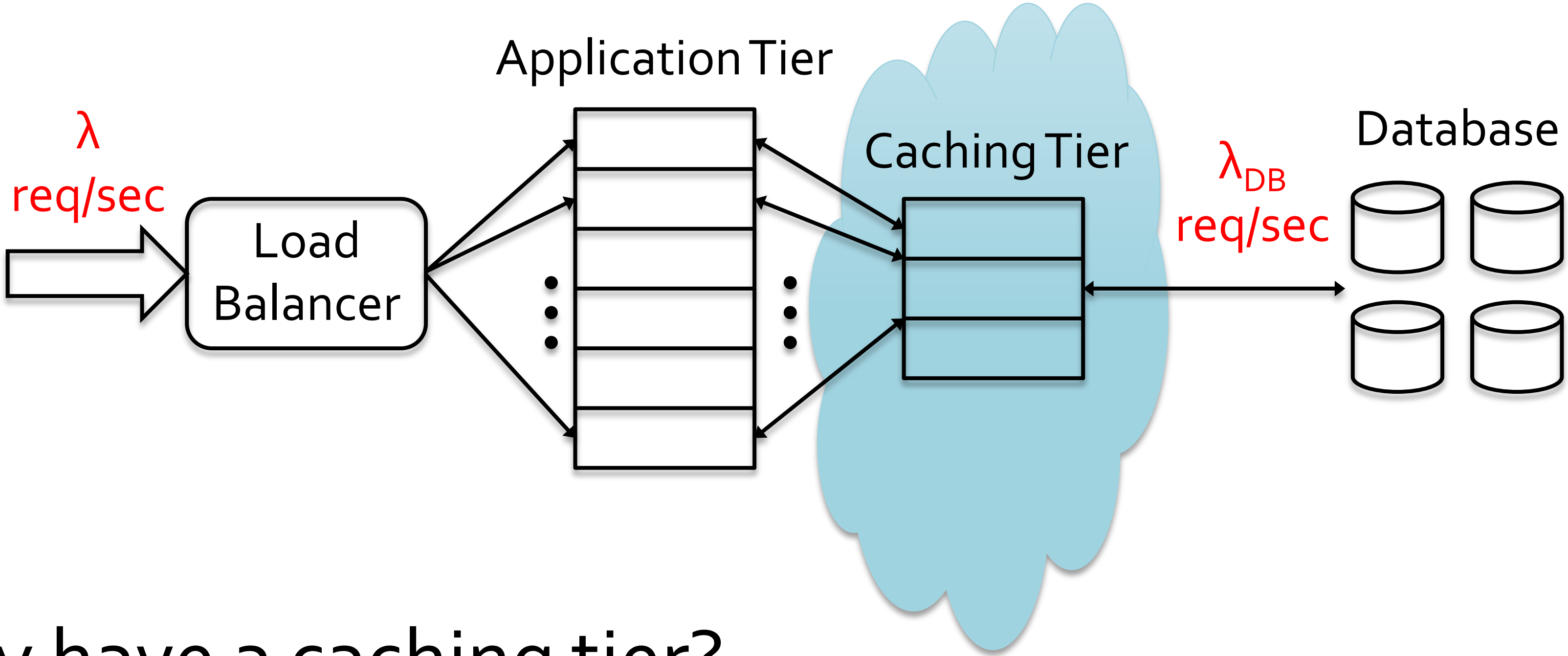
# Saving Cash by Using Less (Mem)Cache

Timothy Zhu
Carnegie Mellon University

Anshul Gandhi, Mor Harchol-Balter
Carnegie Mellon University

Michael A. Kozuch
Intel Labs

# Application in the Cloud

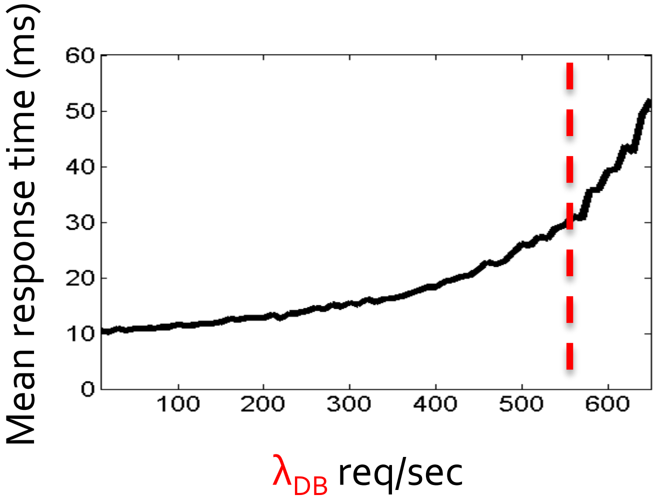


Why have a caching tier?

