

The background is a faded illustration from the anime Attack on Titan. On the right, a large, close-up face of a Titan with a wide, toothy grin and a dark, circular mouth. On the left, a Scout in a brown uniform is shown from the back, holding a vertical maneuvering equipment (VME) and a sword, appearing to be in a combat stance.

Why you should watch Attack on Titan

Why...?

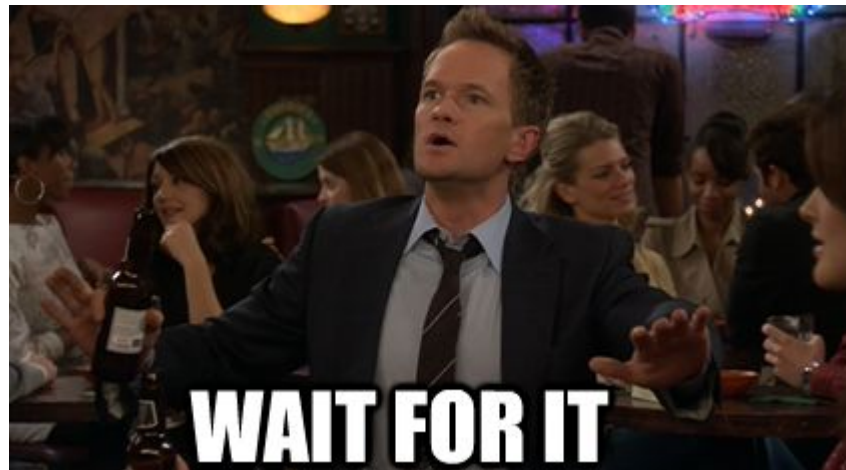
Why Clojure!?

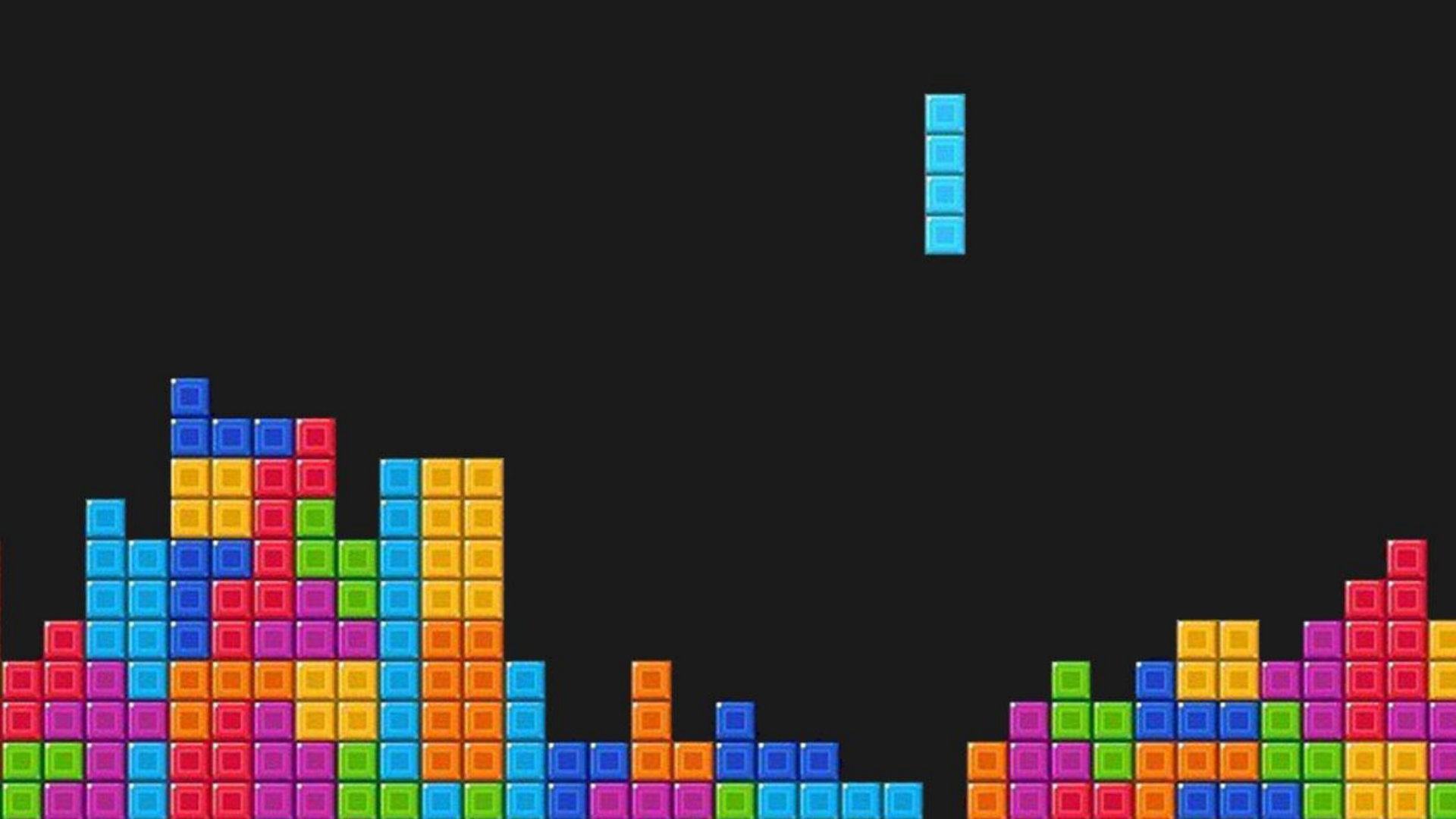
**"A language that doesn't affect the
way you think about programming,
is not worth knowing."**

