



Call: H2020-SC5-2014-two-stage

Topic: SC5-01-2014

PRIMAVERA

Grant Agreement 641727



**PRocess-based climate sIMulation: AdVances in high resolution modelling and
European climate Risk Assessment**

Deliverable D1.3

Strategy and metrics for PRIMAVERA and CMIP6 experiments analysis

Deliverable Title	<i>Strategy and metrics for PRIMAVERA and CMIP6 experiments analysis</i>	
Brief Description	Strategy and package of metrics, in project repository coded using freely available software, for the analysis of the PRIMAVERA and CMIP6 experiments, with a focus on European climate variability and change.	
WP number	1	
Lead Beneficiary	<i>Francisco J. Doblas-Reyes (BSC)</i>	
Contributors	<i>Javier Vegas-Regidor (BSC)</i> <i>Saskia Loosveldt-Tomas (BSC)</i> <i>Pierre-Antoine Bretonnière (BSC)</i> <i>Louis-Philippe Caron (BSC)</i> <i>Jean-Philippe Baudouin (BSC)</i> <i>Eleftheria Exarchou (BSC)</i> <i>Neven Fučkar (BSC)</i> <i>Julien Boé (CERFACS).</i> <i>Emilia Sanchez-Gomez (CERFACS)</i> <i>Panos Athanasiadis (CMCC)</i> <i>Reinhard Schiemann (URead)</i> <i>Irene Mavilia (CNR)</i> <i>Federico Fabiano (CNR)</i> <i>David Docquier (UCL)</i> <i>Marie-Estelle Demory (URead)</i> <i>Benoit Vannière (URead)</i> <i>Ramón Fuentes Franco (SMHI)</i> <i>Alexander Baker (URead)</i> <i>François Massonnet (UCL)</i>	
Creation Date	19/12/2019	
Version Number	2.0.0	
Version Date	29/01/2012	
Deliverable Due Date	31/10/2019	
Actual Delivery Date	30/01/2020	
Nature of the Deliverable	<input checked="" type="checkbox"/>	<i>R - Report</i>
	<input type="checkbox"/>	<i>P - Prototype</i>
	<input type="checkbox"/>	<i>D - Demonstrator</i>
	<input type="checkbox"/>	<i>O - Other</i>
Dissemination Level/ Audience	<input checked="" type="checkbox"/>	<i>PU - Public</i>
	<input type="checkbox"/>	<i>PP - Restricted to other programme participants, including the Commission services</i>
	<input type="checkbox"/>	<i>RE - Restricted to a group specified by the consortium, including the Commission services</i>
	<input type="checkbox"/>	<i>CO - Confidential, only for members of the consortium, including the Commission services</i>

Version	Date	Modified by	Comments
1.0.0	19/12/2019	Saskia Loosveldt Tomas, Javier Vegas Regidor, Pierre-Antoine Bretonnière	Draft
2.0.0	29/01/2020	Francisco J. Doblas-Reyes	Final

Table of Contents

List of Tables	5
List of Figures	5
1. Executive summary	6
2. Project objectives	8
3. Detailed report	9

3.1 ESMValTool development	9
Why did we need ESMValTool 2.0?	9
The road to 2.0	9
Finalising 2.0	10
The future	10
3.2 Using ESMValTool	11
3.3 ESMValTool with PRIMAVERA, general CMIP and observational data	11
Performance metrics for essential climate parameters	12
Climate Variability Diagnostics Package CVDP	13
Capacity factor of wind power: Ratio of average estimated power to theoretical maximum power	15
3.4 PRIMAVERA metrics in ESMValTool	17
Tropical cyclones	18
Ocean heat content	20
K-means cluster analysis of sea-ice thickness variability	20
Land - atmosphere and Cloud - atmosphere interactions in summer	21
Eady Growth Rate	22
Blocking 1D / 2D	23
Jet latitude index	26
Weather regimes	28
Sea-ice drift-strength feedback	28
E-vectors	29
Energy budget	30
Fresh water transport	32
Extreme precipitation	32
Sea ice negative feedback	33
4. Lessons learnt	34
4.1 Positive outcomes	34
4.2 Negative outcomes	34
5. Links built	34

List of Tables

Table 1: Metrics submitted by PRIMAVERA partners	18
Table 2: Sample of the fort.66 output containing cyclone information at one timestep for the model CNRM-CM6-1-HR	20

List of Figures

Figure 1: Sample of the output generated by recipe_perfmetrics run with the PRIMAVERA models. Computation of the root mean square deviation of the models for selected atmospheric variables.	13
Figure 2: Sample of the output generated by recipe_cvdv run with the PRIMAVERA models. Computation of the annual trends deviation of the models for variable tas.	14
Figure 3: Sample of the output generated by recipe_cvdv run with the PRIMAVERA models. Computation of the variance as a function of the frequency for the AMO index.	15
Figure 4: Sample of the output generated by recipe_capacity factor run with the PRIMAVERA models. Computation of the capacity factor over Europe for turbines Enercon E70, Gamesa G80, Gamesa G87, Vestas V100 and Vestas V110.	17
Figure 5: Sample of the output generated by recipe_ocean_heat_content run with PRIMAVERA models and CMIP5 data. Computation of the global ocean heat content per area unit and the time series.	20
Figure 6: Sample of the output generated by recipe_modes_of_variability for CMIP5 data	21
Figure 7: Sample of the 1D output generated by recipe_blocking run with PRIMAVERA models and CMIP6 and Observational data. Computation of the frequency of the blocking index in days per month with respect of the longitude for the period 1979-2014.	24
Figure 8: Sample of the 2D output generated by recipe_blocking run with ERA-Interim data. Computation of the frequency of the blocking index with respect to latitude and longitude.	25
Figure 9: Taylor diagrams comparing the performance reproducing 2D blocking index of CMIP6 models against ERA-Interim for the period 1979-2014.	26
Figure 10: Jet latitude distribution (left) and jet speed distribution (right) during winter for the period 1979-2014.	28
Figure 11: Sample of the output generated by recipe_seaice_drift.yml run with CMIP5 models and OSI-450-nh, PIOMAS and IABP observational datasets cmorized for ESMValTool.	29
Figure 12: Sample of the output generated by recipe_energy_vectors run with one year of the ECMWF-IFS-LR model. Zonal and meridional components of the energy vector.	30
Figure 13: Sample of the output generated by recipe_energy_budget for the PRIMAVERA models. Computation of the global energy flows over a Trenberth diagram.	32

Figure 14: Sample of the output generated by recipe_transports for HadGEM3-GC33-LL CMIP6 model. First plot shows the yearly total time-series, while the second shows the same just for April. 33

Figure 15: Ice formation efficiency (IFE) applied to the CMIP5 ensemble (Massonnet et al., 2018). 34

1. Executive summary

The work performed for this deliverable focused on the development of the Earth System Model eValuation Tool (ESMValTool) version 2.0. New capabilities have been created with the complete rewrite of the tool, with the most relevant being the ability to perform in an efficient way the most common preprocessing tasks by the tool itself and not in diagnostic code. The capabilities added in this new version have been used to evaluate CMIP6 models and integrate the PRIMAVERA metrics into the same framework.

The improvements of version 2.0 of ESMValTool do not focus specifically on PRIMAVERA, but more generally on the development of a standard tool for evaluating model and observational data. Therefore, there have been a lot of efforts spent on features that go beyond the scope of specific PRIMAVERA requirements. Still, in the long run the general developments have been essential for the special needs that the evaluation of PRIMAVERA data require and the adaptation of project-specific diagnostics and metrics into the tool.

The main technical developments of the deliverable are presented in section 3. In section 3.1 of this deliverable, details of the general developments and improvements from v1.0 to v2.0 are given. These developments have allowed the submission of the manuscript ESMValTool

v2.0 - Technical overview (*Righi et al., 2020*). In section 3.2 a brief user-documentation and guidelines for running the tool in the JASMIN platform are explained. In section 3.3 three recipes from ESMValTool related to other projects have been run using the data available from the PRIMAVERA models. Finally, section 3.4 gives an overview of the PRIMAVERA diagnostics and metrics that have been adapted to the common framework.

It needs to be noted that at the time of submission of this deliverable, due to time constraints, the PRIMAVERA metrics are not officially merged into the tool because they are still under review.

To conclude, the lessons learned and links built, along with a brief description of the use of the tool to provide PRIMAVERA material to the next IPCC report, will be listed in section 4 and 5 respectively.

2. Project objectives

With this deliverable, the project has contributed to the achievement of the following objectives (DOA, Part B Section 1.1) WP numbers are in brackets:

No.	Objective	Yes	No
A	To develop a new generation of global high-resolution climate models. (3, 4, 6)		
B	To develop new strategies and tools for evaluating global high-resolution climate models at a process level, and for quantifying the uncertainties in the predictions of regional climate. (1, 2, 5, 9, 10)		
C	To provide new high-resolution protocols and flagship simulations for the World Climate Research Programme (WCRP)'s Coupled Model Intercomparison Project (CMIP6) project, to inform the Intergovernmental Panel on Climate Change (IPCC) assessments and in support of emerging Climate Services. (4, 6, 9)		
D	To explore the scientific and technological frontiers of capability in global climate modelling to provide guidance for the development of future generations of prediction systems, global climate and Earth System models (informing post-CMIP6 and beyond). (3, 4)		
E	To advance understanding of past and future, natural and anthropogenic, drivers of variability and changes in European climate, including high impact events, by exploiting new capabilities in high-resolution global climate modelling. (1, 2, 5)		
F	To produce new, more robust and trustworthy projections of European climate for the next few decades based on improved global models and advances in process understanding. (2, 3, 5, 6, 10)		
G	To engage with targeted end-user groups in key European economic sectors to strengthen their competitiveness, growth, resilience and ability by exploiting new scientific progress. (10, 11)		
H	To establish cooperation between science and policy actions at European and international level, to support the development of effective climate change policies, optimize public decision making and increase capability to manage climate risks. (5, 8, 10)		

3. Detailed report

3.1 ESMValTool development

The PRIMAVERA project has, from the very beginning, the determination to share the metrics developed within it with the whole Earth Sciences community. D1.4 “Report on strategy for integrating metrics software” and WP1 MS2 “Strategy for integrating the metrics software available in the different partner institutions” defined the strategy to follow, setting the focus in integrating PRIMAVERA metrics into ESMValTool. Even more, the decision was also to collaborate in the main development efforts to ensure that the tool will be able to manage high resolution outputs.

Following this strategy, PRIMAVERA has been part of the core development team that has brought the tool to new levels with the development of version 2.0

Why did we need ESMValTool 2.0?

