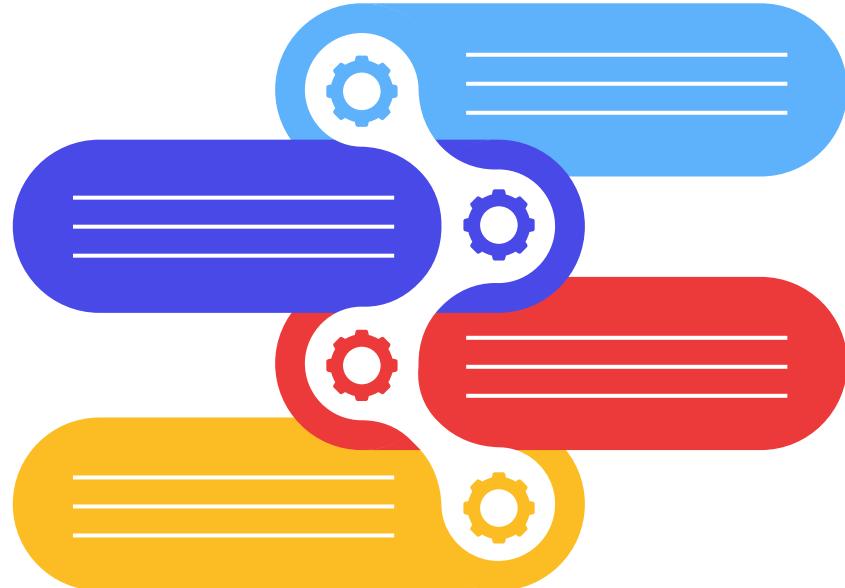


Accelerating the Assembly Process

Reducing the Cycle Time



BUTTON



BUTTON

None

0 Rev

Stage: checks



lint



one
label



requests



one
iles



test



one

```
-----  
1726 Test Suites: 11 skipped, 241 passed, 241 of 252 total  
1727 Tests:       24 skipped, 1560 passed, 1584 total  
1728 Snapshots:   198 passed, 198 total  
1729 Time:         348.178 s  
1730 Ran all test suites.
```

Duration: 11 minutes 32 seconds

Finished: 1 day ago

Queued: 2 seconds

Timeout: 1h (from project) 

Runner: #2610631 (39ydBTSsE) buli



Me 10:23 AM

guys, check my [new MR!3914](#)



Me 10:23 AM

guys, check my [new MR!3914](#)



Teammate 10:40 AM

tests were failed 😢



Me 10:23 AM

guys, check my [new MR!3914](#)



Teammate 10:40 AM

tests were failed 😢



Me 10:42 AM

fixed



Me 10:23 AM

guys, check my [new MR!3914](#)



Teammate 10:40 AM

tests were failed 😢



Me 10:42 AM

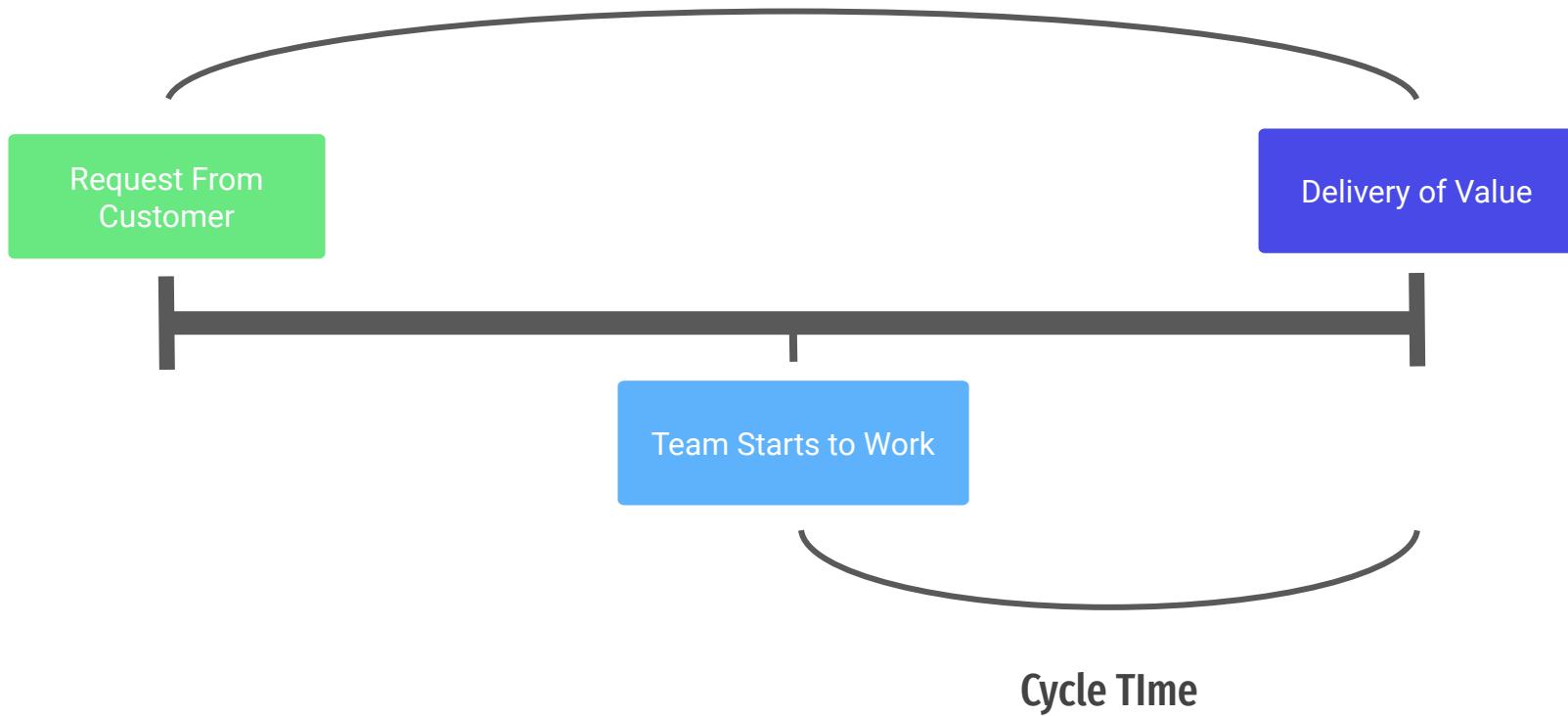
fixed



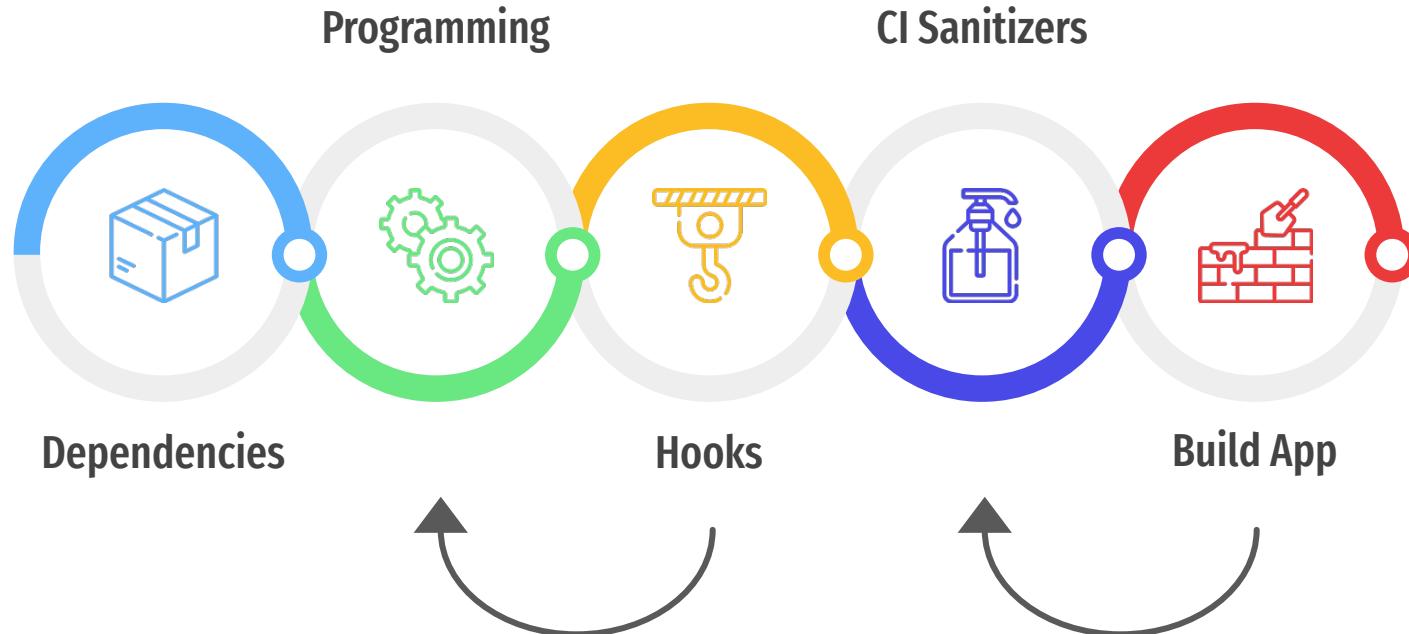
Teammate 11:40 AM

linter is not okay 😓

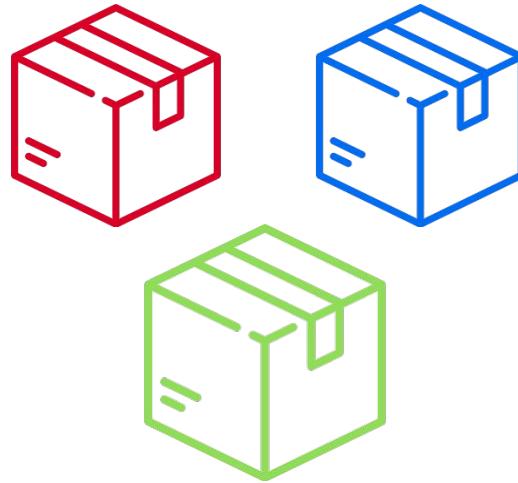
Lead Time



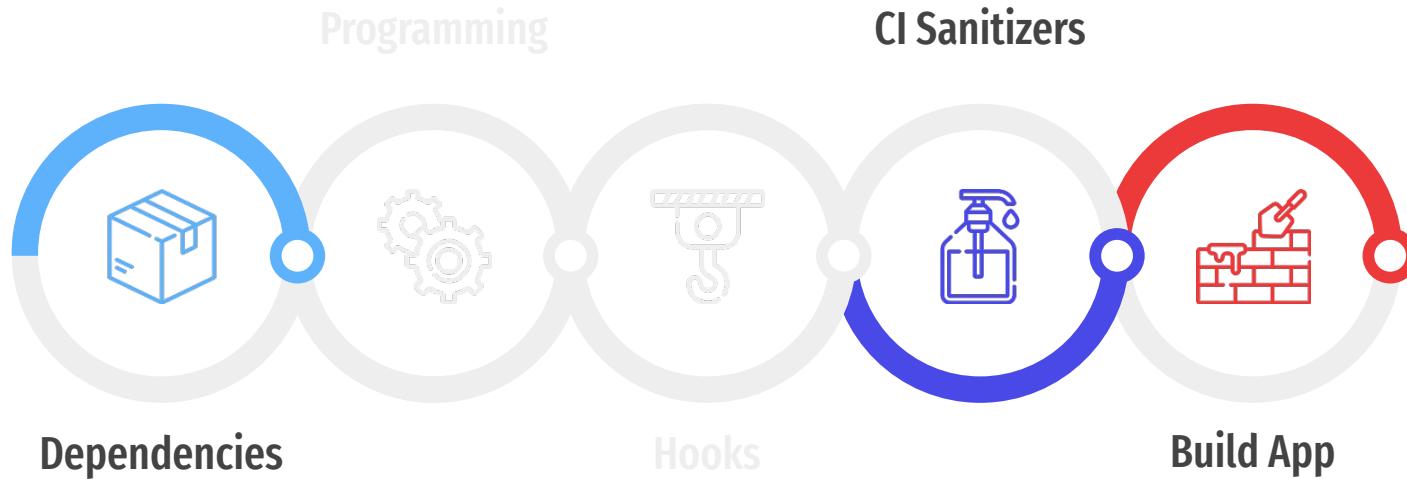
Cycle Time



Package Manager



Package Manager



Package Managers



CI cache

```
test-job:  
  stage: build  
  cache:  
    key: $CI_COMMIT_REF_SLUG  
    paths:  
      - .npm/  
  script:  
    - npm ci --cache .npm --prefer-offline  
    - echo Run tests...
```

