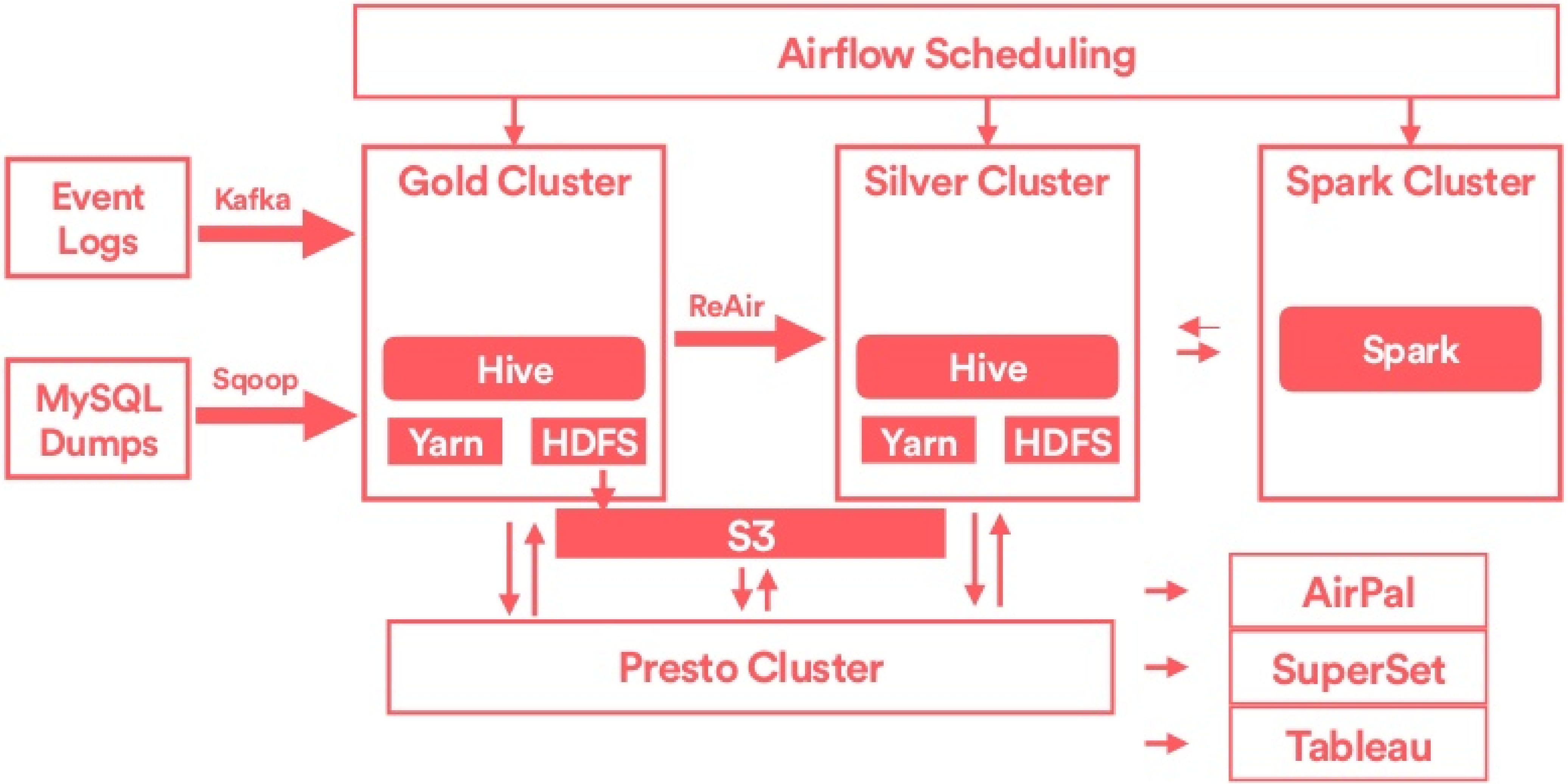
LIYIN TANG & JINGWEI LU

Building Data Products on Spark at Airbnb

airbnb

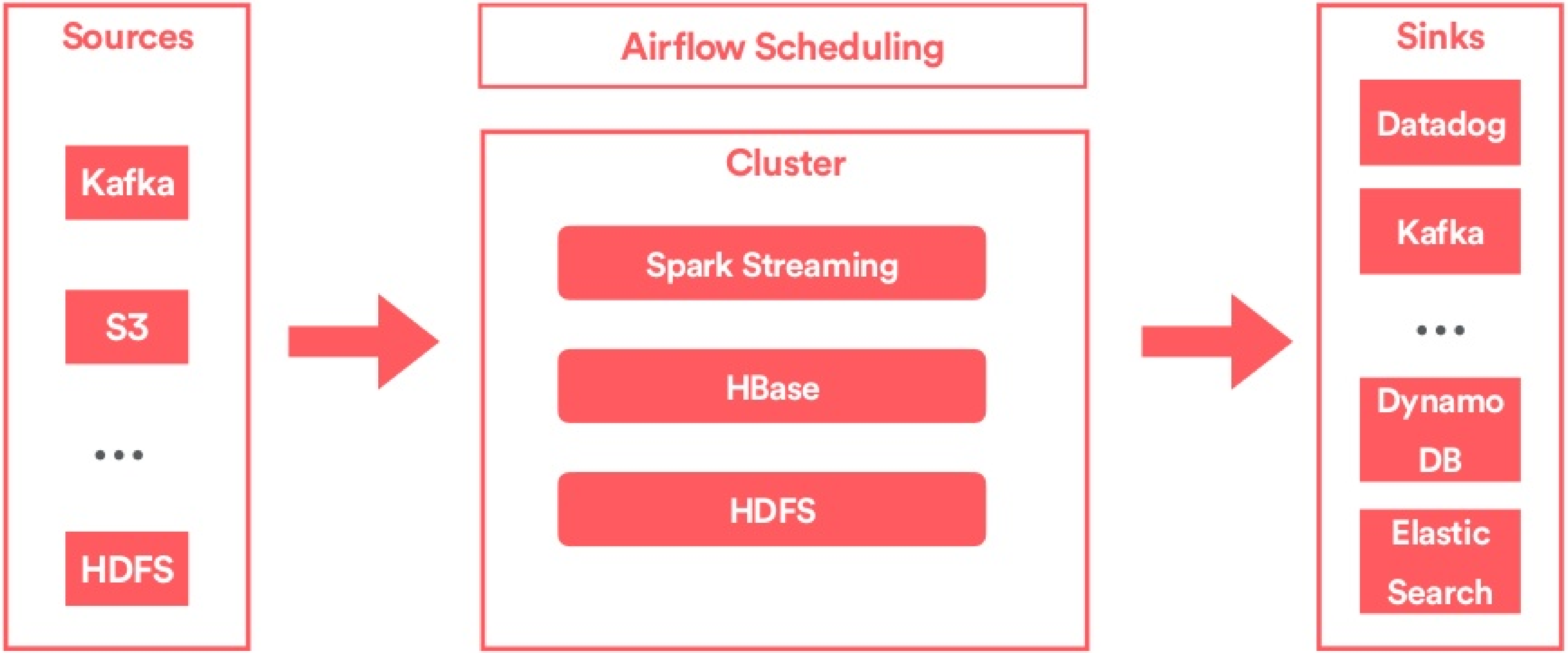
Data Infrastructure at Airbnb

Batch Infrastructure



Liyin Tang and Jingwei Lu

Streaming at Airbnb



Sources

Kafka

S3

• • •

HDFS

Airflow Scheduling

Cluster

Spark Streaming

HBase

HDFS

Sinks

Datadog

Kafka

• • •

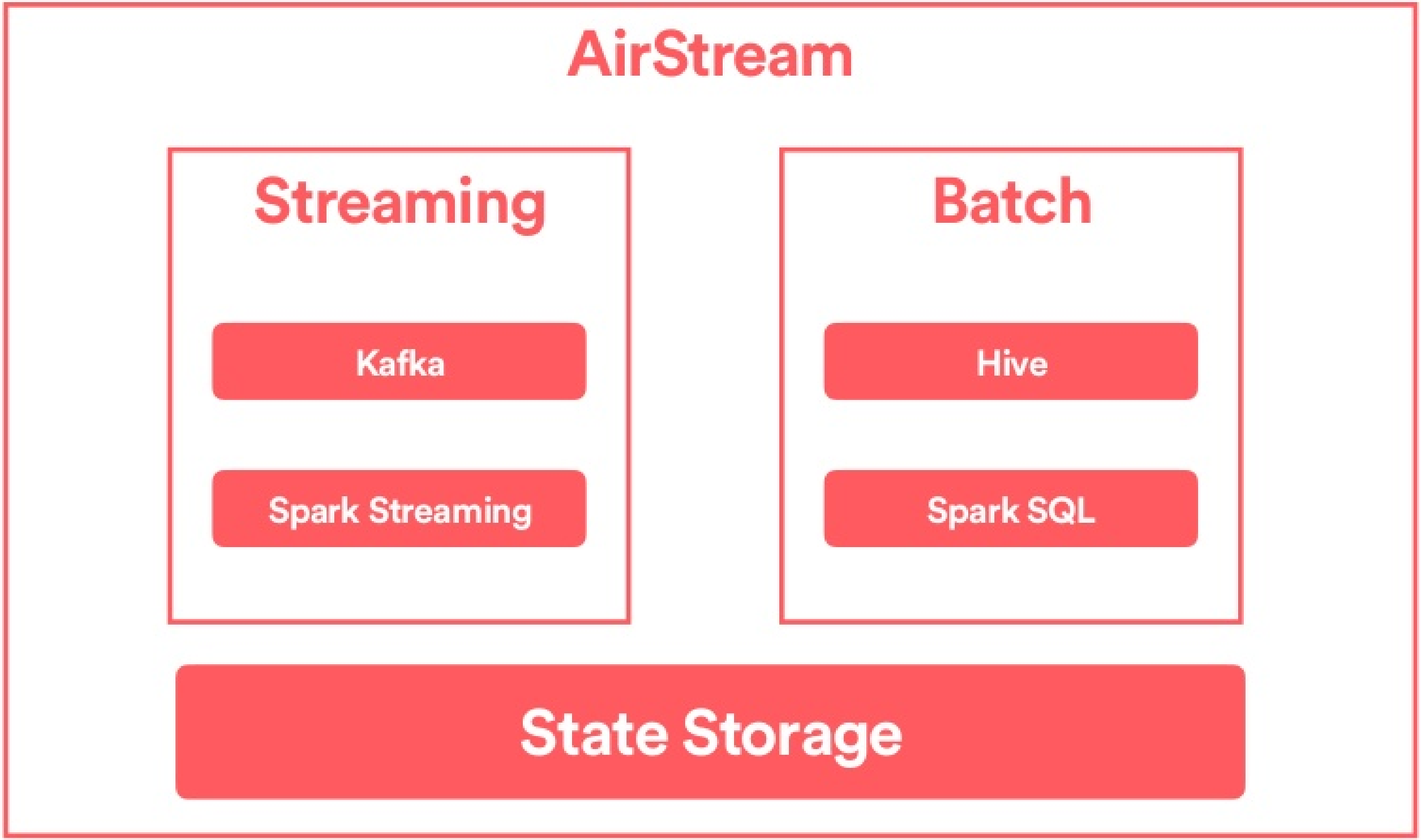
Dynamo DB

Elastic Search

Liyin Tang and Jingwei Lu

Lambda Architecture

Lambda Architecture



AirStream

Streaming

Kafka

Spark Streaming

Batch

Hive

Spark SQL

State Storage

Combine Streaming and Batch Processing

Sources

Streaming

```
source: [
 {
   name: source_example,
   type: kafka,
   config: {
     topic: "example_topic",
   }
 }
]
```

