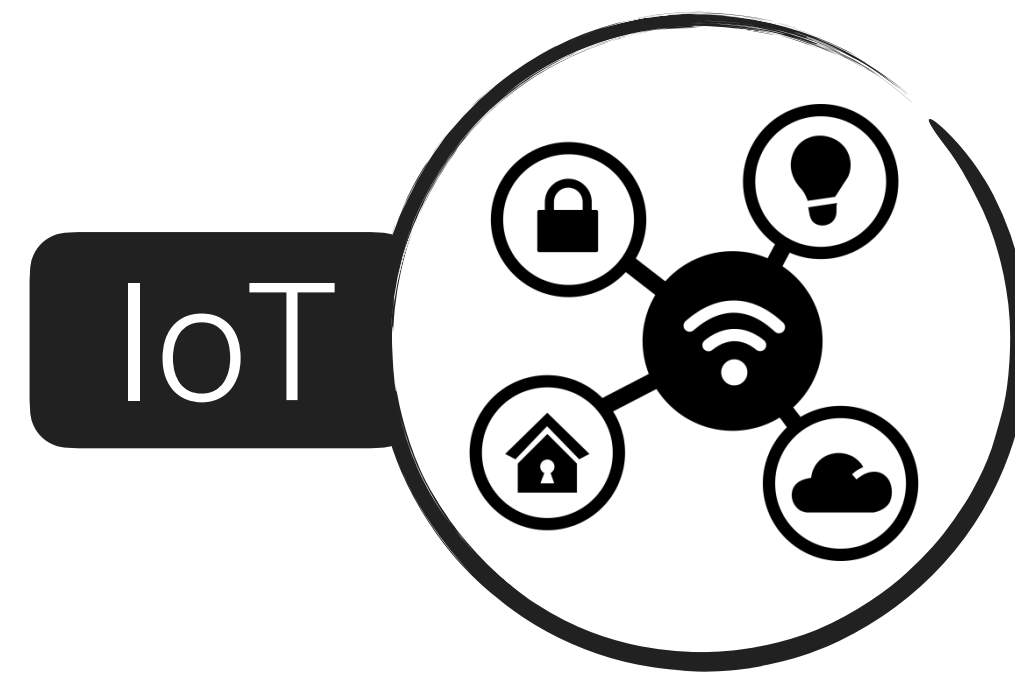




*Toward Building **Deletion Compliant** **Data Systems***

Subhadeep Sarkar



IoT

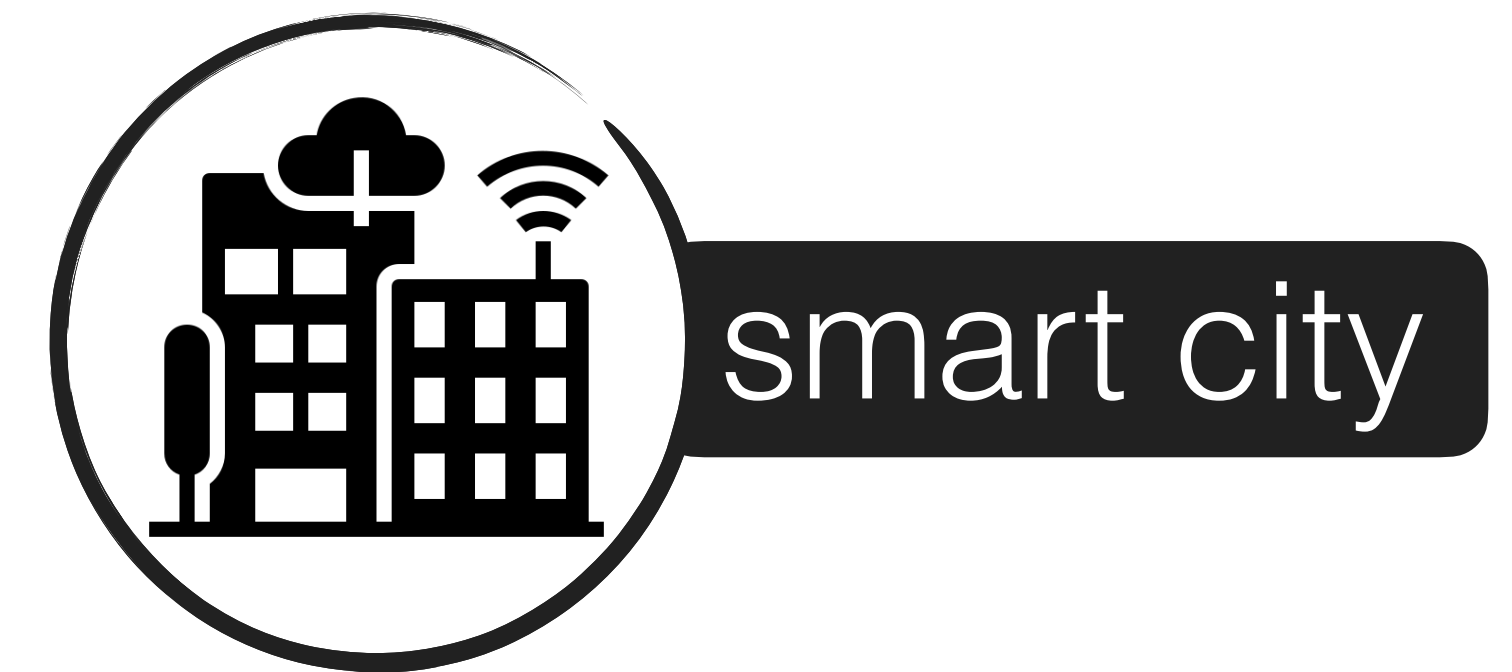
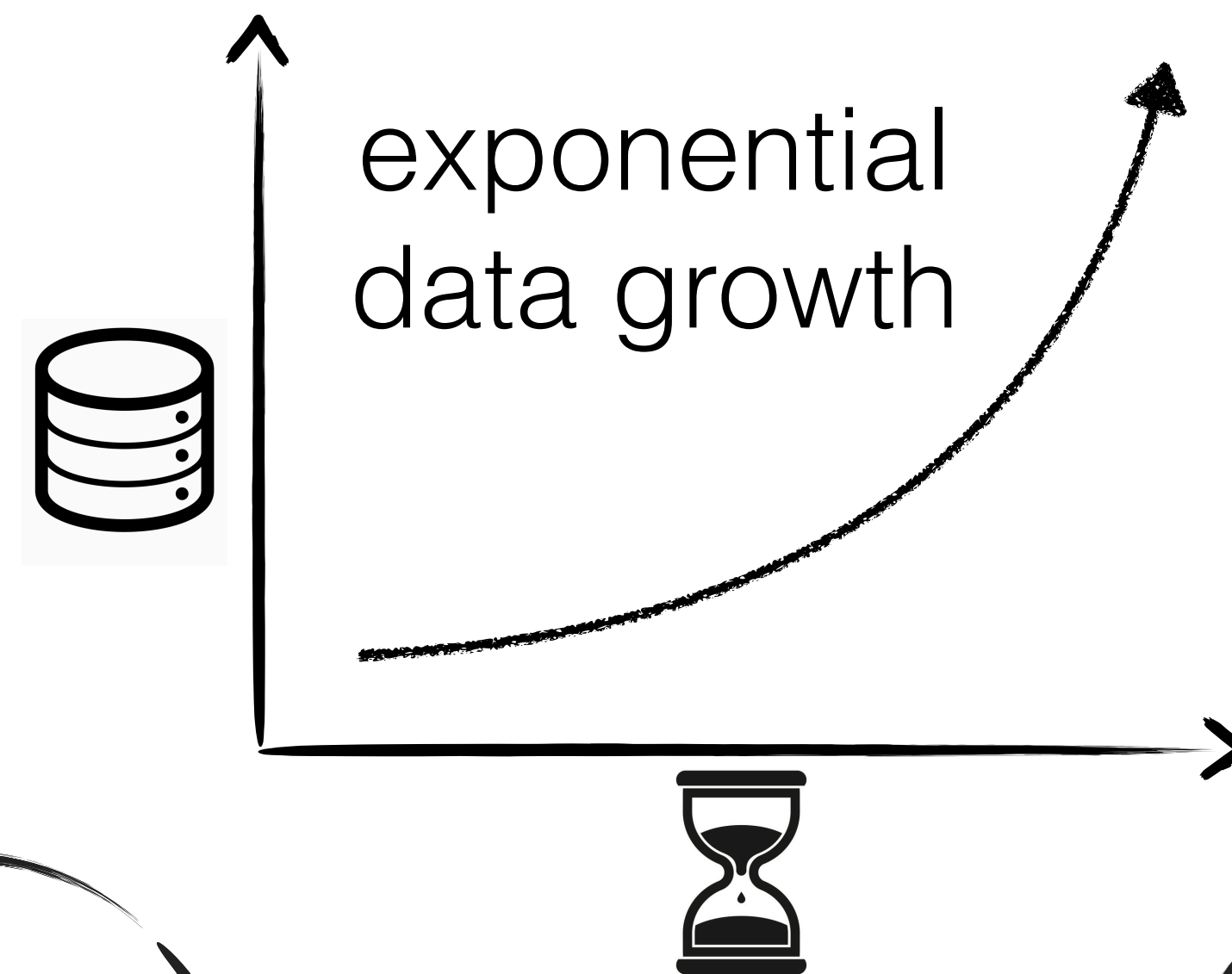


