Single Page Applications & AJAX

Wow. Such buzzwords. Much JavaScript

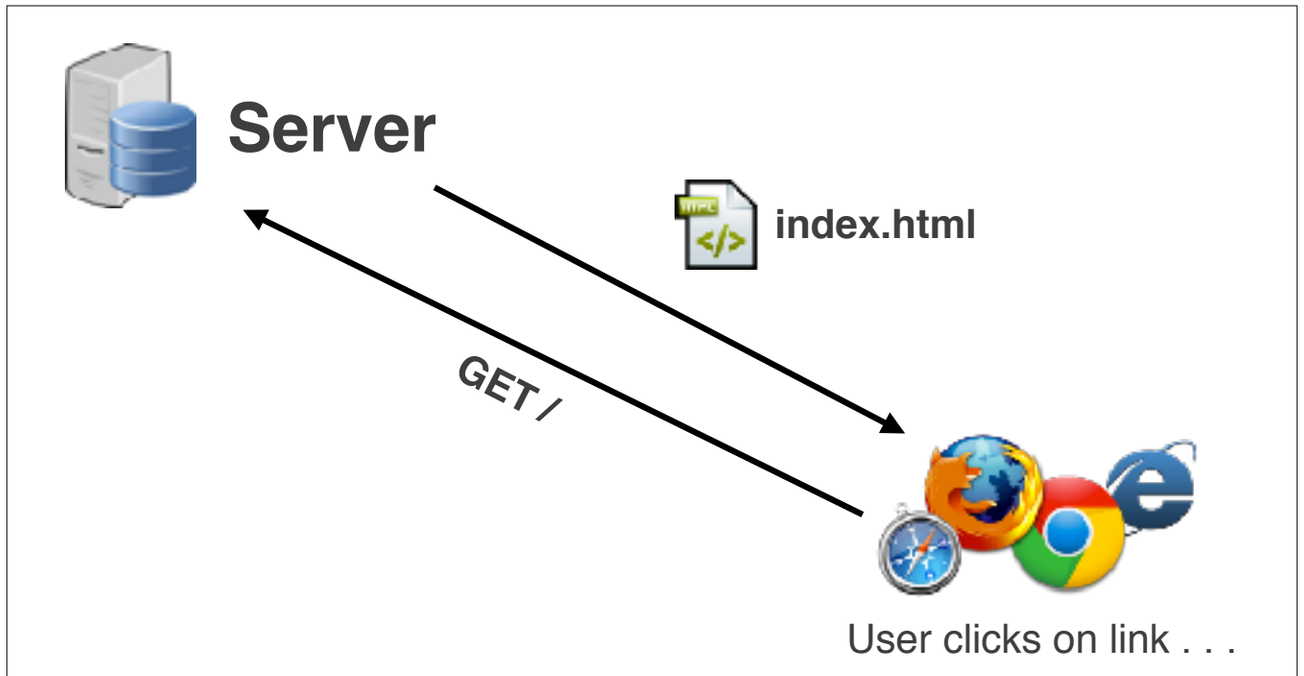
what we have been building so far is not SPA, it’s not the modern (web 2.0) way of making an application. Gmail for example. We want it to feel like a desktop application (no refreshing of the page, just content)

