Docker for Data Scientists

SDSU Data Science Symposium

Rami Krispin, February 5th, 2024

Agenda

Motivation

General Architecture

Workflow

The Dockerfile

Build

Run

Docker Compose

VScode?

https://ramikrispin.github.io/sdsu-docker-workshop/

Motivation

Reproducibility & Production

Seamless Reproducibility & Production

The Reproducibility Problem

set.seed(1234)

Error

Prof.

R 3.5.0
dplyr 0.8.5
tidyr 0.8.0

X = 5.297

X = 5.298

X = 5.3

Student 1

R 3.6.2
dplyr 1.0.0
tidyr 1.0.0

Student 2

R 4.0.0
dplyr dev
tidyr 1.1.2

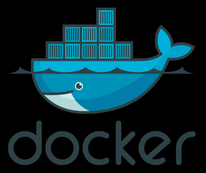
Student 3

R 4.0.2
dplyr 1.0.2
tidyr 0.8.2

The Reproducibility Problem

set.seed(1234)

Error



R 3.6.2
dplyr 1.0.0
tidyr 1.0.0

Prof.
R 3.5.0
dplyr 0.8.5
tidyr 0.8.0

X = 5.297

X = 5.298

X = 5.3

Student 1
R 3.6.2
dplyr 1.0.0
tidyr 1.0.0

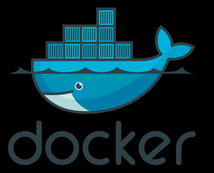
Student 2
R 4.0.0
dplyr dev
tidyr 1.1.2

Student 3
R 4.0.2
dplyr 1.0.2
tidyr 0.8.2

The Reproducibility Problem

set.seed(1234)

X = 5.297



R 3.6.2
dplyr 1.0.0
tidyr 1.0.0

Prof.
R 3.5.0
dplyr 0.8.5
tidyr 0.8.0

X = 5.297

X = 5.297

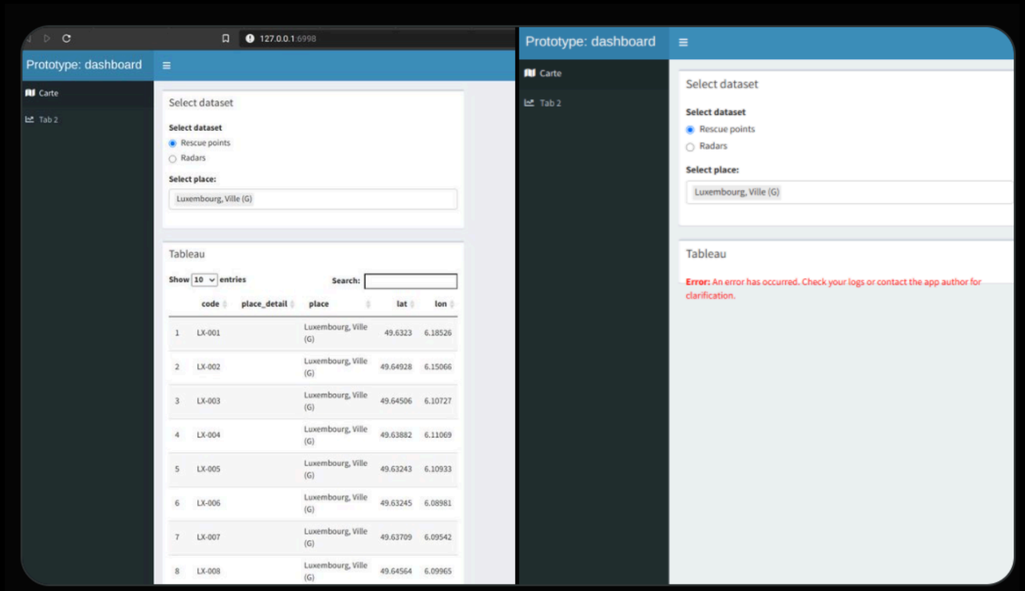
X = 5.297

Student 1
R 3.6.2
dplyr 1.0.0
tidyr 1.0.0

Student 2
R 4.0.0
dplyr dev
tidyr 1.1.2

Student 3
R 4.0.2
dplyr 1.0.2
tidyr 0.8.2

The Production Problem



Shiny app runs locally but fails when deployed due to differing locale

Asked 9 months ago Active 9 months ago Viewed 37 times

0

I have a Shiny app where you can upload a file, and the data is then processed and relevant outputs appear. This is working totally fine locally. However, I've deployed it with shinyapps.io and now although the app appears fine at first, clearly something is going wrong with the data processing.

I use `data.table::fread()` for .csv files and `readxl::read_excel` for excel files. If you upload a .csv file, the app grays out, whereas if you upload an excel file, the outputs appear but basically blank as if no data is there.

The key errors in the logs are all like this:

> Warning in grepl(pattern, vector, ignore.case = ignore.case, fixed = fixed) : input string 2 is

The Overflow Blog

Podcast 276: Be question on Stac

The Overflow #4

Featured on Meta

Responding to th commitments mo

Linked

R Shiny: works locally but failed on server

Asked 2 months ago Active 1 month ago Viewed 27 times

0

My shiny dashboard works successfully on R-studio. Recently, I moved it to the AWS EC2 Ubuntu server. I deployed a test app and it works fine. However, the main shiny app doesn't work at all. It says **"The application failed to start. The application exited during initialization."**

I checked the log and it seems the app cannot recognize any variable from Global Environment which fails the app. Since my data is over 8 GB, the Shiny app would not work if I put "readRDS" inside the app.R file. When I built this app under R-studio, I always load all the files and variables to the global environment before I start my shiny app. It seems this method is not working under the Shiny server.

Is there any other method that I can let my shiny app recognize all the variables that I preloaded to the Global Environment under the shiny-server?

If no, is there any alternative way that I can make my shiny app work and avoid loading 8GB files every time I start it?

Thank you.

Shiny server does not work with my app, which is working in local #2

Open **EnricowithR** opened this issue on Dec 11, 2016 · 3 comments

EnricowithR commented on Dec 11, 2016

I have installed Shiny server on AWS Ubuntu. The default test page works both for rmarkdown and shiny server.
However, if I upload a shiny app, which is working in local, it does not work on the shiny server. I tried with index.Rmd and with index.html; in the first case the page only shows the following message:
Error: An error has occured. Check your logs or contact the app author for clarification.
However the logs, in var/log/shiny-server.log, do not report anything for that.
in the second case the page opens with an empty frame without running the shiny app.

What should I do?

The Production Problem

Wrong path in `addResourcePath`

shiny

RamiKrispin 2 2019-09-26

Hello,

I am getting the following error on a Shiny app:

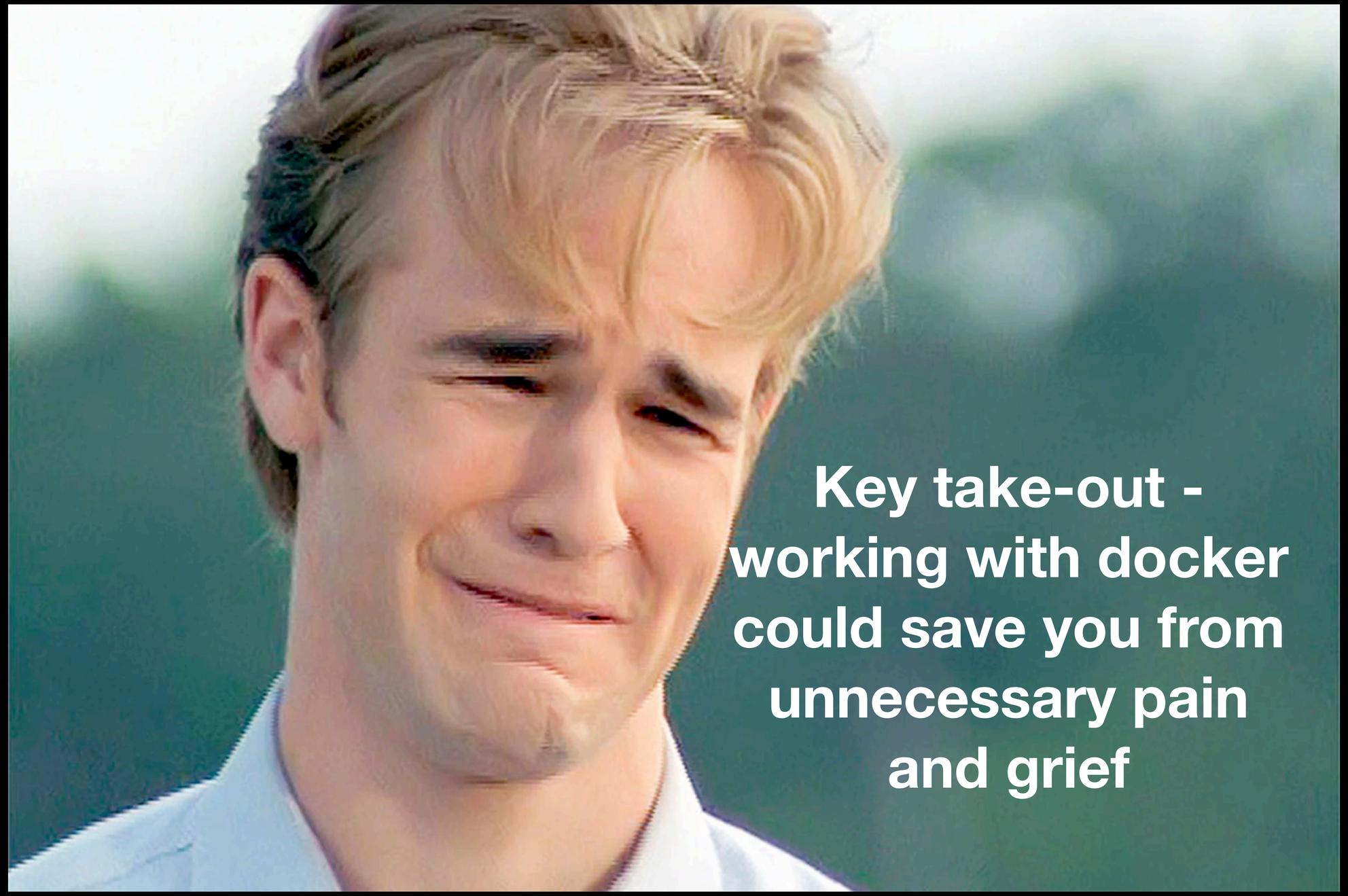
```
Warning: Error in value[[3L]]: Couldn't normalize path in `addResourcePath`,
with arguments: `prefix` = 'crosstalk-1.0.0';
`directoryPath` = '/Library/Frameworks/R.framework/Versions/3.5/Resources/li
  [No stack trace available]
```

