AIAA Aerosciences Session 160-FD-31:

Role of High Performance Computing in Aerospace Engineering

SciDAC: transforming an HPC community

David Keyes

Dept. of Applied Physics & Applied Mathematics,

Columbia University

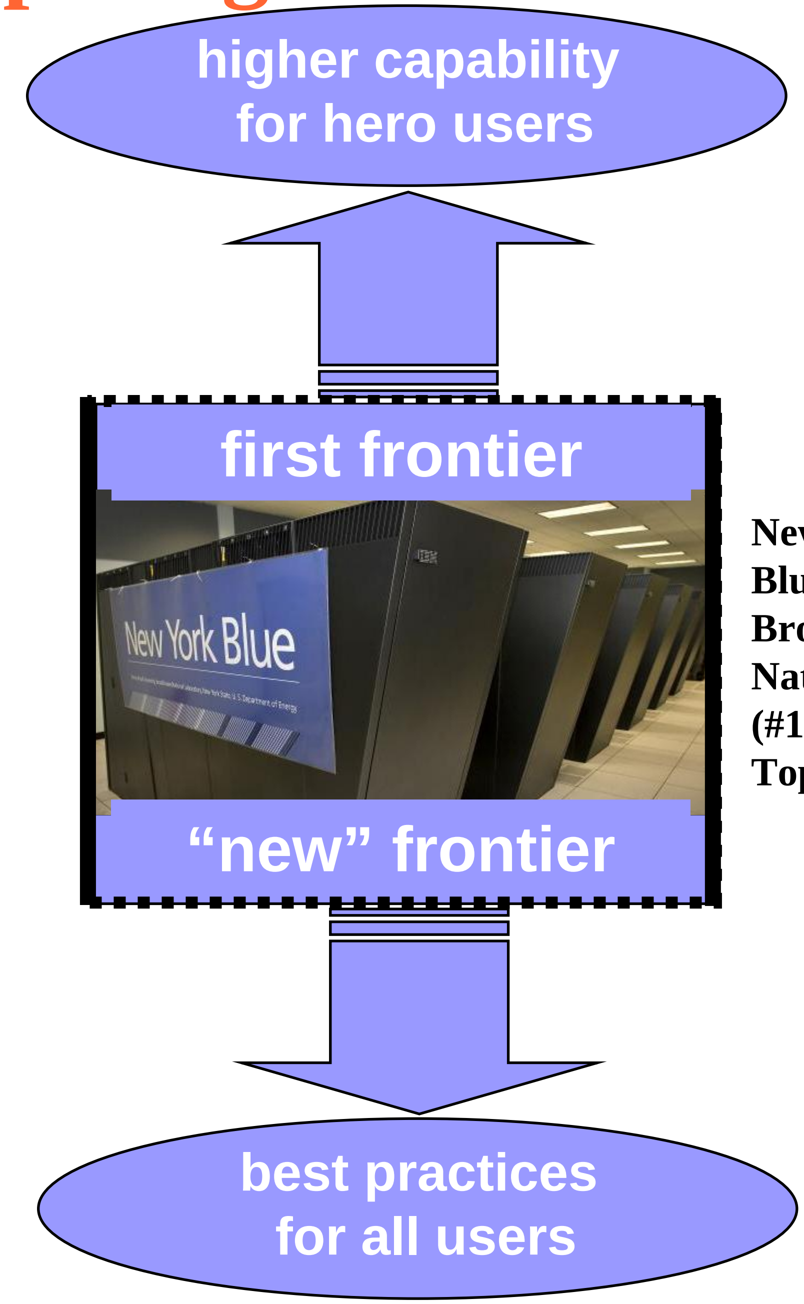
&

Institute for Scientific Computing Research (ISCR)

Lawrence Livermore National Laboratory

High-performance computing: two frontiers

- Two frontiers
  - raise the peak capability for simulation experts
  - lower the HPC simulation entry threshold for people who are expert in something else
- Historically, rewards and attention go to the former
- We describe a cross-cutting effort, DOE's Scientific Discover through Advanced Computing (SciDAC) program that attempts the latter



higher capability
for hero users

first frontier

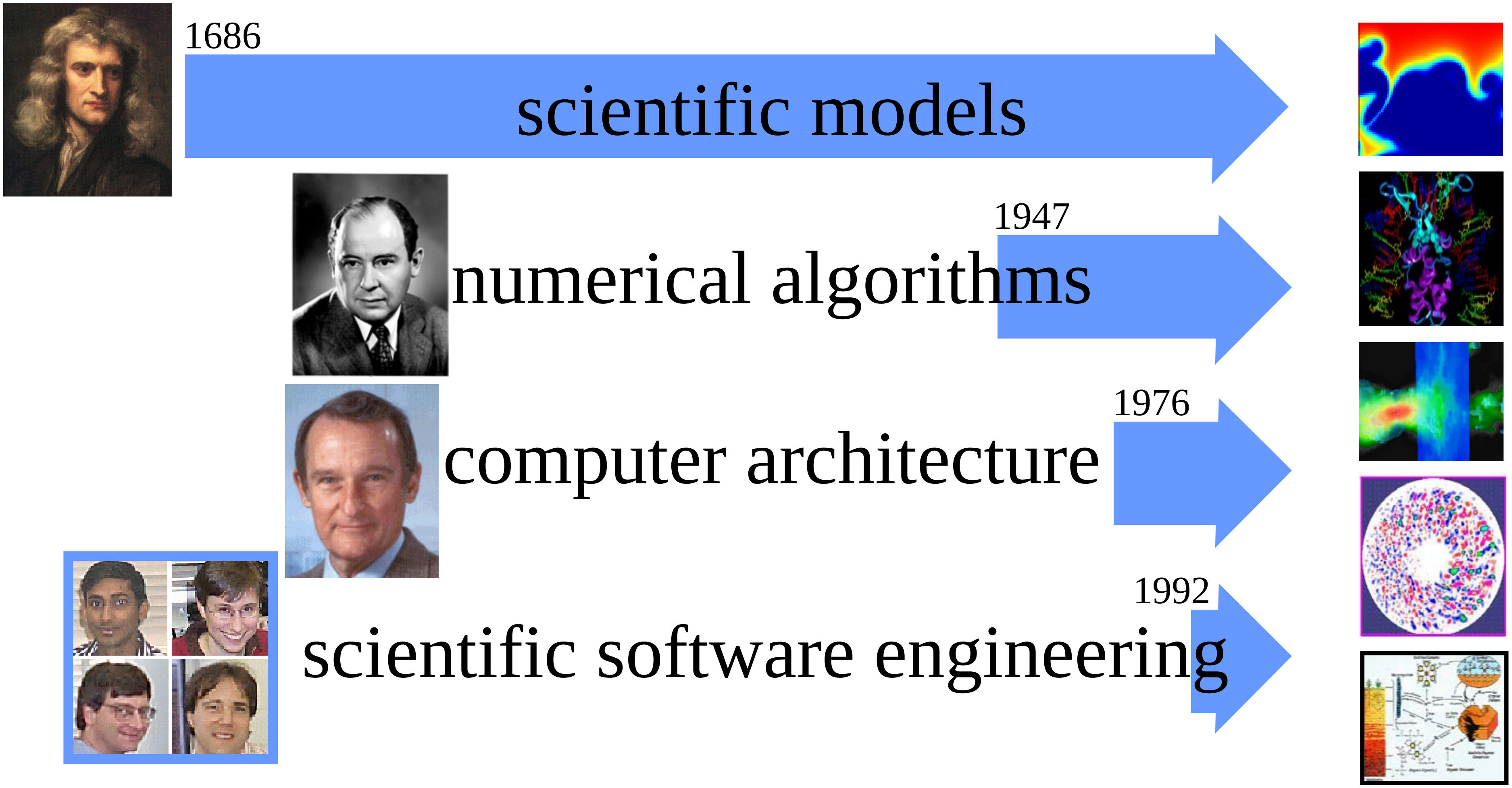
"new" frontier

best practices
for all users

New York Blue at Brookhaven National Lab (#10 on the Top 500)

Today: a perfect season for simulation

(dates are symbolic)



1686 scientific models

1947 numerical algorithms

1976 computer architecture

1992 scientific software engineering

“Computational science is undergoing a phase transition.” – D. Hitchcock, DOE

Some reports on the ascending role of simulation …

Fusion Simulation Project

June 2007

… for DOE and NSF

Hurdles to predictive (or “extrapolative”) simulation

- “Triple finiteness” of computers
  - **finite precision**
  - **finite number of words**
  - **finite processing rate**

Need: stability, optimality of representation & optimality of work

- Curse of dimensionality
  - **Moore’s Law is quickly “eaten up” in 3 space dimensions plus time**

Need adaptivity

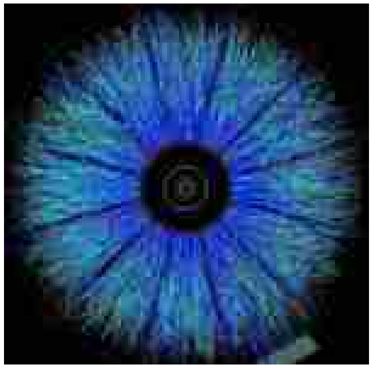
- Curse of uncertainty
  - **models and inputs are often poorly known**

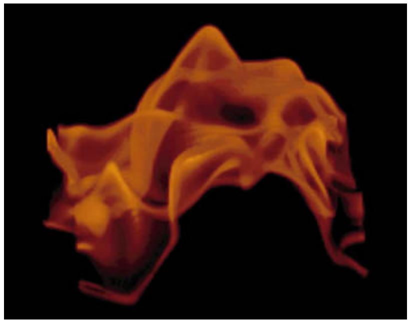
Need UQ methods

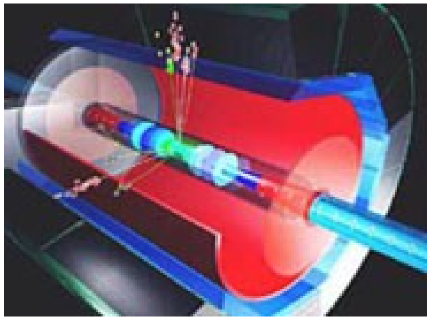
- Curse of knowledge explosion
  - **no one scientist can track all necessary developments**

