software studio

asynchronous calls

Daniel Jackson

some history

in the 1990s

- most web sites issued a whole page at a time
- clunky for users, excessive bandwidth

idea

- update page incrementally
- do it asynchronously, so browser doesn't freeze

in 1999, XMLHttpRequest arrives

- Microsoft invents XHR idea for ActiveX in IE5
- adopted by other browsers for Javascript

in 2005, called "Ajax"

- asynchronous JavaScript and XML

what you can do with Ajax

push updates to page

- status, stock quotes, weather, time

interact with user over input

- drill downs (eg, state>city), autocomplete

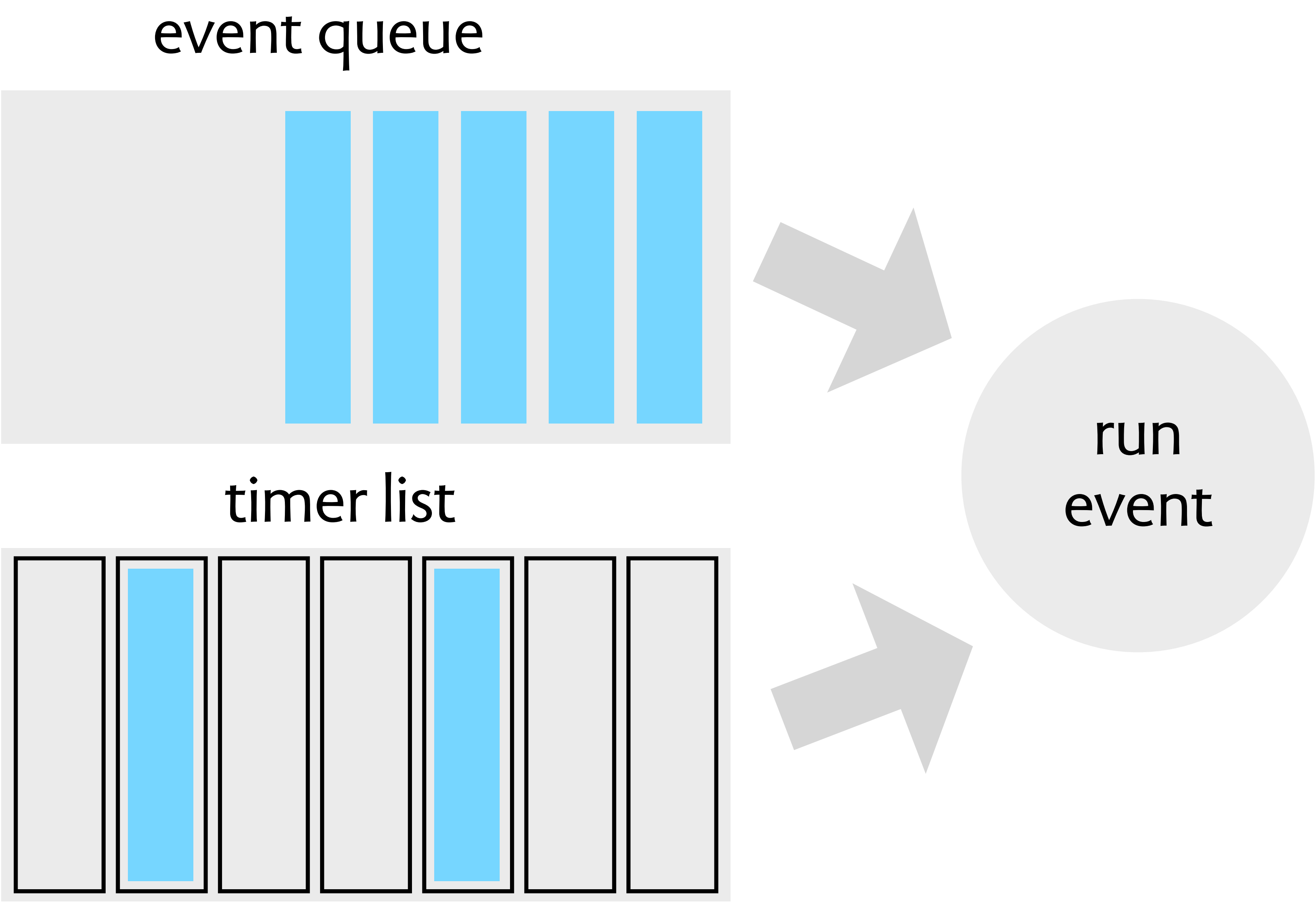
persist user data on server

- Google docs, stickies

execute server-side commands on same page

- vote up/down on posts, email portal

event model



- single thread
- event loop: run timed event if expired or event at front of Q
- events added by: user actions, server responses, events

