



Tucson Python Meetup

Graphical User Interface (GUI) Intro/Demo

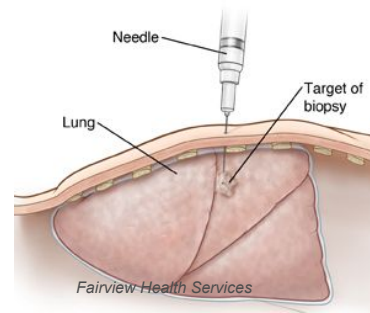
Franklin Ventura

November 15, 2016



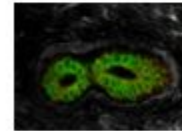
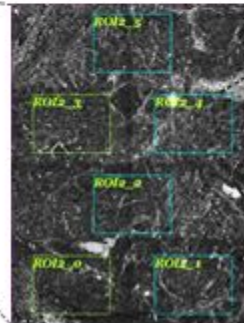
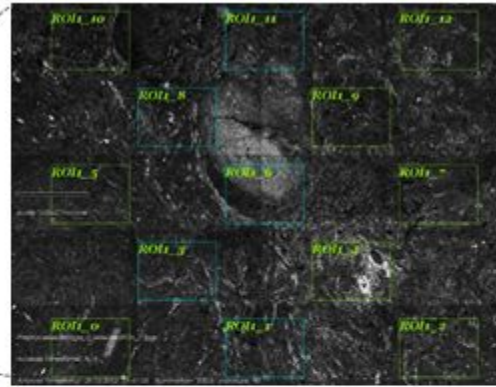
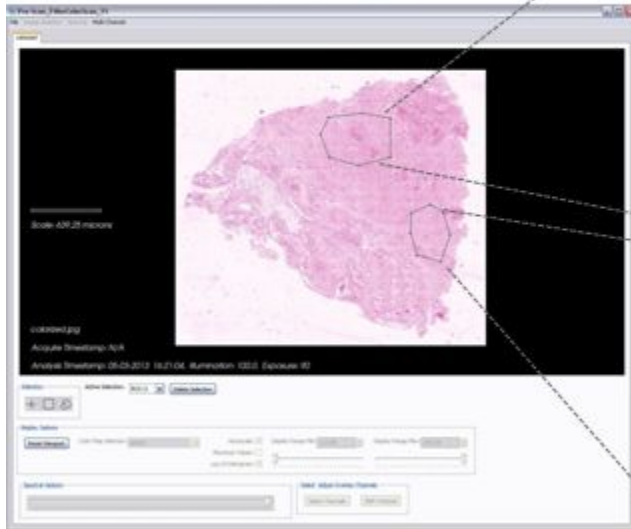
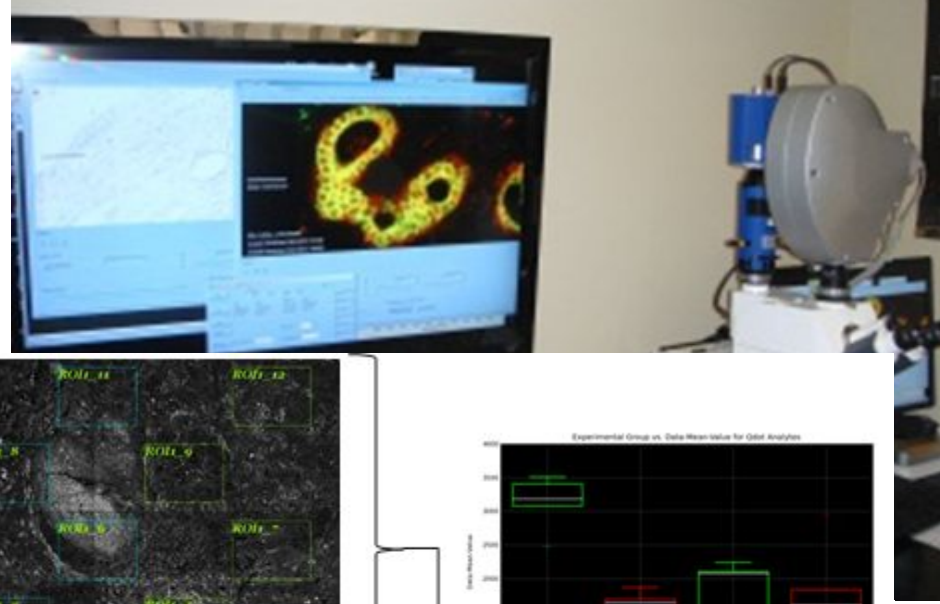
Franklin Ventura

