CAM6 Tutorial Exercises

Set #1

Exercise 1: Get containers up and running. Run ARM97 IOP Default Case.
Questions: What is an IOP? Specific date and time. IOPs use a 'forcing' (specified horizontal advection and temperature) and initial condition data appropriate for a particular place and time.

How is this specified? There is a directory for this in the cam code, that tells the model what defaults to load for this case:
`> ls /opt/ncar/cesm2/components/cam/cime_config/usermods_dirs/scam_arm97`
Look at `shell_commands` and `user_nl_cam`
What do you think is happening here? Note that these commands are similar to sections of the script. It is just automatically applying things to the model.

Look at the model namelist:
`> less /home/scam/work/cases/tutorial.FSCAM.arm97/run/atm_in`
Look for the 'scam_nl' section. What is in there?

Exercise 2: Visualize the output from this case
Go to the jupyter lab notebook (in work/notebooks).
First visualize the 'testdata' and your data. Did you get the same answer that we did in testing?
In the 2nd cell, redefine file 1:
```
file1='/home/scam/work/cases/tutorial.FSCAM.arm97/run/tutorial.FSCAM.arm97.cam.h0.1997-06-18-84585.nc'
```

Now run through the cells (`shift-return`, or Run menu tab → Run All Cells)

Did you get the same answer as in testing?

Now visualize some fields. Let's look at the following (the 1st plotted variable is blue, the 2nd is orange):

a. Single-level water species: Surface precipitation from convection (PRECC) and large scale (PRECL).
b. Multi-level water species: Cloud Fraction (CLOUD), Cloud Liquid (CLDLIQ).
c. Single-level radiative fluxes: Outgoing LW (FLNT), Shortwave (FSNT). SW Cloud Radiative Effect (SWCF), Longwave cloud radiative effect (LWCF).
d. Multi-level radiative fluxes: Atmospheric heating rates for the longwave (QRL) and shortwave (QRS).
e. What is the name of the Relative Humidity variable (look in the 'dataset' array ds1 by just typing its name… you should find a variable named "RELHUM")?
f. Look at variables of your choice. Why is it cloudy or not? Where is there ice (or not)?

Exercise 3: Run a different IOP

Now for something different: let's try an Arctic case with mixed phase clouds. Re run the create_scam6_iop script with the 'mpace' iop.

For traceability, It is recommended to copy the `create_scam6_iop` script to something with a name that ends with '`.mpace`' like `create_scam6_iop.mpace` . This will enable you to go back and very what you did: you can difference the scripts, which are reproducible ways to build the model. It's advised to do that throughout the future exercises.

Go back and re-do the visualization exercise: just use one case, set file0= <mpace output file>.nc and comment out file 1.

Look at the fields above. What's different?

If you want to compare the cases, you can compare 2 files, but not with different time stamps. So here is some code for a vertical average. Copy this into your notebook script:

```
#Make vertical average and plot  y='lev' swaps axes, label sets up legend
v1=ds0[varm].mean(dim='time').plot(y='lev',label=runtx[0],yincrease=False)
v2=ds1[varm].mean(dim='time').plot(y='lev',label=runtx[1],yincrease=False)
#Add title
plt.title('Avg '+varm)
#Add legend
plt.gca().legend()
```

Now compare the cases. How different are the cases (also look at T and TS…)

Exercise 4: Output something else in SCAM

Let's take a look at the total precipitation and the deep convective mass flux
What are the appropriate variable names? Where do you find this?

Answer: In the CAM6 users guide, section 7.6:
http://www.cesm.ucar.edu/models/cesm2/atmosphere/docs/ug6/hist_flds_f2000.html
Hunt around a bit, but the names are not that obvious: CMFMCDZM, PRECT

Now there are two options:
1) The 'right way' (robust). Go to the script and add these names to the `fincl1` part of the `user_nl_cam`:

```
fincl1= 'CDNUMC', 'AQSNOW','ANSNOW','FREQSL','LS_FLXPRC',’CMFMCDZM’,’PRECT’
```

Now: rerun the ARM97 case, and use the Jupyter notebook to visualize the new variables (verify they made it to the output with):

```
> ncdump -h
/home/scam/work/cases/tutorial.FSCAM.arm97/run/tutorial.FSCAM.arm97.cam.h0.1997-06-18-84585.nc
```

2) The 'Quick' Way
Namelist parameters are not part of the build and compile, and they can be picked up at run time. So you could go to the run already done, modify the atmospheric namelist, and then re-execute the pre-compiled code. Note: A smarter SCAM script could not clean things up totally.

