CacheScale: Saving Cash by Using Less Cache
Timothy Zhu, Anshul Gandhi, and Mor Harchol-Balter (Carnegie Mellon University); Michael A. Kozuch (Intel Labs)

Problem Overview

Load varies significantly throughout the day:
Can we shrink the caching tier during low load
to reduce operational costs?

Load Balancer
Application Tier
Caching Tier
Database

\[ \lambda_{DB} = \lambda (1-p) \]

Peak load to low load ratio

% cache reduction

% of data cached

Hit rate, p

Small decrease in
desired hit rate

Large decrease in
required cache size

Small decrease in
required cache size

Substantial savings for a range of Zipf popularity
distributions with varying skew parameters, \( \alpha \)

Larger ratios between peak load and low load
\( \Rightarrow \) More potential for cache size reduction

Results

Load
\( \lambda \) req/sec

\( \lambda \) = \( \lambda (1-p) \)

req/sec

1. Reduce database load
2. Reduce data access latency

Performance can temporarily suffer when shrinking
the cache due to losing a lot of “hot” data

Transferring the hot data before shrinking the
cache mitigates this problem

Before

Retiring
Caching Tier

Primary

Caching Tier

Divide cache instances into retiring group and primary group

After

Retiring
Caching Tier

Primary

Caching Tier

If incoming request hits in
retiring group, then transfer
data to primary group

Transferring “hot” data

Caching Tier

Primary

Transfer

Shrink the cache

Response time stabilizes

Mean response time (ms)

Time (min)