

A Shiny App Deployment Architecture for Enterprise: {renv} + Docker + ShinyProxy

Qinghua Liao

Fosun Pharma, the Representor or the Provider will not warrant the accuracy, the completeness and the timeliness of all information and contents, including predictive description, contained in the PPT documents/visual materials. In the event of any mistake, omission, and inaccuracy, Fosun Pharma, the Representor or the Provider should not be held for any liabilities in this regard.

The PPT documents/visual materials will not include and should not be deemed as any investment proposals. The investor should take their own responsibilities for any determinations so come to based upon the information contained in the PPT documents/visual materials.

Fosun Pharma is entitled to all rights, including copyright, pertaining to the PPT documents/ visual materials. The characters, the designs and other related logos, like “FOSUN PHARMA” and “复星医药” , are the trade name, trademark and the logos legally owned by Fosun Pharma. Without written consent offered by Fosun Pharma, any third party should not utilize such materials and information in any manner, including reprinting.

Agenda



Problems & Solution

{renv}

Docker

ShinyProxy

Summary

Problems & Solution

- How to keep the version of dependency R package unchanged in the new environment?
 - {renv}
- How to ensure the consistency of Shiny App running environment?
 - Docker
- How does Shiny App authenticate and authorize users ?
 - ShinyProxy

Category	Description	Shiny Server Open Source	RStudio Connect	Shinyapps.io
Security & Authentication	Password protect applications		✓	✓ *
	Deploy Shiny applications behind firewalls	✓	✓	
	Controlled access via SSL and LDAP, Active Directory, Google OAuth, PAM, proxied authentication, or passwords		✓	

{renv}: reproducible environments



- **Isolated**

- {renv} gives each project its own private package library. Installing a new or updated package for one project would not break other projects

- **Portable**

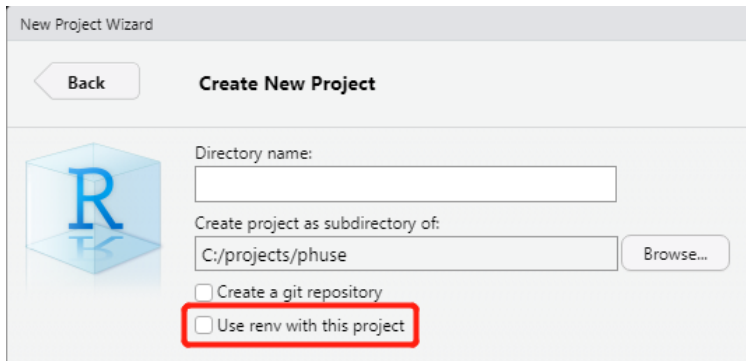
- Easily transport your projects from one computer to another, even across different platforms

- **Reproducible**

- {renv} records the exact package versions you depend on, and ensures those exact versions are the ones that get installed wherever you go.

{renv}: Setup

1



New Project Wizard

Back Create New Project

Directory name:

Create project as subdirectory of:

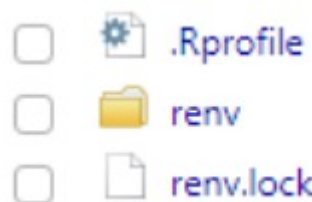
C:/projects/phuse Browse...

☐ Create a git repository

☒ Use renv with this project

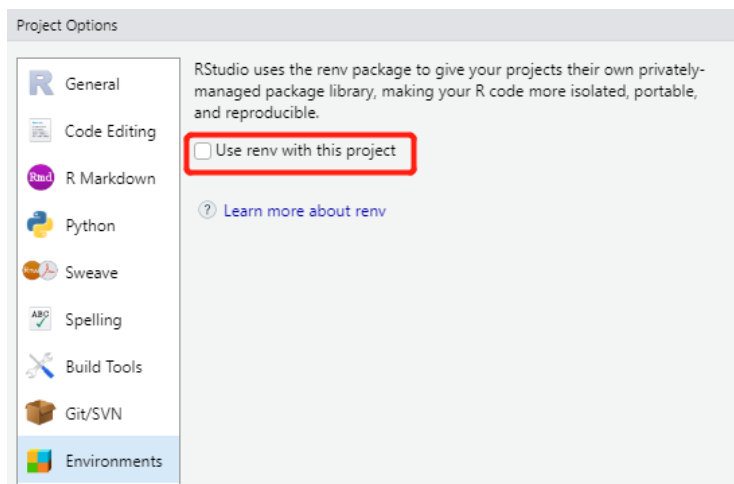
> renv::init()

```
1 {
2   "R": {
3     "Version": "4.1.3",
4     "Repositories": [
5       {
6         "Name": "CRAN",
7         "URL": "https://mirrors.tuna.tsinghua.edu.cn/CRAN"
8       }
9     ]
10  },
11  "Packages": {
12    "renv": {
13      "Package": "renv",
14      "Version": "0.15.5",
15      "Source": "Repository",
16      "Repository": "CRAN",
17      "Hash": "6a38294e7d12f5d8e656b08c5bd8ae34",
18      "Requirements": []
19    }
20  }
21 }
```



1 source("renv/activate.R")

2



Project Options

General

RStudio uses the renv package to give your projects their own privately-managed package library, making your R code more isolated, portable, and reproducible.

