AJAX & Web Services

1. What is Ajax?
2. Other important concepts
   - loose coupling aka weak coupling
   - REST architectural model
   - Same origin policy / effect on proxies
   - Javascript Sandbox
   - SOAP from Web Services lecture
3. Client-side
4. Server-side

---

1) What is Ajax
   - term originated in 2005 (see Readings) but technology had been developed and implemented (by Microsoft) since circa 1998
   - acronym-like term for Asynchronous JavaScript + XML
   - what does it do?
     Allows the web client (browser) to request & load one part of a webpage at a time, instead of all at once

Fig. from Eichorn (2007)
reference at end

FIGURE 1-3
Normal Web application requests versus AJAX requests
How does it do that?

- Ajax is a technique enabled by XMLHttpRequest
- See code by Pruett (2006) for example

→ 5 stages

0 uninitialized (before the open() call)
1 loading (object created but not send() called)
2 loaded (send() called, but nothing retrieved from server, yet)
3 interactive (some data has been received but not finished yet)
4 completed (get data from response Body and responseText properties not null)

Note: not all browsers update the state the same way, but all are ready at stage 4
Concepts (cont.)

loose (or weak) coupling
  - closely related to encapsulation
  - built into layer model

REST architectural model
  - Representational State Transfer
    - coined by Roy Fielding in 2000
    - juxtaposed against the RPC model we saw in the Web Services lecture
      - based on client/server model
        - stateless
        - cachable
        - layered

To communicate between resources

application state functionality

using well-defined operations and a limited number of content-types

allows for heterogeneity and scalability

Meant for large, mega-scale hypermedia, not www in general. Implemented in WebDav (modulo the use of POST) and Ruby on Rails v2.1.

*From Wikipedia, Representational State Transfer

17 March 2007  18:43
Concepts (cont.)

Same Origin Policy (for proxies with Ajax)

Web browsers do not allow XMLHttpRequest connections to servers that did not deliver the original webpage from which the request originates (Pruett (2006), p. 24)

Figures based on
Fig. 4 from
Pruett (2006)

Q: Why does this work?

1. initial request & response
2. new Ajaxian request
   original server acts as proxy
3. second half of the proxy transfer
References

Understanding AJAX
by Joshua Eichorn
(c) 2007 by Pearson Education
Published by Prentice-Hall ISBN 0-13-221635-3

Ajax and Web Services
by Mark Pruet
(c) 2006 by O'Reilly Media, Inc.
O'Reilly ebook with ISBN 0-59-652853-1

In Appendix to these notes: pp. 6, 7, 18-23.

See Also
Tutorials & Readings in the Resources part of the website