



Best practices for Shiny apps with Docker

by Peter Solymos
Analythium.io
@psolymos

The smartest abstraction ever

shiny: Web Application Framework for R

Makes it incredibly easy to build interactive web applications with R. Automatic "reactive" binding between inputs and outputs and extensive prebuilt widgets make it possible to build beautiful, responsive, and powerful applications with minimal effort.

Version:

1.7.1

Depends:

R (≥ 3.0.2), methods

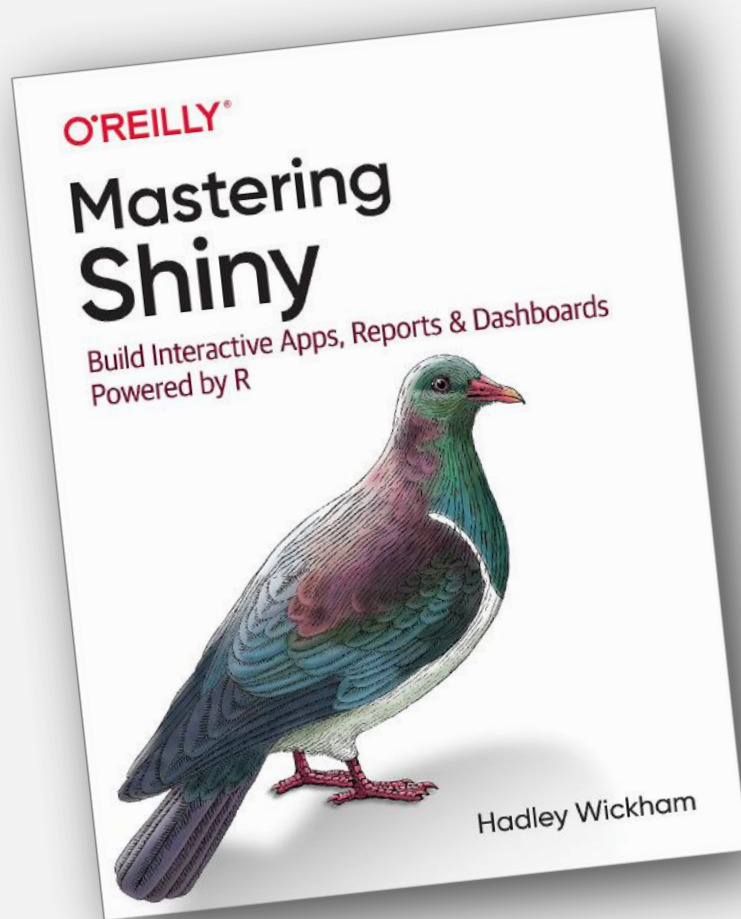
Imports:

utils, grDevices, httpuv (≥ 1.5.2), mime (≥ 0.3), jsonlite (≥ 0.9.16),
xtable, fontawesome (≥ 0.2.1), htmltools (≥ 0.5.2), R6 (≥ 2.0),
sourcetools, later (≥ 1.0.0), promises (≥ 1.1.0), tools, crayon, rlang (≥
0.4.10), fastmap (≥ 1.1.0), withr, commonmark (≥ 1.7), glue (≥ 1.3.2),
bslib (≥ 0.3.0), cachem, ellipsis, lifecycle (≥ 0.2.0)
datasets, Cairo (≥ 1.5-5), testthat (≥ 3.0.0), knitr (≥ 1.6), markdown,
rmarkdown, ggplot2, reactlog (≥ 1.0.0), magrittr, shinytest (≥
1.4.0.9003), yaml, future, dygraphs, ragg, showtext, sass

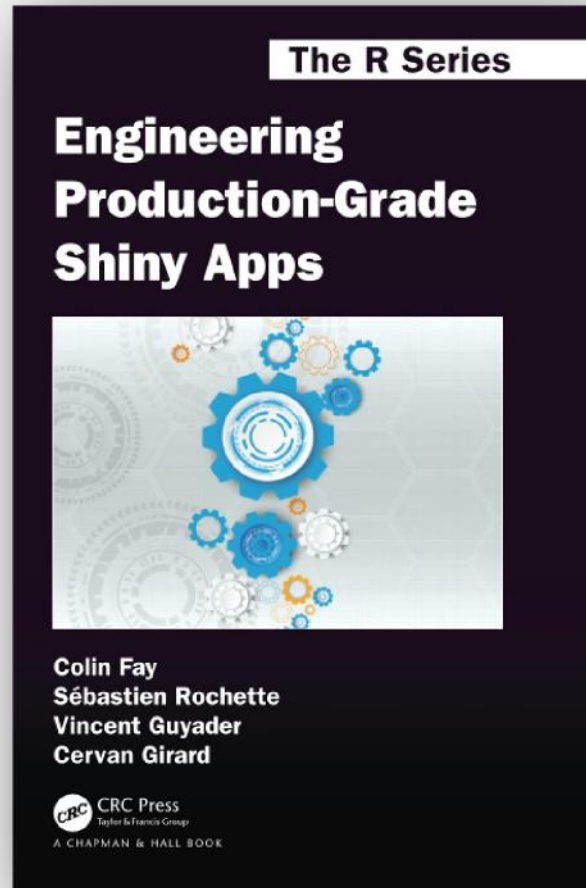
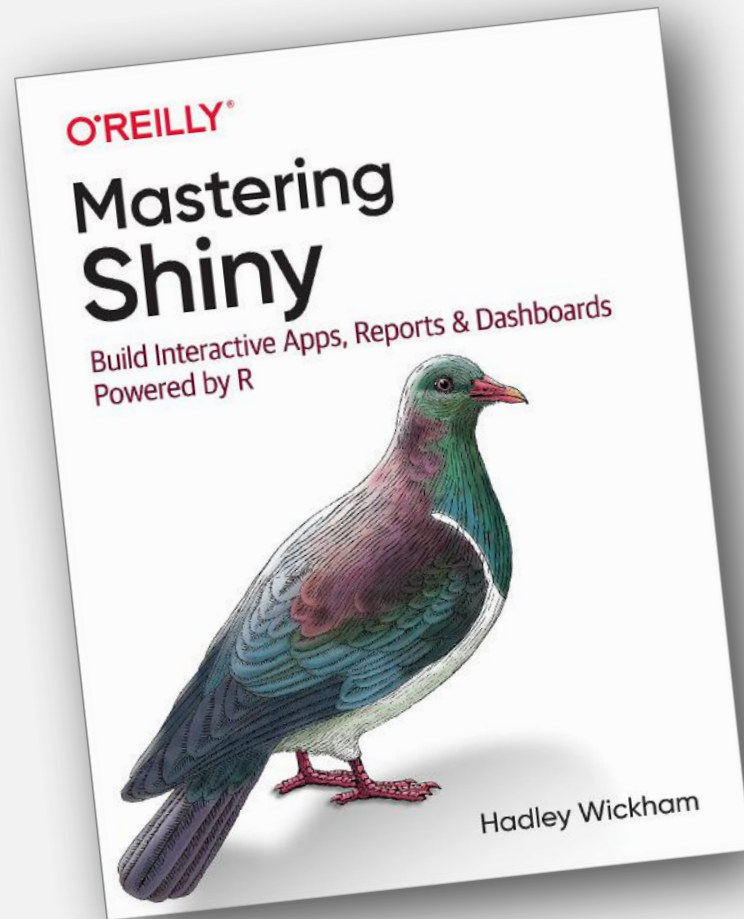
Suggests:



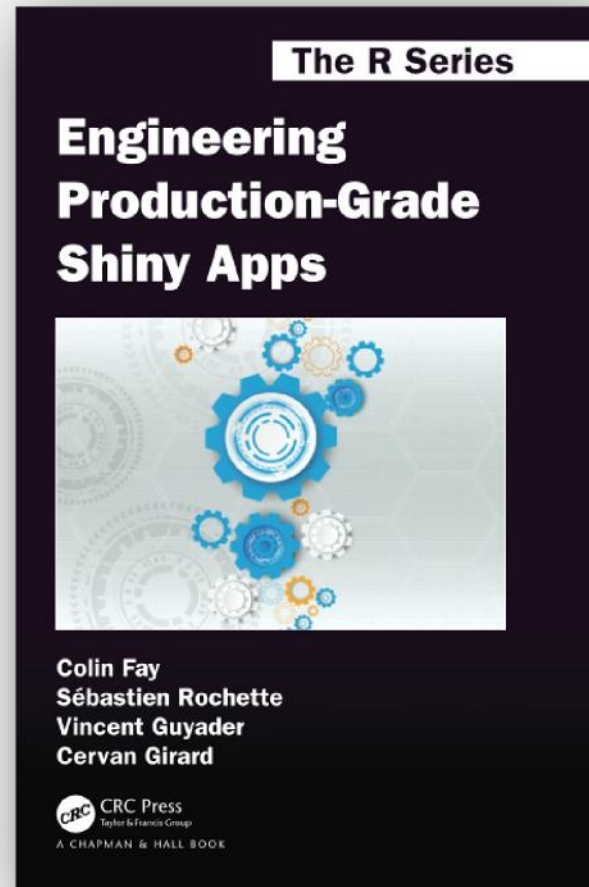
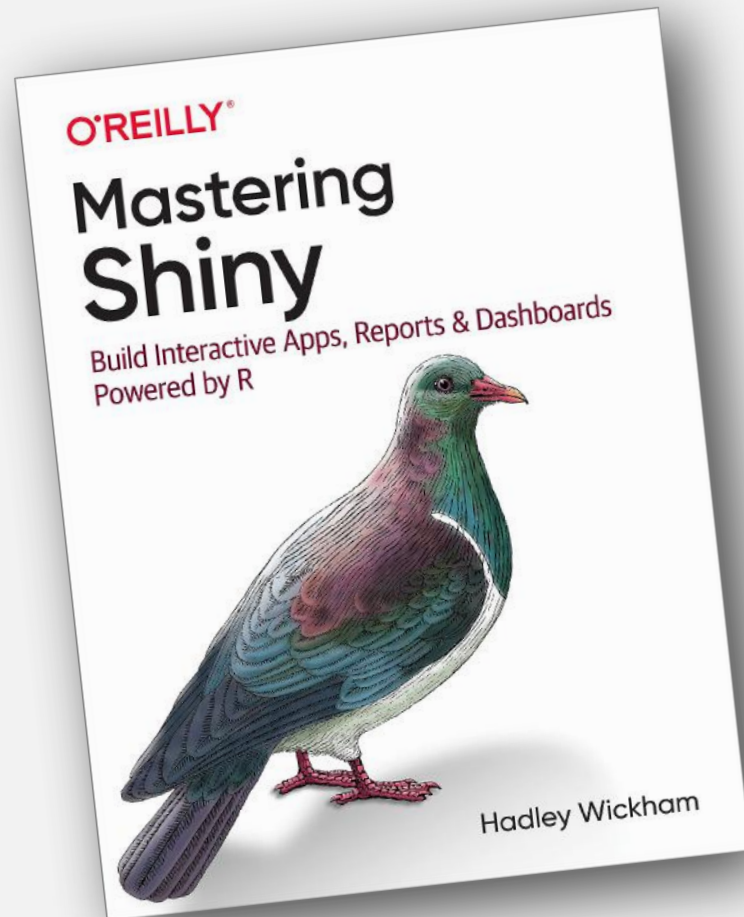
So you mastered Shiny



Your app is production-grade



And your user interface is outstanding



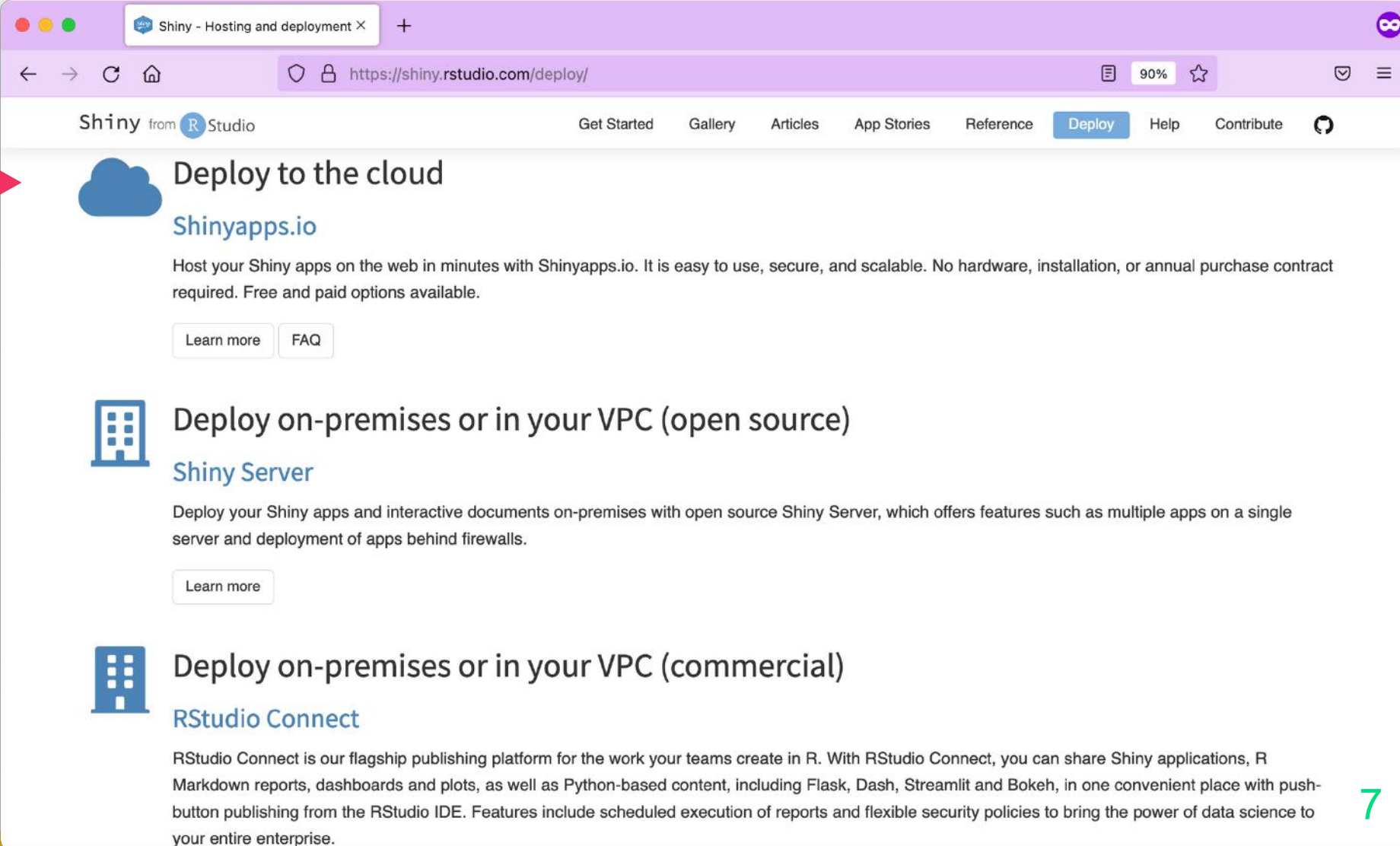
Congratulations!
Now you just need to
host your app ...



