

# RocksDB Festival

Supported by IITP, StarLab.

July 26, 2021  
강정현, 김산, 허진

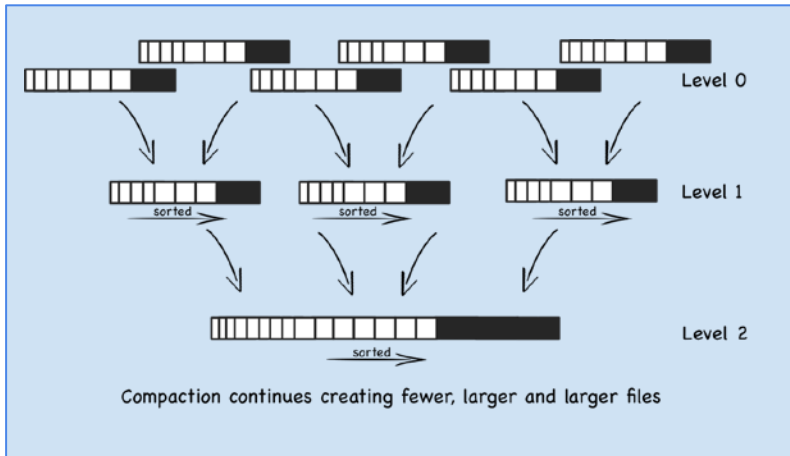
[rilac1@naver.com](mailto:rilac1@naver.com), [waterfog9580@gmail.com](mailto:waterfog9580@gmail.com), [jinh2352@gmail.com](mailto:jinh2352@gmail.com)

Team: JSJ

# RocksDB Festival

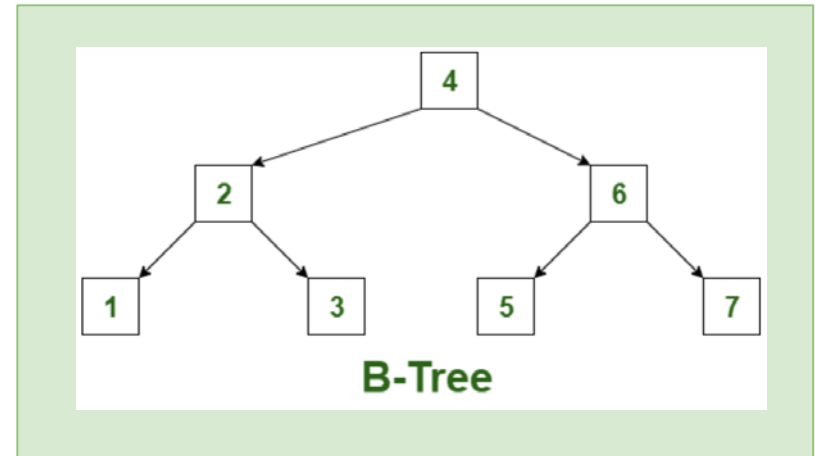
- RocksDB vs MongoDB
  - ✓ Data Structure

## Rocks DB



LSM-Tree  
(out-place update)  
-> Write Good!

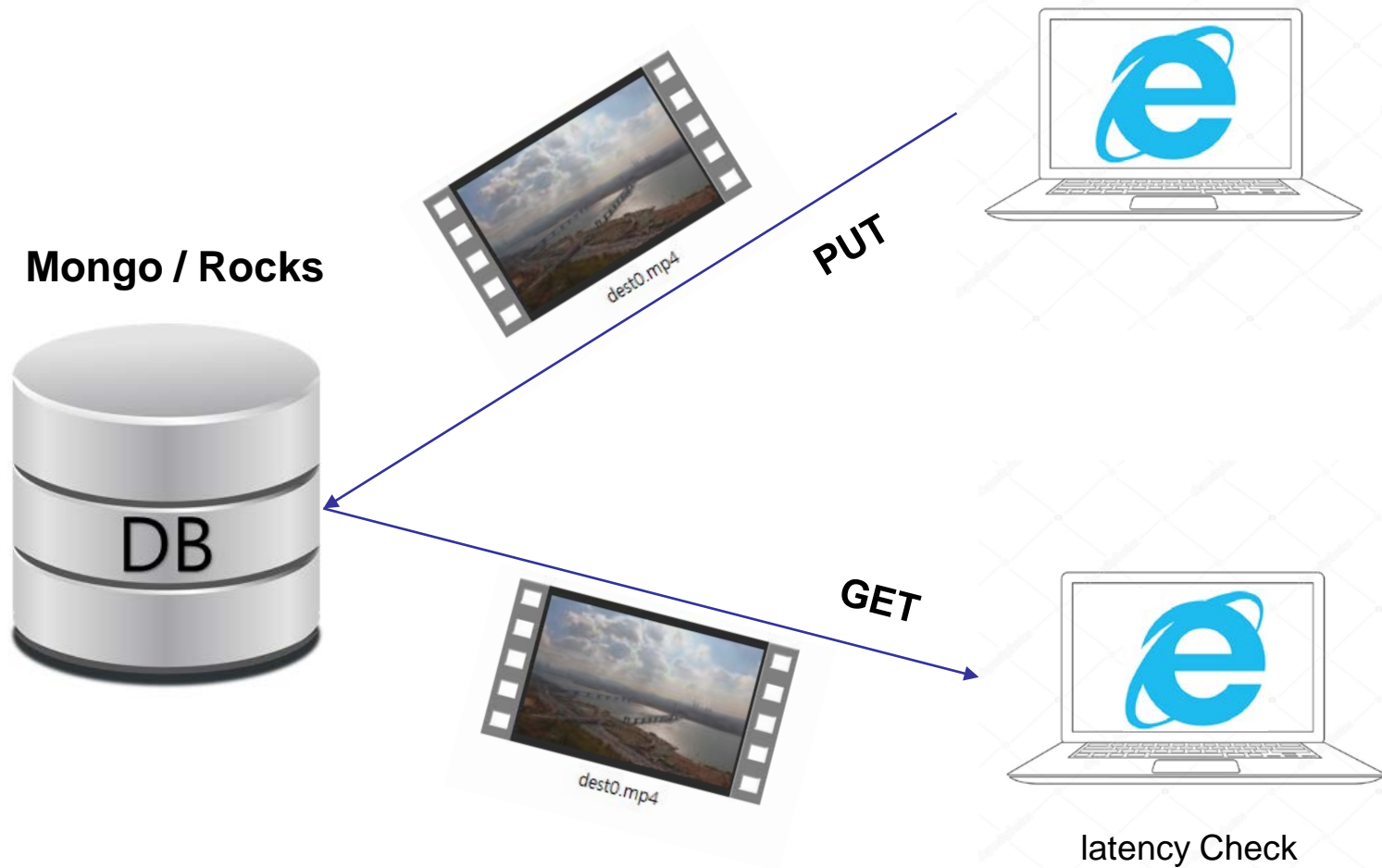
## Mongo DB



B-Tree  
(in-place update)  
-> Read Good!

