# IS-ENES – WP 9

## D 9.4 – Report on the pilot study on the implementation of evaluation toolkits on CNRS-IPSL ESM output

**Abstract:**
The objective of JRA3 was to create an interdisciplinary infrastructure that facilitates the scientific evaluation of complex ESMs. Firstly, a survey of evaluation methods and problems was performed (D9.1) that formed the basis for the building of the Evaluation Portal (D9.2). In Task 9.3, standards for the general description of evaluation software were defined and implemented for a set of four toolkits: AeroCom, CCMval-Diag, HOAPS and MCMS (D9.3). The present deliverable reports on the implementation of these tools on IPSL ESM outputs. The objective was to test their functionality and to iron out any potential problems in their application. AeroCom, CCMval-Diag, and HOAPS tools were shown to be easily applied on basic and more complex diagnostics, thanks to their mature stage. The first test of the MCMS tool showed the need for further development, in order to build a universal procedure no matter the model. This was achieved just in time by the MCMS developers. A technical description of the application of the four tools is provided in Appendix. Few suggestions or recommendations are also proposed in order to improve some limitations, which are mainly related to very last developments of the tools.


| | | | |
|---|---|---|---|
| **Grant Agreement Number:** | **228203** | **Proposal Number:** | **FP7-INFRA-2008-1.1.2.21** |
| **Project Acronym:** | **IS-ENES** | | |
| **Project Co-ordinator:** | **Dr Sylvie JOUSSAUME** | | |

| | | | | | |
|---|---|---|---|---|---|
| **Document Title:** | Pilot study on the implementation of evaluation toolkits on CNRS-IPSL ESM output | | | **Deliverable:** | D 9.4 |
| **Document Id N°:** | | | 1 | **Date:** | 12.03.2013 |
| **Status:** | | | | | |

| | |
|---|---|
| Filename: | ISENES_D9.4.doc / ISENES_D9.4.pdf |
| Authors: | **Dr Brigitte KOFFI and Dr Michael SCHULZ** |

| | |
|---|---|
| **Project Classification:** | |

| **Approval Status** | | |
|---|---|---|
| **Document Manager** | **Verification Authority** | **Project Approval** |
| Brigitte Koffi and Michael Schulz | Sylvie Joussaume | |

# REVISION TABLE

| Version | Date | Modified Pages | Modified Sections | Comments |
|---|---|---|---|---|
| A01 | 2013-01-14 | ALL | ALL | IPSL Initial Version |
| 1 | 2013-03-28 | | | Minor corrections |
| 1.1 | 2013-03-30 | | | Final version |

# Table of Content

# Executive Summary

This document presents results of the testing of selected tools that are used within the European community to evaluate different components of Earth System Models. More specifically, it concerns the AeroCom, CCMVal, HOAPS and MCMS tools that have been already documented as part of D9.3 and on the ENES evaluation portal (https://verc.enes.org/models/evaluation-portal). We provide a technical description of the tools (Appendix A), as well as results and recommendations on their application (Section 3).

The study is based on evaluation tests conducted from September 2012 to January 2013. The version of the tools may have changed since that time. The original AeroCom tool has been run from its own server, while the (simplified versions of the) other tools were installed and tested on the CNRS-IPSL (Institut Paul Simon Laplace) CICLAD (Calcul Intensif pour le Climat, l'Atmosphère et la Dynamique) computing cluster.

Note that the conversion of the model output files to the adequate format was performed by the tools' providers. Our study revealed a good and easy functionality of the AeroCom, CCMVal, and HOAPS tools. Some tool specific aspects could be improved based on the testing here: In the case of AeroCom, we highly recommend providing a tutorial that explains the main tool architecture, the definition and setting of the input and outputs. We also suggest more flexibility in the choice of the options. We propose the CCMVal-Diag developers to provide more detailed/updated information on how to access the observation datasets that are needed to evaluate the performance of the model in reproducing the so-called "E06" benchmark set of diagnostics. No significant need for HOAPS improvement has been identified. For MCMS, the testing was done by the tool providers themselves because of unexpected running problems of the source code. It led to further MCMS developments i.e., to a universal procedure that equally works for all ESM SLP outputs that is still currently being validated.

# 1. INTRODUCTION

IS-ENES is the infrastructure project of the European Network for Earth System Modelling (ENES). It combines expertise in Earth system modelling, computational science, and studies of climate change impacts. It provides a service on models and model results to modelling groups and to the users of model results. The research activities aim at improving the efficient use of high-performance computers, the model evaluation tools, the access to model results and climate services for the impact community.

The objective of JRA3 was to create an interdisciplinary infrastructure that facilitates the scientific evaluation of complex Earth System Models (ESMs). This infrastructure aims to standardize, harmonize and simplify – where possible - the tools and methodologies used in existing or past model inter-comparison projects to evaluate the basic quality of an ESM model. Firstly, a survey of evaluation methods and problems was performed (D9.1) that formed the basis for the building of the Evaluation Portal (D9.2). Task 9.3 consisted in the formatting, development and documentation of observational data and evaluation tools. Standards for the documentation of toolkits were defined and implemented for a set of four tools: AeroCom (Aerosol Comparisons between Observations and Models), CCMVal-Diag (Chemistry-Climate Model Validation Diagnostic), HOAPS (Hamburg Ocean Atmosphere Parameters and Fluxes from Satellite Data) and MCMS (MAP Climatology of Mid-latitude Storminess).

D9.4 reports on the implementation of the four selected tools for IPSL ESM outputs. The objective was to test their functionality and to iron out any potential problems in their application. The original AeroCom tool has been run from its own server, while the (simplified versions of the) other tools were installed and tested on the IPSL CICLAD (Calcul Intensif pour le Climat, l'Atmosphère et la Dynamique) computing cluster. The evaluation tests were conducted from September 2012 to January 2013. Note that the version of the tools may have changed since that time. The pilot study does not intend to provide an exhaustive review of all analyses possible, but rather focuses on basic variables/processes of interest for ESM modeling. In general, the CMIP5 standard IPSL-CM5A model output files required significant pre-processing work to be fed into the evaluation tools. This formatting step is not explicitly reported in the present study: either we used already formatted files (e.g., previous IPSL REPROBUS model outputs in CCMVal format) or the conversion was performed on purpose by the tool's developers (HOAPS, MCMS, AeroCom). In the case of MCMS, significant further development was more generally required in order to build a universal procedure for all ESM SLP outputs.

Results and recommendations are reported in Section 3 in a concise format (the so-called "*Review Table*" hereafter) that varies depending on the tool and its application, but always consists in three "*Downloading/Installation*", "*Setting and Running*" and "*Analysis/Outputs*" sections.

The technical description of the tools and of their application is provided in Appendix A. It consists in the documentation of "*Directories*", "*Scripts*", "*Software needed to run the scripts*", "*Setting and Running the scripts*", and "*Outputs*" sub-sections.

# 2. TESTED TOOLS

The general documentation of the four evaluation tools provided hereafter arises from the D9.3 ISENES deliverable and is repeated hereafter. The technical description, that is part of this D9.4 deliverable, is given in Appendix A.

## AeroCom tool

The AeroCom project [Schulz *et al.*, 2009] intends to document differences of aerosol component modules of global models and to assemble data-sets for aerosol model evaluations. It focuses on the most important aerosol components, i.e. dust, sea salt, sulfate, nitrate, black carbon, and particulate organic matter, and on the sum of these components (called "total aerosol"). The AeroCom tool allows evaluating the model outputs by comparison to different observational datasets. Observed surface concentrations, aerosol optical depths (AOD) and extinction vertical profiles are computed from measurements from global and regional networks (ARM, GAW, AERONET, IMPROVE, EMEP, ACTRIS, Aeroce, Earlinet etc.) and satellite observations (POLDER/Parasol, MODIS, MISR, TOMS, AVHRR and CALIOP). The tool package is not freely downloadable. Modelers are rather offered to upload their model outputs following a standard data-protocol. Evaluation results are then elaborated in joint studies (e.g. in support of the 5th IPCC report, e.g. Myhre et al., 2013). A fast feedback and preview of the evaluation is accessible as an image catalogue through the graphical web interface via the AeroCom website. The functionality of the AeroCom tool has been tested for IPSL model output(s) as reported in section 3.1.

## CCMVal-Diag

From Gettelman et al. [2012]: "The CCMVal-Diag tool is a flexible and extensible open source package that facilitates the complex evaluation of global models. Models can be compared to other models, ensemble members (simulations with the same model), and/or many types of observations. The tool can also compute quantitative performance metrics. The initial construction and application is to coupled Chemistry-Climate Models (CCMs) participating in CCMVal, but the evaluation of climate models that submitted output to the Coupled Model Intercomparison Project (CMIP) is also possible. The package has been used to assist with analysis of simulations for the 2010 WMO/UNEP Scientific Ozone Assessment and the SPARC Report on the Evaluation of CCMs. The CCMVal-Diag tool is described and examples of how it functions are presented, along with links to detailed descriptions, instructions and source code. The CCMVal-Diag tool is supporting model development as well as quantifying model improvements, both for different versions of individual models and for different generations of community-wide collections of models used in international assessments. The code allows further extensions by different users for different applications and types, e.g. to other components of the Earth System. User modifications are encouraged and easy to perform with a minimum of coding." The CCMVal-Diag tool has been applied in the frame of D9.4 of IS-ENES to reproduce some of the plots of Eyring et al., [2006] and Cionni et al. [2011] (see section 3.2).

### HOAPS

The Hamburg Ocean Atmosphere Parameters and Fluxes from Satellite Data (HOAPS) Simulator allows to evaluate Earth System Models directly against satellite observations of evaporation and precipitation [Andersson *et al*., 2011, 2013]. The HOAPS Simulator uses the HOAPS data, the only generally available satellite-based dataset with consistently derived global fields of both evaporation and precipitation and hence of freshwater flux (evaporation minus precipitation) as well as 12 other parameters which provide all basic state variables that are necessary to determine the evaporation. Except for the NODC/RSMAS Pathfinder SST data set, all variables are derived from SSM/I passive microwave satellite data over the ice free global ocean. The HOAPS-3 dataset covers the time span from 1987 to 2005. The climate modellers are given the possibility to evaluate their model results by applying the HOAPS climatology. For this purpose, the HOAPS simulator package provides a specific interface between the spatio-temporally unevenly sampled satellite observations with homogeneous model data. The tool is applied to model output with a high temporal resolution (at least 3-hourly) and mimics the satellite sampling by a retrieval based filter as well as an orbit mask for the satellite positions. Hence, it is possible to quantify (or eliminate for better model judgement) the systematic biases between model simulations and satellite based data due to retrieval properties and/or to the temporal coverage of the satellites. One mask is used to filter the model output according to the satellite orbits. A second filter is generated from the precipitation satellite data and is used to mask out the evaporation (latent heat), wind, and water vapor parameters in case of precipitation events. It is not possible to retrieve those parameters from the satellite when considerable precipitation occurs (more than about 1 mm/h), due to the dominance of the precipitation signal in the retrieved radiances. After generating the filter mask for the precipitation model output, they can be applied to other model variables (evaporation/latent heat flux, wind speed, water vapour) and a comparison with the corresponding data from the HOAPS data set is possible. Such a comparison for IPSL-CM5 model outputs is reported in section 3.3 for precipitation.

### MCMS

The MAP Climatology of Mid-latitude Storminess (MCMS) is essentially an algorithm for the objective recognition of extra-tropical low-pressure systems [Bauer *et al*., 2013]. This storm tracker algorithm is an improved version of the MCMS algorithm, originally described by Bauer and Del Genio [2006]. The algorithm can be performed on gridded sea level pressure data such as those produced by ESM simulations and reanalysis projects. Thus the simulation results can be compared to the observation based reanalysis data. The MCMS algorithm is unique in that apart from allowing for the objective recognition of the low-pressure centers and for the following of their course it also allows for the objective recognition of the extent of the low-pressure systems and for a series of parameters depending on the systems as well as for the systems that are connected to neighbouring deeper systems being encircled by the same closed isobaric lines. Furthermore, the MCMS is a useful tool that can be used in other purposes besides the evaluation of simulations from Global Climate Models. MCMS can be used for the study of cyclones themselves, it may prove useful for researchers who want to contextualize their data (observed or modelled) by the presence or absence of a nearby cyclone, it can be used as a weather sensitive filter for any sort of data or model component, and in ecological, oceanographic, aerosol and pollution studies. Finally the MCMS algorithm may be used for the inter-comparison of simulations. The tool has been applied in the frame of D9.4 of IS-ENES to evaluate the IPSL-CM5 sea level pressure fields (section 3.4).

# 3. TEST RESULTS

The description of the tests and their results are summarized hereafter. For sake of brevity, the full technical descriptions of the tools, of the input/output setting and of the resulting outputs are provided in Appendix A. The test does not include the formatting of the model output files (see introduction).

## AeroCom tool

A working copy of the AeroCom tool was downloaded on the CNRS-IPSL local "asterix" machine by doing a "svn checkout" of the "aerocomIDL" directory from the Aerocom User server (Meteorologisk Institutt, Oslo, Norway). A copy of the attached AeroCom observational database was also performed. The testing consisted in evaluation of the LMDz-INCA component of the IPSL ESM model in its ability to reproduce the aerosol optical depth (variable name: OD550_AER) and aerosol extinction (variable name: EC5503D_AER) at 550 nm. These are bulk aerosol parameters of direct use for evaluating aerosol forcing in an ESM. The model outputs are compared to the CALIOP global aerosol profile product [Koffi *et al.*, 2012] that was developed in the frame WP9 Task 3 of IS-ENES (see deliverable D9.3). The test has been performed at regional (AOD and EC5503D_AER) and local (EC5503D_AER) scales, i.e. over Europe [35°N-70°N; 10°W-70°E] and near the Chernobyl Nuclear Power Plant [50.5°N-52.0°N; 29.0°E-31.0°E], respectively.

Technical documentation of the tool exists only as comment lines in the IDL code. Support is available by the main developers M. Schulz and J. Griesfeller at Met.No. The run-time information (e.g. input/output options and corresponding flags) is provided upon execution to the main routine (aerocom_main.pro) via command-line or run-scripts (see section A.1). We highly recommend providing a readme and/or tutorial that explain the main AeroCom tool architecture, the definition and setting of the input (in ./modellists/*.txt) and output (in ./include/*mic_include_*.pro)* options, as well as the running script ("StartBatch.job" hereafter). While some aspects of the tool could be improved (see Table 1a), its application to the evaluation of the 2006 and 2007 simulated AOD and aerosol extinction profiles against CALIOP 3D global dataset was successful. One concern is that the 2D maps of the aerosol extinction are produced at the $10^{th}$ level for both the (LMDz-INCA) model and the CALIOP data, i.e. at different altitudes (at ~8.7 km and 1km altitudes, respectively). An interpolation of the model extinction at the CALIOP altitude level (i.e., at 1 km for level 10) should be performed instead. We also propose modifying some default options in the running routines and/or adding options choice in the *mic_include_*.pro* file.

