

A black and white photograph of a truck bed filled with large, white, cylindrical sacks. The sacks are tied with twine and some have labels like 'EDISOFT' and 'KANT'. The truck is parked on a dirt surface with a rocky hillside in the background.

Modules, Bundlers, and Components (Oh my).

* * *

Modules

What's a Module?

A unit of code.

Good modules, however, are highly self-contained with distinct functionality, allowing them to be shuffled, removed, or added as necessary, without disrupting the system as a whole. "[Do one thing and do it well.](#)"

What's a Javascript Module?

It depends.

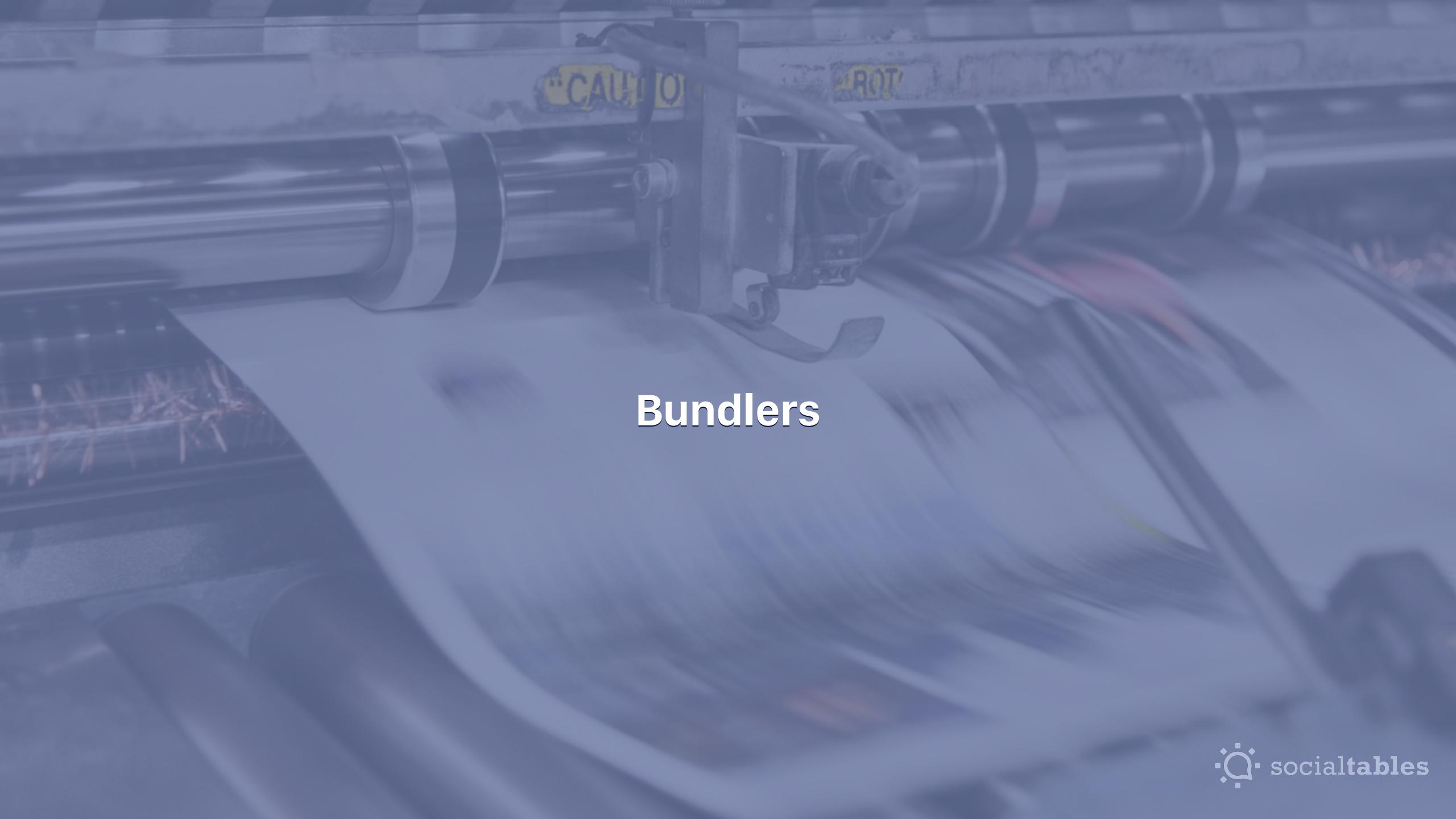
- CommonJS (synchronous load) - Node.js
- AMD (asynchronous load) - Require.js (**do not use**)
- UMD (async or sync) - runs anywhere, even in browser without a loader
- ESM (async) - import and exports - The future is now!

