Web Applications in Clojure

Munich Clojurians

February 7, 2017

Joy Clark

Consultant @ innoQ

joy.clark@innoq.com

@iamjoyclark

@iamjoyclark

Clojure

- Lisp Variant for the JVM
- Functional Language
- Dynamically Typed
- Immutable Data Structures
- Simplicity - Separation of Data and Behavior

@iamjoyclark

Data Structures

3 3.14 3/2

"Hello World" \a #"Ch.*se"

foo :first

("Hello" :first)

{ :name "Joy"
  :company "innoQ" }

[3 4 3]

#{3 4 3}

@iamjoyclark

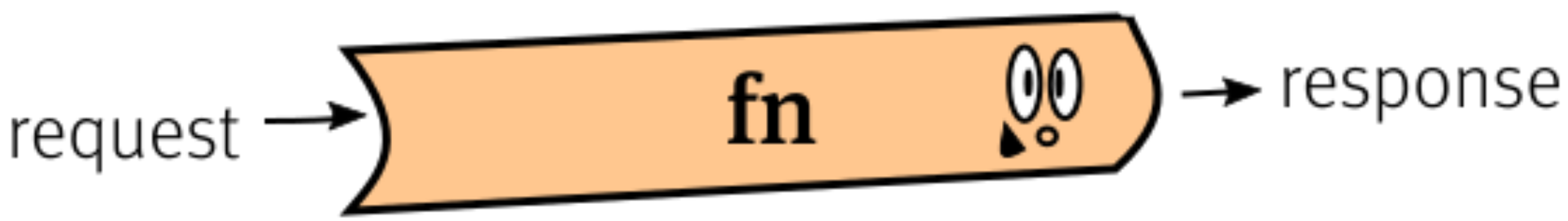
Functions

```
(+ 1 2)
› 3

(:city {:name “innoQ”
        :city “Munich”})
› “Munich”

(map inc [1 2 3])
› (2 3 4)
```

Web Applications

@iamjoyclark

What's a Web Application?



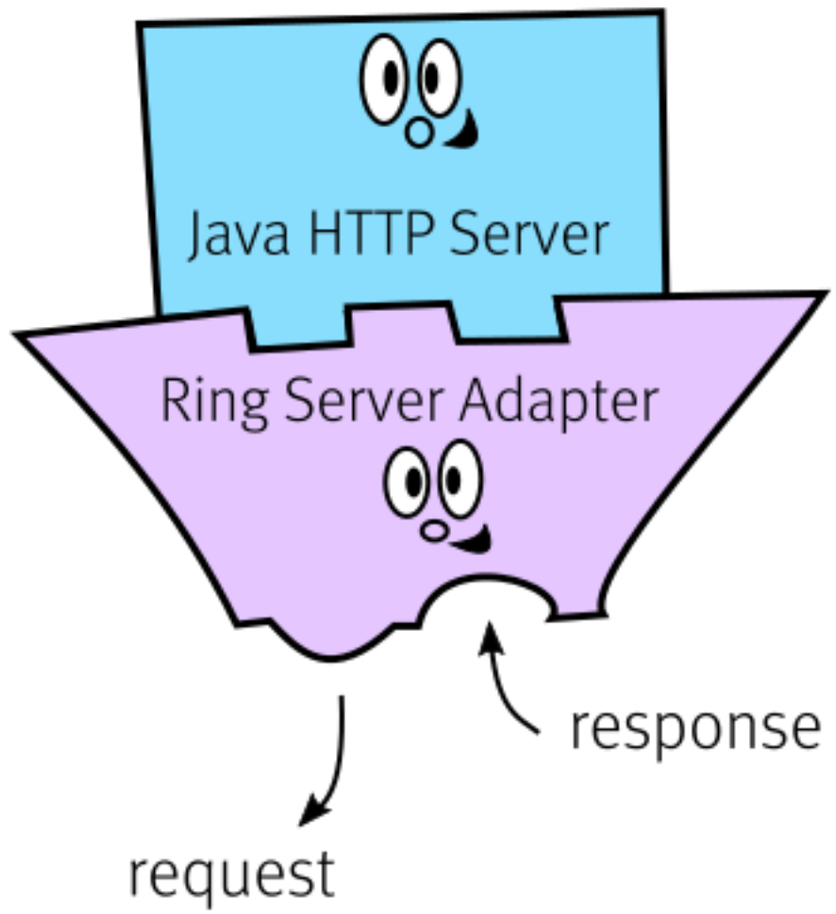
Ring

https://github.com/ring-clojure/ring

- HTTP Server abstraction
- Request & Response are data
- Web App is a function

@iamjoyclark

Ring



@iamjoyclark

Ring: Request

```
{:uri “/”
 :params { :name “Joy” }
 :request-method :get
 :headers { … }
 …}
```

