

# Forecast Infrastructure

2022-07-01

Near-term Ecological Forecasting Initiative Short Course

Jake Zwart (he/him)

With input and slides used from USGS co-workers





**Sprague, Lori A** 10/27 10:58 AM

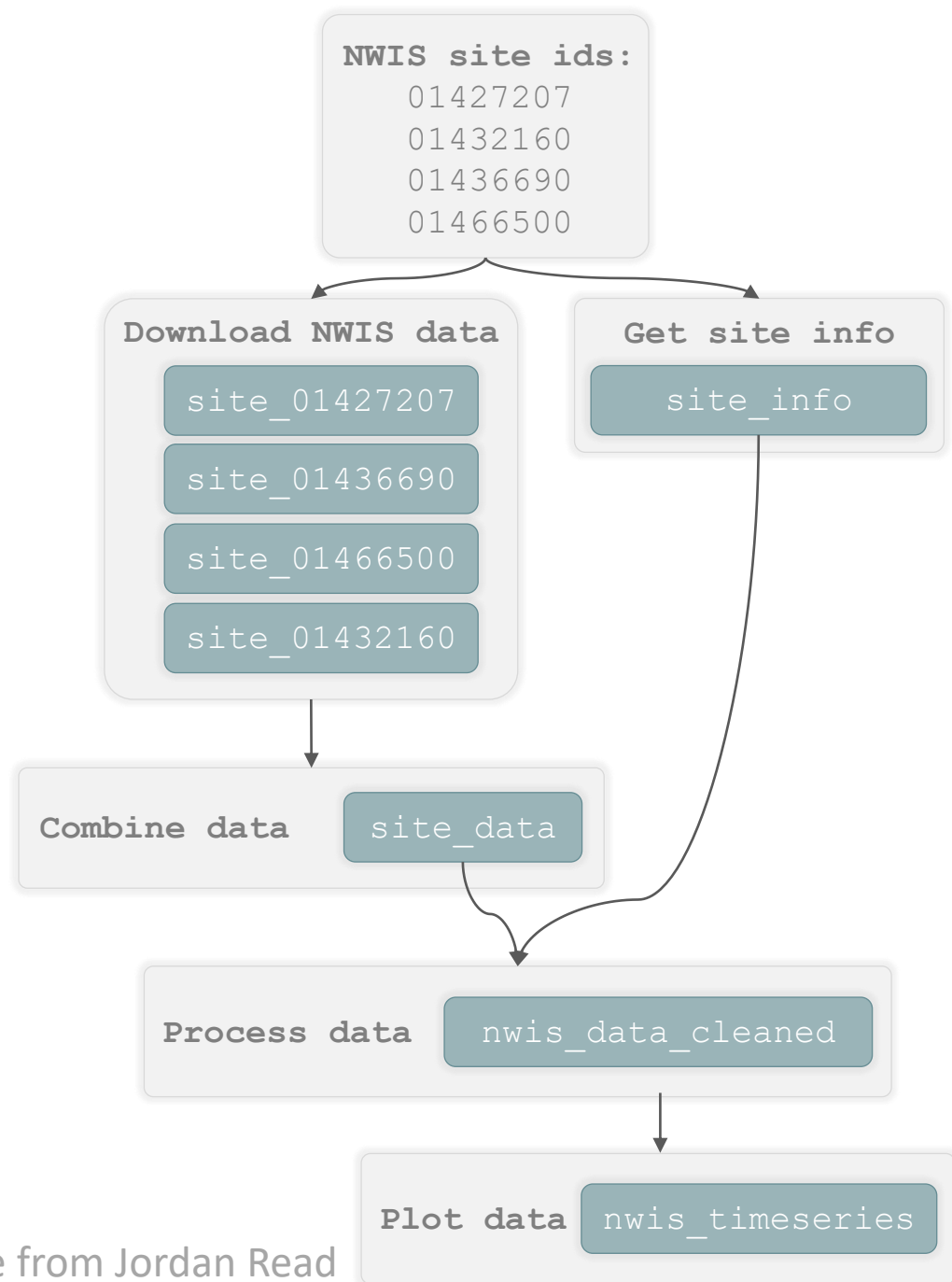
Shoot, I know I told you we were done with the analysis and data release, but I just heard that site 01436690 had a sensor malfunction and they've since corrected the data

10:59 AM

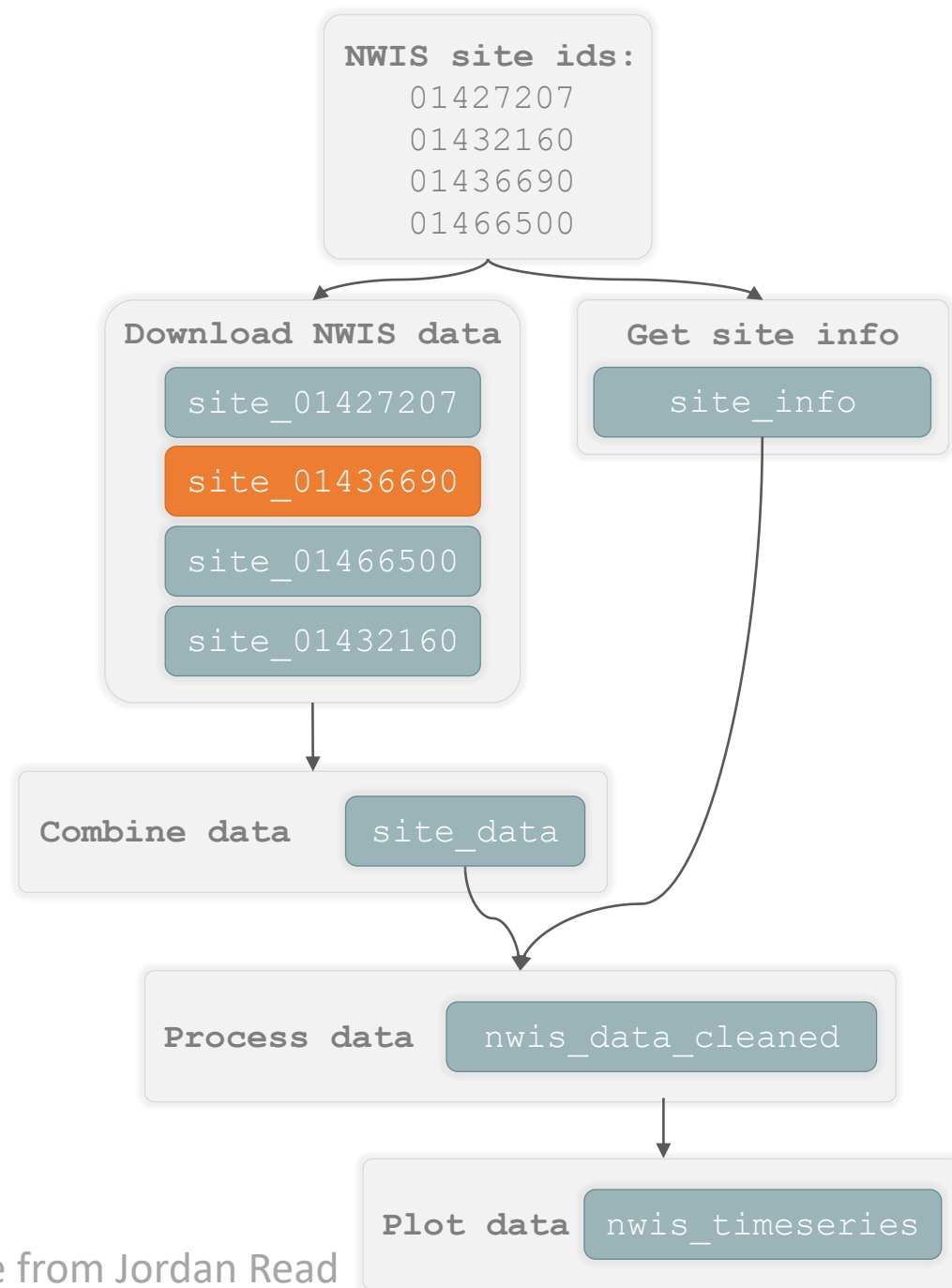


No problem. Easy fix on our end!









```
R Console | C:/Users/usgs_employee/complex_analysis

> run_pipeline()
> Skipping site_01427207
> Building site_01436690
> Skipping site_01466500
> Skipping site_01432160
> Building site_data
> Skipping site_info
> Building nwis_data_cleaned
> Building nwis_timeseries_plot
```

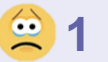




Sprague, Lori A 10/27 10:58 AM

Shoot, I know I told you we were done with the analysis and data release, but I just heard that site 01436690 had a sensor malfunction and they've since corrected the data

