An Analysis of Internet Content Delivery Systems

Stefan Saroiu, Krishna Gummadi, Richard J. Dunn, Steven D. Gribble, and Henry M. Levy

Proceedings of the 5th Symposium on Operating Systems Design and Implementation

Boston, MA, December 9-11, 2002

Outline

- Introduction
- Overview
- Data Collection
- Data Summary
- Caching
- Conclusion

Introduction

- As complexity of data has increased, content delivery mechanisms have changed
- New systems, especially peer-to-peer, have affected the nature of Internet traffic
- Goal: Determine the effect of the latest types of content delivery systems on Internet traffic
  - Web vs. CDN vs. P2P

Introduction

What the authors did:

- Traced all incoming and outgoing traffic at the University of Washington (pop. 60,000) over nine days.
- Totals were 500 million transactions and 20 terabytes of data.
- Separated traffic between Akamai, Kazaa, Gnutella, and regular Web traffic (all use HTTP).

Overview of Content Delivery Systems

World-Wide Web:

- Most objects are 5-10KB
- Heavy-tailed distribution; some objects are very large while most of objects are small
- Increasing number of dynamic objects and total objects is reducing cache effectiveness
- The best existing or proposed caches have hit rates in the 40-50% range

Overview of Content Delivery Systems

Content Delivery Networks:

- Servers distributed throughout the Internet with replicated content
- Largest CDNs have servers at every major point of traffic
- Invisible to users
- CDNs generally carry only static and secondary content

Overview of Content Delivery Systems

Peer-to-Peer Systems:

- Collaborative content exchange systems
- Nodes act as both servers and clients
- Virtually all nodes are operated by end-users
- Most nodes are on slow, unreliable connections, unlike most content delivery systems

Overview of Content Delivery Systems

Peer-to-Peer Systems:

- Systems use one of three methods for content searching
  - Search requests are made directly between peers
  - Central server maintains a complete index
  - Hybrid system: some peers are "supernodes" and index content to nearby peers
- After content has been located, transfer commences between peer and one or more peers that host the content

Data Collection

Methodology

- Passive network monitoring used to collect traces of traffic data
- Switches connected to all campus backbones send data via a monitoring port to a monitoring host
- Kernel-level packet filter delivers packets to a user process for data collection

Data Collection

Distinguishing Traffic Types

- HTTP traffic
  - WWW and Akamai on ports 80, 8080, 443 (SSL)
  - Gnutella on ports 6346 and 6347
  - Kazaa on port 1214
  - Mystery Traffic
- Non-HTTP traffic
  - Any other TCP traffic: NNTP, SMTP and etc.
  - Kazaa and Gnutella search traffic
- Did not include UDP traffic in the analysis.
  - 97% of UW traffic is TCP

Data Collection

Traceability of P2P Traffic

- Kazaa's architecture is proprietary, but some inferences about its operation are made by observing traffic
- Kazaa peers can download objects in fragments, which complicates tracing
- Download requests from external peers seen in their trace are often for fragments rather than entire object

Data Summary

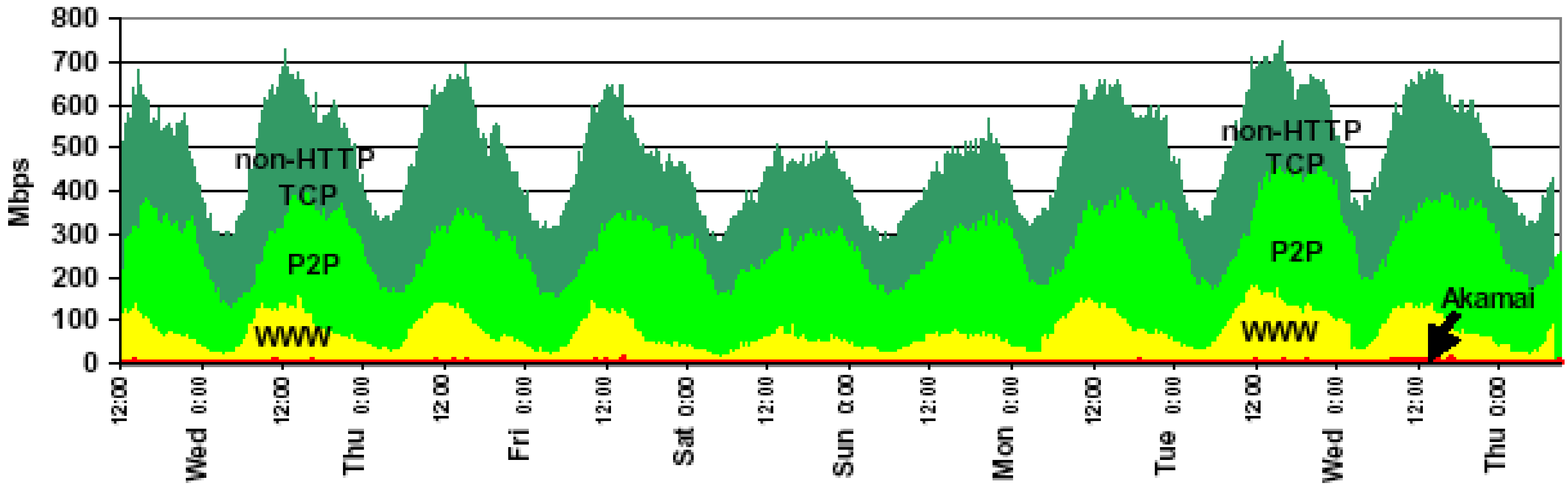
HTTP trace summary statistics

	WWW		Akamai		Kazaa		Gnutella	
	inbound	outbound	inbound	outbound	inbound	outbound	inbound	outbound
HTTP transactions	329,072,253	73,001,891	33,486,508	N/A	11,140,861	19,190,902	1,576,048	1,321,999
unique objects	72,818,997	3,412,647	1,558,852	N/A	111,437	166,442	5,274	2,092
clients	39,285	1,231,308	34,801	N/A	4,644	611,005	2,151	25,336
servers	403,087	9,821	350	N/A	281,026	3,888	20,582	412
bytes transferred	1.51 TB	3.02 TB	64.79 GB	N/A	1.78 TB	13.57 TB	28.76 GB	60.38 GB
median object size	1,976 B	4,646 B	2,001 B	N/A	3.75 MB	3.67 MB	4.26 MB	4.08 MB
mean object size	24,687 B	82,385 B	12,936 B	N/A	27.78 MB	19.07 MB	19.16 MB	9.78 MB

- Compare outbound vs. inbound data
- Compare the volume of data transfer for each system
- Compare the object sizes