particulars

two key facts

- each event runs to completion before next event
- response to Ajax request is not always next

good news

- no need to worry about mutual exclusion

bad news

- long-running event freezes the UI
- timeouts may run late
- when Ajax response comes, context may have changed

JQuery's Ajax API

$.getScript(url, callback)

- get script at url, run it, then execute callback

e.load(url)

- get content at url, and insert into element e

$.getJSON(url, data, callback)

- pass data to server at url
- server sends JSON back; parse and pass to callback

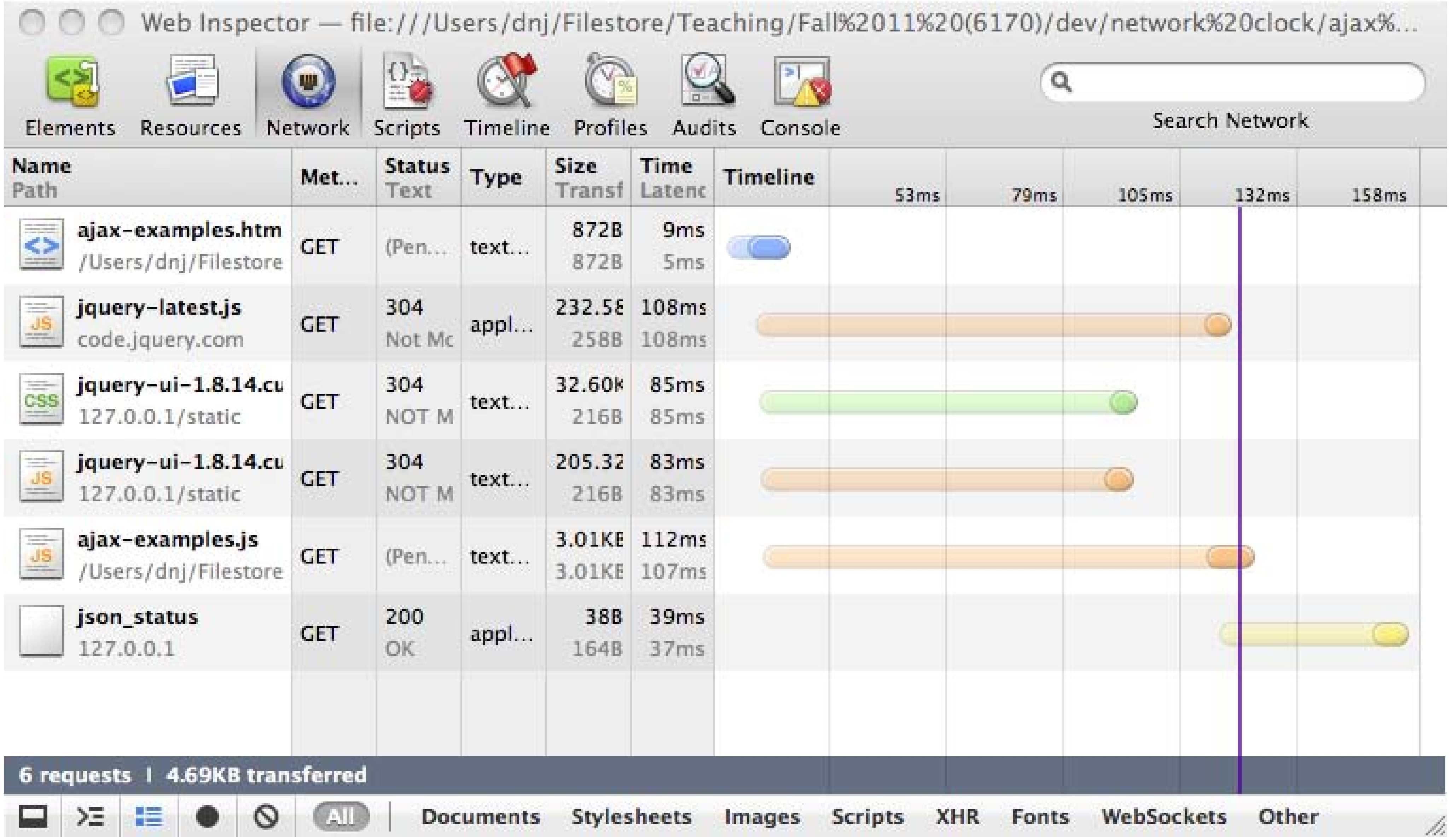
$.get(url, data, callback, type)