Ring: Response

```
{:status 200
 :headers { … }
 :body “Hello!”}
```

@iamjoyclark

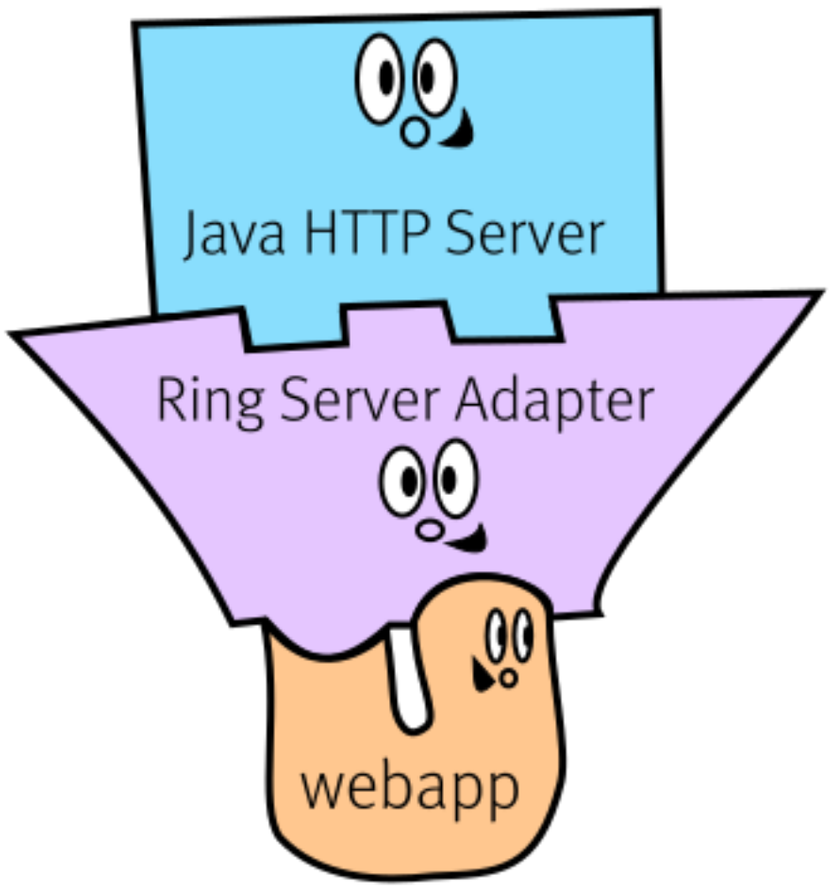
Ring: Handler

```
(defn example-app [request]
  (let [name (get-in request [:params :name])]
    {:status 200
     :headers {“Content-Type” “text/plain”}
     :body (str “Hello, ” name “!”)}))
```

