



DrupalCamp Spain

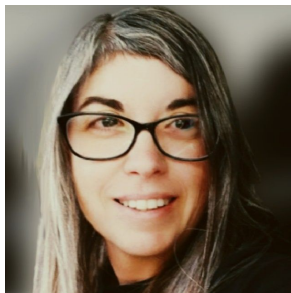
Benidorm 2024

Del 24 al 26 de octubre

En busca de una mejor frontend developer experience (FDX): Automatización de tareas en el front antes y después de Drupal 10

Yolanda Górriz





Yolanda Górriz Expósito (@ygoex) 🙌

- UX/UI Designer & Frontend Developer desde 2014
- Working with Drupal since 2018 @ University of Edinburgh and Cambrico
- Currently on a break/sabbatical → 📖 🧘 ⏳ 🌱



<https://goex.dev>



<https://www.drupal.org/u/ygoex>



<https://github.com/ygoex>



<https://www.linkedin.com/in/ygoex/>



<https://bsky.app/profile/ygoex.bsky.social>



1. Common tools used in Drupal 8/9 (Gulp, Webpack, Laravel Mix)
2. Web Standards Evolution & Drupal 9/10 Frontend Enhancements
3. Status of frontend tooling in 2024
4. Traditional frontend tools in use: Webpack
5. New generation frontend tools in use: Vite
6. New generation frontend tools in use: ESBuild
7. Comparing Vite, Webpack, and ESBuild
8. Conclusion and Q&A



1. Common tools used in Drupal 8/9 (Gulp, Webpack, Laravel Mix)



Common tools used in Drupal 8/9 (Gulp, Webpack, Laravel Mix)



webpack





2. Web Standards Evolution & Drupal 9/10 Frontend Enhancements





Drupal 9



Drupal 10

- [Webp support in 9.2](#)
- [Lazyload images by default](#)

- [JS minification on the fly](#)



Web Standards Evolution & Drupal 9/10 Frontend Enhancements



Drupal 9



Drupal 10

- [Webp support in 9.2](#)
- [Lazyload images by default](#)

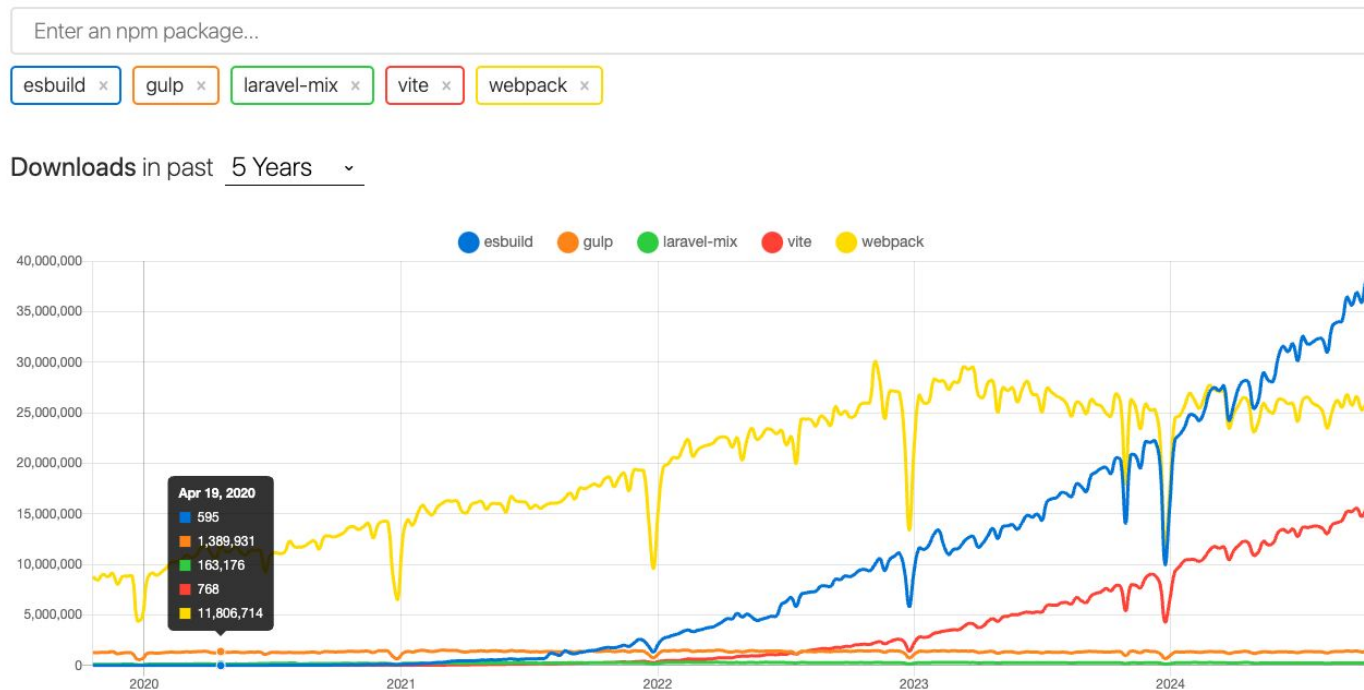
- [JS minification on the fly](#)



