

Simulation Modeling Sciences

CUBIT Fast-Start Tutorial 16. Cubit Scripting with Python



Sandia National Laboratories is a multi-mission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

Python

Simulation Modeling Sciences

- Python is a well established, widely accepted scripting language. Its use within the engineering community continues to grow.
 - Abaqus
 - Paraview
 - PyTrilinos

Some useful links to learn Python

- Official website: www.python.org
- Getting started: www.python.org/about/gettingstarted/
- Tutorial (2.7): docs.python.org/2/tutorial/index.html
- Reference (2.7): docs.python.org/2/reference/index.html



Enabling the Script Tab from Cubit

- Select Tools Options
 Layout
- Select "Show Script Tab"
- The script tab will allow direct entry of python commands.

%>for i in range(10):

Script /\ Command /

cubit.cmd(" Enter cubit command

%> print i

5 Options		
-	@ 💿	Options 📀 😒 😒
ow Script tab will t entry of nmands.	Command Panels Display General Geometry Defaults History Label Defaults Layout Mesh Defaults Mouse Post Meshing • Quality Defaults	Options ? `` `` `` `` `` `` Command Line and Journal Font Font Family DejaVu Sans Font Size 10 Sample Text Toolbars and Docking Windows Aprepro Editor Command Line ? Command Line ? Command Panel ? Display Tools ? File Tools ? Model Tree ? Power Tools ? Select CFD BC Tools ? Select EFA BC Tools ? Select EFA BC Tools ? Select EFA BC Tools ? Select Mesh Tools ? Select Mesh Tools ? Select Mesh Tools ? Show Script Tab ? Show Script Tab
		Window Tab Position • Bottom • Top • Left • Right Save Close
t command		")
Error / \ History /		





Python Journal Editor

Tools Help	Simulation Modeling Sciences
Journal Editor	File Edit View Display Tools Help
Play Journal File	
Custom Toolbar Editor	
Components	Access from toolbar
Options	
n Iournal Editor 🔗 🔿 🔊	aurnal Editor 🔗 🔿 🕅
File Edit Tools	File Edit Tools
🗋 🗁 🔚 🍫 🏞 🗋 🖺 🕨 👶 🍿	🗋 🗁 🔛 🕫 🛷 🖨 🖺 ⊳ 👶 🍿
💷 Untitled 🔇	🥏 Untitled 🔞
reset brick x 10	kubit.cmd('reset') cubit.cmd('brick x 10')
mesh volume 1	cubit.cmd('mesh volume 1')
Journal Editor with Cubit commands	Journal Editor with Python commands



Custom Toolbars

Simulation Modeling Sciences



 Execute a series of Python commands at the click of a button

R 🗧

Example custom toolbar



CUBIT User Tutorial

 Execute a Python script

Custom Toolbars

🍿 🕑 🦳 Custom Toolbar Editor 🛛 🕑 📀 🕚	\otimes
Toolbars Buttons	
All Visible All Visible + -	
✓ MyTools ✓ MyPythonScript	C w ju
Edit Tool Button	
Name PythonExample	
Icon	
Working Dir (Optional)	
Commands	
#!python	
cubit.cmd("reset") cubit.cmd("bri x 10") cubit.cmd("mesh vol 1") print("********** I meshed a brick! **************)	
Show Description	
Help Reset OK Apply Cancel	

Simulation Modeling Sciences

Create a custom tool button and write Python commands in place, just like Cubit commands.

Include "#!python" to tell Cubit to interpret the commands as Python (necessary for this tool only)



Custom Toolbars

	Simulation Modeling Sciences
🍈 💿 Custom Toolbar Editor 🤈 📀 🚫 🖉	
Toolbars Buttons	
✓ All Visible ✓ MyTools ✓ MyTools ✓ MyPythonScript	Create a Python script button and choose a Python script to run.
Edit Python Script	Select the Python script to
Name MyPythonScript	run.
Icon	
Script /home/michael/CurrentTask/SomeCoolScript.py	
Working Dir /home/michael/CurrentTask Description	(Optional) choose a directory
(Optional)	from which to run the script.
Help Reset OK Apply Cancel	E C





Cubit Interface

Simulation Modeling Sciences

Primarily, a query interface into Cubit

- double mesh_size =
 cubit.get_mesh_size("volume", 22);
- Accessible via C++ or python
- Change state by using cubit.cmd(" ... ")
 - import cubit
 - -cubit.cmd("create brick x 10 y 10 z 10")
 - -cubit.cmd("mesh volume 1,3,5")







- Play the script
- Change the parameters
- Play again



Example 2

Simulation Modeling Sciences

In the Cubit Journal File (Python) Editor

 Create a python script to compute and print the minimum shape metric for all volumes. Consider using the following CubitInterface functions

get_entities()
get_volume_hexes()
get_quality_value()





Simulation Modeling Sciences

```
all_vols = cubit.get_entities("volume")
min_quality = 1.0
for vol in all_vols:
    vhexes = cubit.get_volume_hexes(vol)
    for hex in vhexes:
        q = cubit.get_quality_value("hex", hex, "shape")
        if q < min_quality:
            min_quality = q</pre>
```

print 'min quality = ', min_quality



Cubit Extended Interface

Simulation Modeling Sciences

Create "pythonic" objects in Cubit
Reduce (but not eliminate) id issues

```
bri = cubit.brick(10,5,3)
cyl = cubit.cylinder(12,2,2,2)
vols = cubit.subtract([cyl], [bri])
v = vols[0].volumes()
v[0].mesh()
print dir(v[0])
print v[0].id()
v[0].mesh()
```



Python Help

Simulation Modeling Sciences

Documentation

– Help Manual online or built-in Appendix/Python

Python prompt

- print dir(object)



Black Box Cubit

Simulation Modeling Sciences

- Cubit can also be run from inside python
 - Set your environment variable PATH to include the installed Cubit libraries
 - You may also need to set PYTHONPATH to the same place
 - **Run Python**
 - import cubit
 - cubit.init([""])
 - cubit.cmd("brick x 10")
- This allows you to run Cubit programatically and interact with other tools.



Example 3

Simulation Modeling Sciences

In the native operating system using python 2.7

- Copy your script from Example 2 to a text editor
- Add the ability to import a mesh
- Make the script you created above run on the hexes in the mesh and print the result





Simulation Modeling Sciences

```
#!python
import sys
# add Cubit libraries to your path
sys.path.append('/Applications/Cubit-15.4/Cubit.app/Contents/MacOS')
import cubit
cubit.init(['cubit', '-nojournal'])
cubit.cmd('import mesh geom "mesh.g"')
all vols = cubit.get entities("volume")
min quality = 1.0
for vol in all vols:
    vhexes = cubit.get volume hexes(vol)
    for hex in vhexes:
        q = cubit.get quality value("hex", hex, "shape")
        if q < min quality:
            min quality = q
```

print 'min quality = ', min_quality





Customization

Simulation Modeling Sciences

Cubit can support some additions to the GUI

- Add new menu items
- Add new dialogs
- Cannot currently add new control panels

Use PyQt5 - a python interface to Qt

from PyQt5 import QtGui

QtGui.QMessageBox.question(None, "Title", "Hello")