5G

5G



edge
computing



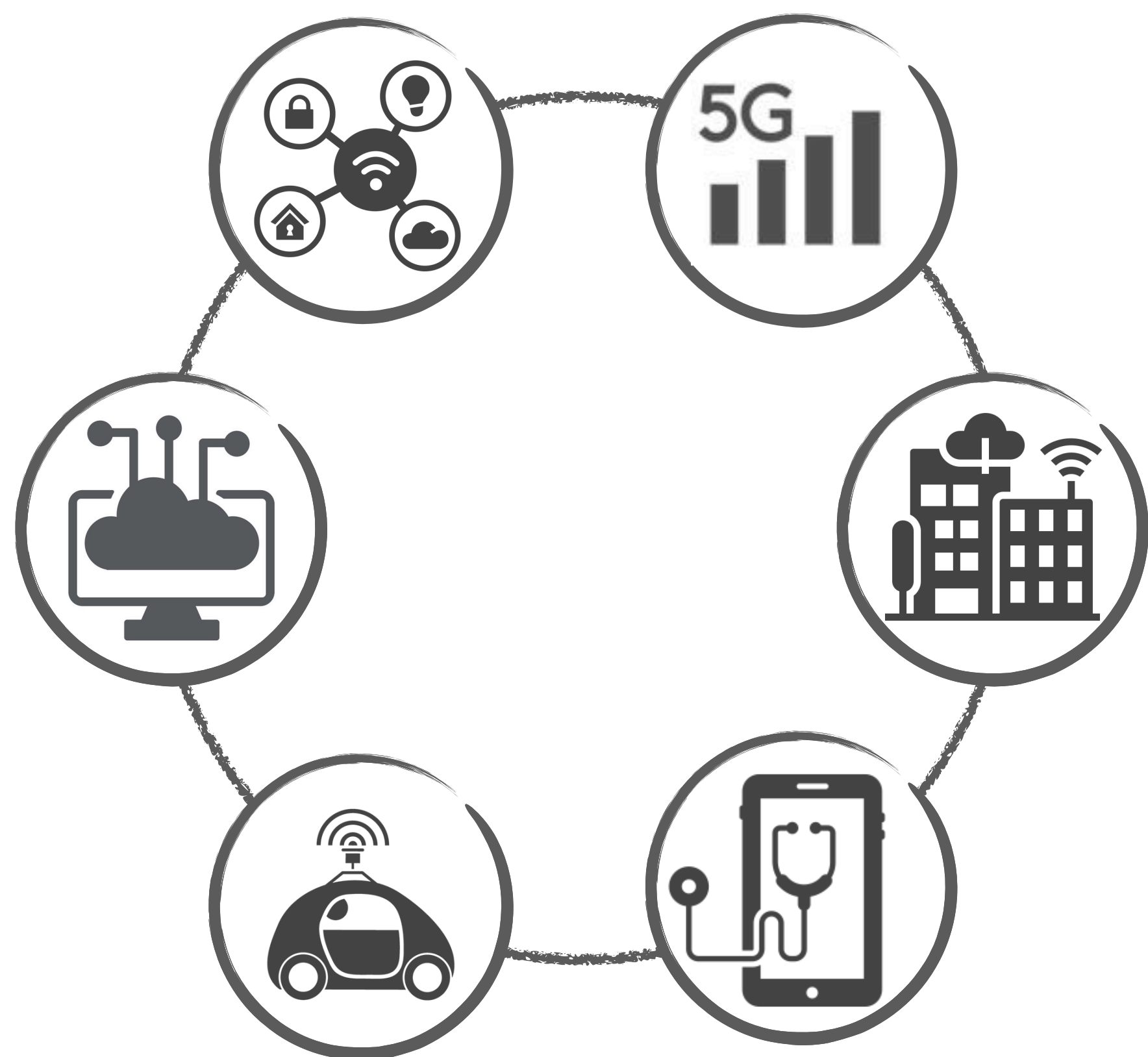
smart city



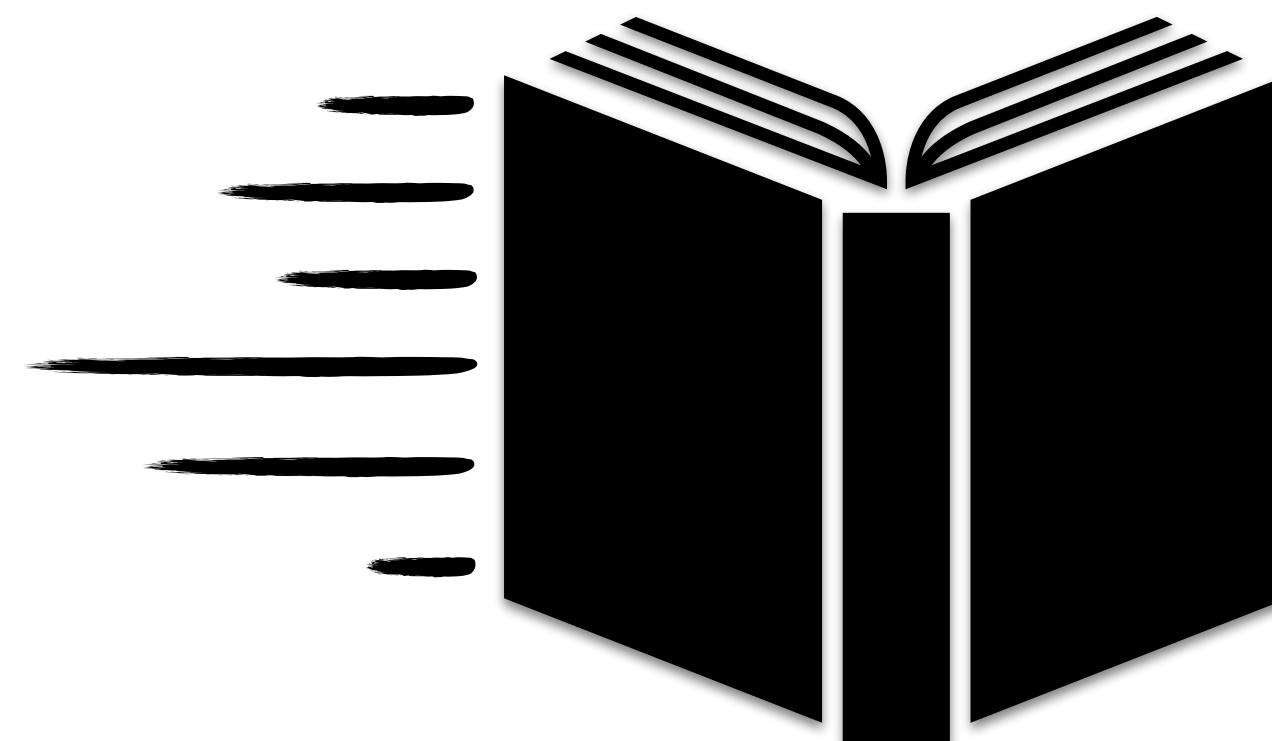
autonomous
vehicles



smart
healthcare



fast writes



fast reads

Out-of-place paradigm

Relational & Array-based

NoSQL

SciDB

VERTICA

[tile]DB

SQLite

monetdb

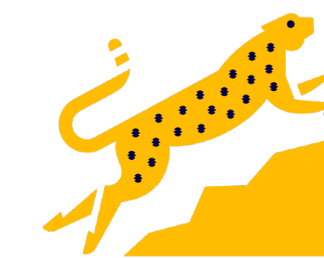
influxdb

SAP HANA

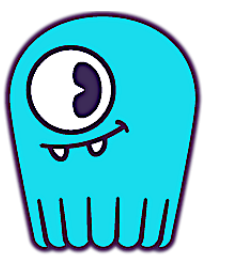


Bigtable

WT



RocksDB



SCYLLA



DynamoDB

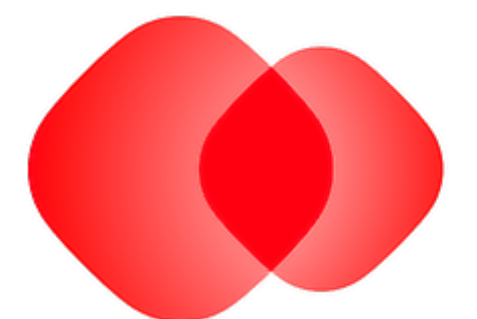


levelDB



QuasarDB

riak

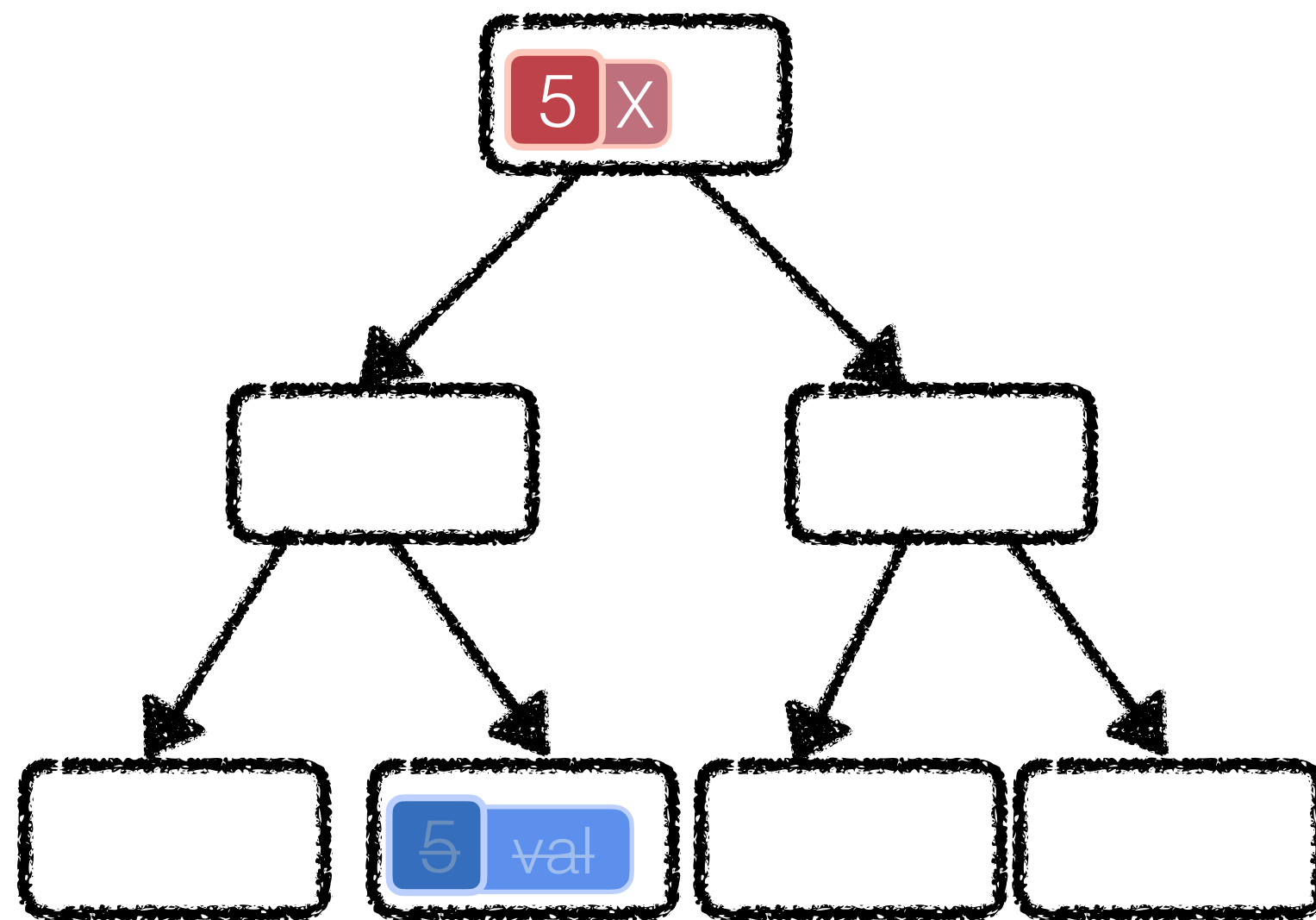


tarantool

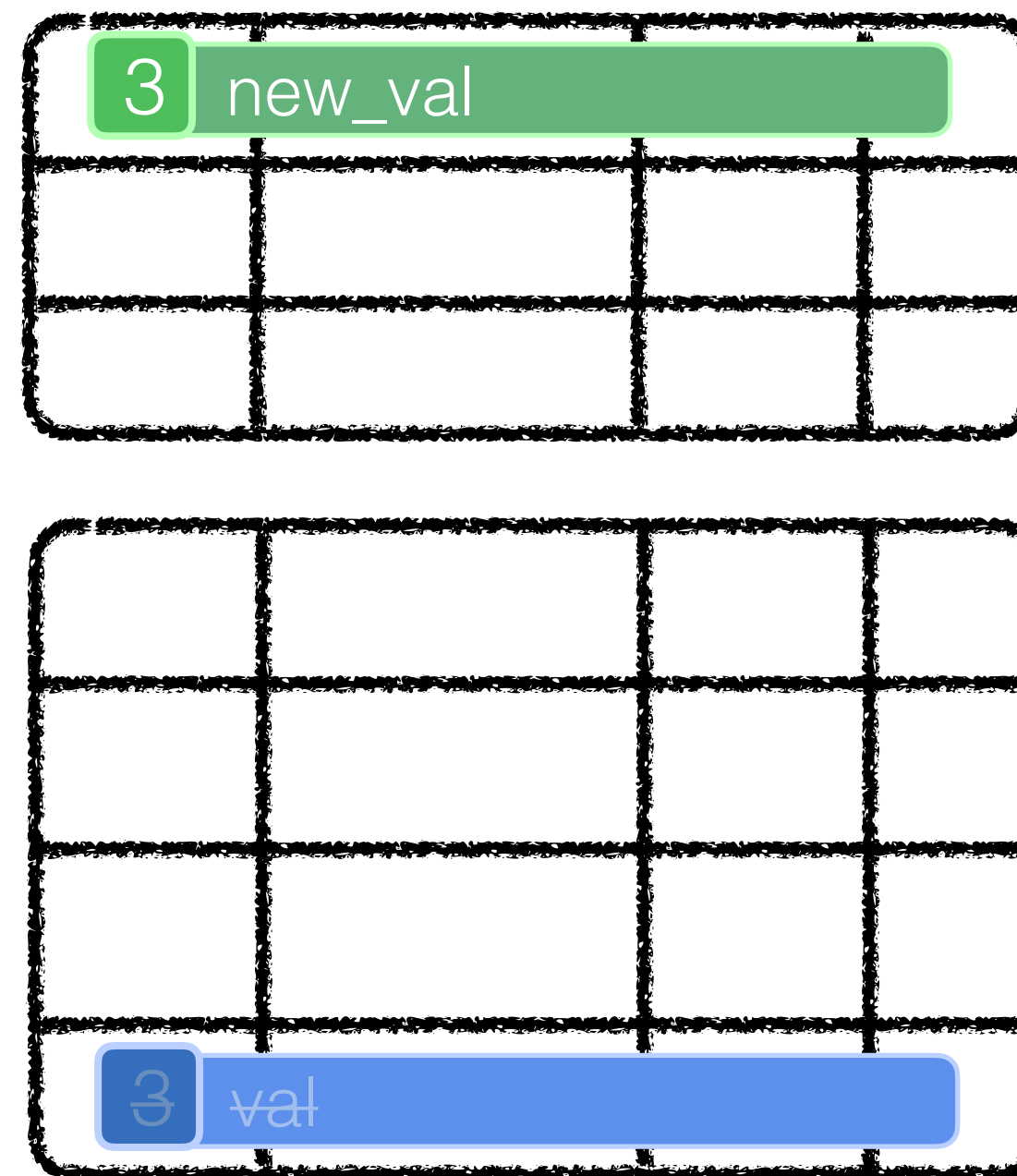
Out-of-place paradigm

↓ Out-of-place updates/deletes

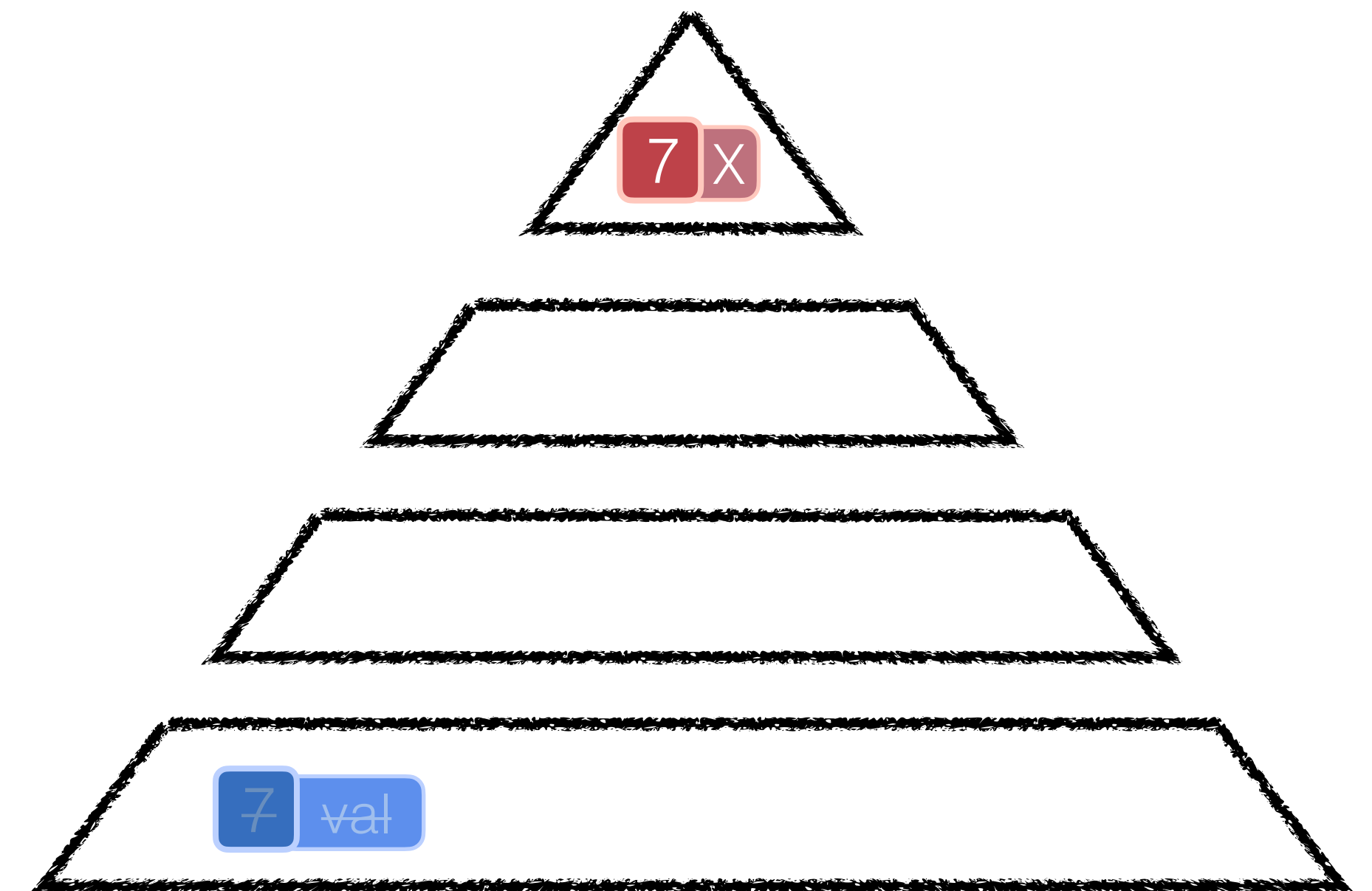
🕒 Merge updates/deletes lazily



BetrFS



VERTICA

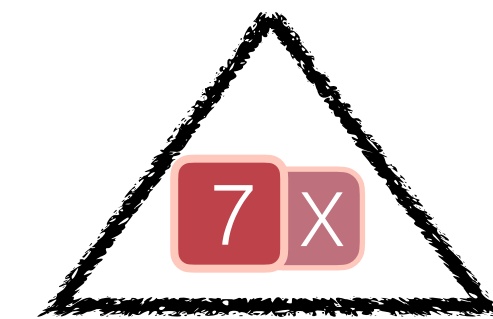


RocksDB

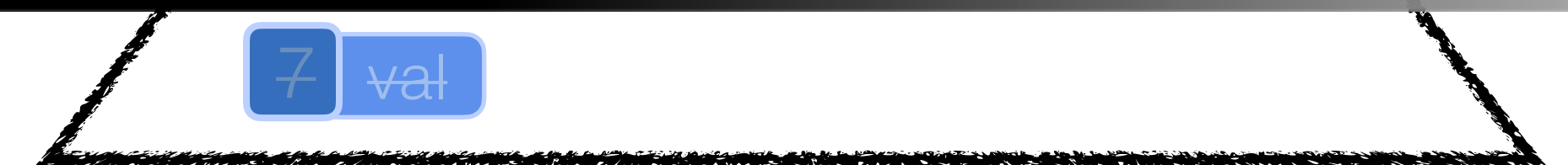
Out-of-place paradigm

↓ Out-of-place updates/deletes

🕒 Merge updates/deletes lazily



Does Not Scale with Deletes!



BetrFS

VERTICA



RocksDB

Large-scale
production

Privacy

Some items will also be deleted from 11 albums.

Delete 6,447 Items

Cancel

ZippyDB

GDPR
(EU, UK)

25.2M deletes
day

data
migration

CCPA
(California)

UP2X

100M merge via
deletes / day

periodic
cleanup

VCDPA
(Virginia)

Large-scale production

Internal DB ops

Privacy

ZippyDB

**25.2M deletes /
day**

UP2X

**100M merge via
deletes / day**

table drop

data
migration

periodic
cleanup



GDPR
(EU, UK)



CCPA
(California)



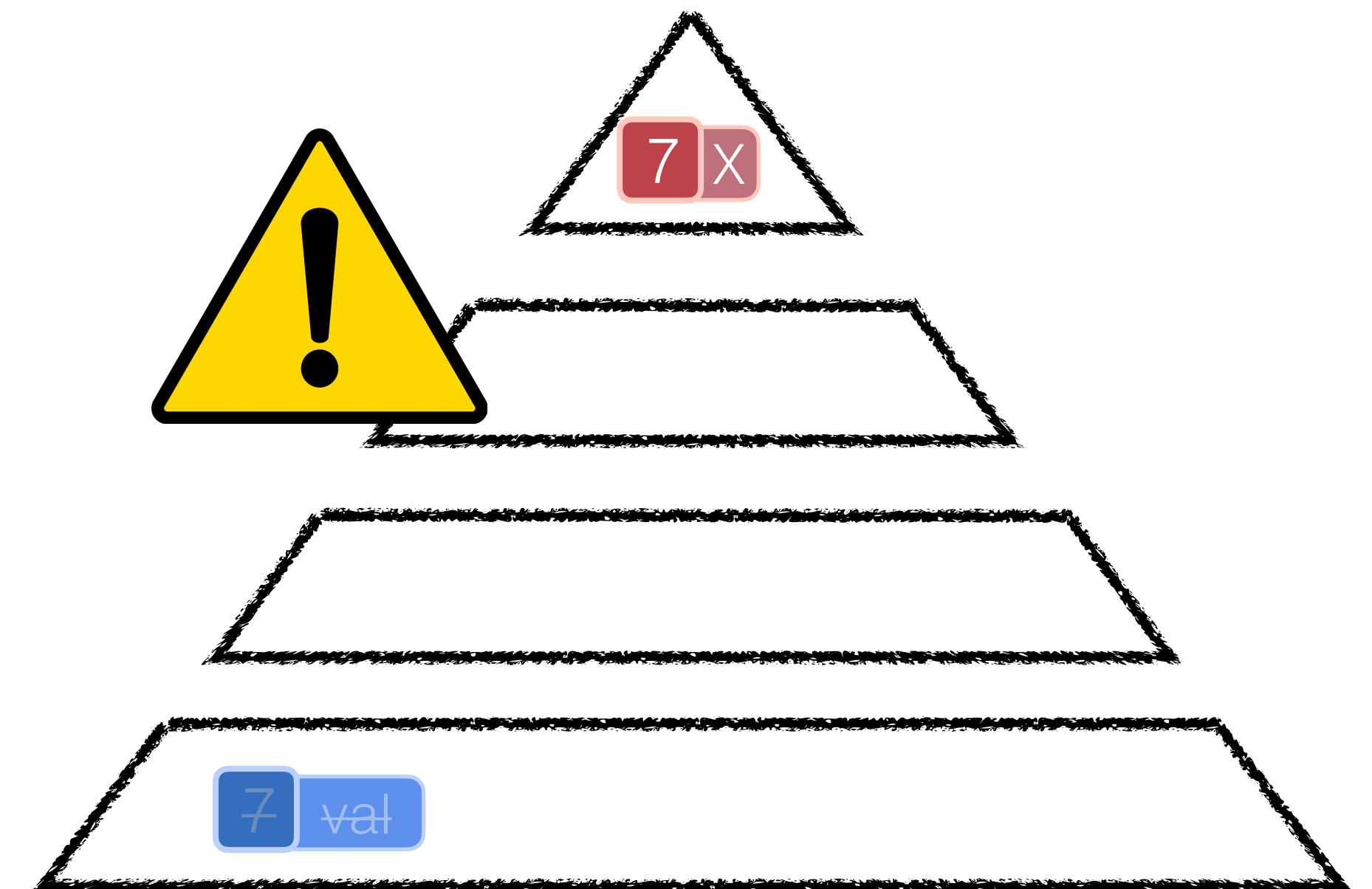
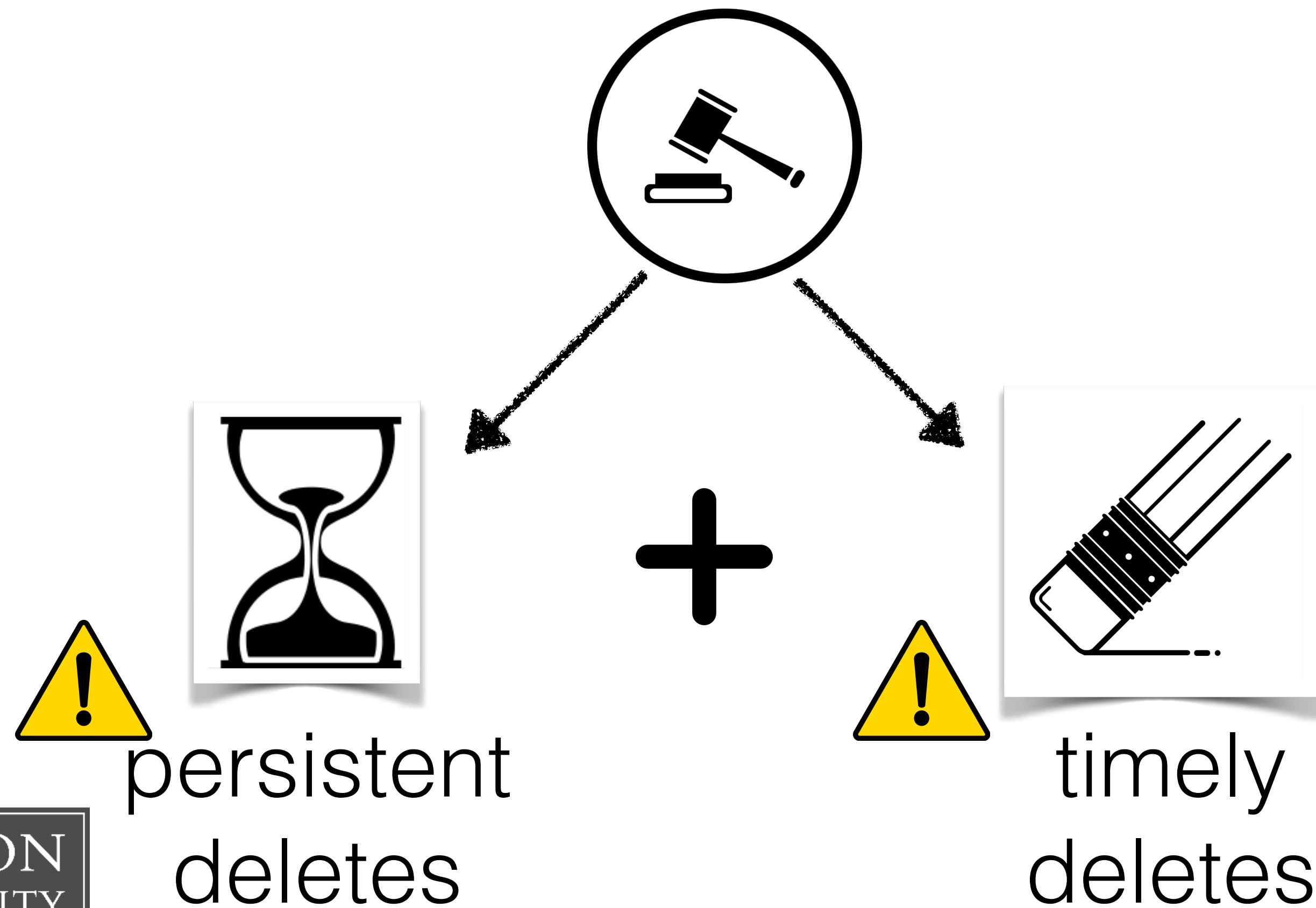
VCDPA
(Virginia)

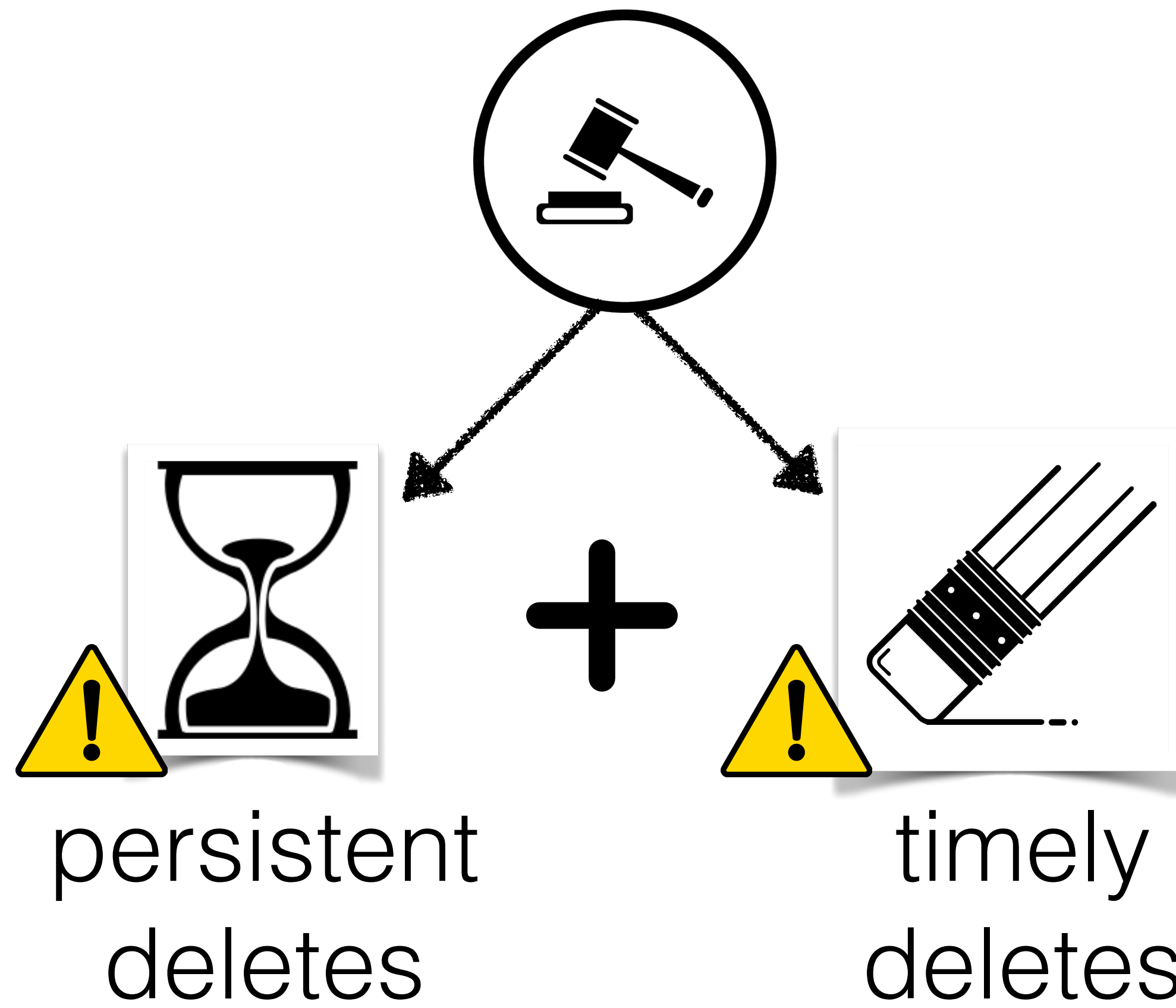
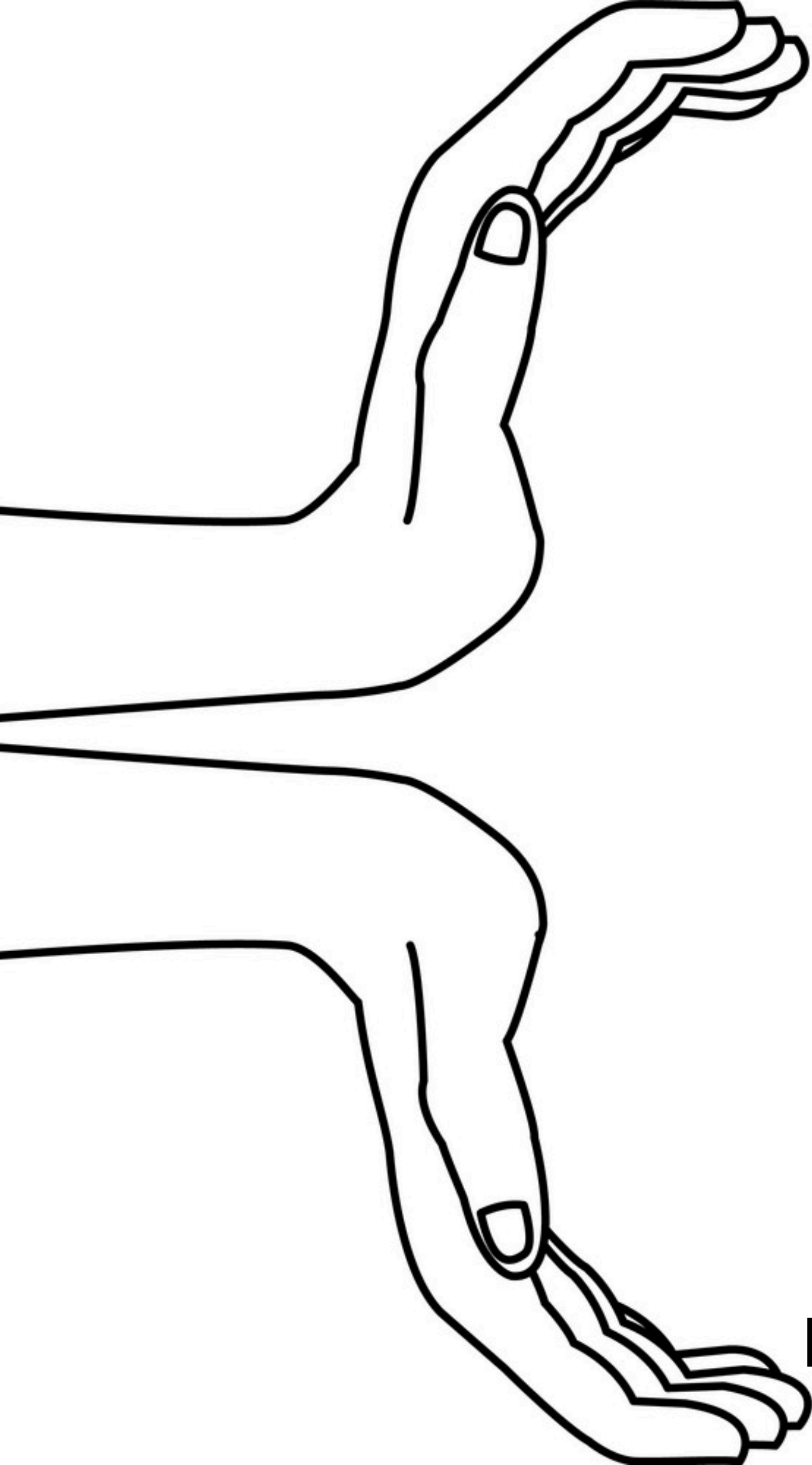
 GDPR
(EU, UK)