Conventional hosting options

Easy & free

Limited uptime
& no HTTPS with
custom domains



The screenshot shows a web browser window with the URL `https://shiny.rstudio.com/deploy/`. The page title is "Shiny - Hosting and deployment". The navigation bar includes links for "Get Started", "Gallery", "Articles", "App Stories", "Reference", "Deploy" (highlighted), "Help", and "Contribute". The main content area lists three hosting options:

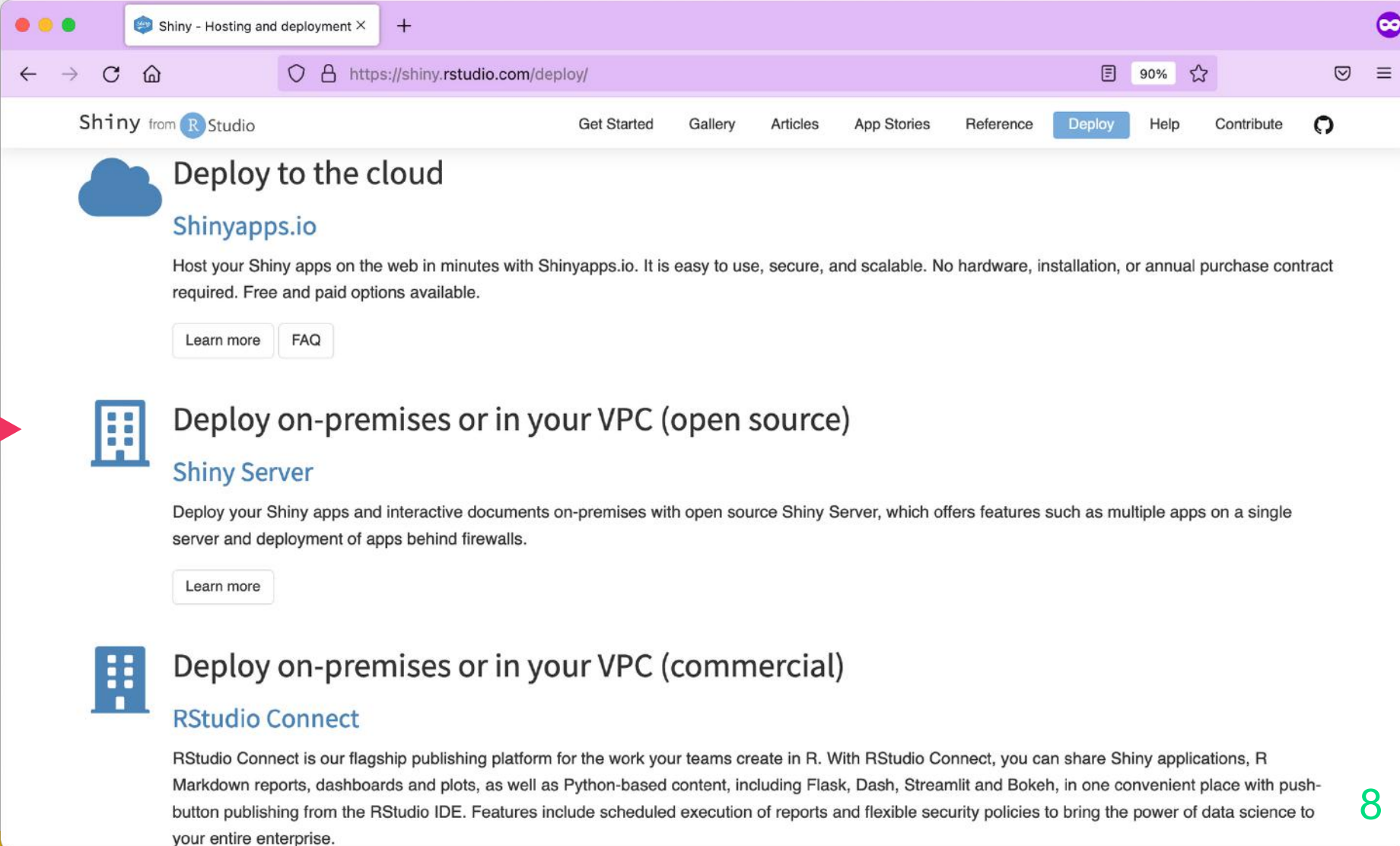
- Deploy to the cloud**
Shinyapps.io
Host your Shiny apps on the web in minutes with Shinyapps.io. It is easy to use, secure, and scalable. No hardware, installation, or annual purchase contract required. Free and paid options available.
[Learn more](#) [FAQ](#)
- Deploy on-premises or in your VPC (open source)**
Shiny Server
Deploy your Shiny apps and interactive documents on-premises with open source Shiny Server, which offers features such as multiple apps on a single server and deployment of apps behind firewalls.
[Learn more](#)
- Deploy on-premises or in your VPC (commercial)**
RStudio Connect
RStudio Connect is our flagship publishing platform for the work your teams create in R. With RStudio Connect, you can share Shiny applications, R Markdown reports, dashboards and plots, as well as Python-based content, including Flask, Dash, Streamlit and Bokeh, in one convenient place with push-button publishing from the RStudio IDE. Features include scheduled execution of reports and flexible security policies to bring the power of data science to your entire enterprise.



Conventional hosting options

Not so easy
& free

AGPL & DIY
watch your
dependencies



The screenshot shows a web browser window with the title "Shiny - Hosting and deployment". The address bar shows the URL "https://shiny.rstudio.com/deploy/". The page features a navigation bar with links: "Get Started", "Gallery", "Articles", "App Stories", "Reference", "Deploy" (highlighted), "Help", and "Contribute". The main content area lists three hosting options:

- Deploy to the cloud**
Shinyapps.io
Host your Shiny apps on the web in minutes with Shinyapps.io. It is easy to use, secure, and scalable. No hardware, installation, or annual purchase contract required. Free and paid options available.
[Learn more](#) [FAQ](#)
- Deploy on-premises or in your VPC (open source)**
Shiny Server
Deploy your Shiny apps and interactive documents on-premises with open source Shiny Server, which offers features such as multiple apps on a single server and deployment of apps behind firewalls.
[Learn more](#)
- Deploy on-premises or in your VPC (commercial)**
RStudio Connect
RStudio Connect is our flagship publishing platform for the work your teams create in R. With RStudio Connect, you can share Shiny applications, R Markdown reports, dashboards and plots, as well as Python-based content, including Flask, Dash, Streamlit and Bokeh, in one convenient place with push-button publishing from the RStudio IDE. Features include scheduled execution of reports and flexible security policies to bring the power of data science to your entire enterprise.



