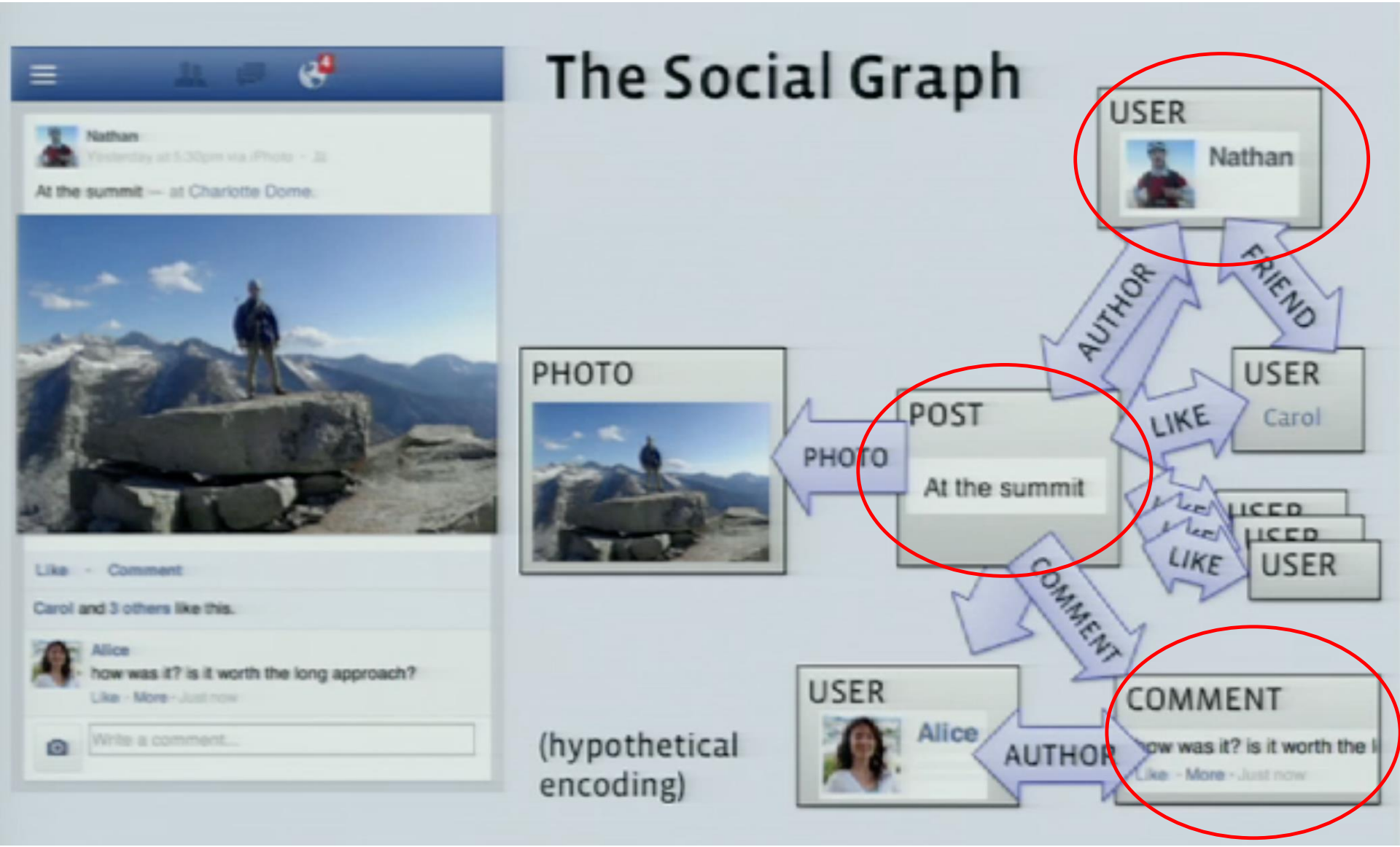
TAO: Facebook's Distributed Data Store for the Social Graph

Authors: Nathan Bronson, *et. al*

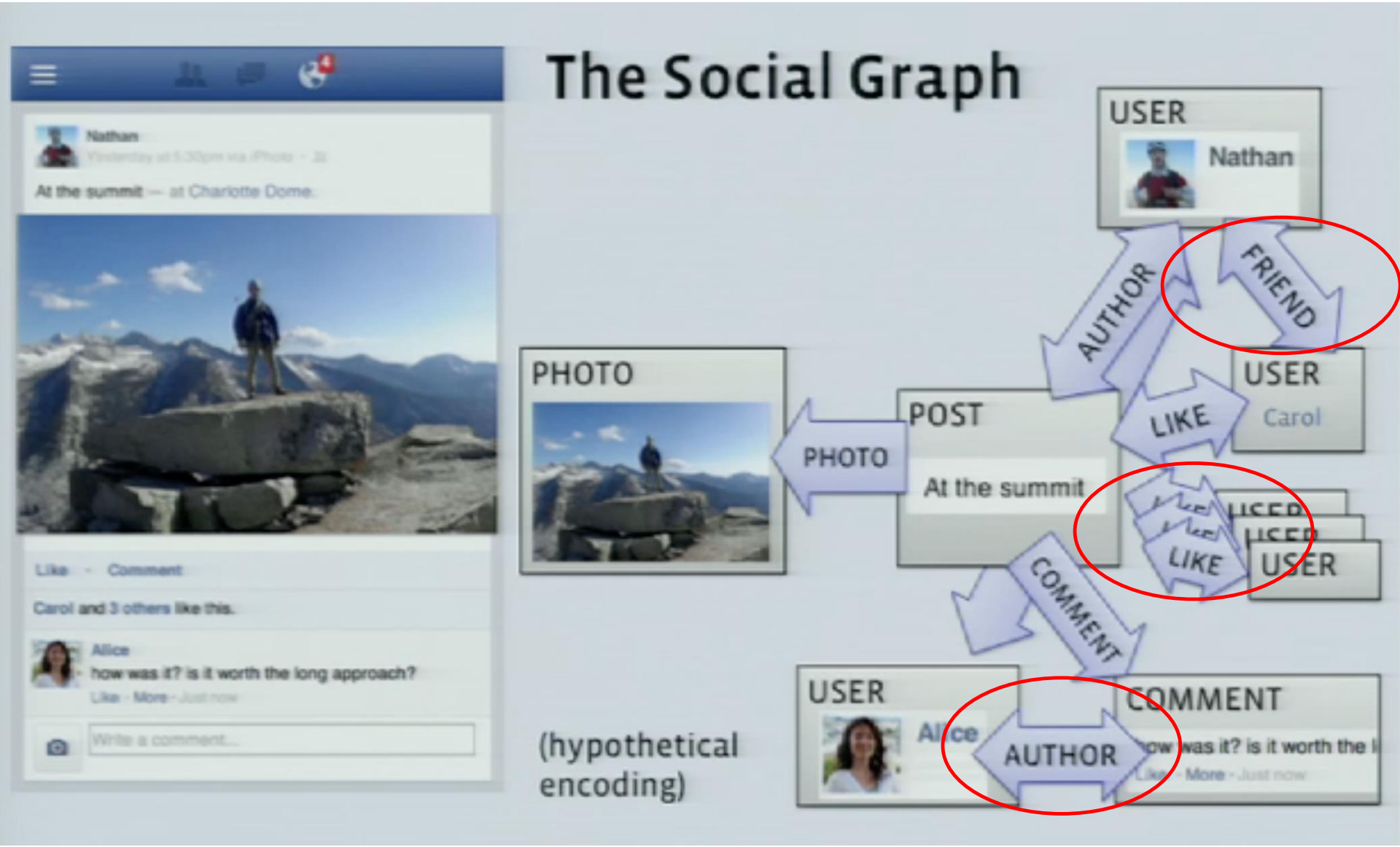
Presentors: Zhichao Cao, Fenggang Wu

10/01/2018

Center for Research in
Intelligent Storage



Source: https://www.usenix.org/node/174510



Center for Research in Intelligent Storage

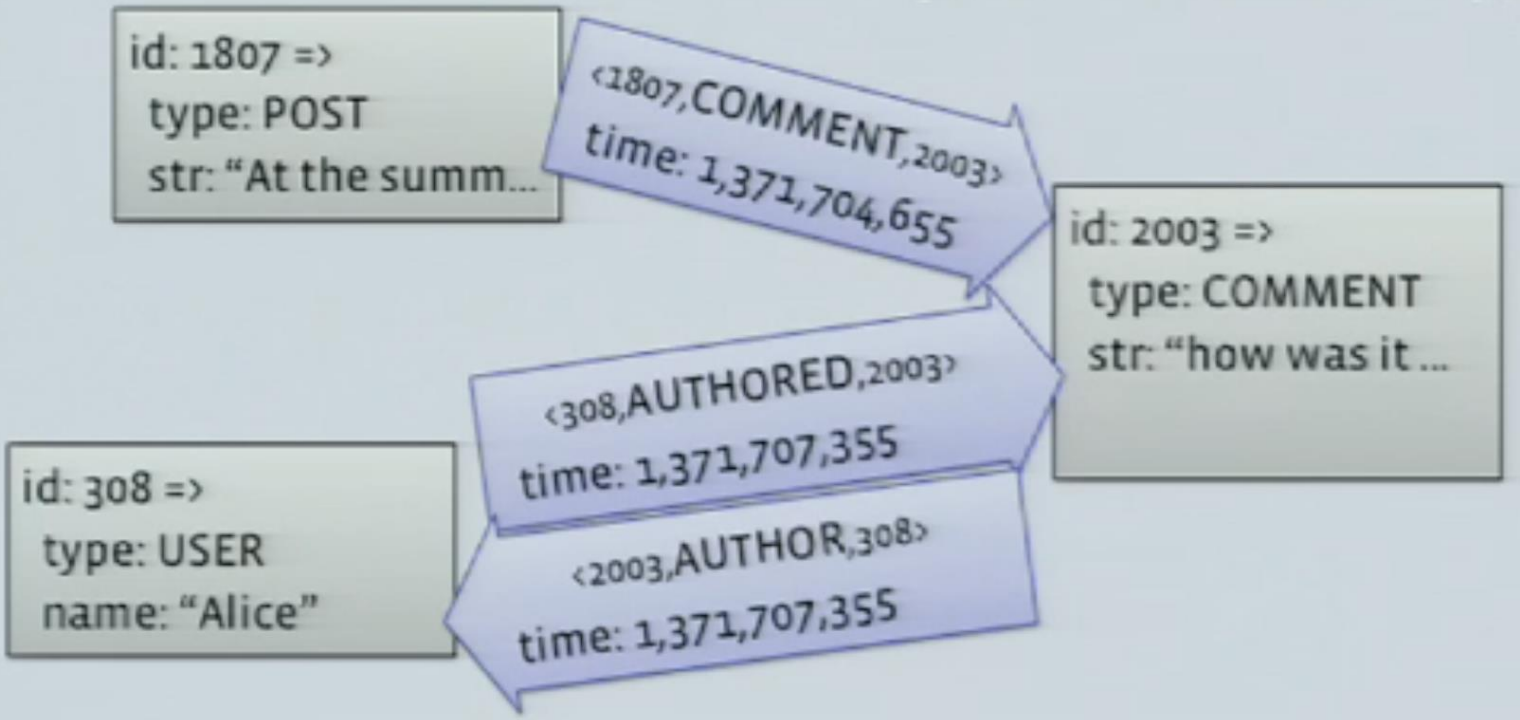
Source: https://www.usenix.org/node/174510

