

Creating Quality UIs with LabVIEW

Grant Heimbach
LabVIEW Product Manager

Agenda

1. Definitions, rules and advice

(not LabVIEW specific, but important)

2. Some cool UI techniques for LabVIEW

(and why you would consider using them in your application)

3. Where to go to download some reusable components

(because everybody loves free stuff)

What is a UI?

- **Literally:** User Interface
- How user interacts with program
- First thing user notices
- Make the user's job easier
- Don't be frustrating

UI and Usability

UI

Superfluous
eye candy

Visual elements that
help the user perform
a task in an efficient
manner

Background
Task or Process

Usability

Some General Rules

1. Don't be innovative
2. Less is more
3. Think About Your User

1. Don't Be Innovative

Use familiar elements

- Buttons
- Icons
- Terminology
- Dialogs
- Menus



1. Don't Be Innovative

The background image shows the interior of a BMW car, viewed from the driver's perspective. The steering wheel is prominent in the foreground, featuring the BMW logo. Behind it, the instrument cluster with three gauges is visible. The center console and gear shift are also in view. The car is parked outdoors, with trees and a road visible through the windshield.

Still some license for creativity

- Don't change the way similar looking things behave
- Polish, don't reinvent

2. Less is More

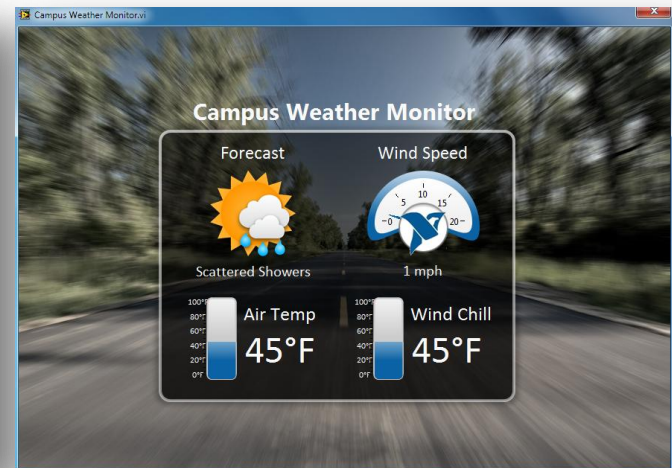
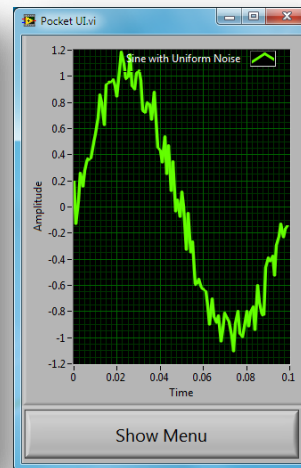
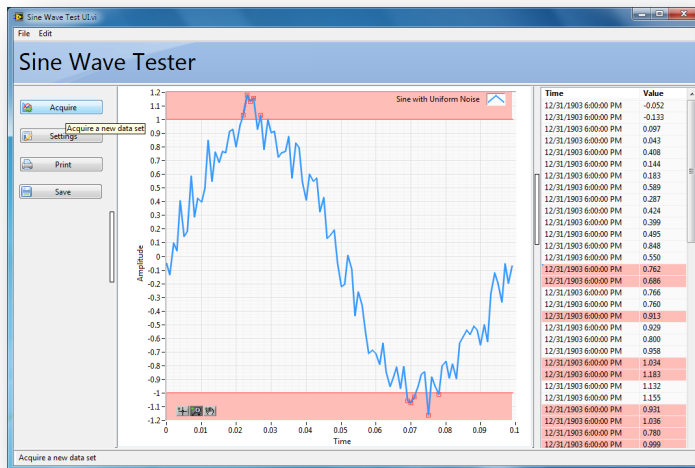
- Too much on screen at once is distracting
- Allow your user to focus on what is important

3. Think About Your User

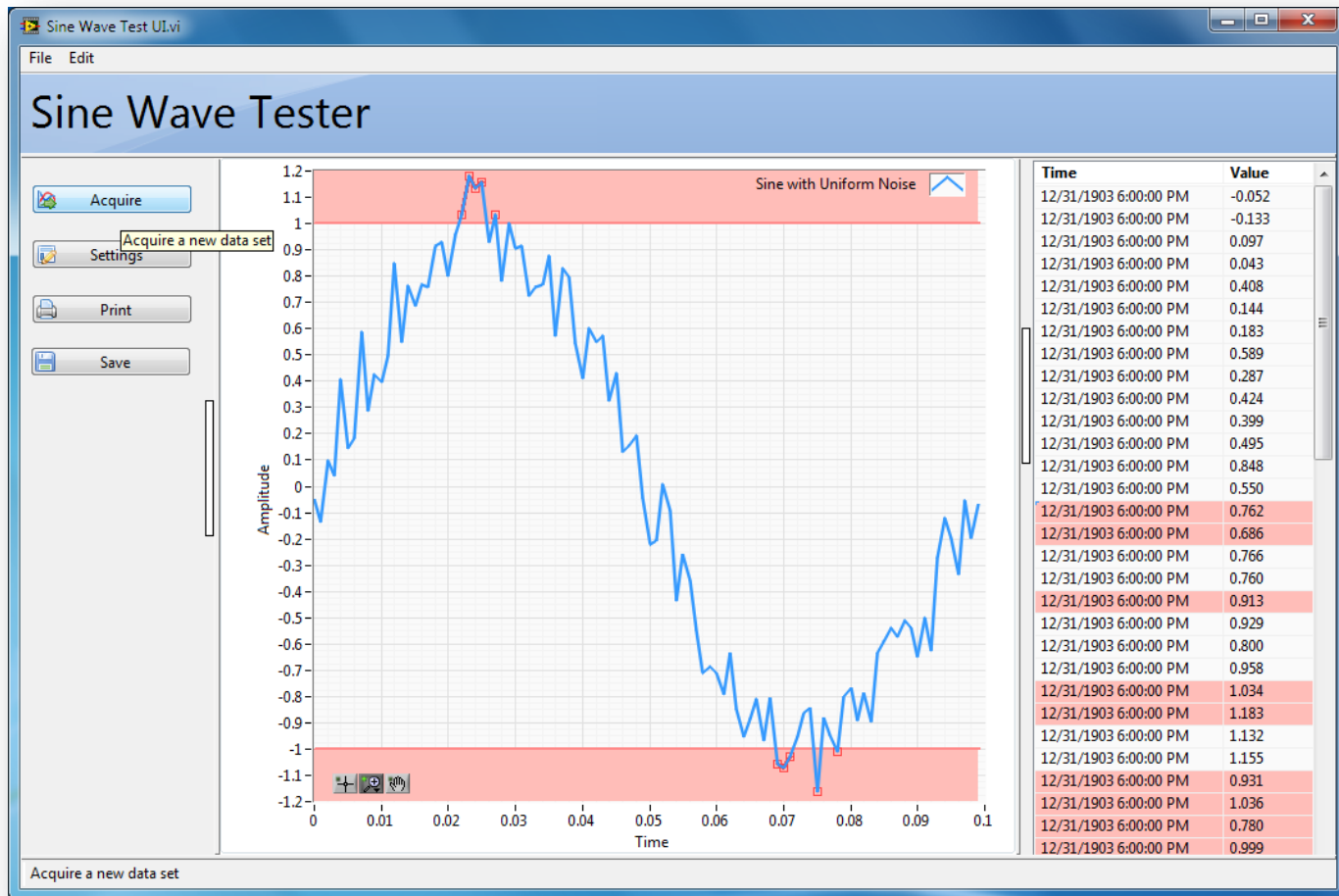
- They probably don't know as much as you
 - Explain what buttons do
 - Keep them informed about what your program is doing
- Know how the user plans on using your application
 - Mouse, Keyboard?
 - Touch Screen → Large Buttons
 - Outdoors → High Contrast

Let's Take a Look at Some UIs

- Windows Desktop App
- Small Touch Screen App
- Informative Kiosk Display



Windows Desktop App



Applying the Rules

Desktop Windows Application

Don't be innovative

- Use System Controls
- Add familiar icons to task buttons
- Use X to close application