- BS Biomedical Engineering UofA
- MS Software/Systems Engr ASU
- Ventana Medical Systems 5 yrs
 - Biomedical/Software Engineer
 - Technology and Applied Research
 - Digital Pathology
 - Cancer Diagnostics
- Applied Imaging Systems Team
 - Karl Garsha, Taras Golota



Research Development

- ◎ Advanced Research Needs
- ◎ Flexible System
- ◎ Easy to Use



```

acq_setting(func_name='set_plane', func_args={'plane': (0.0, 0.0), 'index': 0}),
acq_setting(func_name='set_slice', func_args={'method': 'ABSOLUTE', 'z_pos': 0.0}),
acq_setting(func_name='set_channel', func_args={'index': 0, 'channel': 'channel2'}),
acq_setting(func_name='set_exposure', func_args={'exposure': 20}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 0, 'z_pos': -0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 1, 'z_pos': 0.0}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 2, 'z_pos': 0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_channel', func_args={'index': 1, 'channel': 'channel4'}),
acq_setting(func_name='set_exposure', func_args={'exposure': 40}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 0, 'z_pos': -0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 1, 'z_pos': 0.0}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 2, 'z_pos': 0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_channel', func_args={'index': 2, 'channel': 'channel6'}),
acq_setting(func_name='set_exposure', func_args={'exposure': 60}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 0, 'z_pos': -0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 1, 'z_pos': 0.0}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 2, 'z_pos': 0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_channel', func_args={'index': 3, 'channel': 'channel11'}),
acq_setting(func_name='set_exposure', func_args={'exposure': 10}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 0, 'z_pos': -0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 1, 'z_pos': 0.0}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 2, 'z_pos': 0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_channel', func_args={'index': 4, 'channel': 'channel3'}),
acq_setting(func_name='set_exposure', func_args={'exposure': 30}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 0, 'z_pos': -0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 1, 'z_pos': 0.0}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 2, 'z_pos': 0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_channel', func_args={'index': 5, 'channel': 'channel15'}),
acq_setting(func_name='set_exposure', func_args={'exposure': 50}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 0, 'z_pos': -0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 1, 'z_pos': 0.0}),
acq_setting(func_name='set_image', func_args={'binning': 2}),
acq_setting(func_name='set_slice', func_args={'method': 'RELATIVE', 'index': 2, 'z_pos': 0.5}),
acq_setting(func_name='set_image', func_args={'binning': 2})

```

No one
wants to
have to
work
with this



Or even
this



```

Channels-1:
- 2-AQUA AQUA:
- 6-CY5 CY5:
- 1-DAPI DAPI:
- 4-GOLD GOLD:
- 3-GREEN GREEN:
- 5-RED RED:
Image-1:
  img_type: OME
  method: Reflected
Objectives-1:
- '20.0':
Planes-1:
- - - 1056.709
  - 1714.655
  - -28.0
  - Objectives-1
  - FOV_001
Save Folder: C:\TEST_NEW
Slices-1:
  delta: 0.5
  direction: UP
  end: 0.0
  method: RELATIVE
  slices_num: 7
  start: 0.0
dim_order:
- Planes-1
- Slices-1
- Channels-1
- Image-1

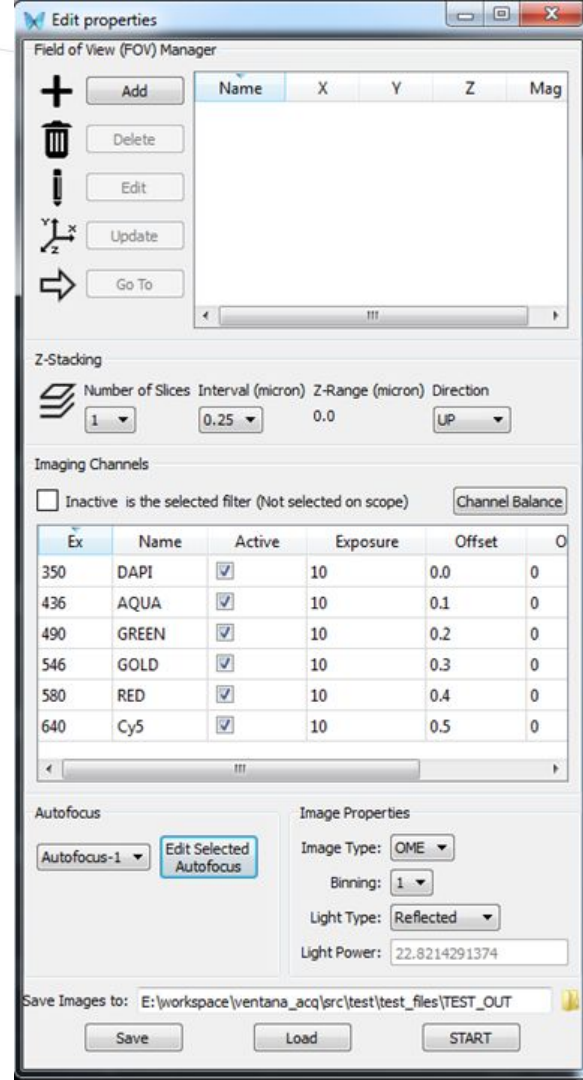
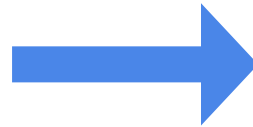
```


