

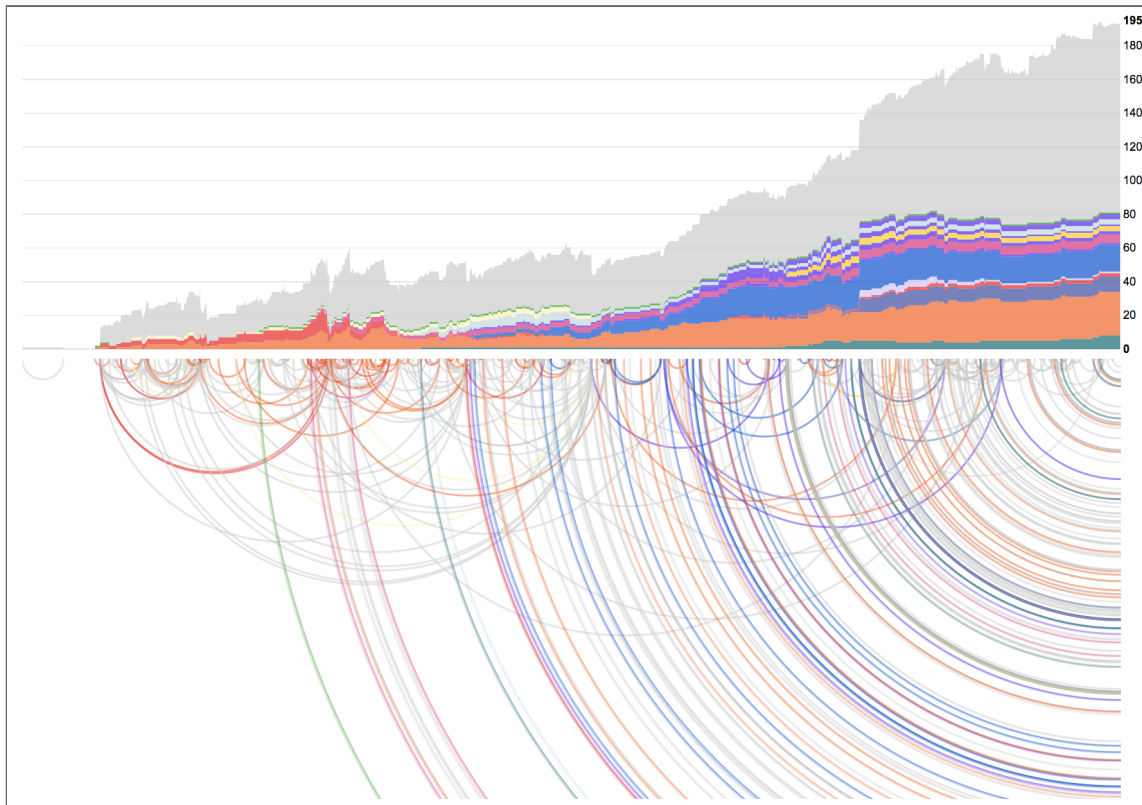
BUMBLEBEE UPDATE

ADSUG 2018

TIM HOSTETLER

SOME STATS FROM LAST YEAR

- 30 Releases
- 517 Commits
- 110 Issues Closed



PARITY WITH CLASSIC

- Co-author affiliation export tool
- Custom export formats
- Personalized user settings
- Classic search translator

OTHER IMPROVEMENTS

- Overall speed increases
 - Reduce superfluous requests
 - Better caching/storage usage
 - Lazy loading highlights
- Better user feedback
 - Loading bar
 - Error messages
- Change in search style
- Better ORCID integration
- Pagination updates

IN THE WORKS

- myADS interface and sign-up screens (in planning)
- Some advanced ADS library features (in progress)
- Save/recall search history
- Export of citation metrics
- “Show similar articles” (in testing?)

CHALLENGES

- Aging tech stack
- Difficult to maintain
- Only so fast, without major refactor
- Lack of front-end devs/specialists

CONSIDERATIONS


- Refactor/Update codebase
- Move to newer front-end frameworks/technologies
- How to transition properly
- Balance priorities

WHAT REFACTORING COULD FIX

- Additional speed improvements (perceived & actual)
- Easier maintenance, less bugs
- Search engines and other non-JavaScript crawlers
- Better UX

...

BUMBLEBEE IS SHOWING ITS AGE




Feedback

ORCID

About

Account

 **astrophysics** data system

Classic Form

Modern Form

Paper Form

QUICK FIELD:

Author

First Author

Abstract

Year

Fulltext

All Search Terms

author

author:"huchra, john"

first author

author:"^huchra, john"

abstract + title

abs:"dark energy"

year

year:2000

year range

year:2000-2005

full text

full:"gravitational waves"

publication

bibstem:ApJ

citations

citations(author:"huchra, j")

references

references(author:"huchra, j")

reviews

reviews("gamma-ray bursts")

refereed


property:refereed

astronomy


database:astronomy

OR


abs:(planet OR star)



