LLNL-PRES-501654

FASTMath: An overview of mathematical algorithms and software

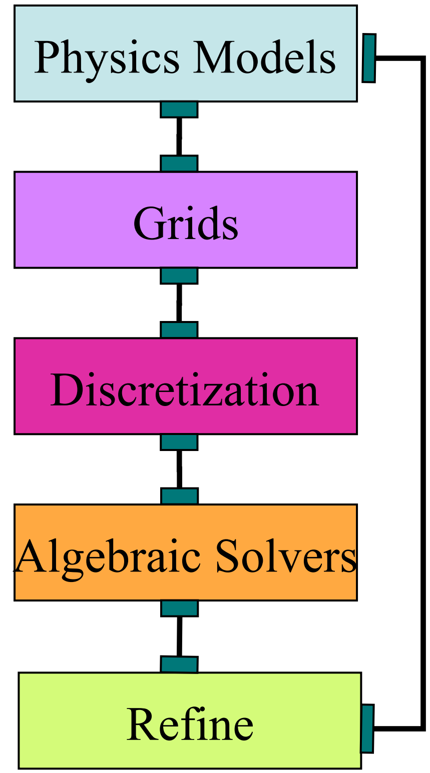
FASTMath Team

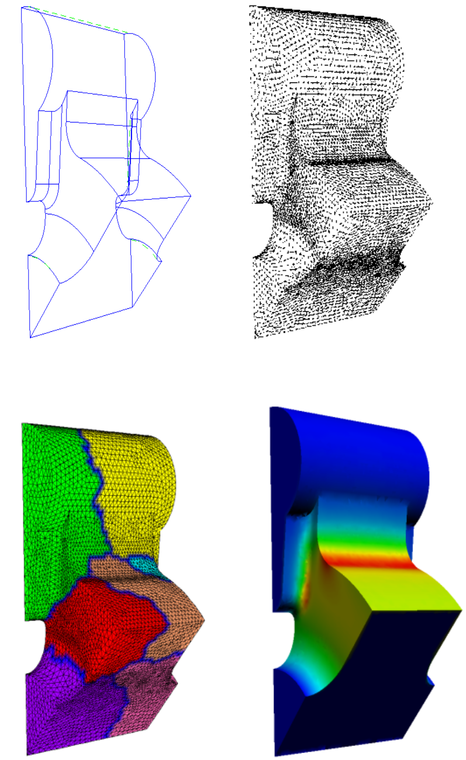
Lori Diachin, Institute Director

FASTMath SciDAC Institute

Mathematical algorithms and software are foundational to HPC simulations

- Develop a mathematical model of the phenomenon of interest
- Approximate the model using a discrete representation
- Solve the discrete representation
- Adapt and refine the mesh or model
- Couple different physics, scales, regions together





These steps require: CAD models, grid generation, high order discretizations, time integration techniques, linear and nonlinear solution of algebraic systems, eigensolvers, mesh refinement strategies, physics coupling methods, particle techniques, etc…

First consider a very simple example

- 1D rod with one end in a hot water bath, the other in a cold water bath
- Mathematical Model

$$\nabla^2 T = 0 \in \Omega$$

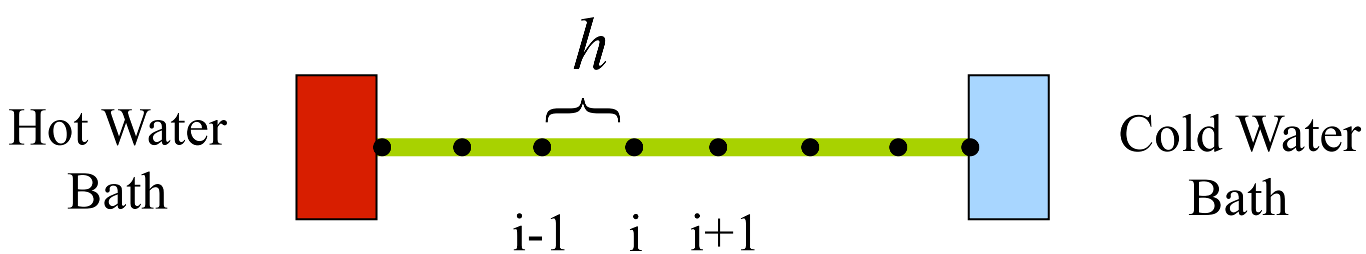
$$T(0) = 180^o \quad T(1) = 0^o$$

Discretizating the equations

- Approximate the derivatives in the continuous equations with a discrete representation that is easier to solve
- One approach: Finite Differences

