Performance Tools Working Group
August 1-2, 2007
Washington, DC

Chairs:
Dan Reed, Renaissance Computing Institute (RENCI)
Bernd Mohr, Research Centre Juelich
Thank You

Thanks to the group for valuable insights
- lots of interchange and good ideas

We have tried to capture the key ideas
- any errors are Bernd/Dan’s
Our Charge: Performance Tools

Topics

- analysis, modeling and optimization
- interactive and automatic approaches
- data management and instrumentation
- hardware and OS support
- visualization and presentation
- etc

Current status

Petascale requirements

Findings

Recommendations

- ordered priority list
- challenge type
  - technical, funding
  - policy, training
- impact
  - high, medium, low
- Probability (risk)
  - high, medium, low
On Performance Tools ...

Sequential  Terascale  Trans-petascale

CHOOSE A NUMBER FROM 0 TO 10 THAT BEST DESCRIBES YOUR PAIN

CHOOSE THE FACE THAT BEST DESCRIBES HOW YOU FEEL
Ecosystem Roles and Interactions

Less disjoint than one might think

– small community with deep and long history
Performance Measurement: A Status Report

- Intelligent use well understood
  - instrumentation, measurement and analysis

- Instrumentation techniques
  - FORTRAN, C, less C++, ...
  - MPI, user functions/regions, less OpenMP, ...

- Measurement techniques
  - sampling, profiling and tracing

- Analysis techniques (weakest of the three)
  - too much data, not enough analysis
  - tools find symptoms, but not root problems
Performance Measurement: A Status Report

- Effective techniques for homogeneous systems
  - heterogeneous challenges coming
    - multicore, specialized processors, ...
- Too much concentration on *time as a metric*
  - need more support for memory analysis, ...
- Largely left to the user
  - true analysis, then optimization/tuning
Performance Modeling: A Status Report

- Limited breakout group discussion
  - not a reflection of lack of importance

- Multiple meanings/uses of modeling
  - system characterization
  - application prediction, ...

- Opinion
  - better than in the measurement community
Qualitative Status Assessment

- Measurement/analysis: WIP
- Modeling: WIP
- Optimization: NC
- Interactive/manual: WIP
- Automatic: WIP/NC
- Data management: WIP
- Instrumentation: WIP
- Hardware and operating system support: WIP
- Visualization/presentation: WIP

Legend:
- Work in progress (WIP)
- No clue (NC)
- In hand (IH)
Performance Tool Ecosystems
Petascale Requirements

» Increased automation
  – anomaly detection
  – correlation and clustering
  – data reduction

» Abstraction support
  – detail/complexity hiding

» Runtime adaptation
  – task topologies, ...

» Heterogeneity
  – programming models: explicit and implicit
  – hardware

» Hierarchy, including sharing
Petascale Requirements

- Fault tolerance/resilience
- Education and training
- Multi-level instrumentation
- Memory and I/O analysis
- Performability
  - hybrid/integrated performance and reliability
- Presentation and insight
  - scalable visualization
- Performance modeling and prediction
- Scaling of known methods and techniques
  - million-way parallelism and beyond
Economic Divergence/Optimization

$\$/teraflop-year
- declining rapidly

$\$/developer-year
- rising rapidly

System complexity
- rising

Applications outlive systems
- by many years

Implications ...
Findings

- Petascale is *not* terascale scaled up
  - higher complexity, heterogeneity
- Petascale method inadequacies
  - manual methods
  - purely static and offline approaches
- Manual method needs
  - anomaly detection and optimization
- Purely static and offline methods
  - complement with online, adaptive methods
Findings

- Crucial interactions
  - users/staff/developers critical
  - education and training
  - feedback

- Insufficient integration
  - among tools
  - component reuse

- No general pathway for release quality tools
  - hardening, documentation, training, support, ...
## Recommendations

<table>
<thead>
<tr>
<th>#</th>
<th>Challenge</th>
<th>Type</th>
<th>Probability (Risk)</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>User engagement and training</strong></td>
<td>Training</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td><strong>Additional information sources e.g. I/O, memory</strong></td>
<td>Technical</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td><strong>Long-term maintenance and support</strong></td>
<td>Funding &amp; policy</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td><strong>Funds for technology transfer and deployment</strong></td>
<td>Funding &amp; policy</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td><strong>Application-driven development of tools</strong></td>
<td>Medium</td>
<td>Medium</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Substantial advances in automation of diagnosis, optimization and anomaly detection</strong></td>
<td>Technical</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td><strong>Developing live techniques to extend post-mortem</strong></td>
<td>Technical</td>
<td>Medium</td>
<td>Medium</td>
</tr>
</tbody>
</table>
## Recommendations (Continued)

<table>
<thead>
<tr>
<th>#</th>
<th>Challenge</th>
<th>Type</th>
<th>Probability (Risk)</th>
<th>Impact</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Integrated, persistent monitoring components</td>
<td>Technical</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td>Support for multi-component and multi-disciplinary applications</td>
<td>Technical</td>
<td>High</td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td>Detection of load imbalance</td>
<td>Technical</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>Support for heterogeneous and hierarchical hardware</td>
<td>Technical</td>
<td>High</td>
<td>High</td>
</tr>
<tr>
<td></td>
<td>Support for new and hybrid programming models</td>
<td>Technical, funding, policy &amp; training</td>
<td>Medium</td>
<td>Medium</td>
</tr>
<tr>
<td></td>
<td>Add performance analysis to CS curriculum</td>
<td>Training &amp; policy</td>
<td>Low</td>
<td>Low</td>
</tr>
</tbody>
</table>