Immutability, Interactivity & JavaScript

cognitect

Copyrighted Material

The Dream Machine.
J. C. R. Licklider
and the Revolution
That Made
Computing Personal.

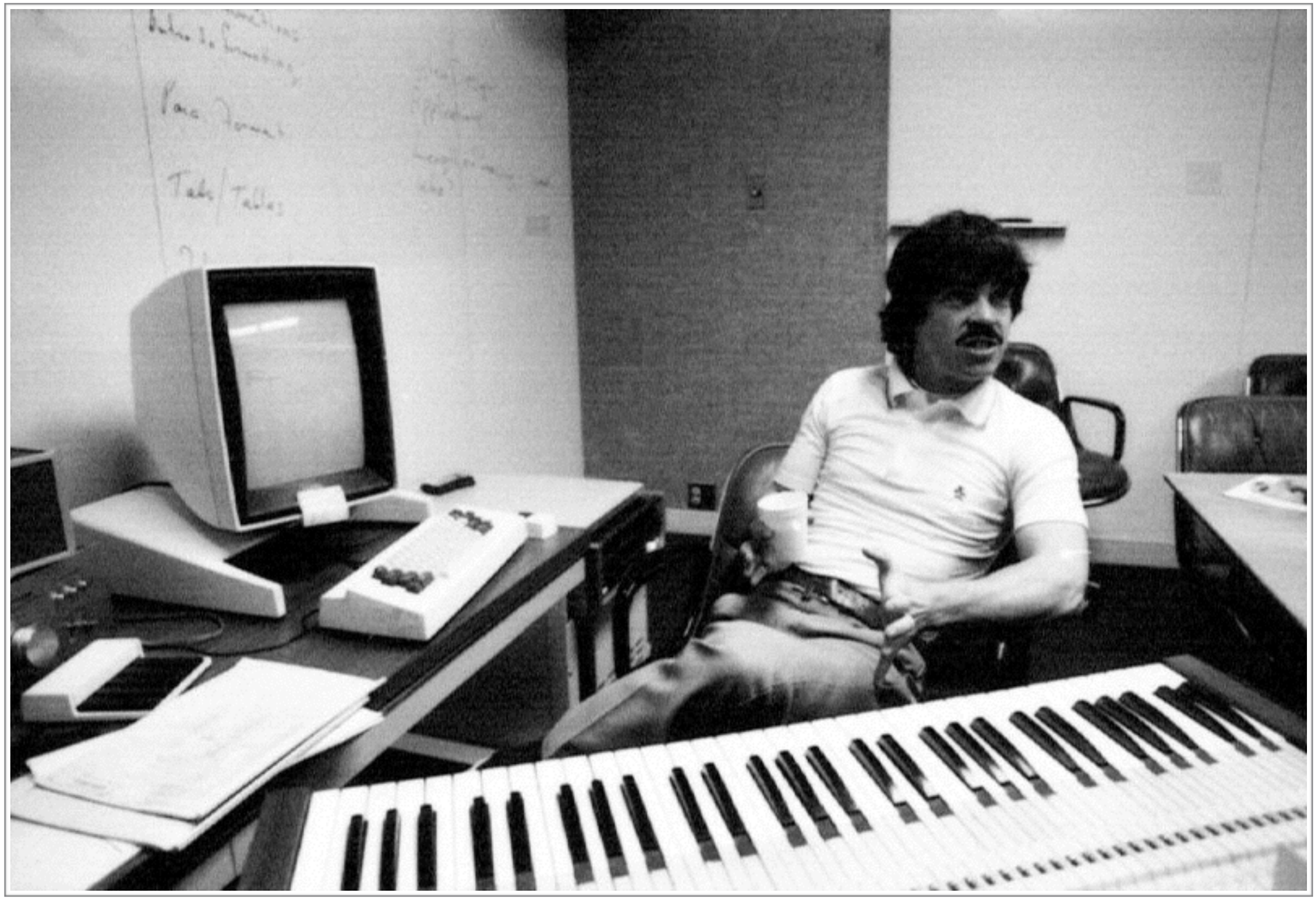
M. Mitchell Waldrop.
author of Complexity.

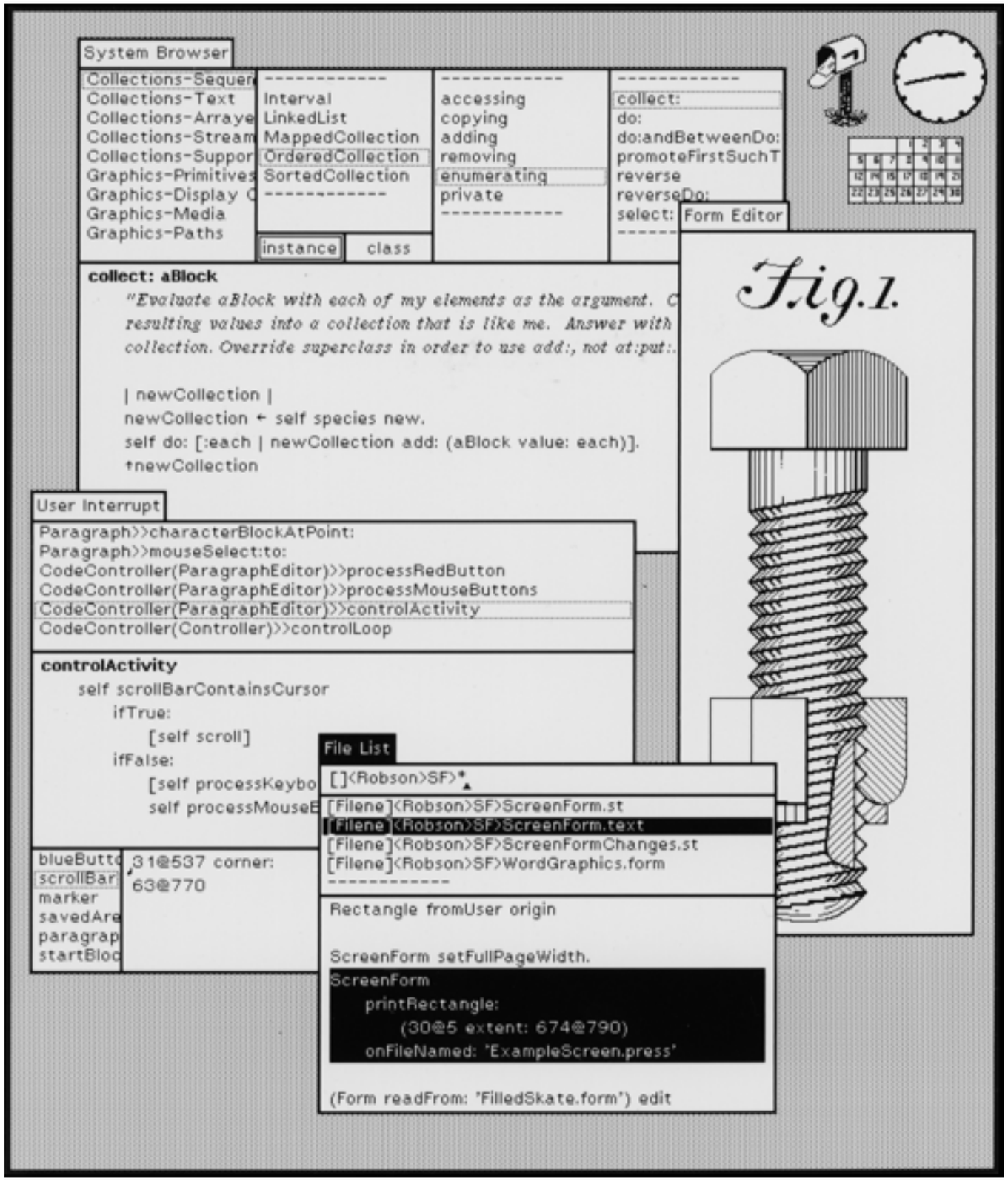
"Waldrop's account of [Licklider's] and many others' world-transforming contributions is compelling."
—John Allen Paulos, The New York Times Book Review

Copyrighted Material

The trie memory scheme is inefficient for small memories, but it be- comes increasingly efficient in using available storage space as memory size increases. The attractive features of the scheme are these: 1) The retrieval process is extremely simple. Given the argument, enter the standard ini- tial register with the first character, and pick up the address of the second. Then go to the second register, and pick up the address of the third, etc. 2) If two arguments have initial characters in common, they use the same storage space for those characters.