☒ Use renv with this project

[Learn more about renv](#)

Code Editing

R Markdown

Python

Sweave

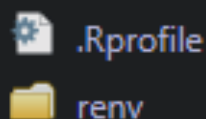
Spelling

Build Tools

Git/SVN

Environments

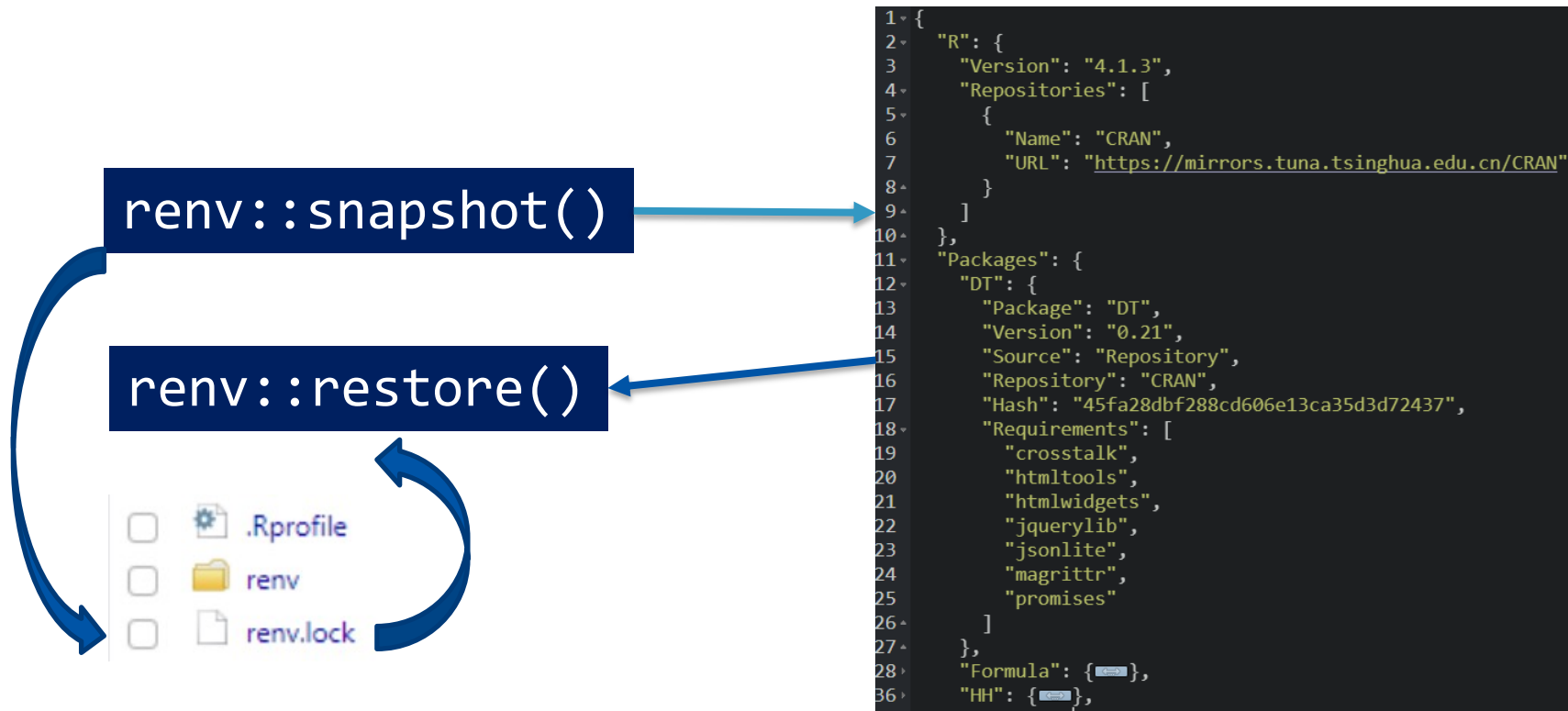
> renv::activate()



```
> .libPaths()
[1] "C:/projects/phuse/renvtest4/renv/library/R-4.1/x86_64-w64-mingw32"
[2] "C:/Program Files/R/R-4.1.3/library"
```

