AWS re:Invent

NOV. 29 – DEC. 3, 2021 | LAS VEGAS, NV

DAT327

Build a serverless, event-driven architecture with AWS services

Lucas Rettenmeier
Associate Solutions Architect
AWS

Sean Shriver
Sr. DynamoDB Specialist SA
AWS

Kirill Bogdanov
Sr. Solutions Architect
AWS

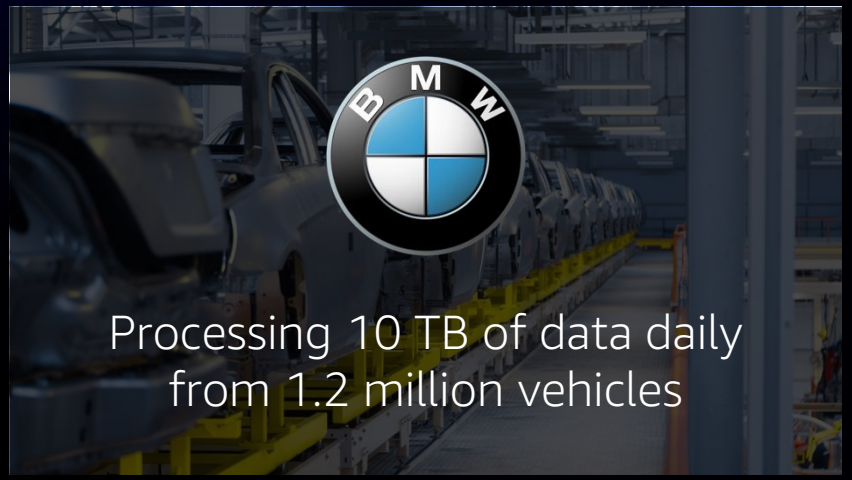
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Agenda

- Introduction to large-scale data processing
- Challenges of reliable data processing
- AWS approach with Amazon DynamoDB, AWS Lambda, and Amazon Kinesis Data Streams
- Workshop containing two labs
- Summary and conclusion

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Large-scale data processing









© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

How do customers process large volumes of data?

- No single server is sufficient to handle the data volumes
- There is value in processing data quickly to react to new events
- Mission critical systems need to be highly available

Distribute load and risk among multiple servers!

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

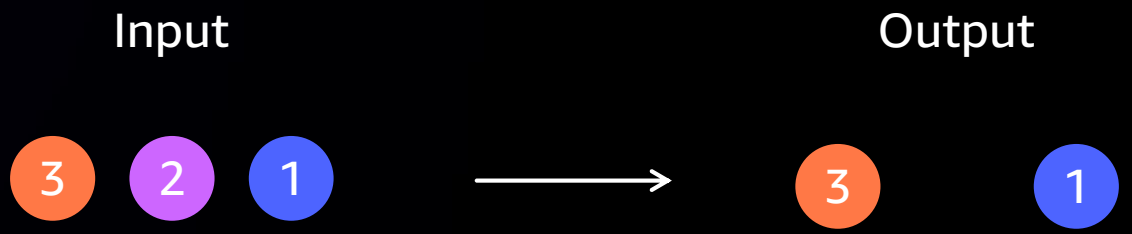
“A distributed system is one in which the failure of a computer you didn’t even know existed can render your own computer unusable.”

Leslie Lamport

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Delivery semantics

- **At most once** semantics
  - Messages might be lost



- **At least once** semantics
  - Messages might be duplicated

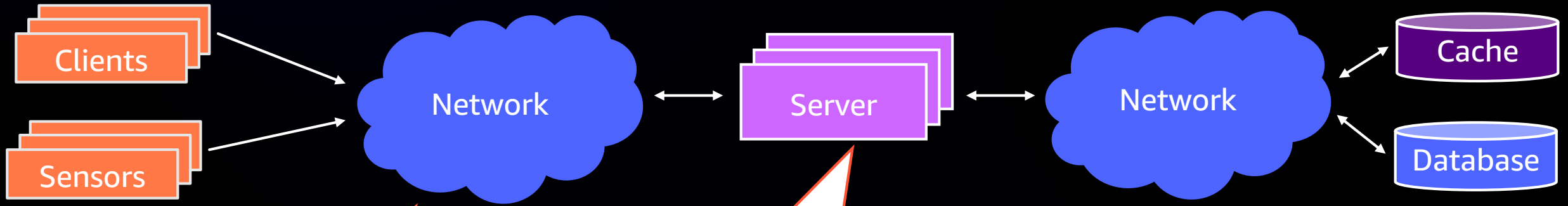


- **Exactly once** semantics
  - No loss, no duplication



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Challenges of building reliable distributed systems