At approximately the same time PRIMAVERA chose to invest in ESMValTool, version 1.0 was released. ESMValTool 1.0 was a great proof of concept, but had at this time some performance, software engineering and usability issues (really high memory requirements, lack of a proper installation mechanism, execution times...) coming from the fact that each piece had been developed separately by users focusing more on the scientific results than general performance. ESMValTool 1.0 was therefore not meeting the requirements needed in the multi user and high resolution environment that PRIMAVERA represents.

ESMValTool’s original development team from DLR was well aware of this and decided to approach the incipient community with the suggestion of creating a new version from scratch. The aim was to fix all those flaws and to set a new base from which further developments would be easier to do. It would bring ESMValTool from a collection of NCL scripts to an object oriented and properly engineered Python tool, that will be the foundation of a great tool for the community in the years to come. The community eagerly accepted the suggestion and development of version 2.0 started shortly after.

PRIMAVERA was not the only project committed to this effort: H2020 CRESCENDO, C3S-MAGIC, CMIP6-DICAD and others also joined to get the tool to a level that none of those projects would be able to reach alone.

The road to 2.0

The development of ESMValTool 2.0 was a big task, as the decision of switching from NCL to Python + Iris as the core technologies for the tool prevented reutilization of nearly all the previous code. The time investment was expected to be considerable but, as was found out later, it was underestimated. One of the keys for this is one problem that most scientists and scientific engineers face: the inability for the core team to fully focus on the task at hand. As most of the collaborators were funded by different scientific projects the work was divided between the long term goal of developing a robust and powerful tool and complying with the obligations that developers had with the funding projects. This led to extended developments on version 1, work spent integrating diagnostics in version 2 before it was mature enough and, in general, lots of duplicated efforts.

The extended development period also comes from the evolution of the tool and the increasing ambition and knowledge of the development team. What in the original plan of version 2 was just the option to preprocess the data and apply the most common preprocessing functionalities has become a highly extensible, highly configurable and easily maintainable preprocessor module that greatly simplifies the development of diagnostics. Even more, it now can be called directly from python diagnostics in case some functionalities are needed after some specific calculations are done, i.e. to regrid or to compute anomalies on diagnostic results.

Finally, in the meantime Iris, the core library used for the development of the tool, liberated version 2. This new version changed the underlying technology to Dask, allowing ESMValTool to do lazy computations, greatly reducing its memory footprint in most cases. An extra effort has been done to ensure that all functionalities were benefiting from this new paradigm.

Finalising 2.0

After more time and effort than initially expected, version 2 core functionality is now complete and ready to be used. The final stable release of v2.0 will be done when the manuscript is published.

However, the release of a stable version does not imply that the development of the tool will be stopped. ESMValTool is being enhanced continuously on it's GitHub repository in order to fit the user's requests.

The future

From the start of the project, PRIMAVERA partners wanted to ensure that all metrics produced within it would be used and maintained well after the project itself has finished. With the integration into ESMValTool, this objective has been achieved: maintenance of the metrics is guaranteed in the long term, as it is part of the development team's responsibilities and funded through the H2020 IS-ENES3 project.

Apart from maintenance of the tool, another objective of this work is to allow the evolution and improvement of the metrics. . The continuous evolution of the preprocessor will allow users to improve the way in which the data is prepared for the run, potentially improving the results without modifying the metric itself. Direct improvements of the metrics are also to be expected, coming both from the original authors and the users, to keep them on par with the state of the art. H2020 IS-ENES3 project also provides funding for the development team to help getting those developments available to the public.

Key Points for section 3.1

- PRIMAVERA has been heavily involved in the core developments and improvements of ESMValTool, making sure that high resolution data from the project could be handled.
- While ESMValTool v2.0 is stable and ready to use, the tool's capabilities can still be extended to fit the needs of the diagnostics developers and users.

3.2 Using ESMValTool

A common installation of the tool has been set up in order to run the recipes on Jasmin. As, at the moment, the stable release is old and PRIMAVERA metrics have not reach the development branch, we provide three different versions of the tool. Those versions are provided in the form of three conda environments in `/gws/smf/j04/primavera/envs`:

- `esmvaltool`: Latest official release. Updated manually.
- `esmval_dev`: Current development branches. Updates on Mondays.
- `esmval_primavera`: GitHub branch `PRIMAVERA_all`¹. This is a development branch that contains all the PRIMAVERA scripts that have already been implemented (see section 3.4). Updated manually

In order to use any of the installed versions, the user will need to activate the appropriate environment, using their own conda installation or with the following command:

```
source /home/users/jvegas/miniconda3/bin/activate  
/gws/smf/j04/primavera/envs/${ENVIRONMENT_NAME}
```

The user configuration file can be found in their own folder in

```
/gws/smf/j04/primavera/conf-files
```

Users are encouraged to create a copy of this file, as they may want to tweak the values in it and tune it for one of the 3 installations.

Specific instructions about how to use ESMValTool's preprocessor and recipes, how to modify recipes and how to develop custom diagnostic scripts are out of the scope of this document, but can be found in the PRIMAVERA wiki (<http://proj.badc.rl.ac.uk/primavera-private/wiki/WP1/ESMValToolRunonJasmin>). More information is available in the tutorial that was prepared for the 3rd PRIMAVERA general assembly, and in ESMValTool documentation (<https://esmvaltool.readthedocs.io/en/latest/>).

Key points for section 3.2

- A common installation of the tool has been prepared to run the PRIMAVERA recipes and already existing recipes in the tool.
- Information about how to use it is available in PRIMAVERA's wiki

3.3 ESMValTool with PRIMAVERA, general CMIP and observational data

The main goal of ESMValTool is not only to provide a benchmarking and evaluation interface, but also to ensure that the suite of diagnostics can be performed easily when new model data becomes available at the Earth System Grid Federation nodes.

To illustrate this capability, three recipes that are already part of the tool were modified to be run using experiments from HighResMIP.

¹ https://github.com/ESMValGroup/ESMValTool/tree/PRIMAVERA_all

It needs to be noted that the main difficulty that may appear when running ESMValTool recipes with new data is the compliance with CMOR standards, as ESMValTool will stop the execution if the data does not adjust with the standards to prevent errors in the results. In the case of PRIMAVERA data that has been used, most of the models were CMOR compliant and passed the checks without any issues. Only data from two institutes presented problems when going through the preprocessing:

- MPI-M models have several metadata issues regarding the coordinates not having the proper standard name. The detected errors have been corrected by adding the proper fixes to ESMValTool.
- There was also a more complex metadata problem with EC-Earth-Consortium data. At the time of preparing this deliverable, after using ESMValTool preprocessors, the output of EC-Earth consisted of empty NetCDF files. Work is currently ongoing to fix the issue.

Performance metrics for essential climate parameters (Gleckler et al. (2008), Righi et al. (2015))

The recipe was run for the highresSST-present experiment against ERA-Interim, NCEP, CERES-EBAF, ESACCI-SST and GPCP-SG observational datasets. The variables that were analysed are the following:

- Monthly Air Temperature (*ta*) from 1980 to 2014. Several preprocessing operations are applied to *ta* in order to extract pressure levels, regrid to the observations, mask the fill values and compute the multi-model mean and medians for the non-observational datasets.
- Monthly Eastward wind (*ua*) from 1980 to 2014, using the same preprocessing operations as in *ta*.
- Monthly Northward wind (*va*) from 1980 to 2014, using the same preprocessing operations as in *ta*.
- Monthly 3D geopotential height (*zg*) from 1980 to 2014, using the same preprocessing operations as in *ta*.
- Monthly Near-Surface Air Temperature (*tas*) from 1980 to 2014, using the same preprocessing operations as in *ta*, but excluding the level extraction.
- Monthly Surface Temperature (*ts*) from 1992 to 2010².
- Monthly Precipitation (*pr*) from 1990 to 2013, using the same preprocessing operations as in *tas*.
- Monthly All-sky longwave radiation (*rlut*) from 2000 to 2013, using the same preprocessing operations as in *tas*.
- Monthly All-sky shortwave radiation (*rsut*) from 2000 to 2013, using the same preprocessing operations as in *tas*.
- Monthly Longwave Cloud Radiative Effect (*lwcre*) from 2000 to 2013. Note that this is not a model direct output variable but an ESMValTool derived variable, computed automatically given the availability of variables *rlut* and *rlutcs*.
- Monthly Shortwave Cloud Radiative Effect from 2000 to 2013. This is also an ESMValTool derived variable computed automatically from variables *rsut* and *rsutcs*.

² The reduction in the time period was due to the non-availability of the data for some years.

The diagnostic script generates a series of plots per model showcasing the global variable values per model, as well as computing multi-model statistics and biases against reference observational datasets. A sample of the plots is illustrated in Figure 1.