3. Status of frontend tooling in 2024




























esbuild vs gulp vs laravel-mix vs vite vs webpack





Frontend tooling in 2024

Stats

			Stars	Issues	Version	Updated [?]	Created [?]	Size
	esbuild	  	38,090	484	0.24.0	a month ago	7 years ago	install size 
	gulp	  	33,007	29	5.0.0	7 months ago	11 years ago	install size 
	laravel-mix	  	5,269	269	6.0.49	2 years ago	8 years ago	install size 
	vite	  	68,068	631	5.4.9	8 days ago	5 years ago	install size 
	webpack	  	64,643	259	5.95.0	a month ago	13 years ago	install size 



4. Traditional frontend tools in use: Webpack





Traditional frontend tools in use: Webpack

JS bundle 

Multiple entry points 

SCSS bundle 



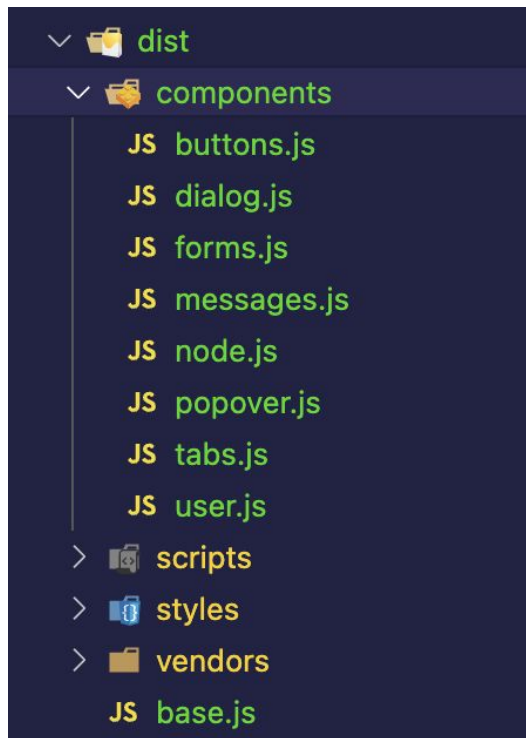
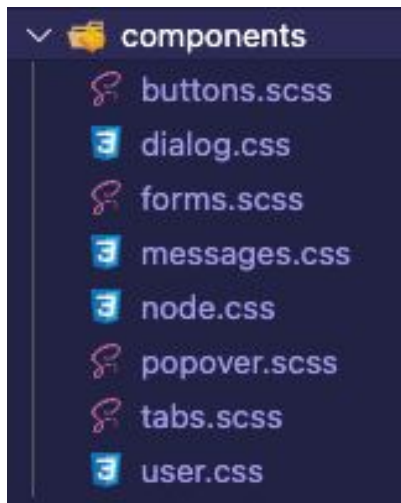
```
import '../styles/base.scss'  
import '../styles/components/buttons.scss'
```