Review Table n° 1a: AeroCom test

| Testing steps | Comments | Suggestions/ Recommendations |
|---|---|---|
| **Download/Installation**<br>A "svn checkout" of "aerocomIDL" directory from the Aerocom User server was done. | Requires an AeroCom user account. | |
| **Setting and running** | A tutorial is missing | Provide a tutorial |
| **Setting of input options**<br>in ./modellists/***LSCE_Caliop***.*txt* | See section A.1 | |
| **Setting of output options**<br>in ./include/*mic_include**_OD550**.pro*<br>in ./include/*mic_include**_EC5503D_AER**.pro* | See section A.1 | |
| **Running**<br>**./** StartBatch.job | The names of the input and output options files are defined in the job. | |
| **Analysis/outputs** | ./plots and ./data subdirectories are created. Output directory is defined in aerocom_main.pro | Provide a tutorial |
| Map of AOD (**Figure 1b**)<br>./plots/*/OD550_AER_*an**YEAR**_m*** _**REG**_MAP.**ext*** | Ok<br>Monthly and annual maps | |
| Plot of AOD zonal mean (**Figure 1b**)<br>./plots/*/OD550_AER_*an**YEAR**_m*** _**REG**_ZONAL.**ext*** | Ok<br>Monthly and annual plots. | |
| Regional extinction profiles values (100m vertical resolution)<br>./data/***_an**YEAR**_m*****monthly_**REG**_*.sav* files | Ok | Retrieving of .sav files to be included in the tutorial |
| Plot of regional aerosol extinction profiles (**Figure 1a**)<br>./plots/*/***_an**YEAR**_m*****monthly_**REG**_REGIONPROFILE.**ext*** | Ok<br>One plot per model with the satellite profile superimposed for each selected region, season and year. | A "multi-year" option would allow year to year comparisons. |
| Map of aerosol extinction at the 10th level (**Figure 1b**)<br>./plots/*/EC5503D_AER_*an**YEAR**_m*****monthly_**REG**_MAP.**ext2*** | Ok<br>Monthly and annual maps at the 10th level.<br>The model/satellite level (i_3dPlotLevel) is set in aerocom_plot_maps_v4.pro | Maps at a given altitude (*e.g. at 1 km*) should be performed instead. |
| Mean aerosol extinction values at the 10th level (**Table 1b**)<br>./data/AerocomTableMAPS_*.txt files | Ok<br>One file per model, completed at each new run with monthly/annual mean values over each processed parameter and region. | |

With *: the model or satellite name (LSCEv2c.A2.CTRL and CALIOP3) following AeroCom nomenclature; **: the monthly (01, 02, .., 12) and annual (ALLYEAR) files; ***: the seasonal (MAM, JJA, SON and DJF) and annual (ALLYEAR) files; ***YEAR***: the selected year (2007); ***REG***: the region ('EUR' and 'CHERNO'); ***ext1***: ps and png file extension; ***ext2***: ps., ps.png (high resolution) and png file extensions.

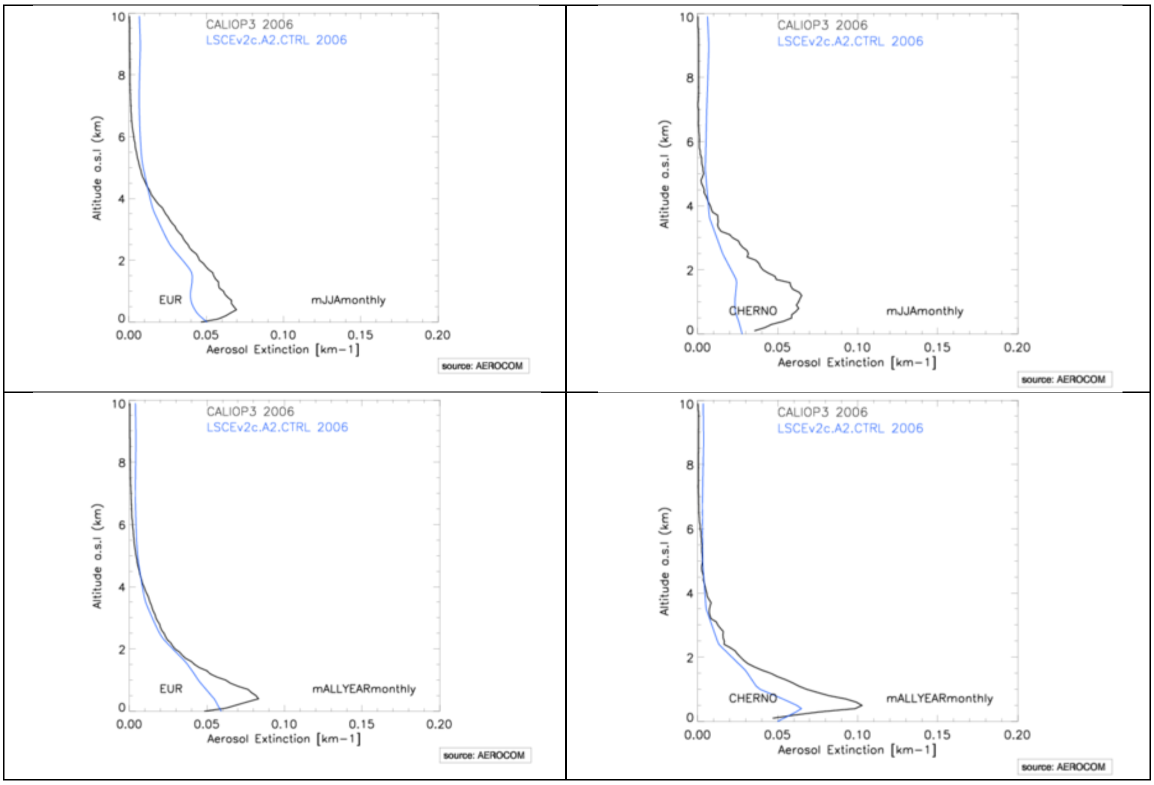



**Figure 1a**: LMDz-INCA versus CALIOP 2006 JJA (top) and annual (bottom) mean aerosol extinction profiles ($km^{-1}$) over Europe (left) and over Chernobyl (right) region.

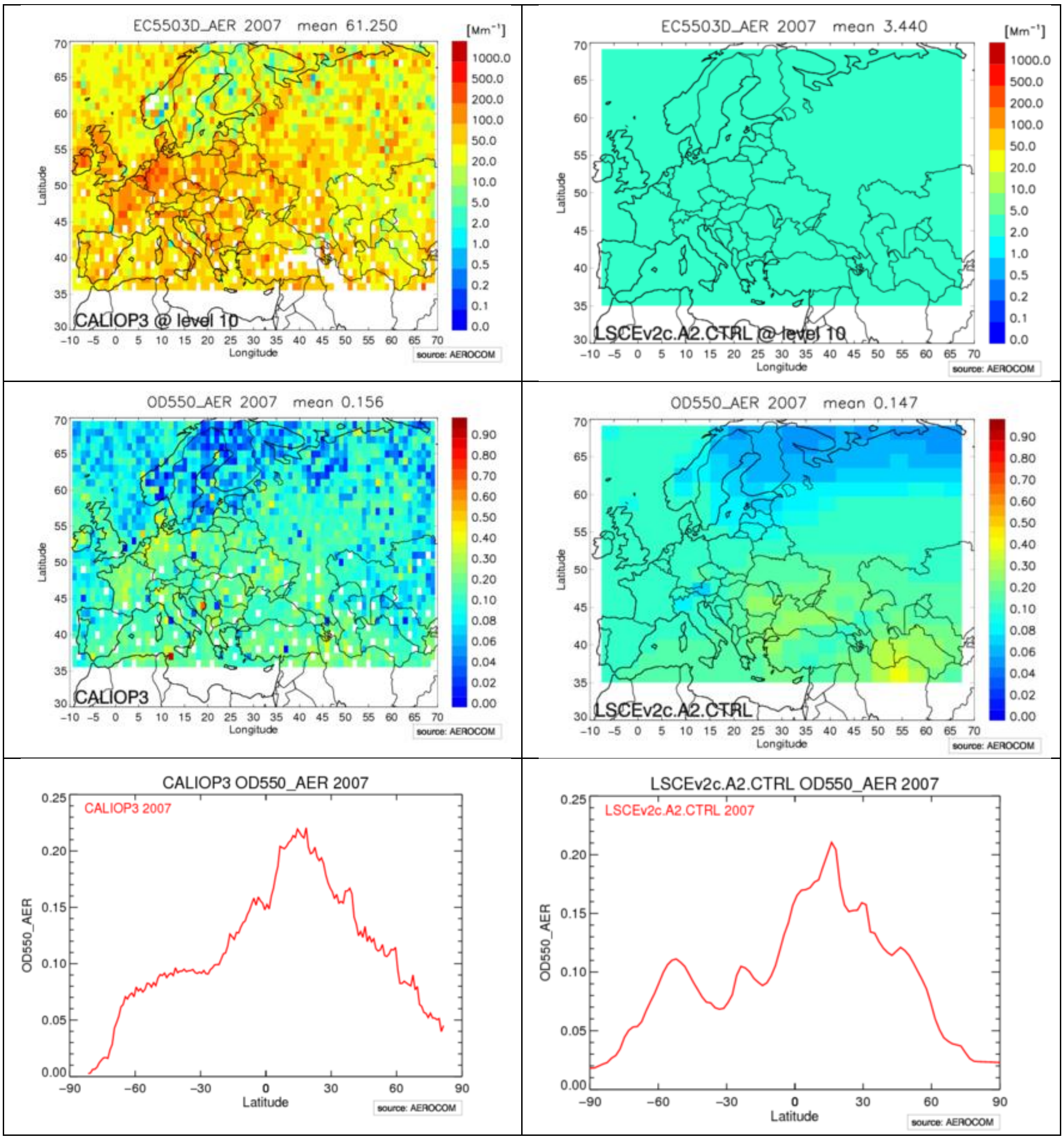



**Figure 1b**: 2007 LMDz-INCA versus CALIOP mean aerosol extinction ($Mm^{-1}$) at the 10th vertical level (top) and AOD (middle) over Europe. The global zonal mean AOD is also provided (bottom). See section 3.1 for comments.

**Table 1b: AerocomTableMAPS_CALIOP3.txt (year 2007)**

AREAMEAN,CALIOP3,2007,0000,OD550_AER,01,EUR,0.1228 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,02,EUR,0.1332 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,03,EUR,0.1527 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,04,EUR,0.1816 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,05,EUR,0.1742 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,06,EUR,0.2042 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,07,EUR,0.2004 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,08,EUR,0.1852 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,09,EUR,0.1563 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,10,EUR,0.1425 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,11,EUR,0.1294 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,12,EUR,0.1056 ,
AREAMEAN,CALIOP3,2007,0000,OD550_AER,ALLYEAR,EUR,0.1559 ,
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,01,EUR,60.33 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,02,EUR,60.61 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,03,EUR,69.71 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,04,EUR,65.70 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,05,EUR,55.52 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,06,EUR,64.63 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,07,EUR,64.40 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,08,EUR,60.27 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,09,EUR,64.05 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,10,EUR,61.88 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,11,EUR,58.60 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,12,EUR,49.61 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,ALLYEAR,EUR,61.25 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,01,CHERNO,120.8 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,02,CHERNO,21.15 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,03,CHERNO,156.5 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,04,CHERNO,54.70 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,05,CHERNO,51.65 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,06,CHERNO,71.60 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,07,CHERNO,87.23 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,08,CHERNO,70.56 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,09,CHERNO,101.2 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,10,CHERNO,125.8 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,11,CHERNO,38.85 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,12,CHERNO,65.54 ,Mm-1
AREAMEAN,CALIOP3,2007,0000,EC5503D_AER,ALLYEAR,CHERNO,82.29 ,Mm-1

## CCMVal-Diag

The CCMVal-Diag tool has been applied to reproduce some of the plots of Eyring et al., [2006] (i.e., to the so-called "E06 diagnostics" hereafter) but for LMDZ-REPRODUS model (stratospheric CCM of the IPSL ESM) only. The plotting of time-height sections of ozone at selected stations (i.e., Figure 11 of Cionni et al., 2011) was also tested. The evaluation summary (Review Table n° 2), as well as the technical details (Appendix A) are only related to these specific analyses. Note also that LMDZ-REPRODUS outputs in CCMVal format were used. Therefore, the CCMVal-Diag option that converts model outputs to CF compliant CCMVal-2 netCDF format is not part of this study.

The CCMVal-Diag installation is well documented and easy. The main concern when applying the CCMVal-Diag tool to "E06" diagnostics [Eyring *et al*., 2006] was to know about the variable/data type combination to be used for each plot (e.g. for E06FIG03). Especially because not all related observation data are provided with the toolkit, which leads to extra work to download them from the BADC website and to rename them following CCMVal-Diag nomenclature. The same apply to the LMDZ-REPRODUS model outputs. Note also that the BADC paths given in the readme file (CCMVal-Diag_README_20090826.txt) are no more valid.

Review Table n° 2: CCMVal-Diag Version 2.1

| | Comments | Recommend |
|---|---|---|
| **Download/Installation** | Download of the file "CCMValDiagTool_Vers2.1.tar" Ok | |
| **Setting and running** | No particular difficulty.<br>First tests failed because of missing observation data (ERA40, NCEP) in the test packaging. | Provide the complete *variable/data/E06FIG* list* in the E06Diag.att file.<br>Provide all observations for "E06" diagnostics in the packing, or an updated list of the BADC paths and files. |
| **Analysis/outputs** | | |
| E06FIG01.ps: **Figure 2a** | Ok | |
| E06FIG02.ps: **Figure 2b** | Ok | |
| E06FIG04.ps: not produced | "conditional expression yields a missing value" in line 40 of proc_E06FIG04.ncl | To be fixed |
| E06FIG05_H20.ps: **Figure 2c** | Ok (=Fig.6 of Eyring et al. 2006) | |
| E06FIG07_H20.ps: **Figure 2d** | Ok | |
| E06FIG08.ps: **Figure 2e** | plotting anomaly | Fix the plotting anomaly.<br>Reduce the title size. |
| E06FIG09.ps: **Figure 2f** | Ok | |
| E06FIG12_Cly.ps: **Figure 2g** | Ok | |
| tropo_OzoneSondes.ps: **Figure 2h** | Only Ok for some stations | |

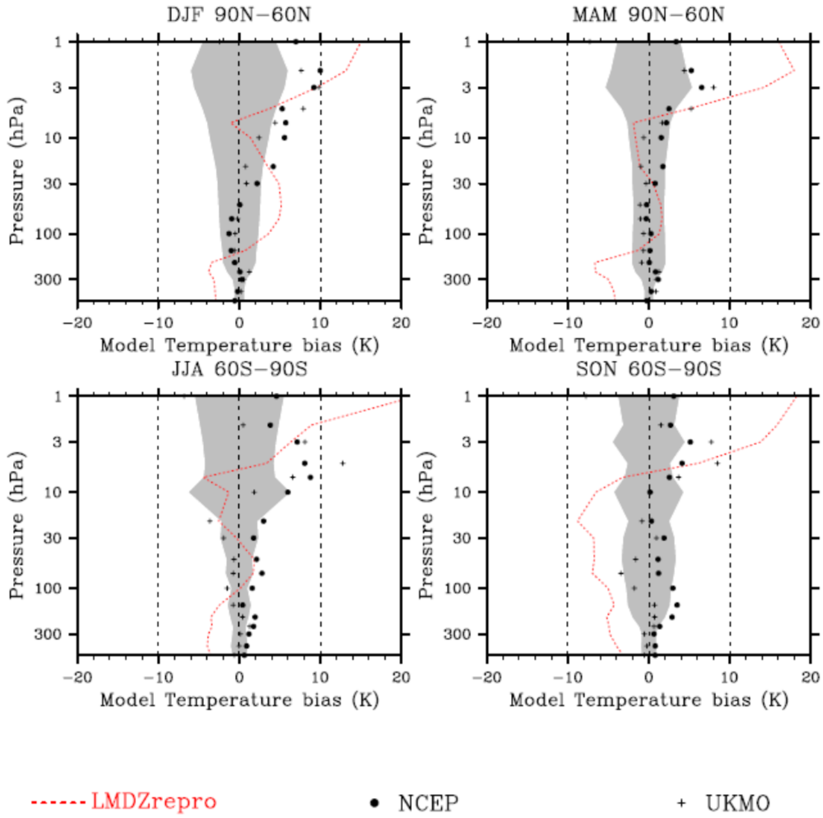



**Figure 2a:** Climatological mean temperature biases for (top) 60–90_N and (bottom) 60–90°S for the (left) winter and (right) spring seasons from LMDz-REPROBUS model simulations. The climatological means for the CCMs and NCEP data from 1980 to 1999 and for UKMO from 1992 to 2001 are included. Biases are calculated relative to ERA-40 reanalyses. The grey area shows ERA-40 plus and minus 1 standard deviation (s) about the climatological mean.