The error is coming from the `addResourcePath` function when calling by the crosstalk package. For some reasons, it assigned wrong path (using version 3.5 instead of version 3.6). Any suggestion how can I modify the path reference?

Below is the output of the `.libPaths()' on my machine:

```
> .libPaths()
[1] "/Library/Frameworks/R.framework/Versions/3.6/Resources/library"
```

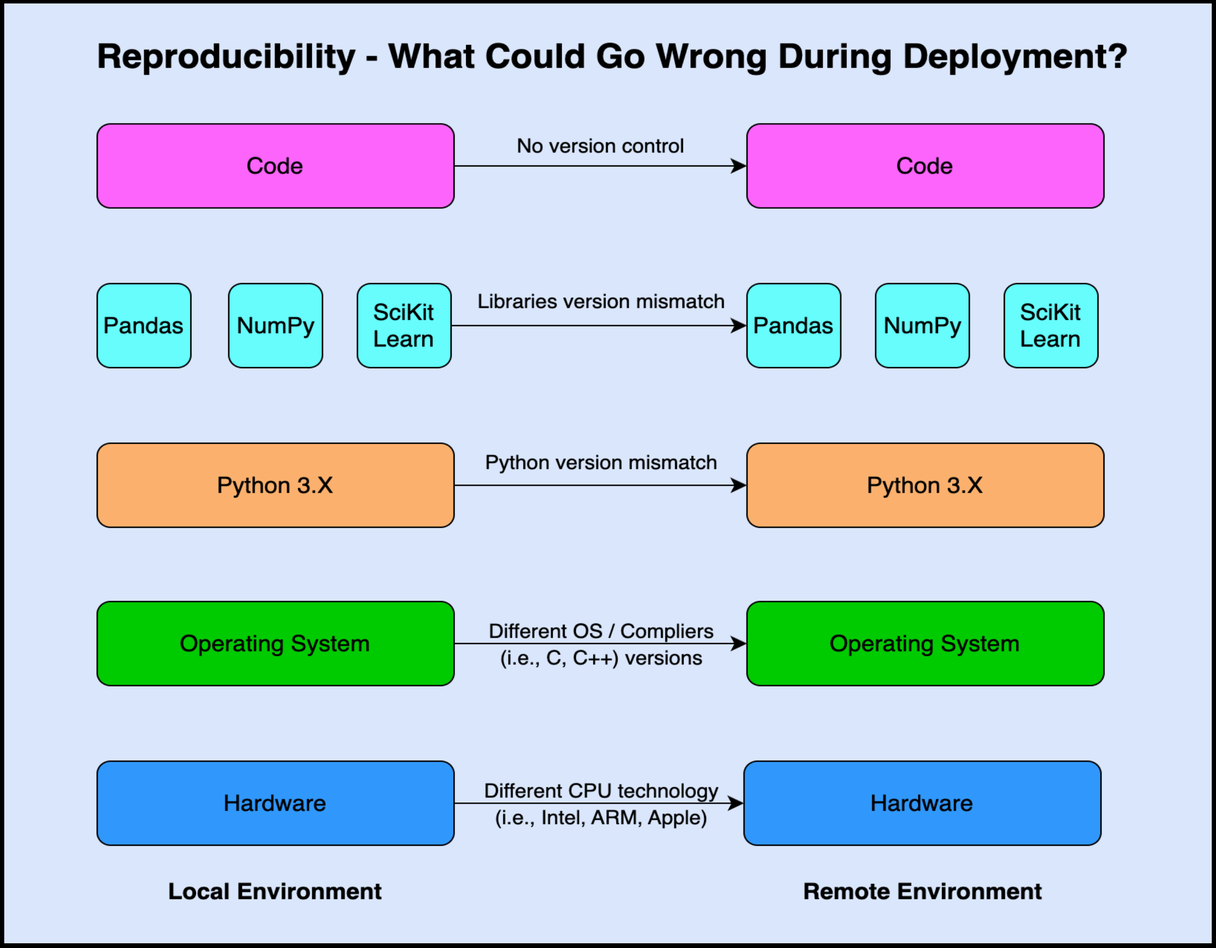
Thanks,
Rami


Key take-out -
working with docker
could save you from
unnecessary pain
and grief

Reproducibility

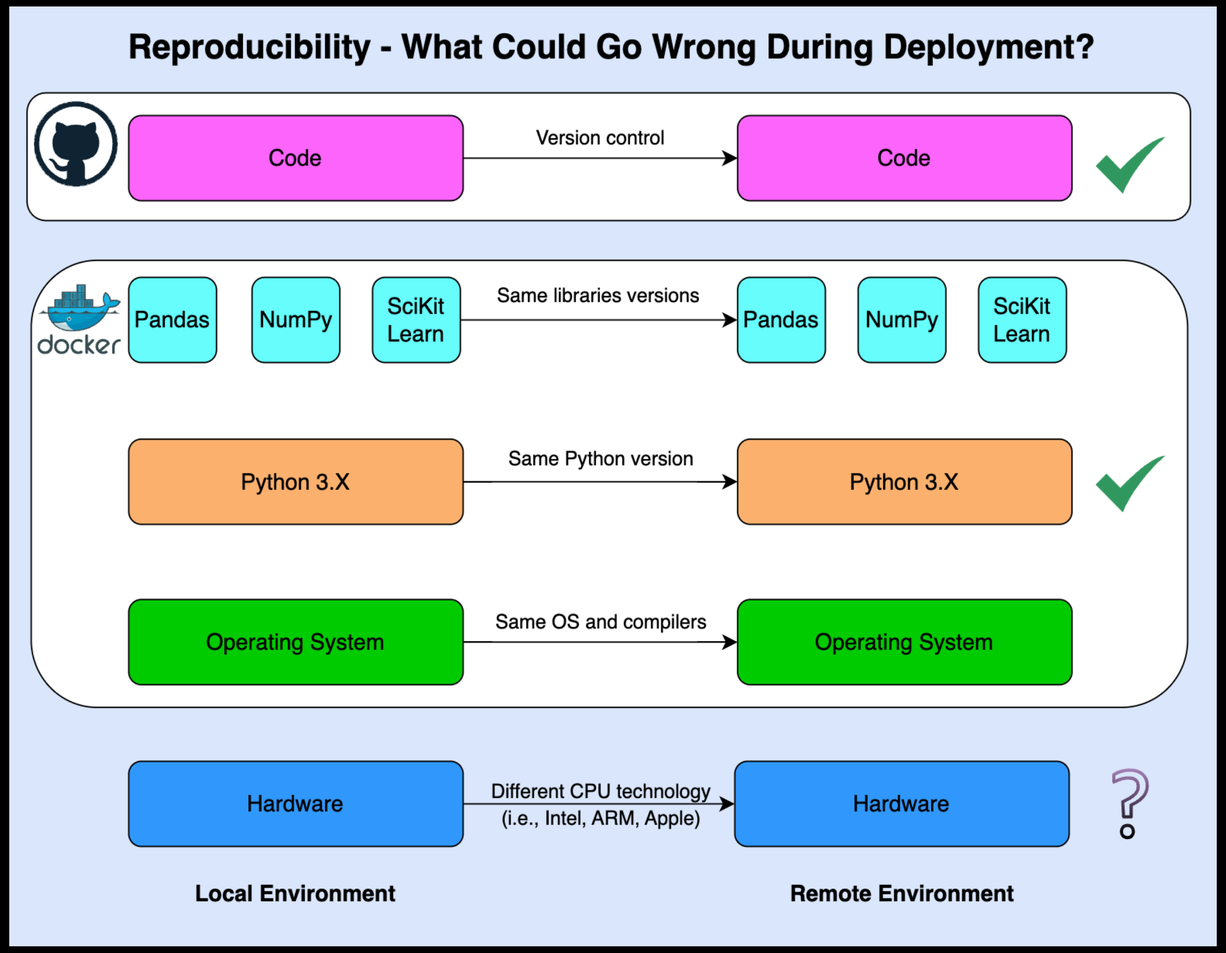
Reproducibility

What Could Go Wrong During Deployment?

