Reducing Network Overhead in Web Applications

Adam Derewecki
given for EECS @ CWRU

$ whoami

- Graduated BSE Comp Eng 2007
  - Research Assistant for Networking Dept
  - TA for ENGR131 (Fall '05, '06)
- IBM Software Engineer, 2008 - 2009
  - DB2 XML Storage and Runtime teams
- Yelp Backend Engineer, 2009 - Present

It's only 30KB, will it make a diff?

- Hell I'll just cache it!
    - YDN caching experiment*
        - 40-60% of user have an empty cache
        - 20% of all page views are empty cache
- Packets are 1460 bytes
    - Two packets for 1461 byte file
    - TCP slow start penalty for >2 packets
- Backbones are fast, last mile isn't
    - EDGE: Best case 150ms, real world average 500ms

- In short: yes, it matters.

* http://developer.yahoo.com/blogs/ydn/posts/2009/10/a_engineers_gui/
http://yuiblog.com/blog/2007/01/04/performance-research-part-2/

AJAX: Don't Repeat Yourself

- Asynchronous JavaScript and XML
  - (doesn't require XML payload)
- Lets browser talk to server via JavaScript after page has finished loading for client
- Why AJAX?
  - load new pages without full page loads
    - search results
    - Google Instant Search is an extreme example
  - don't load more than you have to - lazy loading
    - good for slow components (mint.com)
    - optional components
      - "below the fold"
  - BigPipe
    - http://www.facebook.com/notes/facebook-engineering/bigpipe-pipelining-web-pages-for-high-performance/389414033919

Below the Fold

BBC

Play.com

Amazon.co.uk

New York Times

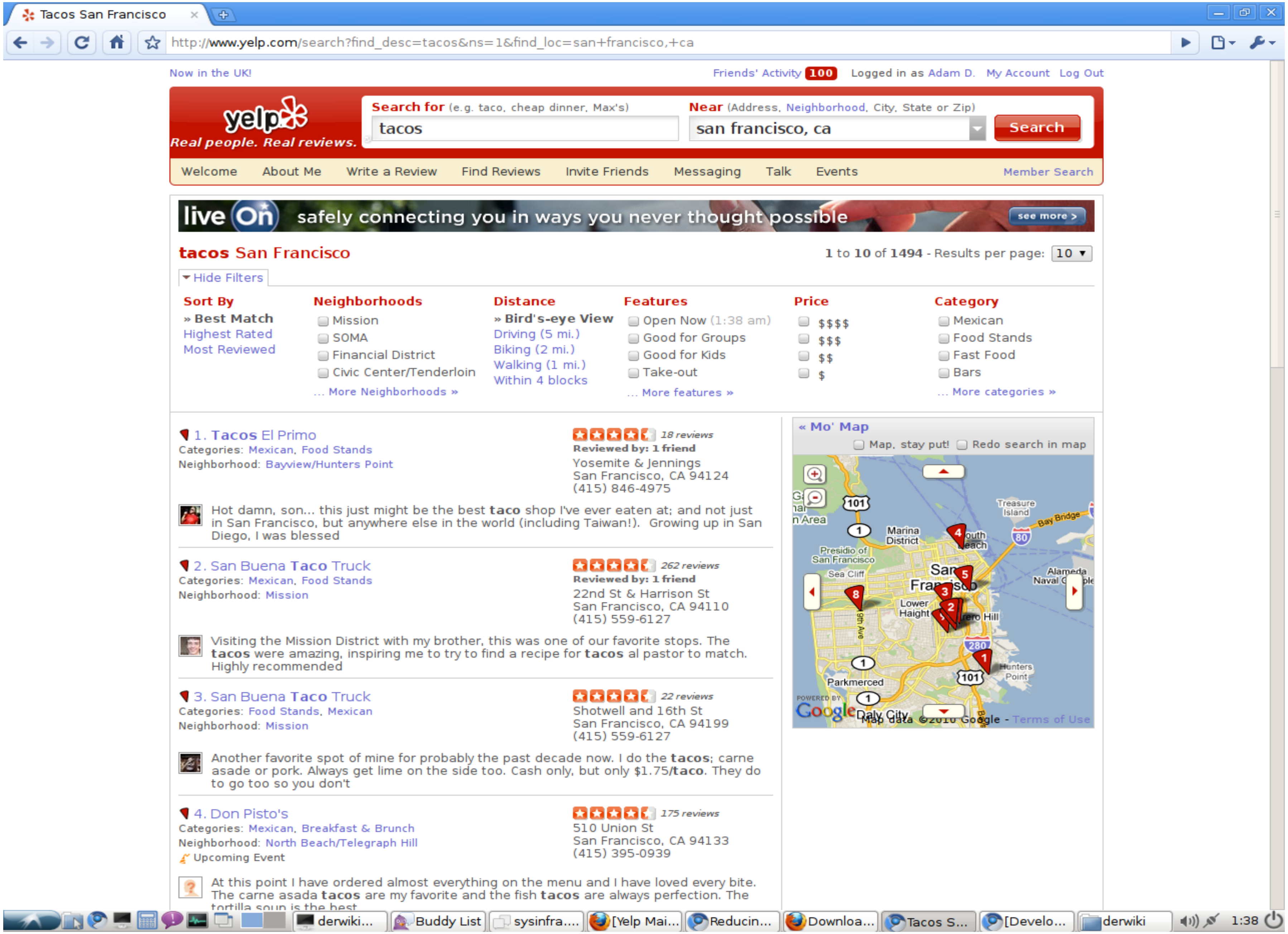
Rolling your own XMLHttpRequest

```
// instantiate an "AJAX" object
var xml = new XMLHttpRequest();

// set up the connection
xml.open('GET', '/some/url', true);
        ^      ^          Async (true) or sync (false)*
        |      |URL: must be on the same server for security reasons
        |protocol: GET or POST. Remember, GET has size restrictions

// send the data payload
xml.send(data); // slam it out there
// Note: While it's AJA-XML, the data sent does not need to be XML. //       We commonly use
JSON and HTML snippets.
// Note: xml.send() is also valid (for example, tracking)
```

