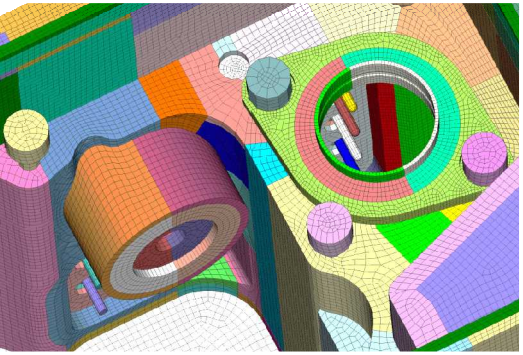
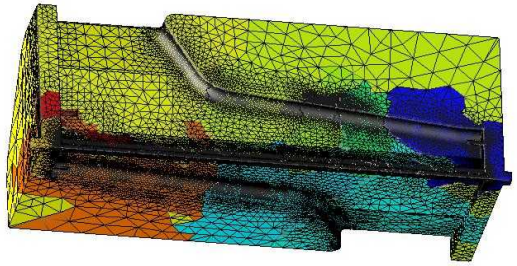
SAND2014-2826C





A Whirlwind Tour Through the Mathematics and Computer Science of Computational Simulation in Industry

Dr. David E. Womble

Senior Manager, Computational Simulation Group

Sandia National Laboratories

Sandia National Laboratories

Exceptional service in the national interest

U.S. DEPARTMENT OF ENERGY

NNSA

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

A Quick Overview of Sandia National Laboratories

Sandia National Laboratories

Albuquerque, New Mexico

Livermore, California

Kauai, Hawaii

Waste Isolation Pilot Plant, Carlsbad, New Mexico

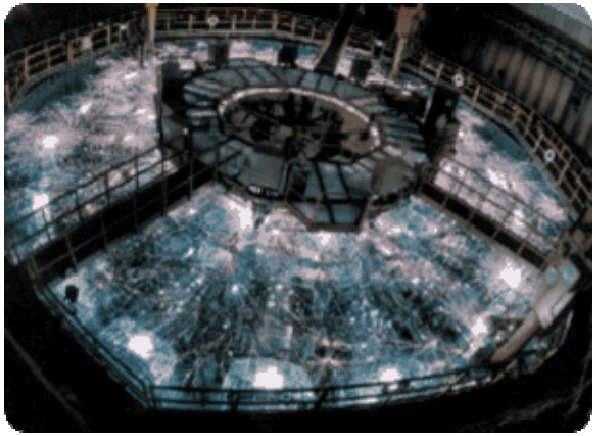
Pantex Plant, Amarillo, Texas

Tonopah, Nevada

Nuclear Weapons

Sandia National Laboratories

Pulsed power and radiation effects sciences

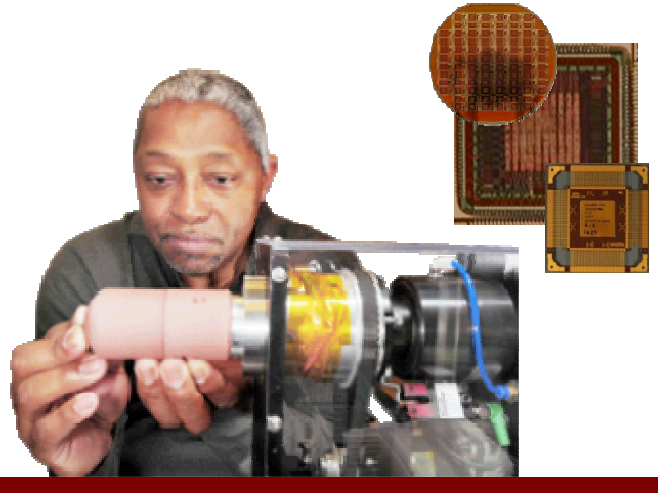


Design agency for nonnuclear components

- Neutron generators
- Arming, fuzing and firing systems
- Safety systems
- Gas transfer systems

Warhead systems engineering and integration

Production agency



Defense Systems and Assessments

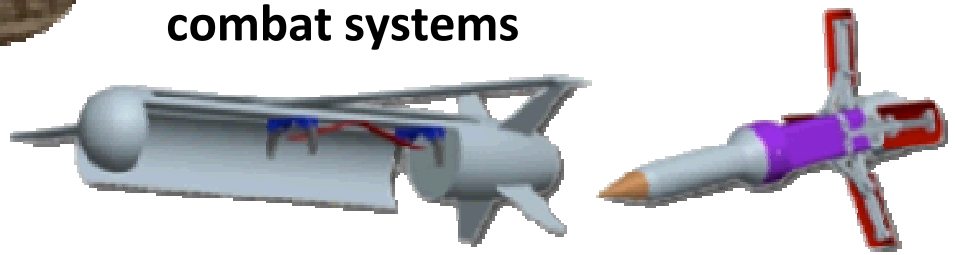
Sandia National Laboratories

Synthetic aperture radar

Support for NASA

Support for ballistic missile defense

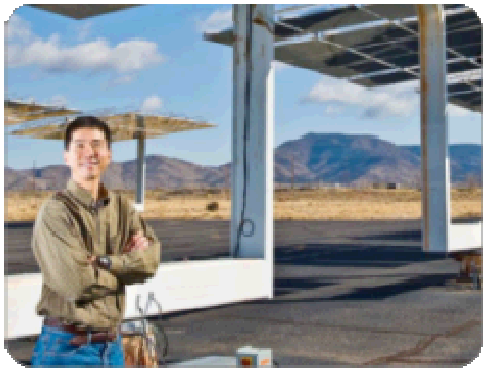
Ground sensors for future combat systems



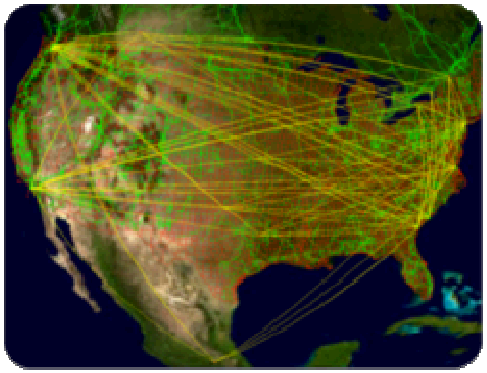
Energy, Climate, and Infrastructure Security

