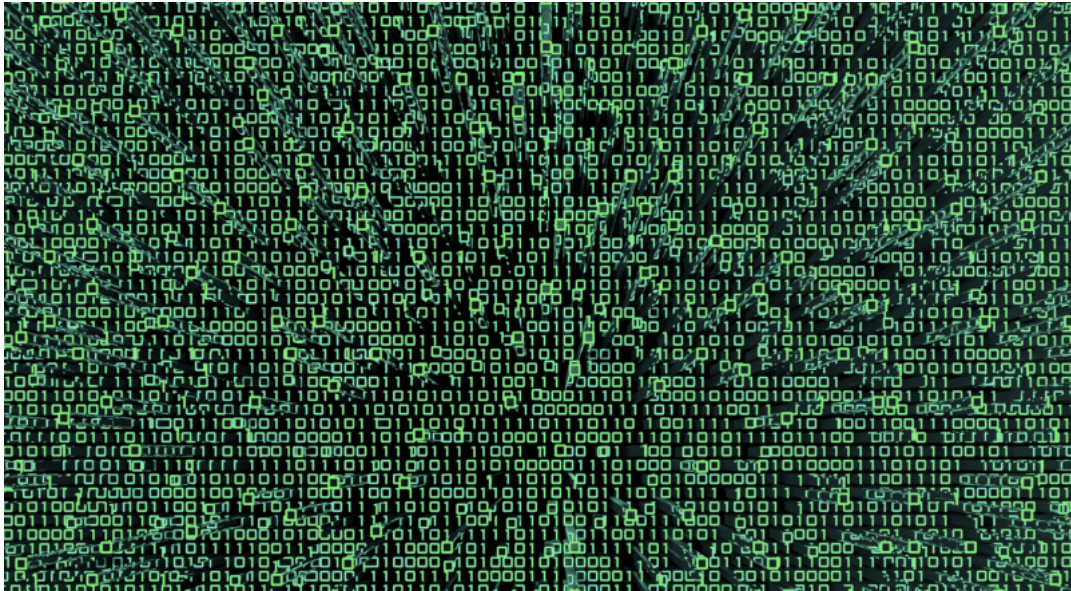


# Was ist das Problem?



Bisher mussten wir die Seite immer neu Laden, wenn wir neue Informationen brauchen.

Nun sind diese Zeiten vorbei

Jetzt laden wir einfach nur noch das, was wir wirklich brauchen!

```
<link rel="stylesheet" href="/css/stylesheet.css">  
//Ich bleibe da!
```

## Ah... Jax?

Wofür steht Ajax?

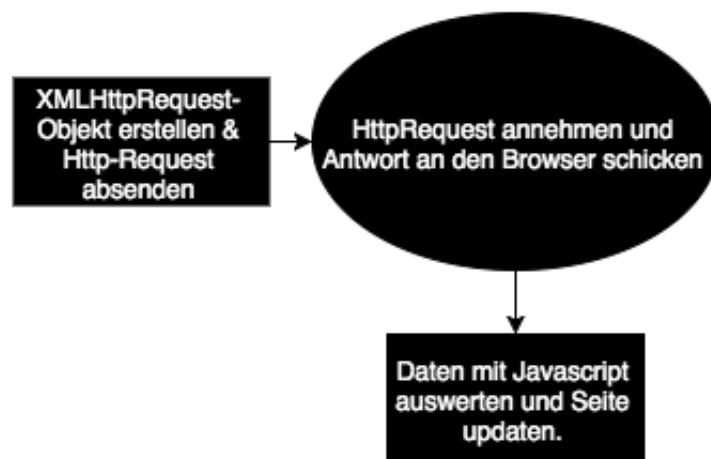
Ajax bedeutet Asynchronous JavaScript and XML

Woher kommt Ajax?

Neben der Microsoft Remote-Scripting-Component 1998 zählt man Jesse James Garrett zu den Begründern der Idee des Ajax und dem formen des Begriffs.

Was ist ein XMLHttpRequest?

