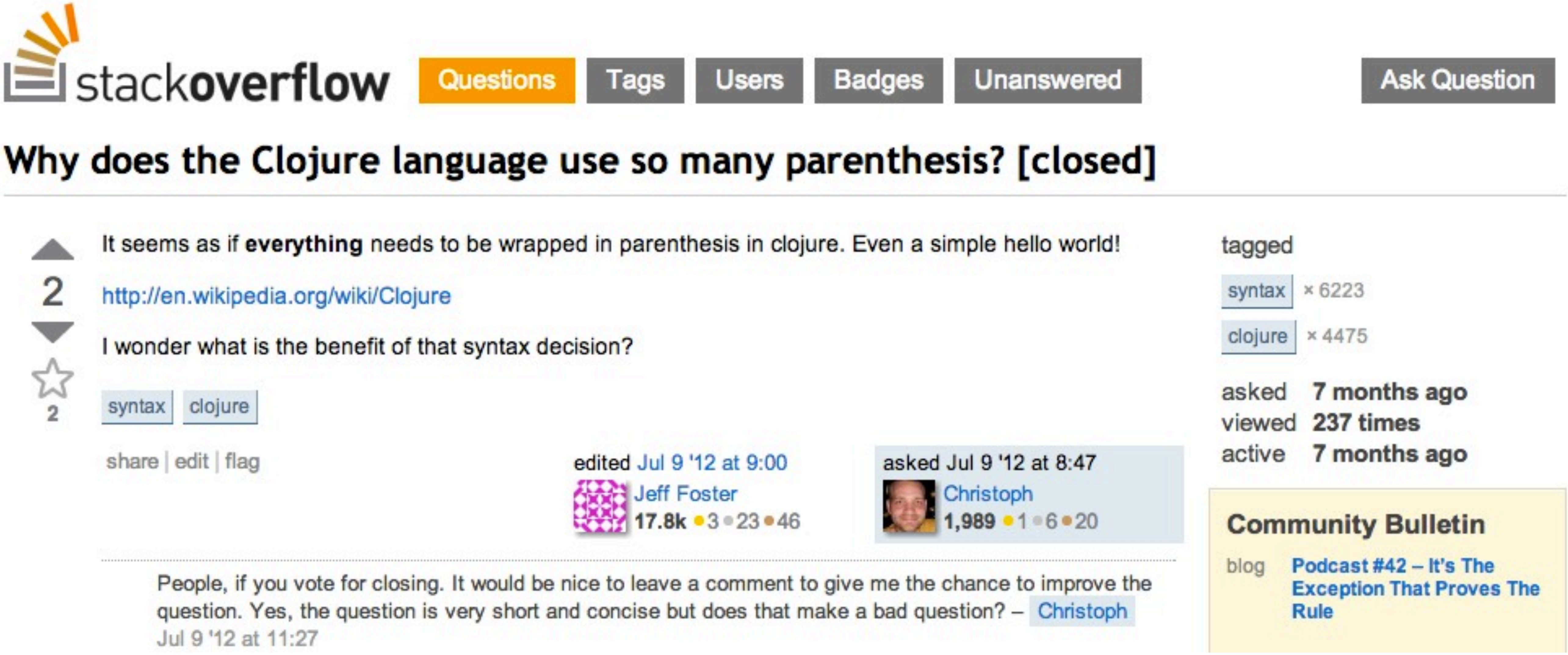
Clojurescript

Guillermo Winkler
@guilespi

Clojure?

- Es un dialecto de Lisp
- Dinámico
- Funcional
    - Énfasis en inmutabilidad y concurrencia
- Hosted en la JVM
    - Compila a código nativo
- No es OO



Why does the Clojure language use so many parenthesis? [closed]

It seems as if **everything** needs to be wrapped in parenthesis in clojure. Even a simple hello world!

http://en.wikipedia.org/wiki/Clojure

I wonder what is the benefit of that syntax decision?

syntax clojure

edited Jul 9 '12 at 9:00
Jeff Foster
17.8k ●3 ●23 ●46

asked Jul 9 '12 at 8:47
Christoph
1,989 ●1 ●6 ●20

People, if you vote for closing. It would be nice to leave a comment to give me the chance to improve the question. Yes, the question is very short and concise but does that make a bad question? – Christoph Jul 9 '12 at 11:27

tagged

syntax	× 6223
clojure	× 4475

asked	**7 months ago**
viewed	**237 times**
active	**7 months ago**

Community Bulletin

blog **Podcast #42 – It's The Exception That Proves The Rule**

Homoiconic

El código se expresa en las estructuras de datos propias del lenguaje.