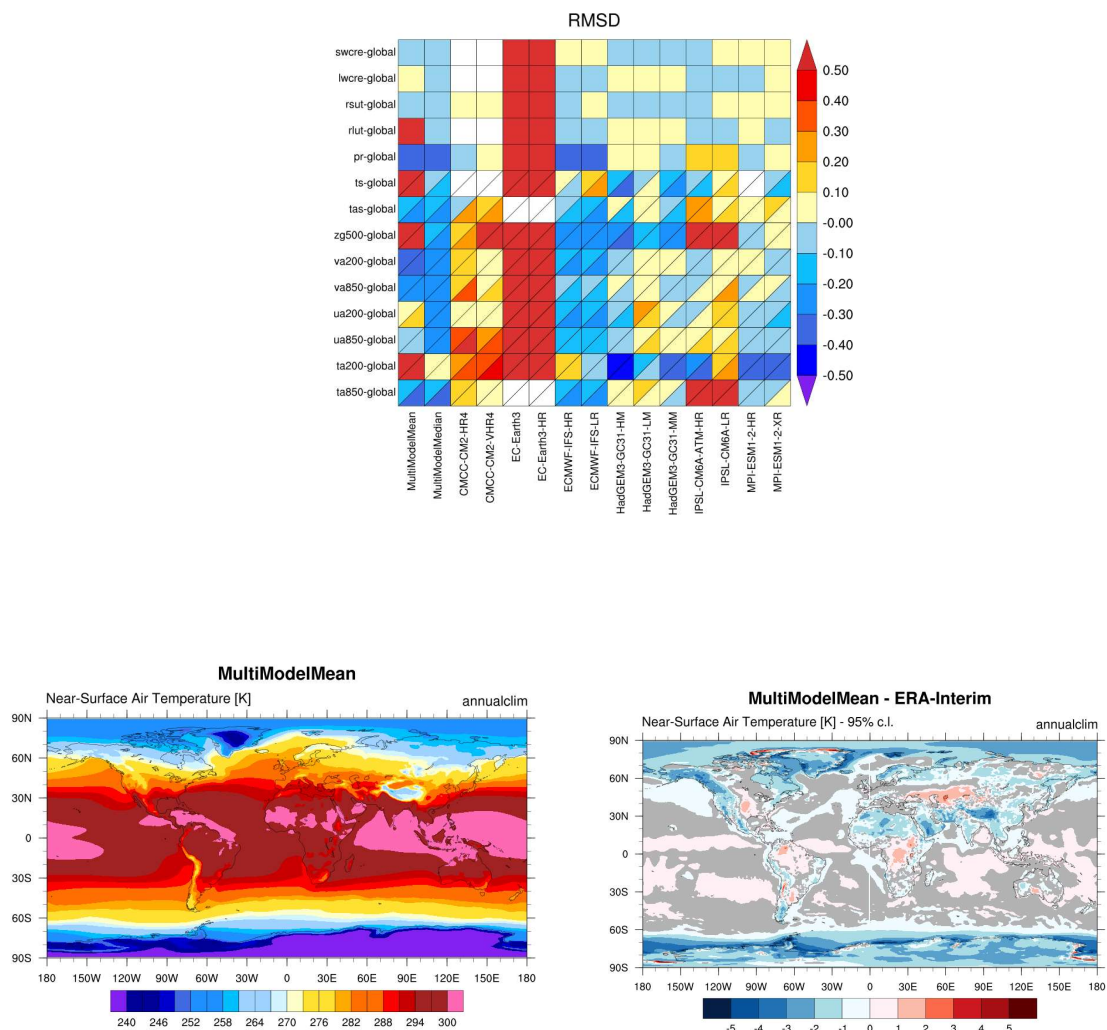


Figure 1: Sample of the output generated by recipe_perfmetrics run with the PRIMAVERA models. Computation of the root mean square deviation of the models for selected atmospheric variables and multimodel means for variable *tas*.

Climate Variability Diagnostics Package | CVDP (Phillips, A. S., C. Deser, and J. Fasullo, 2014)

ESMValTool can also be used to create namelists to run external packages, such as the NCAR CVDPackage. The recipe was run for the hist-1950 experiment from 1950 to 2014. The variables that were analysed are the following:

- Monthly Surface Temperature (*ts*)
- Monthly Near-Surface Air Temperature (*tas*)
- Monthly Precipitation (*pr*)
- Monthly Sea Level Pressure (*psl*)

ESMValTool generates the namelist for each dataset requested in the recipe, thus acting as a wrapper for the CVDP package. In Figure 2 and Figure 3, the plots produced by the ESMValTool integration of CVDP for the annual *tas* trends and monthly AMO are shown for the PRIMAVERA models.

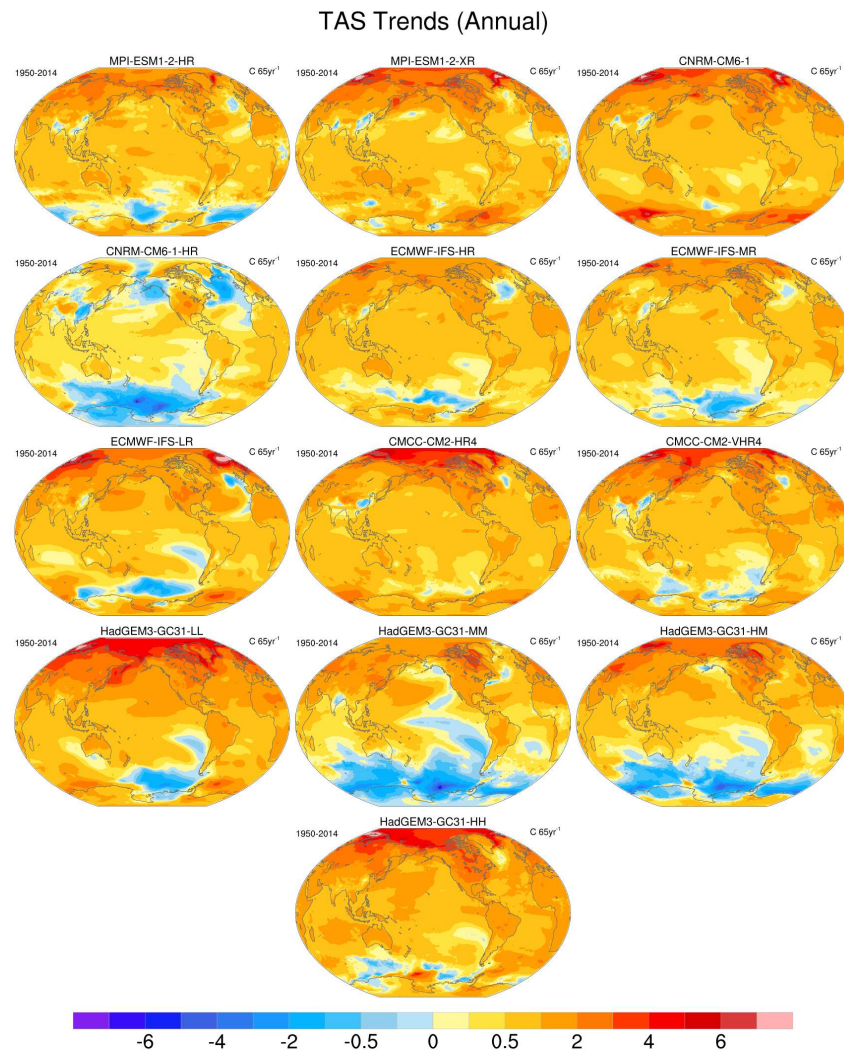


Figure 2: Sample of the output generated by recipe_cvdv run with the PRIMAVERA models. Computation of the annual trends deviation of the models for variable *tas*.

AMO (Monthly)

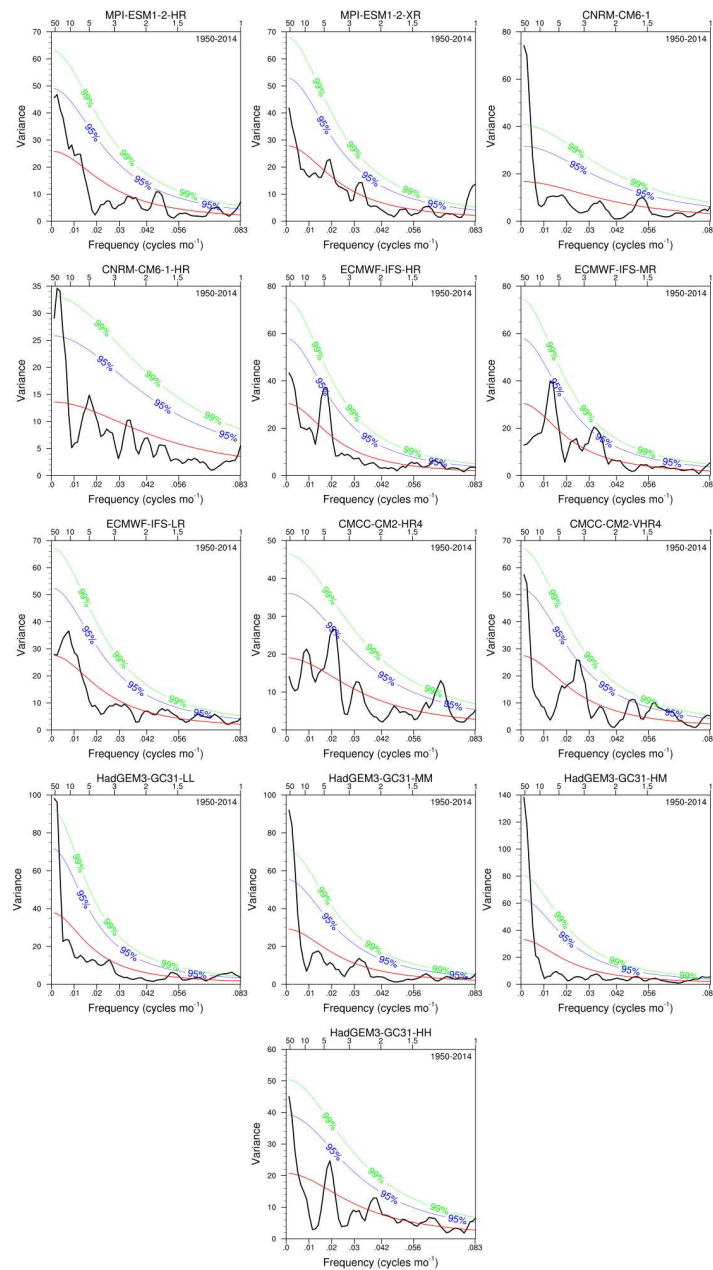
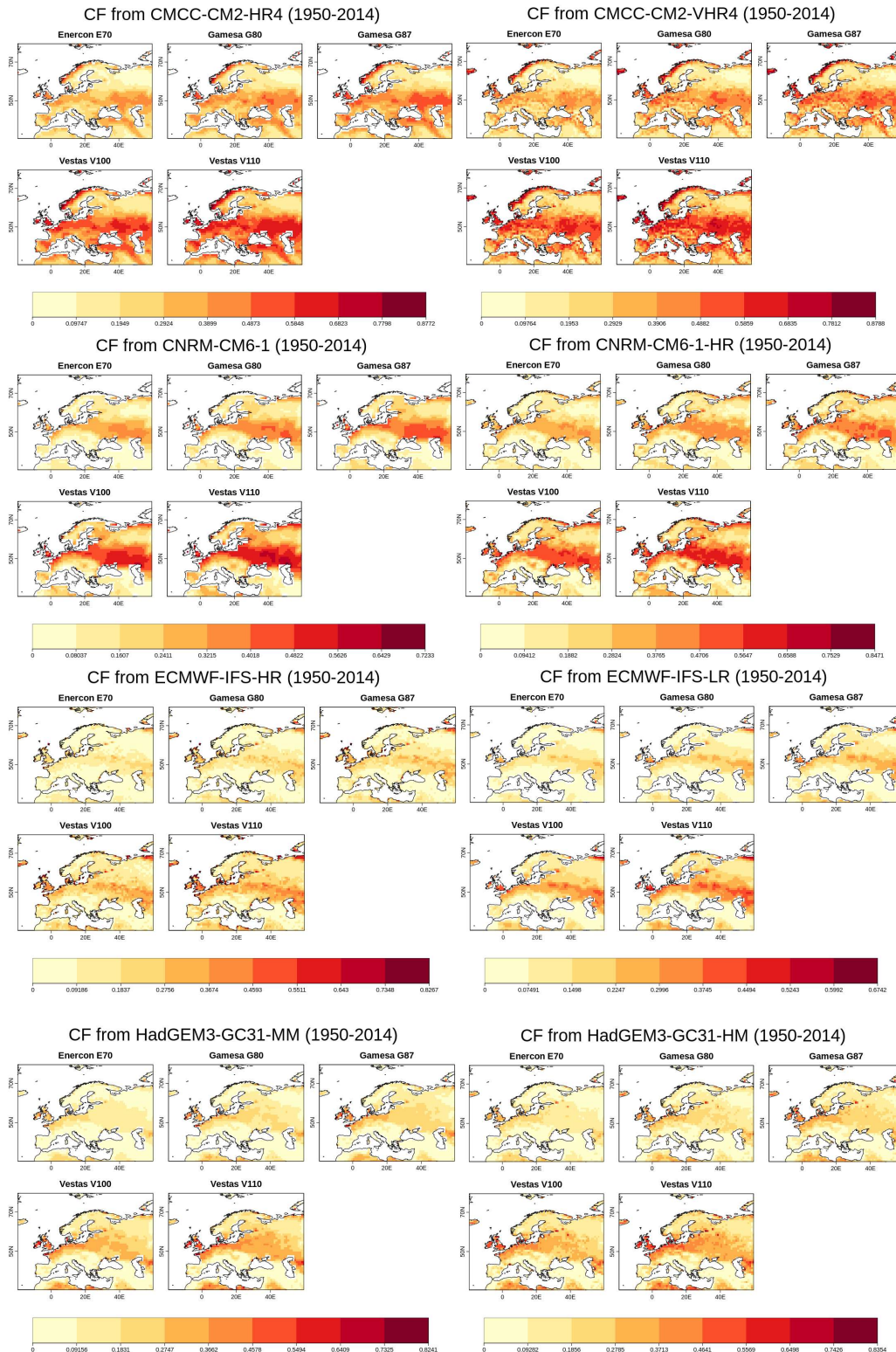


