A two-level distributed architecture for Web content adaptation and delivery

Claudia Canali
University of Parma

Valeria Cardellini
University of Rome
"Tor vergata"

Michele Colajanni
University of Modena

Riccardo Lancellotti
University of Modena

Philip S. Yu
IBM T.J. Watson
research center

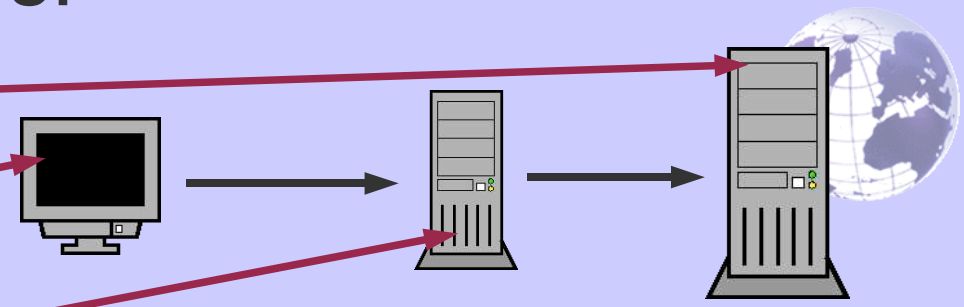
The evolving Web scenario

- Heterogeneous clients
  - Different display, CPU, network
- Heterogeneous user behavior
  - Request for sophisticated,personalized services

Content adaptation
(*transcoding, personalization*)

Content adaptation

- Possible approaches:
  - Server-based
  - Client-based
  - Intermediary-based



- Issue:
  - Content adaptation is computationally expensive
- Typical solutions
  - Caching → reduce adaptation operations
  - Replication → load sharing

Issues for efficient content adaptation

- Caching issue
  - multiple version of every resource→ working set size grows
  - locality improves caching effectiveness
- Replication issue
  - provide adequate load sharing

Contribution:

