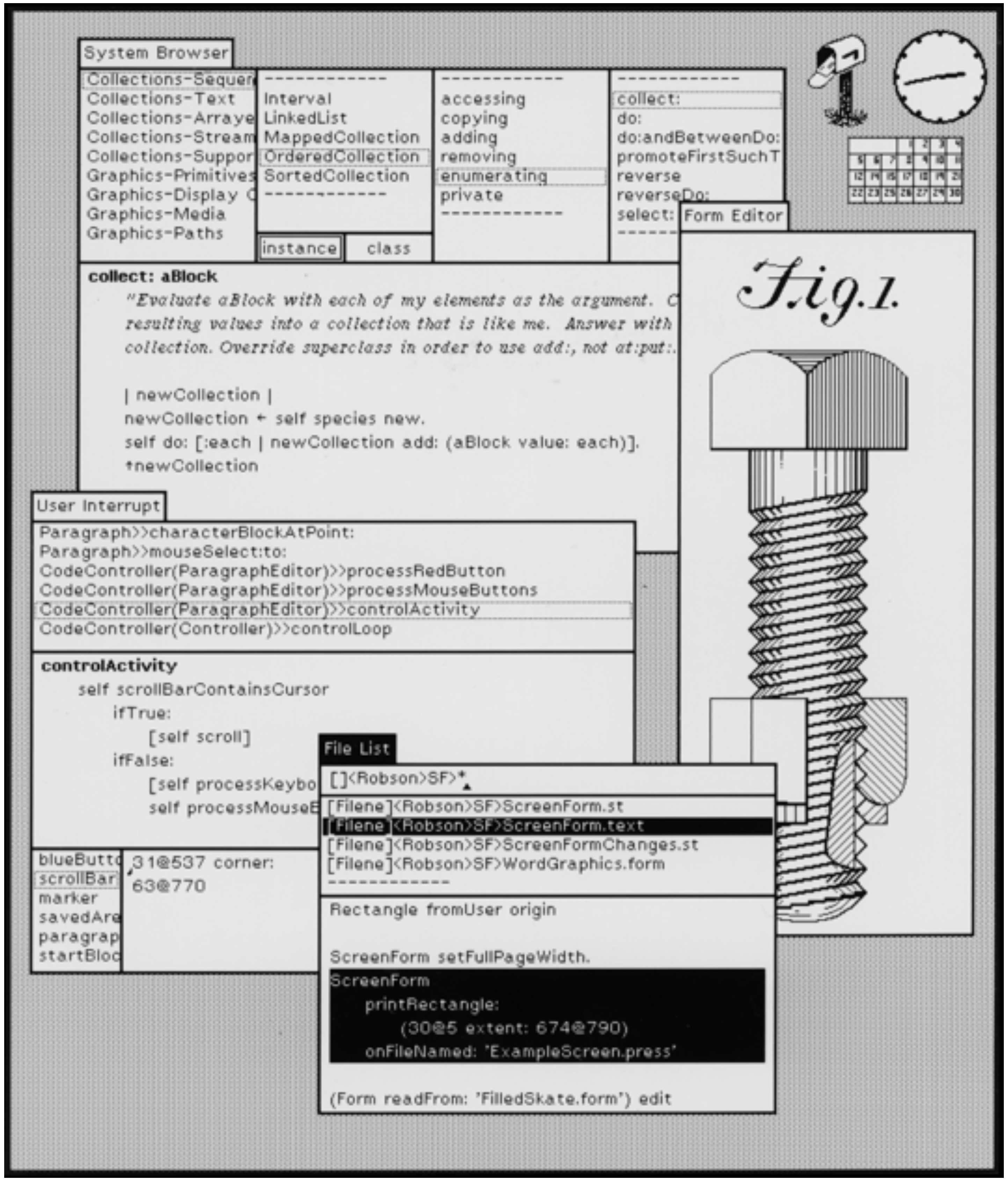
Immutability, Interactivity & JavaScript

Immutability, Interactivity & JavaScript (er ClojureScript)

cognitect



```
Collections-Sequen
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:
-----
```

```
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
```

Fig. 1.

```
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 processMouseB
```

```
blueButto
scrollBar
marker
savedAre
paragrap
startBloc
```

```
31@537 corner:
63@770
```

```
[]<Robson>SF>*
```

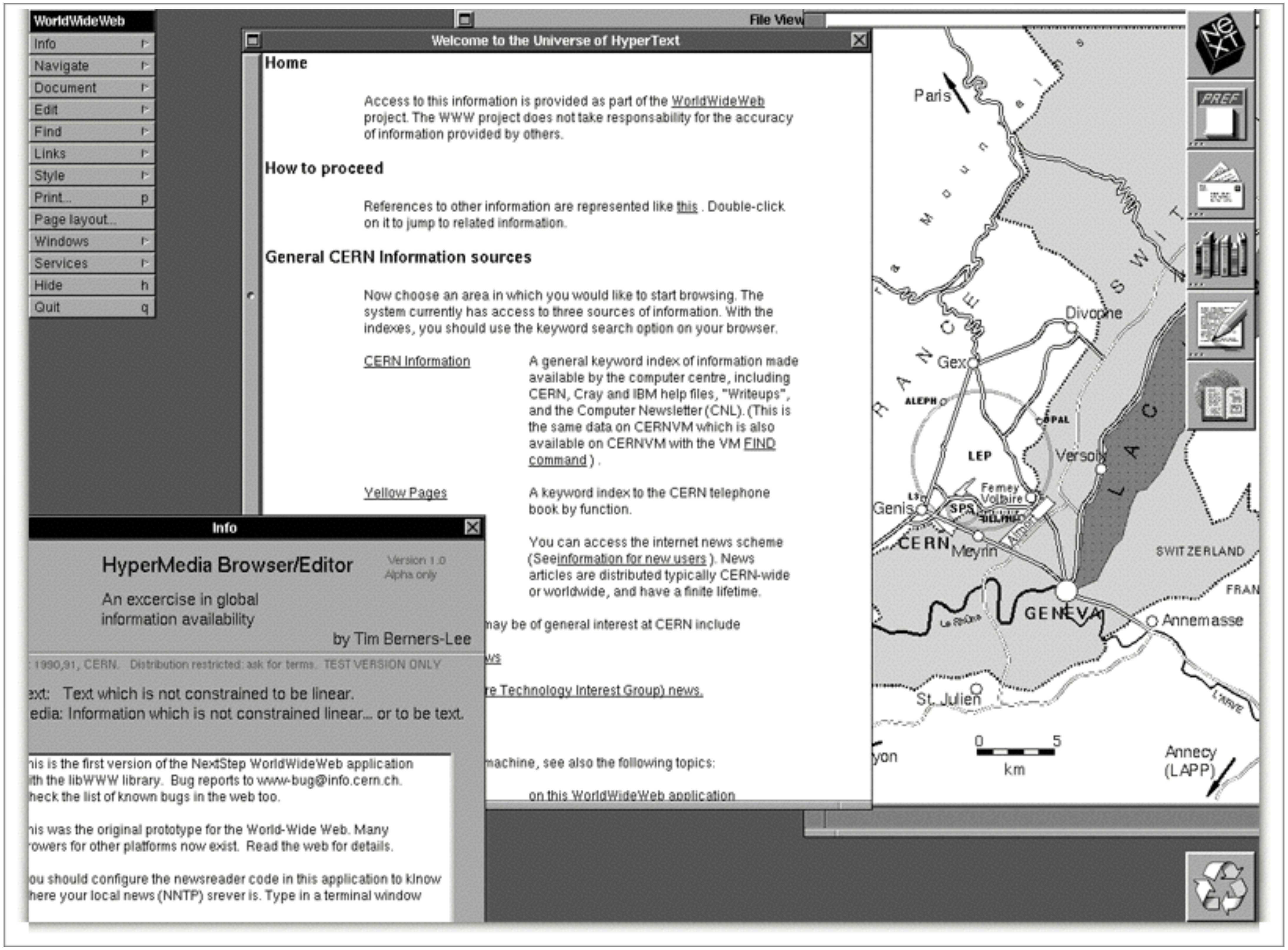
```
[Filene]<Robson>SF>ScreenForm.st
[Filene]<Robson>SF>ScreenForm.text
[Filene]<Robson>SF>ScreenFormChanges.st
[Filene]<Robson>SF>WordGraphics.form
------------
```

```
Rectangle fromUser origin

ScreenForm setFullPageWidth.
ScreenForm
	printRectangle:
		(30@5 extent: 674@790)
	onFileNamed: 'ExampleScreen.press'

(Form readFrom: 'FilledSkate.form') edit
```

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*



Welcome to the Universe of HyperText

Home

Access to this information is provided as part of the WorldWideWeb project. The WWW project does not take responsability for the accuracy of information provided by others.

How to proceed

References to other information are represented like this . Double-click on it to jump to related information.

General CERN Information sources

Now choose an area in which you would like to start browsing. The system currently has access to three sources of information. With the indexes, you should use the keyword search option on your browser.

CERN Information	A general keyword index of information made available by the computer centre, including CERN, Cray and IBM help files, "Writeups", and the Computer Newsletter (CNL). (This is the same data on CERNVM which is also available on CERNVM with the VM FIND command) .
Yellow Pages	A keyword index to the CERN telephone book by function.
	You can access the internet news scheme (Seeinformation for new users). News articles are distributed typically CERN-wide or worldwide, and have a finite lifetime.

may be of general interest at CERN include

ws

re Technology Interest Group) news.

machine, see also the following topics:

on this WorldWideWeb application

Info

HyperMedia Browser/Editor

Version 1.0
Alpha only

An excercise in global information availability

by Tim Berners-Lee

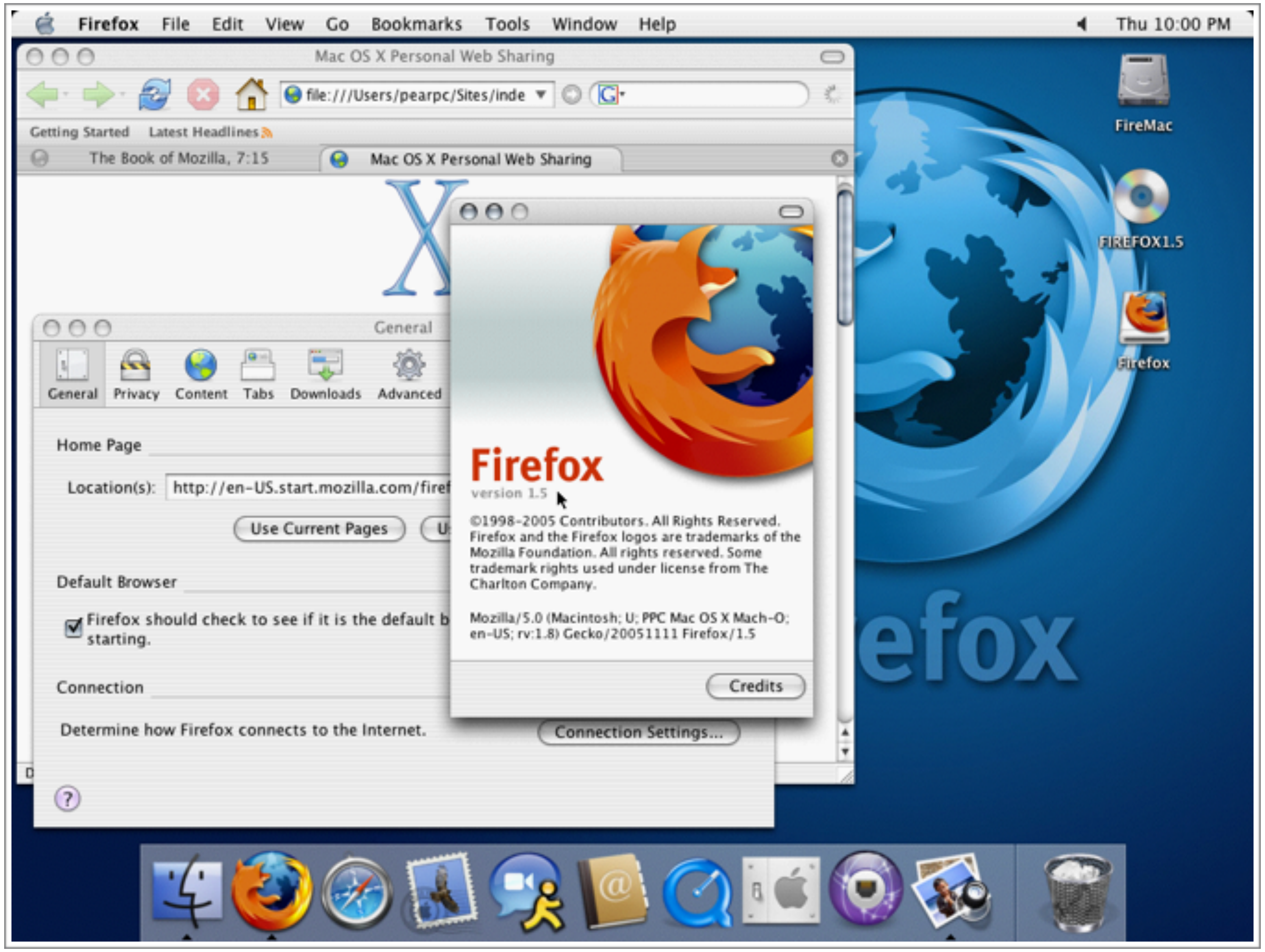
1990,91, CERN. Distribution restricted: ask for terms. TEST VERSION ONLY

ext: Text which is not constrained to be linear.
edia: Information which is not constrained linear... or to be text.

