PYTHON API IN NG

What’s new in NG

6 / 2 / 2016 - Samuli Ahonen

Webinar link: https://attendee.gotowebinar.com/register/1716299542769799681

VISUAL COMPONENTS

- Command Registration
- Localizing commands
- Action Panel
- Selection & Context
- Pick / Snap

VISUAL COMPONENTS

Command Registration

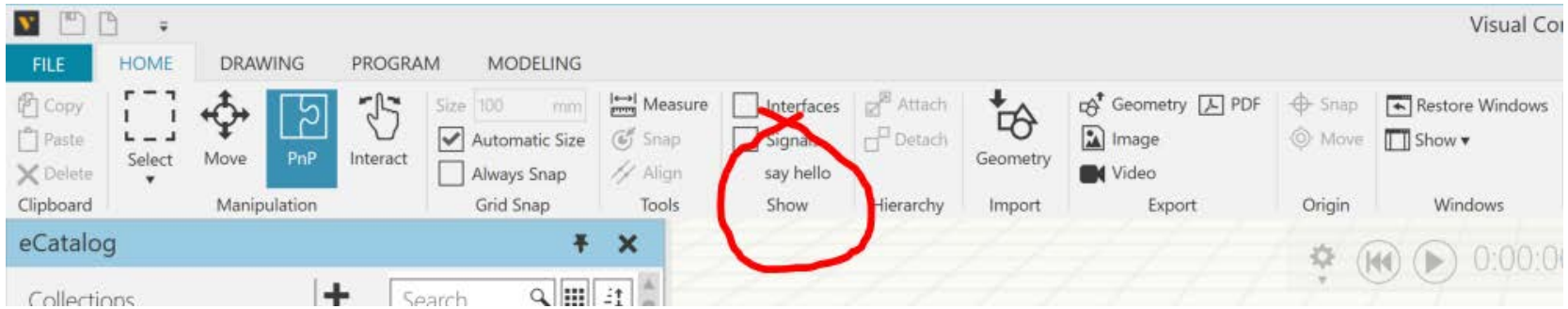
VISUAL COMPONENTS

COMMAND REGISTRATION

- More freedom adding menu items
- Custom buttons in
  - ribbon tabs
  - context menus
  - quick menus
  - …

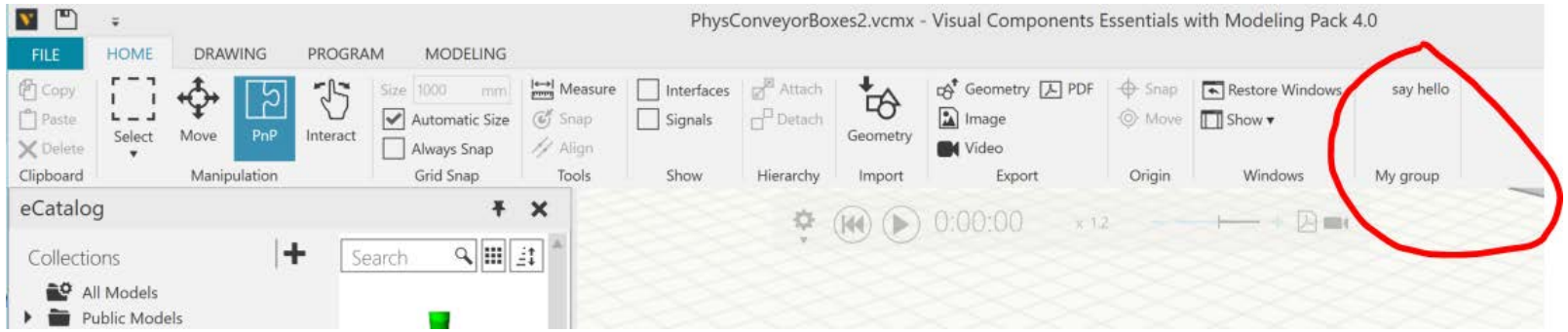
HOME RIBBON

```
cmduri = getApplicationPath() + 'hello.py'
cmd = loadCommand('cmdHello', cmduri)
addMenuItem("VcTabHome/VcRibbonShow", "say hello", -1, "cmdHello")
```