WARNING: Only do this with changes that (A) Are only in the namelist (or they will not be picked up) and (B) Do not change the model answers (i.e., diagnostic output only).

First go to the run directory in the case:

```
> cd /home/scam/work/cases/tutorial.FSCAM.arm97/run/
```

Then edit the 'atm_in' file. You can use vi or emacs in the bash terminal.
Or: browse to this directory in the Jupyter Lab file browser and open it.

Find the '`fincl1`' line, and add the variables:

```
 Fincl1 = 'CDNUMC','AQSNOW','ANSNOW','FREQSL','LS_FLXPRC','CMFMCDZM','PRECT'
```

Now go to a terminal. From the run directory, execute the model manually (last line in the script):

```
>../bld/cesm.exe
```

If you are not in the right directory this will not work

Now: we can visualize the new variables.

Note that the number of vertical levels in some CAM variables, such as CMFMCDZM or CLOUD, is 'ilev'. That means the variable values reside on interfaces, not layers. This will require you to change the name of the 'y' plotting dimension for vertical fields.

Exercise 5: Switch Parameterizations using the namelist
Now it gets really interesting. Why do we need a convective parameterization anyway? Let's turn it off. It's not obvious, and you cannot do this with every parameterization, but you can do it with the deep convection via a namelist variable. Let's find the variable:

Go to the namelist definitions page:
http://www.cesm.ucar.edu/models/cesm2/settings/current/cam_nml.html

And search (box on right side) for ‘deep’. What is the variable that selects the deep scheme? What is the text we want to put in there?

It is recommended to copy the `create_scam6_iop` script to something with a name that ends with ‘`.nodeep`’ like `create_scam6_iop.nodeep.` Remember to use the original arm97 case.

In the script:

1. Change the case name to `tutorial.nodeep`
2. Add a line to the user_nl_cam with the `namelist_var` and ‘`value`’ you just found: namelist_var = ‘value’
3. We suggest also adding the total precipitation (PRECT) to your namelist (via the ‘`fincl`’ part of the namelist). Or you can analyze it as the sum of convective (PRECC) and large scale (PRECL) precipitation rate.

Now run the case.

Spend some time exploring what it does by visualizing it in Jupyter.

Did you turn it off? How do you know? Hint: look in the namelist (atm_in) to see if your change is there, and look at a convective variable (PRECC).

How does the precipitation change? How do the clouds change? What is the impact of deep convection in this ARM97 case?

Advanced: What happens if you try to add CMFMCDZM as an output variable without a deep convective scheme? (See exercise 4 method 2 to quickly try this)

Exercise 6: Modify the code

Now we make you a model developer and get your hands dirty.
Some parameters cannot be changed by namelist, and you have to modify code. This can be as simple as what we will do, or as complex as writing a whole new parameterization.

It is recommended to copy the `create_scam6_iop` script to something with a name that ends with ‘`.smalltau`’ like `create_scam6_iop.smalltau.` Remember to use the original arm97 case.

See these lines in the `create_scam6_iop` script:

```
#-------------------
#  source mods: copy them into case directory
#-------------------
```

```
/bin/cp -f  ${usrsrc}/* SourceMods/src.cam/
```

What is 'usrsrc'? (Further up in the script…)

```
### Set location of user source mods (if any)
setenv this_dir  `pwd`
setenv usrsrc  ${this_dir}/mods/$CASETITLE
```

So what does this mean? If you are running the script in `/home/scam/work`, then it will take files in `/home/scam/work/mods/$CASETITLE` and copy them into the case directory to a place (`./cases/$CASETITLE/SourceMods/src.cam/`) where the model knows to pick up and replace files.

So let's make a `${usrsrc}` directory
From `/home/scam/work`

```
>mkdir mods
>mkdir mods/tutorial.smalltau
```

Now we need to get a piece of code to modify. Where is the model source code?
(look in the `create_scam6_iop` script: `CESMDIR=/opt/ncar/cesm2`)

Where is the deep convection code?
`/opt/ncar/cesm2/components/cam/src/physics/cam/zm_conv.F90`

Copy this into the 'usrsrc' directory.

```
cp /opt/ncar/cesm2/components/cam/src/physics/cam/zm_conv.F90
/home/scam/work/mods/tutorial.smalltau/
```

Now, navigate to this file with the jupyter lab file browser, and open it up.
Search ('Find': command or control-F) '`tau`' in the code, the deep convective time scale.
What is the value? (Hint, line 126). Now change it. Go big, let's decrease it by a factor of 4.

