# Link to GitHub repo for the workshop: [NEONScience/WORKSHOP-AEMON-J-DSOS-2022: This repository contains the code that will be used for the AEMON-J/DSOS workshop: An introduction to workflow management software in ecology (https://aquaticdatasciopensci.github.io/day2-bigdata/).](https://github.com/NEONScience/WORKSHOP-AEMON-J-DSOS-2022)

library(targets)

library(tidyverse)

library(lubridate)

library(neonUtilities)

use\_targets()

# Delete everything :)

# New \_targets.R file...

# Load package required to define the pipeline:

library(targets)

# Set target options:

tar\_option\_set(

# packages that targets runs by default

packages = c("tidyverse"),

# Default storage

format = "rds"

)

# Load the R script with your custom functions:

source("R/functions.R")

# Define the targets in the workflow:

list(

tar\_target(

name = ,

command =

)

)

#This gets typed into the console:

tar\_helper\_raw(path = "R/functions.R")

#Now there is a R folder with functions.R script

# Lets pull some data from the NEON data portal data/neonscience.org using the neonUtilities package and loadByProduct() function

# link to data product landing page: [NEON | Water quality (neonscience.org)](https://data.neonscience.org/data-products/DP1.20288.001)

# Setting up your NEON token: [Using an API Token when Accessing NEON Data with neonUtilities | NSF NEON | Open Data to Understand our Ecosystems (neonscience.org)](https://www.neonscience.org/resources/learning-hub/tutorials/neon-api-tokens-tutorial)

list(

tar\_target(

name = neon\_waq,

command = {

# Download the water quality dataset

waq <- loadByProduct(dpID = "DP1.20288.001",

site = “COMO”,

startdate = “2022-04”,

enddate = “2022-04”,

package = “expanded”,

release = “current”,

#token = Sys.getenv(“NEON\_TOKEN”),

check.size = FALSE)

# Isolate the table of interest

return(waq$waq\_instantaneous)

},

packages = c(“tidyverse”, “neonUtilities”))

)

# Everything we’ve written so far with nicer formatting:

*# Load packages required to define the pipeline:*

**library**(targets)

*# Set target options:*

tar\_option\_set(

*# Packages that your targets need to run*

packages = c("tidyverse"),

*# Default storage format*

format = "rds"

)

*# Load the R script with your custom functions:*

**source**("functions.R")

*# Define the targets in the workflow:*

list(

*# Water quality*

tar\_target(name = neon\_waq,

command = {

*# Download the water quality dataset for COMO*

waq <- loadByProduct(dpID = "DP1.20288.001",

site = "COMO",

startdate = "2022-04",

enddate = "2022-04",

package = "expanded",

release = "current",

*# token = Sys.getenv("NEON\_TOKEN"),*

check.size = F)

*# Isolate the table of interest*

**return**(waq$waq\_instantaneous)

},

packages = c("tidyverse", "neonUtilities"))

)

# Save the \_targets.R file or other files/scripts that are part of the workflow before running the following:

# Type into console to see if the workflow is working:

tar\_make()

# We’re going to add two more targets, can copy and paste the first one as a place to start, changing dpID to download nitrate in surface water ([NEON | DP1.20033.001 (neonscience.org)](https://data.neonscience.org/data-products/DP1.20033.001)) and Continuous discharge ([NEON | Data Product (neonscience.org)](https://data.neonscience.org/data-products/DP4.00130.001)) in the new tar\_target entries. This is what the new list should look like with the three data pulls.

*# Define the targets in the workflow:*

list(

tar\_target(name = neon\_waq,

command = {

*# Download the water quality dataset for COMO*

waq <- loadByProduct(dpID = "DP1.20288.001",

site = "COMO",

startdate = "2022-04",

enddate = "2022-04",

package = "expanded",

release = "current",

*# token = Sys.getenv("NEON\_TOKEN"),*

check.size = F)

*# Isolate the table of interest*

**return**(waq$waq\_instantaneous)

},

packages = c("tidyverse", "neonUtilities")),

*# Nitrate in surface water*

tar\_target(name = neon\_nsw,

command = {

*# Download the nitrate dataset for COMO*

nsw <- loadByProduct(dpID = "DP1.20033.001",

site = "COMO",

startdate = "2022-04",

enddate = "2022-04",

package = "expanded",

release = "current",

*# token = Sys.getenv("NEON\_TOKEN"),*

check.size = F)

*# Isolate the table of interest*

**return**(nsw$NSW\_15\_minute)

},

packages = c("tidyverse", "neonUtilities")),

*# Continuous discharge*

tar\_target(name = neon\_csd,

command = {

*# Download the continuous discharge dataset for COMO*

csd <- loadByProduct(dpID = "DP4.00130.001",

site = "COMO",

startdate = "2022-04",

enddate = "2022-04",

package = "expanded",

release = "current",

*# token = Sys.getenv("NEON\_TOKEN"),*

check.size = F)

*# Isolate the table of interest*

**return**(csd$csd\_continuousDischarge)

},

packages = c("tidyverse", "neonUtilities")),

# Added in after building the function below in this notes doc

#Clean and join fDOM with csd

tar\_target(combined\_fdom\_csd,

combine\_fdom\_csd(neon\_waq = neon\_waq,

Neon\_csd = neon\_csd),

packages = c("tidyverse", "lubridate"))