Why use synchronous?

- Sometimes a blocking call from a page is useful
- Expire a token or some other final page action before bouncing

AJAX: Yelp Search Results



Tacos San Francisco

http://www.yelp.com/search?find_desc=tacos&ns=1&find_loc=san+francisco,+ca

Now in the UK!

Friends' Activity 100 Logged in as Adam D. My Account Log Out

yelp Real people. Real reviews.

Search for (e.g. taco, cheap dinner, Max's) tacos

Near (Address, Neighborhood, City, State or Zip) san francisco, ca

Search

Welcome About Me Write a Review Find Reviews Invite Friends Messaging Talk Events Member Search

live On safely connecting you in ways you never thought possible see more >

tacos San Francisco

1 to 10 of 1494 - Results per page: 10 ▾

▾ Hide Filters

Sort By
- » **Best Match**
- Highest Rated
- Most Reviewed

Neighborhoods
- □ Mission
- □ SOMA
- □ Financial District
- □ Civic Center/Tenderloin
- ... More Neighborhoods »

Distance
- » **Bird's-eye View**
- Driving (5 mi.)
- Biking (2 mi.)
- Walking (1 mi.)
- Within 4 blocks

Features
- □ Open Now (1:38 am)
- □ Good for Groups
- □ Good for Kids
- □ Take-out
- ... More features »

Price
- □ $$$$
- □ $$$
- □ $$
- □ $

Category
- □ Mexican
- □ Food Stands
- □ Fast Food
- □ Bars
- ... More categories »

1. **Tacos** El Primo — 18 reviews
Reviewed by: 1 friend
Categories: Mexican, Food Stands
Neighborhood: Bayview/Hunters Point
Yosemite & Jennings
San Francisco, CA 94124
(415) 846-4975

Hot damn, son... this just might be the best **taco** shop I've ever eaten at; and not just in San Francisco, but anywhere else in the world (including Taiwan!). Growing up in San Diego, I was blessed

2. San Buena **Taco** Truck — 262 reviews
Reviewed by: 1 friend
Categories: Mexican, Food Stands
Neighborhood: Mission
22nd St & Harrison St
San Francisco, CA 94110
(415) 559-6127

Visiting the Mission District with my brother, this was one of our favorite stops. The **tacos** were amazing, inspiring me to try to find a recipe for **tacos** al pastor to match. Highly recommended

3. San Buena **Taco** Truck — 22 reviews
Categories: Food Stands, Mexican
Neighborhood: Mission
Shotwell and 16th St
San Francisco, CA 94199
(415) 559-6127

Another favorite spot of mine for probably the past decade now. I do the **tacos**; carne asade or pork. Always get lime on the side too. Cash only, but only $1.75/**taco**. They do to go too so you don't

4. Don Pisto's — 175 reviews
Categories: Mexican, Breakfast & Brunch
Neighborhood: North Beach/Telegraph Hill
Upcoming Event
510 Union St
San Francisco, CA 94133
(415) 395-0939

At this point I have ordered almost everything on the menu and I have loved every bite. The carne asada **tacos** are my favorite and the fish **tacos** are always perfection. The tortilla soup is the best

« Mo' Map

□ Map, stay put! □ Redo search in map

AJAX: Yelp's 'the hub'