Save the `zm_conv.F90` file.

Modify the script to use the arm97 IOP, but make the case name the same as the mods directory: `tutorial.smalltau`. Make sure you still have the `CMFMCDZM` and `PRECT` variables in your fincl1 and that you are running arm97 again.

Does it compile? Does it run?

Now visualize the output. What does making `tau` smaller do to the convection?

Exercise 7: Play with deep convective parameters
Let's play with a few namelist parameters in the deep convection scheme.

Remember: It is recommended to copy the `create_scam6_iop` script to something with a name that ends with '`.zmke`' like `create_scam6_iop.zmke.` Remember to use the original arm97 case.

1) Convective Evaporation Efficiency

`zmconv_ke` is a tunable evaporation efficiency in ZM deep convection scheme. What is the current value? Look in the namelist for your arm97 case:

```
>less /home/scam/work/cases/tutorial.FSCAM.arm97/run/atm_in
```

Now what are other values used? Look in the namelist definitions web page for this variable. What does it say? Change it and see what happens...how does it affect convection?

2) Convective Rain Formation

`zmconv_c0_lnd` and `zmconv_c0_ocn` are the convective autoconversion coefficients that control how much and when deep convection precipitates. Look at the values in the namelist:

```
>less /home/scam/work/cases/tutorial.FSCAM.arm97/run/atm_in
```

What are the values? Which is bigger? Why?
Answer: The ocean is larger. They represent the propensity to rain. Over land there are more aerosols, more cloud drops, and smaller drops. They are less efficient at raining.

Try making the land efficiency the same as ocean. So add a new line to the user_nl_cam namelist with a value for `zmcon_c0_lnd =` [some value] . Make a new case name (tutorial.zmc0lnd) and re-run the arm97 case. What changes?

Set #2: Try to get at least to #6 from Set 1 first

Exercise 1: Change a physics parameterization with configure

Some things cannot be changed by the namelist since the namelist is applied to code which is already compiled. These changes alter the code the model compiles. For CAM, they often occur using the 'configure' command. This sets up the number of levels and number of advected species.

So if we switch back from the CAM6 to the CAM6 microphysics (MG2 to MG1), the number of advected species is different because rain and snow are prognostic in MG2 and CAM6.

These changes are made by the CESM build process once the case is created, but before it is totally set up (e.g., it creates the build scripts and namelists for the atmosphere and data ocean).

To change the CESM options, you can do 2 things. You could run 'create_newcase' then go in and manually edit the CESM set up files (the `*.xml` files in the `case` directory) before running case.setup. But 3 months from now you would have forgotten this step. So better practice is to build it into the case.

It is recommended to copy the `create_scam6_iop` script to something with a name that ends with '`.mg1`' like `create_scam6_iop.mg1`

Since this is going to be a new experiment and we will be creating a new case the first parameter to edit would be the CASETITLE to reflect the nature of the experiment.

You will want to rename the case for this new experiment. We will call this case tutorial.mg1.

```
set CASETITLE=tutorial.mg1
```

Now we are going to issue a command in the script to change parts of the XML files (`xmlchange`). Find this section in the script:

```
#------------------
# XMLCHANGE OPTIONS HERE
#------------------
```

And let's change the CAM configure options to use mg1 microphysics:

```
### Append to CAM configure options
./xmlchange --append CAM_CONFIG_OPTS=' -microphys mg1'
```

How did we know what to put there? The CAM configure script can be viewed at:

`/opt/ncar/cesm2/components/cam/bld/configure`

Running the configure script with the -h parameter will list all the options available to change. In the case of microphysics (`-microphys` parameter) we see that `mg1, mg2, and rk` are the three microphysic schemes available in CESM2

Now check your case, let's run the `mpace` iop, with the standard namelist parameters you used for the `mpace` case.

Run the case.

Let's look at the output and compare it to the control case with MG2. How is it different than MG2? Look at the water species, especially the ice phase.

