
BC Technical Working Group Documentation

Release 0.1.1

Advanced Photon Source

Aug 17, 2022

CONTENTS

1	Content	3
	Bibliography	17



This [GitHub repository](#) provides a template to add [sphinx / Read The Docs](#) documentation to generate this [Web Doc](#) pages.

These pages are written using [reStructuredText](#) that allows *emphasis*, **strong**, `literal` and many more styles.

You can add a reference [\[A1\]](#), include equations like:

$$V(x) = \left(\frac{1 - \eta}{\sigma \sqrt{2\pi}} \right) \cdot \exp \left(\frac{x^2}{2\sigma^2} \right) + \eta \cdot \frac{\sigma}{2\pi} \cdot \frac{1}{x^2 + \left(\frac{\sigma}{2} \right)^2}$$

or

$$I_{white} = \int_{E_1}^{E_2} I(\theta, E) \cdot F(E) dE.$$

and tables:

Member	Type	Example
first	ordinal	1st
second	ordinal	2nd
third	ordinal	3rd

More examples:

$$X(e^{j\omega}) = x(n)e^{-j\omega n}$$

Warning: Warning text.

Note: Note text.

CONTENT

1.1 About

This web site contains documentation associated with the activity of the Beamline Control Technical Working Group

1.2 How to ...

1.2.1 NIC for FLIR 10Ge

Step-by-step to configure a FLIR 10Gb camera with an Intel Network adapter:

Prerequisites:

1. 64GB memory
2. Cat 6A cable
3. Intel X550T2 ETHERNET CONVERGED Network Adapter X550-T2 (available from Sorcium as Part#: 3E9073)

Warning: The same procedure works also for the latest FLIR supported Myricom ARC Series C-Class network adapter

Tasks:

1. Enable jumbo packet
2. Disable DHCP and set a fixed IP address on the Ethernet port connecting to the FLIR
3. Increase the receive buffer size.
4. Increase the Network parameters in the kernel
5. Set the NIC tx queue length

Description:

1. 2. and 3. are documented at [here](#)
4. is documented both at [flir doc](#) and in the [areadetector doc](#)
4. edit /etc/sysctl.conf and add:

```
net.core.rmem_default=26214400
net.core.rmem_max=268435456
```

5. edit /etc/rc.local and add:

```
/usr/sbin/ifconfig ens1f1 txqueuelen 3000
```

Warning: NIC camera settings and 10GB nic settings in this example the camera is attached to ens1f1

