CCSM Performance Timer Unification

CCSM Performance Timer Unification

Summary: This activity would modify the way timings are collected and reported within CCSM to provide more accurate and consistent timings from each component. This is based on the timers currently in use in CAM and CLM which use the PAPI timer library.

Stage 1:

Derive requirements. Review with each component liaison.

Stage 2:

Identify all events for which timers will be reported. Convert all requirements to concrete design details.

Stage 3:

Implement changes into CAM/CLM and CAM/CLM related scripts.

Stage 4:

Implement changes into each of the other components one at a time.

General Requirements:

[1] Consistent (portable) manner of collecting performance statistics.

[2] Consistent set of reported performance statistics.

Specific Candidate Requirements:

[1] All components shall use a PAPI based timer library which may be the current timers used by CAM and CLM.

[2] A build option shall exist for machines which do not have a PAPI option.

[3] Each component shall report performance statistics for each processor.

[4] Each component processor shall report minimum, maximum, and mean values for identified key timers.

[5] Diffferent levels of timers shall be on or off based on a runtime flag set in to run_conf file.

[6] Different levels of timers shall be on or off bases on a build time flag set in env_mach.<machine> or Macros.<arch>.<machine> (TBD)

Issues:

[1] Shall PAPI be included within the CCSM distribution or shall users be required to provide one for the target machine from the PAPI support web site? If CCSM distributes the code, then CCSM has control over a more complete configuration and can provide more complete testing for supported machines. If PAPI is not included in the CCSM distribution then update/corrections/enhancements to PAPI may require more work for the CCSM staff.

Existing Timing Packages:

CAM-CLM (GPTL-based) Timers