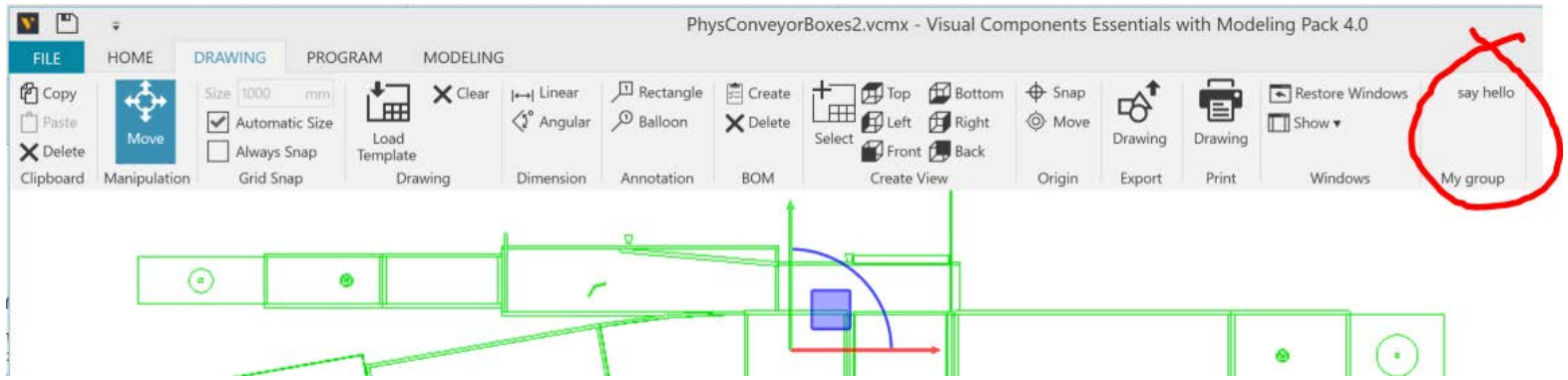
HOME RIBBON

addMenuItem(“VcTabHome/My group”, "say hello", -1, “cmdHello")



DRAWING RIBBON

addMenuItem("VcTabDrawing/My group", "say hello", -1, "cmdHello")



PROGRAM RIBBON

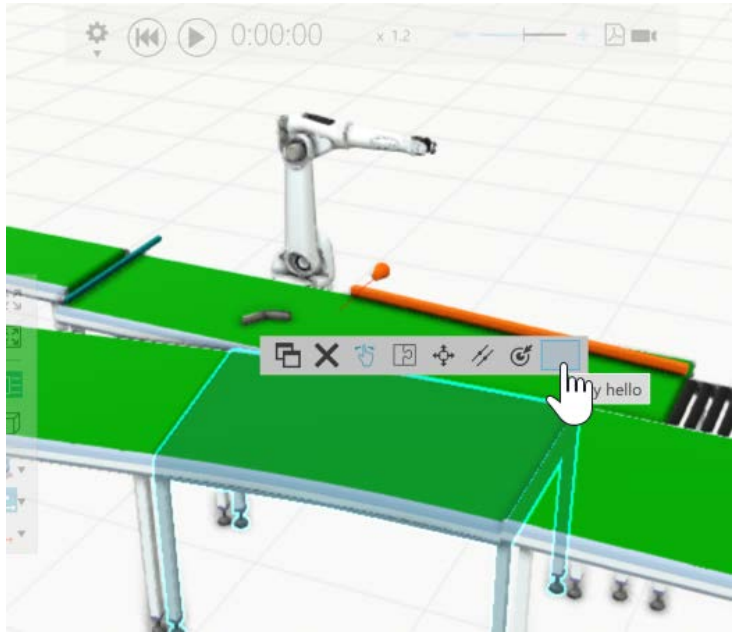
addMenuItem("VcTabTeach/My group", "say hello", -1, "cmdHello")

