Exec Committee
Pete Beckman
Jean-Yves Berthou
Jack Dongarra
Yutaka Ishikawa
Satoshi Matsuoka
Philippe Ricoux
Following the International Exascale Software Initiative (IESP 2008-2012) => **Big Data and Extreme Computing** workshops series (BDEC)

http://www.exascale.org/bdec/

**Overarching goal:**

1. Create an international collaborative process focused on the co-design of software infrastructure for extreme scale science, addressing the challenges of both extreme scale computing and big data, and supporting a broad spectrum of major research domains,

2. Describe funding structures and strategies of public bodies with Exascale R&D goals worldwide

3. Establishing and maintaining a global network of expertise and funding bodies in the area of Exascale computing

1 – BDEC Workshop, Charleston, SC, USA, April 29-May 1, 2013
2 – BDEC Workshop, Fukuoka, Japan, February 26-28, 2014
3 – BDEC Workshop, Barcelona, Spain, January 28-30, 2015
1 - BDEC Workshop, Charleston, SC, USA, April 29-May 1, 2013

Big Data and Extreme-scale Computing (BDEC) Workshop, Charleston, SC, USA, April 29-May 1, 2013

– Workflow Issues
– Architecture Challenges
– Higher Level Data Challenges: Data provenance, Policy based data management, Environments that support new types of data-driven research, Shared software infrastructure for intermediate processing
– Software Challenges: Tools to support real-time monitoring and observation of workflows, Coordination between data movement and compute services, Mechanisms to support fault tolerant workflows in data analysis, Mini-apps to support infrastructure co-design, Integration of widely used BD-capable data libraries into standard packages, Common tools for managing and exploring data, Interoperability Challenges
How Did We Get Here

• Previous BDEC meetings: US & JP
• Application Drivers: Astronomy, Medical, Genomics, Climate, Human Brain, Satellite images (GIS), Social Networks, etc.
• Good discussions on converged / shared problems:
  – Architecture, operations, software stack, algorithms
Instruments & Facilities

- “HPC Instrument” (Tsubame, Mira)
- SDSS, LSST, SKA, LOFAR, ...
- APS(20x), SNS, ...
- DNA Sequencers
- LHC / Atlas
- ARM
### Comparing Architecture

<table>
<thead>
<tr>
<th>Big Data</th>
<th>EC</th>
<th>Extreme Computing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Cost</strong></td>
<td>in memory and interconnect bandwidth</td>
<td>Significant Cost in memory and interconnect bandwidth</td>
</tr>
<tr>
<td><strong>Little Cost</strong> for resilient hardware in data storage</td>
<td>Significant Cost in resilient hardware in shared file system</td>
<td></td>
</tr>
<tr>
<td><strong>Little Cost</strong> for hardware to support system-wide resilience</td>
<td>Significant Cost in resilience hardware to reduce whole-system MTTF</td>
<td></td>
</tr>
<tr>
<td>Significant Cost: increased aggregate IOPs</td>
<td>Significant Cost: cutting-edge CPU performance features</td>
<td></td>
</tr>
<tr>
<td>Often trades performance for capacity</td>
<td>Often trades capacity for performance</td>
<td></td>
</tr>
</tbody>
</table>

### Comparing Operations

<table>
<thead>
<tr>
<th>Big Data</th>
<th>EC</th>
<th>Extreme Computing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Continuous access</strong> to long-lived “services” created by science community</td>
<td>Periodic access to compute resources via job submitted to scheduler and queue</td>
<td></td>
</tr>
<tr>
<td><strong>Time-shared</strong> access to elastic resources</td>
<td><strong>Space-shared</strong> compute resources for exclusive access during jobs</td>
<td></td>
</tr>
<tr>
<td>New hardware capacity purchased incrementally</td>
<td>New tightly integrated system purchased every 4 years</td>
<td></td>
</tr>
<tr>
<td>Users charged for all resources (storage, cpu, networking)</td>
<td>Users charged for CPU hours, storage and networking is free</td>
<td></td>
</tr>
</tbody>
</table>

