High Performance Computing at Livermore, Petascale and Beyond

Presentation to:
LSD, May 2, 2008
Livermore, CA

Dr. Mark K. Seager

ASC Platforms Lead at LLNL

UCRL-PRES-403698
This work performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344.

Talk Overview

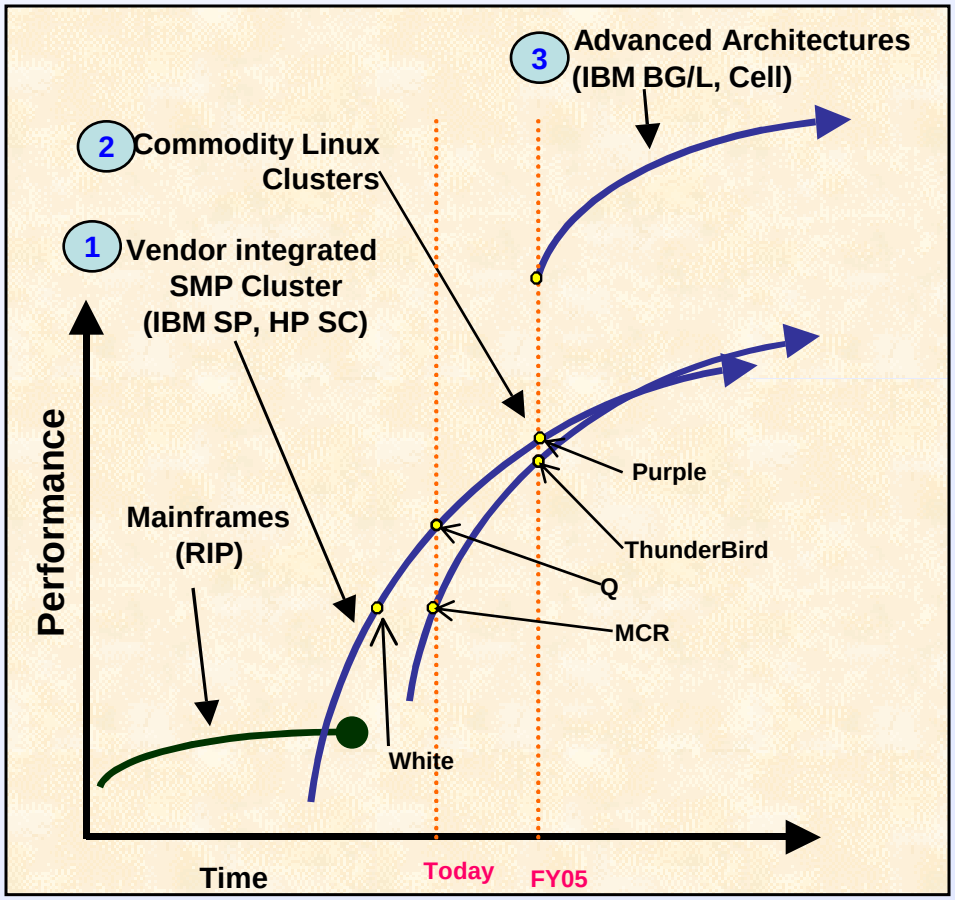
- Current Computing Environments at LLNL
  - Systems & Simulation Environment
- Peta and Exascale Programmatic Drivers
  - Weapons Physics for Know Unknowns
  - Uncertainty Quantification for Existing Stockpile
- Sequoia Target Architecture
- Comments on evolutionary changes required for petascale programming model

Our platform strategy is to straddle multiple technology curves to appropriately balance risk and cost/performance benefit

Three complementary curves…

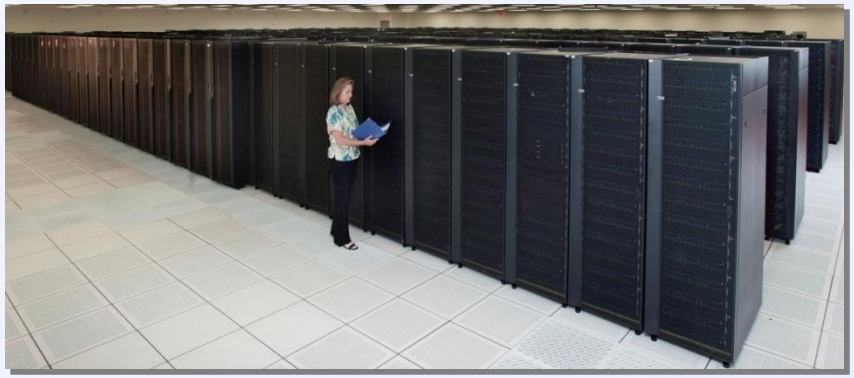
1. Delivers to today's stockpile's demanding needs
   - Production environment
   - For "must have" deliverables now
2. Delivers transition for next generation
   - "Near production", riskier environment
   - Capability system for risk tolerant programs
   - Capacity systems for risk averse programs
3. Delivers affordable path to petaFLOP/s
   - Research environment, leading transition to petaflop systems?
   - Are there other paths to a breakthrough regime by 2006-8?

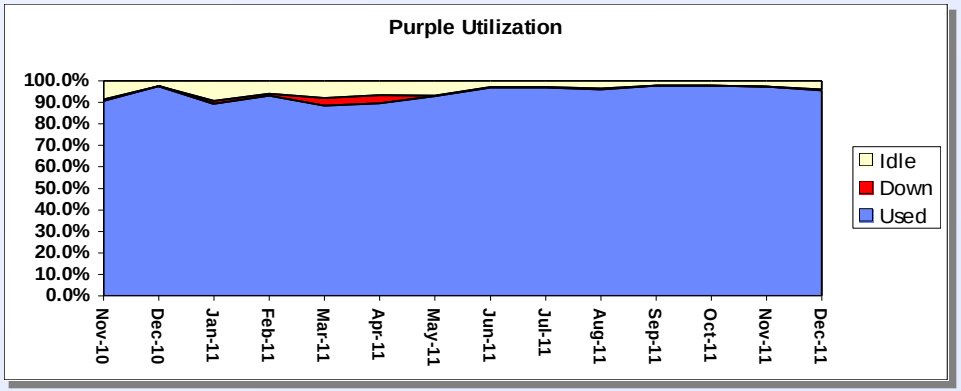
Any given technology curve is ultimately limited by Moore's Law



