)

• **models**

  {'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq**(string, optional) – Resampling frequency

• **output_format**({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result.

Returns

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.tg_Ensemble_Bbox_Process

ensemble_bbox_tg unidecode.unidecode (v0.1)

Parameters

• **lat0**(float) – Minimum latitude.

• **lat1**(float) – Maximum latitude.

• **lon0**(float) – Minimum longitude.

• **lon1**(float) – Maximum longitude.

• **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

```
• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (*{'rcp26', 'rcp45', 'rcp85'}*) – Representative Concentration Pathway (RCP)

• **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to receive the result

Returns

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.tg_Ensemble_GridPoint_Process
ensemble_grid_point_tg unidecode.unidecode (v0.1)

Parameters

• **lat** (*string*) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** (*string*) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (*{'rcp26', 'rcp45', 'rcp85'}*) – Representative Concentration Pathway (RCP)

• **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to receive the result
Returns

- **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

- **output_log** (*text/plain*) – Collected logs during process run.

```python
class finch.processes.xclim.tg_Ensemble_Polygon_Process
ensemble_polygon_tg unidecode.unidecode (v0.1)
```

**Parameters**

- **shape** (*application/vnd.geo+json*) – Polygon contour, as a geojson string.

- **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

- **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

- **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

- **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netcdf files for inputs.

- **rcp** (*{'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

- **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

- **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to receive the result

Returns

- **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

- **output_log** (*text/plain*) – Collected logs during process run.

```python
class finch.processes.xclim.tg_mean_Ensemble_Bbox_Process
ensemble_bbox_tg_mean unidecode.unidecode (v0.1)
```

**Parameters**

- **lat0** (*float*) – Minimum latitude.

- **lat1** (*float*) – Maximum latitude.

- **lon0** (*float*) – Minimum longitude.

- **lon1** (*float*) – Maximum longitude.

- **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• `end_date (string, optional)` – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• `ensemble_percentiles (string, optional)` – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• `dataset_name ({'bccaqv2'}, optional)` – Name of the dataset from which to get netcdf files for inputs.

• `rcp ({'rcp26', 'rcp45', 'rcp85'})` – Representative Concentration Pathway (RCP)

• `models ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)` – When calculating the ensemble, include only these models. By default, all 24 models are used.

• `freq (string, optional)` – Resampling frequency

• `output_format ({'netcdf', 'csv'}, optional)` – Choose in which format you want to receive the result

Returns

• `output (application/x-netcdf, application/zip)` – The format depends on the ‘output_format’ input parameter.

• `output_log (text/plain)` – Collected logs during process run.

class `finch.processes.xclim.tg_mean_Ensemble_GridPoint_Process`

`ensemble_grid_point_tg_mean` unidecode.unidecode (v0.1)

Parameters

• `lat (string)` – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• `lon (string)` – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• `start_date (string, optional)` – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• `end_date (string, optional)` – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• `ensemble_percentiles (string, optional)` – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• `dataset_name ({'bccaqv2'}, optional)` – Name of the dataset from which to get netcdf files for inputs.

• `rcp ({'rcp26', 'rcp45', 'rcp85'})` – Representative Concentration Pathway (RCP)

• `models ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)` – When calculating the ensemble, include only these models. By default, all 24 models are used.

• `freq (string, optional)` – Resampling frequency
• **output_format** (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.tg_mean_Ensemble_Polygon_Process
    ensemble_polygon_tg_mean unidecode.unidecode (v0.1)

**Parameters**

• **shape** (application/vnd.geo+json) – Polygon contour, as a geojson string.

• **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma-separated list of integers.

• **dataset_name**({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)

• **models** (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq**(string, optional) – Resampling frequency

• **output_format** (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.tn10p_Ensemble_Bbox_Process
    ensemble_bbox_tn10p unidecode.unidecode (v0.1)

**Parameters**

• **lat0**(float) – Minimum latitude.

• **lat1**(float) – Maximum latitude.

• **lon0**(float) – Minimum longitude.

• **lon1**(float) – Maximum longitude.
• **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name**(\{'bccaqv2\}', optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp**(\{'rcp26', 'rcp45', 'rcp85'\}) – Representative Concentration Pathway (RCP)

• **models** (\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'GFDL-GS', 'GFDL-CM3', \}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq**(string, optional) – Resampling frequency

• **output_format**(\{'netcdf', 'csv'\}, optional) – Choose in which format you want to receive the result

**Returns**

• **output**(application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log**(text/plain) – Collected logs during process run.

**class** finch.processes.xclim.tn10p_Ensemble_GridPoint_Process

**ensemble_grid_point_tn10p** unidecode.unidecode (v0.1)

**Parameters**

• **lat**(string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon**(string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name**(\{'bccaqv2\}', optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp**(\{'rcp26', 'rcp45', 'rcp85'\}) – Representative Concentration Pathway (RCP)

• **models** (\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', \}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
When calculating the ensemble, include only these models. By default, all 24 models are used.

- freq (string, optional) – Resampling frequency
- output_format (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

- output (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- output_log (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.tn10p_Ensemble_Polygon_Process
ensemble_polygon_tn10p
```

Parameters

- shape (application/vnd.geo+json) – Polygon contour, as a geojson string.
- start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- dataset_name (['bccaqv2'], optional) – Name of the dataset from which to get netcdf files for inputs.
- rcp (['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)
- models (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- freq (string, optional) – Resampling frequency
- output_format (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

- output (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
- output_log (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.tn90p_Ensemble_Bbox_Process
ensemble_bbox_tn90p
```

Parameters

- lat0 (float) – Minimum latitude.
- lat1 (float) – Maximum latitude.
- lon0 (float) – Minimum longitude.
• **lon1** *(float)* – Maximum longitude.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result.

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

class **finch.processes.xclim.tn90p_Ensemble_GridPoint_Process**

```python
ensemble_grid_point_tn90p unidecode.unidecode (v0.1)
```

**Parameters**

• **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
models (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

freq(string, optional) – Resampling frequency

output_format (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

output (application/x-netcdf, application/zip) – The format depends on the 'output_format' input parameter.

output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.tn90p_Ensemble_Polygon_Process
ensemble_polygon_tn90p unidecode.unidecode (v0.1)

Parameters

shape (application/vnd.geo+json) – Polygon contour, as a geojson string.

start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

dataset_name (['bccaqv2'], optional) – Name of the dataset from which to get netcdf files for inputs.

rcp (['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)

models (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

freq(string, optional) – Resampling frequency

output_format (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

output (application/x-netcdf, application/zip) – The format depends on the 'output_format' input parameter.

output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.tn_days_below_Ensemble_Bbox_Process
ensemble_bbox_tn_days_below unidecode.unidecode (v0.1)

Parameters

lat0 (float) – Minimum latitude.
• **lat1** *(float)* – Maximum latitude.
• **lon0** *(float)* – Minimum longitude.
• **lon1** *(float)* – Maximum longitude.
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
• **thresh** *(string, optional)* – Threshold
• **freq** *(string, optional)* – Resampling frequency
• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.
• **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.tn_days_below_Ensemble_GridPoint_Process
ensemble_grid_point_tn_days_below unidecode.unidecode (v0.1)

Parameters

• **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• **dataset_name** ('bccaqv2', optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ('rcp26', 'rcp45', 'rcp85') – Representative Concentration Pathway (RCP)

• **models** ('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3',..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** (string, optional) – Threshold

• **freq** (string, optional) – Resampling frequency

• **output_format** ('netcdf', 'csv'), optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

**class finch.processes.xclim.tn_days_below_Ensemble_Polygon_Process**

**ensemble_polygon_tn_days_below**

unidecode.unidecode (v0.1)

**Parameters**

• **shape** (application/vnd.geo+json) – Polygon contour, as a geojson string.

• **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ('bccaqv2'), optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ('rcp26', 'rcp45', 'rcp85') – Representative Concentration Pathway (RCP)

• **models** ('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3',..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** (string, optional) – Threshold

• **freq** (string, optional) – Resampling frequency

• **output_format** ('netcdf', 'csv'), optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.
• **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.tn_max_Ensemble_Bbox_Process

