

Xbrain Docker:

The Xbrain docker download will create a workspace within a docker container which includes the Xbrain code and the version of intern needed to access your data from DVID

1. Clone the Xbrain repository found here:
`https://github.com/nerdslab/xbrain/tree/master/Demo`
2. Cd in your terminal into the file directory where you cloned the Xbrain repository
 - a. Enter: `docker build -t xbrain:v1 .`
 - b. Enter: `docker run -it --rm -p 8888:8888 -v /path/to/xbrain/clonedRepo/:/home/jovyan/work/ xbrain:v1`

Clone the intern Repository and pip install intern:

Before running your code you should pip install intern. While this should be sufficient in the near future, for the time being you have to clone the repo and pip install the RemoteExtension Branch.

1. Clone the intern repository found here:
<https://github.com/jhuapl-boss/intern.git>
2. Switch to the extension branch in the repository:
`Git checkout RemoteExtension`
3. Install the branch version of intern
`Pip install -U .`
Or
`pip3 install -U .`

Create a Local Dvid Repository Through intern:

The following code will allow you to create a local DVID repository and upload and download data using the intern module.

1. Go to Localhost:8888
2. Open intern → examples → dvid
3. Each file you need to make a new dvid localRepo, upload data onto the repo and download data can be found in this directory.
4. You can copy and paste the code from the examples categorized as (1) and (2) into a Python text editor like Atom or vim running Python3.
5. Before you run your code, remember to change the host name to your current port running DVID. You can do that as follows:
 - a. Open your terminal
 - b. Enter: `ifconfig en0 | grep 'inet ' | awk -v N=2 '{print $N}'`
 - c. Paste the Resulting IP on your host inside the dvid remote declaration:
 - i. `dvid = DVIDRemote({`
 - ii. `"protocol": "http",`
 - iii. `"host": "IP:8000",`
 - iv. `})`
6. You also need to change the path to your data on file (1)

- a. path = "/path/to/where/your/data/is"
7. Now you should be able to run each file in the order they are labeled in
 8. Finally, Add to your Xbrain code a cell in which you enter the code found in intern

```
example(3)
# load data
chan="c7e/MaskedImg14"
dvid = DVIDRemote({ "protocol": "http",
                    "host": "10.105.3.111:8000/",
                    })
input_data = dvid.get_cutout(
    dvid.get_channel(chan), 0,
    [1270,1470], [1445,1645], [290,490]
)
```