

# ENGINEERING SCIENCES WORKFLOWS FOR V&V APPLICATIONS

Robert L. Clay

Ernest J. Friedman-Hill

Edward L. Hoffman

*[rlclay|ejfried|elhoffm]@sandia.gov*

*Sandia National Laboratories*

# The Sandia Analysis Workbench

An integrated environment for engineering analysis

The screenshot displays the Sandia Analysis Workbench interface. The main window is titled "SDM - JoeDemo2/Files/3\_point\_bend\_test1.i - DART Workbench". The interface is divided into several panes:

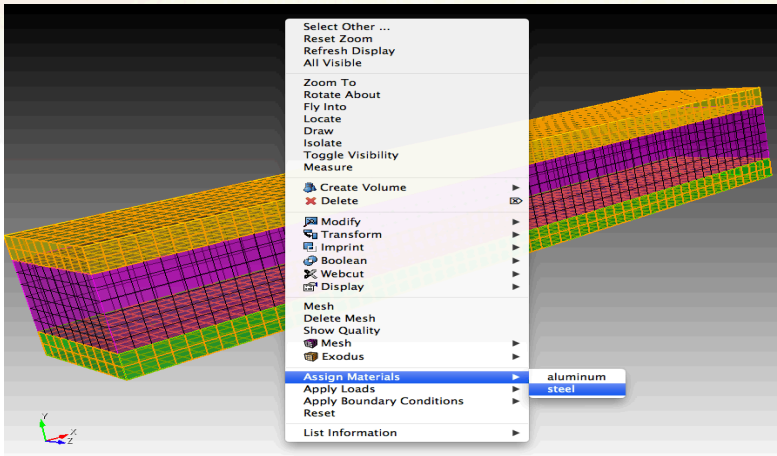
- Project Navigator:** Shows a tree view of the project structure, including folders like "DAKOTA-Milestone", "Ed\_March\_Training", "JoeDemo2", "DS\_Store", "AnalysisFolder", "Deleted Items", "Files", "Materials", "SIMBAWS", and "WorkProductsDocumentFolder".
- Code Editor:** Displays the input file "3\_point\_bend\_test1.i" with the following content:

```
1: Generated by:
2: # SIMBA version 67 Build number 3201(Ad-hoc).
3: # Built on ejfried-dell at 4/23/2009 12:03:47
4: # Exported on Fri Mar 19 10:23:14 MDT 2010
5:
6: begin sierra 3_point_bend_test1
7:
8:   {include("gpm_functions.txt")}
9:   {include("gpm_functions.txt")}
10:
11:   {include("gpm.txt")}
12:
13:
14:
15: begin property specification for material Default
16:   density = 1900.
17:
18: begin parameters for model ELASTIC
19:   poissons ratio = 0.3333
20:   youngs modulus = 3E7
21: end parameters for model ELASTIC
22: end property specification for material Default
23:
24: begin property specification for material mat.1
25:   density = 8000
26:
27: begin parameters for model elastic
28:   poissons ratio = 0.245
29:   youngs modulus = 195.0e+09
30: end parameters for model elastic
31: end property specification for material mat.1
32:
33: # Functions for conditions
```
- Job Status:** A table showing the status of various jobs:

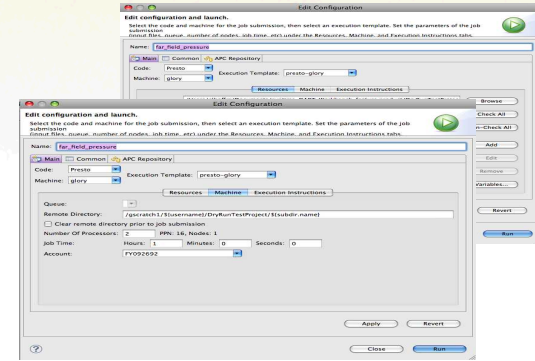
Name	Machine	Stage	Queue Status	Submit Date
Joint_model_SALINAS	shasta	Finished	Completed	Tue Nov 24 15:56:2
Joint_model_SALINAS	shasta	Finished	Completed	Tue Nov 24 16:01:1
Joint_model_SALINAS	shasta	Finished	Completed	Tue Nov 24 16:20:1
dtlb_blivet_060515	thunderbird	Finished	Completed	Tue Nov 24 16:37:1
Joint_model_SALINAS	thunderbird	Finished	Completed	Wed Nov 25 12:09:2
Tail_assy	thunderbird	Finished	Completed	Wed Nov 25 12:25:4
- Model View:** A 3D visualization of a mechanical assembly, showing a cylindrical component with a central hole and a flange, rendered in a wireframe mesh.
- Plot:** A graph showing a linear relationship between two variables, with the x-axis ranging from 0.0 to 1.3 and the y-axis ranging from -0.0007 to 0.0000.
- Team Members:** A table listing team members:

Role	Email	Name
Team Member	mjigbso@sandia.gov	Gibson, Marcus J
Project Manager	elhoffm@sandia.gov	Hoffman, Edward
Team Member	jagreen@sandia.gov	Greenfield, John
- File Explorer:** A file browser showing the contents of the "/gscratch1/elhoffm/JoeDemo2/Files" directory, listing files like "3\_point\_bend\_test1.i", "3\_point\_bend\_test1a.cfg", "3\_point\_bend\_test1a.g", etc.

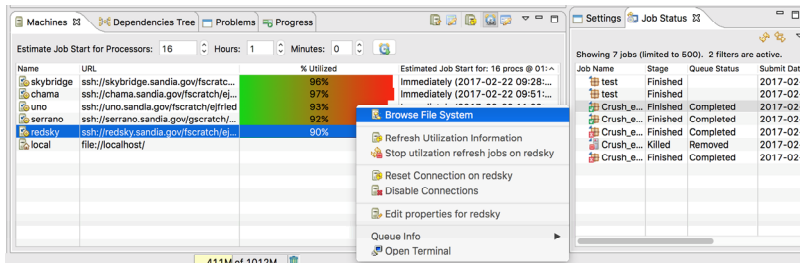
# Historical SAW Capabilities



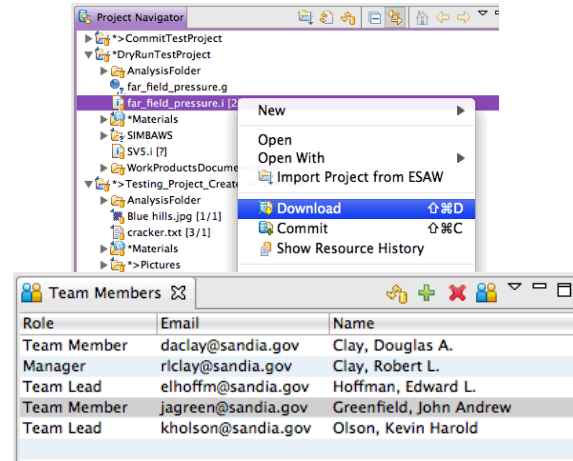
Model Building



Job Submission



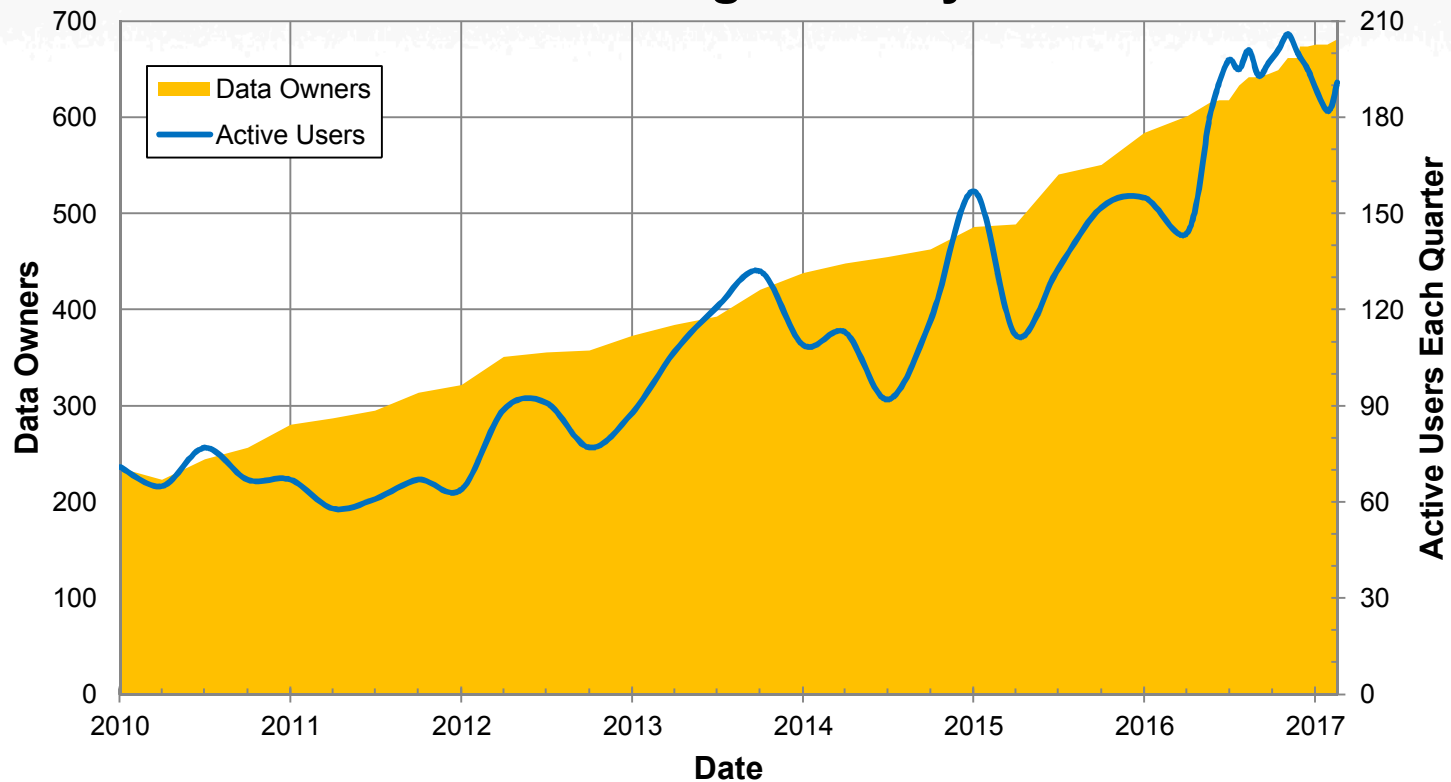
Job Monitoring and Distributed File Management



