KubeCon | CloudNativeCon

North America 2020

Virtual

Elastic Scheduling with TiKV

Song Gao, PingCAP
Yutong Liang, PingCAP

Speaker

Yutong Liang
Engineer at PingCAP

Database engineer
Technical lead of TiKV SIG Scheduling

Github: @rleungx

Song Gao
Engineer at PingCAP

Database engineer
Maintainer of Chaos Mesh®
Committer of TiKV SIG scheduling

Github: @Yisaer

Agenda

- Introduction to TiKV
- Elastic Scheduling background
- Implementation in TiKV
- Future work
- Q&A

Introduction

What is TiKV?

TiKV is an open source **distributed** **transactional** **key-value database**.

CLOUD NATIVE
COMPUTING FOUNDATION

CNCF Graduated

8.2K

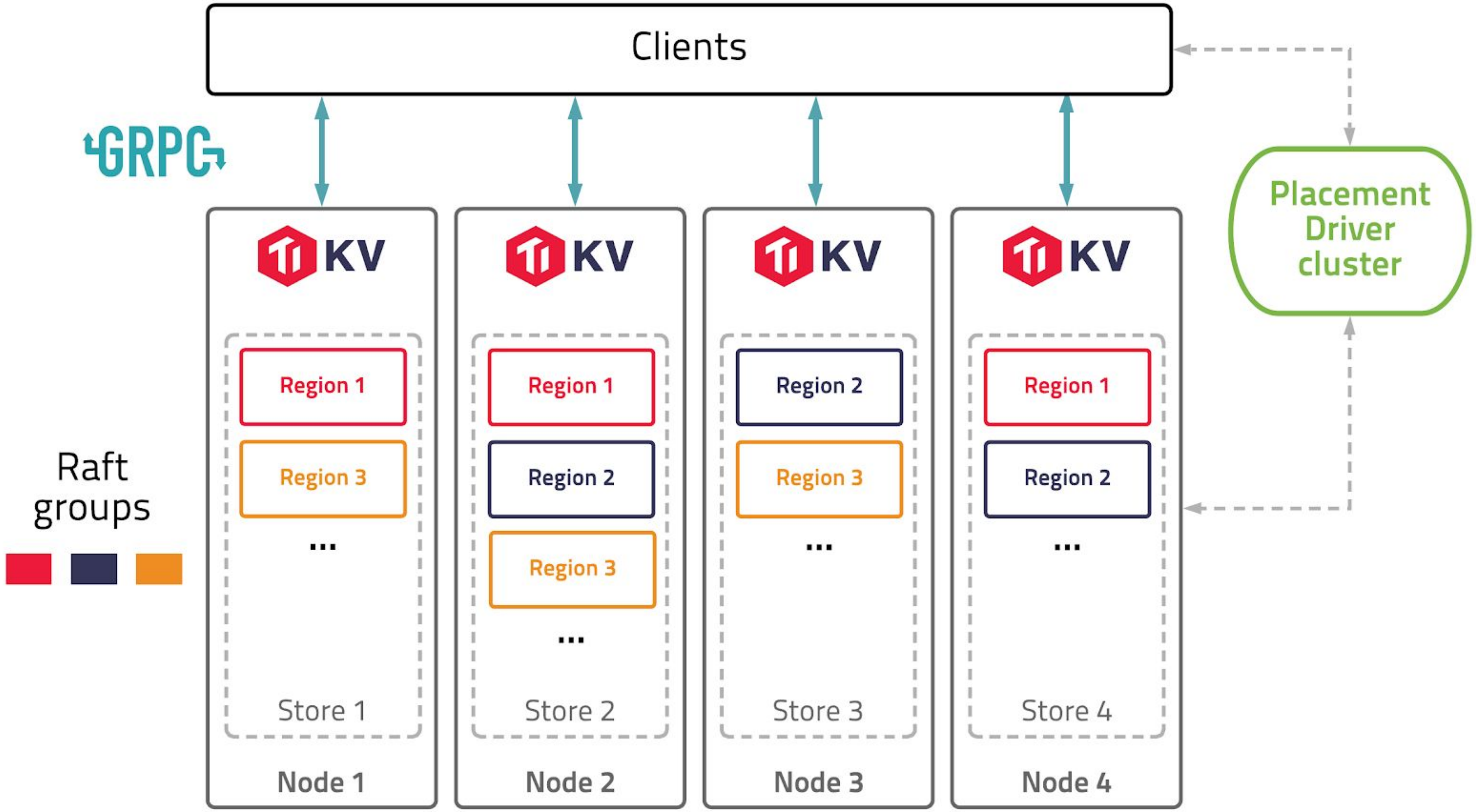
GitHub Stars

264

Contributors

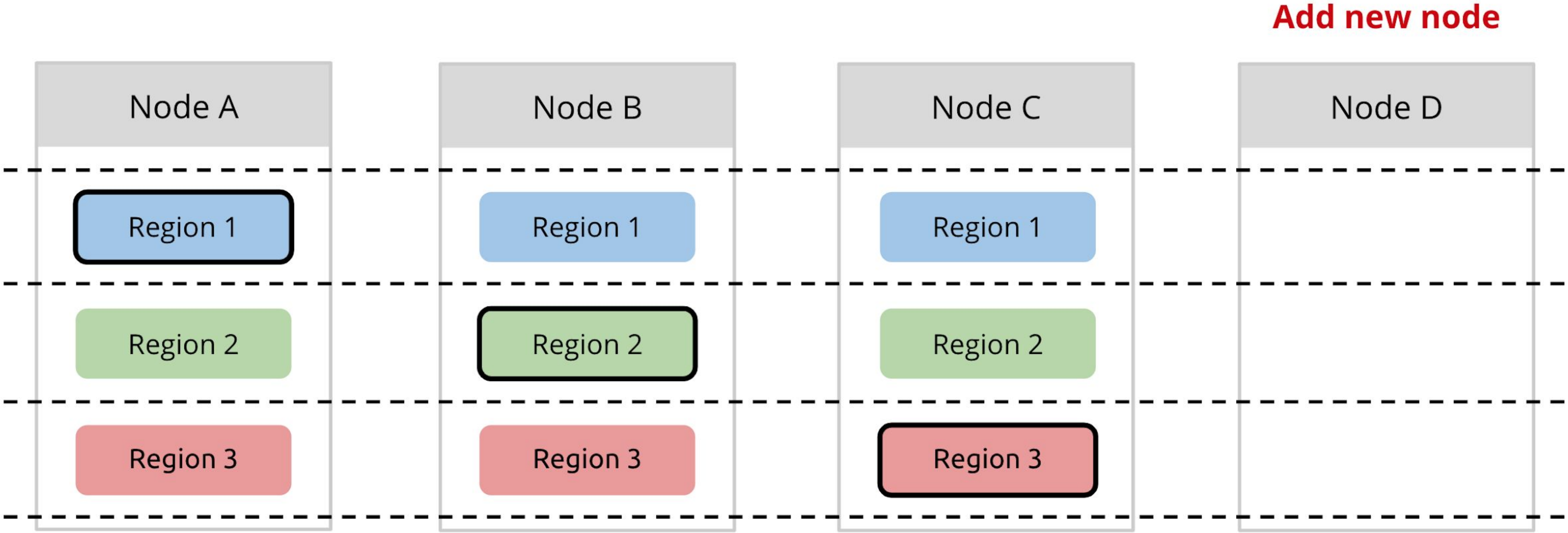
Introduction

TiKV Architecture



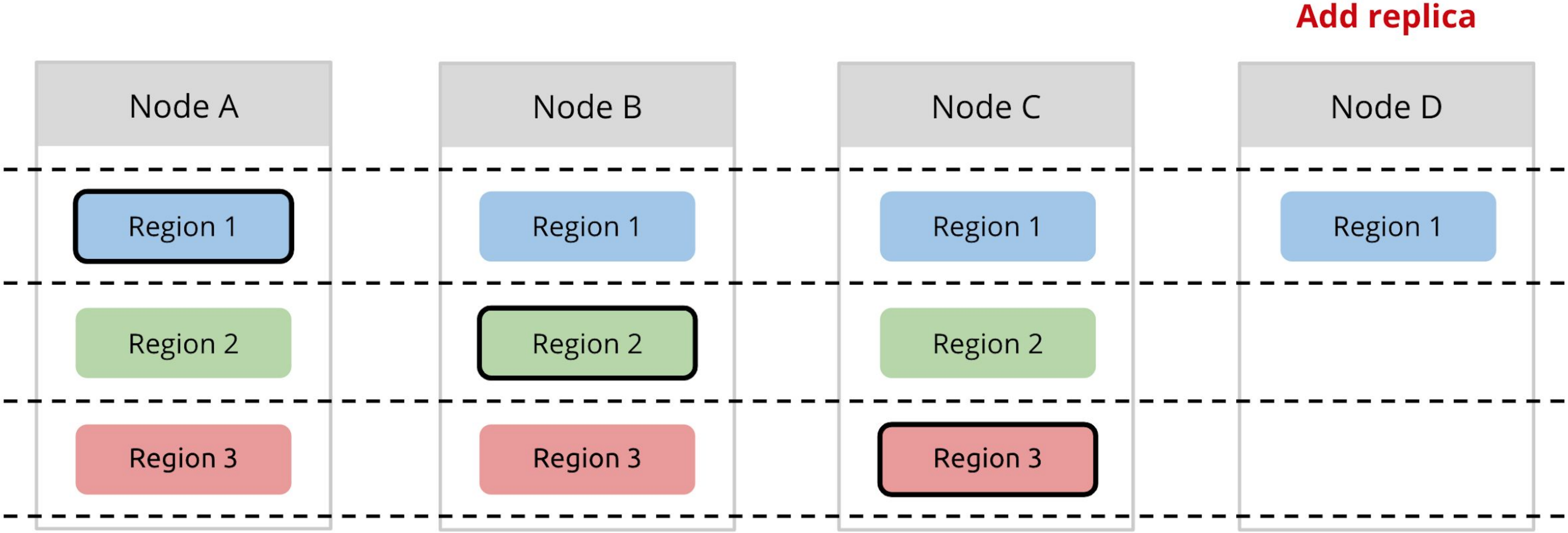
Introduction

- Add a new Node D



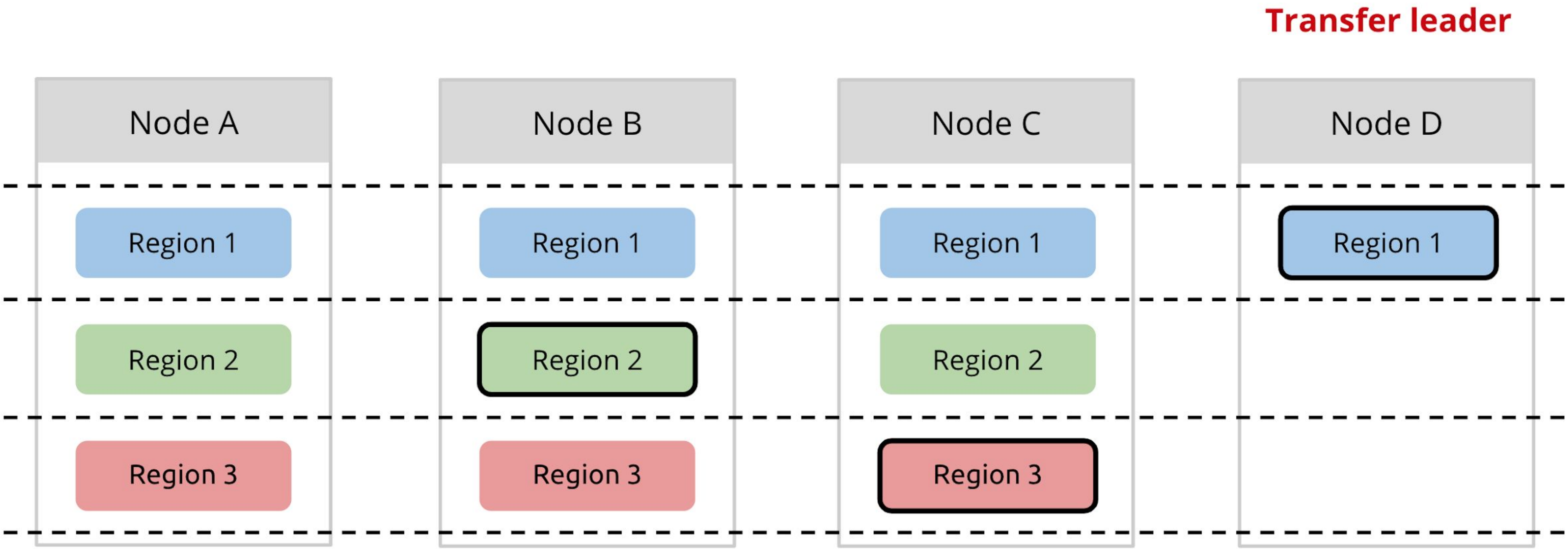
Introduction

- Add a replica of Region 1 in Node D



Introduction

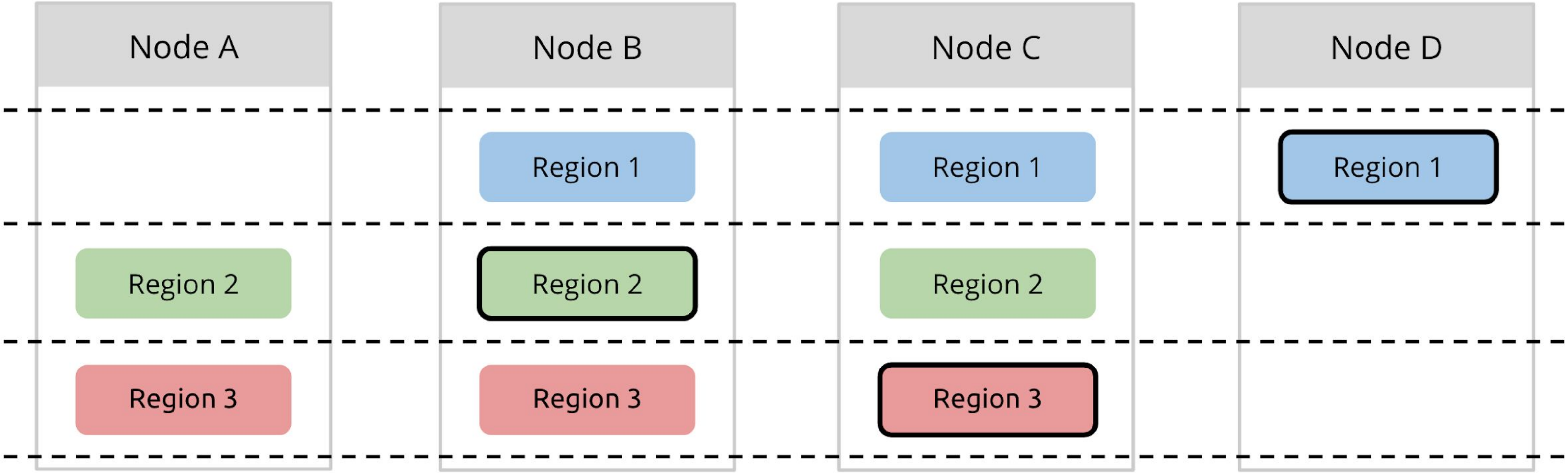
- Transfer leader of Region 1 from Node A to Node D