10:59 AM



Umm...I *think* I can figure out which results are impacted





Sprague, Lori A 10/27 10:58 AM

Shoot, I know I told you we were done with the analysis and data release, but I just heard that site 01436690 had a sensor malfunction and they've since corrected the data

10:59 AM



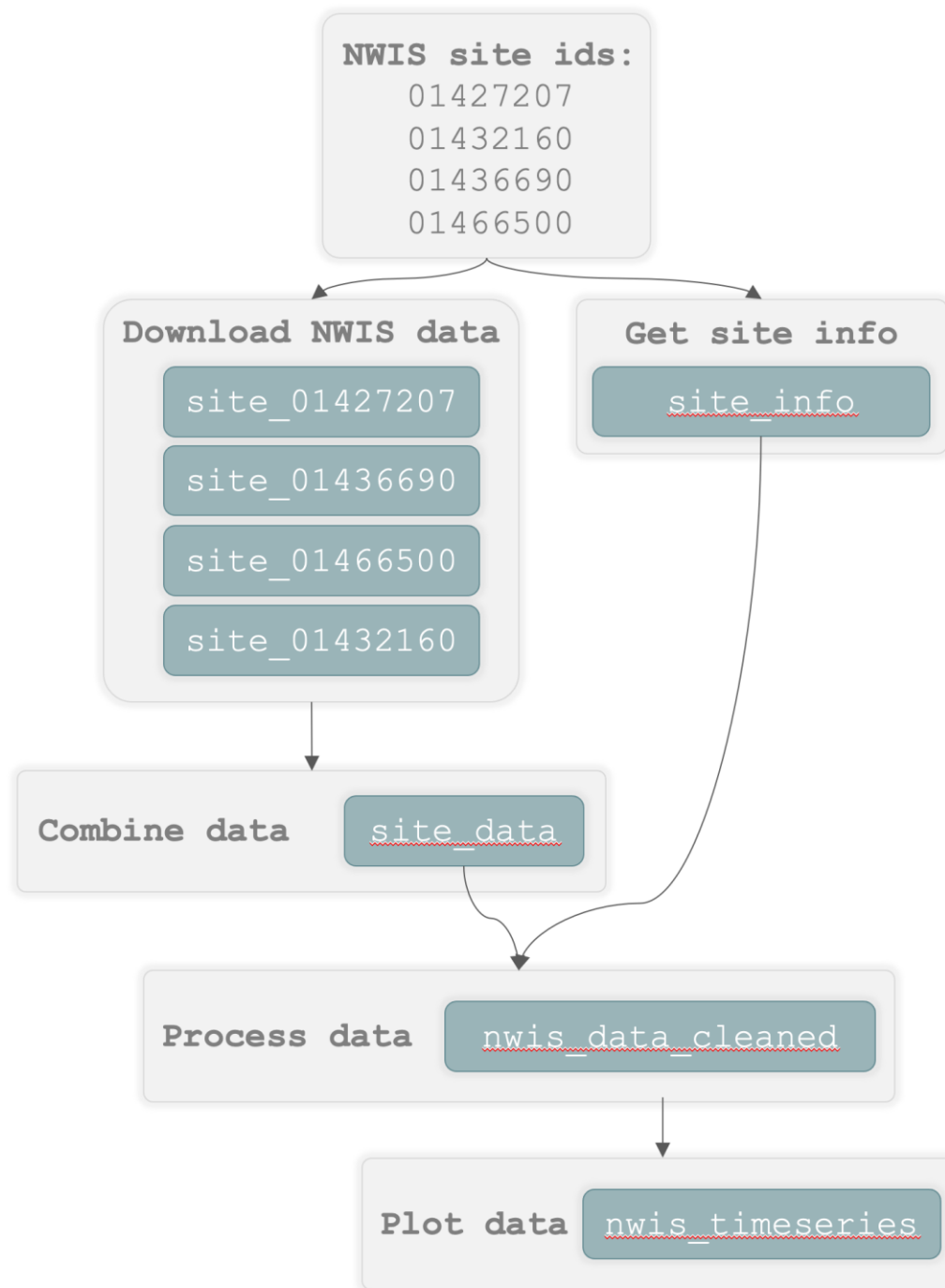
Umm...I *think* I can figure out which results are impacted

