SANS 2D modeling prototype

What are we talking about?:

- Share our recent experience with GUI building.
- Hopefully contribute to “requirements” for Central Services GUI framework by actually showing what we want.

What we talked about recently:

**July**
Mat: non-GUI pyre app

**July**
Paul K.: Plottables

**Future?**
pyre GUI app

**Today**
non-pyre GUI app
What we did:

- This is a requirements task for one of our flagship applications: we give something to the users to get comments.

- Step 1: we wrote a proof-of-principle, risk-addressing, prototype. Not a complete app, but an app that contains most technical difficulties.

  svn://svn@danse.us/sans/sansgui

- Step 2: once we were satisfied that all the risks were addressed, we threw it away and we will slowly rebuild it from scratch with proper design.
Step 2: Goal of the re-write exercise:

- Write a prototype with a design that will not cause us shame.
- Concentrate on GUI components, less on the calculations.
- Come up with a design with reusability in mind (obvious).
- Come up with a design with automated testing in mind.

svn://svn@danse.us/sans/other/prototype_2D-0.0.2

Solution for this prototype:

- Write independent wx panels that only know about the events they send and receive.
- Even the main app doesn’t know how the panels relate to each other, it only places them on the Frame.
- A StateManager is used to manage state changes.
- Use the wx event loop to avoid race conditions between panel actions and getting caught in unknown states.
- Use the wx event loop to inject stimuli during automated testing.
Close correspondence between events and “stimuli”
Common DANSE interface framework?

- The AUI approach is popular among subgroups (3). Maybe we should put some effort is developing a common DANSE AUI framework.

- The developers and expert-users will only need to know about the event loop and the layout.

- CS could provide common panels (for examples, parameter editor for pyre component).

- A sizable amount of GUI testing could be built into the framework.