

Дмитрий Бушенко
d.bushenko@gmail.com

Применение Clojure в web-разработке





Чем Clojure пугает менеджеров?

- Трудно ли найти программистов на Clojure?
- Насколько у Clojure развитая инфраструктура?

Рейтинг ТЮВЕ за февраль 2013

9	9	=	Perl	2.252%	-0.68%
10	12	↑↑	Ruby	1.752%	+0.19%
11	10	↓	JavaScript	1.423%	-1.04%
12	16	↑↑↑↑	Visual Basic .NET	1.007%	+0.21%
13	13	=	Lisp	0.943%	+0.04%
14	15	↑	Pascal	0.932%	+0.12%
15	11	↓↓↓↓	Delphi/Object Pascal	0.886%	-1.08%
16	14	↓↓	Transact-SQL	0.773%	-0.07%
17	75	↑↑↑↑↑↑↑↑↑↑	Bash	0.741%	+0.61%
18	26	↑↑↑↑↑↑↑↑	MATLAB	0.648%	+0.15%

Сравним позиции

25	Fortran	0.462%
26	R	0.461%
27	LabVIEW	0.458%
28	Scheme	0.448%
29	Scratch	0.411%
30	ABAP	0.389%
31	Logo	0.385%
32	Prolog	0.341%
33	Haskell	0.329%
34	Erlang	0.329%
35	Scala	0.327%

Технологии для веб-разработки

- Сервера Java-приложений (jetty, tomcat, jboss, etc.)
- Сервлеты в функциональном стиле
- DSL для маршрутизации
- DSL для доступа к БД
- DSL вместо шаблонизатора
- Clojure на сервере и на клиенте