```python
ensemble_bbox_tn_max unidecode.unidecode (v0.1)
```

**Parameters**

- **lat0** (*float*) – Minimum latitude.
- **lat1** (*float*) – Maximum latitude.
- **lon0** (*float*) – Minimum longitude.
- **lon1** (*float*) – Maximum longitude.
- **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** (‘bccaqv2’, *optional*) – Name of the dataset from which to get netcdf files for inputs.
- **rcp** (‘rcp26’, ‘rcp45’, ‘rcp85’) – Representative Concentration Pathway (RCP)
- **freq** (*string, optional*) – Resampling frequency
- **output_format** (‘netcdf’, ‘csv’, *optional*) – Choose in which format you want to recieve the result

**Returns**

- **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.
- **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.tn_max_Ensemble_GridPoint_Process

```python
ensemble_grid_point_tn_max unidecode.unidecode (v0.1)
```

**Parameters**

- **lat** (*string*) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon** (*string*) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *(\'bccaqv2\', optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *(\'rcp26\', \'rcp45\', \'rcp85\')* – Representative Concentration Pathway (RCP)

• **models** *(\'24MODELS\', \'PCIC12\', \'BNU-ESM\', \'CCSM4\', \'CESM1-CAM5\', \'CNRM-CM5\', \'CSIRO-Mk3-6-0\', \'CanESM2\', \'FGOALS-g2\', \'GFDL-CM3\', ..), optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *(\'netcdf\', \'csv\'), optional)* – Choose in which format you want to receive the result.

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** finch.processes.xclim.tn_max_Ensemble_Polygon_Process

**ensemble_polygon_tn_max** unidecode.unidecode (v0.1)

**Parameters**

• **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *(\'bccaqv2\', optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *(\'rcp26\', \'rcp45\', \'rcp85\')* – Representative Concentration Pathway (RCP)

• **models** *(\'24MODELS\', \'PCIC12\', \'BNU-ESM\', \'CCSM4\', \'CESM1-CAM5\', \'CNRM-CM5\', \'CSIRO-Mk3-6-0\', \'CanESM2\', \'FGOALS-g2\', \'GFDL-CM3\', ..), optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *(\'netcdf\', \'csv\'), optional)* – Choose in which format you want to receive the result.
Returns

- **output** (*application/x-netcdf, application/zip*) – The format depends on the 'output_format' input parameter.
- **output_log** (*text/plain*) – Collected logs during process run.

```python
class finch.processes.xclim.tn_mean_Ensemble_Bbox_Process
ensemble_bbox_tn_mean unidecode.unidecode (v0.1)
```

Parameters

- **lat0** (*float*) – Minimum latitude.
- **lat1** (*float*) – Maximum latitude.
- **lon0** (*float*) – Minimum longitude.
- **lon1** (*float*) – Maximum longitude.
- **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day (%Y-%m-%d). Defaults to first day in file.
- **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day (%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** ("bccaqv2", optional) – Name of the dataset from which to get netcdf files for inputs.
- **rcp** (\{'rcp26', 'rcp45', 'rcp85'\}) – Representative Concentration Pathway (RCP)
- **models** (\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..\}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq** (*string, optional*) – Resampling frequency
- **output_format** (\{'netcdf', 'csv'\}, optional) – Choose in which format you want to receive the result

Returns

- **output** (*application/x-netcdf, application/zip*) – The format depends on the 'output_format' input parameter.
- **output_log** (*text/plain*) – Collected logs during process run.

```python
class finch.processes.xclim.tn_mean_Ensemble_GridPoint_Process
ensemble_grid_point_tn_mean unidecode.unidecode (v0.1)
```

Parameters

- **lat** (*string*) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon** (*string*) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models**  *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.tn_mean_Ensemble_Polygon_Process

    ensemble_polygon_tn_mean unidecode.unidecode (v0.1)

**Parameters**

• **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models**  *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

4.5. Processes
• **output_format** ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.tn_min_Ensemble_Bbox_Process

ensemble_bbox_tn_min unidecode.unidecode (v0.1)

**Parameters**

• **lat0** (float) – Minimum latitude.

• **lat1** (float) – Maximum latitude.

• **lon0** (float) – Minimum longitude.

• **lon1** (float) – Maximum longitude.

• **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• **models** ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (string, optional) – Resampling frequency

• **output_format** ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.tn_min_Ensemble_GridPoint_Process

ensemble_grid_point_tn_min unidecode.unidecode (v0.1)

**Parameters**

• **lat** (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to recieve the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the `output_format` input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

class Finch.processes.xclim.tn_min_Ensemble_Polygon_Process

    ensemble_polygon_tn_min unidecode.unidecode (v0.1)

**Parameters**

• **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

4.5. Processes
When calculating the ensemble, include only these models. By default, all 24 models are used.

- **freq (string, optional)** – Resampling frequency
- **output_format (["netcdf", "csv"], optional)** – Choose in which format you want to receive the result

**Returns**

- **output (application/x-netcdf, application/zip)** – The format depends on the ‘output_format’ input parameter.
- **output_log (text/plain)** – Collected logs during process run.

```python
class finch.processes.xclim.tropical_nights_Ensemble_Bbox_Process
enaense_bbox_tropical_nights unidecode.unidecode (v0.1)
```

**Parameters**

- **lat0 (float)** – Minimum latitude.
- **lat1 (float)** – Maximum latitude.
- **lon0 (float)** – Minimum longitude.
- **lon1 (float)** – Maximum longitude.
- **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name (["bccaqv2"], optional)** – Name of the dataset from which to get netcdf files for inputs.
- **rcp (["rcp26", "rcp45", "rcp85"])** – Representative Concentration Pathway (RCP)
- **models (["24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ...], optional)** – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh (string, optional)** – Threshold
- **freq (string, optional)** – Resampling frequency
- **output_format (["netcdf", "csv"], optional)** – Choose in which format you want to receive the result

**Returns**

- **output (application/x-netcdf, application/zip)** – The format depends on the ‘output_format’ input parameter.
- **output_log (text/plain)** – Collected logs during process run.

