

# Web front-ends with re-frame

---

<https://github.com/danielytics/reframe-talk>

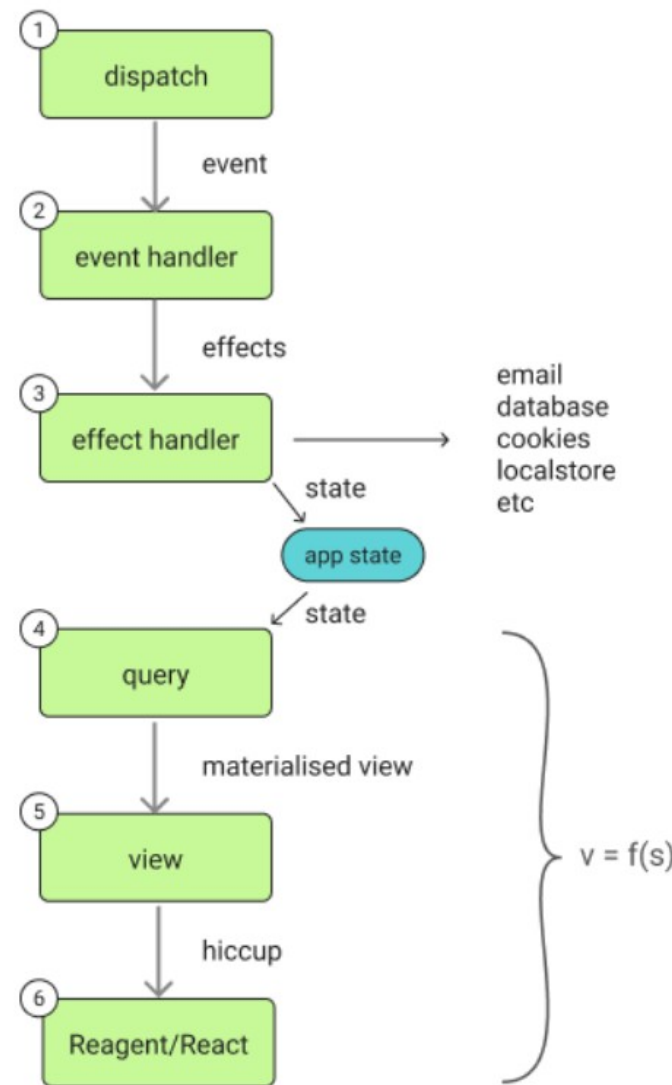
# What is re-frame?

*re-frame is a pattern for writing SPAs in ClojureScript, using Reagent.*

- Kinda like Redux
- Also kinda like Elm
- It gives you..
  - an event loop
  - subscriptions (a reactive view of your data)
  - a model to keep your logic as pure as possible
  - happiness

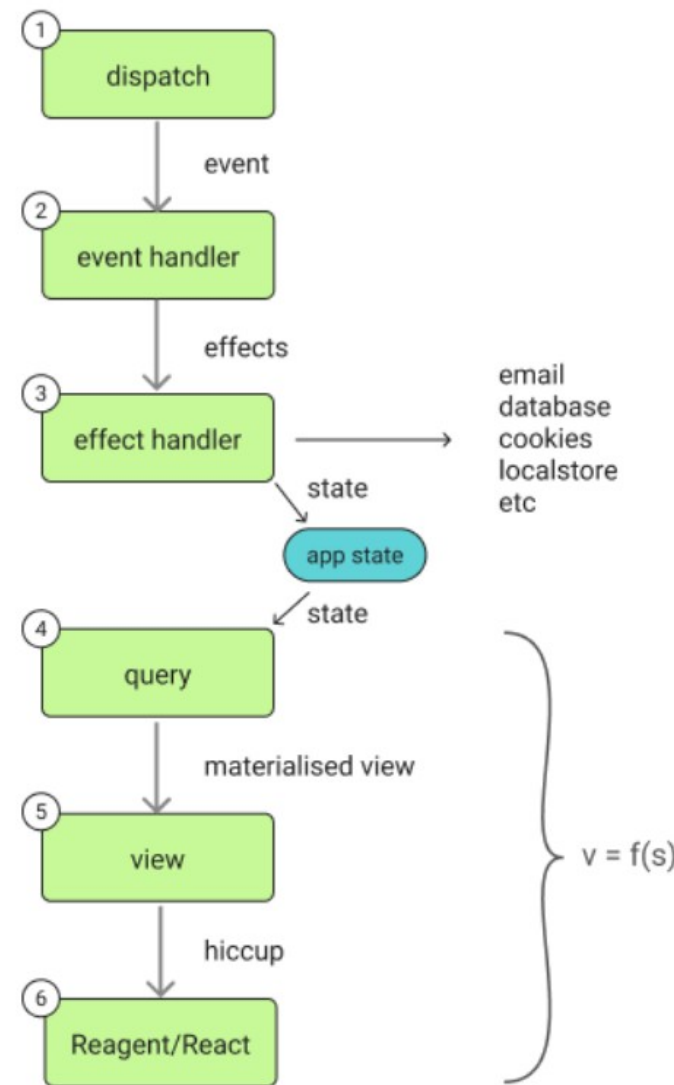
# Application Composition

- Pure-functional event handlers (3)
- Reagent “views” (4)
- Pure-functional “queries” to your data (4)



# Application Composition

- User triggers browser event
- Browser event callback dispatches re-frame event
- Event handler generates collection of “effects” to apply
- Data changes trigger queries to update
- Query updates trigger rendering



# Interacting with the world

- Event handlers are pure...
- They tell what should be done (“effects”)
- Effect handler actually does it (impure)
- Event handlers can also request impure data...
- “coeffects” get this data for them (eg, current time)

event + coeffects  $\rightarrow$  effects

# Subscriptions

- Kinda like database queries
- Describe what application state you want
- A layer of indirection between your actual data
- What not where/how
- Are reactive, if their results change, your view will re-render
- Allow for optimisations