[Node modules in 9.4.0](#)

And then there's registries and package managers.

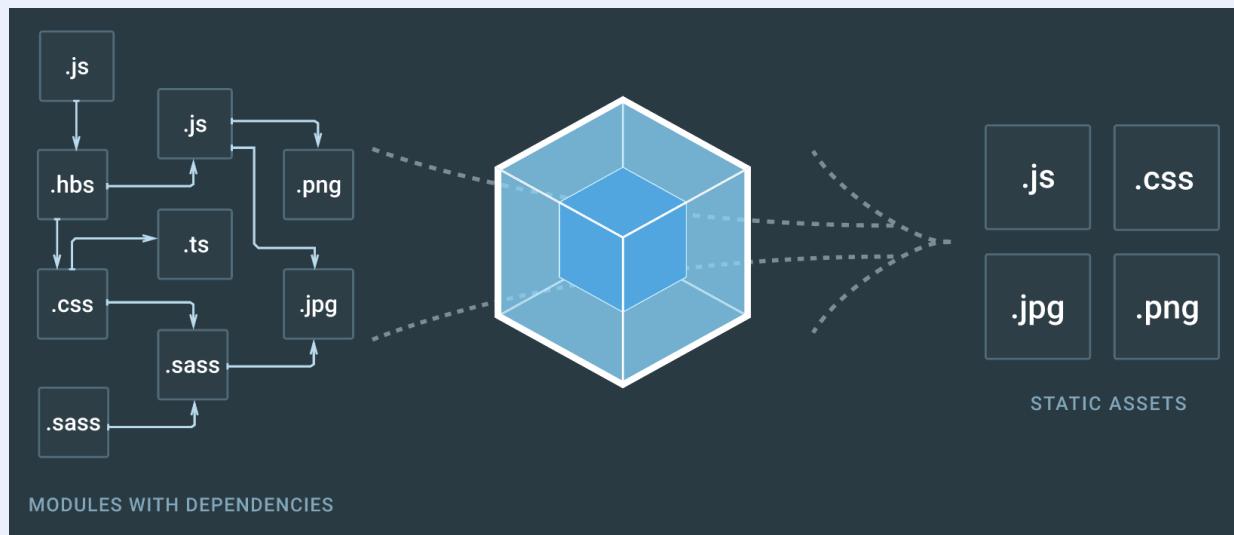
Ewwwwwwwww.

Only **npm** and **yarn** exist (for our purposes) and they use the same format. **package.json** is the source of truth.

A large industrial conveyor belt system is shown in the background, with a prominent yellow "CAUTION" sign attached to one of the metal frames. The conveyor belt itself is dark and appears to be moving. The overall scene has a blue-tinted, slightly blurred effect.

Bundlers

Webpack



Rollup



How do they work?

Magnets.

Important Bits:

- package.json (see the main and module fields)
- input (rollup) or entry (webpack)
- output
- plugins/loaders
- external

What do I use?

Pick one.

In general:

Rollup is better for libraries and modules. Webpack is better for apps.

Either way, you're gonna need...

BABEL

<https://babeljs.io/>

Components

(Let's look at code!)

Examples

White Ravens - <https://github.com/socialtables/white-ravens>

Render Engine - <https://github.com/socialtables/render-engine>

Floor Component (ES Module) - <https://github.com/socialtables/floor-component>

Do

- Use umd and esm as your module formats (CommonJS is ok too)
- Mark React and React-Dom as external - remember devDependencies and peerDependencies
- Consider marking big dependencies like moment or lodash as external

Opinions / Suggestions

- Prefer Rollup for shared components.
- Experiment with es6 modules and tree-shaking.
- Read about esm in node 9.4 with the .mjs extension¹
- Try building a webpack from scratch if you've never²

¹ [The State of Javascript Modules](#)

² [Webpack Academy](#)



Questions?