```python
class finch.processes.xclim.tropical_nights_Ensemble_GridPoint_Process
enaense_grid_point_tropical_nights unidecode.unidecode (v0.1)
```
Parameters

- **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
- **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
- **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **thresh** *(string, optional)* – Threshold
- **freq** *(string, optional)* – Resampling frequency
- **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

- **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.
- **output_log** *(text/plain)* – Collected logs during process run.

```python
class finch.processes.xclim.tropical_nights_Ensemble_Polygon_Process
ensemble_polygon_tropical_nights unidecode.unidecode (v0.1)
```

Parameters

- **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
Finch Documentation, Release 0.5.3

- `rcp` ('rcp26', 'rcp45', 'rcp85') – Representative Concentration Pathway (RCP)
- `models` ('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..), optional – When calculating the ensemble, include only these models. By default, all 24 models are used.
- `thresh` (string, optional) – Threshold
- `freq` (string, optional) – Resampling frequency
- `output_format` ('netcdf', 'csv'), optional – Choose in which format you want to receive the result

Returns
- `output` (application/x-netcdf, application/zip) – The format depends on the `output_format` input parameter.
- `output_log` (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.tx10p_Ensemble_Bbox_Process
ensemble_bbox_tx10p unidecode.unidecode (v0.1)
```

Parameters
- `lat0` (float) – Minimum latitude.
- `lat1` (float) – Maximum latitude.
- `lon0` (float) – Minimum longitude.
- `lon1` (float) – Maximum longitude.
- `start_date` (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- `end_date` (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- `ensemble_percentiles` (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma-separated list of integers.
- `dataset_name` ('bccaqv2'), optional – Name of the dataset from which to get netcdf files for inputs.
- `rcp` ('rcp26', 'rcp45', 'rcp85') – Representative Concentration Pathway (RCP)
- `models` ('24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..), optional – When calculating the ensemble, include only these models. By default, all 24 models are used.
- `freq` (string, optional) – Resampling frequency
- `output_format` ('netcdf', 'csv'), optional – Choose in which format you want to receive the result

Returns
• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

```python
class finch.processes.xclim.tx10p_Ensemble_GridPoint_Process
ensemble_grid_point_tx10p unidecode.unidecode (v0.1)
```

**Parameters**

• **lat** (*string*) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** (*string*) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)

• **models** ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (*string, optional*) – Resampling frequency

• **output_format** ({'netcdf', 'csv'}, optional) – Choose in which format you want to recieve the result

**Returns**

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

```python
class finch.processes.xclim.tx10p_Ensemble_Polygon_Process
ensemble_polygon_tx10p unidecode.unidecode (v0.1)
```

**Parameters**

• **shape** (*application/vnd.geo+json*) – Polygon contour, as a geojson string.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccagv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the `output_format` input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** `finch.processes.xclim.tx90p_Ensemble_Bbox_Process`  
`ensemble_bbox_tx90p` unidecode.unidecode (v0.1)

**Parameters**

• **lat0** *(float)* – Minimum latitude.

• **lat1** *(float)* – Maximum latitude.

• **lon0** *(float)* – Minimum longitude.

• **lon1** *(float)* – Maximum longitude.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccagv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result
Returns

- **output** (*application/x-netcdf, application/zip*) - The format depends on the 'output_format' input parameter.
- **output_log** (*text/plain*) - Collected logs during process run.

```python
class finch.processes.xclim.tx90p_Ensemble_GridPoint_Process
ensemble_grid_point_tx90p unidecode.unidecode (v0.1)
```

Parameters

- **lat** (*string*) - Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **lon** (*string*) - Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- **start_date** (*string, optional*) - Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** (*string, optional*) - Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** (*string, optional*) - Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** (*{'bccaqv2'}, optional*) - Name of the dataset from which to get netcdf files for inputs.
- **rcp** (*{'rcp26', 'rcp45', 'rcp85'}*) - Representative Concentration Pathway (RCP)
- **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) - When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq** (*string, optional*) - Resampling frequency
- **output_format** (*{'netcdf', 'csv'}, optional*) - Choose in which format you want to recieve the result

Returns

- **output** (*application/x-netcdf, application/zip*) - The format depends on the 'output_format' input parameter.
- **output_log** (*text/plain*) - Collected logs during process run.

```python
class finch.processes.xclim.tx90p_Ensemble_Polygon_Process
ensemble_polygon_tx90p unidecode.unidecode (v0.1)
```

Parameters

- **shape** (*application/vnd.geo+json*) - Polygon contour, as a geojson string.
- **start_date** (*string, optional*) - Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *("bccaqv2"): (optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *("rcp26", 'rcp45', 'rcp85')* – Representative Concentration Pathway (RCP)

• **models** *("24MODELS", 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..), optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *("netcdf", 'csv'), (optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** finch.processes.xclim.tx_days_above_Ensemble_Bbox_Process

*ensemble_bbox_tx_days_above* unidecode.unidecode (v0.1)

**Parameters**

• **lat0** *(float)* – Minimum latitude.

• **lat1** *(float)* – Maximum latitude.

• **lon0** *(float)* – Minimum longitude.

• **lon1** *(float)* – Maximum longitude.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *("bccaqv2"): (optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *("rcp26", 'rcp45', 'rcp85')* – Representative Concentration Pathway (RCP)

• **models** *("24MODELS", 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..), optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
• **thresh** (*string, optional*) – Threshold

• **freq** (*string, optional*) – Resampling frequency

• **output_format** (\{'netcdf', 'csv'}\, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.tx_days_above_Ensemble_GridPoint_Process
   
   ensemble_grid_point_tx_days_above unidecode.unidecode (v0.1)

   **Parameters**

• **lat** (*string*) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** (*string*) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (\{'bccaqv2'}\, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (\{'rcp26', 'rcp45', 'rcp85'}\) – Representative Concentration Pathway (RCP)

• **models** (\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..\})\, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** (*string, optional*) – Threshold

• **freq** (*string, optional*) – Resampling frequency

• **output_format** (\{'netcdf', 'csv'}\, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.tx_days_above_Ensemble_Polygon_Process
   
   ensemble_polygon_tx_days_above unidecode.unidecode (v0.1)

   **Parameters**

• **lat** (*string*) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** (*string*) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (\{'bccaqv2'}\, optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (\{'rcp26', 'rcp45', 'rcp85'}\) – Representative Concentration Pathway (RCP)

• **models** (\{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..\})\, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** (*string, optional*) – Threshold

• **freq** (*string, optional*) – Resampling frequency

• **output_format** (\{'netcdf', 'csv'}\, optional) – Choose in which format you want to receive the result

**Returns**

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.
• **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the 'output_format' input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** `finch.processes.xclim.tx_max_Engine_Bbox_Process`

**ensemble_bbox_tx_max** `unidecode.unidecode (v0.1)`

**Parameters**

• **lat0** *(float)* – Minimum latitude.

• **lat1** *(float)* – Maximum latitude.

• **lon0** *(float)* – Minimum longitude.

