ACME SE Update

Robert Jacob

June 21, 2016
21st CESM Workshop
SEWG Meeting
Breckenridge, CO
Establishing, maintaining, and improving the software engineering tools, processes and designs used in ACME.

- **Repository**
  - Tools: source code control system, software access.
  - Processes and designs: code development workflow, code review

- **Testing**
  - Tools: automated testing, test suites
  - Processes and designs: continuous integration testing, system and unit testing procedures

- **Productivity**
  - Tools: wiki’s and task-tracking tools.
  - Processes and designs: bug tracking, build system, programming standards
ACME SE/CPL Group role (2 of 2)

Develop and maintain inter-model infrastructure and top-level architecture of the ACME model.

- Coupler, main driver, other inter-model architecture
  - Modular interfaces and configurability of model
  - “shr” code
  - Model Coupling Toolkit

- I/O layers
  - Parallel I/O (PIO) sub-system
  - Component-level I/O systems
The ACME SE/CPL Group

Hub

- Jed Brown (ANL)
- Wade Burgess (SNL)
- Jim Foucar (SNL)
- Robert Jacob (ANL)
- Jeff Johnson (LBNL)
- Jayesh Krishna (ANL)
- Andy Salinger (SNL)
- Jason Sarich (ANL)
- Andreas Wilke (ANL)

Spokes

- ATM: Balwinder Singh, Susannah Burrows
- LND: Gautum Bisht
- OCN/ICE: Doug Jacobsen

Machine POCs

- NERSC: Noel Keen
- OLCF: Matt Norman
- ALCF: Jayesh Krishna

Notable contributions from: Pat Worley, Az Mametjanov (Perf group), Renata McCoy (project engineer)
Biggest development since last workshop: ACME adopts CIME!

- We were replacing the cesm1_3beta10 script system
- July 30, 2015: First attempt. Oh you also have to modify the individual model’s build systems!
- August 4, 2015: CIME2.0 tagged by CSEG
- August 31, 2015: Second attempt. Lots of work to merge ACME’s developments on the old script system with CIME2.
- Sept 15, 2015: CIME3.0 tagged by CSEG
- Sept 17, 2015: PR issued for CIME2.0.14 in ACME
- Oct 2, 2015: PR closed, CIME2 merged to ACME master. CIME4.0 tagged by CSEG.
Repository: ACME uses git and github for all source code (no svn)

- [http://github.com/ACME-Climate/](http://github.com/ACME-Climate/)
- Entire code base can be checked out with one “git clone” command.
- Some code parts brought in as git “externals”: submodules or subtrees (from other git repos).
- Development since start:
  - 500+ commits  257 branches  49 contributors out of 146 github developers
  - Doesn’t count commits/developers on externals.
- Code Development Workflow: Feature branches; Pull Requests; Staging on Integration branch “next”, Merge to “master”; Maintained Release branches “maint-v1.0”
SE/CPL Repository highlight: Integration of ACME v1 feature branches

- 31 branches integrated to next and master Nov 1-Jan 14
- ACME v1.0.0-alpha.1 tagged on Jan 18th
- v1 alpha testing ongoing.
- Current tag: v1.0.0-alpha.6, May 20th
SE/CPL Group Progress -- Testing

• Testing strategy:
  – acme_developer: <1hr on any platform; required for pull request
  – acme-integration: < overnight on all platforms; required for integration
  – Jenkins-based system to launch testing jobs from single server
  – Cdash dashboard for presenting results
    • http://my.cdash.org/index.php?project=ACME_Climate