@iamjoyclark

Ring: Adapter

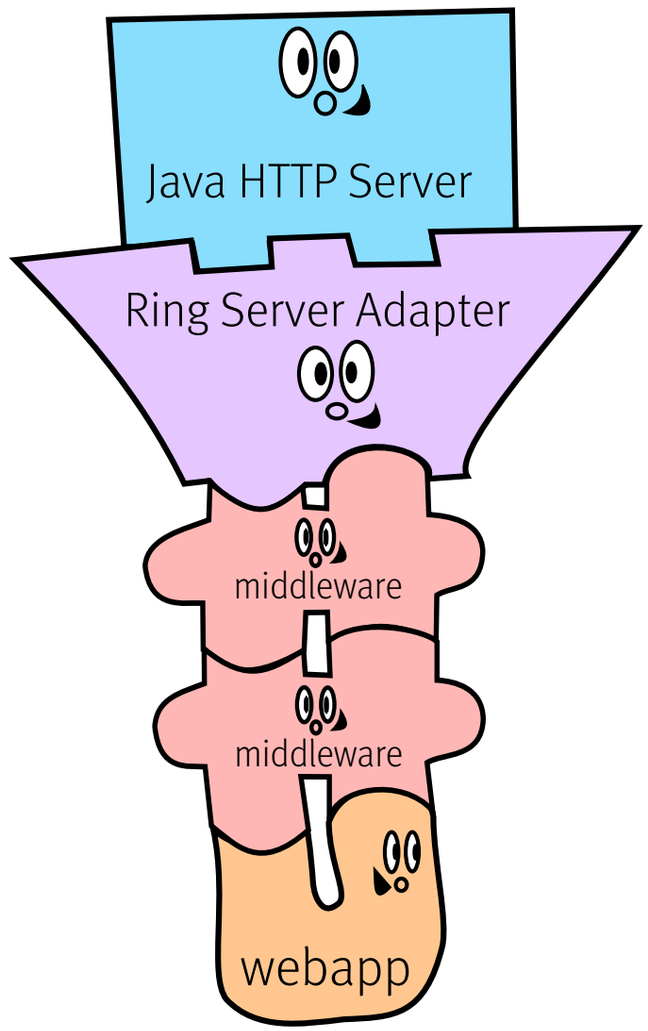
- Jetty
- Servlet
- http-kit
- Tomcat
- ...



@iamjoyclark

Ring: Middleware

```
(defn wrap-logging [handler]
  (fn [request]
    (print request)
    (let [response (handler request)]
      (print response)
      response)))
```



Ring: Middleware

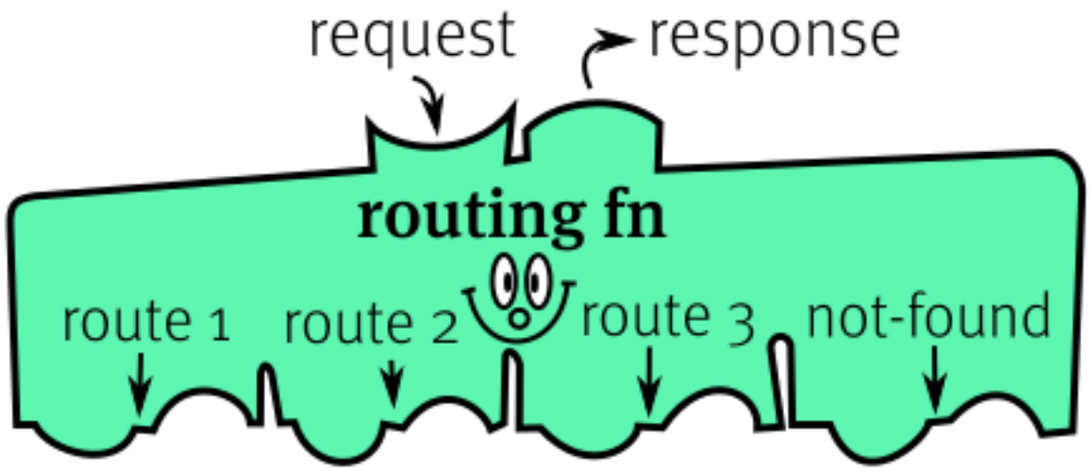
- https://github.com/ring-clojure/ring-defaults
- https://github.com/ring-clojure/ring-json