• **lon1** *(float)* – Maximum longitude.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
• **rcp** ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)

• **models** ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq**(string, optional) – Resampling frequency

• **output_format** ("netcdf", "csv"). optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

**class** finch.processes.xclim.tx_max_Ensemble_GridPoint_Process

**ensemble_grid_point_tx_max** unidecode.unidecode (v0.1)

**Parameters**

• **lat**(string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon**(string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date**(string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date**(string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles**(string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** ("bccaqv2"). optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)

• **models** ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq**(string, optional) – Resampling frequency

• **output_format** ("netcdf", "csv"). optional) – Choose in which format you want to receive the result

**Returns**

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.
class finch.processes.xclim.tx_max_Ensemble_Polygon_Process
ensemble_polygon_tx_max unidecode.unidecode (v0.1)

Parameters

- **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
- **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
- **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)
- **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
- **freq** *(string, optional)* – Resampling frequency
- **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

- **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.
- **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.tx_mean_Ensemble_Bbox_Process
ensemble_bbox_tx_mean unidecode.unidecode (v0.1)

Parameters

- **lat0** *(float)* – Minimum latitude.
- **lat1** *(float)* – Maximum latitude.
- **lon0** *(float)* – Minimum longitude.
- **lon1** *(float)* – Maximum longitude.
- **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• dataset_name ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.
• rcp ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)
• models ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• freq(string, optional) – Resampling frequency
• output_format ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

Returns
• output (application/x-netcdf, application/zip) – The format depends on the 'output_format' input parameter.
• output_log (text/plain) – Collected logs during process run.

class finch.processes.xclim.tx_mean_Ensemble_GridPoint_Process
ensemble_grid_point_tx_mean unidecode.unidecode (v0.1)

Parameters
• lat (string) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• lon (string) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
• start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• dataset_name ({'bccaqv2'}, optional) – Name of the dataset from which to get netcdf files for inputs.
• rcp ({'rcp26', 'rcp45', 'rcp85'}) – Representative Concentration Pathway (RCP)
• models ({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• freq(string, optional) – Resampling frequency
• output_format ({'netcdf', 'csv'}, optional) – Choose in which format you want to receive the result

Returns
• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.tx_mean_Ensemble_Polygon_Process

    ensemble_polygon_tx_mean unidecode.unidecode (v0.1)

Parameters

• **shape** (*application/vnd.geo+json*) – Polygon contour, as a geojson string.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (*string, optional*) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (*string, optional*) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (*{'bccaqv2'}, optional*) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (*{'rcp26', 'rcp45', 'rcp85'}*) – Representative Concentration Pathway (RCP)

• **models** (*{'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional*) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (*string, optional*) – Resampling frequency

• **output_format** (*{'netcdf', 'csv'}, optional*) – Choose in which format you want to receive the result

Returns

• **output** (*application/x-netcdf, application/zip*) – The format depends on the ‘output_format’ input parameter.

• **output_log** (*text/plain*) – Collected logs during process run.

class finch.processes.xclim.tx_min_Ensemble_Bbox_Process

    ensemble_bbox_tx_min unidecode.unidecode (v0.1)

Parameters

• **lat0** (*float*) – Minimum latitude.

• **lat1** (*float*) – Maximum latitude.

• **lon0** (*float*) – Minimum longitude.

• **lon1** (*float*) – Maximum longitude.

• **start_date** (*string, optional*) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name ("bccaqv2", optional)** – Name of the dataset from which to get netcdf files for inputs.

• **rcp ("rcp26", "rcp45", "rcp85")** – Representative Concentration Pathway (RCP)

• **models ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional** – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq (string, optional)** – Resampling frequency

• **output_format ("netcdf", "csv"), optional** – Choose in which format you want to receive the result

**Returns**

• **output (application/x-netcdf, application/zip)** – The format depends on the ‘output_format’ input parameter.

• **output_log (text/plain)** – Collected logs during process run.

**class** `finch.processes.xclim.tx_min_Ensemble_GridPoint_Process`

**ensemble_grid_point_tx_min** unidecode.unidecode (v0.1)

**Parameters**

• **lat (string)** – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon (string)** – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date (string, optional)** – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date (string, optional)** – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles (string, optional)** – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name ("bccaqv2"), optional** – Name of the dataset from which to get netcdf files for inputs.

• **rcp ("rcp26", "rcp45", "rcp85")** – Representative Concentration Pathway (RCP)

• **models ("24MODELS", "PCIC12", "BNU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional** – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq (string, optional)** – Resampling frequency
• **output_format** (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.tx_min_Ensemble_Polygon_Process

ensemble_polygon_tx_min unidecode.unidecode (v0.1)

Parameters

• **shape** (application/vnd.geo+json) – Polygon contour, as a geojson string.

• **start_date** (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** (['bccaqv2'], optional) – Name of the dataset from which to get netcdf files for inputs.

• **rcp** (['rcp26', 'rcp45', 'rcp85']) – Representative Concentration Pathway (RCP)

• **models** (['24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..], optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **freq** (string, optional) – Resampling frequency

• **output_format** (['netcdf', 'csv'], optional) – Choose in which format you want to receive the result

Returns

• **output** (application/x-netcdf, application/zip) – The format depends on the ‘output_format’ input parameter.

• **output_log** (text/plain) – Collected logs during process run.

class finch.processes.xclim.tx_tn_days_above_Ensemble_Bbox_Process

ensemble_bbox_tx_tn_days_above unidecode.unidecode (v0.1)

Parameters

• **lat0** (float) – Minimum latitude.

• **lat1** (float) – Maximum latitude.

• **lon0** (float) – Minimum longitude.

• **lon1** (float) – Maximum longitude.
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ...}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh_tasmin** *(string, optional)* – Threshold

• **thresh_tasmax** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to recieve the result

Returns

• **output** *(application/x-netcdf, application/zip)* – The format depends on the 'output_format' input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

class finch.processes.xclim.tx_tn_days_above_Ensemble_GridPoint_Process

ensemble_grid_point_tx_tn_days_above unidecode.unidecode (v0.1)

Parameters

• **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.
• `rcp` ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)

• `models` ("24MODELS", "PCIC12", "BUU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• `thresh_tasmin` (string, optional) – Threshold

• `thresh_tasmax` (string, optional) – Threshold

• `freq` (string, optional) – Resampling frequency

• `output_format` ("netcdf", "csv"), optional) – Choose in which format you want to receive the result

Returns

• `output` (application/x-netcdf, application/zip) – The format depends on the `output_format` input parameter.

• `output_log` (text/plain) – Collected logs during process run.

```python
class finch.processes.xclim.tx_tn_days_above_Ensemble_Polygon_Process
ensemble_polygon_tx_tn_days_above unidecode.unidecode (v0.1)
```

Parameters

• `shape` (application/vnd.geo+json) – Polygon contour, as a geojson string.

• `start_date` (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• `end_date` (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• `ensemble_percentiles` (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• `dataset_name` ("bccaqv2"), optional) – Name of the dataset from which to get netcdf files for inputs.

• `rcp` ("rcp26", "rcp45", "rcp85") – Representative Concentration Pathway (RCP)

• `models` ("24MODELS", "PCIC12", "BUU-ESM", "CCSM4", "CESM1-CAM5", "CNRM-CM5", "CSIRO-Mk3-6-0", "CanESM2", "FGOALS-g2", "GFDL-CM3", ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.