CI cache

~1min



~29s

Clean Install

npm i



npm ci

Scripts life-cycle

pre<event>

<event>

post<event>

npm ci

- preinstall
- install
- postinstall
- prepublish
- preprepare
- prepare
- postprepare

npm ci --ignore-scripts

- **preinstall**

- **install**

- **postinstall**

- **prepublish**

- **preprepare**

- **prepare**

- **postprepare**

npm ci --ignore-scripts

- `install`
- `prepare`

npm ci --ignore-scripts



npm ci --ignore-scripts

```
curl --silent -o-
https://raw.githubusercontent.com/oresoftware/re
alpath/master/assets/install.sh | bash
```

npm ci --ignore-scripts

~29s



~25s

npm ci --production

- devDependencies
- dependencies

npm ci --production

- ~~devDependencies~~
- dependencies

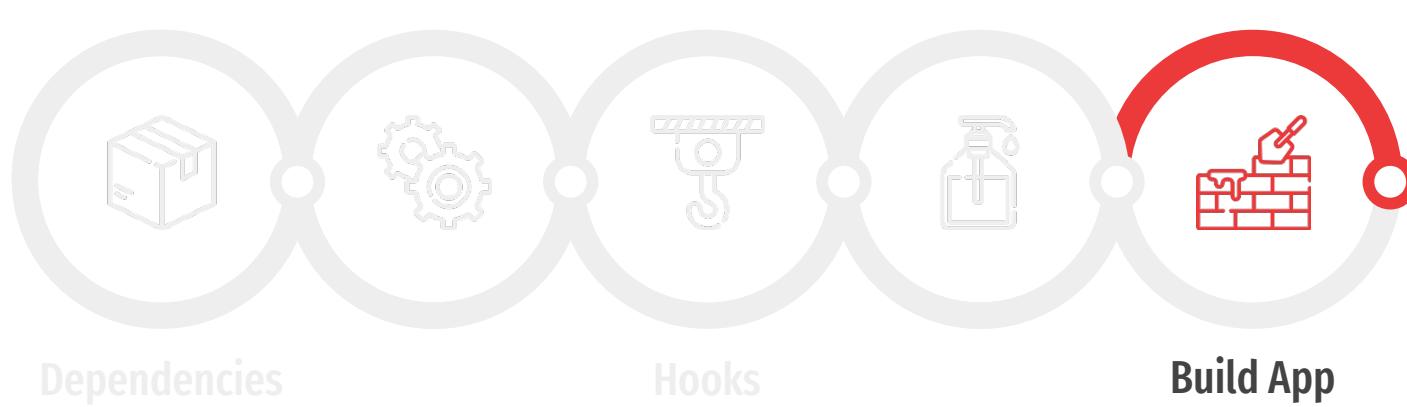
npm ci --production

~29s



~10s

npm ci --production



why-npm-i-so-long

```
why-npm-i-so-long
▲ why-npm-i-so-long ~/project/package.json
my-project's dependencies:

resolve @ 1.14.2
  install size    104 kB
  publish size   94.1 kB

fs-extra @ 8.1.0
  install size    175 kB
  publish size   125 kB

fast-glob @ 3.1.1
  install size    446 kB
  publish size   74.0 kB

moment @ 2.24.0
  install size    2.66 MB
  publish size    2.66 MB

typescript @ 3.7.4
  install size    48.2 MB
  publish size    48.2 MB

~
```

why-npm-i-so-long

publish size

vs

install size

depcheck

- Unused dependencies
- Unused devDependencies
- Missing dependencies

NPM doppelgangers

```
{  
  "name": "library-a",  
  "version": "1.0.0",  
  "dependencies": {  
    "library-b": "^1.0.0",  
    "library-c": "^1.0.0",  
    "library-d": "^1.0.0",  
    "library-e": "^1.0.0"  
  }  
}
```

NPM doppelgangers

```
{  
  "name": "library-b", // library-c  
  "version": "1.0.0",  
  "dependencies": {  
    "library-f": "^1.0.0"  
  }  
}
```

NPM doppelgangers

```
{  
  "name": "library-d", // library-e  
  "version": "1.0.0",  
  "dependencies": {  
    "library-f": "^2.0.0"  
  }  
}
```

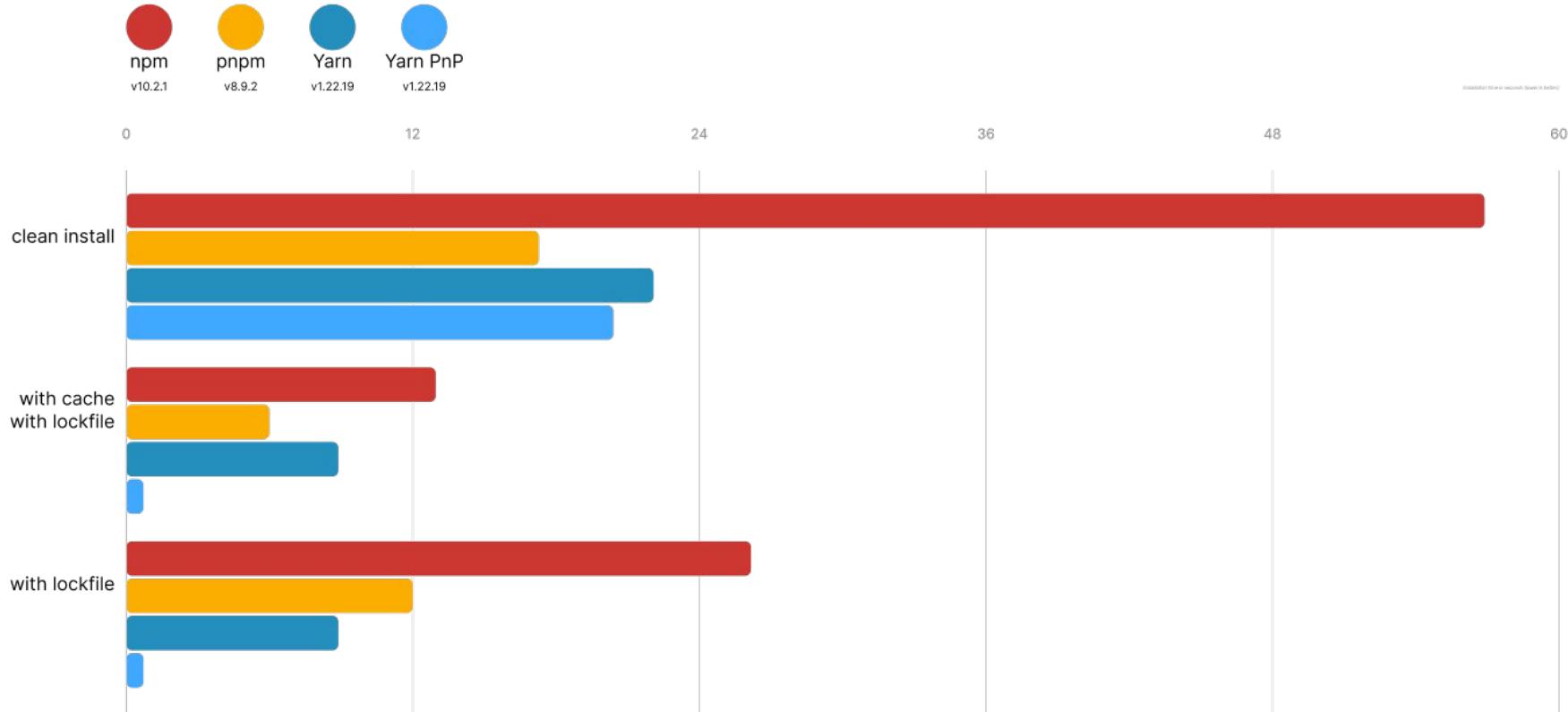
NPM doppelgangers

```
{  
  "name": "library-a",  
  "version": "1.0.0",  
  "dependencies": {  
    "library-b": "^1.0.0", <--library-f@1.0.0  
    "library-c": "^1.0.0", <--library-f@1.0.0  
    "library-d": "^1.0.0", <--library-f@2.0.0  
    "library-e": "^1.0.0" <--library-f@2.0.0  
  }  
}
```

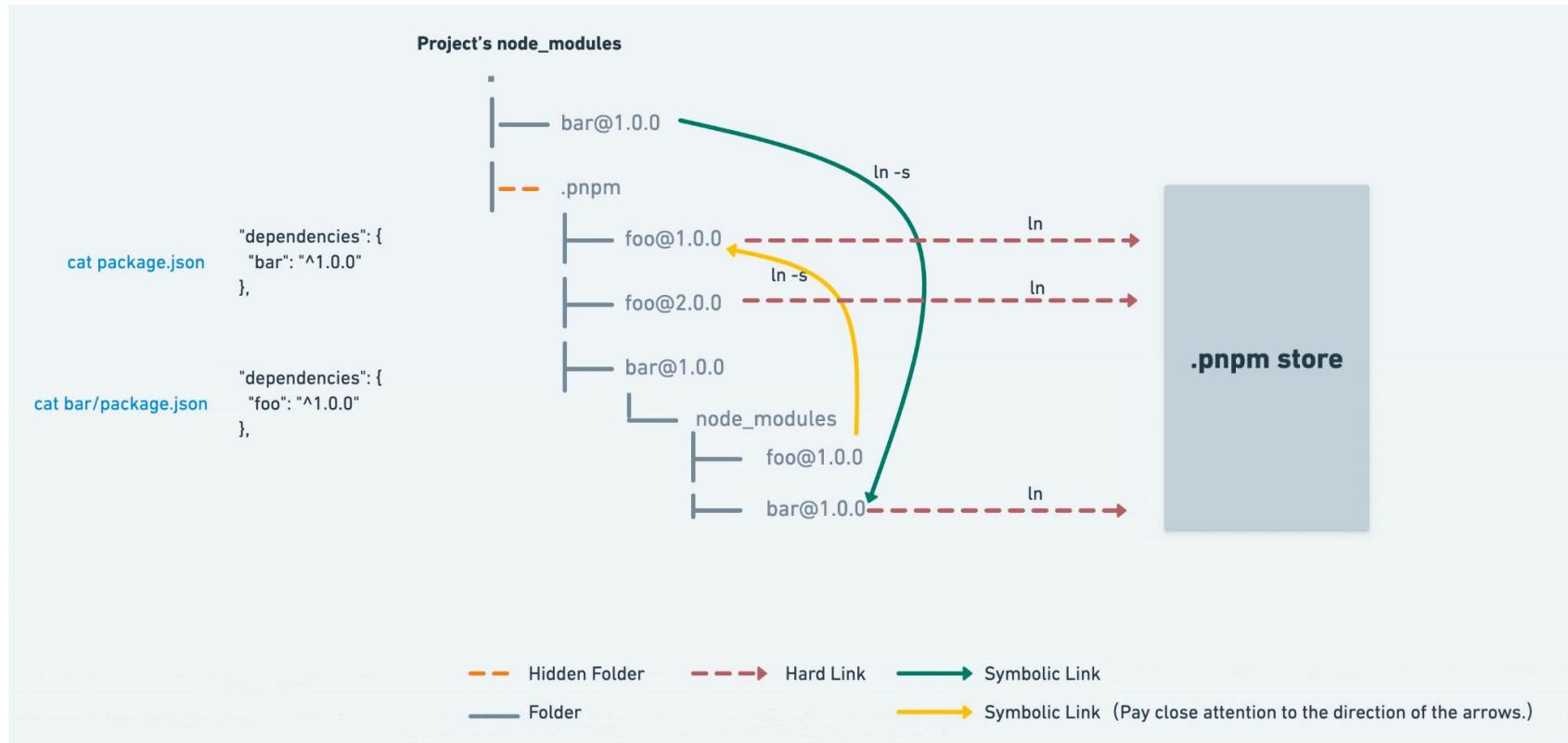
NPM doppelgangers

- node_modules/
 - library-b/
 - library-c/
 - library-d/
 - node_modules/
 - library-f/
 - package.json <-- **library-f@2.0.0**
 - library-e/
 - node_modules/
 - library-f/
 - package.json <-- **library-f@2.0.0**
 - library-f/
 - package.json <-- **library-f@1.0.0**

