Using CMOR for Aerocom

Tim de Paus

MPI Meteorology Hamburg, Germany, in strong co-operation with **CEA-CNRS** Gif-sur-Yvette, France

Goal

- Reformatting x-y-z grid model output
- Reprocessing output variables to CF-compliant standard format
- ***More general***: Creating a uniform format of output of different models

Example of Definition Problem

- Group 1:

Var1=A/b1+A/b2

- Group 2:

Var1=A/b1+A/b2-A/b3

- Inter-comparing var1 leads to differences already by definition

CMOR

- Developed by:

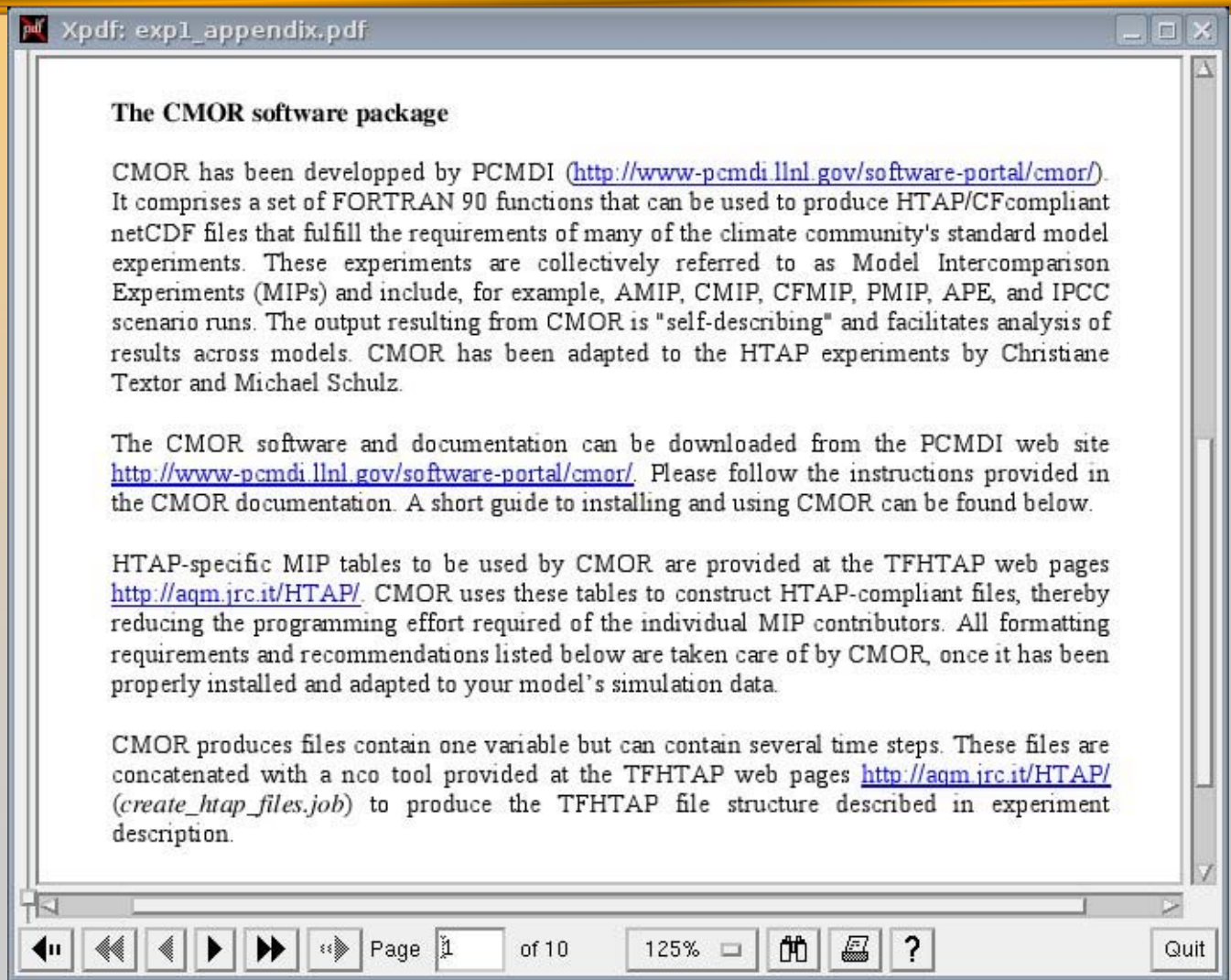
  Karl Taylor & Charles Doutriaux at PCMDI at Lawrence Livermore National Laboratory (LLNL) in Livermore, California and Jean-Yves Peterschmitt at LSCE France

- Adapted to HTAP by Christiane Textor/SA-CNRS-IPSL/France and Michael Schulz, Martin Schultz, Frank Dentener and Philip Tunis.

