CSC469/585: Winter 2011-12
High Availability and Performance Computing: Towards non-stop services in HPC/HEC/Enterprise IT Environments

Chokchai (Box) Leangsuksun,
Associate Professor, Computer Science
Director, eXtreme Computing Research Group
Center for Entrepreneurship and Information Technology
Louisiana Tech University

Box’s 1 minute Bio

- B. Eng (AE 1983): Khon Kean University
- MS and PhD in CS (1995):
- 7 years in industry (Lucent)
  - Highly Reliable Software/system: R&D > 4 major network management products
  - Architect, PM, Tech lead (15-30 team size)
- SWECPO ENDOWED PROFESSOR SINCE 2007, Best Teaching Award 2003
- Louisiana Hero, www.searchKatrina.org
- Associate Professor in CS since 2002.
  - Collaborations with national and industry labs (e.g.ORNL, Intel, Ericsson, NCSA, Dell, etc)
  - Funded research projects by DOD, NSF, DOE
- Research Interest
  - Resilience in HPC, Cluster computing, Fault Tolerance, Reliability, Availability and Serviceability (RAS) in HPC
- Services
  - Various Program committee: IEEE Cluster computing, Grid computing education
  - Co-founder and chair: High Availability and Performance Computing Workshop
Outline

- Background and Motivation
- Current R&D & Educational projects
  - HA-OSCAR Architecture, infrastructure and System management
  - Design & Dependability Analysis
  - Education in HAPC
- Conclusion

High Performance Computing

- **HPC-** “Hardware and Software techniques devised, for building computer systems to *quickly* perform large amounts of computation in the shortest possible time”
- **HPC is not the same as high throughput**
HPC goes mainstream

- Multi core is everywhere available
- GPU is hot!!!
- At SC|05, Bill Gates gave a keynote as HPC goes mainstream
- MS are in HPC cluster (windows)
- More critical applications requires HPC
- Your mobiles and tablets are multicore + GPU

How to achieve HPC

- Work hard – add more powerful unit(s).
  - Faster CPU
  - More CPU, parallel architecture
  - Faster connectivity
- Work smart – better algorithms to take advantage of parallelism
  - Multiple-programming – processing (Unix fork)
  - Multi-threading
  - Parallel programming (MPI, openMP, PVM)
Parallel Architectures

Hardware Architectures for HPC and their Parallel Programming Models

- Distributed Memory Systems, MPP, Clusters - Message Passing
- Shared Memory Systems (SMP), Shared Memory Programming
  - Multicore/Manycores
  - Specialized Architectures, Vector Processing
    Data Parallel Programming (SIMD).
- The Grid, Grid Computing

TOP500.org

The chart is from top500.org
The chart is from top500.org
The chart is from top500.org
The chart is from top500.org
Example of HPC app

Life Sciences: Protein Folding

What is a Protein?

Examples of Protein Function
Structural: keratin (skin, hair, nail), collagen (tendon), fibrin (clot)
Moist: actomyosin (muscle)
Transport: Hemoglobin (blood)
Signaling: Growth factors, insulin, hormones (blood)
Regulation: Transcription factor (gene expression)
Catalysis: Enzymes

Amino acid

Page is excerpted from David Klepacki’s presentation
Example of HPC app

Ab initio Protein Folding
Computational Requirements

What’s the scale of the problem?

<table>
<thead>
<tr>
<th>Description</th>
<th>Count*</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Atoms</td>
<td>~32,000</td>
<td>300 amino acid protein + water</td>
</tr>
<tr>
<td>Force evaluations / time step</td>
<td>$10^3$</td>
<td>Pairwise atom - atom interactions</td>
</tr>
<tr>
<td>FLOPs / force evaluation</td>
<td>150</td>
<td>Typical molecular dynamics</td>
</tr>
<tr>
<td>FLOPs / time step</td>
<td>$1.5 \times 10^{11}$</td>
<td></td>
</tr>
<tr>
<td>Each time step</td>
<td>$10^{-15}$ s</td>
<td>1 - 5 femto second</td>
</tr>
<tr>
<td>Total simulation time</td>
<td>$10^{-3}$ s</td>
<td>Protein folds in ~1 milli second</td>
</tr>
<tr>
<td>Total time steps</td>
<td>$2 \times 10^{11}$</td>
<td></td>
</tr>
<tr>
<td>FLOPs / simulation</td>
<td>$3 \times 10^{12}$</td>
<td>Total FLOPs to fold a protein</td>
</tr>
<tr>
<td>Execution time</td>
<td>$3 \times 10^{13}$ s</td>
<td>1 year</td>
</tr>
<tr>
<td>Required FLOPS</td>
<td>$1 \times 10^{16}$</td>
<td>1 Petaflop/s</td>
</tr>
</tbody>
</table>

Estimate is conservatively based on quadratic algorithm.
Better algorithms will reduce (somewhat) running time, but usual surprises will increase it!
And good science will require multiple simulations.

