AWS re:Invent

DAT325

Amazon DynamoDB: Under the hood of a hyperscale database

James Christopher Sorenson III aka jaso

Senior Principal Engineer

Amazon Web Services

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

Areas for discussion

Overview GetItem / PutItem

Partitioning

Transactions

Global Tables

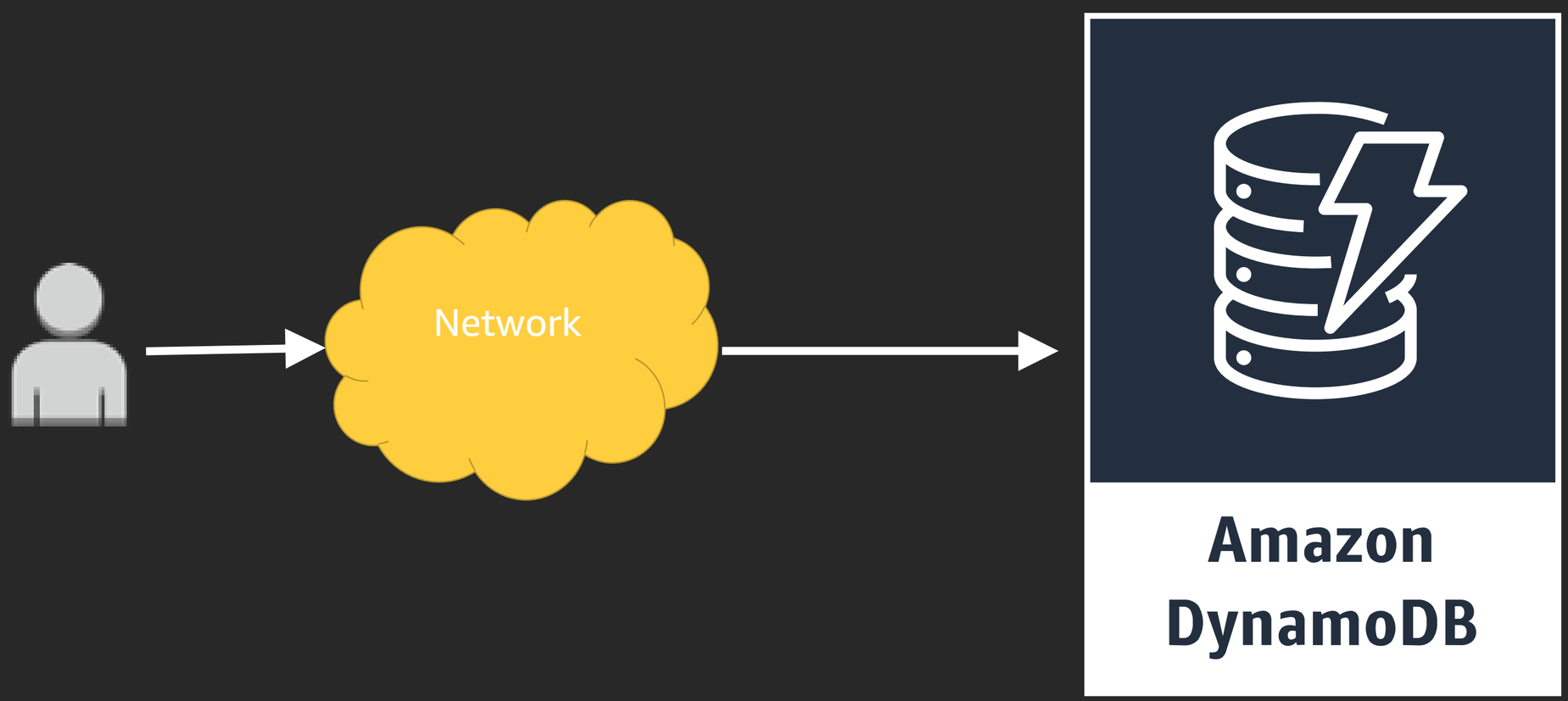
Autoscaling

Backup Restore

GetItem / PutItem

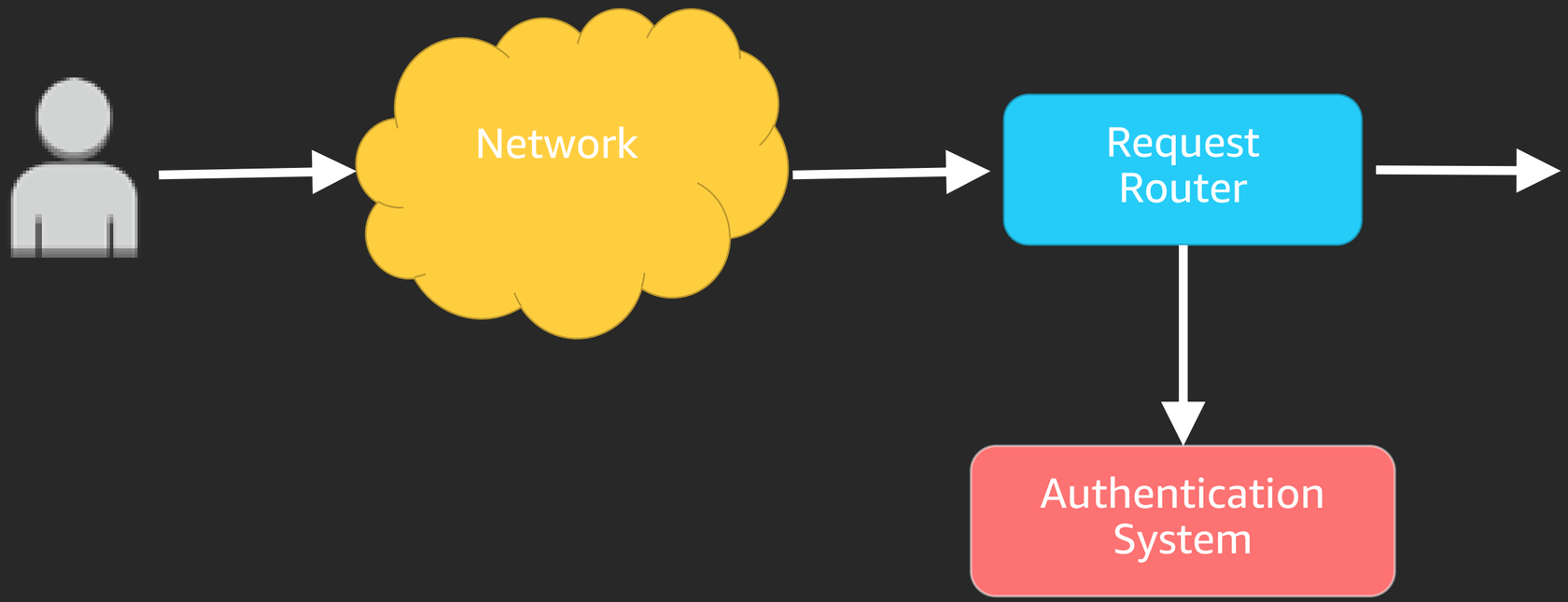
© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

GetItem



Amazon DynamoDB

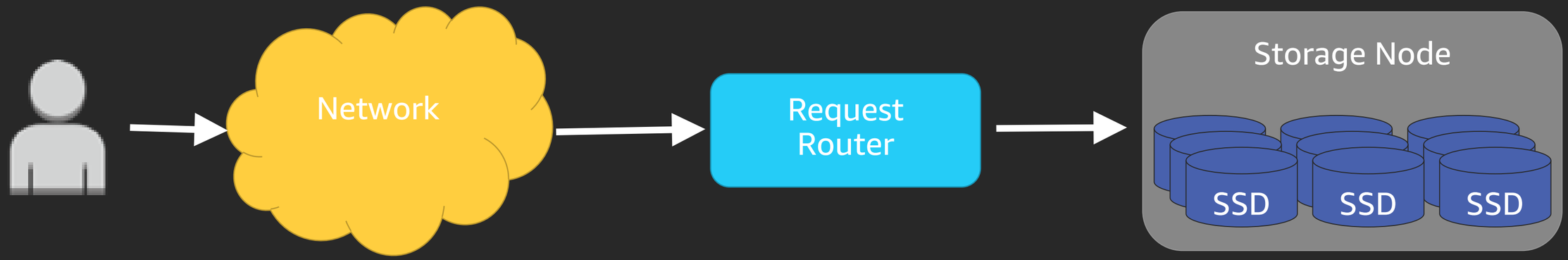
GetItem (Step 1)



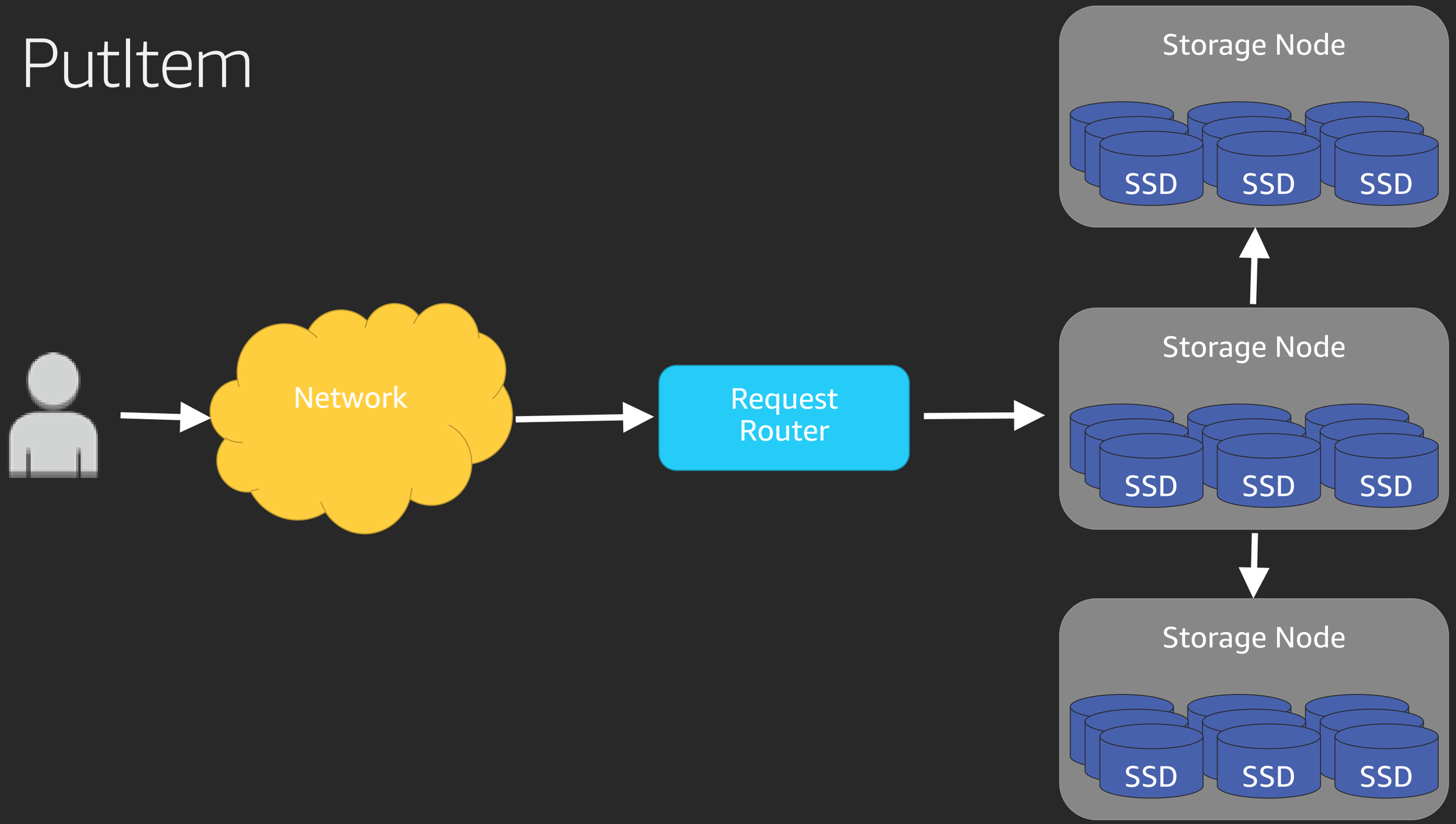
Sample Policy

```
{
 "Version": "2012-10-17",
 "Statement": [
  {
   "Effect": "Allow",
   "Action": [
    "dynamodb:GetItem"
   ],
   "Resource": [
    "arn:aws:dynamodb:us-west-2:1234567890:table/customer"
   ],
   "Condition": {
    "ForAllValues:StringEquals": {
     "dynamodb:LeadingKeys": [
      "${www.amazon.com:user_id}"
     ]
    }
   }
  }
 ]
}
```

GetItem (Step 2)



PutItem


Storage Node
SSD
SSD
SSD
Storage Node
SSD
SSD
SSD
Storage Node
SSD
SSD
SSD
Network
Request
Router

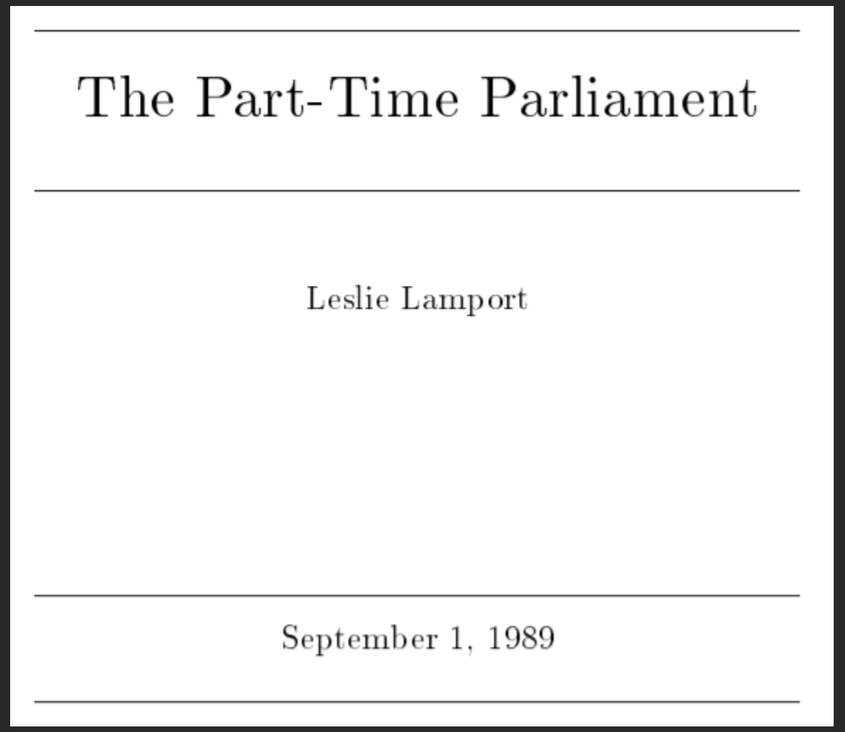
DynamoDB evolved from Dynamo

Dynamo: Amazon's Highly Available Key-value Store

Giuseppe DeCandia, Deniz Hastorun, Madan Jampani, Gunavardhan Kakulapati,
Avinash Lakshman, Alex Pilchin, Swaminathan Sivasubramanian, Peter Vosshall
and Werner Vogels

Amazon.com

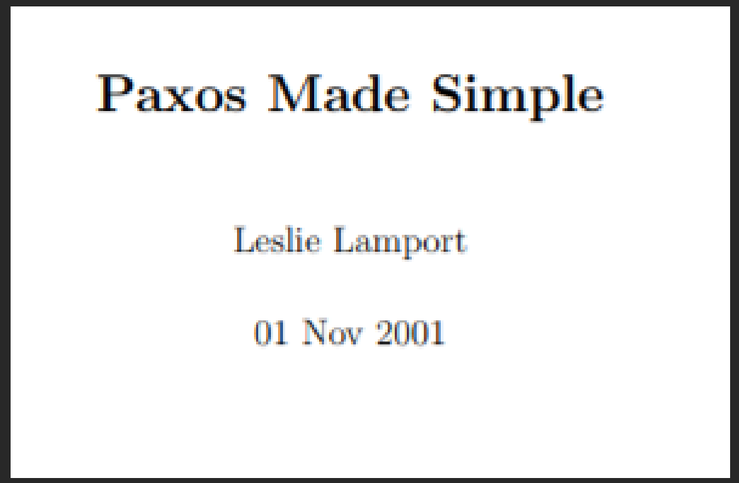
Paxos



The Part-Time Parliament

Leslie Lamport

September 1, 1989

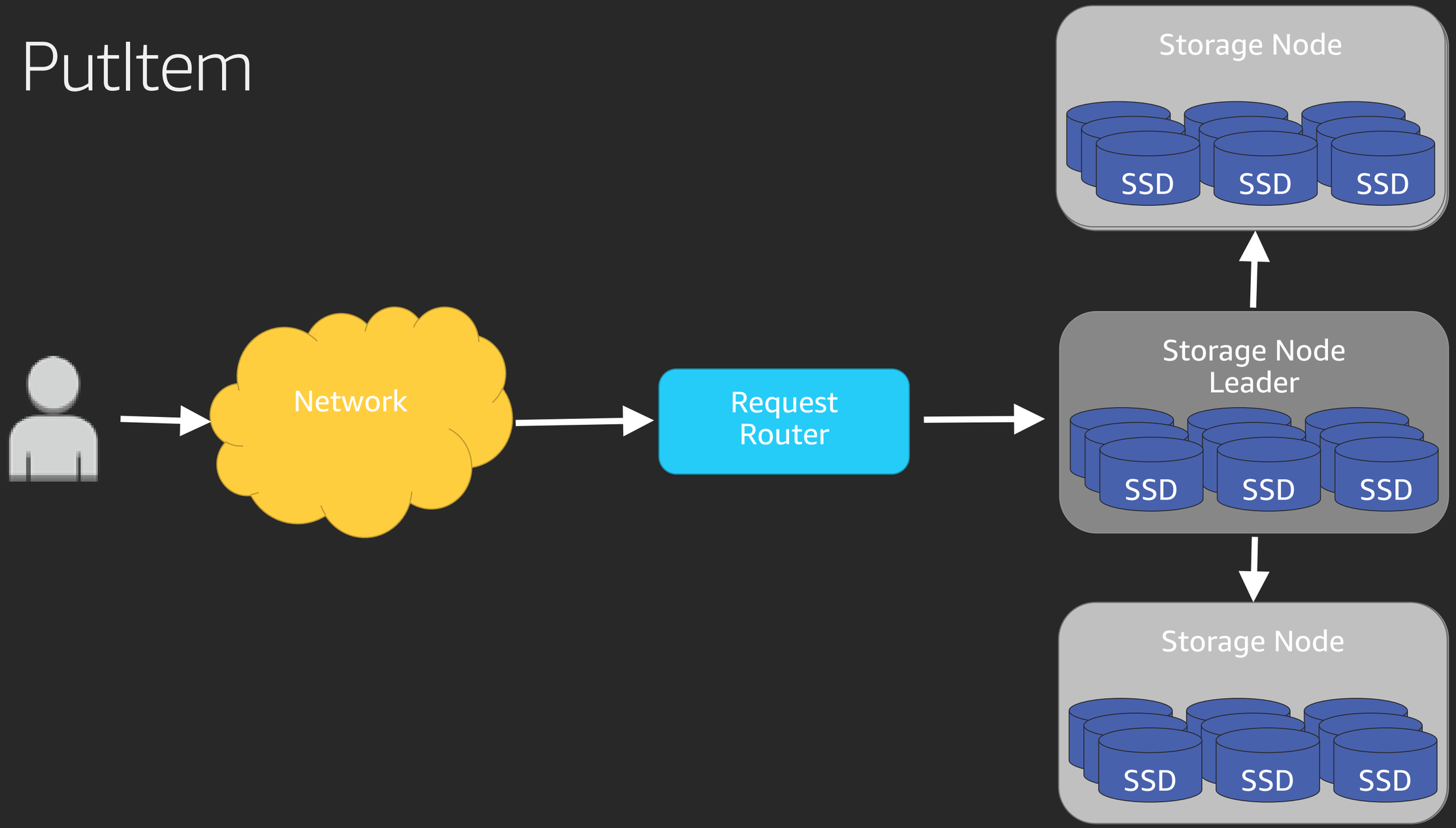


Paxos Made Simple

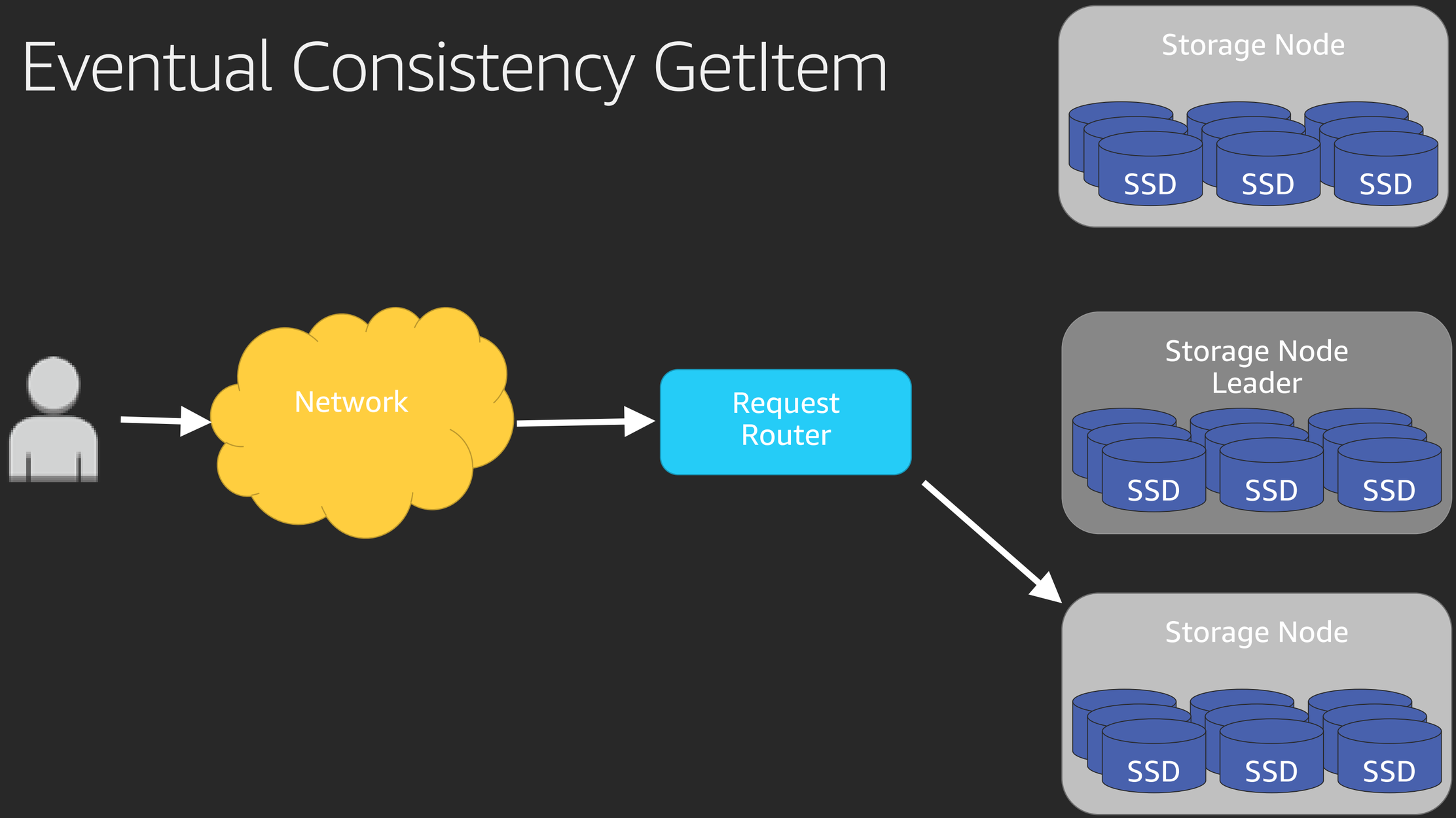
Leslie Lamport

01 Nov 2001

PutItem


Storage Node
SSD
SSD
SSD
Network
Request Router
Storage Node Leader
SSD
SSD
SSD
Storage Node
SSD
SSD
SSD

Eventual Consistency GetItem



Questions

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Partitioning

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Tables

	CustID	Customer Information
	145783	{ name:“Bob”, city:”London”, …}
	236294	{ name:“Sara”, city:”Tampa”, …}
	333363	{ name:“Betty”, city:”Madison”, …}
	445104	{ name:“James”, city:”Miami”, …}
	523422	{ name:“Alex”, city:”London”, …}
	643145	{ name:“Val”, city:”Seattle”, …}
	723342	{ name:“Jeff”, city:”Toledo”, …}

Hashing

Hash Value	CustID	Customer Information
0x9531	145783	{ name:“Bob”, city:”London”, …}
0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}
0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}

Partitioning

Hash Value	CustID	Customer Information
0x9531	145783	{ name:“Bob”, city:”London”, …}
0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}
0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

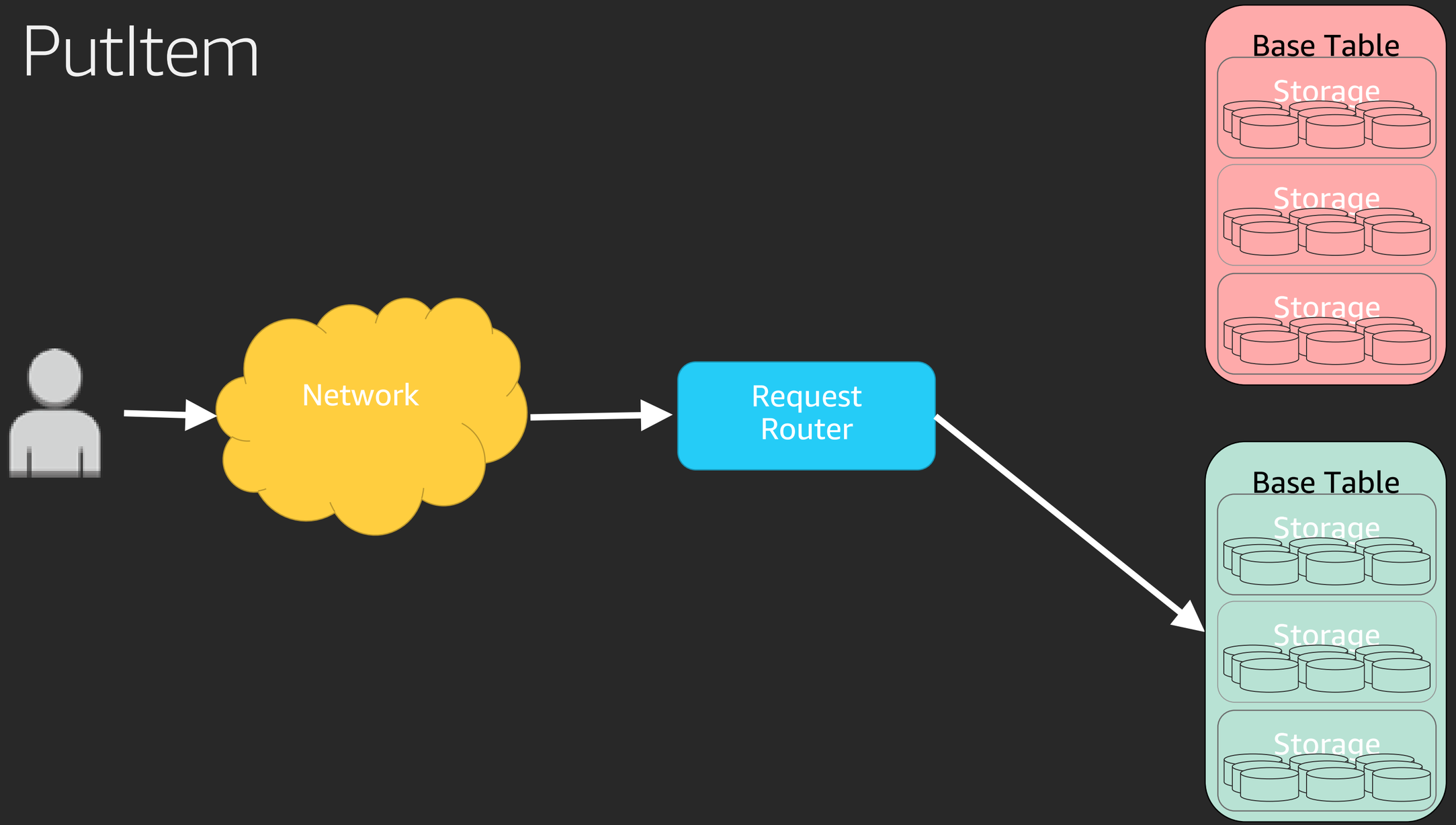
0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