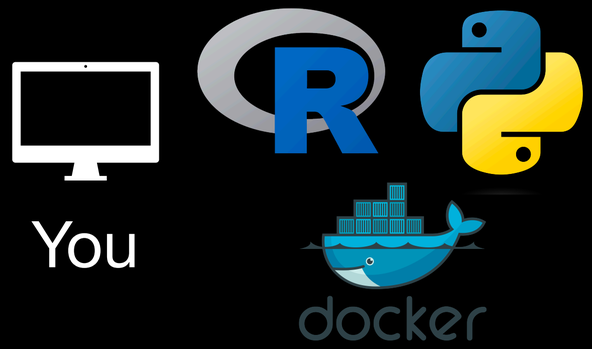


Reproducibility

What Could Go Wrong During Deployment?

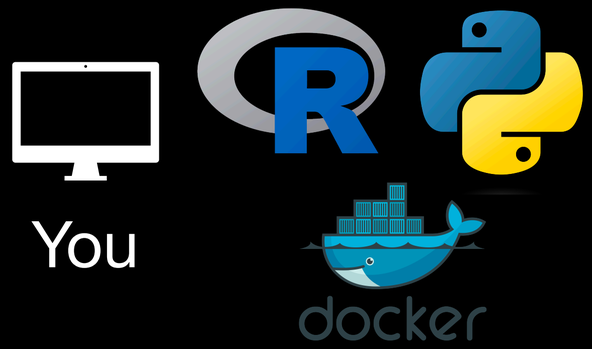


Docker in a Nutshell



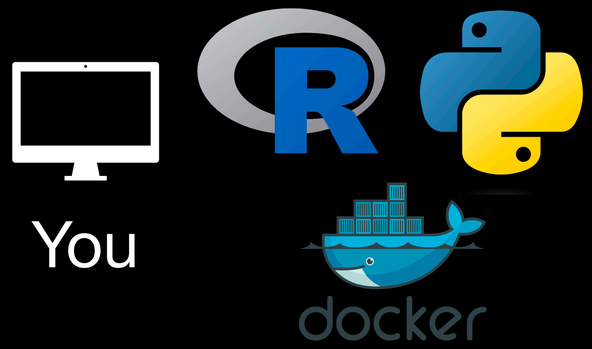

Colleague


Github Actions




Colleague


Github Actions



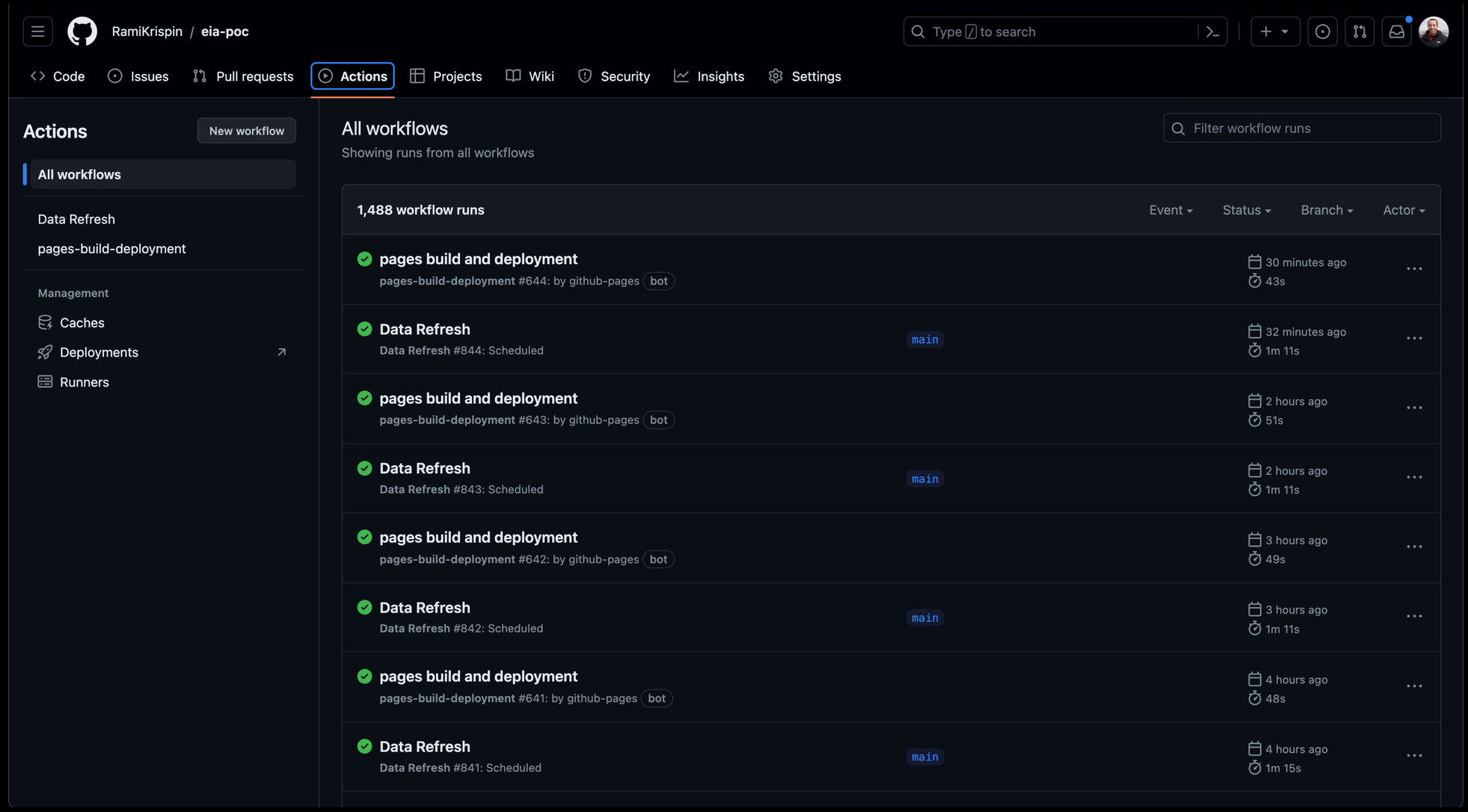

Colleague


Github Actions

Data Science Applications

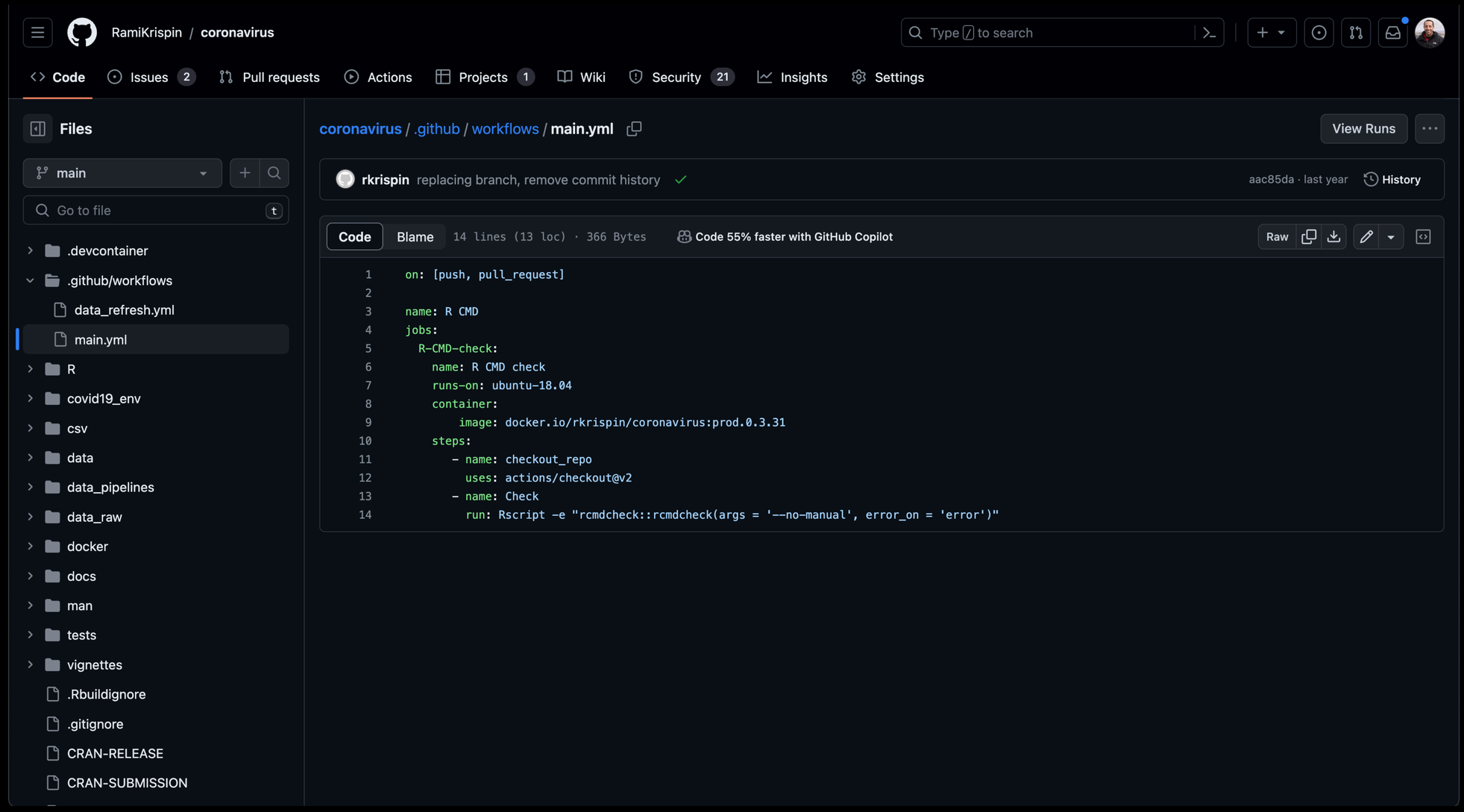