• `thresh_tasmin` (string, optional) – Threshold

• `thresh_tasmax` (string, optional) – Threshold

• `freq` (string, optional) – Resampling frequency

• `output_format` ("netcdf", "csv"), optional) – Choose in which format you want to receive the result

Returns
• **output** `application/x-netcdf, application/zip` – The format depends on the ‘output_format’ input parameter.

• **output_log** `text/plain` – Collected logs during process run.

```python
class finch.processes.xclim.wetdays_E ensemble_bbox_wetdays

Parameters

• lat0 (float) – Minimum latitude.
• lat1 (float) – Maximum latitude.
• lon0 (float) – Minimum longitude.
• lon1 (float) – Maximum longitude.
• start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day (%Y-%m-%d). Defaults to first day in file.
• end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day (%Y-%m-%d). Defaults to last day in file.
• ensemble_percentiles (string, optional) – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.
• dataset_name (’bccaqv2’, optional) – Name of the dataset from which to get netcdf files for inputs.
• rcp (’rcp26’, ’rcp45’, ’rcp85’) – Representative Concentration Pathway (RCP)
• models (’24MODELS’, ’PCIC12’, ’BNU-ESM’, ’CCSM4’, ’CESM1-CAM5’, ’CNRM-CM5’, ’CSIRO-Mk3-6-0’, ’CanESM2’, ”FGOALS-g2”, ”GFDL-CM3”, ..), optional) – When calculating the ensemble, include only these models. By default, all 24 models are used.
• thresh (string, optional) – Threshold
• freq (string, optional) – Resampling frequency
• output_format (’netcdf’, ’csv’, optional) – Choose in which format you want to receive the result

Returns

• **output** `application/x-netcdf, application/zip` – The format depends on the ‘output_format’ input parameter.

• **output_log** `text/plain` – Collected logs during process run.
```
```
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day (%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day (%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *(({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.

• **thresh** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

Returns

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

**class** *finch.processes.xclim.wetdays_Ensemble_Polygon_Process*

**ensemble_polygon_wetdays** *unidecode.unidecode (v0.1)*

**Parameters**

• **shape** *(application/vnd.geo+json)* – Polygon contour, as a geojson string.

• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day (%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day (%Y-%m-%d). Defaults to last day in file.

• **ensemble_percentiles** *(string, optional)* – Ensemble percentiles to calculate for input climate simulations. Accepts a comma separated list of integers.

• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

• **rcp** *({'rcp26', 'rcp45', 'rcp85'})* – Representative Concentration Pathway (RCP)

• **models** *(({'24MODELS', 'PCIC12', 'BNU-ESM', 'CCSM4', 'CESM1-CAM5', 'CNRM-CM5', 'CSIRO-Mk3-6-0', 'CanESM2', 'FGOALS-g2', 'GFDL-CM3', ..}, optional)* – When calculating the ensemble, include only these models. By default, all 24 models are used.
• **thresh** *(string, optional)* – Threshold

• **freq** *(string, optional)* – Resampling frequency

• **output_format** *("netcdf", 'csv'), optional)* – Choose in which format you want to receive the result

**Returns**

• **output** *(application/x-netcdf, application/zip)* – The format depends on the ‘output_format’ input parameter.

• **output_log** *(text/plain)* – Collected logs during process run.

### 4.5.3 Other Processes

**class** `finch.processes.xclim.BCCAQV2HeatWave`

`BCCAQv2_heat_wave_frequency_gridpoint` BCCAQv2 grid cell heat wave frequency computation***Deprecated*** to be removed in a future release (v0.1)

Compute heat wave frequency for all the BCCAQv2 datasets for a single grid cell.

**Parameters**

• **thresh_tasmin** *(string, optional)* – Threshold

• **thresh_tasmax** *(string, optional)* – Threshold

• **window** *(integer, optional)* – Window

• **freq** *(string, optional)* – Resampling frequency

• **lon** *(string)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lat** *(string)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **y0** *(integer, optional)* – Initial year for temporal subsetting. Defaults to first year in file.

• **y1** *(integer, optional)* – Final year for temporal subsetting. Defaults to last year in file.

• **output_format** *("netcdf", 'csv'), optional)* – Choose in which format you want to receive the result

**Returns output** – The format depends on the ‘output_format’ input parameter.

**Return type** application/x-netcdf, application/zip

**class** `finch.processes.xclim.SubsetBboxBCCAQV2Process`

`subset_ensemble_bbox_BCCAQv2` Subset of BCCAQv2 datasets, using a bounding box (v0.1)

For the BCCAQv2 datasets, return the data for which grid cells intersect the bounding box for each input dataset as well as the time range selected.

**Parameters**

• **variable** *("tasmin", 'tasmax', 'pr'), optional)* – Name of the variable in the NetCDF file.

• **rcp** *("rcp26", 'rcp45', 'rcp85'), optional)* – Representative Concentration Pathway (RCP)
• **lon0** *(float, optional)* – Minimum longitude.
• **lon1** *(float, optional)* – Maximum longitude.
• **lat0** *(float, optional)* – Minimum latitude.
• **lat1** *(float, optional)* – Maximum latitude.
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result.
• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

**Returns output** – The format depends on the ‘output_format’ input parameter.

**Return type** *application/x-netcdf, text/plain*

class finch.processes.xclim.SubsetBboxDatasetProcess

subset_bbox_dataset Subset of a dataset, using a bounding box (v0.1)

For the given dataset, return the data for which grid cells intersect the bounding box for each input dataset as well as the time range selected.

**Parameters**

• **variable** *({'tasmin', 'tasmax', 'pr'}, optional)* – Name of the variable in the NetCDF file.
• **rcp** *({'rcp26', 'rcp45', 'rcp85'}, optional)* – Representative Concentration Pathway (RCP)
• **lon0** *(float, optional)* – Minimum longitude.
• **lon1** *(float, optional)* – Maximum longitude.
• **lat0** *(float, optional)* – Minimum latitude.
• **lat1** *(float, optional)* – Maximum latitude.
• **start_date** *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• **end_date** *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
• **output_format** *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result.
• **dataset_name** *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

**Returns output** – The format depends on the ‘output_format’ input parameter.

**Return type** *application/x-netcdf, text/plain*
class Finch Documentation, Release 0.5.3

class finch.processes.xclim.SubsetBboxProcess

subset_bbox Subset with bounding box (v0.1)

Return the data for which grid cells intersect the bounding box for each input dataset as well as the time range selected.

Parameters

- resource (application/x-netcdf, application/x-ogc-dods) – NetCDF files, can be OPEnDAP urls.
- lon0 (float, optional) – Minimum longitude.
- lon1 (float, optional) – Maximum longitude.
- lat0 (float, optional) – Minimum latitude.
- lat1 (float, optional) – Maximum latitude.
- start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
- end_date (string, optional) – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.
- variable ("<property object at 0x7f8257b30130>", optional) – Name of the variable in the NetCDF file.

Returns

- output (application/x-netcdf) – netCDF output
- ref (application/metalink+xml; version=4.0) – Metalink file storing all references to output files.

class finch.processes.xclim.SubsetGridPointBCCAQV2Process

subset_ensemble_BCCAQV2 Subset of BCCAQv2 datasets grid cells using a list of coordinates (v0.1)

For the BCCAQv2 datasets, return the closest grid cell for each provided coordinates pair, for the time range selected.

Parameters

- variable ("tasmin", "tasmax", "pr"); optional) – Name of the variable in the NetCDF file.
- rcp ("rcp26", "rcp45", "rcp85"); optional) – Representative Concentration Pathway (RCP)
- lat (string, optional) – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- lon (string, optional) – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.
- lat0 (string, optional) – Latitude (deprecated, use ‘lat’).
- lon0 (string, optional) – Longitude (deprecated, use ‘lon’).
- start_date (string, optional) – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.
• `end_date` *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• `output_format` *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

• `dataset_name` *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

**Returns** *output* – The format depends on the `output_format` input parameter.

**Return type** `application/x-netcdf, text/plain`

```python
class finch.processes.xclim.SubsetGridPointDatasetProcess
subset_grid_point_dataset
Subset of grid cells from a dataset, using a list of coordinates (v0.1)
For the the given dataset, return the closest grid cell for each provided coordinates pair, for the time range selected.

