

Three Years of Doing Uncertainty Quantification Ensemble Runs at LLNL Using CAM & CESM

17th Annual CESM Workshop
June 19, 2012

John Tannahill

 Lawrence Livermore
National Laboratory

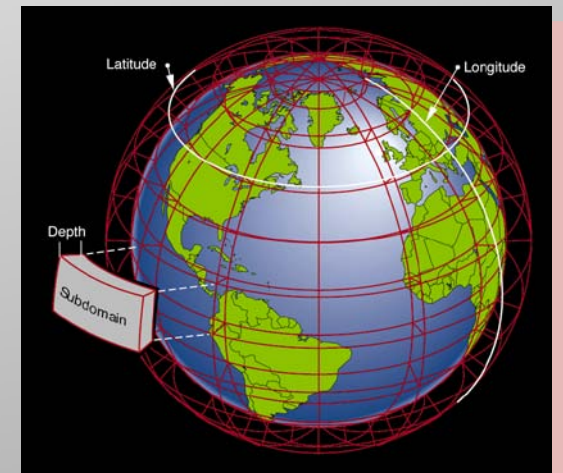


LLNL-PRES-560817

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

The project.

- A 3 year LLNL Strategic Initiative LDRD project, started in October of 2009:
 - *“The Advance of Uncertainty Quantification Science with Application to Climate Modeling, Inertial Confinement Fusion Design, & Stockpile Stewardship Science”*
- Goal is to advance the UQ of multi-physics simulations that utilize large collections of uncertain input parameters.
- PI is Richard Klein.



Project made up of four teams.

- **Climate Team**

Curt Covey

Don Lucas

John Tannahill

Yuying Zhang

- **UQ Pipeline Team**

David Domyancic

Scott Brandon

Rao Nimmakayala

Jim McEnerney

- **Statistics Team**

Gardar Johanneson & others

- **Error Estimation Team**

Carol Woodward & others

CAM/CESM uncertain parameters of interest.

Namelist File	Number of Parameters	
	Started With	Reduced To
atm_in	27	9
drv_in	1	0
ice_in	7	3
Ind_in	1	0
pop2_in	18	10
Totals	54	22

- Established ranges for each parameter.

- Namelist source files included:

- cloud_fraction
- pkg_cldoptics
- pkg_cld_sediment
- cldwat
- hb_diff
- hk_conv
- physpkg
- zm_conv
- shr_flux_mod
- Biogeophysics1Mod
- ice_shortwave
- ice_therm_vertical
- ice_mechred
- hmix_aniso
- hmix_gm
- mix_submeso
- tidal_mixing
- vertical_mix
- vmix_kpp



All CAM/CESM UQ Studies to Date.

Study Type	Number of Runs	Number of Simulated Years	Stored Size (TB)
CAM Atmosphere Data Ocean	3,873	46,476	56
CAM Atmosphere Slab Ocean	823	25,968	47
Data Atmosphere POP2 Ocean	545	8,070	132
CAM Atmosphere POP2 Ocean	327	6,710	128
Totals	5,568	87,224	363

