



The New Qhub 0.3.0

An easy way to deploy/maintain JupyterHub with Dask Gateway

Laís Carvalho

Developer Advocate



April | 2021



Quansight

Contents



- Who I am
- Behind QHub: Quansight
- QHub - what it is and what's new.
- Demo
- Q&A/Discussion



@lais_bsc



laisbsc

Who am I?

You Name

- Developer Advocate at Quansight
- QHub's tech-writer and community-manager
- Passionate about Python and empowering devs with open source tools.

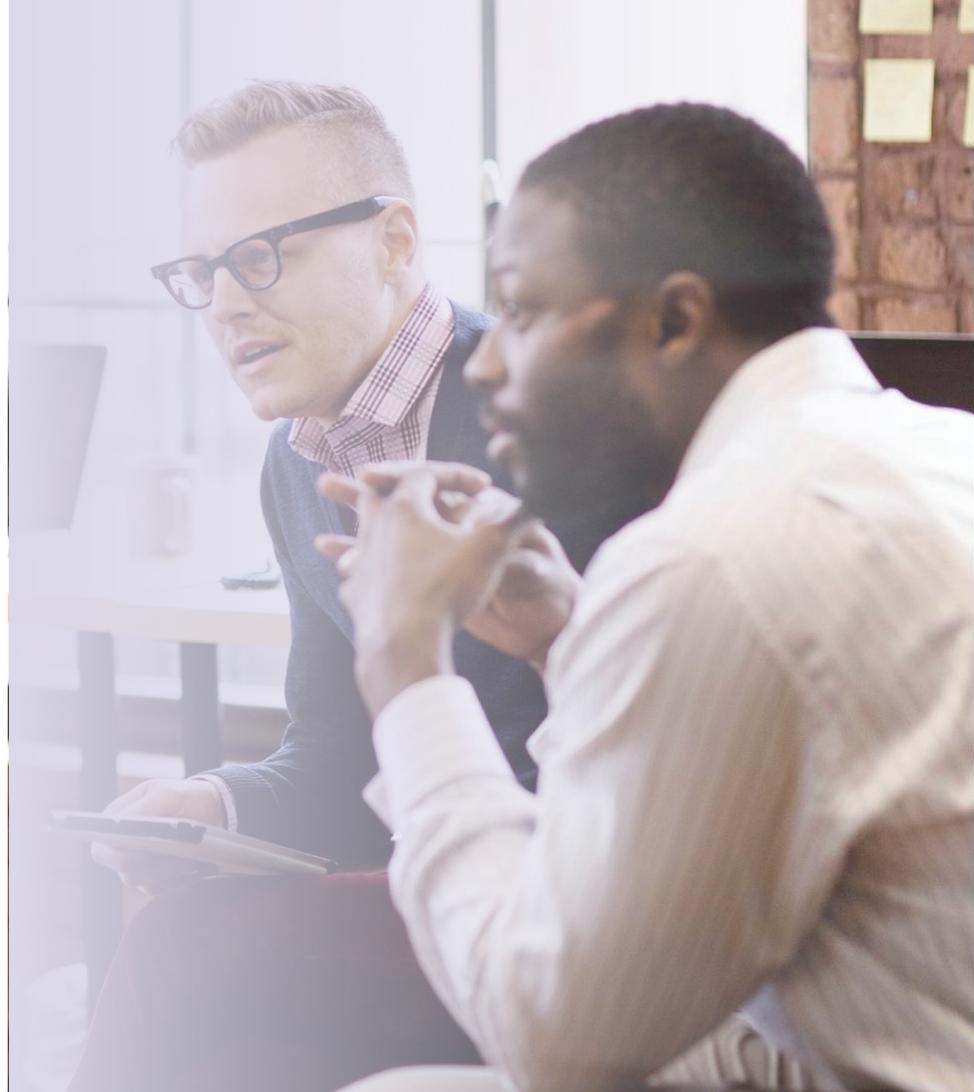
Background

- Civil and environmental engineering
- Final year IT student
- Python-newbie

Quansight

Mission

Bridge the creators and maintainers of Open Source tools with the companies that use them.



Our Story



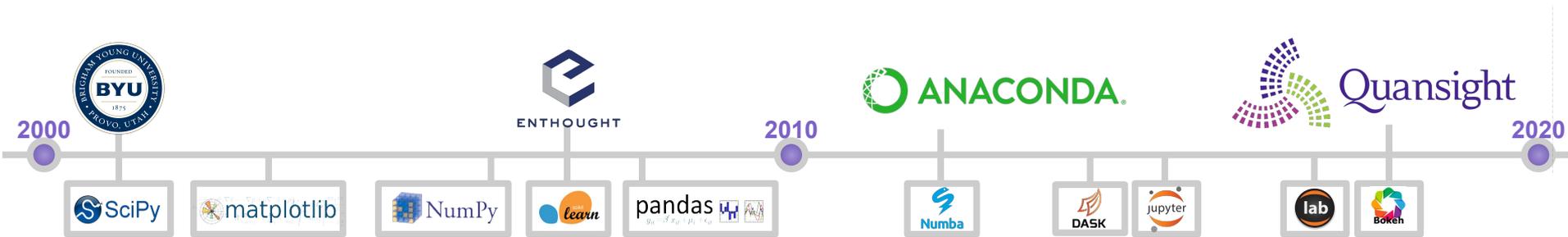
Travis Oliphant
Co-Founder & CEO

Open Source Leader

- Professor of Electrical Engineering at BYU
- Major contributor and leader in the open source software community
- Created SciPy, NumPy and Numba, and managed the creation of many others