Data Summary

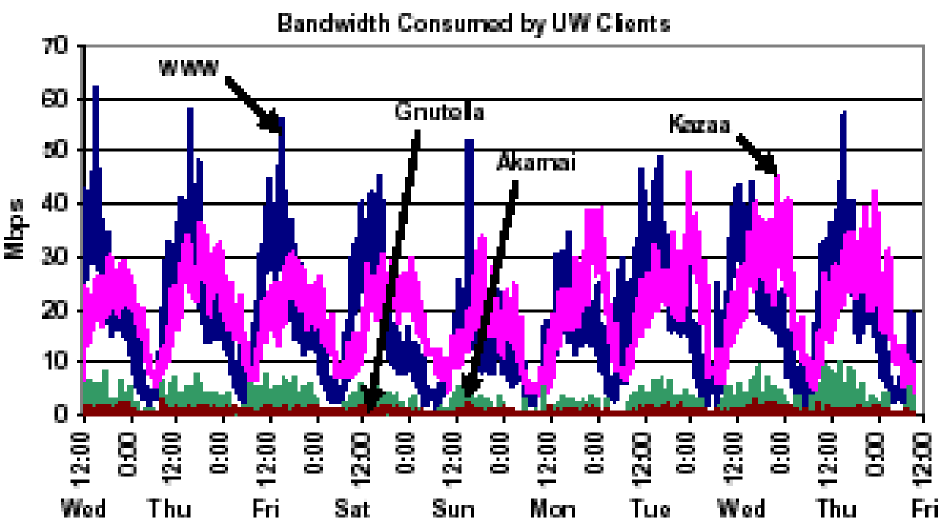
TCP bandwidth

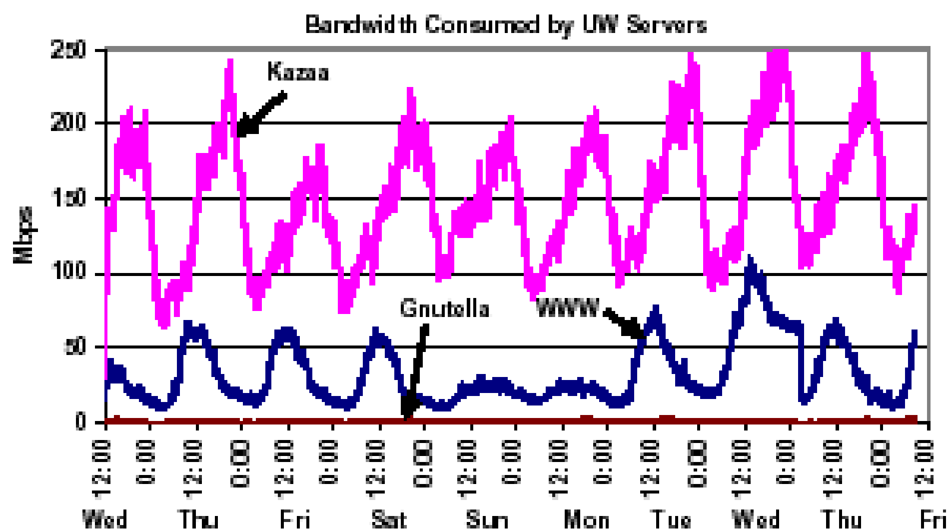


- Notice the pure volume of peer to peer traffic, compared to all TCP traffic (43% of all TCP)