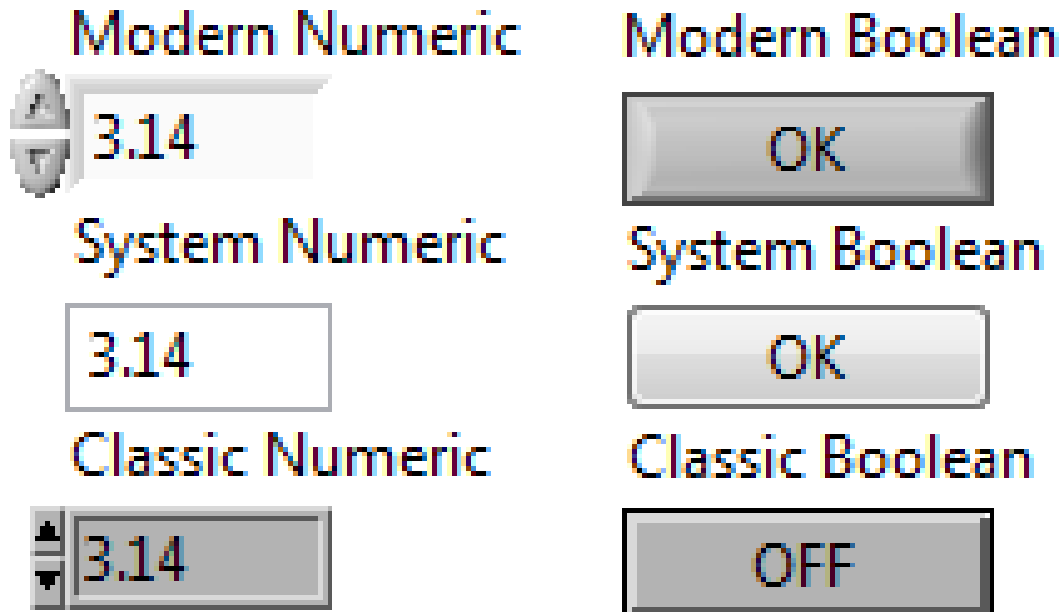
Less is more

- Allow user to hide less important displays
- Hide the LV toolbar
- Don't persist one-time configuration controls for no reason → use temporary dialogs
- Customize the run-time menu

Think About Your User

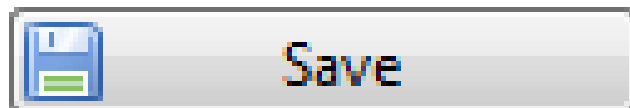
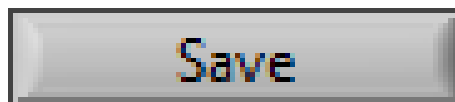
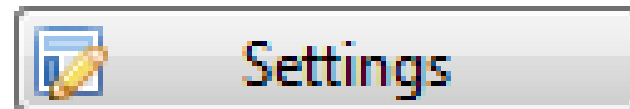
- Create a status bar and use the busy cursor to update user
- Use tooltips to clarify functionality
- Allow the user to cancel long tasks
- Use panes to let user resize your application

Use Appropriate Controls



You can change your default type in **Tools»Options»Front Panel**

Add Decals to Buttons



Add Decals to Buttons - Demo

<http://www.youtube.com/watch?v=2NdqXh67mak>

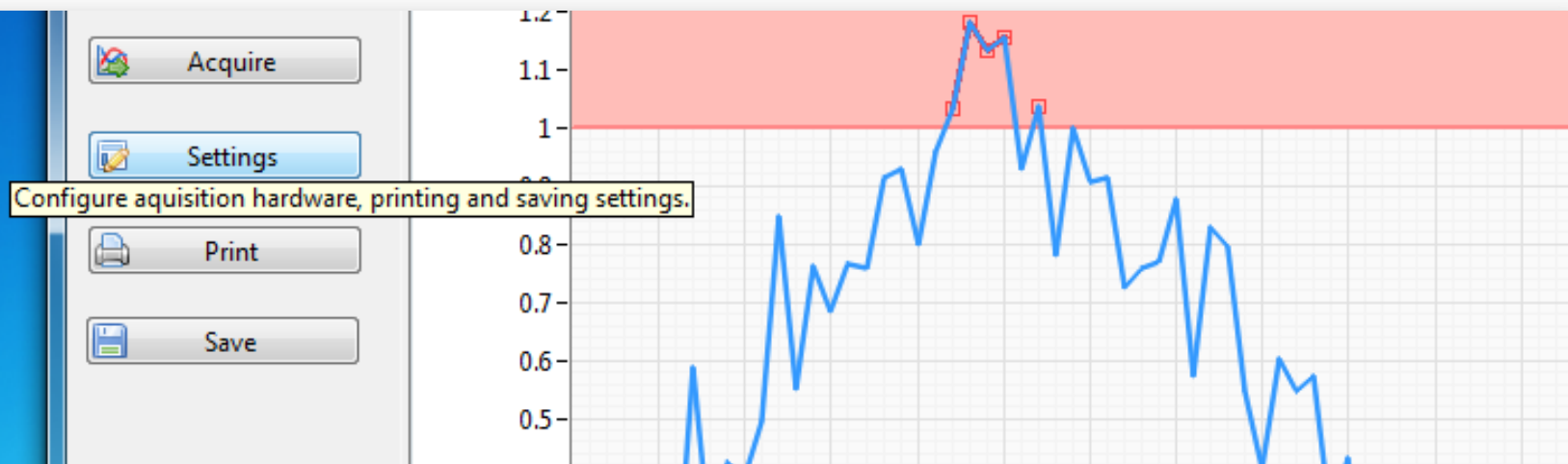
How to Add Decals to Buttons

1. Drop a System OK Button on the front panel.
2. Right click on the control and select **Advanced»Customize...**
3. **Edit»Import Picture to Clipboard** and select your image -
 - PNG files work best since transparency is preserved
4. Right click on the control and select **Import Picture from Clipboard»Decal**
5. Click the wrench icon to change customize mode
6. Move the decal or the text so they aren't overlapping
7. Save your custom control

Reuse tip:

If you give your customized control an icon and save it in **<LabVIEW>\user.lib** it will show up in the controls palette under **User Controls**.

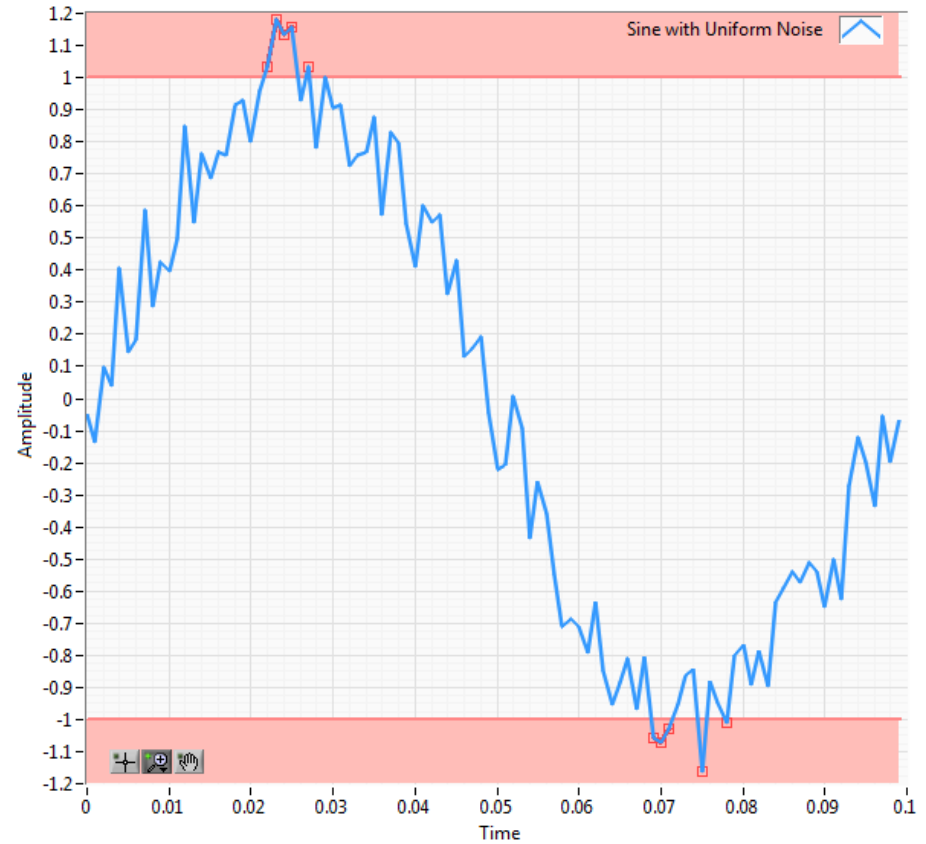
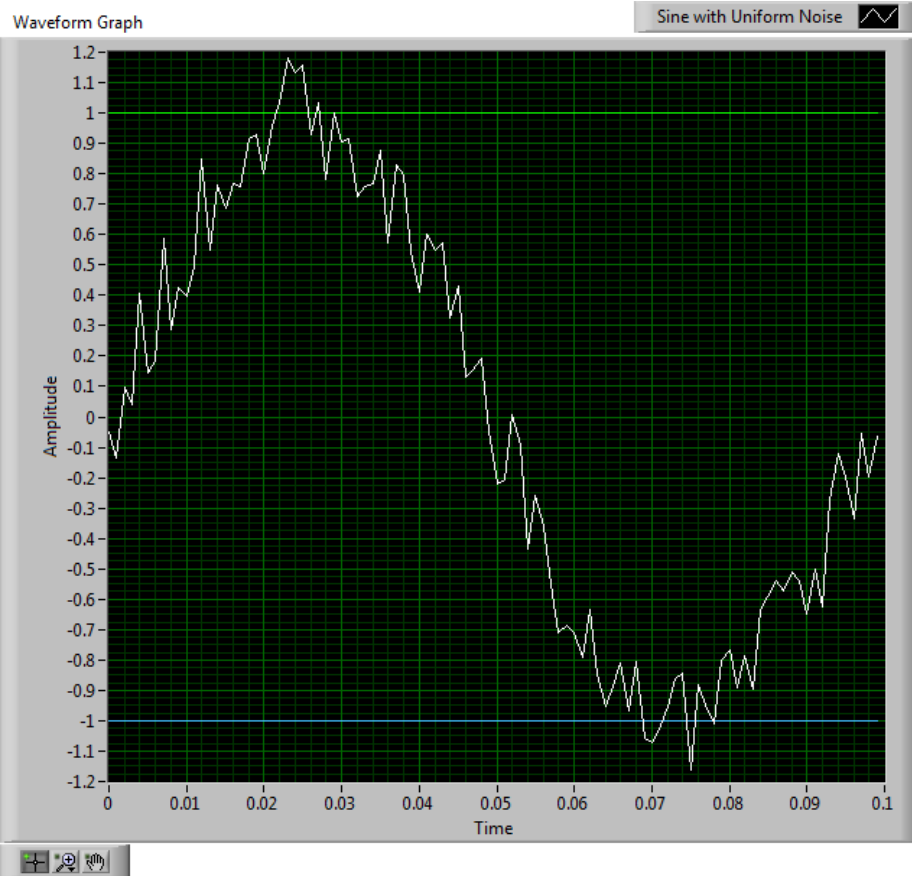
Tooltips



