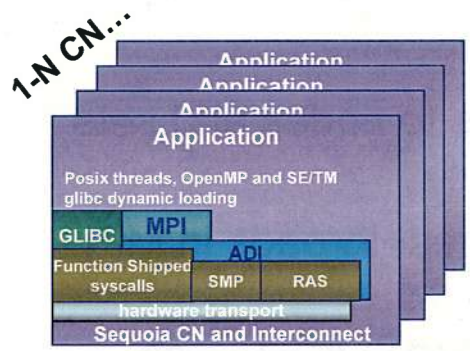
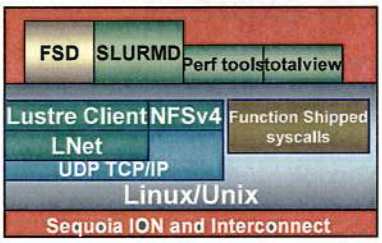
Sequoia will scale utilizing hierarchal software model



Light weight kernel on compute node

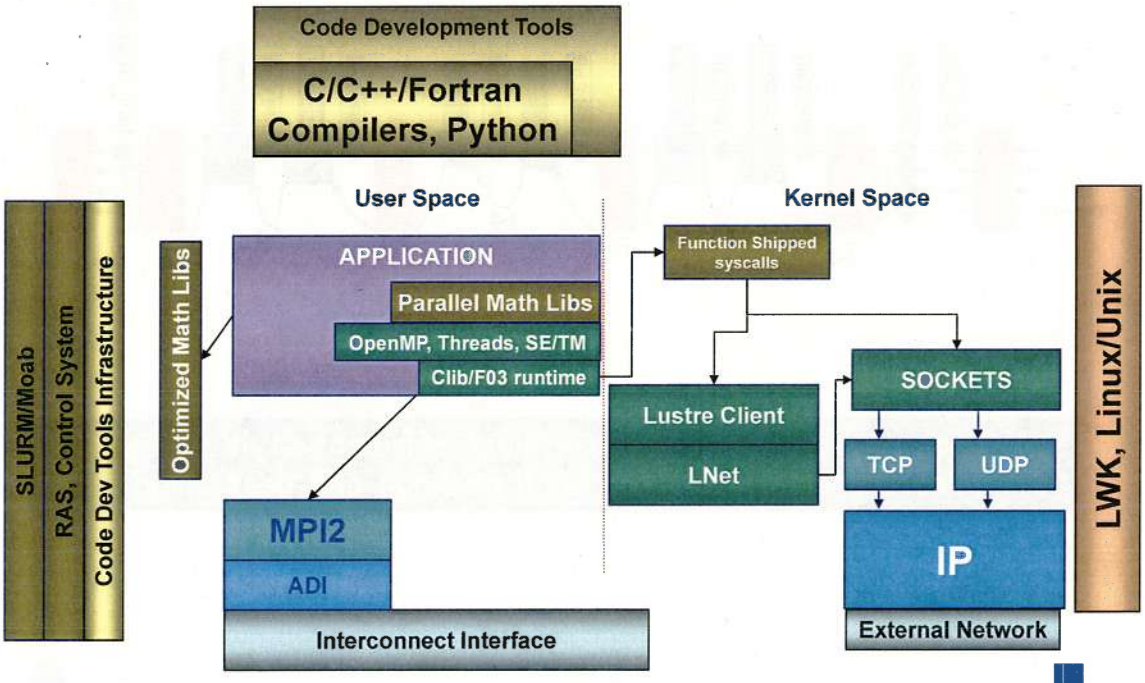
- Optimized for scalability and reliability
  - As simple as possible. Full control
  - Extremely low OS noise
  - Direct access to interconnect hardware
- OS features
  - Linux syscall compatible with IO syscalls forwarded to I/O nodes
  - Support for dynamic libs runtime loading
  - Shared memory regions
- Open source



Linux on I/O Node

- Leverage huge Linux base & community
  - Enhance TCP offload, PCIe, I/O
- Standard File Systems Lustre, NFSv4, etc
- Factor to Simplify:
  - Aggregates N CN for I/O & admin
- Open source