11:00 AM

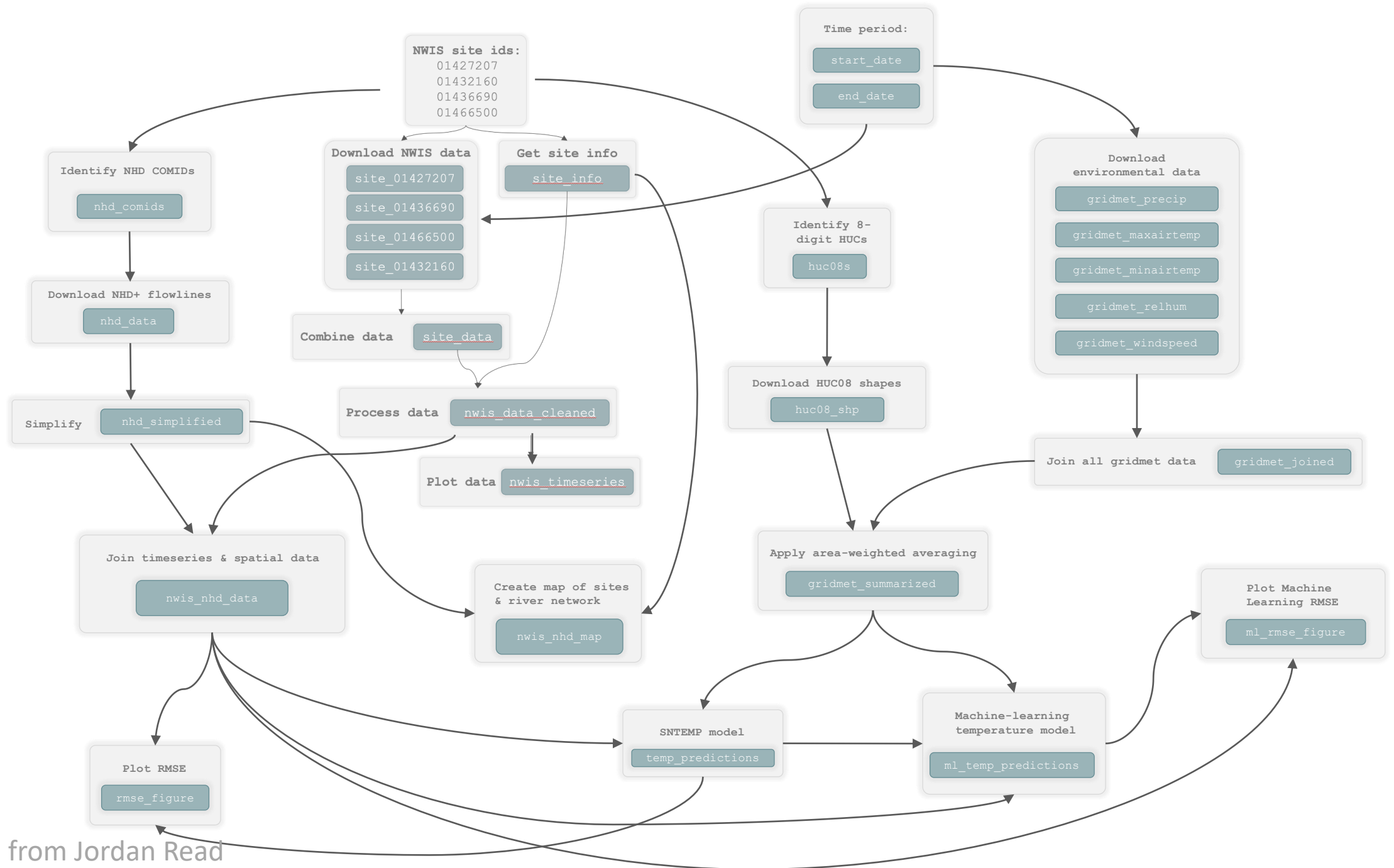


Never mind. I can just re-run the whole thing. Might take some time though











# Tiers of reproducibility and repeatability

**Document and  
rebuild your own  
analysis**

Scripting

**Rebuild, easily reuse  
and troubleshoot,  
plus track changes  
over time**

Scripting

Modular functions

Version control

**Efficiently rebuild a  
full analysis and  
collaborate effectively  
on development**

Scripting

Modular functions

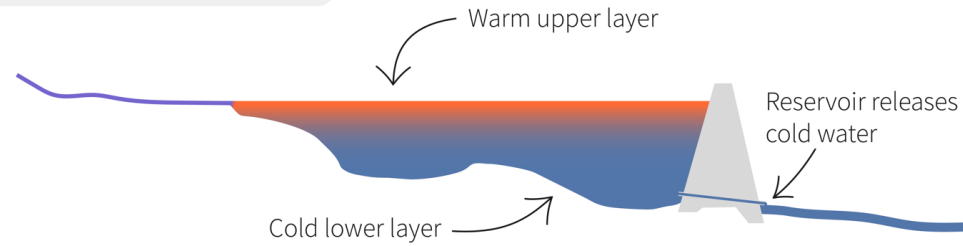
Version control

Dependency tracking

Automated Workflows



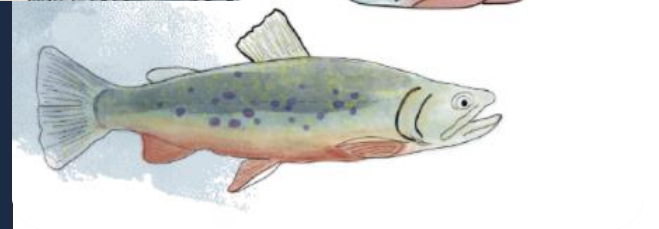
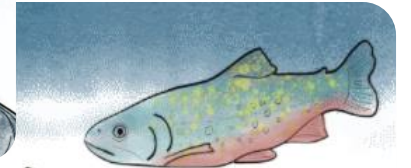
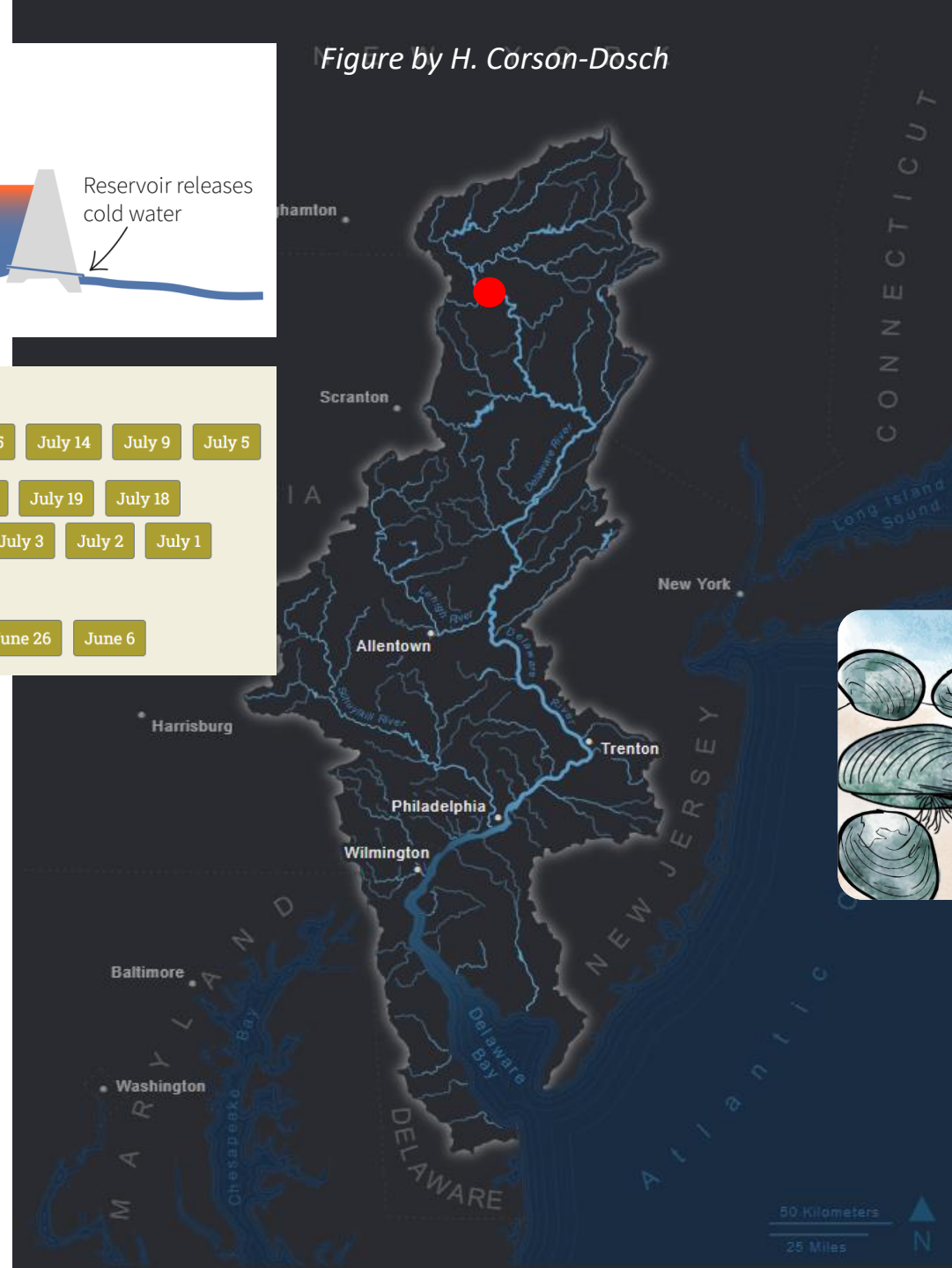
## Reservoir thermal bank



## Thermal Releases

<b>2019</b>	August 2	July 29	July 27	July 19	July 17	July 15	July 14	July 9	July 5
<b>2020</b>	July 31	July 28	July 27	July 26	July 25	July 20	July 19	July 18	July 17
	July 14	July 8	July 7	July 6	July 5	July 4	July 3	July 2	July 1
	June 22	June 20	June 18	June 9					
<b>2021</b>	August 11	August 10	June 29	June 28	June 27	June 26	June 6		

Figure by H. Corson-Dosch





## Operational Needs

Produce daily updating 7-day forecast

Use most recent meteorological forecasts and observation data available

Forecasts visible to NYC reservoir operators by mid-morning EDT



# Daily workflow

12:00am – 4am EDT

NOAA forecasted  
drivers download  
(GEFS)



Process drivers,  
extract forecasts  
for forecast  
locations



Stream  
temperature and  
reservoir release  
data (NWIS)