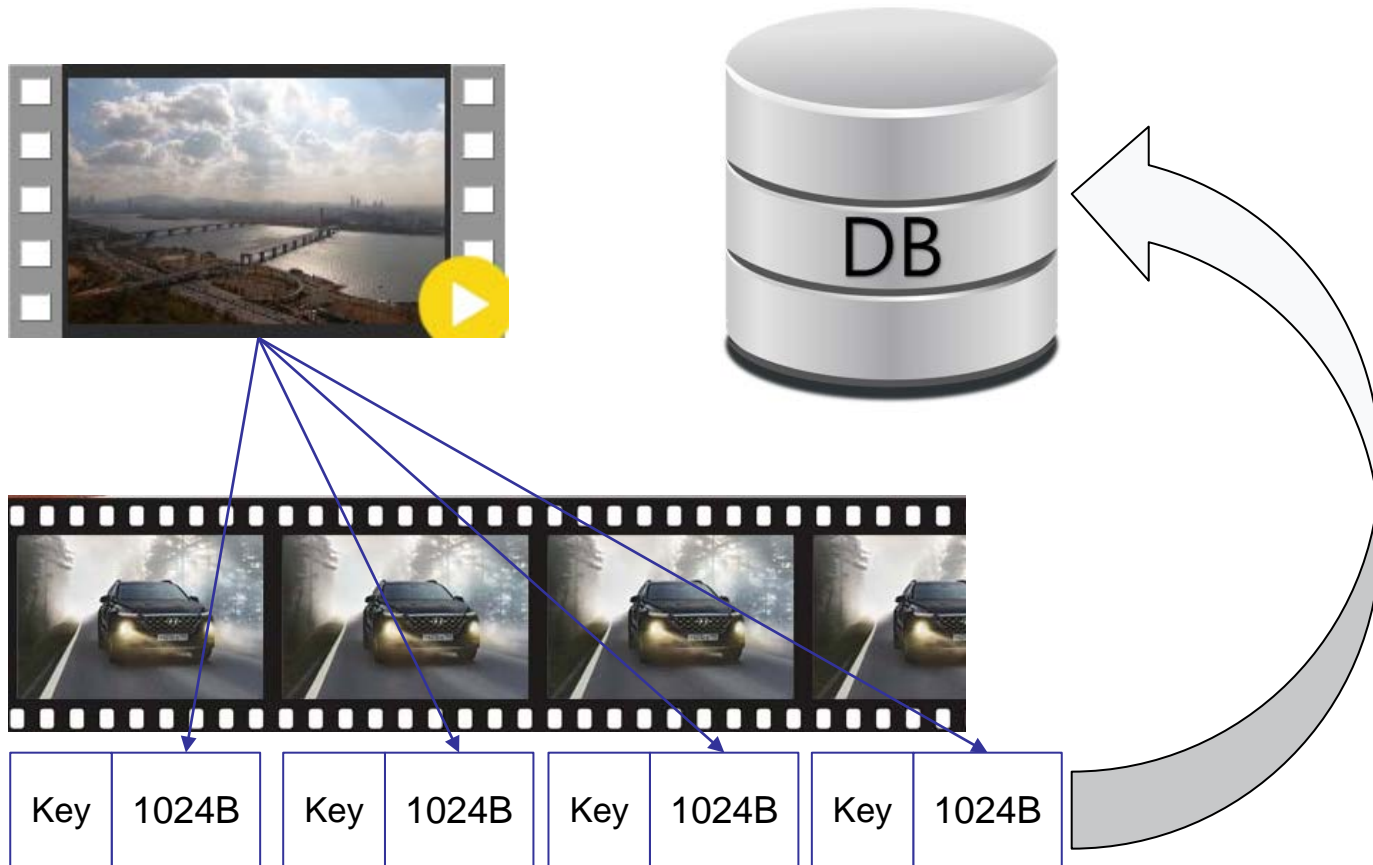
# RocksDB Festival

- Put Large Video Files



# RocksDB Festival

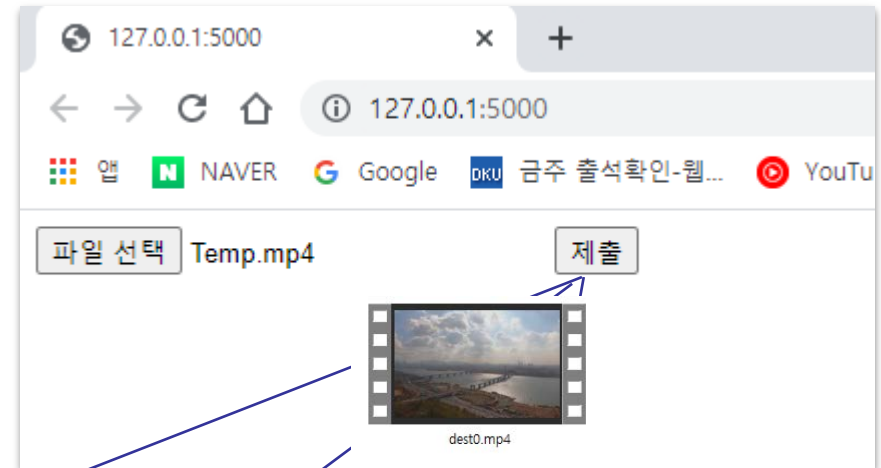
- Put Large Video Files



# RocksDB Festival

- MongoDB
  - ✓ GridFS

```
15 @app.route('/upload', methods=['POST'])
16 def upload():
17     file = request.files.get("file")
18     file_name = request.form.get("file_name")
19     data = file.read()
20     content_type = file.content_type
21     insertimg = gfs.put(data, content_type=content_type, filename=file_name)
22     return str(insertimg)
23
```



```
{
  "_id": {
    "$oid": "60f3fb9ab4bcbc98a6b8b00a"
  },
  "files_id": {
    "$oid": "60f3fb9ab4bcbc98a6b8b009"
  },
  "n": 0,
  "data": {
    "$binary": {
      "base64": "AAAAIGZ0eXBpc29tAAACAGlzb21pc28yYXZjMW1vAAAAADGAAA7pwAAQAAAAQAAAAAAAAAAAAAAAAAAAAAAwA/BAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA"
    }
  }
}
```

```
{
  "_id": {
    "$oid": "60f3fb9ab4bcbc98a6b8b00b"
  },
  "files_id": {
    "$oid": "60f3fb9ab4bcbc98a6b8b009"
  },
  "n": 1,
  "data": {
    "$binary": {
      "base64": "4QtkNZbKaf3deWle+zXvrtld1+EukWF79vngK6yy1vNL8vSyd1y31W33VgnJv+rX2bCTmnmz0G11duLrmpBZv0usycFPqu+EDe+zKgT4n5jd0Rr94pCyICYZcnu+KxBUd+aex1rDQkIGVBev+vItTpRDP0vJWSeTh0/xbMEbcS"
    }
  }
}
```

```
{
  "_id": {
    "$oid": "60f3fb9ab4bcbc98a6b8b00c"
  },
  "files_id": {
    "$oid": "60f3fb9ab4bcbc98a6b8b009"
  },
  "n": 2,
  "data": {
    "$binary": {
      "base64": "eKF64960QzjCVgEy1CDEJHXWmGS51AKA+TvX5tcveQ0Y5OUfkSt1BYCx2BIGVAAuFS4sAMkAAIcgXE6nPnflvn418gADyJXnQJ54GhN6xD2/RX1Huh0Q4Kpm+L5JVCKbkvIrMaeSevIQcdhBrPZ8UelqnVHMUASJ0QwARAEB8AIiIVSmFbzKX6TAAARvE3iwBv70IRqPU7/6f"
    }
  }
}
```