$$\nabla^2 T \approx (T_{i+1} - 2T_i + T_{i-1})/2h = 0$$

$$T_0 = 180^o \qquad T_n = 0^o$$



Solve for the unknowns T_i

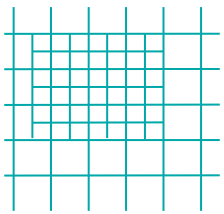
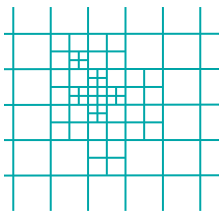
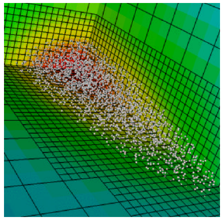
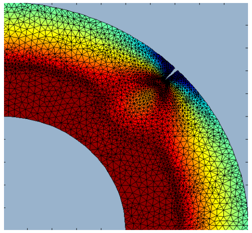
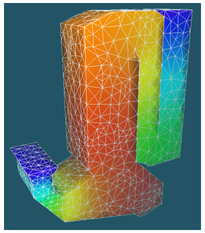
- Set up a matrix of the unknown coefficients
  - include the known boundary conditions
- Solve the linear system for T_i

$$\begin{pmatrix} 2 & -1 & 0 & \ldots\ldots\ldots & 0 \\ -1 & 2 & -1 & 0\ldots\ldots & 0 \\ 0 & -1 & 2 & -1 & 0\ldots.0 \\ & & \ldots\ldots\ldots & & \\ 0\ldots\ldots\ldots 0 & & & -1 & 2 \end{pmatrix} \begin{pmatrix} T_1 \\ T_2 \\ T_3 \\ . \\ T_{n-1} \end{pmatrix} = \begin{pmatrix} 180 \\ 0 \\ 0 \\ . \\ 0 \end{pmatrix}$$

- Visualize and analyze the results

As problems get more complicated so do the steps in the process

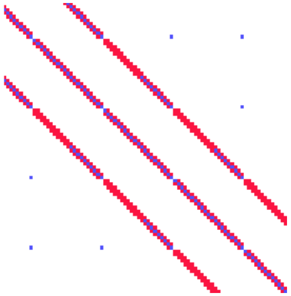
- Different discretization strategies exist for differing needs
  - Efficiency
  - Flexibility

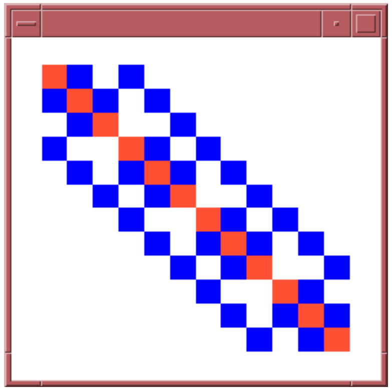






- Most problems are time dependent and nonlinear
  - Need higher algorithmic levels than linear solvers
- Increasingly combining multiple physical processes
  - Interactions require careful handling
- Goal-oriented problem solving required optimization, uncertainty quantification

As problems grow in size so do the corresponding discrete systems

- Targeting applications with billions grid points and unknowns
- Most linear systems resulting from these techniques are LARGE and sparse
- Often most expensive solution step
- Solvers:
  - Direct Methods (e.g. Gaussian Elimination)
  - Iterative Methods (e.g. Krylov Methods)
    - Preconditioning is typically critical
    - Mesh quality affects convergence rate
- Many software tools developed at DOE labs deliver this functionality as numerical libraries
  - PETSc, AZTEC, Hypre, SuperLU, etc.

