High Performance Interactive System Dynamics Visualization

B. Bush, N. Brunhart-Lupo, K. Gruchalla, J. Duckworth

5 February 2016

Goals

Our objective was to develop a system dynamics simulation (SD) framework that supports an end-to-end analysis workflow that is optimized for deployment on ESIF facilities (Peregrine and the Insight Center):

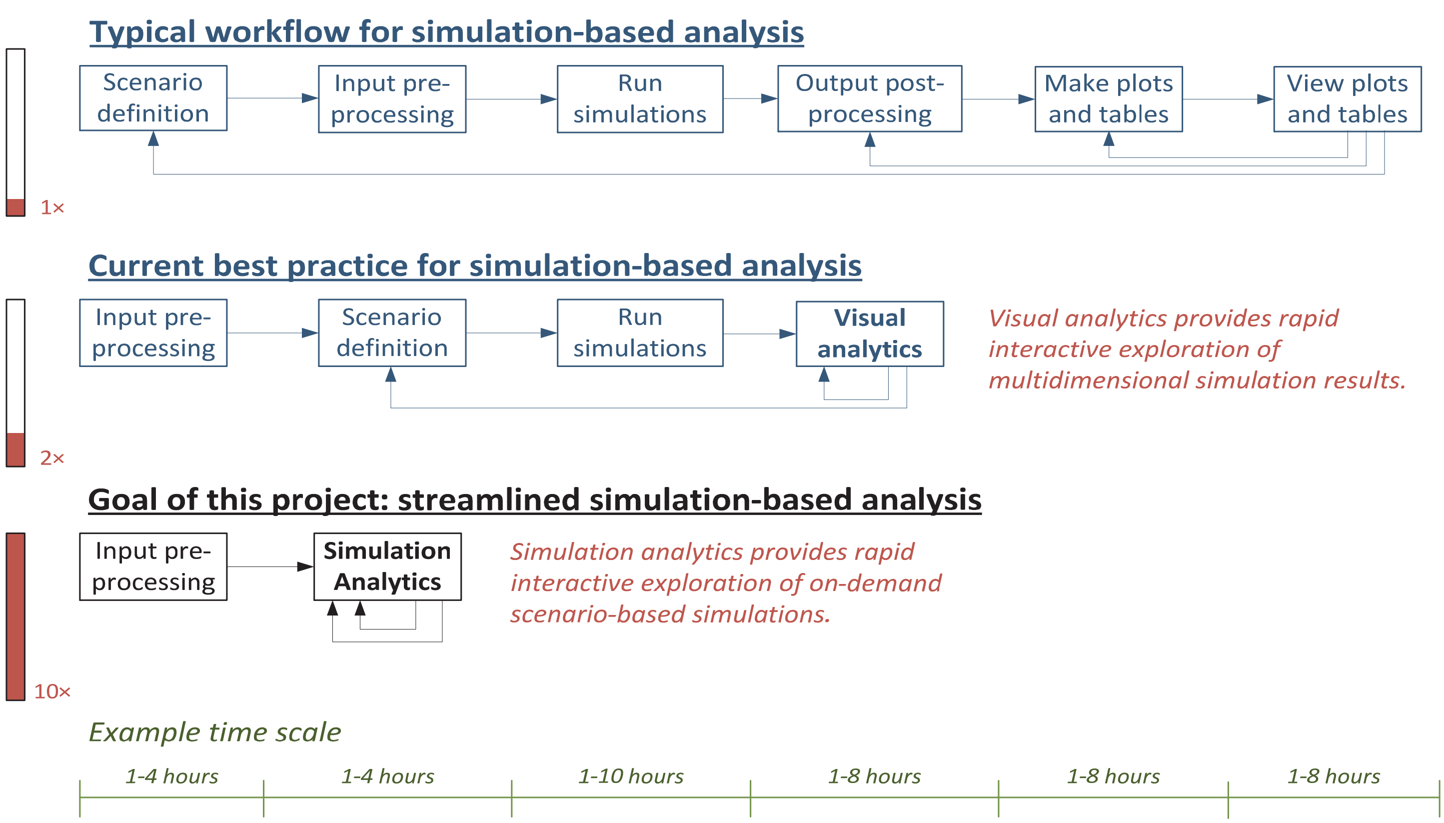
- Parallel and distributed simulation of SD models
- Real-time 3D visualization of running simulations
- Comprehensive database-oriented persistence of simulation metadata, inputs, and outputs

(System dynamics models are just coupled sets of ordinary differential equations with Neumann boundary conditions.)

