

The  
Alan Turing  
Institute

---

# Sharing Reproducible Python Environments with Binder

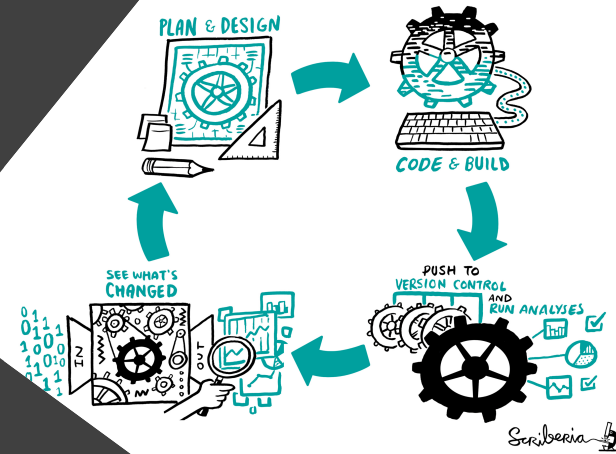
Sarah Gibson

Pronouns: she/her



@drsarahlgibson @mybinderteam #EuroPython  
<https://doi.org/10.5281/zenodo.3937310>

# What is Reproducibility?





		Data	
		Same	Different
Analysis	Same	Reproducible	Replicable
	Different	Robust	Generalisable

Kirstie Whitaker's talk at PyData LDN: <https://youtu.be/IG3PcZ6EhiU>  
<https://the-turing-way.netlify.app/reproducible-research/overview/overview-definitions.html#table-of-definitions-for-reproducibility>

@drsarahlgibson @mybinderteam #EuroPython  
<https://doi.org/10.5281/zenodo.3937310>

		Data	
		Same	Different
Analysis	Same	<div>Repeatable</div> Reproducible	Replicable
	Different	Robust	Generalisable

Kirstie Whitaker's talk at PyData LDN: <https://youtu.be/IG3PcZ6EhiU>  
<https://the-turing-way.netlify.app/reproducible-research/overview/overview-definitions.html#table-of-definitions-for-reproducibility>

@drsarahlgibson @mybinderteam #EuroPython  
<https://doi.org/10.5281/zenodo.3937310>

Is not considered  
for promotion

Held to higher  
standards than  
others

Publication bias  
towards novel  
findings

# Barriers to reproducible research

Requires  
additional  
skills

Plead the 5th

Support additional  
users

Takes time

---

# Market Research



@drsarahlgibson @mybinderteam #EuroPython  
<https://doi.org/10.5281/zenodo.3937310>

---

Have you ever heard...?

*“Oh, it worked on  
my computer?”*

---

Have you ever heard...?

*“Oh, it worked  
yesterday?”*



---

*“Oh, it worked on  
my computer?”*



+ CI

---

*“Oh, it worked  
yesterday?”*

---

# Binder



@drsarahlgibson @mybinderteam #EuroPython  
<https://doi.org/10.5281/zenodo.3937310>

