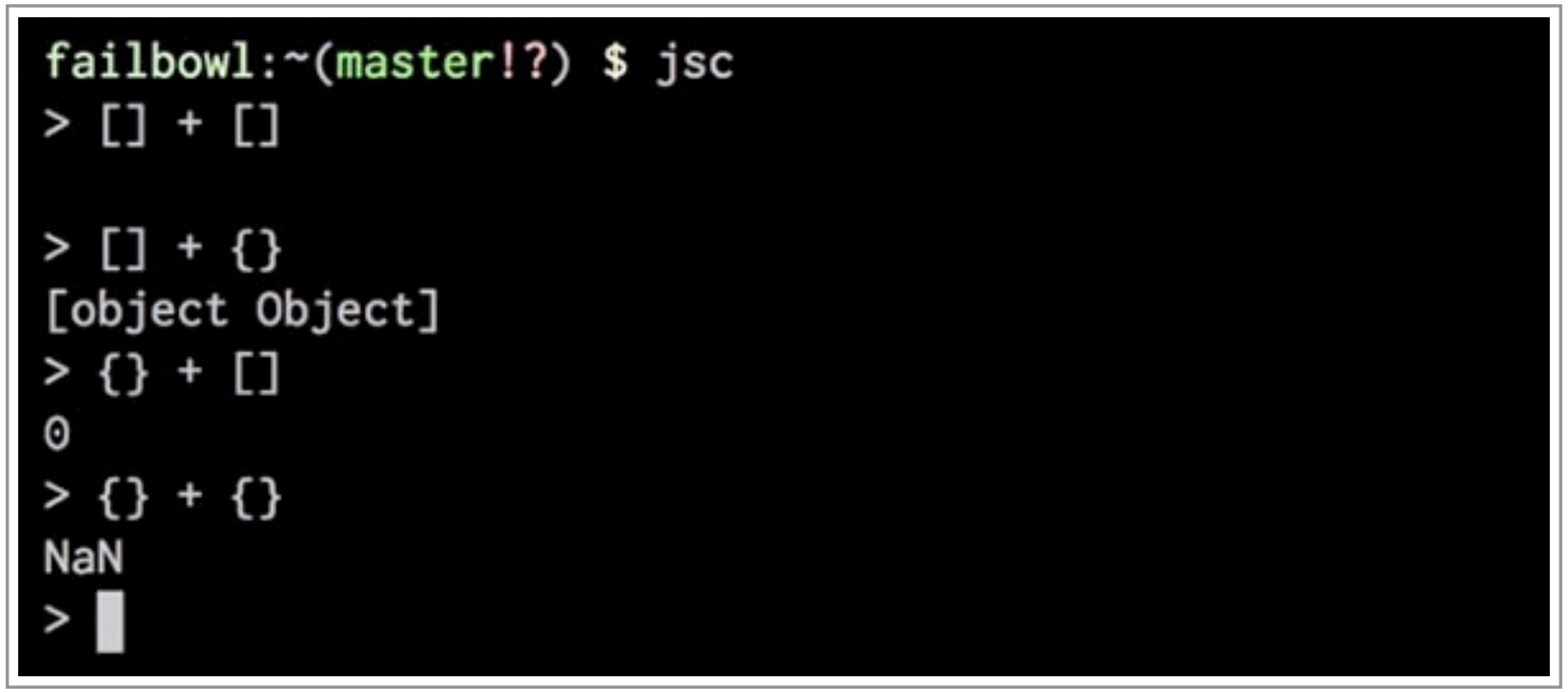
ClojureScript

The Essence of Alchemy

JS

MacBook Air

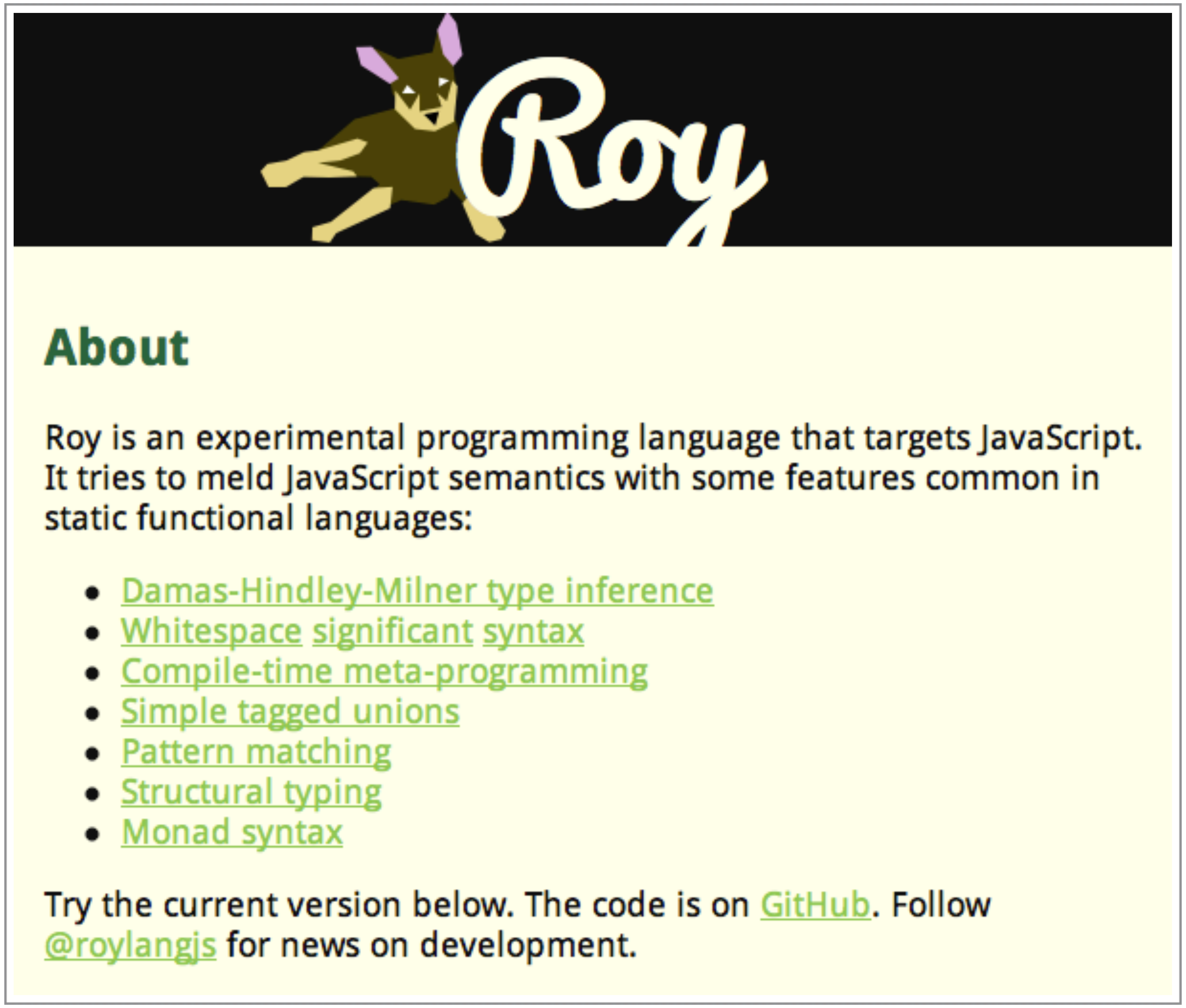


```
failbowl:~(master!?) $ jsc
> [] + []

> [] + {}
[object Object]
> {} + []
0
> {} + {}
NaN
> 
```

