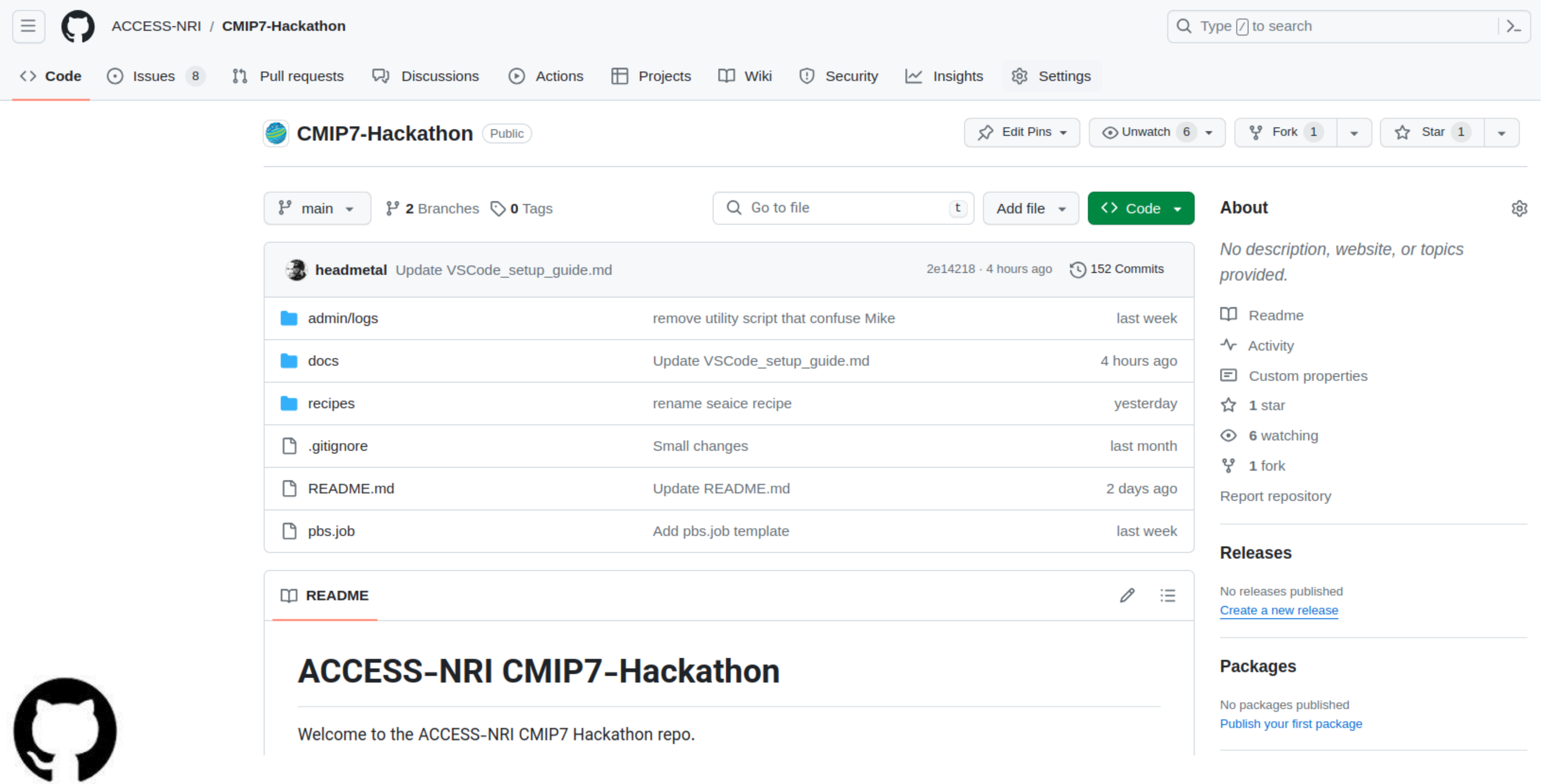
CMIP7 Evaluation Hackathon

The Australian Earth System Simulator
National Research Infrastructure (ACCESS-NRI)

Model Evaluation and Diagnostics

Romain.beucher@anu.edu.au

Everything is on GitHub



headmetal Update VSCode_setup_guide.md 2e14218 · 4 hours ago 152 Commits

admin/logs	remove utility script that confuse Mike	last week
docs	Update VSCode_setup_guide.md	4 hours ago
recipes	rename seaice recipe	yesterday
.gitignore	Small changes	last month
README.md	Update README.md	2 days ago
pbs.job	Add pbs.job template	last week

README

ACCESS-NRI CMIP7-Hackathon

Welcome to the ACCESS-NRI CMIP7 Hackathon repo.

About

No description, website, or topics provided.

- Readme
- Activity
- Custom properties
- 1 star
- 6 watching
- 1 fork

Report repository

Releases

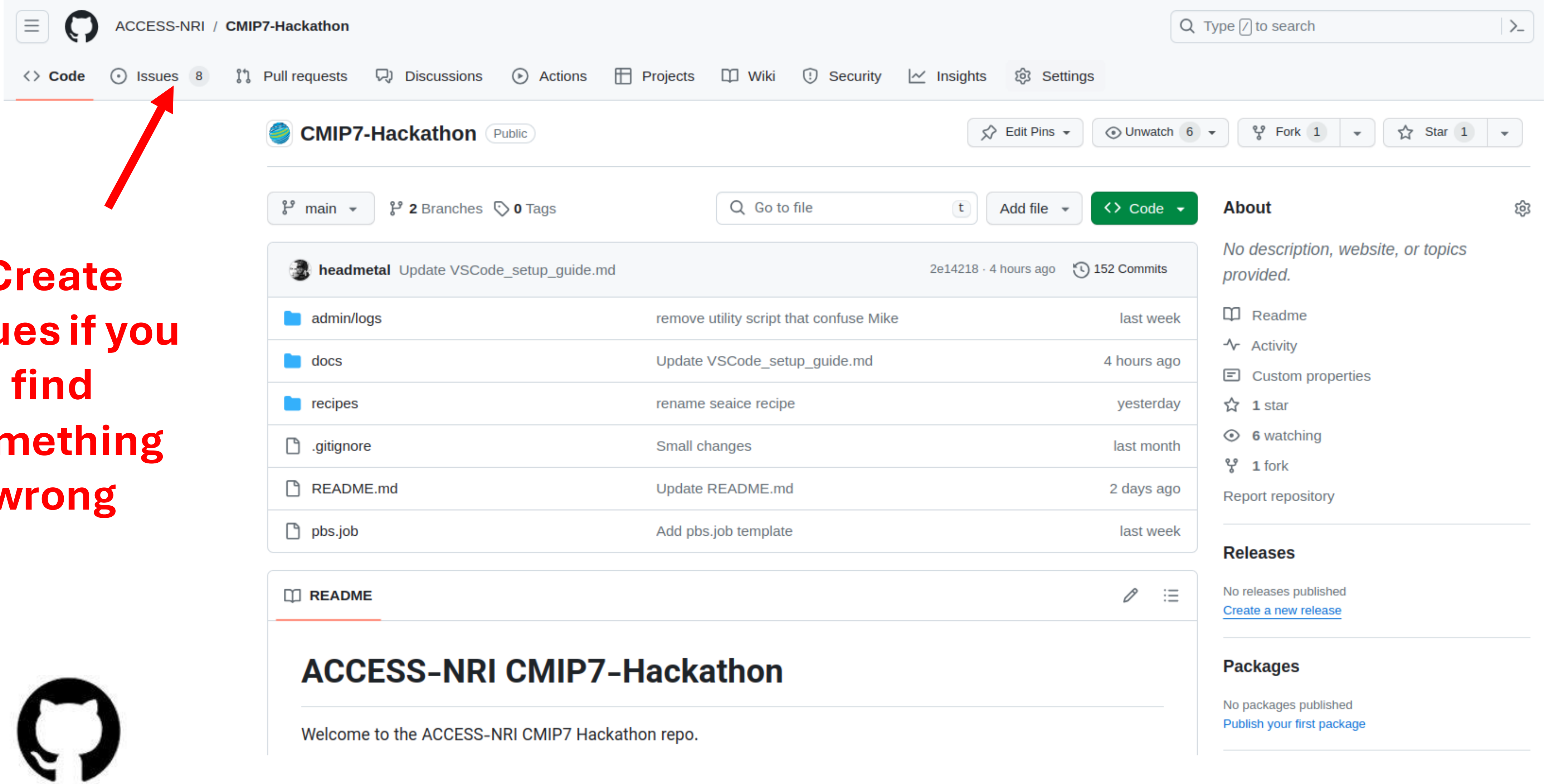
No releases published
Create a new release

Packages

No packages published
Publish your first package

https://github.com/ACCESS-NRI/CMIP7-Hackathon

Everything is on GitHub



headmetal Update VSCode_setup_guide.md 2e14218 · 4 hours ago 152 Commits

admin/logs	remove utility script that confuse Mike	last week
docs	Update VSCode_setup_guide.md	4 hours ago
recipes	rename seaice recipe	yesterday
.gitignore	Small changes	last month
README.md	Update README.md	2 days ago
pbs.job	Add pbs.job template	last week

README

ACCESS-NRI CMIP7-Hackathon

Welcome to the ACCESS-NRI CMIP7 Hackathon repo.

About

No description, website, or topics provided.

Readme
Activity
Custom properties
1 star
6 watching
1 fork
Report repository

Releases

No releases published
Create a new release

Packages

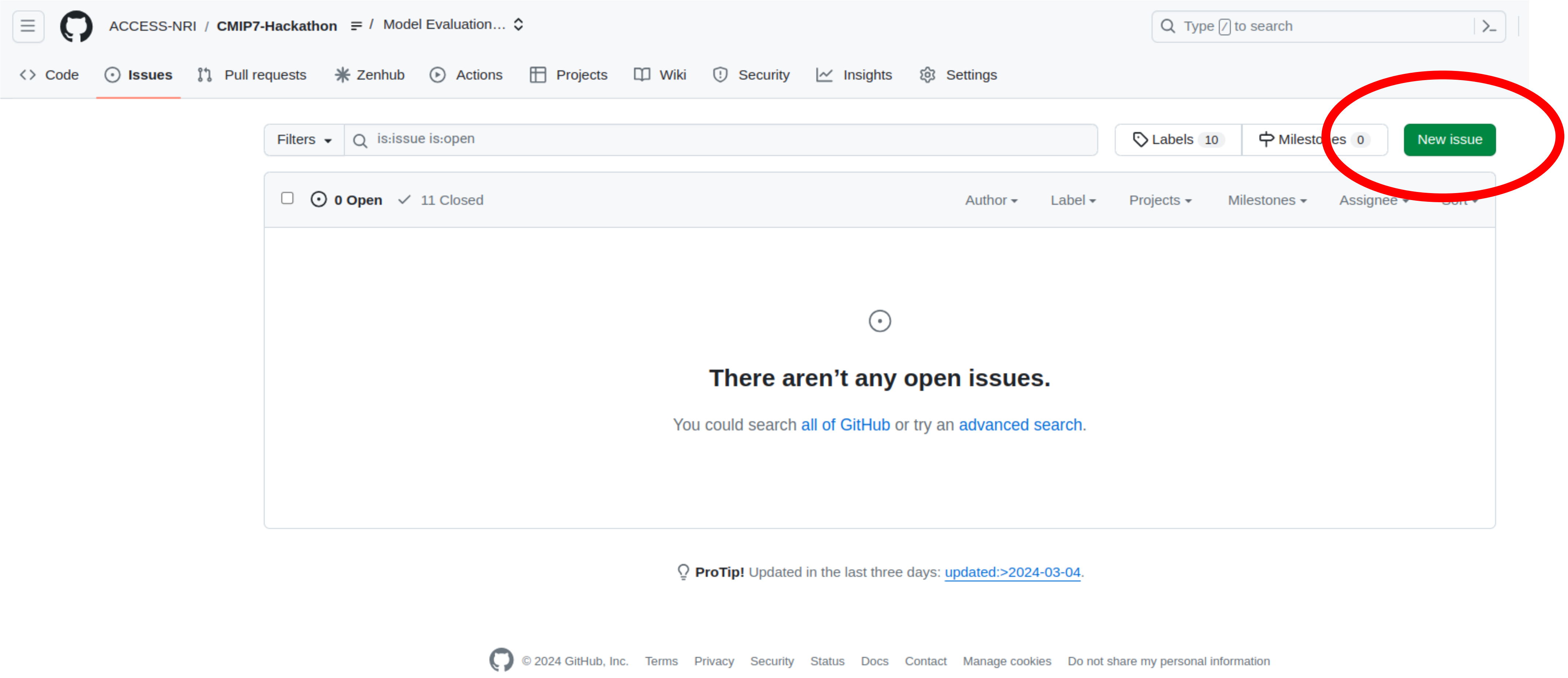
No packages published
Publish your first package

Create issues if you find something wrong

GitHub

https://github.com/ACCESS-NRI/CMIP7-Hackathon

Everything is on GitHub



There aren't any open issues.

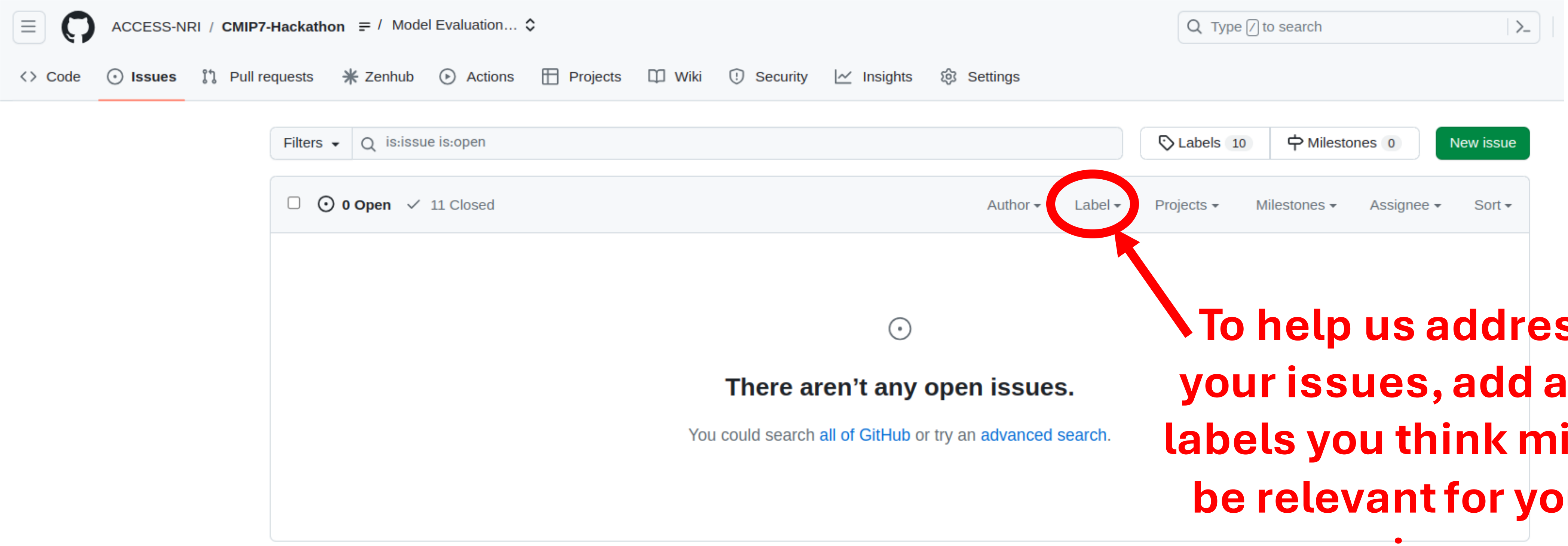
You could search all of GitHub or try an advanced search.

ProTip! Updated in the last three days: updated:>2024-03-04.

© 2024 GitHub, Inc. Terms Privacy Security Status Docs Contact Manage cookies Do not share my personal information

https://github.com/ACCESS-NRI/CMIP7-Hackathon/issues

Labeling GitHub issues



There aren't any open issues.

You could search all of GitHub or try an advanced search.

To help us address your issues, add any labels you think might be relevant for your new issue

Example labels:

By scientific domain

Ocean & Ice

General Diagnostics & Coupled

Land & Carbon Cycle

Atmosphere

By type of request

enhancement

bug

help wanted

question

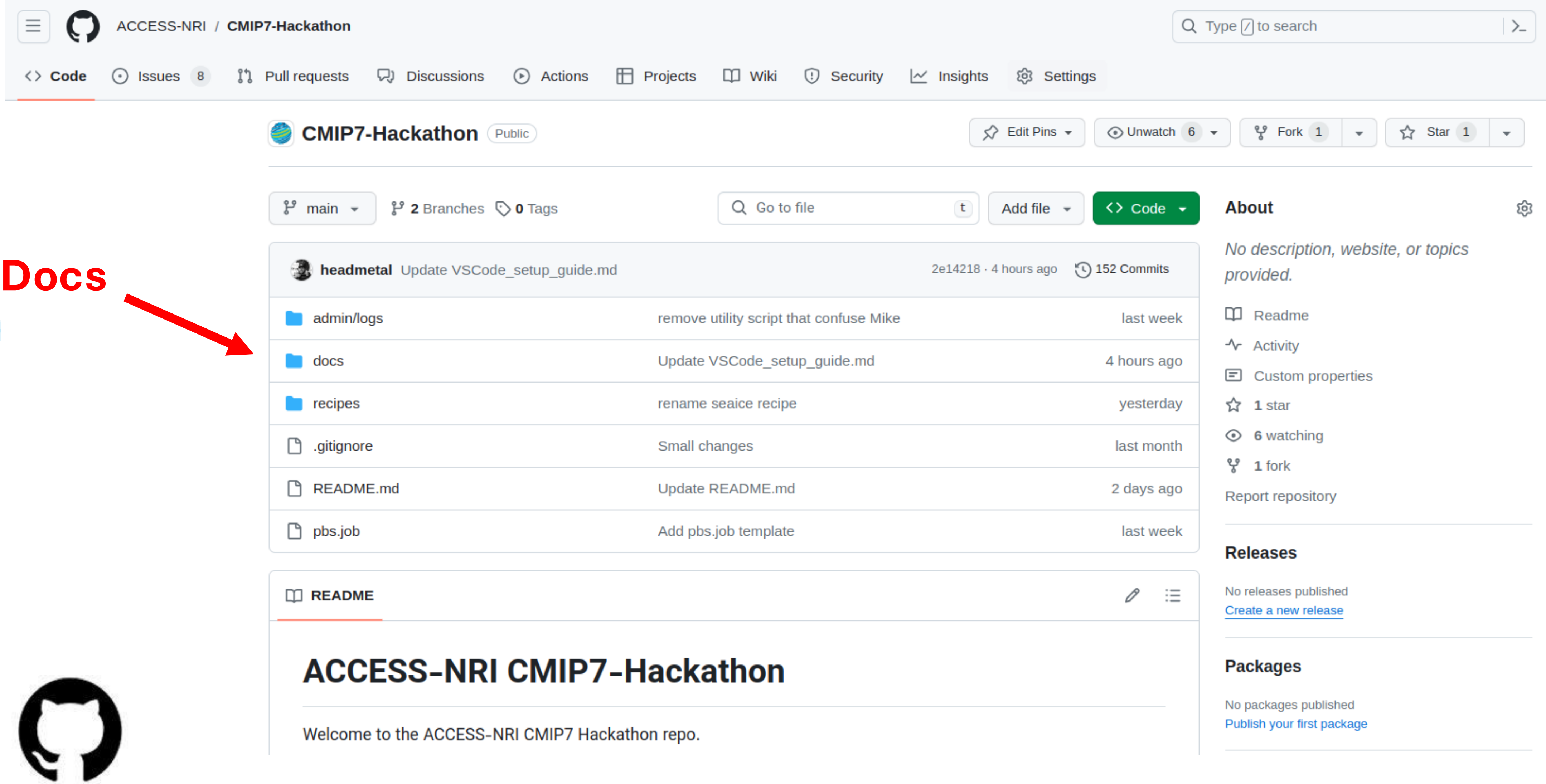
Other

Observational data

Or if you're not sure which label(s) to add, leave them blank and the ACCESS NRI team will add the appropriate ones 🙂

Everything is on GitHub

Docs



headmetal Update VSCode_setup_guide.md 2e14218 · 4 hours ago 152 Commits

admin/logs	remove utility script that confuse Mike	last week
docs	Update VSCode_setup_guide.md	4 hours ago
recipes	rename seaice recipe	yesterday
.gitignore	Small changes	last month
README.md	Update README.md	2 days ago
pbs.job	Add pbs.job template	last week

No description, website, or topics provided.

No releases published
Create a new release

No packages published
Publish your first package

ACCESS-NRI CMIP7-Hackathon

Welcome to the ACCESS-NRI CMIP7 Hackathon repo.

GitHub

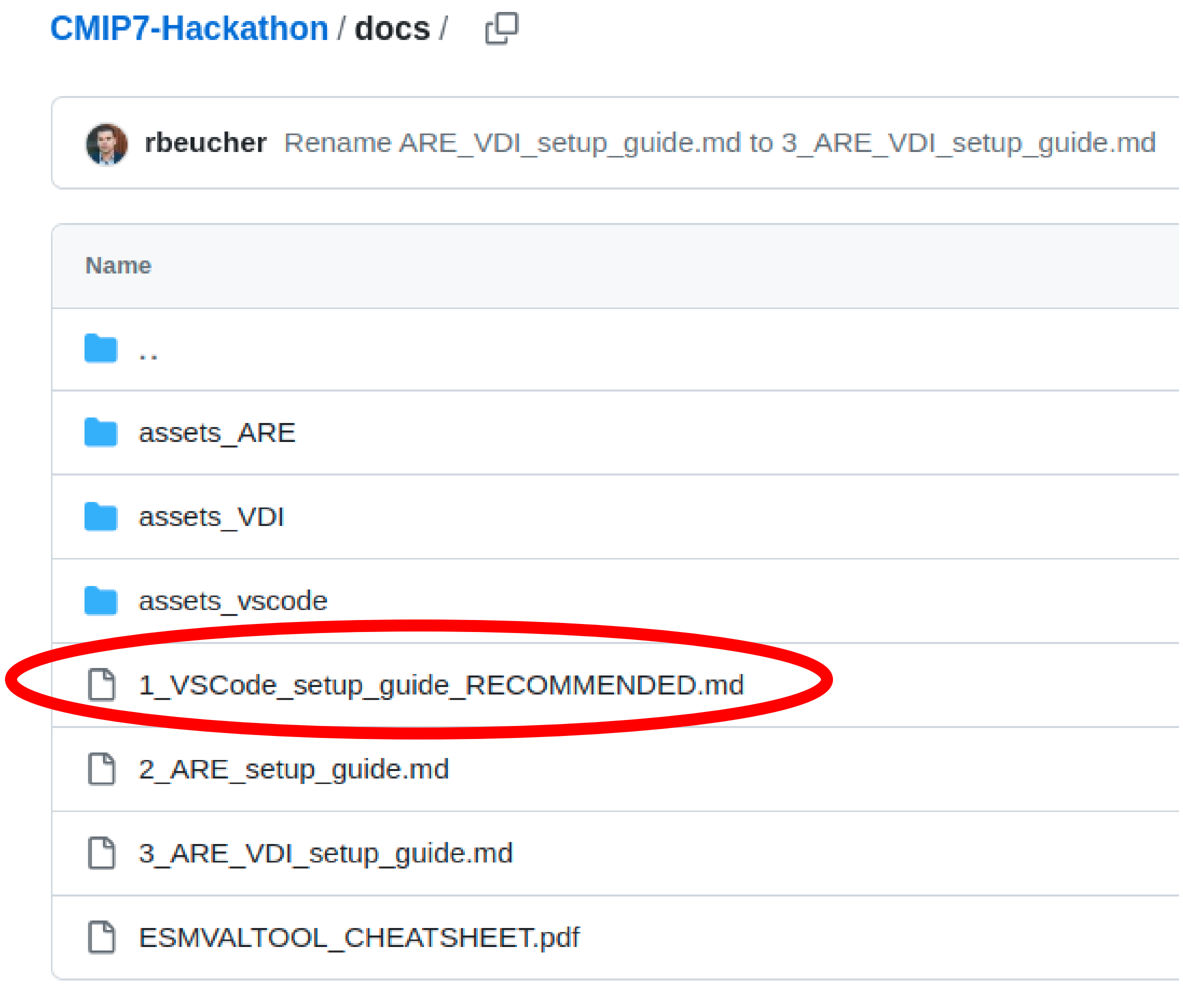
https://github.com/ACCESS-NRI/CMIP7-Hackathon

Getting set up

Docs

Visual Studio Code

Highly recommended!!



