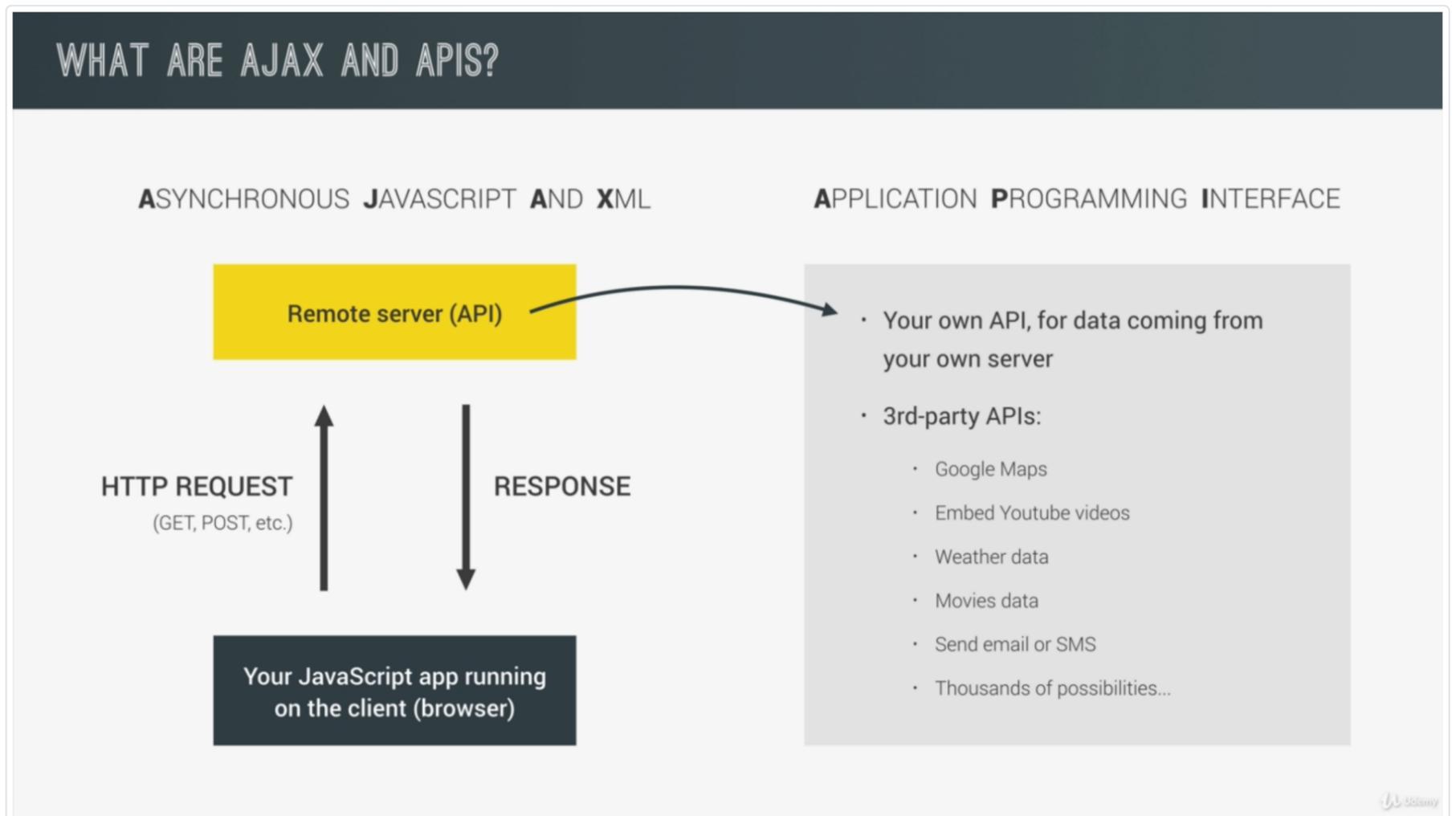


# Asynchronous JavaScript: AJAX & API

Open [ajax\\_api.html](#) for the code | Images are taken from: [JS Course by Jonas Schmedtmann](#)



**AJAX** stands for **A**synchronous **J**avascript **A**nd **X**ML, and basically, AJAX allows us to Asynchronously communicate with Remote Servers.

Let's say we have our JavaScript App running in the browser (which also called "client"), and we want to get some data from the remote server, without having to reload the entire page. With AJAX, we can do a simple GET HTTP Request to the Remote Server, which will then send back a Response containing the data that we requested, and all of this happens asynchronously, in the background (See [event\\_loop.pdf](#)). Also, this not only works for getting data from the remote server, it also works for sending data to the server, by doing a POST Request. In practice, there are many ways to do AJAX in JavaScript, but we will concentrate on the **Fetch** Web API which will allow us to make AJAX calls in a simple way.

Since we are talking about servers, we also have to talk about **APIs**, which stands for **A**pplication **P**rogramming **I**nterface. At an extremely high level, API is a piece of software that can be used by another piece of software in order to allow applications to talk to each other. And that's not just valid for Web Development, but for Programming in general. In reference to Web Development, AJAX & Servers, API is not the server itself, but a part of the Server (Eg: An application that receives requests and sends back responses).

There are two types of APIs that we can use in JavaScript

1. Your own API for data coming from your own server.
2. Third Party APIs:
  - Google Maps,
  - Embed YouTube Videos,
  - Weather Data,
  - Movies Data,
  - Send Email or SMS,
  - etc.

Let's imagine for a second that we are building a travelling app, and we have database with different destinations and tourse that we are offering. Therefore, on our server, we could build our own API that can receive request from our front-end app and send back the results. So that would be our own API hosted on our own server. But now we also need some current weather information for each destination, and that data is not available in our database, so for that, we will use a 3<sup>rd</sup> Party API to get that weather data of the destination. And there are a lot of 3<sup>rd</sup> Party API's and some of them are mentioned above.