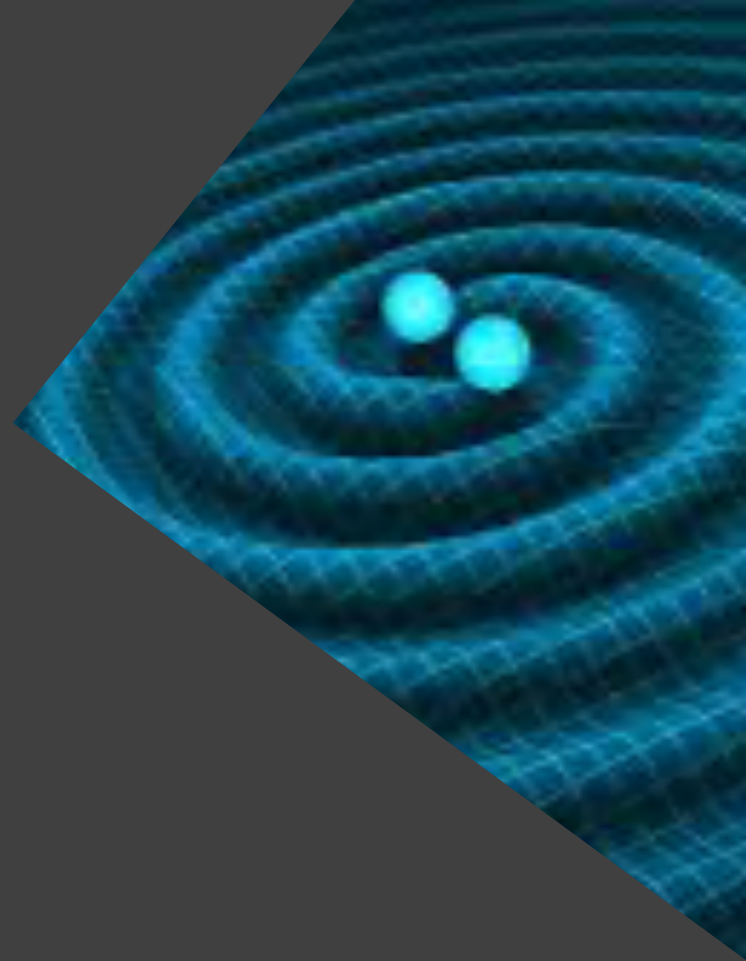


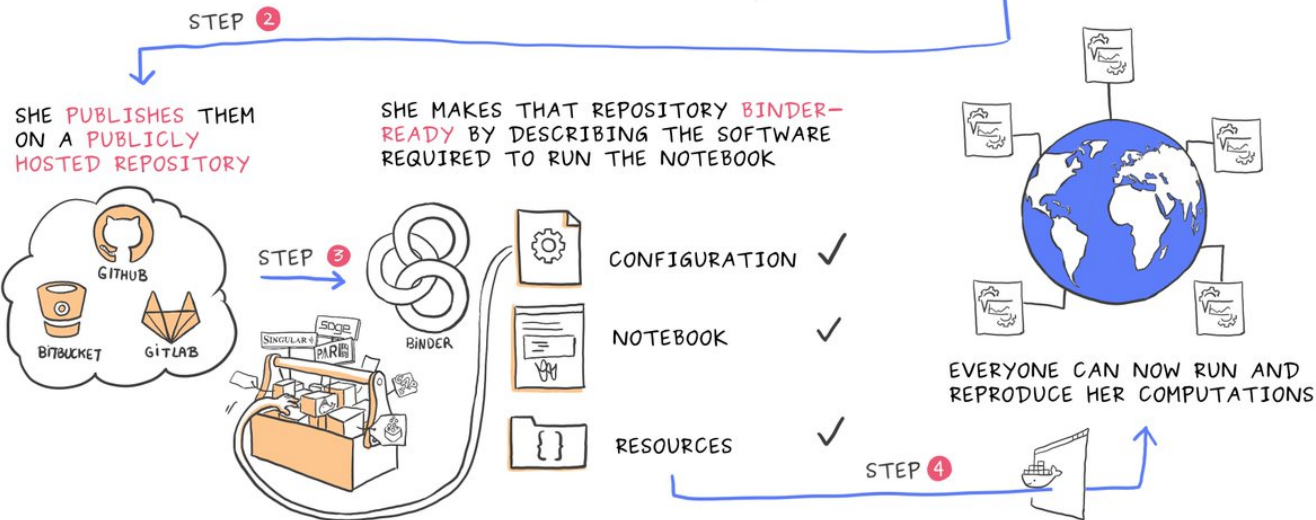
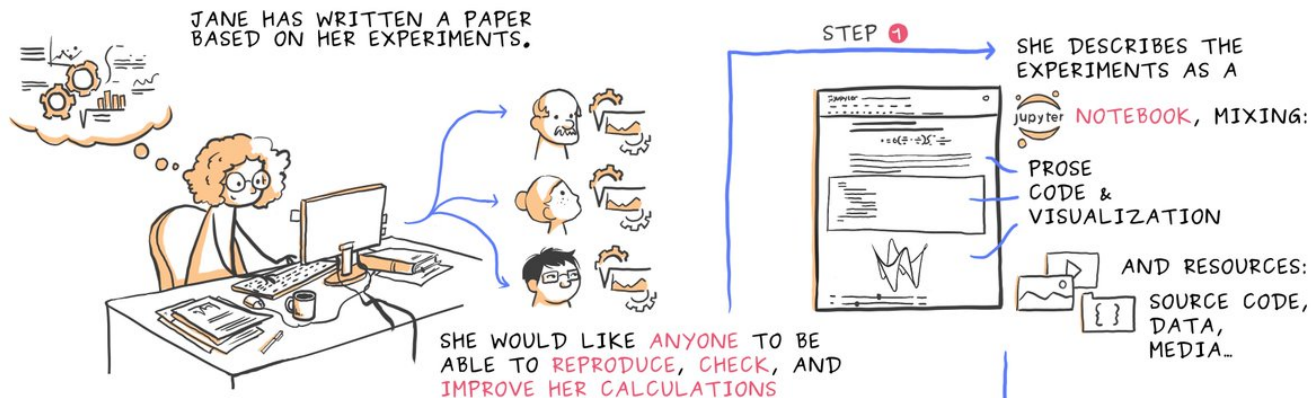
- Project Binder is a global community
- The mybinder.org service allows anyone to launch a complete, interactive computing environment from their browser



---

[gw-openscience.org/tutorials/](http://gw-openscience.org/tutorials/)





mybinder.org

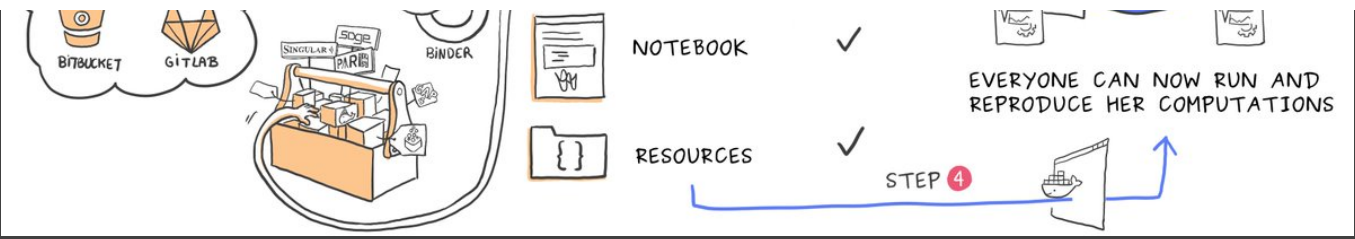
choldgraf Update requirements.txt 21a328d on 21 Jun