### Comparing Software

<table>
<thead>
<tr>
<th>Big Data</th>
<th>EC</th>
<th>Extreme Computing</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Software responds to elastic resource demands</strong></td>
<td>After allocation, resources static until termination</td>
<td></td>
</tr>
<tr>
<td>Data access often <em>fine-grained</em></td>
<td>Data access is <em>large bulk</em> (aggregated) requests</td>
<td></td>
</tr>
<tr>
<td>Services are resilient to fault</td>
<td>Applications restart after fault</td>
<td></td>
</tr>
<tr>
<td>Often <em>customized</em> programming models</td>
<td>Widely <em>standardized</em> programming models</td>
<td></td>
</tr>
<tr>
<td>Libraries help <em>move computation to storage</em></td>
<td>Libraries help <em>move data to CPUs</em></td>
<td></td>
</tr>
<tr>
<td>Users routinely deploy their own services</td>
<td>Users almost never deploy customized services</td>
<td></td>
</tr>
</tbody>
</table>

### Comparing Data

<table>
<thead>
<tr>
<th>Scientific Big Data</th>
<th>EC</th>
<th>Extreme Computing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Inputs <em>arrive continuously</em>, streaming workflows</td>
<td>Inputs <em>arrive infrequently</em>, buffering carefully managed</td>
<td></td>
</tr>
<tr>
<td>Data is <em>unrepeatable</em> snapshot in time</td>
<td>Data often <em>reproducible</em> (repeat simulation)</td>
<td></td>
</tr>
<tr>
<td>Data generated by sensors <em>(error: from measurement)</em></td>
<td>Data generated from simulation <em>(error: from simulation)</em></td>
<td></td>
</tr>
<tr>
<td>Data rate <em>limited by sensors</em></td>
<td>Data rate <em>limited by platform</em></td>
<td></td>
</tr>
<tr>
<td>Data often <em>shared and curated</em> by community</td>
<td>Data <em>often private</em></td>
<td></td>
</tr>
<tr>
<td>Often <em>unstructured</em></td>
<td><em>Semi-structured</em></td>
<td></td>
</tr>
</tbody>
</table>
What can we use from EC for BD?

HPC Software A Good Base

- MPI-IO, HDF5, pnetCDF, HPSS, other ad hoc solutions provide good building blocks
- Needed: Better abstract models, for both high and low level abstractions
  - “DSL” for data manipulation at scale
  - Such systems are data structure + methods (operators)
- Implementations that fully exploit good and clean semantics of access

Interoperability

- HDF5 provides strong support for many aspects of data provenance. Mechanisms exist in pnetCDF.
  - Should a base set be “automatic”, much as file creation/modify time is today?
  - Can we evolve to better interoperability, or are radically new models needed?
- Mathematical representation for continuous data
  - How should the information about the mapping of discrete continuous be stored in the file?
  - How should this be generalized to other representations?
- Accuracy of data values
  - How should accuracy be efficiently stored with file?
- Data formats impact performance and scalability
  - Optimizing for interoperability or performance alone may impede application
  - You cannot pick the format and then (successfully) say “make it fast”

Architecture

- Architecture:
  - What architectural changes are needed for extreme computing storage systems to make them better suited for BD?
    - Better small scale atomic I/O – Solid State Storage?
    - A new storage repository – non POSIX?
    - Seamless storage hierarchies
  - What operational changes are needed to support new storage architectures?
    - Yes – critical resource is bandwidth not CPU
  - Looking at future technologies, what future architectures are possible?
    - Interconnect is the most essential. Processor technology can be whatever it is.
    - Energy efficient memory

