CS 265

Stratos Idreos

BIG DATA SYSTEMS

NoSQL | Neural Networks | Image AI | LLMs | Data Science

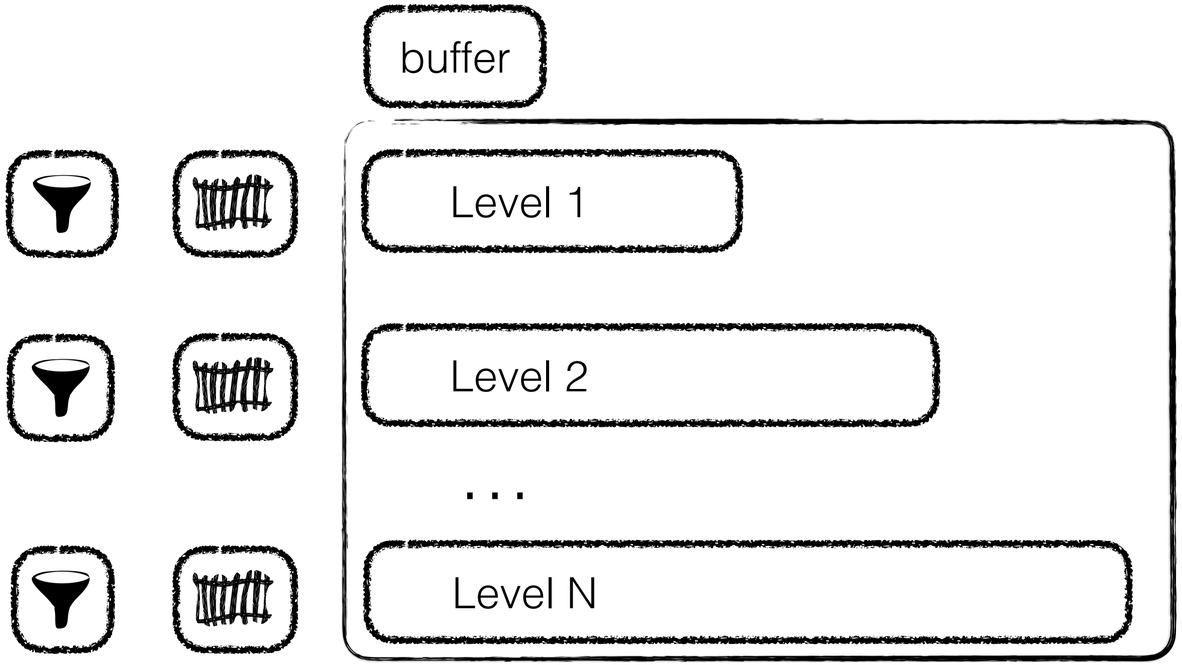
BITS PER ENTRY IN FILTERS: OPTIMIZED OUT

Monkey: **O**ptimal **N**avigable **Key**-Value Store @SIGMOD2017

DASlab @ Harvard SEAS

BITS PER ENTRY IN FILTERS: OPTIMIZED OUT

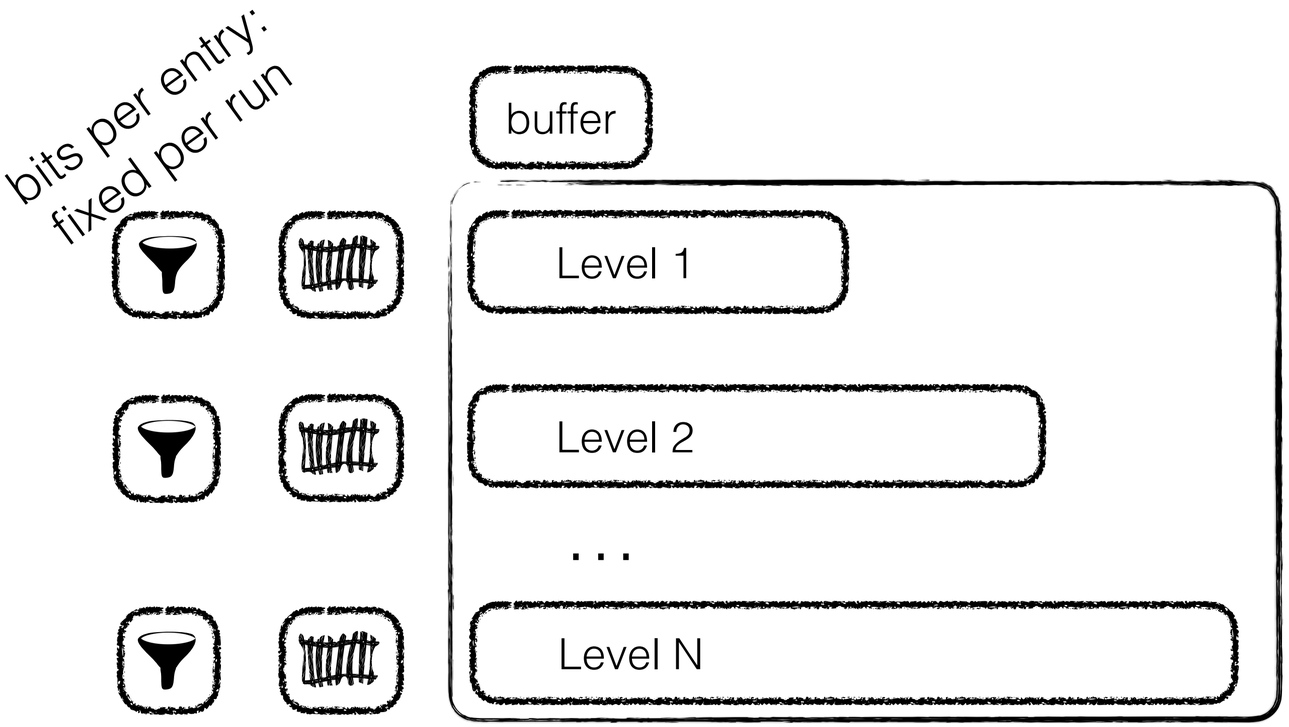
Monkey: **O**ptimal **N**avigable **Key**-Value Store @SIGMOD2017



DASlab @ Harvard SEAS

BITS PER ENTRY IN FILTERS: OPTIMIZED OUT

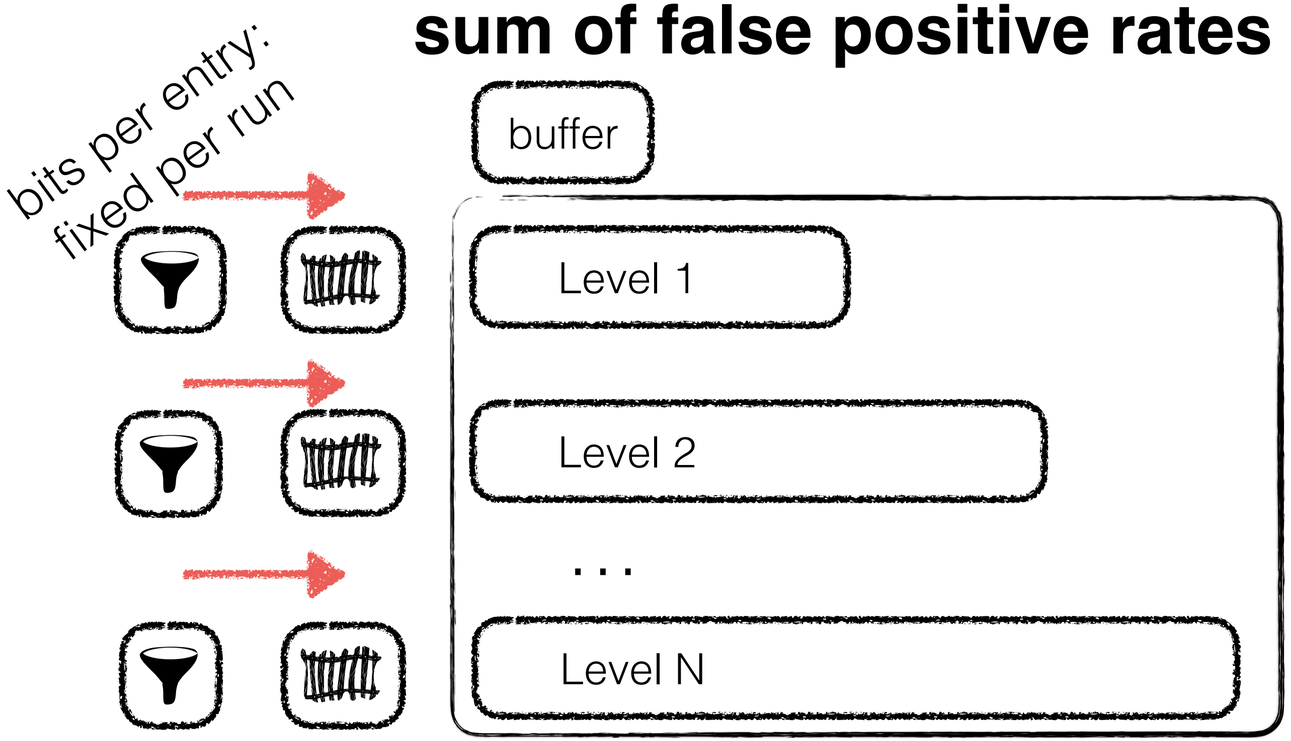
Monkey: **O**ptimal **N**avigable **Key**-Value Store @SIGMOD2017



