

# Clojure Web Development

OnyCloud

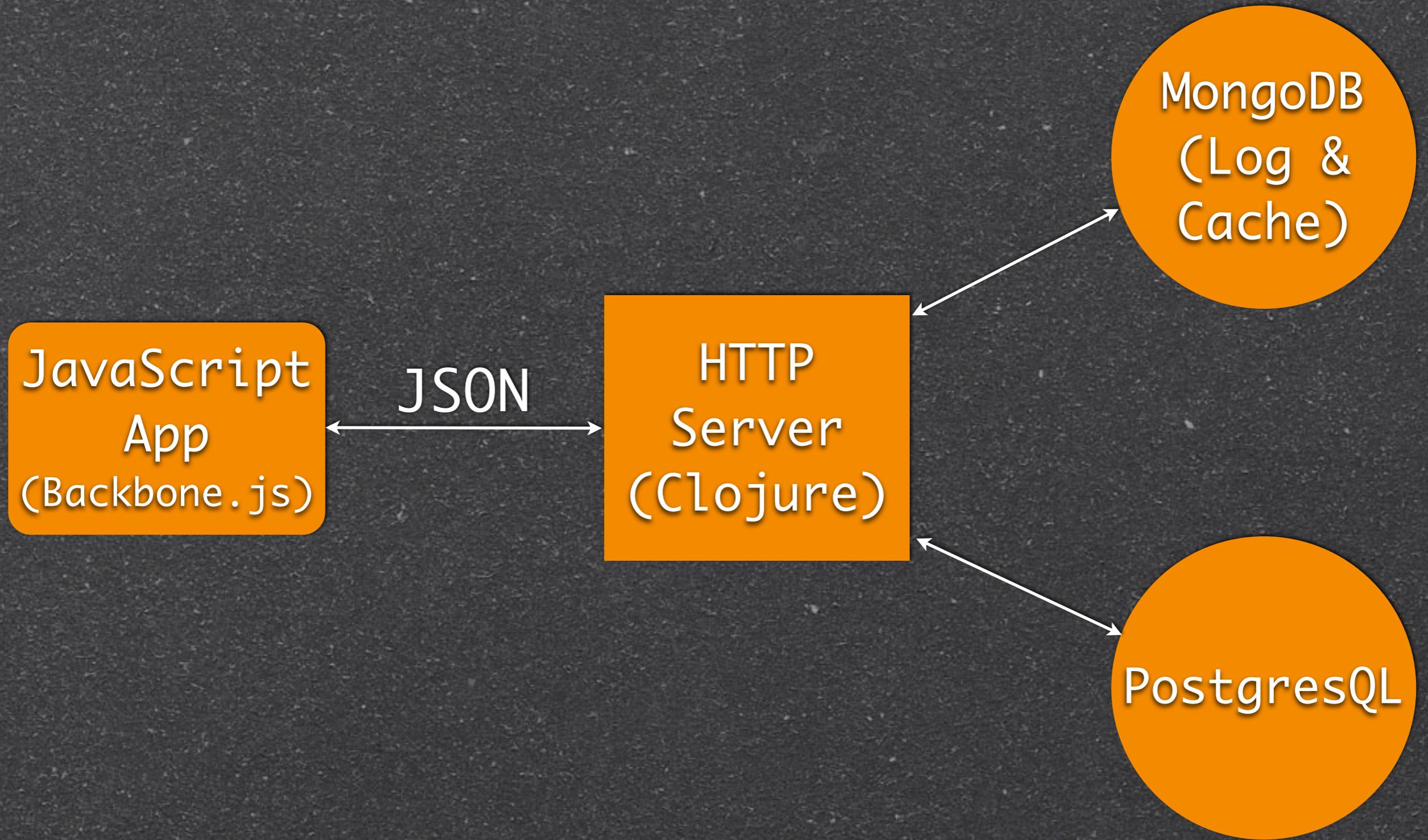
江宏

# Example - Trakr

- <https://trakrapp.com/>
- Web-based project management system
- A product created for our own needs
- Requirements
  - Feature-complete for software devs
    - Modern friendly UI
    - Affordable