TypeScript

PREVIEW



Roy

About

Roy is an experimental programming language that targets JavaScript. It tries to meld JavaScript semantics with some features common in static functional languages:

- Damas-Hindley-Milner type inference
- Whitespace significant syntax
- Compile-time meta-programming
- Simple tagged unions
- Pattern matching
- Structural typing
- Monad syntax

Try the current version below. The code is on GitHub. Follow @roylangjs for news on development.

Challenges

Costs vs. Benefits

Costs vs. Benefits

- Provide significantly better semantics

Costs vs. Benefits

- Provide significantly better semantics
- Interoperate well with the host

Costs vs. Benefits

- Provide significantly better semantics
- Interoperate well with the host
- Debugging story

Costs vs. Benefits

- Provide significantly better semantics
- Interoperate well with the host
- Debugging story
- Be able to reach the performance of the host

Host limitations

Host limitations

- Single threaded

Host limitations

- Single threaded
- Numerics

Host limitations

- Single threaded
- Numerics
- Source only (no bytecode)

Host limitations

- Single threaded
- Numerics
- Source only (no bytecode)
- Some language design choices may be challenging to implement efficiently (messaging? laziness?)

Host benefits

Host benefits

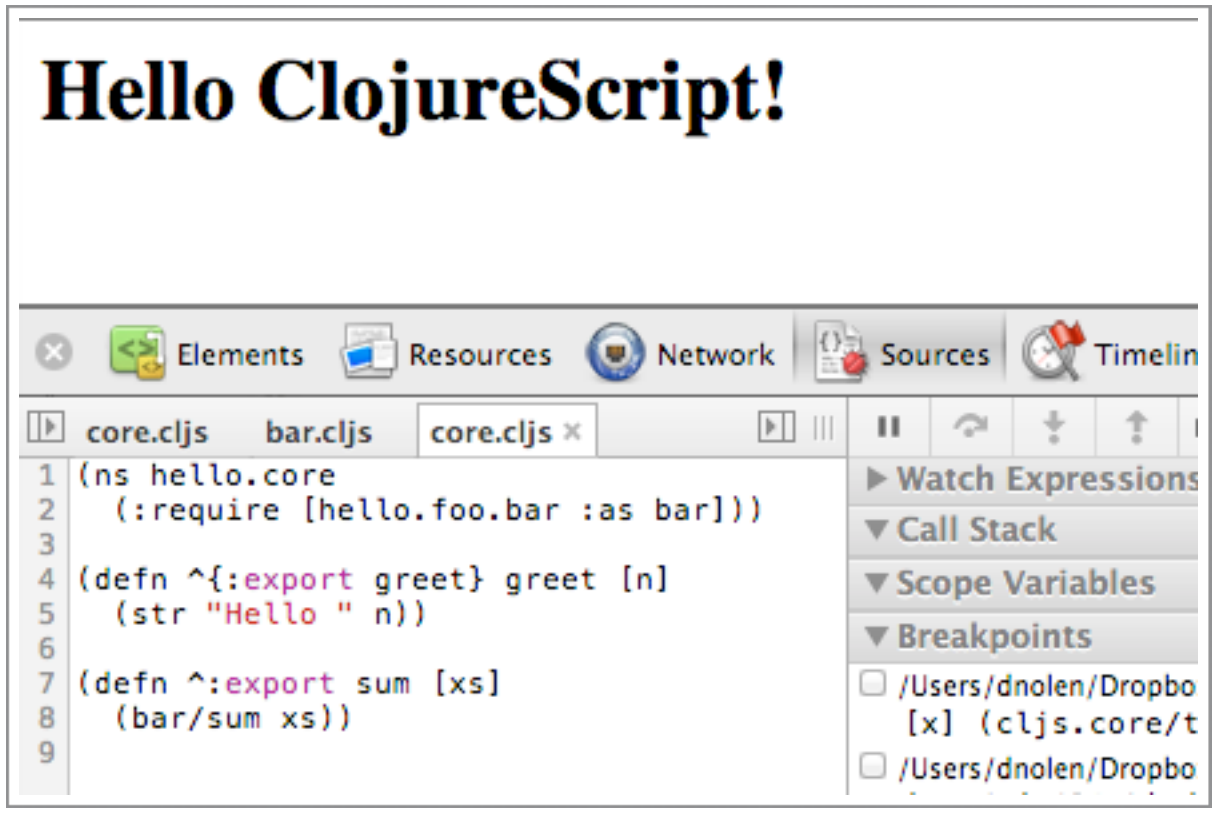
- Rich and rapidly evolving multimedia primitives with incredible reach (HTML Canvas, WebGL, WebAudio, WebRTC, etc.)

Host benefits

- Rich and rapidly evolving multimedia primitives with incredible reach (HTML Canvas, WebGL, WebAudio, WebRTC, etc.)
- Longevity - we can run JS from a decade ago.

Host benefits

- Rich and rapidly evolving multimedia primitives with incredible reach (HTML Canvas, WebGL, WebAudio, WebRTC, etc.)
- Longevity - we can run JS from a decade ago.
- Vendor competition towards standardization

