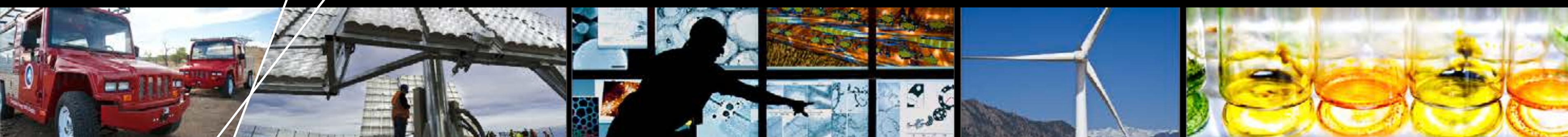
High Performance Interactive System Dynamics Visualization



Laboratory-Directed Research & Development (LDRD) Review

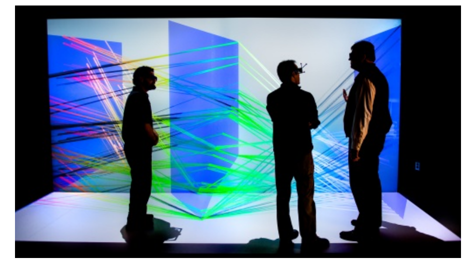
B. Bush, N. Brunhart-Lupo, K. Gruchalla, J. Duckworth

30 September 2015

NREL/PR-6A20-65179

Objectives

- **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):**



Photos: Dennis Schroeder, NREL

  - ✓ 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 coupled sets of ordinary differential equations with Neumann boundary conditions.*

System dynamics models

- **Simple example:**

$$\frac{dx}{dt} = \sigma[(y - x) - w]$$

$$\frac{dy}{dt} = x(\rho - z) - y$$

$$\frac{dz}{dt} = xy - \beta w$$

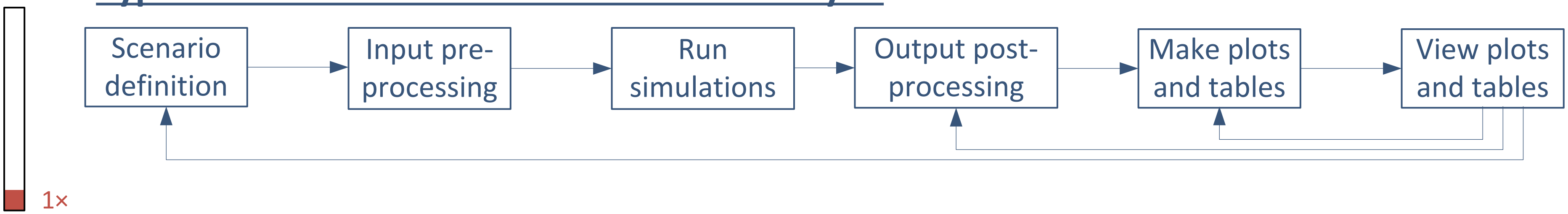
$$\frac{dw}{dt} = z - \alpha wyz$$

- **Real-life models have dozens, hundreds, or thousands of variables.**

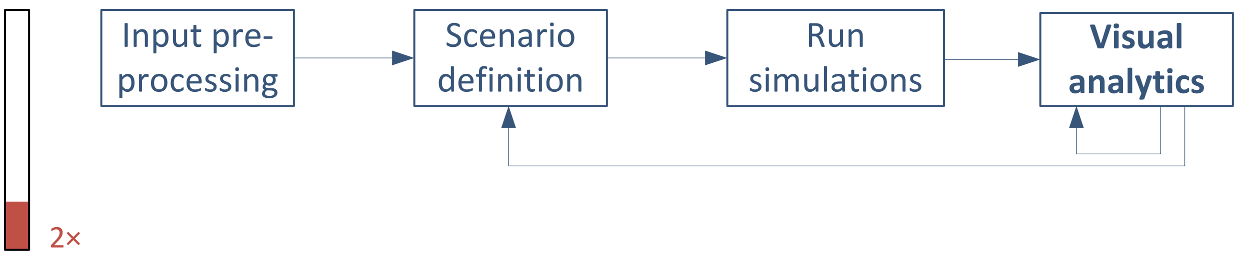
- **NREL models:**
  - Biomass scenario
  - Competition for use of biomass
  - Global land use change
  - Soil carbon
  - Hydrogen dynamic infrastructure and vehicle evolution
  - Industrial learning
  - Wind turbine supply chains
  - Commodity pricing
- **Other well known models:**
  - Meadows *et al.*, "Limits to Growth"
  - Forrester, "World Dynamics"

Motivation for "simulation analytics"

Typical workflow for simulation-based analysis

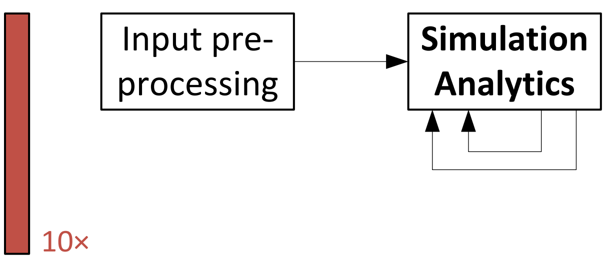


Current best practice for simulation-based analysis



Visual analytics provides rapid interactive exploration of multidimensional simulation results.