Run location-  
specific PGDL-DA  
models, predicting  
7-day max  
temperatures



Evaluate forecasts



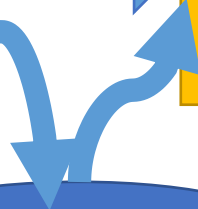
S3 evaluation  
archive and  
website

9:00am EDT

9:30am EDT

10:00am EDT

S3 forecast archive

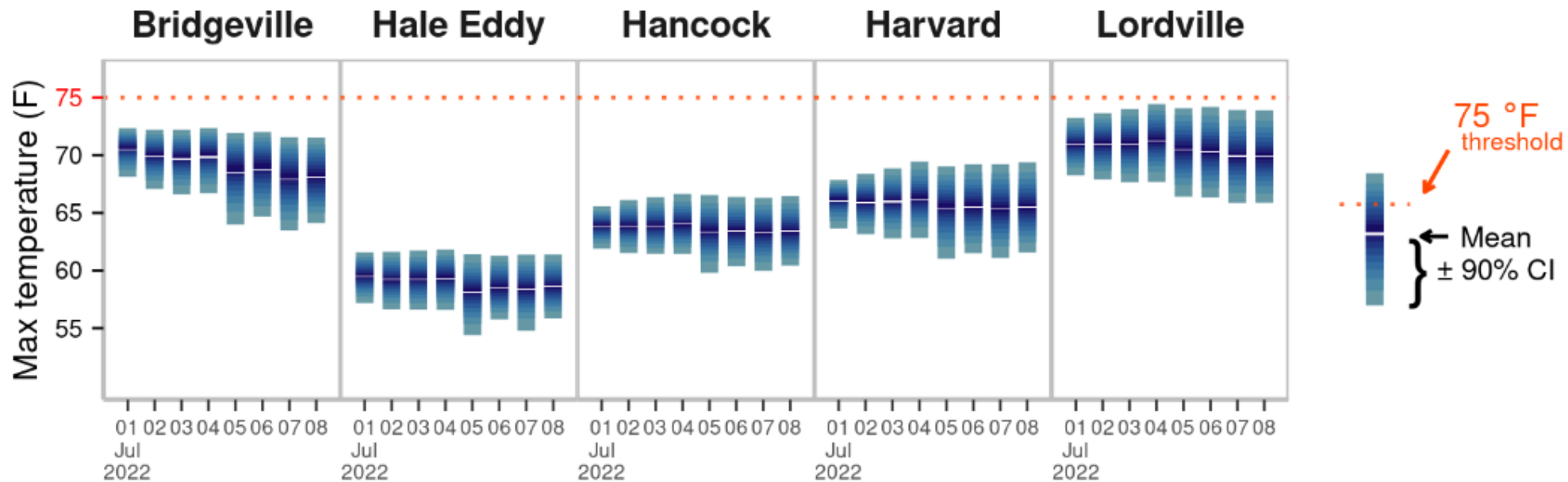




# Stream Temperature Forecasts for Sites in the Delaware River Basin

## Today's Forecast

Issued on 2022-07-01





# Keys to success

---

**Software dev best practices:** Able to isolate operational workflows from code under development

---

**Modular design:** Workflow pieces could be modified and scheduled independently

---

**Containers:** Workflows were instantly portable to cloud from local machines

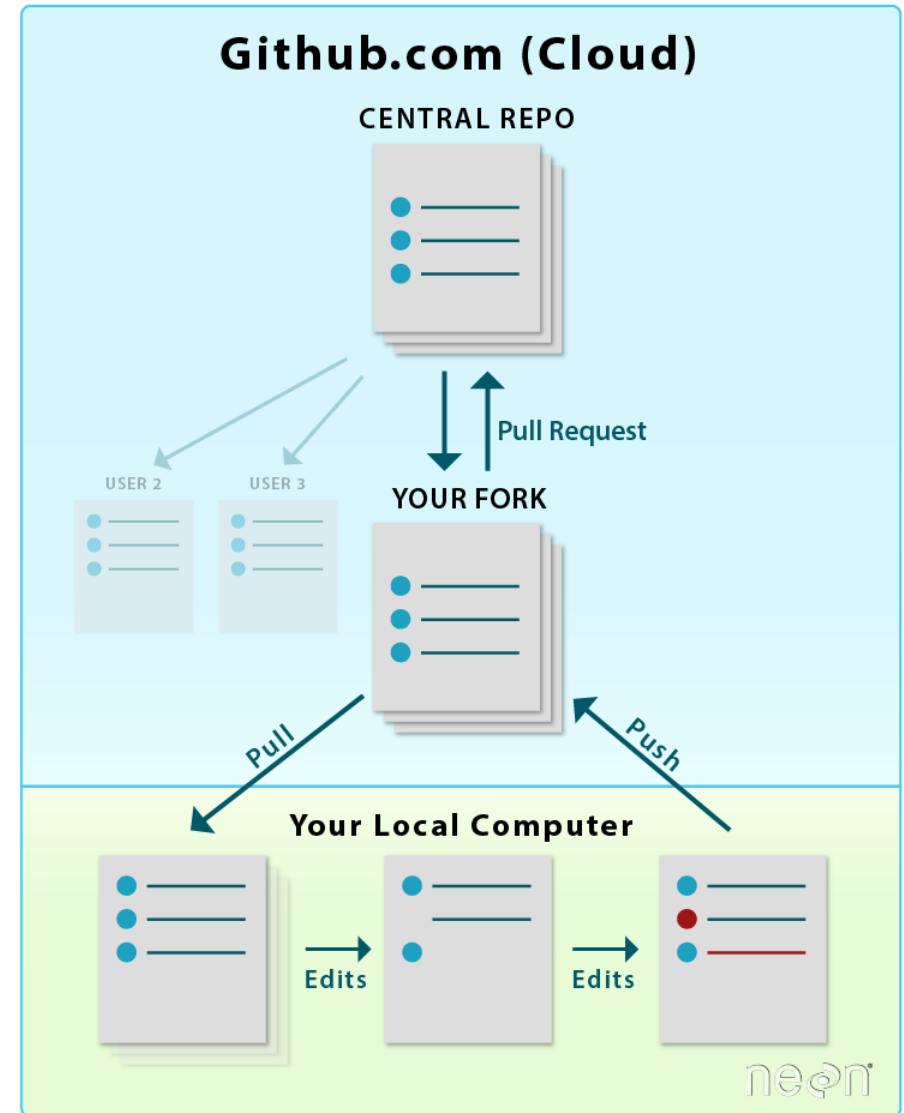
---

**Teamwork:** Clearly defined roles, diversity of skills, and focused communication among staff



# Software development best practices

- Version control – Git and GitHub





# Software development best practices

- Version control – Git and GitHub
- Use tagged releases – snapshots of codebase



Tags give the ability to mark specific points in history as being important

📦 **v1.0**

🔗 [da98b0aa](#) · Merge branch 'fix\_dropout\_en' into 'main' · 1 month ago

🔗 Release [v1.0](#)

`FY22 PGDL-DA temperature forecasting codebase. Major updates from FY21`

📦 **v0.4.2**

🔗 [73ec74f6](#) · Merge branch 'update\_cons\_release' into 'master' · 10 months ago

🔗 Release [v0.4.2](#)

`Updating Cannonsville conservation release`

📦 **v0.4.1**

🔗 [81e7bae0](#) · Merge branch 'update\_cons\_release' into 'master' · 10 months ago

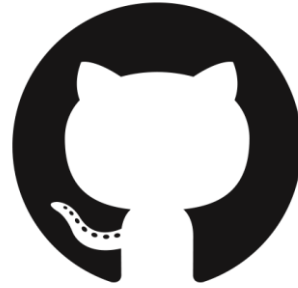
🔗 Release [New reservoir conservation releases for August 2021](#)

`New reservoir conservation releases for August 2021`



# Software development best practices

- Version control – Git and GitHub
- Use tagged releases – snapshots of codebase



Tags give the ability to mark specific points in history as being important

**v1.0**

da98b0aa · Merge branch 'fix\_dropout\_en' into 'main' · 1 month ago

Release v1.0

FY22 PGDL-DA temperature forecasting codebase. Major updates from FY21

**v0.4.2**

73ec74f6 · Merge branch 'update\_cons\_release' into 'master' · 10 months ago

Release v0.4.2

Updating Cannonsville conservation release

**v0.4.1**

81e7bae0 · Merge branch 'update\_cons\_release' into 'master' · 10 months ago

Release New reservoir conservation releases for August 2021

New reservoir conservation releases for August 2021



# Software development best practices

- Version control – Git and GitHub
- Use tagged releases – snapshots of codebase
- Unit testing – ensure parts of code behave as expected



testthat



# Software development best practices

- Version control – Git and GitHub
- Use tagged releases – snapshots of codebase
- Unit testing – ensure parts of code behave as expected
- Continuous integration – automated tests and deployment



testthat



GitHub Actions



**Jenkins**





# Modular design

12:00am – 4am EDT

NOAA forecasted  
drivers download  
(GEFS)



Process drivers,  
extract forecasts  
for forecast  
locations



Stream  
temperature and  
reservoir release  
data (NWIS)