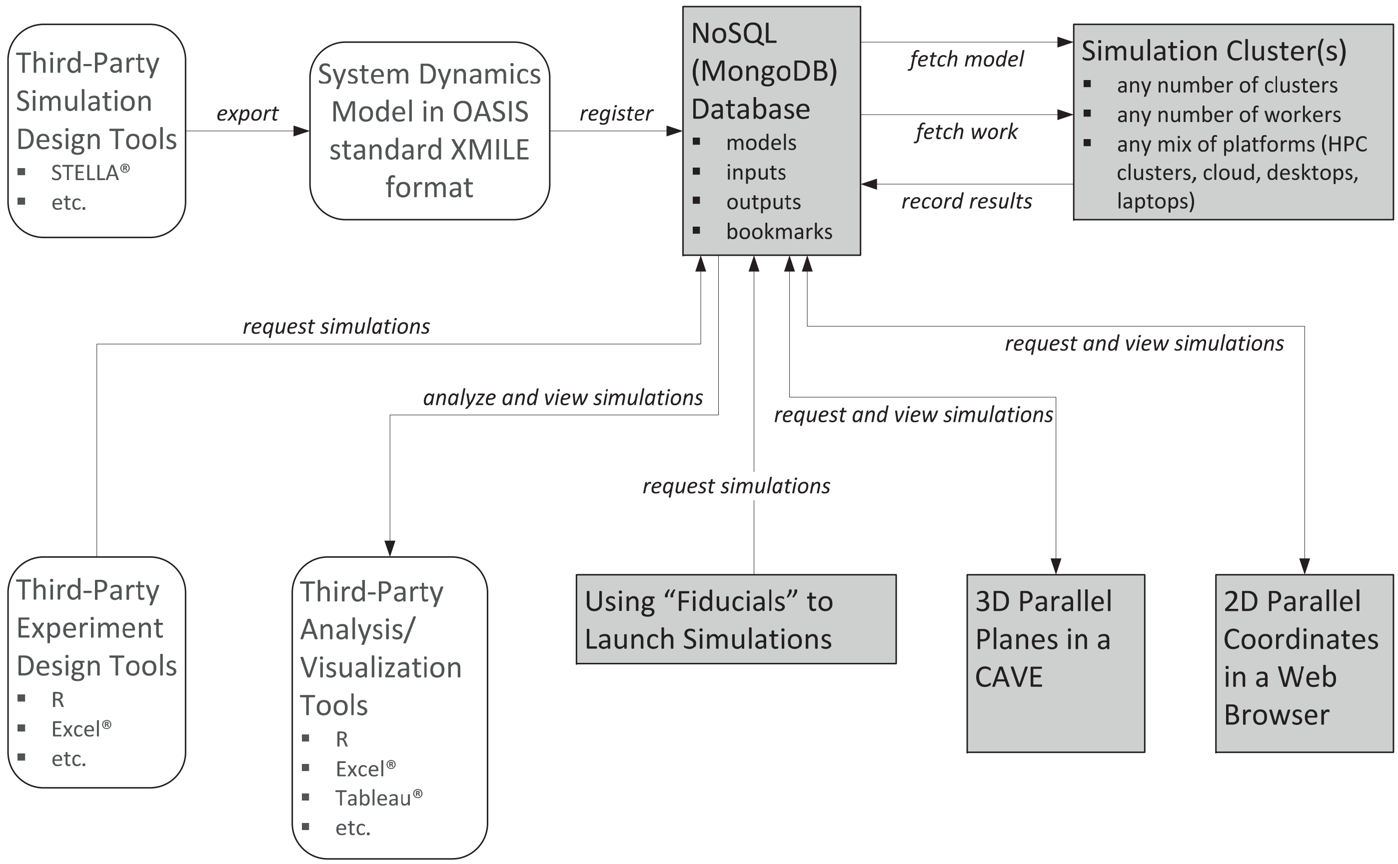
Major Accomplishments

This is the first project to control a live simulation running on Peregrine from an immersive visualization in the Insight CAVE.