Sandia National Laboratories

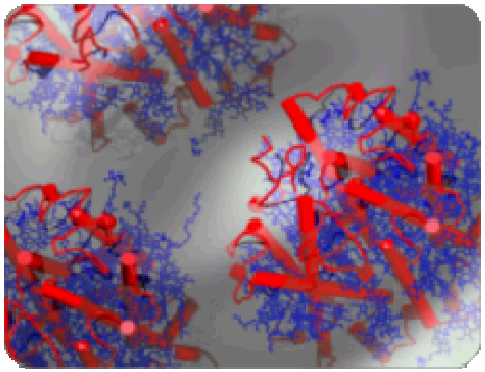
Energy



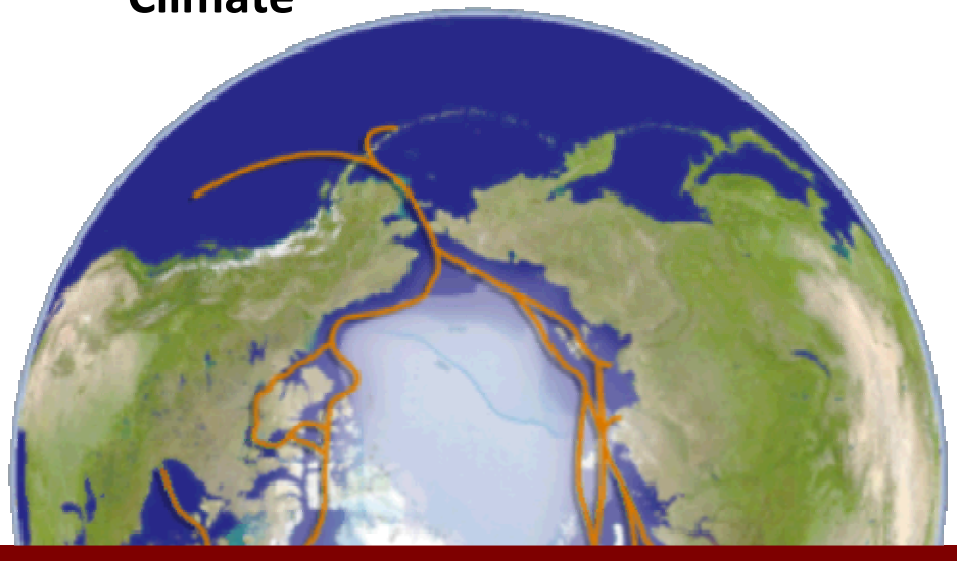
Infrastructure



Crosscuts and enablers



Climate



jbei
Joint BioEnergy Institute

International, Homeland, and Nuclear Security

Sandia National Laboratories

Critical asset protection

Homeland defense and force protection

Homeland security programs

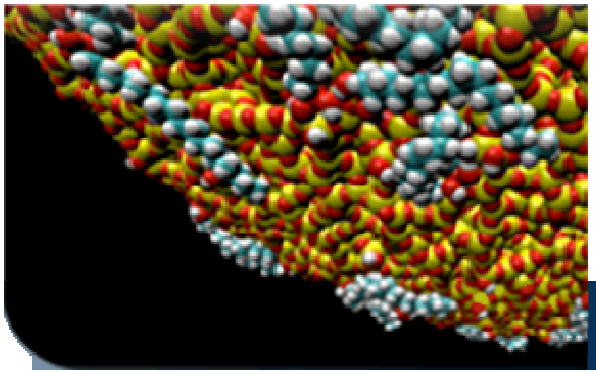
Global security

Science and Engineering Foundations

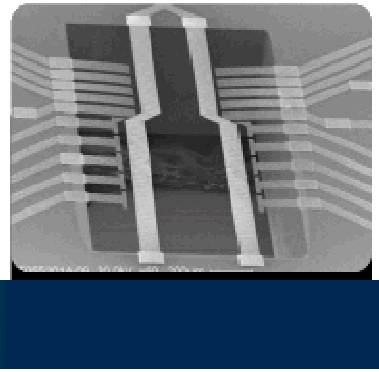
Sandia National Laboratories

Computing and information science

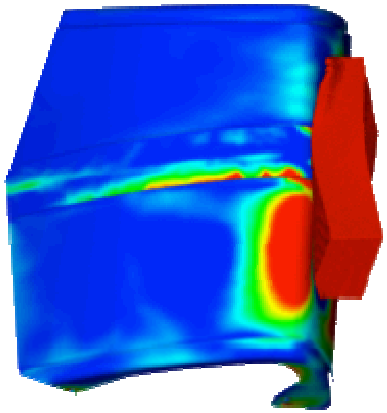
Materials science



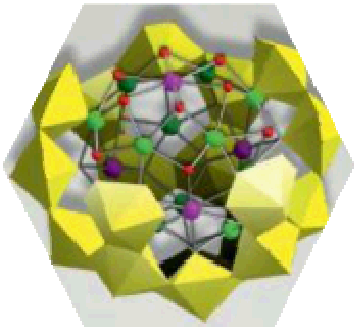
Nanodevices and microsystems



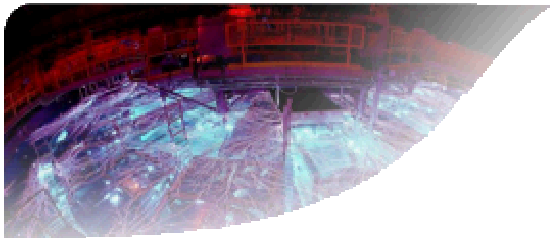
Engineering sciences



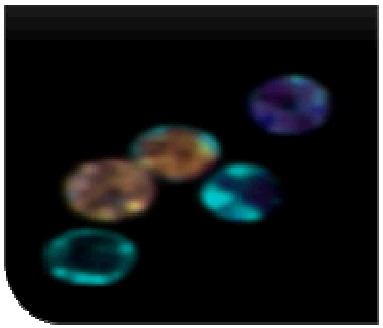
Geoscience



Radiation effects and high-energy density science



Bioscience



Now back to computational simulation
What type of computing am I talking about?

Sandia National Laboratories

Physics-based computing

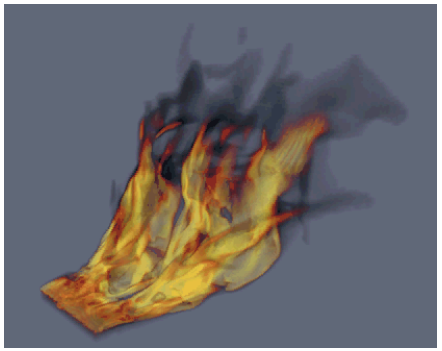
- Starts with a model, often a PDE or conservation law
- And a physical system
- And an initial/boundary conditions
- Then evaluates the model for that system.

Information-based computing

- Starts with data
- And searches for meaningful patterns or conclusions in the data
- Models exist, but are most often implicit and indirectly evaluated. (An analysis of Google FluTrends highlights the problems.)

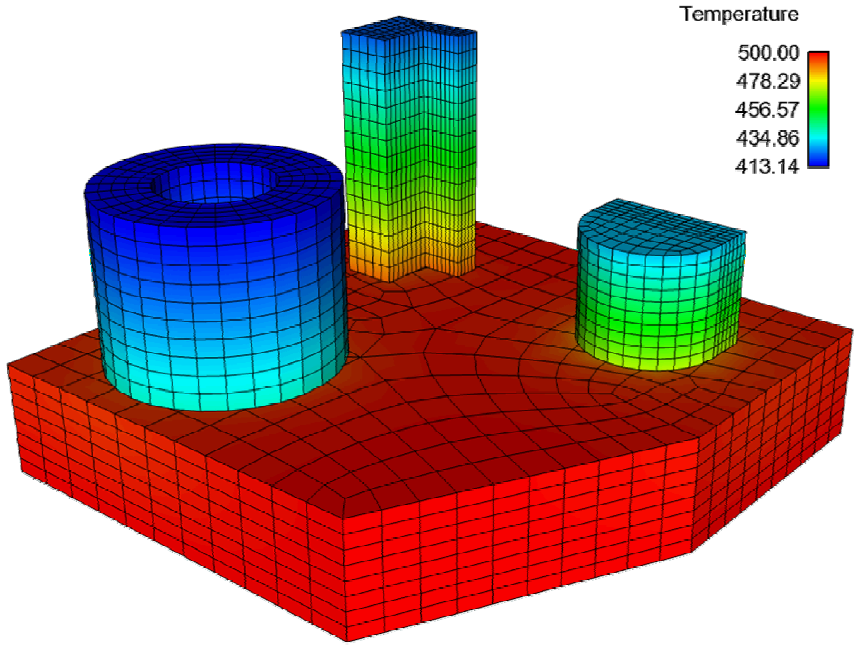
As an example, suppose we want to know the response of a system in a fire…..

- Fire is one source of a thermal “insult”
- There are many types of fires, and each creates a different environment.
- The physics of a fire is interesting, but let’s take that as a boundary condition for now.



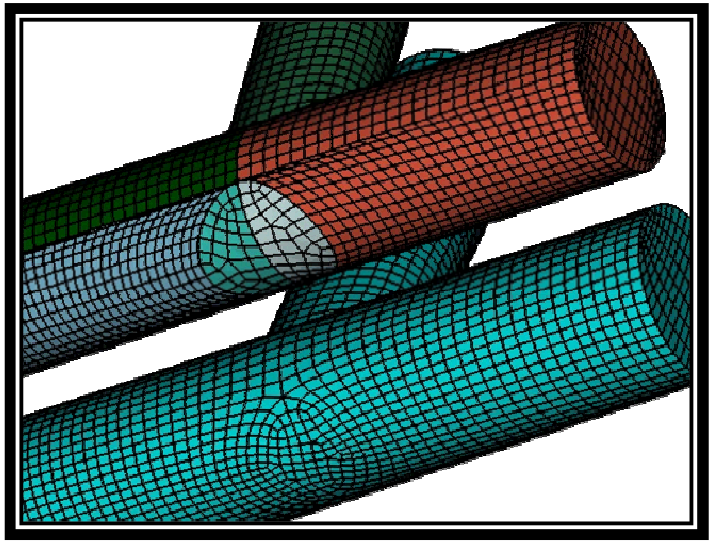
Let's start with thermal transport

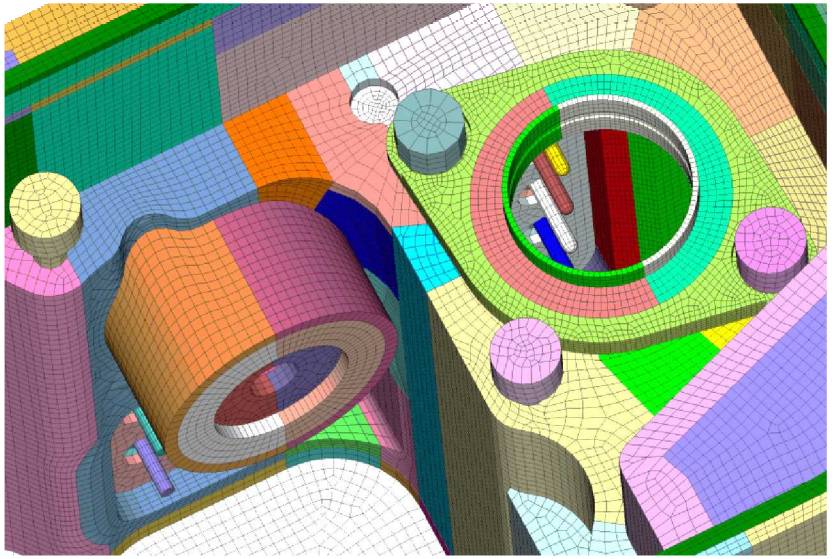
- What are the equations
  - Diffusion
  - Convection
  - Radiation
- What discretization should we use?
  - Finite difference? Finite volume? Finite element?
  - What type of element?
  - What order?



What are some of the other decisions that must be made?

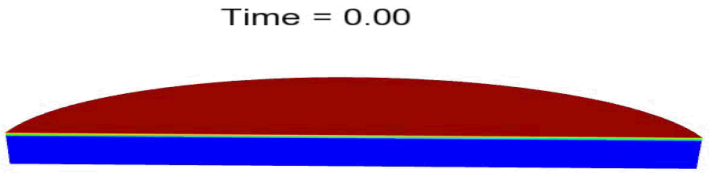
- How do we mesh?
  - Geometry cleanup
  - Defeaturing
  - Meshing and mesh quality
- What solvers will be used?
  - Direct
  - Iterative
  - Preconditioned CG
- Are we ready to solve the thermal problem