MODELING RIBBON

addMenuItem("VcTabAuthor/My group", "say hello", -1, "cmdHello")

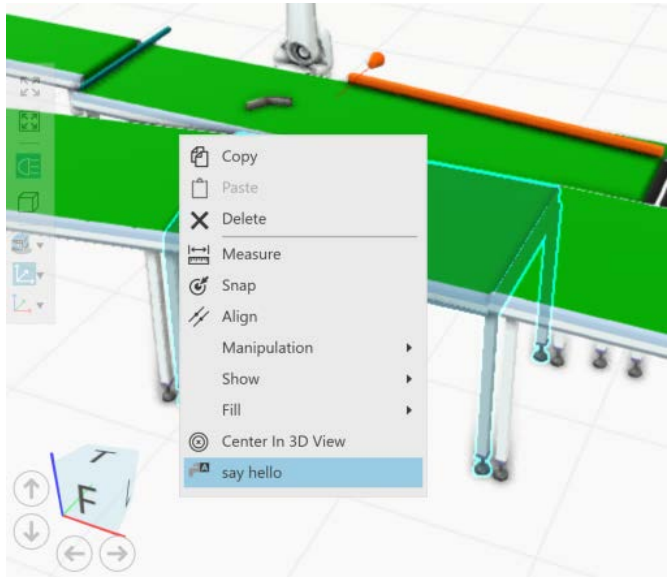
QUICK MENU

addMenuItem("VcConfigureQuickmenu", "say hello", -1, "cmdHello")



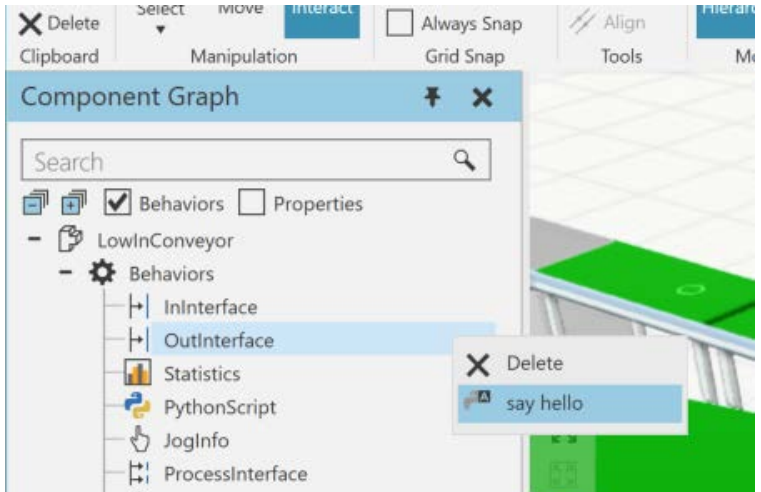
3D CONTEXT MENU

```
addMenuItem("VcConfigure3DContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcTeach3DContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcDrawing3DContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcAuthor3DContextMenu", "say hello", -1, "cmdHello")
```