Introduction

- Remove the original replica of Region 1 from Node A

Remove replica



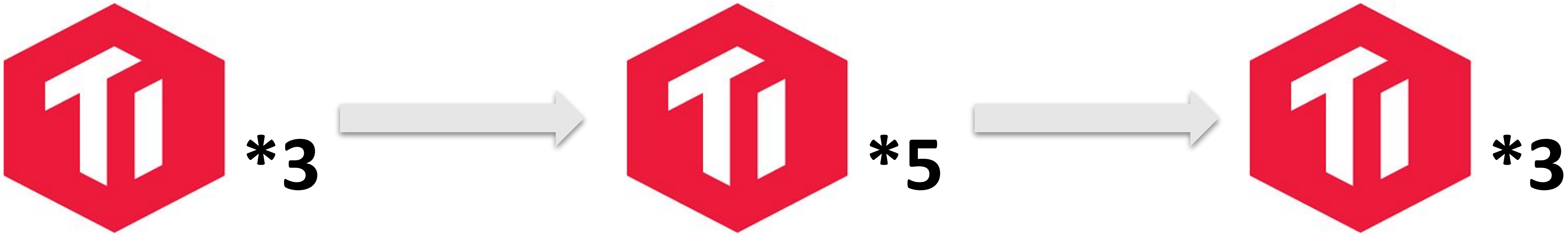
Node A	Node B	Node C	Node D
	Region 1	Region 1	Region 1
Region 2	Region 2	Region 2	
Region 3	Region 3	Region 3	

Elastic scheduling

What is Elastic Scheduling?

Elastic scheduling

Auto scaling by workloads



*3 *5 *3

The load increases

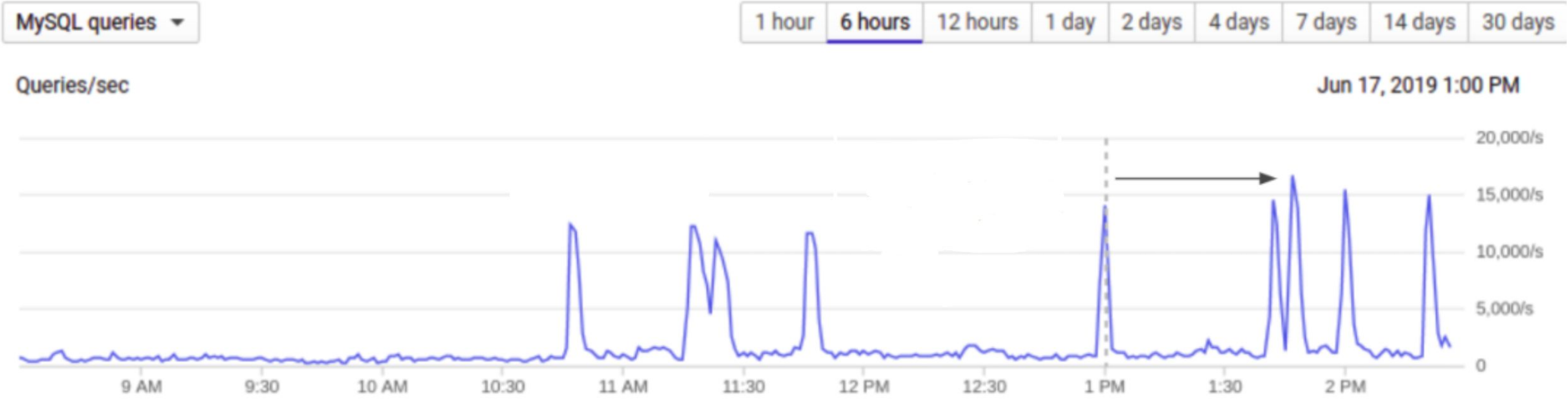
The load decreases

Elastic scheduling

Why Elastic Scheduling?

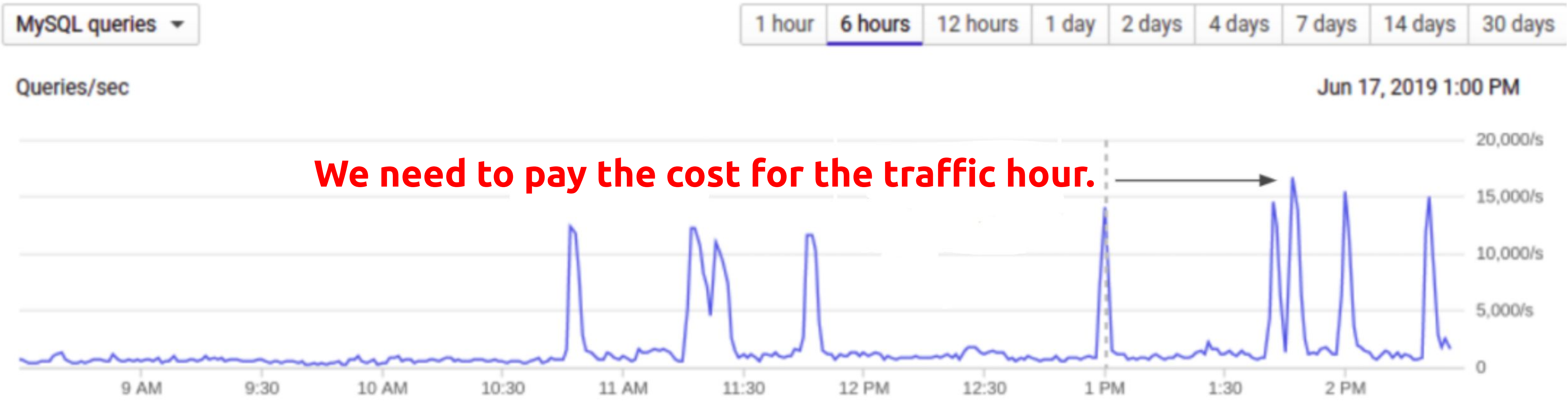
Elastic scheduling

The traffic is unexpected



Elastic scheduling

Some resources are wasted



We need to pay the cost for the traffic hour.