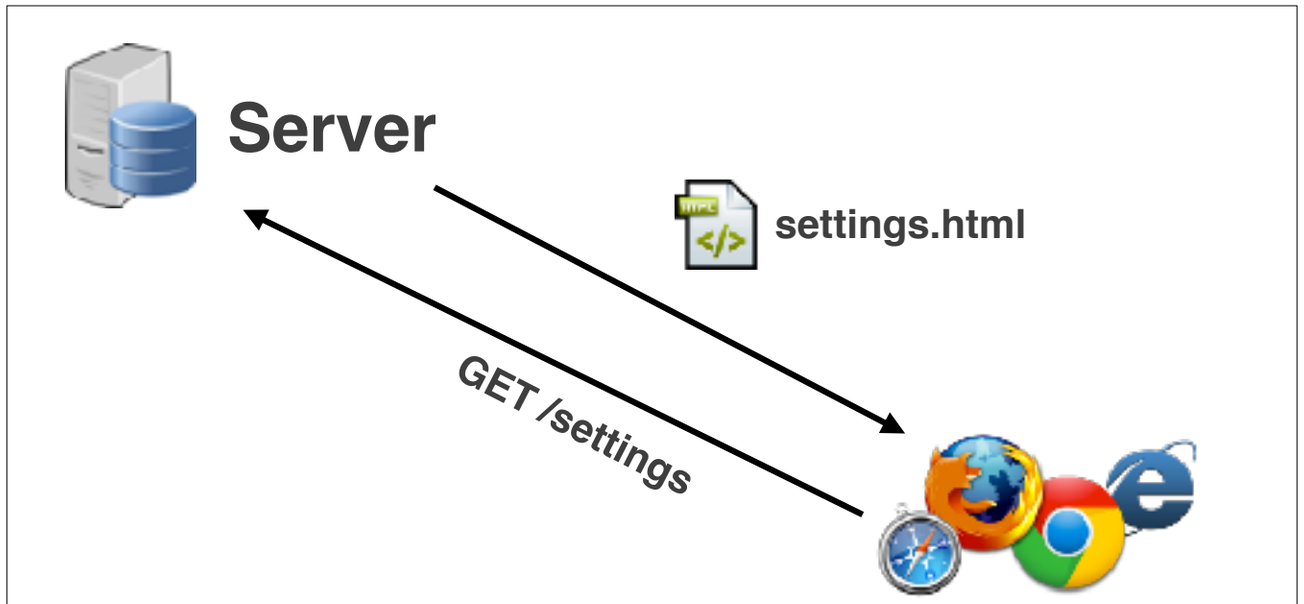
HTTP REQUESTS IN THE BROWSER

- **URL bar**
- **Links**
- **JavaScript**
  - window.location.href = “http://www.google.com”;
- **Submitting forms (GET/POST)**

All of the above make the browser navigate and retrieve new documents.

These all request and retrieve completely new documents, and that’s what we want to learn how to avoid.


Server
index.html
GET /
User clicks on link . . .


Server
settings.html
GET /settings

NOT single page applications

- Views stored on the server, served up as HTML pages.
- When user goes to a new page, the browser navigates in totality, refreshing and retrieving a brand new HTML document.
- Each page, since it is a new page, retrieves stylesheets, script files, etc.

This is not a BAD thing by default; there are many applications where this is better (wikipedia). But for interactive, modern applications, we generally want to use SPA instead.

What is
SPA?

