Introduction to Computer Science

The What, How, and Why of CS

Ioan Raicu
Computer Science Department, Illinois Institute of Technology
Math and Computer Science Division, Argonne National Laboratory

February 14th, 2013
What is Computer Science?

• The scientific and mathematical approach in information technology and computing
• Started in the 1960s from Mathematics or Electrical Engineering
• Today:
  – Arguably one of the most fundamental discipline that touches all other disciplines and people
The advent of computation can be compared, in terms of the breadth and depth of its impact on research and scholarship, to the invention of writing and the development of modern mathematics.

Ian Foster, 2006

Introduction to Computer Science: The What, How, and Why of CS
Science Paradigms

- Thousand years ago: science was **empirical**
  - describing natural phenomena

- Last few hundred years: **theoretical** branch
  - using models, generalizations

- Last few decades: a **computational** branch
  - simulating complex phenomena

- Today: **data exploration** (eScience)
  - unify theory, experiment, and simulation
  - Data captured by instruments or generated by simulator
  - Processed by software
  - Information/knowledge stored in computer
  - Scientist analyzes database/files using data management and statistics
• Theory
  – Theory of computation
  – Information and coding theory
  – Algorithms and data structures
  – Programming language theory
  – Formal methods

• Systems
• Theory

• Systems
  – Artificial intelligence
  – Computer architecture
  – Computer graphics and visualization
  – Computer security and cryptography
  – Computational science
  – Databases and information retrieval
  – Distributed systems
  – Health Informatics
  – Information science
  – Programming Languages
  – Software engineering
• What is a distributed system?

“A collection of independent computers that appears to its users as a single coherent system”

-A. Tanenbaum
Cluster Computing

Computer clusters using commodity processors, network interconnects, and operating systems.
**Supercomputing ~ HPC**

Highly-tuned computer clusters using commodity processors combined with custom network interconnects and customized operating system.
Grids tend to be composed of multiple clusters, and are typically loosely coupled, heterogeneous, and geographically dispersed.
• A large-scale distributed computing paradigm driven by:
  1. economies of scale
  2. virtualization
  3. dynamically-scalable resources
  4. delivered on demand over the Internet

Clouds ~ hosting
Coursework

- CS 451 Introduction to Distributed Computing
- CS 546 Parallel and Distributed Processing
- CS 550 Advanced Operating Systems
- CS 552 Distributed Real-Time Systems
- CS 553 Cloud Computing
- CS 554 Data-Intensive Distributed Computing
- CS 570 Advanced Computer Architecture
• Xian-He Sun
• Zhiling Lan
• Shangping Ren
• Ioan Raicu
• **Research Focus**
  – Emphasize designing, implementing, and evaluating systems, protocols, and middleware with the goal of supporting **data-intensive applications on extreme scale distributed systems**, from many-core systems, clusters, grids, clouds, and supercomputers

• **People**
  – Dr. Ioan Raicu (Director)
  – 5 PhD Students
  – 4 MS Students
  – 5 UG Students

• **Contact**
  – [iraicu@cs.iit.edu](mailto:iraicu@cs.iit.edu)
- $O(10)$ cores commodity architectures
- $O(100)$ cores proprietary architectures
- $O(1000)$ GPU hardware threads

~1000 cores/threads commodity architectures
Exascale Computing

  - O(10K) nodes
  - O(100K) cores
  - ~1M nodes (100X)
  - ~1B processor-cores/threads (10000X)

Top500 Projected Development,
• Relatively new paradigm… 3 years old
• Amazon in 2009
  – 40K servers split over 6 zones
    • 320K-cores, 320K disks
    • $100M costs + $12M/year in energy costs
    • Revenues about $250M/year
• Amazon in 2018
  – Will likely look similar to exascale computing
    • 100K~1M nodes, ~1B-cores, ~1M disks
    • $100M~$200M costs + $10M~$20M/year in energy
    • Revenues 100X~1000X of what they are today
Common Challenges

- **Power efficiency**
  - Will limit the number of cores on a chip (Manycore)
  - Will limit the number of nodes in cluster (Exascale and Cloud)
  - Will dictate a significant part of the cost of ownership

- **Programming models/languages**
  - Automatic parallelization
  - Threads, MPI, workflow systems, etc
  - Functional, imperative
  - Languages vs. Middlewares
Common Challenges

- Bottlenecks in scarce resources
  - Storage (Exascale and Clouds)
  - Memory (Manycore)