his is the first version of the NextStep WorldWideWeb application ith the libWWW library. Bug reports to www-bug@info.cern.ch. heck the list of known bugs in the web too.

his was the original prototype for the World-Wide Web. Many rowers for other platforms now exist. Read the web for details.

ou should configure the newsreader code in this application to know here your local news (NNTP) srever is. Type in a terminal window


Firefox File Edit View Go Bookmarks Tools Window Help
Thu 10:00 PM
Mac OS X Personal Web Sharing
file:///Users/pearpc/Sites/inde
Getting Started
Latest Headlines
The Book of Mozilla, 7:15
Mac OS X Personal Web Sharing
General
General Privacy Content Tabs Downloads Advanced
Home Page
Default Browser
Connection
Use Current Pages
Connection Settings...
FireMac
FIREFOX1.5
Firefox
Credits

Location(s): http://en-US.start.mozilla.com/firef

Firefox should check to see if it is the default b starting.

Determine how Firefox connects to the Internet.

Firefox
version 1.5

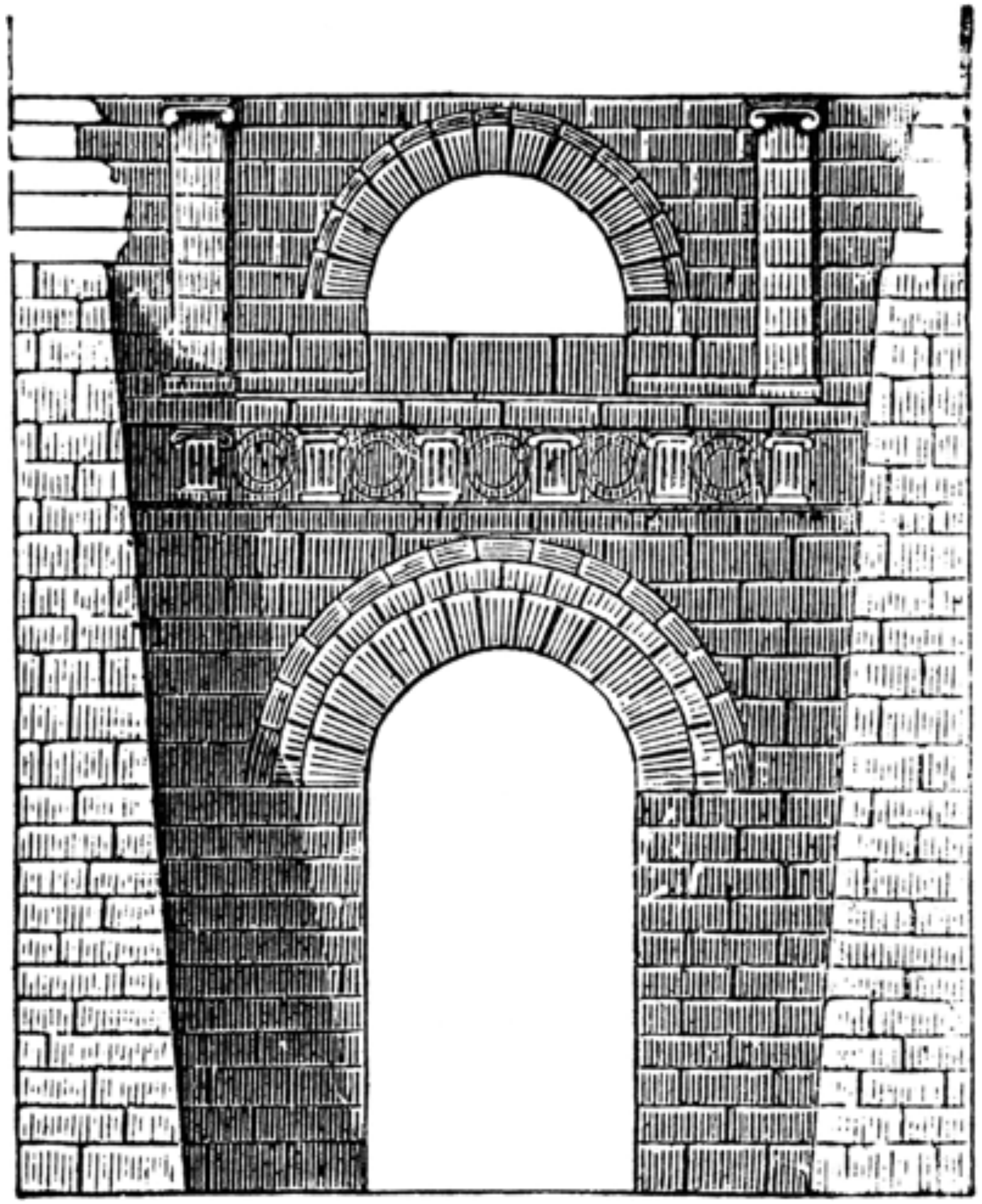
©1998-2005 Contributors. All Rights Reserved. Firefox and the Firefox logos are trademarks of the Mozilla Foundation. All rights reserved. Some trademark rights used under license from The Charlton Company.

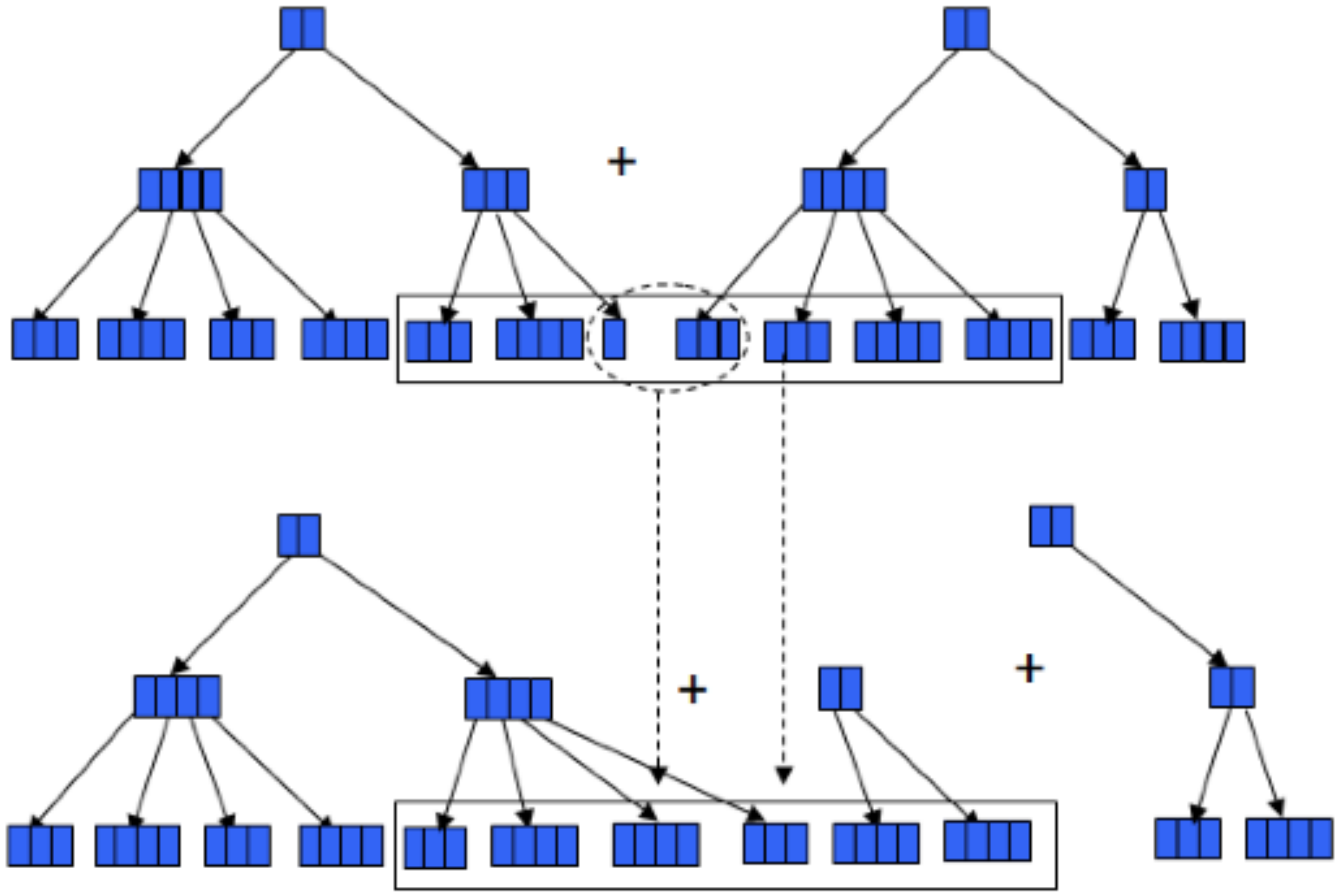
Mozilla/5.0 (Macintosh; U; PPC Mac OS X Mach-O; en-US; rv:1.8) Gecko/20051111 Firefox/1.5

Mutable DOM

Functional Programming?

- Functional Reactive Programming (FRP), still active area of research
    - Rx, doesn't address rendering
- Communicating Sequential Processes (CSP), a coordination language, doesn't address rendering