Use a classic ADS-style form



Learn more about searching the ADS



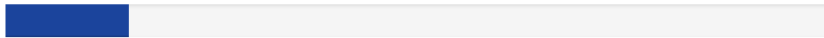
Access ADS data with our API

http://localhost:8001/#/

10/47

USER ISSUES

astrophysics
data system



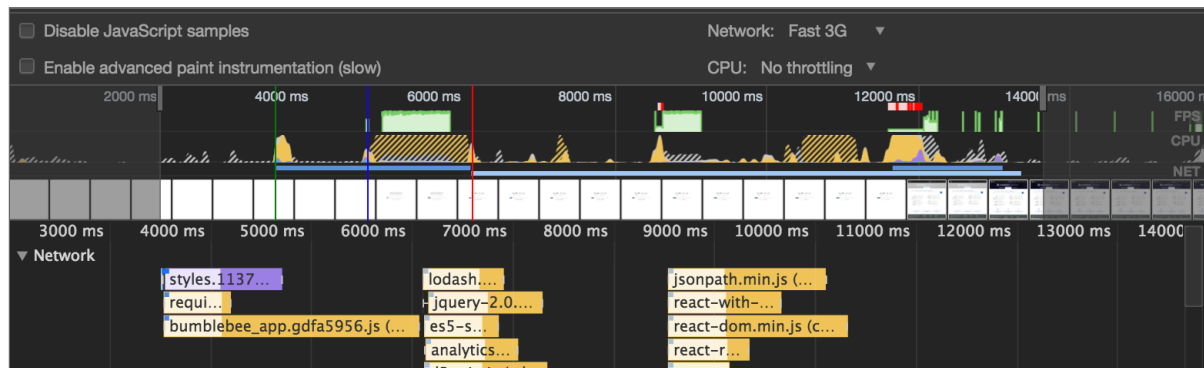
Downloading Assets



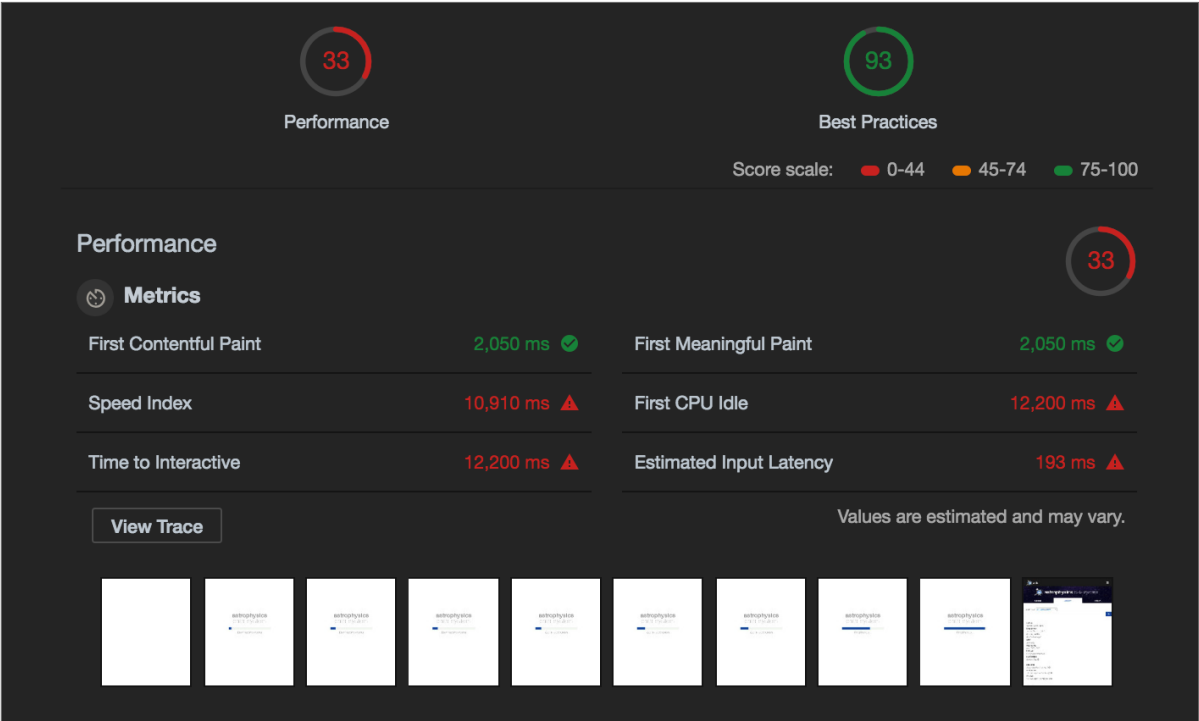
LOAD TIME

Average application load time >1s

Many unnecessary page renders/reflows



LIGHTHOUSE

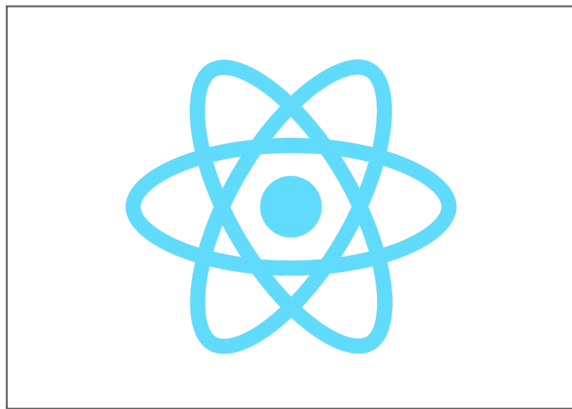


WHY?

- Based on older (+4 years) tech stack
- Hard to maintain
- Hard to integrate with user applications
- Bad mobile performance

TECH STACK

BackboneJs & React



MAINTENANCE

- Separation of concerns
- Complicated builds
- Too many dependencies (100+ 3rd party libs)
- Testing is hard

INTEGRATIONS & SEARCH ENGINE OPTIMIZATION

- Zotero, Bookends, etc.
- Unfurl links
- Scrapers, non-JavaScript users
- Google, Bing, others


UNFURL LINKS

July 21st, 2017


10:37 AM

tim

<https://stackoverflow.com/questions/45418820/how-does-webpack-actually-works-what-does-it-does-behind-the-loaders>

stackoverflow.com

How does Webpack actually works? What does it does behind the loaders?