2 contributors

5 lines (3 sloc) 46 Bytes

Raw Blame History

```
1 numpy==1.16.*
2 matplotlib==3.*
3 seaborn==0.8.1
4
```



&lt;&gt; Code

Issues 2

Pull requests 0

Projects 0

Wiki

Security

Insights

Branch: master ▾

conda / environment.yml

Find file

Copy path



betatim Update environment.yml

89dd429 on 11 Dec 2018

4 contributors



14 lines (13 sloc) | 161 Bytes

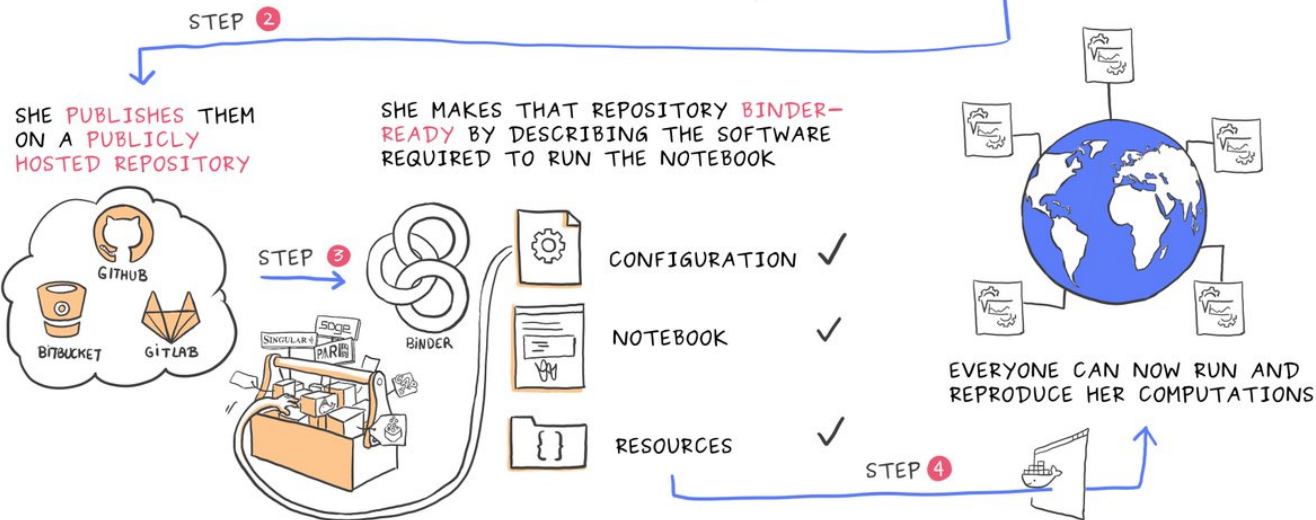
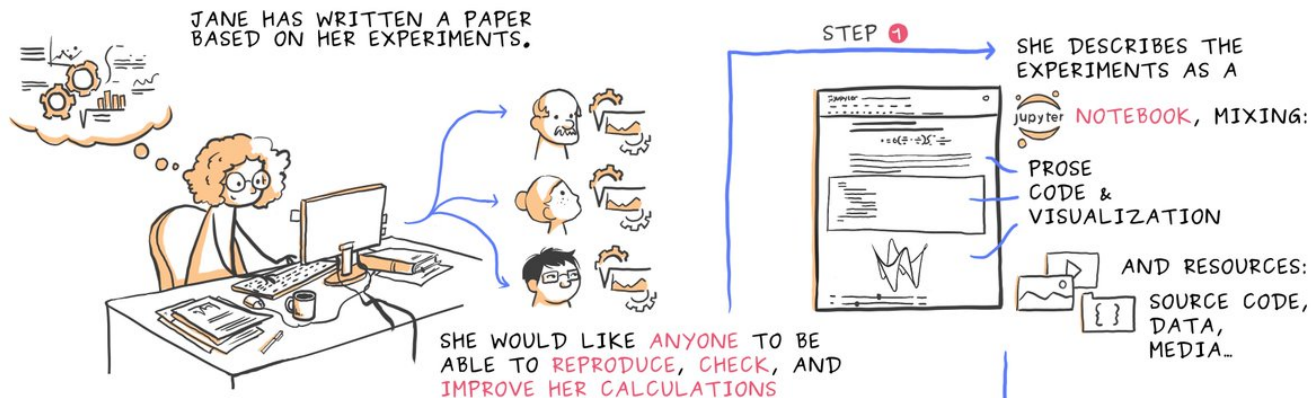
Raw