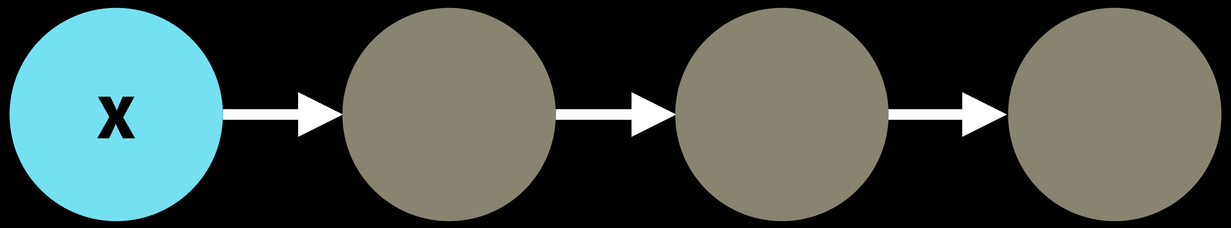


HACKER
SCHOOL
2012

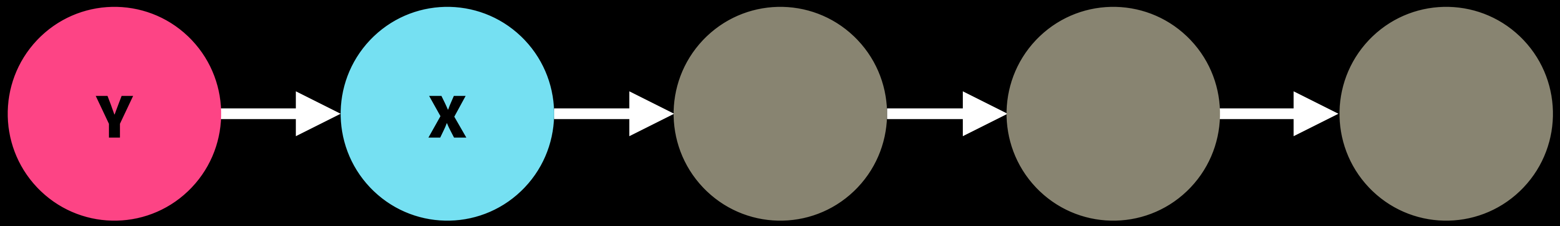
Functional Programming and Data

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

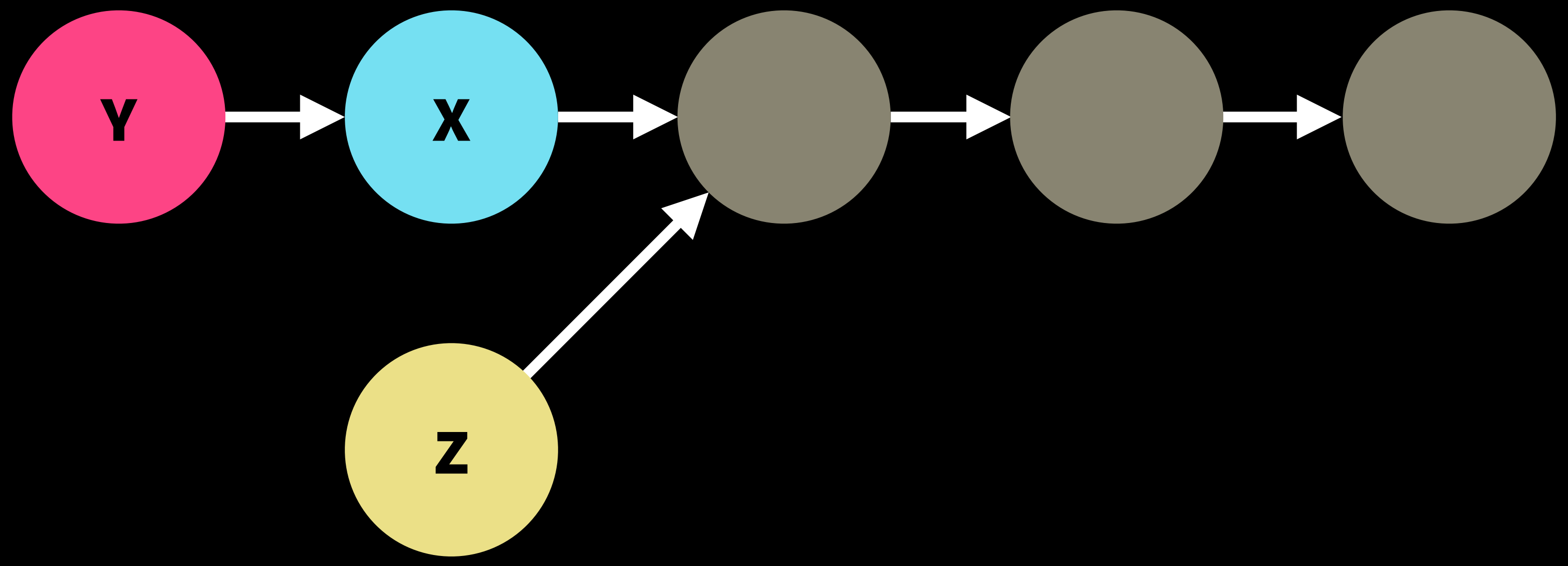
Simple example: Linked List



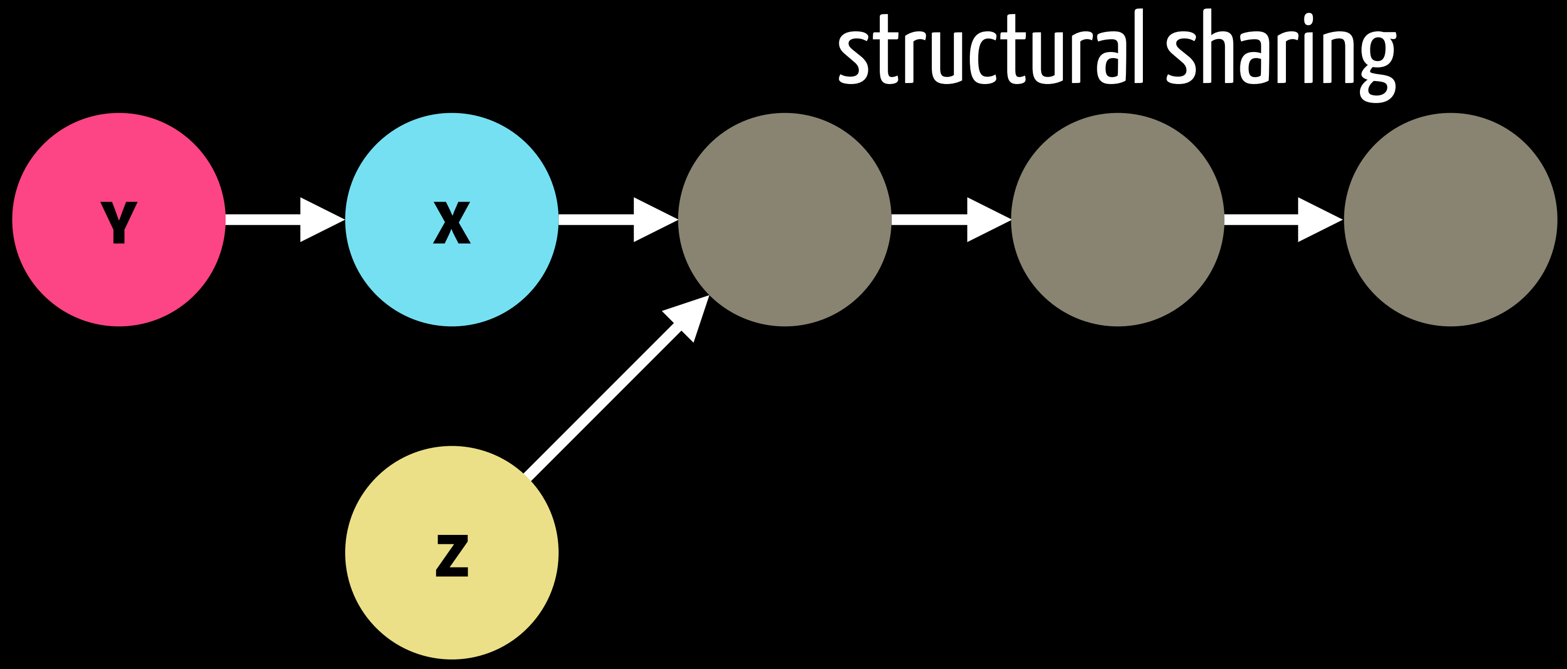
Simple example: Linked List



Simple example: Linked List



Simple example: Linked List



Sharing structure

- space efficiency
- computational efficiency – avoids copying

Phil Bagwell

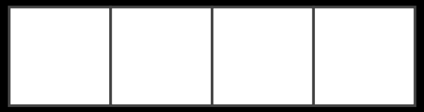
- Array Mapped Trie
- Hash Array Mapped Trie

Bitmapped Vector Trie

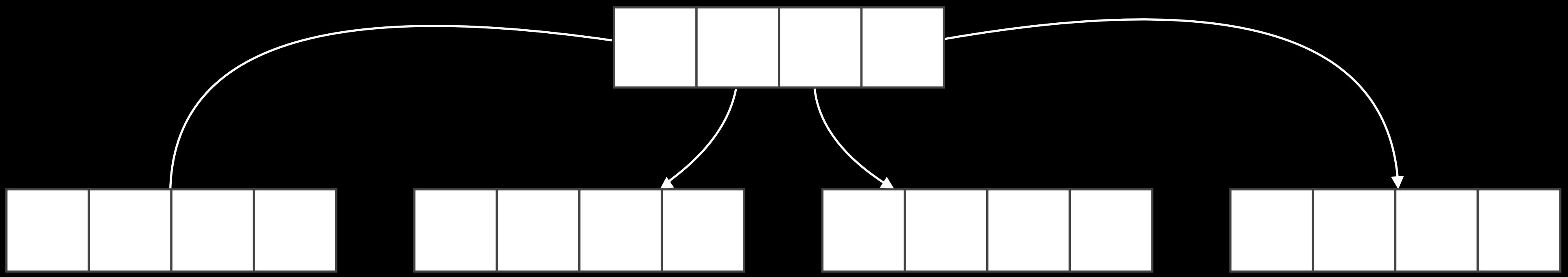
- data lives in the leaves
- e.g. prefix tree used for string lookup
- bitwise trie