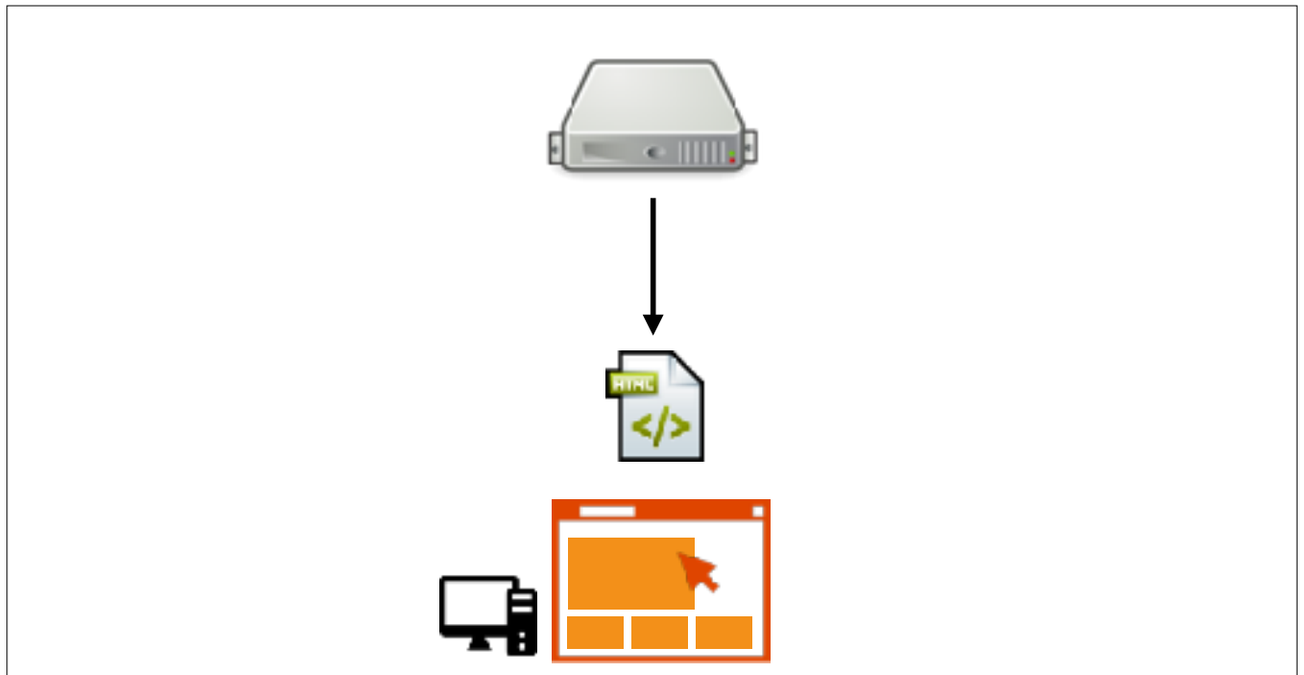
*"A **single-page application** (**SPA**) is a web application or web site that interacts with the user by dynamically rewriting the current page rather than loading entire new pages from a server."*

WIKIPEDIA

or… it fits on a single web page with the goal of providing a more fluid user experience.

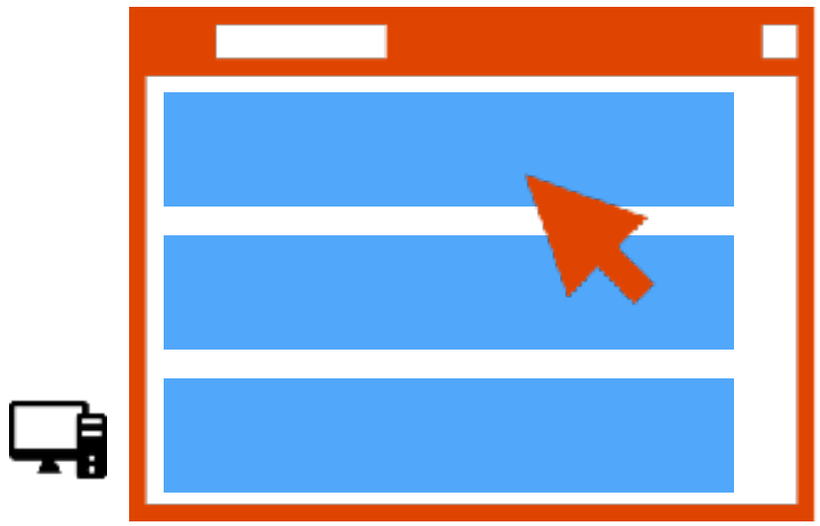
Single Page Applications

- **No reload or "refresh" within the user interface**
  - JavaScript manipulates the DOM as the user interacts
- **User experience similar to a native application**



In Single Page Applications, the client requests a single HTML page

The user interacts w/ the page



JavaScript is used to manipulate the DOM and make the page change without loading a new page.

But… What if additional data is needed to update the page? Well, SPAs rely heavily on AJAX to fetch data.

What is AJAX?

