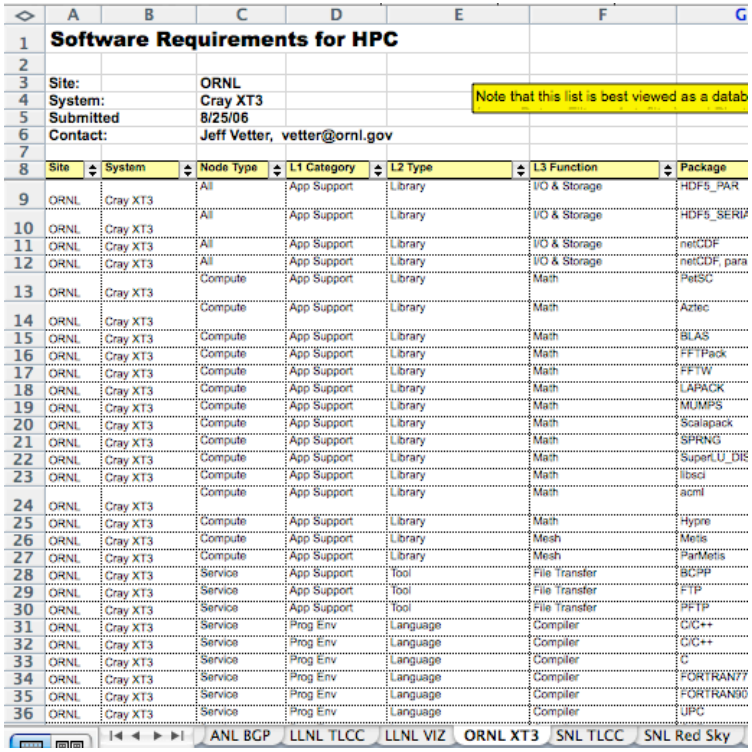
Results from Santa Fe

Pete Beckman

Optimism, Excitement
Good agreement on several themes

- International collaboration is required to address scale of challenge
- Research community and vendors must work together
- Open Source is vital component
- Large changes in platforms expected



Software Requirements for HPC

Site: ORNL
System: Cray XT3
Submitted 8/25/06
Contact: Jeff Vetter, vetter@ornl.gov

Note that this list is best viewed as a datab

Site	System	Node Type	L1 Category	L2 Type	L3 Function	Package
ORNL	Cray XT3	All	App Support	Library	I/O & Storage	HDF5_PAR
ORNL	Cray XT3	All	App Support	Library	I/O & Storage	HDF5_SERIA
ORNL	Cray XT3	All	App Support	Library	I/O & Storage	netCDF
ORNL	Cray XT3	All	App Support	Library	I/O & Storage	netCDF, para
ORNL	Cray XT3	Compute	App Support	Library	Math	PetSC
ORNL	Cray XT3	Compute	App Support	Library	Math	Aztec
ORNL	Cray XT3	Compute	App Support	Library	Math	BLAS
ORNL	Cray XT3	Compute	App Support	Library	Math	FFTPack
ORNL	Cray XT3	Compute	App Support	Library	Math	FFTW
ORNL	Cray XT3	Compute	App Support	Library	Math	LAPACK
ORNL	Cray XT3	Compute	App Support	Library	Math	MUMPS
ORNL	Cray XT3	Compute	App Support	Library	Math	Scalapack
ORNL	Cray XT3	Compute	App Support	Library	Math	SPRNG
ORNL	Cray XT3	Compute	App Support	Library	Math	SuperLU_DIS
ORNL	Cray XT3	Compute	App Support	Library	Math	libsci
ORNL	Cray XT3	Compute	App Support	Library	Math	acml
ORNL	Cray XT3	Compute	App Support	Library	Math	Hypre
ORNL	Cray XT3	Compute	App Support	Library	Mesh	Metis
ORNL	Cray XT3	Compute	App Support	Library	Mesh	ParMetis
ORNL	Cray XT3	Service	App Support	Tool	File Transfer	BCPP
ORNL	Cray XT3	Service	App Support	Tool	File Transfer	FTP
ORNL	Cray XT3	Service	App Support	Tool	File Transfer	PFTP
ORNL	Cray XT3	Service	Prog Env	Language	Compiler	C/C++
ORNL	Cray XT3	Service	Prog Env	Language	Compiler	C/C++
ORNL	Cray XT3	Service	Prog Env	Language	Compiler	C
ORNL	Cray XT3	Service	Prog Env	Language	Compiler	FORTRAN77
ORNL	Cray XT3	Service	Prog Env	Language	Compiler	FORTRAN90
ORNL	Cray XT3	Service	Prog Env	Language	Compiler	UPC