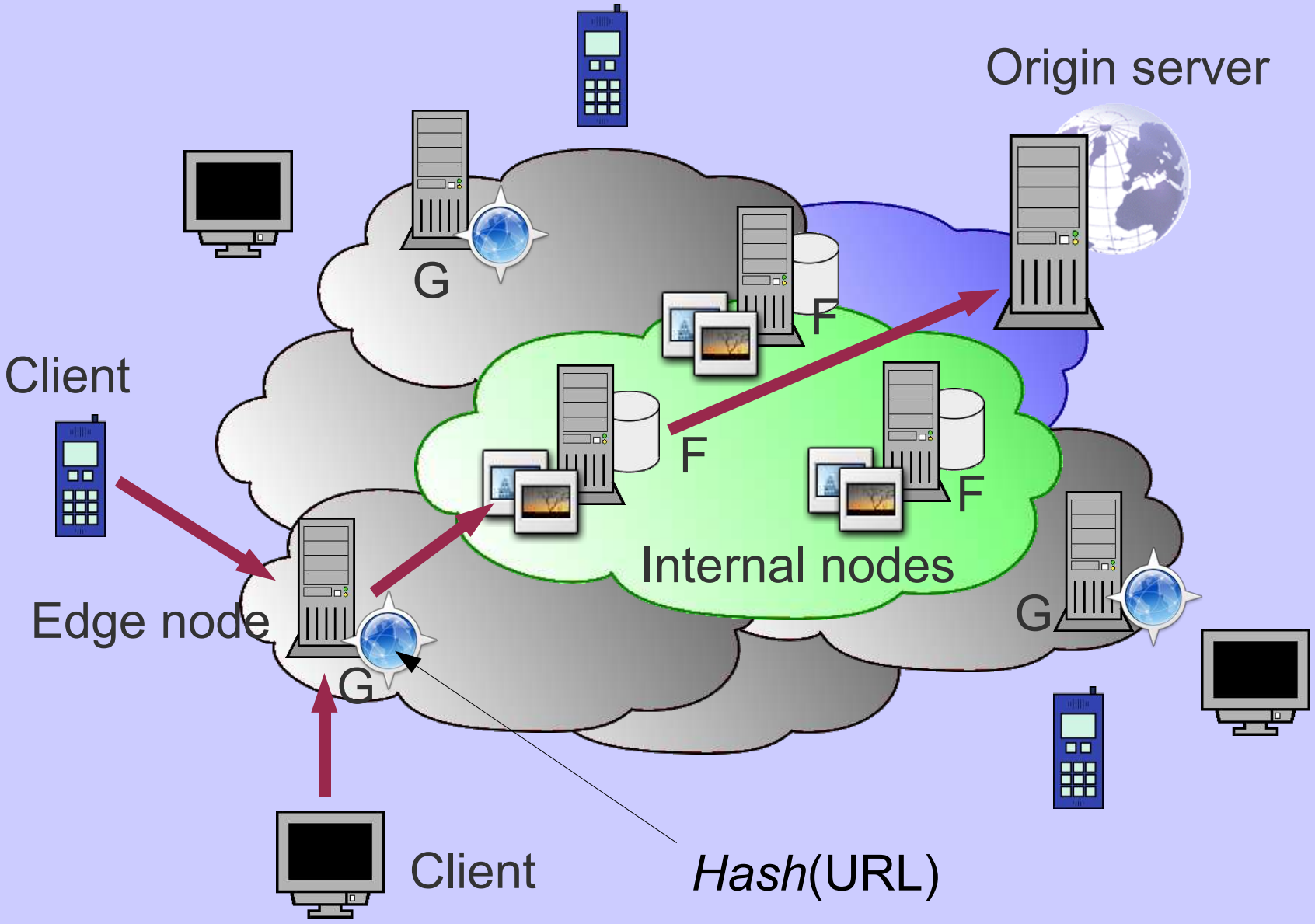
We propose a novel architecture for distributed content adaptation that preserves locality and provides load sharing

Functions in content adaptation

- Main functions:
  - Gateway → G
  - Location 
  - Adaptation 
  - Fetch → F
  - Cache 

