

# Carp

## A Language for the 21st Century

---

Veit Heller

Port Zero

May 16, 2019

- ▶ PL nerd
- ▶ CTO @ Port Zero
- ▶ Carp standard library maintainer
- ▶ Secretly a turtle

I gave a talk of the same title at clojuTRE 2018, almost a year ago.

This is not the same talk.  
Sorry for the naming.

- ▶ a Lisp-1
- ▶ type-inferred
- ▶ borrow-checked
- ▶ compiles to C
- ▶ for realtime applications

- ▶ a Lisp-1
- ▶ type-inferred  $\Rightarrow$  statically typed, at no extra charge
- ▶ borrow-checked  $\Rightarrow$  no GC, at no extra charge
- ▶ compiles to C
- ▶ for realtime applications

whence -v carp

- ▶ Haskell implements a Hindley-Milner type system (with sumtypes) and inference
  - ⇒ You don't have to spell types out!
- ▶ Rust implements borrow checking
  - ⇒ You don't have to manually manage memory, even without a GC!

whence -v carp

Let's put those things together (after simplifying) and rejoice!

⇒ Also add some Lisp macro goodness and a near-seamless C FFI for good measure!



```
; (type f)
; f : (Fn [(Ref (Array a)), Int, Int] a)
(defn f [x y z]
  @(Array.nth x (* y z)))
```

Listing 1: A silly Carp function

```
(deftype (AssocArray a b) [lst (Array (Pair a b))])
```

Listing 2: An associative array type, simplified.

Carp is early stage software.

- ⇒ Small community, few packages
- ⇒ We're less than a handful of maintainers
- ⇒ Insufficient documentation
- ⇒ May change under your feet
- ⇒ May blow up in your face!

The first stable release (0.3) was released on the 1st of July 2019.

# Thank you!

Questions?

Slides at [https://github.com/hellerve/carp\\_talks](https://github.com/hellerve/carp_talks)