Tooltips - Demo

<http://www.youtube.com/watch?v=NGeElmr1q2g>

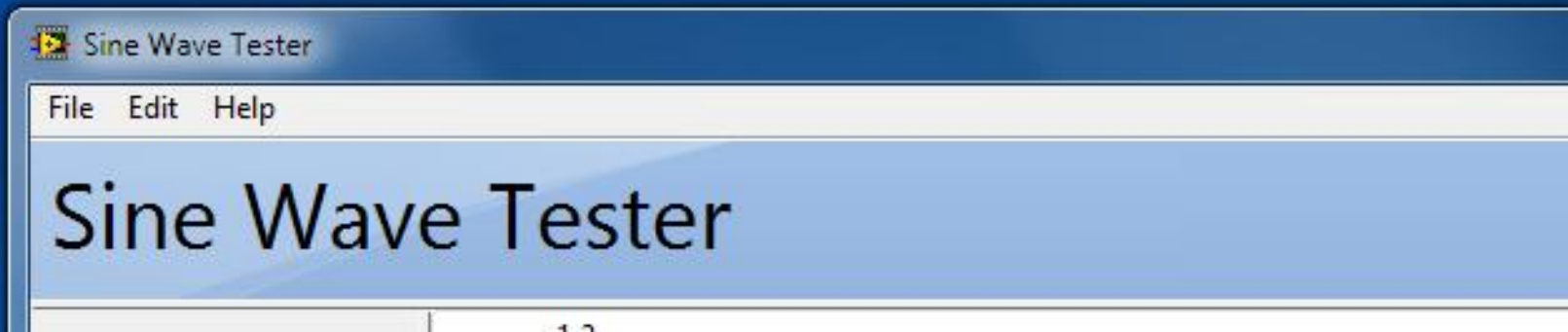
Recolor Graphs



Recolor Graphs - Demo

<http://www.youtube.com/watch?v=rOUcBvyHj5E>

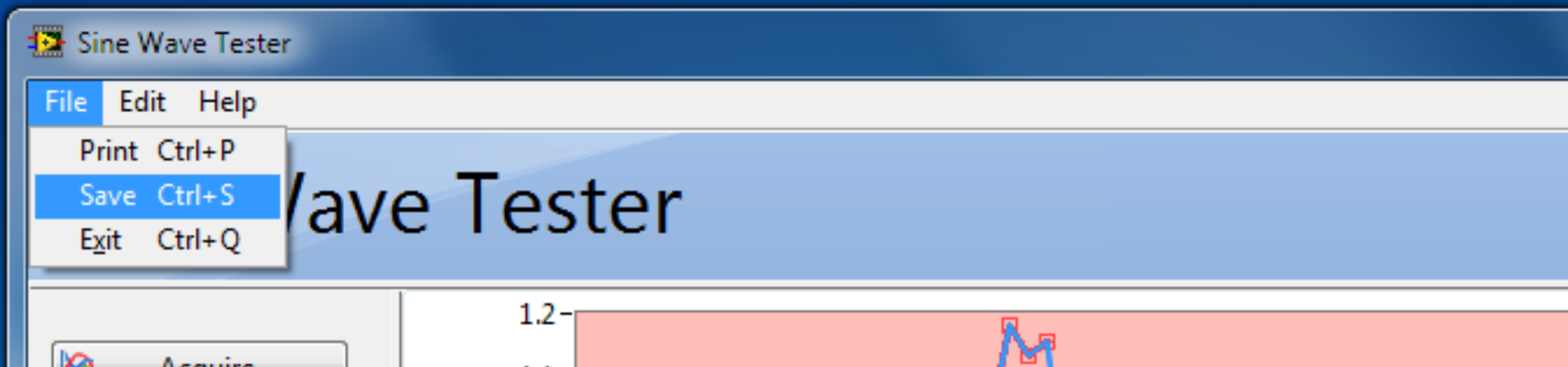
Hide the LabVIEW Toolbar



Hide the LabVIEW Toolbar - Demo

<http://www.youtube.com/watch?v=2CbKuBVGzo0>

Customizing the Run-Time Menu

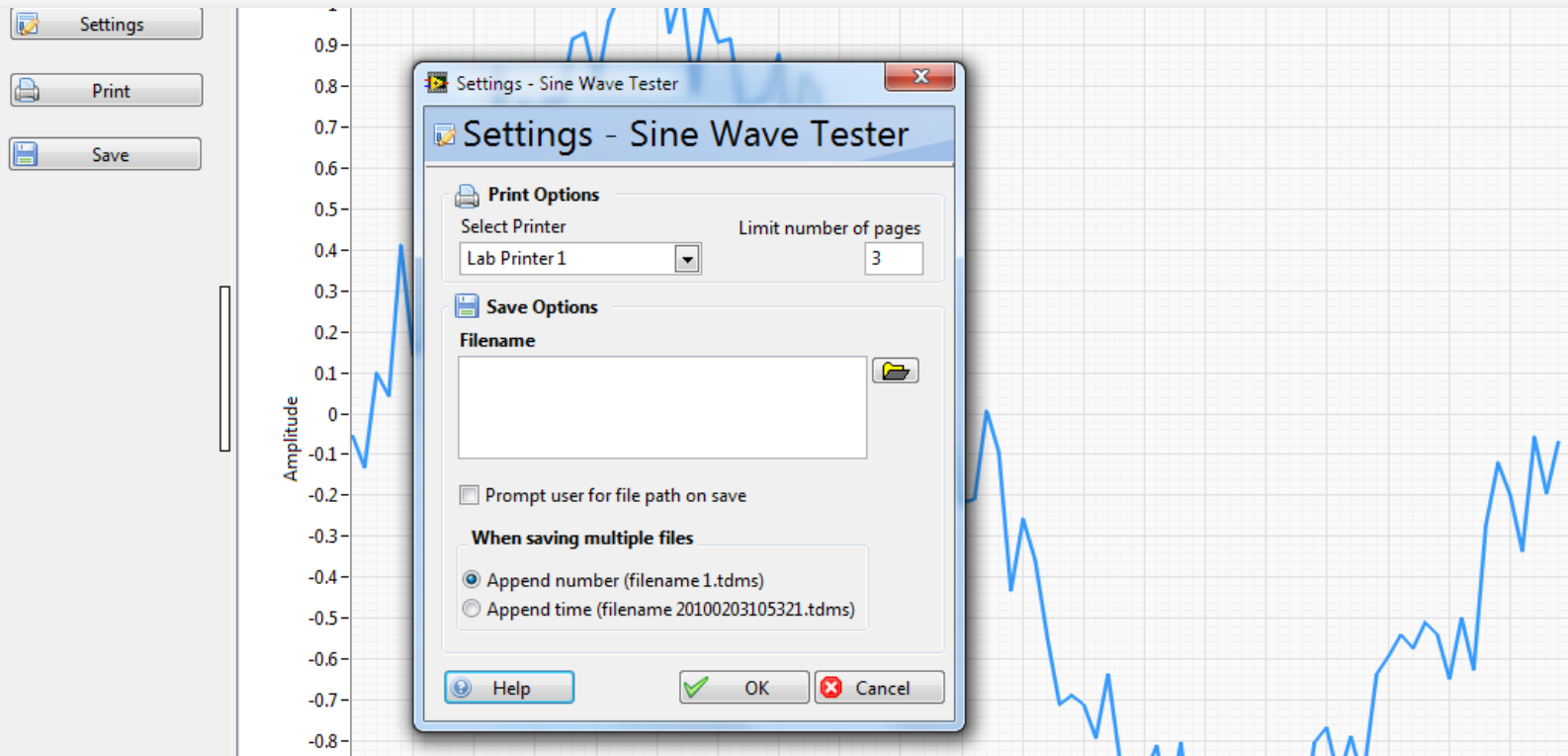


Customizing the Run-Time Menu - Demo

<http://www.youtube.com/watch?v=wkpiAmHFddM>

You can also customize the run-time shortcut (right click) menus for any control by right-clicking on the control and going to Advanced»Run-Time Shortcut Menu»Edit.

