SAND2011-8437 C
SAND2011-8437C

Lowering the HPC Barrier – Development and Usability

Micheal W. Glass
Sandia National Laboratories

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.

NNSA National Nuclear Security Administration

Sandia National Laboratories

From the Perspective of the Sierra Project at Sandia National Laboratories

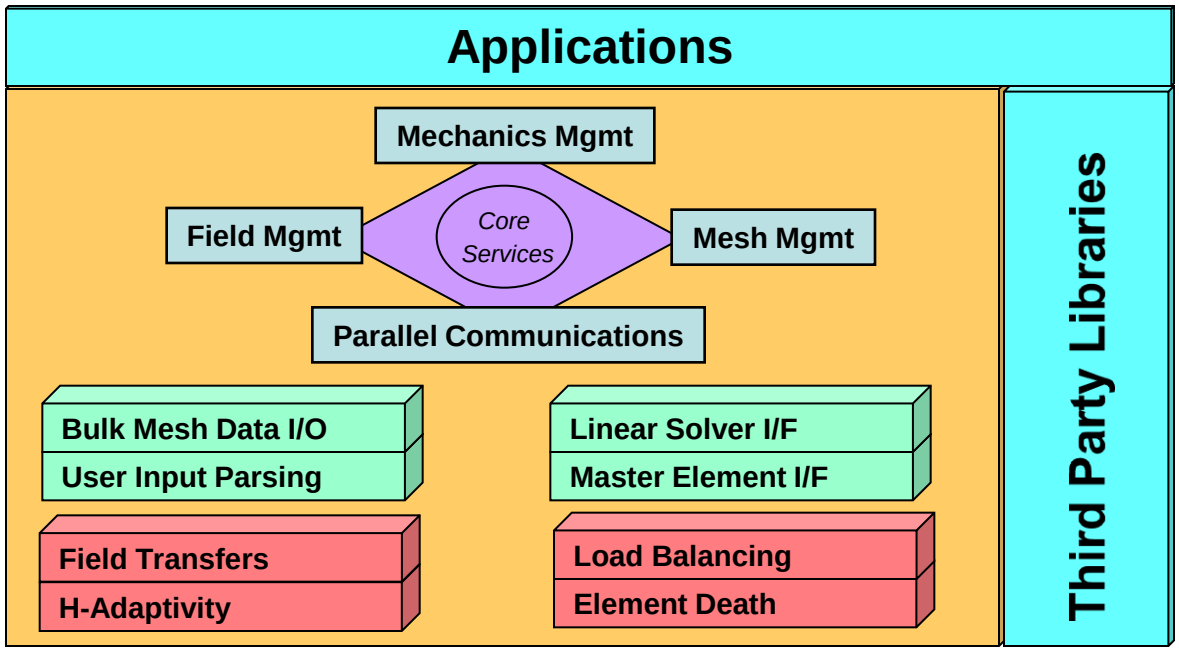
- **How do we make HPC modeling and simulation a larger part of our engineering processes?**
  - **From software development perspective**
    - Sustained Innovative Development
      - Software capabilities
      - Physics capabilities
      - Application domains
  - **From an end-user usability perspective**
    - Ease of use
      - Installation
      - usability

Background: What is Sierra?

- A DOE/NNSA funded ASC project – about 15 years
- An engineering mechanics simulation code suite
  - Solid mechanics
  - Structural dynamics
  - Thermal Mechanics
  - Low mach fluid mechanics
  - High mach fluid mechanics
- massively parallel
- multi-physics coupling
  - Between various modules
  - With other non-sierra applications

Sierra Architecture Overview

- **Sierra applications are built on a common framework.**
- **The framework provides a common set of HPC services for applications.**



Sierra Framework Issues

- **No use outside of Sierra. Why?**
  - **Very heavy-weight with a steep learning curve.**
  - **Has a bloated API and a complex dependency graph.**
  - **Restrictive application architecture.**
  - **All or nothing nature of the framework.**

- **Internally to Sierra**
  - **The tightly coupled Framework is increasingly limiting as new demands continue to be placed on it (new codes, new physics, etc).**
  - **Software design viewed as a barrier to supporting next generation platforms**

Frameworks vs. Toolkits

We decided a few years ago that transitioning from a Sierra Framework to a Sierra Toolkit (STK) would be in the best interests of the project.