CMIP7-Hackathon / docs /

rbeucher Rename ARE_VDI_setup_guide.md to 3_ARE_VDI_setup_guide.md

Name
..
assets_ARE
assets_VDI
assets_vscode
1_VSCode_setup_guide_RECOMMENDED.md
2_ARE_setup_guide.md
3_ARE_VDI_setup_guide.md
ESMVALTOOL_CHEATSHEET.pdf

https://github.com/ACCESS-NRI/CMIP7-Hackathon/tree/main/docs

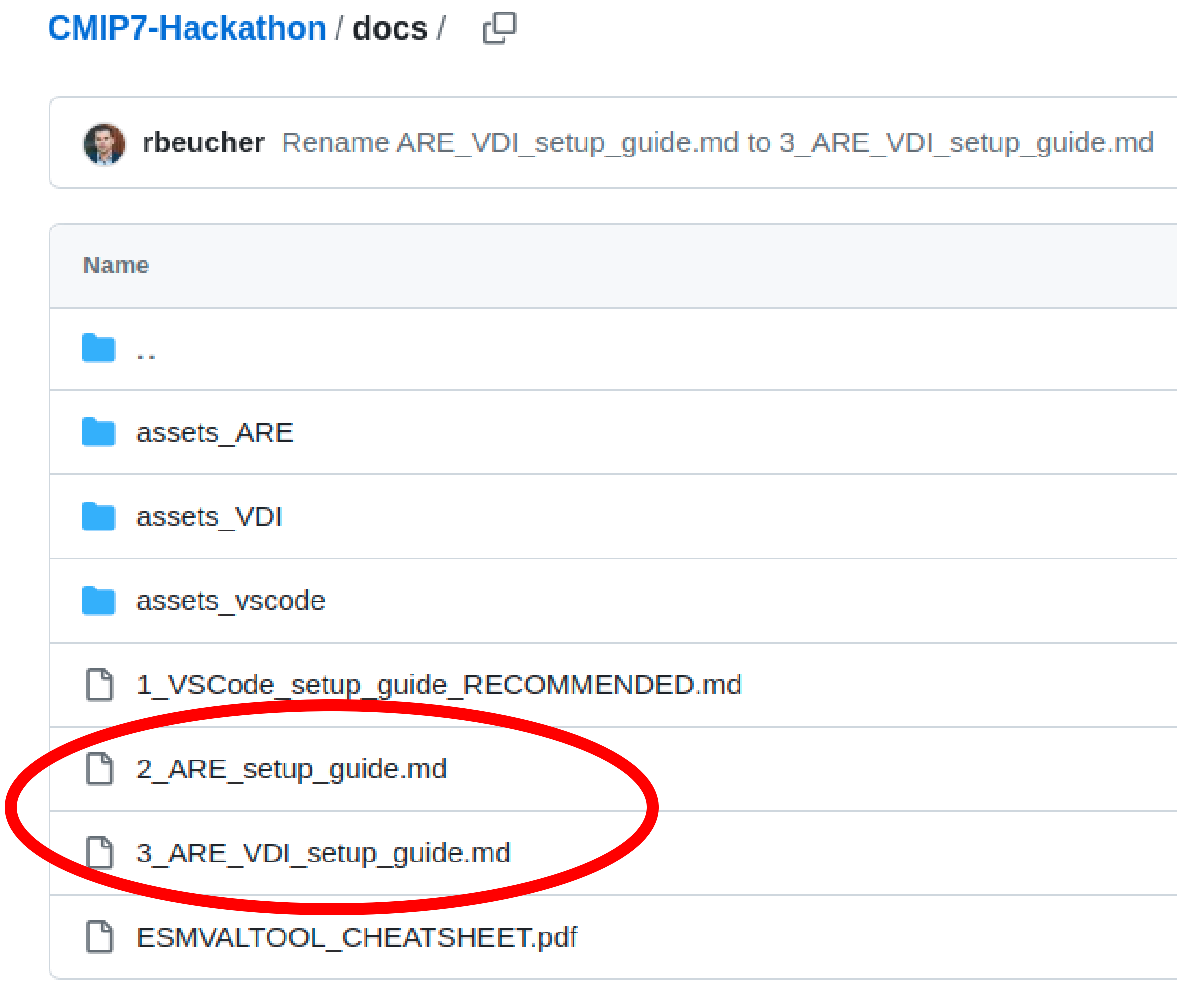
Getting set up

Docs

are

australian research environment

NCI AUSTRALIA



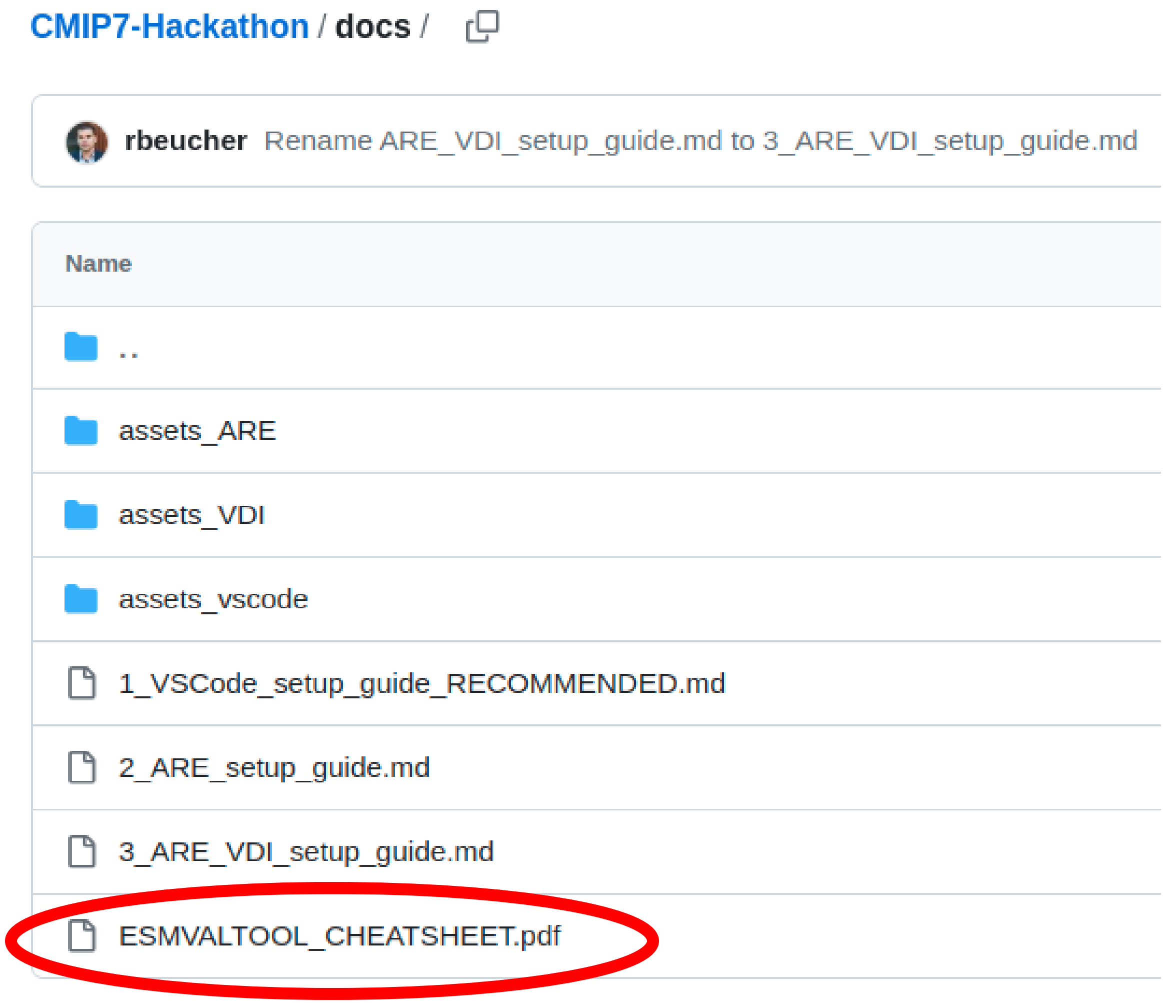
CMIP7-Hackathon / docs /

rbeucher Rename ARE_VDI_setup_guide.md to 3_ARE_VDI_setup_guide.md

Name
..
assets_ARE
assets_VDI
assets_vscode
1_VSCode_setup_guide_RECOMMENDED.md
2_ARE_setup_guide.md
3_ARE_VDI_setup_guide.md
ESMVALTOOL_CHEATSHEET.pdf

https://github.com/ACCESS-NRI/CMIP7-Hackathon/tree/main/docs

FAQ: Check the cheatsheet



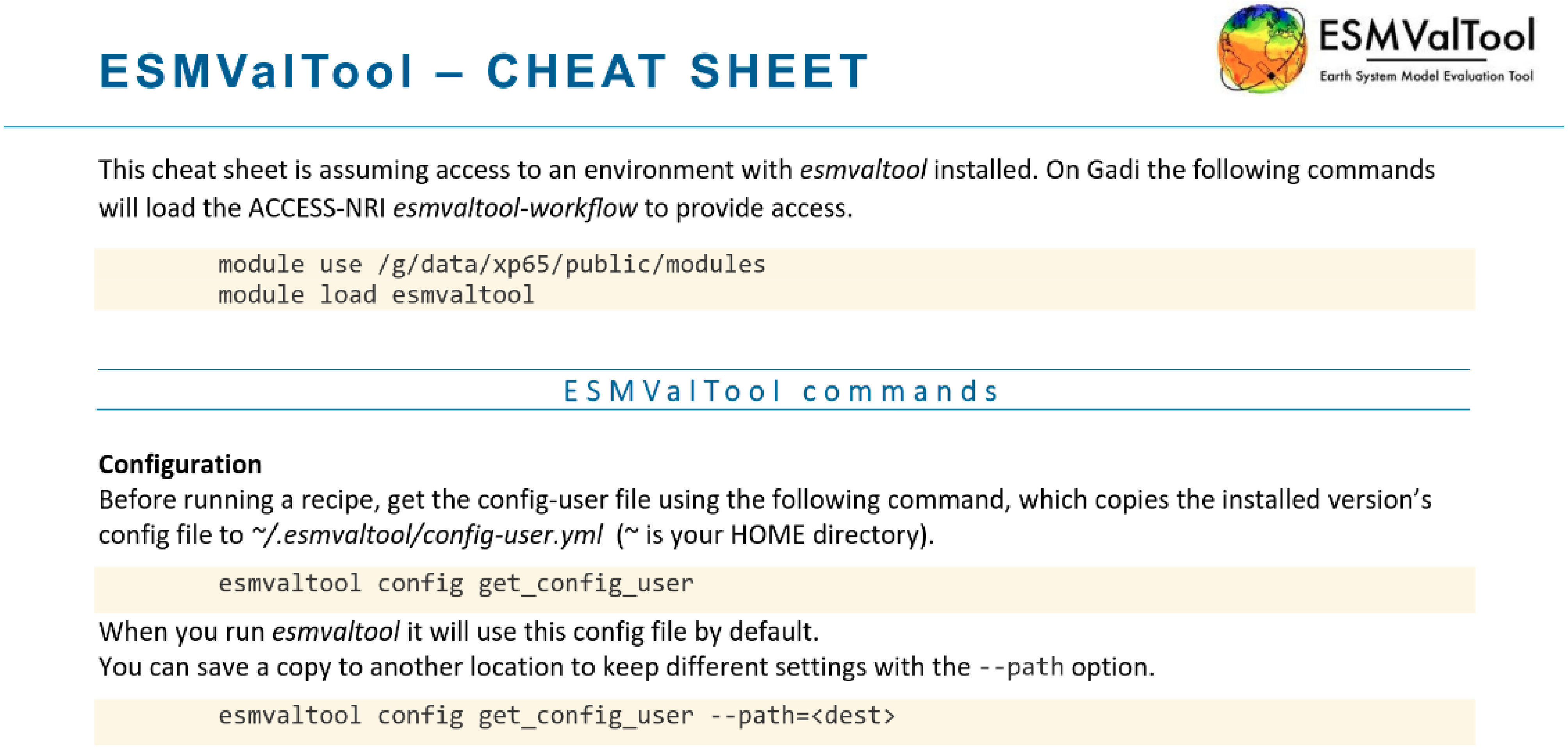
CMIP7-Hackathon / docs /

rbeucher Rename ARE_VDI_setup_guide.md to 3_ARE_VDI_setup_guide.md

Name
..
assets_ARE
assets_VDI
assets_vscode
1_VSCode_setup_guide_RECOMMENDED.md
2_ARE_setup_guide.md
3_ARE_VDI_setup_guide.md
ESMVALTOOL_CHEATSHEET.pdf

https://github.com/ACCESS-NRI/CMIP7-Hackathon/tree/main/docs

FAQ: Check the cheatsheet



ESMValTool – CHEAT SHEET

ESMValTool
Earth System Model Evaluation Tool

This cheat sheet is assuming access to an environment with *esmvaltool* installed. On Gadi the following commands will load the ACCESS-NRI *esmvaltool-workflow* to provide access.

```
        module use /g/data/xp65/public/modules
        module load esmvaltool
```

ESMValTool commands

Configuration

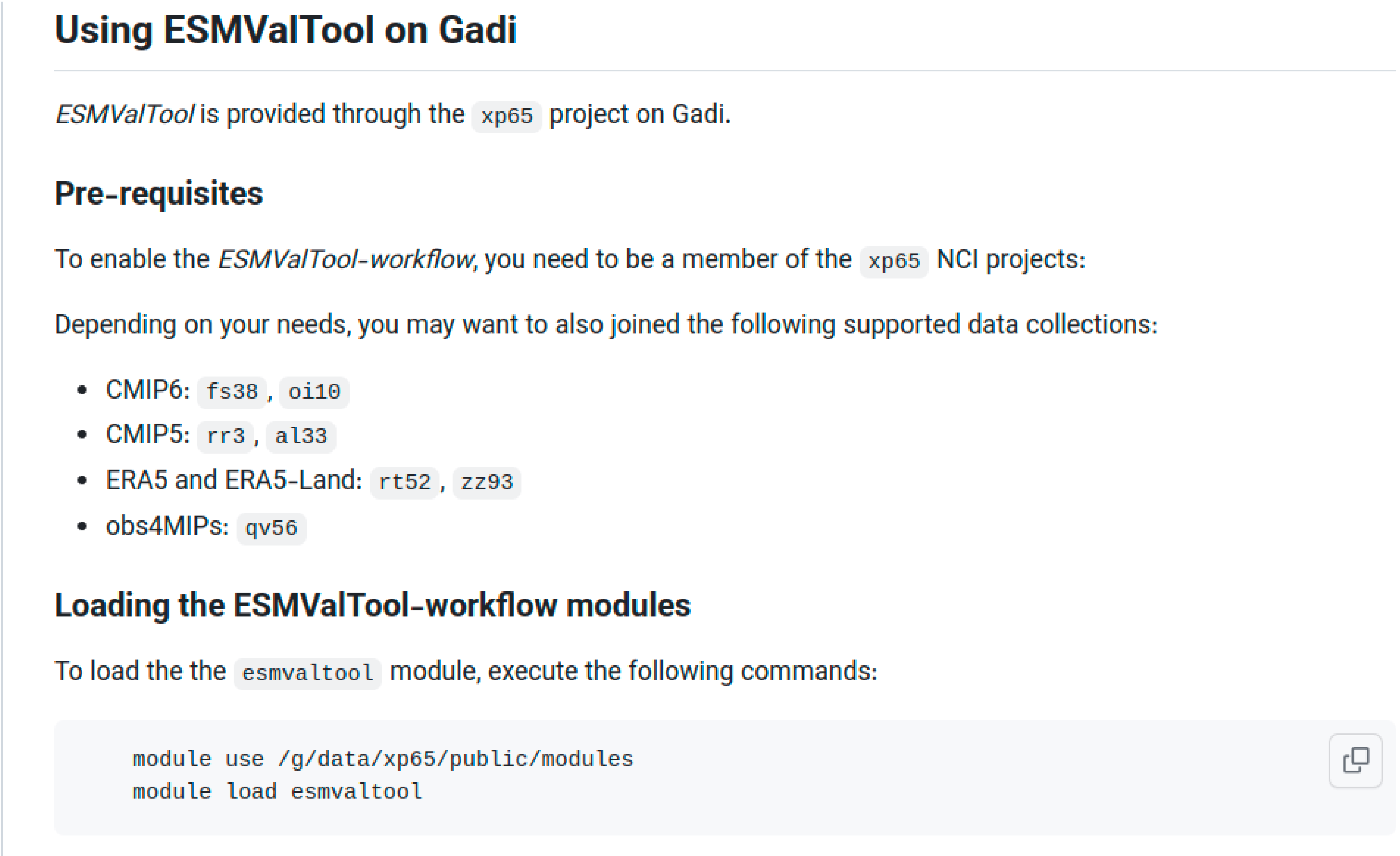
Before running a recipe, get the config-user file using the following command, which copies the installed version's config file to *~/.esmvaltool/config-user.yml* (~ is your HOME directory).

```
        esmvaltool config get_config_user
```

When you run *esmvaltool* it will use this config file by default.
You can save a copy to another location to keep different settings with the `--path` option.

```
        esmvaltool config get_config_user --path=<dest>
```

ESMValTool-Workflow



Using ESMValTool on Gadi

ESMValTool is provided through the `xp65` project on Gadi.

Pre-requisites

To enable the *ESMValTool-workflow*, you need to be a member of the `xp65` NCI projects:

Depending on your needs, you may want to also joined the following supported data collections:

- CMIP6: `fs38`, `oi10`
- CMIP5: `rr3`, `al33`
- ERA5 and ERA5-Land: `rt52`, `zz93`
- obs4MIPs: `qv56`

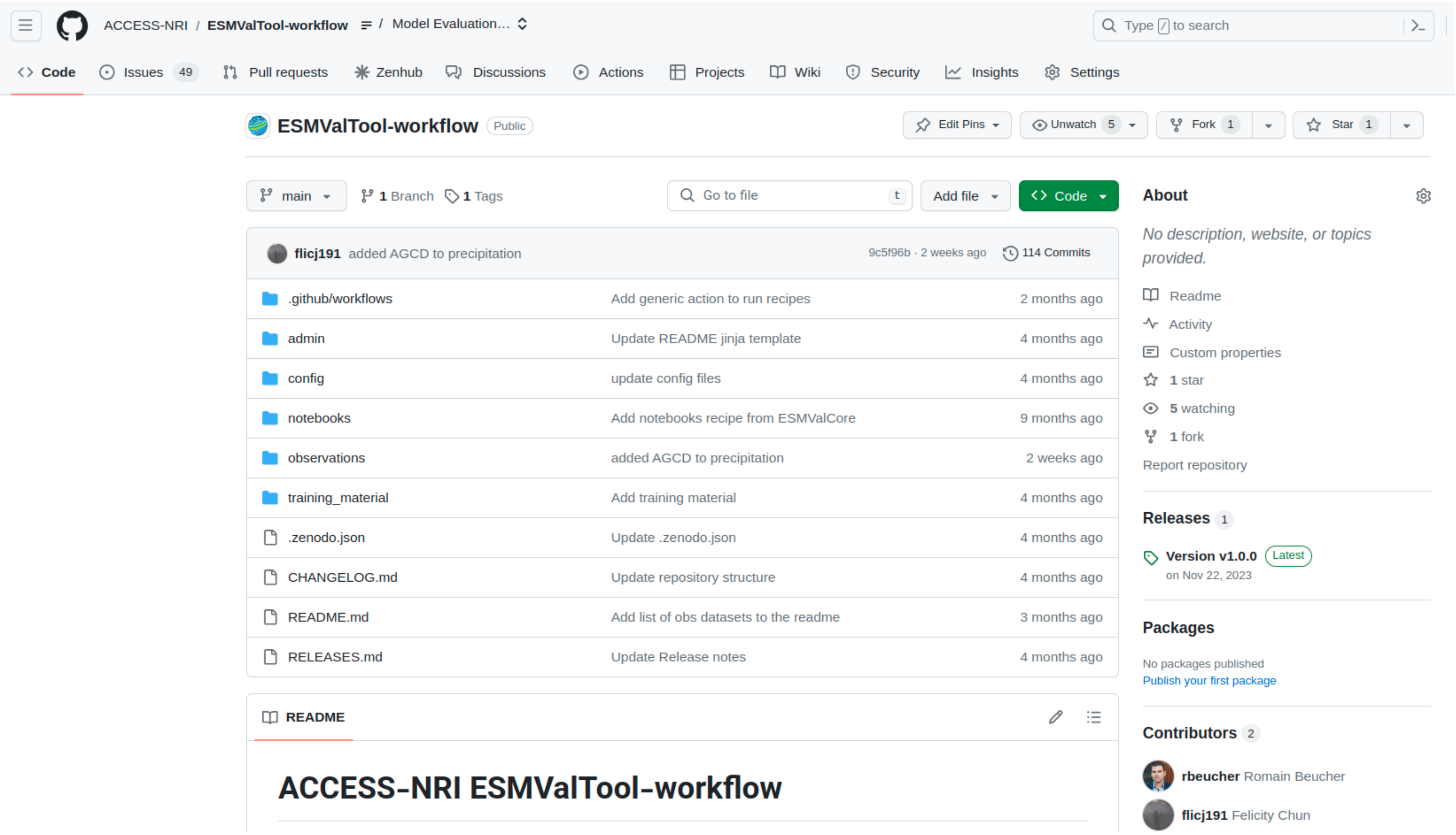
Loading the ESMValTool-workflow modules

To load the the `esmvaltool` module, execute the following commands:

```
module use /g/data/xp65/public/modules
module load esmvaltool
```

ESMValTool-Workflow (ESMValTool deployment on Gadi)

Australian Earth System Simulator
National Research Infrastructure



flicj191 added AGCD to precipitation 9c5f96b · 2 weeks ago 114 Commits

.github/workflows	Add generic action to run recipes	2 months ago
admin	Update README jinja template	4 months ago
config	update config files	4 months ago
notebooks	Add notebooks recipe from ESMValCore	9 months ago
observations	added AGCD to precipitation	2 weeks ago
training_material	Add training material	4 months ago
.zenodo.json	Update .zenodo.json	4 months ago
CHANGELOG.md	Update repository structure	4 months ago
README.md	Add list of obs datasets to the readme	3 months ago
RELEASES.md	Update Release notes	4 months ago

README

ACCESS-NRI ESMValTool-workflow

About

No description, website, or topics provided.

