600.413 Topics in P2P Networked Systems

Week 4

Measurements

Andreas Terzis

Slides from Stefan Saroiu

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?

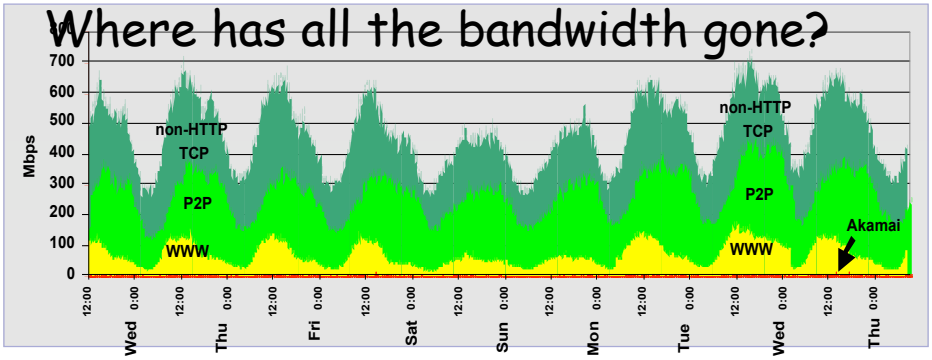
Methodology

- 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?

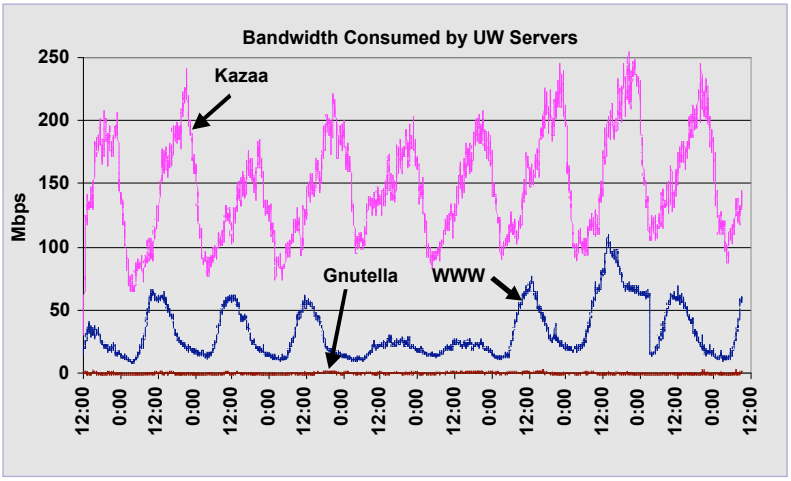
Where has all the bandwidth gone?



Web = 14% of TCP; P2P = 43% of TCP
P2P now dominates Web in bandwidth consumed!!!

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

Bandwidth consumed by UW servers



P2P has diurnal cycle, like the Web
P2P peaks later at night

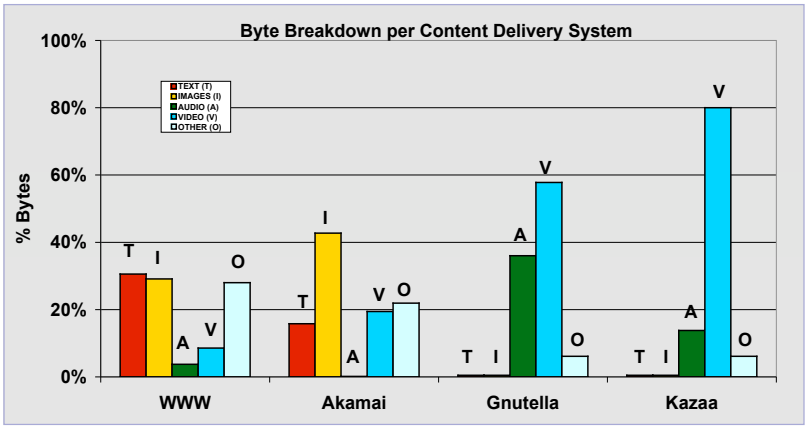
Summary 1: (what is the bandwidth impact?)