BITS PER ENTRY IN FILTERS: OPTIMIZED OUT

Monkey: **O**ptimal **N**avigable **Key**-Value Store @SIGMOD2017

worst lookup cost:
sum of false positive rates

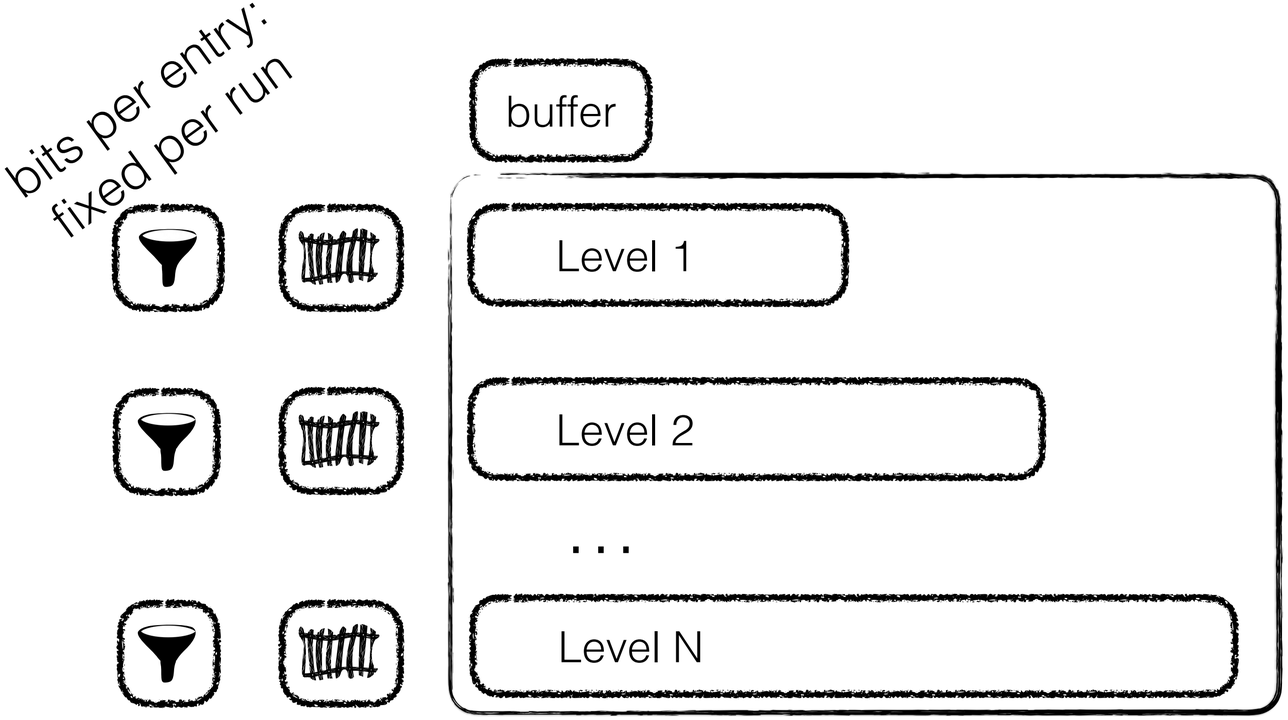


DASlab @ Harvard SEAS

BITS PER ENTRY IN FILTERS: OPTIMIZED OUT

Monkey: **O**ptimal **N**avigable **Key**-Value Store @SIGMOD2017

the same memory budget
is more impactful at smaller levels

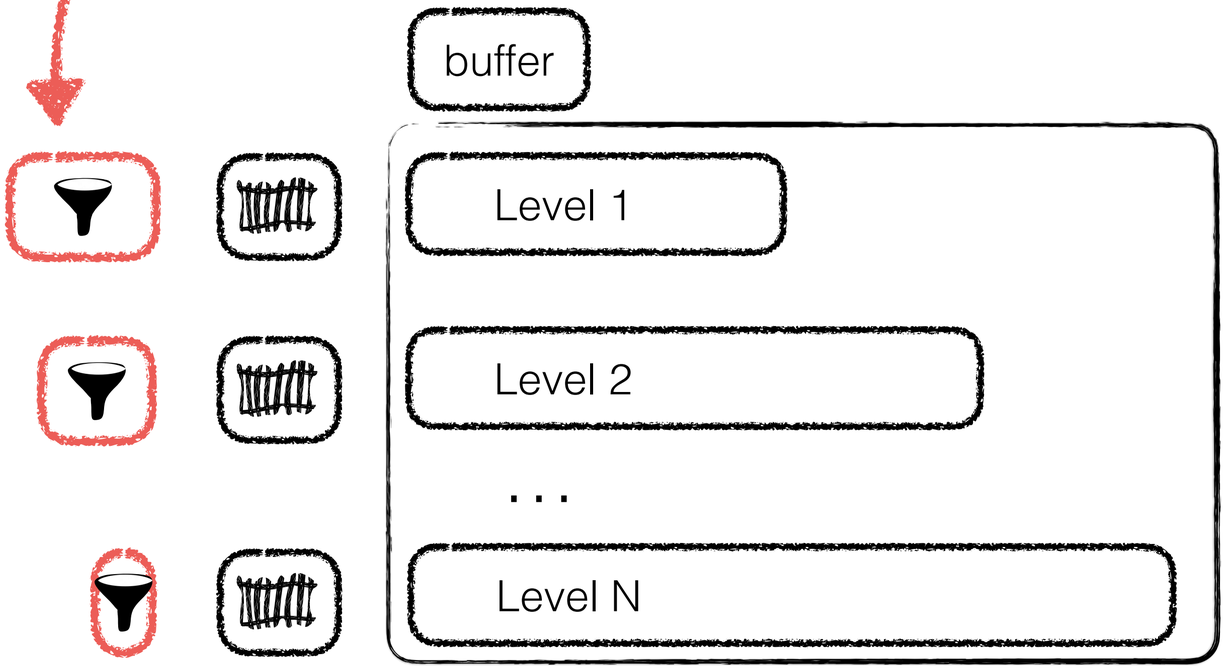


DASlab @ Harvard SEAS

BITS PER ENTRY IN FILTERS: OPTIMIZED OUT

Monkey: **O**ptimal **N**avigable **Key**-Value Store @SIGMOD2017

the same memory budget
is more impactful at smaller levels

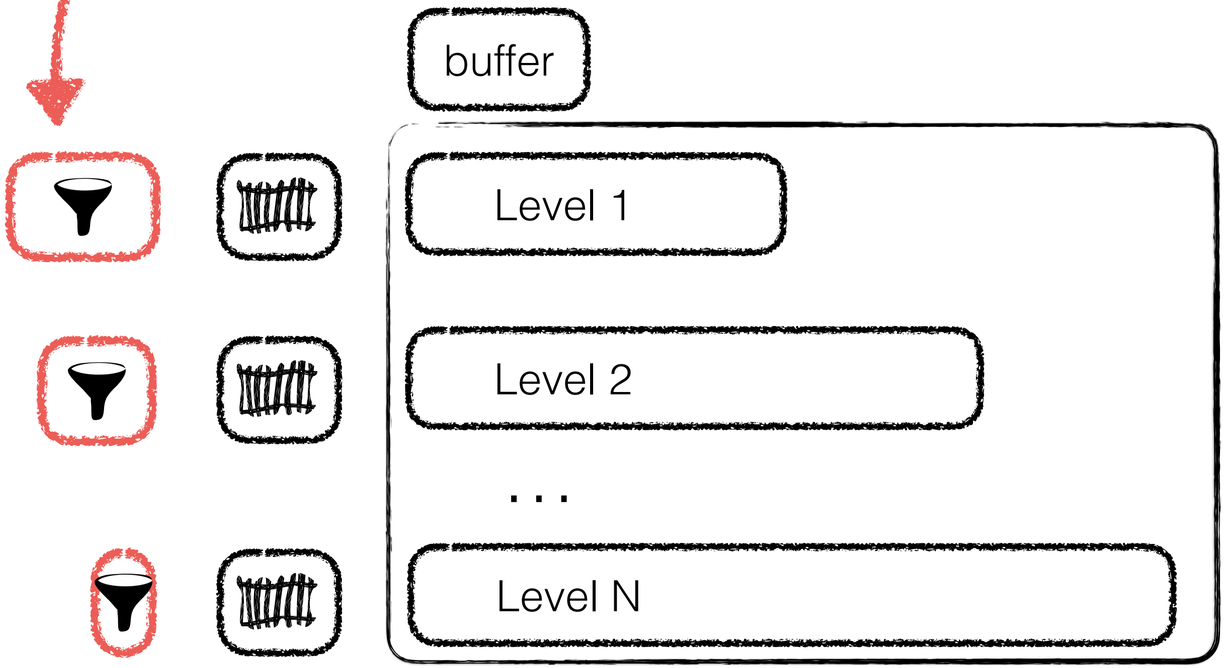


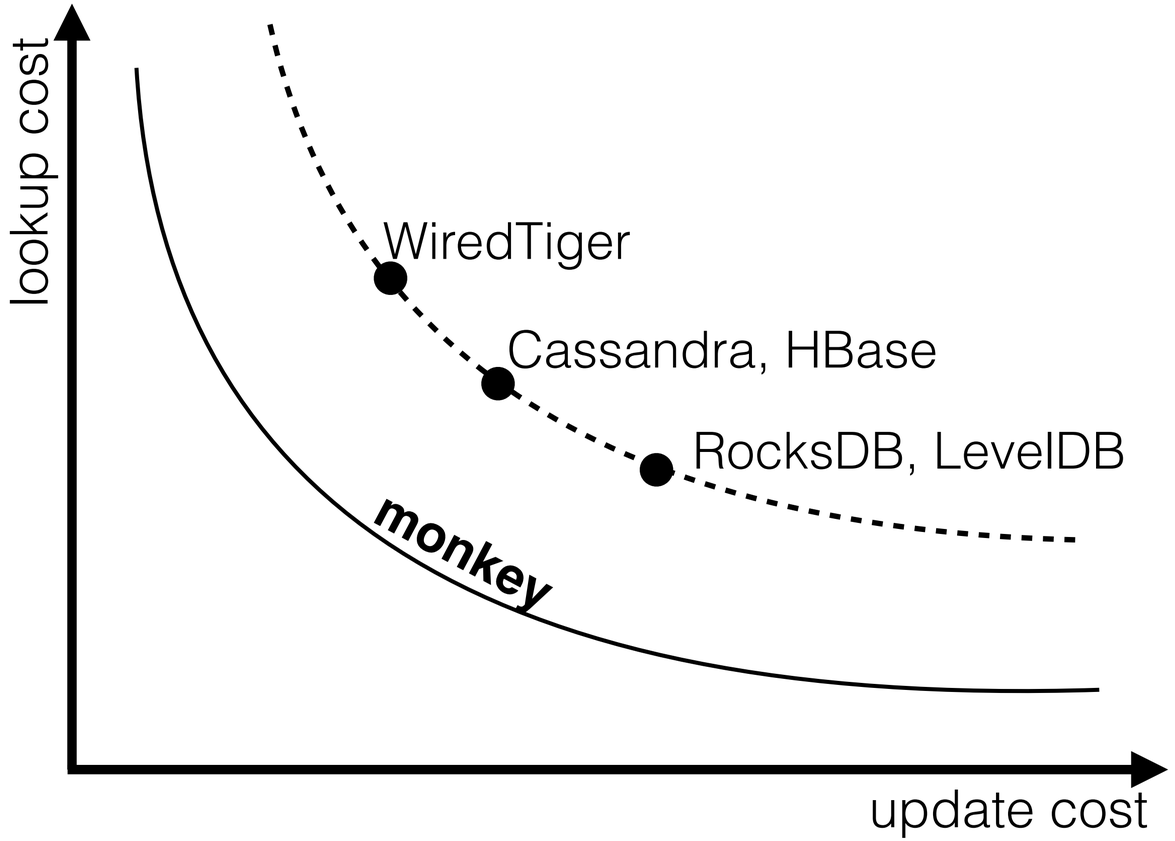
DASlab @ Harvard SEAS

BITS PER ENTRY IN FILTERS: OPTIMIZED OUT

Monkey: **O**ptimal **N**avigable **Key**-Value Store @SIGMOD2017

the same memory budget
is more impactful at smaller levels