Set the NIC as follows:

```
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9000
    inet 169.254.100.68 netmask 255.255.255.0 broadcast 169.254.100.255
    inet6 fe80::260:ddff:fe42:4e2 prefixlen 64 scopeid 0x20<link>
    ether 00:60:dd:42:04:e2 txqueuelen 1000 (Ethernet)
    RX packets 248 bytes 86800 (84.7 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 282 bytes 26319 (25.7 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth1: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9000
    inet 169.254.0.65 netmask 255.255.255.0 broadcast 169.254.0.255
    inet6 fe80::260:ddff:fe42:4e3 prefixlen 64 scopeid 0x20<link>
    ether 00:60:dd:42:04:e3 txqueuelen 1000 (Ethernet)
    RX packets 1273 bytes 445550 (435.1 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 281 bytes 26249 (25.6 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

After the above tasks are completed we need to configure the camera with a static IP address in the same NIC subnet. To accomplish this task install and run [SpinView](#) from FLIR and connect the camera to the same subnet the computer running spinview is connected to, then:

SpinView 2.7.0.128

File View Help

Devices

Serial IP Address

System	Serial	IP Address
Intel(R) Ethernet Connection (10) I219-V		164.54.113.107
Blackfly BFLY-PGE-13H2M	20209422	164.54.113.3
Blackfly BFLY-PGE-23S6C	16434638	164.54.113.78
Blackfly BFLY-PGE-13H2M	20136488	164.54.113.2
Oryx ORX-10G-51S5M		164.54.113.4

Oryx ORX-10G-51S5M 19173710

Acquisition Mode: Continuous

Log Viewer

* Priorit	Timestamp	Logger	Component	Message	Data