Traditional frontend tools in use: Webpack

SCSS Bundle





Traditional frontend tools in use: Webpack

webpack empty js files scss compilation

Stack Overflow
<https://stackoverflow.com> › web... › Traducir esta página

Webpack 5: Empty JS chunks are emitted for CSS entries

21 ene 2022 — Webpack only generates JS, no CSS file from SCSS · 1. Can't create CSS from SASS using Webpack · 0. Webpack2 compiles sass/scss into js instead ...

Webpack production builds css as empty file - Stack Overflow 19 jun 2018

webpack-dev-server not recompiling (JS files AND SCSS files) 9 dic 2019

Not to generate separate empty JS file when have several css ... 9 mar 2017

Avoid downloading empty JavaScript file when using ... 5 mar 2022

Más resultados de stackoverflow.com

NPM
<https://www.npmjs.com> › package › Traducir esta página

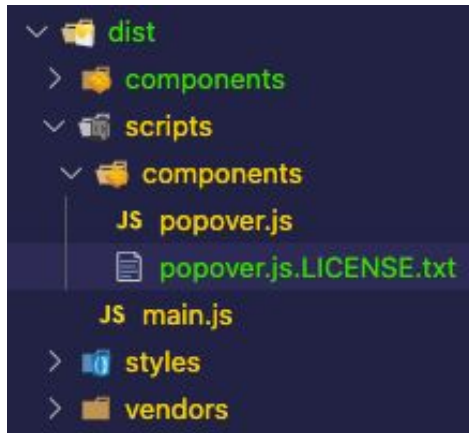
webpack-remove-empty-scripts

28 ago 2023 — This plugin removes generated empty JS files. Warning. This plugin is the Emergency Crutch for the mini-css-extract-plugin issue. The mini- ...

<https://github.com/webpack-contrib/extract-text-webpack-plugin/issues/518>



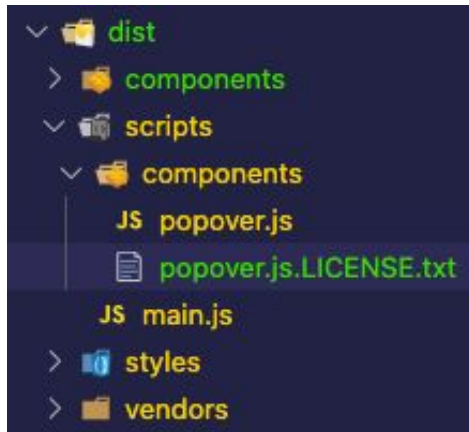
Traditional frontend tools in use: Webpack



```
optimization: {
  minimize: true, // Enable minimization
  minimizer: [
    new TerserPlugin({
      terserOptions: {
        format: {
          comments: false, // Remove comments from the output
        },
      },
      extractComments: false, // Don't extract comments into a separate file
    }),
  ],
},
```




Traditional frontend tools in use: Webpack



```
optimization: {  
  minimize: true, // Enable minimization  
  minimizer: [  
    new TerserPlugin({  
      terserOptions: {  
        format: {  
          comments: false, // Remove comments from the output  
        },  
      },  
      extractComments: false, // Don't extract comments into a separate file  
    }  
  ],  
},
```