Elastic scheduling

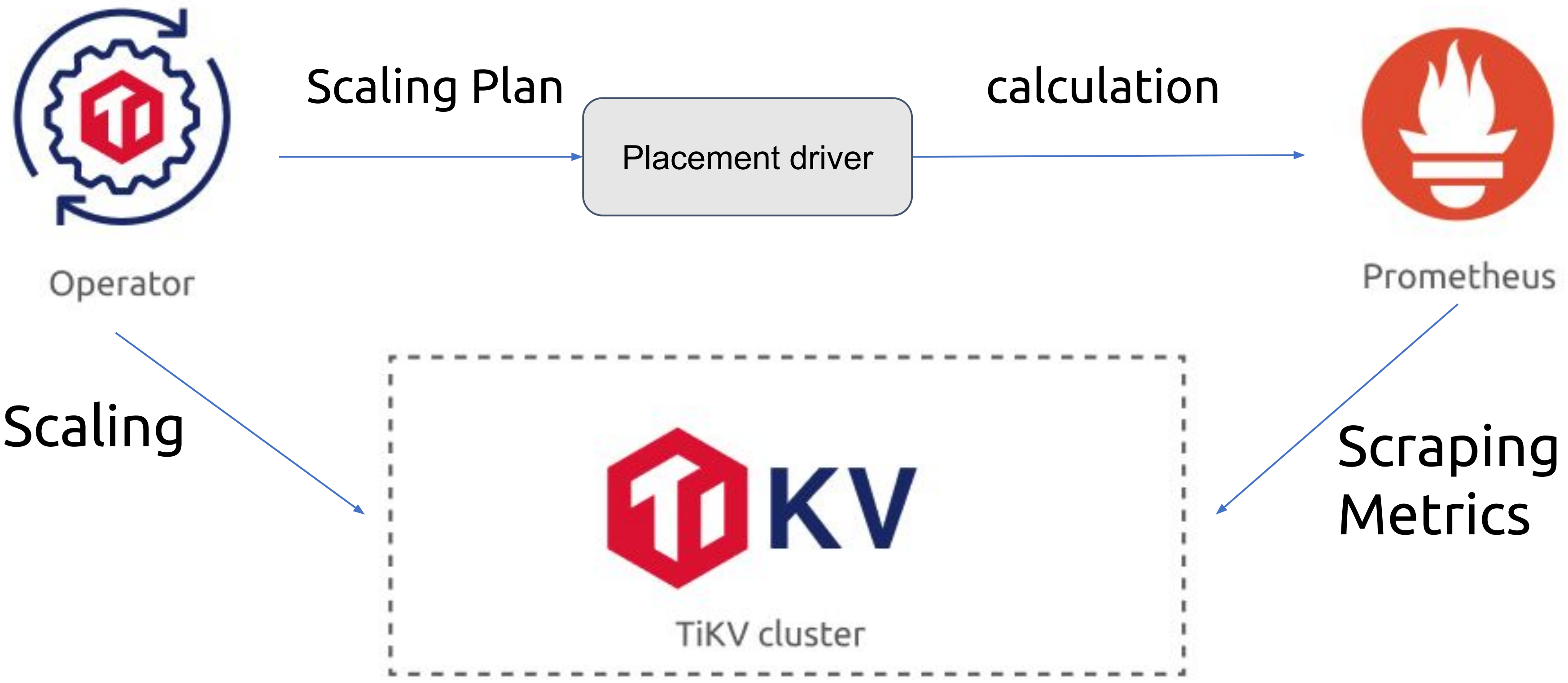
The cloud infra becomes mature.