Data Summary

Client and server TCP bandwidth

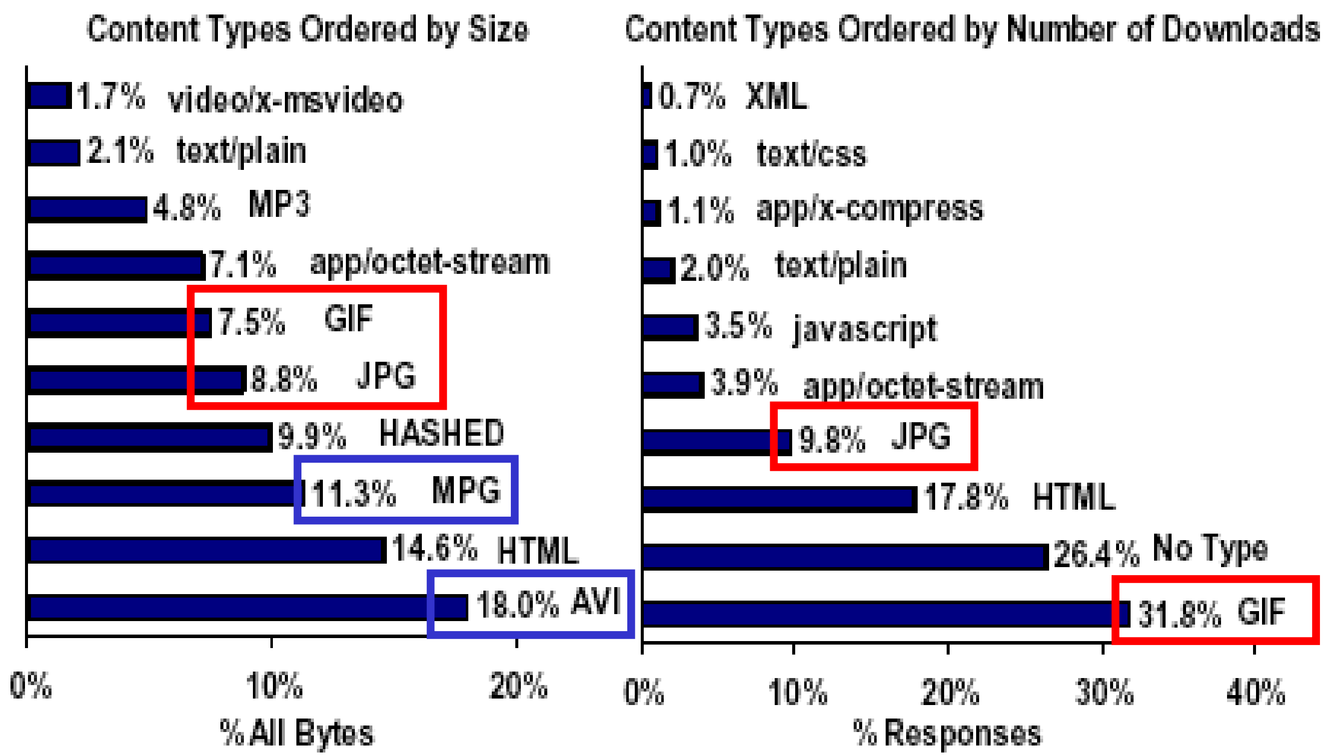




- Notice domination of Kazaa over outgoing bandwidth

Data Summary

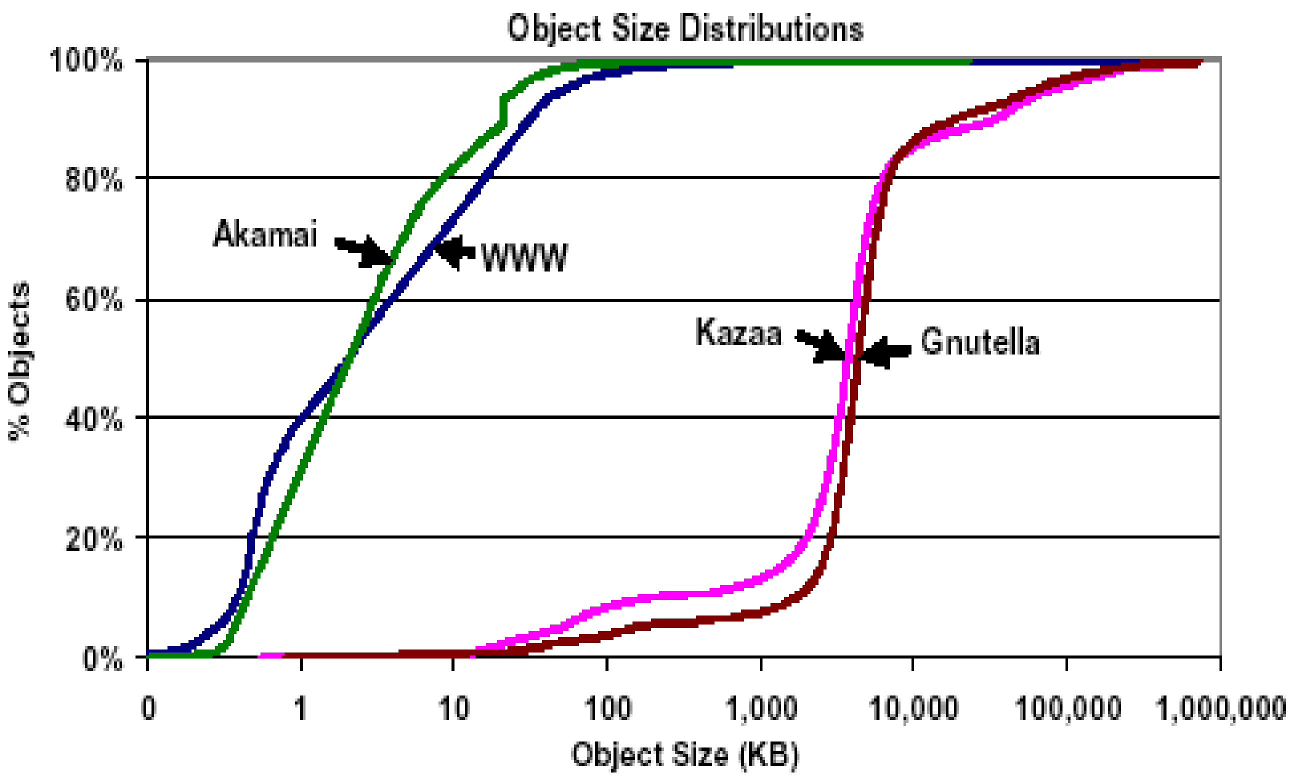
Content types downloaded



- Hashed refers to unidentifiable Kazaa transfers

Data Summary

Object size distribution

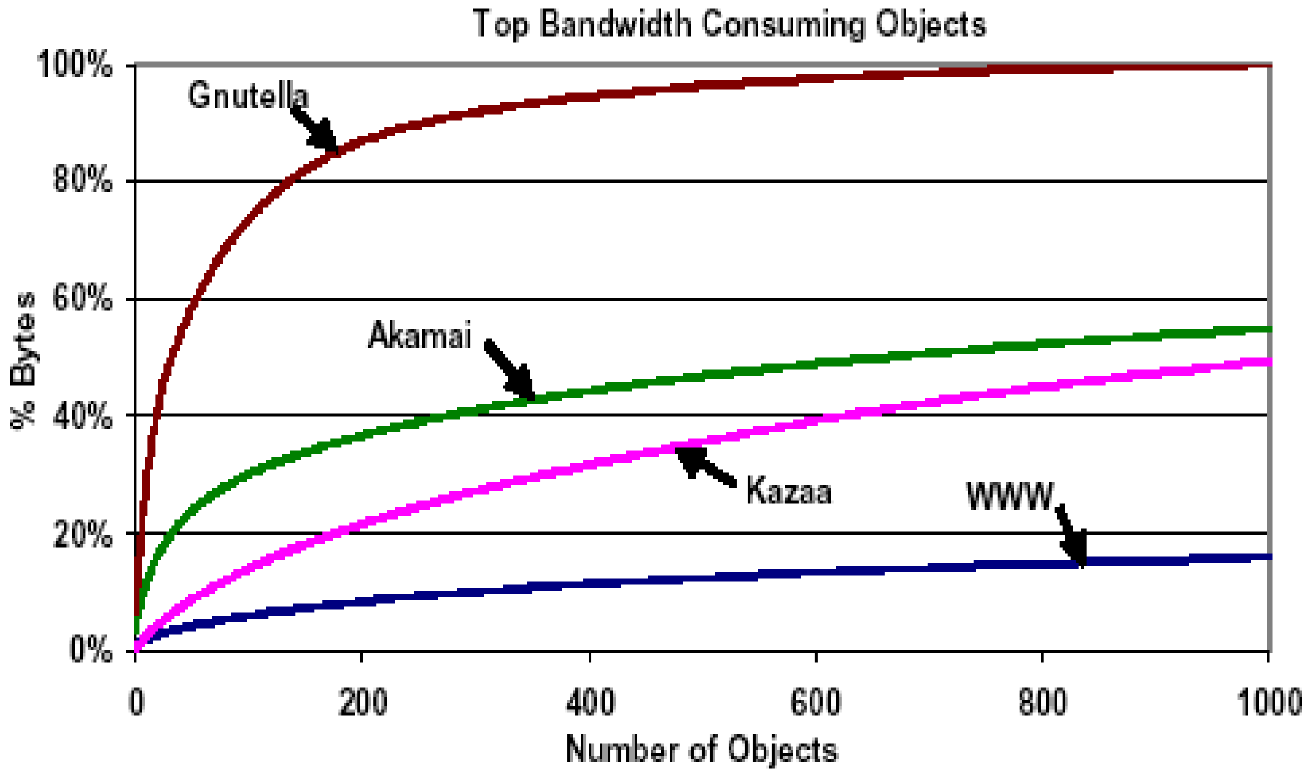


Median Object size:
WWW = 2KB
P2P = 4MB

- Notice x-axis is logarithmic

Data Summary

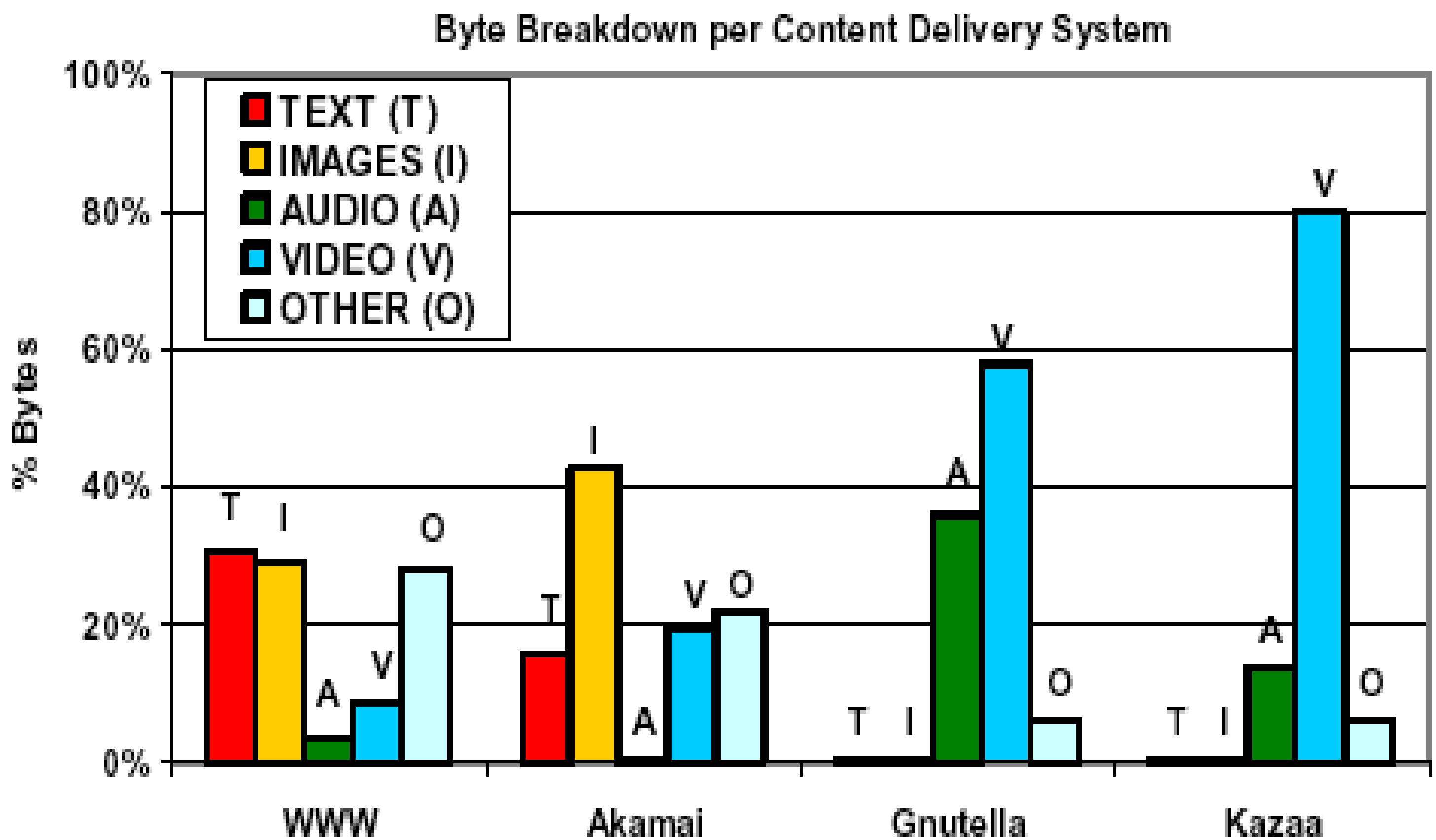
Top bandwidth consuming objects



- Why are 200 objects 90% of Gnutella bandwidth?

