LA-UR-14-28061
Approved for public release; distribution is unlimited.

Title: Exascale Co-design for Materials in Extreme Environments

Author(s): Germann, Timothy Clark

Intended for: ExMatEx All-Hands Meeting, 2014-09-23/2014-09-25 (Atlanta, Georgia, United States)
Web

Issued: 2014-10-15

Exascale Co-design for Materials in Extreme Environments

ExMatEx

All-Hands Meeting

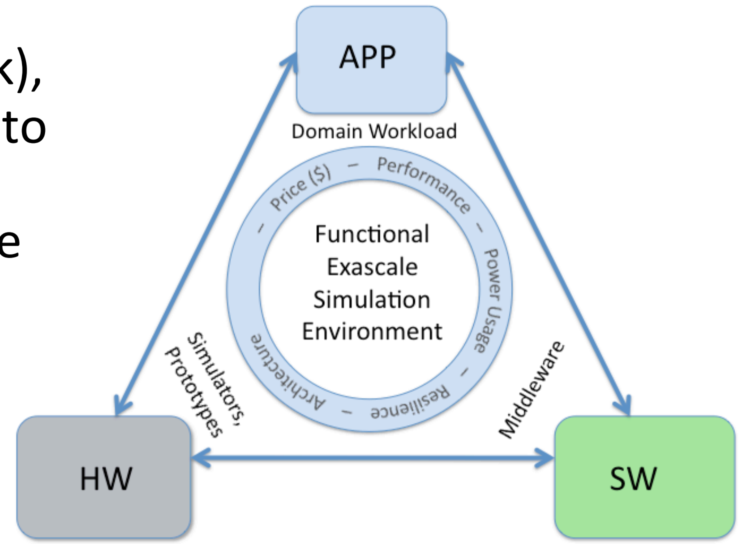
23-25 September 2014

Atlanta, GA

ExMatEx project history

Co-Design Project Goals

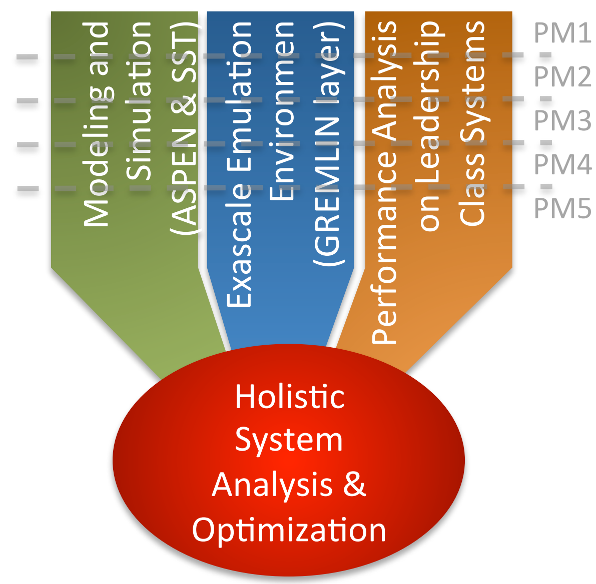
- Our **goal** is to establish the interrelationship between hardware, middleware (software stack), programming models, and algorithms required to enable ***a productive exascale environment*** for multiphysics simulations of materials in extreme mechanical and radiation environments.
- We will exploit, rather than avoid, the greatly increased levels of concurrency, heterogeneity, and flop/byte ratios on the upcoming exascale platforms.



- Our **vision** is an uncertainty quantification (UQ)-driven *adaptive physics refinement* in which meso- and macro-scale materials simulations spawn micro-scale simulations as needed.
  - This *task-based* approach leverages the extensive concurrency and heterogeneity expected at exascale while enabling fault tolerance within applications.
  - The programming models and approaches developed to achieve this will be broadly applicable to a variety of multiscale, multiphysics applications, including astrophysics, climate and weather prediction, structural engineering, plasma physics, and radiation hydrodynamics.

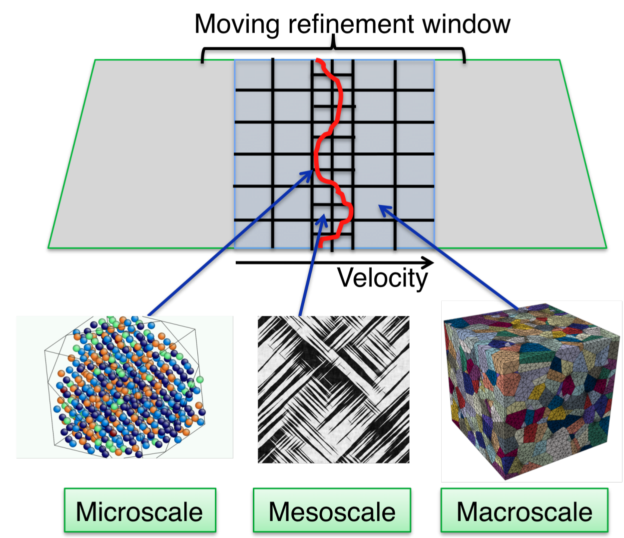
Co-Design Project Objectives

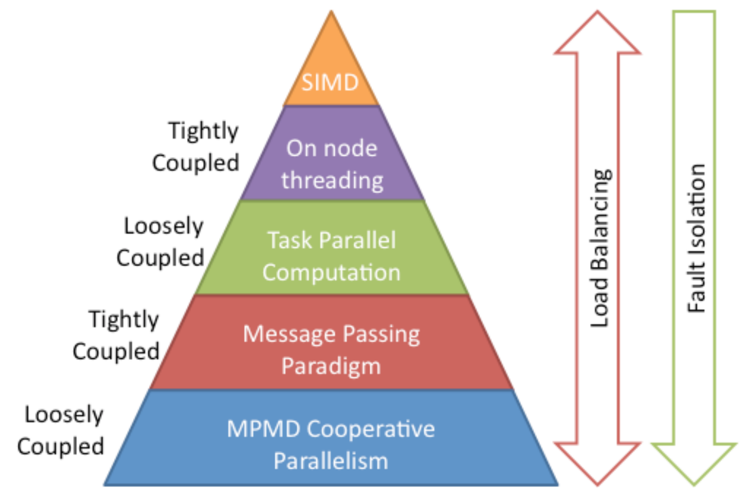
- **Inter-communication of requirements and capabilities between the materials science application community and the exascale hardware and software community**
  - Proxy apps communicate the application workload to the hardware architects and system software developers, and are used in models/simulators/emulators to assess performance, power, and resiliency.
  - Exascale capabilities and limitations will be continuously incorporated into the proxy applications through an agile development loop.
  - Single-scale SPMD proxy apps (e.g. molecular dynamics) will be used to assess node-level data structures, performance, memory and power management strategies.
  - System-level data movement, fault management, and load balancing techniques will be evaluated via the asynchronous task-based MPMD scale-bridging proxy apps.
- **Perform trade-off analysis between competing requirements and capabilities in a tightly coupled optimization loop**
  - A three-pronged approach combining:
    - Node- to system-level models and simulators
    - Exascale emulation layer (GREMLIN) to introduce perturbations similar to those expected on future architectures
    - Performance analysis on leadership-class machines
  - Co-optimization of algorithms and architectures for price, performance, power (chiefly memory and data movement), and resilience (P^3R)



Co-Design Project Objectives

- **Full utilization of exascale concurrency and locality**
  - Heterogeneous, hierarchical MPMD algorithms map naturally to anticipated heterogeneous, hierarchical architectures.
  - Escape the traditional bulk synchronous SPMD paradigm, improve data locality and reduce I/O burden.
- **Application friendly programming models**
  - Must expose hardware capabilities to the application programmer while at the same time hiding the continuous flux and complexity of the underlying hardware through a layer of abstraction that will aid portability.
  - Task-based MPMD approach leverages concurrency and heterogeneity at exascale while enabling novel data models, power management, and fault tolerance strategies.





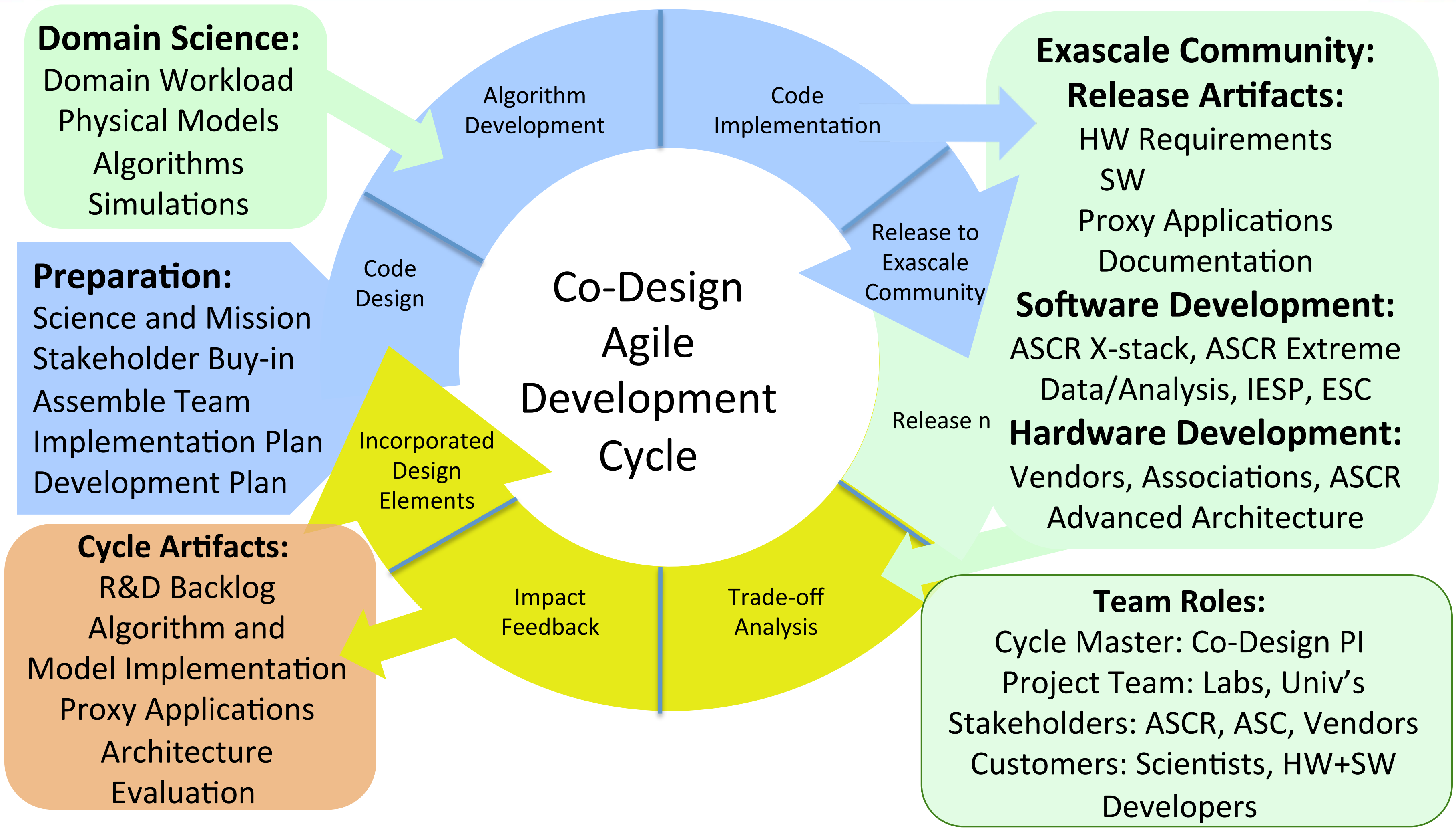
Management Plan

We will manage by adaptive rather than predictive planning.

- **Agile development is an adaptive cycle in which**
  - Initial requirements are gathered from the hardware, software, and domain application communities (e.g. Gordon Bell Prize-winning applications).
  - Application requirements for hardware and software are continuously released to the exascale community in the form of proxy applications and documentation (release artifacts).
  - Application, software, and hardware communities analyze and respond to trade-offs with new requirements and capabilities, both from and to the application.
  - Changes in hardware and software designs are rapidly adapted into proxy applications (cycle artifacts).
  - Repeated iterations converge to the optimal design for the exascale simulation environment for real science applications.

Co-Design Requires Adaptive Methodologies.

Management Plan



To successfully define this exascale simulation environment, our co-design process must be *adaptive, iterative,* and *lightweight* – i.e. agile.