Now there is one complication here. Take a look at the namelists actually used for these cases (`tutorial.FSCAM.mpace/run/atm_in` and `tutorial.mg1.FSCAM.mpace/run/atm_in`)

```
> cd work/cases
> diff tutorial.FSCAM.mpace/run/atm_in tutorial.mg1.FSCAM.mpace/run/atm_in
```

What happened? Why did so many things change? This is a warning that there are 'unintended consequences' due to the way CESM is set up. Unfortunately parts of the model key off of the microphysics to decide whether the model should be CAM6 or CAM5, and other tuning parameters are set on this basis.

To do a 'clean' test of MG2 v. MG1 you would need to set the namelist values for MG1 back to MG2 values where needed by looking at the diffs and then forcing the '`.mg1`' script to set the appropriate fields to the MG2 values and re-run. (note, you could do this without re-compiling if you really wanted to, and are not interested in keeping the first mg1 answers.

Exercise 2: CLUBB Parameters

Let's see how sensitive boundary layer clouds are to CLUBB parameters. We will use the 'cgilsS11' case. This is a point in the Eastern Pacific between California and Hawaii where the clouds change from solid stratus to broken shallow cumulus clouds.

Remember to make a new script name: `create_scam6_iop.cgilsS11` Run the cgilsS11 case using the standard parameter values (with `set IOP = cgilsS11`)

Now do a 2nd run with the following added to `user_nl_cam`:

```
clubb_gamma_coef = 0.4
```

Remember you can make a new script for this if you want:
`create_scam6_iop.cgilsS11_gam0.4`

The second run will have an increased value of the gamma coefficient found in Eqn. (3.45) of the CLUBB tech doc. Increasing gamma will increase the values of all the third-order moments found in Eqns. (3.51-3.57). The direct effect of this is to cause turbulent transport to deepen the cloud layers.

What happens to SWCF and CLDLIQ when `clubb_gamma_coef` is increased?

Exercise 3: Get a different Input file with CESM commands

Let's change the climate with a new input file that has SST+4K from the standard file everywhere

We are going to use a different case cgilsS11 for this.

Now once that is going, we are going to modify the script to get a new SST input file.

The SST file is actually not part of CAM, it is part of the data ocean model (docn). So we need to modify CESM itself to point it to the new file. Go to one of the case directories, and look for a file called '`env_run.xml`'.
`(e.g.: > less ./work/cases/tutorial.FSCAM.arm97/env_run.xml)`

And look for 'SSTICE' in the file. See what it looks like?

Now let's get a new file, with the SST increased uniformly by +4K. Scientifically this follows Cess et al (1990):

Cess, R. D., and others. "Intercomparison and Interpretation of Climate Feedback Processes in 19 Atmospheric General Circulation Models." J. Geophys. Res 95 (1990): 16,601-16,615.

Point a browser to here to get the file:

ftp://ftp.cgd.ucar.edu/pub/andrew/tutorial/sst_HadOIBl_bc_1x1_2000climoP4K_c180814.nc

Now from your local machine (NOT the jupyter lab terminal) copy or move the file to the work directory, in '`testdata`' ({local path}`/work/testdata`).

Verify it is where you think it is from the terminal:
```
>ls /home/scam/work/testdata/sst_HadOIBl_bc_1x1_2000climoP4K_c180814.nc
```

Remember, CESM changes happen once the case is created (i.e., once `env_run.xml` exists), but before it is totally set up (e.g., it creates the namelists for the atmosphere and data ocean).

To change the CESM options, you can do one of 2 things. You could run '`create_newcase`' then go in and manually edit the `env_run.xml` file before running `case.setup`. But 3 months from now you would have forgotten this step. So better practice is to build it into the case.

Instead, it is recommended to copy the `create_scam6_iop` script to something with a name that ends with '`.sst4K`' like `create_scam6_iop.sst4K`

Where you see this section:

```
#------------------
# XMLCHANGE OPTIONS HERE
#------------------
```

Add a reference to the new file:

```
set SST_DATASET=/home/scam/work/testdata/sst_HadOIBl_bc_1x1_2000climoP4K_c180814.nc
./xmlchange  --id SSTICE_DATA_FILENAME --val $SST_DATASET
```

Now check your script to make sure you have the right case and namelist settings, and run it.

Now visualize the case, and how it is different from the original cgilsS11 case.

- What happens to the clouds (CLDLIQ, CLOUD, CLDTOT)?
- What happens to the radiation fields at the top of the atmosphere (FSNT, FLNT, SWCF)?

What you have just done is an experiment often used to find out the cloud response to climate changes. The local cloud response to SST changes is the local 'cloud feedback', usually expressed in Wm^{-2} K^{-1}. So if you take your change in average cloud radiative effect (SWCF and LWCF) and divide by the change in SST (4K), you get the 'cloud feedback'. What is it?

