CAM-chem automated diagnostics

The CESM team has developed a post-processing diagnostics tool using python. This page describes how to use it on chemistry output on the NCAR HPC, Cheyenne. At least a full year of data is needed with two months before and two months after the year. For example, to process 2004 you need output from November 1, 2003 to March 1, 2005. More information is at: https://github.com/NCAR/CESM_postprocessing/wiki, and the Quick Start Guide is here: https://github.com/NCAR/CESM_postprocessing/wiki/cheyenne-and-geyser-quick-start-guide

This automated diagnostic tool is useful for understanding overarching features of the simulation by comparing with a default set of observations and/or a previous simulation. It can also be used to produce timeseries (See Section 5 (below)) for long simulations, reducing the space of output files by compressing the data. This is highly recommended by the CAM-chem team. Also note, always only keep output that you really need for your science!

---

1. First time use: set your shell environment (see quick start guide)

2. Compile the post processing scripts

a) setup the post-processing in your case directory (on cheyenne, not casper)

```
> cd <case_dir>
> cesm_pp_activate (opens the virtual environment)
[(NPL) ] > create_postprocess --caseroot <case_dir>
[(NPL) ] > deactivate (closes the virtual environment)
```

or b) setup post-processing somewhere else

```
> cd /glade/scratch/<user>/post_processing/
> cesm_pp_activate (opens the virtual environment)
[(NPL) ] > create_postprocess --caseroot /glade/scratch/<user>/post_processing/<model-run>
[(NPL) ] > deactivate (closes the virtual environment)
```

If you get the *SUCCESS!* notification, the `<model-run>` folder has been been created in the post_processing location and analysis code has been added. **Note**: the <model-run> has to be the same name as your run folder.

3. Edit the .xml post processing scripts

Within the 'postprocess' directory (that was created in step #2) edit the scripts.

```
> ls *xml
```

Either use pp_config (like xmlchange) or edit the following files directly in an editor.

a) `env_postprocess.xml`

If post processing is occurring somewhere other than in <case_dir>, set the location of the model data:

```
> ./pp_config --set DOUT_S_ROOT=<full archive path of model run output to be analyzed>
```

Example:

```
> ./pp_config --set DOUT_S_ROOT=/gpfs/fs1/scratch/<user>/archive/<model-run>
```

Note: do not add slashes to the end of the path.

Tell the diagnostics what kind of grids to expect. For example the 0.9x1.25 degree resolution:

```
> ./pp_config --set ATM_GRID=0.9x1.25;
> ./pp_config --set ICE_NX=288
> ./pp_config --set ICE_NY=19
> ./pp_config --set LND_GRID=0.9x1.25
```

Other changes:

```
<entry id="GENERATE_TIMESERIES" value="FALSE" />
```

You can leave this setting to "TRUE" if you want to generate timeseries (for longer runs).

b) `env_diags_atm.xml`

Set up to compare with another model run.

```
<entry id="ATMDIAG_MODEL_VS_OBS" value="False" />
<entry id="ATMDIAG_MODEL_VS_MODEL" value="True" />
<entry id="ATMDIAG_CLEANUP_FILES" value="True" />
```

Test dataset (the run you want to analyse)

```
<entry id="ATMDIAG_test_compute_climo" value="True" />
<entry id="ATMDIAG_test_compute_zonalAvg" value="True" />
```

Control dataset (the run you want to compare with)

```
<entry id="ATMDIAG_cntl_casename" value="<cntr_case_name>" />
<entry id="ATMDIAG_cntl_path_history" value="<path-to-comparison-output-on-archive>" />
<entry id="ATMDIAG_cntl_compute_climo" value="True" />
<entry id="ATMDIAG_cntl_compute_zonalAvg" value="True" />
```

Time period of analysis for test and control cases, minimum 1 year and need output for 2 months either side of the full year to analyze.

```
<entry id="ATMDIAG_test_first_yr" value="2014" />
<entry id="ATMDIAG_test_nyrs" value="1" />
<entry id="ATMDIAG_cntl_first_yr" value="2014" />
<entry id="ATMDIAG_cntl_nyrs" value="1" />
```

Other diagnostic variables to set

```
<entry id="ATMDIAG_strip_off_vars" value="False" />
<entry id="ATMDIAG_netcdf_format" value="netcdfLarge" />
```

Diagnostic sets

```
<entry id="ATMDIAG_all_chem_sets" value="False" />
```

Then set chem sets to True manually except for chem set #6 (this one takes a long time).