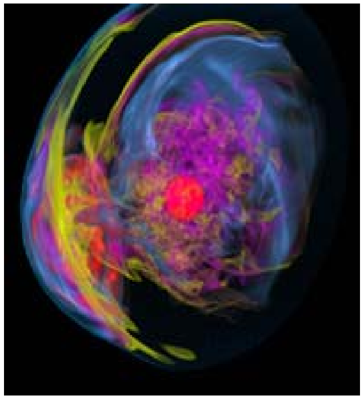
Need good colleagues ☺

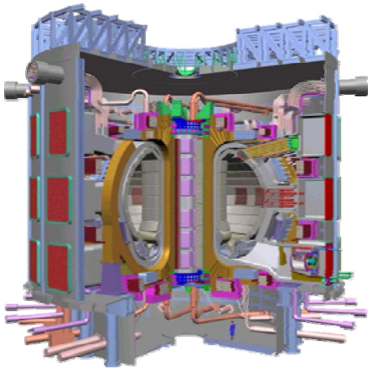
SciDAC: economy in general-purpose "ETs" for specialized "Apps"

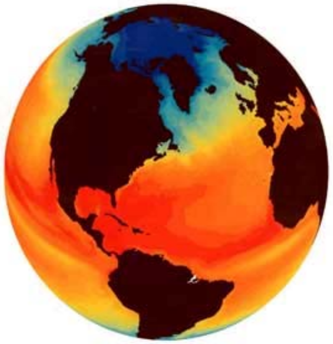




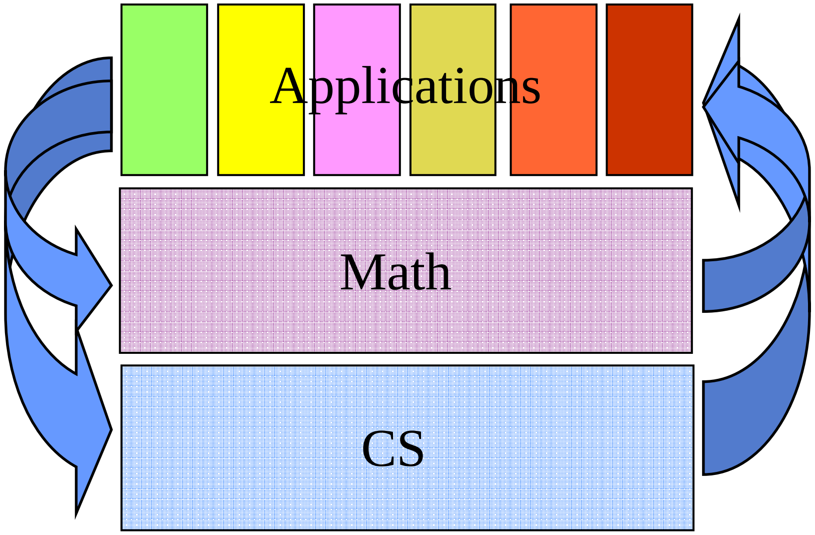








Many applications drive



Applications

Math

CS

Enabling technologies respond to all

Presentation plan

- Describe the structure of the SciDAC program
  - missing aerosciences component
  - worthy of emulation in aerosciences
- Look at one particular set of interdisciplinary interactions
  - magnetically confined plasma fusion – a multiscale, multiphysics app
  - has many challenges in common with aerosciences (fluids, control, etc.)

- Transition to discussion about "cultural" problems in interdisciplinary programs
  - aero industry and NASA face many of the same cultural problems in collaborating with academia

SciDAC

Scientific Discovery through Advanced Computing

- “Enabling technologies” groups to develop reusable software and partner with application groups
- In 2006 renewal, 49 projects share $60M/year, divided between
  - applications projects
  - lab-based Centers for Enabling Technology (CETs)
  - academic-hosted “institutes”
- Plus, petaflop/s-scale IBM BlueGene machines at Berkeley and Argonne, and Cray XT machine available at Oak Ridge for SciDAC researchers

Features of DOE's SciDAC initiative

- **Affirmation of importance of simulation**
  - for new scientific discovery, not just for "fitting" experiments
- **Recognition that leading-edge simulation is interdisciplinary**
  - physicists and chemists not supported to write their own software infrastructure; deliverables intertwined with those of math & CS experts
- **Commitment to distributed hierarchical memory computers**
  - new code must target this architecture type
  - With lookahead to multicore/manycore
- **Commitment to maintenance of software infrastructure**
- **Requirement of lab-university collaborations**
  - complementary strengths in simulation
  - 13 laboratories and 50 universities in first round of projects

SciDAC's
Applied Math Centers & Institutes

- **Interoperable Tools for Advanced Petascale Simulations (ITAPS)**

  addressing complex domain geometry

- **Algorithmic and Software Framework for Partial Differential Equations (APDEC)**

  addressing solution adaptivity

- **Combinatorial Scientific Computing and Petascale Simulation (CSCAPES)**

  addressing partitioning and ordering, dependence analysis

- **Towards Optimal Petascale Simulations (TOPS)**

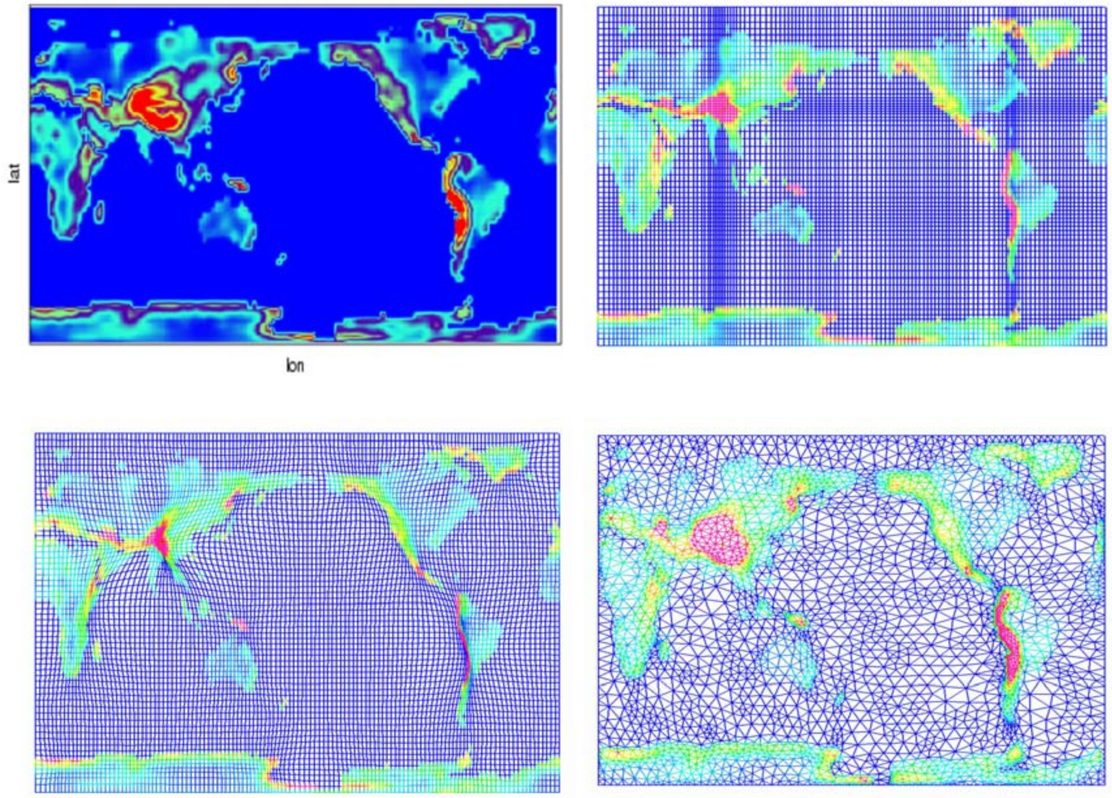
  addressing scalable solution

See: www.scidac.gov/math/math.html

ITAPS

Interoperable Tools for Advanced Petascale Simulations

Develop framework for use of multiple mesh and discretization strategies within a single PDE simulation. Focus on high-quality hybrid mesh generation for representing complex and evolving domains, high-order discretization techniques, and adaptive strategies for automatically optimizing a mesh to follow moving fronts or to capture important solution features.