- AJAX heavy web pages have JS that get messy
    - tangled webs of chained callbacks
- 'the hub'
    - mediocre name, useful technology
    - Create a model of all the interactive elements on the page
    - Publish messages to the hub when things finish
    - Subscribe to do things when messages are published
    - sure beats daisy chains of callbacks
    - http://github.com/Yelp/thehub/blob/master/js/thehub.js

AJAX: Yelp's PubSubHub

```
deals.queueDeal = function(event) {
    deals.doFunctionOnSaveSuccess('admin.deals.queue_status', deals.QueueStatus, function() {
        yelp.hub.publish('admin.deals.queue_status', new deals.QueueStatus('IN_PROGRESS'));
        deals.queueRequest = new yelp.async.ajax.Request('/deal/queue_deal_to_send/' + deals.
getDealId(), {
            method: 'post',
            onSuccess: function(resp) {
                var respJSON = resp.responseJSON;
                if (resp.responseText) {
                    console.log(resp.responseText);
                }
                if (respJSON.success) {
                    yelp.hub.publish('admin.deals.queue_status', new deals.QueueStatus
('COMPLETE_SUCCESS'));
                } else {
                    var error_msgs = deals.collectErrorMessages(respJSON);
                    yelp.hub.publish('admin.deals.queue_status', new deals.QueueStatus('COMPLETE_FAIL',
error_msgs));
                }
            },
            onFailure: function(resp) {
                yelp.hub.publish('admin.deals.save_status', new deals.QueueStatus('COMPLETE_FAIL', 'Show
this to an engineer - code ' + resp.status));
            }
        });
    });
};
```

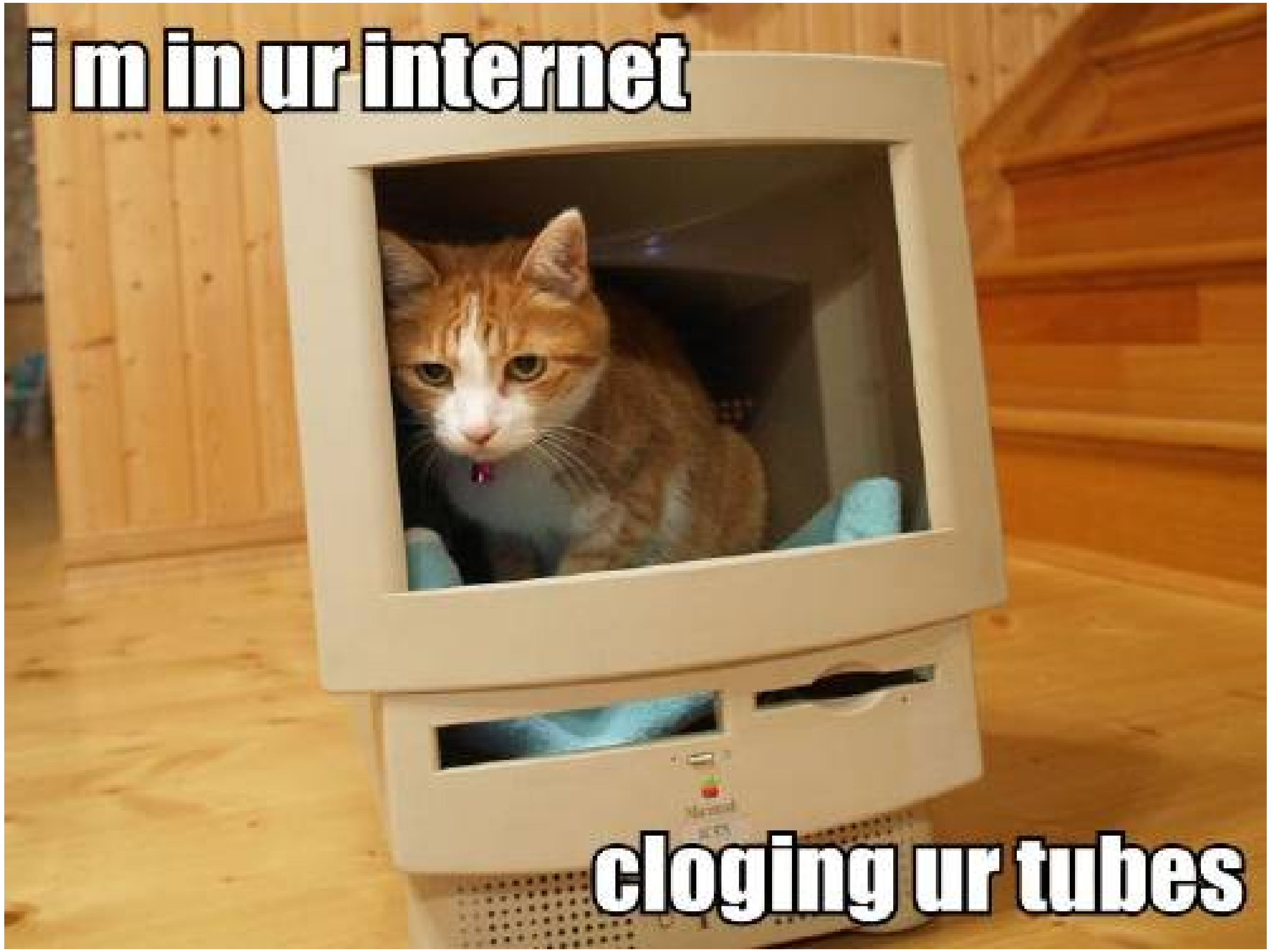
AJAX: Yelp's PubSubHub

```
deals.QueueStatus = function(state, errorMessage) {
    var states = {'IN_PROGRESS': {'success': null, 'message': 'Queuing...'},
              'COMPLETE_FAIL': {'success': false, 'message': 'Failed to add to queue: ' + errorMessage},
              'COMPLETE_TIMEOUT': {'success': false, 'message': 'Failed to add to queue: save timed out.'},
              'COMPLETE_SUCCESS': {'success': true, 'message': 'Deal queued!'}};
    this.type = 'QUEUE_STATUS';
    yelp.pages.admin.deals.initStatus(this, states, state);
};

deals.observeButtons = function() {
    $(document.body).delegate('click', '#queue_deal', deals.queueDeal)
    .delegate('click', '#dequeue_deal', deals.dequeueDeal)
};

deals.hubSubscribe = function() {
    yelp.hub.subscribe('admin.deals.queue_status', deals.updatePageStatus);
};
```

