Lawrence Livermore National Laboratory

Computational Aspects of the UQ Project at LLNL

February 14, 2011

John Tannahill & Don Lucas

Lawrence Livermore National Laboratory, P. O. Box 808, Livermore, CA 94551

This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344

LLNL-PRES-469892

LLNL UQ Strategic Initiative LDRD.

- Richard Klein (PI), Xabier Garaizar (co-PI)
- Climate Team:
  - Curt Covey (team lead)
  - Donald Lucas (modeling & analysis)
  - John Tannahill (s/w architecture & development)
  - Yuying Zhang (observations & analysis)

- UQ Pipeline Team:
  - David Domyancic
  - Scott Brandon
- A number of other UQ researchers:
  - Gardar Johannesson

CAM/CESM particulars for this work.

- Basic CAM/CESM configuration used:
  - 1.9x2.5° horizontal resolution
  - 26 vertical levels
  - Finite-Volume dynamical core
  - CAM3_5_1 => CAM4 physics
- CAM/CESM namelist code modified to allow for 36 parameter values to be input.
- CESM scripting system modified as needed.
- Extensive Python script developed to insulate user from any CAM/CESM specifics.

CAM/CESM UQ parameters of interest on 2/10.

#	Variable Name	Description	File Name (.F90)	Src*
1	**rhminh**	Threshold RH for fraction of high stable clouds	cloud_fraction	**J**
2	**rhminl**	Threshold RH for fraction of low stable clouds	cloud_fraction	**J+S**
3	**rliqice**	Effective radius of liq. cloud droplets over sea ice	pkg_cldoptics	**R**
4	**rliqland**	Effective radius of liquid cloud droplets over land	pkg_cldoptics	**R**
5	**rliqocean**	Effective radius of liquid cloud droplets over ocean	pkg_cldoptics	**R**
6	**ice_stokes_fac**	Scaling factor applied to ice fall velocity	pkg_cld_sedimnent	**S**
7	**capnc**	Cloud particle num. density over cold land/ocean	cldwat	**R**
8	**capnsi**	Cloud particle number density over sea ice	cldwat	**R**
9	**capnw**	Cloud particle number density over warm land	cldwat	**R**
10	**conke**	Evaporation efficiency of stratiform precipitation	cldwat	**J**
11	**icritc**	Threshold for autoconversion of cold ice	cldwat	**S**
12	**icritw**	Threshold for autoconversion of warm ice	cldwat	**S**
13	**r3lcrit**	Critical radius at which autocon. becomes efficient	cldwat	**R**
14	**ricr**	Critical Richardson number for boundary layer	hb_diff	**K/B**
15	**c0**	*Shallow* convection precipitation efficiency	hk_conv	**J**
16	**cmftau**	Time scale for consumption rate of shallow CAPE	hk_conv	**K/B**
17	**alfa**	Initial cloud downdraft mass flux	zm_conv	**J**
18	**c0**	*Deep* convection precipitation efficiency	zm_conv	**J**
19	**dmpdz**	Parcel fractional mass entrainment rate	zm_conv	**S**
20	**ke**	Environmental air entrainment rate	zm_conv	**J**
21	**tau**	Time scale for consumption rate of deep CAPE	zm_conv	**J**

*Source => **J**:Jackson, C, J Clim 21:6698, '08; **K/B**:Klein, S & Bader, D, Suggestion, '09; **R**:Rasch, P, Suggestion, '09; **S**:Sanderson, B, CCSM Workshop, '09

Added UQ parameters of interest since 2/10.

#18 (c0) split, so only counted once here.