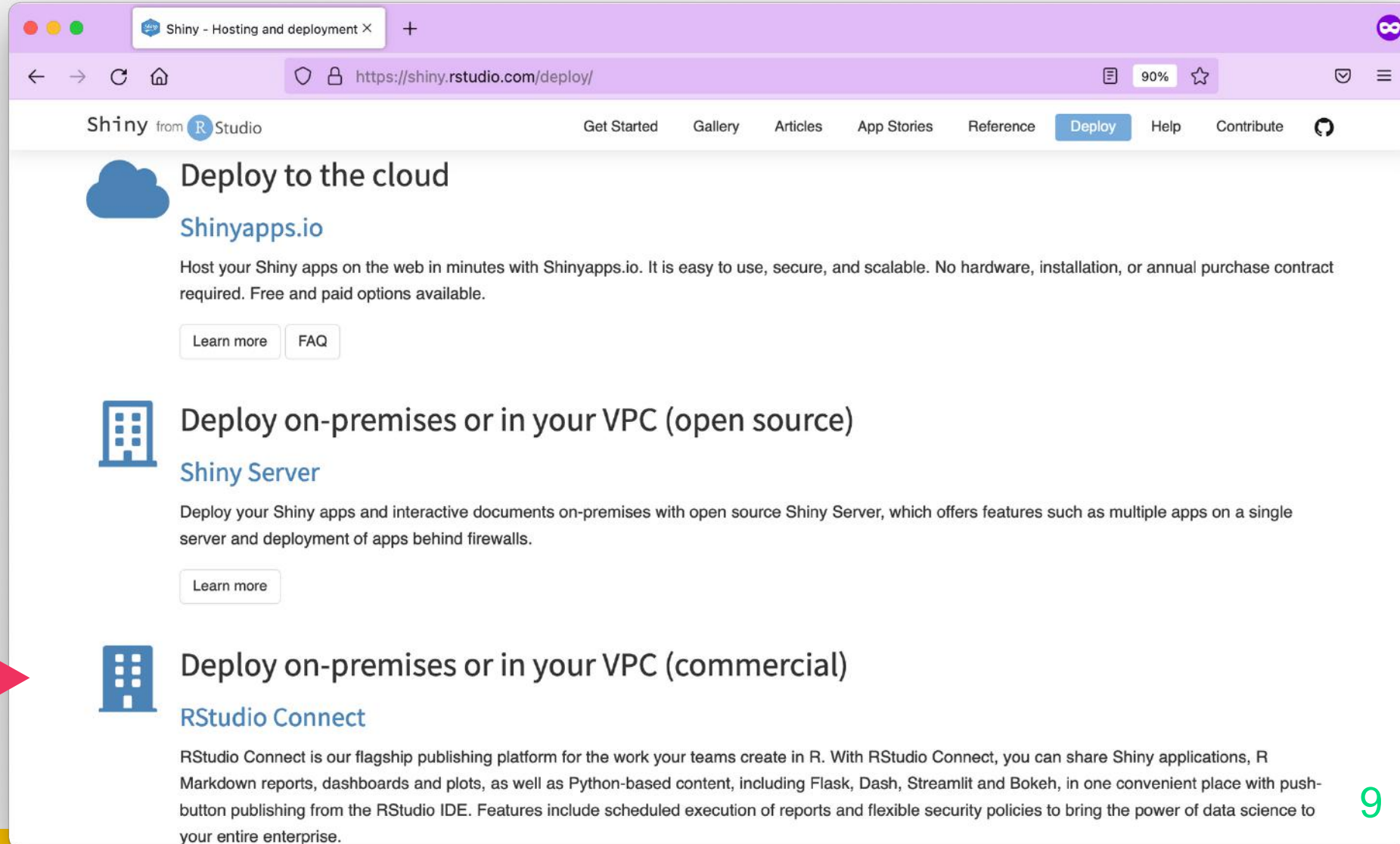
Conventional hosting options

Ultimate
experience for
enterprise

You get a lot more
than Shiny hosting



a8m.io/hosting

A screenshot of a web browser showing the 'Shiny - Hosting and deployment' page. The browser's address bar shows 'https://shiny.rstudio.com/deploy/'. The page has a navigation bar with links: 'Get Started', 'Gallery', 'Articles', 'App Stories', 'Reference', 'Deploy' (highlighted), 'Help', and 'Contribute'. The main content area lists three hosting options: 1. 'Deploy to the cloud' with the Shinyapps.io logo and a description: 'Host your Shiny apps on the web in minutes with Shinyapps.io. It is easy to use, secure, and scalable. No hardware, installation, or annual purchase contract required. Free and paid options available.' with 'Learn more' and 'FAQ' buttons. 2. 'Deploy on-premises or in your VPC (open source)' with the Shiny Server logo and a description: 'Deploy your Shiny apps and interactive documents on-premises with open source Shiny Server, which offers features such as multiple apps on a single server and deployment of apps behind firewalls.' with a 'Learn more' button. 3. 'Deploy on-premises or in your VPC (commercial)' with the RStudio Connect logo and a description: 'RStudio Connect is our flagship publishing platform for the work your teams create in R. With RStudio Connect, you can share Shiny applications, R Markdown reports, dashboards and plots, as well as Python-based content, including Flask, Dash, Streamlit and Bokeh, in one convenient place with push-button publishing from the RStudio IDE. Features include scheduled execution of reports and flexible security policies to bring the power of data science to your entire enterprise.'

What if you wanted

**unlimited apps
& uptime**

**custom domain
with HTTPS**

**for free or not
too expensive**



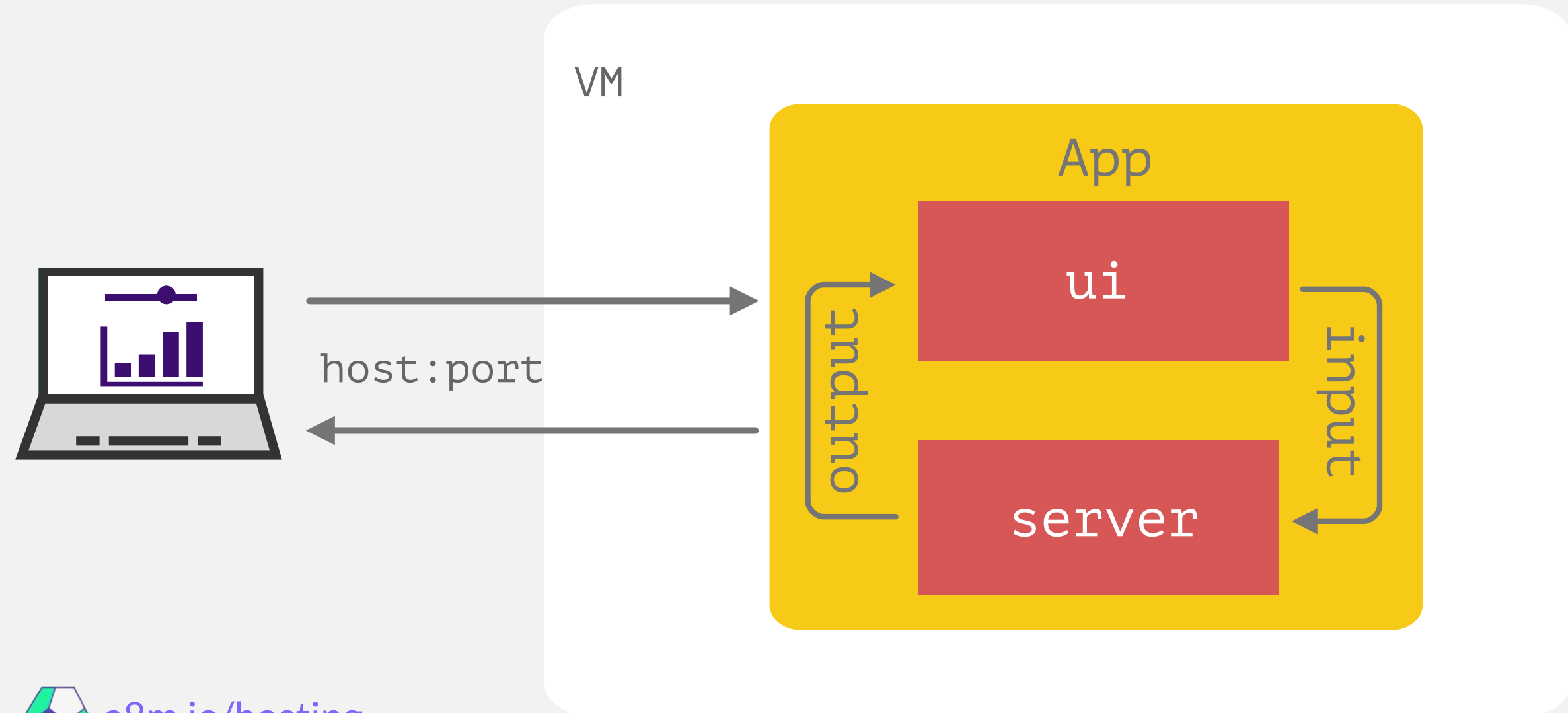
Your wish is granted but ...