NNSA has a sterling record of delivering production computers, setting the standard for national supercomputing user facilities

ASC White [2001]: First shared tri-laboratory resource came in 23% over required peak



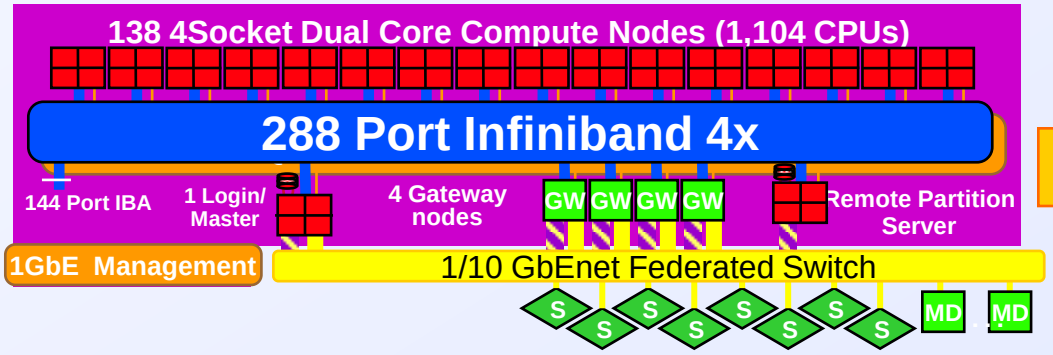


ASC Purple [2005]: First NNSA User Facility came in $50M under budget and is routinely 90+% utilized

Linux Clusters Curve 2 leveraged strategic tri-labs institutional investments to create huge efficiency for institution and programs with scalable COTS and Open Source SW approach

- LLNL institutional investments filled gaps in Linux cluster technology
  - LLNL – Early Lustre adoption leading to integrated simulation environment
  - LLNL – SLURM and Linux Distro development for manageability
  - LLNL – Thunder was TOP500(23) #2
- Real world synergy
  - LLNL Peloton SUs procurement for three institutional clusters
  - Enabled >50% TCO improvement for ASC

Minos - 6 SUs - 33TF



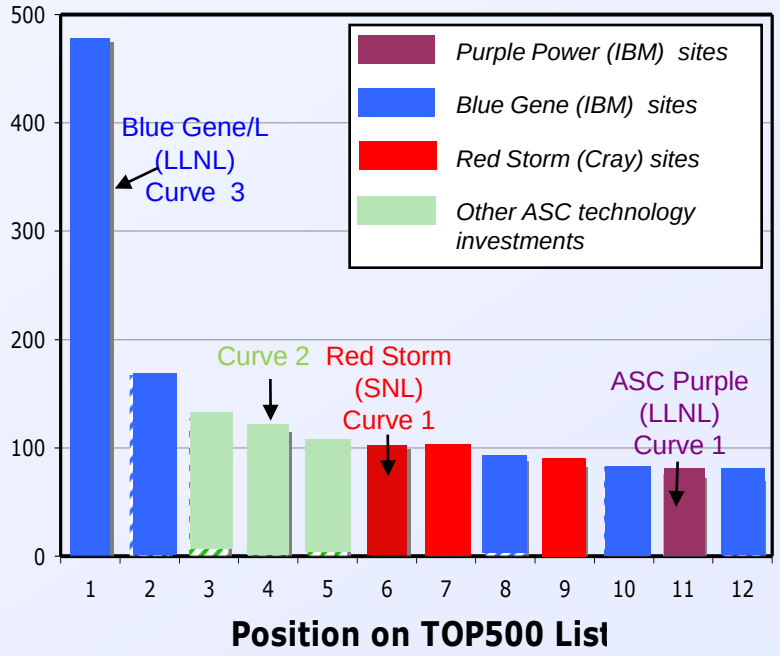
One Scalable Unit building block

Peloton is not a computer - it is a procurement and a way of thinking aggressively about achieving TCO efficiency

America's dominance in simulation today is the result of NNSA's investments and its astute technical choices

Today's Top 12 supercomputers *depended on NNSA investments*

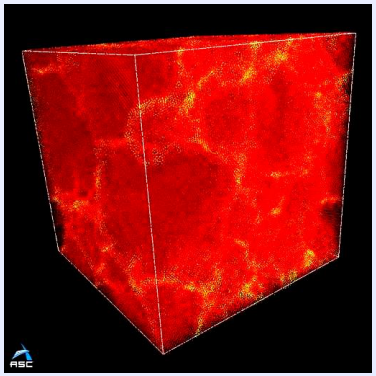
37% of the Top 500 systems today *depended on ASC technology investments*