To get an average over an xarray in python, use:

```
v0mean=ds1[varn].mean(dim='time')
v0mean.values
```

Subtract the two runs and divide by the temperature change to get a SW and LW Cloud Feedback.

Exercise 4: Playing with cloud microphysics

If you are interested in cloud microphysics, there are several sensitivity tests that one can perform, and that are commonly used to ‘tune’ CAM6.

- ***Change high clouds with ice precipitation thresholds.***
  In CAM, ice has a fixed size threshold for when it starts precipitating (turning to snow) this is the critical size (diameter) for conversion of ice to snow. It is called ‘dcs’ in the code. It is also a namelist parameter. So go to:
  http://www.cesm.ucar.edu/models/cesm2/settings/current/cam_nml.html

  And search for ‘dcs’. What is the namelist variable called?

  What is the value of DCS in the control cases? How do you find out?
  Hint: look in the atmospheric namelist. Where is that in a case?

  What is your hypothesis about what will happen if you increase or decrease DCS?
  Remember DCS is the SIZE threshold for when ice turns to snow.
  What might happen to ice mass?

  Now, make a new script (create_scam6_iop.dcs or similar) and set it to the arm97 IOP
  Finally, add the new namelist parameter with a change to the `user_nl_cam`
  (A new line there. Pick a new value)
  And run the script…

  Visualize the output relative to the base case. Look at CLOUD and CLDICE.
  Also NUMICE. What happened? Was your hypothesis correct?

- ***Change the vapor deposition (freezing) efficiency.***
  It is called ‘berg’ in the code. It is also a namelist parameter. So go to:
  http://www.cesm.ucar.edu/models/cesm2/settings/current/cam_nml.html

  And search for ‘berg’. What is the namelist variable called?

  What is the value of ‘berg’ in the control cases? (see above, hint: atmospheric namelist)

  What is your hypothesis about impacts of changes in the vapor deposition efficiency?
  What might happen to ice mass? Liquid?

  Now, make a new script (`create_scam6_iop.berg` or similar)
  Set it to the mpace IOP (we want a cold cloud case for this)
  Finally, add the new namelist parameter with a change to the `user_nl_cam`
  (A new line there. Pick a new value)

  And run the script…

Visualize the output relative to the base mpace case.
Look at CLOUD and CLDICE, CLDLIQ
What happened? Was your hypothesis correct?

Extra Credit: Look at 2D (or 2D x ridge-number) variables the "topofile":

/opt/ncar/inputdata/atm/cam/topo/T42_nc3000_Co060_Fi001_PF_nullRR_Nsw042_20180111.nc

Python (using xarray):

```
tpfile='/opt/ncar/inputdata/atm/cam/topo/T42_nc3000_Co060_Fi001_PF_nu
llRR_Nsw042_20180111.nc'
# Open Data Set
topo = xr.open_mfdataset(tpfile)
```

What are the relative magnitudes of SGH30, SGH and MXDIS(*,*,0)?

```
topo['SGH30'].plot()
topo['SGH'].plot()
mx0=topo['MXDIS']
mx0[0,:,:].plot()
```

Set #3

Exercise 1: What is the impact of CO2 increases only?
SCAM allows us to separate the effect of CO2 increases on the atmosphere (including radiative heating, temperature and clouds) that cause climate changes (increases in surface temperature) from the climate changes themselves. We tested the impact of climate changes with the SST+4K experiment. Now let's look at how increasing CO2 changes the atmosphere without surface temperature change: called a 'fast adjustment'

What value does SCAM use for CO2? What is it?
How would you find it? Hint: look in the namelist for CAM in one of your runs, or the online list.

Note: there are multiple ways that CO_2 interacts in CESM2 in the coupled system, but for CAM we can change CO_2 with a namelist parameter (`co2vmr`).

Now let's quadruple CO_2.
- Make a new case, copying the script (`create_scam6_iop.4xco2`)
- Give it a new case name: `tutorial.4xco2`
- Use the cgilsS11 IOP
- Make a new namelist entry for `co2vmr` in `user_nl_cam` for that is 4 times the current value.
- Run the model

Visualize the new 4xCO2 simulation relative to the base cgilsS11 case.
- What is different with:
    - Heating rates (QRL, QRS): which changes more? Why? (Hint, what bands does CO2 affect)
    - Cloud fields (Fraction water content)
    - Top of atmosphere radiation fields. Also CLEARSKY radiation flux fields: (F*C: FLNTC, FSNTC). What changes and why?
- Remember atmospheric temperature is relaxed, so it is not a good test of the temperature change.

