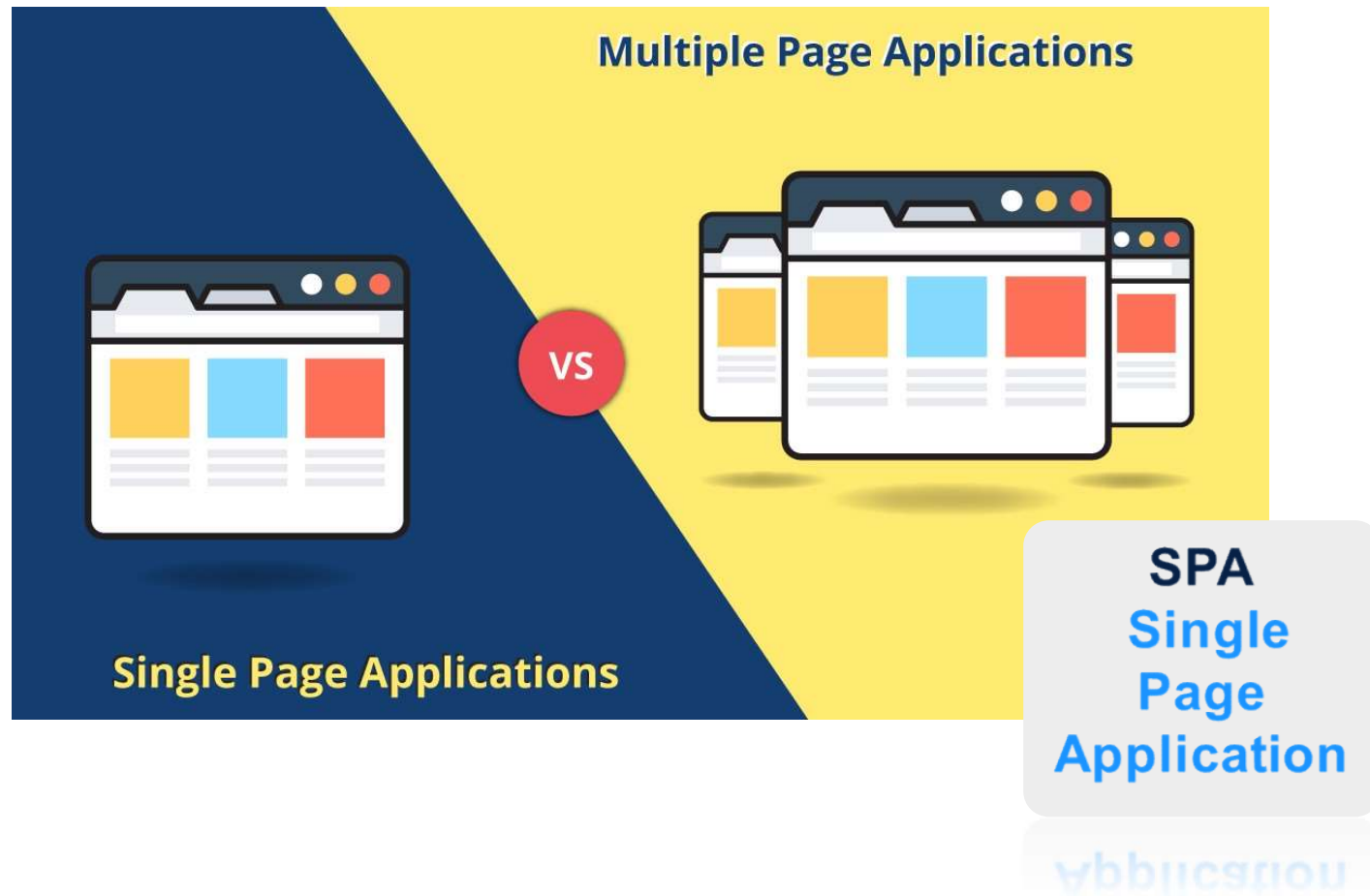




Polymer

SPA | Desarrollo de webapps basado en web components





¿Qué es una web SPA (Single Page Application)?