Alan Perlis

TL;DR ~ Why Clojure?

- Lisp
- Linguagem de propósito geral
- Clojure criado para lidar com concorrência
- Empresas pops usam, uhuu (Amazon, Thoughtworks, Facebook)





Linguagem dinâmica

E meus
tipos?



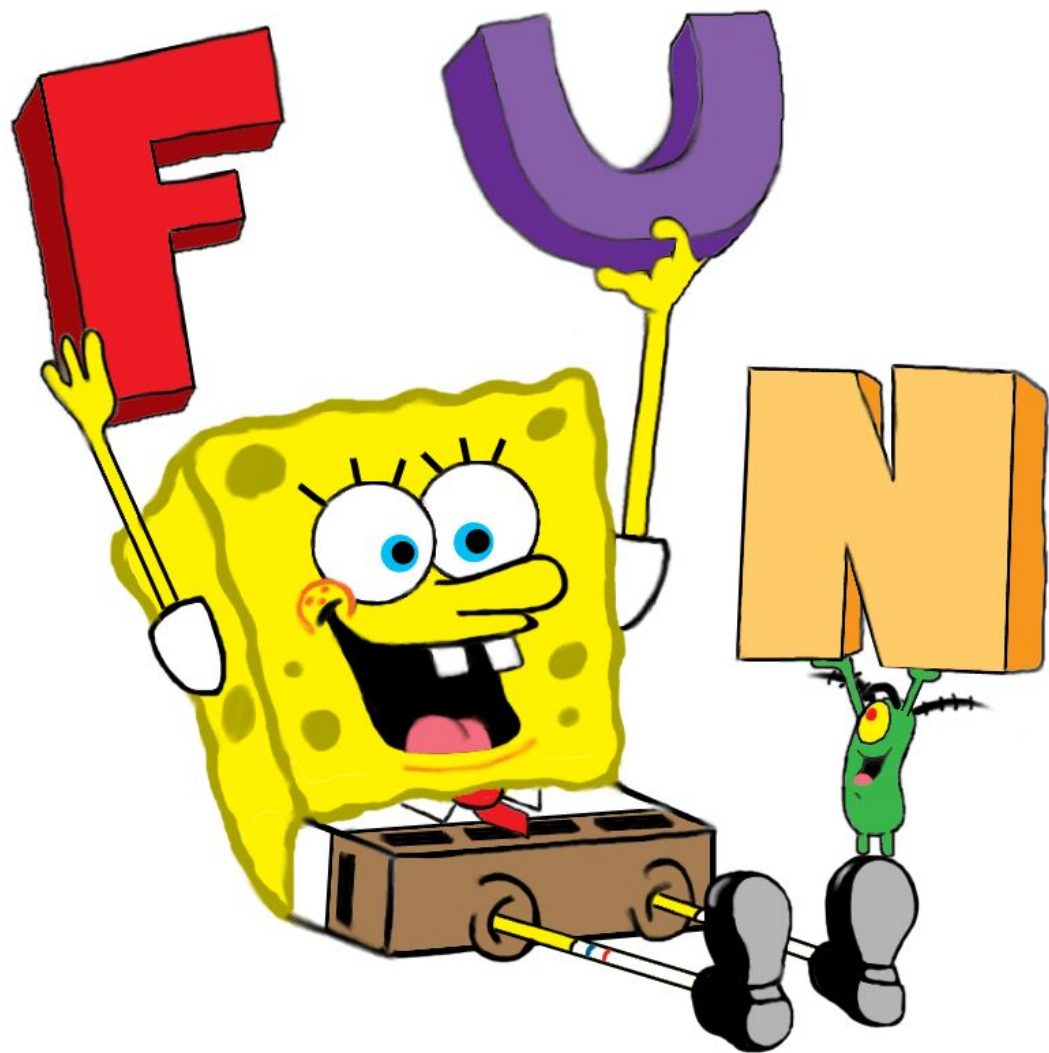
```
(defn yell?  
  [sentence]  
  (and (> (count (re-seq #"[A-Z]" sentence)) 0)  
        (= (count (re-seq #"[a-z]" sentence)) 0)))
```