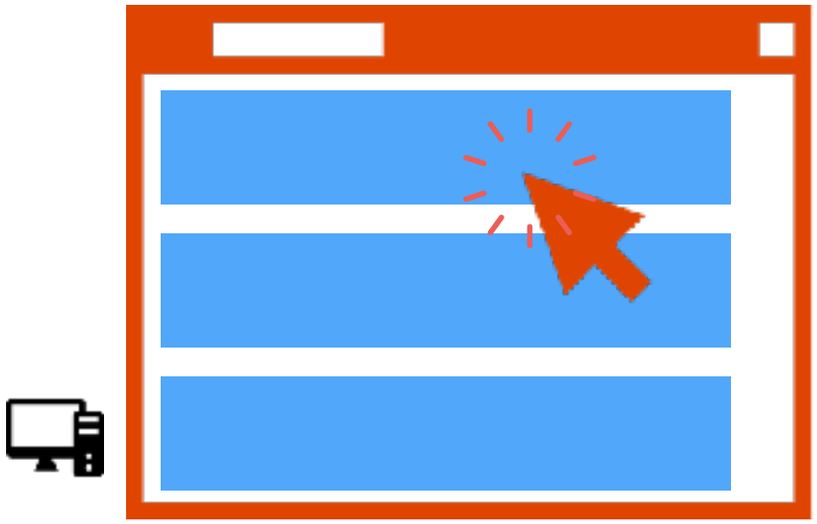
Not something that browsers originally had