Run location-  
specific PGDL-DA  
models, predicting  
7-day max  
temperatures



Evaluate forecasts



S3 evaluation  
archive and  
website

9:00am EDT

9:30am EDT

10:00am EDT

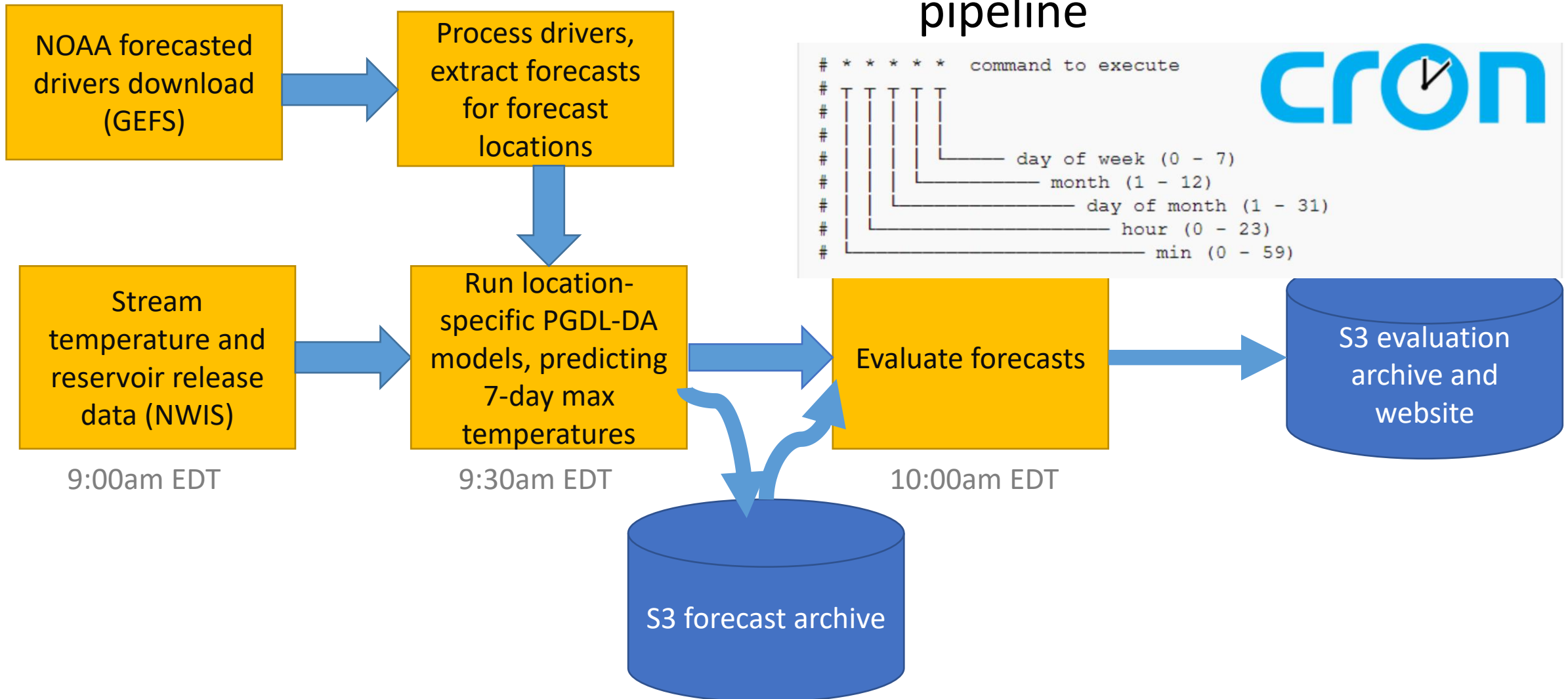


S3 forecast archive



# Modular design

12:00am – 4am EDT



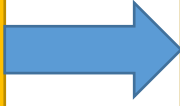
- Separate repository for each part of the entire pipeline



# Modular design

12:00am – 4am EDT

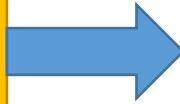
NOAA forecasted  
drivers download  
(GEFS)



Process drivers,  
extract forecasts  
for forecast  
locations



Stream  
temperature and  
reservoir release  
data (NWIS)



Run location-  
specific PGDL-DA  
models, predicting  
7-day max  
temperatures



Evaluate forecasts



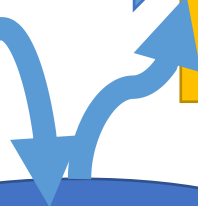
S3 evaluation  
archive and  
website

9:00am EDT

9:30am EDT

10:00am EDT

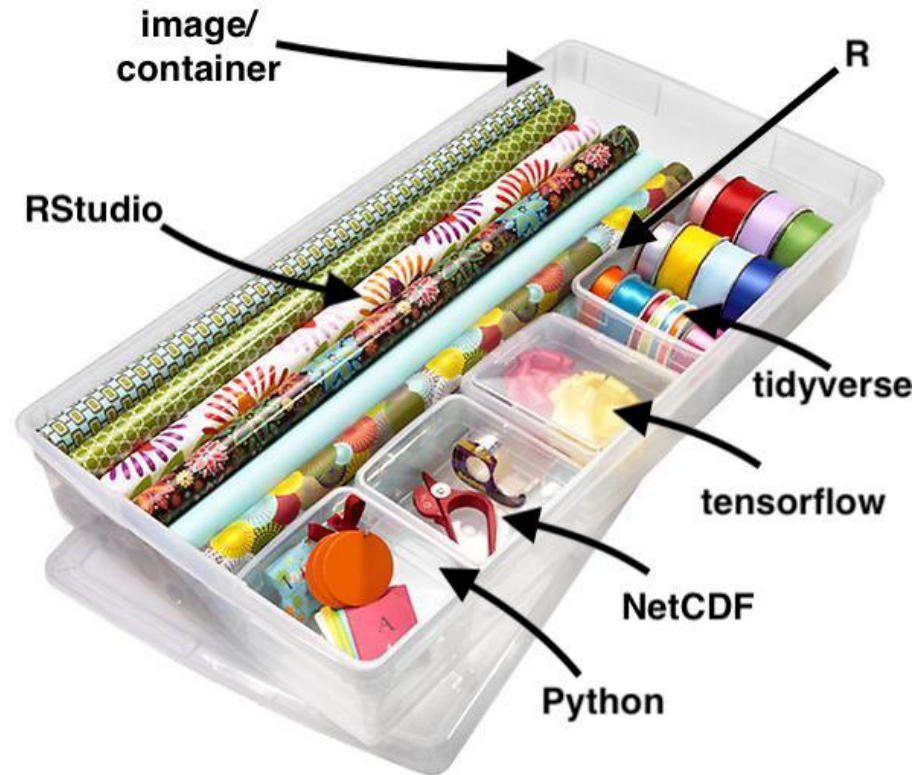
S3 forecast archive



- Separate repository for each part of the entire pipeline
- Within repositories, use functions



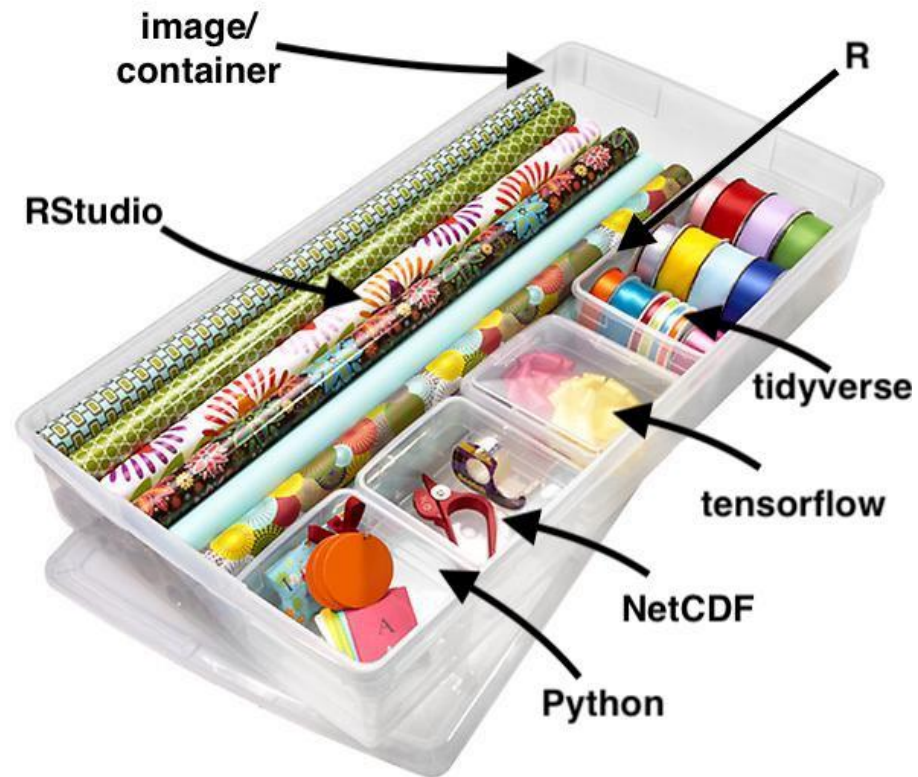
# Containers



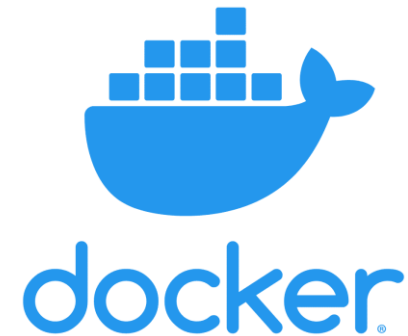
- Box that carries all the tools you need for a set of tasks