Imutável



Paradigma funcional

- Functions as "first class citizens"
- Code as data
- Programação concorrente
- map / filter / reduce
- Funções puras



Macros legais

```
(escreva "Olá Mundo")
```

```
(definição contagem-regressiva
```

```
  (função [valor]
```

```
    (se (= 0 valor)
```

```
      (escreva 0)
```

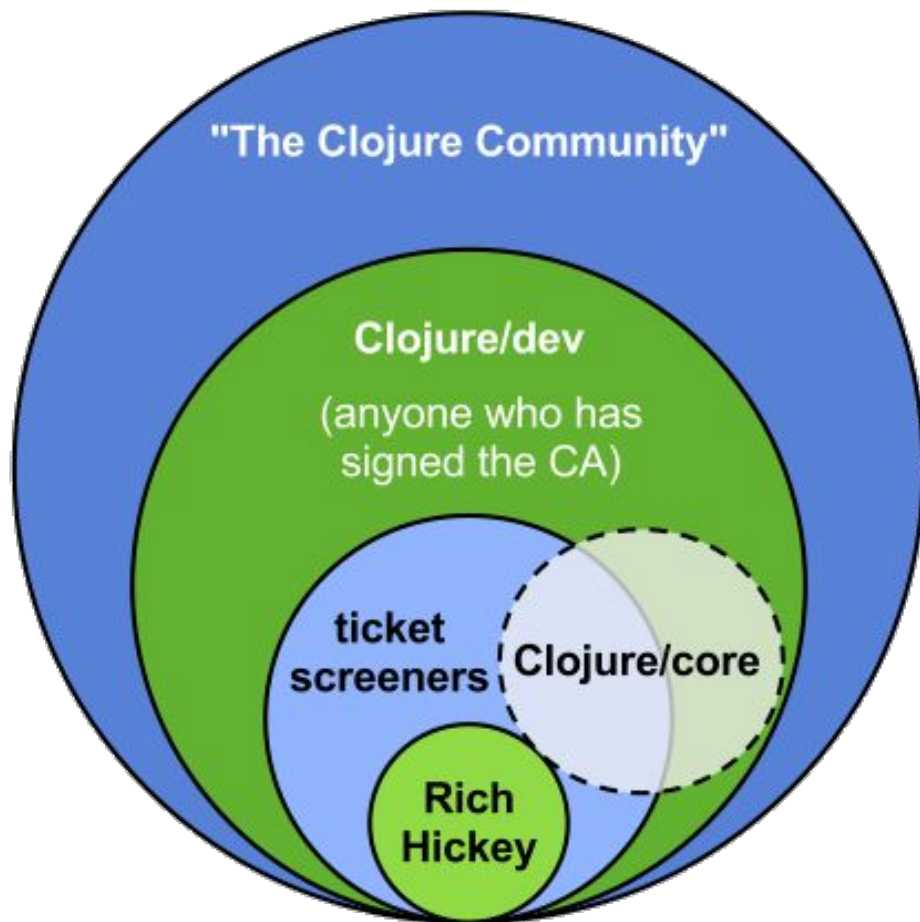
```
      (faça (escreva valor)
```

```
        (contagem-regressiva (- valor 1))))))
```

```
(contagem-regressiva 10)
```

Comunidade

- Empregabilidade - Sites da comunidade:
 - <https://jobs.braveclojure.com/>
 - <https://clojuredocs.org/jobs>
- Comunidade receptiva com iniciantes
 - Basta se inscrever no slack, watch/fork do clojure no github.
 - <https://github.com/clojure/clojure>
 - IRC do clojure: <https://freenode.net/>
 - ClojureTV: <https://www.youtube.com/user/ClojureTV/videos>