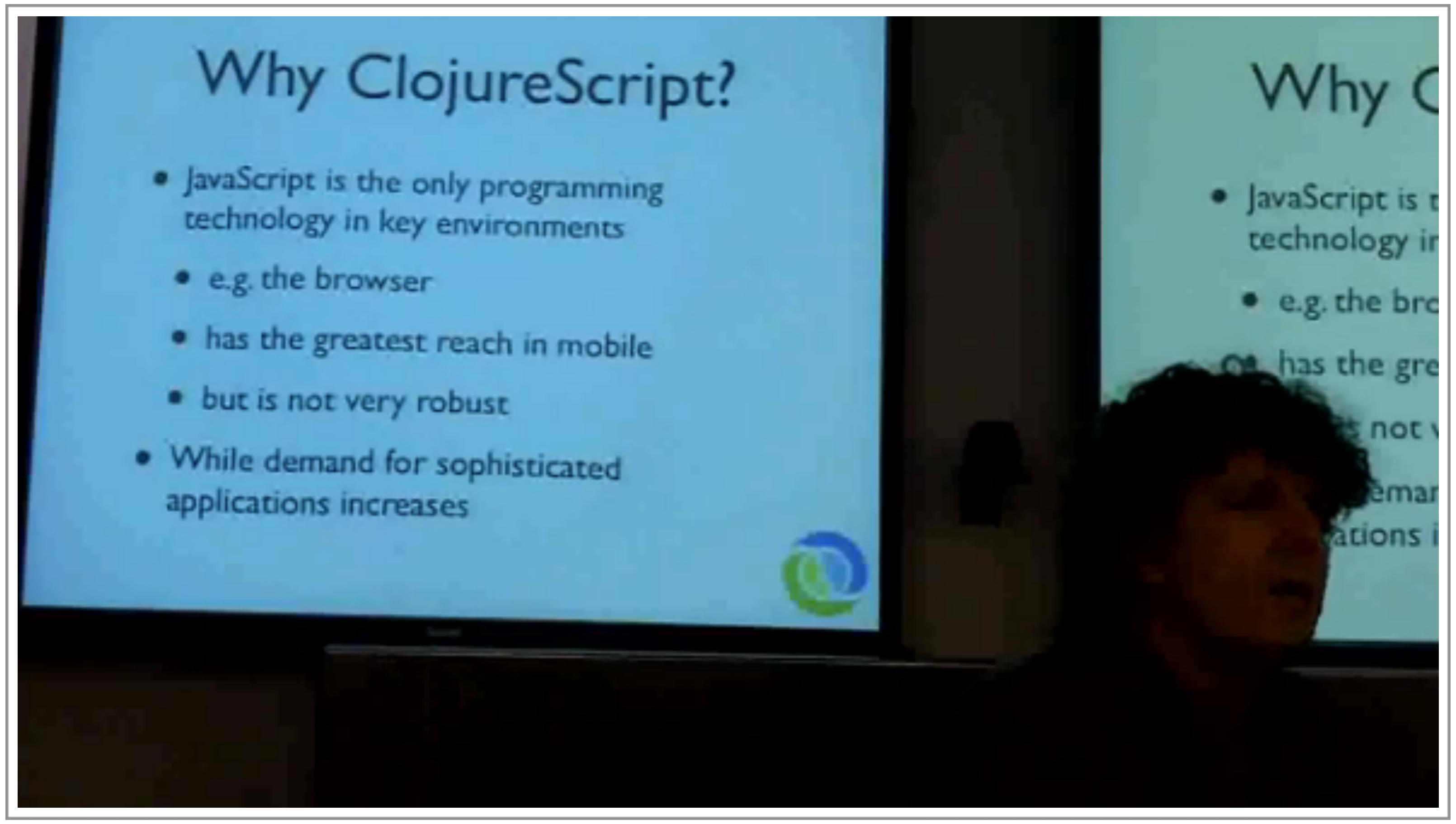

Hello ClojureScript!

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

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

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

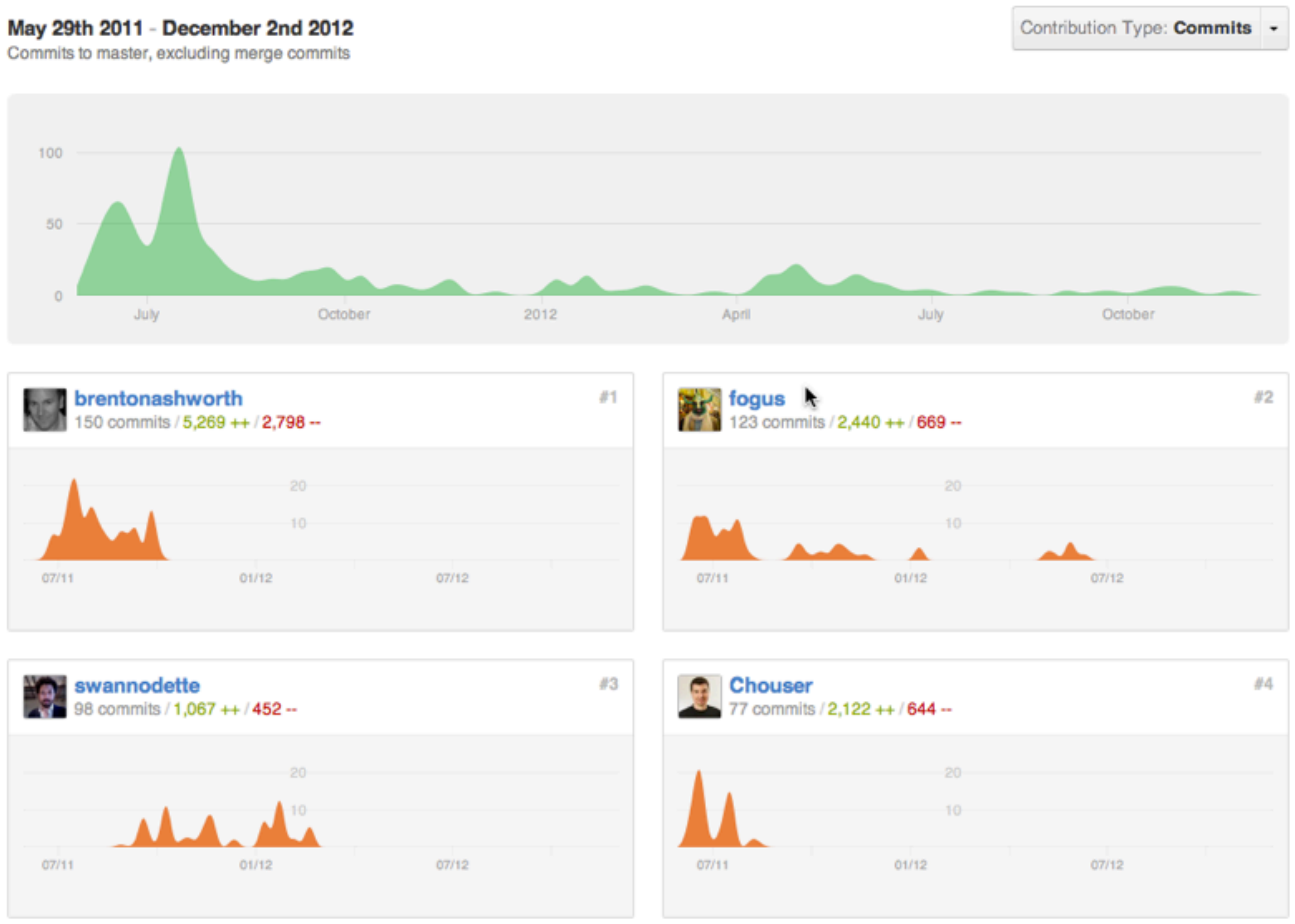
/Users/dnolen/Dropbo
[x] (cljs.core/t
/Users/dnolen/Dropbo



Why ClojureScript?