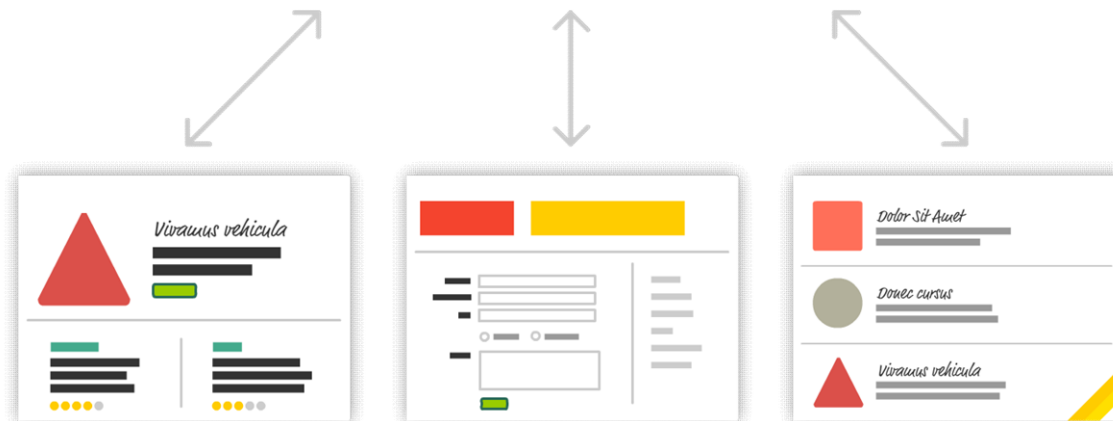
- **Una sola página** con todo el contenido de la web.
- Todo el código HTML, CSS y JavaScript en un **único archivo** que se carga una sola vez.
- Varias vistas (cada vista similar a una página tradicional).
- **Navegación más rápida**: todos los contenidos precargados.
- Nuevos contenidos se cargan **dinámicamente**.
- Se ejecutan en el lado del cliente (navegador).
- Lenguaje de programación: **JavaScript + Bibliotecas y Frameworks** (Angular, React, Polymer, Ember JS, Vue, ...).
- Como backend sólo requieren de una API que proporcione el contenido a mostrar.
- Backend y Frontend totalmente **separados e independientes**.
- Compatibilidad con SEO no es buena (contenido se carga dinámicamente), posible solución: *Server Side Rendering*.