6

{renv}: package save/restore



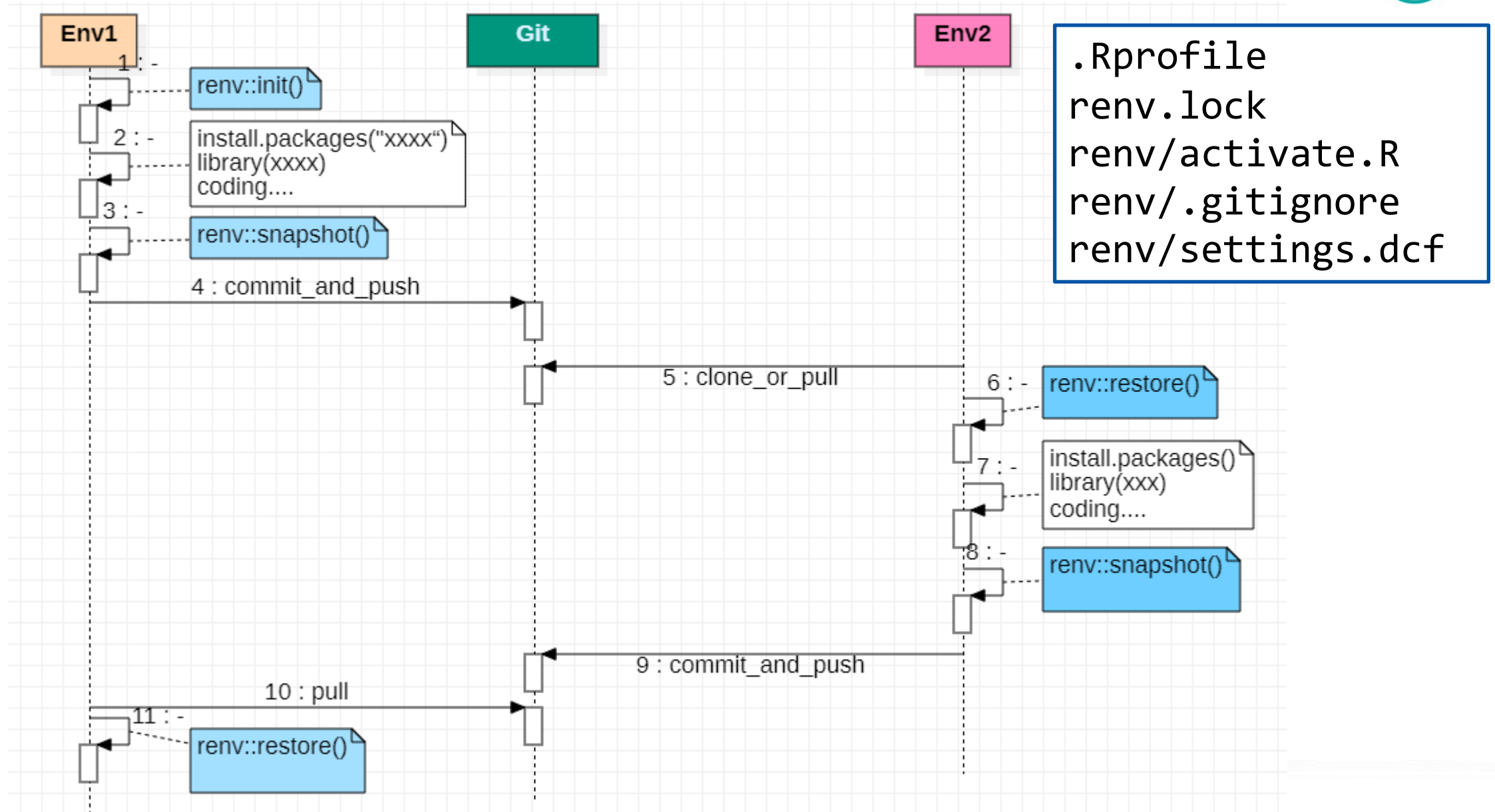
```
> .libPaths()
[1] "C:/projects/phuse/renvtest4/renv/library/R-4.1/x86_64-w64-mingw32"
[2] "C:/Program Files/R/R-4.1.3/library"
```

