Squirrel: A peer-to-peer web cache

Sitaram Iyer (Rice University)

Joint work with

Ant Rowstron (MSR Cambridge)

Peter Druschel (Rice University)

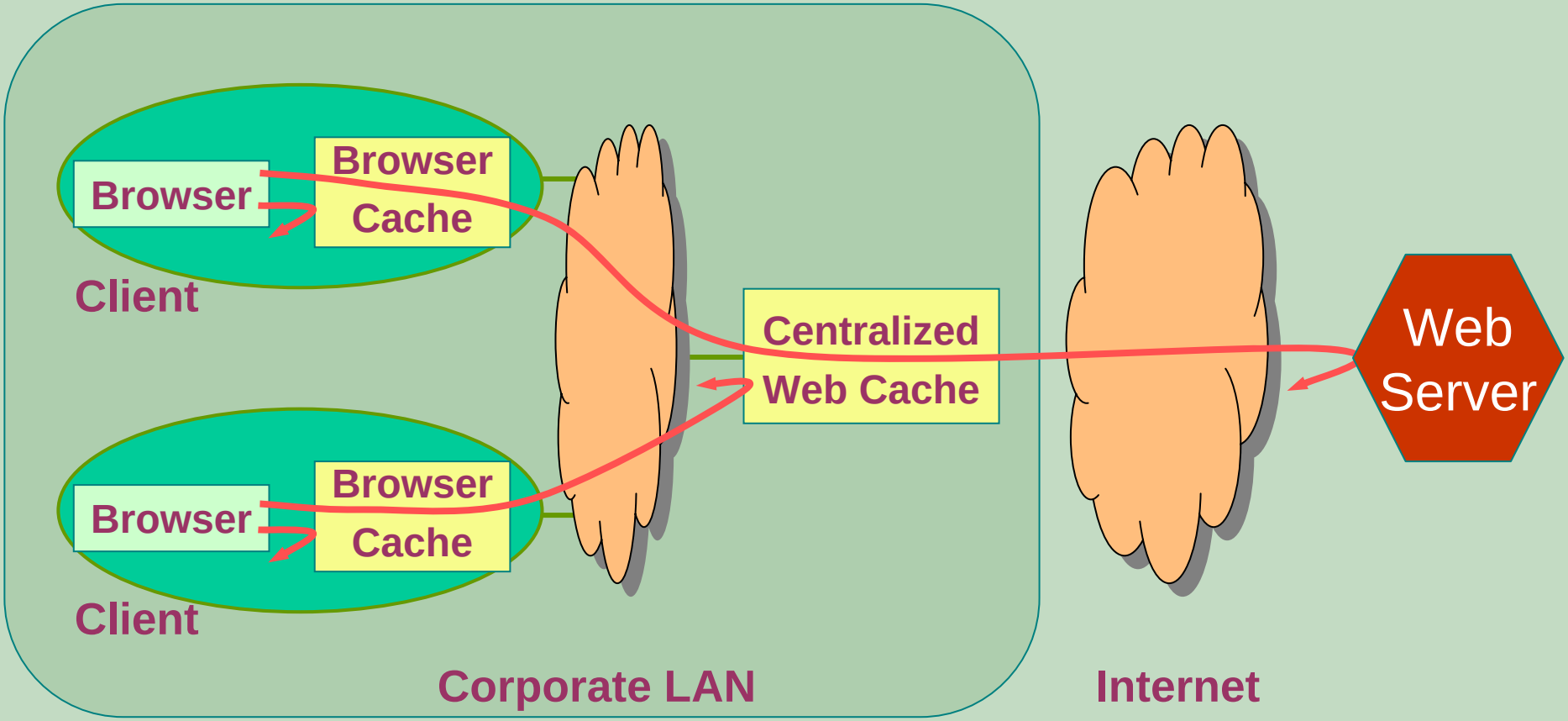
PODC 2002 / Sitaram Iyer / Tuesday July 23 / Monterey, CA

Web Caching

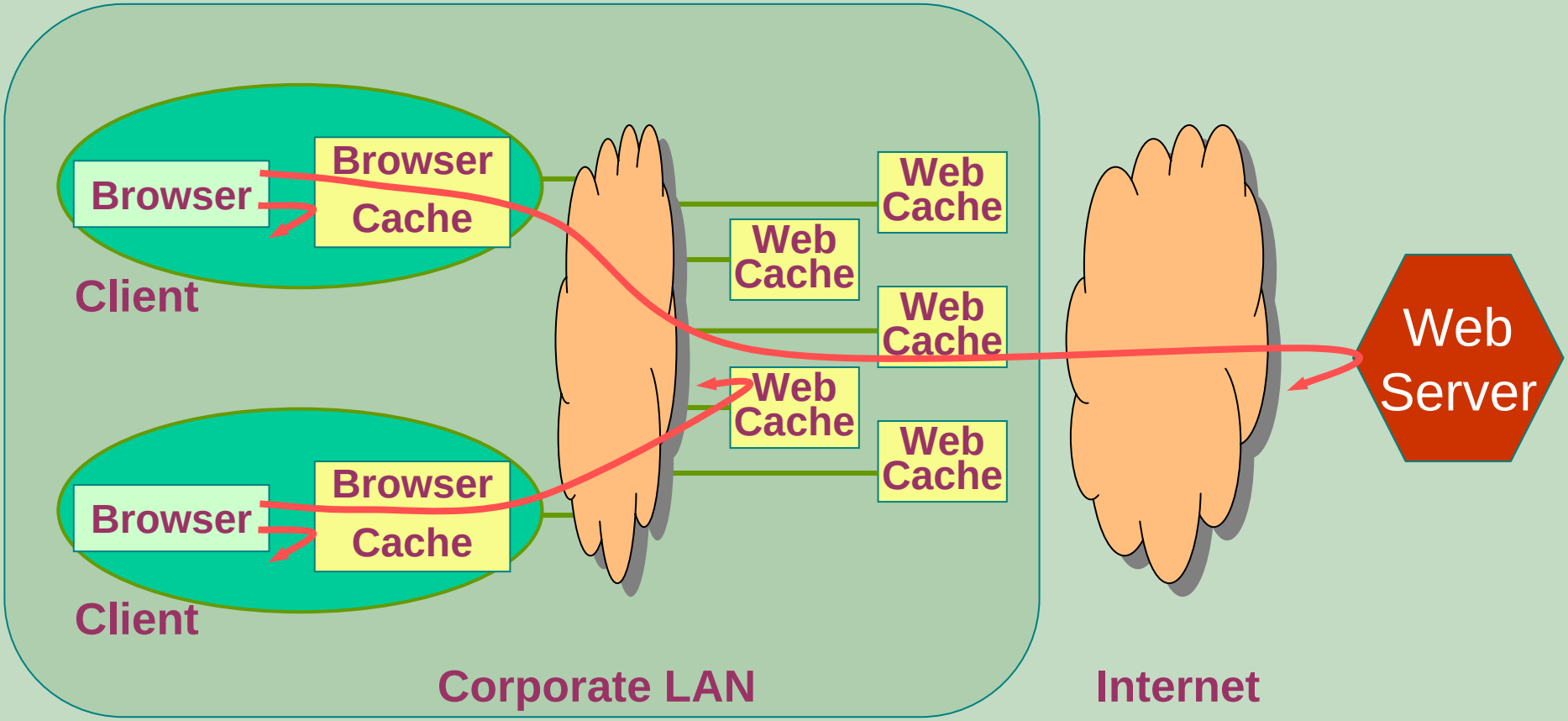
1. Latency,
2. External traffic,
3. Load on web servers and routers.