Data Science Applications

CI/CD



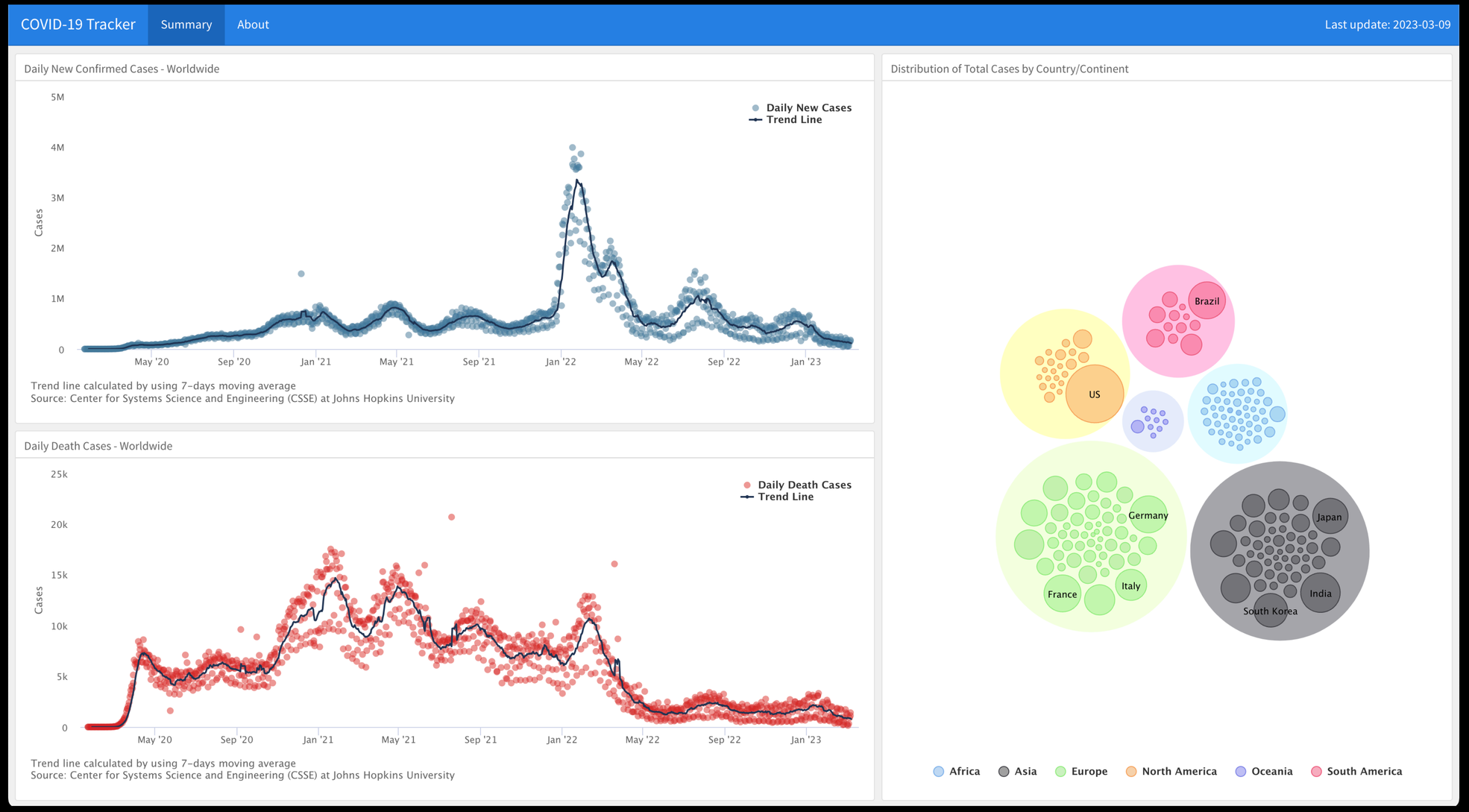
Data Science Applications

Package Development



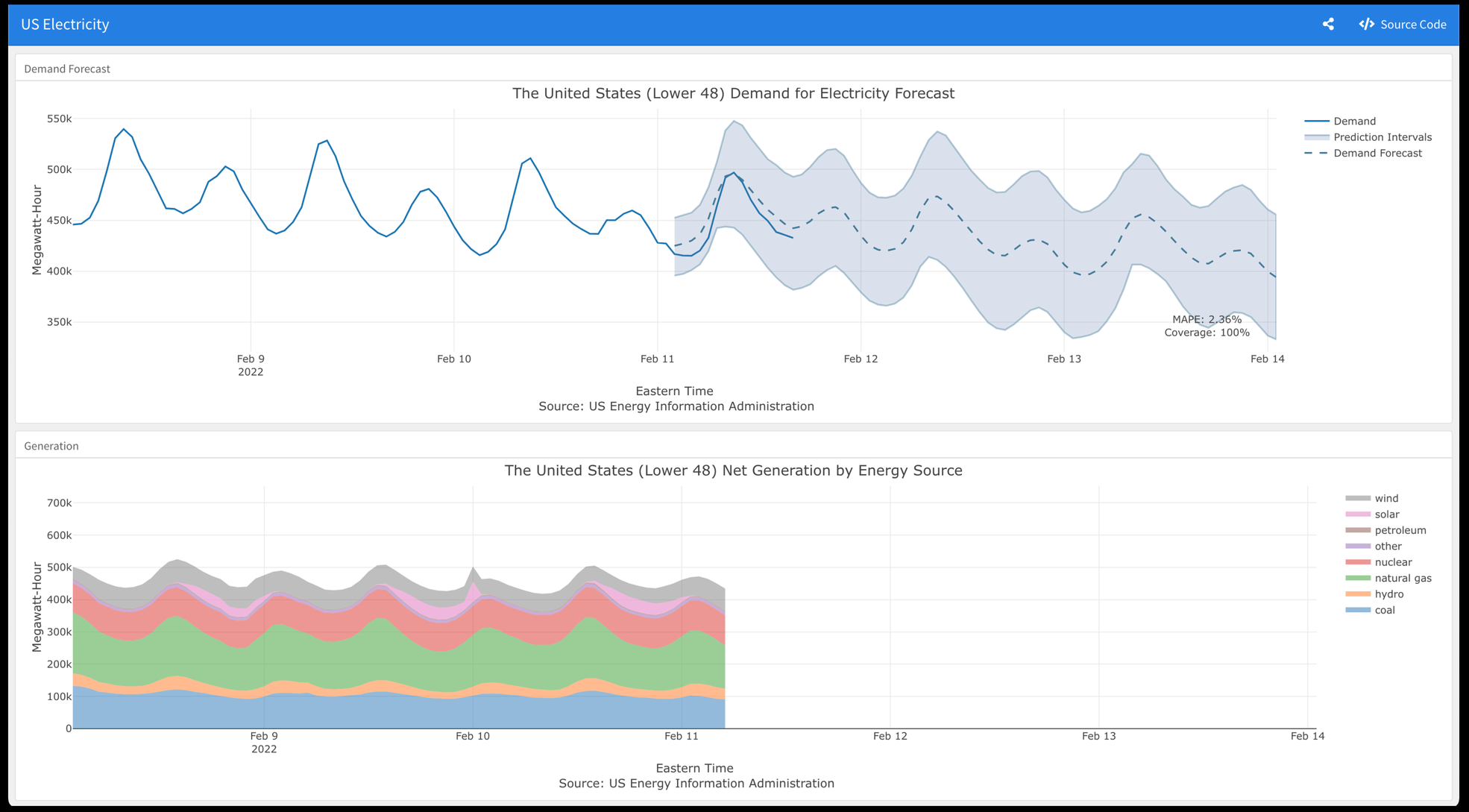
Data Science Applications

Dashboard Automation



Data Science Applications

Statistical Modeling and Machine Learning



Installing Docker Engine

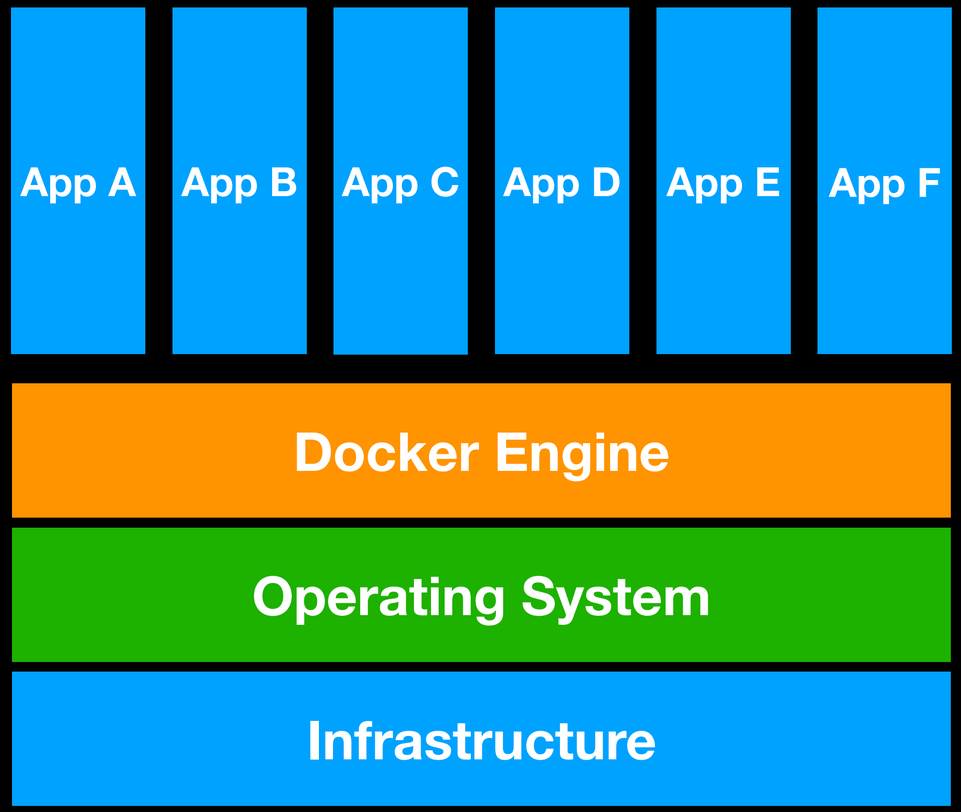
- Linux - direct installation