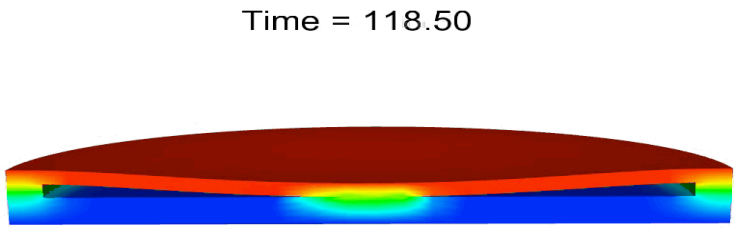




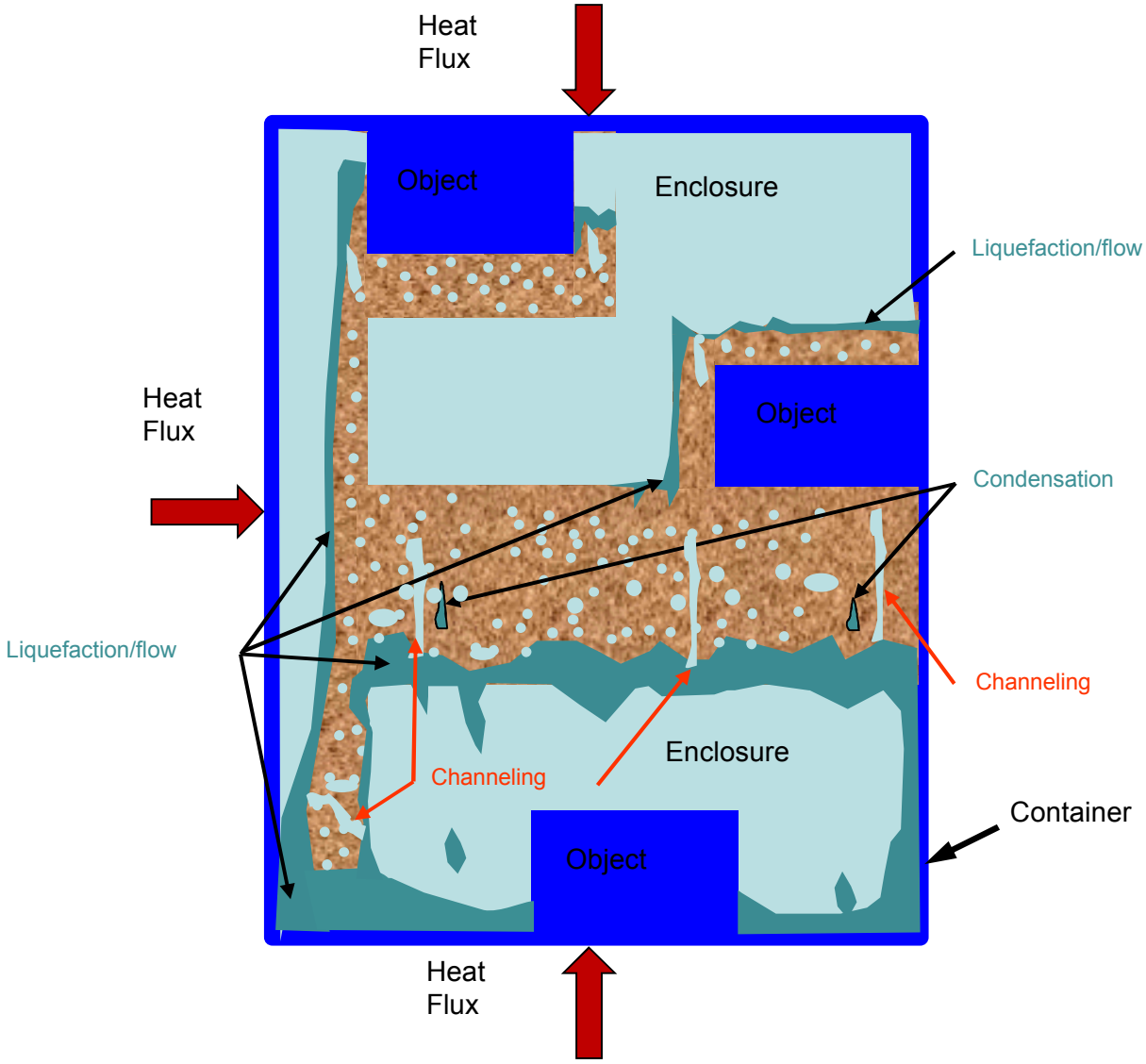
But we also know there are structural changes in thermal environments

- What new physics is introduced?
  - Solid mechanics, including thermal stress
  - Contact can create or eliminates thermal paths
  - Phase change
- We also have to go back and ask the same questions as before
  - Finite difference?, Finite volume? Finite element?
  - What type of element?
  - What order?
- And we need to think through the mathematics of coupling
  - Implicit or explicit
  - Time steps and errors
  - Data transfers

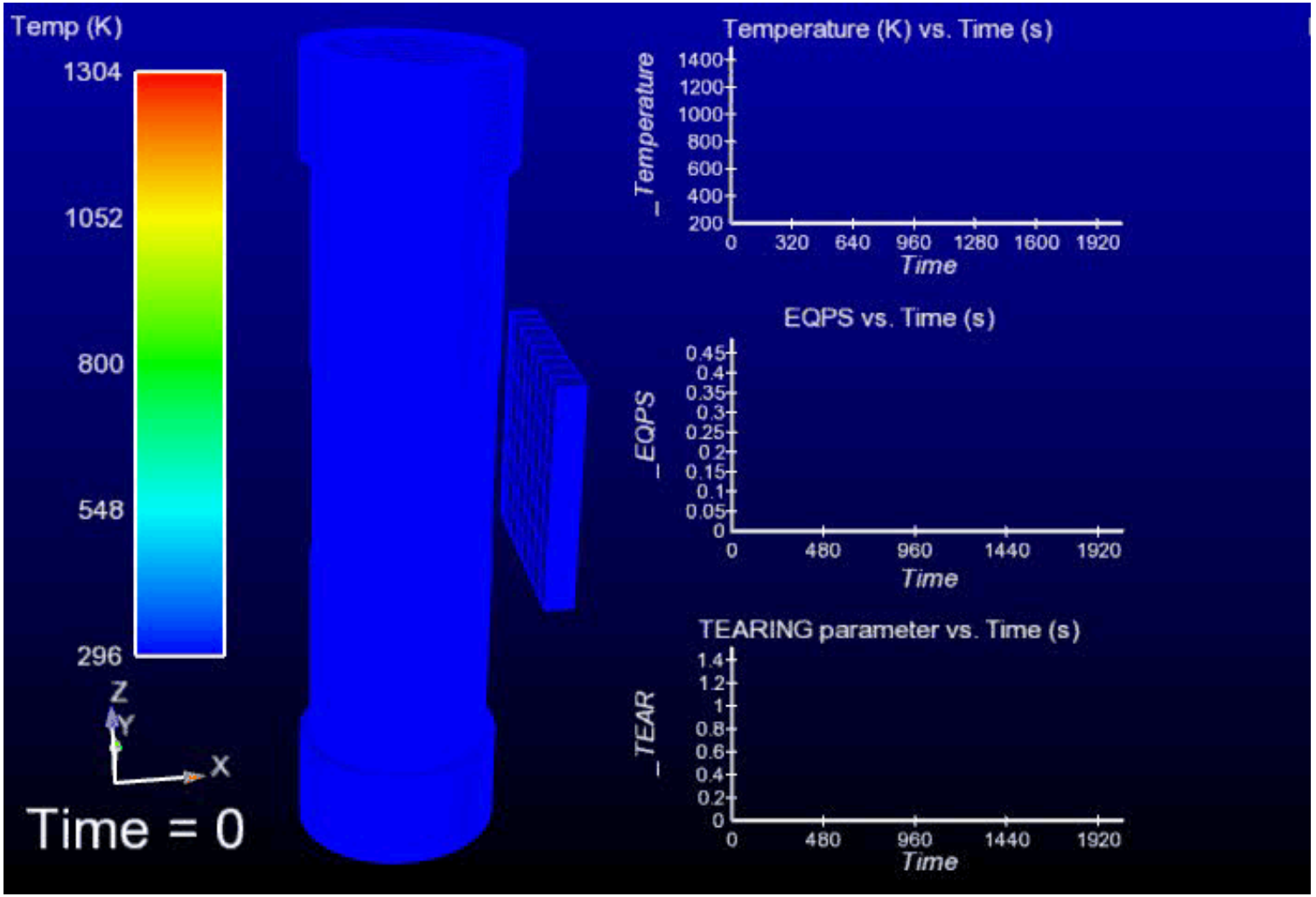




Phase change and multiphase flow is another twist



We also get pressure build-up in enclosed volumes resulting in fracture and failure.

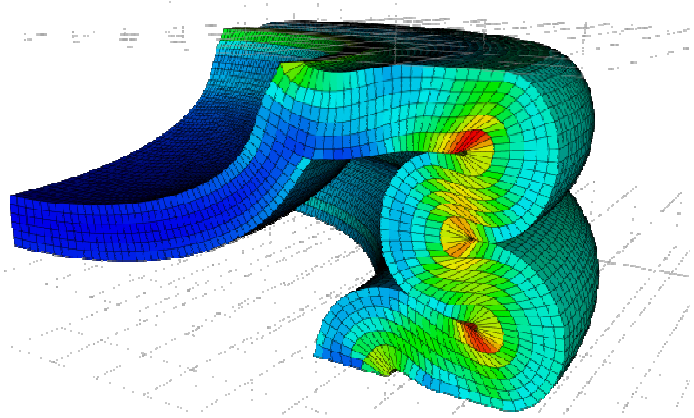


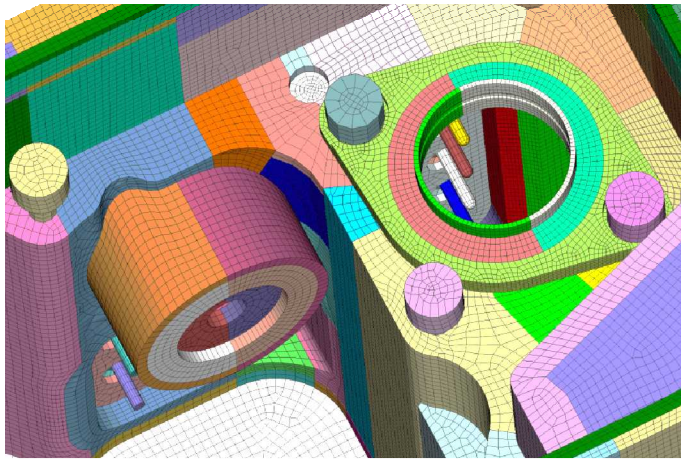
Let's summarize the situation so far

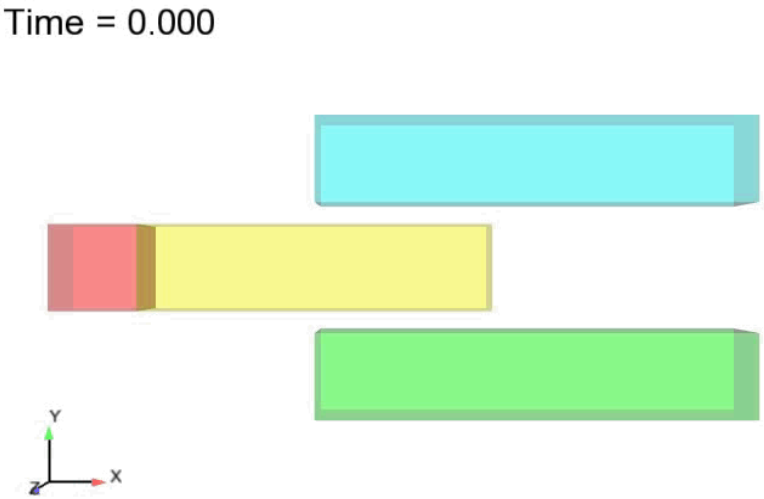
- We are
  - Simulating a system in a thermal environment
  - Coupling mechanical effects due to the thermal environment
  - Pushing the model to the point of system failure
- We are not
  - Modeling the environment itself (fire)
  - Subjecting the system to mechanical or electrical insult
  - Coupling aerodynamics or fluid flow effects
- Where do we need more mathematics or computer science?
  - The basic physics, discretizations and algorithms for the separate physics regimes at these scales is fairly well-known.
  - Need mathematics for coupling, contact, multiscale modeling, solvers, etc.