-J.C.R. Licklider, "Man-Computer Symbiosis" 1960

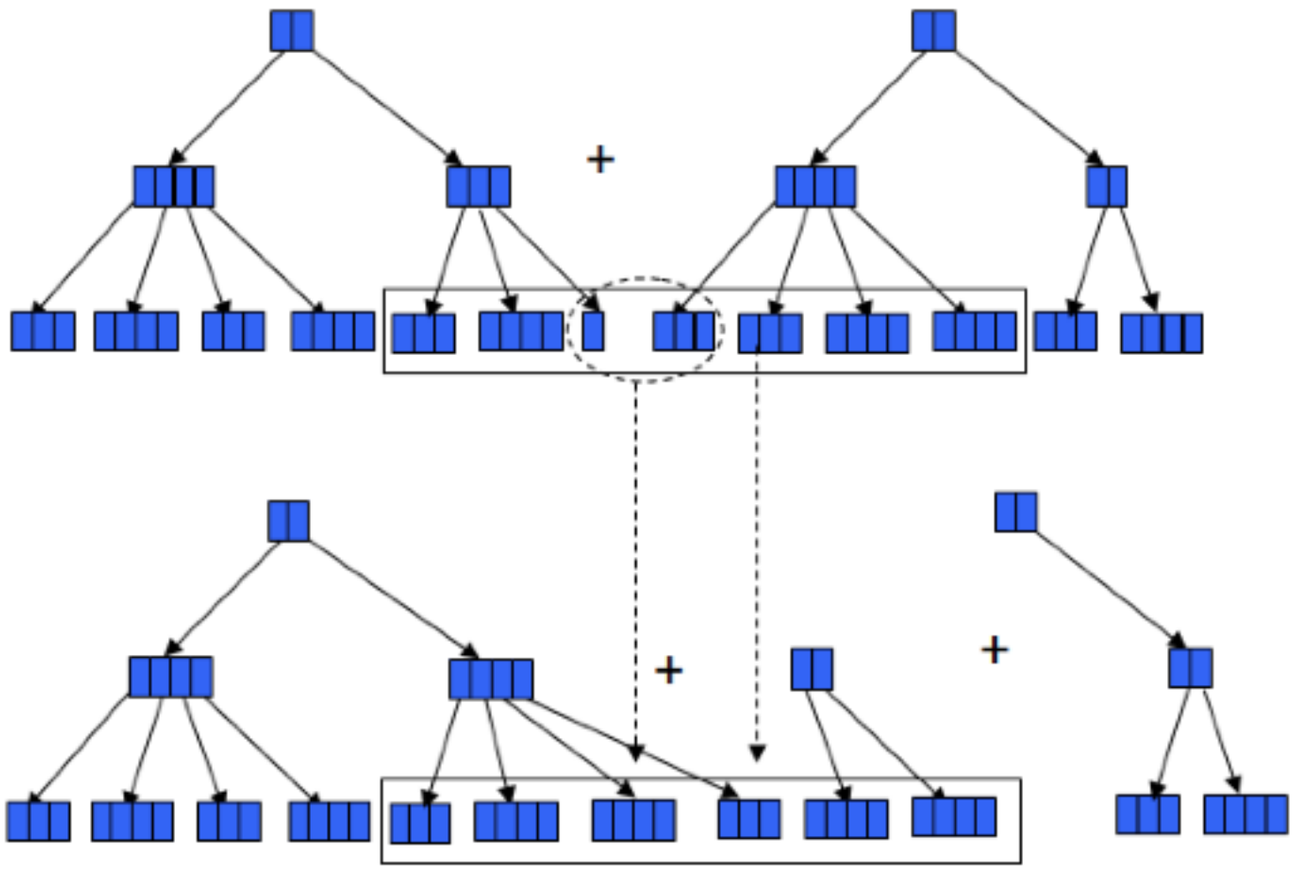



System Browser
Collections-Sequer
Collections-Text
Collections-Arraye
Collections-Stream
Collections-Suppor
Graphics-Primitives
Graphics-Display C
Graphics-Media
Graphics-Paths
Interval
LinkedList
MappedCollection
OrderedCollection
SortedCollection
accessing
copying
adding
removing
enumerating
private
collect:
do:
do:andBetweenDo:
promoteFirstSuchT
reverse
reverseDo:
select:
instance
class
collect: aBlock
"Evaluate aBlock with each of my elements as the argument. C
resulting values into a collection that is like me. Answer with
collection. Override superclass in order to use add:, not at:put:.
| newCollection |
newCollection ← self species new.
self do: [:each | newCollection add: (aBlock value: each)].
↑newCollection
Form Editor
Fig.1.
User Interrupt
Paragraph>>characterBlockAtPoint:
Paragraph>>mouseSelect:to:
CodeController(ParagraphEditor)>>processRedButton
CodeController(ParagraphEditor)>>processMouseButtons
CodeController(ParagraphEditor)>>controlActivity
CodeController(Controller)>>controlLoop
controlActivity
self scrollBarContainsCursor
ifTrue:
[self scroll]
ifFalse:
[self processKeybo
self processMouseE
File List
[]<Robson>SF>*
[Filene]<Robson>SF>ScreenForm.st
[Filene]<Robson>SF>ScreenForm.text
[Filene]<Robson>SF>ScreenFormChanges.st
[Filene]<Robson>SF>WordGraphics.form
Rectangle fromUser origin
ScreenForm setFullPageWidth.
ScreenForm
printRectangle:
(30@5 extent: 674@790)
onFileNamed: 'ExampleScreen.press'
(Form readFrom: 'FilledSkate.form') edit
31@537 corner:
63@770

Model-View-Controller

- first formulated by Trygve Reenskaug Adele Goldberg and others at Xerox PARC in 1979
- long shadow, the basic concepts still prevalent today.

- At a very abstract level MVC is a sound separation of concerns
- Implementations leave much to be desired
  - *Stateful objects everywhere*

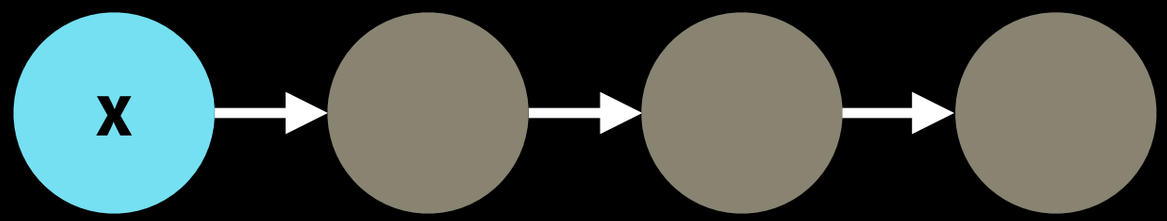


HACKER
SCHOOL
2012

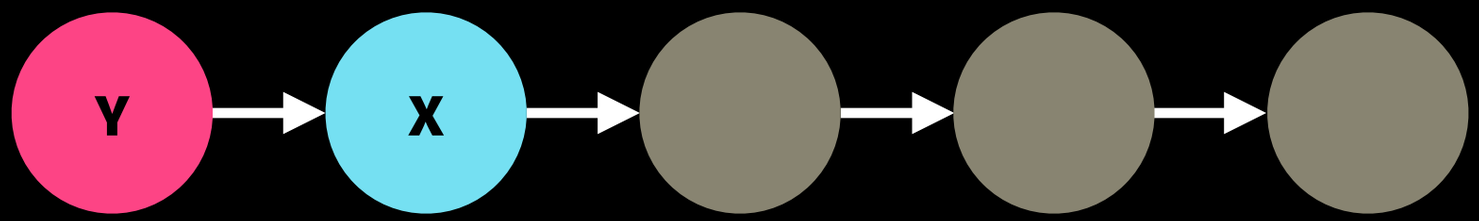
Functional Programming and Data

- immutable values, not mutable objects
- “change” returns a new value, leaving the old one unmodified
- they’re persistent
- they’re fast

