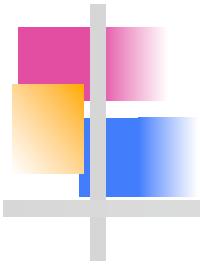


EAD: An Efficient and Adaptive Decentralized File Replication Algorithm in P2P File Sharing Systems

Haiying (Helen) Shen

Dept. of Computer Science & Computer Engineering

University of Arkansas

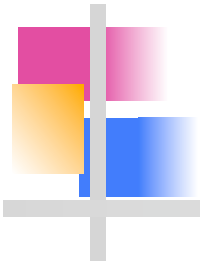
Arkansas, USA

Presenter: Arno Wacker

University of Duisburg-Essen

Distributed System

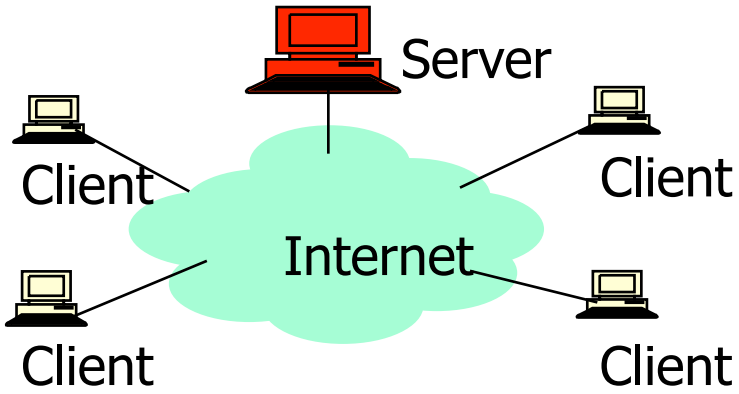
Duisburg, Germany

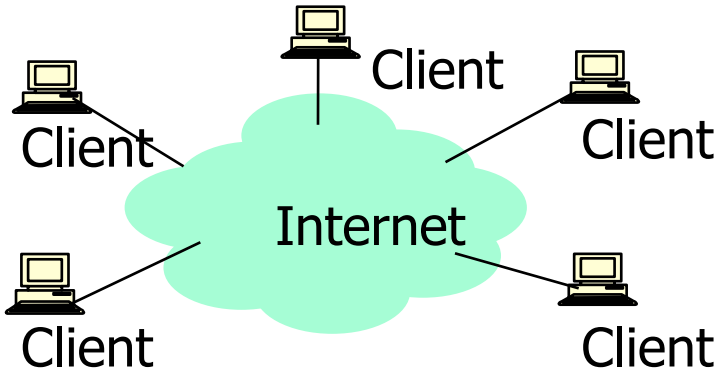


Outline

- Introduction
- Related work
- EAD file replication algorithm
- Evaluation
- Conclusion

Peer-to-Peer (P2P) System

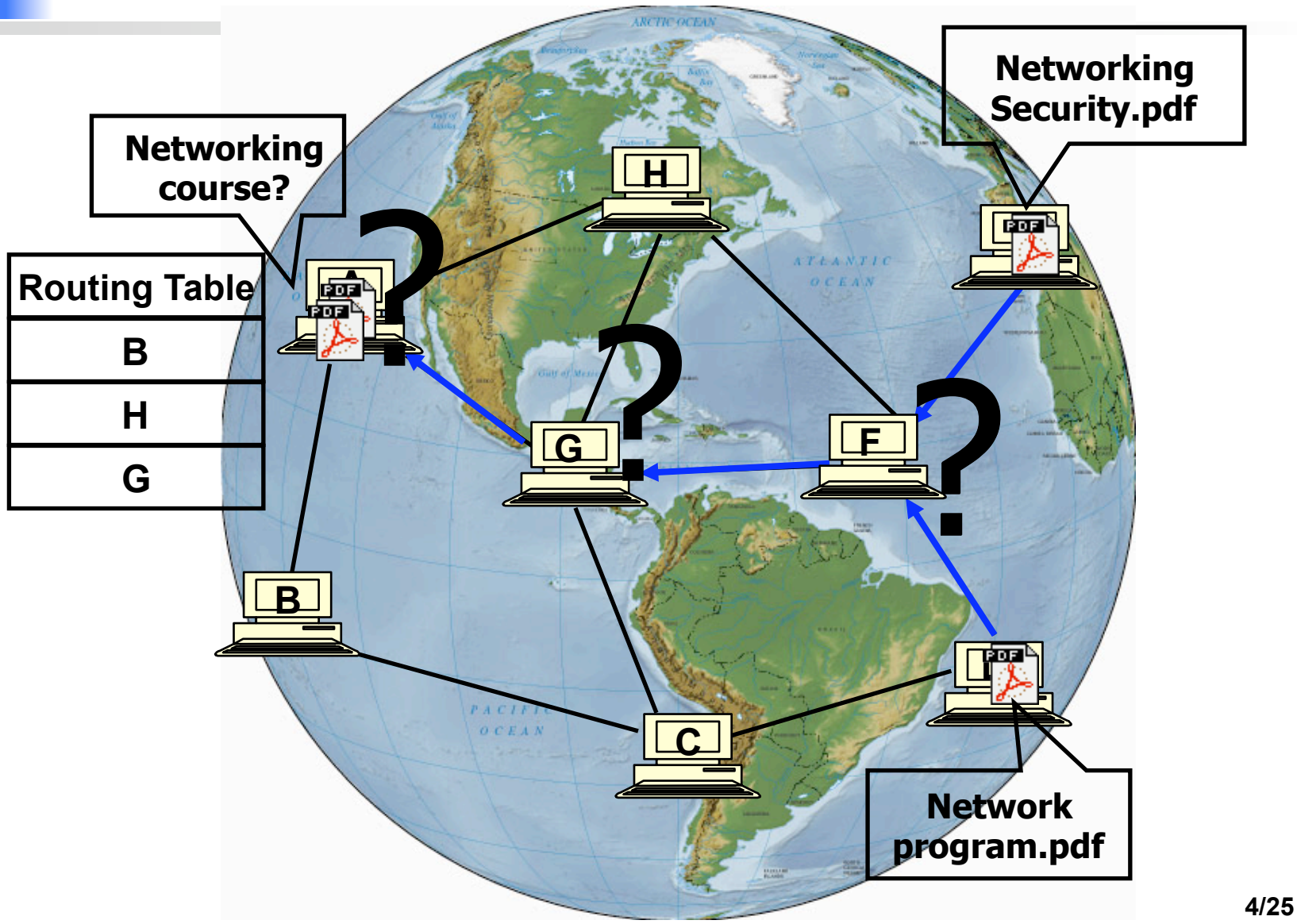




- Client/Server architecture
- Centralized system
  - Scalability is hard to achieve
- HTTP, FTP, Web services

- P2P architecture
- Decentralized system
  - No bottleneck
- BT download, Overnet