- Arbitrary delays
- Client retries → Duplicate messages

+

- Hardware failures → Lost application level messages
- Restarts → Re-processing old messages

=

Inconsistent state

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Building reliable and scalable distributed systems on AWS



Amazon Kinesis



AWS Lambda



Amazon DynamoDB

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon DynamoDB



Performance at scale

Consistent, single-digit-millisecond response times at any scale; build applications with virtually unlimited throughput



Serverless architecture

No hardware provisioning, software patching, or upgrades; scales up or down auto-matically; contin-uously backs up your data



Enterprise security

Encrypts all data by default and fully integrates with AWS Identity and Access Management (IAM) for robust security



Global replication

Build global applications with fast access to local data by easily replicating tables across multiple AWS Regions

aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

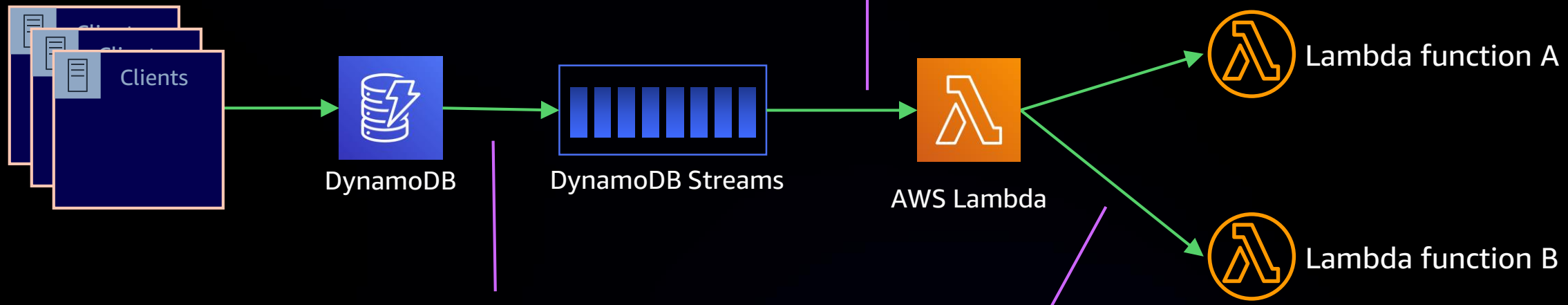
DynamoDB customers around the world



aws

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

DynamoDB Streams



Lambda service polls the stream for new records 4x per second

At least 2 Lambda functions can be subscribed to a single DynamoDB stream

All mutating actions to a DynamoDB table are captured as an item on the stream

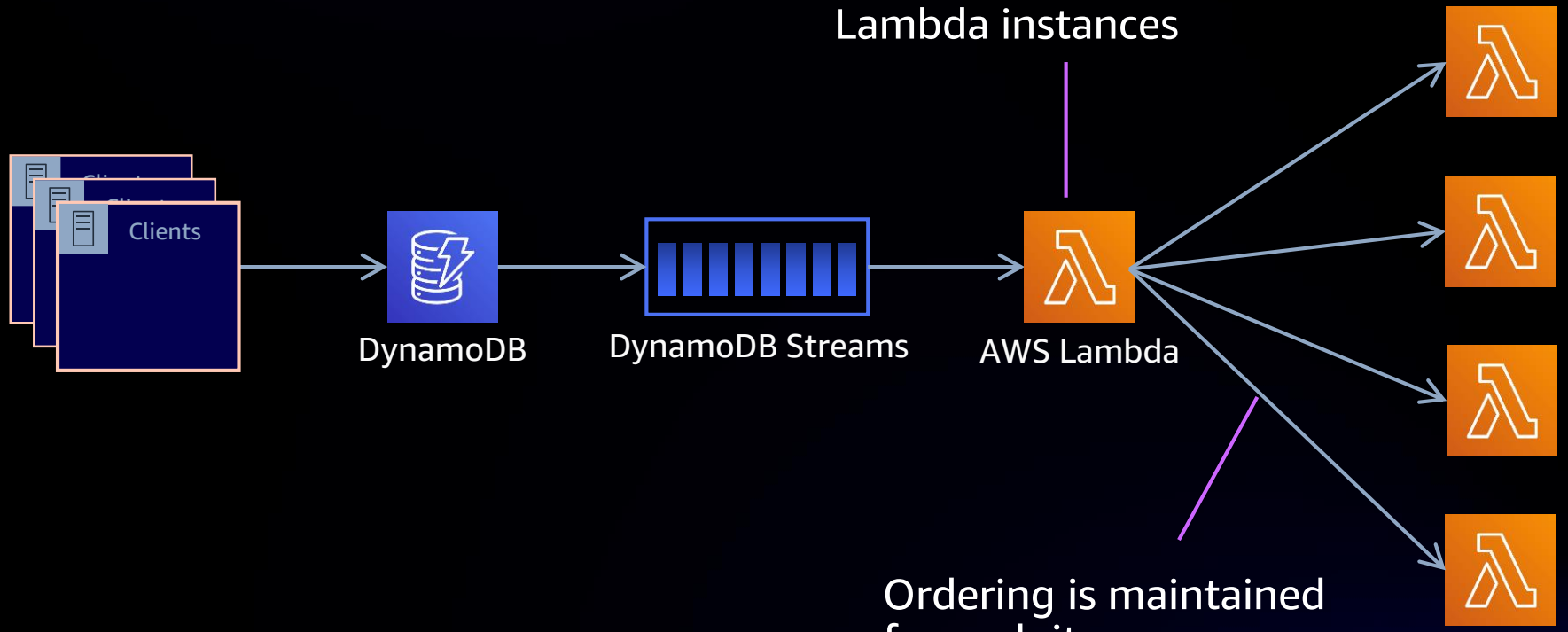
When new records are available, the subscribed Lambda functions are invoked with new records