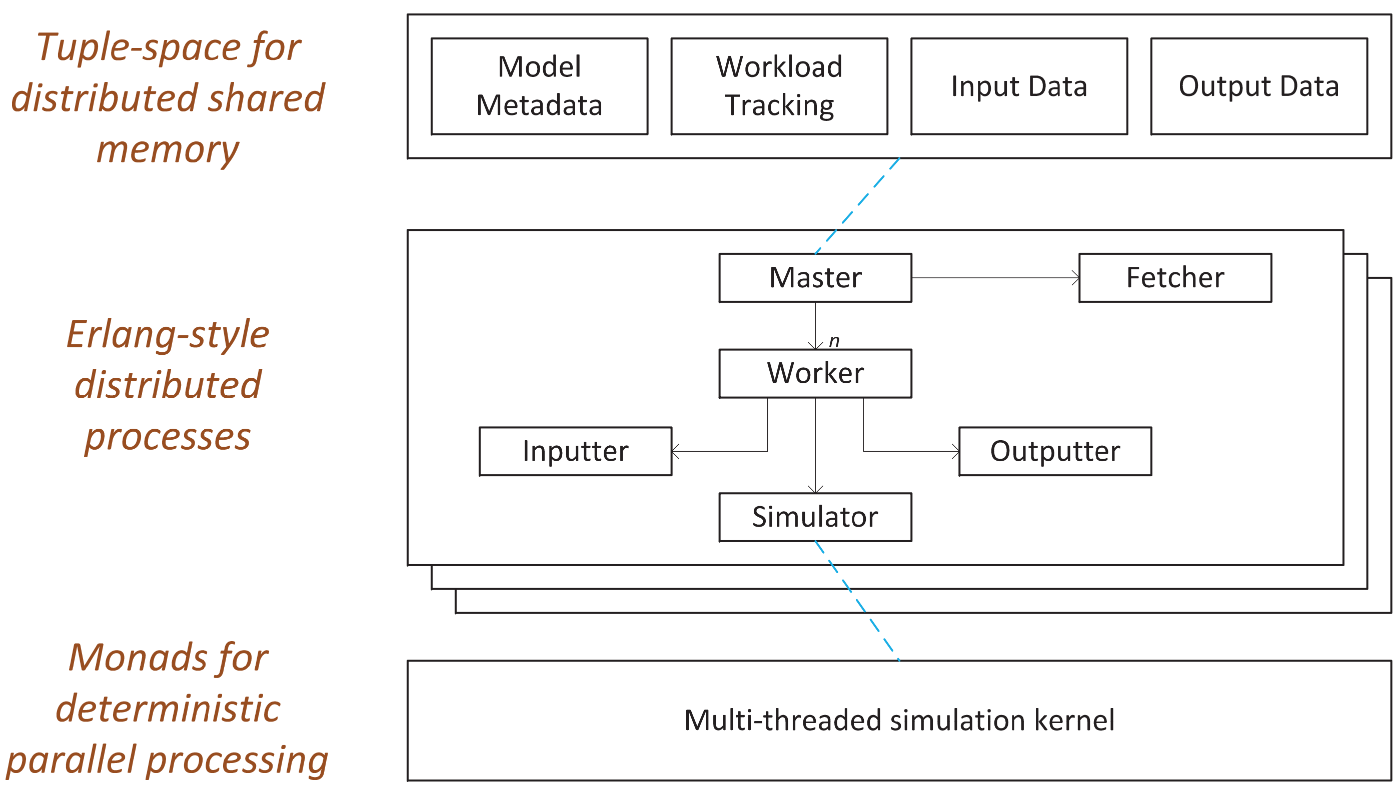
- Distributed computing framework for system dynamics
  - Runs industry-standard XMILE-format system dynamics models
  - Provides a simple interface for specifying input parameters and retrieving output results
  - Executes multiple simulations simultaneously
  - Scales linearly with computing resources
- 3D immersive visualization and control of system dynamics simulations
  - Visualizes any multivariate time series
- 2D web-browser visualization and control of system dynamics simulation
  - Visualizes any multivariate time series