Hosting Option	Total Cost (\$1000 USD)	Number of Apps	PaaS	Docker	Unlimited Hours	High Performance	Auth.	Custom Domain	HTTPS for Custom Domain	Non-Shiny Apps	Admin Access	Data Region	No WebSocket
RStudio Connect	>10	>25	N	N	Y	Y	Y	Y	Y	Y	Y	Y	N
Shinyapps.io, Free	Free	2-5	Y	N	N	N	N	N	N	N	N	N	N
Shinyapps.io, Starter	<1	6-25	Y	N	N	N	N	N	N	N	N	N	N
Shinyapps.io, Basic	<1	>25	Y	N	N	Y	N	N	N	N	N	N	N
Shinyapps.io, Standard	1-10	>25	Y	N	N	Y	Y	Y	N	N	N	N	N
Shinyapps.io, Professional	1-10	>25	Y	N	N	Y	Y	Y	N	N	N	N	N
Heroku, Free	Free	2-5	Y	Y	N	N	N	Y	Y	Y	Y	N	N
Heroku, Hobby/Standard	<1	1	Y	Y	Y	N	N	Y	Y	Y	Y	N	N
Heroku, Performance	1-10	1	Y	Y	Y	Y	N	Y	Y	Y	Y	N	N
DO App Platform, Basic	<1	1	Y	Y	Y	N	N	Y	Y	Y	Y	N	N
DO App Platform, Professional	<1	1	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	N
Shiny Server OS	<1	>25	N	N	Y	Y	N	Y	Y	N	Y	Y	Y
ShinyProxy	<1	>25	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
Docker on VM	<1	>25	N	Y	Y	Y	N	Y	Y	Y	Y	Y	N
systemd on VM	<1	>25	N	N	Y	Y	N	Y	Y	N	Y	Y	N