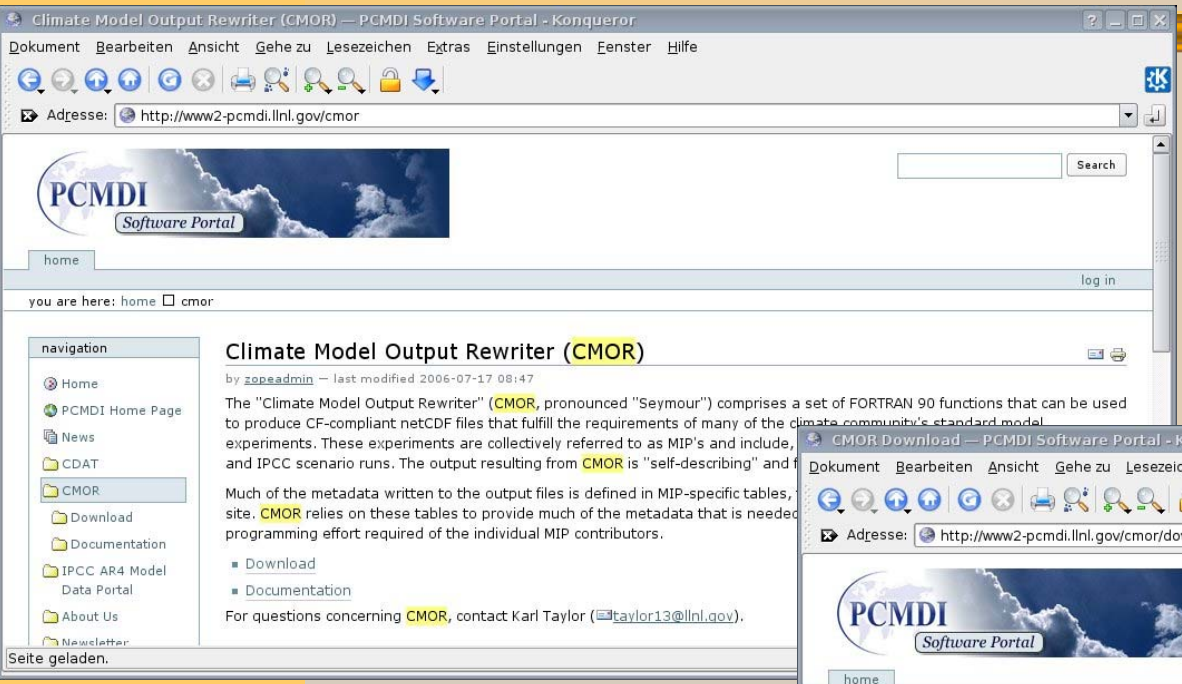
CMOR 2

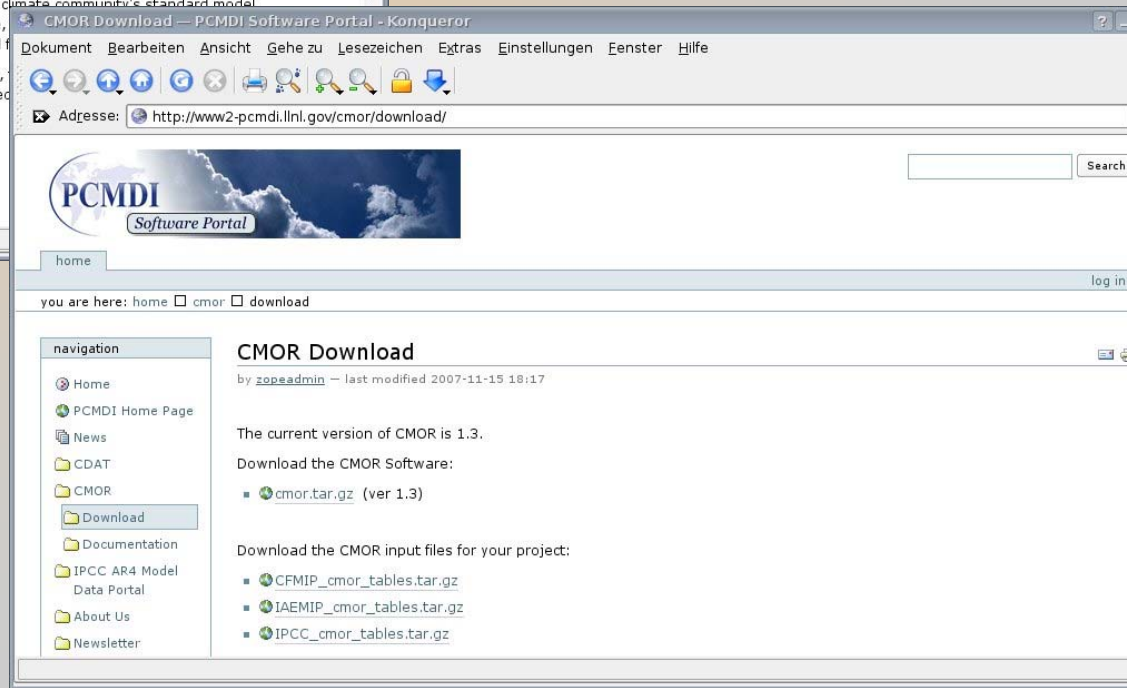
The "Climate Model Output Rewriter" (CMOR, pronounced "Seymour") comprises a set of FORTRAN 90 functions that can be used to produce CF-compliant netCDF files that fulfill the requirements of many of the climate community's standard model experiments

Where to get CMOR?



Website


Climate Model Output Rewriter (CMOR) — PCMDI Software Portal - Konqueror
Dokument Bearbeiten Ansicht Gehe zu Lesezeichen Extras Einstellungen Fenster Hilfe
Adresse: http://www2-pcmdi.llnl.gov/cmor
PCMDI Software Portal
Search
home
log in
you are here: home ☐ cmor
navigation
Home
PCMDI Home Page
News
CDAT
CMOR
Download
Documentation
IPCC AR4 Model Data Portal
About Us
Newsletter
Climate Model Output Rewriter (CMOR)
by zopeadmin — last modified 2006-07-17 08:47
The "Climate Model Output Rewriter" (CMOR, pronounced "Seymour") comprises a set of FORTRAN 90 functions that can be used to produce CF-compliant netCDF files that fulfill the requirements of many of the climate community's standard model experiments. These experiments are collectively referred to as MIP's and include, and IPCC scenario runs. The output resulting from CMOR is "self-describing" and
Much of the metadata written to the output files is defined in MIP-specific tables, site. CMOR relies on these tables to provide much of the metadata that is needed programming effort required of the individual MIP contributors.
Download
Documentation
For questions concerning CMOR, contact Karl Taylor (taylor13@llnl.gov).
Seite geladen.


CMOR Download — PCMDI Software Portal - Konqueror
Dokument Bearbeiten Ansicht Gehe zu Lesezeichen Extras Einstellungen Fenster Hilfe
Adresse: http://www2-pcmdi.llnl.gov/cmor/download/
PCMDI Software Portal
Search
home
log in
you are here: home ☐ cmor ☐ download
navigation
Home
PCMDI Home Page
News
CDAT
CMOR
Download
Documentation
IPCC AR4 Model Data Portal
About Us
Newsletter
CMOR Download
by zopeadmin — last modified 2007-11-15 18:17
The current version of CMOR is 1.3.
Download the CMOR Software:
cmor.tar.gz (ver 1.3)
Download the CMOR input files for your project:
CFMIP_cmor_tables.tar.gz
IAEMIP_cmor_tables.tar.gz
IPCC_cmor_tables.tar.gz

Structure of CMOR: F90

- CMOR functions
- Main program, including subroutines

CMOR functions

- cmor_getinput.f90
- cmor_internal_tables.f90
- cmor_users_functions.F90

These functions were created by Karl Taylor and co. and can be used without changing

