Infrastructure Group
Progress and Plans

Rob Jacob and Chengzhu Zhang

(on behalf of the entire Infrastructure Group)

ESMD-E3SM PI Meeting

October 26, 2020

Infrastructure Group responsibilities

- Develop, maintain and support software that is needed for E3SM but is **not** part of the main prognostic models. Configure, build, test, diagnostics, analysis, workflow, driver/coupler

- Manage data sets

- Define, document, manage the process and procedures used in software development within the E3SM Project.

Everything we do should help make the model development, simulation and analysis happen.

E3SM development process

- The system developed for Phase 1 remains in place
  - Make a feature branch (follow naming conventions!); run e3sm_developer for testing; Issue a github Pull Request when finished.
  - Integrator merges to next for integration testing, then master if it passes.
  - Test suites run nightly on several machines.

- Going forward, will encourage **new** developers in ecosystem projects to use Github “forks”.
  - Fork the main E3SM repo to your own github page.
  - Make feature branches on your fork, issue PR from there.
  - Maintain your fork (update master)
  - Invite others to collaborate on your fork if you want.
  - Doesn’t require write permission to https://github.com/E3SM-Project/e3sm
    - 761 branches and 298 developers is too many

“big picture” main model progress

- Still averaging 1 PR merged per calendar day
- V2 developments mostly done
- Exascale Computing Project fork of E3SM frozen and development of CRM now on E3SM main
- GPU code added (OpenMP-offload in MPAS, YAKL in CRM)
- Source code renamed from cam to eam and clm to elm. (Also CIME compsets and root of output filenames changed accordingly)
- “cime” subdir converted from git subtree to git submodule
- Other submodules added:
  - YAKL (for C++ version of CRM),
  - new submodules in MPAS (CVMix, BGC) are “recursive” submodules for E3SM.
- E3SM switched to use SCORPIO (C with Fortran interfaces) instead of SCORPIO-classic (Fortran) as I/O middleware layer. Faster!

“big picture” main model plans

- Remaining BGC and Cryo v2 PRs
  - Additional submodules: GCAM, MARBLE
- Add CF long names to variables used in CMIP6 in all output.
- Creation of v2beta, v2.0.0 tag, maint-2.0 branch
- Start accepting answer-changing v3, v4 developments to existing components
- Introduce SCREAM as an atmosphere component on par with EAM.

Nightly (and more) testing

compy	e3sm_bgcprod_maint-1_1_intel	0	0	0	0	0	0	4
mappy	e3sm_developer_master_gnu	0	0	1	0	0	1	44
mappy	e3sm_developer_next_gnu	0	0	1	0	0	1^{+1}	44^{+44}
sandiatoss3	e3sm_integration_master_intel	0	0	1	0	0	4	83
sandiatoss3	e3sm_integration_next_intel	0	0	1	0	0	4	83
compy	e3sm_integration_next_pgi	0	0	2	0	0	9^{+1}	78_{-1}
cori-knl	e3sm_prod_maint-1_0_intel	0	0	0	0	0	2^{+2}	1_{-2}
anvil	e3sm_prod_next_intel	0	0	0_{-2}	0	0	3	0
cori-knl	e3sm_prod_next_intel	0	0	0	0	0	0	3
compy	e3sm_prod_next_intel	0	0	0	0	0	3	0
sandiatoss3	homme_integration_master_intel	0	0	0	0	0	0	2
sandiatoss3	homme_integration_next_intel	0	0	0	0	0	0	2