Deployed at: Corporate network boundaries, ISPs, Web Servers, etc.

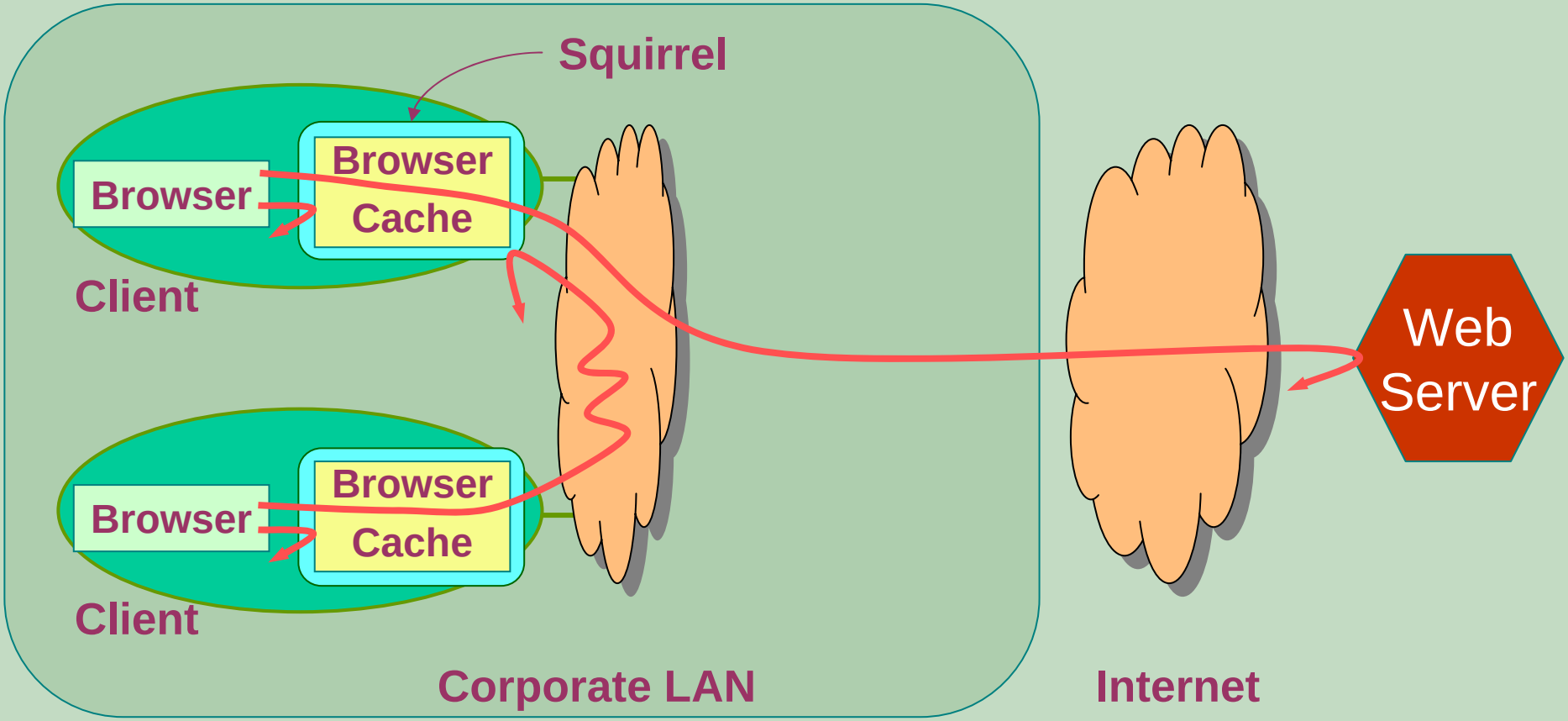
Web Cache


Browser
Browser
Cache
Client
Browser
Browser
Cache
Client
Centralized
Web Cache
Web
Server
Corporate LAN
Internet

Cooperative Web Cache

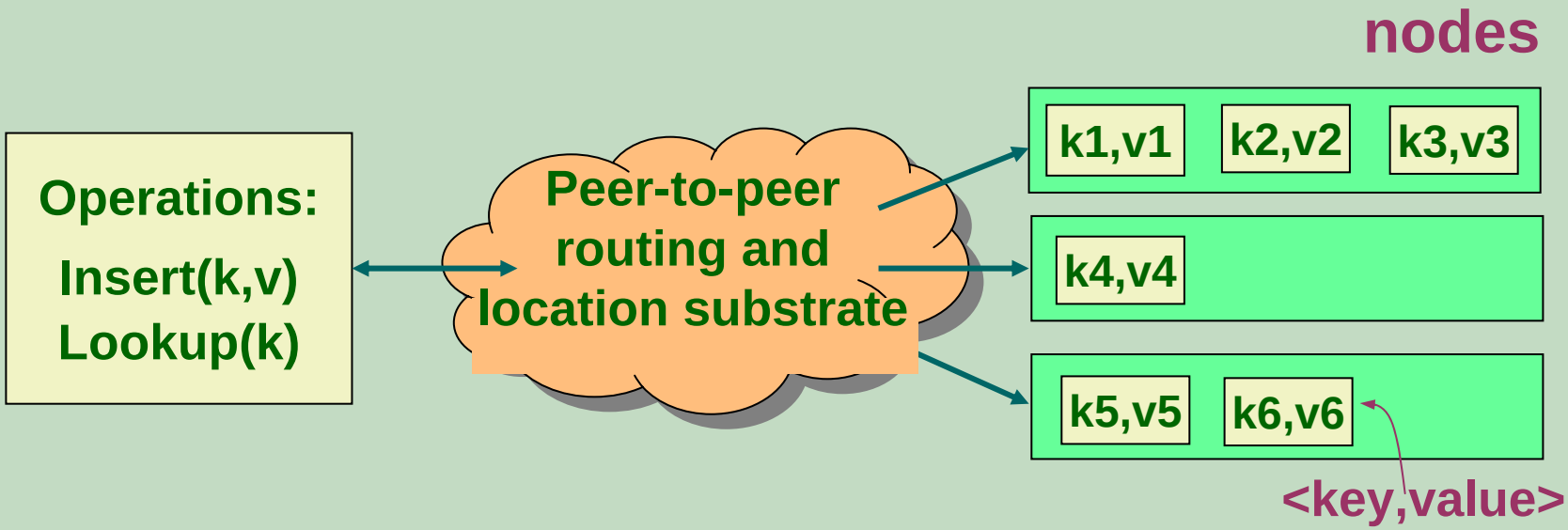


Decentralized Web Cache



Distributed Hash Table

Peer-to-peer location service: Pastry



- Completely decentralized and self-organizing
- Fault-tolerant, scalable, efficient

Why peer-to-peer?