Objects = Nodes

- Identified by unique 64-bit IDs
- Typed, with a schema for fields

Associations = Edges

- Identified by ‹id1, type, id2›
- Bidirectional associations are two edges, same or different type



Source: https://www.usenix.org/node/174510

Objects and Associations API

Reads – 99.8%

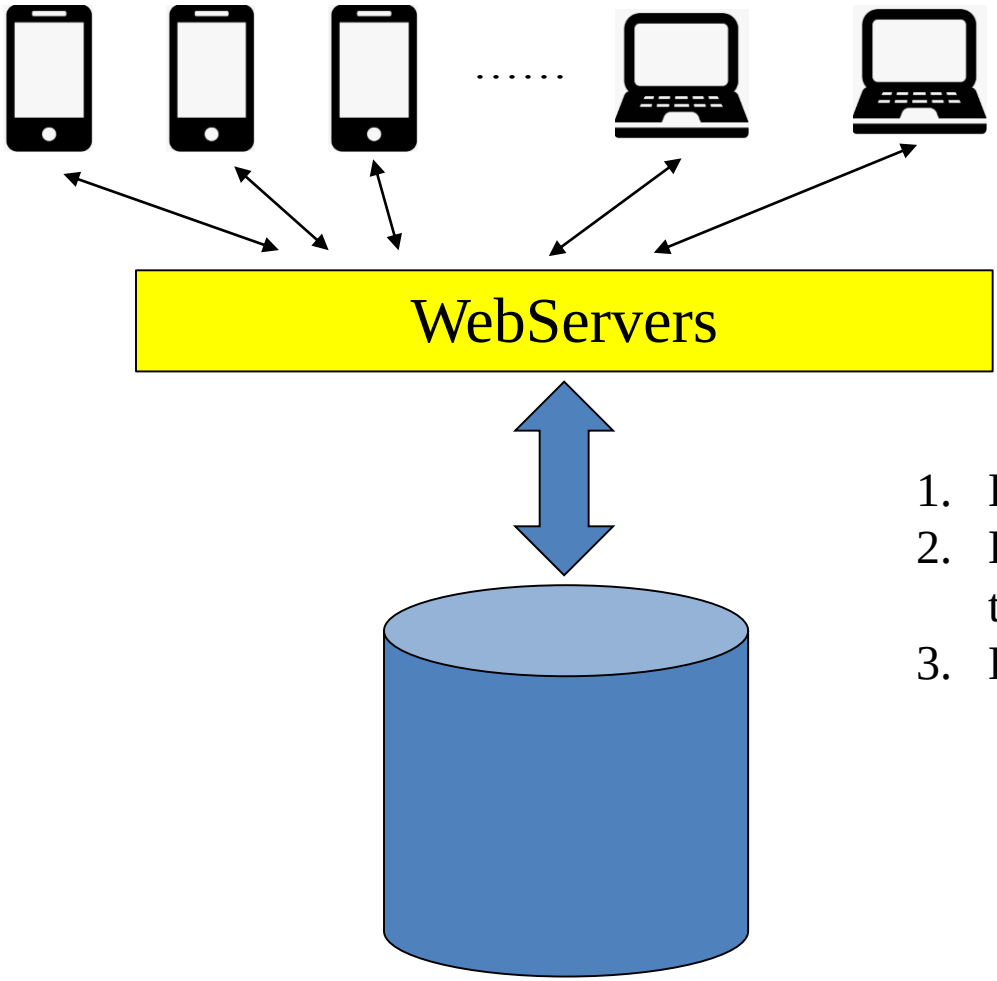
- Point queries
  - `obj_get` 28.9%
  - `assoc_get` 15.7%
- Range queries
  - `assoc_range` 40.9%
  - `assoc_time_range` 2.8%
- Count queries
  - `assoc_count` 11.7%

Writes – 0.2%

- Create, update, delete for objects
  - `obj_add` 16.5%
  - `obj_update` 20.7%
  - `obj_del` 2.0%
- Set and delete for associations
  - `assoc_add` 52.5%
  - `assoc_del` 8.3%

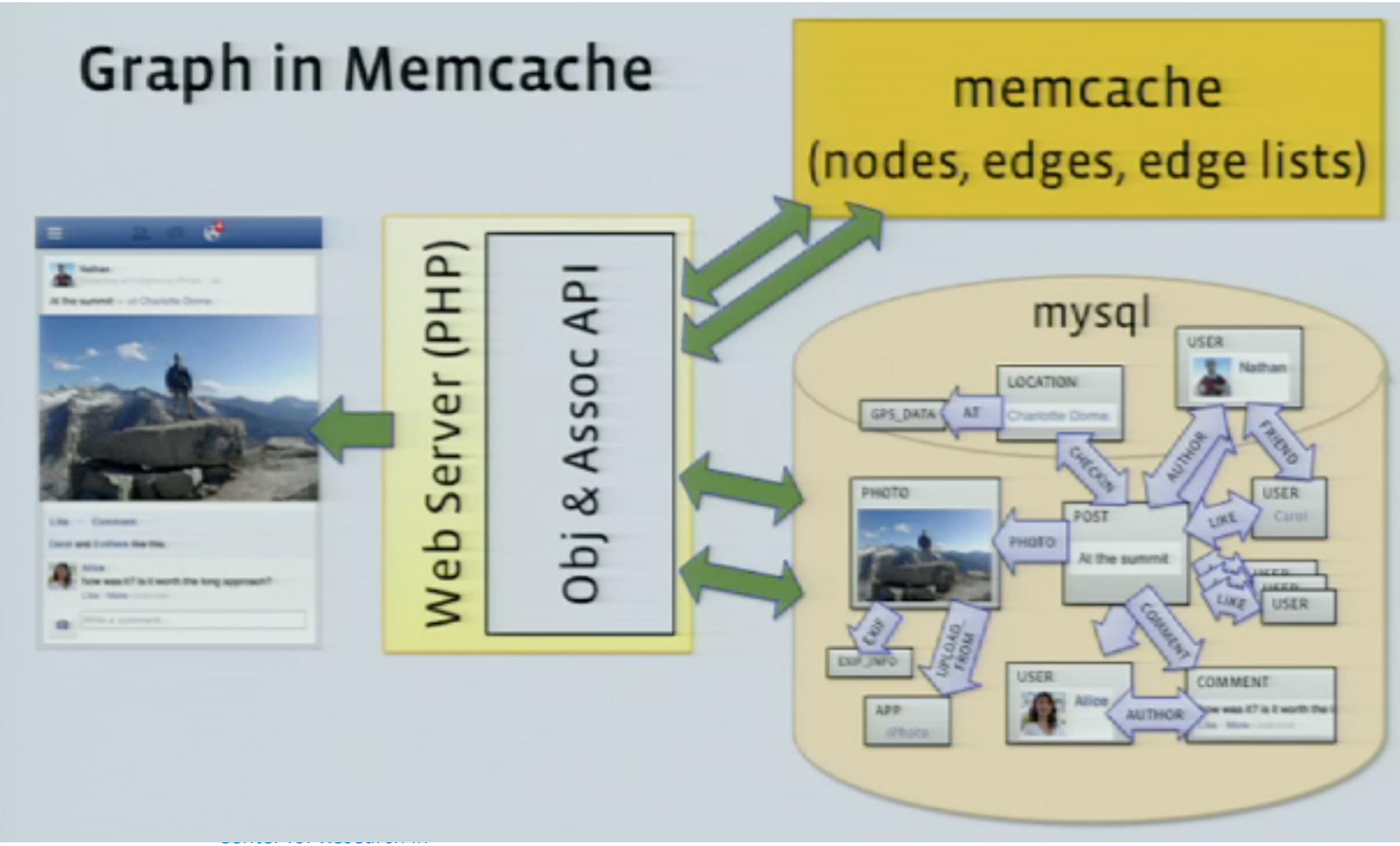
Source: https://www.usenix.org/node/174510

Stage 1: Relational Database Supported



1. Low write latency
2. High read latency, difficult to achieve load balancing
3. Low scalability

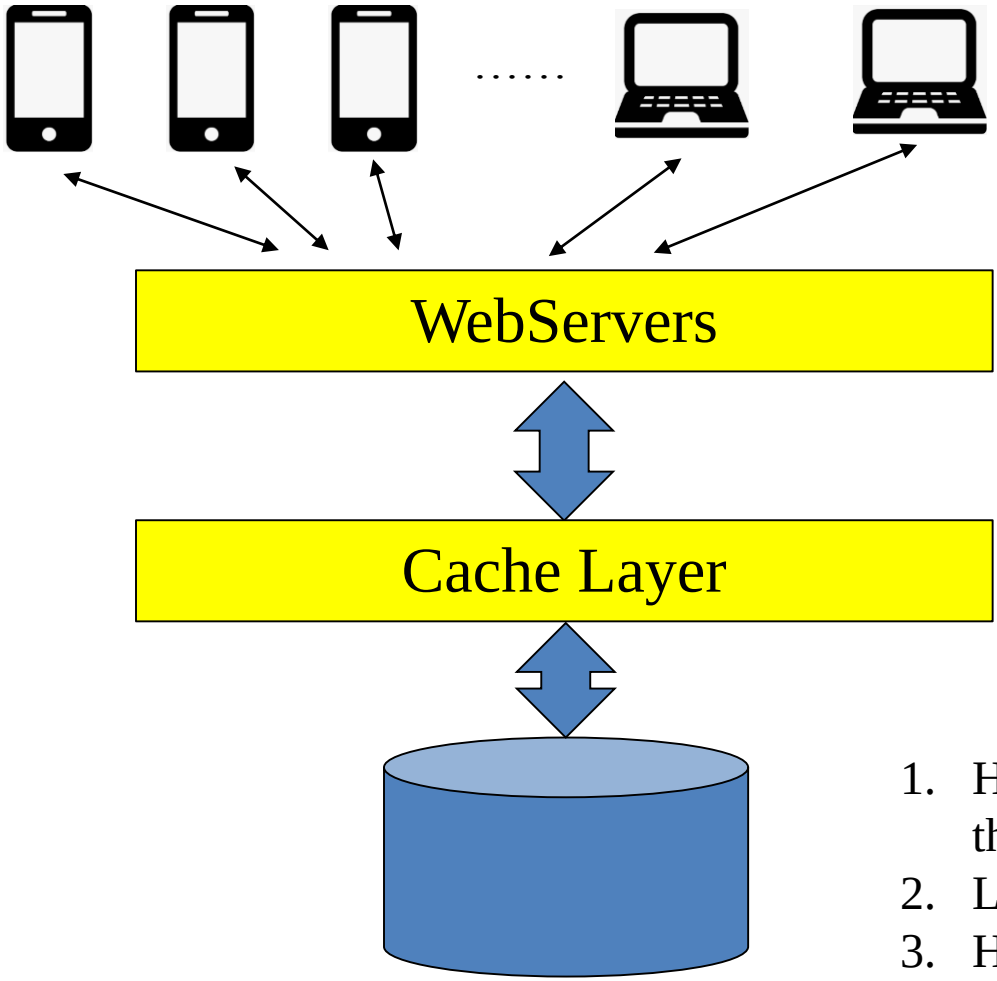
Stage 2: Cache Supported (memcache)