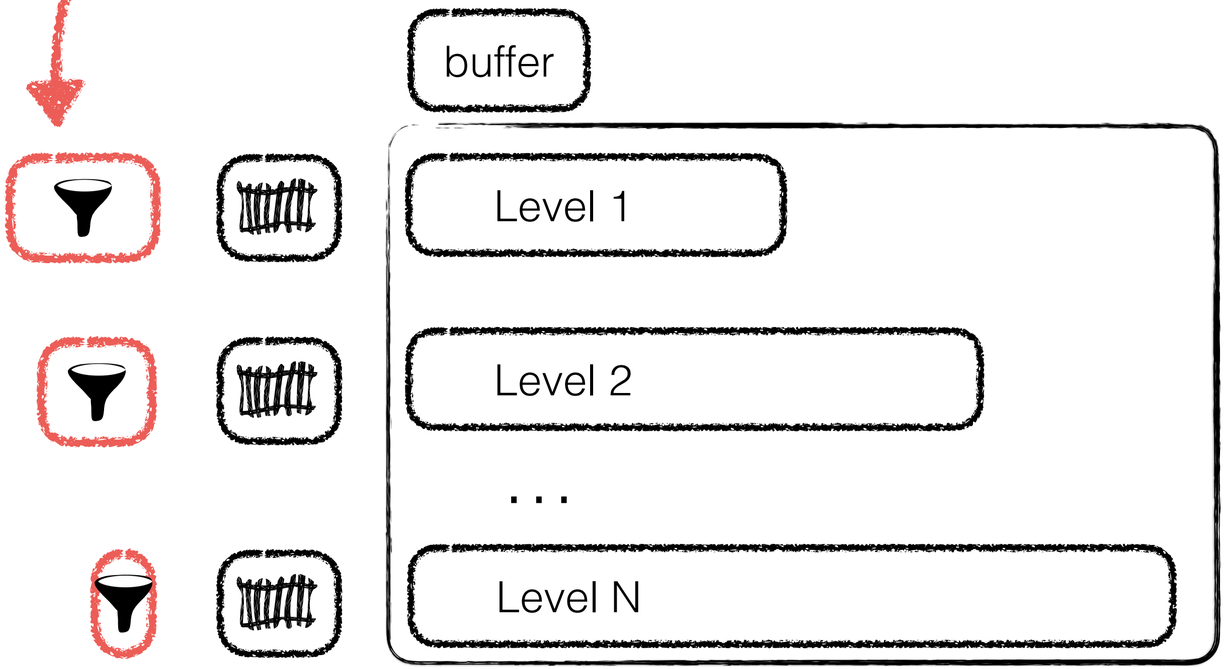


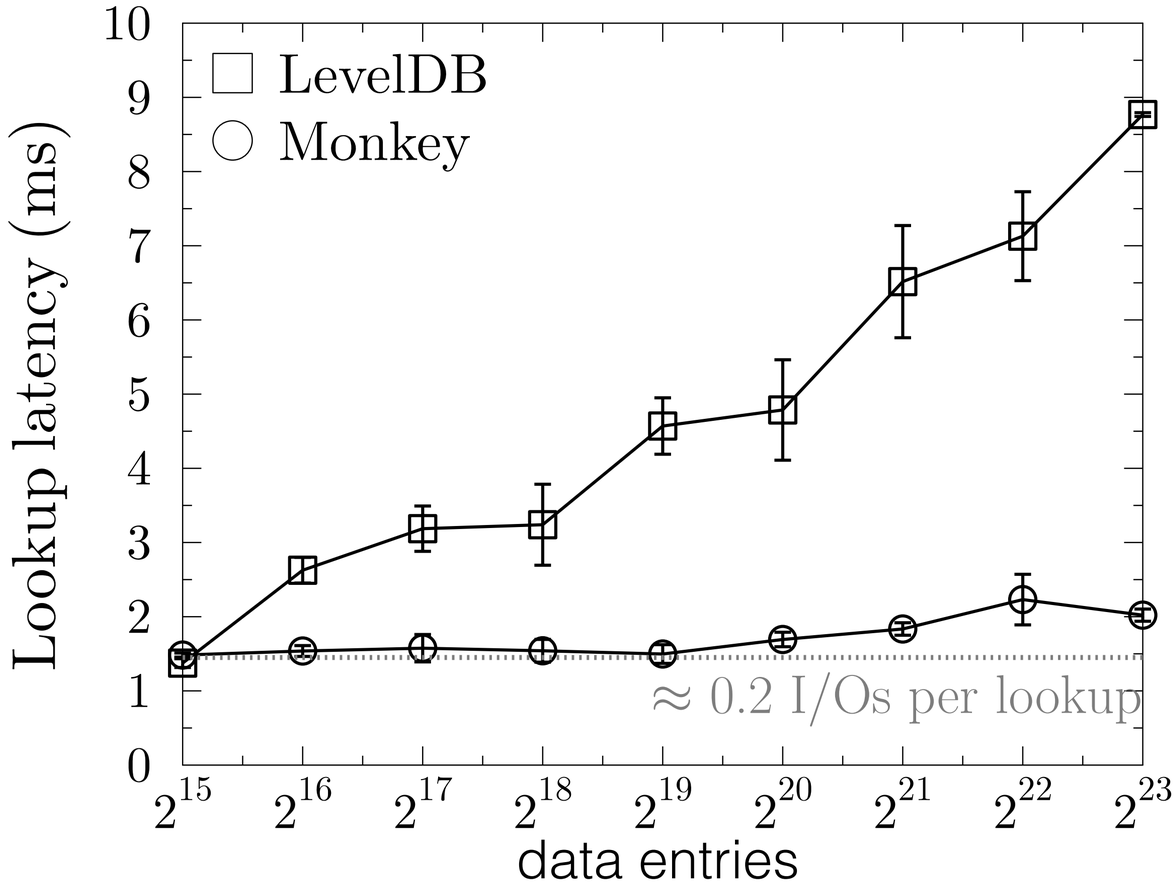
DASlab @ Harvard SEAS

BITS PER ENTRY IN FILTERS: OPTIMIZED OUT

Monkey: **O**ptimal **N**avigable **Key**-Value Store @SIGMOD2017

the same memory budget is more impactful at smaller levels





uniform, zero result, point queries, entry size=1KB

DASlab @ Harvard SEAS

MERGE POLICY: SHOULD BE TUNED

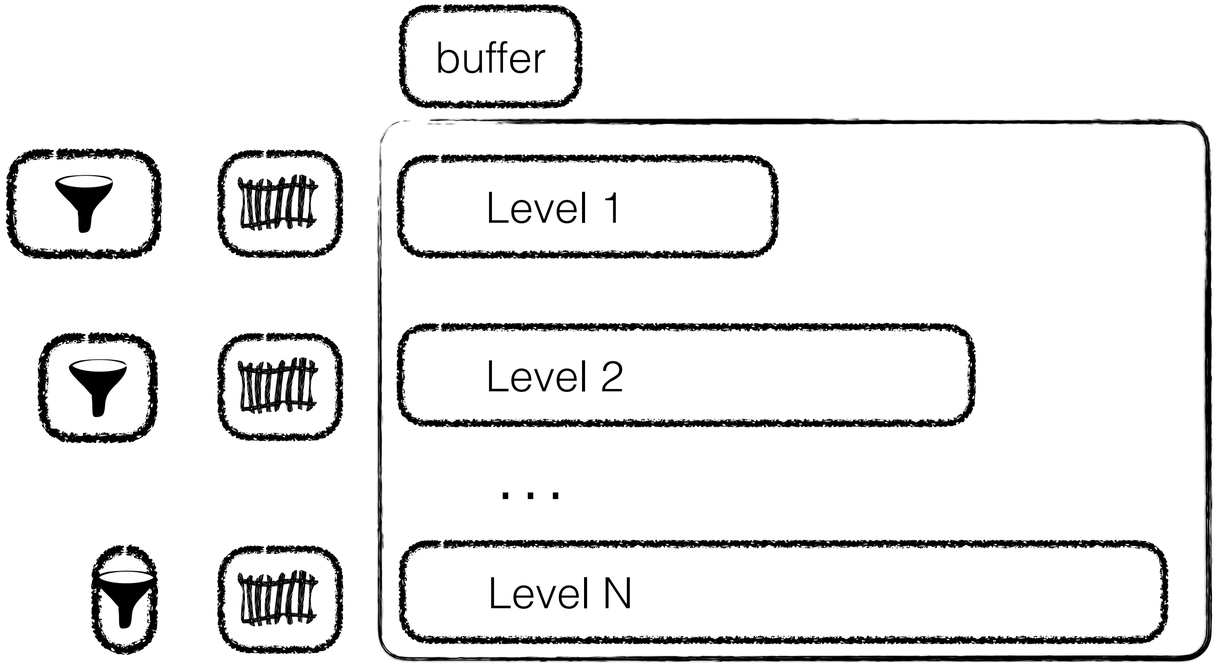
@SIGMOD2018

Dostoevsky: **S**pace-**T**ime **O**ptimized **Ev**olvable **S**calable **K**e**y**-Value Store

MERGE POLICY: SHOULD BE TUNED

@SIGMOD2018

Dostoevsky: **S**pace-**T**ime **O**ptimized **Ev**olvable **S**calable **K**e**y**-Value Store

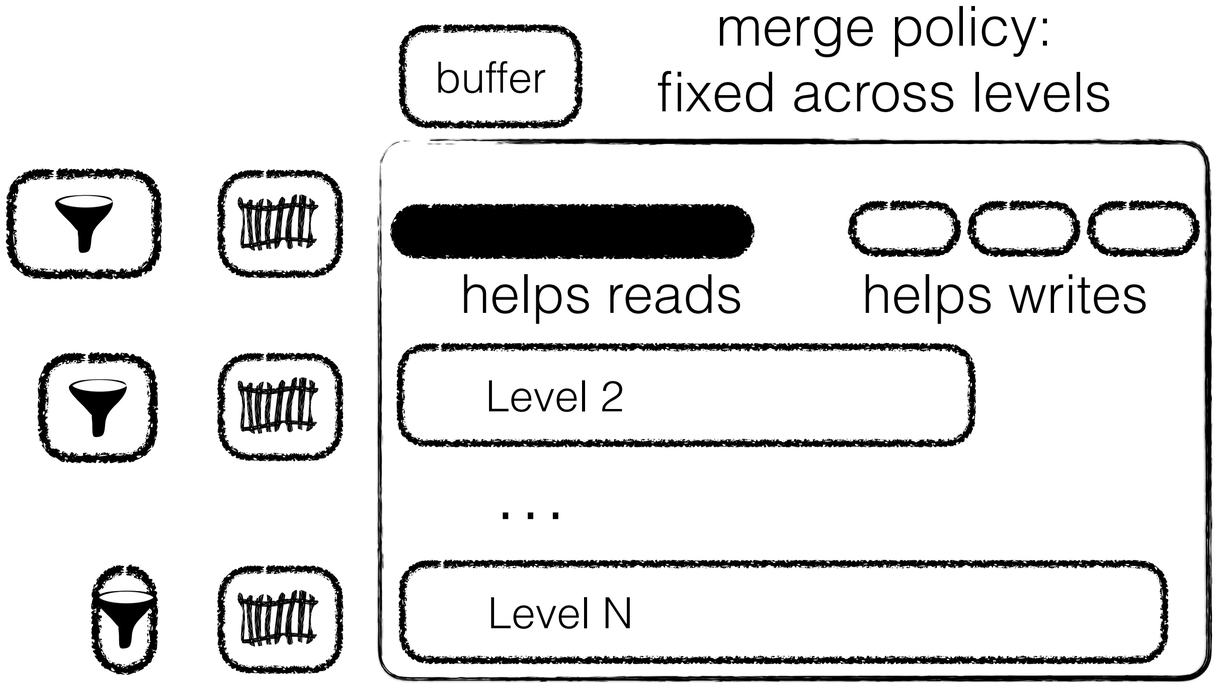


DASlab
@ Harvard SEAS

MERGE POLICY: SHOULD BE TUNED

@SIGMOD2018

Dostoevsky: **S**pace-**T**ime **O**ptimized **Ev**olvable **S**calable **K**e**y**-Value Store

