Center for Scalable Application Development Software: Center Summary and Plans

John Mellor-Crummey (Rice)
Center Organization

• World class team of researchers; decades of HPC experience
  – Pete Beckman - ALCF director; Senior Fellow CI, Univ. Chicago
  – Kathy Yelick - NERSC division director; Professor UC Berkeley
  – Rusty Lusk - MCS Division director; Distinguished Fellow ANL
  – Jack Dongarra – Univ. Distinguished Professor UTK; Member NAE; Director Innovative Computing Lab; Fellow of AAAS, ACM, IEEE, SIAM
  – Keith Cooper – Professor Rice; ACM Fellow
  – Bart Miller – Professor UW Madison; ACM Fellow
  – John Mellor-Crummey – Professor at Rice

• Management
  – coordinate center-wide activities telecons in regular time slot
  – most vertically integrated work is coordinated in pairs and groups
  – face to face discussions at workshops and national meetings
Scientific and Technical Merit

• Research focus: software stack for petascale systems
  – system software for leadership computing systems
    • memory management, I/O, communication
  – communication library optimization
  – multicore math libraries
  – compiler technology
    • dynamic optimization of CCA applications
    • optimization of memory hierarchy performance
    • implementation of PGAS languages
  – performance measurement, analysis, modeling, presentation
    • quantify and pinpoint performance bottlenecks on leadership platforms
  – application studies and engagement

• Advancing state of the art across entire petascale software stack
Research and Development

- Research and development
  - new system software for leadership computing
    - Zepto OS compute node kernel, ZOID I/O daemon
  - communication optimization
    - GASNet optimization for IB, BG/P, Cray XT
    - UPC collective optimizations for multicore processors
  - PLASMA and OSKI: dense and sparse LA on multicore processors
  - performance tools
    - binary analysis for call stack unwinding on Cray XT, BG/P (& SiCortex)
    - pinpoint scalability & performance problems on LCF & multicore
    - user interfaces for effective performance analysis: hpcviewer, Jumpshot
    - performance tool components: Dyninst components, libmonitor
    - performance modeling to understand application bottlenecks
  - open source compilers: LoopTool, Fortran & CAF support in ROSE

- Application engagement in addition to R&D accomplishments
Project Performance - II

• 31 journal articles; 15 conference papers
• 6 theses; many presentations
• Website: http://cscads.rice.edu
  – project publications, selected presentations
  – open source software
  – summer workshop series coordination and dissemination

• Contribute to efficient use of DOE HPC resources?
  – helped GTC and S3D application teams optimize applications
    • S3D production code incorporates LoopTool generated code along with manual performance optimizations (opportunities uncovered with HPCToolkit); improved performance 12+% for benchmark test
    • provided code changes (data & loop restructuring, adaptive reordering) back to GTC team; potential for 20+% improvement
  – worked with vendors on software stack for leadership platforms

• Unrivaled engagement and outreach with CScADS workshops
Appropriateness of Methods

• System software
  – jumbo pages to avoid TLB miss losses; high performance I/O

• Communication library optimization
  – dynamic page pinning for RDMA; optimizing collectives for multicore

• Compiler technology
  – accelerate scientific kernels using expression reassociation
  – dynamic optimization of late binding for CCA applications
  – PGAS languages + memory hierarchy optimization: source to source

• Multicore linear algebra
  – asynchronous dynamic scheduling of work to reduce synch delays

• Performance tools
  – sampling for low overhead, no blind spots
  – call stack unwinding to associate costs with dynamic context
  – binary analysis for broad applicability, accuracy
  – differential analysis for pinpointing scalability bottlenecks
  – effective user interfaces for analyzing performance data
  – components for community leverage
## Software for Leadership Computing

<table>
<thead>
<tr>
<th>Institution</th>
<th>Software</th>
<th>Blue Gene</th>
<th>Cray XT</th>
<th>Linux</th>
<th>Other</th>
</tr>
</thead>
<tbody>
<tr>
<td>Argonne</td>
<td>Zepto OS</td>
<td>yes</td>
<td></td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Argonne</td>
<td>ZOID</td>
<td>yes</td>
<td></td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Berkeley</td>
<td>GASNet, UPC</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Berkeley</td>
<td>OSKI</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Tennessee</td>
<td>PLASMA</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Rice</td>
<td>HPCToolkit</td>
<td>yes</td>
<td>yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rice</td>
<td>hpcviewer</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Rice</td>
<td>libmonitor</td>
<td>yes</td>
<td>yes</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Wisconsin</td>
<td>SymtabAPI</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Wisconsin</td>
<td>InstructionAPI</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
<td>yes</td>
</tr>
<tr>
<td>Wisconsin</td>
<td>StackwalkerAPI</td>
<td>yes</td>
<td>yes</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Collaboration with Centers & Applications