Implementation

Implementation in TiKV

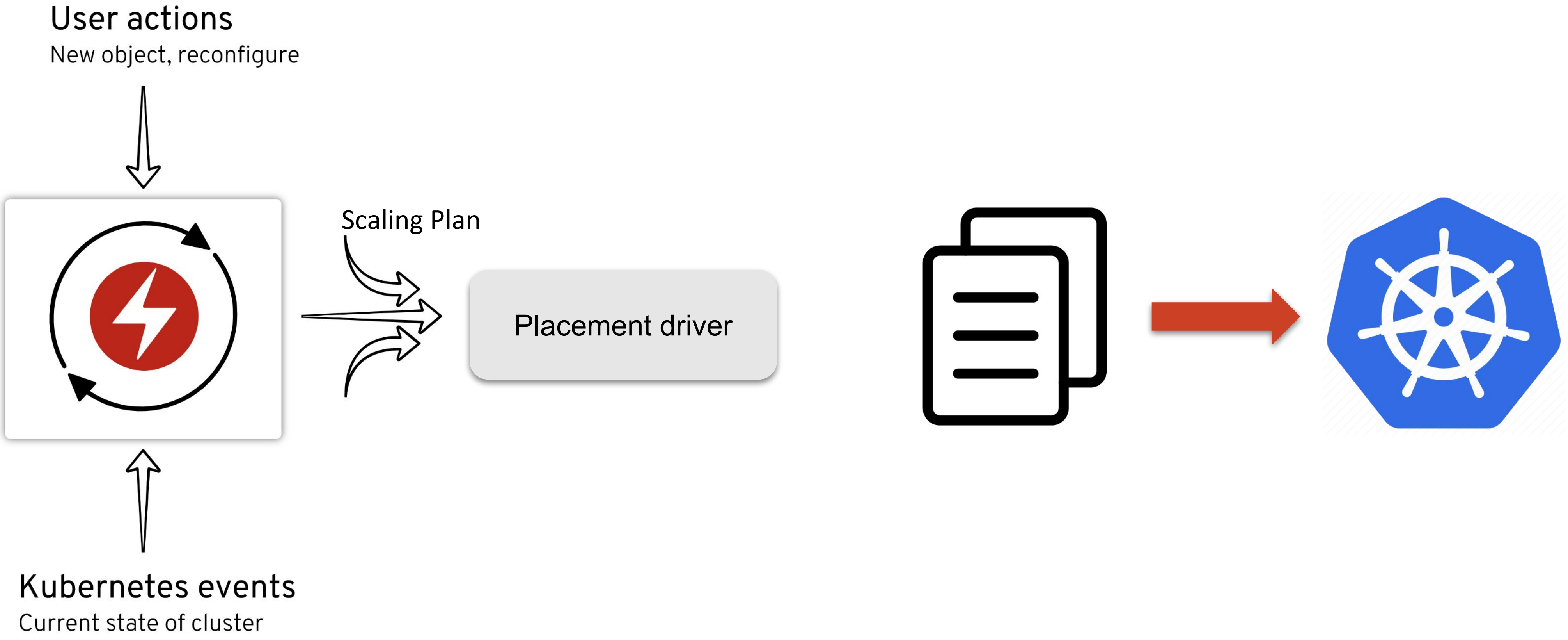
Implementation

Elastic scheduling architecture



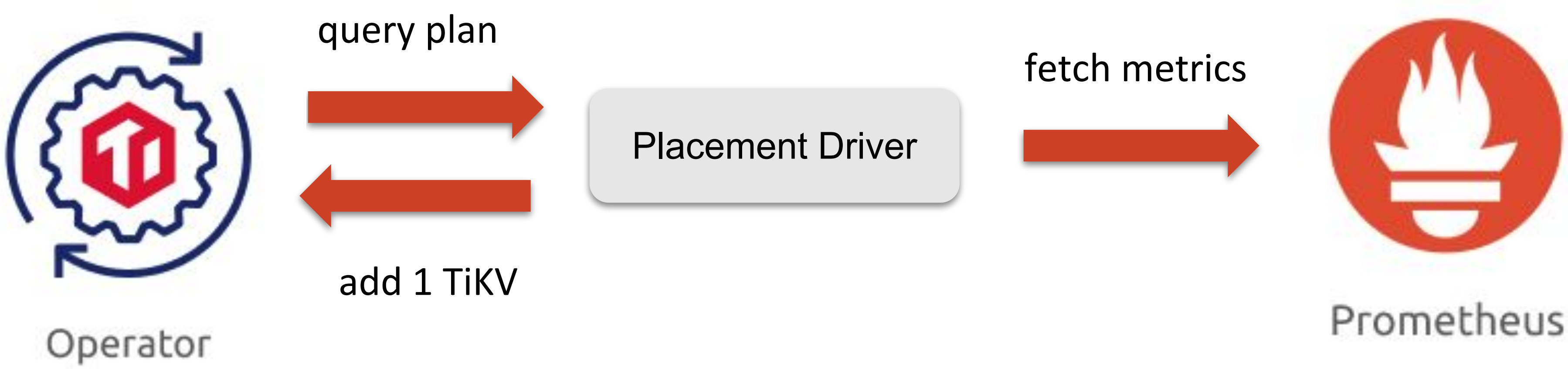
Implementation

Operator side



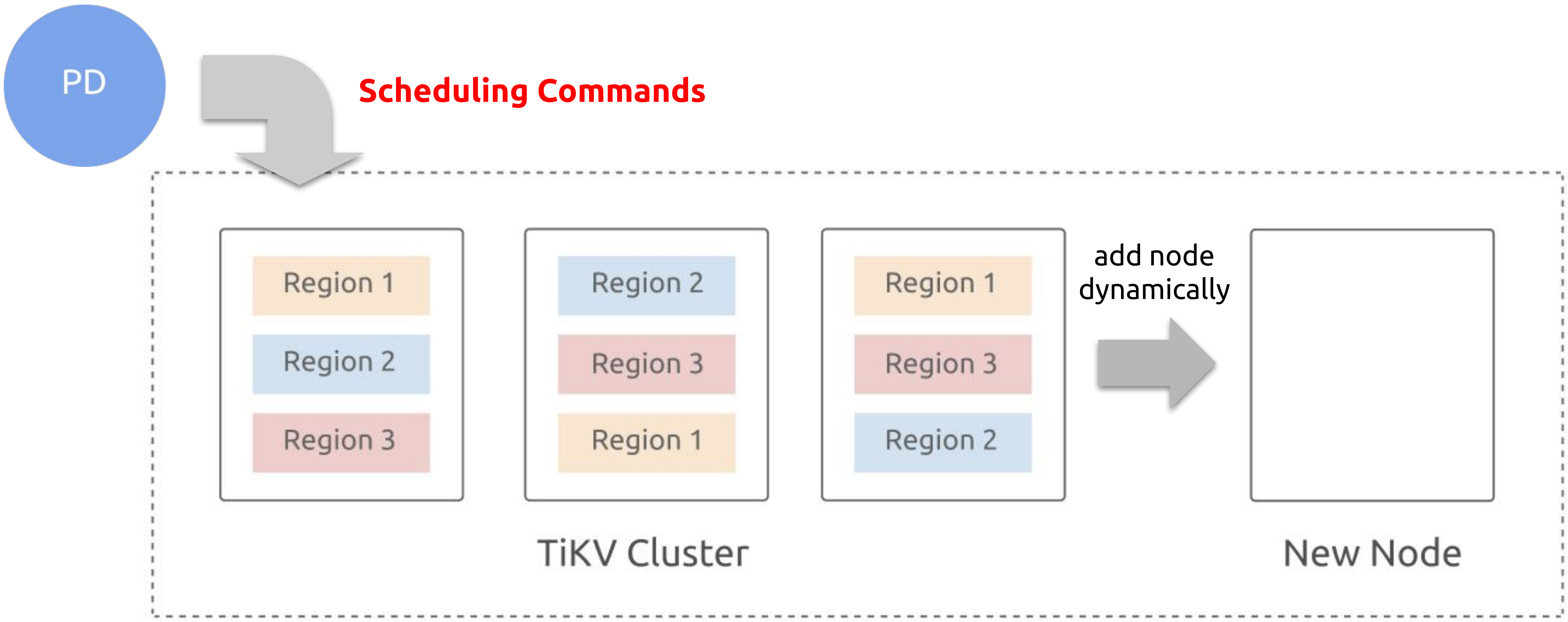
Implementation

Scheduling side



Implementation

Scheduling side



Implementation

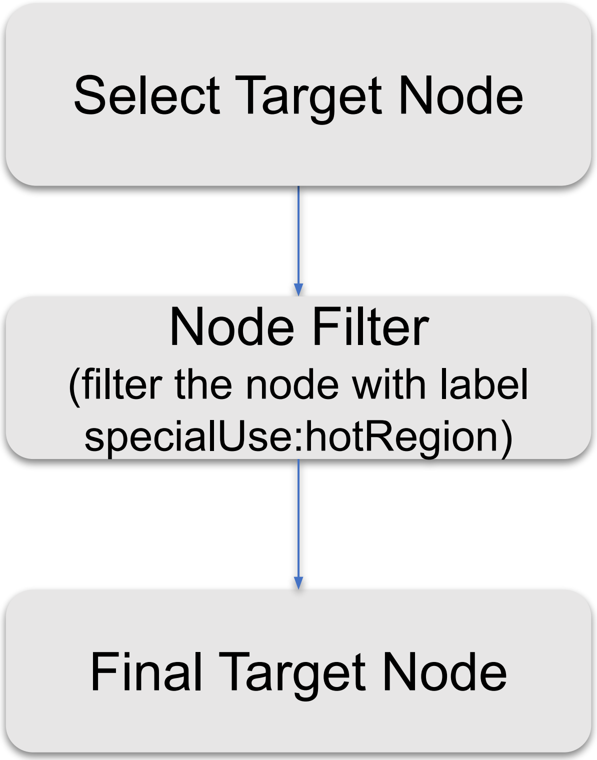
How does PD recognize the hot region?

- PD will maintain caches to record the top N Region write/read flow of each store. The hot Region must meet two conditions:

    - continue to hit the cache
    - write/read flow no less than the minimum threshold

Implementation

For other schedulers


Select Target Node

Node Filter
(filter the node with label specialUse:hotRegion)

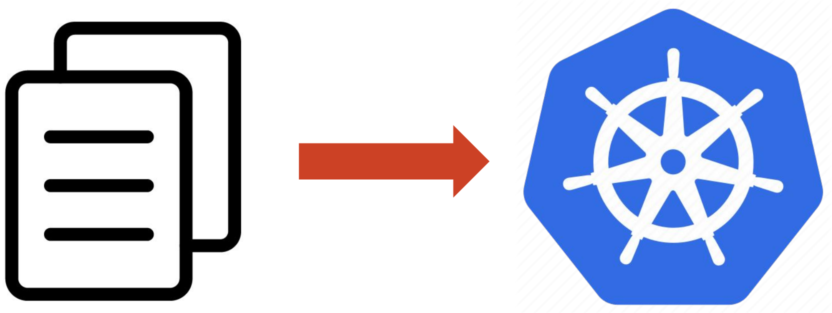
Final Target Node

```
...
"node":{
    "id":1,
    "address":"host:port",
    "labels":[{
            "key":"specialUse",
             "value":"hotRegion"
        }],
    ...
}
...
```

Demo

The API Overview

```yaml
...
spec:
  cluster:
    name: auto-scaling-demo
    namespace: default
  tikv:
    maxReplicas: 4
    metrics:
      - type: "Resource"
        resource:
          name: "cpu"
          target:
            type: "Utilization"
             averageUtilization: 80
```