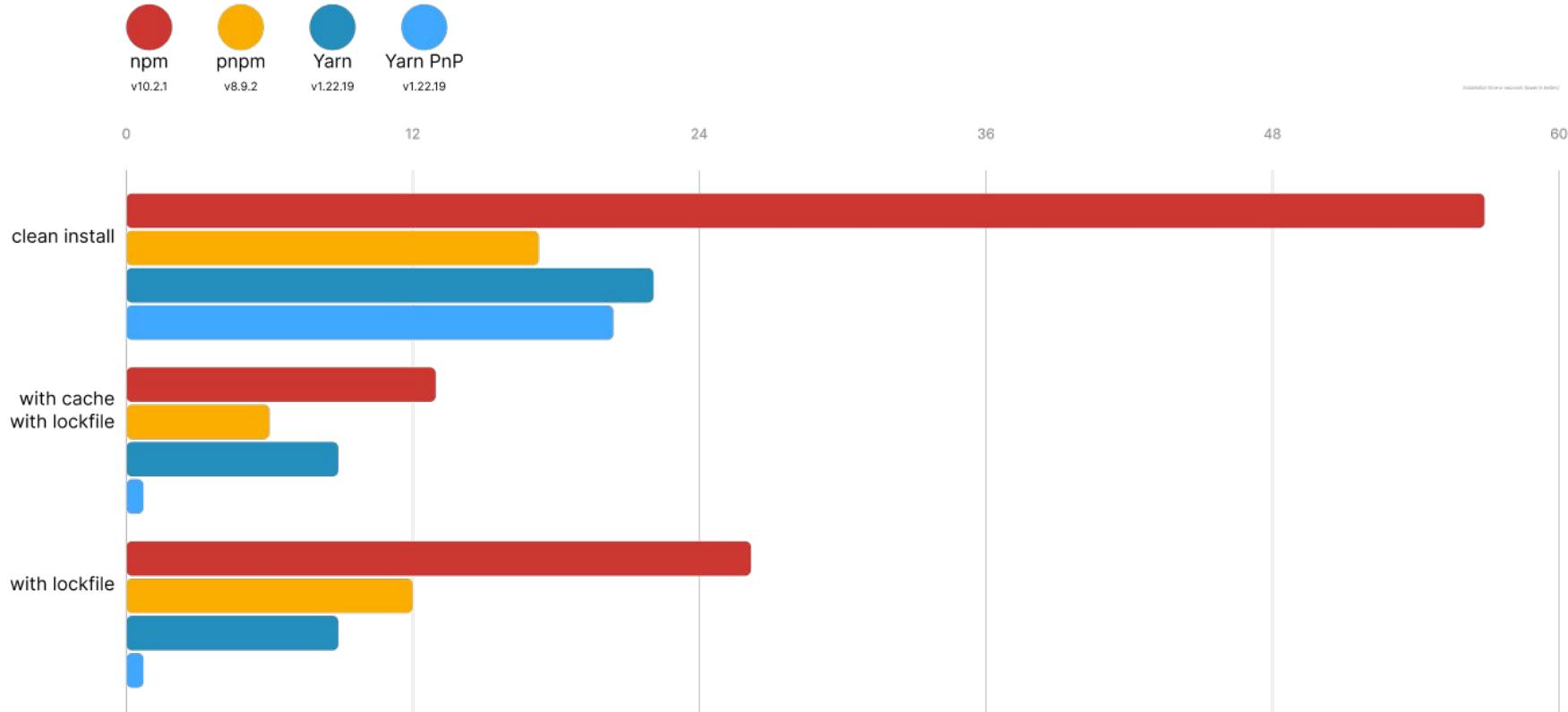
Package Managers



pnpm

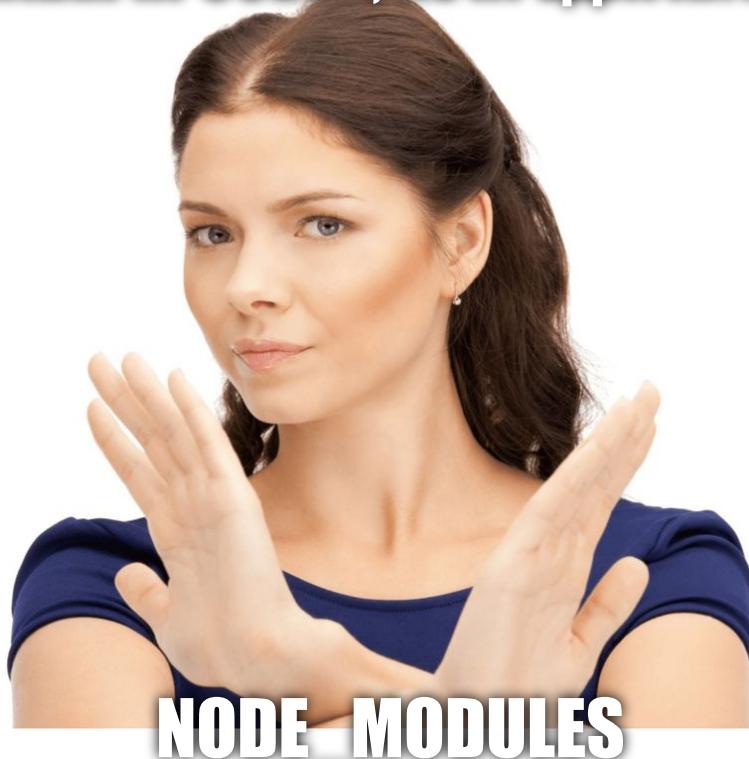


Package Managers



Yarn PnP

ОГРОМНАЯ ПРОСЬБА, НЕ ПРЕДЛГАЙТЕ МНЕ



NODE_MODULES

Yarn PnP

~~node_modules~~



.pnp.cjs



~/.yarn/berry/cache/react-npm-18.2.0-1eae08fee2-10c0.zip/node_modules/react/

Yarn PnP

```
2759 |   ["punycode", "npm:2.3.0"]\n2760 | },\n2761 |   "linkType": "HARD"\n2762 |\}\}\,\n2763 ]],\n2764 [\"queue-microtask\", [\n2765 |  [\"npm:1.2.3\", {\n2766 |   \"packageLocation\": \"./././.yarn/berry/cache/queue-microtask-npm-1.2.3-fcc98e4e2d-10c0.zip/node_modules/queue-microtask/\",\n2767 |   \"packageDependencies\": [\n2768 |     [\"queue-microtask\", \"npm:1.2.3\"]\n2769 |   ],\n2770 |   \"linkType\": \"HARD\"\n2771 |\}\}\,\n2772 ]],\n2773 [\"react\", [\n2774 |  [\"npm:18.2.0\", {\n2775 |    \"packageLocation\": \"./././.yarn/berry/cache/react-npm-18.2.0-1ea08fee2-10c0.zip/node_modules/react/\",\n2776 |    \"packageDependencies\": [\n2777 |      [\"react\", \"npm:18.2.0\"],\n2778 |      [\"loose-envify\", \"npm:1.4.0\"]\n2779 |    ],\n2780 |    \"linkType\": \"HARD\"\n2781 |\}\}\,\n2782 ]],\n2783 [\"react-dom\", [\n2784 |  [\"npm:18.2.0\", {\n2785 |    \"packageLocation\": \"./././.yarn/berry/cache/react-dom-npm-18.2.0-dd675bc1c-10c0.zip/node_modules/react-dom/\",\n2786 |    \"packageDependencies\": [\n2787 |      [\"react-dom\", \"npm:18.2.0\"]\n2788 |    ],\n2789 |    \"linkType\": \"SOFT\"\n2790 |  }\n2791 | }]\n2792 ]]\n2793 }
```

Yarn PnP

ОГРОМНАЯ ПРОСЬБА, НЕ ПРЕДЛГАЙТЕ МНЕ



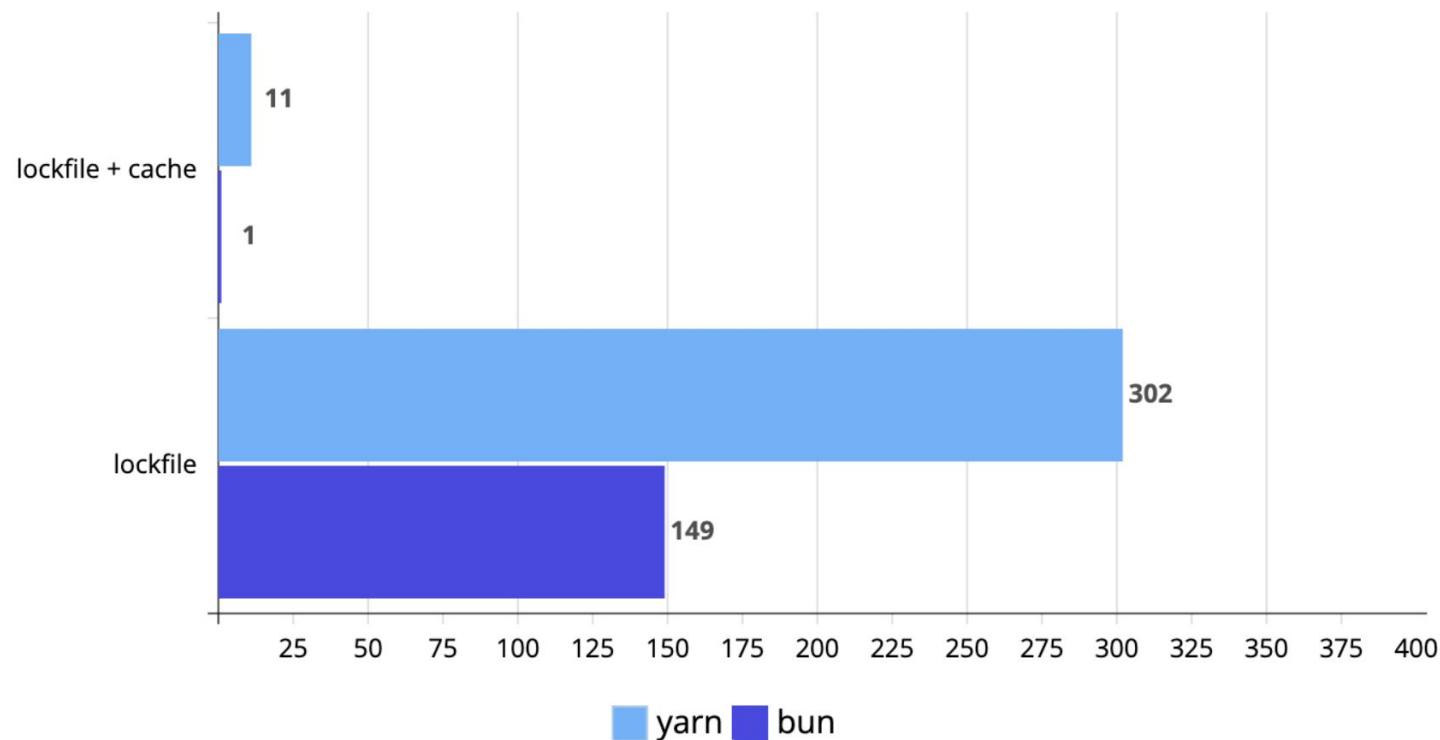
YARN PNP

Bun

| | | |
|------|---|--------|
| Bun |  | 00.36s |
| pnpm | <i>17x slower</i> | 06.44s |
| npm | <i>29x slower</i> | 10.58s |
| Yarn | <i>33x slower</i> | 12.08s |

Installing dependencies from cache for a Remix app.