Persistent Vector

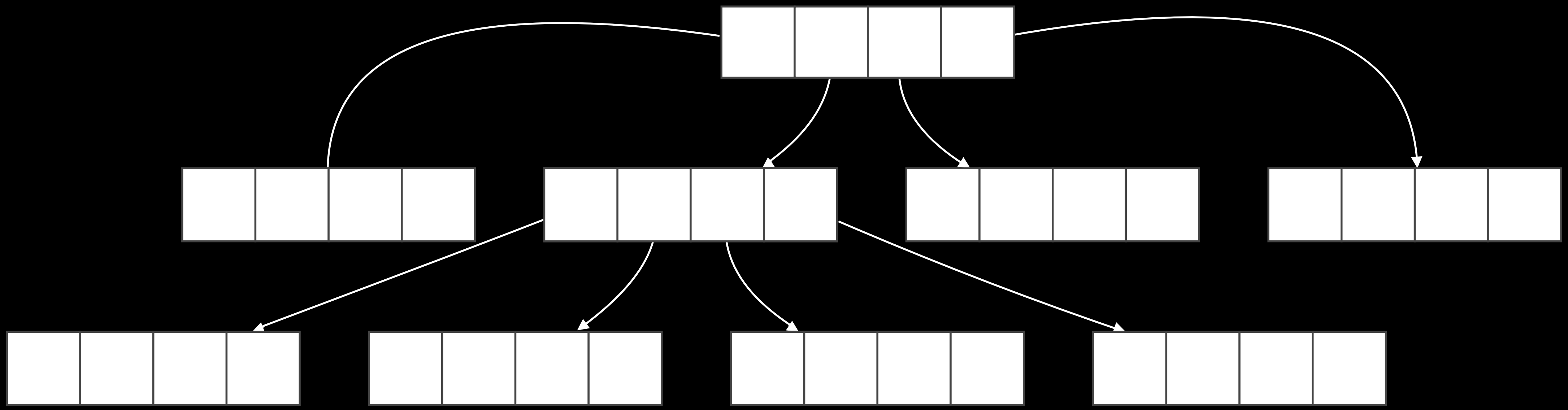
Persistent Vector



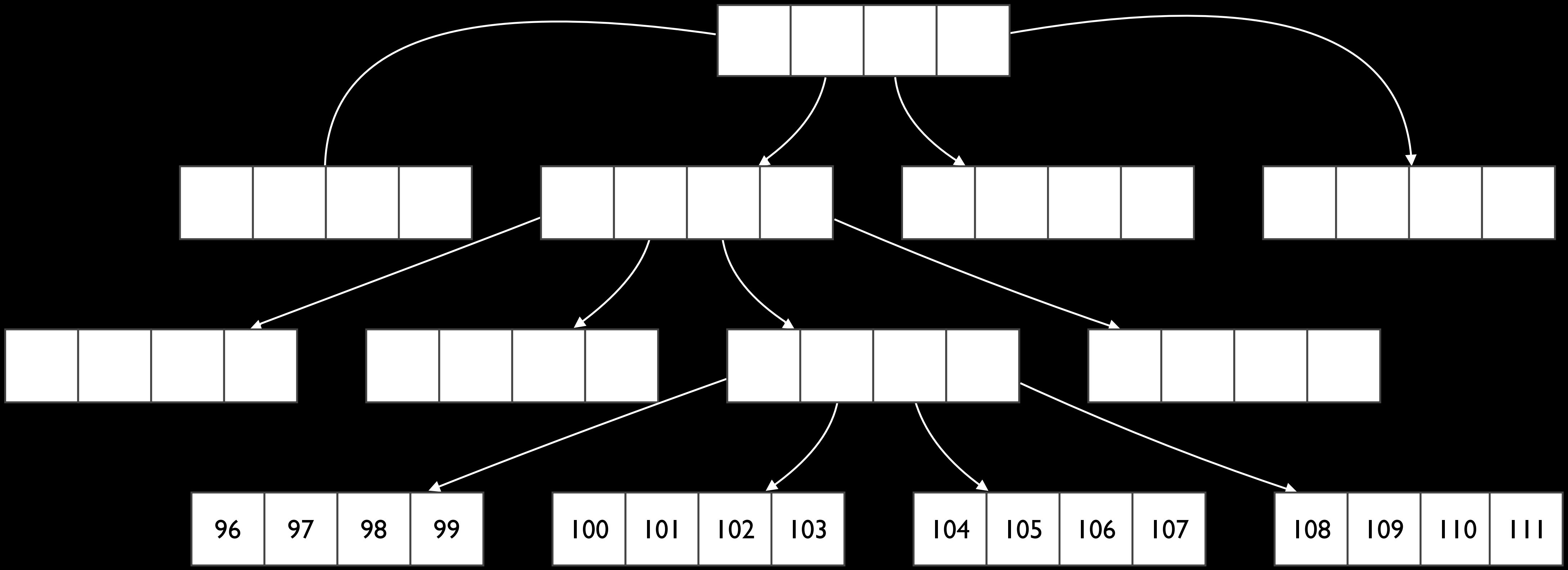
Persistent Vector



Persistent Vector



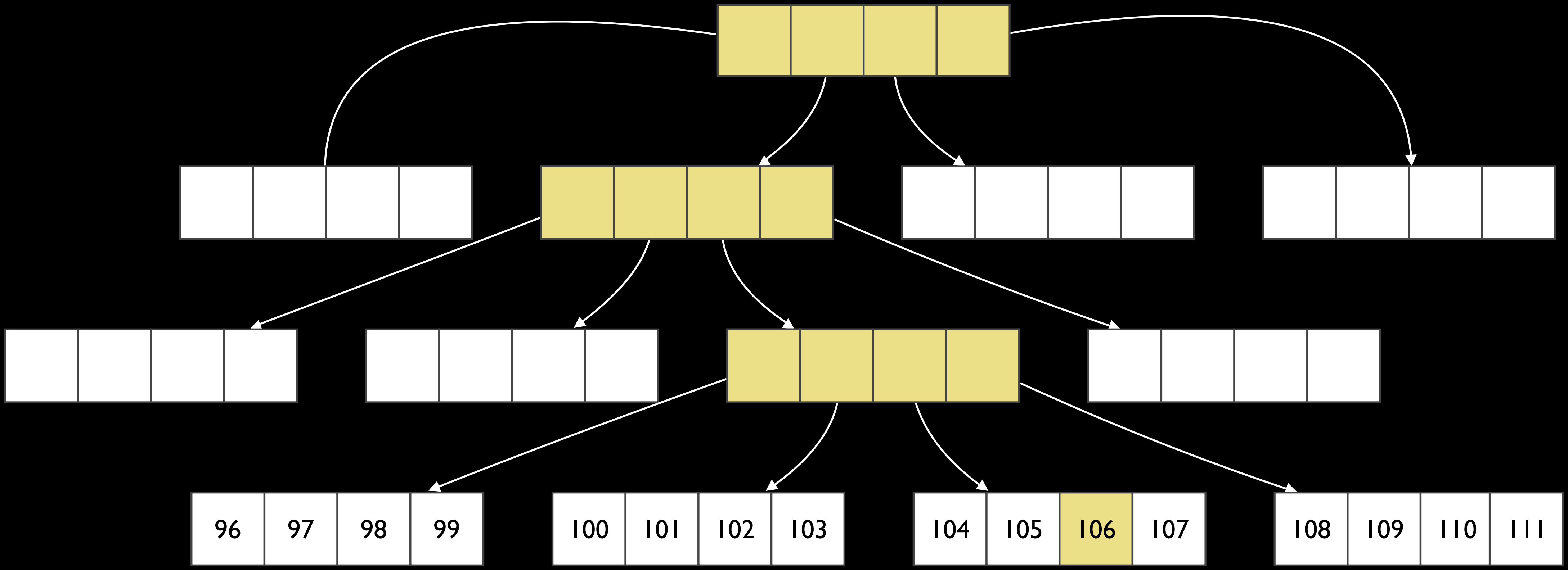
Persistent Vector



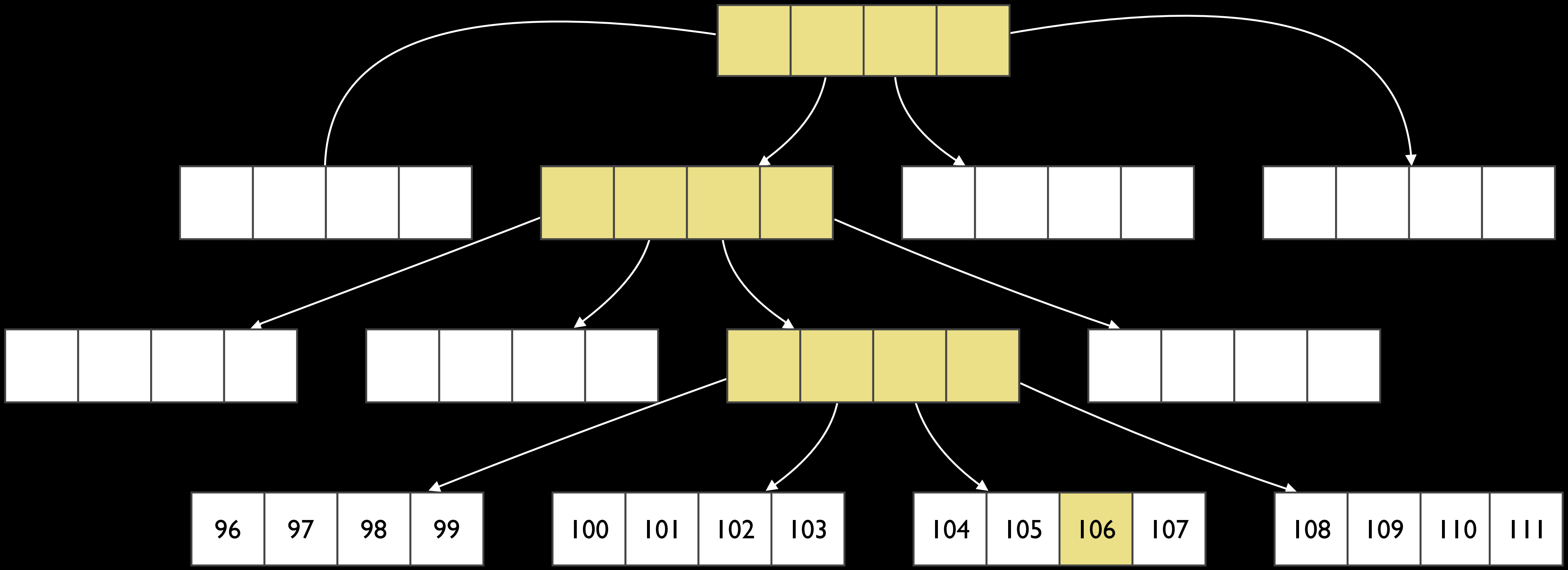
Persistent Vector

`getindex`

Persistent Vector

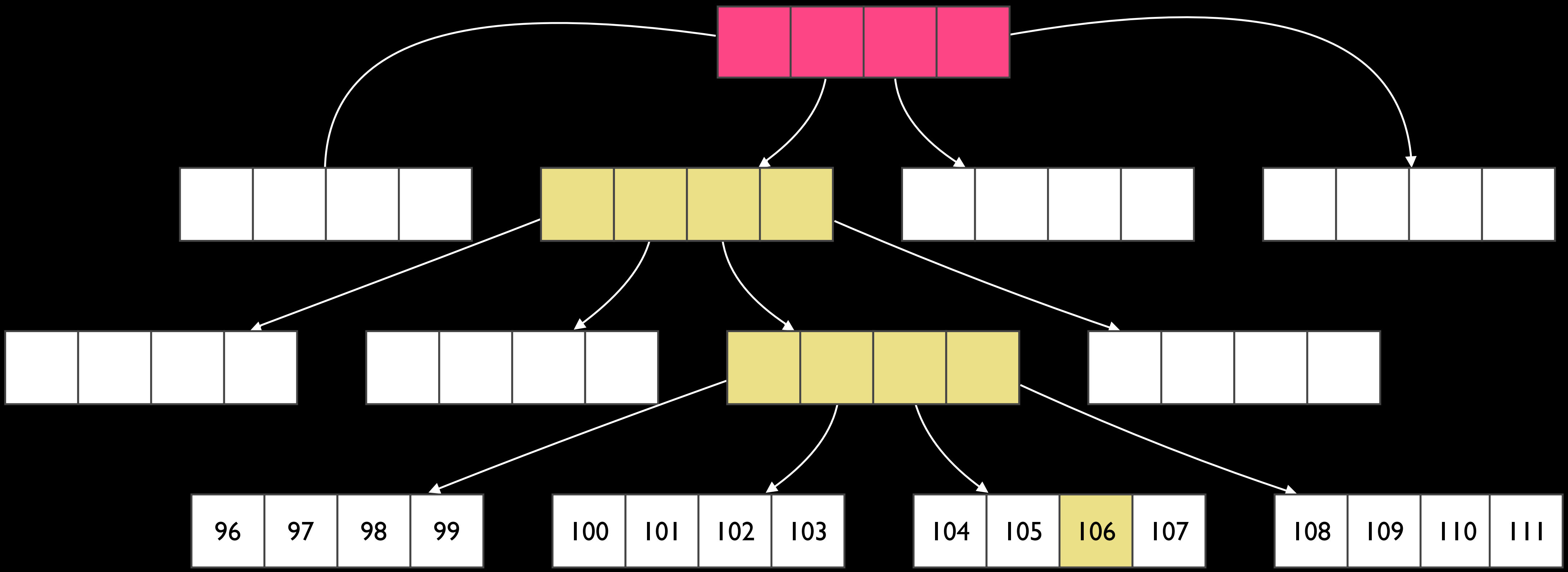