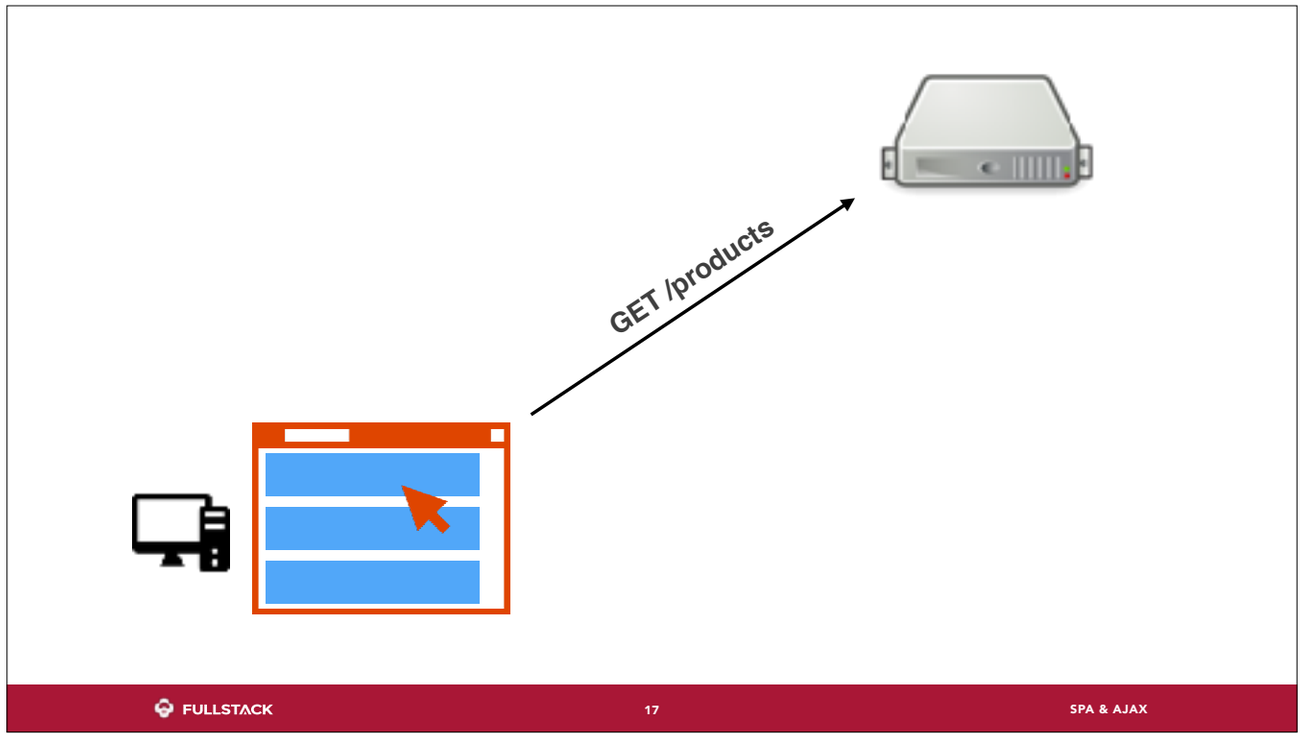
AJAX

- “Asynchronous JavaScript And XML”
- Making background HTTP requests using JavaScript
- Handling the response of those HTTP requests with JavaScript
- No page refresh necessary

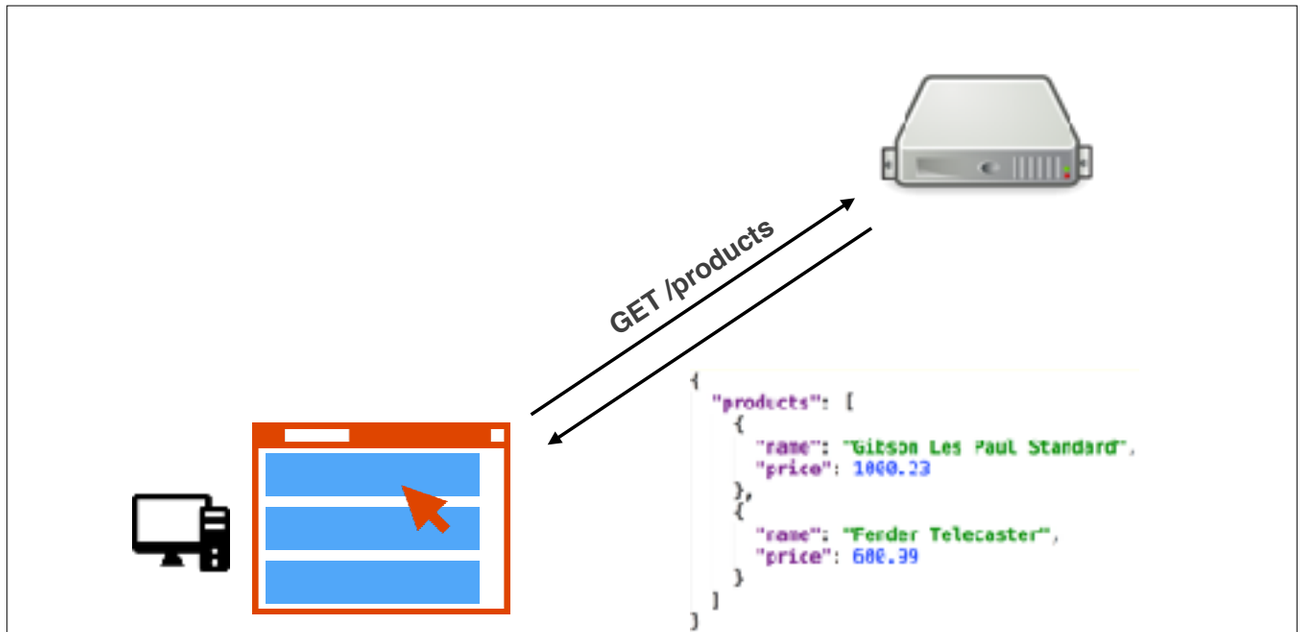
Ajax allows us to not have to reload and refresh, but only update the content we want by manipulating the DOM.