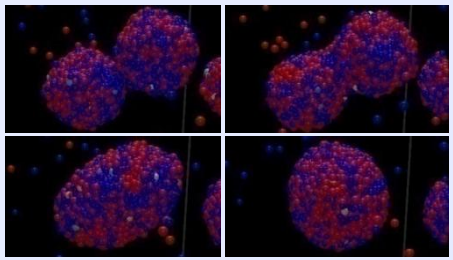


BG/L is #1 for the seventh consecutive Top 500 list

Four Gordon Bell prizes in three years

World's first low-power consuming supercomputer reduces energy footprint by 75%

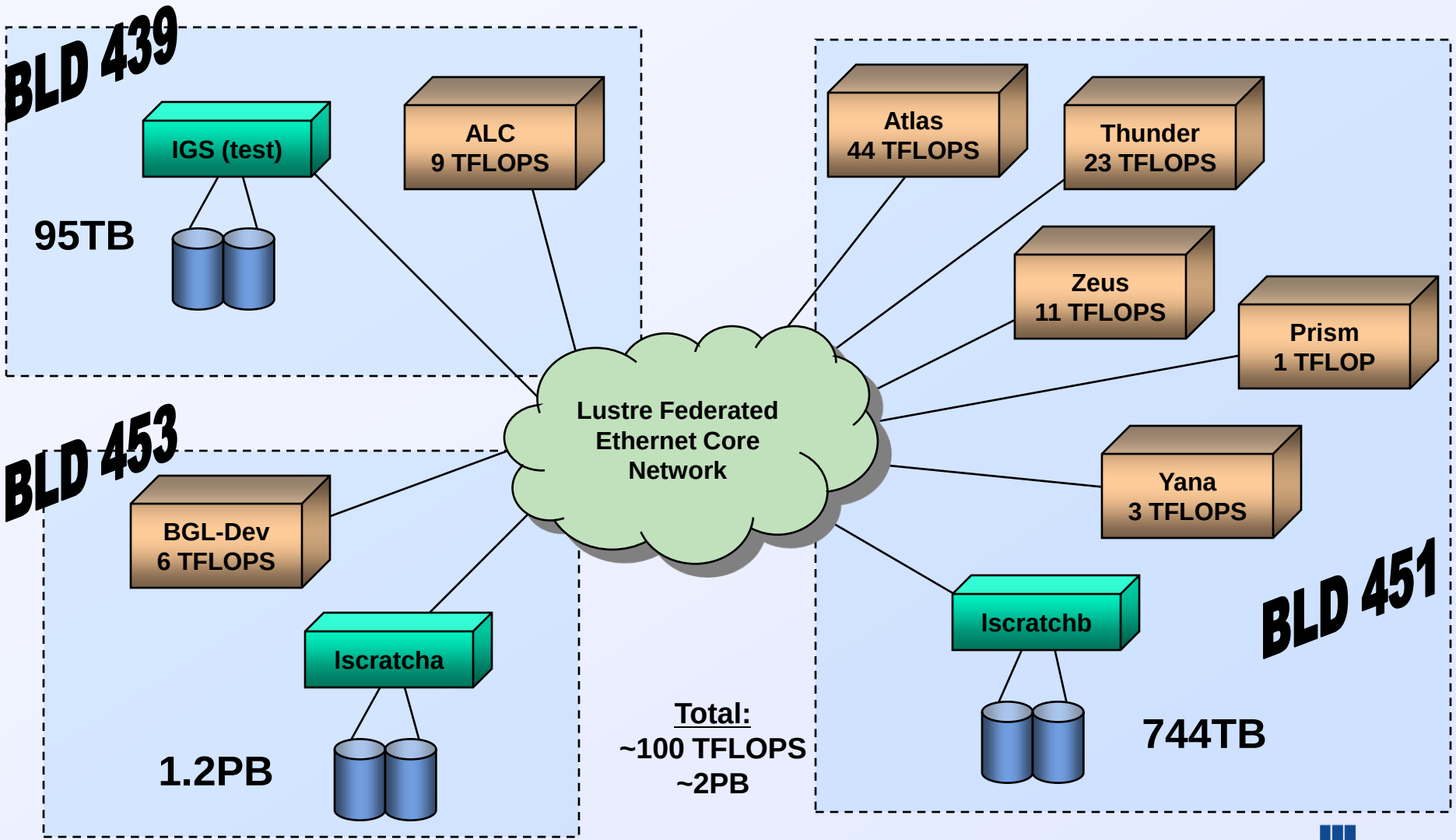




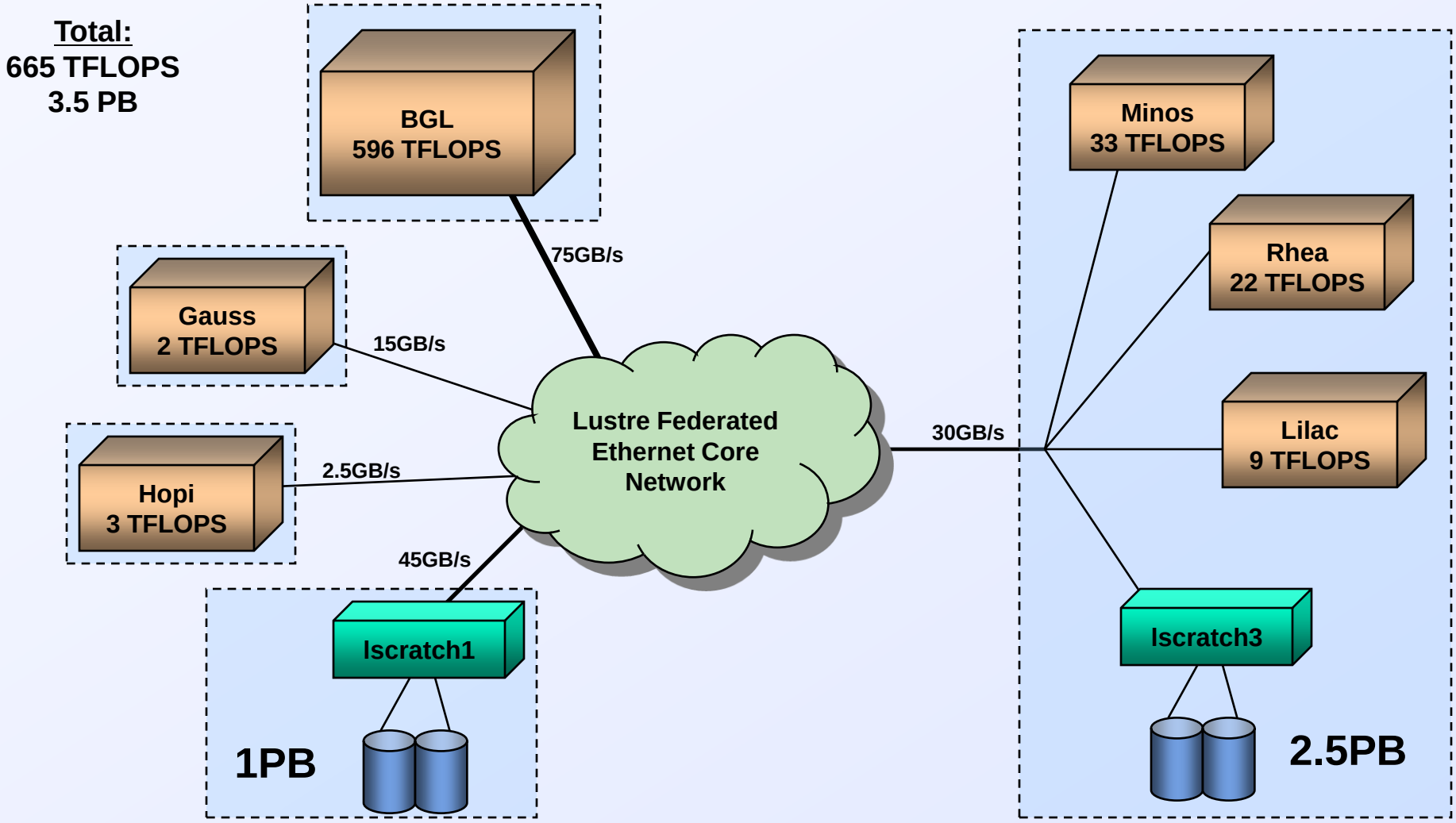
Livermore Currently Has the Worlds Best Computing Infrastructure, by a Wide Margin