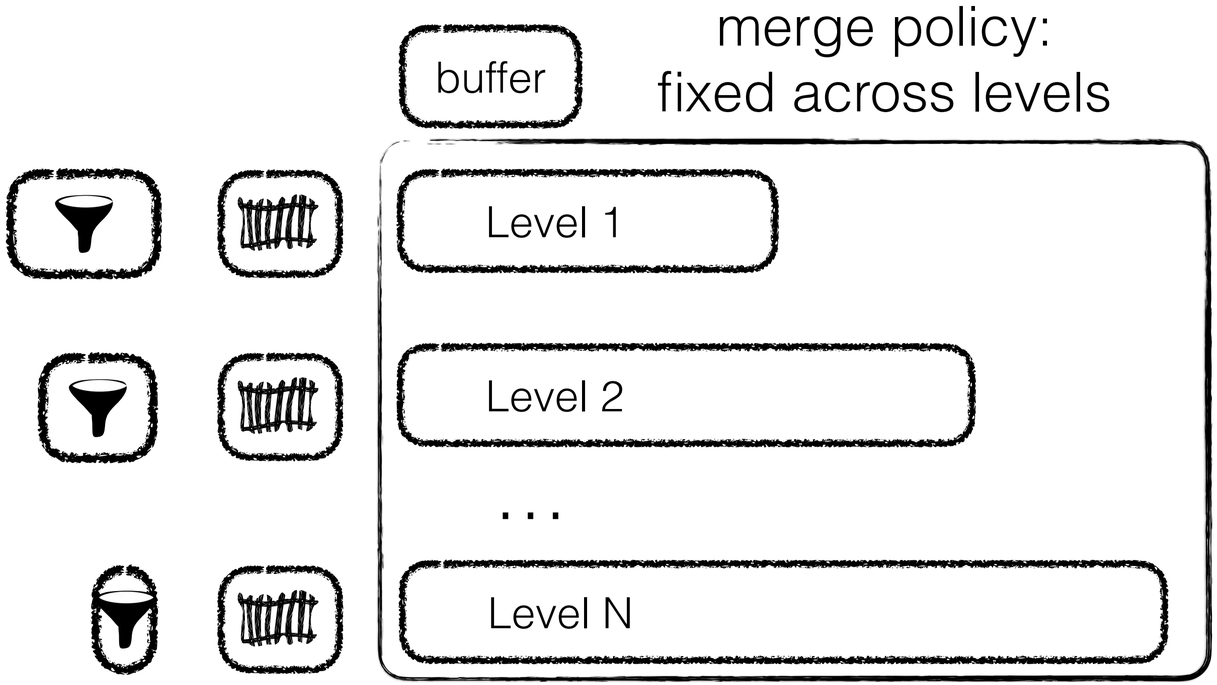


DASlab
@ Harvard SEAS

MERGE POLICY: SHOULD BE TUNED

@SIGMOD2018

Dostoevsky: **S**pace-**T**ime **O**ptimized **Ev**olvable **S**calable **K**e**y**-Value Store



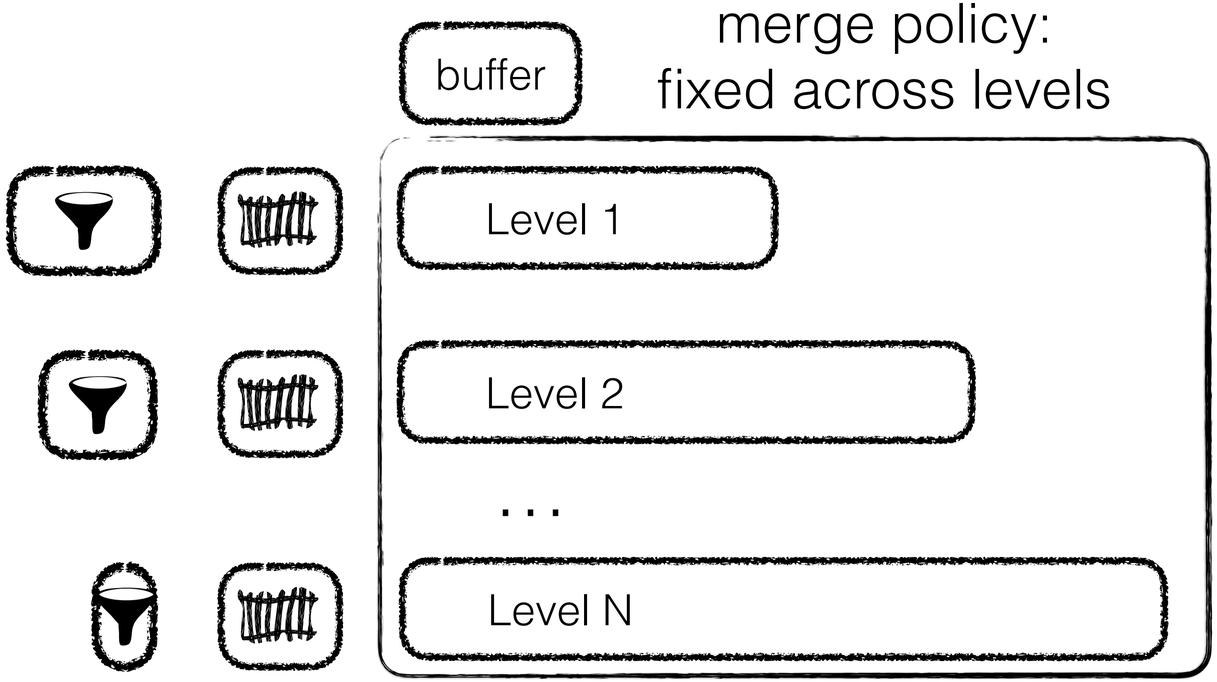
DASlab
@ Harvard SEAS

MERGE POLICY: SHOULD BE TUNED

@SIGMOD2018

Dostoevsky: **S**pace-**T**ime **O**ptimized **Ev**olvable **S**calable **K**e**y**-Value Store

merging small levels does not help that much (point,range,space)

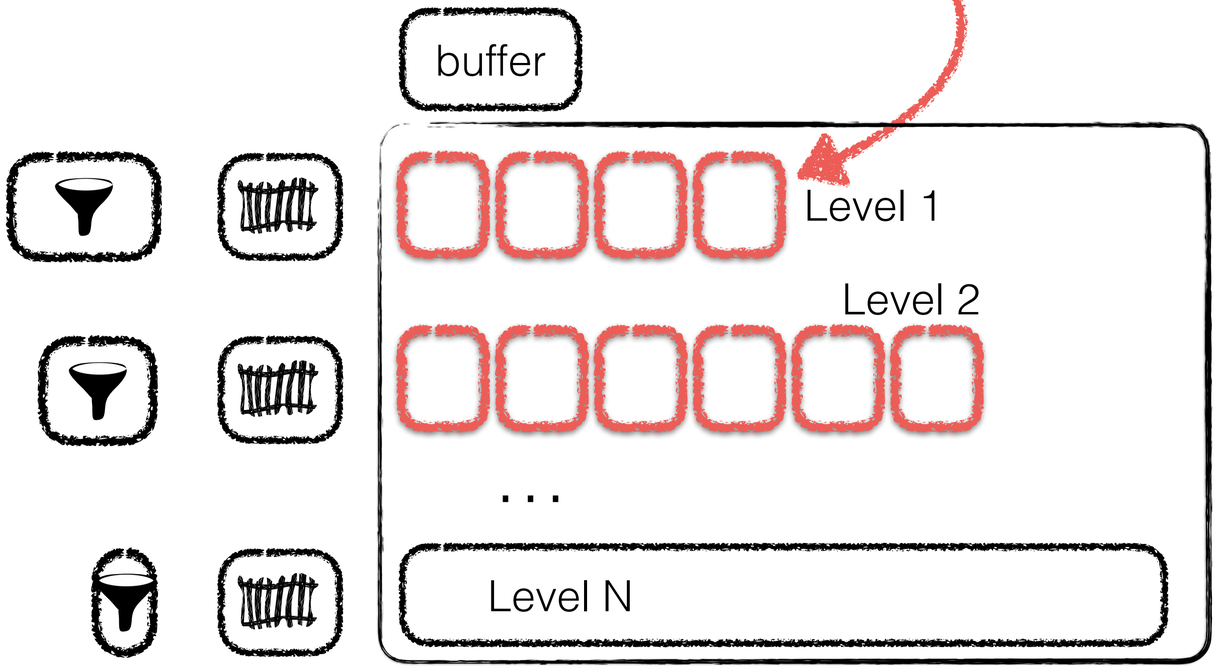


MERGE POLICY: SHOULD BE TUNED

@SIGMOD2018

Dostoevsky: **S**pace-**T**ime **O**ptimized **Ev**olvable **S**calable **K**e**y**-Value Store

merging small levels does not help that much (point,range,space)