# Containers

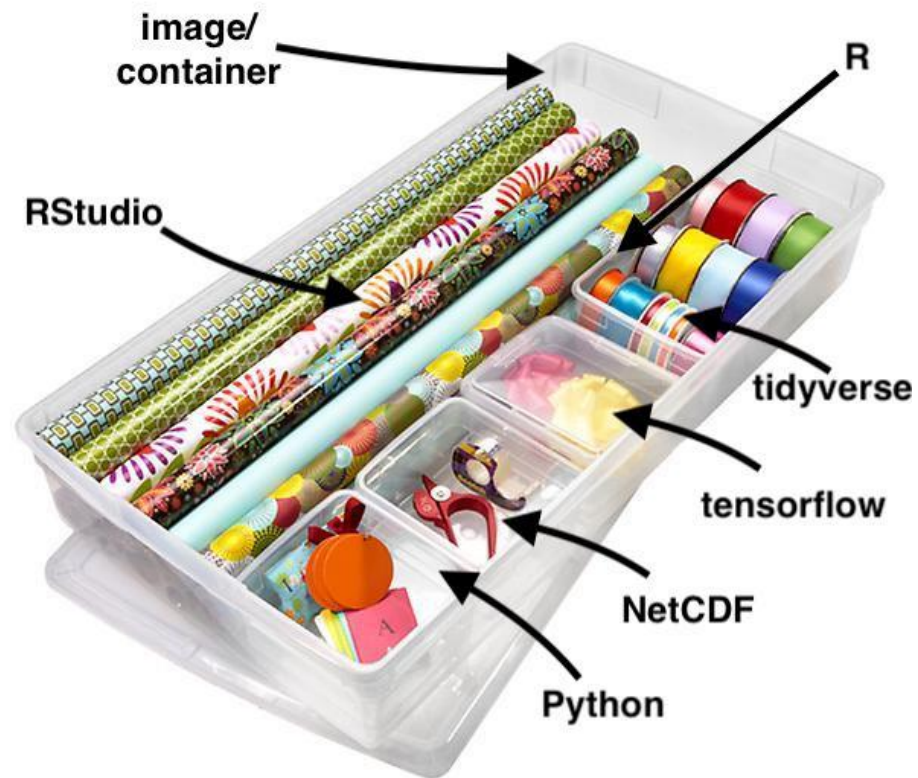


- Box that carries all the tools you need for a set of tasks
- Docker  
<https://docs.docker.com/get-started/overview/>





# Containers



- Box that carries all the tools you need for a set of tasks
- Docker  
<https://docs.docker.com/get-started/overview/>
- rocker – containers for the R environment



The Rocker Project  
Docker Containers for the R Environment



# Containers

## 2 Images

### 2.1 The versioned stack

image	base image	description	pulls
<a href="#">rocker/r-ver</a>	<a href="#">ubuntu</a>	Install R from source and set RSPM as default CRAN mirror	docker pulls 2.4M
<a href="#">rocker/rstudio</a>	<a href="#">rocker/r-ver</a>	Adds RStudio Server	docker pulls 7.9M
<a href="#">rocker/tidyverse</a>	<a href="#">rocker/rstudio</a>	Adds tidyverse packages & devtools	docker pulls 4.8M
<a href="#">rocker/verse</a>	<a href="#">rocker/tidyverse</a>	Adds tex & publishing-related package	docker pulls invalid
<a href="#">rocker/geospatial</a>	<a href="#">rocker/verse</a>	Adds geospatial packages	docker pulls 632k
<a href="#">rocker/binder</a>	<a href="#">rocker/geospatial</a>	Adds requirements to run repositories on <a href="#">mybinder.org</a>	
<a href="#">rocker/shiny</a>	<a href="#">rocker/r-ver</a>	Adds shiny server	docker pulls 1.7M
<a href="#">rocker/shiny-verse</a>	<a href="#">rocker/shiny</a>	Adds tidyverse packages	docker pulls invalid
<a href="#">rocker/cuda</a>	<a href="#">rocker/r-ver</a>	Adds CUDA support to <a href="#">rocker/r-ver</a>	docker pulls invalid
<a href="#">rocker/ml</a>	<a href="#">rocker/cuda</a>	Adds CUDA support to <a href="#">rocker/tidyverse</a>	docker pulls invalid
<a href="#">rocker/ml-verse</a>	<a href="#">rocker/ml</a>	Adds CUDA support to <a href="#">rocker/geospatial</a>	docker pulls 21k

- Box that carries all the tools you need for a set of tasks
- Docker  
<https://docs.docker.com/get-started/overview/>
- rocker – containers for the R environment



The Rocker Project  
Docker Containers for the R Environment



# Containers + Continuous Integration

Do something if  
these files  
change
















Build docker  
container based  
on our recipe

```
39 rules:
40   - changes:
41     - docker/Dockerfile
42     - docker/Makefile
43     - docker/requirements.txt
44     - docker/DOIRootCA2.crt
45     - .gitlab-ci.yml
46   script:
47     - echo "${CI_REGISTRY_PASSWORD}" | docker login -u "${CI_REGISTRY_USER}"
48     - cd docker
49     - apk add make
50     - make build
51     - make push
```





# Containers + Continuous Integration

<input type="checkbox"/>	<b>BUILD_faab01b5</b>  	Published 2 months ago	
	1.87 GiB	Digest: 346448e	
<input type="checkbox"/>	<b>BUILD_fe2a4f73</b>  	Published 2 months ago	
	421.79 MiB	Digest: 7556204	
<input type="checkbox"/>	<b>latest</b>  	Published 2 days ago	
	422.86 MiB	Digest: c21a0ef	
<input type="checkbox"/>	<b>v1.0</b>  	Published 1 month ago	
	422.88 MiB	Digest: 12b66a7	
<input type="checkbox"/>	<b>v1.0.1</b>  	Published 1 month ago	
	422.88 MiB	Digest: 1fc38e8	





# Dissemination

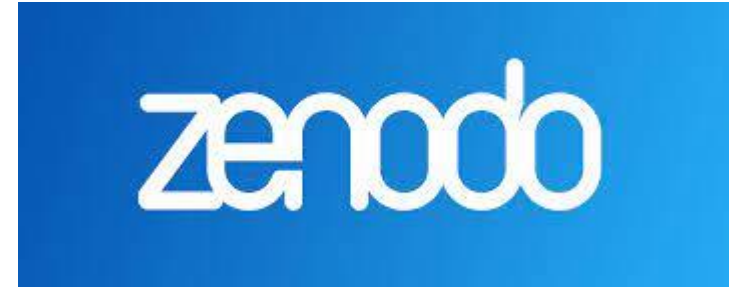
- Publishing code – Zenodo integrates with GitHub





# Dissemination

- Publishing code – Zenodo integrates with GitHub
- Publishing forecasts – AWS, DataOne, Environmental Data Initiative





# Dissemination

- Publishing code – Zenodo integrates with GitHub
- Publishing forecasts – AWS, DataOne, Environmental Data Initiative
- Metadata - <https://ecoevorxiv.org/9dgtq/>



A Community Convention for Ecological Forecasting:  
Output Files and Metadata

**AUTHORS**

Michael Dietze, R. Quinn Thomas, Jody Peters, Carl Boettiger, Alexey N Shiklomanov, Jaime Ashander



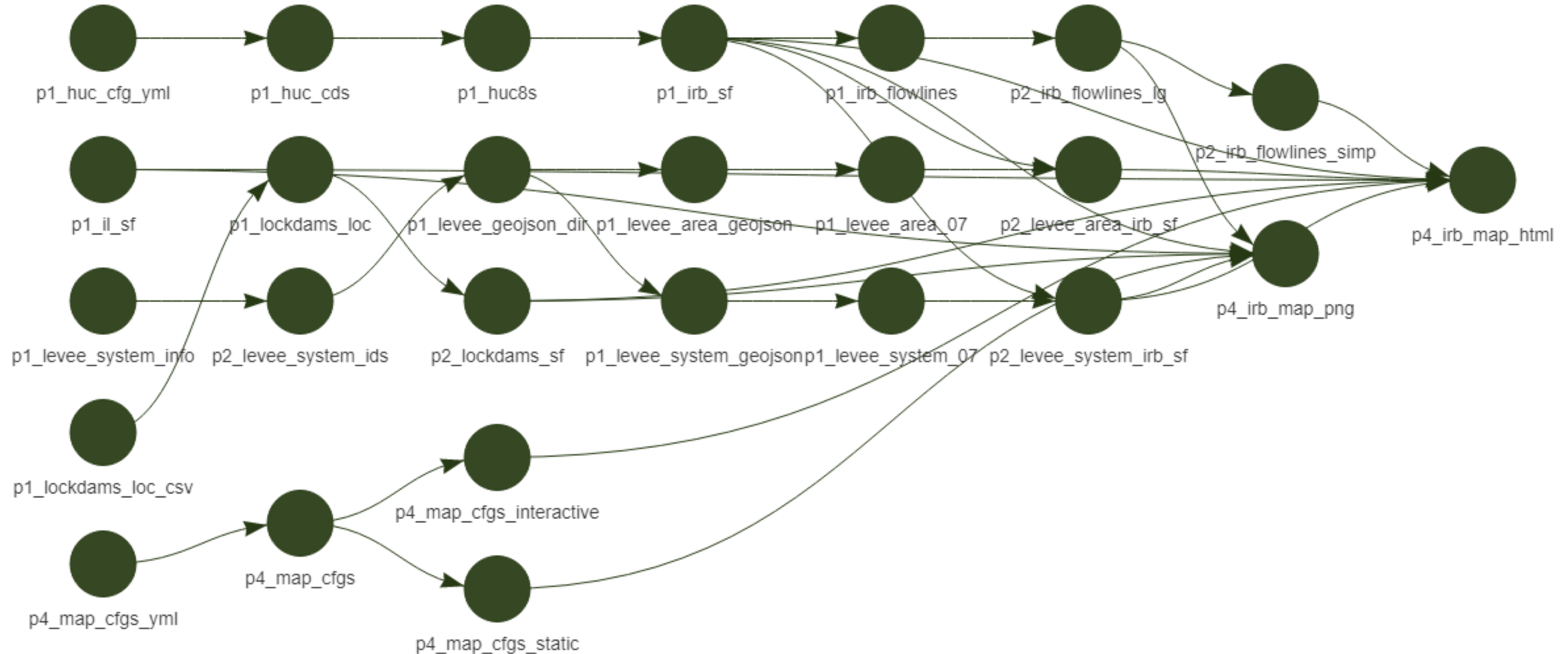
# Pipeline implementation



- R based
- Can integrate parts from other languages
- [books.ropensci.org/targets](https://books.ropensci.org/targets)