Simulation Data Management

# Metrics show widespread adoption of SAW

## SAW Usage History



*Data are for internal Sandia users only*

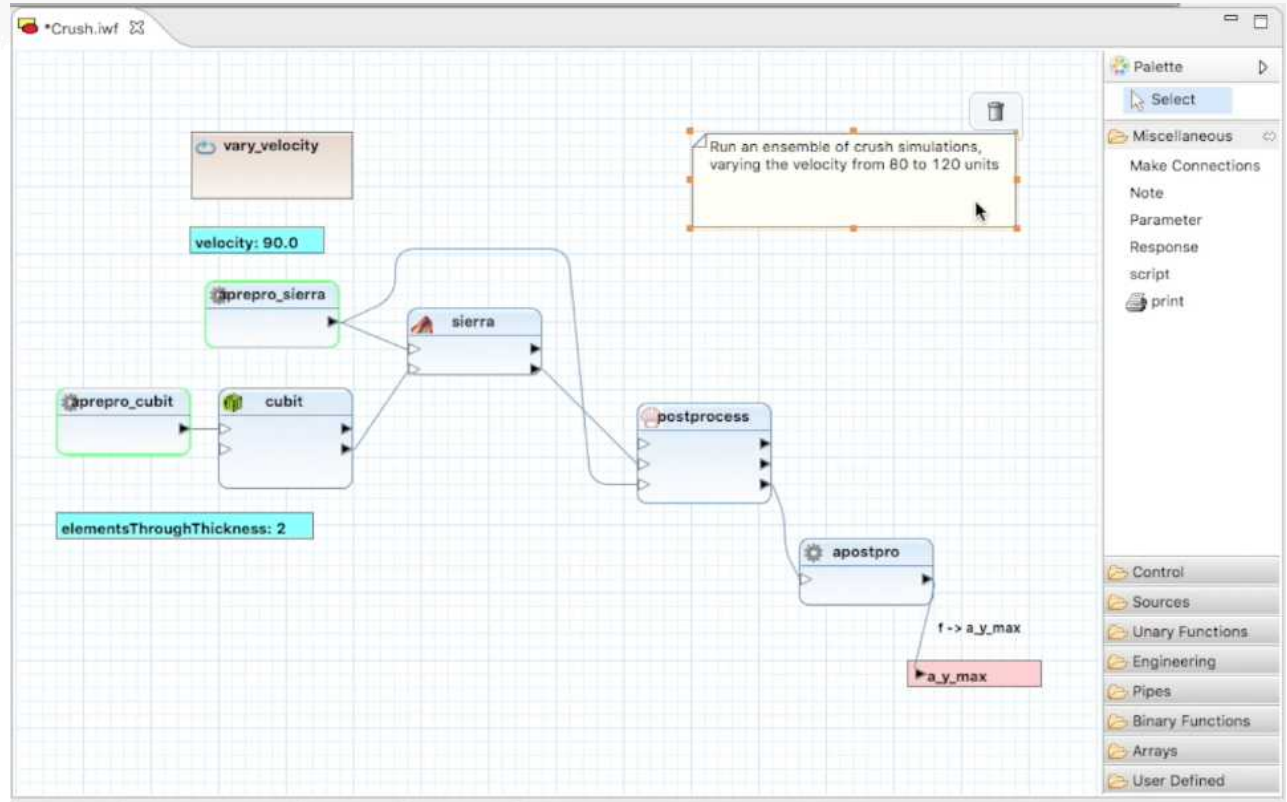
# New Challenges

## *Ensembles*

- ...of multi-step processes
- ...with constant variations
- ...often needing manual intervention
- ...running on unreliable machines
- ...providing high-consequence evidence

# The Solution

**Automated workflow...**  
integrated with everything else in SAW



# Workflow System Requirements

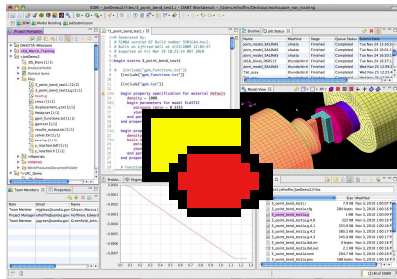
- Workflow engine independence
- Minimal server installation
- Flexible system architecture
  - “Everything executes anywhere”
- Low cognitive load
- Transparent and escapable

# SAW Workflow Architecture

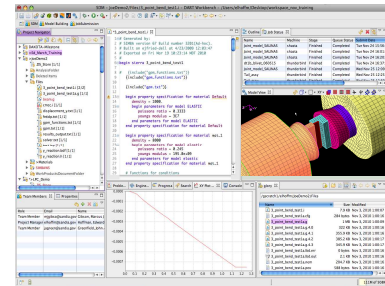
- Graphical Workflow Editor
  - Leverages some tech from Triquetrum/ESWG
  - Data-driven
- Workflow Runtime
  - “Abstract dataflow” (exascale)
  - Integration layer for workflow engine independence
  - Highly portable

# Flexible Architecture

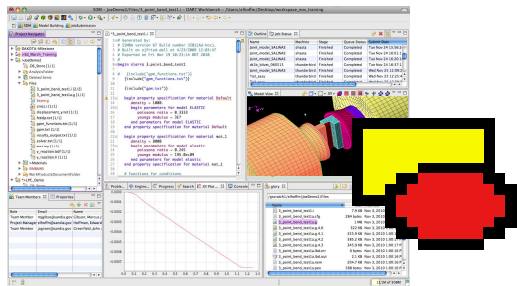
*The same workflow can run in multiple environments. We leverage our existing job submission and monitoring tools to implement this*