Bun



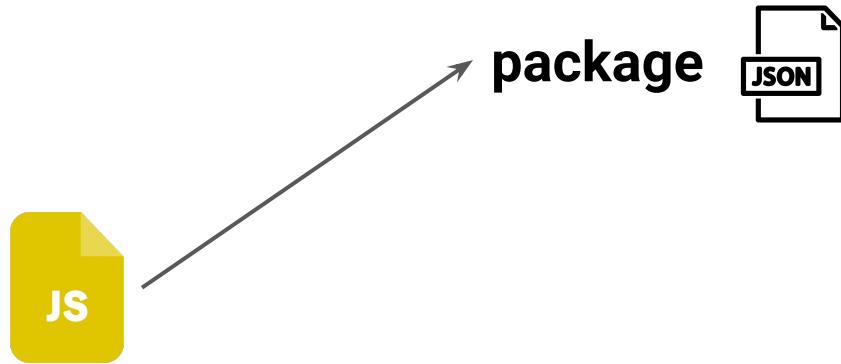
Bump versions

```
5 @adobe/css-tools@^4.0.1:  
6   version "4.0.1"  
7   resolved "https://registry.yarnpkg.com/@adobe/css-tools/-/css-tools-4.  
8   integrity sha512-u76oB43n0HrF4DDWRLWDctci7f3QJoEBigemIdIeTi10Dqjx6Ta  
9  
10 @adyen/adyen-web@^4.7.3:  
11   version "4.7.3"  
12   resolved "https://registry.yarnpkg.com/@adyen/adyen-web/-/adyen-web-4.  
13   integrity sha512-PScBuaIxFlVP0s50z5Yu2tknTQUJzPCboXzVHaFLpJzQrohg0kkC  
14   dependencies:  
15     "@types/applepayjs" "^3.0.2"  
16     "@types/googlepay" "^0.6.2"  
17     classnames "^2.3.1"  
18     core-js "^3.14.0"  
19     preact "^10.5.13"  
20  
21 @ampproject/remapping@^2.1.0:  
22   version "2.2.0"  
23   resolved "https://registry.yarnpkg.com/@ampproject/remapping/-/remapping-  
24   integrity sha512-qRmj8nj9qmLTQXXmaR1cck3UXSRMPrbsLJAasZpF+t3riI71BXec  
25   dependencies:  
26     "@jridgewell/gen-mapping" "^0.1.0"  
27     "@jridgewell/trace-mapping" "^0.3.9"  
28  
29 @babel/code-frame@^7.0.0:  
30   version "7.16.0"  
31   resolved "https://registry.yarnpkg.com/@babel/code-frame/-/code-frame-  
32   integrity sha512-IF4E0MEV+bFyW0mNxGzSnjR2EmQod7f1UX0pZM3l4i4o4QNwzjtJA  
33   dependencies:  
34     "@babel/highlight" "^7.16.0"  
35  
36 @babel/code-frame@^7.10.4", "@babel/code-frame@^7.12.13", "@babel/code-  
37   version "7.18.6"  
38  
39 @adobe/css-tools@^4.0.1:  
40   version "4.3.1"  
41   resolved "https://registry.npmjs.org/@adobe/css-tools/-/css-tools-4.3.  
42   integrity sha512-62yikz7NLScCGAASTSHdnjaDJQBDq0M2muyRTpf2VQhw6StBg2/  
43  
44 @adyen/adyen-web@^4.7.3:  
45   version "4.9.0"  
46   resolved "https://registry.npmjs.org/@adyen/adyen-web/-/adyen-web-4.9.  
47   integrity sha512-qLMt7JaA2SCo0tT2oiindrGU+CuUVqF35JQHYrc13buuWQmMHTQS  
48   dependencies:  
49     "@types/applepayjs" "^3.0.2"  
50     "@types/googlepay" "^0.6.2"  
51     classnames "^2.3.1"  
52     core-js "^3.18.1"  
53     preact "^10.5.13"  
54  
55 @ampproject/remapping@^2.2.0:  
56   version "2.2.1"  
57   resolved "https://registry.npmjs.org/@ampproject/remapping/-/remapping-  
58   integrity sha512-lFMjJTrFL3j7L9yBxwYfcq2k6qqwhyzuUl/XBnif78PWTJYyL/dfc  
59   dependencies:  
60     "@jridgewell/gen-mapping" "^0.3.0"  
61     "@jridgewell/trace-mapping" "^0.3.9"  
62  
63 @babel/code-frame@^7.0.0", "@babel/code-frame@^7.10.4", "@babel/code-  
64   version "7.22.13"  
65   resolved "https://registry.npmjs.org/@babel/code-frame/-/code-frame-7.  
66   integrity sha512-XktuhWlJ5g+3TJXc5upd9Ks1HutSArik6jf2eAjYFyI0f4ej3RN+j  
67   dependencies:  
68     "@babel/highlight" "^7.22.13"  
69     chalk "^2.4.2"  
70  
71 @babel/compat-data@^7.22.9:  
72   version "7.23.2"
```