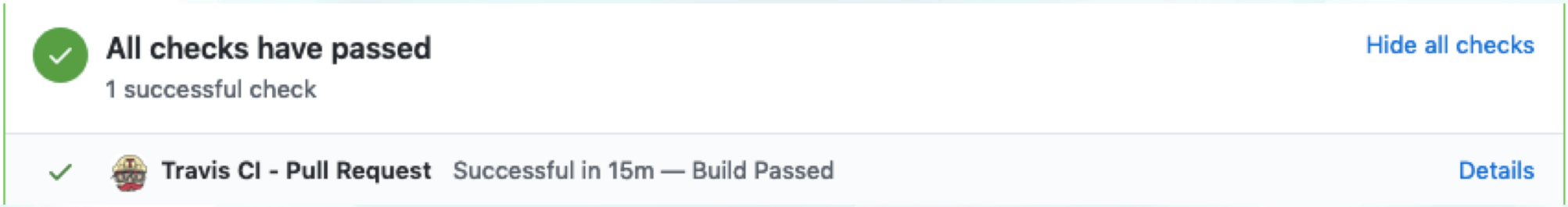
- Standard suites: developer, integration, production. Nightly turnaround.
- System testing: (with baselines) on core set of machines.
- Now testing on compy with pgi (integration) and intel (production)
- **New suites**: `maint-1.1` for v1 BGC cases, `gpu` for gpu-enabled code, `homme` for additional atm dycore testing.

Continue to strike balance among expense of testing/need for overnight results/availability of machine time.

View results at: https://my.cdash.org/index.php?project=E3SM

Recent and upcoming changes to testing

- New Travis-CI testing (a free service from Github): every PR is automatically: merged to master (on a test branch) and built (using gnu) in a fully coupled case.



- Convert all developer, integration, prod testing to use v2 configurations:
  - Theta-l dycore in all EAM cases
  - Use new MPAS-seaice thermodynamic capability (developed by Adrian Turner) to replace CICE in F-cases
  - More tri-grid configs

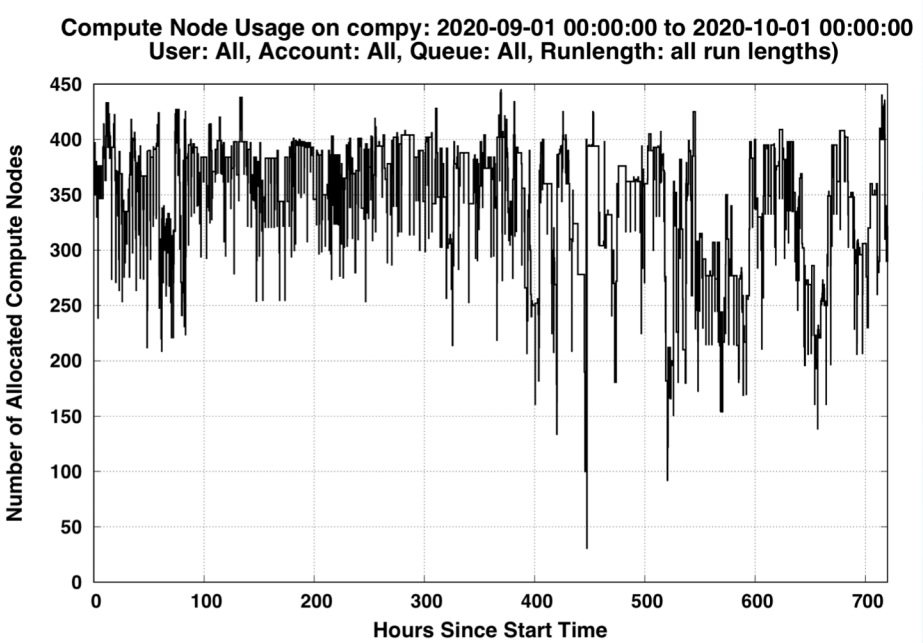
Currently Supported Machines

- Cori-knl at NERSC (the officially supported machine for external users)
  - 9688 Intel Xeon Phi ”Knights Landing” nodes, ERCAP
- Cori-haswell at NERSC
  - 2388 Intel Xeon Haswell nodes, ESCAP
- Compy at PNNL
  - 460 Intel Skylake nodes, Limited to ESMD and RGMA projects
- Theta at ALCF
  - 4392 Intel Xeon Phi “Knights Landing” nodes, INCITE
- Summit at OLCF
  - 4608 IBM Power 9 (2) and NVIDIA V100 (6) nodes, INCITE
- Anvil at ANL
  - 240 Intel Broadwell nodes; Restricted to E3SM SFA.