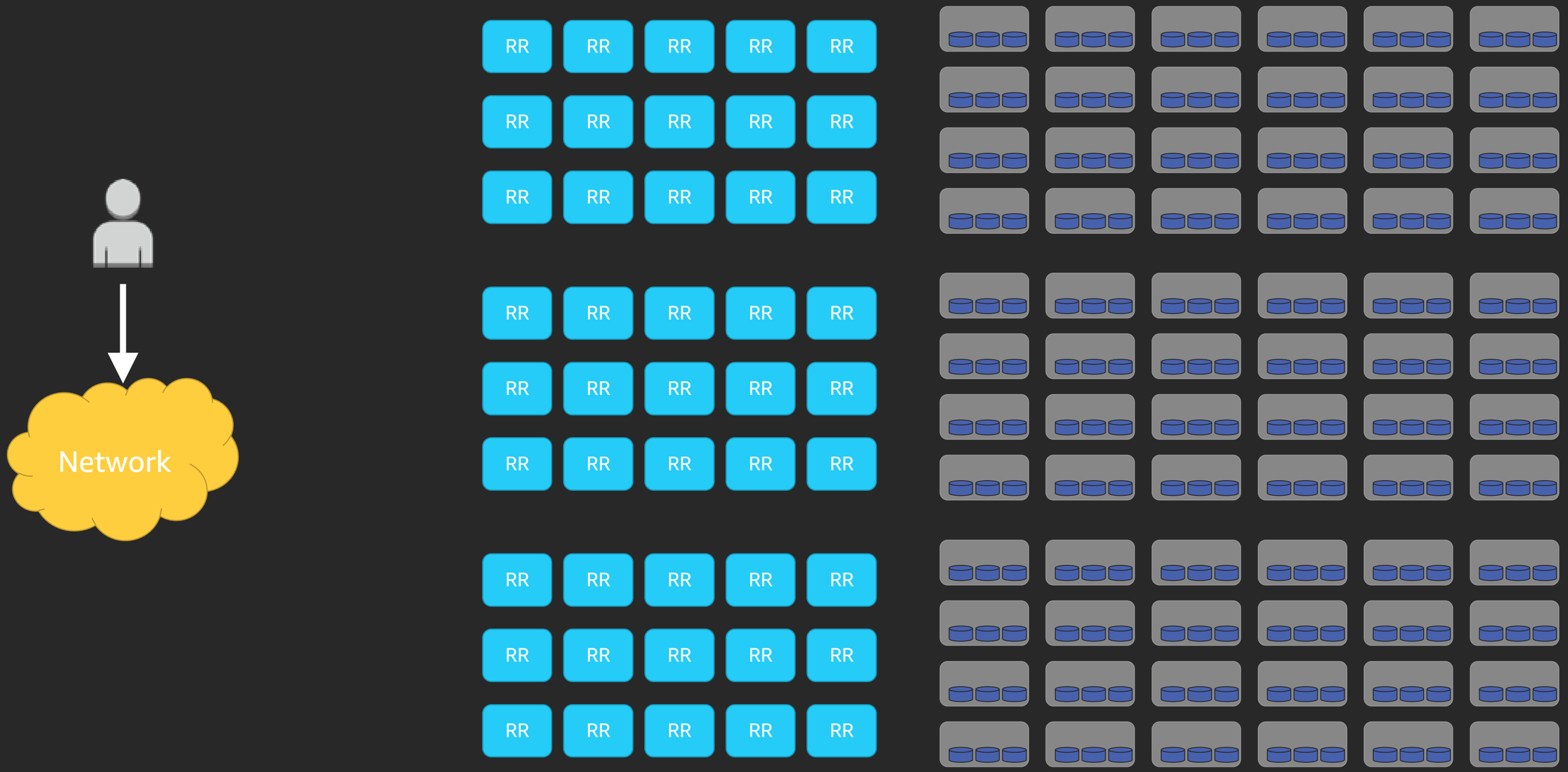
0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

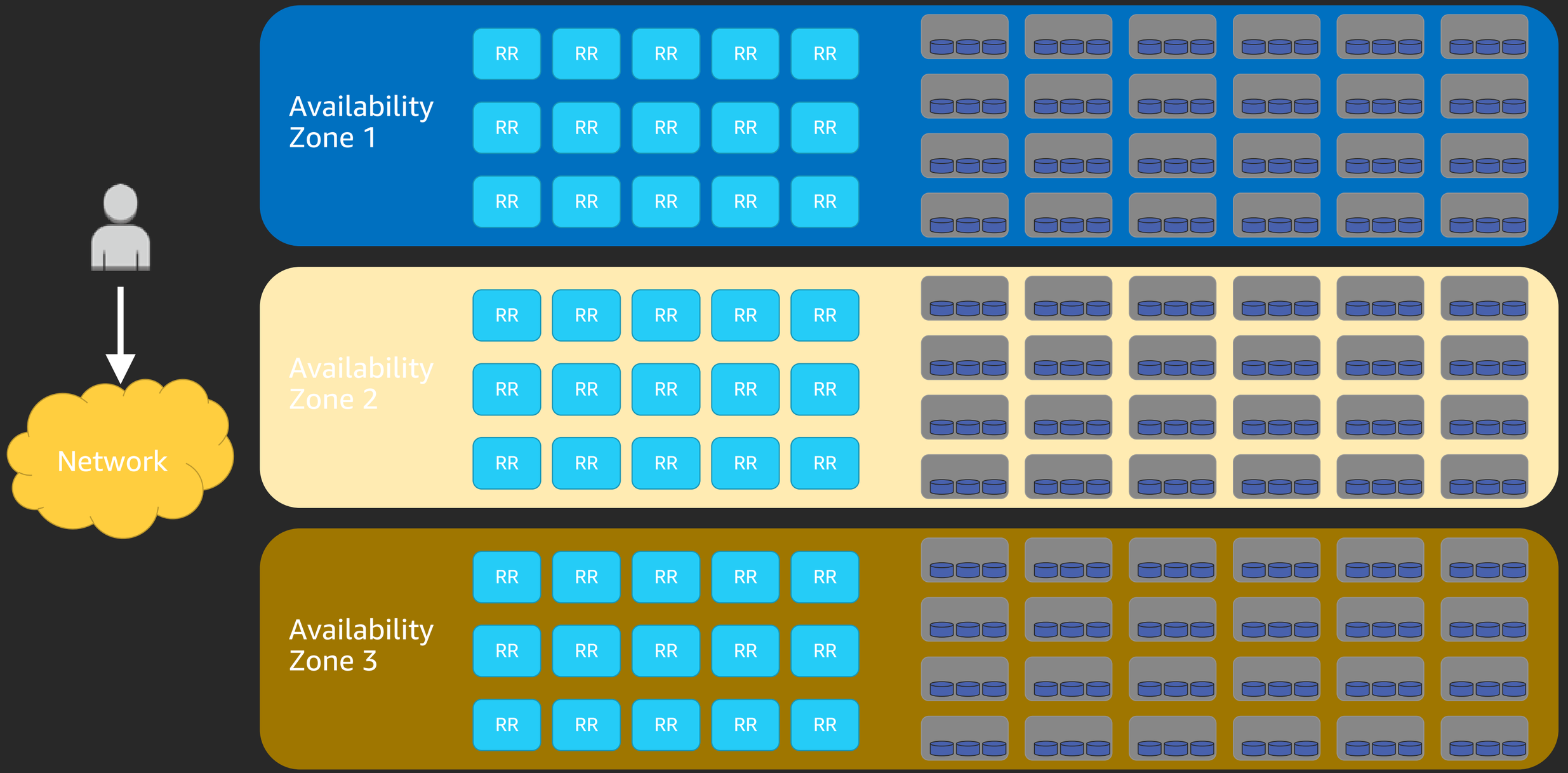
0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

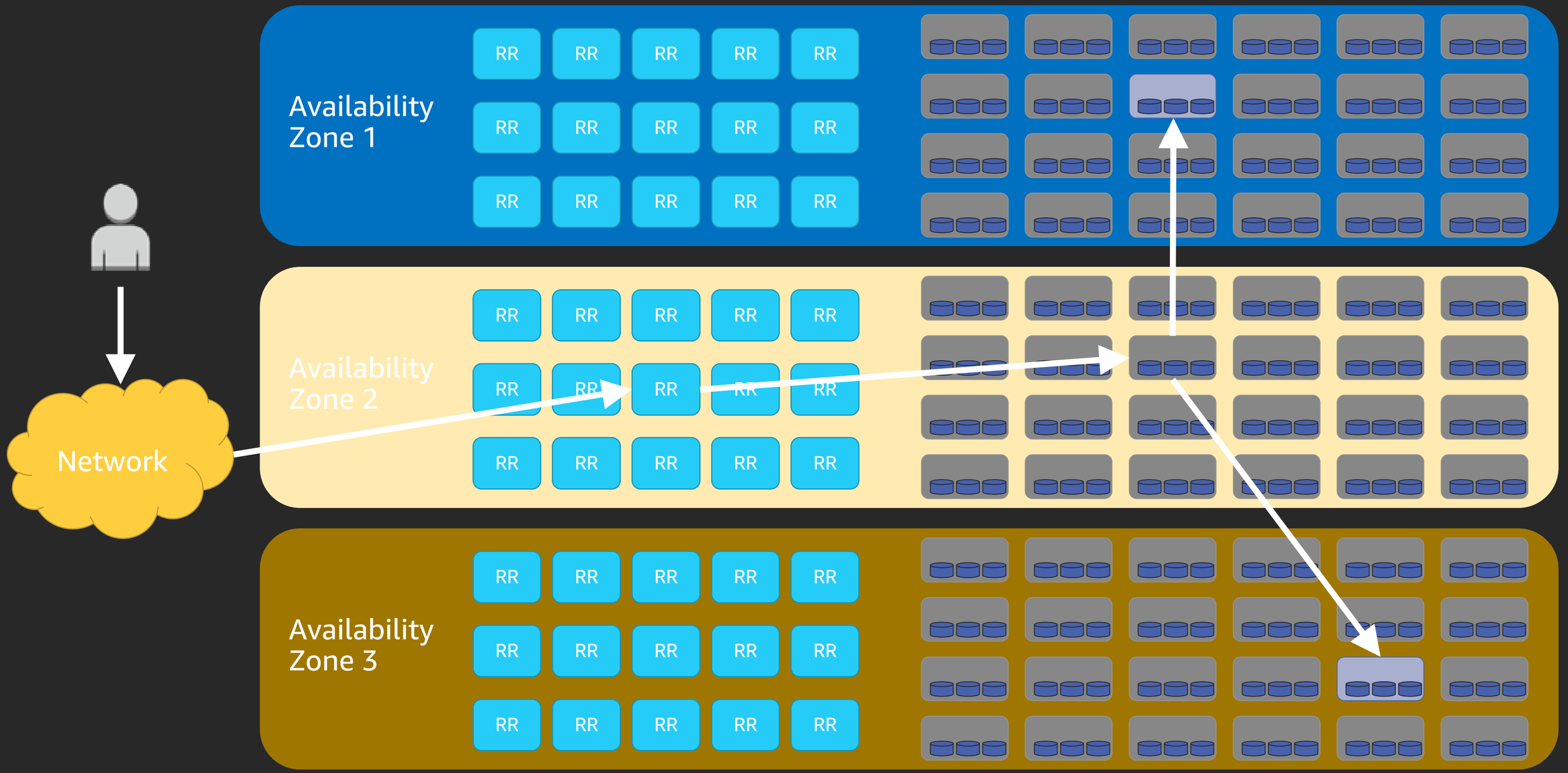
0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

PutItem


Base Table
Storage
Storage
Storage
Network
Request
Router
Base Table
Storage
Storage
Storage


RR
Network


Availability Zone 1
RR
Availability Zone 2
RR
Availability Zone 3
RR
Network

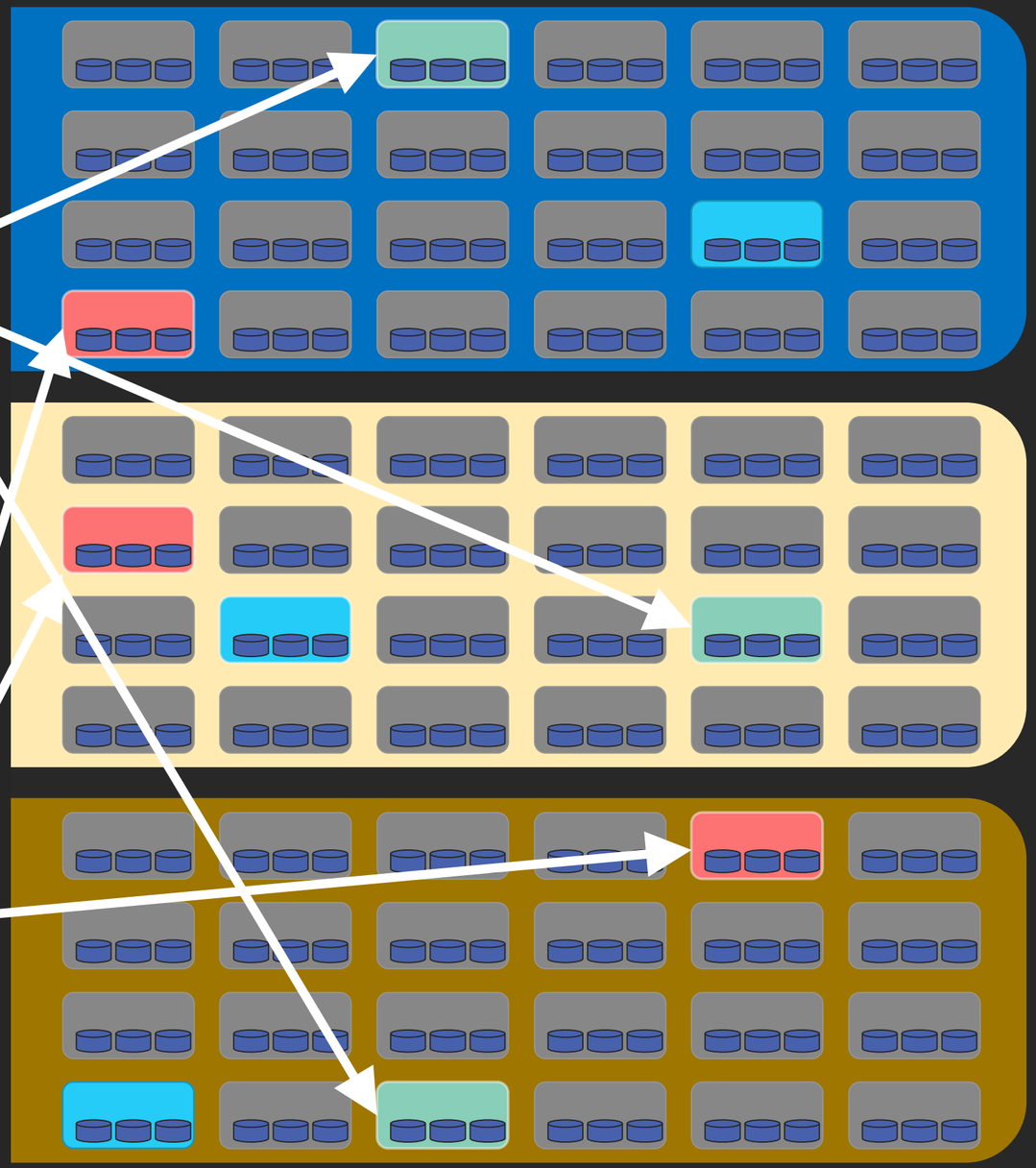

Availability Zone 1
RR
Availability Zone 2
Availability Zone 3
Network

PutItem

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}



Questions

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

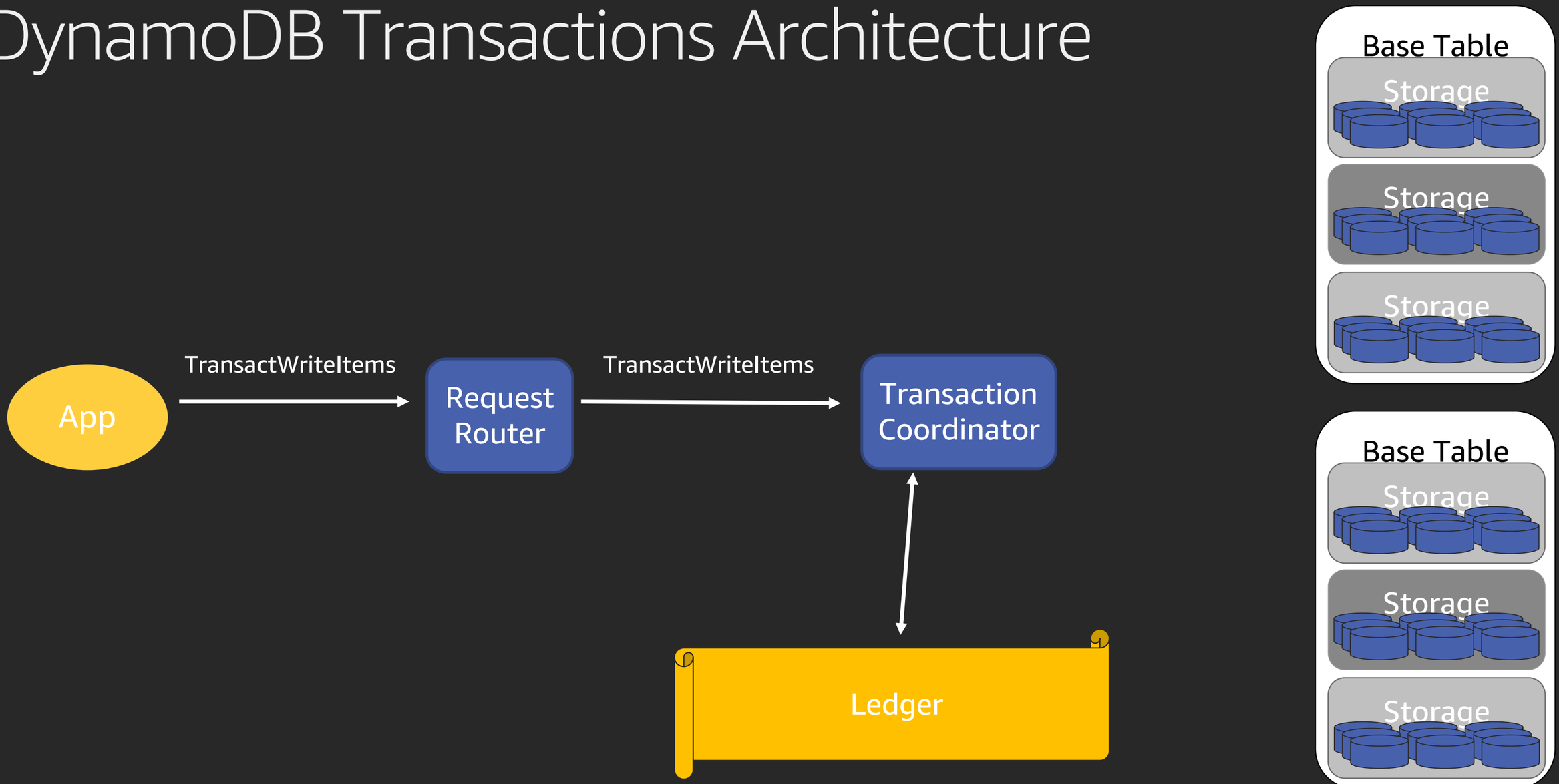
Transactions

AWS re:Invent

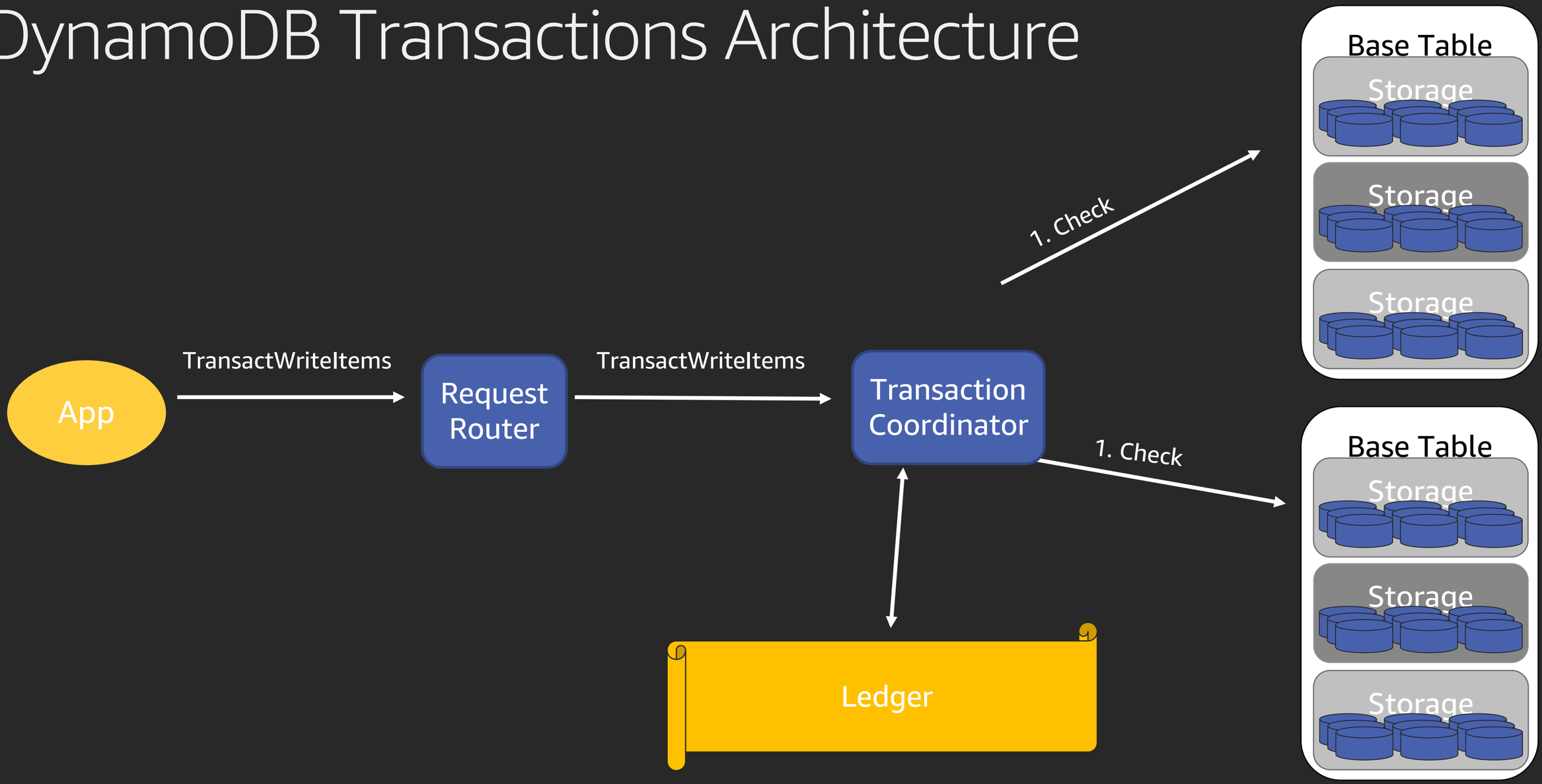
© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