Custom script



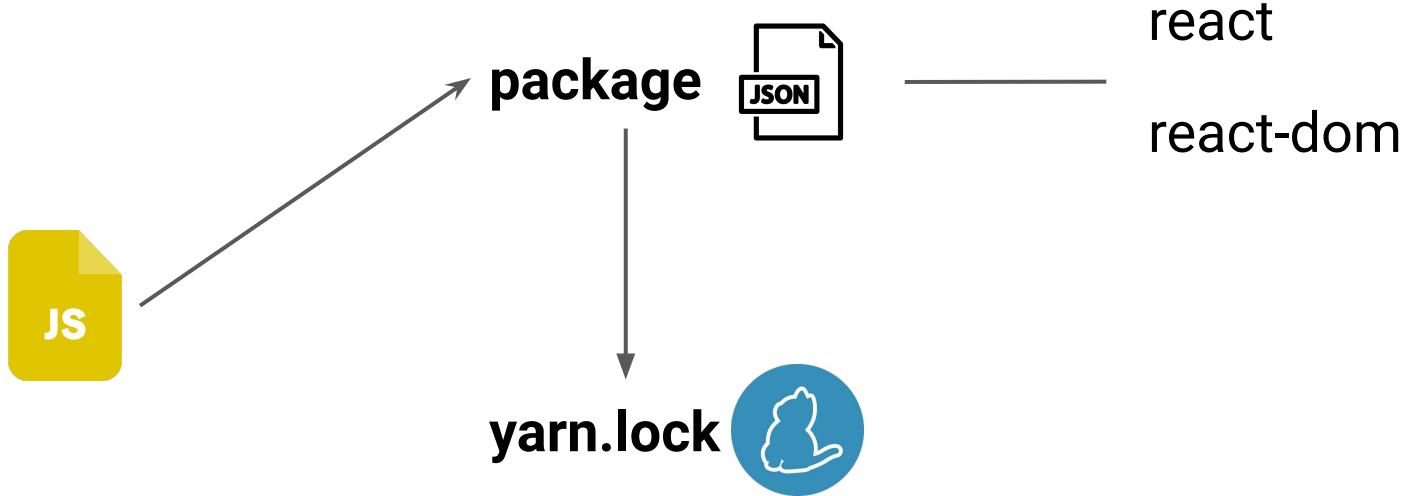
Custom script



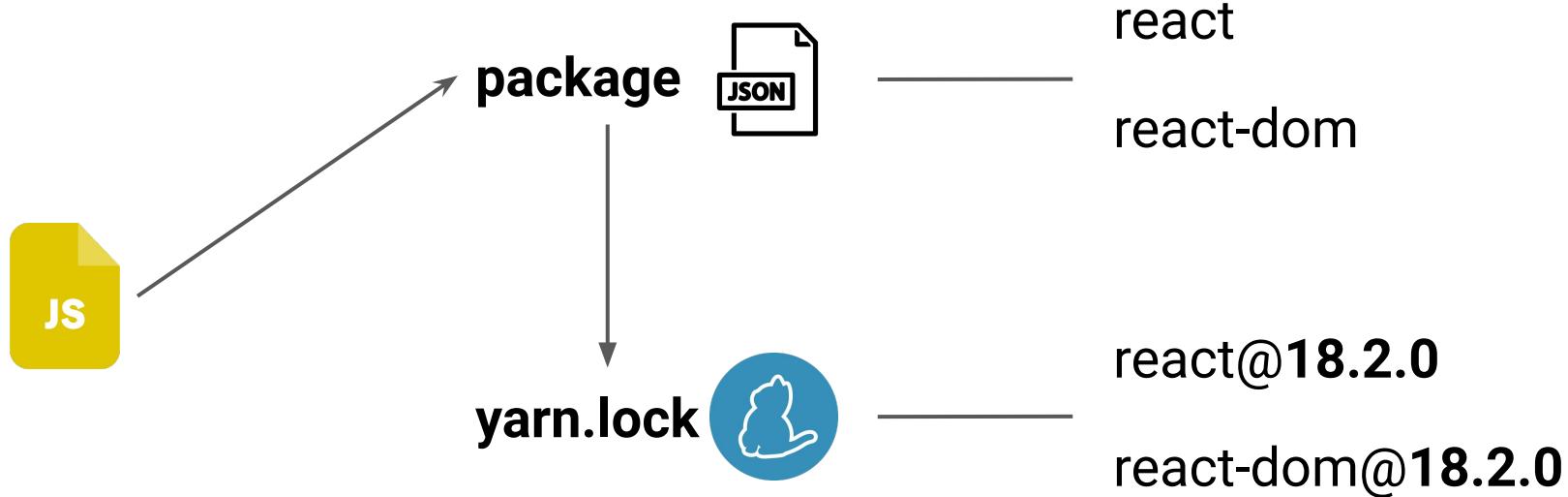
Custom script



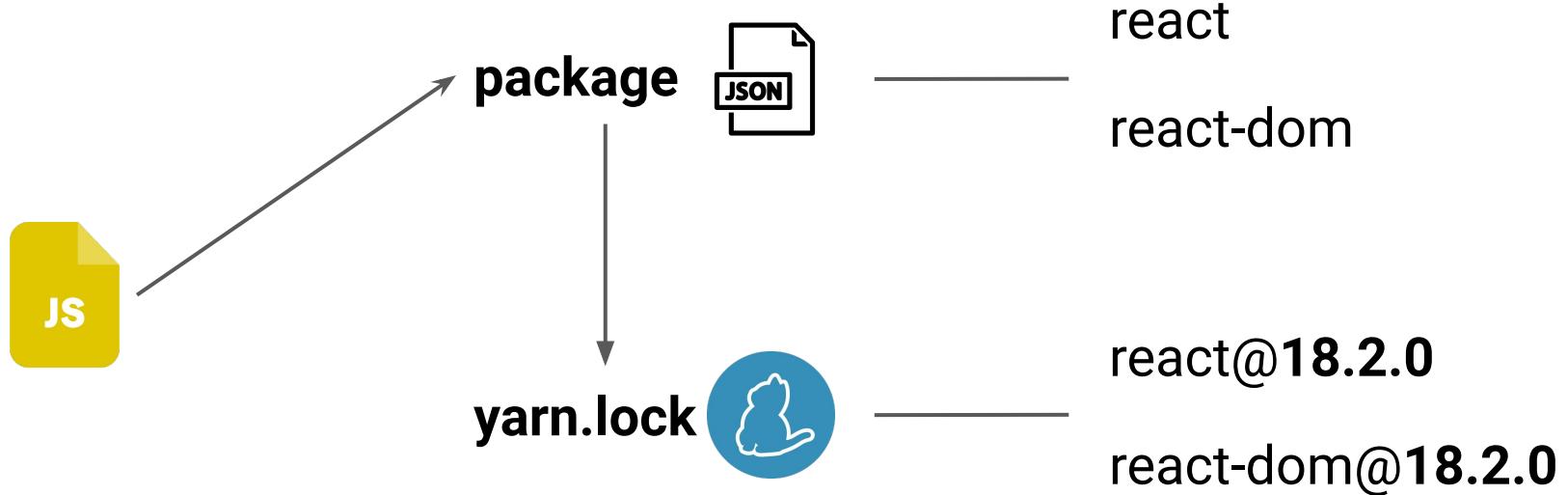
Custom script



Custom script



Custom script



bun install react@18.2.0 react-dom@18.2.0

Sum Up

Improve current PM

- ci cache
- test install flags
- check your deps

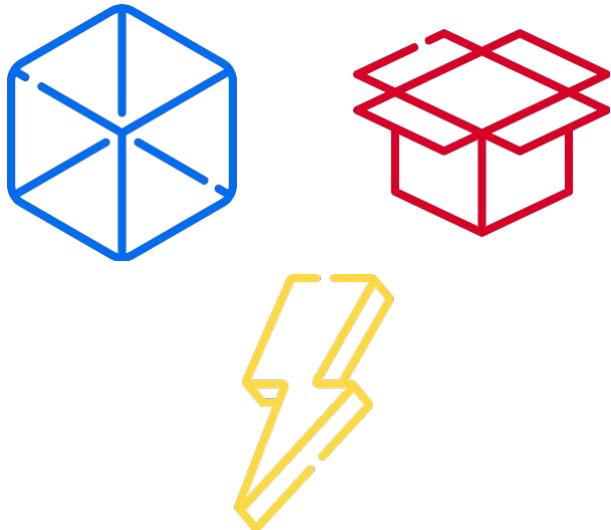
Sum Up

Improve current PM

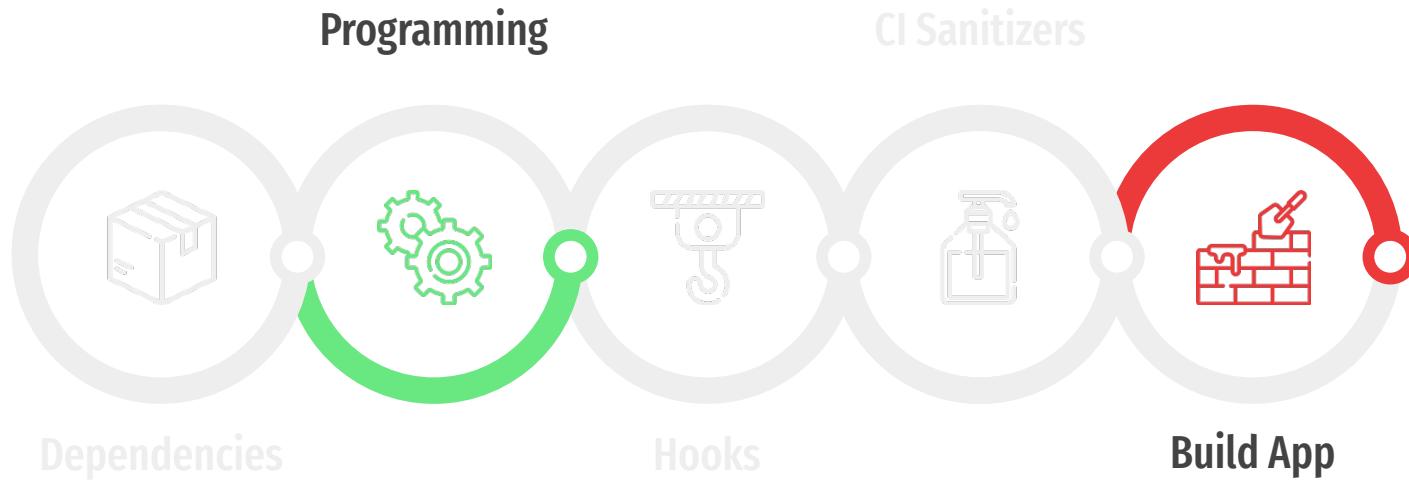
- ci cache
- test install flags
- check your deps

Choose your new PM

Bundler



Bundler



Parcel



Webpack

РЕАЛЬНО УДОБНО



РЕАЛЬНО ПОД СЕБЯ

Rollup



Vite

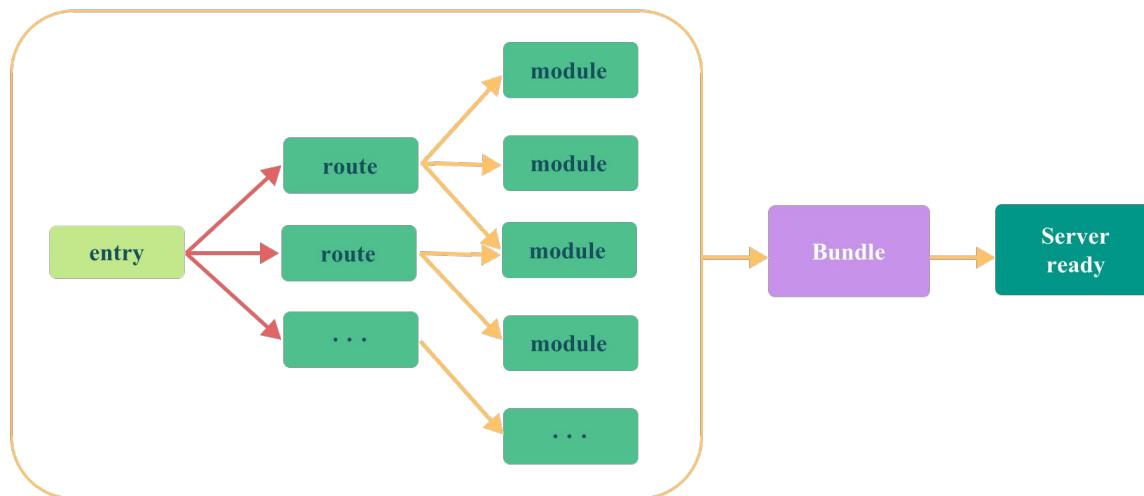
EXTREMELY BLAZING FAST



**ZERO CONFIGURATION WEB
APPLICATION BUNDLER**

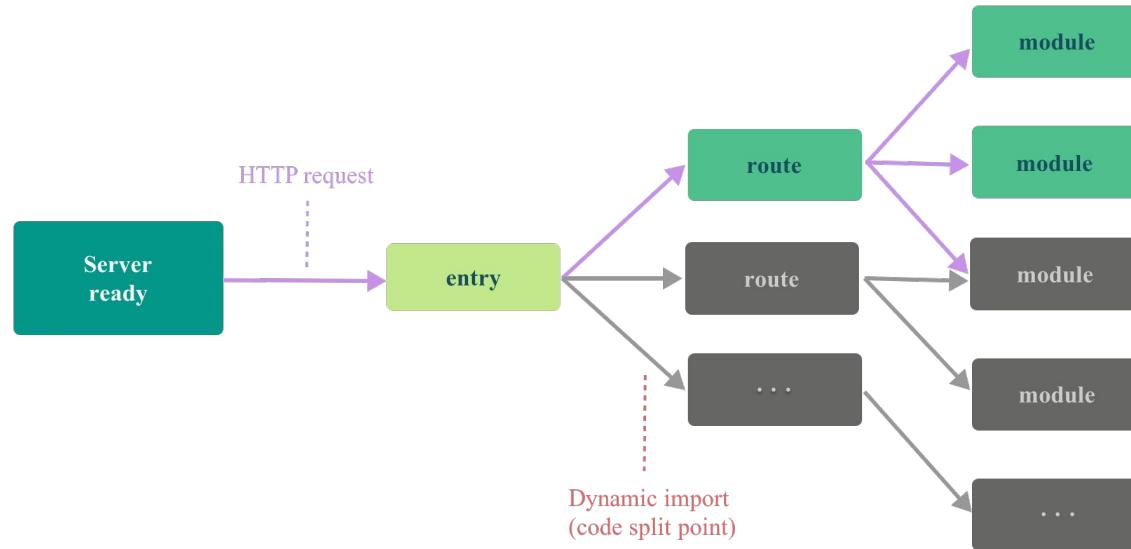
Vite

Bundle based dev server



Vite

Native ESM based dev server



Vite

Esbuild **10-100x faster** than JavaScript-based bundlers.

Native ESM === fast HRM

Vite + React

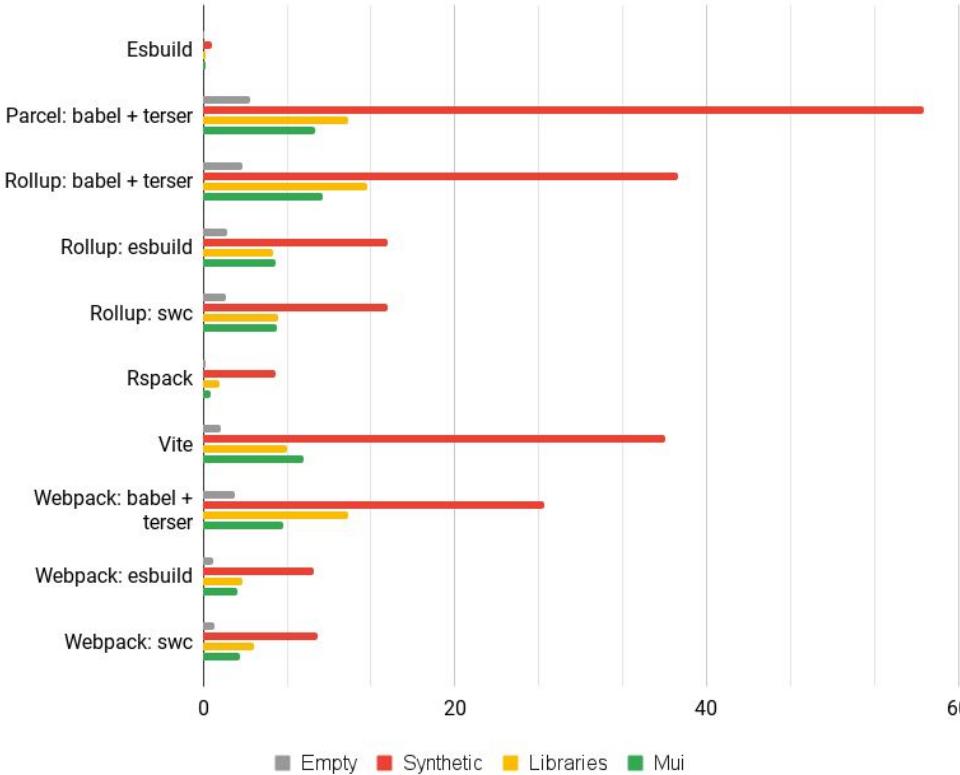
@vitejs/plugin-react(esbuild + babel)

Vite + React

@vitejs/plugin-react(esbuild + babel)

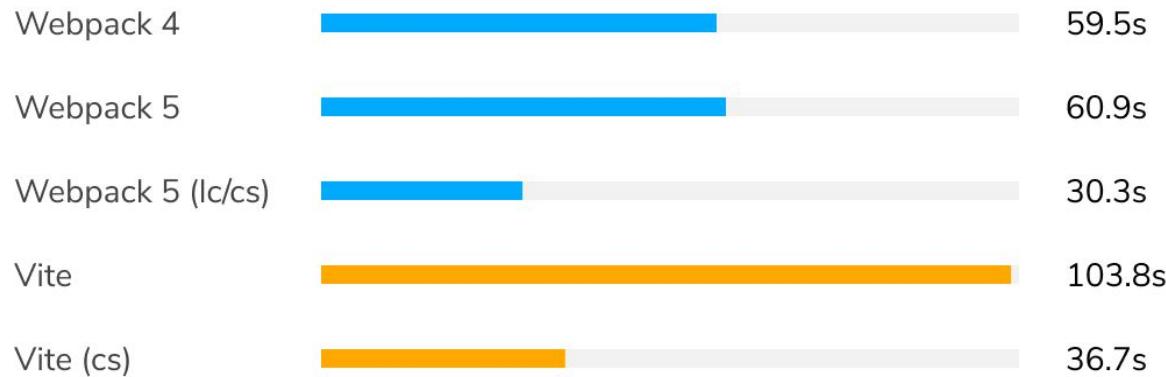
@vitejs/plugin-react-swc(SWC + esbuild)