Material de Estudo

- Clojure for the Brave and True: <http://www.braveclojure.com/>
(beginner)
- The Joy of Clojure: <https://www.manning.com/books/the-joy-of-clojure>
(beginner/intermediate)
- Clojure Applied: <https://pragprog.com/book/vmclojeco/clojure-applied>
(intermediate)
- Clojure CheatSheet: <https://clojure.org/api/cheatsheet>
- Clojure Toolbox: <https://www.clojure-toolbox.com/>

Material para praticar

- Exercism: <http://exercism.io/languages/clojure/about>
- 4Clojure: <http://www.4clojure.com/>
- Clojure Koans: <http://clojurekoans.com/>

Interop com o tio Javão

- Compilador Clojure escrito em Java!
- Tudo vira bytecode no fim das contas

```
(.toUpperCase "fred") ; => "FRED"  
(.getName String) ; => "java.lang.String"  
(.-x (java.awt.Point. 1 2)) ; => 1  
(System/getProperty "java.vm.version") ; => "1.6.0_07-b06-57"  
(Math/PI) ; => 3.141592653589793  
(new java.util.Date) ; => "2012-10-09T21:24:27.229-00:00"
```

Interop com o tio Javão

```
(ns com.domain.tiny
  (:gen-class :name com.domain.tiny
               :methods [#^{:static true} [binomial [int int] double]]))

(defn binomial [n k]
  (let [a (inc n)]
    (loop [b 1
           c 1]
      (if (> b k)
          c
          (recur (inc b) (* (/ (- a b) b) c))))))

(defn -binomial [n k]
  (binomial n k))
```


Interop C#

- Clojure possui versão compatível com a CLR
- Infelizmente ainda estamos num estágio na qual Clojure-CLR < Clojure-JVM
- Arcadia Unity (Clojure in Unity)

Download
INTEROP.DAO.DLL
Error Repair Tool



Download **WinThruster** Now

Interop JavaScript => ClojureScript



jQuery

Interop JavaScript => ClojureScript

- Universo front-end em Lisp (<adicionar todas as buzzwords possíveis aqui>)
- Clojure <3 Node.js
- ClojureScript + ReactNative



Novas formas de pensar

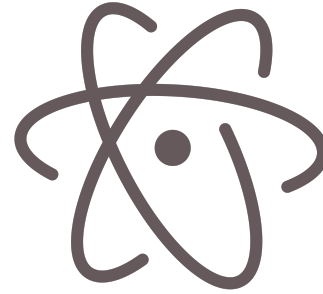
- Gerenciamento de estado
- `clojure.spec`
- TechRadar
- Ciclo de desenvolvimento (REPL Driven Development)
- Existe um mundo de unicórnios fora do encapsulamento de OO

Performance

- Mas qual o host alvo? JVM, CLR, ou Javascript?
- Runtime vs startup time (mas temos o REPL)
- Acrescenta overhead em runtime e é mais lento que Java
- Otimizações manuais podem tornar o código "tão rápido quanto" Java
- Maior possibilidade de resolver o problema explorando concorrência e paralelismo (STM)
- Código imutável => novas otimizações possíveis (ClojureScript + React)
- Ferramentas super battle-tested em Java (profilers, etc)
- Golang Goroutines & Clojure core.async

REPL

Read
Eval
Print
Loop





```
(loop (print (eval (read))))
```

https://en.wikipedia.org/wiki/Read-eval-print_loop

Referências

- <https://github.com/clojure/clojurescript>
- <http://www.spacjer.com/blog/2014/09/12/clojurescript-javascript-interop/>
- ClojureScript CheatSheet: <http://cljs.info/cheatsheet/>
- Video!!! <https://lambdaisland.com/episodes/clojurescript-interop>

Interop C#

- <https://github.com/clojure/clojure-clr>
- <https://arcadia-unity.github.io/>

Muitas outras referências curadas

- <https://github.com/razum2um/awesome-clojure>

Referências - Comunidade

- Github de tudo que é clojure: <https://github.com/clojure>
- Coisas interessantes do StackOverflow:
<http://stackoverflow.com/documentation/clojure/topics>
- Canais no slack: <http://clojurians.net/>
- Clojure Style-Guide: <https://github.com/bbatsov/clojure-style-guide>