Intelligent Storage

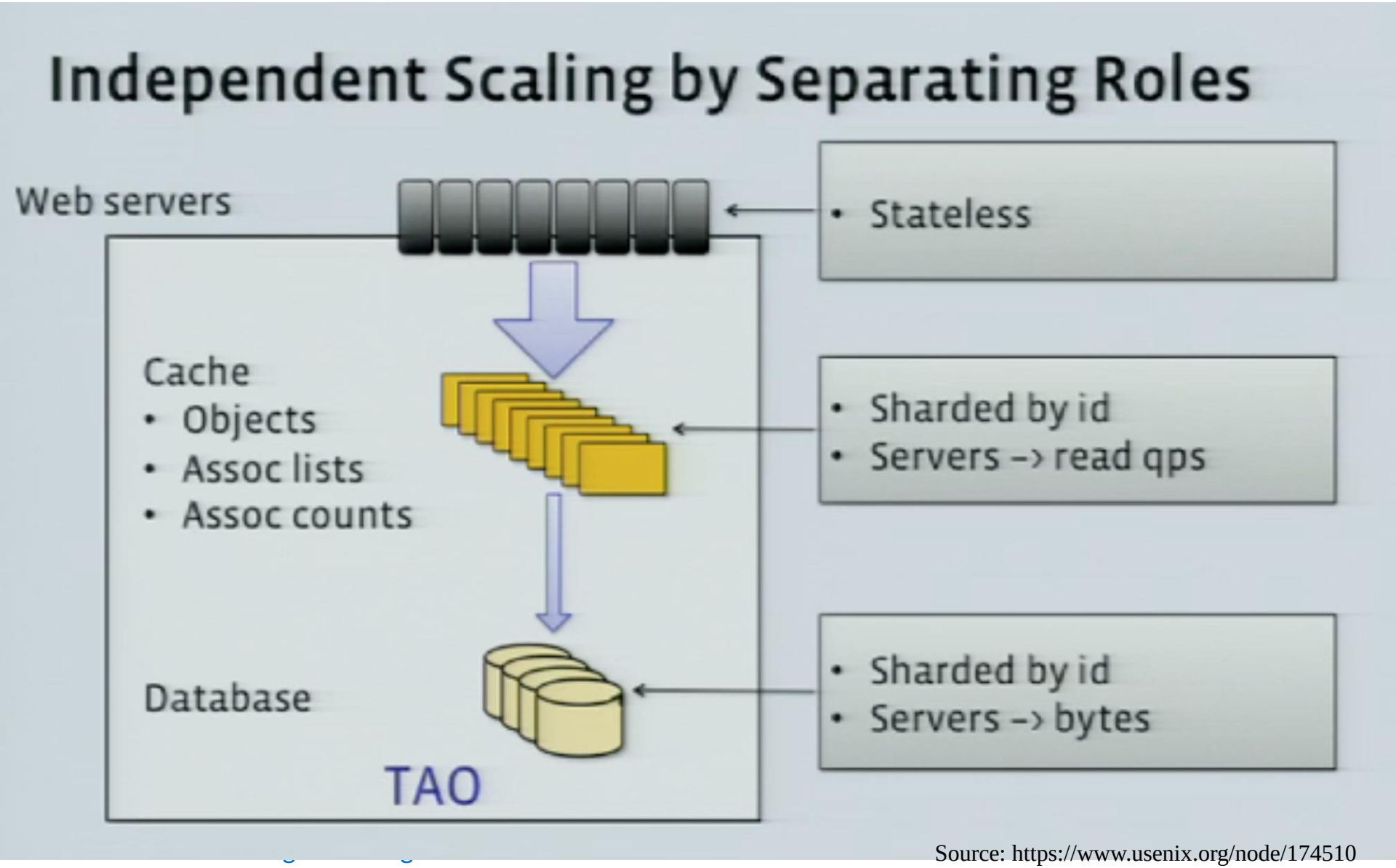
Source: https://www.usenix.org/node/174510

Stage 2: Cache Supported (cache in between)



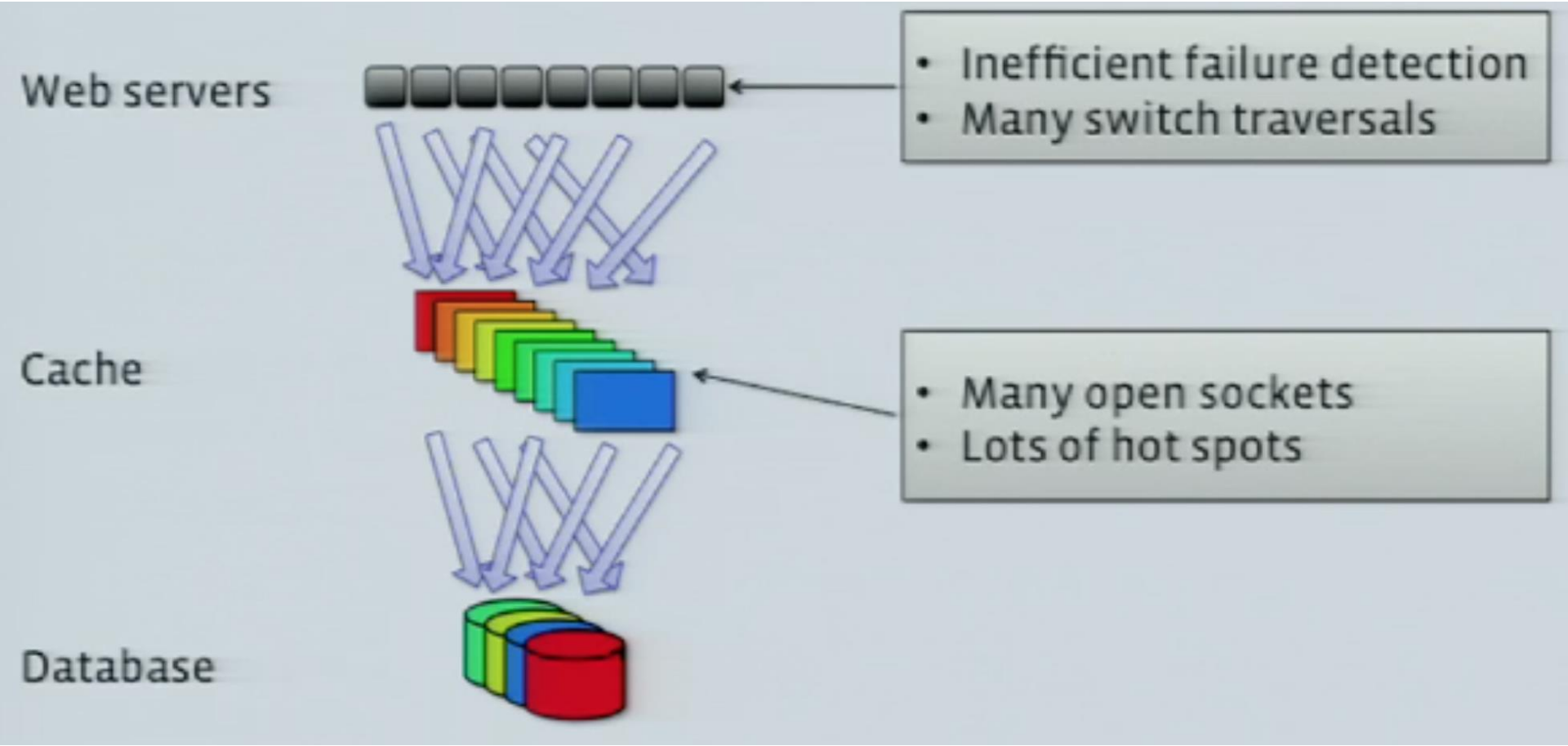
1. Higher write latency (clean the cache)
2. Lower read latency
3. Higher scalability

Stage 3: TAO (The Associations and Objects)



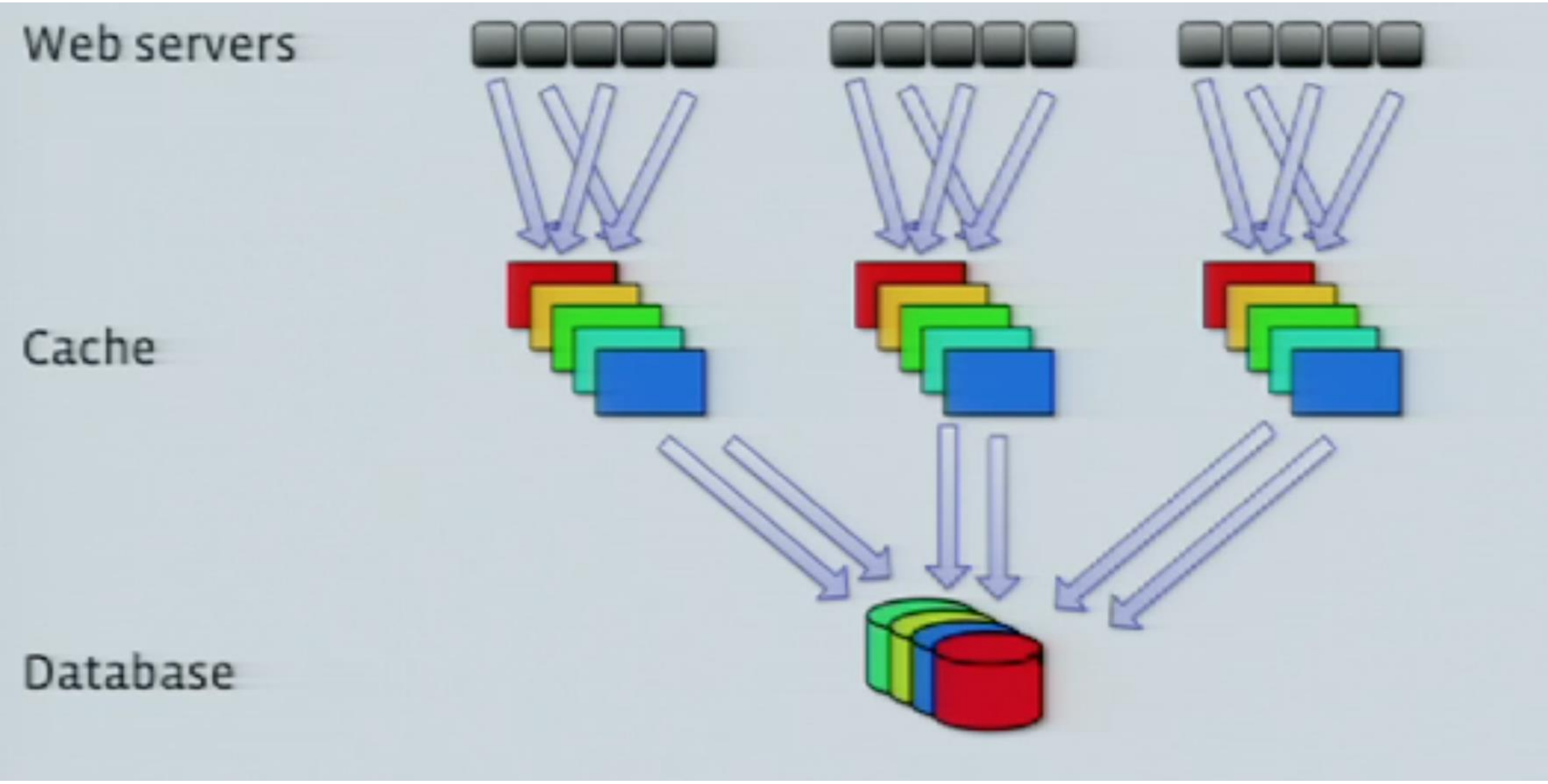
Source: https://www.usenix.org/node/174510