Exercise 2: Stop the Sun
What happens without a diurnal cycle?
CAM is not set up to do this by default, but you can force the radiation code to have a constant solar zenith angle. So this is a code modification.

CESM code regulates orbital parameters, to ensure consistency across components. This is contained in '`shr`' code:
`/opt/ncar/cesm2/cime/src/share/util/shr_orb_mod.F90`

So make a new case with a copy of a script: `create_scam6_iop.cnstsza`

Use the arm97 IOP.
Set the case name (`tutorial.cnstsza`)

Now just as with exercise 6 for the first day (`tutorial.smalltau`):
So let's make a `${usrsrc}` directory
From `/home/scam/work`

```
>mkdir mods
>mkdir mods/tutorial.cnstsza
```

Copy the shared orbital code into the 'usrsrc' directory.

```
cp /opt/ncar/cesm2/cime/src/share/util/shr_orb_mod.F90
/home/scam/work/mods/tutorial.cnstsza/
```

Now, navigate to this file with the jupyter lab file browser, and open it up. Around Line 75, replace the contents of else statement with some constant value, such as:

```
shr_orb_cosz = cos(0.73391095_shr_kind_r8)
```

What angle does this correspond to? Hint: the value is in radians.

Save the `shr_orb_mod.F90` file.

Now go to the script, make sure you are using the arm97 IOP and have a different case name.

Run the script and visualize it.

- Where would you look to see differences?
- Hint: look at the surface fluxes or surface temperatures. Then convective outputs.

Exercise 3: Aerosol radiative forcing

Let's estimate the aerosol radiative forcing. Unfortunately aerosols cannot be easily removed from the prognostic radiation parameterization. However, there is a mechanism in the radiation code to run it again 'diagnostically' and generate a further set of radiation fields. These end up as additional variables with a '`*_d1`' suffix

So, let's go back to the original `tutorial.arm97` case, and add the following (in red) to the `user_nl_cam`:

```
cat >> user_nl_cam << EOF
 use_topo_file          = .true.
 mfilt           = 2500
 nhtfrq          = 1
 fincl1= 'CDNUMC', 'AQSNOW','ANSNOW','FREQSL','LS_FLXPRC'
 prescribed_strataero_cycle_yr          = 2000
```

```
 prescribed_strataero_datapath          = "\$DIN_LOC_ROOT/atm/cam/ozone"
 prescribed_strataero_file =
'ozone_strataero_CAM6chem_2000climo_zm_5day_c171004.nc'
/
&rad_cnst_nl
 rad_diag_1 = 'A:Q:H2O', 'N:O2:O2', 'N:CO2:CO2', 'N:ozone:O3', 'N:N2O:N2O',
    'N:CH4:CH4', 'N:CFC11:CFC11', 'N:CFC12:CFC12'
/
EOF
```

Hint: this is a namelist change which only adds variables but does not change answers. So you could add a single line (`rad_diag_1 = ….`) to the existing atm_in namelist (in the right place) if you are feeling confident.

To visualize the differences, you can compare 2 runs, or make the 2nd run just different variable names from the first one. E.g. in the jupyter notebook: `varn2=varn+'_d1'`

What is the magnitude of the aerosol radiative forcing? (hint, remember the 'mean' method over a dimension in python from day 2, exercise 3:

To get an average over an xarray in python, use:

```
v0mean=ds1[varn].mean(dim='time')
v0mean.values
```

Further exercises: Plan your own

We encourage you now to 'play' with SCAM. See if you can formulate a science question to test with SCAM. Talk to one of the experts about specifics, but start with a set of questions:

1. What regime are you looking at. What IOP might this be? (see Gettelman et al 2019, Table 1). Run a basic case.
2. What processes or affect are you interested in? What parameterization would this be?

From here, namelist changes can be made to explore model behavior:

3. Is there an adjustable parameter you can find? Look in the namelist search function on the name of a parameterization.
4. Modify a parameter. Think about which way it might affect a process (you may need help with this step)
5. Run another case to test your hypothesis

Alternatively: there may be code modifications or configuration options that can be explored.

Or: ask. Some things are possible, some are difficult.