Exploiting Asynchrony for Exascale Computational Materials Science

Timothy C. Germann
Physics and Chemistry of Materials (T-1)
Los Alamos National Laboratory
Los Alamos, NM 87545
tcg@lanl.gov

Salishan Conference on High-Speed Computing

April 25-28, 2011

Abstract

The increasingly heterogeneous and hierarchical nature of computer architectures demands that algorithms, programming models, and tools must mirror these characteristics if they are to thrive in this environment. In particular, the current generation of SPMD codes are unlikely to survive the transition to exascale without a fundamental redesign that avoids traditional bulk synchronous parallelism. We are developing a UQ-driven *adaptive physics refinement* scale-bridging strategy for modeling materials at extreme mechanical and irradiation conditions, in which coarse-scale simulations spawn sub-scale direct numerical simulations as needed. This task-based MPMD approach leverages the extensive concurrency and heterogeneity expected at exascale, while enabling novel data models, power management, and fault tolerance strategies within applications. The programming models and runtime task/ resource management and data sharing systems required to support such an approach would also enable *in situ* visualization and analysis, thus alleviating much of the I/O burden.

Outline

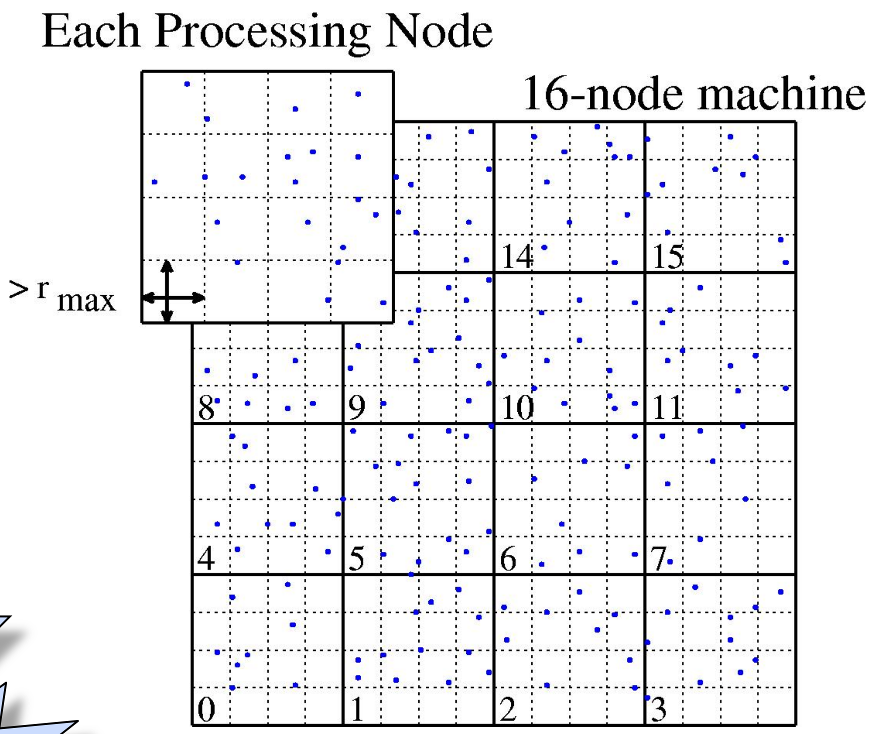
- Why can't we keep doing things the way we've always done?
  - *Case study: molecular dynamics in the massively parallel era: from the Thinking Machines CM-5 and Cray T3D to IBM BlueGene/L*
  - *The (heterogeneous) revolution is now: IBM Roadrunner*
- Exascale Co-design Center for Materials in Extreme Environments
  - *Adaptive physics refinement – sounds good, but what the hell is it?*
    - » Is a task-based materials modeling approach really viable?
    - » Adaptive sampling demonstrations
  - *Prospects for co-design, and why exascale is:*
    - a) Scary
    - b) Difficult
    - c) Exciting
    - d) All of the above

Scalable Parallel Short-range Molecular dynamics (SPaSM) is a high-performance code for studying collective effects in large systems of interacting “particles” (typically atoms)

- Finite-range (r_{max}) interactions ⇒ *O(N)* computational scaling
- Spatial decomposition on shared and distributed memory architectures
- 1993 IEEE Gordon Bell Performance Prize (50 GFlop/s on CM-5)
- 1998 IEEE Gordon Bell Price / Performance Prize (10 GFlop/s on Linu Alpha Beowulf cluster, $15/MFlop)
- 2005 IEEE Gordon Bell Prize Finalist (48 TFlop/s on BlueGene/L)
- 2008 IEEE Gordon Bell Prize Fin (369 TFlop/s on Roadrunner)

Evolution

Revolution



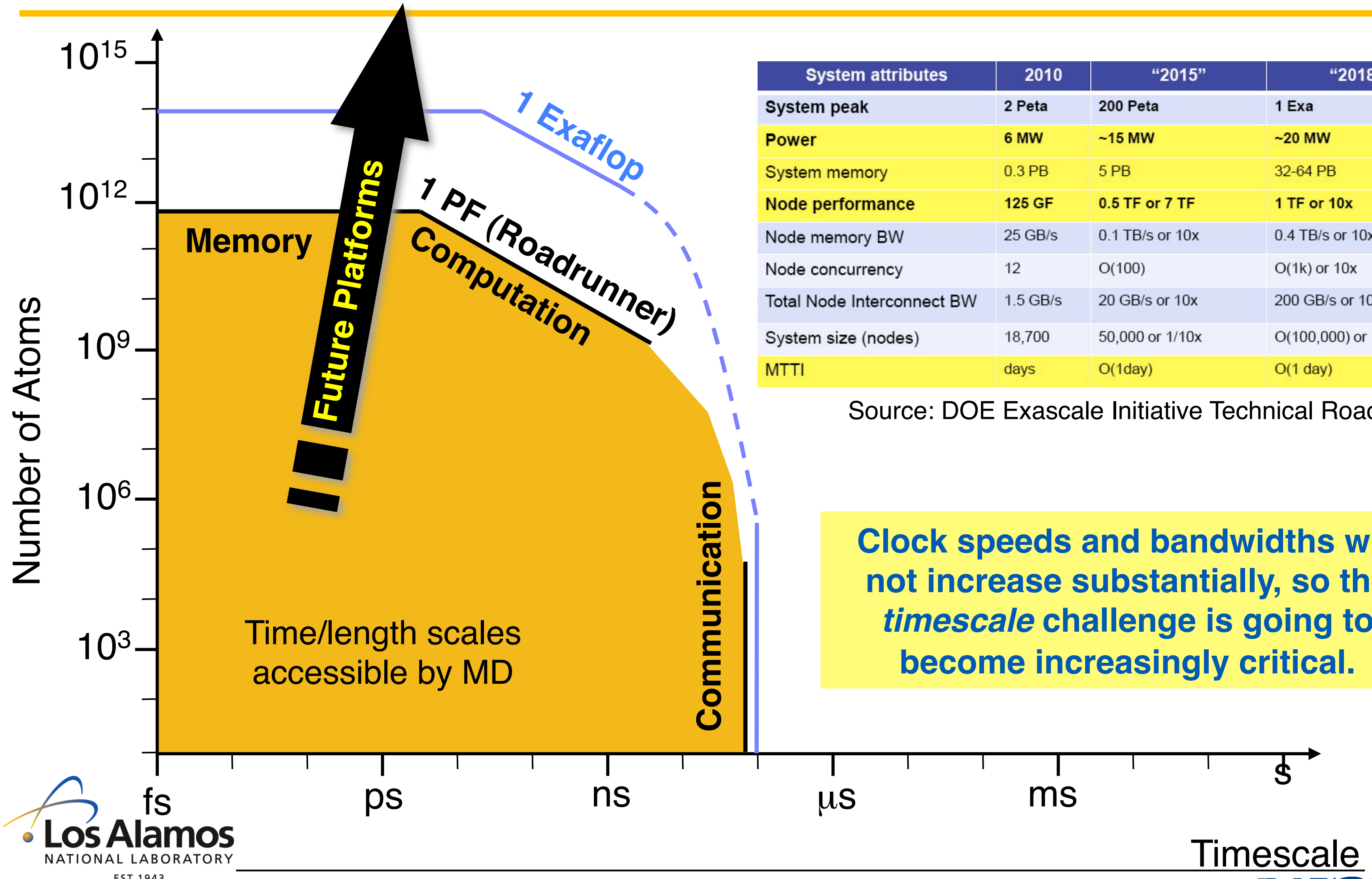
(David Beazley, Peter Lomdahl)