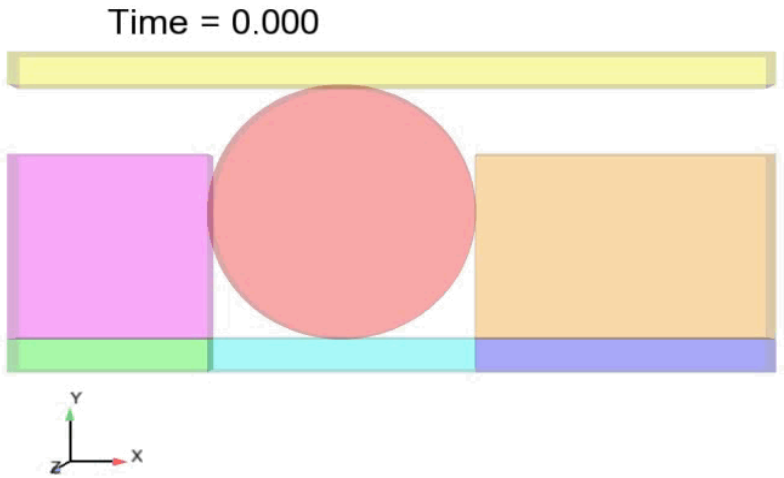
One challenge is contact.

- There are several types of contact



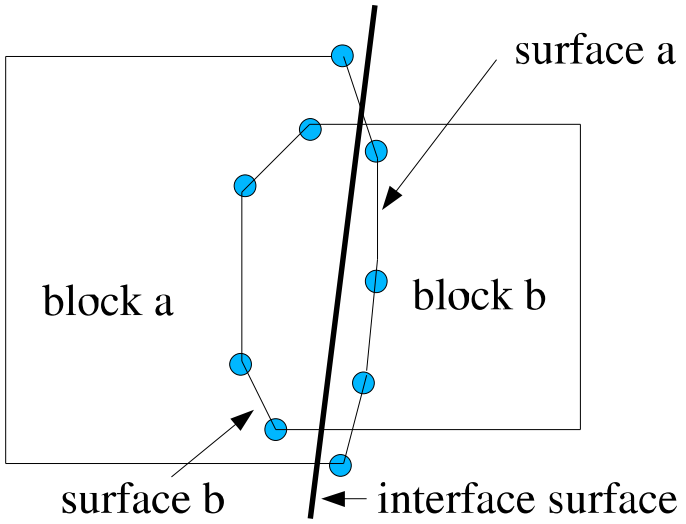


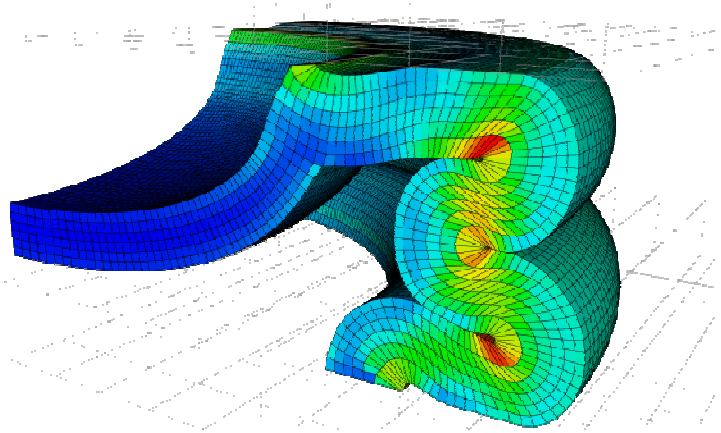




One challenge is contact

- Contact can be local or non-local
- Contact algorithms have different phases and differ in character from FP calculations
  - Block skinning: identify surfaces
  - Search: O(NlogN) but not local in memory
  - Enforcement: Iterative, augmented Lagrange algorithm for contact constraints
- Contact presents challenges for UQ and margin calculations
  - Implicit contact provides pre-loads (initial condition for the transient response
  - Mesh sensitivites and discontinuities

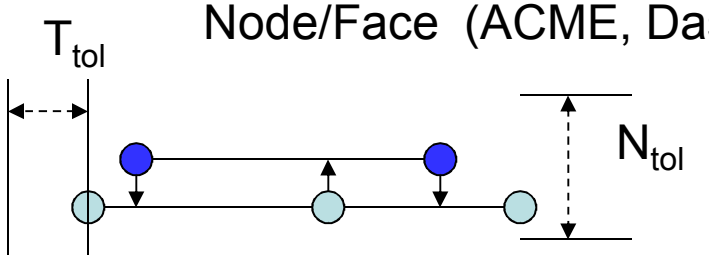




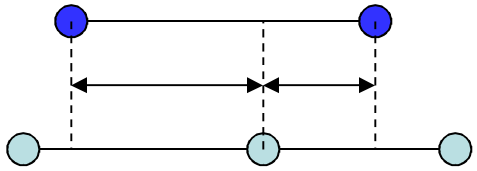
And there are several algorithmic approaches to contact

Sandia National Laboratories

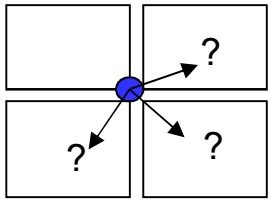
Node/Face (ACME, Dash)



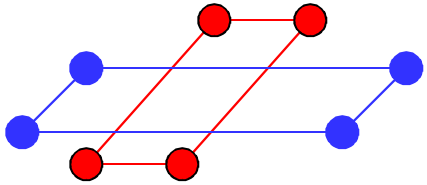
Face/Face (Dash)



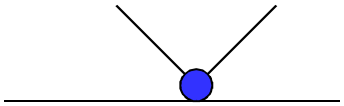
Ambiguous Cases:



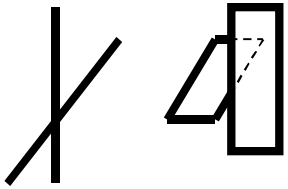
Corner Contacts, **Poor**



Face Coverage, **No Support**

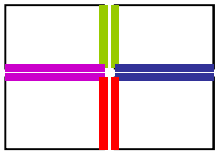


Sharp Point Contact, **Good**

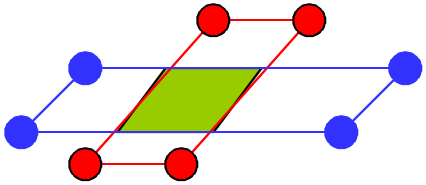


Beam and shell Edges, **No support**

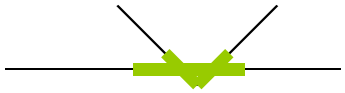
Ambiguous Cases:



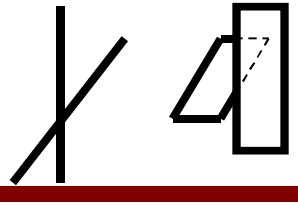
Corner Contacts, **Good**



Face Coverage, **Good**



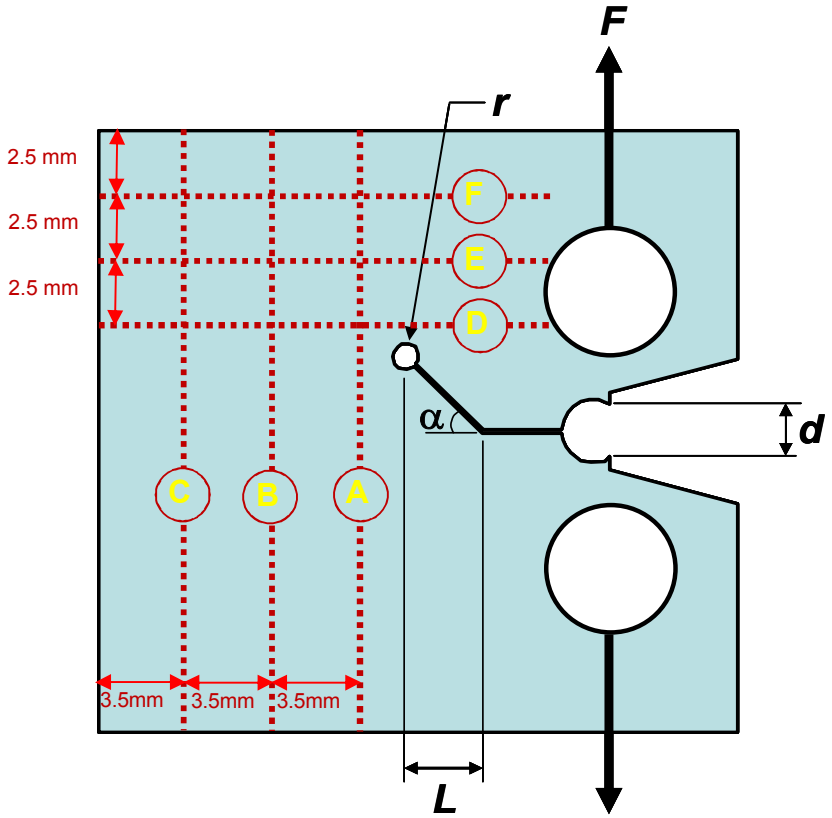
Sharp Point Contact, **Poor**



Beam and shell Edges, **OK**

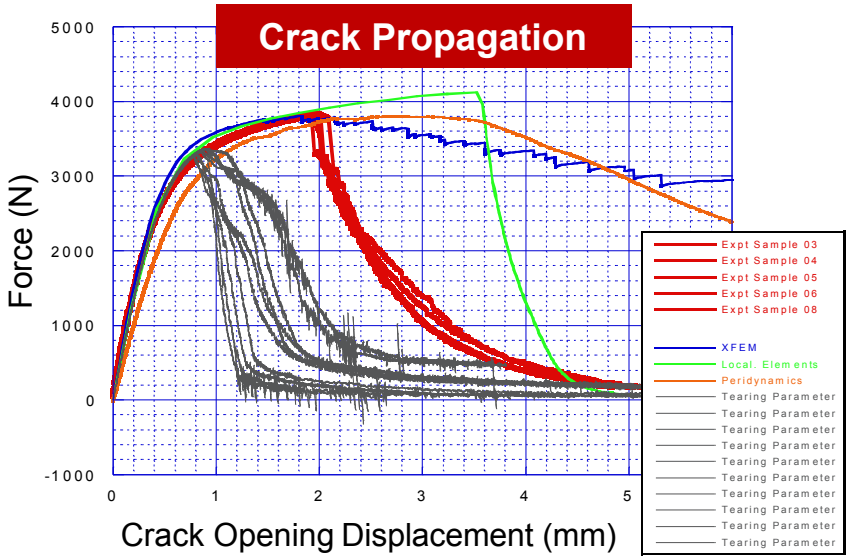
The ability to model fracture and failure remains a research challenge

