Performance comparison of distributed architectures for content adaptation and delivery of Web resources

Claudia Canali
University of Parma

Valeria Cardellini
University of Rome
"Tor Vergata"

Michele Colajanni
University of Modena and
Reggio Emilia

Riccardo Lancellotti
University of Modena and
Reggio Emilia

Current Web scenario

- Heterogeneity:
  - Client devices range from smartphones to high-end workstations
- Critical Web-based services
  - Web is a critical communication channel
  - **Need** for system to enable ubiquitous Web access.

Web content adaptation *on-the- fly*

Functions in a distributed Web content adaptation system

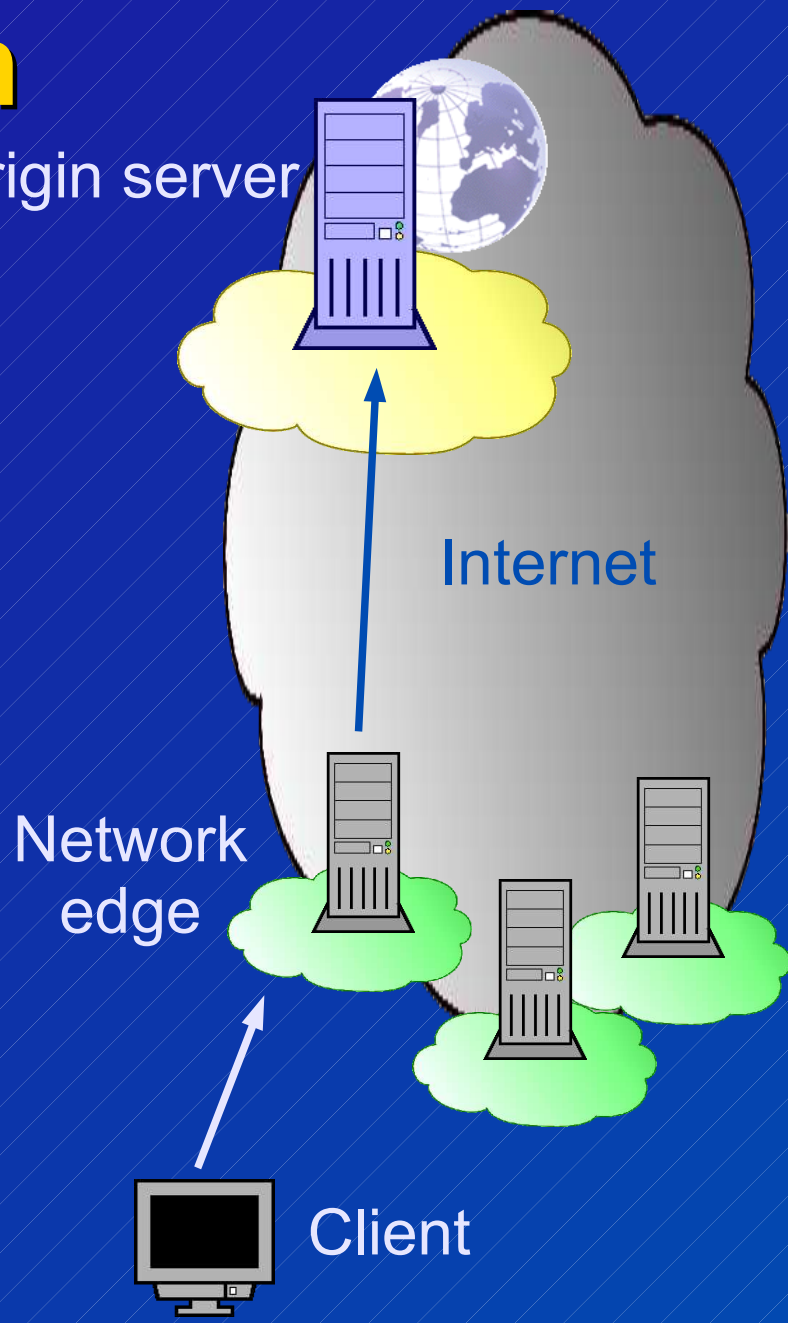
- Content adaptation
  - Computationally expensive (on-the-fly adaptation)
- Client capability/User preferences identification
- Caching
  - Multi-version caching
- Location of (possibly adapted) resources
  - Multi-version lookup process: Exact hit, Useful hit and Miss
- Interaction with Origin server

On which nodes to place these functions?