Clojurescript?

- Es una implementacion de Clojure, no un dialecto.
- Algunas decisiones de diseño difieren pero “se siente” como el lenguaje.
- JS single-threaded, primitivas de concurrencia no tienen sentido
- Macros
- Promises native support for callback fell

Clojurescript?

- 100% Clojure 0% Javascript
- Persistent Data Structures
- No se compila a si mismo, precisa Clojure
  - reader no esta escrito en Clojure
- Immutability va mas alla de la concurrencia

Clojurescript Compilation Process

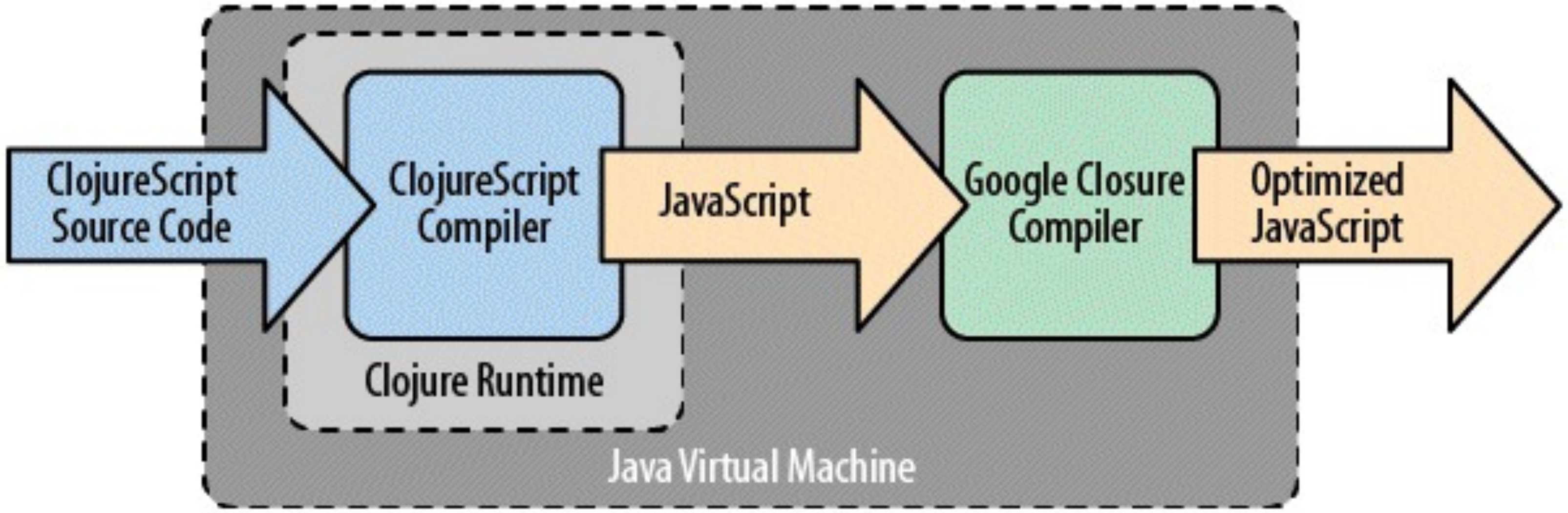


Figure 3-1. ClojureScript Compilation Process

http://www.amazon.com/ClojureScript-Up-Running-Stuart-Sierra/dp/1449327435

Closure, no es un minificador

- Compiler
  - js => js
  - optimizing compiler
  - whole program optimization
  - dead code elmination

- Library
  - Todas las apps de google, gmail, maps, google+
  - Al combinar con el compilador se obtiene el beneficio al reducir size.

Feature número uno de javascript?

Google

Leverage para quienes compilan a JS

La idea no es nueva...

- Google Web Toolkit
- CoffeeScript
- Dart

Rationale

- JS no es robusto, cae en el usuario la responsabilidad
- Sube la demanda por “true applications”
- Programa en lo que quieras, run everywhere

You can write large programs in JavaScript. You just can’t maintain them.