Goal of this project: streamlined simulation-based analysis

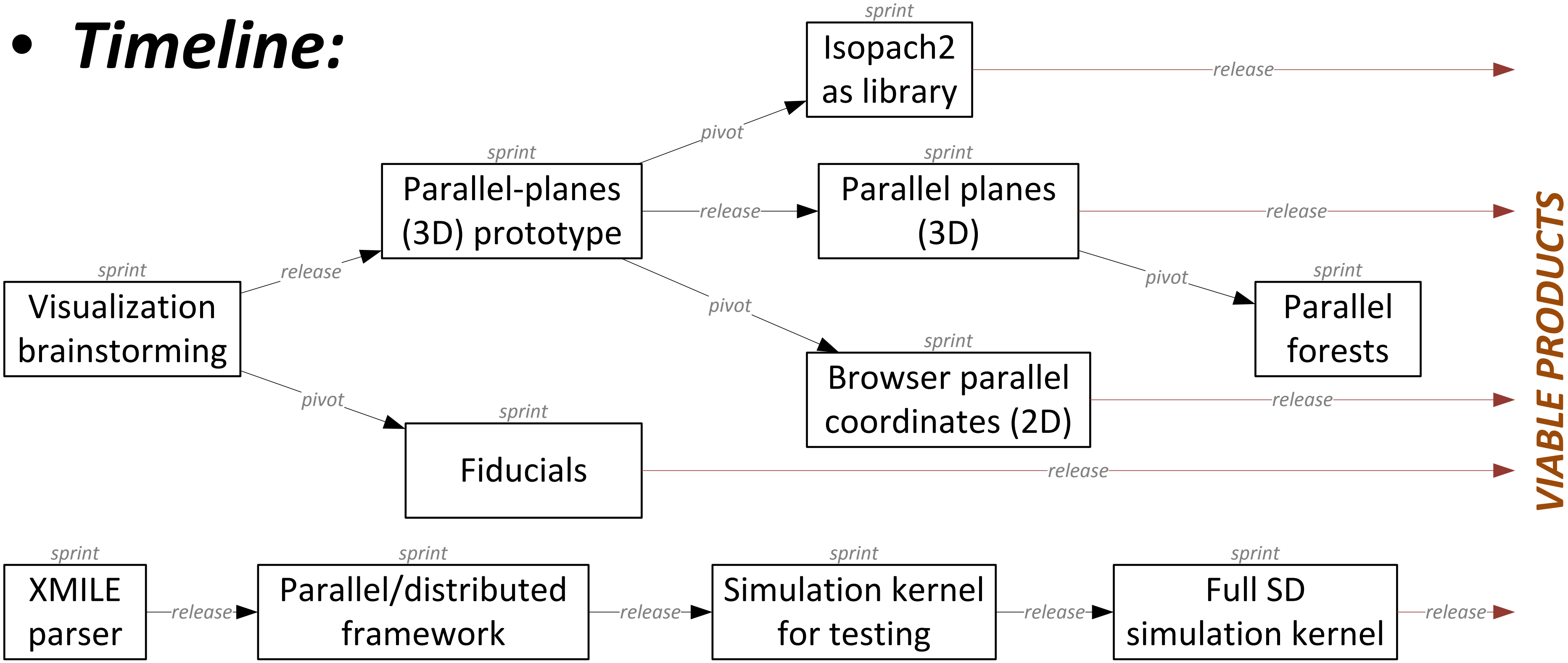


Simulation analytics provides rapid interactive exploration of on-demand scenario-based simulations.

Example time scale

Project management

- ***Budget:*** **$50k**
- ***Period:*** **April through September, 2015**
- ***Methodology:*** **agile / lean**
- ***Timeline:***