Readme
Activity
Custom properties
1 star
5 watching
1 fork
Report repository

Releases 1

Version v1.0.0 Latest
on Nov 22, 2023

Packages

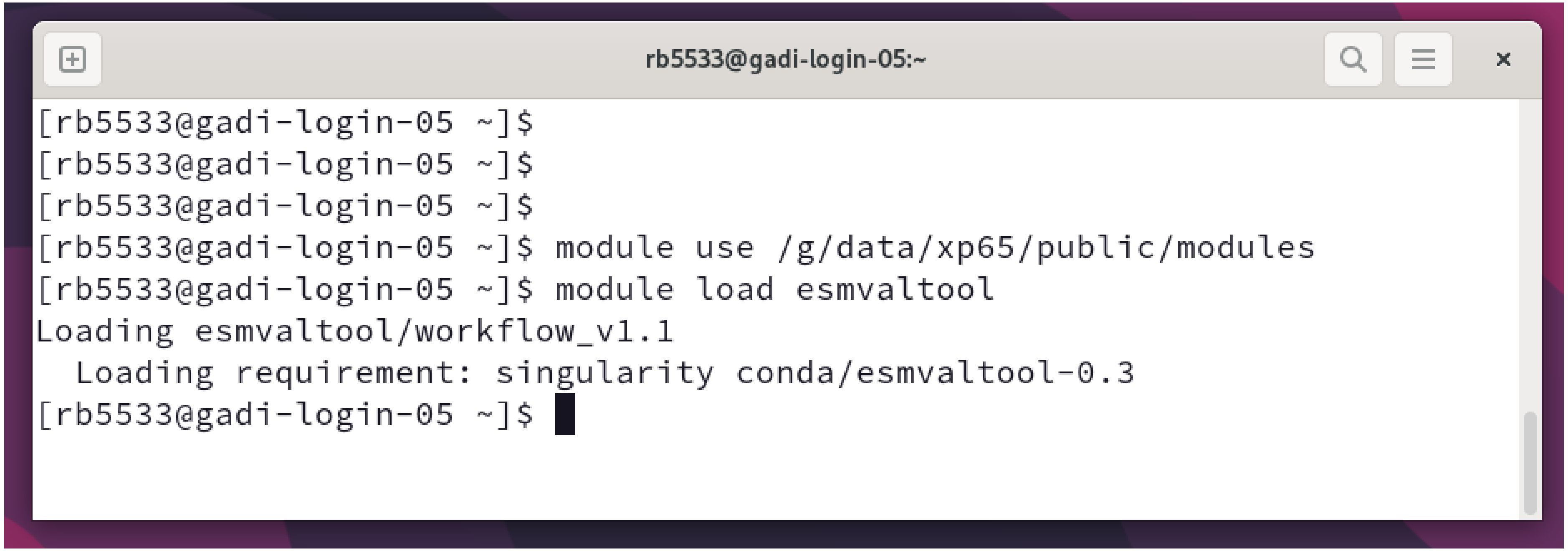
No packages published
Publish your first package

Contributors 2

rbeucher Romain Beucher
flicj191 Felicity Chun

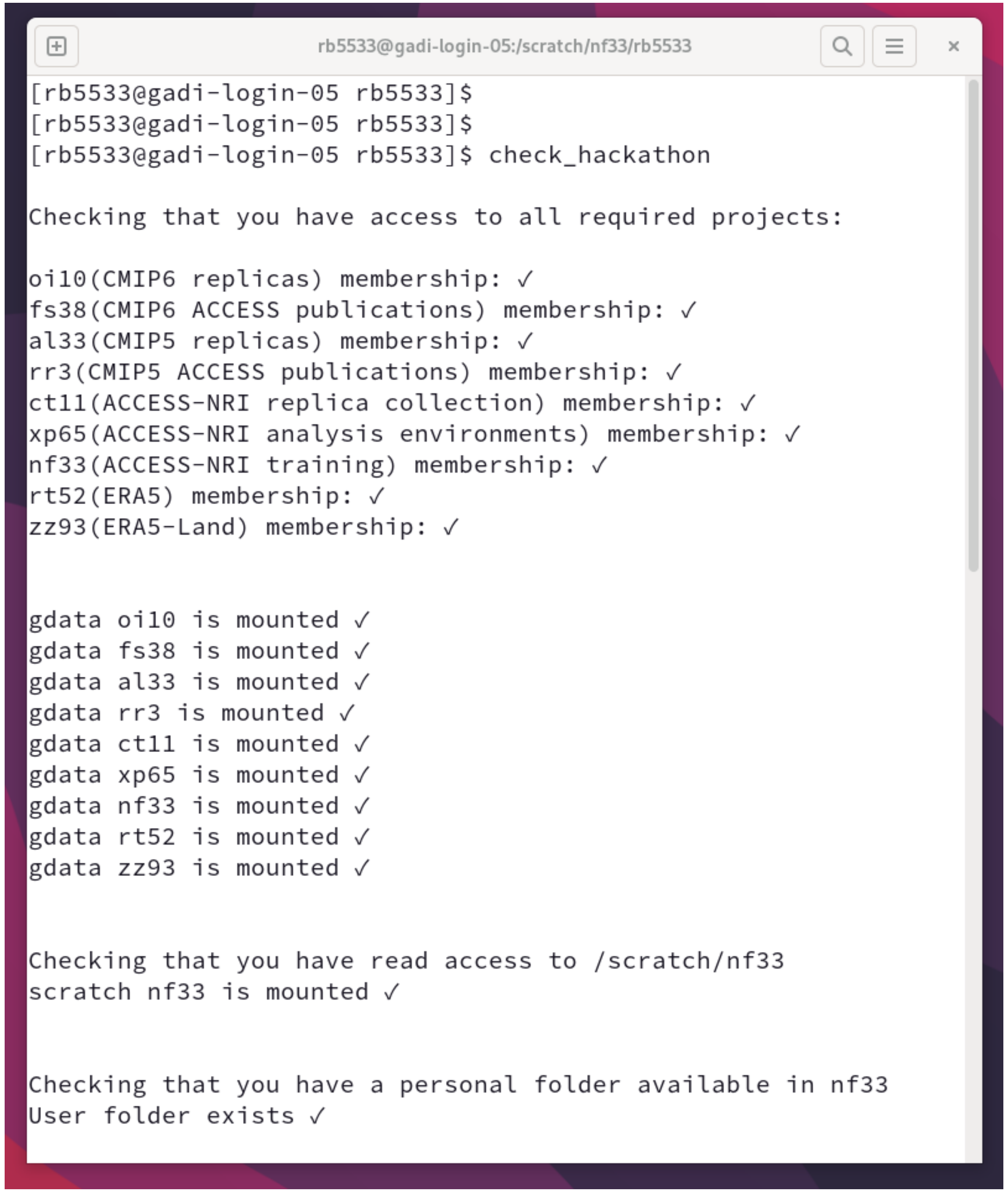
https://github.com/ACCESS-NRI/ESMValTool-workflow

Load ESMValTool on Gadi



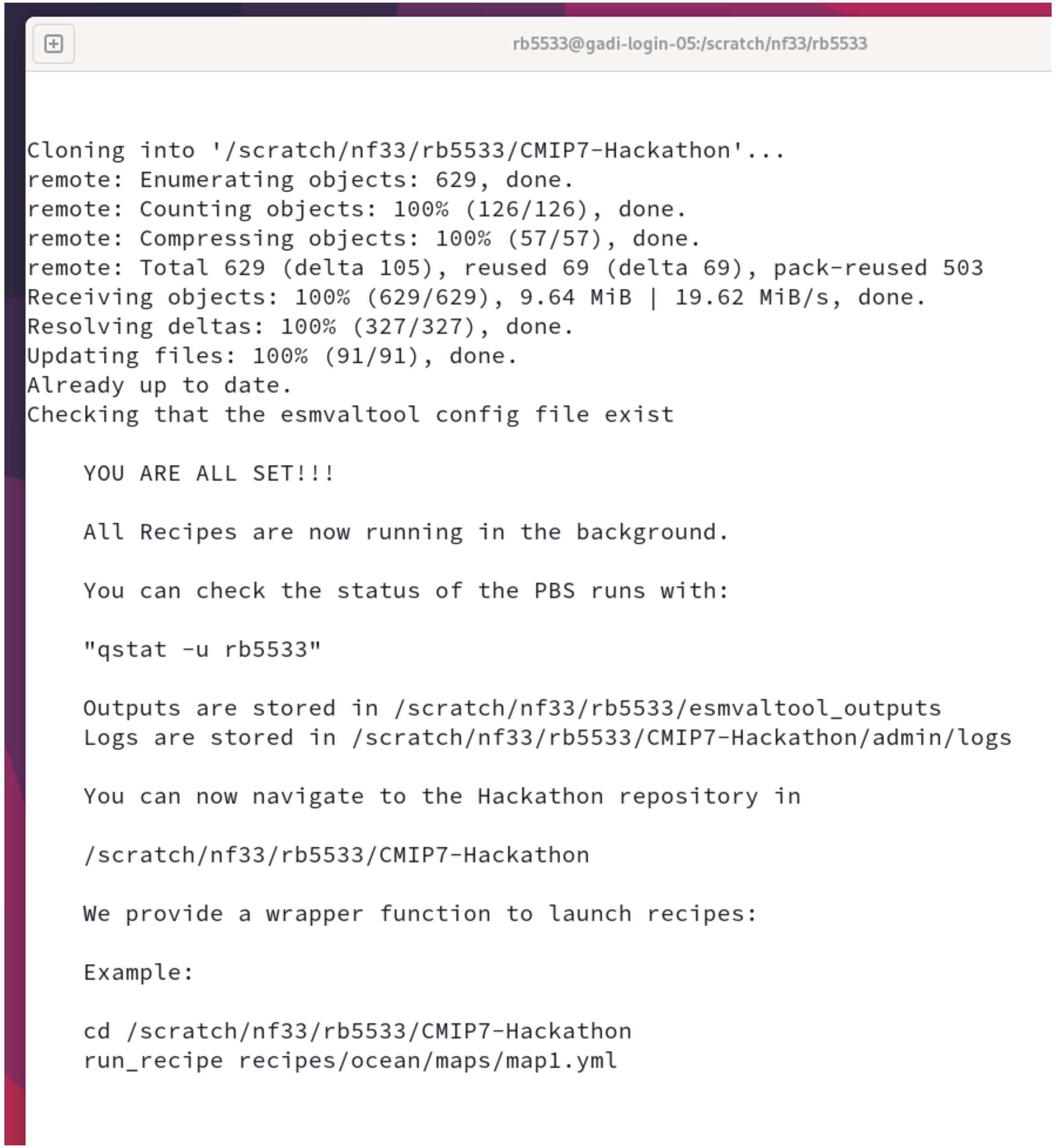
```
[rb5533@gadi-login-05 ~]$
[rb5533@gadi-login-05 ~]$
[rb5533@gadi-login-05 ~]$
[rb5533@gadi-login-05 ~]$ module use /g/data/xp65/public/modules
[rb5533@gadi-login-05 ~]$ module load esmvaltool
Loading esmvaltool/workflow_v1.1
  Loading requirement: singularity conda/esmvaltool-0.3
[rb5533@gadi-login-05 ~]$
```

Run check_hackathon



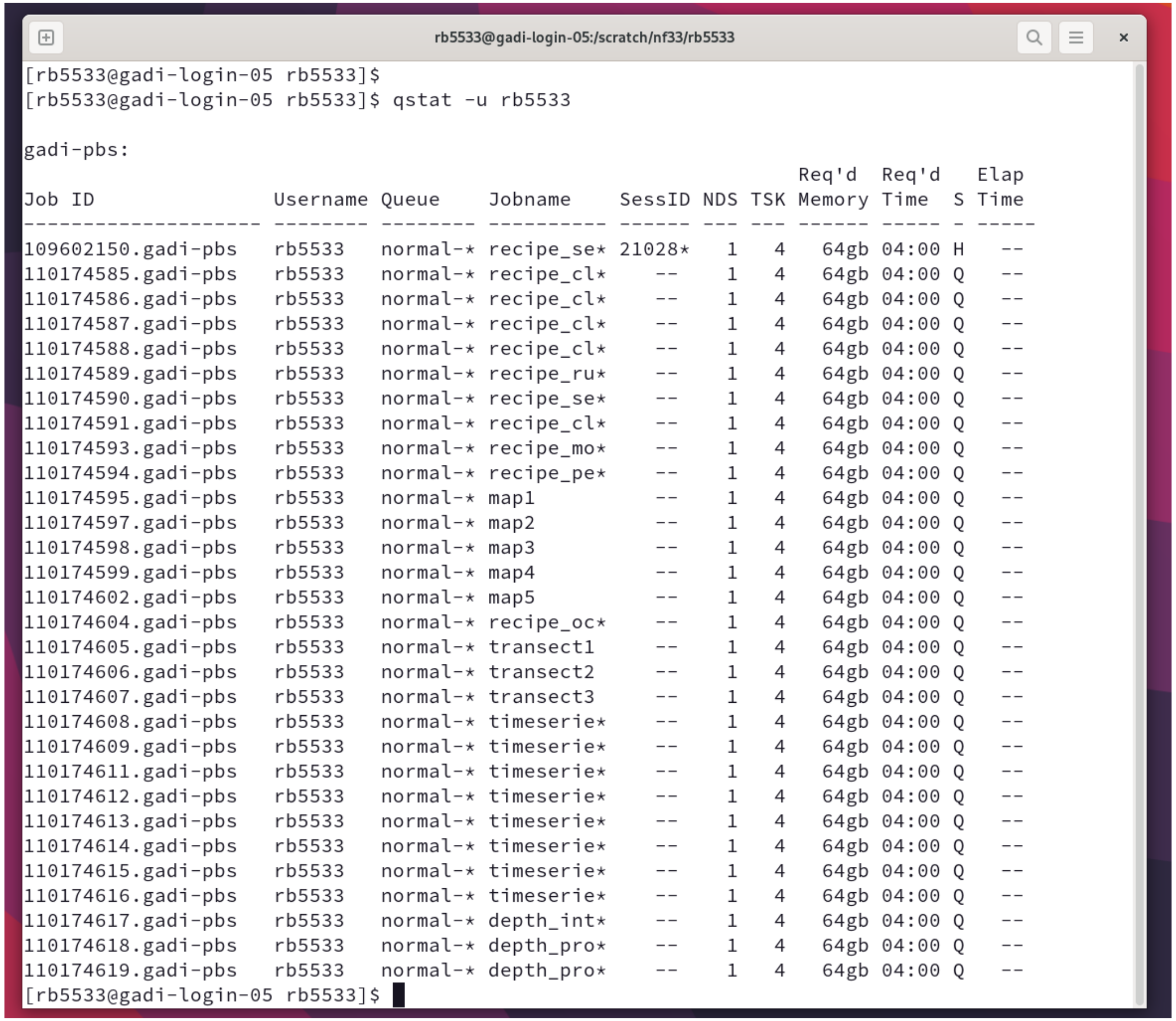
```
[rb5533@gadi-login-05 rb5533]$
[rb5533@gadi-login-05 rb5533]$
[rb5533@gadi-login-05 rb5533]$ check_hackathon

Checking that you have access to all required projects:

oi10(CMIP6 replicas) membership: ✓
fs38(CMIP6 ACCESS publications) membership: ✓
al33(CMIP5 replicas) membership: ✓
rr3(CMIP5 ACCESS publications) membership: ✓
ct11(ACCESS-NRI replica collection) membership: ✓
xp65(ACCESS-NRI analysis environments) membership: ✓
nf33(ACCESS-NRI training) membership: ✓
rt52(ERA5) membership: ✓
zz93(ERA5-Land) membership: ✓


gdata oi10 is mounted ✓
gdata fs38 is mounted ✓
gdata al33 is mounted ✓
gdata rr3 is mounted ✓
gdata ct11 is mounted ✓
gdata xp65 is mounted ✓
gdata nf33 is mounted ✓
gdata rt52 is mounted ✓
gdata zz93 is mounted ✓


Checking that you have read access to /scratch/nf33
scratch nf33 is mounted ✓


Checking that you have a personal folder available in nf33
User folder exists ✓
```

Run check_hackathon



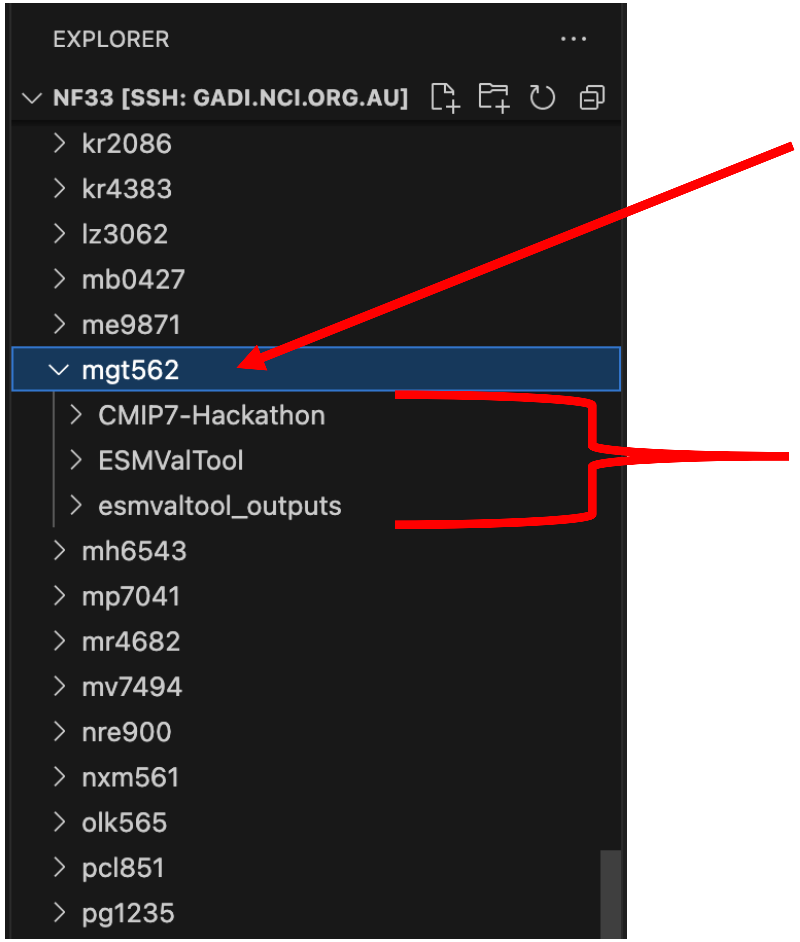
```
Cloning into '/scratch/nf33/rb5533/CMIP7-Hackathon'...
remote: Enumerating objects: 629, done.
remote: Counting objects: 100% (126/126), done.
remote: Compressing objects: 100% (57/57), done.
remote: Total 629 (delta 105), reused 69 (delta 69), pack-reused 503
Receiving objects: 100% (629/629), 9.64 MiB | 19.62 MiB/s, done.
Resolving deltas: 100% (327/327), done.
Updating files: 100% (91/91), done.
Already up to date.
Checking that the esmvaltool config file exist

    YOU ARE ALL SET!!!

    All Recipes are now running in the background.

    You can check the status of the PBS runs with:

    "qstat -u rb5533"

    Outputs are stored in /scratch/nf33/rb5533/esmvaltool_outputs
    Logs are stored in /scratch/nf33/rb5533/CMIP7-Hackathon/admin/logs

    You can now navigate to the Hackathon repository in

    /scratch/nf33/rb5533/CMIP7-Hackathon

    We provide a wrapper function to launch recipes:

    Example:

    cd /scratch/nf33/rb5533/CMIP7-Hackathon
    run_recipe recipes/ocean/maps/map1.yml
```

Run check_hackathon



```
[rb5533@gadi-login-05 rb5533]$
[rb5533@gadi-login-05 rb5533]$ qstat -u rb5533

gadi-pbs:
                                                                 Req'd  Req'd   Elap
Job ID               Username Queue    Jobname    SessID NDS TSK Memory Time  S Time
-------------------- -------- -------- ---------- ------ --- --- ------ ----- - -----
109602150.gadi-pbs   rb5533   normal-* recipe_se* 21028*   1   4   64gb 04:00 H    --
110174585.gadi-pbs   rb5533   normal-* recipe_cl*     --   1   4   64gb 04:00 Q    --
110174586.gadi-pbs   rb5533   normal-* recipe_cl*     --   1   4   64gb 04:00 Q    --
110174587.gadi-pbs   rb5533   normal-* recipe_cl*     --   1   4   64gb 04:00 Q    --
110174588.gadi-pbs   rb5533   normal-* recipe_cl*     --   1   4   64gb 04:00 Q    --
110174589.gadi-pbs   rb5533   normal-* recipe_ru*     --   1   4   64gb 04:00 Q    --
110174590.gadi-pbs   rb5533   normal-* recipe_se*     --   1   4   64gb 04:00 Q    --
110174591.gadi-pbs   rb5533   normal-* recipe_cl*     --   1   4   64gb 04:00 Q    --
110174593.gadi-pbs   rb5533   normal-* recipe_mo*     --   1   4   64gb 04:00 Q    --
110174594.gadi-pbs   rb5533   normal-* recipe_pe*     --   1   4   64gb 04:00 Q    --
110174595.gadi-pbs   rb5533   normal-* map1           --   1   4   64gb 04:00 Q    --
110174597.gadi-pbs   rb5533   normal-* map2           --   1   4   64gb 04:00 Q    --
110174598.gadi-pbs   rb5533   normal-* map3           --   1   4   64gb 04:00 Q    --
110174599.gadi-pbs   rb5533   normal-* map4           --   1   4   64gb 04:00 Q    --
110174602.gadi-pbs   rb5533   normal-* map5           --   1   4   64gb 04:00 Q    --
110174604.gadi-pbs   rb5533   normal-* recipe_oc*     --   1   4   64gb 04:00 Q    --
110174605.gadi-pbs   rb5533   normal-* transect1      --   1   4   64gb 04:00 Q    --
110174606.gadi-pbs   rb5533   normal-* transect2      --   1   4   64gb 04:00 Q    --
110174607.gadi-pbs   rb5533   normal-* transect3      --   1   4   64gb 04:00 Q    --
110174608.gadi-pbs   rb5533   normal-* timeserie*     --   1   4   64gb 04:00 Q    --
110174609.gadi-pbs   rb5533   normal-* timeserie*     --   1   4   64gb 04:00 Q    --
110174611.gadi-pbs   rb5533   normal-* timeserie*     --   1   4   64gb 04:00 Q    --
110174612.gadi-pbs   rb5533   normal-* timeserie*     --   1   4   64gb 04:00 Q    --
110174613.gadi-pbs   rb5533   normal-* timeserie*     --   1   4   64gb 04:00 Q    --
110174614.gadi-pbs   rb5533   normal-* timeserie*     --   1   4   64gb 04:00 Q    --
110174615.gadi-pbs   rb5533   normal-* timeserie*     --   1   4   64gb 04:00 Q    --
110174616.gadi-pbs   rb5533   normal-* timeserie*     --   1   4   64gb 04:00 Q    --
110174617.gadi-pbs   rb5533   normal-* depth_int*     --   1   4   64gb 04:00 Q    --
110174618.gadi-pbs   rb5533   normal-* depth_pro*     --   1   4   64gb 04:00 Q    --
110174619.gadi-pbs   rb5533   normal-* depth_pro*     --   1   4   64gb 04:00 Q    --
[rb5533@gadi-login-05 rb5533]$
```

Navigating **/scratch/nf33**



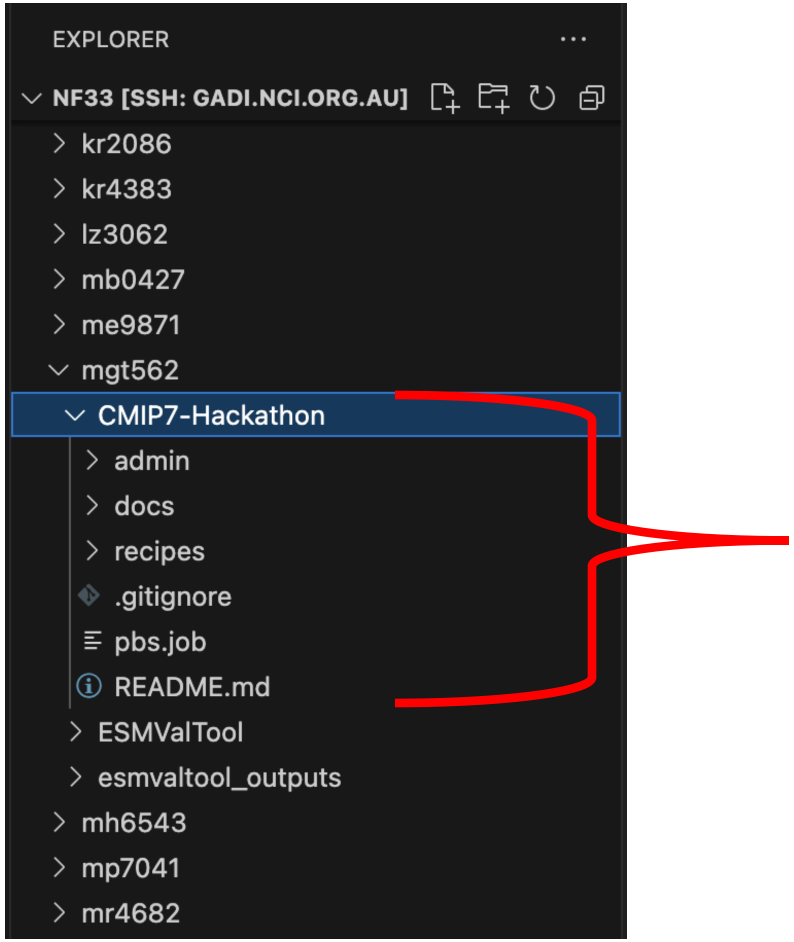
```
EXPLORER
NF33 [SSH: GADI.NCI.ORG.AU]
  kr2086
  kr4383
  lz3062
  mb0427
  me9871
  mgt562
    CMIP7-Hackathon
    ESMValTool
    esmvaltool_outputs
  mh6543
  mp7041
  mr4682
  mv7494
  nre900
  nxm561
  olk565
  pcl851
  pg1235
```

Each Hackathon attendee has a personal folder based on their NCI Account name.