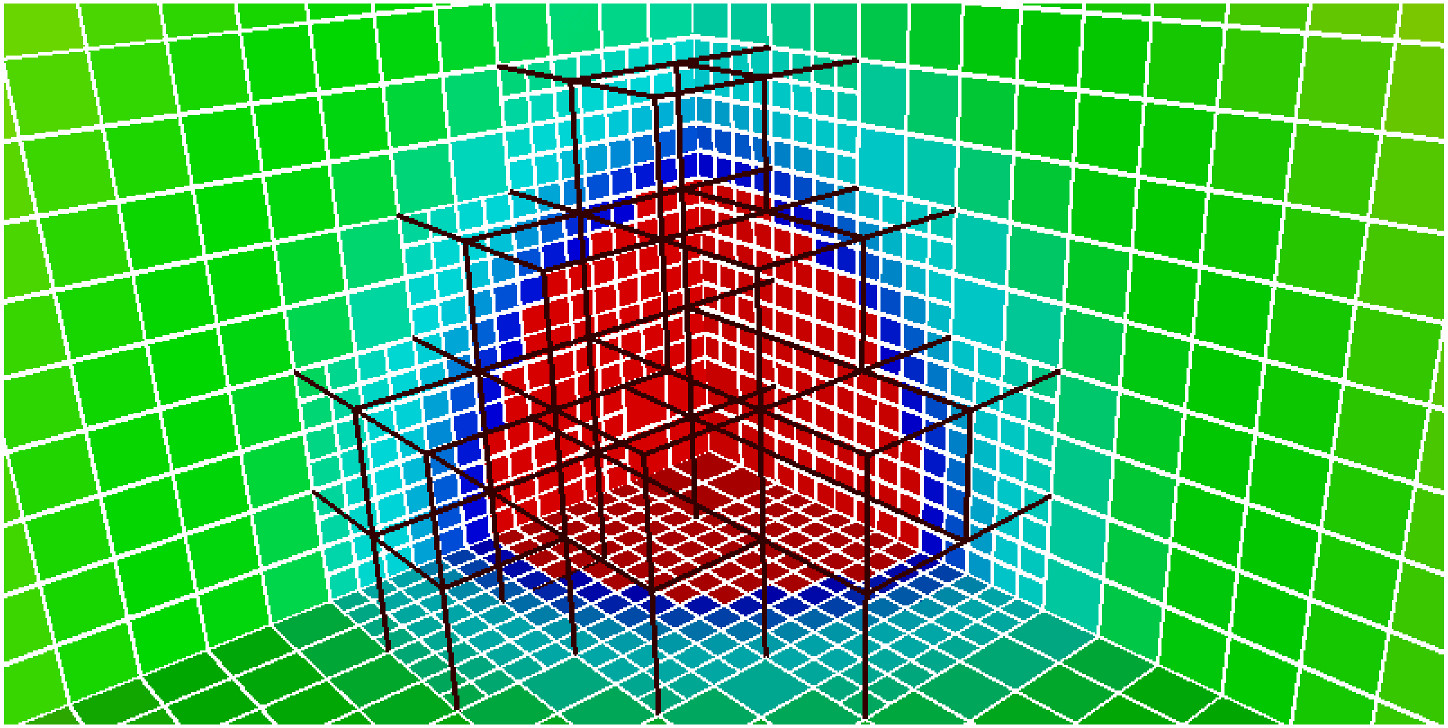


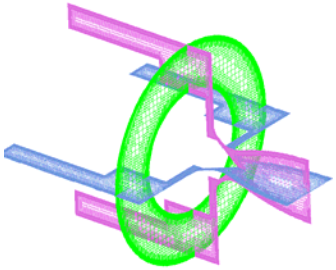
c/o L. Freitag, LLNL

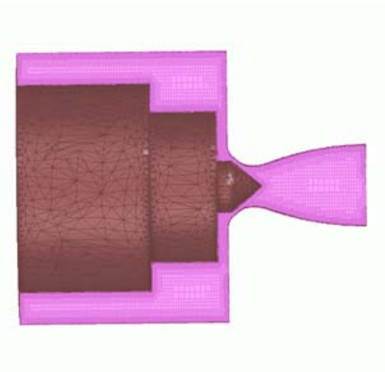
APDEC

Algorithmic and Software Framework for PDEs

Develop framework for PDE simulation based on locally structured grid methods, including adaptive meshes for problems with multiple length scales; embedded boundary and overset grid methods for complex geometries; efficient and accurate methods for particle and hybrid particle/mesh simulations.





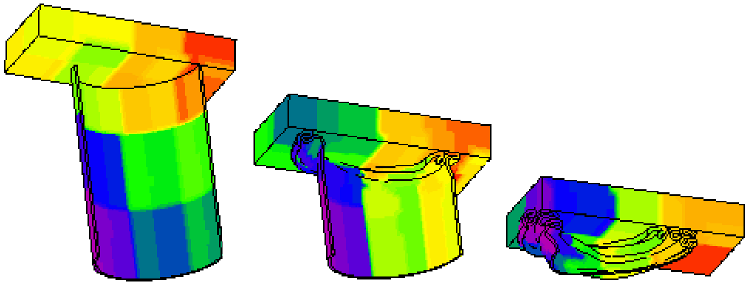


c/o P. Colella, LBNL

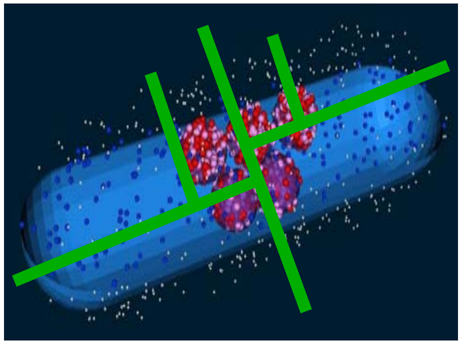
CSCAPES

Combinatorial Scientific Computing and Petascale Simulation

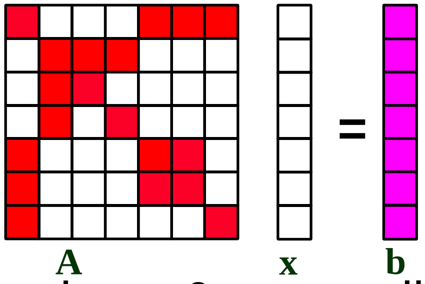
Develop toolkit of partitioners, dynamic load balancers, advanced sparse matrix reordering routines, and automatic differentiation procedures, generalizing currently available graph-based algorithms to hypergraphs



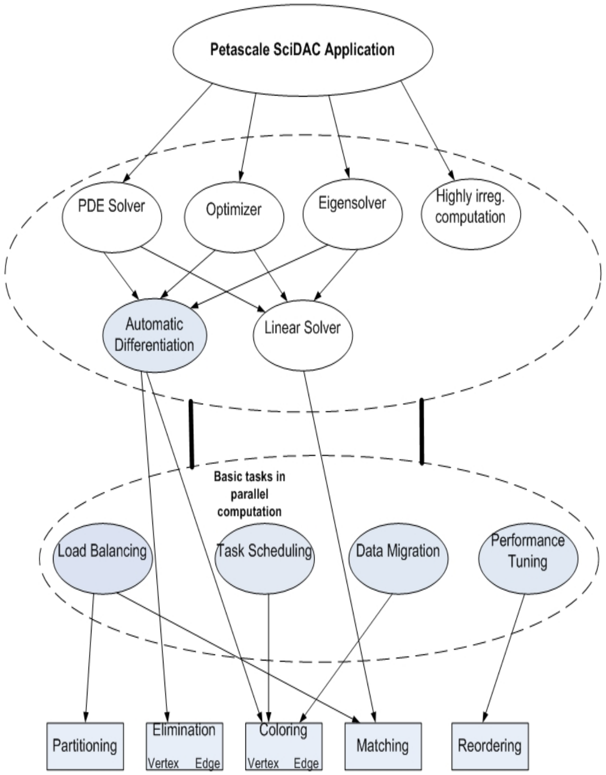
Contact detection



Particle Simulations



Linear solvers & preconditioners



Petascale SciDAC Application

PDE Solver

Optimizer

Eigensolver

Highly irreg. computation

Automatic Differentiation

Linear Solver

Basic tasks in parallel computation

Load Balancing

Task Scheduling

Data Migration

Performance Tuning

Partitioning

Elimination Vertex Edge

Coloring Vertex Edge

Matching

Reordering

c/o A. Pothen, ODU

The TOPS Center for Enabling Technology spans 4 labs & 5 universities

Our mission: Enable scientists and engineers to take full advantage of petascale hardware by overcoming the scalability bottlenecks traditional solvers impose, and assist them to move beyond “one-off” simulations to validation and optimization ($32M/10 years)

Argonne NATIONAL LABORATORY

BERKELEY LAB

TOPS

Towards Optimal Petascale Simulations

Lawrence Livermore
National Laboratory

Sandia National Laboratories

Columbia University

University of Colorado

Berkeley
University of California

The Institute for Computational Engineering and Sciences
ICES

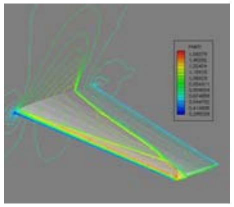
University of Texas

UCSD

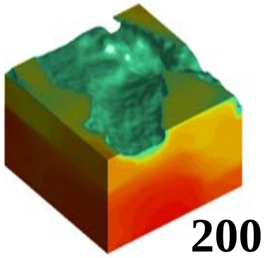
University of California
at San Diego

TOPS software has taken a variety of applications to the architectural edge

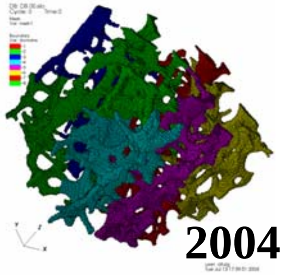
- TOPS is at the heart of three Gordon Bell “Special” Prizes



1999

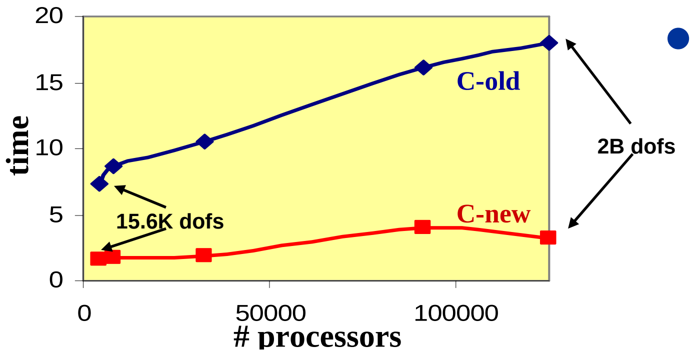


2003



2004

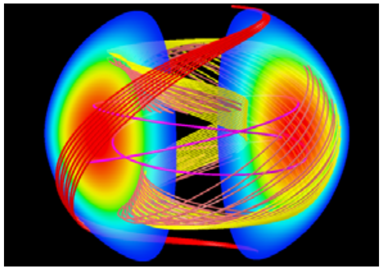
After new coarsening algorithm (red), nearly flat scaled speedup for Algebraic Multigrid



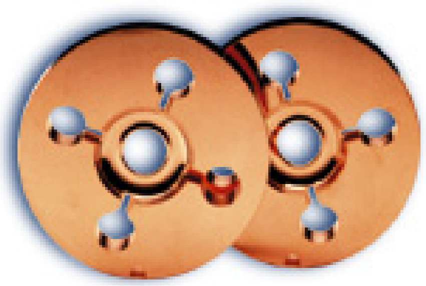
- Scales to the edge of BlueGene/L (131,072 processors, 2B unknowns)

- Powered numerous applications achievements in SciDAC-1

~5X speedup of plasma fusion code through linear solver replacement – like providing “next generation” computer

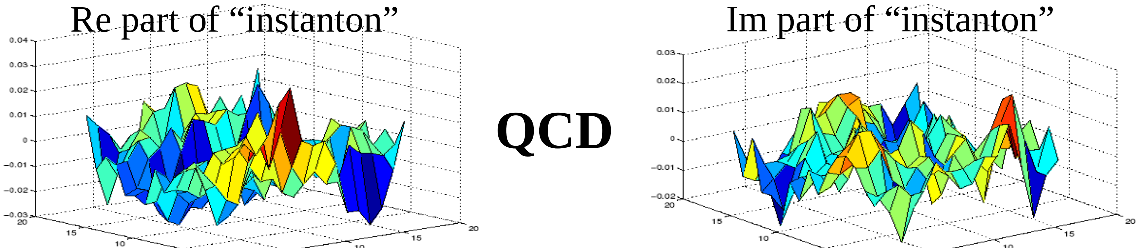


magneto-hydro-dynamics



accelerator design

Prototype shape optimization capability



Robust solution algorithm for zero quark mass, fine lattices

Toolchain for PDE solvers in TOPS project

- Design and implementation of “solvers”
  - Time integrators (w/ sens. anal.)

    $$f(\dot{x}, x, t, p) = 0$$

  - Nonlinear solvers (w/ sens. anal.)