名称

- abind
- askpass
- backports
- base64enc
- bench
- bit
- bit64

Cache(Windows): C:\Users\<user>\AppData\Local\R\cache\R\renv

{renv}: workflow



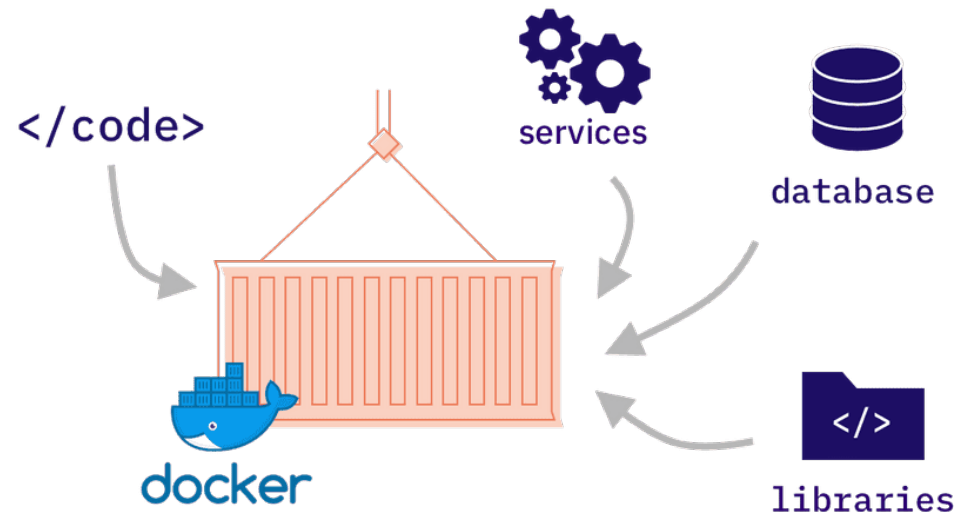
Docker: Image & Container

- **Image**

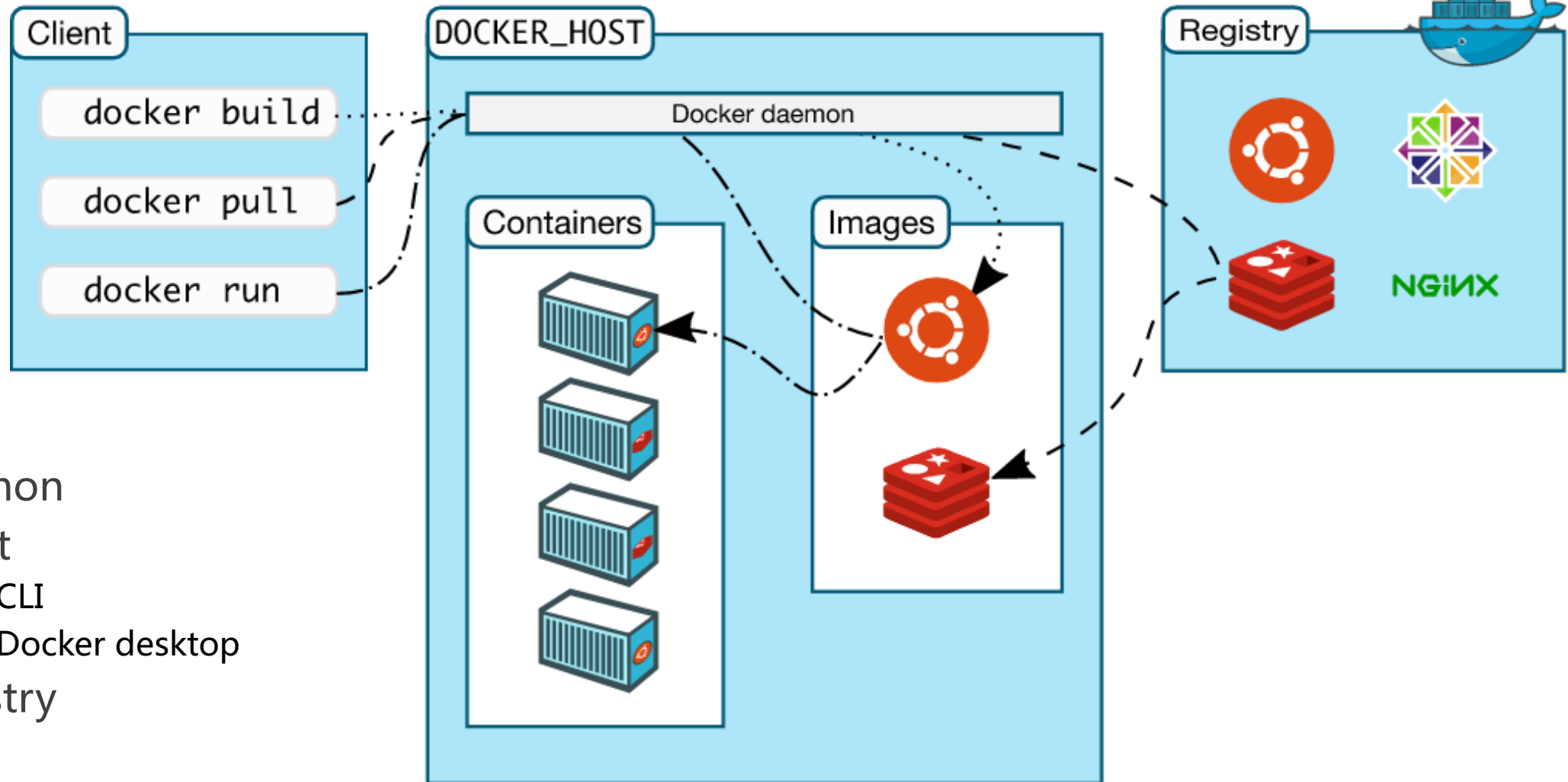
- Contain applications and all the libraries and resources they need. [Analogy: Executable File]
- Build once, run everywhere.

- **Container**

- An instance of image when it runs. [Analogy: Process]