Batch

```
source: [
 {
   name: source_example,
   type: hive,
   sql: {
    select * from db.table where
ds='2017-06-05';
  }
 }
]
```

Computation

Streaming/Batch

```
process: [{
  name = process_example,
  type = sql,
  sql = """
   SELECT  listing_id, checkin_date, context.source as source
   FROM  source_example
   WHERE  user_id IS NOT NULL   """
 }]
```

Sinks

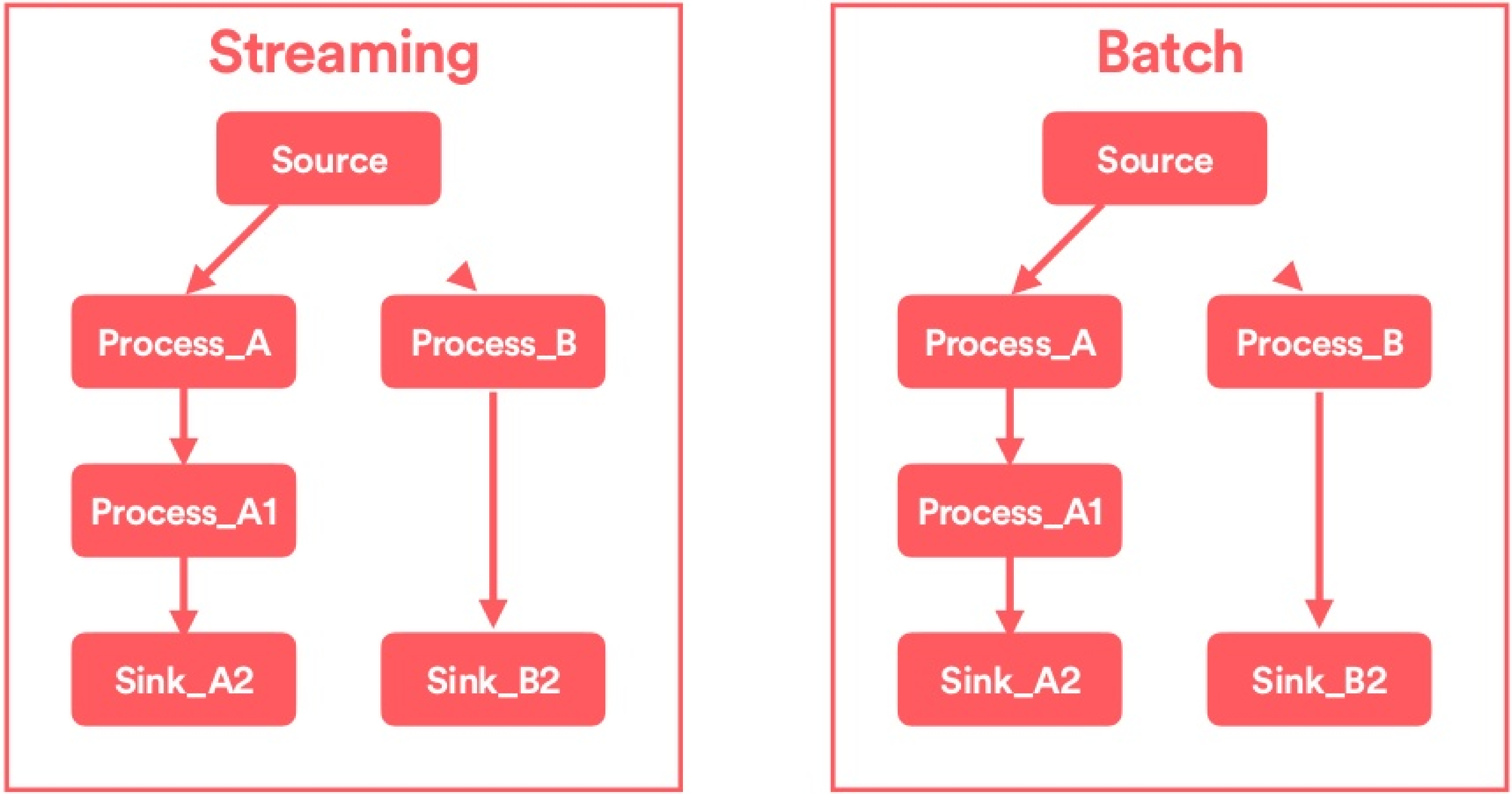
Streaming

```
sink: [
 {
  name = sink_example
  input = process_example
  type = hbase_update
  hbase_table_name = test_table
  bulk_upload = false
 }
]
```

Batch

```
sink: [
 {
  name = sink_example
  input = process_example
  type = hbase_update
  hbase_table_name = test_table
  bulk_upload = true
 }
]
```

Computation Flow



Streaming

Source

Process_A

Process_B

Process_A1

Sink_A2

Sink_B2

Batch

Source

Process_A

Process_B

Process_A1

Sink_A2

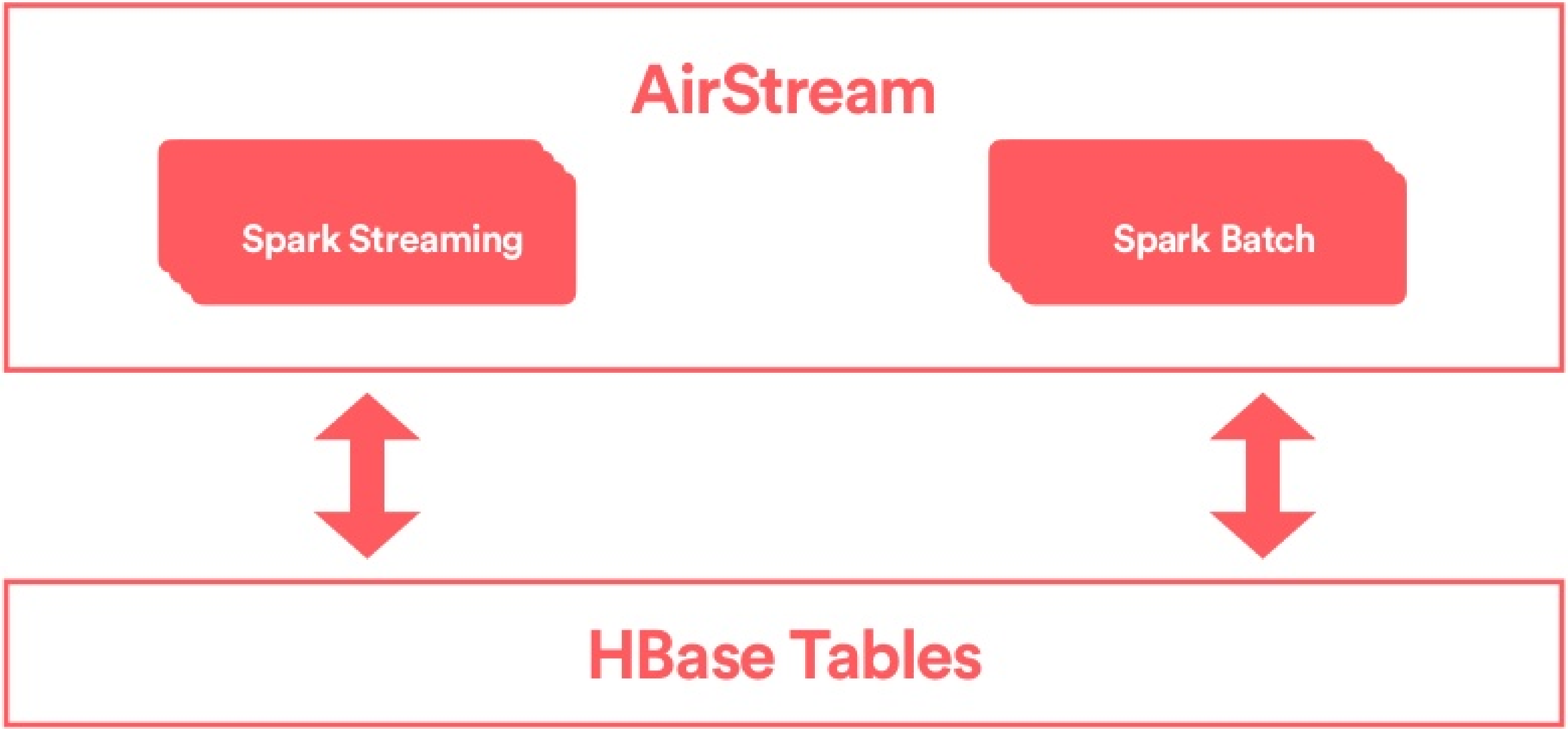
Sink_B2

Unified API through AirStream

- Declarative job configuration
- Streaming source vs static source
- Computation operator or sink can be shared by streaming and batch job.
- Computation flow is shared by streaming and batch
- Single driver executes in both streaming and batch mode job