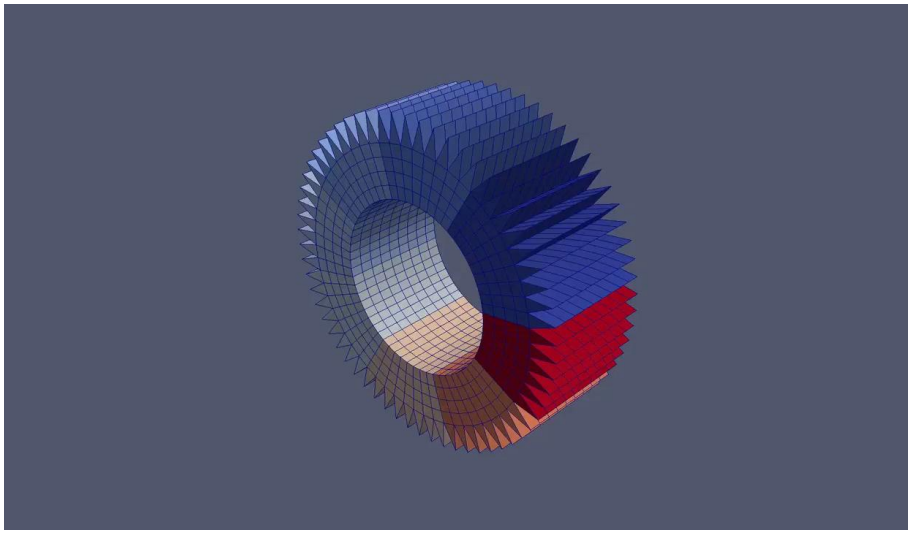
- **Frameworks**
  - **emphasize design reuse**
  - **dictate the architecture of an application**
- **Toolkits**
  - **emphasize code reuse**
  - **supplies functionality for an application**
- **You commit to the design when you use a framework, whereas you commit to a set of functionality when you use a toolkit.**

Sierra Toolkit Vision and Goals

- A broader presence within the HPC community
- Be a collaboration vehicle for algorithm and model development
  - Both internally and externally
  - Enable a faster R-D-A capability migration
- Support a broader application space
- Improve the agility and robustness of Sierra
  - Improved testability - unit, use-case, and performance
  - Pay attention to APIs and implementations
  - Provide some high-level support for hybrid parallel programming models

Sierra Toolkit Highlights

- Currently have 12 modules in the toolkit
- stk_mesh reaching high-level of maturity and robustness.
  - 90% line coverage from unit testing
  - integrated functionality tested with use cases
- Parallel dynamic mesh capability
  - Basic mesh modification and skinning
  - New element death algorithm implementation shows a ~3X performance improvement over the Framework algorithm



Sierra Toolkit Highlights

- **Integrated into Sandia's Trilinos software project.**
  - **Makes the Sierra Toolkit available as open source**
  - **Makes the Sierra Toolkit available to a broader audience and complements the capabilities within Trilinos to provide a very capable open source HPC development toolkit**
  - **Automated nightly integration between Sierra and Trilinos**
    - Sierra Toolkit development occurs within the Sierra development environment
    - Makes new developments in Sierra Toolkit and Trilinos immediately available to both development communities
- **Tutorial at SIAM CS&E Conference (3/2011).**

Sierra Toolkit Highlights

- **Several Sandia projects are currently using or plan to use the Sierra Toolkit**
  - **Production applications**
  - **Research applications**
- **Users outside of Sandia are using the Sierra Toolkit**
  - **For research applications**
  - **For production applications**
  - **For collaborating with Sierra**
  - **Universities and other DoE/DoD entities**
- **Migration of Sierra to the Toolkit is underway.**

Sierra Toolkit Future Plans

- Continue the Sierra migration
- Improve the maturity of existing modules
- Continue collaboration with other SNL projects to improve the R->D->A technology transition
  - Next generation platforms
  - New physics capabilities
- Explore supporting alternate data storage and access implementations behind the same API
- Expand usage internally and externally

Sierra Usability

- **End-User usability historically has not been a strength for DoE physics and engineering software.**
  - **More emphasis on capability than usability**
  - **End-users at Sandia preferred text editors and input files**
- **But...**
  - **End-users were typically a small group of very experienced analysts**
  - **The vision is to have a broader impact and interject modeling and simulation sooner in the design process.**
    - These end-users have a different level of expertise and expectation
- **We are putting more emphasis on documentation and training**
- **Moving to a GUI front-end to the applications**

Desired UI Components

Geometry
- Creation
- Import/Export
- Decomposition
- Cleanup
- Editing
- Tolerances

Simulation Data Management
- Model & results database
- Work groups
- Security & access controls

UQ & Optimization
- Parameter estimation
- Sensitivity/variance

Post-Processing
- Local
- Remote
- On-the fly

Meshing
- Generation
- Import/Export
- Cleanup
- Quality statistics

Model Attributes & Management
- Model assembly
- Define surfaces & sets
- Input definition & validation
- Material database access

Job Submission & Monitoring
- Local resources
- Remote resources
- Status
- Notification
- Auto-restart