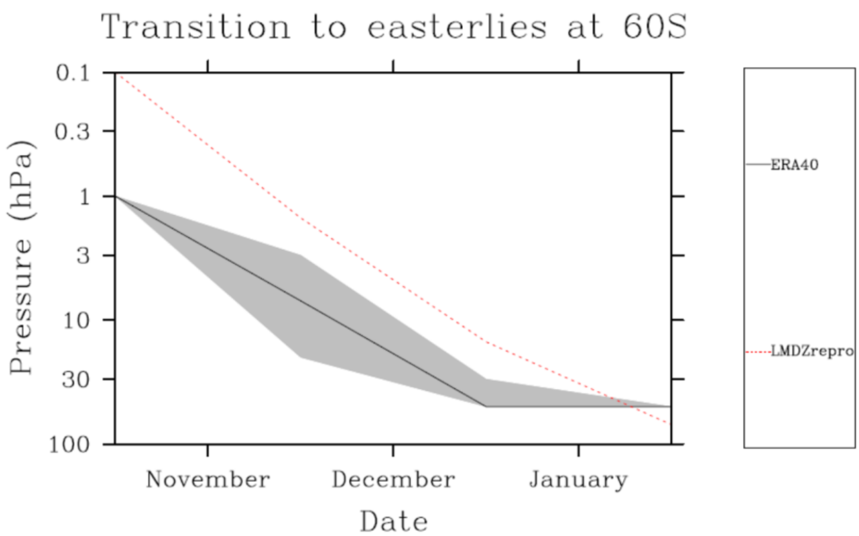



**Figure 2b:** Descent of the zero zonal mean wind lines at 60_S based on the climatological mean annual cycle calculated from the monthly mean zonal mean winds. The grey area indicates the variation of the timing in the transition from westerlies to easterlies for ERA-40 due to a plus or minus one inter-annual standard deviation in the mean annual cycle. Tick marks refer to the first of the month, and climatological means are calculated as in Figure 2a.

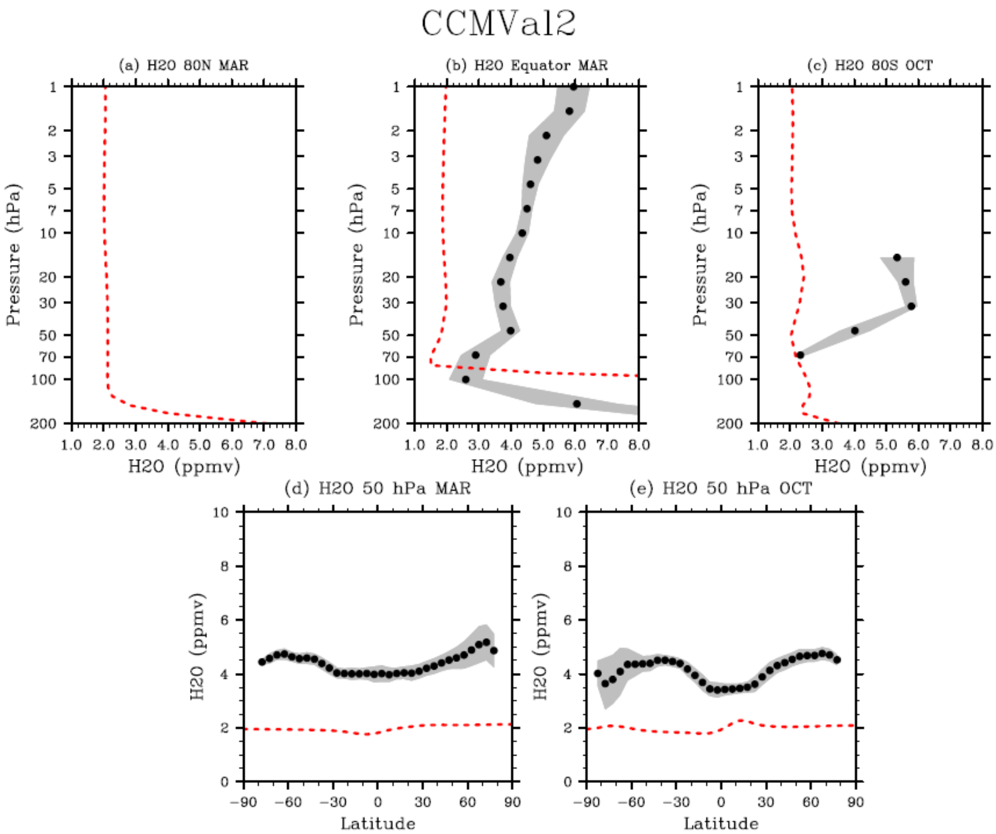



**Figure 2c:** Climatological zonal mean $H_20$ mixing ratios from the LMDz-REPROBUS model (red) and HALOE in ppmv. Vertical profiles at (a) 80_N in March, (b) 0_ in March, and (c) 80_S in October. Latitudinal mean at 50 hPa in (d) March and (e) October. The grey area shows HALOE plus and minus 1 standard deviation about the climatological zonal mean.

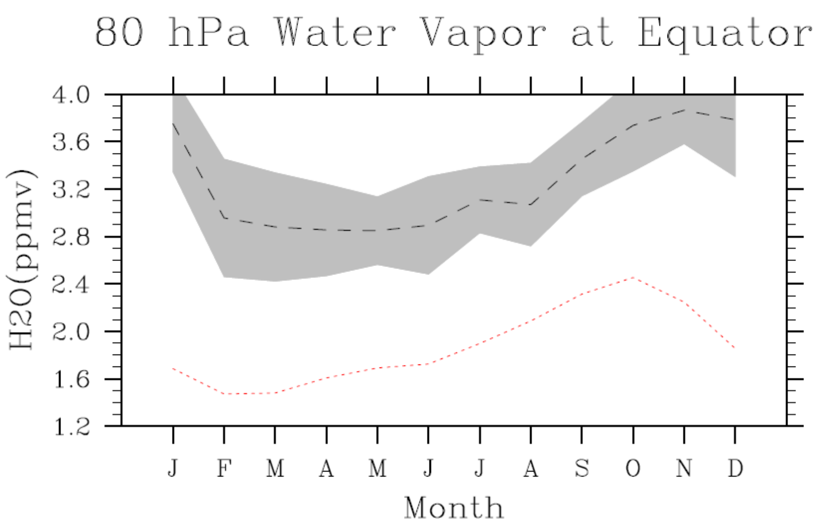



**Figure 2d:** Seasonal variation of climatological means at 100 hPa at the equator for water vapor. LMDz-REPROBUS model fields (red) for the 1990s are compared to the 1991–2002 HALOE water vapor climatology (grey area).

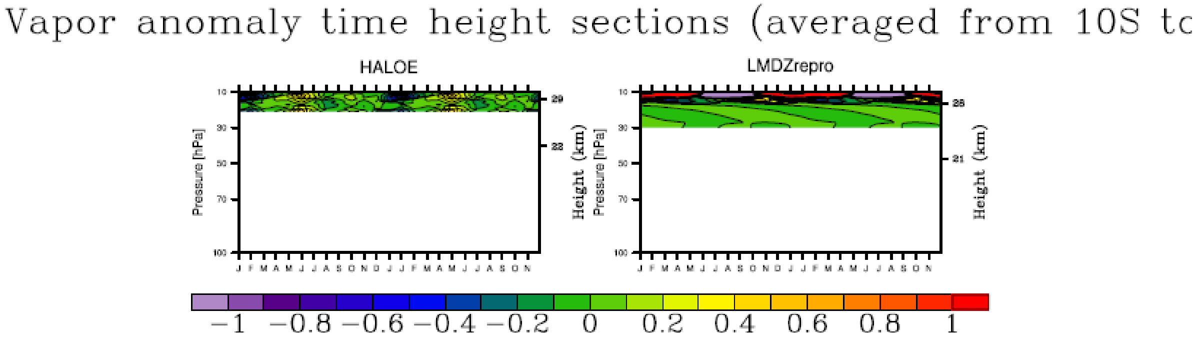



**Figure 2e:** Time-height sections of water vapor mixing ratio shown as the deviation (in parts per million by volume) from the time mean profile, averaged between 10_S and 10_N (''tape recorder'') for the LMDz-REPROBUS model (rigt) and HALOE data (left). Two consecutive cycles (*with a plotting anomaly*) are shown.

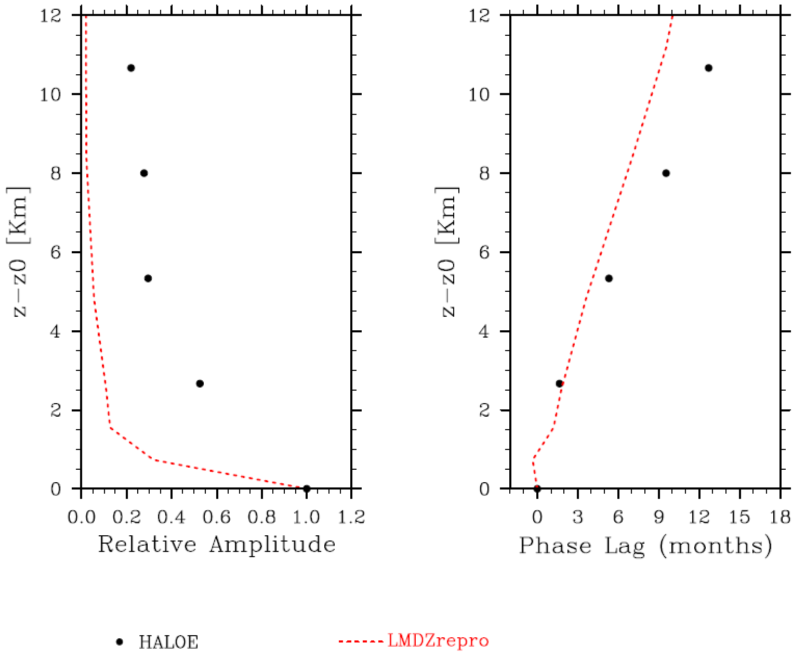



**Figure 2f:** Vertical variation of (a) amplitude and (b) phase lag of annual cycle of water vapor averaged between 10_S and 10_N. The amplitude is normalized to unity and phase lag is set to zero at the level where the amplitude is maximum (between 16 and 20 km). The vertical axis in both plots is the distance from level of maximum amplitude. Solid circles are HALOE observations.

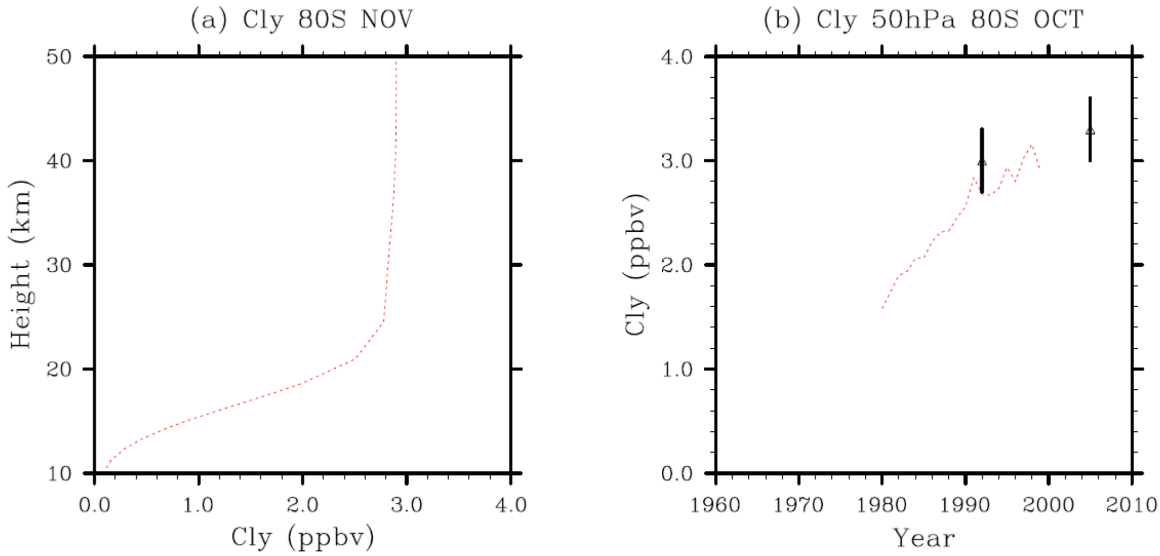



**Figure 2g**: (a) Climatological mean vertical profiles (1980 to 1999) at 80_S in November for Cly in ppbv. (b) Time series of October mean Antarctic Cly at 80_S from LMDz-REPROBUS model simulations. Estimates of Cly from HALOE HCl measurements in 1992 and Aura MLS HCl in 2005 are shown in addition (see Eyring *et al*. [2006] for more details).

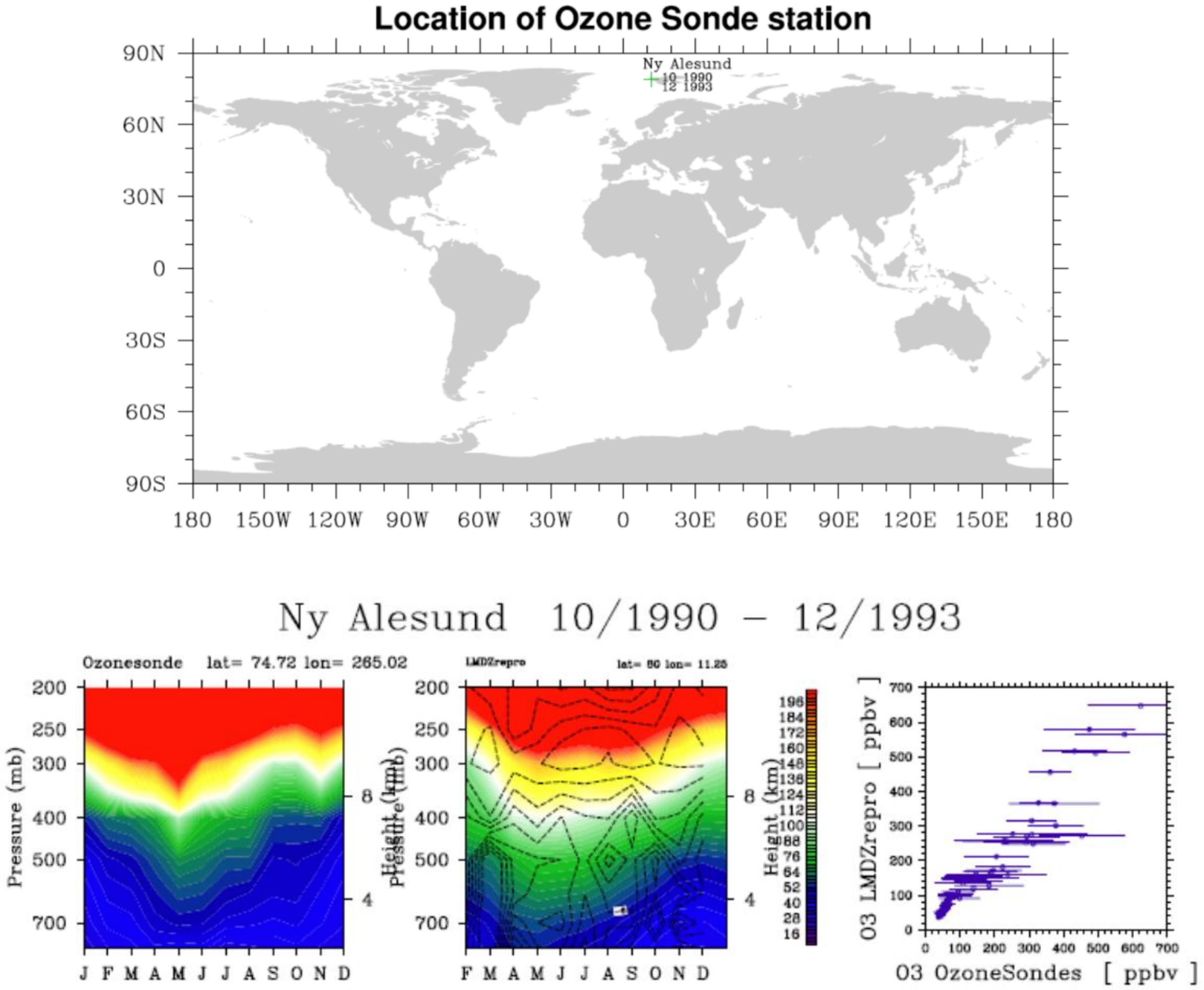



**Figure 2h:** Comparison of tropospheric ozone (ppb) from the LMDz-REPROPUS model outputs with ozonesonde observations at Ny Alesund station. The first column shows the annual cycle in the observations, the second column is the equivalent plot from the model. The final column compares all points, with bars indicating the standard deviation in the observations. Where the model overpredicts (underpredicts) observations by more than one standard deviation, the point is plotted in red (purple); these points are shown in the second column by the solid (dashed) contours. See Cionni *et al.* [2011] for more details on the extension of the CCMVal-Diag tool to tropospheric diagnostics and on the observation data.