Simulation Workflow

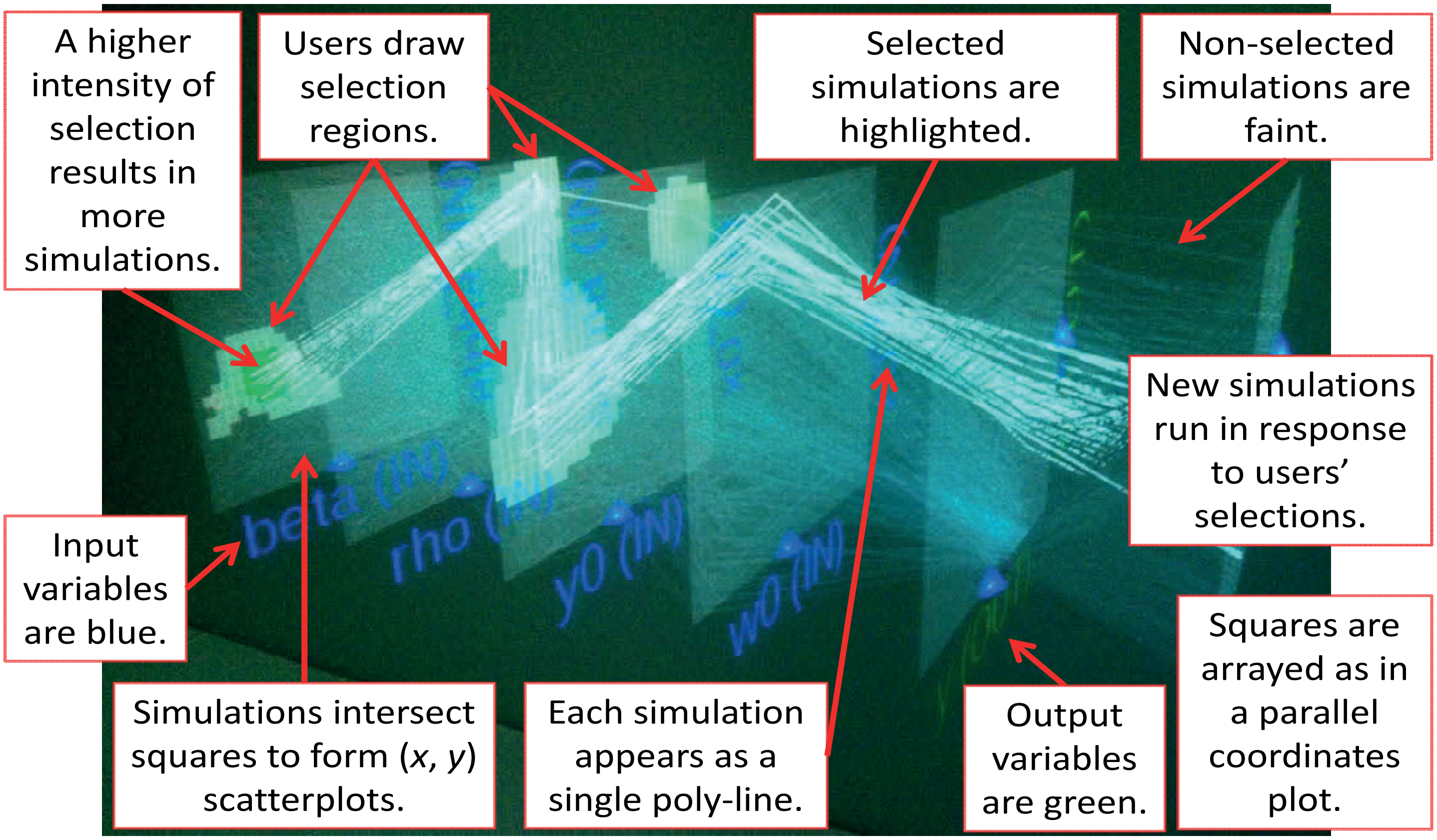


Simulation Architecture



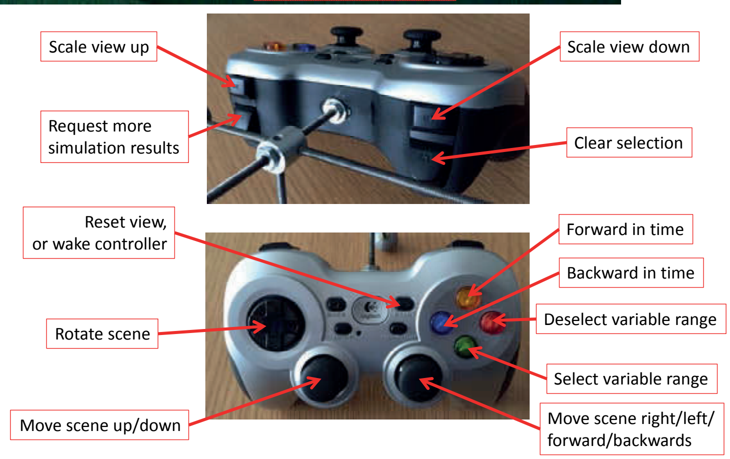
Lazy, pure, statically typed, functional programming for safety and efficiency