Co-Design Project Roadmap (May 2011)

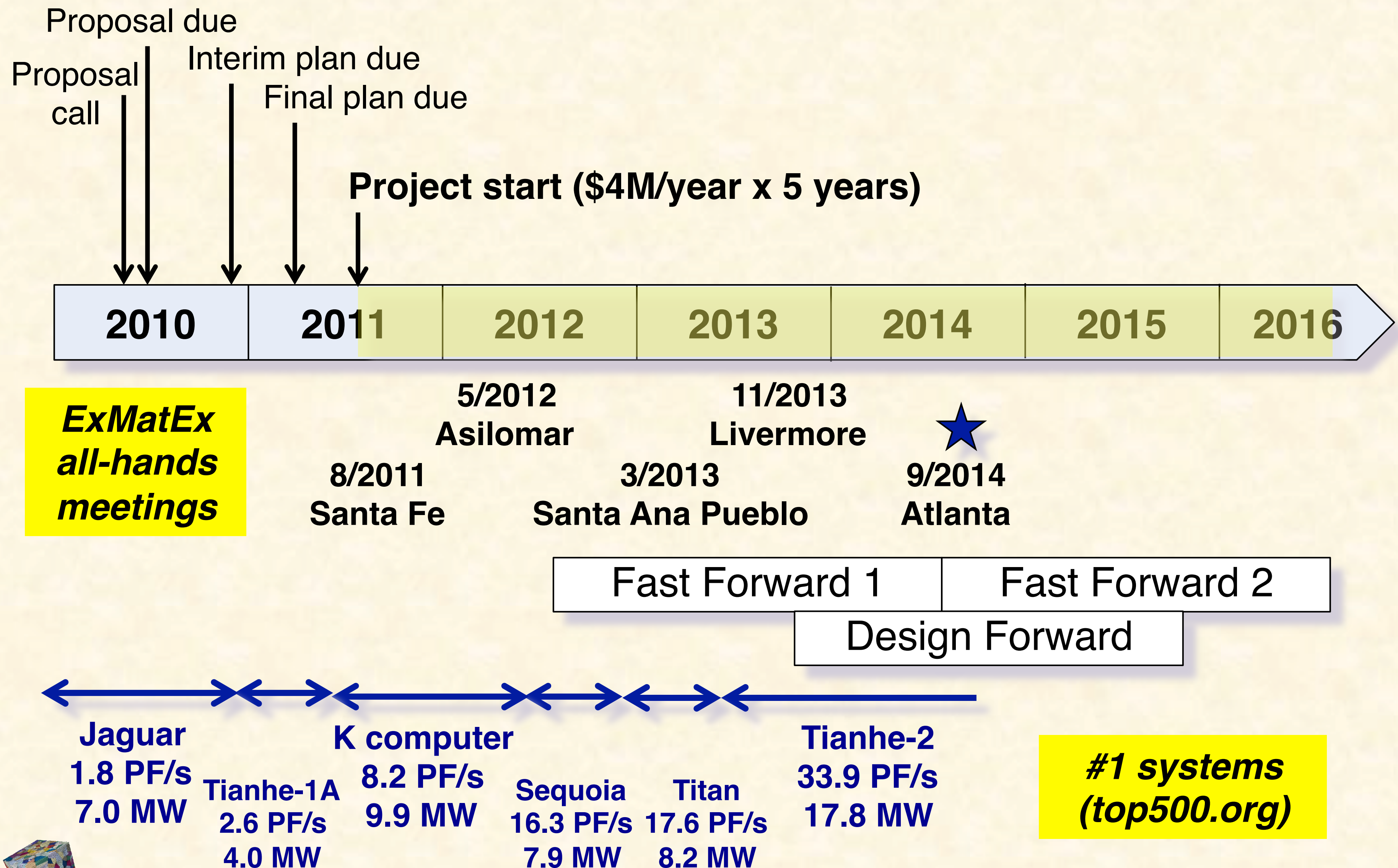
Focus Area	Level 1	Level 2 milestones				
		Year 1	Year 2	Year 3	Year 4	Year 5
Proxy apps	**Y1: Release initial proxy application suite**	1.1 Single-scale SPMD and 2-scale MPMD proxy apps	2.4 Release analysis tool extensions and proxy apps	3.3 Release analysis tool extensions and proxy apps	4.3 Scale-bridging MPMD proxy app	5.4 Deliver open-source exascale materials proxy applications suite
Scale-bridging algorithms	**Y4: Demonstrate scale-bridging on 10+ PF platform**	1.4 Assess and extend scale-bridging algorithms	2.3 Assess data/ resource sharing requirements	3.4 Develop stable, accurate, adaptive macro/meso scale-bridging	4.1 Demonstrate data/resource sharing at 10 PF	
Programming models			2.2 Identify critical features of programming models	3.1 Node-level DSL to coordinate execution and data exchange	4.4 Assess and deliver requirements for task/thread scheduler	
P^3R analysis and optimization		1.2 Evaluate initial single-scale and scale-bridging proxy apps using ASPEN, SST, and scalable tools	2.1 SST/GREMLIN layer	3.2 Develop OUQ V&V framework for multiscale 3.5 Evaluate power management strategies	4.2 Develop and assess fault tolerance strategies and provide API requirements to SW partners	5.1 Deliver documented requirements to HW vendors 5.2 Deliver documented constraints to SW partners
Other	**Y5: Deliver integrated design specification for exascale materials @ extremes**	1.3 Establish liaisons and engagement strategies with exascale HW and SW ecosystem				5.3 Deliver prototype of limited scale-bridging materials science capability

All ExMatEx activities are focused on the two ultimate objectives.

(1) Demonstrating and delivering a ***prototype*** scale-bridging materials science application based upon adaptive physics refinement.

(2) Identifying the ***requirements*** for the exascale ecosystem that are necessary to perform computational materials science simulations (both single- and multi-scale).

ExMatEx project history

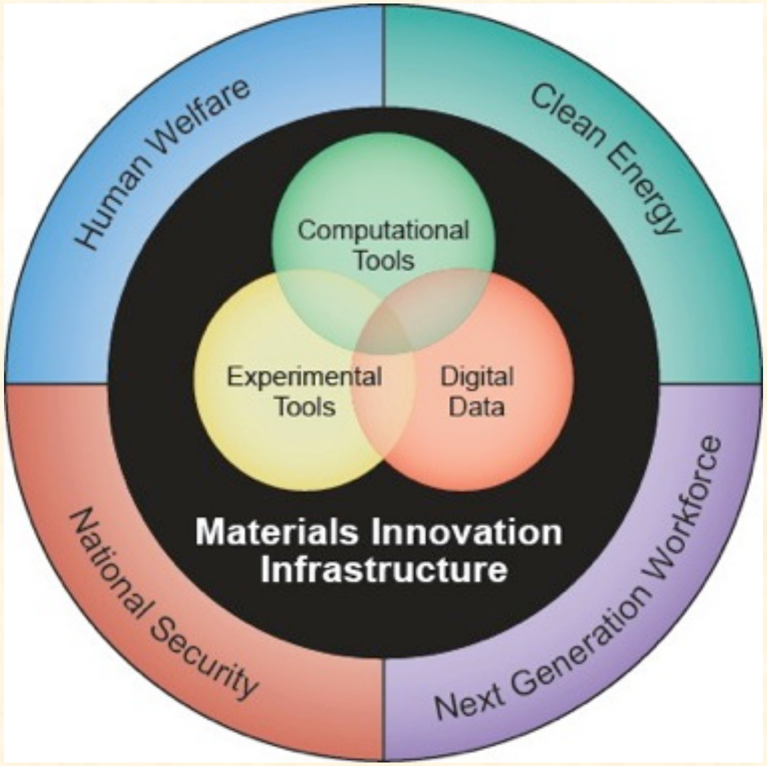


Modeling and simulation is playing an increasing role in materials design and certification

- High-strength, light-weight structural materials are required for products from cars and airplanes to gas, wind, and jet turbine blades

Boeing Frontiers (2010)

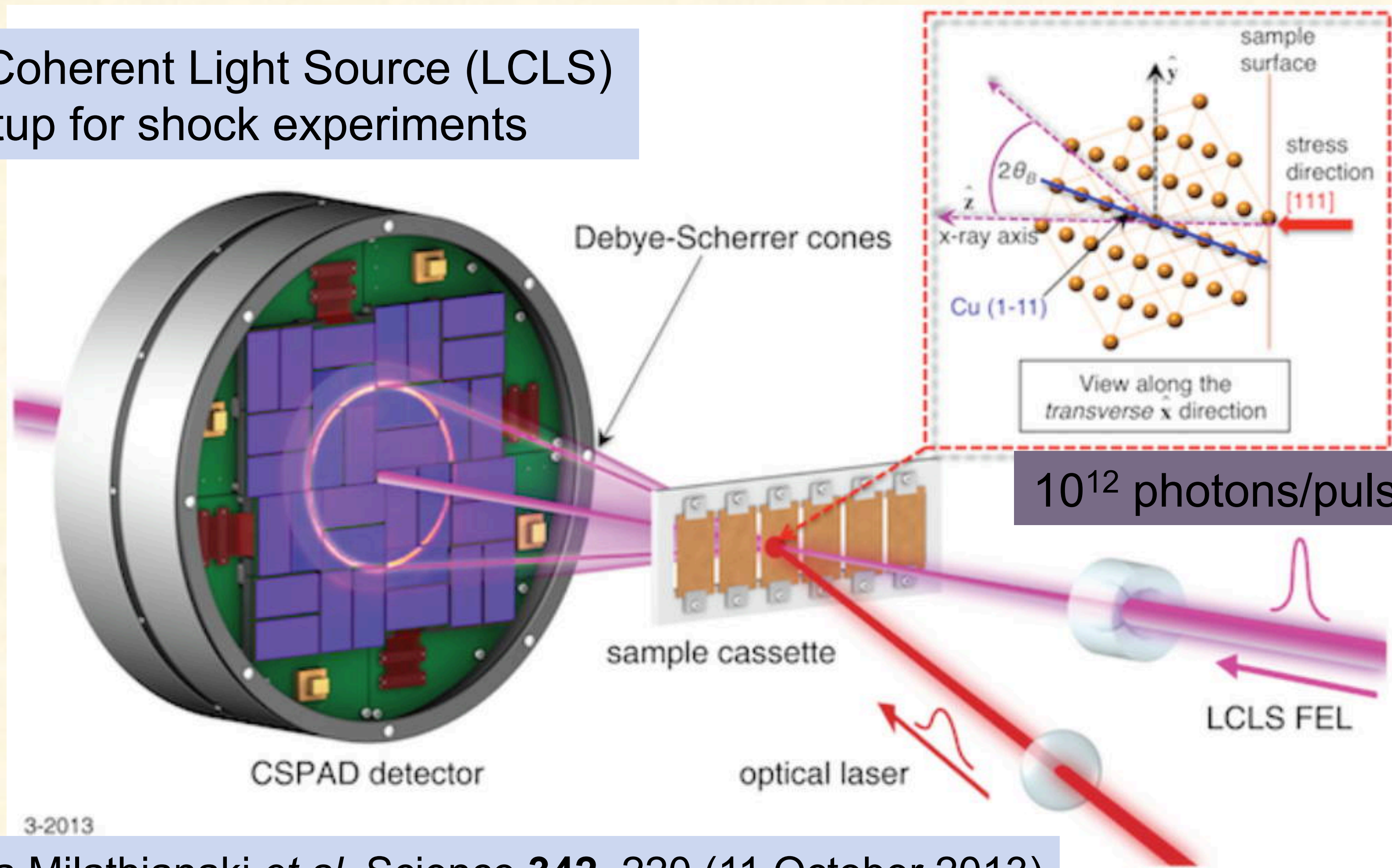
- Materials Genome Initiative



http://www.whitehouse.gov/MGI

New light sources such as LCLS and APS are providing unprecedented resolution and data challenges.