NOTICE	17-08-2022 09:43:23.200	SpinnakerNETGUI	SpinView.SpinView_WPF	[SN:19173710] Device Stream Buffer Handling Mode set to "Newest Only"	None
NOTICE	17-08-2022 09:43:23.197	SpinnakerNETGUI	SpinView.SpinView_WPF	[SN:19173710] Device Initialized	None
NOTICE	17-08-2022 09:43:19.683	SpinnakerNETGUI	SpinnakerNET.GUI.WPFControls.CameraSelectionControl	Discovered 7 interfaces and 4 compatible devices.	None
NOTICE	17-08-2022 09:43:19.671	SpinnakerNETGUI	SpinnakerNET.GUI.WPFControls.CameraSelectionControl	"Oryx ORX-10G-51S5M [SN:19173710]" has been detected on "Intel(R) Ethernet Connection (10) I219-V".	None
NOTICE	17-08-2022 09:43:19.666	SpinnakerNETGUI	SpinnakerNET.GUI.WPFControls.CameraSelectionControl	"Blackfly BFLY-PGE-13H2M [SN:20136488]" has been detected on "Intel(R) Ethernet Connection (10) I219-V".	None
NOTICE	17-08-2022 09:43:19.665	SpinnakerNETGUI	SpinnakerNET.GUI.WPFControls.CameraSelectionControl	"Blackfly BFLY-PGE-23S6C [SN:16434638]" has been detected on "Intel(R) Ethernet Connection (10) I219-V".	None
NOTICE	17-08-2022 09:43:19.664	SpinnakerNETGUI	SpinnakerNET.GUI.WPFControls.CameraSelectionControl	"Blackfly BFLY-PGE-13H2M [SN:20209422]" has been detected on "Intel(R) Ethernet Connection (10) I219-V".	None
NOTICE	17-08-2022 09:43:17.684	SpinnakerNETGUI	SpinnakerNET.GUI.WPFControls.CameraSelectionControl	Discovered 7 interfaces and 4 compatible devices.	None
NOTICE	17-08-2022 09:43:17.671	SpinnakerNETGUI	SpinnakerNET.GUI.WPFControls.CameraSelectionControl	"Oryx ORX-10G-51S5M [SN:19173710]" has been detected on "Intel(R) Ethernet Connection (10) I219-V".	None