Shared State Storage

Shared Global State Store



Why HBase

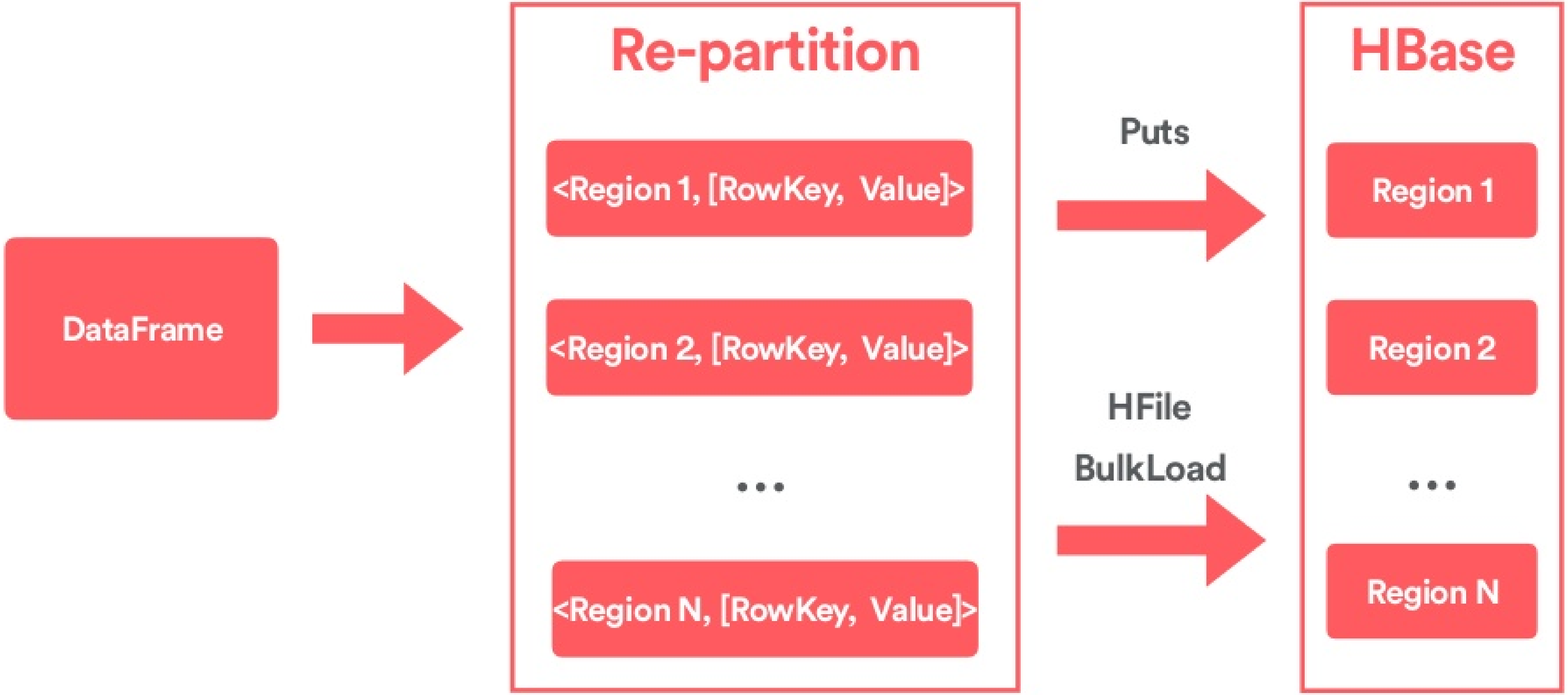
- Well integrated with Hadoop eco system

- Efficient API for streaming writes and bulk uploads

- Rich API for sequential scan and point-lookups

- Merged view based on version

Unified Write API



DataFrame

Re-partition

<Region 1, [RowKey, Value]>

<Region 2, [RowKey, Value]>

...

<Region N, [RowKey, Value]>

Puts

HFile BulkLoad

HBase

Region 1

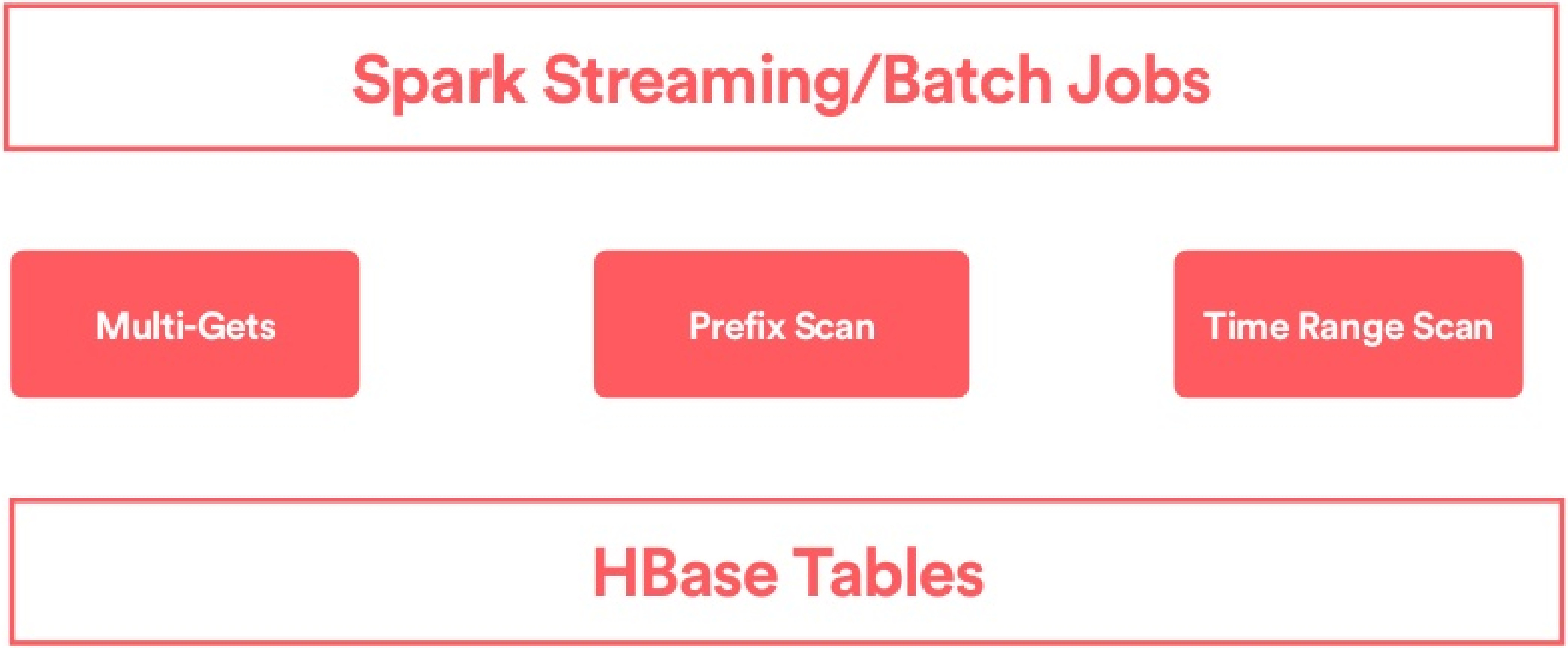
Region 2

...