Figure 3: Sample of the output generated by recipe_cvdv run with the PRIMAVERA models. Computation of the variance as a function of the frequency (cycles mo^{-1}) for the AMO index.

Capacity factor of wind power: Ratio of average estimated power to theoretical maximum power (Lledó, L. (2017))

This recipe was chosen to illustrate that ESMValTool can also be used to develop diagnostics that are oriented for Earth System Services. The capacity factor is a normalized indicator of the suitability of wind speed conditions to produce electricity, irrespective of the size and number of installed turbines. This indicator is provided for three different classes of wind turbines (IEC, 2005) that are designed specifically for low, medium and high wind speed conditions.

The recipe was computed using the Daily Near-Surface Wind Speed (*sfcWind*) for the hist-1950 experiment from 1950 to 2014. The results of the capacity factor for a selected variety of wind turbines are shown in Figure 4. The original diagnostic was improved in order to be run with multiple models at the same time, as the original version only accepted one model per run. This also goes to show that while some features may be already available in ESMValTool, the whole community is welcome to modify and improve the scripts originally developed by someone else.



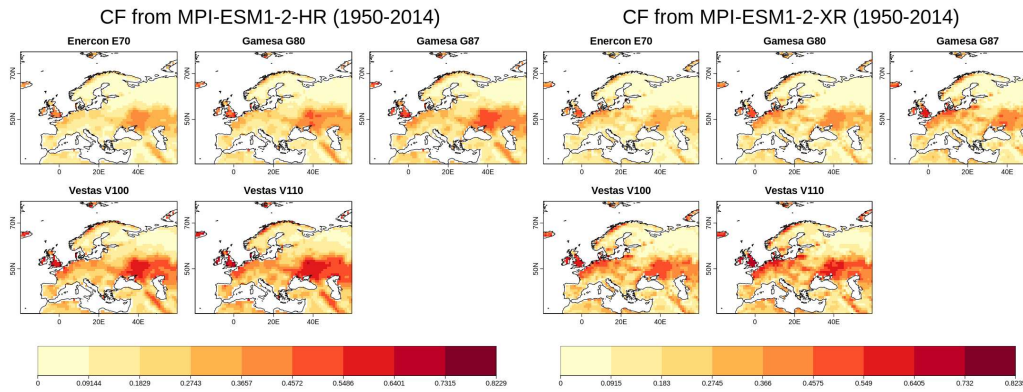


Figure 4: Sample of the output generated by recipe_capacity factor run with the PRIMAVERA models. Computation of the capacity factor over Europe for turbines Enercon E70, Gamesa G80, Gamesa G87, Vestas V100 and Vestas V110.

Key points for section 3.3

- ESMValTool recipes and scripts are able to deal with high resolution data as long as it follows the expected standards.
- Already existing recipes can easily be modified to be run with new datasets.
- Everyone is welcome to improve already existing scripts, as long as the community developing guidelines are followed.

3.4 PRIMAVERA metrics in ESMValTool

The implementation of the metrics proceeded following the list shown in Table 1, available from the PRIMAVERA wiki. It needs to be noted that the ENSO composites metric finally was not submitted. Additionally, three metrics that do not appear on the table were also implemented: the fresh-water transports metric, the extreme precipitation metric and the sea-ice negative feedback metric.

Table 1: Metrics submitted by the PRIMAVERA partners.

REALM	METRIC	INSTITUTE	DEVELOPER
Atmosphere	Tropical cyclones	BSC	Louis-Philippe Caron
Atmosphere	ENSO composites	BSC	Etienne Tourigny
Ocean	Calculation of ocean heat content, options for different layers and basins	BSC	Eleftheria Exarchou
Sea ice	K-means cluster analysis of sea ice thickness variability	BSC	Neven Fuckar
Atmosphere / Land	Land / atmosphere interactions in summer	CERFACS	Julien Boé
Atmosphere	Cloud / temperature interactions	CERFACS	Julien Boé
Atmosphere	Eady Growth Rate	CERFACS	Emilia Sanchez-Gomez
Atmosphere	Blocking 1D	CMCC, BSC	Panos Athanasiadis, Javier Vegas-Regidor (ESMVal Integration)
Atmosphere	Jet Latitude Index	CMCC	Panos Athanasiadis
Atmosphere	Frequency of occurrence, pattern correlation and significance of cluster partition for weather regimes	CNR	Irene Mavilia
Sea ice	Sea ice drift-strength feedback	UCL	David Docquier
Atmosphere	E-vectors	UREAD	Marie-Estelle Demory
Atmosphere	Energy budget and hydrological cycle	UREAD	Benoit Vanniere and Marie-Estelle Demory
Atmosphere	Blocking 2D	UREAD, BSC	Reinhard Schiemann, Javier Vegas-Regidor (ESMVal Integration)

Tropical cyclones

The original scripts for the Tropical Cyclones were submitted by Louis-Philippe Caron and Jean-Philippe Baudouin (BSC). The goal of the metric is to prepare the input and call the BSC

modified version of the GFDL cyclone tracker. Before attempting to run the recipe, the GFDL tracker needs to be compiled separately and the path to the executable needs to be specified in the recipe. The recipe calls the following variables:

- Daily or 6-hourly Sea Level Pressure (*psl*)
- Daily or 6-hourly Eastward wind (*ua*)
- Daily or 6-hourly Northward wind (*va*)
- Daily or 6-hourly Eastward Near-Surface Wind (*uas*)
- Daily or 6-hourly Northward Near-Surface Wind (*vas*)
- Daily or 6-hourly Near-Surface Air Temperature (*ta*)
- Daily or 6-hourly Geopotential Height (*zg*)

Four preprocessors have been defined to be called by the variables:

- *ppregrid*: Re grids *psl* to a 1°x1° grid.
- *ppwind*: Interpolates and extracts the required vertical levels and re grids the wind variables to a 1°x1° grid.
- *ppta*: Interpolates and extracts the required vertical levels and re grids *ta* to a 1°x1° grid
- *ppzg*: Interpolates and extracts the required vertical levels and re grids *zg* to a 1°x1° grid.

The recipe also allows the user to set several parameters needed in order to generate the namelist for the tracker.

The diagnostic script then reorganises the variables, performs a vertical mean for the temperature in order to discriminate warm cores, and stores them into a single file per model and month, as required by the GFDL tracker. The namelist and initialization files with information regarding the time steps also get generated. Finally, the tracker is executed and the following output files get generated per model and per month:

- fort.64: Output in ATCF format.
- fort.66: Contains all the cyclone information. All of the monthly fort.66 files get merged into a single file per model. This merged file can be found in the work directory of every execution.
- fort.67: Summary of the cyclones being tracked at the penultimate time step.
- fort.68: Not documented
- fort.69: Output in modified ATCF format
- fort.72: Not documented
- fort.73: Not documented
- fort.76: Not documented

Table 2: Sample of the fort.66 output containing cyclone information at one timestep for the model CNRM-CM6-1-HR.

TG, 0001, 2013010100_F000_525N_0348W_FOF, 2013010100, 03, test, 000, 525N, 348W, 38, 956, XX, 34, NEQ, 0350, 0344, 0350, 0249, 959, 109, 136, -999, -9999, -9999, N, 163, 54, 204, 219, 178, 190
--

Since the fort.66 file contains the cyclone information needed for (future) post-processing operations, as shown in Table 2, all the fort.66 files per month get merged into a single file per model.

Future developments

While the GFDL tracker is able to work with 6-hourly data, there is currently an unresolved ESMValTool issue that prevents the loading of some 6-hourly data variables. A branch with the fix for this issue is pending to be merged, therefore users interested in using 6-hourly variables could run the recipe with that branch.