Spawning Dialogs

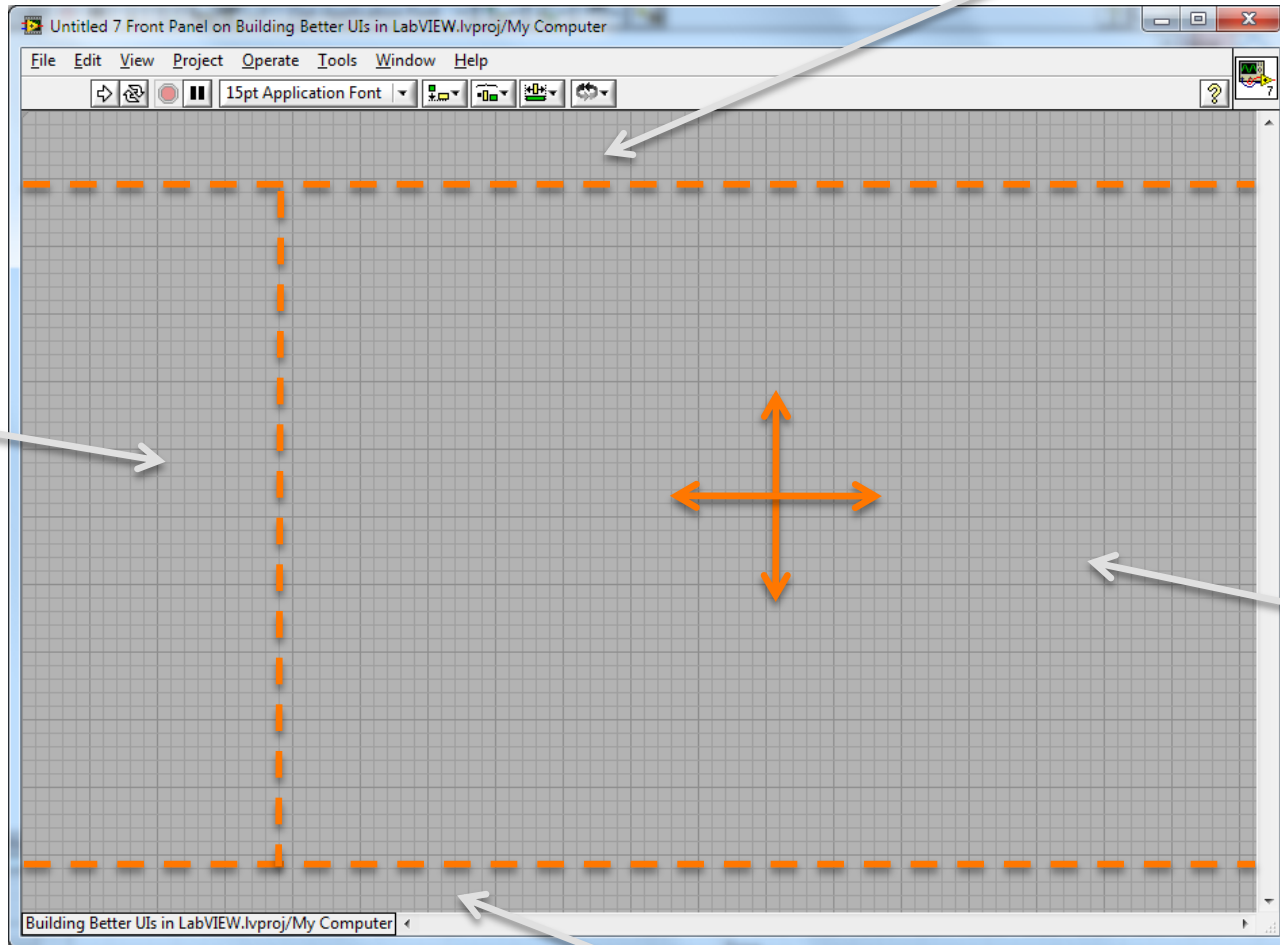


Spawning Dialogs - Demo

<http://www.youtube.com/watch?v=L-hNmzQ9tFc>

Using Panes

Title Area



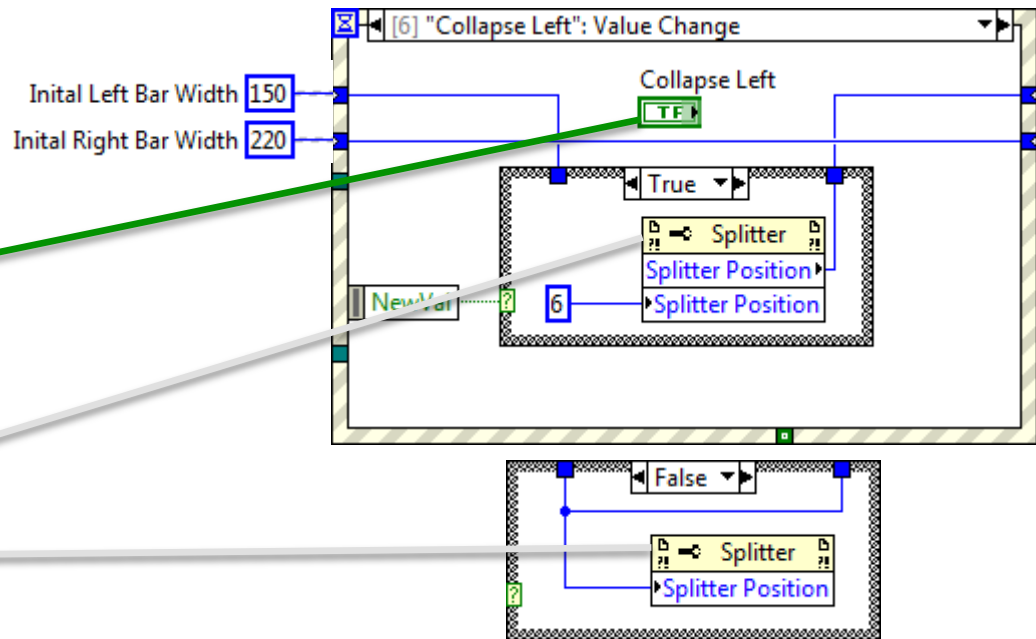
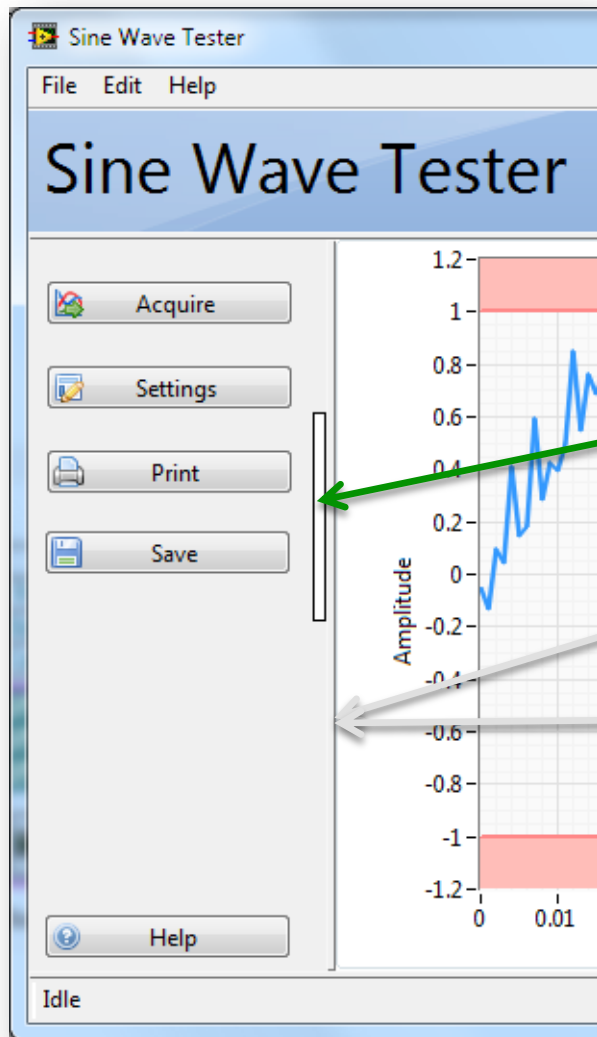
Expandable Content Area

