THE WEEKEND THAT CHANGED THE INTERNET

Context

Over a billion users use Google Maps per month. Its user-friendly interface ,accurate geo-location and state-of-the-art interactivity makes it a go to application for all matters location. However, this marvel didn't happen overnight. It used to be one cumbersome and clunky application without much to behold of. This article seeks to bring to light a critical component in this transformation.

The Birth of Google Maps

Google Maps first started as a **C++ downloadable desktop** program curated by two Danish brothers, Lars and Jens Eilstrup and Noel Gordon and Stephen Ma , in a Sydney based company called **"Where 2 Technologies"** which was founded in early 2003. In October 2004, the company was acquired by Google Inc. where the desktop application was transformed to the Web Application , Google Maps. The Rasmussen brothers , Gordon and Ma joined Google at the time .

In the same month, Google Inc. acquired **Keyhole**, a geospatial data visualization company from which some aspects of its core technology were integrated into Google Maps. In September 2004, Google acquired **ZipDash**, a company that provided real-time traffic analysis.

On February 8th 2005 Google Maps was officially launched.

Google Maps before AJAX

To the non-techies, you might be wondering what AJAX is, hold on, we will get to that a little further down the article.

Here's a more detailed breakdown of how Google Maps worked:

- **Initial Page Load** – When a user first accessed Google Maps, the entire application including the initial map data, would be loaded as part of the initial page request. This has been replaced with capabilities such as caching and use of cookies in modern browsers for subsequent visits to a site.
- **User Interaction** – With every user action such as zooming in or out, panning(scrolling) , searching for a new location or performing an action that required new data , this would trigger a new request to the server.
- **Full page refresh** - The server would process the request , generate new map data(usually in HTML) and send it back to the browser. The browser would then completely replace the existing webpage content with the newly received data causing a noticeable flicker. Picture this , you are using Google Maps for directions while walking, with every step you make, the pointer on the application would have to adjust accordingly hence a full page reload was necessary. This would result in the webpage flickering constantly due to the repeated successive full page refresh.

This operational workflow resulted in the following challenges:

1. **Poor Visual Continuity** -The inability to load data in the background meant screens flickers, white flashes and blank pages between transitions and before map data is fully loaded.

2. **Limited interactivity** – Like many web applications of the time, Google Maps interfaces were static hence user could not drag or zoom smoothly. Instead, they had to use buttons to move in cardinal directions. Likewise, zooming was click-based which was cumbersome and tiresome compared to the modern seamless scrolling and zooming directly on the screen.

3. **High latency and slow loading time** – *Latency* is the time it takes for a request to travel from the browser(client) to the server and the response to travel from the server to the browser(client). Since each user action would trigger a full server request and response, this increased server load thus making interactions laggy. Therefore, users had to wait several seconds for a new map section to load and depending on the user's internet speed , this wait would even be longer.

4. **Inefficient data handling** – Fetching a single data item would necessitate fetching the entire map data i.e images on every request. This resulted in redundant data transfer which increased server load as well.

The Transition

In light of the above limitations, change was necessary. **Bret Taylor**, one of the programmers who was instrumental in converting the acquired desktop application to a Web based application(Google Maps) was tasked with solving these challenges. He looked at the chaos and realized the challenge right away. Instead of seeking more resources or assembling a team, Taylor closed himself in a room on a **Friday evening**. In the course of the next two consecutive days, he rewrote Google Map's foundation code. His plan, to use a new technique called **AJAX (Asynchronous Javascript And XML).**

AJAX is *a set of web development techniques that allows webpages to update content in the background without requiring a full page refresh.* At the time, it was still an experimental technique.

By **Monday morning**, it all had changed. The result, an online application where background data loaded without refreshing the page. Users could instantly click, drag and zoom. The navigation was sleek , the reloads had vanished and it was now very addictive. It was lightning fast, streamlined and groundbreaking.

Bret Taylor did not just fix Google Maps, *he had fixed the internet.* AJAX became the standard for dynamic web application. Other tech giants such as Facebook, X (formerly Twitter) , Instagram etc followed suite.

Google Maps after AJAX implementation

The aftermath:

People abandoned paper maps, legacy systems struggled to keep up and the numbers speak for themselves.

- ✓ **Over 1 billion** users rely on Google Maps on a monthly basis.
- ✓ More than **220 countries and territories** are covered by Google Maps.
- ✓ Over **5 million active apps and websites** use Google Maps' APIs.
- ✓ Google Street View covers over **10 million miles of road**.
- ✓ Over **150 million local businesses** are listed on Google Maps.
- ✓ **40%** of Google searches have **local intent** , emphasizing the role of Google Maps in local SEO(Search Engine Optimization).

Conclusion:

Many a times we enjoy the benefits of modern sophisticated technologies but fail to pause and ask ourselves of how they came into being and the brilliants brains behind these marvels. It is my hope that we would start recognizing the heros and heroines who pioneered these life changing products.

Bret Taylor went on to serve as the *CTO* (Chief Technological Officer) at Facebook, *co-founded* Quip(a cloud-based word processor later acquired by Salesforce). He also became the *co-CEO* at Saleforce and later served as the *Chairman of X* during Elon Musk's acquisition and is currently the **Chairman of OpenAI**- the parent company to ChatGpt.

Sources:

Wikipedia

Loopex Digital for Google Maps statistics.

Authored by

Daniel Chacha Mwita