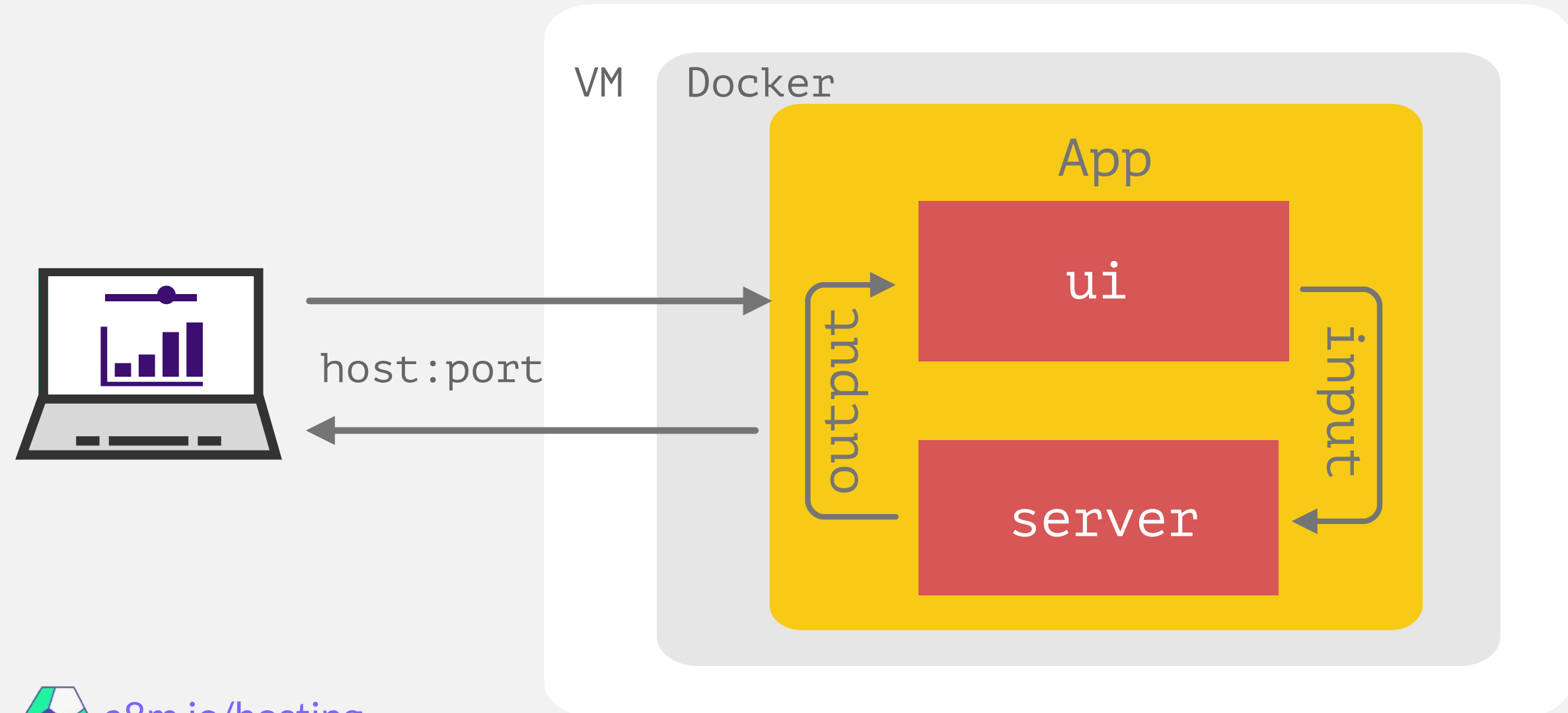
... you might have to learn Docker



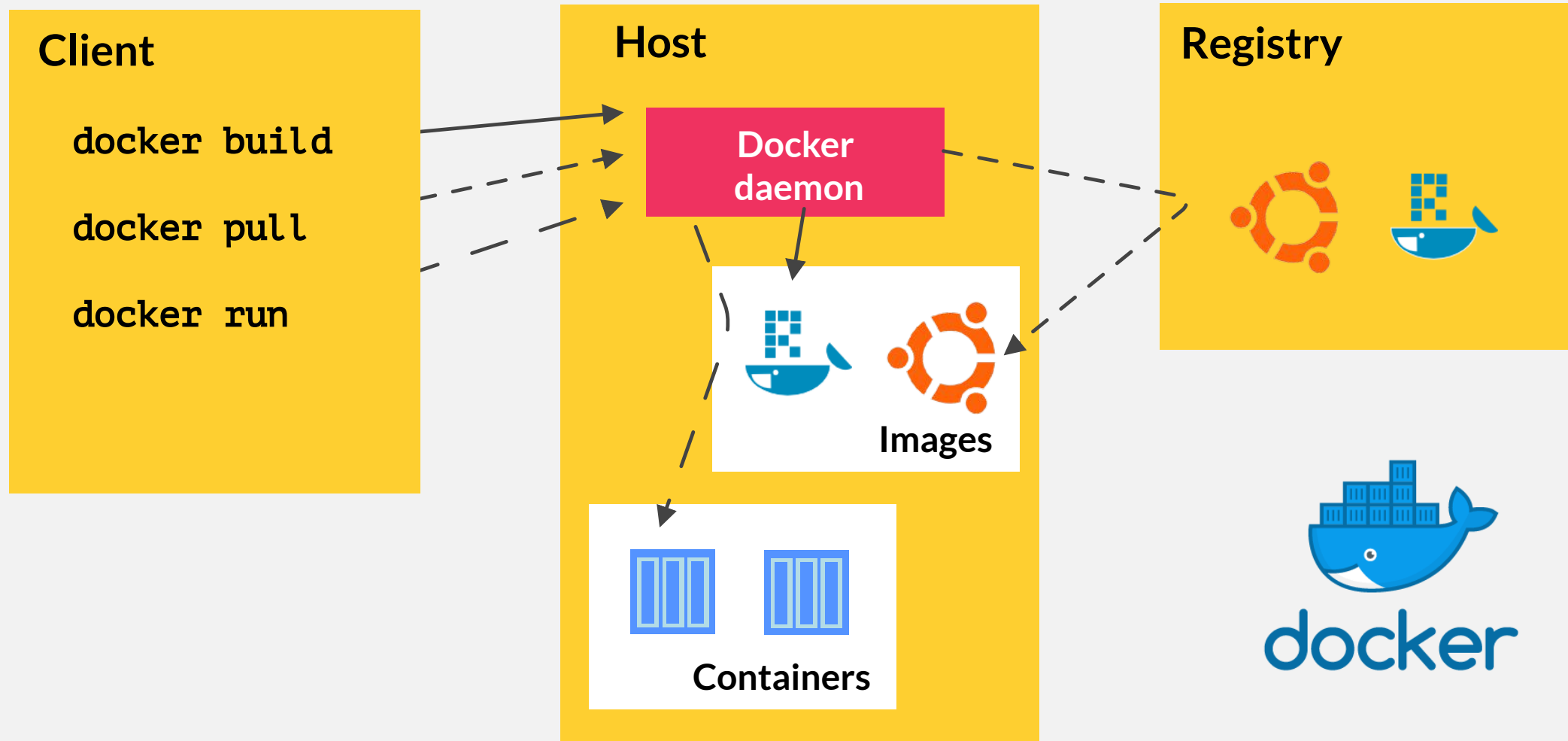
Hello world in Shiny



Docker isolates your app ...



... and connects it to an ecosystem





The Dockerfile

```
FROM rocker/r-base:4.0.4
```

```
RUN apt-get update && apt-get install -y \  
    --no-install-recommends \  
    make libssl-dev libxml2-dev \  
    && rm -rf /var/lib/apt/lists/*
```

```
COPY Rprofile.site /etc/R
```

```
RUN install.r shiny ggplot2 htmltools plotly
```

```
RUN addgroup --system app && adduser --system --ingroup app app
```

```
WORKDIR /home/app
```

```
COPY app .
```

```
RUN chown app:app -R /home/app
```

```
USER app
```

```
EXPOSE 3838
```

```
CMD ["R", "-e", "shiny::runApp('/home/app', port = 3838, host = '0.0.0.0')"]
```





Parent images

```
FROM rocker/r-base:4.0.4
```

```
RUN apt-get update && apt-get install -y \  
    --no-install-recommends \  
    make libssl-dev libxml2-dev \  
    && rm -rf /var/lib/apt/lists/*
```

```
COPY Rprofile.site /etc/R
```

```
RUN install.r shiny ggplot2 htmltools plotly
```

```
RUN addgroup --system app && adduser --system --ingroup app app
```

```
WORKDIR /home/app
```

```
COPY app .
```

```
RUN chown app:app -R /home/app
```

```
USER app
```

```
EXPOSE 3838
```

```
CMD ["R", "-e", "shiny::runApp('/home/app', port = 3838, host = '0.0.0.0')"]
```



Image sizes & build times vary

IMAGE_TAG	PARENT_GB	FINAL_GB	BUILD_MINS
rhub/r-minimal	0.035	0.222	27.2
rocker/r-base	0.761	1.050	11.6
rocker/r-ubuntu	0.673	1.220	11.9
rstudio/r-base	0.894	1.380	3.1
rocker/shiny	1.380	1.610	2.3
rocker/r-bspm	0.758	0.898	1.5
eddelbuettel/r2u	0.804	0.959	0.6



Order of layers

1

```
FROM rocker/r-base:4.0.4
```

2

```
RUN apt-get update && apt-get install -y \
    --no-install-recommends \
    make libssl-dev libxml2-dev \
    && rm -rf /var/lib/apt/lists/*
```

3

```
COPY Rprofile.site /etc/R
RUN install.r shiny ggplot2 htmltools plotly
```

4

```
RUN addgroup --system app && adduser --system --ingroup app app
WORKDIR /home/app
COPY app .
RUN chown app:app -R /home/app
USER app
```

```
EXPOSE 3838
```

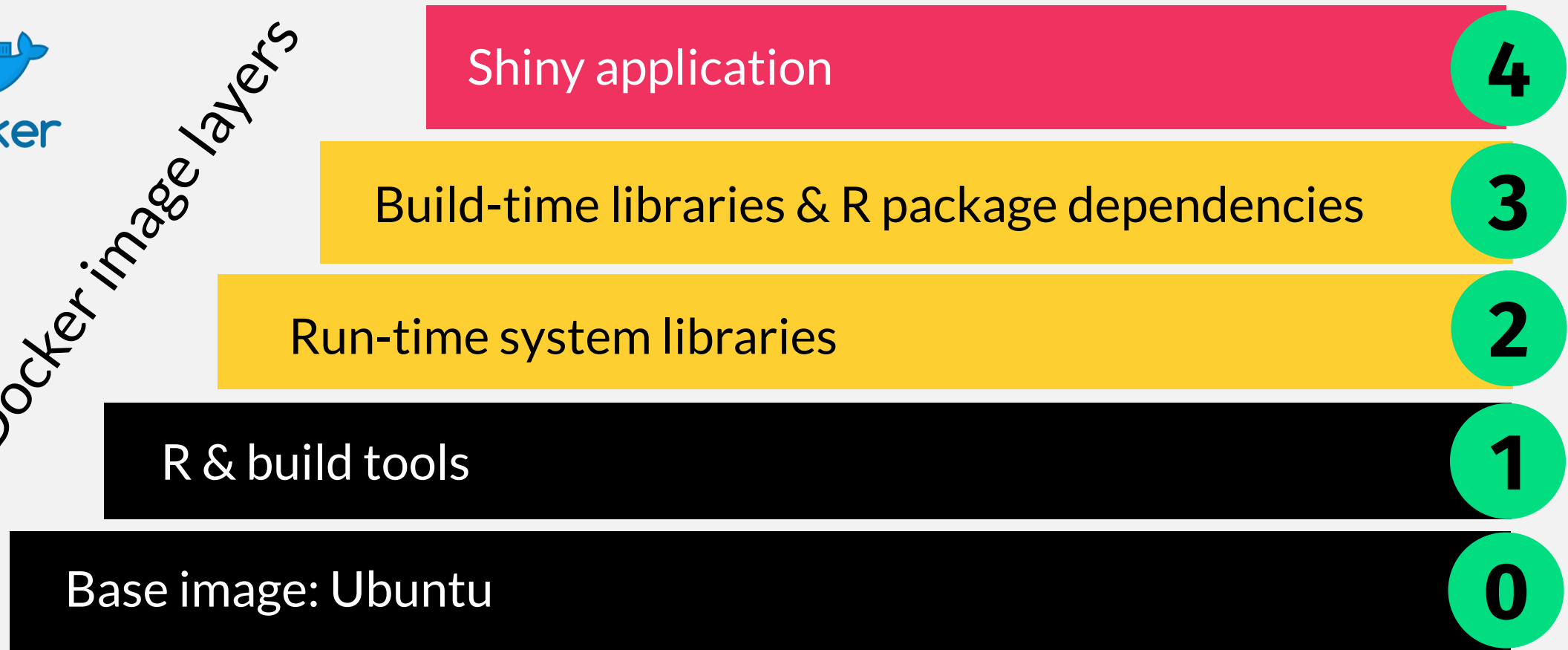
```
CMD ["R", "-e", "shiny::runApp('/home/app', port = 3838, host = '0.0.0.0')"]
```