- The Sandia fracture challenge shows both how far we have come and how far we have to go

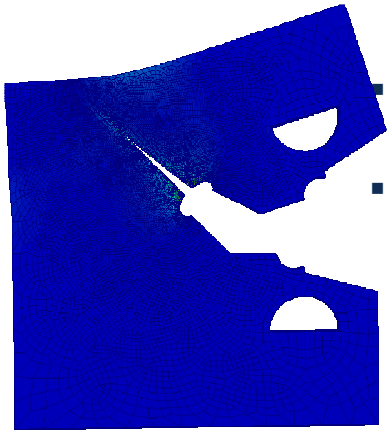


For the specimen shown below

- What is the *load-line displacement* Δd and the *peak force* prior to crack initiation?
- What is the *order of crack propagation* (e.g. A-B-D-C, etc.)?
- What is the *force* and *displacement* at which the crack reaches the 1st line?
- What is the *force* (kN) and load-line *displacement* (mm) at which the crack reaches line E?

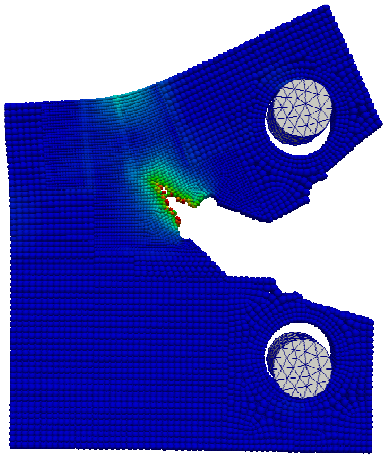


Four mathematical approaches to fracture



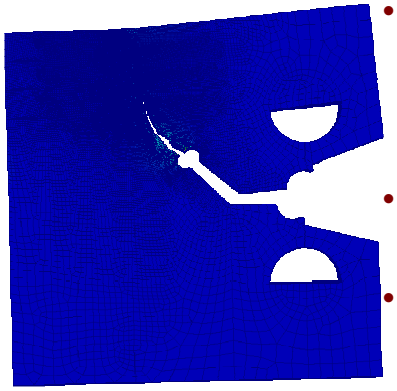
Localization Elements

- Interface elements placed in mesh *a priori*
- 3 potential crack paths competed against each other to determine most likely path



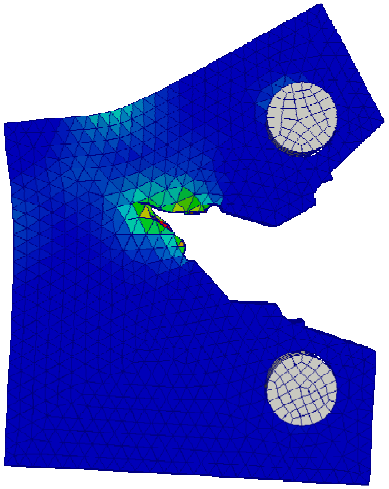
Peridynamics

- **Elastic-plastic continuum material used**
- **Bonds broken based on critical stretch**
- **Pins modeled with finite elements, interact with peridynamics via contact**



Element Death

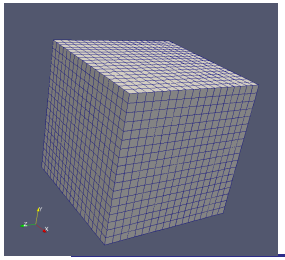
- **Elastic-plastic material with triaxial-based tearing parameter to determine failure**
- **Elements killed when failure criterion reached**
- **Used implicit solver with limited number of failed elements per iteration**



XFEM

- **XFEM in Sierra is still in development stage. This is one of first applications**
- **Elastic-plastic continuum material used, no cohesive zone law**
- **Coarse elements used for time step and because of crack tangling**

And of course, the traditional heart of any simulation is the solver.

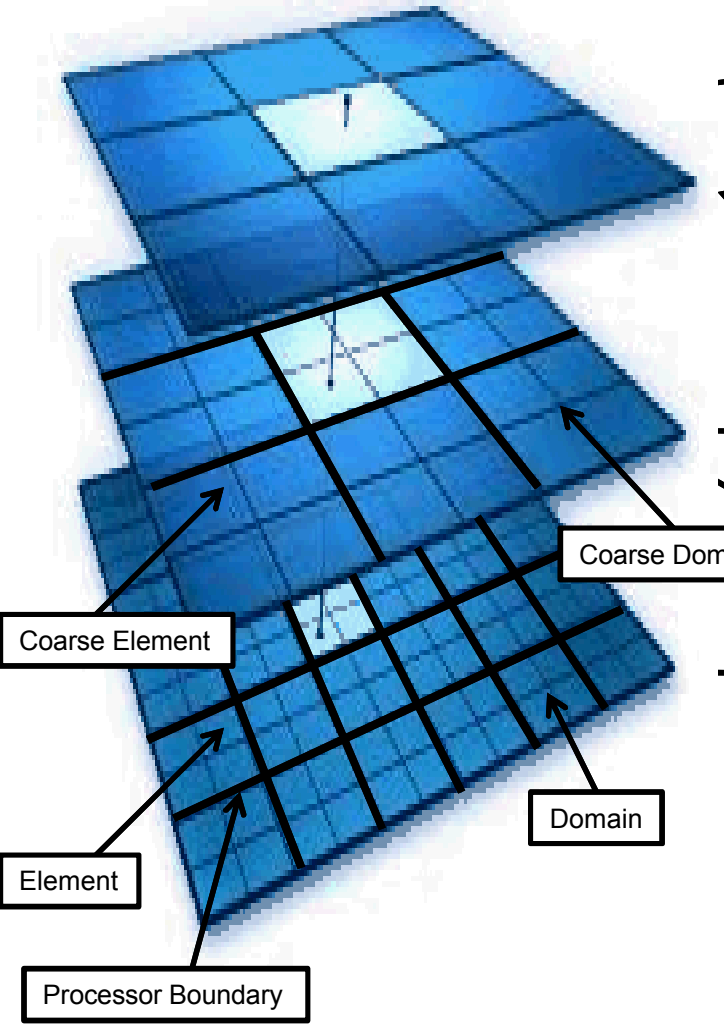


Time to solve one linear system of equations: Ax = b

Cube Mesh	Dense	Skyline	Sparse	FETI
10x10x10	5 s	1 s	1 s	1 s
15x15x15	100 s	6 s	3 s	2 s
20x20x20	1300 s	40 s	10 s	6 s
25x25x25	DNF	180 s	30 s	12 s
30x30x30	-	-	93 s	25 s
1000x1000x1000	**-**	**-**	**-**	**?**

Linear solvers dominate nonlinear implicit solution time. We must have a scaleable technology to be able to take advantage of current and next generation machines

Multilevel solvers represent a path to scalability



$$S_{cc} u_c = s_c$$ Direct on Coarsest Level

$$\begin{bmatrix} K_{rr}^1 & \cdots & 0 & K_{rc}^1 B_c^1 & B_r^1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & K_{rr}^{N_s} & K_{rc}^{N_s} B_c^{N_s} & B_r^{N_s} \\ B_c^{1^T} K_{rc}^{1^T} & \cdots & B_c^{N_s^T} K_{rc}^{N_s^T} & S_{cc} & 0 \\ B_r^1 & \cdots & B_r^{N_s} & 0 & 0 \end{bmatrix}^{LN} \begin{bmatrix} u_r^1 \\ \vdots \\ u_r^{N_s} \\ u_c \\ \lambda \end{bmatrix}^{LN} = \begin{bmatrix} f_r^1 \\ \vdots \\ f_r^{N_s} \\ s_c \\ 0 \end{bmatrix}^{LN}$$

$$\begin{bmatrix} K_{rr}^1 & \cdots & 0 & K_{rc}^1 B_c^1 & B_r^1 \\ \vdots & \ddots & \vdots & \vdots & \vdots \\ 0 & \cdots & K_{rr}^{N_s} & K_{rc}^{N_s} B_c^{N_s} & B_r^{N_s} \\ B_c^{1^T} K_{rc}^{1^T} & \cdots & B_c^{N_s^T} K_{rc}^{N_s^T} & S_{cc} & 0 \\ B_r^1 & \cdots & B_r^{N_s} & 0 & 0 \end{bmatrix}^{L1} \begin{bmatrix} u_r^1 \\ \vdots \\ u_r^{N_s} \\ u_c \\ \lambda \end{bmatrix}^{L1} = \begin{bmatrix} f_r^1 \\ \vdots \\ f_r^{N_s} \\ s_c \\ 0 \end{bmatrix}^{L1}$$

