Domain logic vs. reactive structure

- **Domain logic** is the essential analysis that our app embodies (loading, data manipulation, modeling, visualization)

- **Reactive structure** is the Shiny-specific server code that makes that analysis interactive

Shiny app development equals adding reactive structure to your domain logic.

Now we want to take a Shiny app and **extract the domain logic** back out of the reactive structure.

Domain logic vs. reactive structure

- **Domain logic** is the essential analysis that our app embodies (loading, data manipulation, modeling, visualization)
- **Reactive structure** is the Shiny-specific server code that makes that analysis interactive

Shiny app development equals adding reactive structure to your domain logic.

Now we want to take a Shiny app and **extract the domain logic** back out of the reactive structure.

Converting R script to Shiny

```
downloads <- cranlogs::cran_downloads("ggplot2",
  from = Sys.Date() - 365, to = Sys.Date())

downloads_rolling <- downloads %>%
  mutate(count = zoo::rollapply(count, 7, mean, fill = "extend"))

ggplot(downloads_rolling, aes(date, count)) +
  geom_line() +
  ggtitle("Seven day rolling average")
```