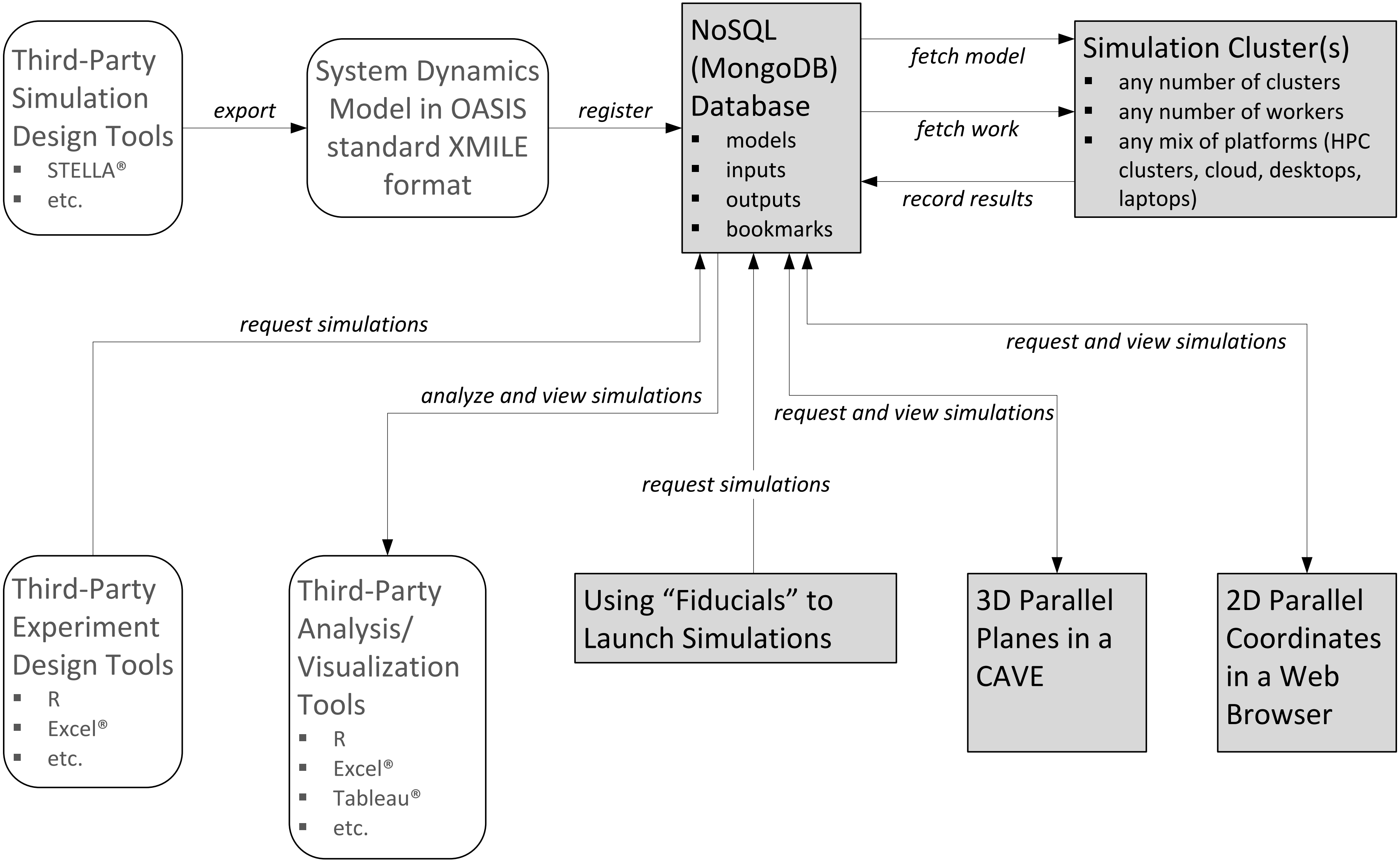


Major accomplishments

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

Architecture and workflow

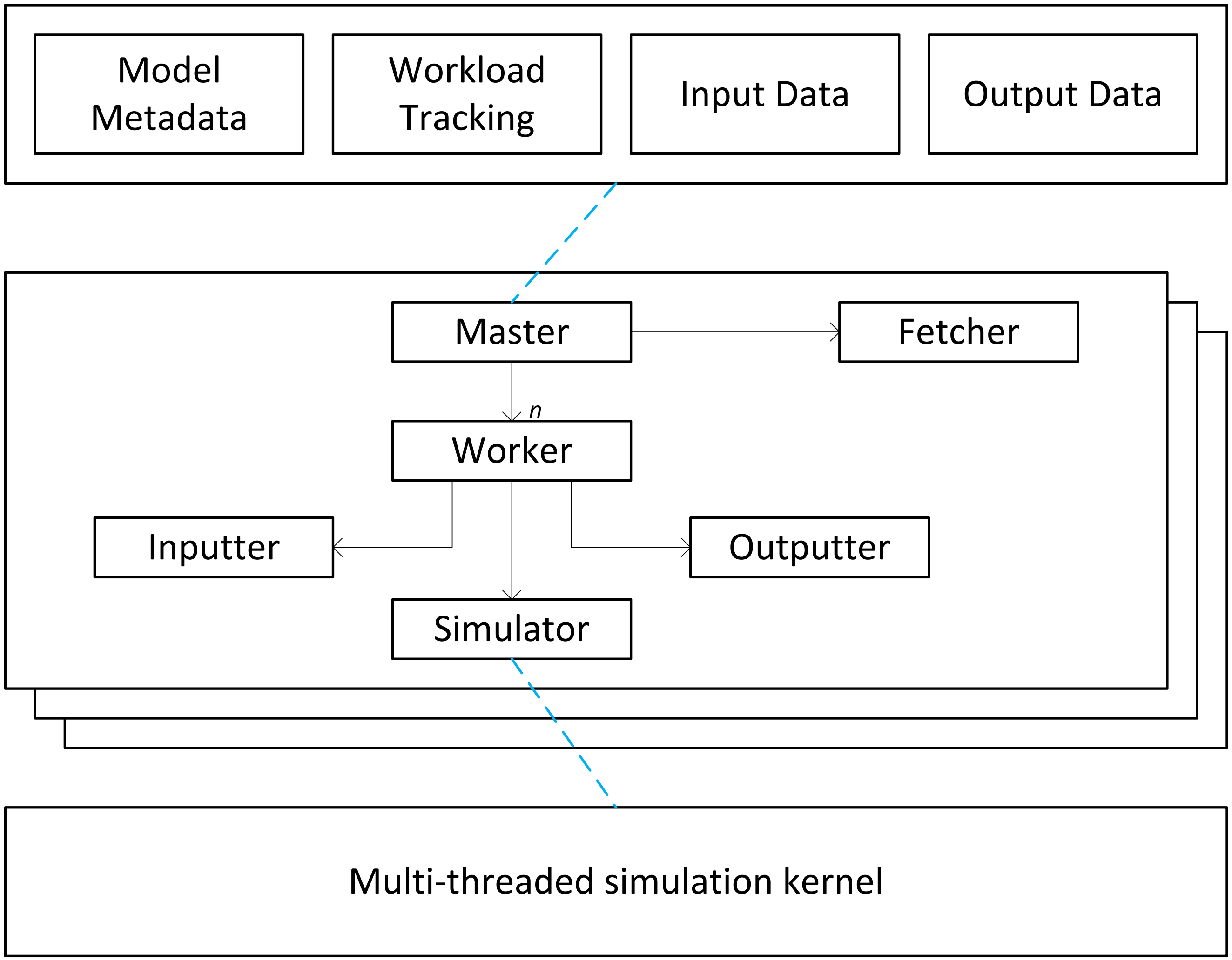


Maximally leveraging modern computing

Tuple-space for distributed shared memory

Erlang-style distributed processes

Monads for deterministic parallel processing

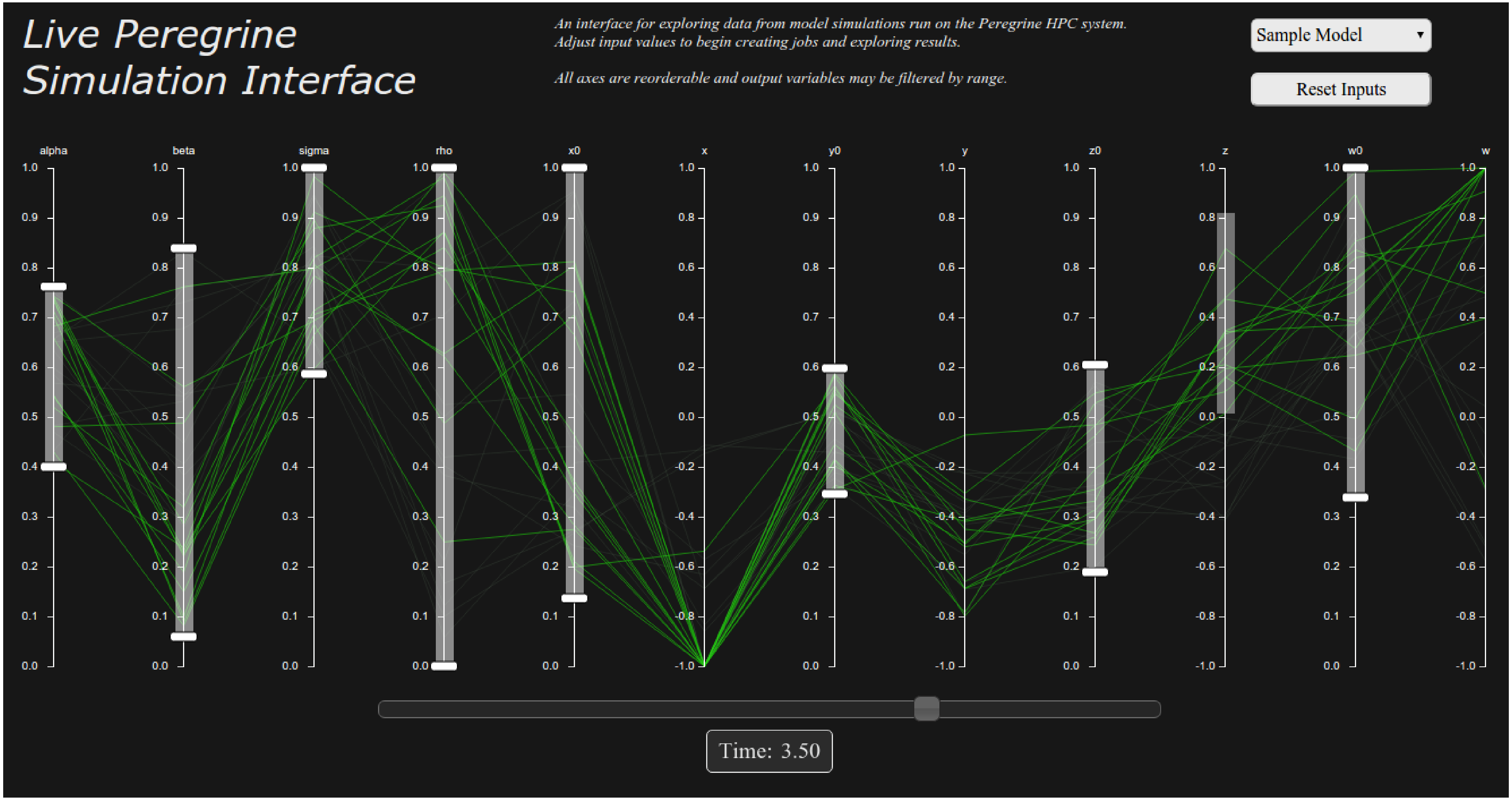


