Zero to EPICS 7 in 5 minutes

Thomas Fors

Argonne National Lab
AES Division
Controls Group

Using Python

This is all you need to get started...

```
wget https://repo.continuum.io/miniconda/Miniconda3-latest-Linux-x86_64.sh
bash ./Miniconda3-latest-Linux-x86_64.sh -b -p $HOME/demo
export PATH=$HOME/demo/bin:$PATH
conda install -y epics::pvapy
```

Argonne
NATIONAL LABORATORY

Step 1

```
$ wget https://repo.continuum.io/miniconda/Miniconda3-latest-Linux-x86_64.sh
$ bash ./Miniconda3-latest-Linux-x86_64.sh -b -p $HOME/demo
```

-b

>   Run install in batch mode (without manual intervention)

-p PREFIX

>   Install prefix, defaults to $HOME/miniconda3, must not contain spaces.

-h

>   Print help message and exit.

Anaconda   https://www.anaconda.com

- Open-source Python distribution for data science
- 6M+ users
- 250+ packages installed, 1400+ available

Conda   https://conda.io/docs

- Open-source package manager written in Python
- Package and distribute software written in any language
- Manages multiple environments with conflicting dependencies

Miniconda   https://conda.io/miniconda.html

- Python + conda

Step 2 — Add Miniconda to path

```
$ export PATH=$HOME/demo/bin:$PATH
```

Step 3 — Install pvaPy and dependencies

```
$ conda install -y epics::pvapy
```

> *-y, --yes*
>> Do not ask for confirmation.
>
> *epics::pvapy*
>> Install pvapy package from the epics channel.

Installs the following packages from the *epics* channel of anaconda.org:

- pvapy (ver 1.1.0)
- epics-base (ver 7.0.1.1)
- pvapy-boost (ver 1.66.0)

pvAccess Server

```python
from math import pi, sin, asin
from time import sleep
import pvaccess as pva

dataStruct = {'ArrayId': pva.UINT, 'Time': [pva.DOUBLE],
              'Sinusoid': [pva.FLOAT], 'Triangle': [pva.FLOAT]}
srv = pva.PvaServer('some_pv', pva.PvObject(dataStruct))

t0 = 0.0
n = 0
dt = 1./1000
while True:
    sleep(0.1)
    time = [t0+dt*i for i in range(0, 100)]
    sinusoid = [sin(2*pi*1.1*t + pi/2) for t in time]
    triangle = [(2/pi)*asin(sin(2*pi*1.1*t)) for t in time]
    pv = pva.PvObject(dataStruct, {'ArrayId': n, 'Time': time,
                                   'Sinusoid': sinusoid, 'Triangle': triangle})
    srv.update(pv)
    t0 = time[-1] + dt
    n += 1
```

Argonne
NATIONAL LABORATORY

pvAccess Server

```python
from math import pi, sin, asin
from time import sleep
import pvaccess as pva

dataStruct =  {'ArrayId': pva.UINT, 'Time': [pva.DOUBLE],
               'Sinusoid': [pva.FLOAT], 'Triangle': [pva.FLOAT]}
srv = pva.PvaServer('some_pv', pva.PvObject(dataStruct))

t0 = 0.0
n = 0
dt = 1./1000
while True:
    sleep(0.1)
    time = [t0+dt*i for i in range(0, 100)]
    sinusoid = [sin(2*pi*1.1*t + pi/2) for t in time]
    triangle = [(2/pi)*asin(sin(2*pi*1.1*t)) for t in time]
    pv = pva.PvObject(dataStruct, {'ArrayId': n, 'Time': time,
                                    'Sinusoid': sinusoid, 'Triangle': triangle})
    srv.update(pv)
    t0 = time[-1] + dt
    n += 1
```

