Probabilistic data structures in Adversarial Settings

Mia Filić ETH Zürich

Probabilistic data structures in Adversarial Settings

Based on joint work with

Anu Unnikrishnan

Kenny Paterson
ETH Zürich

Fernando Virdia
Universidade Nova de Lisboa

Sam A. Markelon
University of Florida

Thomas Shrimpton
University of Florida

Jonas Hofmann
Ella Kummer
Keran Kocher
Andrea Raguso

Probabilistic Data Structures (PDS)

A way to

compactly represent (stream of) data

and

provide approximate **answers to queries** about the data

Probabilistic Data Structures (PDS)

A way to

compactly represent (stream of) data

and

provide approximate **answers to queries** about the data

- <u>Frequency estimation</u>
  How many times does x appear in the set?
  Count-min sketch, HeavyKeeper

Probabilistic Data Structures (PDS)

A way to

compactly represent (stream of) data

and

provide approximate **answers to queries** about the data

- Frequency estimation
  How many times does x appear in the set?
  Count-min sketch, HeavyKeeper

- Membership queries
  Is x in the set?
  Bloom filter, Cuckoo filter

Probabilistic Data Structures (PDS)

A way to

compactly represent (stream of) data

and

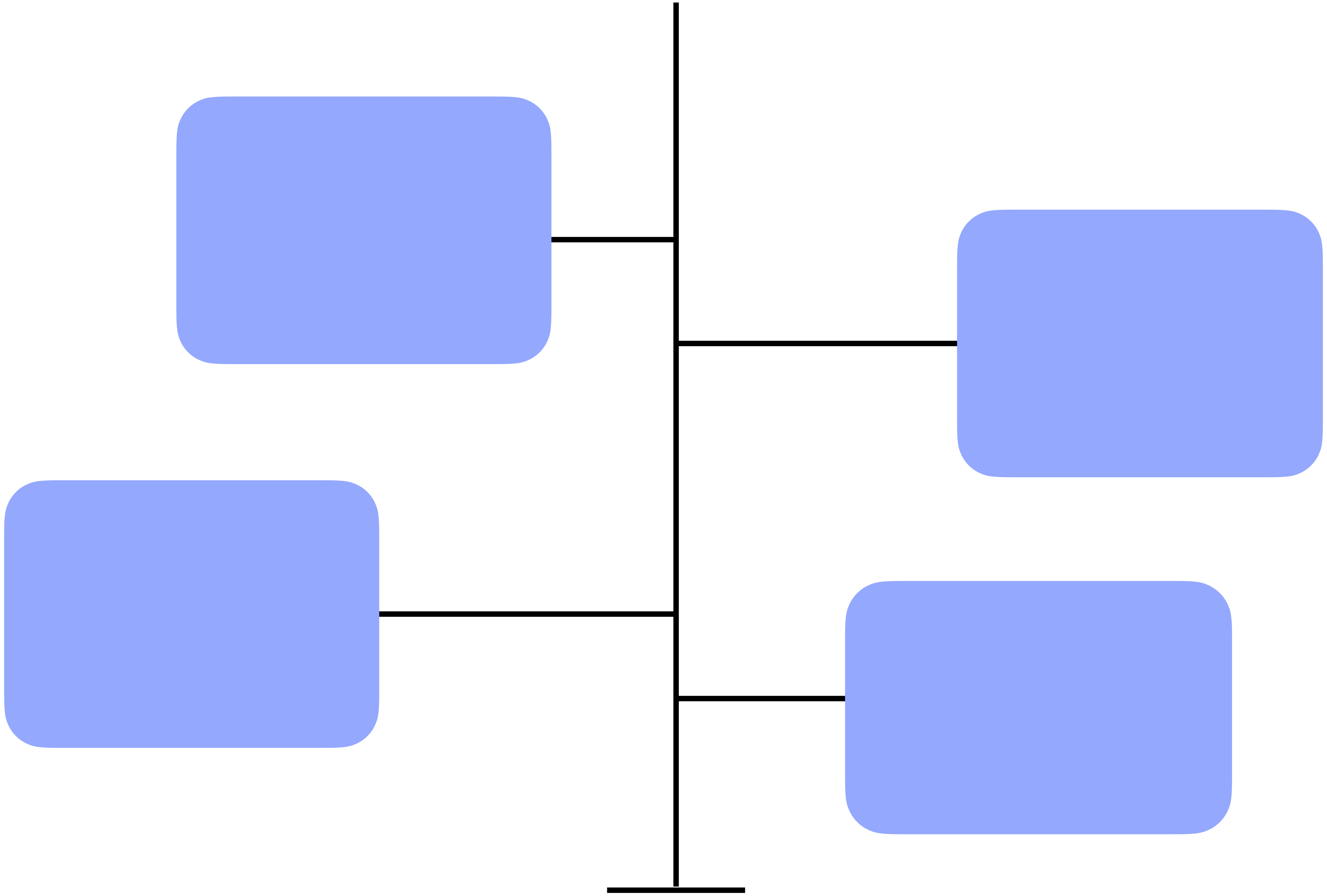
provide approximate **answers to queries** about the data

- Frequency estimation
  How many times does x appear in the set?
  Count-min sketch, HeavyKeeper

- Membership queries
  Is x in the set?
  Bloom filter, Cuckoo filter

- Cardinality estimation
  How many distinct elements in the set?
  HyperLogLog, KMV estimator

PDS help us



PDS help us

Count-min sketch

find the most visited pages on a website

Count-min sketch

identify possible DoS threats (network-monitoring systems)

PDS help us

Count-min sketch

find the most visited pages on a website

Count-min sketch

identify possible DoS threats (network-monitoring systems)

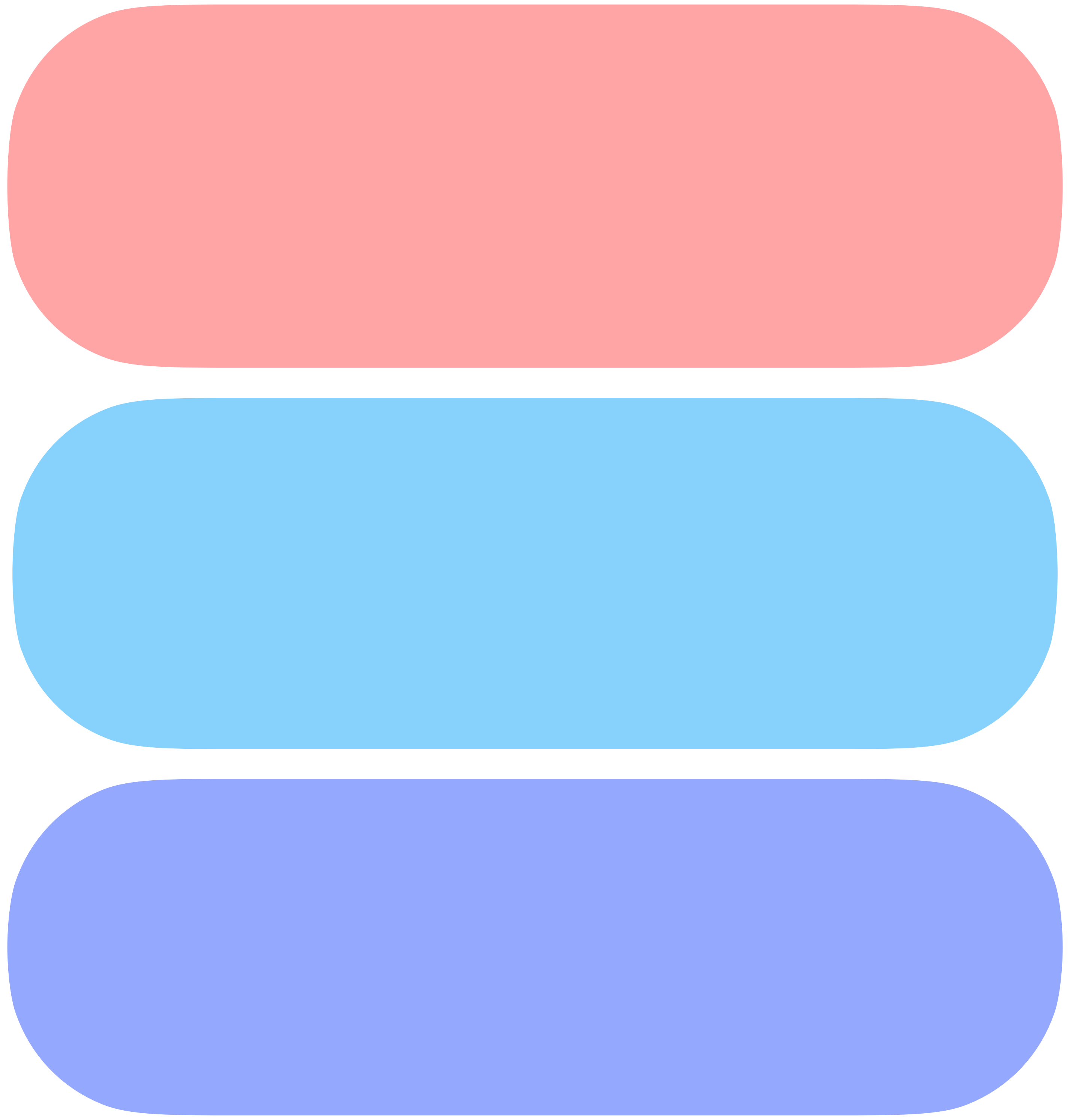
Bloom filter cascade

check revoked certificates in TLS/SSL

HyperLogLog

count the number of distinct Facebook users

PDS in adversarial settings



PDS in adversarial settings

Adversarial correctness

- How can an adversary **interfere** with the correct functionality of the PDS?

PDS in adversarial settings

Adversarial correctness

Privacy

- How can an adversary **interfere** with the correct functionality of the PDS?
- What can an adversary **learn** about the elements stored in the PDS?

PDS in adversarial settings

Adversarial correctness	How can an adversary **interfere** with the correct functionality of the PDS?
Privacy	What can an adversary **learn** about the elements stored in the PDS?
Secure PDS	How can we **provably protect** PDS in adversarial settings?

Our work

- Approximate Membership Query PDS (w/o and w/ deletions)

  Adversarial correcness **Privacy** **Provable security**

Our work

- Approximate Membership Query PDS (w/o and w/ deletions)

  Adversarial correcness   Privacy   Provable security

- Compact Frequency Estimation (CFE) PDS

  Adversarial correcness

  Attacks against
  CMS and HeavyKeeper

  Exploration of a
  more robust CFE PDS

Our work

- Approximate Membership Query PDS (w/o and w/ deletions)