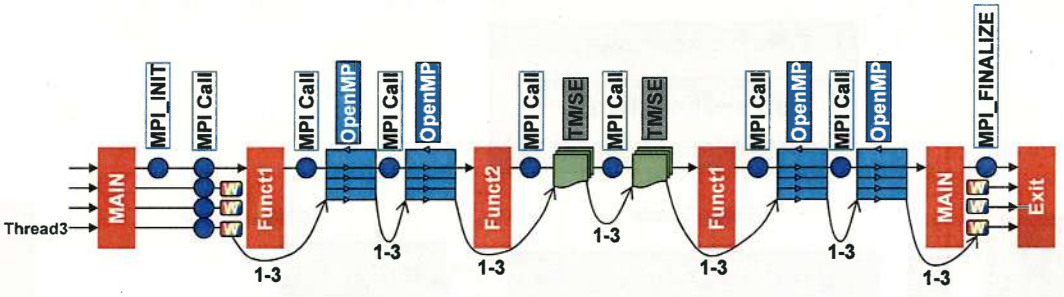
Sequoia Software Stack – Applications Perspective



Application Programming Model Requirements

- MPI Parallelism at top level
  - Static allocation of MPI tasks to nodes and sets of cores+threads
- Effectively absorb multiple cores+threads in MPI task
- Support multiple languages: C/C++/Fortran03
- Allow different physics packages to express node concurrency in different ways

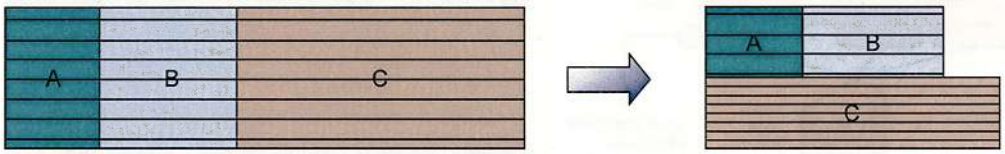
Sequoia's programming model is a simple extension beyond MPI with flexible mechanisms to absorb cores and threads



Weapon physics codes can use most efficient style of multi-core programming for each package and nest them

New approach to parallelization: apply multiple approaches to parallelism at the code and package levels

- Utilize the optimal parallelism methodology for each package
  - Nested Node Concurrency programming model allows different packages to exploit SMP parallelism differently
  - OpenMP, Pthreads and SE/TM available
- Run packages within a code in parallel
  - Can absorb appropriate number of nodes to load level the application



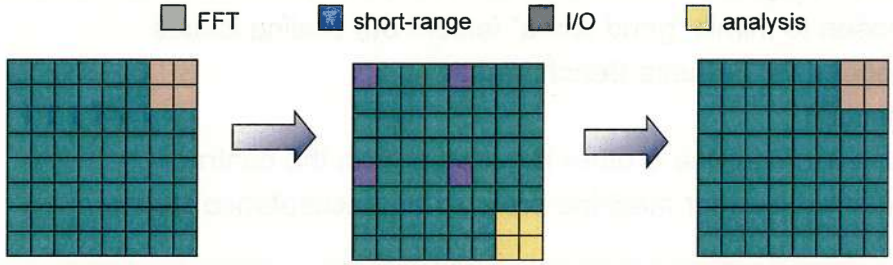
Inefficient: packages A,B,C are run sequentially, with sub-optimal level of parallelism

High-performance: compute-heavy package C is run concurrently with packages A and B

Availability of Dawn as a robust testbed and support of IBM collaboration allows development of novel parallelization model and implementations

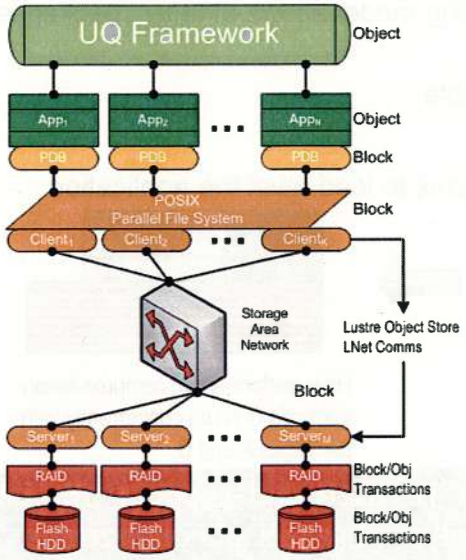
One example: novel parallelization in ddcMD/plasma