 CCPA
(California)

 VCPDA
(Virginia)

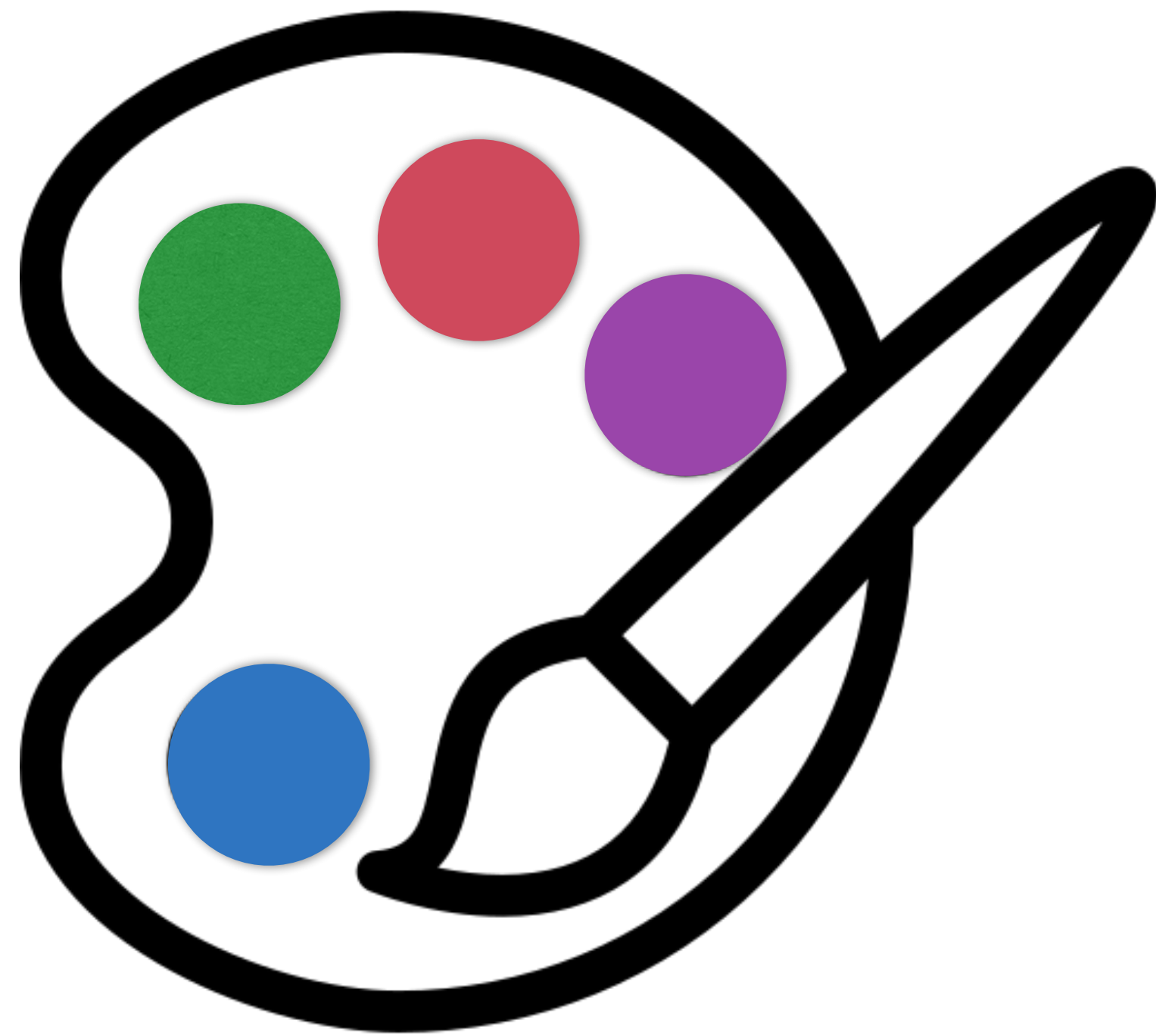
MANDATES





Our **Vision**

PRIVACY-THROUGH-DELETION



- Navigate deletion requirements
- Translate the requirements
- Realize efficient deletes
- Scale with data

Regulations



*Right to be
forgotten*



*Right to
delete*



*Deletion
right*

Regulations

Extracting Requirements



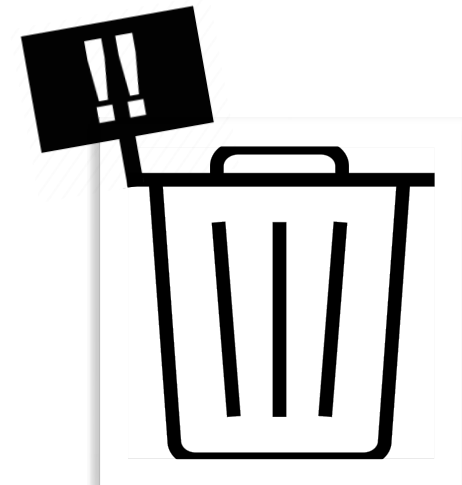
Right to be forgotten



Right to delete



Deletion right



on-demand



retention-based

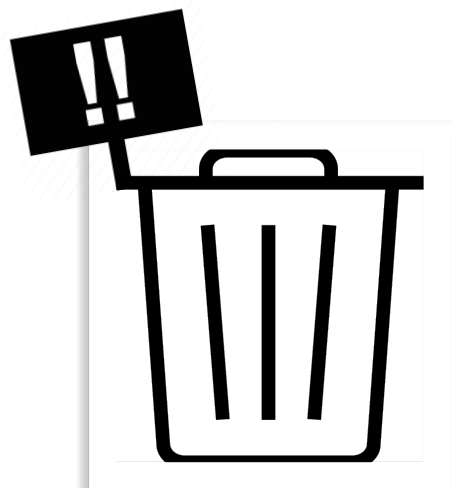
Regulations

 Right to be forgotten

 Right to delete

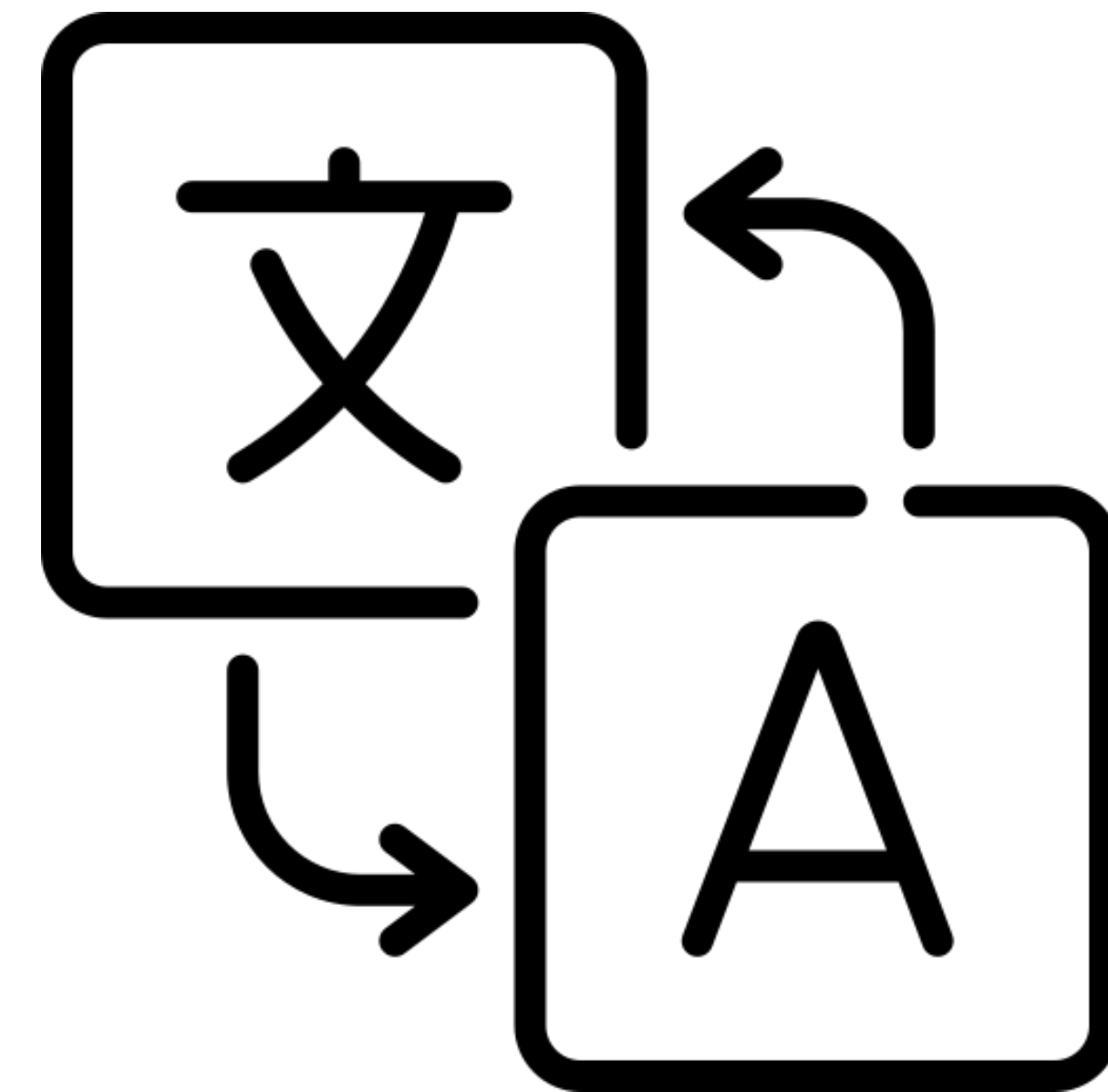
 Deletion right

Extracting Requirements


on-demand


retention-based

Query Language Support



requirement
translation

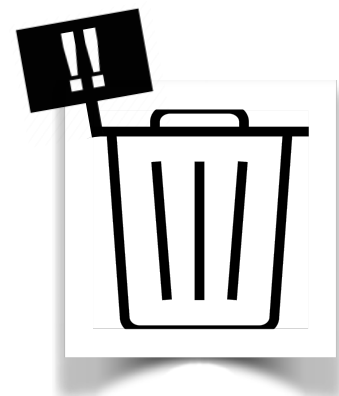
SQL-support for deletes

CREATE TABLE R (column1 type1, column2 type2, ...)



WITH RET_DUR

{ARBITRARY|FIXED (t1 <ret1>, t1 <ret1>, ...)}



WITH DPT

{ARBITRARY|FIXED (d1 <dpt1>, d1 <dpt1>, ...)};

INSERT INTO R (val1, val2, ...)

WITH RET_DUR {<t>|t<i>};

DELETE FROM R

WHERE (...)

WITH DPT {<d>|d<i>};

Regulations



Right to be forgotten

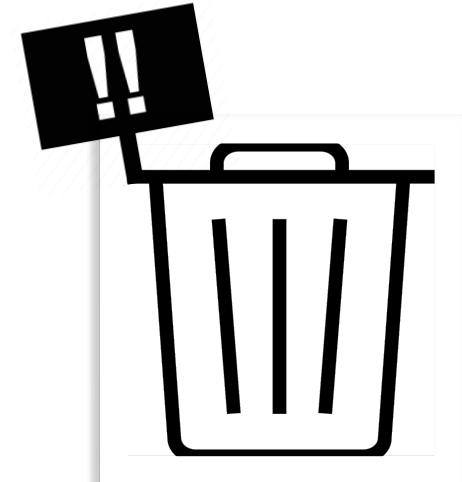


Right to delete



Deletion right

Extracting Requirements



on-demand



retention-based

Query Language Support

```
CREATE TABLE R (...)  
  WITH RET_DUR  
    {ARBITRARY | FIXED(...)}  
  WITH DPT  
    {ARBITRARY | FIXED(...)};
```

```
INSERT INTO R (...)  
  WITH RET_DUR {<t> | t<i>};
```

```
DELETE FROM R  
  WHERE (...)  
  WITH DPT {<d> | d<i>};
```


Regulations



Right to be forgotten

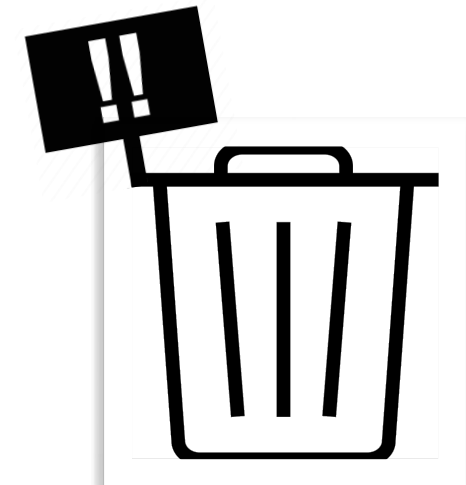


Right to delete



Deletion right

Extracting Requirements



on-demand



retention-based

Query Language Support

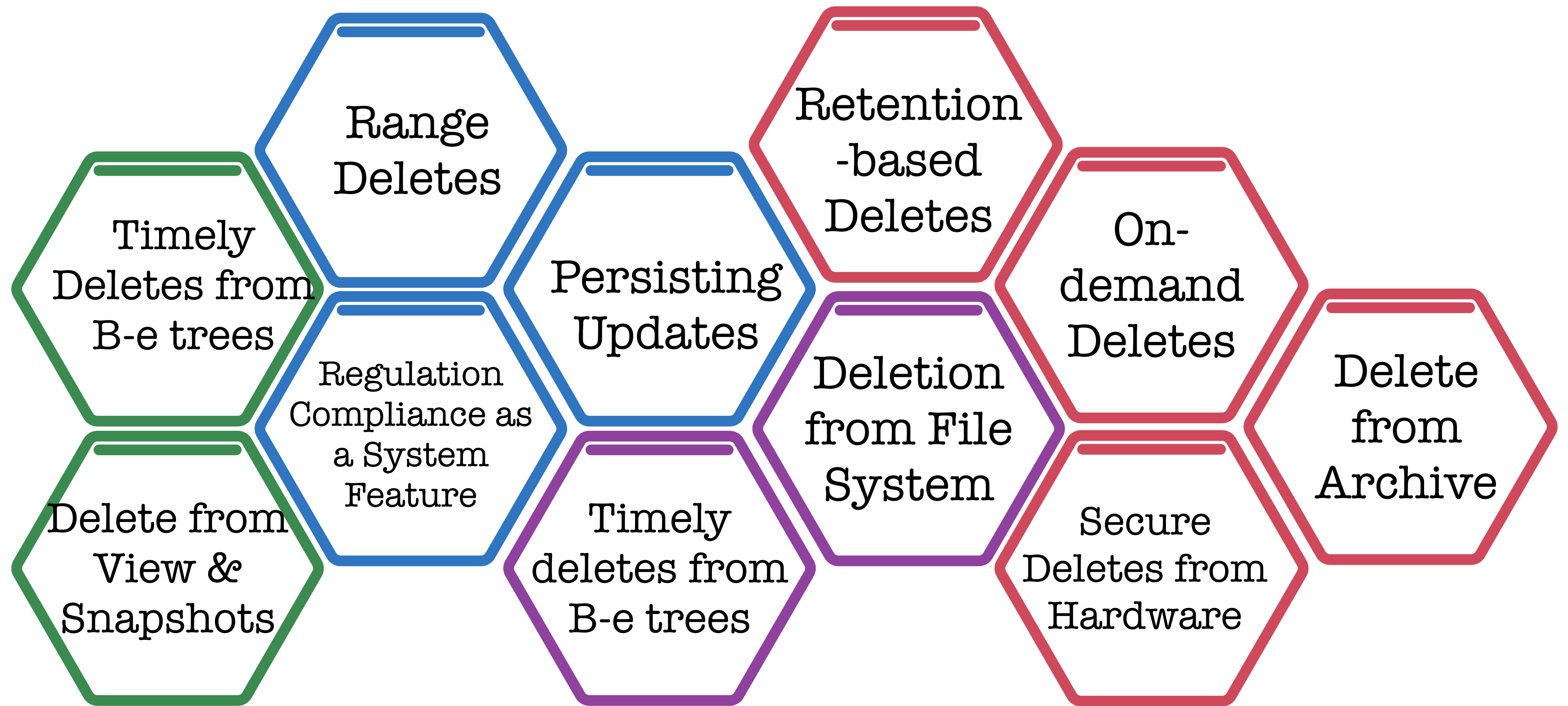
```
CREATE TABLE R (...)  
  WITH RET_DUR  
    {ARBITRARY | FIXED(...)}  
  WITH DPT  
    {ARBITRARY | FIXED(...)};
```

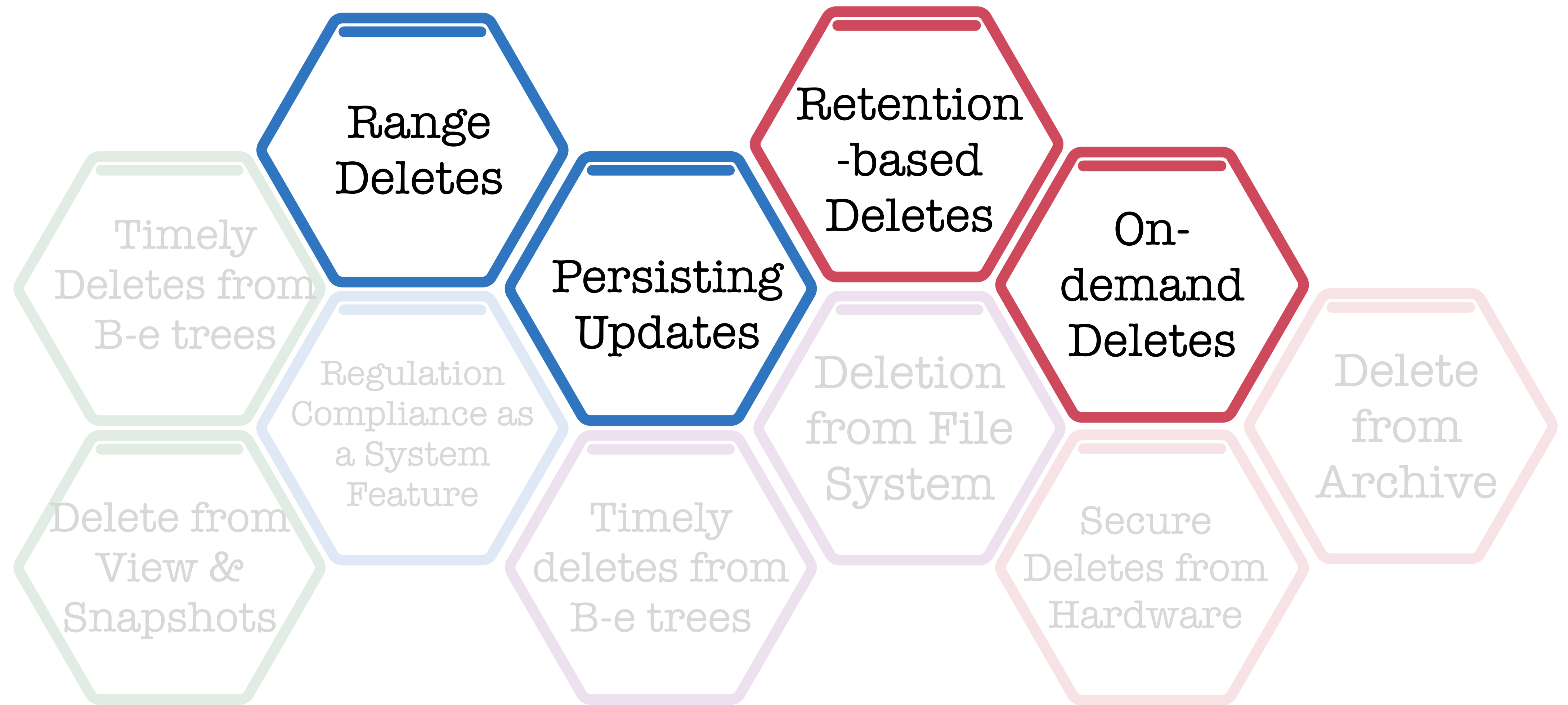
```
INSERT INTO R (...)  
  WITH RET_DUR {<t>|t<i>};
```