)

# Run in the console to check things:

tar\_validate()

# Warnings about R versions are ok, if things look good (i.e. no errors) run the following in the console:

tar\_make()

# It didn’t re-download the waq data since targets knew that it was already downloaded, pretty cool!

#When we come back from break we will include some data analysis in the pipeline

# In files tab, click on R folder and then open the functions.R file (it will probably just have a NULL in it, which you can delete). Now we’re going to write what the function does…

function\_name <- **function**(argument){

*# Body code*

}

*# Load in the results of the target we created earlier (using the console)*

tar\_load(neon\_waq)

tar\_load(neon\_csd)

# Here is the final function

# If you have any specific questions about NEON data that aren’t answered in the documentation on the data product landing pages, feel free to contact us ([Contact Us | NSF NEON | Open Data to Understand our Ecosystems (neonscience.org)](https://www.neonscience.org/about/contact-us)). Also, check out the “introduction links” section on the GitHub page.

combine\_fdom\_csd <- **function**(neon\_waq, neon\_csd){

*# Downstream position*

waq\_down <- neon\_waq %>%

filter(horizontalPosition == "102")

*# Isolate the columns containing Fluorescent Dissolved Organic Matter (fDOM) data*

*# and remove any with quality flags, then round to make joining with CSD easier*

fdom <- waq\_down %>%

*# Pull out only fDOM-related variables and the time stamp*

select(endDateTime, contains("fDOM")) %>%

*# Remove data that failed quality checks*

filter(fDOMFinalQF == 0) %>%

*# Round times for easier joins later*

mutate(endDateTime = round\_date(endDateTime, unit = "1 minute"))

*# Remove quality flags and then round*

csd <- neon\_csd %>%

filter(dischargeFinalQF == 0) %>%

mutate(endDate = round\_date(endDate, unit = "1 minute"))

*# Join the datasets, subset to dates of high flow events*

fdom\_csd <- inner\_join(x = fdom,

y = csd,

by = c("endDateTime" = "endDate")) %>%

filter(endDateTime > "2022-04-21",

endDateTime < "2022-04-25")

**return**(fdom\_csd)

}

# Add the following to the \_targets.R file

#Clean and join fDOM with csd

tar\_target(combined\_fdom\_csd,

combine\_fdom\_csd(neon\_waq = neon\_waq,

neon\_csd = neon\_csd),

packages = c("tidyverse", "lubridate"))

# Check things with tar\_make() in the console to make sure the pipeline is working

# Make sure to save all the files :)

# Now we’re going to combine the nitrate and csd data

tar\_load(neon\_nsw) # neon\_csd is already loaded

combine\_nsw\_csd <- **function**(neon\_nsw, neon\_csd){

*# Remove data with quality flags, then round to make joining with CSD easier*

nsw <- neon\_nsw %>%

filter(finalQF == 0) %>%

mutate(endDateTime = round\_date(endDateTime, unit = "1 minute"))

*# Remove quality flags and then round*

csd <- neon\_csd %>%

filter(dischargeFinalQF == 0) %>%

mutate(endDate = round\_date(endDate, unit = "1 minute"))

*# Join the datasets (every 15 min. the times will match up), subset to dates*

*# of high flow events*

nsw\_csd <- inner\_join(x = nsw,

y = csd,

by = c("endDateTime" = "endDate")) %>%

filter(endDateTime > "2022-04-21",

endDateTime < "2022-04-25")

**return**(nsw\_csd)

}

# Add the following to the \_targets.R file to add in this step

tar\_target(combined\_nsw\_csd,

combine\_nsw\_csd(neon\_nsw, neon\_csd),

packages = c("tidyverse", "lubridate"))

#Use the following in the console to take a “look” at the workflow

tar\_visnetwork()

# No more functions for the day, congrats on finishing that!

# Next we’re going to add two targets to plot the data

tar\_load(combined\_fdom\_csd)

tar\_target(fdom\_cq\_plot,

{

ggplot(data = combined\_fdom\_csd) +

geom\_point(aes(x = maxpostDischarge,

y = fDOM,

color = endDateTime)) +

ggtitle("COMO fDOM vs. Q") +

xlab("Q (L/s)") +

ylab("fDOM (QSU)") +

scale\_color\_viridis\_c() +

theme\_bw() +

theme(legend.position = "none")

})

# No additional packages needed since tidyverse includes ggplot

# Last target to add:

tar\_load(combined\_nsw\_csd)

tar\_target(nsw\_cq\_plot,

{

ggplot(data = combined\_nsw\_csd) +

geom\_point(aes(x = maxpostDischarge,

y = surfWaterNitrateMean,

color = endDateTime)) +

ggtitle("COMO NO3-N vs. Q (subset)") +

xlab("Q (L/s)") +

ylab("NO3-N (uM)") +

scale\_color\_viridis\_c() +

theme\_bw() +

theme(legend.position = "none")

})

# In the console, run:

tar\_glimpse()

tar\_make() # Everything should be run already

# Take a look at plots

tar\_load() or tar\_read()

# If you want to perform a step that the target workflow things is done…

tar\_invalidate(neon\_waq)

*# Check for outdated steps*

*# All of the downstream steps are also set to outdated if an upstream step is invalidated*

tar\_outdated()

tar\_make()

# Extreme!

tar\_destroy()