Linac Coherent Light Source (LCLS) setup for shock experiments



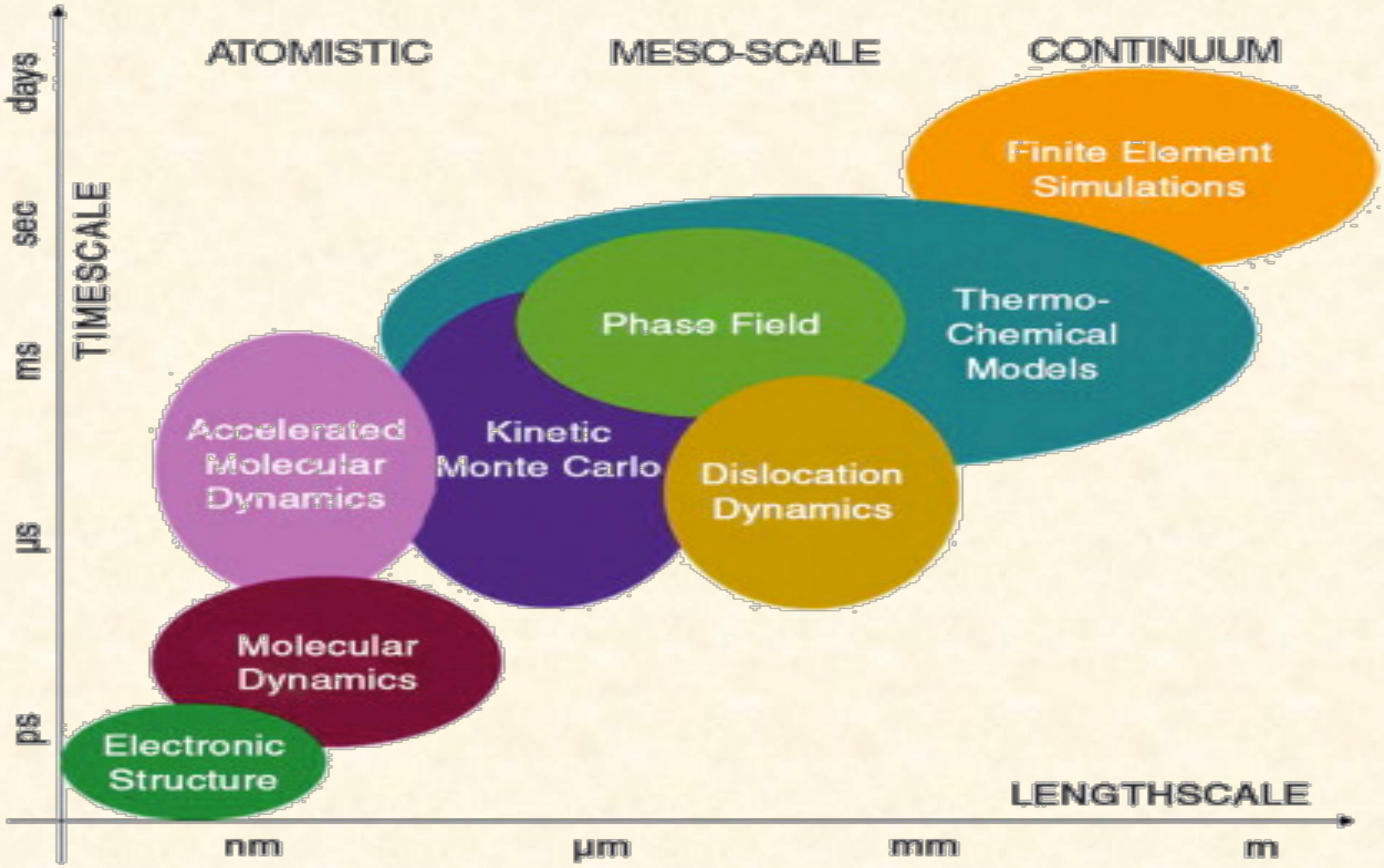
10^{12} photons/pulse

Despina Milathianaki *et al*, Science **342**, 220 (11 October 2013)

ExMatEx

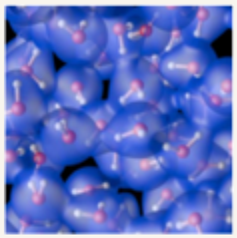
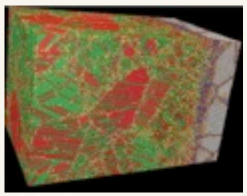
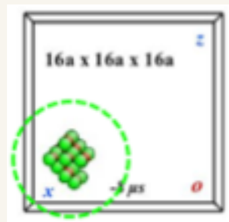
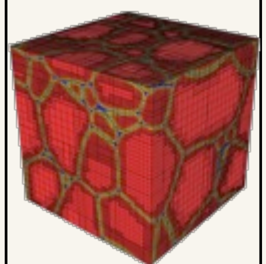
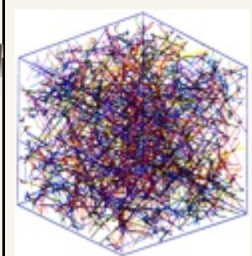
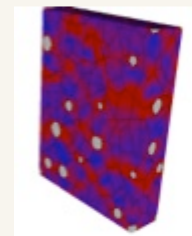
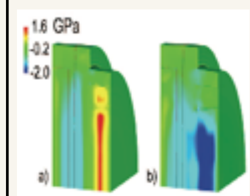
U.S. DEPARTMENT OF ENERGY Office of Science

Computational materials science involves a hierarchy of length and time scales



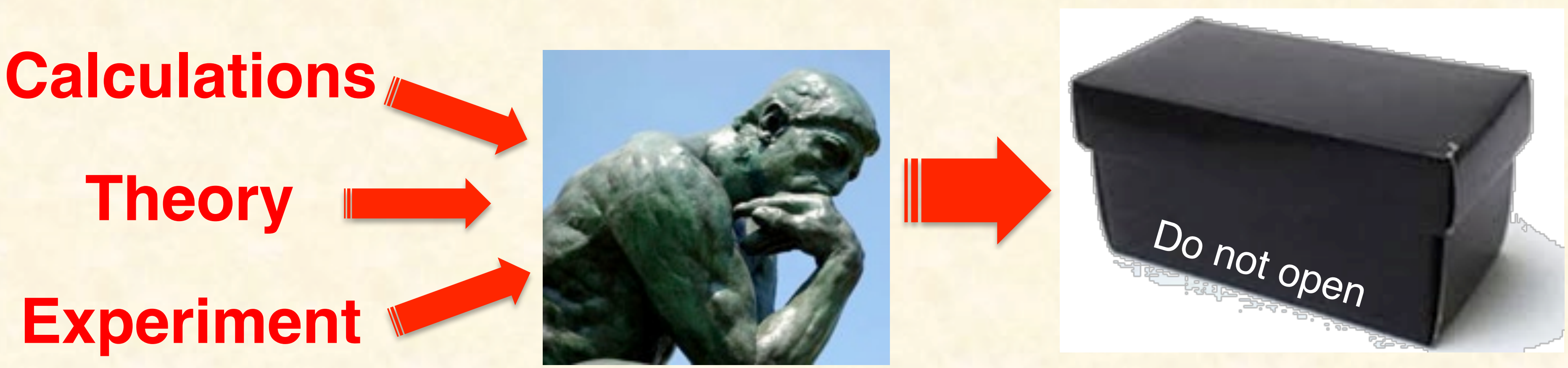
M. Stan, Materials Today **12**(11), 20 (2009)

Seven pillars of computational materials science

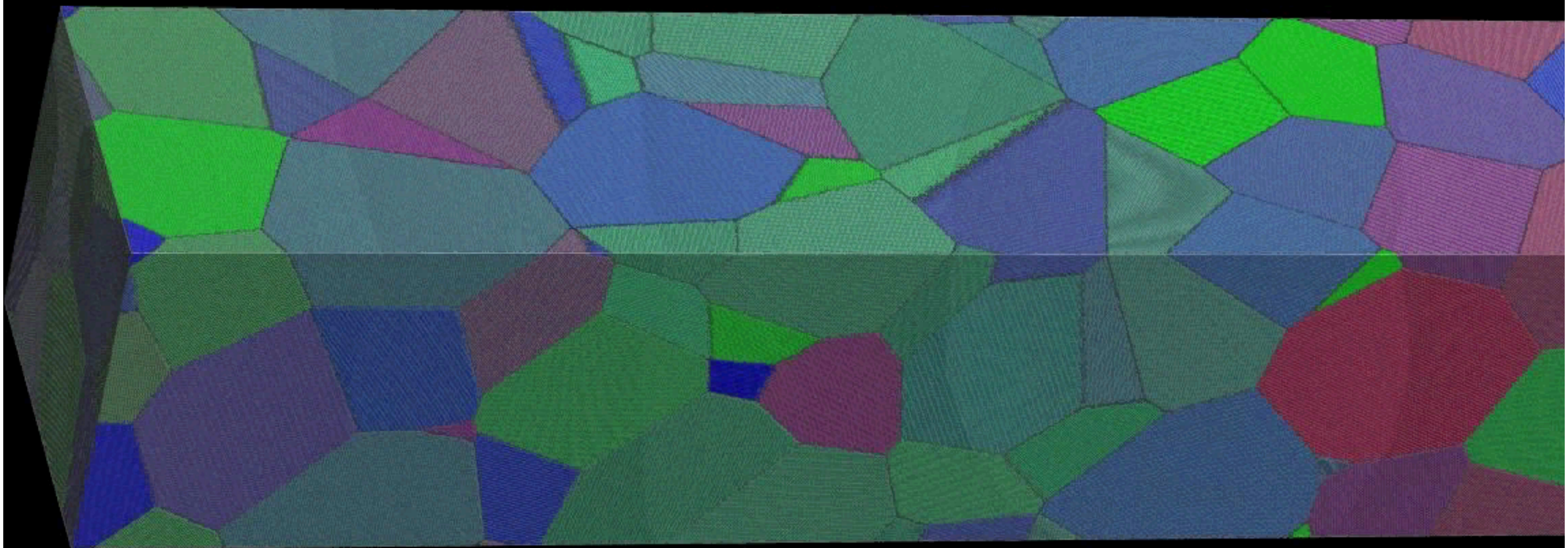
Ab-initio	MD	Long-time	Phase Field	Dislocation	Crystal	Continuum
Inter-atomic forces, EOS	**Defects and interfaces, nucleation**	**Defects and defect structures**	**Meso-scale multi-phase evolution**	**Meso-scale strength**	**Meso-scale material response**	**Macro-scale material response**
						
Code: Qbox/ LATTE Motif: Particles and wavefunctions, plane wave DFT, ScaLAPACK, BLACS, and custom parallel 3D FFTs Prog. Model: MPI + CUBLAS/ CUDA	Code: SPaSM/ ddcMD/CoMD Motif: Particles, explicit time integration, neighbor and linked lists, dynamic load balancing, parity error recovery, and *in situ* visualization Prog. Model: MPI + Threads	Code: SEAKMC Motif: Particles and defects, explicit time integration, neighbor and linked lists, and *in situ* visualization Prog. Model: MPI + Threads	Code: AMPE/ CoGL Motif: Regular and adaptive grids, implicit time integration, real-space and spectral methods, complex order parameter Prog. Model: MPI	Code: ParaDis Motif: "segments" Regular mesh, implicit time integration, fast multipole method Prog. Model: MPI	Code: VP-FFT Motif: Regular grids, tensor arithmatic, meshless image processing, implicit time integration, 3D FFTs. Prog. Model: MPI + Threads	Code: ALE3D/ LULESH Motif: Regular and irregular grids, explicit and implicit time integration. Prog. Model: MPI + Threads

Traditional approach to subscale models: "sequential multiscale"

- Subscale models (e.g. interatomic potentials, equation of state and strength models) are developed from a combination of theory, experiment, and simulation.
  - *The specific combination depends on the developer, and may involve as much art as science.*



Shock-induced plasticity and twinning of nanocrystal Ta



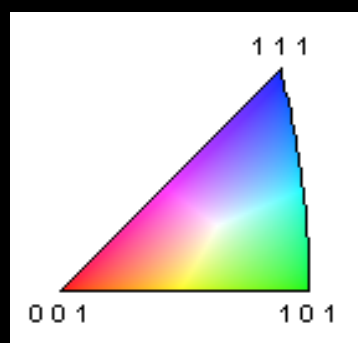
50nm grains

90x90x600 nm

~270 M atoms

u_p=1.2 km/s

P_H= 100 GPa



R. Ravelo, T.C. Germann, et al, *Phys Rev B* **88**, 134101 (2013)

Nov 2013

Milathianaki et al, *Science* **342**, 220 (2013)

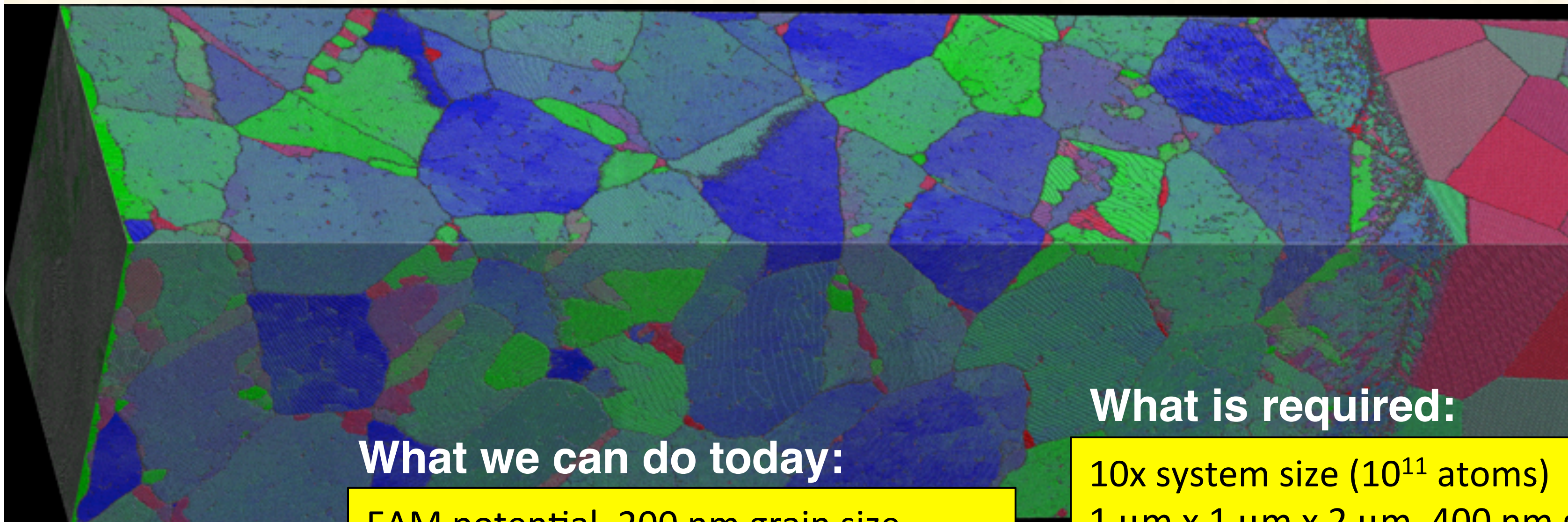
“First, the stress also serves as a **direct test of supercomputer simulations that model how metals behave. The better the data that goes in, the more reliable are the results that come out. That's important in trying to model the exact behavior of metals under stress, say the crash of a car or the impact of a bullet into armor.** And it's especially important for the Office of Science, since several of its labs are home to world-class supercomputers, which researchers are using for everything from simulating the 'subatomic soup' of the early universe to modeling air turbulence and thereby improving airplane performance.