- JavaScript is the only programming technology in key environments
  - e.g. the browser
  - has the greatest reach in mobile
  - but is not very robust
- While demand for sophisticated applications increases

July 21st, 2011



May 29th 2011 - December 2nd 2012
Commits to master, excluding merge commits

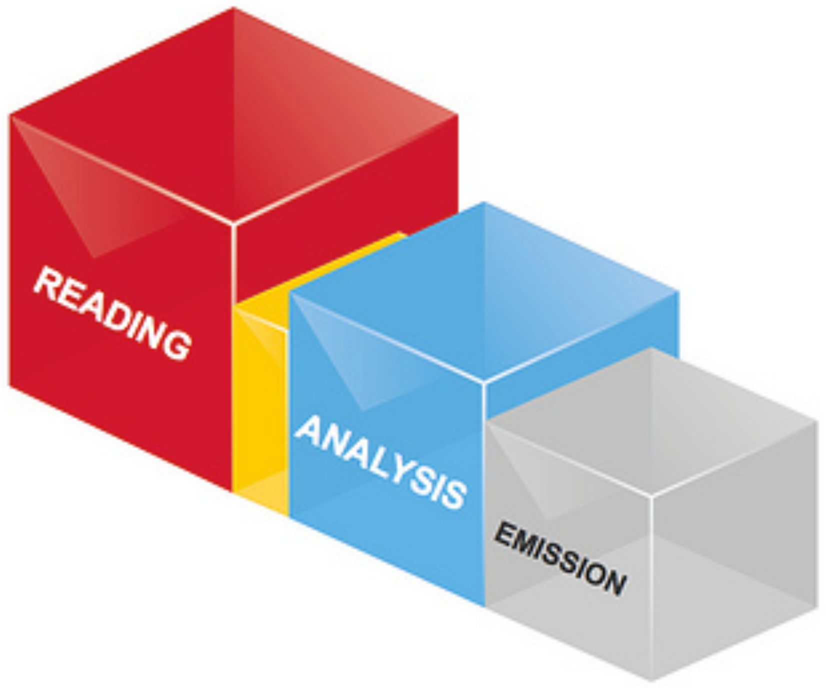
Contribution Type: **Commits**

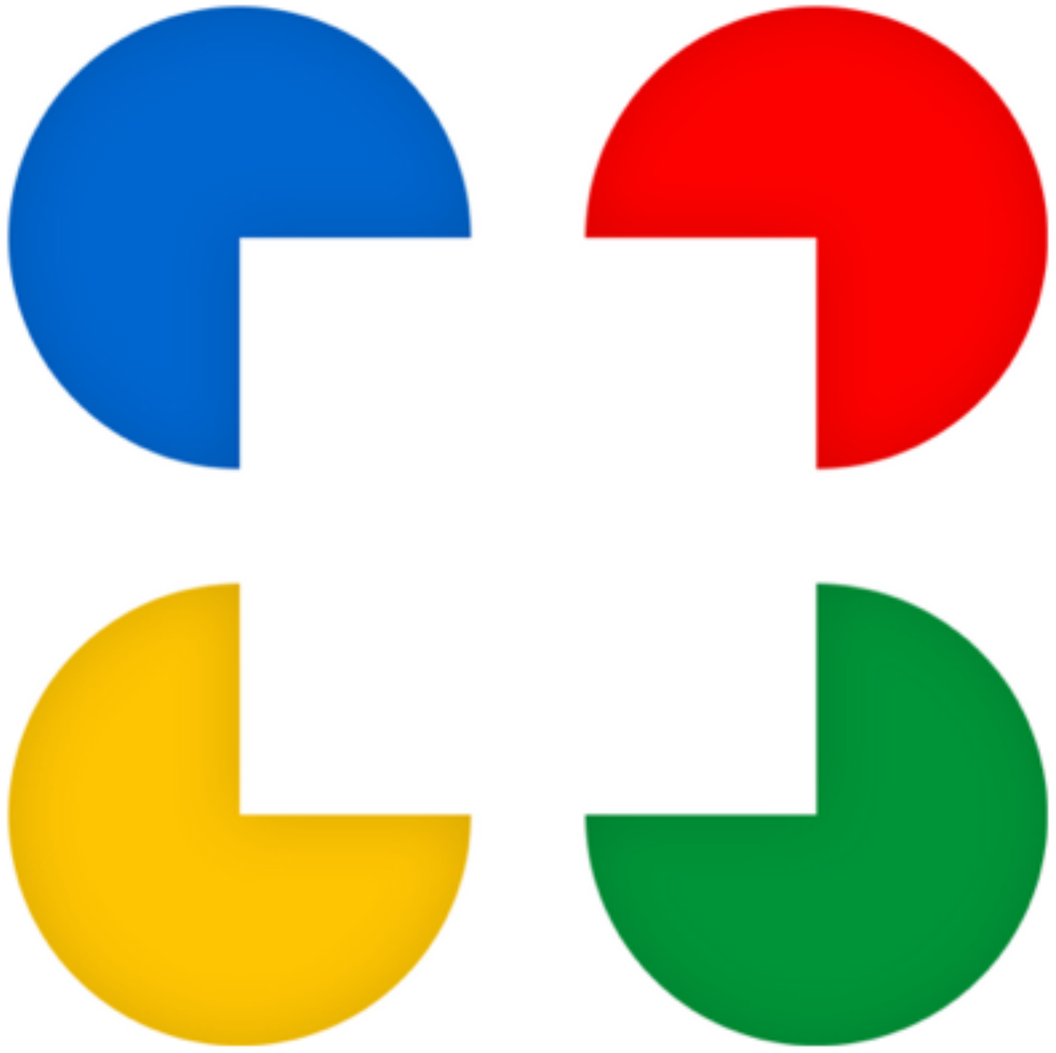
brentonashworth #1
150 commits / 5,269 ++ / 2,798 --

fogus #2
123 commits / 2,440 ++ / 669 --

swannodette #3
98 commits / 1,067 ++ / 452 --

Chouser #4
77 commits / 2,122 ++ / 644 --


READING
ANALYSIS
EMISSION



Google Closure

Google Closure

- Dead code elimination

Google Closure

- Dead code elimination
- Inlining

Google Closure

- Dead code elimination
- Inlining
- Constant folding

Google Closure

- Dead code elimination
- Inlining
- Constant folding
- Minification

Google Closure

- Dead code elimination
- Inlining
- Constant folding
- Minification
- Large library of battle tested JS