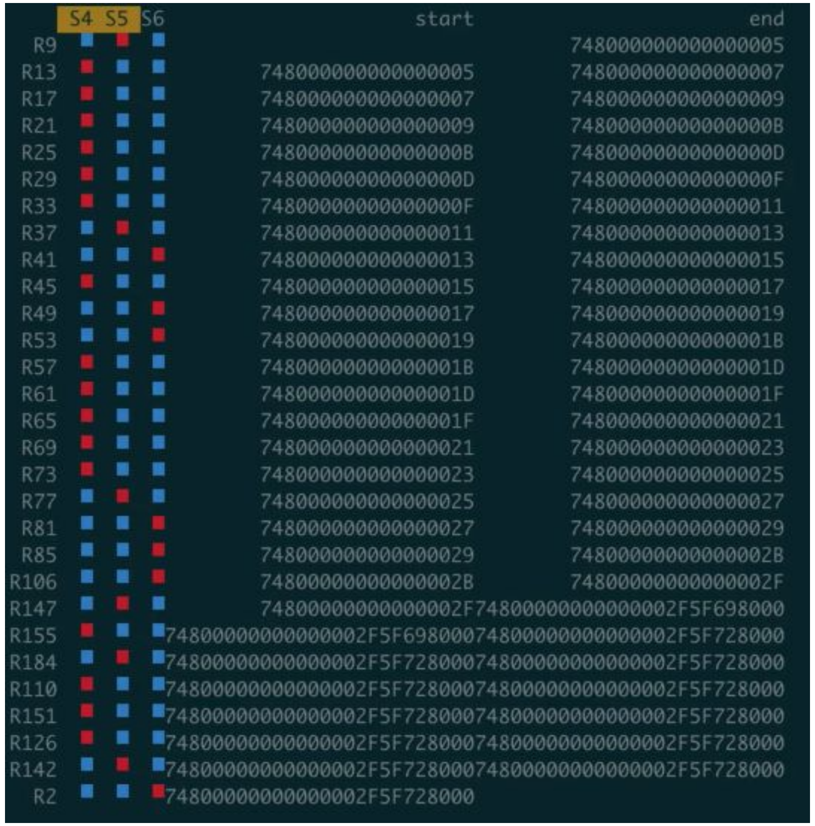


Demo

KubeCon CloudNativeCon North America 2020 Virtual

Initial State

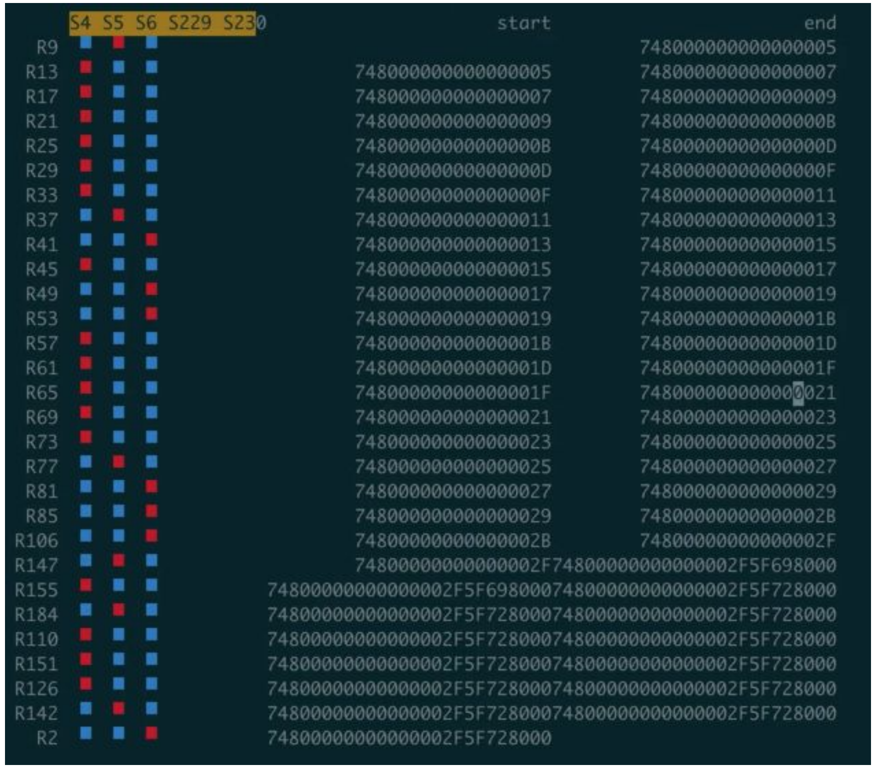
- 3 TiKV
- sysbench: oltp_read_only



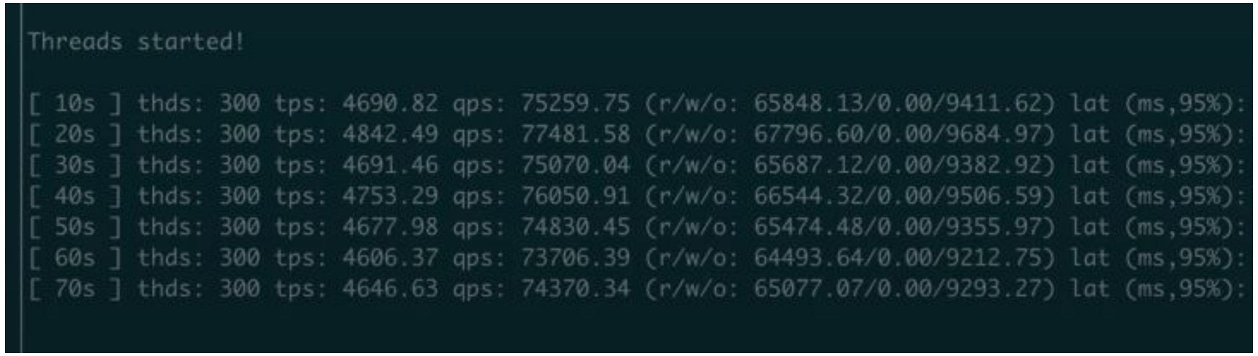
```
      S4 S5 S6             start                       end
  R9  🟦 🟥 🟦                             748000000000000005
 R13  🟥 🟦 🟦       748000000000000005    748000000000000007
 R17  🟥 🟦 🟦       748000000000000007    748000000000000009
 R21  🟥 🟦 🟦       748000000000000009    74800000000000000B
 R25  🟥 🟦 🟦       74800000000000000B    74800000000000000D
 R29  🟥 🟦 🟦       74800000000000000D    74800000000000000F
 R33  🟥 🟦 🟦       74800000000000000F    748000000000000011
 R37  🟦 🟥 🟦       748000000000000011    748000000000000013
 R41  🟦 🟦 🟥       748000000000000013    748000000000000015
 R45  🟥 🟦 🟦       748000000000000015    748000000000000017
 R49  🟦 🟦 🟥       748000000000000017    748000000000000019
 R53  🟦 🟦 🟥       748000000000000019    74800000000000001B
 R57  🟥 🟦 🟦       74800000000000001B    74800000000000001D
 R61  🟥 🟦 🟦       74800000000000001D    74800000000000001F
 R65  🟥 🟦 🟦       74800000000000001F    748000000000000021
 R69  🟥 🟦 🟦       748000000000000021    748000000000000023
 R73  🟥 🟦 🟦       748000000000000023    748000000000000025
 R77  🟦 🟥 🟦       748000000000000025    748000000000000027
 R81  🟦 🟦 🟥       748000000000000027    748000000000000029
 R85  🟦 🟦 🟥       748000000000000029    74800000000000002B
R106  🟦 🟦 🟥       74800000000000002B    74800000000000002F
R147  🟦 🟥 🟦       74800000000000002F74800000000000002F5F698000
R155  🟥 🟦 🟦74800000000000002F5F69800074800000000000002F5F728000
R184  🟦 🟥 🟦74800000000000002F5F72800074800000000000002F5F728000
R110  🟥 🟦 🟦74800000000000002F5F72800074800000000000002F5F728000
R151  🟥 🟦 🟦74800000000000002F5F72800074800000000000002F5F728000
R126  🟥 🟦 🟦74800000000000002F5F72800074800000000000002F5F728000
R142  🟦 🟥 🟦74800000000000002F5F72800074800000000000002F5F728000
  R2  🟦 🟦 🟥74800000000000002F5F728000
```