The computing itself also remains a challenge

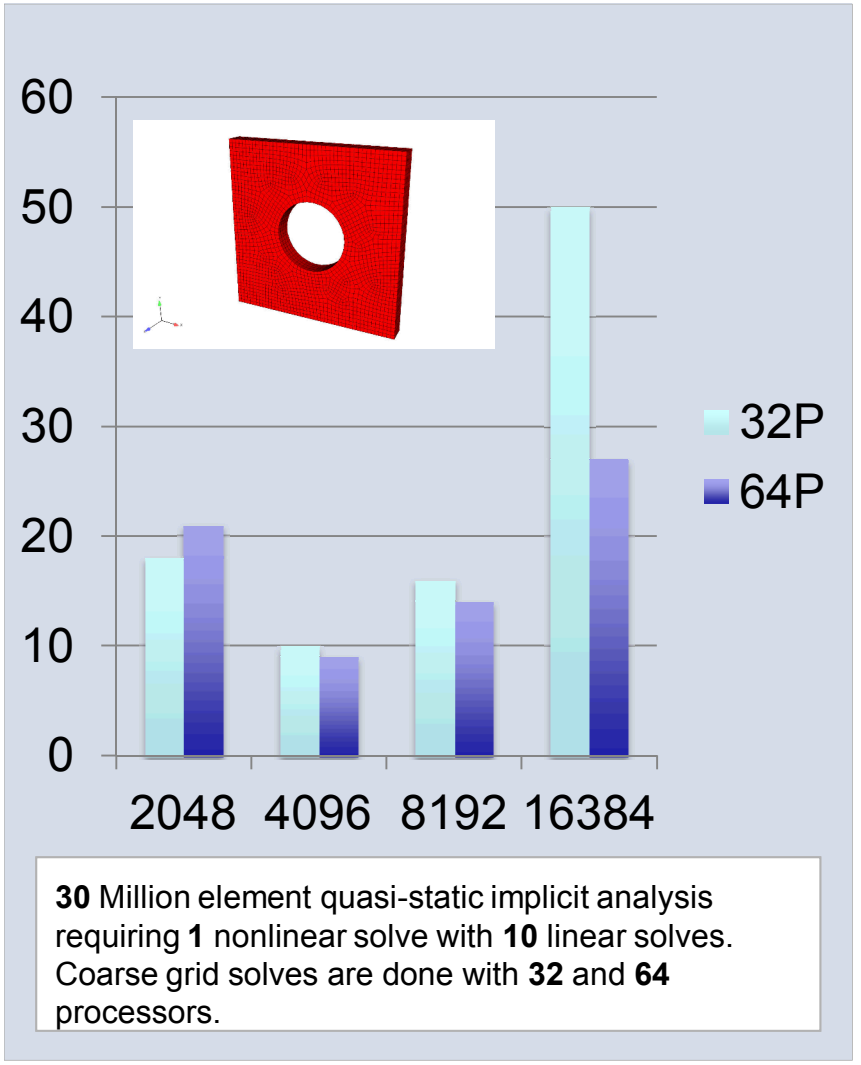
Sandia National Laboratories

- The state of the art in HPC computing is currently explicitly-programmed domain-decomposition-based parallelism. This is about to change.
- Until now, mathematicians and engineers needed little more than a passing knowledge of computer architecture, including for example,
  - What is MPI and message passing?
  - Which algorithms scale logarithmically?
  - What data structures and algorithms facilitate vectorization?
  - What is a cache and what data structures and algorithms are cache friendly?
- Soon we will need to know
  - Hierachical load balancing for heterogeneous computers
  - Resilient algorithms
  - Power management
  - More complex programming models
  - Asynchronous task-based parallelism
  - Etc.

We have to consider a solver's algorithmic and implementation scalability. What size problems are we talking about?

Sandia National Laboratories

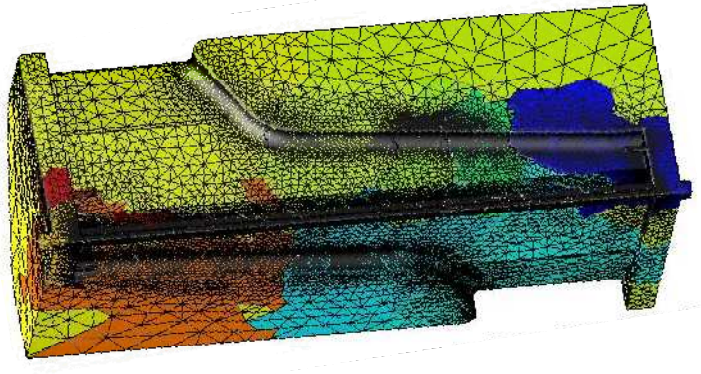
- Implicit Quasi-Statics Scalability
- Successfully ran on 16,384 processors with Implicit Nonlinear and Linear solvers, solving
- ~100 million equations
- 1 Nonlinear Solve, 10 Iterations.
- ~1 minute per linear solve.
- Scalability study identified areas of improvement in 3-Level FETI parallel iterative linear solver.
- Specifically need an automatic algorithm for coarse grid solves.



30 Million element quasi-static implicit analysis requiring **1** nonlinear solve with **10** linear solves. Coarse grid solves are done with **32** and **64** processors.

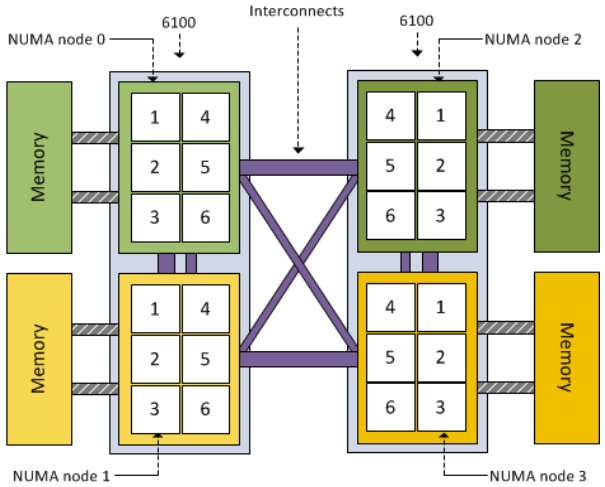
Dynamic and hierarchical load balancing

- We need a much more sophisticated approach to workload and memory management. Example drivers for load balancing
  - Hierarchical architectures and memories
  - Adaptivity
  - Localized multiscale methods
  - Contact
  - Power management



- We have implementing a "topology" parameter
  - TOPOLOGY=96 (96 cores)

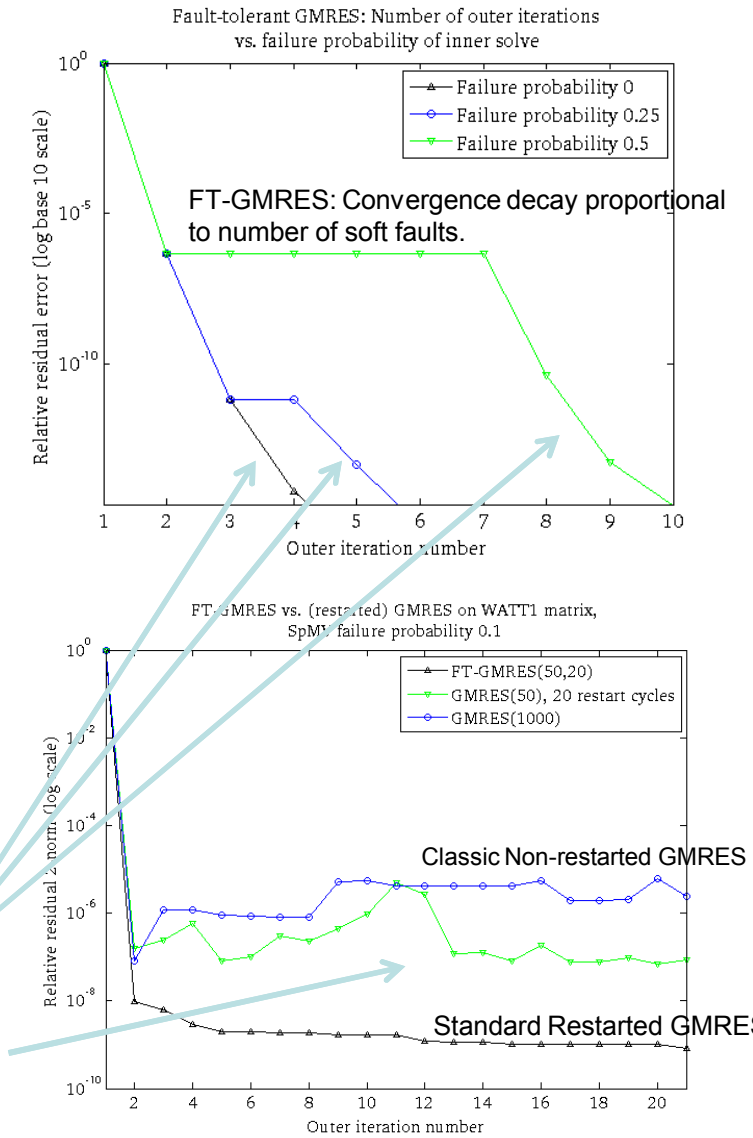
    ⋮
  - TOPOLOGY=4,2,2,6 (4 nodes, 2 sockets, 2 dies, 6 cores)
- Topology-aware load balancing has resulted in a 60% speedup in one (simple) example



http://frankdenneman.nl/2011/01/amd-magny-cours-and-esx/

Resilience and Fault Tolerance

- Exascale systems will be less reliable, and codes will be expected to take more responsibility
  - Exception handling/detect and fail
  - Checkpoint/restart
  - Redundant calculations
  - Asynchrony
  - Fault-tolerant algorithms
- Selective reliability enables new solvers
  - System exposes reliability tradeoffs
  - Algorithm identifies what *must* be reliable
- Fault-tolerant GMRES
  - Inner solver "preconditions" outer solver
  - Inner solver runs unreliably, outer solver reliably
  - Reuse any existing solver stack as "inner solver"
  - Most time spent in cheap unreliable mode
  - Faults only delay convergence; don't prevent it
  - Standard restarting GMRES (simpler approach) is not sufficient.