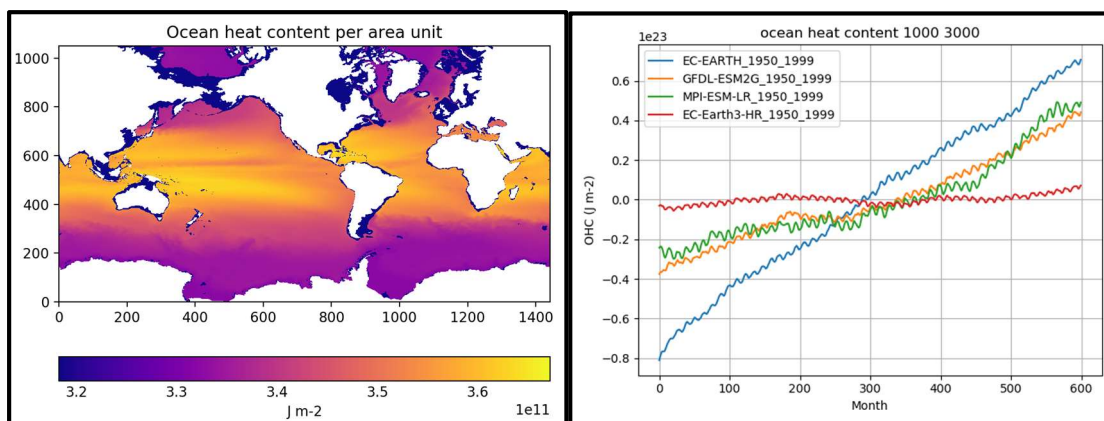
a) Ocean heat content

The original scripts were submitted by Eleftheria Exarchou and Javier Vegas-Regidor (BSC). The metric aims at computing the global ocean heat content and the climatology, as well as the time series between user-given depth layers. The recipe calls the following variables:

- Monthly Sea Water Potential Temperature (*thetao*)
- Grid-Cell Area for Ocean Variables (*areacello*)
- Ocean Model Cell Thickness (*thkcello*)

It also allows the user to decide which plots to generate by setting true or false values accordingly.

The diagnostic script then computes the depth weights and integrates *thetao* over the desired layers and proceeds to compute climatologies and time-series if requested in the recipe. The output that gets generated consists of netcdf files per model. The 2D plots for the global heat content and its climatology, and multi-model comparison plots for the time-series are illustrated in Figure 5. The script can be called several times in order to compute several layers within the same execution.



Figure

5: Sample of the output generated by recipe_ocean_heat_content run with PRIMAVERA models and CMIP5 data. Computation of the global ocean heat content per area unit and the time series.

b) K-means cluster analysis of sea-ice thickness variability

The original scripts were submitted by Neven Fučkar (BSC). The k-means cluster analysis algorithm was already implemented in ESMValTool by BSC for the C3S MAGIC project. The recipe and the diagnostics were modified in order to work for the sea-ice thickness variable. The goal of the metric is to compute modes of variability from a reference or observational

dataset and from a set of climate projections and calculate the root-mean-square error between the mean anomalies obtained for the clusters from the reference and projection data sets. The recipe calls the following variables:

- Monthly Sea ice thickness (*sit*)

The variable undergoes the following preprocessing:

- Extract a rectangular region between $[0^\circ, 360^\circ]$ for the longitudes and $[50^\circ, 90^\circ]$ and regrid to a regular 1×1 grid.

The user can specify the number of clusters to be computed, the detrend order, the cluster method, whether to apply the clustering algorithm to the spatial data or to the EOFs, and the frequency.

The script generates three NetCDF files containing the observed and projected weather regimes and the RMSE between them. It also generates plots, displayed in Figure 6, showing the frequency of occurrence of each variability mode. It needs to be noted that, at the time of the submission of this deliverable, the diagnostic script consists of an old version of the code. The new version is pending to be integrated into the tool.

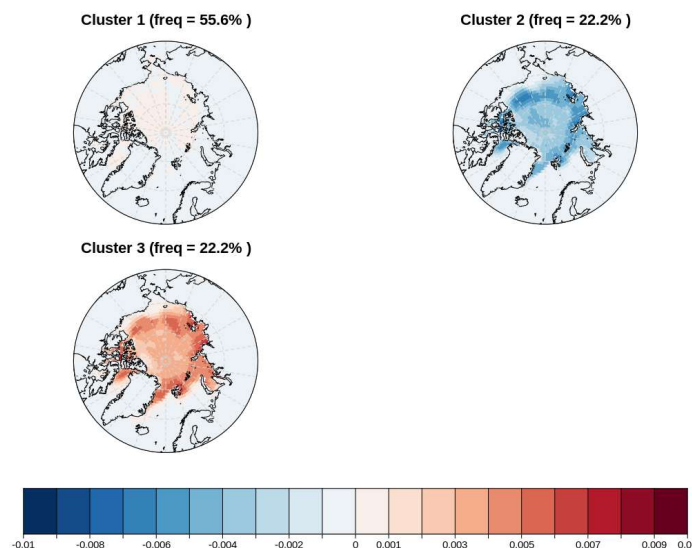


Figure 6: Sample of the output generated by `recipe_modes_of_variability` for CMIP5 data.

c) Land - atmosphere and Cloud - atmosphere interactions in summer

The original scripts were submitted by Julien Boé (CERFACS). The goal of the metric is to compute evaporative fraction statistics and correlation metrics between land and atmosphere variables. The recipe calls the following variables:

- Monthly Surface Upward Sensible Heat Flux (*hfss*).
- Monthly Surface Upward Latent Heat Flux (*hfls*).
- Monthly Surface Downwelling Shortwave Radiation (*rsds*).
- Monthly Total Cloud Cover Percentage (*clt*).
- Monthly Near-Surface Air Temperature (*tas*).
- Monthly Total Soil Moisture Content (*mrso*)
- Percentage of the grid cell occupied by land (including lakes) (*sftlf*)

Four preprocessors have been defined to be applied to some variables:

- ppsummer: Extracts the summer season months (JJA) and performs the summer mean for variables hfls, hfss, rlds, rlds, clt and tas.
- ppmrso: Performs the seasonal means for variable mrso.
- ppsummerdetrend: Extracts the summer season months (JJA), performs the summer mean and detrends variables hfls, hfss, rlds, clt and tas.
- ppmrsodetrend: Performs the seasonal means and detrends variable mrso.

The diagnostic script performs the required computations to obtain:

- The climatological evaporative fraction
- The interannual standard deviation of the evaporative fraction
- The interannual correlation between the latent heat flux and the total radiation at surface
- The interannual correlation between temperature and cloud cover.
- The interannual correlation between soil moisture in spring and soil moisture in summer

The script generates four NetCDF files per model: one file for the evaporative fraction statistics, one file for the correlation metrics, and the same files but with the detrended data. Therefore, it needs to be noted that for this version of the metric each ensemble member is treated separately. The original scripts did not include plotting routines.

A particularity of this metric is the use of masking and regridding at the final steps by using custom generated masks and grids. Instead, ESMValTool expects the use of model output fx variables or Natural Earth masks. Said variables were not available at ESGF for the PRIMAVERA models and experiments. Therefore, although the recipe and diagnostic scripts are finished, they cannot be ran with PRIMAVERA data at the moment.

d) Eady Growth Rate

The original scripts for the Eady Growth Rate metric were submitted by Emilia Sanchez-Gomez (CERFACS). The goal of this metric is to compute the mean annual, winter and summer Eady Growth Rates. The recipe calls the following variables:

- Monthly Near-Surface Air Temperature (*ta*)
- Monthly Geopotential Height (*zg*)
- Monthly Eastward wind (*ua*)

Three preprocessors have been defined to be called by all the variables:

- annual_mean: Computes the annual mean.
- summer_mean: Extracts the summer season months (JJA) and computes the seasonal mean.
- winter_mean: Extracts the winter season months (DJF) and computes the seasonal mean.

The diagnostic script performs the required computations to obtain:

- The Brunt-Väisälä frequency

- The Coriolis force
- The Eady Growth Rate

The Eady Growth Rate values get saved in a NetCDF file per model. The diagnostic script is called three times in the recipe in order to obtain the annual, summer and winter means.

e) Blocking 1D / 2D

The blocking metric is one of the best examples of the advantages of the collaborative effort done in PRIMAVERA. Two different versions were contributed by Panos Athanasiadis (1D metric, CMCC) and Reinhard Schiemann (2D metric, URead). Both are using the same underlying algorithm, yet with some minor differences that are discussed in the respective documentation.

The recipe calls the following variable:

- Daily geopotential height (zg).

The variable calls the following preprocessor:

- extract: Selects the 500 hPa pressure level and extracts a rectangular region between [0°, 360°] for the longitudes and [20°, 90°] for the latitudes.

The recipe performs two metrics: one produces a zonal profile while the other gives a 2D map of blocking frequency. They also use different meridional distances (20° 1D, 15° 2D) to compute the geopotential height gradients to the north and the south of the grid point in consideration. Finally, each algorithm applies some different additional processing:

- The 1D index is applying a smoothing in longitude to account for the large-scale nature of the phenomenon. The results of this operation are showcased in Figure 7 for observational, CMIP6 and PRIMAVERA models.
- The 2D index is applying a persistence criteria to only count appearances that last more than 5 days. The scripts produce both the value of the blocking index per model and the Taylor diagrams against observational sets, as seen in Figure 8 and Figure 9 respectively.

The added recipe will compute both as they were submitted, but the convergence will allow users to customize them to mix and match configurations for research or educational purposes; i.e. computing the 1D index adding the persistence criteria or computing both at the same time with the same configuration.