Region N

Liyin Tang and Jingwei Lu

Rich Read API



Spark Streaming/Batch Jobs

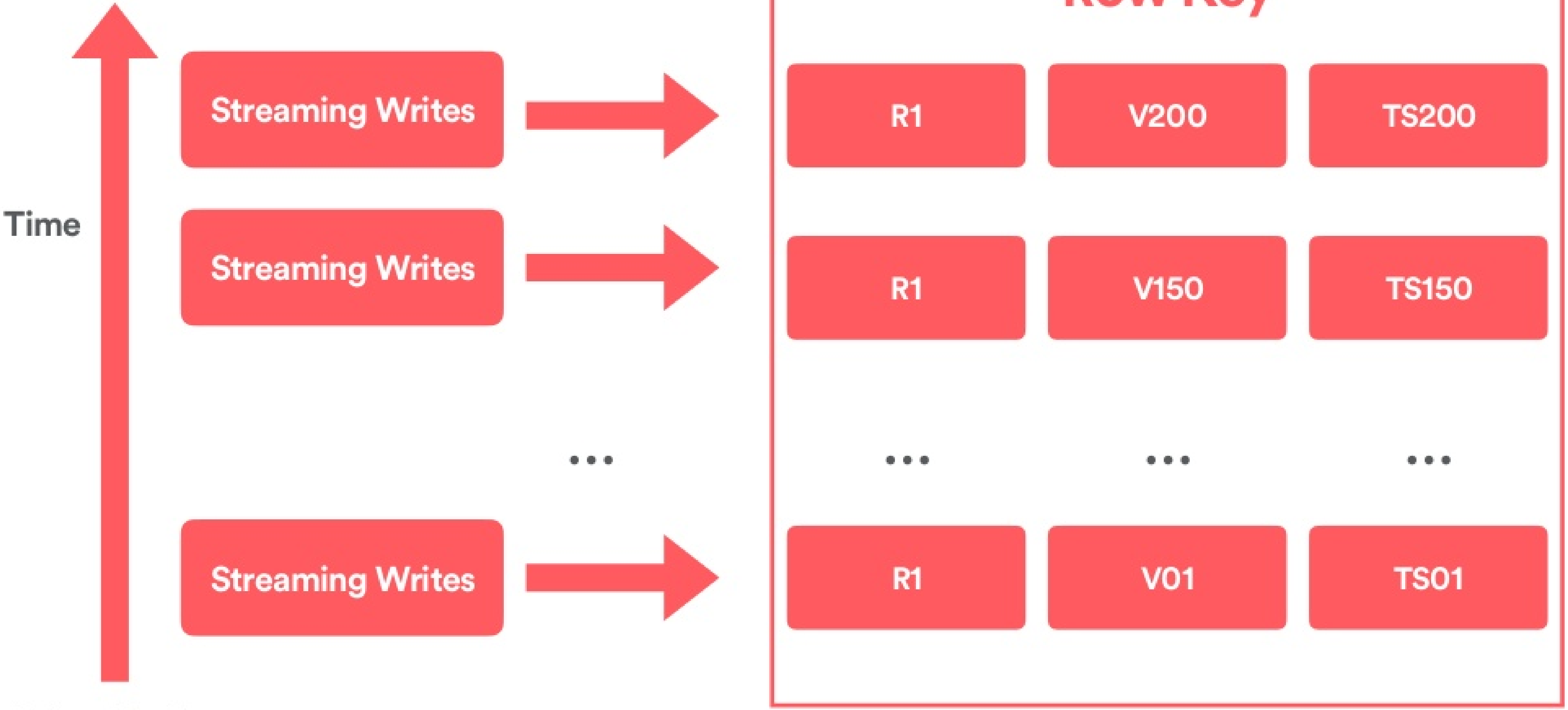
Multi-Gets

Prefix Scan

Time Range Scan

HBase Tables

Merged Views



Time

Streaming Writes

Streaming Writes

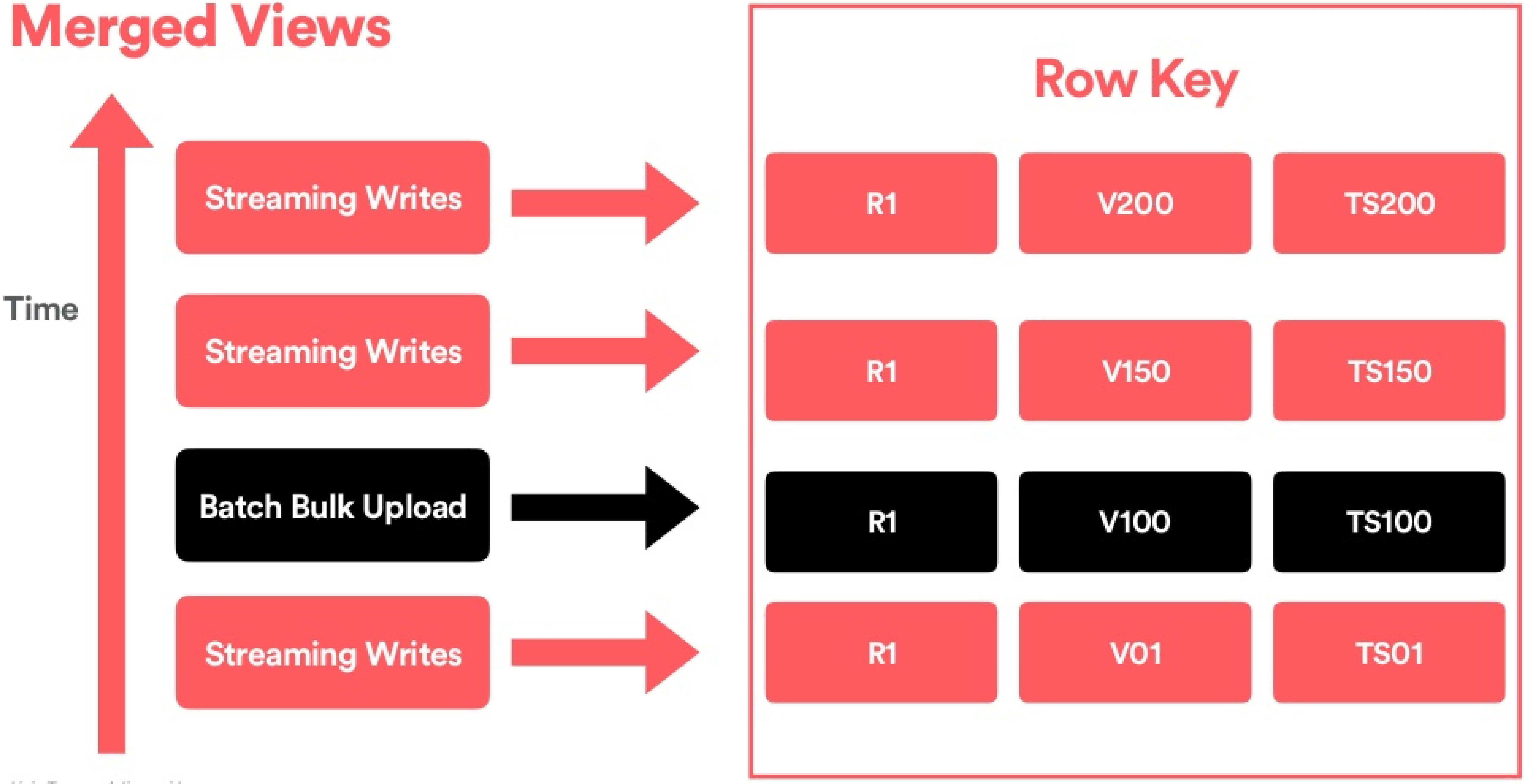
…

Streaming Writes

Row Key

R1	V200	TS200
R1	V150	TS150
…	…	…
R1	V01	TS01

Merged Views



Row Key

R1	V200	TS200
R1	V150	TS150
R1	V100	TS100
R1	V01	TS01

Our Foundations

- Unify streaming with batch process
- Shared global state store

Use Cases

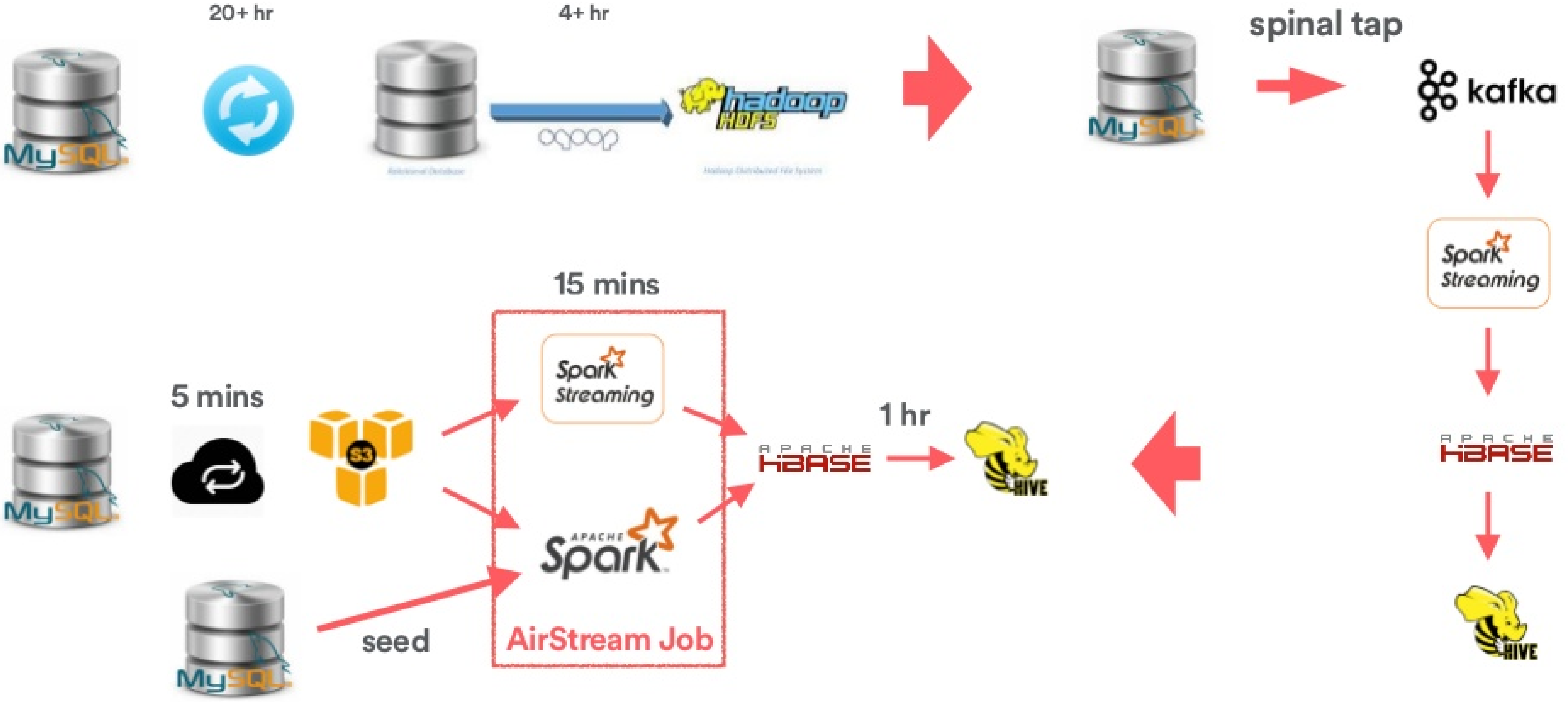
MySQL DB Snapshot Using Binlog Replay

Move Elephant

Database Snapshot

- **Large amount of data:** Multiple large mysql DBs
- **Realtime-ness:** minutes delay/ hours delay
- **Transaction :** Need to keep transaction across different tables
- **Schema change:** Table schema evolves