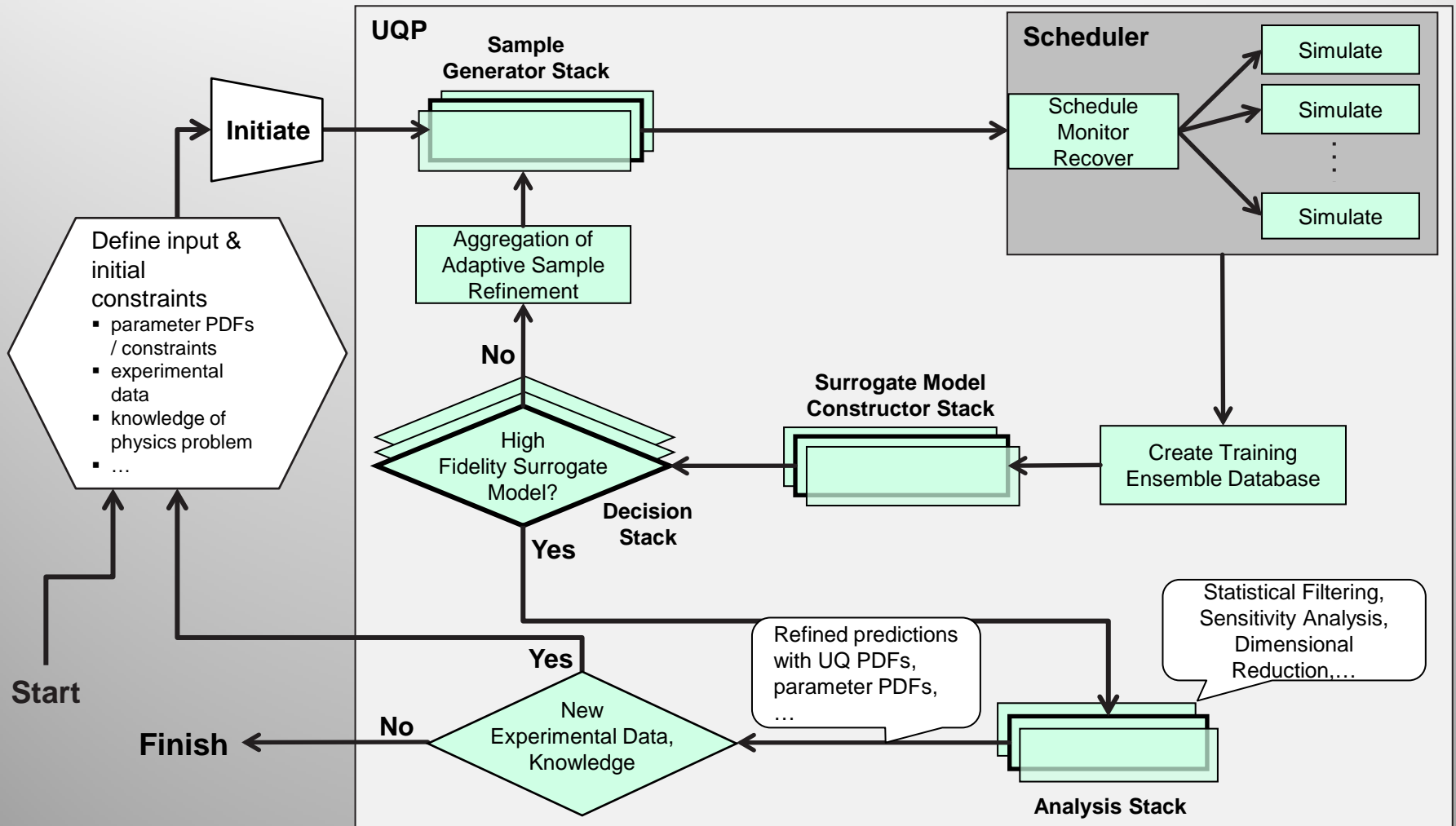
Study particulars.

Study Type	Code Version(s) Used	Resolution	Compset(s) Used	CAM Physics	CAM Dynamics
CAMAtmosphere Data Ocean	CAM3.6.73	1.9x2.5	N/A	CAM3.5.1	FV
	CAM4.1.08 CAM4.1.13	1.9x2.5	N/A	CAM4	FV
	CESM 1.0.1 CESM 1.0.beta22 CESM 1.0.4	1.9x2.5_1.9x2.5	F2000	CAM4	FV
CAMAtmosphere Slab Ocean	CESM 1.0.1 CESM 1.0.beta22	1.9x2.5_gx1v6	E2000	CAM4	FV
Data Atmosphere POP2 Ocean	CESM 1.0.beta22 CESM 1.0.4	1.9x2.5_gx1v6	G_NORMAL_YEAR	N/A	N/A
CAMAtmosphere POP2 Ocean	CESM 1.0.4	1.9x2.5_gx1v6	B_1850-2000_CN B_RCP4.5_CN	CAM4	FV

UQ Methodology: UQ Pipeline.

- Use a s/w tool called the LLNL UQ Pipeline (UQP):
 - Runs, organizes, analyzes, & assesses uncertainties in ensemble-of-models calculations.
- User required to supply two Python interface modules:
 - A UQP/Application Data file:
 - Contains settable data items for UQP & Application.
 - Implemented as a set of Python data dictionaries.
 - An Application Interface file, which must provide four functions:
 - `prep_ensemble()`, `prep_run()`, `post_run()`, & `post_ensemble()`.
- An automated capability to build & run the application is also a prerequisite.

UQP: End-to-end, automated, scientific workflow for UQ analysis.