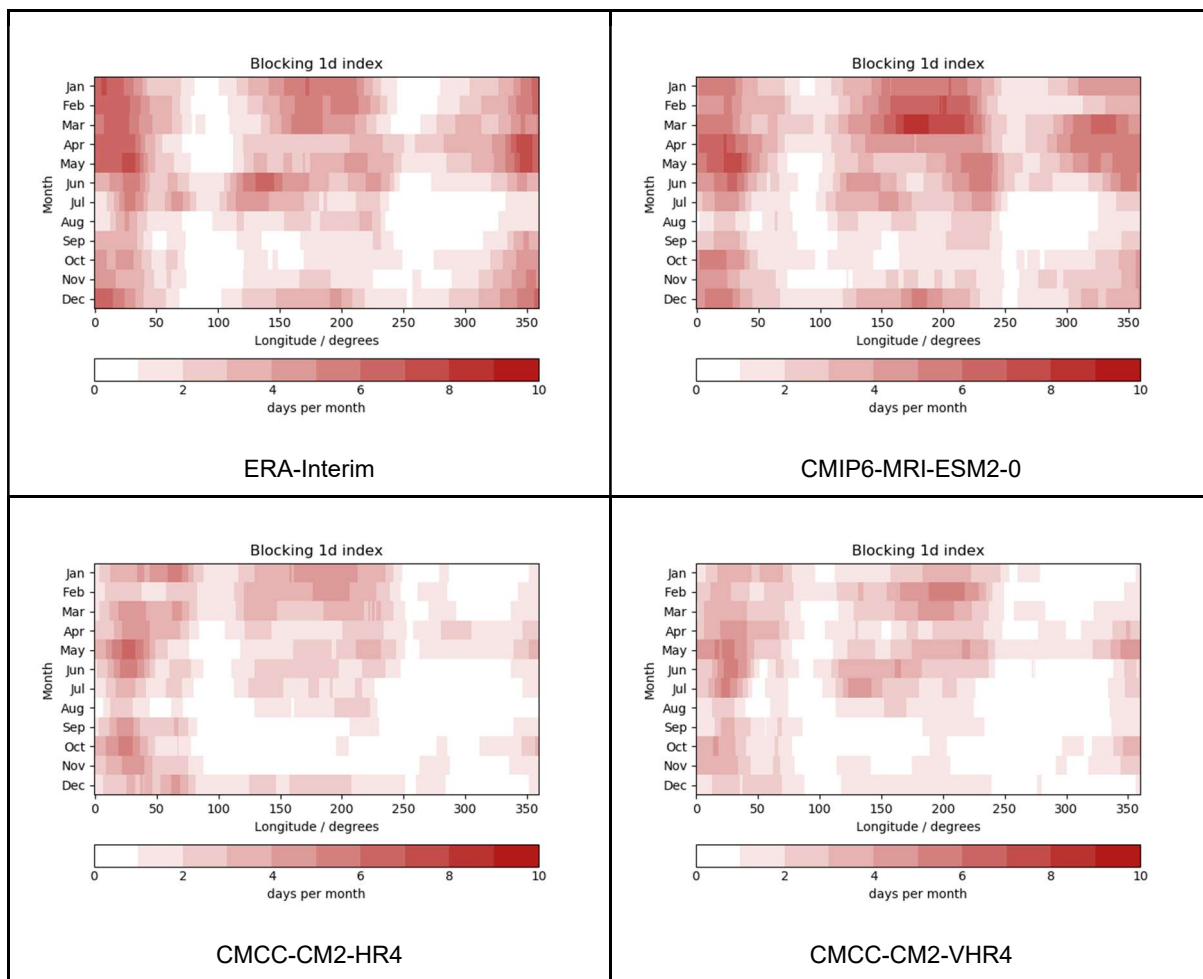


Figure 7: Sample of the 1D output generated by recipe_blocking run with PRIMAVERA models and CMIP6 and Observational data. Computation of the frequency of the blocking index in days per month with respect of the longitude for the period 1979-2014.

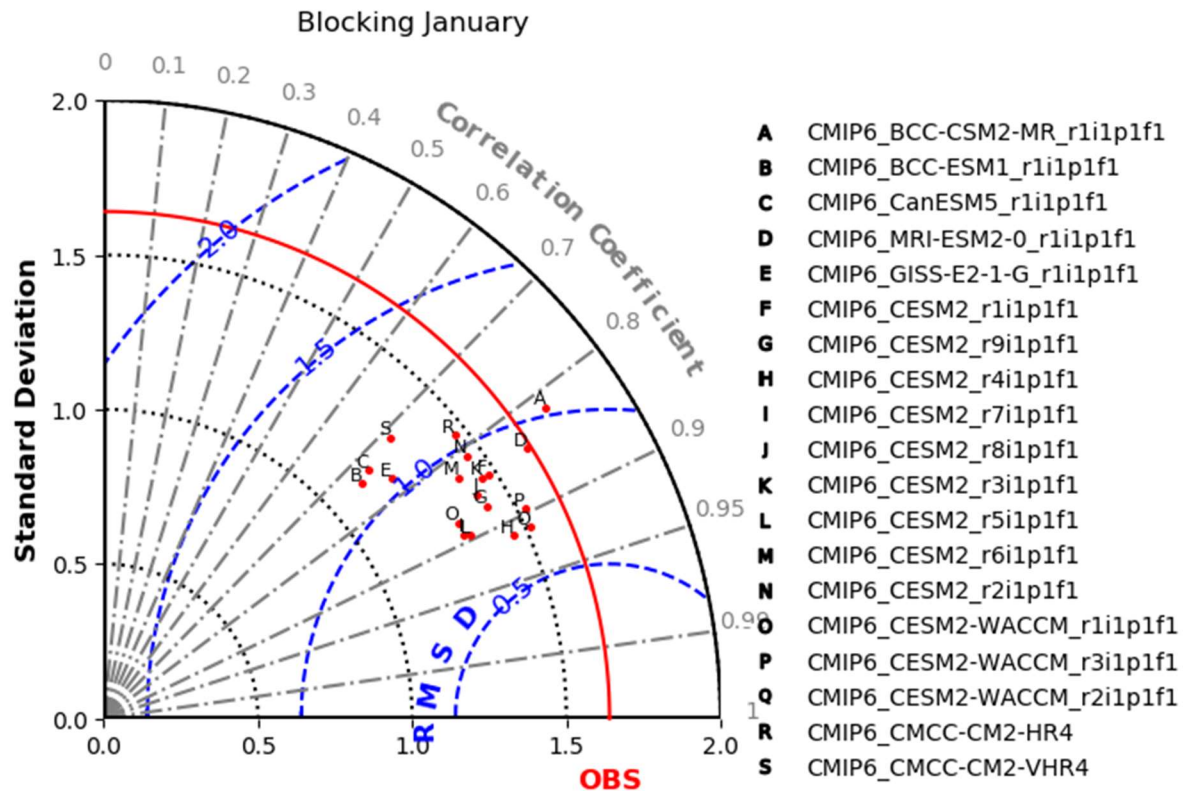


Figure 9: Taylor diagrams comparing the performance reproducing 2D blocking index of CMIP6 models against ERA-Interim for the period 1979-2014. The first one corresponds to the whole year, while the second one corresponds only to January

Future developments

Currently, the 1D index is computed for the same latitude at every longitude, as a simplification. In the future it will be computed at varying latitude for each longitude, following the known climatology or the maximum of the 2D index.

f) Jet latitude index

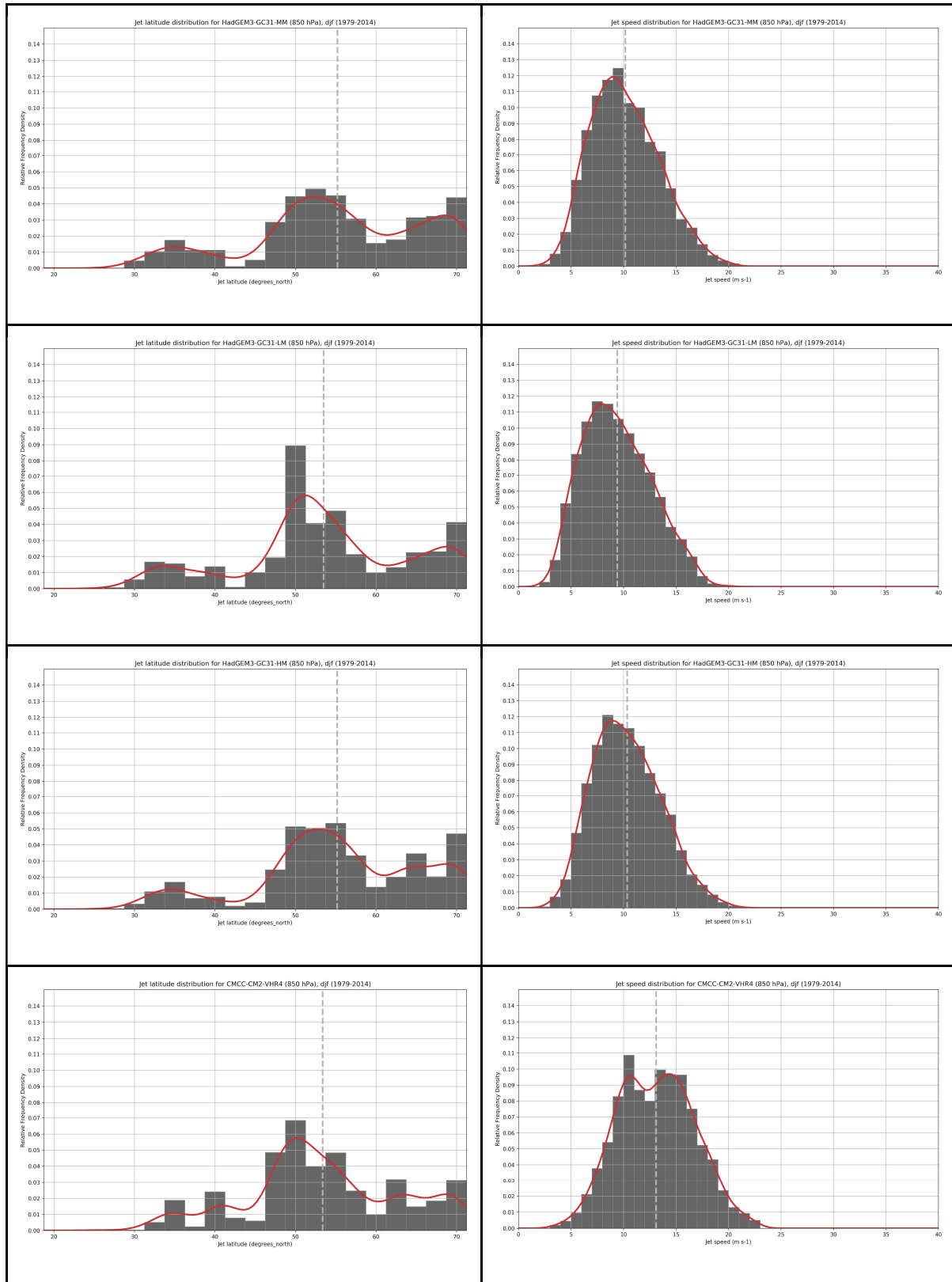
The original scripts were contributed by Panos Athanasiadis (CMCC). The goal of the recipe is to characterize the jet stream behaviour. The recipe calls the following variable:

- Daily Eastward wind (ua).

The following preprocessor gets defined in order to be applied over the variable:

- `get_region`: Extracts the 850 hPa pressure level and extracts a rectangular region between $[0^\circ, 60^\circ]$ for the longitudes and $[20^\circ, 75^\circ]$ for the latitudes. Then it performs the zonal mean over the longitudes.

The recipe starts by applying a Lanczos filter with low and upper cuts set to 0 and 0.1 respectively, and with a ratio of 2.3 to remove the high frequency influence. Then it computes the maximum value and the latitude at which it occurs for each longitude and time. Then it computes the histogram and estimates the Probability Density Function (PDF), as showcased in Figure 10. The metric produces NetCDF files with the histogram and the PDF for both speed and latitude. It also produces a plot for each dataset showing the histogram and the PDF.