Binlog Replay on Spark



20+ hr

4+ hr

spinal tap

15 mins

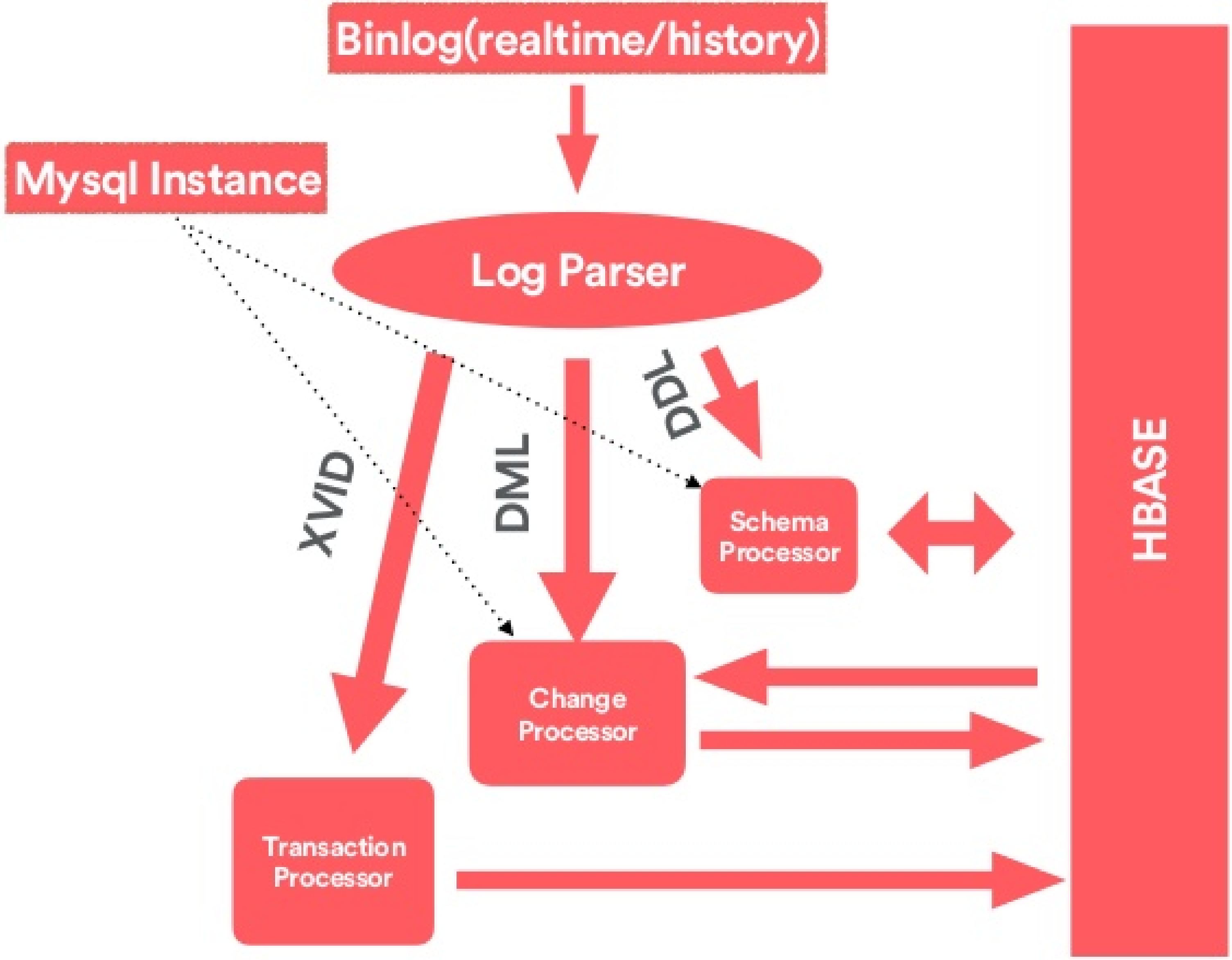
5 mins

1 hr

seed

AirStream Job

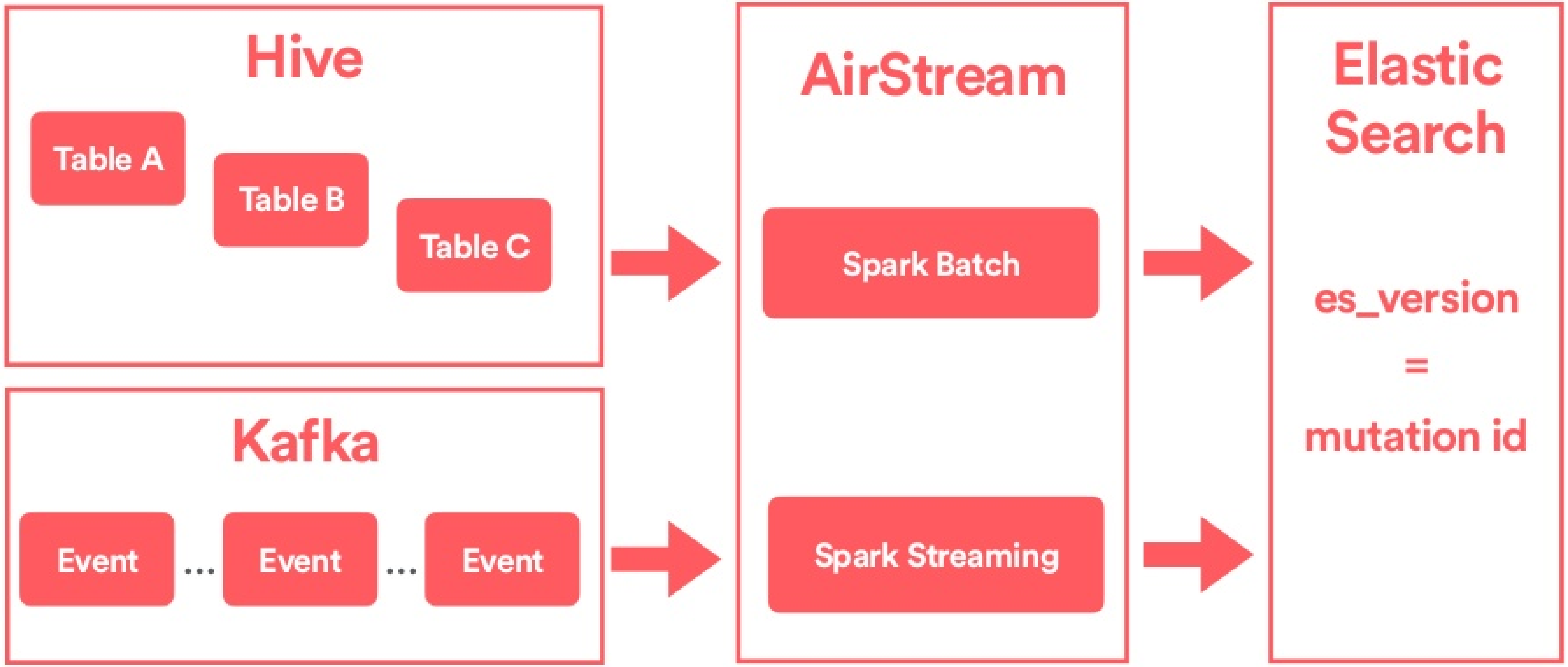
Lambda Architecture



- **Streaming and Batch shares Logic:** Binlog file reader, DDL processor, transaction processor, DML processor.
- **Merged by binlog position:** <filenum, offset>
- **Idempotent:** Log can be replayed multiple times.
- **Schema changes:** Full schema change history.