Those better metal models could, in turn, lead to the design of even stronger and more durable materials. And those materials might come in handy for technologies that operate in extreme environments, such as shielding for satellites and space probes. They'll likely be useful in more everyday applications too.”

Exascale use case: competing dislocation, twinning, and/or phase transitions under shock loading

Direct non-equilibrium molecular dynamics simulation matching time and length scales of LCLS experiments:

- *~1-2 μm thick nanocrystalline samples (Cu, Ti, Fe, Ta), ~400 nm grain size*
- *Laser drive: 10-20 ps rise time, 150 ps duration*
- *50 fs duration X-ray "snapshot" interrogation pulses at 10 ps intervals*



NEMD simulation of shocked nc-Ta on Cielito (R. Ravelo, LANL/UTEP)

What we can do today:

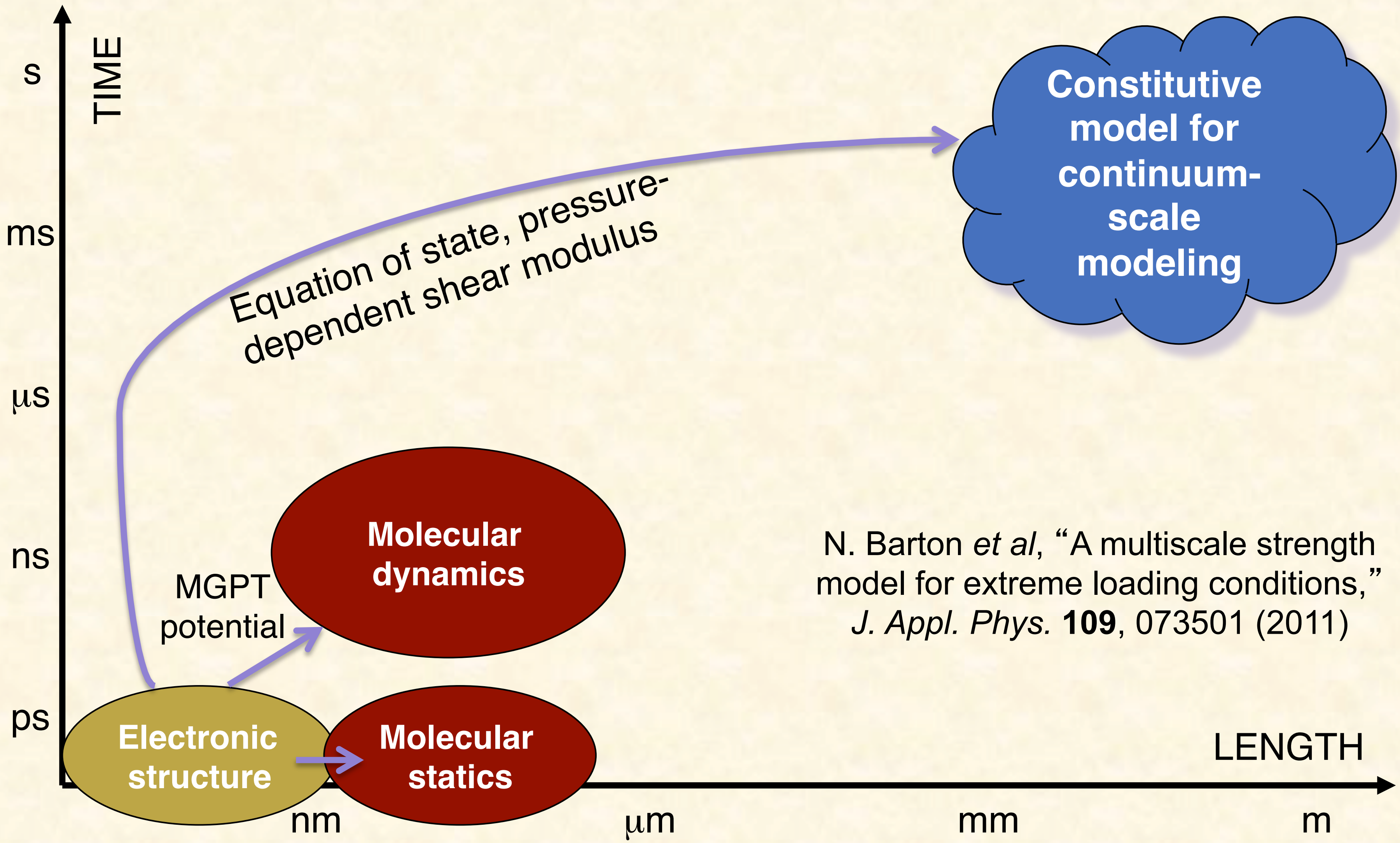
EAM potential, 200 nm grain size
10^{10} atoms (0.5 μm x 0.5 μm x 1.5 μm)
Simulation time: 4 nsec (10^6 steps)
Wall clock: 2 days on Mira (½ Sequoia)

What is required:

10x system size (10^{11} atoms)
1 μm x 1 μm x 2 μm, 400 nm grain size

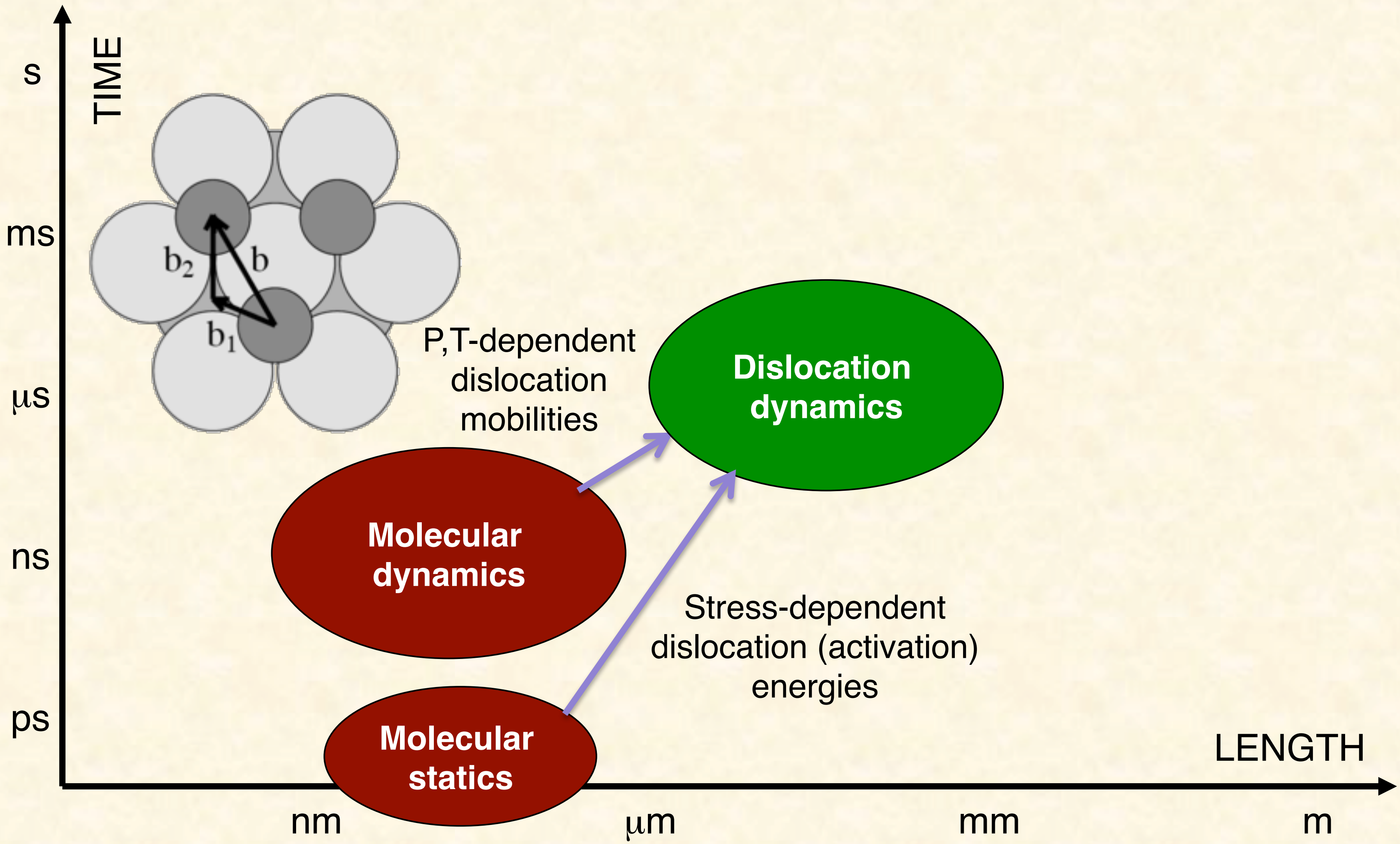
More accurate MGPT potential: 100x
3 weeks on exascale system