“supported” means latest master and maintenance branch versions should compile and run.

Compy configuration and use

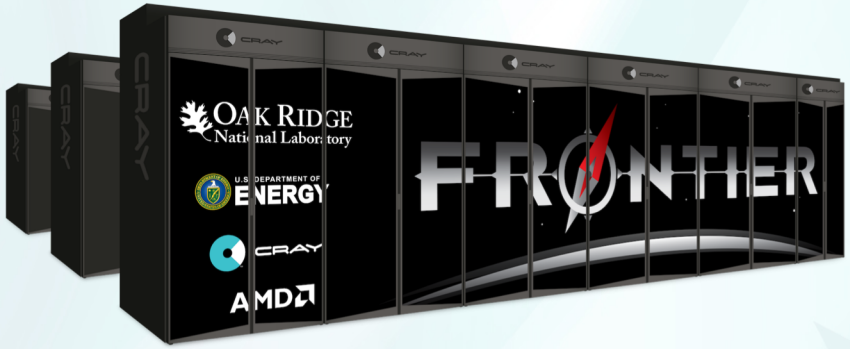
- Job limits: 2 running jobs per user.
- Short queue: 50 nodes set aside for jobs less then 2 hours.
  - Maximum size for a job: 40 nodes
  - If your case fits, it will by default go in short queue.
  - 20 jobs per user (for postprocessing jobs)
  - e3smtest user has higher priority for nightly testing
- 780 TB disk. Stays above 90%
  - Sysadmins monitor usage and tell large users to reduce
- Allocation: 50% E3SM, 35% RGMA, 15% all other ESMD projects
  - PI's should contact program manager for access



- **Be a friendly user: don't run on login nodes (get an interactive node from the queue); monitor disk use**

Upcoming machines

- Chrysalis
  - 512 AMD EPYC nodes; restricted to E3SM SFA
- Perlmutter Phase 1 (NERSC)
  - 1500 CPU-GPU nodes
    - 1 AMD Milan + 4 **NVIDIA** A100 **GPU, 256 GB**
- Perlmutter Phase 2 (NERSC)
  - 3000+ CPU-only nodes
    - 2 AMD Milan per node, 512 GB
    - 84-118PF total (phase 1 and 2)
- Polaris (ALCF)
  - "a CPU/GPU hybrid resource… to prepare and scale codes…on a resource that will look very much like future exascale systems"
- Frontier (OLCF)
  - 1 AMD EPYC + 4 **AMD Radeon GPU;** Exascale
- Aurora (ALCF)
  - 2 Intel Xeon "Sapphire Rapids" + 6 **Intel Xe GPU** Exascale



Another new machine: your laptop/workstation using a container!

- Normally, to build/run E3SM on your laptop, you would have to install compilers and all necessary libraries.
- Instead, install Singularity container software and download the E3SM Singularity container
  - Contains a GNU development environment.
  - Works with your clone of the repo
  - Does not include any input data sets.
- Size of a case is limited by your machine's memory. An ne4 coupled case should run in 32GB.
- Singularity containers can be used on HPC platforms (Theta, Cori, Compy) unlike Docker containers.

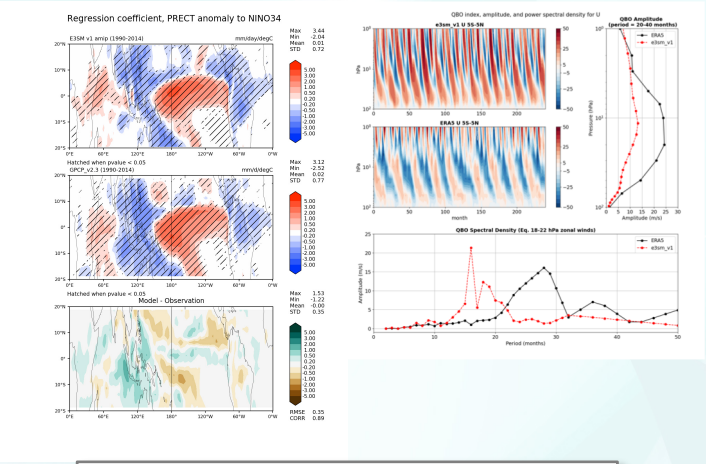
https://esgf.anl.gov/monitor/e3sm.sif created by Lukasz Lacinski