Sandia Has Many of the Components Today

CUBIT
- Creation
- Import/Export
- Decomposition
- Cleanup
- Editing
- Tolerances

DART WORKBENCH
- Model & results database
- Work groups
- Security & access controls

DAKOTA
- Parameter estimation
- Sensitivity/variance

PARAVIEW
- Local
- Remote
- On-the fly

CUBIT
- Generation
- Import/Export
- Cleanup
- Quality statistics

DART WORKBENCH
- Model assembly
- Define surfaces & sets
- Input definition & validation
- Material database access

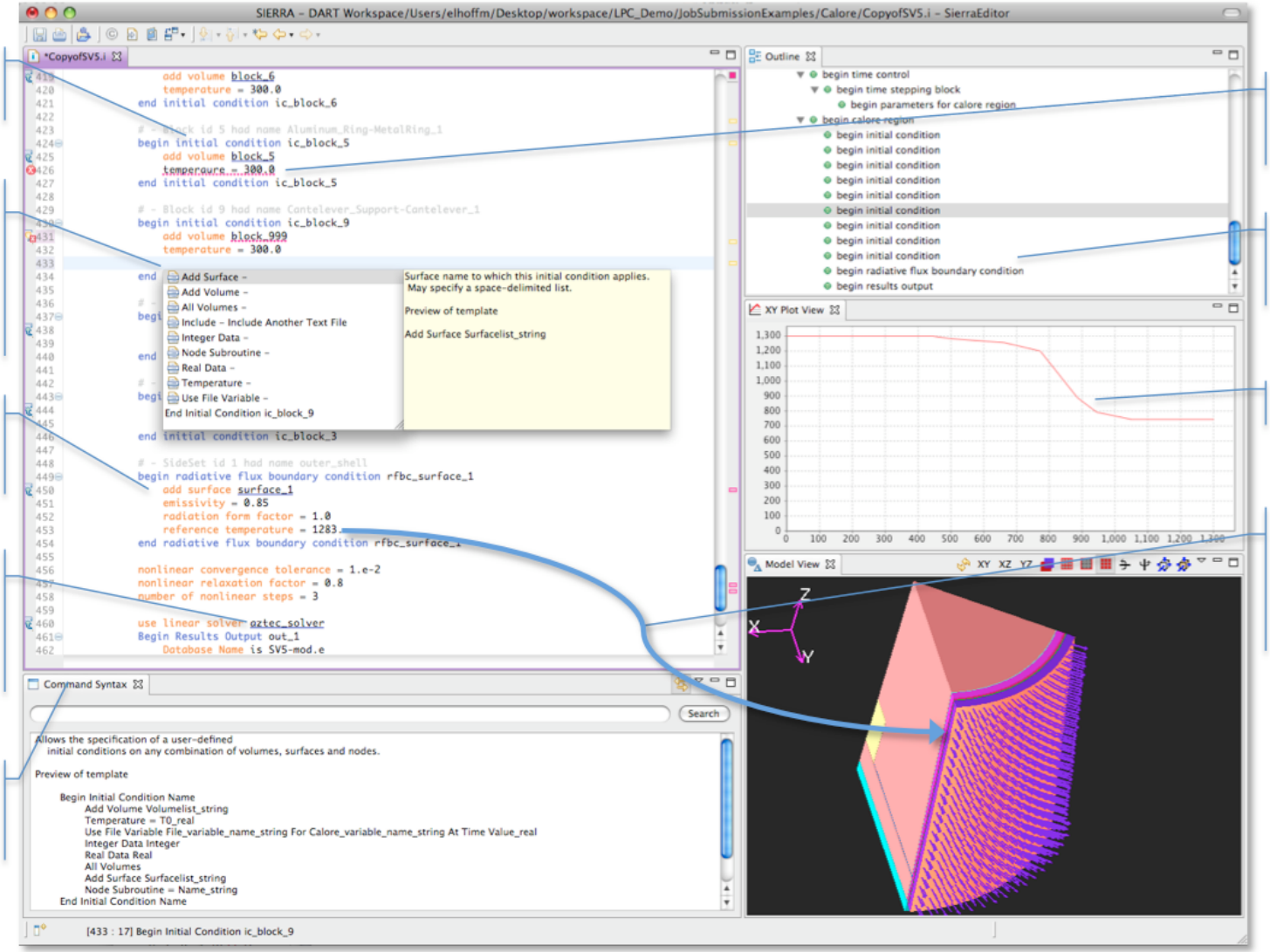
DART WORKBENCH
- Local resources
- Remote resources
- Status
- Notification
- Auto-restart

How Do We Integrate Everything Together?

