The challenges of Exascale systems from an applications perspective

Mark Seager
CTO of the TCG Ecosystem
Legal Disclaimer

Today's presentations contain forward-looking statements. All statements made that are not historical facts are subject to a number of risks and uncertainties, and actual results may differ materially.

NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL’S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER, AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL® PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT. INTEL PRODUCTS ARE NOT INTENDED FOR USE IN MEDICAL, LIFE SAVING, OR LIFE SUSTAINING APPLICATIONS.

Intel does not control or audit the design or implementation of third party benchmarks or Web sites referenced in this document. Intel encourages all of its customers to visit the referenced Web sites or others where similar performance benchmarks are reported and confirm whether the referenced benchmarks are accurate and reflect performance of systems available for purchase.

Intel processor numbers are not a measure of performance. Processor numbers differentiate features within each processor family, not across different processor families. See www.intel.com/products/processor_number for details.

Intel, processors, chipsets, and desktop boards may contain design defects or errors known as errata, which may cause the product to deviate from published specifications. Current characterized errata are available on request.

Intel, Intel Xeon, Intel Core microarchitecture, and the Intel logo are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

*Other names and brands may be claimed as the property of others

Copyright © 2011, Intel Corporation. All rights reserved.
Exascale Challenges

- System power (target of 20MW)
- System resilience (MTBF target of a week)
- The three scalabilities (HW, SW and Applications)
  - Extreme parallelism, data locality, programmability
- The memory wall
  - Bandwidth, latency and power
- Commercial systems through IA ecosystem

Overcoming these challenges requires O($1B) investment in long range R&D
Where are the main factors determining the future

Main Challenges
- Energy Efficiency
- Resiliency
- Cost Performance
- Effective exploitation of concurrency

Technology directions
- Silicon technology
- Memory Technology (will be a key long term differentiator)
- System architecture and program models (They go hand in hand)
- Algorithms, system software.
Linpack Performance 2x/1yr CAGR

Performance CAGR driven in part by geometric increase in node counts. Technology in TOP10 impacts entire TOP500 in 6-8 years.
From 1993-2012 TOP1 doubling rate is 1.061 years
The Earth Simulator Center

Last 12 TOP1 systems over 17 years since ASCI Red have all R&D funded by 3 different Governments
USA=7, China=3, Japan=2

Data courtesy Jack Dongarra

Updated 10/2013
Performance Fraction of TOP10 Systems
Moore’s law delivering core count doubling every 18 months and DRAM component density is doubling every 3 years.

The amount of memory per core will decrease geometrically through peta and Eexascale systems. Also, memory capacity B:F will decrease.
A leading edge parallel storage mechanism is required for Exascale

- Design with system focus that enables end-user applications
- Scalable hardware
  - Simple, Hierarchal
- Scalable Software
  - Factor and solve
  - Hierarchal with function shipping
- Scalable Apps
  - Asynchronous coms and IO
HPC Software that Exascales up and also scales down for transparent user experience
There are many ways to extract parallelism and they will continue.

Serial Code Node Level
Fast Scalar performance, Optimized C/C++, FORTRAN, Threading and Performance Libraries, Debug/Analysis Tools

Parallel Node Level
Multi-core, Multi-Socket, SSE and AVX instructions, OpenMP, Threading Building Blocks, Performance Libraries, Thread Checker, Cilk

Multi-Node / Cluster Level
Cluster Tools, MPI Checker
Evolution of Big Data Analytics

- Batch: ETL
- Interactive: SQL
- Predictive: ML

Users: Devices, Sensors, Users

Sensors

Streaming

HPC + Big Data

BI/D W + Big Data

Cloud + Big Data

- App Development
- Insight Accelerators
- Data Integration
- NoSQL Data Store
- Hadoop
- Hadoop Compatible File Systems
- Massively Scalable Cluster

Models, Queries

Reports

Real Time
HPC+Big Data I/O Architecture Leverages NVRAM for Bandwidth and Disk for Capacity

- **Compute** ~100s PF/s, ~10s PB/s, ~1 PB
- **Burst Buffer** ~10s TB/s, ~10s PB
- **Lustre** ~1s TB/s, ~100s PB
It is time to get serious about power efficiency

With top machine at 2.5 GF/Watt

We remain a long way from the target 50GF/s at Exaflop
We are already at a point where a 300PF/s machine is about 30 MW.
No longer a problem for the future. The systems we are next designing will not be able to be powered unless we act now.