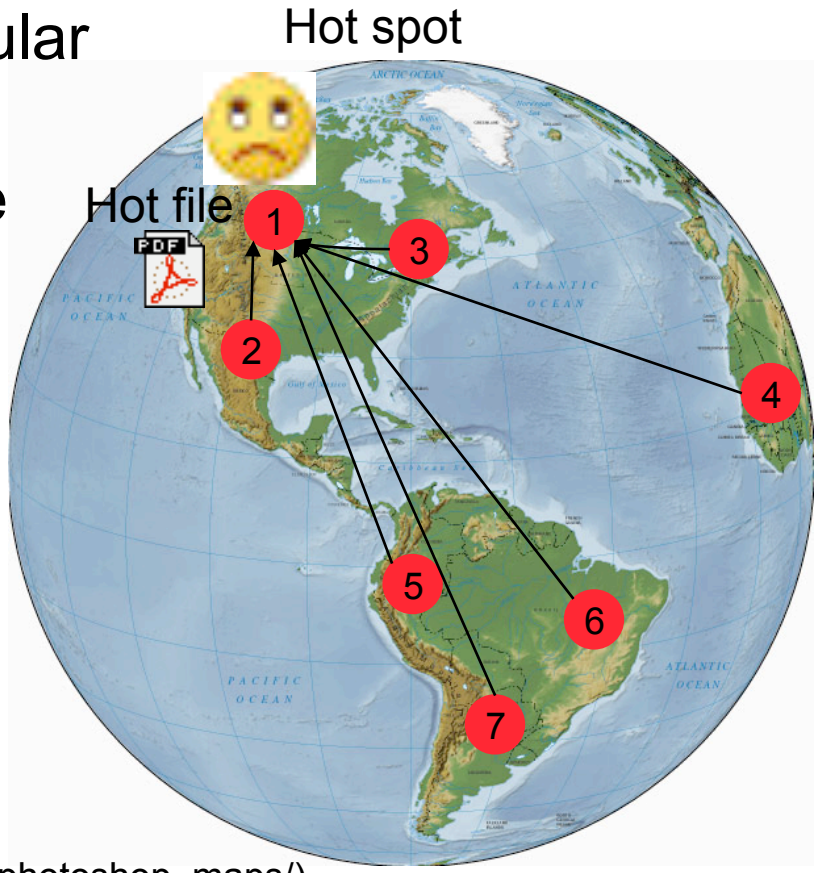
P2P File Sharing System



(Images downloaded from http://www.mapresources.com/photoshop_maps/)

Introduction & Motivation

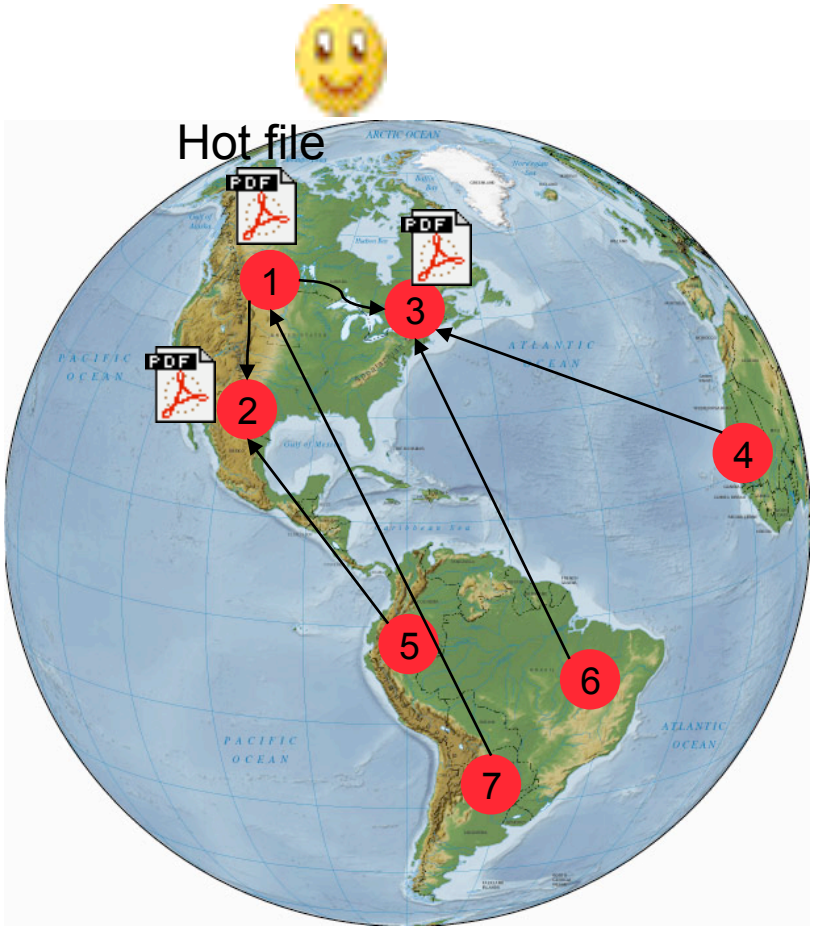
- Peer to peer file sharing system
- File access is highly repetitive and skewed towards the most popular ones
- A hot spot → delayed response
- Resolution: file replication



(Images downloaded from http://www.mapresources.com/photoshop_maps/)

Introduction & Motivation (cont.)

- Resolution: file replication
  - Replicate a hot file to some other nodes
  - Distribute the file query load among a number of nodes
  - Avoid hot spots→ improve file query efficiency

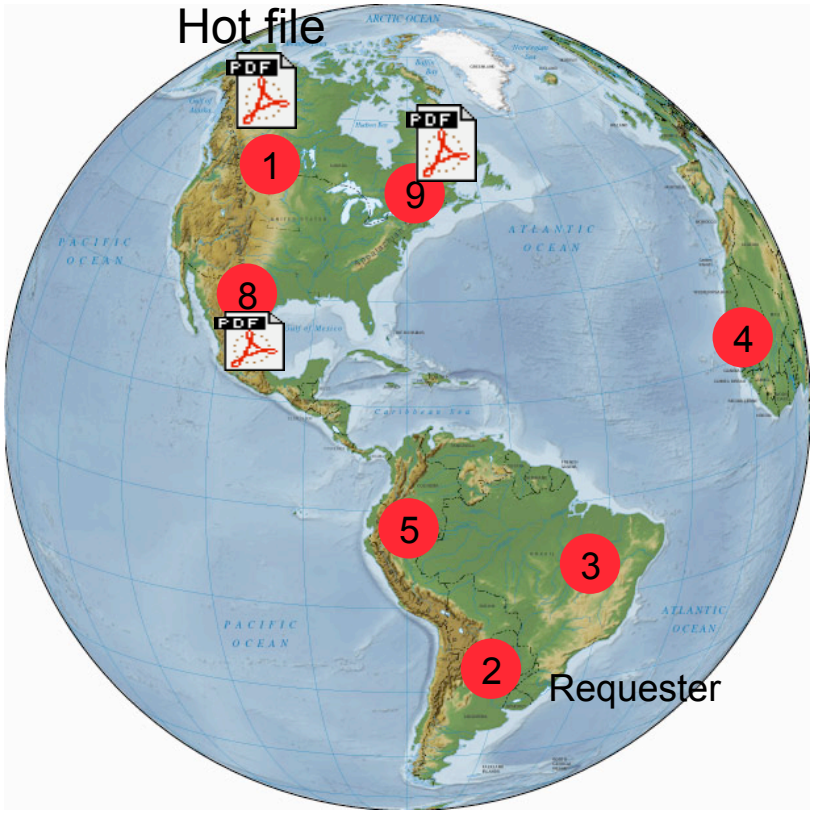


Related Work

- Replicate a file to nodes close to the file owner
  - A. Rowstron and P. Druschel. Storage Management and Caching in PAST, a Large-scale, Persistent Peer-to-Peer Storage Utility. In *Proc. of SOSP*, 2001.