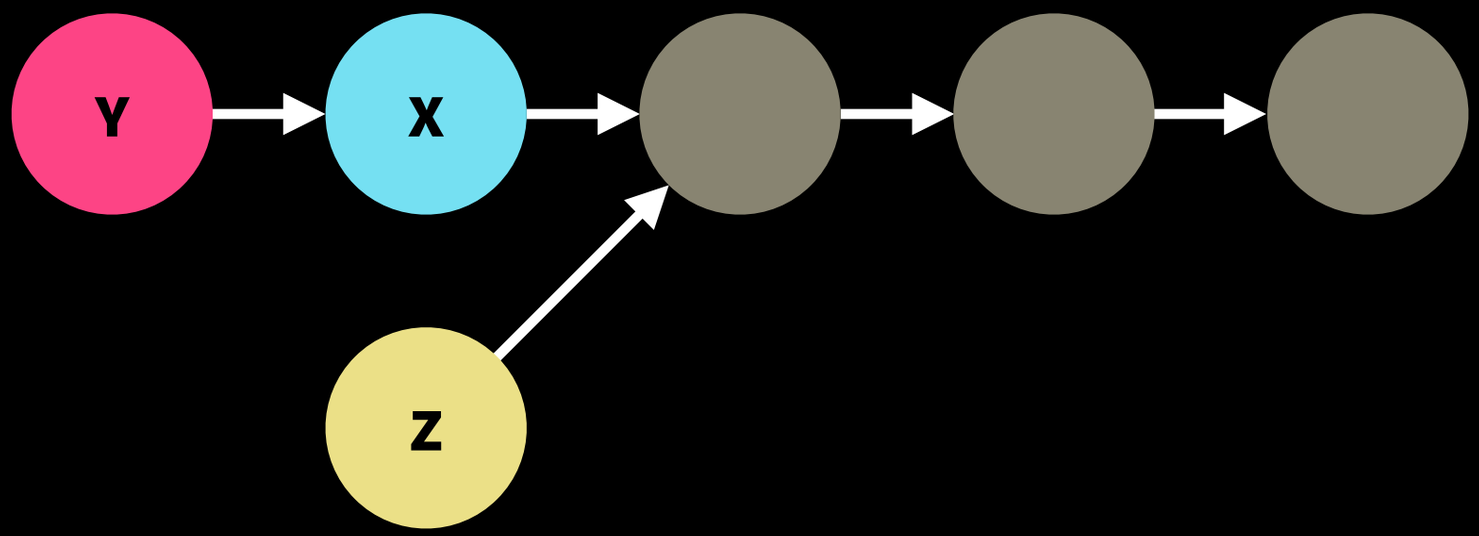
Simple example: Linked List



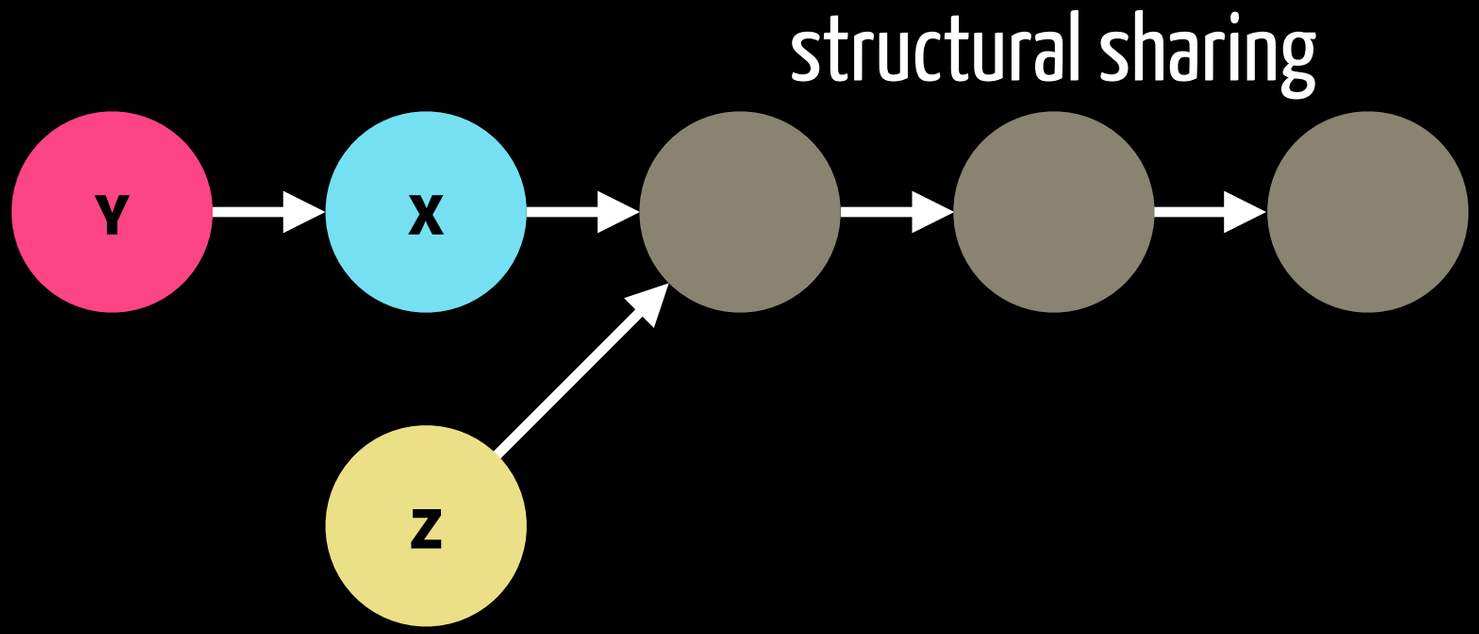
Simple example: Linked List



Simple example: Linked List



Simple example: Linked List



Sharing structure

- space efficiency
- computational efficiency – avoids copying

Phil Bagwell

- Array Mapped Trie
- Hash Array Mapped Trie

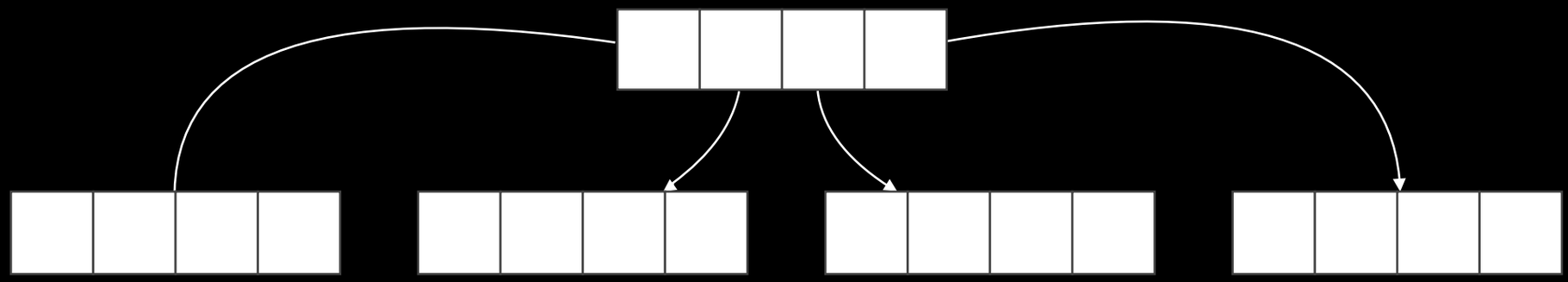
Bitmapped Vector Trie

- data lives in the leaves
- e.g. prefix tree used for string lookup
- bitwise trie