We are okay
with this



```
worker = Thread(name='ImagingQueue',
                 target=self.img_engine.image_creator, args=())
worker.setDaemon(False)
worker.start()
self.system.camera.live_stop()
cont, skip = True, False
for i, acq in enumerate(acq_settings):
    logger.debug("Acquisition: %s Args: %s", acq.func_name,
                 acq.func_args)
    getattr(self, acq.func_name)(**acq.func_args)
    if progress is not None:
        (cont, skip) = progress.update(i)
        if not cont or skip or cont is None:
            self.img_engine.stop = True
            return False
worker.join()
return self.img_engine.fpath, self.img_engine.image_paths
```

Non-programmers
want this



Many Options for GUI development -

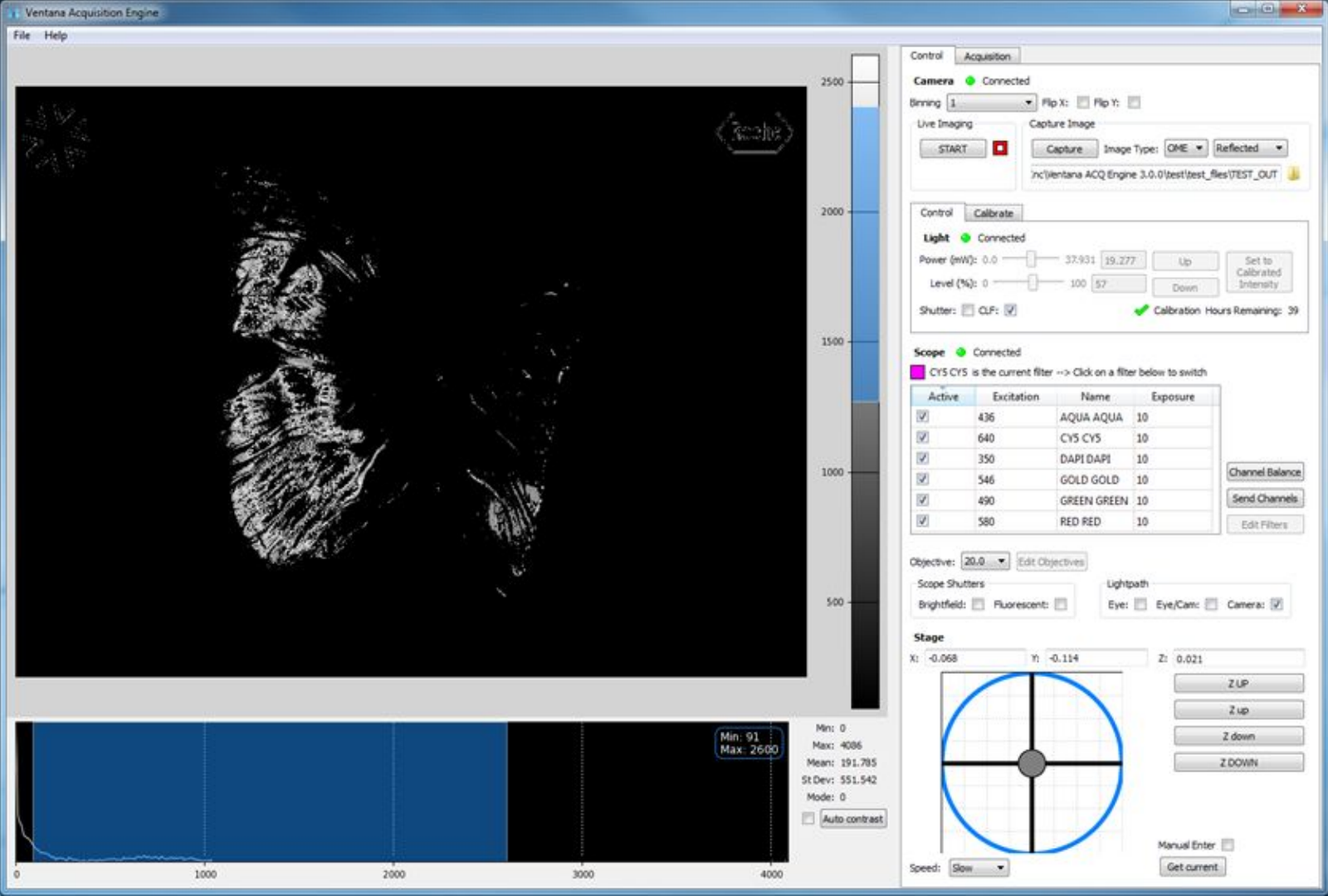
<https://wiki.python.org/moin/GuiProgramming>