Blame

History



```
1 name: example-environment
2 channels:
3   - conda-forge
4 dependencies:
5   - numpy
6   - psutil
7   - toolz
8   - matplotlib
9   - dill
10  - pandas
11  - partd
12  - bokeh
13  - dask
```



mybinder.org

## Overview

Sessions ▾ VS. [Select a metric](#)

Hourly Day Week M

● Sessions

150,000

100,000

50,000

2018

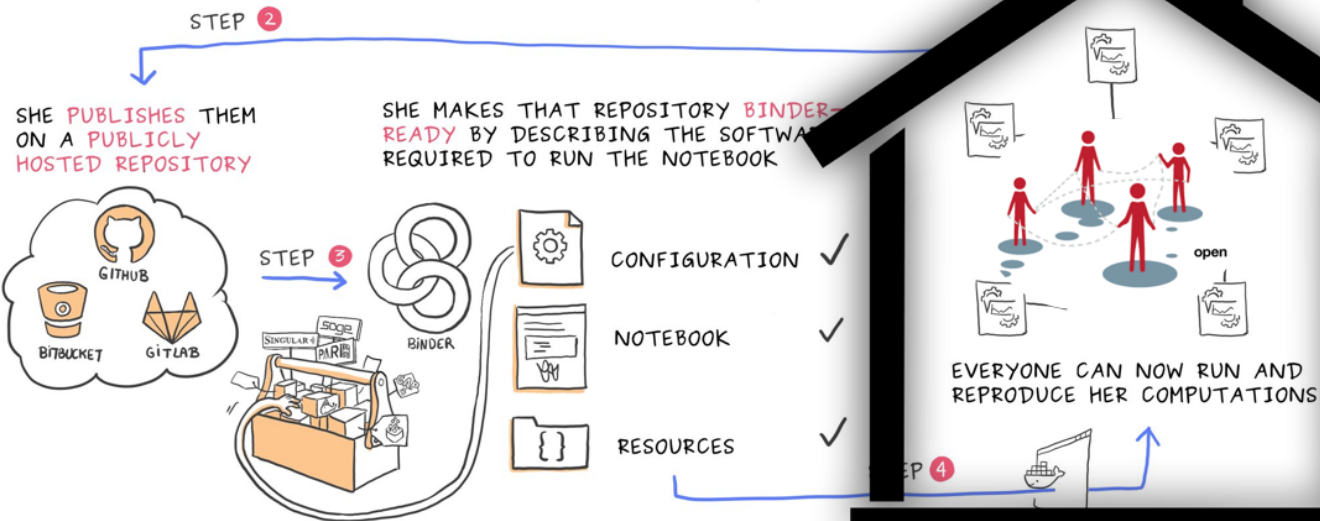
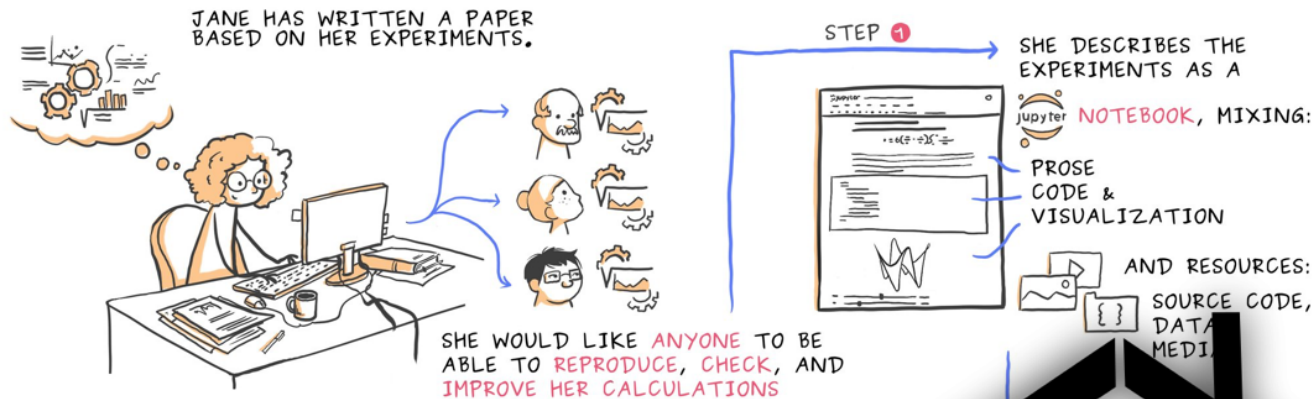
2019

2020

May 10, 2020 - May 16, 2020  
■ Sessions: 144,396

- Launched by Jeremy Freeman in 2015
- First Binder and Jupyter meeting in January 2017
- First half of 2017 spent redeveloping the backend into what is now BinderHub
- Sept 24, 2017 – Moore Foundation Binder proposal accepted





mybinder.org



---

# Technology



@drsarahlgibson @mybinderteam #EuroPython  
<https://doi.org/10.5281/zenodo.3937310>


# BinderHub

Build and launch a repository

GitHub repository name or URL

GitHub ▾

Git branch, tag, or commit



Path to a notebook file (optional)

File ▾

launch

Clone GitHub Repo

1




# BinderHub

Build and launch a repository

GitHub repository name or URL

GitHub ▾

Git branch, tag, or commit

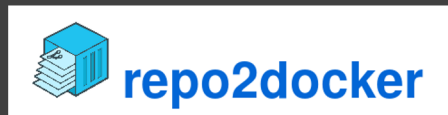


Path to a notebook file (optional)

File ▾

launch

1 Clone GitHub Repo



2 Build image according to instructions contained within the repo

# BinderHub

Build and launch a repository

GitHub repository name or URL

GitHub ▾

Git branch, tag, or commit



Path to a notebook file (optional)

