

Introduction to the ESMValTool

1. General Info

2. Installation

3. Selecting data and diagnostics

4. Recent developments in EMBRACE

5. Modify plots

6. Options to contribute your own diagnostics

7. How to create a new variable and a new diagnostic

Klaus Gottschaldt
Veronika Eyring (PI)

Martin Ewaldsson
Klaus Wyser (PI)

1. General Info: This Tutorial

Goals:

- Enable you to install and use the ESMValTool
- Encourage you to contribute your own diagnostics to the ESMValTool

Practical matters:

- General hands-on session technically not possible
- Slides and live presentation of the ESMValTool
- Option for a hands-on session on a remote server for a limited number of users

1. General Info: ESMValTool

- Designed for **comparing and plotting** climate parameters from **model, reanalysis** and **observational data**, given in NetCDF format.
- A **wealth of analysis routines** is inherited **from** previous and current **multi-model** intercomparison/verification **projects**.

The ESMValTool is an extension of the CCMVal Diagnostic Tool

(Gettelman, A. et al., A community diagnostic tool for chemistry climate model validation, Geosci. Model Dev., 5, 1061-1073, doi:10.5194/gmd-5-1061-2012, http://www.pa.op.dlr.de/CCMVal/CCMVal_DiagnosticTool.html).

- General license for use: **Open Source, no copyleft**
(see CCMValDiagTool_license.txt in the root folder of the tool)
An additional list of **references and acknowledgements** for the parts used is generated during each run.

- **Goal for EMBRACE: compile standard diagnostics** for CMIP5 style data
→ model skill assessment at the push of a button

2. Installation

Software requirements:

- Python 2.*

www.python.org



- NCL 6.1 or higher

www.ncl.ucar.edu



- ESMValTool (not yet officially released → contact us): tarball or from svn repository

- CMIP5 style datasets



e.g.:

esgf-data.dkrz.de/esgf-web-fe

3. Selecting data and diagnostics: Primer

IN

Model Output

- internal `./reformat`
- external shell scripts, cdo ...

Observations

- internal `./plot_type/input_data`
- external like another model

Basic control

- Set global flags
- Specify model / obs names, years and paths
- Specify diagnostic set

`./nml/namelist_*`

Diagnostics

- Plot type
- Variable, Input field type
- optional: additional models for individual diagnostics

`./diag_att/*.att`

Variable attributes

- set `var_att_info@` ...; calculate derived variables

`./var_att/*_att.ncl`

Plot attributes

- set `plot_type_info@` ... parameters for each plot type

`./plot_type_cfg/<diagn. set>/*.cfg`

Output

- Plots, NetCDF files

`./work`

OUT

3. Selecting data and diagnostics: Primer

IN

Model Output

- internal `./reformat`
- external shell scripts, cdo ...

Observations

- internal `./plot_type/input_data`
- external like another model

Basic control

`./nml/namelist_*`

- Set global flags
- Specify model / obs names, years and paths
- Specify diagnostic set

Diagnostics

`./diag_att/*.att`

- Plot type
- Variable, Input field type
- optional: additional models for individual diagnostics