- Plasma modeling requires efficient handling of both short-range and long-range interactions
- Long range interactions are typically calculated using reciprocal space (FFT) methods, which do not scale well



Solution:
- majority of nodes are used for short-range forces that scale extremely well
- use just enough MPI tasks to minimize the communication bottleneck for FFT
- use local threads to efficiently calculate FFT on small number of nodes

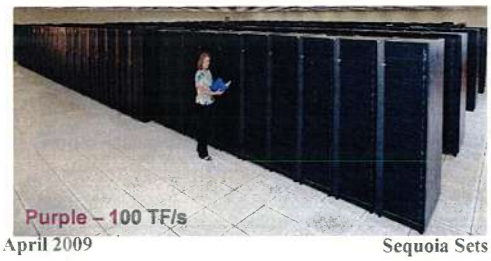
Thinking about applications in the UQ Framework allows new approaches to solving old problems



- Contextual information within the UQ framework and applications could be transmitted through IO path to provide a systematic approach to scientific data management
- Current parallel file systems with block oriented interfaces currently don't encourage accumulating contextual information

Sequoia Platform Target Performance is a Combination of Peak and Application Sustained Performance

- "Peak" of the machine is absolute maximum performance
  - FLOP/s = FLoating point OPeration per second
- Sustained is weighted average of five "marquee" benchmark code "Figure of Merit"
  - Four IDC package benchmarks & one "science" benchmark from SNL
  - FOM chosen to mimic "grind times" factors out scaling issues
- Three purposed for Sequoia Benchmarks
  - RFP selection
  - Bone fides for Marquee & other requirements in the contract
  - Synthetic Workload for machine pre-ship and acceptance testing

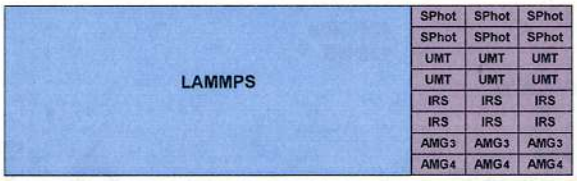


The marquee benchmark strategy for aggregating performance incentivizes IBM in two ways: scalability and throughput

AMG	wFOM = A x "solution vector size" * iter / sec
IRS	wFOM = B x "temperature variables" * iter / sec
SPhot	wFOM = C x "tracks" / sec
UMT	wFOM = D x corners*angles*groups*zones * iter / sec
LAMMPS	wFOM = E x atom updates / sec

Aggregate wFOM = $wFOM_{AMG}$ + $wFOM_{IRS}$ + $wFOM_{SPhot}$ + $wFOM_{UMT}$ + $wFOM_{LAMMPS}$

- **Applications weights**
  - **Normalize the benchmarks to each other on reference systems**
  - **All benchmarks are of equal importance**
  - **Based on the targets of 24X Purple IDC & 20X BG/L Science**



This benchmarking strategy assures Sequoia will deliver *both* UQ and Science to the Stockpile Stewardship Program

Performance Projections for Marquee Benchmarks Delivers on Program Goals for Sequoia