Benchmarks

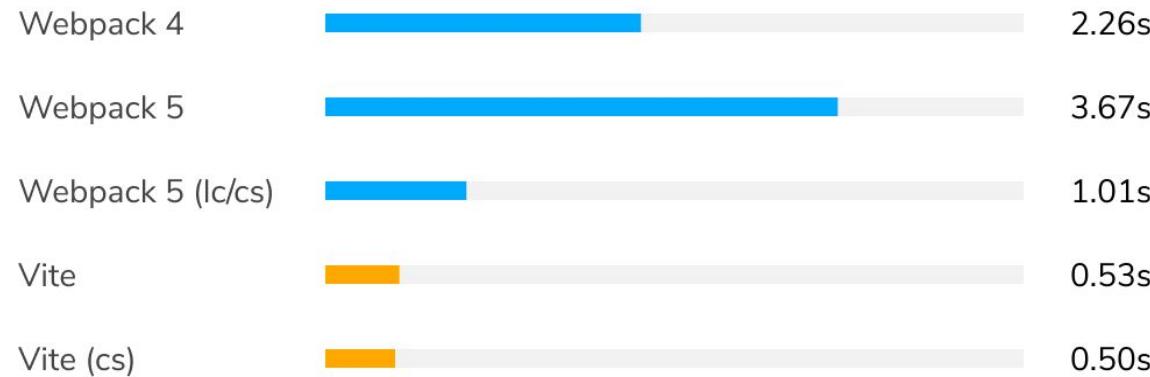


Development (cold start)

Time to first story dev start + story render



Rebuild time (Hot Module Reloading)



Production build



Bundlers

BABEL



ESBUILD ||

SWC



Sum Up

Bundlers

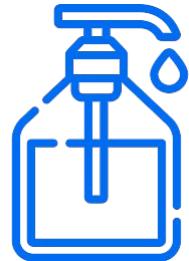
- avoid babel, use esbuild or SWC

Vite

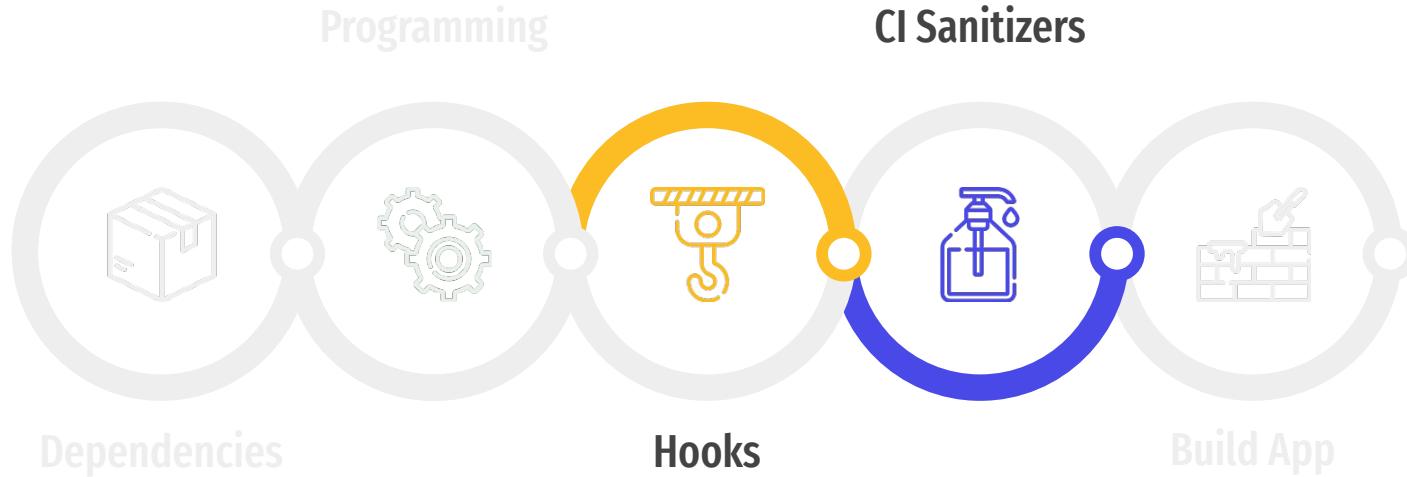
- good for small projects
- good dev experience

Hooks

And sanitizers



Shift Left



Hooks

- Linters
- Prettier
- Unit tests
- Security
- Unused/duped node_modules
- Unused files

Husky



**What'cha
doin'?**

Husky

- Can't run commands in parallel
- Have 30 direct dependencies
- Can't filter files

Husky +  💩 lint-staged

! concurrent mode is on

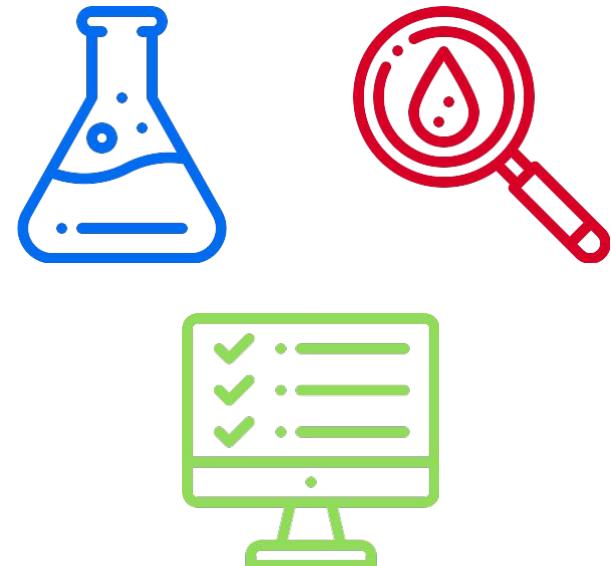
Husky + lint-staged

- In general works only with pre-commit hook
- Have 80 direct dependencies

Lefthook



Unit Test



Jest

| | |
|-----------------|--|
| pre-commit hook | |
| merge request | |
| custom | |

Jest

| | |
|-----------------|---------------|
| pre-commit hook | --onlyChanged |
| merge request | |
| custom | |

Jest

| | |
|-----------------|-----------------------|
| pre-commit hook | --onlyChanged |
| merge request | --changedSince=master |
| custom | |

Jest

| | |
|-----------------|-----------------------|
| pre-commit hook | --onlyChanged |
| merge request | --changedSince=master |
| custom | --findRelatedTests |

Jest parallelization

- `--runInBand -i` !

Jest parallelization

- `--runInBand -i` !
- `--maxWorkers=50%`

Jest parallelization

- `--runInBand -i` !
- `--maxWorkers=50%` [~20% Faster]

Jest parallelization

- `--runInBand -i` !
- `--maxWorkers=50%` [~20% Faster]
- `--workerThreads` !

babel-jest



@swc/jest

@swc/jest

```
module.exports = {
  transform: {
    "^.+\\.(t|j)sx?$$": "@swc/jest",
  },
};
```

jsdom

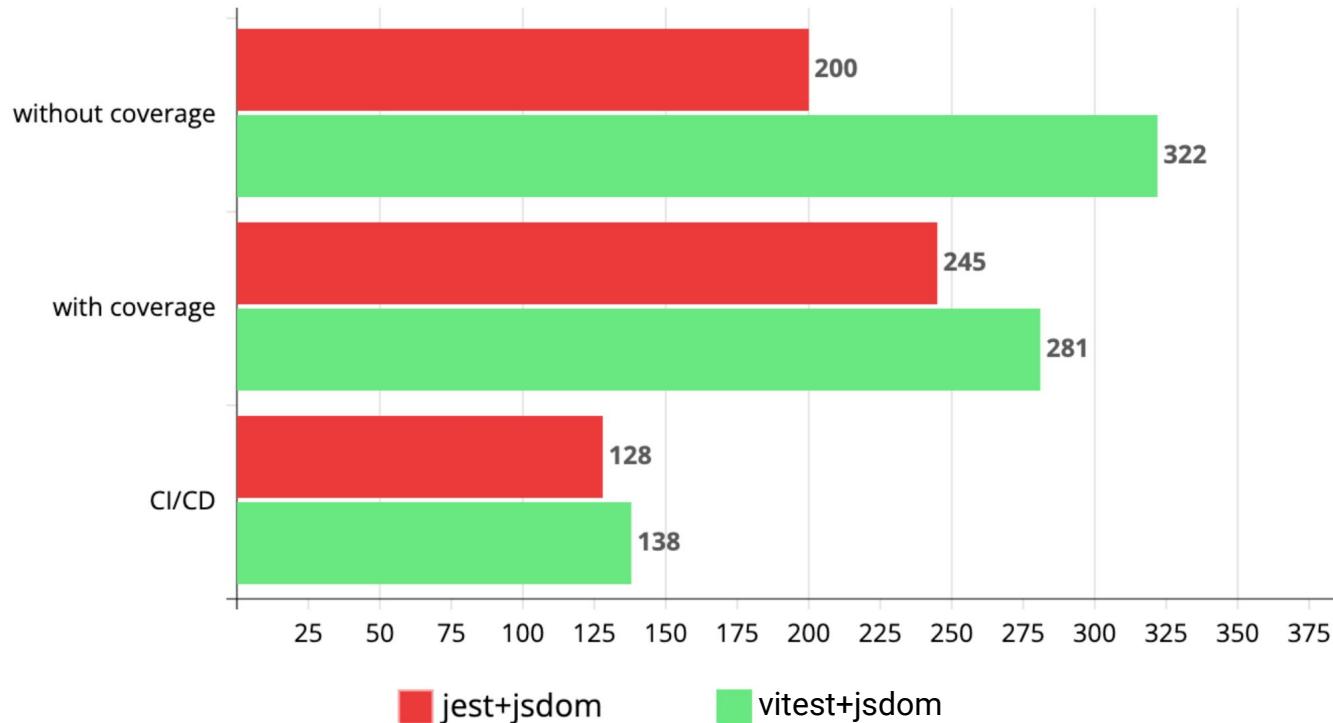


happy-dom

happy-dom

```
module.exports = {
  "testEnvironment": "@happy-dom/jest-environment"
};
```

vitest



Sum Up

Hooks

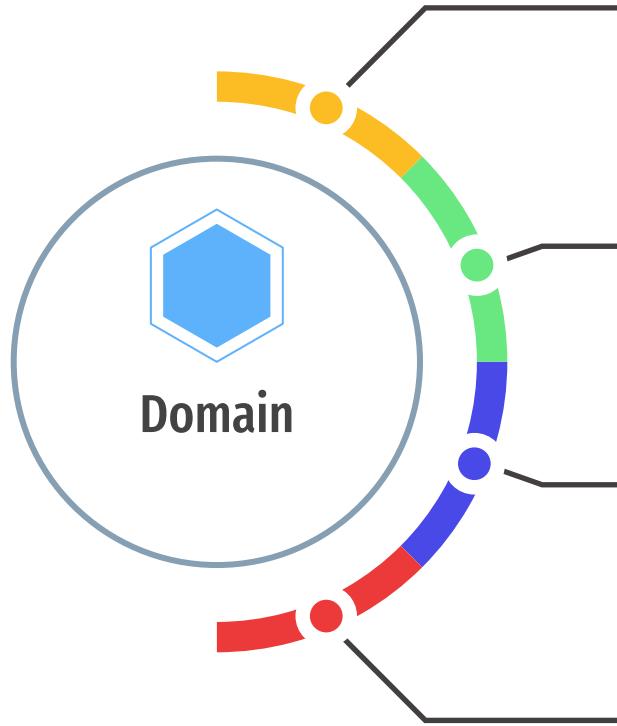
- run jobs in parallel
- split to different hooks
- or disable hooks