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

DynamoDB Streams fan-out

The number of Lambda instances is influenced by the Lambda event source mapping and the number of DynamoDB table partitions*

Lambda service parcels out stream records to Lambda instances



Ordering is maintained for each item

* Reserved concurrency does not affect the number of Lambda instances

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Workshop

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

What to expect

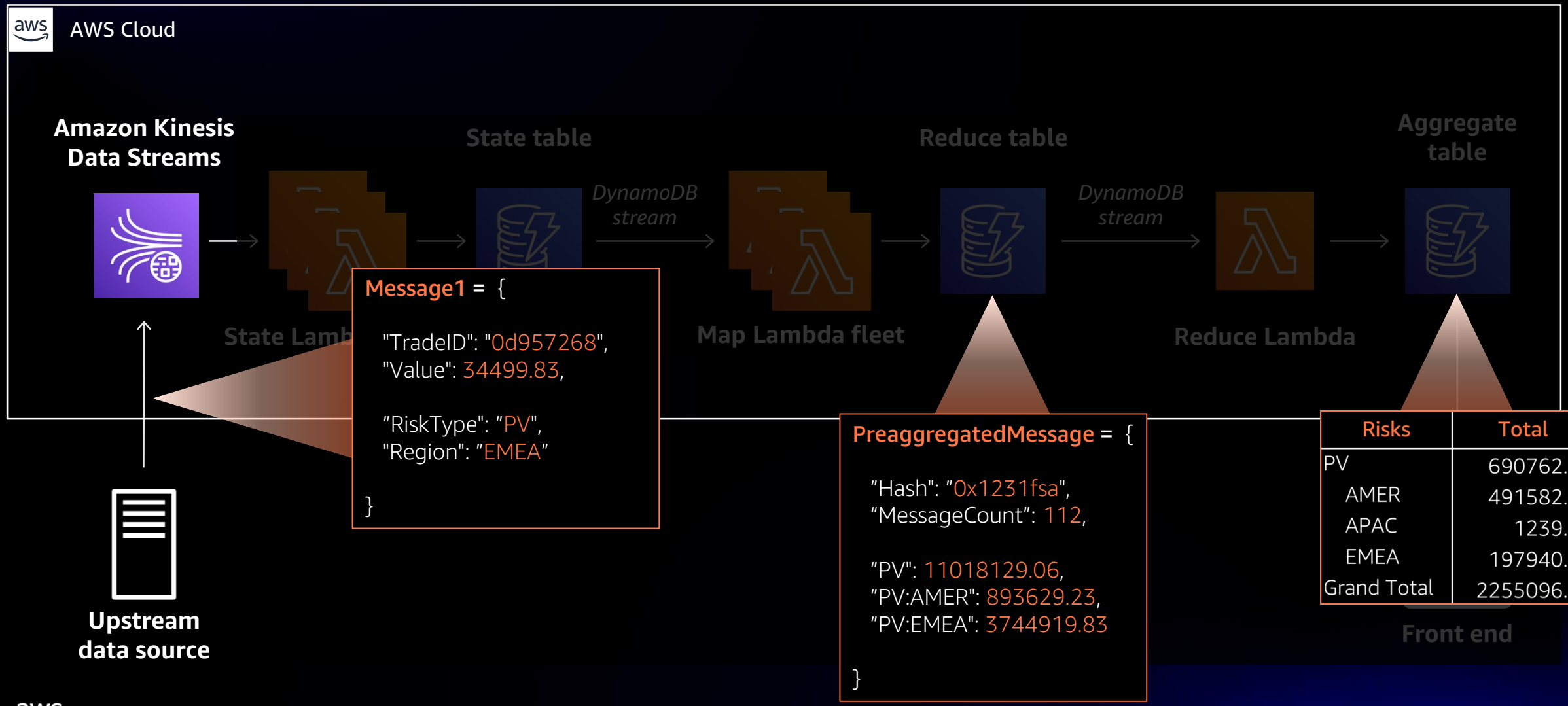
In this workshop you will be presented with a **serverless event-driven data aggregation pipeline**

This pipeline is broken! Moreover, input stream contains duplicates and Lambda functions are randomly failing on purpose!