File ▾

launch



1 Clone GitHub Repo

2

Build image  
according to  
instructions  
contained within the  
repo

3

Execute image


# BinderHub

Build and launch a repository

GitHub repository name or URL

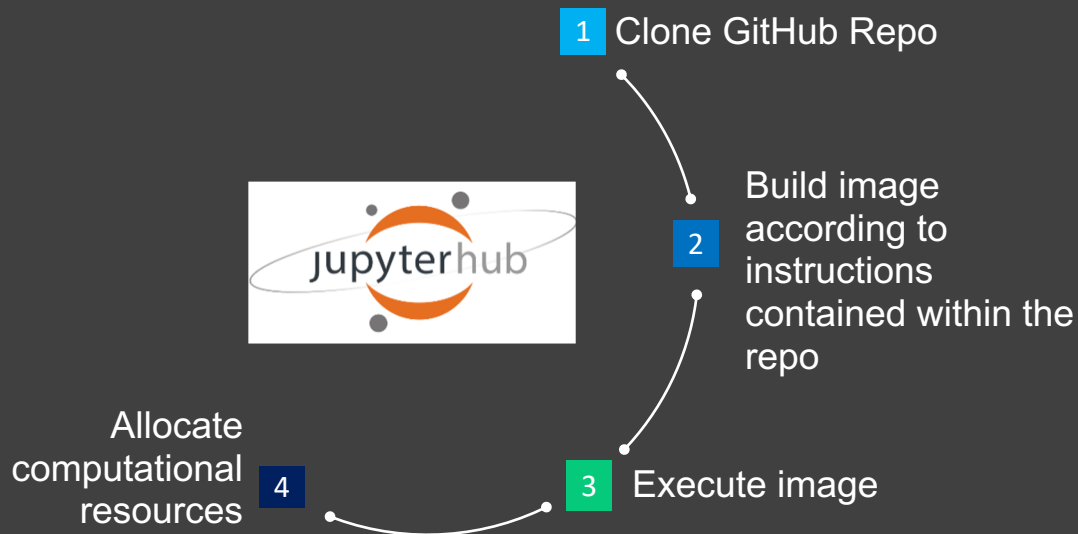
GitHub ▾

Git branch, tag, or commit

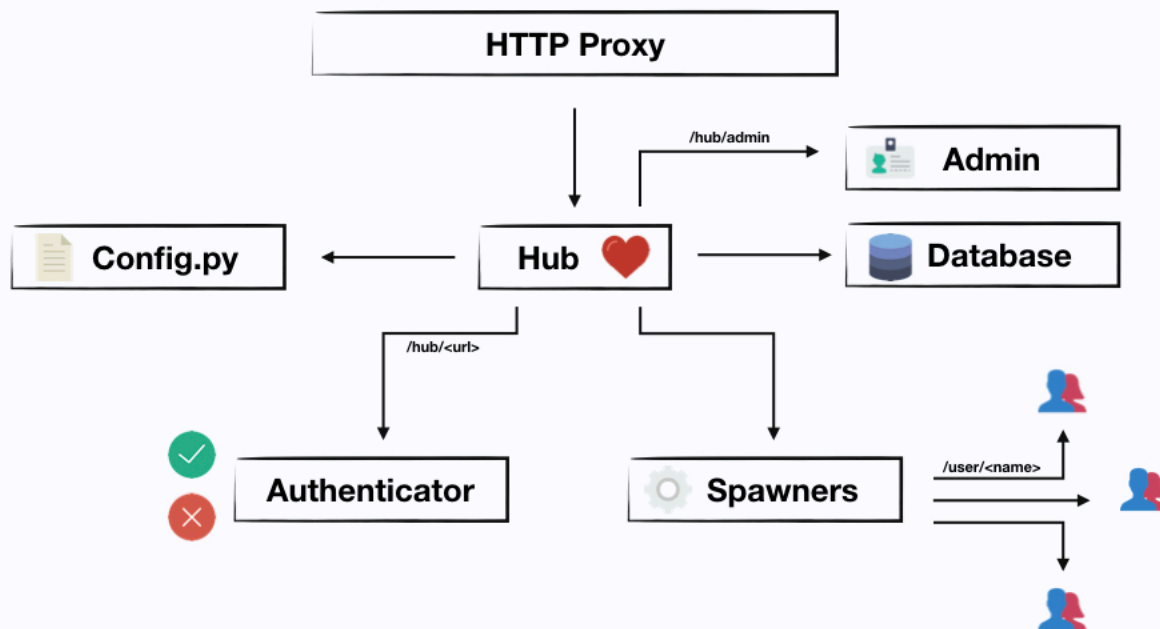


Path to a notebook file (optional)

File ▾



# What is a JupyterHub?



All icons were obtained from Flaticon (<https://www.flaticon.com/packs/essential-collection>)

resources

JupyterHub is a way to help your humans use your computers. With notebooks!

hin the


# BinderHub

Build and launch a repository

GitHub repository name or URL

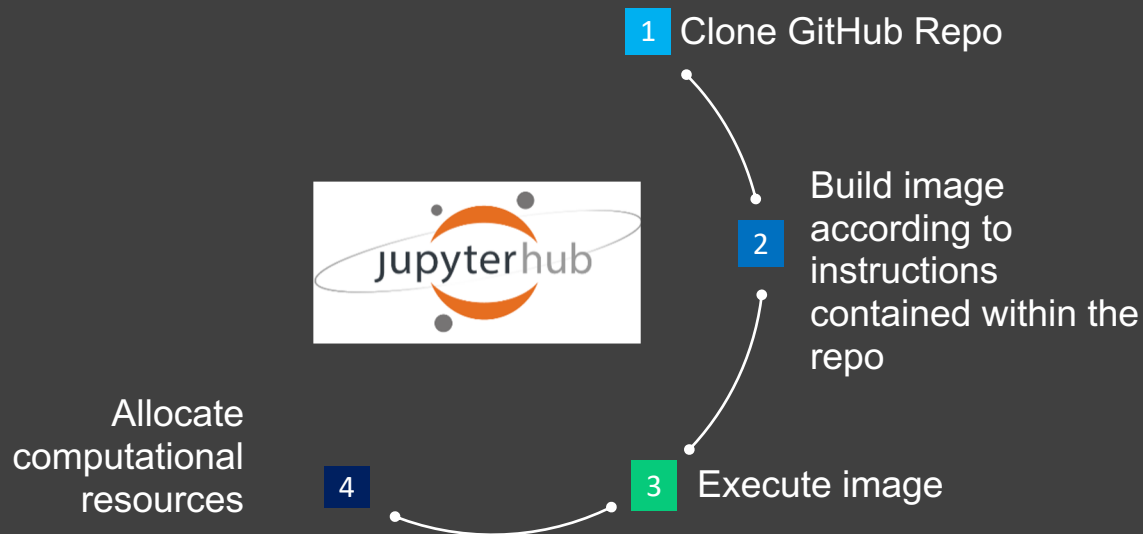
GitHub ▾

Git branch, tag, or commit



Path to a notebook file (optional)

