Facebook Cache Invalidation Pipeline

Melita Mihaljevic (melitam@fb.com)
Production Engineer
### Agenda

<table>
<thead>
<tr>
<th></th>
<th>Introduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>What makes keeping cache consistent challenging?</td>
</tr>
<tr>
<td>3</td>
<td>Cache infrastructure and Invalidation Pipeline</td>
</tr>
<tr>
<td>4</td>
<td>Q&amp;A</td>
</tr>
</tbody>
</table>
Introduction

There are only two hard things in computer science.

Cache invalidation, naming things, and off-by-one errors.
What makes keeping cache consistent challenging?
Cache (in)consistency

Son: dad what is inconsistency

father: I don't know son, we are making text modification
Cache (in)consistency

EVENTUAL CONSISTENCY
Challenges

- Rapidly evolving product
- Massive scale
- Social Graph at the center
Evolving product
Massive scale

- 1.44 billion users
- 829 million daily active users
- 654 million mobile daily active users
- hundreds of thousands of servers
Social graph

People are only one dimension of the social graph
Cache infra and Invalidation Pipeline
TAO
The social graph

Representation for most FB data
• Nodes are “objects”
• Edges are “associations”

Complex queries
• Rendering one page requires multiple accesses to the graph
• Fast access required
TAO
Two level, write-through cache with graph semantics

Server-side graph abstraction
- Simplification of client code

Two level cache
- Leaders coordinate access to DB
- Followers cache most data
- Graceful failover
Cache consistency inside data center

Web servers

Follower cache

Leader cache

Database

Web servers

Follower cache

Leader cache

Database
Cache consistency across data centers

- Master data center
  - Web servers
  - Master data center
  - Writes forwarded to master
  - Inval and refill embedded in SQL
  - Delivery after DB replication done

- Replica data center
  - Replica data center
Detecting inconsistencies

- **Bad:**
  - Users can’t see comments on Zuck’s post

- **Better:**
  - Measuring consistency while not impacting consistency of the system
  - Detecting data that is inconsistent
    - helps finding the root cause
Fixing inconsistencies

• Fixing the root cause is usually enough
  ▪ Invalidation messages will be delivered when the system is healthy
    ▪ Sometimes cleanup is necessary
    ▪ Invalidating shards manually

• “Permanent” inconsistency
  ▪ Bugs in invalidation pipeline
    ▪ Cache gets evicted after ttl expires or is replaced by newer value
      ▪ Bad for things that don’t change often
Consistency is key for keeping your dog healthy and happy!