# RocksDB Festival

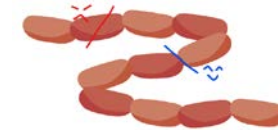
- RocksDB for Video Files Store

## Issue

✓ RocksDB API parameter issue, only supports std::string types.

-> Simply parsing by interval causes problems with metadata access

✓ How to cut it in frames? (O) (X)



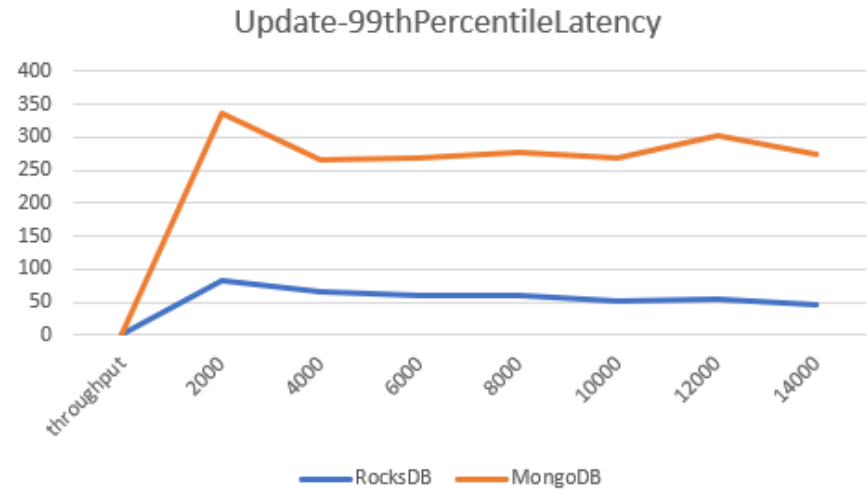
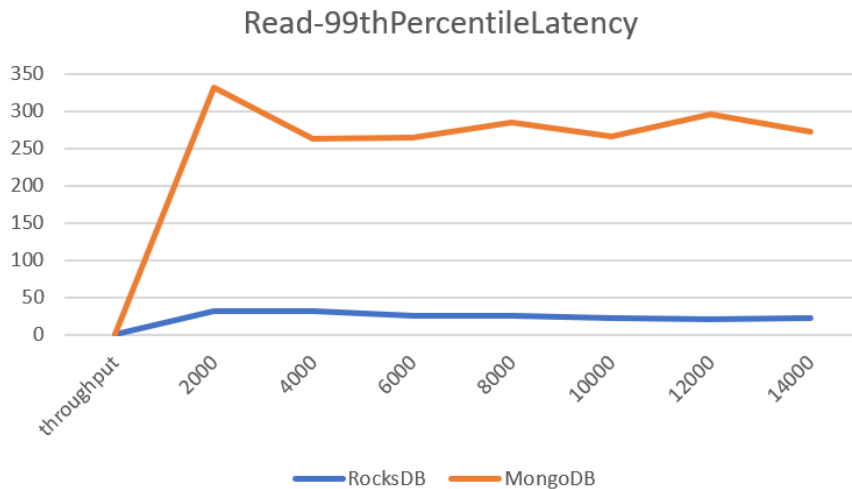
✓ What about metadata access?

-> Simply parsing by interval causes problems with metadata access

Offset	00	01	02	03	04	05	06	07	08	09	10	11	12	13	14	15	ASCII
00000000	80	00	00	18	66	74	79	70	6D	70	34	32	00	00	00	00	...ftypmp42...
00000016	69	71	6F	6D	6D	70	34	32	00	00	24	CD	6D	6F	6F	76	isommp42..\$imov
00000032	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	...lmvhd...f.q.
00000048	CD	9C	71	91	00	00	02	58	00	00	36	00	00	01	00	00	f.q....X.60...
00000064	01	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000080	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....@...
00000096	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000112	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000128	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	.....
00000144	69	6F	64	73	00	00	00	00	00	00	00	00	00	00	00	00	iods.....Ogg).
00000160	FF	00	00	0E	60	74	72	00	00	00	00	00	00	00	00	00	y...`trak...tkh

# RocksDB Festival

- RocksDB vs MongoDB
  - ✓ YCSB Bench

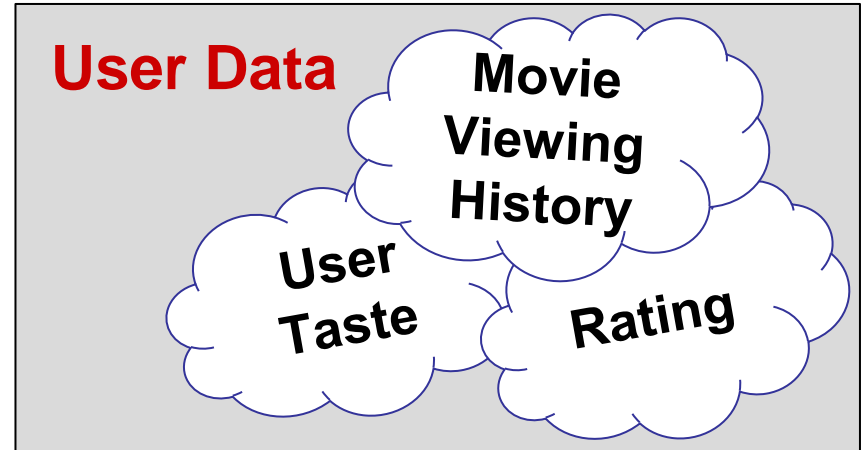


Latency : 99thPercentileLatency(us)  
Throughput : Operation/sec (YCSB의 target 옵션을 통해 제한하면서 실험 진행)  
Record : 100,000  
Workload : Read(50%), Write(50%)