Demo

Add 2 TiKV



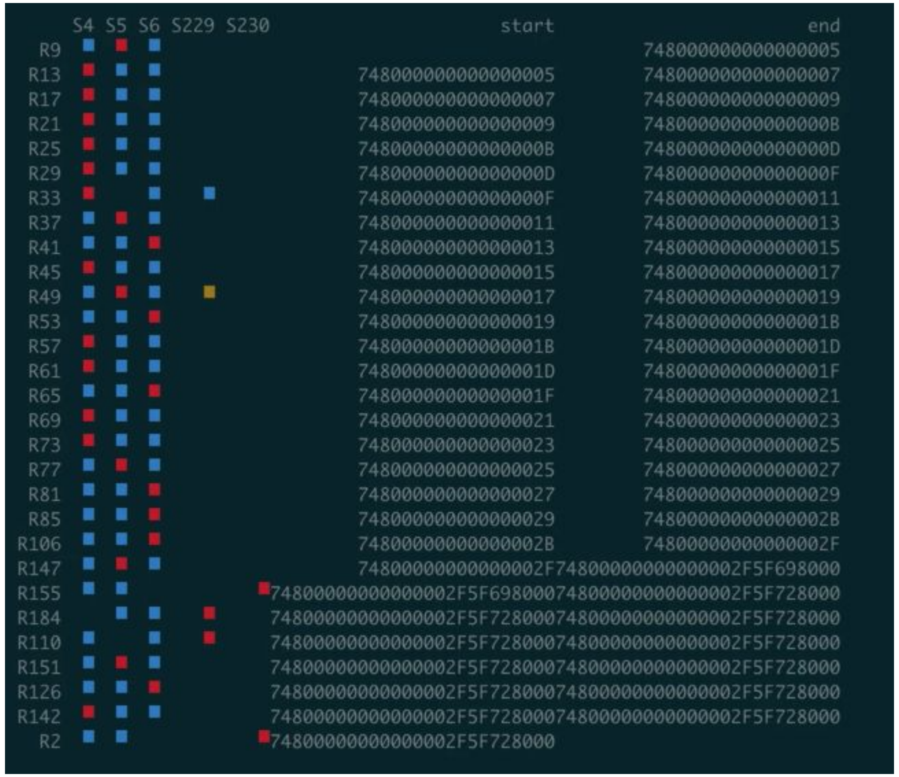
```
       S4 S5 S6 S229 S230                    start                         end
  R9   🟦 🟥 🟦                                               748000000000000005
 R13   🟥 🟦 🟦                  748000000000000005          748000000000000007
 R17   🟥 🟦 🟦                  748000000000000007          748000000000000009
 R21   🟥 🟦 🟦                  748000000000000009          74800000000000000B
 R25   🟥 🟦 🟦                  74800000000000000B          74800000000000000D
 R29   🟥 🟦 🟦                  74800000000000000D          74800000000000000F
 R33   🟥 🟦 🟦                  74800000000000000F          748000000000000011
 R37   🟦 🟥 🟦                  748000000000000011          748000000000000013
 R41   🟦 🟦 🟥                  748000000000000013          748000000000000015
 R45   🟥 🟦 🟦                  748000000000000015          748000000000000017
 R49   🟦 🟦 🟥                  748000000000000017          748000000000000019
 R53   🟦 🟦 🟥                  748000000000000019          74800000000000001B
 R57   🟥 🟦 🟦                  74800000000000001B          74800000000000001D
 R61   🟥 🟦 🟦                  74800000000000001D          74800000000000001F
 R65   🟥 🟦 🟦                  74800000000000001F          748000000000000021
 R69   🟥 🟦 🟦                  748000000000000021          748000000000000023
 R73   🟥 🟦 🟦                  748000000000000023          748000000000000025
 R77   🟦 🟥 🟦                  748000000000000025          748000000000000027
 R81   🟦 🟦 🟥                  748000000000000027          748000000000000029
 R85   🟦 🟦 🟥                  748000000000000029          74800000000000002B
R106   🟦 🟦 🟥                  74800000000000002B          74800000000000002F
R147   🟦 🟥 🟦                  74800000000000002F74800000000000002F5F698000
R155   🟥 🟦 🟦          74800000000000002F5F69800074800000000000002F5F728000
R184   🟦 🟥 🟦          74800000000000002F5F72800074800000000000002F5F728000
R110   🟥 🟦 🟦          74800000000000002F5F72800074800000000000002F5F728000
R151   🟥 🟦 🟦          74800000000000002F5F72800074800000000000002F5F728000
R126   🟥 🟦 🟦          74800000000000002F5F72800074800000000000002F5F728000
R142   🟦 🟥 🟦          74800000000000002F5F72800074800000000000002F5F728000
  R2   🟦 🟦 🟥          74800000000000002F5F728000
```



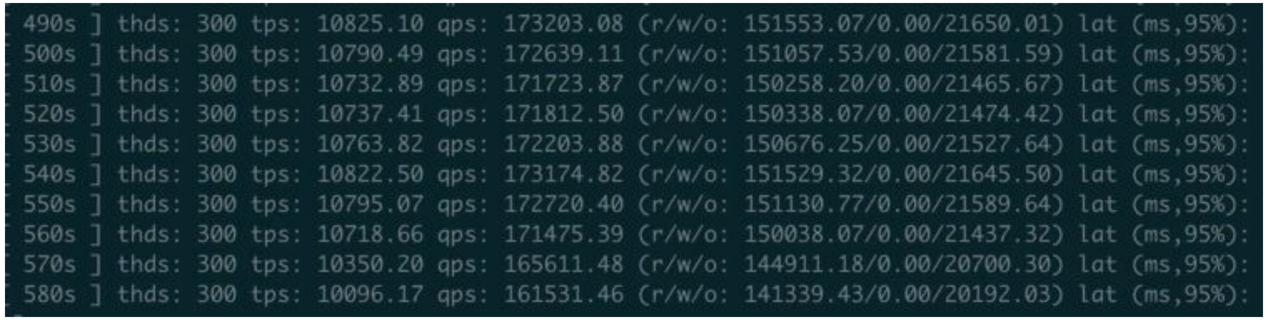
```
Threads started!

[ 10s ] thds: 300 tps: 4690.82 qps: 75259.75 (r/w/o: 65848.13/0.00/9411.62) lat (ms,95%):
[ 20s ] thds: 300 tps: 4842.49 qps: 77481.58 (r/w/o: 67796.60/0.00/9684.97) lat (ms,95%):
[ 30s ] thds: 300 tps: 4691.46 qps: 75070.04 (r/w/o: 65687.12/0.00/9382.92) lat (ms,95%):
[ 40s ] thds: 300 tps: 4753.29 qps: 76050.91 (r/w/o: 66544.32/0.00/9506.59) lat (ms,95%):
[ 50s ] thds: 300 tps: 4677.98 qps: 74830.45 (r/w/o: 65474.48/0.00/9355.97) lat (ms,95%):
[ 60s ] thds: 300 tps: 4606.37 qps: 73706.39 (r/w/o: 64493.64/0.00/9212.75) lat (ms,95%):
[ 70s ] thds: 300 tps: 4646.63 qps: 74370.34 (r/w/o: 65077.07/0.00/9293.27) lat (ms,95%):
```