Providing Web content adaptation

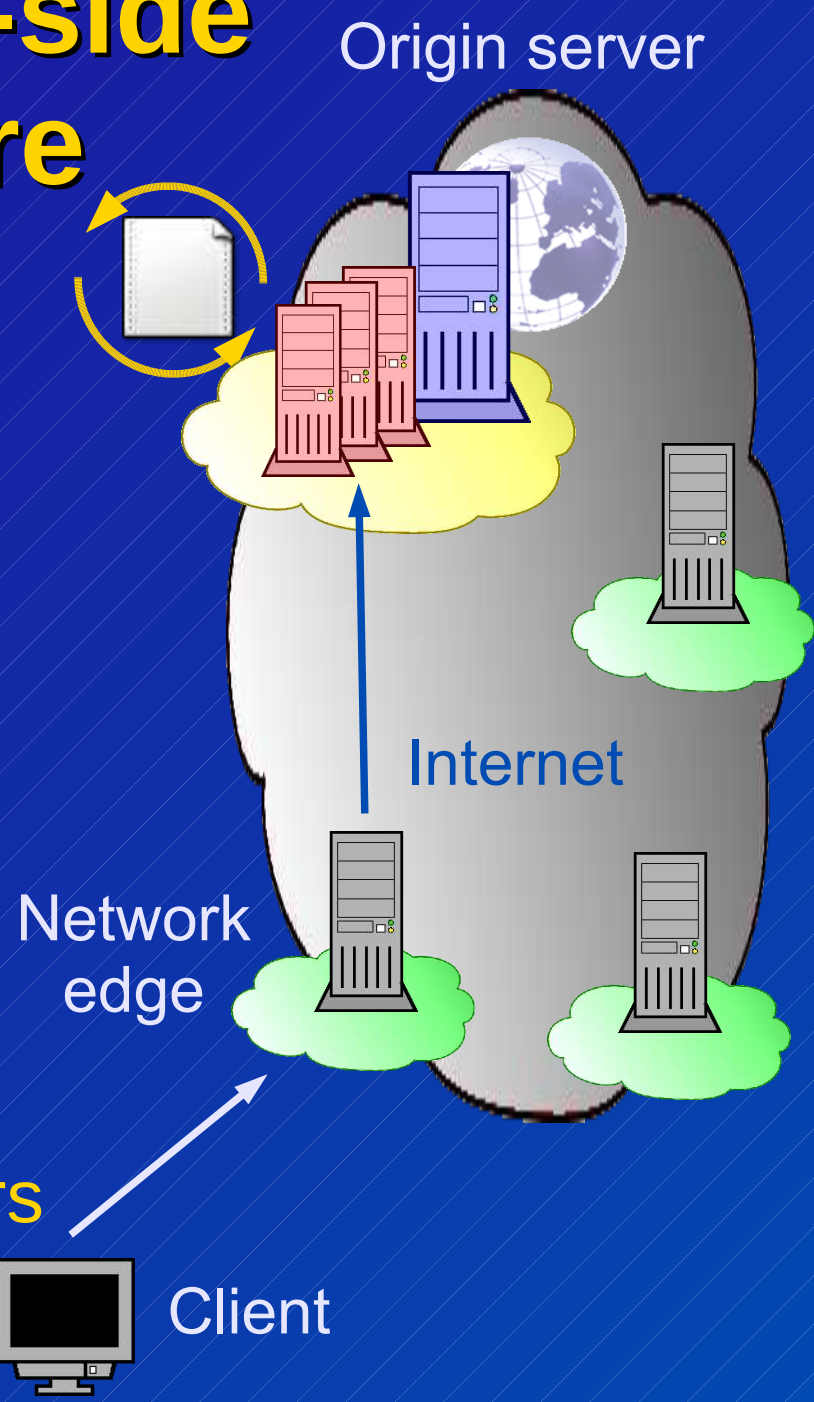
Different approaches for mapping content adaptation functions on the nodes:

- "Keep every function in the origin server area"
- "Move most functions towards the network edge nodes"
  - → Non cooperative edge server-side architecture
  - "Exploit potential of distributed architectures by allowing cooperation among edge nodes"



Origin server-side architecture

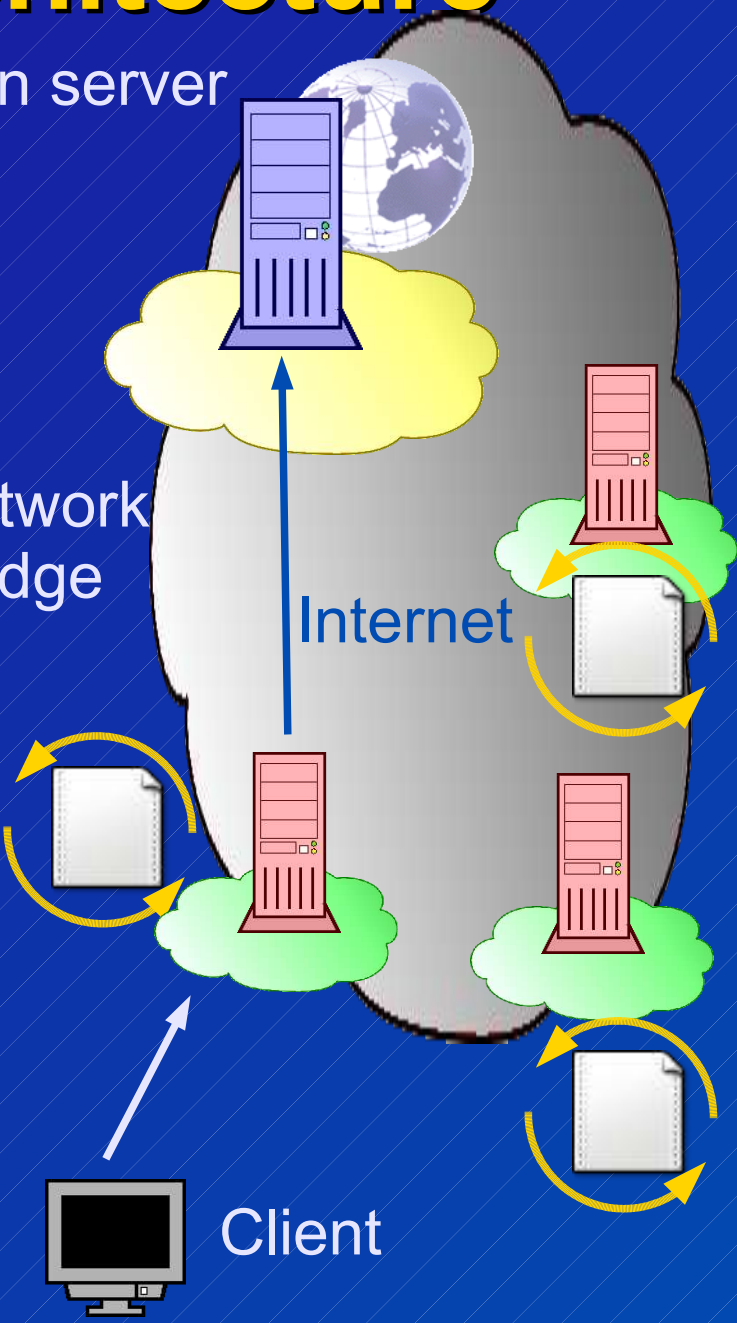
- "Keep every function in the origin server area"
- Potential advantages
  - Simplify interaction with origin server (security / privacy / sophisticated services)
  - Can exploit clusters
- Possible drawbacks
  - Sensitive to network parameters
  - High latency