TAO Design 1: One Tier with Sharding

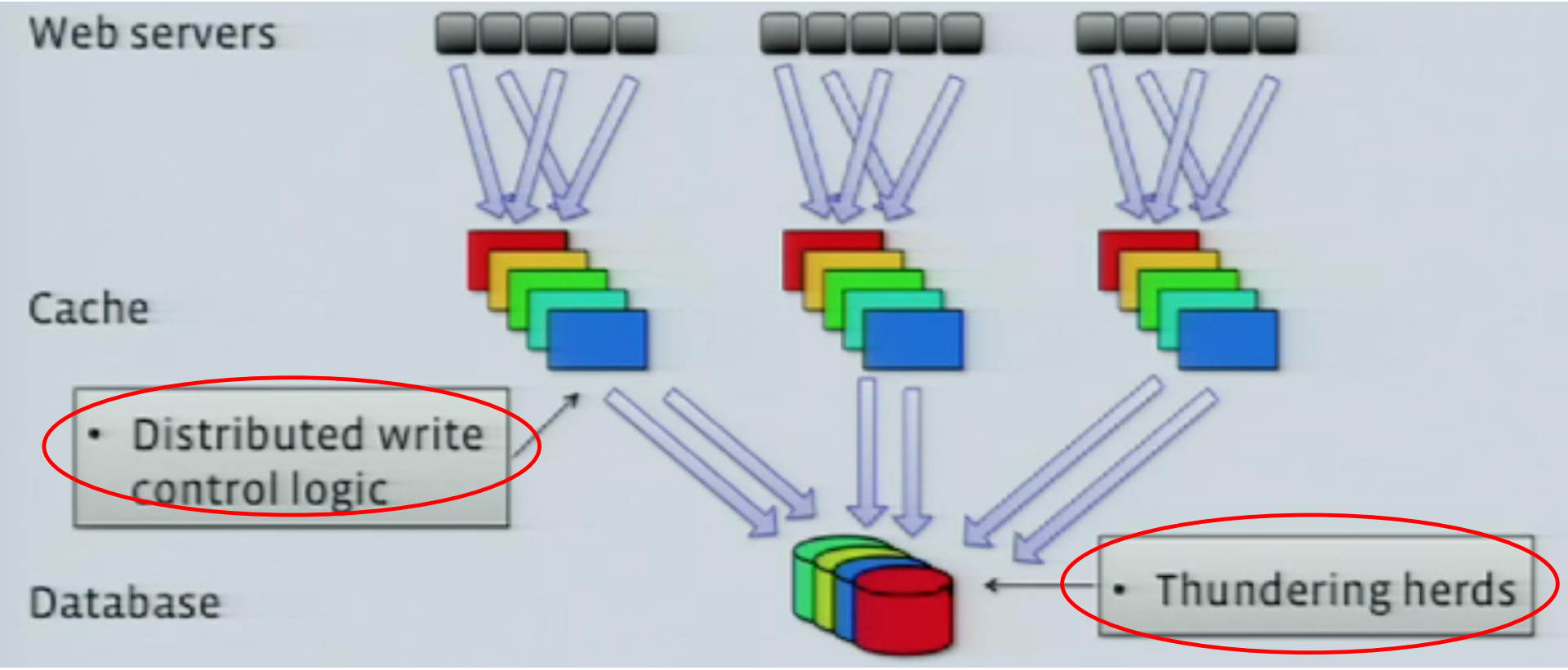


Source: https://www.usenix.org/node/174510

TAO Design 2: Multiple Caching Tiers



Source: https://www.usenix.org/node/174510

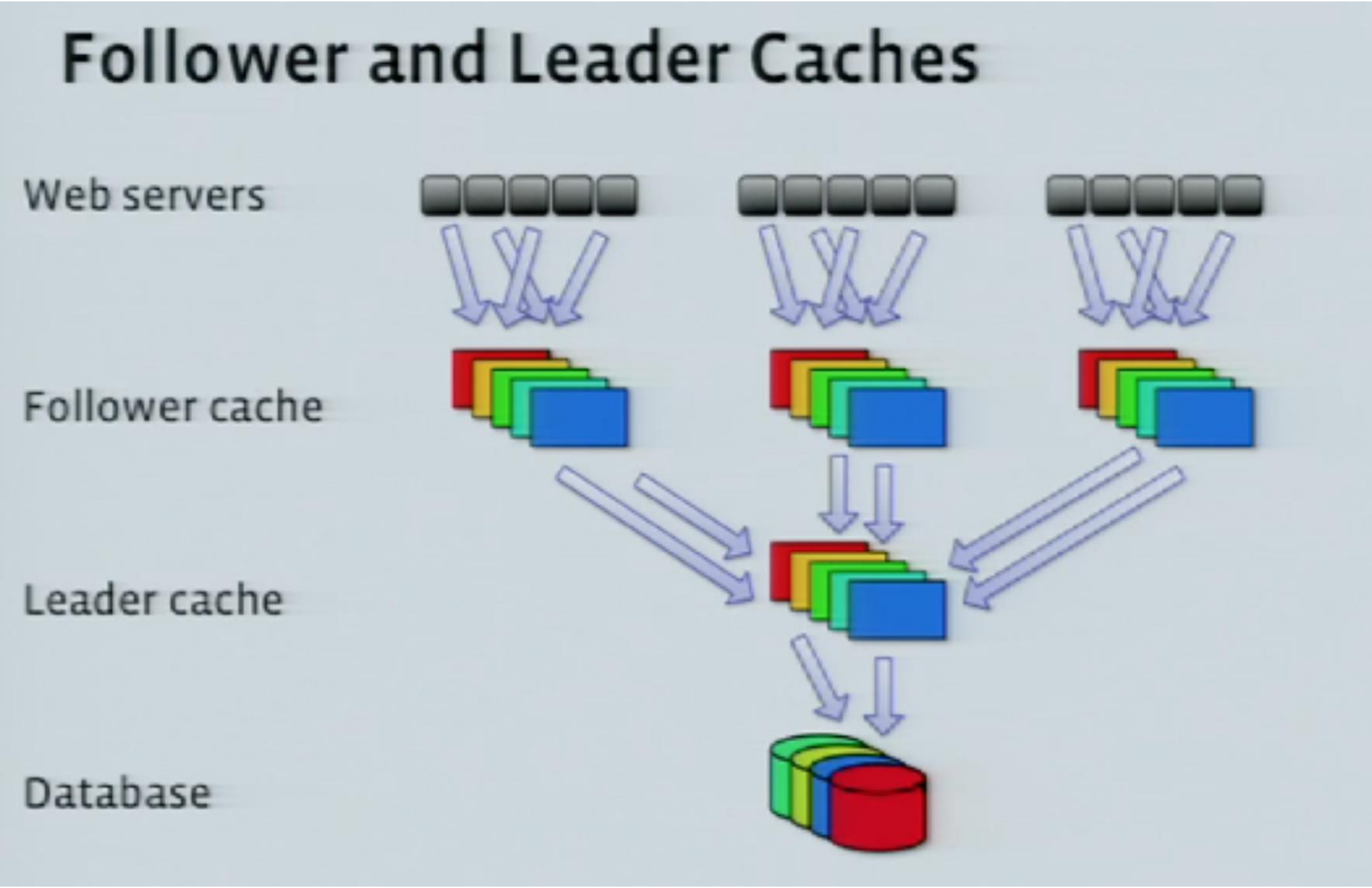


1. Complex distributed write logic to handle to ensure data correctness, consistency

2. A lot of concurrent read or write go to the MySQL server, for the same data

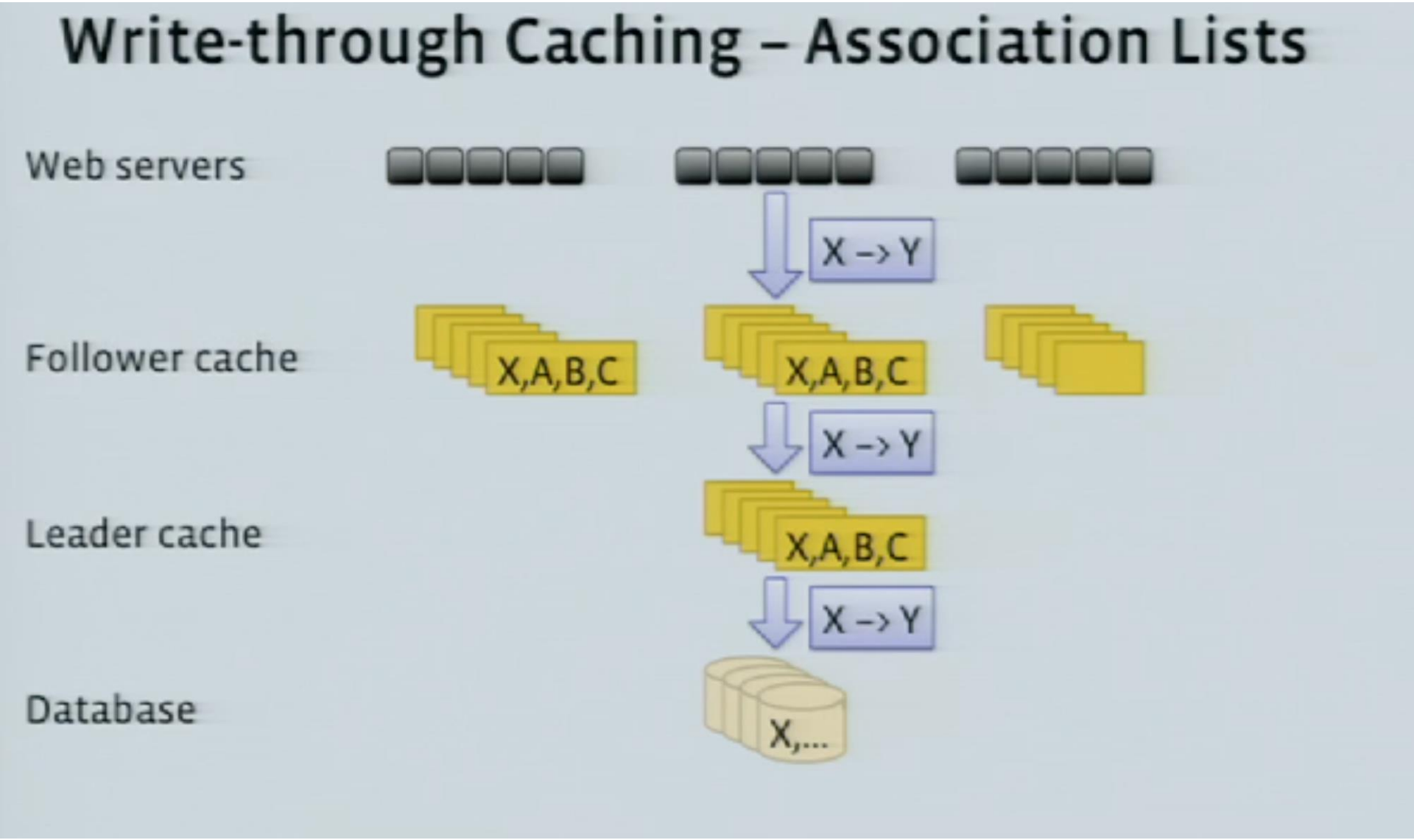
3. Make tradeoffs between limiting the maximum tier size and scaling the cache

