Breakout #2

Summary
Collab and Funding Models (1/2)

- Outline of what a plan would include, and possible outcomes
- Identify R&D models that enable laboratories, universities, and vendors to co-develop coordinated open source HPC software
- What is the value proposition for each international member?
- Examine funding and governance models that support international development
  - How are priorities set?
- How does the IESP fit into other international activities?
- What are the scientific grand challenges this project will address?
- Acceptance tests and support models that can be embraced by both vendors and the open source community
- Who assumes risk for hitting R&D targets? How do we handle failures?
- Educational Outreach
Possible Models (2/2)
(from loose to tight collaboration)

- Identify needs, focus Int’l R&D attention on missing components
- Coordinate features, delivery schedule, interoperability, and improvements across international R&D teams
- IESP community recommends funding for key areas
- Provide forums for vendors and community to work together on roadmaps
- Fund R&D and subsequent deployment of key components
- Fund collaborative relationship with vendors and co-develop components
- Test, integrate, and support internationally developed software components
- Build integrated software that can pass acceptance tests on extreme platforms
Breakout 2:

- **Science Driven**
  - Clear benefit to global challenges (Int’l science drivers that are shared)
  - Organizational involvement (advisory committee?)
  - Is there an existing (mirroring) effort on CompSci?
  - Are there

- We must clearly articulate why participating is advantageous to each country (agency)
  - What is the value proposition

- How do we participate and collaborate with the existing computer science and platform R&D efforts?

- Discussing nomenclature: “System Software”
  - Operating System ➔ math libraries, tools, etc
  - The components that are required for almost all apps
What collaboration models work best for delivering a supported system?

- We reviewed basic acceptance test and deployment plans
- Union of the needs of the first 3 customers defines the initial software stack
- For basic low-level software components (OS, networking, filesystems) local expertise and support is mandatory
- For other components (mpi, math libs, etc) grabbing the current version and porting can work
- Leverage with Open Source groups is key to providing good support, otherwise vendors provide best effort
- The Linux model works well
- We need to develop a sustainable transfer and support plan.
Breakout 2: Where to Invest

- Not all issues are HPC
  - Multicore programming model?
  - Parallel I/O?
  - Compilers?

- What flavor of HPC? We must be clear
  - Not Cloud? Not Grid?

- Where can we have the greatest impact?

- Our plan needs to clearly articulate where we are investing and why
Breakout 2: Collab Models

- Discussion of models that have worked
- HPC Europe Taskforce

- Distributed software development is difficult, but successful models exist
  - Functional decomposition?

- DARPA HPCS experiences:
  - Down-selecting is ok, but transferring responsibility to vendors probably not applicable here

- Lead agency or team required to force it to happen
- Multi-agency support is very valuable
Breakout 2:

- Quality Assurance: Our plan must include how the software will be tested and deployed.
- Verification and Validation discussion.
  - What parts are IESP goals, and what is the responsibility of the application teams?
Breakout 2: Revolutionary / Evolutionary

- Clearly define impact timeline
- They are not mutually exclusive