SPA Single Page Application

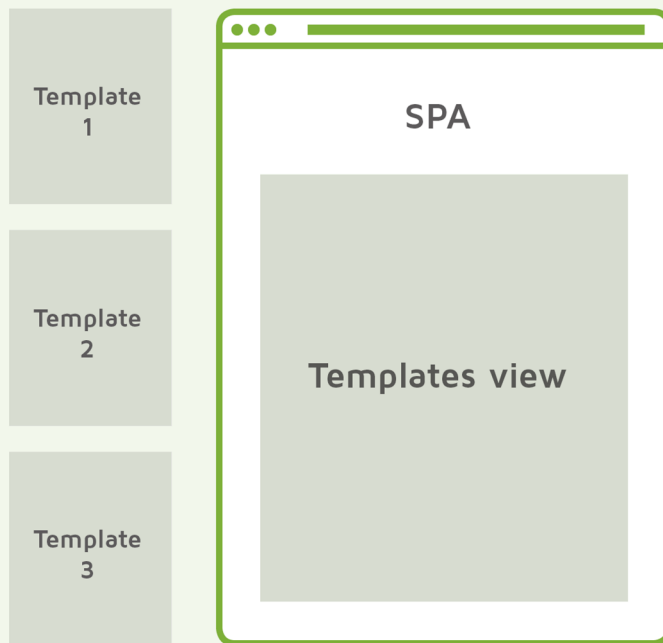


- Gmail
- Google Maps
- Facebook
- Twitter
- Google Drive





Single Page Application



No page refresh on request

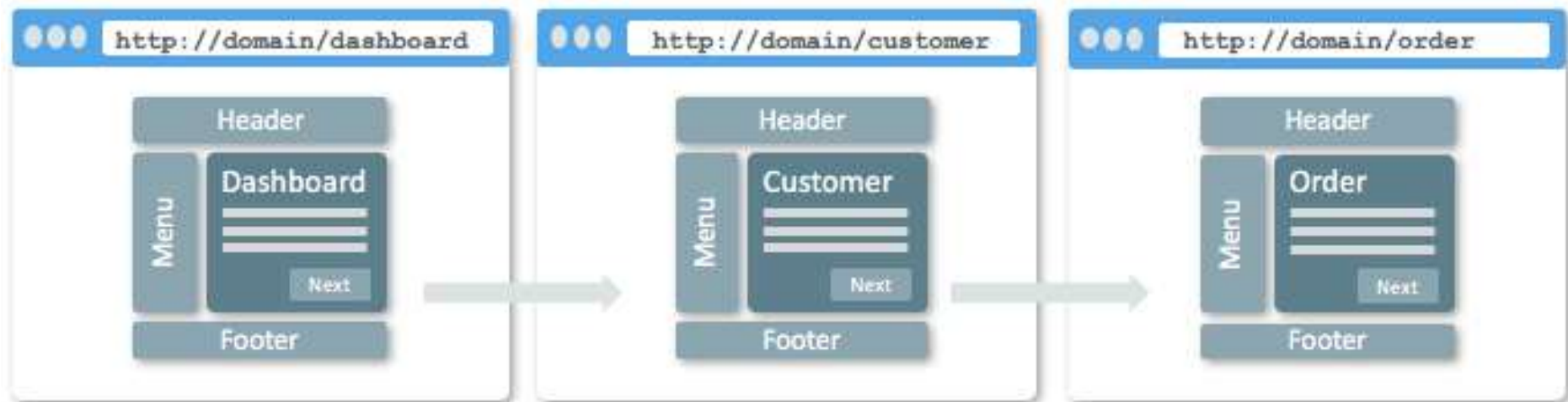
Traditional Web Application



Whole page refresh on request



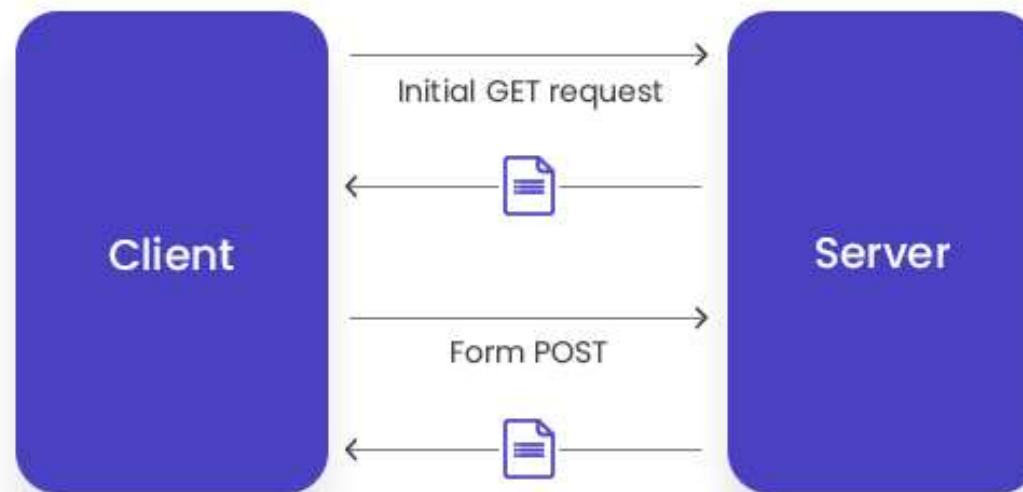
SPA - Single Page Application





SPA Work Differently

Traditional Page Lifecycle



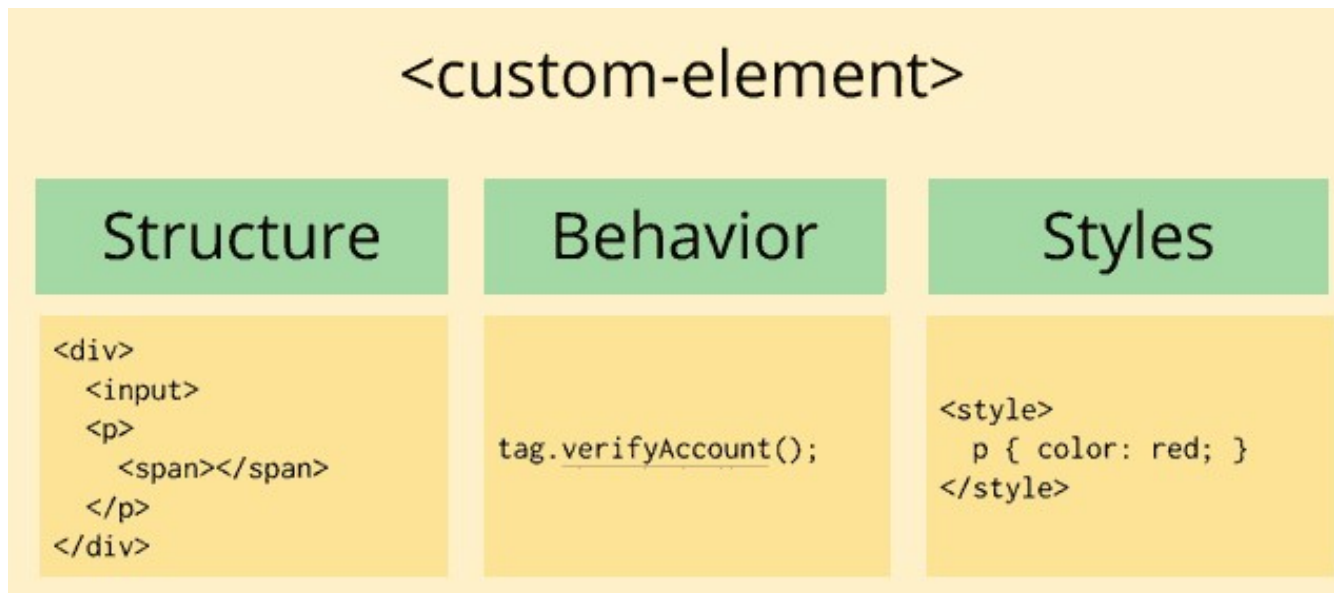


SPA Work Differently





SPA Web Component





Historial de Versiones Polymer



	Mayo 2016	Mayo 2017	Mayo 2018	TDB
Polymer Library (Roadmap)	Polymer 1.0	Polymer 2.0	Polymer 3.0	Polymer Classic