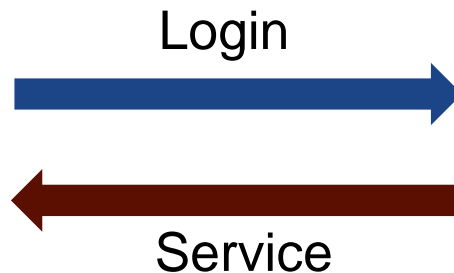
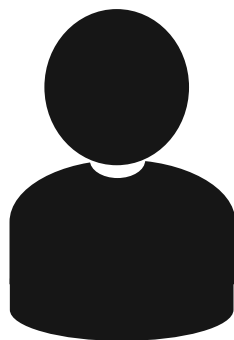
- Who Uses RocksDB
  - Distributed Database
    - Embed RocksDB as Storage Engine
      - Apache Cassandra
      - CockroachDB
      - MySQL (MyRocks)
      - Rockset
  - Outside of Distributed Database
    - **Kafka Streams**
    - Apache Samza
    - **Netflix**
    - Stander UK
    - **Uber**



# RocksDB Festival



- Provide **personalized recommendations** services



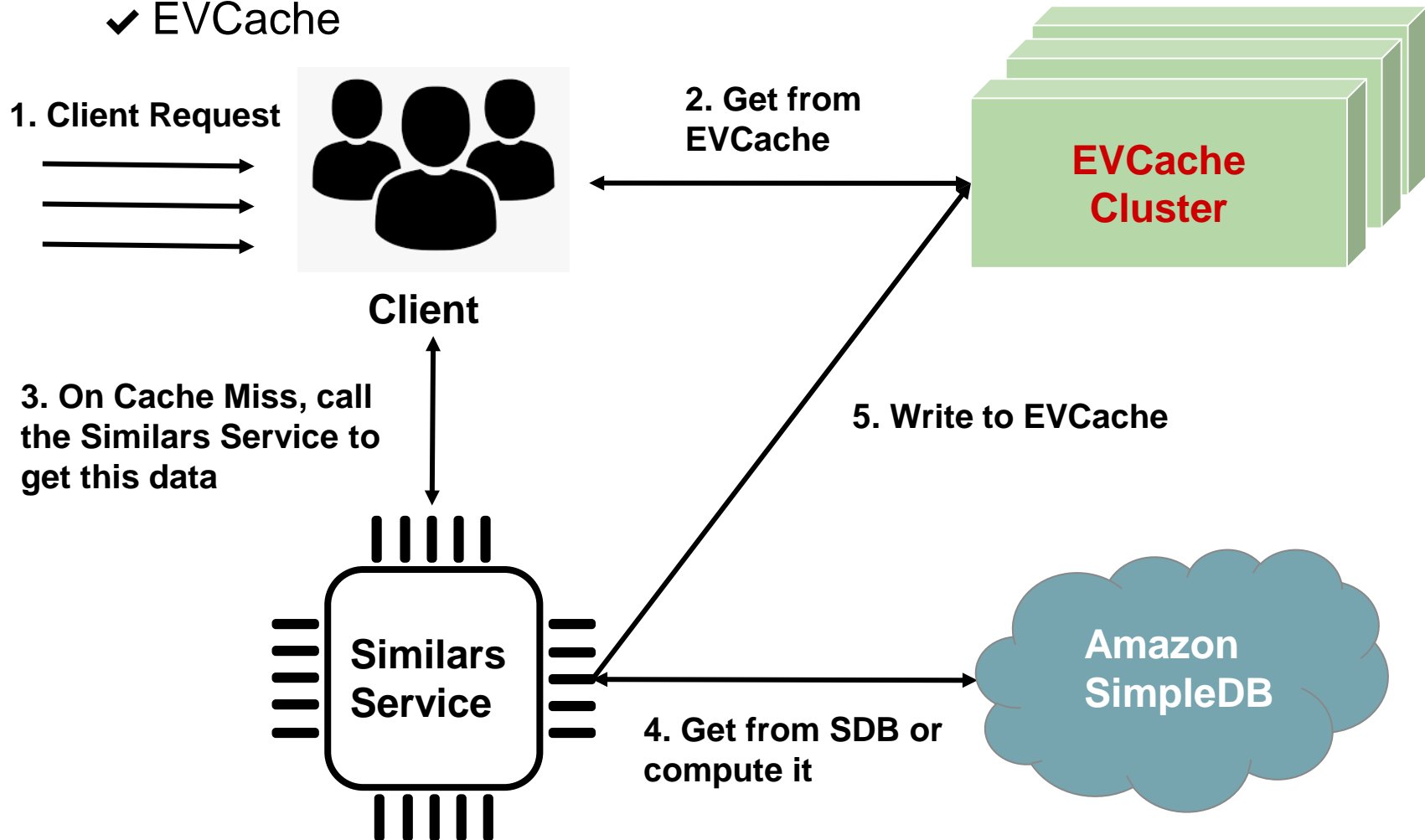
www.netflix.com

Get  
User Data  
from DB



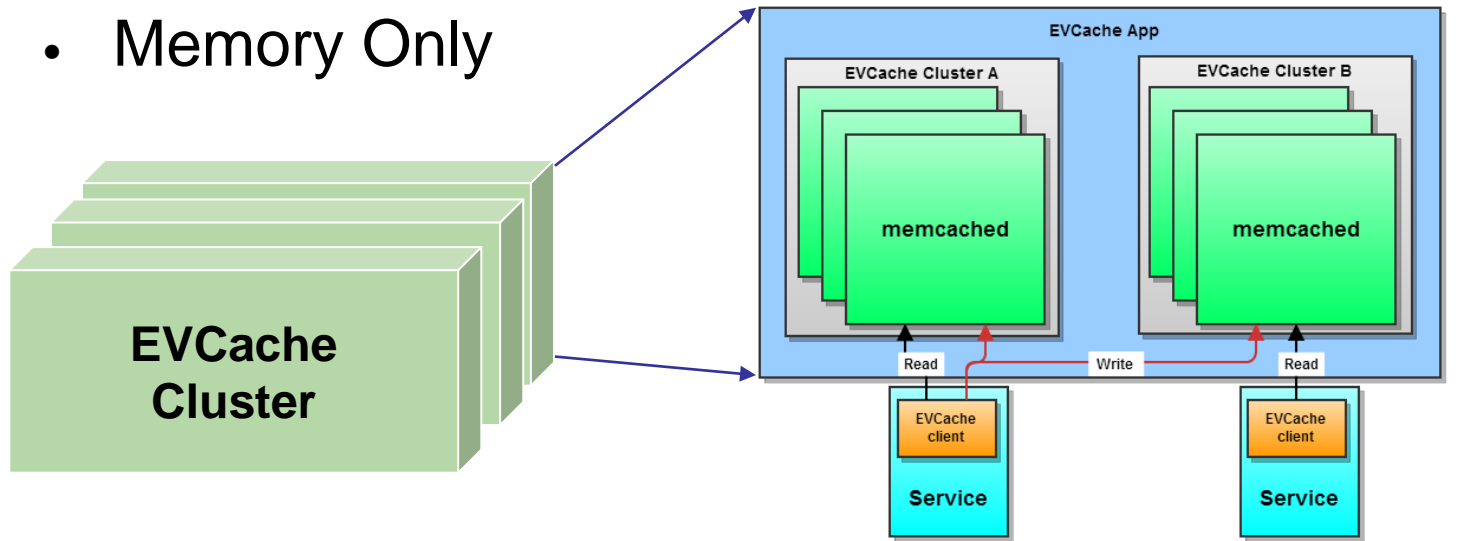
# RocksDB Festival

- Netflix
  - ✓ EVCache

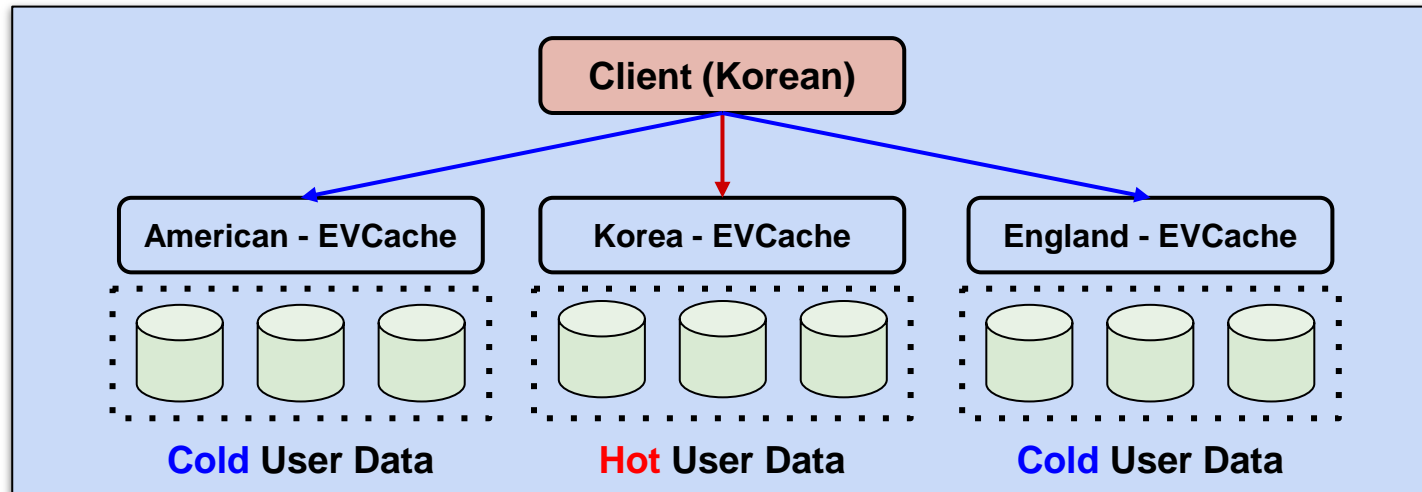


# RocksDB Festival

- Memory Only