Demo

# Architecture



# HTTP Server Structure



The Compojure Framework

# Routes

- Mapping: Request -> Handler:

```
(defroutes app-routes
  (context "/users" []
    (GET "/new" [] accounts/show-signup)
    (POST "/" [email] (accounts/new email)))

  ;; GET /users/new  -> accounts/show-signup
  ;; POST /users/   -> accounts/new
```

# Handlers

- Request in, response out.

```
(defn say-hello [req]
  (let [name (-> req :params :name)]
    {:status 200
     :body (str "Hello " name "!")})
```

# Middleware

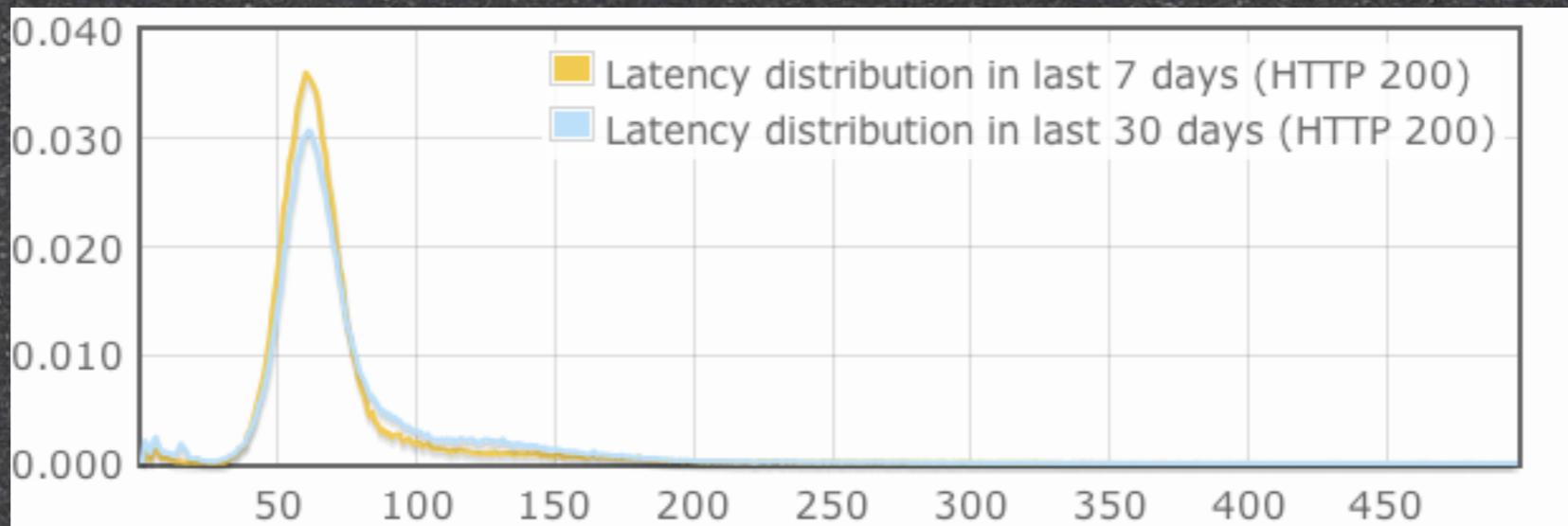
- Transform: Handler -> New Handler

```
(defn wrap-failsafe [handler]
  (fn [request]
    (try (handler request)
         (catch Exception e
           (.printStackTrace e)
           {:status 500
            :body "An error occurred."}))))
```

# Testing

- Unit testing with `clojure.test` and `clojure.contrib.mock`
- Integration testing using Watir
- Demo

# Performance



- No optimization tricks (type hints, transients, etc.)
- Average latency ~ 70ms
- Long tail
- Database performance can be improved (indices).

# Lessons Learned

- Good:
  - High productivity
  - Easy to test (dynamic binding)
  - Relatively easy learning curve

# Lessons Learned

- Bad:
  - Ugly stacktraces.
  - Exposes too much Java class hierarchy.
  - PersistentMap, PersistentStructMap, struct\_map ...
  - (contains? (transient #{:a}) :a)

Thank You