

A large snake with a blue eye is coiled on a black bag. The snake has a patterned body with dark and light brown spots. The background is green foliage.

Commercial For GaudiPython

- One possibility to run applications:
gaudirun.py DaVinci.py
 - ◆ gaudirun controls event loop
- Another possibility: python myDaVinci.py
 - ◆ User in control of event loop
 - ◆ Requires a few additional lines to DaVinci.py
 - ▶ `from GaudiPython import AppMgr`
 - ▶ `appMgr = AppMgr()`
 - ▶ `for n in range(maxEvents) :`
`appMgr.run(1)`

Simple Example

myDaVinci.py :

```
from runDaVinci import *  
importOptions("$DAVINCIROOT/options/MC09-Bs2JpsiPhiDst.py")
```

```
from GaudiPython import AppMgr
```

```
appMgr = AppMgr()
```

```
maxEvents = 10
```

```
for n in range(maxEvents) : appMgr.run(1)
```

(runDaVinci.py can be copied from
\$LHCBRELEASES/PANORAMIX/PANORAMIX_v17r5/Vis/Ga
udiPythonTutorial/pythom/)

Why ?

- ◆ `python -i myDaVinci.py`
- Allows interactive inspection of configuration, event loop sequence, ...
- Allows interactive inspection of TES
- Allows interactive development of code

- `print appMgr.algorithms()`
`['DaVinciInitSeq', 'DaVinciMainSequence',
 'MonitoringSequence', 'DaVinciInit',...]`
- `prop = appMgr.algorithm('DaVinciInitSeq').properties()`
- `for p in prop: print p, ' : ', prop[p].value()`
`... Members : ['DaVinciInit/DaVinciInit',
 'GaudiSequencer/PhysInitSeq',
 'GaudiSequencer/AnalysisInitSeq'] ...`
- `prop = DaVinci().properties()`
- `for p in prop: print p, ' : ', prop[p]`
`... UserAlgorithms : [<GaudiSequencer/Sel09Bu2LLKFilter
 at 0xae950>, <BTagging/Btagging>]
 ETCFile : <no value>
 MoniSequence : [<GaudiSequencer/ExampleSeq>]
 ...`

```

■ muidalg = appMgr.algorithm('MuonIDAlg')
prop = muidalg.properties()
for p in prop: print p, ' : ', prop[p].value()

...
◆ XFOIParameter3   : [0.20000000000000000001, 0.08000000000000000002,
...]
◆ XFOIParameter2   : [11, 3, 1, 1, 31, 28, 21, 17, 30, 31, 27, 22,
28, 33, 35, 47, 31, 39, 56, 151]
◆ MupBinsR4        : [6000, 8000, 10000, 15000]
◆ ...

```

- `evt = appMgr.evtsvc()`
`evt.dump()`
`evt['Rec/Track/Best'].size()`
- **Force loading of all objects from disk to TES:**
`evt.dumpAll()`, **requires** `import gaudigadgets`
- `atrack = evt['Rec/Track/Best'][0]`
`print atrack`
`dir(atrack)`
- `for l in atrack.lhcbIDs():`
`print l.channelID(), l.isVelo()`

- `det = appMgr.detsvc()`
`det.dump()` works, but prints a lot !
- `velo =`
`det['/dd/Structure/LHcb/BeforeMagnetRegion/Velo']`

```
rSensor_10 = velo.rSensor(10)
>>> rSensor_10.rMax(0)
42.0
>>> rSensor_10.rMin(0)
8.169999999999999
>>> rSensor_10.rOfStrip(145)
14.82897361256984
```

```
>>> ac =
    velo.geometry().alignmentCondition()
>>> print ac.printParams()
    dPosXYZ = [0, 0, 0]
    dRotXYZ = [-0.0001, 0, -0.0001]
    pivotXYZ = [0, 0, 0]
```


And last no least

- **Integration with ROOT:** `Import ROOT`
(in order that POOL doesn't decide to switch to batch mode, do:
`from ROOT import TH1F` as first line)
- `hist = appMgr.histSvc()`
`hist.dump()`
- `hist['DaVinciInit.DaVinciMemory/Virtual mem, all entries'].Draw()`
- `myHist = ROOT.TH1F('1', 'test 1', 100, 0, 100)`
`rc = myHist.Fill(42)`
`myHist.Draw()`
- **Of course with Python, the “whole world” can be used together with LHCb software**

■ DST as Big Ntuple

- ◆ Many examples covering non-standard use cases in Vis/GaudiPythonTutorial package

■ GaudiPython Twiki

■ GaudiPython for Analysis (Ulrich Kerzel)

■ More advanced usage, Bender Tutorial (Vanya Belyaev)



The example script

```
from ROOT import TH1F
from runDaVinci import *
importOptions("$DAVINCIROOT/options/MC09-Bs2JpsiPhiDst.py")
from GaudiPython import AppMgr
appMgr = AppMgr()
maxEvents = 10
for n in range(maxEvents) : appMgr.run(1)
#
print appMgr.algorithms()
muidalg = appMgr.algorithm('MuonIDAlg')
prop = muidalg.properties()
for p in prop: print p, ' : ', prop[p].value()
#
evt = appMgr.evtSvc()
evt.dump()
evt['Rec/Track/Best'].size()
atrack = evt['Rec/Track/Best'][0]
dir(atrack)
#
hist = appMgr.histSvc()
hist.dump()
hist['DaVinciInit.DaVinciMemory/Virtual mem, all entries'].Draw()
#
det = appMgr.detsvc()\velo = det['/dd/Structure/LHCb/BeforeMagnetRegion/Velo']
rSensor_10 = velo.rSensor(10)
print rSensor_10.rMax(0), rSensor_10.rMin(0), print rSensor_10.rOfStrip(145)
ac = velo.geometry().alignmentCondition()
print ac.printParams()
```