- type determines preprocessing of response
- $.post similar, but does HTTP post instead

$.ajax(...)

- most flexible, lowest level method

using network inspector



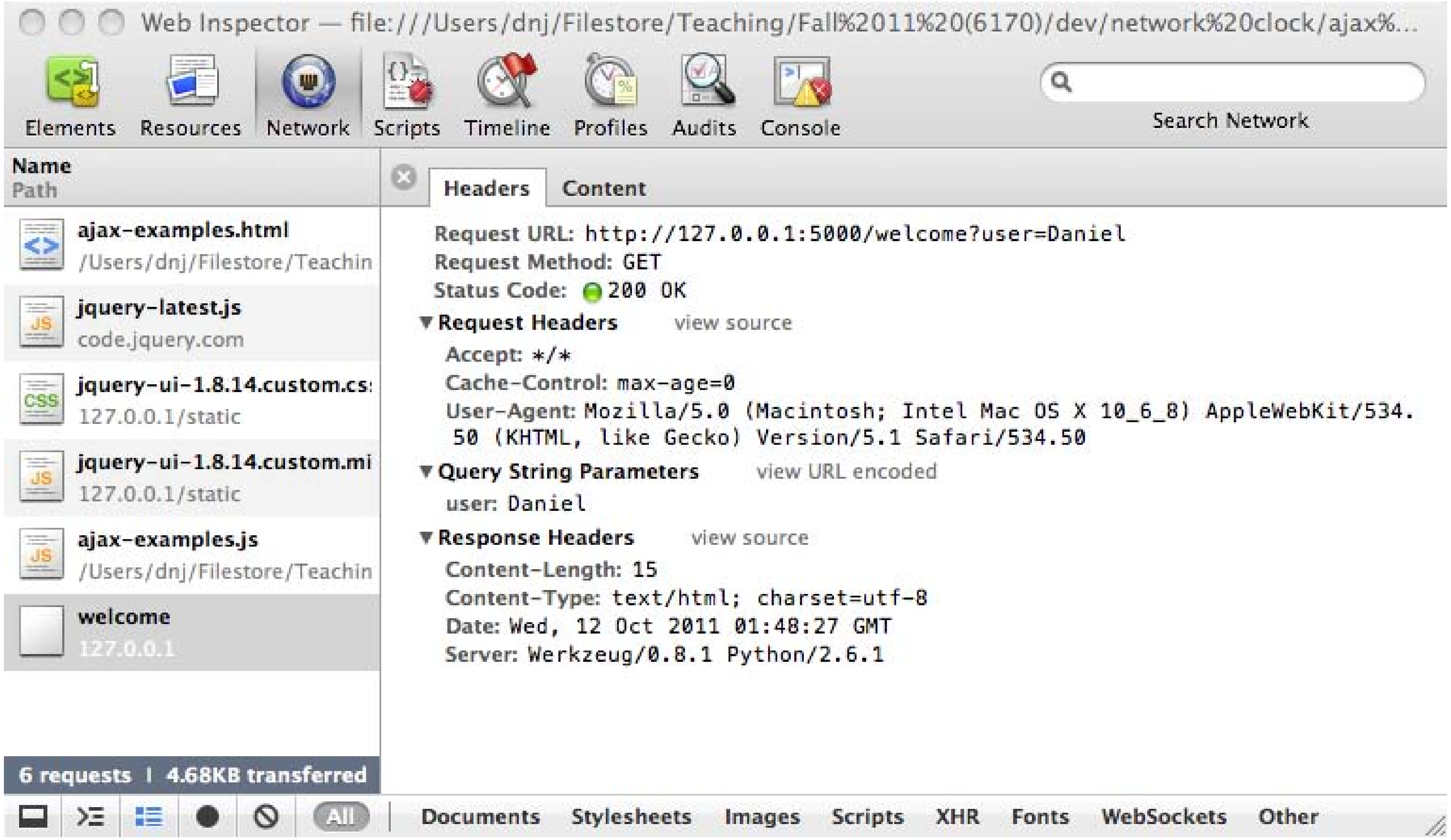
Web Inspector — file:///Users/dnj/Filestore/Teaching/Fall%2011%20(6170)/dev/network%20clock/ajax%...