Variable attributes

`./var_att/*_att.ncl`

- set `var_att_info@` ...; calculate derived variables

Plot attributes

`./plot_type_cfg/<diagn. set>/*.cfg`

- set `plot_type_info@` ... parameters for each plot type

Output

`./plots`

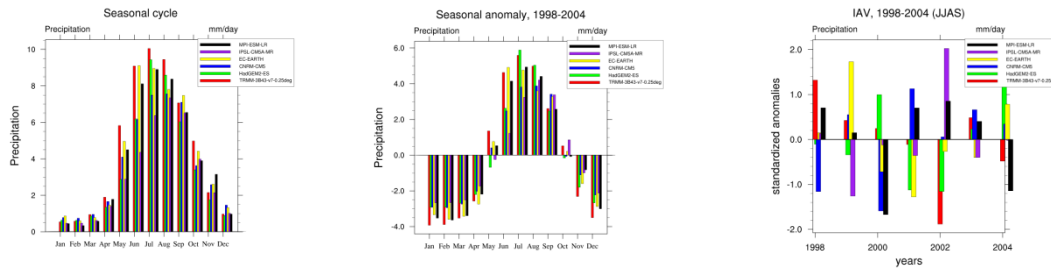
- Plots, NetCDF files

OUT

4. Recent developments in EMBRACE

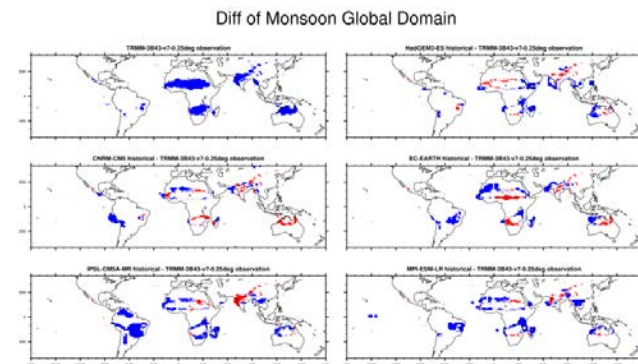
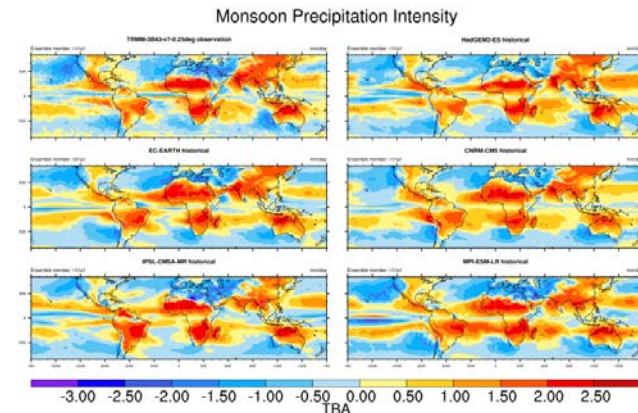
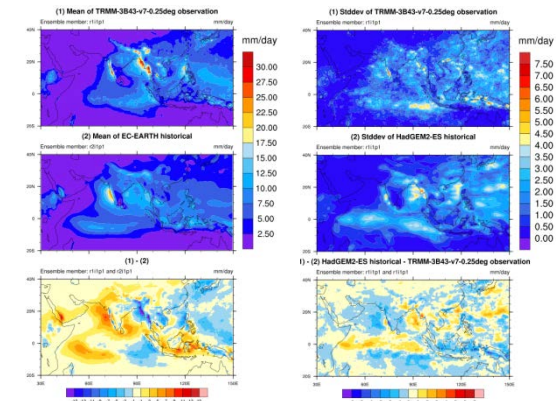
South Asian Monsoon Diagnostics: (MetOffice)

- [SAMonsoon_precip_basic \(_multipanel\)](#):
Mean and standard deviation (across all years) for each model;
Difference of the mean/stddev with respect to a reference model
- [SAMonsoon_precip_seasonal](#):
Climatology, seasonal anomalies and interannual variability



- [SAMonsoon_precip_intensity_multipanel](#):
Contour plot of summer (MJJAS) winter (NDJFM) difference normalized by climatology

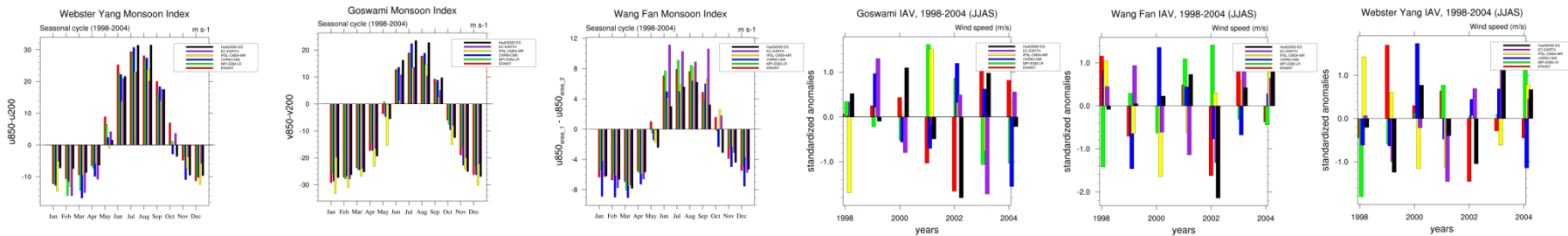
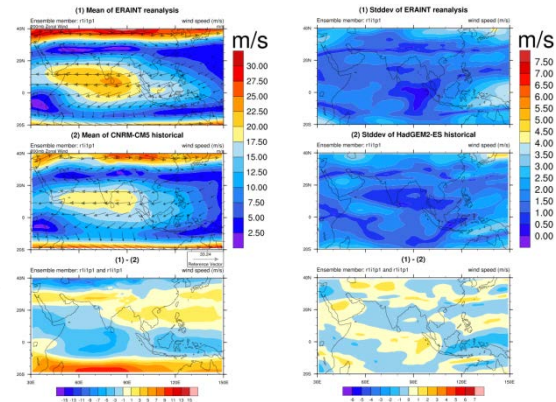
- [SAMonsoon_precip_global_domain_multipanel](#):
Similar to the above but contours precipitation only above a cut off level to identify Monsoon domain areas