Persistent Vector



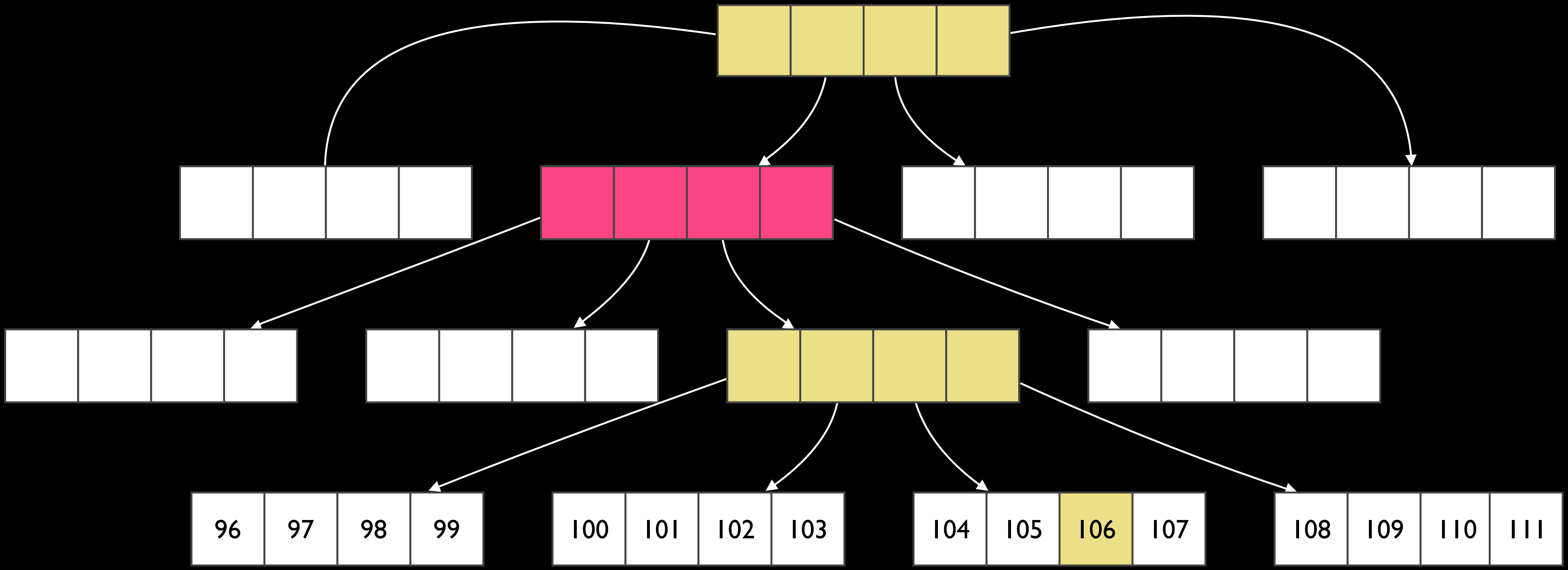
0b01101010

Persistent Vector



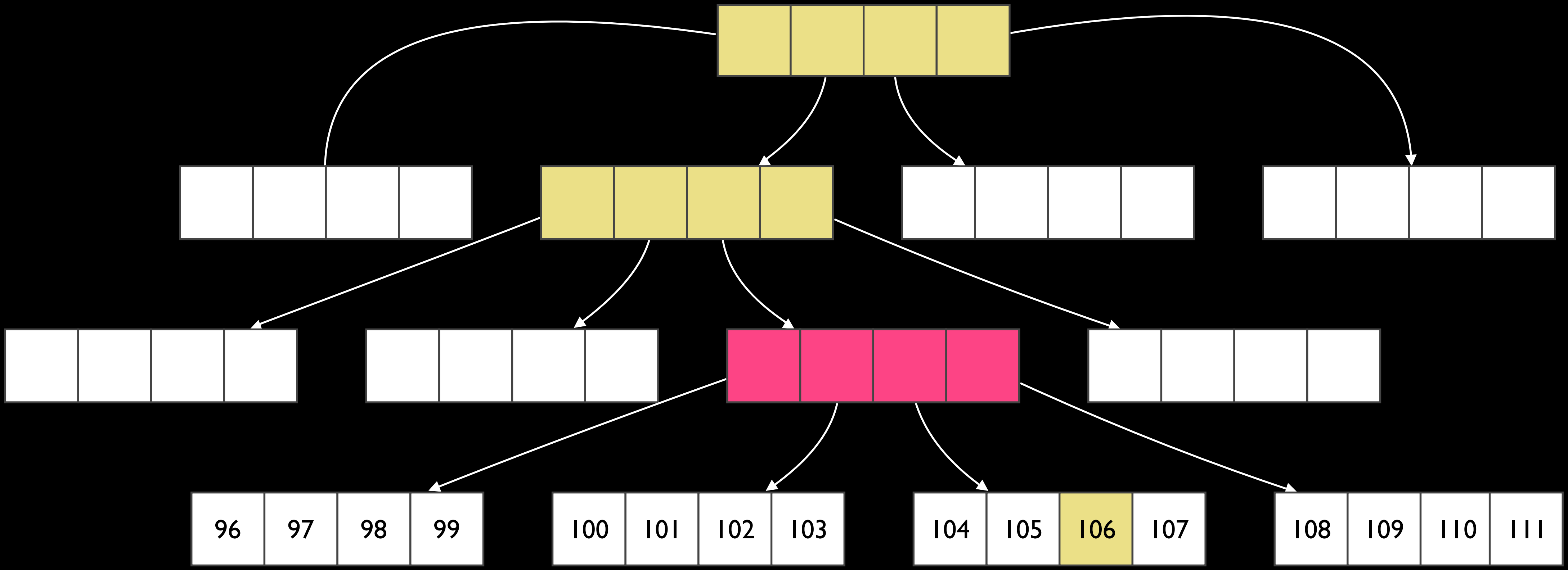
0b01101010

Persistent Vector



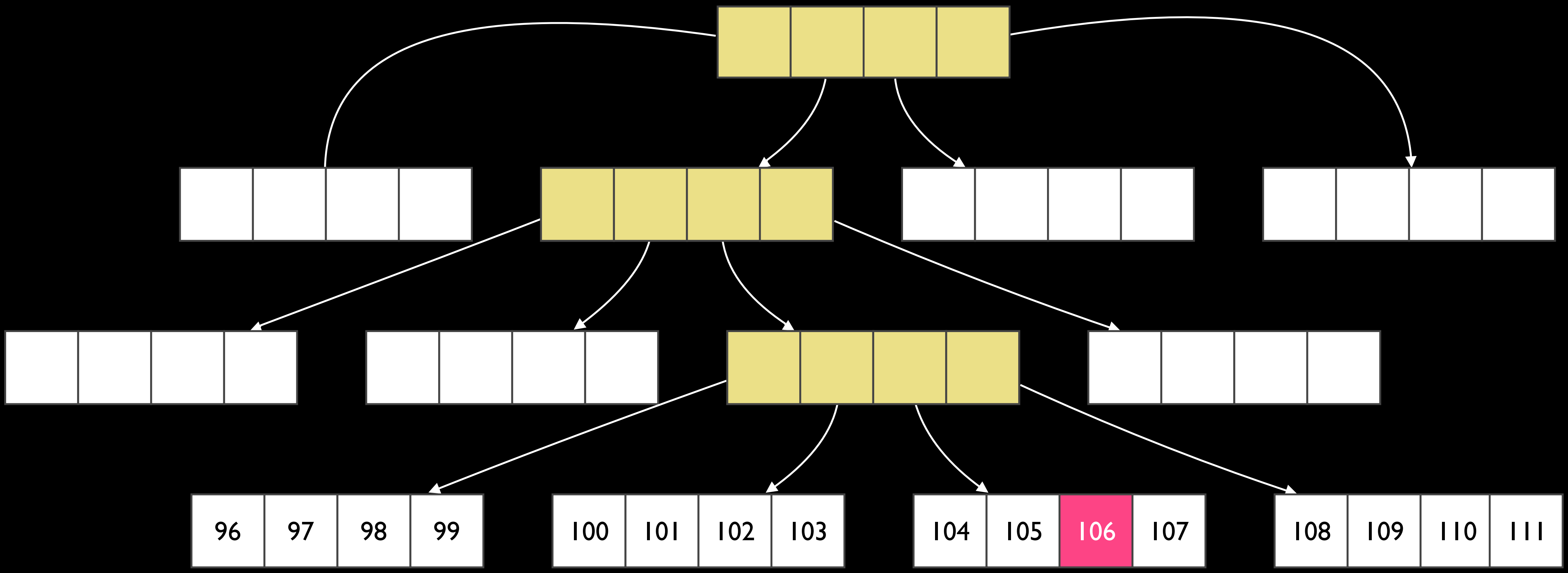
0b01101010

Persistent Vector



0b01101010

Persistent Vector

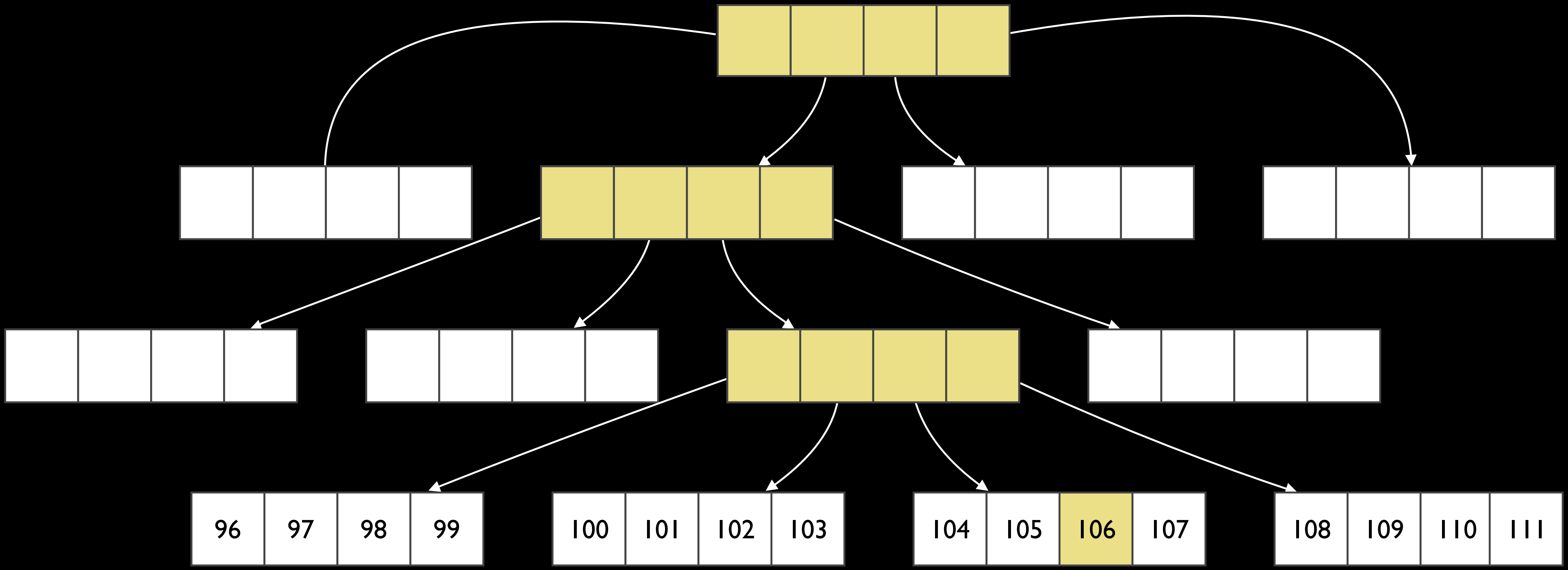


0b01101010

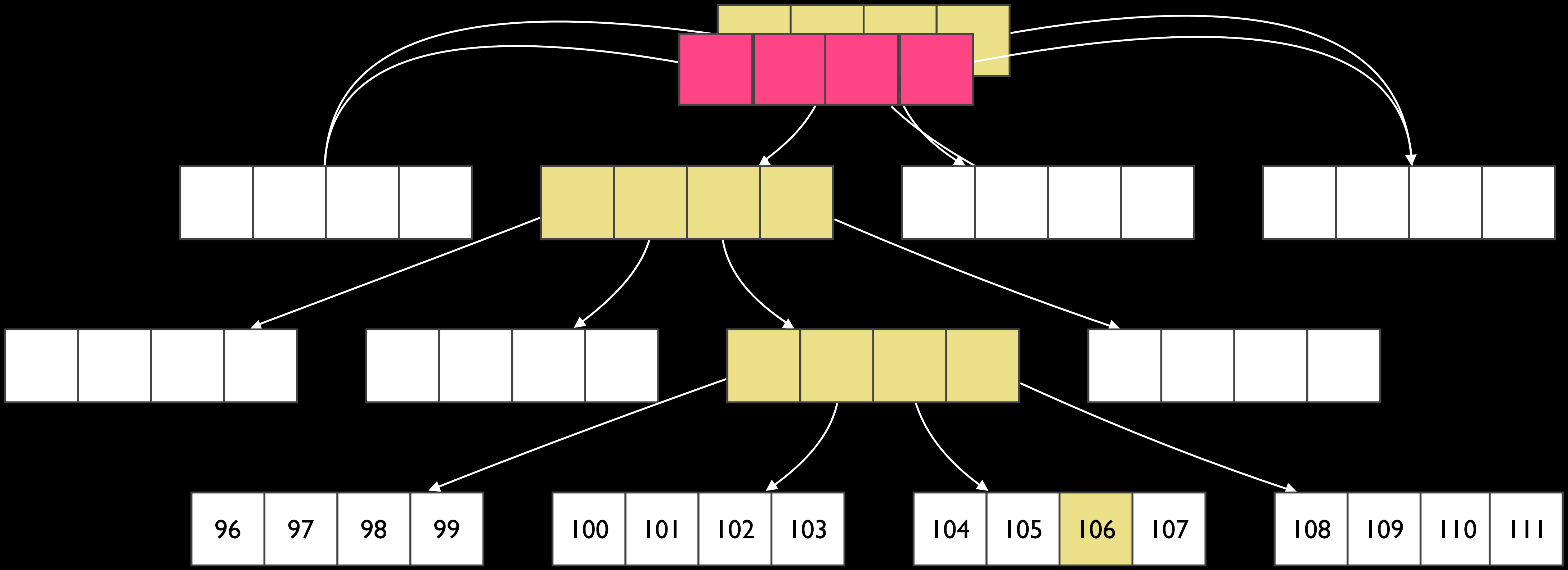