File ▾




# BinderHub

Build and launch a repository

GitHub repository name or URL

GitHub ▾

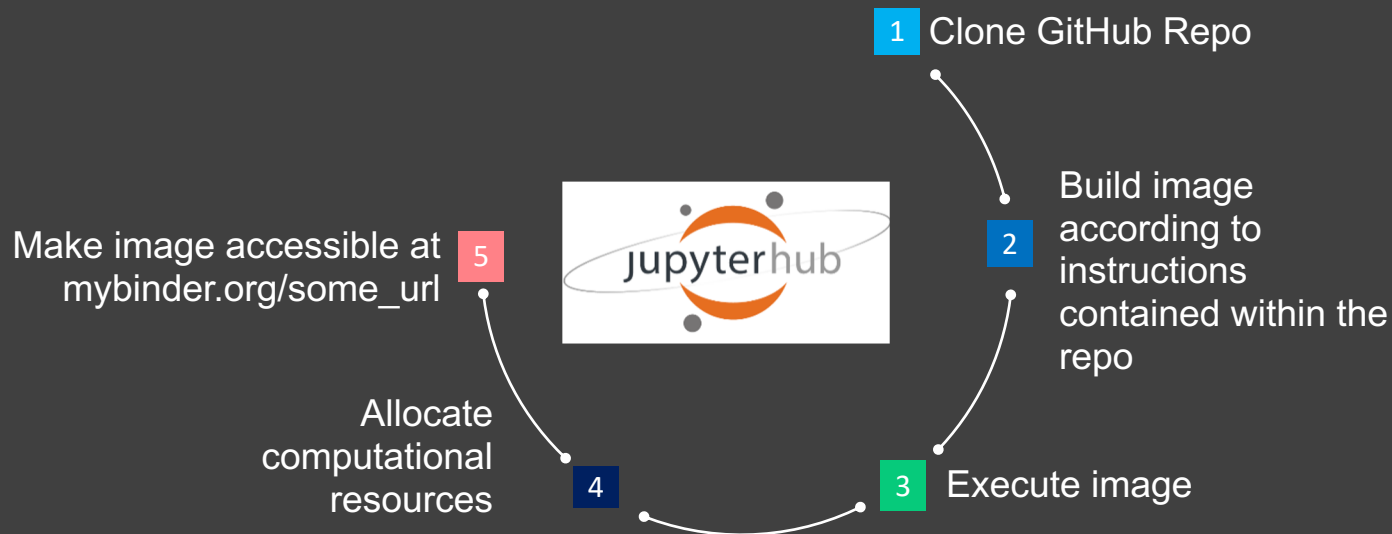
Git branch, tag, or commit



Path to a notebook file (optional)

File ▾

launch






# BinderHub

Build and launch a repository

GitHub repository name or URL

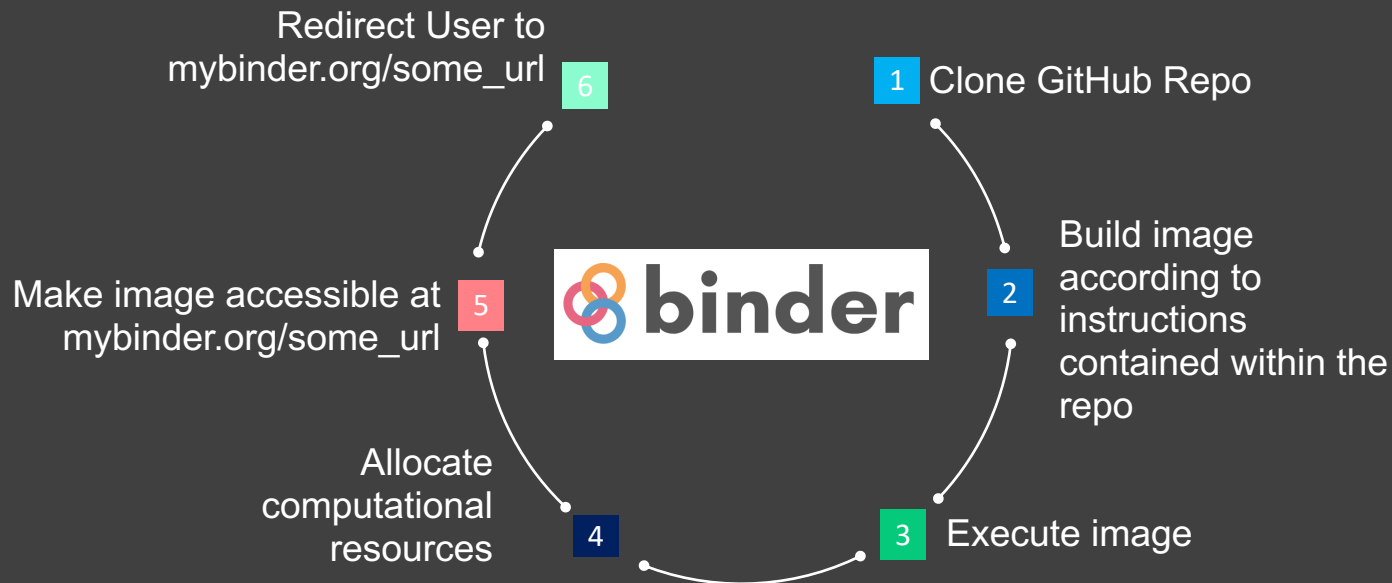
GitHub ▾

Git branch, tag, or commit



Path to a notebook file (optional)