5. New generation frontend tools in use: Vite





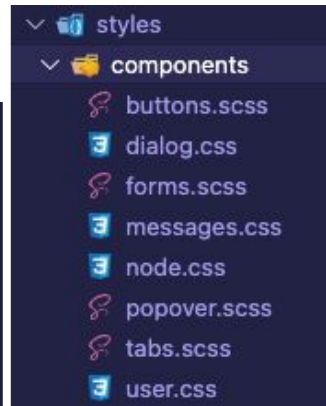
Traditional frontend tools in use: Vite

JS bundle 

Multiple entry points 

SCSS bundle 

```
dist/styles/node.css      0.05 kB  gzip:
dist/styles/user.css      0.95 kB  gzip:
dist/styles/dialog.css    0.97 kB  gzip:
dist/styles/messages.css  1.21 kB  gzip:
dist/styles/tabs.css      3.48 kB  gzip:
dist/styles/popover.css   6.11 kB  gzip:
dist/styles/buttons.css   14.39 kB gzip:
dist/styles/forms.css     22.85 kB gzip: 3.68 kB
dist/styles/base.css      244.82 kB gzip: 31.64 kB
dist/scripts/main.js      0.08 kB  gzip: 0.10 kB
dist/scripts/components/popover.js 81.83 kB gzip: 24.67 kB
[vite-plugin-static-copy] Copied 2 items.
✓ built in 3.15s
```





Traditional frontend tools in use: Vite

It doesn't manage well assets placed in subdirectories ⚠

<https://github.com/vitejs/vite/issues/11012>

dist/styles/components/node2.css	0.05 kB	gzip: 0.07 kB
dist/styles/components/user2.css	0.95 kB	gzip: 0.38 kB
dist/styles/components/dialog2.css	0.97 kB	gzip: 0.41 kB
dist/styles/components/messages2.css	1.21 kB	gzip: 0.44 kB
dist/styles/components/tabs.css	3.48 kB	gzip: 0.88 kB
dist/styles/components/popover.css	6.11 kB	gzip: 1.03 kB
dist/styles/components/buttons.css	14.39 kB	gzip: 1.97 kB
dist/styles/components/forms.css	22.85 kB	gzip: 3.68 kB
dist/styles/base.css	244.82 kB	gzip: 31.64 kB
dist/scripts/main.js	0.08 kB	gzip: 0.10 kB
dist/scripts/components/popover.js	81.83 kB	gzip: 24.67 kB

[vite-plugin-static-copy] Copied 6 items.
✓ built in 3.06s



6. New generation frontend tools in use: ESBuild






Traditional frontend tools in use: ESBuild

Config structure is very similar to Gulp (organised in independent tasks)

JS bundle 

Multiple entry points 

SCSS compilation with subdirectories 

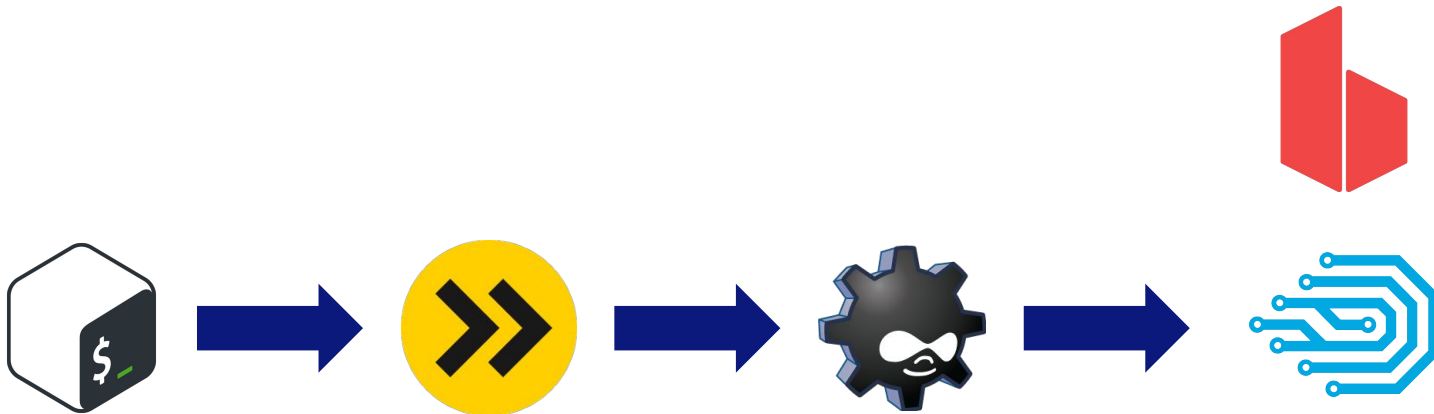
```
// Build function to run all tasks in sequence
const runBuildTasks = async () => {
  console.time('Total Build Time');
  await clean();
  await buildStyles();
  await buildScripts();
  await copyVendors();
  console.timeEnd('Total Build Time');
};

// Run the build based on command line argument
const run = async () => {
  const args = process.argv.slice(2);
  if (args.includes('-css')) {
    await buildStyles();
  } else if (args.includes('-js')) {
    await buildScripts();
  } else if (args.includes('-copy')) {
    await copyVendors();
  } else {
    await runBuildTasks();
  }
};
```