ANL BGP | LLNL TLCC | LLNL VIZ | ORNL XT3 | SNL TLCC | SNL Red Sky

Exascale Target Schedule

- One possible platform schedule, based on strong and continued investment:
- 2012: 10-30pf
- 2015: 100-300pf
- 2018: 1000pf

- What kind of Exaflop?
  - One that can do large multi-scale problems
  - And great dense linear algebra
  - A possible heterogeneous core

Software Barriers

- Evolution or Revolution

- Key Challenges:
  - ***Concurrency***
  - ***Energy (Power)***
  - ***Resilience (Fault tolerance)***

Exploration (1/4)

- Revolutionary storage class memory is coming:
  - 1000x < latency
- A power envelope will become part of design framework

IBM Research

Actions we can take

- Value of storage class memory: Need to have the HPC community united in articulating the value proposition associated with storage class memory. The critical break points in terms of bandwidth, density, cost and latency need to be understood to help guide the technology development.
- Power : This is somewhat a mindset change. Applications will eventually need to think of their computing resource as a total energy budget and they need to optimize within this. Fortunately much of performance tuning also drives toward energy efficiency… but not always. Tools and reports that detail the energy usage need to be accessible to users.
- Reliability: The realistic adoption, cost and risk of fault tolerant algorithms must be assessed and these should be traded off against hardware cost and risk. The systems can not move in a direction that "might" be acceptable from a reliability perspective. This makes a software solution very difficult.

Alan Gara © 2009 IBM Corporation

IBM Research

Revolutionary Software Areas for Exascale:

- Storage class memory is coming: Technology will offer 1000x less latency but there are many other dimensions to this. Need to think through the possible directions to use this technology.
- Systems are transitioning to being power optimized. Application developers are still focused on performance optimization regardless of power. In a world where there is a power budget, software should play a role in optimizing performance through optimization of Perf/Watt. (with total power being a hard facility constraint)
- Reliability: This is not an issue of what will we do when systems can not be made reliable. This issue is making the best trade-offs between hardware, system software and fault tolerant applications.