- Object-oriented scripting language with parallel *in situ* visualization and analysis libraries (runtime “steering”)

A wide range of applications have been studied with SPaSM: 1993-2010 covers

COMPUTERS IN PHYSICS
JUL/AUG 1993
Parallel Programmi
A/D Convert
Multimedia Techniques and Technol
AMERICAN INSTITUTE OF PHYSICS
Los Alamos Science
LOS ALAMOS NATIONAL LABORATORY
22 1994
High-Performance Computing
IEEE COMPUTATIONAL SCIENCE & ENGINEERING
Summer 1995
Molecular Reconnaissance
Also: National Software Exchange
IEEE COMPUTER SOCIETY
MRS BULLETIN
Serving the International Materials Research Community
A Publication of the Materials Research Society
February 1996, Volume 21, No. 2
Interatomic Potentials for Atomistic Simulations
IEEE COMPUTATIONAL SCIENCE & ENGINEERING
Winter 1996
Advancing Interactive Visualization
Multigrid Methods p. 55
The SGI Origin Series p. 93
Annual Index p. 80
COMPUTER SOCIETY
PHILOSOPHICAL TRANSACTIONS OF THE ROYAL SOCIETY A
MATHEMATICAL, PHYSICAL & ENGINEERING SCIENCES
ISSN 1364-503X
volume 368
number 1916
pages 1537–1828
Turbulent mixing and beyond
Papers of a Theme Issue compiled and edited by Snezhana I. Abarzhi and Katepalli R. Sreenivasan
ld's longest running science journal
Royal Society Publishing
13 April 2010
NA-ASC-105R-08-Vol. 1-Rev.0
ASC ROADMAP
National Nuclear Security through Leadership in Weapons Science
The Computer Magazine for Scientists and Engineers
COMPUTERS IN PHYSICS
MAY/JUN 1997
10th Anniversary 1987 1997
Controlling Data Glut
Anniversary Book Reviews
AMERICAN INSTITUTE OF PHYSICS
Inside: Energy Quarterly
MRS Bulletin
December 2010 Vol. 35 No. 12
www.mrs.org/bulletin
Structural metals at extremes
ALSO IN THIS ISSUE
Materials for organic and hybrid inorganic/organic electronics
Scientific Grand Challenges for National Security:
THE ROLE OF COMPUTING AT THE EXTREME SCALE
October 6-8, 2009 • Washington D.C.
U.S. DEPARTMENT OF ENERGY
Sponsored by the Office of Advanced Scientific Computing Research and the National Nuclear Security Administration
April 11, 2006 | vol. 103 | no. 15 | pp. 5633–6074
PNAS
Proceedings of the National Academy of Sciences of the United States of America
www.pnas.org
Intervention simulations for U.S. influenza pandemic
Short-circuited atmosphere convections
Warming and African malaria resurgence
Mouse pancreatic cancer genetics
Poxvirus entry via membrane shedding

Current trends will increase the *length*, but not *time*, scales accessible by molecular dynamics simulation



System attributes	2010	“2015”	“2018”
System peak	**2 Peta**	**200 Peta**	**1 Exa**
Power	**6 MW**	**~15 MW**	**~20 MW**
System memory	0.3 PB	5 PB	32-64 PB
Node performance	**125 GF**	**0.5 TF or 7 TF**	**1 TF or 10x**
Node memory BW	25 GB/s	0.1 TB/s or 10x	0.4 TB/s or 10x
Node concurrency	12	O(100)	O(1k) or 10x
Total Node Interconnect BW	1.5 GB/s	20 GB/s or 10x	200 GB/s or 10x
System size (nodes)	18,700	50,000 or 1/10x	O(100,000) or 1/10 x
MTTI	days	O(1day)	O(1 day)

Source: DOE Exascale Initiative Technical Roadmap

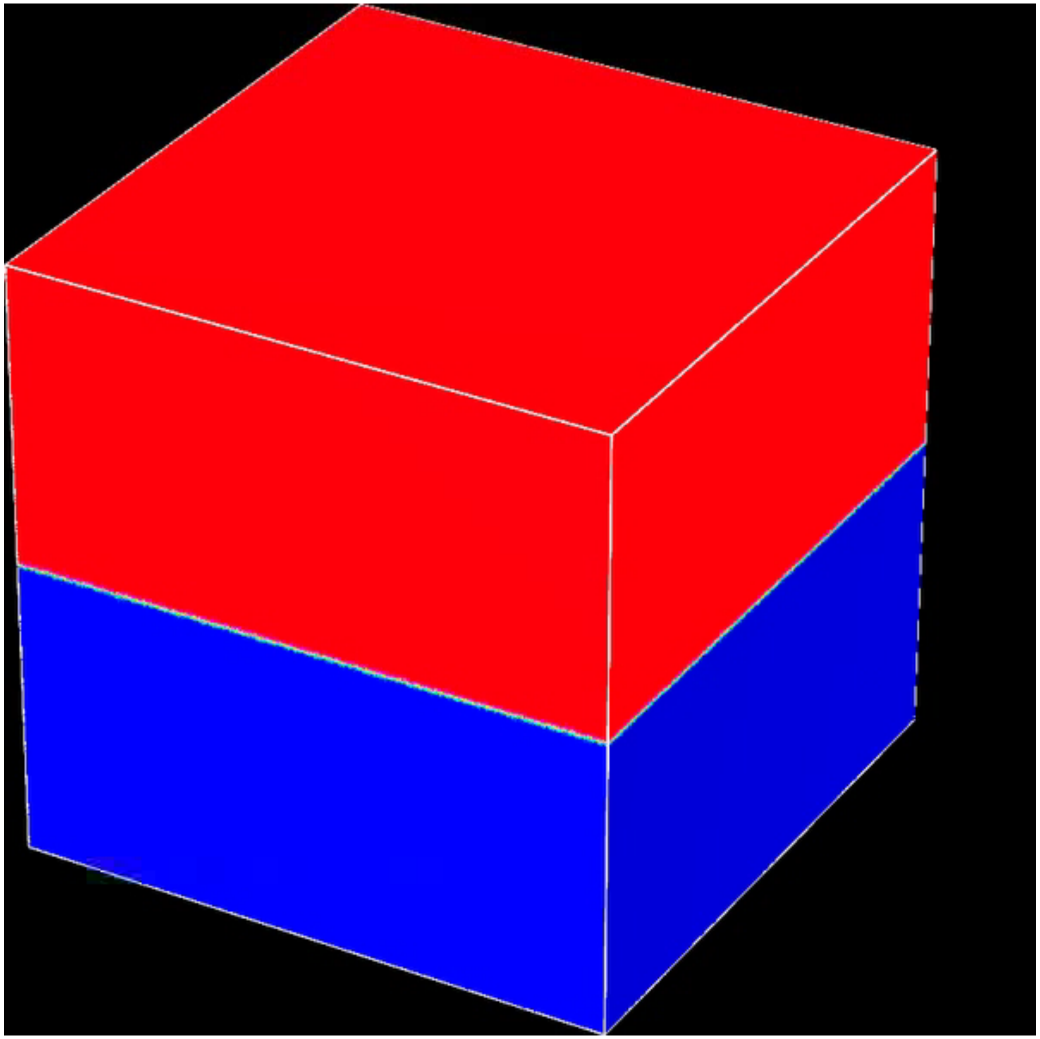
Clock speeds and bandwidths will not increase substantially, so the *timescale* challenge is going to become increasingly critical.