Example: "sequential" multiscale strength model

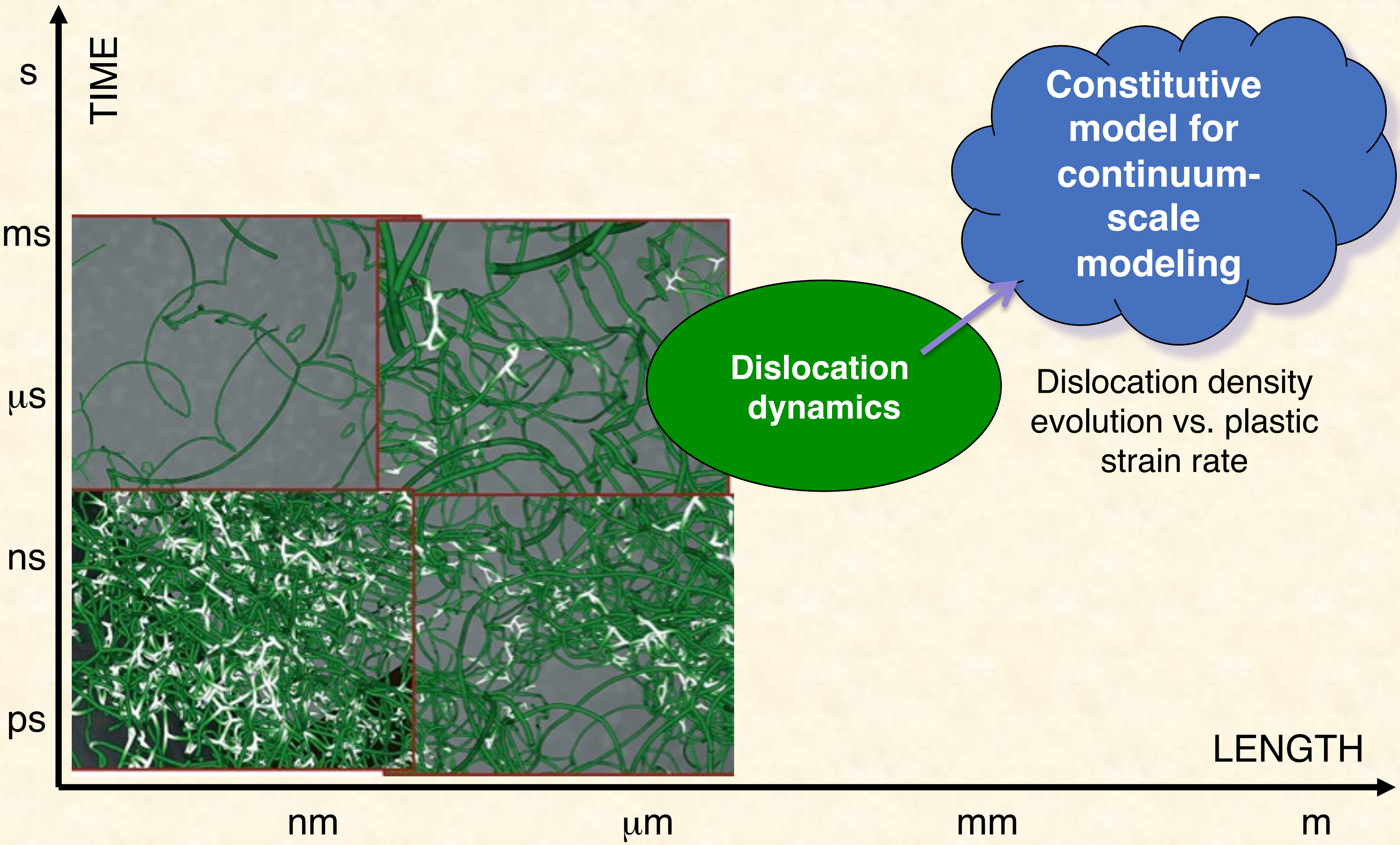


N. Barton *et al*, "A multiscale strength model for extreme loading conditions," *J. Appl. Phys.* **109**, 073501 (2011)

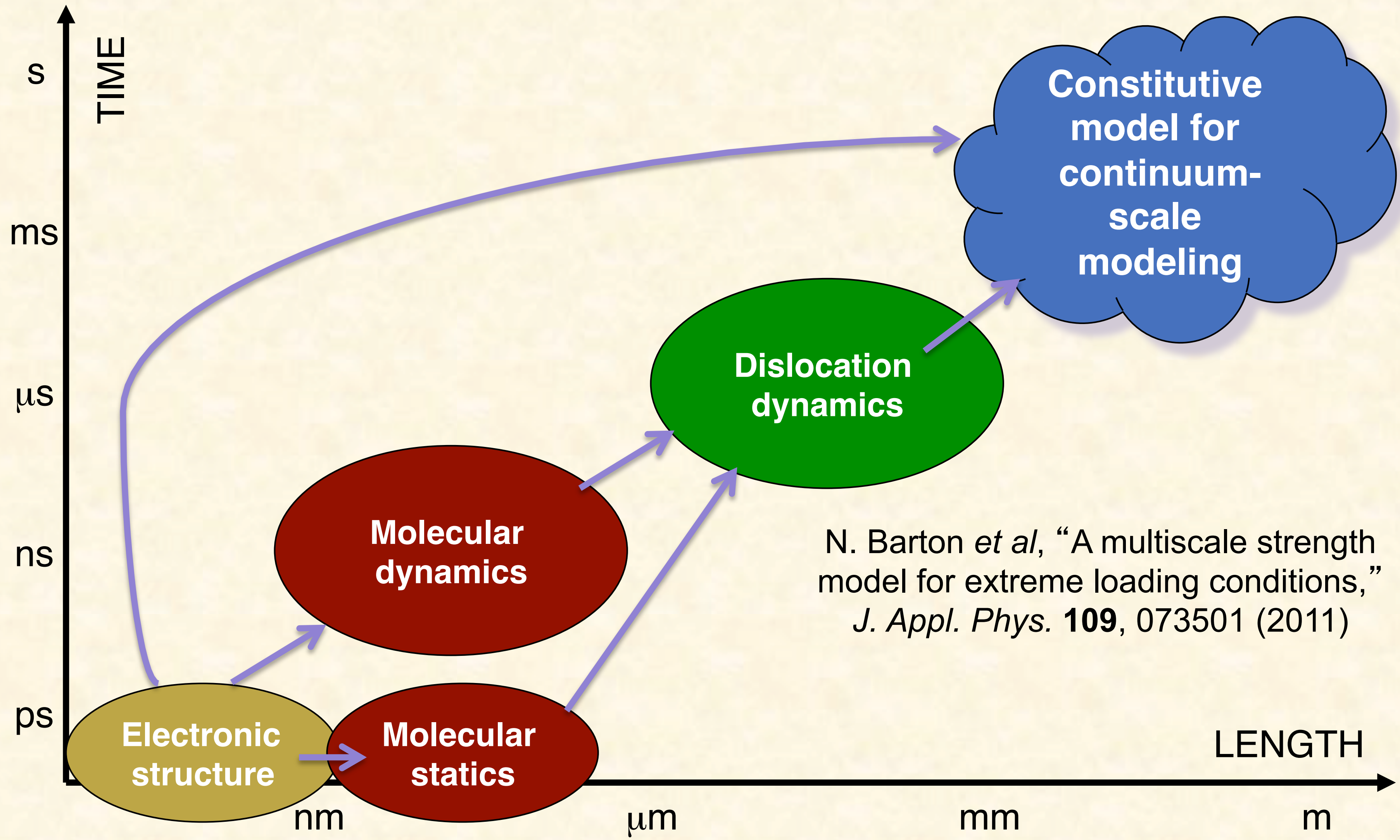
Example: “sequential” multiscale strength model



Example: "sequential" multiscale strength model



Example: "sequential" multiscale strength model

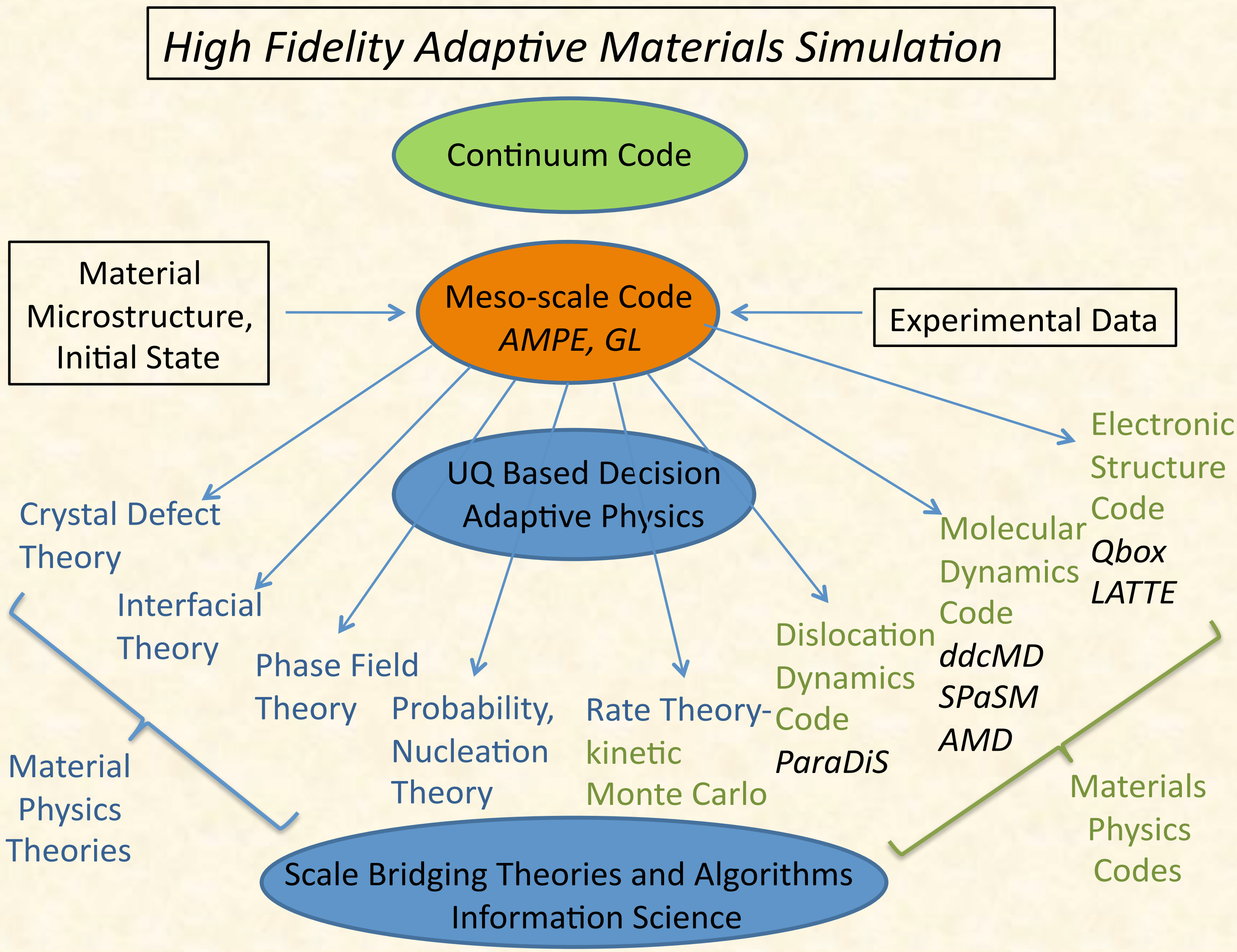


N. Barton *et al*, "A multiscale strength model for extreme loading conditions," *J. Appl. Phys.* **109**, 073501 (2011)

Challenges of a "sequential" multiscale approach

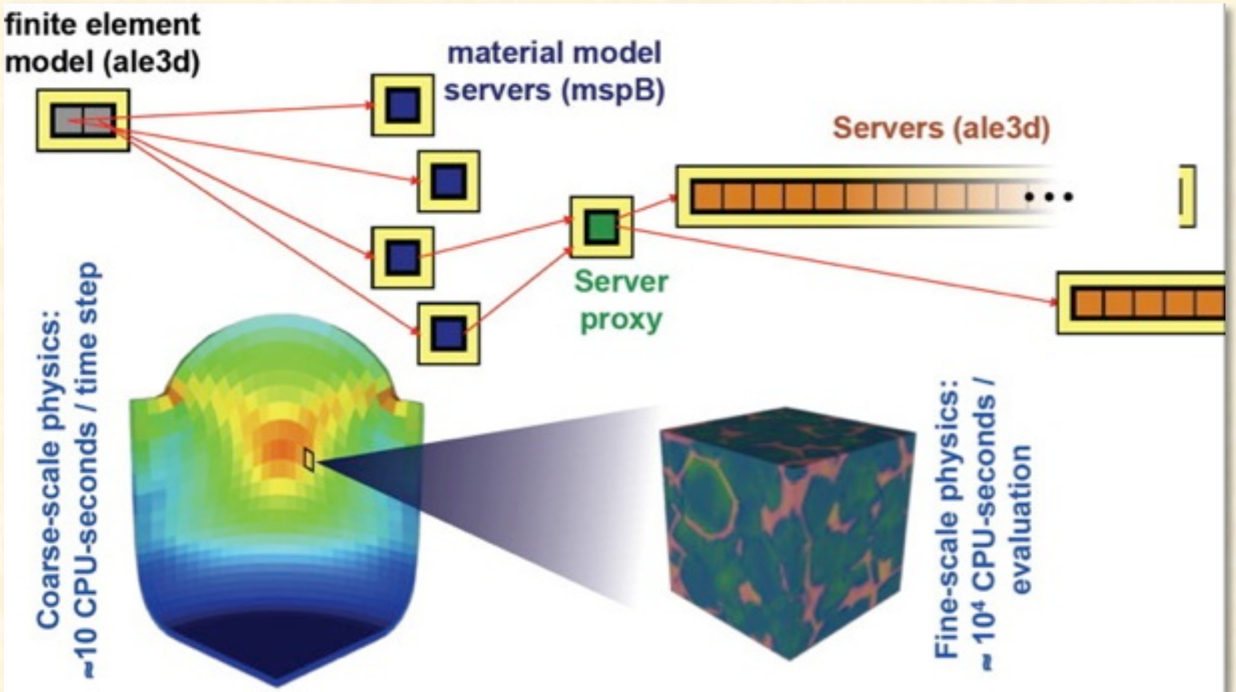
- Information is passed up a hierarchy of coupled length/time scales via a sequence of subscale models and parameters.
- This relies upon understanding how phenomena at shorter length/time scales control the behavior at longer length/time scales.
- Model complexity (and uncertainty) grows with each new physical mechanism.
  - *E.g. adding twinning and/or phase transformations to dislocation-based strength model*
  - *May need to account for coupling/competition between different physical processes*
  - *How does one include path (history) dependence (e.g., what is the strength of a material that has melted and then recrystallized?)*

"Adaptive physics refinement" inverts the traditional sequential scale bridging



Adaptive sampling techniques have been demonstrated under the LLNL "Petascale Initiative" LDRD.

- 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.
- This requires dynamic workload balancing in a task parallel context.



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)

N. R. Barton *et.al.* A call to arms for task parallelism in multi-scale materials modeling. *Int. J. Numer. Meth. Engng* **86**, 744–764 (2011)

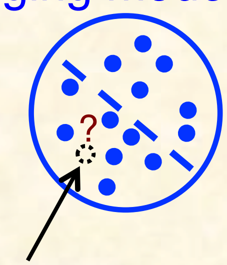
Kriging estimates are based on previously computed fine-scale responses.

Fine-scale responses accumulated in a database are interpolated (with error estimation) via a kriging algorithm.