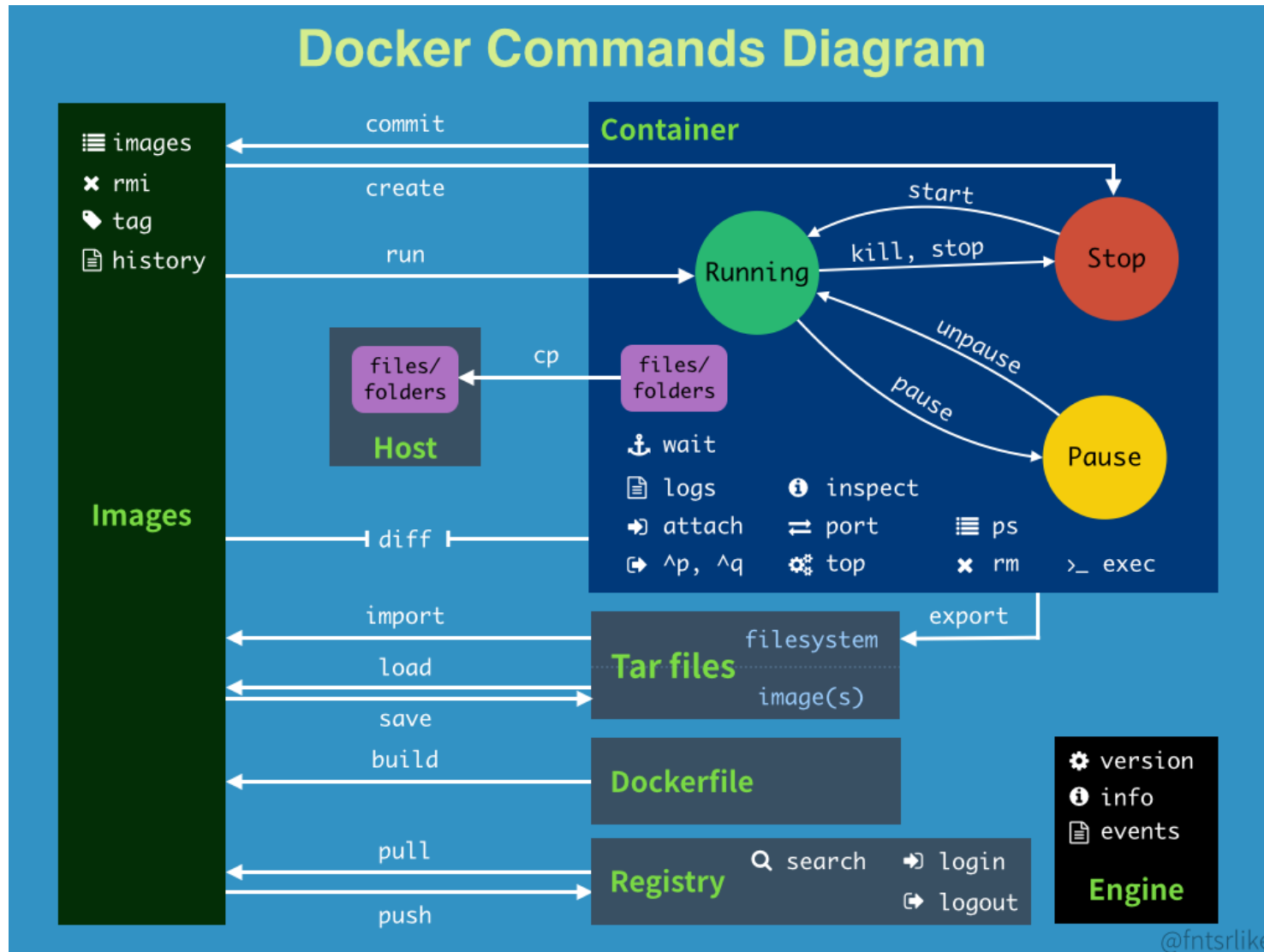
Docker: Architecture



- Daemon
- Client
 - CLI
 - Docker desktop
- Registry

Source: <https://docs.docker.com/engine/images/architecture.svg>

Docker: Commands Diagram



Docker: Dockerfile & Building Process

Dockerfile: a plain-text file, provides the specifications for creating a Docker image.

Command	Description
FROM	Specify the base image
LABEL	Add some meta information
RUN	Execute specified command
WORKDIR	Change current directory in image
COPY	Copy files/directories from host to image
EXPOSE	Specify designated port
CMD	Specify the command when container run



Docker: Shiny App Dockerfile Definition

```
4 FROM rocker/shiny:4.1.1
5 LABEL maintainer="liaoqinghua"
6
7 RUN sed -i 's/\s*\.*\s*/ubuntu/\s*\s*/mirrors.aliyun.com/ubuntu/g' /etc/apt/sources.list \
8     && apt-get update && apt-get install -y \
9     build-essential libtool cmake \
10    libicu-dev libgmp3-dev libmpfr-dev libpng-dev \
11    libxml2-dev libjpeg8-dev libpng-dev \
12    libssl-dev libsodium-dev
13
14 WORKDIR /app
15
20 COPY . .
21 # install renv & restore packages
22 RUN Rscript -e 'install.packages("renv", repos = "https://mirrors.tuna.tsinghua.edu.cn/CRAN/")' \
23     && Rscript -e 'renv::consent(provided = TRUE)' && Rscript -e 'renv::restore()'
24 # expose port
25 EXPOSE 3838
26 # run app on container start
27 CMD ["R", "-e", "shiny::runApp('/app', host = '0.0.0.0', port = 3838)"]
```

Docker: Shiny App Image Building...



```
$ docker build -t myshinyapp-demo:v0.1 .
```

Some R packages should be compiled
when build image :

- 1) Install dependent compile tools
- 2) Install dependent Linux dev components

```
#12 2529.4 ----- ANTICONF ERROR -----  
#12 2529.4 Configuration failed because libxml-2.0 was not found. Try installing:  
#12 2529.4 * deb: libxml2-dev (Debian, Ubuntu, etc)  
#12 2529.4 * rpm: libxml2-devel (Fedora, CentOS, RHEL)  
#12 2529.4 * csw: libxml2_dev (Solaris)  
#12 2529.4 If libxml-2.0 is already installed, check that 'pkg-config' is in your  
#12 2529.4 PATH and PKG_CONFIG_PATH contains a libxml-2.0.pc file. If pkg-config  
#12 2529.4 is unavailable you can set INCLUDE_DIR and LIB_DIR manually via:  
#12 2529.4 R CMD INSTALL --configure-vars='INCLUDE_DIR=... LIB_DIR=...'  
#12 2529.4 -----
```

```
[+] Building 2708.6s (11/12) 0.3s  
=> [internal] load build definition from Dockerfile 0.0s  
=> => transferring dockerfile: 966B 0.0s  
=> [internal] load .dockerignore 0.0s  
=> => transferring context: 2B 0.0s  
=> [internal] load metadata for docker.io/rocker/shiny:4.1.1 11.7s  
=> [internal] load build context 0.3s  
=> => transferring context: 72.56kB 0.3s  
=> CACHED [1/8] FROM docker.io/rocker/shiny:4.1.1@sha256:eclaf7ed446a87086ccea4c8d273ff3e90cc7335f350d2caa38b2eb 0.0s  
=> [2/8] RUN sed -i 's/\|\.*/ubuntu/|/mirrors.aliyun.com/|/ubuntu/g' /etc/apt/sources.list && apt-get u 223.1s  
=> [3/8] WORKDIR /app 0.1s  
=> [4/8] COPY . . 2.5s  
=> [5/8] COPY renv.lock ./renv.lock 0.0s  
=> [6/8] RUN Rscript -e 'install.packages("renv", repos = "https://mirrors.tuna.tsinghua.edu.cn/CRAN/")' 167.0s  
=> [7/8] RUN Rscript -e 'renv::consent(provided = TRUE)' 0.8s  
=> [8/8] RUN Rscript -e 'renv::restore()' 2303.3s  
=> => # OK [moved to cache in 0.57 milliseconds]  
=> => # Installing nloptr [2.0.0] ...  
=> => # OK [built from source]  
=> => # Moving nloptr [2.0.0] into the cache ...  
=> => # OK [moved to cache in 0.63 milliseconds]  
=> => # Installing lme4 [1.1-28] ...
```