1. Reduce database (DB) load   ($\lambda_{DB} << \lambda$)

# Application in the Cloud

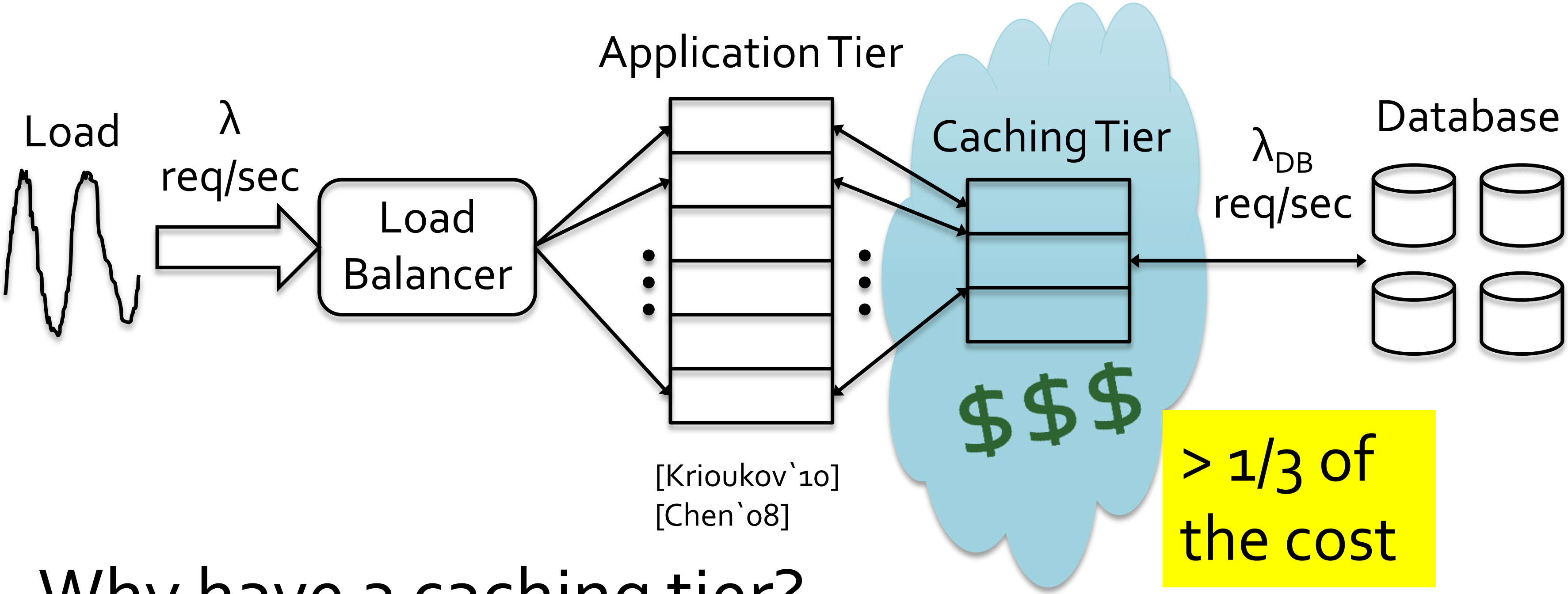
DB response time rapidly increases at high DB load



Why have a caching tier?

1. Reduce database (DB) load ($\lambda_{DB} << \lambda$)

# Application in the Cloud



> 1/3 of the cost

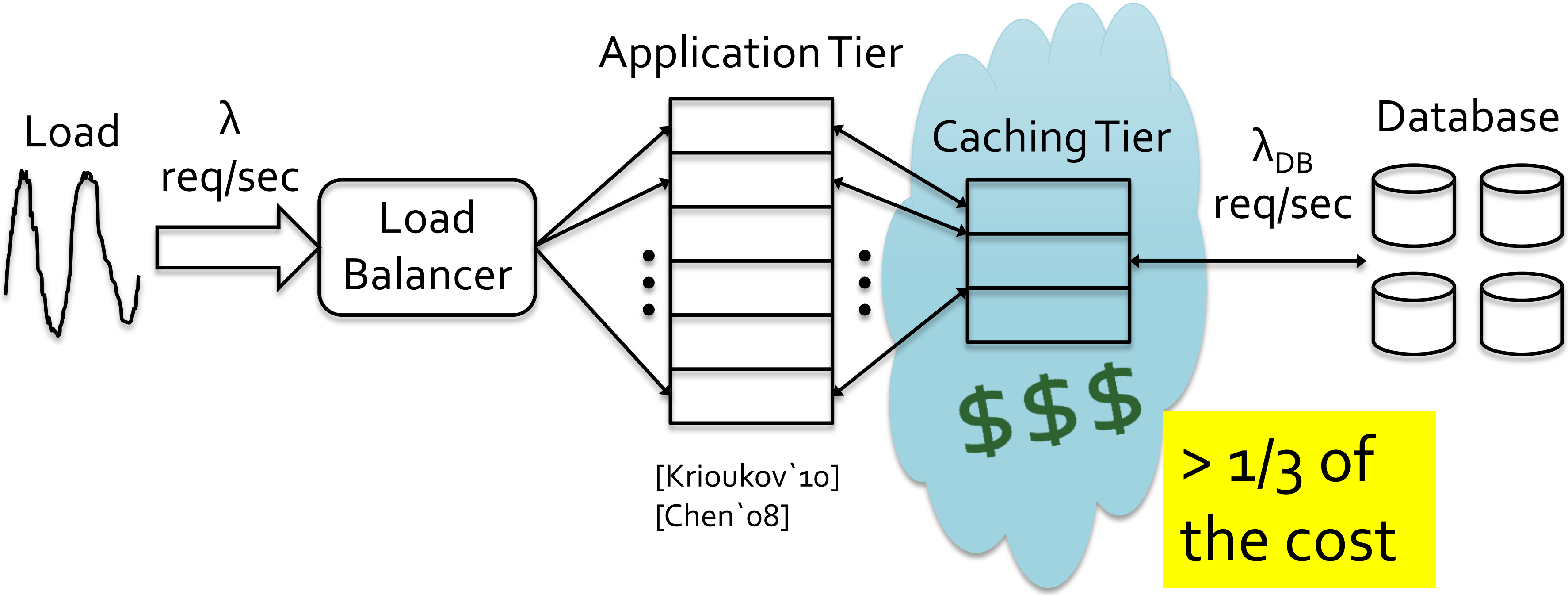
[Krioukov`10]
[Chen`08]