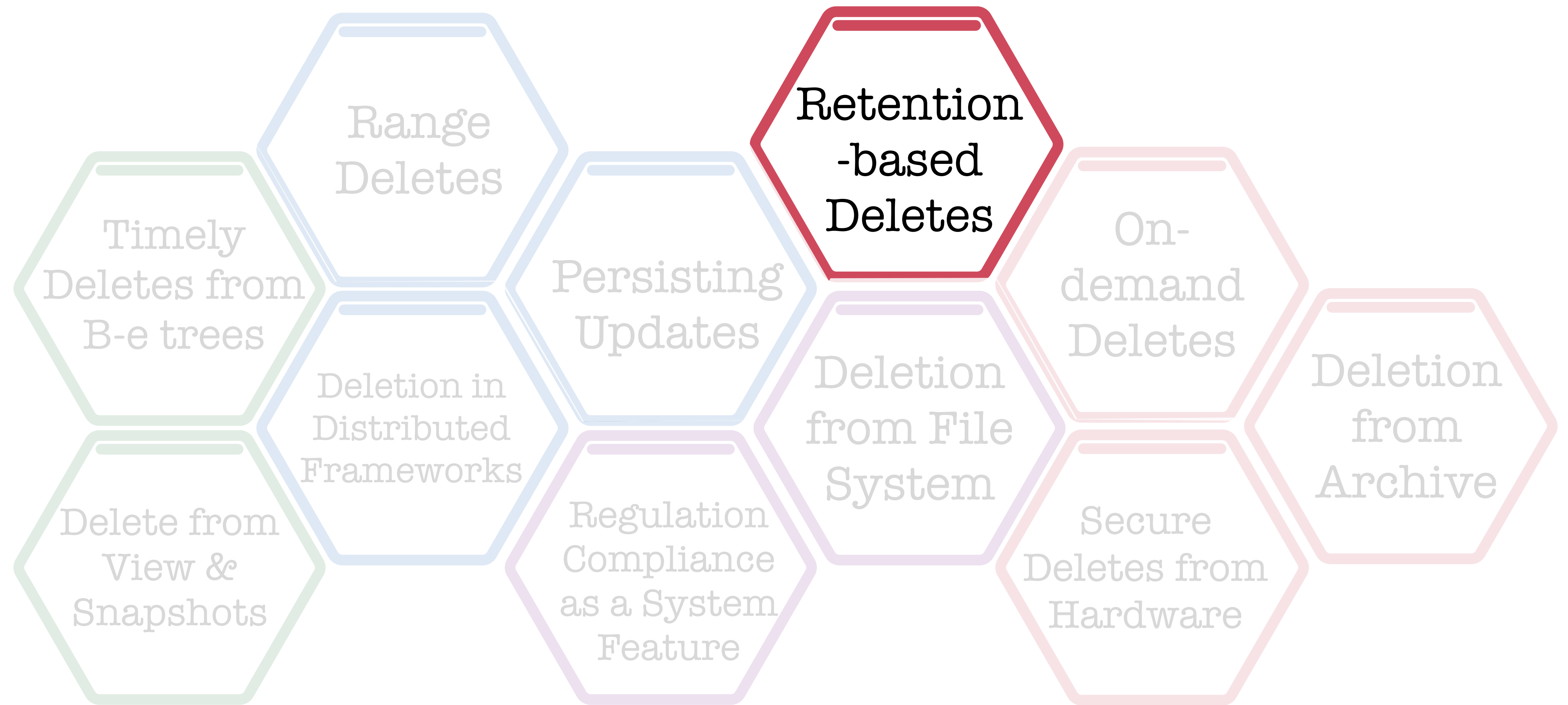
```
DELETE FROM R  
  WHERE (...)  
  WITH DPT {<d>|d<i>};
```

System Support

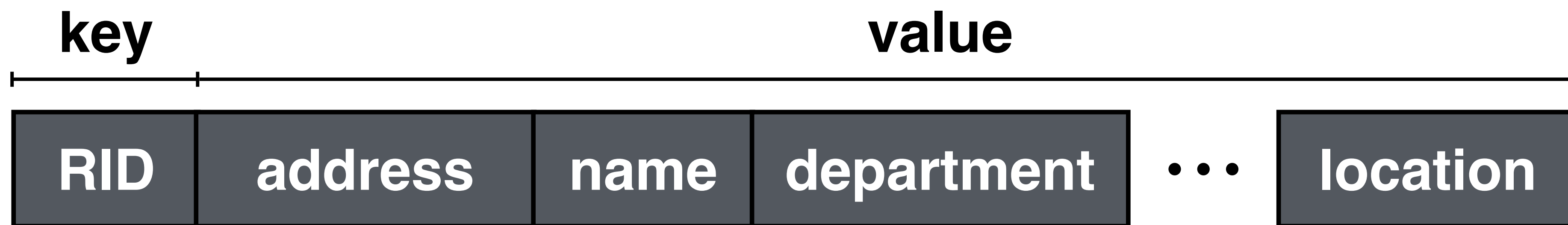


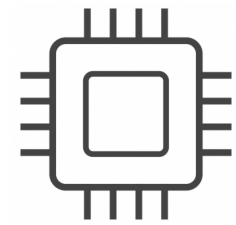






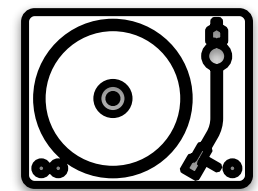
Log-Structured Merge-tree

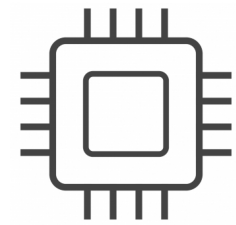




buffer

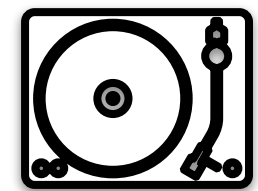
2	6	1	4
---	---	---	---

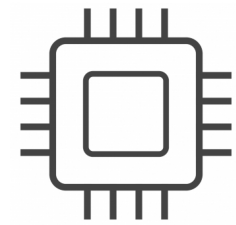




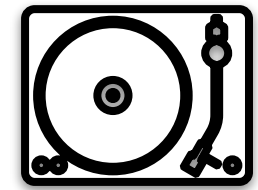
buffer

1	2	4	6
---	---	---	---





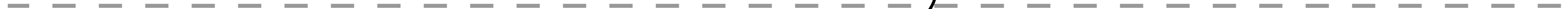
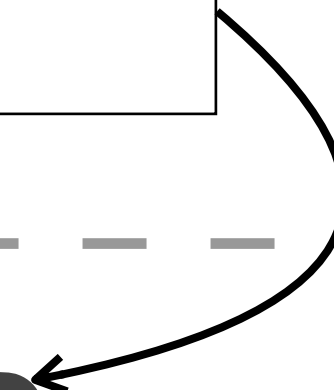
buffer

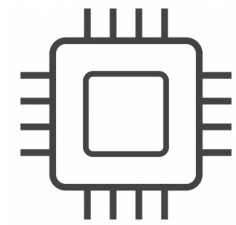


level 1

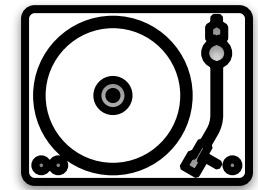
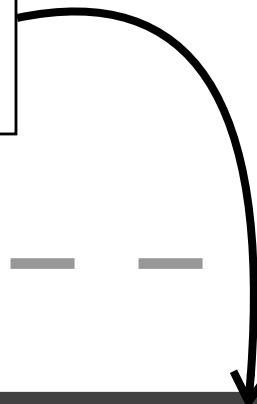
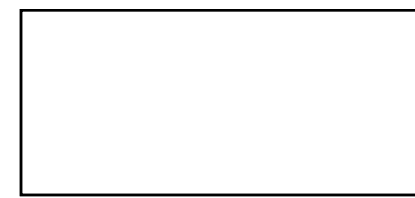


flush



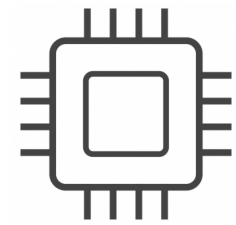


buffer

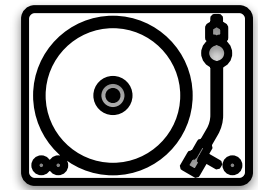


level 1

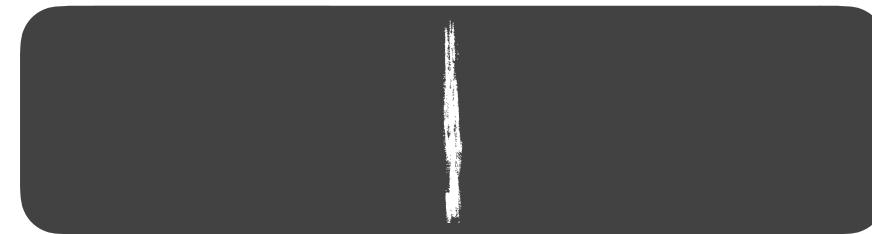




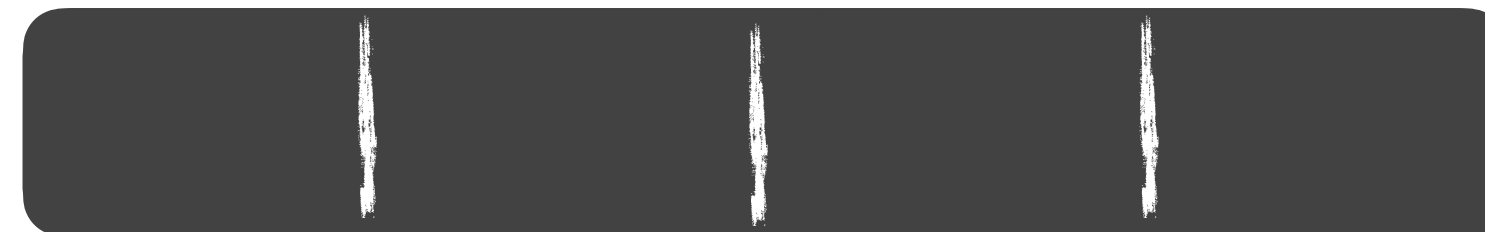
buffer



level 1



level 2



level 3



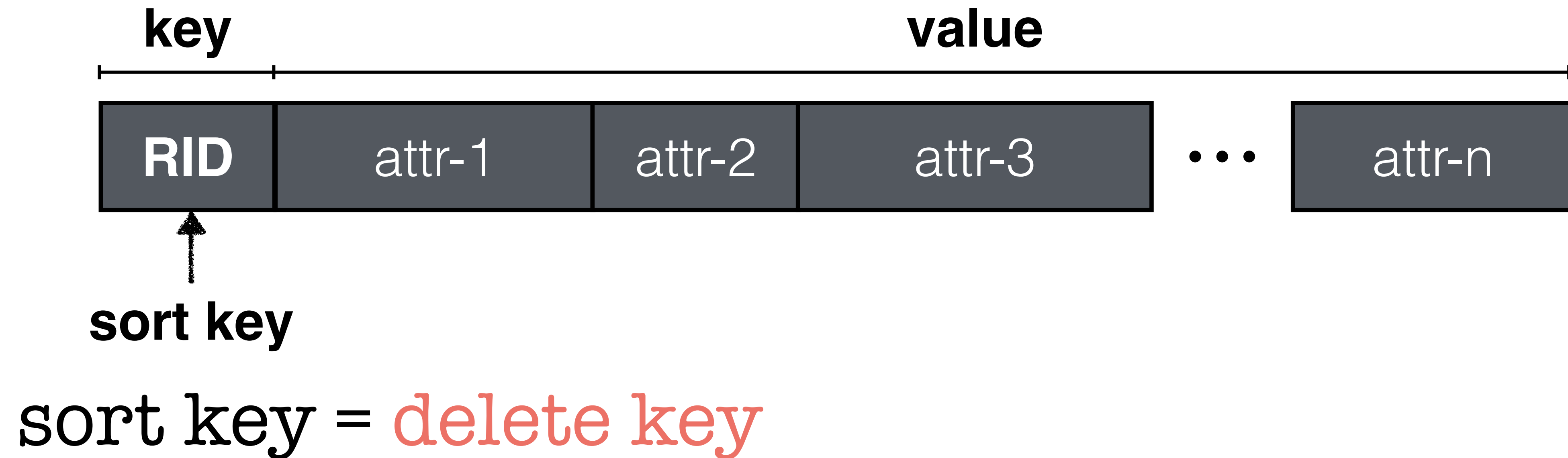
level 4



} sorted out-of-place
component

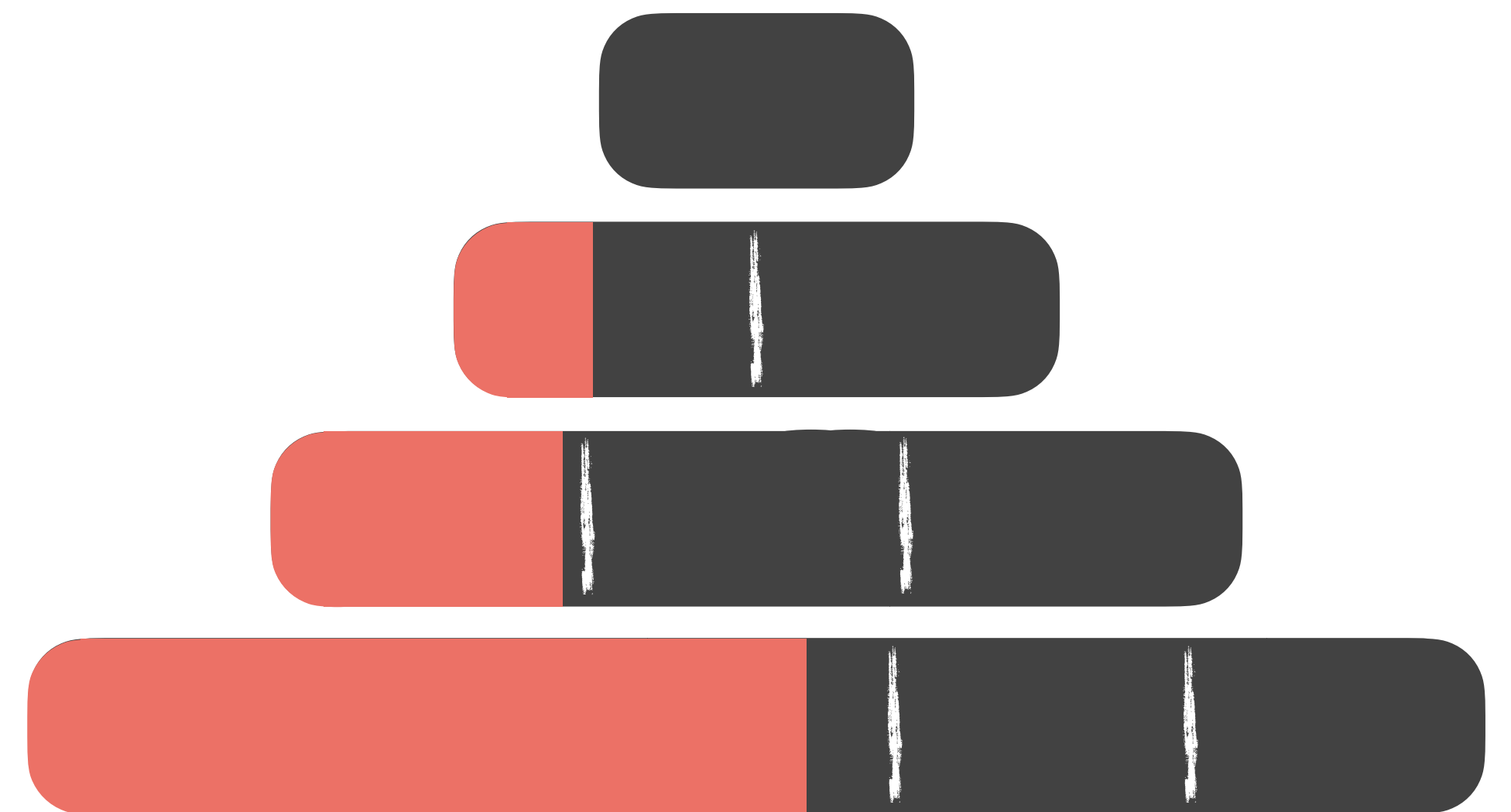
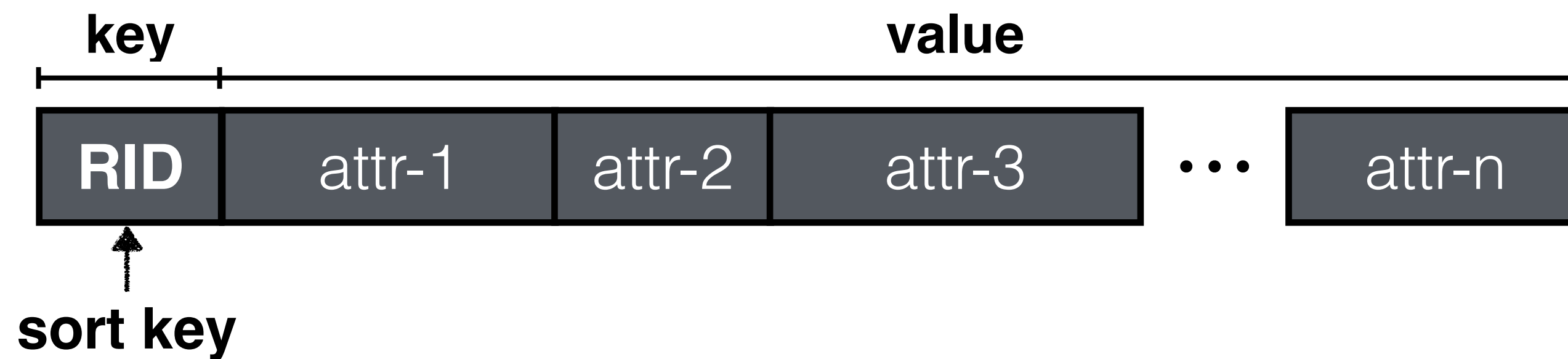
Realizing **Retention-Based Deletes**