Once you have completed the Hackathon setup guide (*see the CMIP7-Hackathon Github repo README*) you will have the following 3 directories:

1. **CMIP7-Hackathon** (clone of github repo)
2. **ESMValTool** (ESMValTool package/dependencies)
3. **esmvaltool_outputs** (location of all recipe outputs)

Navigating **/scratch/nf33**



```
EXPLORER
∨ NF33 [SSH: GADI.NCI.ORG.AU]
  > kr2086
  > kr4383
  > lz3062
  > mb0427
  > me9871
  ∨ mgt562
    ∨ CMIP7-Hackathon
      > admin
      > docs
      > recipes
        .gitignore
        pbs.job
        README.md
    > ESMValTool
    > esmvaltool_outputs
  > mh6543
  > mp7041
  > mr4682
```

The CMIP7-Hackathon repo structure is as follows:

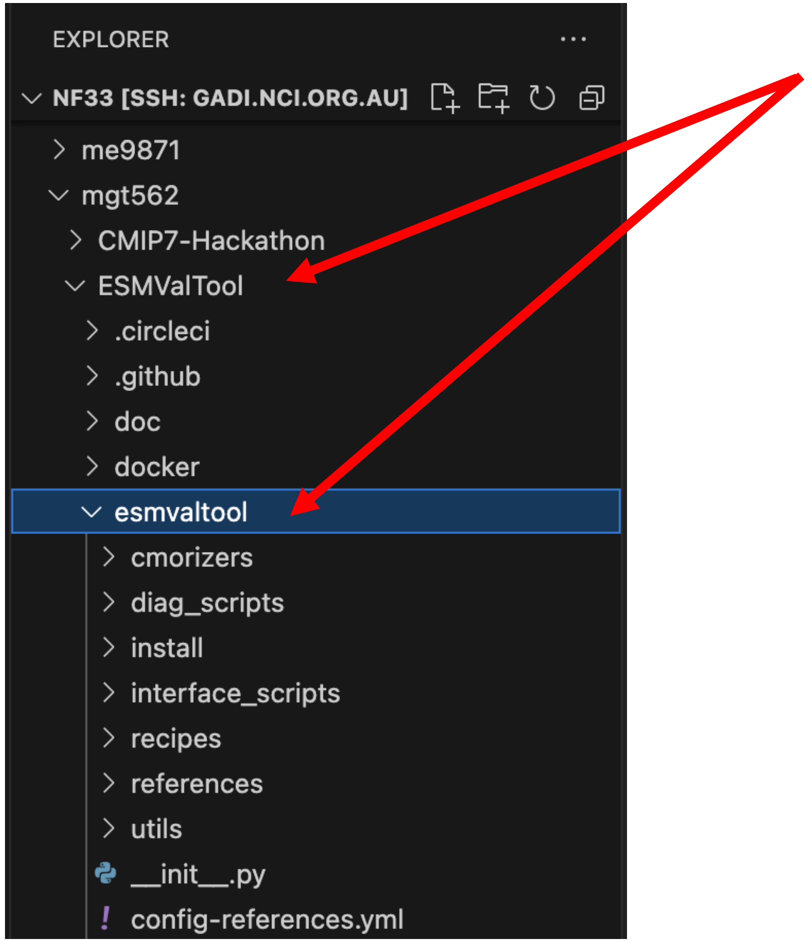
> **admin**
>
> > **logs** (log files generated by running recipes. e.g. *depth_integration.stdout*)

> **docs** (all setup guides linked from README)

> **recipes** (ESMValTool recipes (**.yml*) by domain)
>
> > **atmosphere**
> >
> > **enso**
> >
> > **general**
> >
> > **land**
> >
> > **ocean**
> >
> > **sample_recipe**
> >
> > **sea-ice**

*N.B. Some recipes may also contain a *.py file*

Navigating **/scratch/nf33**



```
EXPLORER                                ...
∨ NF33 [SSH: GADI.NCI.ORG.AU]
  > me9871
  ∨ mgt562
    > CMIP7-Hackathon
    ∨ ESMValTool
      > .circleci
      > .github
      > doc
      > docker
      ∨ esmvaltool
        > cmorizers
        > diag_scripts
        > install
        > interface_scripts
        > recipes
        > references
        > utils
          __init__.py
        ! config-references.yml
```

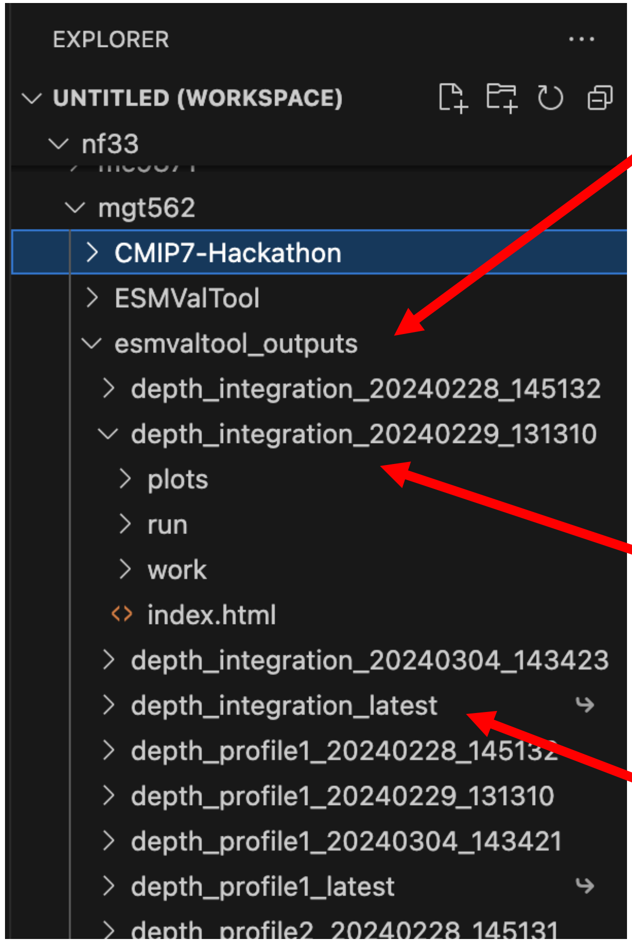
> ESMValTool

ESMValTool-workflow software requirements, including:

- ESMValTool package:
  - core recipes
  - CMORizers
  - diagnostic scripts
  - utilities
  - references

- docker container info
- config files
- python environments
- associated licences
- documentation

Navigating /scratch/nf33



```
EXPLORER                                   ...
∨ UNTITLED (WORKSPACE)
  ∨ nf33
    ∨ mgt562
      > CMIP7-Hackathon
      > ESMValTool
      ∨ esmvaltool_outputs
        > depth_integration_20240228_145132
        ∨ depth_integration_20240229_131310
          > plots
          > run
          > work
          <> index.html
        > depth_integration_20240304_143423
        > depth_integration_latest
        > depth_profile1_20240228_145132
        > depth_profile1_20240229_131310
        > depth_profile1_20240304_143421
        > depth_profile1_latest
        > depth_profile2_20240228_145131
```

> esmvaltool_outputs

When a recipe is successfully run, all ESMValTool outputs are saved here including:

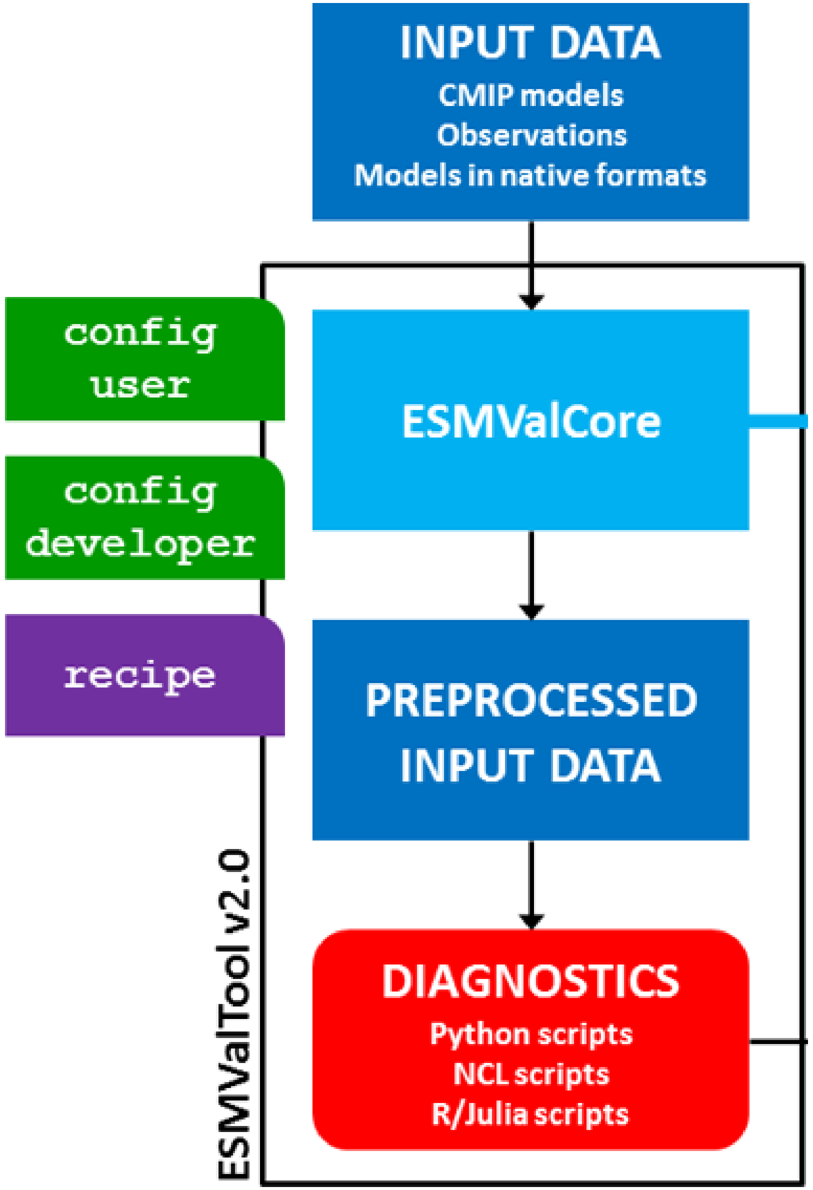
- plots
- log files
- recipe information

Outputs for all unique recipe runs are stored separately in directories using the following naming convention:

recipe_name_YYYYMMDD_HHMMSS

The most recent recipe run can be found in the relevant (symlink) directory named:

recipe_name_latest



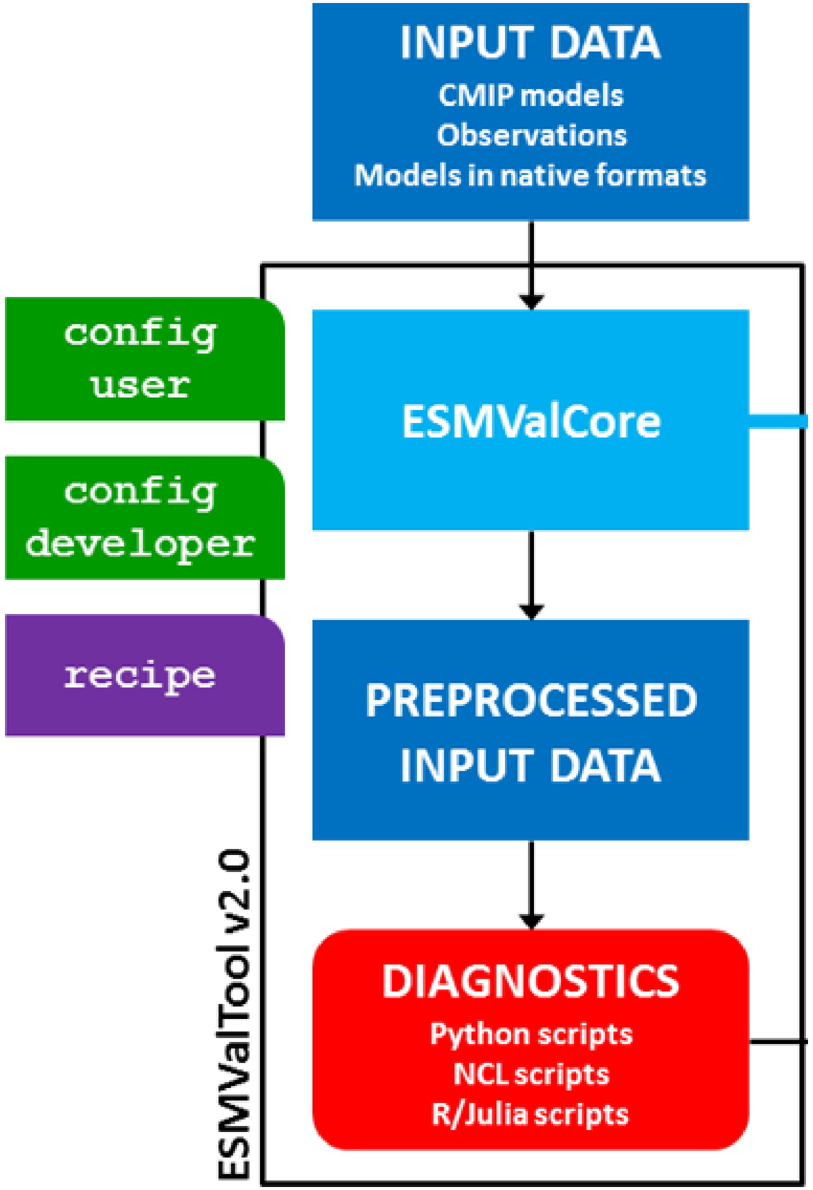
Recipe

Instructions are given to esmvaltool using a YAML script called a recipe.

Instructions like:

- the **datasets** which need to be analyzed,
- the **preprocessors** that need to be applied,
- the specific analysis called **diagnostic** script.

Available recipes:
https://docs.esmvaltool.org/en/latest/recipes/index.html

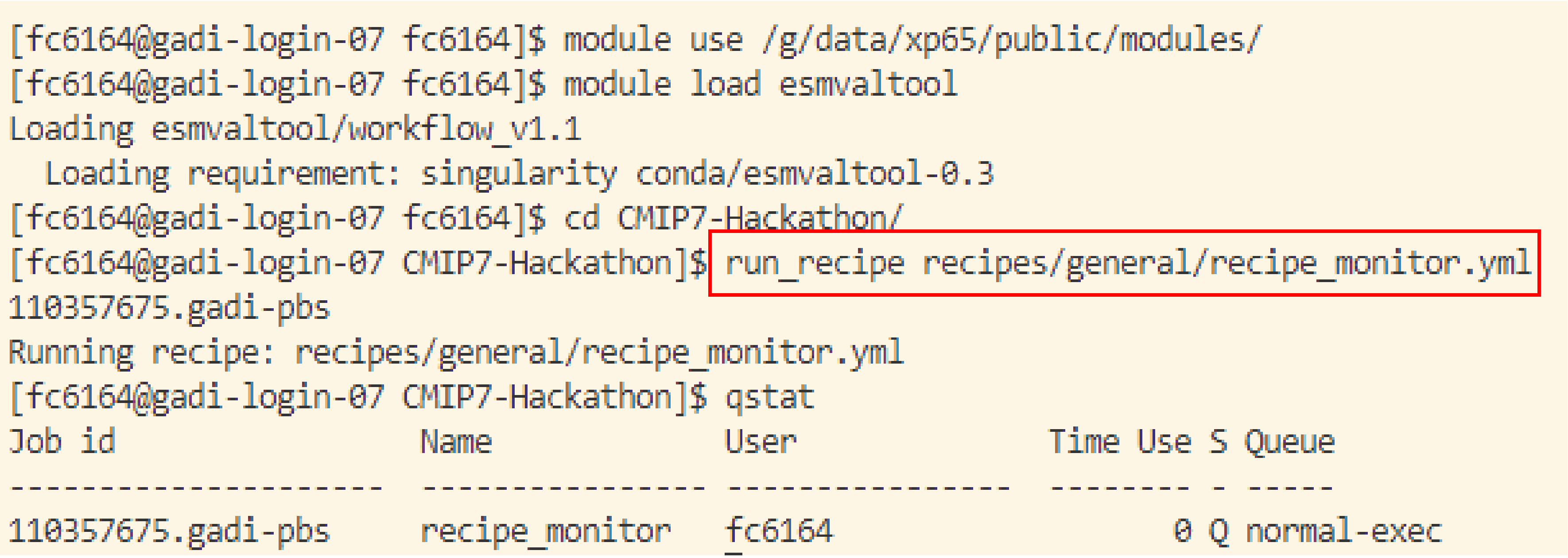


How to run a recipe:

- ESMValTool as a command-line tool:

```
esmvaltool run examples/recipe_python.yml
```

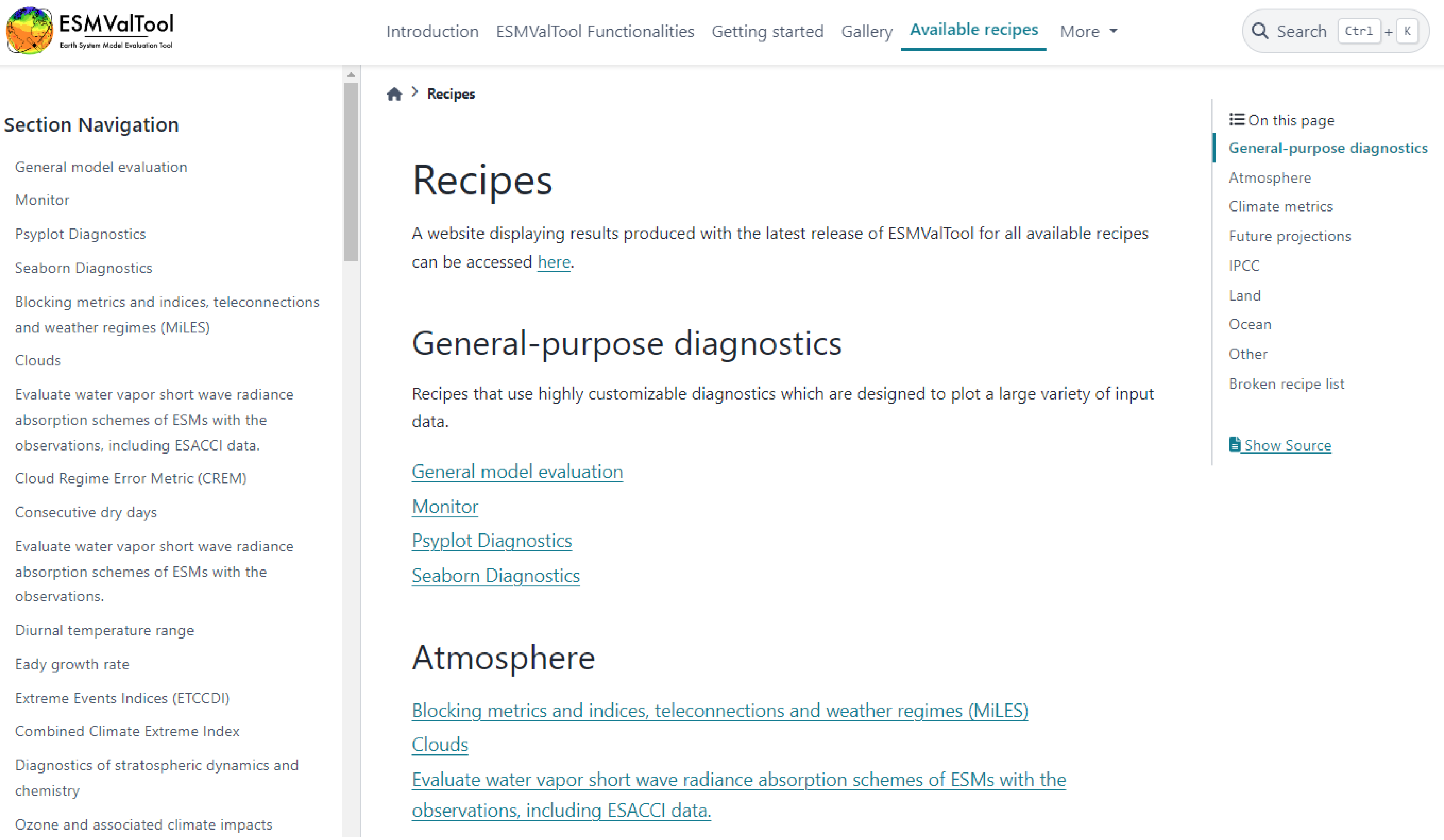
run_recipe



```
[fc6164@gadi-login-07 fc6164]$ module use /g/data/xp65/public/modules/
[fc6164@gadi-login-07 fc6164]$ module load esmvaltool
Loading esmvaltool/workflow_v1.1
  Loading requirement: singularity conda/esmvaltool-0.3
[fc6164@gadi-login-07 fc6164]$ cd CMIP7-Hackathon/
[fc6164@gadi-login-07 CMIP7-Hackathon]$ run_recipe recipes/general/recipe_monitor.yml
110357675.gadi-pbs
Running recipe: recipes/general/recipe_monitor.yml
[fc6164@gadi-login-07 CMIP7-Hackathon]$ qstat
Job id                 Name             User              Time Use S Queue
---------------------  ---------------- ----------------  -------- - -----
110357675.gadi-pbs     recipe_monitor   fc6164                   0 Q normal-exec
```

run_recipe command available in the esmvaltool module in xp65

- ***esmvaltool run* recipe** command in a pbs job
- sets output directory to users esmvaltool_outputs folder in nf33


ESMValTool
Earth System Model Evaluation Tool
Introduction
ESMValTool Functionalities
Getting started
Gallery
Available recipes
More
Search Ctrl + K
Recipes
Section Navigation
General model evaluation
Monitor
Psyplot Diagnostics
Seaborn Diagnostics
Blocking metrics and indices, teleconnections and weather regimes (MiLES)
Clouds
Evaluate water vapor short wave radiance absorption schemes of ESMs with the observations, including ESACCI data.
Cloud Regime Error Metric (CREM)
Consecutive dry days
Evaluate water vapor short wave radiance absorption schemes of ESMs with the observations.
Diurnal temperature range
Eady growth rate
Extreme Events Indices (ETCCDI)
Combined Climate Extreme Index
Diagnostics of stratospheric dynamics and chemistry
Ozone and associated climate impacts
On this page
General-purpose diagnostics
Atmosphere
Climate metrics
Future projections
IPCC
Land
Ocean
Other
Broken recipe list
Show Source

Recipes

A website displaying results produced with the latest release of ESMValTool for all available recipes can be accessed here.