NOTICE	17-08-2022 09:43:17.670	SpinnakerNETGUI	SpinnakerNET.GUI.WPFControls.CameraSelectionControl	"Blackfly BFLY-PGE-13H2M [SN:20136488]" has been detected on "Intel(R) Ethernet Connection (10) I219-V".	None
NOTICE	17-08-2022 09:43:17.669	SpinnakerNETGUI	SpinnakerNET.GUI.WPFControls.CameraSelectionControl	"Blackfly BFLY-PGE-23S6C [SN:16434638]" has been detected on "Intel(R) Ethernet Connection (10) I219-V".	None

Idle Active Loggers: 1 Total Messages: 87 DEBUG: 0 INFO: 0 WARN: 4 ERROR: 1 FATAL: 0 UNKNOWN: 0 Last Updated On: 8/17/2022 9:43:27 AM

Device IP Configuration - Oryx ORX-10G-51S5M

Persistent IP Configuration

IP Address: 169 . 254 . 0 . 51

Subnet Mask: 255 . 255 . 255 . 0

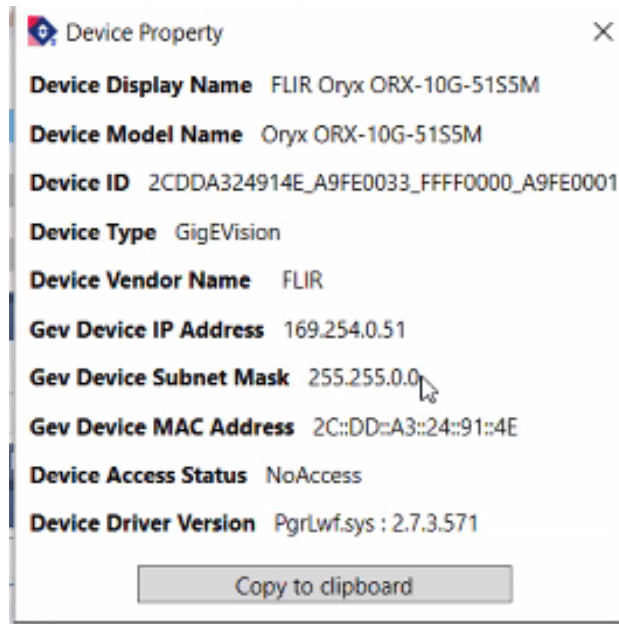
Default Gateway: 169 . 254 . 0 . 1

Set Current and Persistent IP Configuration

Set Persistent IP Configuration

IP Address Assignment Configuration

☒ LLA ☐ DHCP ☒ Persistent IP



Warning: After the above configuration is completed power cycle the camera to implement the IP address changes.

Reconnect the camera to the NIC on the AreaDetector controller computer and ping to its IP address to confirm the connectivity. The configuration is now complete.

1.2.2 Update this doc

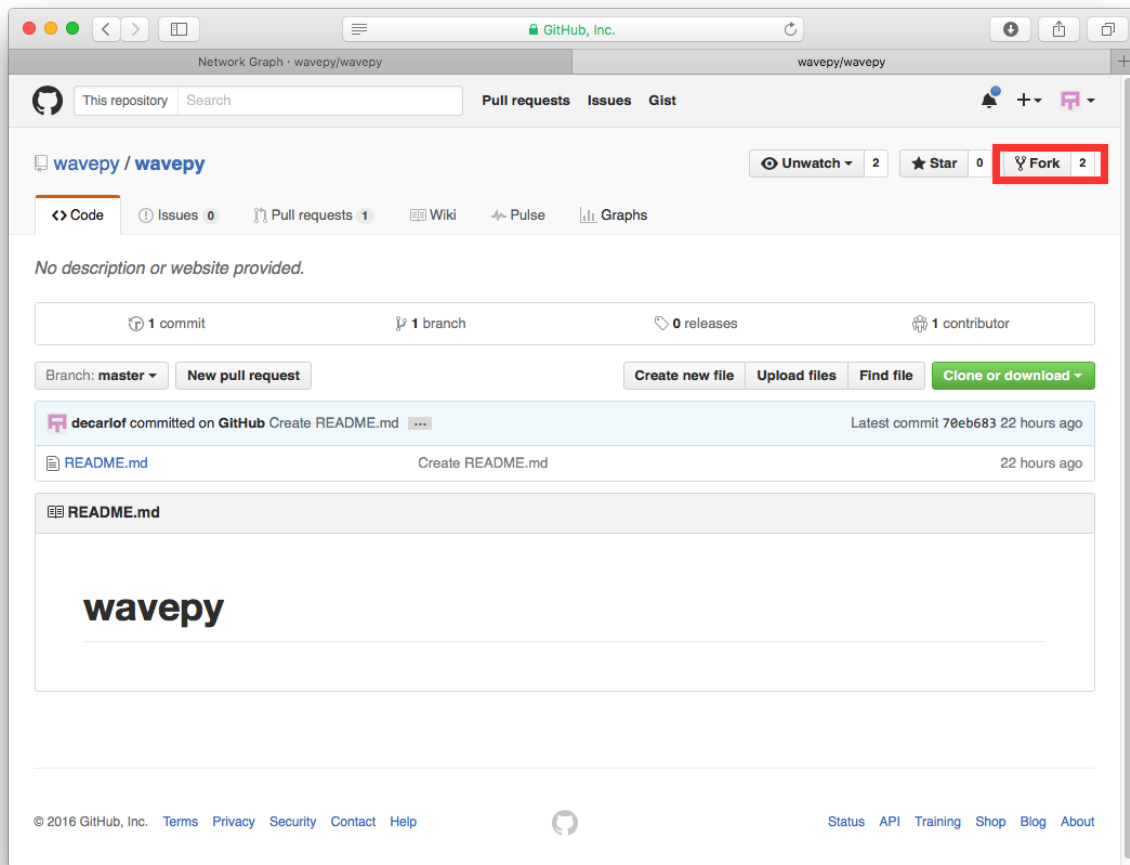
The documentation of this web site is maintained on [github](#) and automatically updated on [readthedocs](#) every time there is a new commit.

To contribute you need to **fork the repository**, **make the changes** in your fork and issue a **pull request**. Below are instructions to accomplish these steps.

Fork the repository

