Towards Exascale: Developments in Hardware and Software

Dr Mike Ashworth
Associate Director,
Computational Science & Engineering Department,
STFC Daresbury Laboratory

mike.ashworth@stfc.ac.uk
Outline

- Exascale Computing Roadmaps
- The Software Challenge
- Programming Heterogeneous Multi-core Platforms
Remarkable Growth in Cost-Efficiency

“If cars had developed at the same rate as computers you would be able to buy a Rolls-Royce for a penny and travel to the moon and back on a thimbleful of fuel”

Unknown
Early years: performance driven by faster clock frequencies and instruction level parallelism – no need to change software.

Recent years: performance driven by massive increase in parallelism – parallel software revolution.
Increase in number of cores

12 systems have >100,000 cores
100 systems have >10,000 cores

Number of cores

TOP500 list

Maximum
Average
Minimum

NCAR workshop
7th September 2011
• Transistor density is still increasing
• Clock frequency is not due to power density constraints
• Cores per chip is increasing, multi-core CPUs (currently 8-16) and GPUs (100s)
• Little further scope for instruction level parallelism

Source: Intel, Microsoft (Sutter) and Stanford (Olukotun, Hammond)
“It would appear that we have reached the limits of what is possible to achieve with computer technology, although one should be careful with such statements, as they tend to sound pretty silly in five years”

John von Neumann, 1949
Exascale Computing Study; Technology Challenges in Achieving Exascale Systems

Peter Kogge, Editor & Study Lead
Kevon Bergman
Shekhar Borkar
Dan Campbell
William Carbon
William Dally
Mandy Dunne
Paul Flanzen
William Harrad
Kerry Hill
Jan Hilsper
Sherron Karp
Stephen Keckler
Dean Klein
Robert Lucey
Mark Richards
Ali Scarpelli
Steven Smial
Allan Snively
Thomas Stroiey
R. Stanley Williams
Katherine Velick

September 28, 2008

This work was sponsored by DARPA IPTO in its Program Manager; AFR contract number FA8650-07-C-1723. This report is published in the interest of scientific and technical information exchange and its publication does not constitute the Government's approval or disapproval of its ideas or findings.

NO!

Using Government drawings, specifications, or other data without the written approval of the Government is a violation of Federal law.

APPROVED FOR PUBLIC RELEASE, DISTRIBUTION UNLIMITED.
Exascale systems achievable by 2017
- Little increase in clock rate
- 10M-100M processor cores
- Heterogeneous (1:100 full instruction set to lightweight cores)
- 3D packaging with memory close to cores
- Mixed optical plus electrical interconnect
- 10PB-100PB memory
- Fault tolerance with support in hardware and software

For total system power of 10-20 MW, need 50-100 GF/W cf. Green500 #1 Nov10 1.68 GF/W

Exascale Software issues being taken forward by IESP

www.exascale.org
# The Changing Balance of Systems

<table>
<thead>
<tr>
<th>System</th>
<th>Terascale (HPCx 2002)</th>
<th>Petascale (Jaguar 2009)</th>
<th>Exascale (DARPA strawman)</th>
</tr>
</thead>
<tbody>
<tr>
<td># of nodes</td>
<td>160</td>
<td>18,688</td>
<td>223,872</td>
</tr>
<tr>
<td># cores/ node</td>
<td>8</td>
<td>12</td>
<td>742</td>
</tr>
<tr>
<td># of cores</td>
<td>1280</td>
<td>224,256</td>
<td>166,113,024</td>
</tr>
<tr>
<td># racks</td>
<td>40</td>
<td>284</td>
<td>583</td>
</tr>
<tr>
<td>Total Mem (TB)</td>
<td>1.28</td>
<td>300</td>
<td>3,580</td>
</tr>
<tr>
<td>Disk (TB)</td>
<td>18</td>
<td>600</td>
<td>3,580</td>
</tr>
<tr>
<td>Tape (TB)</td>
<td>35</td>
<td>10,000</td>
<td>3,580,000</td>
</tr>
<tr>
<td>Peak (Petaflop/s)</td>
<td>0.0067</td>
<td>2.33</td>
<td>1000</td>
</tr>
<tr>
<td>Total Power (MW)</td>
<td>0.5</td>
<td>7.0</td>
<td>68</td>
</tr>
<tr>
<td>Gflops/ W</td>
<td>0.013</td>
<td>0.33</td>
<td>14.73</td>
</tr>
<tr>
<td>Bytes/ Flop</td>
<td>0.5</td>
<td>0.2</td>
<td>0.0036</td>
</tr>
</tbody>
</table>
“I know how to make 4 horses pull a cart – I don't know how to make 1024 chickens do it”

