

# Module bundlers

**DEV.F**  
DESARROLLAMOS(PERSONAS);

dev

# Module bundlers

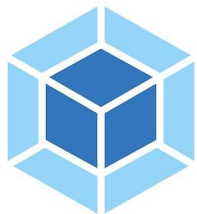
Cuando desarrollamos aplicaciones web modernas, es común:

- Dividir el código en múltiples archivos para una mejor organización y mantenibilidad.
- Desarrollar con frameworks que tienen una sintaxis especial (no solo html, css o js).

Considerando que el navegador solo entiende html, css y js y que existen una complicación con los dos puntos anteriores.

Los "empaquetadores de módulos", son herramientas utilizadas en el desarrollo web para **combinar múltiples archivos de código fuente** en uno solo, facilitando así la gestión de dependencias y la optimización de recursos para su entrega en entornos de producción.

# Ejemplos de module boudlers



webpack



rollup.js



PARCEL

Blazing fast, zero configuration web application bundler



DEV.F

# Transpiladores

**DEV.F**  
DESARROLLAMOS(PERSONAS);

dev

# Transpiladores

Como **JS no es un lenguaje compilado, sino interpretado** no se requiere compilar. Sin embargo, cuando se escribe sintaxis de ES6, código de frameworks u otras herramientas modernas, se requiere de un proceso para convertir todos esos archivos y formato de código en algo 100% compatibles para el navegador (todos los navegadores).

En jerga informática, al proceso anterior se le conoce como **compilación** o más precisamente **transpiración** de código.

- **Babel.**
- **SWC (Speedy Web Compiler).**

# Babel

**DEV.F**  
DESARROLLAMOS(PERSONAS);

dev

# Babel

Es un **transpilador** (como un "compilador") para JavaScript. **Permite transformar código escrito con las últimas y novedosas características** de JavaScript (EsN) y transformarlo **en un código que sea entendido por navegadores más antiguos** (ES5).

Babel es necesario dado que la velocidad de actualización del estándar no es la misma que la velocidad de adopción de los navegadores o no siempre es retrocompatible en ocasiones se requiere transpilar.



# SWC Speedy Web Compiler

**DEV.FX**  
DESARROLLAMOS(PERSONAS);

dev



Es un compilador de JavaScript y TypeScript de alto rendimiento y bajo consumo de recursos. Fue desarrollado para abordar algunas de las limitaciones de los compiladores tradicionales. Es especialmente útil en proyectos grandes o en entornos donde la velocidad de compilación es crítica.



# Empaquetadores

**DEV.F**  
DESARROLLAMOS(PERSONAS);

dev

# Webpack

**DEV.F**  
DESARROLLAMOS(PERSONAS);

dev

# Webpack

Es un empaquetador de módulos para crear entornos de desarrollo y bundles para ambientes productivos.

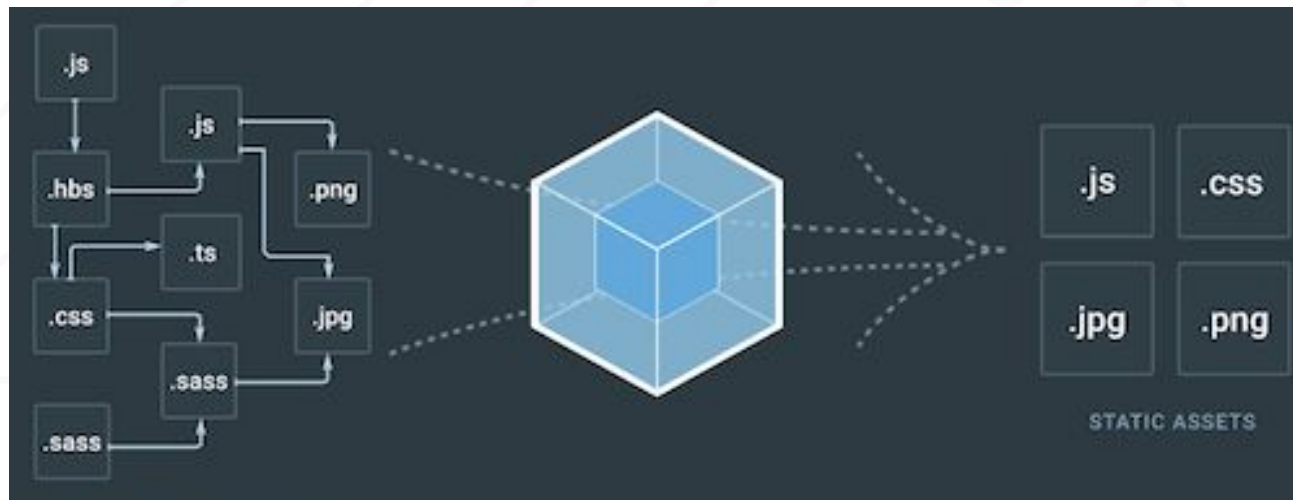
Trabaja con archivos estáticos de front como son:

- HTML.
- CSS.
- JavaScript.
- Archivos estáticos.
- Imágenes.
- Fuentes.
- Iconos.



# Uso de webpack

- Builder de proyectos front end (html, css, js).
- Administrador de la configuración local y de producción.
- Es utilizado por los CLI de herramientas como Vue, React y Angular.
- Minificación de código.
- Simplifica la importación y exportación de módulos.
- **create-react-app usa webpack, babel y otras herramientas por detrás.**



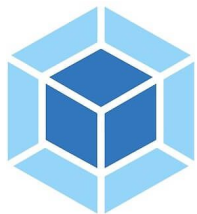
# ¿Por qué se utiliza?

- Las herramientas de desarrollo front end modernas, utilizan sintaxis especial sobre el código escrito en html, css y js. Por ejemplo:
  1. **Vistas** => **HTML** => **React (jsx)** ó **Angular, Vue**
  2. **Estilos** => **CSS** => **Sass, Scss, LESS y Stylus**
  3. **Interactividad** => **JS (es5)** => **ES6 y posterior + frameworks y librerías de front end**

## Conclusiones

Webpack es un generador de bundles para proyectos, tiene muchas configuraciones para convertir nuestros archivos de proyecto en un formato óptimo y compatible. Entre sus herramientas suele usar Babel para el transpilado del JS.

# Ejemplos de module boudlers



webpack



PARCEL

Blazing fast, zero configuration web application bundler



rollup.js



Snowpack