- Reliability
  - How to keep systems operational in face of failures
  - Checkpointing (Exascale)
  - Node-level replication enabled by virtualization (Exascale and Clouds)
  - Hardware redundancy and hardware error correction (Manycore)
CPU Cores: 130816
Tasks: 1048576
Elapsed time: 2483 secs
CPU Years: 9.3

Throughput (tasks/sec)

Tasks Completed
Number of Processors
Time (sec)
Processors
Active Tasks
Tasks Completed
Throughput (tasks/sec)

Speedup: 115168X (ideal 130816)
Efficiency: 88%

Introduction to Computer Science: The What, How, and Why of CS
**Applications Pharmaceuticals**

- **ZINC 3-D structures**
  - protein (1MB)
    - 2M structures (6 GB)
      - (1 per protein: defines pocket to bind to)

- **DOCK6 rec file**
  - Receptor
    - (1 per protein: defines pocket to bind to)

- **FRED rec file**

- **NAB Script template**
  - NAB script parameters
    - (defines flexible residues, #MDsteps)

- **BuildNABScript**

- **Amber Score**
  - 1. AmberizeLigand
  - 2. AmberizeReceptor
  - 3. AmberizeComplex
  - 4. perl: gen nabscript

- **GCMC**

- **For 1 target:**
  - 4 million tasks
  - 500,000 cpu-hrs (50 cpu-years)

---

**PDB protein descriptions**

- **ZIM structures**
  - protein (1MB)

- **FRED**
  - ~4M x 60s x 1 cpu
  - ~60K cpu-hrs

- **Select best ~5K**

- **DOCK6**
  - ~10K x 20m x 1 cpu
  - ~3K cpu-hrs

- **Select best ~5K**

- **Amber**
  - ~10K x 20m x 1 cpu
  - ~3K cpu-hrs

- **Select best ~500**

- **GCMC**
  - ~500 x 10hr x 100 cpu
  - ~500K cpu-hrs

---

**Report**

- **Ligands**
- **Complexes**

---


Introduction to Computer Science: The What, How, and Why of CS
CPU cores: 118784
Tasks: 934803
Elapsed time: 2.01 hours
Compute time: 21.43 CPU years
Average task time: 667 sec
Relative Efficiency: 99.7%
(from 16 to 32 racks)
Utilization:
- Sustained: 99.6%
- Overall: 78.3%

• **Purpose**
  – On-demand “stacks” of random locations within ~10TB dataset

• **Challenge**
  – Processing Costs:
    • $O(100\text{ms})$ per object
  – Data Intensive:
    • 40MB:1sec
  – Rapid access to 10-10K “random” files
  – Time-varying load

---

<table>
<thead>
<tr>
<th>Locality</th>
<th>Number of Objects</th>
<th>Number of Files</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>111700</td>
<td>111700</td>
</tr>
<tr>
<td>1.38</td>
<td>154345</td>
<td>111699</td>
</tr>
<tr>
<td>2</td>
<td>97999</td>
<td>49000</td>
</tr>
<tr>
<td>3</td>
<td>88857</td>
<td>29620</td>
</tr>
<tr>
<td>4</td>
<td>76575</td>
<td>19145</td>
</tr>
<tr>
<td>5</td>
<td>60590</td>
<td>12120</td>
</tr>
<tr>
<td>10</td>
<td>46480</td>
<td>4650</td>
</tr>
<tr>
<td>20</td>
<td>68460</td>
<td>2025</td>
</tr>
<tr>
<td>30</td>
<td>23695</td>
<td>790</td>
</tr>
</tbody>
</table>

---


[TG06] “AstroPortal: A Science Gateway for Large-scale Astronomy Data Analysis”
• Be the one creating and shaping the future of technology, not just the user
• Employment at the best technology companies in the world (see next slide)
• Be the next Steve Jobs (Apple), Bill Gates (Microsoft), Sergei Brin (Google), or Zach Zuckerberg (Facebook)
• Be part of the most amazing revolution to date: The Computing Revolution!
Employment Opportunities Distributed Systems

- Google
- Yahoo
- Microsoft
- Amazon
- IBM
- Apple
- VMWare
- Netflix
- Cray
- Intel
- NVIDIA
- Facebook
- LinkedIn
- Salesforce.com
- Rackspace
- Red Hat
- Cleversafe
- UnivaUD
- Greenplum
- AsterData
- Proprietary Trading Companies
- Department of Energy Laboratories
- NASA
- Academic supercomputer centers
- Many more…
More Information

• More information:
  – http://www.cs.iit.edu/~iraicu/
  – http://datasys.cs.iit.edu/

• Contact:
  – iraicu@cs.iit.edu

• Questions?