Main Program and subroutines

You have to adapt these Fortran files for your own model

One file for each variable type:

- 2D-Monthly: A1b
- 3D-Monthly: A1c
- 2D-Daily: A1d
- 3D-Daily: A1f

Aerocom Tables

- One Aerocom table per file:
- Aerocom_table_2D-M
- Aerocom_table_3D-M
- Aerocom_table_2D-D
- Aerocom_table_3D-D

which can be found on:

http://www-lscedods.cea.fr/aerocom/CMOR/

Example Aerocom Table

Aerocom_table_3D-D (~/TabellenAerocomNet_toMakeMichael_23072008) - VIM

File Edit View Terminal Tabs Help

```
cmor_version: 1.0           ! version of CMOR that can read this table
cf_version:   1.0-AEROCOM   ! version of CF that output conforms to
project_id:   AEROCOM experiments! project id
table_id:     Table 3D-D      ! table id
table_date:   June 2008      ! date this table was constructed

expt_id_ok:   'AEROCOM'


magic_number: -1          ! used to check whether this file has been
                          !   altered from the official version.
                          !   should be set to number of non-blank
                          !   characters in file.
approx_interval:  1.      ! approximate spacing between successive time
                          !   samples (in units of the output time
                          !   coordinate), but this is ignored if set to 0.
missing_value: 1.e20      ! value used to indicate a missing value
                          !   in arrays output by netCDF as 32-bit IEEE
                          !   floating-point numbers (float or real)

!*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#
!
! SUBROUTINE ARGUMENT DEFAULT INFORMATION
!
!*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#*#
!
!  set default specifications for subroutine arguments to:
!     required/indeterminate/optional/ignored/forbidden
!    (indeterminate may or may not be required information, but is not always
!     required as an argument of the function call)
!
!
!============
subroutine_entry: cmor_axis
!============
!
required: table axis_name units length coord_vals cell_bounds
ignored: interval
!
!============
subroutine_entry: cmor_variable
!============
!
required: table table_entry units axis_ids
indeterminate: missing_value
optional: tolerance original_name history comment
ignored: positive
!
!============
subroutine_entry: cmor_write
!============
!
required:  var_id data
indeterminate: ntimes_passed time_vals time_bnds store_with
optional: file_suffix
```

Aerocom_table_3D-D (~/TabellenAerocomNet_toMakeMichael_23072008) - VIM

File Edit View Terminal Tabs Help

```
! Axis attributes:
!----------------------------------
standard_name:    atmosphere_hybrid_sigma_pressure_coordinate
units:            1
axis:             Z
positive:         down
long_name:        hybrid sigma pressure coordinate
!----------------------------------
!
! Additional axis information:
!----------------------------------
out_name:         lev
stored_direction: decreasing
valid_min:        0.0
valid_max:        1.0
formula:          p(n,k,j,i) = a(k)*p0 + b(k)*ps(n,j,i)
z_factors:        p0: p0 a: a b: b ps: ps
z_bounds_factors: p0: p0 a: a_bnds b: b_bnds ps: ps
!----------------------------------
!
!
!============
axis_entry:  alternate_hybrid_sigma
!============
!
!    Override default argument specifications for cmor_axis
!------------
optional: units
!------------
!
! Axis attributes:
!----------------------------------
standard_name:    atmosphere_hybrid_sigma_pressure_coordinate
units:            1
axis:             Z
positive:         down
long_name:        hybrid sigma pressure coordinate
!----------------------------------
!
! Additional axis information:
!----------------------------------
out_name:         lev
stored_direction: decreasing
valid_min:        0.0
valid_max:        1.0
formula:          p(n,k,j,i) = ap(k) + b(k)*ps(n,j,i)
convert_to:       standard_hybrid_sigma
z_factors:        p0: p0 ap: ap b: b ps: ps
z_bounds_factors: p0: p0 ap: ap_bnds b: b_bnds ps: ps
!----------------------------------
!
!
!============
axis_entry:  hybrid_height
!============
```