		ES6 Classes Custom Elements V1, Shadow DOM v1	Javascript modules NPM	Lit Element
Cells Support	YES	Yes	Under Development	



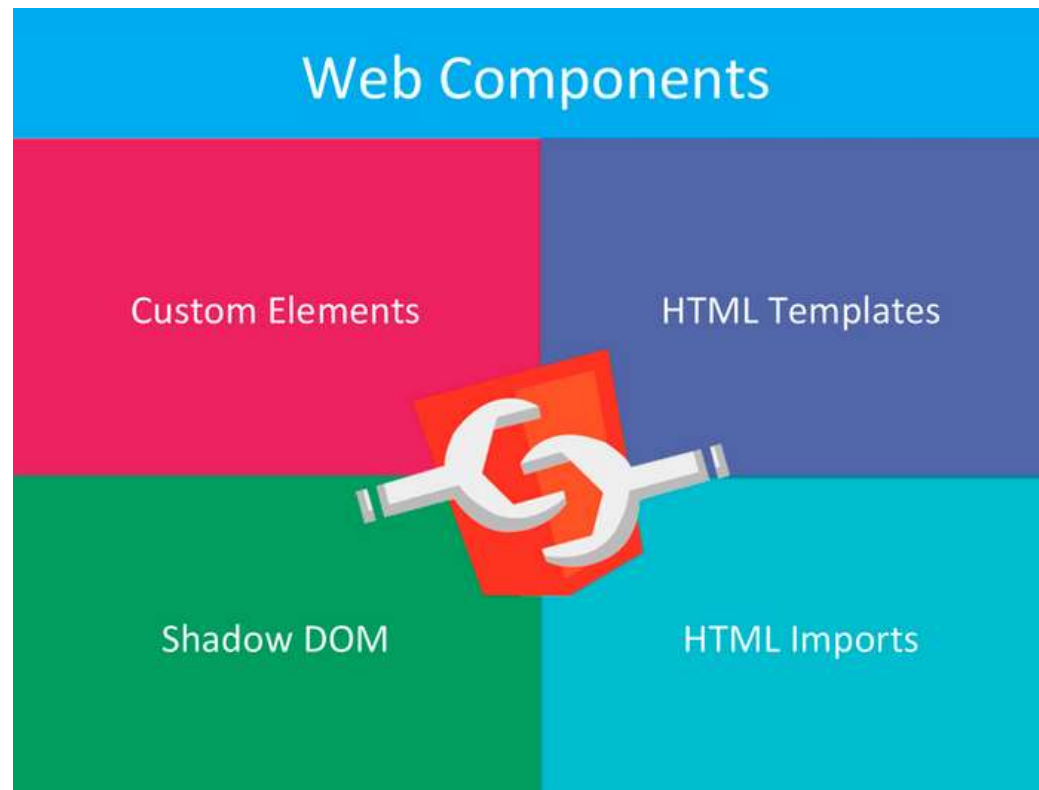
Polymer 2

Herramientas de desarrollo

- Bower (sustituído por **npm***)
- Polymer CLI



Polymer





¿Qué nos ofrece Polymer por encima de Web Components v1?

Funcionalidades del **Core**

- Renderización del template en el **Shadow DOM**
- Propagación del estado (**Data Binding**)
 - **one-way** binding `[[]]`
 - **two-way** binding `{{ }}`
- Definición declarativa de **Manejadores de Eventos** personalizados
- Sincronización de los atributos (en la etiqueta del componente) a propiedades (dentro del propio componente)