Every video about what Webpack is just tell you about general concepts like "it's a module bundler, it's take a module and bundle it", but I always wanted to know, how it's engine works?

today

5:02 PM

tim

<https://ui.adsabs.harvard.edu/#abs/2018EPJWC.18608001A/abstract>

ui.adsabs.harvard.edu

NASA/ADS Search

A powerful, streamlined new Astrophysics Data System

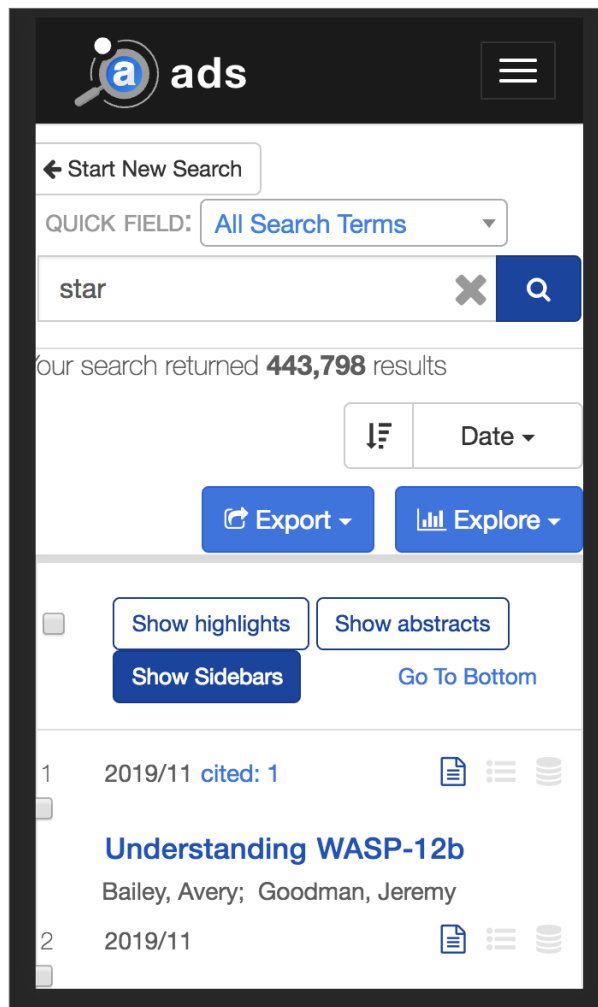
+

Jot something down

USER EXPERIENCE

- Not completely responsive
- Some negative user feedback
- Confusing or hidden features

MOBILE RESPONSIVENESS



ACCESSIBILITY

- Facets not usable by screen readers
- Page regions can get readers into loops
- Some sections inaccessible

IT GETS BETTER

TYPES OF WEB APPLICATIONS

- Server-rendered (traditional)
- Client-rendered (single page application)
- Hybrid (universal)

HYBRID APPROACH

- Lives on both client and server
- Search engine optimization is easier
- Less load on server

PROPOSAL

- Hybrid web app
- New tech stack
- Updated UI framework

NEW TECH STACK

- Vue.js
- Node.js server



HOW WILL IT BE BETTER?

- Better state management
- Easier maintenance
- search engine optimization, non-JavaScript crawler support
- Responsive

PERFORMANCE GOALS

- Consistent load times < 1s
- Fewer page renders, utilizing virtual DOM
- Pre-rendering on server, hydrate on client

SEO, NON-JAVASCRIPT GOALS

- Metadata rendered server-side
- Non-JavaScript users possible
- Google, Bing, unfurl crawlers should just work

USER EXPERIENCE

- Consolidate and simplify interface
- Easier to implement user requested features
- Better mobile usability
- Prioritize based on user feedback

EASIER MAINTENANCE

- Support newer language features
- Better tooling
- Batteries included (fewer dependencies)
- Simplified builds
- Easier testing

HOW WE GET THERE

- Visual components need to be refactored
- User-facing part should stay mostly untouched
 - Improvements to responsiveness
 - Restructure of certain areas to provide better UX
- Gradual transition to new system