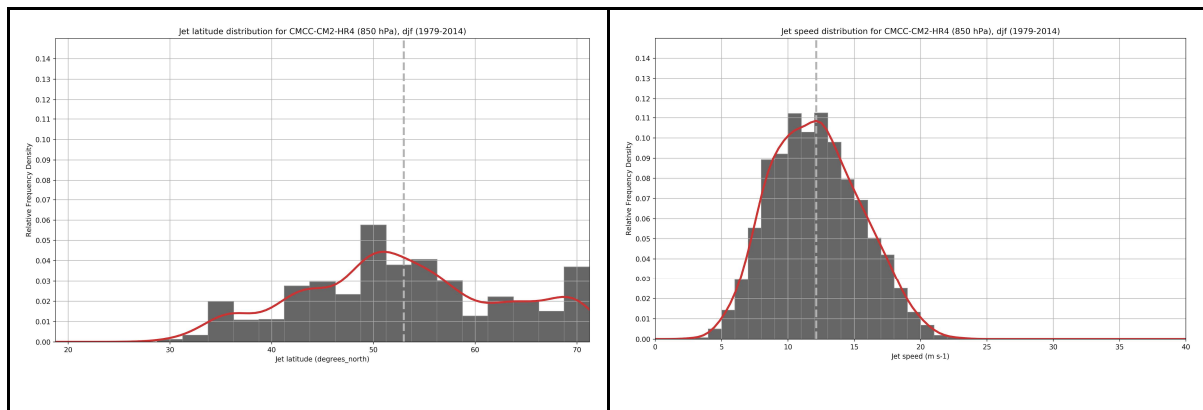


Figure 10: Show the jet latitude distribution (left) and jet speed distribution (right) during winter for the period 1979-2014. Histogram correspond to the actual values and the estimated PDF is plotted as a red line. Models shown are (top to bottom) HadGEM3-GC31-MM, HadGEM3-GC31-LM, HadGEM3-GC31-HM CMCC-CM2-VHR4 and CMCC-CM2-HR4 from HighResMIP

g) Weather regimes

The original scripts were submitted by Irene Mavilia (CNR) and then maintained by Federico Fabiano (CNR). Part of the metric is based on a couple of Fortran files compiled with f2py that has been impossible to compile inside the ESMValTool environment. To circumvent that issue, Federico Fabiano produced a new version that can work even if the fortran code is not available.

h) Sea-ice drift-strength feedback

The original scripts were submitted by David Docquier (UCL). The sea ice drift metric allows to quantify the relationships between Arctic sea-ice drift speed, concentration and thickness. A decrease in concentration or thickness, as observed in recent decades in the Arctic Ocean, leads to reduced sea-ice strength and internal stress, and thus larger sea-ice drift speed. This in turn could provide higher export of sea ice out of the Arctic Basin, resulting in lower sea-ice concentration and further thinning. Olason and Notz (2014) investigate the relationships between Arctic sea-ice drift speed, concentration and thickness using satellite and buoy observations. They show that both seasonal and recent long-term changes in sea ice drift are primarily correlated to changes in sea ice concentration and thickness. Our recipe allows to quantify these relationships in climate models.

The recipe requires the following variables:

- Daily Sea Ice Area Fraction / Percentage (*sic* for CMIP5 / *siconc* for CMIP6)
- Daily Sea Ice Thickness (*sithick*)
- Daily Sea-Ice Speed (*sispeed*)
- Grid-Cell Area for Ocean Variables (*areacello*)

The following preprocessors have been defined to be called by all the variables:

- **extract:** Performs a rectangular spatial extraction between $[0^\circ, 360^\circ]$ for the longitudes and $[40^\circ, 90^\circ]$ for the latitudes. After the regional extraction, it performs a multi-year monthly mean.

Four process-based metrics are computed based on the multi-year monthly mean sea-ice drift speed, concentration and thickness, averaged over the Central Arctic:

- Ratio between the modelled drift-concentration slope and the observed drift-concentration slope.
- Ratio between the modelled drift-thickness slope and the observed drift-thickness slope.
- Normalised distance between the model and observations in the drift-concentration space.
- Normalised distance between the model and observations in the drift-thickness space.

The output per model consists of a plot with the mentioned metrics, a sample of which is shown in Figure 11, and a csv file containing the slope, the interception, the slope ratio and the error.

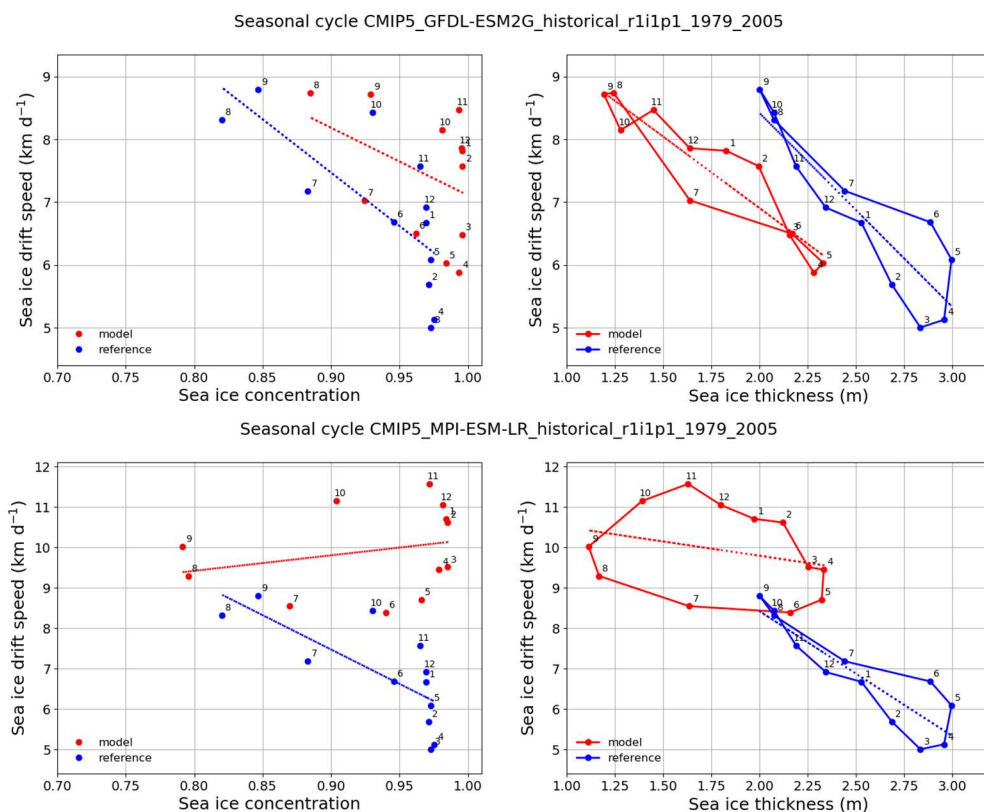


Figure 11: Sample of the output generated by recipe_seaice_drift.yml run with CMIP5 models and OSI-450-nh, PIOMAS and IABP observational datasets cmorized for ESMValTool. Seasonal cycles showing the normalised distance between models and observations.

i) E-vectors

The original script was submitted by Marie-Estelle Demory (former URead). The recipe calls the following variables to compute both the energy vectors and the heat flux metrics:

- Daily Eastward wind (u_a).
- Daily Northward wind (v_a).
- Daily Near-Surface Air Temperature (t_a).

Two preprocessors have been defined:

- `extract_200`: Extracts 200 hPa pressure level for *ua* and *va* to compute the energy vectors.
- `extract_850`: Extracts 850 hPa pressure level for *va* and *ta* to compute the heat fluxes.

The recipe also requires the user to set a window in days in order to define the window to use for the low pass Lanczos filter it uses to remove high frequency effects. The recipe also allows the user define the limits of the color scale in the different plots.

The recipe produces NetCDF files for the values two components of the energy vectors and heat flux and plots the global climatologies of each one of them. Both plots are illustrated in the following Figure 12.

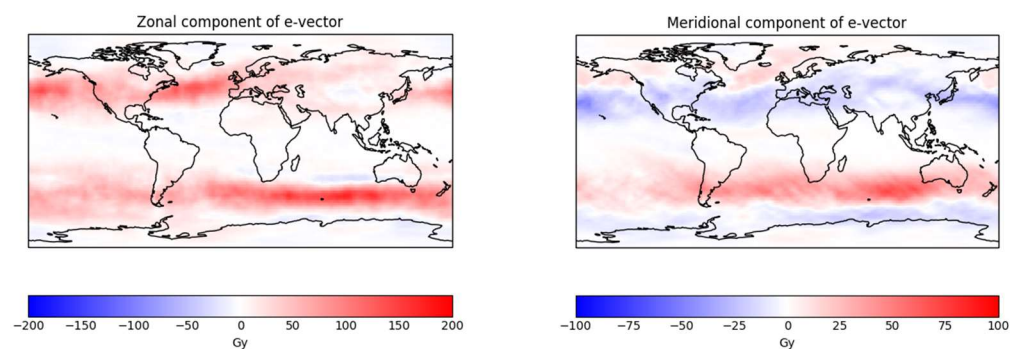


Figure 12: Sample of the output generated by `recipe_energy_vectors` run with one year of the ECMWF-IFS-LR model. Zonal and meridional components of the energy vector.

j) Energy budget

The original scripts for the energy budget metric were submitted by Benoit Vannière (URead). The goal of the metric is to compute the balance of fluxes of energy in and out of the Earth system and plot the results over a Trenberth Diagram template. The recipe calls the following variables :

- Monthly TOA Incident Shortwave Radiation (*rsdt*).
- Monthly TOA Outgoing Shortwave Radiation (*rsut*).
- Monthly Surface Downwelling Shortwave Radiation (*rsds*).
- Monthly Surface Net Downward Shortwave Radiation (*rsns*). This is an ESMValTool derived variable computed from the upwelling and downwelling Shortwave Radiation fluxes (*rsds* and *rsus*, respectively).
- Monthly All-sky longwave radiation (*rlut*).
- Monthly Surface Downwelling Longwave Radiation (*rls*).
- Monthly Surface Net Downward Longwave Radiation (*rlns*). This is an ESMValTool derived variable computed from the upwelling and downwelling Longwave Radiation fluxes (*rls* and *rlus*, respectively).
- Monthly Surface Upward Sensible Heat Flux (*hfss*).
- Monthly Surface Upward Latent Heat Flux (*hfls*).

Three preprocessors have been defined to be called by all the variables. They are a combination of several preprocessing operations available by ESMValTool:

- `global_mean`: Computes the spatial and temporal means over the whole domain.
- `land_mean`: Masks the ocean domain and proceeds to compute the spatial and temporal means.
- `ocean_mean`: Masks the land domain and proceeds to compute the spatial and temporal means.