Alan Gara © 2009 IBM Corporation

IBM Research

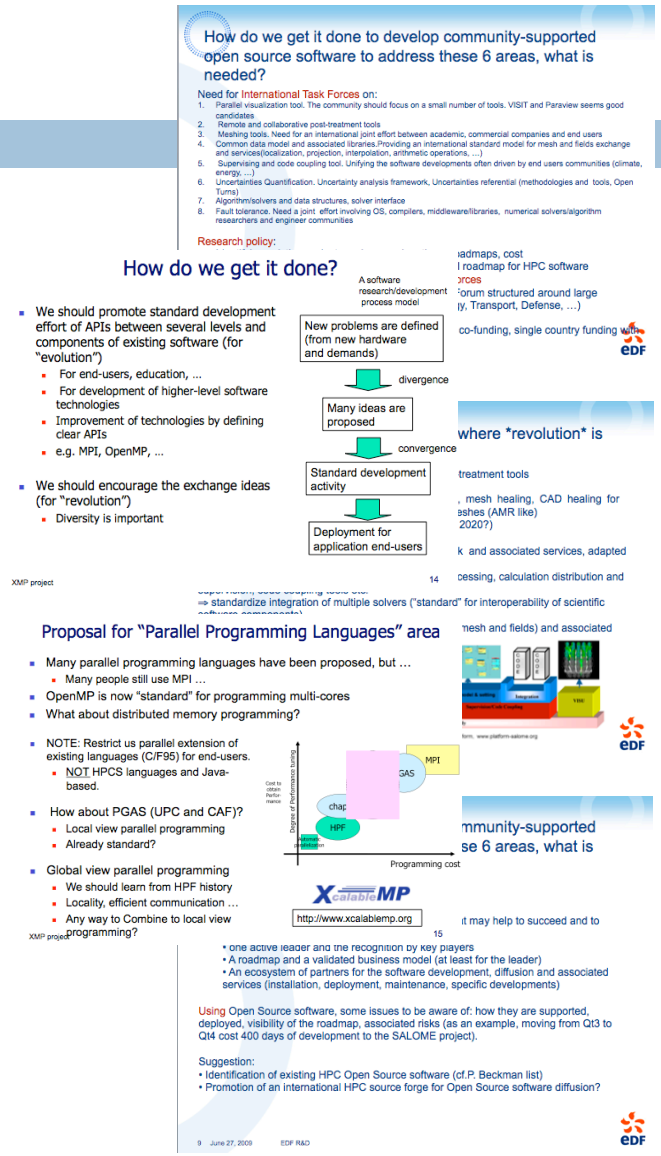
Evolutionary Software Areas for Exascale:

- Extend current program models through single node threading of messaging. Eliminate "per task" scaling terms in messaging layer to allow for higher "flat scaling".
- Allow for mixed programming models to coexist. We need a bridge to new programming models that is not an all or nothing proposition.
- Enhance job flow to enable many concurrent capability scale jobs. (similar to the emerging approach at LLNL) This is likely to be a common early usage model for Exascale.
- Open source can be a very good thing for vendors and end users but we need to find a way share the responsibility and …… risk.
- Educate young people in parallel programming.

Alan Gara © 2009 IBM Corporation

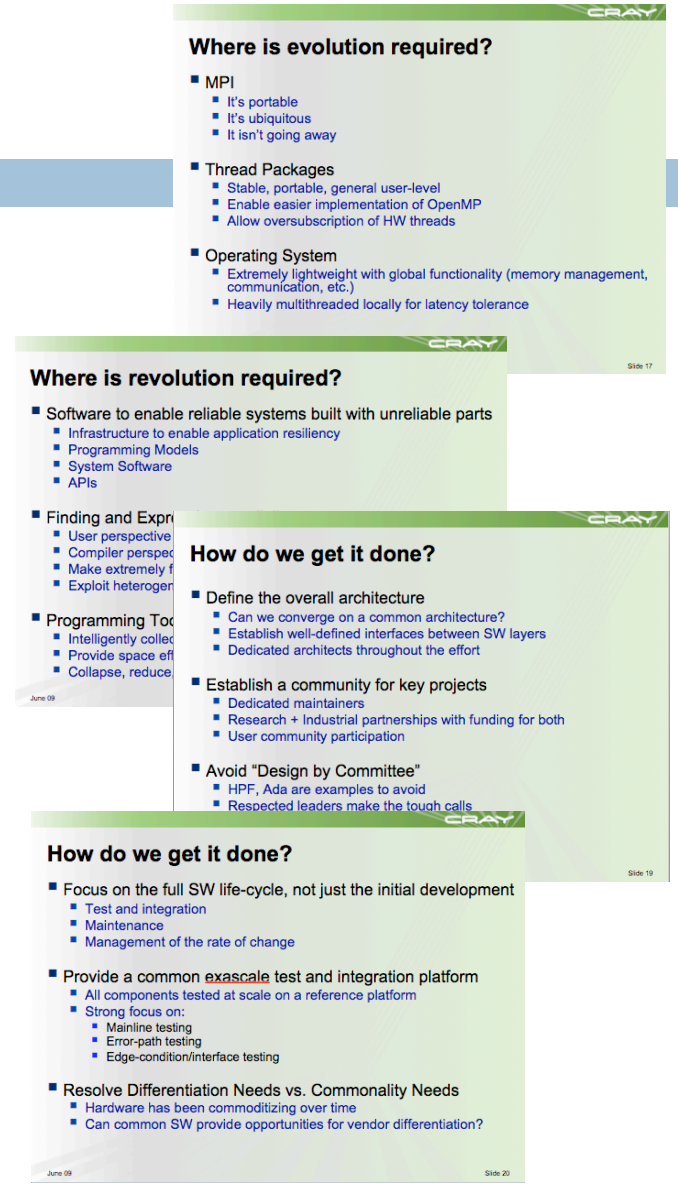
Exploration (2/4)