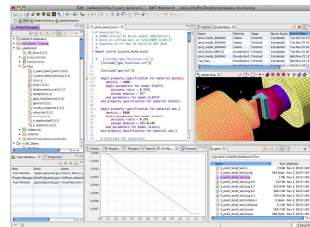
Interactive



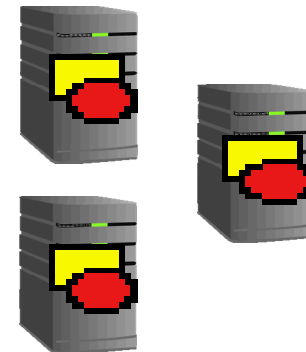
Server



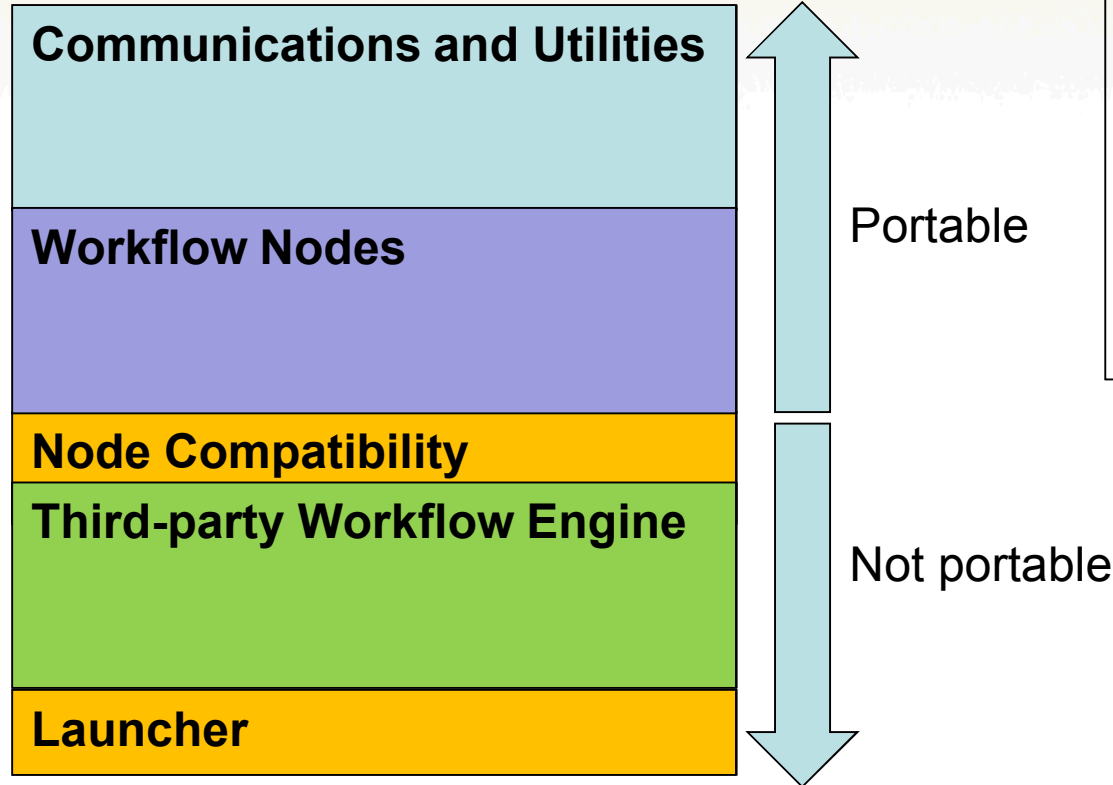
Batch



Distributed

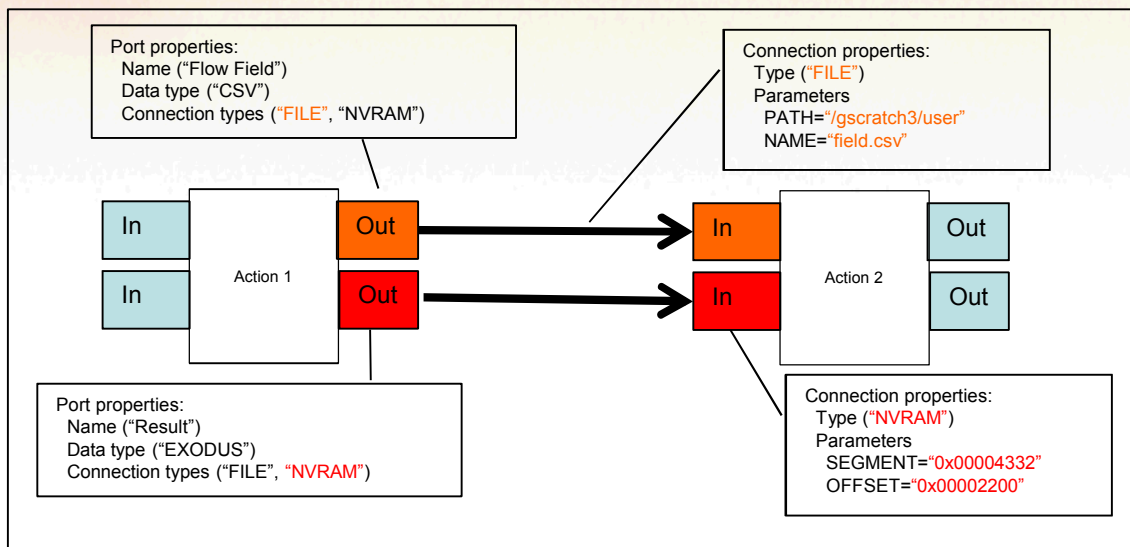


# Workflow Engine Independence



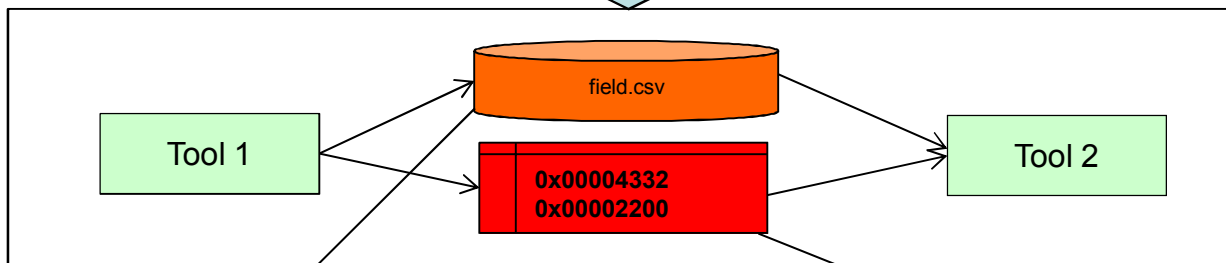
*By using a thin compatibility layer, we can deploy on arbitrary third-party Java workflow engines with little effort*

# Abstract Dataflow



Descriptions of all inputs and outputs of workflow actions include available connection types. At runtime, in-situ connections are used as available. Result is distributed-area workflow containing embedded in-situ workflows.

Execute using COTS Workflow Engine



"Distributed-Area" Link

"In-Situ" Link

# Model Building

Edit model components naturally using familiar tools

The screenshot displays a multi-paneled software interface for model building. The top-left pane shows a workflow diagram with components: 'Vary the velocity from 60 to 120', 'vary\_velocity', 'velocity: 90.0', 'aprepro\_Sierra', 'aprepro\_Cubit', 'cubit', 'sierra', 'bashScript', 'apostpro', and 'a\_y\_max'. The top-right pane shows a code editor with a Fortran-like script for defining functions. The bottom-left pane shows a 'definition for function' dialog for 'pmdi20\_modulus' with a 'piecewise linear' type and a table of data points. The bottom-right pane shows a 'Plot View' of the function, which is a straight line decreasing from approximately 1.25 at time -53.90 to 0.75 at time 81.65.

```
92 ordiante is temperature
93 abscissa is time
94 begin values
95   -500.00 1.0
96     0.00 1.0
97   500.00 1.0
98 end values
99 end definition for function pmdi20_constant
100
101 begin definition for function pmdi20_modulus
102   type is piecewise linear
103   ordiante is temperature
104   abscissa is time
105   begin values
106     -53.90 1.23
107     21.10 1.0
108     73.90 0.80
109     82.20 0.75
110   end values
111 end definition for function pmdi20_modulus
112
113 begin definition for function pmdi20_rate
114   type is piecewise linear
115   ordiante is temperature
116   abscissa is time
```