Entrepreneur

- President of Enthought, (2007-2011).
- Founded Anaconda (2011 - 2018), created industry standard platform for Python data science
- Founded Quansight in 2018 with the mission to bridge between creators of open source and the enterprises that use it.



Define QHub

Qhub is an open source library that uses an infrastructure-as-code approach to deploy a shared data science environment, on premises or in the Cloud.

Tech Stack

- QHub (on the Cloud with K8s)



- QHub HPC
 - MPI support





“If you know GitHub and feel comfortable generating and using API keys, you should have **all it takes** to deploy and maintain your own compute infrastructure without the need for a dedicated DevOps team.”

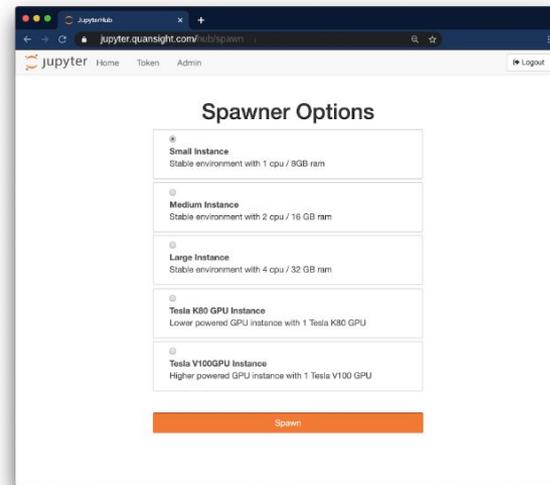
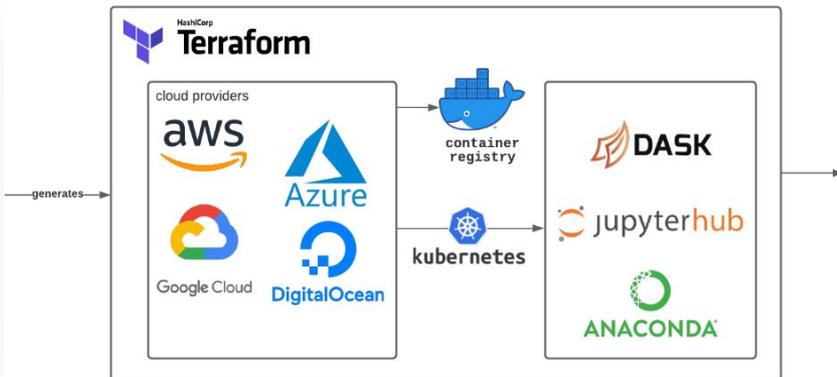


Born from the need to quickly deploy compute platforms on the Cloud.

QHub has a *DevOps for non-DevOps people* approach.

```
qhub-config.yml

security:
  users:
    <username>@quansight.com:
      ...
  ...
google_cloud_platform:
  region: us-east-1
  ...
general:
  min_nodes: 1
  max_nodes: 100
  accelerators: nvidia-tesla-v100
  ...
profiles:
  ...
environments:
  ...
```



Deploy with

1

qhub init ...

Creates 'qhub-config.yaml' file and optionally automates oath, DNS registry, repo creation

2

qhub deploy ...

Deploys QHub to cloud based on Terraform files

Server Options



Small Instance

Stable environment with 1 cpu / 1 GB ram



Medium Instance

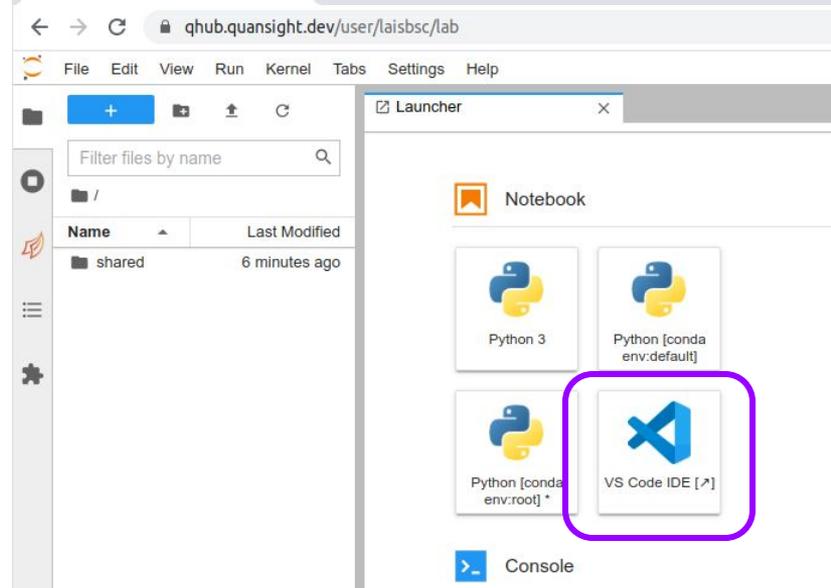
Stable environment with 1.5 cpu / 2 GB ram

Start

New on QHub

1 Integration with VSCode (through code-server)

Edit your code using a browser-based version of VSCode (open source)



2 Revamped Dask Gateway

Access Dask clusters on the Cloud from your local computer.