• Progress:
  – Test suites developed – adding more all the time
    • acme_developer: 26 tests -- 5 machines
    • acme_integration: 56 tests -- 3 machines
  – Distinction of namelist changes versus answer change on dashboard.
  – python version of create_test developed by Jim Foucar and in use starting Oct, 2015. Test lists are a python “dictionary”
  – Added more machines for running suite with Jenkins.
<table>
<thead>
<tr>
<th>Site</th>
<th>Build Name</th>
<th>Not Run</th>
<th>Fail</th>
<th>Pass</th>
<th>Time</th>
<th>Build Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>melvin</td>
<td>acme_developer_master_gnu</td>
<td>0</td>
<td>0</td>
<td>26</td>
<td>0s</td>
<td>18 hours ago</td>
</tr>
<tr>
<td>melvin</td>
<td>acme_developer_next_gnu</td>
<td>0</td>
<td>0</td>
<td>26</td>
<td>0s</td>
<td>21 hours ago</td>
</tr>
<tr>
<td>cetus</td>
<td>acme_developer_next_ibm</td>
<td>0</td>
<td>5</td>
<td>21</td>
<td>0s</td>
<td>58 minutes ago</td>
</tr>
<tr>
<td>edison</td>
<td>acme_developer_next_intel</td>
<td>0</td>
<td>2</td>
<td>24</td>
<td>0s</td>
<td>Jun 17, 2016 - 23:46 EDT</td>
</tr>
<tr>
<td>corip1</td>
<td>acme_developer_next_intel</td>
<td>0</td>
<td>2</td>
<td>24</td>
<td>0s</td>
<td>Jun 11, 2016 - 07:42 EDT</td>
</tr>
<tr>
<td>redsky</td>
<td>acme_integration_master_intel</td>
<td>0</td>
<td>2</td>
<td>54</td>
<td>0s</td>
<td>Jun 18, 2016 - 05:39 EDT</td>
</tr>
<tr>
<td>blues</td>
<td>acme_integration_next_gnu</td>
<td>0</td>
<td>6</td>
<td>50</td>
<td>0s</td>
<td>Jun 16, 2016 - 23:56 EDT</td>
</tr>
<tr>
<td>skybridge</td>
<td>acme_integration_next_intel</td>
<td>10</td>
<td>1</td>
<td>45</td>
<td>0s</td>
<td>Jun 19, 2016 - 10:42 EDT</td>
</tr>
<tr>
<td>melvin.sandia.gov</td>
<td>acme_scripts_regression</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>11m 24s</td>
<td>5 hours ago</td>
</tr>
</tbody>
</table>
SE/CPL Group Progress – Collaboration Tools, Licensing

• Continuing to make use of Confluence, Gotomeeting, Github issues.
• Still trying to integrate JIRA for task tracking.
• New tool: Slack chat service. Excellent at providing connectivity to a physically distributed team

ACME Licensing plans (draft)
• Modified-BSD (common, low-hassle Open Source license)
• Release from ACME-Climate GitHub organization
SE/CPL Group Progress – PIO

- New runtime configuration options
- Continuing to find optimum settings for different machines/cases.
- Can switch ACME between PIO1 and PIO2 for testing

Coupler and Configurability/Modularity

- Mostly deferred
- Coupler support for subgrid orography and dynamic ice sheets starting.
SE/CPL Group – Plans 1

• Get on manifold of 100% tests passing on all targeted machines – and stay there
  – Machine POCs have had tough job, large learning curve

• Make sure all features are covered by tests
  – Continue to educate developers on adding tests
  – Restart work on Unit Testing infrastructure

• Speed up configure/build for comprehensive tests:
  – Parallel build (make –j 40)
  – Reuse of compile components for several executables
  – Reuse same executable for many tests (run time configurability)
SE/CPL Group – Plans 2

- Refactor of configure/build/testing scripts as part of CIME5 development
  - All Python – get rid of csh and perl
  - Configure/build more condensed, modular, built in error checking, fully configurable by XML
  - Stand-alone testing of test scripts
    - Parallel builds of tests
    - Bundling of tests.

U.S. DEPARTMENT OF ENERGY
SE/CPL Group – Plans 3

• Repository:
  – Will soon need to manage release branch “maint-v1.0” as well as master
    • Will take some training and documentation
  – Support developers

• Machines:
  – Continue to support changing machines, compilers, queue systems
  – Support developers

• Collaboration tools:
  – Need a simple, unified JIRA workflow across all ACME groups

• PIO: Switch to PIO2

• Coupler: Performance tuning, elevation classes
ACME-SM Proposal to CMDV-SE will accelerate and expand several of these activities, if funded (currently in review):

- Build system upgrade (try Cmake everywhere)
- Unit Tests (expand to component models)
- Climate Reproducibility tests (several methods: perurb growth, multivar)
- Verification of Atm Physics (is code doing what you thought it should do?)
- Single Column Model development (dycore independent)
- Refactor of Atm Phys and Dynamics driver (physics and dynamics in parallel)
- Next-Gen Coupler (develop MOAB-based coupler)
- SE Dycore using Trilinos/Kokkos/C++ (maximize use of DOE math/CS libraries)
- SE Education (on-site training in git, TDD, development workflow)
Other ACME SE/CPL Progress: MCT Release!

- At last meeting (June 16, 2015) promised “MCT 2.9 out tomorrow!” (June 17)
- MCT 2.9.0 actually released on June 18!
More information

http://climatemodeling.science.energy.gov/projects/accelerated-climate-modeling-energy