Demo

Transferring hot regions



	S4	S5	S6	S229	S230	start	end
R9	🟦	🟥	🟦				748000000000000005
R13	🟥	🟦	🟦			748000000000000005	748000000000000007
R17	🟥	🟦	🟦			748000000000000007	748000000000000009
R21	🟥	🟦	🟦			748000000000000009	74800000000000000B
R25	🟥	🟦	🟦			74800000000000000B	74800000000000000D
R29	🟥	🟦	🟦			74800000000000000D	74800000000000000F
R33	🟥		🟦	🟦		74800000000000000F	748000000000000011
R37	🟦	🟥	🟦			748000000000000011	748000000000000013
R41	🟦	🟦	🟥			748000000000000013	748000000000000015
R45	🟥	🟦	🟦			748000000000000015	748000000000000017
R49	🟦	🟥	🟦	🟨		748000000000000017	748000000000000019
R53	🟦	🟦	🟥			748000000000000019	74800000000000001B
R57	🟥	🟦	🟦			74800000000000001B	74800000000000001D
R61	🟥	🟦	🟦			74800000000000001D	74800000000000001F
R65	🟦	🟦	🟥			74800000000000001F	748000000000000021
R69	🟥	🟦	🟦			748000000000000021	748000000000000023
R73	🟥	🟦	🟦			748000000000000023	748000000000000025
R77	🟦	🟥	🟦			748000000000000025	748000000000000027
R81	🟦	🟦	🟥			748000000000000027	748000000000000029
R85	🟦	🟦	🟥			748000000000000029	74800000000000002B
R106	🟦	🟦	🟥			74800000000000002B	74800000000000002F
R147	🟦	🟥	🟦			74800000000000002F	74800000000000002F5F698000
R155	🟦	🟦			🟥	74800000000000002F5F698000	74800000000000002F5F728000
R184		🟦	🟦	🟥		74800000000000002F5F728000	74800000000000002F5F728000
R110	🟦		🟦	🟥		74800000000000002F5F728000	74800000000000002F5F728000
R151	🟦	🟥	🟦			74800000000000002F5F728000	74800000000000002F5F728000
R126	🟦	🟦	🟥			74800000000000002F5F728000	74800000000000002F5F728000
R142	🟥	🟦	🟦			74800000000000002F5F728000	74800000000000002F5F728000
R2	🟦	🟦			🟥	74800000000000002F5F728000	

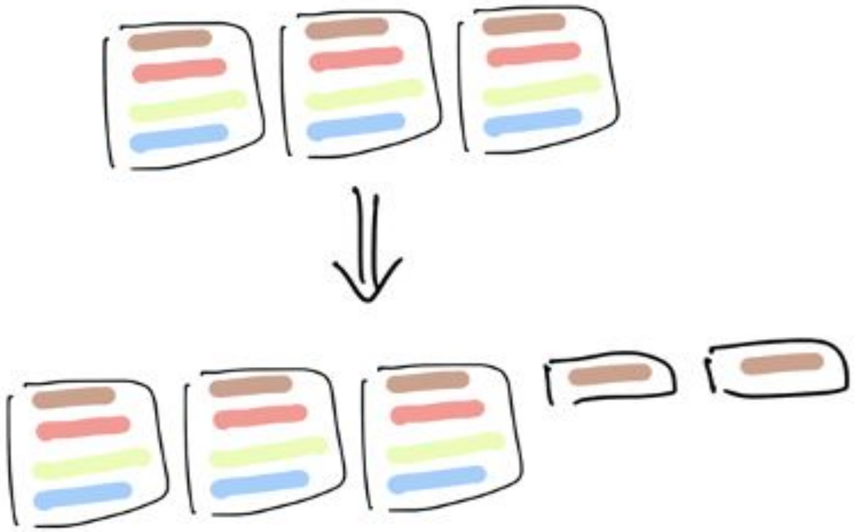


```
[ 490s ] thds: 300 tps: 10825.10 qps: 173203.08 (r/w/o: 151553.07/0.00/21650.01) lat (ms,95%):
[ 500s ] thds: 300 tps: 10790.49 qps: 172639.11 (r/w/o: 151057.53/0.00/21581.59) lat (ms,95%):
[ 510s ] thds: 300 tps: 10732.89 qps: 171723.87 (r/w/o: 150258.20/0.00/21465.67) lat (ms,95%):
[ 520s ] thds: 300 tps: 10737.41 qps: 171812.50 (r/w/o: 150338.07/0.00/21474.42) lat (ms,95%):
[ 530s ] thds: 300 tps: 10763.82 qps: 172203.88 (r/w/o: 150676.25/0.00/21527.64) lat (ms,95%):
[ 540s ] thds: 300 tps: 10822.50 qps: 173174.82 (r/w/o: 151529.32/0.00/21645.50) lat (ms,95%):
[ 550s ] thds: 300 tps: 10795.07 qps: 172720.40 (r/w/o: 151130.77/0.00/21589.64) lat (ms,95%):
[ 560s ] thds: 300 tps: 10718.66 qps: 171475.39 (r/w/o: 150038.07/0.00/21437.32) lat (ms,95%):
[ 570s ] thds: 300 tps: 10350.20 qps: 165611.48 (r/w/o: 144911.18/0.00/20700.30) lat (ms,95%):
[ 580s ] thds: 300 tps: 10096.17 qps: 161531.46 (r/w/o: 141339.43/0.00/20192.03) lat (ms,95%):
```

Future work

Future work

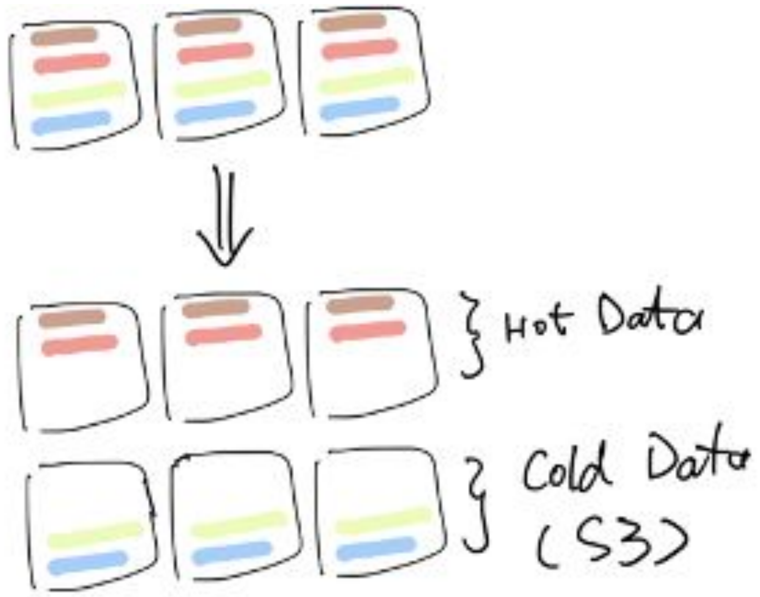
Future work



- **Replication by workloads**

Changing the replication for some regions according to different workloads.

Future work



- **Separate hot and cold data**

Using cheaper storage media to store the cold data.

Join us

- GitHub: https://github.com/tikv/tikv
- Website: https://tikv.org/
- Twitter: @tikvproject
- Slack: #sig-scheduling in Slack

KEEP CLOUD NATIVE

KubeCon | CloudNativeCon

North America 2020

Virtual

EVERYWHERE