Content Delivery is Changing

- Thirst for data continues to increase (more data & users)
- New types of data have emerged - audio, video
- People use new means to exchange this data

The result - Internet is now seeing a mixture of old and new content delivery systems:
- Conventional Web servers and Web clients
- CDNs: Akamai, Digital Island, Speedera
- P2Ps: Kazaa, Gnutella, Napster, AudioGalaxy
Goals of this Work

- Explore and characterize content delivery in the new Internet
  - World Wide Web, Akamai CDN, Kazaa & Gnutella P2P

- High-level questions:
  - What is the impact of these systems on the Internet?
  - What are the characteristics of the new delivery systems?

Methodology

- Collect anonymized traces at UW border routers
  - Extension of existing infrastructure [Wolman & Voelker]

- Identify and log HTTP traffic:
  - Akamai – ports 80, 8080, 443 and served by Akamai server
  - WWW – ports 80, 8080, 443 (exclude Akamai)
  - Gnutella – ports 6346, 6347
  - Kazaa – port 1214

- In this talk, results are based on a 9-day trace
  - May 28th - June 6th, 2002
  - 500 million Xactions, 20TB of HTTP data
Question 1:

What is the bandwidth impact of the new content delivery systems?

Where has all the bandwidth gone?

Web = 14% of TCP; P2P = 43% of TCP
P2P now dominates Web in bandwidth consumed!!!
### Bandwidth consumed by UW servers

<table>
<thead>
<tr>
<th></th>
<th>WWW inbound</th>
<th>WWW outbound</th>
<th>Kazaa inbound</th>
<th>Kazaa outbound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bytes Xferred</td>
<td>1.51TB</td>
<td>3.02TB</td>
<td>1.78TB</td>
<td>13.6TB</td>
</tr>
<tr>
<td>Unique objects</td>
<td>72,818,997</td>
<td>3,412,647</td>
<td>111,437</td>
<td>166,442</td>
</tr>
<tr>
<td>Clients</td>
<td>39,285</td>
<td>1,231,308</td>
<td>4,644</td>
<td>611,005</td>
</tr>
<tr>
<td>Servers</td>
<td>403,087</td>
<td>1,463</td>
<td>281,026</td>
<td>3,888</td>
</tr>
</tbody>
</table>

P2P has diurnal cycle, like the Web
P2P peaks later at night
Summary 1: (what is the bandwidth impact?)

- P2P traffic is the largest bandwidth consumer
  - Three times bigger than Web, at UW
- Uploads significantly exceed downloads
  - at least within UW-like environments

Question 2:

- What are the bytes carrying?
Specialized Content Delivery Systems

Web = text + images
Kazaa = video
Akamai = images
Gnutella = video + audio

HTTP Bytes: Now vs. Then

<table>
<thead>
<tr>
<th>Content Type</th>
<th>May 1999</th>
<th>May 2002</th>
</tr>
</thead>
<tbody>
<tr>
<td>multipart/mixed-replace</td>
<td>0.9</td>
<td>0.8</td>
</tr>
<tr>
<td>audio/x-pn-realaudio</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>application/pdf</td>
<td>0.9</td>
<td>0.8</td>
</tr>
<tr>
<td>application/x-machcore</td>
<td>0.15</td>
<td>0.12</td>
</tr>
<tr>
<td>video/mpeg</td>
<td>0.7</td>
<td>0.8</td>
</tr>
<tr>
<td>No Content Type</td>
<td>1.9</td>
<td>1.9</td>
</tr>
<tr>
<td>text/plain</td>
<td>2.6</td>
<td>2.4</td>
</tr>
<tr>
<td>video/quicktime</td>
<td>0.9</td>
<td>0.9</td>
</tr>
<tr>
<td>video/x-sgi</td>
<td>5.7</td>
<td>5.9</td>
</tr>
<tr>
<td>application/octet-stream</td>
<td>8.4</td>
<td>11.3</td>
</tr>
<tr>
<td>image/jpeg</td>
<td>18.9</td>
<td>18.9</td>
</tr>
<tr>
<td>image/gif</td>
<td>20.9</td>
<td>25.0</td>
</tr>
<tr>
<td>image/tiff</td>
<td></td>
<td></td>
</tr>
<tr>
<td>text/html</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Content Types Ordered by Size