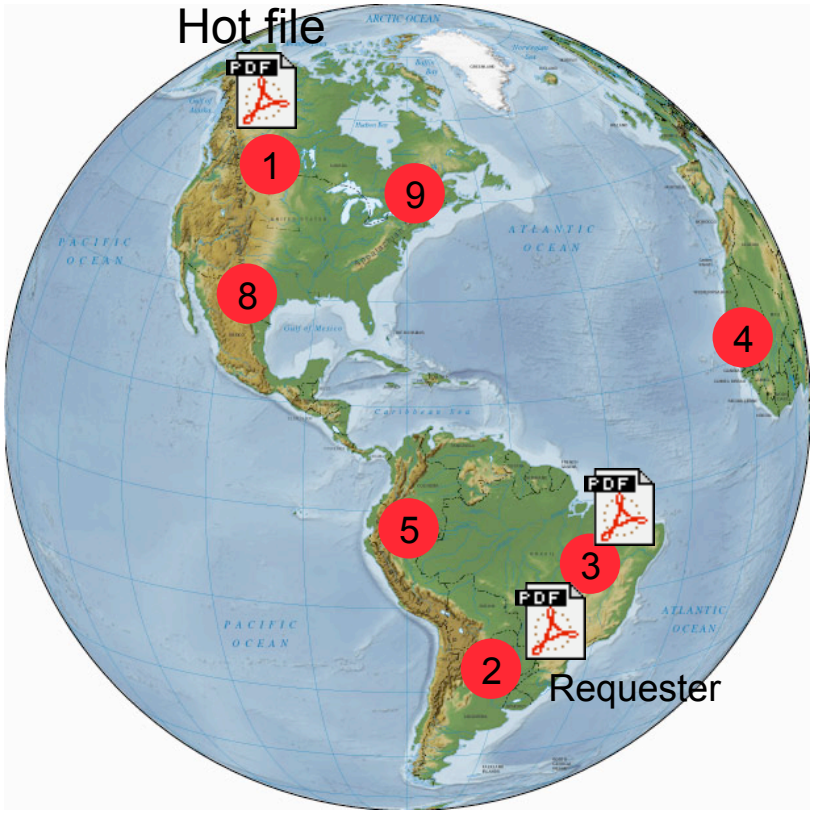
- Advantage:
  - Enhance replica hit rate
  - Enhance lookup efficiency

- Disadvantage:
  - Produce overloaded nodes
  - Cannot significantly reduce query path length



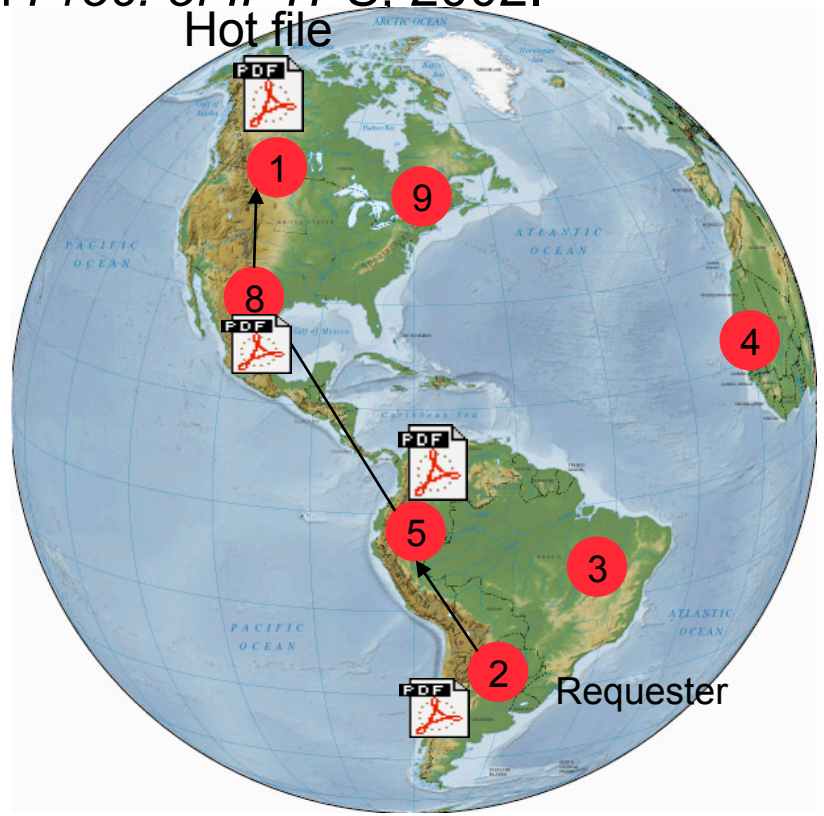
Related Work (cont.)

- Replicate a file to requesters
  - V. Gopalakrishnan, B. Silaghi, and et al. Adaptive Replication in Peer-to-Peer Systems. In Proc. of ICDCS, 2004.
- Advantage:
  - Enhance lookup efficiency
- Disadvantage:
  - Low replica hit rate
    since replicas cannot shared
    by other nodes



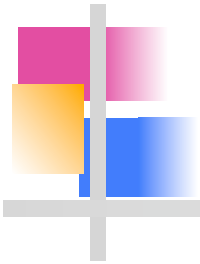
Related Work (cont.)

- Replicate a file to nodes along the query path
  - R. Cox, A. Muthitacharoen, and R. T. Morris. Serving DNS using a Peer-to-Peer Lookup Service. In *Proc. of IPTPS*, 2002.



- Advantage:
  - Enhance replica hit rate
  - Significantly enhance lookup efficiency

