INFO/CS 4302
Web Information Systems
FT 2012
Week 3: The Web Architecture:
  hands-on http
  (Lecture 5)

Theresa Velden
Housekeeping

• Progress Team Formation
• Cross Cutting Issue Poll: still open for another few hrs
  – Internet Censorship
  – Internet Surveillance
  – Net Neutrality & Openness
RECAP: IDENTIFICATION & INTERACTION
Recap: A web resource is

- An entity with an identity (URI)
- An abstract: you can’t see, smell, touch…
- A service point for initiating protocol (HTTP) actions
- A target of hyperlinks
  `<a href=“http://google.com”>`
Recap: (http) URIs

• identifiers for web resources associated with the hierarchical namespace governed by a DNS authority
  – who potentially could set up a http origin server as a host at the given address, listening for TCP connections on a given port

• http URI syntax:

http://www.infosci.cornell.edu:80/Courses/info4302/2012fa/
Recap: Cool URIs

What makes a cool URI?
A cool URI is one which does not change.
What sorts of URI change?
URIs don't change: people change them.

Tim Berners-Lee (http://www.w3.org/Provider/Style/URI)

• Generic vs. content-type specific URIs
  – Cool URIs don’t change with the emergence of a new internet media type for web resource representations
    • Generic:
    • Content type specific:

• Remember that a content-type specific URI represents a ‘Leap of faith’: there is no guarantee that a representation conforms to a particular Internet Media Type that is indicated by the URI string
Recap: A representation is

- The result of applying a service request upon a resource
- What the server determines to be the state of the resource
  - Parameters: time, space, request parameters
- A package
  - Metadata about the request, server actions, agent
  - Data (payload) in a specific Internet Media Type (MIME)
- The entity processed by a web agent (browser, crawler)
  - Agents such as crawlers make extensive use of metadata (e.g. last-modified)
- The entity that is the source of links
  - `<a href="http://google.com">`
Refined View of The Web Architecture

Warning: overuse of content negotiation can be bad for the web’s health

Time Berners-Lee - “Cool URIs”
Recap: http

- http defines an interface for interaction with a resource identified by an URI
- Presumes a reliable underlying transport protocol guaranteeing in-order delivery of requests and responses
  - by default TCP/IP with port:80 unless client is configured otherwise (e.g. proxy server)
Recap: http Verbs

• Retrieve a representation of a resource: GET
• Create a new resource: PUT and get a new URI, POST and specify a new URI
• Modify an existing resource: PUT to an existing URI
• Delete an existing resource: DELETE
• Get metadata about an existing resource: HEAD
• See which verbs a resource understands: OPTIONS
http session:
sequence of request-response

- an HTTP client initiates a request
- it uses DNS to resolve domain name
- it establishes a TCP connection to a particular port (typically 80) on a host (e.g. google.com)
- an HTTP Server listening on that port waits for a client request message
- upon receiving the request, the server sends back a status line (e.g., "HTTP/1.1 200 OK") and a message of its own (body, error message, some other information)
http session example

dhcp103-45:~ theresa$velden$ curl -v http://www.infosci.cornell.edu/Courses/info4302/2012fa/

* About to connect() to www.infosci.cornell.edu port 80 (#0)
* Trying 128.84.97.36... connected
* Connected to www.infosci.cornell.edu (128.84.97.36) port 80 (#0)
> GET /Courses/info4302/2012fa/ HTTP/1.1
> User-Agent: curl/7.19.7 (universal-apple-darwin10.0) libcurl/7.19.7 OpenSSL/0.9.8r zlib/1.2.3
> Host: www.infosci.cornell.edu
> Accept: */*
>
< HTTP/1.1 200 OK
< Connection: close
< Date: Wed, 05 Sep 2012 22:52:09 GMT
< Content-Type: text/html
< Server: Microsoft-IIS/6.0
< X-Powered-By: PHP/4.4.0
< MicrosoftOfficeWebServer: 5.0_Pub
< X-Powered-By: ASP.NET
<
<!doctype html>
.
.
.
* Closing connection #0
http request

Start line:
- Consists of method, path, version, e.g.
  GET /Courses/info4302/2012fa/ HTTP/1.1