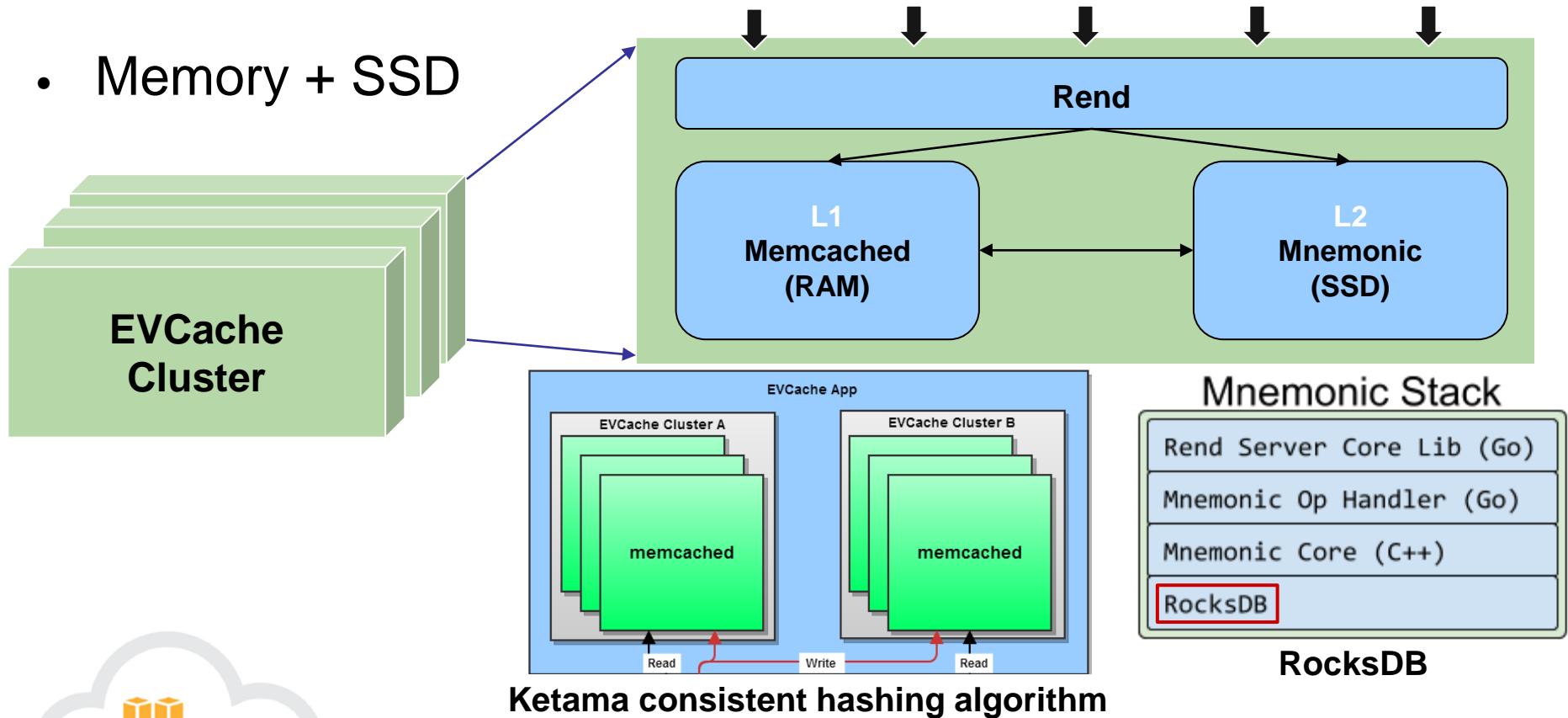


Ketama consistent hashing algorithm



# RocksDB Festival

- Memory + SSD



\* Linux 기준

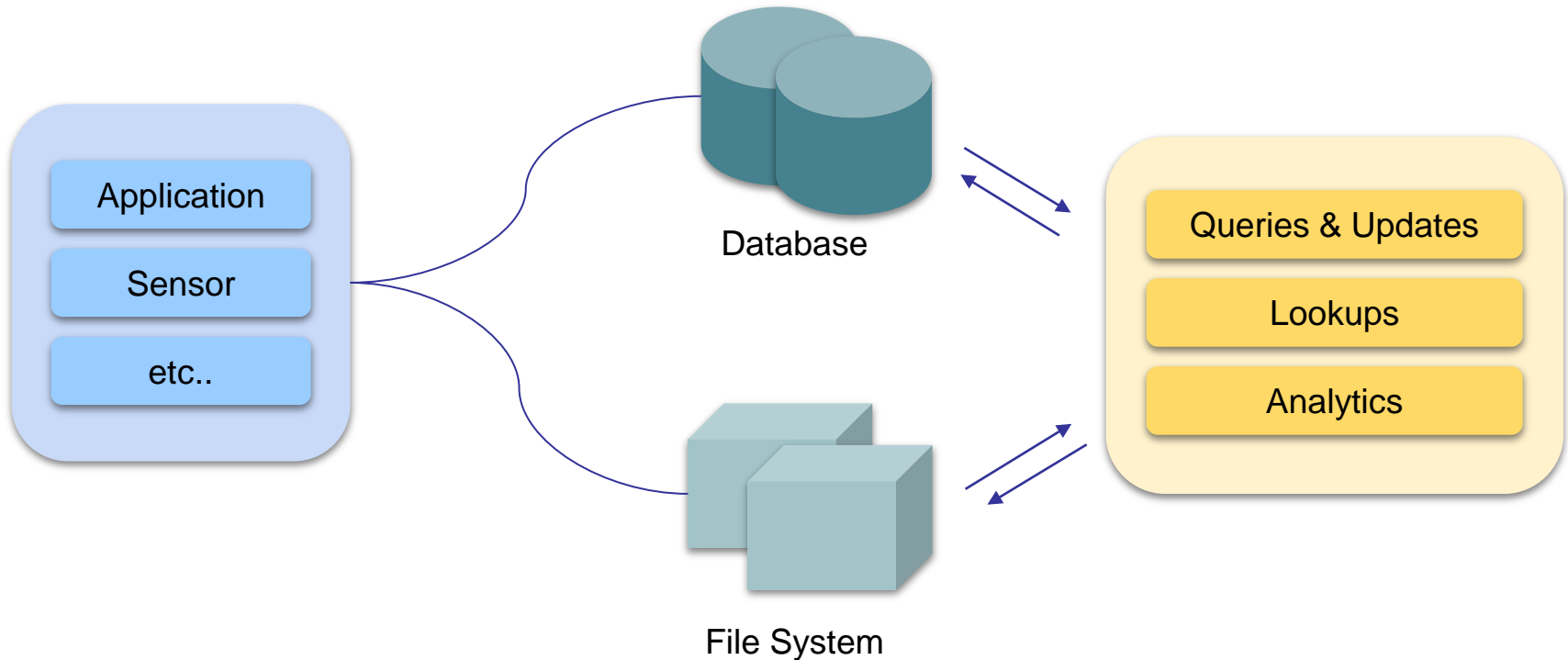
\* 1 GiB = 1.07 GB

인스턴스 패밀리	인스턴스 유형	vCPU	ECU	메모리	인스턴스 스토리지 (SSD)	시간 당 가격
메모리 최적화	<b>r3.xlarge</b>	4	13	30.5 GiB	80 GB	0.332 USD
스토리지 최적화	<b>i2.xlarge</b>	4	14	30.5 GiB	800 GB x 100	0.853 USD