Traditional frontend tools in use: ESBuild

Why do we need this level of granularity?





Traditional frontend tools in use: ESBuild

```
# Function to watch CSS
watch_css() {
  echo "Watching CSS files..."
  chokidar 'src/styles/**/*.scss,css' -c 'npm run esbuild-css && npm run drush-cr && browser-sync reload'
}

# Function to watch JS
watch_js() {
  echo "Watching JS files..."
  chokidar 'src/scripts/**/*.js' -c 'npm run esbuild-js && npm run drush-cr && browser-sync reload'
}

# Function to watch both CSS and JS
watch_all() {
  echo "Watching CSS and JS files..."
  chokidar 'src/styles/**/*.scss,css' -c 'npm run esbuild-css && npm run drush-cr && browser-sync reload' &
  chokidar 'src/scripts/**/*.js' -c 'npm run esbuild-js && npm run drush-cr && browser-sync reload'
}
```




7. Comparing Vite, Webpack, and ESBuild





Comparing Vite, Webpack, and ESBuild

Number and size of packages (common dependencies such as sass, glob, etc):

WEBPACK (62MB)	VITE (65MB)	ESBUILD (23MB)
webpack → 17.3MB	vite → 32MB	esbuild → 23MB
webpack-cli → 1.01MB	vite-plugin-static-copy → 33MB	fs-extra → 112KB
sass-loader → 351kB		
css-loader → 913KB		
mini-css-extract-plugin → 21MB		
copy-webpack-plugin → 21MB		https://pkg-size.dev/



Comparing Vite, Webpack, and ESBuild

Compilation times

webpack 5.95.0 compiled with **89 warnings** in 3425 ms

```
dist/styles/node.css
dist/styles/user.css
dist/styles/dialog.css
dist/styles/messages.css
dist/styles/tabs.css
dist/styles/popover.css
dist/styles/buttons.css
dist/styles/forms.css
dist/styles/base.css
dist/scripts/main.js
dist/scripts/components/popover
[vite-plugin-static-copy] Copied 2 items.
✓ built in 3.15s
```

```
buildStyles: 1.065s
buildScripts: 25.223ms
copyVendors: 1.763ms
Total Build Time: 1.095s
```



8. Conclusion and Q&A





esbuild vs gulp vs laravel-mix vs vite vs webpack





Allows modular workflows that adapt to the needs of each project and each developer profile.



Fewer dependencies and ease of configuration.

Frees developers from repetitive and manual processes = increasing consistency, efficiency, productivity, etc.

¡Gracias!
Thanks!





Thanks to our **PLATINUM** Sponsors!!

hiberus

minsa1t
An Indra company



Thanks to our **GOLD**, **SILVER** and **BRONZE** Sponsors!!

BALIDEA

FOR
CONTU
DRUPAL TRAINING & CONSULTING

PANTHEON

SEIDOR

Cambrico
DRUPAL EXPERTS

dropsolid
DIGITAL EXPERIENCE
COMPANY

JAKALA

soluciones

ymbra

<tag>

ZOOCHA

metadrop

1x INTERNET

digrup

idealista

paginar

SNGULAR

NTT DATA