Status Bar

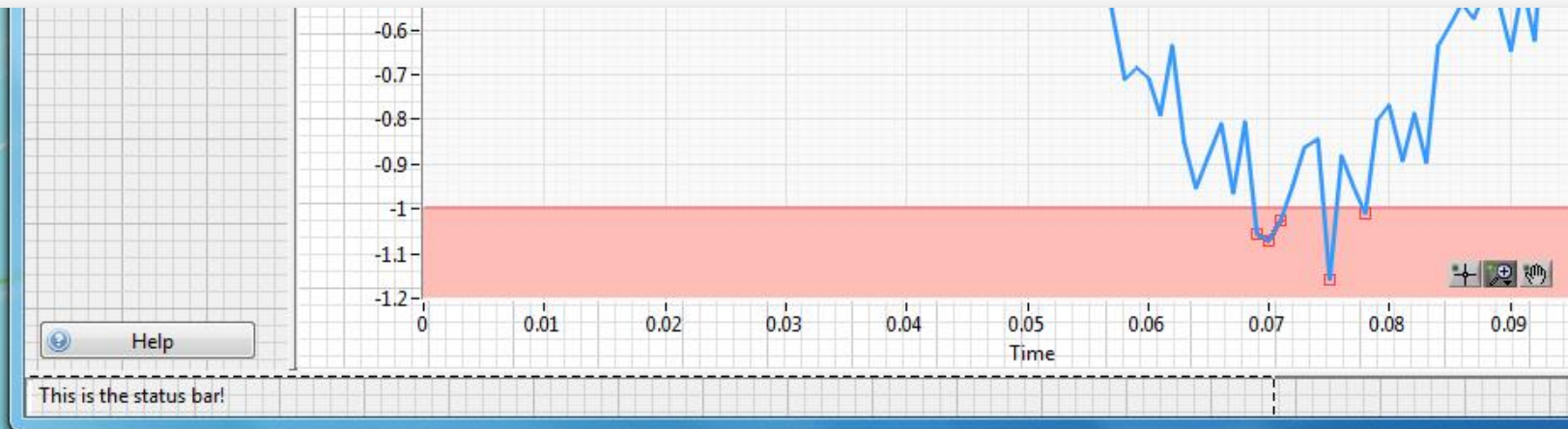
Using Panes – Demo

<http://www.youtube.com/watch?v=hZ180R7ADto>

Hiding Panes



Status Bar



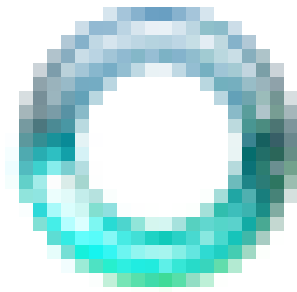
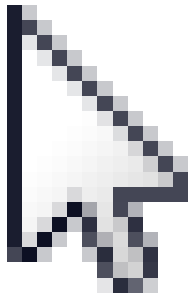
abc Status Bar Text

This is the status bar! Status Bar Text

Another special-use case for panes can be to create a persistent status bar. In the bottom left of your status bar add a string control and simply update the text contained in the status bar via local variable wherever you need to.

This simple technique is one of the most effective ways of keeping your user informed as to what state your application is in or what task it is busy performing.

Busy Cursors

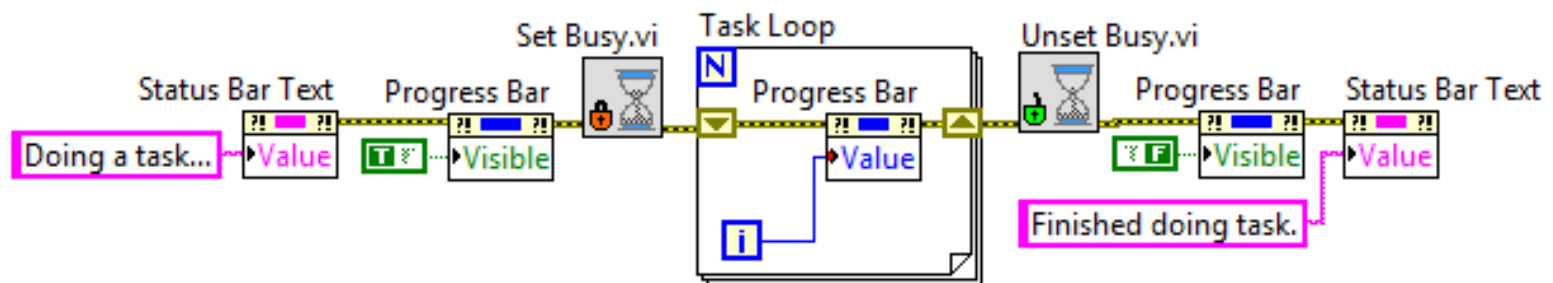
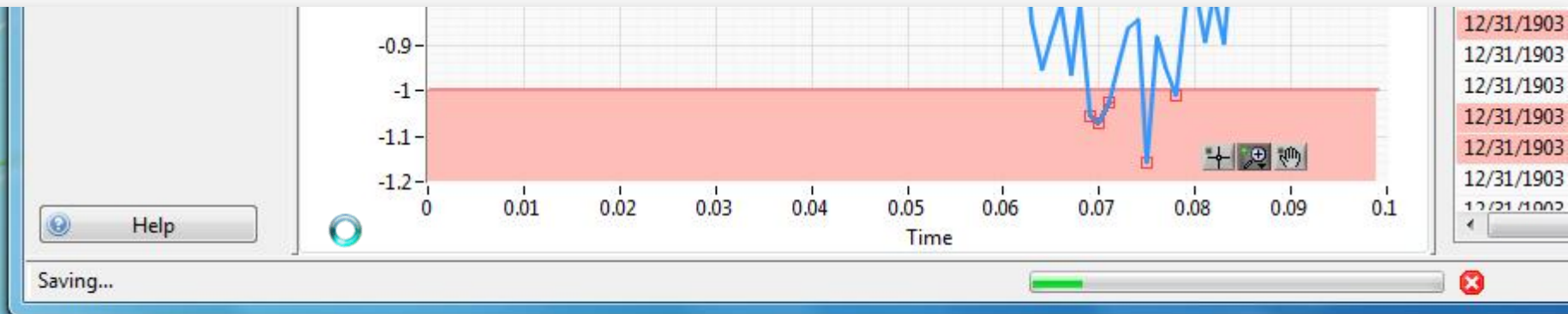


Along the lines of keeping the user updated – LabVIEW lets you change the cursor to a busy cursor programmatically. The busy cursor is an OS-wide UI element that most users are quite familiar with (it means – “wait, I’m trying to do something!”).

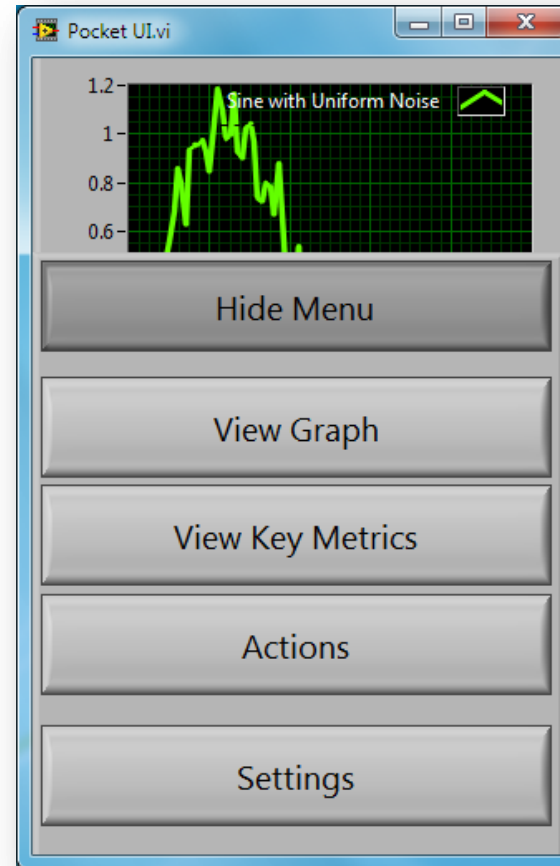
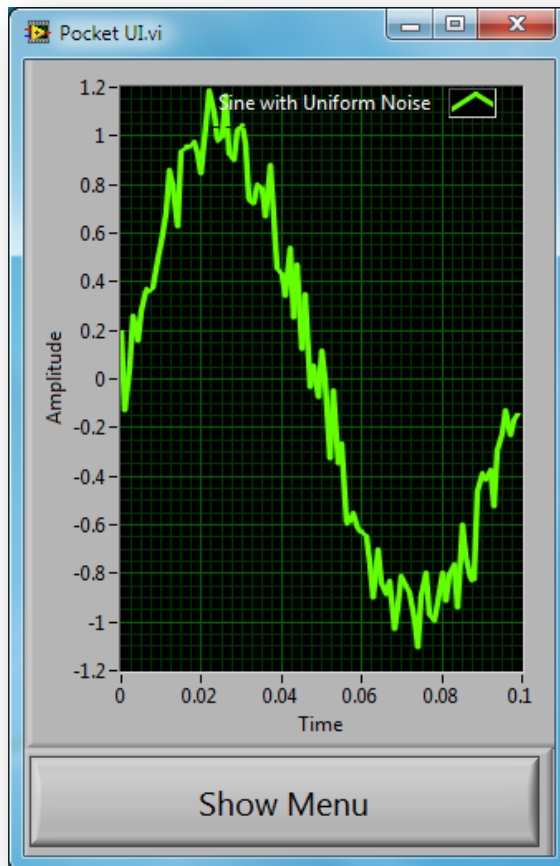
Busy Cursors - Demo