Parameters

• `variable` *({'tasmin', 'tasmax', 'pr'}, optional)* – Name of the variable in the NetCDF file.

• `rcp` *({'rcp26', 'rcp45', 'rcp85'}, optional)* – Representative Concentration Pathway (RCP)

• `lat` *(string, optional)* – Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• `lon` *(string, optional)* – Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• `lat0` *(string, optional)* – Latitude (deprecated, use ‘lat’).

• `lon0` *(string, optional)* – Longitude (deprecated, use ‘lon’).

• `start_date` *(string, optional)* – Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• `end_date` *(string, optional)* – Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• `output_format` *({'netcdf', 'csv'}, optional)* – Choose in which format you want to receive the result

• `dataset_name` *({'bccaqv2'}, optional)* – Name of the dataset from which to get netcdf files for inputs.

**Returns** *output* – The format depends on the `output_format` input parameter.

**Return type** `application/x-netcdf, text/plain`
```

```python
class finch.processes.xclim.SubsetGridPointProcess
subset_gridpoint
Subset with a grid point (v0.2)
Return the data for which grid cells includes the point coordinates for each input dataset as well as the time range selected.

Parameters

```
• **resource** *(application/x-netcdf, application/x-ogc-dods) –* NetCDF files, can be OPeNDAP urls.

• **lon** *(string) –* Longitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **lat** *(string) –* Latitude coordinate. Accepts a comma separated list of floats for multiple grid cells.

• **start_date** *(string, optional) –* Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional) –* Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **variable** *(ipher property object at 0x7f8257b30130>, optional) –* Name of the variable in the NetCDF file.

**Returns**

• **output** *(application/x-netcdf) –* netCDF output

• **ref** *(application/metalink+xml; version=4.0) –* Metalink file storing all references to output files.

class finch.processes.xclim.\_SubsetPolygonProcess

subset\_polygon Subset with one or more polygons (v0.1)

Return the data for which grid cells center are within the polygon for each input dataset as well as the time range selected.

**Parameters**

• **resource** *(application/x-netcdf, application/x-ogc-dods) –* NetCDF files, can be OPeNDAP urls.

• **shape** *(application/vnd.geo+json) –* Polygon contour, as a geojson string.

• **start_date** *(string, optional) –* Initial date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to first day in file.

• **end_date** *(string, optional) –* Final date for temporal subsetting. Can be expressed as year (%Y), year-month (%Y-%m) or year-month-day(%Y-%m-%d). Defaults to last day in file.

• **variable** *(ipher property object at 0x7f8257b30130>, optional) –* Name of the variable in the NetCDF file.

**Returns**

• **output** *(application/x-netcdf) –* netCDF output

• **ref** *(application/metalink+xml; version=4.0) –* Metalink file storing all references to output files.
4.6 Changes

4.6.1 0.5.3 (2020-04-23)
- fix to chunk regions of subsetted files

4.6.2 0.5.2 (2020-03-25)
- fix to remove realization coordinate from ensembles
- added chunk datasets for local files also
- update xclim to == 0.15.2

4.6.3 0.5.1 (2020-03-18)
- fix local bccaq2 files filtering

4.6.4 0.5.0 (2020-03-18)
- update xclim to 0.15.0
- add french translation of processes abstract and descriptions

4.6.5 0.4.1 (2020-03-12)
- fix #103 (drs_filename) add defaults when project_id is unknown
- drs_filenames: use dash instead of underscores in variable names
- fix #80 frequency attrs of computed datasets

4.6.6 0.4.0 (2020-03-10)
- Add ensembles processes
- Allow ensemble process to specify which models are included
- Accept multiple files for processing
- Update from latest cookie-cutter template
- Add grid point indicator processes
- Add ensemble bbox processes
- Add support for percentiles inputs
- Update xclim to 0.14
- Pin PyWPS to 4.2.4
- Add DODS to supported formats for resources
4.6.7 0.3.x (2020-01-17)

- Extract common inputs and outputs to wpsio.py
- Speed up CSV creation
- Explicitly close thread pool
- Tests for CSV conversion
- Added subset_shape process
- Pin PyWPS to ~4.2.3
- Add start and end date to bccaqv2 subset
- Identifier DAP link by header
- Datetime fix when replacing hour to 12
- deprecate lon0 and lat0 for SubsetGridPointBCCAQV2Process
- change point subset processes to accept a comma separated list of floats for multiple grid cells

4.6.8 0.2.7 (2019-12-09)

- Fix for segmentation fault in libnetcdf (pin version to 4.6.2 until a fix is released)

4.6.9 0.2.6 (2019-12-04)

- Notebooks are tested by Travis-CI
- Bug fix
- Update xclim to >= 0.12.2
- Update pywps to > 4.2.3

4.6.10 0.2.5 (2019-10-03)

- Add test for DAP input to subsetting
- Update notebook to run on the Jenkins test suite

4.6.11 0.2.3 (2019-05-27)

- Allow creating CSV output
- Keep global attributes when computing indices
- Add BCCAQV2HeatWave process
- Add basic usage notebook
4.6.12 0.2.1 (2019-05-06)

- Require Python>=3.6
- Fix percentages in status update
- Improve loggin

4.6.13 0.2 (2019-05-02)

- Added subset_gridpoint process
- Support DAP links
- Added bounding box subsetting
- Threshold arguments passed as strings with units
- Added test for heat_wave_frequency
- Use sentry to monitor error messages
- Include Dockerfile
- Use processes instead of threads

4.6.14 0.1 (2018-11-15)

- First release.
INDICES AND TABLES

• genindex
• modindex
• search
INDEX

B

BCCAQV2HeatWave (class in finch.processes.xclim), 131

cdd_Ensemble_Bbox_Process (class in finch.processes.xclim), 36
cdd_Ensemble_GridPoint_Process (class in finch.processes.xclim), 36
cdd_Ensemble_Polygon_Process (class in finch.processes.xclim), 37
cdd_Indicator_Process (class in finch.processes.xclim), 20
cold_spell_days_Ensemble_Bbox_Process (class in finch.processes.xclim), 38
cold_spell_days_Ensemble_GridPoint_Process (class in finch.processes.xclim), 38
cold_spell_days_Ensemble_Polygon_Process (class in finch.processes.xclim), 39
cold_spell_days_Indicator_Process (class in finch.processes.xclim), 21
cold_spell_duration_index_Ensemble_Bbox_Process (class in finch.processes.xclim), 40