One more real quick AJAX thing

- Staged Loading
  - Load core JS in one file
  - Have JS load additional, non-essential JS after the main page has loaded

- Inline JS
  - JavaScript is single threaded, blocking
  - Hiding JS in a comment and eval'ing it
  - http://www.stevesouders.com/blog/2009/12/07/downloading-javascript-as-strings/


i m in ur internet
cloging ur tubes

Longpolling

- AJAX is cool and all -- but how do you push dynamic content to the client?
- You can't -- but you can!
    1. Have the browser open a long running AJAX request to a webserver
    2. The web server holds the connection until it has something to say back to the client
    3. After the server pushes a notification back to the client, the client starts a new long running AJAX request to the server to wait for more updates

- This is how push services (or at least Yelp's) are done
    - "I'm still here" every 15 minutes
    - High water mark, 100,000 active connections

Longpolling: Server considerations

- Traditional polling methods don't keep open connections, server doesn't need to keep as many at once
- Per-process/per-thread models tend to have a lot of extra weight for each additional connection
- Solution: epoll!
  - asynchronous sockets (non-blocking)
  - single event loop instead of process/thread models
  - each iteration of the loop looks at a set of file descriptors (sockets) you have interest in, and handles a bit of reading or writing on each of them
  - Tornado and node.js are two popular frameworks that let you handle I/O this way