It needs to be noted that the masking step requires the variable `sftlf` to mask every model properly. But if it's not found, Natural Earth masks are used without stopping the execution.

The preprocessing operations simplify greatly the diagnostic script: the input variables are now scalar quantities, instead of fields. This preprocessed data is then used as an input to the diagnostic script, which performs the required computations to obtain the following fluxes:

- Upward shortwave reflected by surface
- Shortwave reflected by clouds
- Shortwave absorbed by the atmosphere
- Upward longwave emitted by surface
- Net surface radiation
- Radiation absorbed by surface
- Radiation Net TOA
- and the Bowen Ratio.

The new computed quantities get stored in a single NetCDF file per model. Some of the variables get added in the Trenberth Diagram template for all the models, displayed in Figure 13. The template is included inside ESMValTool and it's path needs to be provided in the recipe. If not provided, the only output that gets generated is the NetCDF file.

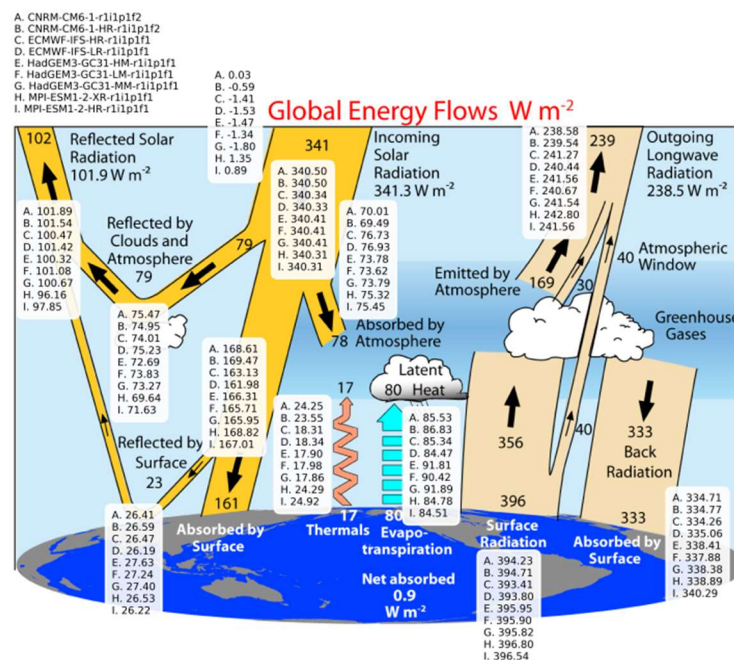


Figure 13: Sample of the output generated by `recipe_energy_budget` for the PRIMAVERA models. Computation of the global energy flows over a Trenberth diagram.

For the water budget, it needs to be noted that the original scripts were calling models and variables that are not generated in neither PRIMAVERA nor CMIP experiments. Therefore the implementation has not been done.

k) Fresh water transport

The original scripts were submitted by Ramón Fuentes Franco (SMHI). The goal of this metric is to estimate the fresh water transport across several straits.

- Monthly Sea Water Salinity (*so*)
- Monthly Sea Water Y Velocity (*vo*)

The following preprocessor has been defined in order to be called by the variable:

- `extract`: extract the region above 40° N to reduce the amount of data requested.

The recipe requires the user to provide the location and name of the desired straits. It then finds the best transect to compute the transport in the desired zone and computes it.

The recipe produces NetCDF files with the transport values for each strait and plots the timeseries of the annual and monthly transports, with both quantities displayed in Figure 14.

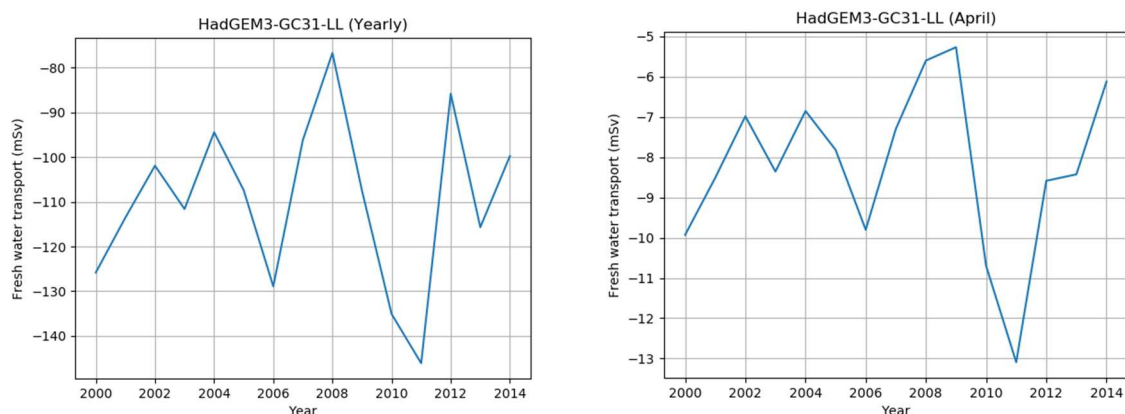


Figure 14: Sample of the output generated by `recipe_transports` for HadGEM3-GC33-LL CMIP6 model. First plot shows the yearly total time-series, while the second shows the same just for April.

l) Extreme precipitation

The original scripts were submitted by Alexander Baker (URead). The goal of this metric is to estimate the intensity of extreme precipitation events by computing the . The recipe calls the following variable:

- Daily Precipitation (*pr*)

The following preprocessor has been defined in order to be called by the variable:

- `season`: Performs temporal extraction for the dataset to start in month 3 of the starting year and finish in month 11 of the end year. Then, it performs a rectangular spatial

extraction between $[220^\circ, 320^\circ]$ for the longitudes and $[30^\circ, 80^\circ]$ for the latitudes. Finally it computes the maximum values for each season.

The recipe requires the user to provide the return period values in a list, as well as the confidence interval. The metric performs a Generalized Extreme Value (GEV) analysis to estimate the intensity of the precipitation extremes predicted by the models in the provided return periods. The calculations are performed by the extRemes R package through rpy2 Python package as this is a Python metric. For that reason, this is a good example for those users that may want to use a functionality that is only available in R but prefer to code in Python.

The recipe produces NetCDF files with the values of the GEV parameters and with the values of each one of the return periods requested.

I) Sea ice negative feedback

The original scripts were submitted by François Massonnet (UCL). The goal of the metric is to compute a process-based diagnostic named Ice Formation Efficiency (IFE) in order to evaluate the strength of the negative sea-ice thickness/growth feedback. This feedback causes late-summer negative anomalies in sea-ice area and volume to be partially recovered during the next growing season (Notz and Bitz, 2017). A chief cause behind the existence of this feedback is the non-linear inverse dependence between heat conduction fluxes and sea-ice thickness, which implies that thin sea ice grows faster than thick sea ice. The recipe calls the following variables:

- Monthly Sea Ice Thickness (*sit*)
- Grid-Cell Area for Ocean Variables (*areacello*)

It also allows the user to set several plot settings. To evaluate the strength of the negative sea-ice thickness/growth feedback, the diagnostic script computes the anomalies of the annual sea-ice volume estimated north of 80°N . The choice of this domain is motivated by the desire to minimize the influence of dynamic processes but also by the availability of sea-ice thickness measurements. Then, the increase in sea-ice volume until the next annual maximum is computed for each year. The IFE is defined as the regression of this ice volume production onto the baseline summer volume anomaly and the values per model get plotted in a common plot showcased in Figure 15.

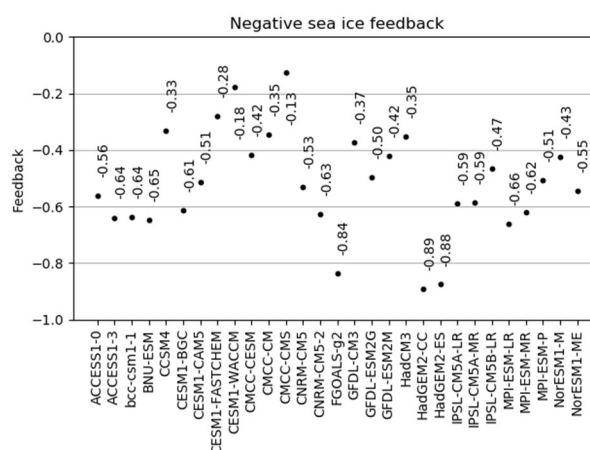


Figure 15: Ice formation efficiency (IFE) applied to the CMIP5 ensemble (Massonnet et al., 2018).

Key points for section 3.4

- PRIMAVERA scripts submitted by a wide variety of partners have been adapted in order to be integrated into ESMValTool.
- Due to ESMValTool strict coding guidelines, the quality of the scripts has improved notoriously.
- ESMValTool has proven to be a very flexible tool. PRIMAVERA scripts are very heterogeneous and still, they have been adapted to a common framework.
- Having an open common framework is key to reproducibility because it avoids situations in which mobility of researchers and students between institutes may result in loss of information.

4. Lessons learnt

4.1 Positive outcomes

- ESMValTool has evolved into a powerful and flexible tool that simplifies the development of routines for the post-process and multi-model comparison of climate variables.
- Adapting the scripts to a common framework and following a common structure simplifies greatly the sharing of scripts between partners by not requiring them to understand the particular way in which each one was developed.
- The PRIMAVERA focus to keep its metrics alive after the end of the project was an anomaly and, for that reason, an experiment in itself. The ESMValTool framework offers a possibility for this legacy to happen. All the experience gathered in how to achieve this goal will benefit future projects that decide to embrace this paradigm.
- The work done has allowed substantial contributions of PRIMAVERA to the second-order draft of the WGI Sixth Assessment Report of the IPCC, especially in Chapter 10.

4.2 Negative outcomes

Due to the lack of a stable version of ESMValTool and the long process required to merge a new metric into it, the arrival of the PRIMAVERA metrics to the master branch is not yet done (although any user can access them from the PRIMAVERA branch), but they are expected to be included in the coming months. The BSC is committed to achieve this goal.

5. Links built

The PRIMAVERA contribution to the ESMValTool has played a key part in the consolidation of the ESMValTool's community as a key player in the evaluation of Earth system models. We encourage partners to strengthen this link, as the future of the tool looks promising.

In the framework of this community, close collaboration with other projects like C3S-MAGIC, H2020-Crescendo, CMIP6-DICAD and others has been the norm during the development of ESMValTool, to the point that it felt more like a unique team than a collaboration between



projects. Continuation of this effort will continue as new projects like H2020 IS-ENES3 come to bolster the development of ESMValTool.