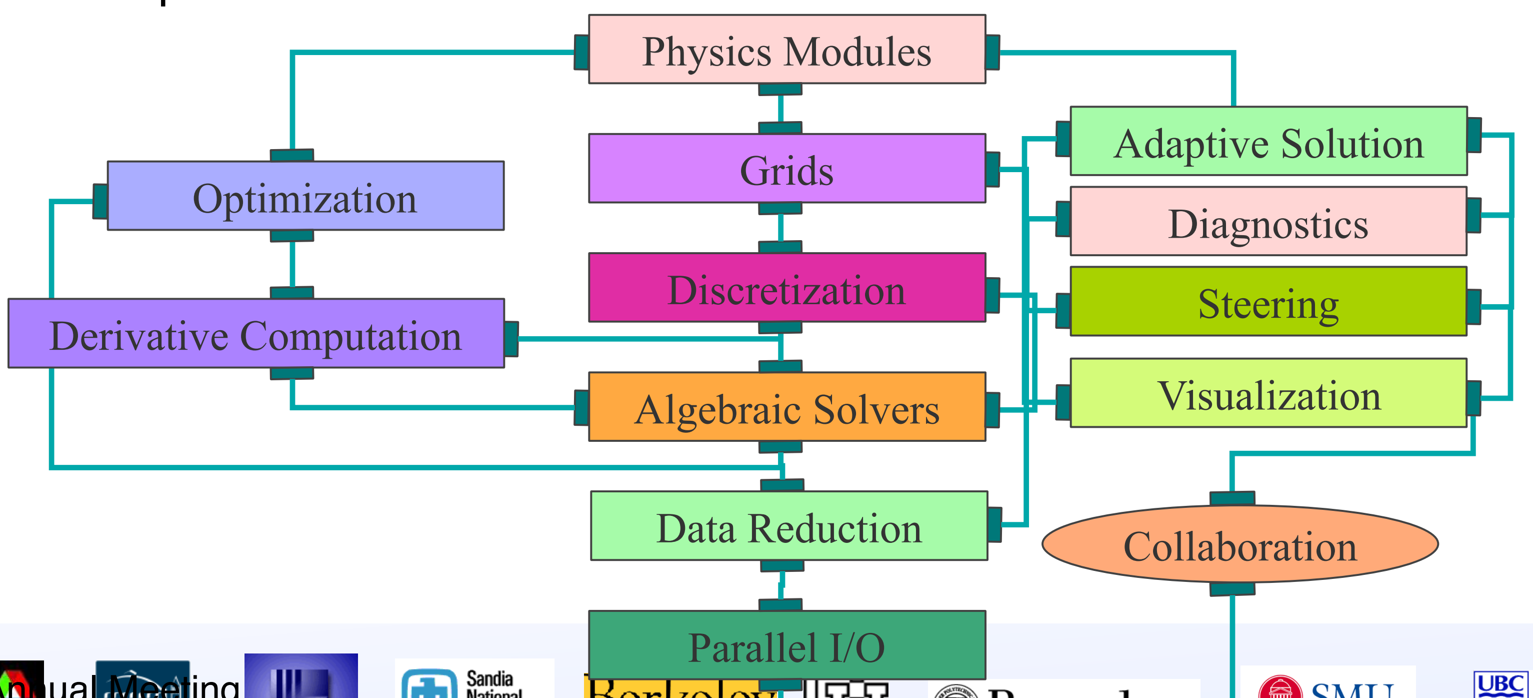




Modern scientific application development involves many different tools, libraries, and technologies

Observation: Exascale computing will enable high-fidelity calculations based on multiple coupled physical processes and multiple physical scales

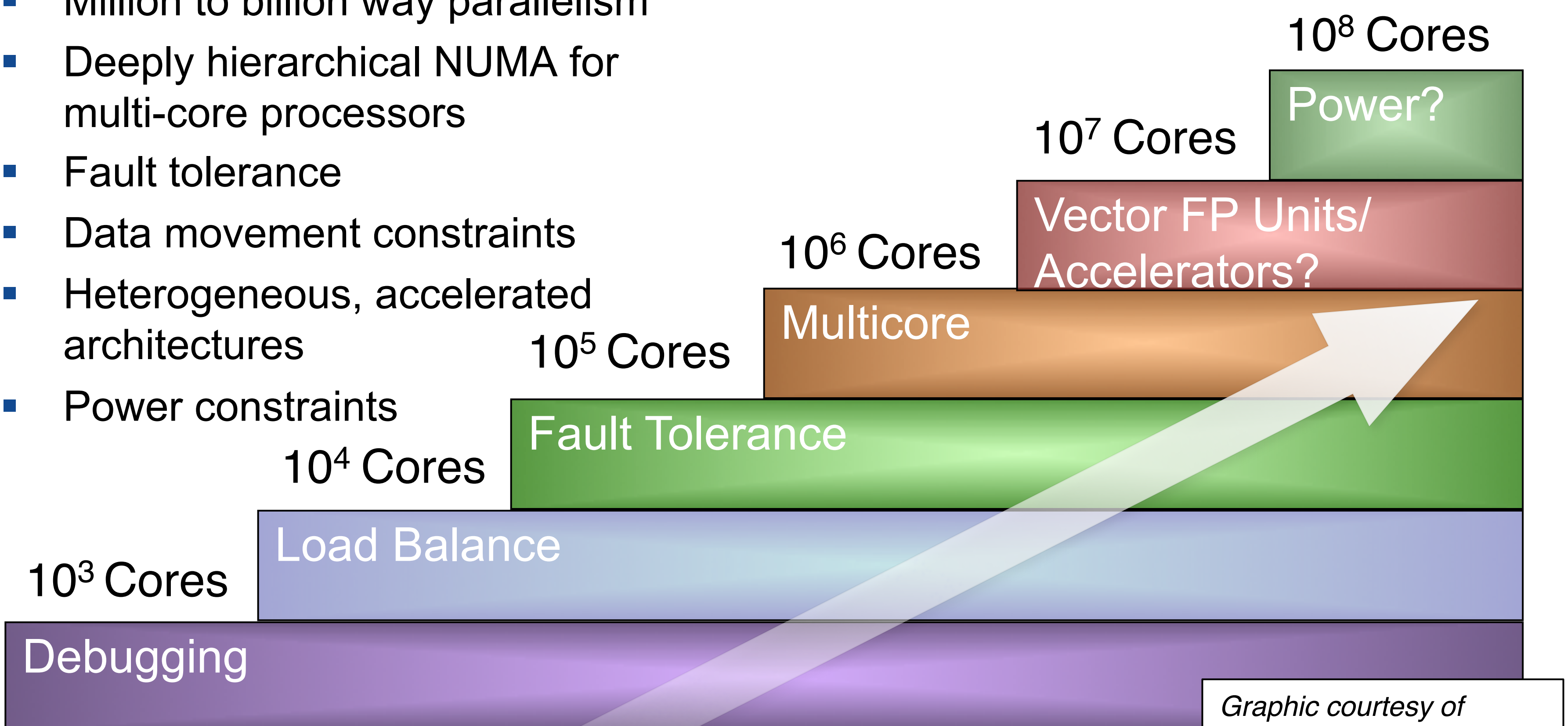
- Adaptive algorithms
- Composite or hybrid solution strategies
- High-order discretization strategies
- Sophisticated numerical tools