http://www.youtube.com/watch?v=_mosr-oTgRM

Keeping the User Updated



Small Touch Screen App



Applying the Rules

Small Touch Screen

Don't be innovative

- Use large controls & indicators that resemble their physical equivalents
- Simple is best

Less is more

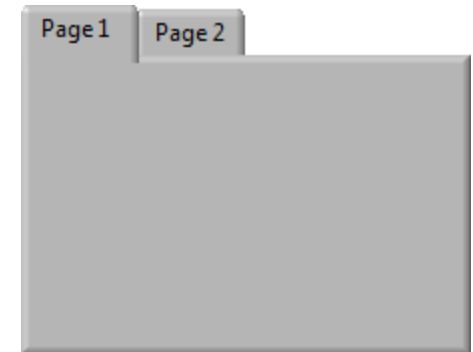
- Screen real-estate is valuable, use it wisely
- Use trays, tabs or different screens to stretch screen space

Think About Your User

- Glare may be an issue → use more contrast
- Touch screens require more spacing
- Users fingers may obscure part of the screen

Tab Controls

- Tab controls are a familiar way to put more information on a screen than would otherwise fit
- Since the tabs themselves can be hidden and changed programmatically they are also useful for some less obvious UI techniques



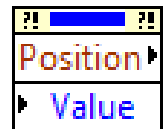
Tab Control



Tab Control

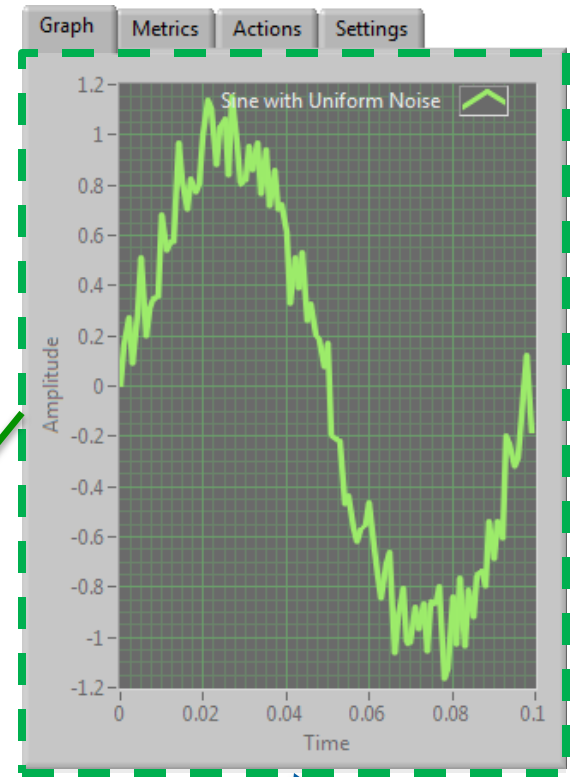
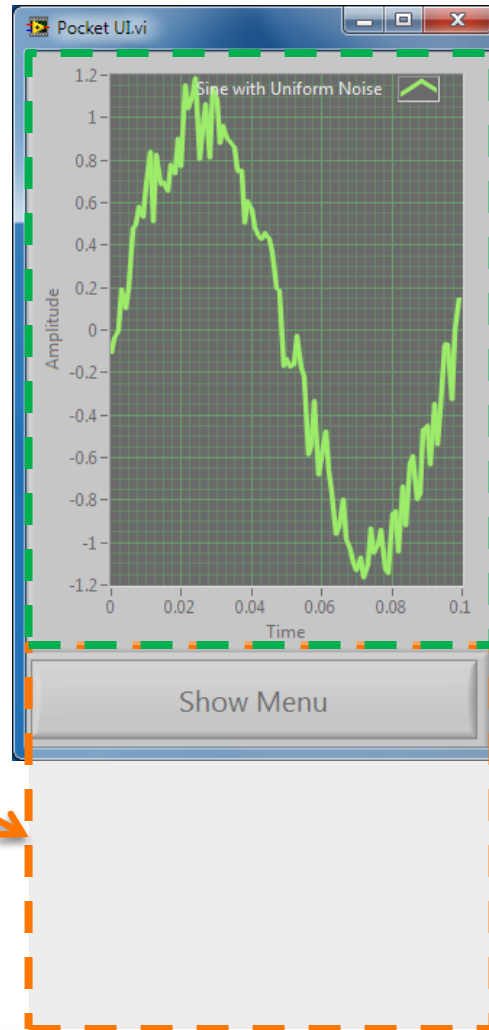
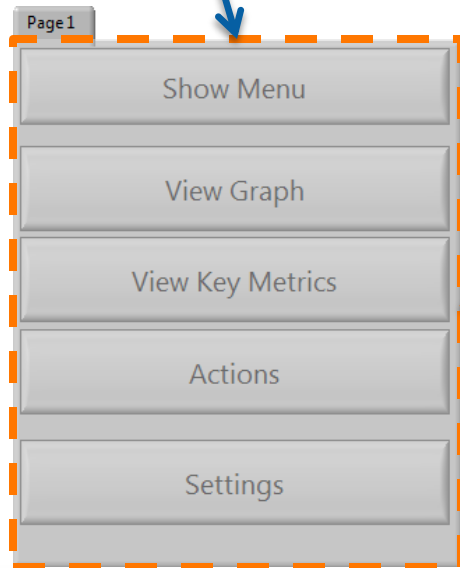


Tab Control



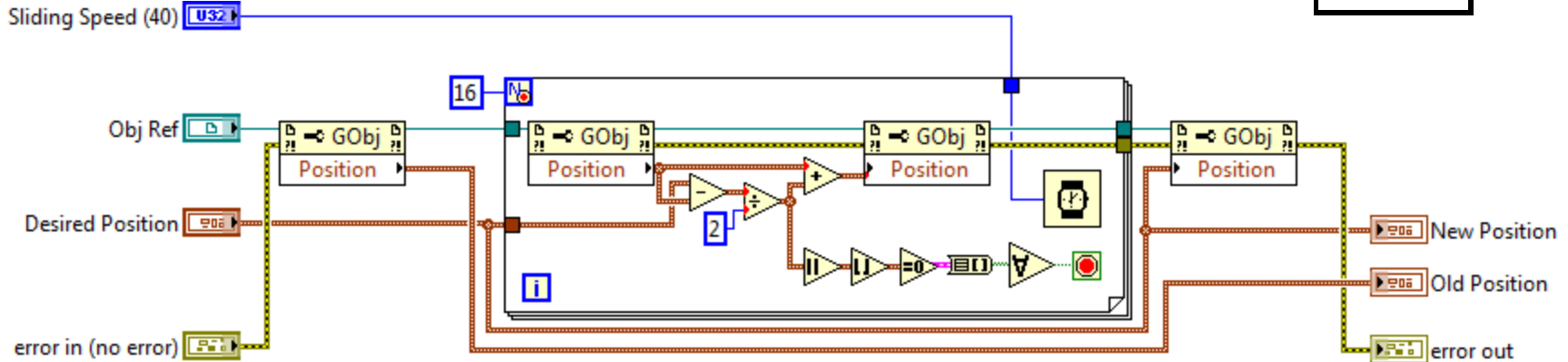
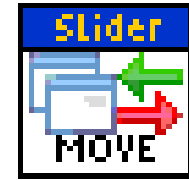
“Hidden” Tab Controls