As system sizes increase, atomistic resolution is necessary in a diminishing volume fraction

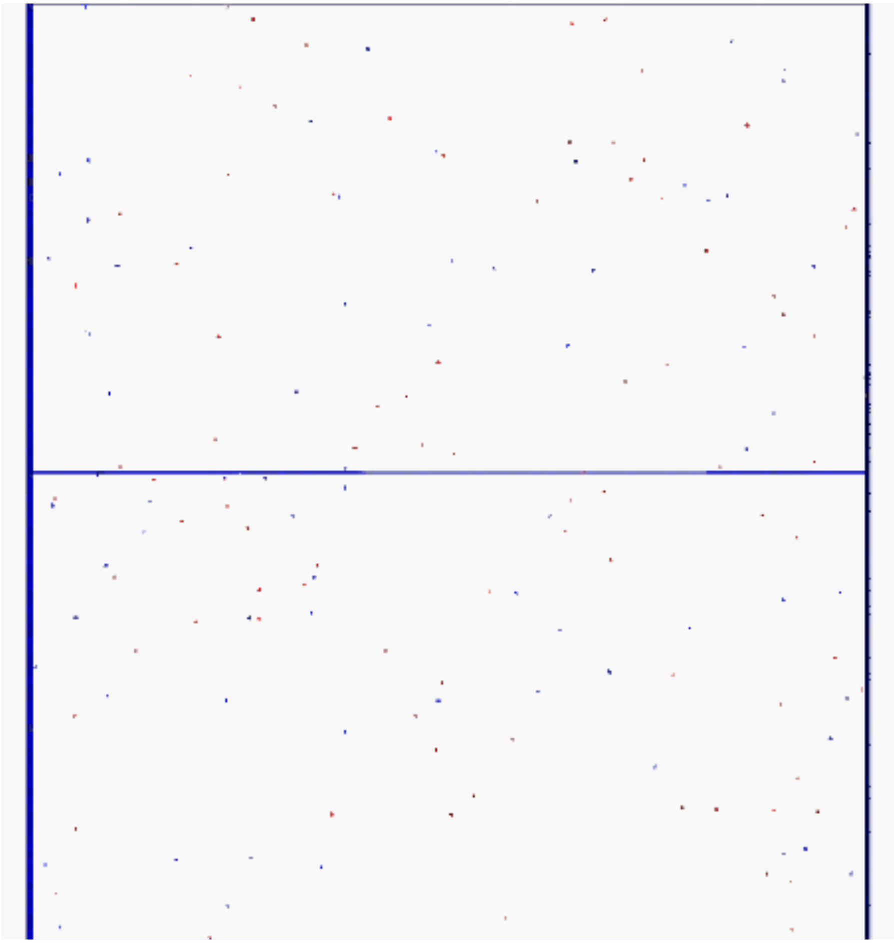
# of atoms (computational effort) scales with volume, while features of interest often scale with area

Examples include:

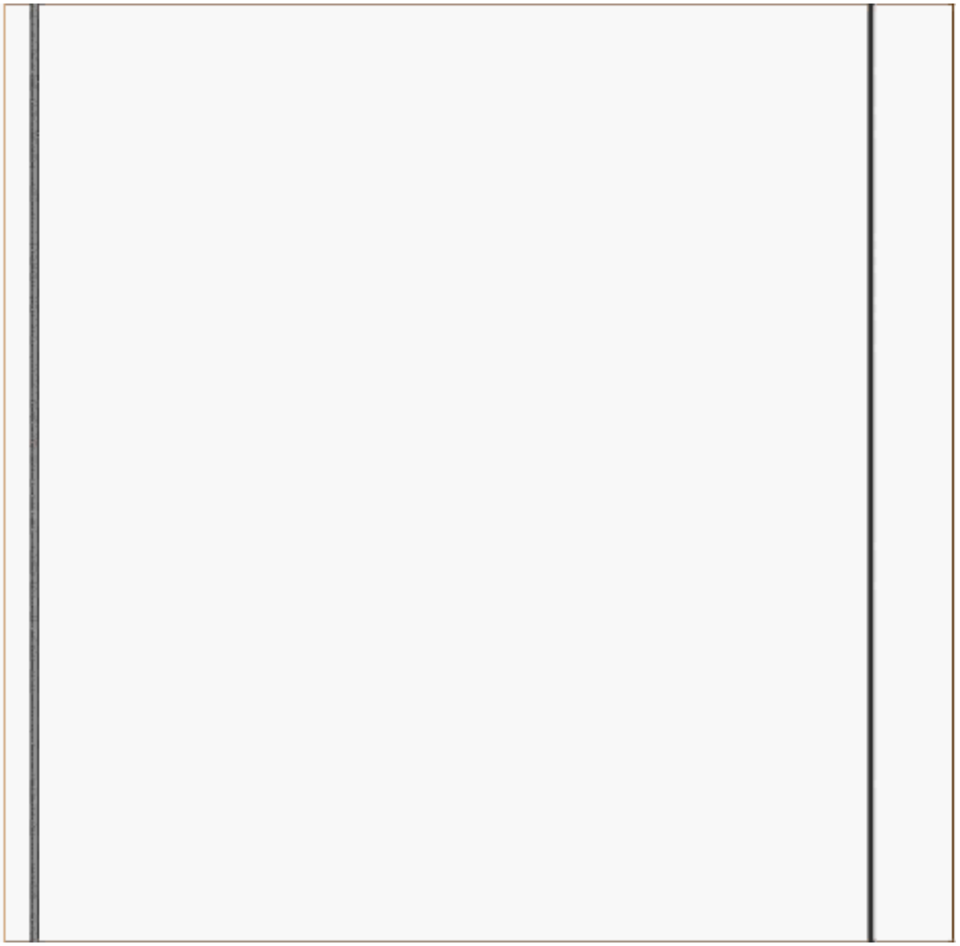
- Interface instabilities
  - Example: 7.4 billion atom Rayleigh-Taylor (Jan `06 BlueGene/L)
- Shock fronts
- Phase boundaries
  - Product phase nucleation and growth within a parent phase



Void nucleation and growth leading to ductile spall failure in a shocked copper bicrystal



Lattice defects (dislocations)



Surface atoms (voids)