Modeling and simulation is significantly complicated by the change in computing architectures

Scientific computing software must address ever increasing challenges:

- Million to billion way parallelism
- Deeply hierarchical NUMA for multi-core processors
- Fault tolerance
- Data movement constraints
- Heterogeneous, accelerated architectures
- Power constraints



Graphic courtesy of Bronis de Supinski, LLNL

Argonne National Laboratory · Berkeley Lab · Sandia National Laboratories · Berkeley University of California · Rensselaer · SMU · UBC The University of British Columbia

These complexities results in common challenges facing application scientists

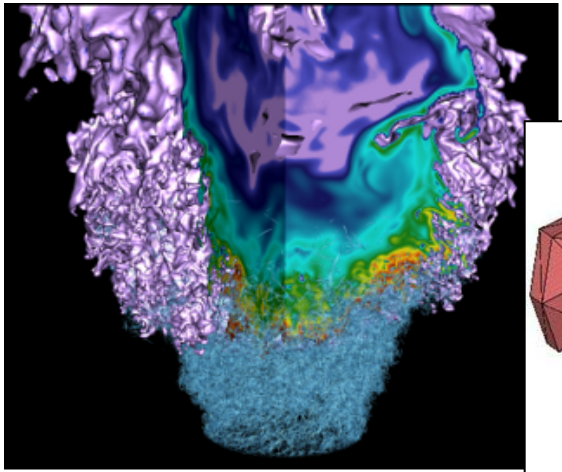
- Reliability
  - Accurate, stable discretizations
  - Robust solution algorithms
  - Error minimization
- Software Complexity
  - Interoperating numerical software
  - New algorithms (e.g., interactive/ dynamic techniques, algorithm composition)
  - New programming models
- Performance
  - Load balancing (perhaps dynamic)
  - Portability across architectures
  - Massive scale

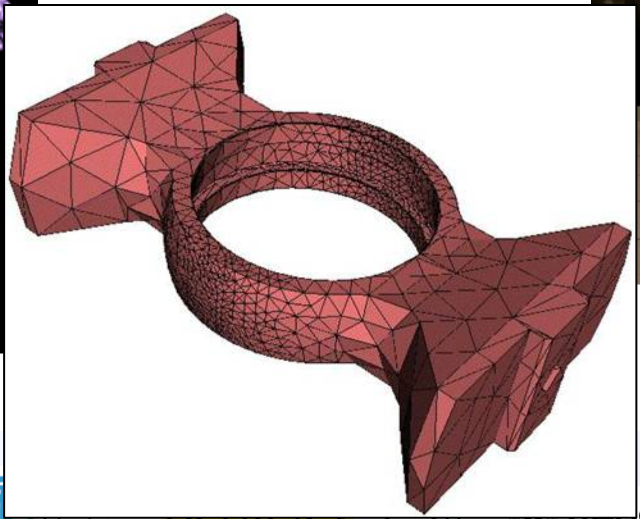
Application life cycle costs are increasing
- Require the combined use of software developed by different groups
- Difficult to leverage expert knowledge and advances in subfields
- Difficult to obtain portable performance

Too much energy focused on too many details
- Little time to think about modeling, physics, mathematics
- Fear of bad performance without custom code
- Even when code reuse is possible, it is far too difficult

The FASTMath SciDAC project addresses these challenges for mathematics software libraries

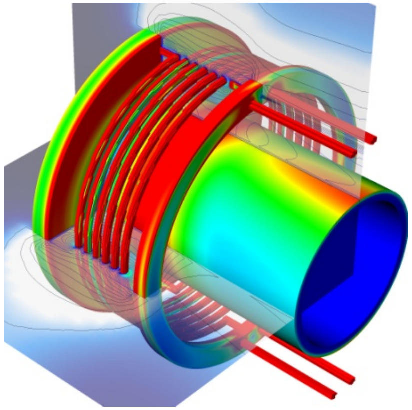
The FASTMath SciDAC Institute develops and deploys scalable mathematical algorithms and software tools for reliable simulation of complex physical phenomena and collaborates with DOE domain scientists to ensure the usefulness and applicability of FASTMath technologies





FASTMath is helping application scientists overcome two fundamental challenges

1. Improve the quality of their simulations
   - Increase accuracy
   - Increase physical fidelity
   - Improve robustness and reliability

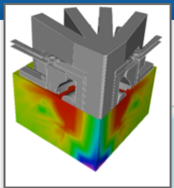


2. Adapt computations to make effective use of LCFs
   - Million way parallelism
   - Multi-/many-core nodes

FASTMath will help address both challenges by focusing on the interactions among mathematical algorithms, software design, and computer architectures

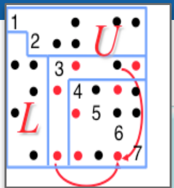
FASTMATH

FASTMath encompasses three broad topical areas



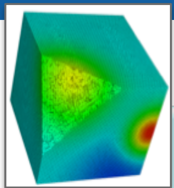
Tools for Problem Discretization

- Structured grid technologies
- Unstructured grid technologies
- Adaptive mesh refinement
- Complex geometry
- High-order discretizations
- Particle methods
- Time integration