Header fields:
- The HTTP/1.1 protocol version requires a Host: field
  Host: www.infosci.cornell.edu
- Many others: list of header fields at

Optional body content
http response

< HTTP/1.1 200 OK
< Connection: close
< Date: Wed, 05 Sep 2012 22:52:09 GMT
< Content-Type: text/html
< Server: Microsoft-IIS/6.0
< X-Powered-By: PHP/4.4.0
< MicrosoftOfficeWebServer: 5.0_Pub
< X-Powered-By: ASP.NET
<
<!doctype html>

Start line:
• Consists of HTTP version, status code and reason phrase
  HTTP/1.1 200 OK

Header fields, e.g.:
  Content-Type: text/html

Many others: list of header fields at

Content, e.g.
<br>
http Connection

HTTP/1.1 200 OK
Connection: close
Date: Wed, 05 Sep 2012 19:02:53 GMT
Content-Type: text/html
Server: Microsoft-IIS/6.0
X-Powered-By: PHP/4.4.0
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET

dhcp103-45:~ theresavelden$ curl --head http://www.infosci.cornell.edu/Courses/info4302/2012fa/default.php#main
HTTP/1.1 400 Bad Request
Connection: Keep-Alive
Content-Length: 34
Date: Wed, 05 Sep 2012 19:03:03 GMT
Content-Type: text/html

Default behavior in HTTP 1.1
Web Architecture

ADVANCED TOPICS
Web Forms and Content Negotiation?

- **Forms** enable interactions with web resources that may result in new resources (addressable or non-addressable) or change the state of a resource (reflected in a changed representation)

- Content negotiation is about providing an alternative (equivalent) representation of a web resource in response to a GET request
Web Forms and Content Negotiation?

Static Interface and State Transitions

1. POST http://restbucks.com/order
   - new order (create resource)

2. POST http://restbucks.com/order/1234
   - update order (only if state is at "payment expected")

3. DELETE http://restbucks.com/order/1234
   - order cancelled (only if state is at "payment expected")

4. PUT https://restbucks.com/payment/1234
   - Payment accepted

5. DELETE http://restbucks.com/order/1234
   - Barista prepared order

6. GET http://restbucks.com/order/1234
   - order received (only if state is at "ready")
   - return latest representation of the resource

REST in Practice – Webber, Parastatidis, Robinson
URI Encoding

• URL encoding converts characters into a format that can be transmitted over the Internet
  – i.e. ascii ("American Standard Code for Information Interchange", 128 characters)

• http URIs can contain non-ascii characters, but need to be escaped when communicated over the internet e.g. in an http request

http://www.w3schools.com/tags/ref_urlencode.asp
Fragments

- A URI reference identifies a target resource
- A user agent resolves the URI reference to its absolute form to obtain target URI
- Target URI excludes a potential fragment identifier component
- Fragment identifier components are reserved for client side processing
Fragment Identifier

Discussions

• http://www.w3.org/DesignIssues/Fragment.html

• http://www.w3.org/QA/2011/05/hash_uriis.html
HANDS-ON Web Architecture
Useful Debugging Tools

• Browser add-ons: Developer View
• Command line tool: curl
Web Developer View: Example 1

Unsing Safari: Develop > Show Web Inspector)

Request URL:
http://www.cs.cornell.edu/~tvelden/
Analysis:

- Processing and rendering of retrieved resource representations is determined by user agent.
- Web browser interprets URI references in HTML potentially triggering a sequence of resource requests:
  - Value of the `href` attribute:
    - `<a href="http://schema.org/">schema.org</a>`
    - `<link href="apple-touch-icon.png"/>`
  - Value of `src` attribute:
    - `<img class="foto" src="picture.png"/>`
Web Developer View: Example 2

Using Safari: Develop > Show Web Inspector

• Request URL:

  http://www.cs.cornell.edu/~tvelden
Request URL http://www.cs.cornell.edu/%7Etvelden

![Theresa's Homepage](http://www.cs.cornell.edu/%7Etvelden/)