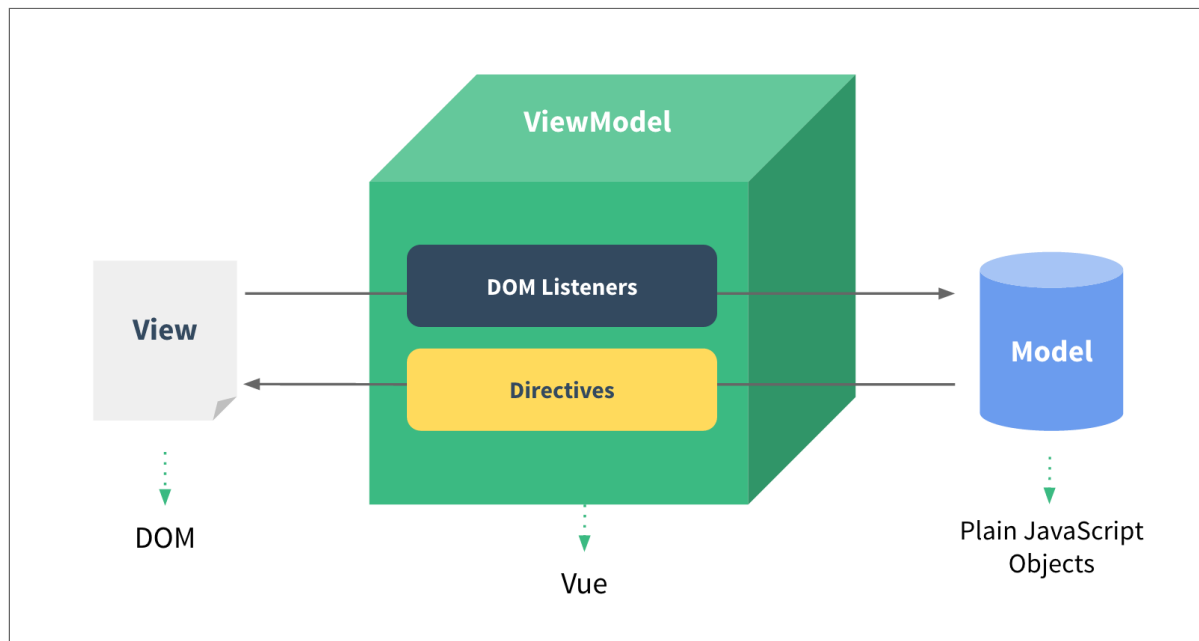
QUESTIONS?

...