• = fine scale evaluation

--- = linear regression model

Kriging model 1



Sample point near existing model and satisfies tolerance:

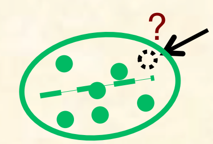
- Just interpolate (saves fine-scale evaluation)

Kriging model 2

Sample point too far from existing models:

- Evaluate fine scale
- Create new model

Kriging model 3



Sample point near existing model, but fails error tolerance:

- Evaluate fine scale
- Add to existing model

Use Case: Shaped-charge jets, breakup and 3D effects (e.g. spinning) require crystal plasticity and anisotropy

What is required:

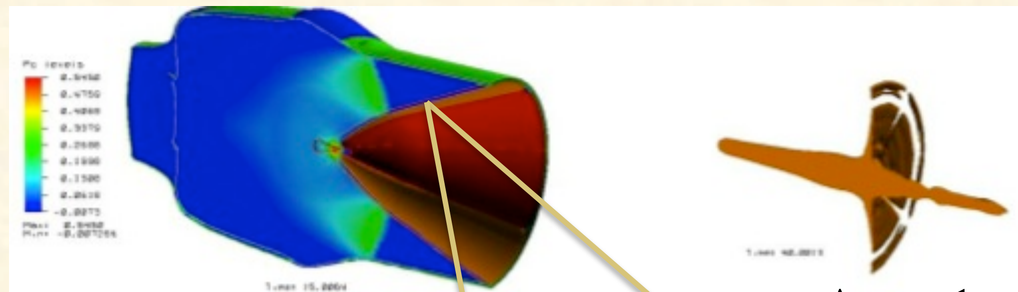
Resolution: 10^{12} zones (10 cm cube)
Simulation time: 100 μsec (10^5 steps)
Strain rate: 10^6 /sec
Strain: 1-3

Using Small Strain Crystal Plasticity Model:
~10^4 sec (~3 h) wall clock on 10^9 cores

Large Strain Crystal Plasticity Model: 10x

Twinning / Scale Bridging Model: 100x

ALE3D simulation of shaped-charge jet
(Rose McCallen, LLNL)



$\Delta\varepsilon \geq 1$

$\Delta\varepsilon = 0.15$

What we can do today:

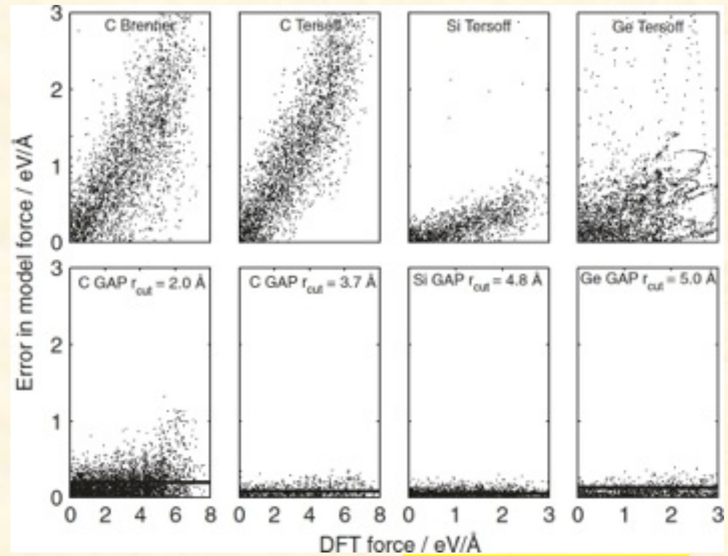
Crystal plasticity simulation of high rate deformation (Nathan Barton, LLNL)
Model: Small Strain Crystal Plasticity
Number Zones: 10^7 (100 micron cube)
Simulation time: 10 μsec (10^4 steps)
Strain rate: 10^6 /sec
Strain: 0.15
Wall Clock: 1 day on 1/10 Cielo

ExMatEx

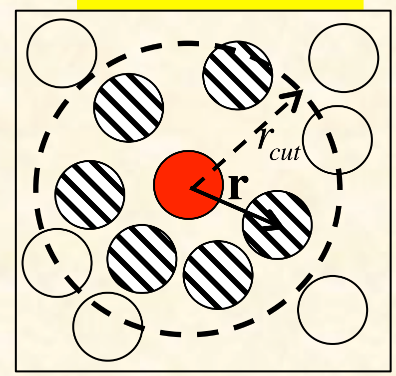
U.S. DEPARTMENT OF ENERGY | Office of Science

Concurrent scale-bridging approaches are being pursued in other materials science contexts

- Directly computing a potential surface from *ab initio* calculations
  - *Distinct from* ab initio *molecular dynamics*
  - *GAP: Gaussian approximation potentials*
    - » Bartók et al, PRL **104**, 136403 (2010)
  - *SNAP: Spectral neighbor analysis potentials*
    - » Aidan Thompson et al, SNL-NM
  - *Configuational database-driven dynamics*
    - » Jones and Shaughnessy, SNL-CA



GAP/SNAP



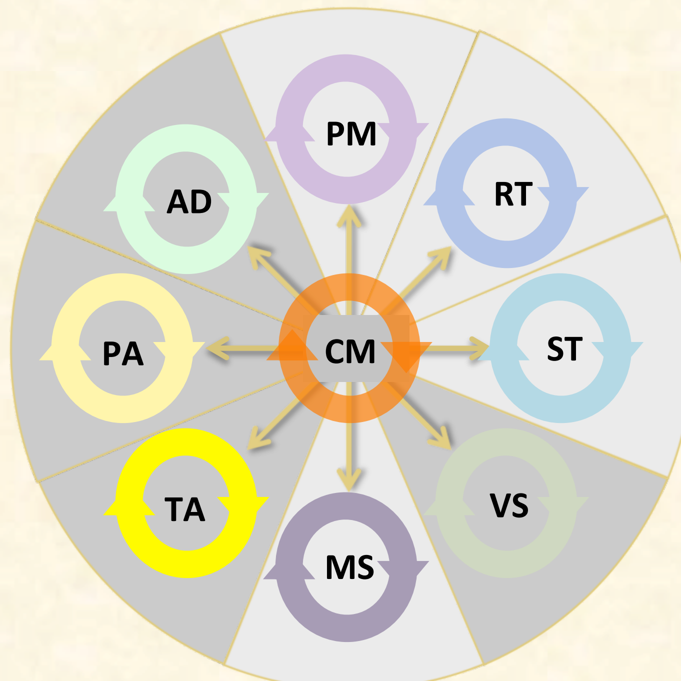
- On-the-fly kinetic Monte Carlo
  - *Henkelman and Jonsson, J. Chem. Phys.* ***115****, 9657 (2001)*
- Self-learning kinetic Monte Carlo
  - *Trushin et al, Phys. Rev. B* **72**, 115401 (2005)
- Self-evolving atomistic kinetic Monte Carlo
  - *Xu, Osetsky, and Stoller, Phys. Rev. B* ***84,*** *132103 (2011)*

Management Plan – Year 1

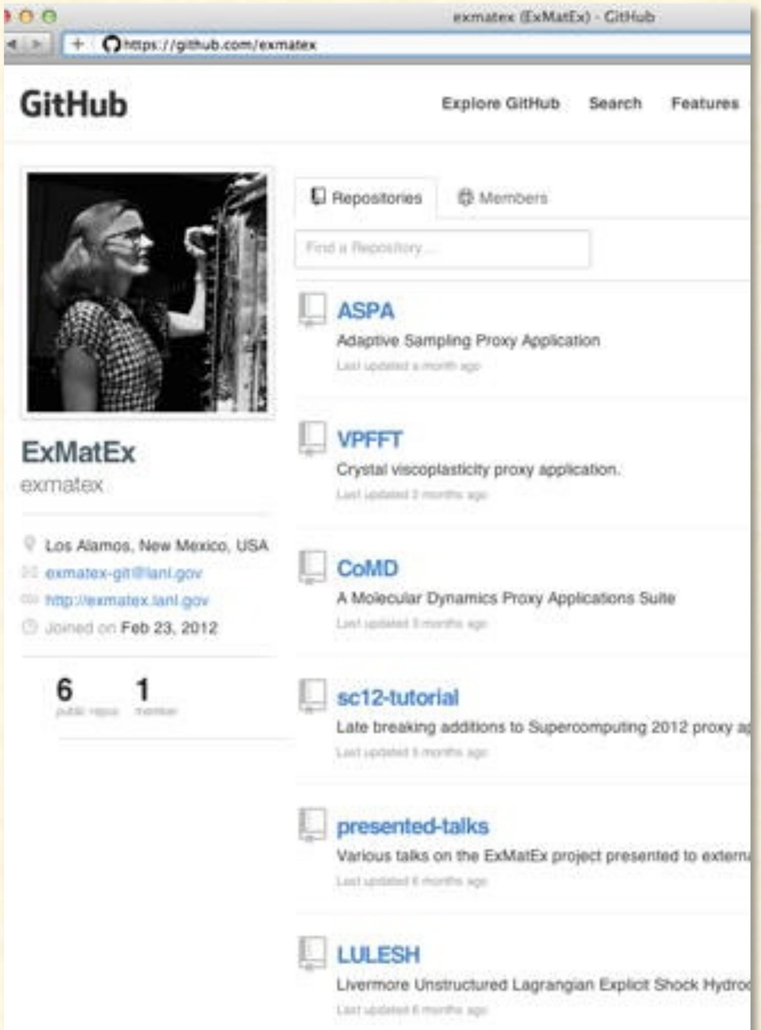
Year	L1 Milestone	Supporting Capabilities / L2 Milestones	Contributing L3 tasks
1	Establish co-design cycle elements, and release initial proxy app suite	1.1 Create initial suite of single-scale SPMD and 2-scale MPMD proxy apps	PA
		1.2 Evaluate proxy apps using ASPEN, SST, and scalable tools	MS, TA
		1.3 Establish liaisons and engagement strategies with exascale software community and vendor partnership(s)	VS
		1.4 Assess and extend scale-bridging algorithms	AD

In Y1 we established the necessary components of the co-design cycle by developing representations:

- **of the applications to the hardware through proxy apps, and**
- **of the hardware to the applications through analysis tools.**



Our focus during the first 18 months was establishing the initial suite of single-scale SPMD proxy apps.

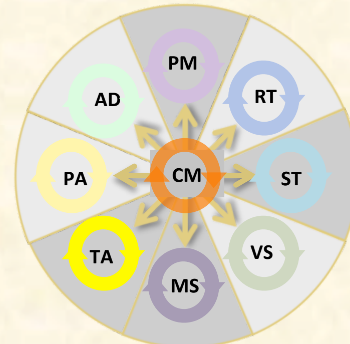


github.com/exmatex

- Single-scale proxies primarily address node-level SPMD issues:
  - *Microscale: CoMD*
    - Molecular dynamics; particle-based
  - *Mesoscale: VPFFT, CoGL*
    - Crystal plasticity, phase field; regular Eulerian grids (Fourier- & real-space alternatives)
  - *Macroscale: LULESH*
    - Shock hydro; unstructured Lagrangian mesh
- CoMD and LULESH are two of the small set (~6) of compact applications that several of the vendor FastForward teams have focused on as part of their projects.
- Several hackathons and deep dives have enhanced this collaboration.

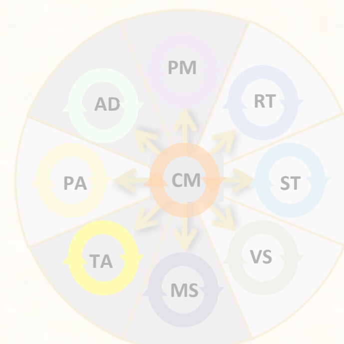
Management Plan – Years 2 and 3

Yr	Supporting Capabilities / L2 Milestones	L3 areas
2	2.1 Use SST simulation and GREMLIN interface layer to mimic exascale machine behavior on petascale platforms	PM, ST, MS
	2.2 Identify critical features of programming models	PM
	2.3 Assess & deliver data/resource sharing requirements, both for scale-bridging and *in situ* analysis/viz, to exascale SW partners	PM, ST
	2.4 Release latest instantiation of ASPEN/SST, GREMLIN, scalable tools used for evaluation and proxy apps to exascale ecosystem	PA, TA



In Y2 we execute the co-design optimization cycle.