- 1.7% video/x-msvideo
- 2.7% text/plain
- 4.8% MP3
- 7.1% application/octet-stream
- 7.5% GIF
- 8.8% JPG
- 9.8% HASHED
- 11.3% MPG
- 14.6% HTML
- 18.0% AVI

(b) By Bytes

% All Bytes

0% 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%

May 2002

HTML

AVI

MPG

JPEG

HASHED

GIF

MP3

application/octet-stream

image/jpeg

image/tiff

image/gif

text/plain

text/html

video/x-sgi

video/quicktime

video/mpeg

application/octet-stream

image/jpeg

image/tiff

image/gif

text/html

(b) By Bytes

% All Bytes

0% 10% 20% 30% 40% 50% 60% 70% 80% 90% 100%
Summary 2: (what are the bytes?)

- Systems have become specialized in the type of content they deliver
- P2P = video + audio bytes
- It’s not the music – a single video is 200 MP3s
- Web = text + image bytes

Question 3:

- How do workload characteristics differ in the new delivery systems, compared to the old ones?
Object size

![Object Size CDF Graph]

Most bandwidth consuming object

<table>
<thead>
<tr>
<th></th>
<th>WWW (inbound)</th>
<th>Kazaa(inbound)</th>
<th>Kazaa(outbound)</th>
</tr>
</thead>
<tbody>
<tr>
<td>object size</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(MB)</td>
<td></td>
<td>object size</td>
<td></td>
</tr>
<tr>
<td>requests</td>
<td></td>
<td>(MB)</td>
<td></td>
</tr>
<tr>
<td># clients</td>
<td>0.009</td>
<td>694.4</td>
<td>696.9</td>
</tr>
<tr>
<td># servers</td>
<td>1.4 mil.</td>
<td>20</td>
<td>397</td>
</tr>
<tr>
<td># clients</td>
<td></td>
<td>164</td>
<td>1</td>
</tr>
<tr>
<td># servers</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Top 5 bandwidth consuming objects

<table>
<thead>
<tr>
<th>WWW (inbound)</th>
<th>Kazaa(inbound)</th>
<th>Kazaa(outbound)</th>
</tr>
</thead>
<tbody>
<tr>
<td>object size (MB)</td>
<td># requests</td>
<td>object size (MB)</td>
</tr>
<tr>
<td>1</td>
<td>0.009</td>
<td>1.4 mil.</td>
</tr>
<tr>
<td>2</td>
<td>0.002</td>
<td>3 mil.</td>
</tr>
<tr>
<td>3</td>
<td>333</td>
<td>21</td>
</tr>
<tr>
<td>4</td>
<td>0.005</td>
<td>1.4 mil.</td>
</tr>
<tr>
<td>5</td>
<td>2.23</td>
<td>1,457</td>
</tr>
</tbody>
</table>

Few UW clients and servers exchange big, popular objects

Server Availability

Most P2P servers are unresponsive
Summary 3: (how do characteristics differ?)