E3SM IG software covers all phases of the workflow

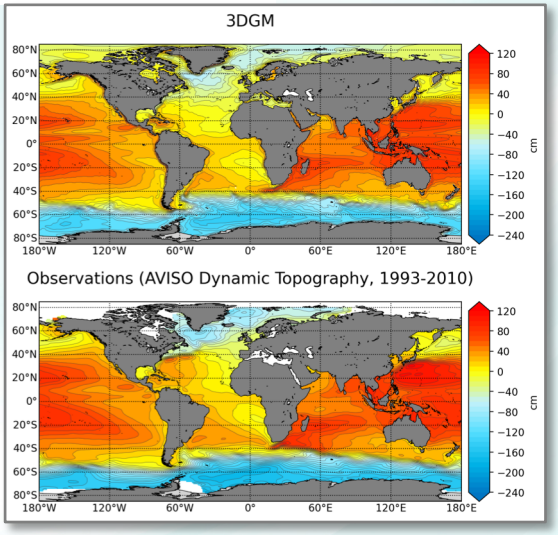
- Before the model run:
  - Zoo of programs to create input data, grids, maps. Have brought some under test. Working to document.
  - Configure and build with CIME Case Control System (jointly developed with CESM; python, cmake)
- Running the model:
  - Submit with CIME Case Control System (python interface with slurm, others)
  - Top level driver/coupler: cp7/MCT, cpl7/MOAB
  - parallel I/O library: SCORPIO
- Immediately after the run:
  - Restart, short-term archive with CCS
  - Archive to disk with zstash
- Diagnostics and Analysis
  - NCO
  - e3sm_diags, MPAS-Analysis, Aprime
- Data Publication
  - e3sm_to_cmip
  - data_checker.py

New capabilities in Diagnostic/Analysis tools (since last PI meeting)

- E3sm_diags: Capability to use monthly time series data (NCO format or CMIP like), ENSO and QBO evaluations. Land and River components analysis: Runoff and Streamflow evaluation. Diurnal Cycle of precipitation Capability to process sub-monthly output.



- MPAS-Analysis: 12 releases, Added 14 new types of analysis, for 39 total types of analysis, Analysis output available for all E3SM v1 simulations, Diagnostics used to debug, validate and tune E3SM v2



New capabilities in Diagnostic/Analysis tools (since last PI meeting)

- NCO: New vertical interpolation options, Improved Parallelization, Global mean timeseries from splitter, High-frequency (i.e., resolves diurnal cycle) climos, timeseries

- Zstash: Developed from a prototype to a production software and now used as standard long term archive tool

- Continue to get all of the above through “e3sm_unified” conda package.
- Tutorials on tools (incl. youtube videos) produced for broader community use.

(partial) Plans for Diagnostics/Analysis tools

- E3SM_diags: ARM data-oriented diagnostics, TC analysis Stratospheric ozone diagnostics, Dust aerosol, Precipitation intensity Atmospheric CO2 diagnostics/metrics, Key land variables

- MPAS-Analysis: Add node parallelism using parsl, show transects on the native MPAS-Ocean mesh, create developers guide for adding new analysis.

- NCO: support MPAS-Iandice output, Remap ELM output fields stored in sparse array format, ncremap supports mbtempest, ncclimo takes ncremap options

- Other: settle on one workflow management tool. Expand provenance capabilities of PACE.

More info in Poster Session 1, D4S1- Breakout #3 and Tools Talk (Thursday plenary).

v1 Data Publication Progress (as of Oct 24th 2020)

CMIP6

- 17 simulations
- 105 variables per simulation
- 1808 datasets
- 39,550 files
- 6.55TB of data

Standard

- 38 simulations
- 356 datasets
- 510, 471 files
- 360TB of data

ESGF
Earth System Grid Federation

Publication has been expanded to include 3hr, 6hr, and daily files in addition to the previously published monthly datasets. All E3SM project publications include 65 regridded time-series variables as well as seasonal climatologies.