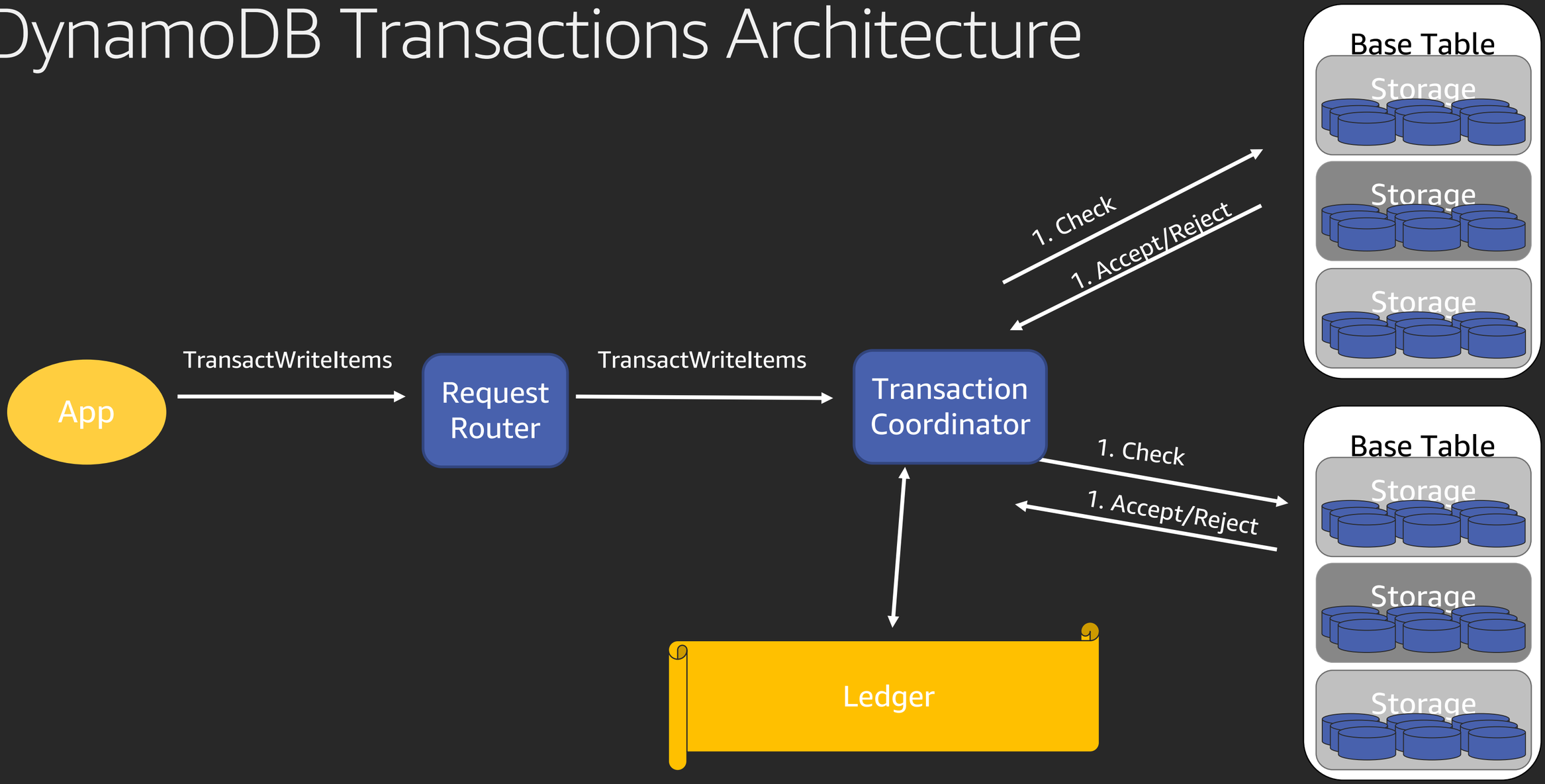
DynamoDB Transactions Architecture



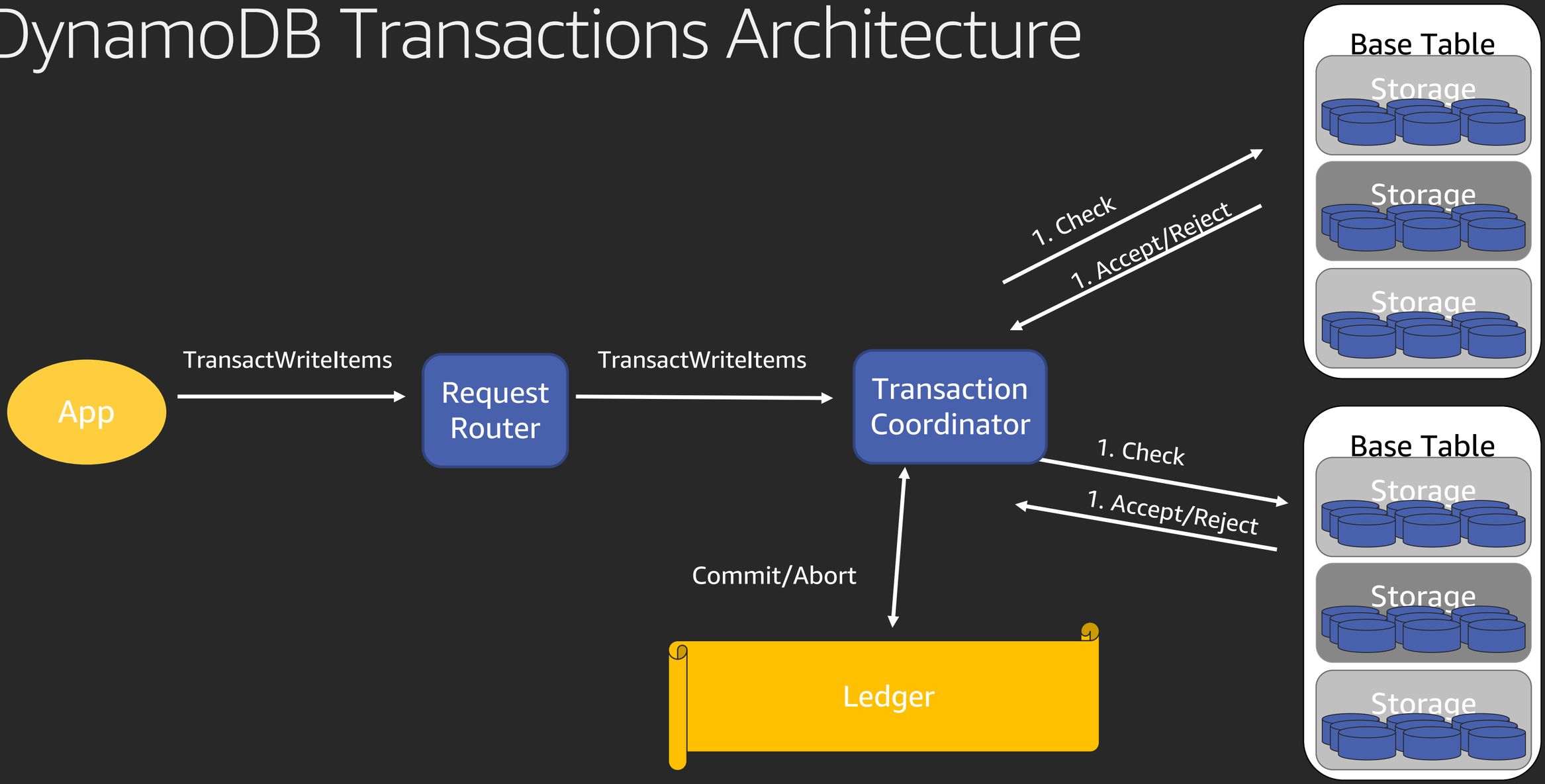
DynamoDB Transactions Architecture



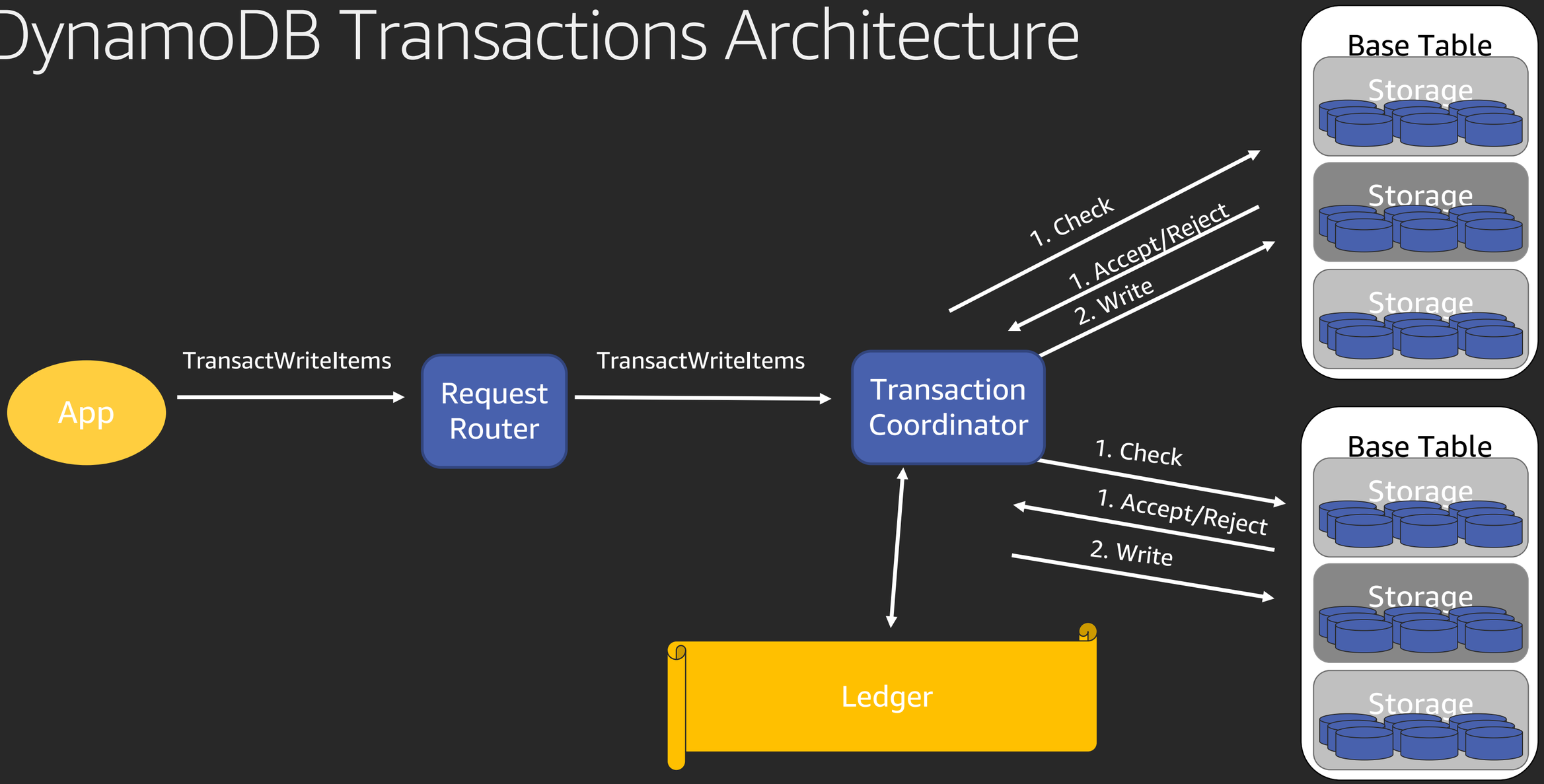
DynamoDB Transactions Architecture



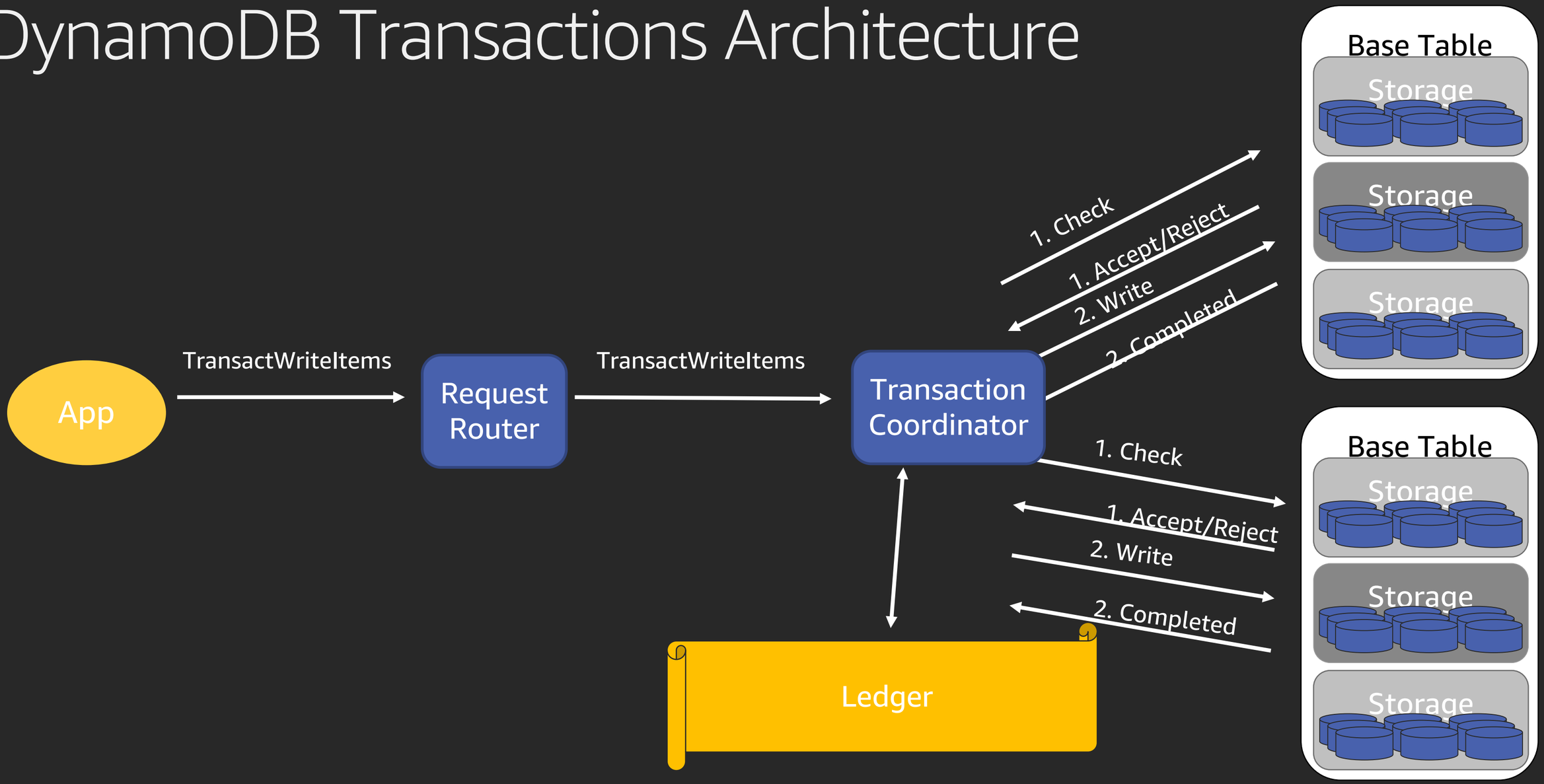
DynamoDB Transactions Architecture



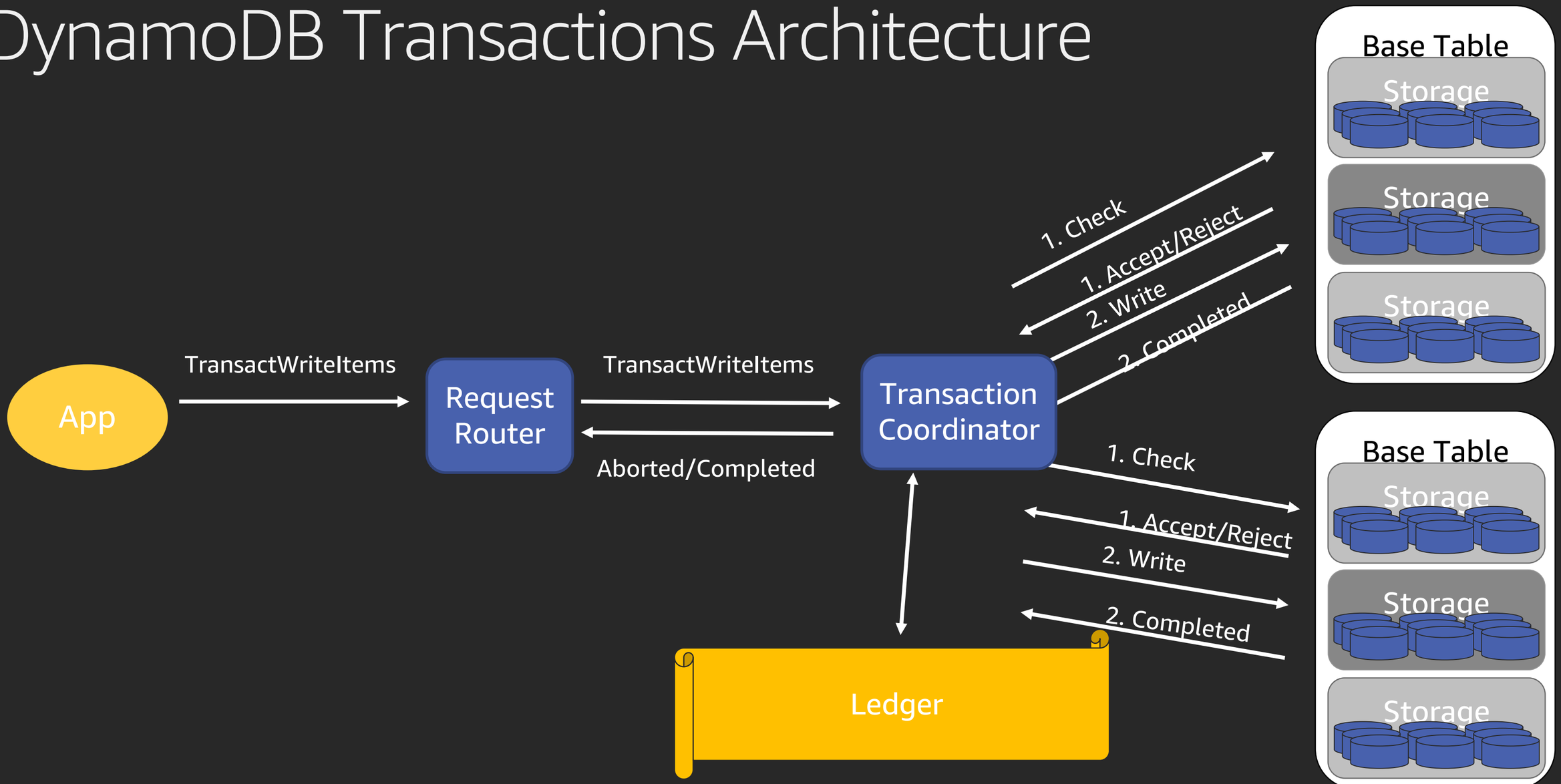
DynamoDB Transactions Architecture



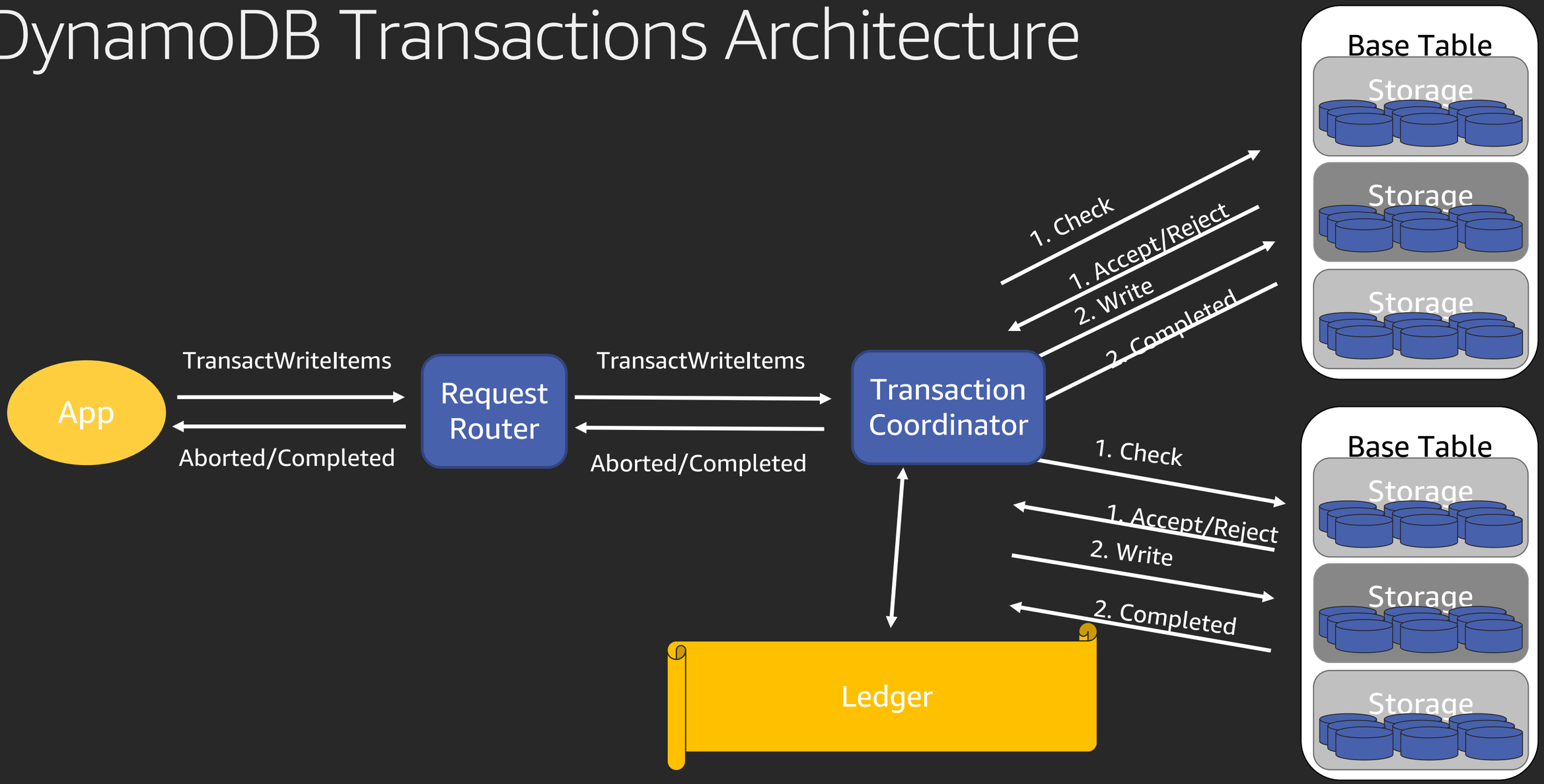
DynamoDB Transactions Architecture



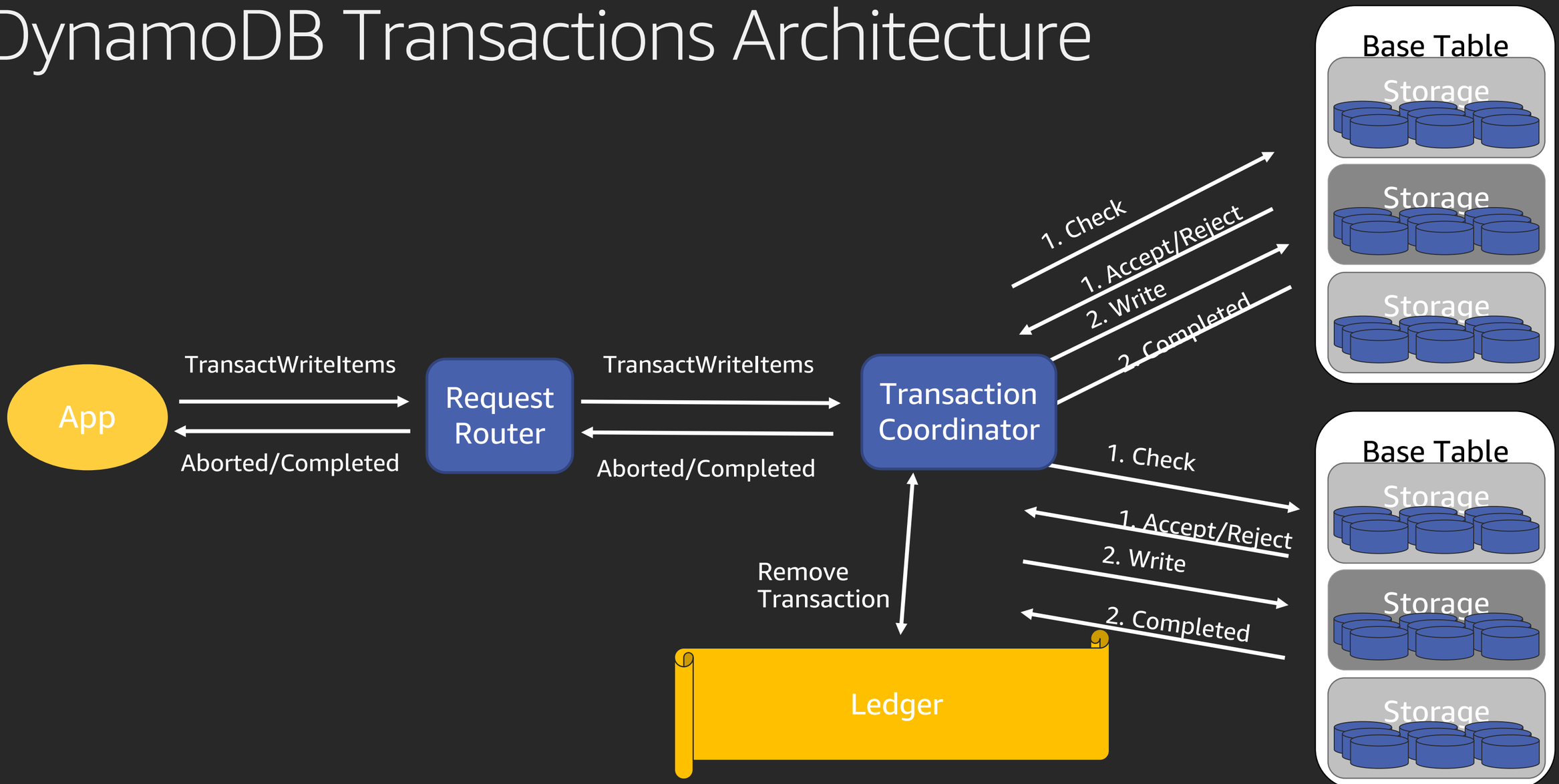
DynamoDB Transactions Architecture



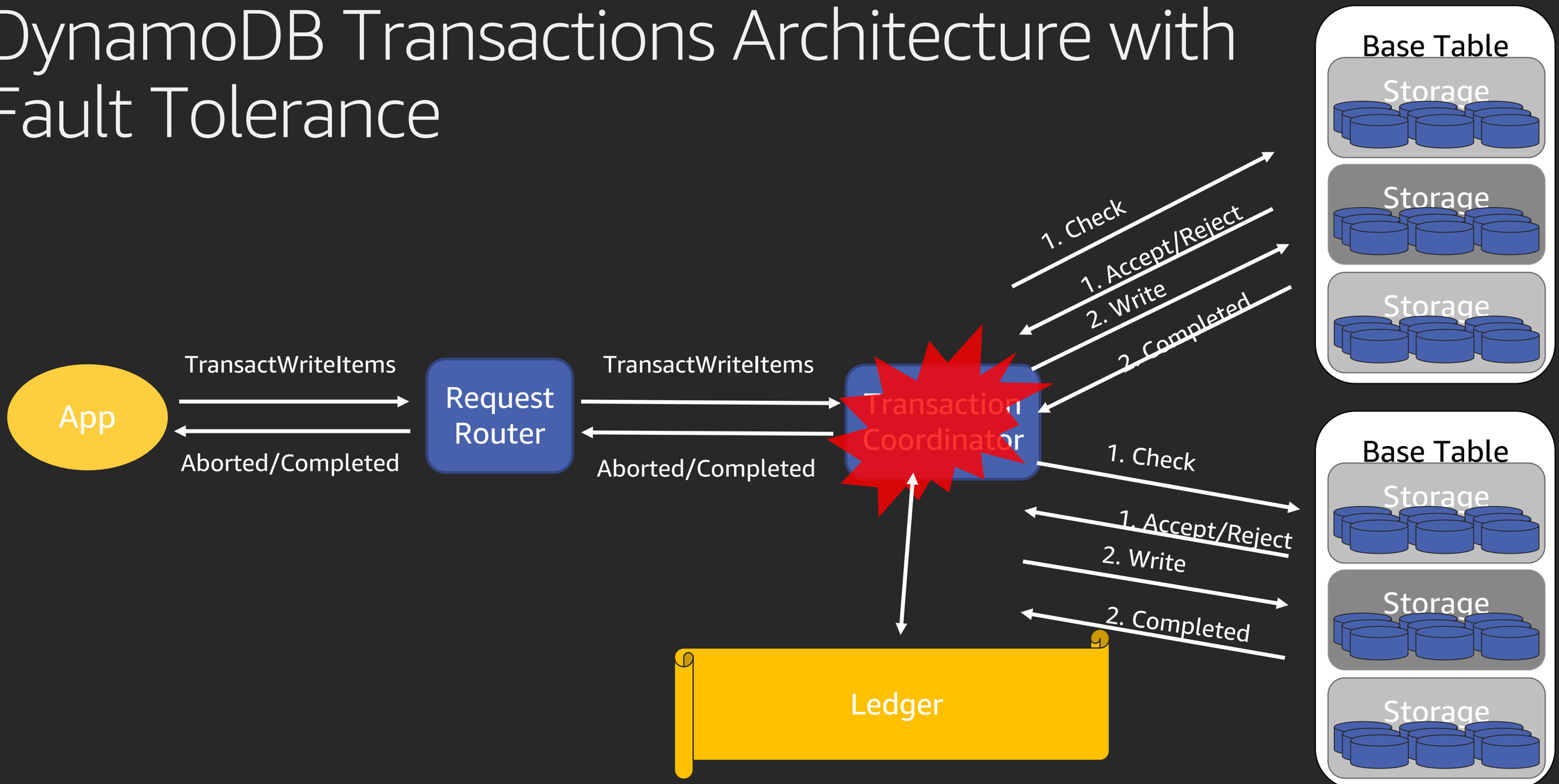
DynamoDB Transactions Architecture



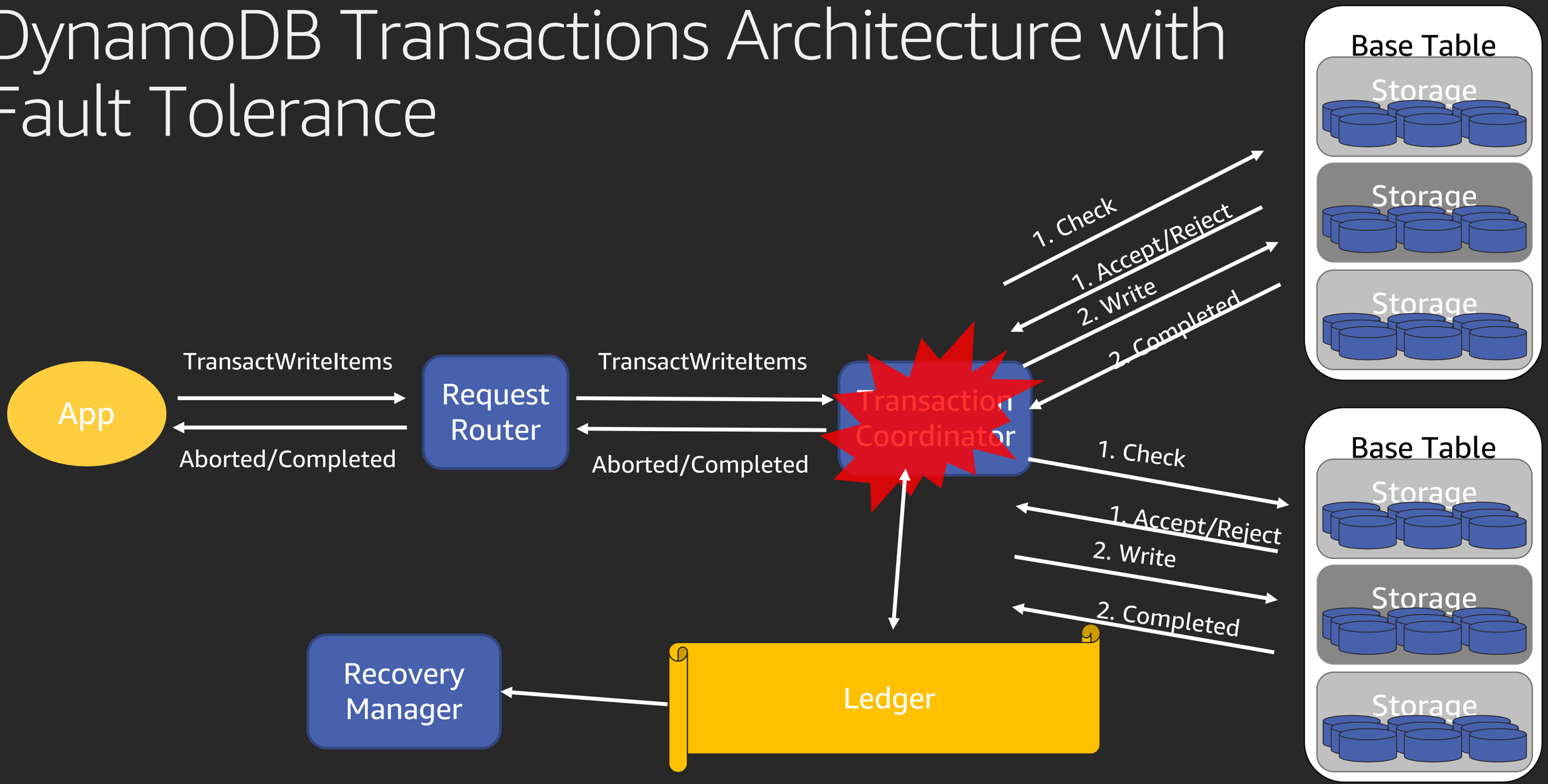
DynamoDB Transactions Architecture



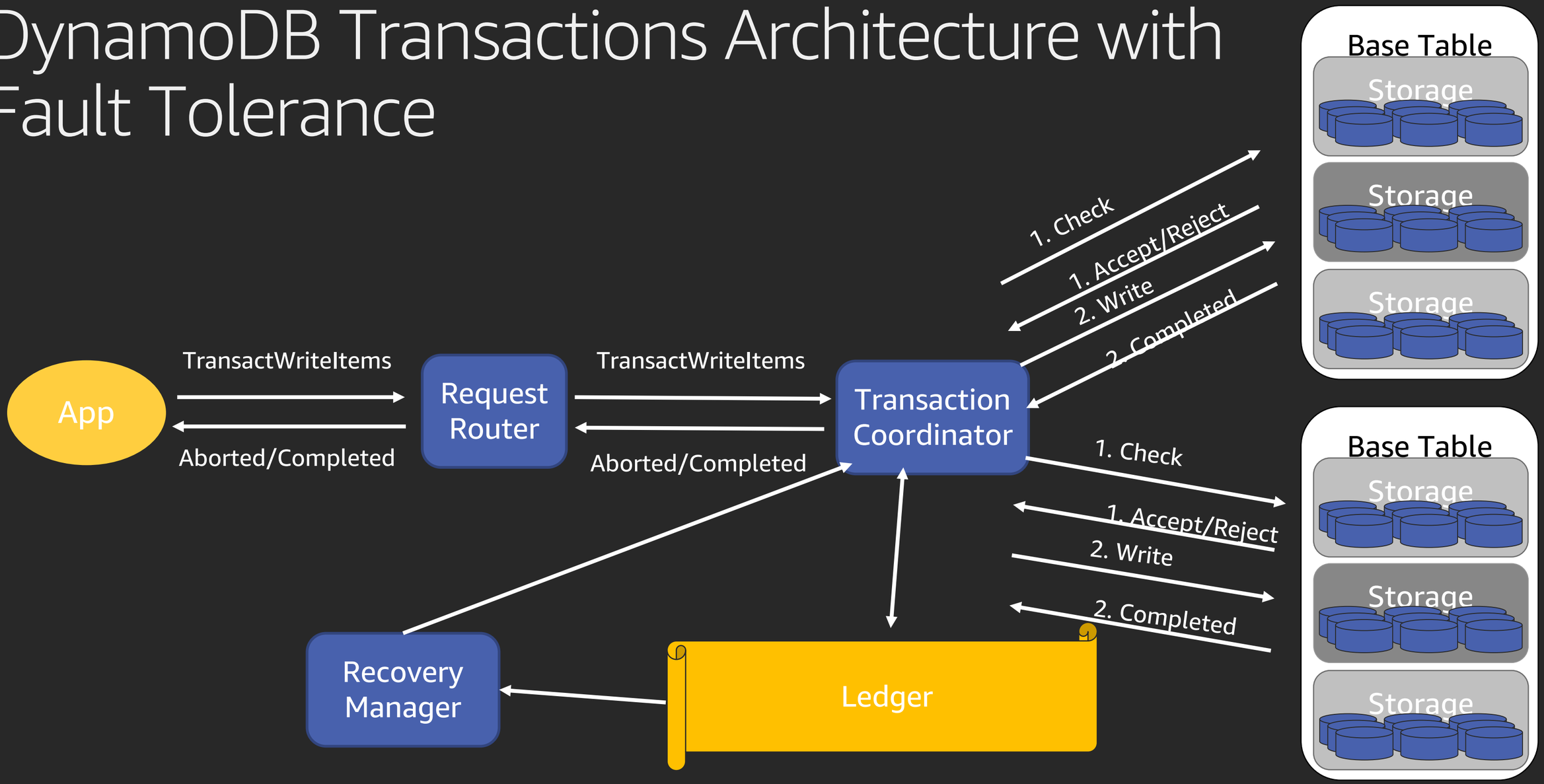
DynamoDB Transactions Architecture with Fault Tolerance



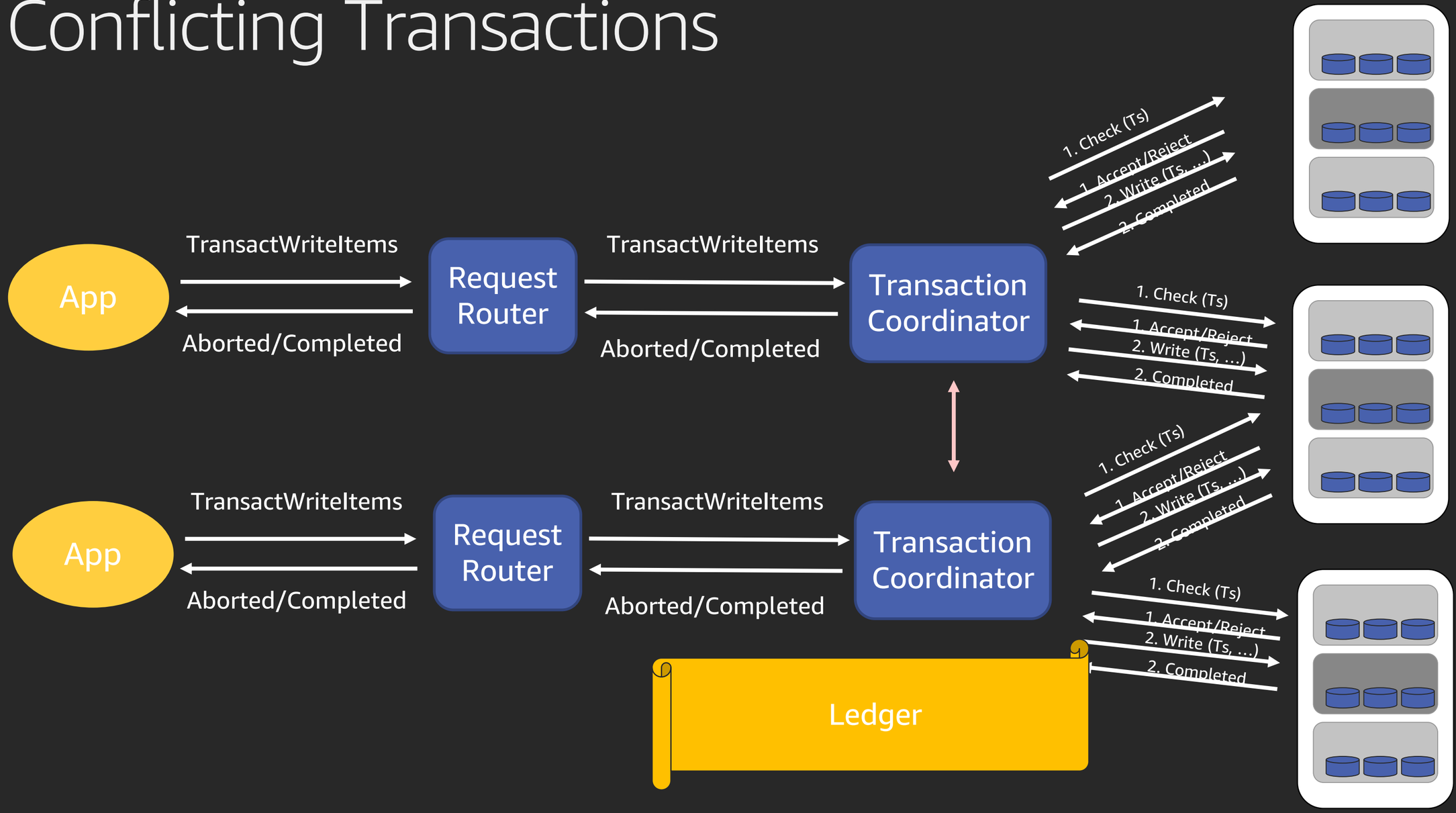
DynamoDB Transactions Architecture with Fault Tolerance



DynamoDB Transactions Architecture with Fault Tolerance



Conflicting Transactions



