Bundle-analyzer and performance article outline/ notes

overview

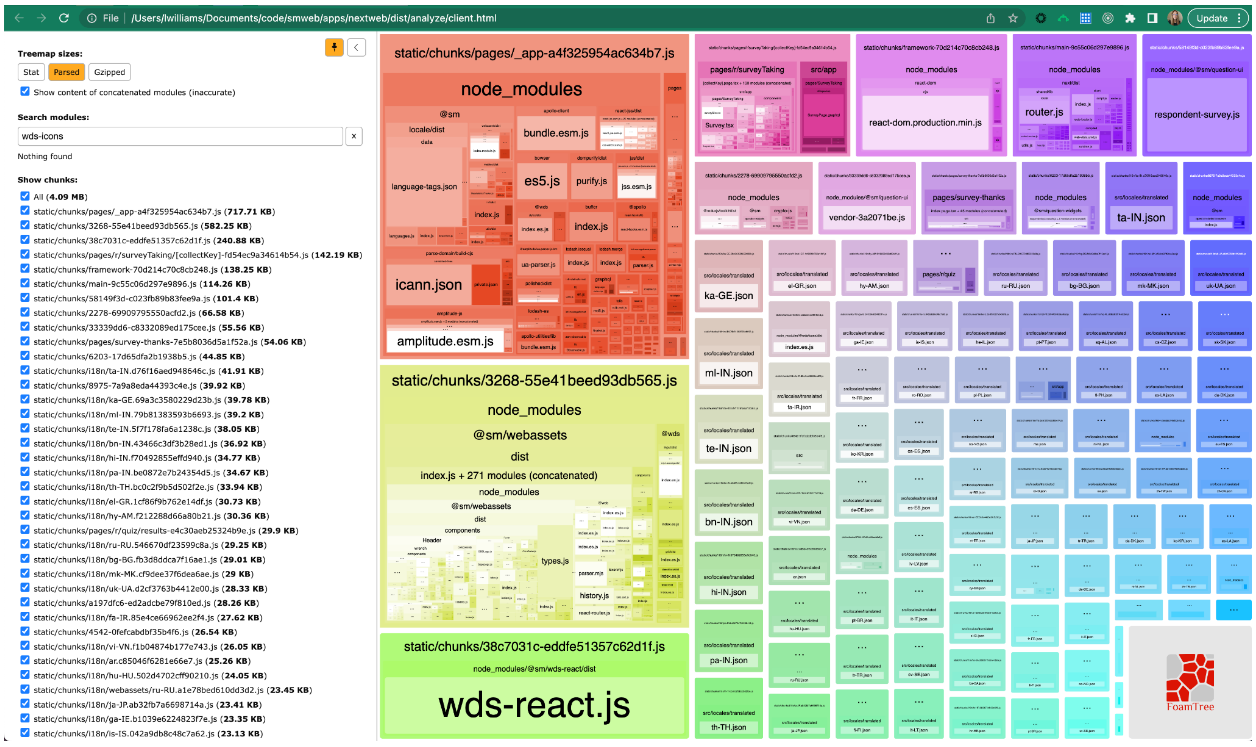
- Explain the importance of optimizing an application's performance
    - larger bundle size can significantly increase page load times
- Introduce the concept of bundle size and its impact on page load times
    - The size of your js that needs to be loaded in the browser, the larger the bundle the longer your page will take to load
    - Chunking
    - Code splitting
    - Tree shaking
- Problems / Pitfalls. what are we looking to fix. what can we improve.
    - Duplicates (2 versions of same package)
    - Unused stuff (not tree shaken)
- Solutions
- Tools and approaches. how to improve

Intro:

- Explain the importance of optimizing an application's performance
    - larger bundle size can significantly increase page load times
- Introduce the concept of bundle size and its impact on page load times
    - The size of your js that needs to be loaded in the browser, the larger the bundle the longer your page will take to load
    - Chunking