New on QHub



CDS Dashboarding

Make visualization dashboards using your preferred tool.

Share with other users on your Qhub cluster.



Streamlit

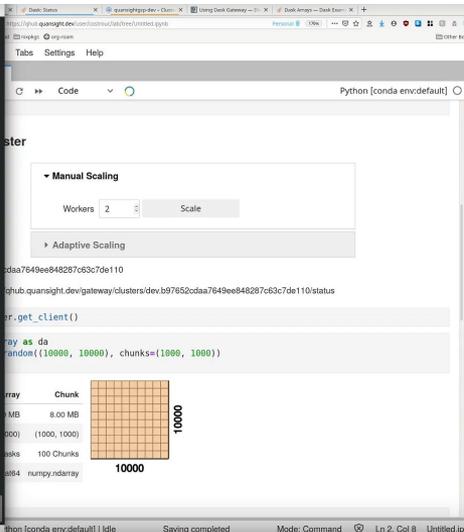


```
File Edit View Bookmarks Settings Help
sk_gateway/client.py", line 455, in list_clusters
return self.sync(self._clusters, status=status,
s, **kwargs)
File "/opt/conda/lib/python3.8/site-packages/dask_gateway/client.py", line 343, in sync
return future.result()
File "/opt/conda/lib/python3.8/concurrent/futures/_base.py", line 439, in result
return self._get_result()
File "/opt/conda/lib/python3.8/concurrent/futures/_base.py", line 388, in _get_result
raise self._exception
File "/opt/conda/lib/python3.8/site-packages/dask_gateway/client.py", line 434, in _clusters
resp = await self._request("GET", url)
File "/opt/conda/lib/python3.8/site-packages/dask_gateway/client.py", line 419, in _request
resp.raise_for_status()
File "/opt/conda/lib/python3.8/site-packages/aiohttp/client_reqrep.py", line 1000, in raise_for_status
raise ClientResponseError(
aiohttp.client_exceptions.ClientResponseError: 401, message='Unauthorized', url=URL('https://qhub.quansight.dev/gateway/api/v1/clusters/')
>>> gateway = Gateway(address='https://qhub.quansight.dev/gateway', auth='jupyterhub', proxy_address='tcp://qhub.quansight.dev:8786')
>>> gateway.list_clusters()
[ClusterReport(name=dev_b97652cdaa7649ee848287c63c7de110, status=RUNNING)]
>>> cluster = gateway.new_cluster(options)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'options' is not defined
>>> cluster = gateway.new_cluster()
>>> client = cluster.get_client()
>>> client
<Client: 'tls://10.72.0.2:8786' processes=0 threads=0, memory=0 B>
>>>
```



Connect to JupyterHub-SSH

Connect to QHub via SSH or using local code editors like VSCode, PyCharm, etc.



New on QHub

5 Install QHub on existing K8s clusters

Test your existing Kubernetes clusters locally using Minikube.

Setup by simply specifying `kube_context`.



Local (Existing) Kubernetes Cluster ¶

Deploying to a local existing Kubernetes cluster has different options than the cloud providers. `kube_context` is an optional key that can be used to deploy to a non-default context. The default node selectors will allow pods to be scheduled anywhere. This can be adjusted to schedule pods on different labeled nodes. Allowing for similar functionality to node groups in the cloud.

```
local:  
  kube_context: minikube  
  node_selectors:  
    general:  
      key: kubernetes.io/os  
      value: linux  
    user:  
      key: kubernetes.io/os  
      value: linux  
    worker:  
      key: kubernetes.io/os  
      value: linux
```

6 Added support for Microsoft Azure

Choose the cloud provider to host your computing platform (AWS, GCP, Azure or DigitalOcean).

Demo

Let's take a look at some of the main features of the latest release.



Summary (TIL)

- Qhub is an OS tool that helps to build and maintain compute infrastructure
- Enables collaboration - several users on the same notebook
- Uses infrastructure-as-code to deploy
- GitHub Actions to automate CI/CD deployment (GitOps approach)
- Kubernetes & Terraform for non-DevOps devs.
- Newest release has a revamped Dask Gateway, integration with VSCode (code-server), CDS Dashboarding and much more!



Questions?



<https://github.com/Quansight/qhub>



<https://github.com/Quansight/qhub-hpc>

contact@quansight.com

Austin, Texas



Quansight.
YOUR DATA EXPERTS