Data Summary

Downloaded bytes by object type



Data Summary

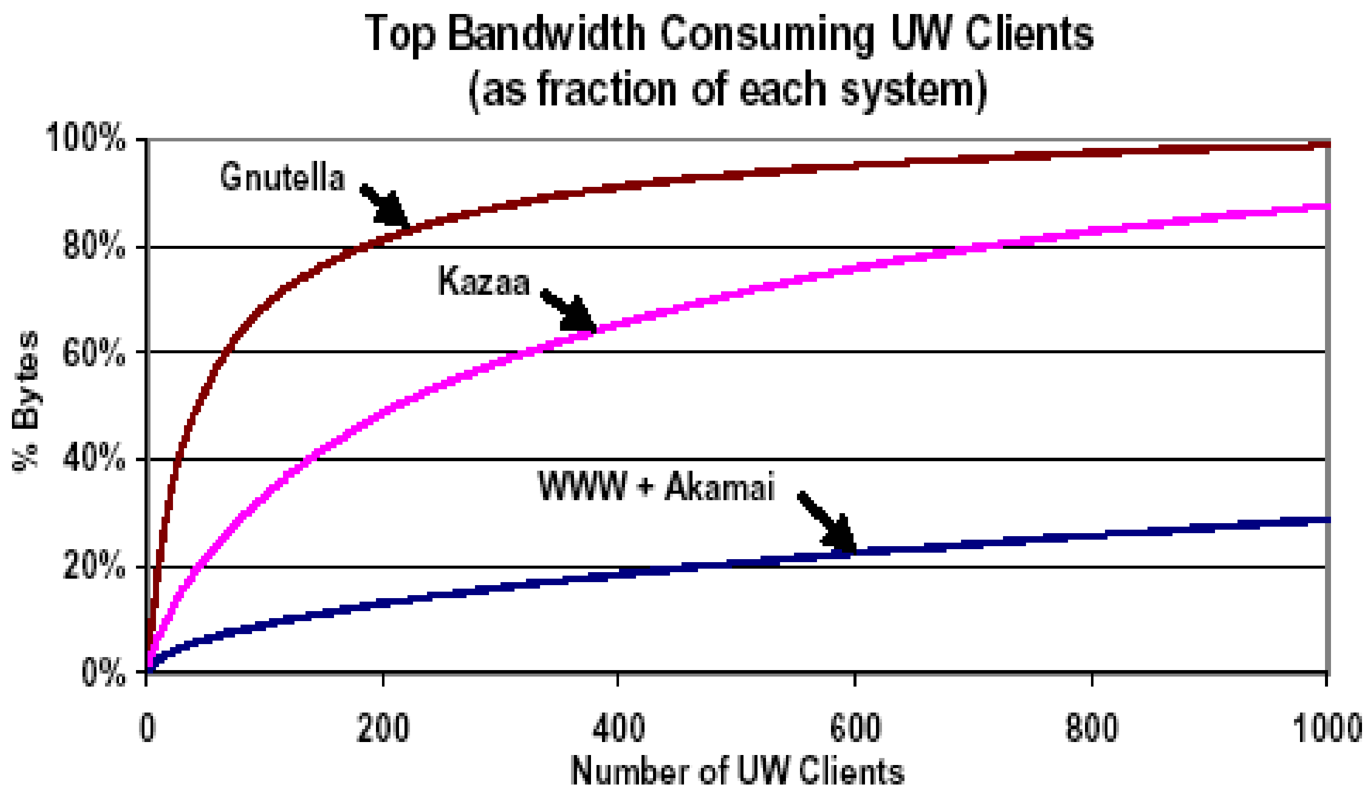
Top bandwidth consuming objects

	WWW (inbound)			Akamai			Kazaa (inbound)				Kazaa (outbound)			
	object size (MB)	GB consumed	# requests	object size (MB)	GB consumed	# requests	object size (MB)	GB consumed	# clients	# servers	object size (MB)	GB consumed	# clients	# servers
1	0.009	12.29	1,412,104	22.37	4.72	218	694.39	8.14	20	164	696.92	119.01	397	1
2	0.002	6.88	3,007,720	0.07	2.37	45,399	702.17	6.44	14	91	699.28	110.56	1000	4
3	333	6.83	21	0.11	1.64	68,202	690.34	6.13	22	83	699.09	78.76	390	10
4	0.005	6.82	1,412,105	9.16	1.59	2,222	775.66	5.67	16	105	700.86	73.30	558	2
5	2.23	3.17	1,457	13.78	1.31	107	698.13	4.70	14	74	634.25	64.99	540	1
6	0.02	2.69	126,625	82.03	1.14	23	712.97	4.69	17	120	690.34	64.97	533	10
7	0.02	2.69	122,453	21.05	1.01	50	715.61	4.49	13	71	690.34	54.90	447	16
8	0.03	1.92	56,842	16.75	1.00	324	579.13	4.30	14	158	699.75	49.47	171	2
9	0.01	1.91	143,780	15.84	0.95	68	617.99	4.12	12	94	696.42	43.35	384	14
10	0.04	1.86	47,676	15.12	0.80	57	167.18	3.83	39	247	662.69	42.28	151	2

- Notice the 9KB object, it alone accounts for 1.9% of WWW traffic.

