What is ClojureScript?

Michael Langford

Looks like this

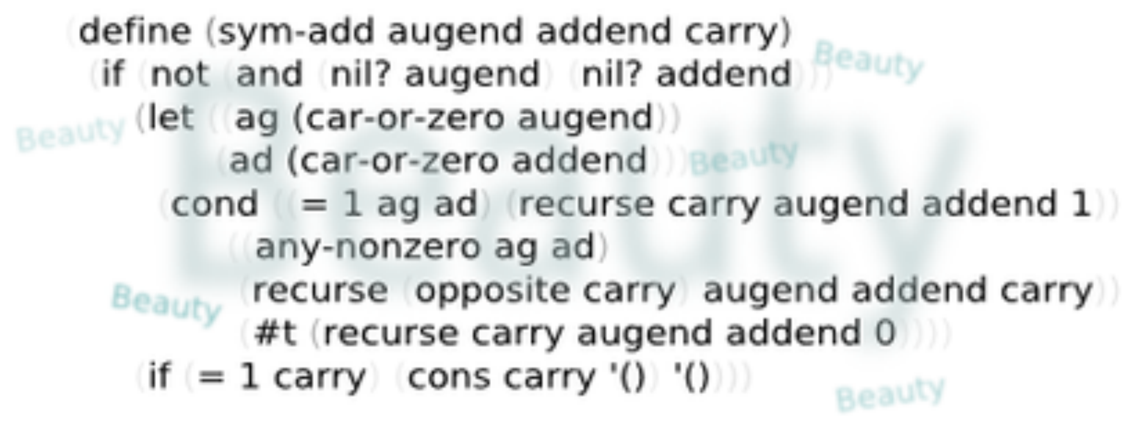
```
(enable-console-print!)

;this is a comment
(defn logged-alert
    "alerts the user of a problem"
    [message]
    (println (str "alert: " message))
    (js/alert message))

(sample-alert "We have liftoff")
```