[Ousterhout`10]

Why have a caching tier?

1. Reduce database (DB) load $(\lambda_{DB} << \lambda)$
2. Reduce latency

# Application in the Cloud



> 1/3 of the cost

[Krioukov`10]
[Chen`08]

[Ousterhout`10]

Shrink your cache during low load

# Key Questions

1. Will cache misses overwhelm the DB?

   $\lambda_{DB}$ too high?

2. Are the savings significant?

3. What about the “hot” data?

# Key Questions

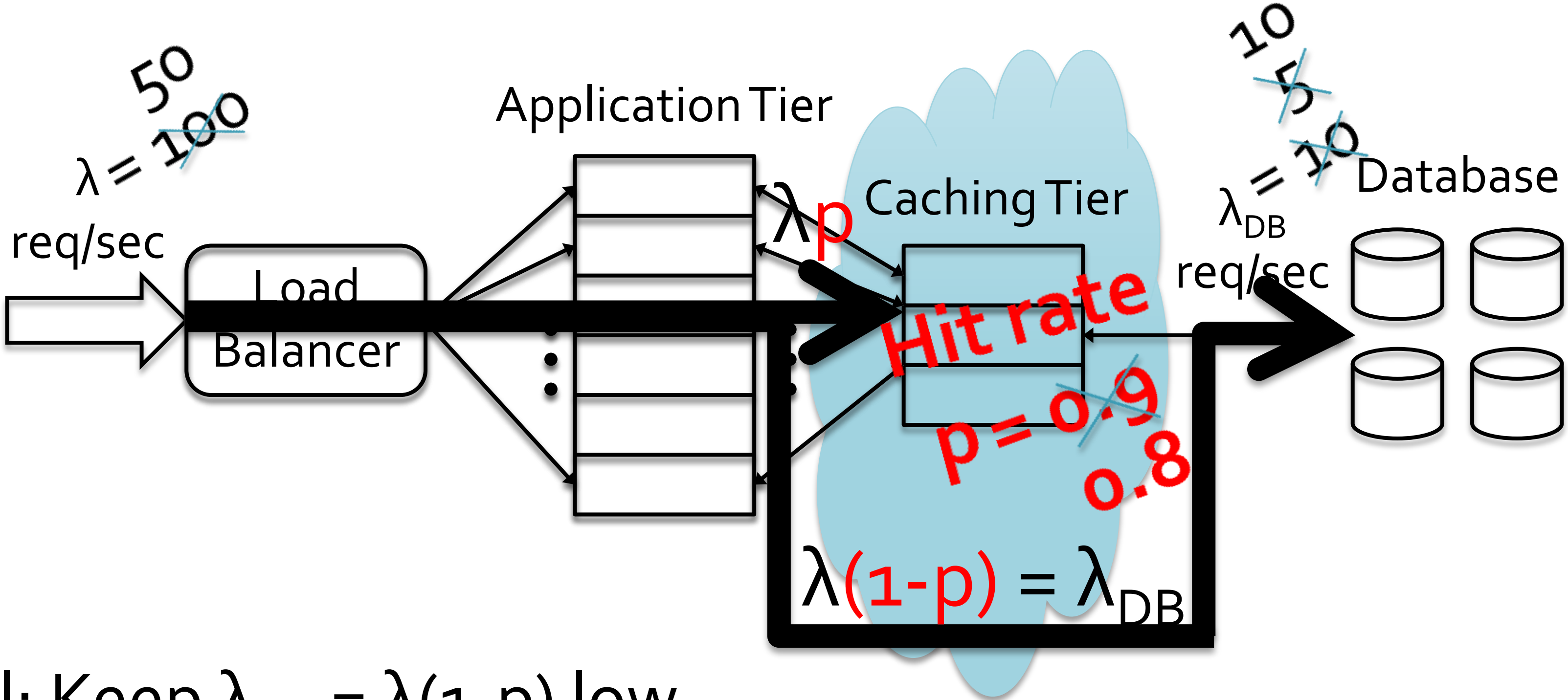
1. Will cache misses overwhelm the DB?