Argonne
NATIONAL LABORATORY

pvget

```
$ pvget -m -r "field()" some_pv
```

-*m*

Monitor mode

-*r*

Request, specifies what fields to return and options, default is 'field(value)'
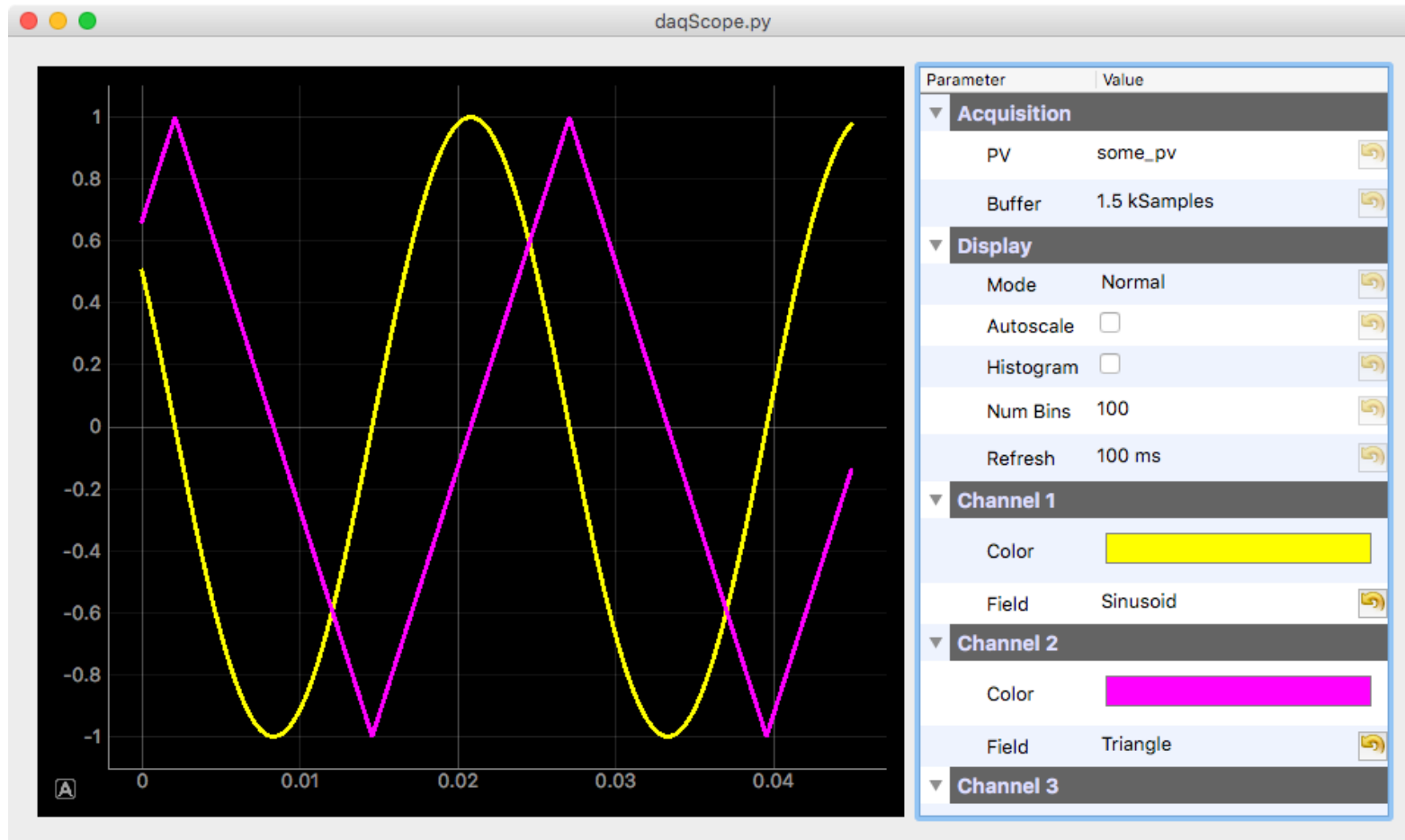
pvapy

```python
import pvaccess as pva

def monitor_func(data):
    print(data['Sinusoid'])

chan = pva.Channel('some_pv')
chan.monitor(monitor_func)
```

daqScope.py

```
$ conda install -y pyqtgraph
$ ./daqScope.py --pv some_pv
```

DEMO

APS-U DAQ / Video

Tomorrow 3 PM

APS Gallery (Below the Auditorium)

More Information

pvaPy

Topic	Resource
pvaPy documentation	https://epics.anl.gov/extensions/pvaPy/production/
pvaPy examples	https://github.com/epics-base/pvaPy/tree/master/examples/

conda

Topic	Resource
Download Miniconda	https://conda.io/miniconda.html
Conda package manager documentation	https://conda.io/docs/
Full Anaconda Distribution	https://www.anaconda.com/what-is-anaconda/
Building your own conda packages	https://conda.io/docs/user-guide/tutorials/index.html

Argonne
NATIONAL LABORATORY