Not handled

Not handled

- Invocation

Not handled

- Invocation
  - Rest arguments

Not handled

- Invocation
    - Rest arguments
    - Multi-arity fns

Not handled

- Invocation
  - Rest arguments
  - Multi-arity fns
  - Keywords

Not handled

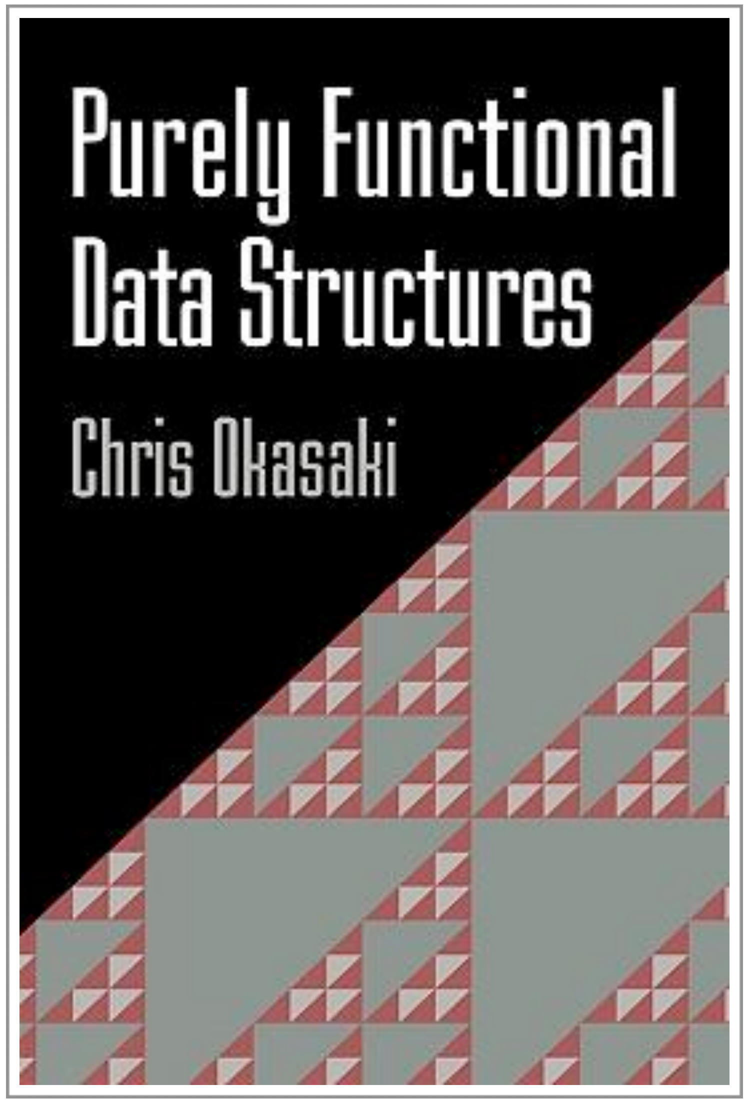
- Invocation
  - Rest arguments
  - Multi-arity fns
  - Keywords
  - Inline protocol implementations

Not handled

- Invocation
  - Rest arguments
  - Multi-arity fns
  - Keywords
  - Inline protocol implementations
- Boolean inference

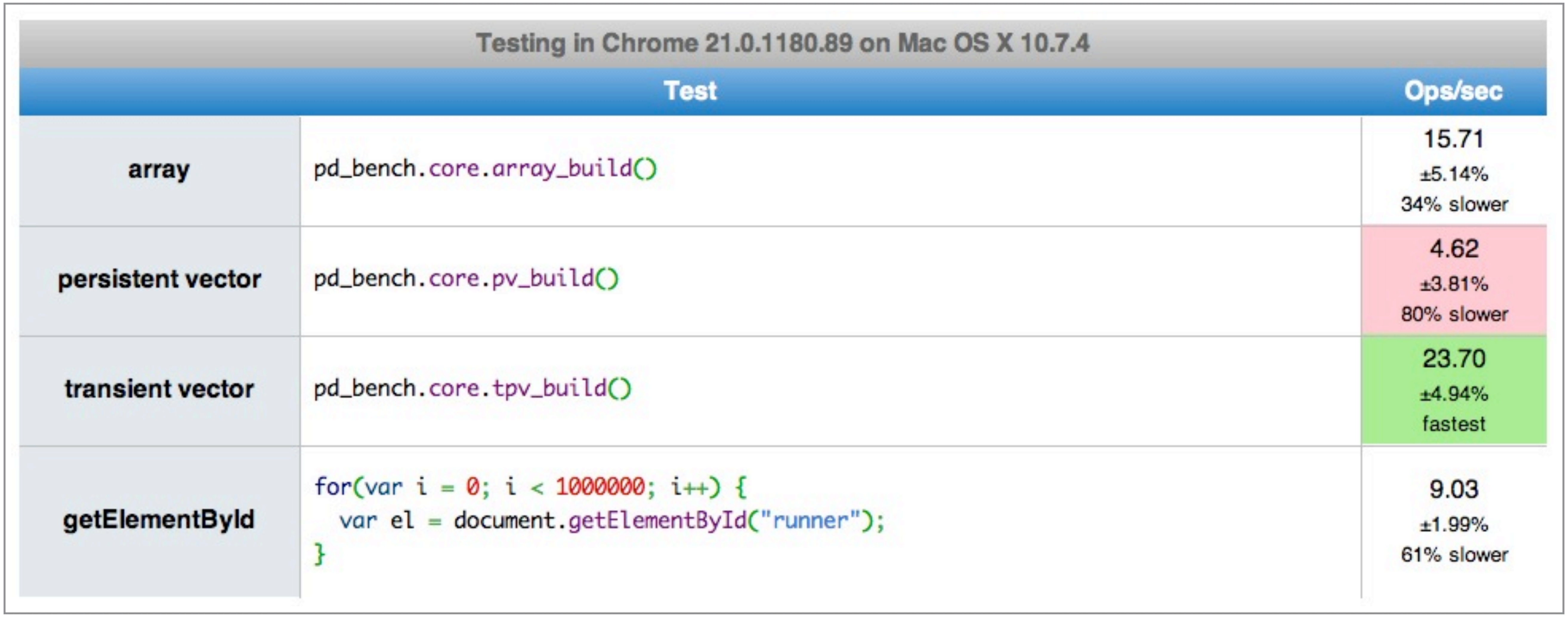
Not handled

- Invocation
  - Rest arguments
  - Multi-arity fns
  - Keywords
  - Inline protocol implementations
- Boolean inference
- Numerics