Realtime Indexing

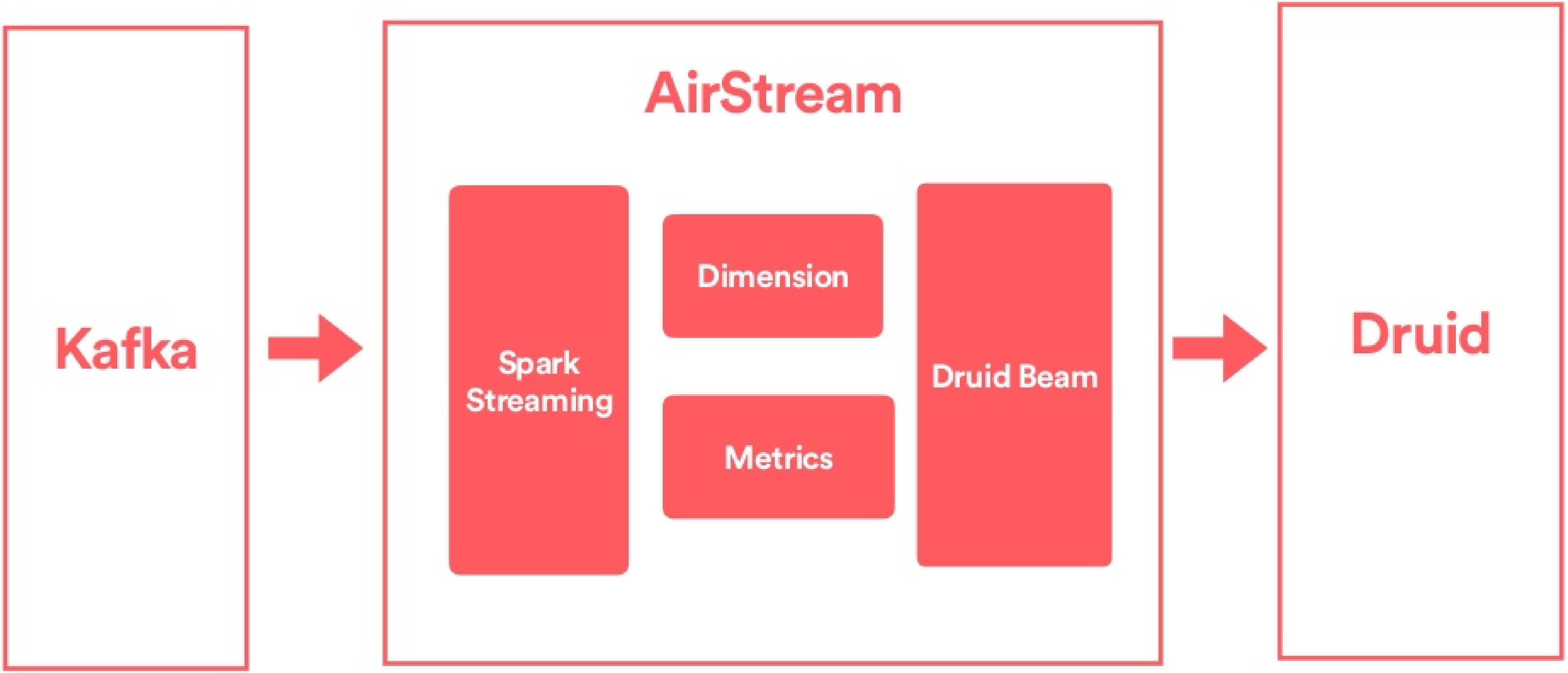
Realtime Indexing



Liyin Tang and Jingwei Lu

Realtime OLAP with Druid

Druid Ingestion



Kafka

AirStream

Spark Streaming

Dimension

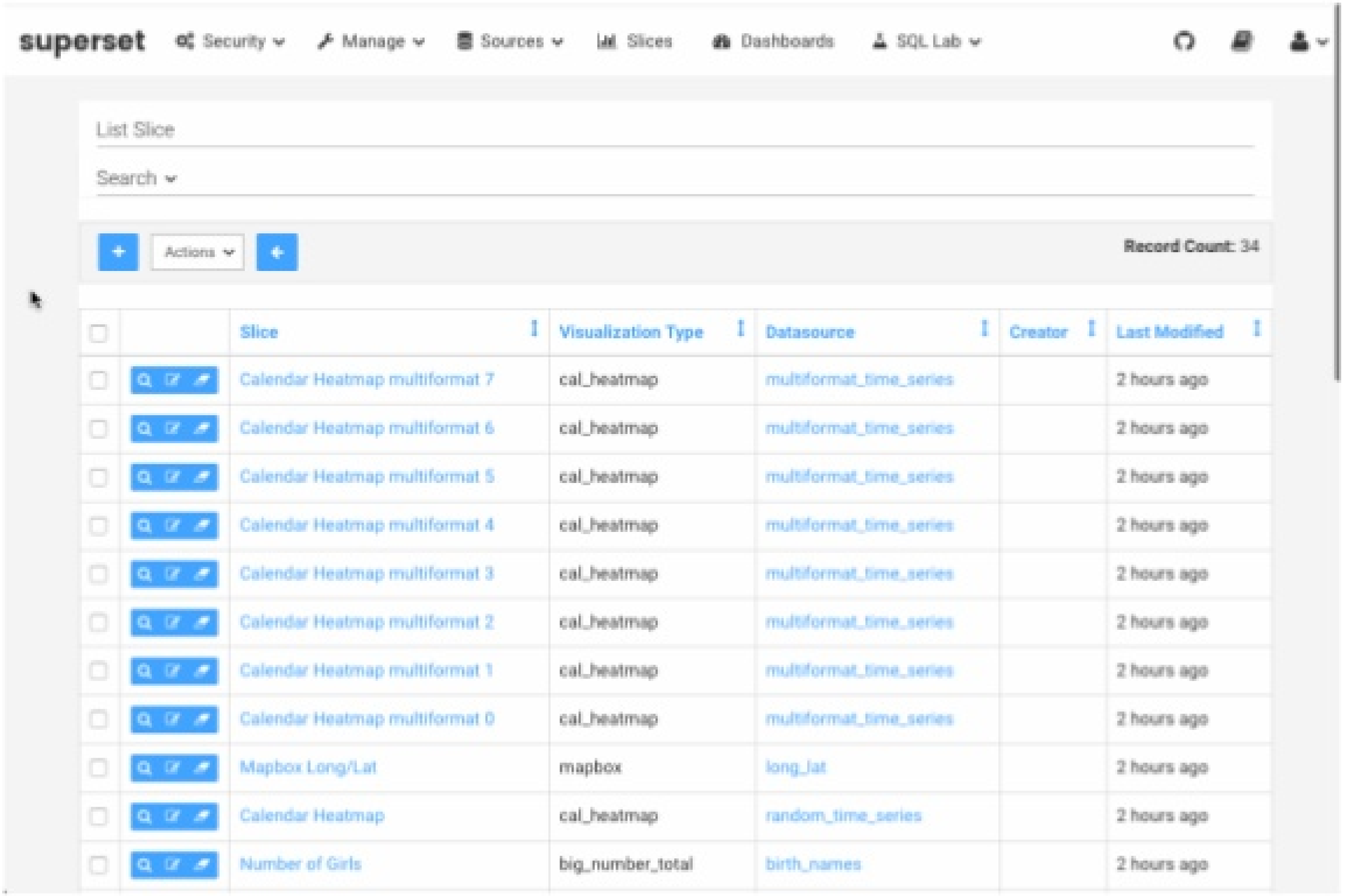
Metrics

Druid Beam

Druid

Liyin Tang and Jingwei Lu

Superset Powered by Druid



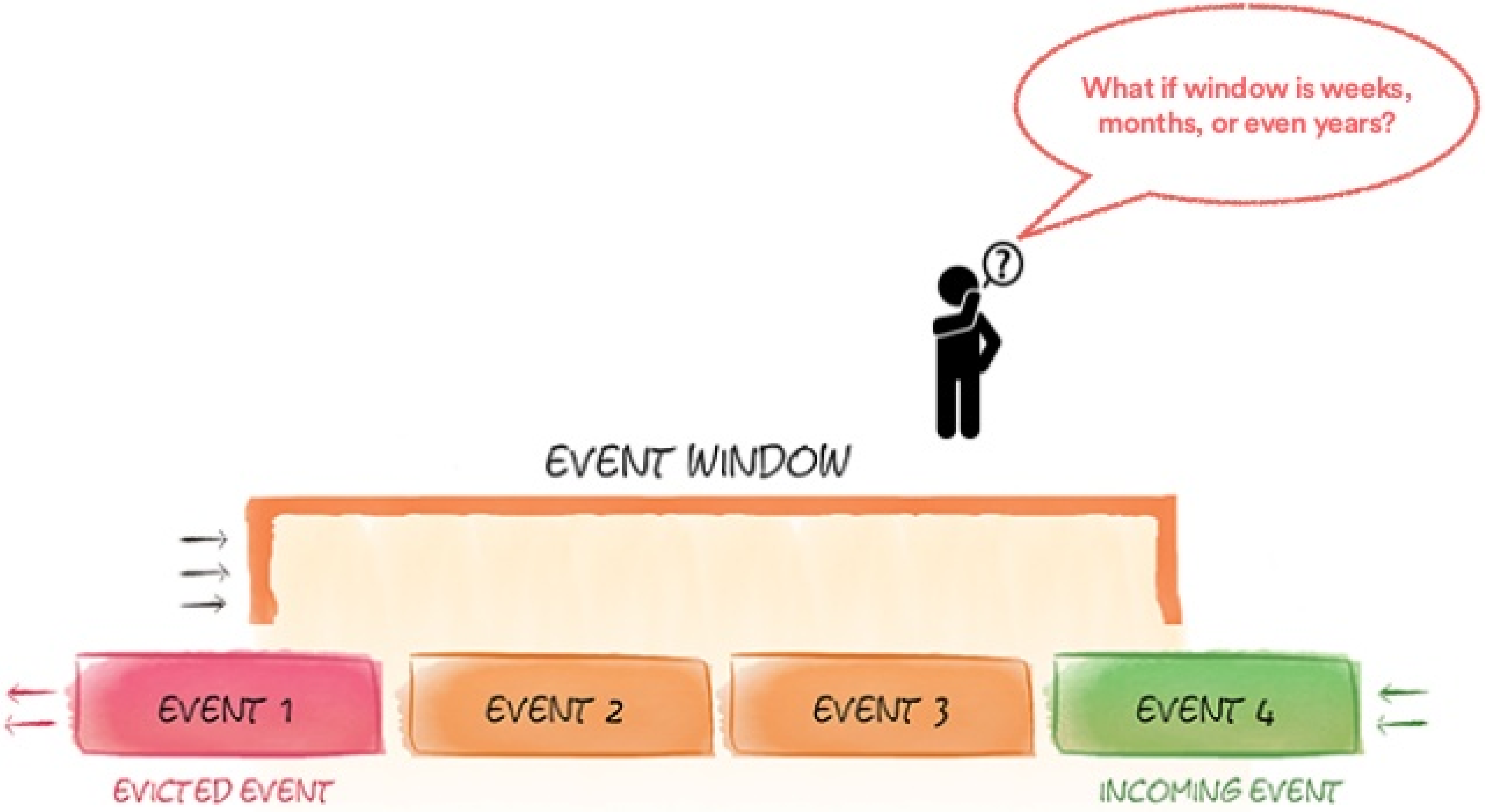
Record Count: 34

□		Slice	Visualization Type	Datasource	Creator	Last Modified
□		Calendar Heatmap multiformat 7	cal_heatmap	multiformat_time_series		2 hours ago
□		Calendar Heatmap multiformat 6	cal_heatmap	multiformat_time_series		2 hours ago
□		Calendar Heatmap multiformat 5	cal_heatmap	multiformat_time_series		2 hours ago
□		Calendar Heatmap multiformat 4	cal_heatmap	multiformat_time_series		2 hours ago
□		Calendar Heatmap multiformat 3	cal_heatmap	multiformat_time_series		2 hours ago
□		Calendar Heatmap multiformat 2	cal_heatmap	multiformat_time_series		2 hours ago
□		Calendar Heatmap multiformat 1	cal_heatmap	multiformat_time_series		2 hours ago
□		Calendar Heatmap multiformat 0	cal_heatmap	multiformat_time_series		2 hours ago
□		Mapbox Long/Lat	mapbox	long_lat		2 hours ago
□		Calendar Heatmap	cal_heatmap	random_time_series		2 hours ago
□		Number of Girls	big_number_total	birth_names		2 hours ago