delete all entries older than: TS_x



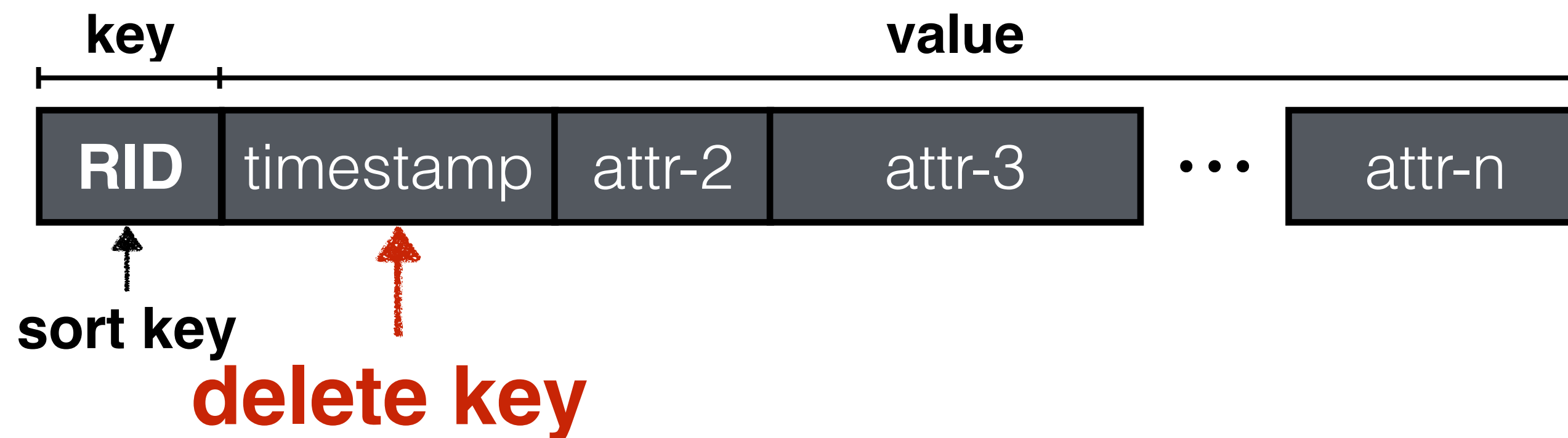
Realizing Retention-Based Deletes

delete all entries older than: TS_x

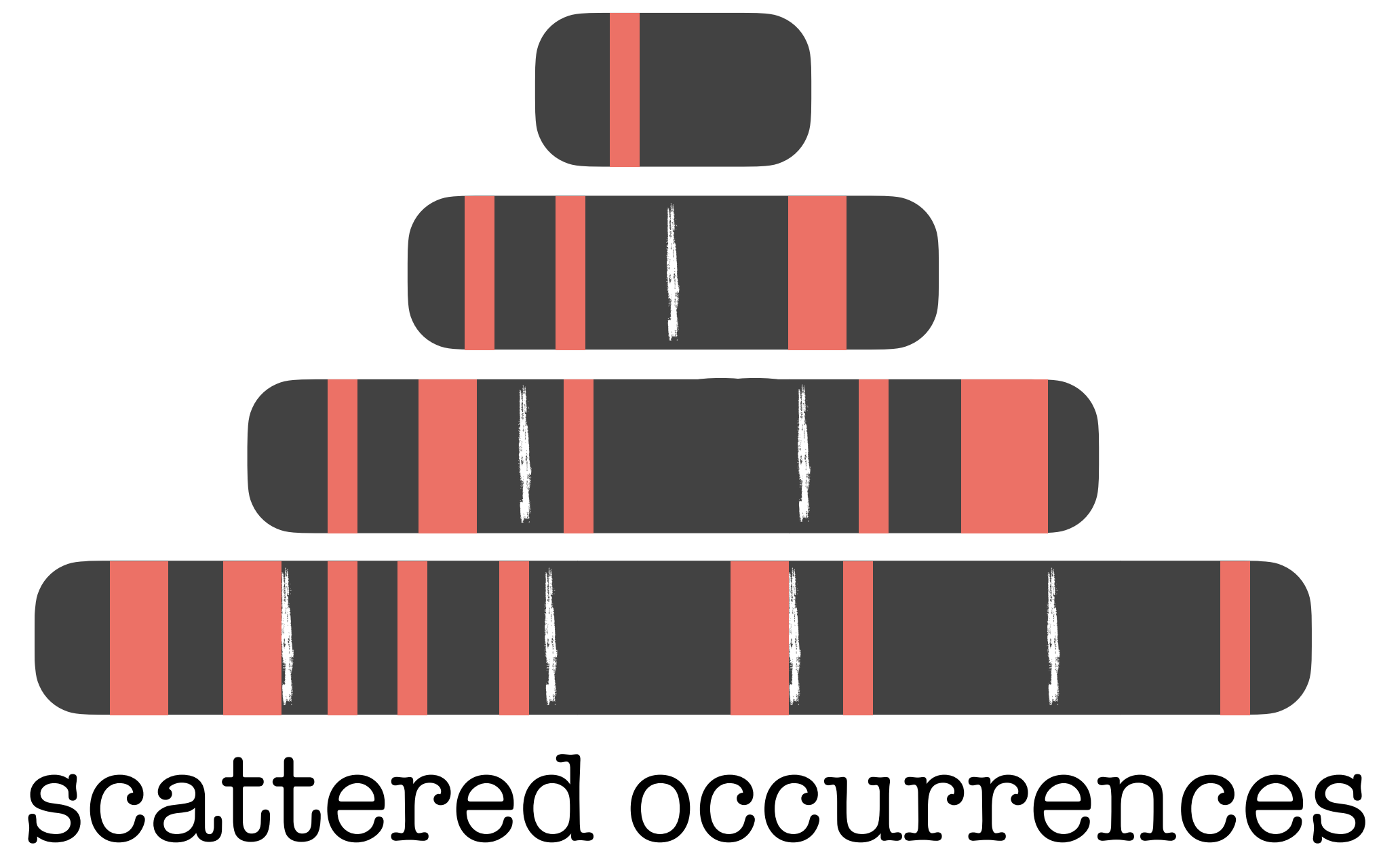


Realizing Retention-Based Deletes

delete all entries older than: TS_x

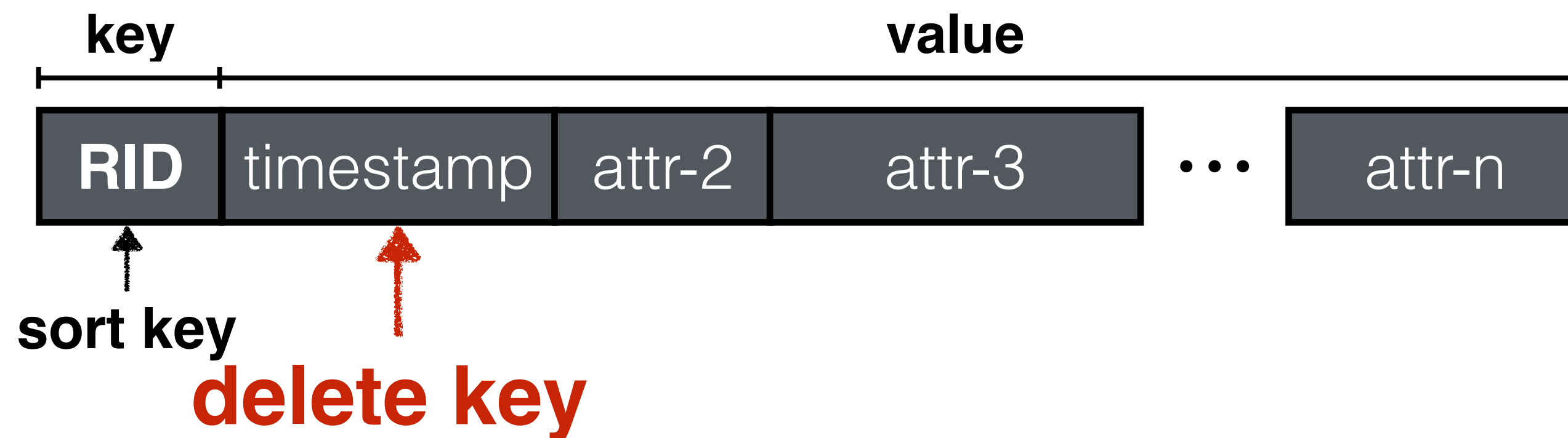


sort key $\neq$ delete key

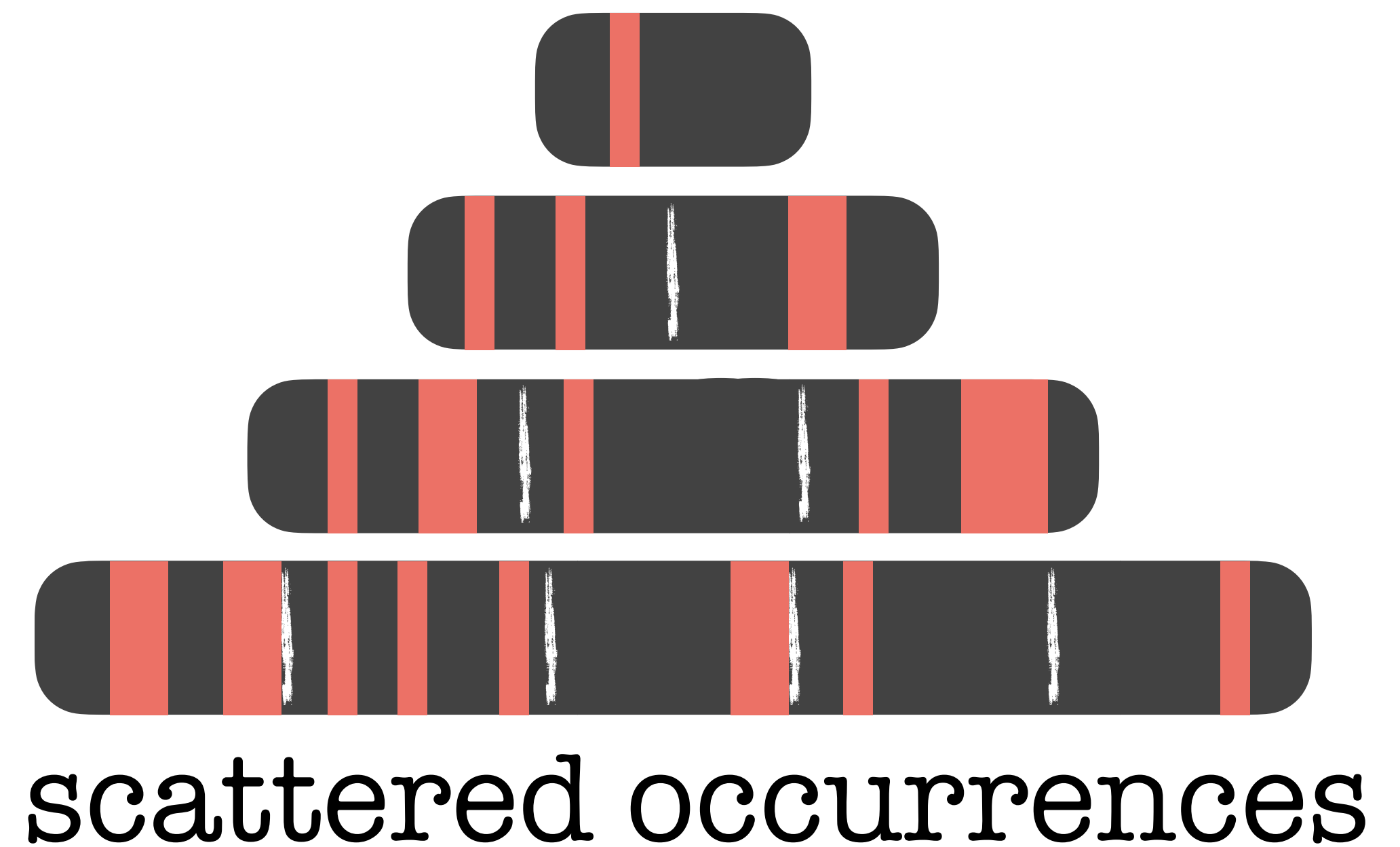


Realizing Retention-Based Deletes

delete all entries older than: TS_x

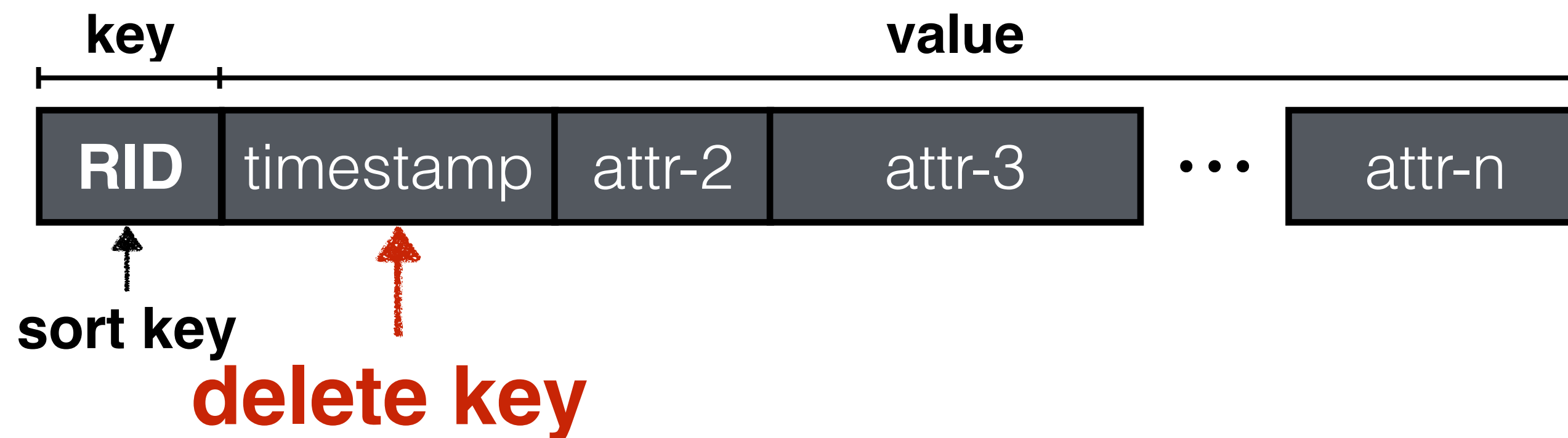


sort key $\neq$ delete key



Realizing Retention-Based Deletes

delete all entries older than: TS_x

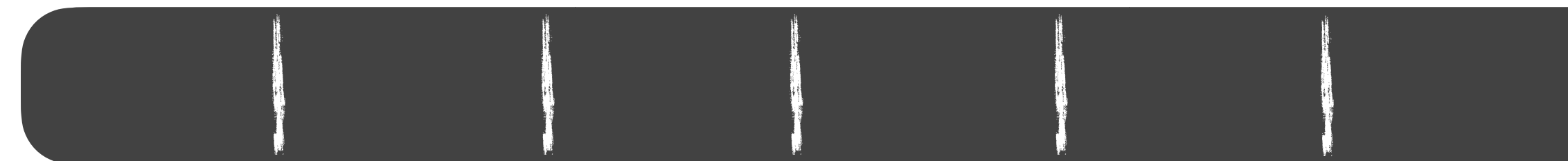


sort key $\neq$ delete key

latency spikes



superfluous I/Os

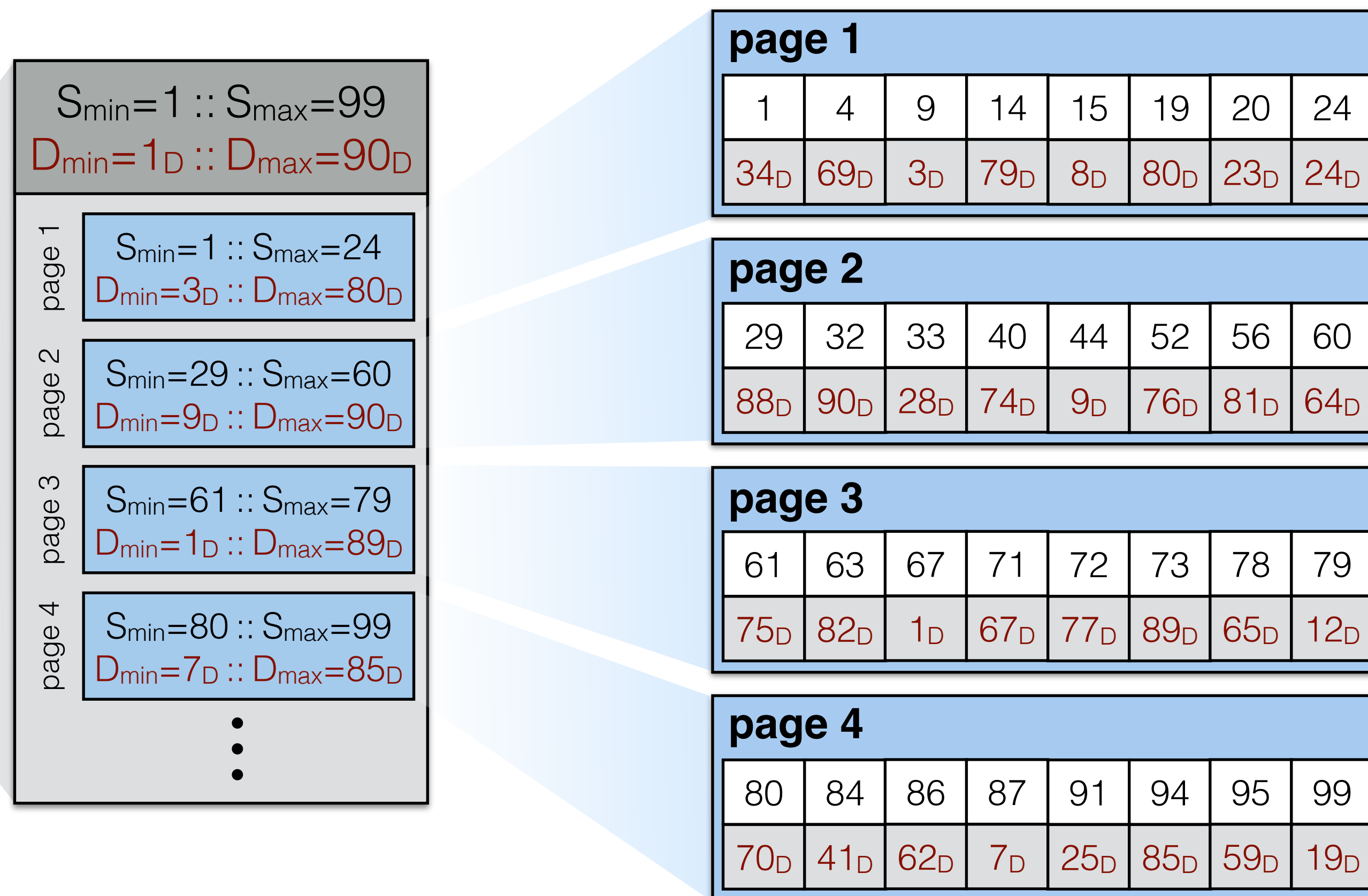


Realizing Retention-Based Deletes

delete all entries older than $\leq 65_D$



SST file

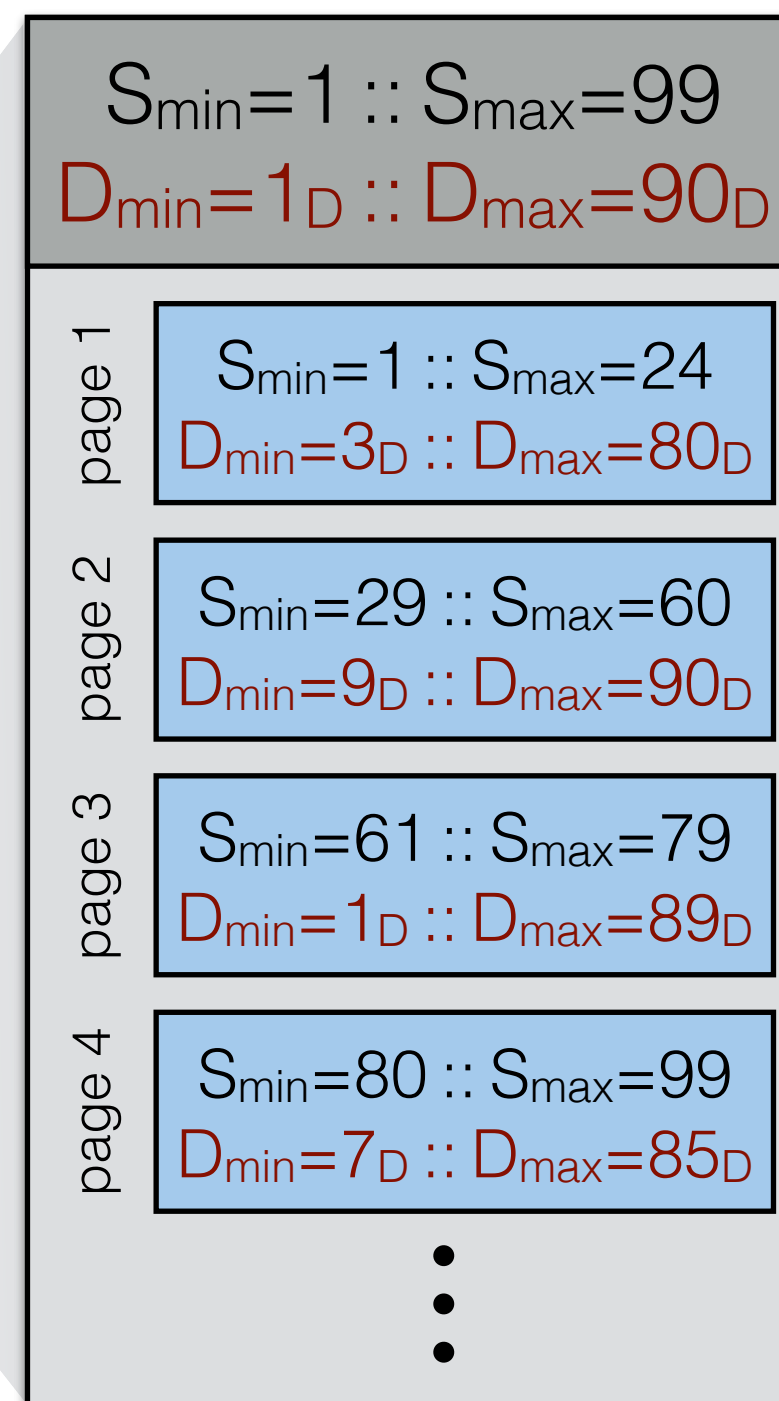


Realizing Retention-Based Deletes

delete all entries older than $\leq 65_D$



SST file



page 1

1	4	9	14	15	19	20	24
34 _D	69 _D	3 _D	79 _D	8 _D	80 _D	23 _D	24 _D

1 I/O

page 2

29	32	33	40	44	52	56	60
88 _D	90 _D	28 _D	74 _D	9 _D	76 _D	81 _D	64 _D

1 I/O

page 3

61	63	67	71	72	73	78	79
75 _D	82 _D	1 _D	67 _D	77 _D	89 _D	65 _D	12 _D

1 I/O

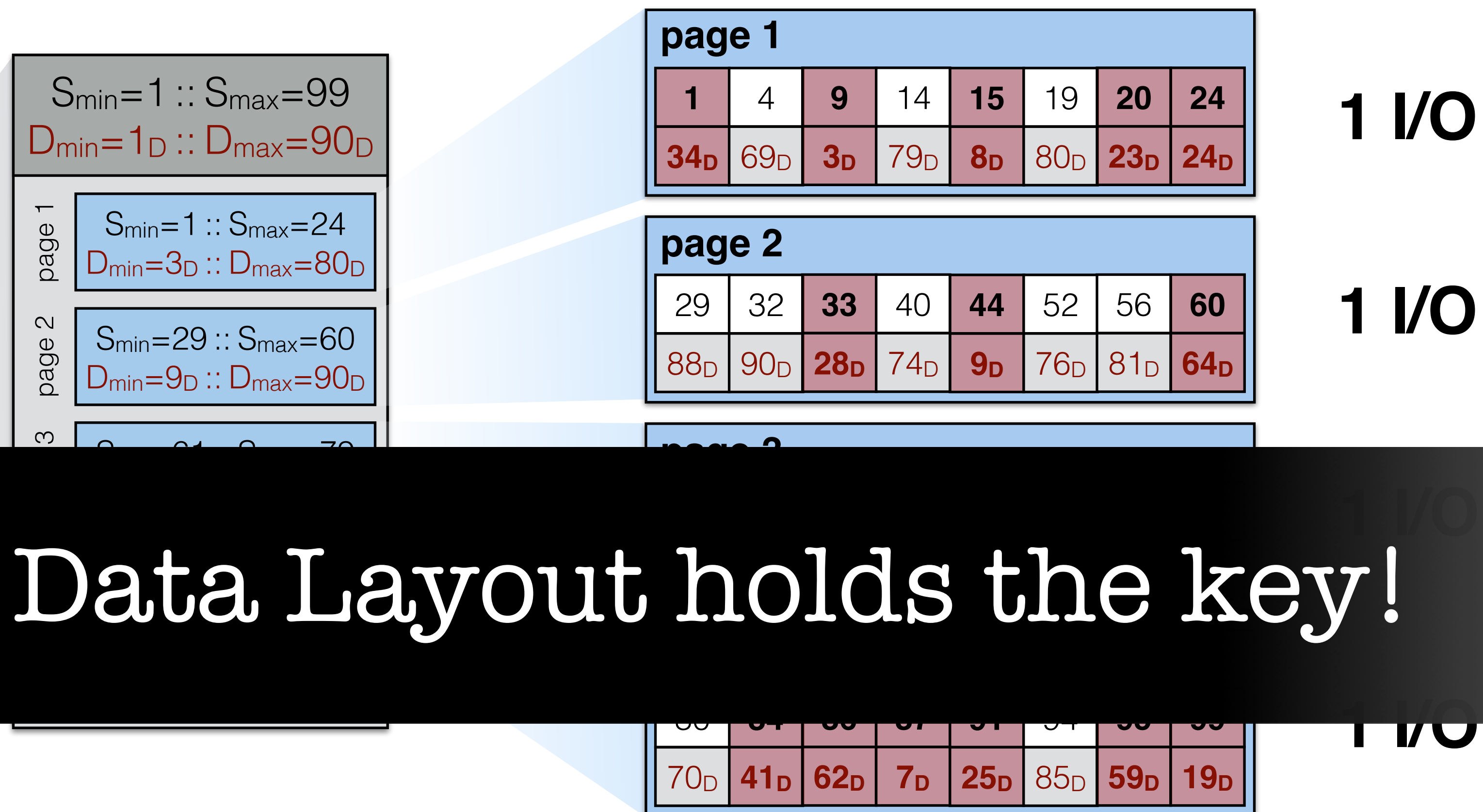
page 4

80	84	86	87	91	94	95	99
70 _D	41 _D	62 _D	7 _D	25 _D	85 _D	59 _D	19 _D

1 I/O

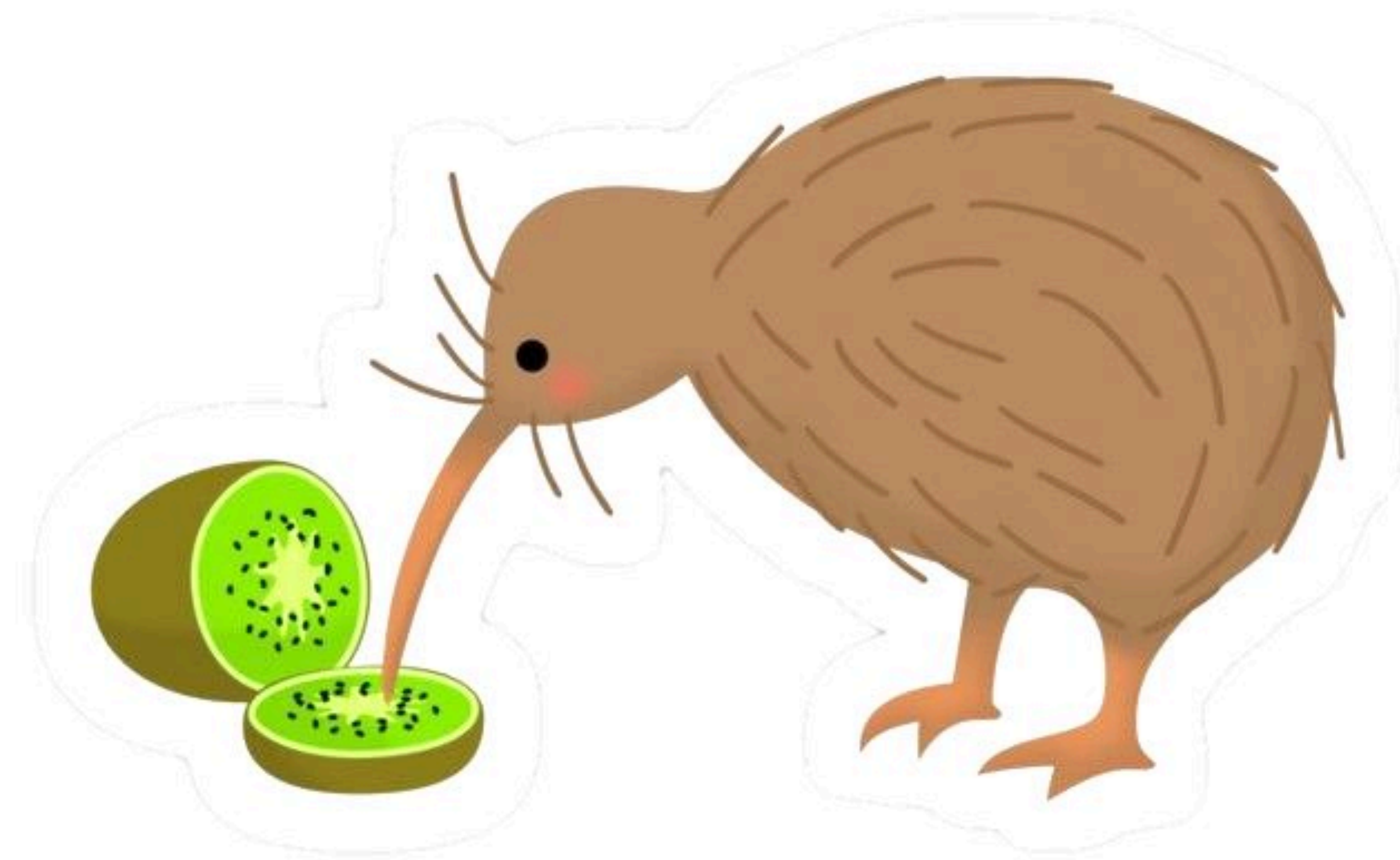
Realizing Retention-Based Deletes

delete all entries older than $\leq 65_D$



Data Layout holds the key!