File ▾

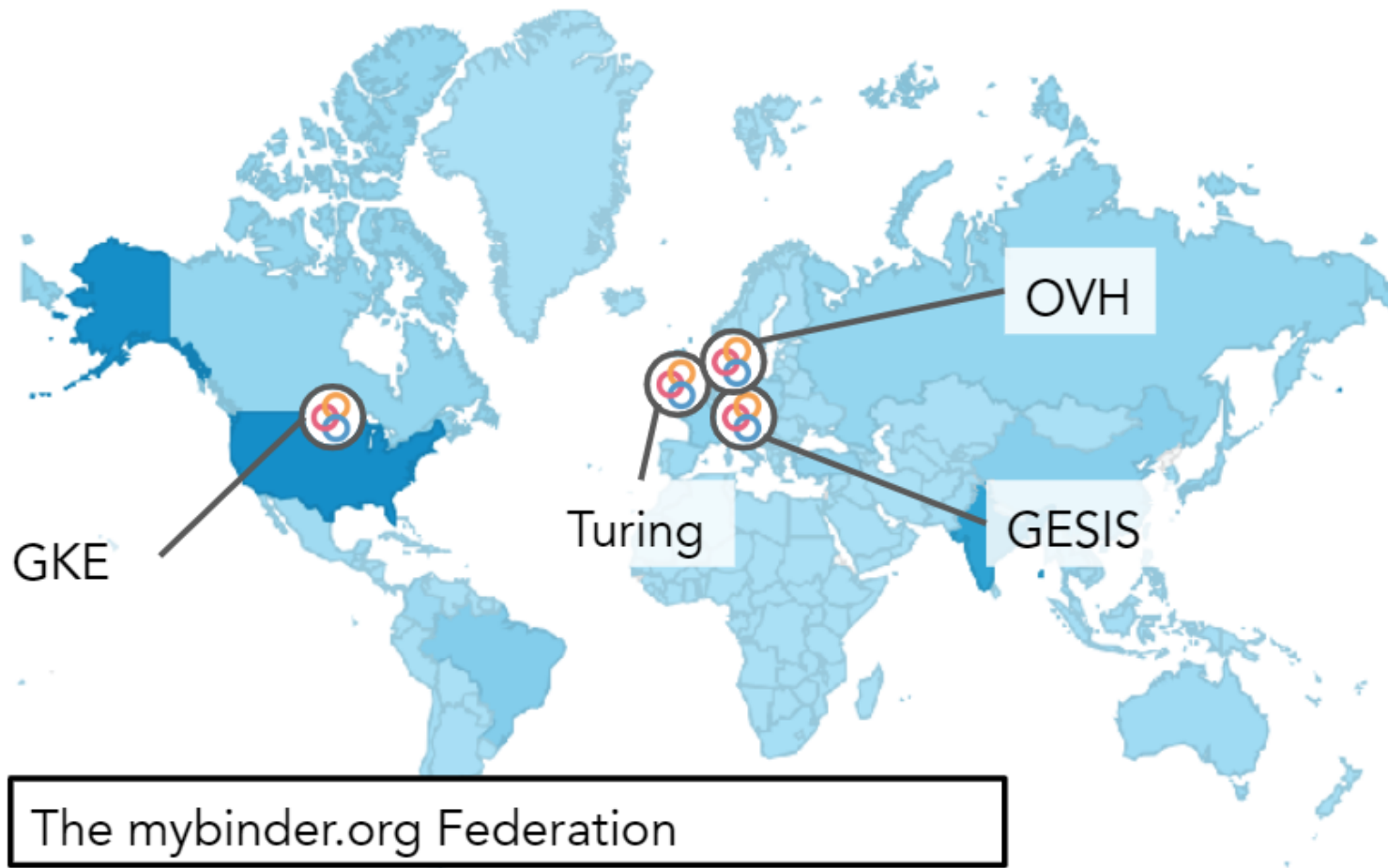


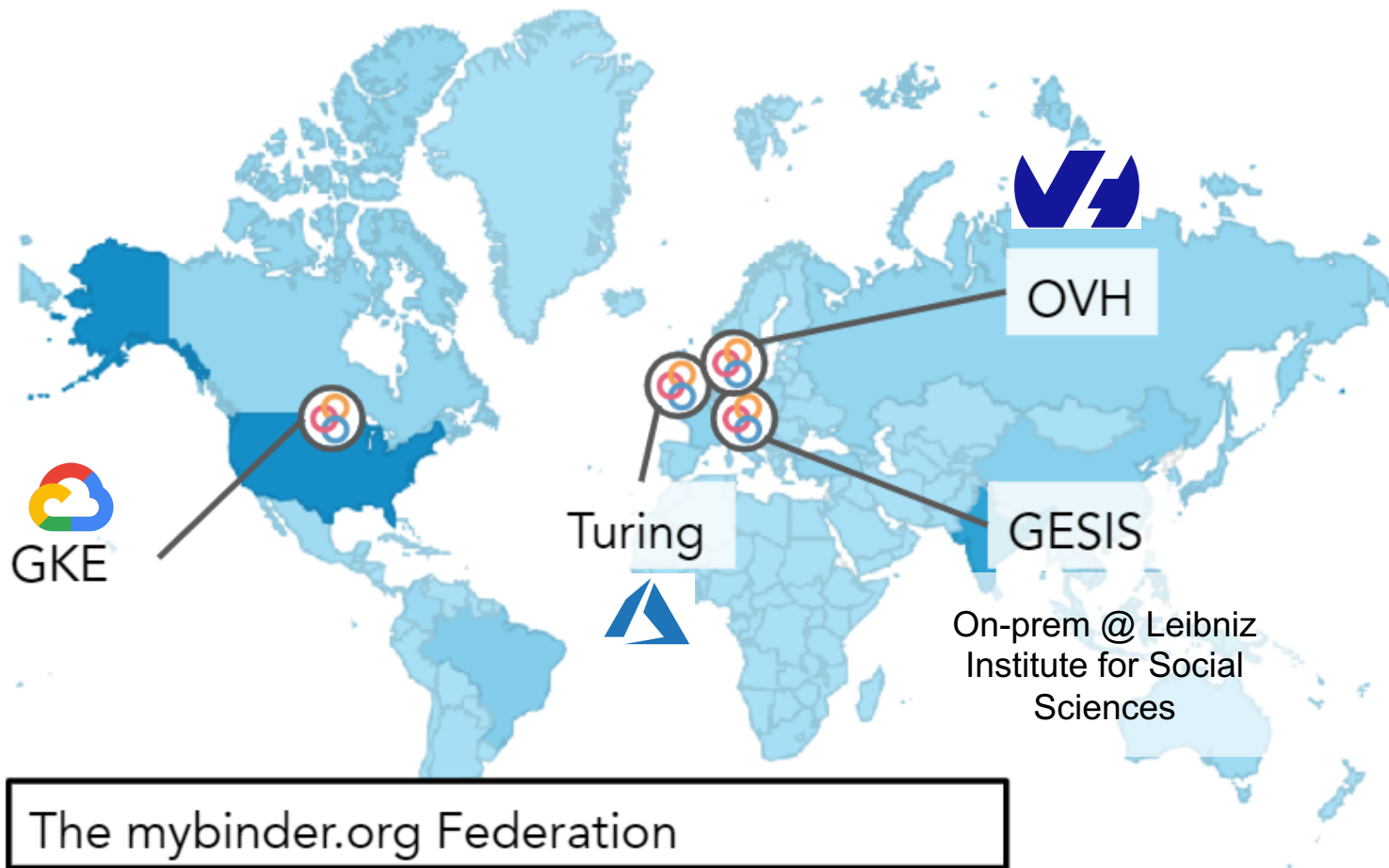
---

# Scaling up with the Federation

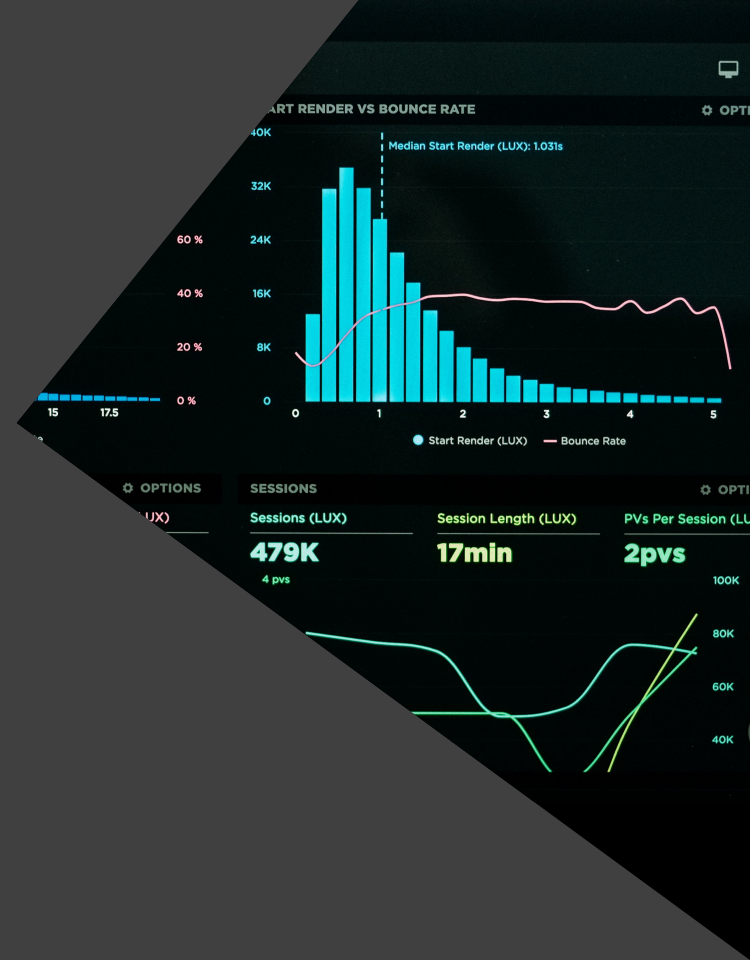


@drsarahlgibson @mybinderteam #EuroPython  
<https://doi.org/10.5281/zenodo.3937310>