Tests

- do not run all tests
- use new tools for jest
- vitest is overrated

Architecture

And Design



Architecture

Modular

Architecture

Modular

- Reusability

Architecture

Modular

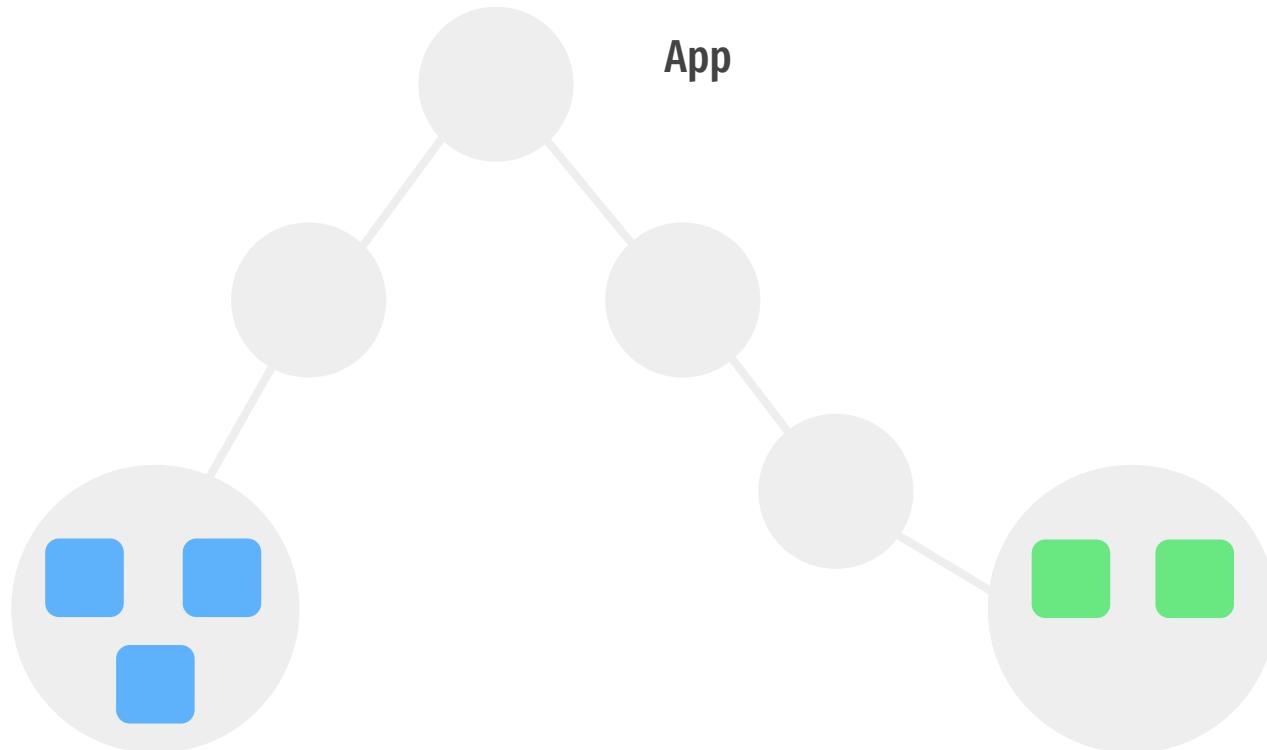
- Reusability
- Quality Assurance

Architecture

Modular

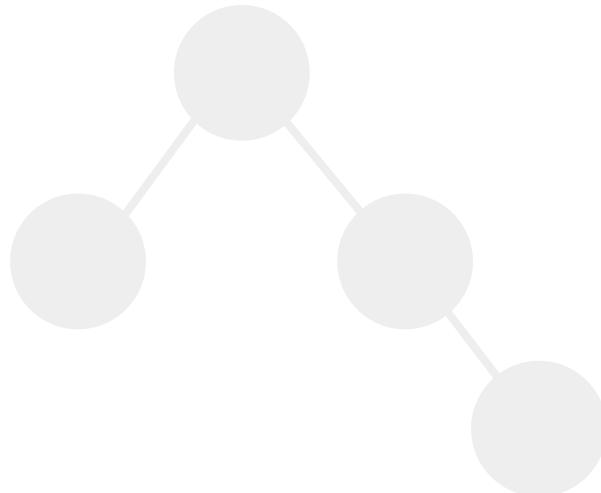
- Reusability
- Quality Assurance
- Scalability

Architecture



Architecture

App Rootable Components



Reusable Components



Libs & Data



Architecture

Module#1

own hooks

own tests

own build

Architecture

Module#1

own hooks

own tests

own build

Module#2

own hooks

own tests

own build

Architecture

Modular

- Reusability
- Quality Assurance
- Scalability

Communication

Architecture



Teammate #1 @teammate1 · 1 hour ago



So we can either: `renderToPipeableStream` immediately to early flush, and wrap the skeleton in Suspense, which means the page is blank without Javascript Await the network queries first to avoid the Suspense boundary, which gets the skeleton displayed without Javascript, but will delay early flush