   $\lambda_{DB}$ too high?

2. Are the savings significant?

3. What about the “hot” data?

# Will cache misses overwhelm the DB?

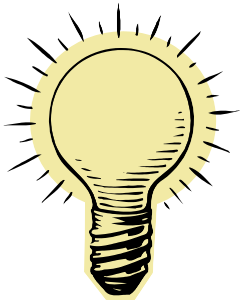


Goal: Keep $\lambda_{DB} = \lambda(1-p)$ low

If $\lambda$ drops → (1-p) can be higher

→ p can be lower

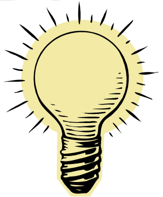
→ SAVE $$$



# Key Questions

1. Will cache misses overwhelm the DB?

   No, we can afford a lower hit rate at low load



2. Are the savings significant?

3. What about the "hot" data?

# Key Questions

1. Will cache misses overwhelm the DB?

   No, we can afford a lower hit rate at low load

2. Are the savings significant?

3. What about the "hot" data?

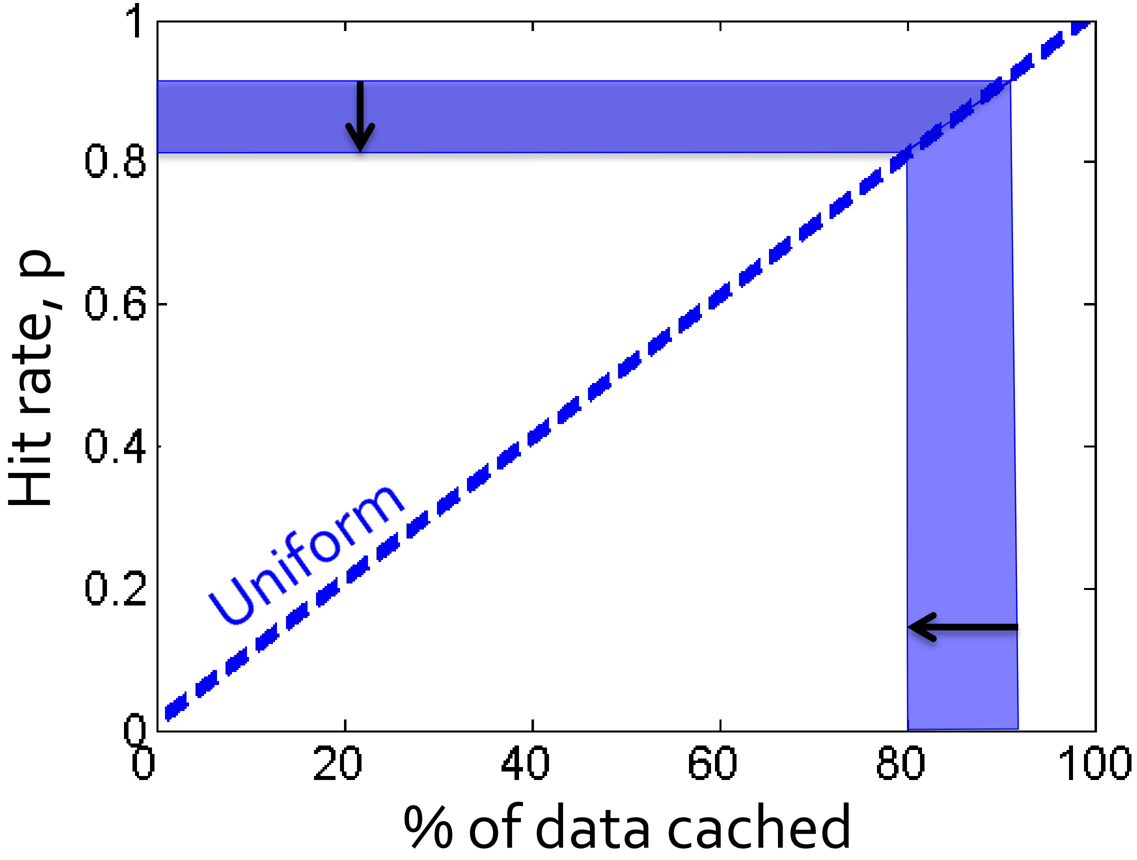
# Are the savings significant?