- Mac & Windows - Docker Desktop

https://docs.docker.com/get-docker/

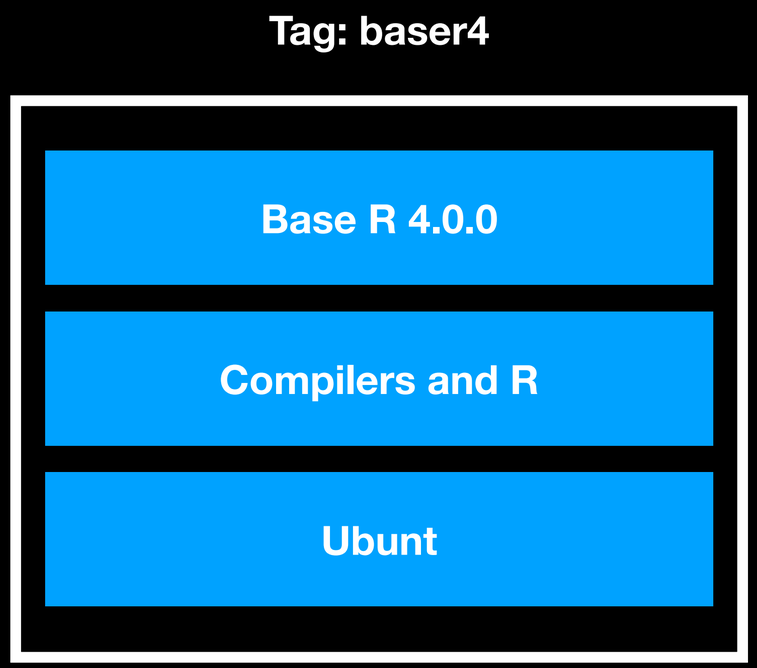
General Architecture

General Architecture

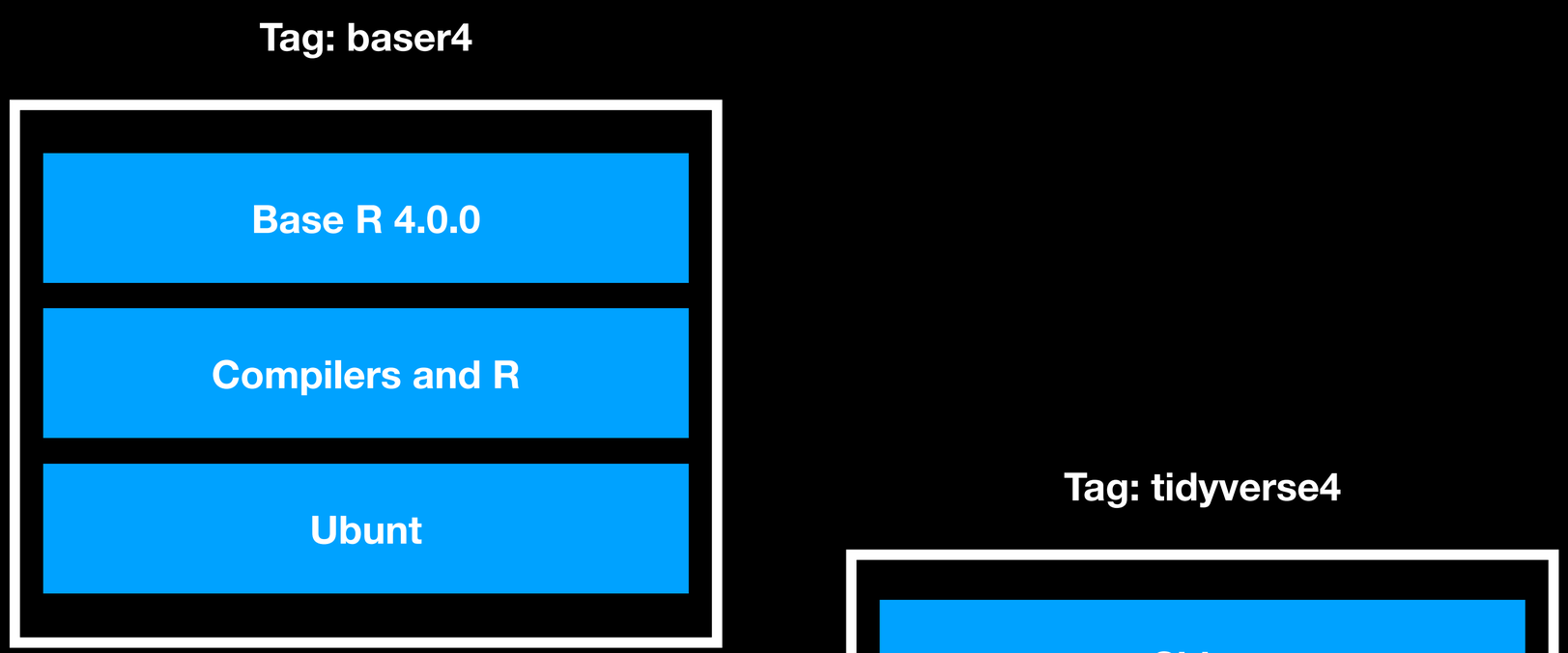
- A platform for OS-level virtualization
- Based on isolated containers
- Package different components of a software/app
- Enable seamless shipment and deployment
- Open source, free and enterprise versions available