cold_spell_duration_index_Ensemble_GridPoint_Process (class in finch.processes.xclim), 41
cold_spell_duration_index_Ensemble_Polygon_Process (class in finch.processes.xclim), 41
cold_spell_duration_index_Indicator_Process (class in finch.processes.xclim), 21
consecutive_frost_days_Ensemble_Bbox_Process (class in finch.processes.xclim), 42
consecutive_frost_days_Ensemble_GridPoint_Process (class in finch.processes.xclim), 43
consecutive_frost_days_Ensemble_Polygon_Process (class in finch.processes.xclim), 43
cold_indicators_Ensemble_Bbox_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_GridPoint_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_Polygon_Process (class in finch.processes.xclim), 22

cdd_Ensemble_Bbox_Process (class in finch.processes.xclim), 36

cdd_Ensemble_GridPoint_Process (class in finch.processes.xclim), 36

cdd_Ensemble_Polygon_Process (class in finch.processes.xclim), 37

cdd_Indicator_Process (class in finch.processes.xclim), 20

cold_spell_days_Ensemble_Bbox_Process (class in finch.processes.xclim), 38
cold_spell_days_Ensemble_GridPoint_Process (class in finch.processes.xclim), 38
cold_spell_days_Ensemble_Polygon_Process (class in finch.processes.xclim), 39
cold_spell_days_Indicator_Process (class in finch.processes.xclim), 21
cold_spell_duration_index_Ensemble_Bbox_Process (class in finch.processes.xclim), 40
cold_spell_duration_index_Ensemble_GridPoint_Process (class in finch.processes.xclim), 41
cold_spell_duration_index_Ensemble_Polygon_Process (class in finch.processes.xclim), 41
cold_spell_duration_index_Indicator_Process (class in finch.processes.xclim), 21
consecutive_frost_days_Ensemble_Bbox_Process (class in finch.processes.xclim), 42
consecutive_frost_days_Ensemble_GridPoint_Process (class in finch.processes.xclim), 43
consecutive_frost_days_Ensemble_Polygon_Process (class in finch.processes.xclim), 43
consecutive_frost_days_Indicator_Process (class in finch.processes.xclim), 21

cold_indicators_Ensemble_Bbox_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_GridPoint_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_Polygon_Process (class in finch.processes.xclim), 22

cold_indicators_Ensemble_Bbox_Process (class in finch.processes.xclim), 22

cold_indicators_Ensemble_GridPoint_Process (class in finch.processes.xclim), 22

cold_indicators_Ensemble_Polygon_Process (class in finch.processes.xclim), 22

cold_indicators_Ensemble_Bbox_Process (class in finch.processes.xclim), 22

cold_indicators_Ensemble_GridPoint_Process (class in finch.processes.xclim), 22

cold_indicators_Ensemble_Polygon_Process (class in finch.processes.xclim), 22

D

dlyfrzthw_Ensemble_Bbox_Process (class in finch.processes.xclim), 48
dlyfrzthw_Ensemble_GridPoint_Process (class in finch.processes.xclim), 49
dlyfrzthw_Ensemble_Polygon_Process (class in finch.processes.xclim), 50
dlyfrzthw_Indicator_Process (class in finch.processes.xclim), 22
dlyfrzthw_Indicator_Process (class in finch.processes.xclim), 22

dlyfrzthw_Ensemble_Bbox_Process (class in finch.processes.xclim), 48

dlyfrzthw_Ensemble_GridPoint_Process (class in finch.processes.xclim), 49

dlyfrzthw_Ensemble_Polygon_Process (class in finch.processes.xclim), 50
dlyfrzthw_Indicator_Process (class in finch.processes.xclim), 22
dlyfrzthw_Indicator_Process (class in finch.processes.xclim), 22

E

cold_indicators_Ensemble_Bbox_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_GridPoint_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_Polygon_Process (class in finch.processes.xclim), 22

cold_indicators_Ensemble_Bbox_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_GridPoint_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_Polygon_Process (class in finch.processes.xclim), 22

cold_indicators_Ensemble_Bbox_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_GridPoint_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_Polygon_Process (class in finch.processes.xclim), 22

cold_indicators_Ensemble_Bbox_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_GridPoint_Process (class in finch.processes.xclim), 22
cold_indicators_Ensemble_Polygon_Process (class in finch.processes.xclim), 22

141
etr_Ensemble_GridPoint_Process (class in finch.processes.xclim), 55
etr_Ensemble_Polygon_Process (class in finch.processes.xclim), 56
etr_Indicator_Process (class in finch.processes.xclim), 23
freshet_start_Ensemble_Bbox_Process (class in finch.processes.xclim), 57
freshet_start_Ensemble_GridPoint_Process (class in finch.processes.xclim), 58
freshet_start_Ensemble_Polygon_Process (class in finch.processes.xclim), 58
freshet_start_Indicator_Process (class in finch.processes.xclim), 24
frost_days_Ensemble_Bbox_Process (class in finch.processes.xclim), 59
frost_days_Ensemble_GridPoint_Process (class in finch.processes.xclim), 60
frost_days_Ensemble_Polygon_Process (class in finch.processes.xclim), 60
frost_days_Indicator_Process (class in finch.processes.xclim), 24
heat_wave_frequency_Ensemble_Bbox_Process (class in finch.processes.xclim), 63
heat_wave_frequency_Ensemble_GridPoint_Process (class in finch.processes.xclim), 64
heat_wave_frequency_Ensemble_Polygon_Process (class in finch.processes.xclim), 65
heat_wave_frequency_Indicator_Process (class in finch.processes.xclim), 25
heat_wave_index_Ensemble_Bbox_Process (class in finch.processes.xclim), 65
heat_wave_index_Ensemble_GridPoint_Process (class in finch.processes.xclim), 66
heat_wave_index_Ensemble_Polygon_Process (class in finch.processes.xclim), 67
heat_wave_index_Indicator_Process (class in finch.processes.xclim), 25
heat_wave_max_length_Ensemble_Bbox_Process (class in finch.processes.xclim), 68
heat_wave_max_length_Ensemble_GridPoint_Process (class in finch.processes.xclim), 69
heat_wave_max_length_Indicator_Process (class in finch.processes.xclim), 26
heat_wave_total_length_Ensemble_Bbox_Process (class in finch.processes.xclim), 70
heat_wave_total_length_Ensemble_GridPoint_Process (class in finch.processes.xclim), 71
heat_wave_total_length_Ensemble_Polygon_Process (class in finch.processes.xclim), 72
heat_wave_total_length_Indicator_Process (class in finch.processes.xclim), 26
ice_days_Ensemble_Bbox_Process (class in finch.processes.xclim), 74
ice_days_Ensemble_GridPoint_Process (class in finch.processes.xclim), 75