6



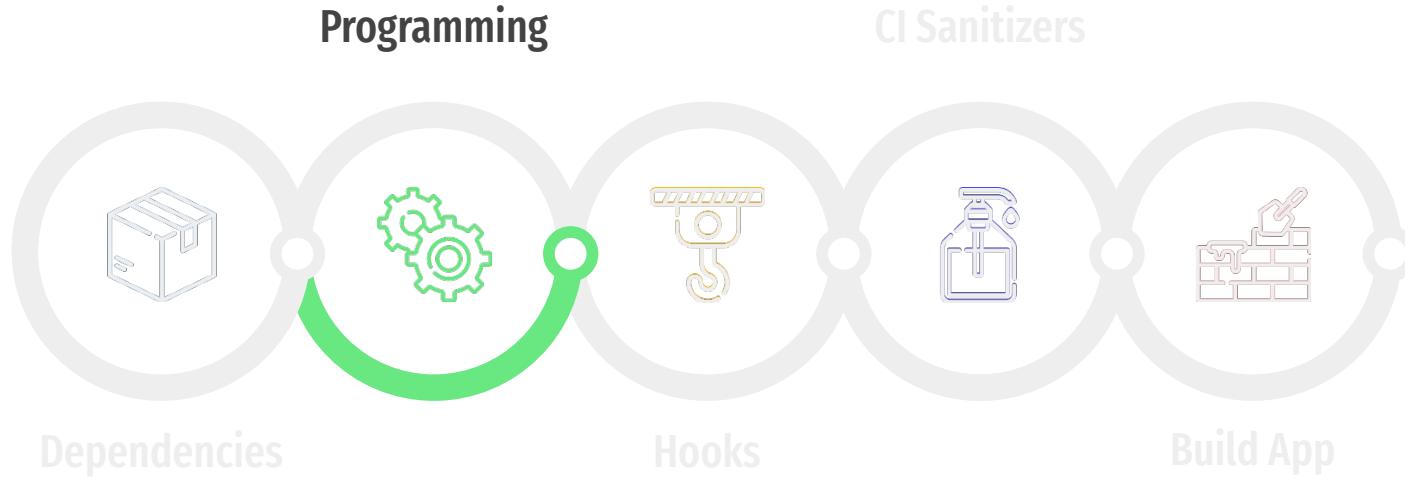
+ 2



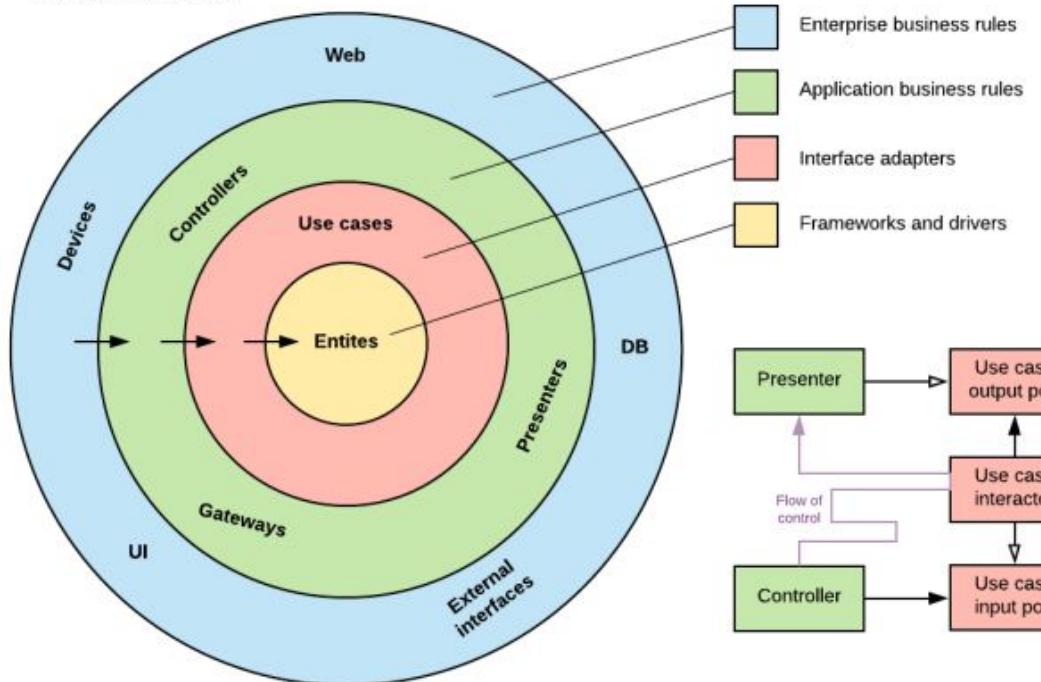
42 replies

Last reply by Teammate #3 10 minutes ago

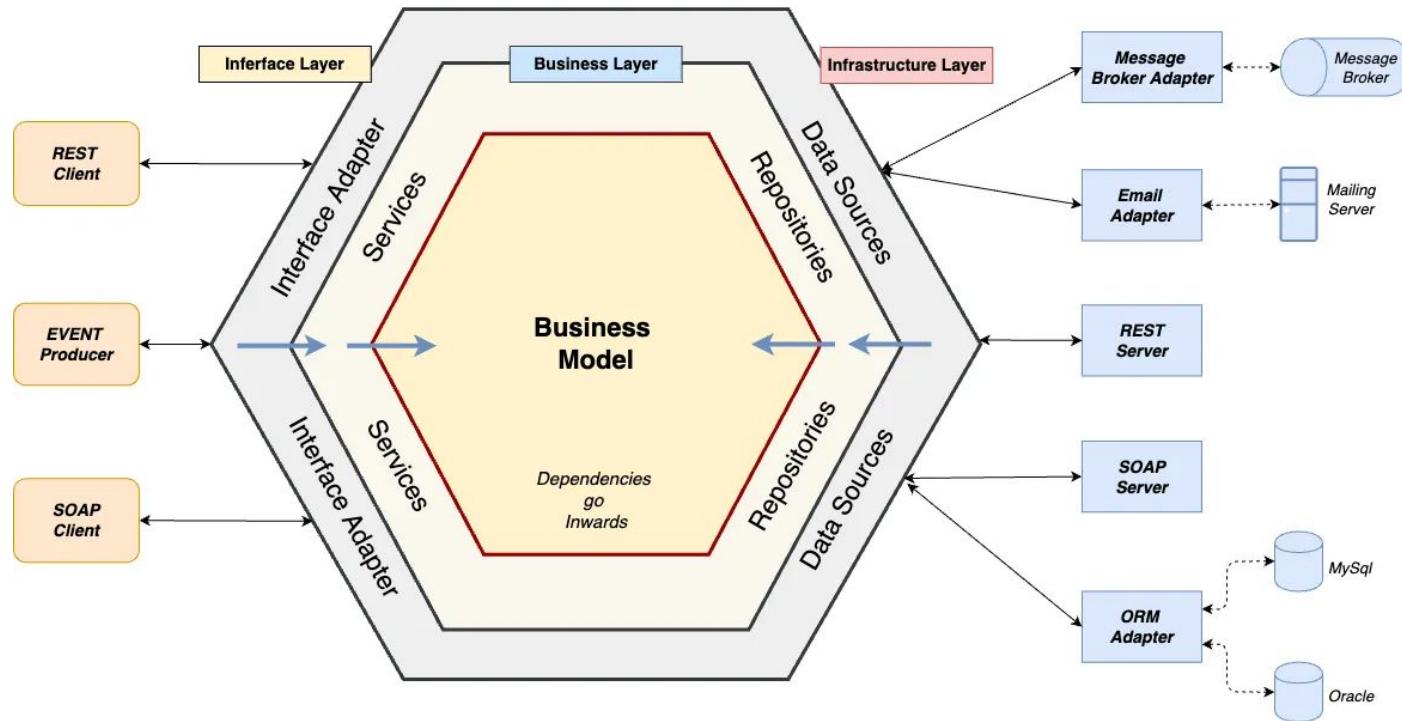
Architecture



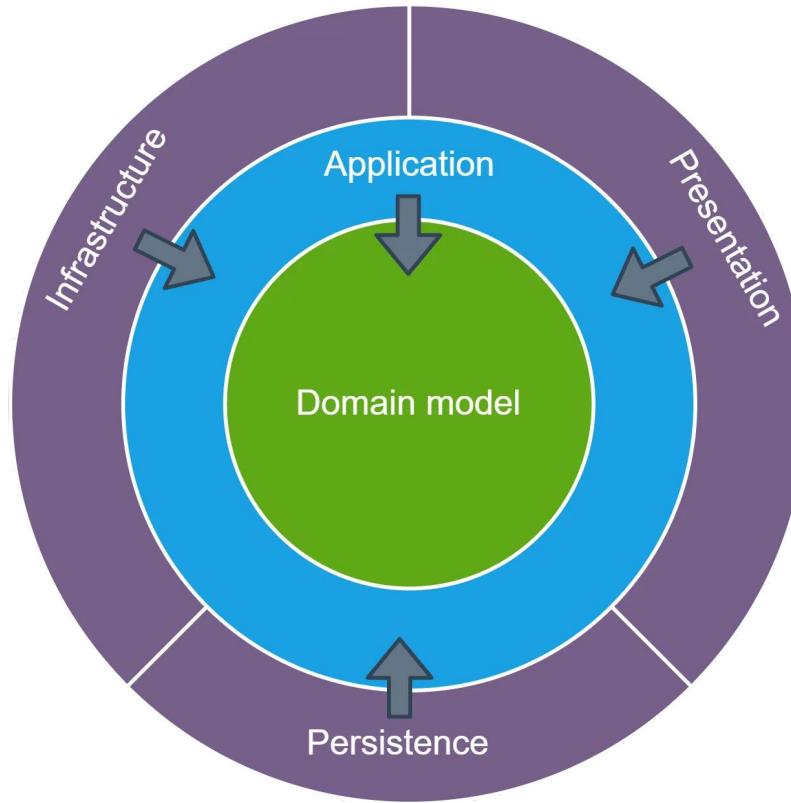
Clean Architecture



Hexagonal Architecture



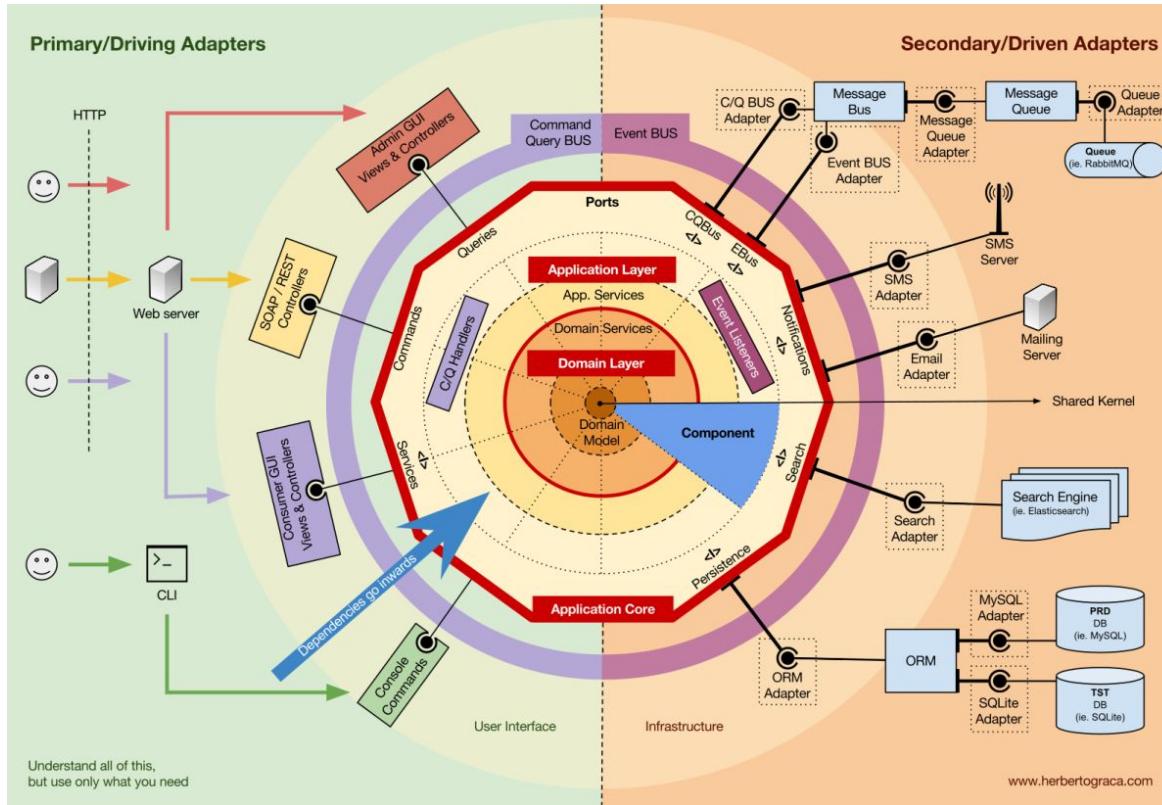
Onion Architecture



Everything Everywhere All at Once



Everything Everywhere All at Once



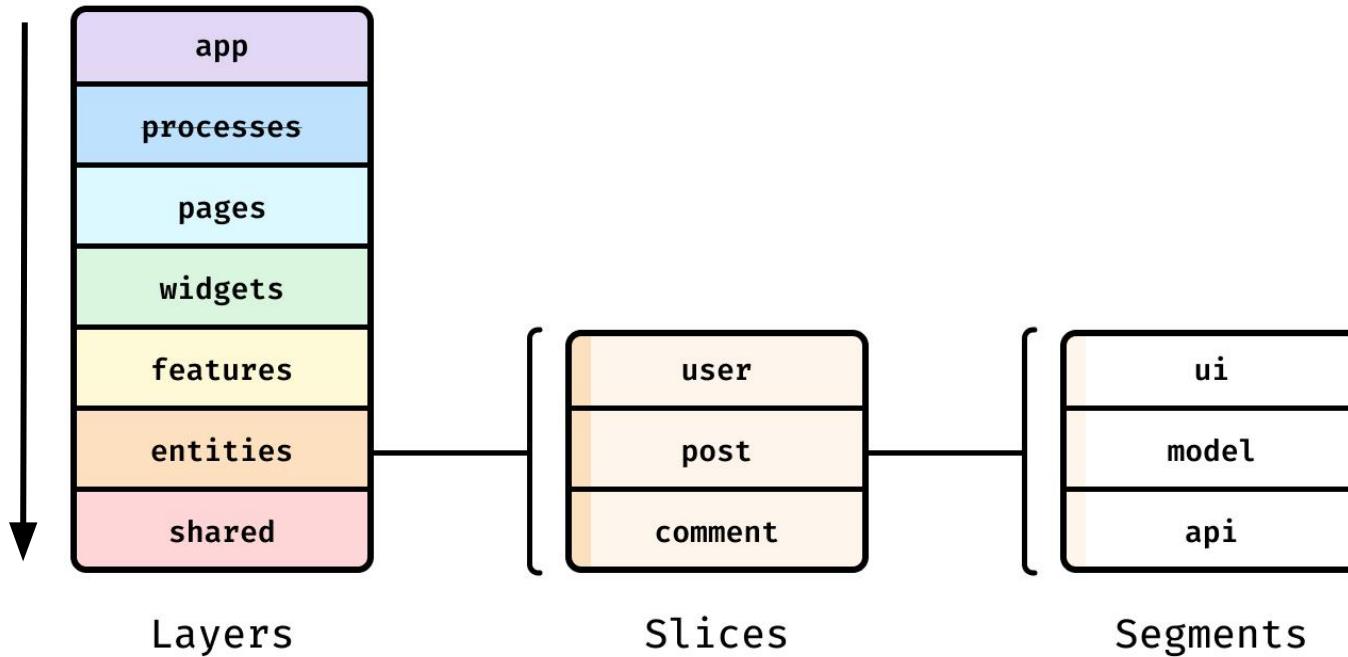
Architecture

Clean Architecture in Frontend 🤔

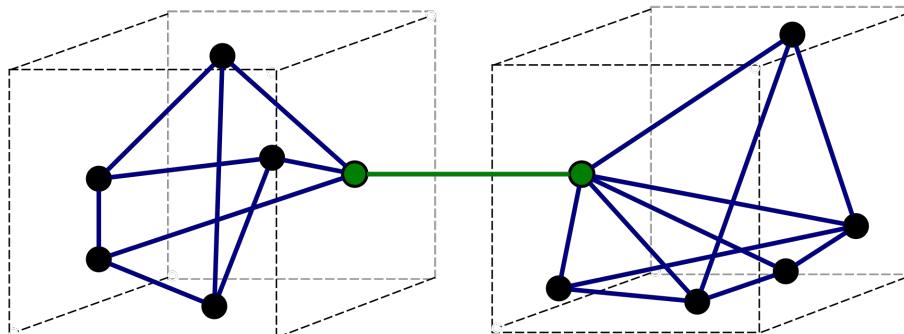
Architecture

- >  api
- >  assets
- >  components
- >  context
- >  features
- >  fragments
- >  hooks
- >  infrastructure
- >  libs
- >  pages
- >  server
- >  store

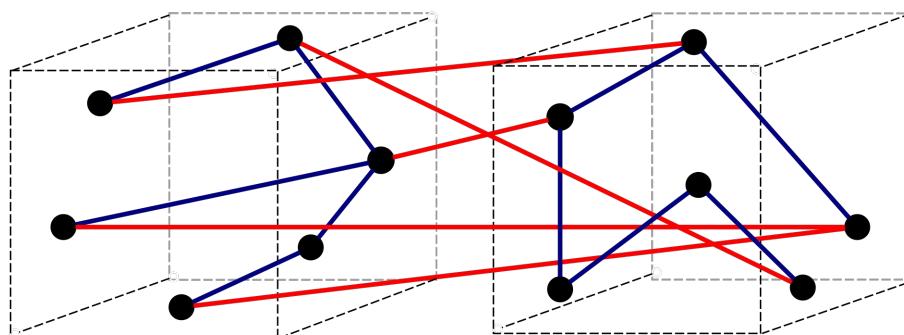
Feature-Sliced Design



Feature-Sliced Design



a) Good (loose coupling, high cohesion)



b) Bad (high coupling, low cohesion)

Recap

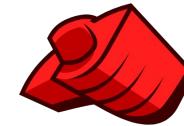
Set up your npm or switch to X



Switch to a new gen transpiler



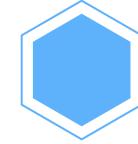
Run jobs in parallel



Do not run all tests



Do not create yet another architecture



Recap

Set up your npm or switch to X



Switch to a new gen transpiler



Run jobs in parallel



Do not run all tests



Do not create yet another architecture



Have a good one