Dependencies before app code



Docker image layers





Dependencies

Explicit

```
RUN install.r shiny ggplot2 htmltools plotly
```

DESCRIPTION file

```
RUN install.r remotes
```

```
COPY DESCRIPTION .
```

```
RUN Rscript -e "remotes::install_deps()"
```

renv package

```
RUN install.r remotes renv
```

```
COPY ./renv.lock .
```

```
RUN Rscript -e "options(renv.consent=TRUE);  
renv::restore(lockfile='/home/app/renv.lock')"
```





Security

```
FROM rocker/r-base:4.0.4
```

```
RUN apt-get update && apt-get install -y \  
    --no-install-recommends \  
    make libssl-dev libxml2-dev \  
    && rm -rf /var/lib/apt/lists/*
```

```
COPY Rprofile.site /etc/R
```

```
RUN install.r shiny ggplot2 htmltools plotly
```

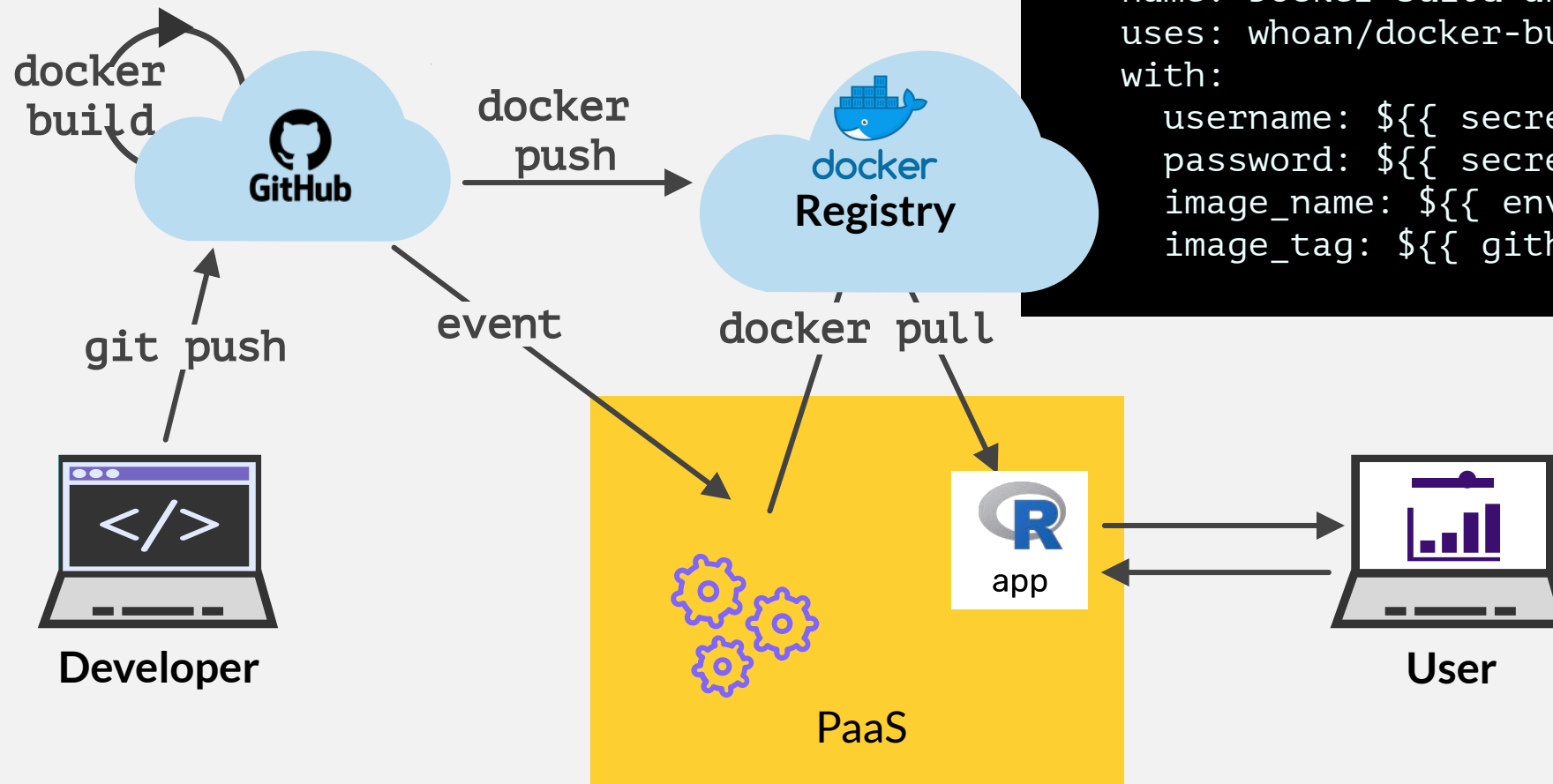
```
RUN addgroup --system app && adduser --system --ingroup app app  
WORKDIR /home/app  
COPY app .  
RUN chown app:app -R /home/app  
USER app
```

```
EXPOSE 3838
```

```
CMD ["R", "-e", "shiny::runApp('/home/app', port = 3838, host = '0.0.0.0')"]
```



Caching & CI/CD



```
- name: Docker build and push with cache
  uses: whoan/docker-build-with-cache-action@v5
  with:
    username: ${ secrets.DOCKERHUB_USERNAME }
    password: ${ secrets.DOCKERHUB_ACCESS_TOKEN }
    image_name: ${ env.REGISTRY }/${ env.IMAGE_NAME }
    image_tag: ${ github.sha }
```