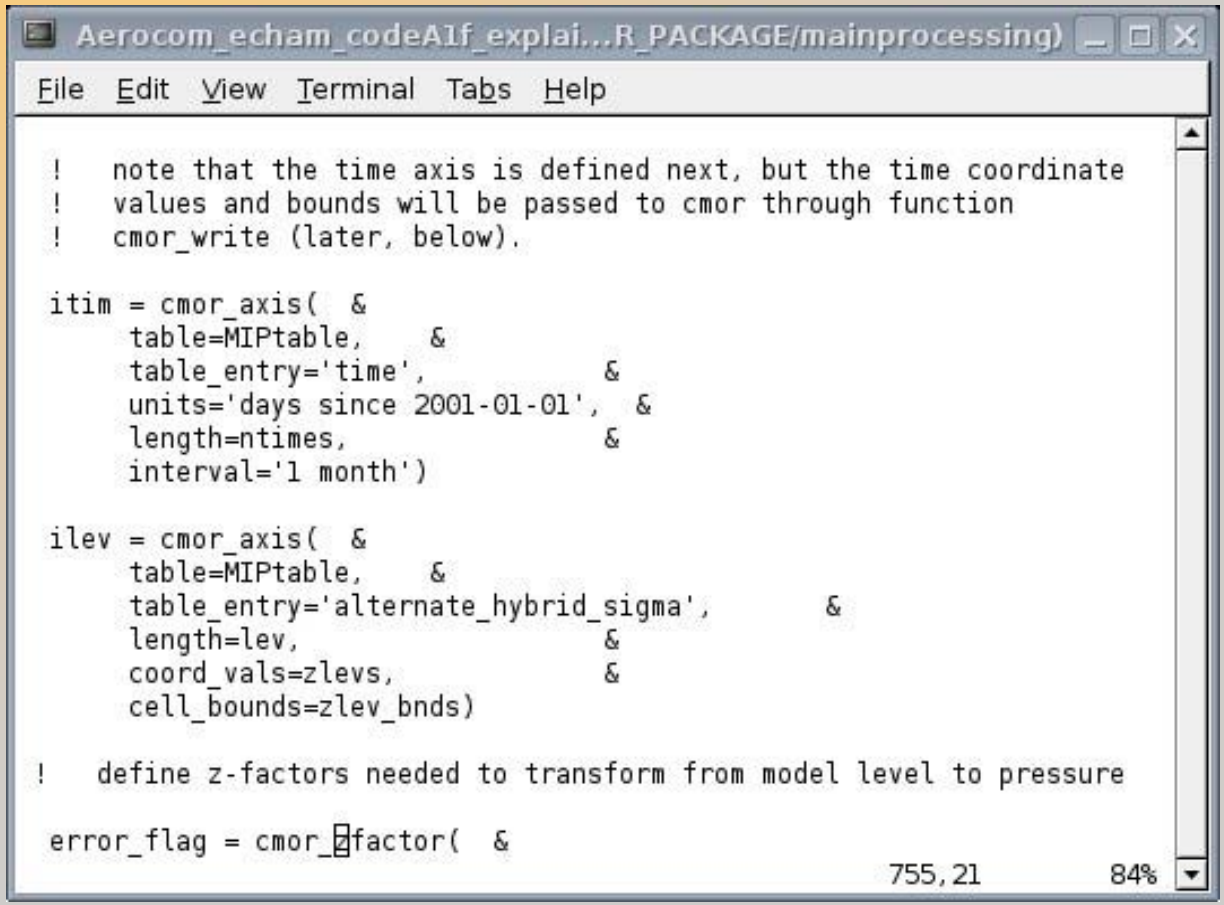
Example Aerocom Table II

Aerocom_table_3D-D (~/TabellenAerocomNet_toMakeMichael_23072008) - VIM

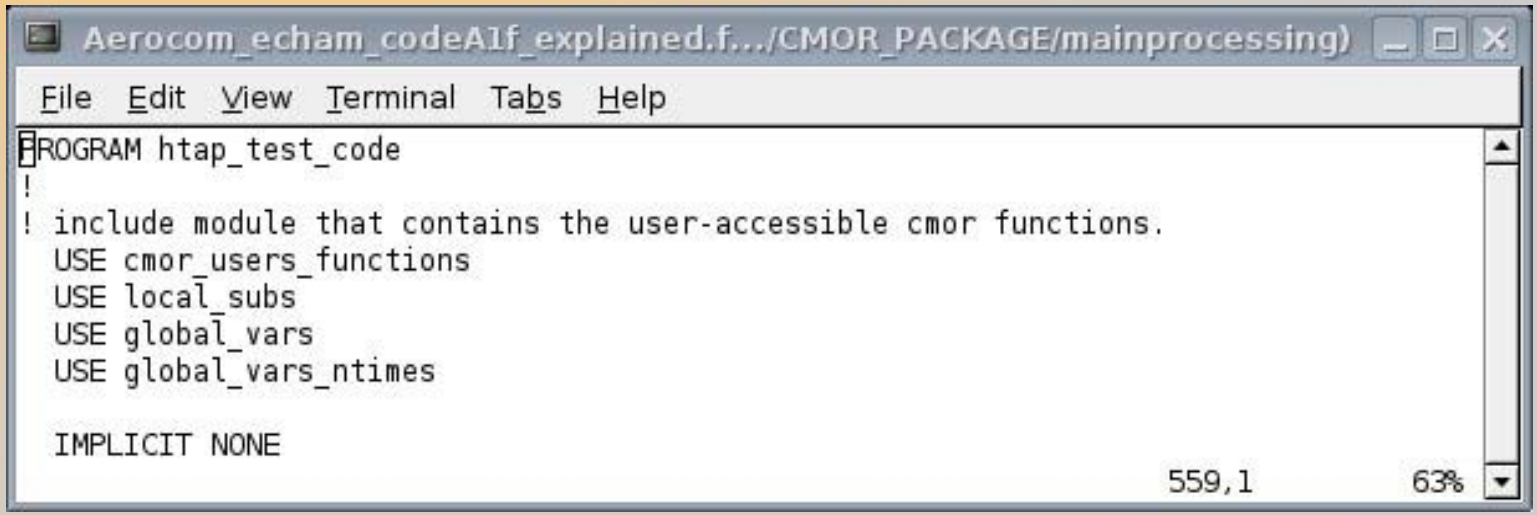
File Edit View Terminal Tabs Help

```
!----------------------------------
standard_name:   surface_air_pressure
units:           Pa
cell_methods:    time: mean
long_name:       Surface Pressure
!----------------------------------
!
! Additional variable information:
!----------------------------------
dimensions:       longitude latitude time
valid_min:        0.48e5
valid_max:        1.2e5
ok_min_mean_abs:  0.8e5
ok_max_mean_abs:  1.2e5
!----------------------------------
!
! ****************************************************************
!
! Aerocom Table 3D-D Daily-mean 3-d atmosphere data
!
! ****************************************************************
!
!
!
!============
variable_entry: clt3D
!============
!
! Variable attributes:
!----------------------------------
standard_name:  cloud_area_fraction
units:          1
cell_methods:   time: mean
long_name:      Cloud fraction
!----------------------------------
!
! Additional variable information:
!----------------------------------
dimensions:         longitude latitude zlevel time
valid_min:          0.0
valid_max:          400.0
!----------------------------------
!
!
!
!============
variable_entry: ec550_aer
!============
!
! Variable attributes:
!----------------------------------
standard_name:  atmosphere_extinction_due_to_ambient_aerosol
units:          m-1
cell_methods:   time: mean
long_name:      Aerosol extinction at 550nm
                                                        695,1         95%
```