- **Build on the experience and success of the DART Workbench project**
  - **Open architecture and sustainability**
  - **Based on Eclipse framework**

Phase I – The Sierra Editor

Sierra Editor Overview



Color Coded Syntax

Context Aware Syntax Completion & Dynamic Help

Automatic Formatting/ Indentation

Hyperlink to references (functions, solvers, blocks, etc)

Context Aware Syntax Guide

Dynamic Validation with Error Marking

Outline View Linked to Input Deck

Function Plots

Visual Verification of Geometry (blocks, side-sets, node-sets)

Phase II – Initial Integration

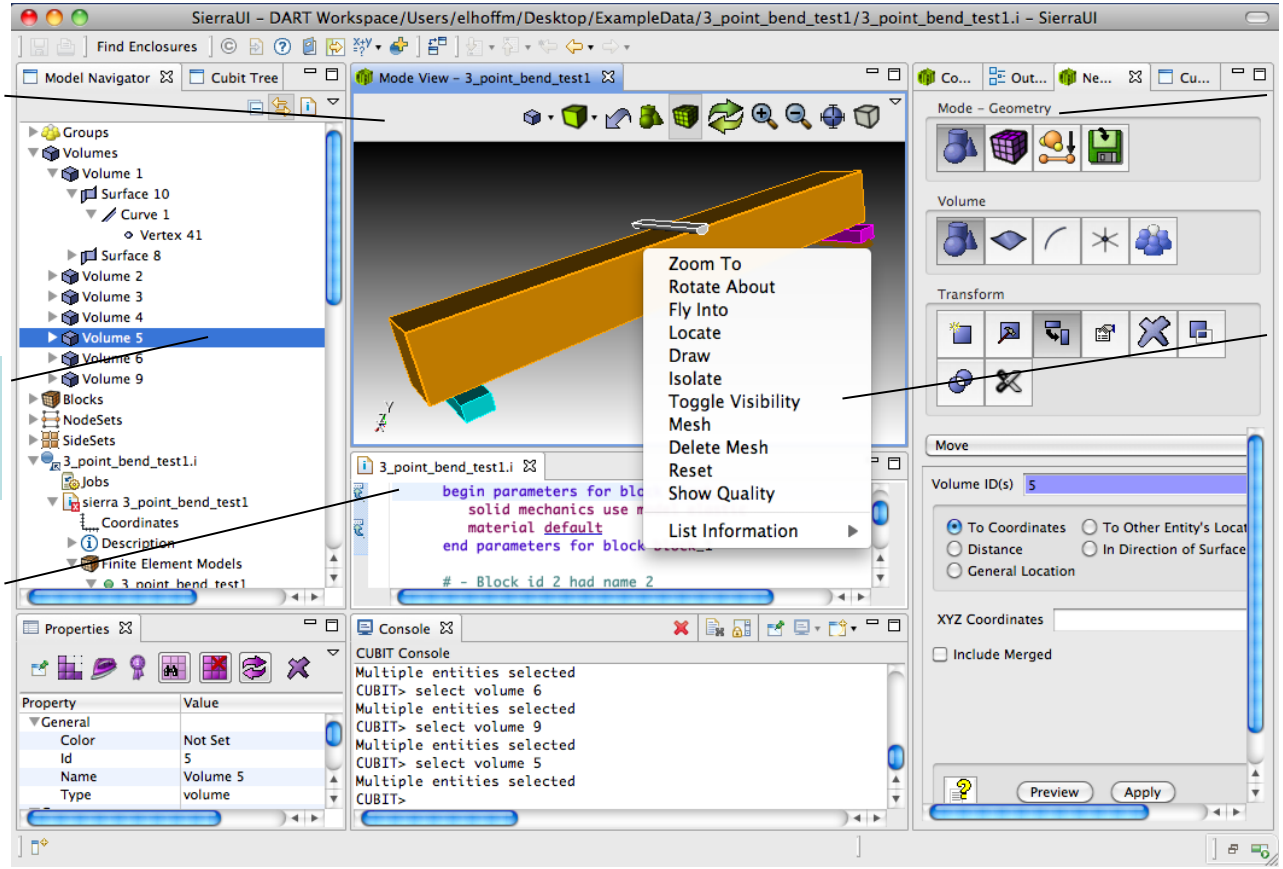
- **This effort is currently under development**
  - **Initial release planned for April 2012.**
- **Integrate more components**
  - **Cubit: geometry and meshing**
  - **Sierra Editor: Model setup**
  - **Job Submission**
- **Model tree navigation**

Fully Integrated Sierra Editor + Geometry Creation and Meshing Tools (Cubit)

Integrated VTK Geometry and Mesh

Unified Model Tree

Sierra Input Deck (Text and Tree)