Lazy, pure, statically typed, functional programming for safety and efficiency

Results and metrics of simulator development

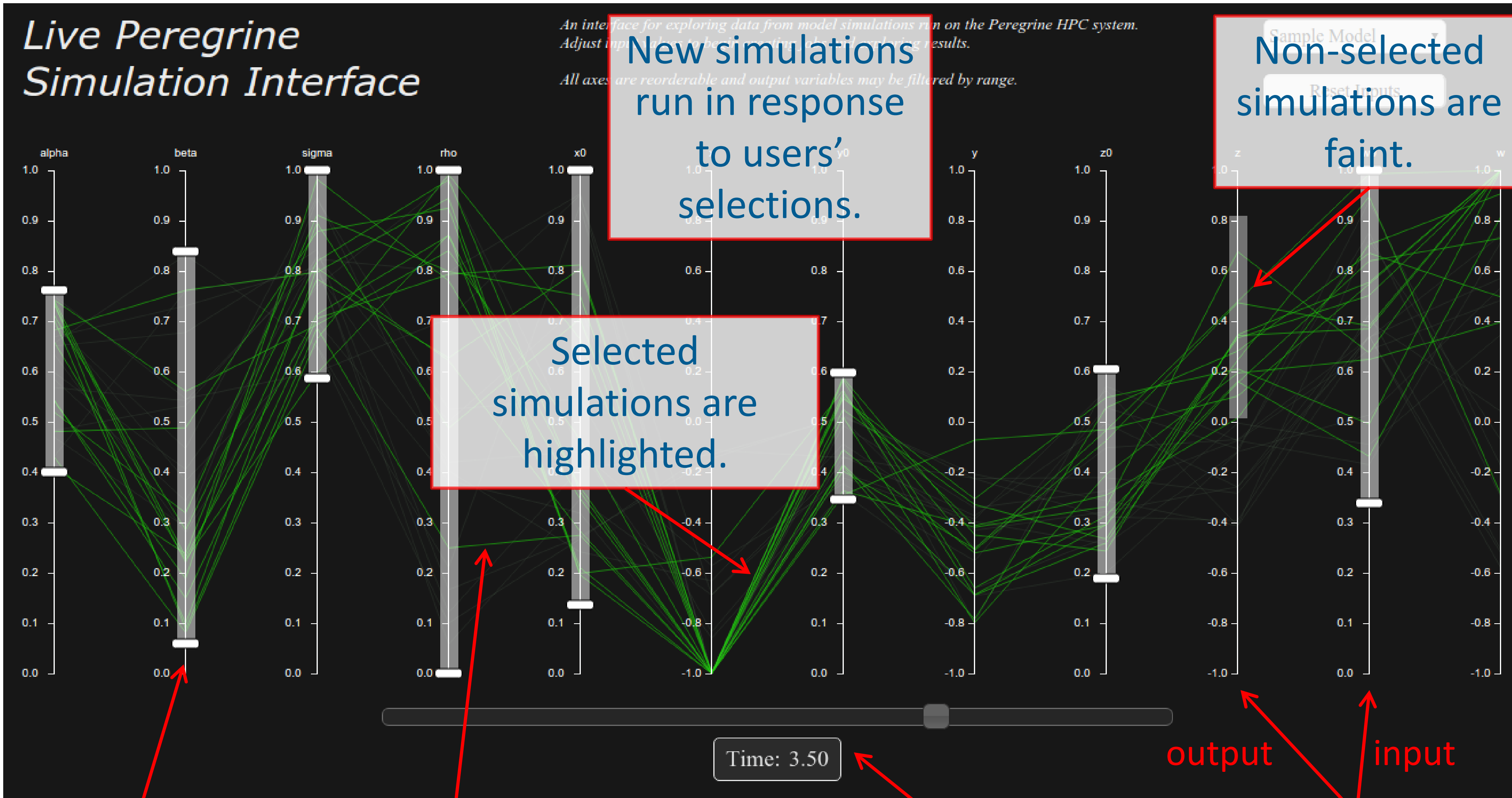
- **Implemented eight Haskell packages**
  - ~8100 effective lines of code
  - Deployable on Linux, Windows, Mac OS X, etc.
- **Used nine real-life STELLA® models of varied complexity (dozens to thousands of equations) for verification and validation**
  - Parsing has been verified for all models.
  - Simulation results have so far been completely validated for six models and partially validated for three more models.
    - The validation of simulation results for large models can be labor intensive and requires new automation tools.
  - Simulations load as fast as in STELLA® and run somewhat slower, but the software has not yet been optimized for fast simulation, so this is not a significant concern.