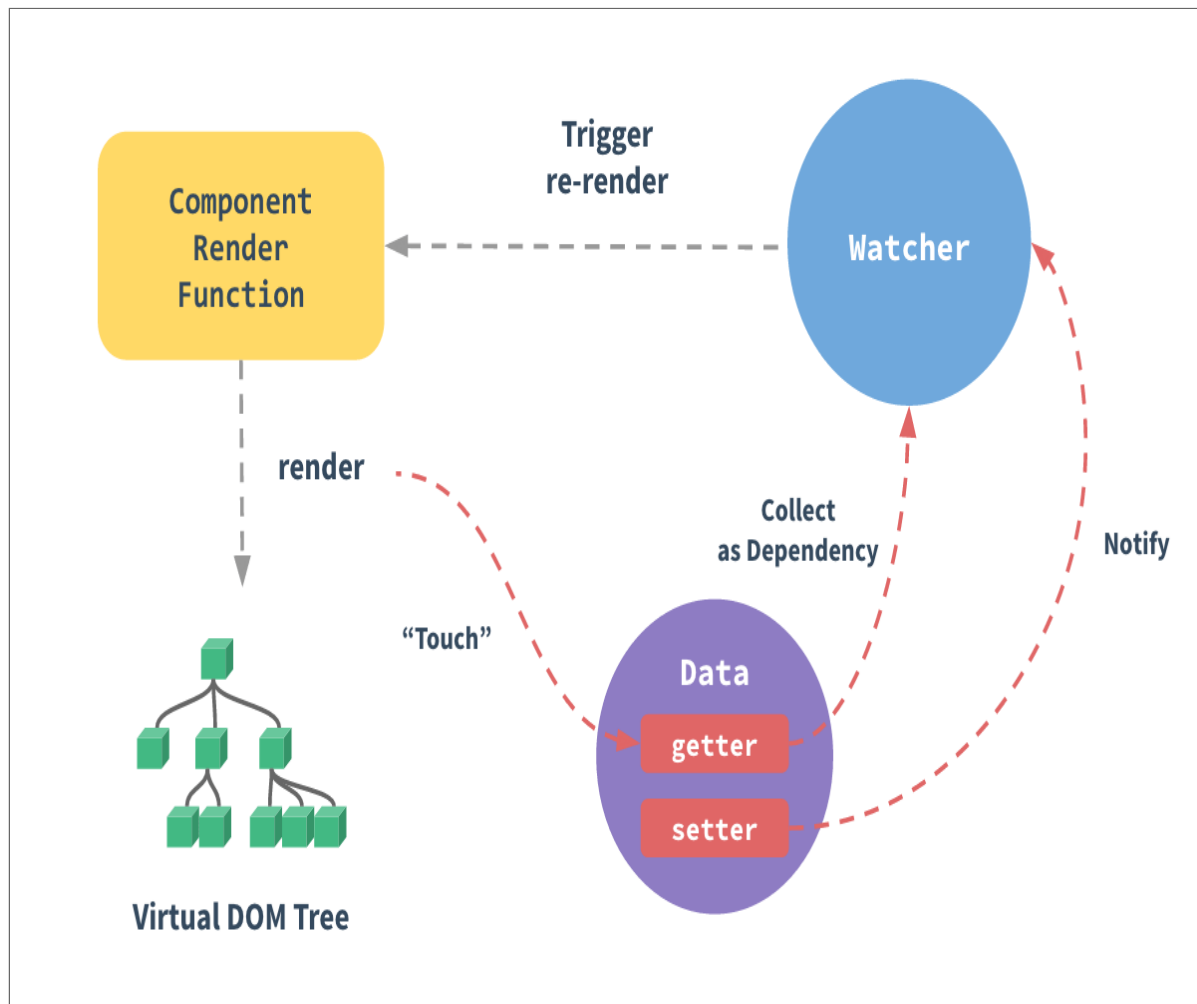
VUE INTERNALS

- MVVM pattern
- Action -> Model -> Diff -> Render
- Virtual DOM

VUE MVVM