Anders Hejlsberg

http://www.youtube.com/watch?v=lT07c3OREL8

Code?

```
var x = 1 + 1;
```

```
(def x (+ 1 1))
```

```
var x = (+).apply(null, [1, 2]);
```

```
(def x (apply + [1 1]))
```

```
var x = foo[0];
```

```
(def x (aget foo 0))
```

```clojure
(def x [+ aget])
```

```
if (“”) {
    // ...
} else {
    // execute
}
```

WAT?

```
(if "" true
false)
```

```
if (cljs.core.truth_("")) {
  // execute
} else {
  // ...
}
```

```
(deftype Foo [] IFn
(-invoke [this s]
(.log js/console s)))

((Foo.) "I'm a function!")
```

```
obj.call(null, 1, 2);
```

ClojureScript Macromagic

```
(defn example [datum]
  [:li [:a {:href (str "#show/" (:key datum))}
        [:div.class1.class2 {:id (str "item" (:key datum))}
           [:span.anchor (:name datum)]]]])
```

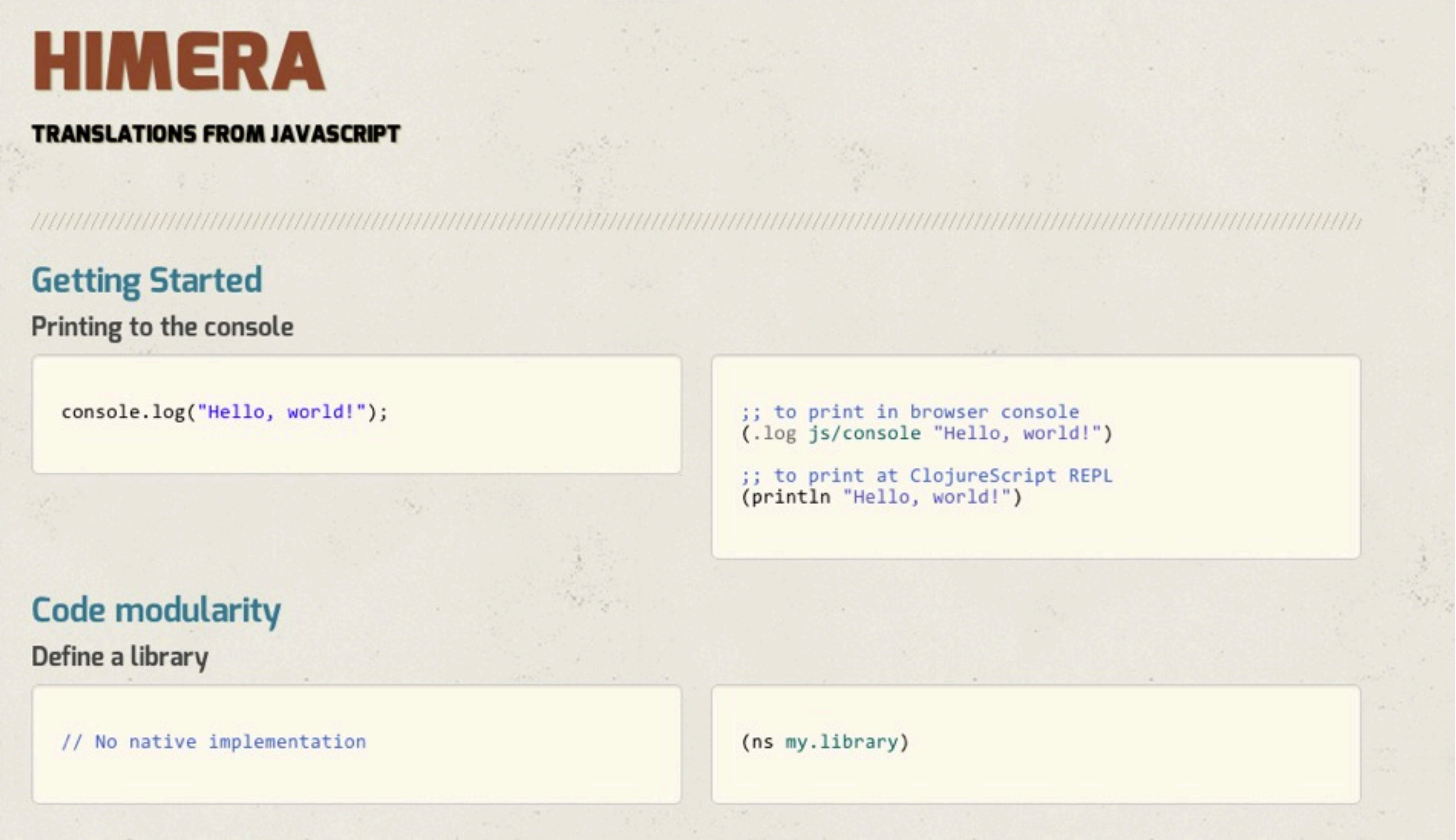
jQuery: 1.57 secs

dommy-macro: 0.44 secs

```
$('li').append(
      $('<a>').attr('href', '#show/' + datum.key)
              .addClass('anchor')
              .append( $('<div>').addClass('class1').addClass('class2')
                                 .attr('id', 'item' + datum.key)
                              .append( $('<span>').text(datum.name) )))
```

http://blog.getprismatic.com/blog/2013/1/22/the-magic-of-macros-lighting-fast-templating-in-clojurescript

http://himera.herokuapp.com/synonym.html



HIMERA

TRANSLATIONS FROM JAVASCRIPT

Getting Started

Printing to the console

```
console.log("Hello, world!");
```

```
;; to print in browser console
(.log js/console "Hello, world!")

;; to print at ClojureScript REPL
(println "Hello, world!")
```