- Briefly introduce the tools that will be used in the tutorial
    - Brief intro about webpack and its role
        - Webpack is a JS module bundler that complies, transforms and packages your application's source code, along with its dependencies, into output files that can be loaded in your browser
        - https://webpack.js.org/concepts/
        - vite? alternatives to webpack.
        - this still applies to next. this means almost 100% of cases
    - How it knows how to bundle things (imports, exports)
        - show the output, without uglify. u can actually see the bundle objects. you could color coat.
    - Webpack Bundle Analyzer(https://www.npmjs.com/package/webpack-bundle-analyzer)
    - This will generate a visualization of your application's bundle
    - Webpack Bundle Analyzer is a powerful tool that provides you with a visual representation of your JS bundle size, making it easier to identify areas for optimization

    - It can show your larger dependencies, duplicated modules and unused modules that may be able to be removed to reduce your bundle size
    - we load some subset of the chunks. not sure how this is relevant



Treemap sizes:

Stat [✓ Parsed] Gzipped

- [✓] Show content of concatenated modules (inaccurate)

Search modules:

wds-icons

Nothing found

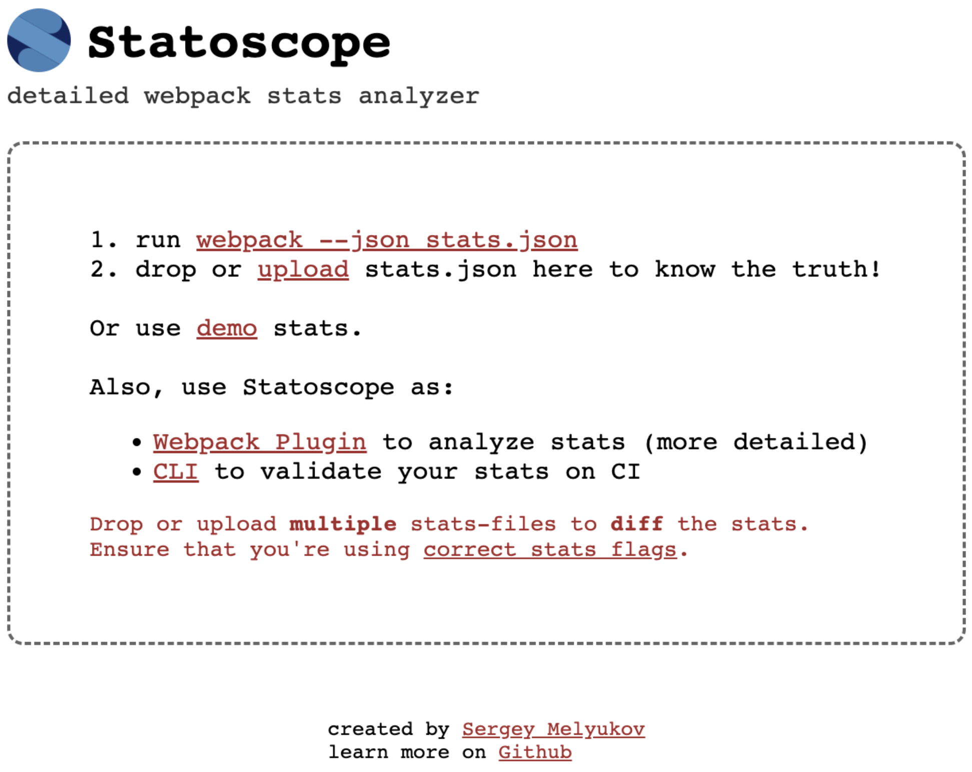
Show chunks:

- [✓] All (**4.09 MB**)
- [✓] static/chunks/pages/_app-a4f325954ac634b7.js (**717.71 KB**)
- [✓] static/chunks/3268-55e41beed93db565.js (**582.25 KB**)
- [✓] static/chunks/38c7031c-eddfe51357c62d1f.js (**240.88 KB**)
- [✓] static/chunks/pages/r/surveyTaking/[collectKey]-fd54ec9a34614b54.js (**142.19 KB**)
- [✓] static/chunks/framework-70d214c70c8cb248.js (**138.25 KB**)
- [✓] static/chunks/main-9c55c06d297e9896.js (**114.26 KB**)
- [✓] static/chunks/58149f3d-c023fb89b83fee9a.js (**101.4 KB**)
- [✓] static/chunks/2278-69909795550acfd2.js (**66.58 KB**)
- [✓] static/chunks/33339dd6-c8332089ed175cee.js (**55.56 KB**)
- [✓] static/chunks/pages/survey-thanks-7e5b8036d5a1f52a.js (**54.06 KB**)
- [✓] static/chunks/6203-17d65dfa2b1938b5.js (**44.85 KB**)
- [✓] static/chunks/i18n/ta-IN.d76f16aed948646c.js (**41.91 KB**)
- [✓] static/chunks/8975-7a9a8eda44393c4e.js (**39.92 KB**)
- [✓] static/chunks/i18n/ka-GE.69a3c3580229d23b.js (**39.78 KB**)
- [✓] static/chunks/i18n/ml-IN.79b81383593b6693.js (**39.2 KB**)
- [✓] static/chunks/i18n/te-IN.5f7f178fa6a1238c.js (**38.05 KB**)
- [✓] static/chunks/i18n/bn-IN.43466c3df3b28ed1.js (**36.92 KB**)
- [✓] static/chunks/i18n/hi-IN.f70492855effd940.js (**34.77 KB**)
- [✓] static/chunks/i18n/pa-IN.be0872e7b24354d5.js (**34.67 KB**)
- [✓] static/chunks/i18n/th-TH.bc0c2f9b5d502f2e.js (**33.94 KB**)
- [✓] static/chunks/i18n/el-GR.1cf86f9b762e14df.js (**30.73 KB**)
- [✓] static/chunks/i18n/hy-AM.f212288d66a80b21.js (**30.36 KB**)
- [✓] static/chunks/pages/r/quiz/results-e4c30aeb25324b9e.js (**29.9 KB**)
- [✓] static/chunks/i18n/ru-RU.546670df23599c8a.js (**29.25 KB**)
- [✓] static/chunks/i18n/bg-BG.fb3d8ddca7f16ae1.js (**29.01 KB**)
- [✓] static/chunks/i18n/mk-MK.cf9dee37f6dea6ae.js (**29 KB**)
- [✓] static/chunks/i18n/uk-UA.d2cf3763b4412e00.js (**28.33 KB**)
- [✓] static/chunks/a197dfc6-ed2adcbe79f810ed.js (**28.26 KB**)
- [✓] static/chunks/i18n/fa-IR.85e4ce66962ee2f4.js (**27.62 KB**)
- [✓] static/chunks/4542-0fefcabdbf35b4f6.js (**26.54 KB**)
- [✓] static/chunks/i18n/vi-VN.f1b04874b177e743.js (**26.05 KB**)
- [✓] static/chunks/i18n/ar.c85046f6281e66e7.js (**25.26 KB**)
- [✓] static/chunks/i18n/hu-HU.502d4702cff90210.js (**24.05 KB**)
- [✓] static/chunks/i18n/webassets/ru-RU.a1e78bed610dd3d2.js (**23.45 KB**)
- [✓] static/chunks/i18n/ja-JP.ab32fb7a6698714a.js (**23.41 KB**)
- [✓] static/chunks/i18n/ga-IE.b1039e6224823f7e.js (**23.35 KB**)
- [✓] static/chunks/i18n/is-IS.042a9db8c48c7a62.js (**23.13 KB**)

- Install the Webpack Bundle Analyzer package in your project
    - Add some code snippets to show this