epoll server

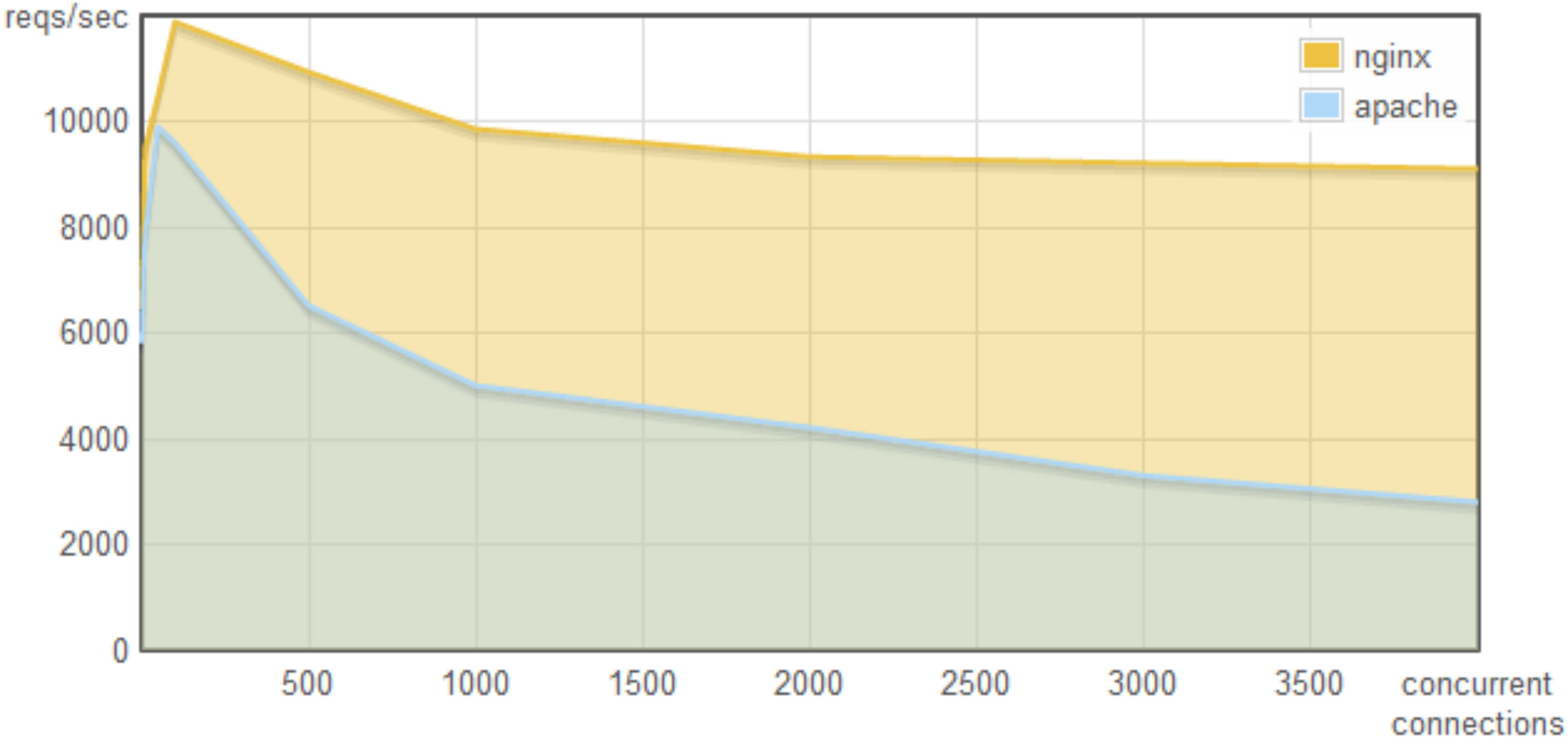
```
import socket, select
serversocket = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
serversocket.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
serversocket.bind(('0.0.0.0', 9090))
serversocket.listen(1)
serversocket.setblocking(0) # nonblocking
epoll = select.epoll()
epoll.register(serversocket.fileno(), select.EPOLLIN)
default_resp = str(range(1000))
cxns, reqs, resps = {}, {}, {}
try:
    while True:
        events = epoll.poll(1)
        for fileno, ev in events:
            if fileno == serversocket.fileno():
                # new cxn
                conn, addr = serversocket.accept()
                conn.setblocking(0)
                epoll.register(conn.fileno(), select.EPOLLIN)
                cxns[conn.fileno() = conn
                reqs[conn.fileno()] = ''
                resps[conn.fileno()] = default_resp
            elif ev & select.EPOLLIN:
                reqs[fileno] += cxns[fileno].recv(1024)
                if any(eol in reqs[fileno] for eol in (b'\n\n', b'\n\r\n')):
                    epoll.modify(fileno, select.EPOLLOUT)
                    print reqs[fileno].decode()[:-2]
            elif ev & select.EPOLLOUT:
                bytes_written = cxns[fileno].send(resps[fileno])
                resps[fileno] = resps[fileno][bytes_written:]
                if len(resps[fileno]) == 0:
                    epoll.modify(fileno, 0)
                    cxns[fileno].shutdown(socket.SHUT_RDWR)
            elif ev & select.EPOLLHUP:
                epoll.unregister(fileno)
                cxns[fileno].close()
                del cxns[fileno]
finally:
    epoll.unregister(serversocket.fileno())
    epoll.close(); serversocket.close()
```