Web-стек Clojure

Enlive, Dommy

ClojureScript

Hiccup, Clostache, Enlive, clj-soy

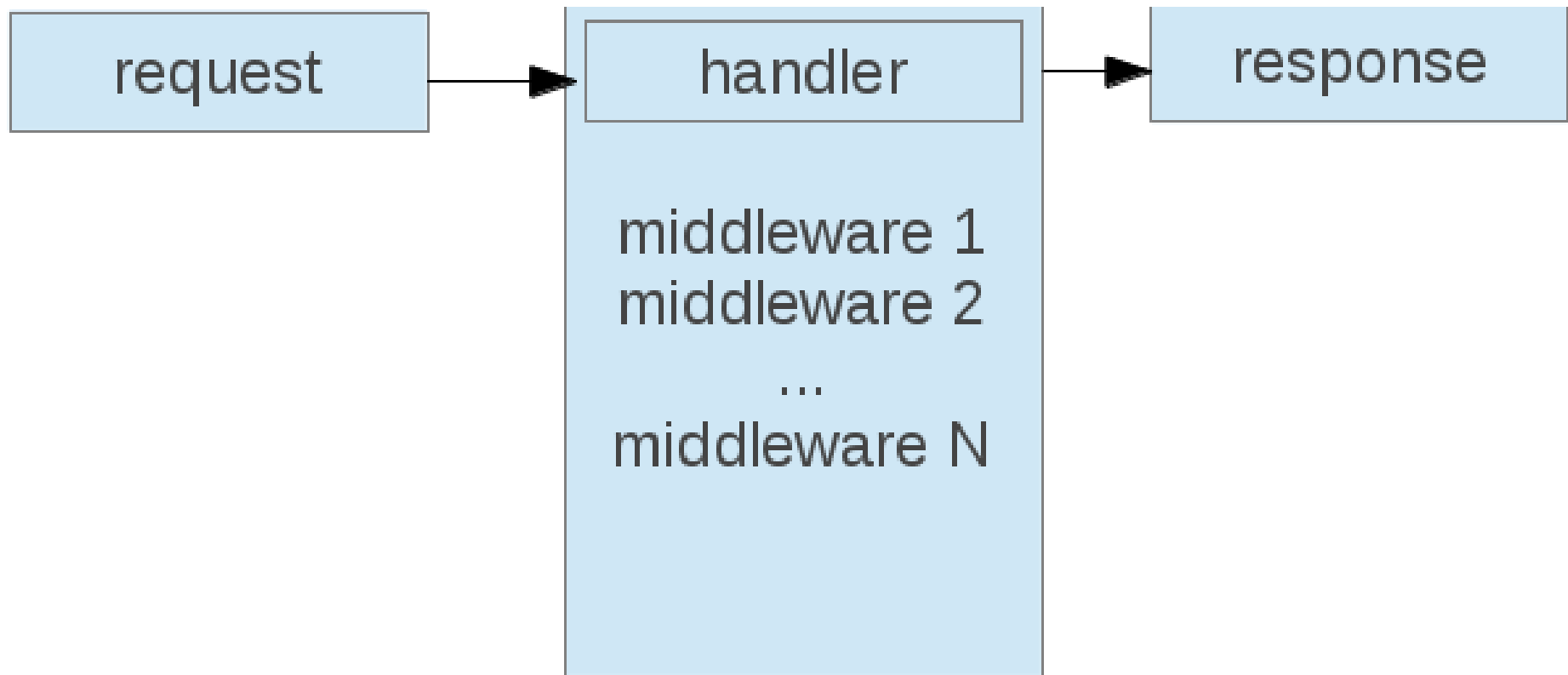
Korma, ClojureQL, JDBC, Monger

Compojure, Moustache, lib-noir

Ring

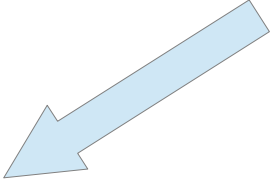
Jetty, Tomcat, Immutant

Ring-сервлет



Сервлет в функциональном стиле

```
{:headers  
 :query-string  
 :body}
```



```
(defn what-is-my-ip [request]  
  {:status 200  
   :headers {"Content-Type" "text/plain"}  
   :body (:remote-addr request)})
```


Просто, не так ли?

```
(defroutes app-routes
  (GET "/users" [] (list-users))
  (GET "/user/:id" [id] (view-user id))
  (POST "/user/delete" {params :params}
        (delete-user (:id params)))
  (POST "/user/create" {params :params}
        (create-user params))
  (POST "/user/update" {params :params}
        (update-user params)))
```

SQL? Еще проще!

```
(select users  
  (where {:login login}))
```

```
(update users  
  (where {:id id})  
  (set-fields {:password_hash p-hash  
               :admin is-admin}))
```

Html? Ну вы уже поняли

```
(:html
```

```
  (:head (:title "Add two numbers"))
```

```
  (:body
```

```
    (if (and a b)
```

```
      (htm (:p (str (+ a b))))))
```

```
  (:form :method :get
```

```
    (:input :type :text :name "a")
```

```
    (:input :type :text :name "b")
```

```
    (:input :type :submit))))
```

Enlive/Enfocus

```
(defsnippet create-close-buttons  
"/html/userslist.html" [:#create-close]  
  [create-fn]  
  [:#create-btn] (set-attr :onclick create-fn))
```

```
(defsnippet create-user-view  
"/html/userslist.html" [:#create-user-view] [])
```

JavaScript? ClojureScript!

```
(ns hello-clojurescript  
  (:use [jquery.core :only [$ delegate toggle]]))
```

```
(def $body ($ :body))
```

```
(delegate $body :#clickable :click  
  (fn [e]  
    (toggle ($ :#toggle))))
```

Clojure vs ...

Слой	Clojure	Java	Ruby
Маршрутизатор	Clojure	Аннотации XML	Ruby
База данных	Clojure	ORM HQL JPQL	ORM SQL
Шаблонизатор	Clojure	JSP JSF	ERB Haml
Клиентский код	Clojure	JavaScript	JavaScript CoffeeScript
Шаблонизатор на клиенте	Clojure	Весь JS-зоопарк	Весь JS-зоопарк

Сюжуре на всех уровнях web-стека

- Нет необходимости интегрировать разные технологии
- Одни и те же типы данных на всех уровнях
- Отпадает необходимость в дополнительных протоколах (soap, json и т.д.)
- Повышается возможность повторного использования кода

Выводы

- Clojure обладает всей необходимой инфраструктурой для построения современных веб-приложений
- Инфраструктура Clojure своеобразная и отличается от привычных веб-стеков в Java и Ruby
- Решения на Clojure выглядят просто и элегантно