Code modularity

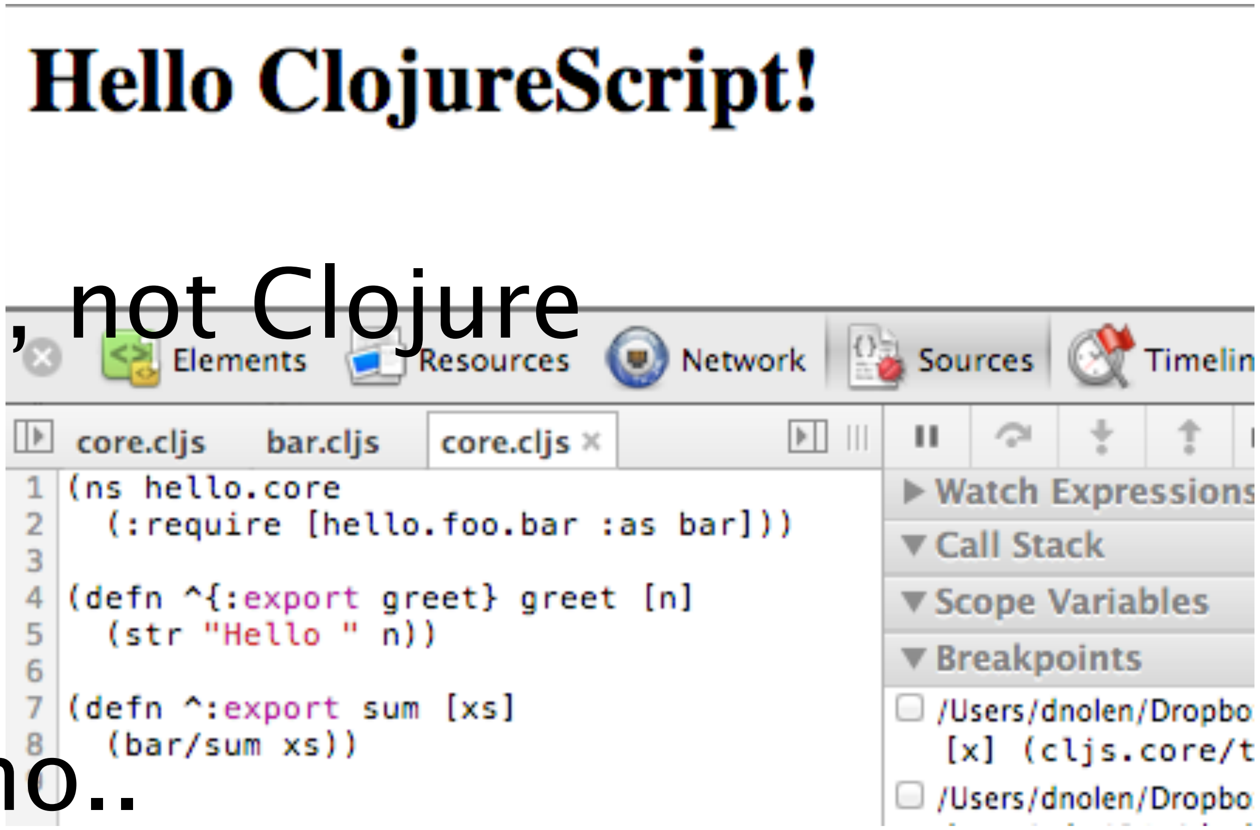
Define a library

```
// No native implementation
```

```
(ns my.library)
```

Pain points

- JS arithmetic underneath, not Clojure numerics
- Debugging
  - Source-maps en camino..
- Not bootstrapped - requires Clojure



Hello ClojureScript!

```
(ns hello.core
  (:require [hello.foo.bar :as bar]))

(defn ^{:export greet} greet [n]
  (str "Hello " n))

(defn ^:export sum [xs]
  (bar/sum xs))
```

DEMO

Preguntas?