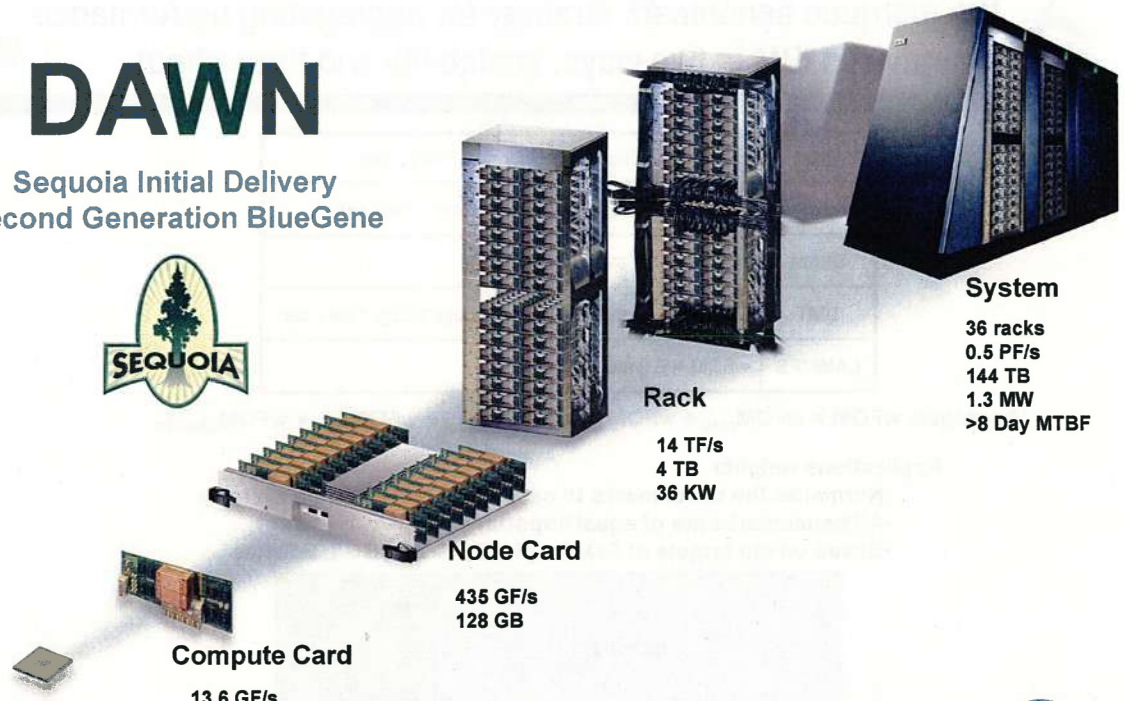
M = P + S = 20.0 + 28.3 = 48.3

Benchmark	Raw FOM ratio of BGQ relative to Purple/BGL node	Est. Speed Factor From 6 copies	
AMG	**1.07**	**6.4**	**Aggregate Target 24X Est. 36.4X**
IRS	**1.95**	**11.7**	
SPhot	**1.82**	**10.9**	
UMT	**1.23**	**7.4**	
LAMMPS	**18.0**	**20.3**	**Target 20X**

Multi-physics runs on 1,024 BGQ nodes with 8MPI/8OMP and 1,024 Purple nodes w/8MPI
LAMMPS run on 72K BGQ nodes with 16MPI/4OMP and 64K BGL nodes w/2MPI

DAWN

Sequoia Initial Delivery
Second Generation BlueGene



System
36 racks
0.5 PF/s
144 TB
1.3 MW
>8 Day MTBF

Rack
14 TF/s
4 TB
36 KW

Node Card
435 GF/s
128 GB

Compute Card
13.6 GF/s
4.0 GB DDR2
13.6 GB/s Memory BW
0.75 GB/s 3D Torus BW

Chip
850 MHz PPC 450
4 cores/4 threads
13.6 GF/s Peak
8 MB EDRAM

Dawn acceptance is complete and early science runs have commenced

- Dawn hardware delivery started 19 Jan 2009
- Rapid deployment of 36 racks completed ahead of an aggressive schedule
- Full Synthetic Workload acceptance test successfully completed 26 March 2009
- Initial full system science runs currently underway
- Transition to classified service mid July, depending on science run demands

The first half of DAWN (initial delivery of Sequoia) was received at the TerascaleSimulation Facility in late January, 2009

Scalable Applications Preparation (SAP) Project assists code teams in fully exploiting Sequoia capability

- Training and seminars on key technologies for multi-core programming

- Leverage: PCET LDRD, LLNL/ANL R&D partnership to accelerate Sequoia First Wave Applications

- User Guide and Performance Tuning Documentation developed by LC User Training and Hotline staff

- Engagement of Tri-Lab code teams with site visits for training, workshops, and regular video conferences

- Use Dawn, hardware and software simulators to provide early access to new technologies for Sequoia