- 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

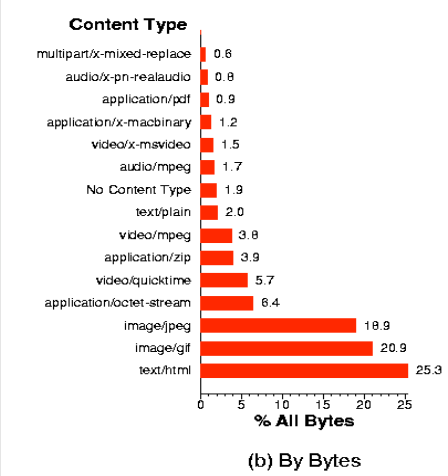
Akamai = images

Kazaa = video

Gnutella = video + audio

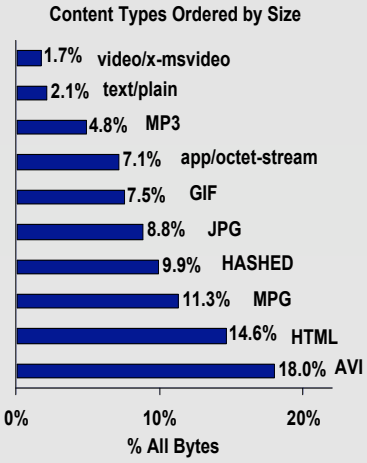
HTTP Bytes: Now vs. Then

May 1999



(b) By Bytes

May 2002



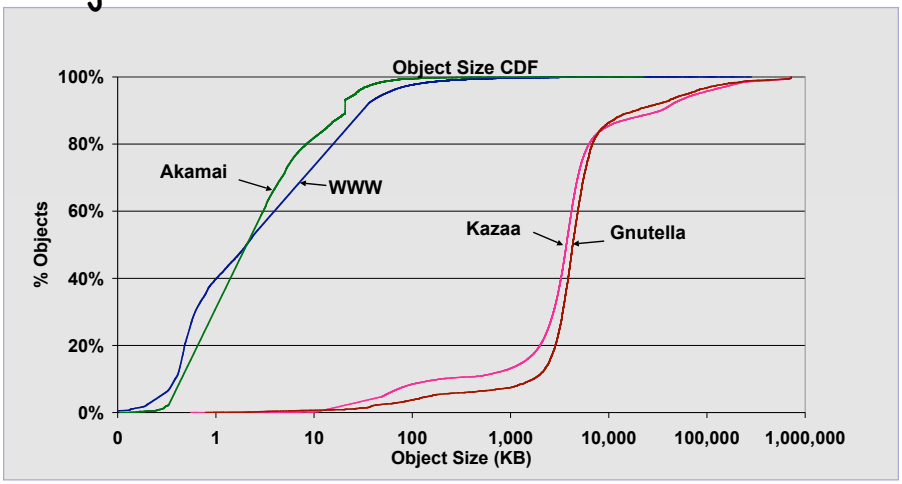
Summary 2: (what are the bytes?)

- 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



Most bandwidth consuming object

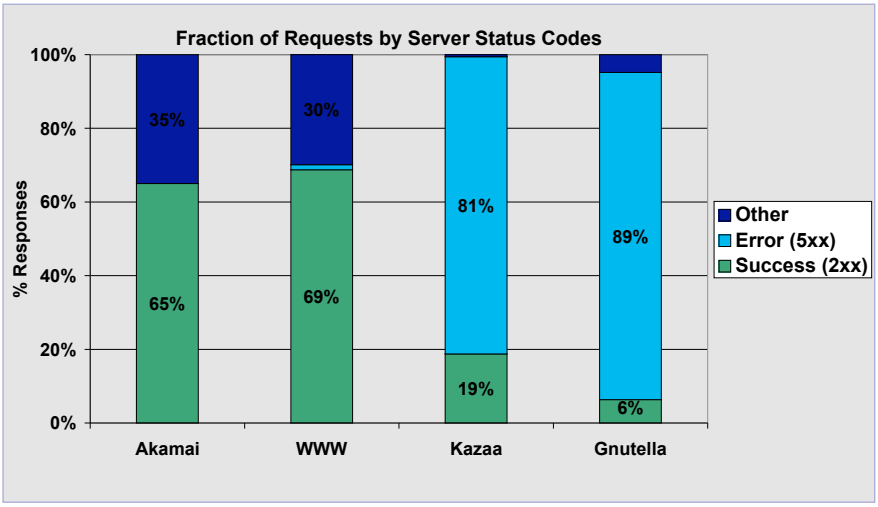
	WWW (inbound)		*Kazaa(inbound)*			*Kazaa(outbound)*		
	object size (MB)	*# requests*	*object size (MB)*	*# clients*	*# servers*	*object size (MB)*	*# clients*	*# servers*
1	0.009	1.4 mil.	694.4	20	164	696.9	397	1

Top 5 bandwidth consuming objects

	WWW (inbound)		*Kazaa(inbound)*			*Kazaa(outbound)*		
	object size (MB)	*# requests*	*object size (MB)*	*# clients*	*# servers*	*object size (MB)*	*# clients*	*# servers*
1	0.009	1.4 mil.	694.4	20	164	696.9	397	1
2	0.002	3 mil.	702.2	14	91	699.2	1000	4
3	333	21	690.3	22	83	699.1	390	10
4	0.005	1.4 mil.	775.6	16	105	700.9	558	2
5	2.23	1,457	698.1	14	74	634.2	540	1

Few UW clients and servers exchange big, popular objects

Server Availability



Most P2P servers are unresponsive

Summary 3: (how do characteristics differ?)