Architectures differ in the mapping of these functions

Two-level content adaptation architecture

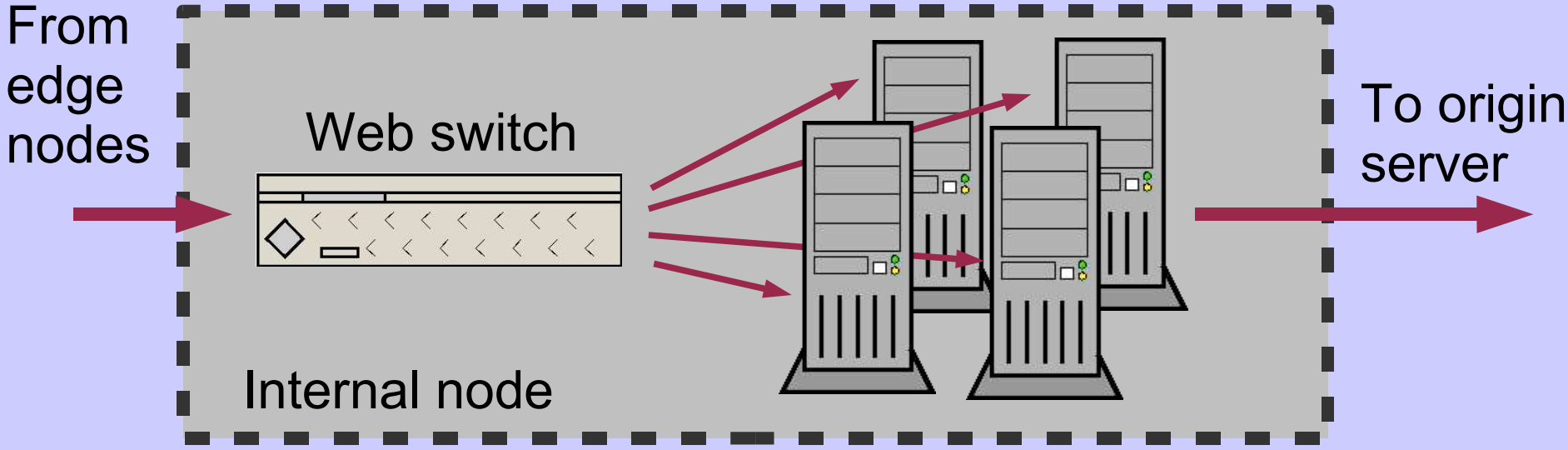


Benefit of hashing

- Hash computed only on URL (no version)
  - every version of the same resource is on the same node (simplify lookup)
  - improve locality
- URL-space partitioned
  - no cache duplicates (efficient use of cache)
- Hash-based request distribution
  - evenly distributed requests (load sharing)

Two-level content adaptation architecture

- Few powerful *internal nodes:*
  - Improves security (easy to control)
  - Improves privacy (sensitive information)
  - Solves management issues (few nodes)
  - Solves data consistency issues (hashing)
  - Internal nodes can be locally replicated to fur-ther increase computational power (cluster)