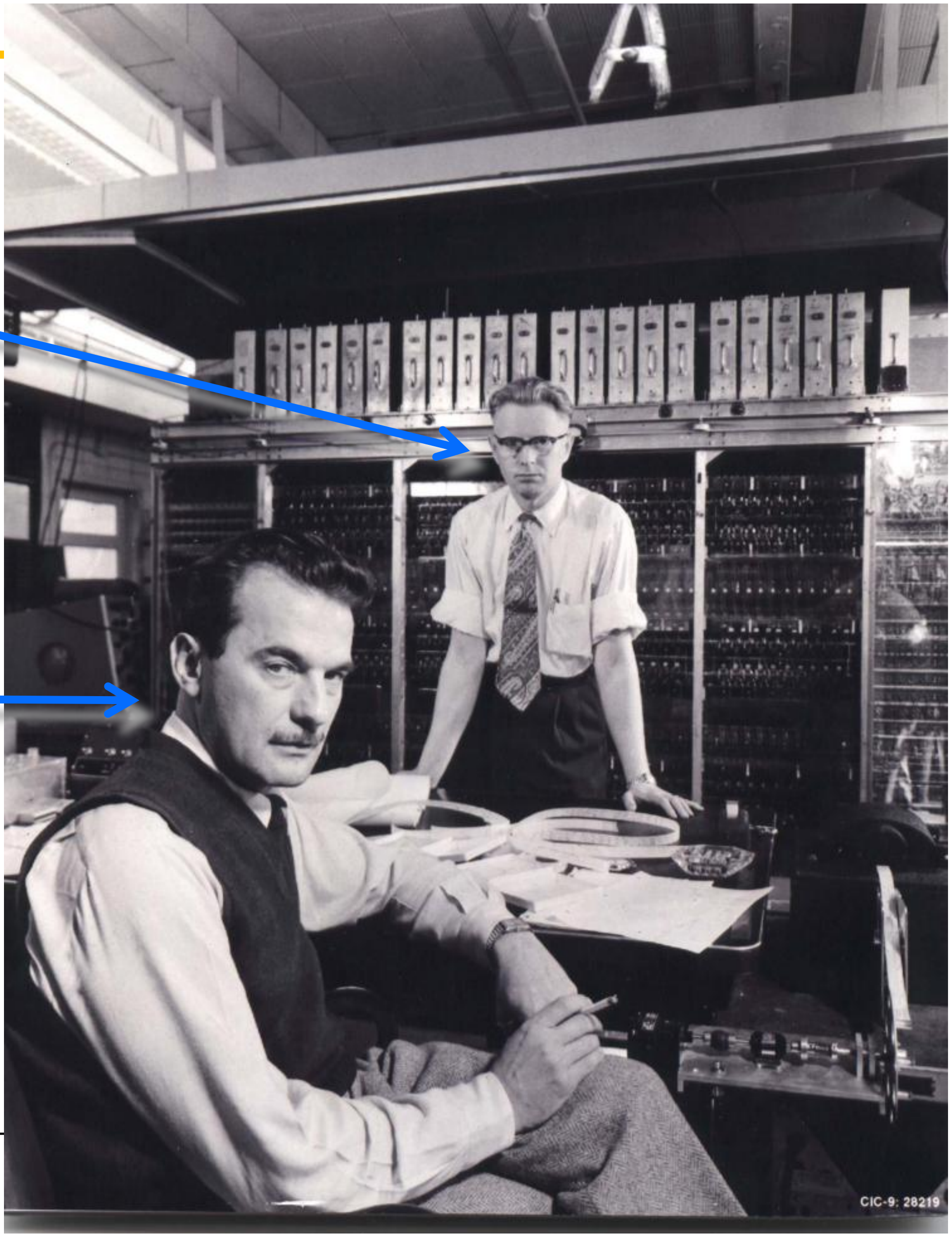
Preparing for exascale: issues to confront

- Computer architectures are becoming increasingly **heterogeneous** and **hierarchical**, with greatly increased flop/byte ratios.
- The algorithms, programming models, and tools that will thrive in this environment must mirror these characteristics.
- SPMD bulk synchronous (10^9-way) parallelism will no longer be viable.
- Power, energy, and heat dissipation are increasingly important.
- Traditional global checkpoint/restart is becoming impractical.
  - *Local flash memory?*
- Fault tolerance and resilience
  - *Recovering from soft and hard errors, and anticipating faults*
  - *MPI/application ability to drop or replace nodes*
  - *The curse of silent errors*
- Analysis and visualization
  - *In situ, e.g. "active storage" using I/O nodes?*

Computational co-design at Los Alamos, circa 1950

Hardware architect
(Richardson)

Application scientist
(Metropolis)



MANIAC I

Exascale Co-design Center for Materials in Extreme Environments

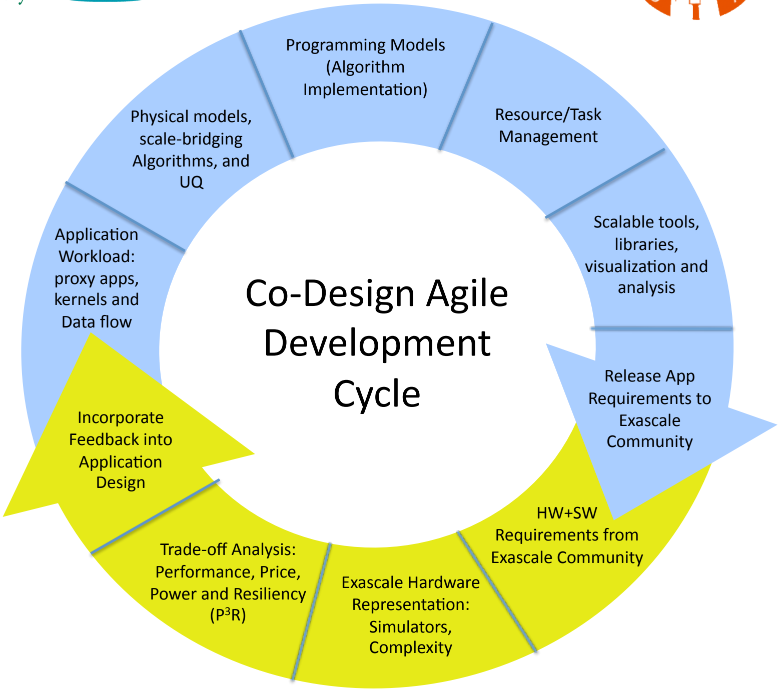
Center Director: Tim Germann (LANL) **Deputy Director: Jim Belak (LLNL)**

Los Alamos National Laboratory EST. 1943 · Lawrence Livermore National Laboratory · Oak Ridge National Laboratory · Sandia National Laboratories · Stanford University · California Institute of Technology 1891

Exascale computing will transform computational materials science by enabling the pervasive embedding of microscopic behavior into meso- and macroscale materials simulation. To achieve this, our Center has a focused effort in 4 areas:

- Scale-bridging algorithms
- Proxy applications
- Hierarchical programming models
- Holistic analysis and optimization

A tightly coupled co-design loop will optimize algorithms and architectures for performance, memory and data movement, power, and resiliency.



Co-Design Agile Development Cycle

Programming Models (Algorithm Implementation)

Resource/Task Management

Scalable tools, libraries, visualization and analysis

Release App Requirements to Exascale Community

HW+SW Requirements from Exascale Community

Exascale Hardware Representation: Simulators, Complexity

Trade-off Analysis: Performance, Price, Power and Resiliency (P^{3}R)