1. Cost of dedicated web cache

   No additional hardware

2. Administrative effort

   Self-organizing network

3. Scaling implies upgrading

   Resources grow with clients

Setting

- Corporate LAN
- 100 - 100,000 desktop machines
- Located in a single building or campus

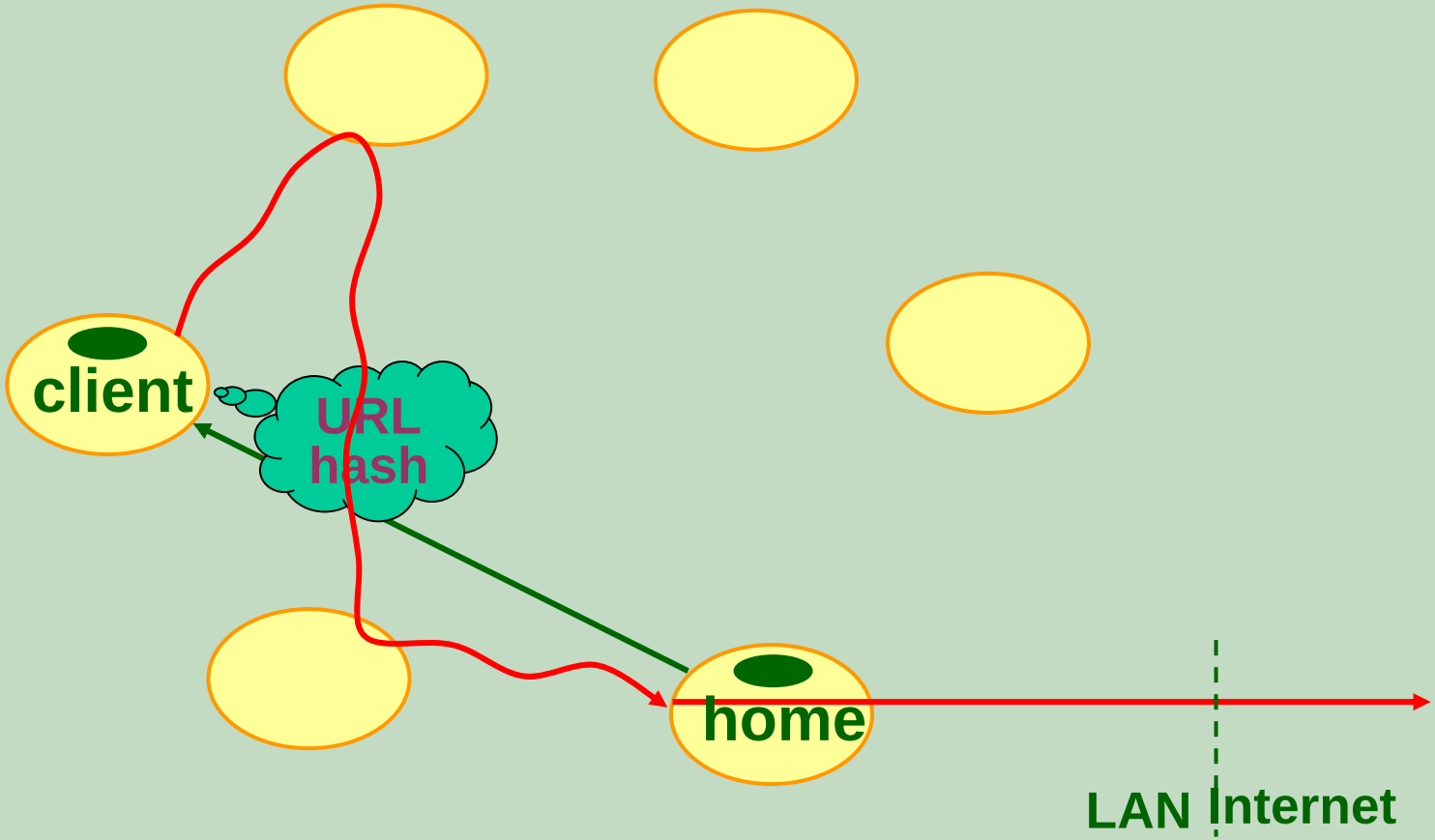
- Each node runs an instance of Squirrel
- Sets it as the browser’s proxy

Mapping Squirrel onto Pastry

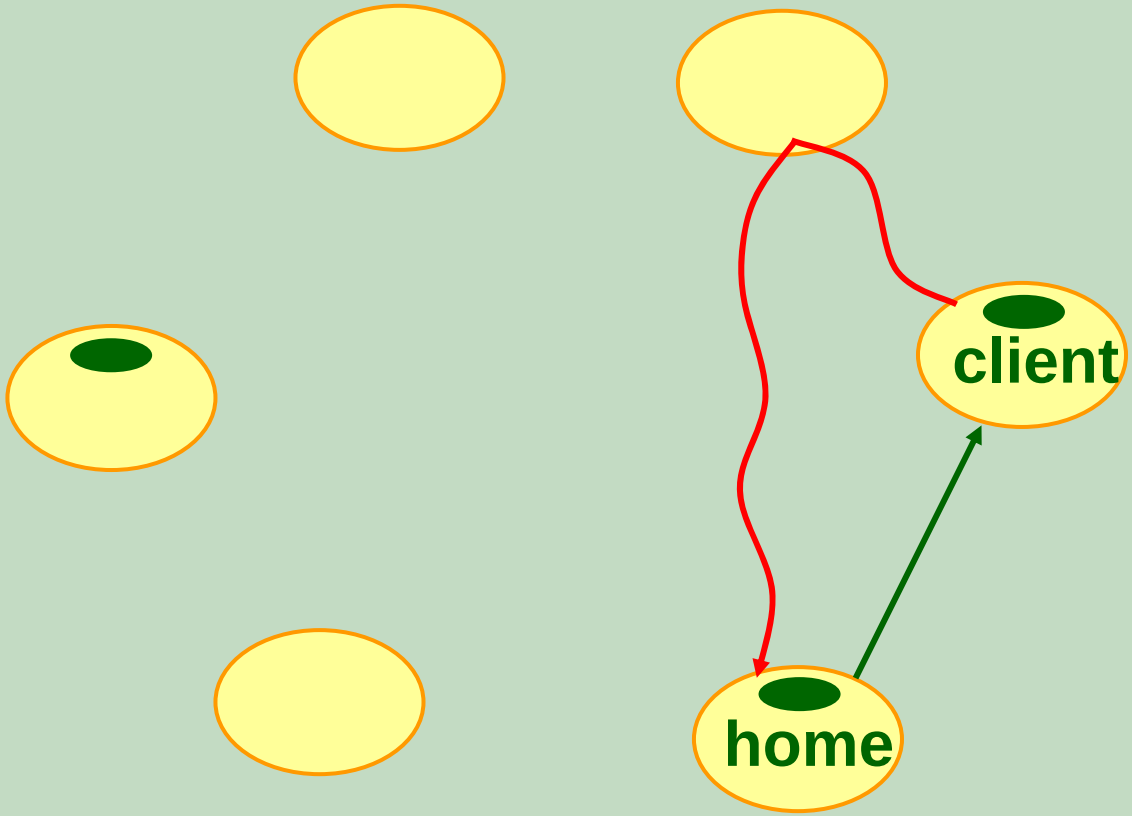
Two approaches:

- **Home-store**
- **Directory**

Home-store model



Home-store model



…that's how it works!