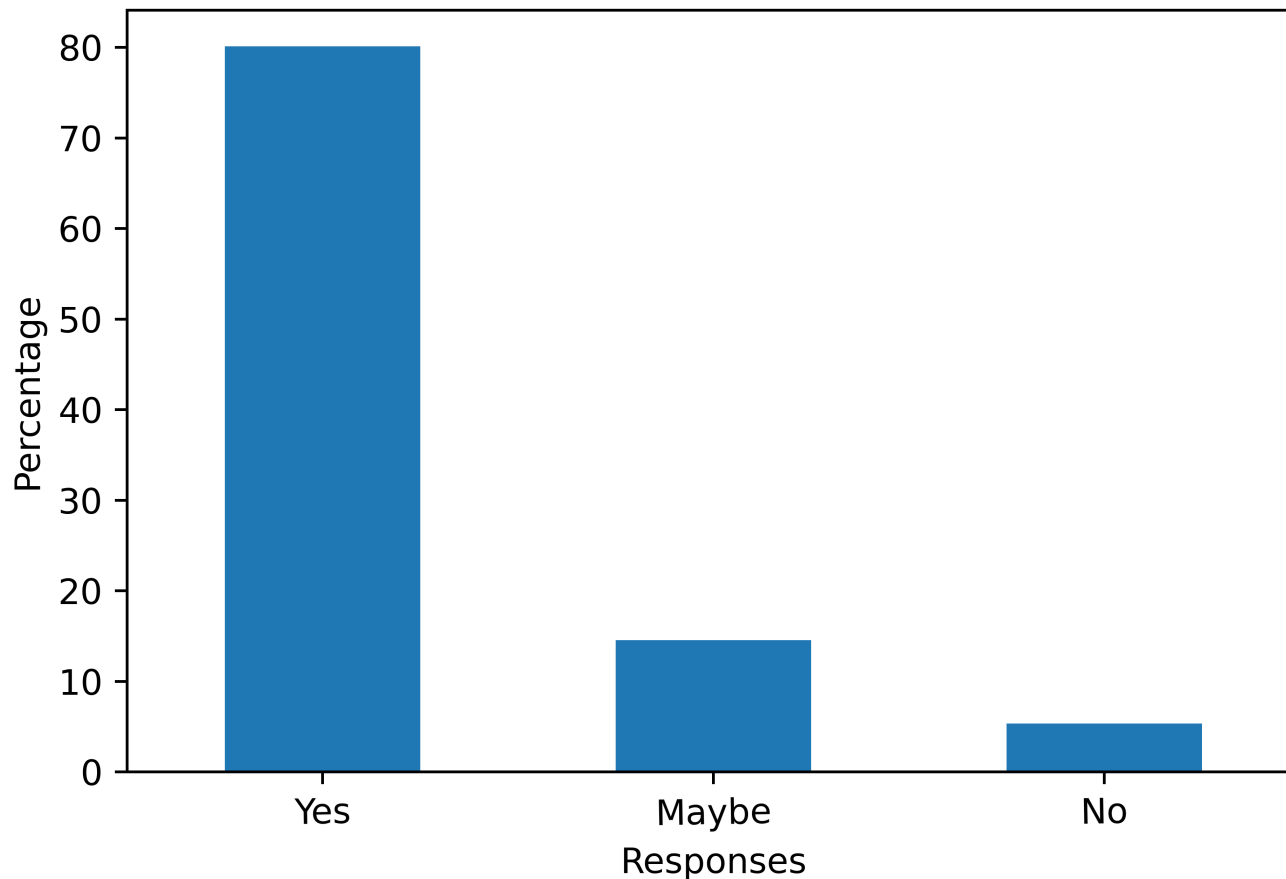




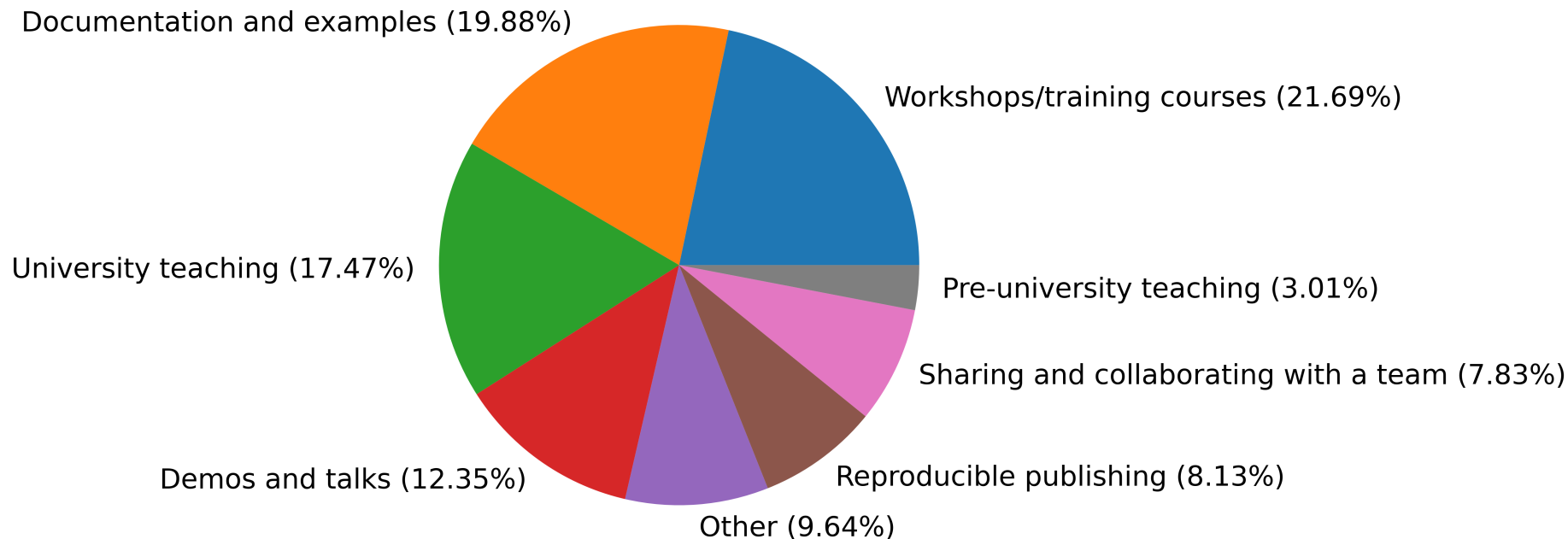
# User Survey



## Would you recommend mybinder.org to a friend?



## What do you (mainly) use mybinder.org for?

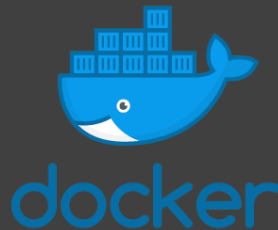






# Picking up speed

**UPSTREAM**



# Community Guidance

We have written advice on best practices for achieving faster launch times



## How to reduce mybinder.org repository startup time

■ Binder ■ discuss

### What affects launch time?

The challenge between running [mybinder.org](#) <sup>2</sup> vs. a different cloud service such as Colab is that Binder is meant to run *arbitrary environments* that you define in a GitHub repository. Most of the time when a repository is (very) slow (more than 30s) to launch it is because the environment for that session must be built and initialized. This mostly happens to people “developing” on a repository (constantly changing things and launching right away).

For most users of a Binder link the environment is already built. This is because someone else has previously launched the same version. this can still be slow but not very slow (more than 30s).

[mybinder.org](#) <sup>2</sup> runs on Kubernetes, which runs a cluster that grows and shrinks as necessary to take on new users. Each time a user clicks a Binder link, these things happen:

1. A slot (called a “pod”) is reserved on one of the cloud machines
2. Binder looks to see if a Docker image exists for that repository
  - If it doesn't, Binder must first build the image for that repo using `repo2docker` (*this takes time*)
3. Binder looks for a built image on the *machine* the user will use
  - If it isn't on the machine, Binder must first *pull* the image onto that machine (*this takes time*)
4. Binder launches the user's session



## How to reduce mybinder.org repository startup time

■ Binder ■ discuss

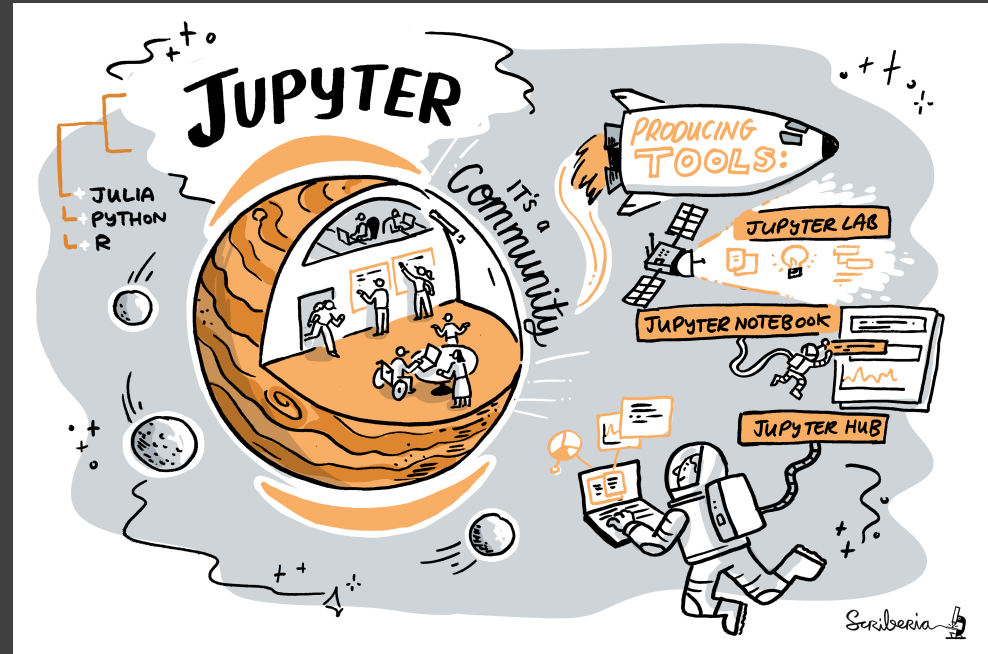
### How can I reduce my launch time?

With that being said, in order to reduce the amount of time it takes your repository to launch, try these steps:

- **Make your repository environment more light-weight** - A repository with fewer dependencies and a smaller size will be faster to both build and download into the Binder session.
- **Ensure your repository gets a lot of clicks** - The more often that a repository is launched, the more likely it will already be built and downloaded to a machine when a user starts a new session. As a result, the more popular a repository is, the faster launches will tend to take.
- **Use two repositories: one for the environment, one for your content** - many people change their content much more often than they change the environment needed for it. However, Binder will re-build the environment for *any* changes to a repository. A hack to get around this is to define an “environment repository” that Binder builds, and use a hook to *pull in new content at launch* from a “content repository”. This means that your “environment repository” changes less-often, which should result in fewer new builds and reduced launch times. [See the instructions in this post](#) to get started.
- **Use the `nbgitpuller`.link page to automate separate content/environment repos**. The above step can be (mostly) automated by using `nbgitpuller.link`. This is a little web form

# Community

- Value meeting communities where they are
- Diversify our skills



Software  
Sustainability  
Institute

---

# Find out more and get involved

- GitHub: <https://github.com/jupyterhub/binderhub>
- Website: <https://mybinder.readthedocs.io> and <https://mybinder.org>
- Discourse: <https://discourse.jupyter.org/c/binder>
- Gitter: <https://gitter.im/jupyterhub/binder>
- Twitter: [@mybinderteam](https://twitter.com/mybinderteam)
- Binder tutorial: [bit.ly/zero-to-binder-python](http://bit.ly/zero-to-binder-python)
- Build a BinderHub: [bit.ly/zero-to-binderhub-workshop](http://bit.ly/zero-to-binderhub-workshop)