- ◎ Guidata
- ◎ Kivy
- ◎ PyForms
- ◎ PyQt
- ◎ Pywebview
- ◎ TkInter
- ◎ Traits/TraitsUI
- ◎ VTK
- ◎ WxPython
- ◎ PyJamas
- ◎ PyMUI
- ◎ PyWebkitGtk
- ◎ PySide
- ◎ Pyglet

What I use/used







Scale: 104.62 microns

DefaultFilename

Acquire Timestamp: N/A

Analysis Timestamp: 11-15-2016 13:42:59

Selection

Selection Type

☒ 'AND' ROI Type ☐ 'OR' ROI Type ☐ 'NOT' ROI Type

Selection Tools



Edit Selections

Active Selection: No ROIs ▾ Delete Selection

Display Options

Reset Viewport

Color Map Selection: WHITE

Sync Zoom/Pan: ☒

Autoscale: ☒

Display Range Min: 1.00

Display Range Max: 1.00

Maximum Values: ☐Log 10 Histogram: ☒

Spectral Options

Select **_Adjust Overlay Channels**

Select Channels

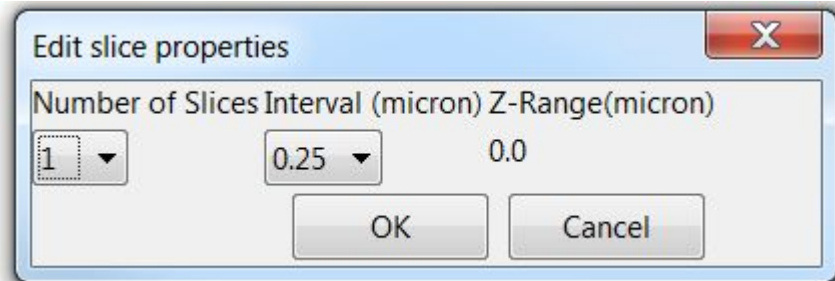
Shift Channel



wxPython

```
number = wx.Choice(self, ['1', '3', '5', ...])
interval = wx.Choice(self, ['0.25', '0.5', '0.75', ...])
number_text = wx.StaticText(self, "Number of Slices")
interval_text = wx.StaticText(self, "Interval (micron)")
zrange_text = wx.StaticText(self, u"Z-Range(micron)")
zrange = wx.StaticText(self, "0.0")
```

```
app = wx.App(False)
dialog = slices_example(None)
dialog.Show(True)
app.MainLoop()
```

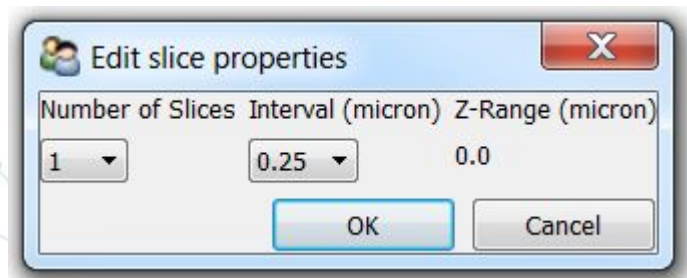


```
sizer = wx.GridBagSizer(3, 3)
sizer.Add(self.number_text, pos=(0,0))
sizer.Add(self.interval_text, pos=(0,1))
sizer.Add(self.zrange_text, pos=(0,2))
sizer.Add(self.number, pos=(1,0))
sizer.Add(self.interval, pos=(1,1))
sizer.Add(self.zrange, pos=(1,2))
```