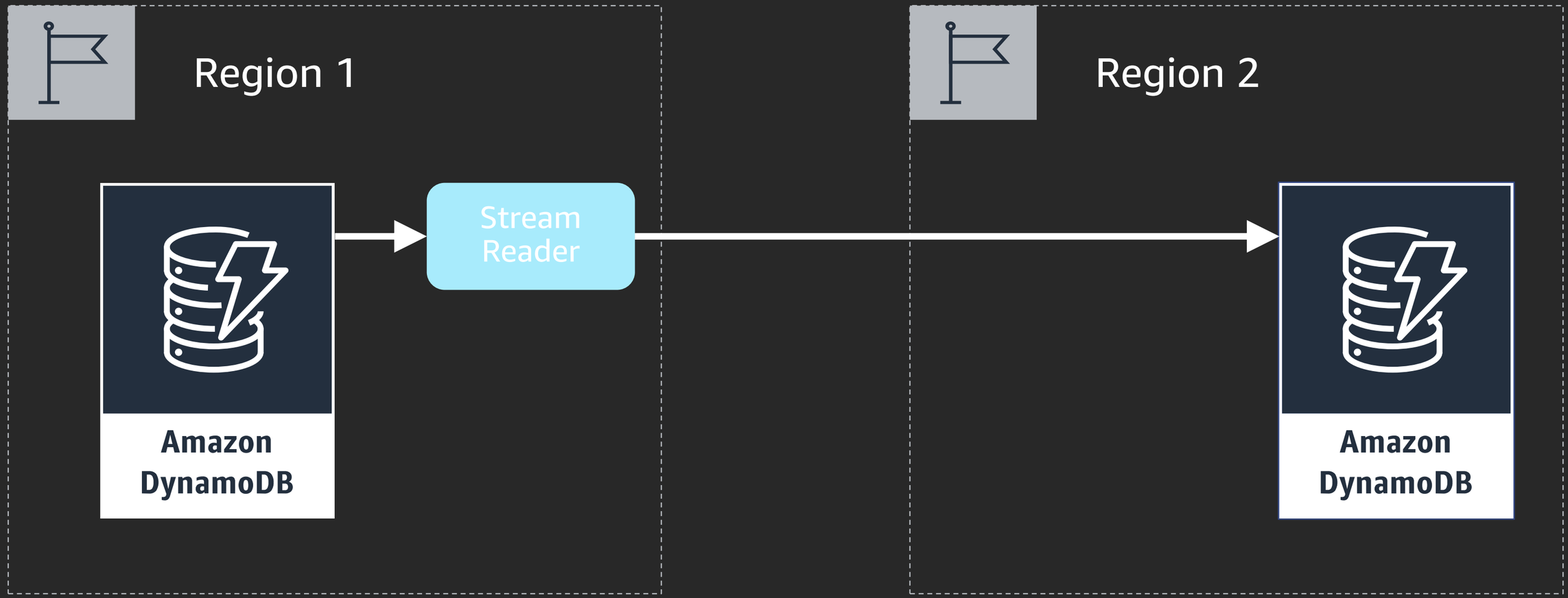
Questions

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

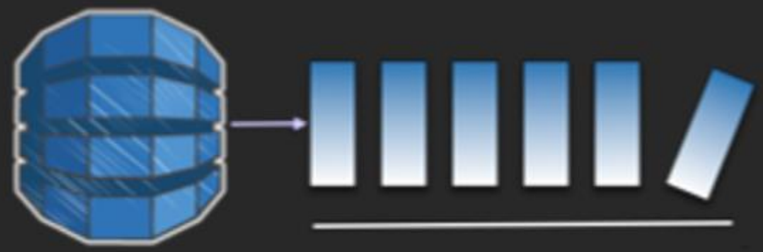
Global Tables

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

High-Level Architecture



DynamoDB Streams



- All table mutations (Put, Update, Delete)
- No duplicates
- In Order (By Key)
- New and Old Item Image Available

DynamoDB Streams

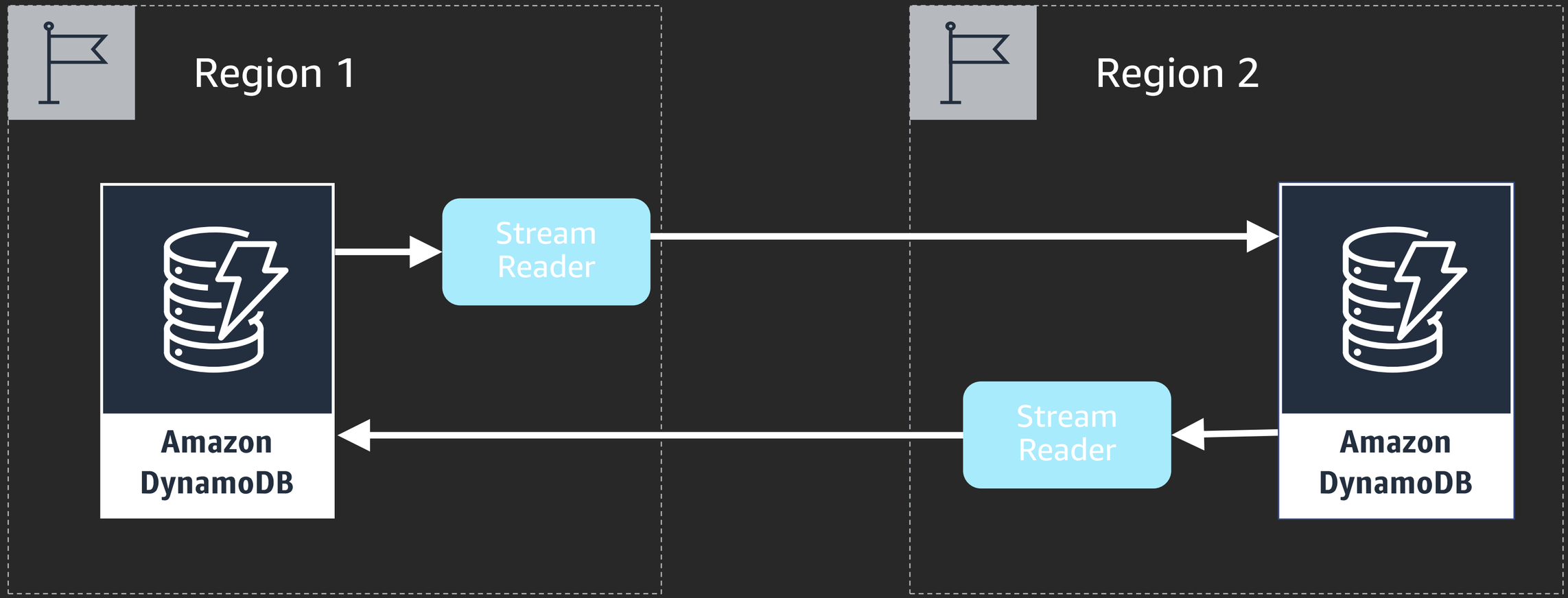
Shards

Kinesis Client Library

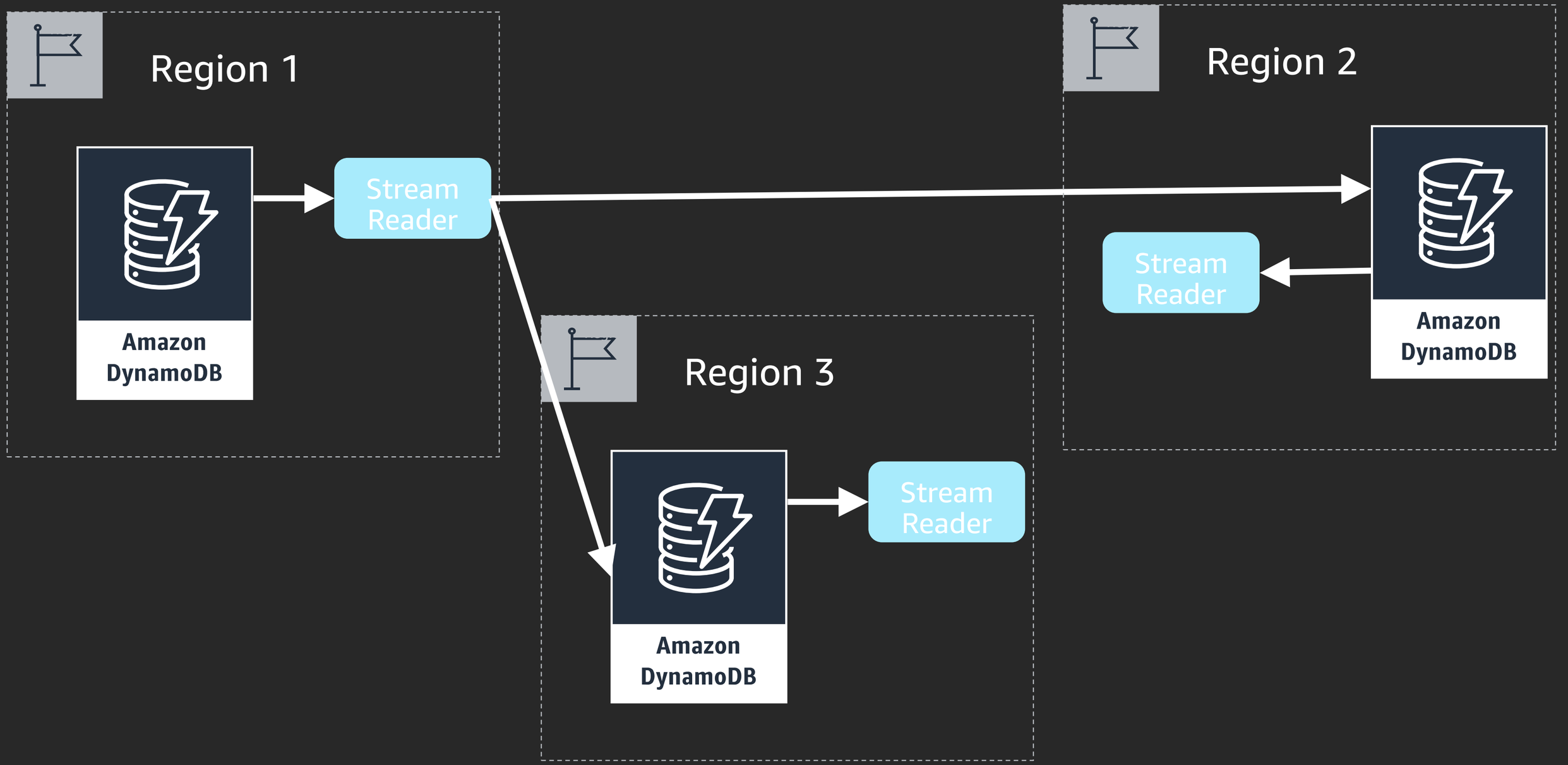
Records

Stream Checkpointing

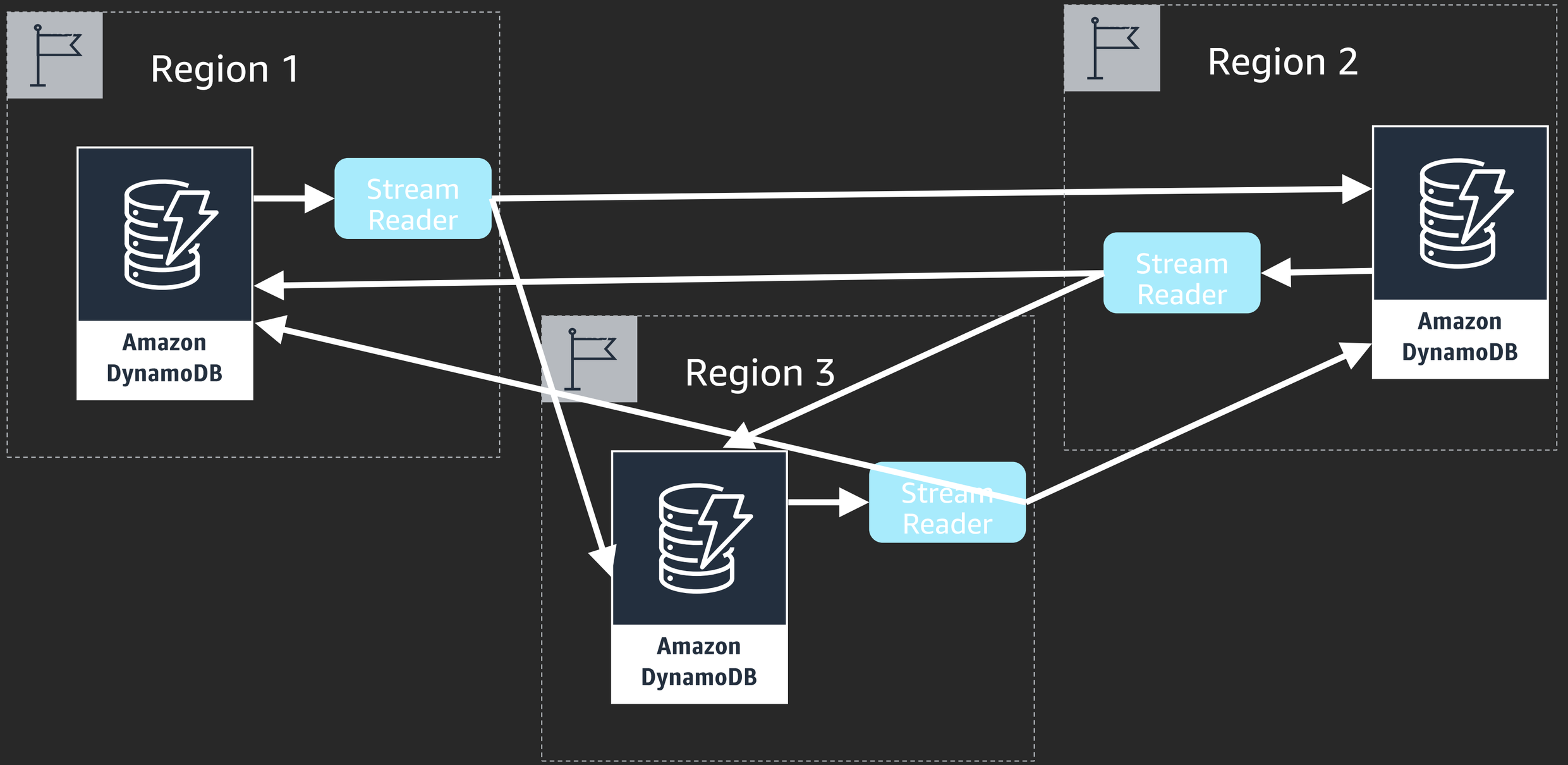
Multi-master replication



Multi-region replication



Multi-region



Questions

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Autoscaling

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Provisioning Example

300 Read Capacity Units (RCU)

Ignoring Writes

Provisioning

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

100 RCUs

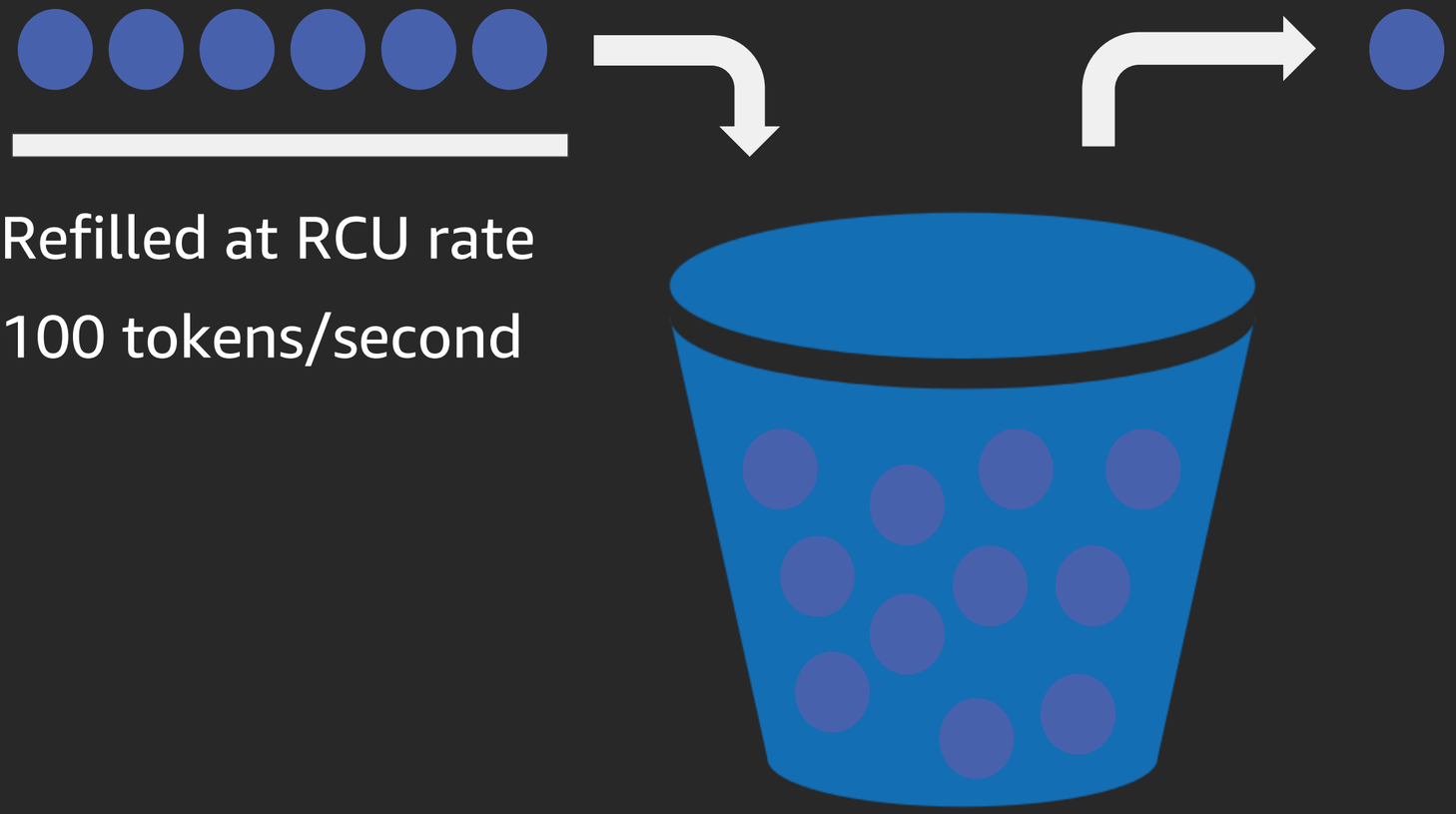
0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