Enrico Clementi, mid1980s?

We are now “pulling carts” with 294,912 fruit flies!
The Software Challenge of Petascale and Exascale

- Huge growth in no. of threads of execution ($10^6$, $10^7$, ...)
- Restricted memory per thread (~1GB per core and dropping)
- Limited bandwidth to memory (bytes per flop)
- Limited interconnect (bytes per flop)
- Complex software ecosystem (OS, runtime, compilers, libraries, tools etc.)
- Software evolution much slower than hardware
- Software revolution is very expensive

Many of the software issues are constrained by the power issue
Flops are ‘free’
Data storage & movement are becoming severely limited by power
Hybrid architectures
CPUs + accelerators
Millions of threads
Limited memory b/w
Limited communications
The software challenge

Programming is static
- Basic techniques have changed little since the 1960s
- Languages are out of touch with architecture

Software is a critical and valuable resource
- Software lifetime (10-20 years) exceeds hardware (2-5 years)
- Major investment by skilled staff (10s-100s person-years)
- Major differentiator when running on commodity hardware

Complexity is rising dramatically
- Complex architectures; many-core, accelerators, heterogeneity
- Multi-disciplinary coupled applications
- Multiple levels of nested parallelism
- Integrated stack: OS, compilers, libraries, applications
Heterogenous Multi-core Platforms

**CPUs:** currently 8-16 cores; core counts still increasing with Moore’s Law (till 2020?)

**GPUs:** aggressive development roadmap with improving d-p performance and memory bandwidths

**Integrated CPU/GPU:** AMD Fusion, nVidia project Denver

**Intel Many Integrated Core (MIC):**
Knights Ferry currently 32 cores; increase to 50 cores with 22nm

**Common Issues:** performance of access to shared resources: memory & interconnect, programming models
Practical Software
Evolution and Revolution

- Systems will soon have multi-CPU nodes (~32 cores) with or without accelerators (~200 cores)

- Evolutionary development of existing codes
  - Use hybrid MPI/OpenMP to increase concurrency, while maintaining computational intensity and load balance
  - Use PGAS languages (e.g. Co-Array Fortran) to speed-up communications and overlap with computation
  - OpenMP standards are emerging targeting accelerators e.g. compiling OpenMP regions for accelerators

- Revolutionary development
  - New codes and new algorithms targeting many-core memory-light architectures
NEMO is the leading ocean model in use in Europe with a multi-national development team.

GNEMO is a 12-month project to assess feasibility of porting to GPUs: Jan-Dec 2011.

Ported two routines:

- lateral diffusion of tracers (traldf_iso); modest speed-up: 6-core Westmere CPU = 69% of Tesla GPU
- ice rheology (lim_rhg); slower on GPU

Issues: limited parallelism, low computational intensity, frequent halo exchanges.
If GPUs are the answer what the **** was the question?

Q. What is the best computer architecture for weather, climate and earth-system modelling (WC&ES)?

A. GPUs? I don’t think so!

Q. What hardware gives you high single-precision performance for certain regular problems?

A. GPUs? YES!

Q. Can we overcome the limitations of GPUs, to achieve the promised price/performance benefits for WC&ES?

A. MAYBE?
Conclusions

The path to multi-Petascale and Exascale systems requires new heterogeneous and multi-core architectures.

Where is the software going? How should programming models adapt?

How can we in the WC&ES community best target our limited development resources to develop the next-generation of codes able to exploit the next-generation architectures?
Thank you for your attention

Mike Ashworth
mike.ashworth@stfc.ac.uk