Incorporate Feedback into Application Design

Application Workload: proxy apps, kernels and Data flow

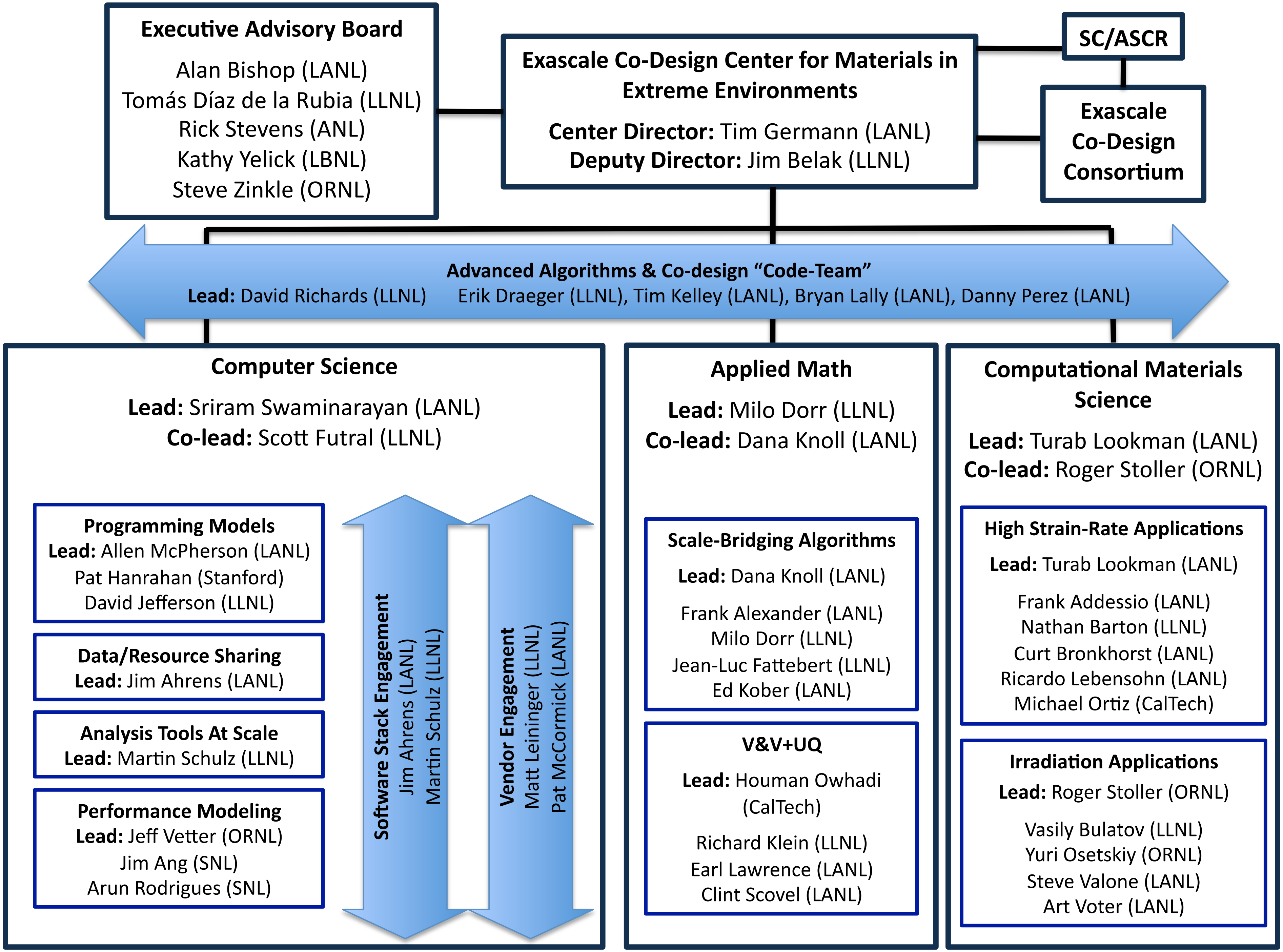
Physical models, scale-bridging Algorithms, and UQ

Our objective is to establish the interrelationship between algorithms, system software, and hardware required to develop a multiphysics exascale simulation framework for modeling materials subjected to extreme mechanical and radiation environments.

Mission drivers

A predictive understanding of the response of materials to extreme conditions (mechanical and/or irradiation) underpins many DOE programs, including:

- *NNSA Advanced Simulation and Computing*
- *NNSA Enhanced Surveillance Campaign*
- *NIF – National Ignition Facility (LLNL)*
- *MaRIE – Matter-Radiation Interactions in Extremes (LANL)*
- *Nuclear Energy Hub*
  - » CASL – Consortium for Advanced Simulation of Light Water Reactors (ORNL)
- *BES Energy Frontier Research Centers*
  - » CMIME – Center for Materials at Irradiation and Mechanical Extremes (LANL)
  - » CPD – Center for Defect Physics (ORNL)
  - » CMSNF – Center for Materials Science of Nuclear Fuel (INL)



Executive Advisory Board

Alan Bishop (LANL)
Tomás Díaz de la Rubia (LLNL)
Rick Stevens (ANL)
Kathy Yelick (LBNL)
Steve Zinkle (ORNL)

Exascale Co-Design Center for Materials in Extreme Environments

Center Director: Tim Germann (LANL)
Deputy Director: Jim Belak (LLNL)

SC/ASCR

Exascale Co-Design Consortium

Advanced Algorithms & Co-design "Code-Team"
Lead: David Richards (LLNL) Erik Draeger (LLNL), Tim Kelley (LANL), Bryan Lally (LANL), Danny Perez (LANL)

Computer Science

Lead: Sriram Swaminarayan (LANL)
Co-lead: Scott Futral (LLNL)

Programming Models
Lead: Allen McPherson (LANL)
Pat Hanrahan (Stanford)
David Jefferson (LLNL)

Data/Resource Sharing
Lead: Jim Ahrens (LANL)

Analysis Tools At Scale
Lead: Martin Schulz (LLNL)

Performance Modeling
Lead: Jeff Vetter (ORNL)
Jim Ang (SNL)
Arun Rodrigues (SNL)

Software Stack Engagement
Jim Ahrens (LANL)
Martin Schulz (LLNL)

Vendor Engagement
Matt Leininger (LLNL)
Pat McCormick (LANL)

Applied Math

Lead: Milo Dorr (LLNL)
Co-lead: Dana Knoll (LANL)

Scale-Bridging Algorithms

Lead: Dana Knoll (LANL)

Frank Alexander (LANL)
Milo Dorr (LLNL)
Jean-Luc Fattebert (LLNL)
Ed Kober (LANL)

V&V+UQ

Lead: Houman Owhadi (CalTech)

Richard Klein (LLNL)
Earl Lawrence (LANL)
Clint Scovel (LANL)

Computational Materials Science

Lead: Turab Lookman (LANL)
Co-lead: Roger Stoller (ORNL)

High Strain-Rate Applications

Lead: Turab Lookman (LANL)

Frank Addessio (LANL)
Nathan Barton (LLNL)
Curt Bronkhorst (LANL)
Ricardo Lebensohn (LANL)
Michael Ortiz (CalTech)

Irradiation Applications

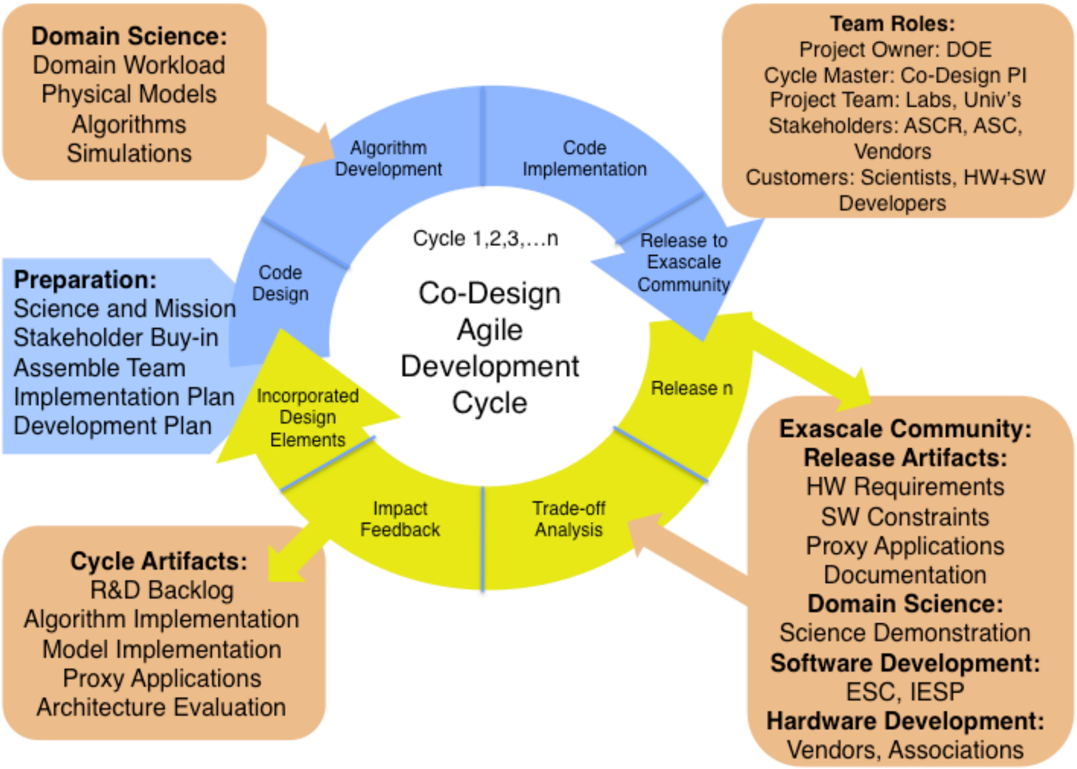
Lead: Roger Stoller (ORNL)

Vasily Bulatov (LLNL)
Yuri Osetskiy (ORNL)
Steve Valone (LANL)
Art Voter (LANL)
)

Co-design as an Agile Development Cycle

- Hardware requirements and software constraints are continuously released to the exascale community.
- The performance, power, price, and resiliency (P^{3}R) trade-off is analyzed to create the optimal design for hardware, software and applications needed for the exascale simulation environment.