    $$F(x, p) = 0$$

  - Constrained optimizers

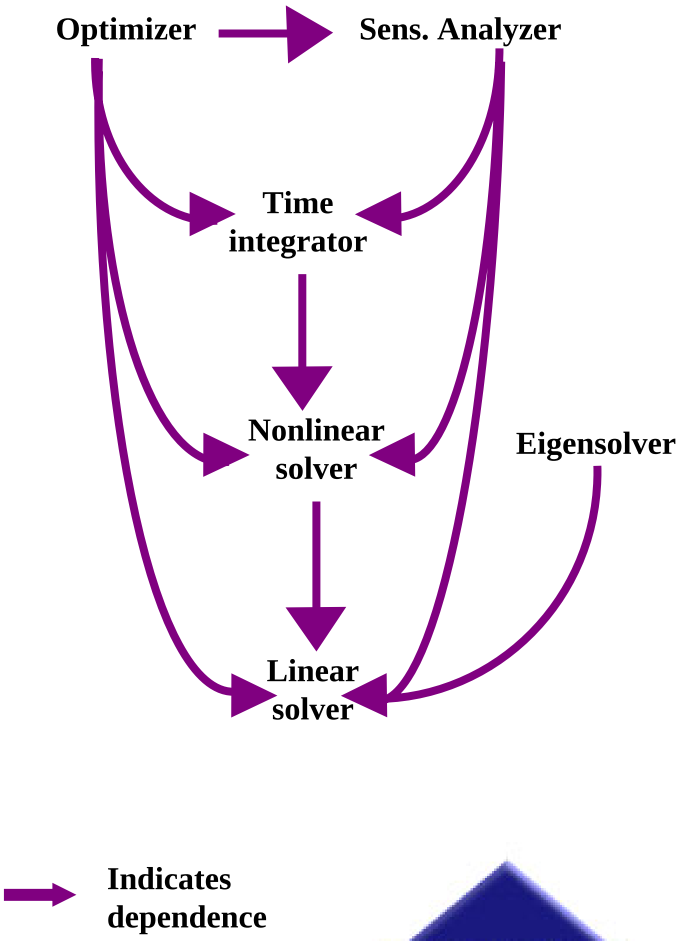
    $$\min_u \phi(x,u)\ s.t.\ F(x,u) = 0, u \geq 0$$

  - Linear solvers

    $$Ax = b$$

  - Eigensolvers

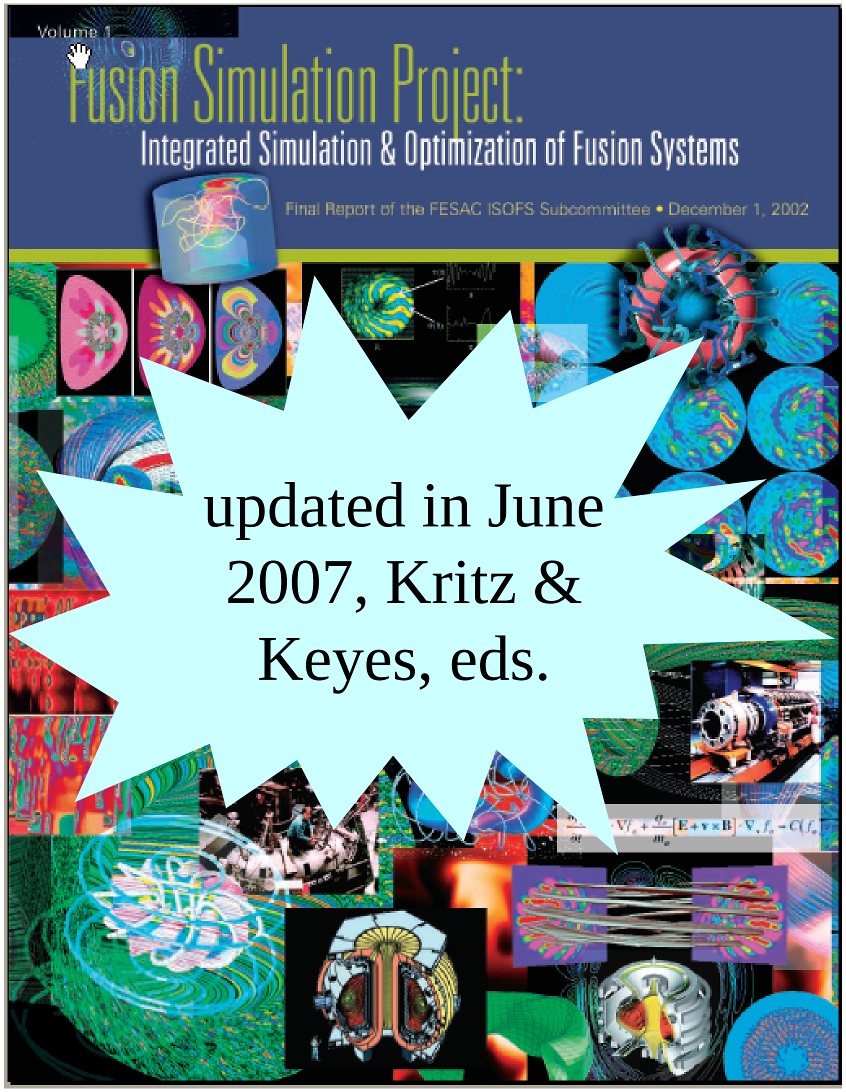
    $$Ax = \lambda Bx$$



- Software integration
- Performance optimization

Terascale Optimal PDE Simulations

SciDAC’s Fusion Simulation Project: support of the international fusion program



Volume 1

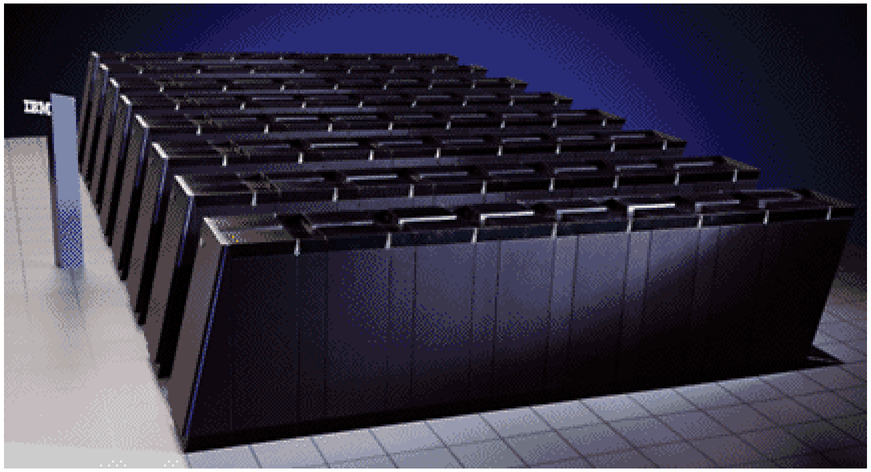
Fusion Simulation Project:

Integrated Simulation & Optimization of Fusion Systems

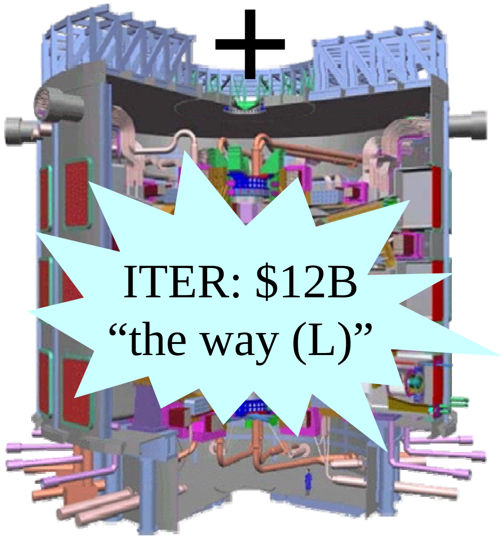
Final Report of the FESAC ISOFS Subcommittee • December 1, 2002

updated in June 2007, Kritz & Keyes, eds.

J. Fusion Energy 20: 135-196 (2001)

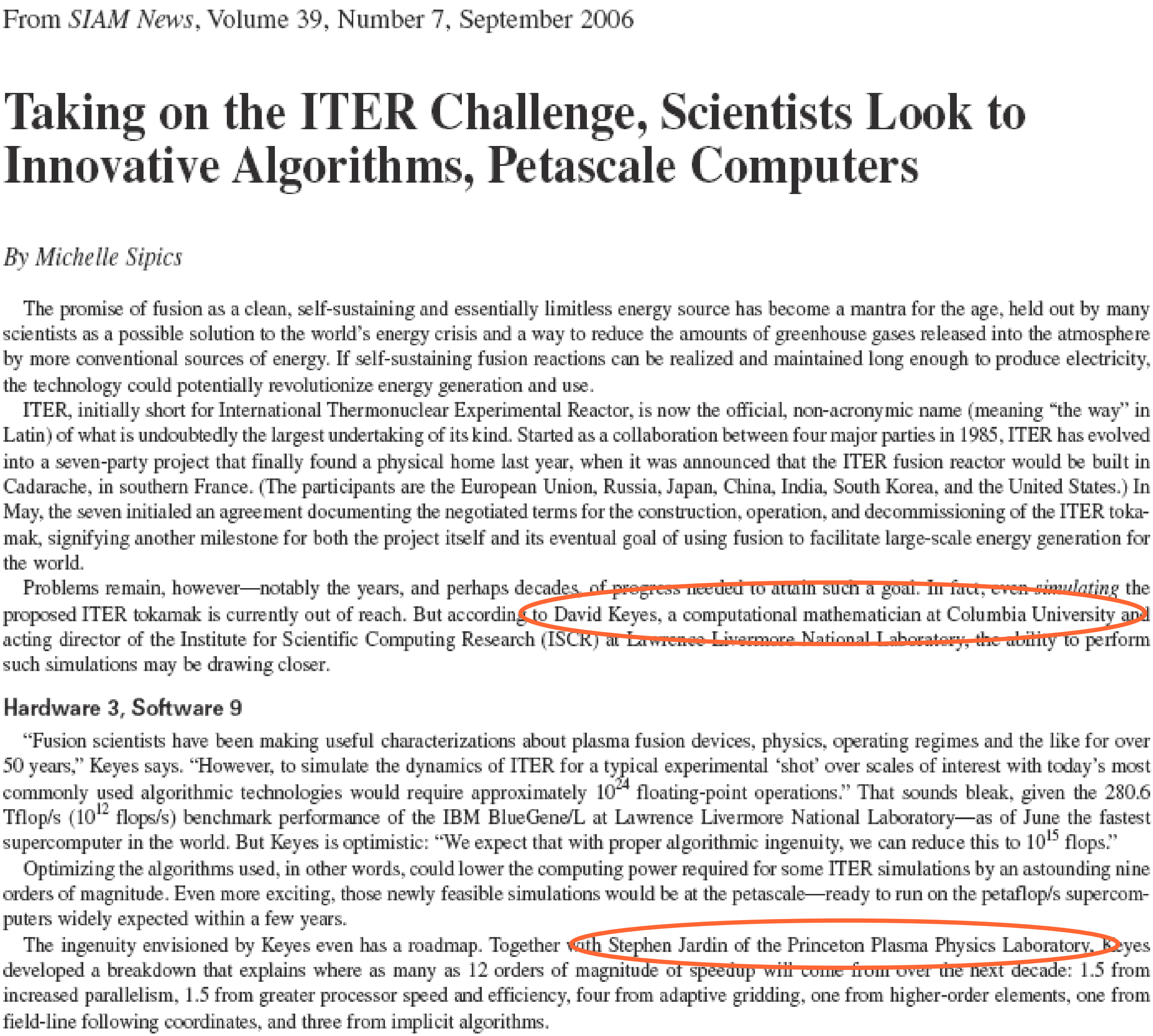


+



ITER: $12B “the way (L)”