General-purpose diagnostics

Recipes that use highly customizable diagnostics which are designed to plot a large variety of input data.

General model evaluation

Monitor

Psyplot Diagnostics

Seaborn Diagnostics

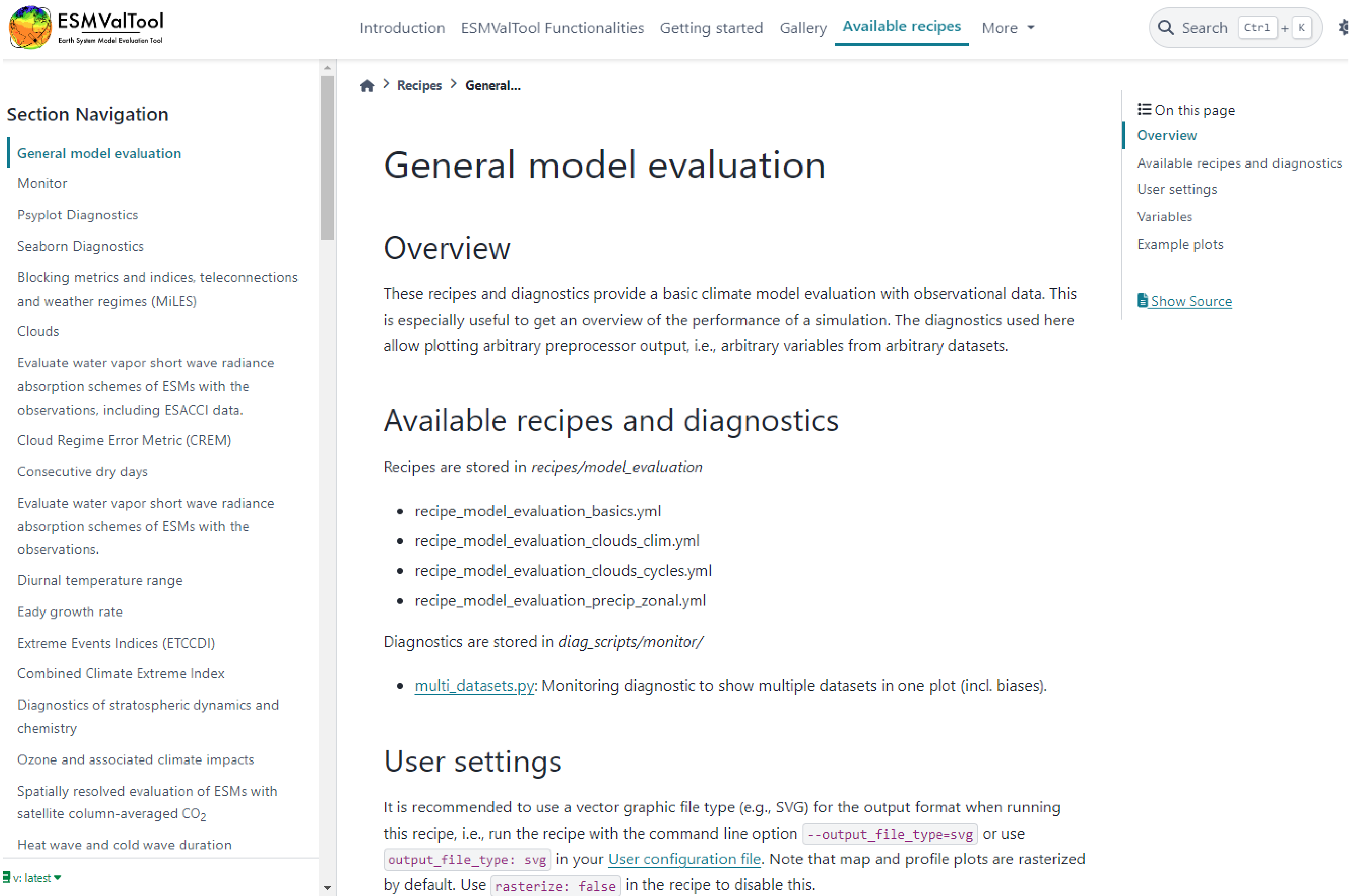
Atmosphere

Blocking metrics and indices, teleconnections and weather regimes (MiLES)

Clouds

Evaluate water vapor short wave radiance absorption schemes of ESMs with the observations, including ESACCI data.

https://docs.esmvaltool.org/en/latest/recipes/index.html


ESMValTool
Earth System Model Evaluation Tool
Introduction
ESMValTool Functionalities
Getting started
Gallery
Available recipes
More
Search Ctrl + K
Section Navigation
General model evaluation
Monitor
Psyplot Diagnostics
Seaborn Diagnostics
Blocking metrics and indices, teleconnections and weather regimes (MiLES)
Clouds
Evaluate water vapor short wave radiance absorption schemes of ESMs with the observations, including ESACCI data.
Cloud Regime Error Metric (CREM)
Consecutive dry days
Evaluate water vapor short wave radiance absorption schemes of ESMs with the observations.
Diurnal temperature range
Eady growth rate
Extreme Events Indices (ETCCDI)
Combined Climate Extreme Index
Diagnostics of stratospheric dynamics and chemistry
Ozone and associated climate impacts
Spatially resolved evaluation of ESMs with satellite column-averaged CO2
Heat wave and cold wave duration
v: latest
Recipes > General...
On this page
Overview
Available recipes and diagnostics
User settings
Variables
Example plots
Show Source

General model evaluation

Overview

These recipes and diagnostics provide a basic climate model evaluation with observational data. This is especially useful to get an overview of the performance of a simulation. The diagnostics used here allow plotting arbitrary preprocessor output, i.e., arbitrary variables from arbitrary datasets.

Available recipes and diagnostics

Recipes are stored in *recipes/model_evaluation*

- recipe_model_evaluation_basics.yml
- recipe_model_evaluation_clouds_clim.yml
- recipe_model_evaluation_clouds_cycles.yml
- recipe_model_evaluation_precip_zonal.yml

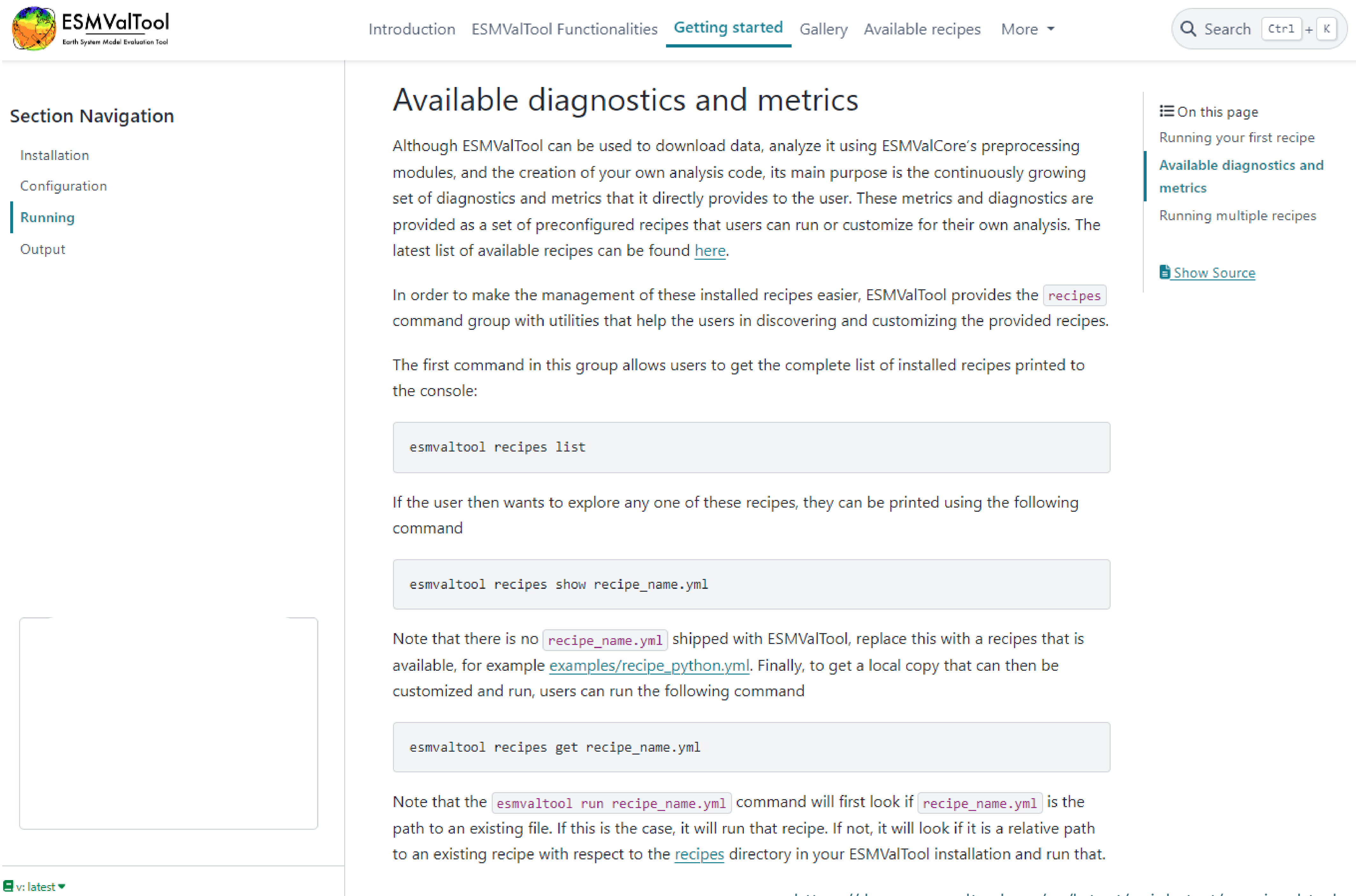
Diagnostics are stored in *diag_scripts/monitor/*

- multi_datasets.py: Monitoring diagnostic to show multiple datasets in one plot (incl. biases).

User settings

It is recommended to use a vector graphic file type (e.g., SVG) for the output format when running this recipe, i.e., run the recipe with the command line option `--output_file_type=svg` or use `output_file_type: svg` in your User configuration file. Note that map and profile plots are rasterized by default. Use `rasterize: false` in the recipe to disable this.

https://docs.esmvaltool.org/en/latest/recipes/recipe_model_evaluation.html



Available diagnostics and metrics

Although ESMValTool can be used to download data, analyze it using ESMValCore's preprocessing modules, and the creation of your own analysis code, its main purpose is the continuously growing set of diagnostics and metrics that it directly provides to the user. These metrics and diagnostics are provided as a set of preconfigured recipes that users can run or customize for their own analysis. The latest list of available recipes can be found here.

In order to make the management of these installed recipes easier, ESMValTool provides the `recipes` command group with utilities that help the users in discovering and customizing the provided recipes.

The first command in this group allows users to get the complete list of installed recipes printed to the console:

```
esmvaltool recipes list
```

If the user then wants to explore any one of these recipes, they can be printed using the following command

```
esmvaltool recipes show recipe_name.yml
```

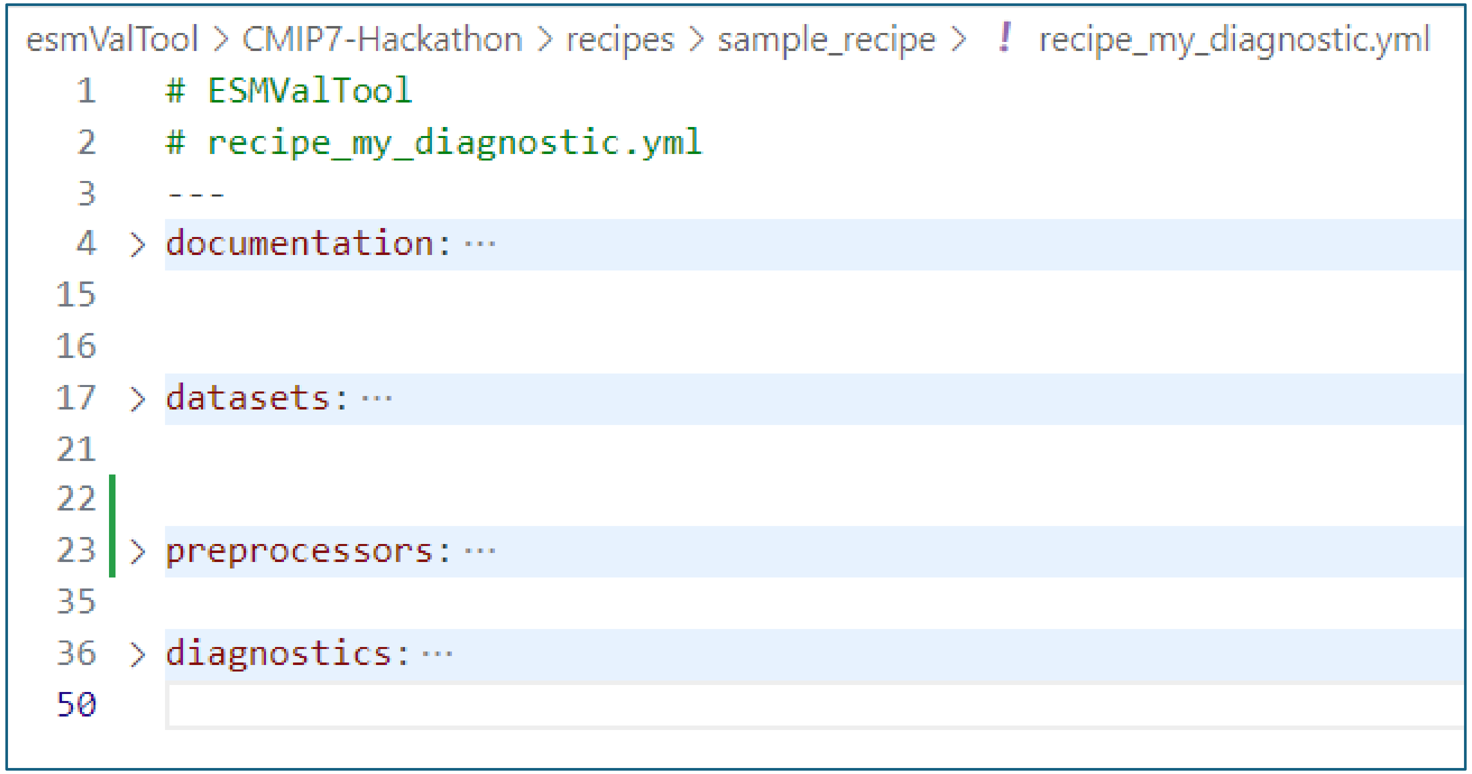
Note that there is no `recipe_name.yml` shipped with ESMValTool, replace this with a recipes that is available, for example examples/recipe_python.yml. Finally, to get a local copy that can then be customized and run, users can run the following command

```
esmvaltool recipes get recipe_name.yml
```

Note that the `esmvaltool run recipe_name.yml` command will first look if `recipe_name.yml` is the path to an existing file. If this is the case, it will run that recipe. If not, it will look if it is a relative path to an existing recipe with respect to the recipes directory in your ESMValTool installation and run that.

https://docs.esmvaltool.org/en/latest/quickstart/running.html

ESMValTool recipe yml



esmValTool > CMIP7-Hackathon > recipes > sample_recipe > ! recipe_my_diagnostic.yml

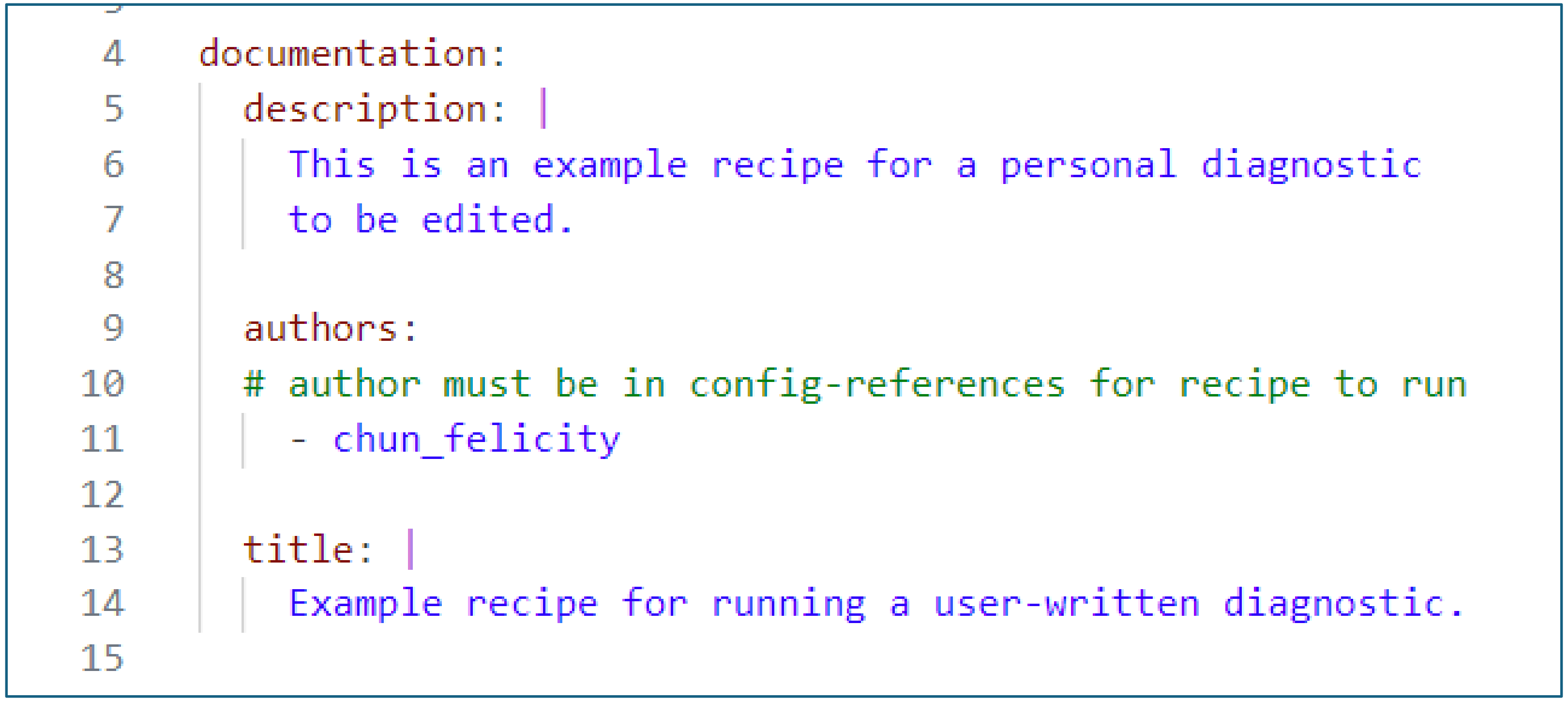
```yaml
# ESMValTool
# recipe_my_diagnostic.yml
---
documentation: ···


datasets: ···


preprocessors: ···

diagnostics: ···

```

documentation

- description
- authors
- title

Others:

maintainer,
references, projects

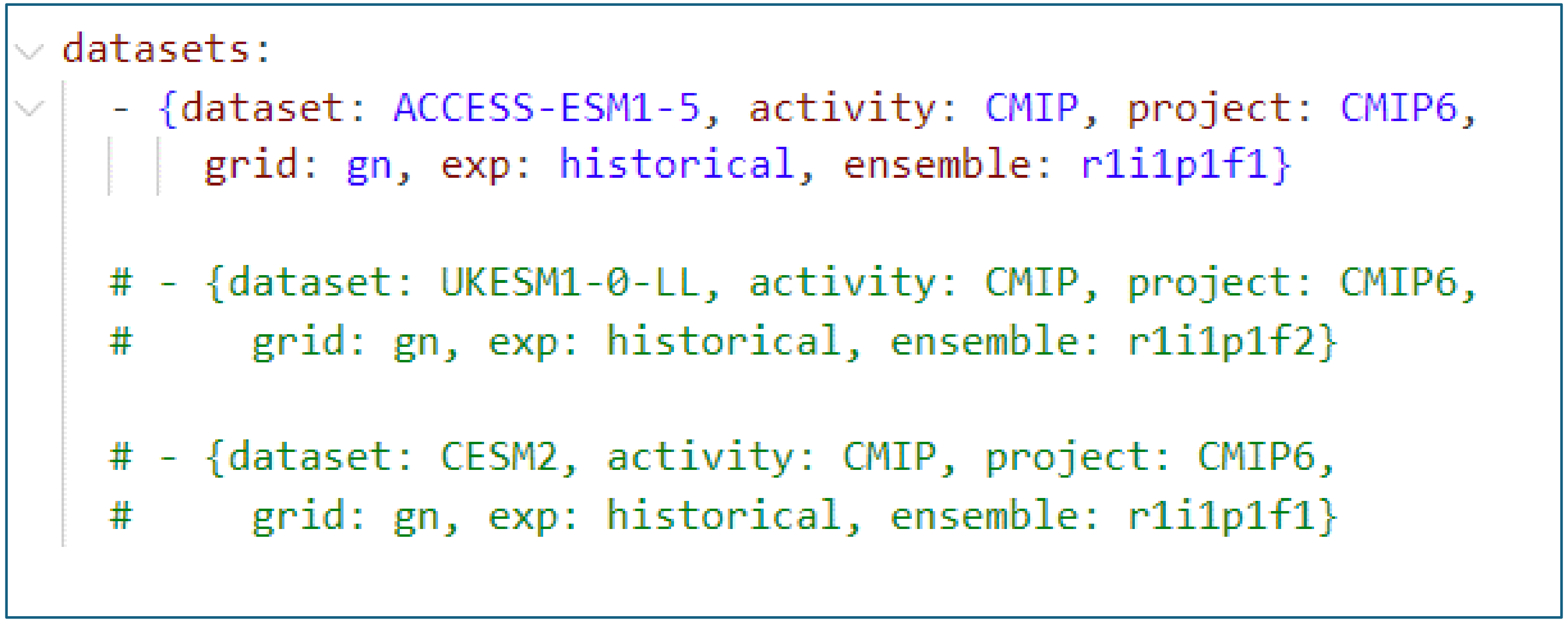


```
documentation:
  description: |
    This is an example recipe for a personal diagnostic
    to be edited.

  authors:
  # author must be in config-references for recipe to run
    - chun_felicity

  title: |
    Example recipe for running a user-written diagnostic.