- P2P objects are three orders of magnitude bigger than Web objects
- P2P servers are largely unresponsive (but it’s not clear whether it matters to users)

Question 4:

- How is bandwidth use distributed among objects, clients, servers?
### Object Popularity at UW

<table>
<thead>
<tr>
<th>Bytes Xferred</th>
<th>WWW</th>
<th>Kazaa</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>inbound</td>
<td>outbound</td>
</tr>
<tr>
<td>Unique objects</td>
<td>72,818,997</td>
<td>3,412,647</td>
</tr>
<tr>
<td>Clients</td>
<td>39,285</td>
<td>1,231,308</td>
</tr>
<tr>
<td>Servers</td>
<td>403,087</td>
<td>1,463</td>
</tr>
</tbody>
</table>

1,000 Kazaa objects (out of 111K) responsible for 50% of bytes transferred
<table>
<thead>
<tr>
<th>Bytes Xferred</th>
<th>WWW inbound</th>
<th>WWW outbound</th>
<th>Kazaa inbound</th>
<th>Kazaa outbound</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unique objects</td>
<td>1.51TB</td>
<td>3.02TB</td>
<td>1.78TB</td>
<td>13.6TB</td>
</tr>
<tr>
<td>Clients</td>
<td>72,818,997</td>
<td>3,412,647</td>
<td>111,437</td>
<td>166,442</td>
</tr>
<tr>
<td>Servers</td>
<td>39,285</td>
<td>1,231,308</td>
<td>4,644</td>
<td>611,005</td>
</tr>
<tr>
<td></td>
<td>403,087</td>
<td>1,463</td>
<td>281,026</td>
<td>3,888</td>
</tr>
</tbody>
</table>

**UW Clients**

Small number of clients account for large portion of traffic

- 200 Web + Akamai = 13% of traffic
- 200 Kazaa = 50% of traffic (20% of all incoming HTTP)
### UW Servers

<table>
<thead>
<tr>
<th></th>
<th>WWW</th>
<th></th>
<th>Kazaa</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>inbound</td>
<td>outbound</td>
<td>inbound</td>
<td>outbound</td>
</tr>
<tr>
<td>Bytes Xferred</td>
<td>1.51TB</td>
<td>3.02TB</td>
<td>1.78TB</td>
<td>13.6TB</td>
</tr>
<tr>
<td>Unique objects</td>
<td>72,818,997</td>
<td>3,412,647</td>
<td>111,437</td>
<td>166,442</td>
</tr>
<tr>
<td>Clients</td>
<td>39,285</td>
<td>1,231,308</td>
<td>4,644</td>
<td>611,005</td>
</tr>
<tr>
<td>Servers</td>
<td>403,087</td>
<td>1,463</td>
<td>281,026</td>
<td>3,888</td>
</tr>
</tbody>
</table>

#### Top Bandwidth Consuming UW Servers

- **WWW**
- **Kazaa**
- Gnutella

20 servers supply 80% of Web traffic
334 servers (10%) supply 80% of Kazaa traffic

Load is not uniformly spread across peers!
Summary 4: (how is bandwidth distributed?)

- A very small number of P2P objects, clients, and servers consume an enormous fraction of traffic

Question 5:

- Could P2P data be cached?
Caching Methodology

- Simulation of an ideal cache:
  - Infinite storage, bandwidth, CPU, etc...
  - All objects are cached forever
    - P2P objects are immutable, no dynamic data
- Evaluated:
  - Both inbound and outbound P2P data cacheability
  - Objects are big:
    - Byte hit rate more important than object hit rate
- Gives an upper bound on potential for caching

---

P2P Data Cacheability

Potential for caching is high in both directions
UW inbound cache needs a month to warm-up
Summary 5: (is it cacheable?)

- Large potential for data cacheability in P2P
- P2P caches take a long time to warm-up

Conclusions

- The Internet is being used in a completely different way:
  - P2P traffic is the largest bandwidth consumer
  - Peers consume significant bandwidth in both directions
- P2P objects are 1,000 times larger than Web objects
- Small number of huge objects are responsible for an enormous fraction of bytes transferred
  - 300 Kazaa objects consumed 5.6TB bandwidth!
- Few P2P peers are causing much of the traffic
- Large potential for caching in P2P
Final Thoughts...

- The average Kazaa peer consumes 90x more bandwidth than the average Web client
- Adding another 450 Kazaa peers is equivalent to doubling the entire UW web client population
- Is the current architecture of Kazaa scalable?