- Revolution in multi-scale/multi-physics simulation frameworks
- Revolution in visualization and post-processing
- Need new programming models for hybrid programming
- Need unification for hetero-core programming models



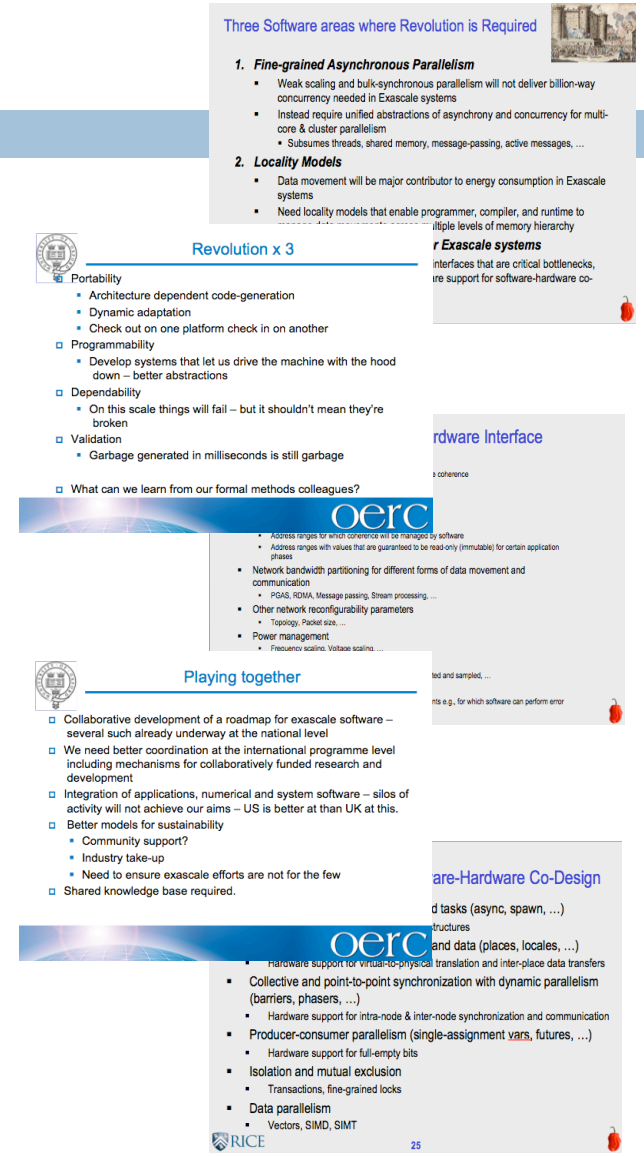
Exploration (3/4)

- Revolution Needed
  - Reliability
  - Expressing Parallelism
  - Programming Tools
- Other issues
  - Full SW life cycle
  - Product differentiation
  - Testing and development practices



Exploration (4/4)

- Revolution:
  - Fine-grain Async parallelism
  - Locality
  - Co-design HW/SW
- Follow best-practices from community software development
- Validation and Uncertainty Quantification important



Non-architectural Issues

- Science drivers must be closely tied to effort
- What funding and governance model are most supportive?
- How do we set priorities?
- Who assumes risk for hitting R&D targets?
  - How do we handle failures?
- More integrated testing and development vital
- Close communication with coordination with funding agencies is required for success