Command Panel

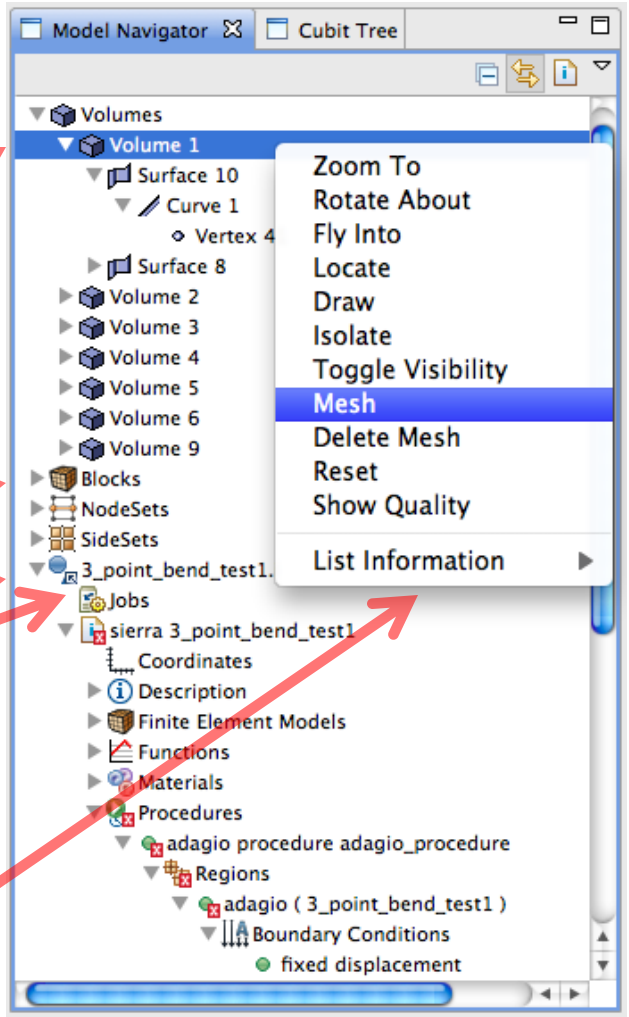
Contextual Menus

Fully Integrated Model Tree

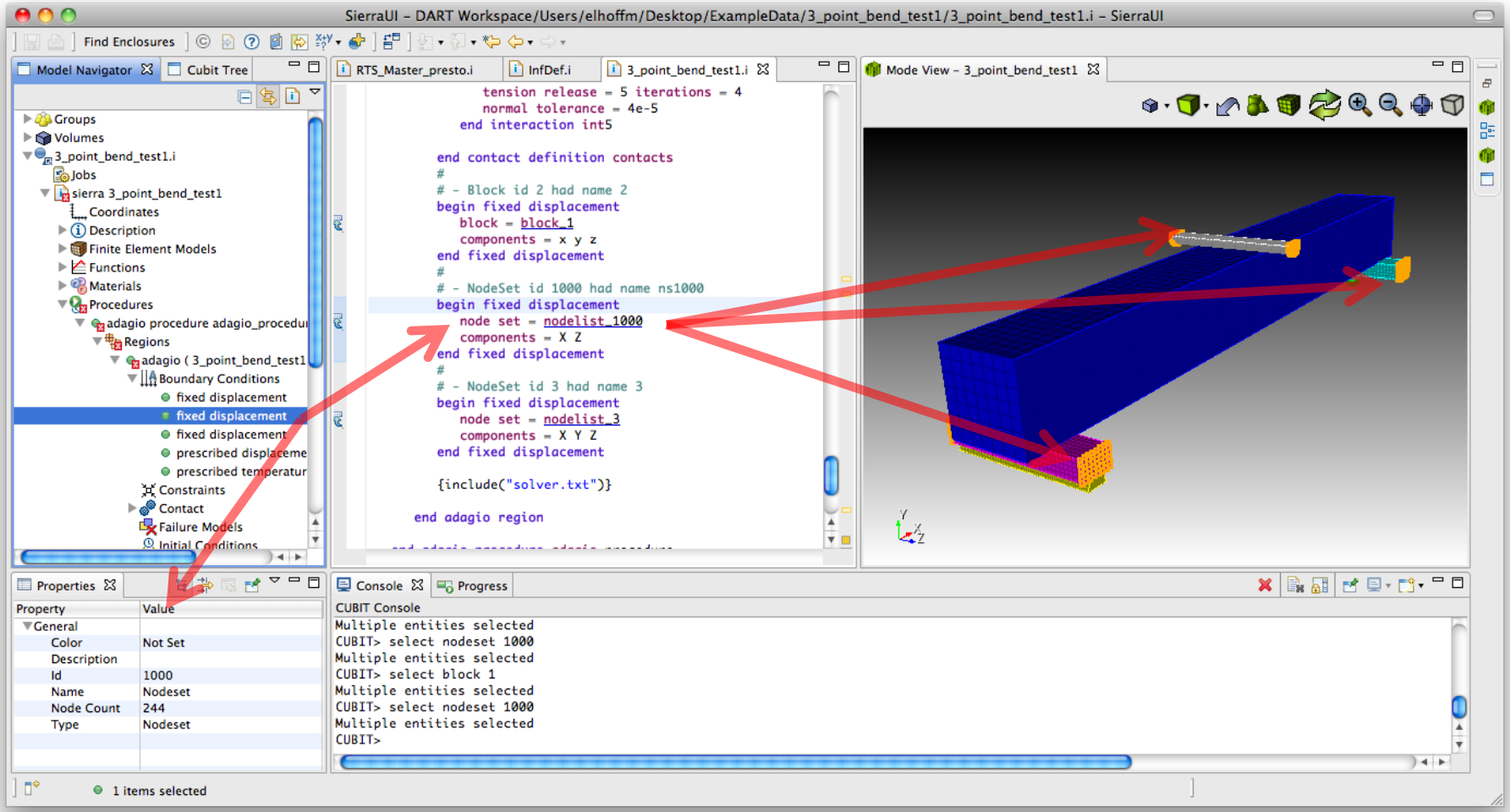
Presentation of distributed data (multiple files on multiple file systems) in a unified model-centric tree for content creation and editing of:

- **Geometry**
  - **Fully integrated geometry data and tools for geometry processing**
- **Mesh**
  - **Fully integrated meshing tools**
- **Input Deck**
  - **Tree presentation of Sierra syntax**
- **Job Submissions**
  - **All job data/metadata for jobs submitted locally and on HPC platforms**

With contextual menus present relevant actions for selected entities



Fully Synchronized Contextual Linking

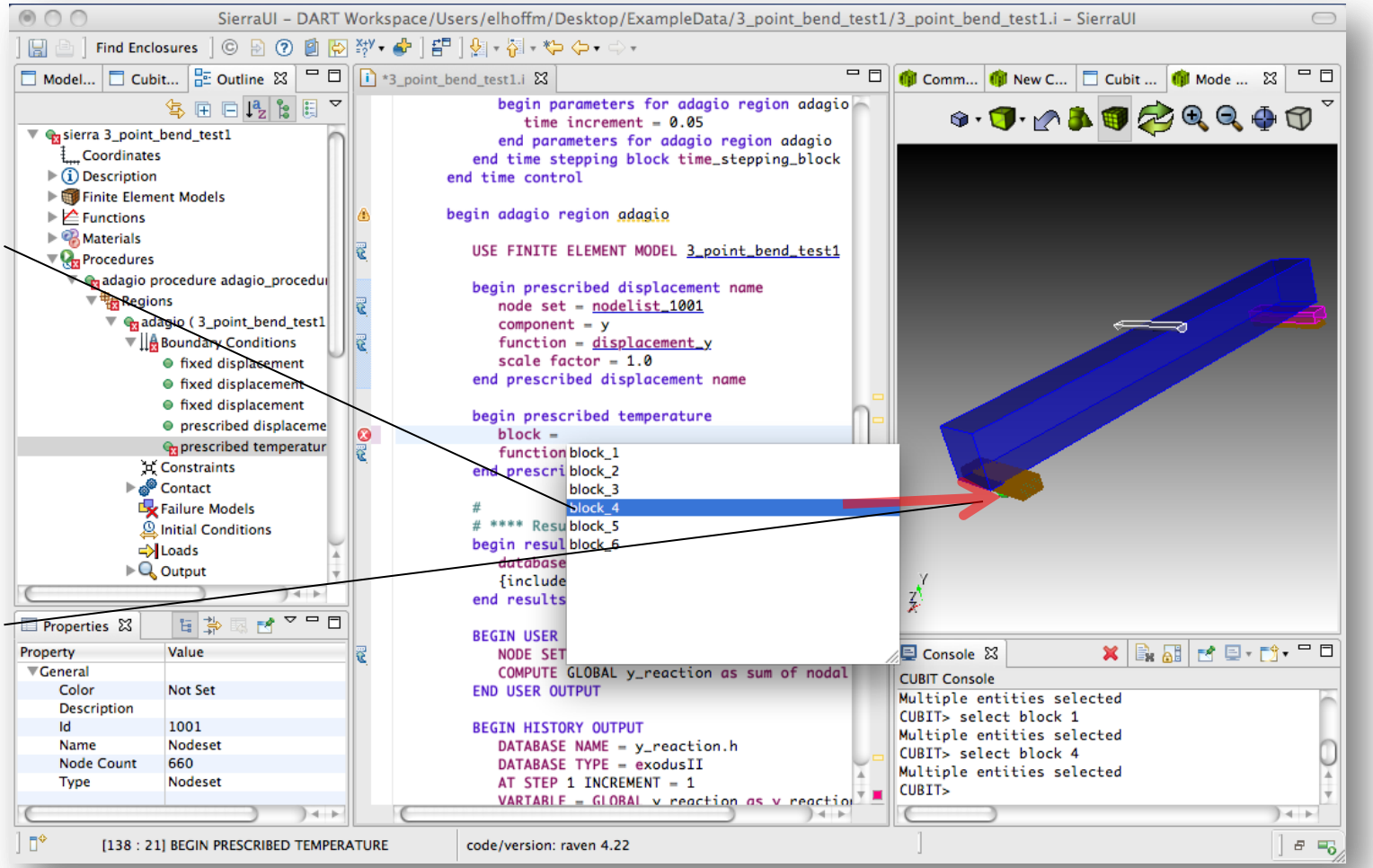


- User selection in the tree highlights input deck text, corresponding mesh entities in the graphics view, and metadata

Sierra UI Supports Traditional Text Based Data Entry

Syntax completion of block list

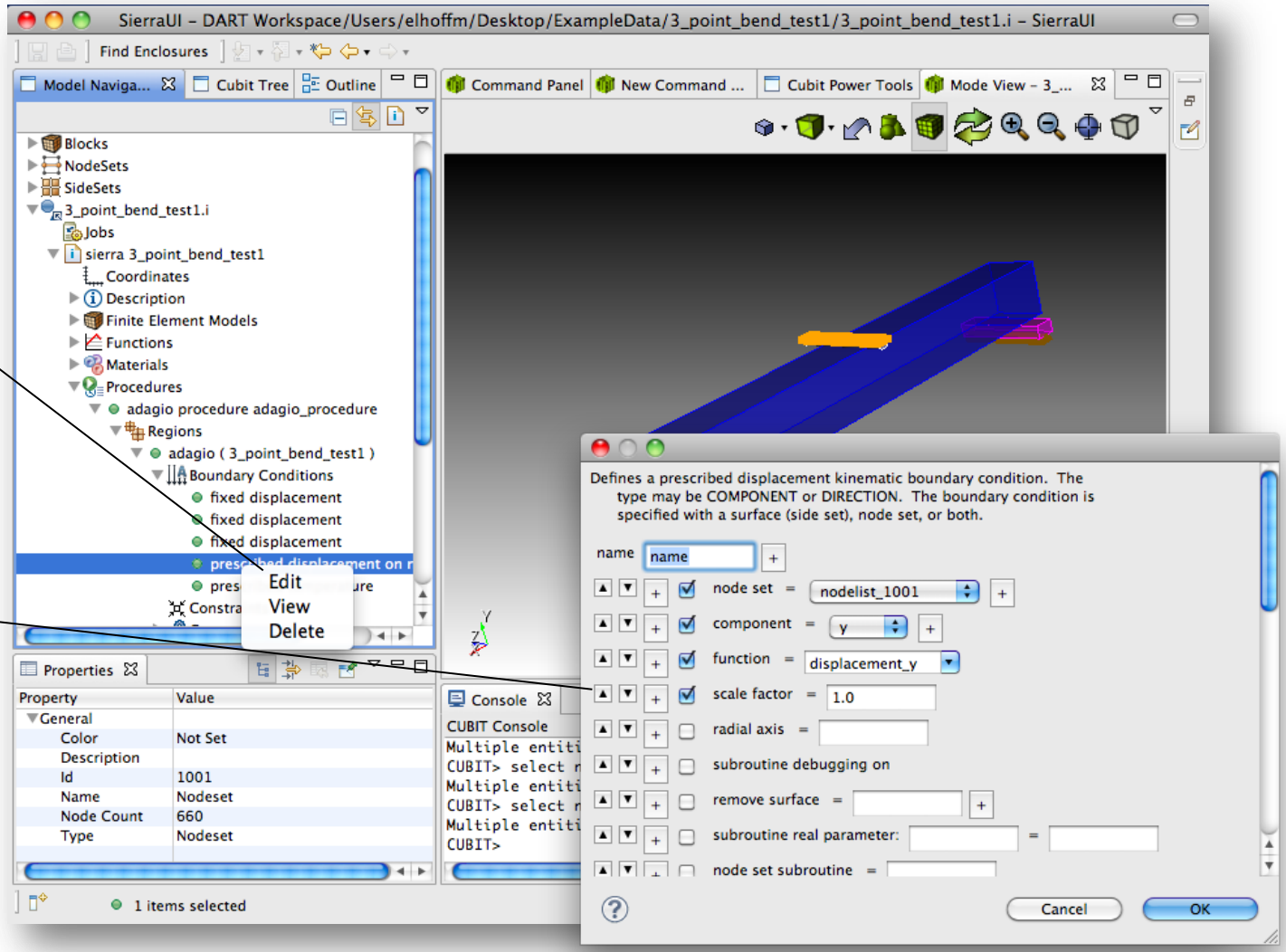
Visual linking when user mouses over list entries