- Disadvantage:
  - High overhead



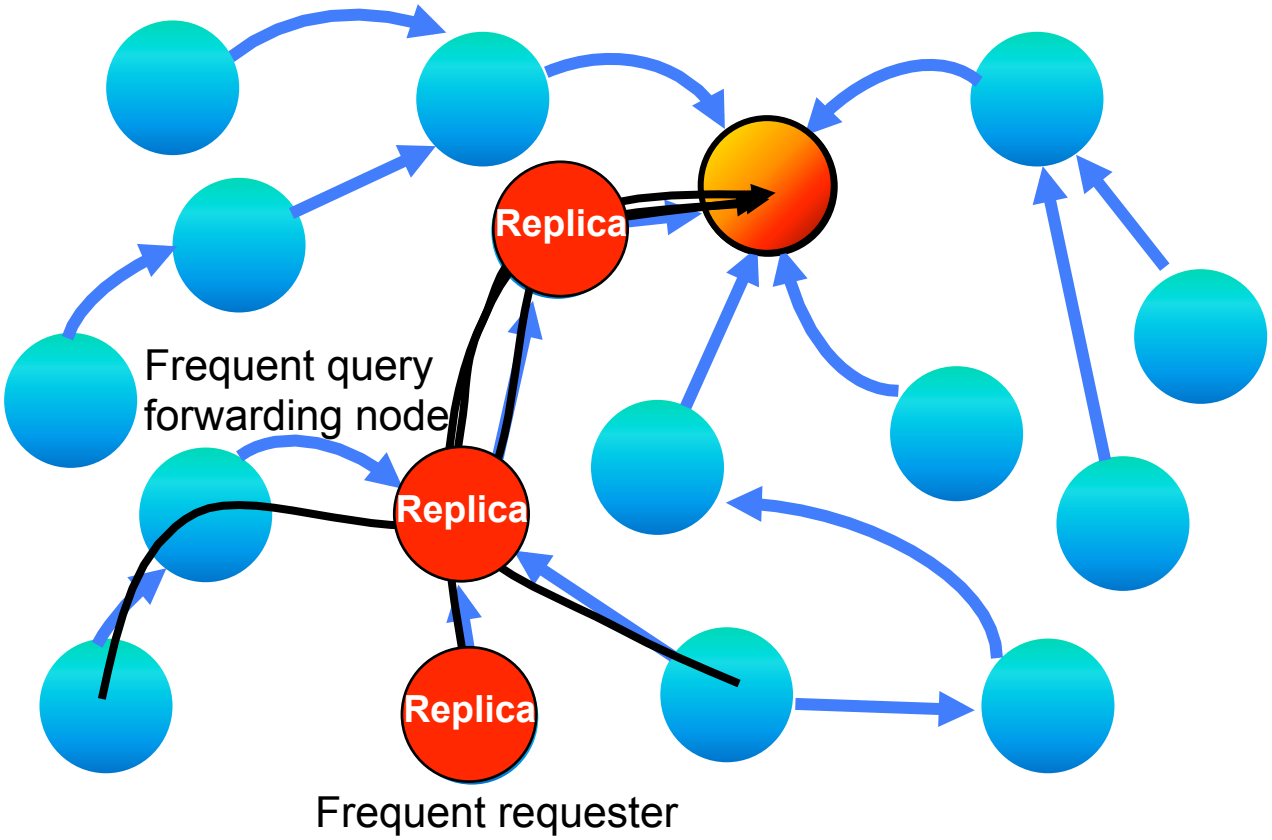
EAD: Efficient and Adaptive Decentralized File Replication

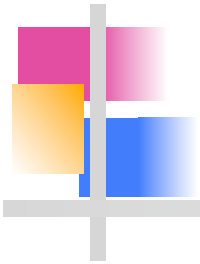
- Goals:
  - Enhance replica hit rate
  - Significantly enhance lookup efficiency
  - Low overhead

- Problems to achieve the goals:
  - (1) where to replicate files so that the file query can be significantly expedited and the replicas can be fully utilized?
  - (2) how to remove under-utilized file replicas so that the overhead for consistency maintenance is minimized?

Overview of EAD

- File replication:
  - Traffic junction nodes carry more query traffic load
  - Replica nodes: frequent requesters & traffic junction nodes

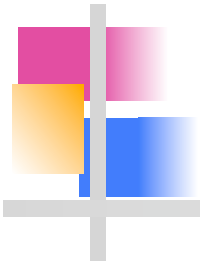




EAD Efficient File Replication

- In P2P file sharing systems, some nodes carry more query traffic load than others

- Reasons:
  - Node interests are different. There will be more query traffic along the query paths from the frequent file requesters and the file owner
  - File popularity is non-uniform and time-varying
  - Nodes are located in different places and may have different number of neighbors in P2P overlay network



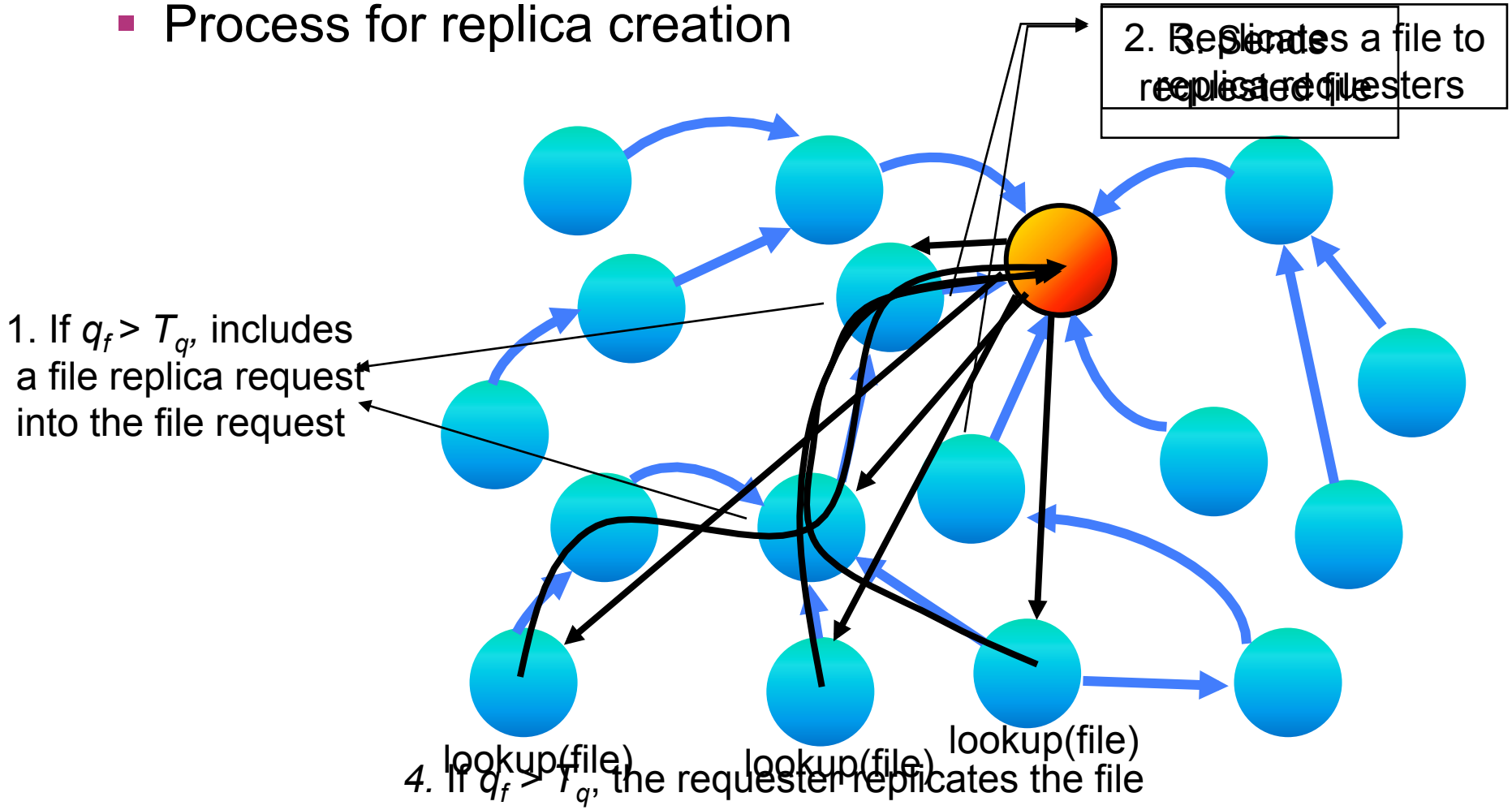
EAD Efficient File Replication -- Replica creation

