CScADS June 29, 2012
Performance Tools for Extreme-scale Computing

Out-briefing:
Variable Clock Rates/HWC Event Validation

Marty Itzkowitz, Dave Levinthal, Dan Terpstra, Stéphane Eranian, Vince Weaver, Robert Richter, Dave Goodell
Variable Clock Rate I

• Problem is here to stay, and we need to cope with it.
• Intel's PTU recorded a timestamp with each event
  – For two successive samples on a core:
    • Get tick counts of variable clock & TSC counter (reference clock)
    • Ratio gets frequency
• One TSC per core; each hyperthread gets its own TSC
  – Ambiguity about TSC correlation across cores
  – Especially in C-states in older chips
  – Newer chips may or may not be better in this respect
Variable Clock Rate, II

- Both Intel & AMD have a really-fixed frequency counter
  - Independent of turbo mode and frequency scaling
- PAPI intends to provide ratio:
  - Reference (fixed) clock
  - Variable pipeline clock, as seen by the process
- Question -- what really matters?
  - Is it amount of work done in that number of cycles?
  - Or is it elapsed real time?
Variable Clock Rate, III

• One methodology
  – Use both fixed (MPERF) and variable (APERF) counters
  – Record both in signal handler
  – Use them to compute instantaneous clock

• On Intel, Bit 38, MSR1A0 (probably) controls turbo mode
  – AMD has similar technique
  – Alas, no such hook for frequency scaling
    • Only doable in BIOS at boot time
HWC Event Validation, I

• Dave Levinthal's Law:
  – "Open-ended, never-ending pain -- the the sad reality of it all"

• Need well-defined set of computational kernels
  – Predict event counts, then measure for each chip
  – Gooda has small set of kernels, available under Apache license
  – Dave Levinthal would welcome additional kernels
    • As long as they can be under same license.

• Validation across hyperthreading is even more difficult
  – Hyperthreads interfere with each other, in ways not well-understood
  – Cross-pollution of counts, possibly due to HW race conditions
HWC Event Validation, II

• DL's Historical observation:
  – Intel tock (new architecture) chips tend to have problems
  – Intel tick often fix them
  – Westmere seems to have best coverage of any processor

• Discussion was mostly on x86; issues are generic
  – Apply to all architectures, ARM, Power, SPARC...
  – Same issues for HW counters on GPUs
HWC Event Validation, III

- It would be good to organize effort to collect large apps
  - Calibrate counting rates for both 1 and 2 threads per-core
- How wide is the interest in doing this?
  - Is code.google.com a reasonable place to manage it?