Purely Functional Data Structures

Chris Okasaki

Data all the things



Testing in Chrome 21.0.1180.89 on Mac OS X 10.7.4

	Test	Ops/sec
array	`pd_bench.core.array_build()`	15.71 ±5.14% 34% slower
persistent vector	`pd_bench.core.pv_build()`	4.62 ±3.81% 80% slower
transient vector	`pd_bench.core.tpv_build()`	23.70 ±4.94% fastest

getElementById	`for(var i = 0; i < 1000000; i++) {`
	`  var el = document.getElementById("runner");`
	`}`
		9.03 ±1.99% 61% slower

demo

Goodies

Pattern Matching

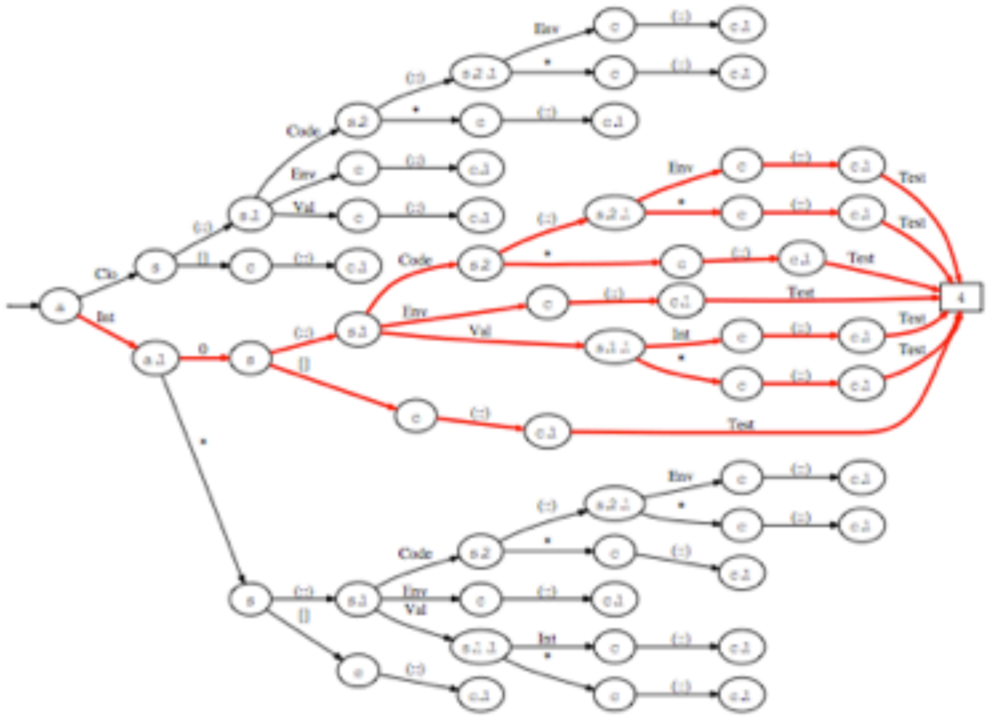


Figure 6. Naive decision tree for example 3

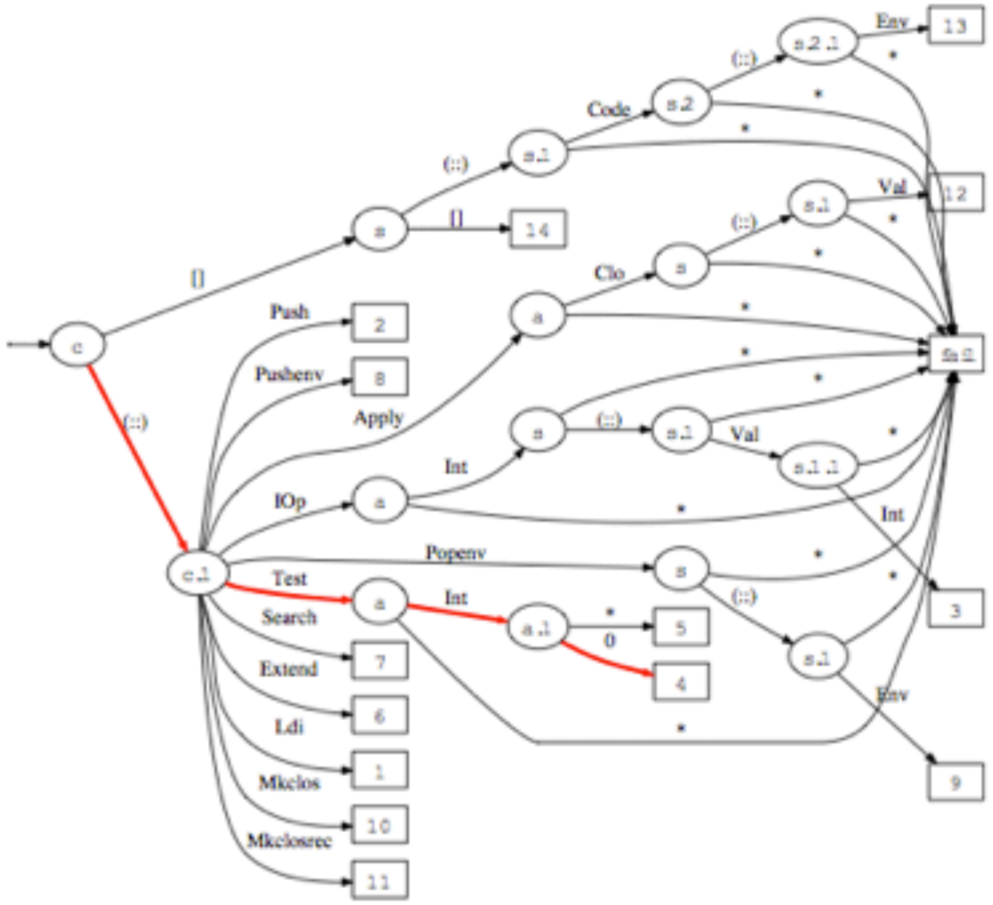


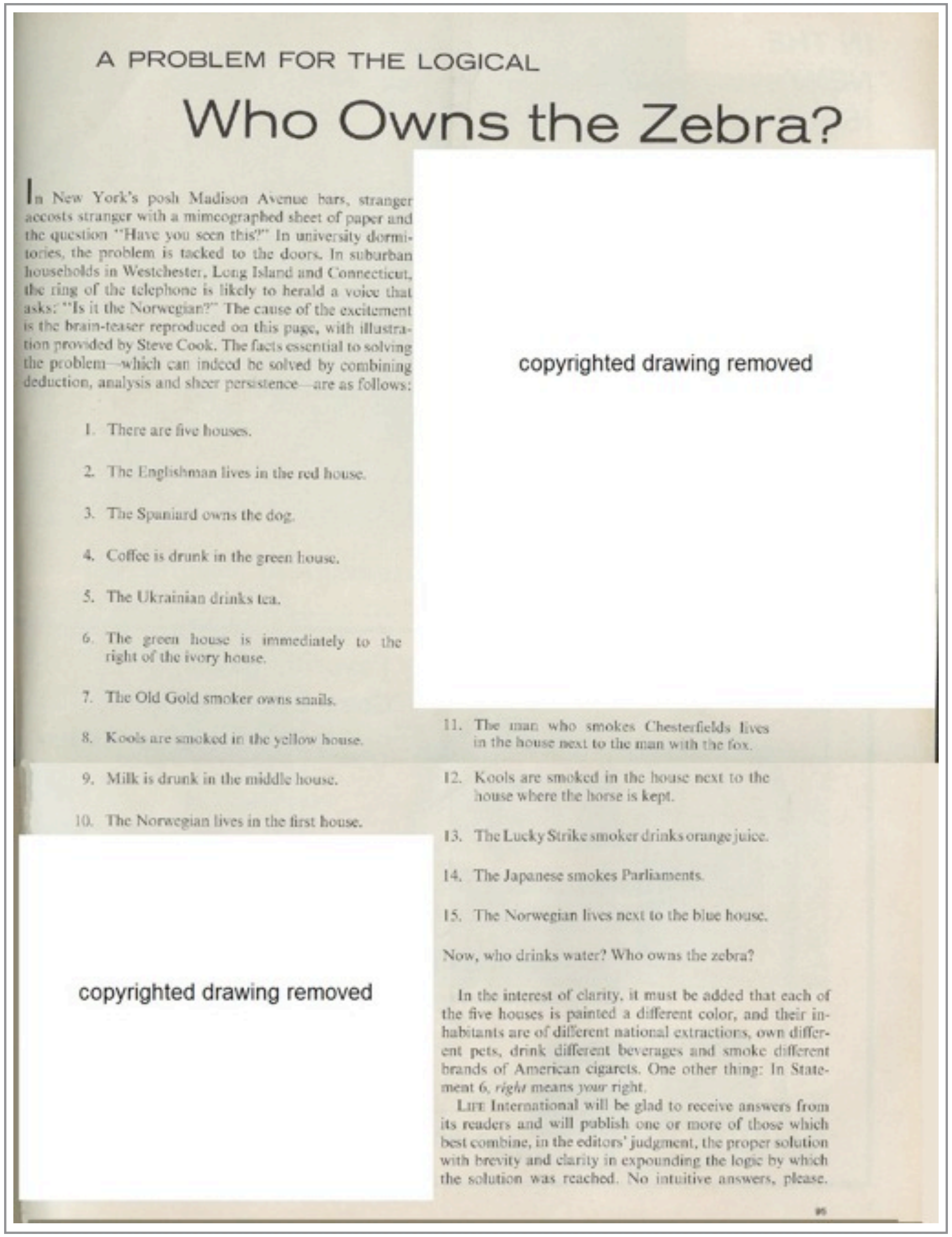
Figure 7. Minimal decision tree for example 3

demo

Logic Programming

TWENTY-FIVE CENTS
DECEMBER 21, 1962
THE SPORT OF THE '60s
TIME
THE WEEKLY NEWSMAGAZINE
GREEN BAY COACH
VINCE LOMBARDI
VOL. LXXX NO. 25

December 17, 1962



A PROBLEM FOR THE LOGICAL

Who Owns the Zebra?

In New York's posh Madison Avenue bars, stranger accosts stranger with a mimeographed sheet of paper and the question "Have you seen this?" In university dormitories, the problem is tacked to the doors. In suburban