Edge server-side architecture

- "Move most functions towards the network edge nodes"
- Potential advantages
  - Caching is more effective
  - Reduce bandwidth usage
- Possible drawbacks
  - Higher complexity than origin server-side approach

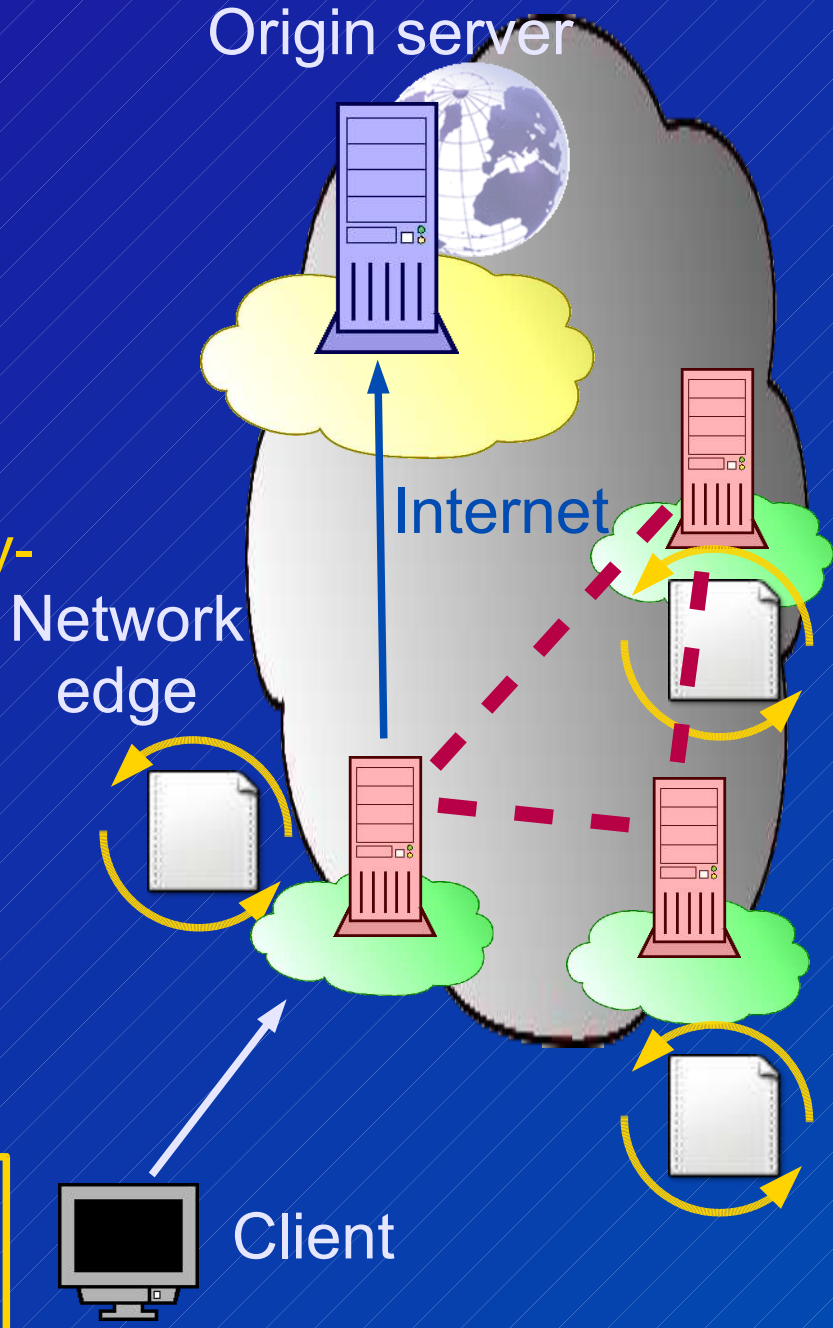
What is the performance gain from pushing services on the network edge?