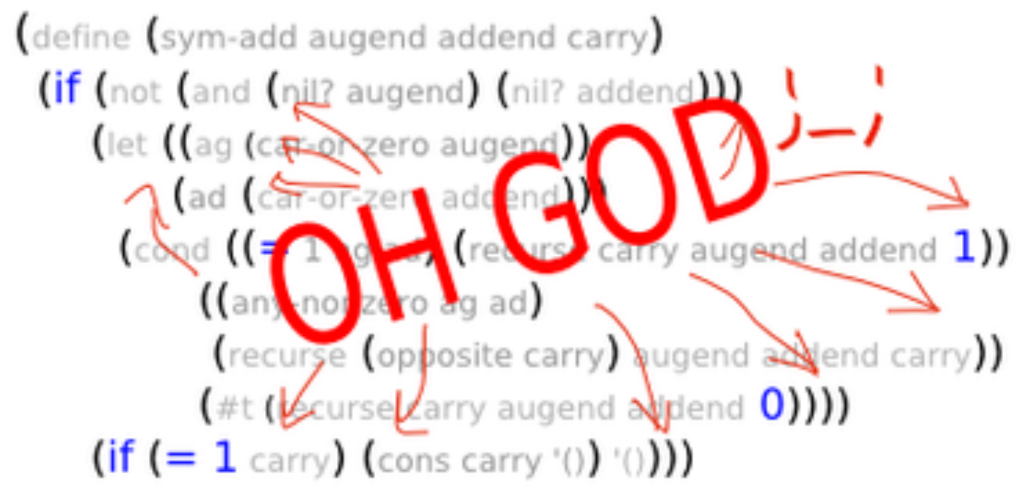
I promise this happens

What I see

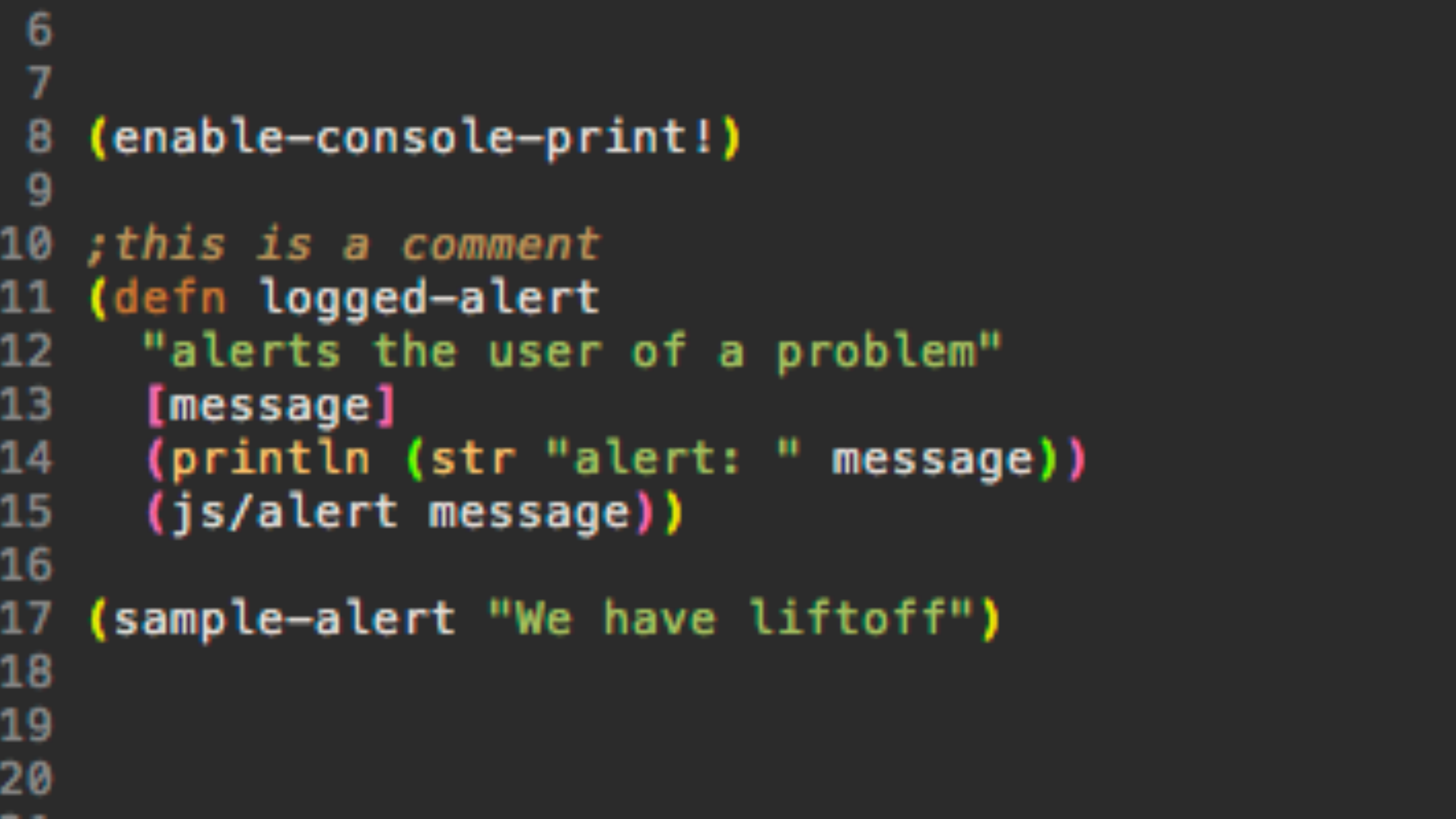

Beauty
Beauty
Beauty
Beauty
Beauty
Beauty

```
(define (sym-add augend addend carry)
 (if (not (and (nil? augend) (nil? addend)))
   (let ((ag (car-or-zero augend))
         (ad (car-or-zero addend)))
     (cond ((= 1 ag ad) (recurse carry augend addend 1))
           ((any-nonzero ag ad)
            (recurse (opposite carry) augend addend carry))
           (#t (recurse carry augend addend 0))))
   (if (= 1 carry) (cons carry '()) '())))
```

What the non-Lisper sees



```
(define (sym-add augend addend carry)
 (if (not (and (nil? augend) (nil? addend)))
   (let ((ag (car-or-zero augend))
         (ad (car-or-zero addend)))
     (cond ((= 1 ag ad) (recurse carry augend addend 1))
           ((any-nonzero ag ad)
            (recurse (opposite carry) augend addend carry))
           (#t (recurse carry augend addend 0))))
   (if (= 1 carry) (cons carry '()) '())))
```



```clojure


(enable-console-print!)

;this is a comment
(defn logged-alert
  "alerts the user of a problem"
  [message]
  (println (str "alert: " message))
  (js/alert message))

(sample-alert "We have liftoff")



```

Transpiled version of Clojure that is hosted in the JS language

- Uses prefix notation for everything
- Transpiled to JavaScript, then tree shaken by Google V8 to cut down on unneeded code
- Prizes immutability and consistency far more than JavaScript
- Is fun and fast to develop in, like all Lisp family languages
- Is very expressive (something programmers love) and concise

Advantages over plain JS

- Very low wat factor (real numbers, consistent true/false, easier to find errors)
- You can add new keywords to the language (macros)
- Allows for some immutability guarantees which allow some performance gains
- Can use JS libs, while has more stable approaches over time than JS does
- Great tooling support (Emacs/CIDER and CursiveClojure, as

Disadvantages over plain JS

- Requires a transpilation step
- For small apps, a small but certain group of libraries won't get tree shaken, so a couple hundred k will stick around
- The java based CLJS compiler is a bit slow on startup time (the self hosted one is lightning fast)
- Single language folks complain *loudly* until they don't
- Some learning of Leiningen/boot (the build tools) required

Thanks

@mj_langford