  All elements of the agile development cycle operate concurrently.



Domain Science:
Domain Workload
Physical Models
Algorithms
Simulations

Team Roles:
Project Owner: DOE
Cycle Master: Co-Design PI
Project Team: Labs, Univ's
Stakeholders: ASCR, ASC, Vendors
Customers: Scientists, HW+SW Developers

Preparation:
Science and Mission
Stakeholder Buy-in
Assemble Team
Implementation Plan
Development Plan

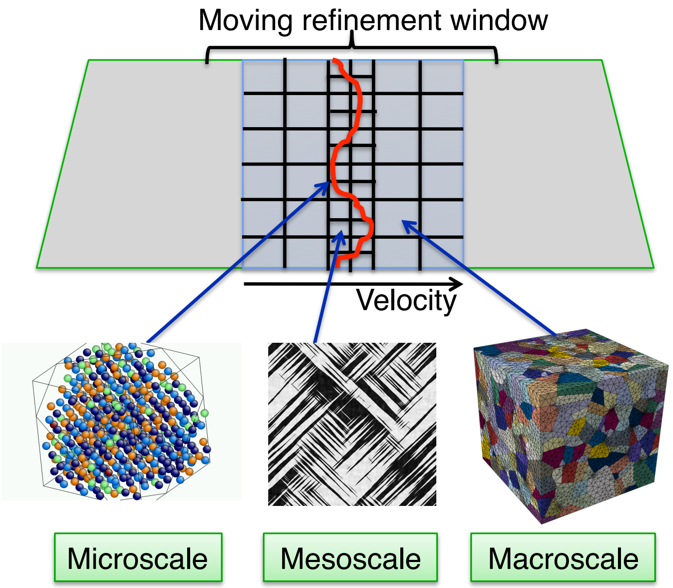
Cycle Artifacts:
R&D Backlog
Algorithm Implementation
Model Implementation
Proxy Applications
Architecture Evaluation

Exascale Community:
Release Artifacts:
HW Requirements
SW Constraints
Proxy Applications
Documentation
Domain Science:
Science Demonstration
Software Development:
ESC, IESP
Hardware Development:
Vendors, Associations

Embedded Scale-Bridging Algorithms

- Our goal is to introduce more detailed physics into computational materials science applications in a way which escapes the traditional synchronous SPMD paradigm and exploits the heterogeneity expected in exascale hardware.

- To achieve this, we are developing a UQ-driven *adaptive physics refinement* approach.
- Coarse-scale simulations dynamically spawn tightly coupled and self-consistent fine-scale simulations as needed.
- This *task-based* approach naturally maps to exascale heterogeneity, concurrency, and resiliency issues.

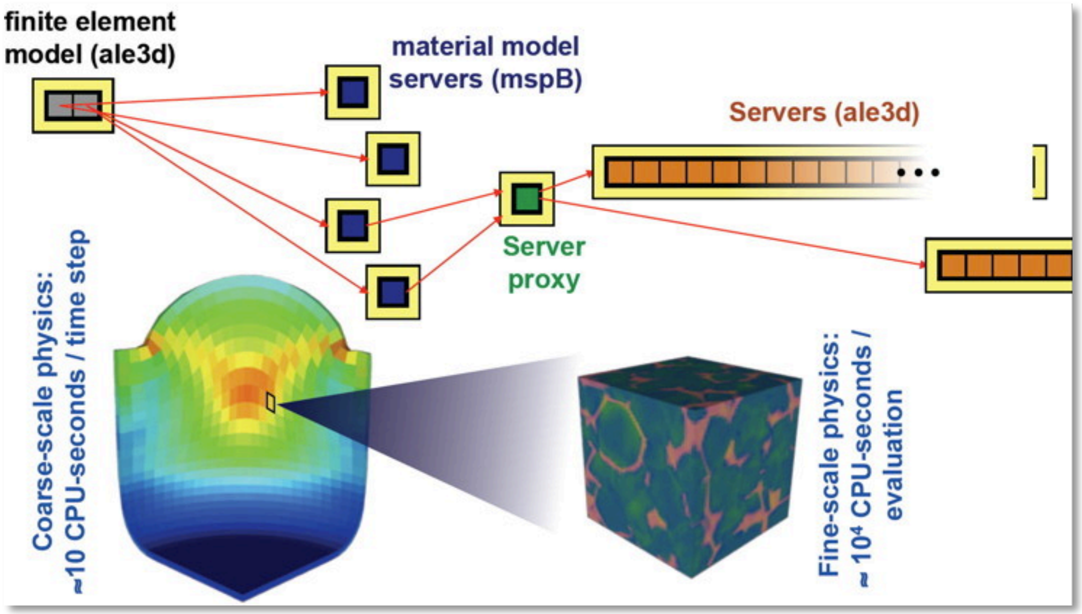


Embedded Scale-Bridging Algorithms

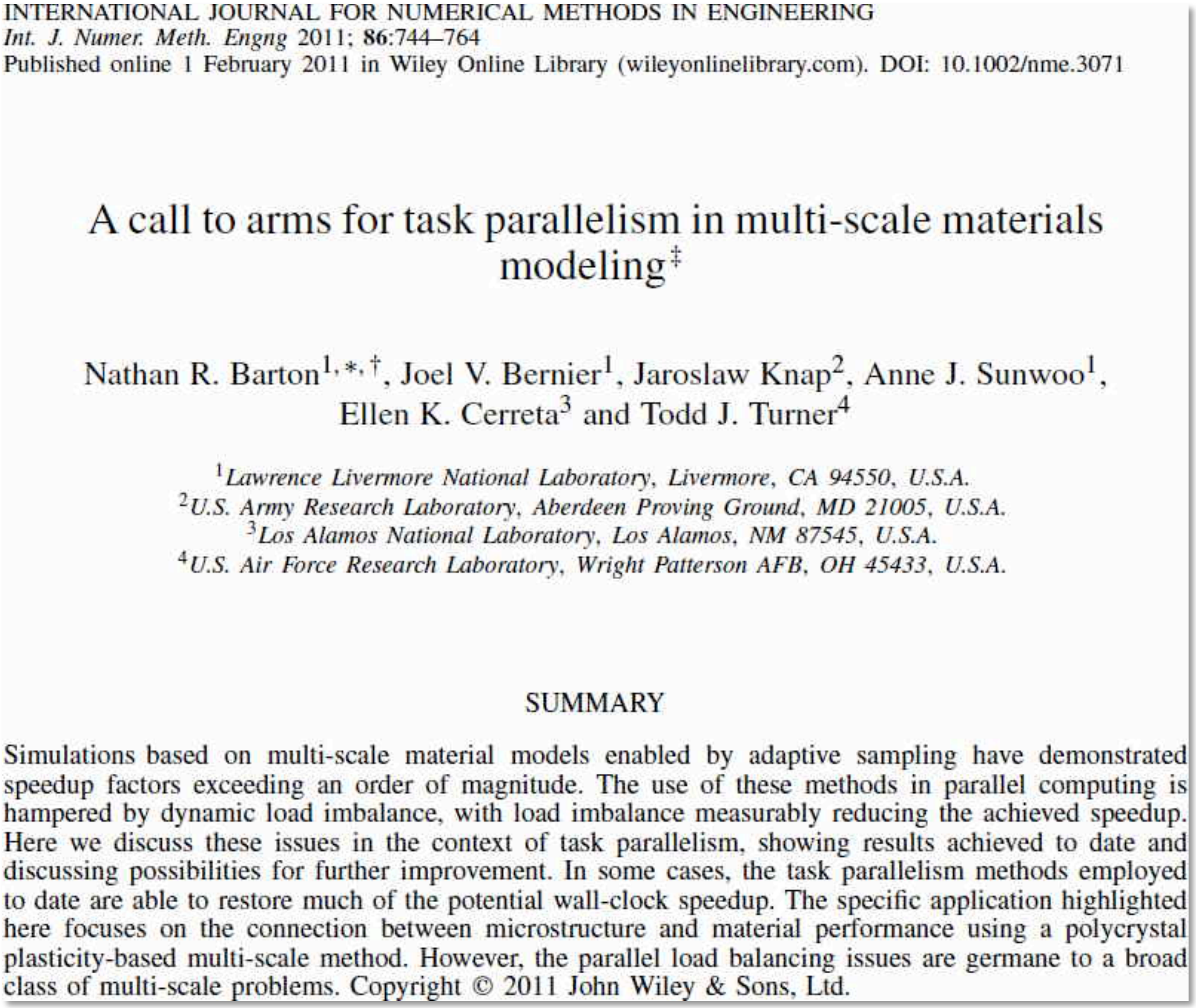
- Scale-bridging algorithms require a consistent two-way algorithmic coupling between temporally evolving distinct spatial levels; they are not "modeling", and not one-way information flow.
- Our focus is on coupling between macro (coarse-scale model) and meso (fine-scale model) scales with all unit physics being deterministic.
- We begin by building off of our adaptive sampling success, but move to the use of temporally evolving mesoscale and spatial adaption.
- Similar concepts apply in the time domain, e.g. using *ab initio* techniques to compute activation energies for a rate theory or kinetic Monte Carlo model ("on-the-fly kMC") applied to radiation damage modeling.