Fusion by 2017; criticality by 2022



From *SIAM News*, Volume 39, Number 7, September 2006

Taking on the ITER Challenge, Scientists Look to Innovative Algorithms, Petascale Computers

By Michelle Sipics

The promise of fusion as a clean, self-sustaining and essentially limitless energy source has become a mantra for the age, held out by many scientists as a possible solution to the world's energy crisis and a way to reduce the amounts of greenhouse gases released into the atmosphere by more conventional sources of energy. If self-sustaining fusion reactions can be realized and maintained long enough to produce electricity, the technology could potentially revolutionize energy generation and use.

ITER, initially short for International Thermonuclear Experimental Reactor, is now the official, non-acronymic name (meaning "the way" in Latin) of what is undoubtedly the largest undertaking of its kind. Started as a collaboration between four major parties in 1985, ITER has evolved into a seven-party project that finally found a physical home last year, when it was announced that the ITER fusion reactor would be built in Cadarache, in southern France. (The participants are the European Union, Russia, Japan, China, India, South Korea, and the United States.) In May, the seven initialed an agreement documenting the negotiated terms for the construction, operation, and decommissioning of the ITER tokamak, signifying another milestone for both the project itself and its eventual goal of using fusion to facilitate large-scale energy generation for the world.

Problems remain, however—notably the years, and perhaps decades, of progress needed to attain such a goal. In fact, even *simulating* the proposed ITER tokamak is currently out of reach. But according to David Keyes, a computational mathematician at Columbia University and acting director of the Institute for Scientific Computing Research (ISCR) at Lawrence Livermore National Laboratory, the ability to perform such simulations may be drawing closer.

Hardware 3, Software 9

"Fusion scientists have been making useful characterizations about plasma fusion devices, physics, operating regimes and the like for over 50 years," Keyes says. "However, to simulate the dynamics of ITER for a typical experimental 'shot' over scales of interest with today's most commonly used algorithmic technologies would require approximately 10^{24} floating-point operations." That sounds bleak, given the 280.6 Tflop/s (10^{12} flops/s) benchmark performance of the IBM BlueGene/L at Lawrence Livermore National Laboratory—as of June the fastest supercomputer in the world. But Keyes is optimistic: "We expect that with proper algorithmic ingenuity, we can reduce this to 10^{15} flops."

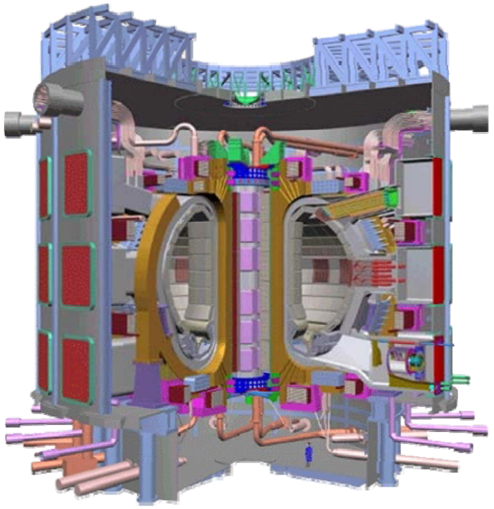
Optimizing the algorithms used, in other words, could lower the computing power required for some ITER simulations by an astounding nine orders of magnitude. Even more exciting, those newly feasible simulations would be at the petascale—ready to run on the petaflop/s supercomputers widely expected within a few years.

The ingenuity envisioned by Keyes even has a roadmap. Together with Stephen Jardin of the Princeton Plasma Physics Laboratory, Keyes developed a breakdown that explains where as many as 12 orders of magnitude of speedup will come from over the next decade: 1.5 from increased parallelism, 1.5 from greater processor speed and efficiency, four from adaptive gridding, one from higher-order elements, one from field-line following coordinates, and three from implicit algorithms.

Scaling fusion simulations up to ITER

			Small tokamak	Large tokamak	Huge tokamak
name	**symbol**	**units**	**CDX-U**	**DIII-D**	**ITER**
Field	B_0	Tesla	0.22	1	5.3
Minor radius	a	meters	.22	.67	2
Temp.	T_e	keV	0.1	2.0	8.
Lundquist no.	S		1×10^4	7×10^6	5×10^8
Mode growth time	$\tau_A S^{1/2}$	s	2×10^{-4}	9×10^{-3}	7×10^{-2}
Layer thickness	$aS^{-1/2}$	m	2×10^{-3}	2×10^{-4}	8×10^{-5}
zones	$N_R\times N_\theta\times N_\phi$		3×10^6	5×10^{10}	3×10^{13}
CFL timestep	$\Delta X/V_A$ (Explicit)	s	2×10^{-9}	8×10^{-11}	7×10^{-12}
Space-time pts			6×10^{12}	1×10^{20}	6×10^{24}

10^{12} needed (explicit uniform baseline)



International Thermonuclear Experimental Reactor

2017 – first experiments, in Cadaraches, France

c/o S. Jardin, PPPL

Where to find 12 orders of magnitude in 10 years?

Hardware: 3

- 1.5 orders: ...rease... ...ocessor ...peed and efficiency
- 1.5 orders: ... ...y

Software: 9

- 1 ord...
  - Sa... ...elements
- 1 ...
- 4 ord...
  - ...es req... ...lume and resolution requireme...s awa... ...s severe
- 3 orders: implicit solver
  - Mode growth time 9 orders longer than A...fven-limited CFL

Algorithmic improvements bring yottascale (10^{24}) calculation down to petascale (10^{15})!

Comments on ITER simulation roadmap

- Increased processor speed
  - 10 years is 6.5 Moore doubling times
- Increased concurrency
  - BG/L is already 2^{17} procs, MHD now routinely at ca. 2^{12}
- Higher-order discretizations
  - low-order preconditioning of high-order discretizations
- Flux-surface following gridding
  - in SciDAC, this is ITAPS; evolve mesh to approximately follow flux surfaces
- Adaptive gridding
  - in SciDAC, this is APDEC; Cartesian AMR
- Implicit solvers
  - in SciDAC, this is TOPS; Newton-Krylov w/multigrid preconditioning

Challenges in magnetic fusion

- Conditions of interest possess two properties that pose great challenges to numerical approaches—anisotropy and stiffness
  - Anisotropy produces subtle balances of large forces, and vastly different transport properties, parallel and perpendicular to magnetic flux surfaces
  - Stiffness reflects the vast range of time-scales in the system: targeted physics is slow (~transport scale) compared to waves
- These have led to a family of codes specialized to numerous regimes (52 DOE codes inventoried in 2002)

Collaboration illustrations from computational MHD in SciDAC

- *Meeting at* a well-defined traditional interface
  - swapping in new linear solvers (production codes NIMROD, M3D, GTC)
- *Changing* the traditional interface
  - changing the basis – still linear (production code AORSA)
- *Moving* the traditional interface
  - backing up to nonlinearly implicit formulation (GEM prototype codes)

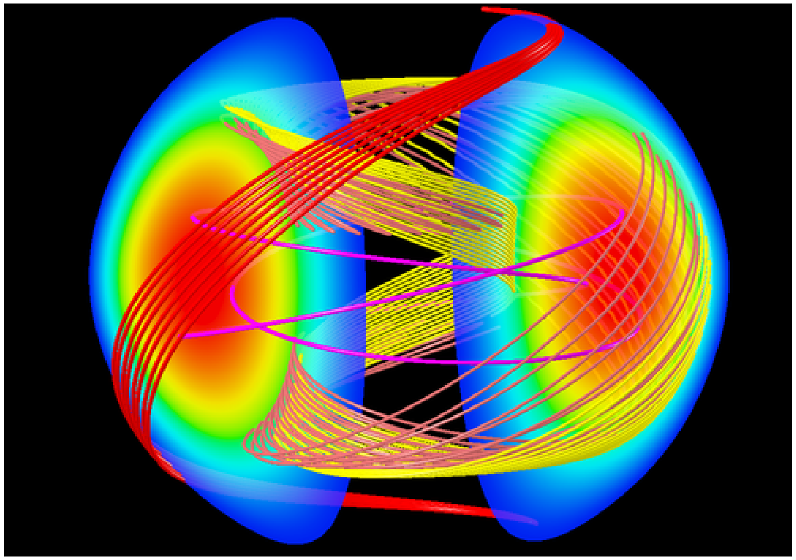
“Off-the-shelf” with computational MHD

- **NIMROD code (UWisconsin)**
  - **direct elimination replaces PCG solver for robustness in 2D**
  - **scalable implementation of old algorithm for *Ax=b***
- **M3D and GTC codes (Princeton)**
  - **Algebraic multigrid replaces block Jacobi/ASM preconditioner for optimality**
  - **new algorithm callable across *Ax=b* interface**
- **AORSA code (ORNL)**
  - **configuration space discretization replaces Fourier space**
  - **new formulation of *Ax=b***

Significance: The fusion community may use more cycles on unclassified U.S. DOE computers than any other (e.g., 32% of all cycles at NERSC in 2003). Well over 90% of these cycles are spent solving *linear systems* in M3D, NIMROD, and AORSA, which are prime U.S. code contributions to the designing of ITER.