TraitsUI/PyQT

```
number = Enum(1, 3, 5, ...)  
interval = Enum(0.25, 0.5, ...)  
zrange = Float  
slice_image = Image(paths.slice)
```

```
dialog = slices_example()  
dialog.configure_traits()
```



```
View(HGroup(  
    Item('slice_image', show_label=False),  
    VGroup(  
        Item(label='Number of Slices'),  
        Item('number', show_label=False,  
             width=0.25)),  
    VGroup(  
        Item(label='Interval (micron)'),  
        Item('interval', show_label=False,  
             width=0.25)),  
    VGroup(  
        Item(label='Z-Range (micron)'),  
        Item('zrange', style='readonly',  
             show_label=False,  
             width=0.25)))
```


View Connection to Model/Controller

wxPython

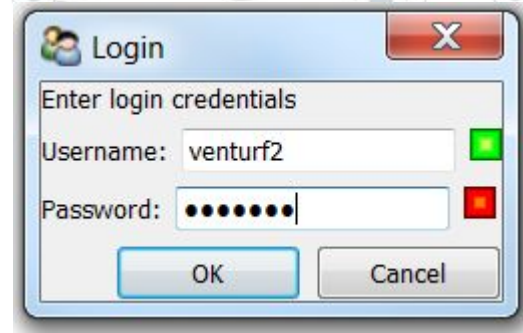
```
self.number.Bind(wx.EVT_CHOICE, self._zrange_update)
self.interval.Bind(wx.EVT_CHOICE, self._zrange_update)
```

```
def _zrange_update(self, event):
    num = int(self.number.GetStringSelection())
    inter = float(self.interval.GetStringSelection())
    zrange = ( num - 1 ) * inter
    self.zrange.SetLabel(str(zrange))
```

TraitsUI/PyQT

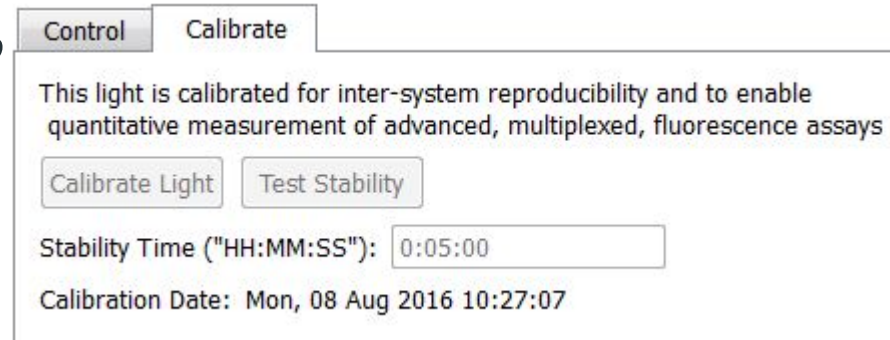
```
@on_trait_change('number, interval')
def _zrange_update(self):
    self.zrange = ( self.number - 1 ) * self.interval
```

Password Control



```
global_access = Bool(False)  
access = Property(Bool, depends_on=['global_access'])
```

```
Item(name='calibrate_button', label='Calibrate Light',  
      enabled_when='access',  
      show_label=False)
```



Chaco

Interactive data visualization and exploration

- ◎ Flexible drawing and layout
- ◎ Modular and extensible architecture
- ◎ Data model for ease of extension and embedding

```
self.image = ArrayPlotData()  
self.image.set_data(self.image_data)  
plot = Plot(self.image, 'top left')  
plot.tools.append(PanTool(plot))  
zoom = CustomZoomTool(plot, 'box')  
self.image_tool =  
ImageInspectorTool(plot )  
  
im = self.acquire()  
viewer.image_data = im.astype(int)
```

cx_Freeze - Create Executable

- ◎ All dependencies packaged together
- ◎ No Python installed

```
build_info = Executable(  
    script="gui/launchpad.py",  
    base='Win32GUI',  
    targetName="Ventana Acquisition Engine",  
    icon=r"gui\images\icon.ico")
```

```
setup(  
    version=VERSION,  
    name="Ventana Acquisition Engine",  
    executables=[build_info])
```



Inno Setup Compiler

- © Installer File
- © Even easier to distribute





Thank you

- © Karl Garsha
 - © Dustin Harshman
 - © Taras Golota
- 

A decorative network diagram in the top right corner, featuring a complex web of interconnected nodes and lines. The nodes are represented by small circles, some of which are highlighted with a double outline. The lines connecting them are thin and grey.

GUI Workshop

Tuesday November 29, 2016
6 pm Keating Room 107

A decorative network diagram in the bottom left corner, similar to the one in the top right, showing a cluster of nodes and connecting lines.