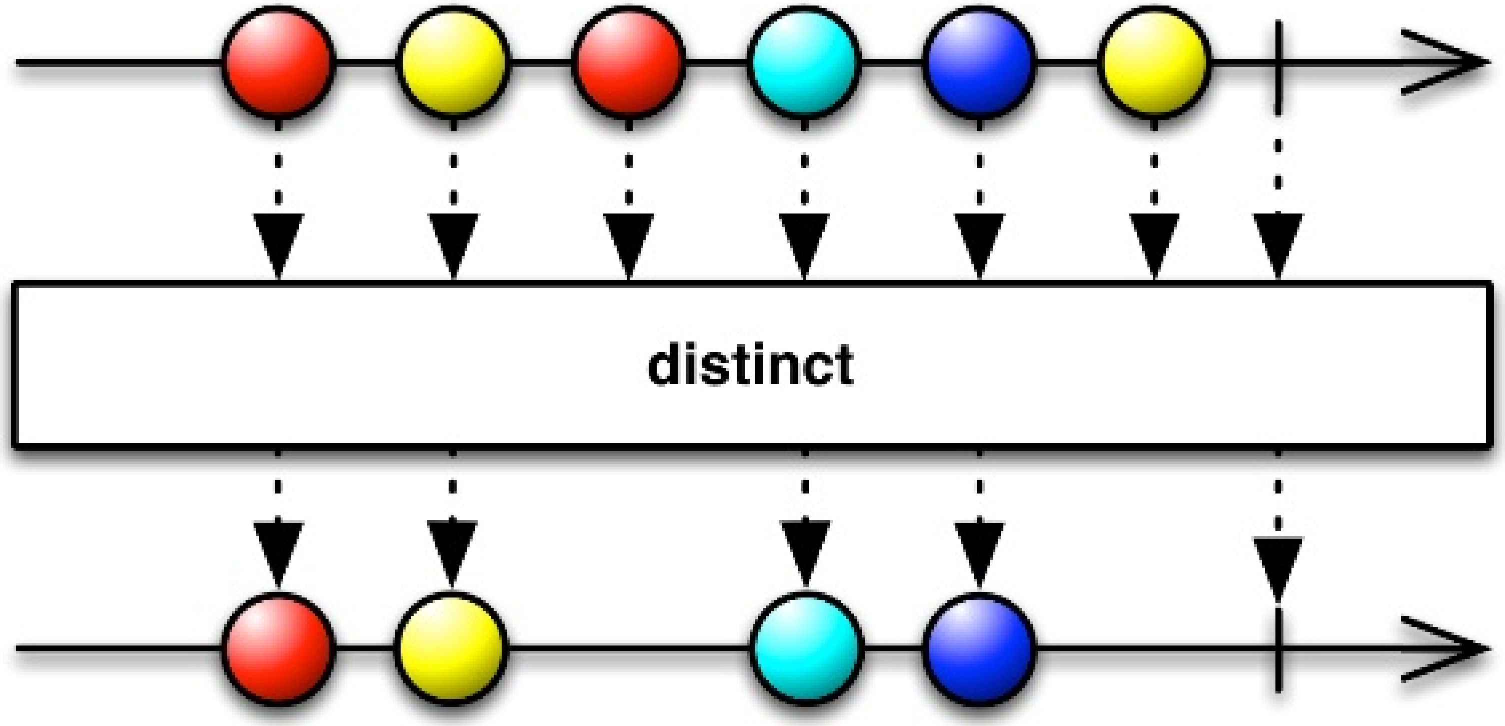
Tips

Moving Window Computation

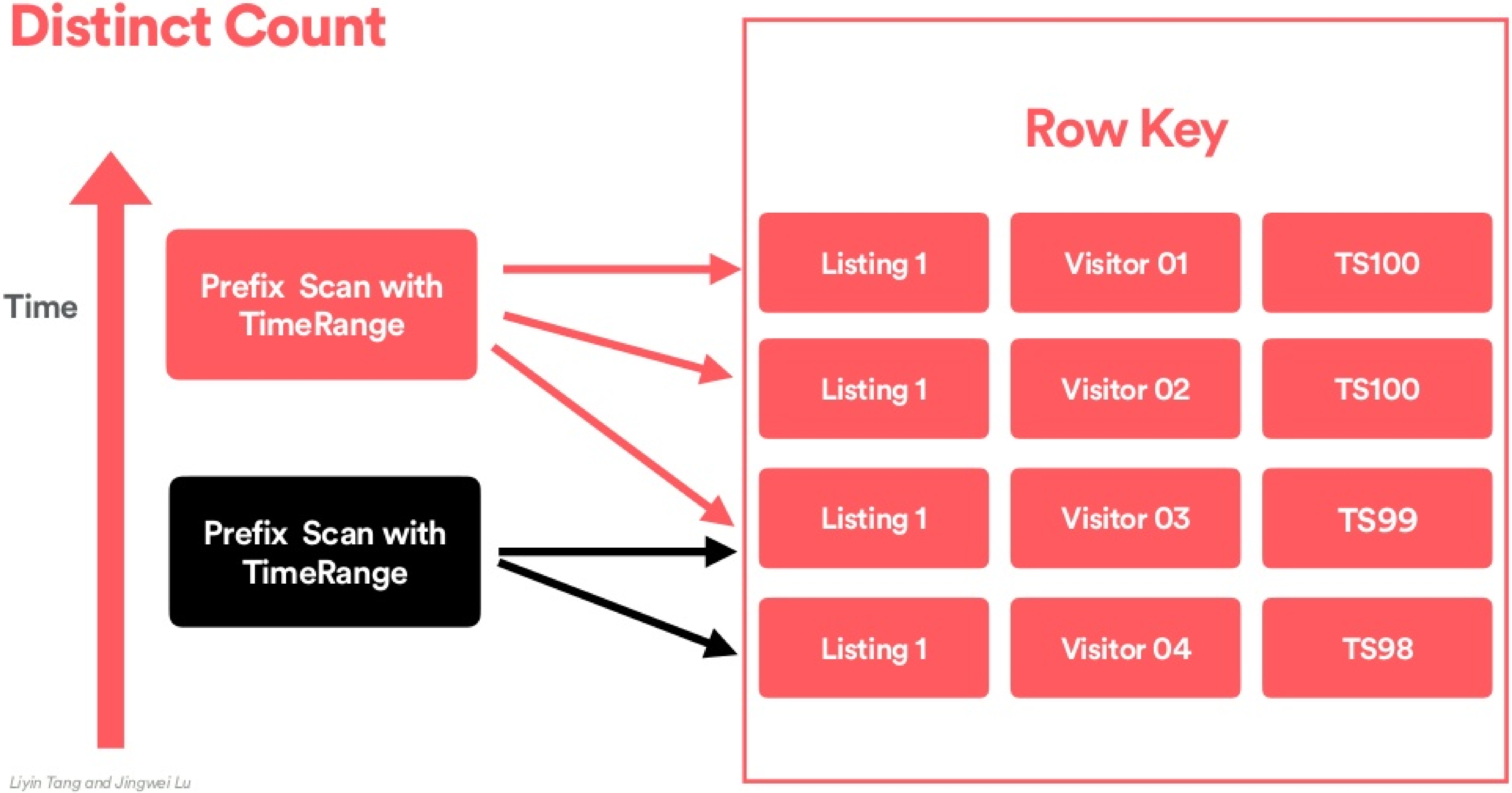
Long Window Computation



Distinct in a Large Window

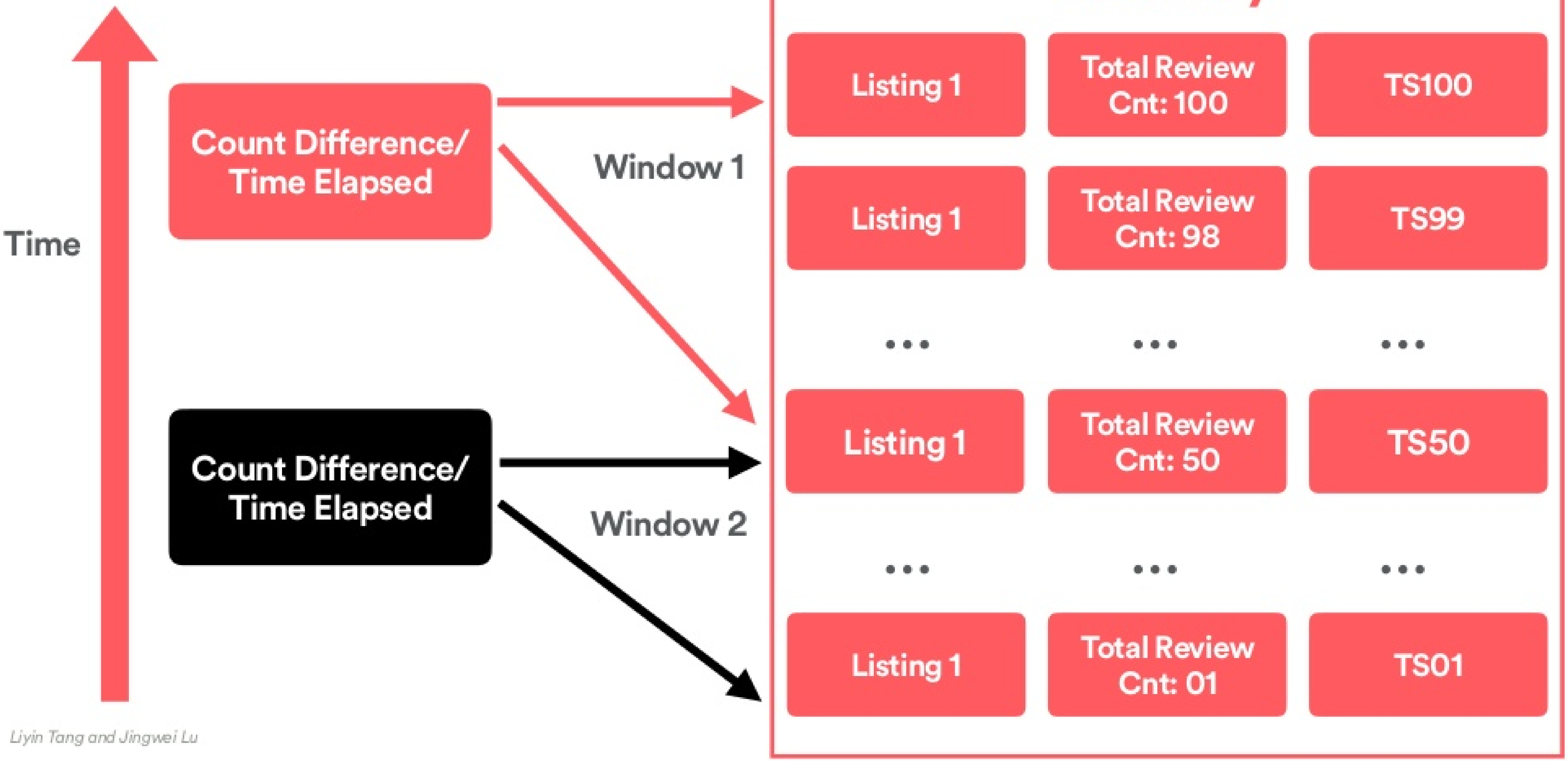


Distinct Count



Listing 1	Visitor 01	TS100
Listing 1	Visitor 02	TS100
Listing 1	Visitor 03	TS99
Listing 1	Visitor 04	TS98