Source: https://www.usenix.org/node/174510



Source: https://www.usenix.org/node/174510

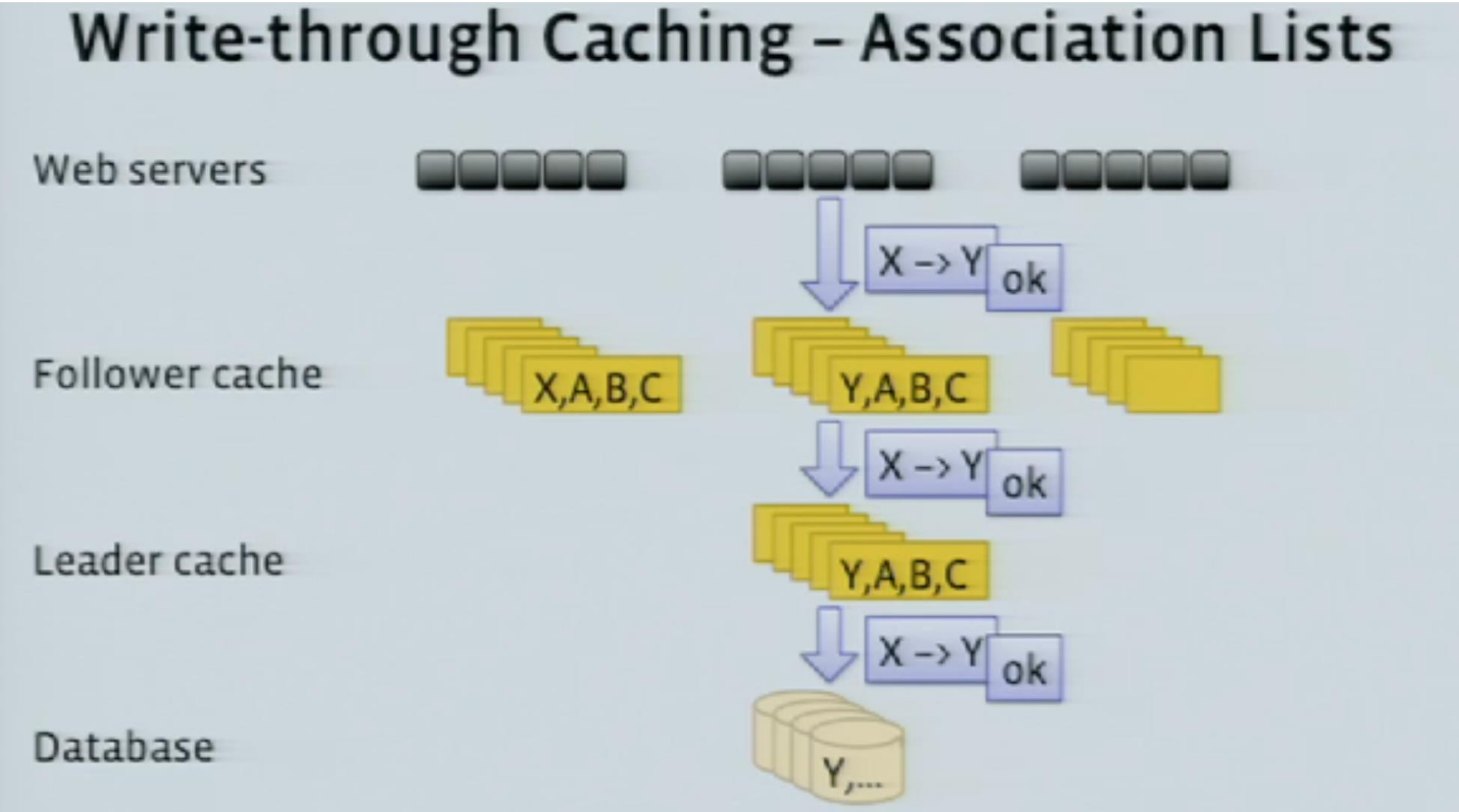
Write-through Caching – Association Lists



Center for Research in
Intelligent Storage

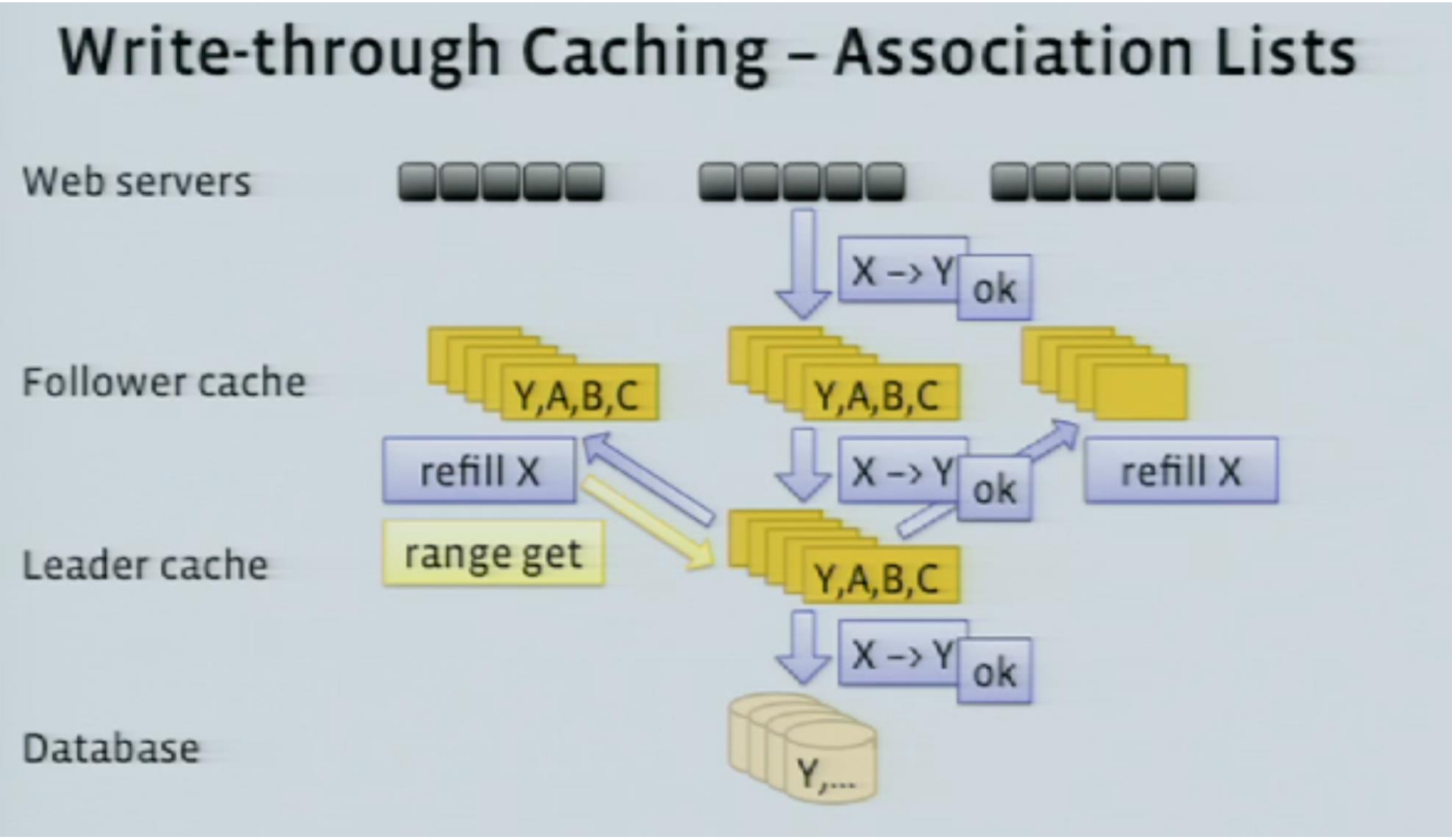
Source: https://www.usenix.org/node/174510

Write-through Caching – Association Lists



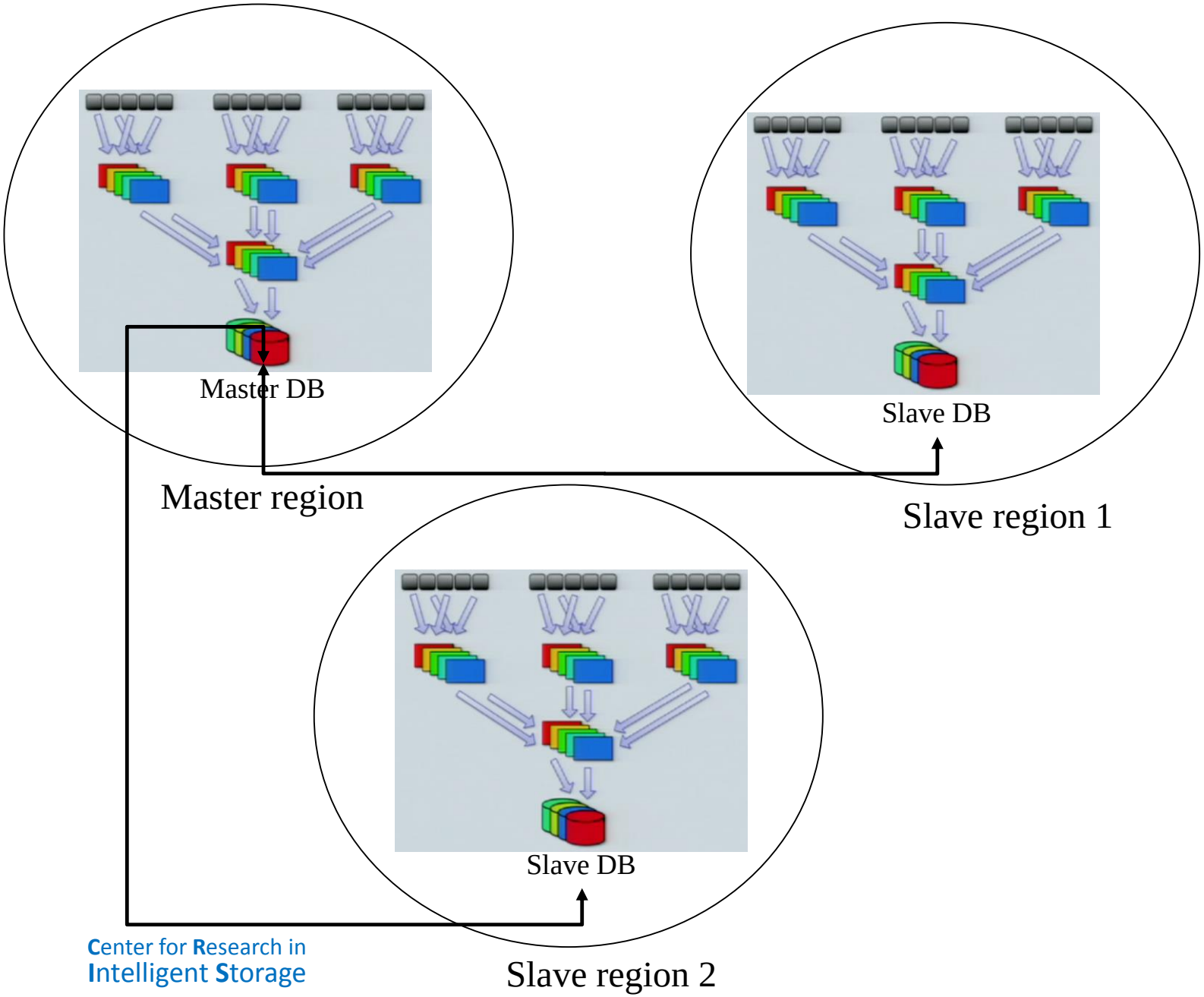
Source: https://www.usenix.org/node/174510