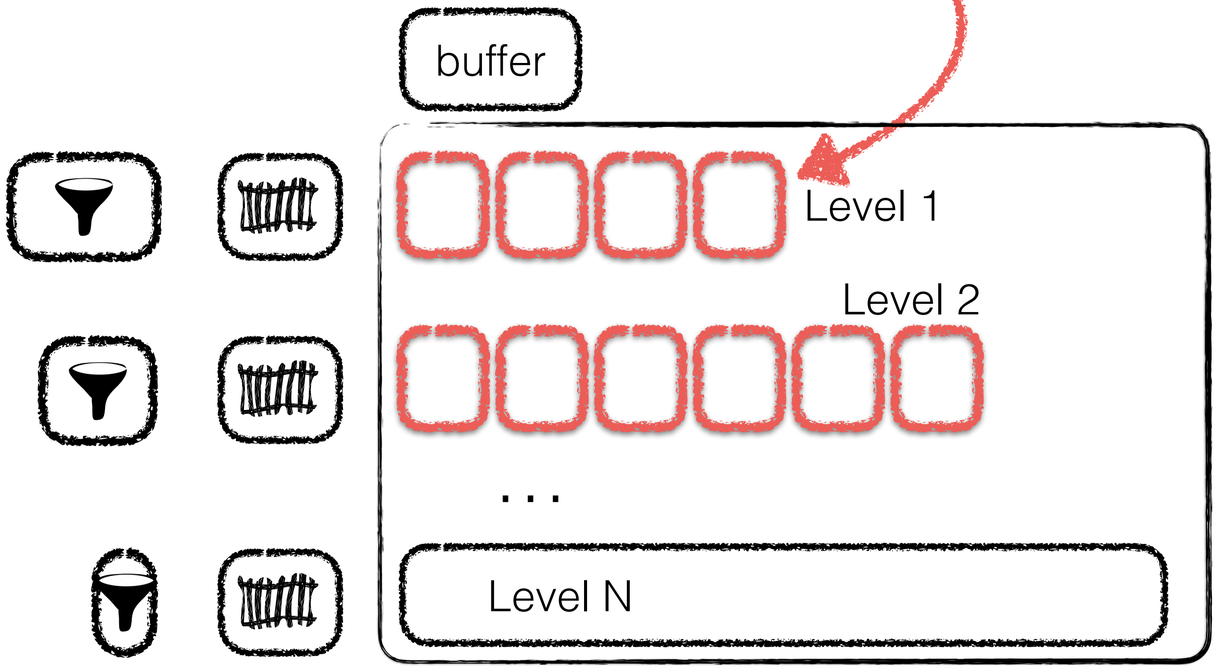
DASlab @ Harvard SEAS

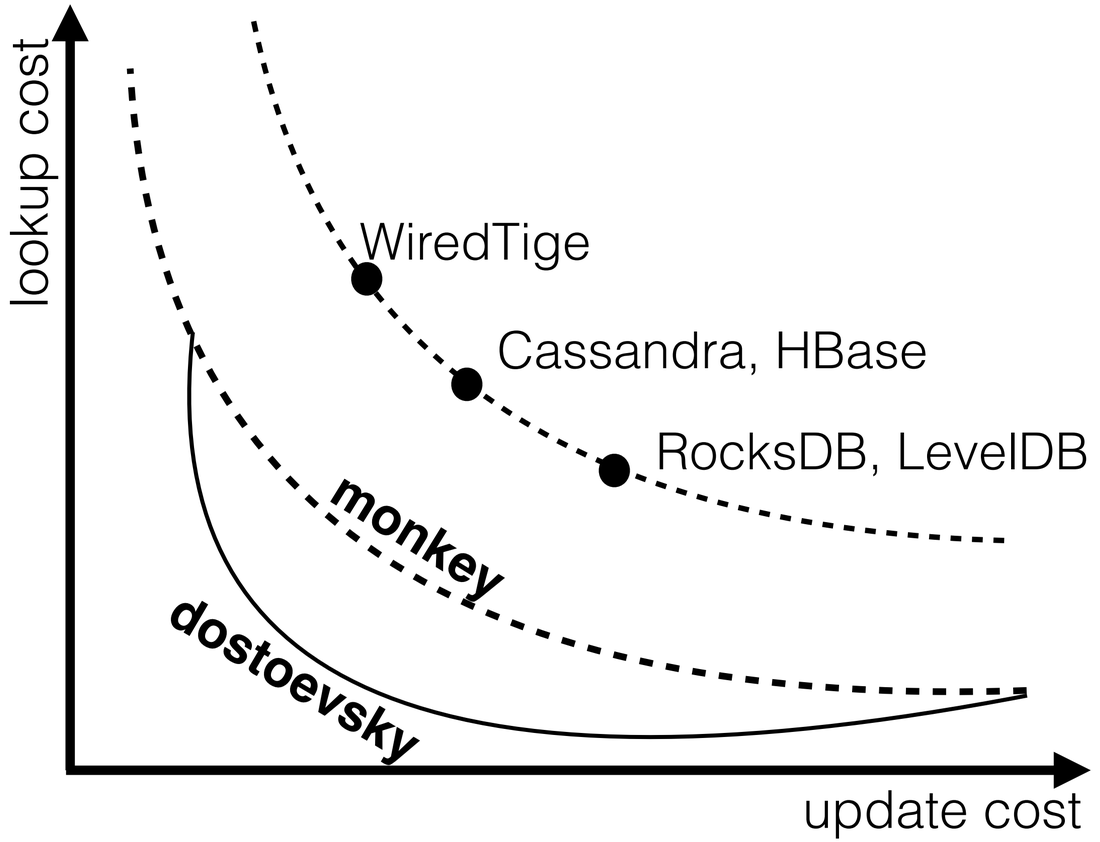
MERGE POLICY: SHOULD BE TUNED

@SIGMOD2018

Dostoevsky: **S**pace-**T**ime **O**ptimized **Ev**olvable **S**calable **K**e**y**-Value Store

merging small levels does not help that much (point,range,space)





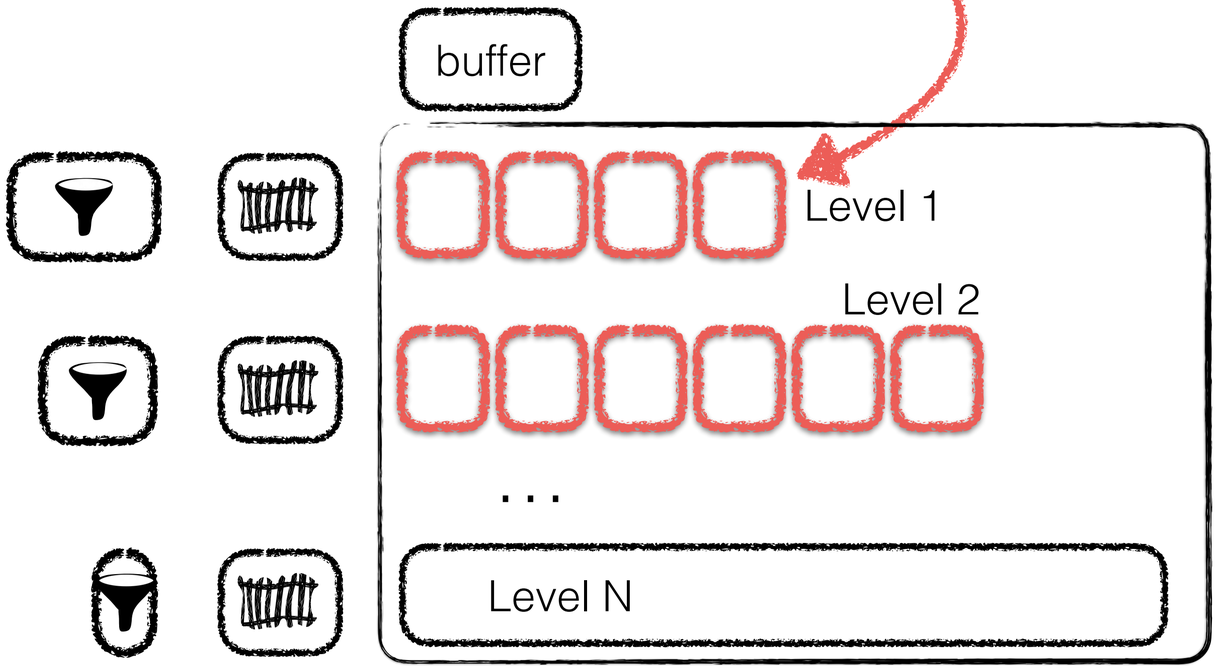
DASlab @ Harvard SEAS

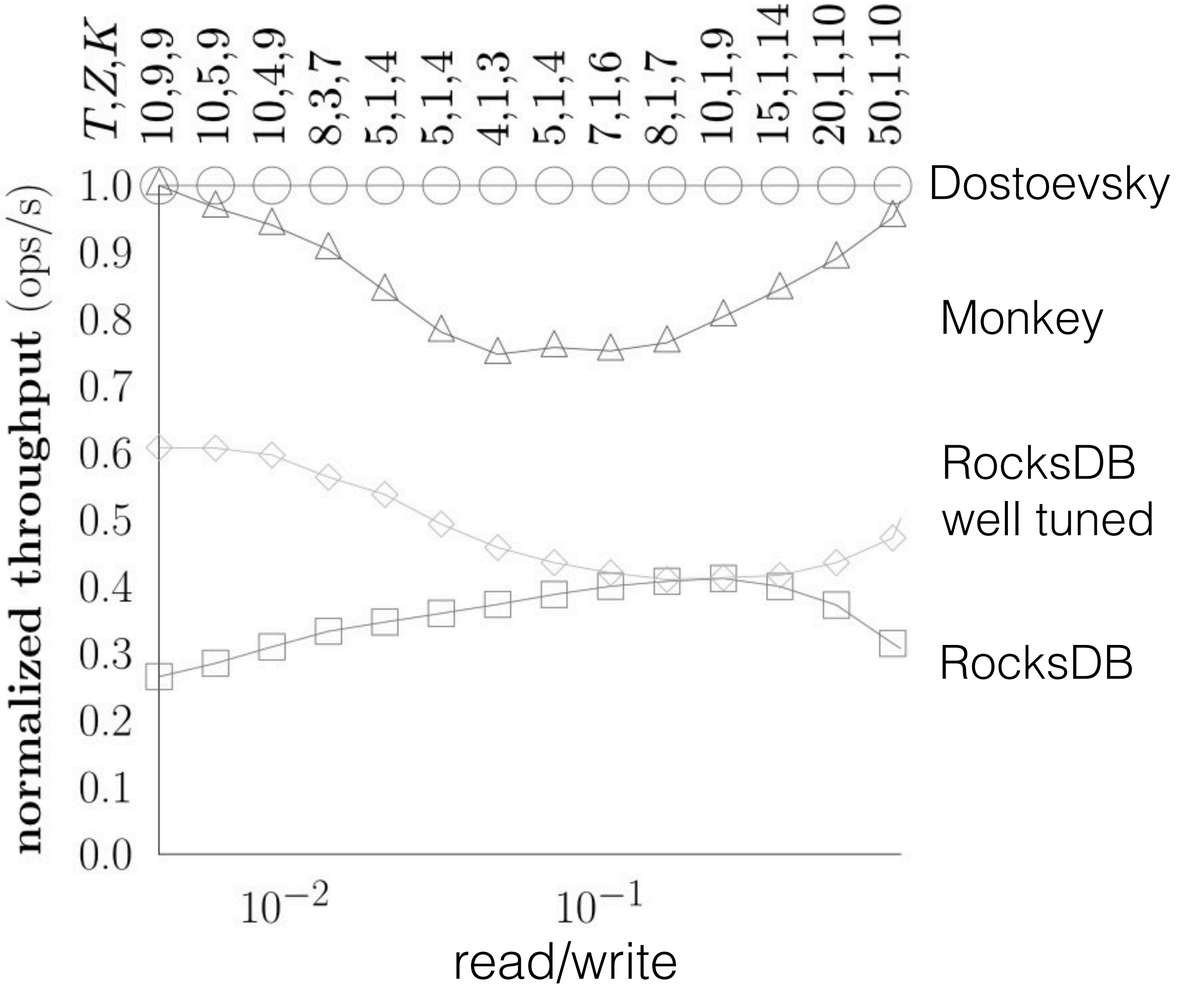
MERGE POLICY: SHOULD BE TUNED

@SIGMOD2018

Dostoevsky: **S**pace-**T**ime **O**ptimized **Ev**olvable **S**calable **K**ey-**V**alue Store

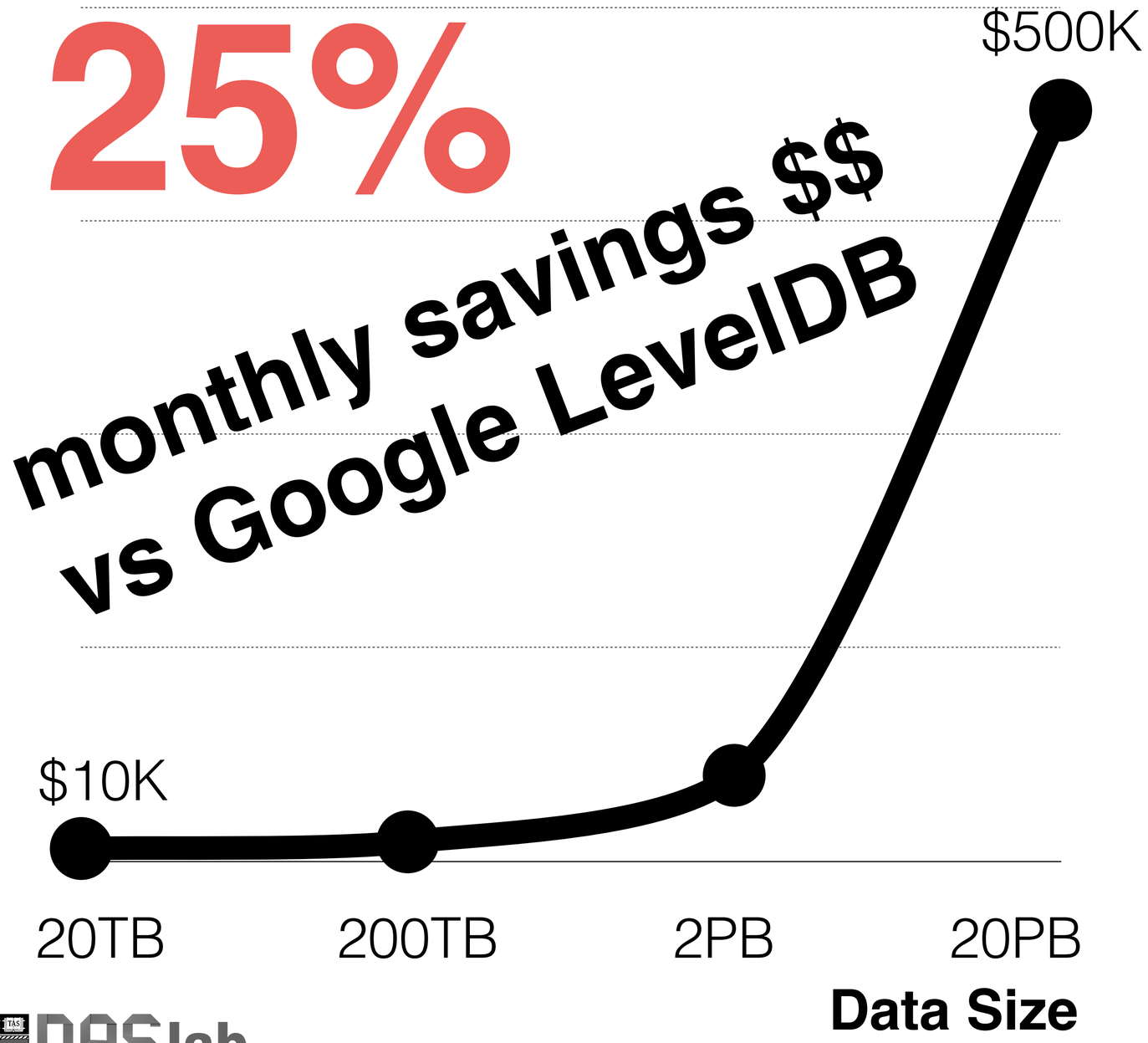
merging small levels does not help that much (point,range,space)





