WebCloud: Enabling more direct content exchange between web clients

Fangfei Zhou†§, Liang Zhang†§, Eric J. Franco†§, Richard Revis†, Alan Mislove†, Ravi Sundaram‡

†Northeastern University  ‡Jandrell Pearson & Revis Ltd  †§Student

Motivation

Previously (pre-2005), web content — in the form of web pages, images, audio, and video — was primarily created by a small minority of Media and corporations, and was delivered to a large audience of web users.

Thus, network workload was
- From the “center” (media)
- To the “edge” (users)

Recent trends such as the rise in popularity of online social networking; the ease of content creation using digital devices like smart phones, cameras, and camcorders; and the ubiquity of Internet access have democratized content creation.

Today, Internet users are creating content that makes up a significant fraction of Internet traffic.

How is new content being delivered?

Traditional "centralized" architectures:
- Akamai, Limelight
- Facebook serves most of its own content

Mismatch between infrastructure and workload:
- Workload is naturally decentralized
  - Every Facebook upload goes via CA
  - Works today with photos, but videos?
- Can we move towards more decentralized web distribution architectures?

Design

We propose WebCloud, a content distribution system designed to support the workloads present in existing online social networking websites.

Key insight: leverage the local storage and bandwidth resources of the users themselves to help serve content to other users. (i.e., p2p content distribution for the web)

Saves bandwidth: due to the geographic locality that often exists between friends in online social networks, content exchanges often stay in the ISP

Scalable: each additional user provides additional resources

Implementation

Goal: make it work with today’s sites, browsers

Idea: Introduce a middlebox to allow browsers to communicate

To build WebCloud, we need two components:

Client-side changes
- Need to turn web browser into server
- Implement WebCloud in JavaScript
- Add it to the site’s pages

Middlebox
- Add redirector proxies in each ISP:
  - Like Akamai, but stores no content
- Clients connect to proxy:
  - Inform of locally stored content
- Client request content from proxy:
  - Proxy checks other local clients
  - If Found:
    - fetches content from other client
  - Otherwise:
    - fetches content from original site

Evaluation

Is there additional latency?

<table>
<thead>
<tr>
<th>Accessed from</th>
<th>Served from</th>
</tr>
</thead>
<tbody>
<tr>
<td>LAN</td>
<td>Cable modem</td>
</tr>
<tr>
<td>668 ms</td>
<td>63 ms</td>
</tr>
<tr>
<td>690 ms</td>
<td>153 ms</td>
</tr>
</tbody>
</table>

No, in fact, always faster than getting from Facebook

What WebCloud hit rate can we expect?

![Hit rate graph]

Does it work with today’s browsers/sites?

- Deployed WebCloud within Northeastern College of Computer Science
  - 17 users for 10 days
  - Total of 2,060 photos viewed
- Works from Firefox, Safari, Chrome
  - Average browser could store 56 photos

Mobile Users

- Implemented a prototype WebCloud app for iOS
  - Tested with 3,513 requests when connected via 3G
  - iPhone’s battery lasted an estimated 23 hours
  - Even under the heaviest workload, the most-loaded user uploaded a total of 71.9MB, when the average user uploaded only 2.0MB

<table>
<thead>
<tr>
<th>Number of Photos Uploaded per User</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
</tr>
</tbody>
</table>

Simulated 1 week deployment
- Between 23% and 58% hit rate