What we need to do…

• Aggressive system level power saving techniques
  • Put power where needed…full system optimization for perf/W.
  • Fabric, memory system and processor.
  • Enable software to assist/hint.
• R&D focused on perf/W
• Must focus on perf/W for real applications. Linpack will come along as a result.
Re-think System Level Memory Architecture

- Emerging memory technologies
- New levels of memory hierarchy
- Minimize data movement across hierarchy
- Innovative packaging and IO solutions
The next step in perf/$$

Typical DRAM Memory Die (2016) ~ 8Gb will be about 100 mm^2 (as always)

Processor floating point unit ~0.03 mm^2 (2 Flops/cycle) (see below)

Even if the core is 100x bigger than the FPU, At 1.0 GB/core we have >100x more silicon in memory than processing. This is not cost balanced.

Threading gives us a mechanism to change this balance if we have enough bandwidth to support much higher compute/memory.

New memory architectures allow us to get a significant step in perf/$$

<table>
<thead>
<tr>
<th>Year</th>
<th>Tech (nm)</th>
<th>V</th>
<th>Area (mm^2)</th>
<th>E/Op (pJ)</th>
<th>f (GHz)</th>
<th>Watts/Exaflops</th>
<th>Watts/FPU</th>
</tr>
</thead>
<tbody>
<tr>
<td>2004</td>
<td>90</td>
<td>1.10</td>
<td>0.50</td>
<td>100</td>
<td>1.00</td>
<td>1.0E+08</td>
<td>0.10</td>
</tr>
<tr>
<td>2007</td>
<td>65</td>
<td>1.10</td>
<td>0.26</td>
<td>72</td>
<td>1.38</td>
<td>7.2E+07</td>
<td>0.10</td>
</tr>
<tr>
<td>2010</td>
<td>45</td>
<td>1.00</td>
<td>0.13</td>
<td>45</td>
<td>2.00</td>
<td>4.5E+07</td>
<td>0.09</td>
</tr>
<tr>
<td>2013</td>
<td>32</td>
<td>0.90</td>
<td>0.06</td>
<td>29</td>
<td>2.81</td>
<td>2.9E+07</td>
<td>0.08</td>
</tr>
<tr>
<td>2016</td>
<td>22</td>
<td>0.80</td>
<td>0.03</td>
<td>18</td>
<td>4.09</td>
<td>1.8E+07</td>
<td>0.07</td>
</tr>
<tr>
<td>2019</td>
<td>16</td>
<td>0.70</td>
<td>0.02</td>
<td>11</td>
<td>5.63</td>
<td>1.1E+07</td>
<td>0.06</td>
</tr>
</tbody>
</table>

Table 7.4: Expected area, power, and performance of FPUs with technology scaling.

DARPA: Exascale computing study: Exascale_Final_report_100208.pdf
Which of the Emerging Memory Technologies have sufficient potential & maturity to warrant accelerated R&D?
Revise DRAM Architecture

1. Need exponentially increasing BW (GB/sec)
   - Traditional DRAM:
     - Activates many pages
     - Lots of reads and writes (refresh)
     - Small amount of read data is used
     - Requires small number of pins
   - New DRAM architecture:
     - Activates few pages
     - Read and write (refresh) what is needed
     - All read data is used
     - Requires large number of IO’s (3D)

2. Need exponentially decreasing energy (pJ/bit)

Graph:
- X-axis: (nm)
- Y-axis: dB
- Lines indicate decreasing energy and increasing BW with decreasing feature size.
1Tb/s HMC DRAM Prototype

- 3D integration technology
- 1Gb DRAM Array
- 512 MB total DRAM/cube
- 128GB/s Bandwidth
- <10 pj/bit energy

<table>
<thead>
<tr>
<th></th>
<th>Bandwidth</th>
<th>Energy Efficiency</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDR-3 (Today)</td>
<td>10.66 GB/Sec</td>
<td>50-75 pJ/bit</td>
</tr>
<tr>
<td>Hybrid Memory Cube</td>
<td>128 GB/Sec</td>
<td>8 pJ/bit</td>
</tr>
</tbody>
</table>

10X higher bandwidth, 10X lower energy

Source: Micron
Innovative Packaging & IO Solutions

- Pins required + IO Power limits the use of traditional packaging
  - Tighter integration between memory and CPU
  - High BW and low latency using memory locality

Two different directions (2025)

New memory technologies replace/augment DRAM

- Memory capacity per compute 5x-10x better than DRAM
- Modest need for threading when new technologies available.
- Program model changes focus on increasing task scaling.

DRAM remains the dominant load-store memory technology

- Memory capacity per performance drops 10x to 20x from current levels.
- Aggressive threading is commonplace/necessary.
- Program model changes focus on thread scaling.
Industry focus on Linpack will move us down the wrong path

DGEMM Bytes:Flops

Courtesy Jawad Nasrullah
What will matter 10 years from now...