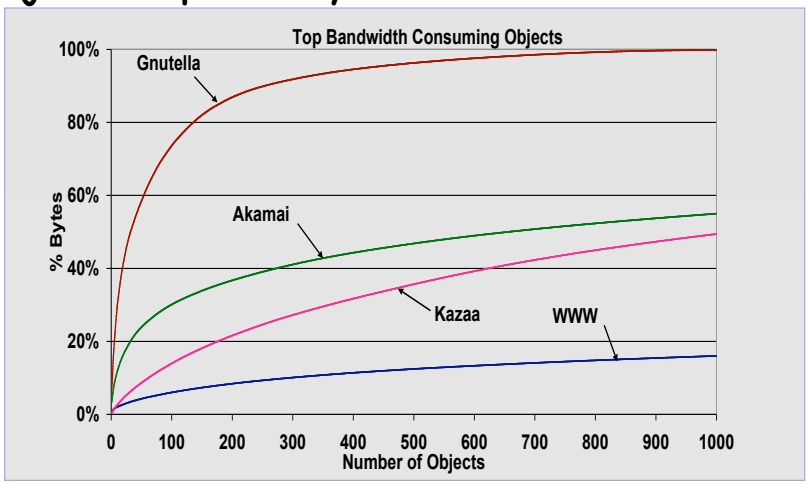
- P2P objects are three orders of magnitude bigger than Web objects
- P2P servers are largely unresponsive (but it's not clear whether it matters to users)

Question 4:

- How is bandwidth use distributed among objects, clients, servers?

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

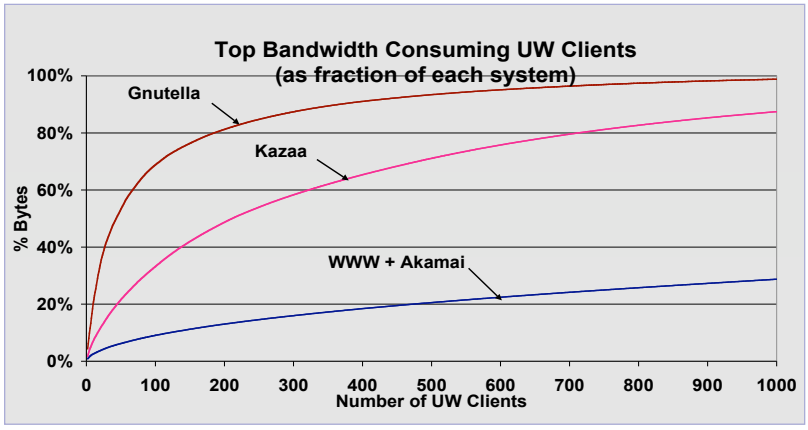
Object Popularity at UW



1,000 Kazaa objects (out of 111K) responsible for 50% of bytes transferred

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

UW Clients

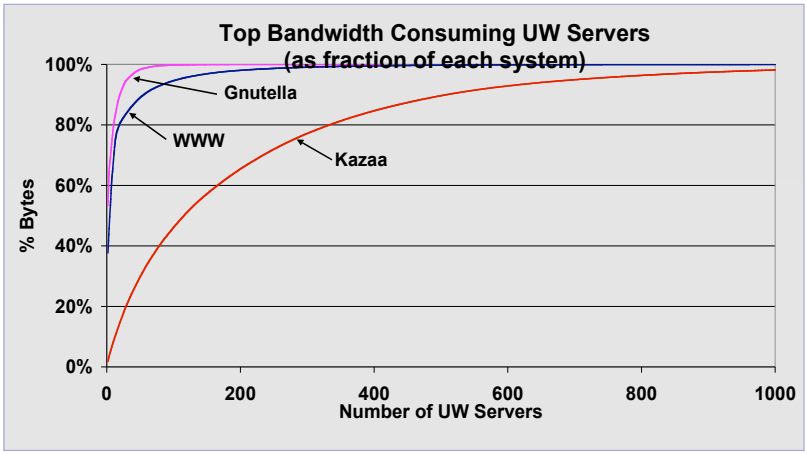


Small number of clients account for large portion of traffic

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

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

UW Servers



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

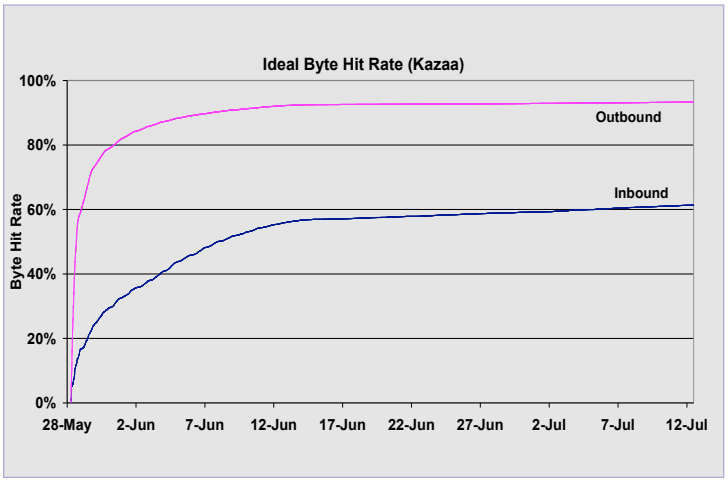
Question 5:

- 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?