Compojure

https://github.com/weavejester/compojure

@iamjoyclark

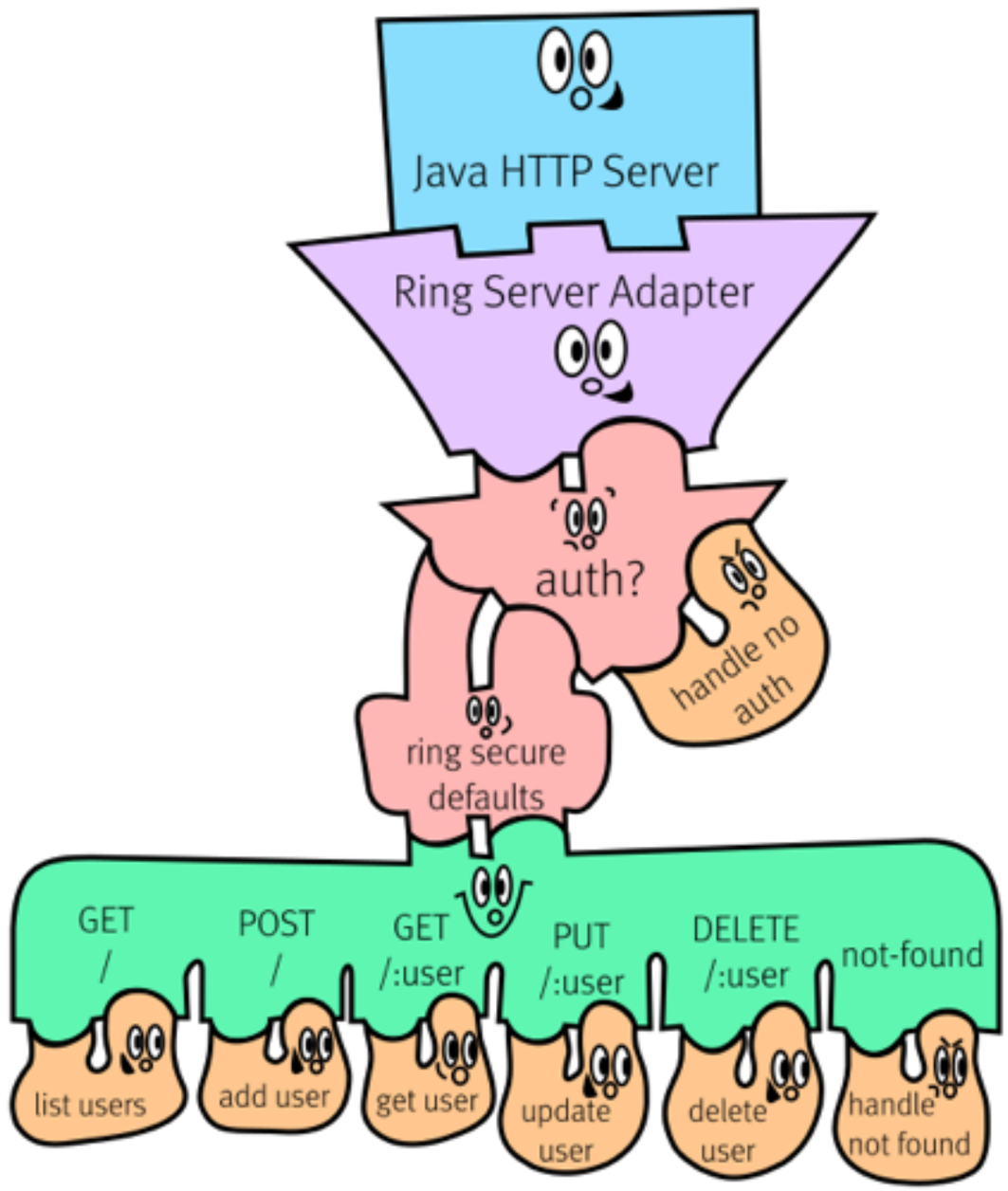
Compojure


request
response
routing fn
route 1
route 2
route 3
not-found

Compojure: Handler

```
(defroutes app-routes
  (GET "/" request (list-users request))
  (POST "/" {params :params :as request}
    (add-user request params))
  (GET "/:username" [username :as request]
    (get-user request username)))
```

Putting it all together



@iamjoyclark

HTML Templating

@iamjoyclark

Hiccup

```
<div id="foo">
  <span>bar</span>
</div>
```

```
[:div {:id "foo"}
 [:span "bar"]]
```

https://github.com/weavejester/hiccup

@iamjoyclark

Hiccup

Make sure to escape HTML when using Hiccup!

```
(defn render-html [input]
  (hiccup.util/escape-html
    (hiccup.core/html input)))
```

@iamjoyclark

Hiccup: Alternatives

- https://github.com/cgrand/enlive
- https://github.com/yogthos/Selmer

Summary

@iamjoyclark

Summary

- Functional Language ideal for Web Apps
- Clojure usage is increasing
- Libraries vs. Frameworks
- Community is healthy and welcoming

@iamjoyclark

Interesting Libraries

- https://github.com/technomancy/leiningen
- https://github.com/weavejester/environ
- https://github.com/krisajenkins/yesql

“Frameworks”

- https://github.com/weavejester/duct
- http://www.luminusweb.net
- https://github.com/otto-de/tesla-microservice
- https://github.com/metosin/compojure-api

Thank you!

Tutorial for Getting Started:

https://github.com/innoq/tutorial-clj-webapp

www.innoq.com

Joy Clark | @iamjoyclark | joy.clark@innoq.com