Cooperative Edge server-side architecture

- "Exploit potential of distributed architectures by allowing cooperation among edge nodes"
- We focus on the best performing algorithm for cooperative lookup (query-based)
- Potential advantages
  - Increased efficiency
- Potential drawbacks
  - Higher complexity

What is the advantage from cooperation?

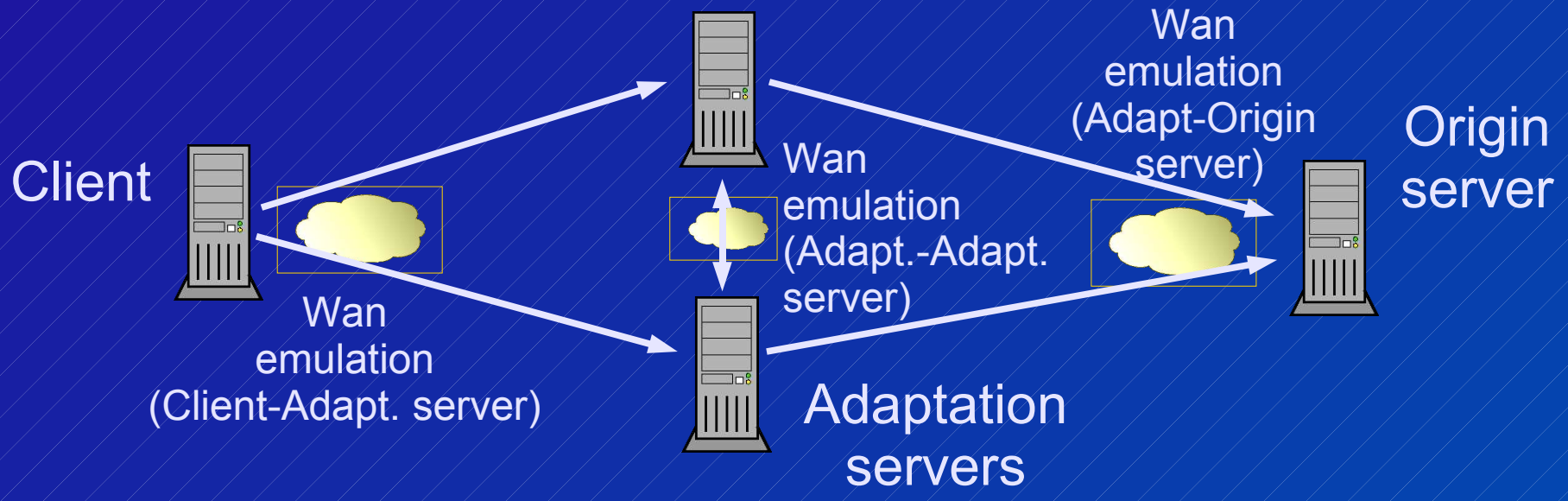


Main goals:

- Comparison of leading solutions for content adaptation
  - What is the gain from pushing content adaptation on the network edge? Under which circumstances this performance gain is more evident?
  - What is the advantage achieved through cooperation?
- Performance evaluation with real prototypes in a controlled environment
  - Different workloads
  - WAN emulation with multiple network scenarios

Performance evaluation

- Experimental setup
  - 16 nodes with content adaptation capabilities (adaptation servers)
  - 1 Web server (Origin server) + 1 client emulator
  - WAN emulation (*NetEm* network scheduler: delay, packet loss, bandwidth limitation



Performance evaluation

- Two workload models (prevalent static resources)
  - IRcache (from IRcache logs)
  - Photo album (heavy content adaptation tasks)
- Multiple WAN setups, we report the most significant results (sensitivity to bandwidth)

Architecture	WAN-emulated links	Bandwidth [Mbit/s]	Delay [ms]	Loss
Origin server-side	Client-Adapt. server	8, 16, 32	100	1,00%
Edge server-side	Adapt.-Origin server	8, 16, 32	100	1,00%
Cooperative edge server-side	Adapt.-Origin server	8, 16, 32	100	1,00%
	Adapt.-Adapt. server	8, 16, 32	25	1,00%