Page is excerpted from David Klepacki’s presentation

Production HPC system in the real world.

ASCI White
Los Alamos National Laboratory
Availability density of each node in the ASC White (was one of the top HPC)

The majority of the availability of the each node is above 0.95 with a few of them below 0.8.

The average is 0.98, with standard deviation 0.033.

This indicates that, compared to others, some nodes manifest outages more.

If the runtime systems are not aware of these nodes unreliability, it may result in low system total performance, extended application completion or failure.

Nodes MTTF density (in hours)

The maximum is 5592 hours.

The minimum is 1230 hours.

The average is 3923, with standard deviation 1217.

Nodes MTTF for White
Nodes Downtime (in hours)

The average is 355 hours, with standard deviation 56. Most of the total downtime for each node is around 100 hours.

Reliability Differences on the same HW

- “AND Survivability” analysis based on
  - at 10, 100, 1000 nodes all have to survive.
  - Each node MTTF at 5000 hours
- N=10, \( \text{MTTF} = 492.424242 \)
- N=100, \( \text{MTTF} = 49.9902931 \)
- N=1000, \( \text{MTTF} = 4.99999003 \)

- Reliability and Availability info - Better Job scheduling and execution
**Availability**

- A measurement represents a ratio of uptime vs. total times
- High availability - ability of a system to perform its function continuously (without interruption) for a significantly longer period of time than the reliabilities of its individual components would suggest.
- High availability is most often achieved through fault tolerance.
**Availability Model**

- Server up
- Server down & repair

**Availability model**

- S1
- S2

**HA-OSCAR dual head model**

- S1&S2

**Availability (continued)**

- **Availability** = uptime/total time
- **MTTF** = Mean Time To Failure
  - Average time to failure, when it is *not* repairable
- **MTBF** = Mean Time Between Failure
  - Average time to failure, when it is repairable
- **MTTR** = Mean Time To Repair
- **Availability** = MTTF/(MTTF+MTTR)
## Degree of Availability

<table>
<thead>
<tr>
<th>System Type</th>
<th>Unavailability (minutes/year)</th>
<th>Availability (in percent)</th>
<th>Availability Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unmanaged</td>
<td>50,000</td>
<td>90</td>
<td>1</td>
</tr>
<tr>
<td>Managed</td>
<td>5,000</td>
<td>99</td>
<td>2</td>
</tr>
<tr>
<td>Well-managed</td>
<td>500</td>
<td>99.9</td>
<td>3</td>
</tr>
<tr>
<td>Fault-tolerant</td>
<td>50</td>
<td>99.99</td>
<td>4</td>
</tr>
<tr>
<td>High Availability</td>
<td>5</td>
<td>99.999</td>
<td>5</td>
</tr>
<tr>
<td>Very High Availability</td>
<td></td>
<td>99.9999</td>
<td>6</td>
</tr>
<tr>
<td>Ultra Availability</td>
<td></td>
<td>99.99999</td>
<td>7</td>
</tr>
</tbody>
</table>

## Quiz. Find out for each 9’s in one year

<table>
<thead>
<tr>
<th>System Type</th>
<th>Unavailability (minutes/year)</th>
<th>Availability (in percent)</th>
<th>Availability Class</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unmanaged</td>
<td></td>
<td>90</td>
<td>1</td>
</tr>
<tr>
<td>Managed</td>
<td></td>
<td>99</td>
<td>2</td>
</tr>
<tr>
<td>Well-managed</td>
<td></td>
<td>99.9</td>
<td>3</td>
</tr>
<tr>
<td>Fault-tolerant</td>
<td></td>
<td>99.99</td>
<td>4</td>
</tr>
<tr>
<td>High Availability</td>
<td></td>
<td>99.999</td>
<td>5</td>
</tr>
<tr>
<td>Very High Availability</td>
<td></td>
<td>99.9999</td>
<td>6</td>
</tr>
<tr>
<td>Ultra Availability</td>
<td></td>
<td>99.99999</td>
<td>7</td>
</tr>
</tbody>
</table>
Quiz #2

- Say if the machine costs $20M
- Availability is 92.1%
- What is the downtime and cost of downtime?

Unavailability = No performance and functionality

- Availability enables Performance/functionality
- Performance oriented Enterprise/Shared Major computing resources - 7/24/365
- Losses of $195K - $58M with 3.5 hrs (Meta Group report)
- Service provider Regulation/Mandate
  - FCC mandate (Class 5 local switch)
- Losses time and opportunities
- Life-threatening
- National Security (Home Land defense)
What involves (non-stop services)?

Goals

- Towards Non-stop services in HPC/HEC environments
  - High Availability (Reliability)
  - High Serviceability (planned downtime)
  - High Performance Computing (HPC)
  - We want them all