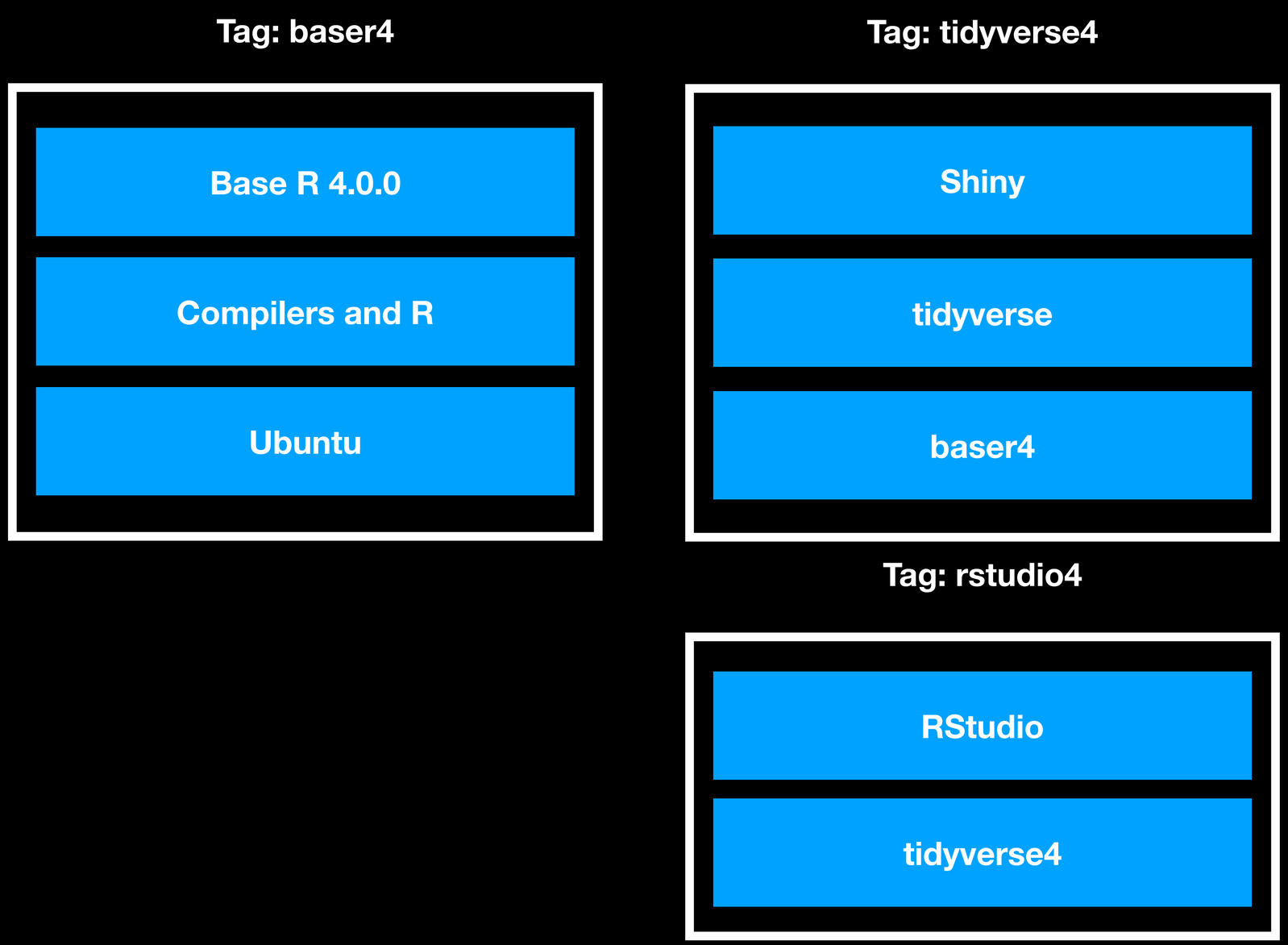
Docker Layers



Docker Layers

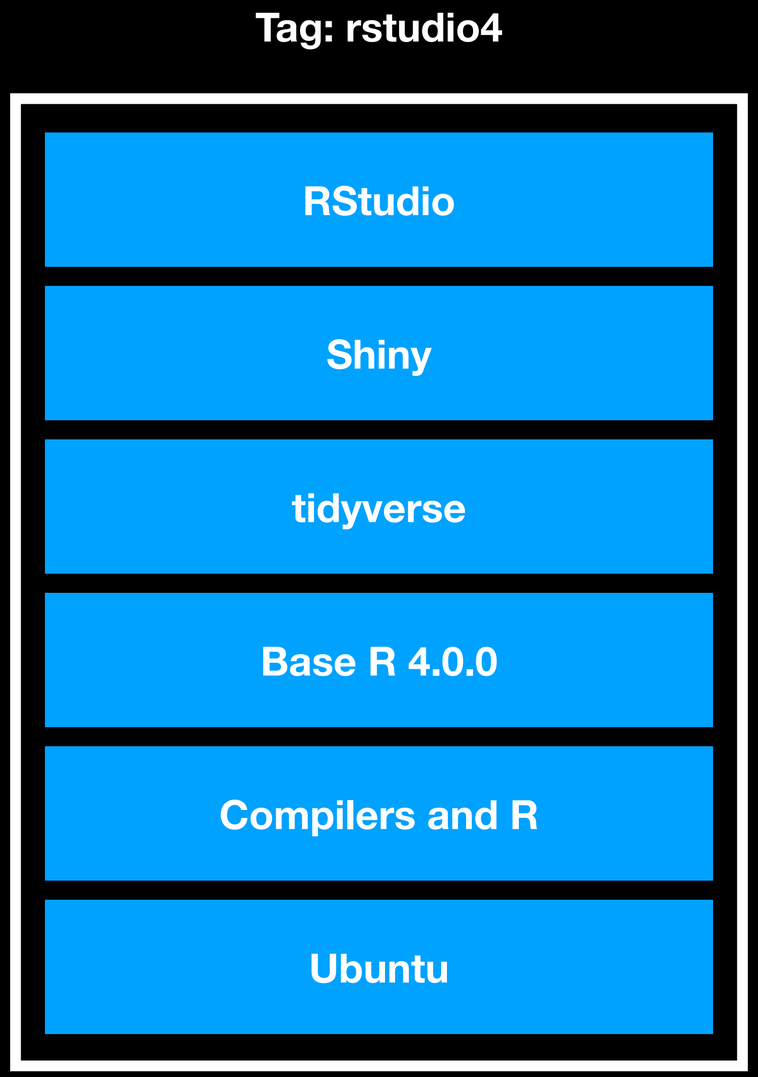


Docker Layers

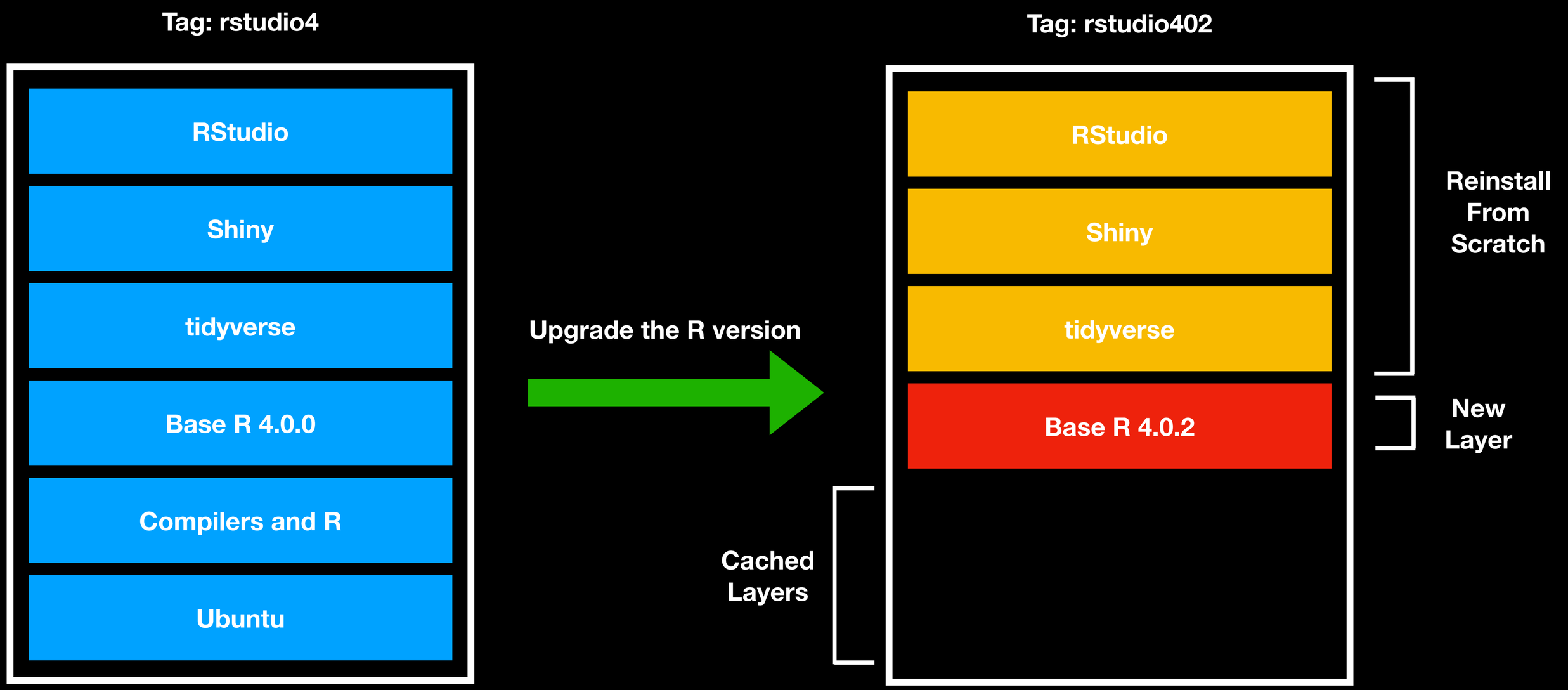


Docker Layers

Docker Layers



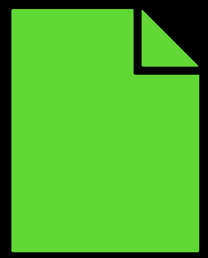
Docker Layers



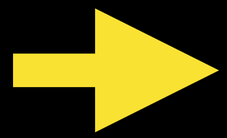
Workflow

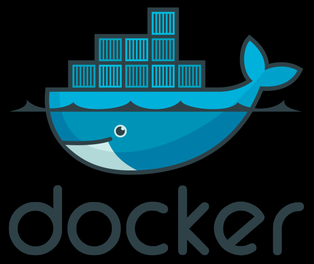
Workflow

Simple Workflow

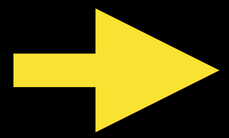


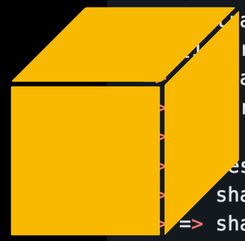
Dockerfile



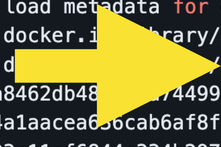


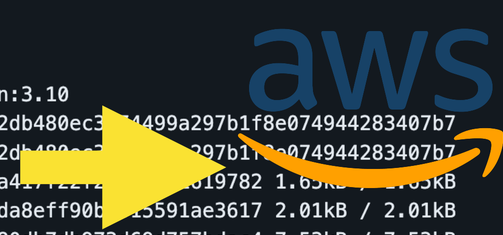
Build





Image





Container

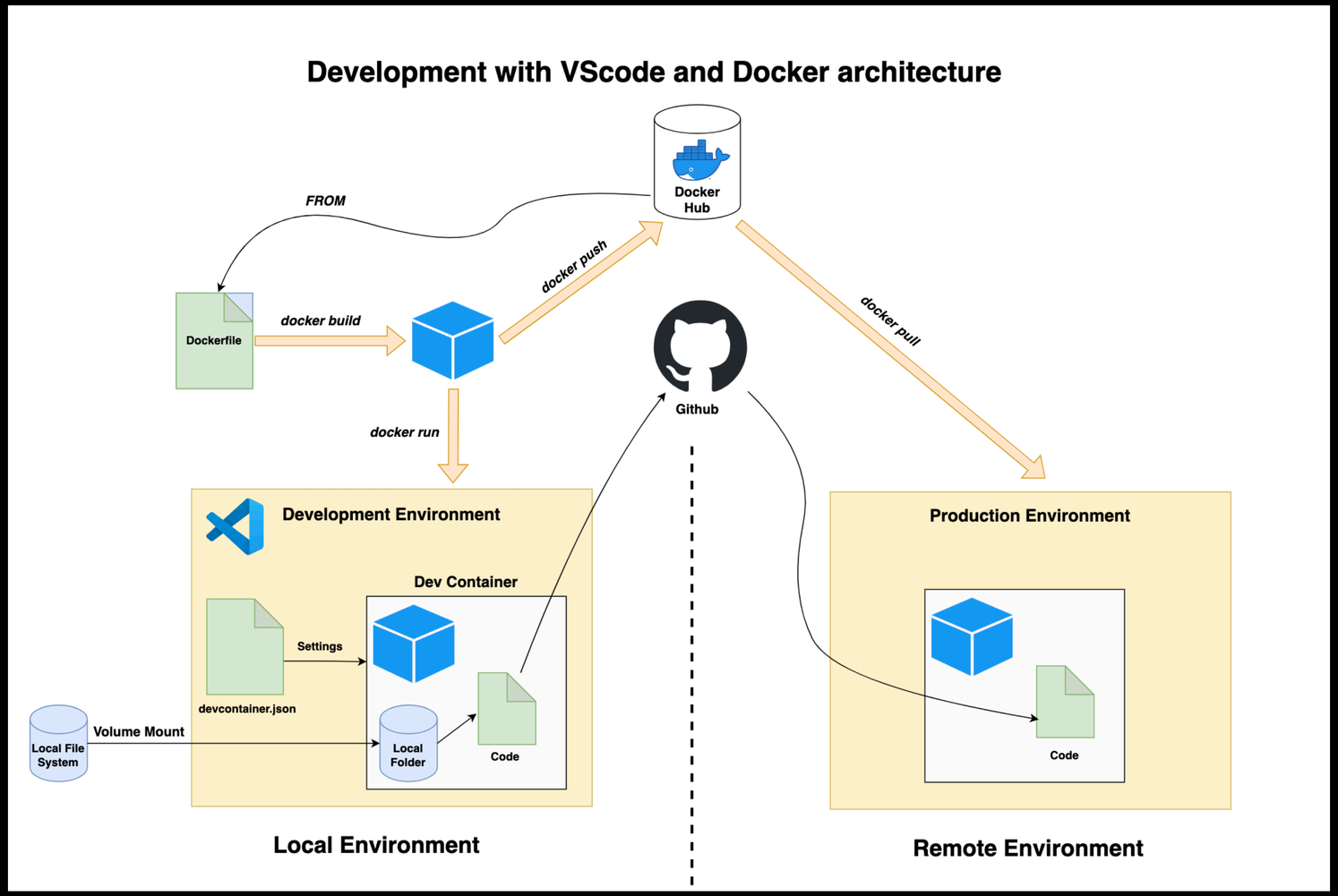
Deployment

```
[+] Building 94.2s (6/6) FINISHED
 => [internal] load build definition from Dockerfile
 => => transferring dockerfile: 162B
 => [internal] load .dockerignore
 => => transferring context: 2B
 => [internal] load metadata for docker.io/library/python:3.10
 => [1/2] FROM docker.io/library/python@sha256:a8462db480ec3...4499a297b1f8e074944283407b7
 => => resolve docker.io/library/python@sha256:a8462db480ec3...297b1f8e074944283407b7
 => => sha256:a8462db48... 7b7a4174... 1519782 1.65kB / 1.65kB
 => => sha256:4a1aacea6... 15591ae3617 2.01kB / 2.01kB
 => => sha256:23e11cf6844c334b2970fd265fb09cfe88ec250e1e80db7db973d69d757bdac4 7.53kB / 7.53kB
 => => sha256:bba7bb10d5baebcaad1d68ab3cbfd37390c646b2a688529b1d118a47991116f4 49.55MB / 49.55
 => => sha256:ec2b820b8e87758dde67c29b25d4cbf88377601a4355cc5d556a9beebc80da00 24.03MB / 24.03
 => => sha256:284f2345db055020282f6e80a646f1111fb2d5dfc6f7ee871f89bc50919a51bf 64.15MB / 64.15
 => => sha256:fea23129f080a6e28ebff8124f9dc585b412b1a358bba566802e5441d2667639 211.00MB / 211.
 => => sha256:7c62c924b8a6474ab5462996f6663e07a515fab7f3fcdd605cae690a64aa01c7 6.39MB / 6.39MB