```

datasets

- list

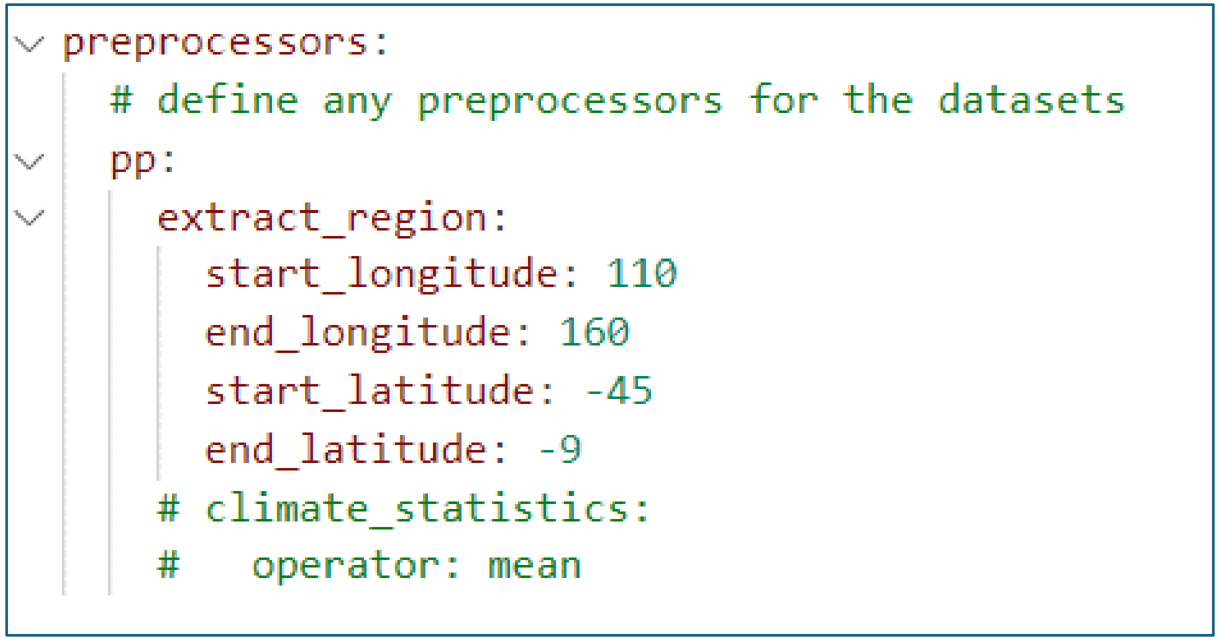
Facets not included here can be defined in variables in diagnostic



```
datasets:
  - {dataset: ACCESS-ESM1-5, activity: CMIP, project: CMIP6,
      grid: gn, exp: historical, ensemble: r1i1p1f1}

  # - {dataset: UKESM1-0-LL, activity: CMIP, project: CMIP6,
  #     grid: gn, exp: historical, ensemble: r1i1p1f2}

  # - {dataset: CESM2, activity: CMIP, project: CMIP6,
  #     grid: gn, exp: historical, ensemble: r1i1p1f1}
```

preprocessors

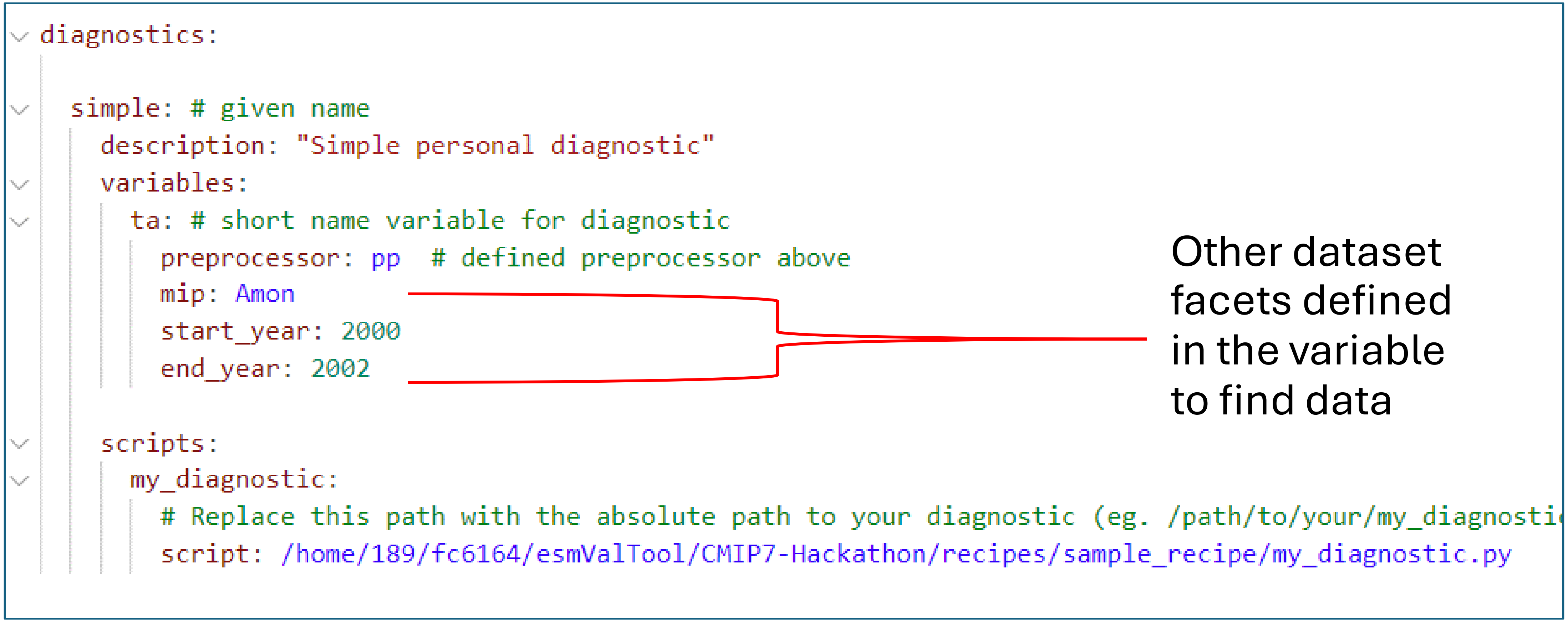
Define a ‘preprocessor’ to be used in the diagnostic



```yaml
preprocessors:
  # define any preprocessors for the datasets
  pp:
    extract_region:
      start_longitude: 110
      end_longitude: 160
      start_latitude: -45
      end_latitude: -9
    # climate_statistics:
    #   operator: mean
```

https://docs.esmvaltool.org/projects/ESMValCore/en/latest/recipe/preprocessor.html

diagnostics



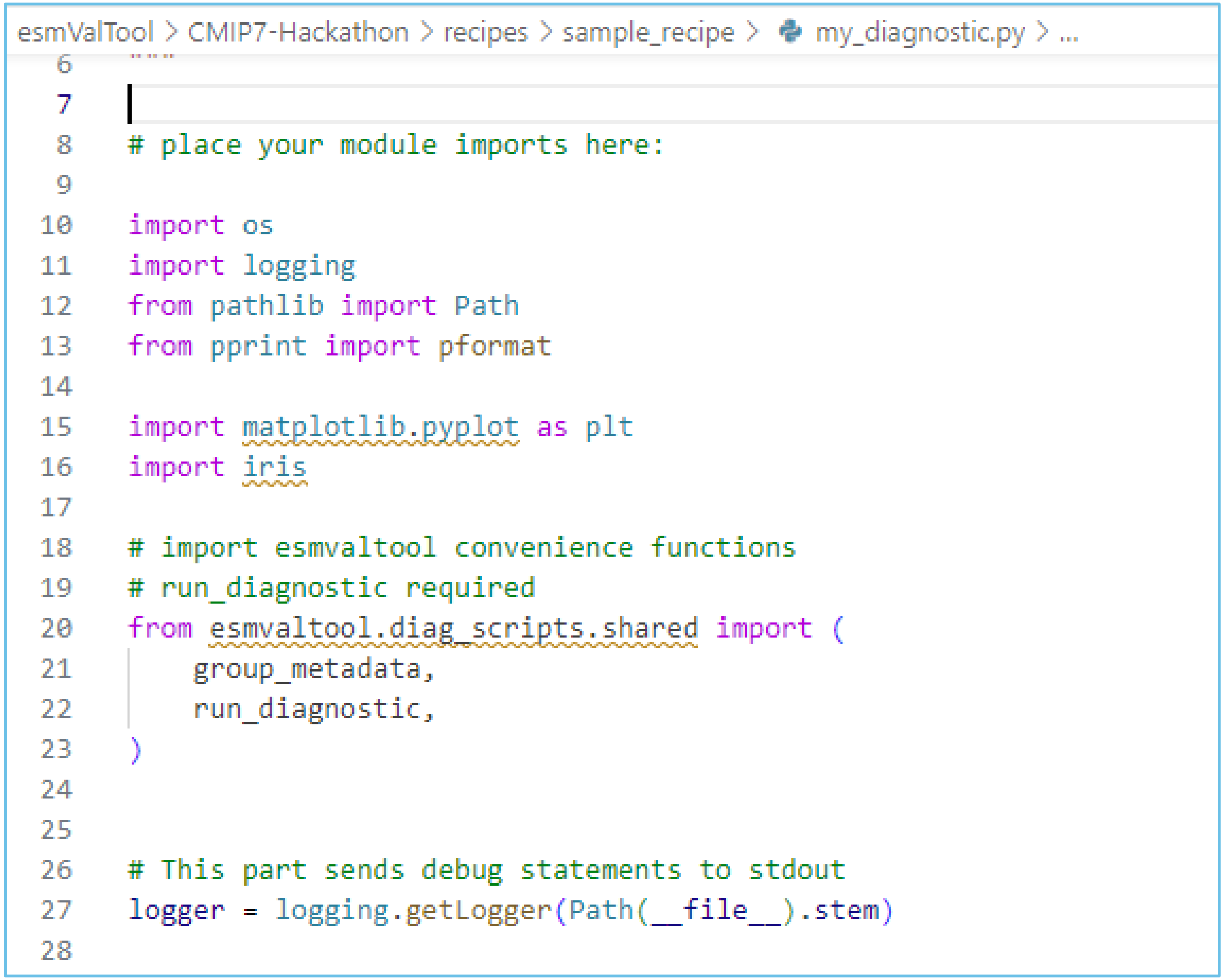
```
diagnostics:

  simple: # given name
    description: "Simple personal diagnostic"
    variables:
      ta: # short name variable for diagnostic
        preprocessor: pp  # defined preprocessor above
        mip: Amon
        start_year: 2000
        end_year: 2002

    scripts:
      my_diagnostic:
        # Replace this path with the absolute path to your diagnostic (eg. /path/to/your/my_diagnosti
        script: /home/189/fc6164/esmValTool/CMIP7-Hackathon/recipes/sample_recipe/my_diagnostic.py
```

Other dataset facets defined in the variable to find data

scripts

Python script example

Import modules including *esmvaltool* functions

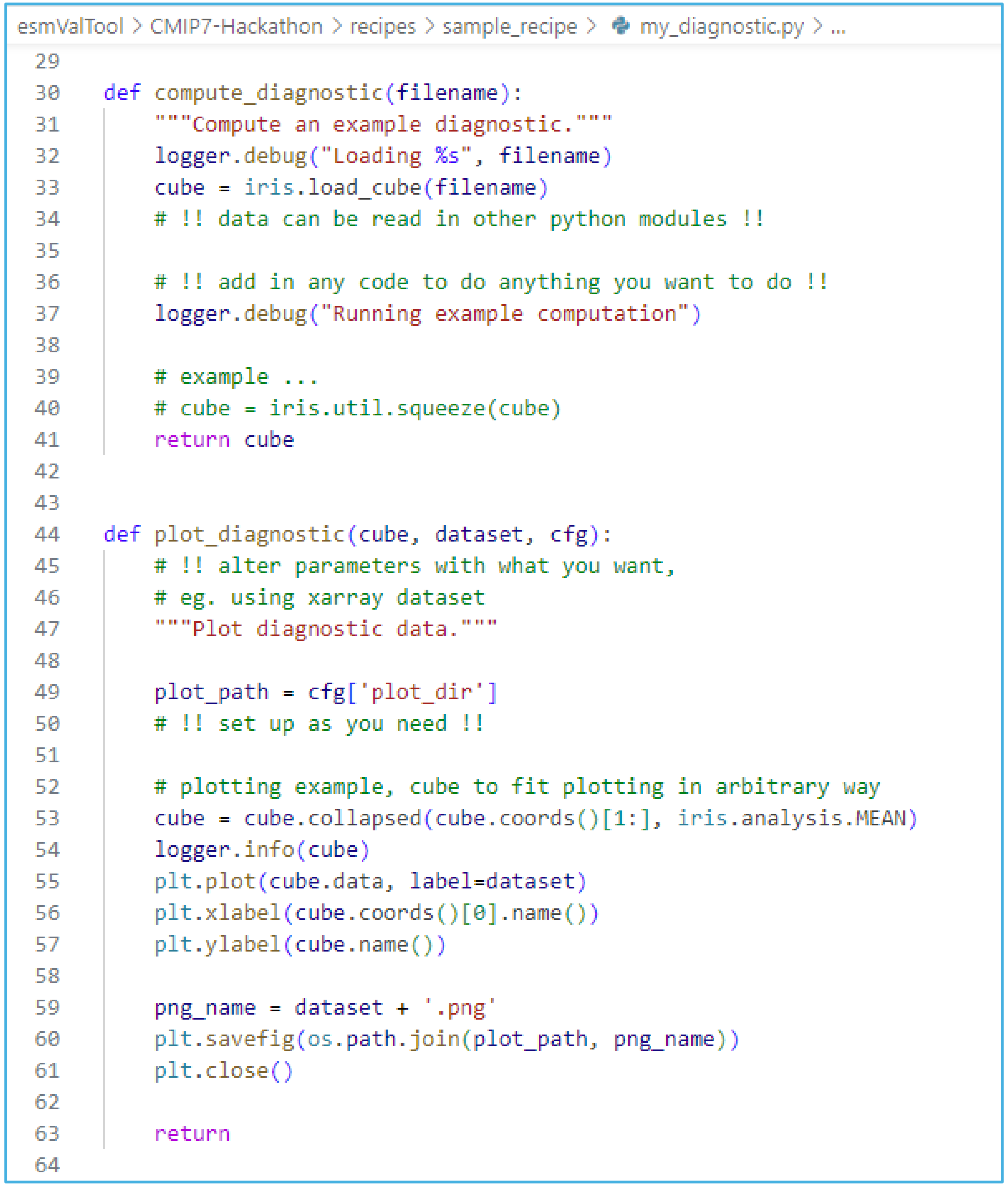

esmValTool > CMIP7-Hackathon > recipes > sample_recipe > my_diagnostic.py > ...

```python
"""

# place your module imports here:

import os
import logging
from pathlib import Path
from pprint import pformat

import matplotlib.pyplot as plt
import iris

# import esmvaltool convenience functions
# run_diagnostic required
from esmvaltool.diag_scripts.shared import (
    group_metadata,
    run_diagnostic,
)


# This part sends debug statements to stdout
logger = logging.getLogger(Path(__file__).stem)

```

scripts

Python script example cont.

Functions created that are called in main

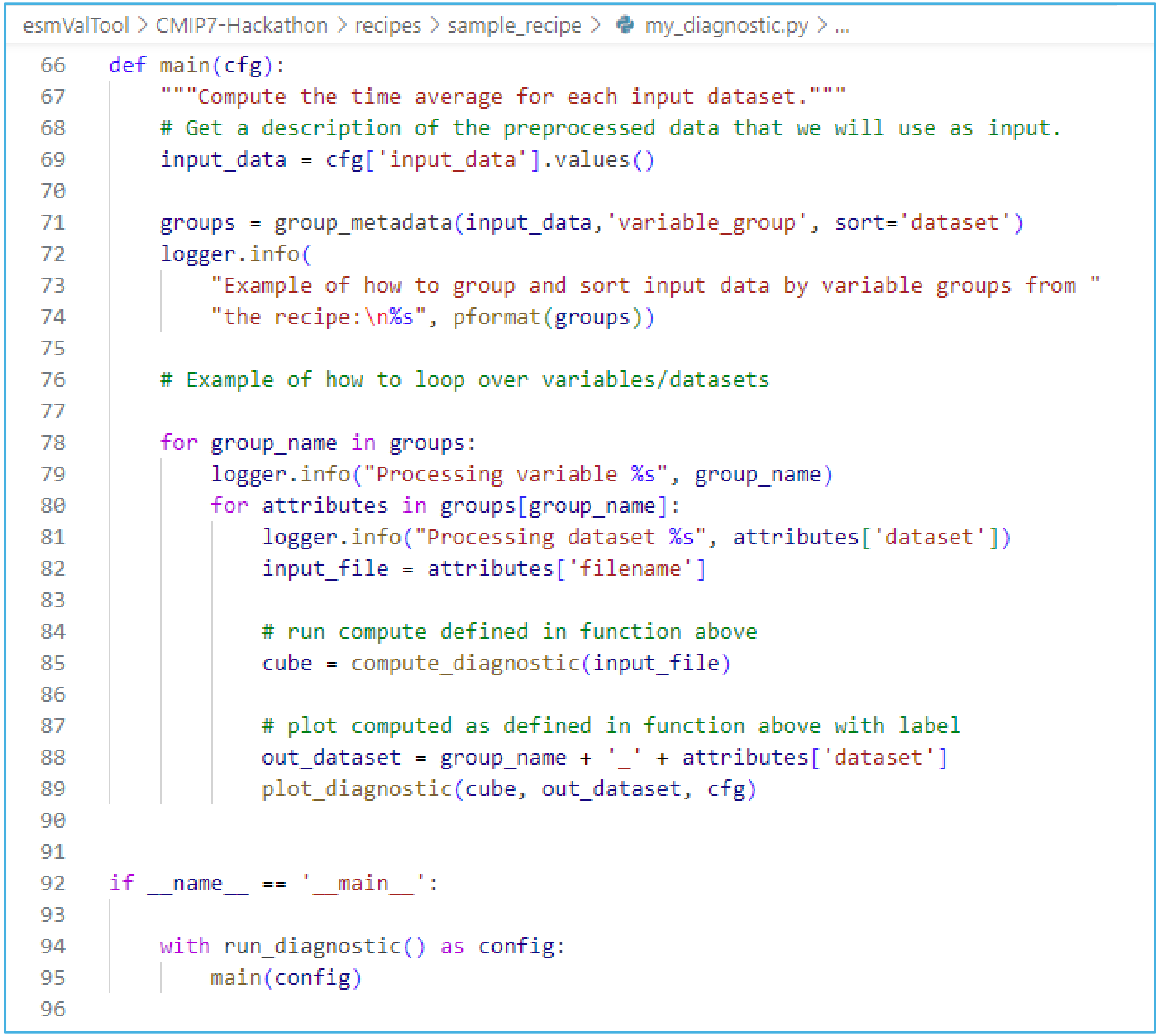

esmValTool > CMIP7-Hackathon > recipes > sample_recipe > my_diagnostic.py > ...

```python

def compute_diagnostic(filename):
    """Compute an example diagnostic."""
    logger.debug("Loading %s", filename)
    cube = iris.load_cube(filename)
    # !! data can be read in other python modules !!

    # !! add in any code to do anything you want to do !!
    logger.debug("Running example computation")

    # example ...
    # cube = iris.util.squeeze(cube)
    return cube


def plot_diagnostic(cube, dataset, cfg):
    # !! alter parameters with what you want,
    # eg. using xarray dataset
    """Plot diagnostic data."""

    plot_path = cfg['plot_dir']
    # !! set up as you need !!

    # plotting example, cube to fit plotting in arbitrary way
    cube = cube.collapsed(cube.coords()[1:], iris.analysis.MEAN)
    logger.info(cube)
    plt.plot(cube.data, label=dataset)
    plt.xlabel(cube.coords()[0].name())
    plt.ylabel(cube.name())

    png_name = dataset + '.png'
    plt.savefig(os.path.join(plot_path, png_name))
    plt.close()

    return

```

scripts

Python script example cont.

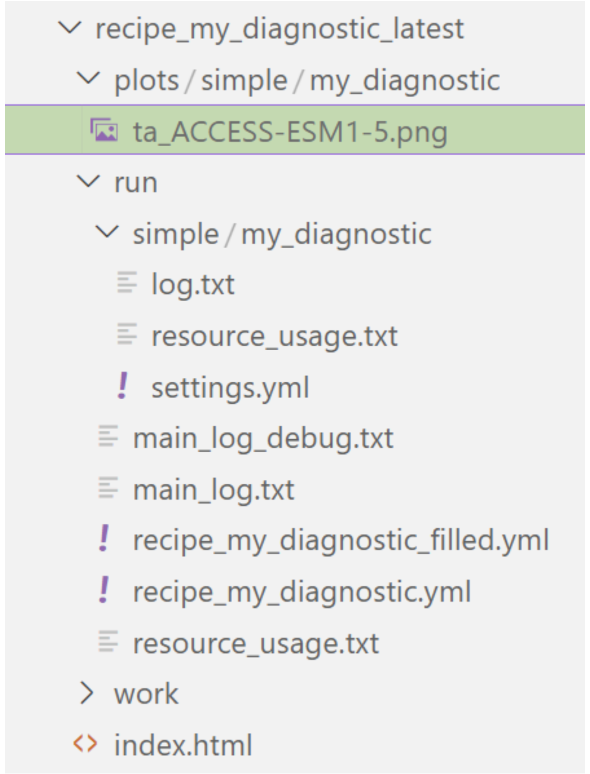
main function which takes config from recipe


esmValTool > CMIP7-Hackathon > recipes > sample_recipe > my_diagnostic.py > ...