  Adversarial correcness **Privacy** **Provable security**

- Compact Frequency Estimation (CFE) PDS

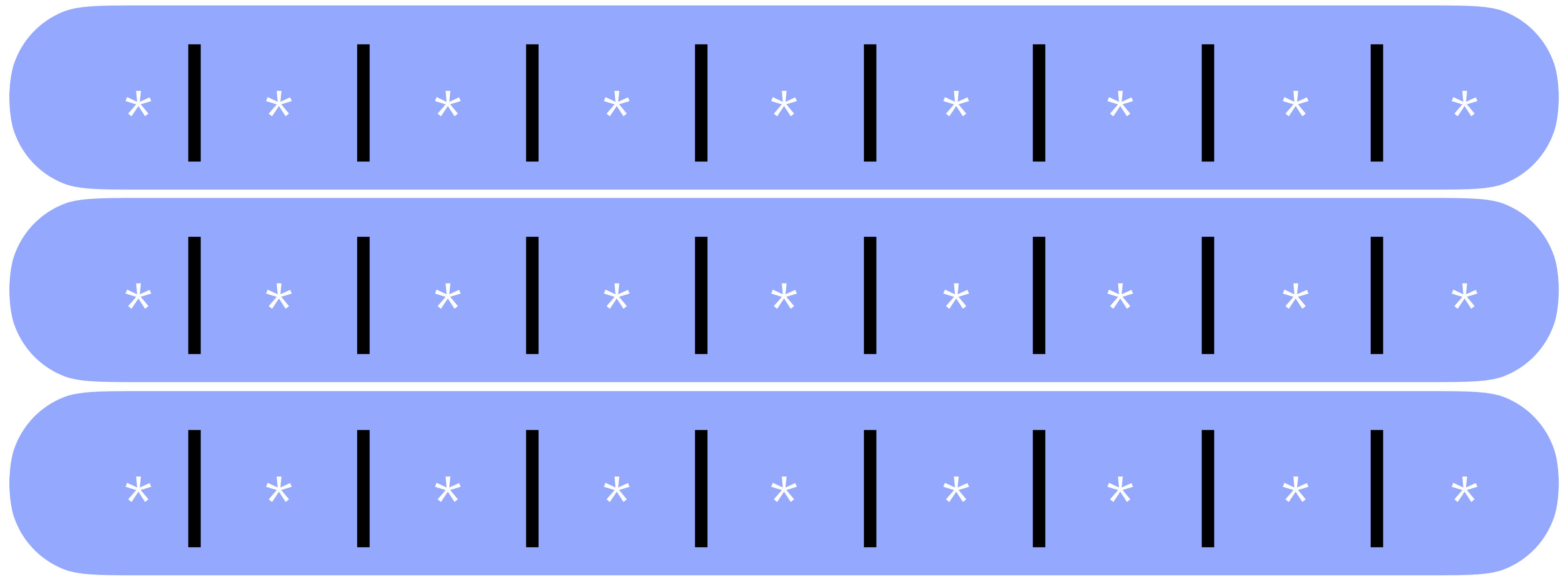
  Adversarial correcness

- Practical implementation

  Adversarial correcness

Compact Frequency Estimation (CFE) PDS

Compact Frequency Estimation (CFE) PDS

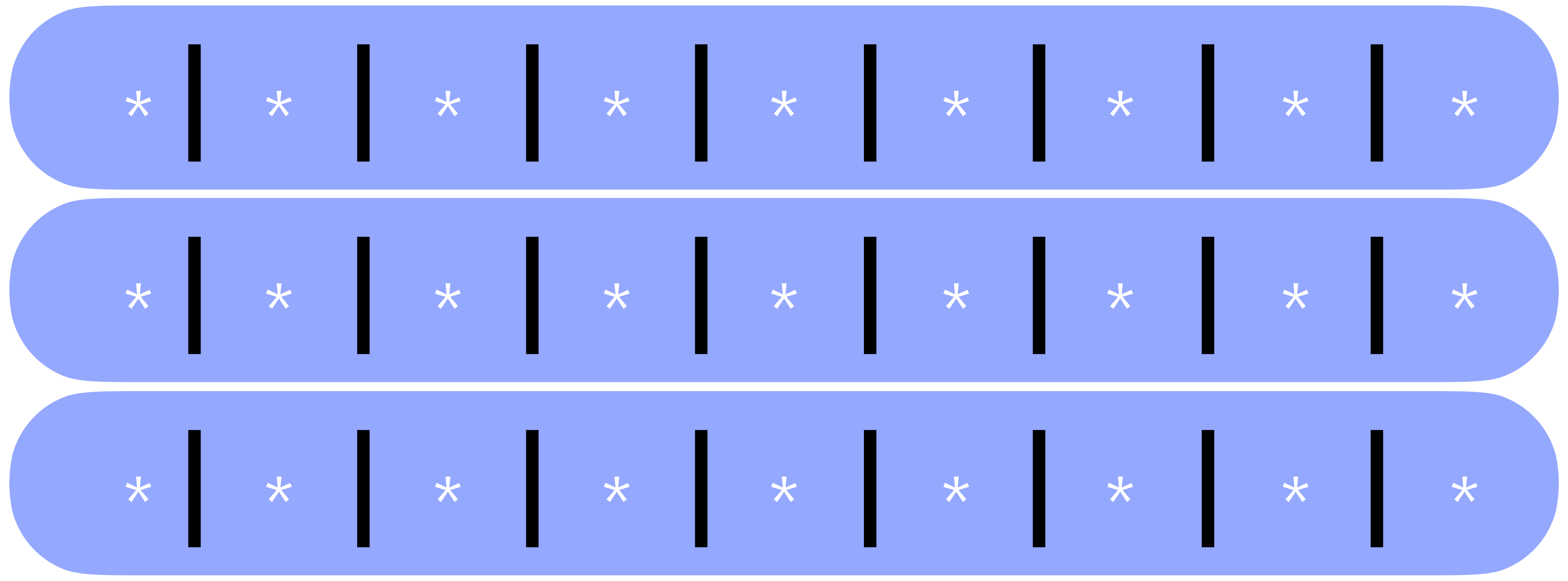


CFE PDS

Stream

n,z,r,p,t,w,l,l,n,s,k,g,o,i,w,…

Compact Frequency Estimation (CFE) PDS

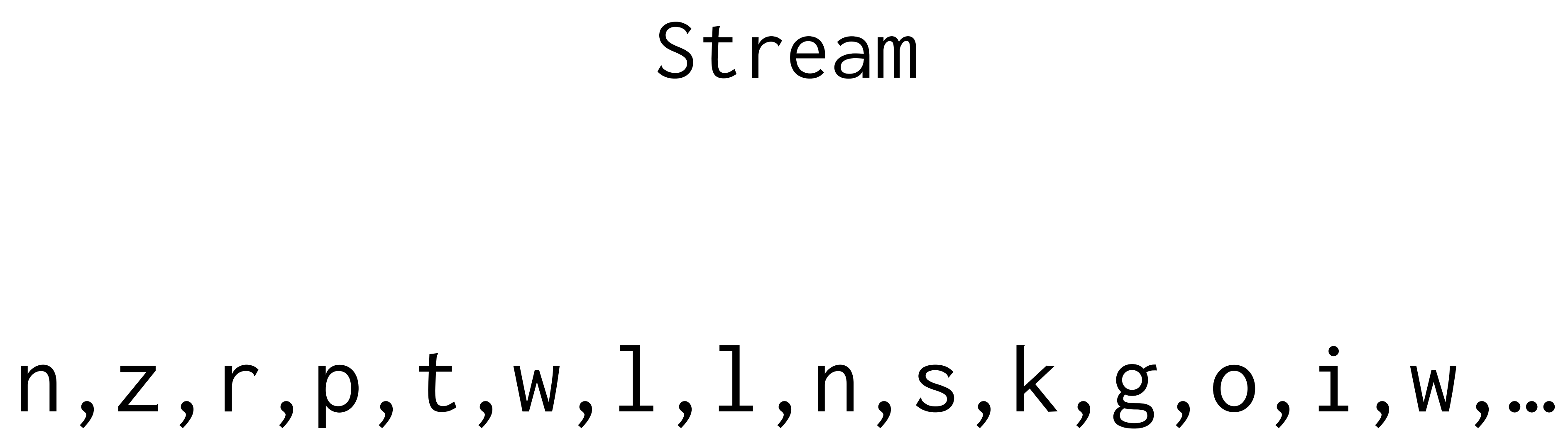


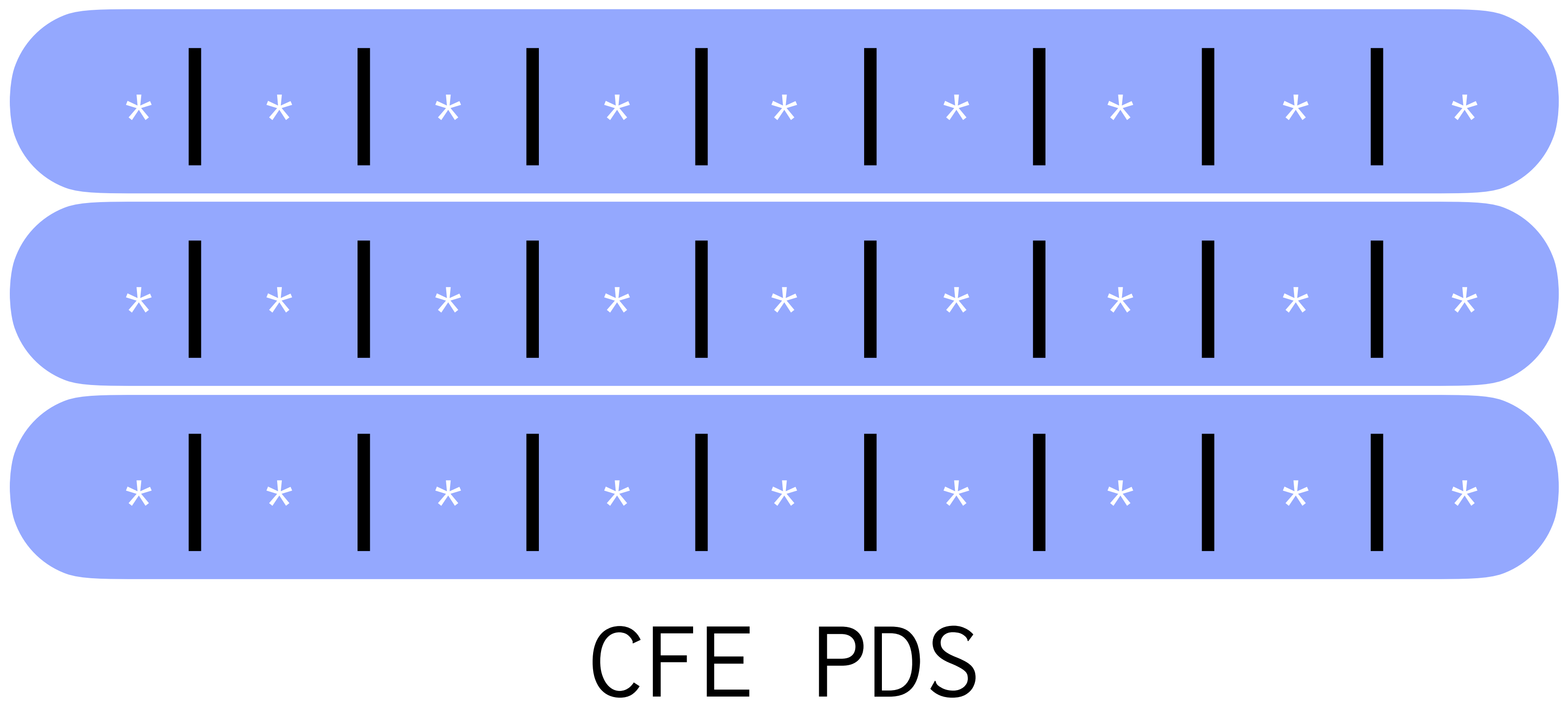
CFE PDS

Stream

n,z,r,p,t,w,l,l,n,s,k,g,o,i,w,…

Compact Frequency Estimation (CFE) PDS

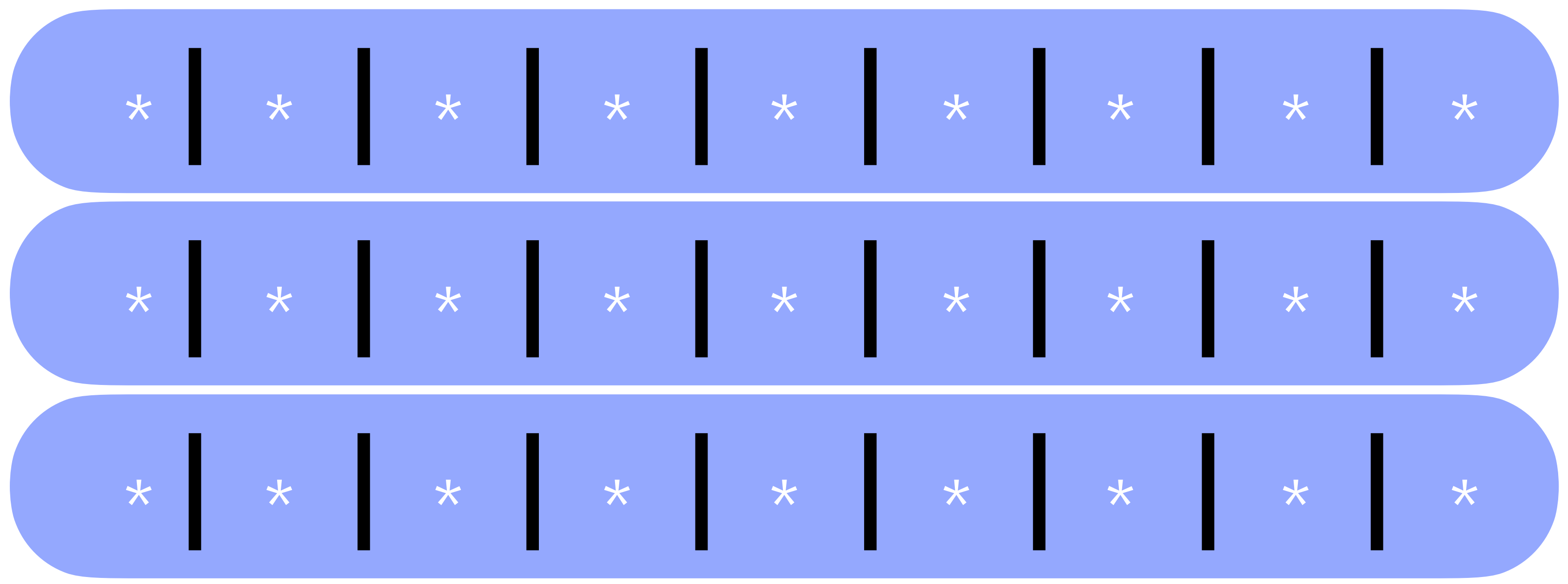




Can CFE PDS misbehave?