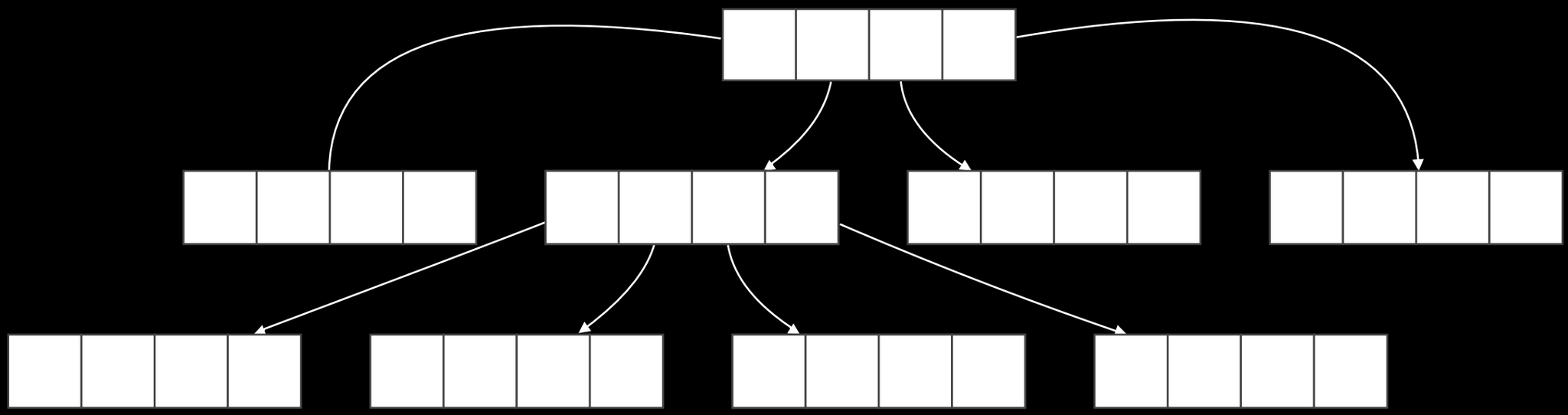
Persistent Vector

Persistent Vector

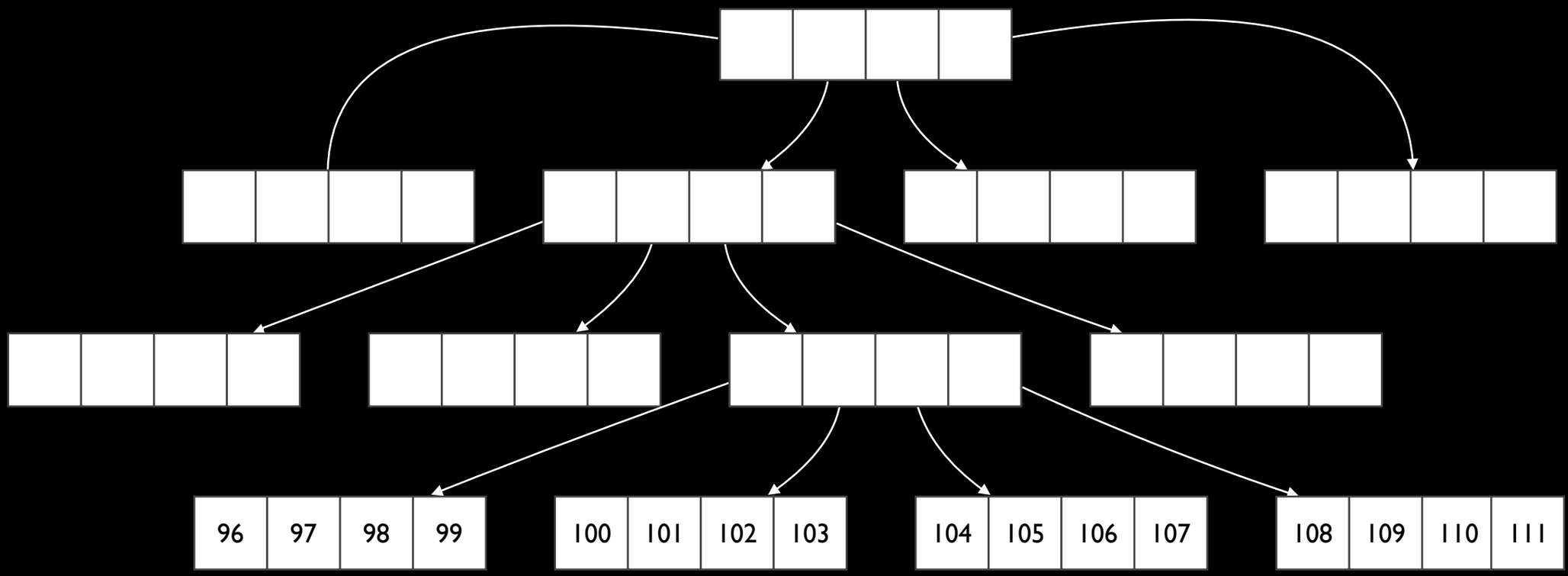
Persistent Vector



Persistent Vector



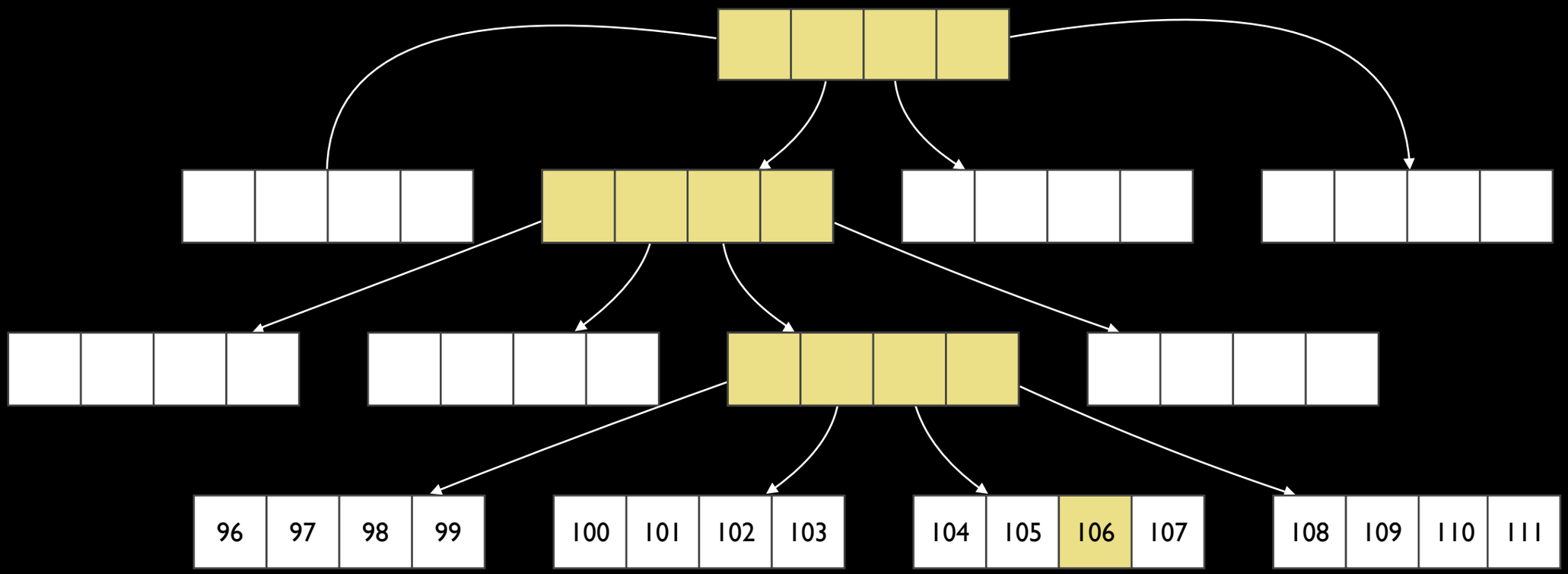
Persistent Vector



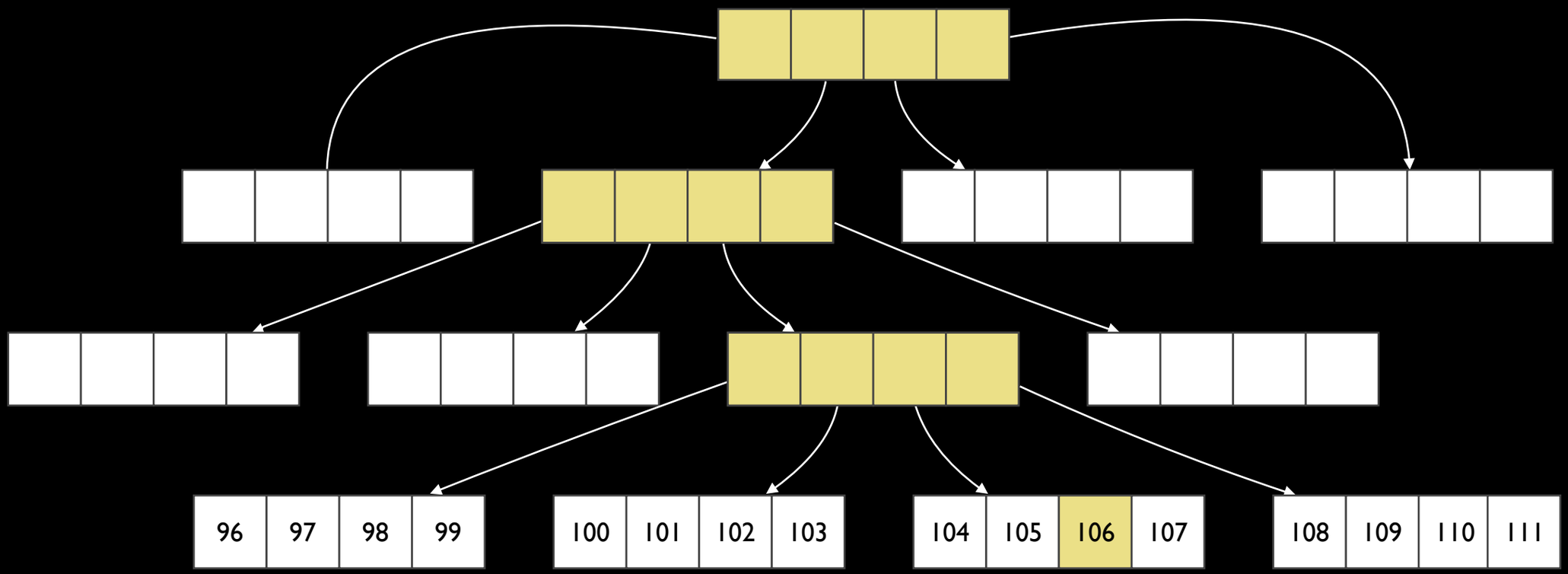
Persistent Vector

`getindex`

Persistent Vector

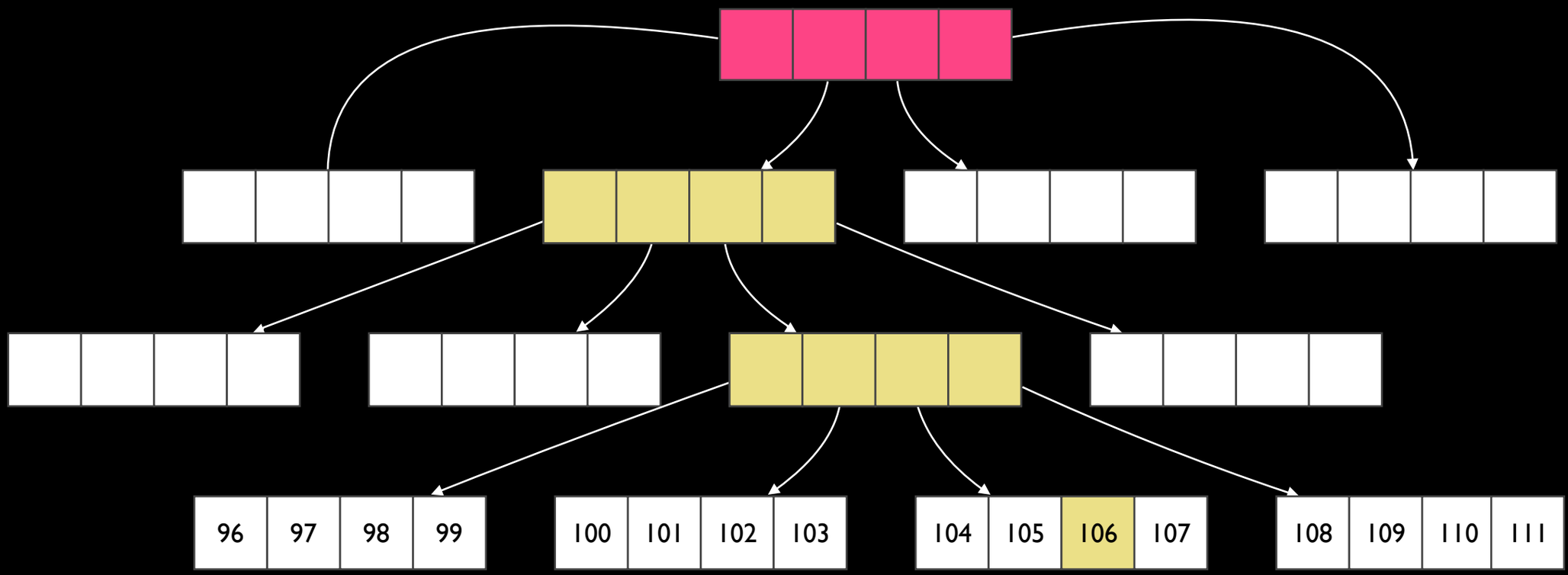