TOPS-CEMM collaboration: direct elimination for robustness in NIMROD

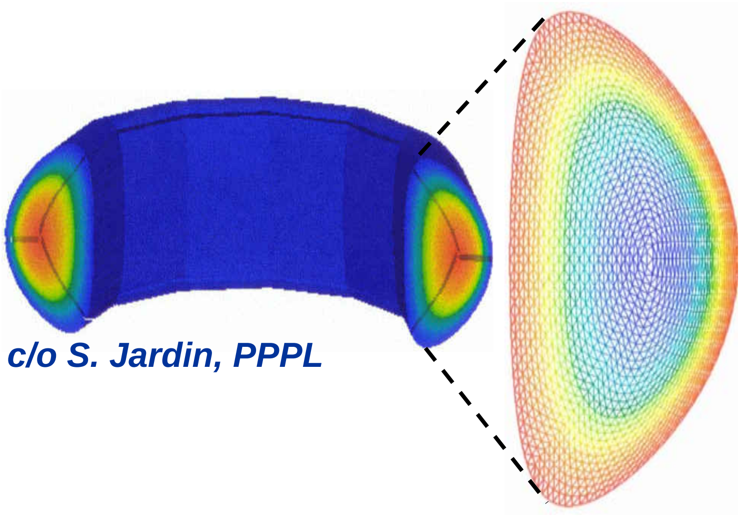
- **NIMROD code**
  - FFT in toroidal direction
  - high-order finite elements in poloidal
  - complex, nonsymmetric linear systems with 10K-100K unknowns (>90% exe. time)
- **TOPS collaboration**
  - replacement of diagonally scaled Krylov with SuperLU, a supernodal parallel sparse direct solver
  - 3D production runs are 4-5 × faster



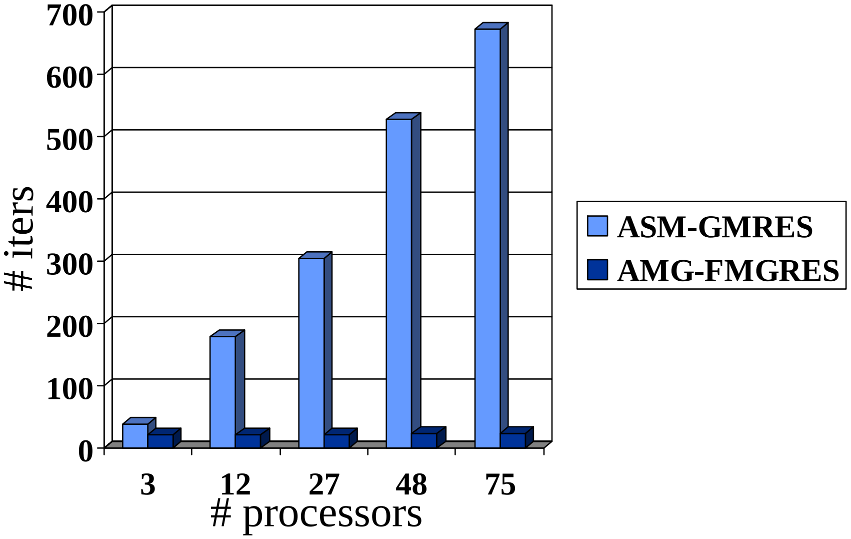
c/o D. Schnack, UWisconsin

TOPS-CEMM collaboration: multigrid for optimality in M3D

- **M3D code**
  - Finite differences in toroidal direction
  - unstructured mesh, hybrid FE/FD discretization with C0 elements in poloidal planes
  - Sequence of real Poisson-like scalar systems **(>90% exe. time)**
- **TOPS collaboration**
  - reformulation of Poisson solves to exploit symmetry and coefficient reuse
  - replacement of additive Schwarz (ASM) preconditioner with algebraic multigrid (AMG)
  - achieved mesh-independent convergence rate
  - 4-5 × improvement in solver time

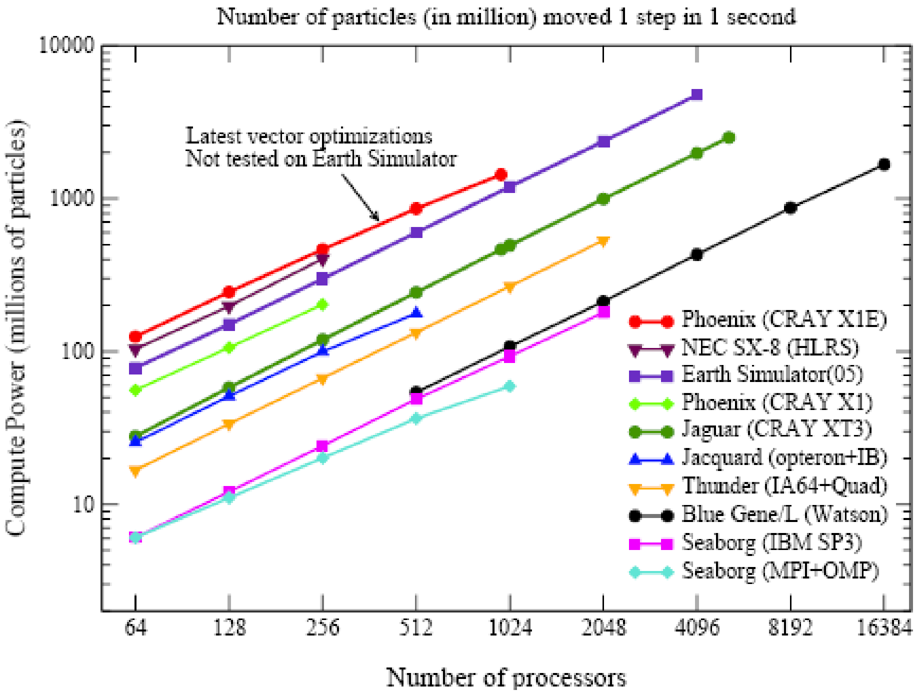


c/o S. Jardin, PPPL



TOPS-GPSC collaboration: multigrid for optimality in GTC

- **GTC code**
  - **hybrid particle/field code**
  - **limit to convergence scalability: need to extend diagonally dominant Poisson problems solved in poloidal crossplanes (for "adiabatic electron" limit) to pure Poisson ("nonadiabatic")**
  - **limit to concurrency: partitioning in poloidal direction only**
  - **One of three codes just selected for first access to DOE petascale hardware in 4Q 2008**

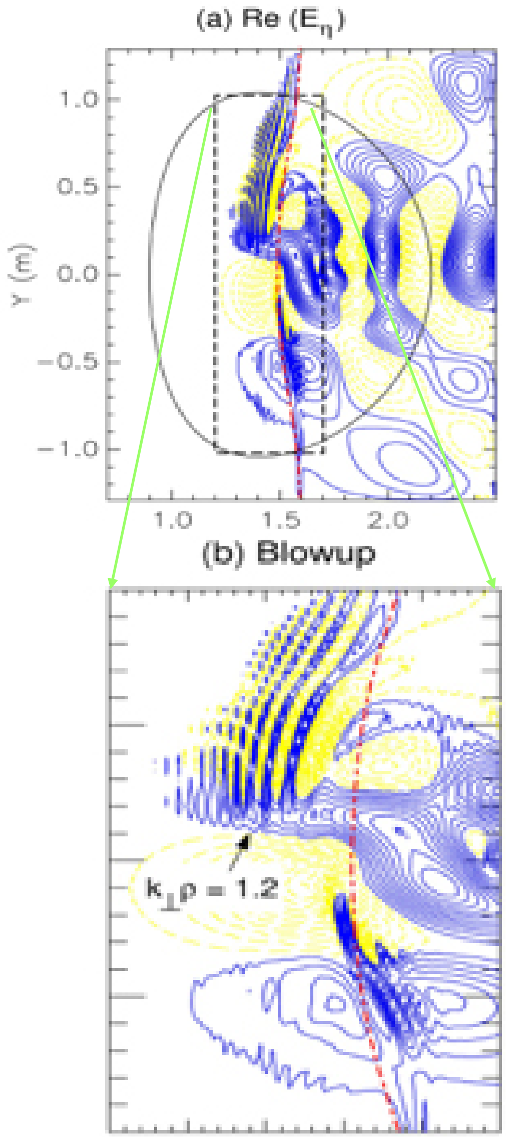


c/o S. Ethier, PPPL

- **TOPS collaboration**
  - **introduce multidimensional domain decomposition**
  - **replace diagonally scaled Krylov solver with algebraic multigrid to extend to "nonadiabatic" electrons**

TOPS-AORSA collaboration: new bases for storage

- ## AORSA code
  - fully spectral harmonic Maxwell formulation for RF plasma heating
  - large, dense systems with 780 GB of matrix data
- ## TOPS collaboration
  - replacement of Fourier formulation with physical ("configuration") space
  - 3D production runs are 27 × faster (the linear systems are solved 100 × faster)
  - storage is only 26 GB