Stream

n,z,r,p,t,w,l,l,n,s,k,g,o,i,w,…

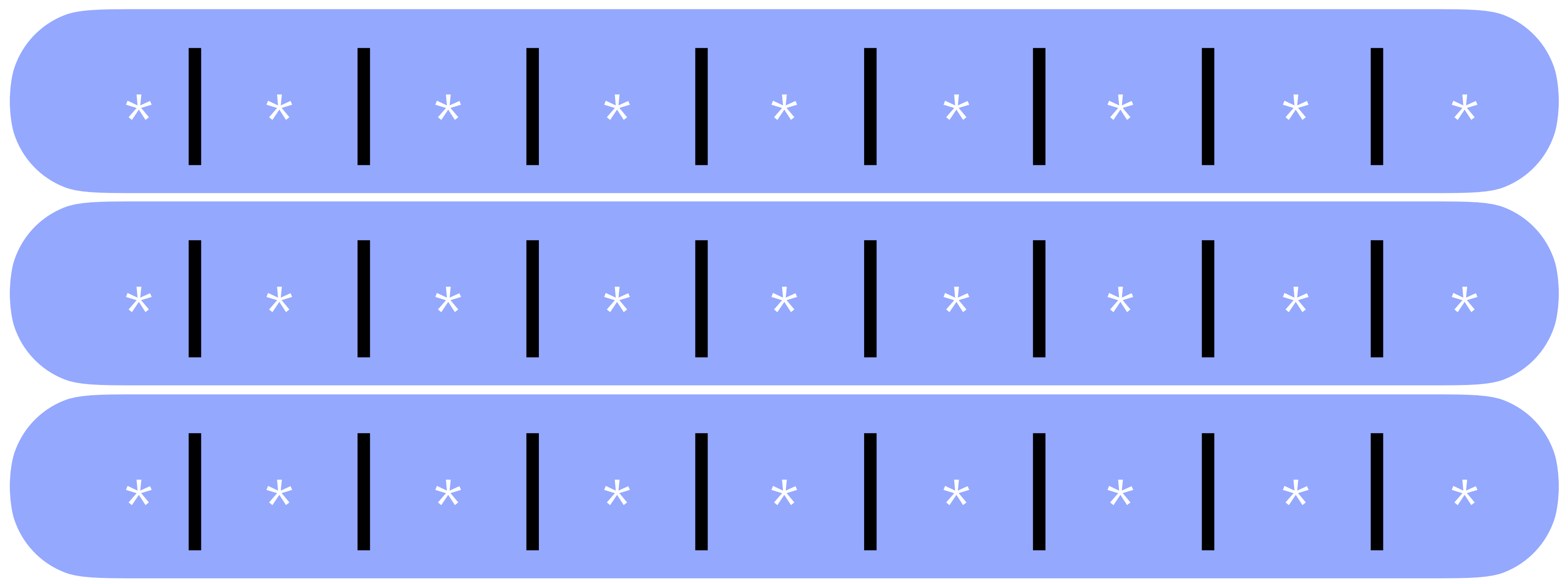


CFE PDS

Can CFE PDS misbehave?

Stream

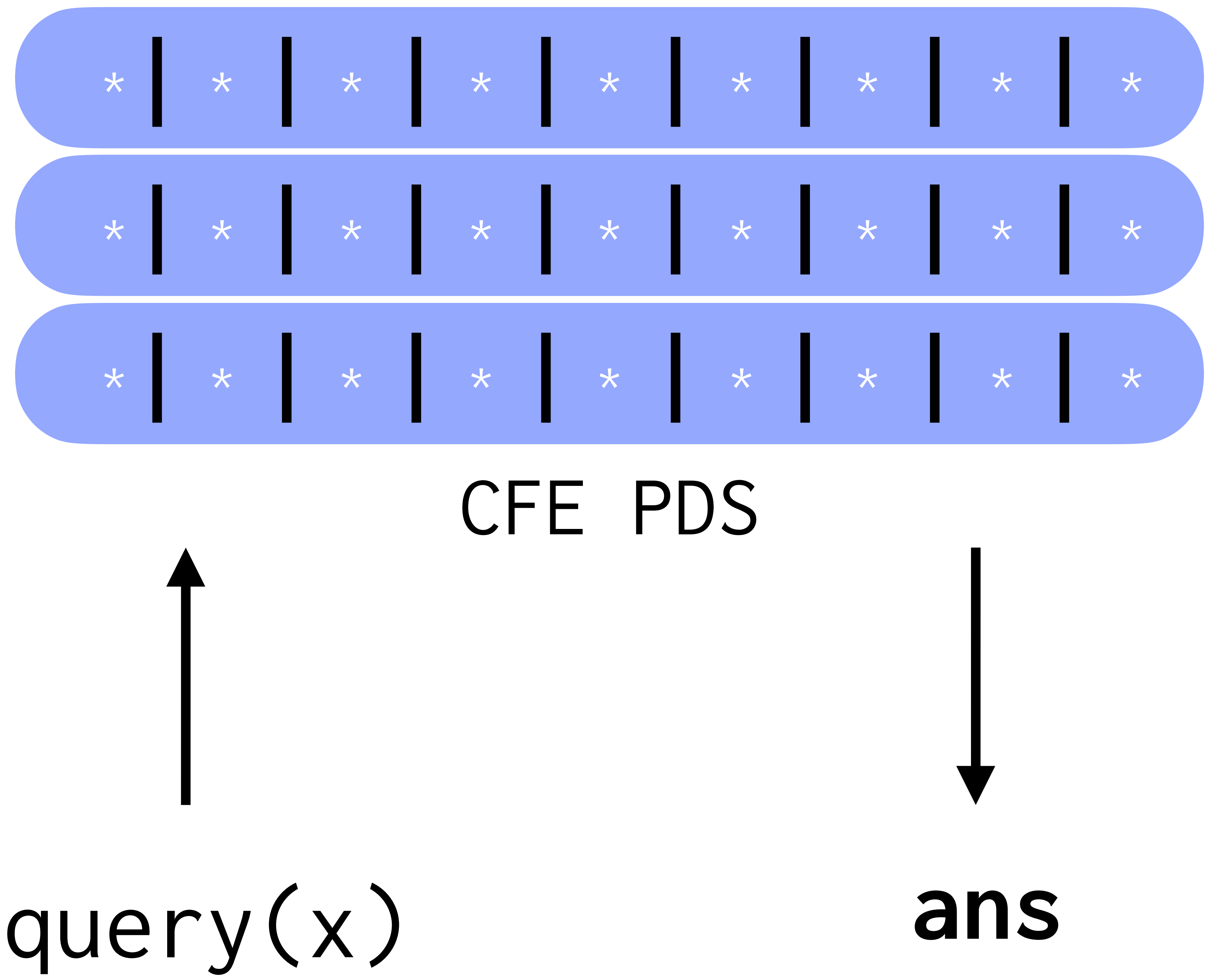
n,z,r,p,t,w,l,l,n,s,k



CFE PDS

query(x)

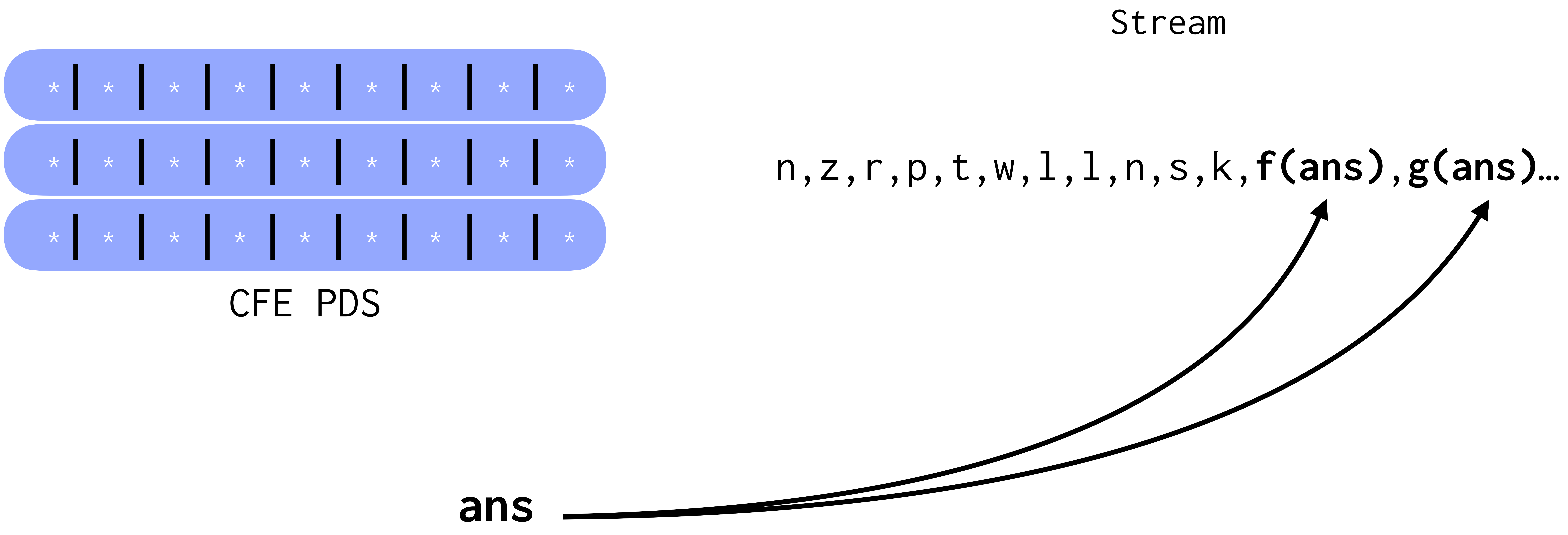
Can CFE PDS misbehave?



Stream

n,z,r,p,t,w,l,l,n,s,k

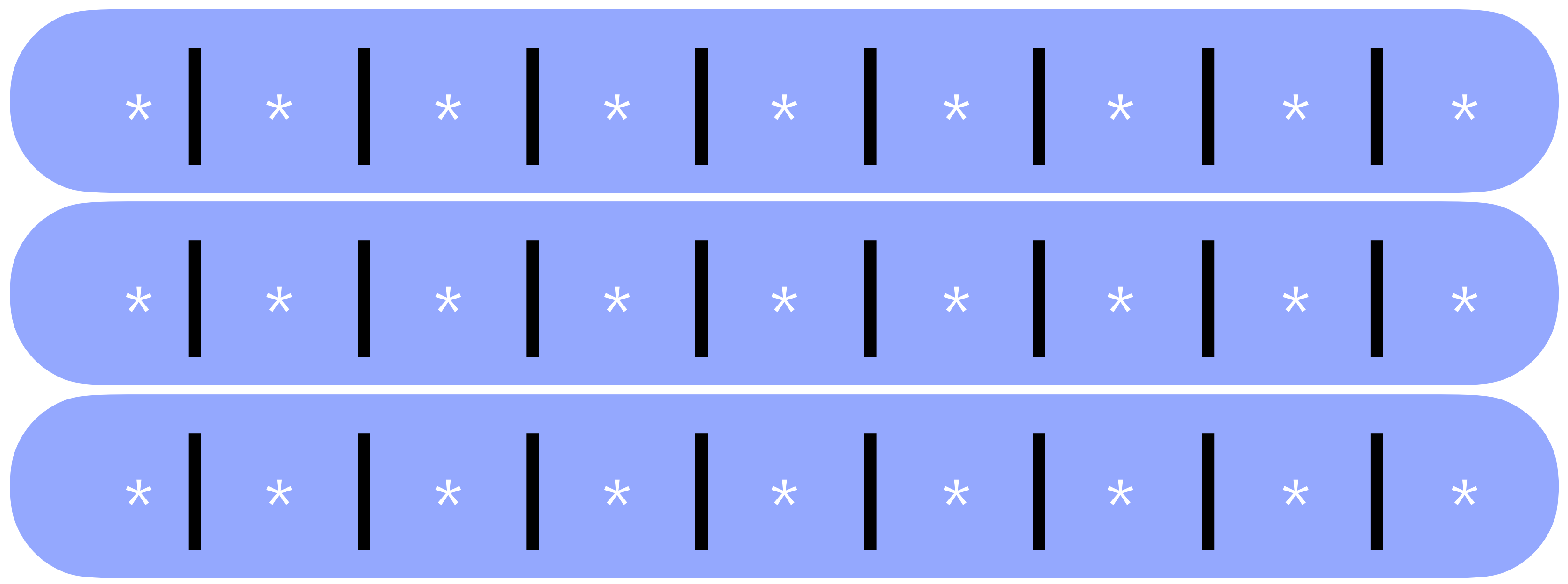
Can CFE PDS misbehave?



Can CFE PDS misbehave?