Adaptive sampling techniques have been successfully demonstrated by LLNL

- A coarse-scale model (e.g. FEM) calls a lower length-scale model (e.g. polycrystal plasticity) and stores the response obtained for a given microstructure, each time this model is interrogated
- A microstructure-response database is thus populated
- The fine-scale workload varies dramatically over the coarse-scale spatial and temporal domain
- Dynamic workload balancing in a task parallel context


finite element model (ale3d)
material model servers (mspB)
Servers (ale3d)
Server proxy
Coarse-scale physics: ≈10 CPU-seconds / time step
Fine-scale physics: ≈ 10^4 CPU-seconds / evaluation

N. R. Barton, J. Knap, A. Arsenlis, R. Becker, R. D. Hornung, and D. R. Jefferson. Embedded polycrystal plasticity and adaptive sampling. *Int. J. Plast*. **24**, 242-266 (2008)



INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING
Int. J. Numer. Meth. Engng 2011; **86**:744–764
Published online 1 February 2011 in Wiley Online Library (wileyonlinelibrary.com). DOI: 10.1002/nme.3071

A call to arms for task parallelism in multi-scale materials modeling[‡]

Nathan R. Barton[1,*,†], Joel V. Bernier[1], Jaroslaw Knap[2], Anne J. Sunwoo[1],
Ellen K. Cerreta[3] and Todd J. Turner[4]

[1] *Lawrence Livermore National Laboratory, Livermore, CA 94550, U.S.A.*
[2] *U.S. Army Research Laboratory, Aberdeen Proving Ground, MD 21005, U.S.A.*
[3] *Los Alamos National Laboratory, Los Alamos, NM 87545, U.S.A.*
[4] *U.S. Air Force Research Laboratory, Wright Patterson AFB, OH 45433, U.S.A.*

SUMMARY

Simulations based on multi-scale material models enabled by adaptive sampling have demonstrated speedup factors exceeding an order of magnitude. The use of these methods in parallel computing is hampered by dynamic load imbalance, with load imbalance measurably reducing the achieved speedup. Here we discuss these issues in the context of task parallelism, showing results achieved to date and discussing possibilities for further improvement. In some cases, the task parallelism methods employed to date are able to restore much of the potential wall-clock speedup. The specific application highlighted here focuses on the connection between microstructure and material performance using a polycrystal plasticity-based multi-scale method. However, the parallel load balancing issues are germane to a broad class of multi-scale problems. Copyright © 2011 John Wiley & Sons, Ltd.

Agile Proxy Application Development

- Petascale single-scale SPMD and scale-bridging MPMD proxy apps will be used to explore algorithm and programming model design space with domain experts, hardware architects and system software developers.
- These proxy applications will not be "toy models", but will realistically encapsulate the workload, data flow and mathematical algorithms of the full applications.

1998 GORDON BELL PRIZE

Perform

Michael S. Warren, Timothy
(Lo

John K. Salm

For Th
Avalon: An Alpha/Linu

IEEE COMPUTER SOCIETY IEEE

SC07

Reno-Sparks Convention Center • November 10 - 16, 2007

Gordon Bell Prize

James N. Glosli, Kyle J. Caspersen,
David F. Richards, Robert E. Rudd,
Frederick H. Streitz,
Lawrence Livermore National Laboratory
John A. Gunnels, IBM Corporation

Extending Stability Beyond CPU-Millennium:
Micron-Scale Atomistic Simulation of Kelvin-Helmholtz Instability

Becky J. Verastegui
Becky Verastegui, Conference Chair

David H Bailey
David H. Bailey, Gordon Bell Prize Chair

SC06
POWERFUL BEYOND IMAGINATION

Gordon Bell Prize WINNER

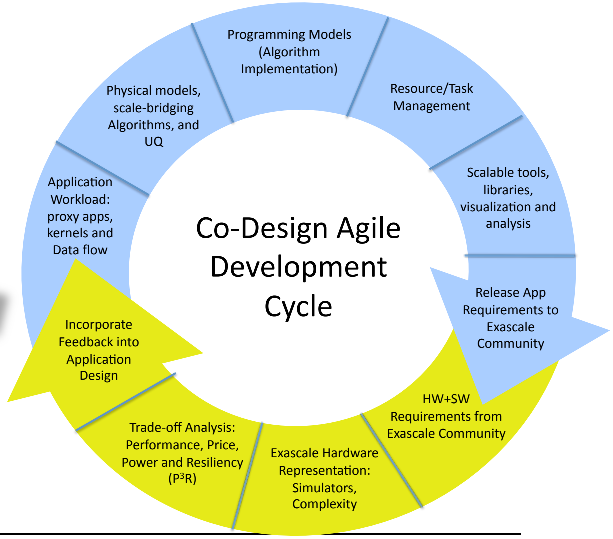
Peak Performance Award

"Large-scale Electronic Structure Calculations
of High-Z Metals on the BlueGene/L Platform"

Francois Gygi, *University of California, Davis;*
Erik W. Draeger, Martin Schulz, Bronis R. de Supinski, *Lawrence Livermore National Laboratory;*
John A. Gunnels, Vernon Austel, James C. Sexton, *IBM;*
Franz Franchetti, *Carnegie Mellon University;* and
Stefan Kral, Christoph W. Ueberhuber, Juergen Lorenz, *Vienna University of Technology*

SC06 GORDON BELL PRIZE CHAIR

Barbara Horner-Miller
SC06 GENERAL CHAIR



Co-Design Agile Development Cycle
Programming Models (Algorithm Implementation)
Resource/Task Management
Scalable tools, libraries, visualization and analysis
Release App Requirements to Exascale Community
HW+SW Requirements from Exascale Community
Exascale Hardware Representation: Simulators, Complexity
Trade-off Analysis: Performance, Price, Power and Resiliency (P³R)
Incorporate Feedback into Application Design
Application Workload: proxy apps, kernels and Data flow
Physical models, scale-bridging Algorithms, and UQ

Agile Proxy Application Development

- Proxy apps for single-scale SPMD applications (e.g. molecular dynamics) will be used to assess node-level issues including:
  - *Data structures*
  - *Hierarchical memory storage and access*
  - *Power management strategies*
  - *Node-level performance*
- The asynchronous task-based MPMD scale-bridging proxy apps will be used to optimize:
  - *System-level data movement*
  - *Resilience (fault management)*
  - *Load balancing techniques*
  - *Performance scalability*
- These proxy apps are **not** static entities, but the central mechanism for our co-design process.