- EAD replicates a file in nodes that
  - carry more query traffic of the file
  - query the query frequently

- *Query rate* for file f: q_f
  - The number of queries for *f* initiated or forwarded by the node during a unit time
  - Threshold for query rate: T_q
  - When a file requester's $q_f > T_q$, the node adds a replica request into the file request
  - The file owner sends the file to the replica requesters

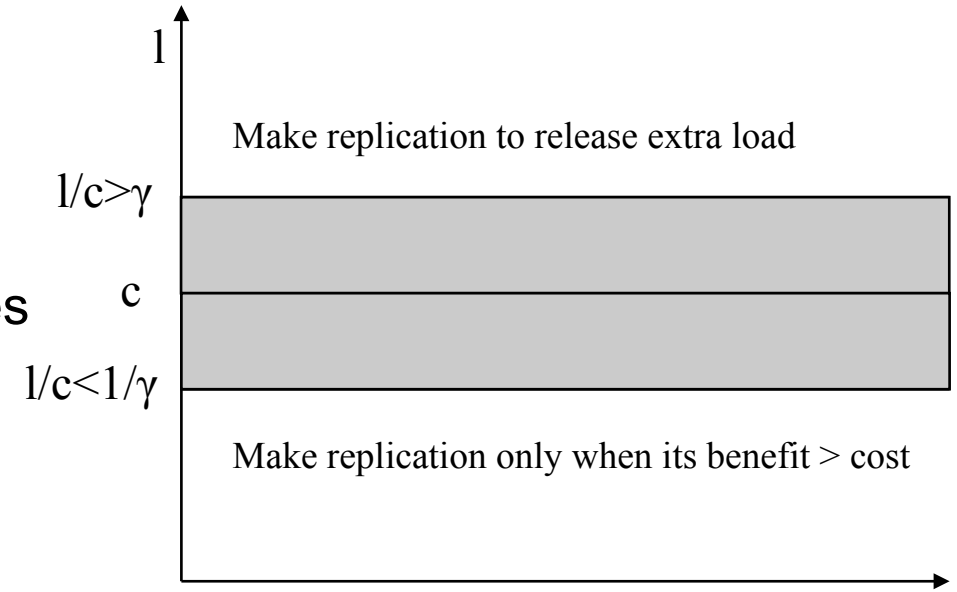
EAD Efficient File Replication -- Replica creation (cont.)

- Process for replica creation



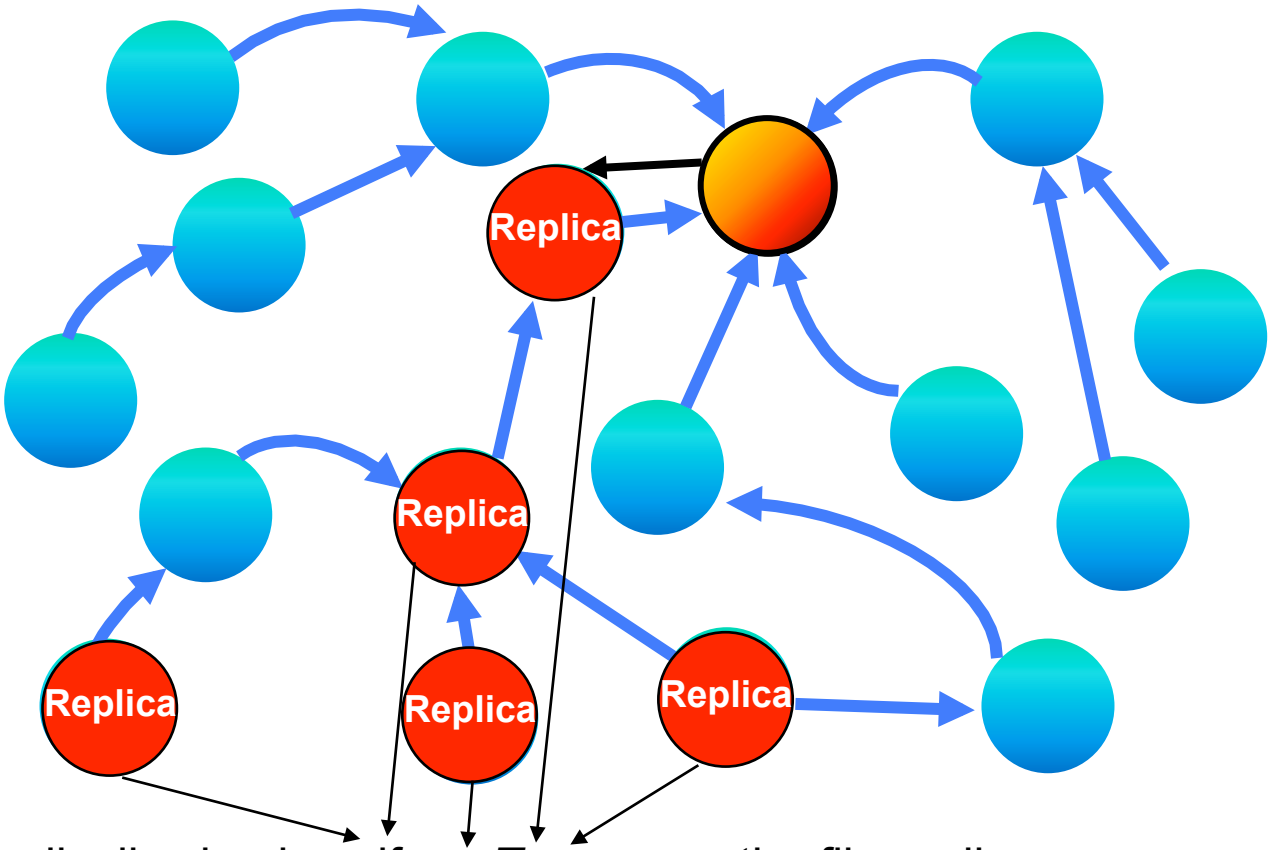
EAD Efficient File Replication -- Replica creation (cont.)

- Server: the original file owner and replica nodes
- A server's visit rate represents its query load l
- A node's capacity c: the number of queries it can respond during T
- Factor: γ_l
- Server i periodically checks l_i
- $l_i / c_i > \gamma_l$:
  - release $(l_i - \gamma_l c_i)$ by replicating files
- $l_i / c_i < 1/ \gamma_l$:
  - replicate file when benefit>cost

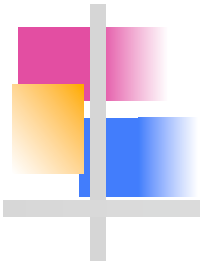


EAD Efficient File Replication -- Replica adaptation

- File replicas become under-utilized when there are few queries for the files
- Replica adaptation: guarantee high replica utilization



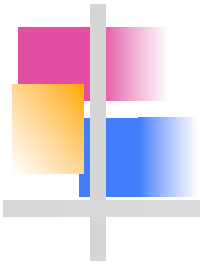
Periodically check q_f, if $q_f < T_q$, remove the file replica