Solution of Algebraic Systems

- Iterative solution of linear systems
- Direct solution of linear systems
- Nonlinear systems
- Eigensystems
- Differential variational inequalities



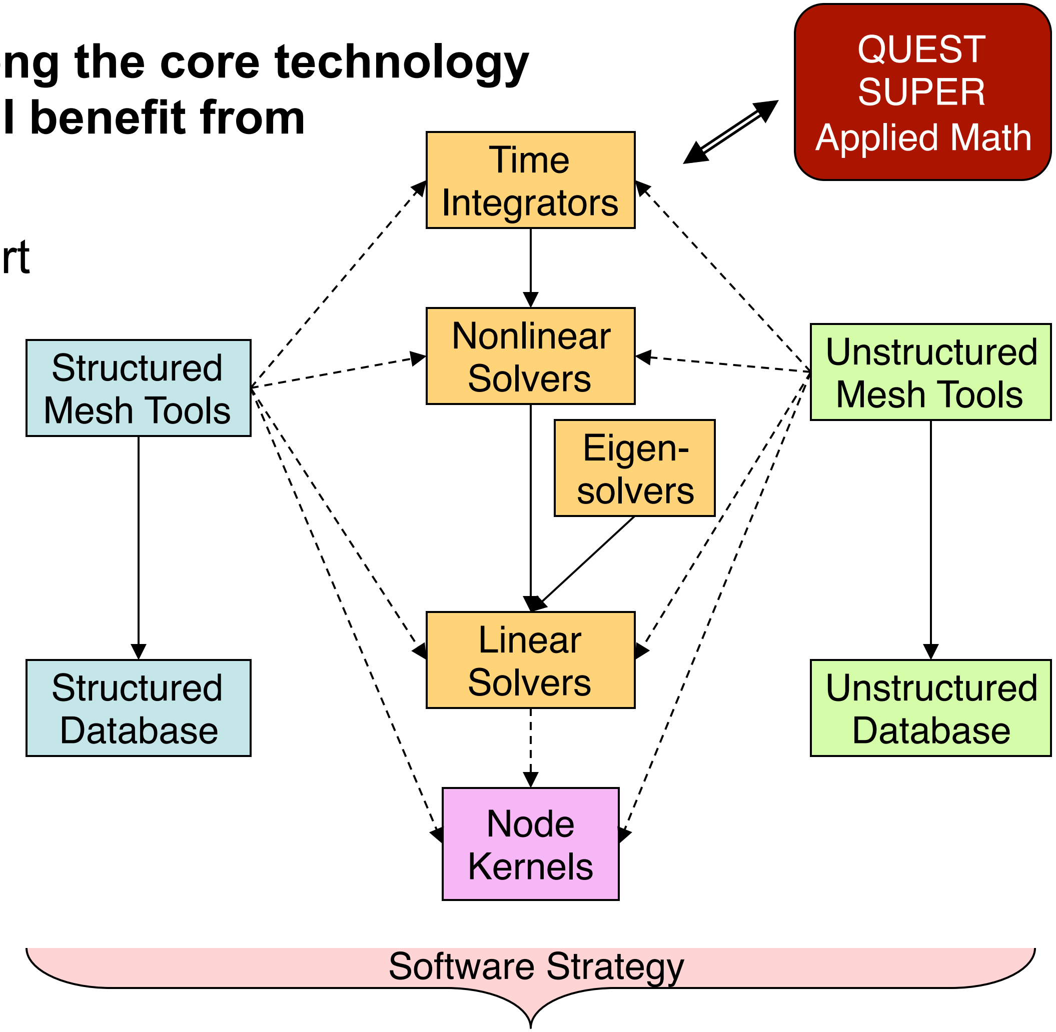
High Level Integrated Capabilities

- Adaptivity through the software stack
- Management of field data
- Coupling difference physics domains
- Mesh/particle coupling methods

Argonne National Laboratory · Berkeley Lab · Sandia National Laboratories · Berkeley University of California · Rensselaer · SMU · UBC The University of British Columbia

FASTMath provides a unique opportunity to develop integrated capabilities

- **As we provide integration among the core technology areas, science applications will benefit from**
  - Expanded capabilities
  - Decreased development effort
- **Development efforts for expanded capability integration**:
  - Adaptivity through the software stack
  - Field data and manipulation
  - Coupling strategies for multi-physics applications
  - Architecture-aware compute node kernels
  - Unified software strategy



Argonne NATIONAL LABORATORY · BERKELEY LAB · Sandia National Laboratories · Berkeley University of California · Rensselaer · SMU · UBC THE UNIVERSITY OF BRITISH COLUMBIA

FASTMath brings a spectrum of software tools in these areas to the SciDAC Program

FASTMath Toolset

Structured Mesh Tools
- **BoxLib** (Ann Almgren)
- **Chombo** (Phil Colella)

Geometry Tools
- CGM (Tim Tautges)

Linear Solvers
- **Hypre** (Rob Falgout)
- **PETSc** (Barry Smith)
- SuperLU (Sherry Li)
- ML/Trilinos (Jonathan Hu)