3D Parallel Planes in an Immersive Environment (a CAVE)

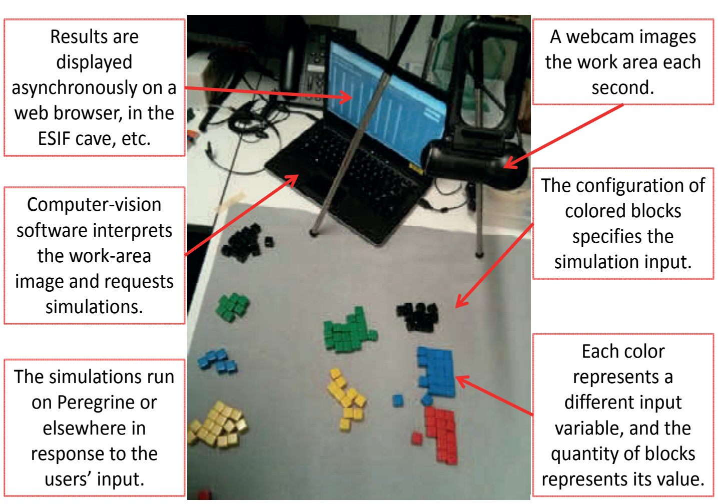


Results of Simulation Development

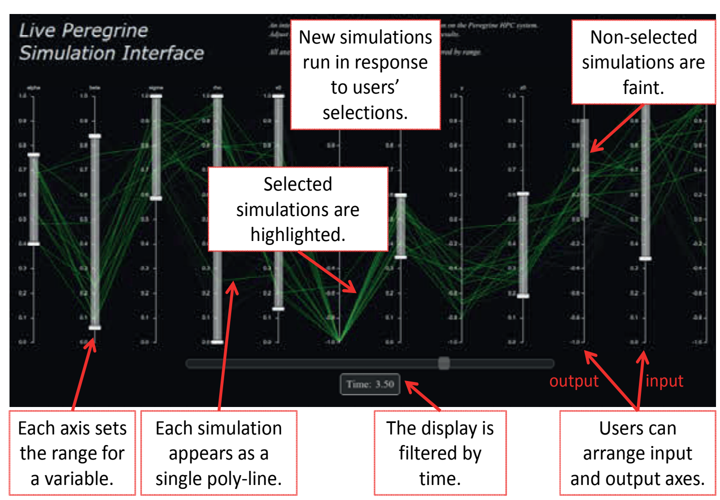
- Implemented eight Haskell packages
  - ~8100 effective lines of code
  - Deployable on Linux, Windows, Mac OS X, etc.
- Used nine real-life STELLA® models of varied complexity (dozens to thousands of equations) for verification and validation
  - Parsing has been verified for all models.
  - Simulation results have so far been completely or partially validated.