Persistent Vector



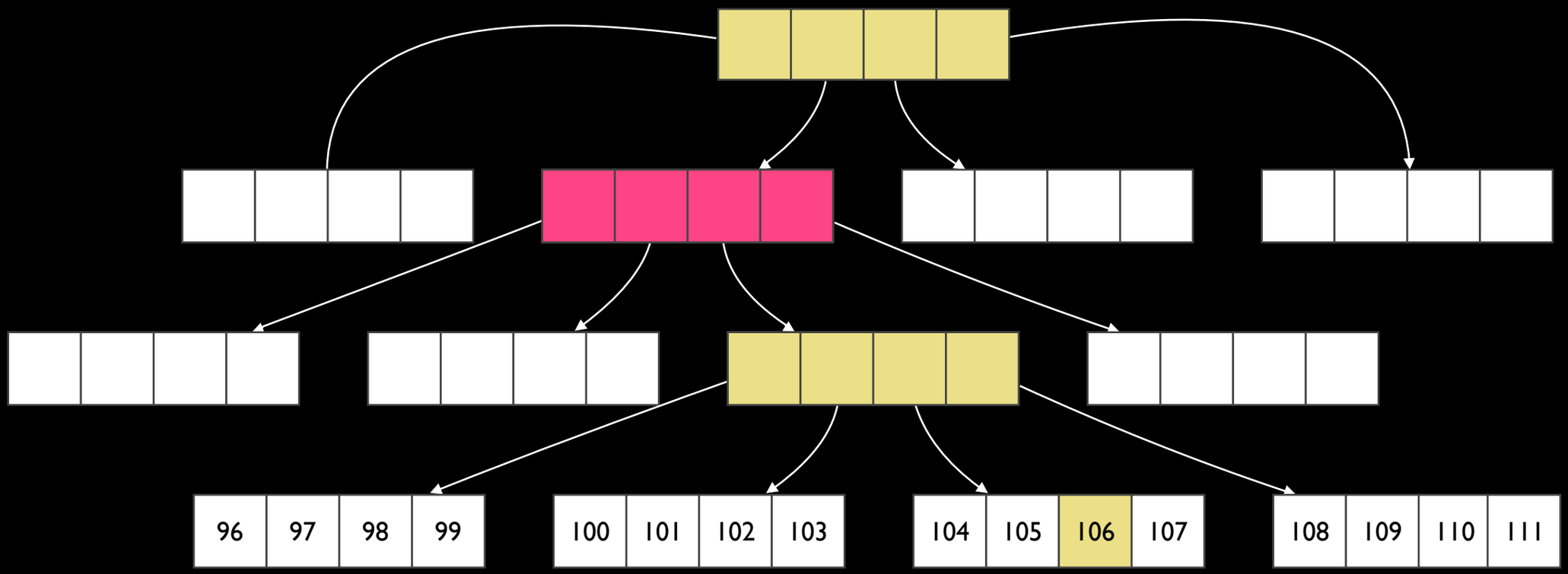
0b01101010

Persistent Vector



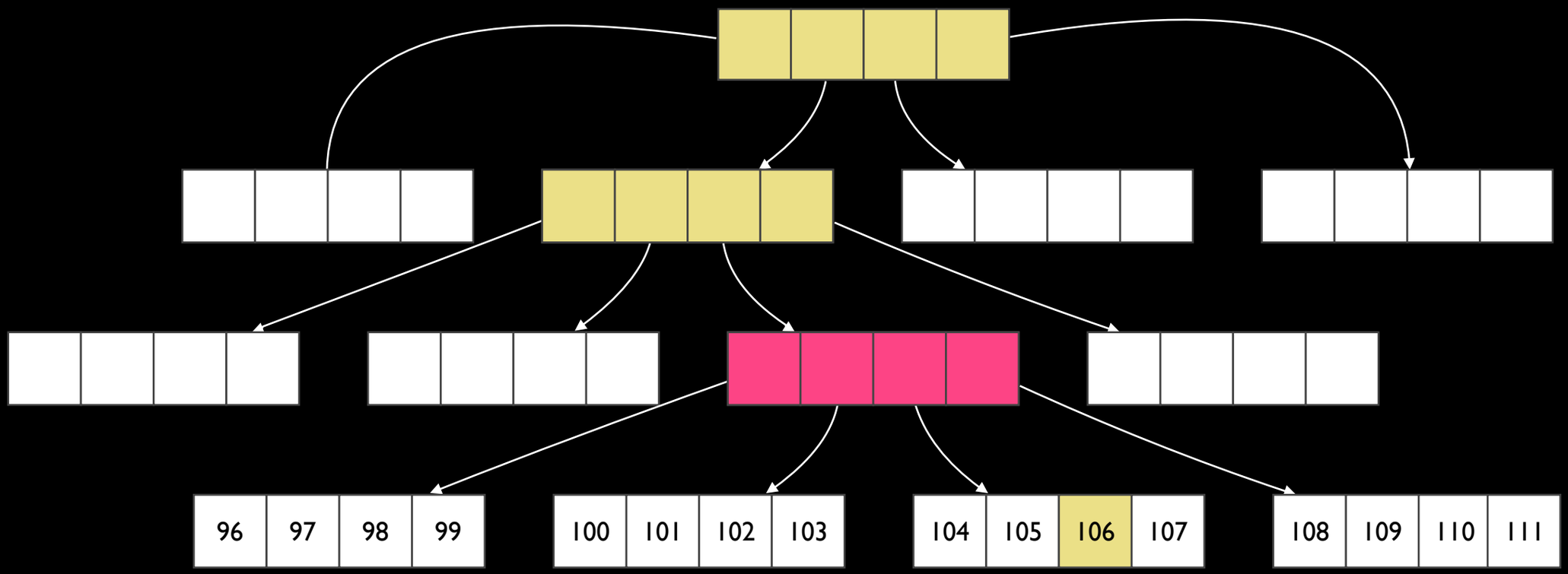
0b01101010

Persistent Vector



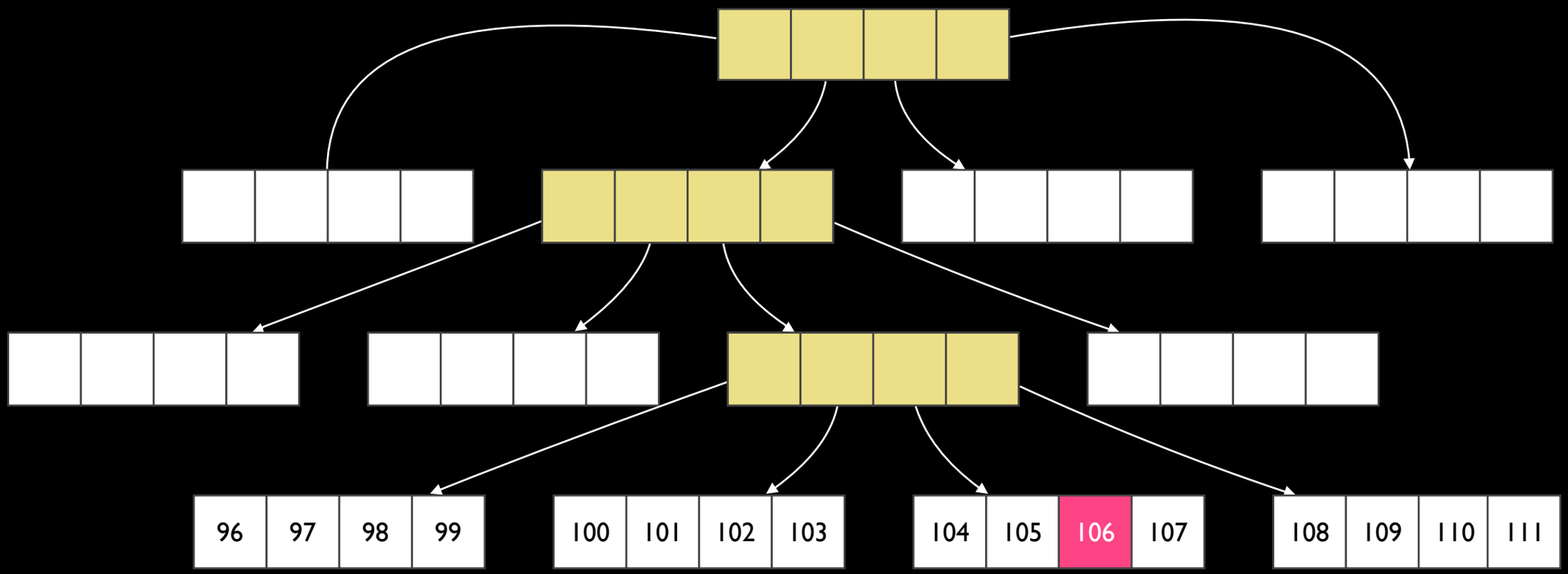
0b01101010

Persistent Vector



0b01101010

Persistent Vector

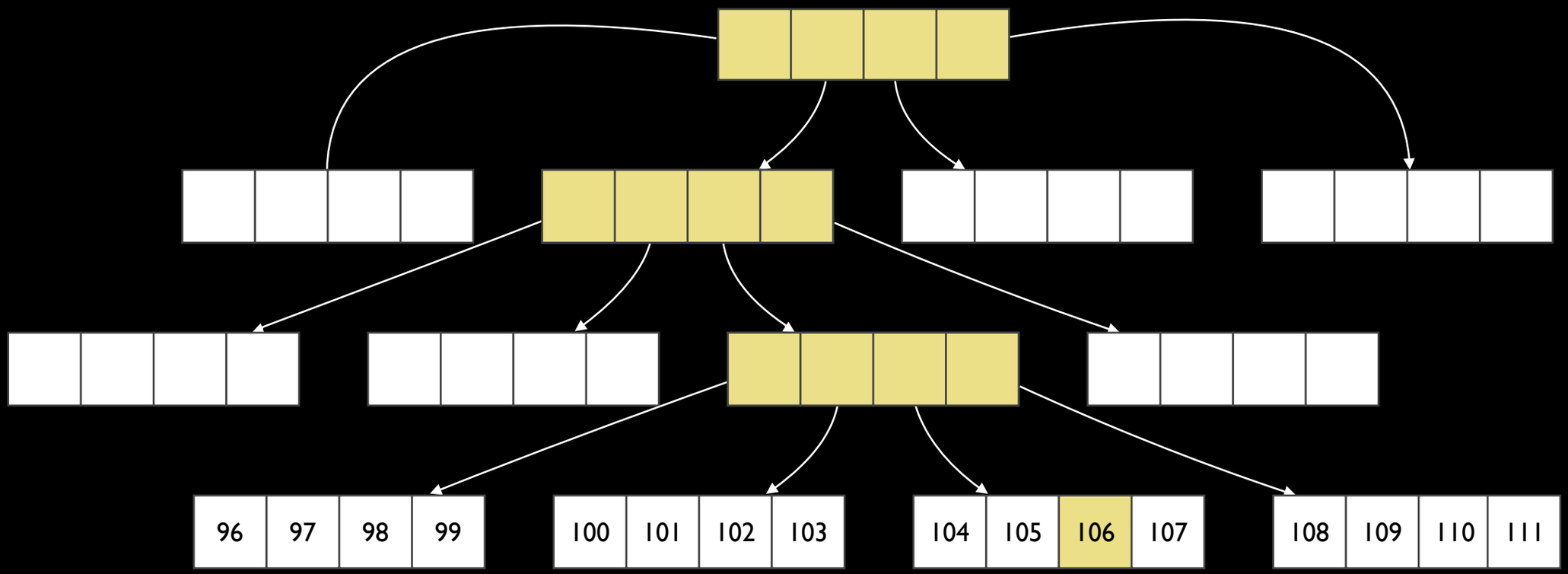


0b01101010