- It depends on the popularity distribution

Small decrease in hit rate

Uniform

Small decrease in caching tier size

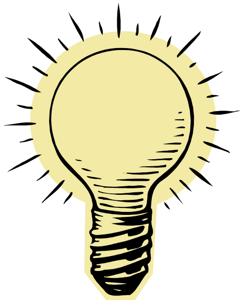


# Are the savings significant?

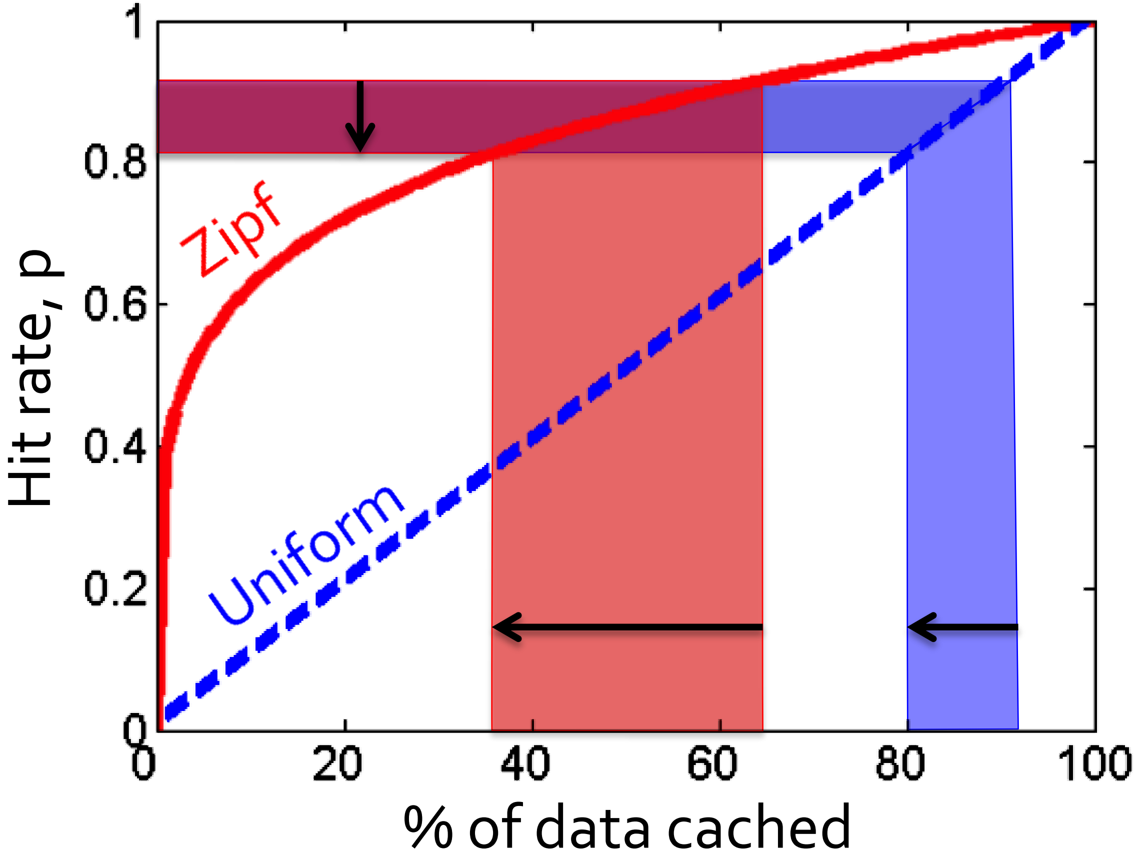
- It depends on the popularity distribution