FT-GMRES

Embedded UQ

- Current methods are sample based and use capacity computing
- Embedding leads to
  - Better accuracy
  - Overall more efficient than ensembles
  - Reordering allows more parallelism and use of threads
- Can be coupled with traditional sample-based methods
- Steady-state stochastic problem (for simplicity):

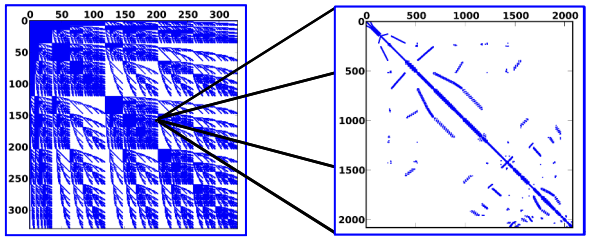
$$\text{Find } u(\xi) \text{ such that } f(u,\xi) = 0,\ \xi : \Omega \to \Gamma \subset R^M, \text{ density } \rho$$

- Stochastic Galerkin method (Ghanem and many, many others…):

$$\hat{u}(\xi) = \sum_{i=0}^{P} u_i \psi_i(\xi) \to F_i(u_0, \ldots, u_P) = \frac{1}{\langle \psi_i^2 \rangle} \int_\Gamma f(\hat{u}(y), y) \psi_i(y) \rho(y) dy = 0, \quad i = 0, \ldots, P$$

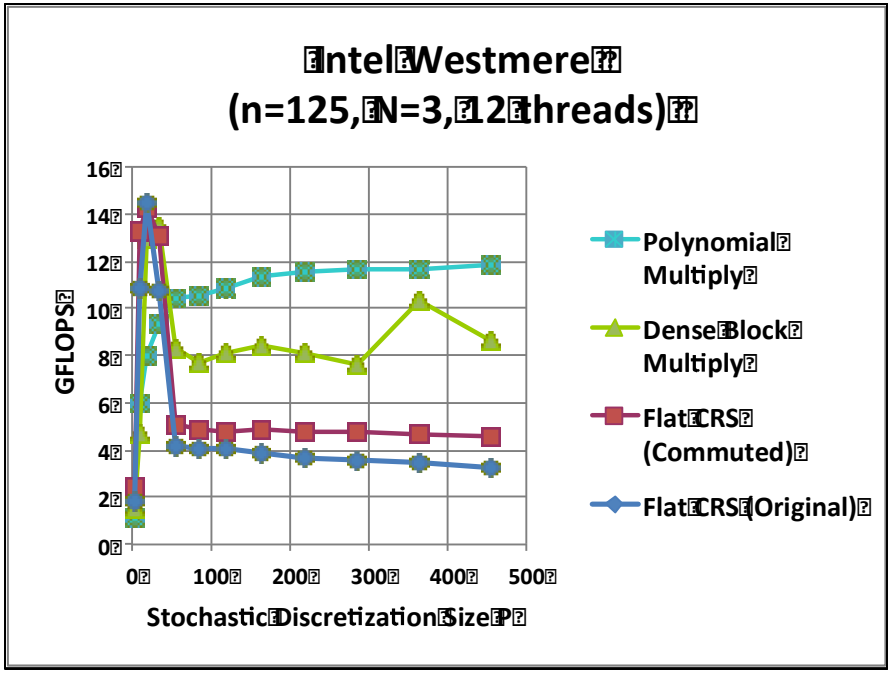
- Method generates new coupled spatial-stochastic nonlinear problem (intrusive)

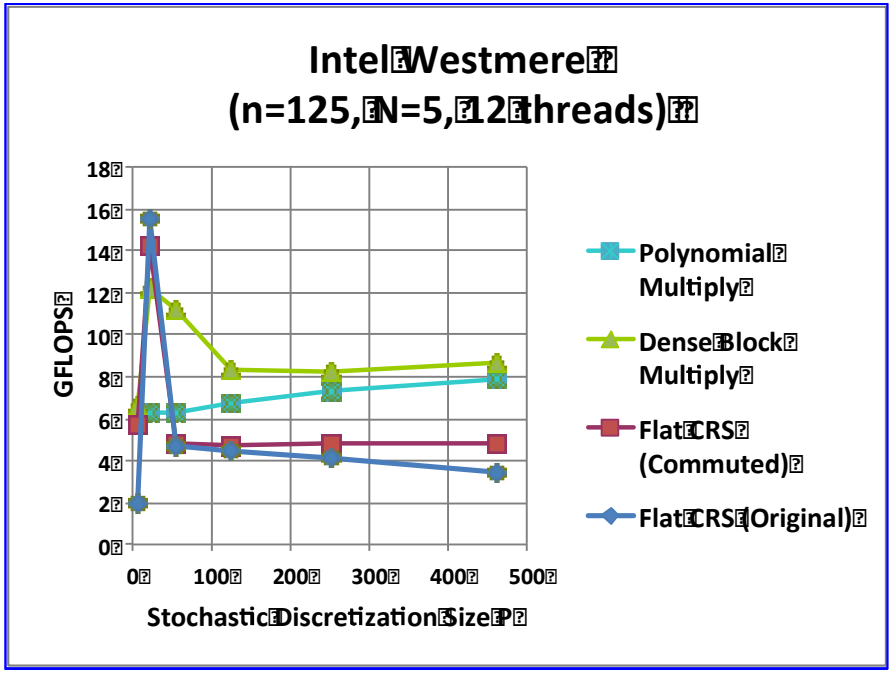
$$0 = F(U) = \begin{bmatrix} F_0 \\ F_1 \\ \vdots \\ F_P \end{bmatrix}, \quad U = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_P \end{bmatrix} \qquad \frac{\partial F}{\partial U}:$$



Stochastic sparsity Spatial sparsity

The computational advantage of moving UQ calculation to the inner loop (GFLOPS – Intel Westmere)

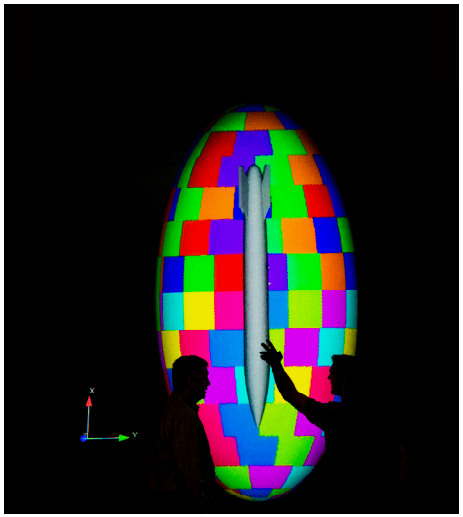




- Standard 3-D first-order FEM grid (5x5x5)
  - Small FEM size due to limited GPU memory, large usage by block and CRS approaches
- N = polynomial order (larger N, denser blocks)

And finally, there are a few potentially revolutionary advances

- Optimization and design
  - Asking a different set of questions
- Predictive simulations
  - Coupled scales
  - reduced-order models and surrogates
- Full life cycle (cradle-to-grave) engineering
  - Environmental specification for design
  - Design
  - Qualification and testing
  - Manufacturing and infrastructure
  - Surveillance and maintenance
  - Decommissioning



Sandia National Laboratories

Long pause ------- deep breath

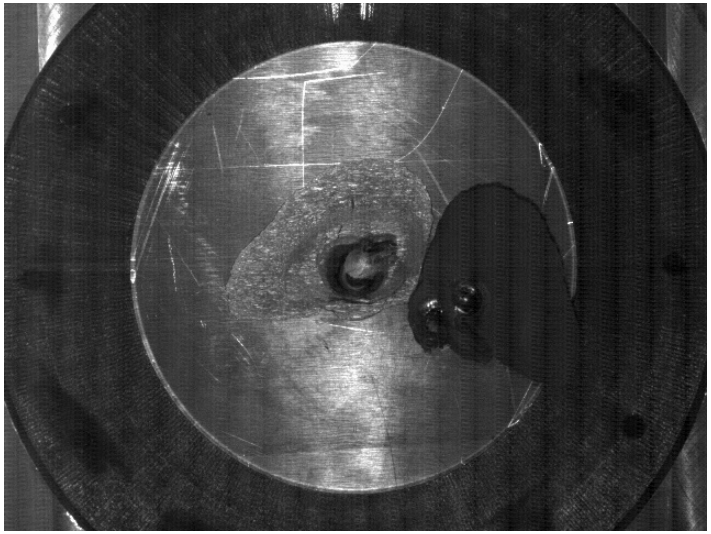
We know the physics and the algorithms. And we have a machine to run on. But is it correct?

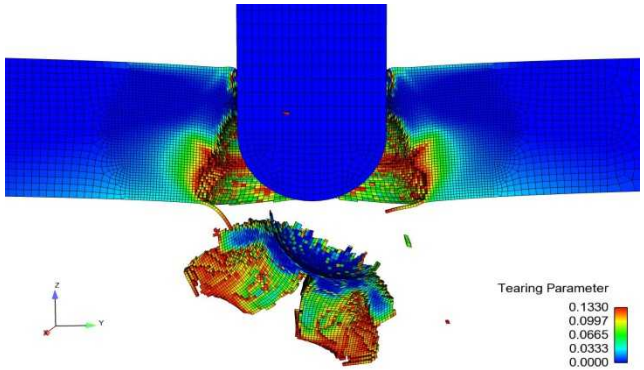
To answer that question is a long process

- Software verification and testing
- Solution verification
- Uncertainty quantification, sensitivities, error estimations and quantified margins
- Validation, including experiments and simulations

V&V process and methods were tested in a recent L2 milestone.

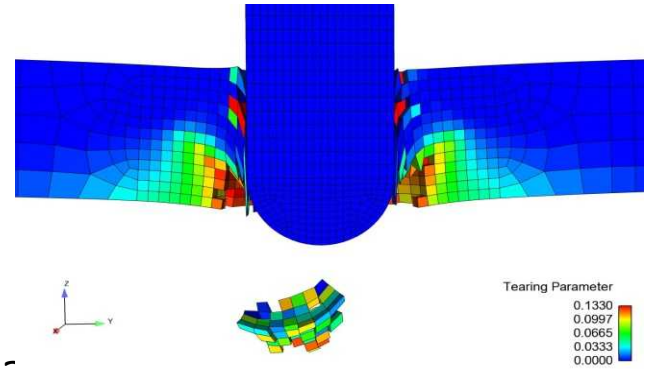
- **Objective:** Assess predictions of the minimum penetration velocity using Sierra/SM w/tearing parameter as the failure criterion by comparing to test data.





Fine Mesh
82.5 hrs/48 proc

Coarse Mesh
9.5 hrs/4 proc

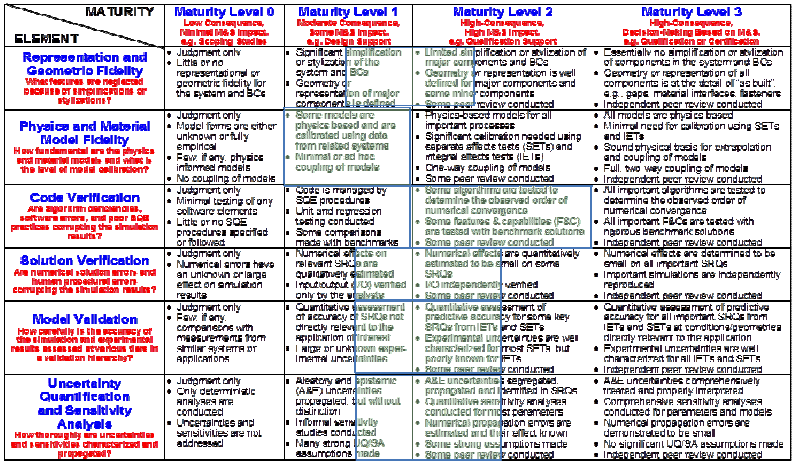


- **More questions:** Typically, we can only afford to use a "coarse" mesh when many function evaluations are involved! Are these models still valuable? Why do we believe in them? What error do they carry with them?