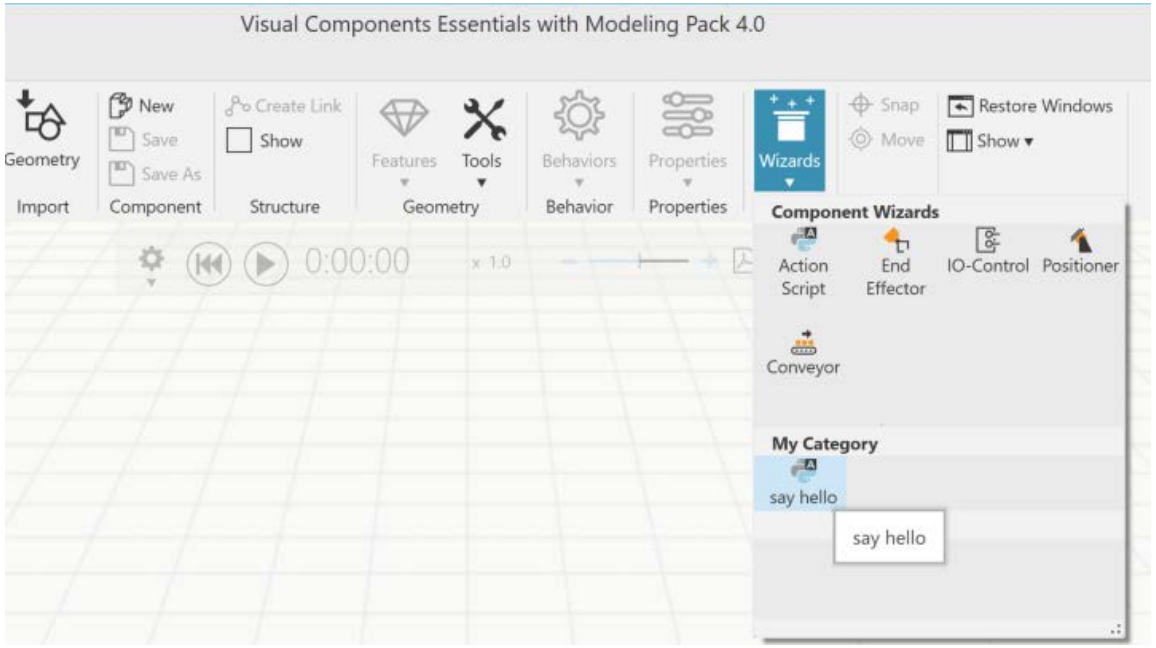
COMPONENT GRAPH TREE CONTEXT MENU

```
addMenuItem("VcAuthorTreeViewContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcAuthorTreeViewBehaviorsContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcAuthorTreeViewPropertiesContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcAuthorTreeViewRobotFramesContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcAuthorTreeViewFeaturesContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcAuthorTreeViewNodesContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcAuthorTreeViewToolContextMenu", "say hello", -1, "cmdHello")
addMenuItem("VcAuthorTreeViewBasesContextMenu", "say hello", -1, "cmdHello")
```



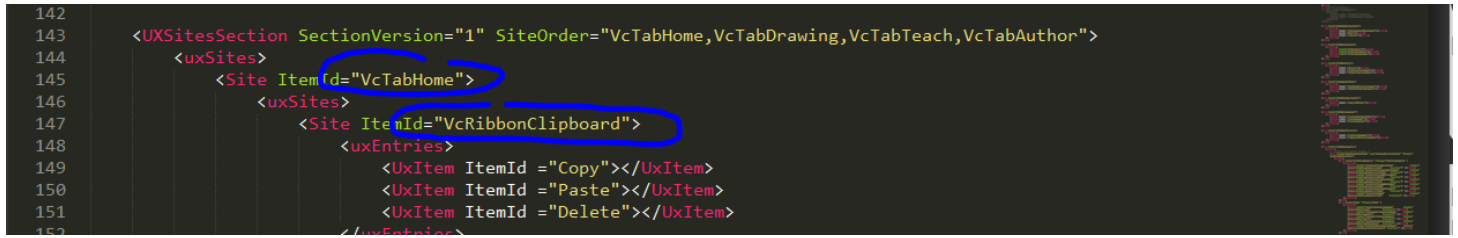
GALLERY

addMenuItem(VC_MENU_MODELING_WIZARDS + "/My Category", "say hello", -1, "cmdHello")



DOCUMENTATION

- Not yet fully documented ☹
- Use the config file as raw doc ☺
  - C:\Program Files\Visual Components\Visual Components Essentials with Modeling Pack 4.0\VisualComponents.Essentials.exe.config