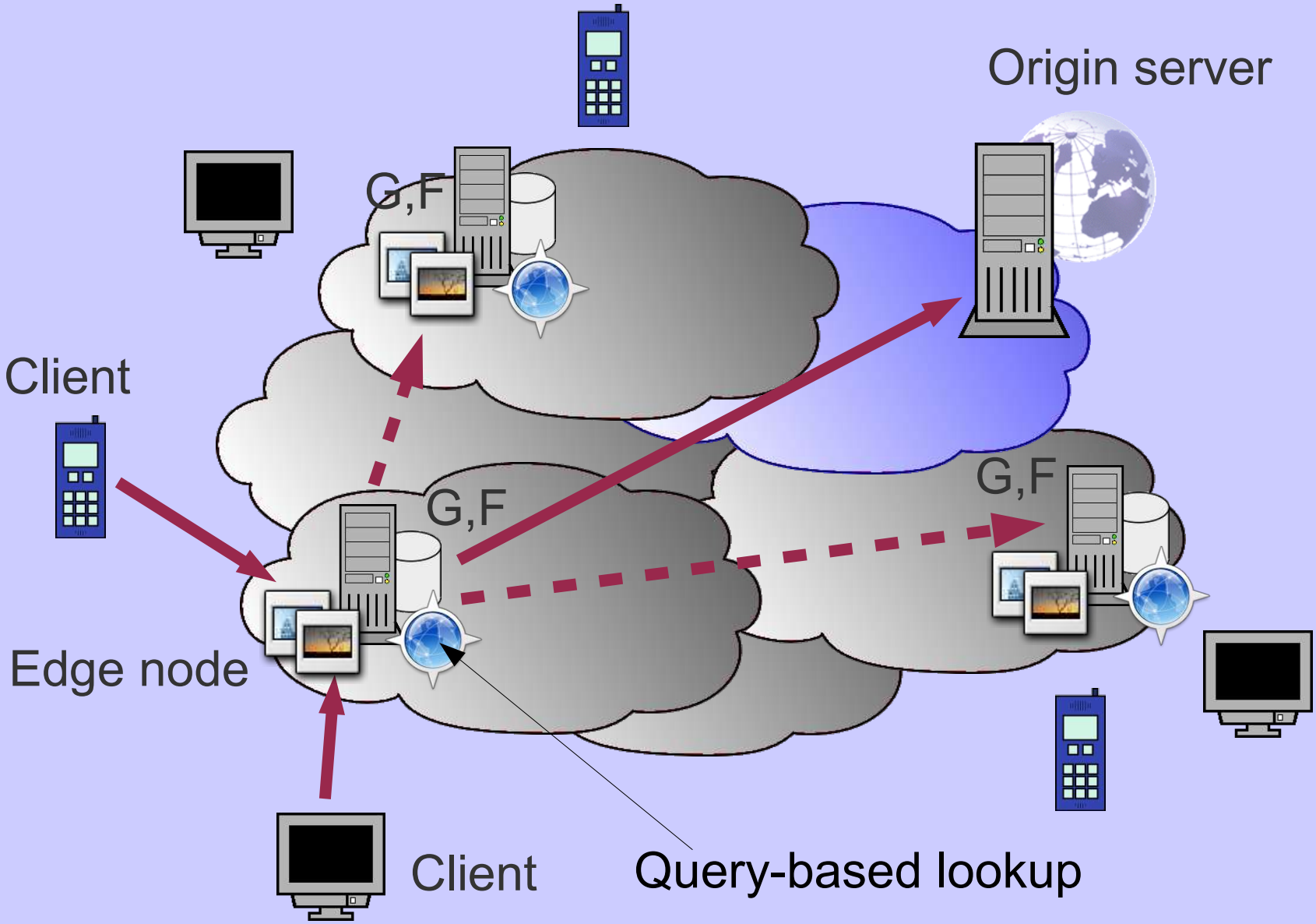


Two-level content adaptation architecture

- Many simple *edge nodes*
  - no management required
  - no computational power required
  - can be highly distributed

- Drawback of having two levels
  - Two steps for every request
  - We compare the two-level architecture with a *flat architecture*