Yr	Supporting Capabilities / L2 Milestones	L3 areas
3	3.1 Deliver DSL at kernel level that schedules and coordinates the execution and data interchange between scale-bridging kernels at the node level	PM
	3.2 Develop OUQ V&V framework for hierarchical/multi-scale structures.	AD, PM
	3.3 Release latest instantiation of ASPEN/SST, GREMLIN, scalable tools used for evaluation and proxy apps to exascale ecosystem	PM, TA
	3.4 Develop stable accurate adaptive macro-mesoscale-bridging algorithm	AD
	3.5 Evaluate power management strategies with SPMD proxy apps and provide node-level API requirements to vendor partners	PM, MS, TA

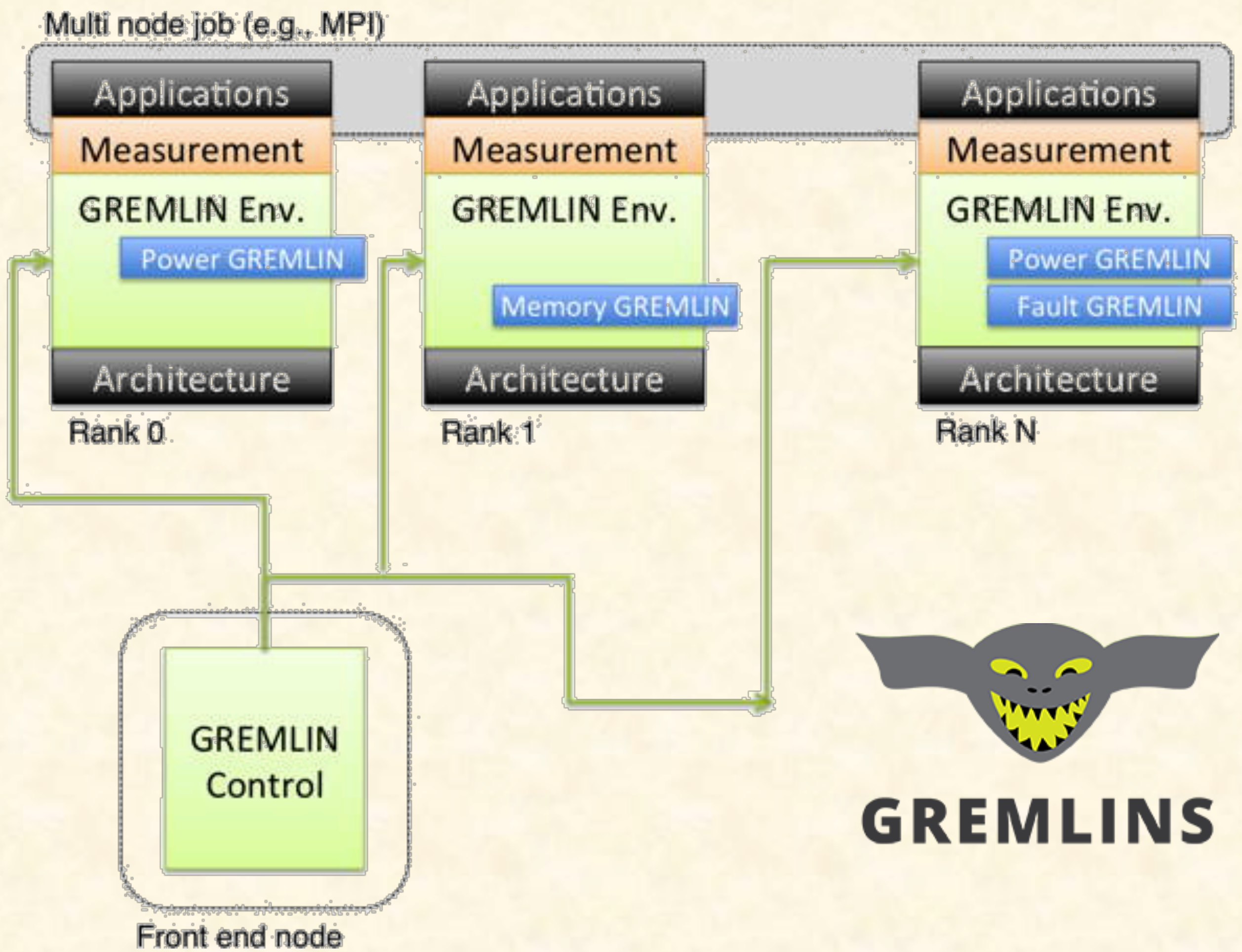


In Y3 we complete the second 18-month optimization cycle and deliver programming model and OUQ V&V frameworks.

2.1) Use SST simulation and GREMLIN interface layer to mimic exascale machine behavior on petascale platforms

We have developed several classes of GREMLINS to evaluate application-level impacts and strategies for:

- Power
- Thermal
- Resilience
  - *Fault injection*
- Memory latency/ bandwidth
  - *Limiting resources*
- Noise
  - *System jitter*



GREMLINS

2.2) Identify critical features of programming models

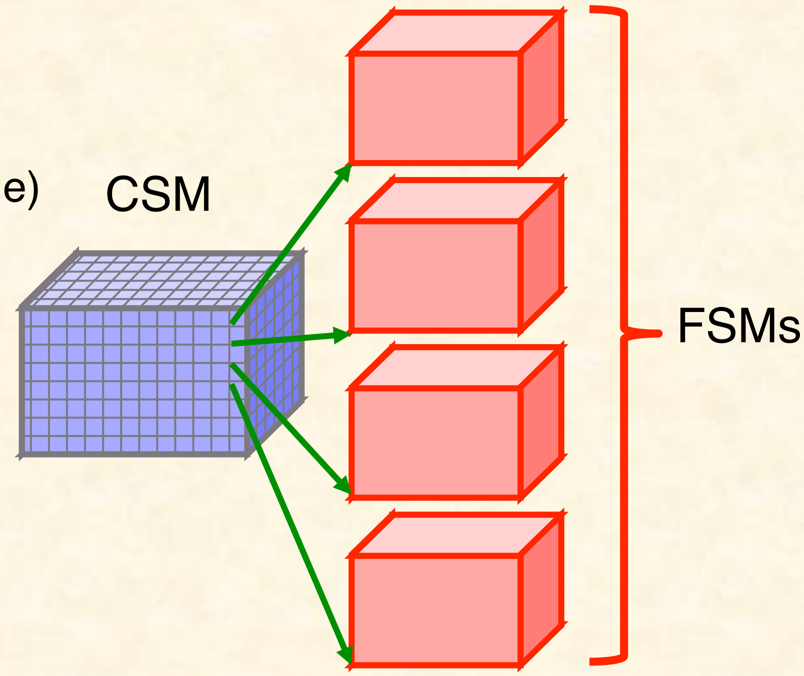
The single-scale proxy apps developed in Year 1, primarily CoMD and LULESH, were used as the primary vehicle for the co-design process, notably several "hackathons" with vendor and X-stack partners.

From these activities, and exploration of various node and component-level programming models, several critical features were identified. Namely, they need to enable the developer to:

- Express control of workflow beyond communicating serial processes
- Express information (e.g. data dependencies) for higher-level dynamic control of workflow
- Express fine grain concurrency
- Express data locality / data layout
- Express asynchrony
- Express heterogeneity and hierarchy

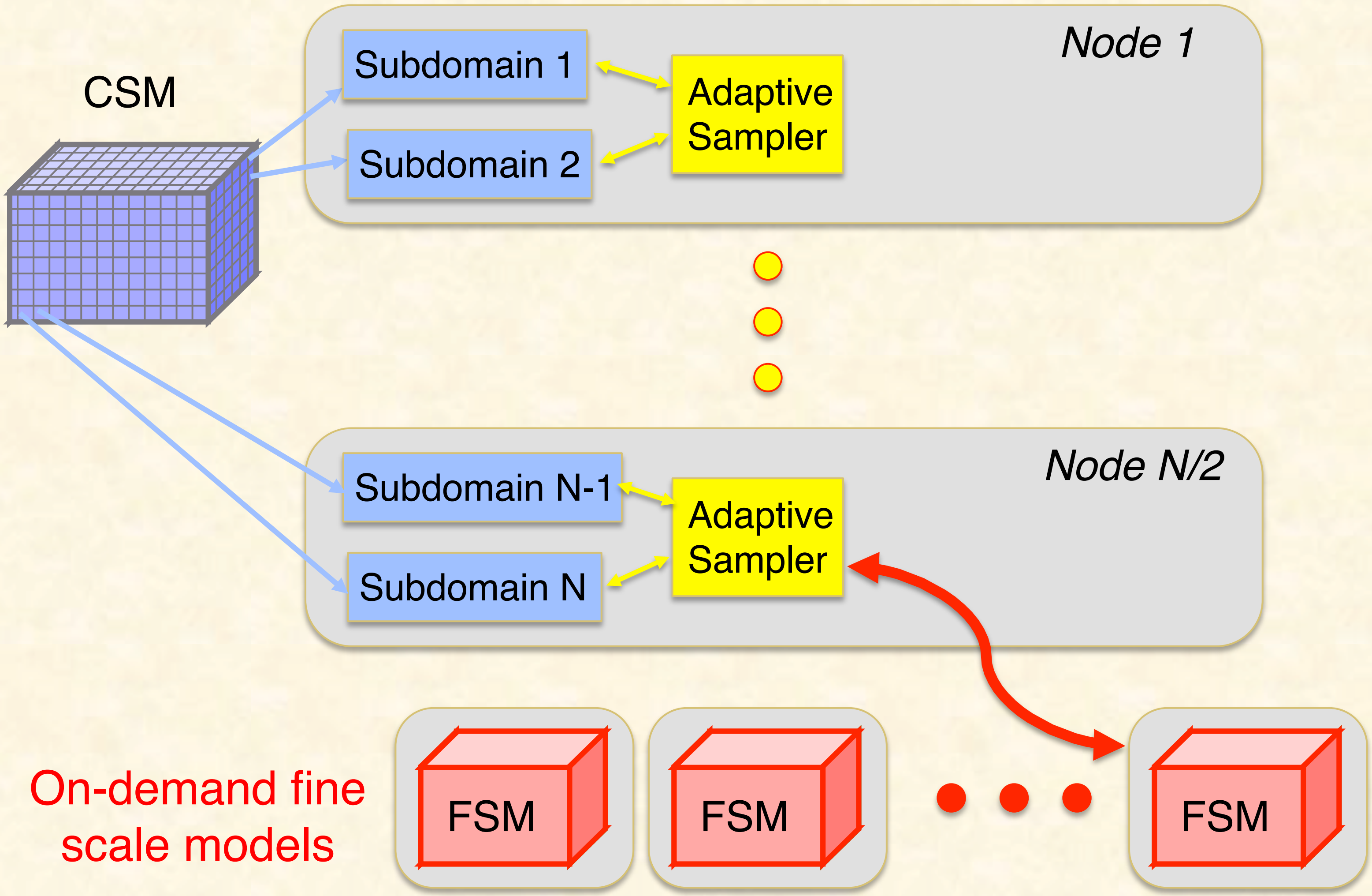
Our work on scale-bridging has followed two complementary paths.

- “Top-down”
  - We (Milo Dorr, LLNL) have developed an Adaptive Sampling Proxy App (ASPA) that represents the fine-scale query, database lookup, and kriging interpolation steps.
  - LULESH (coarse-scale) and VPFFT (fine-scale) proxies are coupled via ASPA to study the workflow for our target application problems.
    - » “Speeds & feeds”
    - » What are the frequency, number, and duration of fine-scale calculations?
    - » What size and type of data are communicated between scales?