100 RCUs

0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

100 RCUs

Token Bucket Algorithm



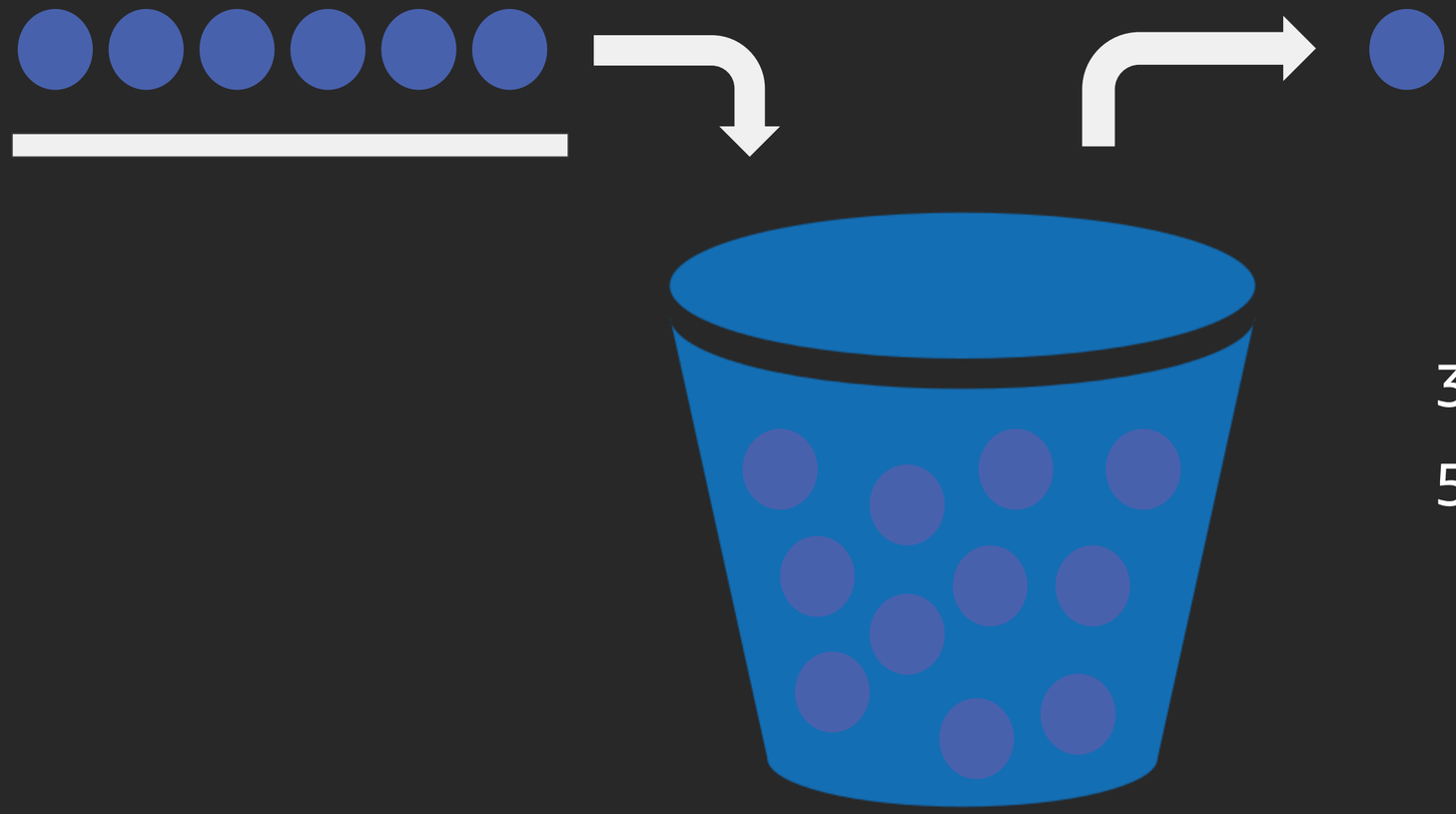
Refilled at RCU rate

100 tokens/second

Emptied 1 Token per Request*

*Tokens deducted depends on Item size

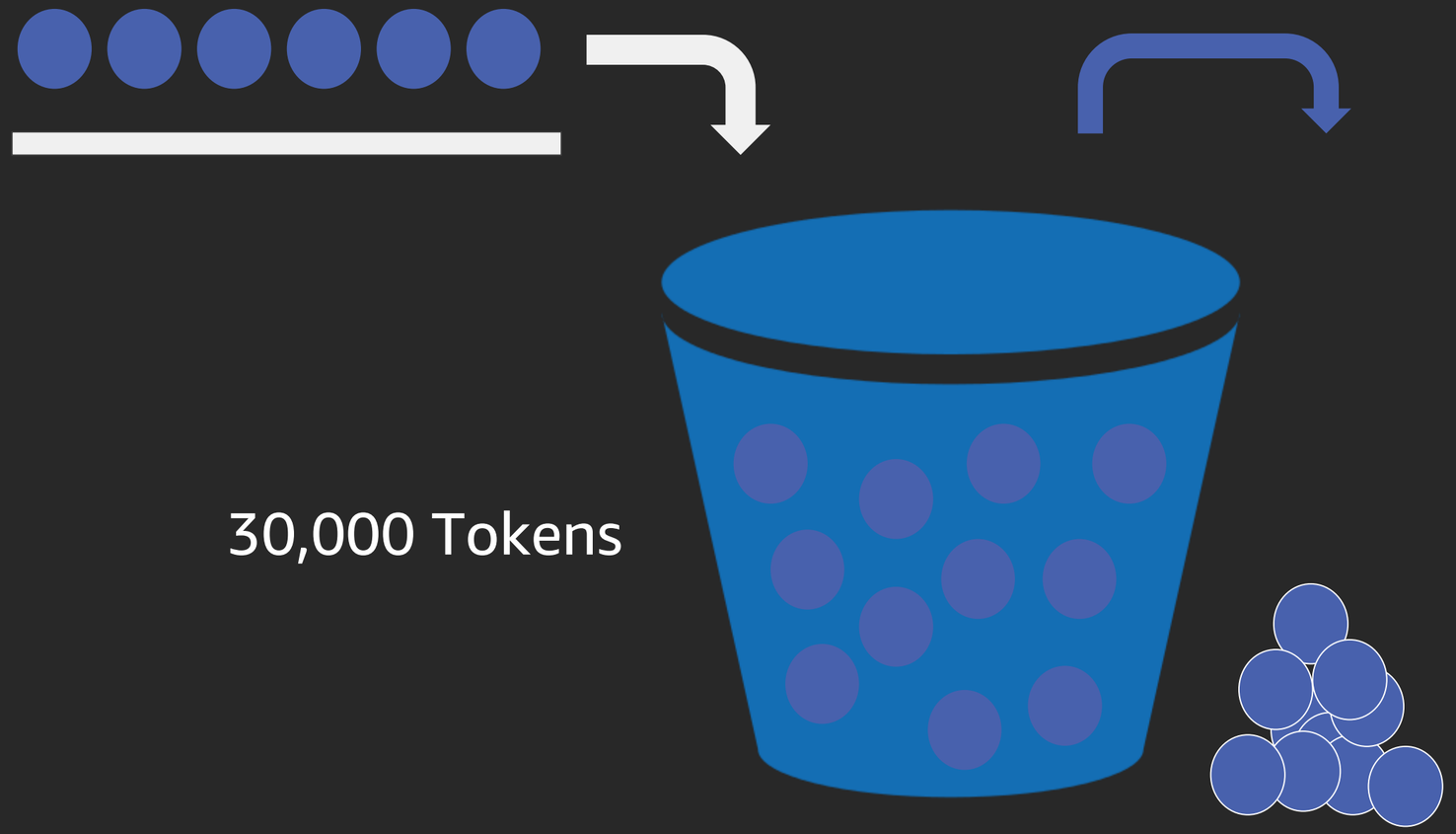
Token Bucket Capacity



300 * RCUs

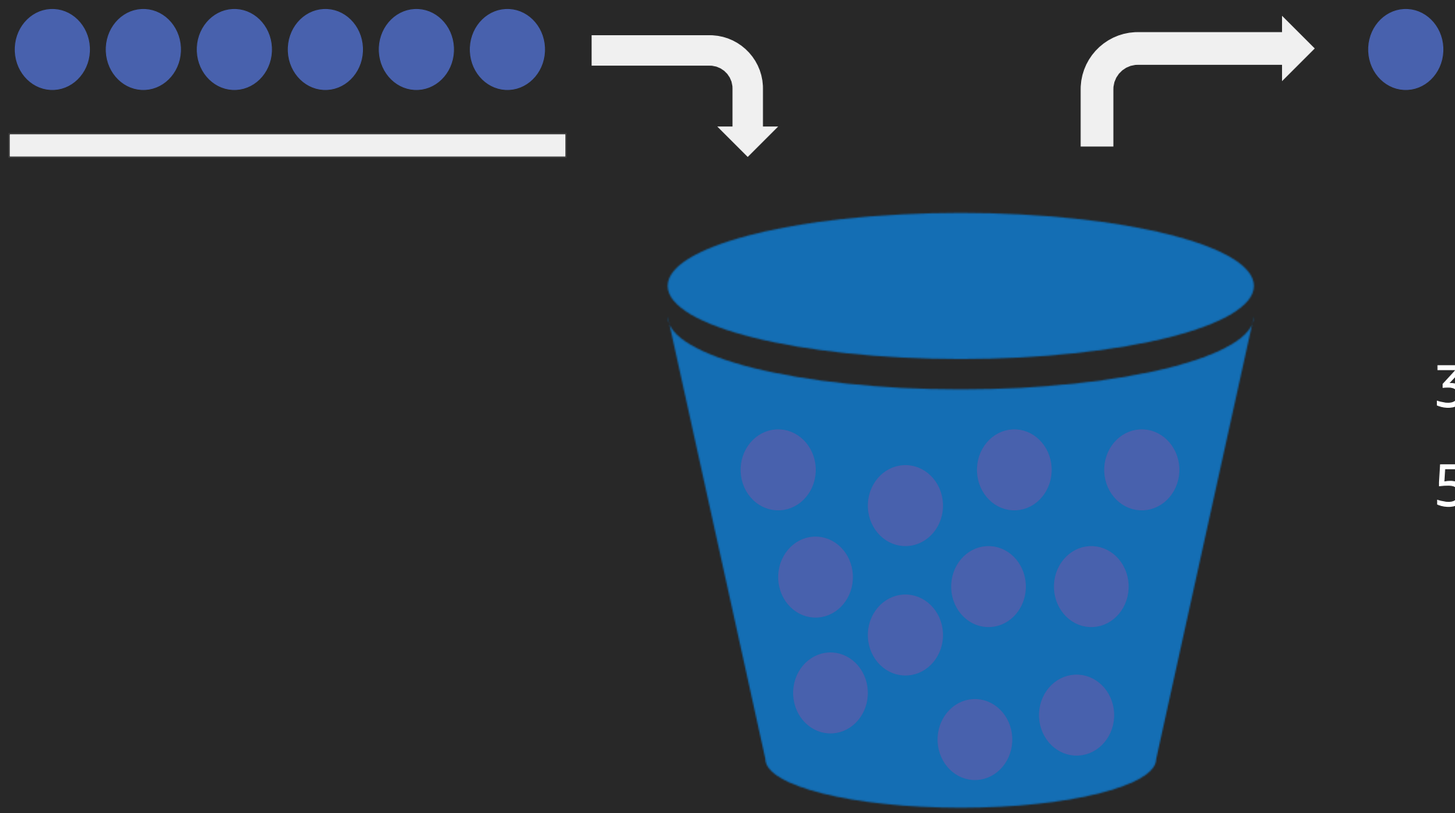
5 minutes

Full Token Bucket



30,000 Tokens

Partition Bursting



300 * RCUs

5 minutes

Unbalanced Load

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

25 RCU attempted

0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

150 RCU attempted

0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

50 RCU attempted

Unbalanced Load

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

75 RCU unused

25 RCU admitted

OK

0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

100 RCU admitted

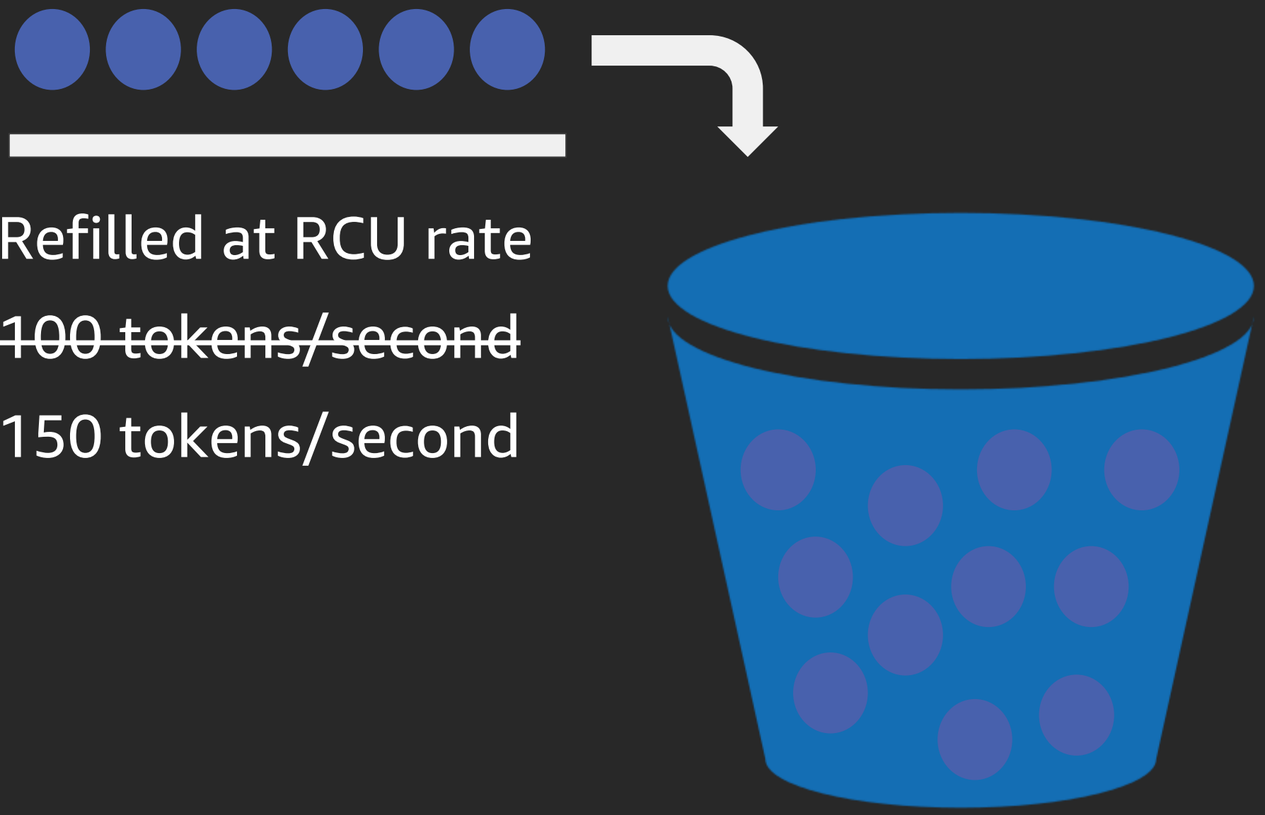
50 throttles per second

0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

50 RCU admitted

OK

Adaptive Capacity

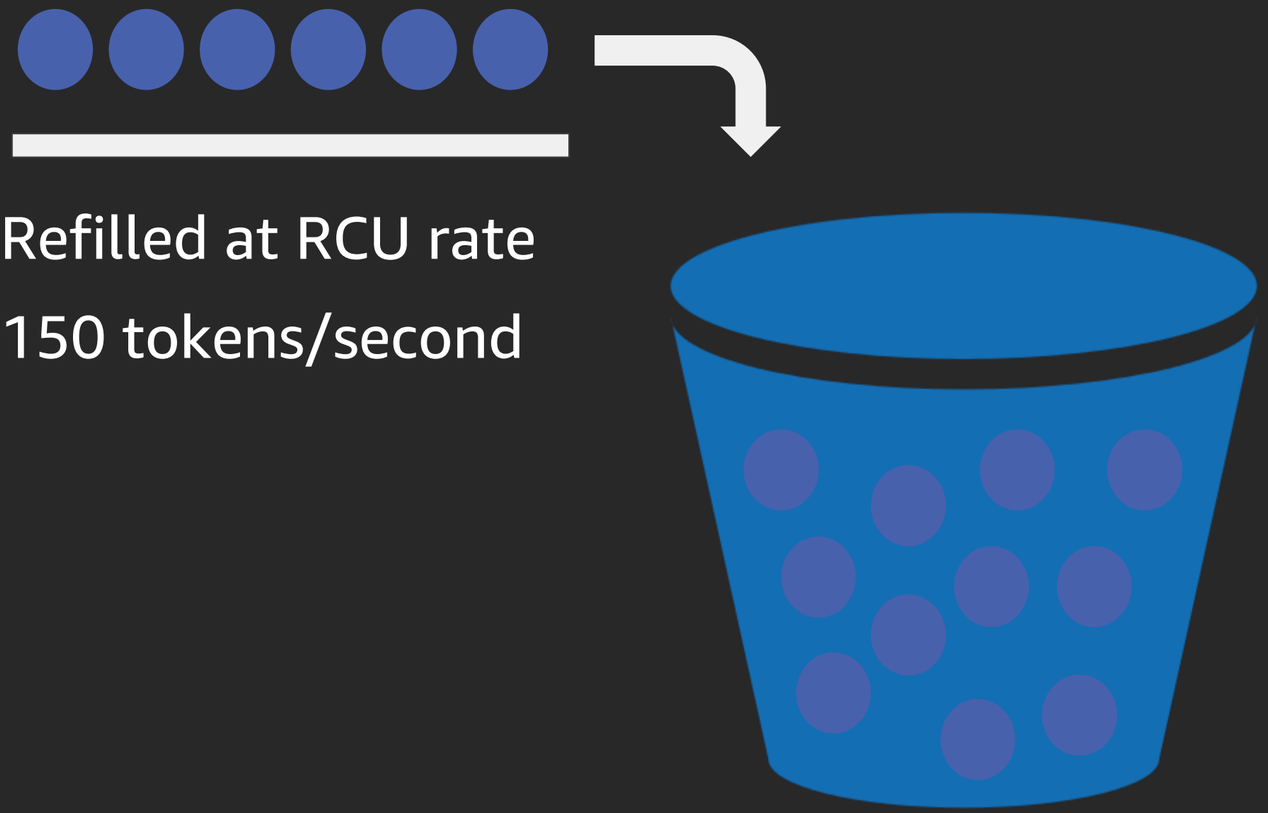


Refilled at RCU rate

~~100 tokens/second~~

150 tokens/second

Adaptive Capacity



Refilled at RCU rate

150 tokens/second

Adaptive Capacity

Multiplier = 1.5

Adaptive Capacity Active

75 RCU unused

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

25 RCU admitted OK

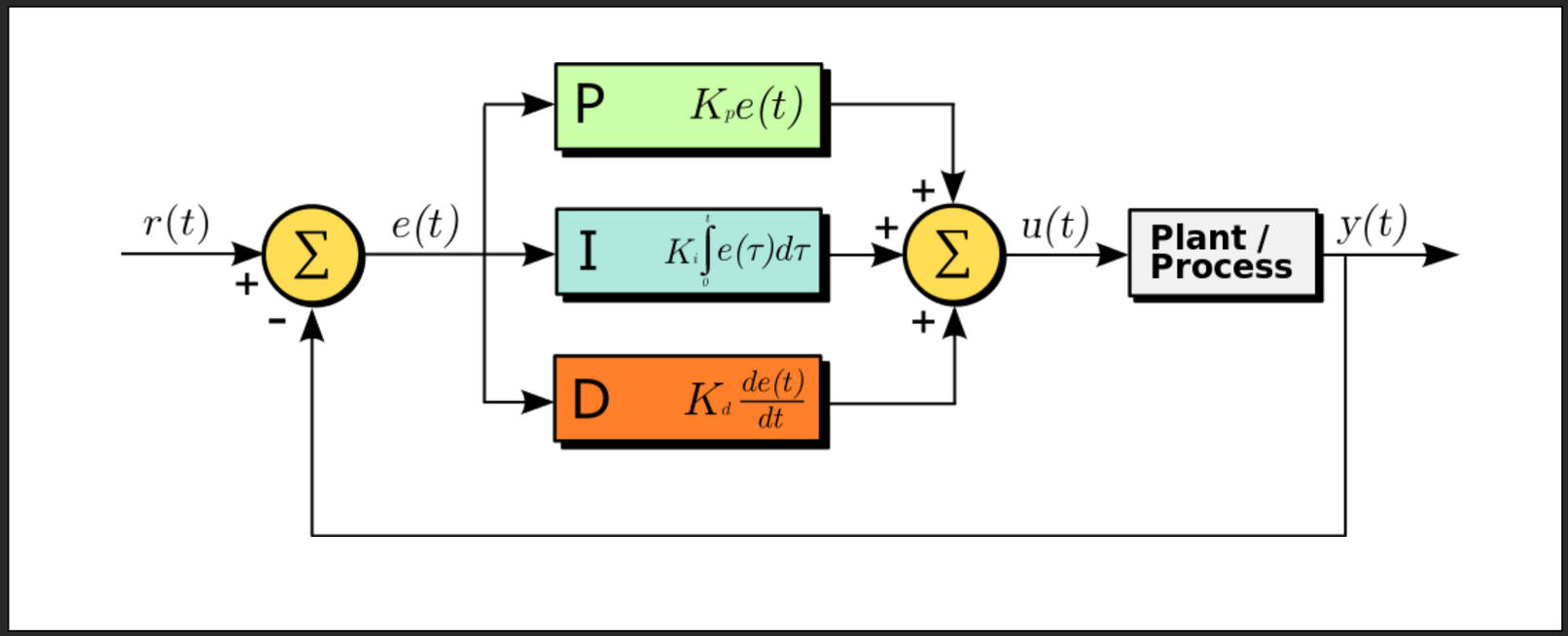
0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

150 RCU admitted OK

0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

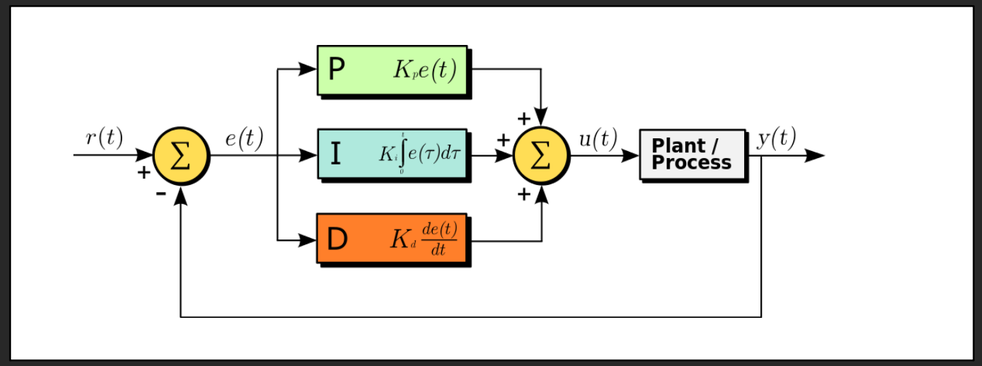
50 RCU admitted OK

PID Controller



https://commons.wikimedia.org/wiki/File:PID.svg

PID Controller

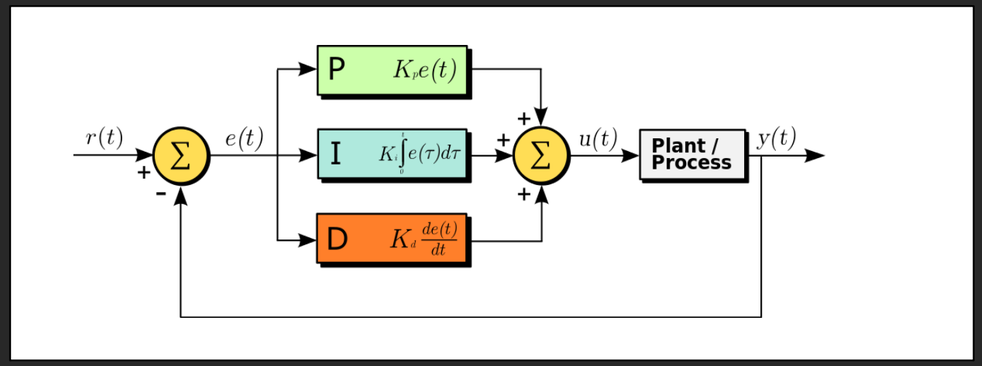


https://commons.wikimedia.org/wiki/File:PID.svg

PID inputs:

- Consumed Capacity
- Provisioned Capacity
- Throttling Rate
- Current Multiplier

PID Controller



https://commons.wikimedia.org/wiki/File:PID.svg

PID inputs:

Consumed Capacity

Provisioned Capacity

Throttling Rate

Current Multiplier

PID output:

New Multiplier

Too Much Allowed

150 RCU over provisioned

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

150 RCU admitted

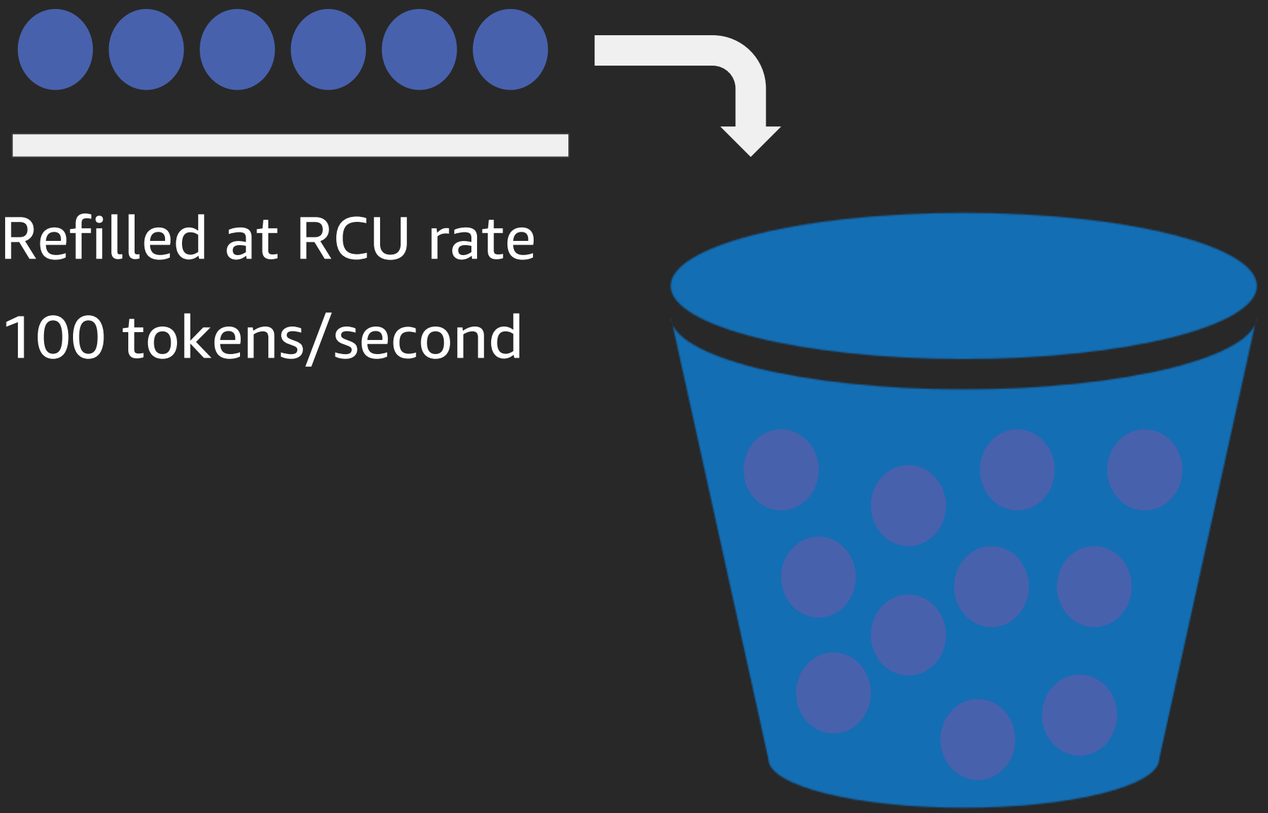
0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