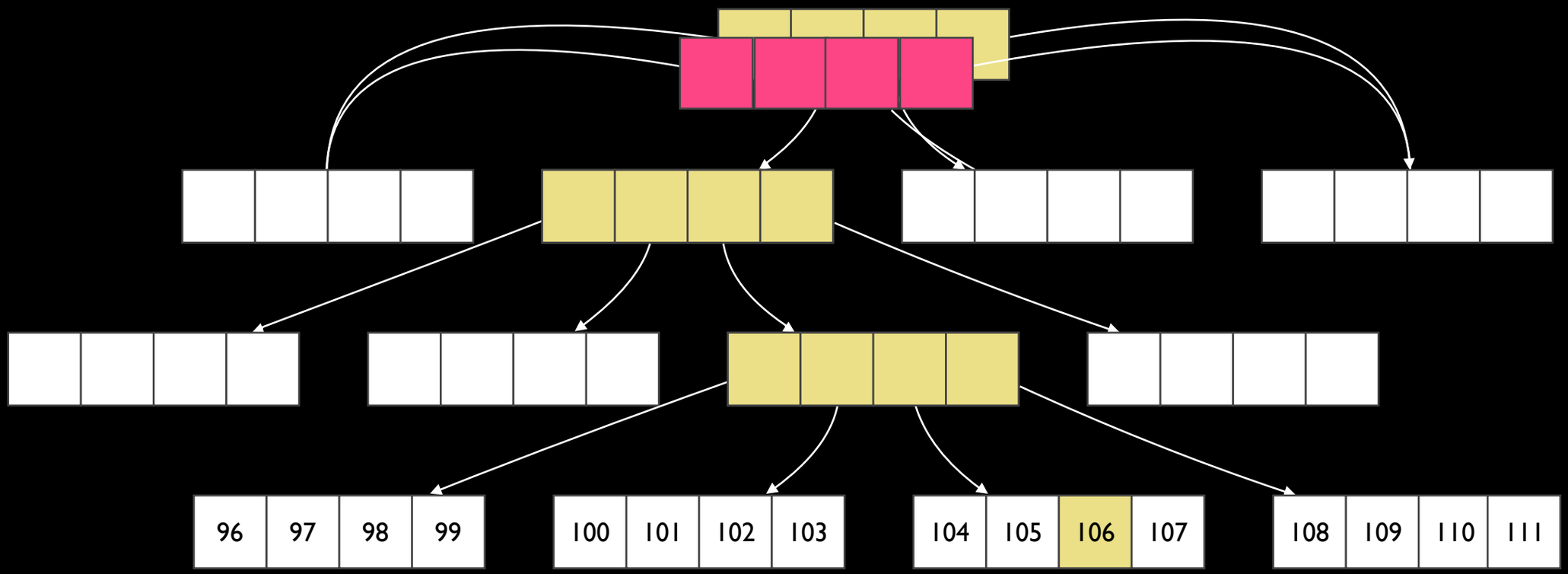
Persistent Vector

assoc

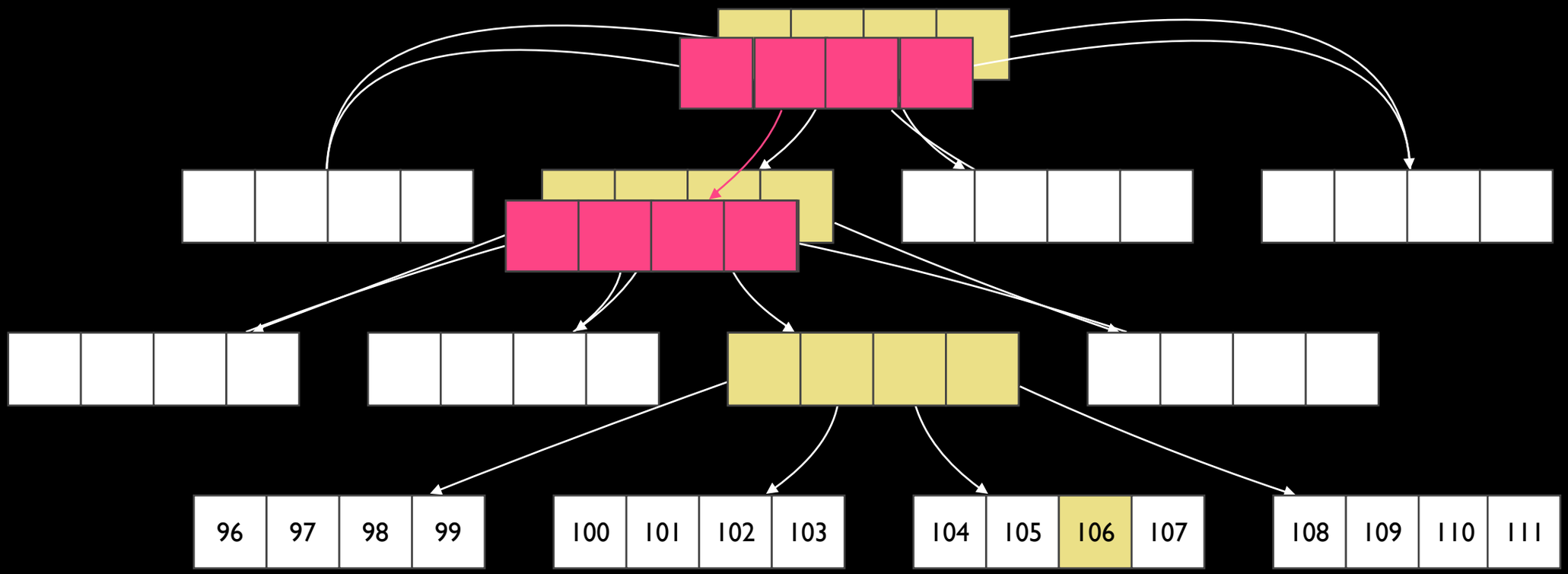
Persistent Vector



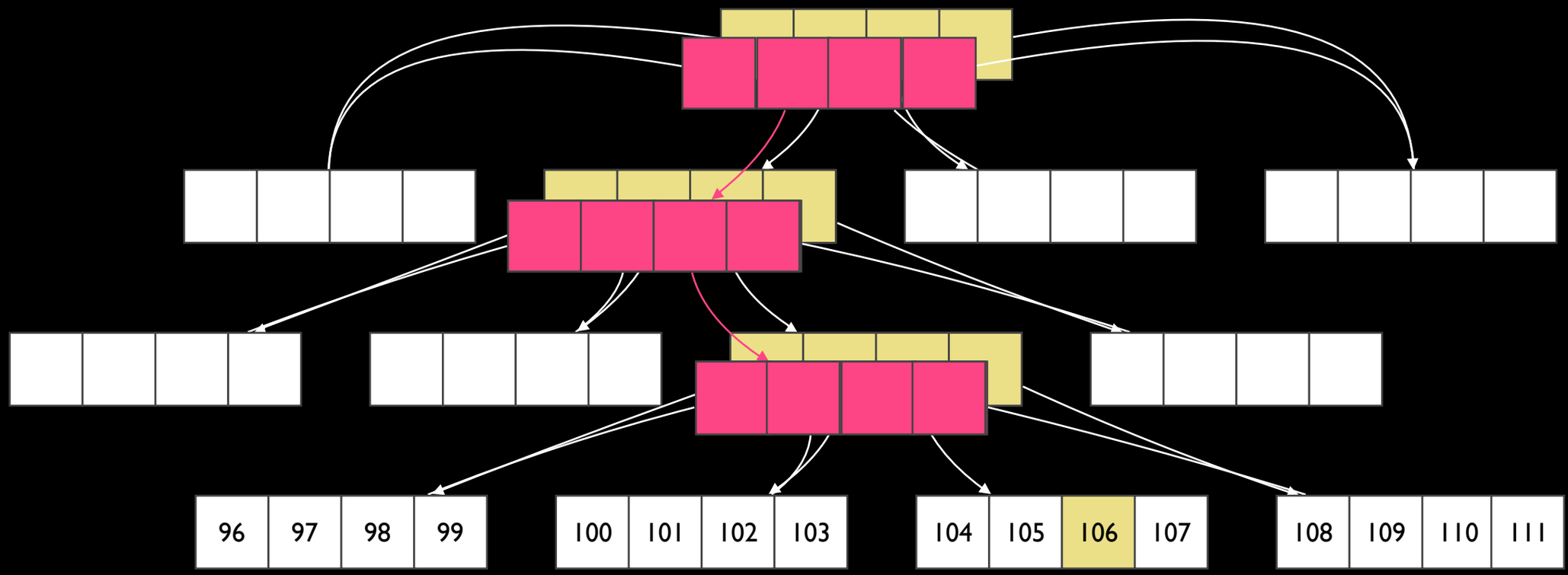
Persistent Vector



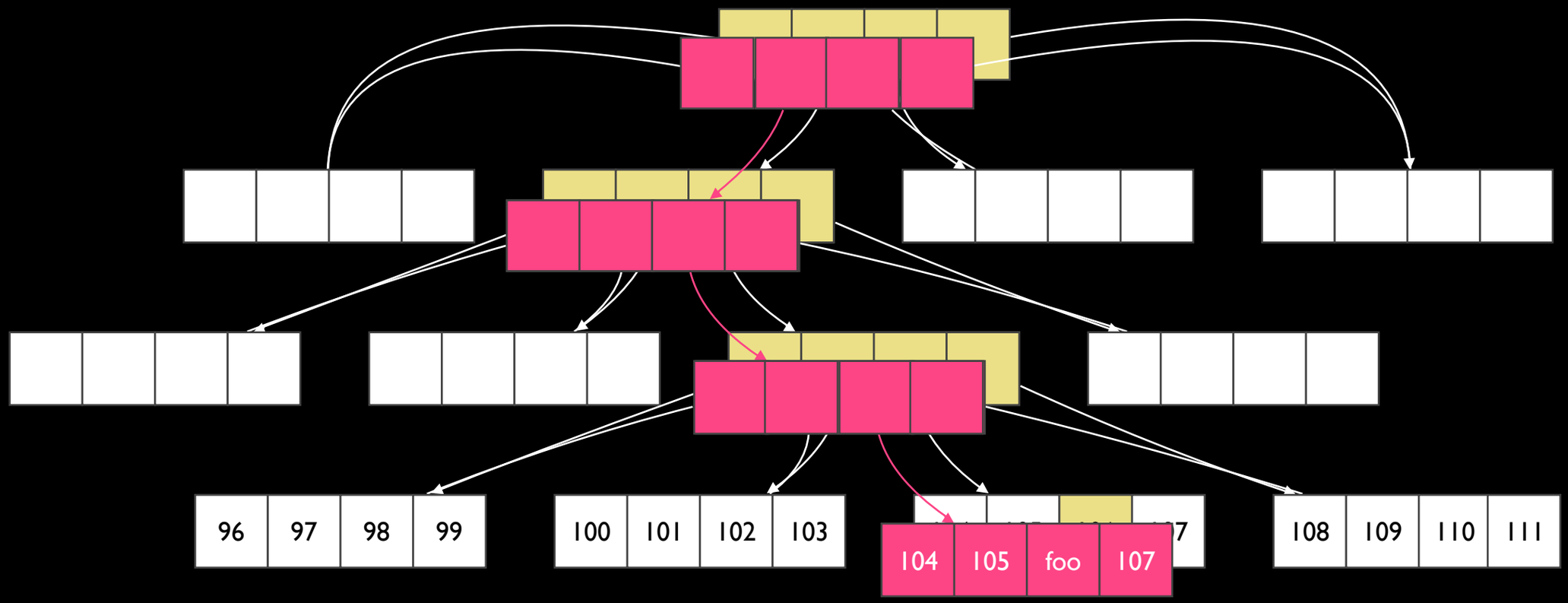
Persistent Vector



Persistent Vector



Persistent Vector



Persistent Vector

Length 4 internal vectors?