DASlab @ Harvard SEAS

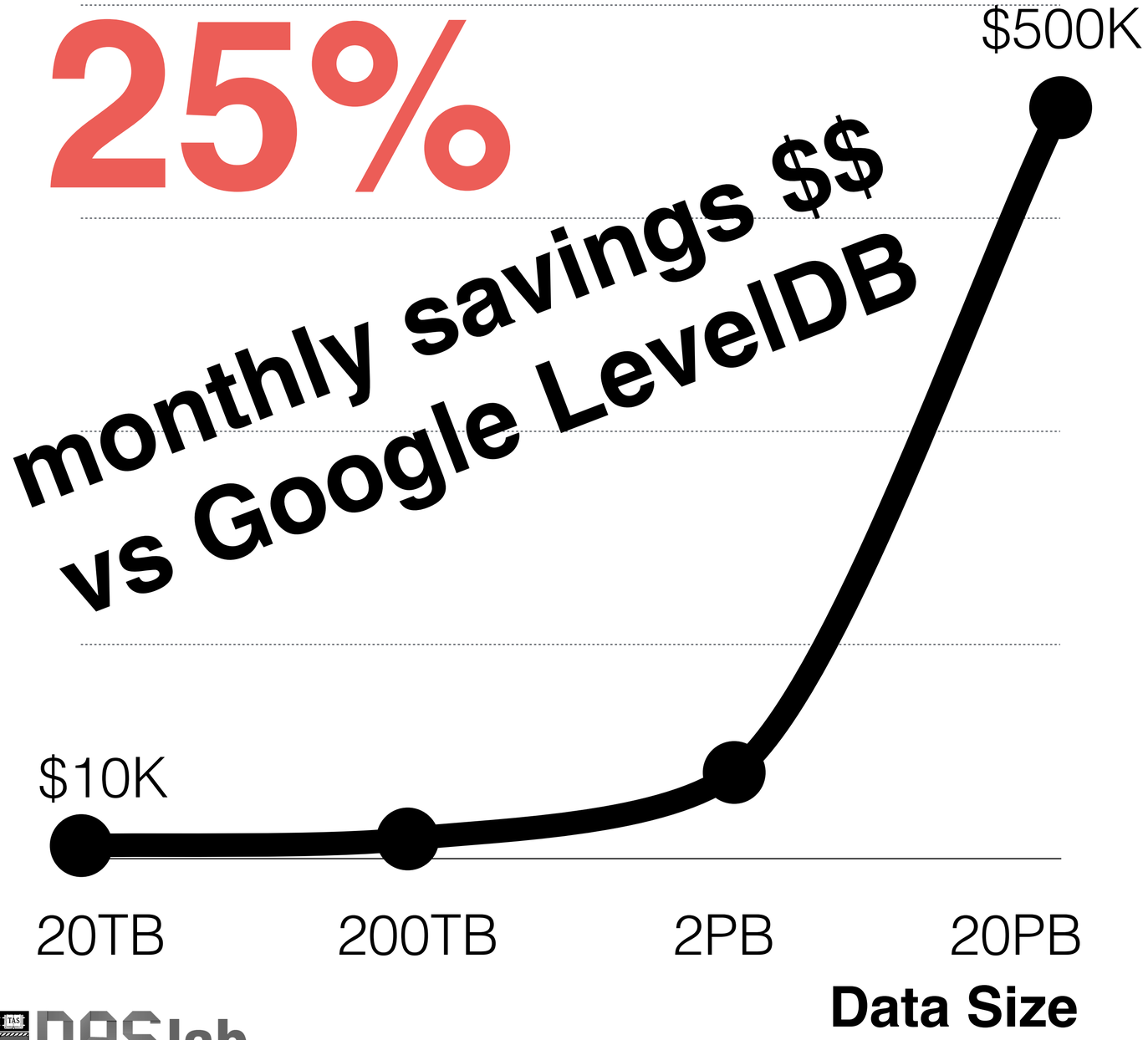
25%

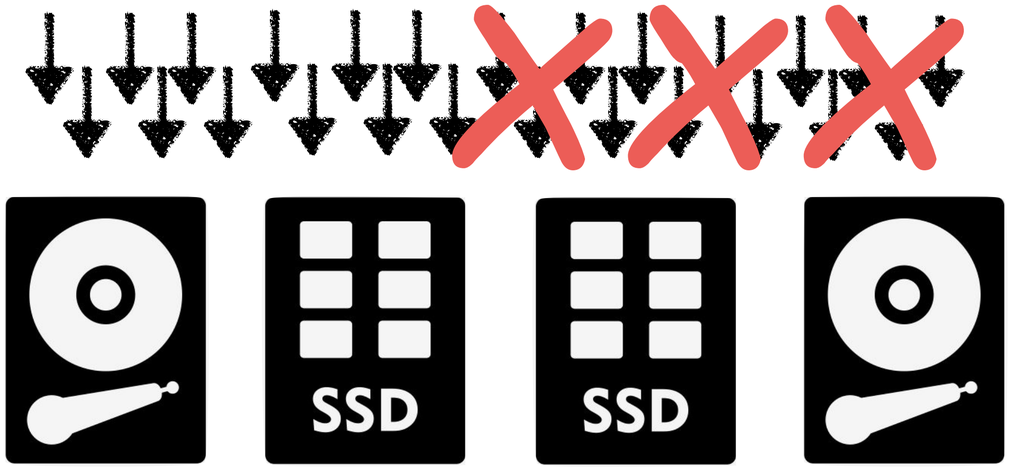


DASlab
@ Harvard SEAS

Amazon Cloud (North America)

25%



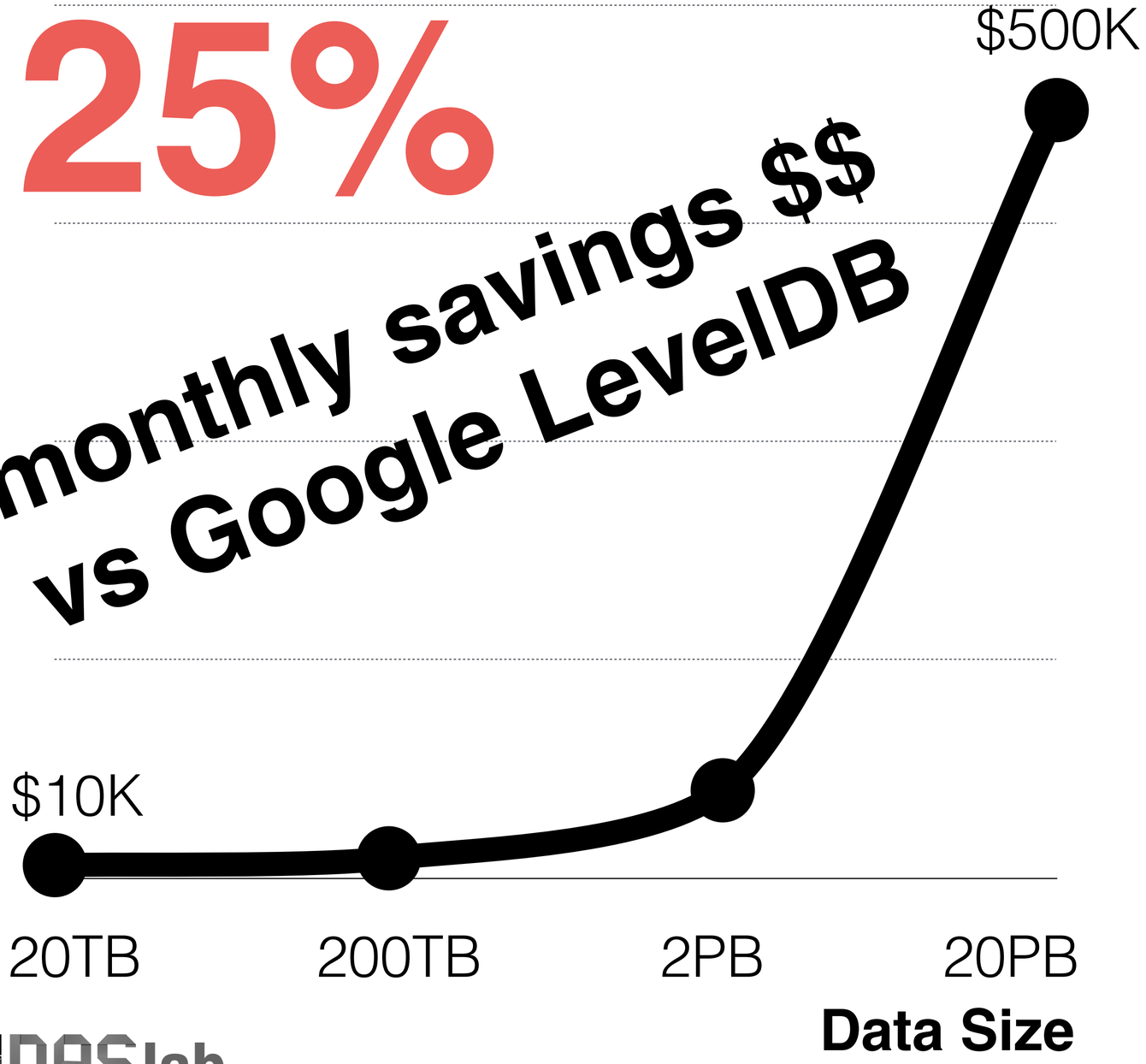


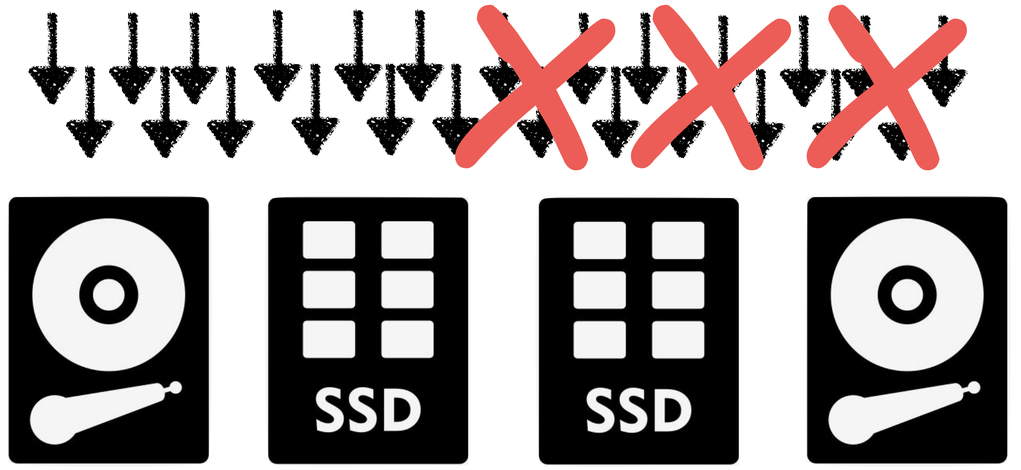
DASlab
@ Harvard SEAS

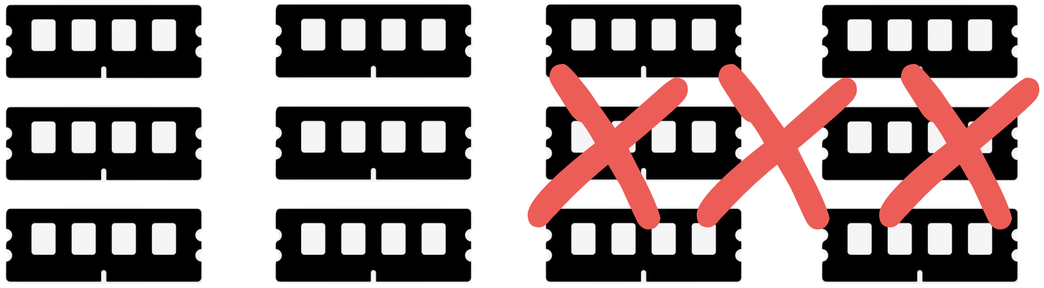
Amazon Cloud (North America)