ice_days_Ensemble_Polygon_Process (class in finch.processes.xclim), 76
ice_days_Indicator_Process (class in finch.processes.xclim), 27
liquidprcptot_Ensemble_Bbox_Process (class in finch.processes.xclim), 76
liquidprcptot_Ensemble_GridPoint_Process (class in finch.processes.xclim), 77
liquidprcptot_Ensemble_Polygon_Process (class in finch.processes.xclim), 78
liquidprcptot_Indicator_Process (class in finch.processes.xclim), 27
max_n_day_precipitation_amount_Ensemble_Bbox_Process (class in finch.processes.xclim), 79
max_n_day_precipitation_amount_Ensemble_GridPoint_Process (class in finch.processes.xclim), 79
max_n_day_precipitation_amount_Ensemble_Polygon_Process (class in finch.processes.xclim), 80
max_n_day_precipitation_amount_Indicator_Process (class in finch.processes.xclim), 27
prcptot_Ensemble_Bbox_Process (class in finch.processes.xclim), 81
prcptot_Ensemble_GridPoint_Process (class in finch.processes.xclim), 81
prcptot_Ensemble_Polygon_Process (class in finch.processes.xclim), 82
prcptot_Indicator_Process (class in finch.processes.xclim), 28

rain_frzgr_Ensemble_Bbox_Process (class in finch.processes.xclim), 83
rain_frzgr_Ensemble_GridPoint_Process (class in finch.processes.xclim), 83
rain_frzgr_Ensemble_Polygon_Process (class in finch.processes.xclim), 84
rain_frzgr_Indicator_Process (class in finch.processes.xclim), 28

rx1day_Ensemble_Bbox_Process (class in finch.processes.xclim), 85
rx1day_Ensemble_GridPoint_Process (class in finch.processes.xclim), 86
rx1day_Ensemble_Polygon_Process (class in finch.processes.xclim), 86
rx1day_Indicator_Process (class in finch.processes.xclim), 28

sdii_Ensemble_Bbox_Process (class in finch.processes.xclim), 87
sdii_Ensemble_GridPoint_Process (class in finch.processes.xclim), 88
sdii_Ensemble_Polygon_Process (class in finch.processes.xclim), 88
sdii_Indicator_Process (class in finch.processes.xclim), 29
solidprcptot_Ensemble_Bbox_Process (class in finch.processes.xclim), 89
solidprcptot_Ensemble_GridPoint_Process (class in finch.processes.xclim), 90
solidprcptot_Ensemble_Polygon_Process (class in finch.processes.xclim), 90
solidprcptot_Indicator_Process (class in finch.processes.xclim), 29

SubsetBboxBCCAQV2Process (class in finch.processes.xclim), 131
SubsetBboxDatasetProcess (class in finch.processes.xclim), 132
SubsetBboxProcess (class in finch.processes.xclim), 132
SubsetGridPointBCCAQV2Process (class in finch.processes.xclim), 133
SubsetGridPointDatasetProcess (class in finch.processes.xclim), 134
SubsetGridPointProcess (class in finch.processes.xclim), 134
SubsetPolygonProcess (class in finch.processes.xclim), 135

tg10p_Ensemble_Bbox_Process (class in finch.processes.xclim), 91
tg10p_Ensemble_GridPoint_Process (class in finch.processes.xclim), 92
tg10p_Ensemble_Polygon_Process (class in finch.processes.xclim), 92
tg10p_Indicator_Process (class in finch.processes.xclim), 29
tg90p_Ensemble_Bbox_Process (class in finch.processes.xclim), 93
tg90p_Ensemble_GridPoint_Process (class in finch.processes.xclim), 94
tg90p_Ensemble_Polygon_Process (class in finch.processes.xclim), 95
tg90p_Indicator_Process (class in finch.processes.xclim), 30
tg_Ensemble_Bbox_Process (class in finch.processes.xclim), 95
tg_Ensemble_GridPoint_Process (class in finch.processes.xclim), 96
tg_Ensemble_Polygon_Process (class in finch.processes.xclim), 97
tg_Indicator_Process (class in finch.processes.xclim), 30
tg_mean_Ensemble_Bbox_Process (class in finch.processes.xclim), 97
tg_mean_Ensemble_GridPoint_Process (class in finch.processes.xclim), 98
tg_mean_Ensemble_Polygon_Process (class in finch.processes.xclim), 99
tgmean_Indicator_Process (class in finch.processes.xclim), 30	n10p_Ensemble_Bbox_Process (class in finch.processes.xclim), 99	n10p_Ensemble_GridPoint_Process (class in finch.processes.xclim), 100	n10p_Ensemble_Polygon_Process (class in finch.processes.xclim), 101	n10p_Indicator_Process (class in finch.processes.xclim), 31
tn90p_Ensemble_Bbox_Process (class in finch.processes.xclim), 101	nn90p_Ensemble_GridPoint_Process (class in finch.processes.xclim), 102	nn90p_Ensemble_Polygon_Process (class in finch.processes.xclim), 103
tn90p_Indicator_Process (class in finch.processes.xclim), 31

tx90p_Indicator_Process (class in finch.processes.xclim), 117

tx90p_Ensemble_Polygon_Process (class in finch.processes.xclim), 117

tn_days_below_Ensemble_Bbox_Process (class in finch.processes.xclim), 103

tn_days_below_Ensemble_GridPoint_Process (class in finch.processes.xclim), 104

tn_days_below_Ensemble_Polygon_Process (class in finch.processes.xclim), 105

tn_days_below_Indicator_Process (class in finch.processes.xclim), 31

tn_max_Ensemble_Bbox_Process (class in finch.processes.xclim), 106

tn_max_Ensemble_GridPoint_Process (class in finch.processes.xclim), 106

tn_max_Ensemble_Polygon_Process (class in finch.processes.xclim), 107

tn_max_Indicator_Process (class in finch.processes.xclim), 32

tn_min_Ensemble_Bbox_Process (class in finch.processes.xclim), 110

tn_min_Ensemble_GridPoint_Process (class in finch.processes.xclim), 110

tn_min_Ensemble_Polygon_Process (class in finch.processes.xclim), 110

tn_min_Indicator_Process (class in finch.processes.xclim), 32

tropical_nights_Ensemble_Bbox_Process (class in finch.processes.xclim), 112

tropical_nights_Ensemble_GridPoint_Process (class in finch.processes.xclim), 112

tropical_nights_Ensemble_Polygon_Process (class in finch.processes.xclim), 113

tropical_nights_Indicator_Process (class in finch.processes.xclim), 33

tx10p_Ensemble_Bbox_Process (class in finch.processes.xclim), 114

tx10p_Ensemble_GridPoint_Process (class in finch.processes.xclim), 115

tx10p_Ensemble_Polygon_Process (class in finch.processes.xclim), 115

tx10p_Indicator_Process (class in finch.processes.xclim), 33

tx90p_Ensemble_Bbox_Process (class in finch.processes.xclim), 116

tx90p_Ensemble_GridPoint_Process (class in finch.processes.xclim), 117

tx90p_Ensemble_Polygon_Process (class in finch.processes.xclim), 117

W

wetdays_Ensemble_Bbox_Process (class in finch.processes.xclim), 129
wetdays_Ensemble_GridPoint_Process (class in finch.processes.xclim), 129
wetdays_Ensemble_Polygon_Process (class in finch.processes.xclim), 130
wetdays_Indicator_Process (class in finch.processes.xclim), 35

Index