Stream

n,z,r,p,t,w,l,l,n,s,k,**g,o**,i,w,…

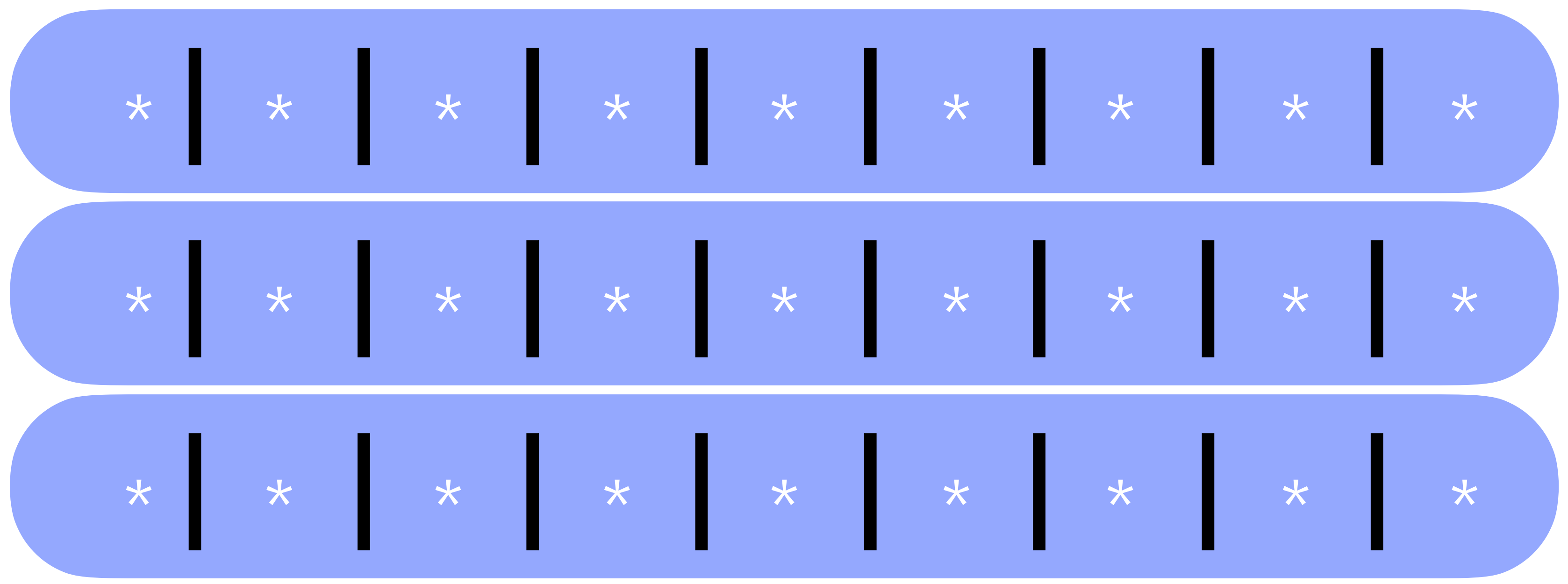


CFE PDS

query(x)

** >> freq(x)
or
** << freq(x)

Can CFE PDS misbehave?



CFE PDS

query(x)

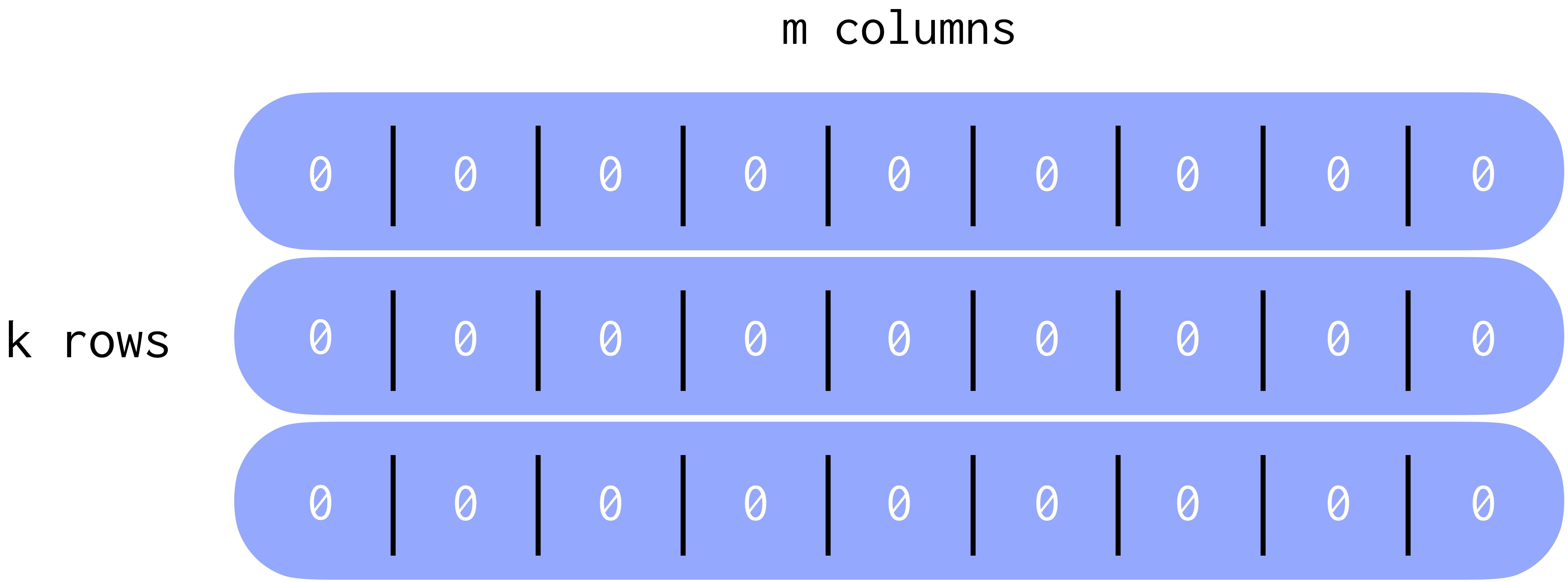
** >> freq(x)
or
** << freq(x)

Stream

n,z,r,p,t,w,l,l,n,s,k,**g**,**o**,i,w,…

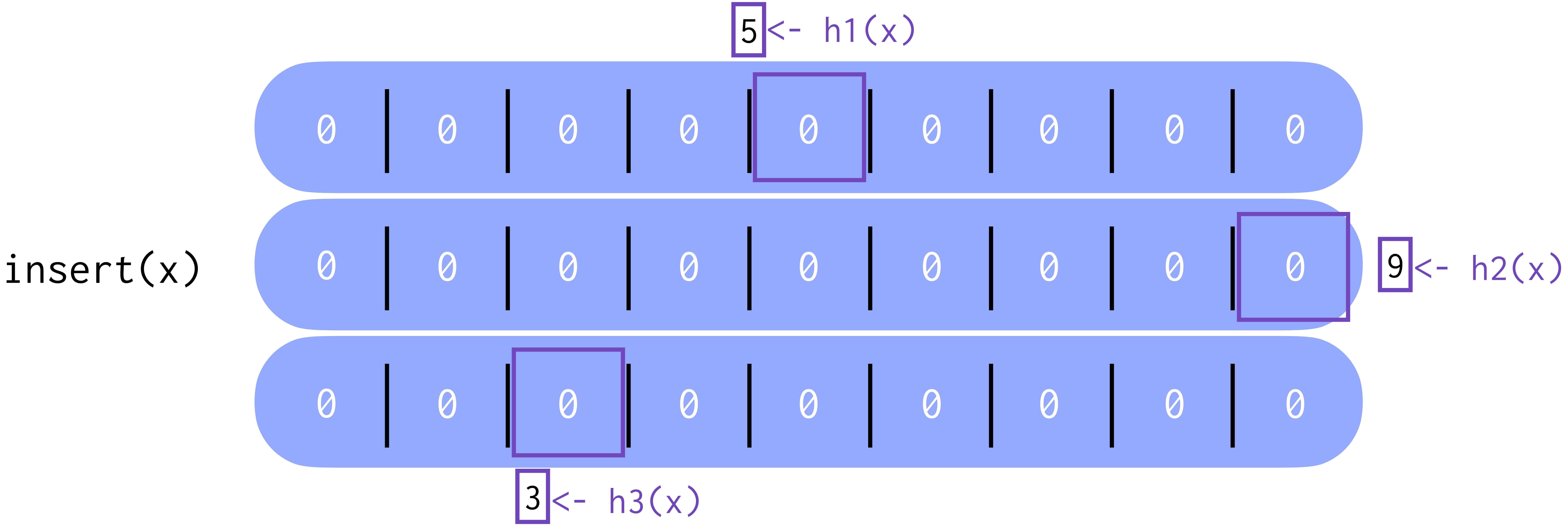
Attacks against CMS, HeavyKeeper, Count sketch, CMS w/ conservative updates

Count-min sketch (CMS)

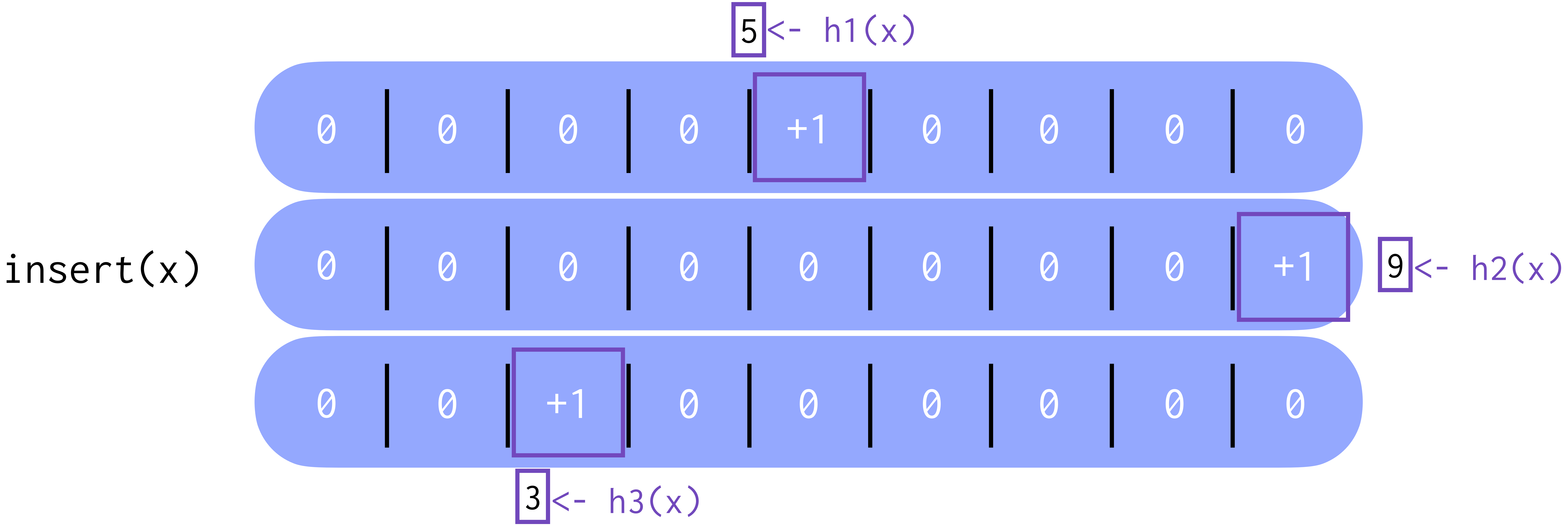


CMS: how does it work?

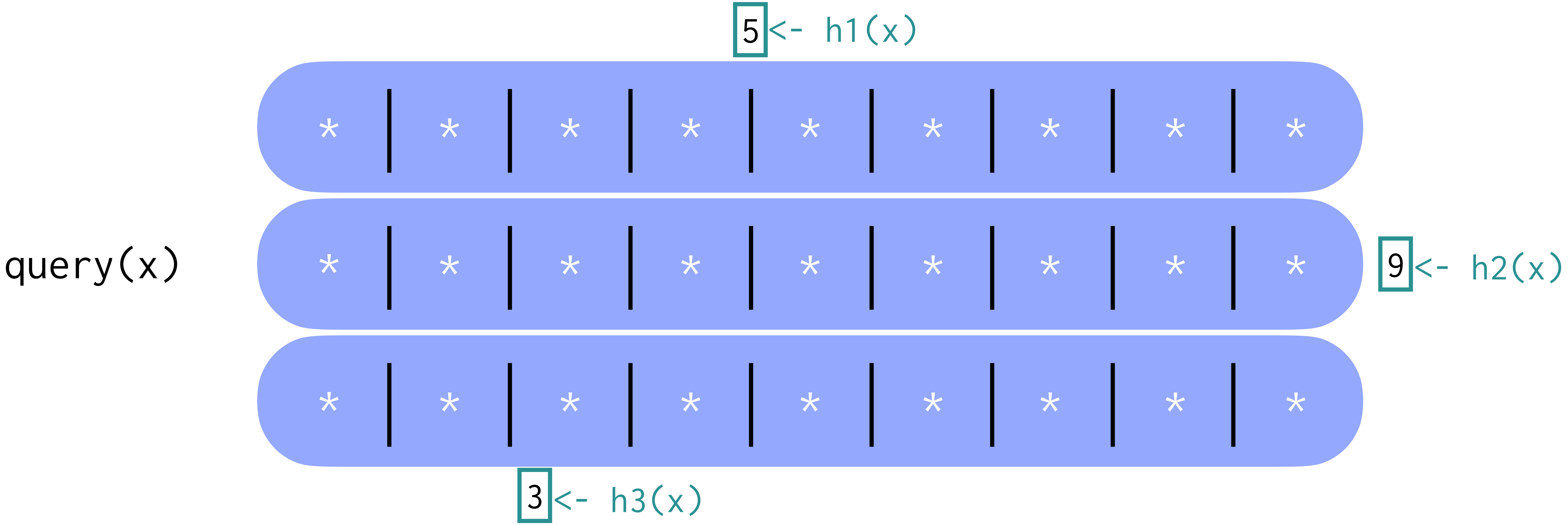
CMS: how does it work?



CMS: how does it work?