Docker: After Building...

Image

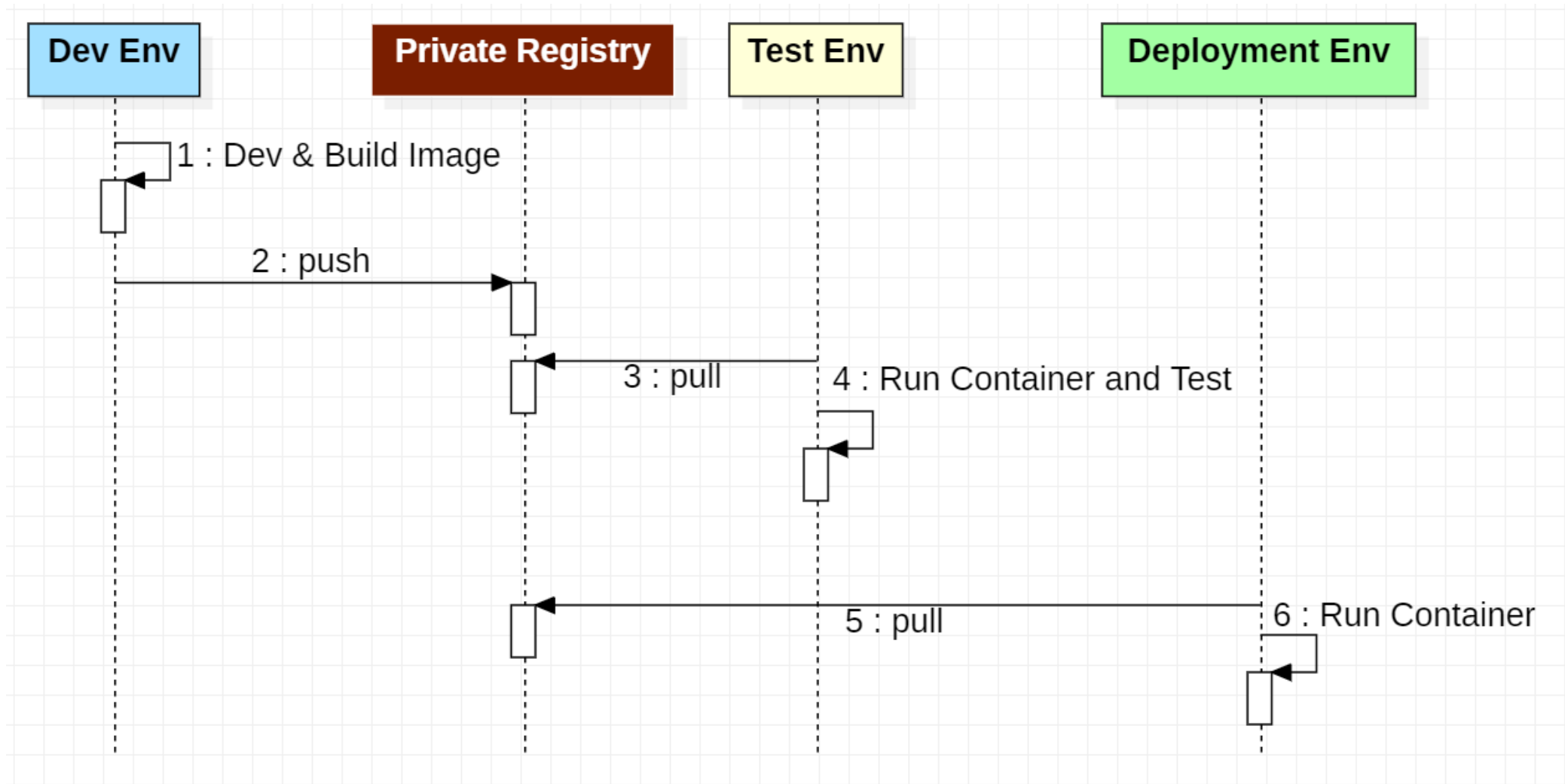
myshinyapp-demo	IN USE	v0.1	80dfe001692f	2 months ago	2.8 GB
portainer/portainer-docker-extens...	IN USE	2.14.2	ab836adaa325	3 months ago	278.38 MB