Small decrease in hit rate

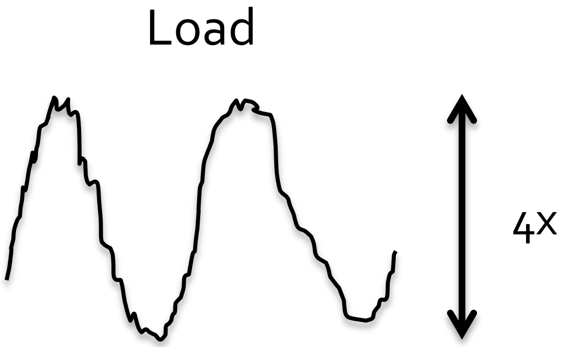
Zipf

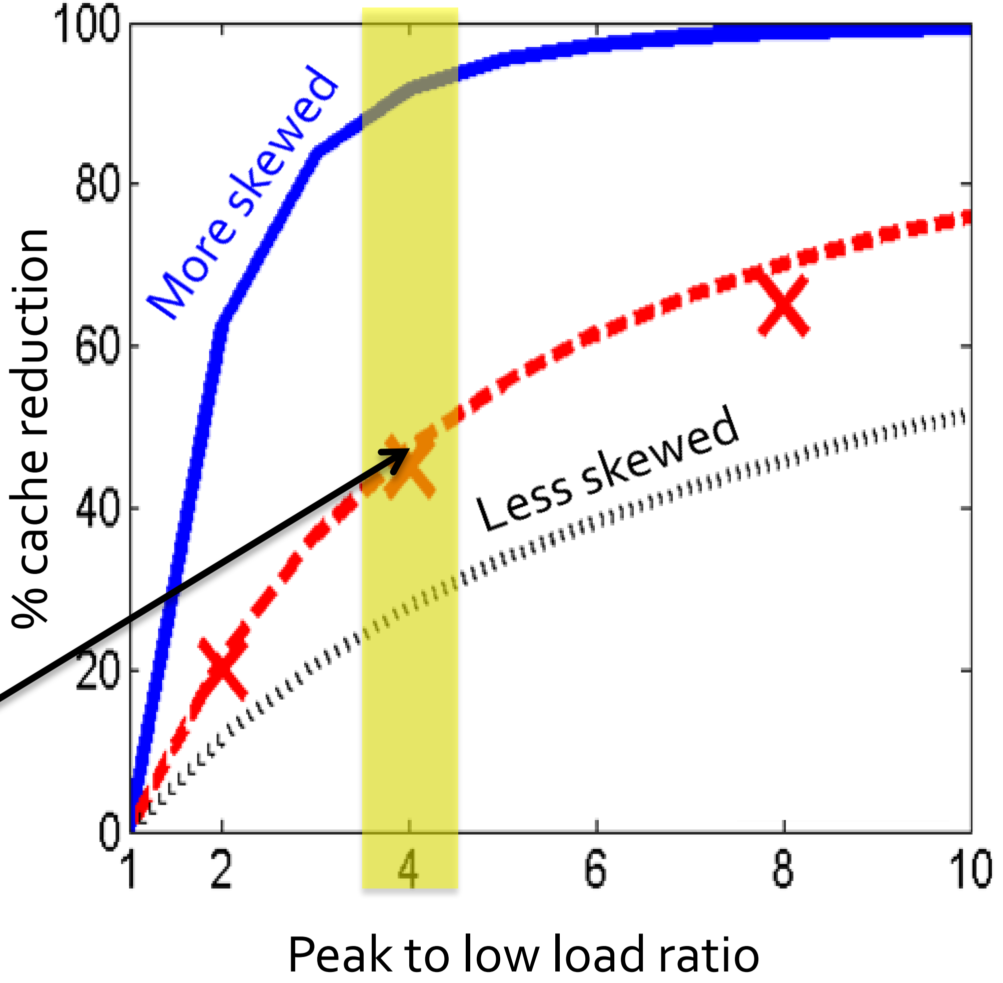


Large decrease in caching tier size



# Savings





50% cache savings

# Key Questions

1. Will cache misses overwhelm the DB?

   No, we can afford a lower hit rate at low load

2. Are the savings significant?

   Small decrease in hit rate $\xrightarrow[\text{Zipf}]{}$ **Large** decrease in caching tier size

3. What about the "hot" data?

# Key Questions

1. Will cache misses overwhelm the DB?

   No, we can afford a lower hit rate at low load

2. Are the savings significant?

   Small decrease in hit rate $\xrightarrow[\text{Zipf}]{}$ **Large** decrease in caching tier size

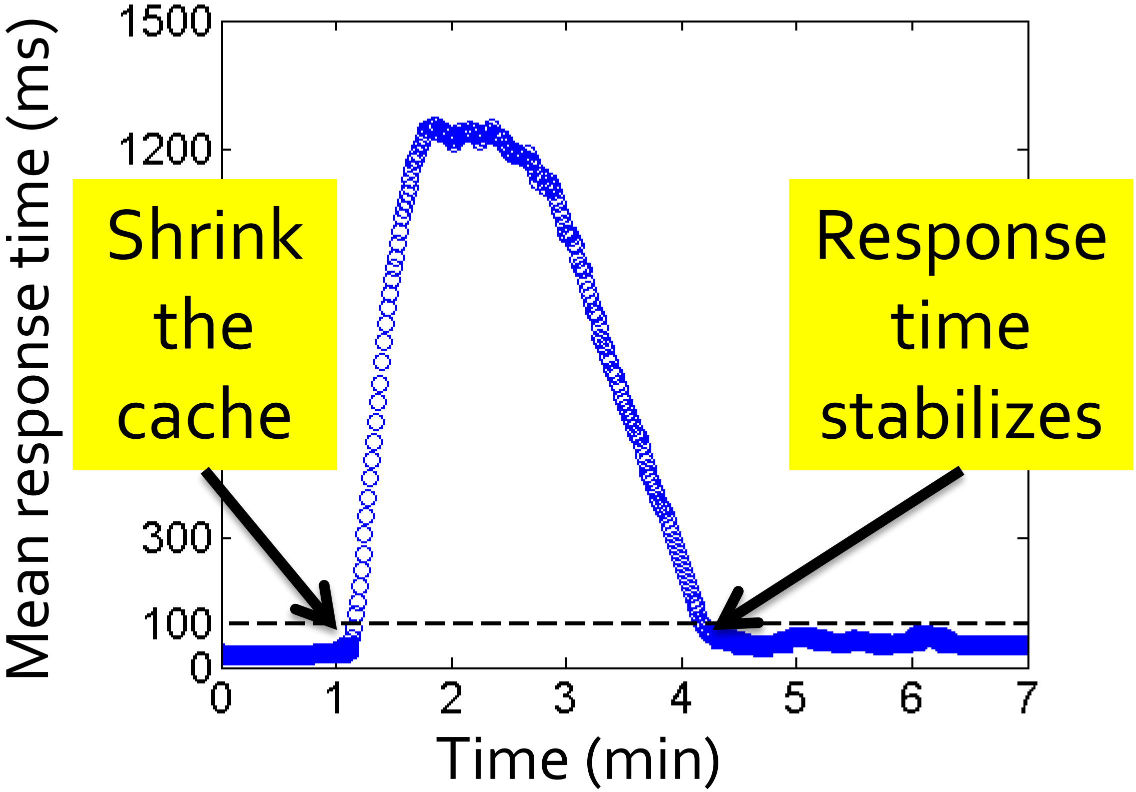
3. What about the "hot" data?

   a. Is there a problem?

   b. What can we do about it?