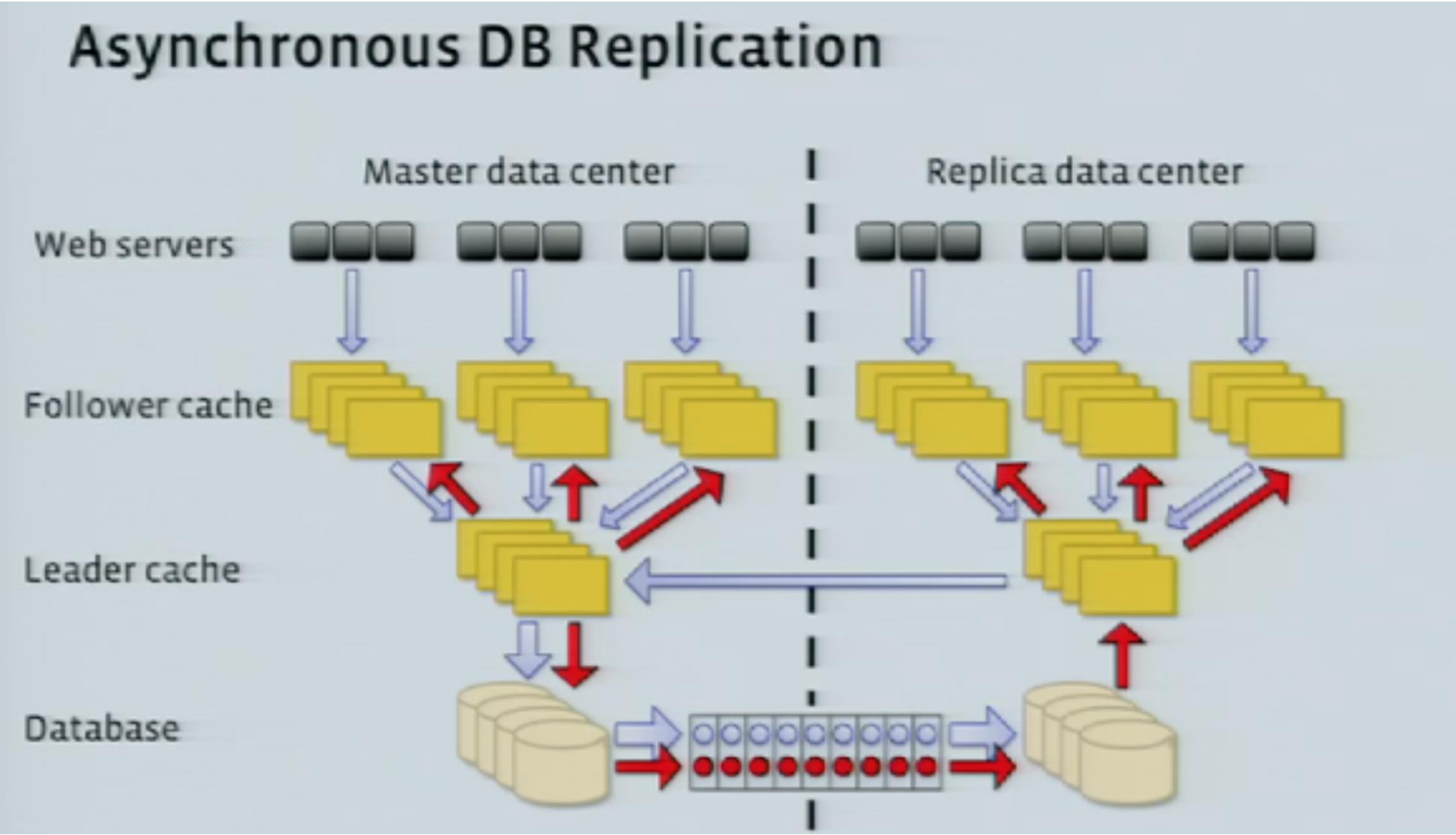
Write-through Caching – Association Lists



Source: https://www.usenix.org/node/174510

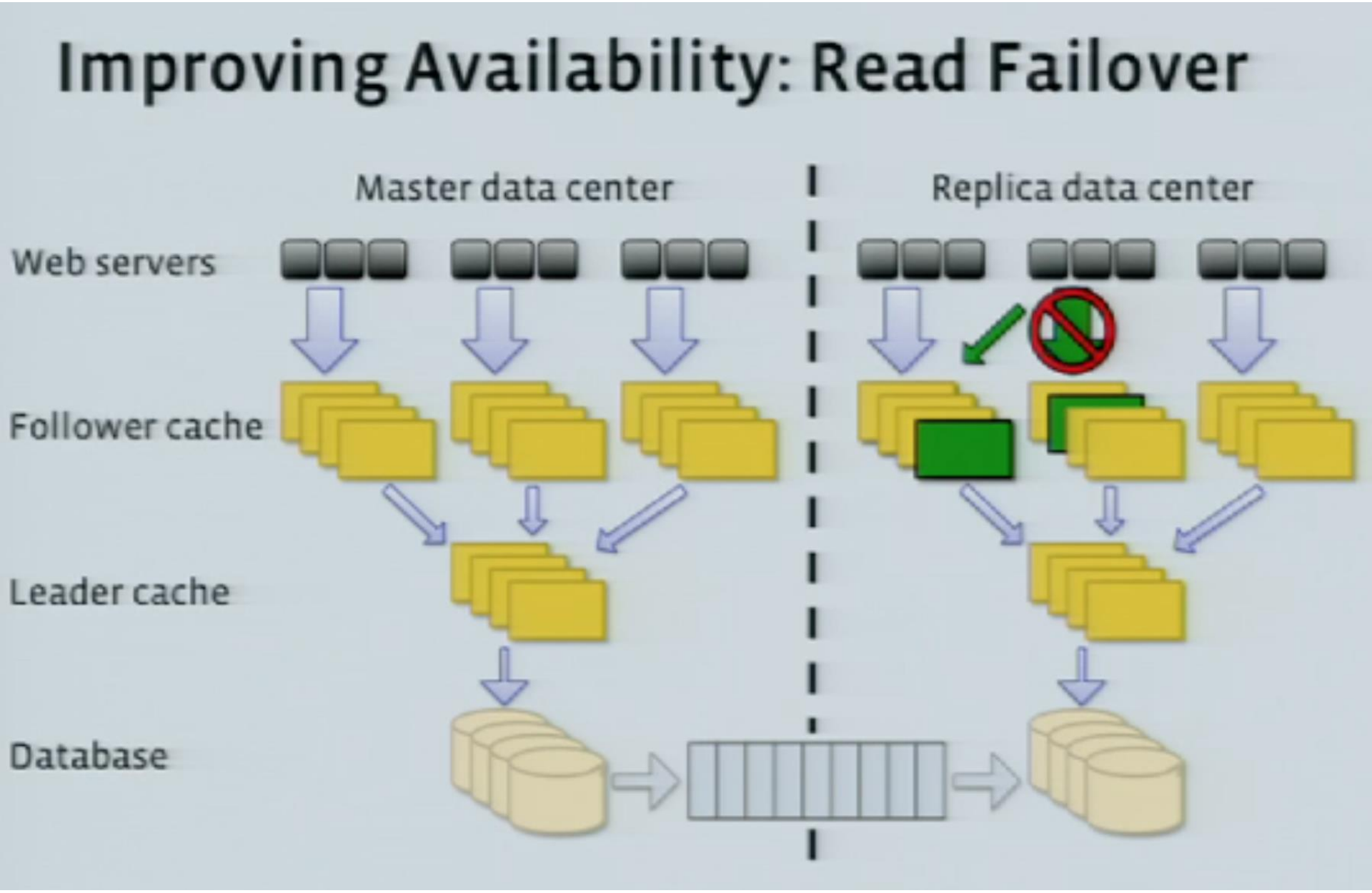
How to Scale out





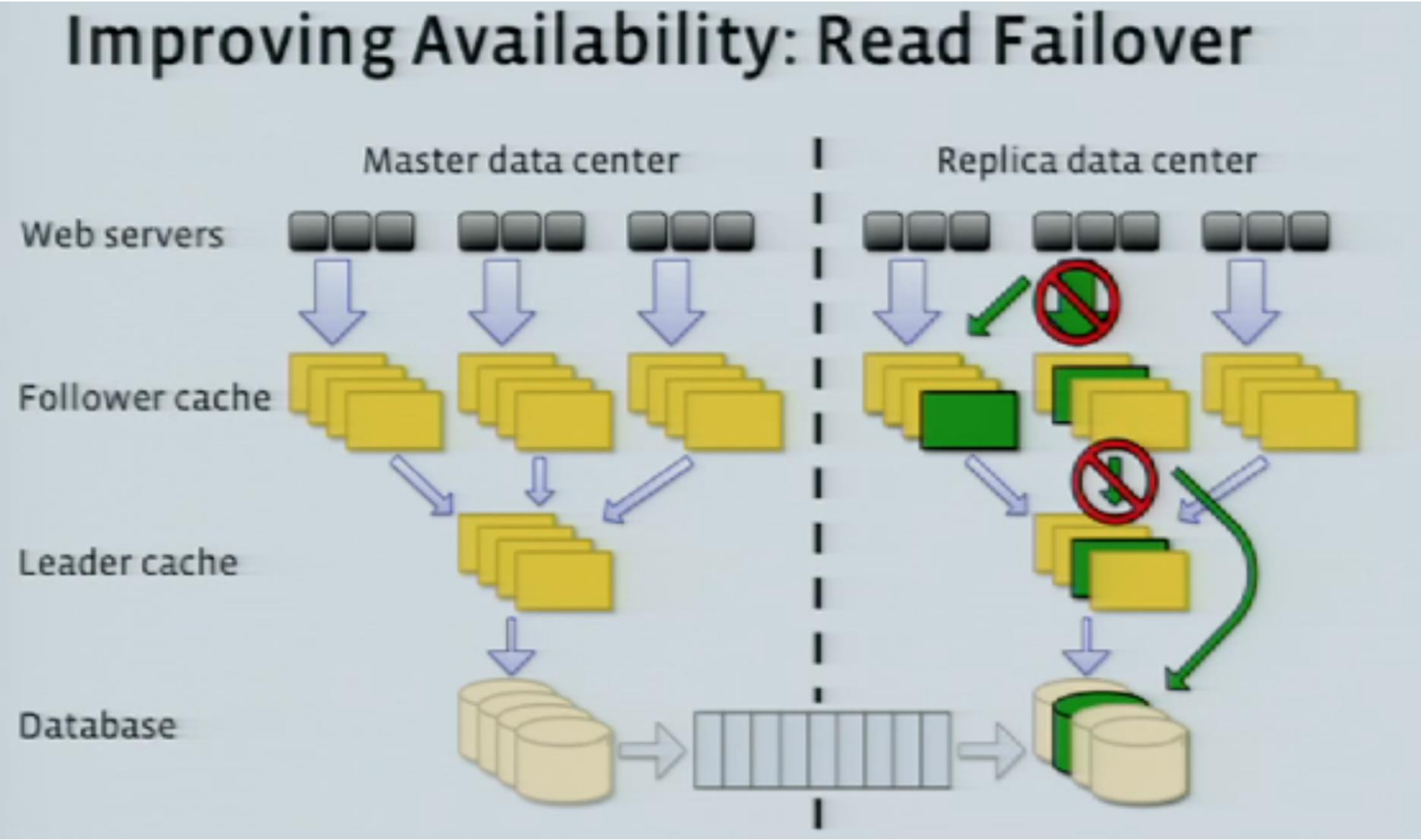
Source: https://www.usenix.org/node/174510