Persistent Vector

assoc

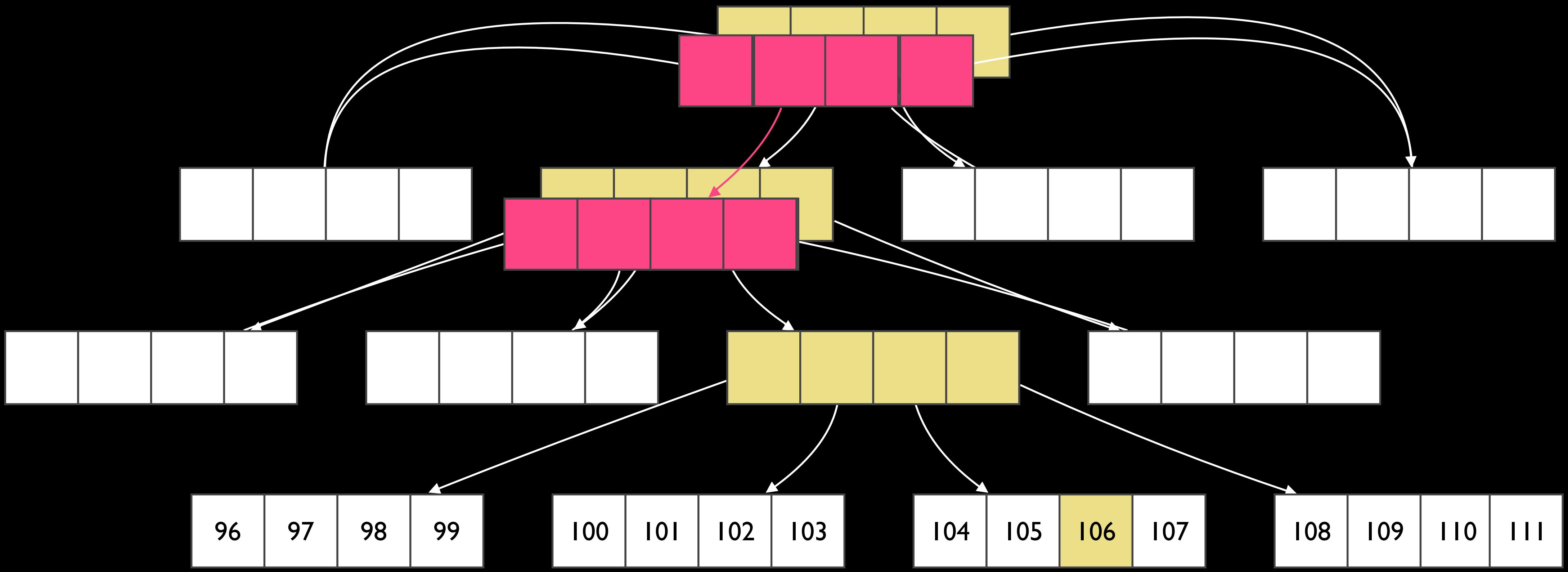
Persistent Vector



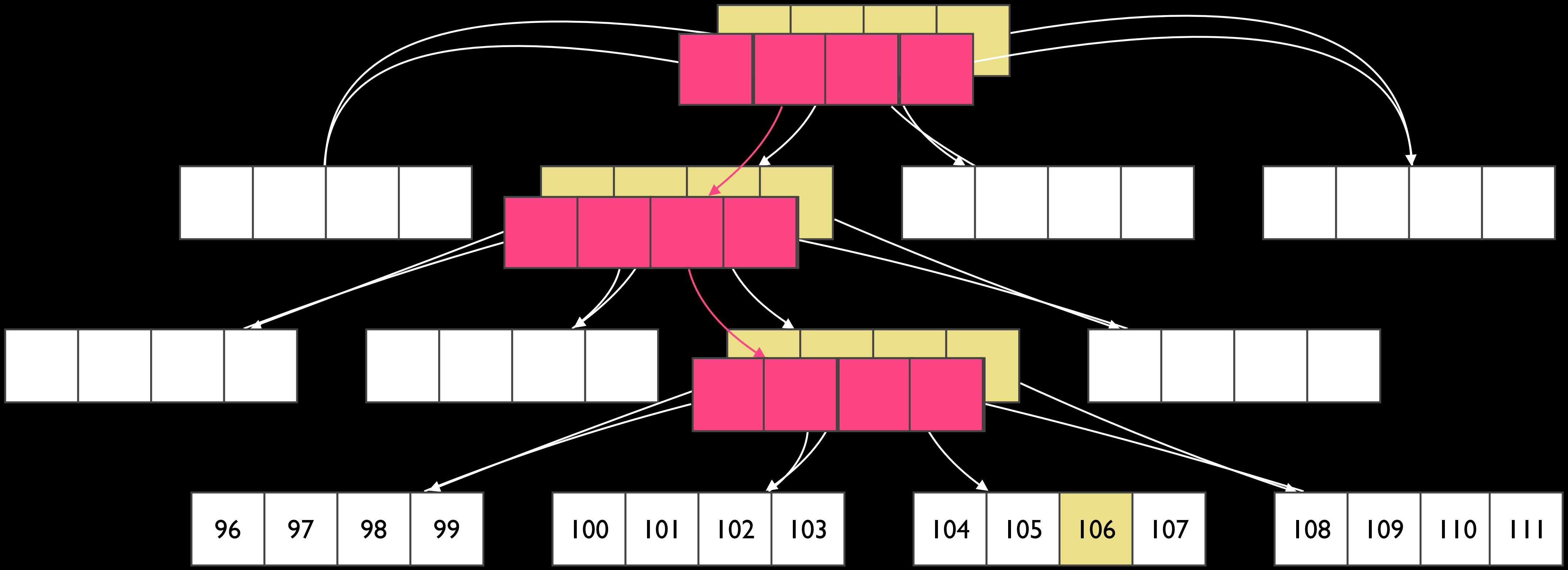
Persistent Vector



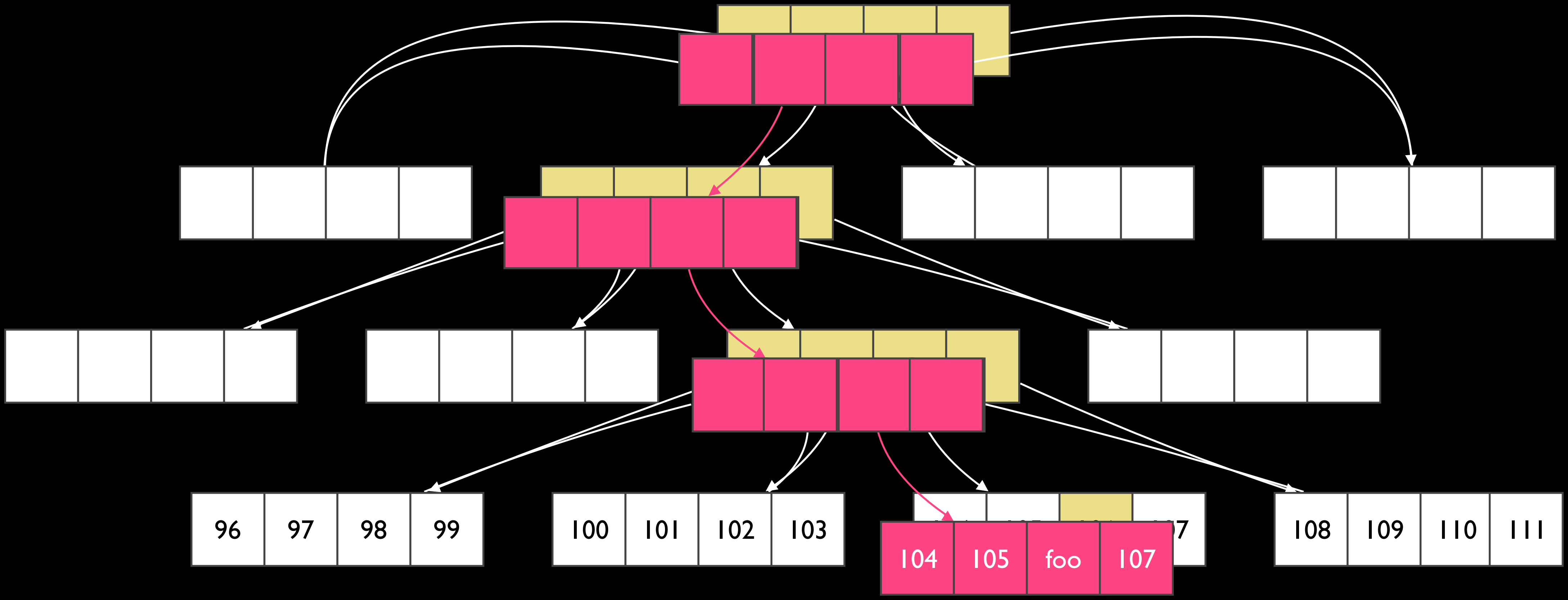
Persistent Vector



Persistent Vector



Persistent Vector

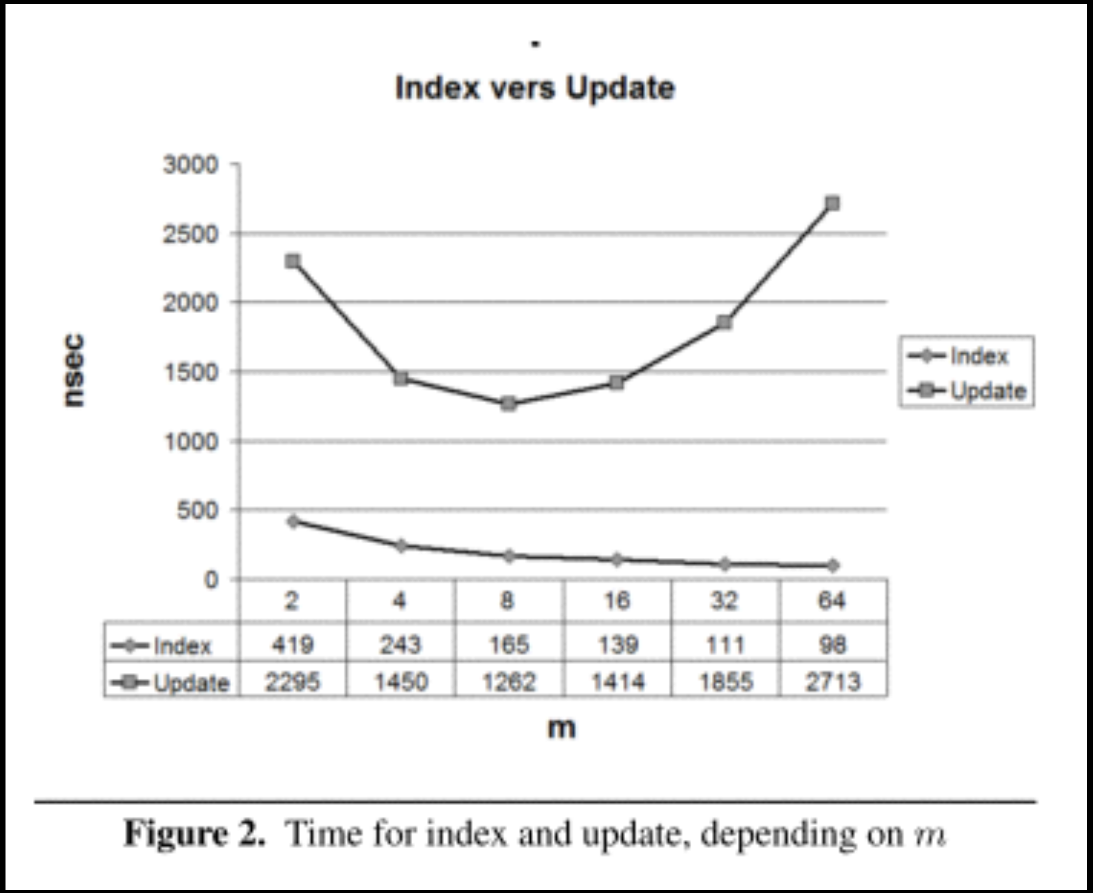


Persistent Vector

Length 4 internal vectors?