x 2.5

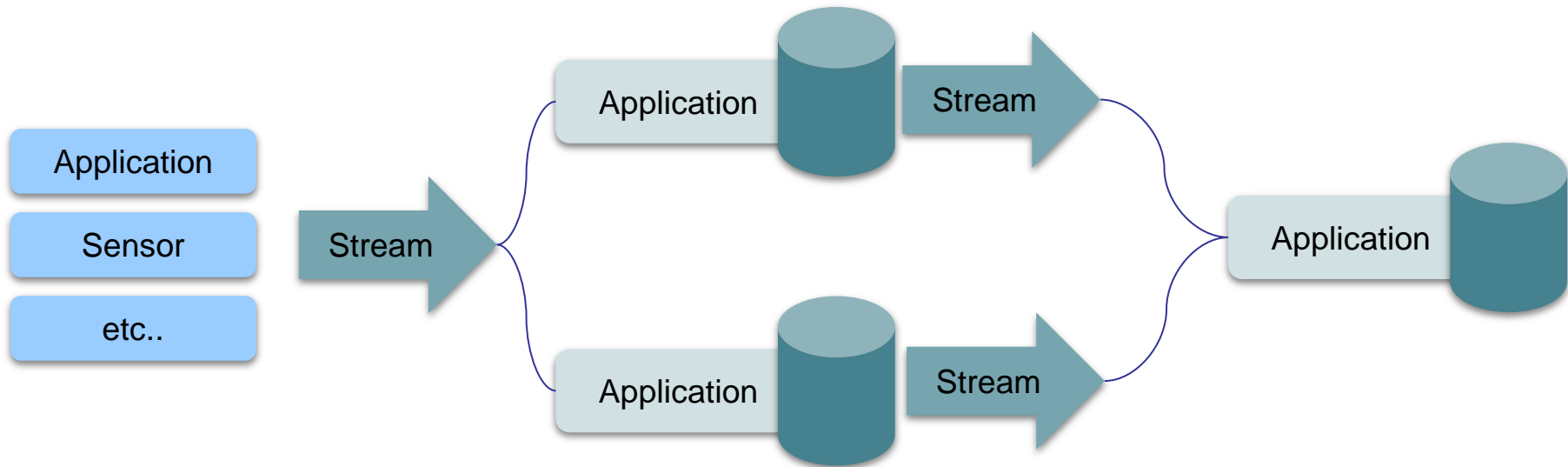
- Kafka Streams

- Before Stream processing



- Kafka Streams

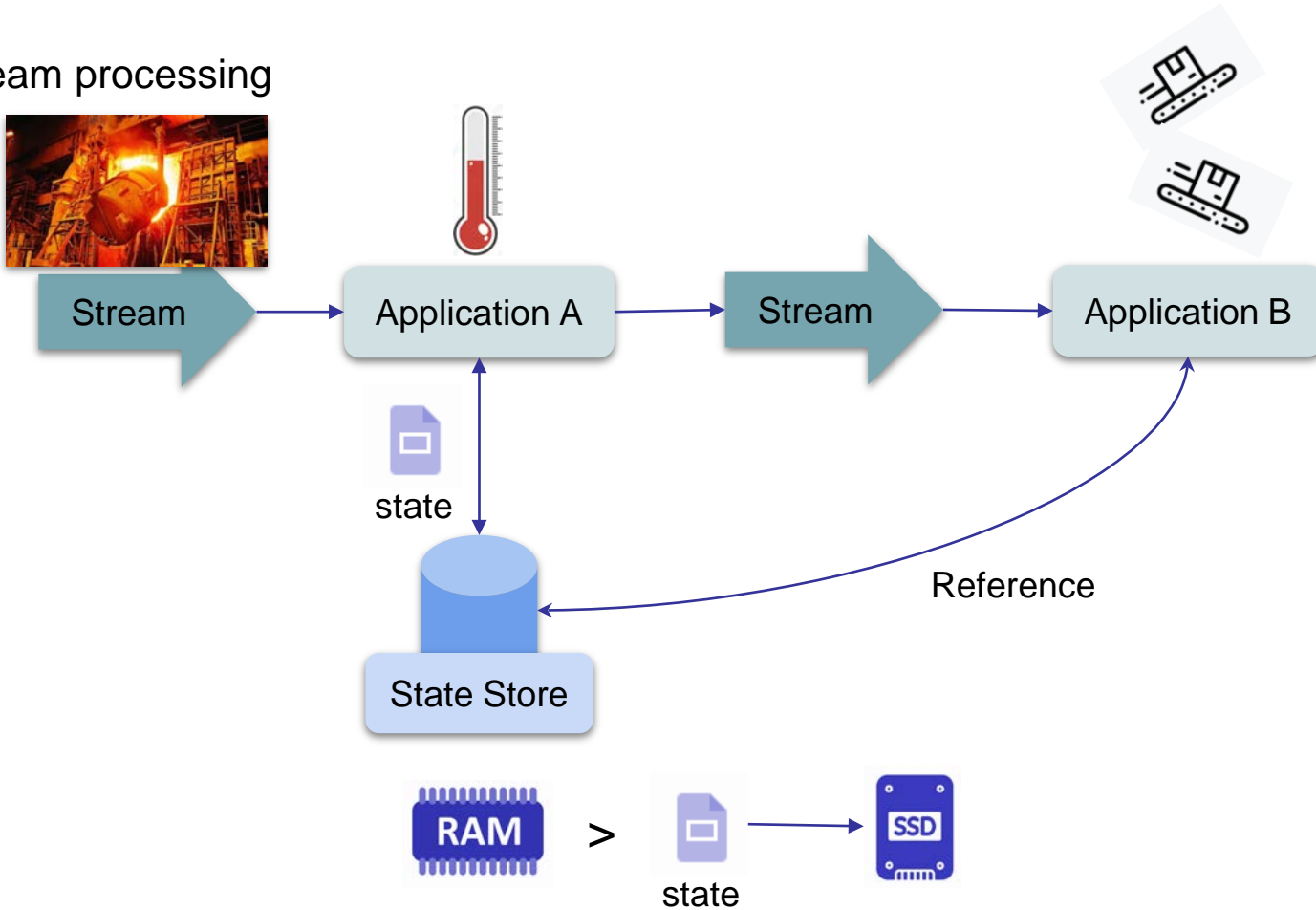
- Stream processing



# RocksDB Festival

- Kafka Streams

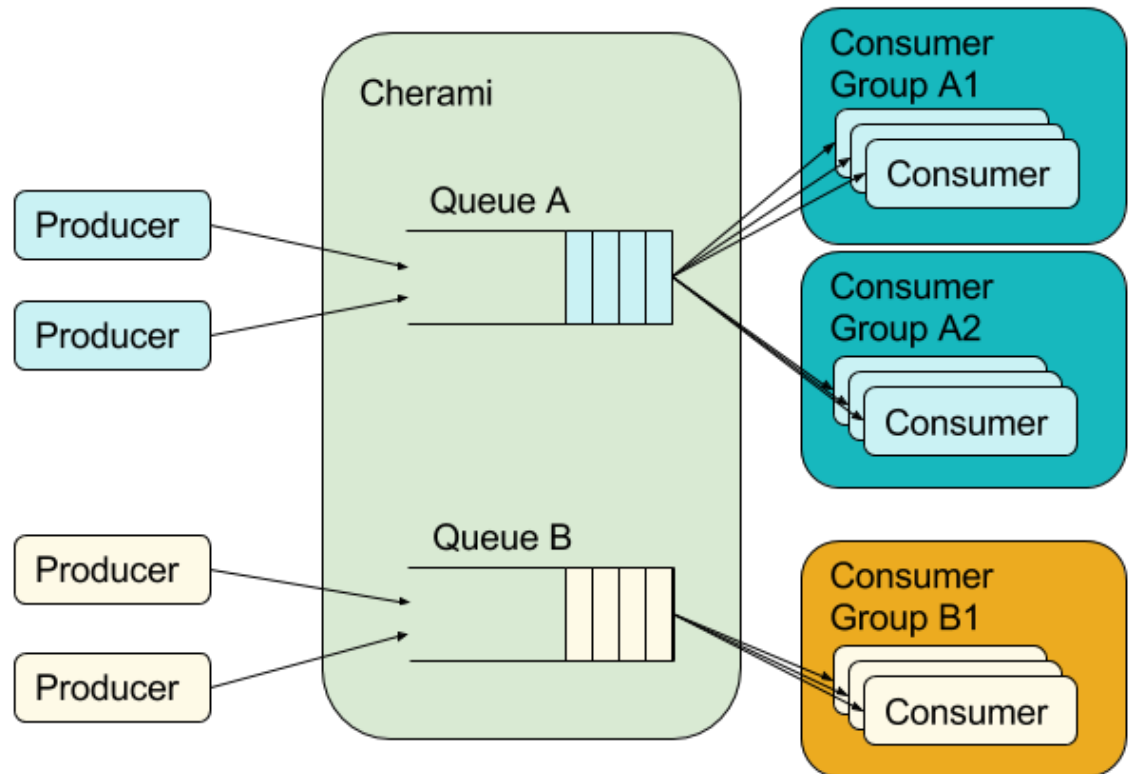
- Stream processing



state larger than available main memory can be supported

# RocksDB Festival

- Uber : Cherami (Message Queue System)
  - Message Queue System

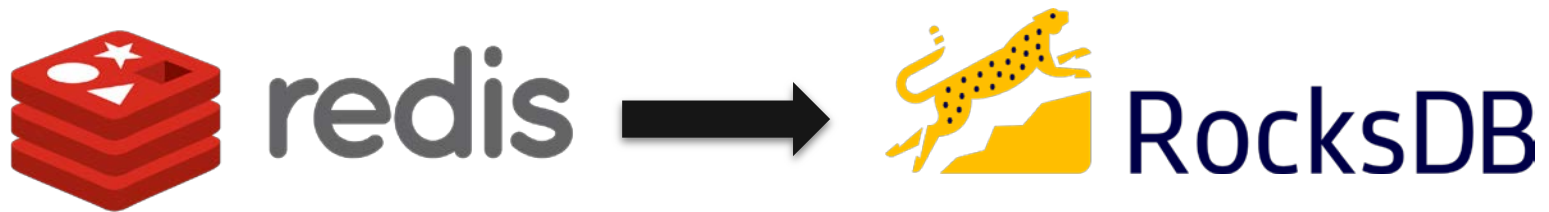