## HOAPS

The HOAPS Simulator allows for the evaluation of the Earth System Models directly against satellite observations of evaporation and precipitation. It has been applied to the precipitation fields of both the low (IPSL-CM5A-LR; 2000-2005 run) and the medium (IPSL-CM5A-MR; historical 1990-2005 run) resolution versions of the IPSL ESM model. The outputs (pr3hr_IPSL-CM5A-LR_200001010130-200512312230_pr_2000-2005_grads.eps and pr_3hr_IPSL-CM5A-MR_historical_r1i1p1_199001010130-200512312230_pr_2000-2005_grads.eps) are shown in Figure 3a and 3b, respectively. The HOAPS tool was found to be well and concisely documented and its application is easy. The main concern when applying the tool was to first re-gridding HOAPS data to IPSL-CM5 grid, which was done by the MPI partner. The NCL re-gridding script (regrid_hoaps.ncl) and the *_IPSL-CM5A-LR_File_Info.txt that can be adapted to other models, together with the description of the setting and running steps are provided in Appendix A.3.

| | Comments | Recommendation |
|---|---|---|
| **Download/Installation** | HOAPS_SIM_IPSL.tar downloaded on cilcad | |
| **Setting and Running** | The HOAPS data used for comparison have first to be remapped to the model resolution. | Keep the script that re-grids HOAPS data to IPSL ESM model in the standard tool packaging as an example. |
| **extract_year.sh** | Ok | |
| **run idl_batch.bat** | Ok | |
| **run filter_data.sh** | Ok | |
| **Analysis/outputs** | 12 plots in the same output file (precipitation parameter) | |
| Climate mean: model | Ok (**Figure 3**; Line 1) | Increase the size of axis labels |
| Climate mean: model with the simulator applied | Ok (**Figure 3**; Line 1) | Increase the size of axis labels |
| Climate mean: HOAPS data | Ok (**Figure 3**; Line 1) | Increase the size of axis labels |
| Climate mean: Model – model with simulator | Ok (**Figure 3**; Line 2) | Increase the size of axis labels |
| Climate mean : HOAPS - model | Ok (**Figure 3**; Line 2) | Increase the size of axis labels |
| Climate mean : HOAPS – model with simulator | Ok (**Figure 3**; Line 2) | Increase the size of axis labels |
| Climate mean: Model – model with simulator (%) | Ok (**Figure 3**; Line 3) | Increase the size of axis labels |
| Climate mean: HOAPS – model (%) | Ok (**Figure 3**; Line 3) | Increase the size of axis labels |
| Climate mean: HOAPS – model with simulator (%) | Ok (**Figure 3**; Line 3) | Increase the size of axis labels |
| Zonal means | Ok (**Figure 3**; Line 4) | Increase the size of axis labels |
| Global mean time series | Ok (**Figure 3**; Line 4) | Increase the size of axis labels |
| Global mean annual cycle | Ok (**Figure 3**; Line 4) | Increase the size of axis labels |

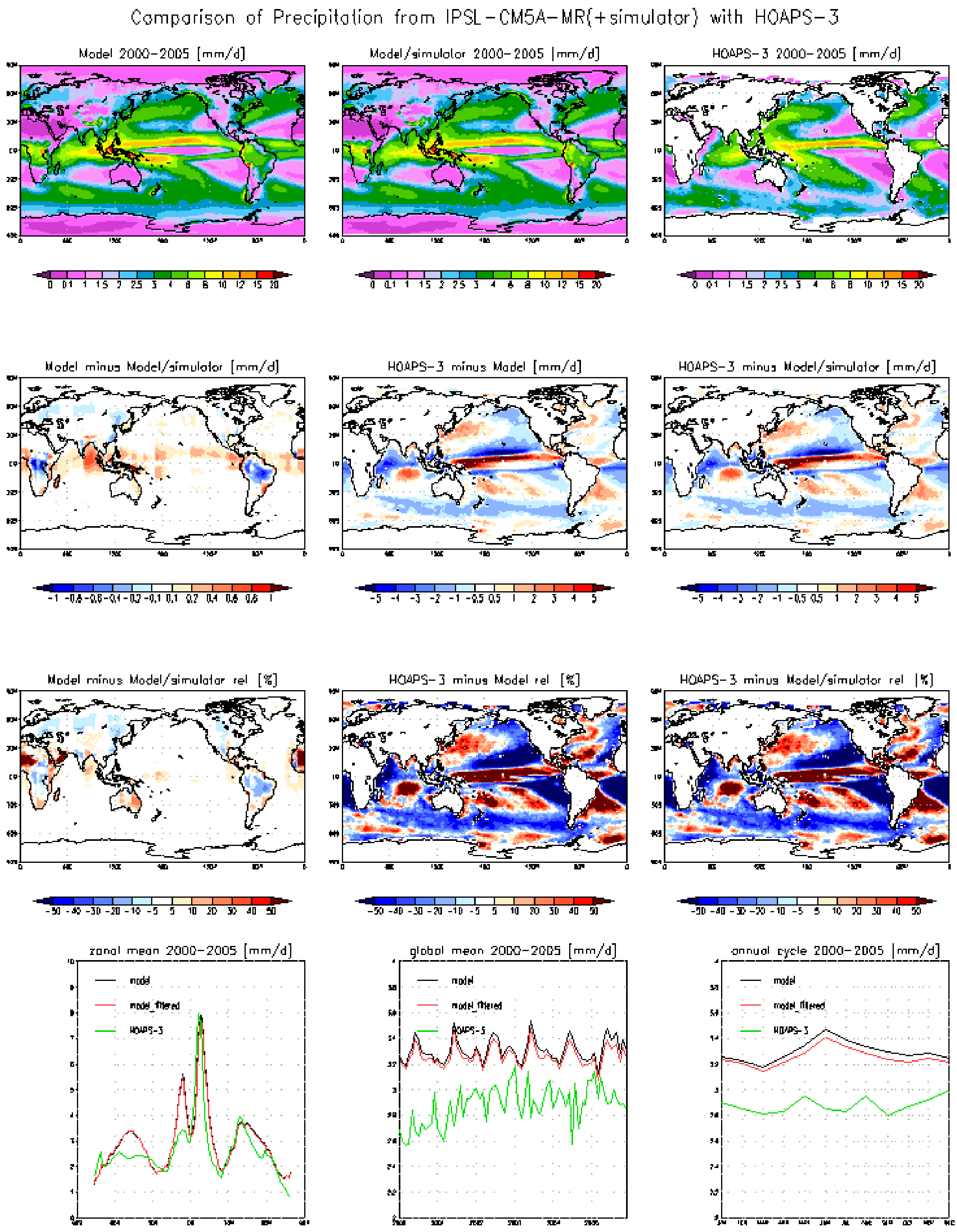



**Figure 3a**: Comparison of IPSL-CM5A-MR model (with and without HOAPS simulator) and HOAPS satellite derived 2000-2005 mean precipitation fields.

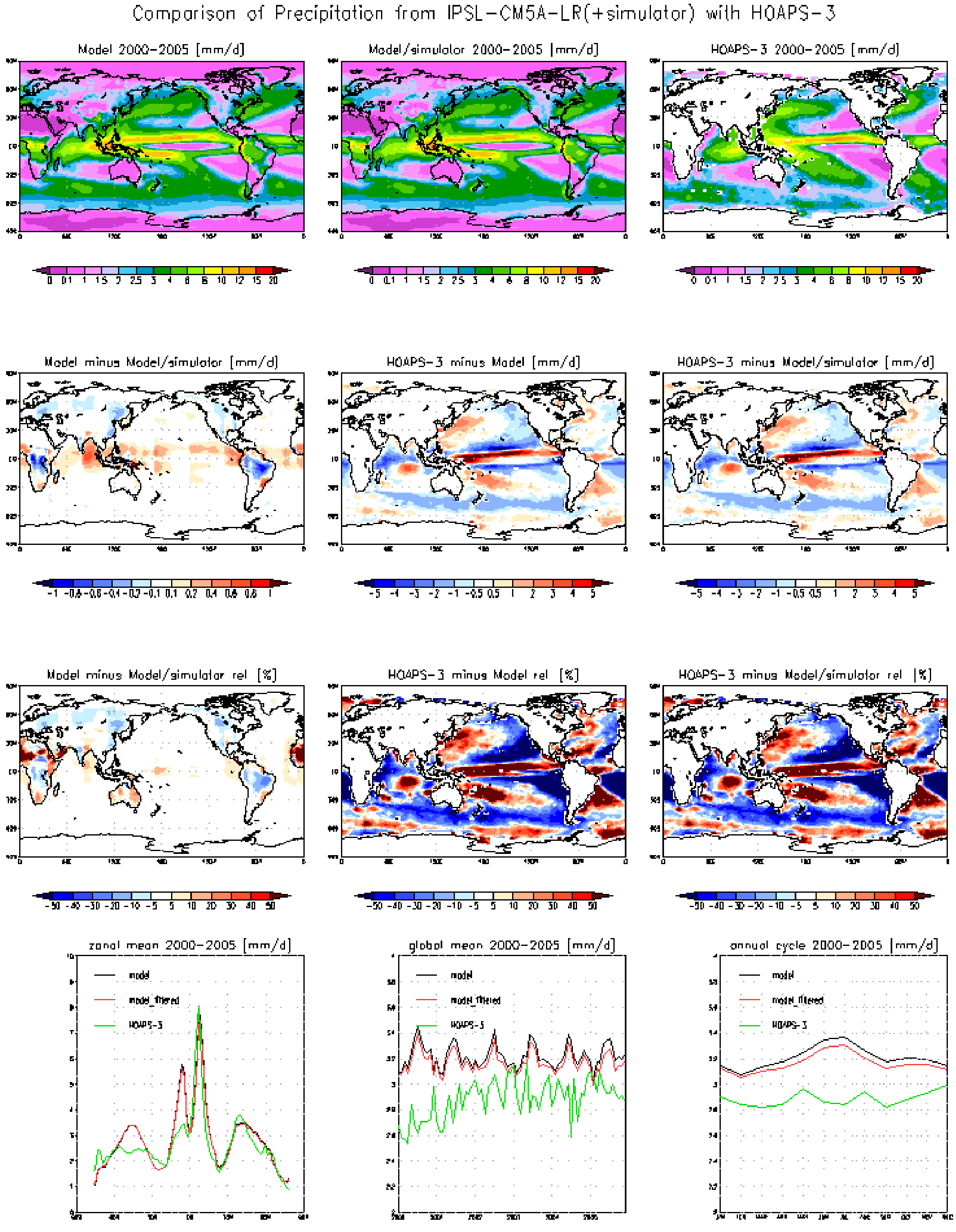



**Figure 3b**: Comparison of IPSL-CM5A-LR model (with and without HOAPS simulator) and HOAPS satellite derived 2000-2005 mean precipitation fields.

**MCMS**

The MCMS tool provides the basic parameters of each low-pressure system (LPS) in the observation data and in the simulation results for each time step, i.e. coordinates of the LPS-centers, the area of influence considered as the low pressure bowl around the LPS-center and some parameters concerning the LPS, namely the perimeter, min, max and mean distance between the LPS-center and the perimeter, area in km2 of the area of influence, its intensity and its depth. The MCMS web page provides the documentation and methods for acquiring and running the source code. The software structure and the main running steps are summarized in Appendix A.4. A Python routine that uses the IPSL model as an example of how to setup MCMS for a new model is also provided in section A4. First tests on the Sea Level Pressure (SLP) fields from the IPSL-CM5A-MR historical run showed the need for further MCMS development, i.e., for a universal procedure that gives acceptable results no matter the model. This was performed, thanks to a collaboration with US, by the main MCMS developer, Mike Bauer (NASA/GISS, Columbia University) at the very end of the project. Because of lack of time and further setting needs, he has also been responsible for testing the new method on the IPSL ESM model outputs and NCEP-2 re-analysis data. Figure 4a and 4b show the 1990-2005 raw center counts from the “Center Finder” (CF) and the more refined “Track Finder” (TF), respectively. Due to the difference in grid size, the counts are not the same even if the model and re-analysis have the same climatology. Model discrepancies appear near high topography and the North pole, which might relate either to the way the model calculated SLP from surface pressure or to the MCMS interpolation method. Work is currently done to try to understand and resolve it. In the last statistical analysis, the “Cyclone Density” (Figure 4c), the representation is independent of the resolution of the data and the time step. It shows the occupation of an area of $10^6$ $km^2$ with a cyclone (in %). Hence a value of 10 denotes a chance of 10% to find a cyclone within this area at any time (the patterns are smoothed to 1000 km to represent means in $10^6$ $km^2$).