150 RCU admitted

0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

150 RCU admitted

Partition Bursting



Refilled at RCU rate

100 tokens/second

Adaptive Capacity

~~Multiplier = 1.5~~

Multiplier = 1.0

Throttling

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

100 RCU admitted 50 Throttled

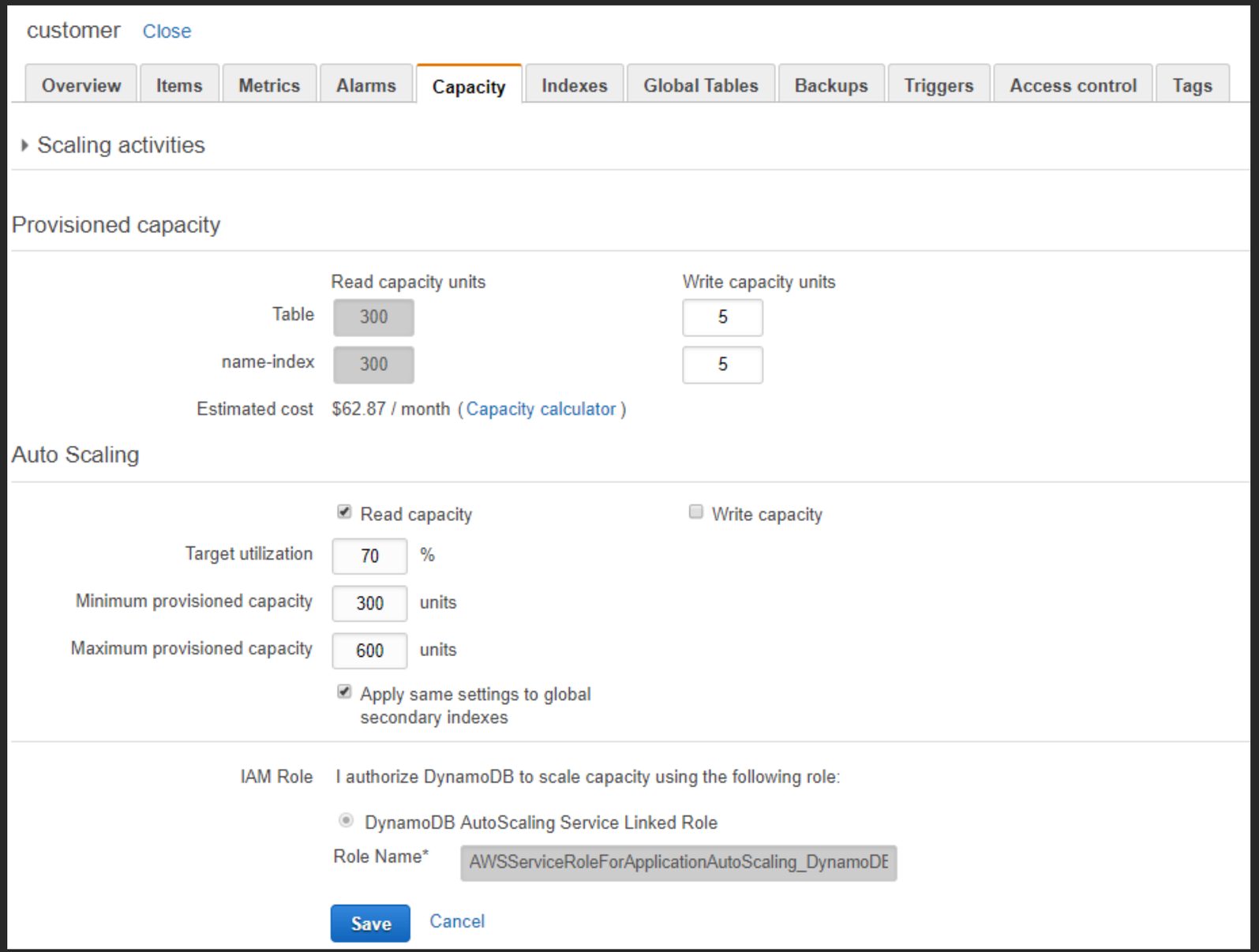
0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

100 RCU admitted 50 Throttled

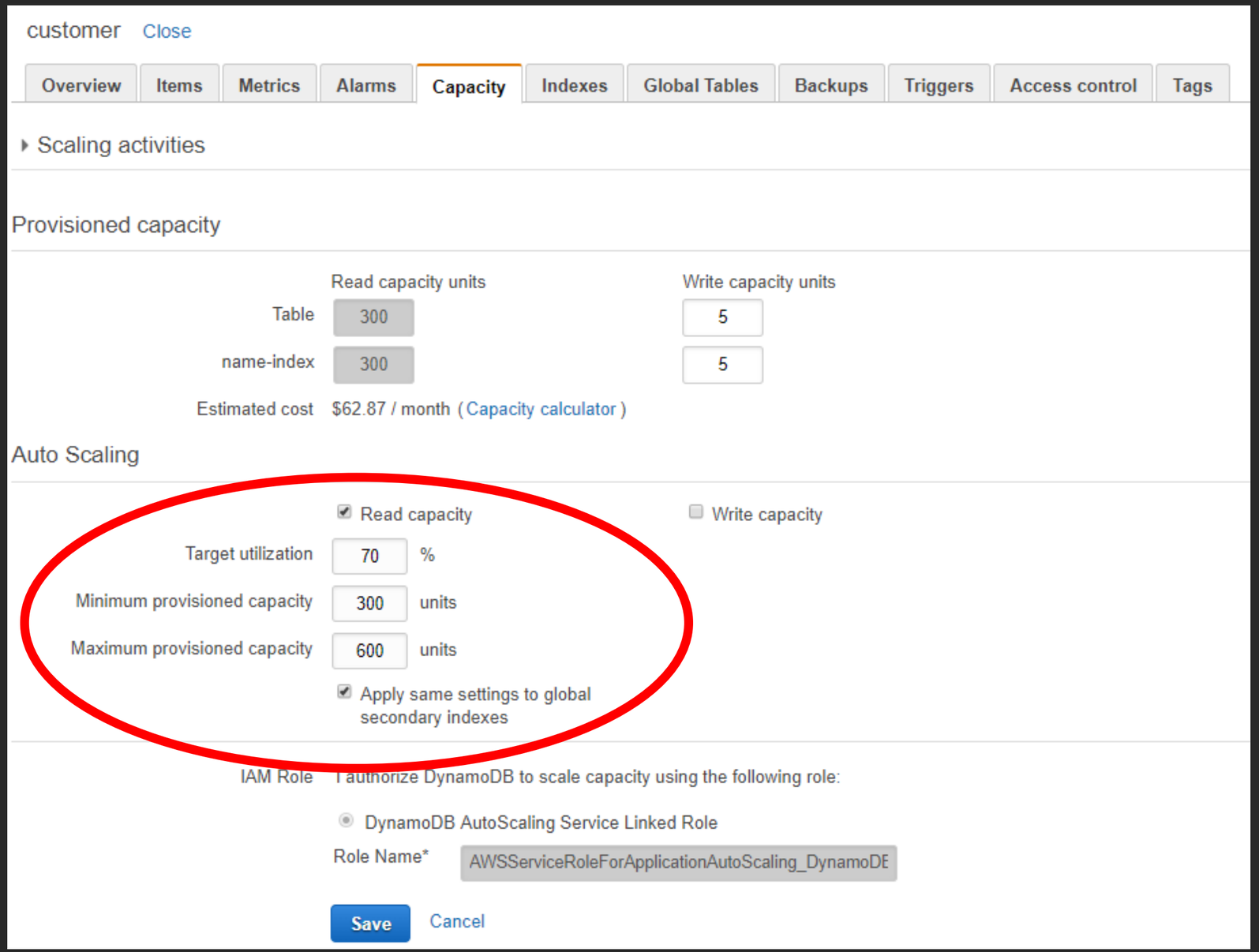
0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

100 RCU admitted 50 Throttled

Auto scaling



Auto scaling



Auto scaling

300 → 640 provisioned

70% of 640 ~ 450

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

150 RCU admitted

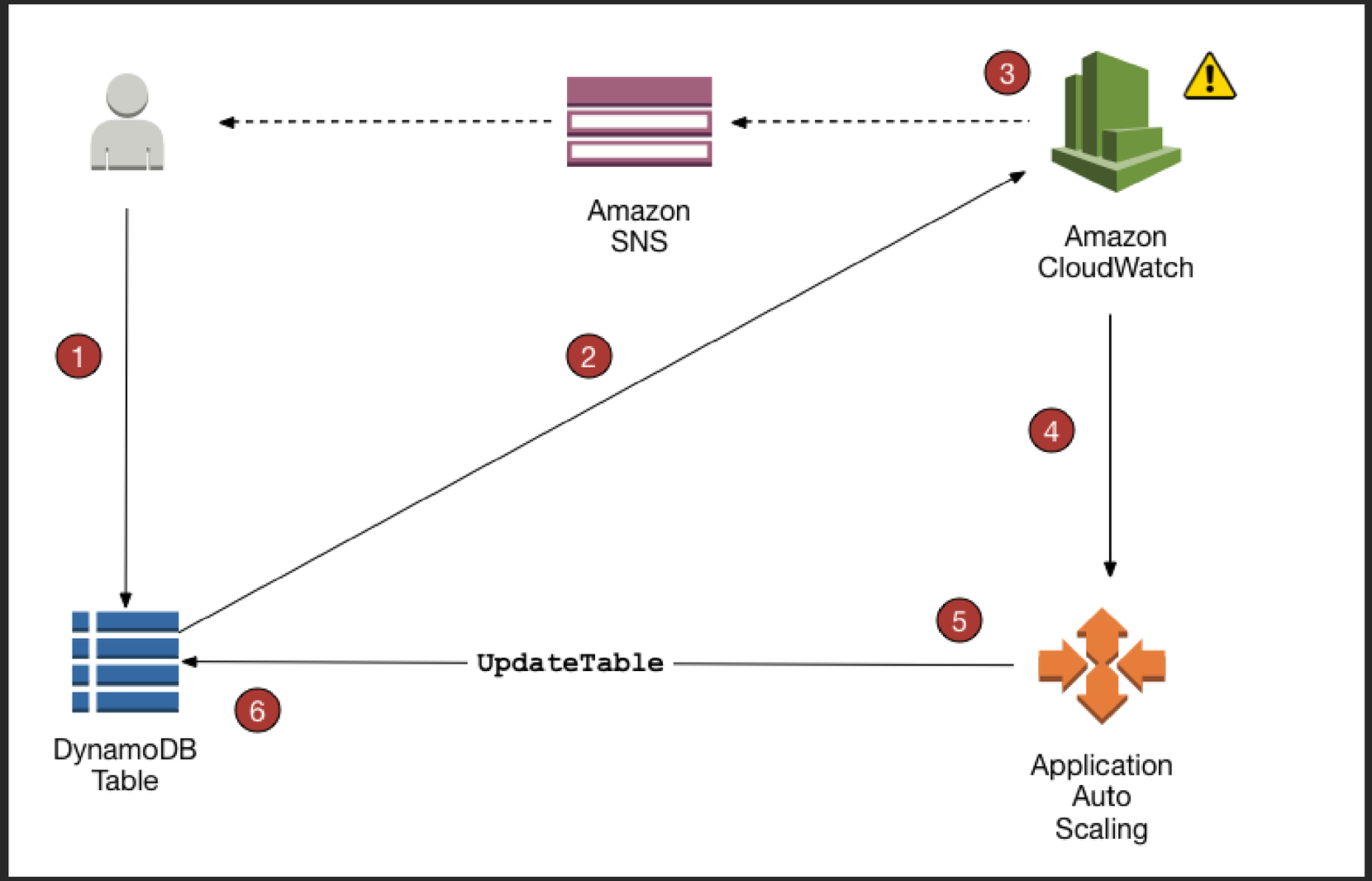
0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

150 RCU admitted

0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

150 RCU admitted

Auto scaling



Auto scaling

450 → 43 provisioned

70% of 43 ~ 30

0x12A8	236294	{ name:“Sara”, city:”Tampa”, …}
0x3391	445104	{ name:“James”, city:”Miami”, …}
0x6134	333363	{ name:“Betty”, city:”Madison”, …}

10 RCU Consumed

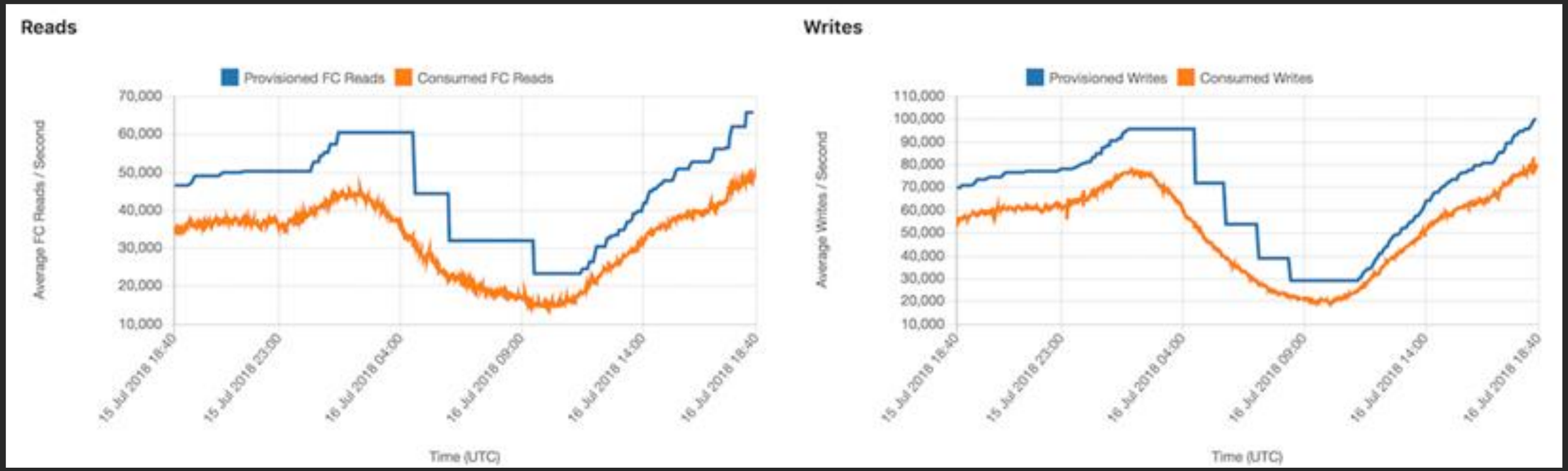
0x9531	145783	{ name:“Bob”, city:”London”, …}
0xB082	643145	{ name:“Val”, city:”Seattle”, …}

10 RCU Consumed

0xEA8A	723342	{ name:“Jeff”, city:”Toledo”, …}
0xF355	523422	{ name:“Alex”, city:”London”, …}