Define Consistency Models for Access and Update

- Need consistency models that match use in applications
  - Or trade accuracy for speed
  - Already happened in search, e-commerce, even when solution is to trade accuracy for speed
    - Witness Amazon’s pseudo cart implementation – items aren’t really under your control (“in your cart”) until you complete the purchase. But greatly simplifies data model.
    - Even though it angers customers on popular deals
- POSIX consistency model is stronger than sequential consistency and almost never what applications require
  - Even when strong consistency is needed, it is almost always on the granularity of a data object, not bytes in a file
  - Long history of file systems falsely claiming to be POSIX
- A bad alternative is the “do what is fast” consistency model – usually but not always works
  - Some systems have taken this route – both I/O and RDMA
New NSF Project to Support This Kind of Research

(see Kate Keahey)

CHAMELEON: A POWERFUL AND FLEXIBLE EXPERIMENTAL INSTRUMENT

- Large-scale instrument
  - Targeting Big Data, Big Compute, Big Instrument research
  - ~650 nodes (~14,500 cores), 2 sites (IU and TACC) connected with 100G network, 5 PB of storage over two sites,
- Reconfigurable instrument
  - Bare metal reconfiguration, operated as single instrument
  - Infrastructure based on OpenStack+Ironic and Grid’5000 technology
- Connected instrument
  - Workload and Trace Archive
  - Partnerships with production clouds: CERN, OSDC, Rackspace, Google, and others
  - Partnerships with users: wrap up your framework as an experimental environment that others can use
- Complementary instrument
  - Complementing GENI, Grid’5000, and potentially other testbeds
Look at Results from Previous Breakout Summaries (exascale.org)

Original Charge: Work toward *identifying the research questions and promising directions*, not the answers, or how to spend other people’s money

- Gaps in the Core Facilities / software arch
  - No elasticity, VMs not well supported, Identity and sharing difficult, etc...
- Missing workflows and BDEC mini-apps for key communities, linking to instruments, etc.
- Apps: Data integration, analysis, classification
  - Cleaning, filtering, transforming, classification, mapping/registration, event detection, prediction
- Data
  - Shared analysis difficult, composition and workflow poorly supported, no benchmarks, etc.
White Papers

• Presenters:
  – White paper presenters will have 6 minutes for their presentations and allowed to have only 4 slides.
  – Please send a pdf file with your presentation your session chair before the beginning of your session.
Goals for This Meeting: How do we Converge?

• Prepare for an initial draft report
  – Barcelona: Material needed to write draft
  – Present initial draft at BOF at SC15

• Report (2 parts)
  – A) What are the current plans / strategies in Asia, Europe, and the US for handling Big Data?
  – B) How do we get BD/EC Convergence?
    • What is missing, specific to each of these respective regional plans? What could be a coordinated, converged plan for BDEC, at international level
Breakout Sessions

Introduction

• Breakout Questions to Help:
  – What are the main differences and commonalities between the HPC and BDA requirements/technologies/working-assumptions in this area?
  – Are there common needs/problems/interfaces could serve as the basis (or as stepping stones) along a path to (some reasonable level of) infrastructure and application convergence?
  – Are there interdomain testbeds that combine BDA and HPC workflows in ways that could help uncover pathways toward convergence?
  – What is/are the technology or new research that may be a game changer?
  – What action would be your number one priority to be taken rapidly to ensure success of the converge of Extreme computing and Big Data infrastructures?
  – What action would be your number one priority to be taken rapidly to ensure the emergence of efficient Extreme computing and Big Data applications?
  – How would you measure the success of the BDEC initiative?
Goals For This Meeting: How Do We Converge?

Breakout groups

Applications and Science
Chairs: David Keyes, Rosa Badia, Jean-Claude Andre

Architecture and Operation/Comprehensive Production Services
Chairs: Bill Kramer, Ewa Deelman, Francois Bodin

Algorithm and Applied Mathematics
Chairs: Hiroshi Nakashima, Philippe Ricoux, Alison Kennedy

Software Stack
Chairs: Franck Cappello, Kate Keahey, Satoshi Matsuoka