Sierra UI Supports Tree-Based Data Entry

Contextual Menu for Tree-Based Editing of Input Deck

Data Entry Dialogs Auto-Generated from Sierra XML

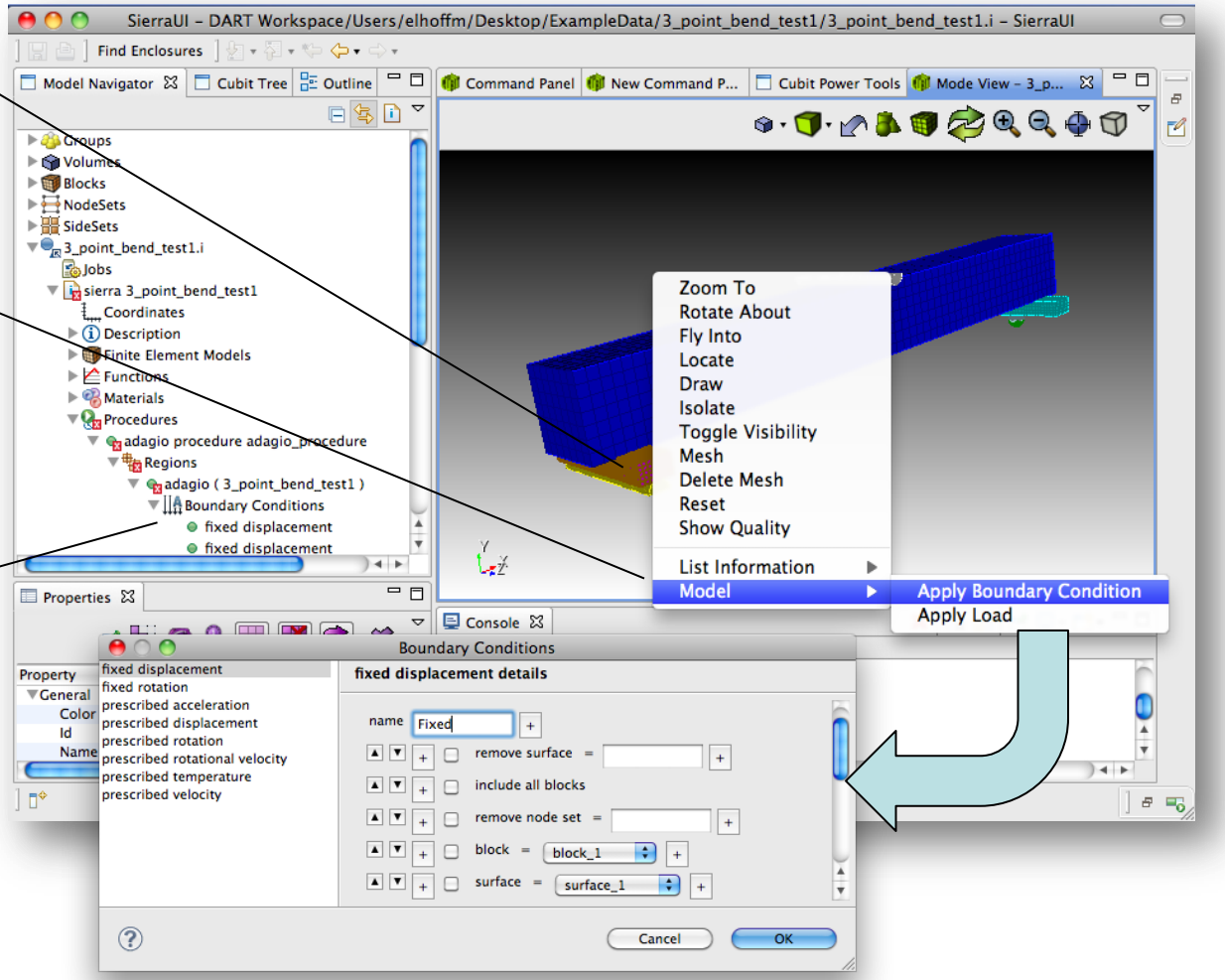


Ultimately, Sierra UI will Support Graphics-Based Data Entry

User Visually Selects Entity of Interest

Contextual Menu in Graphics Viewer

Tree and Input Deck Fully Synchronized (no need to see input deck)



Future Plans

- **Integrate Paraview based post-processing capability**
- **Integrate Dakota based UQ & Optimization capability**
- **Develop other application interfaces**

Acknowledgements

- Sierra Toolkit Team
- CUBIT Team
- DART Workbench Team