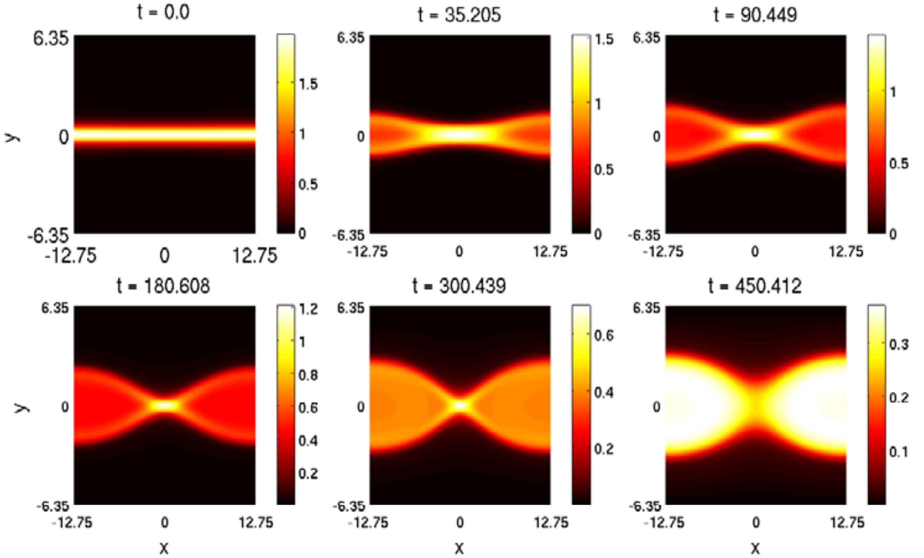


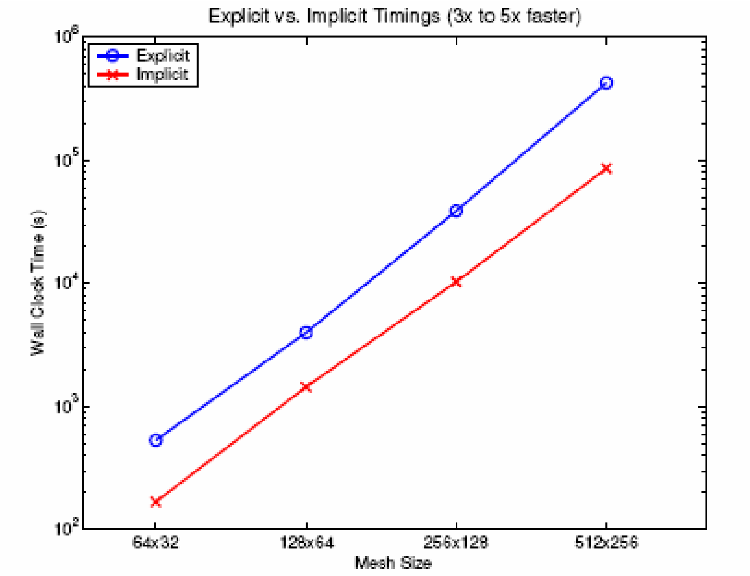
c/o D. Batchelor, ORNL

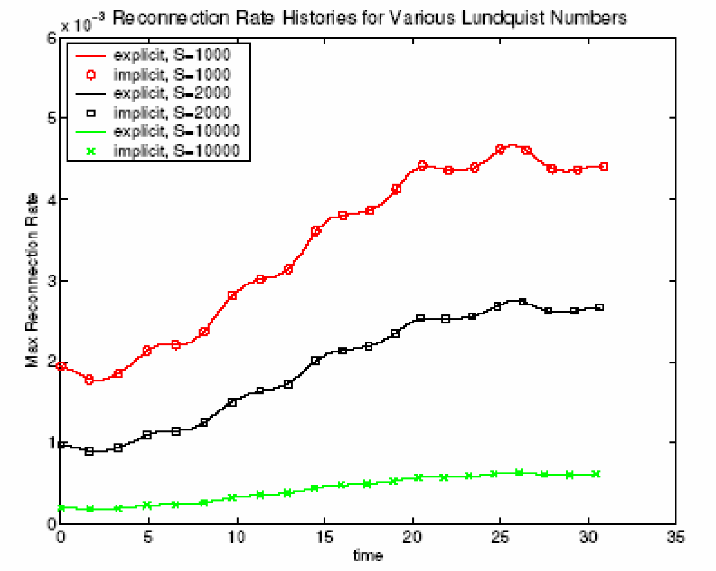
APDEC-TOPS-CEMM collaboration: nonlinear implicitness for multiple scales

- **GEM model magnetic reconnection problem**
- **Implemented in APDEC's Chombo framework (but without the AMR turned on for this study)**
- **Solved with TOPS' SUNDIALS matrix-free Newton-Krylov solver**

Current





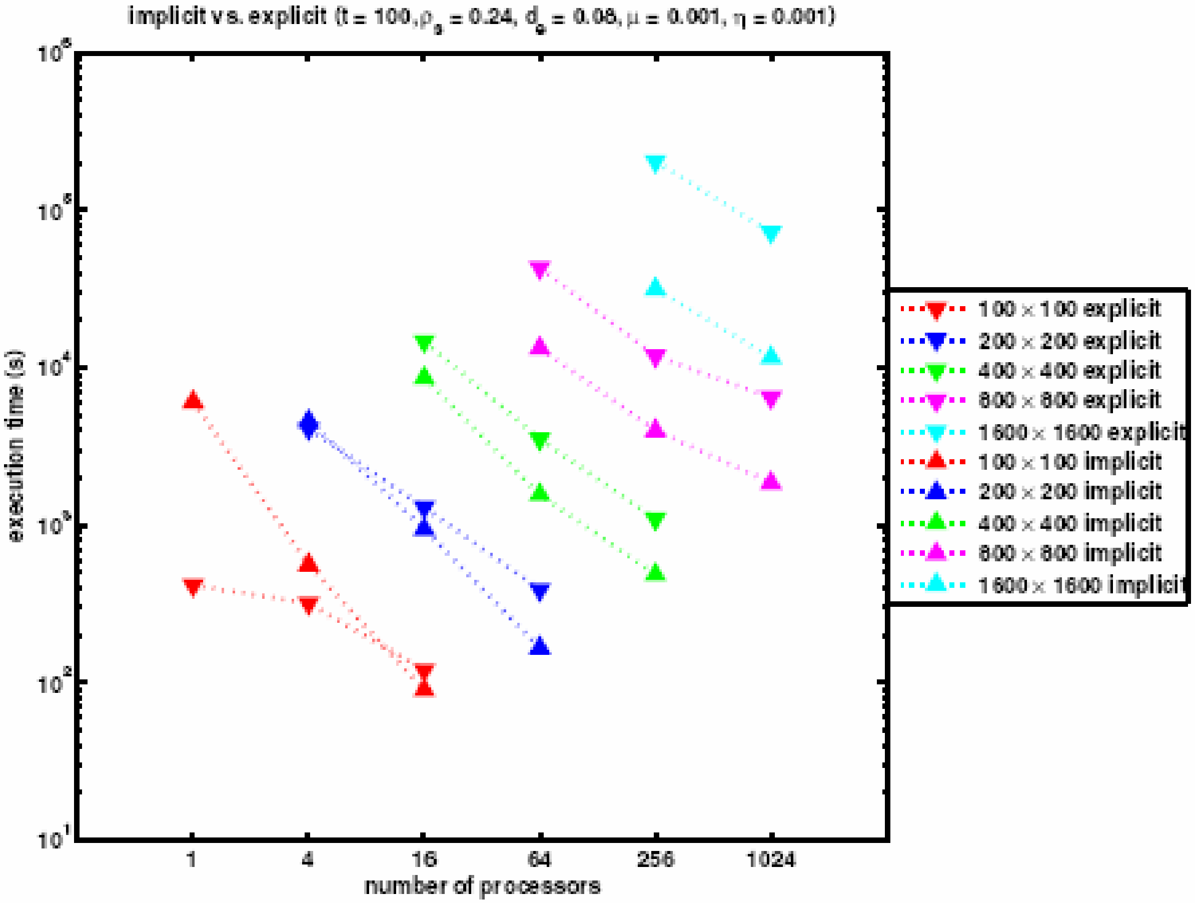


J. Brin et al., "Geospace Environmental Modeling (GEM) magnetic reconnection challenge," **J. Geophys. Res.** 106 (2001) 3715-3719.

c/o R. Samtaney, PPPL & D. Reynolds, UCSD

TOPS-CEMM collaboration: nonlinear implicitness for multiple scales

explicit/implicit execution time comparison

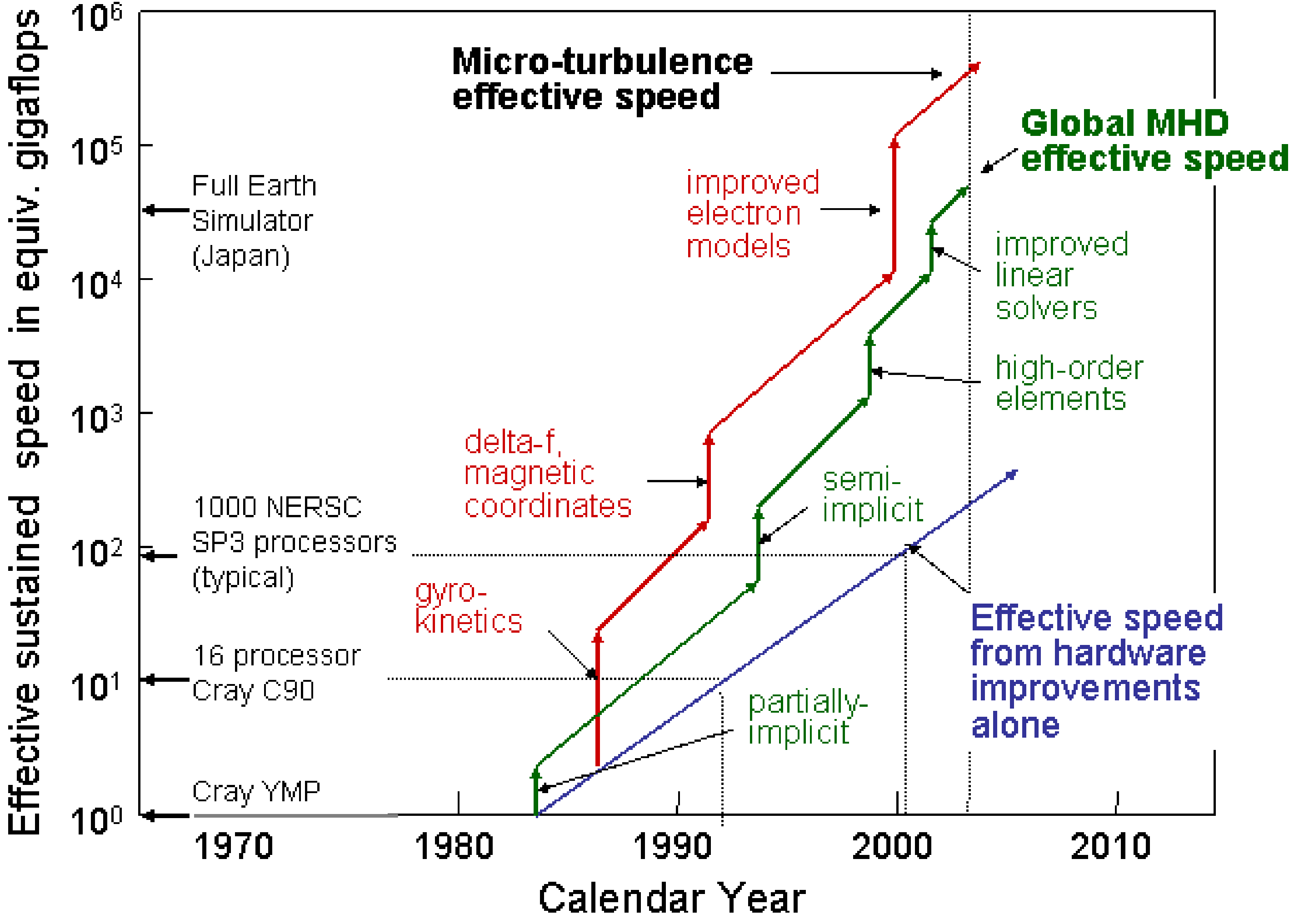


c/o F. Dobrian, et al.