definition for function

General Function Info

pmdi20\_modulus Plot

Type piecewise linear

Function Definition

Value Properties

Use Data File

x value	y value
-53.90	1.23
21.10	1.0
73.90	0.80
82.20	0.75

Plot View

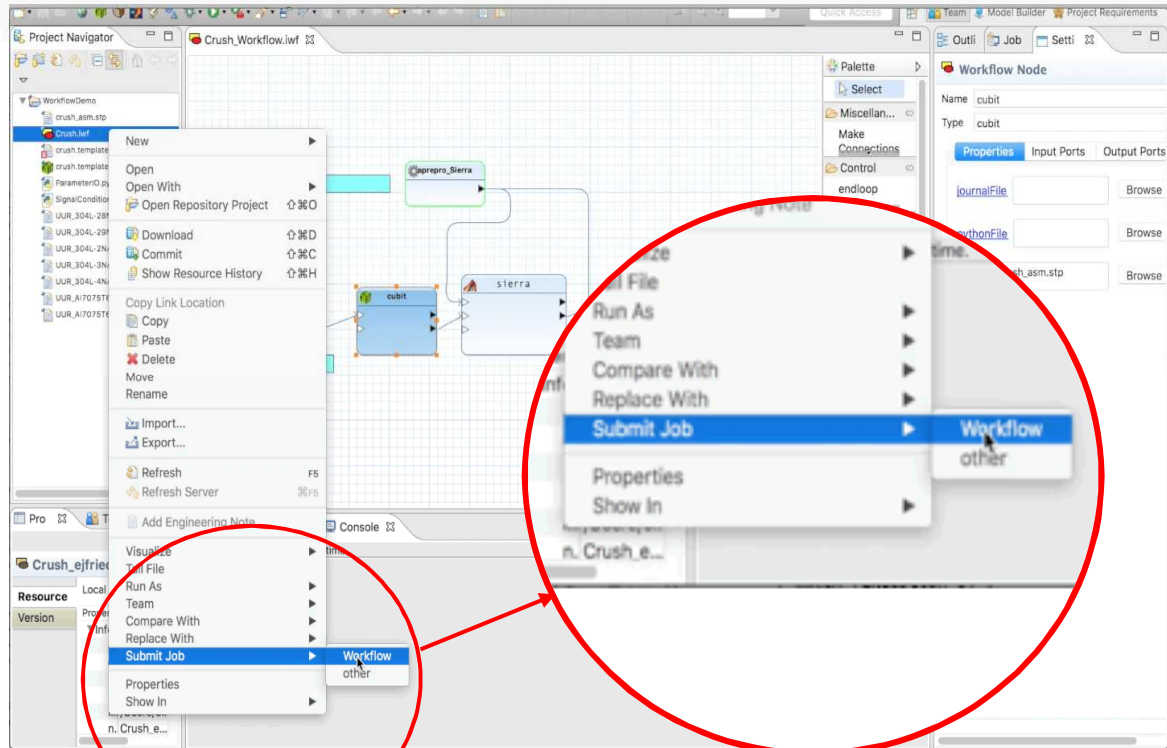
temperature

time

pmdi20\_modulus

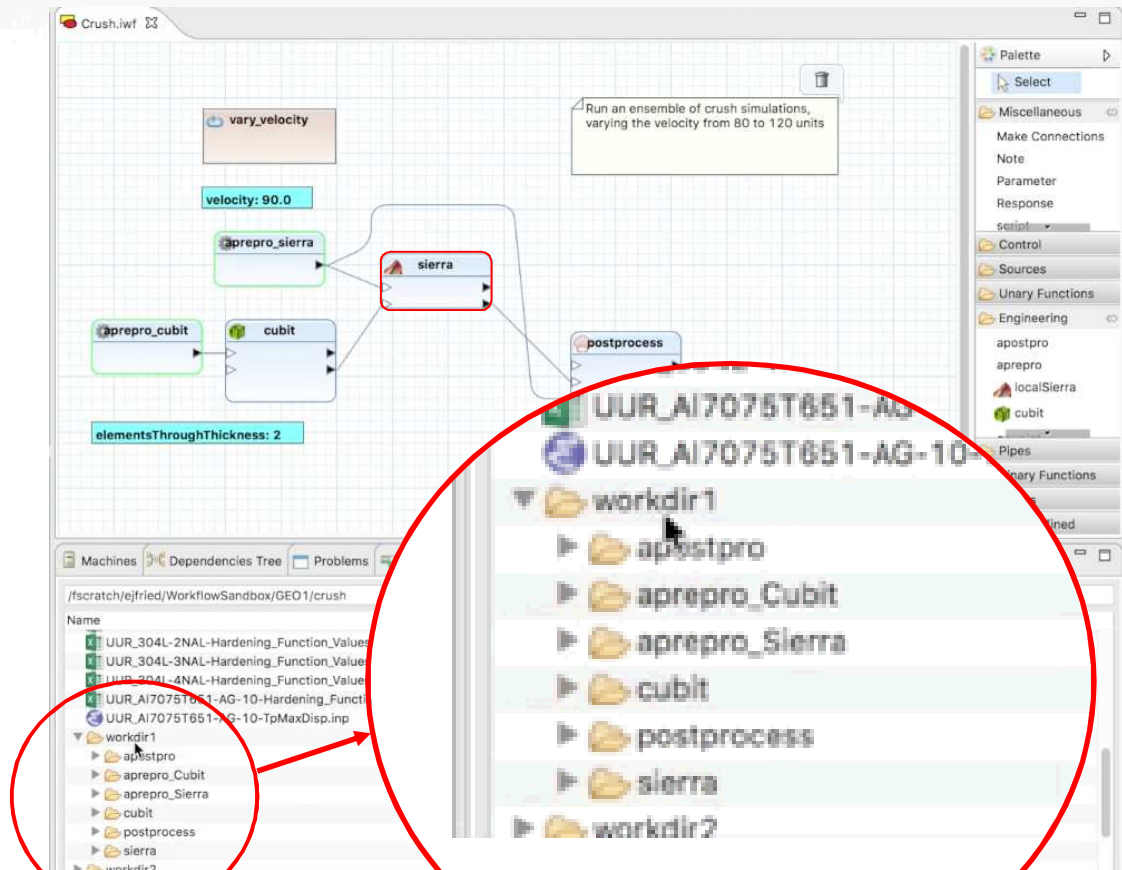
# Job Submission

Workflow  
executed locally  