Flat content adaptation architecture



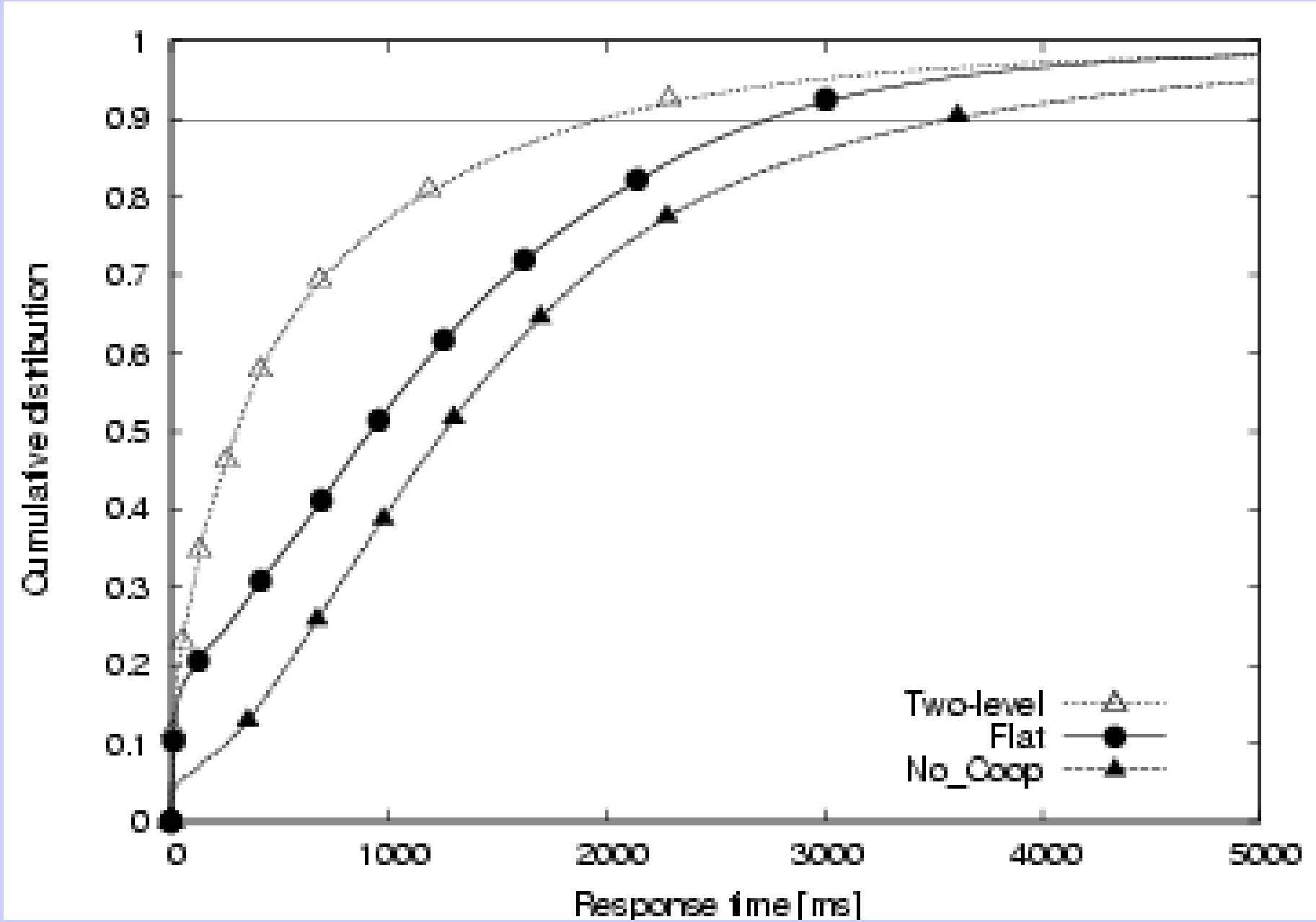
Flat content adaptation architecture

- Pros:
  - Highly distributed
  - No need for two steps
- Cons:
  - Privacy issues
  - Data consistency issues
  - Does not guarantee load sharing

Performance evaluation

- Workload models:
  - Working set with heavy impact on adaptation
  - Synthetically generated traces
- We compare:
  - **Two-level** architecture
  - **Flat** architecture
  - **No cooperation** architecture
- Two network scenario:
  - Real network scenario
  - WAN-emulated scenario