Realizing **Retention-Based Deletes**



KiWi

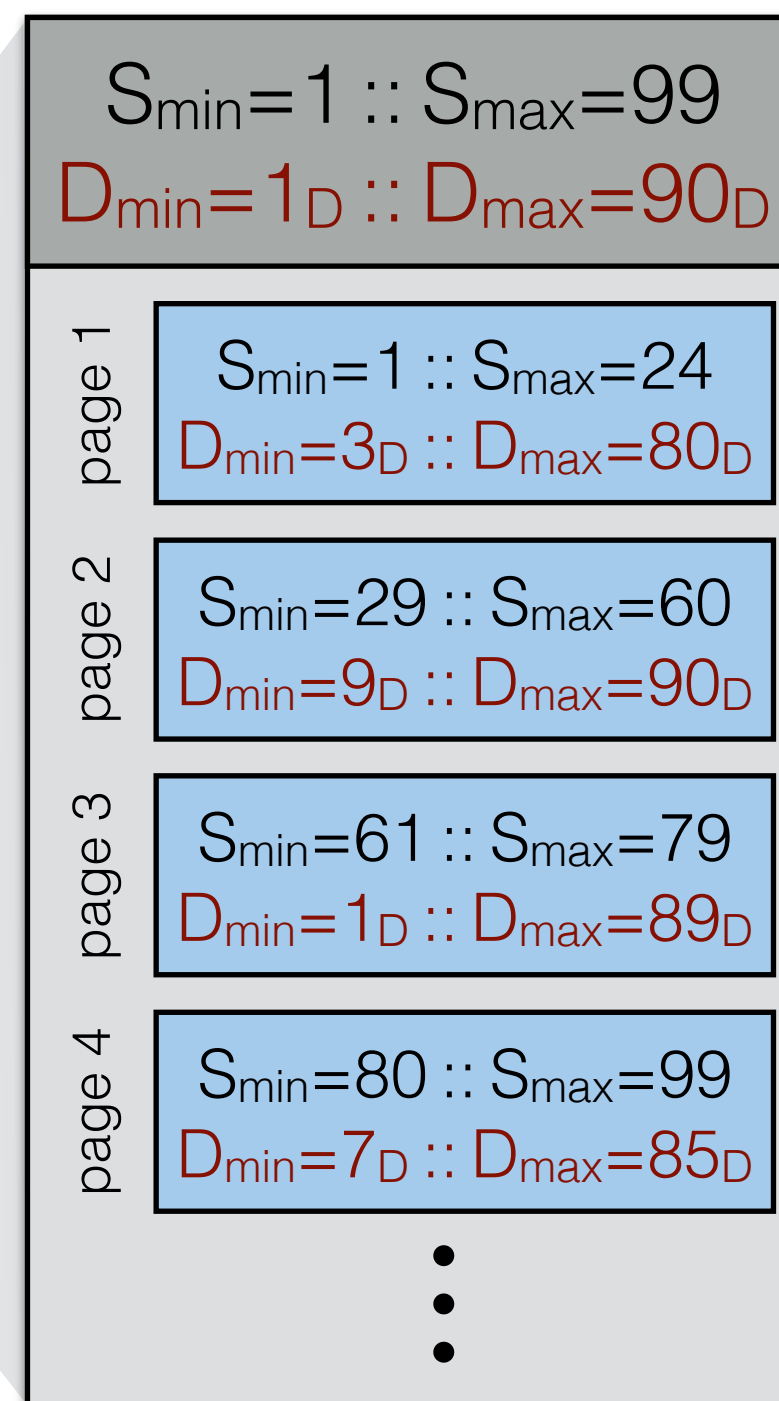
Key Weaving storage layout

Key Weaving storage layout

delete all entries older than $\leq 65_D$



SST file

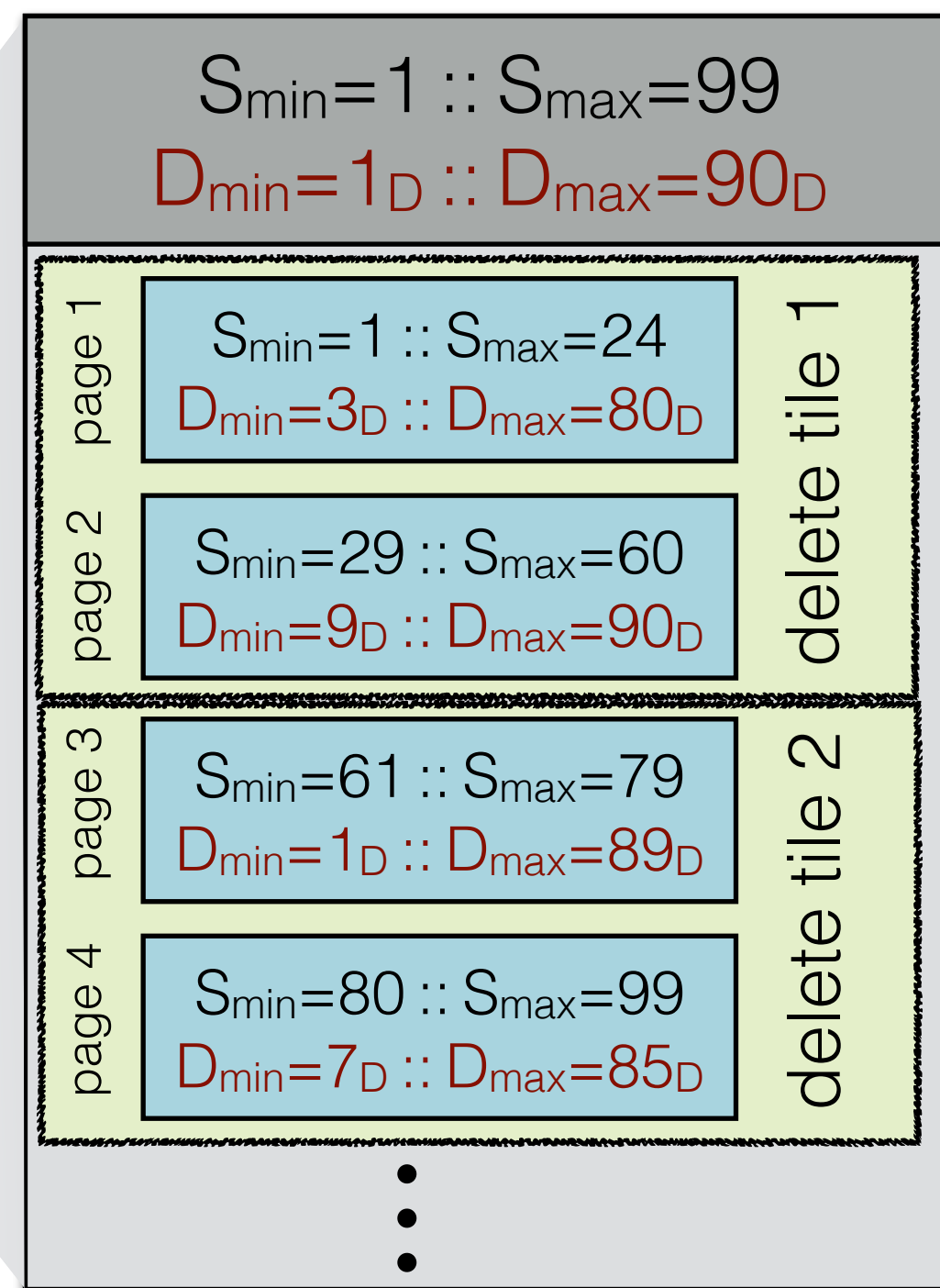


Key Weaving storage layout

delete all entries older than $\leq 65_D$



SST file



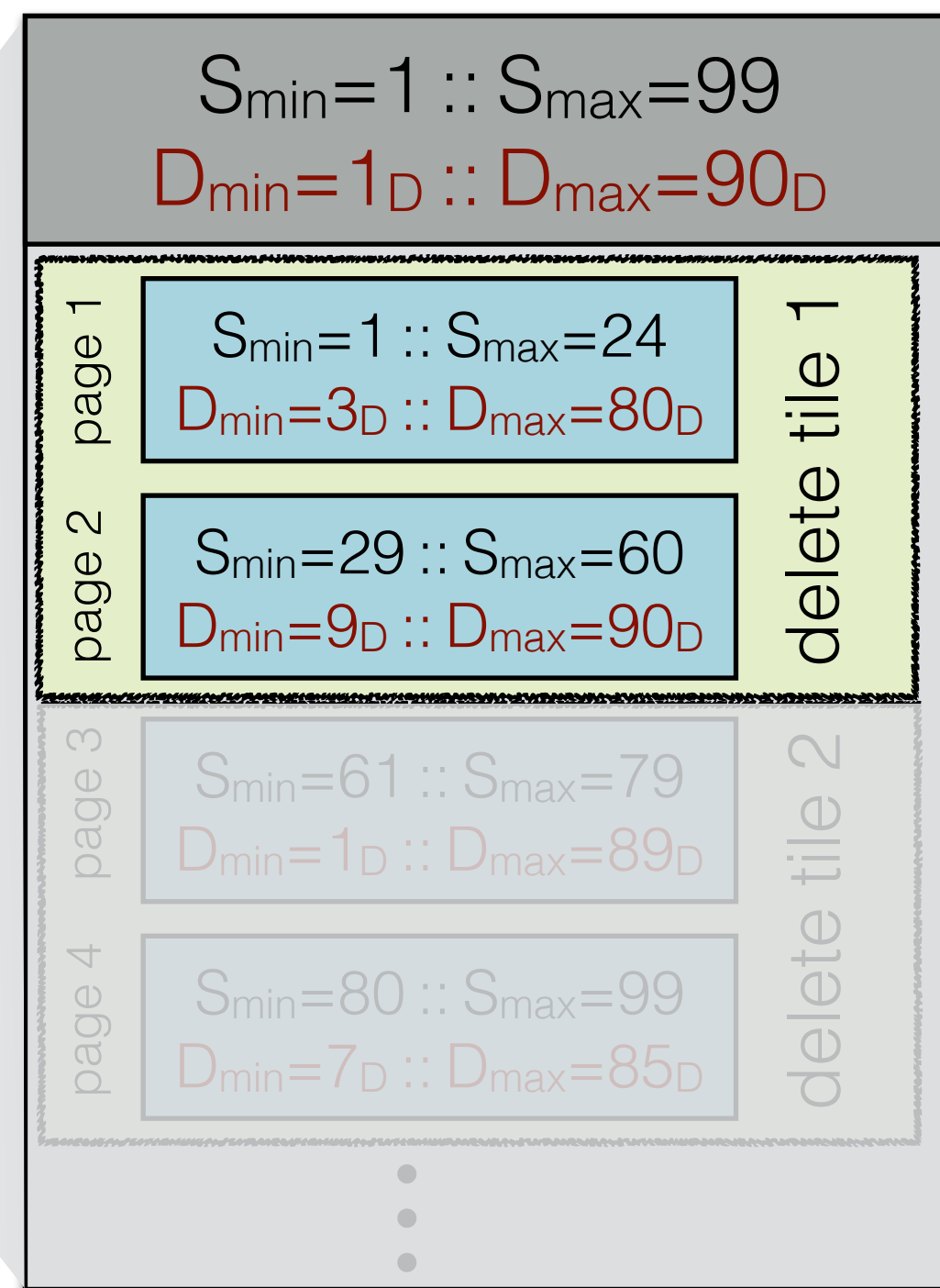
partitioned on S

Key Weaving storage layout

delete all entries older than $\leq 65_D$



SST file



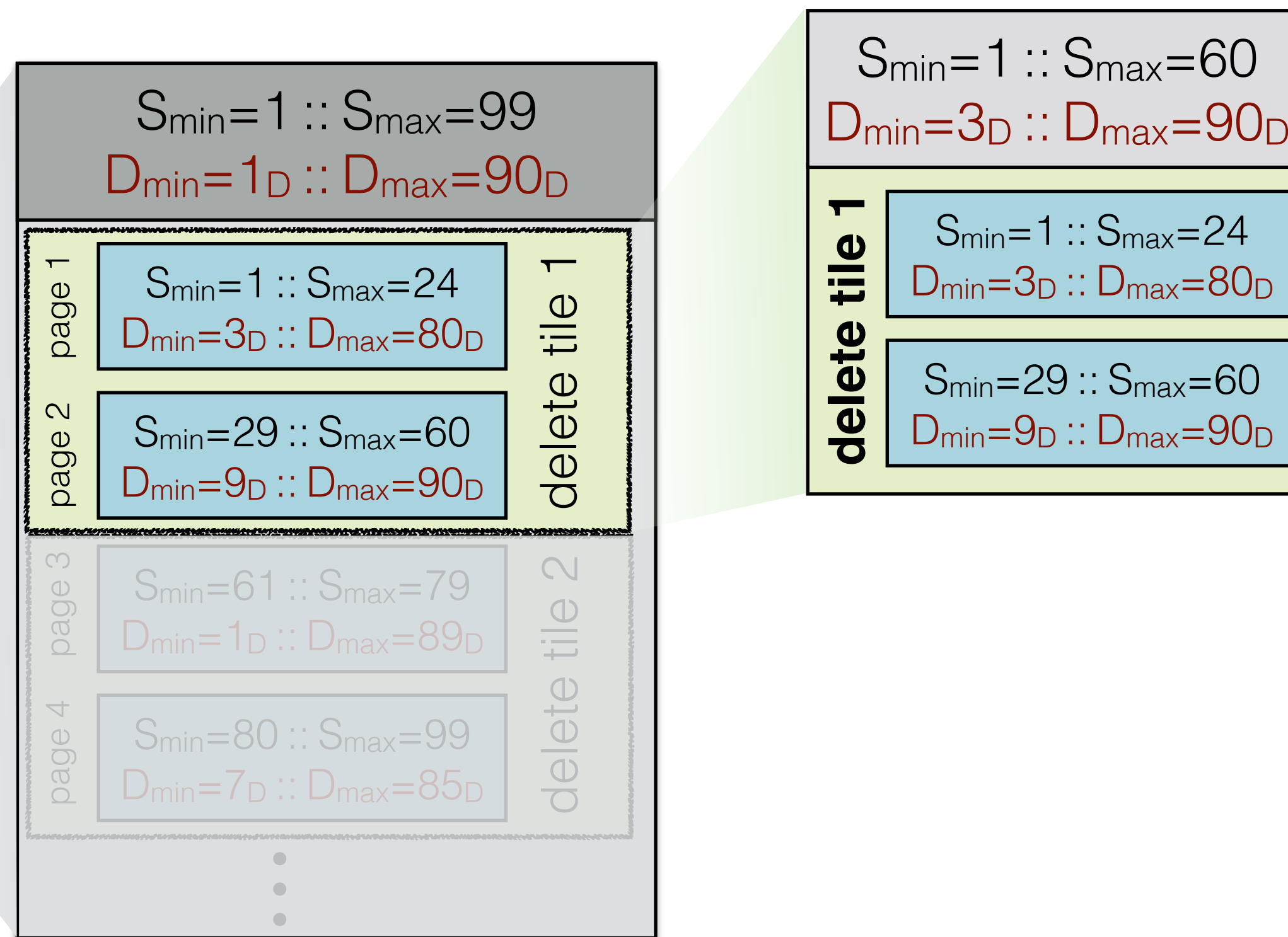
partitioned on S

Key Weaving storage layout

delete all entries older than $\leq 65D$



SST file



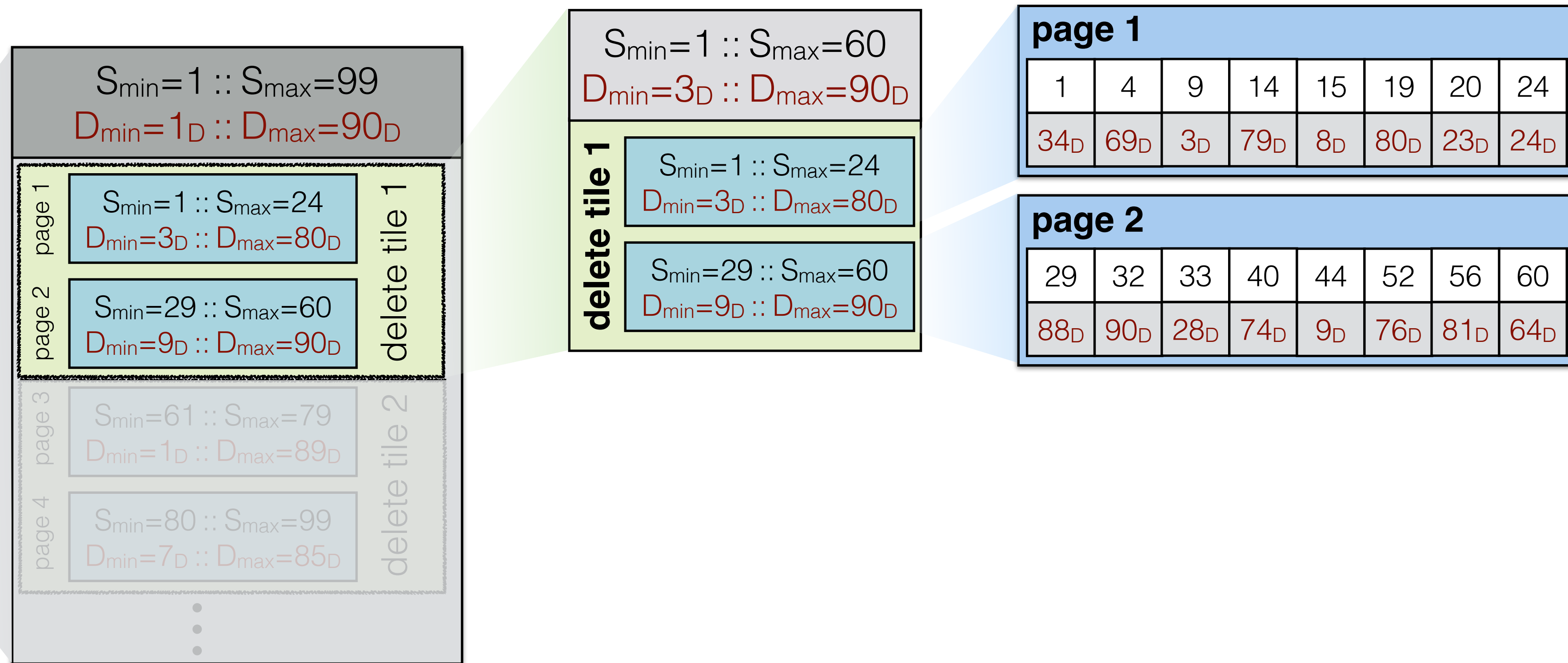
partitioned on S

Key Weaving storage layout

delete all entries older than $\leq 65_D$



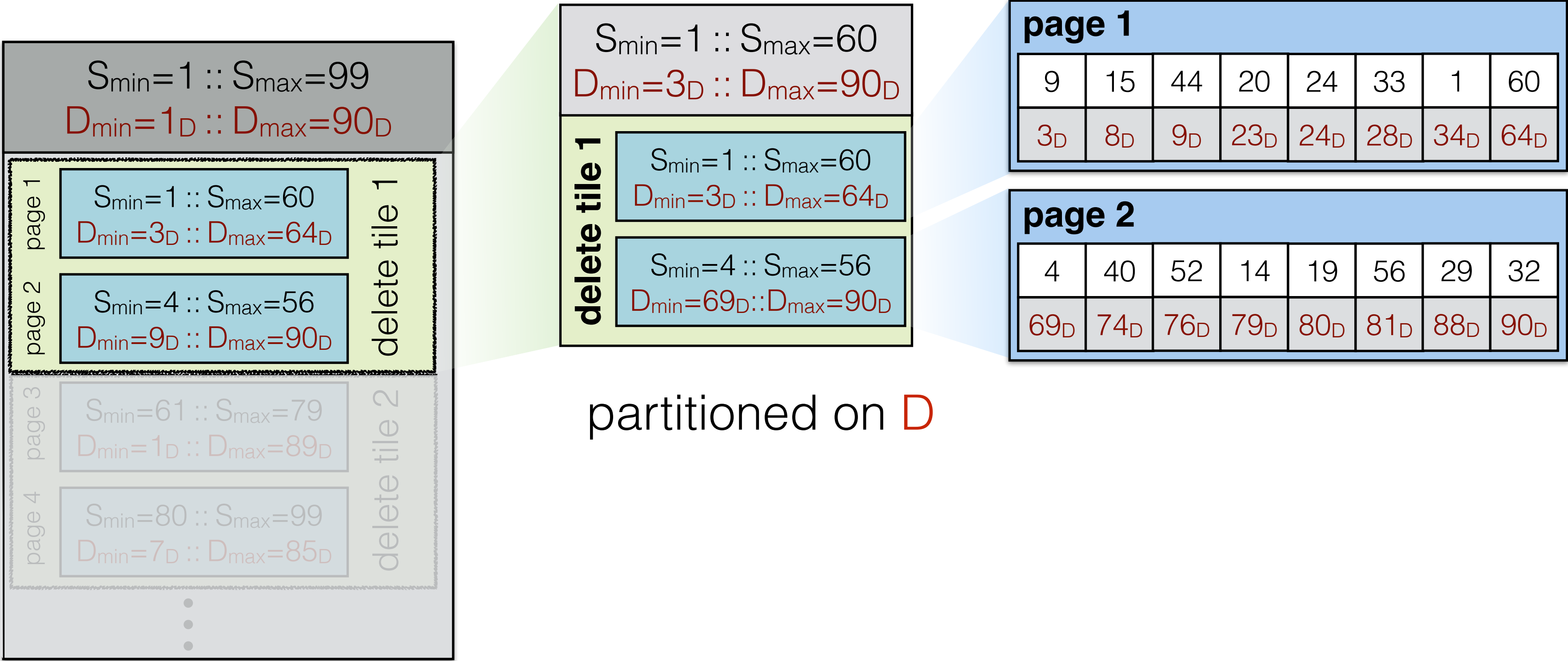
SST file



partitioned on S

Key Weaving storage layout

delete all entries older than $\leq 65_D$