Proxy application suite (single-scale)

- First-Principles Molecular Dynamics (MD): Qbox
  - *Dense linear algebra and spectral transform operations*
  - *2006 Gordon Bell Prize (2005 finalist)*
- Tight-Binding MD: LATTE
- Classical MD (Pair-like potentials): SPaSM
  - *Particle-based, spatial (linked-cell) domain decomposition*
  - *In situ visualization demonstrated to 1 trillion atoms on BlueGene/L*
  - *1993, 1998 Gordon Bell Prizes (2005, 2008 finalist)*
- Classical Molecular Dynamics (Many-body potentials): ddcMD
  - *Particle-based, particle domain decomposition*
  - *Soft error recovery demonstrated to CPU-millenium on BlueGene/L*
  - *2005, 2007 Gordon Bell Prizes (2009 finalist)*
- Phase Field Method: AMPE/GL
- Polycrystal plasticity: VP-FFT

Hierarchical Programming Models

- The challenge for programming models in the context of this project is that they need to expose hardware capabilities to the application programmer while at the same time hiding the continuous flux and complexity of the underlying hardware.
- A hierarchy of programming models exposes and exploits the heterogeneity while providing a transparent layer of abstraction that insulates the application programmer from the flux and complexity of the underlying hardware.
- The programming models and approaches developed to achieve our scale-bridging materials application will be broadly applicable to a variety of multiscale, multiphysics applications:

Astrophysics & the structure of the universe

Climate and weather prediction

Nuclear reactor simulation

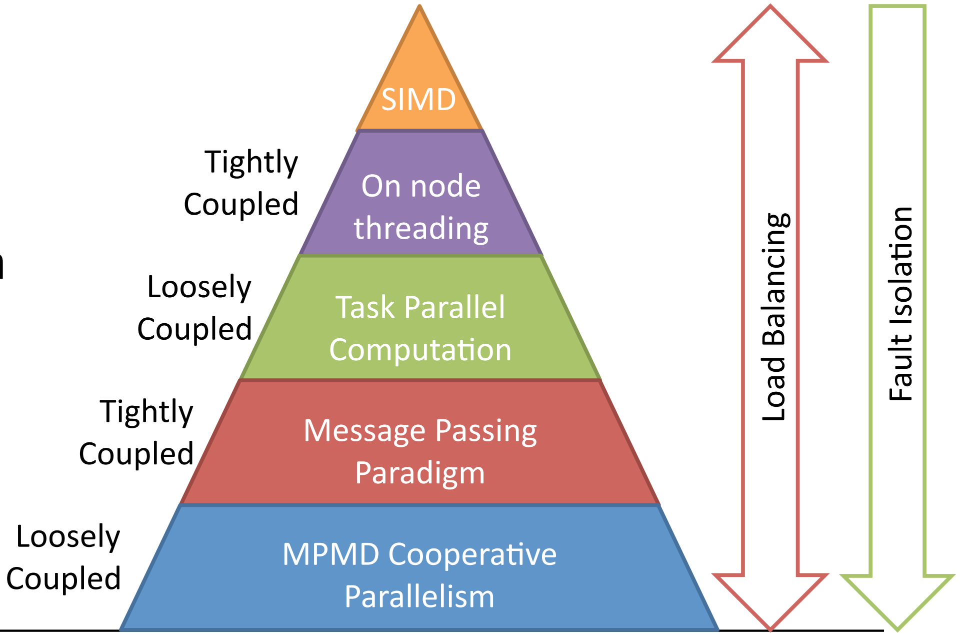
Structural engineering

Plasma physics

Radiation hydrodynamics

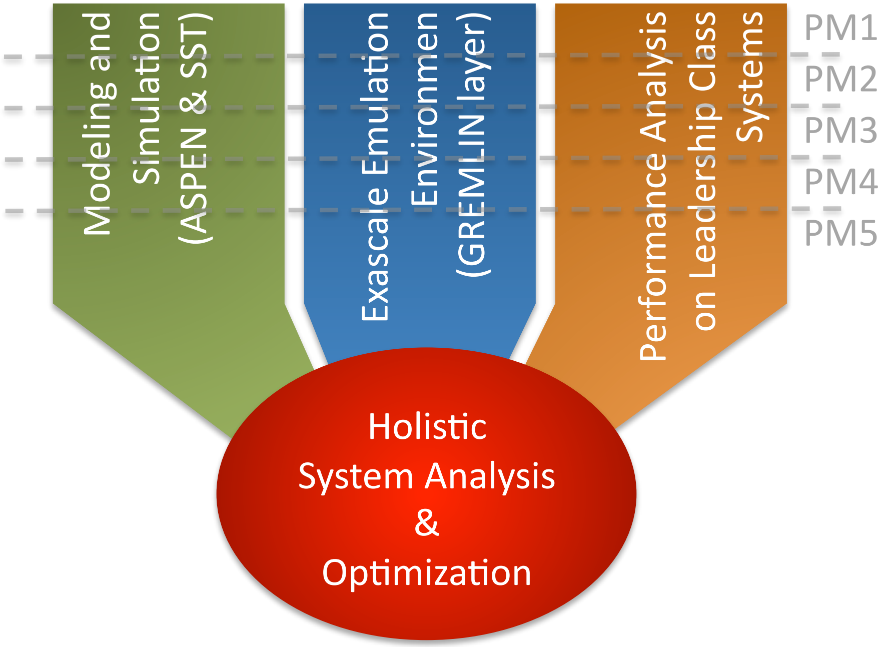
Hierarchical Programming Models

- This hierarchy will replace the traditional bulk synchronous parallel paradigm:
  - *On-node task parallelism* will allow us to couple multiple tightly coupled application components or segments while exploiting on-node resources to their full extent.
  - *Inter-node cooperative parallelism* will provide the necessary capabilities to execute scalable, dynamically structured MPMD applications.
  - *Domain specific languages* aim to encapsulate these levels, enable programmer productivity, and bridge disparate architectures.



Holistic Analysis and Optimization

- A hierarchy of performance models, simulators, and emulators are used to explore algorithm, programming model, and hardware design space before the application is fully constructed.
  - ASPEN: Rapid exploration of design space using application skeletons
  - SST: Detailed simulation of data flow, performance and energy/power cost
  - GREMLIN: Emulation layer to mimic exascale complexity by injecting faults, OS jitter, and other noise to "stress test" the application/SW stack


Modeling and Simulation (ASPEN & SST)
Exascale Emulation Environmen (GREMLIN layer)
Performance Analysis on Leadership Class Systems
PM1
PM2
PM3
PM4
PM5
Holistic System Analysis & Optimization

Summary

- Our objective is to establish the interrelationship between algorithms, system software, and hardware required to develop a multiphysics exascale simulation framework for modeling materials subjected to extreme mechanical and radiation environments.
- This effort is focused in four areas:
  - *Scale-bridging algorithms*
    - UQ-driven adaptive physics refinement
  - *Programming models*
    - Task-based MPMD approaches to leverage concurrency and heterogeneity at exascale while enabling fault tolerance
  - *Proxy applications*
    - Communicate the application workload to the hardware architects and system software developers, and used in performance models/simulators/emulators
  - *Co-design analysis and optimization*
    - Optimization of algorithms and architectures for performance, memory and data movement, power, and resiliency