This is a store and we are clicking on a product to view its detail

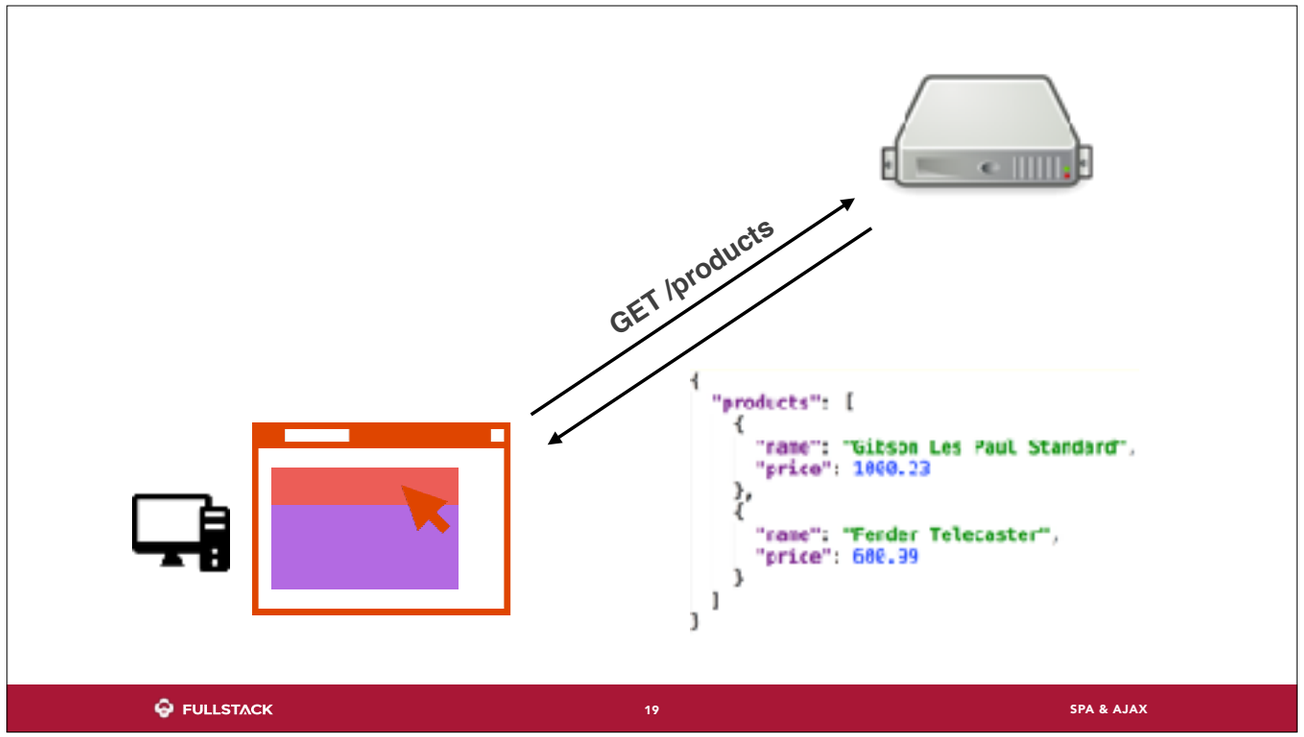




We request the products



And get back JSON


GET /products
{
"products": [
{
"name": "Gibson Les Paul Standard",
"price": 1060.23
},
{
"name": "Fender Telecaster",
"price": 600.39
}
]
}

DATA SERIALIZATION

How our data is transported… the “stringified” version of our data
JSON is an example of how we can serialize js objects (so is XML… but we won’t use that)

DATA SERIALIZATION FORMATS

- How real data is transported or stored
- Used for delivery of data over limited transports, like HTTP.
- Examples: JSON, XML, YAML

RECAP

- **AJAX is the method of sending background HTTP requests with JavaScript**
- **Single Page Applications load a single HTML page and dynamically update the DOM as the user interacts with the app or data gets fetched from the server (using AJAX).**
- **The** Network **tab in Developer Tools is your friend**

window.fetch()