System	Top500 Rank	Program	Manufacturer / Model	OS	Interconnect	Nodes	CPUs	Memory (GB)	Peak TFLOP/s
Unclassified Network (OCF)									**98.37**
uBG/L	*TBD*	*ASC*	*IBM*	*Linux*	*IBM*	*32,768*	*65,536*	*17,408*	*183.50*
Atlas (Peloton)	29	M&IC	Appro	Linux	IB DDR	1,152	9,216	18,432	44.24
Thunder	47	M&IC	California Digital	Linux	Elan4	1,024	4,096	8,192	22.94
Zeus (Peloton)	241	M&IC	Appro	Linux	IB DDR	288	2,304	4,608	11.06
ALC	414	ASC	IBM xSeries	Linux	Elan3	960	1,920	3,840	9.22
uPurple		ASC	IBM SP	AIX	Federation	108	864	3,456	6.57
Yana		M&IC	Appro	Linux	N/A	80	640	1,600	3.07
Prism		ASC	GraphStream	Linux	IB SDR	128	256	2,048	1.23
Snowbert		M&IC	IBM SP	AIX	Colony	8	64	32	0.06
Classified Network (SCF)									**978.05**
BlueGene/L	1	ASC	IBM	Linux	IBM	106,496	212,992	69,632	596.38
BlueGene/L v3	*TBD*	*ASC*	*IBM*	*Linux*	*IBM*	*65,536*	*131,072*	*49,152*	*367.00*
Purple	11	ASC	IBM SP	AIX	Federation	1,532	12,288	49,152	93.39
Juno (TLCC)	*TBD*	*ASC*	*Appro*	*Linux*	*IB DDR*	*1,152*	*18,432*	*36,864*	*162.20*
Eos (TLCC)	*TBD*	*ASC*	*Appro*	*Linux*	*IB DDR*	*288*	*4,608*	*9,216*	*40.55*
Minos (Peloton)	38	ASC	Appro	Linux	IB DDR	864	6,912	13,824	33.18
Rhea (Peloton)	61	ASC	Appro	Linux	IB SDR	576	4,608	9,216	22.12
Lilac	449	ASC	IBM xSeries	Linux	Elan3	768	1,536	3,072	9.19
UM		ASC	IBM p655	AIX	Federation	128	1,024	2,048	6.14
UV		ASC	IBM p655	AIX	Federation	128	1,024	2,048	6.14
Hopi		ASC	Appro	Linux	N/A	80	640	1,488	3.07
Gauss		ASC	GraphStream	Linux	IB SDR	256	512	3,072	2.46
Ace		ASC	Rackable Systems	Linux	N/A	176	352	704	1.97
Queen		ASC	Rackable Systems	Linux	N/A	63	126	252	0.71
Tempest		ASC	IBM Power5	AIX	N/A	12	84	480	0.55

Open Computing Facility Simulation Environment is Based on Lustre



Secure Computing Facility Simulation Environment Replicates Many Aspects of the OCF



Predictive simulation roadmap through exascale

DP Goals:

Develop and demonstrate capability to certify aging weapons with codes calibrated to past UGTs

Enables

Certify LEPs and RRWs (near neighbors to the test base)

Enables

Assess & certify *without* requiring reliance on UGTs

ASC Goals:

- Create 3D integrated design codes
- Calibrate codes with AGEX and UGT data

- Ad hoc factors replaced by physics based models
- Transition to quantified 3D predictive capability

- Validated predictive capabilities
- First principles/atomistic calculations of key physics
- Design with new materials or components in months

Principal uncertainties:

Energy Balance

Initial Conditions For Boost

Boost

Secondary Performance

Technical need for future nuclear tests eliminated

96 97 98 99 00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26

Red Storm 124

Purple 100

BG/L 590

BG/L 360

Roadrunner

Dawn

14-20PF

150PF

1EF

Computing Power:

Terascale systems

Petascale systems

Exascale systems

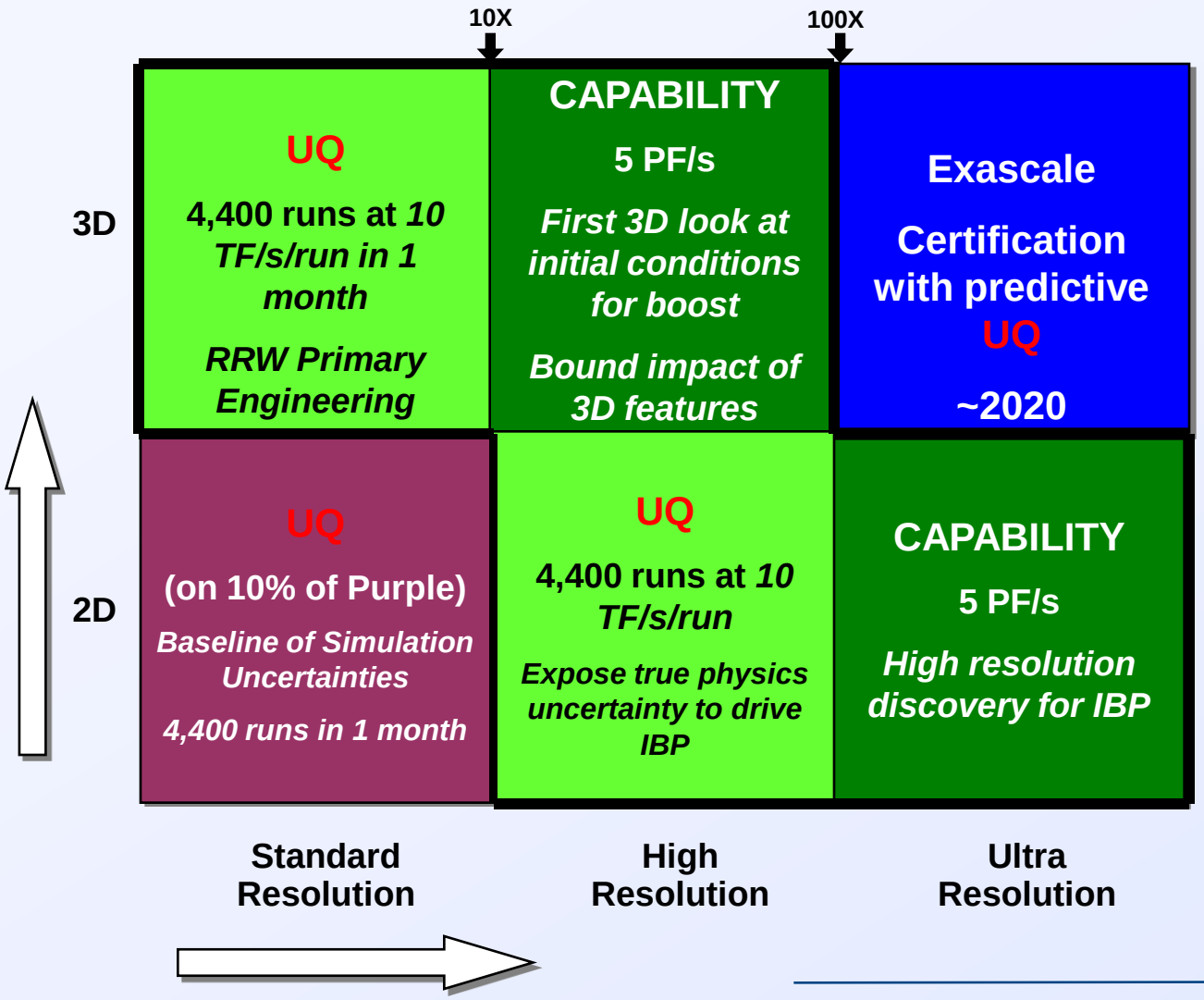
Uncertainty Quantification

Predicting stockpile performance drives five separate classes of petascale calculations

1. Quantifying uncertainty (for all classes of simulation)
2. Identify and model missing physics (e.g., boost)
3. Improving accuracy in material property data
4. Improving models for known physical processes
5. Improving the performance of complex models and algorithms in macro-scale simulation codes

Each of these mission drivers require petascale computing

Sequoia is the key integrating tool for Stockpile Stewardship in 2011-2015 time frame



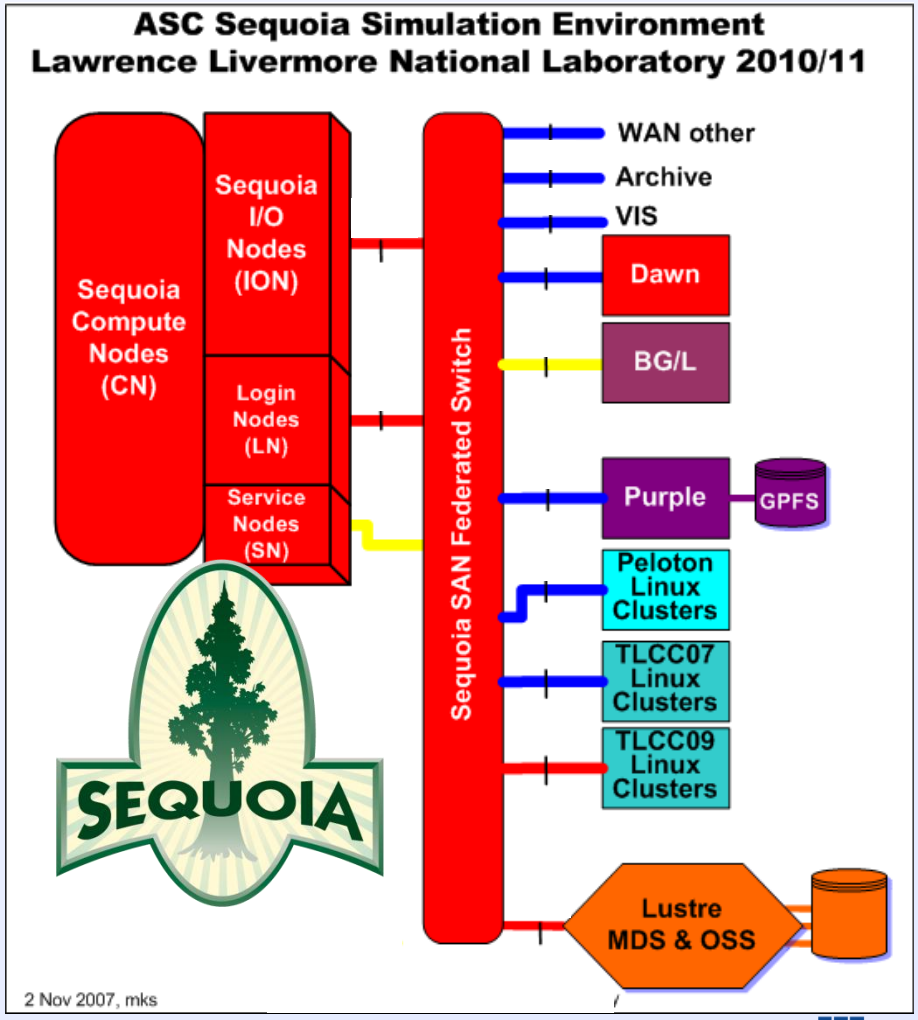
Sequoia will provide credible UQ for stockpile certification

- **Execution of Integrated Boost Program (IBP) requires Sequoia – IBP can't be done on Purple**
- **IBP is necessary to achieve certification with predictive UQ**