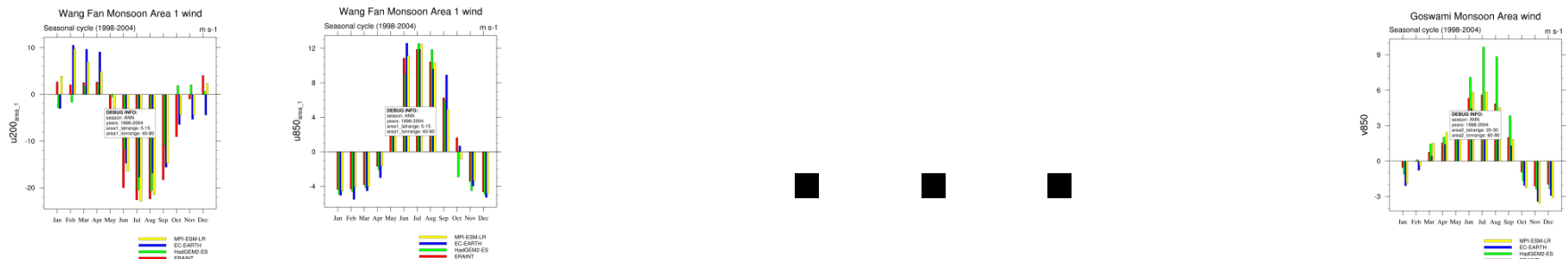
4. Recent developments in EMBRACE

South Asian Monsoon Diagnostics: (MetOffice)

- [SAMonsoon_wind_basic](#) ([_multipanel](#)): Mean and standard deviation (across all years) for each model; Difference of the mean/stddev with respect to a reference model.
- [SAMonsoon_wind_seasonal_index](#): Various monsoon indices computed over the monsoon season (JJAS) and as annual cycles.



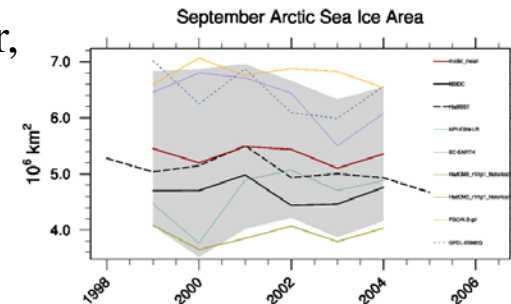
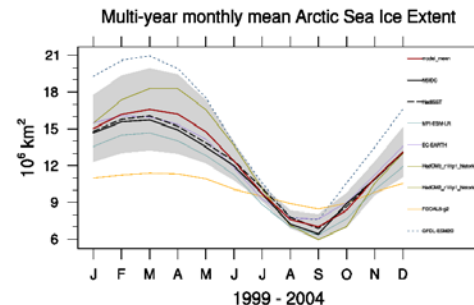
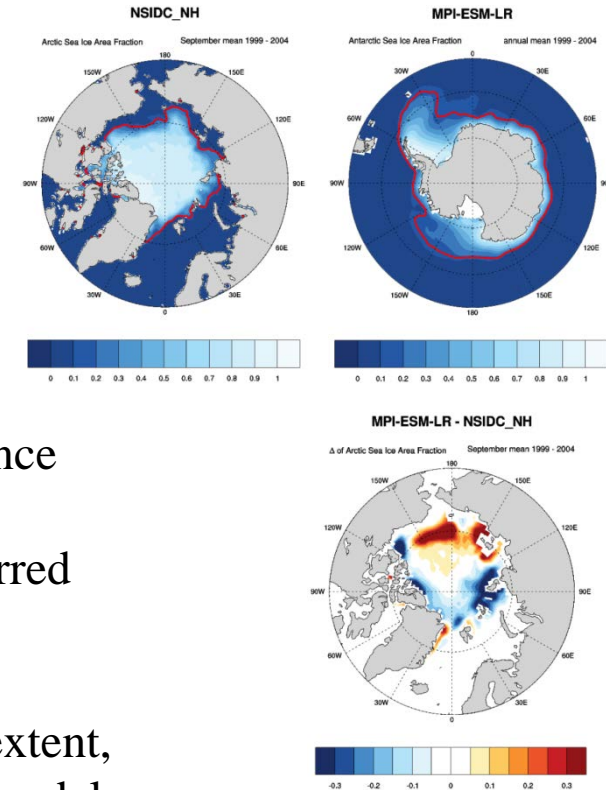
- [SAMonsoon_wind_seasonal](#): The components used when computing the above monsoon indices