Container

		NAME	IMAGE	STATUS	PORT...	STARTED	
☐		nice_ramanuja 00c7ef948906	hzibra	Running	-	3 minutes ago	    
☐		demo1 4f41a2017584	myshi	Running	3838	13 hours ago	    

OPEN WITH BROWSER

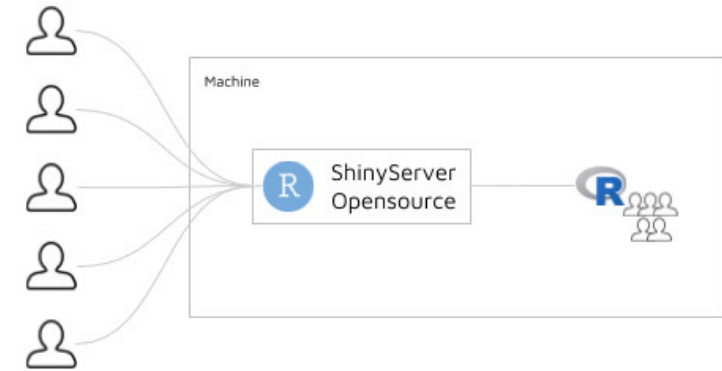
Docker: Dev/Test/Deploy Workflow



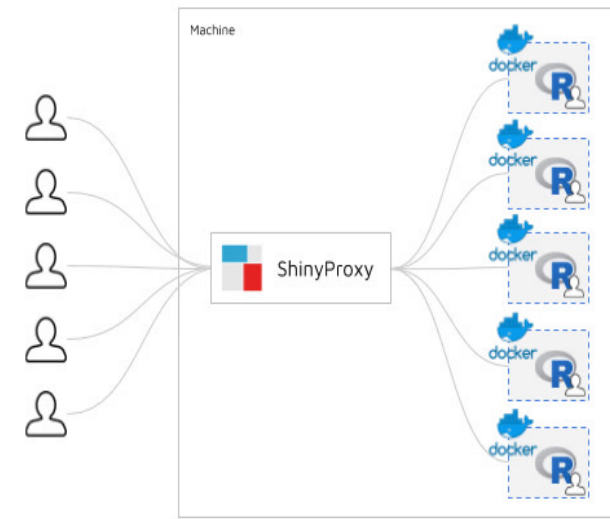
ShinyProxy



- ShinyProxy is a free and open-source platform
 - [Apache License version 2.0](#)
- Enterprise authentication/authorization
 - LDAP (the default)
 - Single-Sign On with Keycloak
 - OpenId Connect and services that implement it e.g. Auth0
 - SAML2.0
 - Simple authentication
 - No authentication
- Deploy containerized web applications
 - User sessions are fully isolated



appslab.com



appslab.com

ShinyProxy: Main Page Setting

- Logo
- Title
- Shiny App Name & Description
- Navbar

Please sign in:

User name
Password

Sign in



FOSUN PHARMA
复星医药

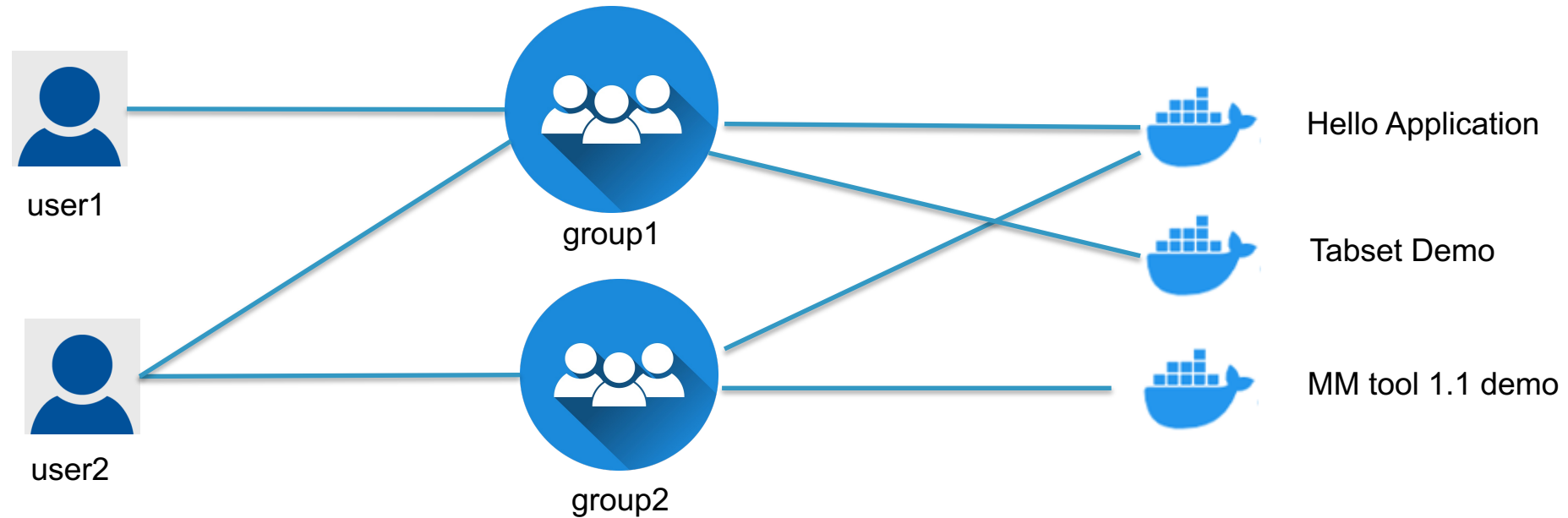
Shiny App Demo

 user2

Sign Out

- [Hello Application](#)
Application which demonstrates the basics of a Shiny app
- [Tabset Demo](#)
Tabset demonstration application
- [MM tool 1.1 demo](#)
MM tool 1.1 demo in docker