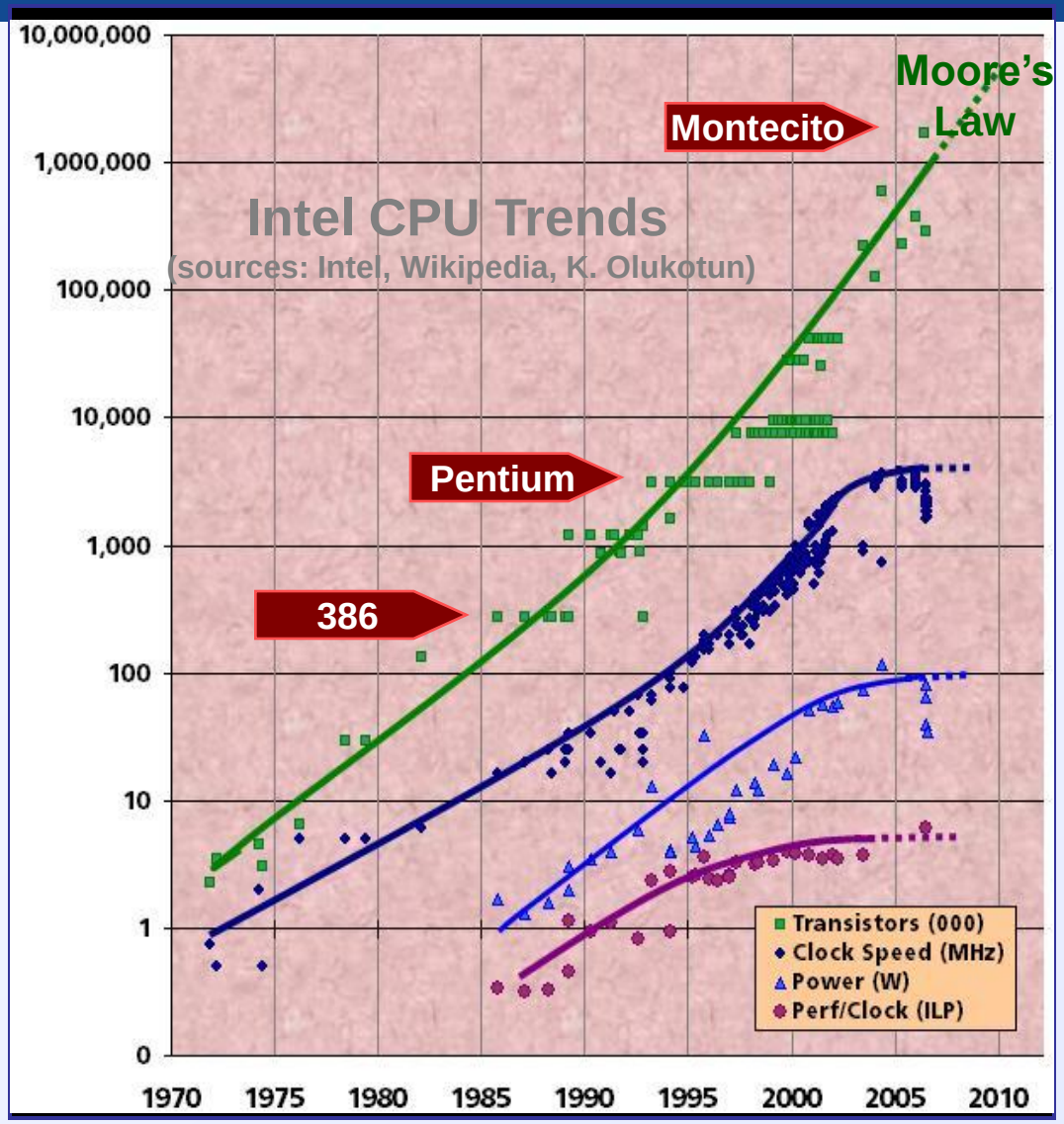
Sequoia Target Architecture Leverages LLNL Success with Five Generations of ASC Platforms

- Sequoia Peta-Scale Architecture

  - Smoothly integrates into LLNL environment
  - Architecture suitable for weapons science and Integrated Design Codes (IDC)
    - 24x Purple for design codes in support of certification
    - 20-50x for weapon science requirements
  - Risk Reduction woven into every aspect of Sequoia
    - Associated R&D contracts accelerate technology development
    - Targets and Off-ramps allow innovation with shared risk
  - ASC HQ supports: $250M Total Project Cost
    - Sequoia, ID system and R&D



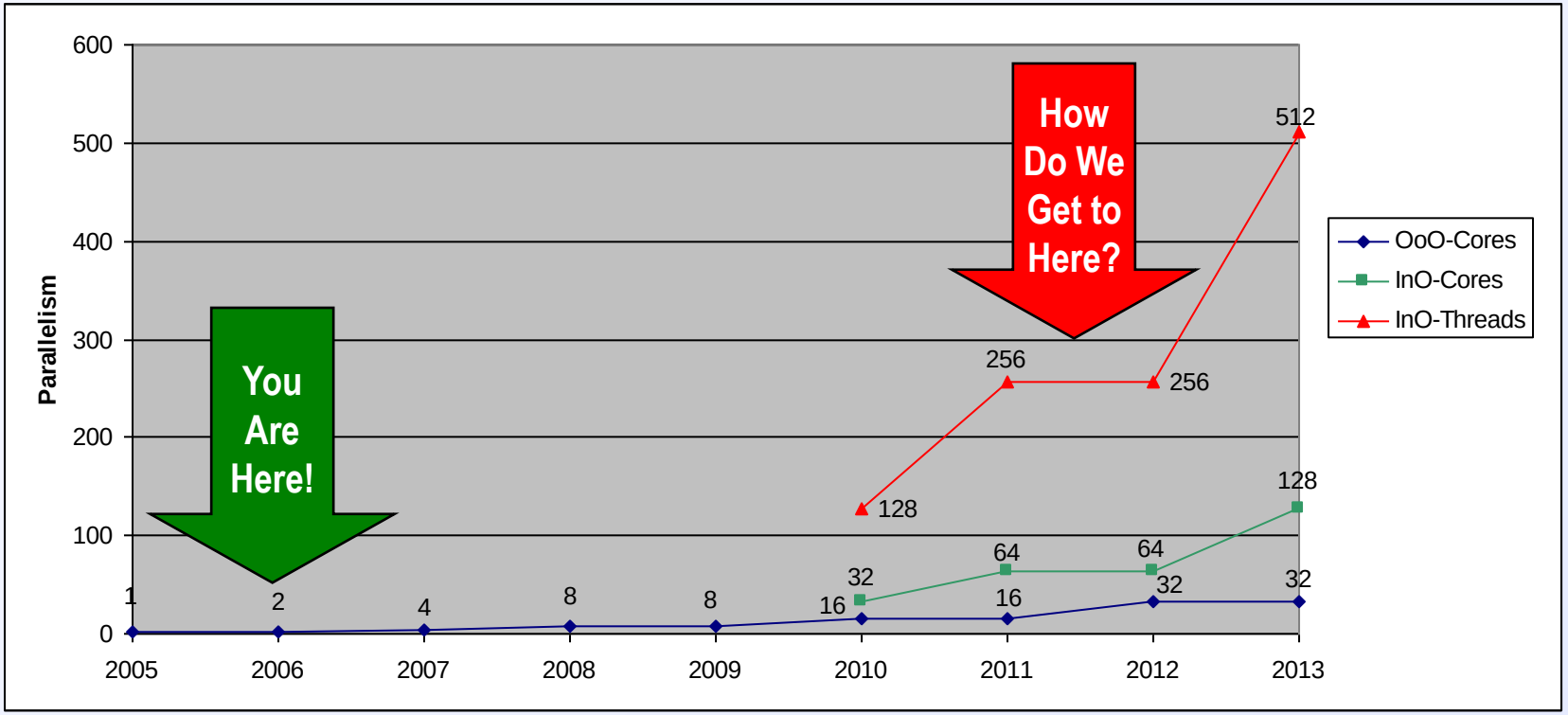
Each year we get ~~faster~~ more processors