Architecture comparison: Origin server *vs.* Edge server-side

IRcache workload

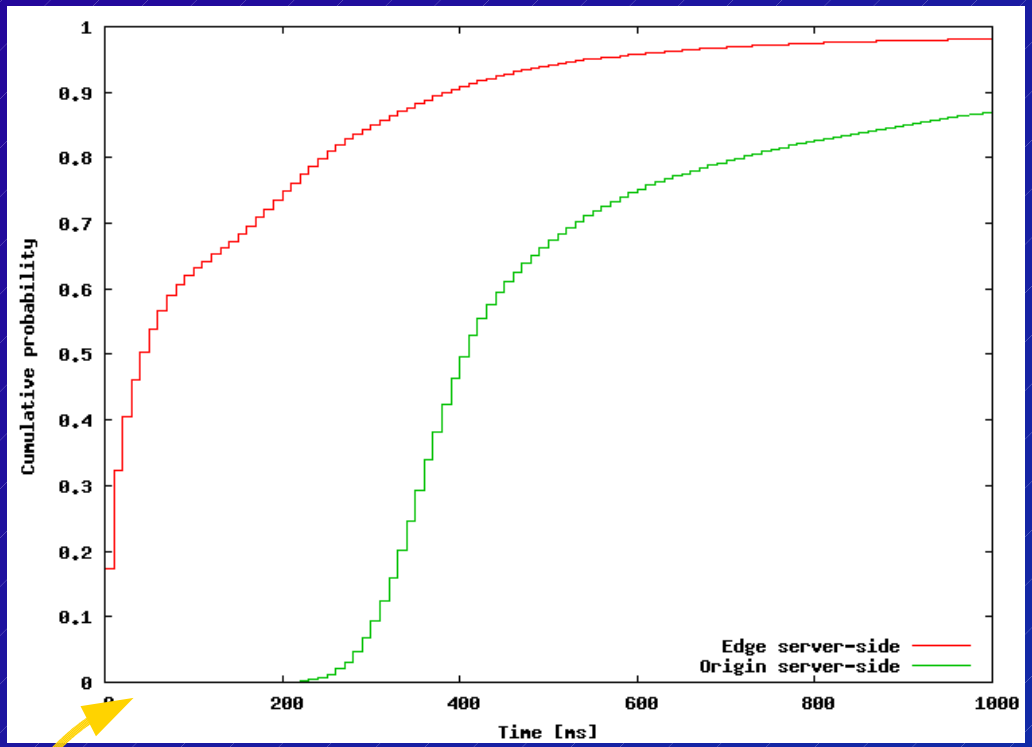
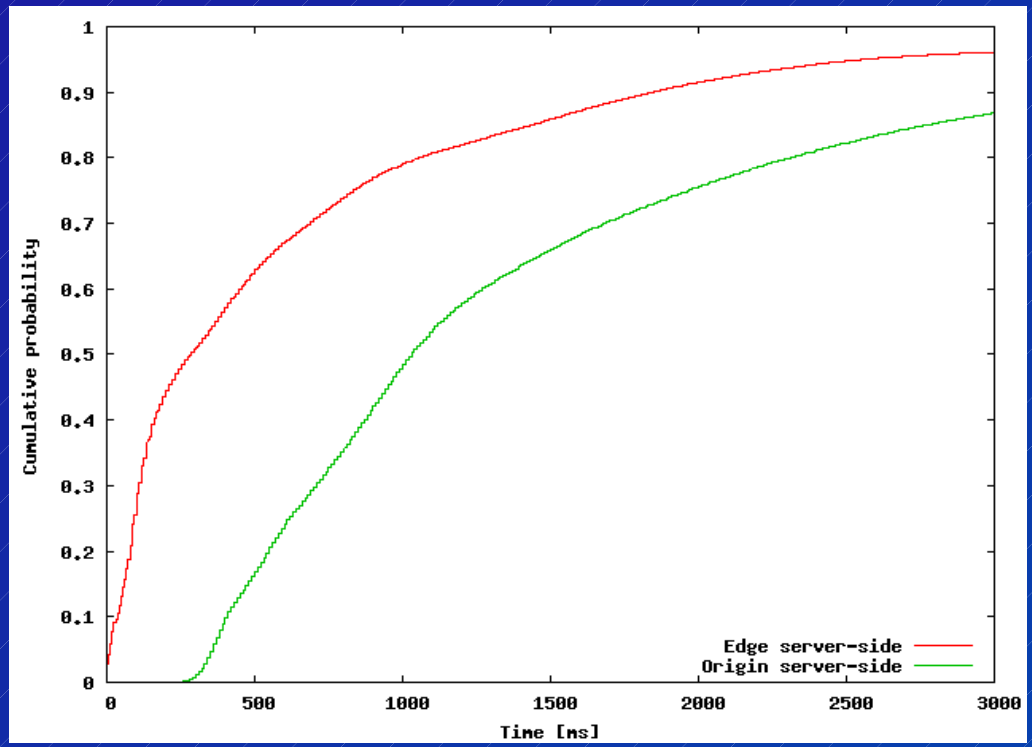


Photo album workload



- Edge server-side always outperforms Origin server-side
- Performance gain is more significant in the case of light workload (IRcache)