Data Summary

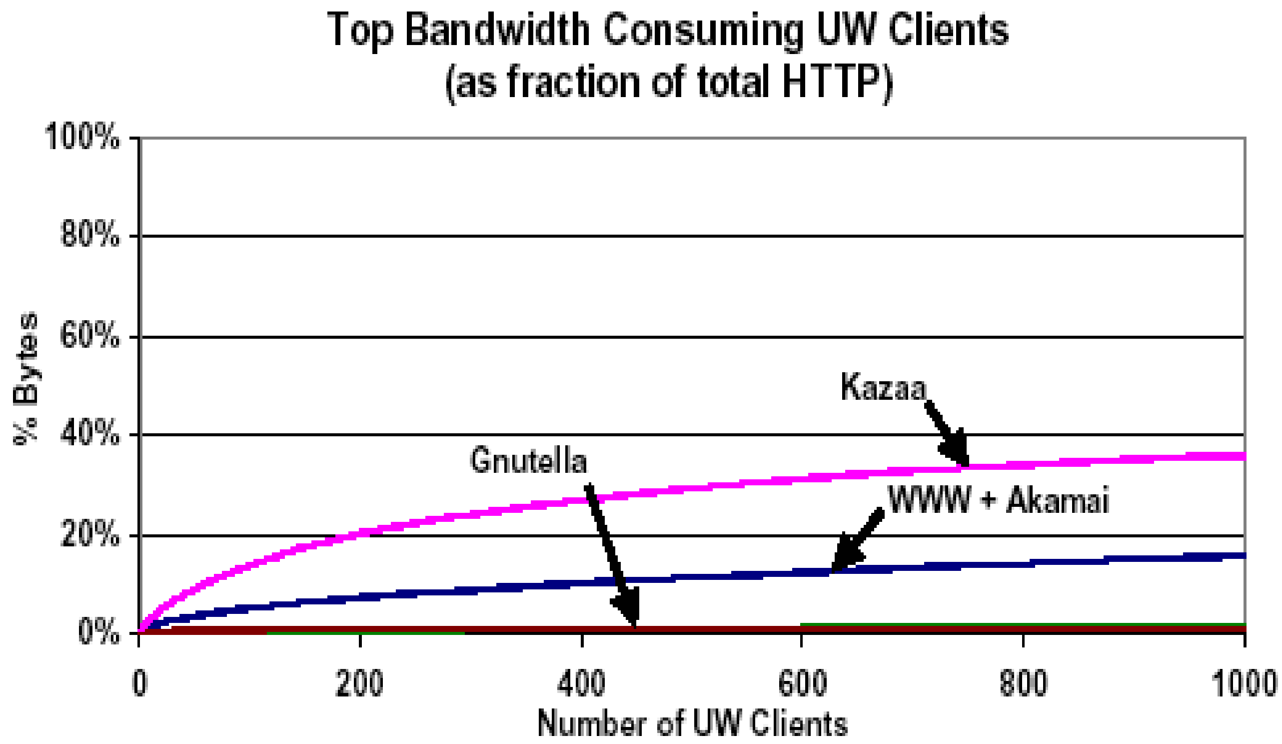
Top bandwidth consuming Clients



- 200 Clients consume 80% of Gnutella bandwidth

Data Summary

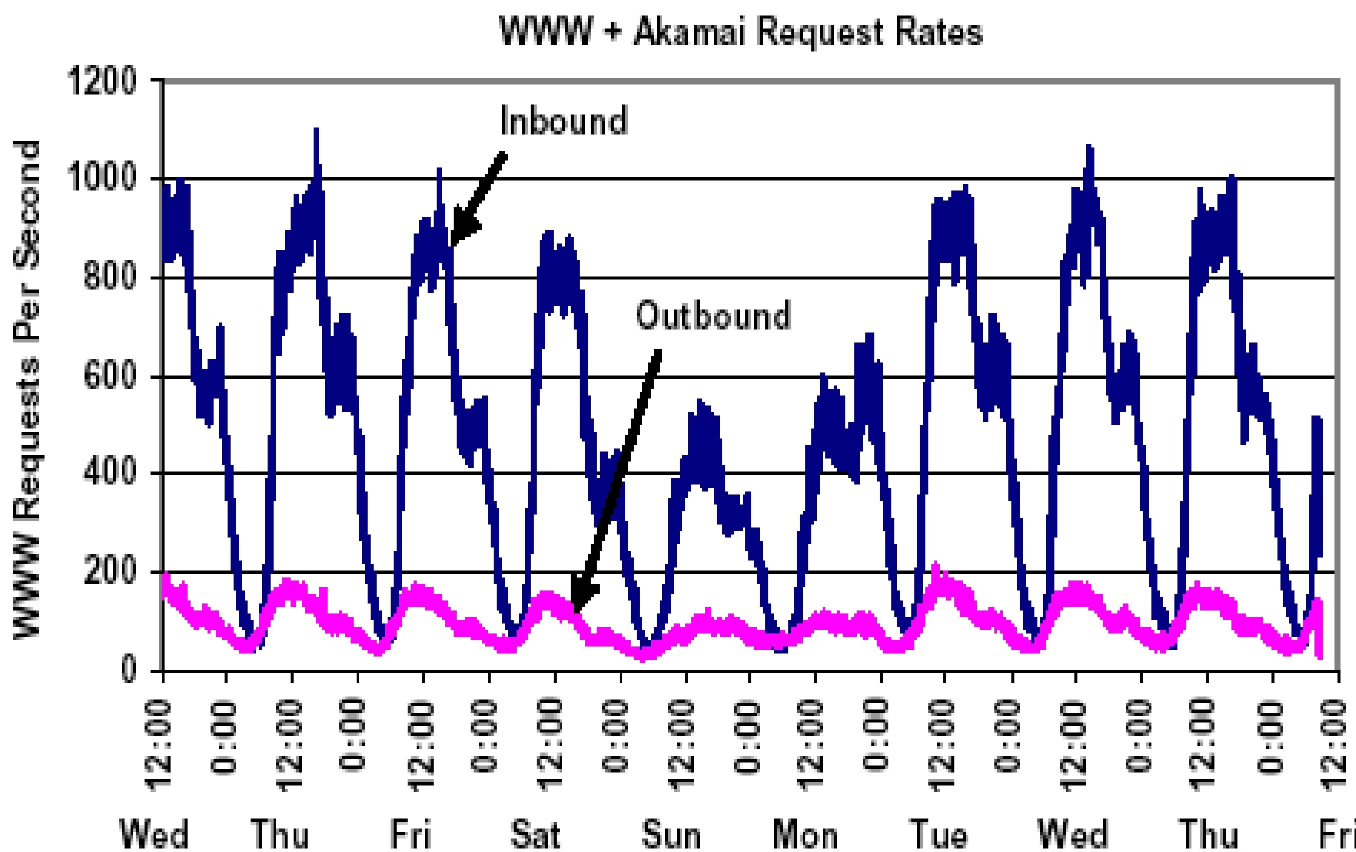
Top bandwidth consuming Clients



- 200 Kazaa clients consume 20% of all HTTP bandwidth

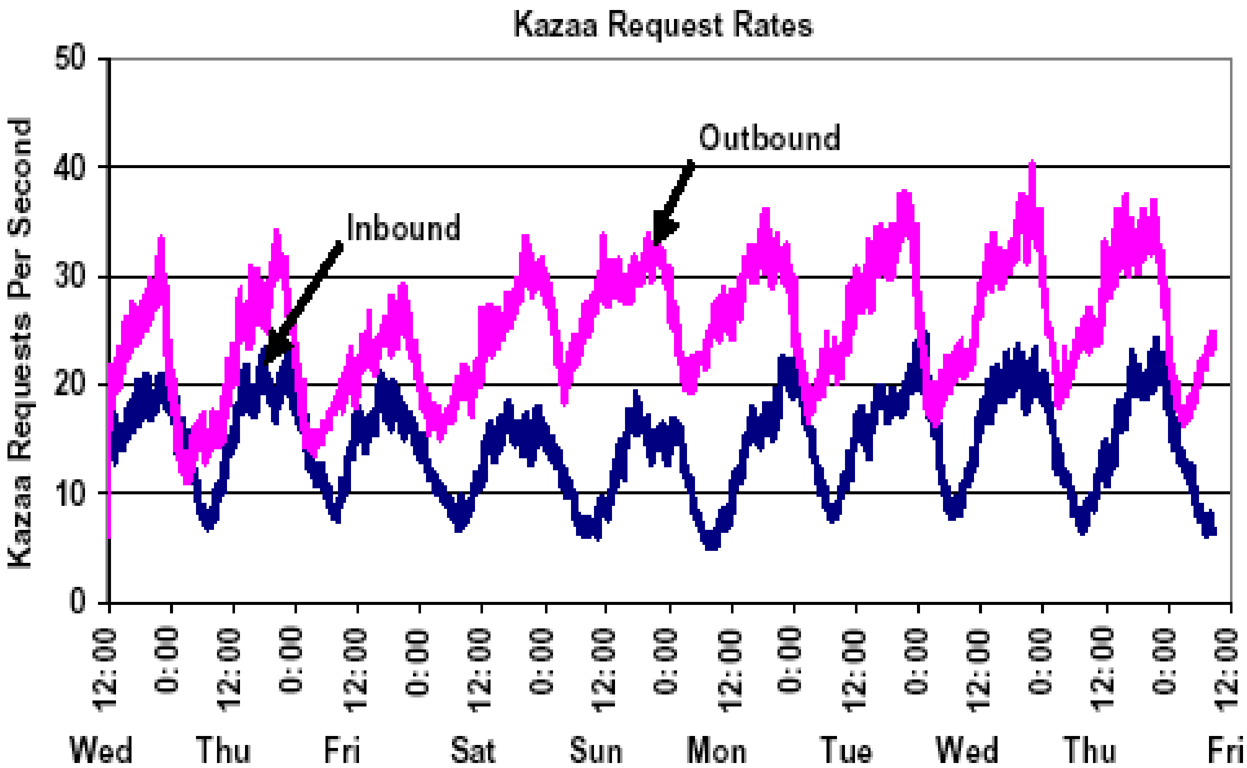