This section explains the basics for developers who wish to contribute to a project maintained on GitHub and using the [fork / pull request](#) mechanism for accepting developer contributions.

The project is maintained on GitHub, which is a version control and a collaboration platform for software developers. To start first register on [GitHub](#) and fork the [Project repository](#) by clicking the **Fork** button in the header of the [Project repository](#):

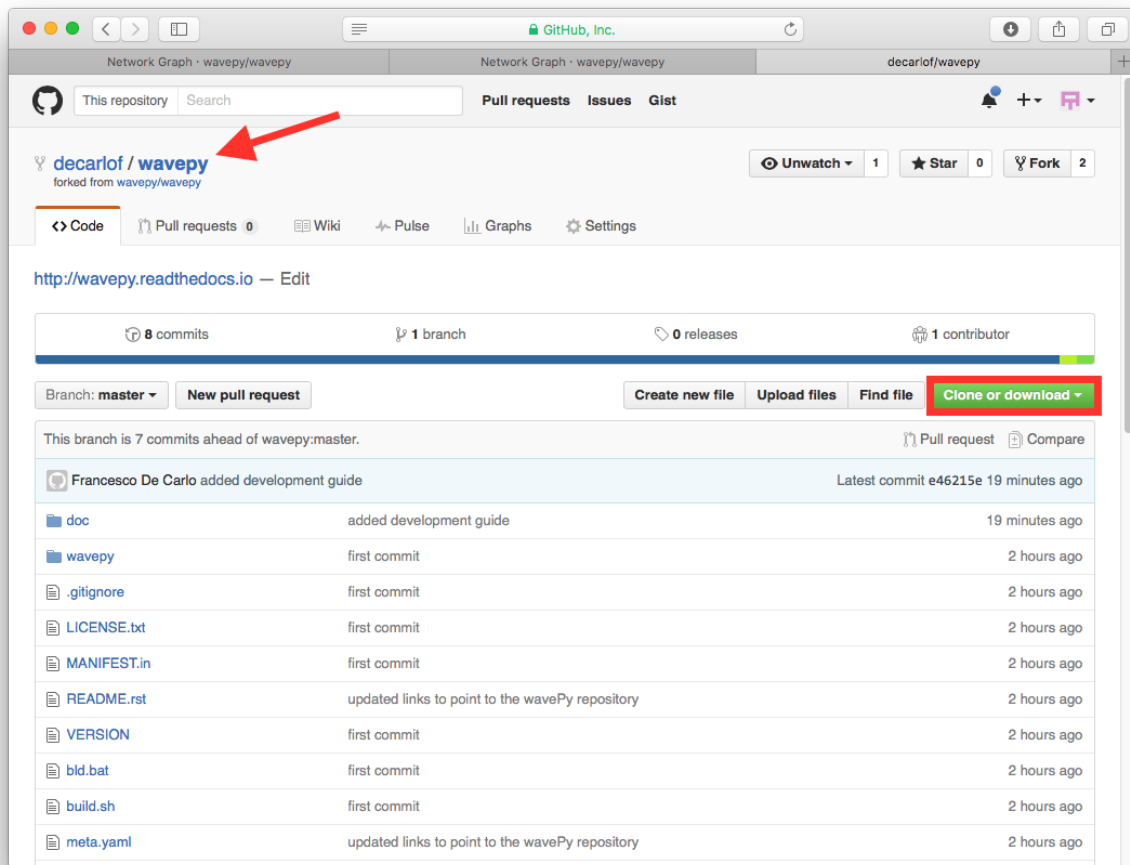


This successfully creates a copy of the project in your personal GitHub space.

Clone the repository

The next thing you want to do is to clone the repository you just created in your personal GitHub space to your local machine.

You can do this by clicking the **Clone in Desktop** button in the bottom of the right hand side bar:



This will launch the GitHub desktop application (available for both [Mac](#) and [Win](#)) and ask you where you want to save it. Select a location in your computer and feel comfortable with making modifications in the code.

Make the changes

Once you have a local copy of the documentation repository you can start making changes. To verify the new resulting documentation you can build the web pages locally and inspect them with the browser.

Build web pages locally

In your python anaconda environment create a new conda environment called docs:

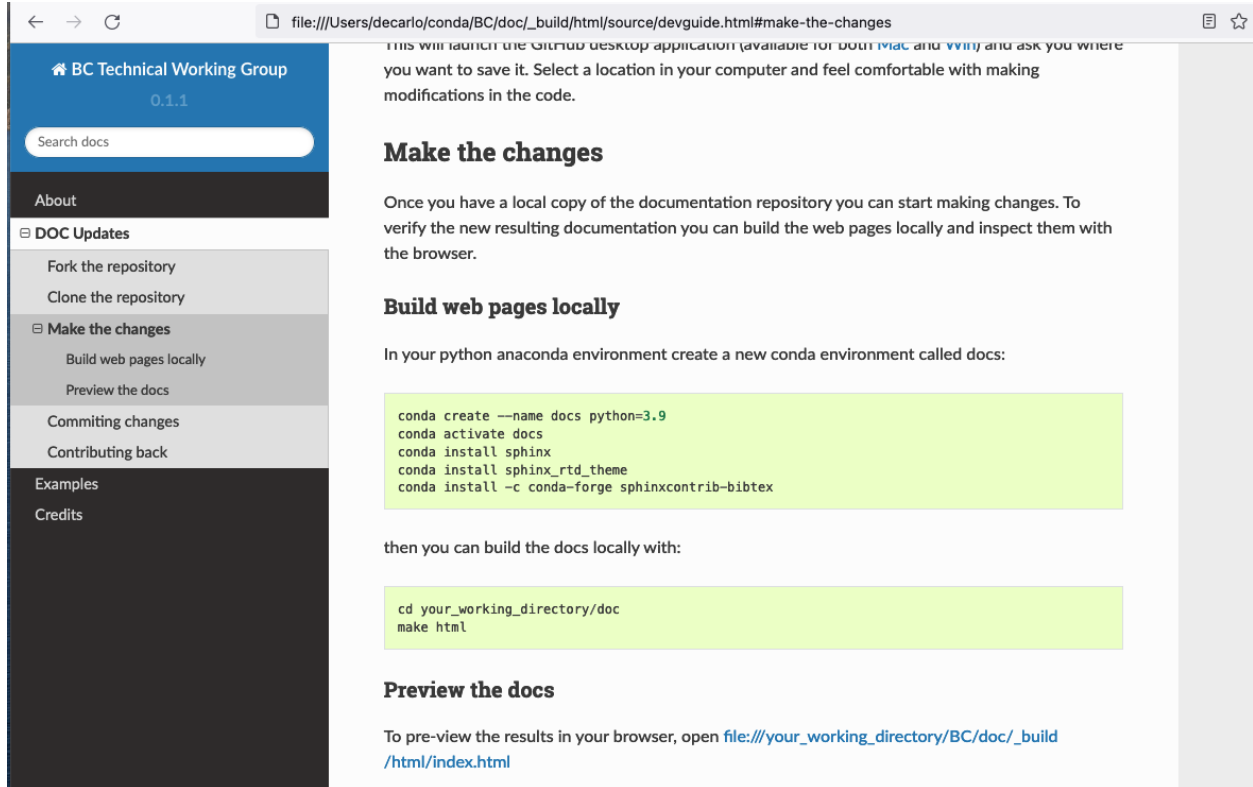
```
conda create --name docs python=3.9
conda activate docs
conda install sphinx
conda install sphinx_rtd_theme
conda install -c conda-forge sphinxcontrib-bibtex
```

then you can build the docs locally with:

```
cd your_working_directory/doc
make html
```

Preview the docs

To pre-view the results in your browser, open `file:///your_working_directory/BC/doc/_build/html/index.html`

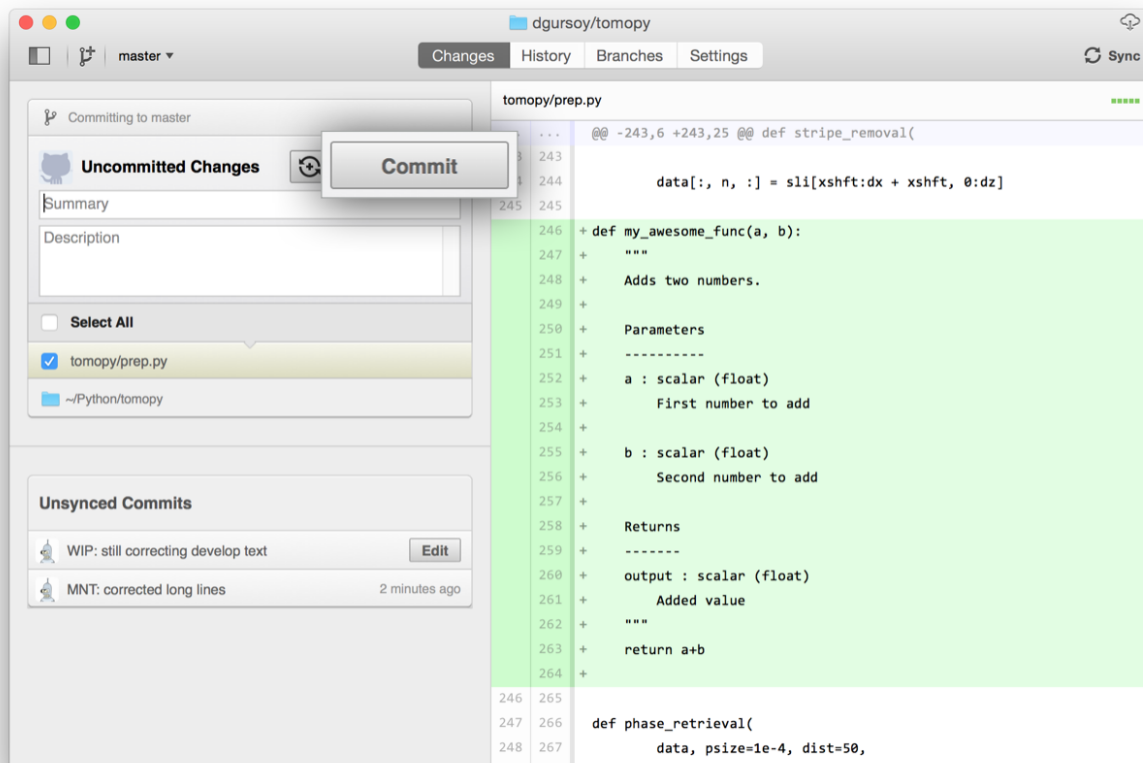


to repeat the doc build:

```
make clean
make html
```

Committing changes

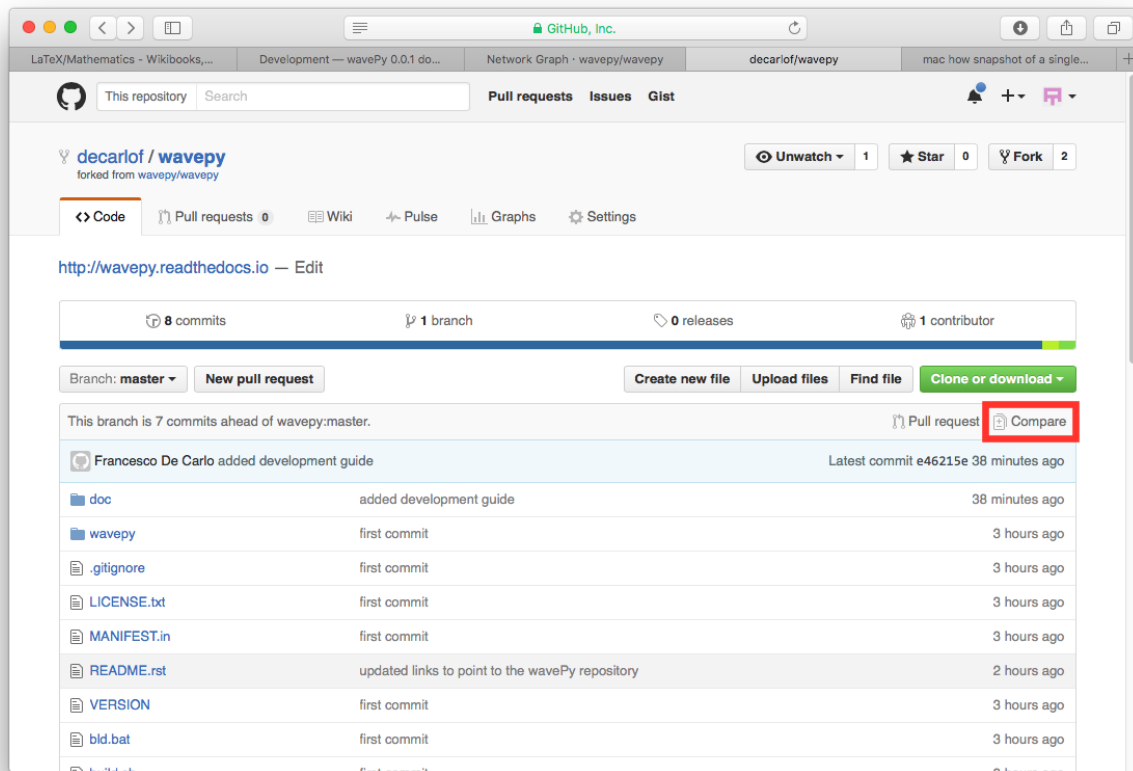
After making some changes in the code, you may want to take a *snapshot* of the edits you made. That's when you make a *commit*. To do this, launch the GitHub desktop application and it should provide you all the changes in your code since your last commit. Write a brief *Summary* and *Description* about the changes you made and click the **Commit** button:



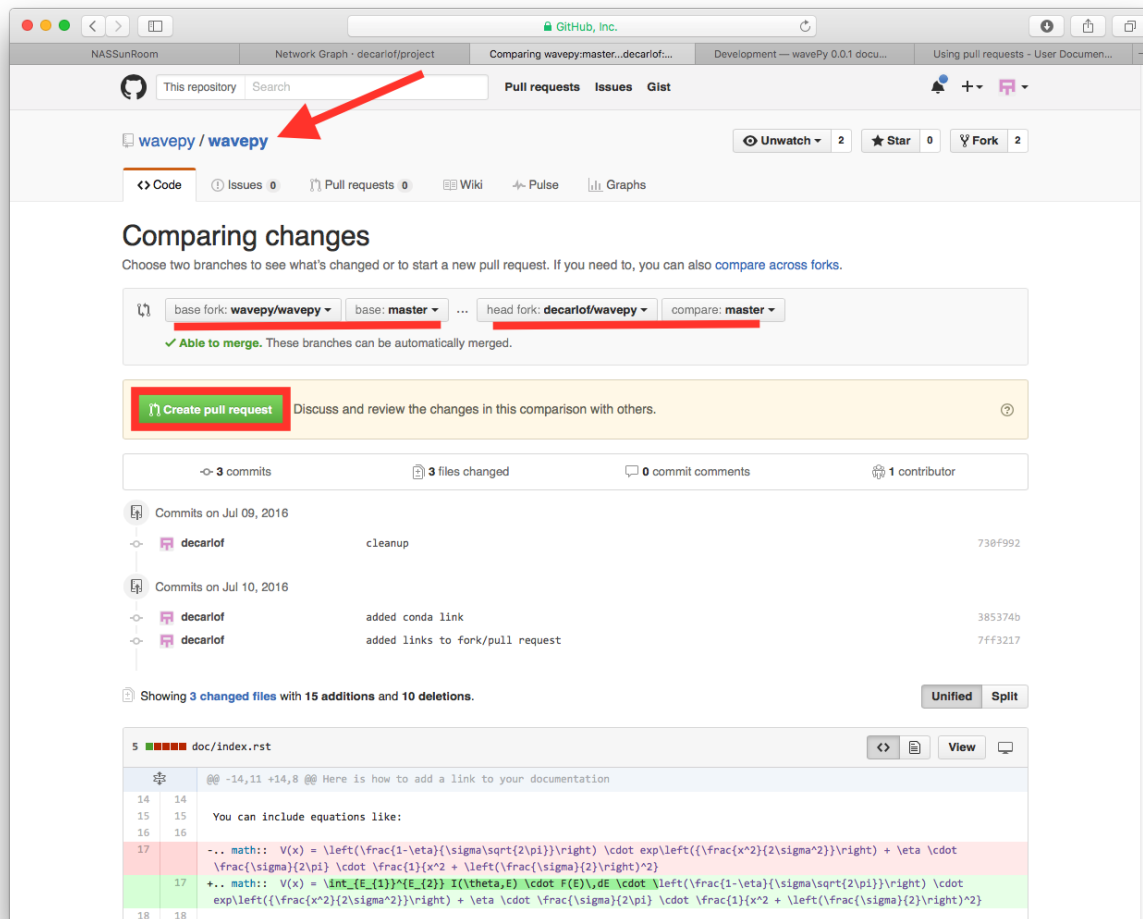
You can continue to make changes, add modules, write your own functions, and take more *Commit snapshots* of your code writing process.

Contributing back

Once you feel that the functionality you added would benefit the community, then you should consider contributing back to the project. For this, go to your online GitHub repository of the project and click on the *compare* button to compare, review and create a pull request.



After clicking on this button, you are presented with a review page where you can get a high-level overview of what exactly has changed between your forked branch and the original project repository. When you're ready to submit your pull request, click **Create pull request**:



Clicking on **Create pull request** sends you to a discussion page, where you can enter a title and optional description. It's important to provide as much useful information and a rationale for why you're making this Pull Request in the first place.

When you're ready typing out your heartfelt argument, click on **Send pull request**.

You're done!

1.3 Links

Here is a list of useful links

Name	Description
AreaDetector	The Area Detector documentation

1.4 Meetings

The BC Technical Working groups meets monthly. Minutes are kept [here](#) while slides of the presentations are accessible in the following table:

2022_08_11
2022_07_14
2022_06_09
2022_05_19
2022_04_14
2022_03_10
2022_02_10
2022_01_13
2021_09_23

1.5 Code Examples

Here we list some usefull code examples:

1.5.1 AD and PVA

This section contains a python example of how to grab an image from AreaDetector using PVA.

Download file: `example_01.py`