#	Variable Name	Description	File Name (.F90)	Src*
22	**fac**	ustar parameter in PBL height diagnosis	hb_diff	**Z**
23	**fak**	Constant in surface temperature excess	hb_diff	**Z**
24	**betamn**	Minimum overshoot parameter	hk_conv	**Z**
25	**sgh_scal_fac**	Land roughness scaling factor	physpkg	**T**
26	**c0_lnd**	*Deep* convec. precipitation efficiency over land	zm_conv	**J**
	c0_ocn	*Deep* convec. precipitation efficiency over ocean	zm_conv	**J**
27	**capelmt**	Threshold value for CAPE for deep convection	zm_conv	**Z**
28	**cdn_scal_fac**	Ocean roughness scaling factor	shr_flux_mod	**Z**
29	**z0m_scal_fac**	Mois. & heat resistance to vegetation scaling factor	Biogeophysics1Mod	**Z**
30	**dt_mlt_in**	Temperature at which melt begins	ice_shortwave	**B**
31	**r_ice**	Sea ice tuning parameter	ice_shortwave	**B**
32	**r_pnd**	Ponded ice tuning parameter	ice_shortwave	**B**
33	**r_snw**	Snow tuning parameter	ice_shortwave	**B**
34	**rsnw_melt_in**	Maximum snow grain radius	ice_shortwave	**B**
35	**ksno**	Thermal conductivity of snow	ice_therm_vertical	**B**
36	**mu_rdg**	Gives e-folding scale of ridged ice	ice_mechred	**B**

*Source => **B**:Bailey,D, etal, Suggestion, '11; **J**:Jackson, C, J Clim 21:6698, '08; **T**:Taylor,M, Suggestion '10; **Z:**Zhang, M, Suggestion, '10

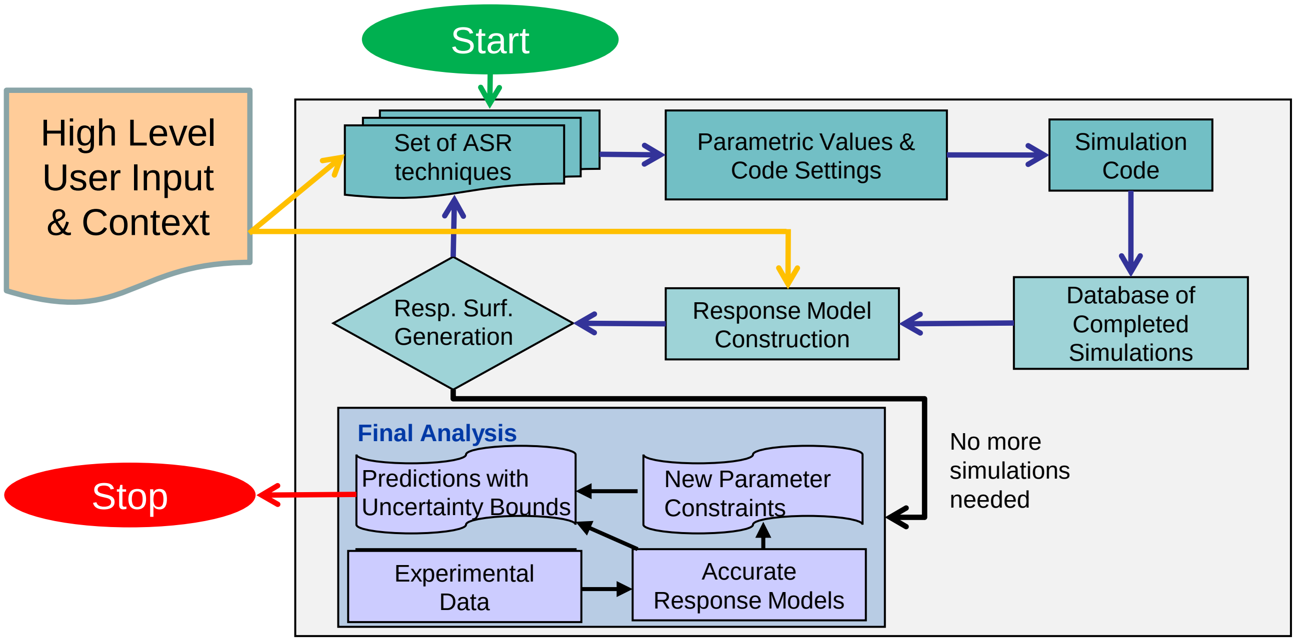
CAM/CESM UQ runs.

