Social Hash:
an Assignment Framework for Optimizing Distributed Systems Operations on Social Networks

Alon Shalita, Brian Karrer, Igor Kabiljo, Arun Sharma, Alessandro Presta, Aaron Adcock, Herald Kllapi, and Michael Stumm

March 2016
Assignment Problem

Front-end clusters

- Cache
- Cache
- Cache

Point-of-Presence (PoP)

Alon’s HTTP requests

Igor’s HTTP requests
Assignment Problem Optimization

Point-of-Presence (PoP)

Front-end clusters

Cache

Cache

Cache
Solution Requirements

Alon’s HTTP requests

Front-end clusters

Adaptive

Point-of-Presence (PoP)

Cache

Cache

Cache
Solution Requirements

Alon’s HTTP requests

Point-of-Presence (PoP)

Stable

Front-end clusters

Cache

Cache

Cache
Solution Requirements

Alon’s HTTP requests

Point-of-Presence (PoP)

Fast decision

Front-end clusters

Cache

Cache

Cache
Social Hash framework
Social Hash framework

Components (e.g., compute clusters or storage subsystems)

Groups

Objects (e.g., data records or HTTP requests)

Static assignment

Dynamic assignment
Static assignment

- Goal: assign similar objects sent to the same group
  - Data access pattern -> represent as graph -> graph partitioning
  - Large-scale optimization: slow, time-consuming
Dynamic assignment

- Goal: adapt to maintain load balance by altering group -> component
  - hardware changes
  - dynamic workload
  - addition and removal of objects

- Two-level framework separates optimization from adaptation
  - Slow optimization -> static
  - Fast adaptation -> dynamic

- Group-to-component ratio controls tradeoff
HTTP Request Routing
Social Hash for Facebook’s web routing

- Objects: HTTP request identified by user, Components: front-end clusters
- PoP: Dynamic assignment by hash ring
Edge locality for Facebook’s web routing

- Production routing: 21k groups for 10’s of front-end clusters
- Over half of friendships are within groups
- Updated on a weekly basis (~1% movement)
Live traffic experiment: TAO miss rate

- Orange: traffic shifts
- Red: duration of test
- Green: updated Social Hash table
Storage Sharding
Assignment Problem 2: Storage sharding

Arun’s query

Objects: data records
Components: storage machines
Assignment Problem 2: Storage sharding

Arun’s query

Objects: data records
Components: storage machines
Static assignment

- Minimize fanout through bipartite graph partitioning
- Graph contains recent queries and data records
  - edge => query accesses data record
- Dotted: edge locality optimization
- Solid: fanout optimization
Storage sharding deployment

- Graph database with thousands of storage servers
  - Group-to-component ratio of 8
  - Static assignment every few months

- Results:
  - Average latencies decreased by over 50%
  - CPU utilization decreased by over 50%
Summary

- Assignment problems are common in distributed systems design
- Proposed Social Hash framework for solving assignment problems
  - Two-level design optimizes performance with graph partitioning
- Two Facebook integrations in production for over a year
  - HTTP Request Routing: > 25% reduction in TAO miss rate
  - Storage Sharding: Latency and CPU utilization reduced by over 50%