E3SM Infrastructure Group Update

Rob Jacob (reporting on the work of many people)

2019 CESM Workshop
Software Engineering Working Group Meeting
June 19, 2019
E3SM IG 3 major areas of concern

- Develop, maintain and support software that is needed for E3SM but is not part of the main prognostic models. CIME, e3sm_diags, MPAS-Analysis, processflow.

- Manage data sets (ESGF publication, CMIP6 conversion)

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

- IG was new group in E3SM Phase 2 which started July 1, 2018.
Testing

- System testing with CIME create_test continues (with baselines) on core set of machines.
- Integration and testing process running (mostly) smoothly.
- Slowly adding tests for new capabilities.
- Continue to strike balance among expense of testing/need for overnight results/availability of machine time.
Model Development

- Roughly 10 PRs a week getting merged to master.
  - Accepting changes for E3SM 2.0 (and further versions if appropriate). Feature freeze on June 30th.
  - Follow along at [https://github.com/E3SM-Project/E3SM](https://github.com/E3SM-Project/E3SM)

- Maint-1.0 branch – stable. fixed a bug for reading DECK v1 restart sets, adding SSP compsets for CMIP6 runs.
  - Created April 20, 2018

- Maint-1.1 branch – in use for v1 BGC simulations
  - Created Aug 2, 2018
IG Code releases

(and very brief list of features)

• NCO 4.7.8, 4.7.9, 4.8.0: CMIP6 support including vertical interpolation

• MPAS-Analysis 1.2.3: surface BGC, transect plots, iceberg concentration

• E3SM-diags 1.7.0: save more provenance, analysis of time-series files.

• Processflow 2.2.0: call MPAS-Analysis, more debugging.
Data update

• V1 DECK publication
  – All data from our DECK sims so far has been published on ESGF in non-CMIP6 “raw” format.
  – Will now focus on CMOR-ization of data for CMIP6.
  – “e3sm_to_cmip” python script handles conversion to CMIP6 format.
    • [https://github.com/E3SM-Project/e3sm_to_cmip](https://github.com/E3SM-Project/e3sm_to_cmip)

• Found some data corruption on data retrieved from tape at NERSC HPSS. Had to re-run a few sims for some periods.
Machine changes

- NERSC “Edison” is no more. “Cori-KNL” is now the single supported machine for external users of E3SM.

- OLCF Summit only used for ECP “E3SM-MMF” project. No regular testing by E3SM. Titan goes away next month (we stopped testing a while ago).

- ALCF Theta (and Cori-KNL) used for ongoing high-resolution simulations. Mira not used.

- New E3SM-only machine: “CompyMcNodeFace” or just “Compy”.
  - 460 dual-socket Intel Skylake nodes; 40 cores/node, 192 GB/node, Hosted at PNNL.
  - Intel OmniPath interconnect, 1 PB Lustre filesystem
  - 50% E3SM, 35% RGMA, 15% ESM other.
  - Model is ported. Finding balanced pe-layouts.
  - Intel-MPI is MUCH faster (2x) then mvapich2 on this machine for some cases. Still investigating why
MOAB Coupler status at last workshop

- Can compile and build E3SM that
  - Instantiates MOAB
  - Defines full meshes in HOMME and MPAS
  - Migrates full meshes to coupler
  - Computes intersection in coupler
  - Computes weights
  - Migrates data from atmosphere to coupler
  - Applies weights
  - Migrates regridded data to ocean
MOAB Coupler status now

- Can compile and build E3SM that
  - Instantiates MOAB
  - Defines full meshes in HOMME and MPAS and ELM
  - Migrates full meshes to coupler
  - Computes intersection in coupler
  - Computes weights
  - Migrates data from atmosphere to coupler
  - Applies weights
  - Migrates regridded data to ocean
  - Migrates data from land to coupler
  - Copy data from atmosphere to land
  - Define mesh in MOSART
  - Migrate data from river to coupler
  - Compute l2r and r2o weights
  - Apply weights for l2r and r2o
MOAB Coupler status

• More info: http://sigma.mcs.anl.gov/moab-library/

• Now in http://github.com/ESMCI/cime

cime/src/drivers/mct/
cime/src/drivers/moab/
Cime/src/drivers/nuopc

• See branch “sarich/use-moab-driver” in http://github.com/E3SM-Project/E3SM for details
Diagnostics future work:

- Recently used 5 of regular IG telecon times to do overview of our major python codes
  - Typical use, documentation, output, code structure, testing, dependencies
  - MPAS-Analysis, E3SM-diags, processflow, LIVVkit, CIME CCS
  - Will inform discussion on common standards for our python diagnostic codes and future development plans.