- Run on LLNL's Atlas Linux Cluster.
- Use LLNL's UQ Pipeline tool to automate runs.
  - Python interface scripts developed.
- Runs to date:

CAM/CESM version	ocn mode	# params	run dates	# runs	simyrs/ run	total simyrs	tasks/ run	thrs/ task	cores/ run	concur. runs	cores/ job
CAM3.6	AMIP	21	1/10-5/10	1,242	12	14,904	192	2	384	11	4,224
CAM4	AMIP	21-28	5/10-1/11	1,695	12	20,340	192	2	384	11	4,224
CESM1	AMIP	29	1/11	59	12	708	192	2	384	11	4,224
CESM1	SOM	36	2/11-	Test	30-60?	Test	192	4	768	4	3,072
Total				2,996		35,952					

LLNL UQ Pipeline tool.

- Stages & executes a set of concurrent ensemble simulations.
- Provides a wide range of sampling strategies, as well as analysis capabilities like “MARS”.
- Self-guiding, self-adapting technologies are being developed that will automatically & efficiently steer the study parameter space to explore.



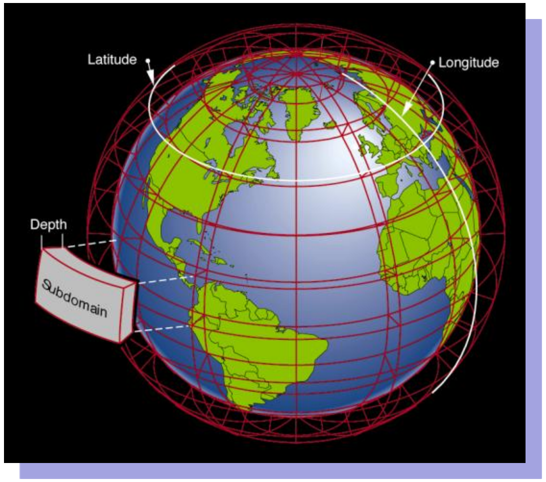
LLNL CESM/UQ system is fully automated.

- "Push the button" & in theory you can do a study made up of 100's of runs without intervening.
- Study is relatively simple to set up; 3 files required:

- uq_info.py:
  - Contains pipeline/application interface data dictionaries.
  - Generally modify a couple of items here at the beginning of each study.
- appl_interface.py:
  - Provide prep_ensemble, prep_run, post_run, & post_ensemble interface functions required by pipeline.
  - Generally no need to modify once established.
- lcesm_run.py:
  - Shields user from CESM details.
  - Generally no need to modify.

Migration from CAM standalone to CESM.

- More difficult than anticipated.
- "UQ ensemble runs" does not appear to be a use case that the CESM scripts currently facilitate?
- The primary CESM setup functionality that is needed for our work is just:
  - Configure/Build the code.
  - Set up the namelist files.
- Many thanks to the various NCAR people who helped with this transition.



“Atlas Node 587” issue – Discovered by chance.

