

HTTP 6

Recall

HTTP equiv

Keywords and description

Refresh -- similar to 301. Lower quality, but easy to do
.shtml

HEAD Request

HEAD / HTTP/1.1

Like GET, but just gets the header.

Handy to see if a resource is there or what type it is, but without actually
downloading it

Caching

Copy of recent requests kept nearer to client. First try to satisfy request out of
cache.

Browser keeps local cache in memory + on local disk.

Proxy cache -- one or more machines between the client and the server. The
browser is set to make its requests to the proxy (see "proxy" in your browser
prefs). The proxy will re-issue the request to the real "origin" server if needed.

Cache Effects

Advantages

Helps with back button (browser cache). Ideally, hitting the back button,
could just show the state as it appeared earlier very quickly -- get
everything from local cache, and make no network traffic. The reality is
more complex, since some content is marked as "no-cache".

Helps common web items, e.g. little graphic buttons on yahoo main page,
cached for all users

Gives better performance -- a cache on your local LAN is much faster than the
Internet at large

Reduces the "upstream" bandwidth bill of an ISP

Disadvantages

More complex -- has a reputation of creating tech support hassles, so some
severs turn it off

Could screw up unique pages, such as a view of your shopping cart, that only
make sense for one user, and which change over time. The system needs to
know to not cache a page like that.

Screws up "hit counting" on the origin server. (solvable)

As HTTP becomes as common as TV, some sort of caching structure may be
critical to keep performance ok

Cache Technology

HTTP includes many features, directives, etc. to facilitate caching. Clients and servers can indicate: the last mod time of an element, whether something should be cached and for how long

The most common optimization is the "If-modified-since" GET that checks if the cached version is up to date. Saves bandwidth with images. (One reason web designers err towards image-heavy pages. The images load up the designers cache, so they don't see how terribly slow the page is on the first-hit for users at large.)

Cache server can communicate "number of hits" back to the origin server with the "meter" header

Proxy

A machine between the client and the server

Cache is most common application

1. Could be used to filter content
2. Could be used to contact the origin sever in a special way -- say over an encrypted channel. Or say for example, if www.maosucks.com is blocked from inside China. The proxy contacts the blocked site and relays the content back to the client.
3. Could re-write the content in some way

1. Formal Proxy

Browser has a formal "proxy" preference -- put in the machine:port, and the browser will make all its requests to the proxy, instead of the origin server

2. URL Proxy Filtering

The URL points to the proxy machine, and the desired URL is passed as an argument on that URL, typically as a query string

<http://www.degraeve.com/scripts/babel2/babel.cgi?d=piglatin&url=http://www.yahoo.com>

The browser makes its request in the usual way to the proxy server, it turns around makes the real request, and then forwards the results back to the client

Web Proxy Example

I run our old webecho.pl server on elaine0.stanford.edu:8181

I configure my browser to use elaine0.stanford.edu:8181 as its "HTTP proxy"

Then, a request for www.yahoo.com, gets sent to the proxy with the following form...

Here's what we got from the client:

```
GET http://www.yahoo.com/ HTTP/1.1
Host: www.yahoo.com
User-Agent: Mozilla/5.0 (Macintosh; U; PPC; en-US; rv:1.0rc1) Gecko/20020417
Accept:
text/xml,application/xml,application/xhtml+xml,text/html;q=0.9,text/plain;q=0.8,video/xmng,image/png,image/jpeg,image/gif;q=0.2,text/css,*/*;q=0.1
Accept-Language: en-us, en;q=0.50
Accept-Encoding: gzip, deflate, compress;q=0.9
Accept-Charset: ISO-8859-1, utf-8;q=0.66, *;q=0.66
Keep-Alive: 300
Proxy-Connection: keep-alive
```

Pig-Latin Example

What does our course page look like translated into pig-latin?

<http://www.degraeve.com/scripts/babel2/babel.cgi?d=piglatin&url=http://www.stanford.edu/class/cs193i/>

CS193i Internetyay Echnologiestay

Elcomeway otay CS193i orfay Ingspray 2002.

....

It translates the HTML, and also updates all the href= URLs to also go through the proxy. Here is our handouts link...

```
<a
href="http://www.degraeve.com/scripts/babel2/babel.cgi?d=piglatin&url=http://www.stanford.edu/class/handouts013/">Andoutshay</a>
```

There are many such URL proxy filter "services" on the net -- Swedish chef, ... This is a form of server-side CGI program, which is our next topic.