UQ Methodology:

UQP/CESM Interface module details.

- `prep_ensemble()`:
 - Establishes general CESM framework:
 - Directory structure, namelist templates, copies files, etc.
 - Configures CESM, including processor layout.
 - Builds CESM.
- `prep_run()`:
 - Sets up a specific ensemble run.
 - Updates namelist templates with parameter values from UQP.
- `post_run()`:
 - Computes diagnostics.
 - Stores/cleans up output files.
- `post_ensemble()`:
 - Currently just use UQP default analysis.

UQ Methodology: UQP/CESM Data module example (1).

```
uqdd = {'adjust_code_concurrency': True,  
       'bank': 'uqsci',  
  
       'code_concurrency': 4,          # Number of concurrent runs.  
  
       'max_failed_runs': 4,  
       'setup_depend_job': True,  
  
       'uqmgr_mode': 'batch',        # [batch|debug|interactive]  
       'uqmgr_timeout': 960}        # Job time limit (min.).
```

- Only need to list items that you want to change from their default values.
- Items that a user might change:
 - Brown text: infrequently
 - Red text: fairly often

UQ Methodology: UQP/CESM Data module example (2).

```
aprd = {'exe_name': 'ccsm.exe'}

aprd['params_avail'] = \
    {'cldfrc_rhminh': [0.65, 0.80, 0.85],
     'cldfrc_rhminl': [0.80, 0.91, 0.99],

     'cldopt_rliqice': [8.4, 14.0, 19.6],
     'cldopt_rliqland': [4.8, 8.0, 11.2],
     'cldopt_rliqocean': [8.4, 14.0, 19.6]}

aprd['params_to_use'] = \
    [('cldfrc_rhminl', 'lin 0 0 1'),

    ('cldopt_rliqland', 'lin 0 0 1'),
    ('cldopt_rliqocean', 'lin 0 0 1')]
```

Library of UQ parameters available; with low, default, & high values.

Actual UQ parameters to use in a study; often a subset of available parameters.

- Items that a user might change:
 - Brown text: infrequently
 - Red text: fairly often

UQ Methodology: CESM build/run script.

- Created a standalone comprehensive Python CESM build/run script:
 - Calls the standard CESM scripts.
 - Frees the user from knowing much about CESM.
 - Completely data driven; flexible.
- Contains additional functionality that the Application Interface module uses when conducting UQ studies.
 - Here the UQP does the actual execution of the runs.
 - All the needed data now comes from the UQ Data module.
- Currently script is ~2K lines of Python code.

UQ Methodology: CESM modifications.

- CESM source code:
 - Mods to 22 files to add needed namelist input capability.
- CESM scripts:
 - General mods to machine-dependent CESM script utilities.
 - `ccsm_buildexe.csh` / `ccsm_postrun.csh` / `getTiming.csh`:
 - Commented out all default “gzips” & copying of log files.
 - Changed default handling of timing files.
 - Small mod to `pop2_in_build.csh` to keep pop2 diagnostic files from being overwritten.

UQ Methodology: Putting it all together.

- Determine CESM parameters of interest & their ranges.
 - Add any needed namelist functionality to CESM.
- Implement UQP/CESM Data & Interface modules.
 - Including any post-run diagnostics as desired.
 - Makes use of UQ functionality provided by CESM build/run script.
- For each study:
 - Make any desired changes to UQ Data file settings.
 - Generally not many.
 - Run the study:
 - Pretty much a push-button operation for a study that may contain hundreds of independent runs.

Concluding comments (1).

- Differences in how the various CESM components do their namelist input can be considerable.
- Self-discovery as to how the CESM scripts work “under the covers” is difficult.
- The pop2 ocean model has been somewhat problematic for us:
 - Default output data settings can be quite voluminous:
 - Model crashed when we tried to cut down on this.
 - Appears to be quite sensitive to its parameter settings:
 - Solver fails to converge & floating point exceptions.
 - Initially, we unknowingly over-wrote some diagnostic files.

Concluding comments (2).

- System issues are a major source of problems:
 - Node failures, Lustre disk issues, mpi problems, orphaned system nodes, ...
 - Ran into one quite difficult bug:
 - One core on one node was intermittently flipping the least significant bit during a CESM floating point calculation.
 - Ever since, we have done an additional 2 month run with each longer run & check for a bit-for-bit match.