Using "Fiducials" to Launch Simulations



2D Parallel Coordinates in a Web Browser

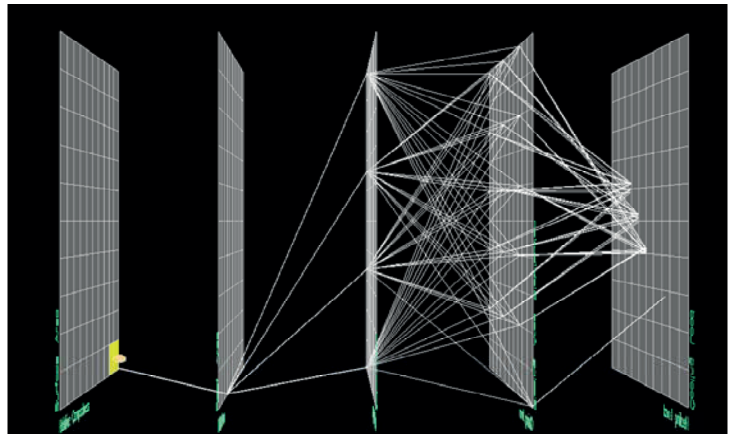


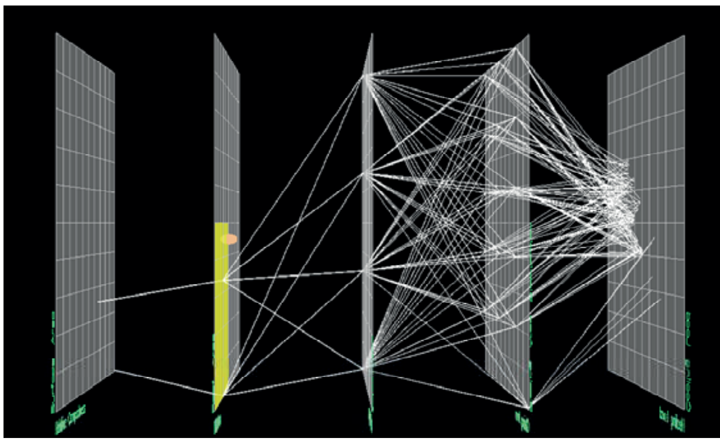
Example Workflow for Exploring Simulated Scenarios with Parallel-Planes Visualization

The diagrams below illustrate an example exploration for a very simple simulation with eight input variables (shown in the four leftmost planes of the diagram) and two output variables (shown in the rightmost plane).

Initially, the users selects a single region of the leftmost plane, which corresponds to a small rectangular range of the first two input variables.

This triggers simulations to be run using that range for the first two input variables and where the other six input variables chosen uniformly at random: the figure at the right shows the visualization once those simulations have completed.

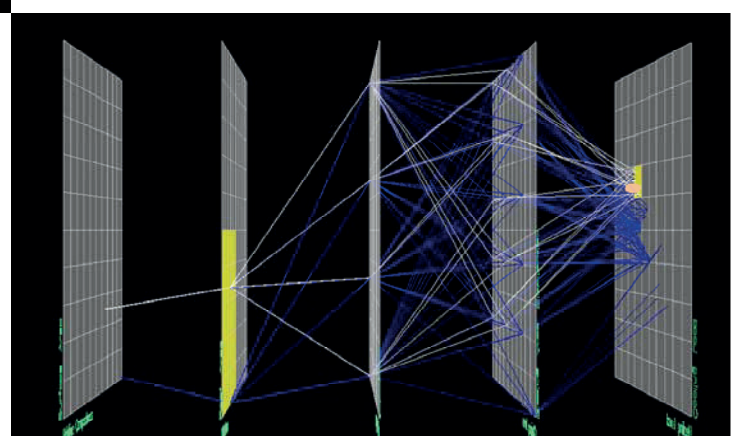


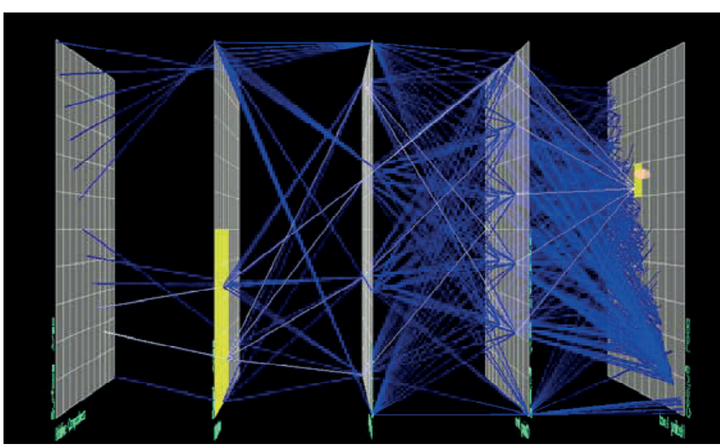


At this point, a user would probably want to explore a broader region of the input space.

They might select, as shown in the figure at the left, a large region of the second plane from the right, triggering simulations for that range of the third and fourth input variables.

In order to explore the rich set of correlations in the simulation output, the user might select a portion of the rightmost output plane, highlighting the observations that lie both in the previously selected inputs and the particular output regions that has been selected: the figure at the right shows those simulations highlighted and the other simulations displayed faintly.





In typical workflows, a user would move between selecting input variables to generate more simulation output and selecting output variables to identify patterns in the data.

This can be thought of as an iterative process of generating and testing hypotheses about the correlation of the outputs among themselves and between the inputs and outputs.