2D parallel coordinates on a web browser



Demonstration available at <http://cn05.bigde.nrel.gov:8080/index.html>.

2D parallel coordinates on a web browser



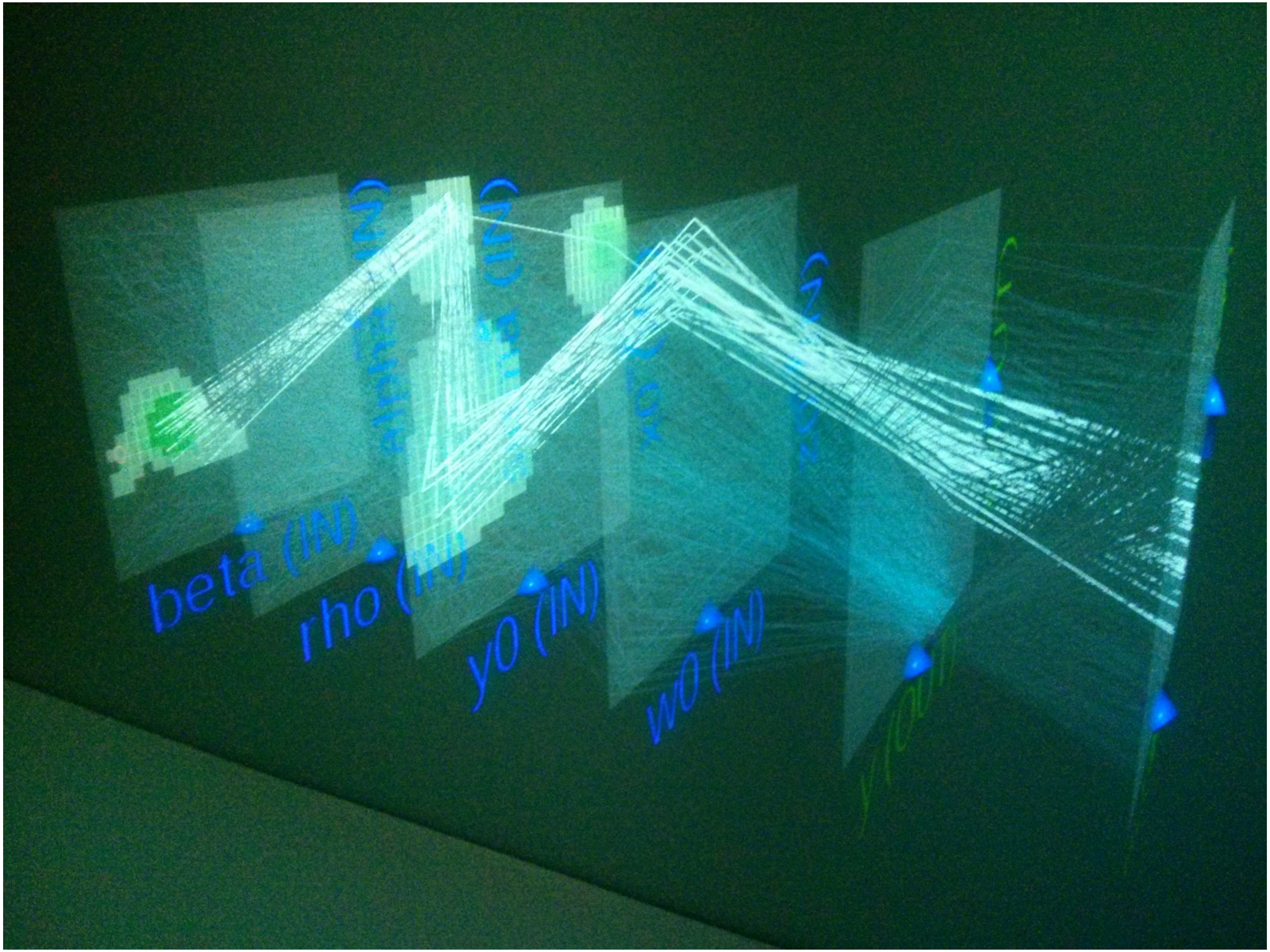
Each axis sets the range for a variable.

Each simulation appears as a single poly-line.

