Compiler Project

CERN openlab II quarterly review

20 September 2006

Sverre Jarp

History

- Project is active since openlab I
  - Initial drive
    - Improve benchmarks from HEP C++ frameworks (such as ROOT, GEANT4, etc.) on Itanium
  - The problem is that our execution profiles are very flat
    - No unique “hot” routine or loop
  - Project has now taken a broader view
    - Try to influence SPEC to increase focus on C++
    - Provide HEP snippets as focus on generic compiler issues
    - Continue benchmarking, but focus on both Xeon/64-bit and Itanium

SPEC 2006

- We submitted both GEANT4 and ROOT to new suite
  - Both rejected!
    - GEANT4
      - Individual results too dependent on FLP accuracy
    - ROOT/stress
      - Too I/O intensive
- Good news, nevertheless
  - 7 C++ programs present in suite (~25%)
    - 3 (out of 12) in SPECint, 4 (out of 17) in SPECfp
- Also: openlab has acquired the benchmark CD
  - Will use SPECint2006 for testing new systems
    - Together with FIO

Recent activities

- Summer student achievements
  - More snippets
    - 15 in total from CLHEP, GEANT4, and ROOT
  - Renormalization of timing
    - Norm: 20-25 s on Woodcrest @ 2.67 GHz
  - Improved correctness checking and cycle distribution
  - Timings with most recent compilers: gcc4 and icc9.1
  - Report being finalized
- In parallel
  - In-depth compiler paper (S.J.)
    - Plan to publish short version in CERN Courier
- Evangelization
    - Performance talks at Hepix (Rome, Apr06) and Gelato (SanJose, May06)
    - Keynote speech at “Journées Informatiques IN2P3” (two days ago)

Usefulness of snippets

- Snippets chosen to highlight one (or a few) compiler features
  - Such as:
    - memory disambiguation
    - inlining
    - loop unrolling
    - etc.
  - Excellent for compiler critique and regression testing

- Not really intended as benchmarks
  - Too small, too cache friendly, etc.
  - Nevertheless, Core 2 Duo gives outstanding results across all snippets
    - Little variation when altering compiler options
    - icc 9.1 gets good competition from gcc 4.2.0
      - but not gcc 4.1.1

Outlook

- Continue efforts to
  - Evangelize inside our community
  - Expand snippets and use them for consistent regression testing
  - Maintain close collaboration with compiler teams
    - Intel, Santa Clara: focus on forthcoming 10.0
    - GNU community: focus on 4.3.0
    - Open64 1.0 (Gelato community & HP Labs)
  - Understand compiler implications of many-core
  - Monitor SPEC2006 C++ optimization