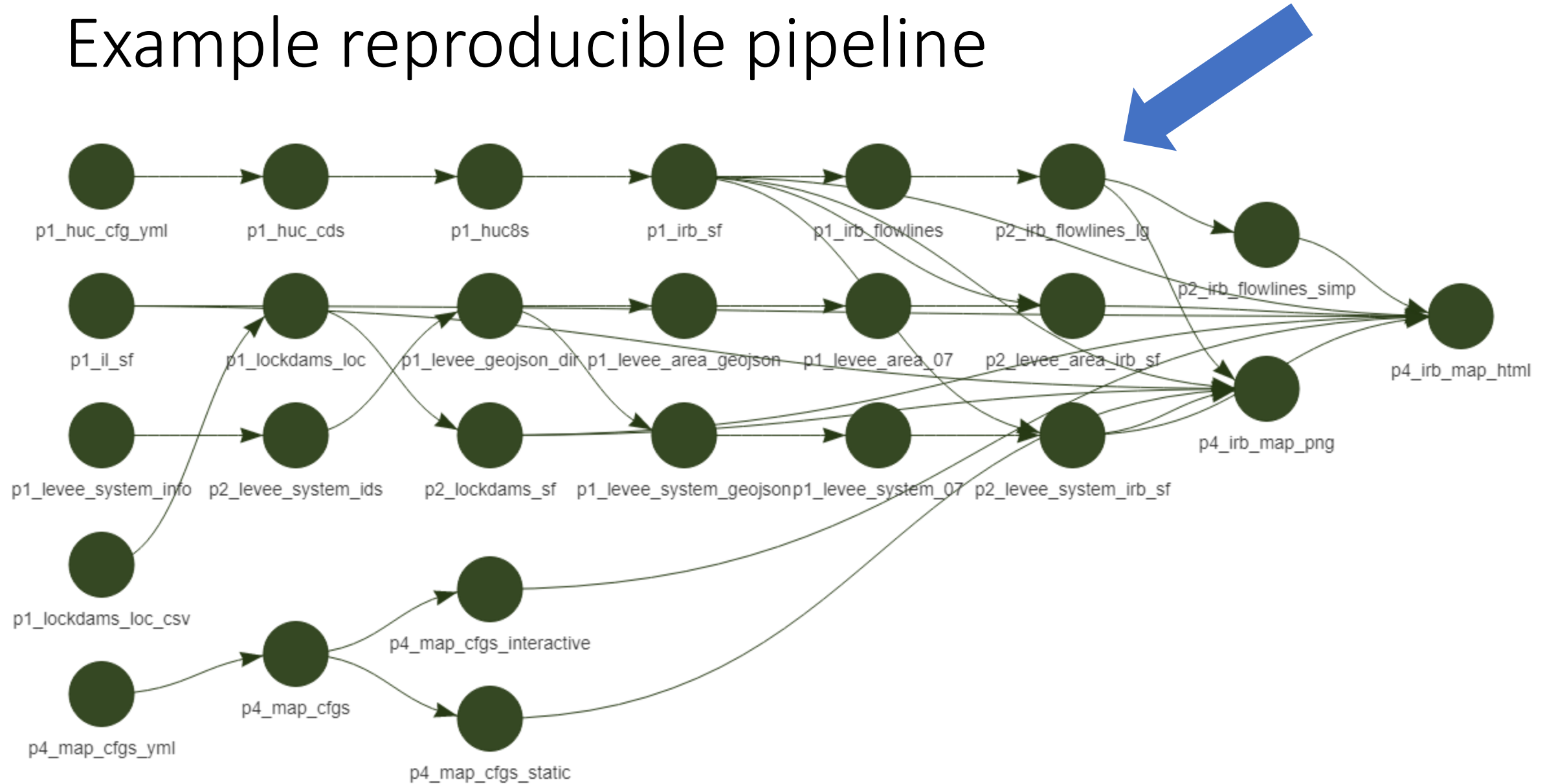


# Example reproducible pipeline



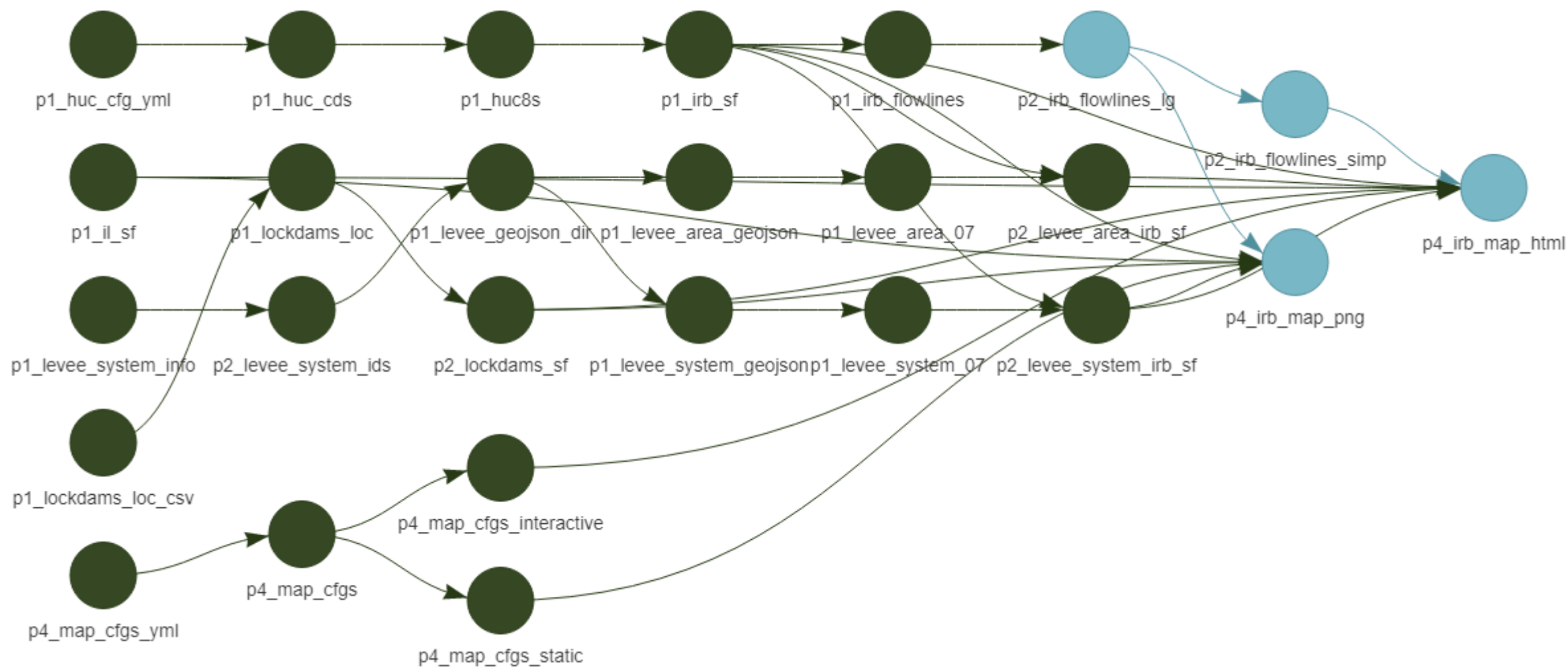


# Example reproducible pipeline



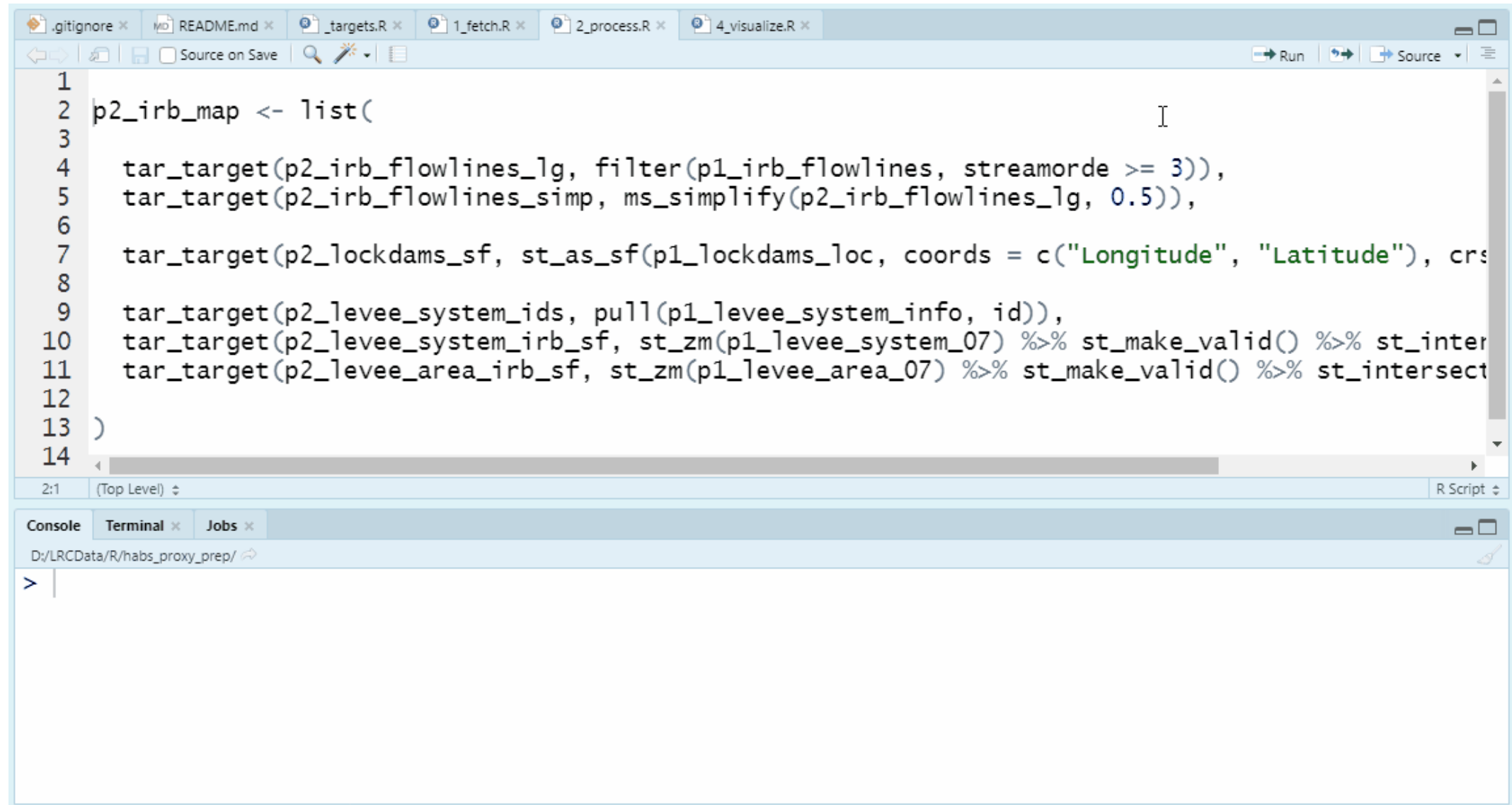


# Example reproducible pipeline





# Example reproducible pipeline



The screenshot shows the RStudio IDE interface. The top pane displays the script `4_visualize.R` with the following R code:

```
1  
2 p2_irb_map <- list(  
3  
4   tar_target(p2_irb_flowlines_lg, filter(p1_irb_flowlines, streamorde >= 3)),  
5   tar_target(p2_irb_flowlines_simp, ms_simplify(p2_irb_flowlines_lg, 0.5)),  
6  
7   tar_target(p2_lockdams_sf, st_as_sf(p1_lockdams_loc, coords = c("Longitude", "Latitude"), crs=  
8  
9   tar_target(p2_levee_system_ids, pull(p1_levee_system_info, id)),  
10  tar_target(p2_levee_system_irb_sf, st_zm(p1_levee_system_07) %>% st_make_valid() %>% st_inter  
11  tar_target(p2_levee_area_irb_sf, st_zm(p1_levee_area_07) %>% st_make_valid() %>% st_intersect  
12  
13 )  
14
```

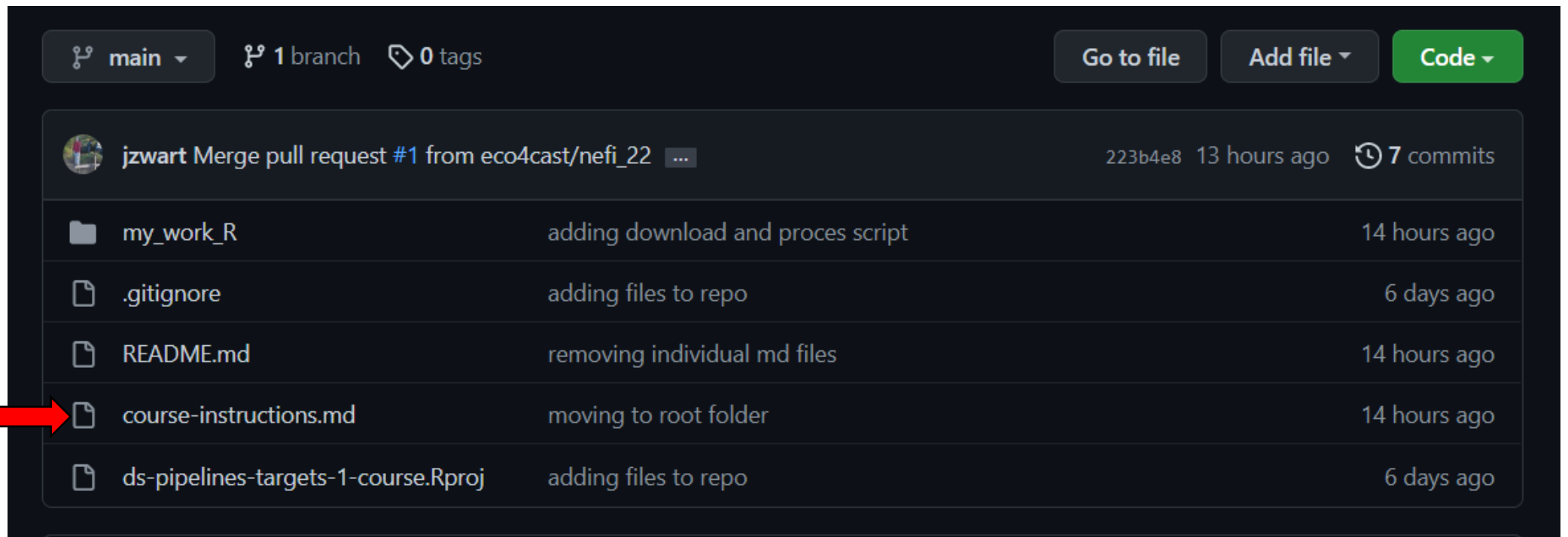
The bottom pane shows the console with the prompt `>` and the current working directory `D:/LRCDData/R/habs_proxy_prep/`.



jzward@usgs.gov

# Hands On Activity

- <https://github.com/eco4cast/ds-pipelines-targets-1-course>