 => => extracting sha256:bba7bb10d5baebcaad1d68ab3cbfd37390c646b2a688529b1d118a47991116f4
 => => sha256:c48db0ed1df2d2df2dccd680323097bafb5decd0b8a08f02684b1a81b339f39b 17.15MB / 17.15
 => => extracting sha256:ec2b820b8e87758dde67c29b25d4cbf88377601a4355cc5d556a9beebc80da00
 => => sha256:f614a567a40341ac461c855d309737ebccf10a342d9643e94a2cf0e5ff29b6cd 243B / 243B
 => => sha256:00c5a00c6bc24a1c23f2127a05cfddd90865628124100404f9bf56d68caf17f4 3.08MB / 3.08MB
 => => extracting sha256:284f2345db055020282f6e80a646f1111fb2d5dfc6f7ee871f89bc50919a51bf
 => => extracting sha256:fea23129f080a6e28ebff8124f9dc585b412b1a358bba566802e5441d2667639
 => => extracting sha256:7c62c924b8a6474ab5462996f6663e07a515fab7f3fcdd605cae690a64aa01c7
 => => extracting sha256:c48db0ed1df2d2df2dccd680323097bafb5decd0b8a08f02684b1a81b339f39b
 => => extracting sha256:f614a567a40341ac461c855d309737ebccf10a342d9643e94a2cf0e5ff29b6cd
 => => extracting sha256:00c5a00c6bc24a1c23f2127a05cfddd90865628124100404f9bf56d68caf17f4
 => [2/2] RUN apt-get update && apt-get install -y --no-install-recommends curl
 => exporting to image
 => => exporting layers
 => => writing image sha256:a8e4c6d06c97e9a331a10128d1ea1fa83f3a525e67c7040c2410940312e946f5
 => => naming to docker.io/rkrispin/vscode-python:ex1
```

```
1   FROM python:3.10
2
3   ARG QUARTO_VER="1.3.450"
4   ARG VENV_NAME="my_project"
5   ENV QUARTO_VER=$QUARTO_VER
6   ENV VENV_NAME=$VENV_NAME
7   RUN mkdir requirements
8   COPY install_requirements.sh install_quarto.sh install_dependencies.sh requirements/
9
10  RUN bash ./requirements/install_dependencies.sh
11  RUN bash ./requirements/install_quarto.sh $QUARTO_VER
12  COPY requirements.txt requirements/
13  RUN bash ./requirements/install_requirements.sh $VENV_NAME
```