---

**Theresa Velden**

Research Fellow  
School of Information  
University of Michigan

<table>
<thead>
<tr>
<th>Name Path</th>
<th>Method</th>
<th>Status</th>
<th>Type</th>
<th>Size Transferred</th>
<th>Time Latency</th>
<th>Timeline</th>
</tr>
</thead>
<tbody>
<tr>
<td>%7Etvelden</td>
<td>GET</td>
<td>301</td>
<td>text/...</td>
<td>0B</td>
<td>45ms</td>
<td>82ms</td>
</tr>
<tr>
<td>/%7Etvelden/</td>
<td>GET</td>
<td>200</td>
<td>text/...</td>
<td>2.63KB</td>
<td>24ms</td>
<td>123ms</td>
</tr>
<tr>
<td>home.css</td>
<td>GET</td>
<td>200</td>
<td>text/...</td>
<td>13.57K</td>
<td>37ms</td>
<td>164ms</td>
</tr>
<tr>
<td>picture-bw5.png</td>
<td>GET</td>
<td>200</td>
<td>image/...</td>
<td>7.58KB</td>
<td>180ms</td>
<td>205ms</td>
</tr>
</tbody>
</table>
Analysis

• URI equivalence via “301 Moved Permanently”

• Documentation of http requests/response incomplete
curl

curl –v URI
  – Verbose, shows entire request and response
  – GET is default verb used in request

curl --head URI
  – Only resource metadata: HEAD verb
curl: Example 1(a)

curl -v http://www.infosci.cornell.edu/Courses/info4302/2012fa/

As seen before:
• TCP/IP part
• HTTP GET Request (Method, path, Protocol Version)
• HTTP Response Headers (HTTP/1.1 200 OK)
• HTTP Response Body
• TCP/IP Connection closed

Note: no secondary web resources retrieved; curl acts not like a browser does but executes only single URI dereferencing
curl: Example 1(b)

curl -v http://www.infosci.cornell.edu/Courses/info4302/2012fa/picture-bw5.png

As seen before:
• TCP/IP part
• HTTP GET Request (Method, path, Protocol Version)
• HTTP Response Headers (HTTP/1.1 200 OK)
• HTTP Response Body
• TCP/IP Connection closed

Body is now a png file (not rendered)
Comments

• Curl option –v shows record of entire interaction
  – TCP IP, HTTP Request, HTTP Response, TCP/IP

• no secondary web resources retrieved
  – curl acts not like a browser does but executes only single http request to dereference URI

• Curl default: GET request
• Curl -- head/-l: HEAD request
Content Negotiation:
Option to express client preferences

- **Accept**: specifies certain media type responses that are acceptable to the client (e.g., application/json, application/atom+xml)
  - **Accept-Charset**: indicates in which character sets the response should be represented that are acceptable to the client
  - **Accept-Encoding**: restricts the content encodings. Can be used to indicated compression (compress;q=0.5, gzip;q=1.0)
  - **Accept-Language**: restricts the set of natural languages that are preferred as a response to the request

- **User-Agent** header can also be used for content negotiation (e.g., serve different representation for mobile devices)
curl: Example 2

Language Negotiation

curl -v --head --header "Accept-Language: fr"
http://www.google.com
curl: Example 3 (a)

Format Negotiation

- curl -I --head -H "Accept: text/html"
  http://vocab.deri.ie/dcat

- curl -I --head -H "Accept: application/rdf+xml"
  http://vocab.deri.ie/dcat
curl: Example 3 (a)

Format Negotiation

dhcp103-45:~ theresavelden$ curl -I --head -H "Accept: text/html"
http://vocab.deri.ie/dcat

HTTP/1.1 200 OK
Date: Thu, 06 Sep 2012 12:23:02 GMT
Server: Apache/2.2.9 (Debian) PHP/5.2.6-1+lenny4 with Suhosin-Patch
X-Powered-By: PHP/5.2.6-1+lenny4
Set-Cookie: SESS972ddc872c5c8bd5c673d923b3fb5ebf=b1fc21cc1d55dcbeeb8dba8499363f5e;
expires=Sat, 29 Sep 2012 15:56:22 GMT; path=/; domain=.vocab.deri.ie
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Last-Modified: Thu, 06 Sep 2012 12:23:02 GMT
Cache-Control: store, no-cache, must-revalidate
Cache-Control: post-check=0, pre-check=0
Vary: Accept,Accept-Encoding
Content-Location: http://vocab.deri.ie/dcat.html
Access-Control-Allow-Origin: *
Content-Type: text/html; charset=utf-8
curl: Example 3 (a)