```
1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3
4  """
5  How to grab images from AreaDetector using pva
6  """
7
8
9  from epics import PV
10 import pvaccess as pva
11 import time
12 import numpy as np
13 import matplotlib.pyplot as plt
14
15
16 def streaming(buff):
```

(continues on next page)

(continued from previous page)

```

17 PVA_prefix = '2bmbPG1:Pva1:'
18
19 pva_data      = pva.Channel(PVA_prefix + 'Image')
20 pva_data_type = pva.Channel(PVA_prefix + 'DataType_RBV')
21 pva_image     = pva_data.get('')
22 pva_data_dict = pva_image.getStructureDict()
23
24 image        = pva_image['value']
25 width        = pva_image['dimension'][0]['size']
26 height       = pva_image['dimension'][1]['size']
27
28 print(image[0]['ubyteValue'])
29 image_np = np.reshape(image[0]['ubyteValue'], (width, height))
30 print(width, height)
31 print(image_np.shape)
32
33 # plt.imshow(image_np)
34 # plt.show()
35
36 print(pva_data_dict)
37
38 ix = width//2
39 iy = width//2
40 iz = height//2
41
42 # buffer is continuously update with monitoring
43 # the detector pv (function addProjection), called inside pv monitor
44 databuffer = np.zeros([buff, height, width], dtype='float32')
45
46 def addProjection(pv):
47     curid = pv['uniqueId']
48     print(curid)
49     databuffer[np.mod(curid, buff)] = pv['value'][0]['ubyteValue'].reshape(
50         height, width).astype('float32')
51
52 pva_data.monitor(addProjection, '')
53
54 while(True):
55     datap = databuffer.copy()
56     # print('new image')
57 if __name__ == "__main__":
58
59     buff = 50
60     streaming(buff)

```

1.5.2 WebCam

This section contains a python example of how to grab an image from a Panasonic iPro Extreme WV-S6130 WebCam.

Download file: `example_02.py`

```

1  #!/usr/bin/env python
2  # -*- coding: utf-8 -*-
3
4  """
5  How to grab images from a Panasonic iPro Extreme WV-S6130 web cam
6  """
7
8  import os
9  import cv2
10 import pathlib
11 import json
12 import matplotlib.pyplot as plt
13 from dotenv import load_dotenv
14
15 def read_json(fname):
16     with open(fname, 'r') as fp:
17         alist = json.load(fp)
18     return alist
19
20 # Create a file in the home directory called access.json
21 # {
22 #     "p" : "password",
23 #     "u" : "username",
24 #     'ip' : "192.168.1.1" # replace with the webcam IP address
25 # }
26
27
28 access_fname = os.path.join(str(pathlib.Path.home()), 'access.json')
29 access_dic = read_json(access_fname)
30
31 try:
32
33     ret, frame = cv2.VideoCapture('http://' + access_dic['u'] + ':' + access_dic['p'] + '@
↪ ' + access_dic['ip'] + '/cgi-bin/mjpeg?stream=1').read()
34     if ret:
35         plt.imshow(frame)
36         plt.show()
37
38 except FileNotFoundError:
39     log.error('username and password are missing. Set them in a file called: %s' %
↪ access_fname )

```

1.6 Credits

1.6.1 Citations

Here is an example of how to add a reference: [A1] or [B1].

BIBLIOGRAPHY

- [A1] ML Rivers and others. Areadetector: epics software for 2-d detectors. In *Proc. 16th Int. Conf. on Accelerator and Large Experimental Physics Control Systems*. 2017. URL: <https://doi.org/10.18429/JACoW-ICALEPCS2017-THDPL03>.
- [B1] Mark L. Rivers. Areadetector: software for 2-d detectors in epics. *AIP Conference Proceedings*, 1234(1):51–54, 2010. URL: <https://aip.scitation.org/doi/abs/10.1063/1.3463256>, arXiv:<https://aip.scitation.org/doi/pdf/10.1063/1.3463256>, doi:10.1063/1.3463256.