Query Rate Determination

- EAD employs exponential moving average technique to reasonably determine file query rate over time period T
- EMA assigns more weight to recent observations without discarding older observations entirely
  - It applies weighting factors to older observed q_f , so that the weight for each older q_f decreases exponentially
- The observation at a time period t is designated y_t, and the value of the query rate at any time period t is designated q_{ft}
- Query rate:

$$q_{f\,t} = \beta y_{t-1} + (1-\beta) q_{f\,t-1}$$

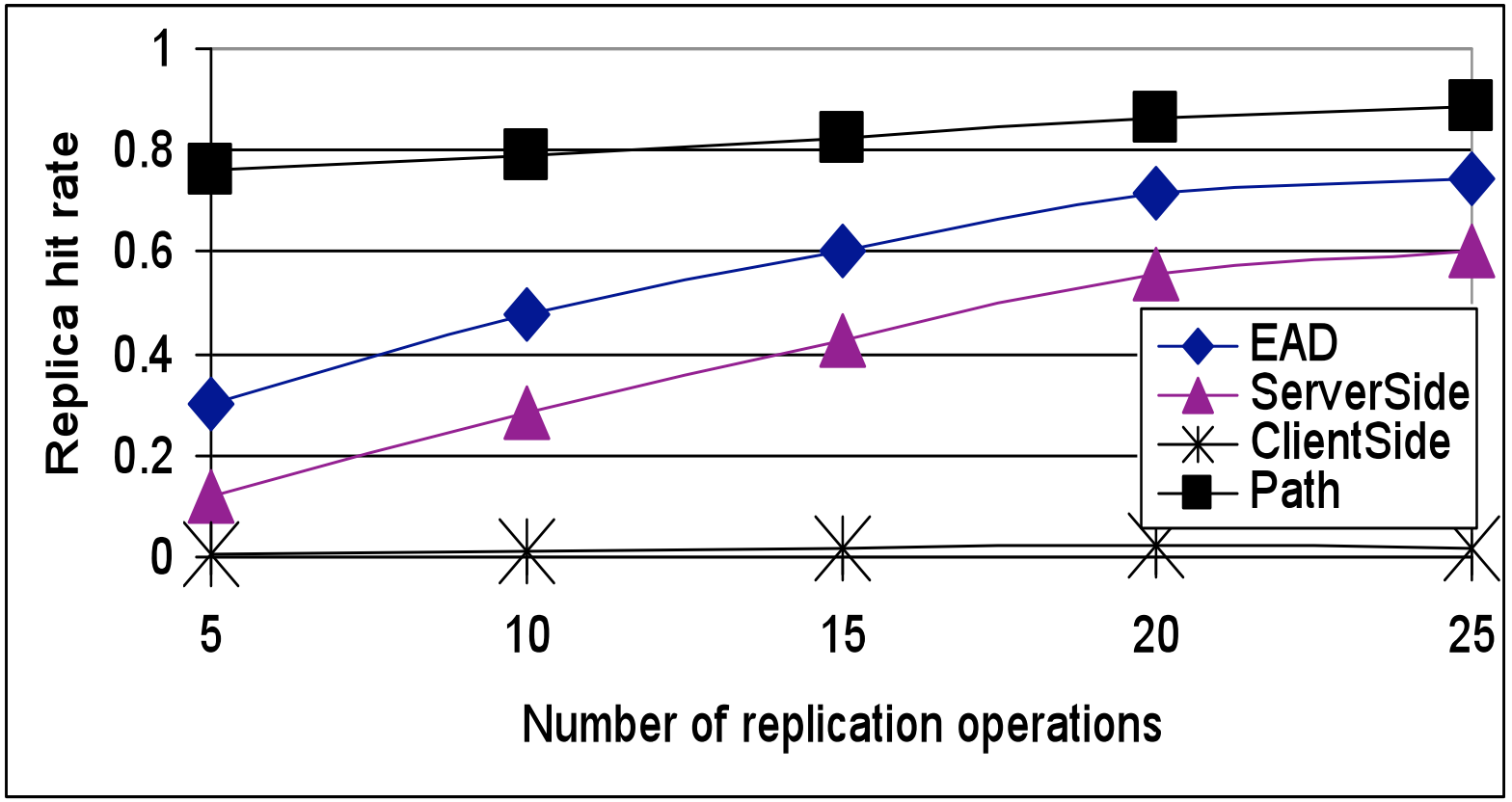


Performance Evaluation

- Experiments
  - Compare EAD with ServerSide[1], ClientSide[2] and Path[3]
  - Number of nodes: 4096
  - Number of queried files: 50
  - Number of queries per file: 1000
  - Number of replication operations: 5-25
- Performance
  - Replica hit rate
  - Overhead
  - Load balance
  - Lookup efficiency
  - Replica adaptation

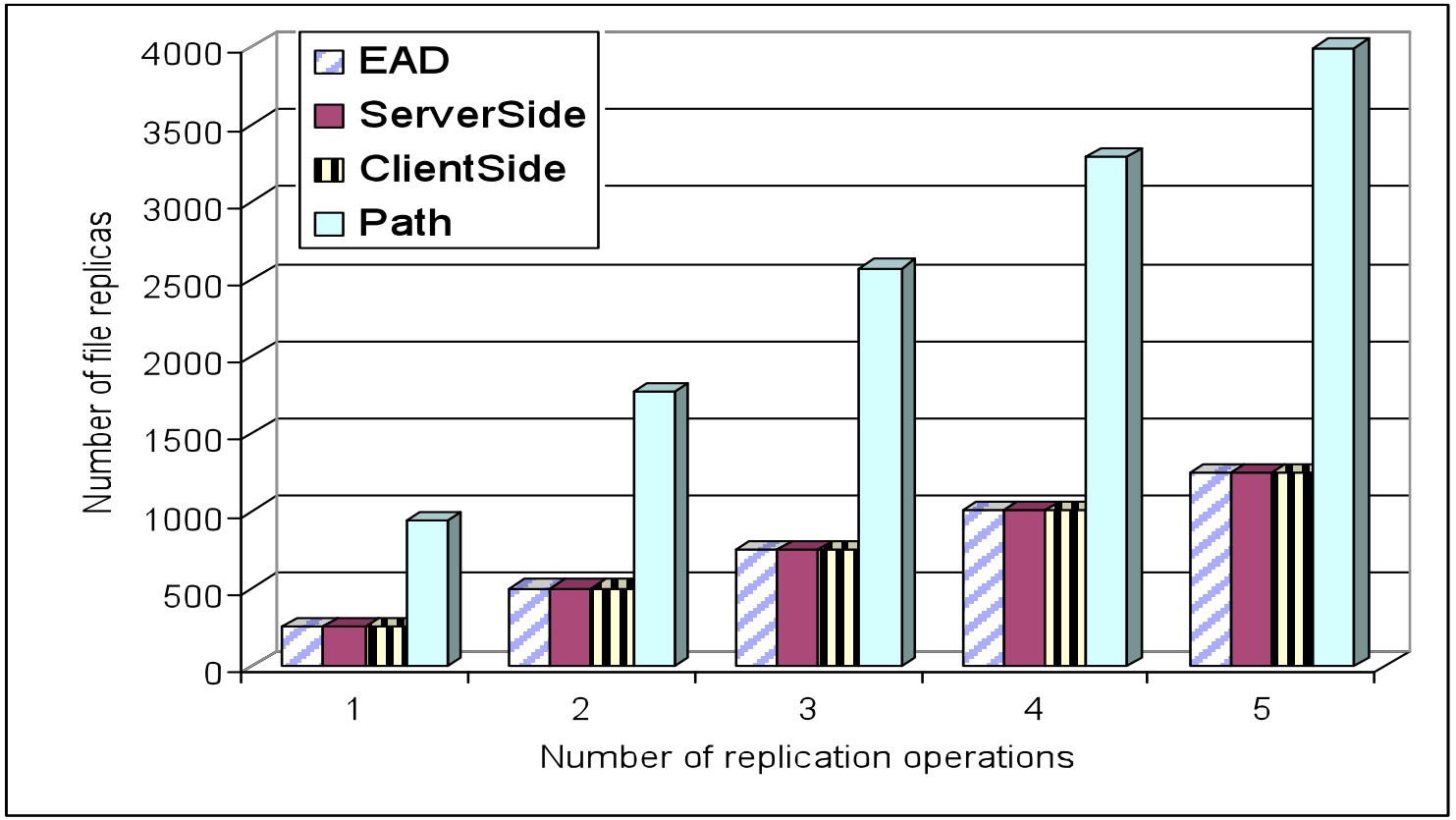
[1] A. Rowstron and P. Druschel. Storage Management and Caching in PAST, a Large-scale, Persistent Peer-to-Peer Storage Utility. In *Proc. of SOSP*, 2001.
[2] V. Gopalakrishnan, B. Silaghi, and et al. Adaptive Replication in Peer-to-Peer Systems. In *Proc. of ICDCS*, 2004.
[3] R. Cox, A. Muthitacharoen, and R. T. Morris. Serving DNS using a Peer-to-Peer Lookup Service. In *Proc. of IPTPS*, 2002.