4. Recent developments in EMBRACE

Sea Ice Diagnostics:

- **SeaIce_polcon:**
Polar stereographic plots of sea ice area concentration and extent for individual models or observational data sets, for Arctic and Antarctic regions, with flexible panelling.
- **SeaIce_polcon_diff:**
Polar stereographic plots of sea ice area concentration difference between individual models and reference data, for Arctic and Antarctic regions, with flexible panelling. All data are transferred to a common grid (Gaussian $1^\circ \times 1^\circ$) before comparison.
- **SeaIce_tslne:**
Time series line plots of total sea ice area and extent, for Northern and Southern hemispheres, with optional multi-model mean and standard deviation. One value is used per model per year, either annual mean or the mean value within a selected month.
- **SeaIce_ancyc:**
As above, but for the annual cycle (multi-year monthly mean values).



5. Modify plots

IN

Model Output

- internal `./reformat`
- external shell scripts, cdo ...

Observations

- internal `./plot_type/input_data`
- external like another model

Basic control

- Set global flags
- Specify model / obs names, years and paths
- Specify diagnostic set

`./nml/namelist_*`

Diagnostics

- Plot type
- Variable, Input field type
- optional: additional models for individual diagnostics

`./diag_att/*.att`

Variable attributes

- set `var_att_info@ ...`; calculate derived variables

`./var_att/*_att.ncl`

Plot attributes

- set `plot_type_info@ ...` parameters for each plot type

`./plot_type_cfg/<diagn. set>/*.cfg`

Output

- Plots, NetCDF files

`./work`

OUT

6. Options to contribute your own diagnostics:

1. Join the core development team with full access to:

[Subversion repository](#)

[Mantis bug tracker](#)

[Teamsite & Wiki](#)

ESM-Diagnostic - Revision 753: /source/trunk

- ..
- CCMValDiagTool license.txt
- README
- diag_att/
- doc/
- main.ncl
- main.py
- masks/
- ncl_code/
- nml/
- plot_type/
- plot_type_cfg/
- python_code/
- reformat/
- rgb/
- temp_ncl_code/
- var_att/

Powered by [Apache Subversion](#) version 1.7.6 (r137077)

DLR

Logged in as: [gott_kl](#) (Klaus-Dirk Gottschaldt - manager) 19-06-2013 21:09 CEST

[Main](#) | [My View](#) | [View Issues](#) | [Report Issue](#) | [Change Log](#) | [Roadmap](#) | [Summary](#)

Assigned to Me (Unresolved) [^] (1 - 6 / 6)	
0008661	Put tutorial templates MyDiag & MyVar into svn Implementation - 18-06-13 17:45
0010049	Implement regridding to a common grid and dependent diagnostics Implementation - 18-06-13 17:21
0008659	merge fix for function Z into trunk Requirement - 18-04-13 10:35
0008664	Move observational data out of source and rename them Unknown - 20-02-13 11:32
0008666	merge bug fix for T3M and T2Ms in E06FIG07.ncl to svn Requirement - 12-12-12 18:52
0008656	Implement the functionality of \$PATH Implementation - 12-12-12 11:02

Unassigned	
0009835	Using Imple
0009113	*Outs Imple
0008701	Limit Imple
0008703	Data Requ