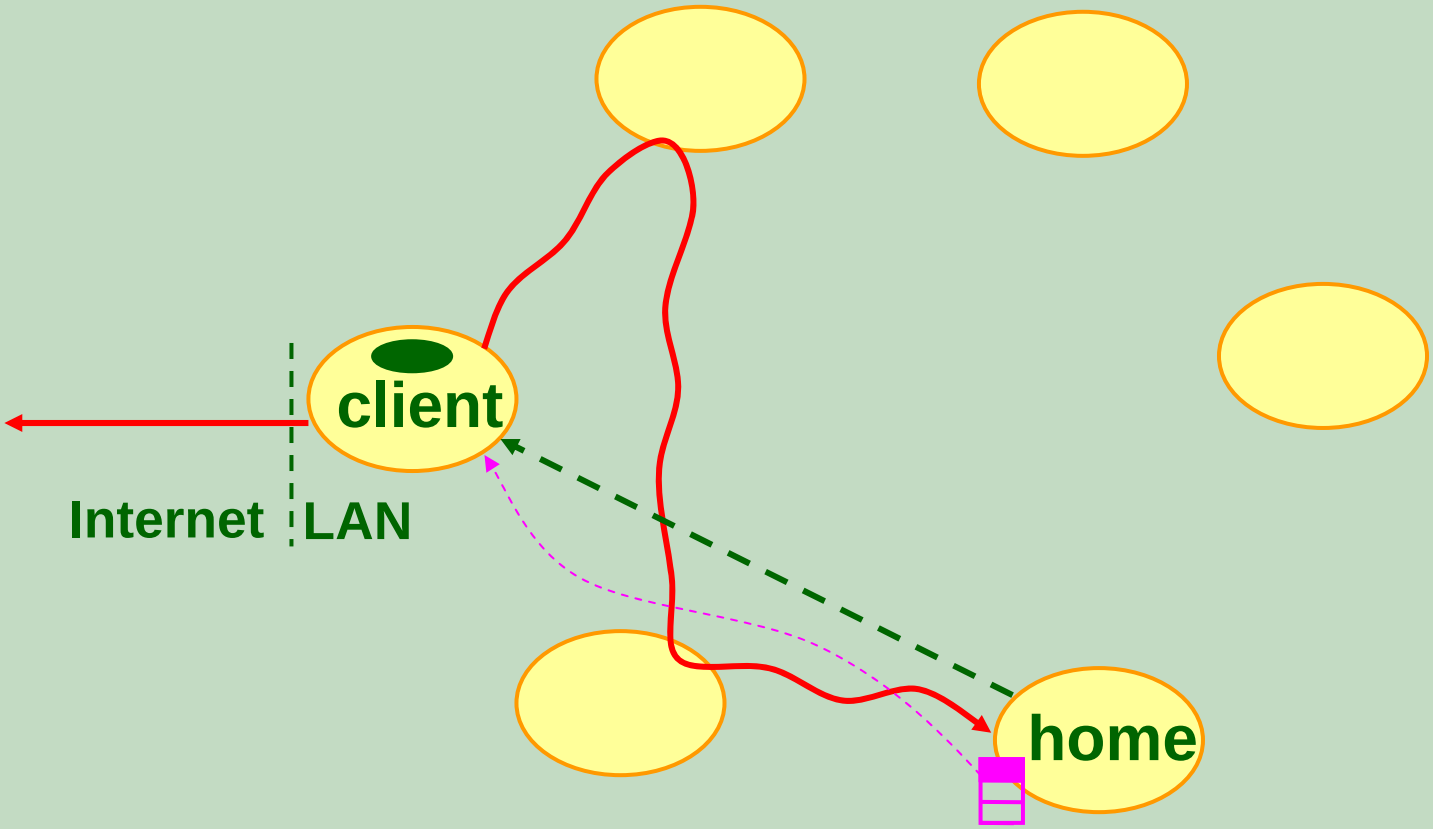
Directory model

Client nodes always cache objects locally.

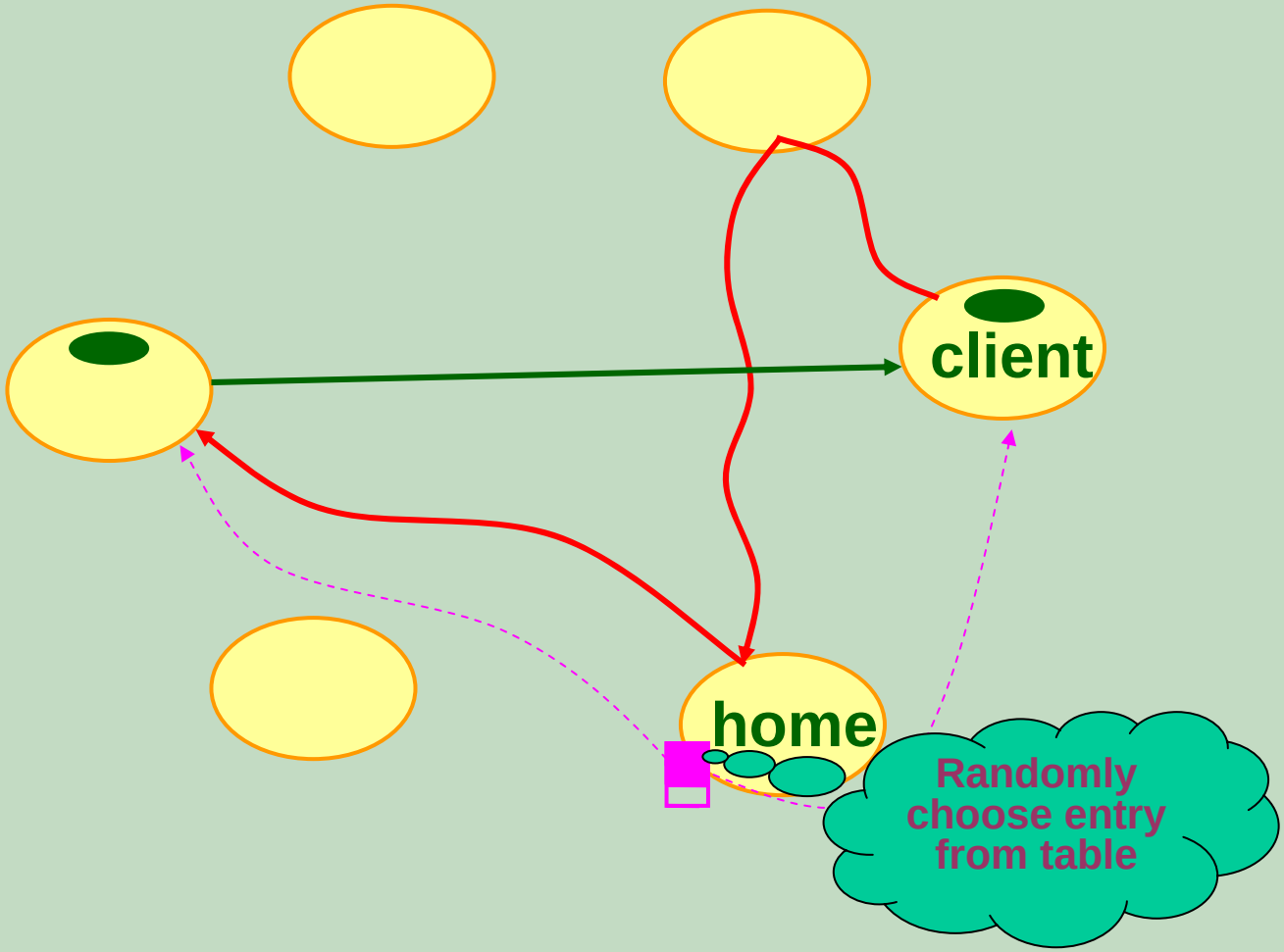
Home-store: home node also stores objects.