households in Westchester, Long Island and Connecticut, the ring of the telephone is likely to herald a voice that asks: "Is it the Norwegian?" The cause of the excitement is the brain-teaser reproduced on this page, with illustration provided by Steve Cook. The facts essential to solving the problem—which can indeed be solved by combining deduction, analysis and sheer persistence—are as follows:

copyrighted drawing removed

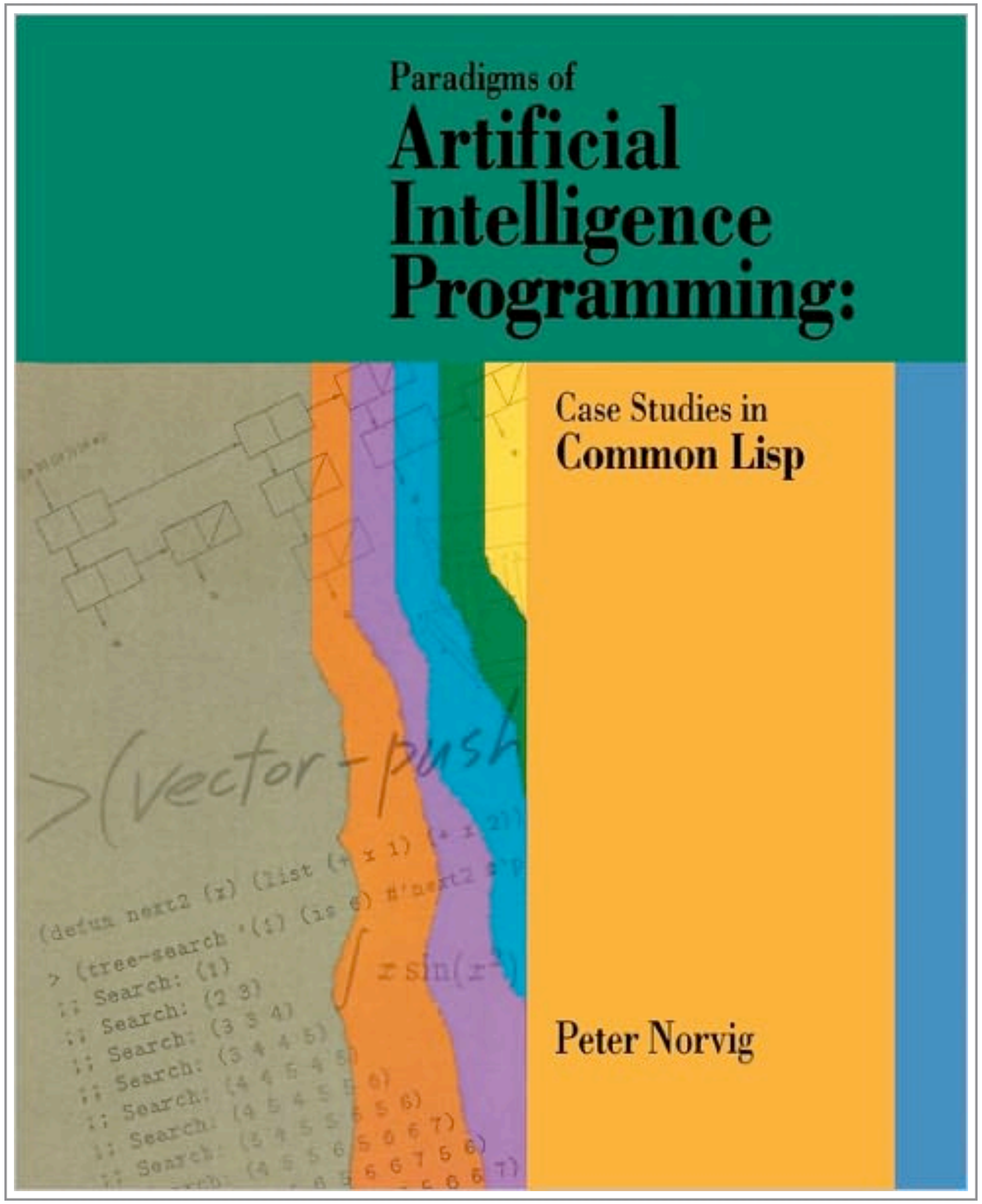
1. There are five houses.
2. The Englishman lives in the red house.
3. The Spaniard owns the dog.
4. Coffee is drunk in the green house.
5. The Ukrainian drinks tea.
6. The green house is immediately to the right of the ivory house.
7. The Old Gold smoker owns snails.
8. Kools are smoked in the yellow house.
9. Milk is drunk in the middle house.
10. The Norwegian lives in the first house.
11. The man who smokes Chesterfields lives in the house next to the man with the fox.
12. Kools are smoked in the house next to the house where the horse is kept.
13. The Lucky Strike smoker drinks orange juice.
14. The Japanese smokes Parliaments.
15. The Norwegian lives next to the blue house.