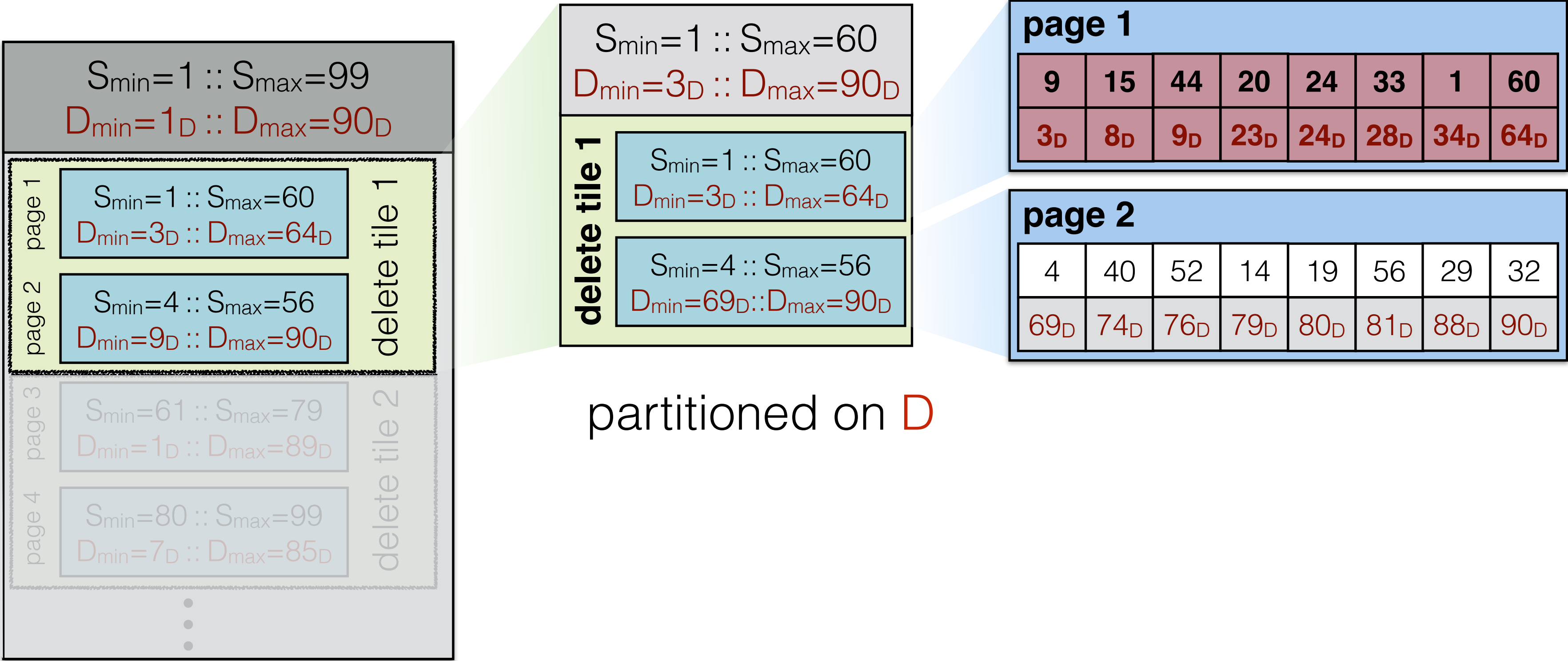


partitioned on D

partitioned on S

Key Weaving storage layout

delete all entries older than $\leq 65_D$



drop
page

partitioned on D

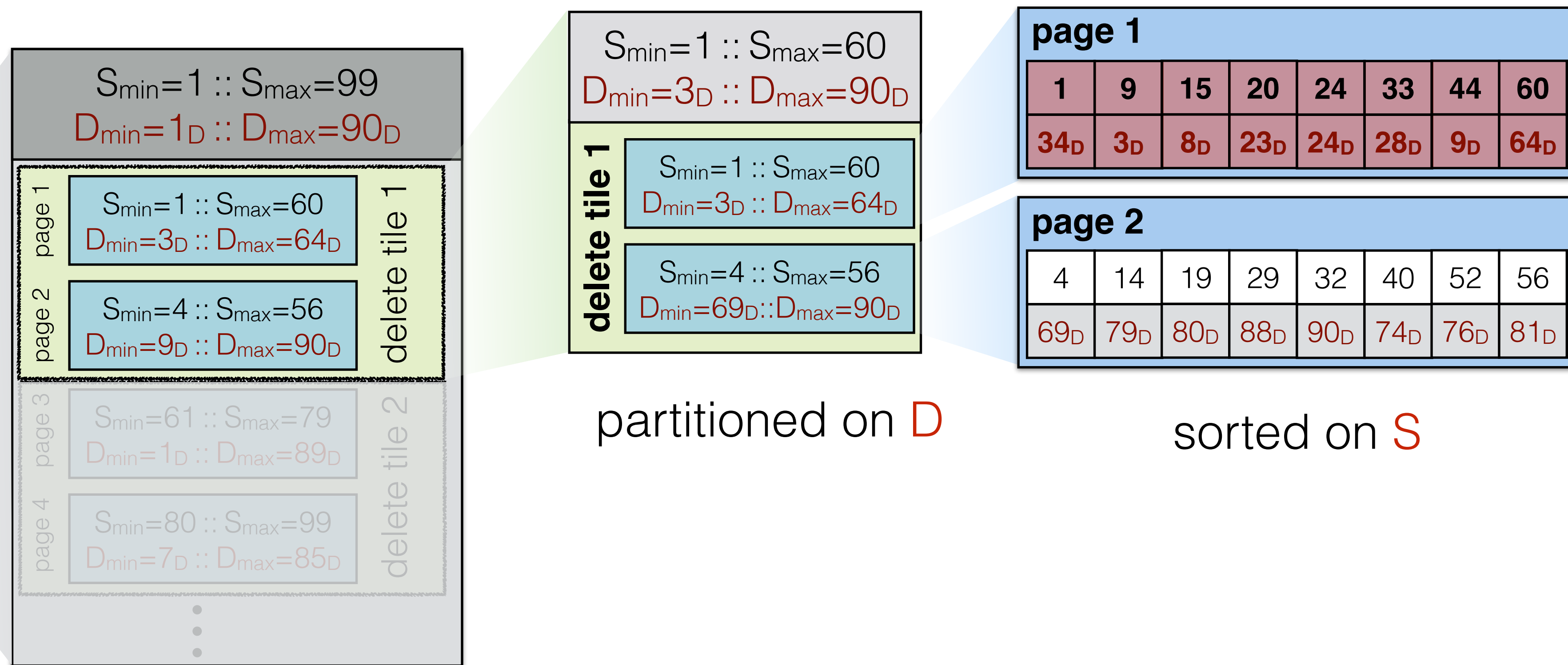
partitioned on S

Key Weaving storage layout

delete all entries older than $\leq 65_D$



SST file



drop
page

partitioned on D

sorted on S

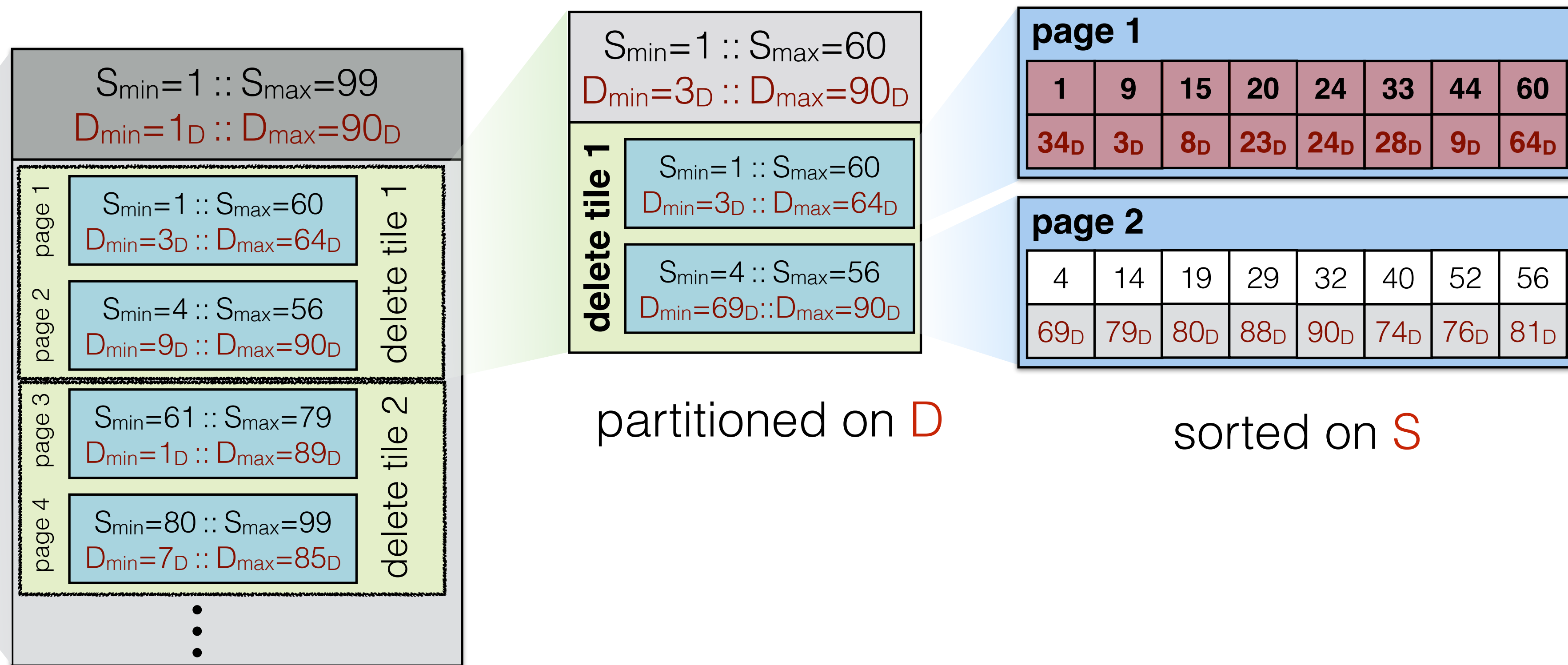
partitioned on S

Key Weaving storage layout

delete all entries older than $\leq 65_D$



SST file



partitioned on S

partitioned on D

sorted on S

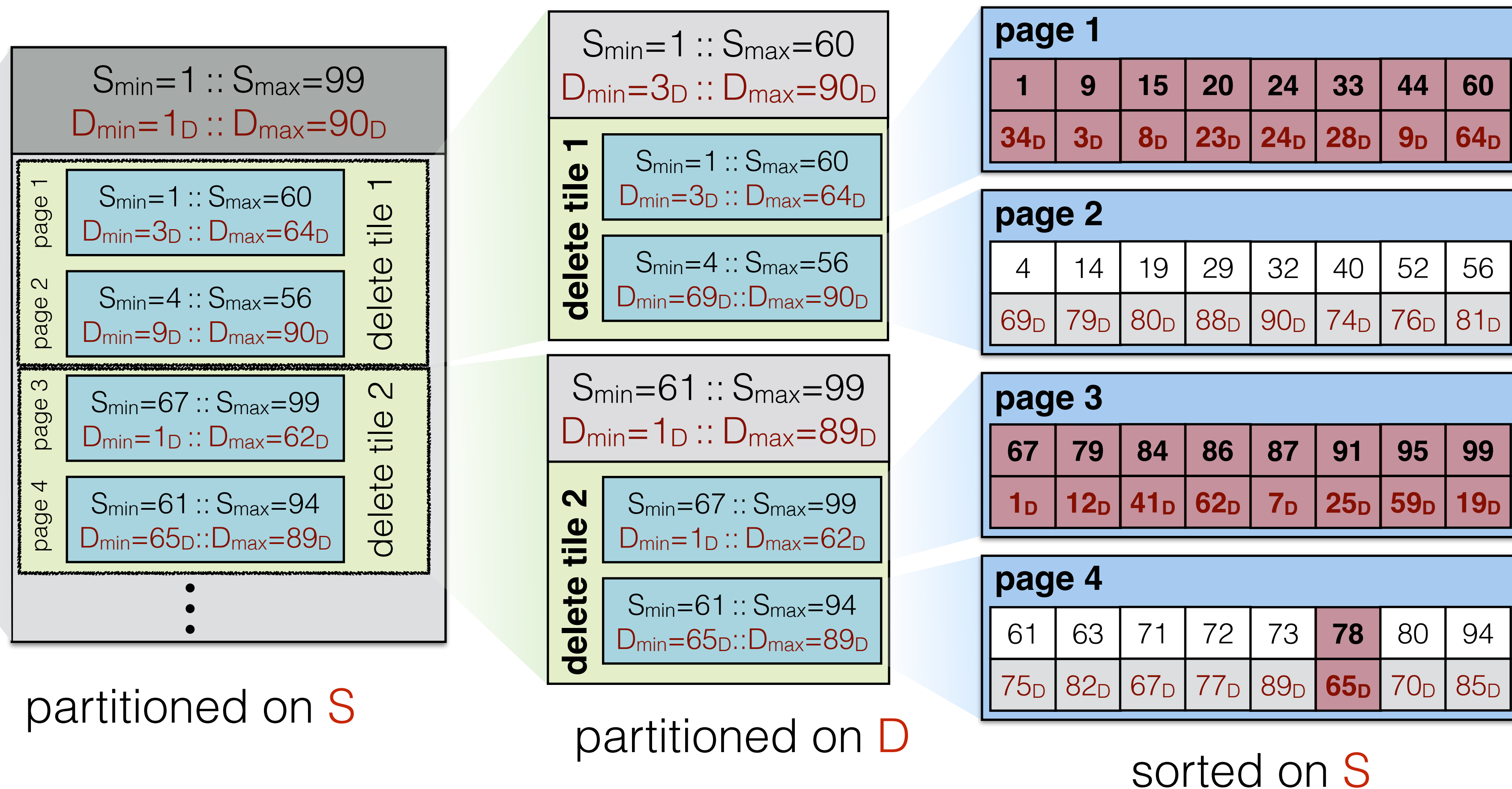
drop
page

Key Weaving storage layout

delete all entries older than $\leq 65_D$

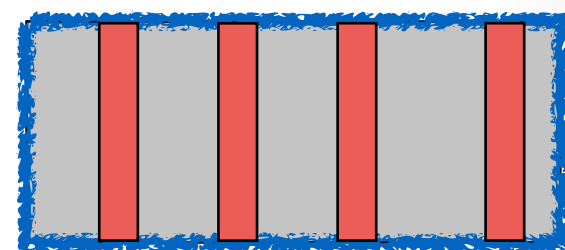


SST file

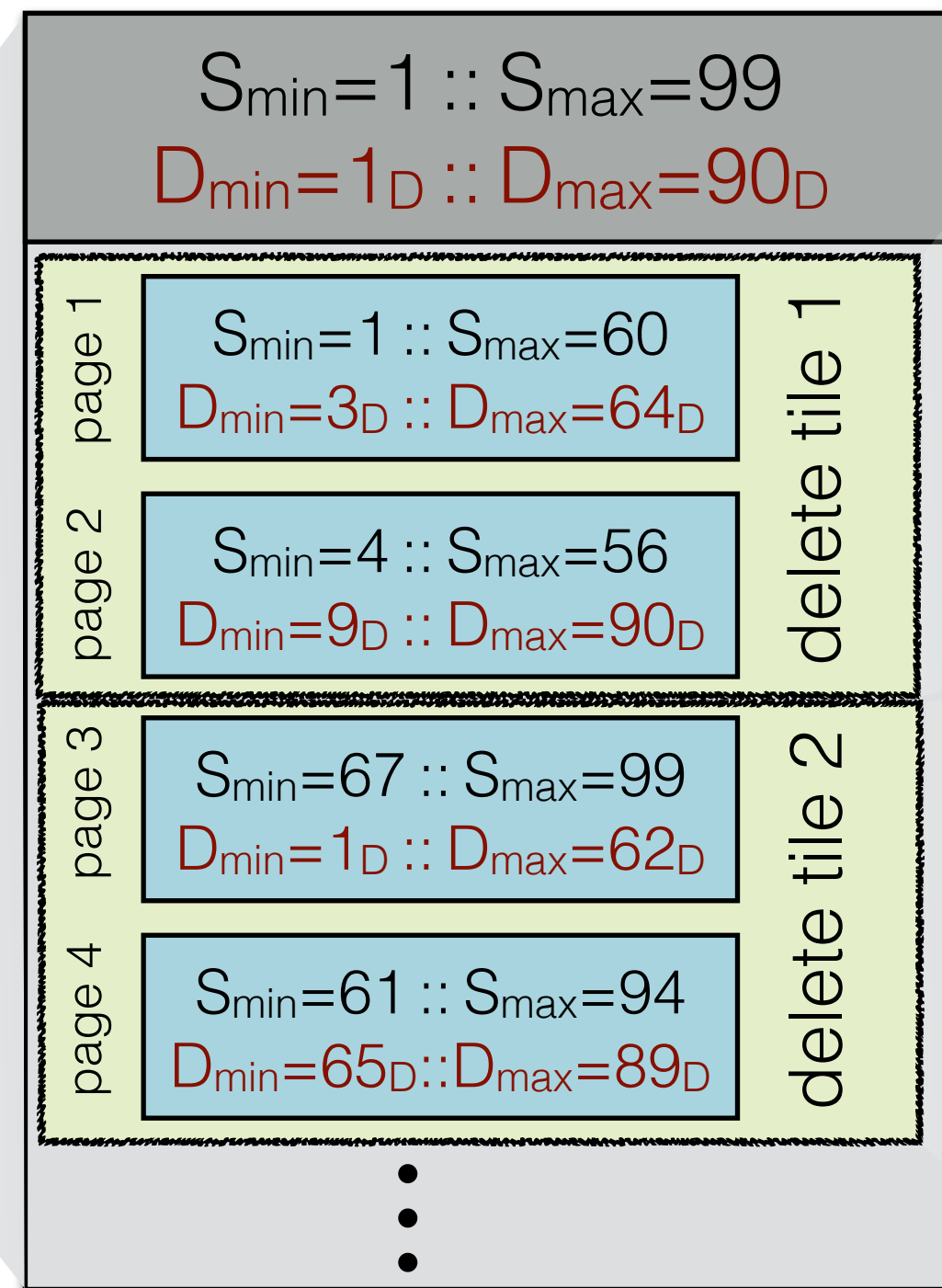


Key Weaving storage layout

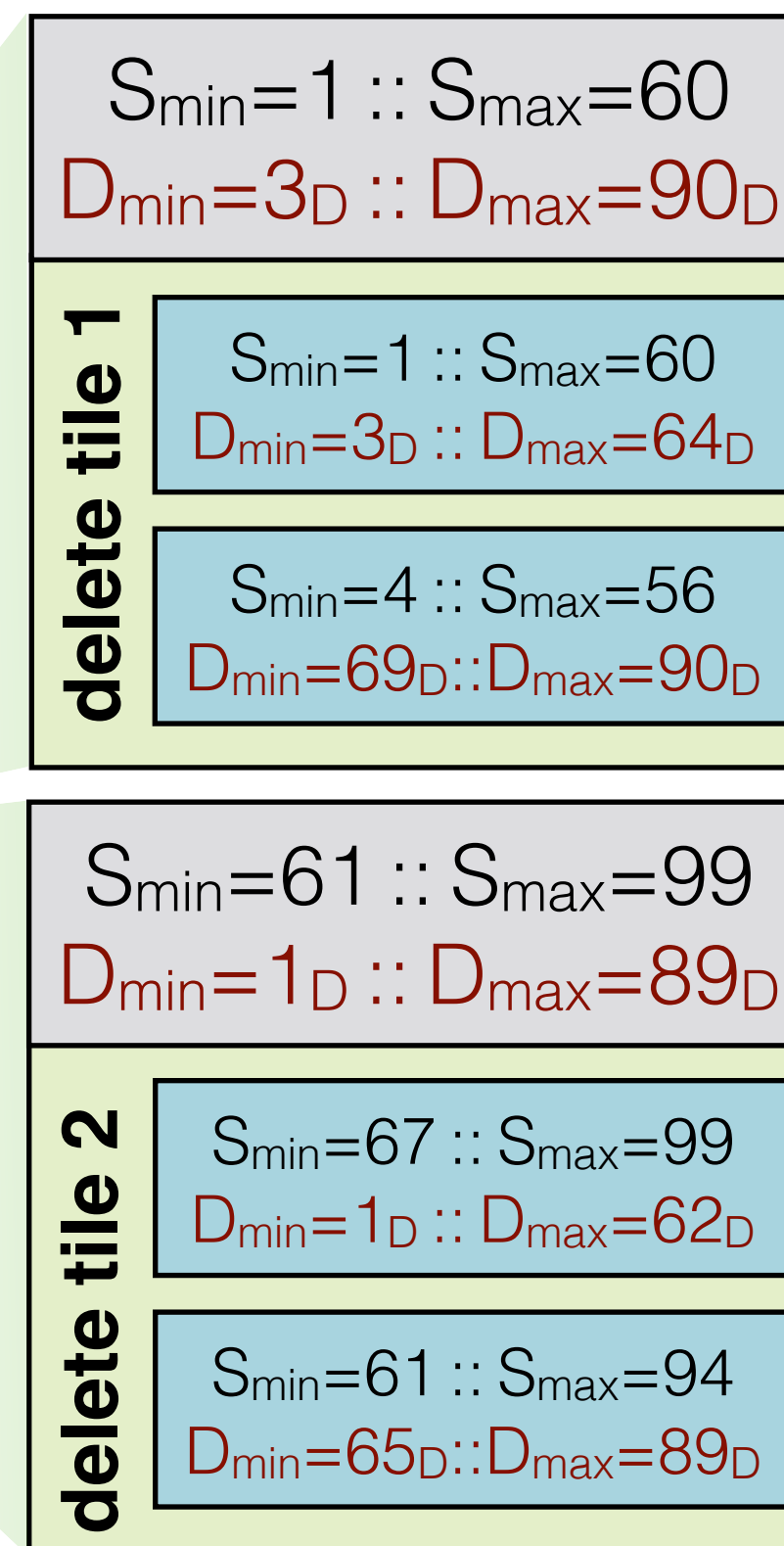
get(14)