Review Table n° 4: MCMS

| | Comments | Recommend |
|---|---|---|
| **Download/Installation** | Performed by the MCMS provider. | Use of the Mercurial repository |
| **Setting and running** | Performed by the MCMS provider. | Use of the NCO operators to make the input model files conform with MCMS's expectations |
| **Analysis/outputs** | | |
| **Figure 4a:** CF center counts | Ok | |
| **Figure 4b:** TF center count | Ok | |
| **Figure 4c:** Center Density | Ok | |

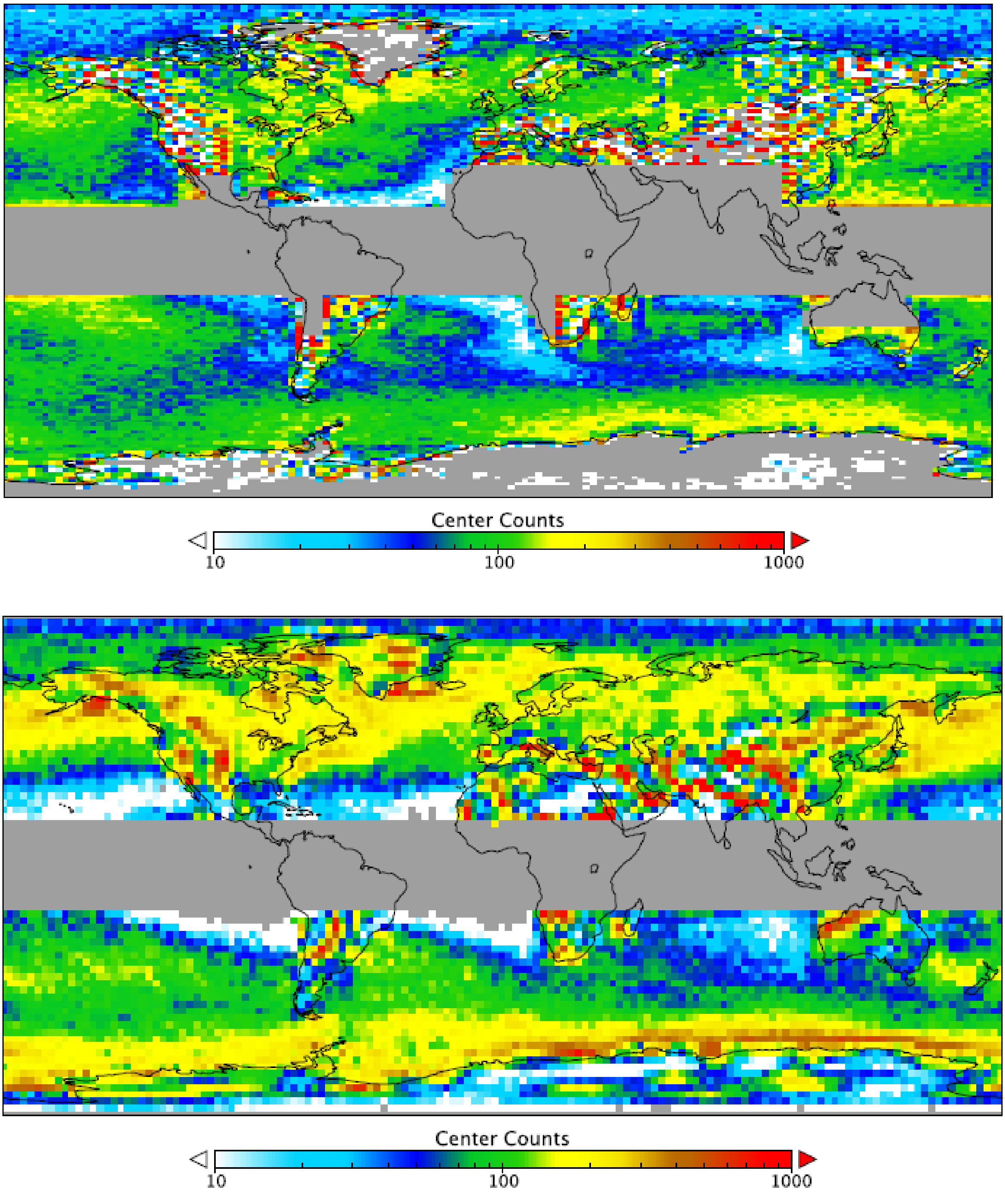



**Figure 4a**: Comparison of IPSL-CM5A-MR model (top) and NCEP Reanalysis 2 (bottom) 1990-2005 raw center counts from the "Track Finder" MCMS analysis.

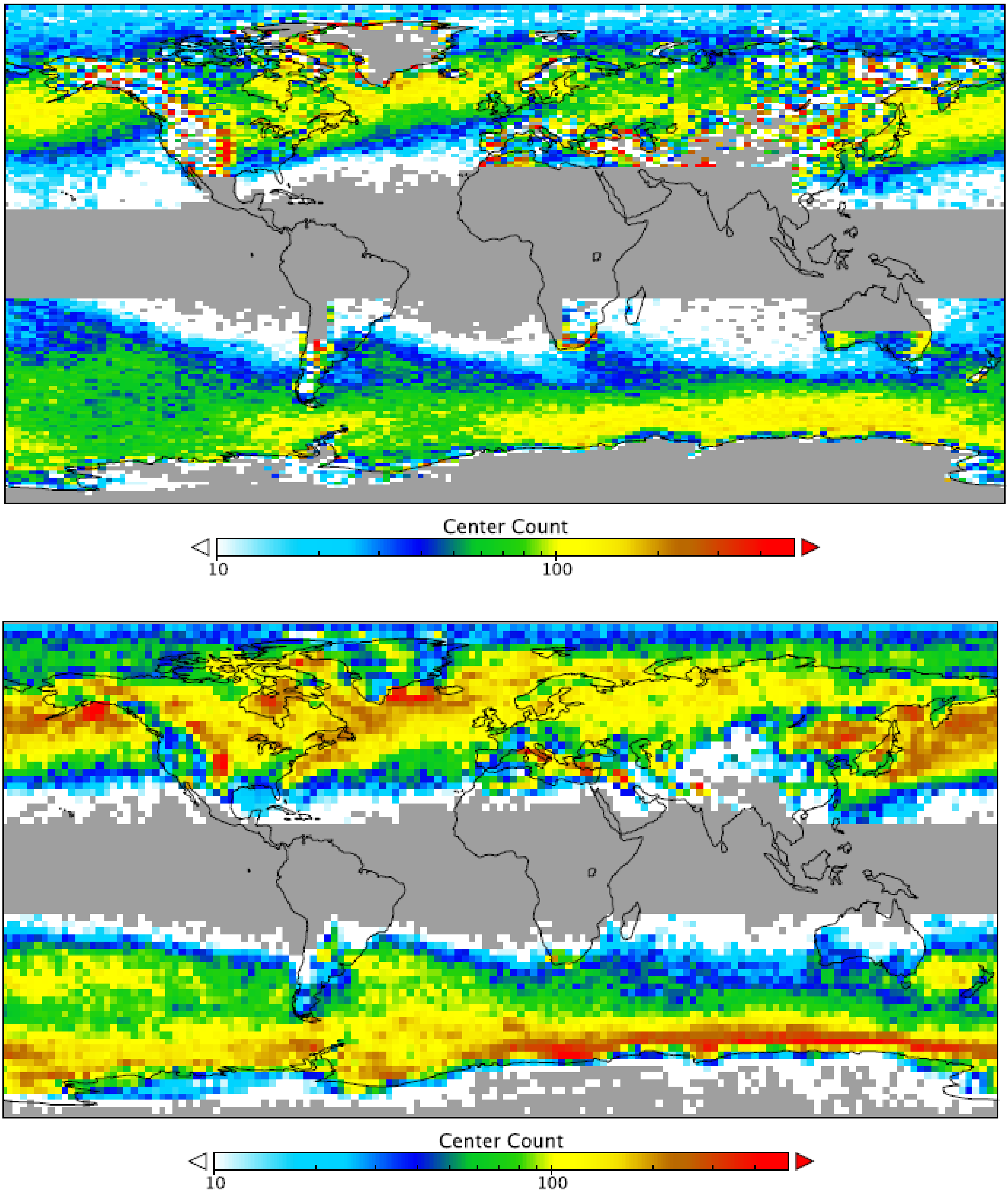



**Figure 4b**: Comparison of IPSL-CM5A-MR model (top) and NCEP Reanalysis 2 (bottom) 1990-2005 raw center counts from the "Center Finder" MCMS analysis.

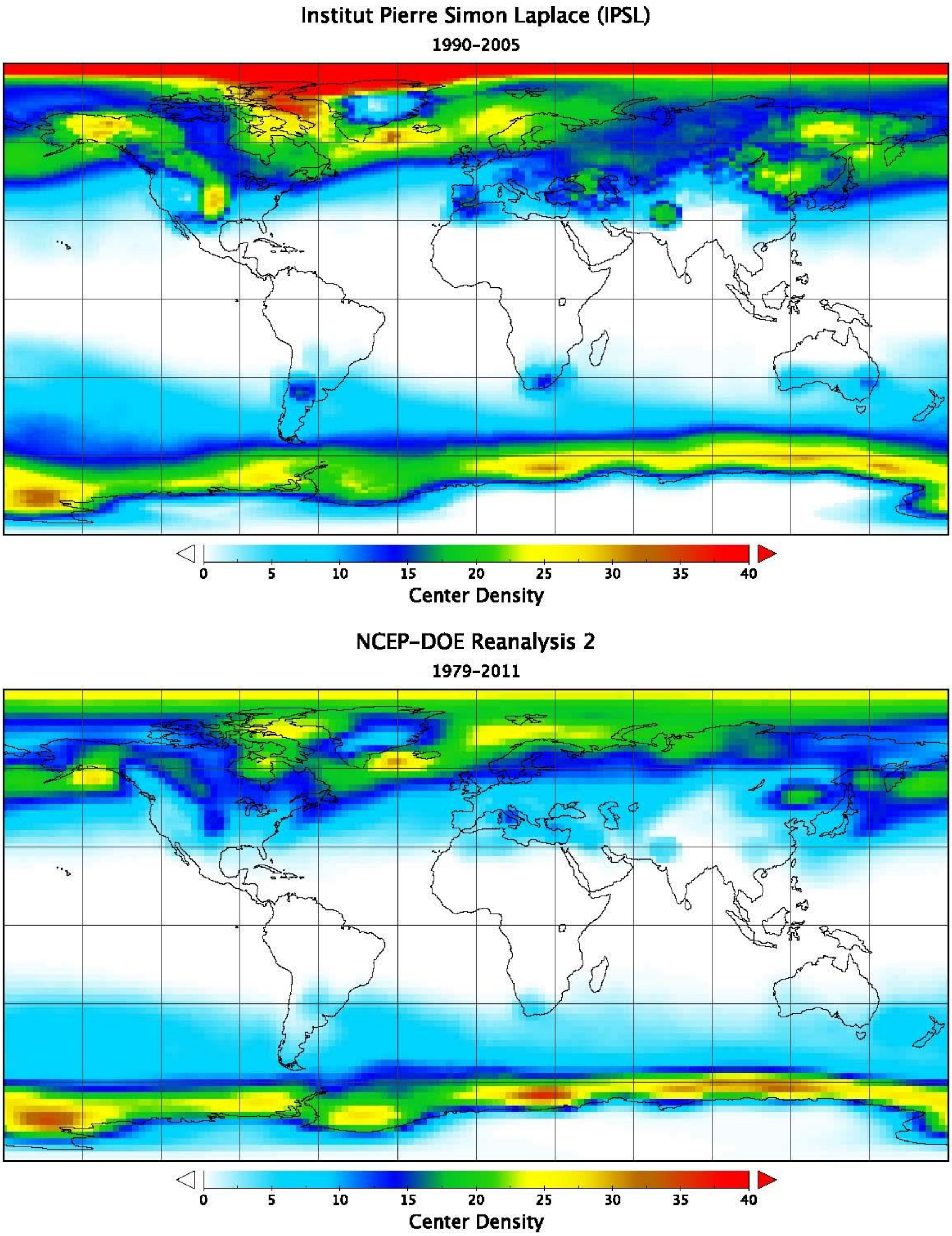



**Figure 4c**: Comparison of IPSL-CM5A-MR model (1990-2005) and NCEP DOE Reanalysis 2 (1979-2011) MCMS Center Density (frequency of occurrence per degree latitude squared).

# 3. REFERENCES


Andersson, A., C. Klepp, K. Fennig, S. Bakan, H. Graßl, and J. Schulz, 2011: Evaluation of HOAPS-3 ocean surface freshwater flux components, Journal of Applied Meteorology and Climatology, 50, 379-398, doi:10.1175/2010JAMC2341.1.

Andersson, A., S. Bakan, 2013: Evaluation of MPI-ESM Ocean Surface Fluxes, Journal of Advances in Modeling Earth Systems (JAMES), in preparation.

Bauer, M. and Del Genio, A. D., 2006: Composite analysis of winter cyclones in a GCM: influence on climatological humidity, J. Climate, 19, 1652–1672.

Bauer M, Tselioudis G, Rossow WB, 2013, A new climatology for investigating storminess influences on the extratropics. J Climate, accepted.

Cionni, I., Eyring, V., Lamarque, J. F., Randel, W. J., Stevenson, D. S., Wu, F., Bodeker, G. E., Shepherd, T. G., Shindell, D. T., and Waugh, D. W., 2011: Ozone database in support of CMIP5 simulations: results and corresponding radiative forcing, Atmos. Chem. Phys., 11, 11267-11292, doi:10.5194/acp-11-11267-2011.

Eyring, V., Butchart, N., Waugh, D. W., Akiyoshi, H., Austin, J.,Bekki, S., Bodeker, G. E., Boville, B. A., Br¨uhl, C., Chipperfield, M. P., Cordero, E., Dameris, M., Deushi, M., Fioletov, V.E., Frith, S. M., Garcia, R. R., Gettelman, A., Giorgetta, M. A.,Grewe, V., Jourdain, L., Kinnison, D. E., Mancini, E., Manzini,E., Marchand, M., Marsh, D. R., Nagashima, T., Newman, P. A., Nielsen, J. E., Pawson, S., Pitari, G., Plummer, D. A., Rozanov,E., Schraner, M., Shepherd, T. G., Shibata, K., Stolarski, R. S., Struthers, H., Tian, W., and Yoshiki, M., 2006: Assessment of temperature,trace species, and ozone in chemistry-climate model simulations of the recent past, J. Geophys. Res., 111, D22308, doi:10.1029/2006JD007327.

Gettelman, A., Eyring, V., Fischer, C., Shiona, H., Cionni, I., Neish, M., Morgenstern, O., Wood, S. W., and Li, Z., 2012: A community diagnostic tool for Chemistry Climate Model Validation, Geosci. Model Dev. Discuss., 5, 1229-1261, doi:10.5194/gmdd-5-1229-2012.

Koffi, B., M. Schulz, F.-M., Bréon, J. Griesfeller, D. Winker, Y. Balkanski, S. Bauer, T. Berntsen, M. Chin, W. D. Collins, F. Dentener, T. Diehl, R. Easter, S. Ghan, P. Ginoux, S.Gong, L. W. Horowitz, T. Iversen, A. Kirkevåg, D. Koch, M. Krol, G. Myhre, P. Stier, T.Takemura, 2012, Application of the CALIOP layer product to evaluate the vertical distribution of aerosols estimated by global models: AeroCom phase I results, J. Geophys. Res., 117, D10201, doi: 10.1029/2011JD016858.

Myhre, G., Samset, B. H., Schulz, M., Balkanski, Y., Bauer, S., Berntsen, T. K., Bian, H., Bellouin, N., Chin, M., Diehl, T., Easter, R. C., Feichter, J., Ghan, S. J., Hauglustaine, D., Iversen, T., Kinne, S., Kirkevåg, A., Lamarque, J.-F., Lin, G., Liu, X., Lund, M. T., Luo, G., Ma, X., van Noije, T., Penner, J. E., Rasch, P. J., Ruiz, A., Seland, Ø., Skeie, R. B., Stier, P., Takemura, T., Tsigaridis, K., Wang, P., Wang, Z., Xu, L., Yu, H., Yu, F., Yoon, J.-H., Zhang, K., Zhang, H., and Zhou, C., 2013, Radiative forcing of the direct aerosol effect from AeroCom Phase II simulations, Atmos. Chem. Phys., 13, 1853-1877, doi:10.5194/acp-13-1853-2013.

Schulz, M., Chin, M., Kinne S., 2009, The Aerosol Model Comparison Project, AeroCom, Phase II: Clearing Up Diversity**,** IGAC Newsletter, No 41, May.

# APPENDIX A : TOOLS' TECHNICAL DESCRIPTION

This Appendix provides a detailed description of the structure and of one application of each tested tool. See section 2 and IS-ENES evaluation portal for a general description of the selected tools.

## A.1 Technical description of AeroCom

The AeroCom tool reads standard (netCDF CF) model outputs and station (e.g., AERONET, AEROCE) or satellite (e.g.,MODIS, CALIOP ) observational datasets. It interpolates, calculates medians and writes various statistics. It also generates on-line plots (gif or png) of maps and graphs for an automatic comparison between observations and model outputs (see D.3). Evaluation results are then provided to the modellers through the graphical evaluation environment of the AeroCom website.

The testing of the tool consisted in evaluation the LMDz-INCA component of the IPSL ESM model in its ability in reproducing the aerosol optical depth and vertical distribution over Europe. Mean Aerosol Optical Depth (AOD) and extinction profiles were compared to the CALIOP-derived global aerosol profile product [Koffi *et al.*, 2012] for the years 2006 and 2007. The CALIOP dataset was developed and included to the AeroCom server in the frame WP9 Task 3of IS-ENES (see D9.3). Therefore, the technical details (directories, set up and running processes, outputs, ) provided hereafter do not intend to be exhaustive but more specifically related to these diagnostics.