Sliding Tab Control



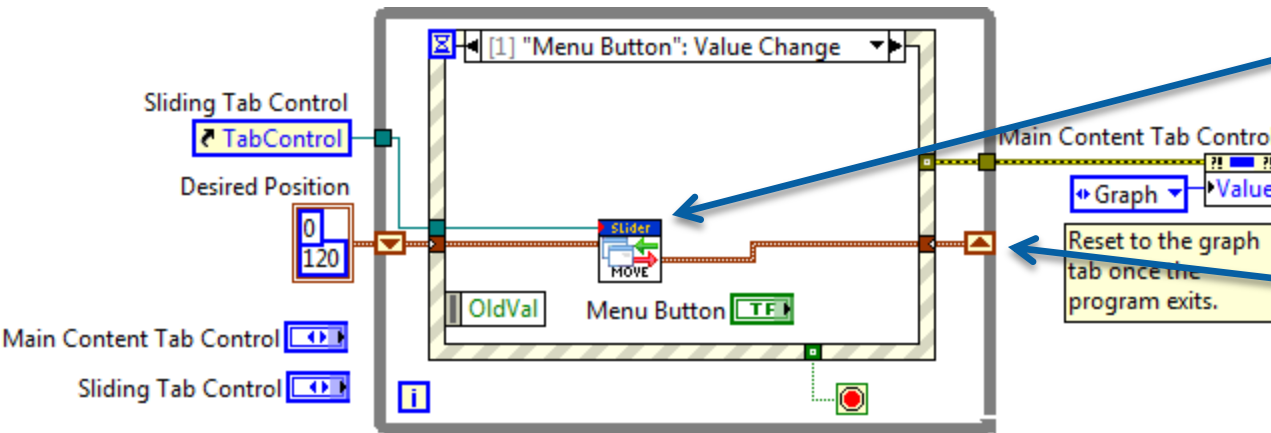
Main Content Tab Control

Sliding a Control – Move.vi



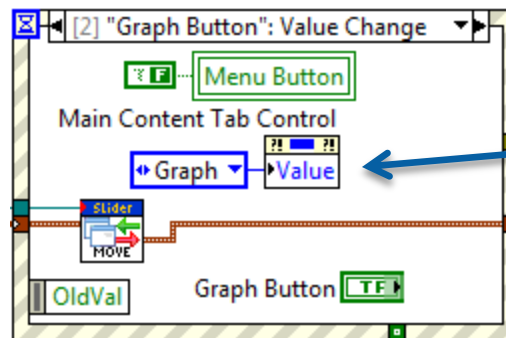
- Moves an object to the Desired Position
- Moving half the remaining distance in each loop iteration gives a natural sliding appearance

Putting It Together



When the Menu Button is clicked slide the invisible tab control into view.

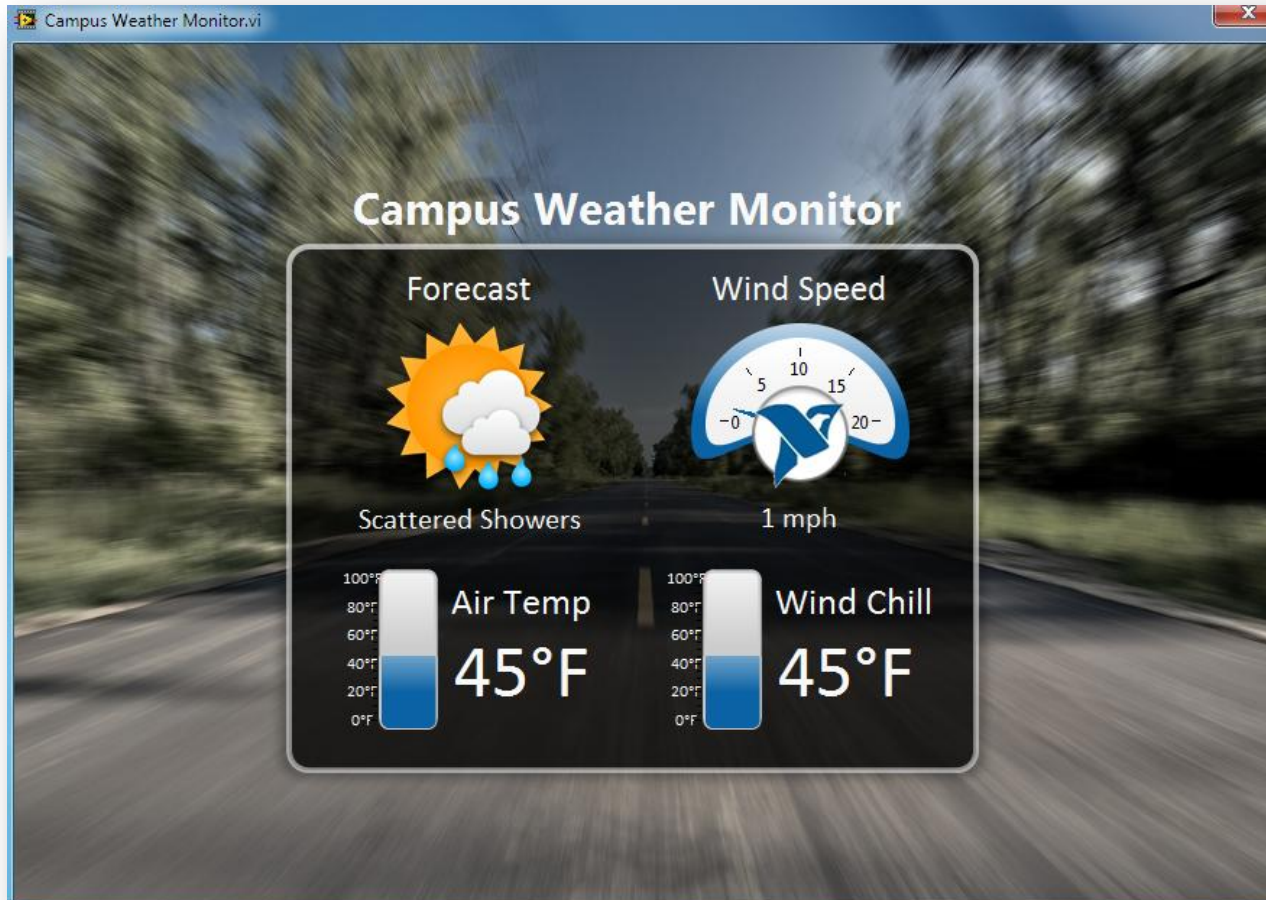
Store the old position of the menu so we can slide it back.



When a new view is selected, change the Main Content Tab Control to the selected page

Again, store the old position of the menu so we can slide it back.

Informative Kiosk Display



Applying the Rules

Informative Console Display

Don't be innovative

- Take inspiration from TV, websites or similar applications

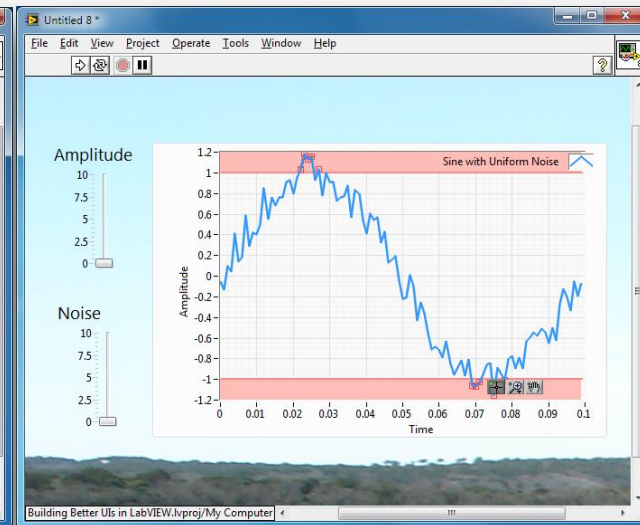
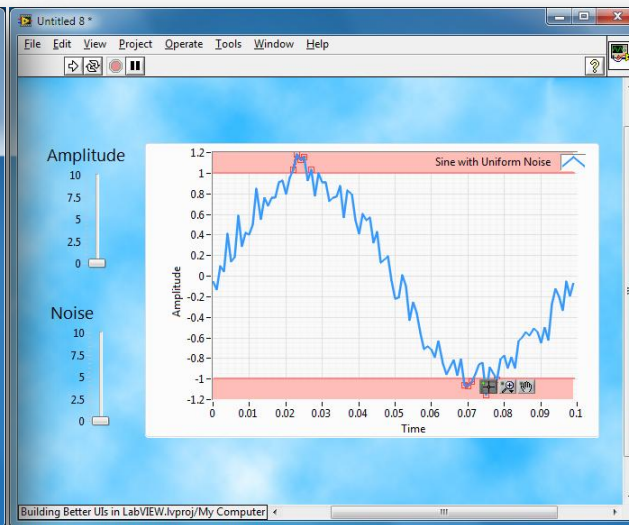
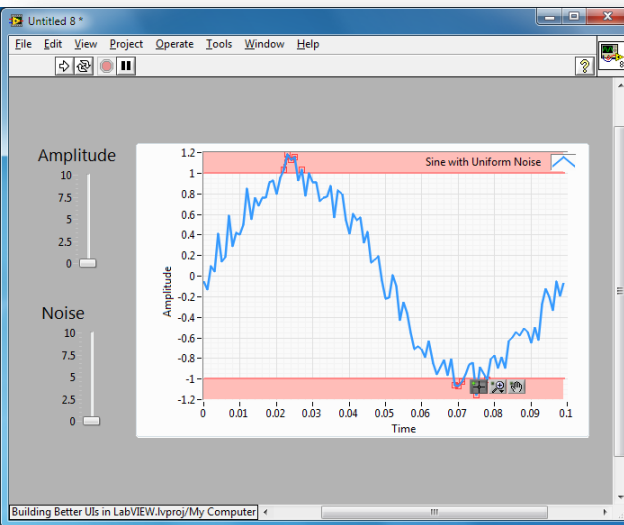
Less is more