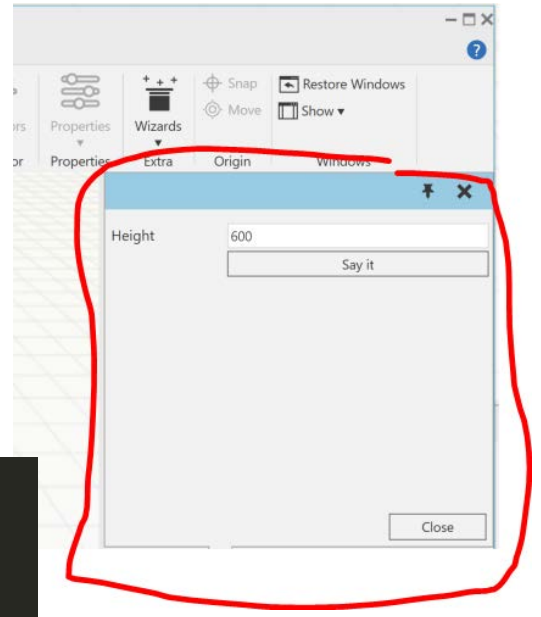
- Find examples in the python command installation folder ☺
  - C:\Program Files\Visual Components\Visual Components Essentials with Modeling Pack 4.0\Python\VisualComponents\FeatureTreeTools
  - C:\Program Files\Visual Components\Visual Components Essentials with Modeling Pack 4.0\Python\VisualComponents\Wizards

Action Panel

VISUAL COMPONENTS

USING ACTION PANEL WITH COMMANDS

- Create properties to the command
- executeInActionPanel()
- Properties flagged visible are shown in the action panel




```python
from vcCommand import *

createProperty(VC_REAL, 'Height')
createProperty(VC_BUTTON, 'Say it')

def OnStart():
  global button
  button = getProperty('Say it')
  button.OnChanged = say_it
  executeInActionPanel()

def say_it(arg):
  hProp = getProperty('Height')
  print 'Hello %.2f' % hProp.Value

addState( None )

```

Localizing commands

VISUAL COMPONENTS

LOCALIZE GUI

- Localize
  - Command names (”Text”)
  - Tooltips (”Tooltip”)
  - GUI Icons (”Icon”)

```
from vcApplication import *

icon1 = """M13.913,3.517h-2.414V0.585C11.499,0.264,11.241,0,10.916,0H6.119C5.973,0,5.831,0.054,5.724,0.154
    L1.857,3.695c-0.121,0.111-0.19,0.268-0.19,0.431v7.77c0,0.324,0.26,0.586,0.586,0.586h2.41v2.93c0,0.326,0.261,0.59,0.584,0.59
    h8.666c0.323,0,0.585-0.264,0.585-0.59V4.101C14.498,3.781,14.235,3.517,13.913,3.517z M5.675,1.789v2.18H3.299L5.675,1.789z
     M4.853,7.213c-0.122,0.108-0.19,0.27-0.19,0.434v3.666H2.838V5.14h3.424c0.325,0,0.587-0.264,0.587-0.588V1.171h3.481v2.346H9.115
    c-0.149,0-0.288,0.055-0.395,0.154L4.853,7.213z M8.675,5.304v2.18H6.293L8.675,5.304z M13.327,14.827H5.834V8.659h3.422
    c0.323,0,0.587-0.268,0.587-0.589V4.689h3.483V14.827z"""

icon2 = 'C:/Program Files/Visual Components/Visual Components Essentials with Modeling Pack 4.0/Icons/FeatureTreeTools/rExplode.svg'

netCommand = findCommand('netCommand')
#1
netCommand.execute('SetLocalizationCommand', 'English', 'Python', 'cmdLocalized1', 'Text', 'Localized Eng 1')
netCommand.execute('SetLocalizationCommand', 'English', 'Python', 'cmdLocalized1', 'Tooltip', 'Localized tooltip Eng 1')
netCommand.execute('SetLocalizationCommand', 'English', 'Python', 'cmdLocalized1', 'Icon', icon1)
#2
netCommand.execute('SetLocalizationCommand', 'English', 'Python', 'cmdLocalized2', 'Text', 'Localized Eng 2')
netCommand.execute('SetLocalizationCommand', 'English', 'Python', 'cmdLocalized2', 'Icon', icon2)
```