### A.1.1 DIRECTORIES

| | |
|---|---|
| aerocomIDL | Running directory that contains all aerocom_* IDL routines |
| aerocomIDL/modellists | Contains files with lists of models (c_Models) and years (c_Years), as well as observation years (c_ObsYears). |
| aerocomIDL/include | Contains mic_include_[parameter].pro files = parameter to be analyzed. |
| aerocomIDL/plots | Default directory for output plots |
| aerocomIDL/data | Default directory for output data files |

### A.1.2 SCRIPTS AND MAIN ROUTINES

| | |
|---|---|
| StartBatch.job | Small script where are defined the modellists/*.txt (input) and include/mic_include_[parameter].pro (output) files and that calls the main running job (StartBatchParam.job). |
| StartBatchParam.job | Script that starts AeroCom IDL tool |
| aerocom_main.pro | Main IDL routine called by StartBatchParam.job. |
| aerocom_*.pro | Other IDL routines called by aerocom_main.pro |

### A.1.3 SOFTWARE NEEDED TO RUN THE SCRIPTS

An IDL license is required.

### A.1.4 RUNNING THE SCRIPTS

No tutorial exists. The main information is provided as comment lines within the scripts and routines.

The set up and running processes consist in four main steps:

1. Define the model runs an years to process: modellists/*.txt file
2. Define the parameter and the output options: include/mic_include_[parameter].pro file
3. Edit of the master "StartBatch.job"
   - write the name of the *.txt file in Line 4 (without '.txt' extension)
   - write the name of  the include/mic_include_*.pro file in Line 8 (without '.pro' extension)
4. Run of the aerocom tool

   *./ StartBatch.job (or via "qsub" batch options)*

The modellists/*.txt , include/mic_include_[parameter].pro and StartBatch.job files used in this testing study are provided below. Blu italic comments have been added to original files in order to clarify the settings' meaning and options.

**aerocomIDL/StartBatch.job**

```
#!/bin/ksh

# List of possible model sets (edited in ./modellists); with model years
set -A mlist LSCE&caliop.txt


# List of chosen parameters, refering to include directory and mic_include_[parameter].pro files
set -A ilist EC5503D_AER

for il in ${ilist[*]} ; do
file=./include/mic_include_${il}.pro
if  [ -f  $file ] ; then
./StartBatchParam.job $file ${mlist[*]}
fi
done
exit
```

Line 4 calls aerocomIDL/modellists/**LSCE_Caliop.txt** file with model runs and CALIOP years to be processed.  Line 8 calls aerocomIDL/include/**mic_include_EC5503D_AER.pro** file

**Input setting**

Satellite gridded data such as CALIOP and MODIS are processed as model outputs. Therefore c_ObsYears is set to '0000'.

***LSCE_Caliop.txt***

| |
|---|
| c_Models=('LSCE.A2.CTRL'  'CALIOP3') |
| c_Years=('2006' ‘2007’) |
| c_ObsYears=('0000') ; *In case of satellite observations, c_ObsYears is set to '0000'* |

**Output setting**

In this study, the 3D aerosol distribution (c_ModelVars=['EC5503D_AER']) is analyzed over a European domain [35°N-70°N; 10°W-70°E] and around Chernobyl (Ukraine) Nuclear Power Plant domain [50°N-52°N; 29°E-31°E] (c_MicStationFilters=['EUR', ‘CHERNO’]), but the same can be performed for any pre-defined region or latitudinal band. Regions’ name, shortname and coordinates are defined in *aerocomIDL/aerocom_plot_include.pro*.

***mic_include_EC5503D_AER.pro***

```
i_VerboseFlag=1
i_areaweightflag=1 ;
c_ModelDataType='M';
c_ModelVars=['EC5503D_AER']; variable(s) to be processed
i_ReadSubVarsFlag=1 ; 1 allows processing sub-variable(s)if needed
c_SubVars=strarr(n_elements(c_modelvars),1); sub-parameters
c_SubVars[0,0:0]=['Z3D']; Altitude subvariable is needed for ploting EC5503D_AER profiles
s_PlotTableFlag=['REGIONPROFILE']:
c_MicStationFilters=['EUR', 'CHERNO']; regions are defined in aerocom_plot_include.pro
i_PlotMapFlag=3; 1:monthly; 2:yearly; 3: both
i_MakePNG=1 ;
i_SendFlag=0 ;
i_ObsNetworktype=[0] ;
```

### A.1.5 OUTPUTS

Results of the above-described processing of EC5503D_AER diagnostic, as well as the results of OD550 diagnostic over Europe for the year 2007, are presented and discussed in section 3.1.

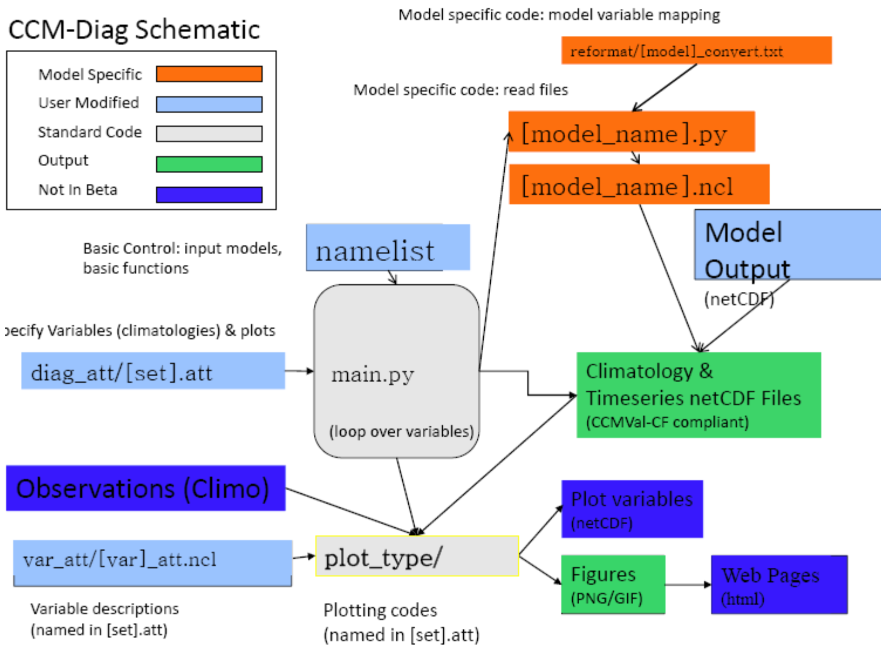
## A.2 Technical description of CCMVAL-DIAG

The CCM-diag code takes as input netCDF files either from (a) model output or (b) in CCMVal-2 data request format. If (a), it processes model output into (b). The code further creates climatology and timeseries files for the specified variables. Customization is required for each model to get output into the right format. This initial release comes with code for translating NCAR Community Climate System Model (CCSM) format netCDF files as a template. Each model needs its own piece of code to do this. The code reads *most* files in CCMval-2 format. The diagnostic code runs by running a main Python script (*main.py*). This script calls a control namelist (*namelist.py*) which specifies (a) global flags, (b) model output to process and (c) a file of diagnostic sets to run. The diagnostic set file is another namelist type file which lists each variable, and which 'diagnostics' to run on it. A diagnostic is specified as an ncl program. It can be as simple as a standard 'plot type' (zonal mean latitude height, lat-lon at a level, etc), or include a complex set of functions.

Variable names are either standard names from the CCMVal-2 CF specification, or 'derived' variables. Each variable name must have an attribute file. The variable attribute file is ncl code that sets parameters for plotting, as well as the function and other variables needed if it is a 'derived' variable. For now, observations can be simply treated as another 'model' to include in the set, and diagnostics and variables for each observation will have to be specified. The code is set up to read in netCDF files with either one time sample per file or multiple time samples per file.

### Diagnostic structure

• Main script (python, reads namelist)
• Loop for each variable identified
– Option: Convert to CF compliant CCMVal-2 netCDF
– Make climatologies (python)
– Loop for plots for each variable (NCL Code)
• Plotting
– Read and/or Calculate the variable if derived
– Loop for plots:
• based on namelists and standard plot types
• Create graphics and/or netCDF files




*Source: Gettelman et al. 2012*

The testing of the tool consisted in reproducing some of the plots of Eyring et al. [2006] paper ("E06" hereafter) but for the LMDZ-REPRODUS model only. The recent extension of the CCMVal-Diag code to the evaluation of the tropospheric ozone **[**Cionni et al., ACPD, 2011] has been also tested. The technical details here provided (directories, set up and running processes, outputs) are not exhaustive but related to these specific tests (user specific directories and files are highlighted in blue).

### A.2.1 DIRECTORIES AND MAIN FILES

| | |
|---|---|
| */home/koffi/CCMVAL/*Ver2.1/ | **Running directory** |
| CCMVal-Diag_README_20090826.txt<br>CCMVal-diag-Ver2.02_README_1.txt | Readme on CCMVal Diag Ver2.0 (2009).<br>Changes between Ver2.0. and Ver2.02 (2010) |
| namelist_*<br>*namelist_E06_CCMVal2* | Namelist files<br>*Namelist file to create Eyring et al., [2006] plots* |
| */home/koffi/CCMVAL/*Ver2.1/rgb/ | |
| */home/koffi/CCMVAL/*Ver2.1/reformat/ | |
| */home/koffi/CCMVAL/*Ver2.1/ncl_code/ | |
| */home/koffi/CCMVAL/*Ver2.1/plot_type | |
| */home/koffi/CCMVAL/*Ver2.1/diag_att/*.att<br>*E06Diag_bk.att* | diagnostic set files<br>*= to create Eyring et al., [2006] plots+ tropospheric ozone evaluation* |
| */home/koffi/CCMVAL/*Ver2.1/var_att/*_att.ncl | variable attribute files |
| | **Input and output directories** |
| */data/ user/CCMVAL/*Ver2.1/climo | |
| */data/ user/CCMVAL/*Ver2.1/REF-B1/ | Reference Runs: directory B1 |
| */data/ user/CCMVAL/*Ver2.1/REF-B1/ ERA40 | *Was empty* |
| */data/ user/CCMVAL/*Ver2.1/REF-B1/ *LMDZrepro* | *LMDZ-REPROBUS outputs in CMMval format (CCMVal2_REF-B1_LMDZrepro_*nc files) downloaded from BADC.* |
| */data/user/CCMVAL/*Ver2.1/plot_type/input_data/OBS | CCMVal files for testing E06FIG |
| */data/user/*CCMVAL/Ver2.1/plots_E06/ | *Directory with "E06" plots (created by running).* |

### A.2.2. SCRIPTS

| | |
|---|---|
| */home/koffi/CCMVAL/*Ver2.1/ | **Running directory that contains all the scripts** |
| main.py | main python script used to run the conversion and plotting codes |
| main.ncl | main ncl code |
| namelist.py | reads in a namelist file |
| diagatt.py | reads in a diagnostic attribute file |
| ccsm.py | model specific formatting code; concatenates a single variable from several CCSM h0 (monthly mean) files into one netcdf |
| cf_convert.py | calls a csh script to convert to CF 1.1 compliant netcdf file |
| climat.py | call ncl script that calculates climatology |
| ncquick.py | python interface for NetCDF |

| | |
|---|---|
| sanity_check.py | sanity checks for model files |
| var_required.py | handles required variables |

### A.2.3 SOFTWARE NEEDED TO RUN THE SCRIPTS:

- Python
- NCAR Command Language (NCL)

### A.2.4 RUNNING THE SCRIPTS

The set up process and running processes are described in details in the CCMVal-diag-Ver2.02_README_1.txt. The 'diag_att/E06Diag.att' namelist is used to produce Eyring et al., 2006 standard plots ("E06" hereafter). To control which plots are created, the variable name (CF or derived type) is linked to a data type ($H_2O$, O3, temperature, etc) and plot type (E06FIG01, E06FIG02, etc). CCMVal-diag application to "E06 diagnostics" and to tropospheric ozone evaluation consisted in the four following steps:

1. Edit of the master namelist "namelist-E06_CCMVal2" *(see page 28)*
   - ***Selection of options***
   - ***Specification of directories***
   - ***Selection of the model runs to process***

   The code specifies models with 6 arguments separated by spaces on a single line: Model name, Simulation (REF1), Ensemble ('1'), Start year, End Year, Directory to output.

   - ***Selection of the diagnostic set***

   This needs to be a file name in the 'diag_att' directory. The *E06Diag*.att* file (called E06Diag_bk.att in this test) is a plotting script functional on netCDF ccmval-1 output (available from the BADC), but not currently functional for new model output.

2. Edit of the diagnostic set 'diag_att\E06Diag_bk.att' *(see page 29)*

   Define variables and figures to process here. Ensure that all requested input files exist.

3. Ensure all variable attribute files exist in var_att directory

   Note that for non-derived fields, these files can be copied from existing files. The variable attribute files contain information to control the way the associated plots are generated.

4. Run the main.py with the correct master namelist.

   *python main.py namelist_E06_CCMVal2*

*namelist-E06_CCMVal2*

```
 plot_dir     $wrk_dir/plots_E06/
#-------------------------------------------------------------------
# specify the directory where the climo files are produced by hte main # python program
  climo_dir    $wrk_dir/climo/
#-------------------------------------------------------------------
# plot variables are written in CF netCDF format in addition to a file
# these *.nc files will be written into the directory that is created
# $plot_dir/E06FIGXX/*.nc
  write_plot_vars yes
#------------------------------------------------------------------
# In Version1 the path for the observations and the input file name is # specified in the attribute files, so please edit ./var_att/*_att.ncl.
# This will be improved for the next release.
#-------------------------------------------------------------------
# Models to use
MODELS
#
LMDZrepro  REF-B1  1  1980 1999 /data/koffi/CCMVAL/Ver2.1/REF-B1/LMDZrepro
# ERA40 for E06FIG01 and E06FIG02
ERA40       REF-B1     1  1980 1999  /data/koffi/CCMVAL/Ver2.1/REF-B1/ERA40
######################################################
# Diagnostics to do
######################################################
DIAGNOSTICS
  # specify the namelist of the diagnostics
  # the program assumes this namelist is in ./diag_att and has the #extention '.att' e.g. './diag_att/E06Diag.att' if specified as below
  E06Diag_bk
```

***diag_att/E06Diag_bk.att***

```
#########################################################################
# 'diag_att\E06Diag.att' is a namelist that selects the E06 plots
# that are produced by the tool. To control which plots are created, # the variable name (CF or derived type) is  linked to a data type
# (ta, ua, H2O, ect) and plot type (E06FIG01, E06FIG02, etc).                    #
# For the standard output plots, see
# Eyring et al., Assessment of temperature, trace species and ozone
# in chemistry-climate model simulations of the recent past, J.
# Geophys. Res., 111, D22308, doi:10.1029/2006JD007327, 2006.                #(hereinafter "E06")
##########################################################################
# Authors and Contact:
# Irene Cionni (irene.cionni@dlr.de) and Veronika Eyring
# (veronika.eyring@dlr.de)#
##########################################################################
# The namelist is part of the CCMVal diagnostic tool
# (PIs: Andrew Gettelman & Veronika Eyring)
# THIS NAMELIST IS CALLED BY main.py
#####################################################################
# Figures below that start with # are not produced; all others are (tested)
########################################################################
ta  T2Mz  E06FIG01  #OK
ua  T2Mz  E06FIG02  # OK
ta  T2Mz  E06FIG04  # Fails
H2O  T2Mz  E06FIG05 # OK (=Fig.6 of Eyring et al. 2006)
ta  T2Mz  E06FIG07  # OK
H2O  T2Mz E06FIG07  # OK
H2O  T2Mz E06FIG08  # Plots but with anomaly
H2O  T2Mz E06FIG09  # OK
Cly  T2Mz  E06FIG12 # OK
Cly  T2Mz  E06FIG12B  # OK

#HCl  T2Mz E06FIG05# not tested
#CH4 T2Mz E06FIG05# not tested
#HCl  T2Mz  E06FIG05A# not tested
#O3 T2Mz  E06FIG05# not tested
#toz  T2Ms E06FIG14 # # not tested (input data not found)
#toz  T2Ms E06FIG15 # # not tested (input data not found)

# In addition to E06 diagnostics, the tropospheric ozone [see Cionni et al., 2011] was evaluated
O3 T3M Logan_ozonesondes # Plots but does not complete all plots
```

### A.2.5 OUTPUTS

Eyring et al. [2006] standard plots:

| Fig.n° | Short Name | Variable | Description |
|---|---|---|---|
| E06FIG01 | vertline | T | Line plot of vertical profile differences |
| E06FIG02 | windzero | U | zero wind line descent in pressure |
| E06FIG04 | linets | T | 1-D timeseries plot (like tsline) |
| E06FIG05a | vertval | $O_3$, $CH_4$, $H_2O$, HCl | zonal mean profile plot |
| E06FIG05b | meridval | $O_3$, $CH_4$, $H_2O$, HCl | zonal mean line plot |
| E06FIG07 | linemon | T, $H_2O$ | annual cycle line plot (like monline) |
| E06FIG08 | vertts | $H_2O$ | vertical profiles over time |
| E06FIG09 | vertamp | $H_2O$ | amplitude and phase lag in vertical |
| E06FIG12 | profilets | Cly | profile and timeseries line plots (2 panels) |
| E06FIG14 | surfann | Column $O_3$ | contour plot of 2D zonal mean over month |
| E06FIG15 | tsclimo | Column $O_3$ | combination of timeseries and climatology |

Results of the test of CCMVal-diag tool on Eyring et al. [2006] above highlighted standard plots, and the plotting of time-height sections of ozone at selected stations are discussed in section 3.2. See also Deliverable D9.3 and Gettelman *et al.* [2012] for other analyses performed by the CCMVal-diag tool.