Teh Intertubes
They is clogged
ICANHASCHEEZBURGER.COM

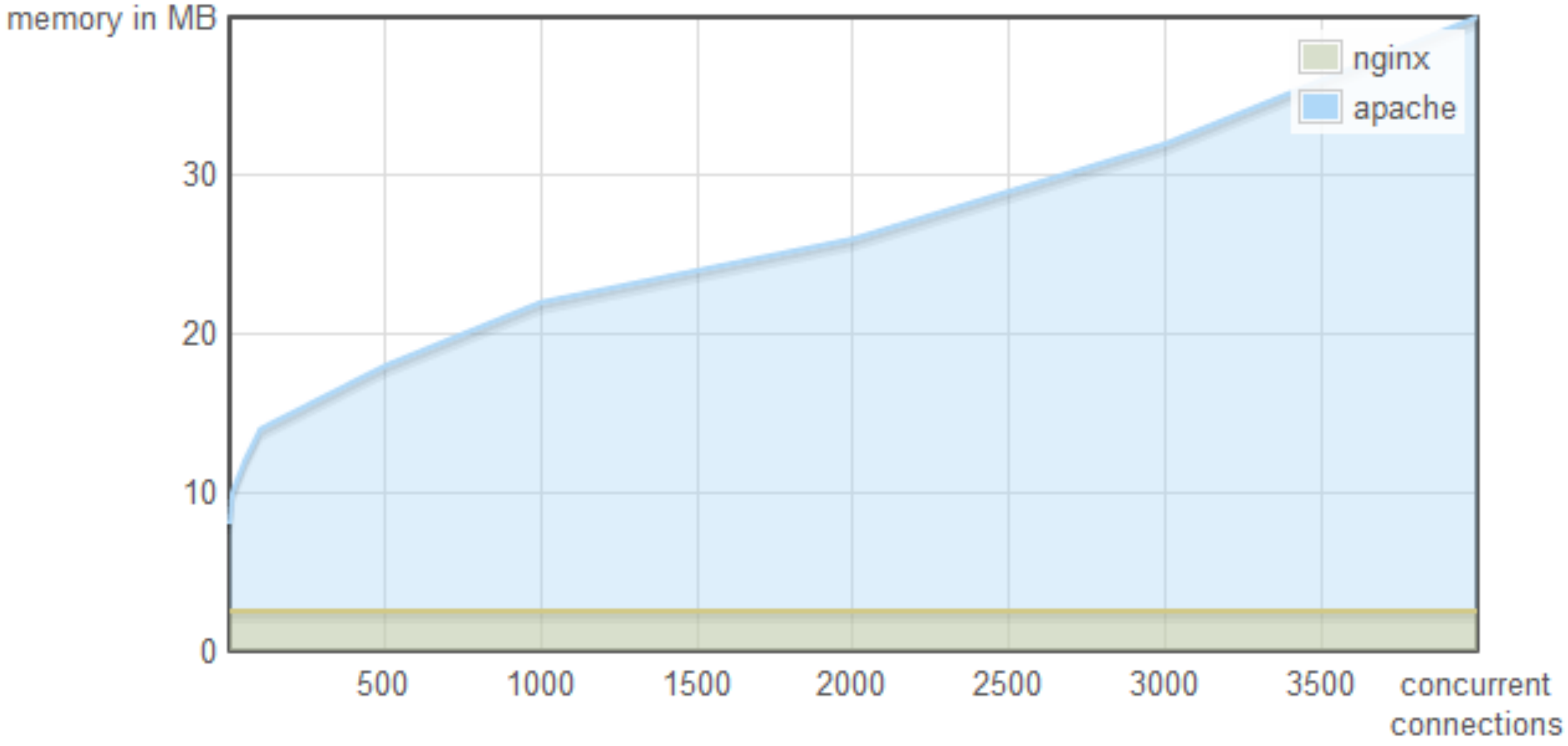
epoll resources

- http://scotdoyle.com/python-epoll-howto.html
    - previous slide borrowed heavily from scot's code
- http://sheddingbikes.com/posts/1280829388.html
    - Zed Shaw, of internet fame
- Frameworks
    - http://github.com/facebook/tornado (Python)
        - developed at FriendFeed (including two CWRU grads!)
    - http://nodejs.org/ (server-side JS)
    - http://github.com/eventmachine/eventmachine (Ruby)