- Fork to your GitHub
- Follow details in course-instruction.md





# Forecast Example

- <https://github.com/eco4cast/neon4cast-example>

```
1  on:
2    workflow_dispatch:
3    #schedule:
4    #- cron: "0 20 * * *"
5
```

The screenshot displays the GitHub Actions interface. The top navigation bar includes links for Code, Pull requests, Actions (which is highlighted), Projects, Wiki, Security, Insights, and Settings. On the left sidebar, under the 'Workflows' section, there is a 'New workflow' button and a list of workflow files, with '.github/workflows/do\_prediction...' selected. The main area is titled 'All workflows' and shows 'Showing runs from all workflows'. A search bar labeled 'Filter workflow runs' is present. Below this, a table lists '5 workflow runs'. The first two runs are visible: a successful run (green checkmark) and a failed run (red X). Both runs are for the file '.github/workflows/do\_prediction.yml' and were manually triggered by 'jzward'.

	Event ▾	Status ▾	Branch ▾	Actor ▾
✓ .github/workflows/do_prediction.yml github/workflows/do_prediction.yml #5: Manually run by jzward		19 hours ago 47m 25s		...
✗ .github/workflows/do_prediction.yml github/workflows/do_prediction.yml #4: Manually run by jzward		19 hours ago 1m 31s		...