- Interpreting the results of the visualization
    - An interactive treemap is generated and this represents the size and structure of your JS bundle
    - Organized into nested rectangles which each represent a module or dependency
    - The size of each rectangle corresponds to the overall size of the module in the bundle, larger rectangle = larger modules
    - The hierarchy and grouping helps you understand the overall structure of your project and how modules are related to each other
    - Hovering over a module will show a tooltip that gives you more info like size, name and path for each module
    - Using the search feature on the left side allows you to search for specific modules and can be useful, especially when trying to see if I module has been successfully removed from your project

- Statoscope (https://statoscope.tech/)



Statoscope

detailed webpack stats analyzer

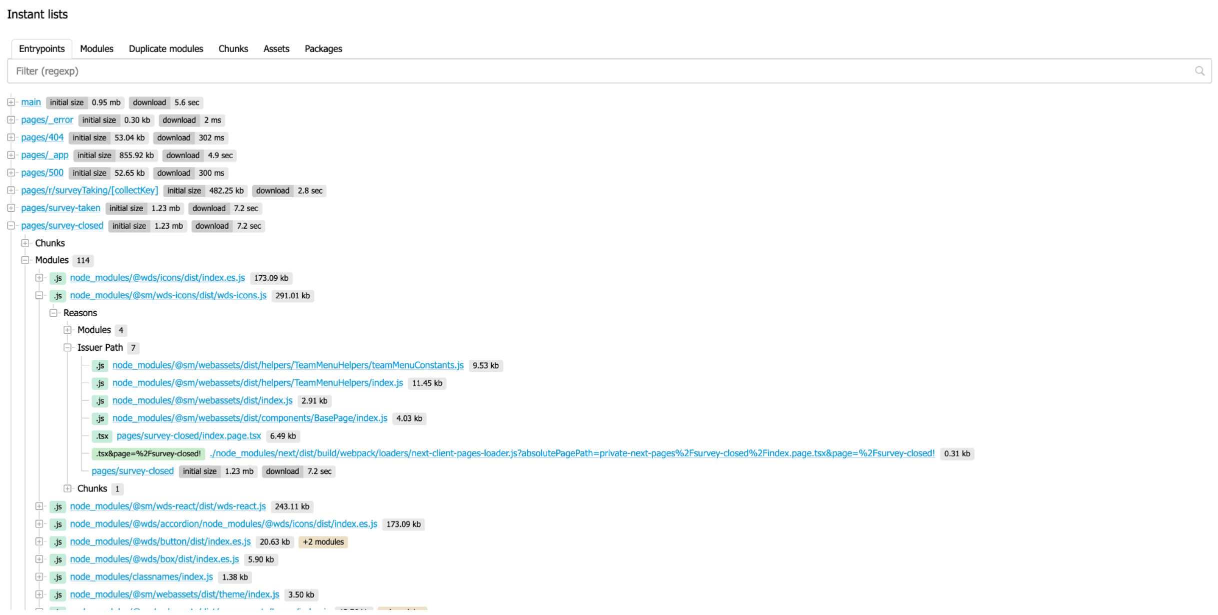
```
1. run webpack --json stats.json
2. drop or upload stats.json here to know the truth!

Or use demo stats.

Also, use Statoscope as:

  • Webpack Plugin to analyze stats (more detailed)
  • CLI to validate your stats on CI

Drop or upload multiple stats-files to diff the stats.
Ensure that you're using correct stats flags.
```

created by Sergey Melyukov
learn more on Github

- Once you have used Webpack to create a stats.json file, you can drop that file into the Statoscope tool and it will analyze it for you and create a report
- Where the webpack bundle analyzer helps you identify which modules may be large, duplicated or unused, it doesn't give as clear a picture as to where (why?) these are being used (included?)
- Statoscope can help in finding where certain modules are used (imported?) to help you safely remove them
- Statoscope is a webpack stats analyzer
    - You can examine details about your projects individual modules, including their size, dependencies, and location within the codebase (screenshot or something, i dont fully understand what you mean by "location")
    - This report gives you a lot of information and insight as to the size of bundles, chunks and assets in your codebase
    - The section I found the most useful was the Entrypoints -> Modules -> clicking into the module I was trying to remove, looking at the Reasons -> Issuer Path
    - The Reasons can include explicit import statements in your code, dynamic imports, or dependencies requires by another module
    - The Issuer Path shows the chain of modules that led to the inclusion of a specific module, it's essentially a list of parent modules that required or imported the module we're looking for