• Strong collaboration with PERI
  – partially supports development of HPCToolkit
  – HPCToolkit used by PERI
    • outreach to application teams
      – Lattice QCD (Brower), Madness (Harrison), Chombo (Colella)
    • internally by PERI autotuning researchers
  – CScADS participates in and extends work of PERI Tiger Teams
    • GTC (Ethier), S3D (Chen), FLASH (Dubey), PFLOTRAN (Lichner)
  – broad participation by PERI in CScADS ET workshops
• Working with TOPS on sparse linear algebra
• Exploring dynamic compilation to accelerate CCA codes
Training

• Rice

• Wisconsin

• Berkeley
FY09 Plans

- **Rice**
  - performance tools
    - release HPCToolkit on the leadership computing platforms
    - begin work on support for analysis of data from a huge # of cores
  - compilers
    - release LoopTool for use by application teams
    - continue work on dynamic optimization, ROSE, scripting languages

- **Argonne**
  - continue replacing components in BG/P s/w stack with open source
  - release the first version of ZeptoOS for BG/P to the public

- **Berkeley**
  - release UPC & GASNet with improved support for BG/P, XT, and IB
  - continue optimizing sparse linear algebra for multicore (with UTK)

- **Tennessee**
  - explore dynamic and adaptive out-of-order execution patterns for linear algebra on multicore and heterogeneous nodes

- **Wisconsin**
  - continue development of InstructionAPI and ControlFlowAPI
FY10-11 Plans (Rice)

• **Performance tools**
  – develop & deploy parallel analysis of HPCToolkit measurement data
  – incorporate sionlib for parallel I/O of HPCToolkit measurement data
  – enhance hpcviewer to work with out-of-core data for 100K PE
  – develop and deploy hpctraceview for out-of-core data for > 1K PE
  – work with Wisconsin to consolidate unwinding in StackwalkerAPI

• **Compilers and communication libraries**
  – develop and deploy source-to-source CAF 2.0 for leadership computers
    • collaborate with LLNL & LANL on ROSE
    • collaborate with UC Berkeley on GASNet & IBM on APGAS runtime
  – develop and deploy and dynamic optimization for HPC applications
  – continue to work with Fortran standards committee on CAF

• **Continue application engagement**
  – continue collaboration with PERI Tiger teams
  – direct engagement with application teams and ET centers
FY10-11 Plans

• Argonne
  – make Big Memory a per-core resource for BG’s virtual node mode
  – add IBM’s recent improvements to the communication software stack
  – make high-performance file I/O transparent and easy for applications
  – enhance Jumpshot to display more summary data for > 10K nodes

• Berkeley
  – continue work on communication optimization and libraries
  – create autotuning environment for collectives
  – collaborate with Rice on GASNet & IBM on APGAS runtime

• Tennessee
  – continue exploration of math libraries for multicore and heterogeneous hybrid systems

• Wisconsin
  – continue work on InstructionAPI
  – development of graph API's (CFG, DDG, PDG), Dyninst for Cray XT
  – work with Rice to consolidate unwinding in StackwalkerAPI
Advancing SciDAC Goals

• R&D of software to enable petascale science
  – system software, communication libraries, math libraries, performance tools, compiler technology for leadership computing

• Collaborate with other SciDAC centers
  – collaborate with enabling technology centers
    • PERI @ UNC extends our impact by engaging application teams with HPCToolkit performance tools (e.g. Lattice QCD, Madness)
    • working on dynamic compilation to accelerate CCA
  – directly engage SciDAC application teams
    • extend work of PERI Tiger teams: Flash, GTC, S3D, PFLOTRAN
    • ensure performance tools meet application needs

• Engage the community in SciDAC
  – outreach workshops to foster interaction between ET and applications
  – ET workshops engage the broader research community
    • accelerate R&D of technologies supporting SciDAC mission