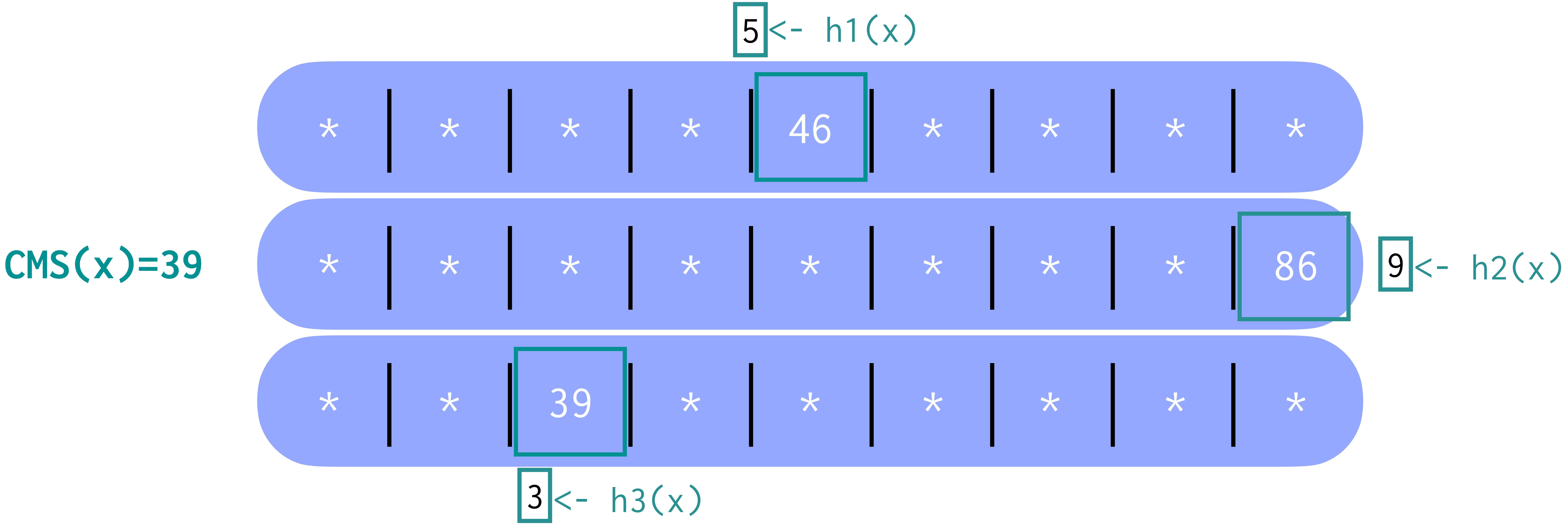


CMS: how does it work?

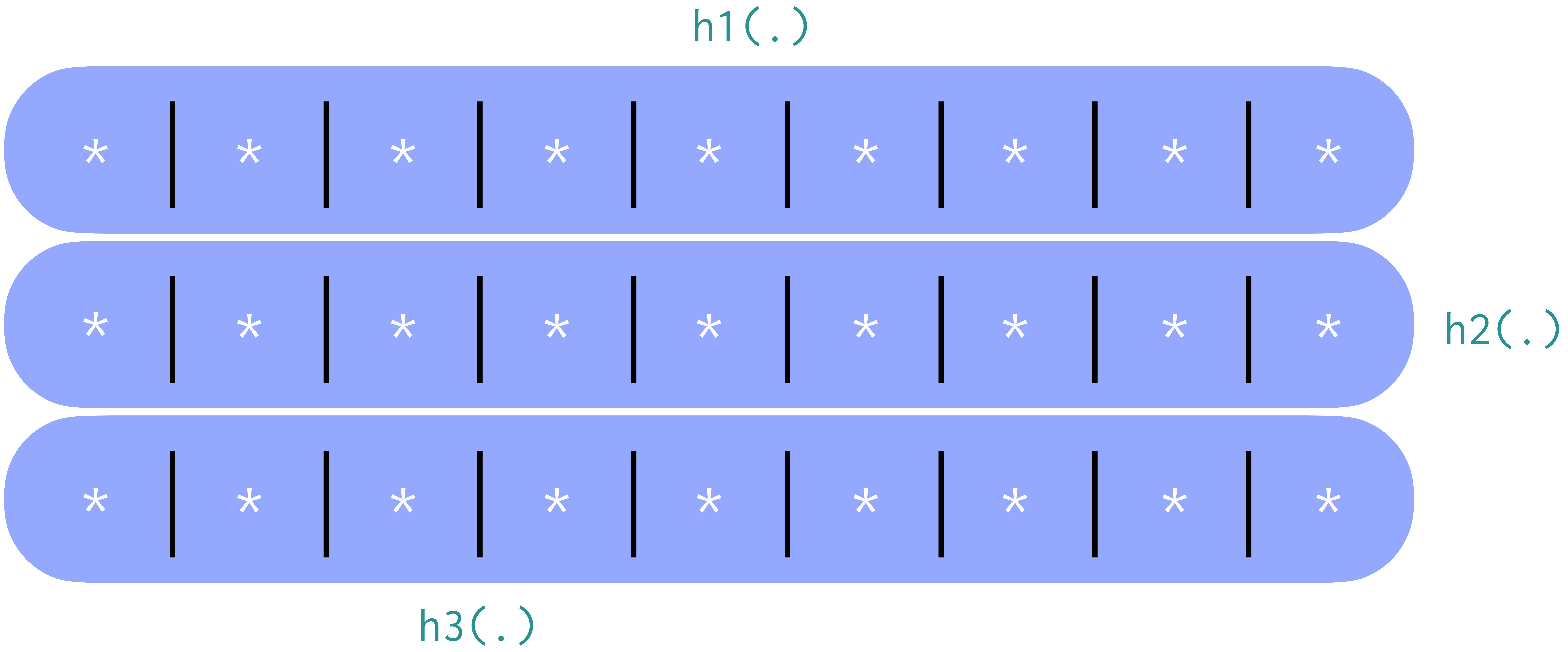


CMS: how does it work?

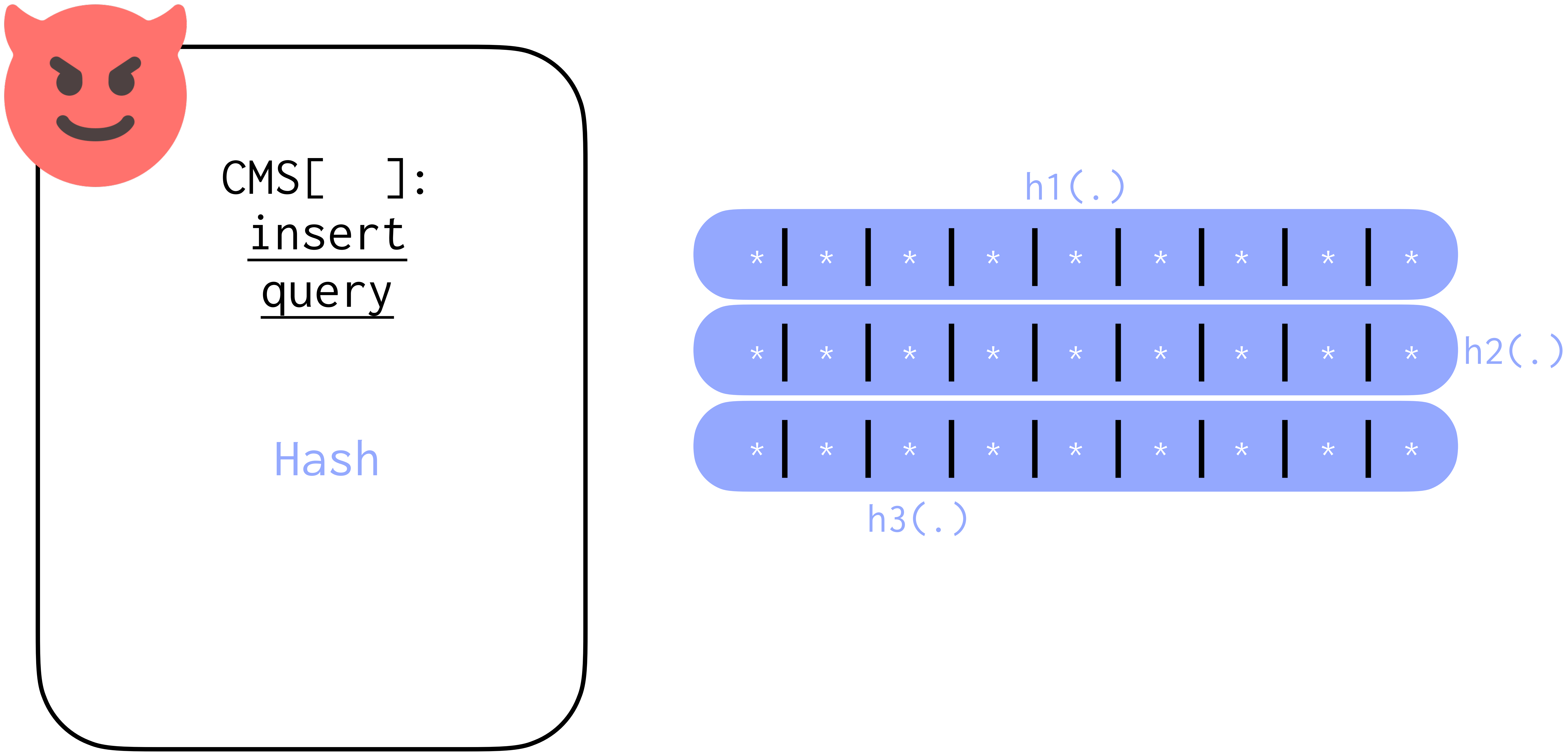
CMS: how does it work?