Example CMOR code A1f-3D_D



Example code A1f Main Program



Example code A1f Main Program-II

Aerocom_echam_codeA1f_explained.f90 (~/CMOR_PACKAGE/mainprocessing) - VIM

File Edit View Terminal Tabs Help

```
  REAL, ALLOCATABLE :: ps(:,:,:), aps(:,:,:)

!!!==== USER NEEDS TO MODIFY HERE======!!!
!!!     replace by USER variable names as they will be used in read_3d_input_files routine.
!!!     Absent fields: replace variable name by 'DUMMY' place holder
!!!                    to avoid editing the subsequent fields

!! Here one has to define the variable names of your model corresponding to
!the model variable names as required by Aerocom (afterwards defined)
                                ! USER variable names for MIP Table fields
   CHARACTER (LEN=14), DIMENSION(n3d) :: &
      varin3d= (/ 'DUMMY         ','DUMMY           ','aclcac       ' /)

! Here the variable units are defined
                                ! USER Units appropriate to my data
! other udunits than given in MIP tables can be used here. If udunits package is available
! this is used to recalculate. This example assumes units are as in MIP table
   CHARACTER (LEN=12), DIMENSION(n3d) :: &
      units3d=   (/'m-1 ', ' sr-1 ', '1' /)

!! Here the variable names as required by Aerocom are defined. One should not
!change these entries.
                     ! Corresponding HTAP MIP Table entry (variable name)
  CHARACTER (LEN=14), DIMENSION(n3d) :: &
      entry3d= (/ 'ec550_aer'     ,'bs550_aer'           ,'clt3D' /)

!  uninitialized variables used in communicating with CMOR:
                                                                618,1         69%
```

Example code A1f Main Program - III

Aerocom_echam_codeA1f_explained...CMOR_PACKAGE/mainprocessing

File Edit View Terminal Tabs Help

```
!   define z-factors needed to transform from model level to pressure

 error_flag = cmor_zfactor(  &
      zaxis_id=ilev,                        &
      zfactor_name='p0',                    &
      units='Pa',                           &
      zfactor_values = p0)

 error_flag = cmor_zfactor(  &
      zaxis_id=ilev,                         &
      zfactor_name='ap',                      &
      axis_ids= (/ ilev /),                  &
      units='Pa',                            &
      zfactor_values = a_coeff,              &
      zfactor_bounds = a_coeff_bnds )

 error_flag = cmor_zfactor(  &
      zaxis_id=ilev,                         &
      zfactor_name='b',                      &
      axis_ids= (/ ilev /),                  &
      zfactor_values = b_coeff,              &
      zfactor_bounds = b_coeff_bnds  )

 zfactor_id = cmor_zfactor(  &
      zaxis_id=ilev,                           &
      zfactor_name='ps',                       &
      axis_ids=(/ ilon, ilat, itim /), &
      units='Pa'         )

  print * ,"After ps cmor_zfactor"

 !  Define variables appearing in HTAP MIPtable

 DO m=1,n3d
    if ( TRIM(ADJUSTL(varin3d(m))) .ne. 'DUMMY') then
       var3d_ids(m) = cmor_variable(    &
         table=MIPtable,  &
         table_entry=entry3d(m),     &
         units=units3d(m),           &
         axis_ids=(/ ilon, ilat, ilev, itim /), &
         missing_value=1.0e20,       &
         original_name=varin3d(m))
    endif
                                                          795,8         89%
```

Example code A1f Main Program - IV

Aerocom_echam_codeA1f_explained...CMOR_PACKAGE/mainprocessing

File Edit View Terminal Tabs Help

```
    DO m=1,n3d

       ! The user must write the code that fills the arrays of data
       ! that will be passed to CMOR.

       if ( TRIM(ADJUSTL(varin3d(m))) .ne. 'DUMMY') then
          call read_3d_input_files(file_in,it,varin3d(m), data3d)

          ! append a single time sample of data for a single field to
          ! the appropriate netCDF file.

          error_flag = cmor_write(                                    &
               var_id        = var3d_ids(m),                          &
               data          = data3d,                                &
               ntimes_passed = 1,                                     &
               time_vals     = time,                                  &
               time_bnds     = bnds_time  )

          error_flag = cmor_write(                                    &
               var_id        = zfactor_id,                            &
               data          = dataps,                                &
               ntimes_passed = 1,                                     &
               time_vals     = time,                                  &
               time_bnds     = bnds_time,                             &
               store_with    = var3d_ids(m) )


          IF (error_flag < 0) THEN
             ! write diagnostic messages to standard output device
             write(*,*) ' Error encountered writing HTAP ', MIPtable &
                  // 'field ', entry3d(m), ', which I call ', varin3d(m)
             write(*,*) ' Was processing time sample: ', time

          END IF
       endif

    END DO

END DO time_loop

!   Close all files opened by CMOR.

error_flag = cmor_close()
                                                          872,11        98%
```

Pre-, post- and processing

- Pre-processing:
- ksh scripts
- Main Processing
- tcl/tk stearing of cmor programs
- Post-processing
- tcl/tk scripts