Real network scenario



Hit rate

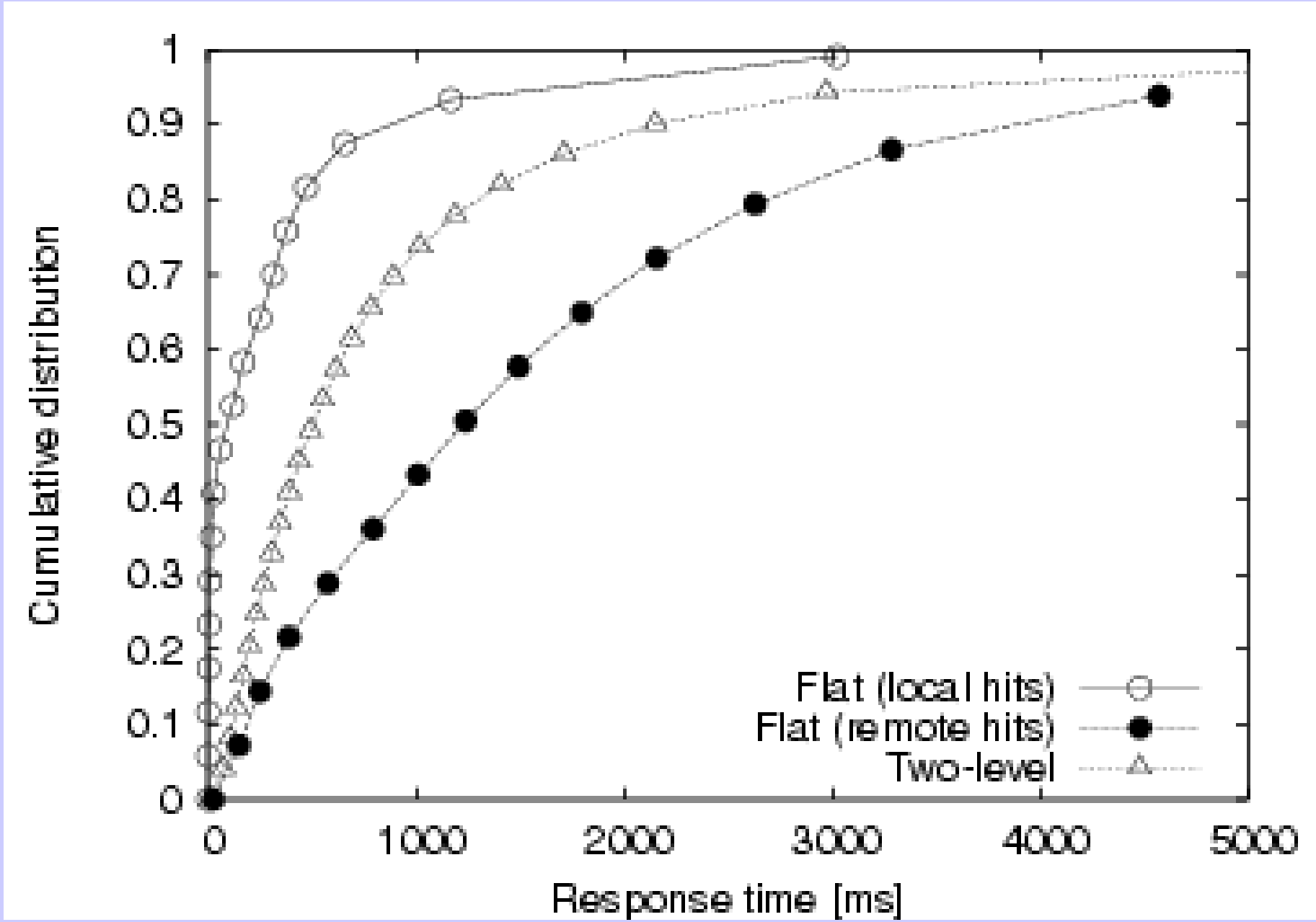
	local		remote		global
	exact	**useful**	**exact**	**useful**	
No_Coop	8.3%	6.4%	n/a	n/a	14.7%
Flat	8.0%	7.0%	25.1%	26.8%	66.9%
Two-level	n/a	n/a	60.2%	21.0%	81.2%

- Two-level provides the highest hit rate
- Hash-based partition is effective in optimizing cache usage