    - It helps you trace back to the root entry that caused the inclusion of the specified module

- Below is an example of the output from Statoscope where you can transverse through the modules and where they are loaded



```
Instant lists

Entrypoints  Modules  Duplicate modules  Chunks  Assets  Packages
Filter (regexp)

⊞ main  initial size 0.95 mb  download 5.6 sec
⊞ pages/_error  initial size 0.30 kb  download 2 ms
⊞ pages/404  initial size 53.04 kb  download 302 ms
⊞ pages/_app  initial size 855.92 kb  download 4.9 sec
⊞ pages/500  initial size 52.65 kb  download 300 ms
⊞ pages/r/surveyTaking/[collectKey]  initial size 482.25 kb  download 2.8 sec
⊞ pages/survey-taken  initial size 1.23 mb  download 7.2 sec
⊟ pages/survey-closed  initial size 1.23 mb  download 7.2 sec
    ⊞ Chunks
    ⊟ Modules 114
        ⊞ .js node_modules/@wds/icons/dist/index.es.js  173.09 kb
        ⊟ .js node_modules/@sm/wds-icons/dist/wds-icons.js  291.01 kb
            ⊟ Reasons
                ⊞ Modules 4
                ⊟ Issuer Path 7
                    .js node_modules/@sm/webassets/dist/helpers/TeamMenuHelpers/teamMenuConstants.js  9.53 kb
                    .js node_modules/@sm/webassets/dist/helpers/TeamMenuHelpers/index.js  11.45 kb
                    .js node_modules/@sm/webassets/dist/index.js  2.91 kb
                    .js node_modules/@sm/webassets/dist/components/BasePage/index.js  4.03 kb
                    .tsx pages/survey-closed/index.page.tsx  6.49 kb
                    .tsx&page=%2Fsurvey-closed! ./node_modules/next/dist/build/webpack/loaders/next-client-pages-loader.js?absolutePagePath=private-next-pages%2Fsurvey-closed%2Findex.page.tsx&page=%2Fsurvey-closed!  0.31 kb
                    pages/survey-closed  initial size 1.23 mb  download 7.2 sec
                ⊞ Chunks 1
        ⊞ .js node_modules/@sm/wds-react/dist/wds-react.js  243.11 kb
        ⊞ .js node_modules/@wds/accordion/node_modules/@wds/icons/dist/index.es.js  173.09 kb
        ⊞ .js node_modules/@wds/button/dist/index.es.js  20.63 kb  +2 modules
        ⊞ .js node_modules/@wds/box/dist/index.es.js  5.90 kb
        ⊞ .js node_modules/classnames/index.js  1.38 kb
        ⊞ .js node_modules/@sm/webassets/dist/theme/index.js  3.50 kb
```

- Interpreting the report to see where your modules are being loaded
    - Removing the desired modules from the pages it's loaded on
- Re-running webpack-bundle-analyzer and Statoscope to verify these changes and improvements
    - in a video this would be straight forward. i would like to know the process and thinking that you do to arrive at "hypotheses" / "things to try" and then verify. obviously there are many ways, just walk us through your method.
- SVGator (honourable mention?)
    - icons tend to be one of the heaviest dependency packages. you dont use that many of them
    - additional visualizations for presentation?

- Summarize the benefits of analyzing and optimizing loaded modules

- Encourage developers to continuously monitor and optimize their applications
    - automated tools for this? part of pipeline. you have to explicitly increase size restrictions if you need it. like a "debt ceiling"