Improving Availability: Read Failover



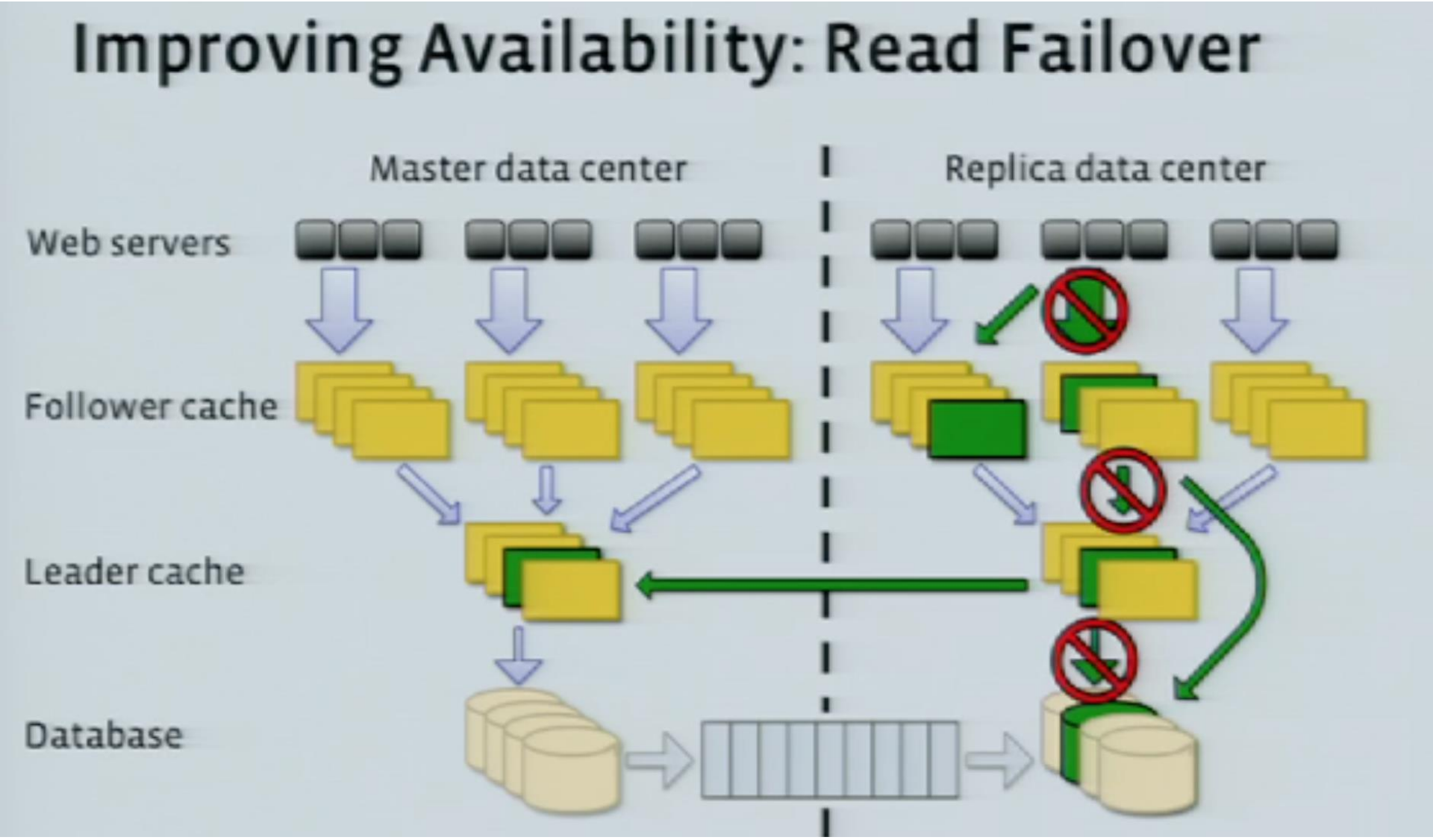
Source: https://www.usenix.org/node/174510

Improving Availability: Read Failover



Source: https://www.usenix.org/node/174510

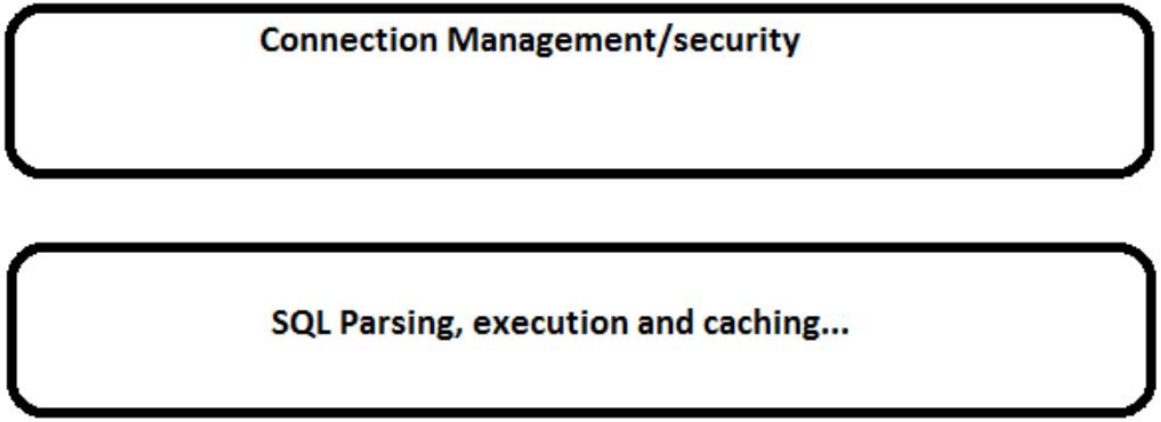
Improving Availability: Read Failover



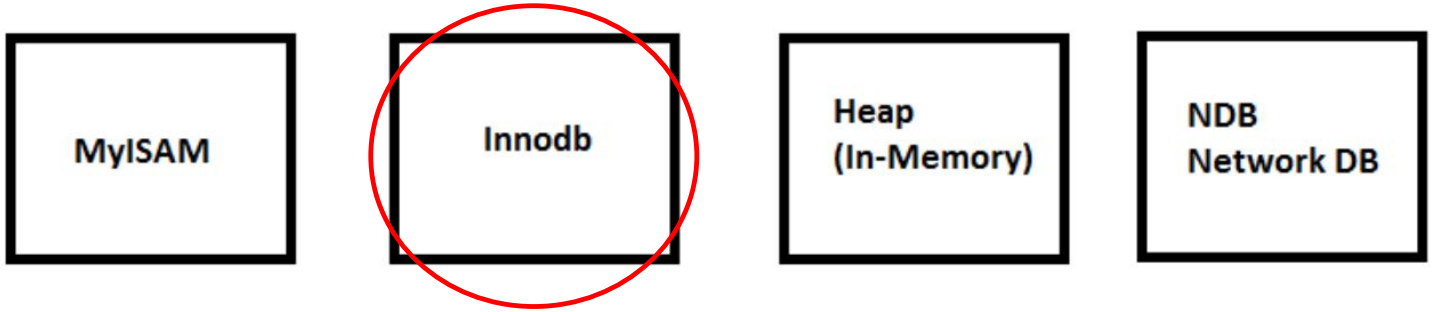
Source: https://www.usenix.org/node/174510

MySQL Architecture

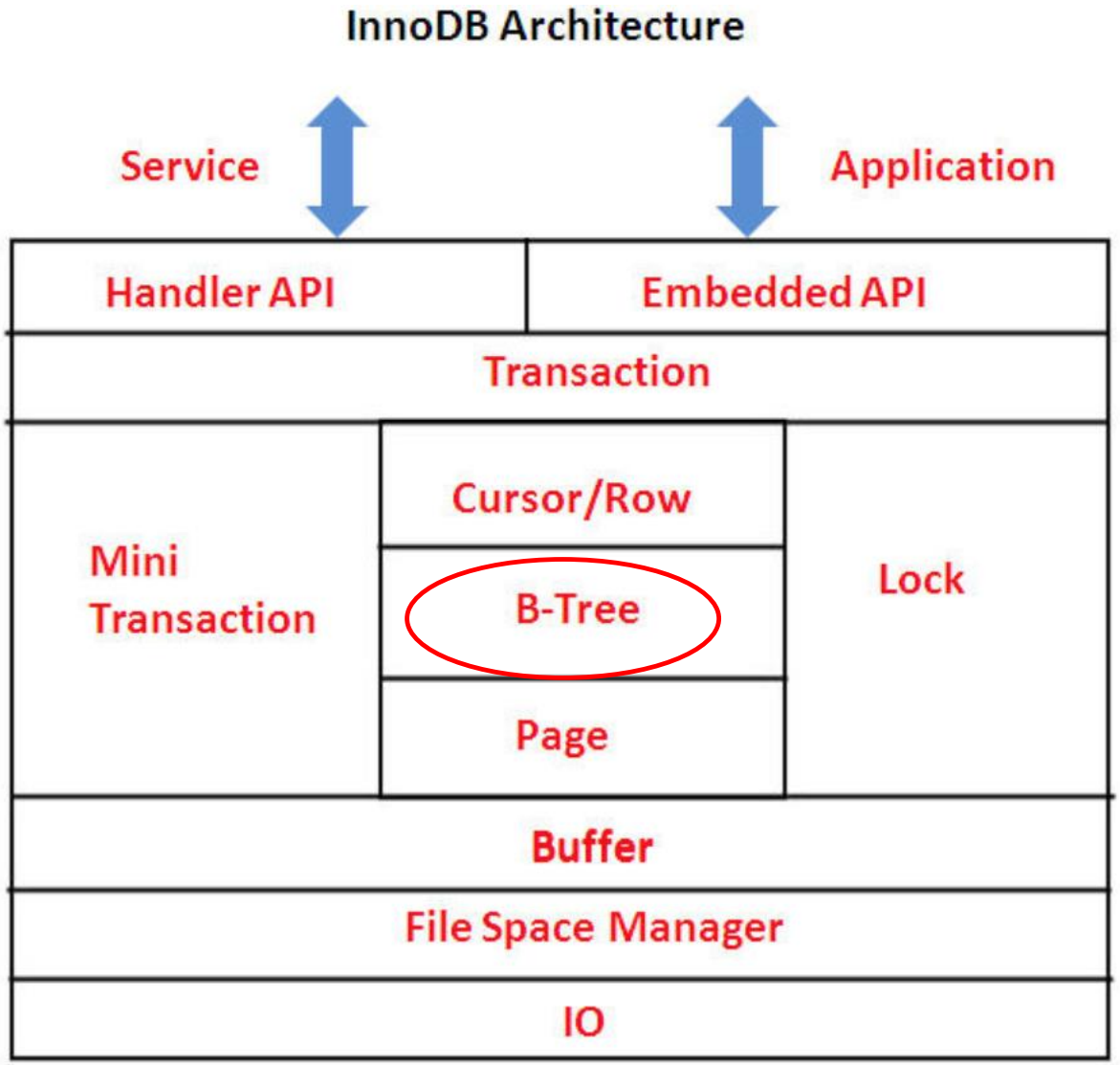
Server's functionality like connection management/authentication... is done in this layer