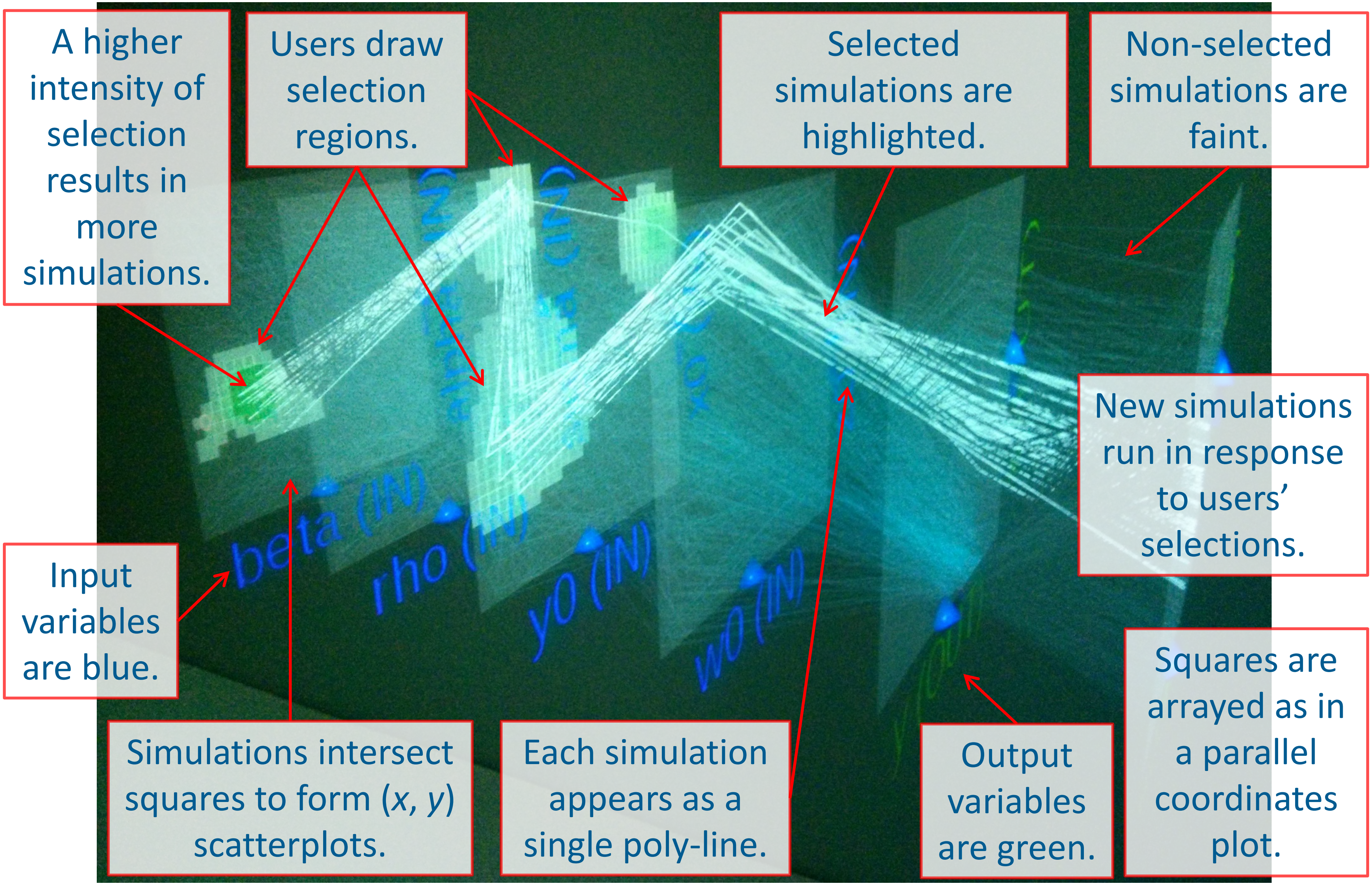
The display is filtered by time.

Users can arrange input and output axes.