Unstructured Mesh Tools
- **MeshAdapt** (Mark Shephard)
- **MOAB** (Tim Tautges)
- Mesquite (Lori Diachin)
- PHASTA (Ken Jansen)

Nonlinear Solvers/Differential Variational Inequalities
- **SUNDIALS** (Carol Woodward
- **PETSc** (Barry Smith)
- NOX/Trilinos (Andy Salinger)

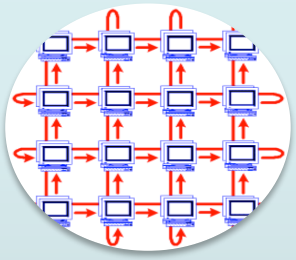
Partitioning Tools
- **Zoltan** (Karen Devine)
- **ParMA** (Mark Shephard)

Eigensolvers
- PARPACK (Chao Yang)

Time Integrators
- **SUNDIALS** (Carol Woodward)
- **PETSc** (Barry Smith)

All FASTMath technologies focus on performance engineering for multi-/many-core architectures

MPI Level Parallelism



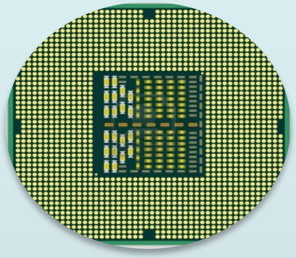
Operate efficiently at 10^5 to 10^6 cores

Architecture-aware and multi-objective load balancing

Communication avoiding and latency tolerant algorithms

Synchronization reducing algorithms

Node Level Parallelism

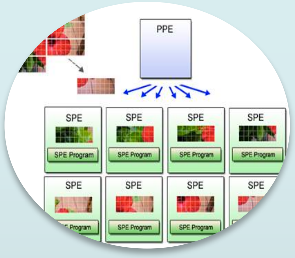


Use of threading techniques

Multi-core kernels and data ordering

Exploit compilers, code transformation tools, programming models and run-time systems as they become available

Data Locality



Hierarchical partitioning and local data ordering methods

Shared efficient data layouts in software packages to prevent re-organization

Code transformation systems, domain specific language extensions to gain performance while maintaining reusability

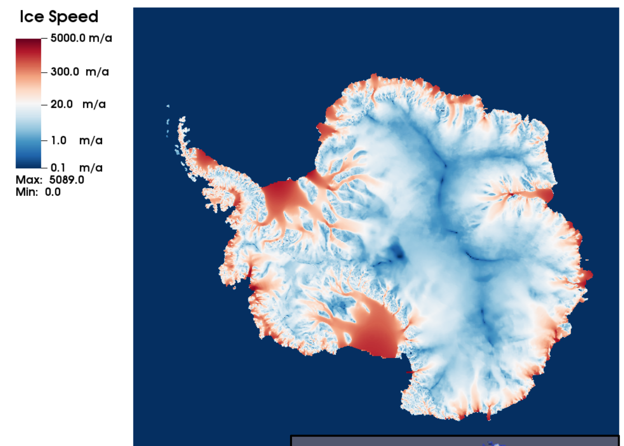
Coordinated parallelism between different levels (MPI, node, instruction)

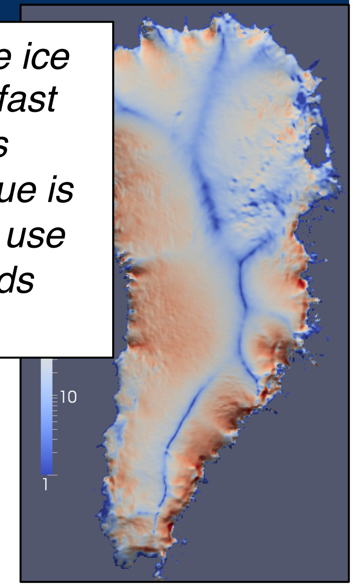
FASTMath technologies are used in many DOE application partnerships (examples)