XMLHttpRequest ist eine Schnittstelle für JavaScript um Daten über das HTTP-Protokoll zu laden. Kurz gesagt: Es ermöglicht einem Script, ohne die Seite neu zu laden Daten aus einem Server abzurufen und einzubetten.



## Was brauche ich denn dafür?

Eigentlich gar nicht viel. Du solltest HTML(5) kennen und auch Javascript. CSS ist praktisch, aber nicht notwendig! Ein Editor wird gebraucht, bevorzugt natürlich einer mit Syntax-Highlighting, zur Not geht aber auch der normale Texteditor. Dann legt folgende Datenstruktur in eurem Arbeitsordner an:

```
/index.html  
/css/styleSheet.css //optional  
/script/javascript.js  
/script/testfile.txt
```

# Dann mal los! (Der Grundbau)

Zuerst brauchen wir ein HTML-Dokument, in das wir unsere index.html laden:

```
<html>
  <head>
    <script type="text/javascript"
src="/script/javascript.js"></script>
  </head>

  <body>
    <h1> Dies ist eine Überschrift </h1>
    <p> Dies ist ein Statischer Text, er wird nicht
verändert!</p>
    <button type="button" id="mybutton"
onclick="getMore()">Lade Daten</button>
    <div id="demo">Was sagt das Orakel?</div>
  </body>
</html>
```

In die testfile.txt kann irgendein Satz geschrieben werden.

Dies ist ein Satz. Wenn du ihn in deinem HTML siehst,  
dann hat es funktioniert!

## getMore() function!

Nun wird die /script/javascript.js geöffnet:

Mieps!?

In diese Datei muss nun eine Funktion getMore(),

welche uns die neuen Daten holt.

# Was ist denn jetzt ein XMLHttpRequest?

Wir brauchen ein Objekt, dass die Anfrage an neuen Inhalt sendet, die Daten annimmt und eine Funktion, diese zu implementieren.

```
function getMore() {  
    'use strict';  
    var xhttp = new XMLHttpRequest();  
    xhttp.onreadystatechange = function () {  
        if (xhttp.readyState == 4 && xhttp.status == 200) {  
            document.getElementById("contents").innerHTML =  
xhttp.responseText;  
        }  
    }  
  
    xhttp.open("GET", "/ajax-test-  
package/script/text.txt", true);  
    xhttp.send();  
}
```

## Erläuterung der Funktionen:

'use strict' //Nicht so wichtig, aber ohne könnten bestimmte Regeln im JS vernachlässigt werden, was wir nicht wollen.

```
xhttp = new XMLHttpRequest();
```

Ein neues Objekt vom Typ XMLHttpRequest() wurde erstellt und xhttp genannt. Sollte man aus AP noch

kennen, da Syntax ähnlich zu Java.

```
http.onreadystatechange = function ()
```

Diese funktion testet, ob sich etwas geändert hat und führt dann die Funktion aus, die in den {} definiert wird.

```
(xhttp.readyState == 4 && xhttp.status == 200)
{
    document.getElementById("contents").innerHTML =
xhttp.responseText;
}
```

Zwei Bedingungen müssen erfüllt sein:

readyState muss 4 sein. Wobei Gilt:

0: request not initialized  
1: server connection established  
2: request received  
3: processing request  
4: request finished and response is ready  
State muss 200 sein, 404 wäre zum Beispiel ein "NOT-FOUND" fehler.

```
document.getElementById("contents").innerHTML =
xhttp.responseText;
```

Der das Element mit der ID "contents" wird genommen und in das HTML wird das eingesetzt, was xhttp als Antwort erhält.

```
xhttp.open("GET", "/ajax-test-package/script/text.txt",
true);
```

Die Anfrage, die das Request an den server stellt, in diesem Falle eine Anfrage mit GET nach einer text-Datei. Das true am ende ist für die Asymmetrie zuständig. Setzt man es auf False, dann findet die Übertragung nicht mehr Asynchon statt.

```
xhttp.send();
```

Sendet das Paket dann ab.

# Puh, das war jetzt viel auf einmal, oder?

Hier, zur Entspannung, eine Babykatze!



## Zurück zum Ernst des Lebes: Der Ablauf beim Klicken auf den Button