LOCALIZE PROPERTIES

- Localize
  - property names
  - tooltips
- Localize before creating the property

```
1  from vcCommand import *
2
3  localize = app.findCommand('netCommand')
4  # property names
5  localize.execute('SetLocalizationCommand', 'Klingon', 'Python', 'cmdHello.Height', 'Name', 'Heighten')
6  # property tooltip
7  localize.execute('SetLocalizationCommand', 'Klingon', 'Python', 'cmdHello.Height', 'Description', 'Adjustungen heighten sta 1000 een 2000')
8
```

Selection and Context

VISUAL COMPONENTS

SELECTION

- New Selection manager in python API
  - vcApplication.SelectionManager
  - Use this instead of
    - app.getSelection(...)
    - app.CurrentSelection

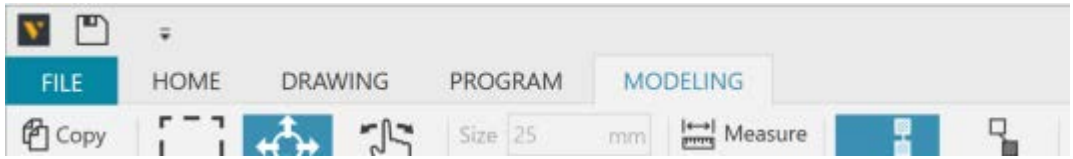
```
from vcScript import *
comp = getComponent()
app = getApplication()
sm = app.SelectionManager
sm.setSelection(comp)

def foo():
  print 'selection event'

sm.SelectionChanged = foo
```

CONTEXT

- Context controls
  - What can be selected in 3D
  - What panels are shown and where




```
VC_CONTEXT_CONFIGURE   = “Configure“
VC_CONTEXT_DRAWING     = "Drawing“
VC_CONTEXT_TEACH       = "Teach“
VC_CONTEXT_AUTHOR      = "Author"
```

CONTEXT

- Python API
  - Access all available contexts
  - Set and get active context
  - Event for context change
  - Access active objects in context

```
from vcScript import *
app = getApplication()

def contextEvent():
  print '-' * 99
  for c in app.Contexts:
    print c.Id, c.Id == app.CurrentContext.Id

app.ContextChanged = contextEvent
```

CONTEXT

- Set the active context

```
from vcScript import *
app = getApplication()
# set context
app.CurrentContext = app.Contexts[1]
'OR'
app.setContext('Drawing')
```

- Access active objects

```
from vcScript import *
app = getApplication()
comp2 = app.findComponent('NewComponent #2')
comp = getComponent()
context = app.CurrentContext
if context.Id == 'Author':
  print context.ActiveComponent
  print context.ActiveNode
  print context.ActiveFeatures
```

Snap

VISUAL COMPONENTS

SNAP COMMAND

- Interface same as in 2014 but with small additions
  - cmd. SnapOnBoundType
  - cmd.clear()
- Improved control for highlighting

SNAP COMMAND

- SnapOnBoundType

```
from vcScript import *

app = getApplication()
snap = app.findCommand('interactiveSnap')

def snapped(cmd):
  print cmd.TargetNode, cmd.TargetPosition

snap.OnTargetSet = snapped
snap.SnapOnBoundType = -1   # <===
snap.execute()
```

Cherry on the cake

VISUAL COMPONENTS

RECOMPILING PYTHON COMMANDS

- No need to recompile commands manually
- Save the .py file and execute the command again
- The command is automatically recompiled