10 RCU Consumed

Actual Auto Scaled Table



Questions

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

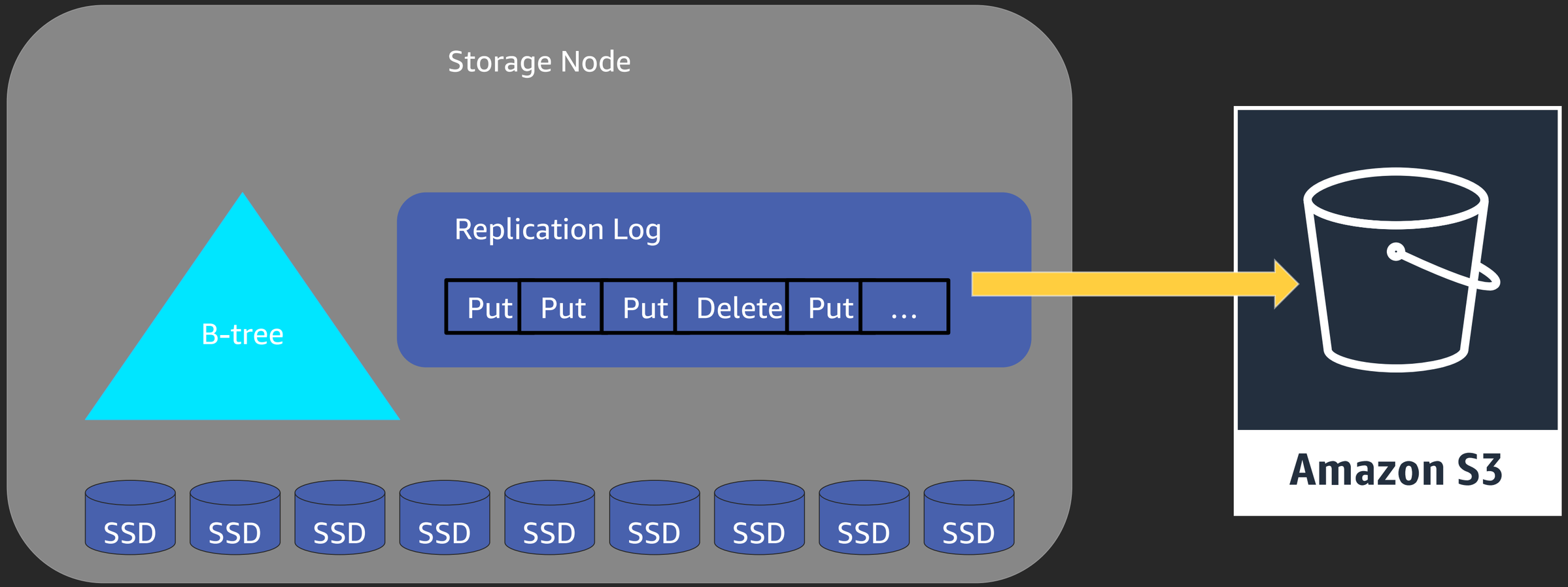
Backup / Restore

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

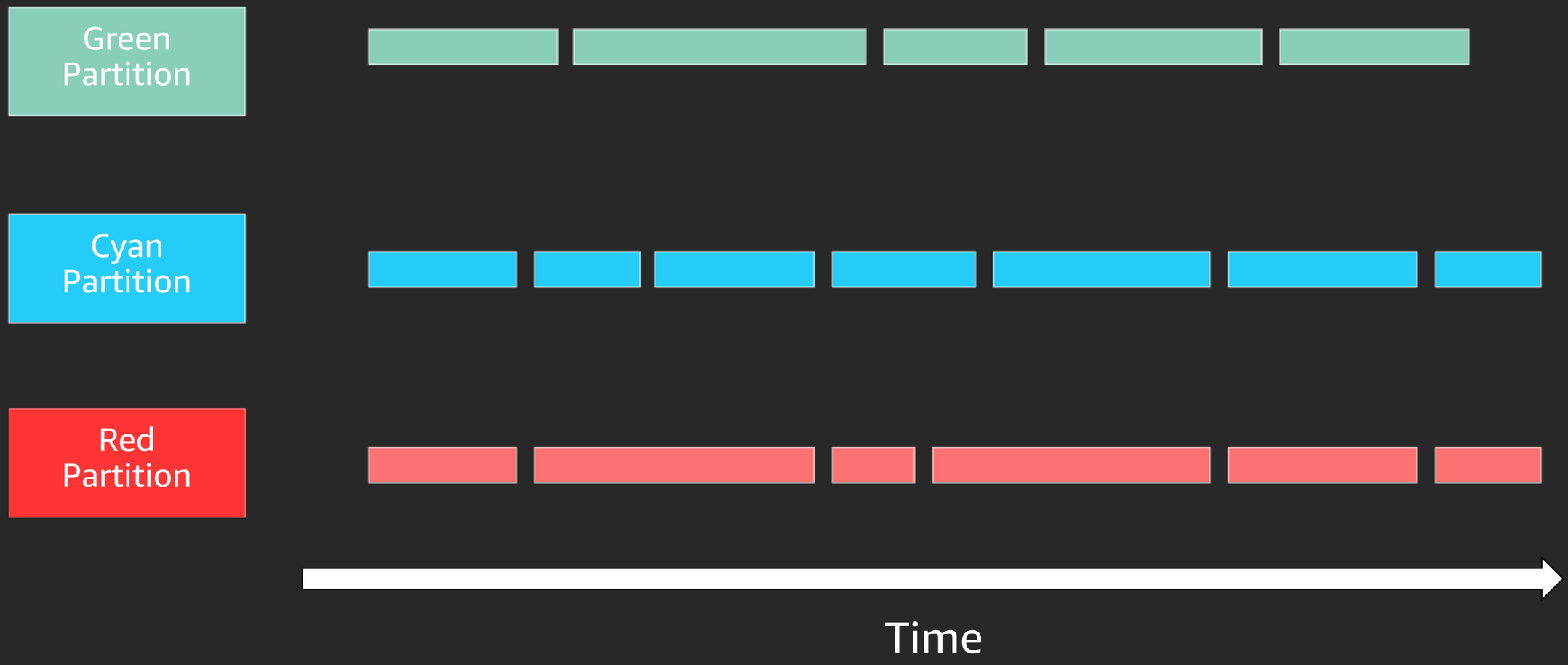
Backup and Restore

- Point in Time
- On-demand Backups

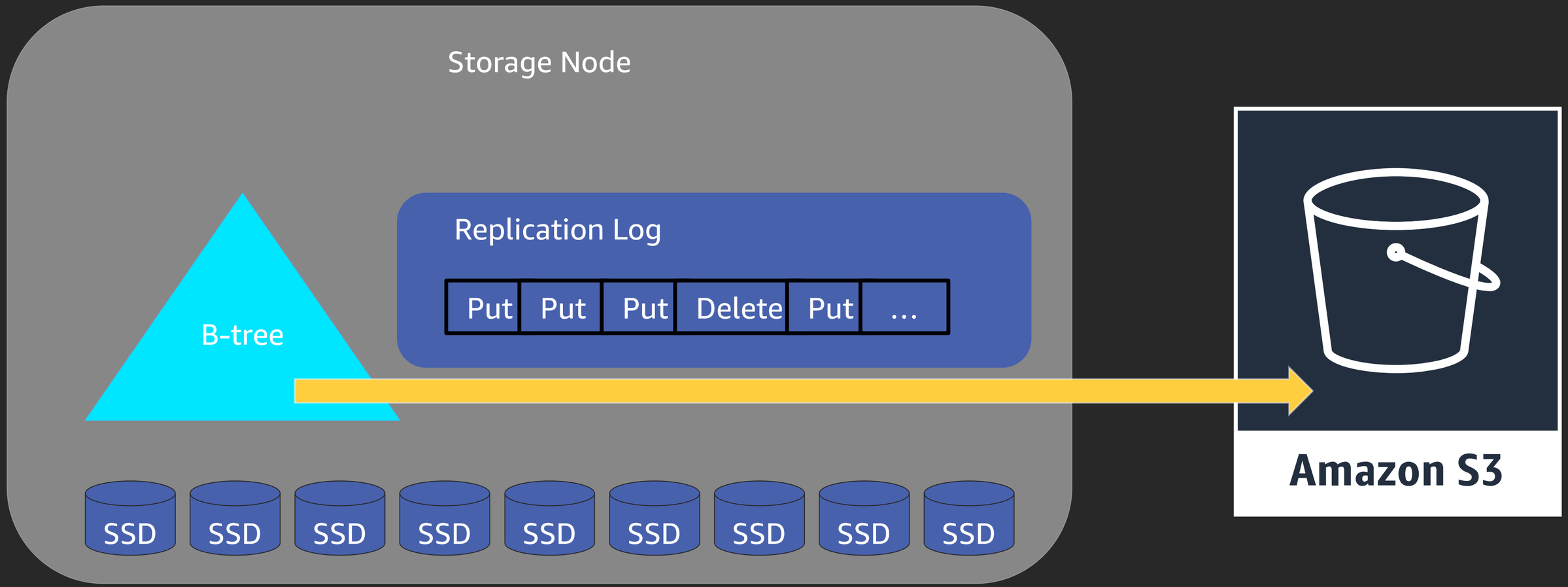
Replication Log to S3



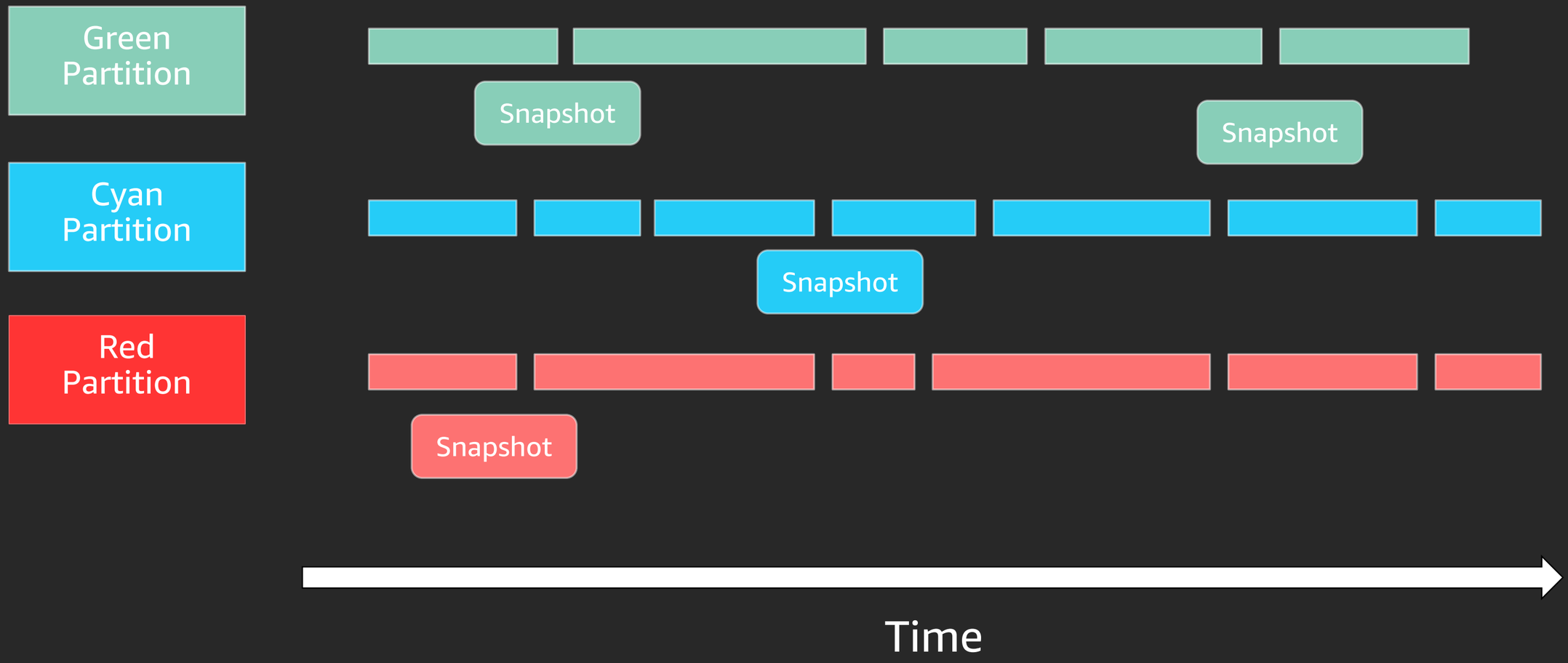
Replication Log to S3



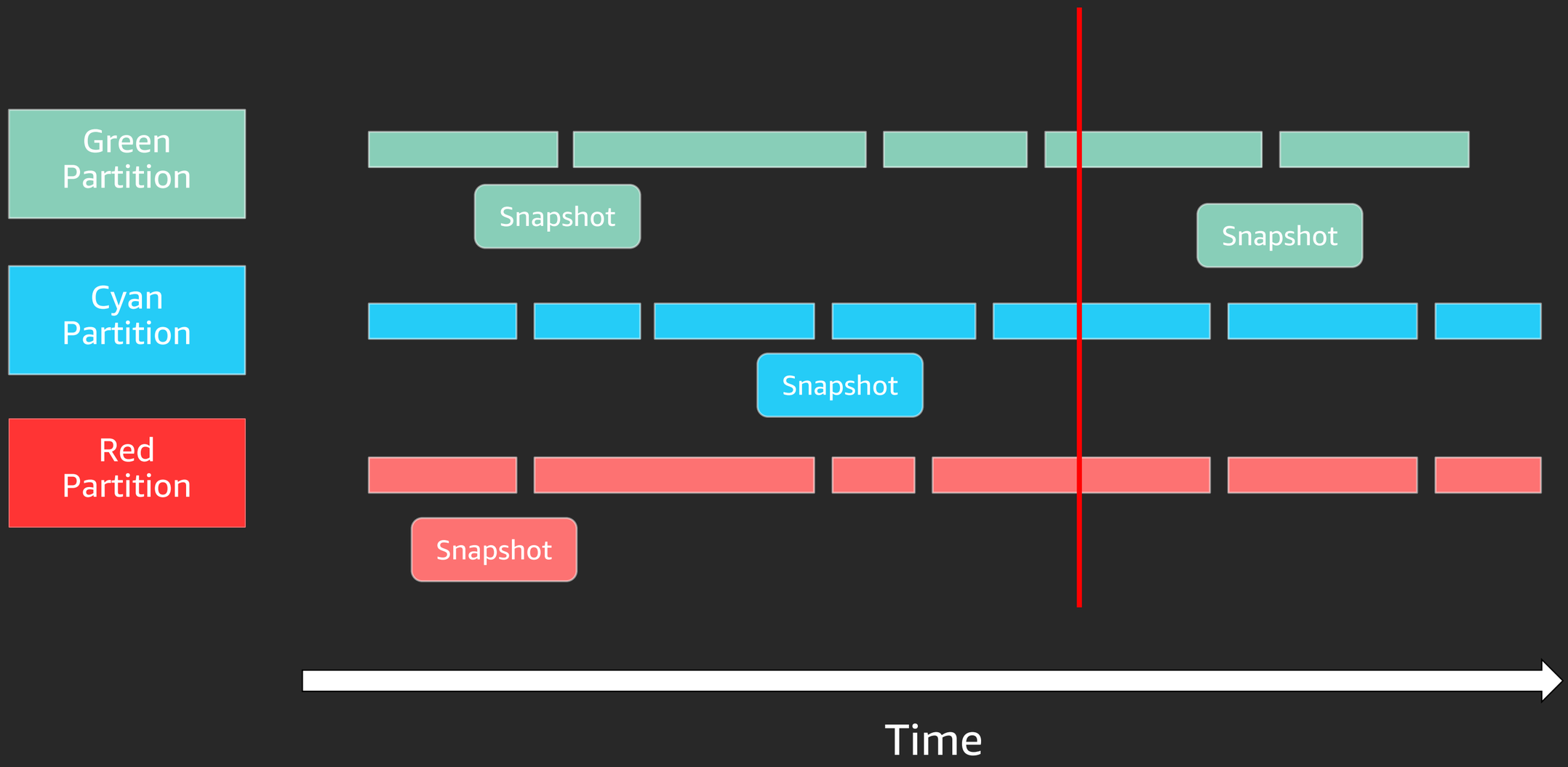
“Snapshot" the B-tree to S3



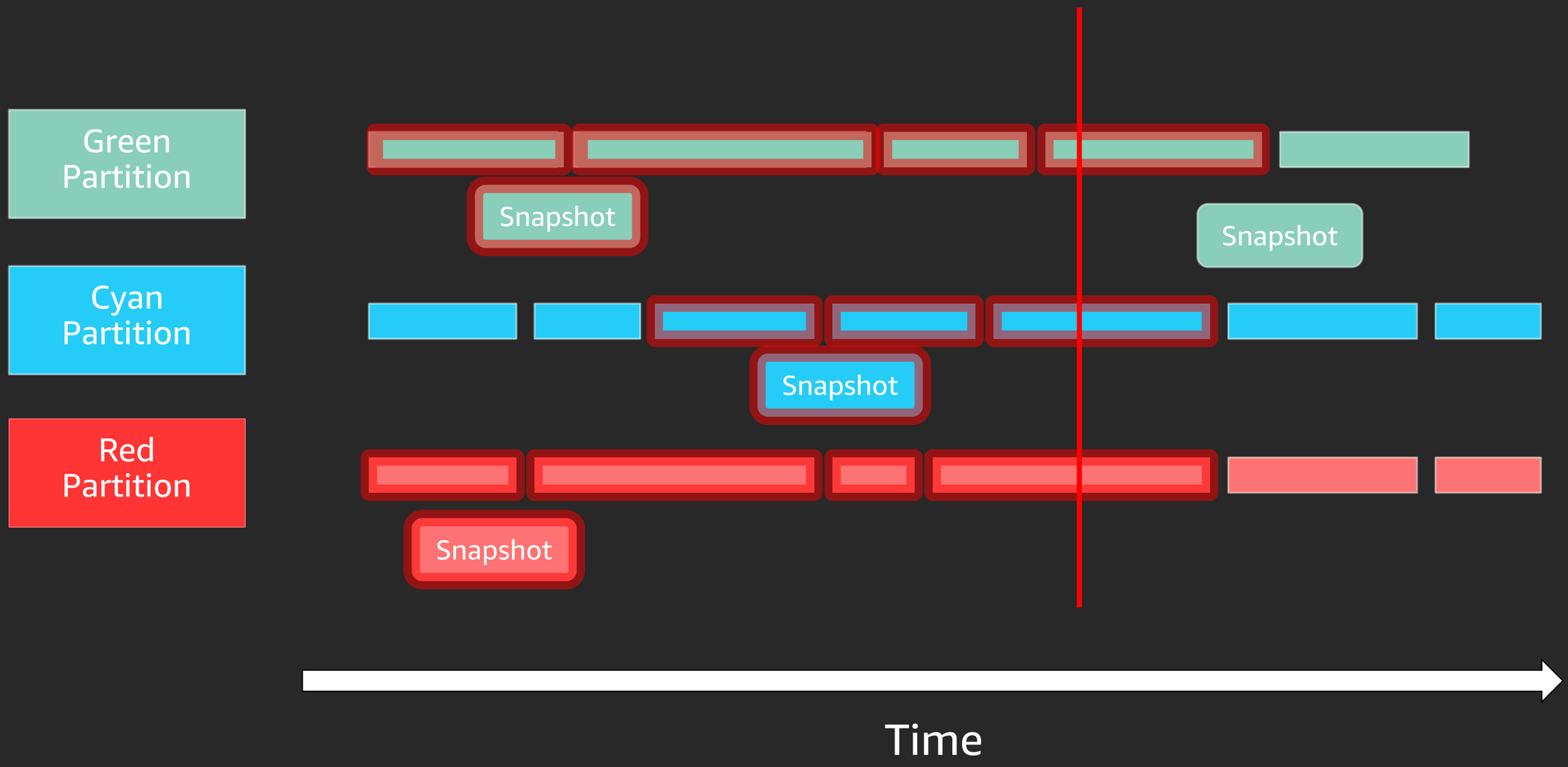
“Snapshot" the B-tree to S3



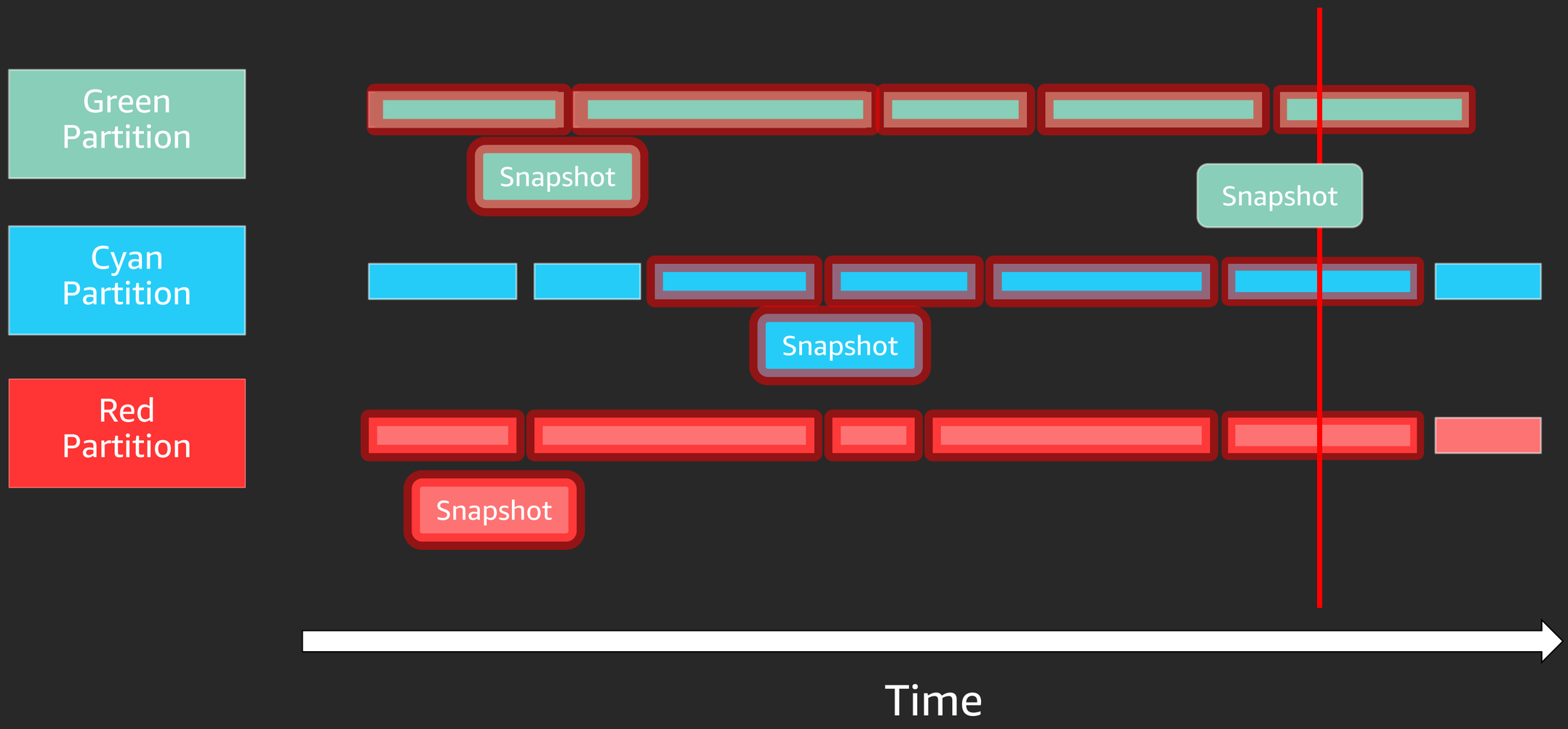
Restore to a Point in Time



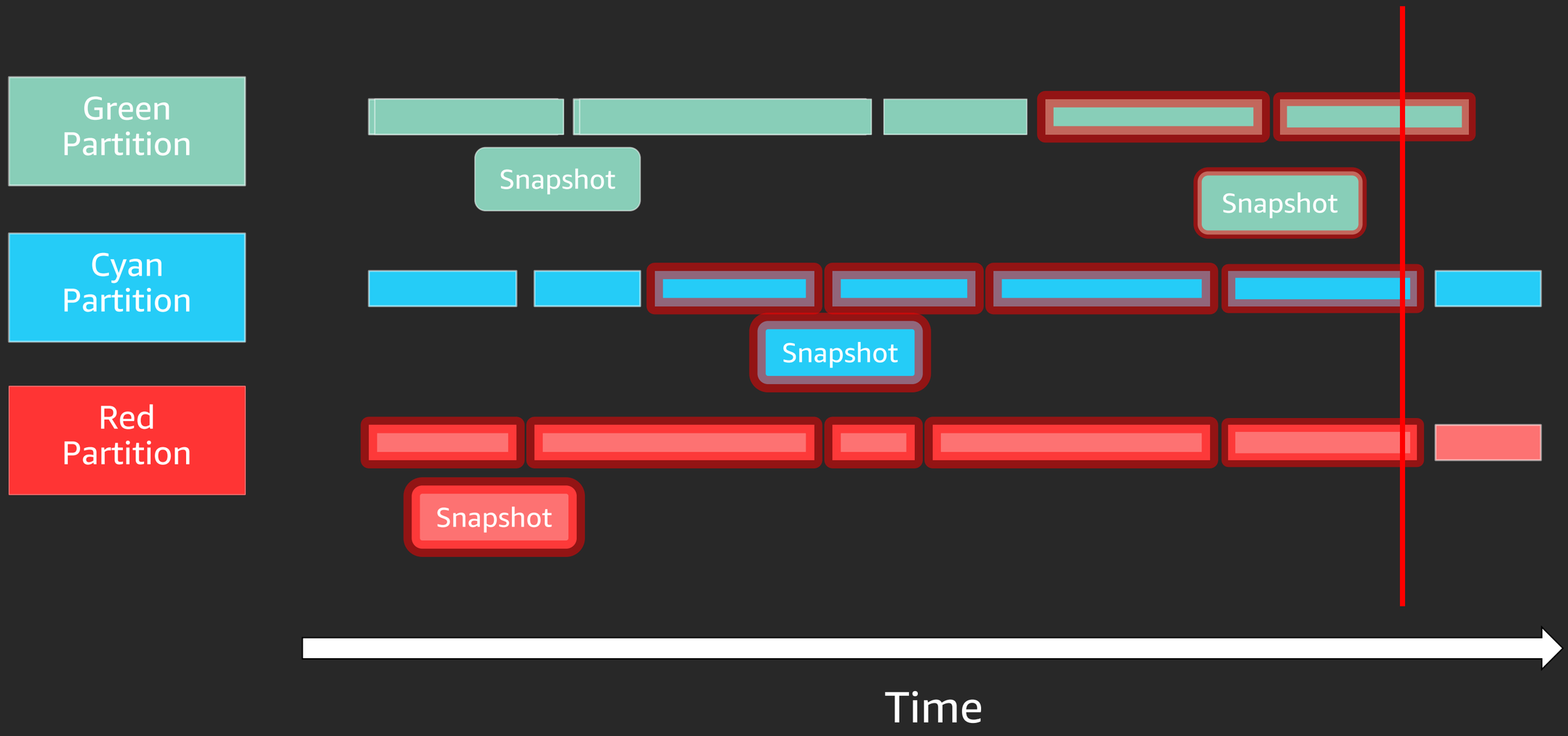
Restore to a Point in Time



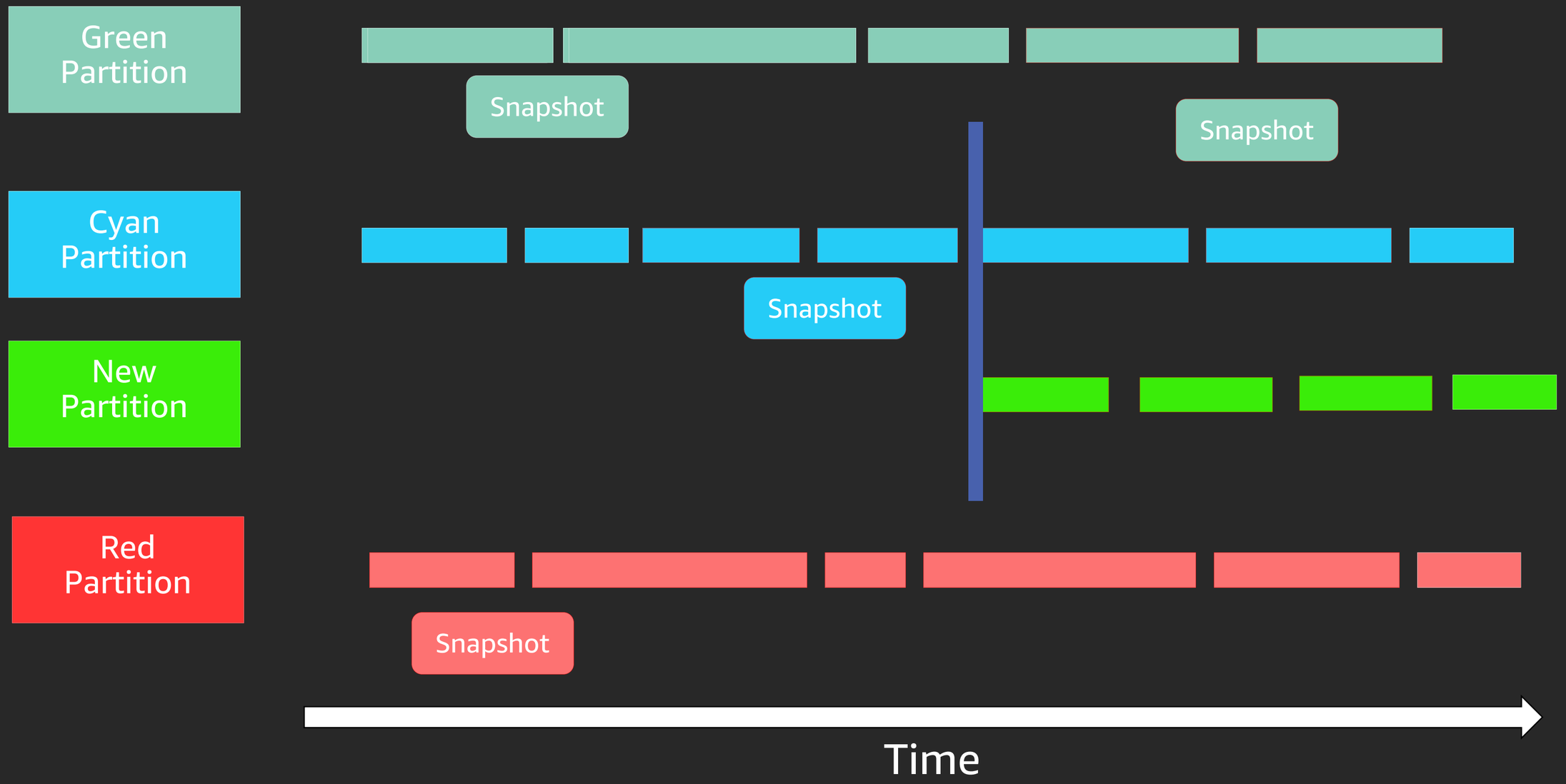
Restore to a Point in Time



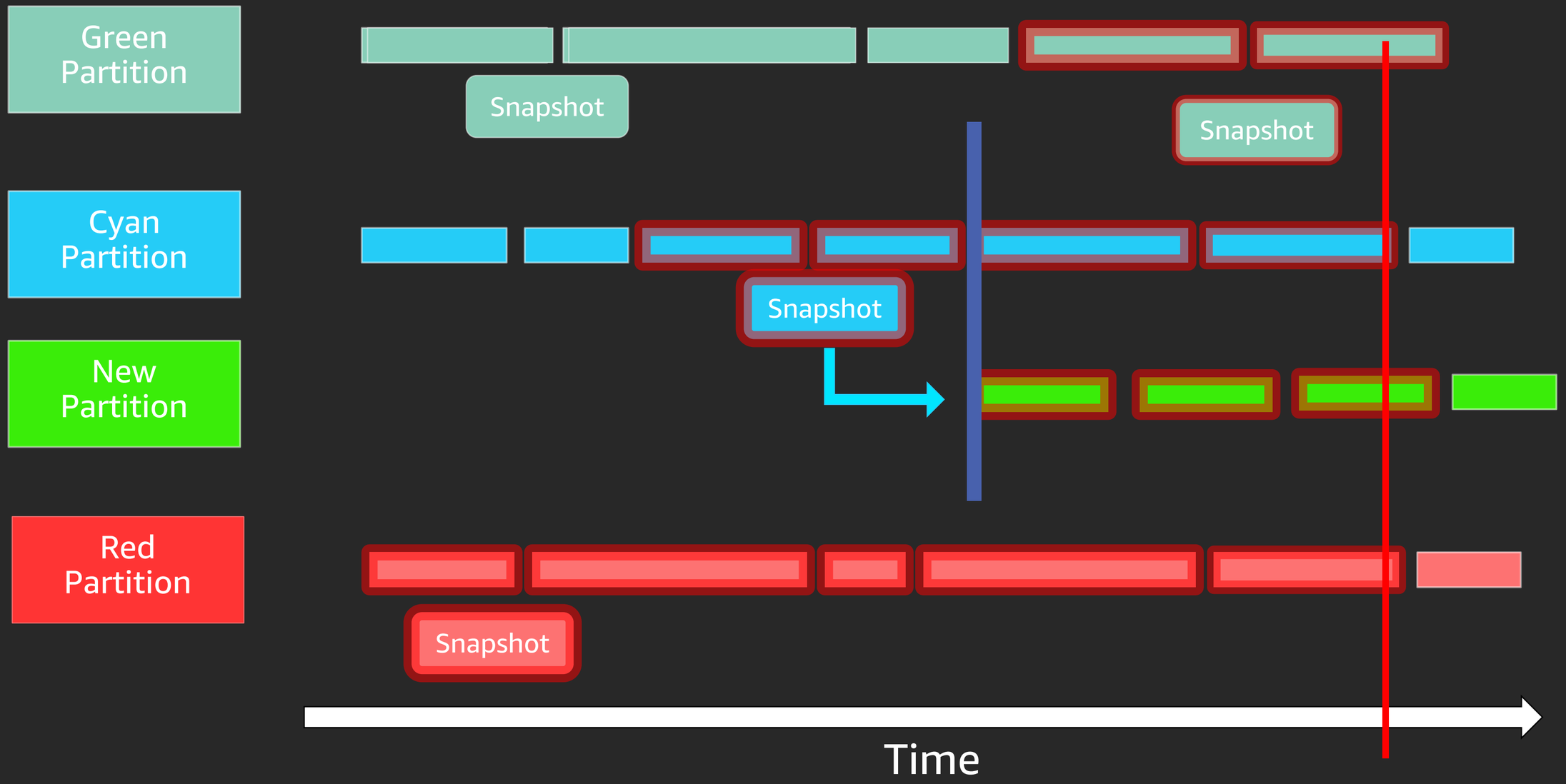
Restore to a Point in Time



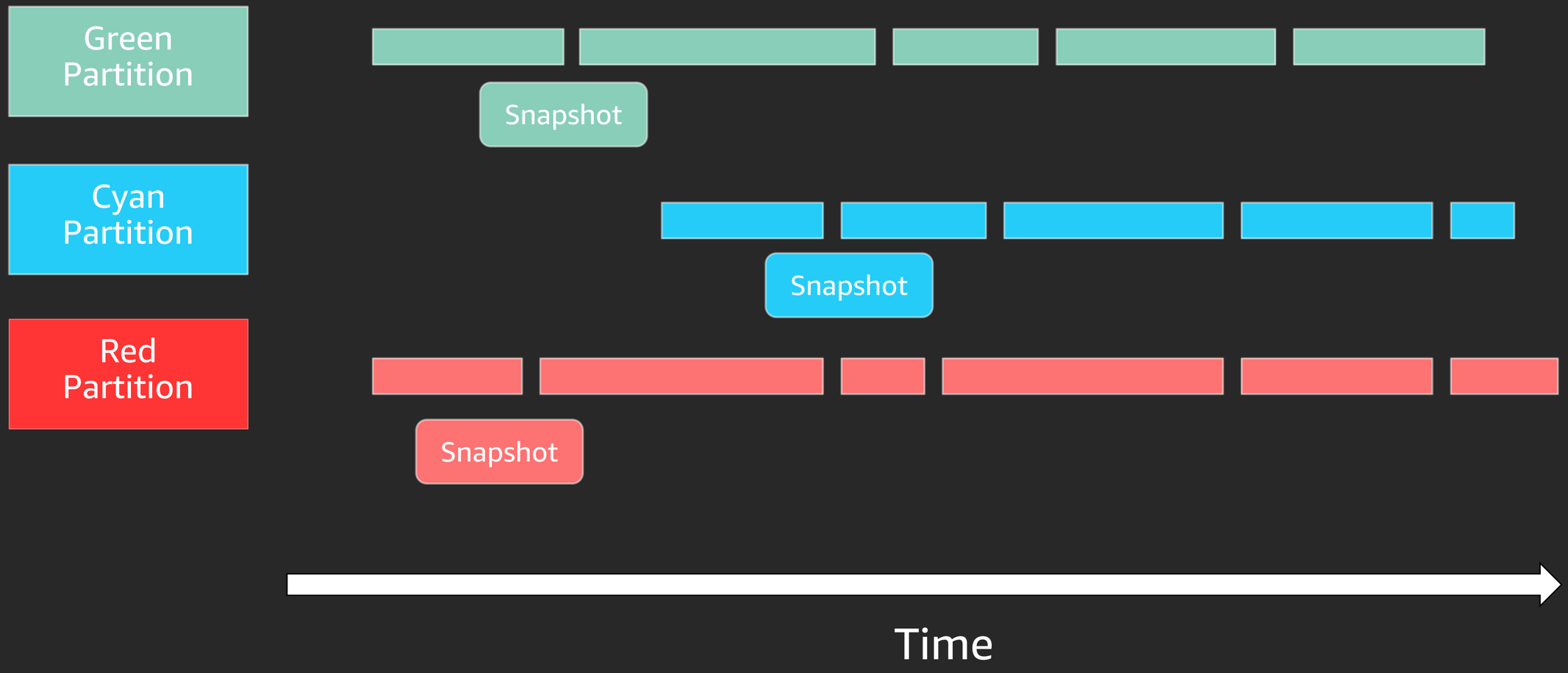
Partition Splits