- Show only the important information in an instantly recognizable way

Think About Your User

- Passive audience
→ visual appeal is more important

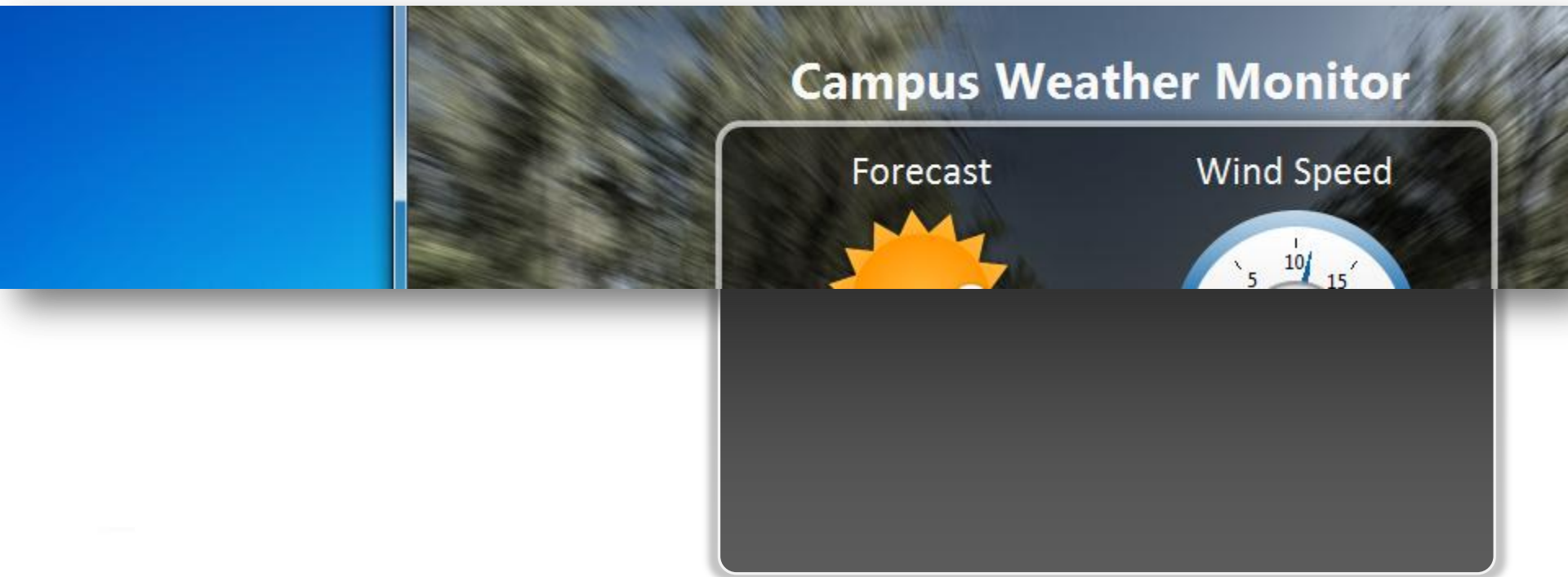
Panel Background



Panel Background - Demo

<http://www.youtube.com/watch?v=gxXJfonTIFc>

Create Decorations in PowerPoint



Create Decorations in PowerPoint - Demo

<http://www.youtube.com/watch?v=gjYfghlv2hQ>

Transparent Indicators



Transparent Indicators - Demo

<http://www.youtube.com/watch?v=jgUB1oDmf-4>

Transparent PNGs in a Picture Ring

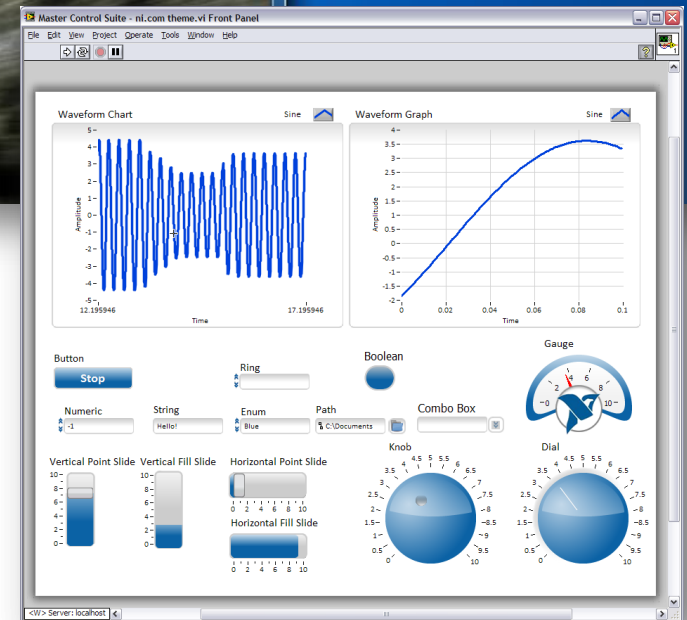
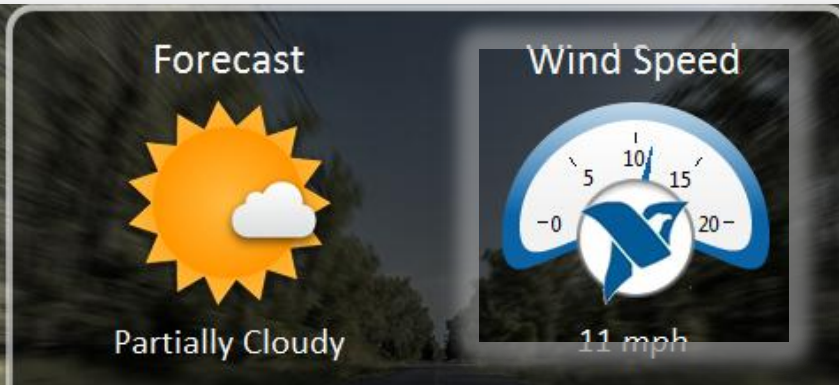


The weather icon indicator is a special case of the previous technique. By combining a transparent picture ring control with some transparent PNGs you can create some pretty powerful, nice looking graphics which will add some visual appeal to your application and also increase the usability of your UI by providing instantly recognizable icons for various states.

Transparent PNGs in a Picture Ring - Demo

<http://www.youtube.com/watch?v=AmDLCsnOegw>

Heavily Customized Controls



Most of the default controls in LabVIEW can be customized by replacing most of the source imagery using the control editor. Since doing so requires a fair amount of artistic ability and knowledge of external images editors we aren't going to cover exactly how to create your own controls in this presentation.

This particular control is part of the ni.com Inspired Control Suite available for download from the ni.com community. In the community you can also find a tutorial explaining the details of how to create customized controls.

Free Stuff - UI Interest Group

The image displays a collage of LabVIEW graphical user interface (GUI) examples. In the top left, a window titled 'Master Control Suite - ni.com theme.vi Front Panel' shows two waveform graphs: one with a blue sine wave and another with a blue curve. Below them are various controls like a 'Stop' button, numeric and string inputs, and sliders. In the top right, a window titled 'Untitled 1' shows a yellow sine wave on a 'Waveform Graph' and a control panel with sliders and a 'Group' button. In the bottom left, a window titled 'Car UI.vi' features a car dashboard simulation with a speedometer, a tachometer, and a fuel gauge, along with a 'Ni Noise Engrapy' label and a '1600 m' display. In the bottom right, a window titled 'Tab 1' shows a blue waveform graph and a control panel with 'Amplitude' and 'Frequency' sliders, and a 'Stop' button. A central semi-transparent box contains a stylized 'UI' logo with a yellow triangle and a red arrow, and the URL <http://decibel.ni.com/content/groups/ui>.

Key Take Aways

- The “Rules”
 1. Don't be innovative
 2. Less is more
 3. Think about your user
- Take advantage of what LabVIEW gives you:
 - Transparency
 - Different Controls/Control Customization
 - Panes/Tabs
- UI Interest Group on the community