## A.3 Technical description of HOAPS

The purpose of this package is to compare HOAPS satellite data to climate model output. Two simulators have been implemented in order to assess possible biases due to the satellite sampling (diurnal cycle effects) and due to specific properties of the satellite retrieval (limited retrievals in case of precipitation). The tool is applied offline to climate model output with a high temporal resolution (at least 3.hourly), which is needed in order to achieve meaningful results. The simulator generates two masks that can be applied independently to the model output.

The mask is used to filter the model output according to the satellite orbits (SSMIOverpassSim). While model output usually consists of continuous fields, satellite data often contains data-gaps due to uneven spatial and temporal sampling. This applies in particular to data from polar orbiting satellites. The simulator uses pre-calculated orbit files as input. These are currently available for the years 1998-2005.

The other filter is generated from the precipitation parameter and is used to mask the evaporation (latent heat), wind, and water vapour parameters in case of precipitation events. It is not possible to retrieve those parameters from the satellite when considerable precipitation occurs (~ more than 1 mm/h), due to the dominance of the precipitation signal in the retrieved radiances. This behaviour can be simulated with the precipitation filter, which can be either calculated from the total precipitation parameter or from separated input fields of large scale and convective precipitation (defined in the _File_Info.txt)

After generating the filter files from the precipitation model output, they can be applied to other variables (evaporation/latent heat flux, wind speed, water vapour) and a comparison with the corresponding data from the HOAPS data set is possible. The script filter_data.sh applies the masks generated with the simulator to the model output and calculates monthly mean values. A GrADS script is used to plot the results.

### A.3.1 DIRECTORIES

Example under /data/koffi/HOAPS/HOAPS_SIM_IPSL /HOAPS_SIM_IPSL/

| | |
|---|---|
| data/ | model input and generated simulator files |
| doc/ | Documentation including HOAPS_SIMULATOR_README.doc |
| grads/ | ancillary scripts for GrADS |
| hoaps/ | hoaps data for comparison (remapped to IPSL resolution) |
| idl/ | ancillary scripts for idl |
| plots/ | output directory for plots |
| ssmi-orbit-data/ | SSM/I orbit data (generated separately) |

### A.3.2 SCRIPTS

| | |
|---|---|
| extract_year.sh | Extract single year from input data set |
| idl_batch.bat* | IDL batch file to call HOAPS_SAT_SIM |
| HOAPS_SAT_SIM.pro | Main IDL routine<br>Ancillary files are in the directory idl/ |
| pr3hr_IPSL-CM5A-LR_File_Info.txt* | Input definitions for IPSL-LR Model output (example for precipitation) |
| pr3hr_IPSL-CM5A-MR_File_Info.txt* | Input definitions for IPSL-MR Model output<br>(example for precipitation) |
| filter_data.sh | Apply filters from the simulator to the data and plot results |
| plotH3SIM.gs | GrADS Plotting script (called from filter_data.sh)<br>Ancillary files are in the directory grads/ |
| regrid_hoaps.ncl* | NCL script to regrid HOAPS data to IPSL grid |

*provided page 33 (to be adapted according to the model grid)

### A.3.3 SOFTWARE NEEDED TO RUN THE SCRIPTS

- cdo (https://code.zmaw.de/projects/cdo)
- IDL
- GrADS (http://www.iges.org/grads/)

### A.3.4 RUNNING THE SCRIPTS

The included scripts are preconfigured to run for the timespan 2000-2005. To generate the simulator files and output plot, the following scripts have to be adapted to the desired input/output parameters. The first three steps generate the simulator masks and have to be run only once for all parameters. Step 4 has to be run for each parameter.

1. run
   extract_year.sh <input_file> <startyear> [<endyear>]
   to extract yearly files from input data (give range of years to extract multiple years.
   If more than one precipitation parameter is used as retrieval filter (e.g. convective and stratiform), they have to be merged into one input. Merging can be done with:
   cdo merge <ifiles> <ofile>
2. adapt <Model_parameter>_File_Info.txt for the IDL program
   ***see pr3hr_IPSL-CM5A-LR_File_Info.txt*** *(page 34)*
3. adapt and run idl script:
   idl idl_batch.bat
   ***see idl_batch.bat*** *(page 35)*

4. adapt and run the masking and plotting script (to be done for each compared parameter) filter_data.sh

### A.3.5 OUTPUTS

| Short Names | Variables | Description |
|---|---|---|
| pr3hr,.. | precipitation, .. | Climate mean: model |
| pr3hr,.. | precipitation, .. | Climate mean: model with the simulator applied |
| pr3hr,.. | precipitation, .. | Climate mean: HOAPS data |
| pr3hr,.. | precipitation, .. | Climate mean: Model – model with simulator |
| pr3hr,.. | precipitation, .. | Climate mean : HOAPS - model |
| pr3hr,.. | precipitation, .. | Climate mean : HOAPS – model with simulator |
| pr3hr,.. | precipitation, .. | Climate mean: Model – model with simulator (%) |
| pr3hr,.. | precipitation, .. | Climate mean: HOAPS – model (%) |
| pr3hr,.. | precipitation, .. | Climate mean: HOAPS – model with simulator (%) |
| pr3hr,.. | precipitation, .. | Zonal means (model, model with simulator, HOAPS) |
| pr3hr,.. | precipitation, .. | Global mean time series (model, model with simul., HOAPS) |
| pr3hr,.. | precipitation, .. | Global mean annual cycle (model, model with simul. HOAPS) |

*regrid_hoaps.ncl*

```
;*************************************************
; regrid_hoaps.ncl
;*************************************************
;
; regrid hoaps-g to IPSL LR and MR resolution
load "$NCARG_ROOT/lib/ncarg/nclscripts/csm/contributed.ncl"
load "/home/aanderss/nfshome/NCL/functions/HOAPS.ncl"
;*************************************************
begin

 hpararray=(/"rain","wind","evap","late","wvpa"/)

 IPSLfileLR=addfile("pr3hr_IPSL-CM5A-LR_200001010130-200512312230.nc","r")
 IPSLfileMR=addfile("pr_3hr_IPSL-CM5A-MR_historical_r1i1p1_199001010130-200512312230.nc","r")

 IPSLprecipLR=lonFlip(IPSLfileLR->pr(0,:,:))
 IPSLprecipMR= lonFlip(IPSLfileMR->pr(0,:,:))

 do i=0,dimsizes(hpararray)-1
   hpar=hpararray(i)

   hoapsfile=get_HOAPSG_fileList(hpar,"r30","3.0")
   hoapsdata=short2flt(hoapsfile[:]->$hpar$(0:227,::-1,:))

   hoaps_remapLR   = linint2_Wrap(hoapsdata&lon,hoapsdata&lat, hoapsdata, True , IPSLprecipLR&lon,
IPSLprecipLR&lat, 0)

   hoaps_remapMR   = linint2_Wrap(hoapsdata&lon,hoapsdata&lat, hoapsdata, True , IPSLprecipMR&lon,
IPSLprecipMR&lat, 0)

   system("/bin/rm -f hoaps-g.IPSL-?R.m01."+hpar+".nc")

   outLR = addfile("hoaps-g.IPSL-LR.m01."+hpar+".nc" ,"c")  ; open output netCDF file
   outMR = addfile("hoaps-g.IPSL-MR.m01."+hpar+".nc" ,"c")  ; open output netCDF file
   outLR->$hpar$ = hoaps_remapLR
   outMR->$hpar$ = hoaps_remapMR

   att_names = getvaratts(hoapsfile[0])         ; get CCM file's global attributes
   do iatt = 0,dimsizes(att_names)-1
     ;print("copy_fileatts: global attributes->" + att_names(iatt))
     outLR@$att_names(iatt)$ = hoapsfile[0]@$att_names(iatt)$    ; copy global attributes
     outMR@$att_names(iatt)$ = hoapsfile[0]@$att_names(iatt)$    ; copy global attributes
   end do
   outLR@Average_Map_Resolution_remapped = "IPSL-LR (remapped from R30)"
   outMR@Average_Map_Resolution_remapped = "IPSL-MR (remapped from R30)"
 end do

end
~
```

*pr3hr_IPSL-CM5A-LR_File_Info.txt*

```
#<File_Info>
Dataset=IPSL-LR
Institution=IPSL
# Dims: coordinate dimensions in input file
Lonvar=lon
Latvar=lat
Timevar=time
# Timerange DD-MM-YYYY
# information about the timerange for which the simulator should be run
StartDate=01-01-2000
EndDate=31-12-2005
FileTimeIncrement=year
# File pattern: give the pattern of the input file name
# Date keys: %YYYY %MM %DD
FilePattern=pr3hr_IPSL-CM5A-LR_200001010130-200512312230_%YYYY.nc
inputDataDir=./data/
######################
# FLux Simulator
######################
# Variable definitions for HOAPS flux simulator
# precipitation variables that are used for filtering (comma separated list)
# give netcdf variable name of precipitation variables that are to be used as filter
hoapsFluxSimRainvars=pr
# Types of precipitation for above vars. (comma separated list)
# Recognized keywords: convective,largescale,snow,total
hoapsFluxSimRainvartypes=total
# Filter thresholds for above precipitation vars (comma separated list)
# unit mm/hr
hoapsFluxSimThresh=1.
# Threshold scaling factor to comply threshold unit with model data
# e.g.: 3600 to convert mm/hr in kg/(m^2 s) (division by 3600)
hoapsFluxSimThreshScale=3600.
######################
# Orbit Simulator
######################
# input file pattern for SSMI overpasses
SSMIOverpassFilePattern=overpass_RipslLR_%YYYY.sav
# input directory for overpass files
SSMIOverpassBaseDir=ssmi-orbit-data
# Variable from which the the time coordinate
# should be used for the overpass Simulator
hoapsOverpassSimVar=pr
# time window in hours for the SSMI overpass simulator
# (typically the simulation time step)
SSMIaccumulationHrs=3
#</File_Info>
```

***idl_batch.bat***

```
; IDL batch file to run HOAPS simulator
; compile simulator
.comp HOAPS_SAT_SIM.pro

; run simulator
; give configuration file as parameter
HOAPS_SAT_SIM, FILE_DESCRIPTOR='pr3hr_IPSL-CM5A-MR_File_Info.txt'
HOAPS_SAT_SIM, FILE_DESCRIPTOR='pr3hr_IPSL-CM5A-LR_File_Info.txt'

exit
```

## A.4 Technical description of MCMS

The MAP Climatology of Mid-latitude Storminess (MCMS) is essentially an algorithm for the objective recognition of extra-tropical low-pressure systems [Bauer *et al*., 2013]. The information provided in this section is derived from the very well documented and recently updated **MCMS web page**. The following links detail the software dependences and requirements for reading and making MCMS datasets.

The MCMS software is kept in a distributed source control management tool (Mercurial) and the public repository is hosted on Bitbucket. To obtain the software for creating new MCMS datasets use (MCMS_HOME defined below):

```
$ cd /MCMS_HOME
$ hg clone https://mcmsproject@bitbucket.org/mcmsproject/mcms_make
```

or download the files directly with a browser here. The main advantage of using the Mercurial repository is the ability to pull fixes/updates and merge them with the code you've been working on, whereas use of the zip files requires a manual merge.

### A.4.1 DIRECTORIES AND MAIN FILES

MCMS version 1.1 consists of three directories: 1) A directory called mcms (MCMS_HOME) where the MCMS software is installed. 2) A directory called data (DATA_HOME) where the SLP data is stored (a separate subdirectory for each model) and 3) A directory called output (OUTPUT_HOME) where MCMS stores its output (a separate subdirectory for each model).

Example (application to *psl_6hrPlev_IPSL-CM5A-MR_historical_r1i1p1_1990010103-2005123121.nc* file):

| **MCMS_HOME** | | **Main directory (code base)** |
|---|---|---|
| /Users/mbauer/mcms | mcms_setup.py | Runs all the setup routines (MCMS preprocessing) |
| | mcms_attribute_finder.py | Applies a contouring algorithm |
| | mcms_center_finder.py | Extracts cyclone centers from a series of SLP fields |
| | mcms_refine_centers.py | Assigns an intensity to a cyclone |
| | mcms_track_finder.py | Extracts cyclone tracks |
| | /cfg | set of basic configuration files |
| | /util | |
| | /extra | |
| | /rf_files/data_root.py | Set the root path to the SLP files |
| | /rf_files/output_root.py. | Set the path where to store output |
| | /rf_files/rf_ipsl_*.py | Set up of IPSL SLP data |
| **DATA_HOME** | | |
| /Volumes/Scratch/data | | **Obs. and model SLP data** |
| | /era40 | ERA40 reanalysis |
| | /merrac | MERRA (reduced "C" resolution) |
| | /nra2 | NCEP-DOE Reanalysis 2 |

| | | |
|---|---|---|
| | /ipsl | IPSL model output |
| | / IPSL-CM5A-MR | /1990-2005 historical run |
| **OUTPUT_HOME** | | |
| /Volumes/SSD/output | | **MCMS output** |
| | /era40 | |
| | /era40_files | |
| | /merrac | |
| | /merrac_files | |
| | /nra2 | |
| | /nra2_files/ | |
| | nra2_cf_passed_1990-2005.pdf<br>nra2_tf_passed_1990-2005.pdf<br>nra2_circ_5_cden_annual_1979_2011_eq1.pdf | See Figure 4a<br>See Figure 4b<br>See Figure 4c |
| | | |
| | /ipsl_files/ | |
| | setup_logfile.txt | Check the setup here |
| | ipsl_cf_passed_1990-2005.pdf<br>ipsl_tf_ passed_1990-2005.pdf<br>ipsl_circ_5_cden_annual_1990_2005_eq1.pdf | See Figure 4a<br>See Figure 4b<br>See Figure 4c |

### A.4.2 SOFTWARE NEEDED TO RUN THE SCRIPTS

Python is necessary for the compilation and run of the scripts. The NetCDF library is also necessary to process input data in NetCDF format.