Architecture comparison: Impact of cooperation

IRcache workload

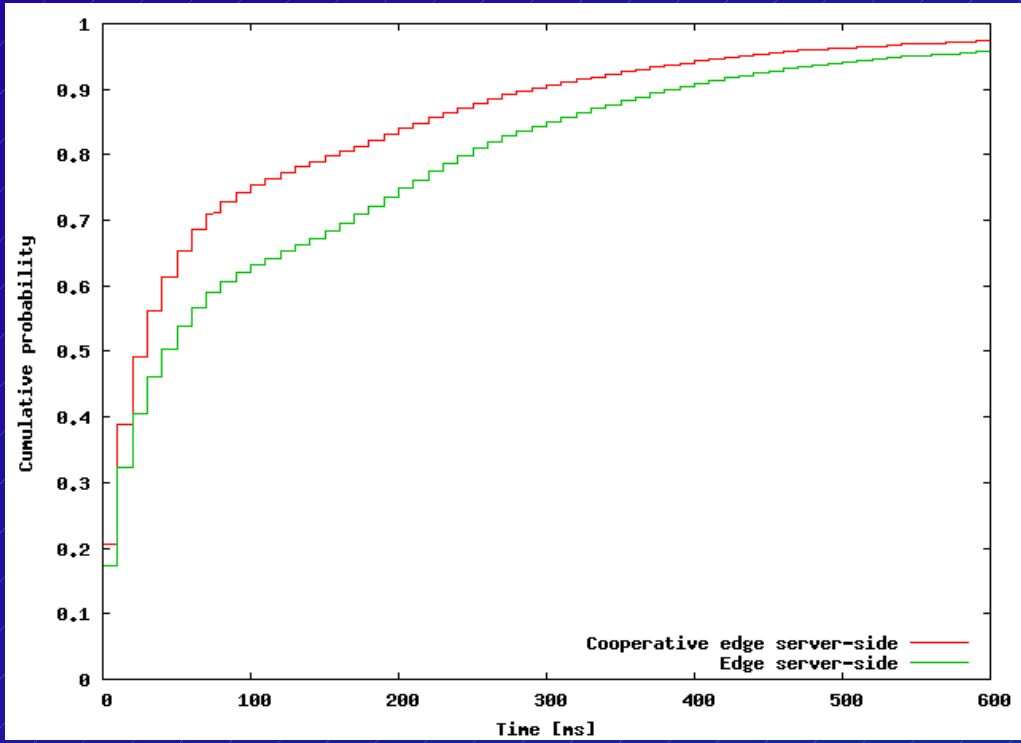
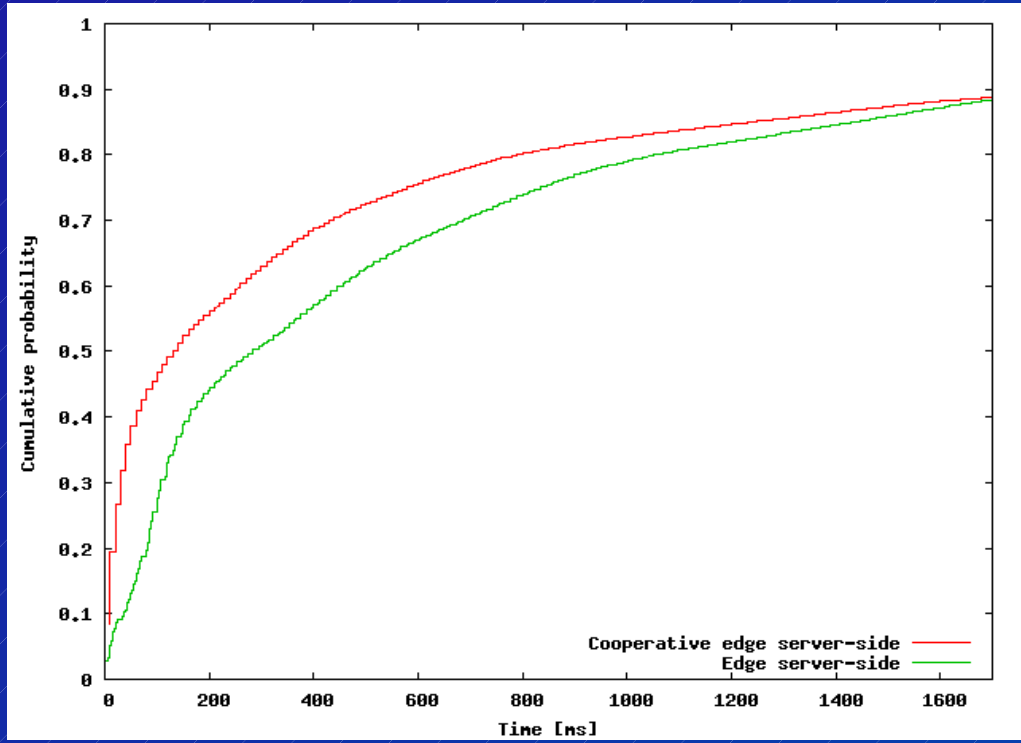