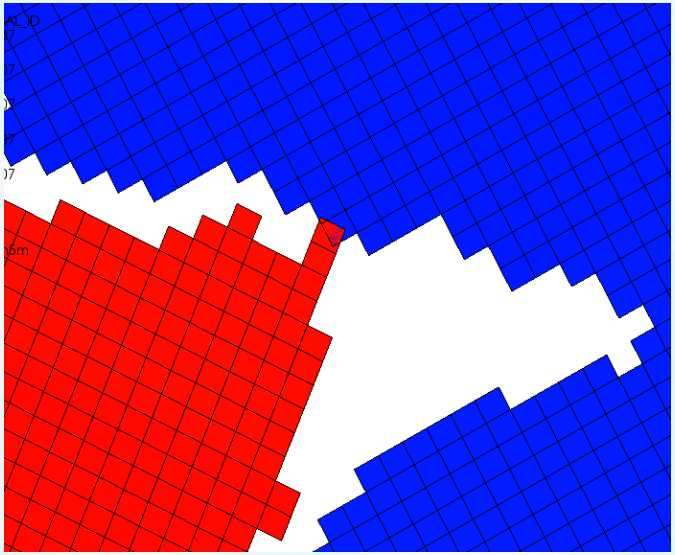
Publications include BGC, Cryosphere, WC DECK, and HighResMIP simulation campaigns.

Revamped and Streamlined Publication Processing

- Implemented spinning-disk E3SM Archive (at LLNL) to unify data access (~1PB on-hand)
  - Manual retrieval of NERSC HPSS zstashed archives conducted “up front”
  - Variant archive structures and content “mapped” for automated access
- Processing for CMIP submission
  - **Published ~7TB to CMIP6 so far**
  - CMIP6 processing is now (mostly) automated and parallelized
    - New “e3sm_to_cmip” script.
  - Support Ecosystem and external projects on CMIP processing/publication
- Supports “On Demand” publication of well-prepared datasets
- V2 datasets will be published much faster (once cleared).

MOAB-based coupler and offline tools

- Developed offline “mbtempest” tool for generating maps
  - Allows data-parallel versions of TempestRemap algorithms. Same code used in MOAB-coupler
  - Introduced new intersection algorithm for meshes with holes.
  - Examined several problems with HYDRO1K mesh while trying to make a map from 10min to ne1024. (problems found by C. Zender)



- Got MOAB-coupler to send from 2 different atm grids (SE and physics) to coupler (map to land and ocean). Have now redone this for tri-grid. MOAB-coupler will be an option in v2.

Infrastructure Group works for YOU

- IG should be working on tools **you** want/need to use

- If there are **any** problems with IG software, always FILE AN ISSUE.
  - Only way to let others experiencing the problem to know its been reported.
  - Group leads can prioritize the work and track progress.
  - E3SM Documentation will have links to each package's github page.

IG schedule this week

- Today: this talk. Also see related talk in D1S4: “NGD Software and Algorithms”
- Tuesday: Poster Session 1
  - “E3SM Ocean and Sea-ice Diagnostics with MPAS-Analysis” – Xylar Asay-Davis
  - “Zstash v0.4.2: HPSS Long-Term Archiving Tool” – Ryan Forsyth
  - “Introduction to E3SM Diagnostics Package (e3sm_diags v2)” – Jill Zhang
- Wednesday:
  - related talk in D3S1: “Coupling Approaches for Next Generation Architectures (CANGA)”
  - D3S4 – Breakout #3 “Computational Science”
- Thursday:
  - D4S1 – Breakout #4 “Infrastructure + NGD Software and Algorithms”
    - ‘Converting E3SM model output to the CMIP6 data standard’ – Sterling Baldwin
    - Discussion
  - D4S3 – E3SM Tools overview