Replica hit rate



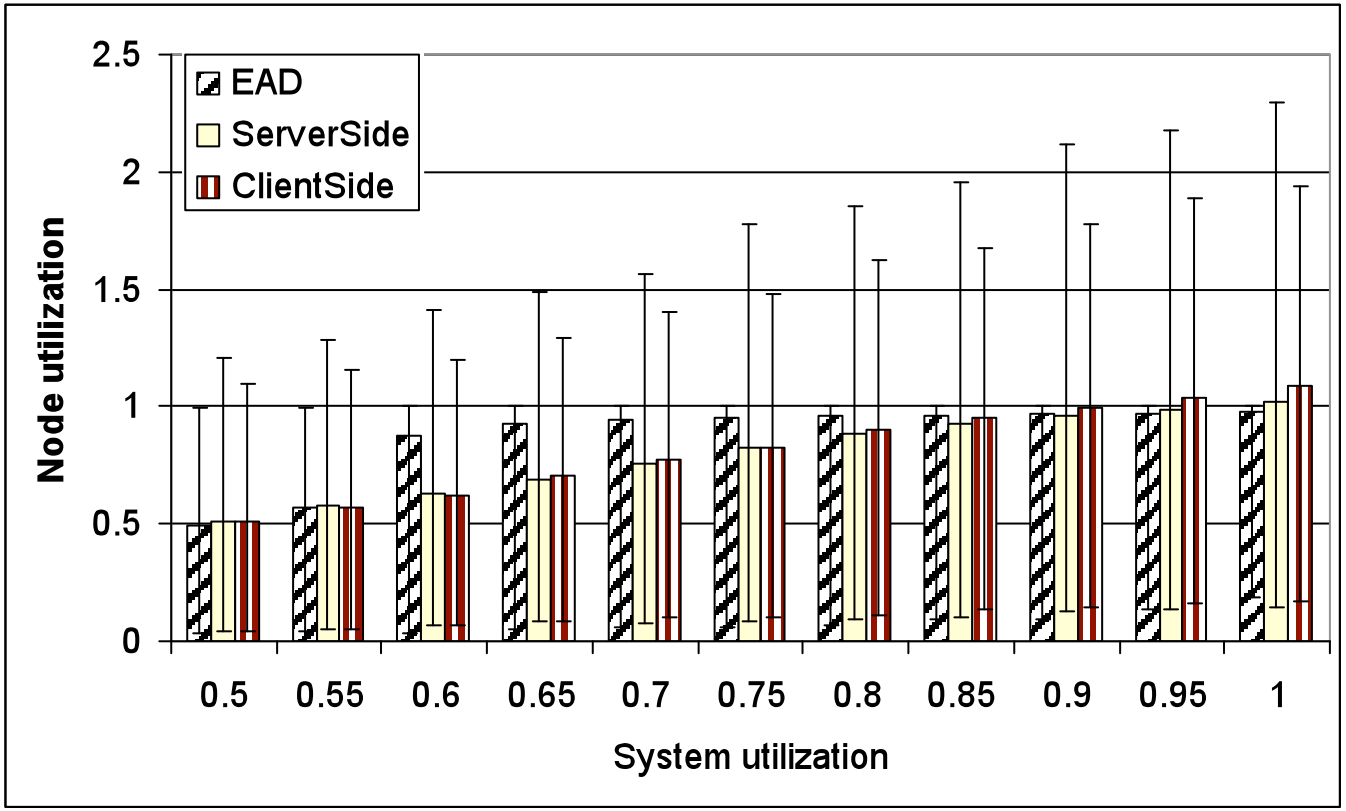
- Replica hit rate: Path>EAD>ServerSide>ClientSide
- EAD has high replica hit rate
- Path achieves high replica hit rate at cost of much more replicas

Number of Replicas



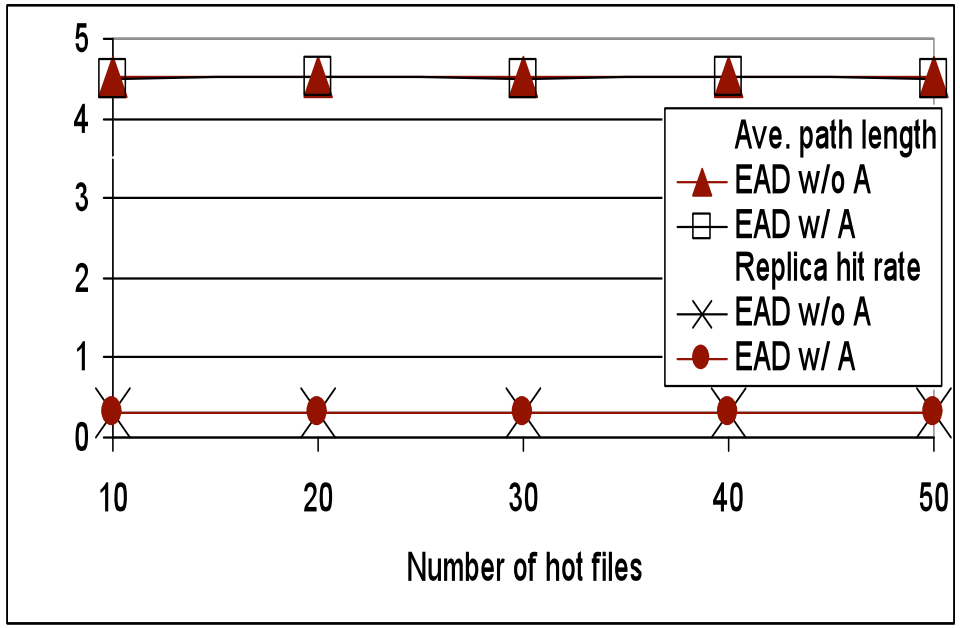
- Number of replicas: Path>EAD=ServerSide=ClientSide
- Path generates much higher overhead than other algorithms