nginx (epoll) vs Apache: speed



nginx vs. Apache: memory



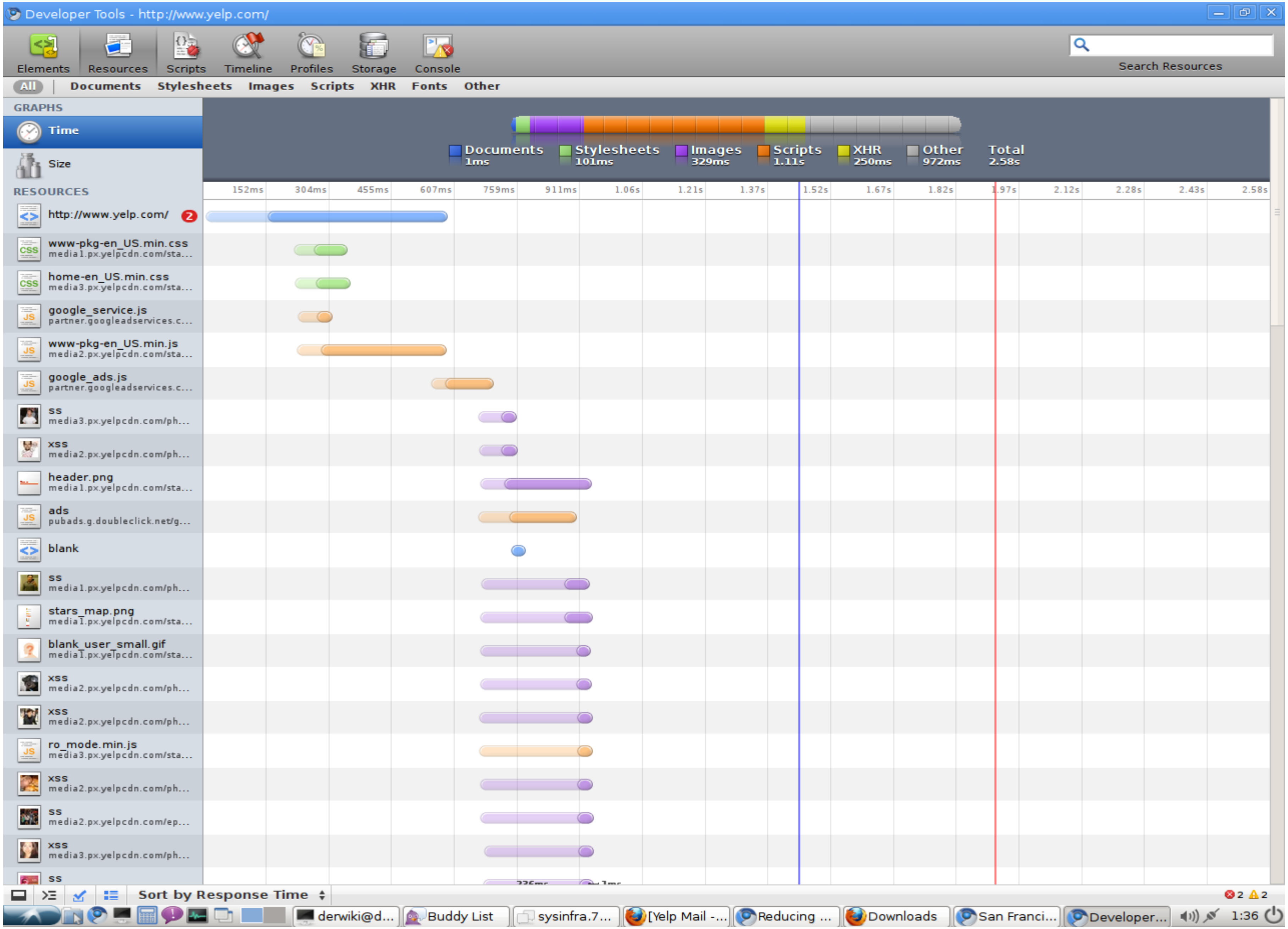
- Ouch, that hurts!

Longpolling: easier on the network

- 100,000 concurrent connections
- Average idle time of 5 minutes
- Poll every 30 seconds
    - 3333 requests per second
- Longpoll, only take action every 5 minutes
    - 333 requests per second

- Longpolling combined with epoll reduces network traffic and also scaled better than traditional poll-thread/process systems

Measuring front end performance



Developer Tools - http://www.yelp.com/

http://www.yelp.com/
www-pkg-en_US.min.css media1.px.yelpcdn.com/sta...
home-en_US.min.css media3.px.yelpcdn.com/sta...
google_service.js partner.googleadservices.c...
www-pkg-en_US.min.js media2.px.yelpcdn.com/sta...
google_ads.js partner.googleadservices.c...
ss media3.px.yelpcdn.com/ph...
xss media2.px.yelpcdn.com/ph...
header.png media1.px.yelpcdn.com/sta...
ads pubads.g.doubleclick.net/g...
blank
ss media1.px.yelpcdn.com/ph...
stars_map.png media1.px.yelpcdn.com/sta...
blank_user_small.gif media1.px.yelpcdn.com/sta...
xss media2.px.yelpcdn.com/ph...
xss media2.px.yelpcdn.com/ph...
ro_mode.min.js media3.px.yelpcdn.com/sta...
xss media2.px.yelpcdn.com/ph...
ss media2.px.yelpcdn.com/ep...
xss media3.px.yelpcdn.com/ph...
ss