Data Summary

Request rates over time



Data Summary

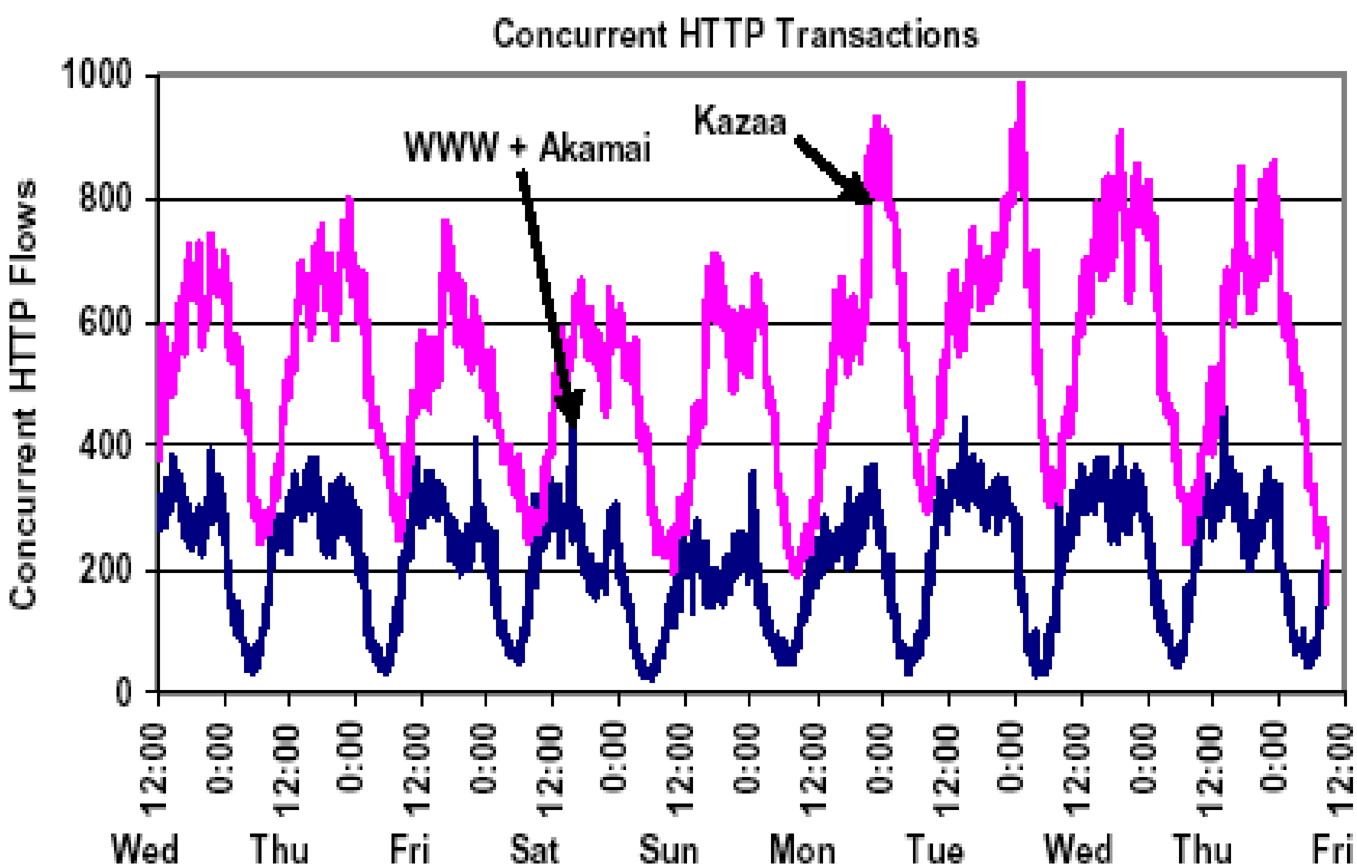
Request rates over time



- Notice significantly fewer total requests per second than WWW and Akamai

Data Summary

Concurrent HTTP transactions



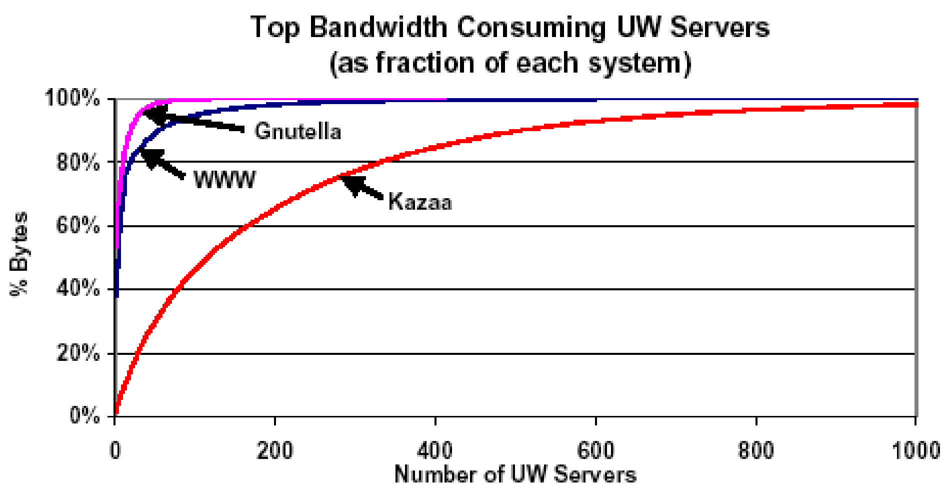
Medium duration of a request:

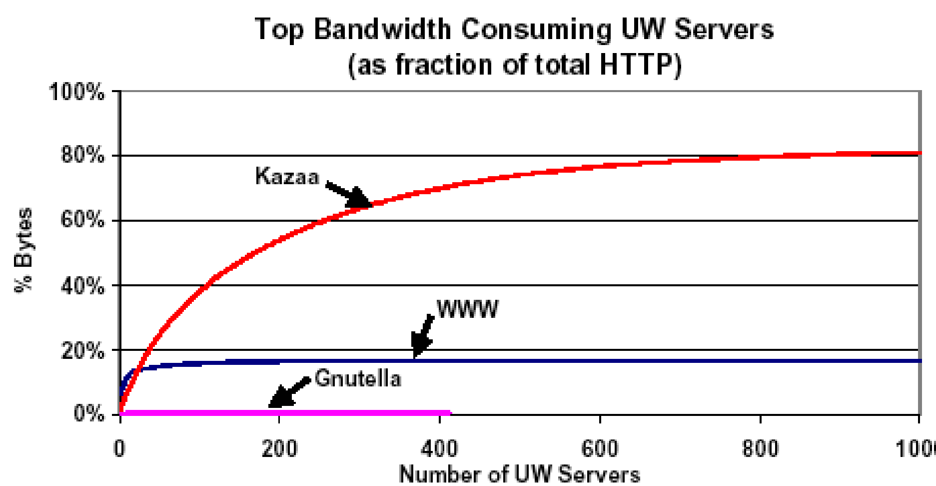
WWW = 120 ms

Kazza = 130 sec

Data Summary

Top UW-internal bandwidth producing servers

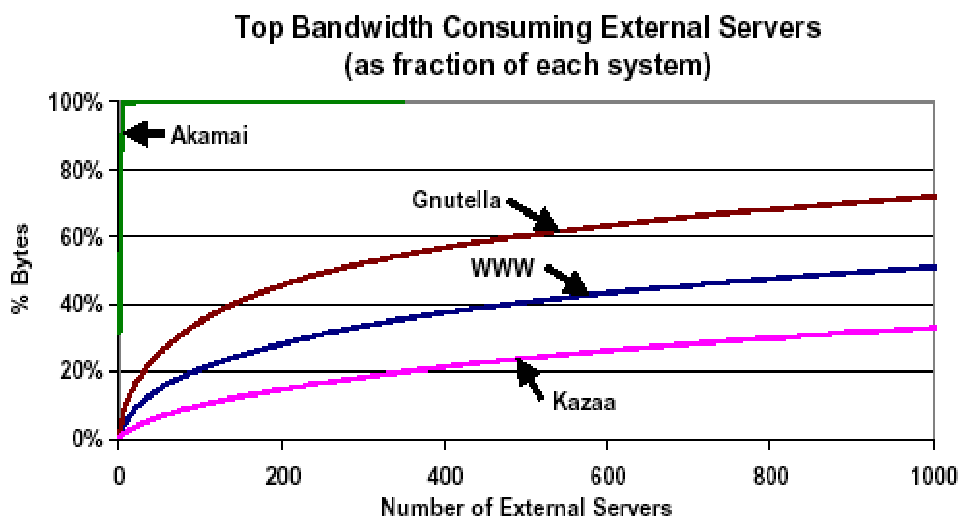


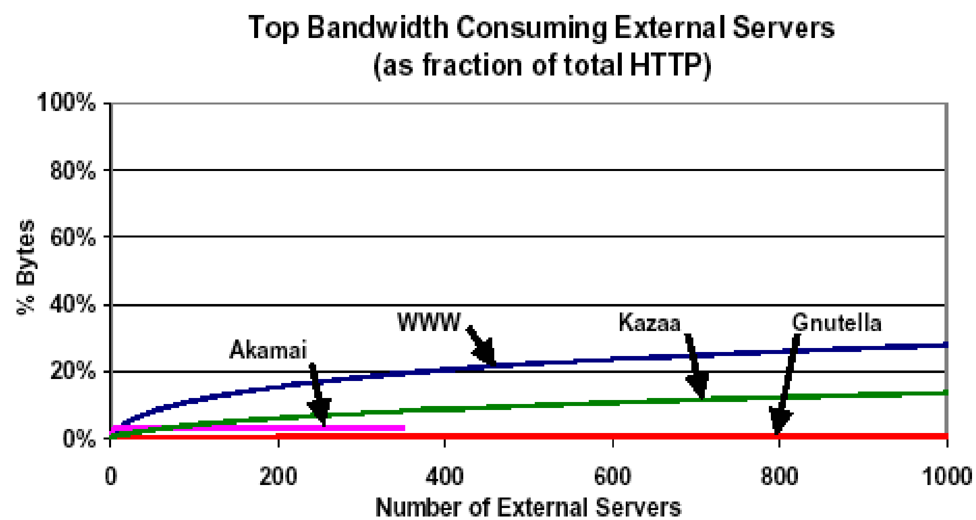


- Notice there are only 410 Gnutella servers at UW
- A very small amount of WWW servers are responsible for 40% of outgoing bytes.

Data Summary

Top UW-external bandwidth producing servers

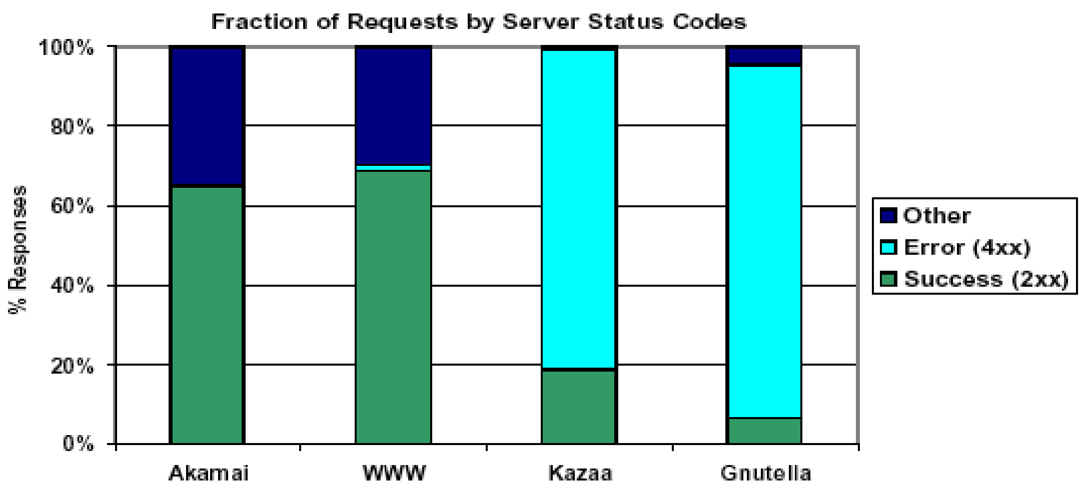


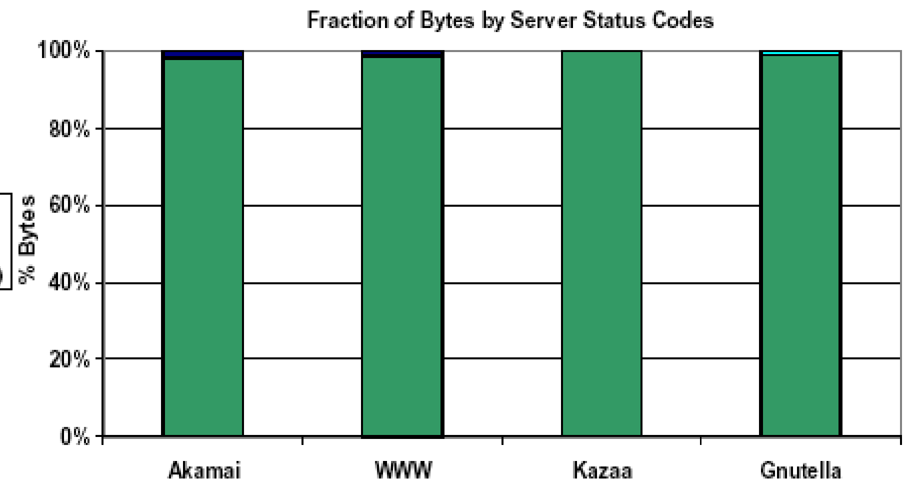


- Of about 350 Akamai servers, only about 10 account for the majority of the bandwidth

Data Summary

Server status codes





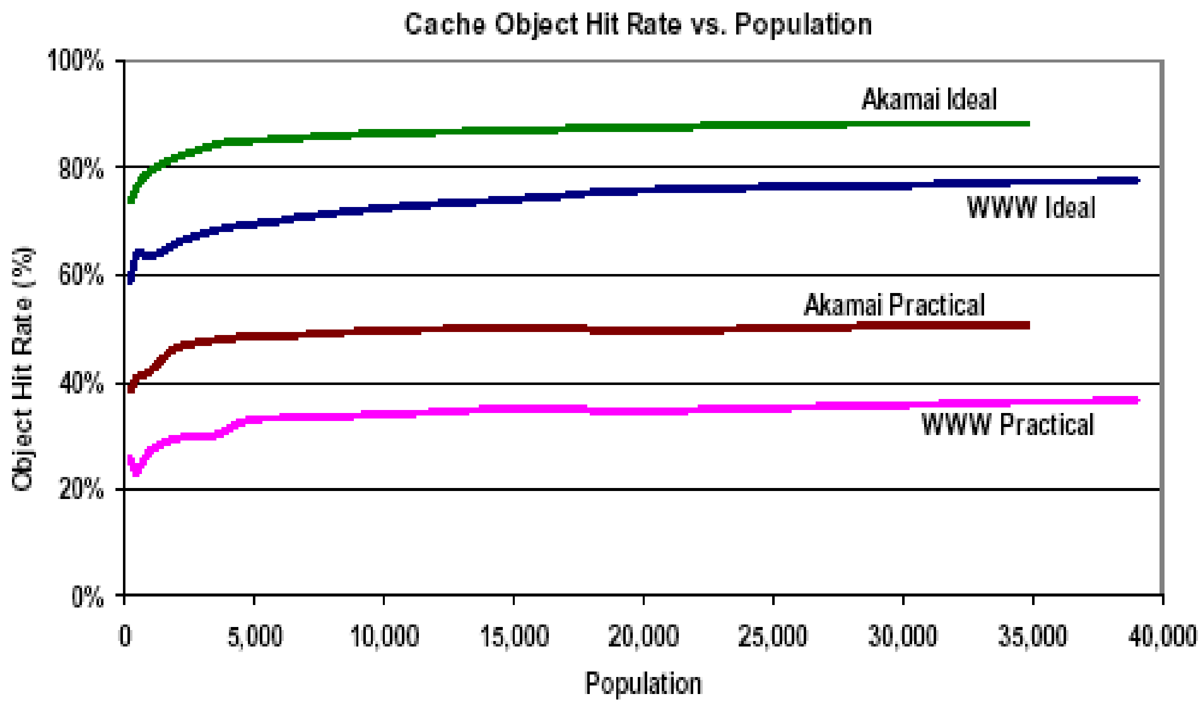
- Why does Akamai have no apparent error rate?
- Why are Kazaa and Gnutella error rates high?