The Milestone included verification activities.

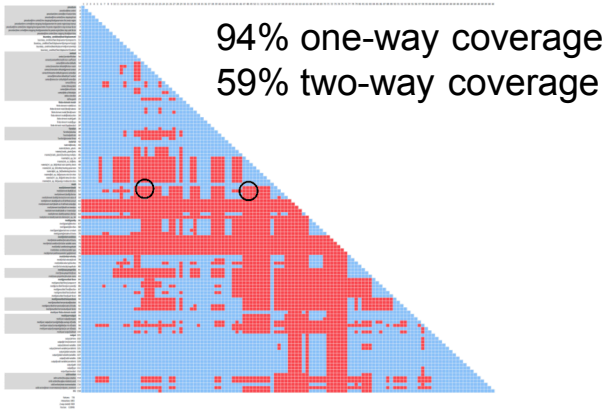
- Credibility Assessments:

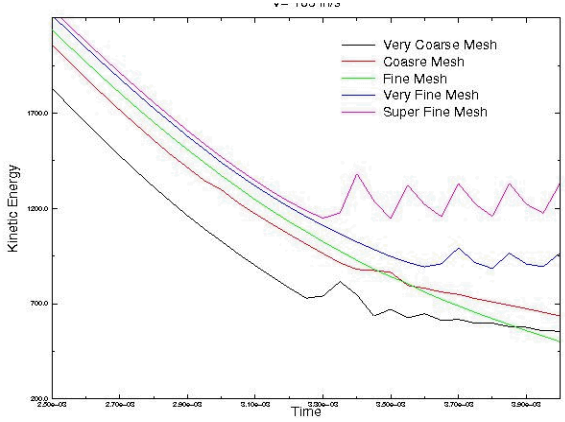
Phenomena	Consensus Importance	Adequacy: Math Model	Adequacy: Sierra/SM Code	Adequacy: Validation
Large elastic-plastic deformation of metals	H	H	M	M
Ductile material failure	H	M	M	L
Contact	H	H	M	M
Friction between punch and test item	M	M	M	L
Enforcement of boundary conditions	L	H	H	L
Inertial loads	H	H	H	M



MATURITY / ELEMENT	Maturity Level 0 Low Consequence, Minimal M&S Impact, e.g. Scoping Studies	Maturity Level 1 Moderate Consequence, Some M&S Impact, e.g. Design Support	Maturity Level 2 High-Consequence, High M&S Impact, e.g. Qualification Support	Maturity Level 3 High-Consequence, Decision-Making Based on M&S, e.g. Qualification or Certification
Representation and Geometric Fidelity What features are neglected because of simplifications or stylizations?	• Judgment only • Little or no representation or geometric fidelity for the system and BCs	• Significant simplification or stylization of the system and BCs • Geometry or representation of major components is defined	• Limited simplification or stylization of major components and BCs • Geometry or representation is well defined for major components and some minor components • Some peer review conducted	• Essentially no simplification or stylization of components in the system and BCs • Geometry or representation of all components is at the detail of 'as built', e.g., gaps, material interfaces, fasteners • Independent peer review conducted
Physics and Material Model Fidelity How fundamental are the physics and material models and what is the level of model calibration?	• Judgment only • Model forms are either unknown or fully empirical • Few, if any, physics informed models • No coupling of models	• Some models are physics based and are calibrated using data from related systems • Minimal or ad hoc coupling of models	• Physics-based models for all important processes • Significant calibration needed using separate effects tests (SETs) and integral effects tests (IETs) • One-way coupling of models • Some peer review conducted	• All models are physics based • Minimal need for calibration using SETs and IETs • Sound physical basis for extrapolation and coupling of models • Full, two way coupling of models • Independent peer review conducted
Code Verification Are algorithm deficiencies, software errors, and poor SQE practices corrupting the simulation results?	• Judgment only • Minimal testing of any software elements • Little or no SQE procedures specified or followed	• Code is managed by SQE procedures • Unit and regression testing conducted • Some comparisons made with benchmarks	• Some algorithms are tested to determine the observed order of numerical convergence • Some features & capabilities (F&C) are tested with benchmark solutions • Some peer review conducted	• All important algorithms are tested to determine the observed order of numerical convergence • All important F&Cs are tested with rigorous benchmark solutions • Independent peer review conducted
Solution Verification Are numerical solution errors and human procedural error corrupting the simulation results?	• Judgment only • Numerical errors have an unknown or large effect on simulation results	• Numerical effects on relevant SRQs are qualitatively estimated • Input/output (I/O) verified only by the analysts	• Numerical effects are quantitatively estimated to be small on some SRQs • I/O independently verified • Some peer review conducted	• Numerical effects are determined to be small on all important SRQs • Important simulations are independently reproduced • Independent peer review conducted
Model Validation How carefully is the accuracy of the simulation and experimental results assessed over those that is a validation hierarchy?	• Judgment only • Few, if any, comparisons with measurements from similar systems or applications	• Quantitative assessment of accuracy of SRQs not directly relevant to the application of interest • Large or unknown experimental uncertainties	• Quantitative assessment of predictive accuracy for some key SRQs from IETs and SETs • Experimental uncertainties are well characterized for most SRQs but poorly known for IETs • Some peer review conducted	• Quantitative assessment of predictive accuracy for all important SRQs from IETs and SETs at conditions/geometries directly relevant to the application • Experimental uncertainties are well characterized for all IETs and SETs • Independent peer review conducted
Uncertainty Quantification and Sensitivity Analysis How thoroughly are uncertainties and sensitivities characterized and propagated?	• Judgment only • Only deterministic analyses are conducted • Uncertainties and sensitivities are not addressed	• Aleatory and epistemic (AIE) uncertainties propagated, but without distinction • Informal sensitivity studies conducted • Many strong UQ/SA assumptions made	• AIE uncertainties segregated propagated and identified in SRQs • Quantitative sensitivity analyses conducted for most parameters • Numerical propagation errors are estimated and their effect known • Some strong assumptions made • Some peer review conducted	• AIE uncertainties comprehensively treated and properly interpreted • Comprehensive sensitivity analyses conducted for parameters and models • Numerical propagation errors are demonstrated to be small • No significant UQ/SA assumptions made • Independent peer review conducted

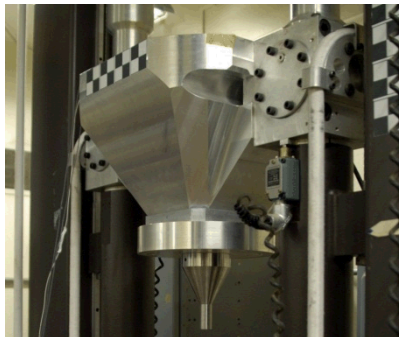
- Code/Solution Verification:





Evaluate mesh convergence rates before convergent behavior degrades

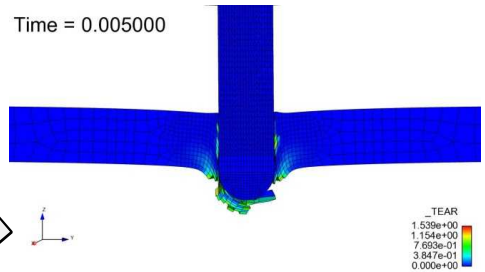
It also included validation activities.

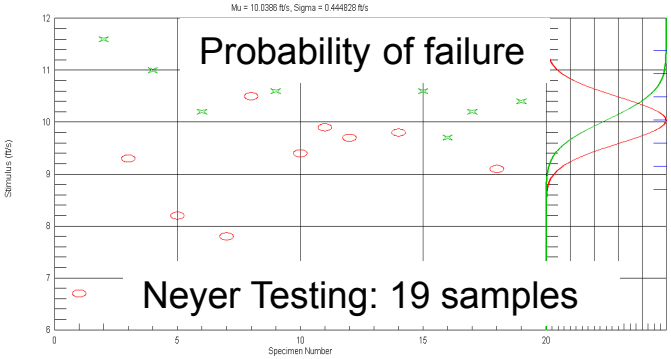


Experimental Uncertainties

- Bungee force
- Friction (punch and plate/tube)
- Velocity measurement
- Material variability-> characterization process

Sensitivity analysis helped identify important factors → reduce scope



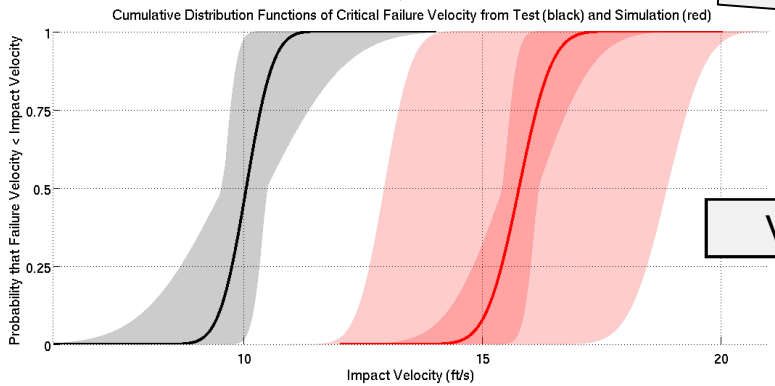


Uncertainties

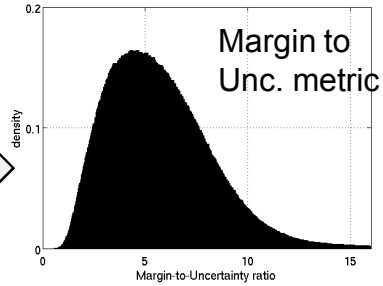
Uncertainties

Numerical Uncertainties

- Discretization Error
- Algorithmic parameters



Validation metrics



Summary

- Computational simulation may not replace experiment, but it has fundamentally changed the way we approach engineering.
- It has the potential to revolutionize engineering
- Mathematics and computer science (whether taught in traditional Mathematics and C.S. departments or taught in engineering programs) are critical to the success of computational simulation.
- We have moved well beyond the physical regimes in which engineering intuition is sufficient.
- The complexity of the problem (physics, engineered systems, algorithms, code, architectures) required an integrated multidisciplinary approach