Reported by Me [^] (1 - 8 / 8)	
0008661	Put tutorial templates MyDiag & MyVar into svn Implementation - 18-06-13 17:45

Earth System Model Validation Tool > Test_Wiki > SeaIce diagnostics

A diagnostic and performance metric tool for the evaluation of Earth System Models with observations

Home

Recently Modified

MyDiag diagnostic
Home

SeaIce diagnostics
SeaIce_polcon
SeaIce_polcon_diff

Sea Ice diagnostics

This page gives an overview of the SeaIce diagnostics implemented in ESMValTool

- Available diagnostics with figure examples
- How to run a test suite of the SeaIce diagnostics

Available diagnostics

Each diagnostic is referred to as a plot type, which on the technical side core documents the existing plot types for the SeaIce diagnostics. All diagnostics listed below to see an example figure of respective

- SeaIce_polcon**: Polar stereographic plots of sea ice area concentration Antarctic regions, with flexible panelling.
- SeaIce_polcon_diff**: Polar stereographic plots of sea ice area concentration Antarctic regions, with flexible panelling. All data are transferred to a coordinate standard deviation. One value is used per model per year, either annual or monthly.
- SeaIce_tslines**: Time series line plots of total sea ice area and extent, for standard deviation. One value is used per model per year, either annual or monthly.
- SeaIce_ancyc**: As above, but for the annual cycle (multi-year monthly)

SeaIce test suite

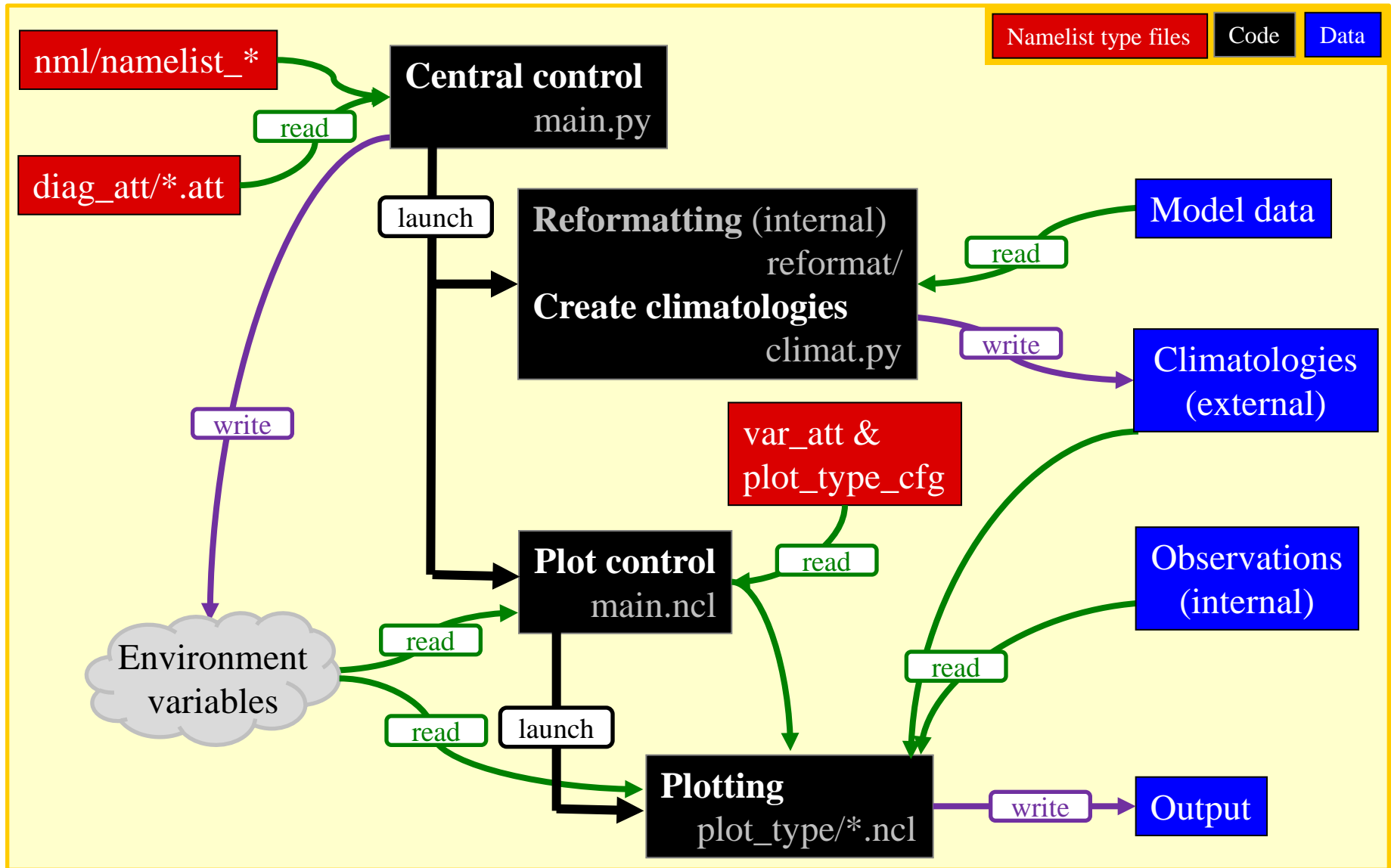
2. Implement your changes in a snapshot of the ESMValTool (tarball or checkout from repository)