ShinyProxy: Authentication/Authorization Setting



FOSUN PHARMA
复星医药

Shiny App Demo

user1

Sign Out

- [Hello Application](#)
Application which demonstrates the basics of a Shiny app
- [Tabset Demo](#)
Tabset demonstration application

FOSUN PHARMA
复星医药

Shiny App Demo

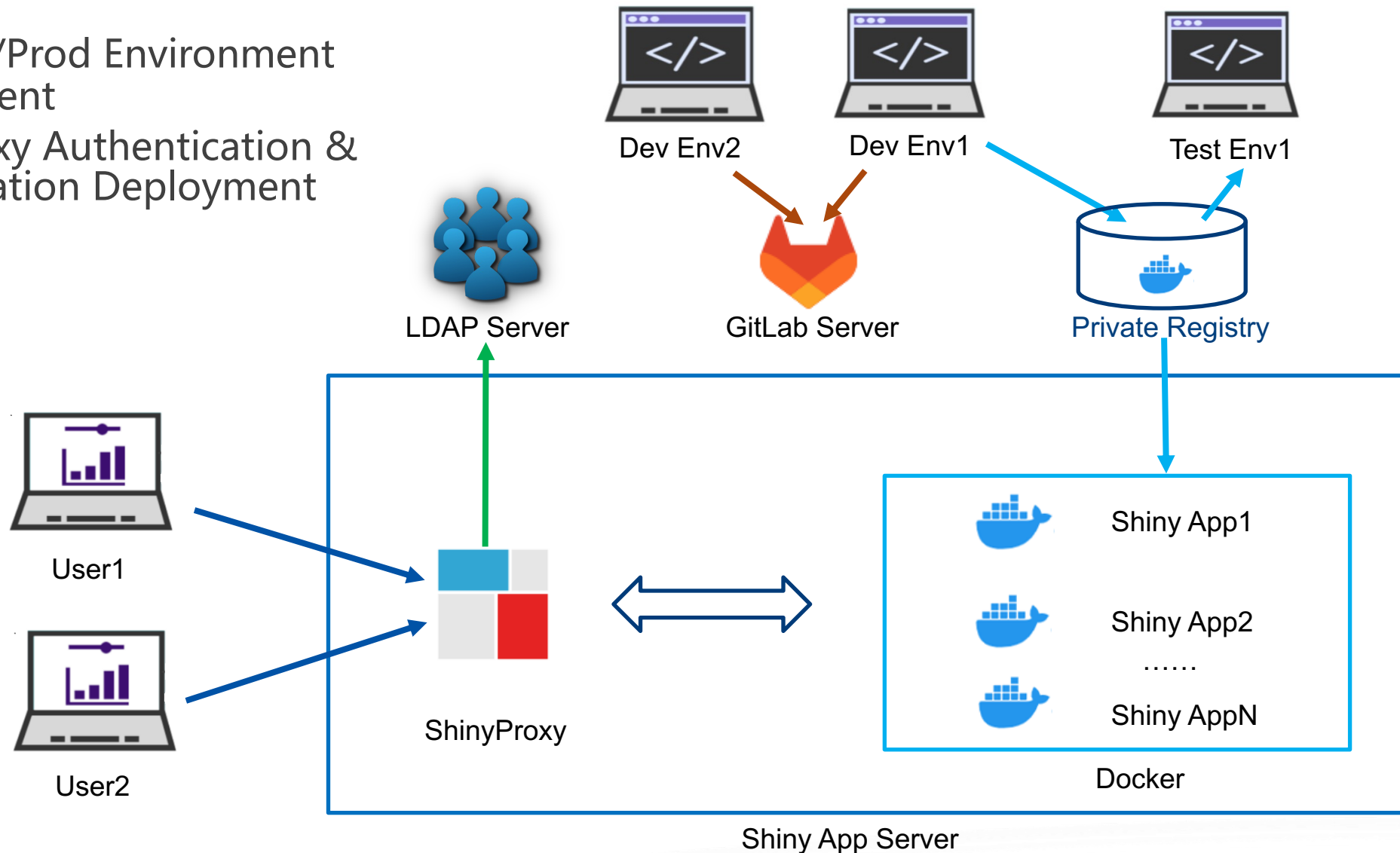
user2

Sign Out

- [Hello Application](#)
Application which demonstrates the basics of a Shiny app
- [Tabset Demo](#)
Tabset demonstration application
- [MM tool 1.1 demo](#)
MM tool 1.1 demo in docker

Summary : Deployment Architecture

- Dev/Test/Prod Environment Deployment
- ShinyProxy Authentication & Authorization Deployment



The End : Q&A

Thank You!



FOSUN PHARMA

复星医药

持续创新 乐享健康



复星医药微信公众号
www.fosunpharma.com

Docker: Which base image is available?

- From r-base image
 - FROM r-base:4.1.1
 - FROM rstudio/r-base:4.1.1-focal
- From R Shiny image
 - FROM rocket/shiny:4.1.1