<table>
<thead>
<tr>
<th></th>
<th>Now</th>
<th>2025</th>
</tr>
</thead>
<tbody>
<tr>
<td>Perf/$</td>
<td>Linpack, Real Applications</td>
<td>Real Applications</td>
</tr>
<tr>
<td>Perf/Watt</td>
<td>Limited by worst case application</td>
<td>All applications will be able to run at</td>
</tr>
<tr>
<td></td>
<td></td>
<td>chosen power level. Dynamic, optimal energy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>management.</td>
</tr>
<tr>
<td>Reliability</td>
<td>Use of file system checkpoint restart</td>
<td>Transparent hardware and system</td>
</tr>
<tr>
<td></td>
<td>(spinning disks)</td>
<td>software recovery. Checkpoints in non-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>mechanical media.</td>
</tr>
<tr>
<td>Big Data</td>
<td>Parallel IO</td>
<td>New storage paradigm</td>
</tr>
</tbody>
</table>
New approach to storage hierarchy: applications driven object oriented data storage

- UQ, Applications define objects
- Storage of objects is abstracted
- Includes remote method invocation for user computations near the data
- Access transformed from `shell+ls` ➔ Python
- Metadata is accreted during object creation and IO
- Enables distributed data intensive computing model
- Enables Lustre ecosystem
- Enables analytics
Flexible I/O Software Infrastructure for Exascale and Big Data

• HDF5
  – Complex data models
    – Arbitrary connected graphs
  – Instantiate/persist (not read/write)
    – Transactional = consistent thru failure
  – End-to-end application data/metadata integrity

• I/O staging
  – Function shipping to I/O nodes
  – Aggregation & placement into DAOS objects
  – NVRAM burst buffer absorbs peak write load
  – NVRAM pre-staging cache
  – Scheduler integration

• DAOS
  – 10s of billions of objects distributed over thousands of OSSs
  – Scalable create/destroy, read/write
  – Transactional
Doing this at commercial scale…

… requires some form of distributed computation.
## A Paradigm Shift Is Underway

### Past priorities

<table>
<thead>
<tr>
<th>Performance</th>
<th>Single thread through frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming productivity</td>
<td>Architecture features for productivity</td>
</tr>
</tbody>
</table>
| Constraints                 | (1) Cost  
|                              | (2) Reasonable Power/Energy |

### Future priorities

<table>
<thead>
<tr>
<th>Performance</th>
<th>Throughput with reasonable single thread</th>
</tr>
</thead>
<tbody>
<tr>
<td>Programming productivity</td>
<td>Architecture features for energy efficiency while holding programmer productivity high</td>
</tr>
</tbody>
</table>
| Constraints                 | (1) Programmer productivity  
|                              | (2) Power  
|                              | (3) Cost/Reliability |
We have some difficult choices to make… co-design is critical

- Need to make the appropriate trade-off between…
  - Power efficiency --- near memory capacity --- cost

- Dynamic energy management.
  - Will drive some degree of non-reproducibility.

- RAS
  - Again trade off between gradual degradation and more costly redundancy.

- Single thread performance
  - Need to pick the optimal point for real applications at scale

- New storage paradigm
  - New object based for modern devises and applications use cases
System level R&D must inform direction

R&D effort must have:
• Multigenerational continuity
• Conduit to product
• Partnership with Industry and USG
Concurrency & Energy Efficiency is best achieved with a HW-SW Co-Design.
What value Co-design brings

For Industry:

• Access to world class extreme scale application experts.
• Immediate expert feedback on architectural options.
• Real partnerships allow for greater risk as the risk is shared.

For Partners/Users:

• Access to industry architects and visibility of current and future constraints.
• More visibility to emerging technologies and their “warts”.
• Allows the exploration of algorithms, new and old, on future architectures.

Our future is very bright if we can manage this… if not….someone else will.
Create Self Aware Systems

- **Introspective**: Observes itself, reflects on its behavior and learns
- **Goal Oriented**: Client specifies the goal; System figures out how to get there
- **Adaptive**: Computes delta between goal and observed state; takes action to optimize
- **Self Healing**: Continues to function through faults and degrades gracefully
- **Approximate**: Does not expend more effort than necessary to meet goals
System components will all look different at Exascale

- Processor core: Will have large dynamic range
- Memory hierarchy: New technologies and integration will drive different balance
- Network: Integration and photonics is inevitable
- Storage system: Will exploit new technologies
- Programming models: Support for new and old models

System components will change but users experience will evolve