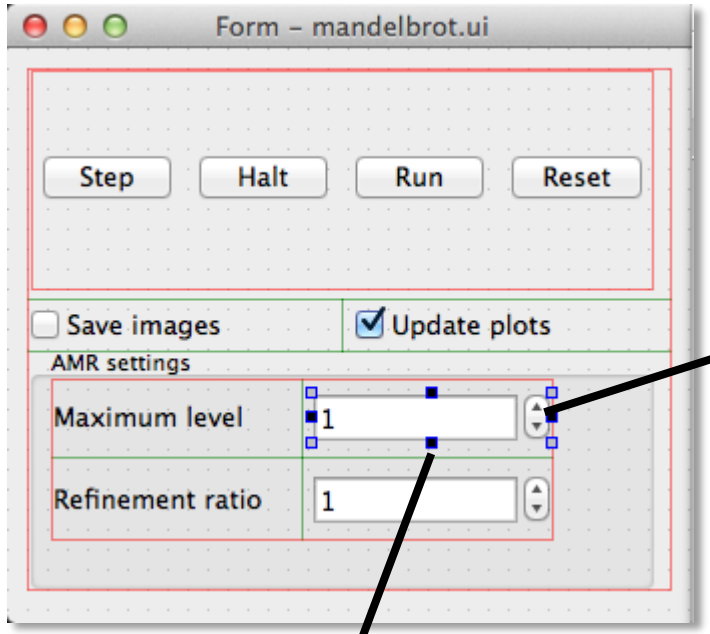
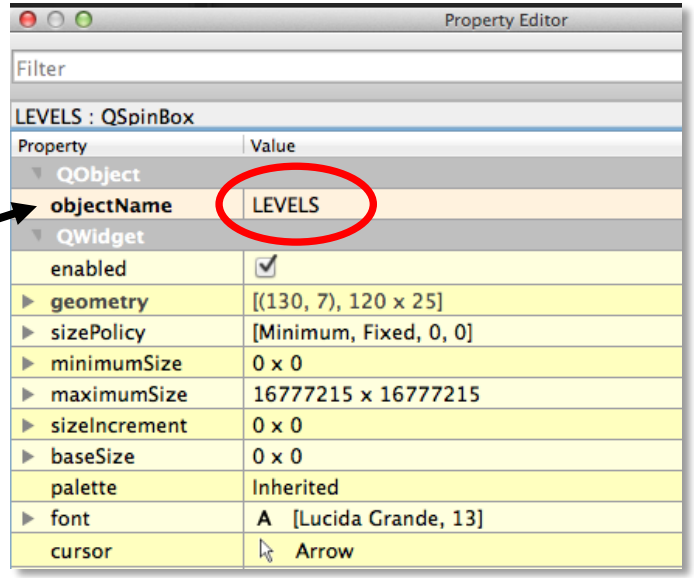
# Setting up simulation UI

1. Make a UI in Qt Designer. Give the widgets expressive names since you'll use the names to get/set properties using the widgets.
2. Tell the VisItInitializeAndDumpSimFile() function about the UI file
3. Copy the UI file into your *~/.visit/ui* directory
4. Register widget callbacks in your simulation.
5. Add function calls to send new values to the UI widgets.




Properties in Qt Designer




Register callbacks

```
/* Register some ui actions */
VisItUI_clicked("STEP", ui_step_clicked, &sim);
VisItUI_clicked("HALT", ui_halt_clicked, &sim);
VisItUI_clicked("RUN", ui_run_clicked, &sim);
VisItUI_clicked("RESET", ui_reset_clicked, &sim);
VisItUI_valueChanged("LEVELS", ui_levels_changed, &sim);
VisItUI_valueChanged("REFINEMENTRATIO", ui_ratio_changed, &sim);
VisItUI_stateChanged("SAVEIMAGES", ui_saveimages_changed, &sim);
VisItUI_stateChanged("UPDATEPLOTS", ui_updateplots_changed, &sim);
```

```
void
ui_levels_changed(int value, void *cbdata)
{
    simulation_data *sim = (simulation_data *)cbdata;
    printf("ui_levels_changed: %d\n", value);
    sim->max_levels = value;
}
```

**Alter sim state in UI callback function**

Send new data to UI

```
/* Update some UI elements */
VisItUI_setValueI("LEVELS", sim->max_levels, 1);
VisItUI_setValueI("REFINEMENTRATIO", sim->refinement_ratio, 1);
VisItUI_setValueI("SAVEIMAGES", sim->savingFiles, 1);
VisItUI_setValueI("UPDATEPLOTS", sim->autoupdate?1:0, 1);
```

*Note: These "VisItUI" functions are probably not exposed from Python right now*