Photo album workload

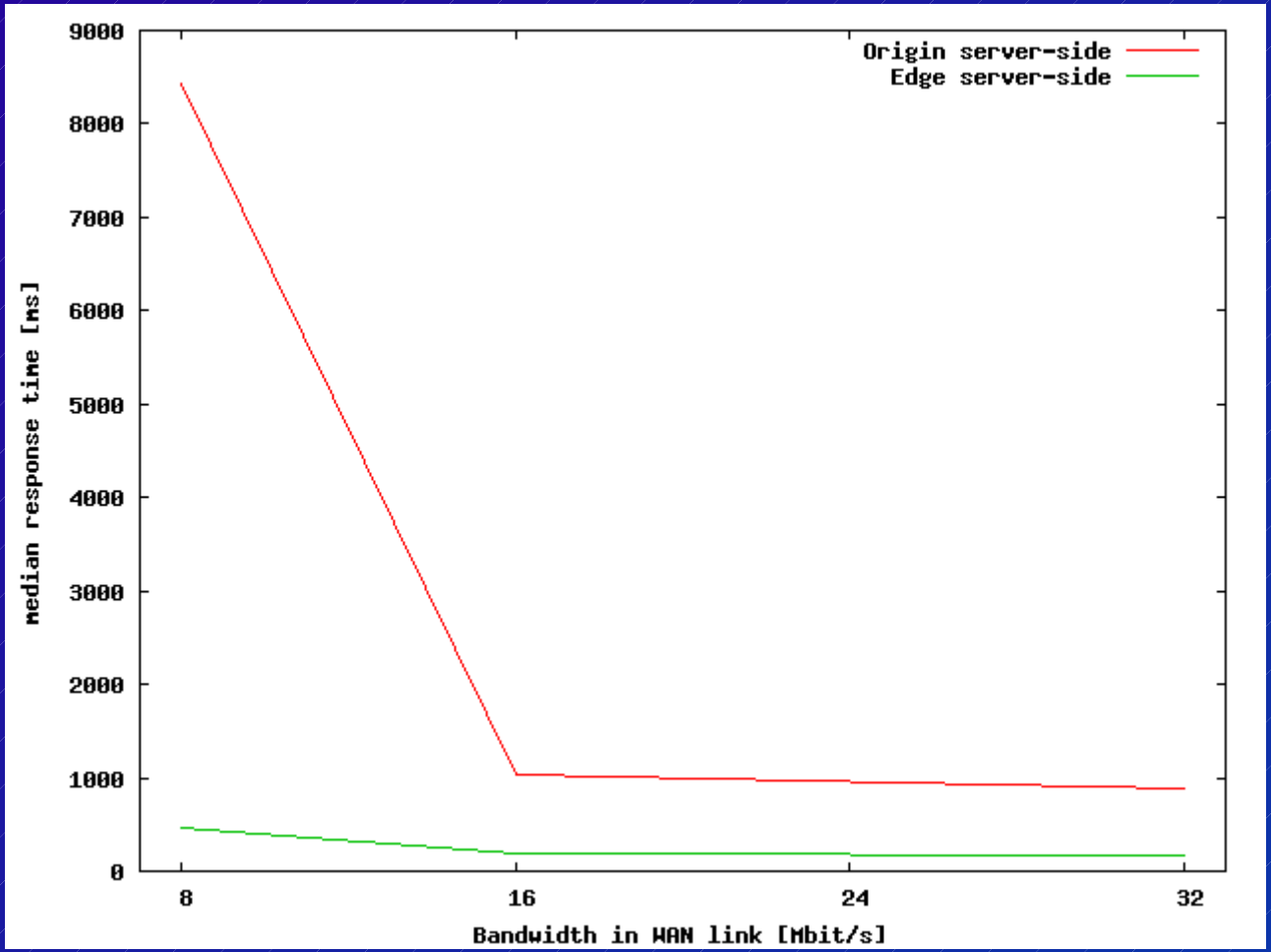


- Performance improvement on median response time (cooperation can improve cache hit rate)
- Less advantage for the 90-percentile (a miss is more expensive in the case of cooperation)

Summary (architecture comparison)

- Pushing content adaptation on the network edge has a significant performance gain in the case of “light” services
  - Network-related time is dominant
  - In the case of a cache hit we save a connection to the origin server
- Performance gain from cooperation is related to the effectiveness of cooperative caching. Limited global performance gain
  - Cooperation increases the hit rate
  - No gain in the case of cache miss

Sensitivity to network parameters: Origin vs. Edge server-side



Median response time
Photo album workload

- Edge server-side provides better performance
  - Lower response time
  - Reduced sensitivity to bandwidth
  - Reduced number of open sockets (less parallel requests)

Sensitivity to network parameters: impact of cooperation

- In the case of poor network bandwidth cooperation increases dramatically performance
- The cooperation reduces sensitivity to network effects

Bandwidth [Mbit/s]	Edge server-side architecture		Cooperative edge server-side architecture			
	Adapt.-Origin server		Adapt.-Origin server		Adapt.-Adapt. server	
	Response time		Response time		Response time	
	median	**90-perc.**	**median**	**90-perc.**	**median**	**90-perc.**
8	470	54680	170	2030	150	1960
16	180	1848	130	1870	130	1870
32	170	1630	110	1660	110	1790

Summary (sensitivity to network)

- Edge server-side architecture reduces network utilization with respect to the Origin server-side approach
  - Reduction in the sensitivity to network parameters
- Cooperation further reduces the load on the network links
- The real advantage from cooperation lies in the limited sensitivity to network parameters

Conclusions

- Gain from pushing content adaptation on the network edge
  - Edge server-side approach is always best
  - The performance gain is more evident in the case of services with lower computational complexity

→ We should move “light” services towards the edge

- Advantages achieved through cooperation
  - **Reduction in sensitivity to network parameters**