- Currently working to unify observation data used by analysis tools.

- More use of containers for testing and deployment.
Containers in E3SM

• Container for e3sm_diags available with documentation.

• Container for e3sm_unified (a python conda environment with multiple diagnostic tools) used for Travis nightly build on github.
  – Travis is building e3sm-diags in a Singularity container, and then ensuring that MPAS-Analysis and processflow help commands work. Catches any build errors and any issues with upstream dependencies.
  – constrained by the Travis CI system limits like memory, disk, etc.

• Container for E3SM development being worked on.

• Containers are supported at NERSC (Shifter), ALCF, OLCF (Singularity)
Building E3SM with Cmake  (Jim Foucar, SNL)

<table>
<thead>
<tr>
<th>Setup</th>
<th>Build</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case_setup</td>
<td>Case_build</td>
</tr>
<tr>
<td>* BuildTools.configure()</td>
<td>Components/CMakeLists.txt</td>
</tr>
<tr>
<td>* Create_namelists</td>
<td></td>
</tr>
<tr>
<td>* For each component:</td>
<td></td>
</tr>
<tr>
<td>$comp/cime_config/buildnl</td>
<td></td>
</tr>
<tr>
<td>$comp/bld/configure</td>
<td></td>
</tr>
</tbody>
</table>

- Case.setup pieces now create info for Cmake.
- Case.build calls Cmake-generated Makefiles.
- Cmake won’t be asked to handle all the logic in giant perl “configure” scripts. Will convert those to python.
- Done for E3SM atmosphere and land so far (on master)
- Expect to see build speed-up when complete.
E3SM Next Generation Development

• E3SM Phase 2 has
  – Core groups: Water Cycle, BGC, Cryosphere, Infrastructure, Performance
  – many NGDs: land, atmosphere, coastal ocean, and software.
    • NGDs targeting v3 or later versions.

Will comment on:
• NGD Software and Algorithms (managed by Andy Salinger)
• NGD Non hydrostatic atmosphere SCREAM (managed by Peter Caldwell)
Update on E3SM NGD-Software

• Developing tests for individual physics pieces. MAM. Python notebooks to share verification. “cmdv-test-runner” will run one or several tests in code and report results.

• New system tests in CIME for detecting climate changes: NGP (perturbation growth), TSC (time step convergence). Requires additional python packages.

• Machine-aware task placement with Zoltan2: experiments being conducted with MOAB-based coupler.

• Also: IMEX methods, MPAS-ocean barotropic solver, HOMME SL transport, chemistry nonlinear solver.
Background/Update on NGD Non-Hydrostatic Atm (SCREAM)

- 3 ways to program GPUs
  - Special language for kernels (Cuda for NVIDIA, HIP for AMD, SYCL for Intel)
  - Directives in your regular Fortran code: OpenACC (NVIDIA-only), OpenMP (multiple but support lagging)
  - C++ and templated programming. Kokkos (Sandia) Raja/Umpire (LLNL). Compiler backend provides code customized for the specific GPU. C++ only.
  - In all cases, the GPU needs a LOT of work to be useful

- SCREAM is building a 3km atmosphere using C++/Kokkos
  - Converting physics pieces from Fortran to C++. SHOC, P3.
  - Also building unit tests as they go.
  - Estimate: 1 SYPD on half of Summit (3km global provides plenty of work)
  - Progress:
    - Several unit tests already in regular operation.
    - Can run 6km EAM with full physics; Can run 3km HOMME with dry dynamics.
Thank you!

**E3SM Infrastructure Group:** Xylar Assay-Davis, Sterling Baldwin, Jim Foucar, Rob Jacob, Jayesh Krishna, Joe Kennedy, Lukasz Lacinski, Salil Mahajan, Bibi Mathew, Jason Sarich, Zeshawn Shaheen, Chengzhu Zhang, Wade Burgess, Iulian Grindeanu, Vijay Mahadevan, Rick Wagner