# RocksDB Festival

---

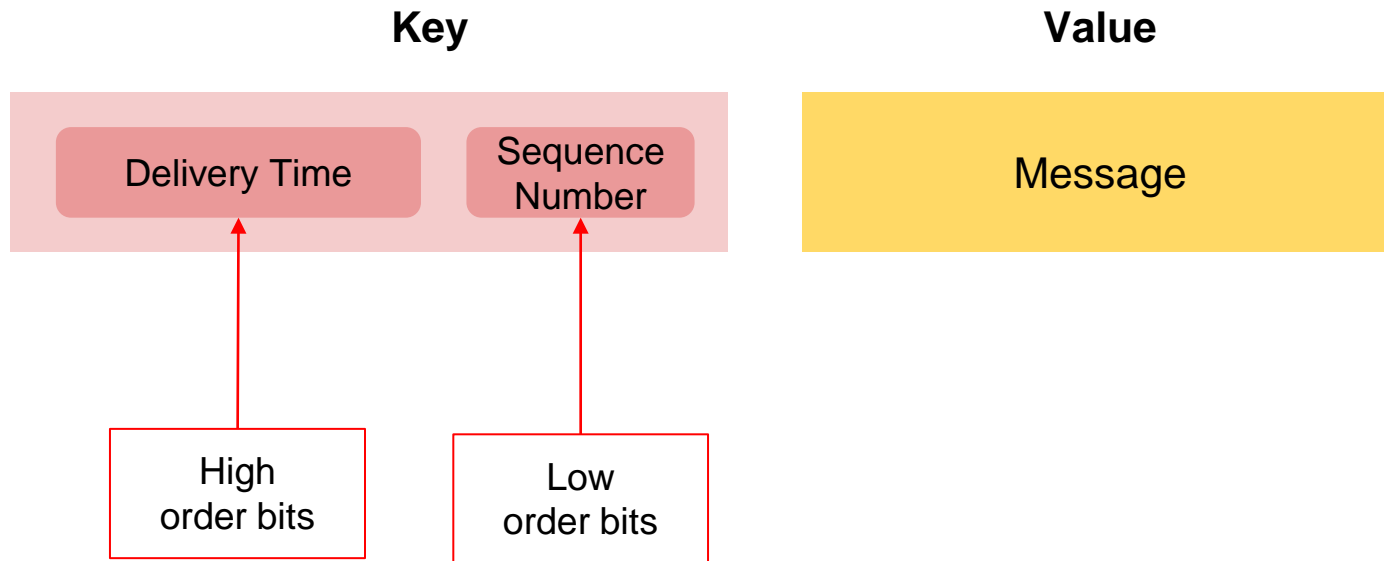
- Uber : Cherami (Message Queue System)
  - Redis to RocksDB



**In-memory** Key-Value Store  
Isn't as **durable** or **scalable** as Uber needed.

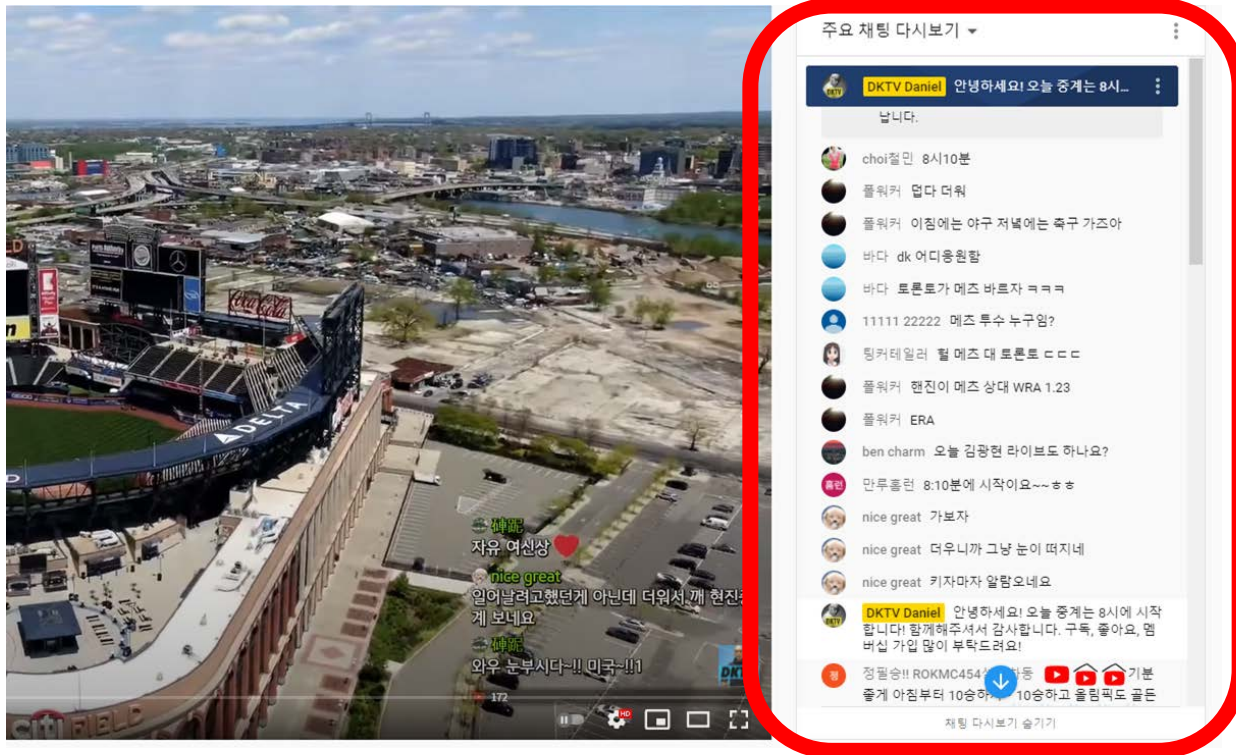
# RocksDB Festival

- Uber : Cherami (Message Queue System)
  - Implement Message Queue by RocksDB



# RocksDB Festival

- Real workload(live chatting data) - interface implement



# Discussion

---