25%

monthly savings $$
vs Google LevelDB







DASlab
@ Harvard SEAS

Amazon Cloud (North America)

Summary

Once you understand the design, you can think of new ideas.
Just keep asking “why”.

Summary

Once you understand the design, you can think of new ideas.
Just keep asking “why”.

Tons of opportunities in big data as everything is new and changing.

Summary

Once you understand the design, you can think of new ideas.
Just keep asking “why”.

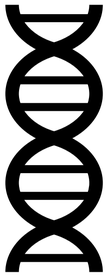
Tons of opportunities in big data as everything is new and changing.

Once you think of a new idea, then it is just about following good research practices = requires technical skills but easier (just following steps).

DESIGN SPACE

COST SYNTHESIS

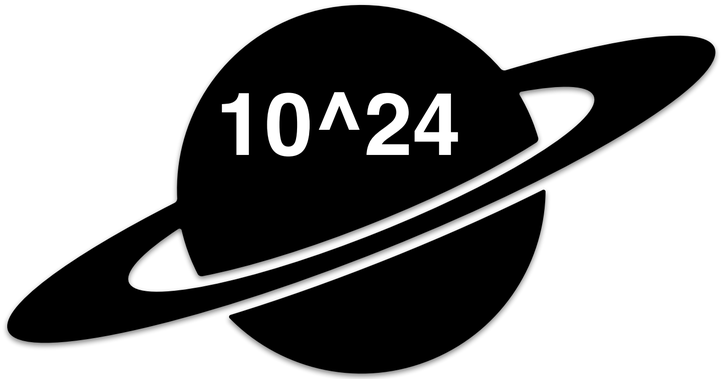
WHAT-IF