Persistent Vector

32



	2	4	8	16	32	64
Index	419	243	165	139	111	98
Update	2295	1450	1262	1414	1855	2713

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

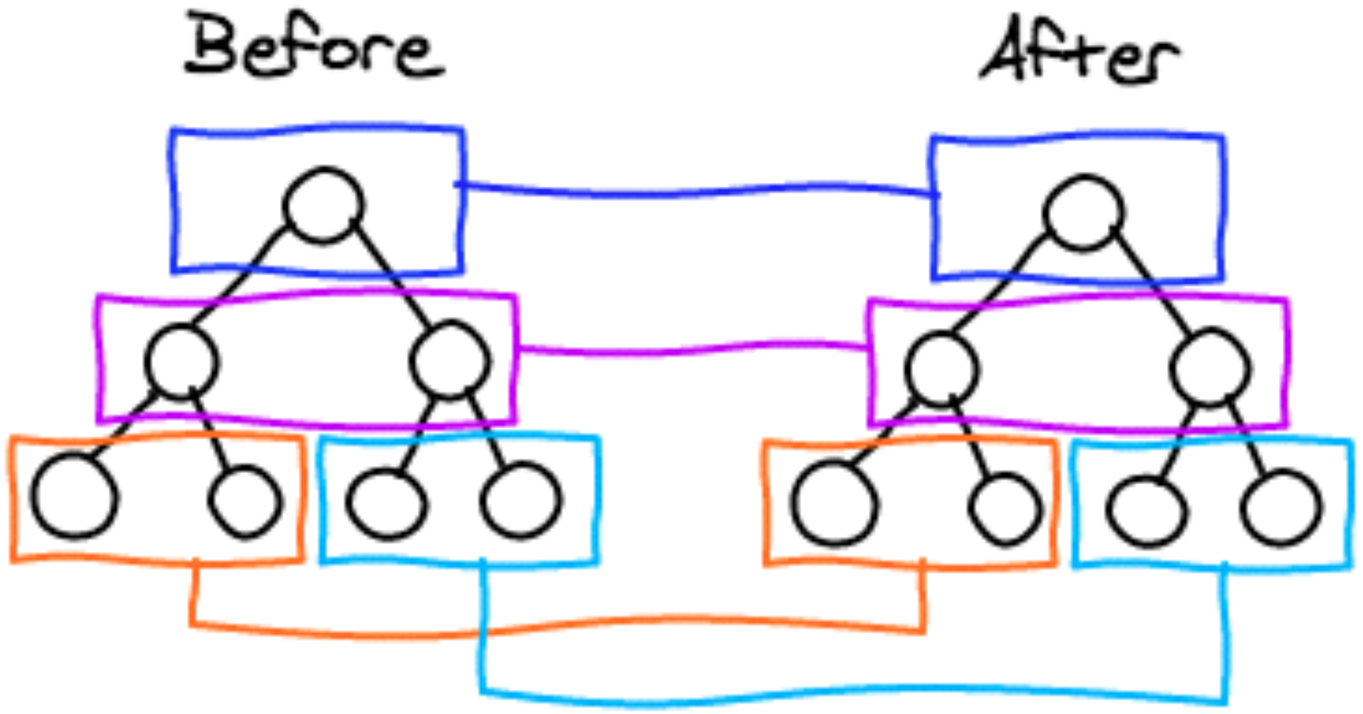
React

A JAVASCRIPT LIBRARY FOR BUILDING USER INTERFACES

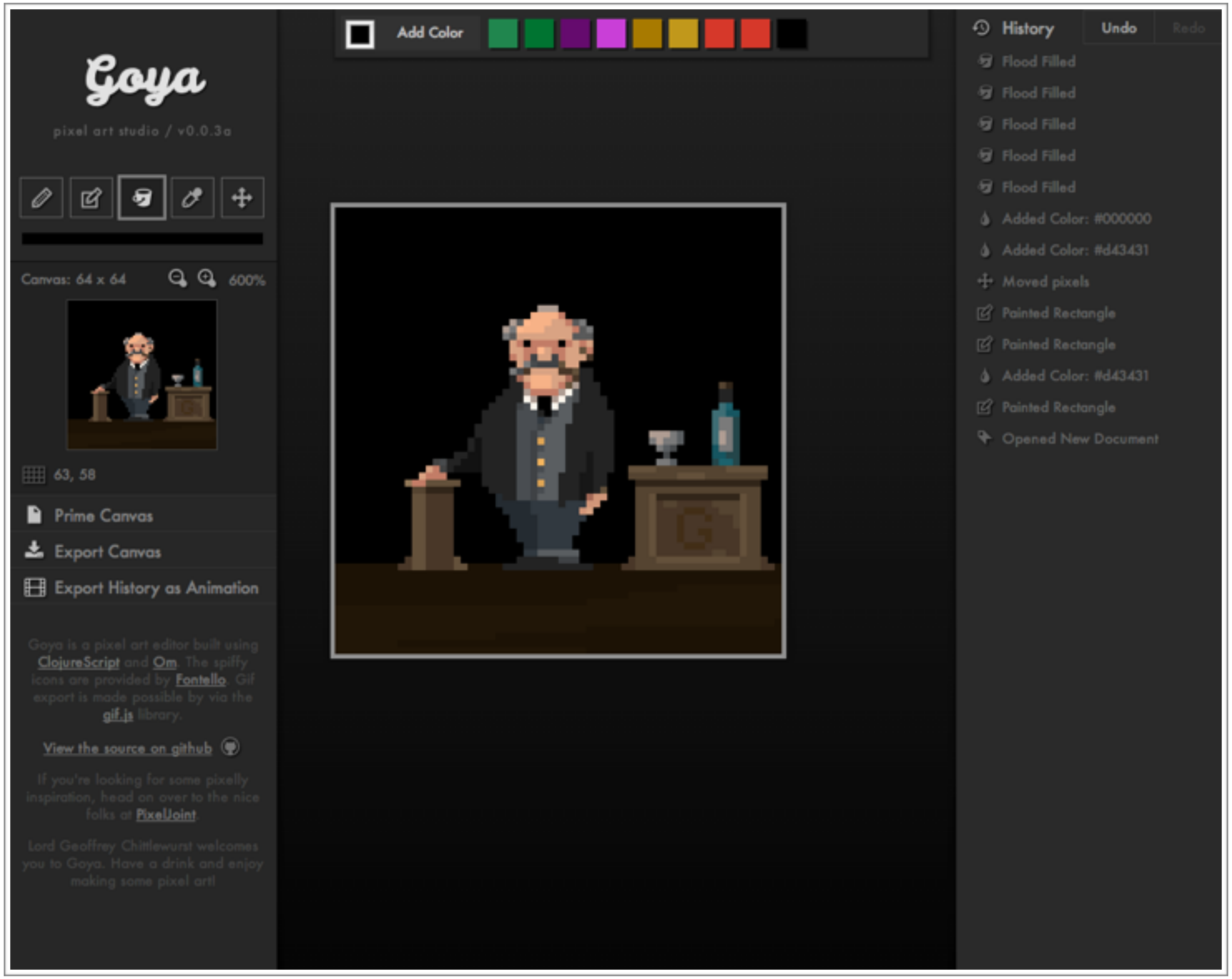
$$f(D_0) = V_0$$

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

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


Before
After

demo


Goya
pixel art studio / v0.0.3a
Add Color
Canvas: 64 x 64
600%
63, 58
Prime Canvas
Export Canvas
Export History as Animation
View the source on github
History
Undo
Redo

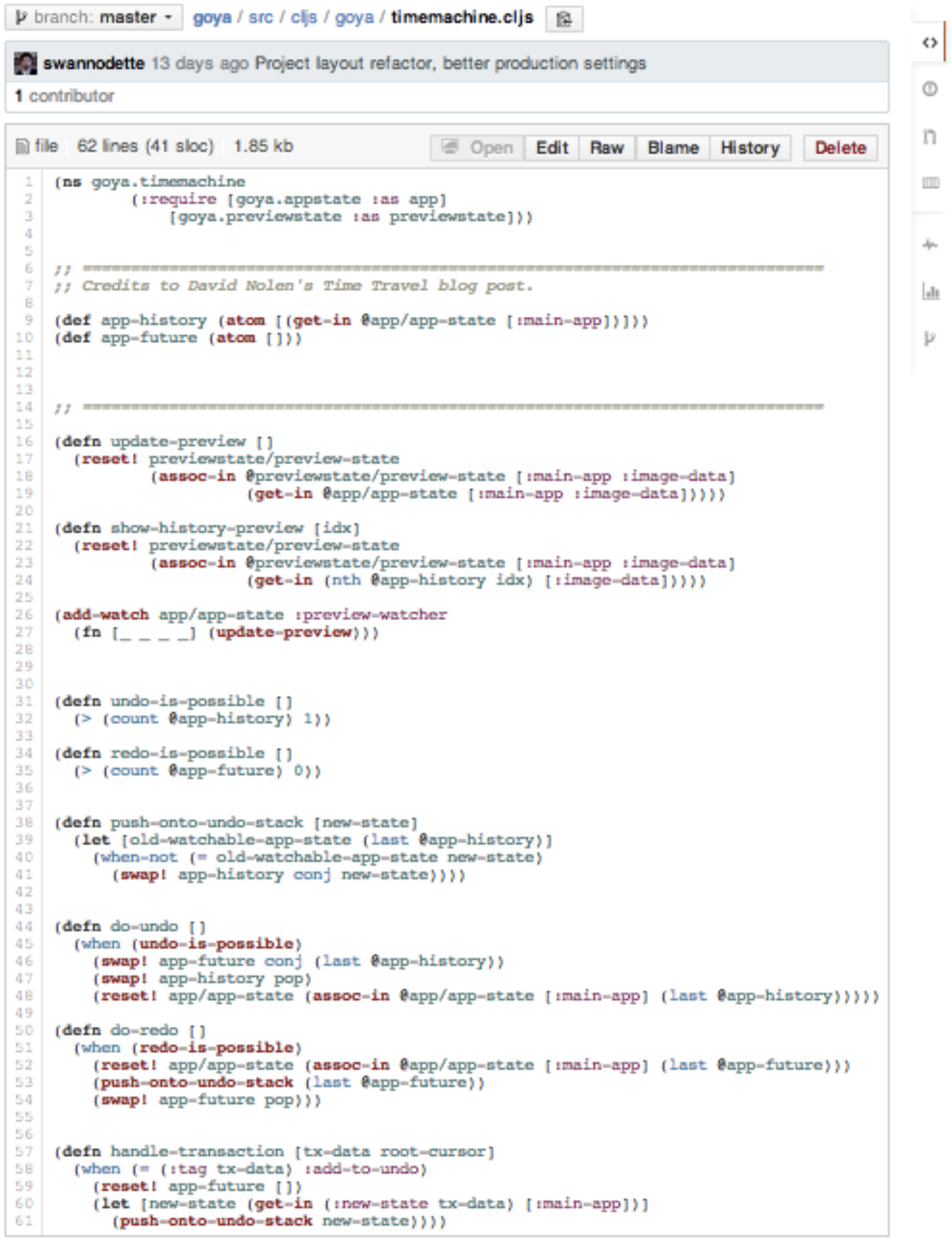
Goya is a pixel art editor built using ClojureScript and Om. The spiffy icons are provided by Fontello. Gif export is made possible by via the gif.js library.

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!