Recap of best practices

Choose your base images wisely

Pay attention to dependencies

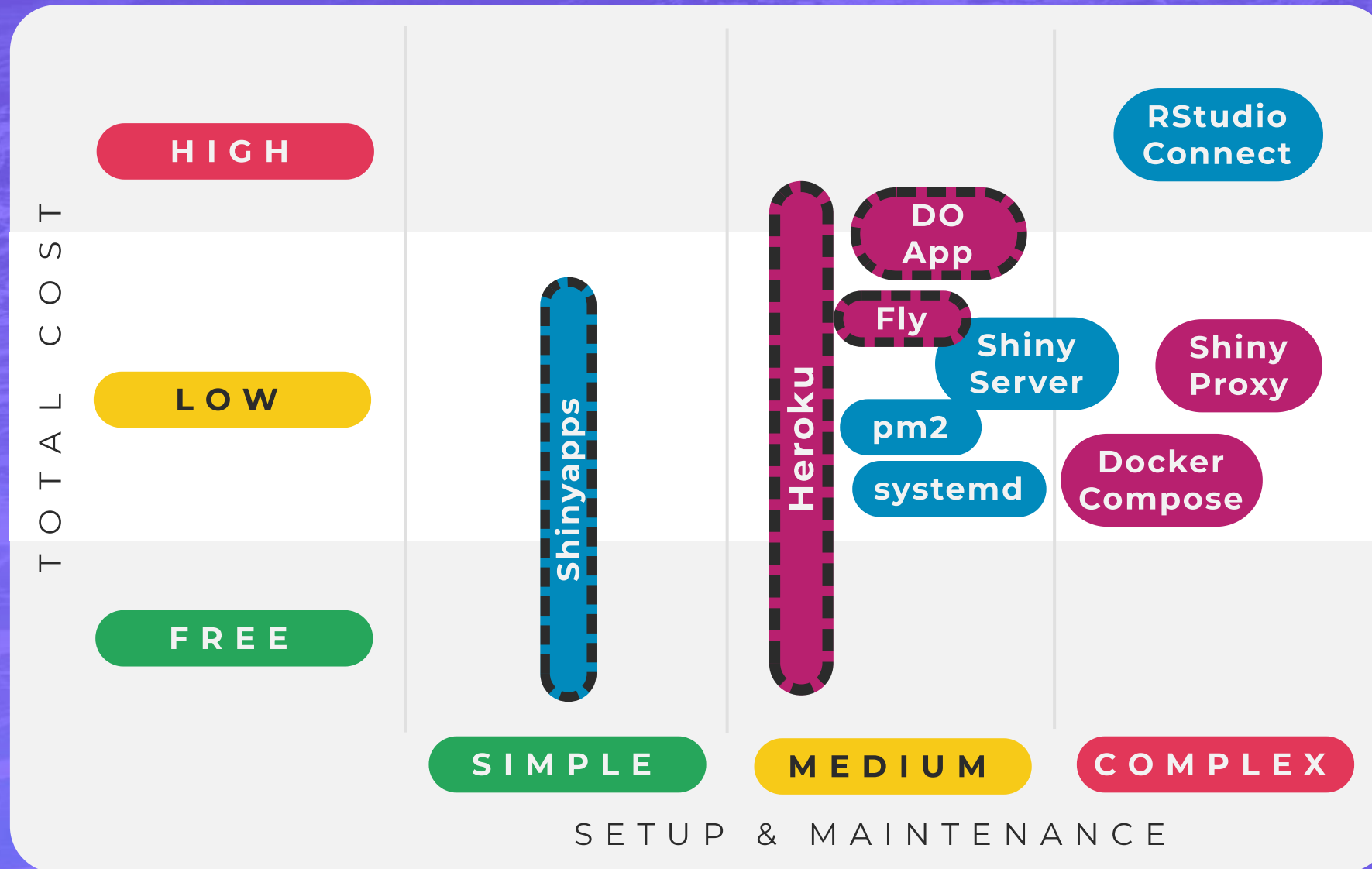
Set a non-root user

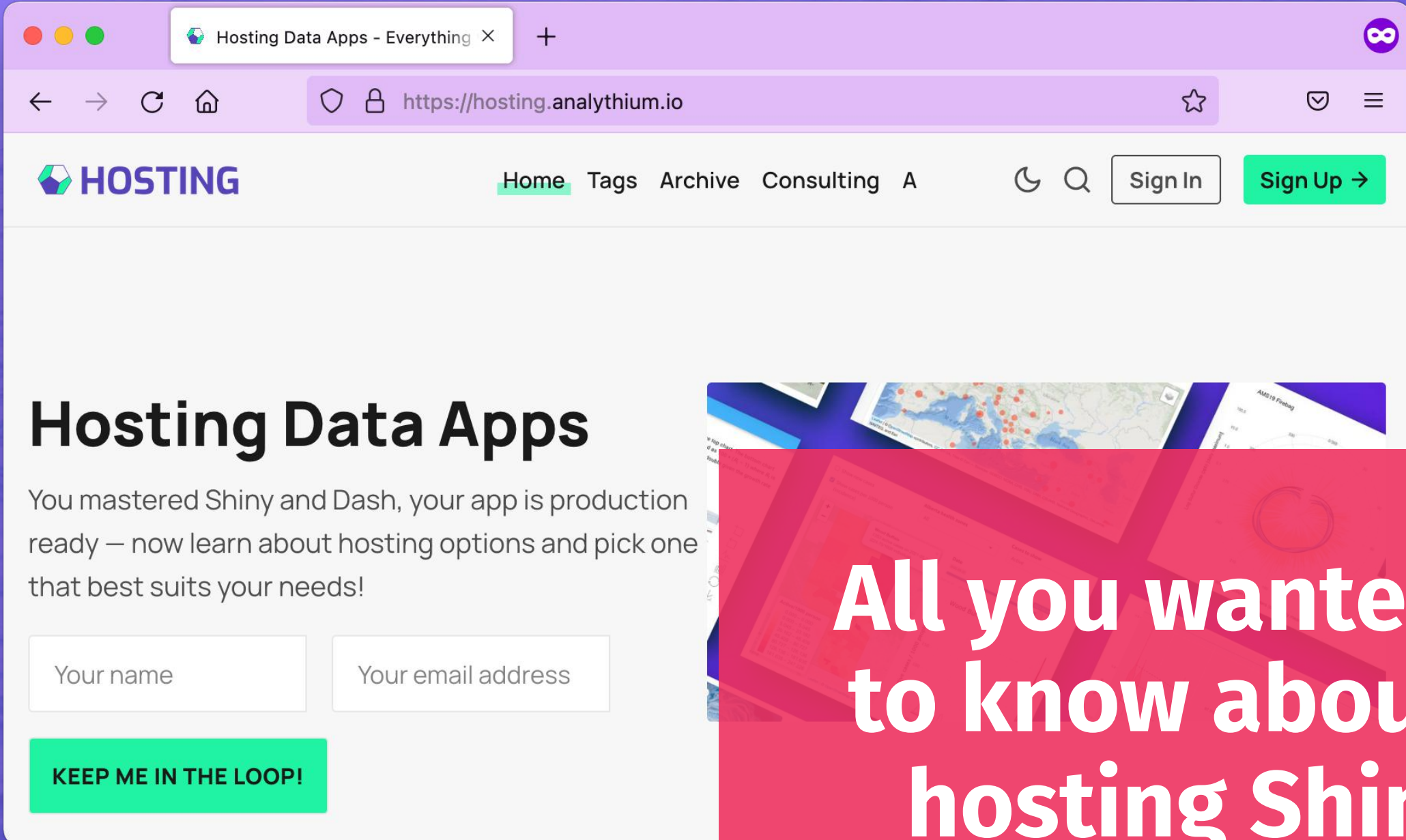
Order layers based on how often they change

Use caching with CI/CD and save the Planet



Shiny hosting comparison



A screenshot of a web browser displaying the 'Hosting Data Apps' website. The browser's address bar shows 'https://hosting.analythium.io'. The website has a purple header with the 'HOSTING' logo and navigation links: 'Home', 'Tags', 'Archive', 'Consulting', and 'A'. There are also 'Sign In' and 'Sign Up' buttons. The main content area features the title 'Hosting Data Apps' and a paragraph: 'You mastered Shiny and Dash, your app is production ready – now learn about hosting options and pick one that best suits your needs!'. Below this is a form with two input fields: 'Your name' and 'Your email address', and a green button labeled 'KEEP ME IN THE LOOP!'. A red semi-transparent overlay on the right side of the page contains the text 'All you wanted to know about hosting Shiny'.