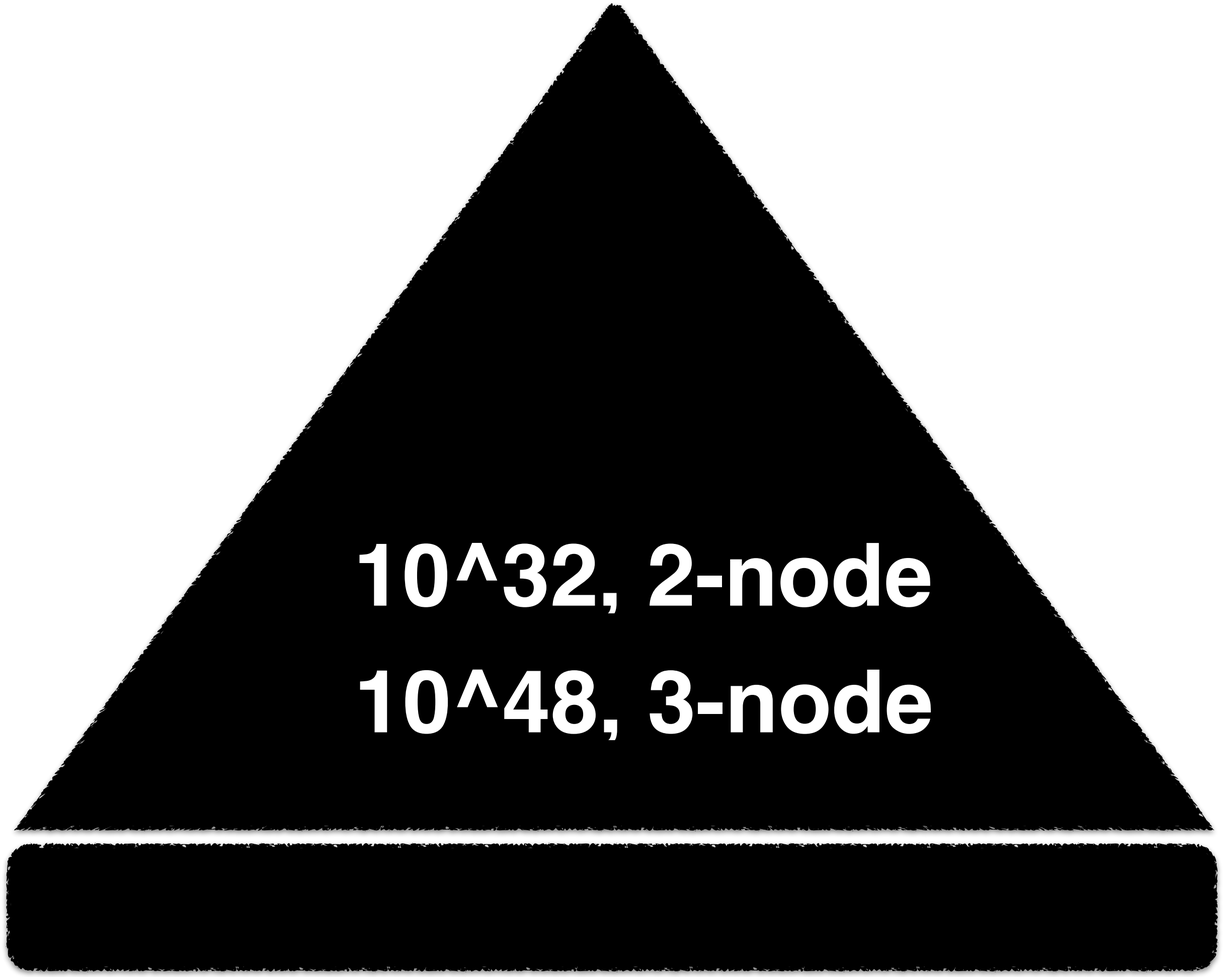
DESIGN SPACE

COST SYNTHESIS

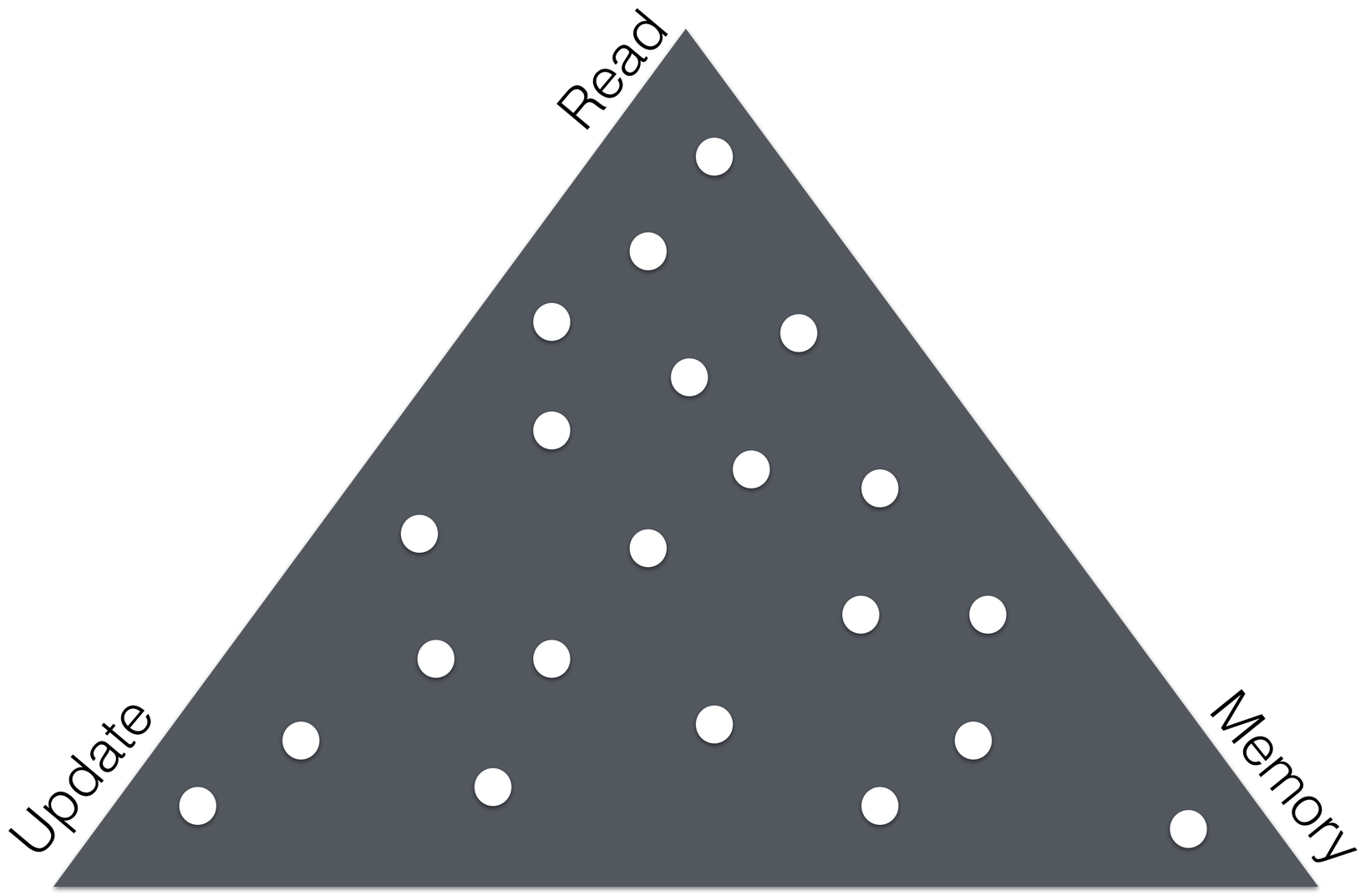
WHAT-IF

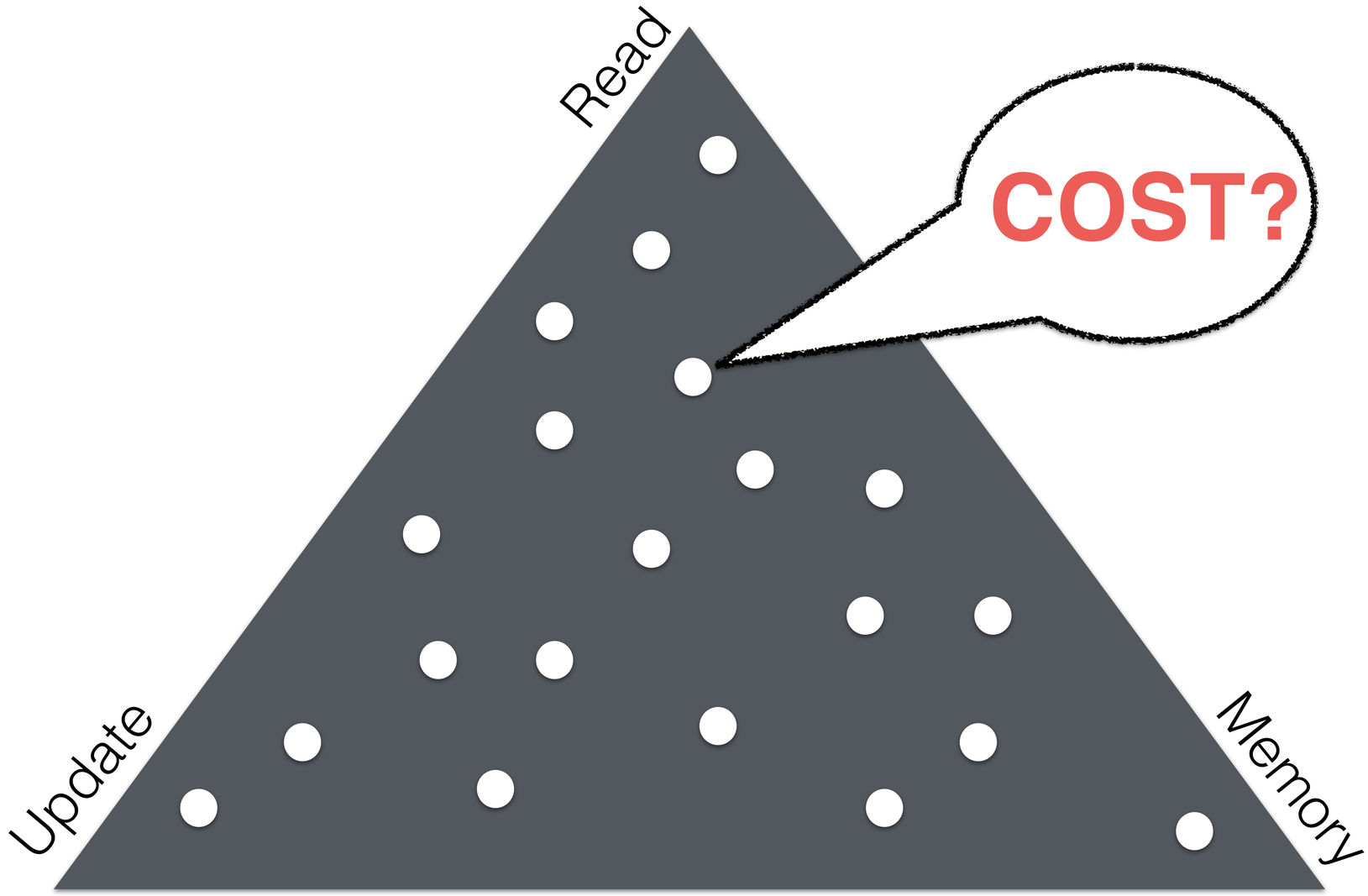


STARS IN THE SKY

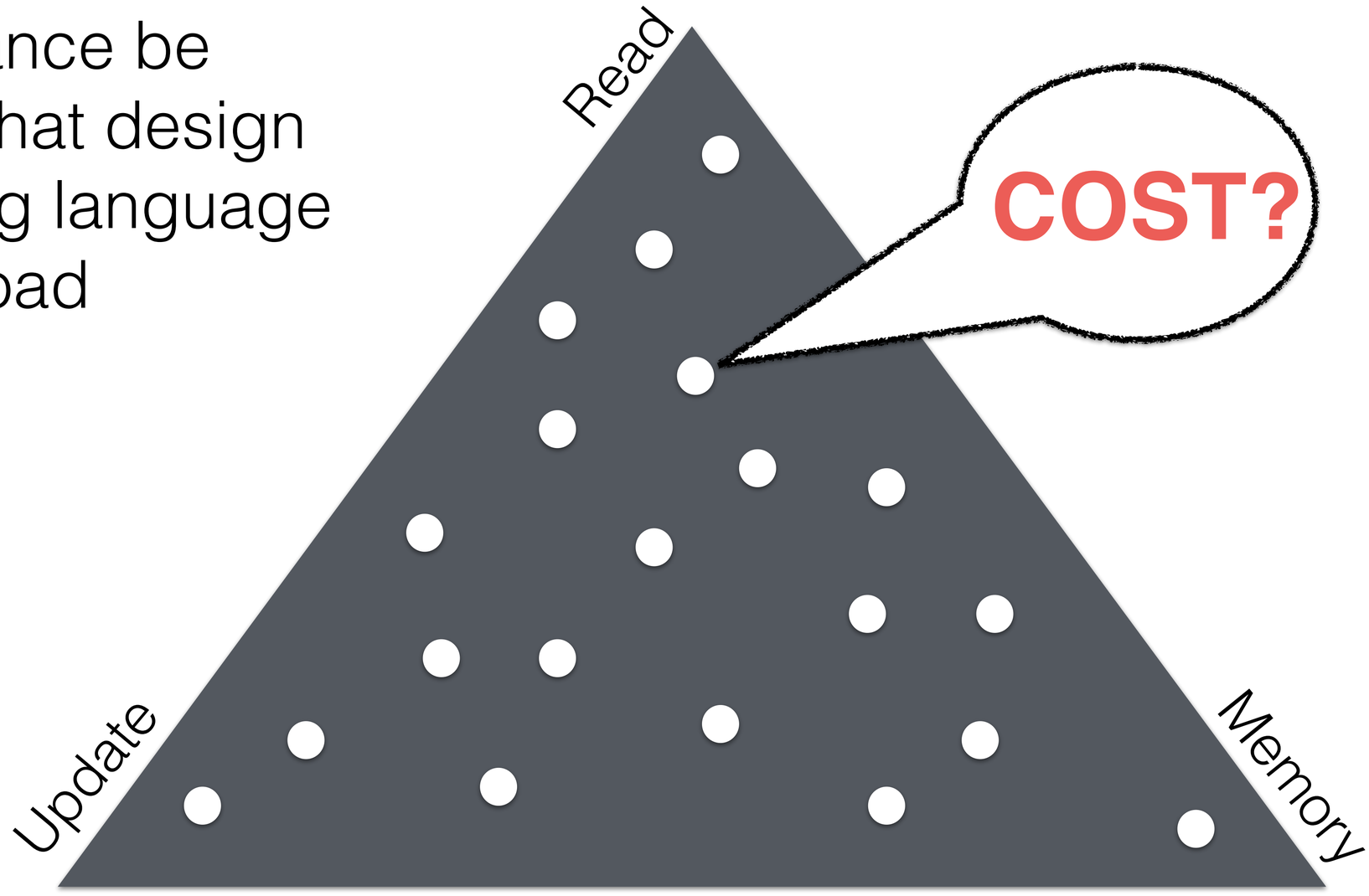


POSSIBLE DATA STRUCTURES


Read
Update
Memory

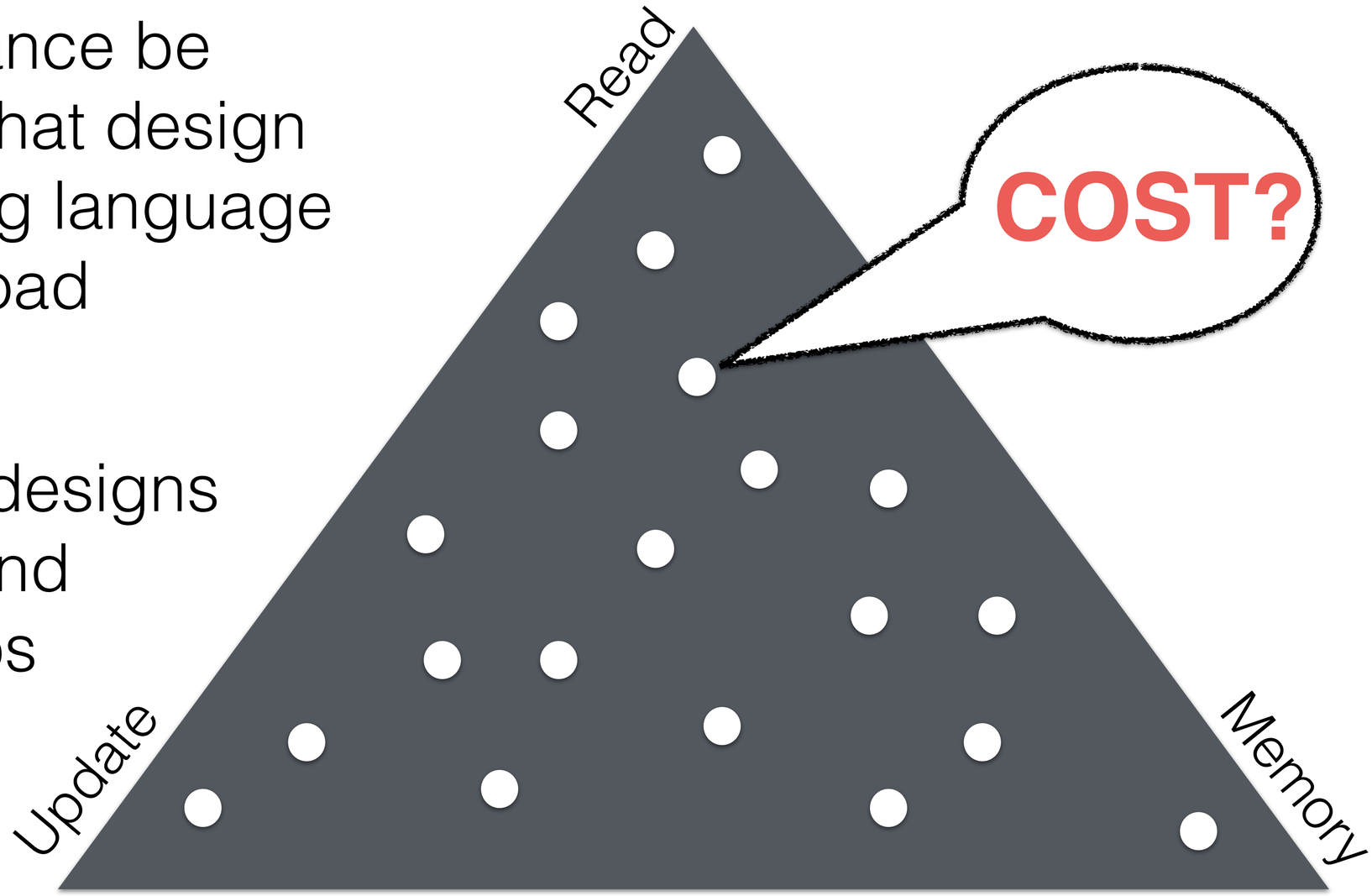

Read
COST?
Update
Memory

What would the performance be
if we were to implement that design
in a specific programming language
and test a specific workload
on a specific hardware



What would the performance be
if we were to implement that design
in a specific programming language
and test a specific workload
on a specific hardware

If we have the cost for 2 designs
we can compare them, and
we can build search algos



CS 265

Stratos Idreos

BIG DATA SYSTEMS

NoSQL | Neural Networks | Image AI | LLMs | Data Science