- *Magnetic reconnection*: previous example was compressible – primitive variable; this example is incompressible – streamfunction/vorticity
- Replace explicit updates with implicit Newton-Krylov from **PETSc** with factor of ~5× in speedup

“Moore’s Law” for MHD simulations

Magnetic Fusion Energy: “Effective speed” increases came from both faster hardware and improved algorithms



“Semi-implicit”:

All waves treated implicitly, but still stability-limited by transport

“Partially implicit”:

Fastest waves filtered, but still stability-limited by slower waves

Figure from SCaLeS report, Volume 2

Simulation claims

- **Simulation will become *increasingly cost-effective* relative to experiment, while never fully replacing experiment**
- **Simulation may define today's *limit to progress* in areas that are already theoretically well modeled**
- **Simulation *aids model refinement* in areas not already well modeled (via interplay with theory)**
- **Advanced simulation makes scientists and engineers *more productive* (can partially offset national disadvantage in workforce recruiting)**

Lessons from SciDAC

- **Much high pay-off work to be done in large-scale simulation is *at the interface* between disciplines**
- ***Mission-oriented laboratories* and *idea-oriented universities* make good partners in developing the "science" of simulation**

Transition to discussion…

- **Where in this milieu are the aero companies and NASA?**
- **How will aero companies and NASA participate in the next challenging phase of the simulation agenda – multiphysics code coupling?**

Benefits industry and labs can confer

- Validation of research relevance
- Identification of relevant research challenges and opportunities
- Lobbying for federal research initiatives
- “In-kind” collaborations
- Direct research support

Benefits industry and labs can receive

- Demonstrated ideas from “blue sky” investments
- Workforce training
  - new employees
  - continuing education
- Prototype software (to harden internally)

SciDAC's software cultural challenges

- **We are in a constantly changing, uncharted environment**
  - **scant social/organizational research on how organizations can make best use of HPC resources**
- **Ability to build big machines has outpaced our ability to build codes that can make effective use of these machines**
  - **big codes are large, interdisciplinary projects, whose team dynamics can be difficult to manage (Douglass E. Post *et al.* studies of ASC codes)**
  - **no universal socio-managerial understanding or convention of how code teams form, evolve, and disband**
- **Simulation for scientific discovery appears to "have it easier" than simulation for policy-making, treaty negotiation, or engineering plant investment?**
  - ***NO!* SciDAC intends to provide support for: climate modeling, the designers of ITER, the designers of ILC, major telescope projects, etc.**

SciDAC's software cultural challenges (cont.)

- **Though addressing common goals, teams have different expectations, different æsthetics, and different rewards**
- **For applications ("Apps") people**
  - **Modeling and simulation technologies aren't precise 'tools': they're epistemological artifacts**
    - ***Approximations of complex problems and necessarily incomplete***
    - ***Codes are never stable; always evolving – like the collective's knowledge***
  - **A code may be subordinated to a particular scientific question or campaign**
    - ***Codes, per se, do not bring the professional rewards***
- **For enabling technology ("ET") people**
  - **Software toolkits have well defined inputs, outputs, and requirements**
    - ***We (often) can measure how accurate, efficient, and effective we are***
    - ***We (often) have abstract, timeless, well-posed targets***
  - **Software toolkits are intended to have long lifetimes, be of general purpose**
    - ***Codes, algorithms, etc., are their own professional rewards***

SciDAC’s software cultural challenges (cont.)

- Though addressing common goals, we have different priorities
- For “Apps” people
  - A code that is *finally* ready to use is hard-earned scientific currency

    There is a desire to control distribution

    There is a reluctance to allow bugs or inefficiencies to be exposed
  - A code that can *finally* do something should be used now, and improved later
- For “ET” people
  - Software toolkits are made to be given away

    There is a desire to distribute widely, together with demo applications

    There is a strong interest in catching inefficiencies and publishing bug fixes
  - Software toolkits should be improved as much as possible as soon as possible, and improvements “pushed” immediately into applications in the field

Summarizing some asymmetries

- **Challenging asymmetries, Apps *vs.* ETs**
  - **Complex, elusive models *vs.* (often) clean, timeless kernels**
  - **Subordination to particular science campaigns *vs.* general-purpose promotion**
  - **Value enhanced by guarding closely *vs.* value enhanced by distributing openly**
  - **Many papers published & grants written once a code achieves a certain threshold of functionality *vs.* papers published & grants written continually as toolkit is continually improved**
- **Synergistic asymmetries also exist (next few slides)**
  - ***Vive la différence!***

What Apps want (in toolkits from ETs)

- Develop code “without having to make bets”
  - accomplish certain abstract mathematical tasks
  - stay agnostic about particular solution methods and codes
  - run on laptops (on travel), low-cost unmetered clusters (at work), and on unique shared national resources
- Ordered goals (need them all for production use)
  - usability and robustness
  - portability
  - algorithmic efficiency (optimality) and implementation efficiency (within a processor and in parallel)
- Algorithmic optimality and software stability
  - large-scale problems require scalable components
  - don’t want to “hand code” for evanescent environments ever again

Challenges for software designers

- **The old challenge:**

  Increase functionality and capability for a small number of users who are expert in the domain of the software

- **A new challenge:**

  Increase ease of use (for correctness *and* efficiency) for a large number of users who are expert in something else (thank goodness!)

What's "new" in SciDAC toolkit software?

- Philosophy of library usage
  - complex algorithms with lots of callbacks to user code (e.g., to physics routines by implicit solvers)
  - polyalgorithms for adaptivity to applications and architecture
  - extensibility
  - good neighborliness: argument checking, profiling, etc.
- SciDAC provides resources for development, maintenance, and support (not enough for the full collaborative "embedded expert" demand, but…)
  - not just for one-off "dissertation scope" implementations, either!
- Past experience on terascale computers
  - DOE ASCI, NASA HPC, NSF centers
- Integration mandate
  - Within Centers & Institutes projects and *across* C&Is

An emergent design principle: multiple layers

- **Top layer (all users)**
  - **Abstract interface featuring language of application domain, hiding details, with conservative parameter defaults**
  - ***Robustness, correctness, ease of use***
- **Middle layers (experienced users)**
  - **Rich collection of state-of-the-art methods and data structures, exposed upon demand, highly configurable**
  - ***Capability, algorithmic efficiency, extensibility, composability, comprehensibility of performance and resource use***
- **Bottom layer (implementors)**
  - **Support for variety of execution environments**
  - ***Portability, implementation efficiency***

Relationships are built on trust and presence

- **Enabling technology software must be**
  - **Supported**

    by designated person or (better) an always-online bug-tracking team

    not an “automated 800 number”
  - **Stable**

    must be confidence in support for lifetime of app,

    interface should change only when justified by immediate major improvement in functionality, performance, interoperability, or portability
  - **Quality**

    must do things that a physics graduate student cannot do on her/his own
- **Enabling technologists should make effort to learn the simulation state of the art to display service mentality**
  - **Huddle with the Apps lead PI**
  - **Attend all-hands Apps meetings**
  - **Read some review papers in the Apps domain**

Idealized structure of SciDAC interactions

- **Inclusion of both sides in setting priorities**
- **Well defined responsibilities**
  - **tasks and dates**
- **Well defined metrics for success**
  - **how much bigger, faster, more accurate, more scalable, more robust, more portable, etc., with what consequences to bottom-line scientific discovery?**
- **Well defined software interfaces**
  - **“good fences make good neighbors”**
- **Well defined human interfaces**
  - **consistent, dedicated personnel the best**

Some vectors for exchange of technical ideas

- ## The technical paper
  - a highly refined scientific artifact designed to navigate tortuous paths en route to publication and to accomplish many extra things
  - candid and clear communication of the usefulness of an idea to a community of nonexperts in their context is not usually one of them
  - neither is “blank screen” codability
- ## Software
  - a very precise way of packaging expert knowledge for nonexperts, provided that the interface is good
  - not a good way for humans to communication with humans
- ## The *post-doc!*
  - a wonderful mechanism for Apps-to-ET impedance matching, in both directions
  - a prime beneficiary of the SciDAC experience
  - DOE should make an effort to provide each important link with a post-doc in each direction to grow the CS&E workforce, and almost incidentally, to accomplish great things for SciDAC

Pre-nups

- **How will intellectual property be protected?**
  - **reserve full physics capability of the Apps team code while releasing a demo version that showcases the use of the ET team toolkit**
- **Who will publish with whom, where, how, and when?**
  - **may exist different conventions for inclusion of project PI, inclusion of students, sequencing of multiple papers, relative of prestige of proceedings/journals**
- **On which collection of production machines is it important, ultimately, to have high performance?**
  - **may exist different priorities for exploiting features not available on all machines, like vectors, OpenMP, multiple FPU instructions**

Some "gotchas"

- For Apps
  - Standard interface of general purpose toolkit may not map well onto data structures and calling sequences of legacy code
  - Presently unneeded functionality may slightly complicate adoption or slightly degrade performance relative to a narrower implementation of the tools needed
- For ETs
  - Overall code performance/scalability may be captive to a bottleneck in physics implementation, not related to ET toolkit, reducing motivation for state-of-the-art software in other areas
  - Application code may not port to development environment of ET toolkit

ETs desire that Apps users will…

- **Understand range of algorithmic options w/tradeoffs**

  ***e.g.*, memory *vs.* time, comp. *vs.* comm., inner iteration work *vs.* outer**

- **Try all reasonable options “easily”**

  without recoding or extensive recompilation

- **Know how their toolkits are performing**

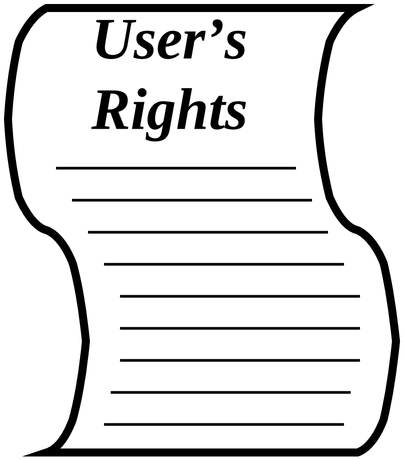
  with access to detailed profiling information

- **Intelligently drive toolkit research**

  ***e.g.*, publish joint papers with C&I researchers**

- **Simulate *truly new physics* free from previous limits**

  ***e.g.*, finer meshes, complex coupling, navigable data**



URLs

- **SciDAC homepage**

  http://www.scidac.gov/

- **TOPS SciDAC project on solvers**

  http://www.scidac.gov/math/TOPS.html

- **The SCaLeS report**

  http://www.pnl.gov/scales/