Moving Average



Time

Count Difference/ Time Elapsed

Window 1

Count Difference/ Time Elapsed

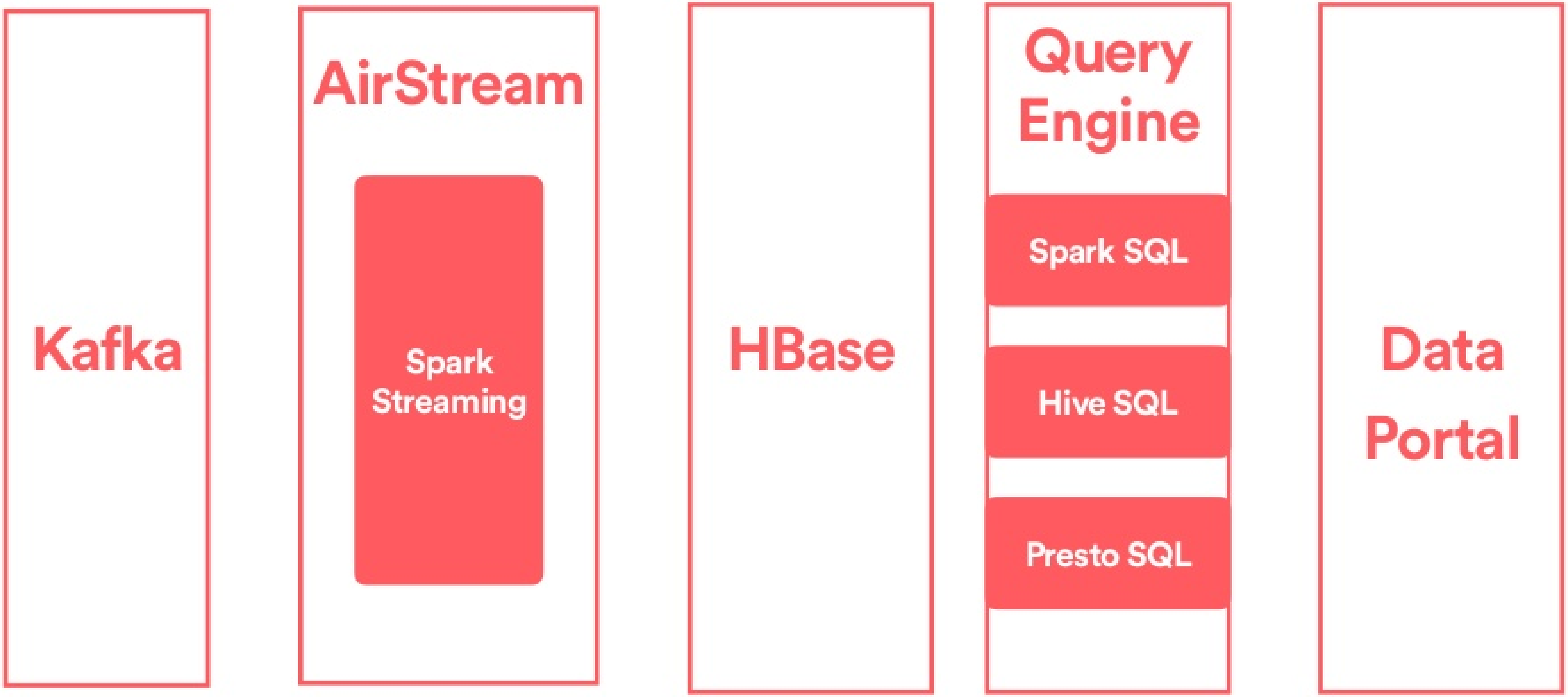
Window 2

Row Key

Listing 1	Total Review Cnt: 100	TS100
Listing 1	Total Review Cnt: 98	TS99
...	...	...
Listing 1	Total Review Cnt: 50	TS50
...	...	...
Listing 1	Total Review Cnt: 01	TS01

Streaming Ingestion & Realtime Interactive Query

Realtime Ingestion and Interactive Query



Kafka

AirStream

Spark Streaming

HBase

Query Engine

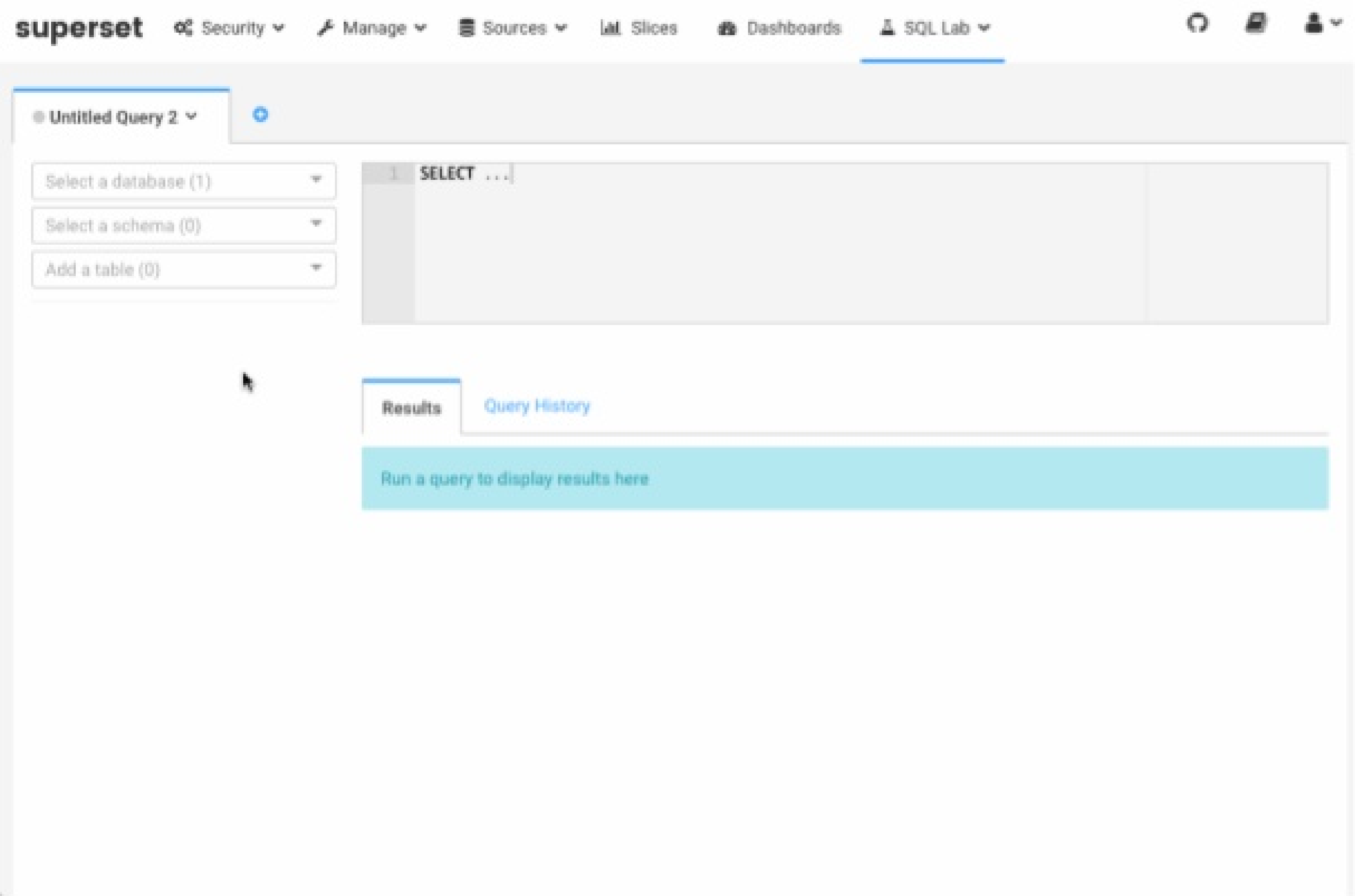
Spark SQL

Hive SQL

Presto SQL

Data Portal

Interactive Query in SqlLab

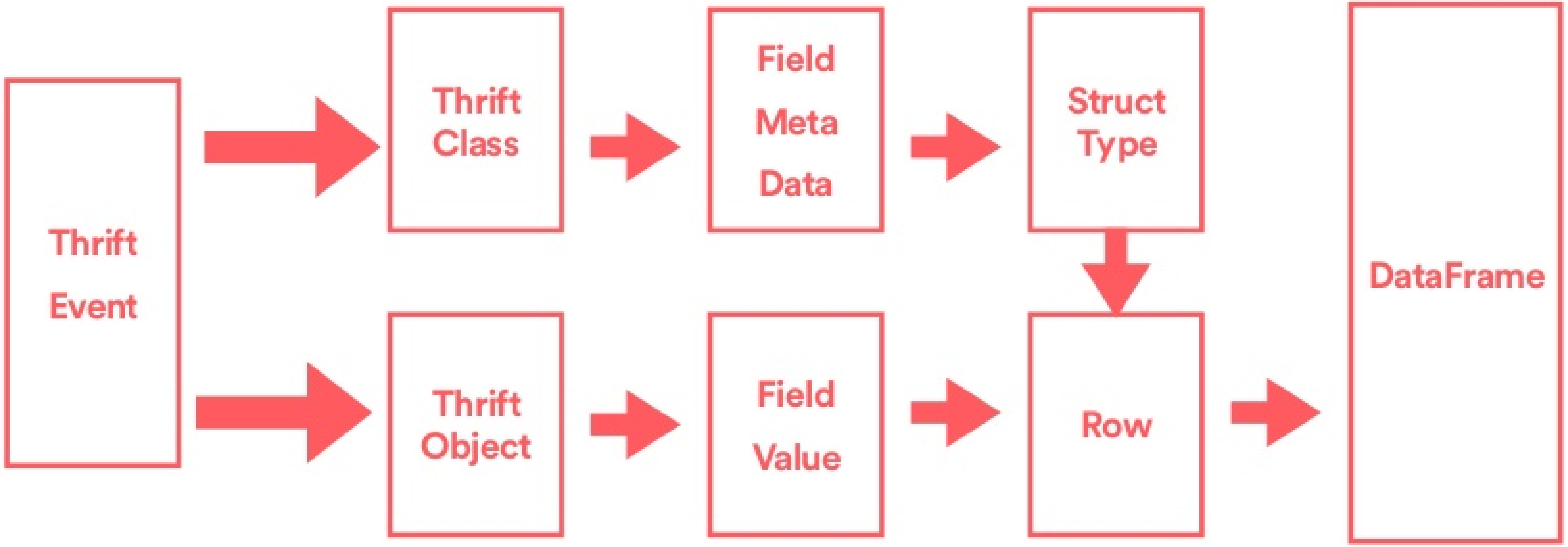


```
SELECT ...
```

Run a query to display results here

Schema Enforcement
Streaming Events

Thrift -> DataFrame



https://github.com/airbnb/airbnb-spark-thrift

Summary

Unify Batch and Streaming Computation

Global State Store Using HBase

airbnb

We are hiring

Happy Hour:
6pm, B Restaurant & Bar, 720 Howard St, SF