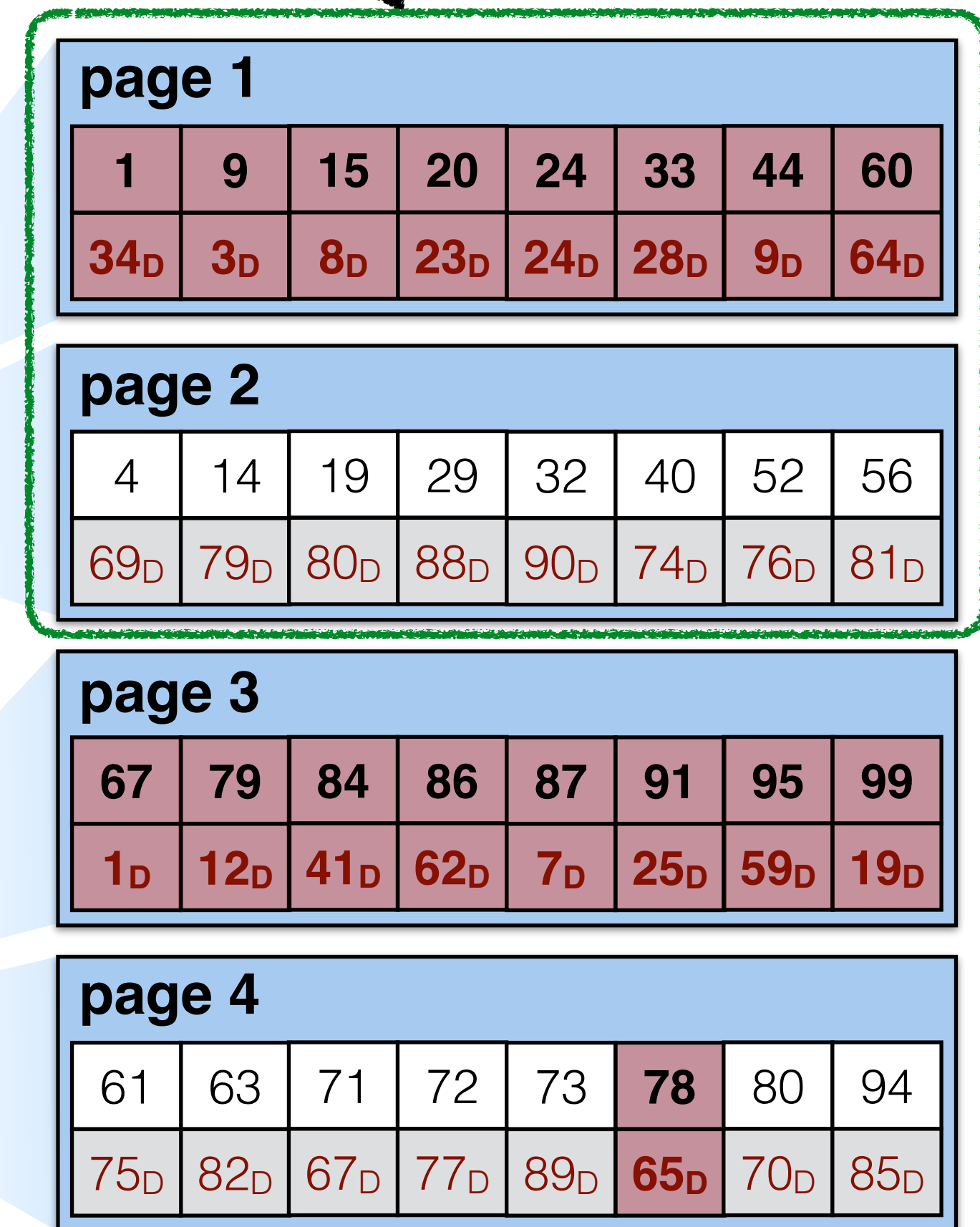
SST file



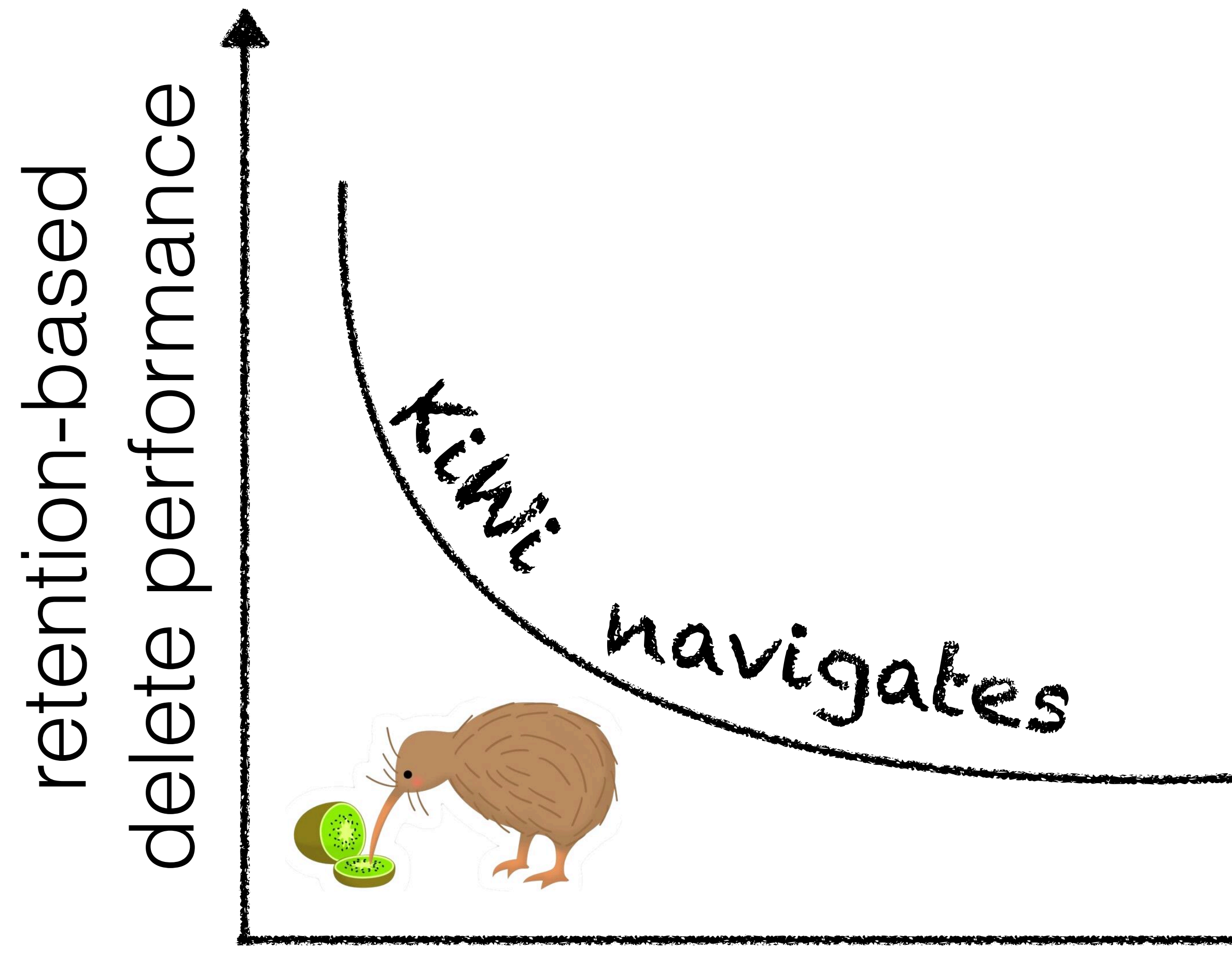
partitioned on S



partitioned on D



sorted on S



lookup
performance

retention-based
delete performance



kiwi
navigates

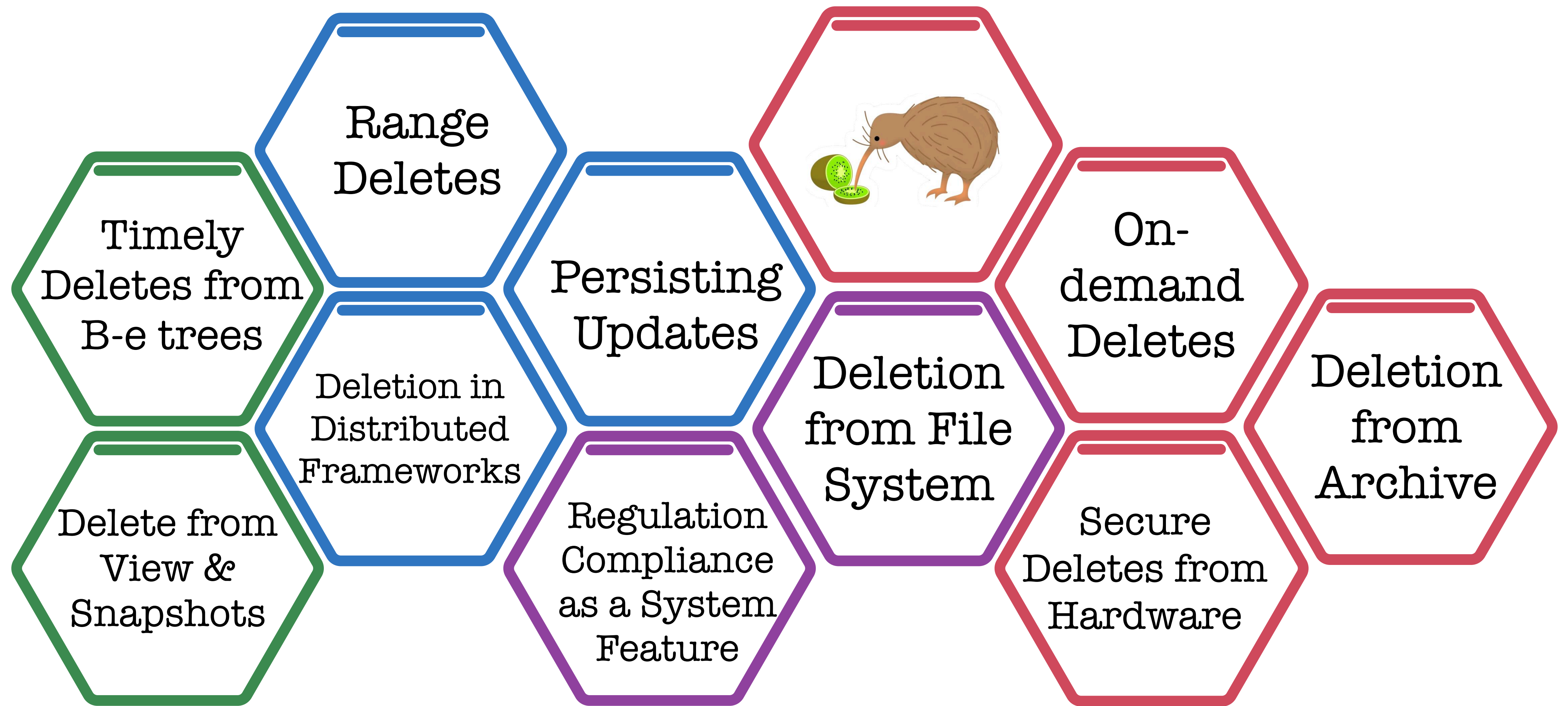
lookup
performance

workload

data
structure

h^*

$$h^* = \sqrt{\frac{f_{\text{SRD}} \cdot \frac{N}{B}}{\phi \cdot L \cdot (f_{\text{EPQ}} + f_{\text{PQ}}) + L \cdot f_{\text{SRQ}}}}$$



SIGMOD '20

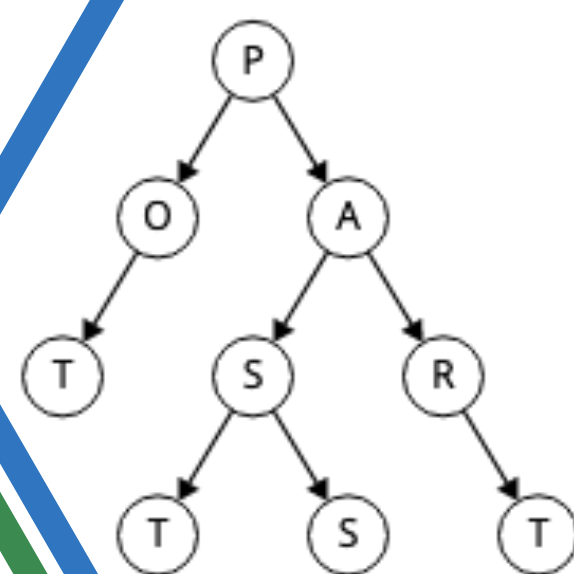
VLDB '21

EDBT '22

DEBull '22

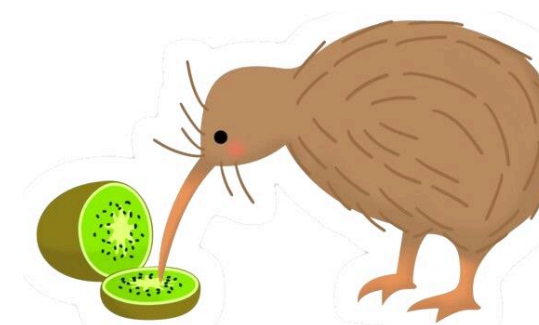
SIGMOD '22

Timely
Deletes from
B-e trees



Deletion in
Distributed
Frameworks

100110
010101
1010



Deletion
from File
System



Deletion
from
Archive

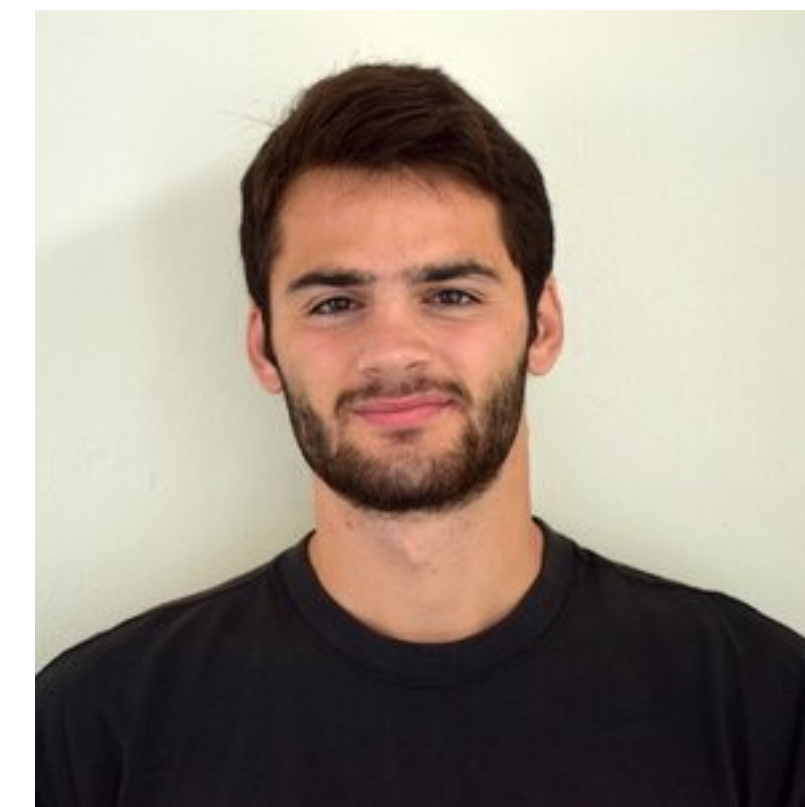
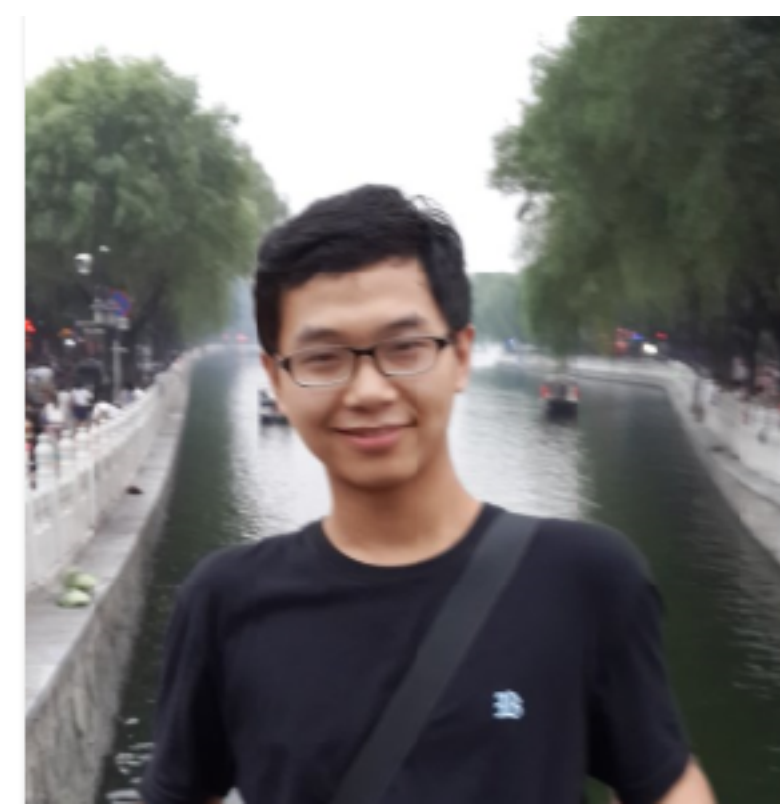
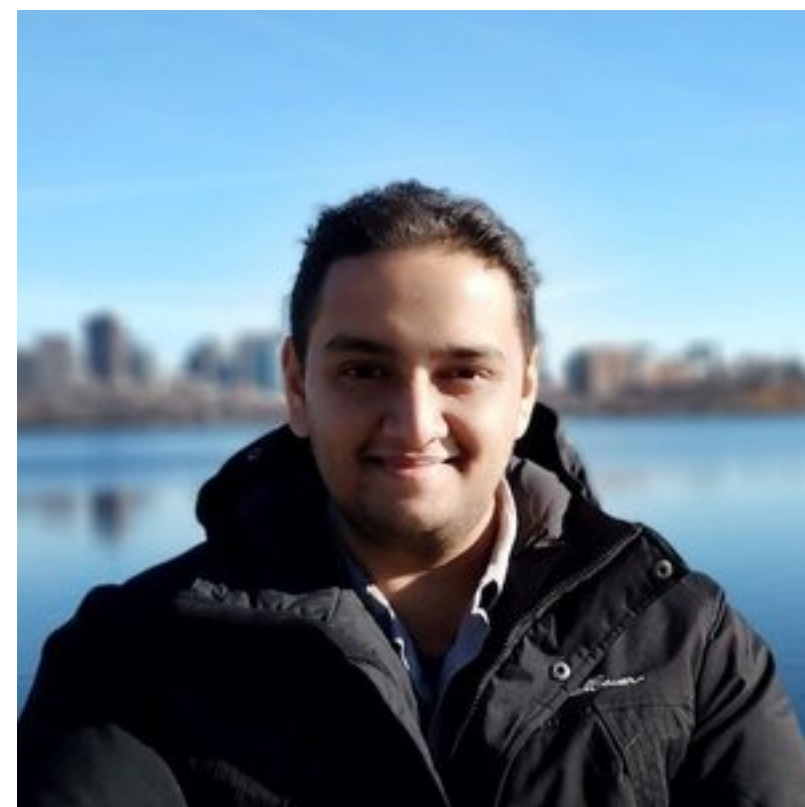
Delete from
View &
Snapshots

Regulation
Compliance
as a System
Feature

Secure
Deletes from
Hardware

<https://disc.bu.edu/persistent-deletes>

My **Amazing** Collaborators



Thank You!

<https://disc.bu.edu/persistent-deletes>

BOSTON
UNIVERSITY



facebook
research