Note 1: Chemistry diagnostic set 2 (Cset2) will only be calculated when performing a model-model comparison.

Note 2: To ensure all seasons are calculated make sure...

4. Run post-processing scripts

In `atm_averages` and `atm_diagnostics` files, make sure the #PBS account flag is set:

```
#PBS -A <account_number>
```

a) Run atm_averages (make sure run time is long enough to produce climo files)

```
> qsub atm_averages
```

Calculates the climatological values for test and control cases (~40 mins for 5 years), check the log files in logs folder.

Find climo files in: `$DOUT_S_ROOT/atm/proc/climo/$ATMDIAG_test_casename/ and: $DOUT_S_ROOT/atm/proc/climo /$ATMDIAG_cntl_casename/`

b) After atm_averages completes run atm_diagnostics (make sure run time is long enough to produce climo files)

```
> qsub atm_diagnostics
```

If instructions are followed as above, this step calculates model versus model values from the climo data created in step 4a) and creates diagnostic output (~10 mins for 5 years), check the log files in logs folder.

Find diagnostic files in: `$DOUT_S_ROOT/atm/proc/diag/$ATMDIAG_test_casename-$ATMDIAG_cntl_casename/`

To visualize the output, open `index.html` in a web browser.

5. Produce timeseries from model output to reduce storage space

Timeseries production with this tool can be completed for selected or all the output streams (e.g. atm, ocn). The user can specify whether to write out one large timeseries for the entire run (for example a full 100 years) or instead produce files containing smaller time-chunks (e.g. of 10 or 20 years). Time chunks (even 100 years) have to be always define.

a) Change the following general settings in `env_postprocess.xml` to generate any timeseries

```
<entry id="GENERATE_TIMESERIES" value="TRUE" />
```

Then, determine whether you would allow partial time-chunks to be produced or only full timeseries chunks.

```
<entry id="TIMESERIES_COMPLETECHUNK" value="FALSE" /
```

If this is set to TRUE, years that are not contained in a complete a time-chunk will not be processed. Therefore setting this to "FALSE" makes sure all your output gets processed.

The default setting is that all timeseries are generated. To allow user-control over this, set:

```
<entry id="TIMESERIES_GENERATE_ALL" value="FALSE" /
```

b) Define input to timeseries by modifying `env_timeseries.xml`

Turn on the output streams you wish to process. For example to process the atmosphere output stream:

```
<comp_archive_spec name="cam">
    <rootdir>atm</rootdir>
    <multi_instance>True</multi_instance
```

To **not** process this output stream, you have to set

```
<multi_instance>False</multi_instance
```

Next, you have to identify the length of each timeseries chunk, for all the different output streams you wish to process. For example, processing the atmosphere monthly output in 10-year time-chunks looks like:

```
<file_extension suffix=".h0.[0-9]">
        <subdir>hist</subdir>
        <tseries_create>TRUE</tseries_create>
        <tseries_output_format>netcdf4c</tseries_output_format>
        <tseries_tper>month_1</tseries_tper>
        <tseries_filecat_tper>years</tseries_filecat_tper>
        <tseries_filecat_n>10</tseries_filecat_n>
</file_extension>
```

c) Once timeseries specifications are set for all the output streams, run the script

```
>qsub timeseries
```

As for the other scripts, you need to make sure your project number is correctly set (`#PBS -A <account_number>`). If you have a lot of output, you may have to increase your PE-layout with more cores to finish. Usually, atm monthly data are processed very fast. Daily and sub-daily output can take a bit longer depending on the variables you are using.