Steps of processing

Terminal <7>

File Edit View Terminal Tabs Help

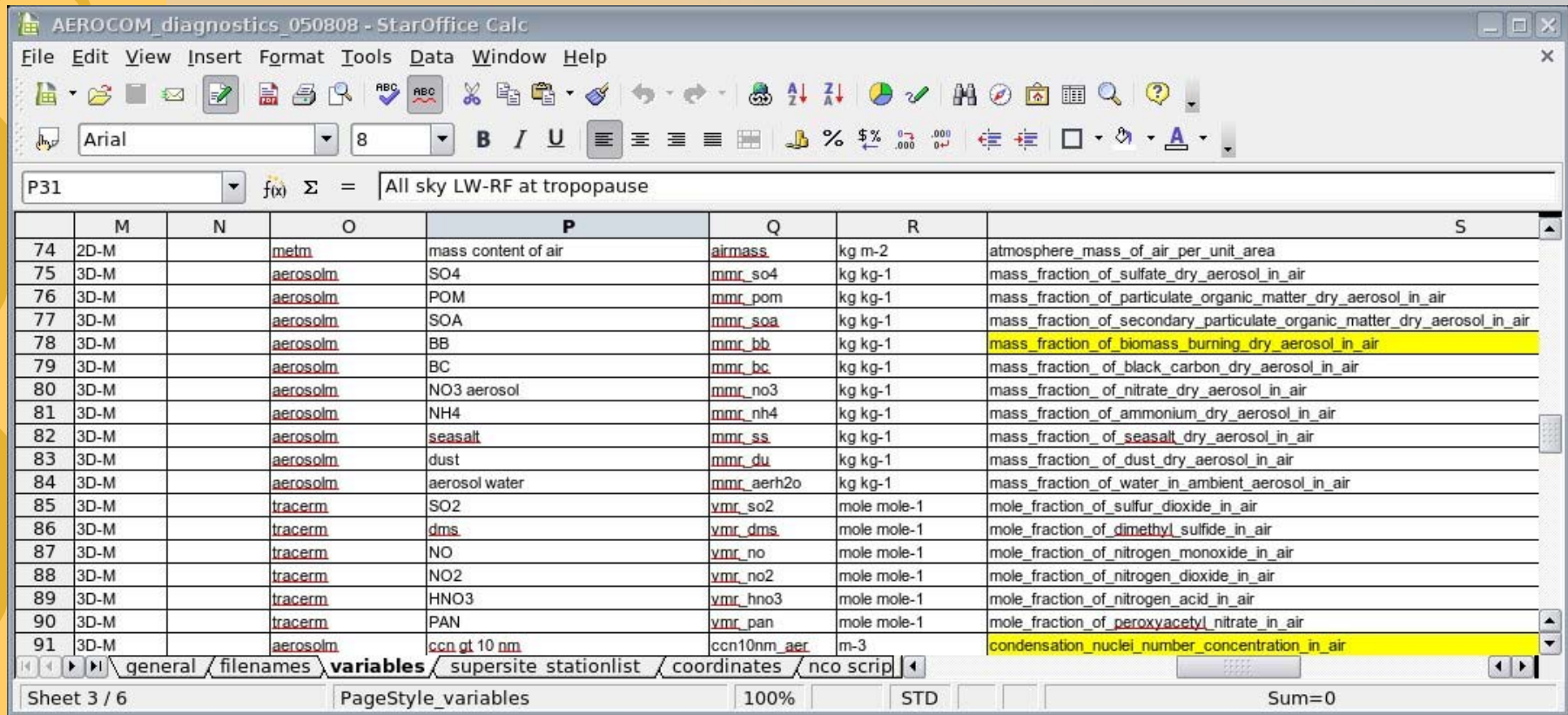
```
m222101@mpipc38:~> cd CMOR_PACKAGE/
m222101@mpipc38:~/CMOR_PACKAGE> ls
mainprocessing  postprocessing  preprocessing
m222101@mpipc38:~/CMOR_PACKAGE> cd preprocessing/
m222101@mpipc38:~/CMOR_PACKAGE/preprocessing> ls
multiplemonths_merge_echam_parameters_A1b.sh          multiplemonths_merge_echam_parameters_A1d.sh
multiplemonths_merge_echam_parameters_A1c.sh          multiplemonths_merge_echam_parameters_A1d_tryflex.sh
multiplemonths_merge_echam_parameters_A1c_preproc.sh  multiplemonths_merge_echam_parameters_A1f.sh
m222101@mpipc38:~/CMOR_PACKAGE/preprocessing> cd ../mainprocessing/
m222101@mpipc38:~/CMOR_PACKAGE/mainprocessing> ls
Aerocom_echam_codeA1b      Aerocom_echam_codeA1c.f90  Aerocom_table_A1b  HAM_200001.01_2d_d_daymean.nc  ToTarMain
Aerocom_echam_codeA1b.f90  Aerocom_echam_codeA1d      Aerocom_table_A1c  HAM_200001.01_2d_m_md.nc       ToTarMain.tar
Aerocom_echam_codeA1c      Aerocom_echam_codeA1d.f90  Aerocom_table_A1d  HAM_200001.01_all_merged.nc
m222101@mpipc38:~/CMOR_PACKAGE/mainprocessing> cd ../postprocessing/
m222101@mpipc38:~/CMOR_PACKAGE/postprocessing> ls
step1  step2  step3
m222101@mpipc38:~/CMOR_PACKAGE/postprocessing> ls *
step1:
run_cmor_over_months_splitdate_A1b_A1d.tcl  run_cmor_over_months_splitdate_A1d.tcl
run_cmor_over_months_splitdate_A1c.tcl      run_cmor_over_months_splitdate_A1f.tcl

step2:
merge_output_cmor_A1b_lists_A1bA1d.tcl  merge_output_cmor_A1d_lists.tcl

step3:
merge_monthlytoannual_output_cmor_A1bA1d.tcl  merge_vars_monthly_output_cmor_A1d.tcl
m222101@mpipc38:~/CMOR_PACKAGE/postprocessing>
```

Pre-processing

- Ksh scripts to convert ECHAM output variables to variables defined as described in Aerocom Tables



	M	N	O	P	Q	R	S
74	2D-M		metm	mass content of air	airmass	kg m-2	atmosphere_mass_of_air_per_unit_area
75	3D-M		aerosolm	SO4	mmr_so4	kg kg-1	mass_fraction_of_sulfate_dry_aerosol_in_air
76	3D-M		aerosolm	POM	mmr_pom	kg kg-1	mass_fraction_of_particulate_organic_matter_dry_aerosol_in_air
77	3D-M		aerosolm	SOA	mmr_soa	kg kg-1	mass_fraction_of_secondary_particulate_organic_matter_dry_aerosol_in_air
78	3D-M		aerosolm	BB	mmr_bb	kg kg-1	mass_fraction_of_biomass_burning_dry_aerosol_in_air
79	3D-M		aerosolm	BC	mmr_bc	kg kg-1	mass_fraction_ of_black_carbon_dry_aerosol_in_air
80	3D-M		aerosolm	NO3 aerosol	mmr_no3	kg kg-1	mass_fraction_ of_nitrate_dry_aerosol_in_air
81	3D-M		aerosolm	NH4	mmr_nh4	kg kg-1	mass_fraction_of_ammonium_dry_aerosol_in_air
82	3D-M		aerosolm	seasalt	mmr_ss	kg kg-1	mass_fraction_ of_seasalt_dry_aerosol_in_air
83	3D-M		aerosolm	dust	mmr_du	kg kg-1	mass_fraction_ of_dust_dry_aerosol_in_air
84	3D-M		aerosolm	aerosol water	mmr_aerh2o	kg kg-1	mass_fraction_of_water_in_ambient_aerosol_in_air
85	3D-M		tracerm	SO2	vmr_so2	mole mole-1	mole_fraction_of_sulfur_dioxide_in_air
86	3D-M		tracerm	dms	vmr_dms	mole mole-1	mole_fraction_of_dimethyl_sulfide_in_air
87	3D-M		tracerm	NO	vmr_no	mole mole-1	mole_fraction_of_nitrogen_monoxide_in_air
88	3D-M		tracerm	NO2	vmr_no2	mole mole-1	mole_fraction_of_nitrogen_dioxide_in_air
89	3D-M		tracerm	HNO3	vmr_hno3	mole mole-1	mole_fraction_of_nitrogen_acid_in_air
90	3D-M		tracerm	PAN	vmr_pan	mole mole-1	mole_fraction_of_peroxyacetyl_nitrate_in_air
91	3D-M		aerosolm	ccn gt 10 nm	ccn10nm_aer	m-3	condensation_nuclei_number_concentration_in_air

