



Ajax

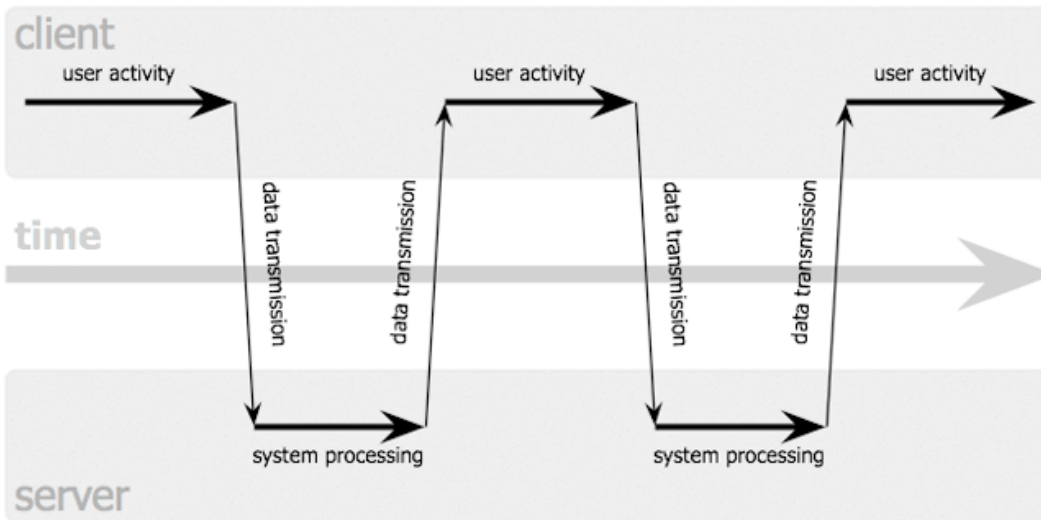
Matt Novinger

E-day +1

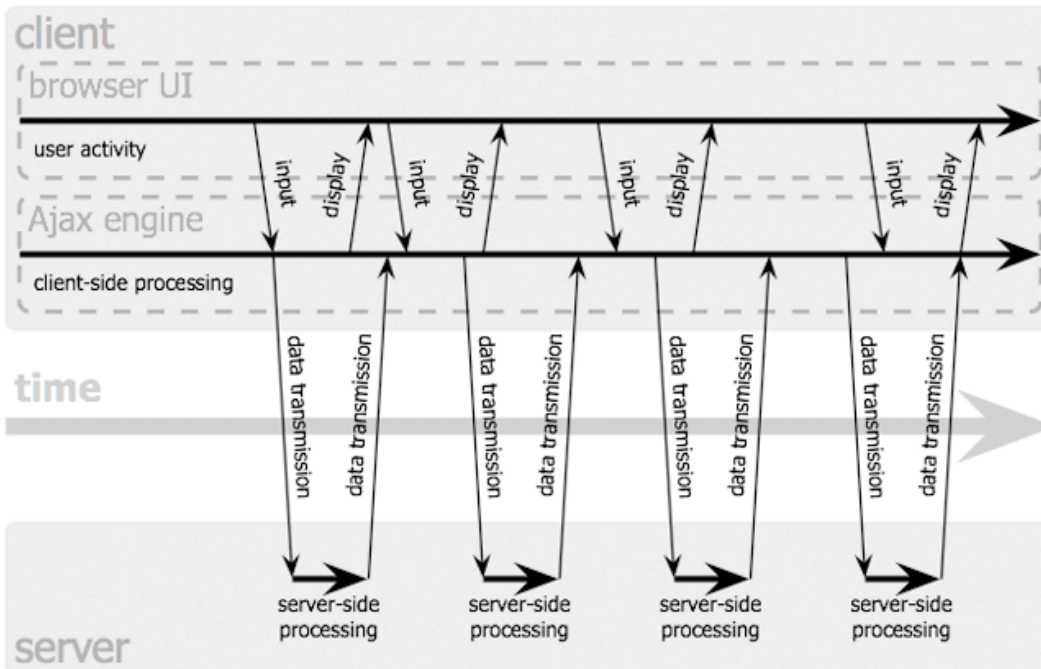
Asynchronous JavaScript and XML

- ◆ Asynchronous
 - Makes requests in the background, without interfering with the current page
- ◆ JavaScript
 - Used to initiate the back ground request and process the response
- ◆ XML
 - The format of the data that is returned from the server. This is in contrast to the typical html page that is returned from a normal request

classic web application model (synchronous)



Ajax web application model (asynchronous)



Why?

- ◆ Allows you to load content without reloading the entire page
- ◆ You can fetch new content based on user input
- ◆ Perform long running tasks in the background

How?

- ◆ XMLHttpRequest
- ◆ “a DOM API that can be used by JavaScript and other web browser scripting languages to transfer XML and other text data between a web server and a browser.” – wikipedia
- ◆ DOM stands for Document Object Model; it defines an API for interacting with elements of an HTML or XML page

example

```
function ajax(url, vars, callbackFunction) {  
    var request = new XMLHttpRequest();  
    request.open("POST", url, true);  
    request.setRequestHeader("Content-Type",  
        "application/x-www-form-urlencoded");  
  
    request.onreadystatechange = function() {  
        var done = 4, ok = 200;  
        if (request.readyState == done && request.status == ok) {  
            if (request.responseText) {  
                callbackFunction(request.responseText);  
            }  
        }  
    };  
    request.send(vars);  
}
```

the easy way.

- ◆ There are many JavaScript frameworks available that hide the details of XMLHttpRequest.
- ◆ Prototype

```
new Ajax.Request('/some_url', { method:'get' });
```

- ◆ YUI (yahoo! user interface)

```
YAHOO.util.Connect.asyncRequest('GET', sUrl, callback);
```

- ◆ jQuery

```
$.get("test.php");
```