3. Give us your diagnostics „as is“ (if they are written in NCL)

Your contribution is very welcome!

Please contact us.

7. How to create a new variable and a new diagnostic: Code components



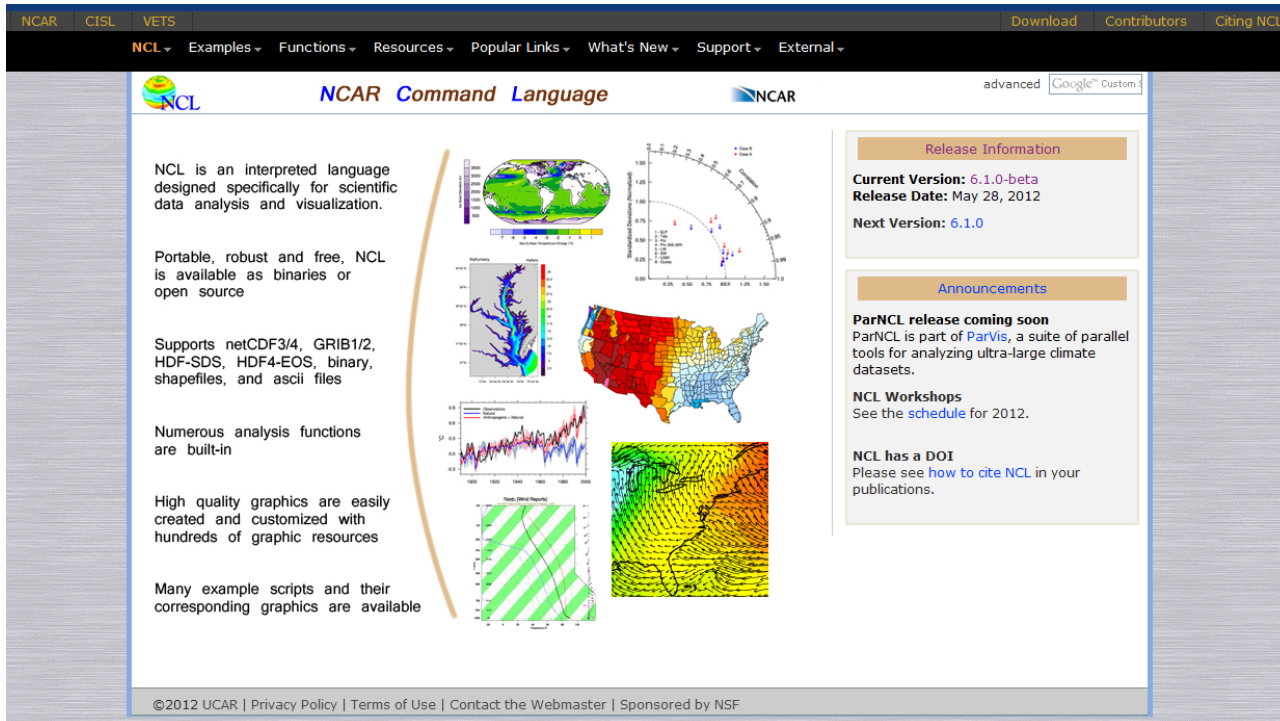
7. How to create a new variable and a new diagnostic