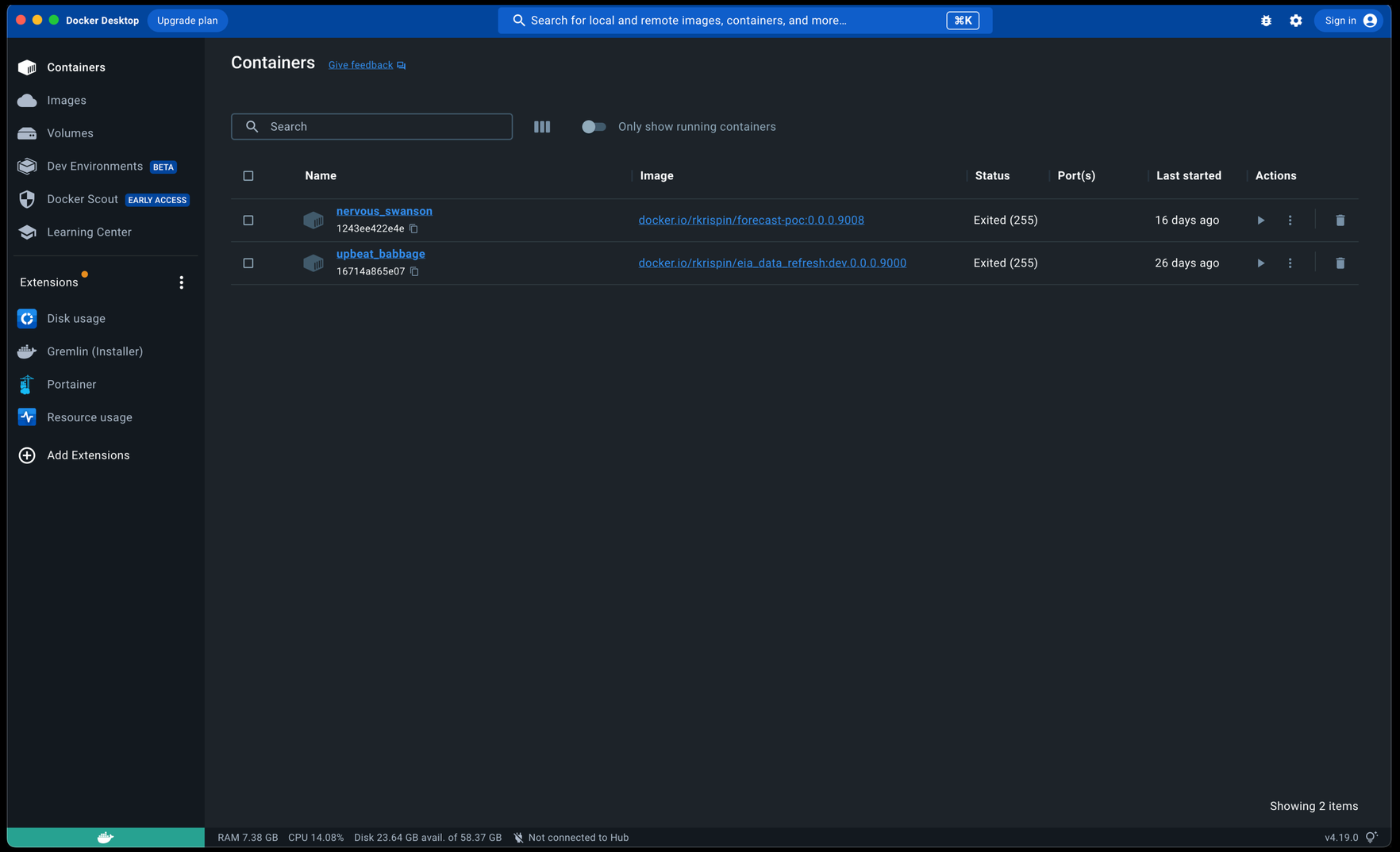
Workflow

VScode and the Dev Containers Extension

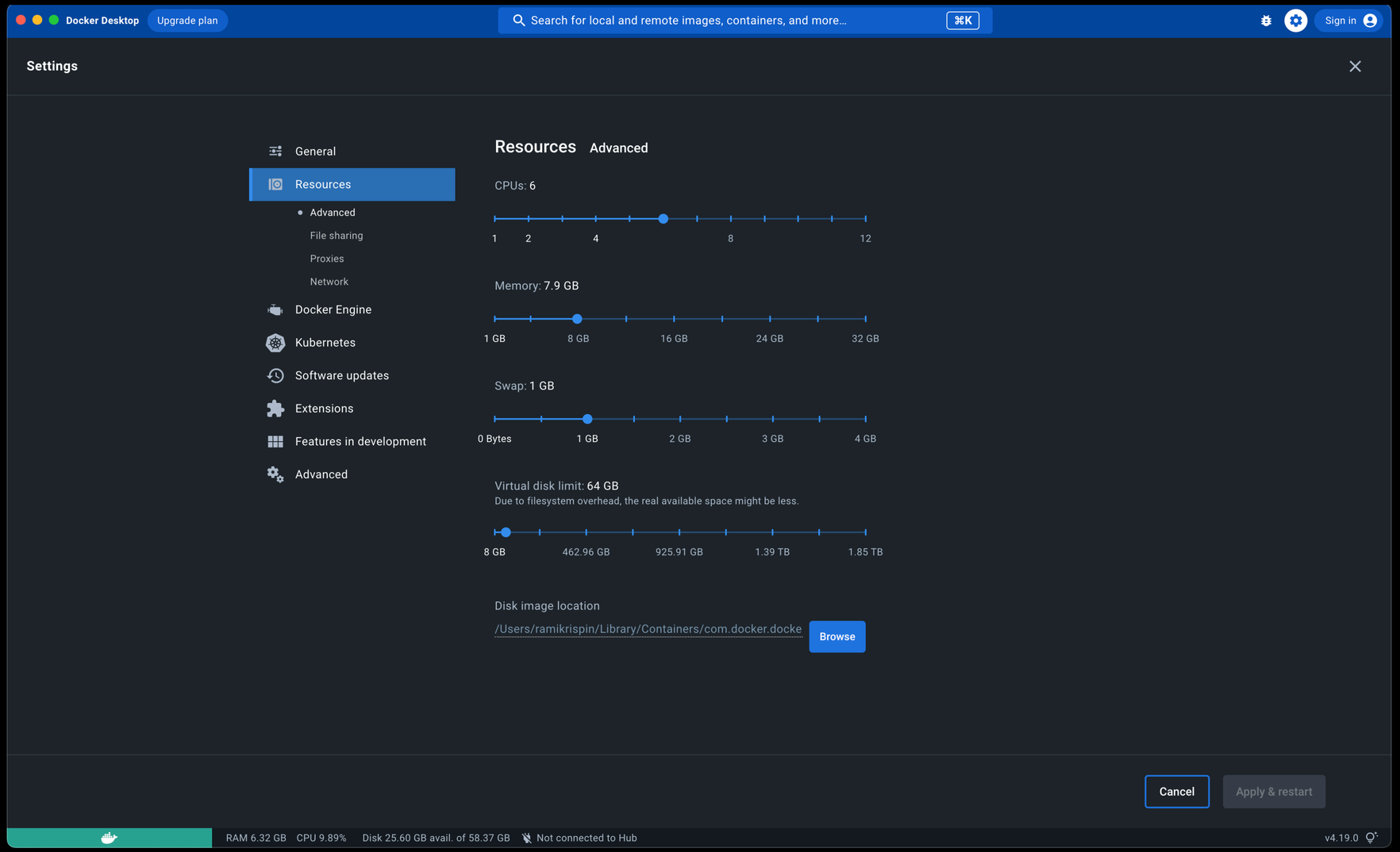


Docker Desktop

Docker Desktop



Docker Desktop



Command Line Commands

Command Line Tools

docker run --interactive --tty ubuntu

- ls - list all files and directories on the present folder
- cd - change directory folder
- mkdir - make new directory
- rm - delete file or directory
- pwd - show the current working directory
- clear - clear the terminal

Command Line Tools

docker run --interactive --tty ubuntu

- sudo - substitute user do / super user do
- apt - advanced packaging tool
- wget - web get
- curl - command-line tool for transferring data specified with URL syntax
- Rscript - calling R from the terminal
- bash - Unix shell and command language

Break?

The Dockerfile

The Dockerfile

Main Commands

- FROM
- LABEL
- RUN
- COPY
- ENV
- CMD

Build

Build

docker build [build arguments]

Build

```
docker build . -f Dockerfile -t container_name:version
```

Run

Run

docker run [options] image [command] [arguments]

Run

```
docker run --interactive --tty python:3.10
```

Other Commands

Inspect

docker inspect python:3.10

Images

docker images

PS

docker ps

Image RM

docker image rm python:3.10

System Prune

docker system prune -a

Let’s Practice!

Break?

Running RStudio inside a Docker

Running RStudio inside a Container

```
docker run --rm -ti -e PASSWORD=yourpassword -p 8787:8787 rocker/rstudio
```

Source: https://rocker-project.org/

Setting Python Env - The Hard Way

Bind Folders

Bind Folders

RStudio

```
docker run -v local_folder_path:/home/rstudio/my_scripts
 —rm -ti -e PASSWORD=yourpassword -p 8787:8787 rocker/rstudio
```

Source: https://rocker-project.org/

Bind Folders

Python

```
docker run -v local_folder_path:/my_scripts —interactive —tty python:3.10
```

Docker Compose

Bind Folders

RStudio

docker run -v local_folder_path:/home/rstudio/my_scripts —rm -ti -e PASSWORD=yourpassword -p 8787:8787 rocker/rstudio

```yaml
version: "3.9"
services:
  rstudio:
    image: "rocker/rstudio"
    ports:
      - "8787:8787"
    volumes:
      - type: "bind"
        source: "."
        target: "/home/rstudio"
      - type: "bind"
        source: "$RSTUDIO_CONFIG_PATH"
        target: "/home/rstudio/.config/rstudio"
    environment:
      - PASSWORD=yourpassword
```

Questions?