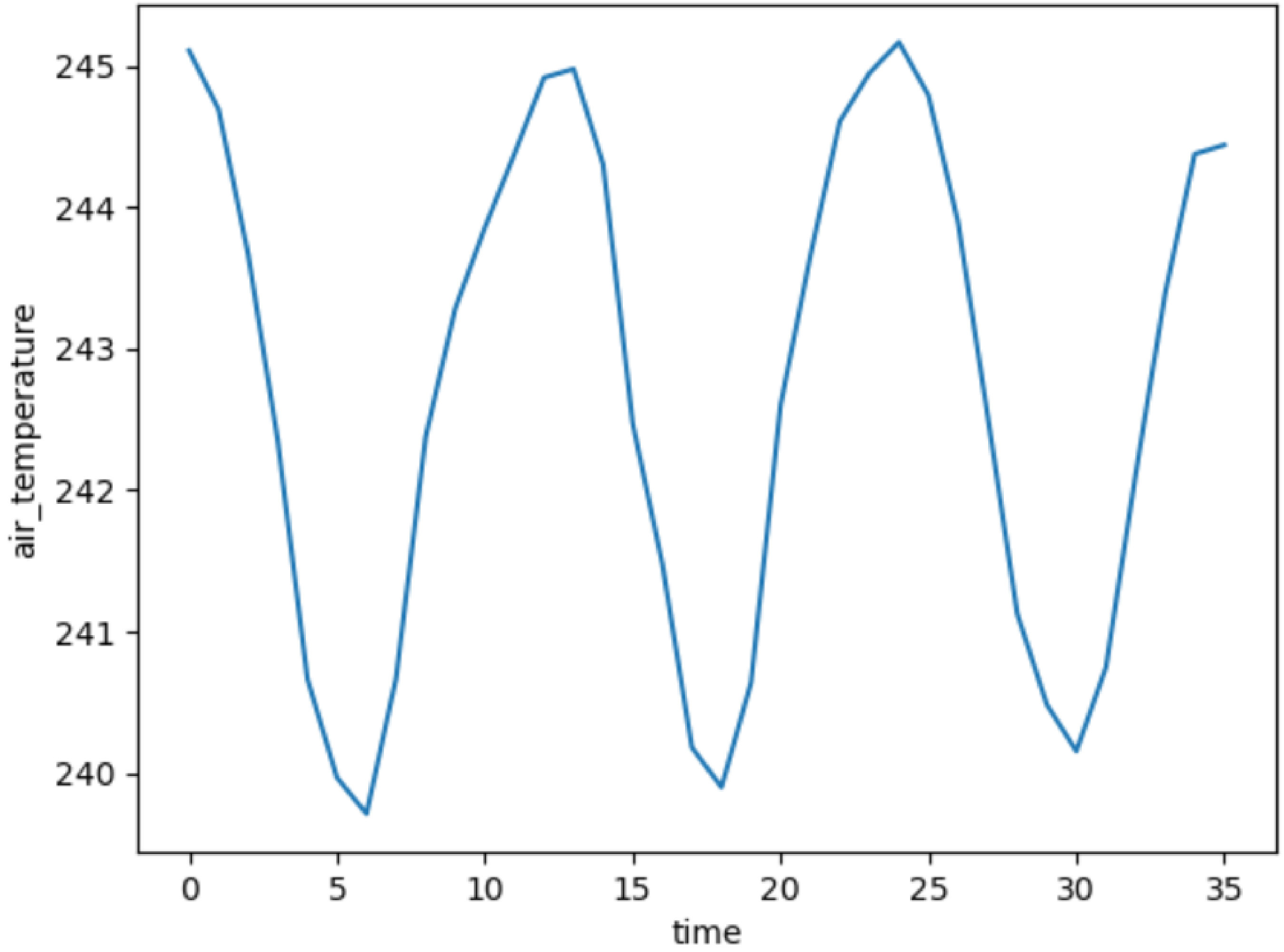
```python
def main(cfg):
    """Compute the time average for each input dataset."""
    # Get a description of the preprocessed data that we will use as input.
    input_data = cfg['input_data'].values()

    groups = group_metadata(input_data,'variable_group', sort='dataset')
    logger.info(
        "Example of how to group and sort input data by variable groups from "
        "the recipe:\n%s", pformat(groups))

    # Example of how to loop over variables/datasets

    for group_name in groups:
        logger.info("Processing variable %s", group_name)
        for attributes in groups[group_name]:
            logger.info("Processing dataset %s", attributes['dataset'])
            input_file = attributes['filename']

            # run compute defined in function above
            cube = compute_diagnostic(input_file)

            # plot computed as defined in function above with label
            out_dataset = group_name + '_' + attributes['dataset']
            plot_diagnostic(cube, out_dataset, cfg)


if __name__ == '__main__':

    with run_diagnostic() as config:
        main(config)

```

output

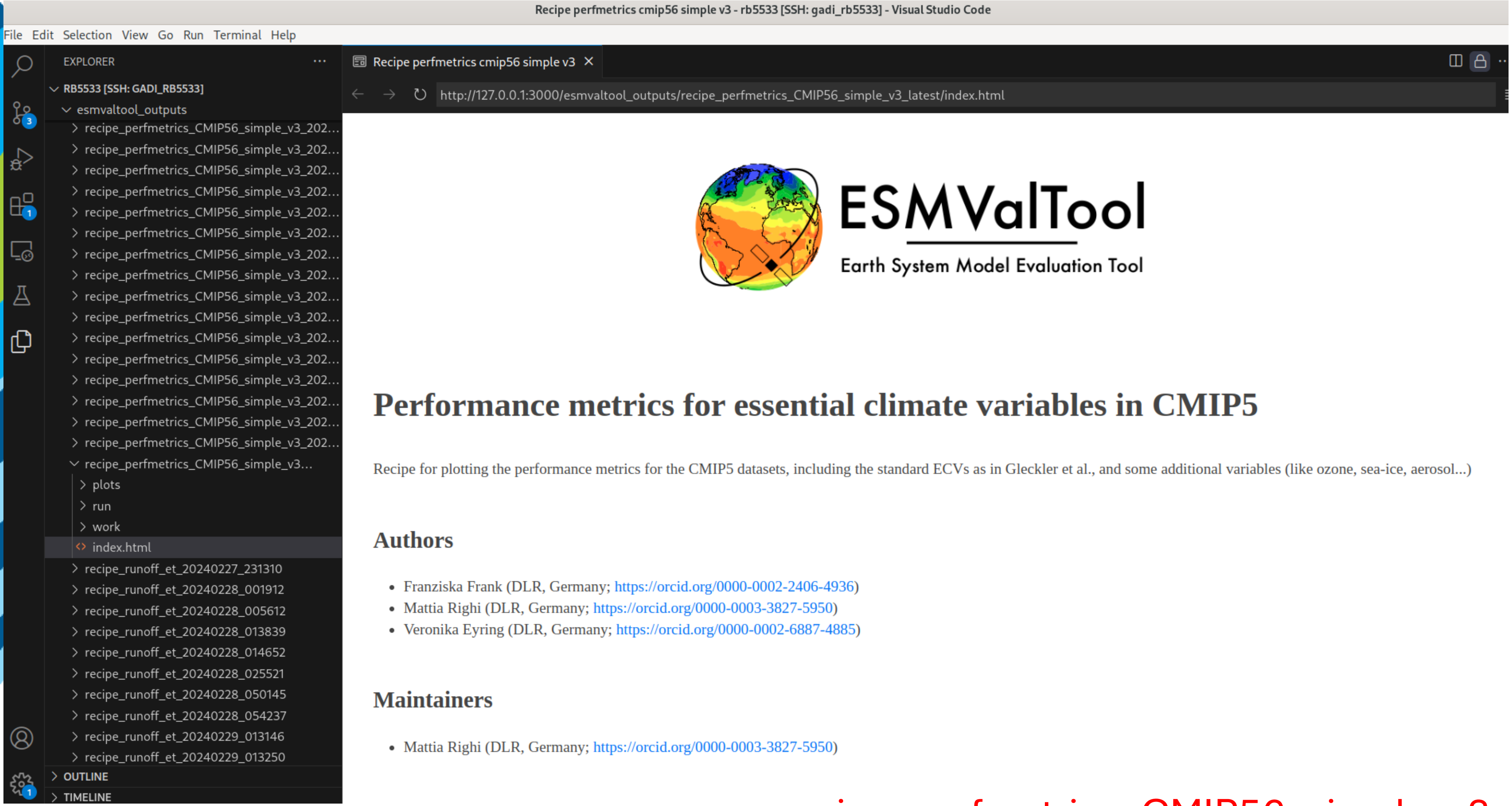
esmvaltool_outputs > recipe_my_diagnostic_latest > plots > simple > my_diagnostic > ta_ACCESS-ESM1-5.png



```
v recipe_my_diagnostic_latest
  v plots / simple / my_diagnostic
      ta_ACCESS-ESM1-5.png
  v run
    v simple / my_diagnostic
        log.txt
        resource_usage.txt
      ! settings.yml
      main_log_debug.txt
      main_log.txt
    ! recipe_my_diagnostic_filled.yml
    ! recipe_my_diagnostic.yml
      resource_usage.txt
  > work
  <> index.html
```



General CMIP Performance metrics



Recipe perfmetrics cmip56 simple v3 - rb5533 [SSH: gadi_rb5533] - Visual Studio Code
File Edit Selection View Go Run Terminal Help
EXPLORER
RB5533 [SSH: GADI_RB5533]
esmvaltool_outputs
recipe_perfmetrics_CMIP56_simple_v3_202...
recipe_perfmetrics_CMIP56_simple_v3...
plots
run
work
index.html
recipe_runoff_et_20240227_231310
recipe_runoff_et_20240228_001912
recipe_runoff_et_20240228_005612
recipe_runoff_et_20240228_013839
recipe_runoff_et_20240228_014652
recipe_runoff_et_20240228_025521
recipe_runoff_et_20240228_050145
recipe_runoff_et_20240228_054237
recipe_runoff_et_20240229_013146
recipe_runoff_et_20240229_013250
OUTLINE
TIMELINE
Recipe perfmetrics cmip56 simple v3
http://127.0.0.1:3000/esmvaltool_outputs/recipe_perfmetrics_CMIP56_simple_v3_latest/index.html
ESMValTool
Earth System Model Evaluation Tool

Performance metrics for essential climate variables in CMIP5

Recipe for plotting the performance metrics for the CMIP5 datasets, including the standard ECVs as in Gleckler et al., and some additional variables (like ozone, sea-ice, aerosol...)

Authors

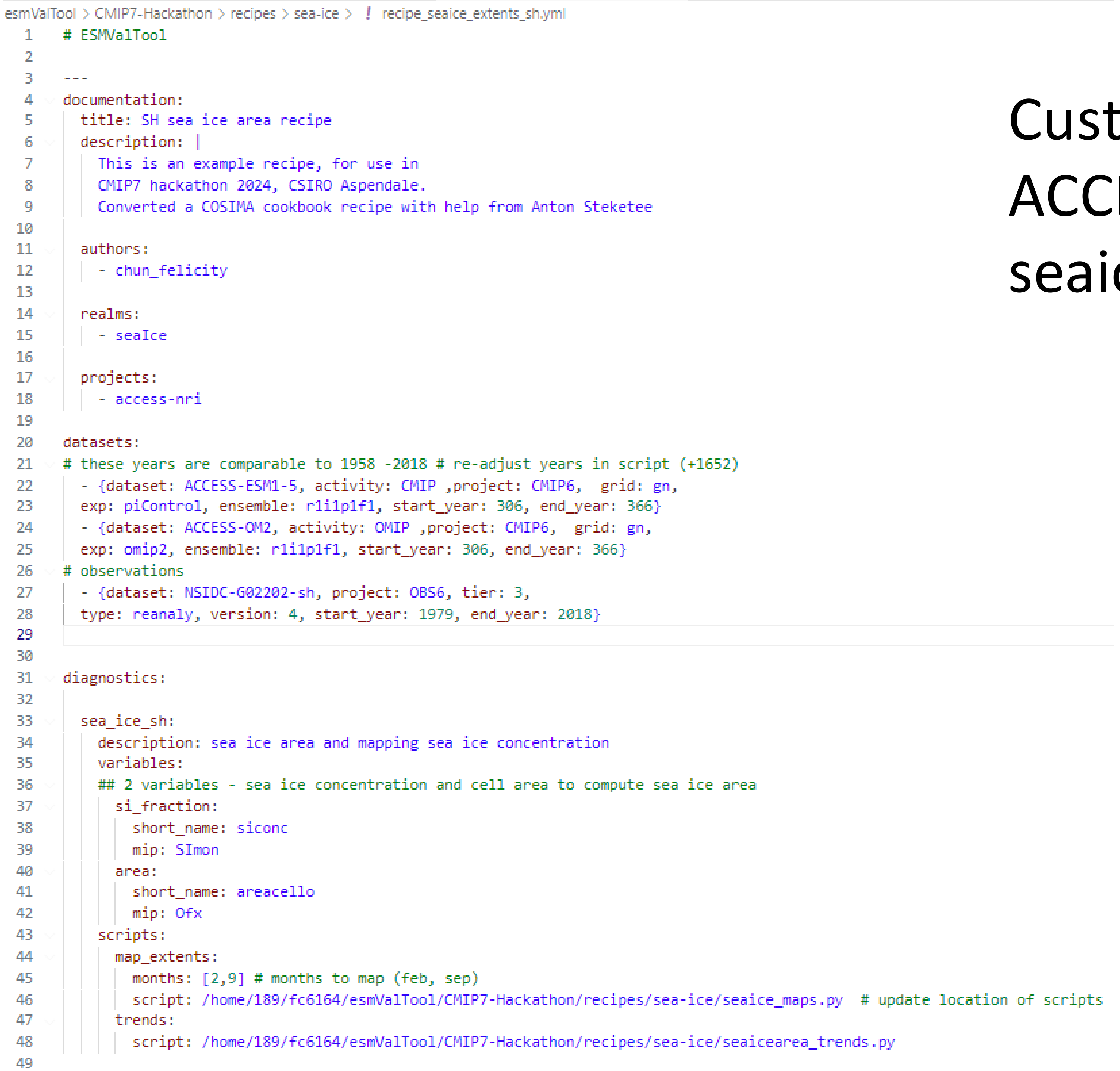
- Franziska Frank (DLR, Germany; https://orcid.org/0000-0002-2406-4936)
- Mattia Righi (DLR, Germany; https://orcid.org/0000-0003-3827-5950)
- Veronika Eyring (DLR, Germany; https://orcid.org/0000-0002-6887-4885)

Maintainers

- Mattia Righi (DLR, Germany; https://orcid.org/0000-0003-3827-5950)

recipe_perfmetrics_CMIP56_simple_v3.yml

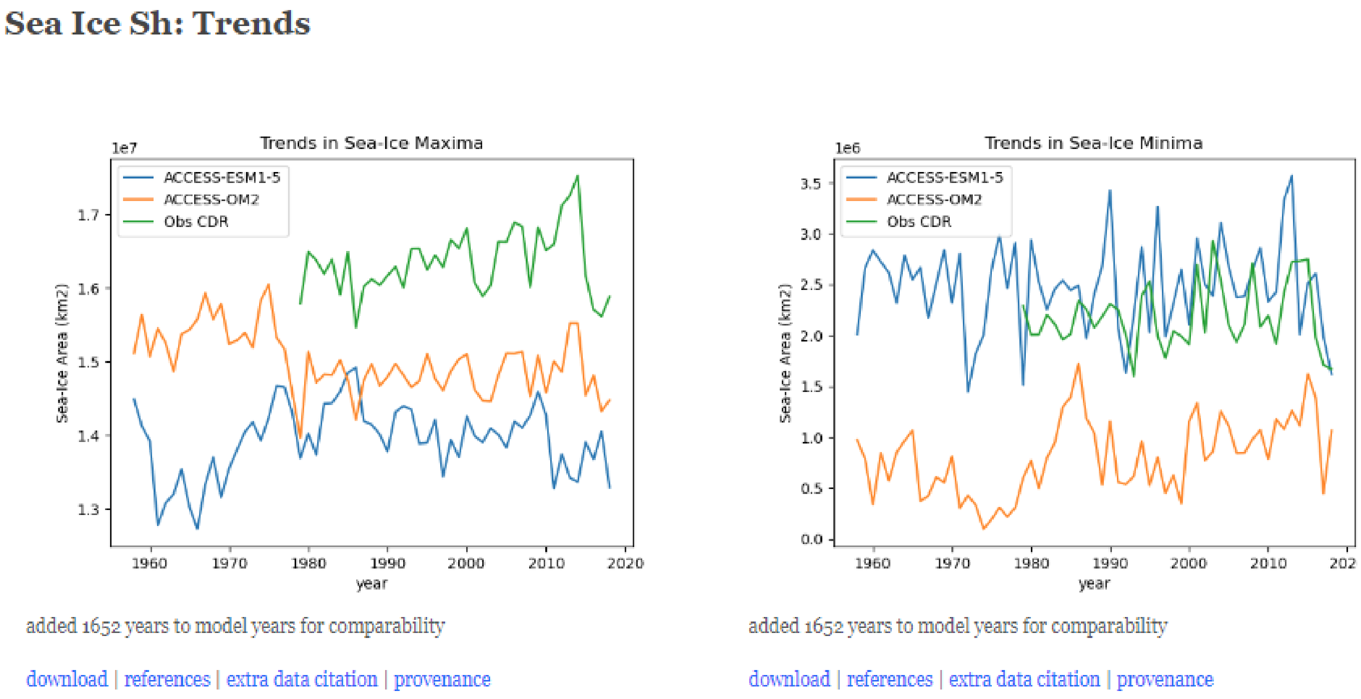
Custom ACCESS-NRI recipe seaice extents sh



esmValTool > CMIP7-Hackathon > recipes > sea-ice > ! recipe_seaice_extents_sh.yml

```yaml
# ESMValTool

---
documentation:
  title: SH sea ice area recipe
  description: |
    This is an example recipe, for use in
    CMIP7 hackathon 2024, CSIRO Aspendale.
    Converted a COSIMA cookbook recipe with help from Anton Steketee

  authors:
    - chun_felicity

  realms:
    - seaIce

  projects:
    - access-nri

datasets:
# these years are comparable to 1958 -2018 # re-adjust years in script (+1652)
  - {dataset: ACCESS-ESM1-5, activity: CMIP ,project: CMIP6,  grid: gn,
  exp: piControl, ensemble: r1i1p1f1, start_year: 306, end_year: 366}
  - {dataset: ACCESS-OM2, activity: OMIP ,project: CMIP6,  grid: gn,
  exp: omip2, ensemble: r1i1p1f1, start_year: 306, end_year: 366}
# observations
  - {dataset: NSIDC-G02202-sh, project: OBS6, tier: 3,
  type: reanaly, version: 4, start_year: 1979, end_year: 2018}


diagnostics:

  sea_ice_sh:
    description: sea ice area and mapping sea ice concentration
    variables:
    ## 2 variables - sea ice concentration and cell area to compute sea ice area
      si_fraction:
        short_name: siconc
        mip: SImon
      area:
        short_name: areacello
        mip: Ofx
    scripts:
      map_extents:
        months: [2,9] # months to map (feb, sep)
        script: /home/189/fc6164/esmValTool/CMIP7-Hackathon/recipes/sea-ice/seaice_maps.py  # update location of scripts
      trends:
        script: /home/189/fc6164/esmValTool/CMIP7-Hackathon/recipes/sea-ice/seaicearea_trends.py

```

Output



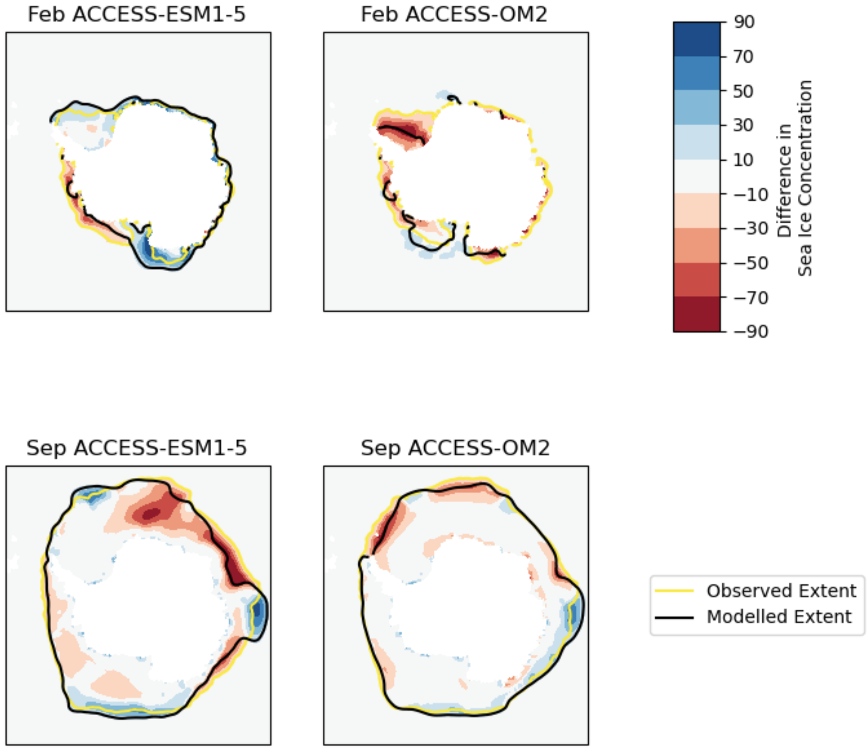
Sea Ice Sh: Trends

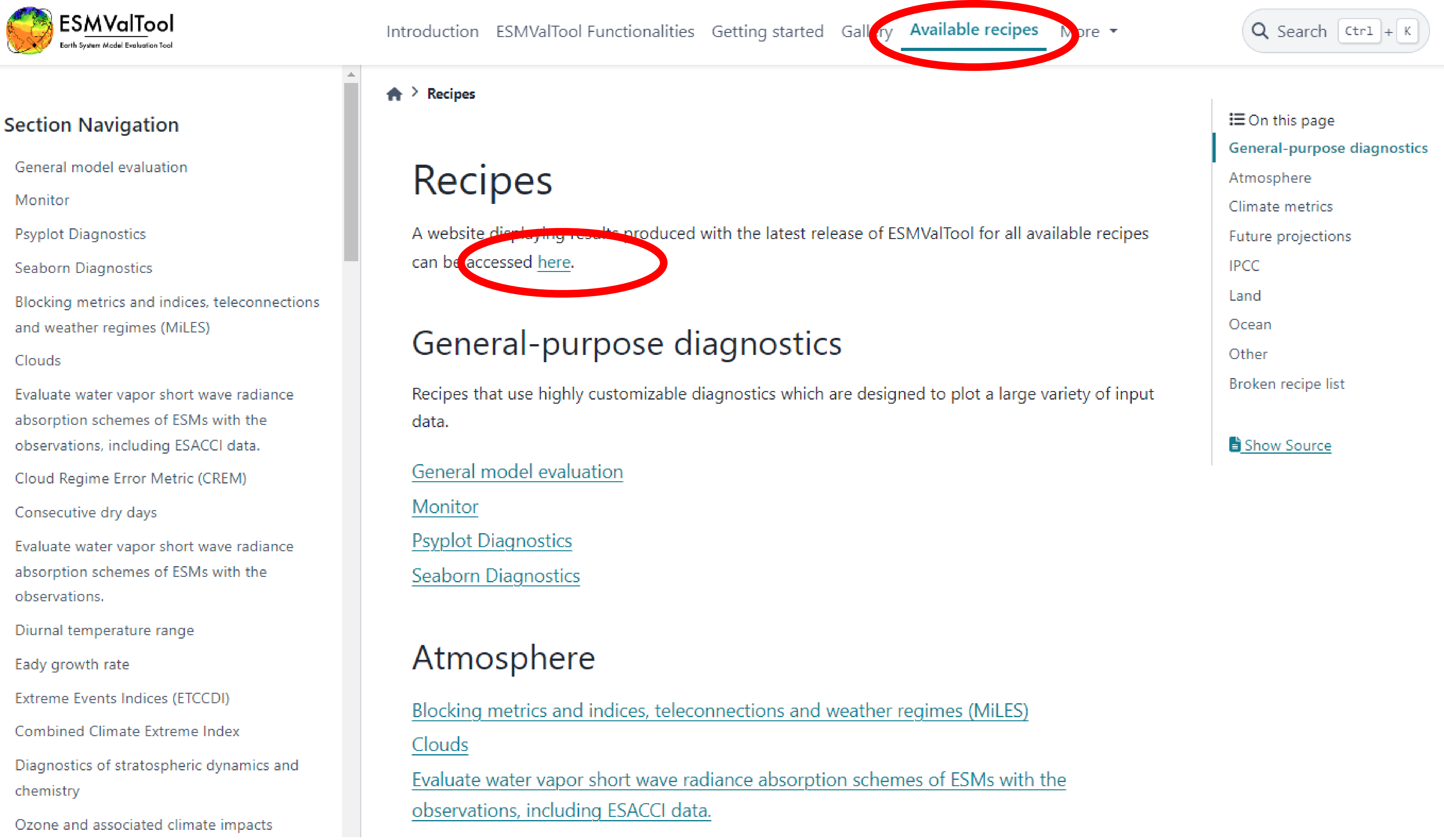
added 1652 years to model years for comparability

download | references | extra data citation | provenance

added 1652 years to model years for comparability

download | references | extra data citation | provenance





Recipes

Section Navigation

- General model evaluation
- Monitor
- Psyplot Diagnostics
- Seaborn Diagnostics
- Blocking metrics and indices, teleconnections and weather regimes (MiLES)
- Clouds
- Evaluate water vapor short wave radiance absorption schemes of ESMs with the observations, including ESACCI data.
- Cloud Regime Error Metric (CREM)
- Consecutive dry days
- Evaluate water vapor short wave radiance absorption schemes of ESMs with the observations.
- Diurnal temperature range
- Eady growth rate
- Extreme Events Indices (ETCCDI)
- Combined Climate Extreme Index
- Diagnostics of stratospheric dynamics and chemistry
- Ozone and associated climate impacts

Recipes

A website displaying results produced with the latest release of ESMValTool for all available recipes can be accessed here.