or remotely  
using SAW job  
submission



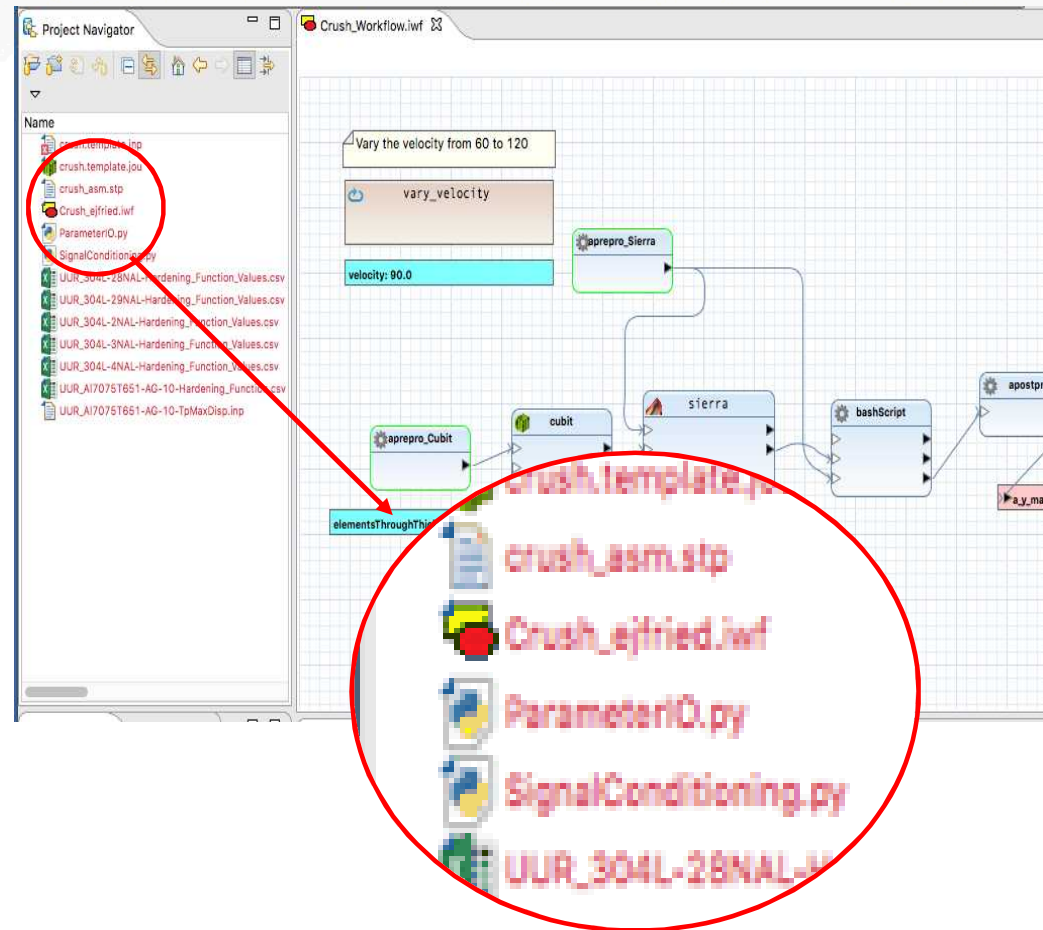
# Job Monitoring and Distributed File Management

Workflow status reported graphically using standard views; clicking a workflow node opens its runtime data directory



# Simulation Data Management

A workflow is just a file, and it can be stored along with all the data it uses



# Benefits

- Reproducibility
  - It's easy to make sure you're performing the same steps
- Reliability
  - Errors handled robustly
- Reusability
  - No reinventing the wheel
- Team communication
  - Intrinsically documented processes

# Acknowledgements

- Marcus Gibson
- Matt Glickman
- Andrew Rothfuss
- Kevin Olson
- George Orient (V&V apps)

*[www.sandia.gov/saw](http://www.sandia.gov/saw)*