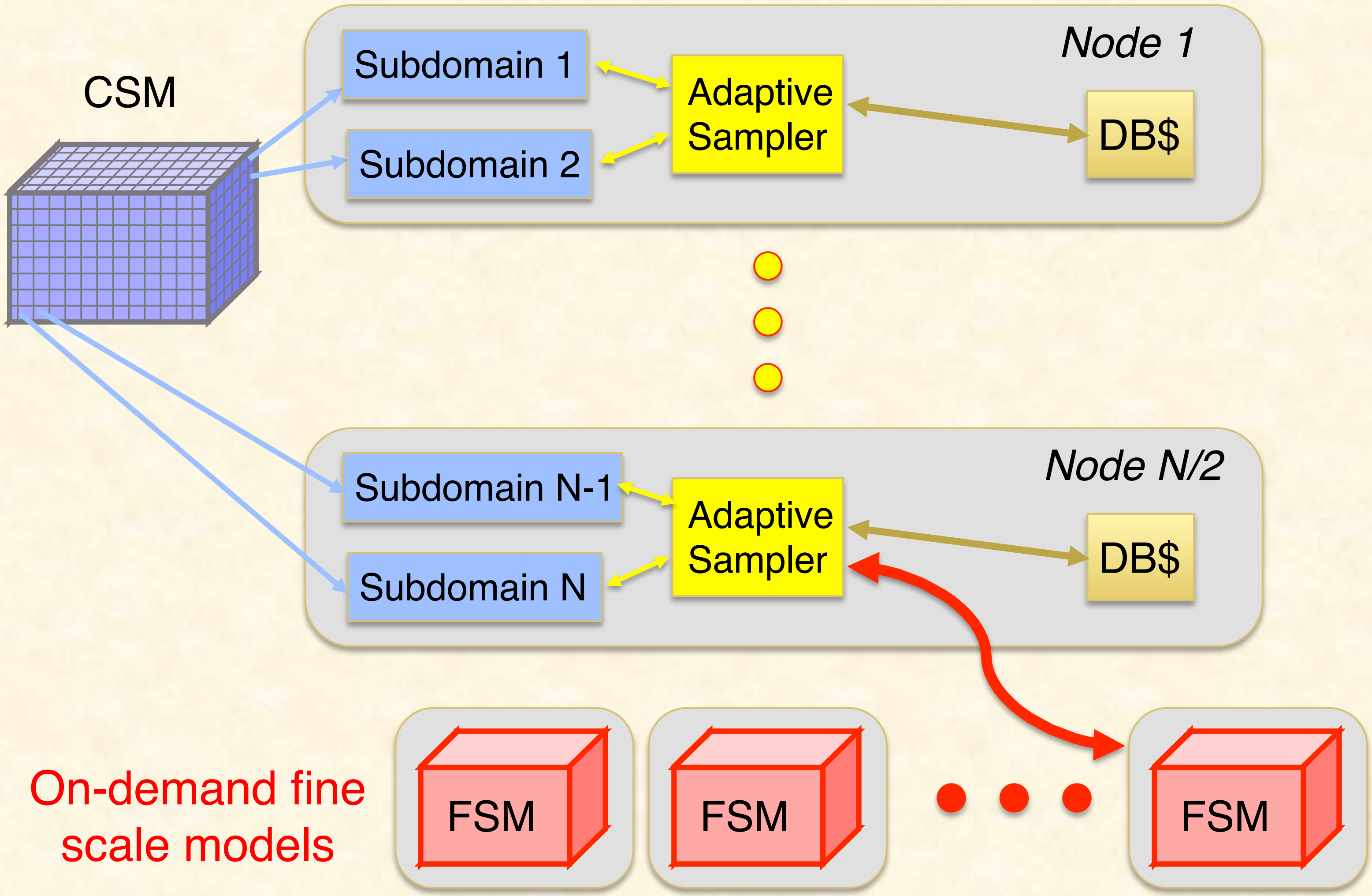


- “Bottom-up”
  - We (Kip Barros et al, LANL) have developed a tractable scale-bridging proxy (CoHMM) that represents the basic task-based modeling approach we are targeting.
  - It is being used to evaluate task-based OS/runtime requirements.

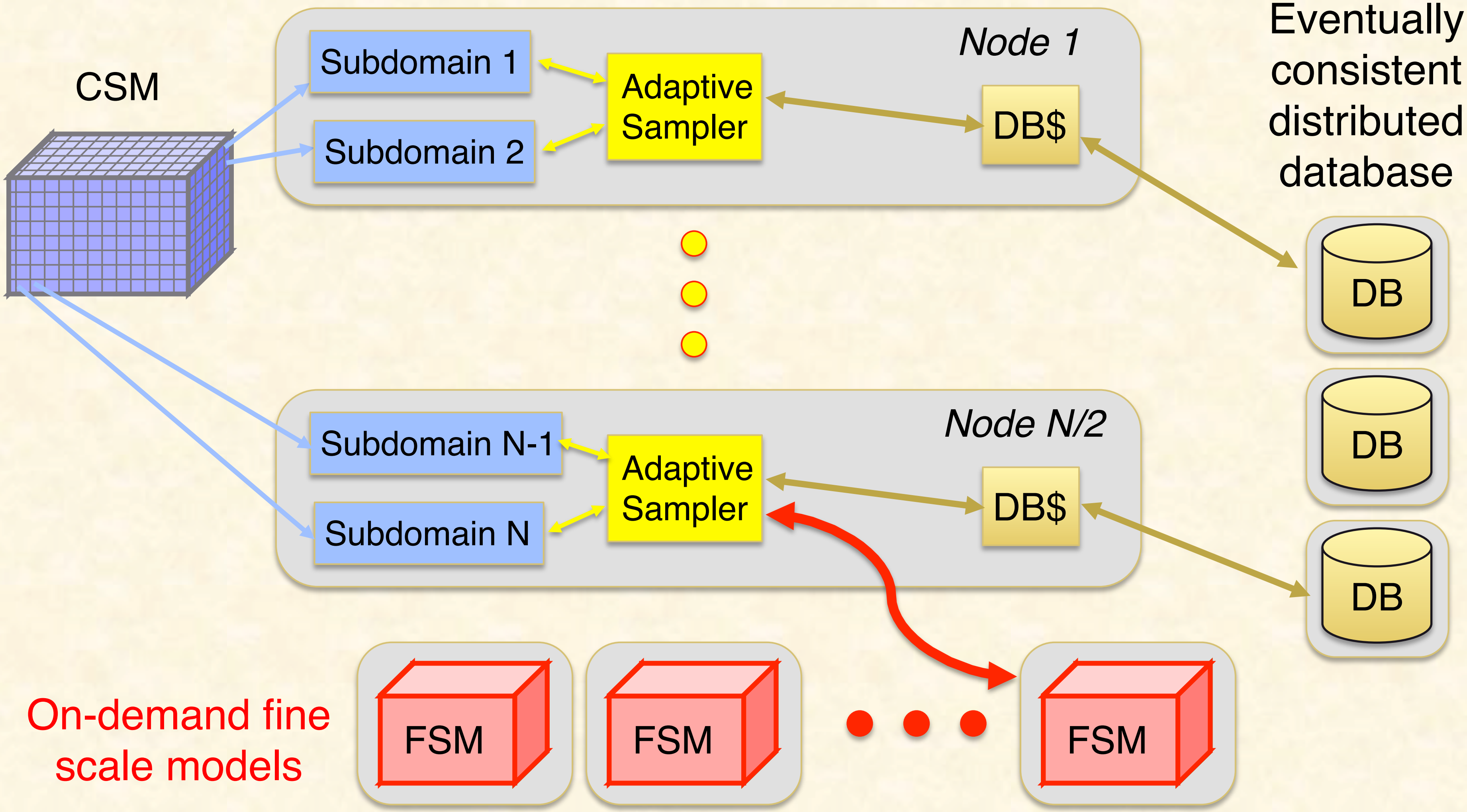
Emerging approach to subscale models: “concurrent multiscale”



Emerging approach to subscale models: "concurrent multiscale"



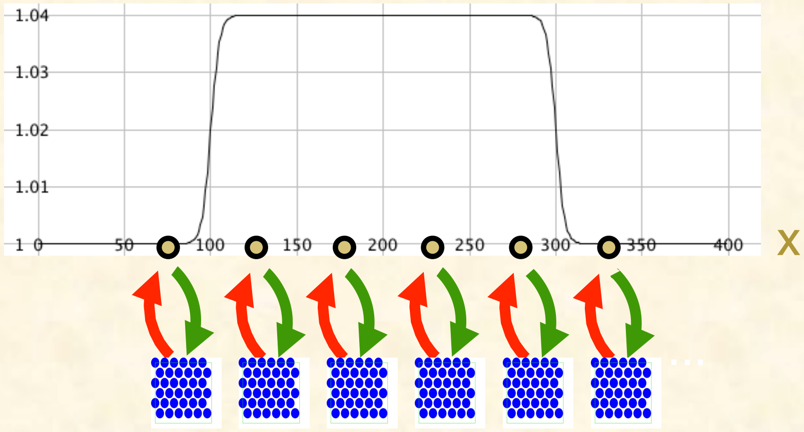
Emerging approach to subscale models: "concurrent multiscale"



We are using the Heterogeneous Multiscale Method* as a scale-bridging prototype

- CoHMM presents the basic workflow requirements of a scale-bridging materials application.
- A full fine scale model (FSM, e.g. a crystal plasticity or molecular dynamics model) is run for every zone & time step of coarse scale model (CSM, e.g. an ALE code).
- It is being used to assess *basic* requirements for task-based runtime systems.
  – *The original HMM* is limited by its predictable, uniform workload pattern.*
  – *Adaptive coarsening provides a more dynamic and realistic workload.*

Deformation gradient



*Xiantao Li and Weinan E, "Multiscale modeling of the dynamics of solids at finite temperature," *J. Mech. Phys. Solids* **53**, 1650–1685 (2005)

Co-Design Project Roadmap (Nov 2013)

Focus Area	Level 1	Level 2 milestones				
		Year 1	Year 2	Year 3	Year 4	Year 5
Proxy apps	**Y1: Release initial proxy application suite**	1.1 Single-scale SPMD and 2-scale MPMD proxy apps	2.4 Release analysis tool extensions and proxy apps	3.6 Release updated proxy apps and analysis tools/ extensions	4.4 Release updated proxy apps and analysis tools/extensions	5.4 Deliver open-source exascale materials proxy applications suite
Scale-bridging algorithms	**Y4: Demonstrate scale-bridging on 10+ PF platform**	1.4 Assess and extend scale-bridging algorithms	2.3 Assess data/resource sharing requirements	3.1 Define scale-bridging targets and smaller-scale prototype app 3.3 Assess scale-bridging uncertainty requirements and implement within prototype app	4.1 Demonstrate petascale data/ resource sharing for scale-bridging target problem	
Programming models			2.2 Identify critical features of programming models	3.2 Establish and document requirements of single-physics and scale-bridging programming models	4.3 Assess and deliver requirements for task/thread scheduler	
P^3R analysis and optimization		1.2 Evaluate initial single-scale and scale-bridging proxy apps using ASPEN, SST, and scalable tools	2.1 SST/ GREMLIN layer	3.4 Use power and resilience analysis to inform programming models and runtime services 3.5 Develop ASPEN model for scale-bridging app, and assess scalability w/coupled ASPEN/SST	4.2 Develop and assess fault tolerance strategies and provide API requirements to SW partners	5.1 Deliver documented requirements to HW vendors 5.2 Deliver documented constraints to SW partners
Other	**Y5: Deliver integrated design specification for exascale materials @ extremes**	1.3 Establish liaisons and engagement strategies with exascale HW and SW ecosystem				5.3 Deliver prototype of limited scale-bridging materials science capability

Agenda: remainder of this morning

9:20 Adaptive Sampling for Materials Science/Engineering.................. Nathan Barton
Lawrence Livermore National Laboratory
Ricardo Lebensohn
Los Alamos National Laboratory

9:30 CoEVP proxy application .... Milo Dorr
Lawrence Livermore National Laboratory

Embedded ViscoPlasticity Scale-Bridging Proxy Application, representing the adaptive sampling scale-bridging workload. Available at: https://github.com/exmatex/CoEVP

10:00 CoHMM proxy application... Kipton Barros
Los Alamos National Laboratory

Simplified elastodynamics scale-bridging workload, based on the Heterogeneous Multiscale Method. Available at: https://github.com/exmatex/CoHMM

10:20 Break

10:30 External co-design and stakeholder engagements.............. Moderator: Jim Belak
Lawrence Livermore National Laboratory

Brief summaries (5-10 minutes each) from partners & stakeholders attending in person or remotely. As appropriate, these summaries may include project goals and objectives, & experiences and/or expectations of application co-design centers.

This afternoon

- Main conference room (w/ BlueJeans videoconference): discussion of materials science & engineering challenge problems and use cases (with external domain science stakeholders), following the suggested agenda below.
- Two small classrooms (each with Polycom) will be available for parallel side discussions on CS topics, including programming models and tools and techniques for performance, power, and resilience (P2R) optimization. WebEx conferencing may also be available for remote participants (contact Martin Schulz).

1:30 ExMatEx single-scale proxy applications.. David Richards
Lawrence Livermore National Laboratory

2:00 ExMatEx Y4 scale-bridging challenge problem(s)............................ Nathan Barton
Lawrence Livermore National Laboratory
Ricardo Lebensohn
Los Alamos National Laboratory

2:30 Open Discussion (refreshments provided)

What aspects of more general materials science & engineering workflows expected over the next 5-10 years do our proxy apps and target problem capture (or miss) for other application areas, including (but not limited to): integrated experiment/simulation (e.g. DCS@APS, MaRIE), materials genome/ICME, additive manufacturing, etc.

Tomorrow morning

- Main conference room (w/ BlueJeans videoconference): discussion of programmability, usability, and tools (with external CS stakeholders), following the suggested agenda below.
- Two small classrooms (each with Polycom) will be available for parallel side discussions. WebEx conferencing may also be available for remote participants (contact Martin Schulz).

8:30 Working Breakfast &
Summaries of CS side discussions..Allen McPherson
Los Alamos National Laboratory

9:00 Recap of 2013-14 co-design summer schools .... Jim Belak
Lawrence Livermore National Laboratory
Christoph Junghans
Los Alamos National Laboratory

9:30 Open Discussion..Moderator: Jim Belak
Lawrence Livermore National Laboratory

Discussion of programmability and usability: What are the emerging themes/lessons learned regarding X-stack and other programming models evaluated in the context of our applications? What should X-stack 2.0 look like to be useful to real-world application developers? What is the role and future of DSLs?

10:30 Open Discussion..Moderator: Allen McPherson
Los Alamos National Laboratory

Tools and techniques for performance, power, and resilience (P^2R) optimization.

ExMatEx