3D parallel planes in a CAVE



3D parallel planes in a CAVE



Control of 3D parallel planes

Scale view up

Request more simulation results

Scale view down

Clear selection

Reset view, or wake controller

Rotate scene

Move scene up/down

Forward in time

Backward in time

Deselect variable range

Select variable range

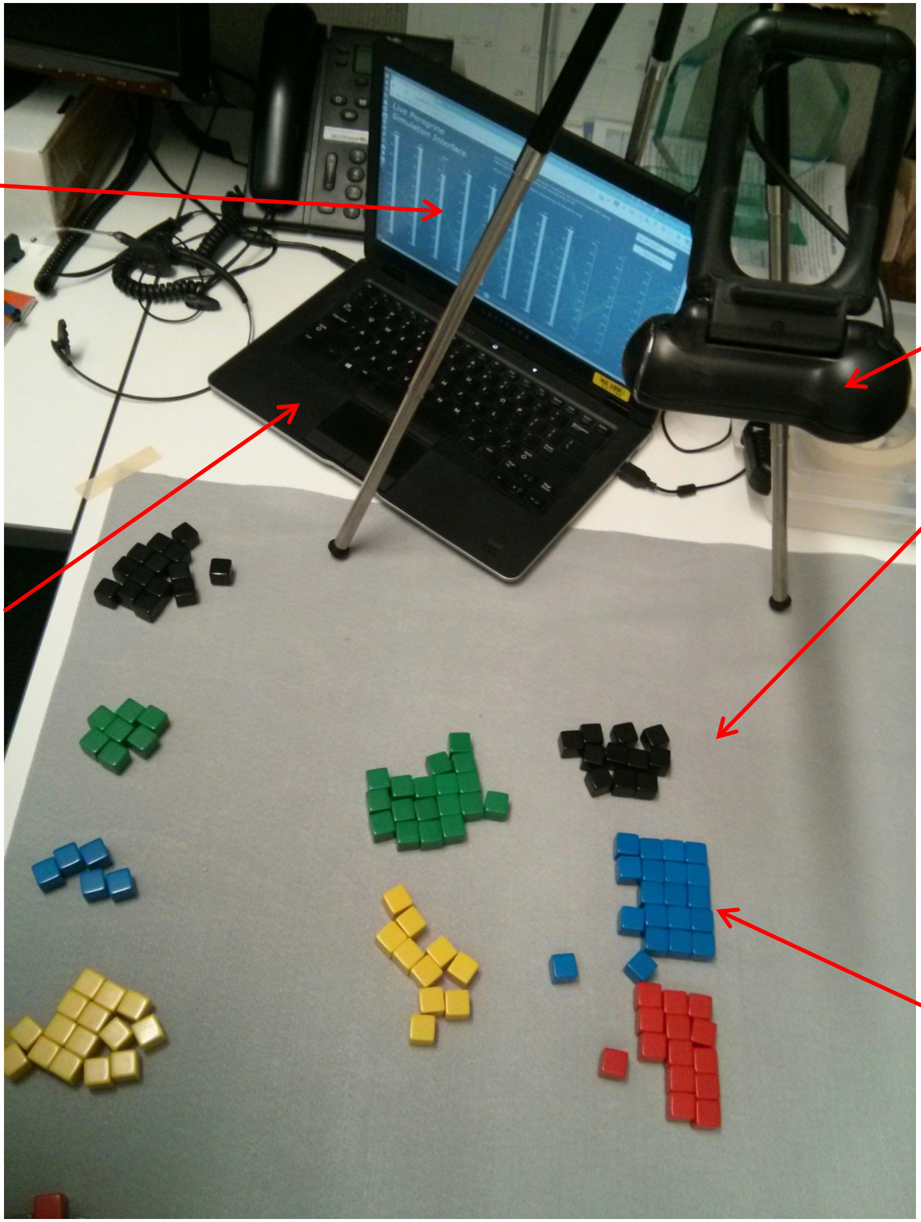
Move scene right/left/ forward/backwards

Using "fiducials" to launch simulations

Results are displayed asynchronously on a web browser, in the ESIF cave, etc.

Computer-vision software interprets the work-area image and requests simulations.

The simulations run on Peregrine or elsewhere in response to the users' input.



A webcam images the work area each second.

The configuration of colored blocks specifies the simulation input.

Each color represents a different input variable, and the quantity of blocks represents its value.

Collaboration

- **System-dynamics simulator**
  - Brian Bush (SEAC)
- **CAVE (3D) visualization**
  - Nicholas Brunhart-Lupo (CSC)
  - Brian Bush (SEAC)
  - Kenny Gruchalla (CSC)
- **Web (2D) visualization**
  - Jonathan Duckworth (SEAC)
- **Visualization consulting**
  - Steve Smith (Los Alamos Visual Analytics)
- **Informal discussions**
  - Steve Peterson (Lexidyne LLC)

Artifacts

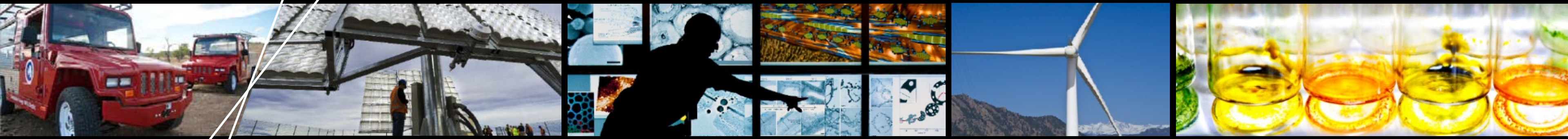
- **Distributed system dynamics software**
  <https://github.nrel.gov/stats/sdsv>
- **3D parallel planes software**
  <https://github.nrel.gov/nbrunhar/SDSVVis>
- **2D parallel coordinates software**
  <https://github.nrel.gov/dav-gis/simulation-viz>
- **Poster, brochure, and slide deck**
- **Video demonstration**
- **Draft white paper**
- **Software Record for "on-demand, distributed, system-dynamics simulation"**
- **Record of Invention for "simulation fiducials"**

Source: NREL / Photo: Dennis Schroeder

New frontiers

- **Collaborative exploration of simulations**
  - User experience testing and refinement
    - Working with a variety of user types and group configurations
  - Mixed modalities
    - CAVE, browsers, R, and/or fiducials
    - Local and remote users interacting with the same simulations
  - “Bookmarking”, “snapshotting”, and sharing of simulation visualizations
  - “Pruning” of simulation spaces
- **Scaling up the number of simulations displayed**
  - “Bundling” of similar simulations in displays
- **Increasing simulation throughput and response time**
  - Use of graphics processing units (GPUs) for simulation
  - More aggressive parallelization
  - Predictive precomputation of simulations
  - Optimization (compiling, rewriting) of simulation models via a domain-specific language (DSL), perhaps with a full fledged compiler

Thank you!



- Questions
- Observations
- Recommendations
- Discussion of next steps