Persistent Vector

32

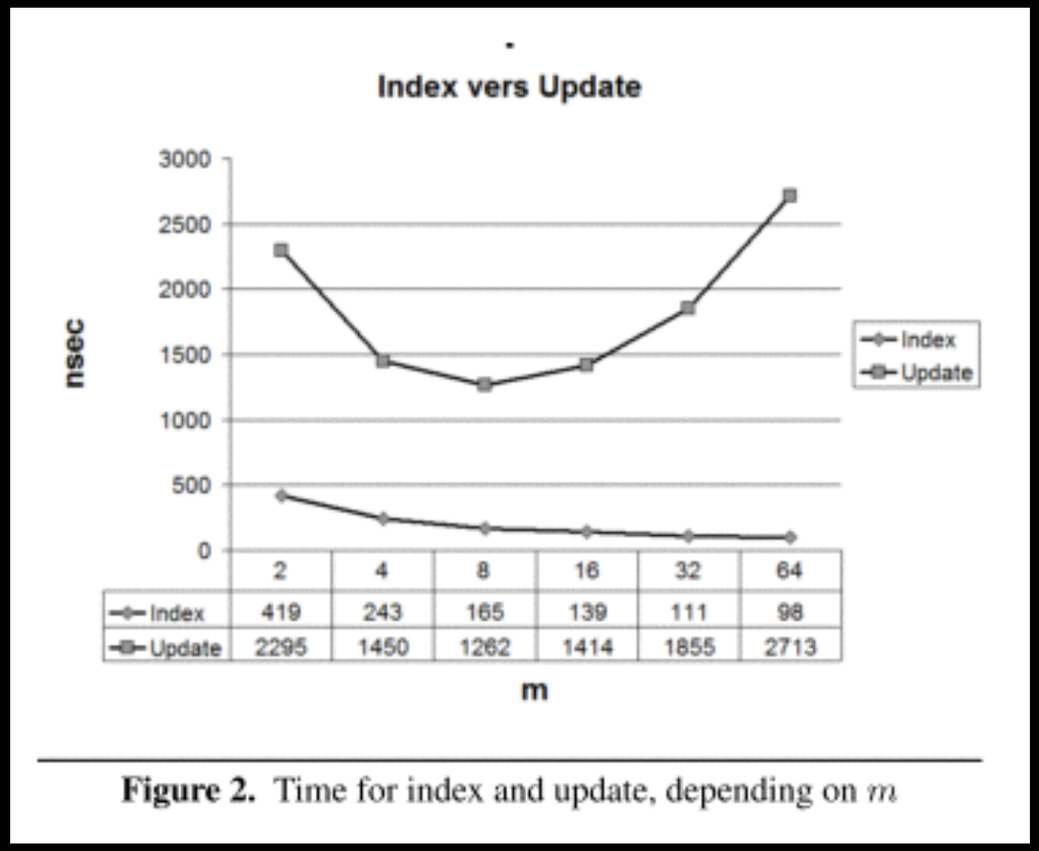


Figure 2. Time for index and update, depending on m

From Bagwell, Rompf 2011

$$32^7$$

34,359,738,368 elements

Om

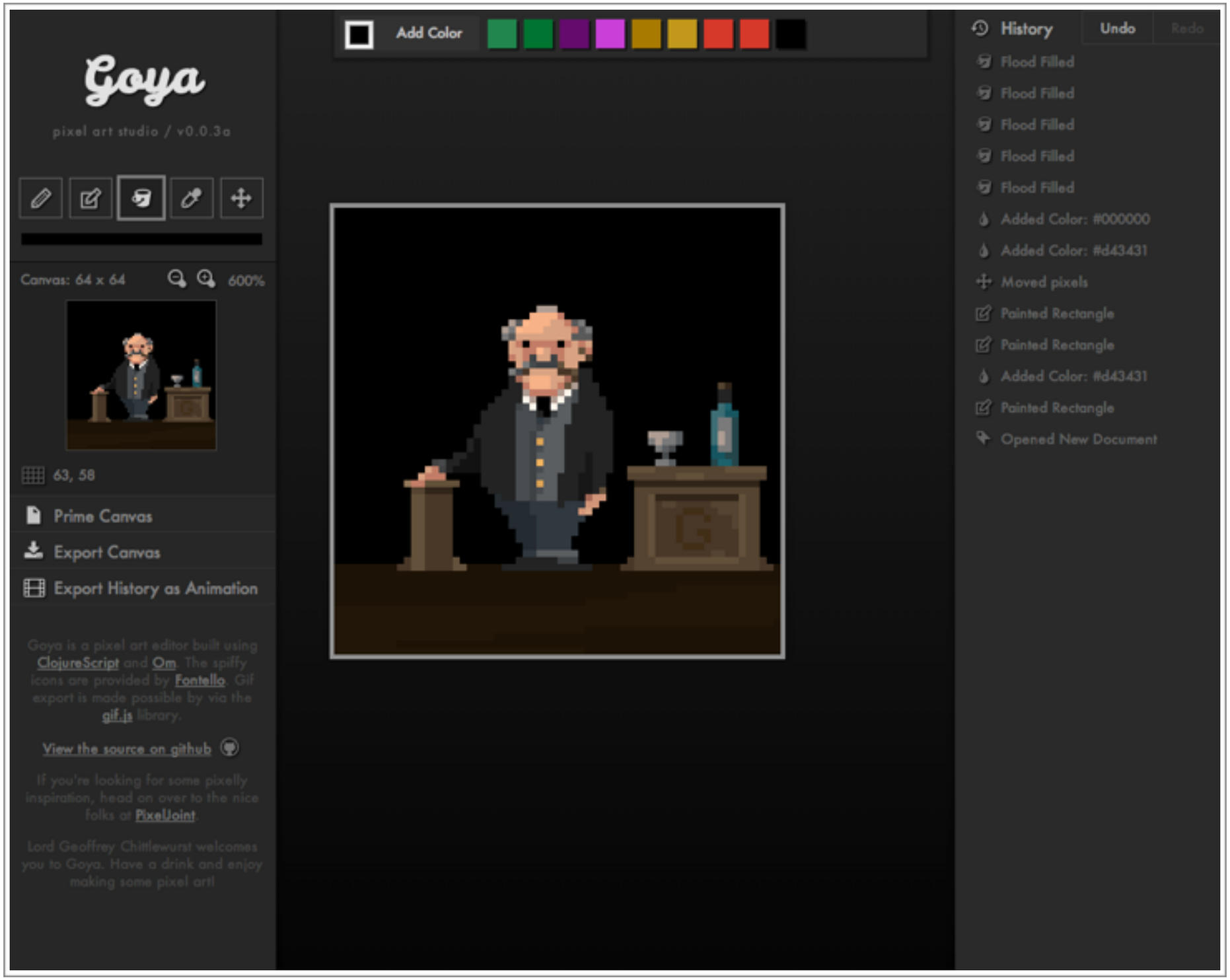
React | A JavaScript library
facebook.github.io/react/
React
React
A JAVASCRIPT LIBRARY FOR BUILDING USER INTERFACES

$$f(D_0) = V_0$$

$$\mathrm{f}(\mathrm{D}_1) = \mathbf{V}_1$$

$$\mathrm{diff}(\mathrm{V}_0,\mathrm{V}_1) = \mathrm{CHANGES}$$

Om Demo


Goya
pixel art studio / v0.0.3a
Add Color
History
Undo
Redo
Flood Filled
Flood Filled
Flood Filled
Flood Filled
Flood Filled
Added Color: #000000
Added Color: #d43431
Moved pixels
Painted Rectangle
Painted Rectangle
Added Color: #d43431
Painted Rectangle
Opened New Document
Canvas: 64 x 64
600%
63, 58
Prime Canvas
Export Canvas
Export History as Animation
Goya is a pixel art editor built using ClojureScript and Om. The spiffy icons are provided by Fontello. Gif export is made possible by via the gif.js library.
View the source on github
If you're looking for some pixelly inspiration, head on over to the nice folks at PixelJoint.
Lord Geoffrey Chittlewurst welcomes you to Goya. Have a drink and enjoy making some pixel art!

demo

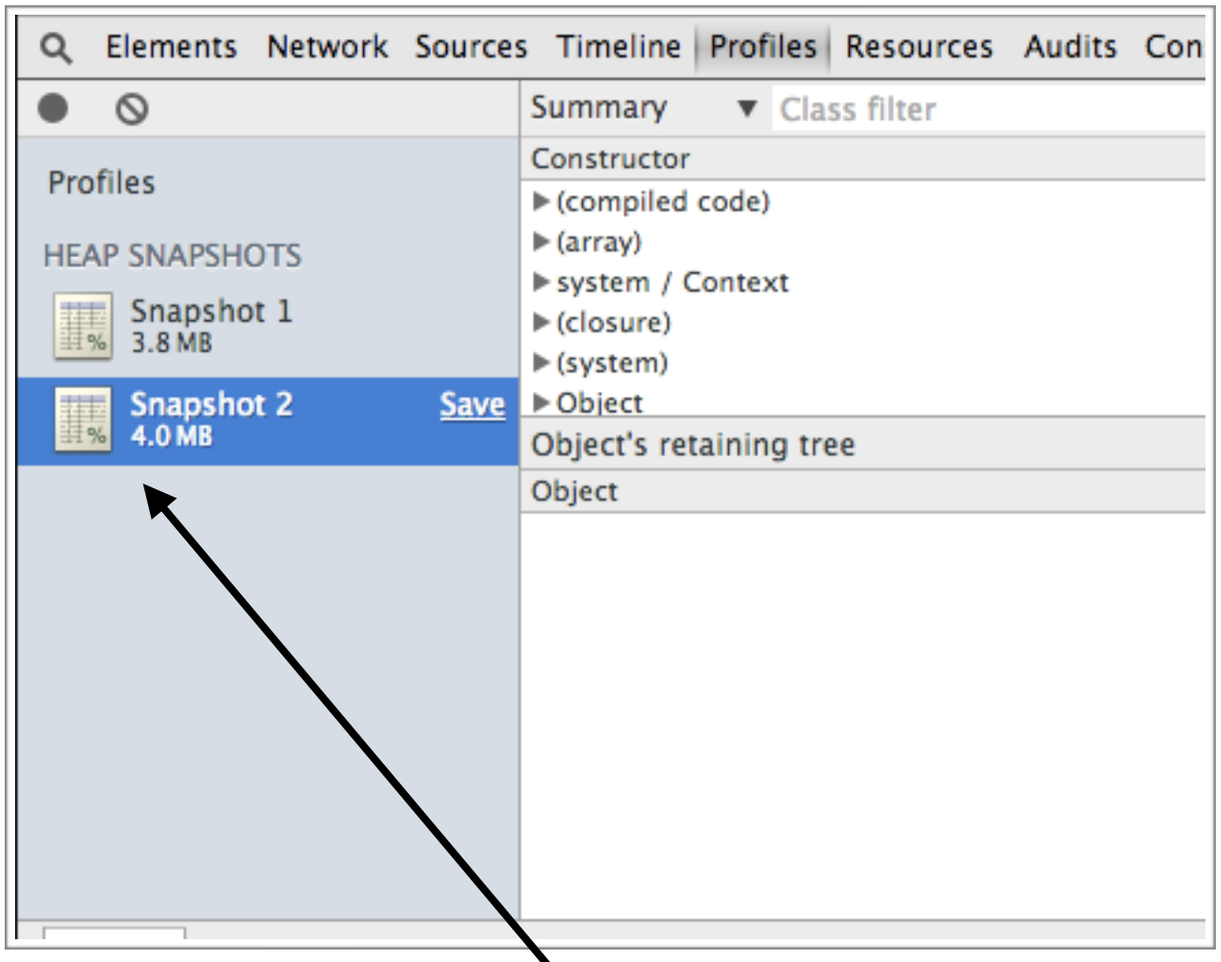
branch: master · goya / src / cljs / goya / timemachine.cljs