Directory: the home node only stores pointers to recent clients, and forwards requests.

Directory model



Directory model



Directory: Advantages

Avoids storing unnecessary copies of objects.

Rapidly changing directory for popular objects seems to improve load balancing.

Home-store scheme can incur hotspots.

Directory: Disadvantages

Cache insertion only happens at clients, so:

- **active clients store all the popular objects,**
- **inactive clients waste most of their storage.**

Implications:

1. **Reduced cache size.**
2. **Load imbalance.**

Directory: Load spike example

- Web page with many embedded images, or
- Periods of heavy browsing.

Many home nodes point to such clients!

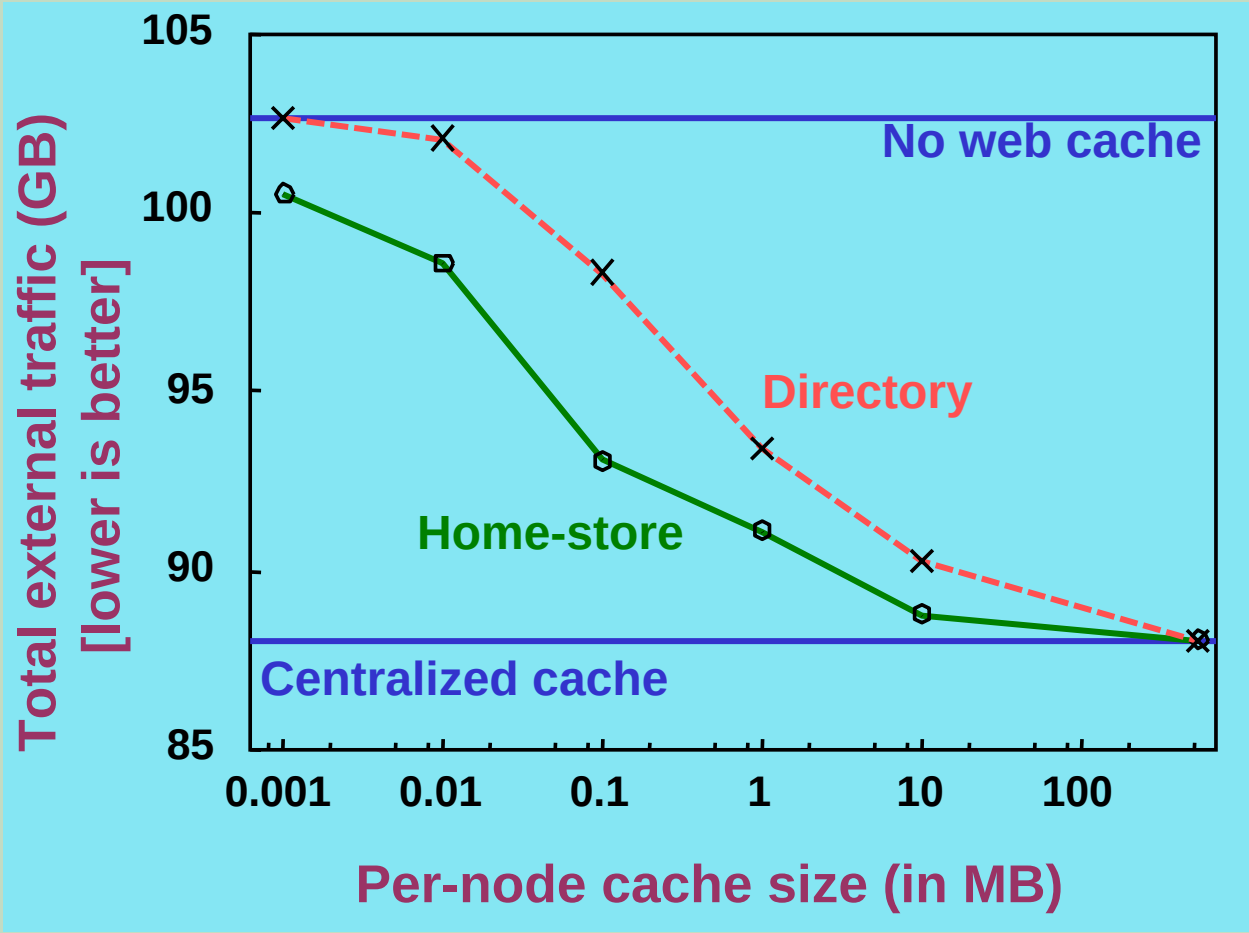
Evaluate …

Trace characteristics

Microsoft in :	Redmond	Cambridge
Total duration	1 day	31 days
Number of clients	36,782	105
Number of HTTP requests	16.41 million	0.971 million
Peak request rate	606 req/sec	186 req/sec
Number of objects	5.13 million	0.469 million
Number of cacheable objects	2.56 million	0.226 million
Mean cacheable object reuse	5.4 times	3.22 times