# Is there a problem?

- Performance can temporarily suffer if we lose a lot of hot data

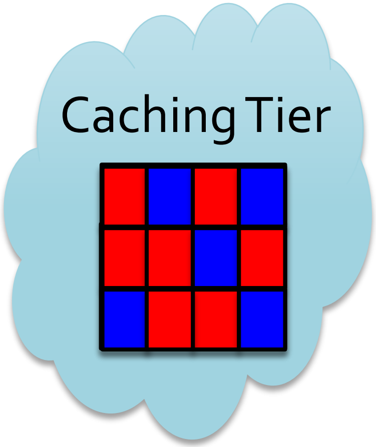


Shrink the cache

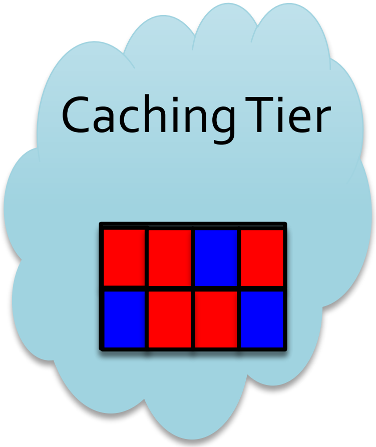
Response time stabilizes

# What can we do about the hot data?

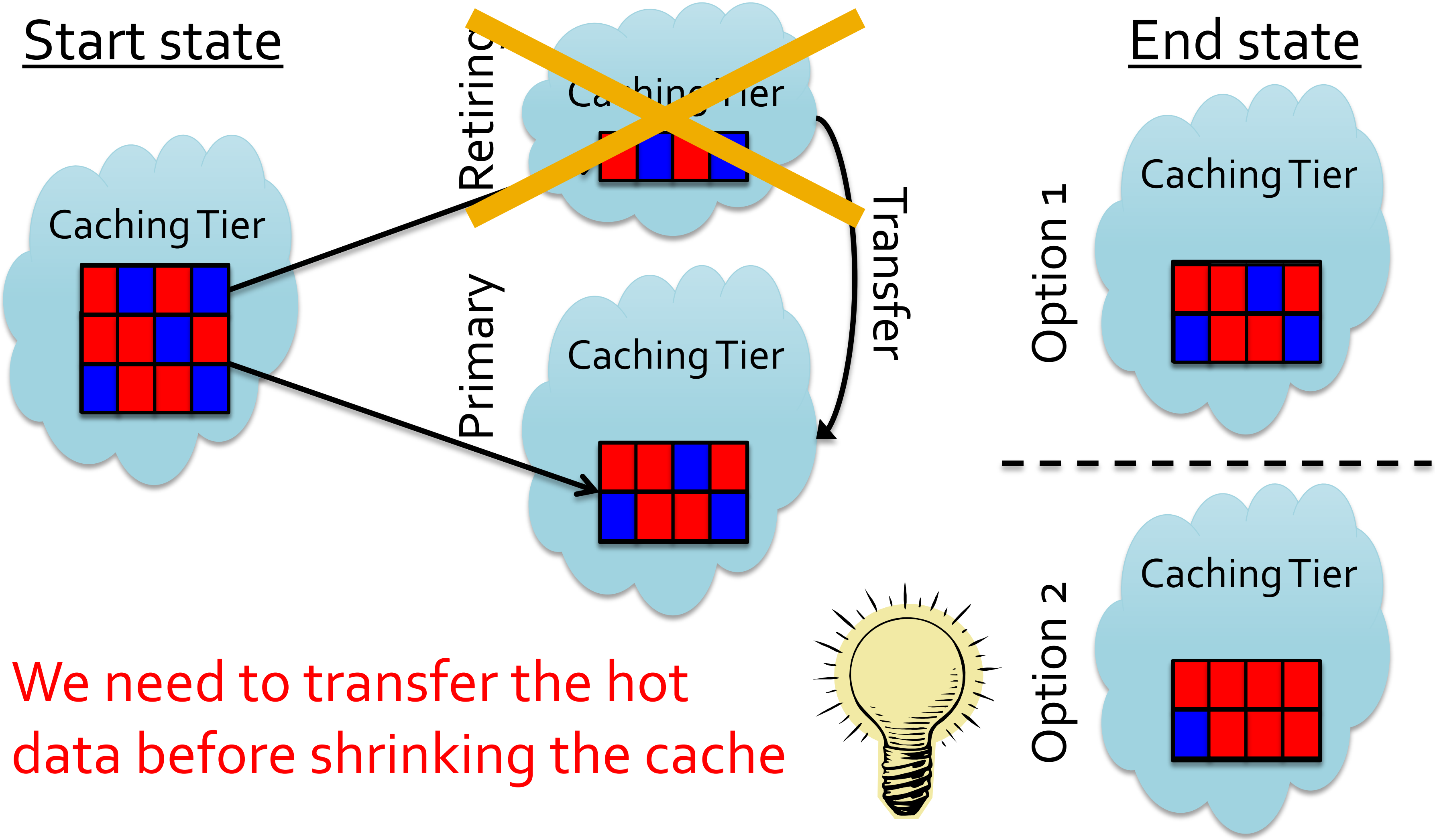
Start state



End state



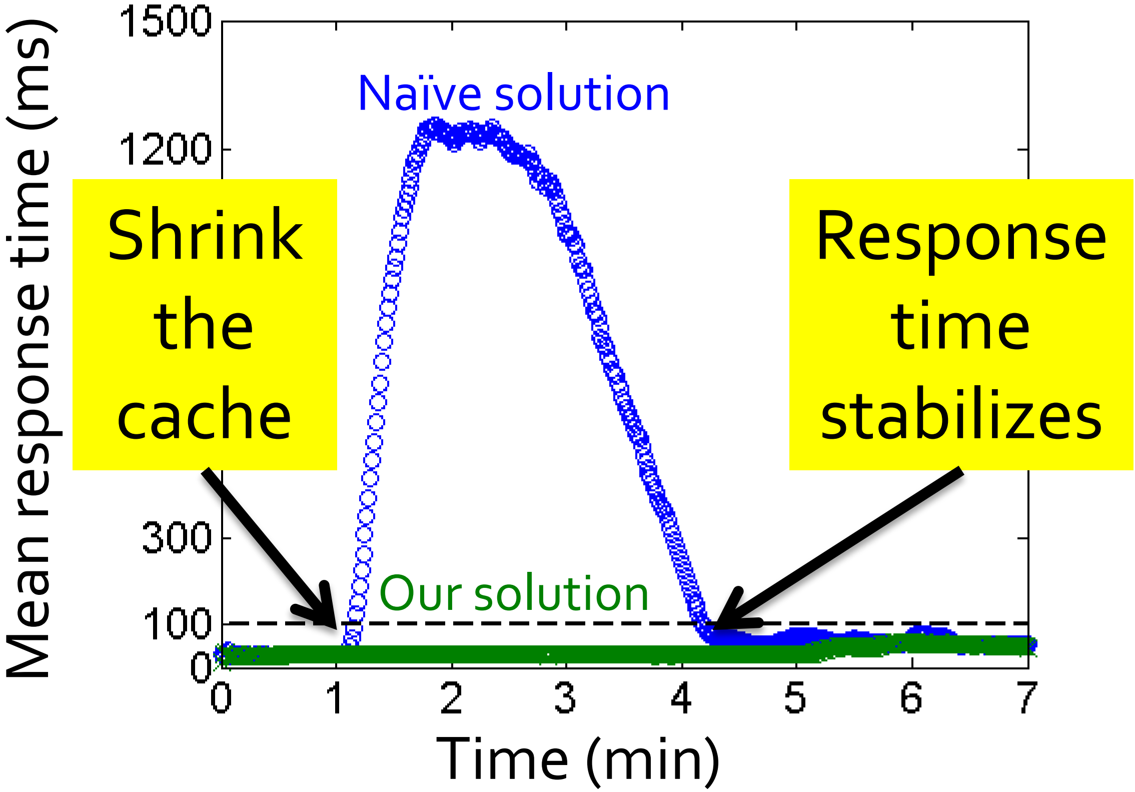
# What can we do about the hot data?



We need to transfer the hot data before shrinking the cache

# Effect of transferring hot data

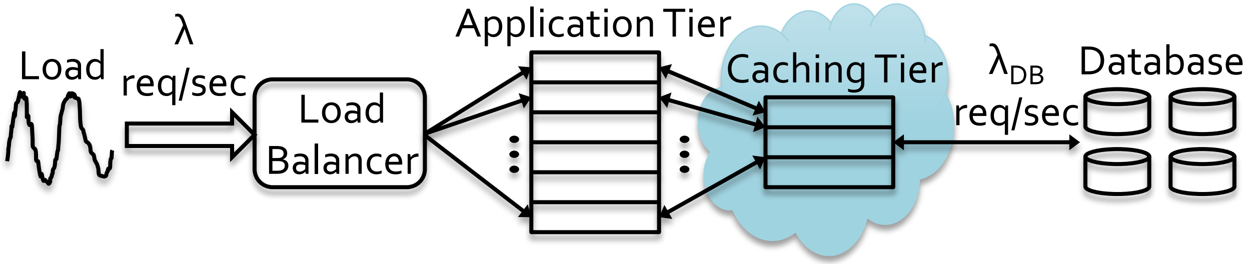
- Transferring the hot data before shrinking the cache eliminates performance degradation



Shrink the cache

Response time stabilizes

# Conclusion



1. Will cache misses overwhelm the DB?

   No, we can afford a lower hit rate at low load

2. Are the savings significant?

   Small decrease in hit rate $\xrightarrow{\text{Zipf}}$ **Large** decrease in caching tier size

3. What about the “hot” data?

   We need to transfer the hot data before shrinking the cache

Use less cache $\xrightarrow{\text{Low load}}$ Save $$$