General-purpose diagnostics

Recipes that use highly customizable diagnostics which are designed to plot a large variety of input data.

General model evaluation

Monitor

Psyplot Diagnostics

Seaborn Diagnostics

Atmosphere

Blocking metrics and indices, teleconnections and weather regimes (MiLES)

Clouds

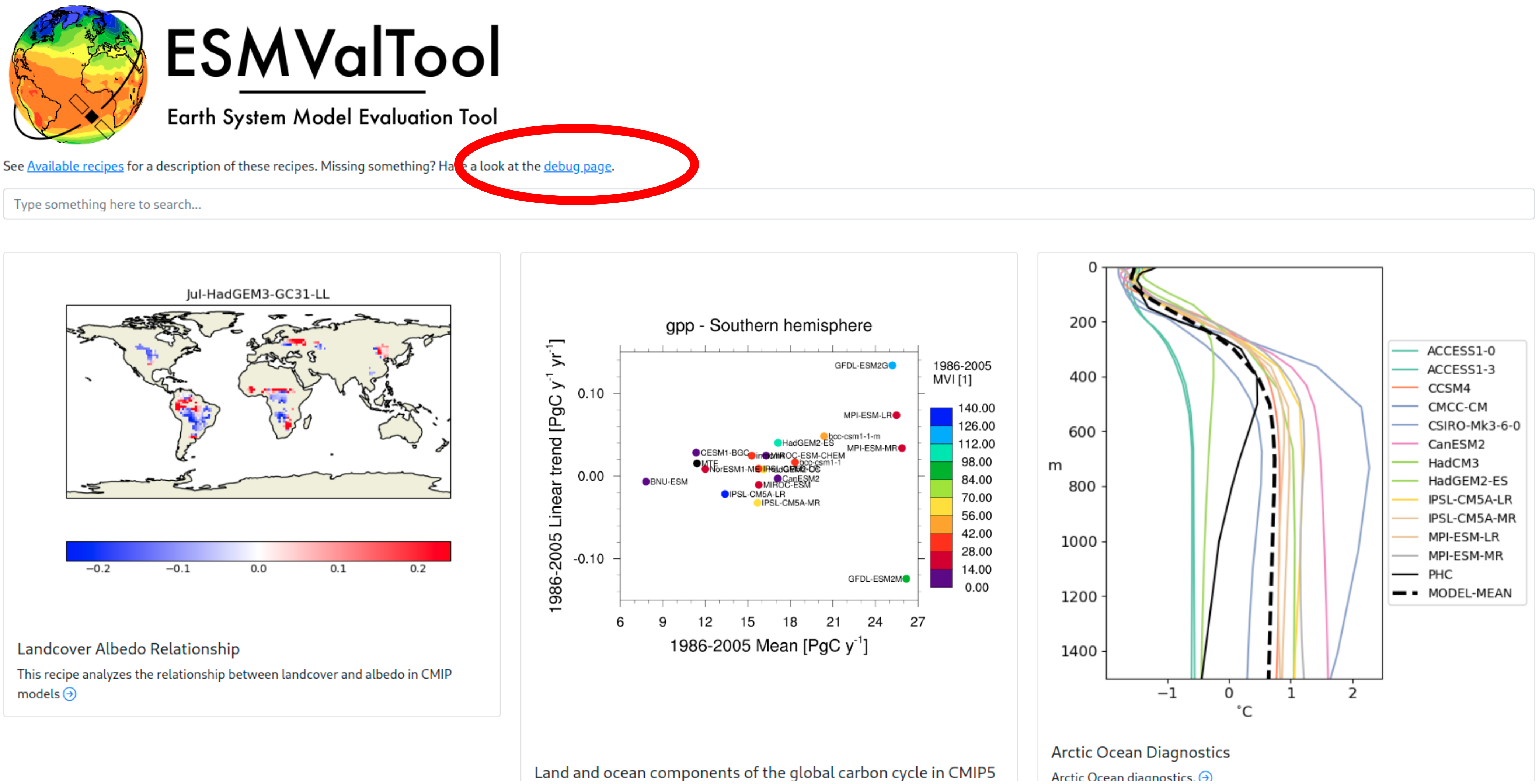
Evaluate water vapor short wave radiance absorption schemes of ESMs with the observations, including ESACCI data.

On this page

- General-purpose diagnostics
- Atmosphere
- Climate metrics
- Future projections
- IPCC
- Land
- Ocean
- Other
- Broken recipe list

Show Source

https://docs.esmvaltool.org/en/latest/recipes/index.html



ESMValTool

Earth System Model Evaluation Tool

See Available recipes for a description of these recipes. Missing something? Have a look at the debug page.

Type something here to search...

Landcover Albedo Relationship

This recipe analyzes the relationship between landcover and albedo in CMIP models

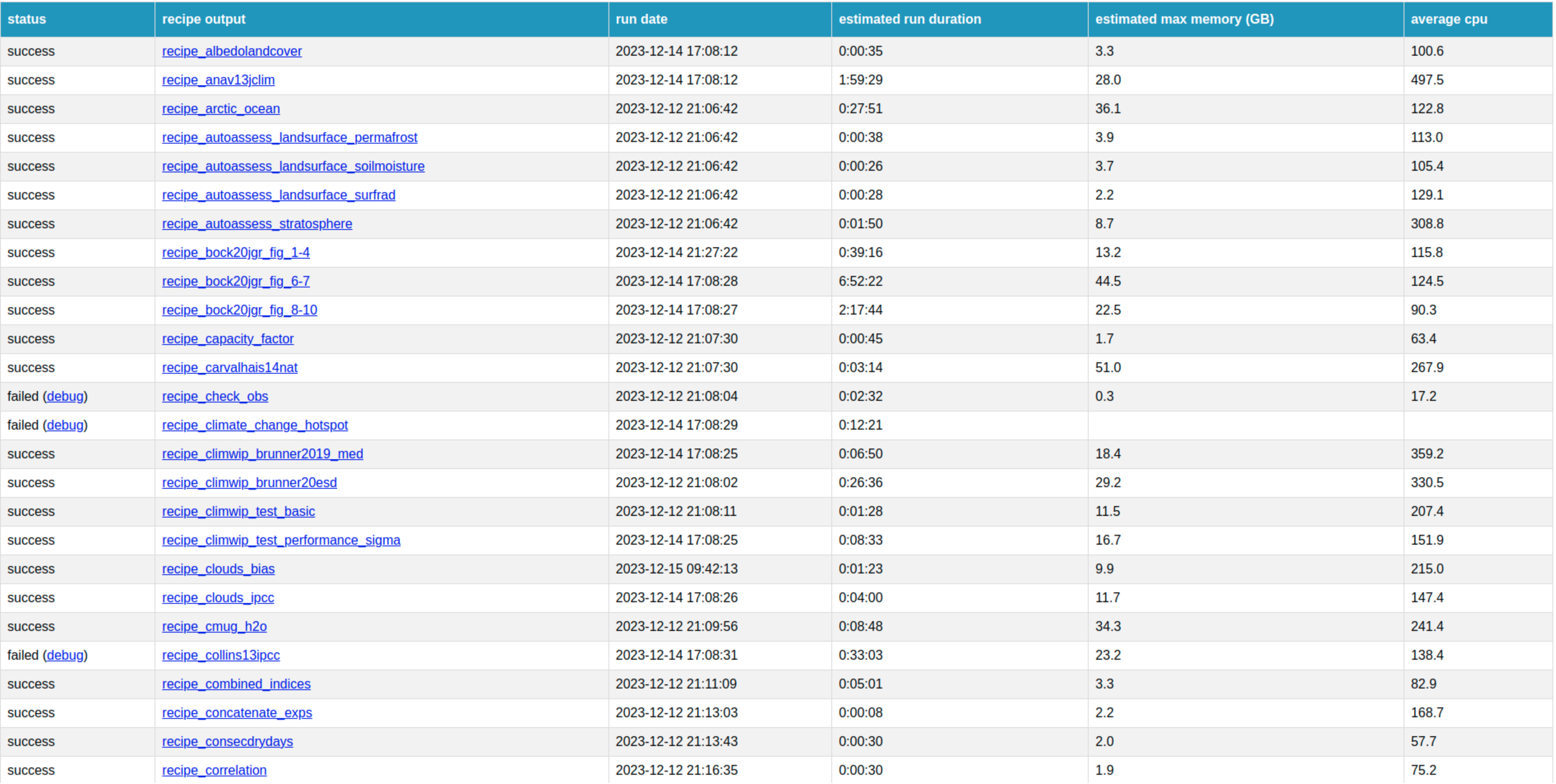
Land and ocean components of the global carbon cycle in CMIP5

Arctic Ocean Diagnostics

Arctic Ocean diagnostics.

https://esmvaltool.dkrz.de/shared/esmvaltool/stable_release/

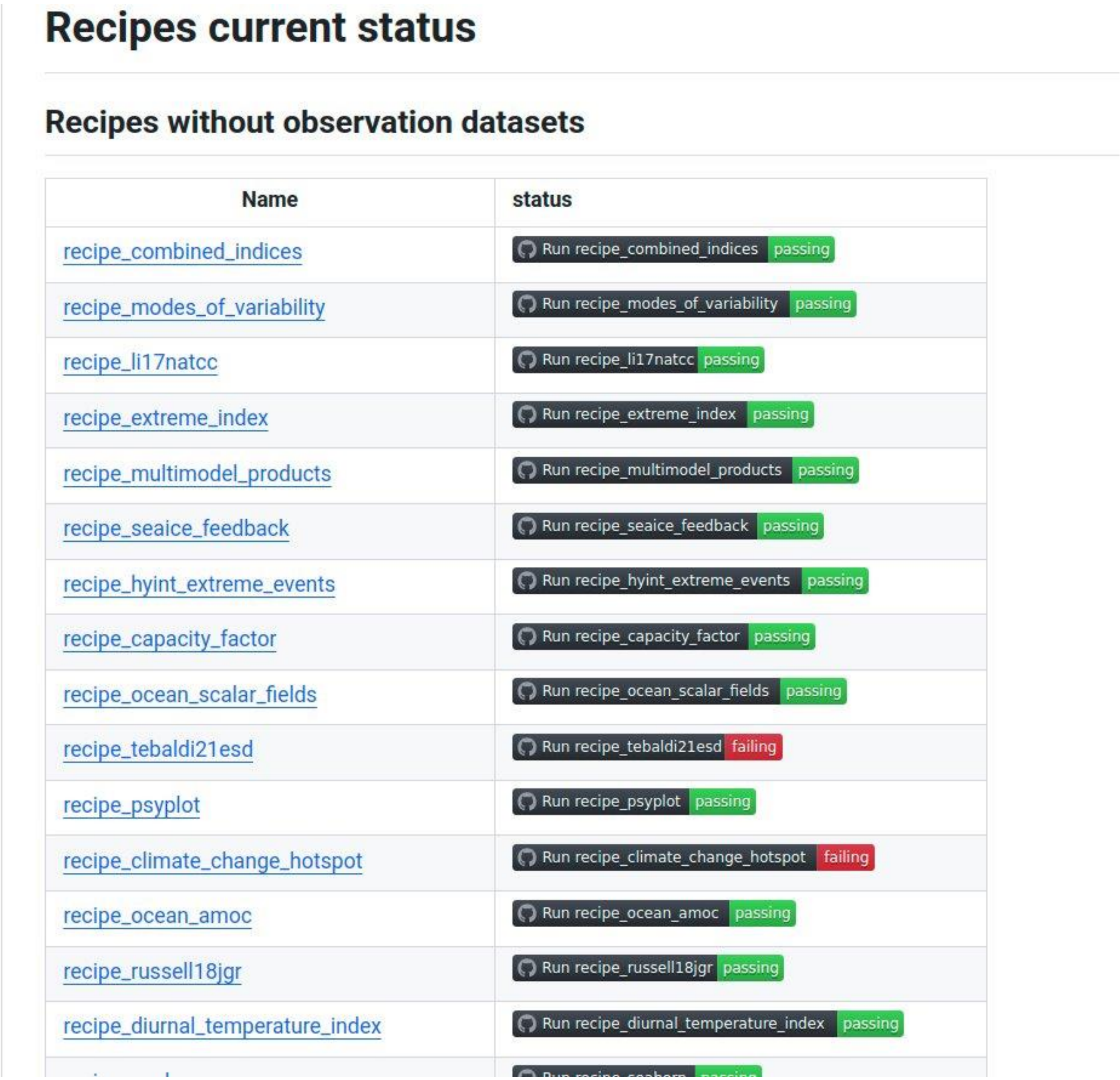
Recipe status and estimated resources at DKRZ



status	recipe output	run date	estimated run duration	estimated max memory (GB)	average cpu
success	recipe_albedolandcover	2023-12-14 17:08:12	0:00:35	3.3	100.6
success	recipe_anav13jclim	2023-12-14 17:08:12	1:59:29	28.0	497.5
success	recipe_arctic_ocean	2023-12-12 21:06:42	0:27:51	36.1	122.8
success	recipe_autoassess_landsurface_permafrost	2023-12-12 21:06:42	0:00:38	3.9	113.0
success	recipe_autoassess_landsurface_soilmoisture	2023-12-12 21:06:42	0:00:26	3.7	105.4
success	recipe_autoassess_landsurface_surfrad	2023-12-12 21:06:42	0:00:28	2.2	129.1
success	recipe_autoassess_stratosphere	2023-12-12 21:06:42	0:01:50	8.7	308.8
success	recipe_bock20jgr_fig_1-4	2023-12-14 21:27:22	0:39:16	13.2	115.8
success	recipe_bock20jgr_fig_6-7	2023-12-14 17:08:28	6:52:22	44.5	124.5
success	recipe_bock20jgr_fig_8-10	2023-12-14 17:08:27	2:17:44	22.5	90.3
success	recipe_capacity_factor	2023-12-12 21:07:30	0:00:45	1.7	63.4
success	recipe_carvalhais14nat	2023-12-12 21:07:30	0:03:14	51.0	267.9
failed (debug)	recipe_check_obs	2023-12-12 21:08:04	0:02:32	0.3	17.2
failed (debug)	recipe_climate_change_hotspot	2023-12-14 17:08:29	0:12:21		
success	recipe_climwip_brunner2019_med	2023-12-14 17:08:25	0:06:50	18.4	359.2
success	recipe_climwip_brunner20esd	2023-12-12 21:08:02	0:26:36	29.2	330.5
success	recipe_climwip_test_basic	2023-12-12 21:08:11	0:01:28	11.5	207.4
success	recipe_climwip_test_performance_sigma	2023-12-14 17:08:25	0:08:33	16.7	151.9
success	recipe_clouds_bias	2023-12-15 09:42:13	0:01:23	9.9	215.0
success	recipe_clouds_ipcc	2023-12-14 17:08:26	0:04:00	11.7	147.4
success	recipe_cmug_h2o	2023-12-12 21:09:56	0:08:48	34.3	241.4
failed (debug)	recipe_collins13ipcc	2023-12-14 17:08:31	0:33:03	23.2	138.4
success	recipe_combined_indices	2023-12-12 21:11:09	0:05:01	3.3	82.9
success	recipe_concatenate_exps	2023-12-12 21:13:03	0:00:08	2.2	168.7
success	recipe_consecdrydays	2023-12-12 21:13:43	0:00:30	2.0	57.7
success	recipe_correlation	2023-12-12 21:16:35	0:00:30	1.9	75.2

https://esmvaltool.dkrz.de/shared/esmvaltool/stable_release/debug.html

Recipe status and on Gadi



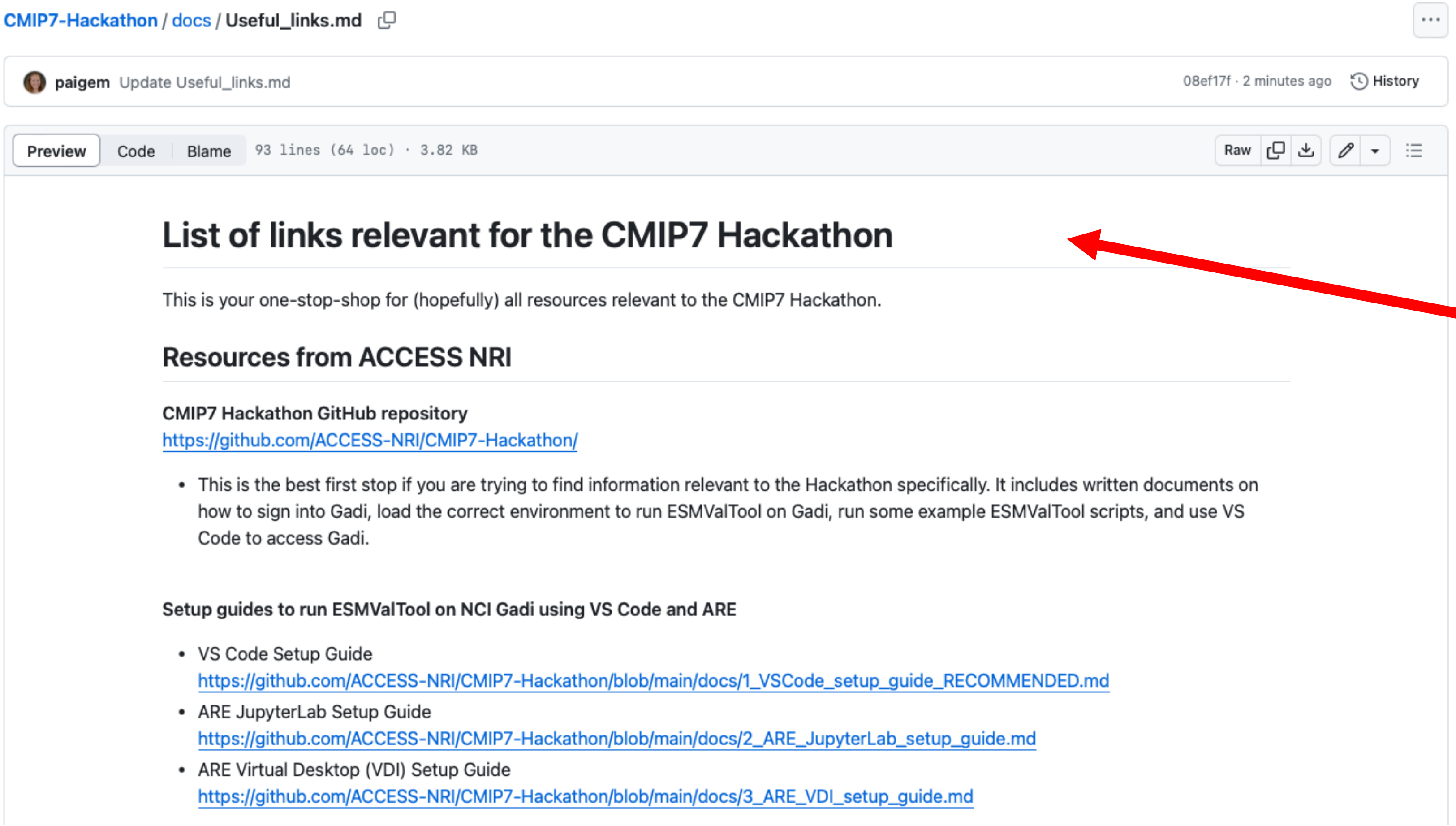
Recipes current status

Recipes without observation datasets

Name	status
recipe_combined_indices	Run recipe_combined_indices passing
recipe_modes_of_variability	Run recipe_modes_of_variability passing
recipe_li17natcc	Run recipe_li17natcc passing
recipe_extreme_index	Run recipe_extreme_index passing
recipe_multimodel_products	Run recipe_multimodel_products passing
recipe_seaice_feedback	Run recipe_seaice_feedback passing
recipe_hyint_extreme_events	Run recipe_hyint_extreme_events passing
recipe_capacity_factor	Run recipe_capacity_factor passing
recipe_ocean_scalar_fields	Run recipe_ocean_scalar_fields passing
recipe_tebaldi21esd	Run recipe_tebaldi21esd failing
recipe_psyplot	Run recipe_psyplot passing
recipe_climate_change_hotspot	Run recipe_climate_change_hotspot failing
recipe_ocean_amoc	Run recipe_ocean_amoc passing
recipe_russell18jgr	Run recipe_russell18jgr passing
recipe_diurnal_temperature_index	Run recipe_diurnal_temperature_index passing
[illegible]	Run recipe_seaborn passing

https://github.com/ACCESS-NRI/ESMValTool-workflow?tab=readme-ov-file#recipes-current-status

Resources available


CMIP7-Hackathon / docs / Useful_links.md
paigem Update Useful_links.md
08ef17f · 2 minutes ago
History
Preview
Code
Blame
93 lines (64 loc) · 3.82 KB
Raw

List of links relevant for the CMIP7 Hackathon

This is your one-stop-shop for (hopefully) all resources relevant to the CMIP7 Hackathon.

Resources from ACCESS NRI

CMIP7 Hackathon GitHub repository
https://github.com/ACCESS-NRI/CMIP7-Hackathon/

- This is the best first stop if you are trying to find information relevant to the Hackathon specifically. It includes written documents on how to sign into Gadi, load the correct environment to run ESMValTool on Gadi, run some example ESMValTool scripts, and use VS Code to access Gadi.

Setup guides to run ESMValTool on NCI Gadi using VS Code and ARE

- VS Code Setup Guide
  https://github.com/ACCESS-NRI/CMIP7-Hackathon/blob/main/docs/1_VSCode_setup_guide_RECOMMENDED.md
- ARE JupyterLab Setup Guide
  https://github.com/ACCESS-NRI/CMIP7-Hackathon/blob/main/docs/2_ARE_JupyterLab_setup_guide.md
- ARE Virtual Desktop (VDI) Setup Guide
  https://github.com/ACCESS-NRI/CMIP7-Hackathon/blob/main/docs/3_ARE_VDI_setup_guide.md

Single document with links to useful resources for the Hackathon and ESMValTool

CMIP7-Hackathon / docs / Useful_links.md

https://github.com/ACCESS-NRI/CMIP7-Hackathon/blob/main/docs/Useful_links.md