- Historically: Boost single-stream performance via more complex chips, first via one big feature, then via lots of smaller features.
- Now: Deliver more cores per chip.
- The free lunch is over for today's sequential apps and many concurrent apps (expect some regressions). We need killer apps with lots of latent parallelism.
- A generational advance >OO is necessary to get above the "threads+locks" programming model.

From Herb Sutter <hsutter@microsoft.com>

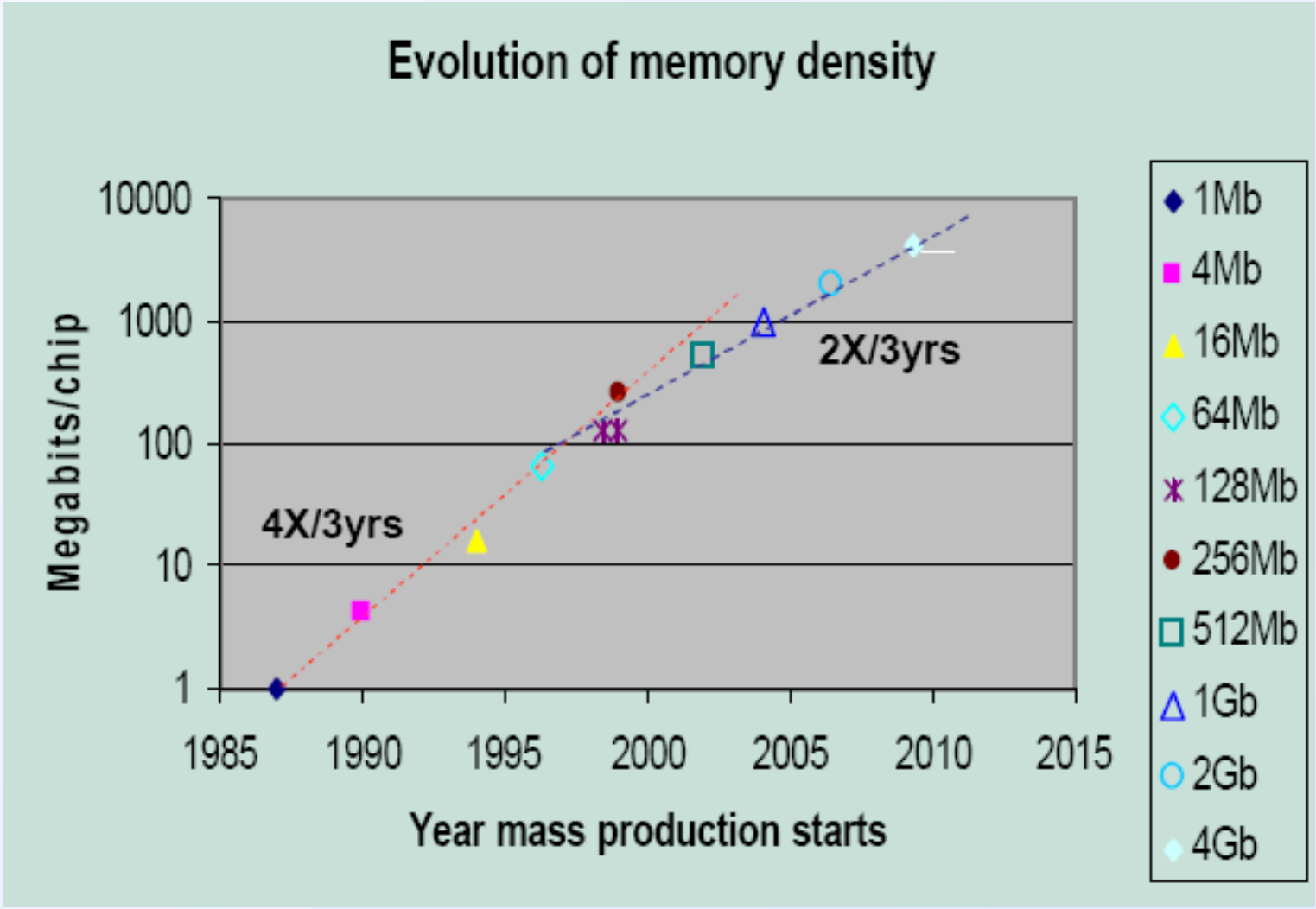
How many cores are you coding for?



