The Jamstack, in a nutshell

AN INFOGRAPHIC BY
DARREN DUBE

J

Javascript

Since its introduction in 1995, Javascript has been recognised as the language of the Web. The Jamstack uses client-side Javascript to design web applications using libraries such as React, Vue, and Angular.

A

APIs

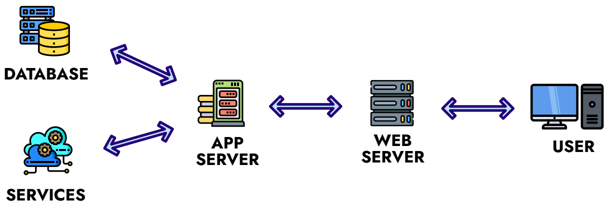
APIs are what allow your website to talk to other services e.g. email providers, authentication services, and contact form providers. Jamstack handles all server-side processing with APIs which can be through third parties, or custom built.

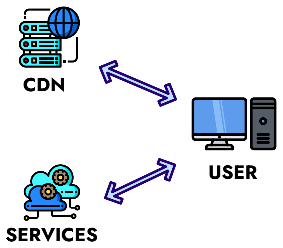
M

Mark-up

Pre-built static HTML is served directly to the client without any processing at load-time. This Static HTML is generated from content (usually written in Markdown, another mark-up langauge) combined with templates for design and structure.

THE TRADITIONAL WEBSITE ARCHITECTURE involves a web server pulling content from a database and generating HTML files **every time** a user loads a page. This makes websites slow - sometimes even useless for people with old devices and unstable internet connections.





THE JAMSTACK'S ARCHITECTURE takes a different approach. Instead of building the website everytime there is a page load, the website is pre-built so that, when a page is loaded, a pre-built static file is immediately served - no precessing needed!

Advantages

1. **Performance**
   Because website pages don't have to be built on the go everytime someone loads them, Jamstack websites are up to 10x faster than traditional websites.
2. **Scalability**
   Serving web traffic from CDNs instead of web servers brings down costs dramatically, increasing scalability.
3. **Security**
   Jamstack websites have fewer points of vulnerability, since no processes/applications have to run to serve a web page.

Disadvantages

1. **Compexity**
   Jamstack websites don't come with a dashboard or GUI - you have to write actual (albeit simple) code to update your website.
2. **Not designed for real time updates**
   The Jamstack may not work well for websites with real-time or user-generated content content (e.g. forums or social media sites) because the website has to be rebuilt everytime something is changed.
3. **Services don't come out of the box**
   Commenting, contact forms, and authentication, among other things, don't come built-in and have to be implemented through third parties.

Resources (clickable links)

Static Site Generators - an Introduction
Netlify's Blog
WTF is Jamstack?

Read my other posts at:
darrendube.com

DARREN DUBE