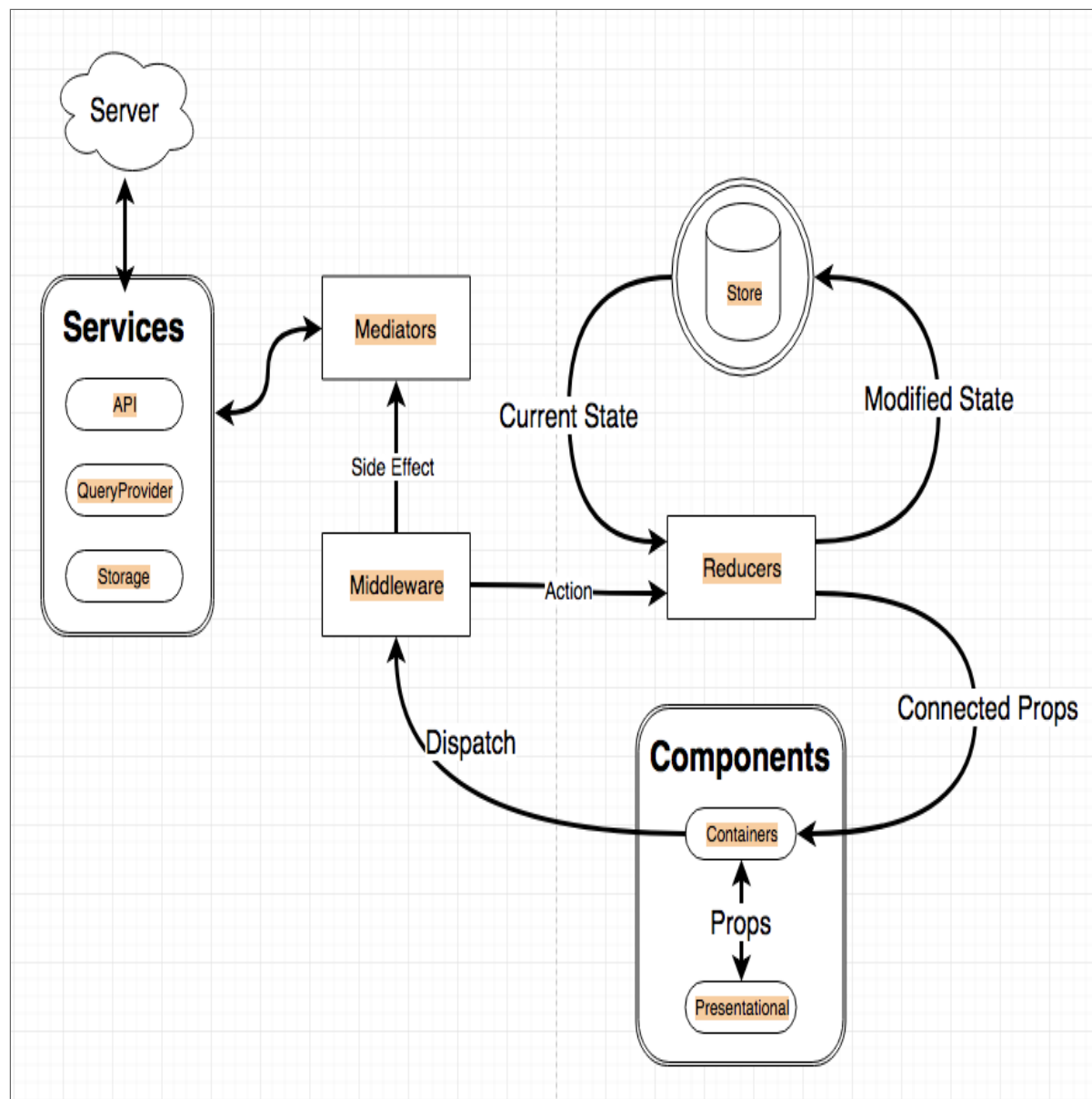


VUE DATABINDING

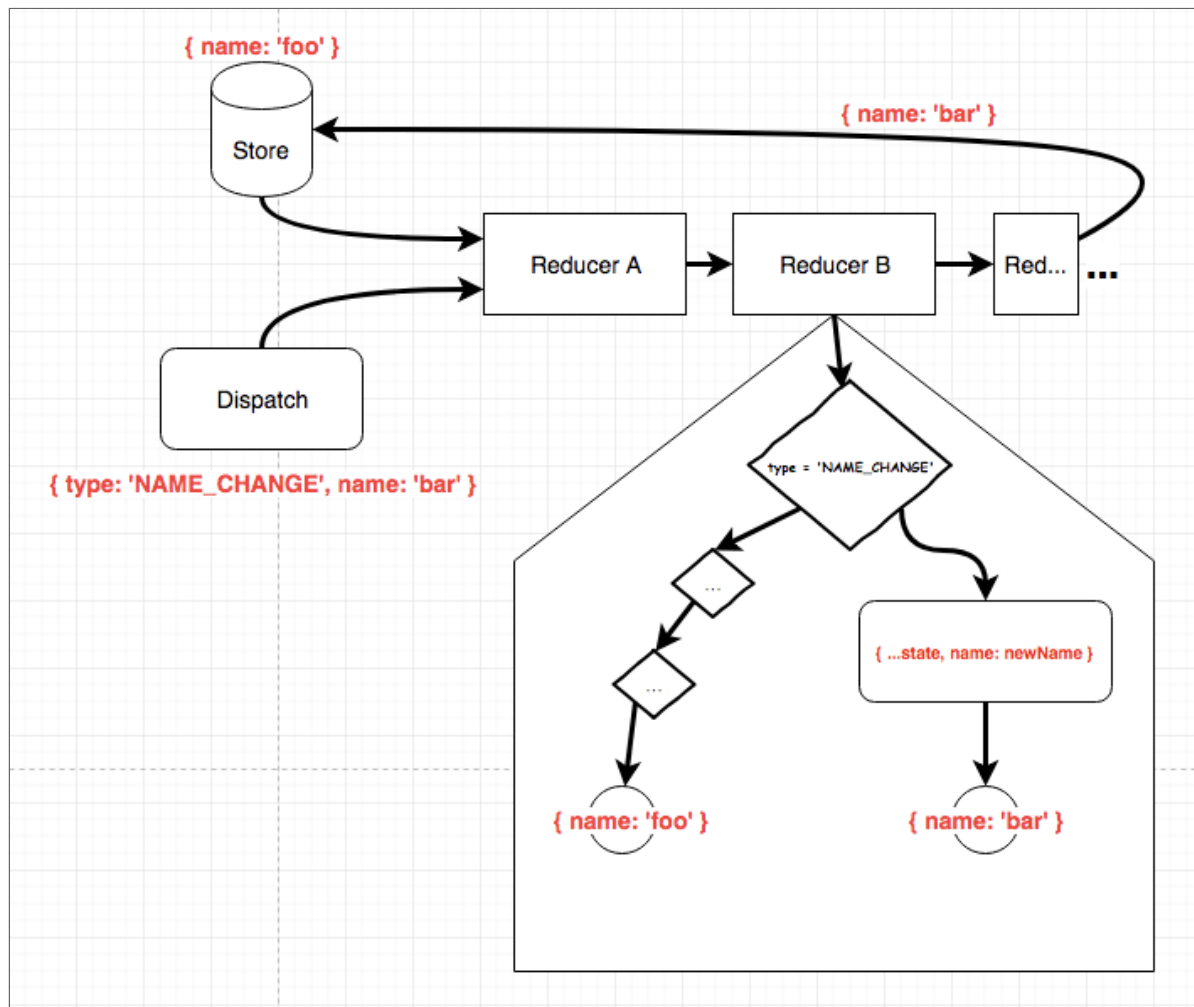


FLUX

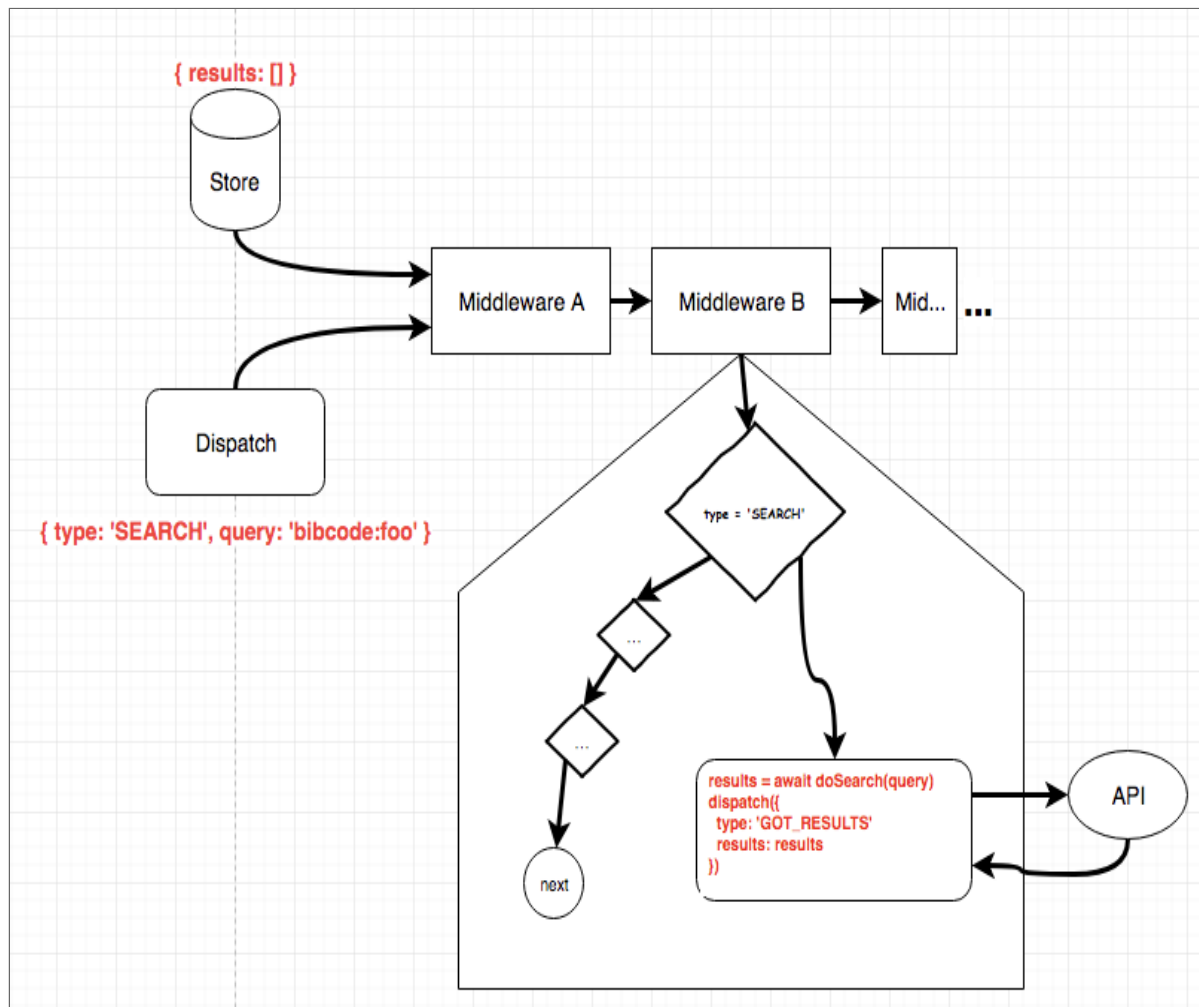
- Technique for state management
- Reducers and single store
- Non-immutable data structures
- Pure functions