→ We should exploit cooperation in the case of poor network conditions (e.g., low bandwidth and/or network congestions)

Performance comparison of distributed architectures for content adaptation and delivery of Web resources

For more information:

http://weblab.ing.unimo.it/research/trans_caching.shtml

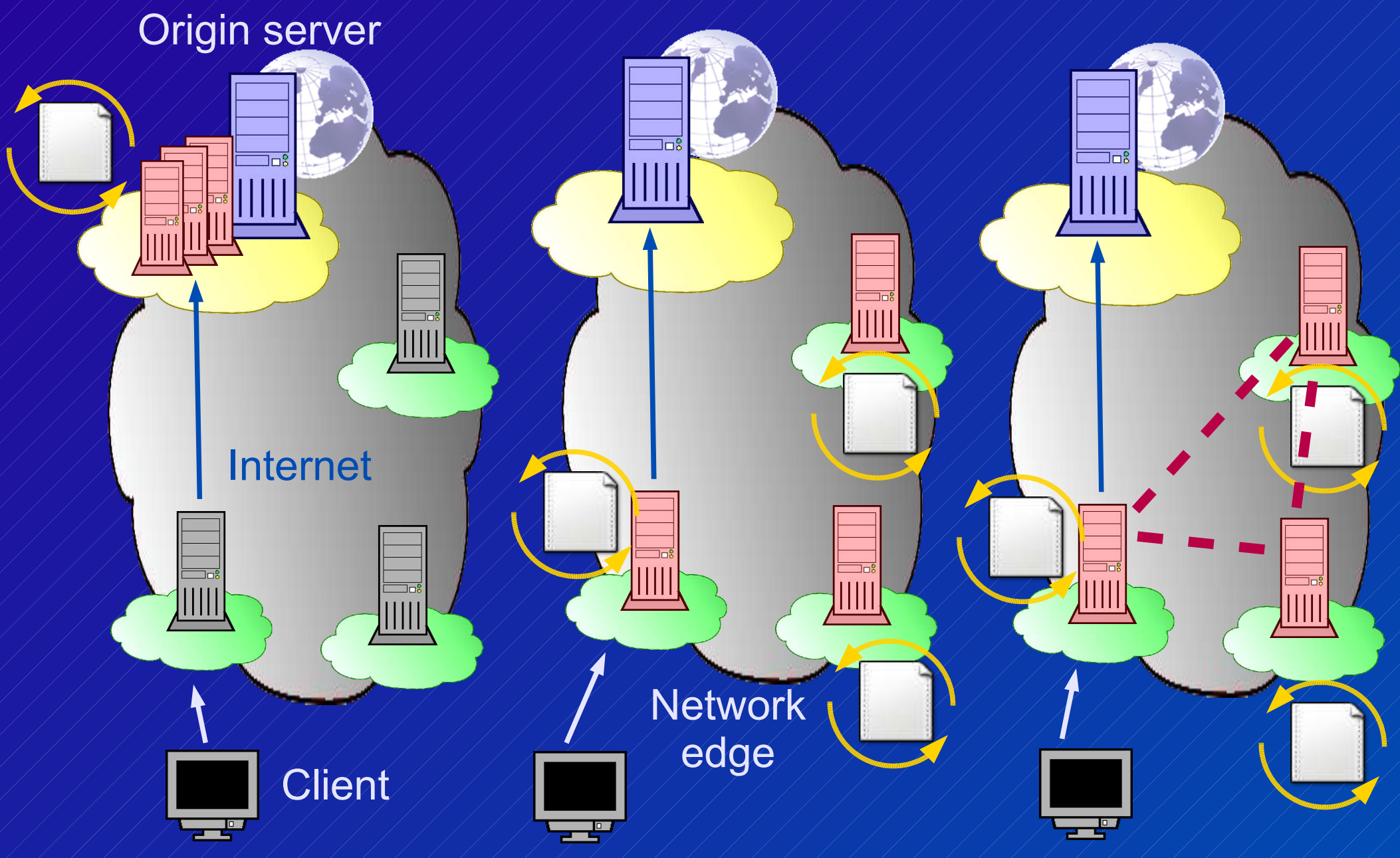
Conclusions

- Edge server-side architecture outperforms Origin server-side approach
  - Performance gain is more evident when content adaptation time is reduced
  - Performance gain increases dramatically in the case of low bandwidth links
- Cooperation in edge server-side architecture provides better performance
  - Performance gain less evident then in the case of origin-server side architecture
  - Cooperation reduces sensitivity to network parameters

Critical Issue

- Content adaptation is computationally expensive
  - Can take advantage from caching
  - We can reduce computational load by exploiting already-adapted resources
- Caching in a content adaptation system is more complex than traditional Web caching
  - Multiple versions of the same resource
  - We need multi-version lookup
  - We have a rich caching semantics: a lookup can result in Exact hit, Useful hit and Miss

Architectures



Providing Web content adaptation

- Three base approaches
  - ~~Client-side~~
  - Origin server-side
  - Edge-side (possibly cooperative)
- Drawbacks of the client-side approach
  - Limited computation power on edge nodes (not efficient)
  - Requires client-specific implementation (not general)
  - Does not save bandwidth (not effective)