Mini-App Effort in Japan

Naoya Maruyama

RIKEN Advanced Institute for Computational Science

nmaruyama at riken.jp

SC13 BoF: Library of Mini-Applications for Exascale
Component-Based Performance Modeling

京
K computer

Background

- Ongoing Japanese Exascale projects
  - Funded by the MEXT "Feasibility Study" projects
    - Architecture studies exploring key technology components for the exascale era
    - Application studies exploring potential impacts by exascale applications to sciences
    - 7/2012 – 3/2014
  - (Hopefully) will be continuously funded by follow-up projects
- Collaborative projects among Japanese computational scientists
  - Co-PIs: Hirofumi Tomita (RIKEN AICS), Satoshi Matsuoka (Tokyo Tech)

Mini-Apps

- MD
  - Two variants: FFT-based and FMM-based ones ("Marble" and "Modylas", respectively)
- QCD
  - "CCS-QCD"
- CFD
  - "FFVC"
- Genome sequence matching
  - "NGS Analyzer"
  - I/O intensive

Under Development

- First principles density functional theory
  - "CONQUEST"
- Quantum Monte Carlo
  - "ALPS/looper"
- Climate simulations
  - Spectral and icosahedral models ("NICAM" and "DCPAM")

Will be available around Spring 2014

Example: Molecular Dynamics

- Two alternative algorithms for solving equivalent problems
  - Particle Mesh Ewald
    - Bottlenecked by all-to-all communications at scale
    - Example implementation: MARBLE (Ikeguchi et al.)
  - Fast Multipole Method
    - Tree-based problem formulation with no all-to-all communications
    - Example implementation: MODYLAS (Okazaki et al.)
- Allows algorithmic comparisons
  - FFT vs. FMM?

Example: Molecular Dynamics

- Simplified problem settings
  - Only simulates water molecules in the NVE setting
  - Can reduce the codebase significantly
  - Easier to create input data sets of different scales
  - Whether it's sufficient is still under discussions
- Kernels: Pairwise force calculation + Long-range updates (FFT or FMM)
- Two reference implementations to study performance implications by algorithmic differences
  - MARBLE (20K SLOC)
  - MODYLAS (16K SLOC)

Mini-App Example: QCD

- Based on a benchmark code developed at University of Tsukuba
- Very compact codebase: 3K SLOC
- Main kernel: BiCGStab

Discussions

- Characteristics you expect from a mini application
  - Reflects full-scale application performance characteristics
  - Captures end-to-end application behavior with manageable codebase
    - Unlikely to be a single-loop kernel
  - Downloadable source code
  - Performance models for explorative performance studies
- Features to have the impact in the community
  - Allows for application-level performance studies
    - Used to explore architecture parameters in the Exascale Feasibility Study projects
- Gaps that you can identify
  - Mechanism to check whether mini-apps accurately reflect original full-scale apps
  - No comprehensive, permanent performance database

Performance Data Repository

- Database to store:
  - Performance results
  - Performance models
  - Application and system information (meta data)
- Allows for comparisons across machines and applications
- Allows for “what-if” studies with performance models
- Ongoing collaborative projects between Tokyo Tech and ORNL