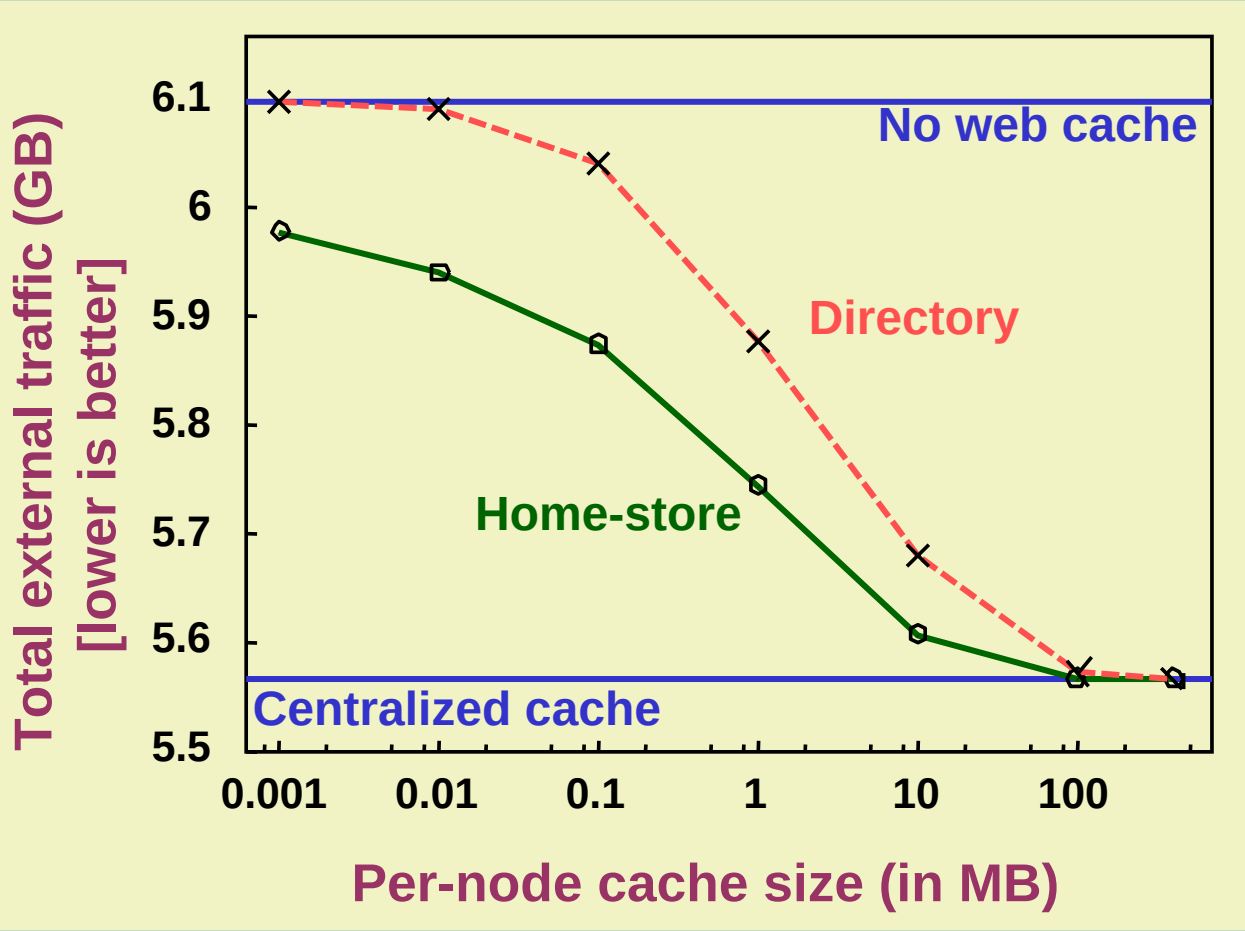
Redmond

Total external traffic



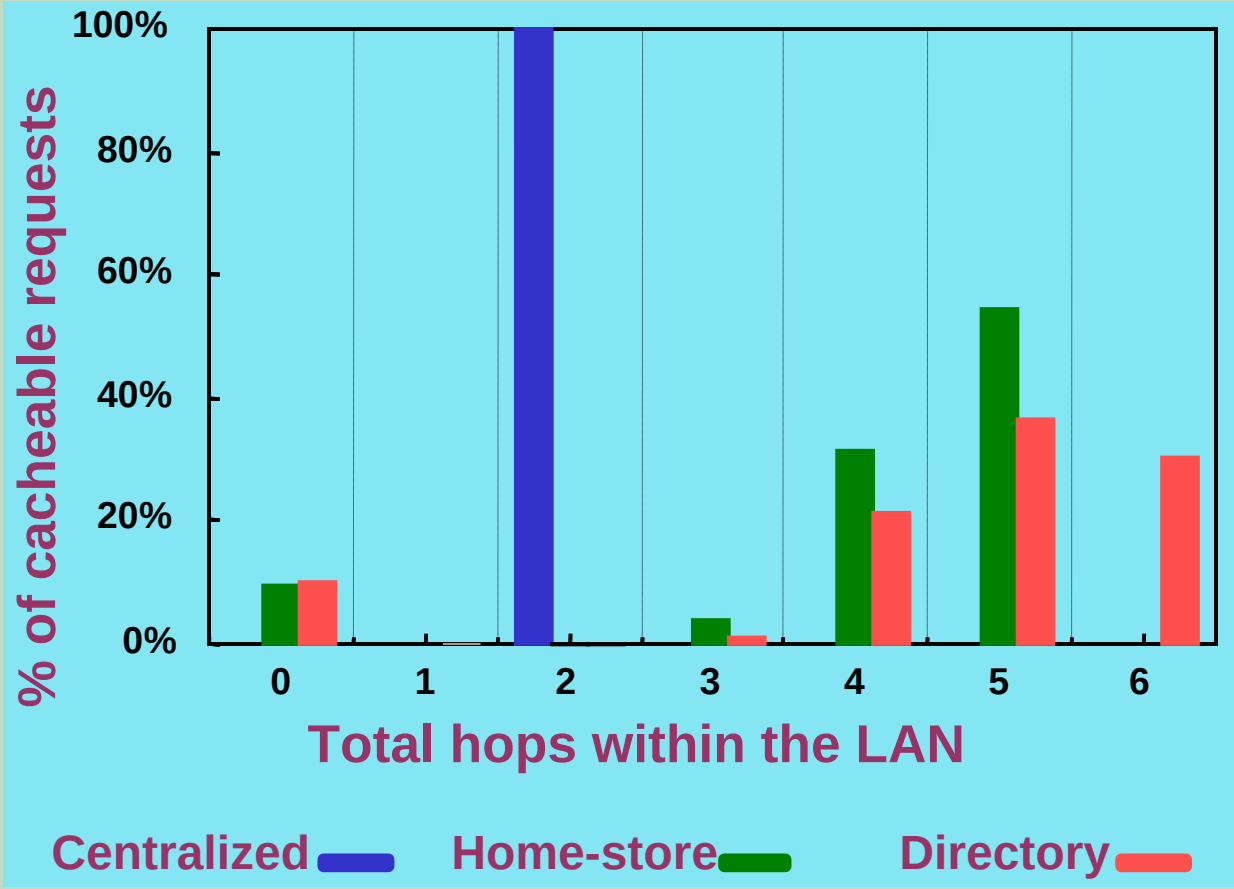
Cambridge

Total external traffic



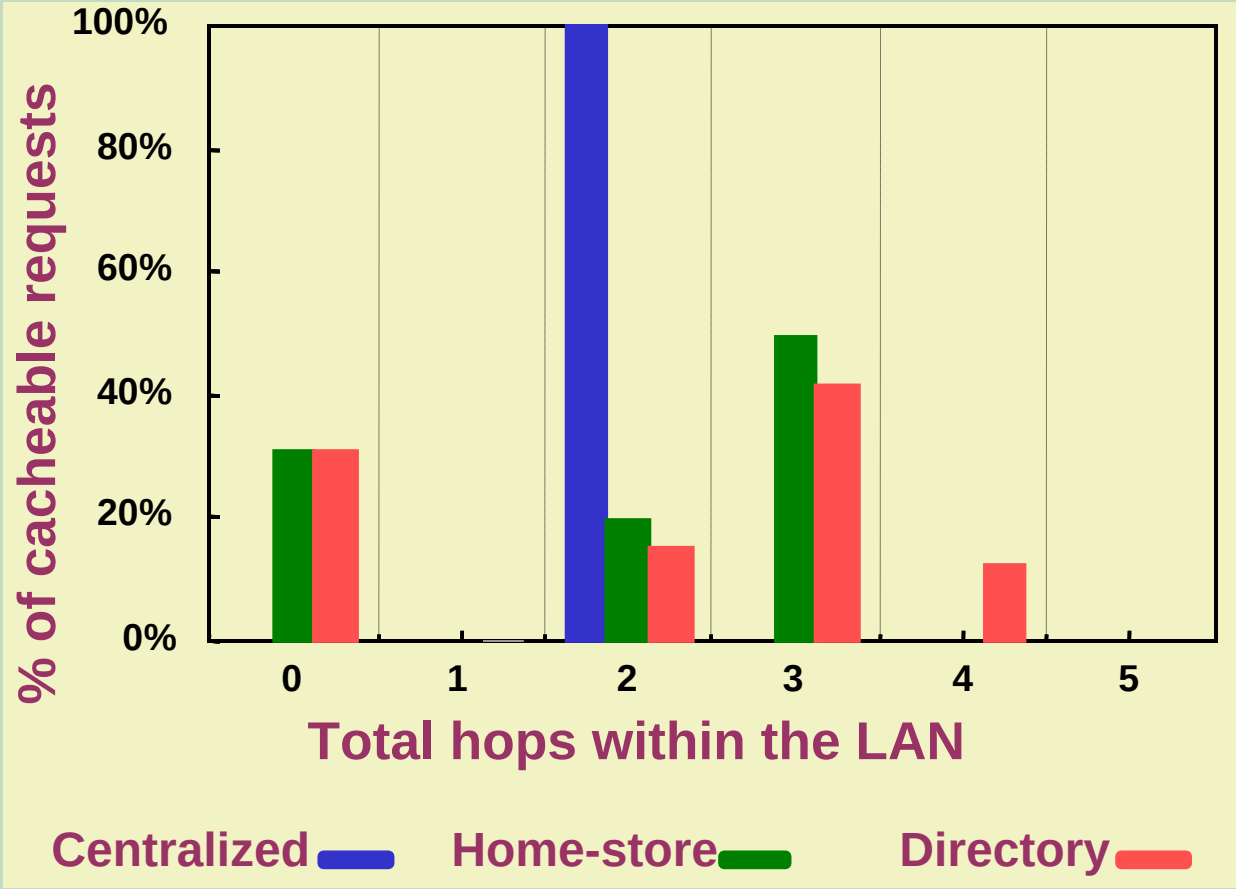
Redmond

LAN Hops



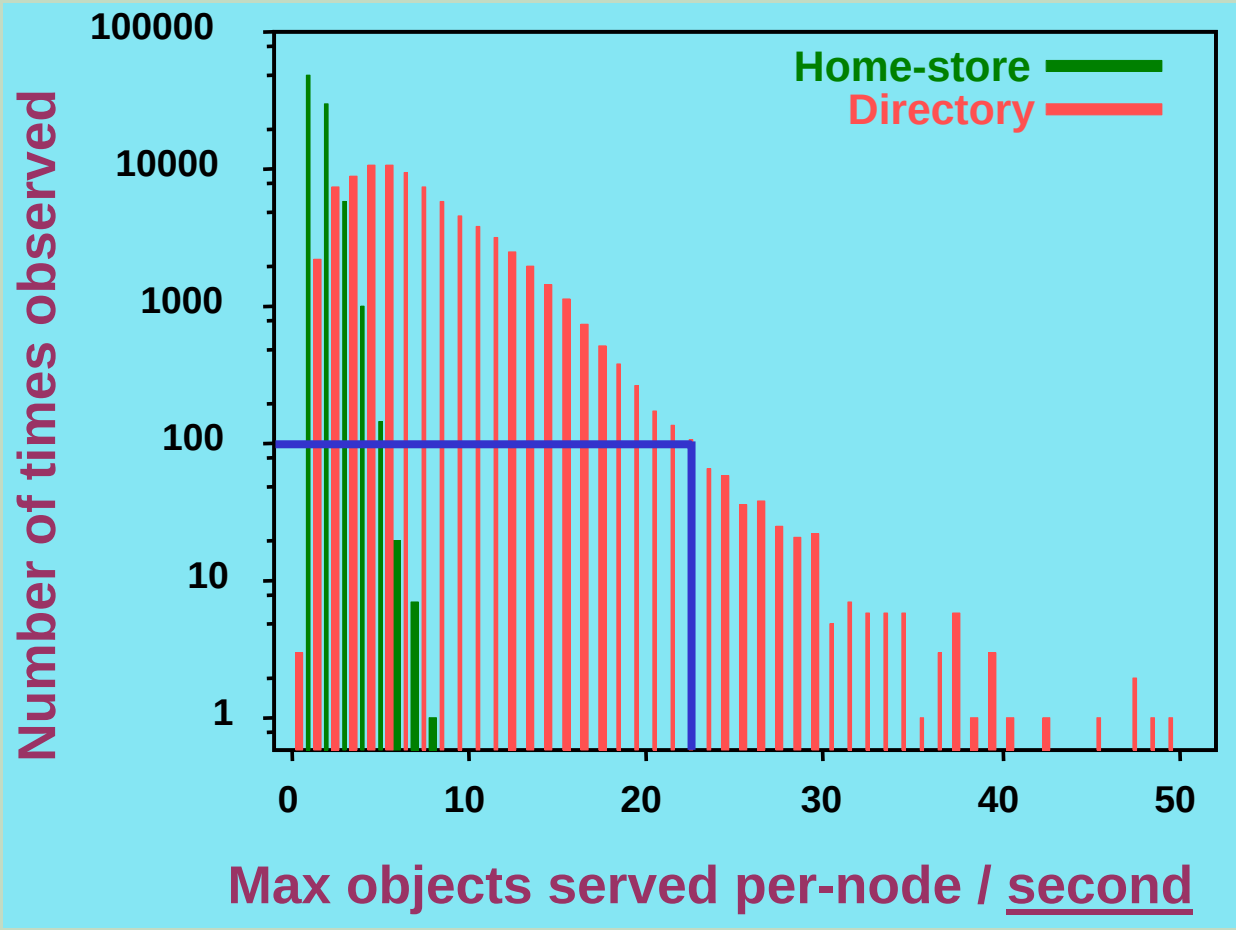
Cambridge

LAN Hops



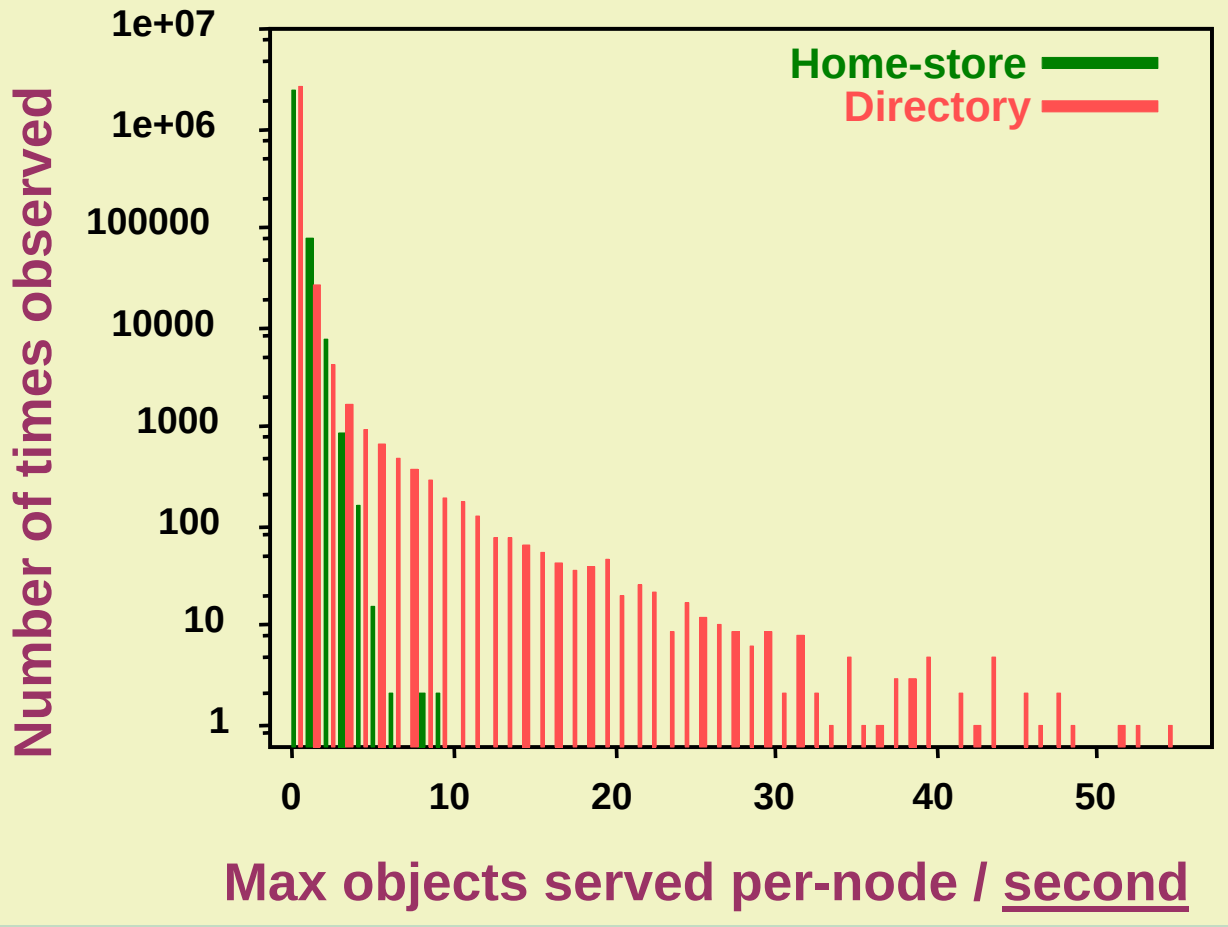
Redmond

Load in requests per sec



Cambridge

Load in requests per sec

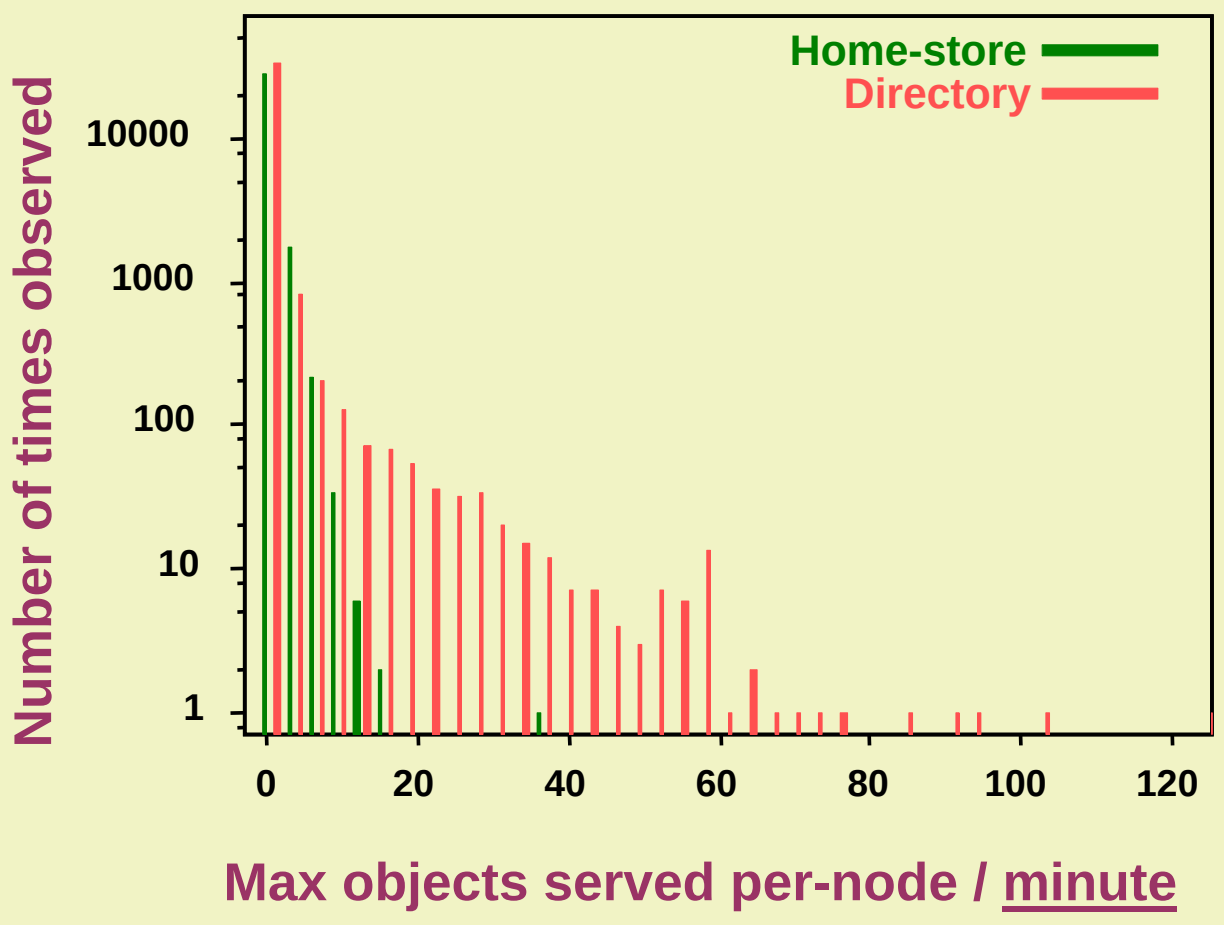


Redmond

Load in requests per min

Cambridge

Load in requests per min



Fault tolerance

Sudden node failures result in

partial loss of cached content.

Home-store: **Proportional to failed nodes.**

Directory: **More vulnerable.**

Fault tolerance

If 1% of Squirrel nodes abruptly crash, the fraction of lost cached content is:

	Home-store	Directory
Redmond	Mean 1% Max 1.77%	Mean 1.71% Max 19.3%
Cambridge	Mean 1% Max 3.52%	Mean 1.65% Max 9.8%

Conclusions

- Possible to decentralize web caching.
- Performance comparable to a centralized web cache,
- Is better in terms of cost, scalability, and administration effort, and
- Under our assumptions, the home-store scheme is superior to the directory scheme.

Other aspects of Squirrel

- **Adaptive replication**
  - **Hotspot avoidance**
  - **Improved robustness**
- **Route caching**
  - **Fewer LAN hops**

Thanks.