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

demo

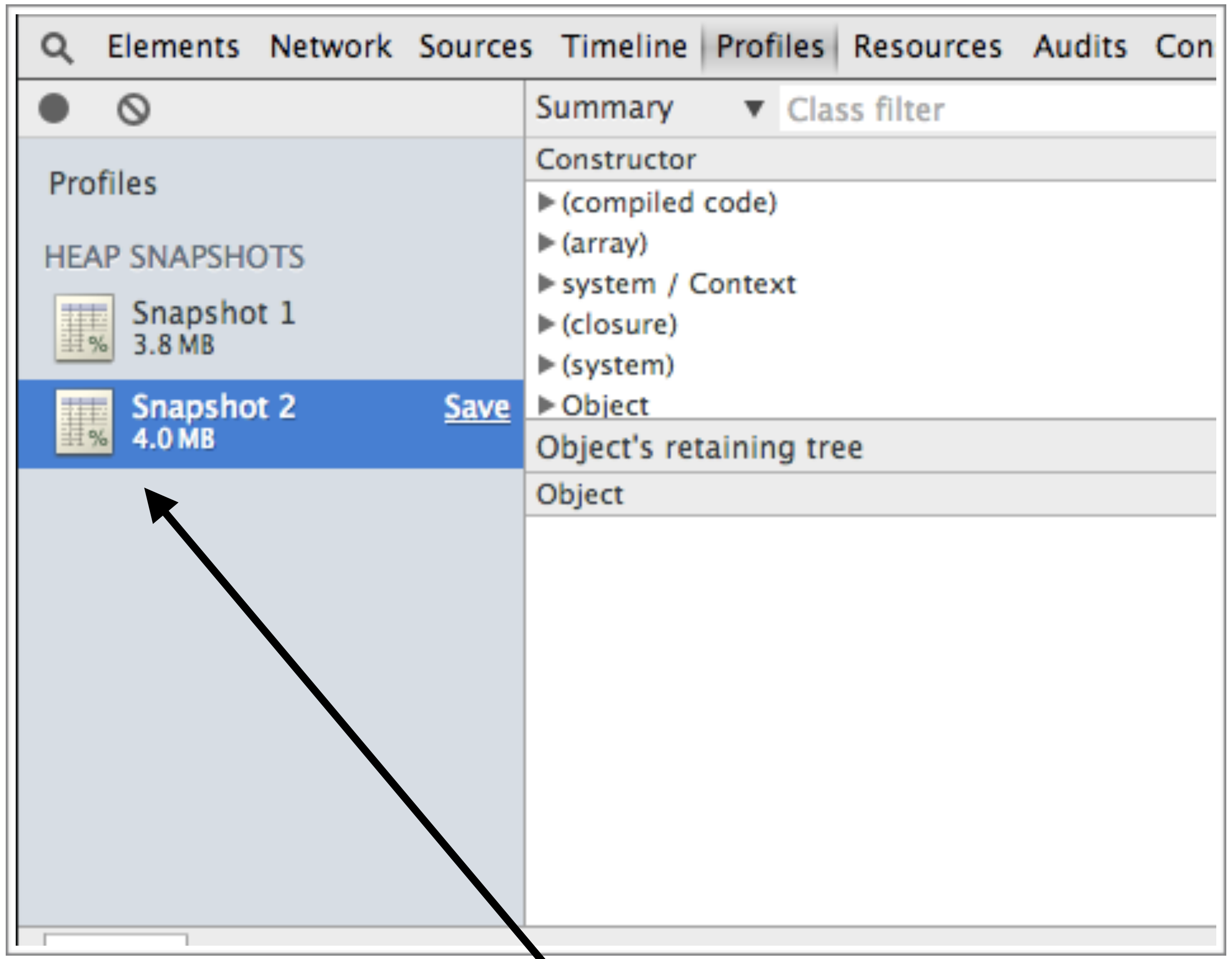

branch: master goya / src / cljs / goya / timemachine.cljs

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

```clojure
(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))))
```



Elements Network Sources Timeline Profiles Resources Audits Con

Summary ▼ Class filter

Profiles

HEAP SNAPSHOTS

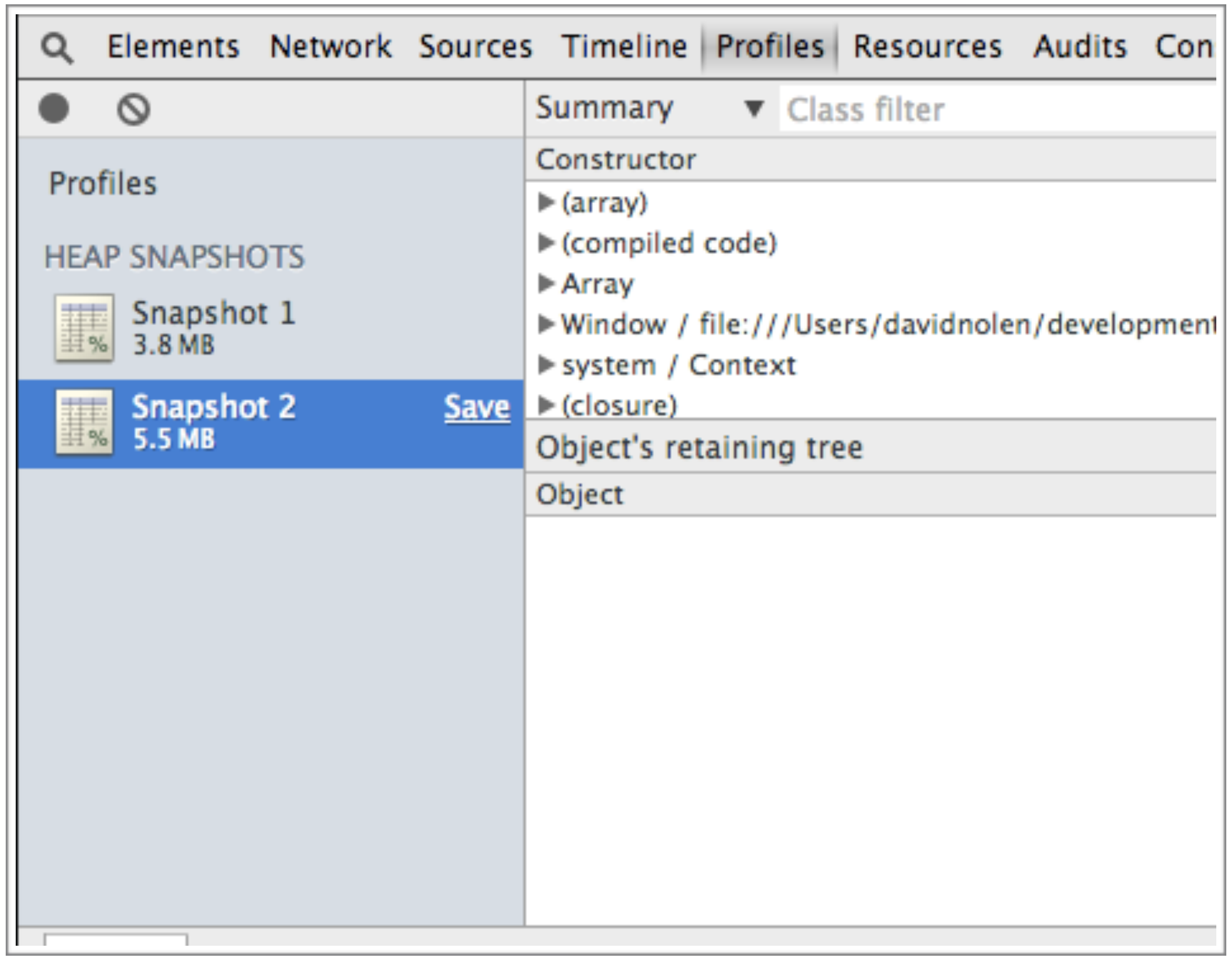
Snapshot 1
3.8 MB

Snapshot 2 Save
4.0 MB

Constructor
▶ (compiled code)
▶ (array)
▶ system / Context
▶ (closure)
▶ (system)
▶ Object

Object's retaining tree

Object



Elements Network Sources Timeline Profiles Resources Audits Con

Summary ▼ Class filter

Profiles

HEAP SNAPSHOTS

Snapshot 1
3.8 MB

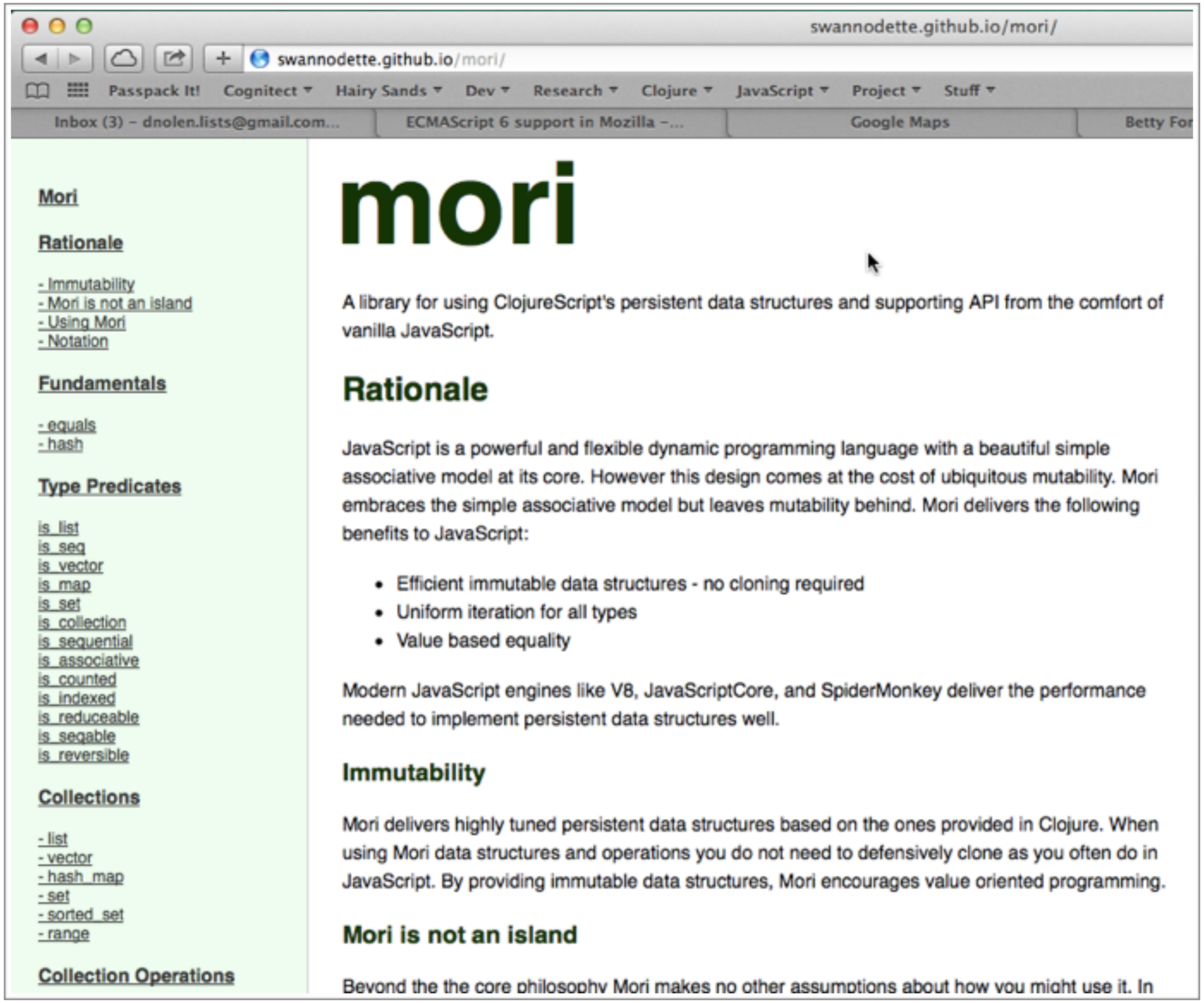
Snapshot 2 Save
5.5 MB

Constructor
▶ (array)
▶ (compiled code)
▶ Array
▶ Window / file:///Users/davidnolen/developmen
▶ system / Context
▶ (closure)

Object's retaining tree

Object

Persistent Data Structures ... ROCK



swannodette.github.io/mori/
Passpack It!
Cognitect
Hairy Sands
Dev
Research
Clojure
JavaScript
Project
Stuff
Inbox (3) – dnolen.lists@gmail.com...
ECMAScript 6 support in Mozilla –...
Google Maps
Betty For

mori

A library for using ClojureScript's persistent data structures and supporting API from the comfort of vanilla JavaScript.

Rationale

JavaScript is a powerful and flexible dynamic programming language with a beautiful simple associative model at its core. However this design comes at the cost of ubiquitous mutability. Mori embraces the simple associative model but leaves mutability behind. Mori delivers the following benefits to JavaScript:

- Efficient immutable data structures - no cloning required
- Uniform iteration for all types
- Value based equality

Modern JavaScript engines like V8, JavaScriptCore, and SpiderMonkey deliver the performance needed to implement persistent data structures well.

Immutability

Mori delivers highly tuned persistent data structures based on the ones provided in Clojure. When using Mori data structures and operations you do not need to defensively clone as you often do in JavaScript. By providing immutable data structures, Mori encourages value oriented programming.

Mori is not an island

Beyond the the core philosophy Mori makes no other assumptions about how you might use it. In

Questions?