Responsible for storage and retrival of all available information

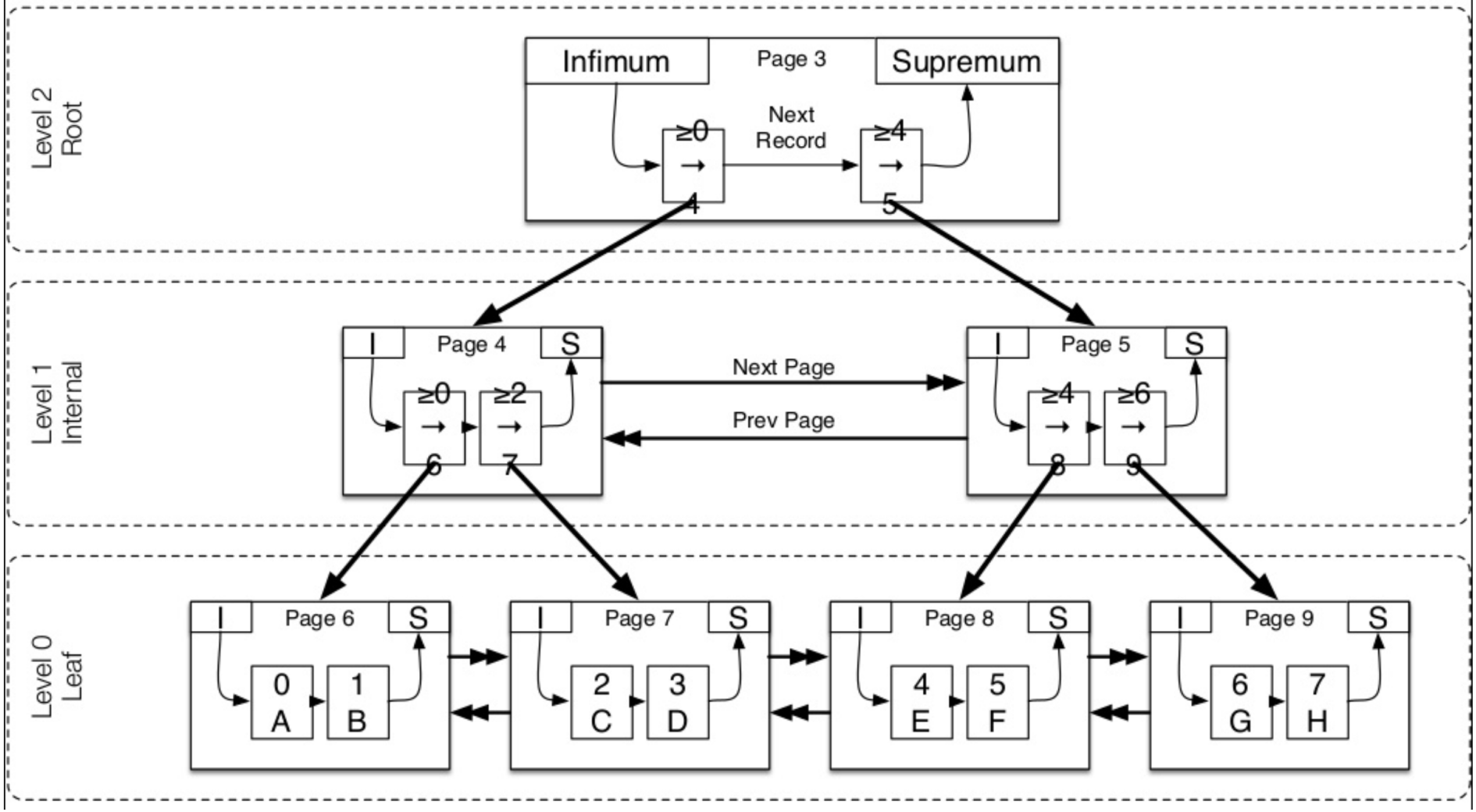


Source: https://techsoftcompute.wordpress.com/2013/08/19/mysql-internal-architecture/



Source: http://data-base-recovery.blogspot.com/2015/10/storage-engines-in-mysql.html

Index Structure



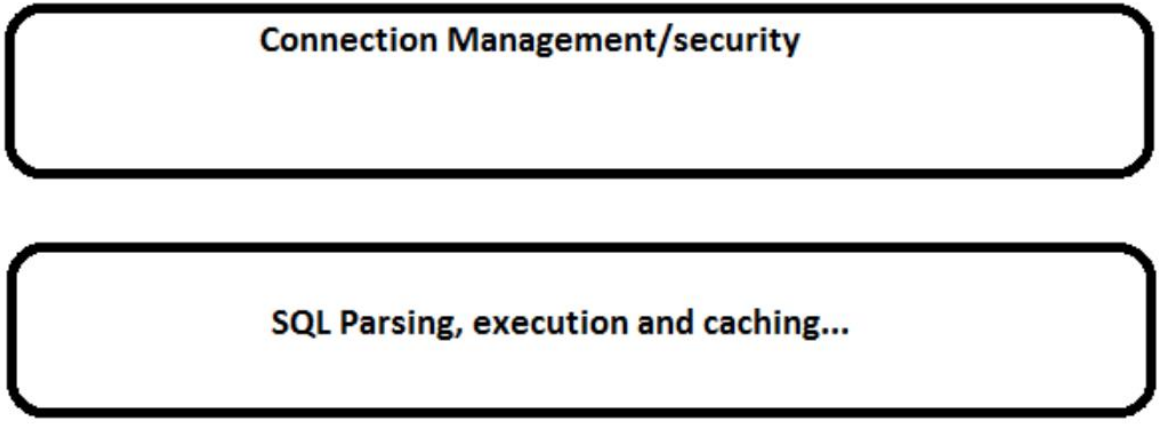
Source: https://www.slideshare.net/morgo/inno-db-presentation

Pros?

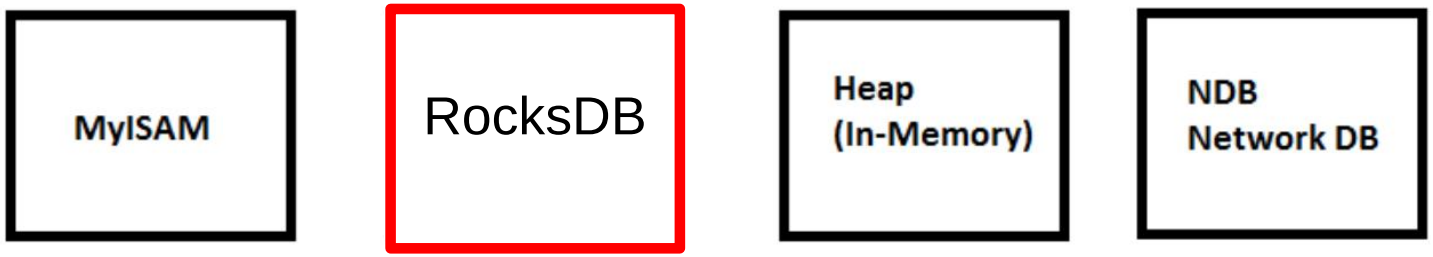
Cons?

MySQL Architecture in FaceBook

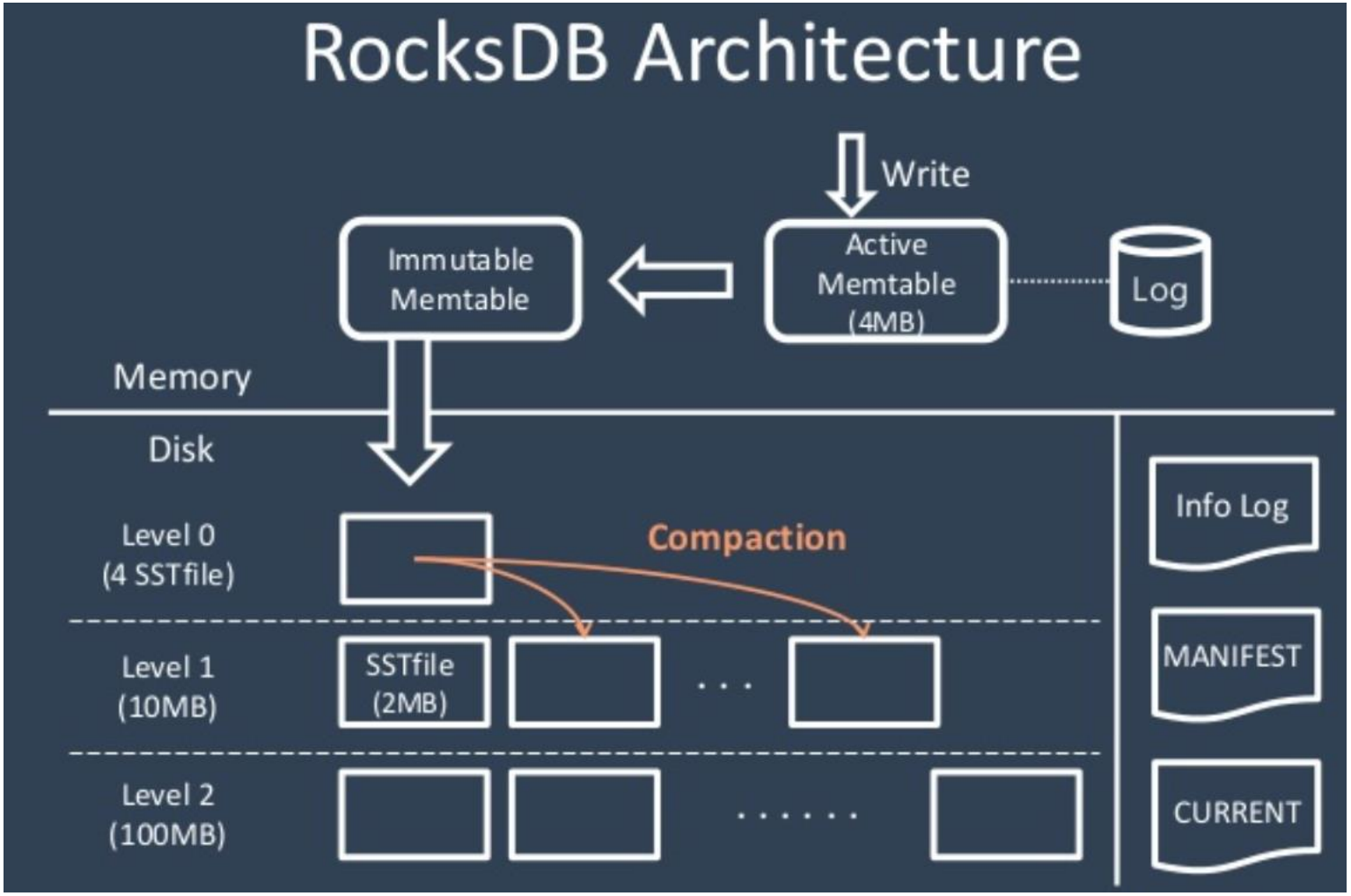
Server's functionality like connection management/authentication... is done in this layer



Responsible for storage and retrival of all available information



Source: https://techsoftcompute.wordpress.com/2013/08/19/mysql-internal-architecture/



Source: https://www.slideshare.net/meeeejin/rocksdb-detail

TAO Summary

Efficiency at scale Read latency	• Separate cache and DB • Graph-specific caching • Subdivide data centers
Write timeliness	• Write-through cache • Asynchronous replication
Read availability	• Alternate data sources

Source: https://www.usenix.org/node/174510

Thanks
Q/A