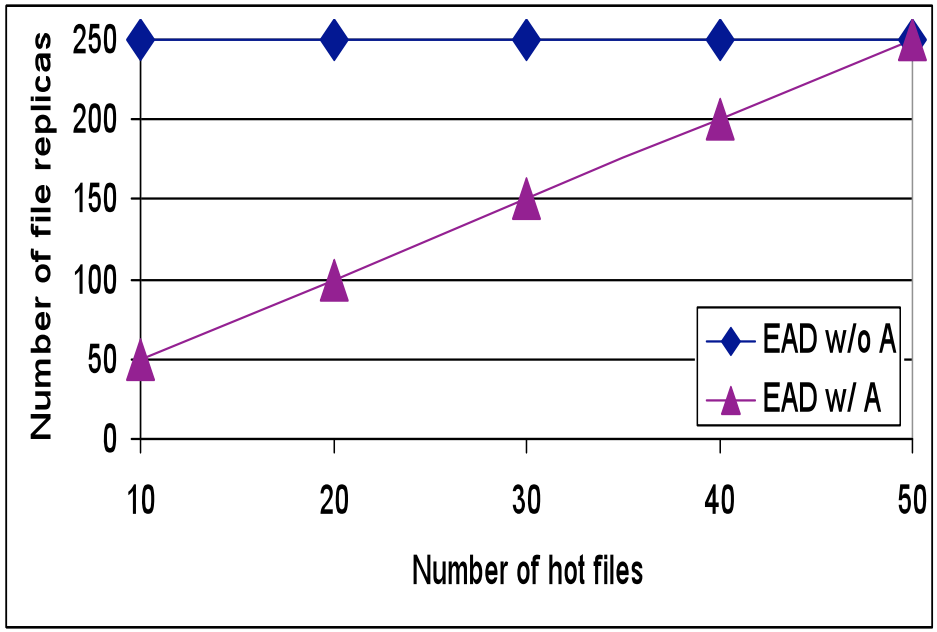
Node utilization



- Node utilization: ServerSide>ClientSide>EAD
- With node load status consideration, EAD does not overload replica nodes

Replica Adaptation



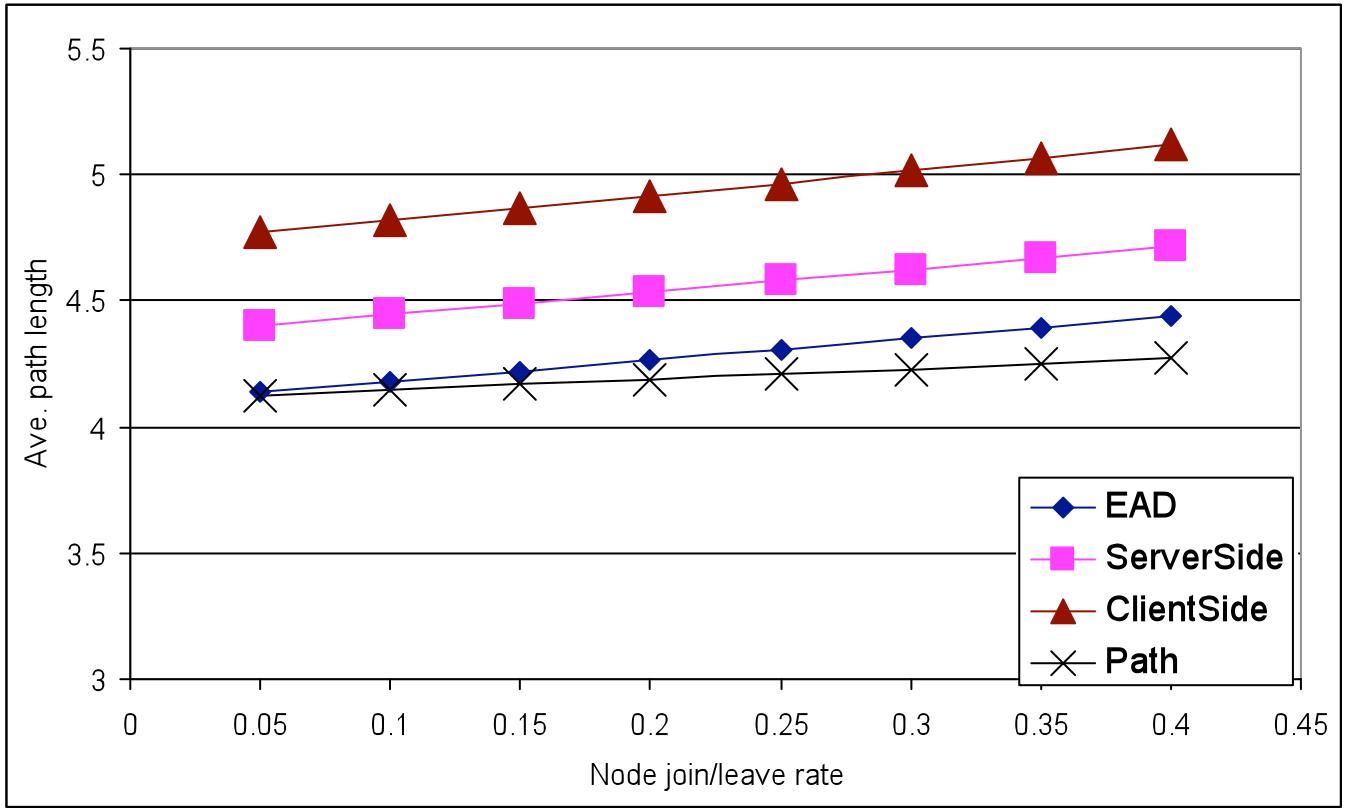


EAD w/ A: EAD with replica adaption

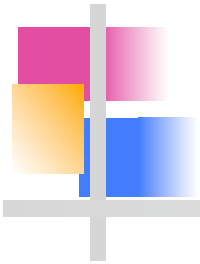
EAD w/o A: EAD without replica adaption

- Ave. path length & replica hit rate: EAD w/ A = EAD w/o A
- Number of replicas: EAD w/ A < EAD w/o A
- Replica adaption helps to reduce overhead

Ave. Path Length in Churn

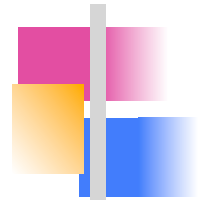


- Path length: ClientSide > ServerSide > EAD > Path
- EAD has high lookup efficiency in churn



Conclusions

- Traditional file replication methods
  - Not effective enough to improve file query efficiency
  - Incur prohibitively high overhead
- EAD file replication algorithm
  - Nodes actively determine the need for file replica creation and deletion
  - Nodes frequently request a file or forward the requests of a file become replica nodes of the file
- Features:
  - High replica hit rate
  - High lookup efficiency
  - Low overhead



Thanks.
Questions & Comments?
Please email Dr. Shen

Haiying (Helen) Shen, Assistant professor

Pervasive Communication Laboratory

University of Arkansas, Arkansas, US

HTTP://comp.uark.edu/~hshen