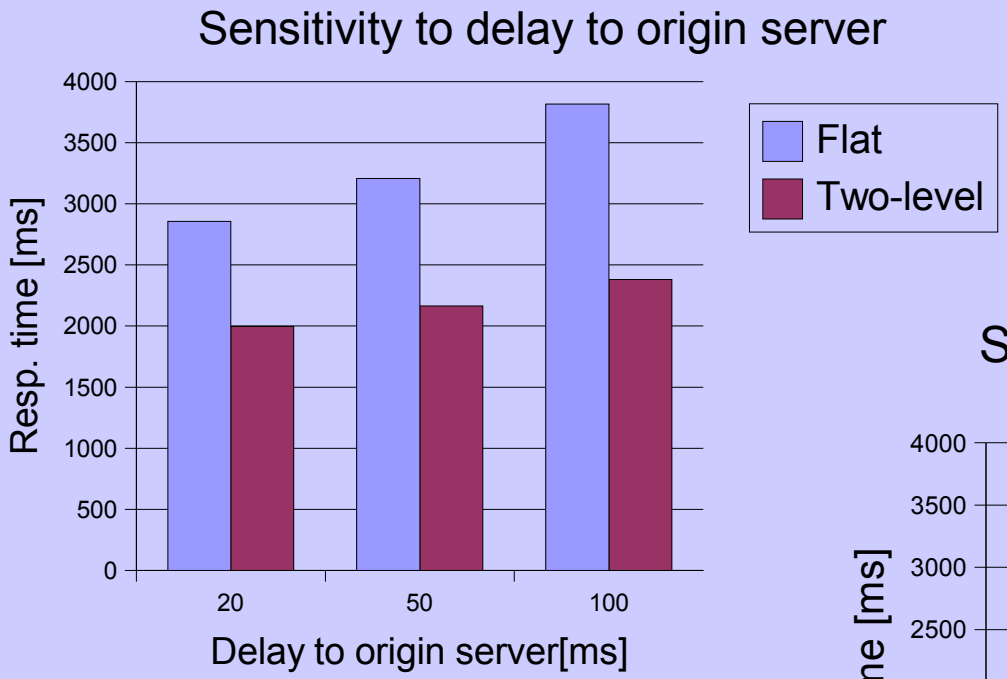
Summary of findings

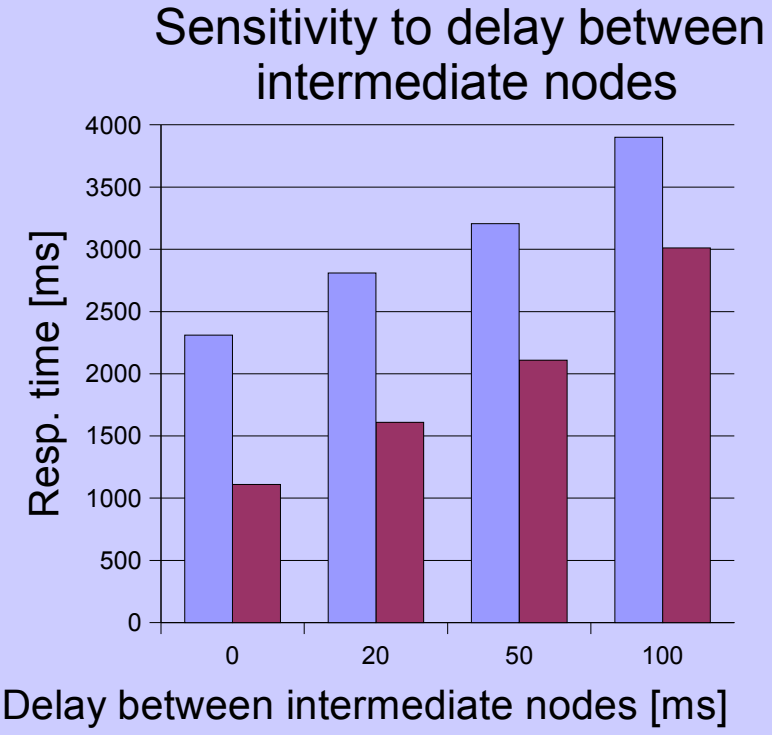
- Two-level architecture:
  - Hashing avoids duplicates $\rightarrow$ efficient cache space usage, high hit rate
  - High cache hit rate $\rightarrow$ load is reduced
- Focus on two-step penalty: sensitivity to network

Two-step penalty



Sensitivity to network parameters





- Two-level:
  - High hit rate → less sensitive to delay to origin server
  - Two steps → sensitivity to delay between edge and internal nodes

Conclusion

- Two-level architecture constantly outperforms other architecture in our experiments
- Two-level architecture is sensitive to network delays between the nodes of the intermediate infrastructure
- Two level architecture is less sensitive than flat architecture to delay to origin server

Future work

- Flat architecture and two-level are two extreme cases
  - Flat: every node provides every function
  - Two-level: node functions partitioned
- *In medio stat virtus*
  - Intermediate hybrid architectures are a whole new space of investigation

A two-level distributed architecture for Web content adaptation and delivery

Claudia Canali
University of Parma

Valeria Cardellini
University of Rome
"Tor vergata"

Michele Colajanni
University of Modena

Riccardo Lancellotti
University of Modena

Philip S. Yu
IBM T.J. Watson
research center