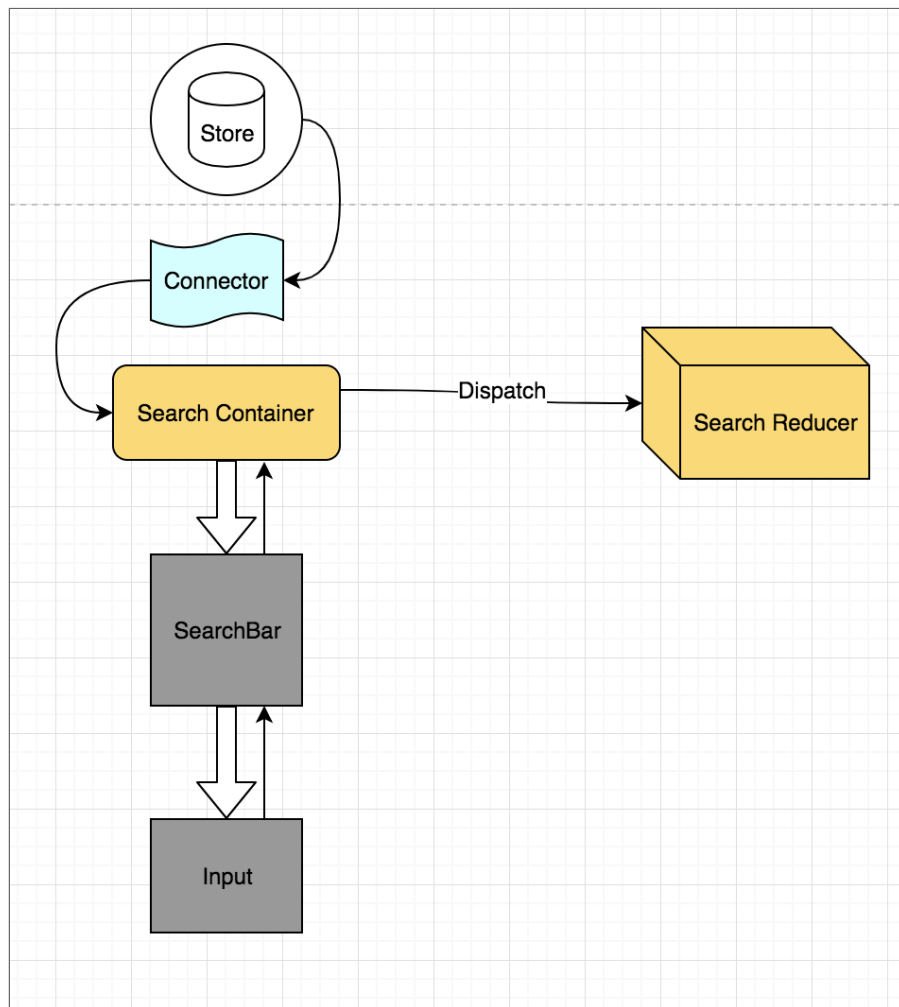
REDUCERS



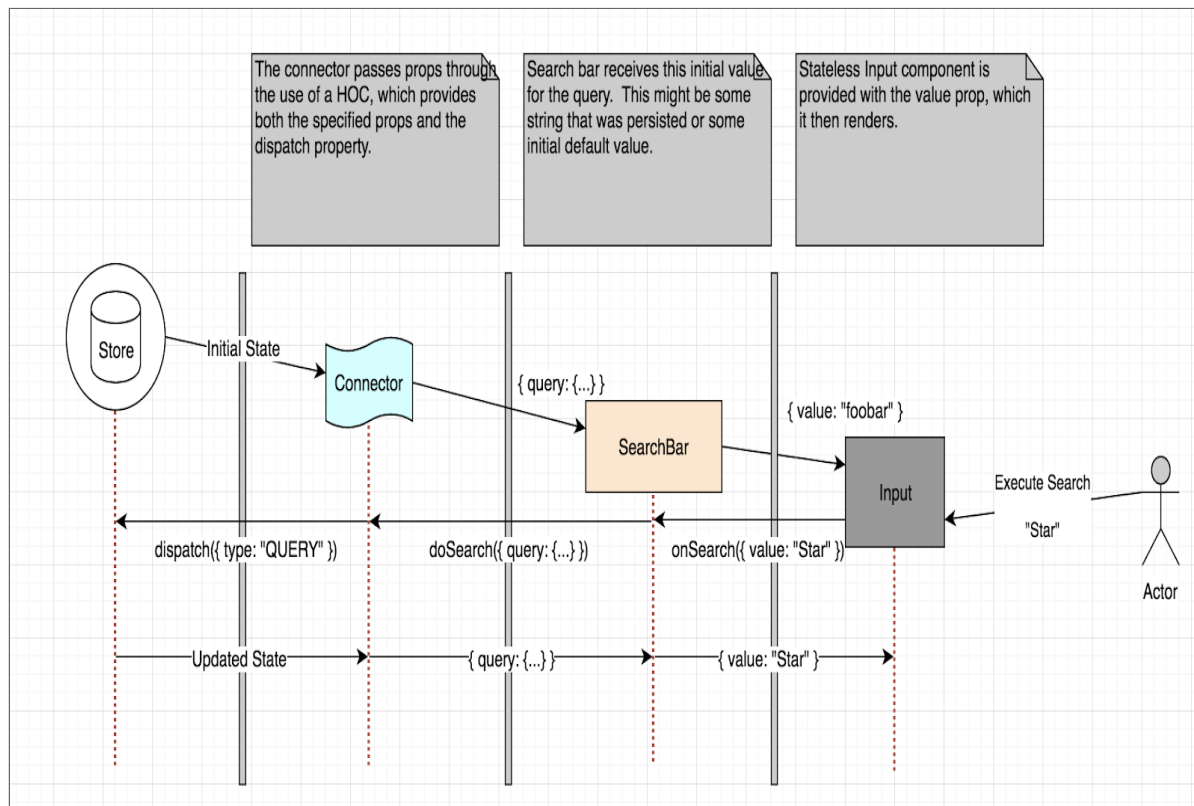
MIDDLEWARE



COMPONENT COMMUNICATION



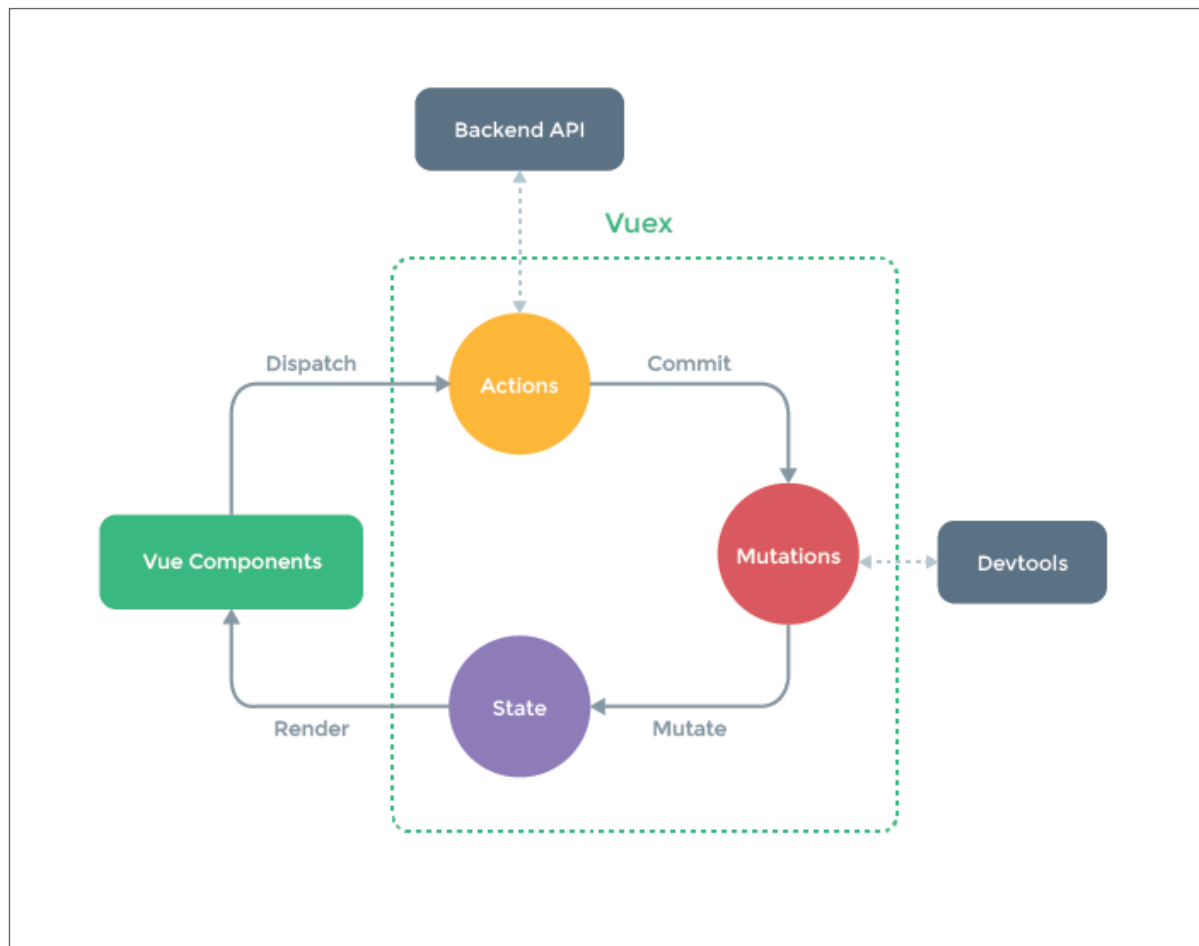
CONNECTED COMPONENTS



VUEX

- Vue's implementation of Flux
- Baked into the core framework

VUEX



NUXT

- SSR skeleton/framework for Vue
- Provides scripts, hooks, middleware, etc.
- Based on Node.js server
- Selective rendering
- Bundling of client/server automatically