Caching

- The paper addresses the following questions:
  - Is the use of Akamai as an external CDN justified for content providers, given the existence of local caches?
  - Can caching be applied to Peer to Peer networks effectively?

Caching

Akamai vs. WWW



- Notice in both ideal and practical cases, Akamai objects can be more effectively cached than normal web traffic

Caching

- Conclusion drawn:
  - Given the cache-ability of Akamai objects, the need for specialized content delivery networks like Akamai is greatly reduced if proxy caches are widely deployed enough
- Akamai objects are frequently requested after they expire from the cache. If properly marked with a longer time-to-live, the number of cache hits would increase significantly

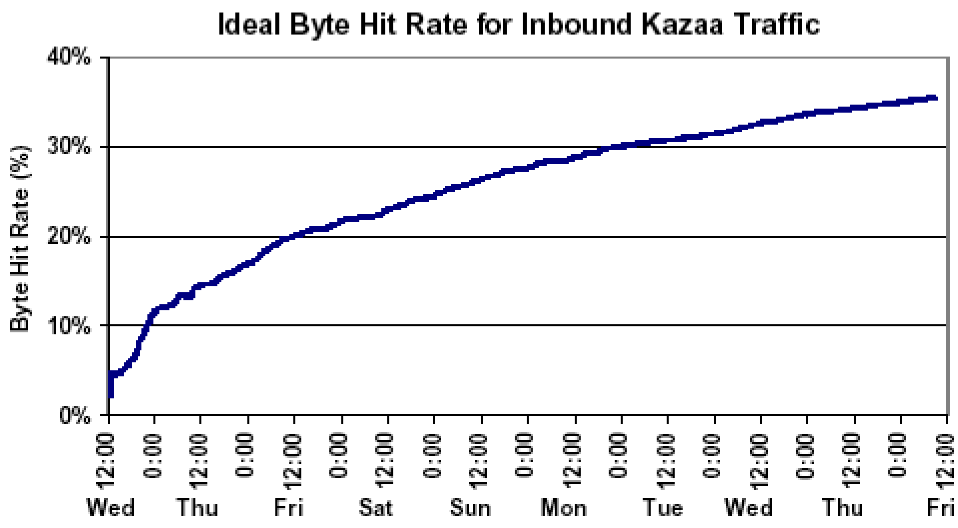
Caching

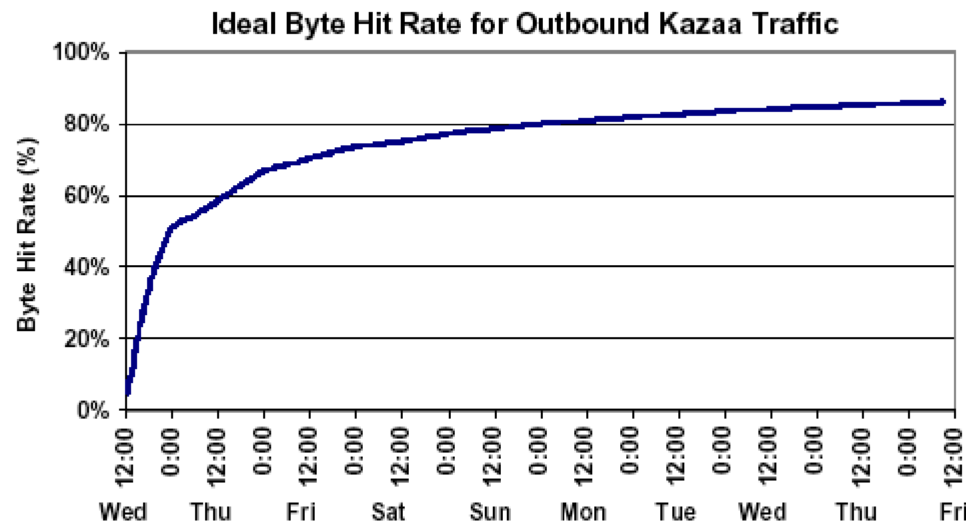
Peer-to-Peer Caching

- Authors designed a cache simulator for Kazaa
- The cache stores objects as 32K blocks
  - This differs from normal caches because it is possible that only parts of an object are cached, not necessarily a complete object.

Caching

Peer-to-Peer Caching





- After 2 days, three quarters (3/4) of outbound Kazaa traffic could come directly from the cache
- Why is inbound much lower?

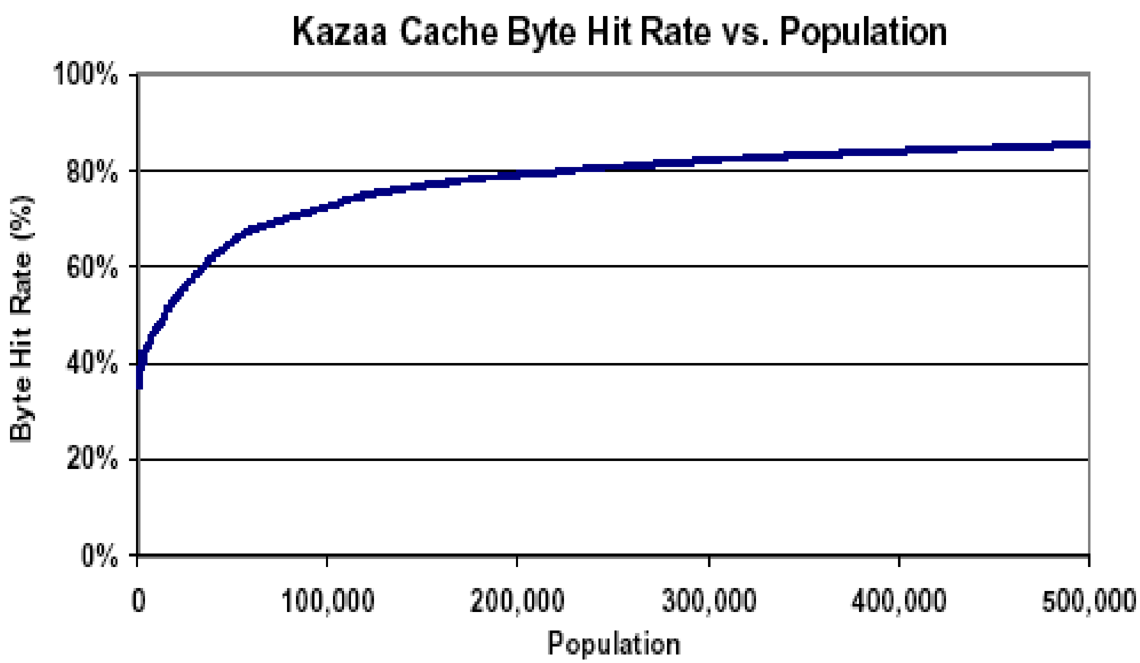
Caching

Peer-to-Peer Caching

- The top 300 bandwidth-consuming objects on Kazaa have an average size of 614MB. Why?
- A cache that stores the top 300 objects would have a hit rate of 38%

Caching

Kazaa cache byte hit rate vs. population



- Even for a small population, the hit rate is very good, starting at 40%.

Caching

Peer to Peer Conclusion

- A cache for a system like Kazaa would be overwhelmingly beneficial, especially by cache standards
- This begs the question: Why are there no Kazaa caches?

Conclusion

- Peer to peer traffic accounts for the majority of HTTP traffic, despite relatively small amount of peer to peer clients vs. web clients.
- Unlike web traffic, peer to peer traffic creates a heavy bandwidth flow in both directions
- The majority of peer to peer bandwidth is taken up by a small number of very large objects
- Peer to peer caching would be greatly beneficial