**Basic Requirements**

Python related:

- Python (version >= 2.6). Most os-x (Apple) and linux setups come with python preinstalled. Note that earlier versions (< 2.6) or the newest 3.x branch will cause problems with MCMS.
- SciPy (pronounced "Sigh Pie") python routines for mathematics, science, and engineering.
- Numpy a fundamental package for scientific computing with Python (usually installed with SciPy).
- python-dateutil extends the standard datetime module (usually installed with SciPy).
- netcdf4-python reads and writes netCDF 3 and 4 files.
- http://matplotlib.org and http://matplotlib.org/basemap/
- matplotlib a python 2D plotting library.
- Basemap a mapping toolkit for matplotlib.
- Cython allows python to be rewritten and compiled as C for speed.

Additional Software:

- netCDF4 self-describing, machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data.
- The netCDF Operators (NCO) command-line programs for manipulating netCDF files (optional).
- Generic Mapping Tools (GMT) for tools for manipulating geographic and Cartesian data sets (optional).
- Mercurial the source control management tool used work with the MCMS repository (optional, but recommended).

A.4.3 RUNNING THE SCRIPTS

As stipulated in introduction, the CMIP5 IPSL model output files generally required significant pre-processing work to be fed into the evaluation tools. In the case of MCMS, this has been done by the main developer, M. Bauer at NASA/GISS (Colombia University). Brief information and additional links on the files preparation and configuration is also provided below.

**Set up**

The MCMS software is designed to work with netCDF files. Sometimes these netCDF files differ from what MCMS expects; usually this involves attributes to key variables (MCMS expects COARDS type entries). The easiest solution is to use the NCO operators (netCDF Operators) to add/delete/modify the netcdf files to conform with MCMS's expectations (see more here).

MCMS uses a set of basic configuration files (in the subdirectory */MCMS_HOME/cfg*) which serve the whole project. If the model is among those included with the MCMS codebase you should still double check the settings and it if the model is not listed then you'll have to add new entries (see more). Then, the configuration of MCMS for the model/SLP source can be performed. This is done by running a setup program that reads the settings in /cfg and does a bunch of pre-calculations and stores them in special files that MCMS can read in as it works. To do this a resource file has first to be created for the setup routine.

Example:

- Create the resource file (note the -c "create" option and the _cf ending**):**
  $ cd /MCMS_HOME
  $ python mcms_setup.py -c rf_files/rf_ipsl_setup.py

(see more)

- Run the resource file (note the lack of the -c "create" option):
  $ cd /MCMS_HOME
  $ python mcms_setup.py rf_files/rf_ipsl_setup.py

The basic correctness of the set up can be checked by examining the log file located in **setup_logfile.txt under** /OUTPUT_HOME/*ipsl_files/*.

The “rf_ipsl_setup.py” routine is provided in pages 41-43 as an example of setup routine.

**Running**

1. Center finder
   The first step is to run the center finder (**mcms_center_finder.py**). This works in a time independent manner and simply locates candidate cyclone centers as SLP extrema and applies a series of filters to remove less likely cyclones (i.e., noise). As with mcms_setup.py this requires that you create and then run a resource file.

   Example:
   - Create the resource file (note the -c "create" option and the _cf ending**):**
     $ cd /MCMS_HOME
     $ python mcms_center_finder.py -c rf_files/rf_ipsl_cf.py

   (see more)

   - Run the resource file (note the lack of the -c "create" option):

$ cd /MCMS_HOME
$ python mcms_center_finder.py rf_files/rf_ipsl_cf.py

2. Cyclone tracks
After center finding, MCMS will try to organize these centers into time sequences. That is, cyclone tracks. This is done by running **mcms_track_finder.py**, which as before requires its own resource file.
Example:
- Create the resource file (note the -c "create" option and the _cf ending**):**
  $ cd /MCMS_HOME
  $ python mcms_track_finder.py -c rf_files/rf_ipsl_tf.py

(see more)
- Run the resource file (note the lack of the -c "create" option):
  $ cd /MCMS_HOME
  $ python mcms_track_finder.py rf_files/rf_ipsl_tf.py

3. Attribution (i.e., cyclone area analysis)
In the third step MCMS defines the size/area occupied by each cyclone by determining the largest set of close SLP contours around each center. This step, called attribution, is the most time consuming aspect of MCMS.

Example:
- Create the resource file (note the -c "create" option and the _cf ending**):**
  $ cd /MCMS_HOME
  $ python mcms_attribution_finder.py -c rf_files/rf_ipsl_af.py

(see more)
- Run the resource file (note the lack of the -c "create" option):
  $ cd /MCMS_HOME
  $ python mcms_attribution_finder.py rf_files/rf_ipsl_af.py

4. Center Density analysis
The MCMS calculates the occupation of an area of $10^6$ $km^2$ with a cyclone (in %).

### A.4.4 OUTPUTS

As previously mentioned (see introduction and section 3), the SLP fields from the IPSL ESM model were provided to the MCMS main developer (Mike Bauer, NASA/GISS, Colombia University) who performed himself the test. Results of the "Center finder", "Cyclone tracks" and "Center Density" analyses are shown and compared to NCEP Reanalysis 2 in section 3. Other examples of MCMS outputs can be also found on MCMS web page.

## *rf_ipsl_setup.py*

```
#!/usr/bin/env python -tt
"""
Default run_file template for mcms_setup.py

Input:
^^^^^
None

Output:
^^^^^^
Various python data structures.

Examples:
^^^^^^^^

Notes/Warnings:
^^^^^^^^^^^^^^

Author: Mike Bauer  <mbauer@giss.nasa.gov>

Log:
2011/01 MB - File created
"""

# Standard library imports
import sys
import os

# MCMS module imports: Need to have mcms in your PYTHONPATH environment variable!
from cfg import define_source
from cfg import define_figure
from cfg import define_dirs
from cfg import define_defs
from cfg import define_vars
from cfg import ensure_path

# ---------------------------------------------------------------------------
#                          Main Settings Set
# ---------------------------------------------------------------------------
verbose = 0

dotline = "%s" % ("-"*40)
msg = '\n%s\n%40s\n%s\n' % (dotline,"Main Settings Set",dotline)
if verbose: print msg

# ---------------------------------------------------------------------------
# Pick data source see files in /cfg
#   See cfg/mcms_source_base_defs.py
# ---------------------------------------------------------------------------
pick_source = define_source()
msg = pick_source.make_pick("ipsl")
if verbose: print msg
model = pick_source.selection
model_def = pick_source.selections[model][0]

# ---------------------------------------------------------------------------
# Full path to the root directory where pick specific output will be stored.
#   If multiple data sets under same model (say GCM runs) add a subdirectory
#   to the model root
# ---------------------------------------------------------------------------
#experiments = {"e12k" : "PSL-12k",
#           "e14k" : "PSL-14k"}
#blobs = {"e12k" : "cam_b30.12_2kaDVTn_T85.cam2.h1",
#         "e14k" : 'cam_b30.14_37kaDVT_T85.cam2.h1'}
#experiment = experiments['e14k']
#blob = blobs['e14k']
ensemble = ""
experiment = ""
blob = ""
if ensemble:
    subdir = "%s/%s/" % (experiment,ensemble)
else:
    if experiment:
        subdir = "%s/" % (experiment)
    else:
        subdir = ""
out_path = "/Users/mbauer/Scratchf/output/%s/%s/" % (model,subdir)
out_path = ensure_path(subdir,out_path,check_exists=0)

# Create needed directories if they don't exist.
#   See cfg/mcms_directories_base_defs.py
pick_dirs = define_dirs()
msg = pick_dirs.make_pick("standard")
dirs_set = pick_dirs.selection
need_dirs = ["%s%s" % (out_path,x) for x in dirs_set]
if subdir:
    op = out_path.replace("/%s" % subdir,"")
    need_dirs.insert(0,op)
for ndir in need_dirs:
    while ndir.find("//") != -1:
        ndir = ndir.replace("//","/")
    if not os.path.exists(ndir):
        try:
```

```
            os.mkdir(ndir)
            if verbose:
                print "\tMade: %s" % ndir
        except:
            sys.exit("ERROR: creating %s" % ndir)
if verbose: print "OUT_PATH: %s" % out_path

# ---------------------------------------------------------------------------
# Full path to the directory where the SLP data files are stored.
# ---------------------------------------------------------------------------
slp_path = "/Users/mbauer/Scratchf/data/%s/%s/" % (model,subdir)
slp_path = ensure_path(subdir,slp_path)
msg = "SLP_PATH: %s" % slp_path
if verbose: print msg

# ---------------------------------------------------------------------------
# Directory to be created for storing temporary pick specific files.
# ---------------------------------------------------------------------------
if subdir:
    shared_path = "/Users/mbauer/Scratchf/output/%s_files/" % model
else:
    shared_path = "%s_files/" % out_path[:-1]
shared_path = ensure_path(subdir,shared_path)
msg = "SHARED PATH: %s"
if verbose: print msg % shared_path

# -------------------------------------------------------------------------
# Define the definitions to be read in.
#
# Hows how to alter a parameter in defs w/out having to alter the file
# itself. Here I use setup_all as a flag to alter the defs so that all data
# that can be pre-calculated and saved are, rather than the default which is
# to read those data from file.
# -------------------------------------------------------------------------

# If set to 1, then topography information will not be used to determine
#  potentially troublesome locations for SLP dependent analysis. For
#  example, regions of very high or steep topography can result in erroneous
#  SLP values that either mimic or obscure cyclones. Generally, the
#  results are better with no_topo set to 0.
no_topo = 0

# If set to 1, then the land_sea mask is not used to separate results
#  that occur over land or ocean grids. This info is not required for
#  full analysis. Note: the topography field can be used for this in  a pinch.
no_mask = 0

# Select wavenumber for regional screens.
wavenumbers = [4.0,8.0,13.0,18.0,26.0,52.0]

# This value is used by center_finder_vX.py to screen for regional minima
#  status. Generally, using too low wavenumbers means overly screening
#  centers and too large values have little effect other than to
#  make the analysis run longer. Good rule of thumb is 8-26.
wavenumber = wavenumbers[4]

# Set the latitude equivalent *radius* in degrees for center/track density
# calculations.
c_size = 5


# Prevent centers from being found in at the pole (reduces a lot of problems).
#   The default is to screen centers poleward of -86 and 89. Change this
#   below if wanted (use degrees). Setting to -90,90 disables the screen.
polar_exclusion = [-86.0,89.0]
polar_exclusion = [-90.0,90.0]

# -------------------------------------------------------------------------
# Alter default behavior found in util/defs.py
#   See cfg/mcms_defs_base_defs.py
# -------------------------------------------------------------------------
pick_defs = define_defs()
msg = pick_defs.make_pick(model)
if verbose: print msg
defs_set = pick_defs.selection
defs_set.update({'tropical_boundary':15, "tropical_boundary_alt":30,
    "critical_radius":0.0,"wavenumber":wavenumber,"use_gcd":True})
# match RAIBLE et. all 2007
#defs_set = {'max_cyclone_speed': 42.0,'age_limit':72.0}

# -------------------------------------------------------------------------
# Names associated with the netcdf files for the source
#   See cfg/mcms_ncvars_base_defs.py
# -------------------------------------------------------------------------
source_vars = define_vars()
msg = source_vars.make_pick(model)
if verbose: print msg
nc_vars = source_vars.selection
# -------------------------------------------------------------------------
#                        Extra Settings Set
# -------------------------------------------------------------------------
msg = '\n%s\n%40s\n%s\n' % (dotline,"Extra Settings Set",dotline)
if verbose: print msg

# Save center finding plots (requires matplotlib, 2x memory footprint)
save_plot = 1
msg = 'save_plot = %d' % save_plot
if verbose: print msg
```

```
# ---------------------------------------------------------------------------
# What sort of figures (png faster, eps/pdf better quality)
#    See cfg/mcms_figure_base_defs.py    'pdf' and 'eps' much slower than 'png', but higher
quality
# ---------------------------------------------------------------------------
pick_figure = define_figure()
msg = pick_figure.make_pick("pdf")
#msg = pick_figure.make_pick("eps")
#msg = pick_figure.make_pick("png")
if verbose: print msg
fig_format  = pick_figure.selection

# Halt program on error or just warn?
exit_on_error = 1
msg = 'exit_on_error = %d' % exit_on_error
if verbose: print msg
print
```

# APPENDIX B : ACRONYMS

AEROCE: Aerosol Oceanic Chemistry Experiment
AeroCom: Aerosol Comparisons between Observations and Models
AERONET: AErosol RObotic NETwork
AOD: Aerosol Optical depth
ASCII: American Standard Code for Information Interchange
ARM: Atmospheric Radiation Measurement platform
AVHRR: Advanced Very High Resolution Radiometer
BADC: British Atmospheric Data Centre
CALIOP: Cloud Aerosol LIdar with Orthogonal Polarization
CCM: Chemistry-Climate Model
CCMVal: Chemistry-Climate Model Validation
CCMVal-Diag : Chemistry-Climate Model Validation Diagnostic
CDO: Climate Data Operators
CF: Climate and Forecast
$CH_4$ : Methane
CICLAD : Calcul Intensif pour le Climat, l'Atmosphère et la Dynamique
$Cl_y$: inorganic chlorin
CMIP: Coupled Model Intercomparison Project
CNRS: Centre national de la recherche scientifique
EARLINET: European Aerosol Research Lidar Network
EMEP: European Monitoring and Evaluation Program
ENES: European Network for the Earth System Modelling
ESM: Earth System Model
GAW: Global Atmosphere Watch
GCM: Global Climate Model
GrADS: Grid Analysis and Display System
HALOE : Halogen Occultation Experiment
$H_2O$: Water
HCl: hydrogen chloride
HOAPS: Hamburg Ocean Atmosphere Parameters and Fluxes from Satellite Data
IDL: Interactive Data Language,
IMPROVE : Interagency Monitoring of Protected Visual Environments
IPSL : Institut Paul Simon Laplace, Paris, France
IPSL-CM5A : IPSL Climate Model Version 5A
IPSL-CM5A-LR: IPSL IPSL-CM5A-Model Low Resolution version
IPSL-CM5A-MR: IPSL IPSL-CM5A-Model Medium Resolution version
IS-ENES: InfraStructure for the European Network for the Earth System Modelling
JRA3 : Joint Research Activity 3 (of IS-ENES project in this study)
INCA : Interaction Chimie – Aérosols (tropospheric chemistry module of LMDz-INCA model)
LMD : Laboratoire de Météorologie Dynamique (IPSL laboratory)
LMDz :LMD General Circulation model
LMDz-INCA : Tropospheric CCM of IPSL ESM
LMDz-REPROBUS : Stratospheric CCM of IPSL ESM
LPS: Low Pressure System
MCMS : MAP Climatology of Mid-latitude Storminess
MISR: Multi-angle Imaging SpectroRadiometer
MLS: Microwave Limb Sounder
MODIS: Moderate-Resolution Imaging Spectroradiometer
NCAR: National Center for Atmospheric Research (Boulder, U.S.A)
NCEP: National Center for Environmental Prediction
NCEP/DOE: National Center for Environmental Prediction/Department of Energy

NCL: NCAR Command Language
NetCDF: Network Common Data Form
NODC: National Oceanographic Data Center
$O_3$:Ozone
PARASOL: Polarisation et Anisotropie des Réflectances au sommet de l'Atmosphère, couplées avec un Satellite d'Observation emportant un Lidar (french satellite)
POLDER: Polarization and Directionality of the Earth's Reflectances (optical imager)
REPROBUS: Reactive Processes Ruling the Ozone Budget in the Stratosphere
RSMAS: Rosenstiel School of Marine and Atmospheric Science (Miami, U.S.A)
SLP: Sea Level Pressure
SPARC: Stratospheric Processes And their Role in Climate (World Climate Research Programme)
SSM/I: Special Sensor Microwave/Imager
SST: Sea Surface Temeperature
T: Temperature
TOMS: Total Ozone Mapping Spectrometer
U: East-West component of the wind
UKMO : United Kingdom Meteorological Office
UNEP : United Nations Environment Programme
WMO: World Meteorological Organization