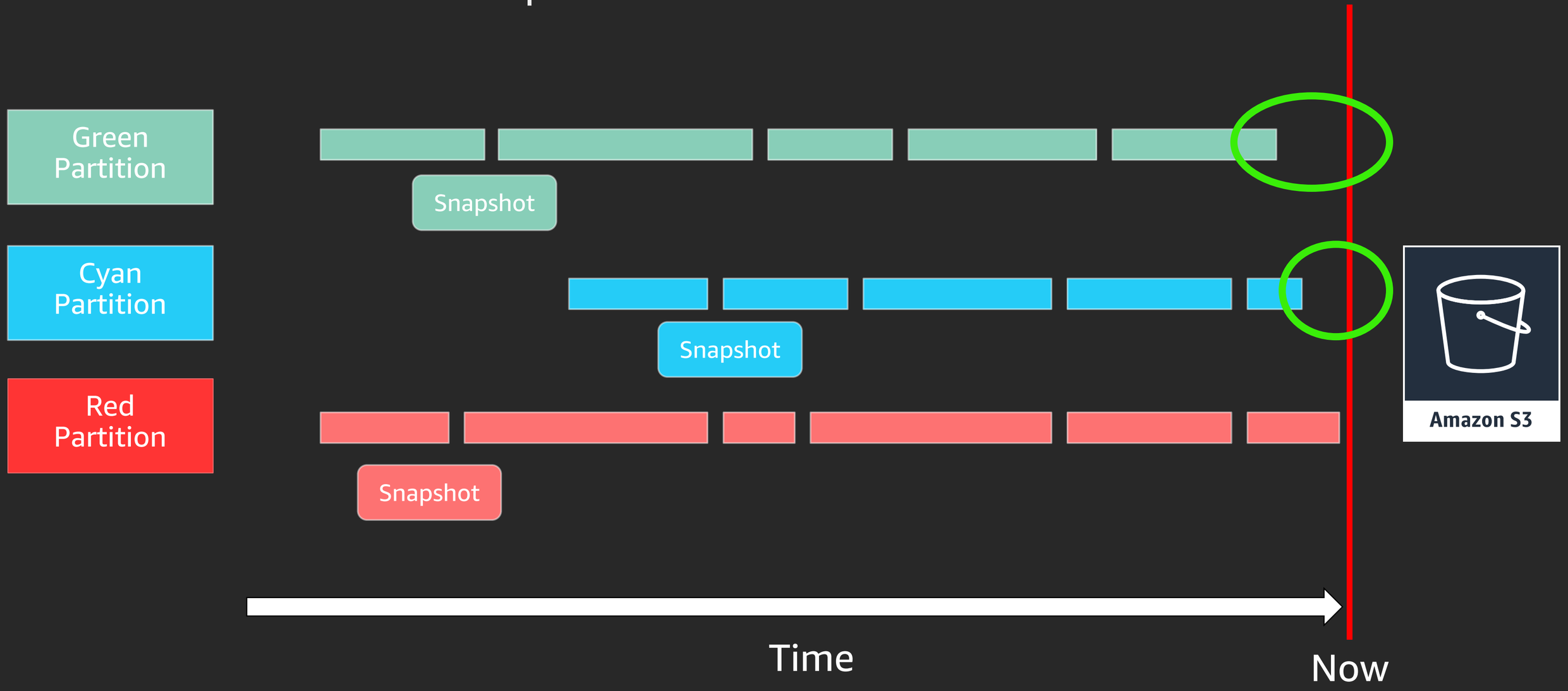
Restore to a Point in Time



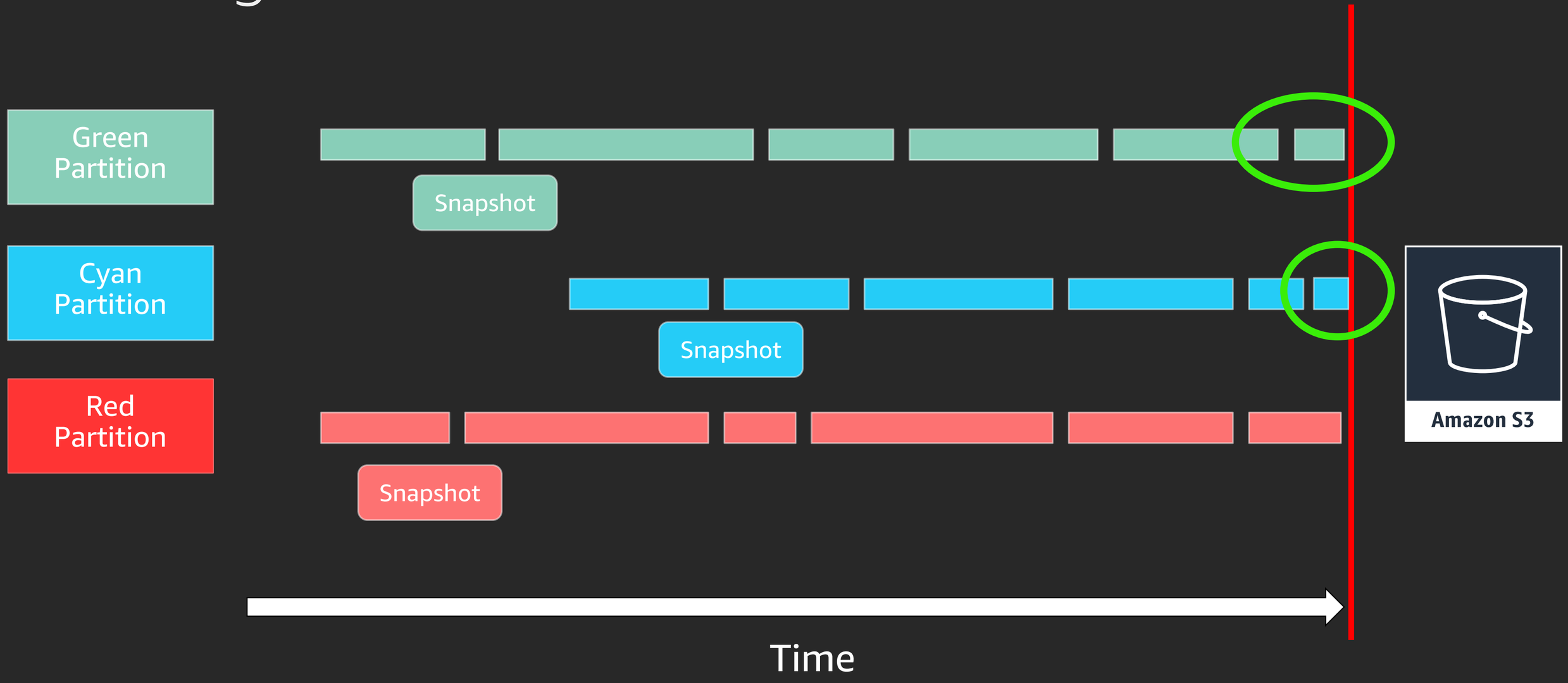
On-demand Backups



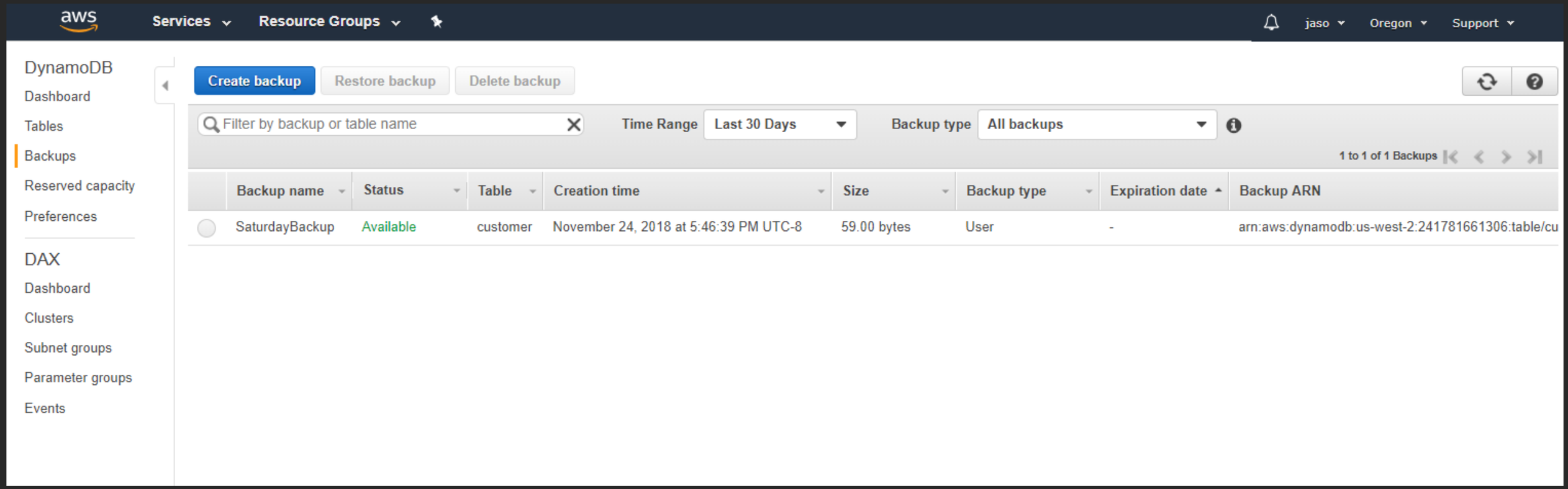
On-demand Backups



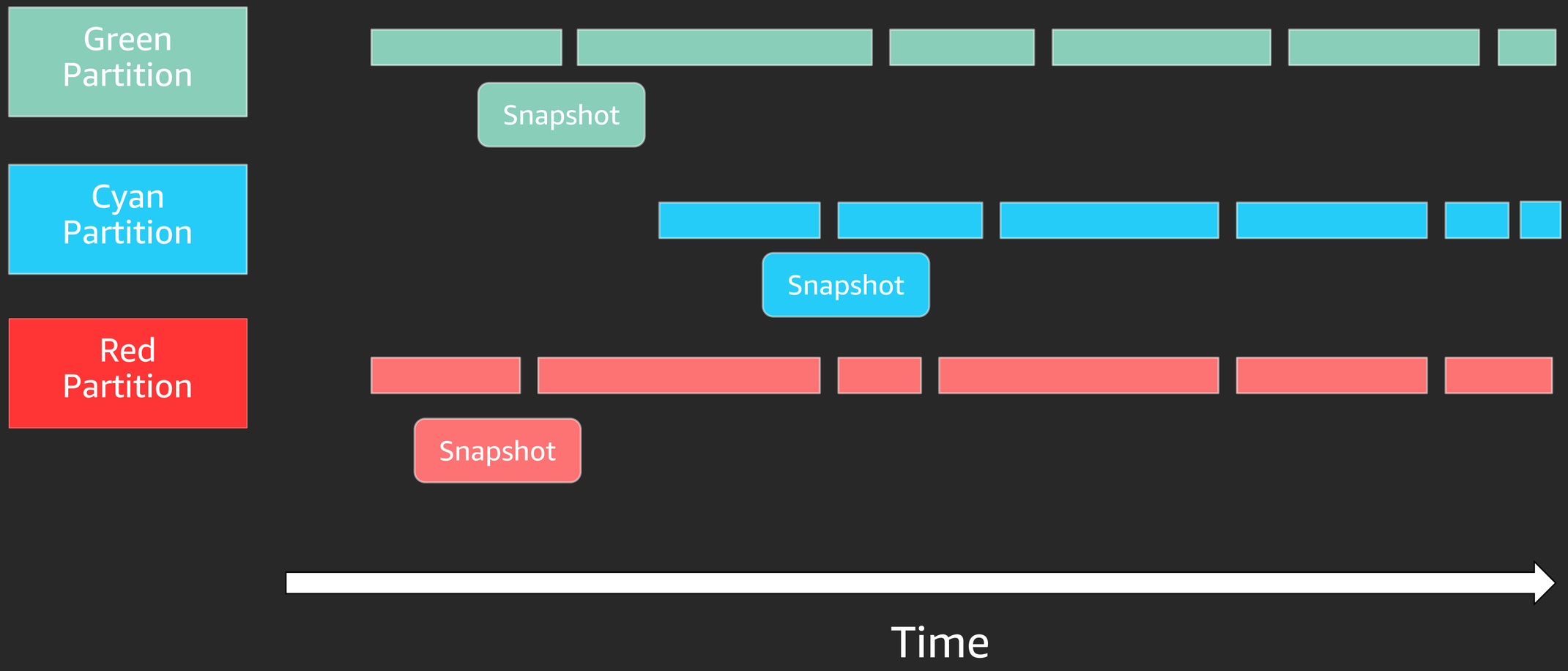
Write Logs to S3



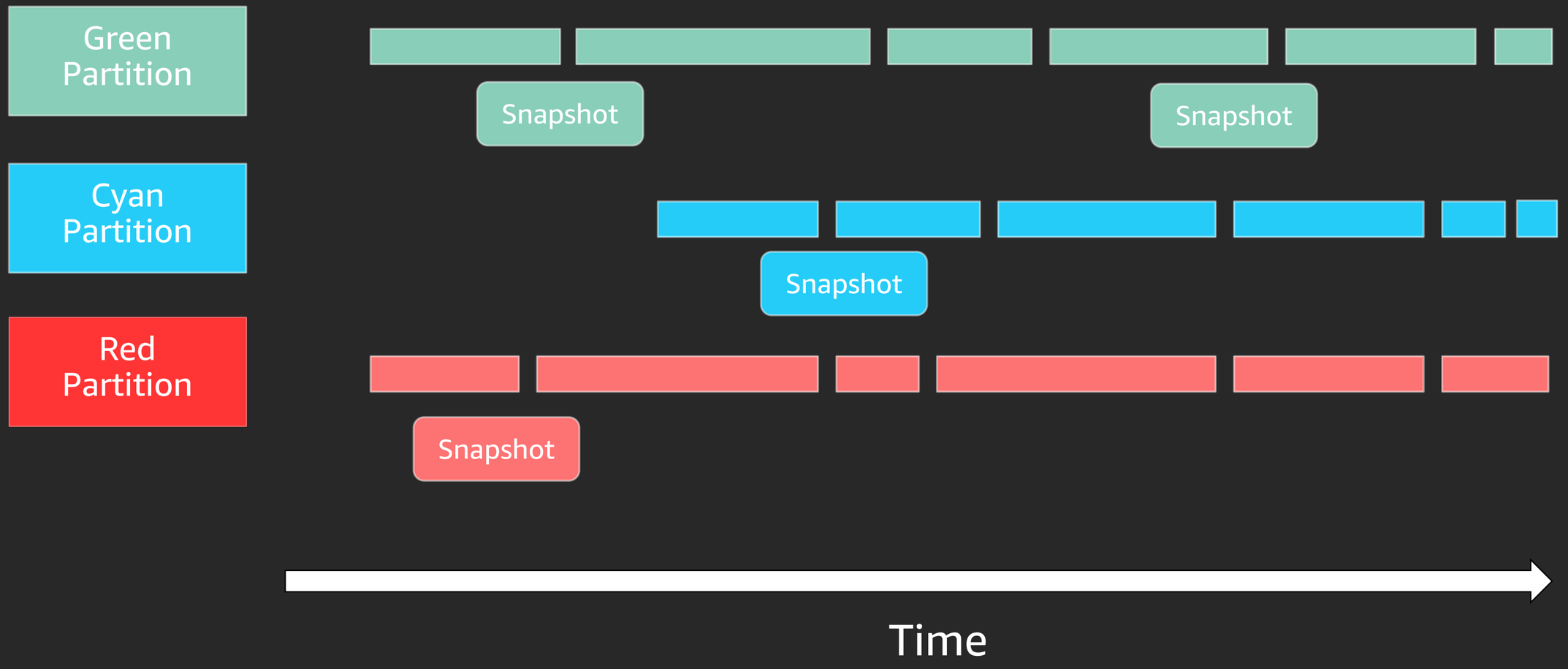
On-demand backup



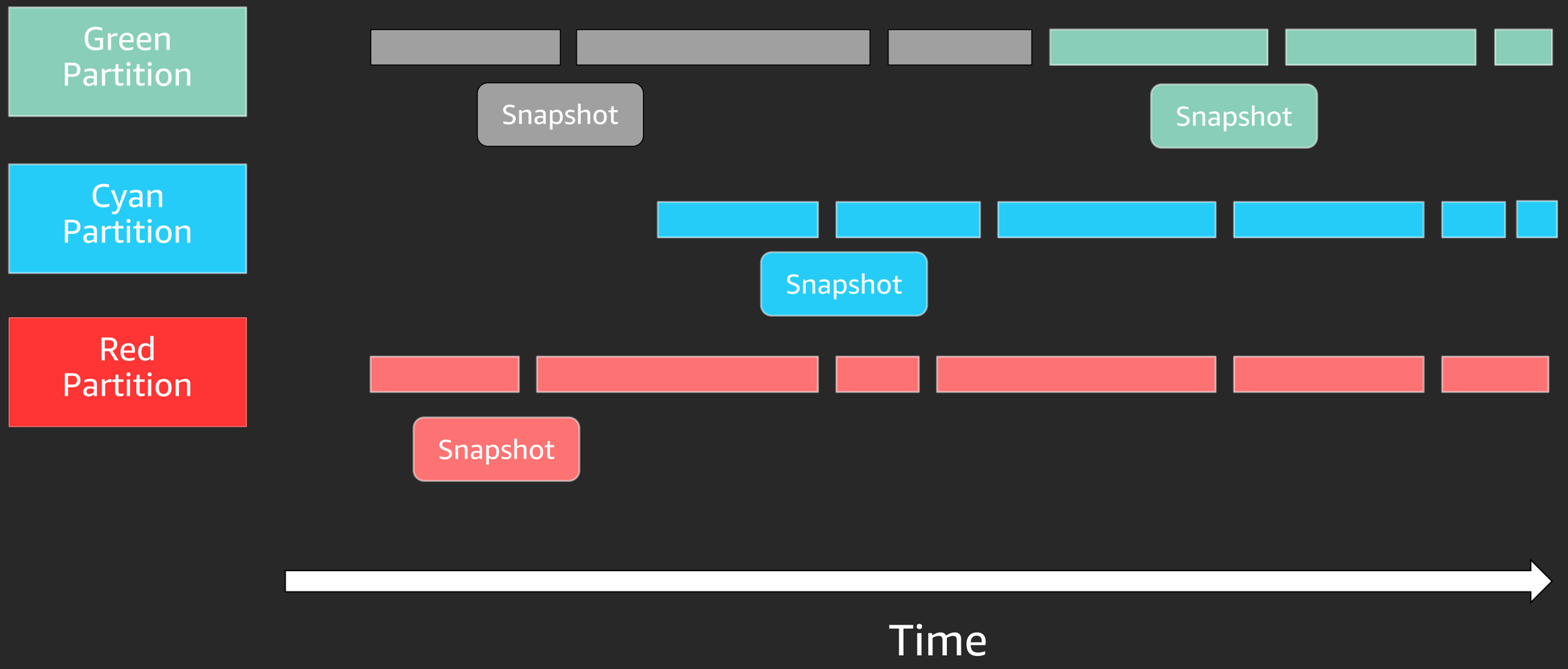
Without PITR enabled



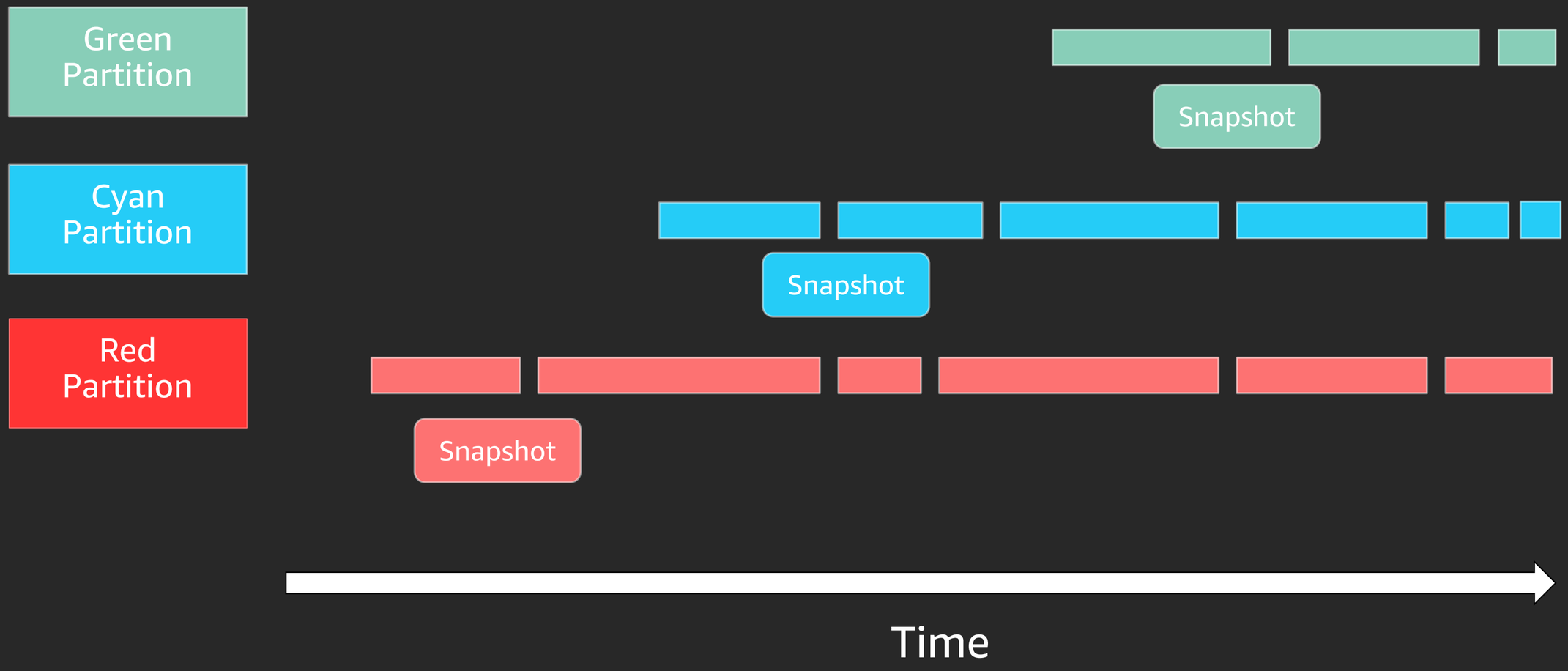
Without PITR enabled



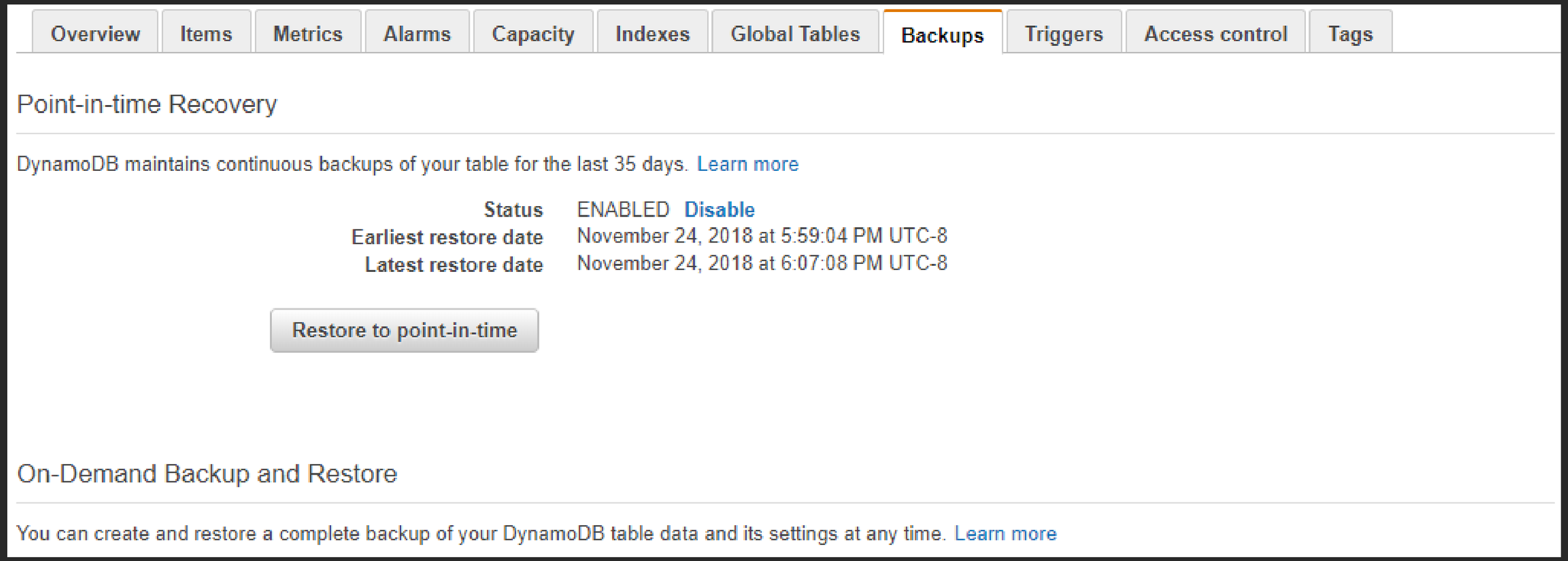
Without PITR enabled



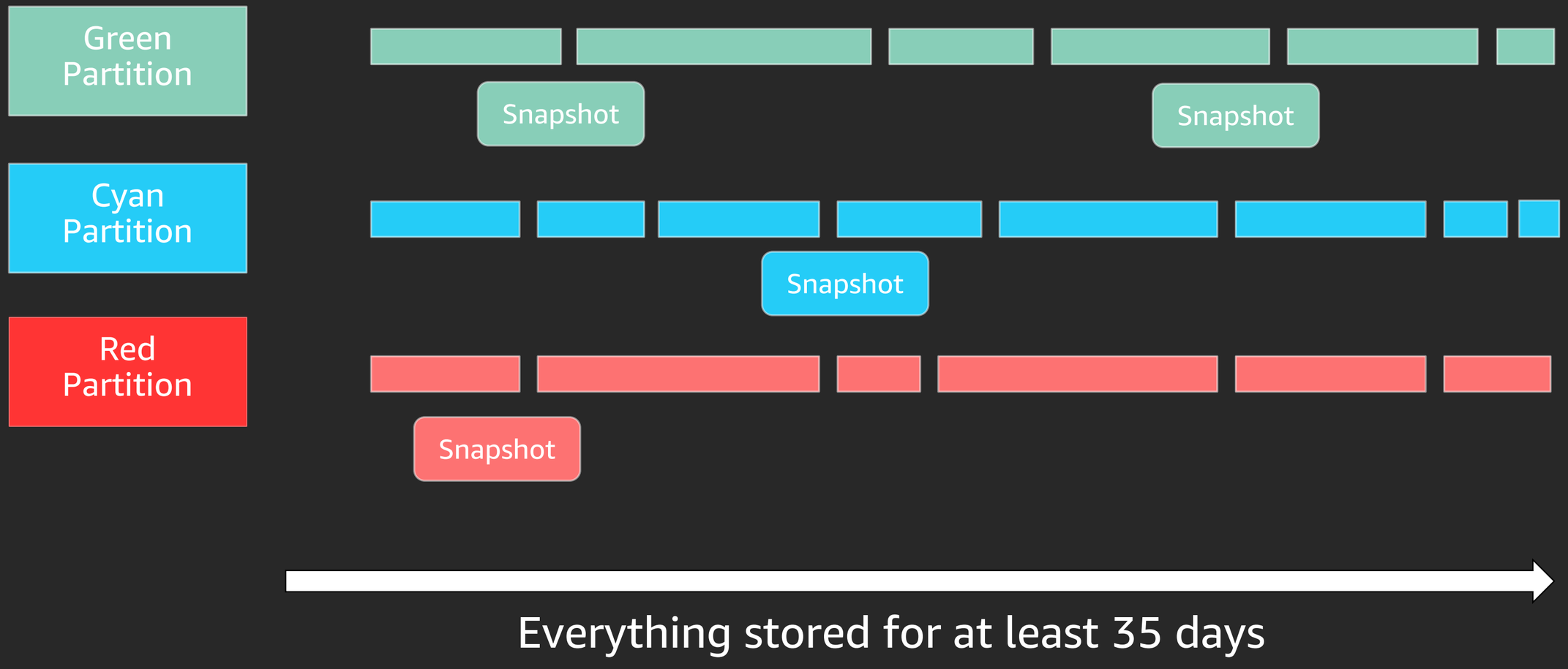
Without PITR enabled



With PITR enabled



With PITR enabled



Questions

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

James Christopher Sorenson III aka jaso

jaso@amazon.com

AWS re:Invent

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

aws

Please complete the session survey in the mobile app.

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.