Format Negotiation

dhcp103-45:~ theresavelden$ curl -I --head -H "Accept: application/rdf+xml"
http://vocab.deri.ie/dcat

HTTP/1.1 200 OK
Date: Thu, 06 Sep 2012 12:23:06 GMT
Server: Apache/2.2.9 (Debian) PHP/5.2.6-1+lenny4 with Suhosin-Patch
X-Powered-By: PHP/5.2.6-1+lenny4
Set-Cookie: SESS972ddc872c5c8bd5c673d923b3fb5ebf=abd3c6d239034c89f19fc57212ca4f54;
expires=Sat, 29 Sep 2012 15:56:26 GMT; path=/; domain=.vocab.deri.ie
Expires: Sun, 19 Nov 1978 05:00:00 GMT
Last-Modified: Thu, 06 Sep 2012 12:23:06 GMT
Cache-Control: store, no-cache, must-revalidate
Cache-Control: post-check=0, pre-check=0
Vary: Accept,Accept-Encoding
Content-Location: http://vocab.deri.ie/dcat.rdf
Access-Control-Allow-Origin: *
Content-Type: application/rdf+xml; charset=utf-8
Content Negotiation: Importance of Client Preferences

• Quality values (qvalue) are short floating point numbers to indicate the relative importance (weight) of various negotiation parameters
  – 0 is the minimum value (= "not acceptable")
  – 1 is the maximum value
curl: Example 3 (b)

Format Negotiation w relative importance

- curl -I --head -H "Accept: application/rdf+xml;q=0.2" -H "Accept: text/html;q=0.2" http://vocab.deri.ie/dcat

- curl -I --head -H "Accept: application/rdf+xml;q=0.5" -H "Accept: text/html;q=0.2" http://vocab.deri.ie/dcat
Comments

• Format negotiation: final decision with server
Curl Example 4

*Conditional GET*

- curl --head -H "If-Modified-Since: Sun, 02 Sep 2012 00:00:00 GMT" [http://www.cs.cornell.edu/~tvelden/](http://www.cs.cornell.edu/~tvelden/)
Curl Example 4

dhcp103-45:~ theresavelden$ curl --head -H "If-Modified-Since: Sun, 02 Sep 2012 00:00:00 GMT" http://www.cs.cornell.edu/~tvelden/

HTTP/1.1 304 Not Modified
Connection: Keep-Alive
Date: Thu, 06 Sep 2012 12:38:04 GMT
Content-Location: http://webpub.cs.cornell.edu/~tvelden/index.html
ETag: "03c662acd80cd1:5897"
Server: Microsoft-IIS/6.0
Last-Modified: Thu, 23 Aug 2012 01:18:13 GMT
Accept-Ranges: bytes
MicrosoftOfficeWebServer: 5.0_Pub
X-Powered-By: ASP.NET
Comments

• eTag field:
  – provides the current value of the entity tag for the requested variant
curl: Example 5

- curl -I -H "Accept: application/rdf+xml"
  [URL]
- curl -I -H "Accept: text/html"
  [URL]
curl: Example 5


Response:
HTTP/1.1 303 See Other
Date: Thu, 06 Sep 2012 15:45:04 GMT
Server: Jetty(6.1.1)
Location: http://www4.wiwiss.fu-berlin.de/dblp/data/person/103481
Content-Type: text/plain
curl: Example 5

- curl -I -H "Accept: text/html"
  http://www4.wiwiss.fu-berlin.de/dblp/resource/person/103481

Response:
HTTP/1.1 303 See Other
Date: Thu, 06 Sep 2012 15:48:39 GMT
Server: Jetty(6.1.1)
Location: http://www4.wiwiss.fu-berlin.de/dblp/page/person/103481
Content-Type: text/plain
Homework 1

WEB SCIENCE / LINKED DATA
Ethical Principles of Web Science

- Decentralization
- Openness
- Fairness
Linked Data

- Challenge & Solution
Resources

• Tutorials http://www.w3schools.com/
• cURL http://curl.haxx.se/
Next Week:

- Third component of Web Architecture:
  - Standardized Document Formats (HTML, XML)