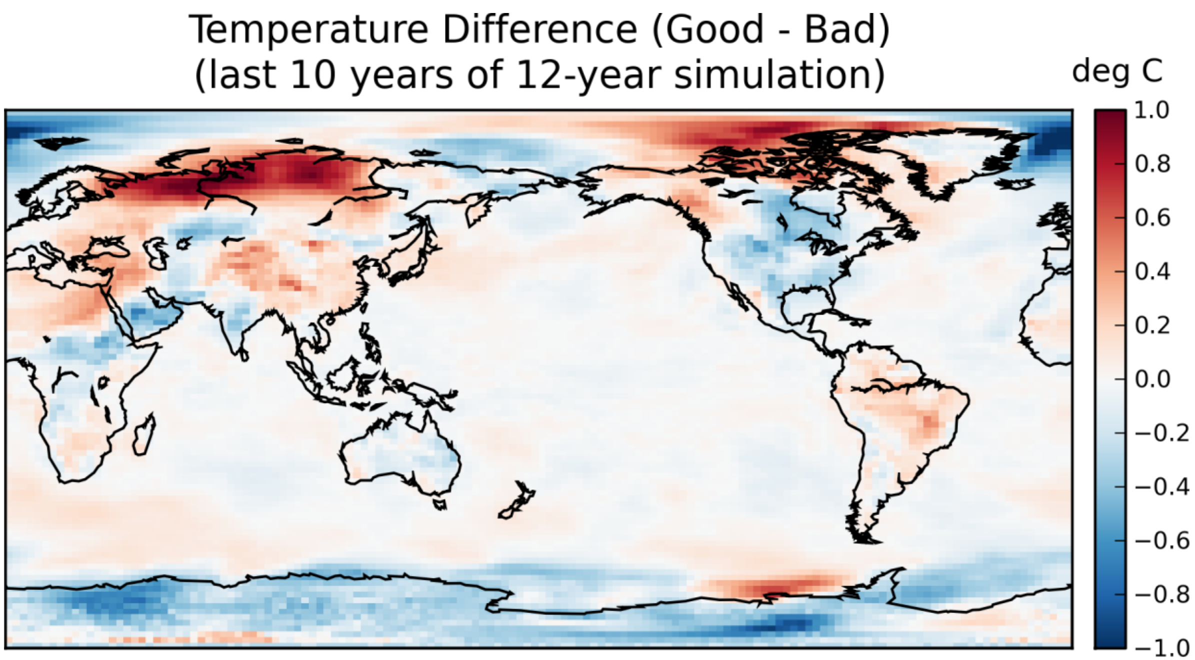
- Inadvertently deleted some files from a study & had to redo some runs.

- Old summary diagnostic files should have matched the new ones “bit for bit”, but some did not?
- Ended up taking ~2 months to resolve this issue.
  - A concerted effort by a number of people.

“Atlas Node 587” issue – The symptoms.

- Made many many test runs, which eventually led to the following observations:
  - Atlas runs using Node 587 could lead to “some sporadic bad diagnostic output”.
    - Bad output values were still “plausible”.
    - Bad output values were “significantly different” from what they should have been.
  - Atlas runs not using Node 587 produced no bad output.
  - Running on a different, but very similar machine produced no bad output.
- A co-worker, Art Mirin, had also been experiencing similar problems with his CAM runs.

Node 587 effects – Last 10 years, 12-yr CAM run



(CAM/UQ MOAT3 study, run0035, monthly average history files)

Regional Max Temperature Difference of ~1°C between Good & Bad runs.

"Atlas Node 587" issue – The cause & remedy.

- Our systems people ended up finding an intermittently occurring small floating point roundoff error on one core of Node 587:
  - Order of one off in the least significant digit.
  - These roundoff errors propagated to significant differences in CAM diagnostic values over time.

```
-2.181631512210958e+18
-2.181631512210958e+18
-2.181631512210959e+18
-2.181631512210959e+18
-2.181631512210959e+18
-2.181631512210958e+18
-2.181631512210959e+18
-2.181631512210958e+18
```

- Checked all other nodes on Atlas & similar LLNL machines.
  - Found one more on another machine.
- Permanently removed both nodes.
  - Nodes sent to AMD for further analysis.

"Atlas Node 587" issue – The fallout for us.

- Only made test runs for several weeks.
- Reran all runs that had used Node 587 – roughly 100.
- Used ~1M CPU-hrs?
- Now for every UQ run, we actually do two, one for 2 months & one for the desired number of years, & compare their outputs.
  - If they do not match, we know there is a problem.
  - The fact that they do match is a good thing, but does not guarantee that there is not a problem.
  - The cost of the extra 2 month runs is "tolerable".
  - This would have caught the "Atlas Node 587" issue.

"Atlas Node 587" issue – Further investigation.

- Made some runs using different compiler optimization levels to look at CAM floating point precision issues in general (-O1/-O2).
- Found the level of variation induced by compiler optimization to be minor (~10%).
  - Similar level of variation to that seen in Node 587 runs.
- However, as ensemble results start to be filtered with observational data, these precision variations will become relatively more significant & will need to be reassessed.