Example ECHAM/HAM

multiplemonths_merge_echam_parameters_A1c.sh (~/CMOR_PACKAGE/preprocessing) - VIM

File Edit View Terminal Tabs Help

```
## Mass Fraction

# Mass Fraction of SO4
cdo expr,'MASS_SO4=3*(SO4_NS+SO4_KS+SO4_AS+SO4_CS);' ${VAR}_tracerm.nc ${VAR}_tracerm_mass_so4.nc
#cdo expr,'MASS_SO4=SO4_gas+SO4_NS+SO4_KS+SO4_AS+SO4_CS;' ${VAR}_tracerm.nc ${VAR}_tracerm_mass_so4.nc

# Mass Fraction of Black Carbon
cdo expr,'MASS_BC=BC_KS+BC_AS+BC_CS+BC_KI;' ${VAR}_tracerm.nc ${VAR}_tracerm_mass_bc.nc

# Mass Fraction of POM=OC
cdo expr,'MASS_POM=OC_KS+OC_AS+OC_CS+OC_KI;' ${VAR}_tracerm.nc ${VAR}_tracerm_mass_oc.nc

# Mass Fraction of Seasalt
cdo expr,'MASS_SS=SS_AS+SS_CS;' ${VAR}_tracerm.nc ${VAR}_tracerm_mass_ss.nc

# Mass Fraction of Dust
cdo expr,'MASS_DU=DU_AS+DU_CS+DU_AI+DU_CI;' ${VAR}_tracerm.nc ${VAR}_tracerm_mass_du.nc

# Mole Fraction of SO2
cdo expr,'MOLE_SO2=(28.97/32.066)*SO2;' ${VAR}_tracerm.nc ${VAR}_tracerm_mole_so2.nc

# Mole Fraction of DMS
cdo expr,'MOLE_DMS=(28.97/32.066)*DMS;' ${VAR}_tracerm.nc ${VAR}_tracerm_mole_dms.nc
                                                                    68,1          63%
```

Small is beautiful

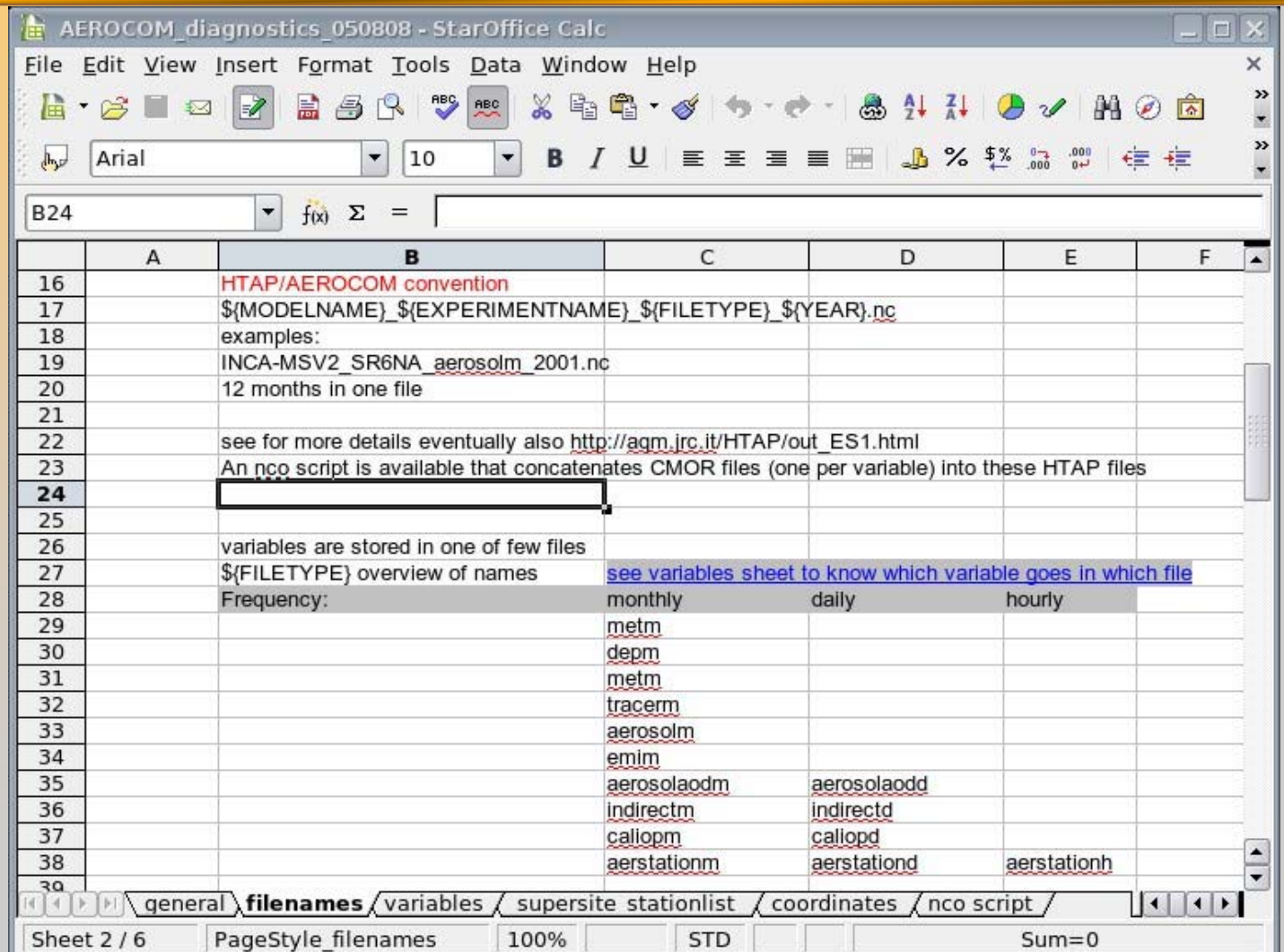
- Tcl script 1:

Runs CMOR executable for 12 months and puts output in 12 directories

- Tcl script 2:

Combines output script 1 to Aerocom specific file format output for each month

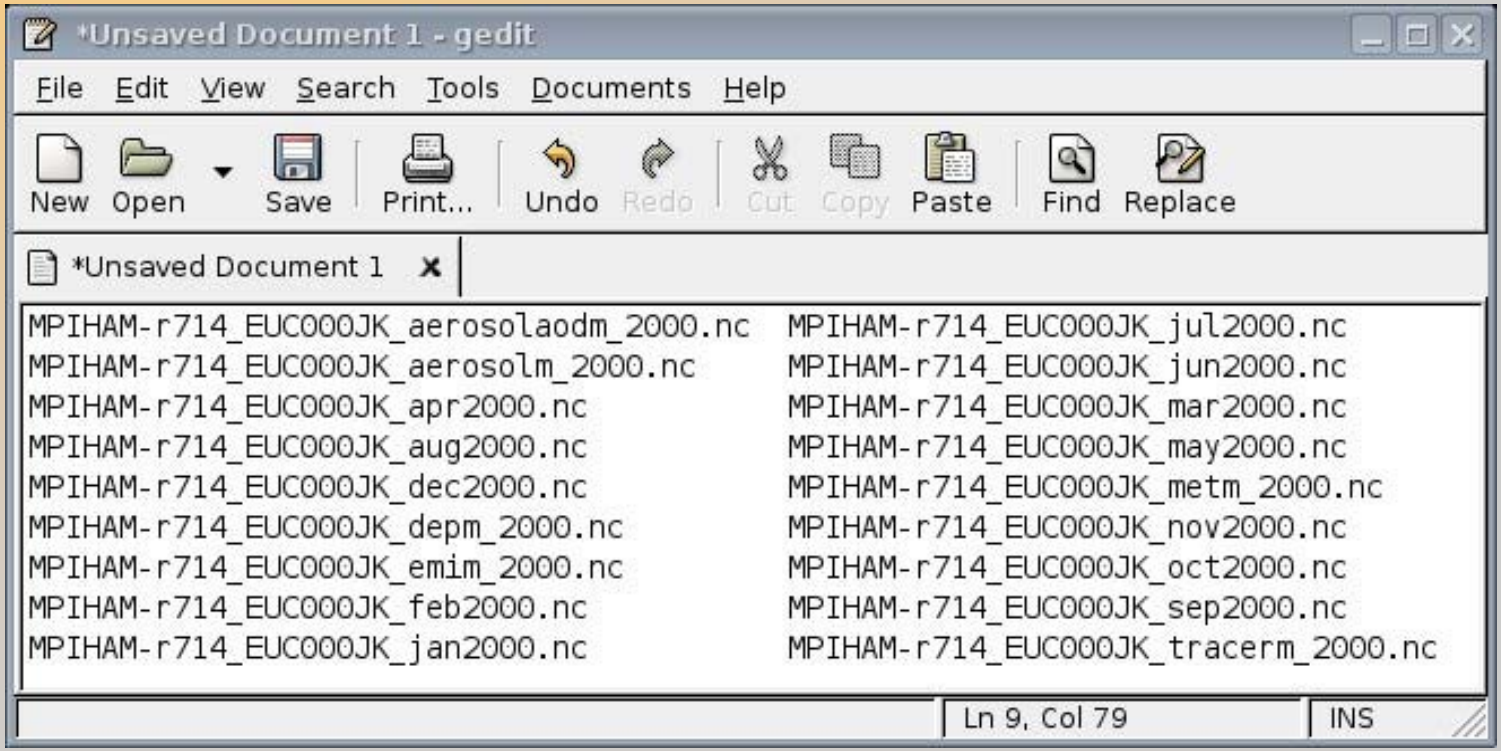
Aerocom filenames



Tcl script 3:

- For monthly data:
- puts all monthly Aerocom categorised files into a yearly file
- For daily data (files for 2D-D and 3D-D separately):
- puts all variables of daily files into monthly files

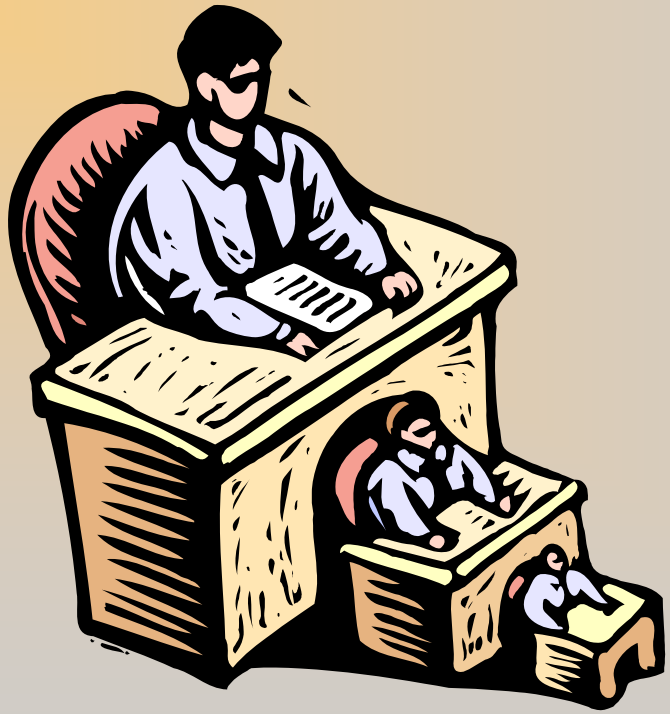
Example output



Still to do

- Optimise code (more flexible)
- Run CMOR for more ECHAM output

Good luck & have fun using CMOR!



Extra info

CMOR download:
http://www2-pcmdi.llnl.gov/cmor/download/
http://www-pcmdi.llnl.gov/software/cmor/cmor_users_guide.pdf

CMOR Tabellen CNRS:
http://www-lscedods.cea.fr/aerocom/CMOR/Aerocom home:
http://nansen.ipsl.jussieu.fr/AEROCOM/aerocomhome.html
http://nansen.ipsl.jussieu.fr/AEROCOM/protocol.html
See for explanations and details the excel file:
AeroCom_diagnostics.xls