copyrighted drawing removed

Now, who drinks water? Who owns the zebra?

In the interest of clarity, it must be added that each of the five houses is painted a different color, and their inhabitants are of different national extractions, own different pets, drink different beverages and smoke different brands of American cigarets. One other thing: In Statement 6, *right* means *your* right.

LIFE International will be glad to receive answers from its readers and will publish one or more of those which best combine, in the editors' judgment, the proper solution with brevity and clarity in expounding the logic by which the solution was reached. No intuitive answers, please.


Paradigms of

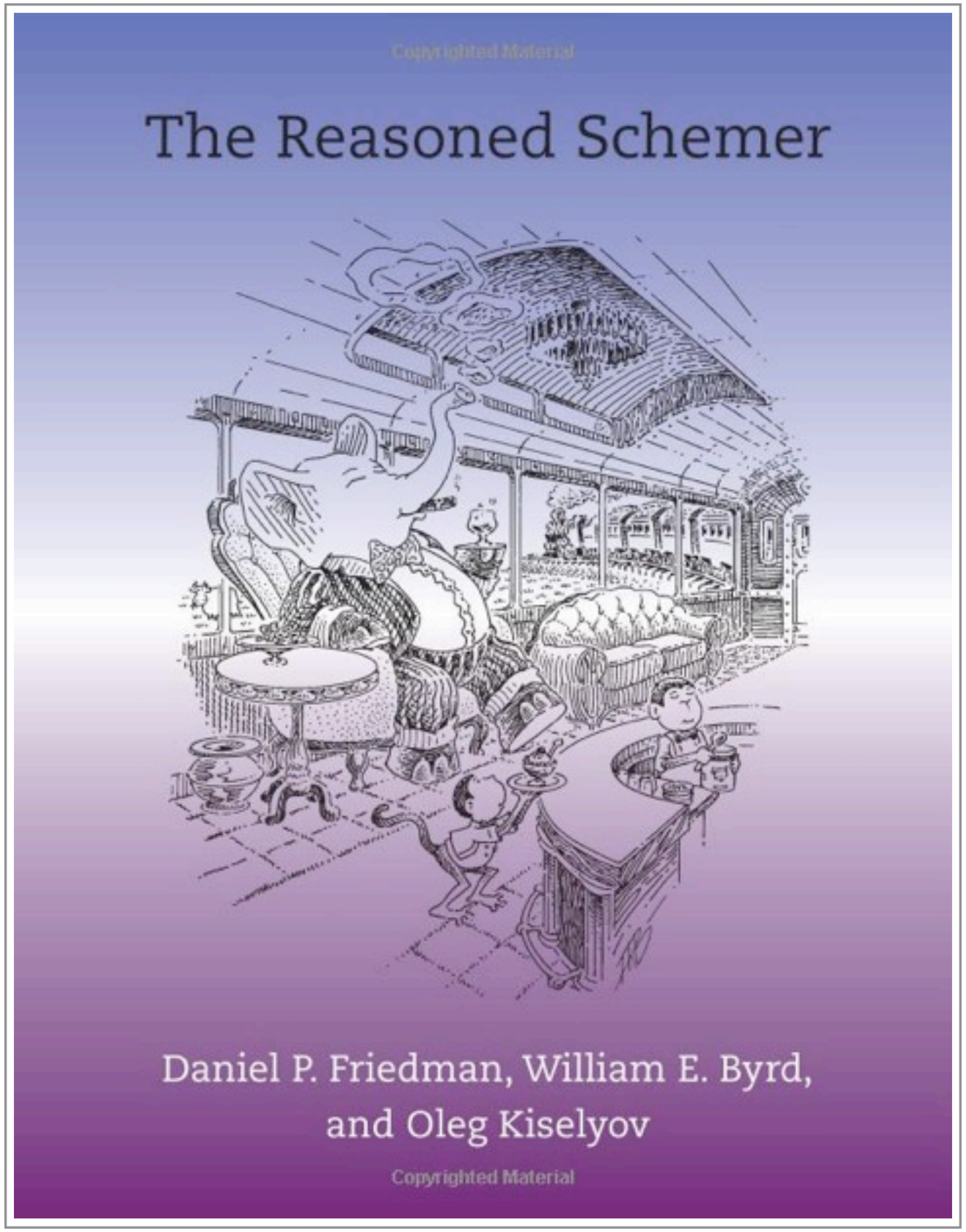
Artificial Intelligence Programming:

Case Studies in Common Lisp

Peter Norvig

As Peter Norvig points out,
24 billion candidate solutions

demo



The Reasoned Schemer

Daniel P. Friedman, William E. Byrd, and Oleg Kiselyov

Types

demo

JavaScript

JavaScript

- A solid target for programming languages

JavaScript

- A solid target for programming languages
  - We need more efficient implementations of programming languages - uncover new implementation strategies

JavaScript

- A solid target for programming languages
  - We need more efficient implementations of programming languages – uncover new implementation strategies
- JavaScript is not likely to go anywhere

JavaScript

- A solid target for programming languages
  - We need more efficient implementations of programming languages - uncover new implementation strategies
- JavaScript is not likely to go anywhere
  - But do we still want to be programming JavaScript 10 years from now?

Questions?