Adjust template files

- **./nml/namelist_MyDiag.xml**: Global flags, diagnostic sets to do, and models that shall be evaluated by all those diagnostics are specified here
- **./diag_att/diag_MyDiag.xml**: This in general contains a collection of diagnostics. Each diagnostic is defined by a plot type, a variable and the corresponding (input) field type. Data that shall be evaluated by specific diagnostics only may be added with *<model>* specifiers (as in ./nml/namelist_MyDiag.xml) to the respective diagnostics
- **./plot_type_cfg/MyDiag/MyDiag.cfg**: This file contains plot specific control parameters in NCL syntax, specified as attributes of the variable *plot_type_info*. All *.cfg files for a diagnostic set need to be in the same folder, as specified in ./diag_att/diag_MyDiag.xml
- **./var_att/MyVar.att**: Variable specific parameters are defined here in NCL syntax, as attributes of the variable *var_att_info*. Derived variables require a *calculate function*. Here temperature at 200 hPa is extracted from the CMIP5 *ta* variable and defined as *MyVar*.
- **./plot_type/MyDiag.ncl**: This is the actual plotting routine, as specified by a *<plot_type>* entry in ./diag_att/diag_MyDiag.xml
- **./doc/MASTER_authors-refs-acknow.txt**: This is the central lookup table for references & acknowledgements that might be selected within the plot_type routines

7. How to create a new variable and a new diagnostic

- Please take existing code from the ESMValTool as template that comes close to your needs, and consult the NCL website (<http://www.ncl.ucar.edu/>)



The screenshot shows the NCAR Command Language website. The header includes navigation links for NCAR, CISL, and VETS, along with Download, Contributors, and Citing NCL. The main content area features the NCL logo and the title "NCAR Command Language". On the left, there is a list of features: "NCL is an interpreted language designed specifically for scientific data analysis and visualization.", "Portable, robust and free, NCL is available as binaries or open source", "Supports netCDF3/4, GRIB1/2, HDF-SDS, HDF4-EOS, binary, shapefiles, and ascii files", "Numerous analysis functions are built-in", "High quality graphics are easily created and customized with hundreds of graphic resources", and "Many example scripts and their corresponding graphics are available". The center of the page displays several example plots: a global map, a polar plot, a river network map, a US map with a color scale, a time-series plot, and a topographic map. On the right, there are sections for "Release Information" (Current Version: 6.1.0-beta, Release Date: May 28, 2012, Next Version: 6.1.0) and "Announcements" (ParNCL release coming soon, NCL Workshops, NCL has a DOI). The footer contains copyright information: ©2012 UCAR | Privacy Policy | Terms of Use | Contact the Webmaster | Sponsored by NSF.

Good to know about NCL ...

- Parameters are global by default and available in all routines, even if not explicitly passed
- Parameters need to be deleted explicitly before changing dimensions or type
- Parameter exchange with Python is via environment variables and temporary text files
- Index count starts from 0

8. Hands on session

To try out ESMValTool at the test server, (note: max eight users), see separate paper for login procedure

Getting started:

1. ssh from your own machine
2. unpack the tarball, *tar xf ESMValTool_revision780.tar.gz*
3. try running one of the shorter namelists,
 1. *./main.py nml/namelist_MyDiag.xml*
 2. *./main.py nml/namelist_standardized.xml*
4. examine and edit the configuration files, rerun namelist
 1. *diag_att/* - collection of diagnostics in namelist
 2. *var_att/* - variable specific settings/transforms
 3. *plot_type_cfg* – plot specific settings
5. The *display* command is available to visualize the figures in the *plots/-*directory

Connect with	Password
ssh wp4-tutorial-1@54.217.232.211	wp4-two-1

Connect with	Password
ssh wp4-tutorial-2@54.217.232.211	wp4-two-2

Connect with	Password
ssh wp4-tutorial-3@54.217.232.211	wp4-three-3

Connect with	Password
ssh wp4-tutorial-4@54.217.232.211	wp4-four-4

Connect with	Password
ssh wp4-tutorial-1@54.217.232.212	wp4-two-1

Connect with	Password
ssh wp4-tutorial-2@54.217.232.212	wp4-two-2

Connect with	Password
ssh wp4-tutorial-3@54.217.232.212	wp4-three-3

Connect with	Password
ssh wp4-tutorial-4@54.217.232.212	wp4-four-4