Name Path	Met...	Status Text	Type	Size Transf	Time Latenc	Timeline
ajax-examples.htm /Users/dnj/Filestore	GET	(Pen...	text...	872B 872B	9ms 5ms	
jquery-latest.js code.jquery.com	GET	304 Not Mc	appl...	232.58 258B	108ms 108ms	
jquery-ui-1.8.14.cu 127.0.0.1/static	GET	304 NOT M	text...	32.60k 216B	85ms 85ms	
jquery-ui-1.8.14.cu 127.0.0.1/static	GET	304 NOT M	text...	205.32 216B	83ms 83ms	
ajax-examples.js /Users/dnj/Filestore	GET	(Pen...	text...	3.01KE 3.01KE	112ms 107ms	
json_status 127.0.0.1	GET	200 OK	appl...	38B 164B	39ms 37ms	

6 requests | 4.69KB transferred

- example from Safari

using network inspector



ajax-examples.html
/Users/dnj/Filestore/Teachin

jquery-latest.js
code.jquery.com

jquery-ui-1.8.14.custom.cs:
127.0.0.1/static

jquery-ui-1.8.14.custom.mi
127.0.0.1/static

ajax-examples.js
/Users/dnj/Filestore/Teachin

welcome
127.0.0.1

6 requests | 4.68KB transferred

```
Request URL: http://127.0.0.1:5000/welcome?user=Daniel
Request Method: GET
Status Code: ● 200 OK
▼ Request Headers      view source
  Accept: */*
  Cache-Control: max-age=0
  User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_6_8) AppleWebKit/534.
  50 (KHTML, like Gecko) Version/5.1 Safari/534.50
▼ Query String Parameters      view URL encoded
  user: Daniel
▼ Response Headers      view source
  Content-Length: 15
  Content-Type: text/html; charset=utf-8
  Date: Wed, 12 Oct 2011 01:48:27 GMT
  Server: Werkzeug/0.8.1 Python/2.6.1
```

- can see here that request was get

encoding data for transit

XML

- parsing built into browser (XHR)
- comes back as DOM: not convenient

JSON

- Javascript object literals
- JQuery uses parser, not eval (why?)

```
<person>
  <firstName>John</firstName>
  <lastName>Smith</lastName>
  <age>25</age>
  <address>
    <streetAddress>21 2nd
Street</streetAddress>
    <city>New York</city>
    <state>NY</state>
    <postalCode>10021</
postalCode>
  </address>
  <phoneNumber type="home">212
555-1234</phoneNumber>
  <phoneNumber type="fax">646
555-4567</phoneNumber>
</person>
```

```
{
     "firstName": "John",
     "lastName" : "Smith",
     "age"      : 25,
     "address"  :
     {
         "streetAddress": "21 2nd Street",
         "city"         : "New York",
         "state"        : "NY",
         "postalCode"   : "10021"
     },
     "phoneNumber":
     [
         {
           "type"  : "home",
           "number": "212 555-1234"
         },
         {
           "type"  : "fax",
           "number": "646 555-4567"
         }
     ]
 }
```

examples from http://en.wikipedia.org/wiki/JSON