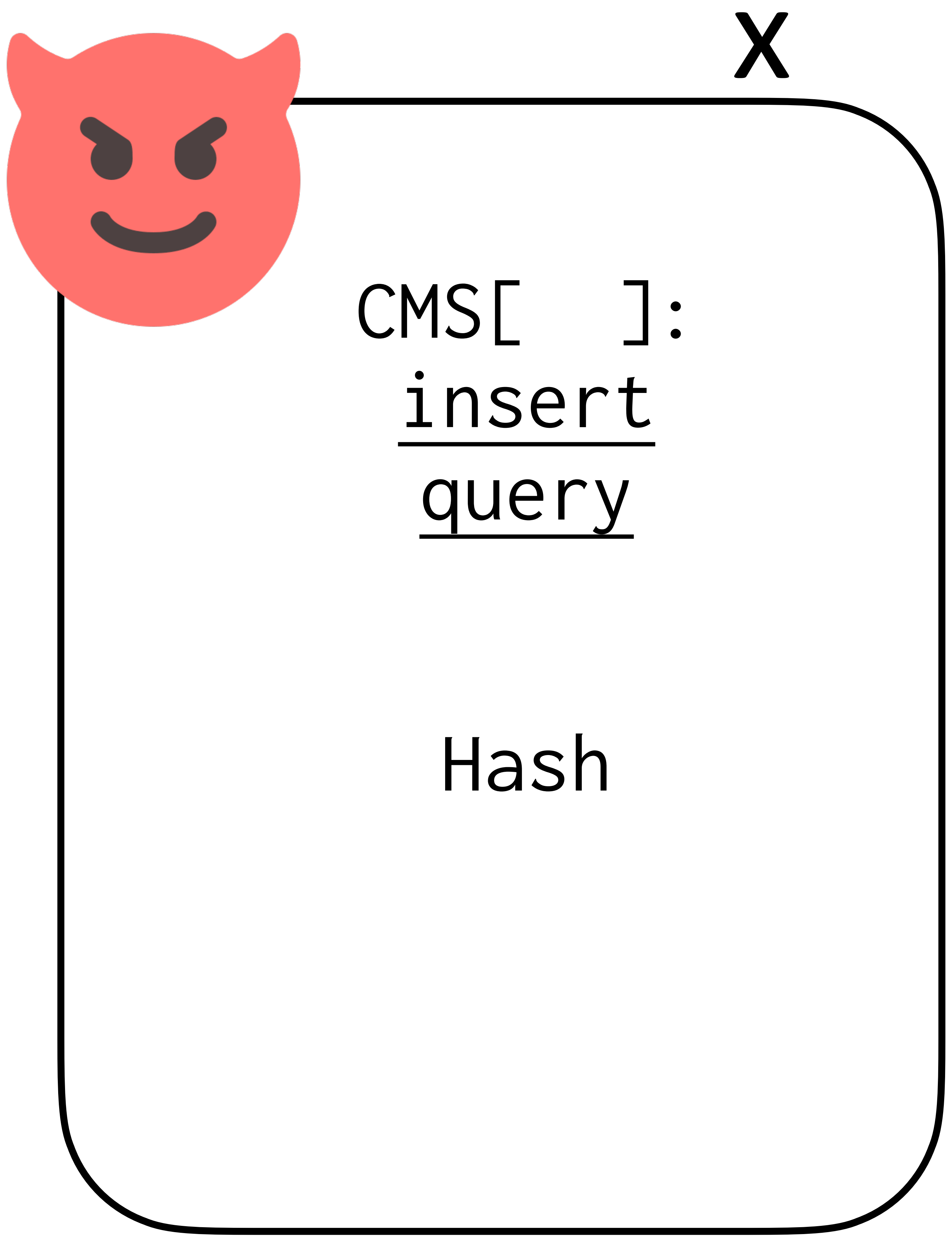
CMS



CMS: attack model



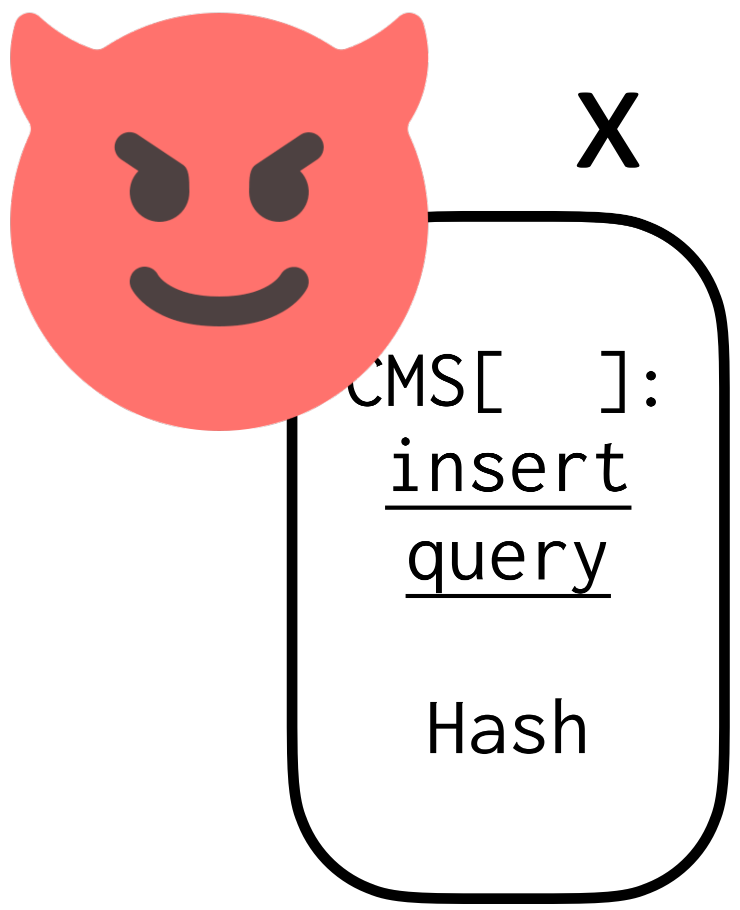
CMS: attack goal

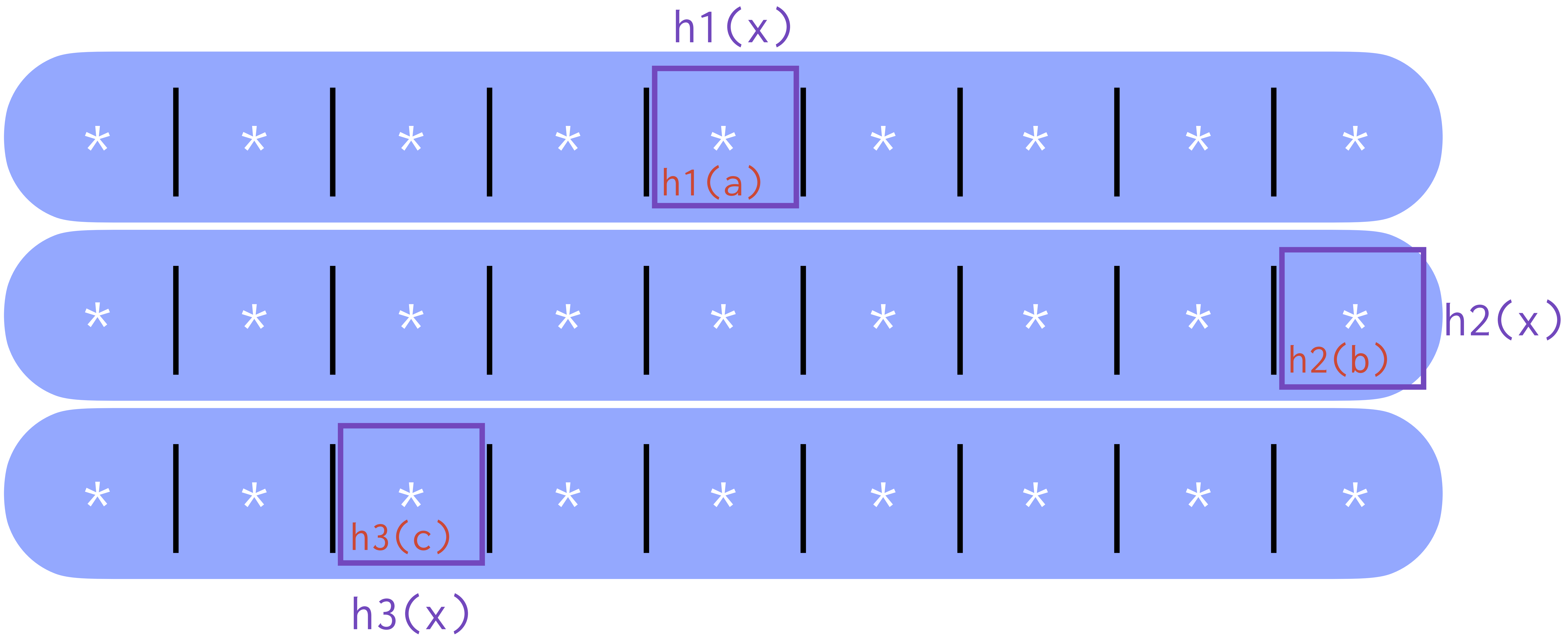


Maximise
CMS error

query(x) >> true_frequency(x)

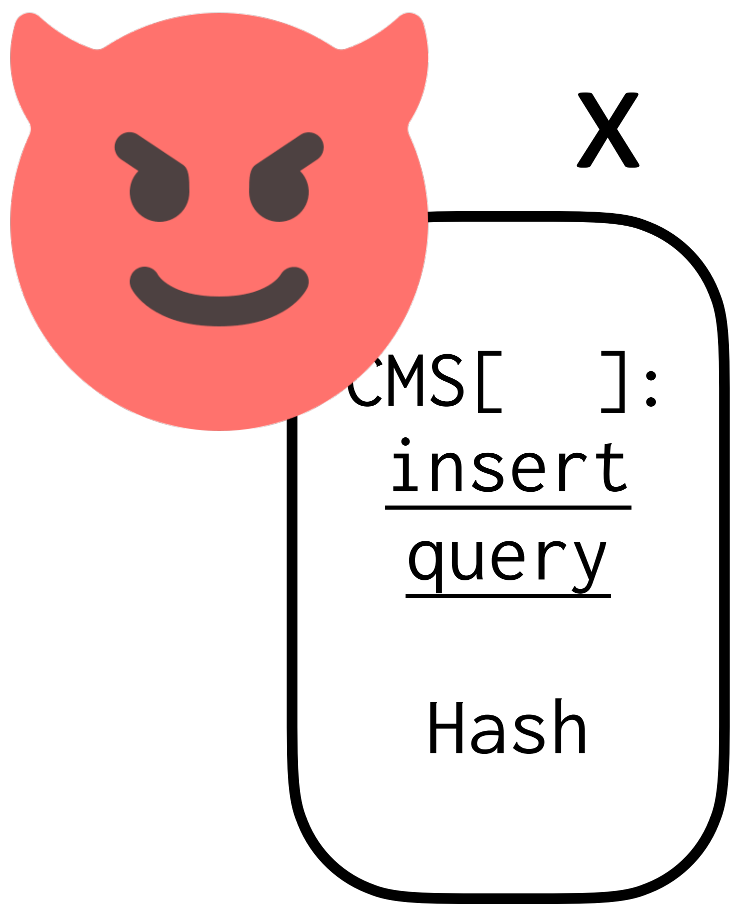
CMS: attack

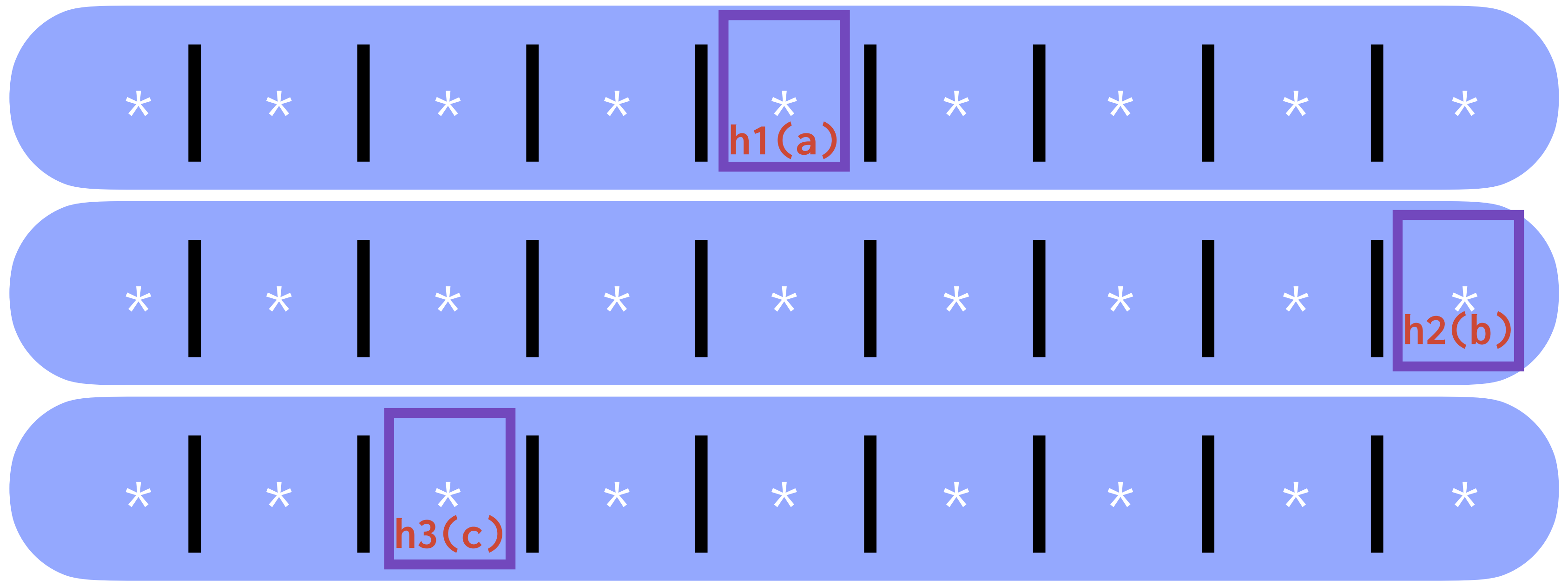




Cover set = {a, b, c}

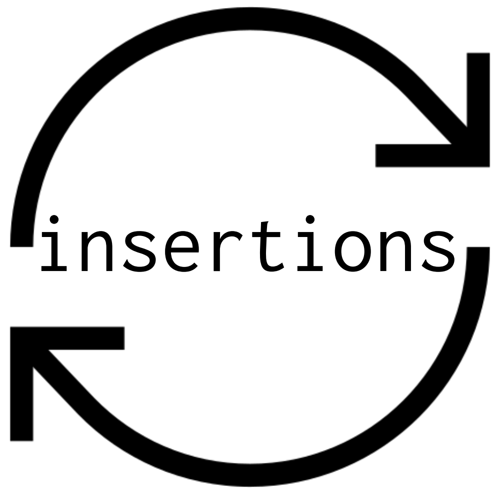
CMS: attack



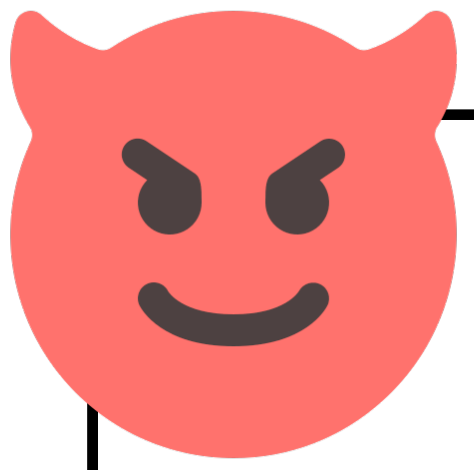


Cover set = {a, b, c}

Cover set



CMS: attack



X

CMS[ ]:

insert

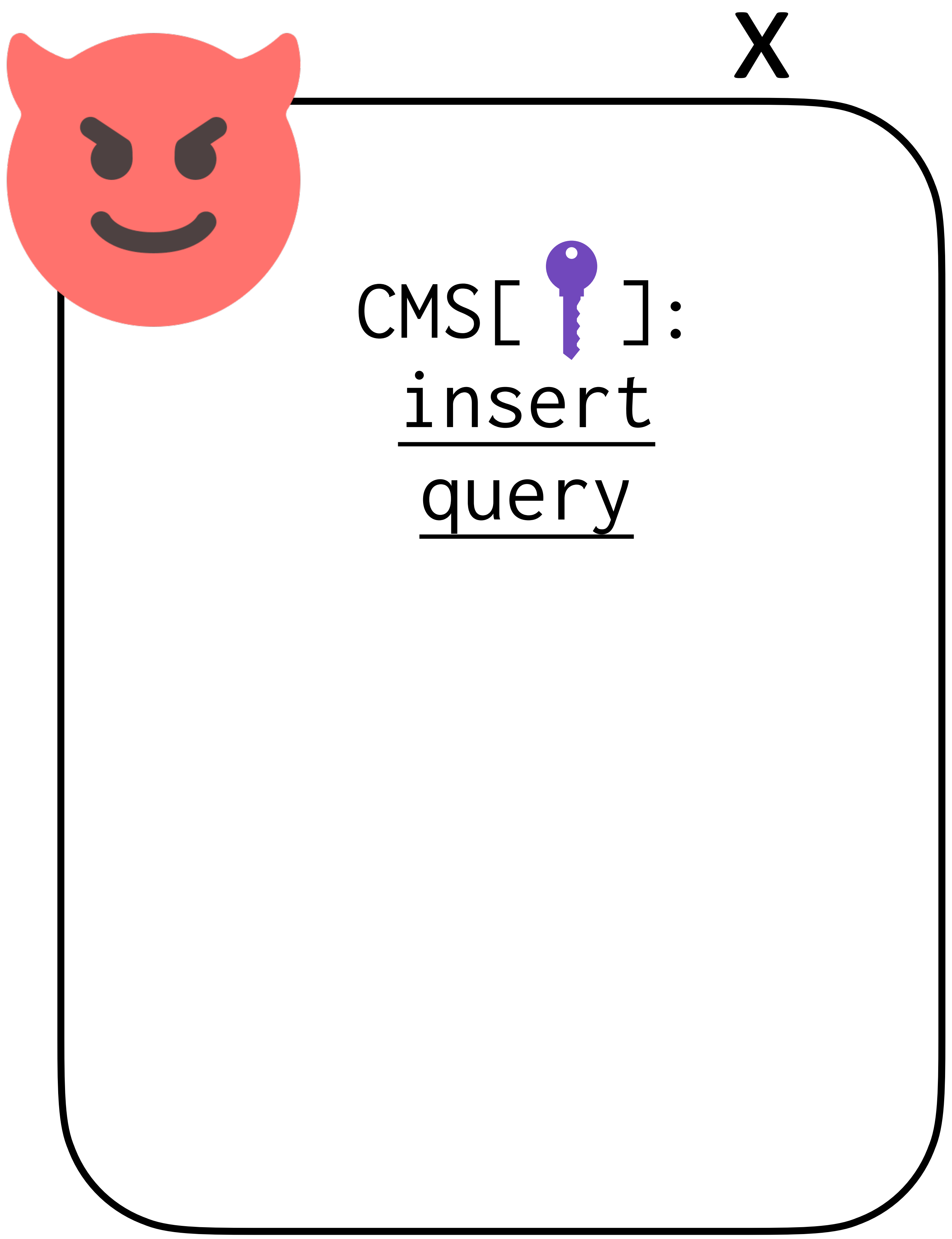
query

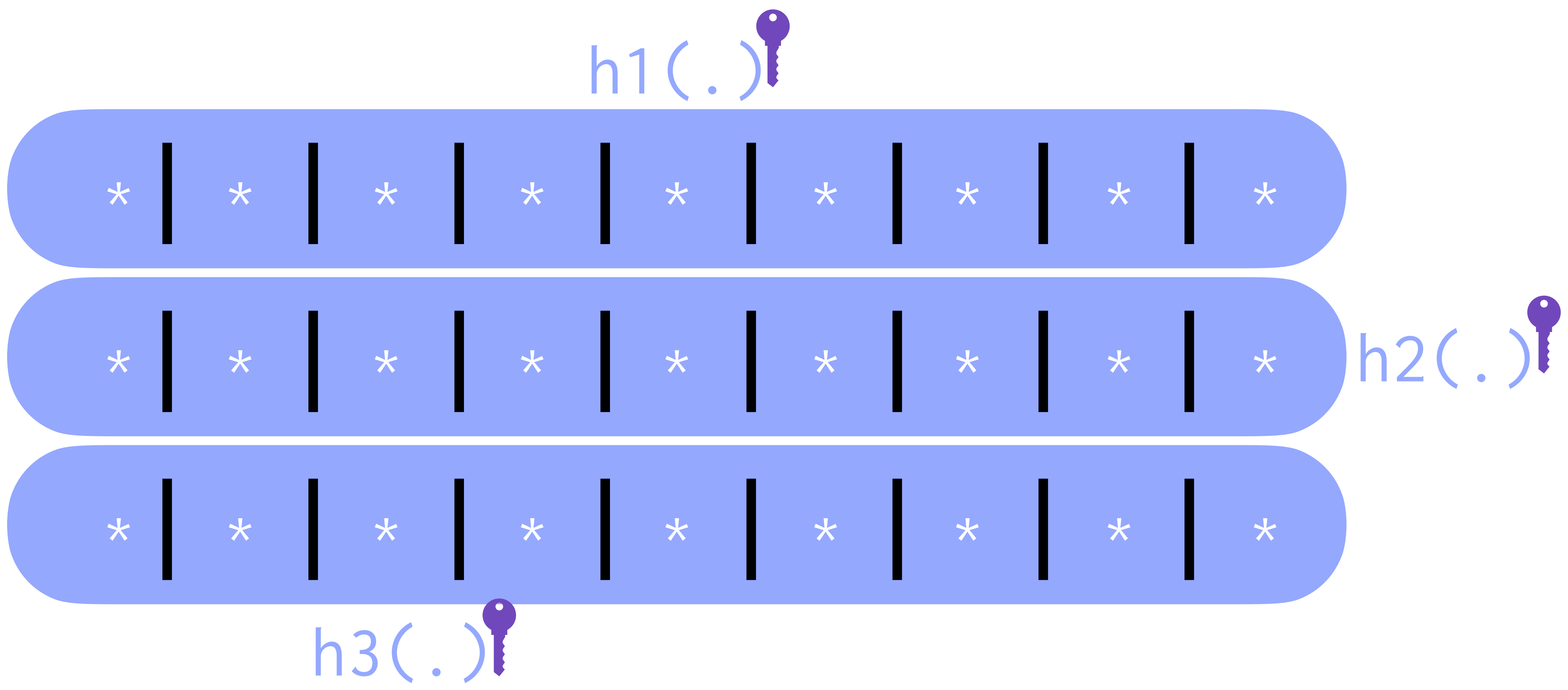
Hash

Err:

insertions/k

CMS: attack model cont.





CMS: attack

X

CMS[ ]:

insert

query

Err:

insertions/k - m Hk

Our attacks make

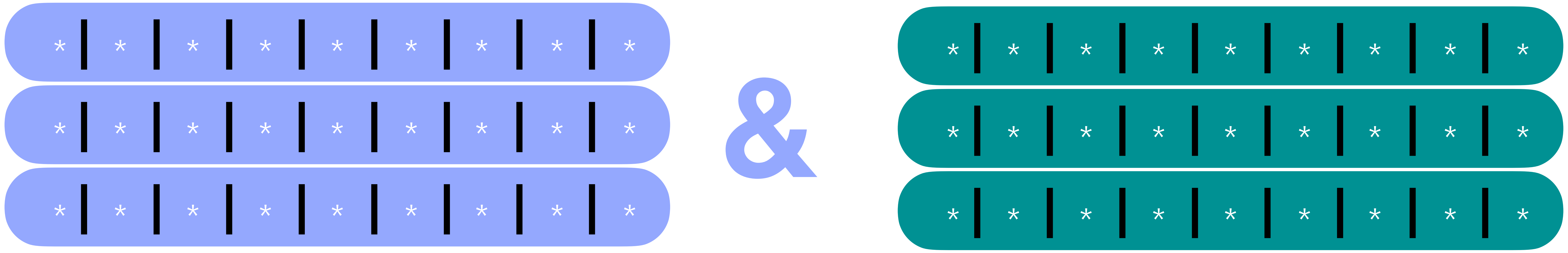
elements **absent** from the stream marked as **heavy**

Our attacks make

elements **absent** from the stream marked as **heavy**
or
high-frequency elements marked as **absent**.

More robust CFE PDS: Overestimator + Underestimator

More robust CFE PDS: Overestimator + Underestimator

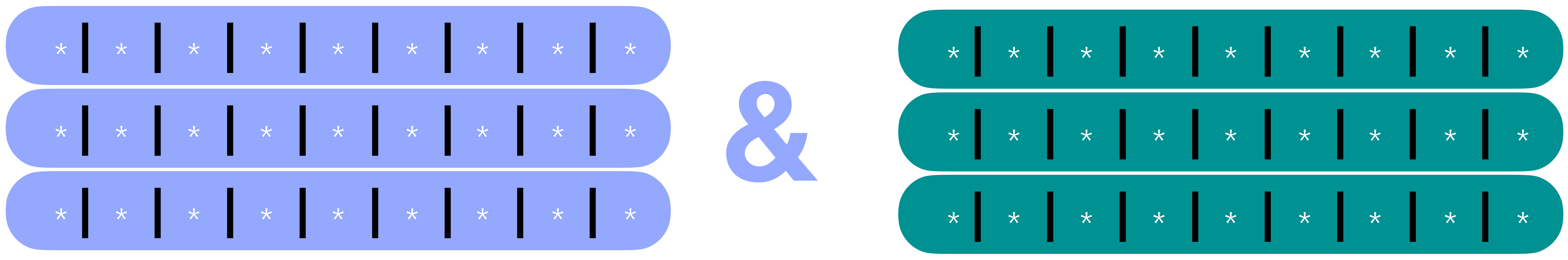


CMS M

*HeavyKeeper A

CMS est & *HeavyKeeper est ---refine---> final est

Count Keeper (CK)



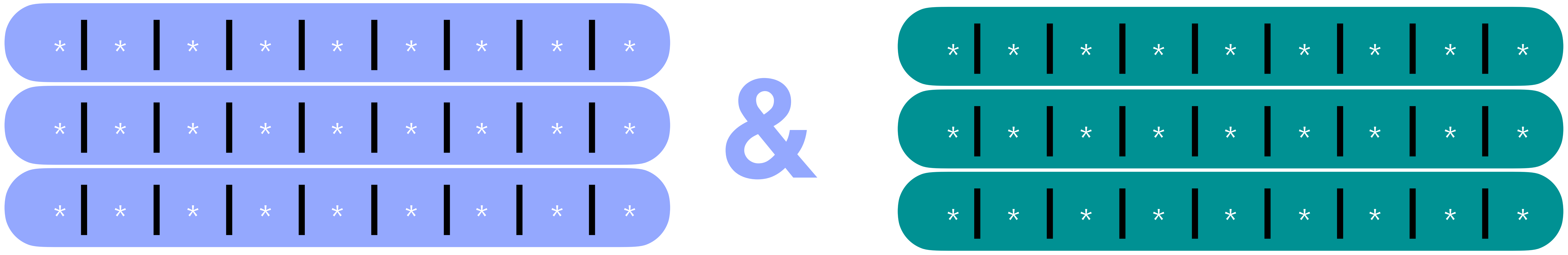
CMS M

*HeavyKeeper A

CK ~~ CMS ~~ HK

Honest setting experiments

Count Keeper (CK)



CMS M

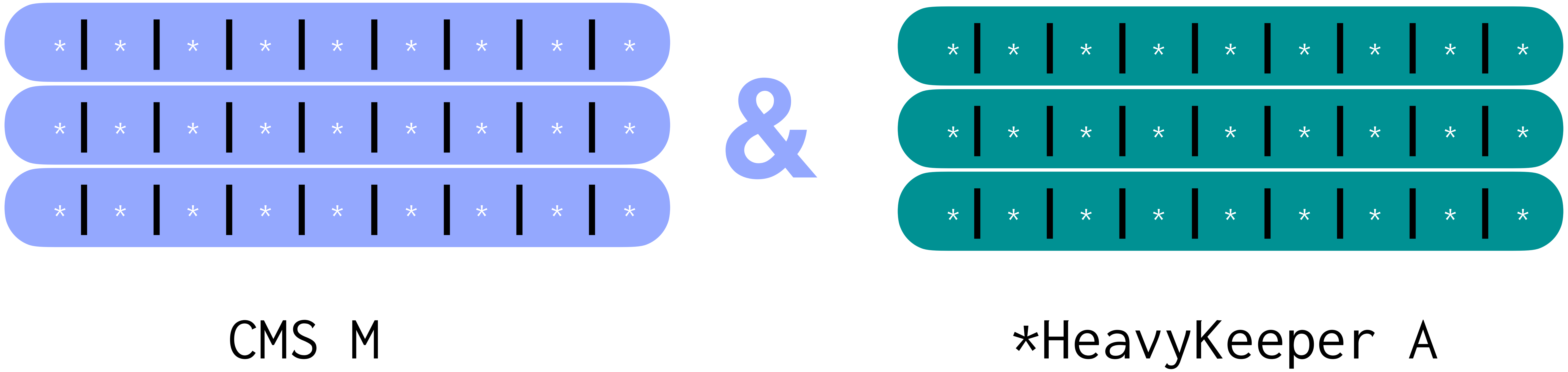
*HeavyKeeper A

Err:

CK < 1/2 CMS
CK << 1/2 HeavyKeeper

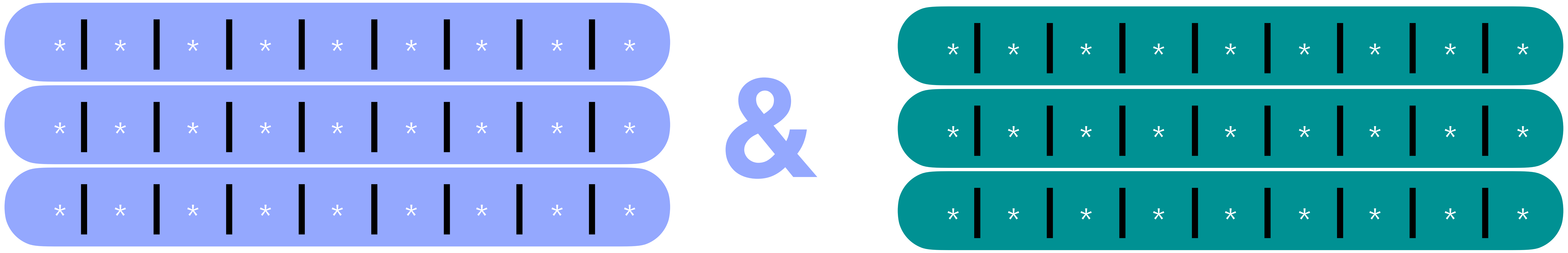
Attack experiments

Count Keeper (CK)



+ error related properties (see CCS23 paper) :)

Count Keeper (CK)

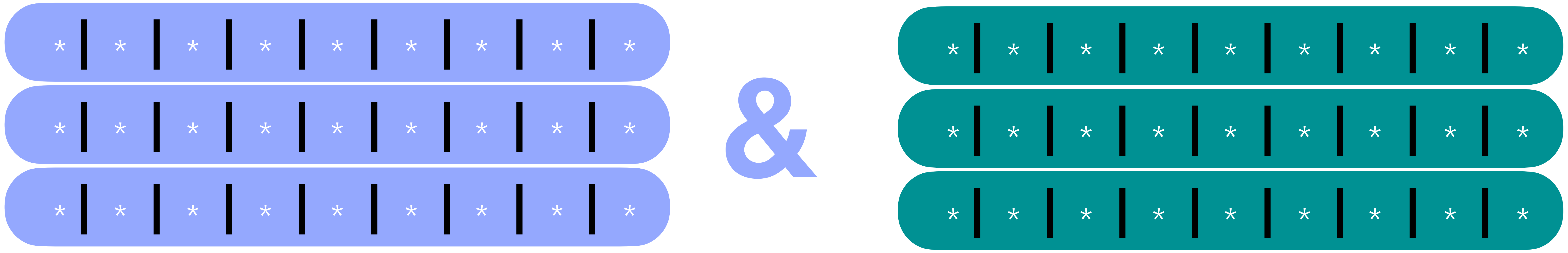


CMS M

*HeavyKeeper A

CK can detect suspicious estimates

Count Keeper (CK)

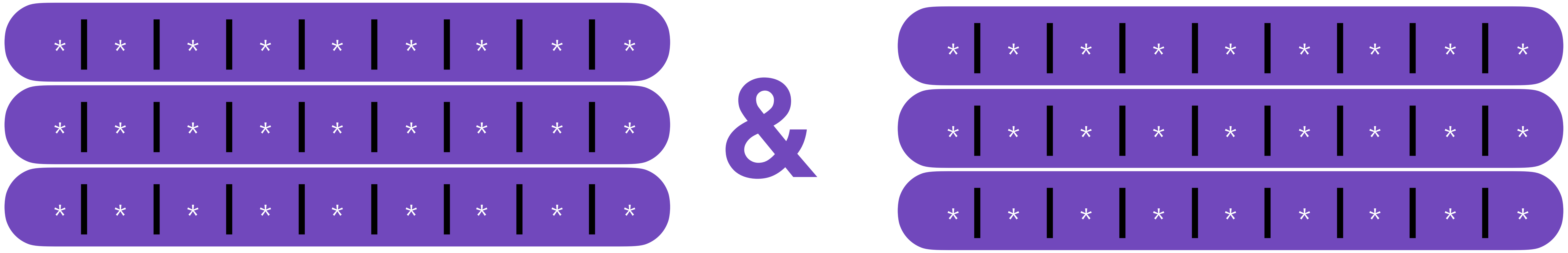


CMS M

*HeavyKeeper A

CK can detect suspicious estimates

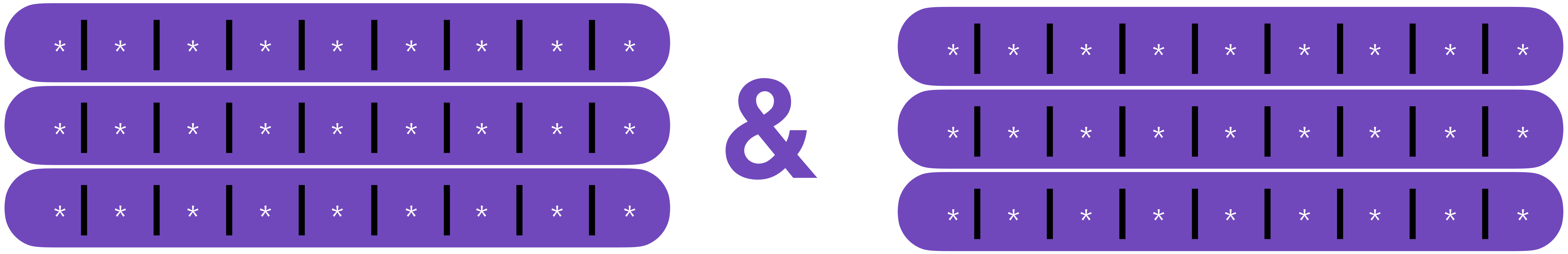
Open problems & Future work



Overestimator ?

Underestimator ?

Open problems & Future work



PDS A ?

PDS B ?

Our work / Open problems

- Approximate Membership Query PDS (w/o and w/ deletions)

  Adversarial correcness   Privacy   Provable security

- Compact Frequency Estimation (CFE) PDS

  Adversarial correcness   Privacy   Provable security

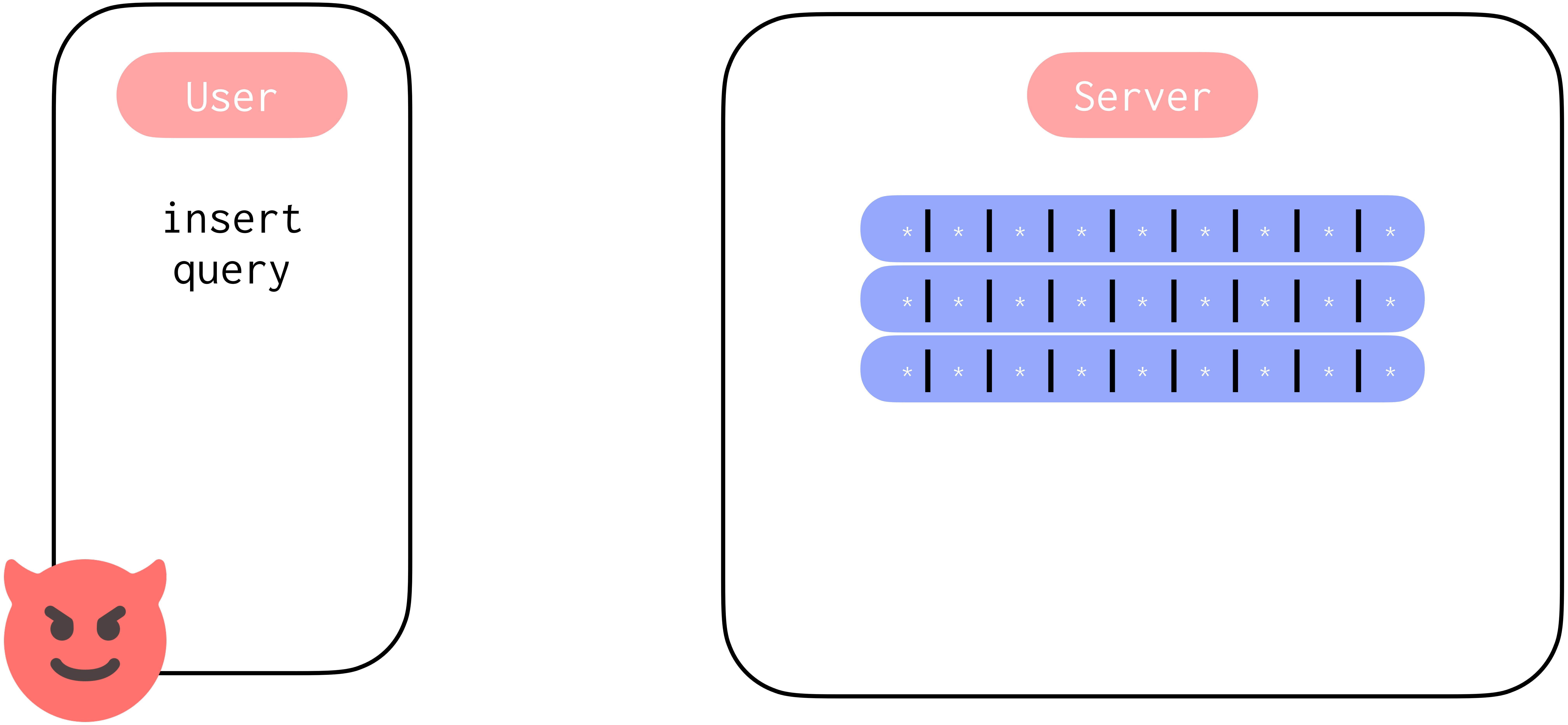
- Other PDS

  Adversarial correcness   Privacy   Provable security

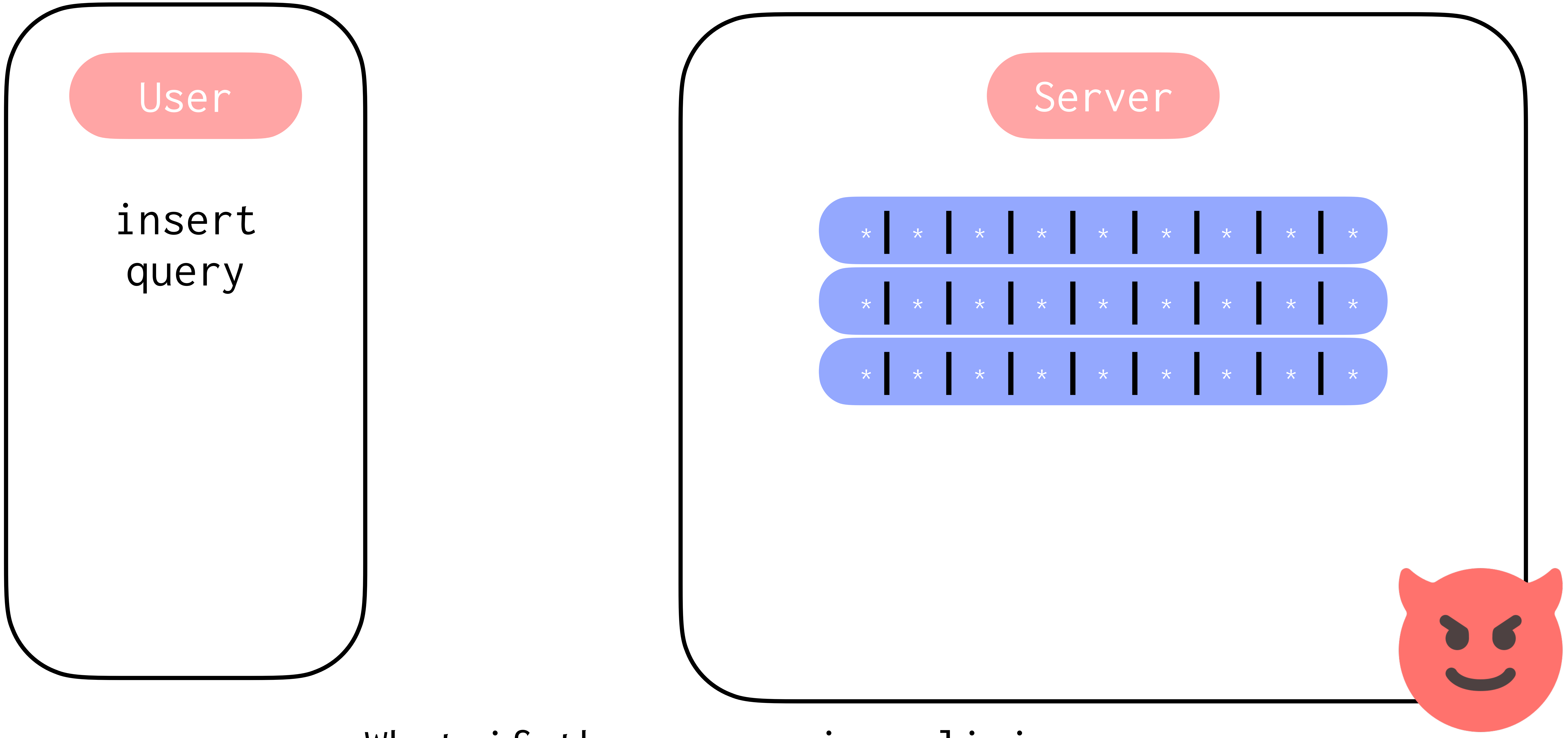
- Practical implementation

  Adversarial correcness

Future work



Future work



What if the server is malicious
and the user is honest?

Thank you!

Thank you!

Approximate Membership Query PDS (CCS22)

Compact Frequency Estimation (CFE) PDS (CCS23)