Microprocessor parallelism will increase exponentially (2x/1.5yr) in the next decade

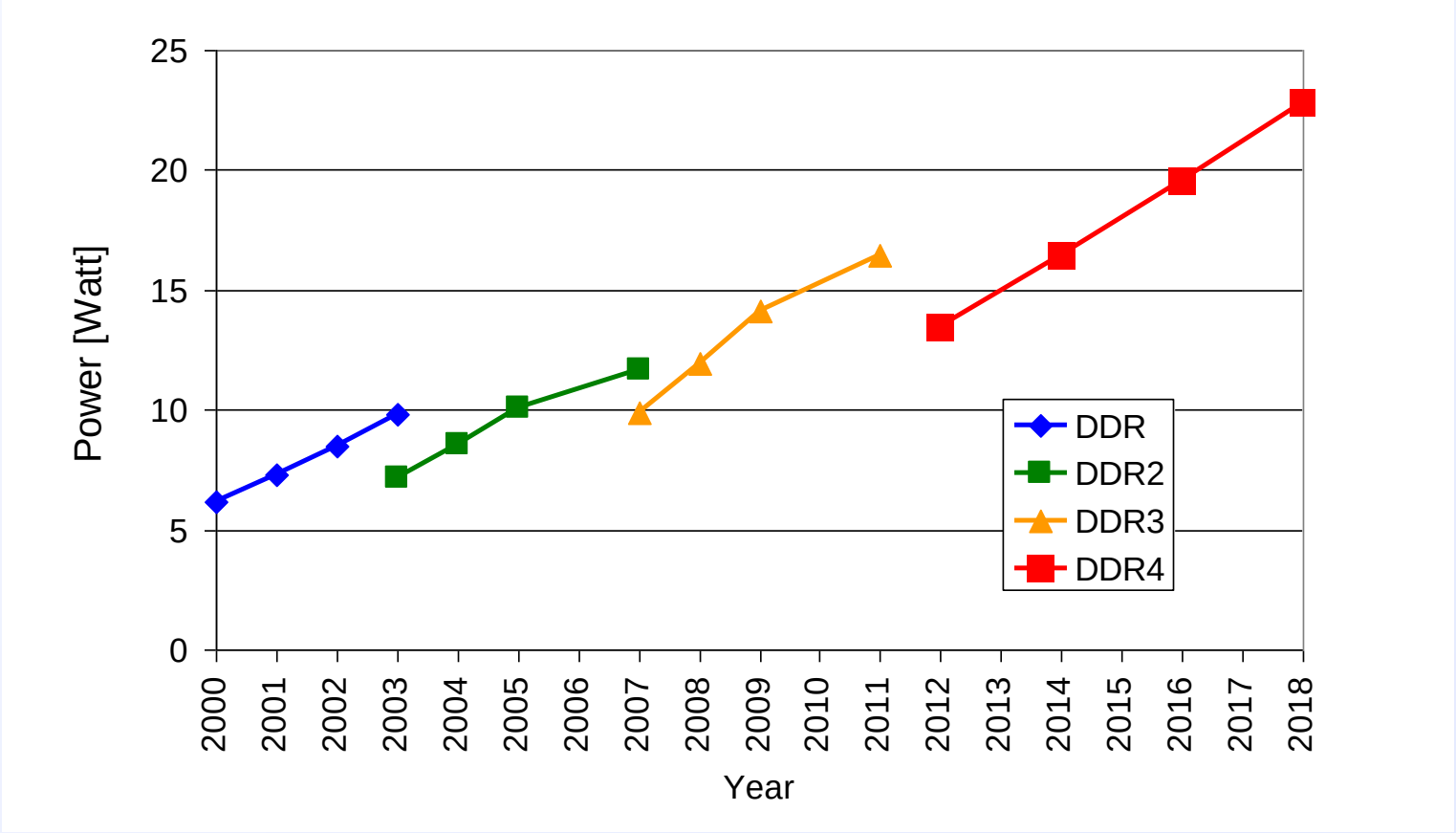
From Herb Sutter <hsutter@microsoft.com>

DRAM component density is doubling every 3 years



Source: IBM

Memory power projection (Quad-rank DIMM)

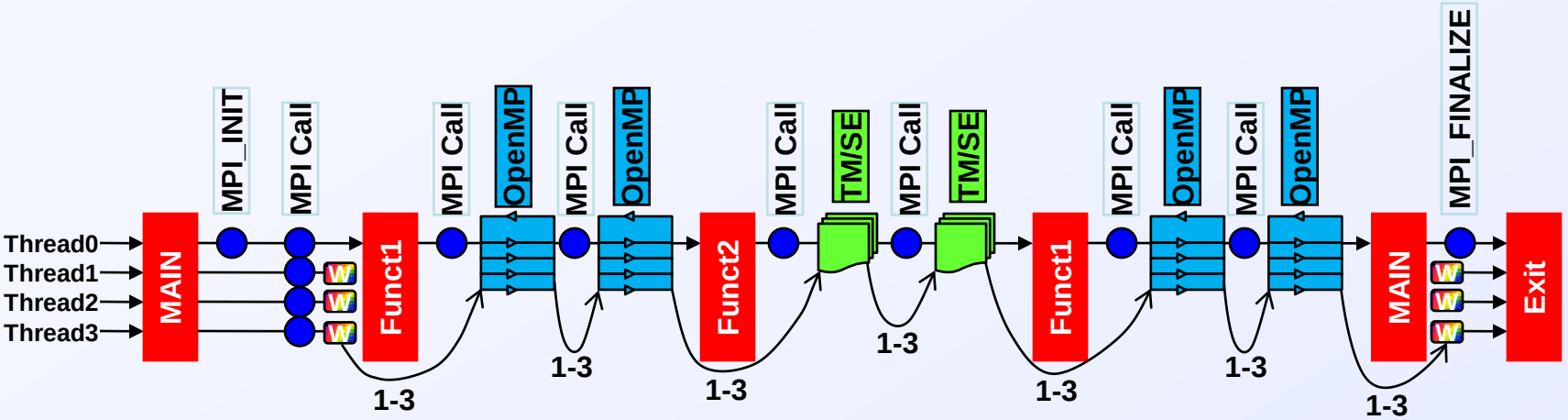


Source: IBM

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

Unified Nested Node Concurrency



1) Pthreads born with MAIN
2) Only Thread0 calls functions to nest parallelism
3) Pthreads based MAIN calls OpenMP based Funct1
4) OpenMP Funct1 calls TM/SE based Funct2
5) Funct2 returns to OpenMP based Funct1
6) Funct1 returns to Pthreads based MAIN

- MPI Tasks on a node are processes (one shown) with multiple OS threads (Thread0-3 shown)
- Thread0 is “Main thread” Thread1-3 are helper threads that morph from Pthread to OpenMP worker to TM/SE compiler generated threads via runtime support
- Hardware support to significantly reduce overheads for thread repurposing and OpenMP loops and locks

Summary

- The petascale (in aggregate) era is here!
- LLNL has a roadmap through Exascale
- Next generation Sequoia platform will enable stockpile stewardship program with a 20 petaFLOP/s simulation environment