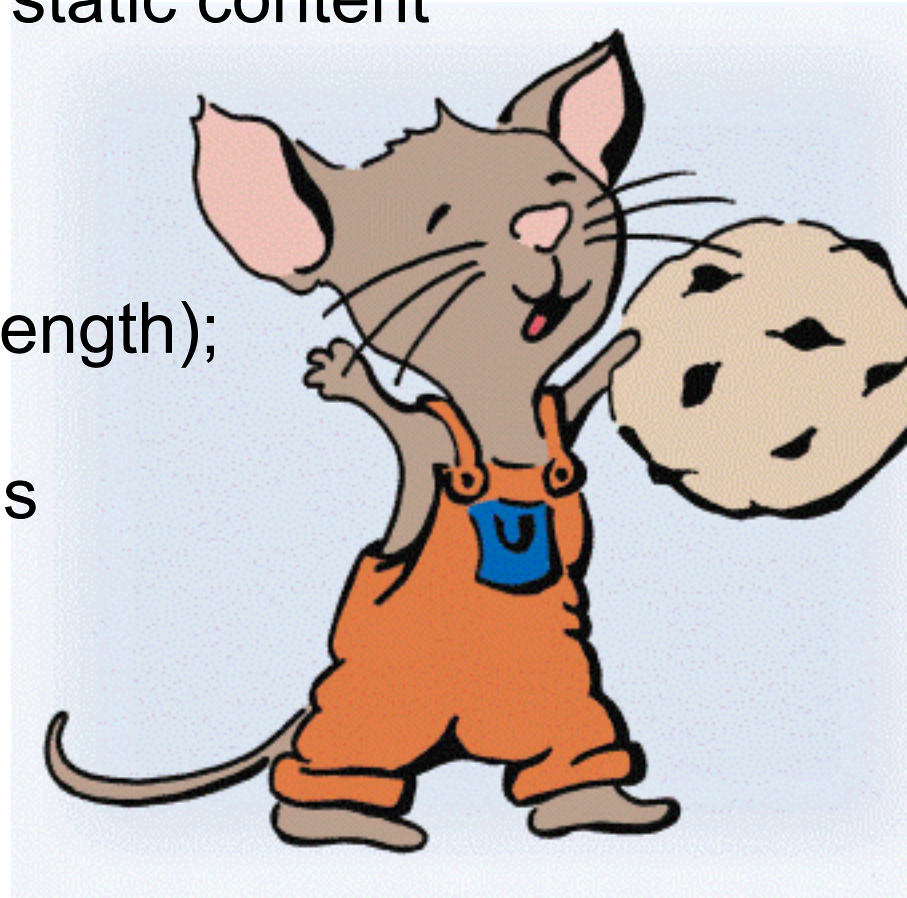
Making your page load faster

- Gzip. Gzip. Gzip.
    - http://paulbuchheit.blogspot.com/2009/04/make-your-site-faster-and-cheaper-to.html
- Reduce number of assets you serve
- ... and the size of them
    - spriting images
    - combining JS/CSS
    - minifying JS/CSS
- Reduce number of DNS lookups
- JavaScript loading is a blocking operation
    - only load essential JS when you need to
- Performance tools
    - Chrome, Safari have built in tools
    - Firebug (www.getfirebug.com)
    - Yahoo YSlow (developer.yahoo.com/yslow)

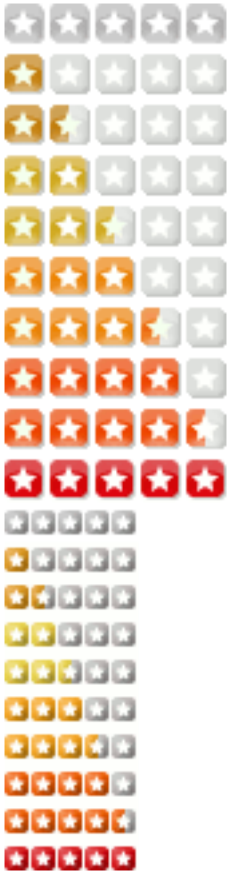
Cookies

- If you give an [asset] a cookie...
    - He'll send it along every request to your domain
    - Like real cookies, those calories (bytes) add up
    - Use another domain name for static content
        - www.yelpcdn.com

- Test sites yourself
- javascript:alert(document.cookie.length);
    - www.yelp.com - 1985 bytes
    - www.facebook.com - 450 bytes
    - www.twitter.com - 1123 bytes
    - www.google.com - 401 bytes



Spriting



- Combine 20 images used frequently into one
- Reduces number of HTTP requests
- Caching (during this session at least)
- CSS to crop to what you need

```
img.stars_3 {
    clip: rect(95px 83px 110px 0px);
    top: -95px;
}
```

The Final Slide

- Any of this sound interesting? We're hiring!
    - www.yelp.com/careers
    - 50 person engineering team
    - Open source
        - github.com/yelp
        - engineeringblog.yelp.com
    - Located in downtown San Francisco