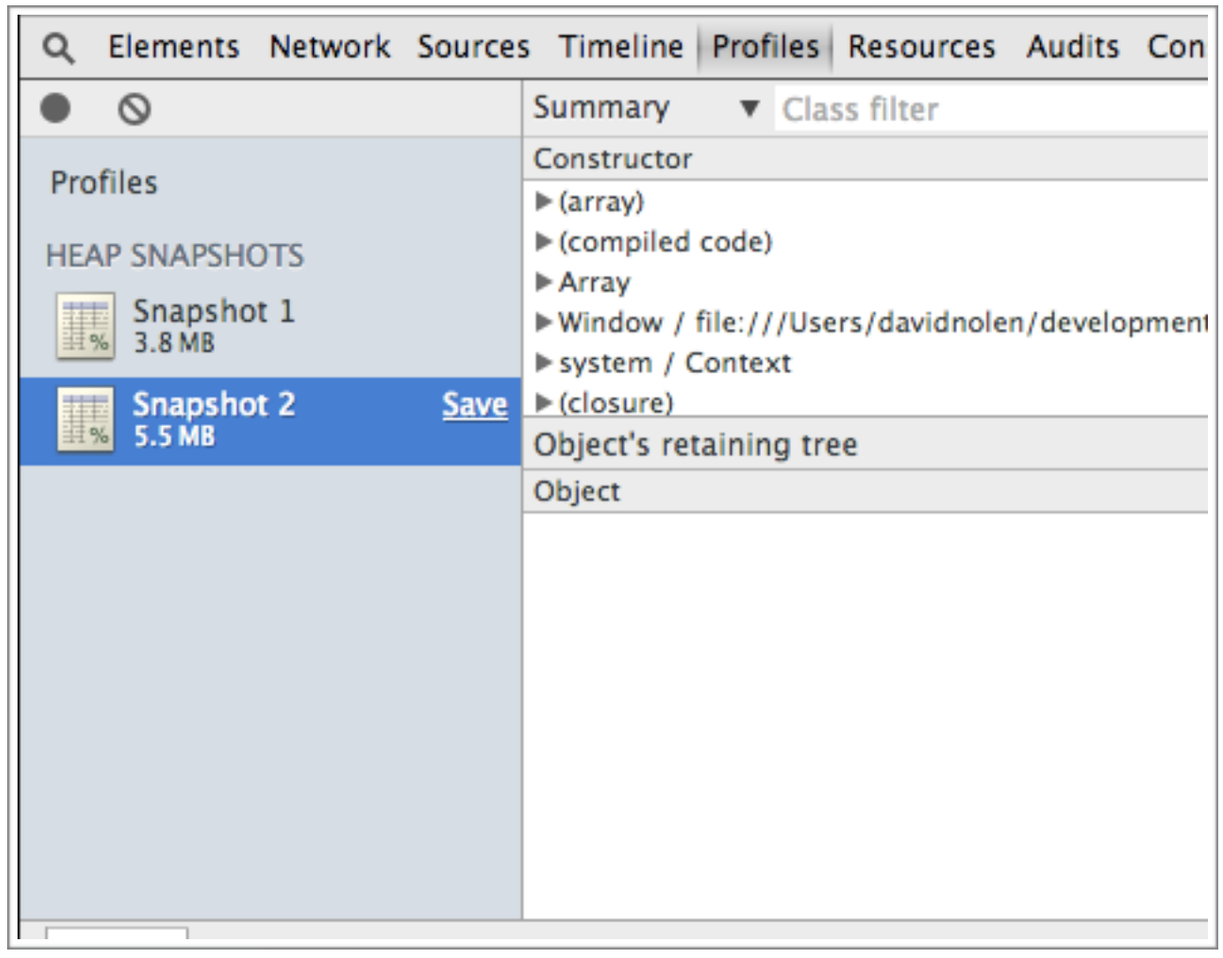
swannodette 13 days ago Project layout refactor, better production settings

1 contributor

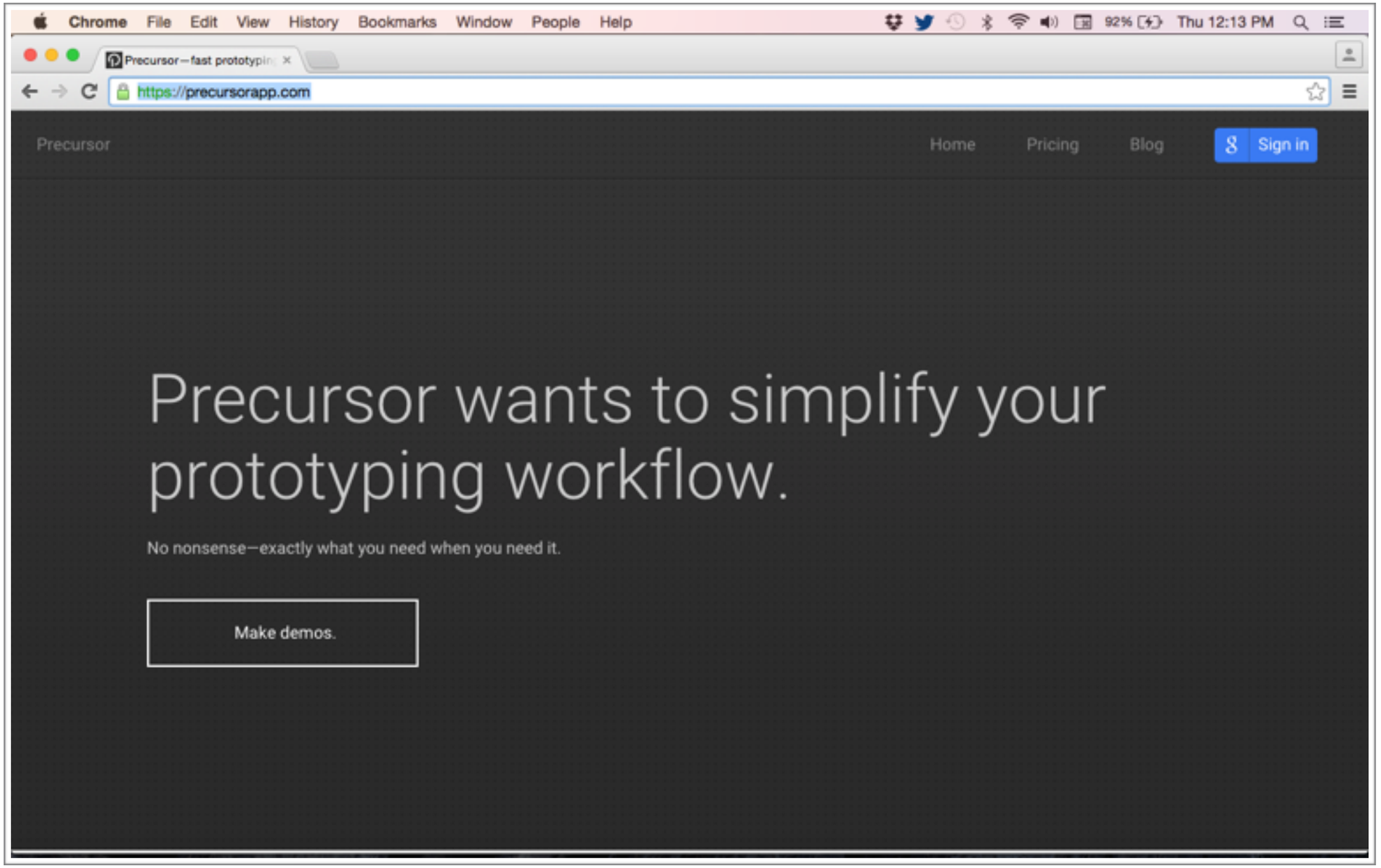
file 62 lines (41 sloc) 1.85 kb — Open | Edit | Raw | Blame | History | Delete

```
(ns goya.timemachine
        (:require [goya.appstate :as app]
            [goya.previewstate :as previewstate]))


;; ================================================================================
;; Credits to David Nolen's Time Travel blog post.

(def app-history (atom [(get-in @app/app-state [:main-app])]))
(def app-future (atom []))



;; ================================================================================

(defn update-preview []
  (reset! previewstate/preview-state
          (assoc-in @previewstate/preview-state [:main-app :image-data]
                    (get-in @app/app-state [:main-app :image-data]))))

(defn show-history-preview [idx]
  (reset! previewstate/preview-state
          (assoc-in @previewstate/preview-state [:main-app :image-data]
                    (get-in (nth @app-history idx) [:image-data]))))

(add-watch app/app-state :preview-watcher
  (fn [_ _ _ _] (update-preview)))



(defn undo-is-possible []
  (> (count @app-history) 1))

(defn redo-is-possible []
  (> (count @app-future) 0))


(defn push-onto-undo-stack [new-state]
  (let [old-watchable-app-state (last @app-history)]
    (when-not (= old-watchable-app-state new-state)
      (swap! app-history conj new-state))))


(defn do-undo []
  (when (undo-is-possible)
    (swap! app-future conj (last @app-history))
    (swap! app-history pop)
    (reset! app/app-state (assoc-in @app/app-state [:main-app] (last @app-history)))))

(defn do-redo []
  (when (redo-is-possible)
    (reset! app/app-state (assoc-in @app/app-state [:main-app] (last @app-future)))
    (push-onto-undo-stack (last @app-future))
    (swap! app-future pop)))


(defn handle-transaction [tx-data root-cursor]
  (when (= (:tag tx-data) :add-to-undo)
    (reset! app-future [])
    (let [new-state (get-in (:new-state tx-data) [:main-app])]
      (push-onto-undo-stack new-state))))
```





Persistent Data Structures ... ROCK


Chrome File Edit View History Bookmarks Window People Help
92% Thu 12:13 PM
Precursor—fast prototypin
https://precursorapp.com
Precursor
Home
Pricing
Blog
Sign in
Precursor wants to simplify your prototyping workflow.
No nonsense—exactly what you need when you need it.
Make demos.

facebook / **immutable-js**

Unwatch 96 | Unstar 2,683 | Fork 85

Immutable Data Collections for Javascript

368 commits | **1** branch | **14** releases | **15** contributors

branch: **master** | **immutable-js** / +

Update README.md ···

leebyron authored 9 days ago — latest commit `9fedc9883a`

__tests__	Ensure equality works correctly for Set. #96	21 days ago
dist	Ensure equality works correctly for Set. #96	21 days ago
resources	lowercase require module name, simplifying case-sensitive file systems.	a month ago
src	Ensure equality works correctly for Set. #96	21 days ago
type-definitions	renamed deepMerge -> mergeDeep	14 days ago
.gitignore	Clean up gruntfile, add dist	3 months ago
CONTRIBUTING.md	Moving over to fb's team github page	2 months ago
Gruntfile.js	Use unminified source in node, minified in scripts. #69	a month ago

Code

Issues 27

Pull Requests 0

Wiki

Pulse

Graphs

HTTPS clone URL

https://github.com

You can clone with HTTPS, SSH, or Subversion.

Clone in Desktop

Download ZIP

```
var rdr = transit.reader("json", {
  arrayBuilder: {
    init: function(node) { return Immutable.List().asMutable(); },
    add: function(ret, val, node) { return ret.push(val); },
    finalize: function(ret, node) { return ret.asImmutable(); },
    fromArray: function(arr, node) {
      return Immutable.List.from(arr);
    }
  },
  mapBuilder: {
    init: function(node) { return Immutable.Map().asMutable(); },
    add: function(ret, key, val, node) { return ret.set(key, val); },
    finalize: function(ret, node) { return ret.asImmutable(); }
  }
});
```

Links

- ClojureScript
- React
- Om
- transit-js
- mori
- Immutable-js

Questions?