Title	PI	FASTMath involvement	Office	Technology
Predicting Ice Sheet and Climate Evolution at Extreme Scales (PISCEES)	William Lipscomb, LANL	Esmond Ng, Andy Salinger	BER	Algebraic
Multiscale Methods for Accurate, Ecient, and Scale-Aware Models of the Earth System	William Collins, LBNL	Carol Woodward	BER	Algebraic
Applying Computationally Efficient Schemes for BioGeochemical Cycles	Forrest Hoffman, ORNL	Tim Tautges	BER	Discretization
Charge Transfer and Charge Transport in Photoactivated Systems	Chris Cramer, Minnesota	Esmond Ng, Chao Yang	BES	Algebraic
Simulating the generation, evolution and fate of electronic excitations in molecular and nanoscale materials with first principles methods.	Martin Head-Gordon	Sherry Li, Esmond Ng, Chao Yang	BES	Algebraic
Advanced Modeling of Ions in Solutions, on Surfaces, and in Biological Environments	Roberto Car, Princeton	Esmond Ng, Chao Yang	BES	Algebraic
Scalable Computational Tools for Discovery and Design -- Excited State Phenomena in Energy Materials	James Chelikowsky, UT Austin	Chao Yang	BES	Algebraic
Discontinuous methods for accurate, massively parallel quantum molecular dynamics: Lithium ion interface dynamics from first principles	John Pask, LLNL	Chao Yang	BES	Algebraic
Optimizing Superconductor Transport Properties through Large-scale Simulation	Andreas Glatz, ANL	Todd Munson	BES	Algebraic
Plasma Surface Interactions: Bridging from the Surface to the Micron Frontier through Leadership Class Computing	Brian Wirth, ORNL	Tim Tautges, Emil Constantinescu, Barry Smith	FES	Discretization, Algebraic
Center for Edge Physics Simulation	CS Chang, PPPL	Mark Adams, Mark Shephard	FES	Discretization
Searching for Physics Beyond the Standard Model: Strongly-Coupled Field Theories at the Intensity and Energy Frontiers	Paul Mackenize, Fermilab	Rob Falgout	HEP	Algebraic
Computation-Driven Discovery for the Dark Universe	Salmon Habib, ANL	Ann Almgren	HEP	Discretization
ComPASS: High performance Computing for Accelerator Design and Optimization	Panaggiotis Spentzouris, Fermilab	Esmond Ng, Phil Colella, Sherry Li, Todd Munson, Chao Yang	HEP	Discretization, Algebraic
A MultiScale Approach to Nuclear Structure and Reactions: Forming the Computational Bridge between Lattice QCD and Nonrelativistic Many-Body Theory (\CalLAT")	Wick Haxton, LBNL	Esmong Ng, Rob Falgout	NP	Algebraic
Nuclear Computational Low-energy Initiative	Joseph Carlson, LANL	Esmond Ng, Chao Yang	NP	Algebraic
ParaDiS Dislocation Dynamics	Tom Arsenlis, LLNL	Carol Woodward	NNSA	Algebraic

PISCEES Ice sheet application team focuses on modeling evolving land ice in global climate models

- **Project Goal:** to develop predictive capability for ice sheet evolution:
  - Future **Land Ice** component of global earth system models
  - Focus primarily on Greenland and Antarctic ice sheets
- Developing a new dynamical core that contains the following new technologies:
  - Adaptive, refined meshes to focus resolution (Chombo, Trilinos)
  - Improved nonlinear solvers for steady and quasi steady state solves using homotopy and FAS multigrid (NOX, Chombo/PETSc)
  - Improved linear solver preconditioners using multigrid (MLU, hypre, PETSc)





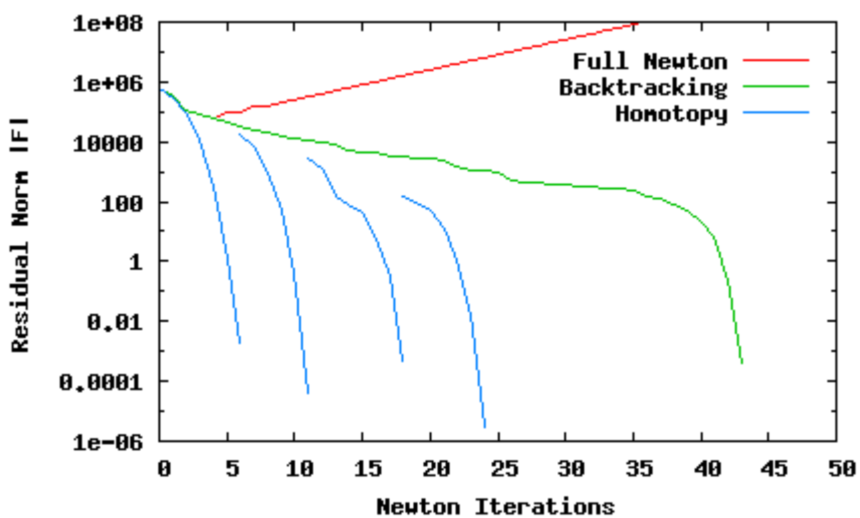
The colors are ice flow – red is fast and needs refinement, blue is slow and can use coarser grids

FASTMath collaboration enabled a fundamentally new capability be developed quickly

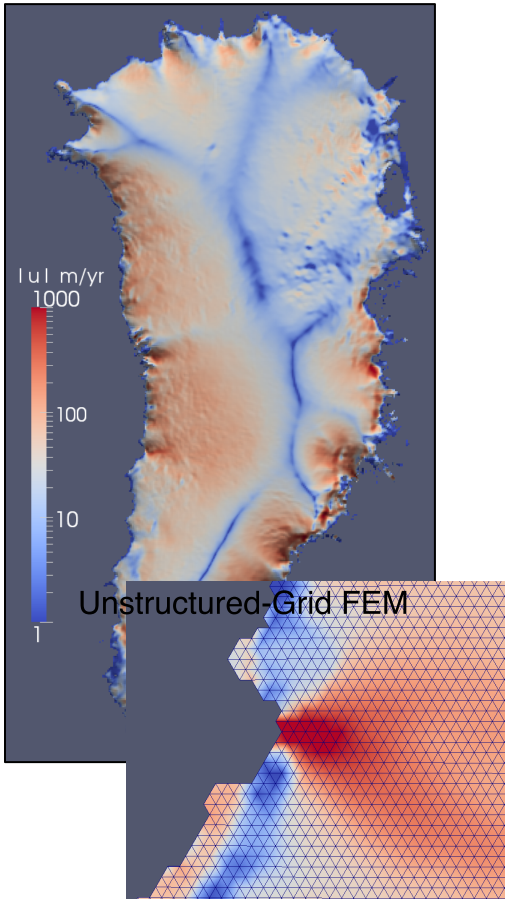
Parallel, 3D Unstructured-grid FEM, Implicit, Robust, Verified, Tested, PDE code written in less then 1 year:

- Greenland & Antarctic simulations
- Scalable Multi-level linear solves (Trilinos/ML)
  - Model problem with 182M unknowns on 4096 cores
  - 2km Greenland grid, steady-state solve, requires less then 1 minute on 9600 cores
- Robust nonlinear solves using homotopy (Trilinos/NOX)

❖ Linked to Dakota for UQ and Calibration



[Salinger, Kalashnikova, Perego, Tuminaro, Worley]



Greenland Ice Sheet Surface Velocities
(constant friction model)

The FASTMath team includes experts from four national laboratories and five universities

Lawrence Berkeley National Laboratory
- Mark Adams
- Ann Almgren
- John Bell
- Phil Colella
- Anshu Dubey
- Dan Graves
- Sherry Li
- Terry Ligocki
- Mike Lijewski
- Peter McCorquodale
- Esmond Ng
- Brian Van Straalen
- Chao Yang

Berkeley University
- Jim Demmel

Lawrence Livermore National Laboratory
- Lori Diachin
- Milo Dorr
- Rob Falgout
- Jeff Hittinger
- Mark Miller
- Carol Woodward
- Ulrike Yang

Sandia National Laboratories
- Karen Devine
- Jonathan Hu
- Vitus Leung
- Andrew Salinger

University of British Columbia
- Carl Ollivier-Gooch

Argonne National Laboratory
- Mihai Anitescu
- Lois Curfman McInnes
- Todd Munson
- Barry Smith
- Tim Tautges
- Jungho Lee

Rensselear Polytechnic Institute
- Mark Shephard
- Onkar Sahni

Southern Methodist University
- Dan Reynolds

Colorado University at Boulder
- Ken Jansen

FASTMath Tutorial Goals

- Provide a basic understanding of a variety of applied mathematics algorithms for scalable linear, nonlinear, and ODE solvers as well as discretization technologies (e.g., adaptive mesh refinement for structured and unstructured grids)
- Provide an overview of FASTMath software tools available to perform these tasks on HPC architectures
- Practice using one or more of these software tools on basic demonstration problems

FASTMath Tutorial Schedule: Friday, August 2

- 2:00 ***FASTMath: An Overview of Mathematical Algorithms and Software***
  - Lori Diachin (LLNL)
- 2:30 ***Algebraic Solvers in FASTMath***
  - Barry Smith (ANL)
- 3:20 Break
- 3:50 ***HYPRE: High Performance Preconditioners***
  - Rob Falgout (LLNL)
- 4:15 ***SUNDIALS: Suite of Nonlinear and Differential/Algebraic Equation Solvers***
  - Carol Woodward (LLNL)
- 4:40 ***Unstructured Mesh Handling for Extreme-Scale Computing***
  - Tim Tautges (ANL)
- 5:30 Dinner and ***Teaming in the DOE Environment***
  - Lori Diachin (LLNL)
- 6:30 ***FASTMath Hands-on Exercises***
  - Mark Miller (LLNL) and the FASTMath team

FASTMATH

FASTMath Tutorial Schedule: Saturday, August 3

- 7:30 Continental Breakfast
- 8:30 ***Block Structured AMR Libraries and Their Interoperability with Other Math Libraries***
  - Anshu Dubey and Mark Adams (LBNL)
- 9:30 ***Infrastructure for Parallel Adaptive Unstructured Mesh Simulations***
  - Mark Shephard and Cameron Smith (RPI), Glen Hansen (SNL)
- 10:20 Break
- 10:50 ***FASTMath Hands-on Exercises***
  - Mark Miller (LLNL) and the FASTMath team
- 12:00 Lunch and Hands-on Exercises

For more information, please contact any of the following or visit our web site

- FASTMath Institute Director:
  - Lori Diachin,
    diachin2@llnl.gov, 925-422-7130
- FASTMath Executive Council
  - Phil Colella, Structured Mesh Tools
    pcolella@lbl.gov, 510-486-5412
  - Esmond Ng, Nonlinear/Eigensolvers
    egng@lbl.gov, 510-495-2851
  - Andy Salinger, Integrated Technologies
    agsalin@sandia.gov, 505-845-3523
  - Mark Shephard, Unstructured Mesh Tools
    shephard@scorec.rpi.edu, 518-276-8044
  - Barry Smith, Linear Solvers,
    bsmith@mcs.anl.gov, 630-252-9174






http://www.fastmath-scidac.org

Auspices and Disclaimer

Support for this work was provided through Scientific Discovery through Advanced Computing (SciDAC) program funded by U.S. Department of Energy, Office of Science, Advanced Scientific Computing Research

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