Over the course of two labs you will have to **first** correctly connect all the elements of the pipeline and **then** update the Lambda functions to avoid message loss or duplication under failures

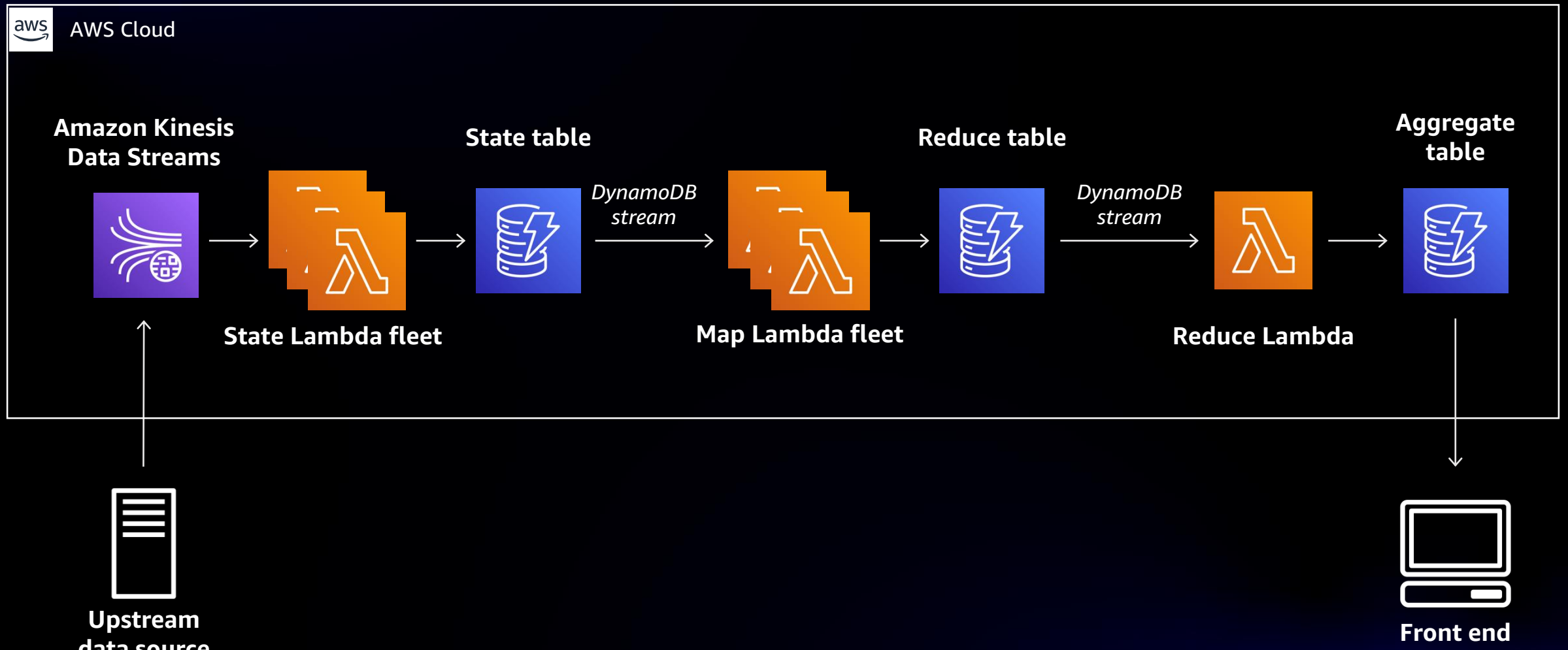
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pipeline architecture



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Pipeline architecture



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Example

Message1 = {

"TradeID": "0d957268",
"Value": 34499.83,

"RiskType": "PV",
"Region": "EMEA"

}

Message2 = {

"TradeID": "k38sl0a33",
"Value": 83629.23,

"RiskType": "PV",
"Region": "AMER"

}

PreaggregatedMessage = {

"Hash": "0x1231fsa",
"MessageCount": 2,

"PV": 118129.06,
"PV:AMER": 83629.23,
"PV:EMEA": 34499.83

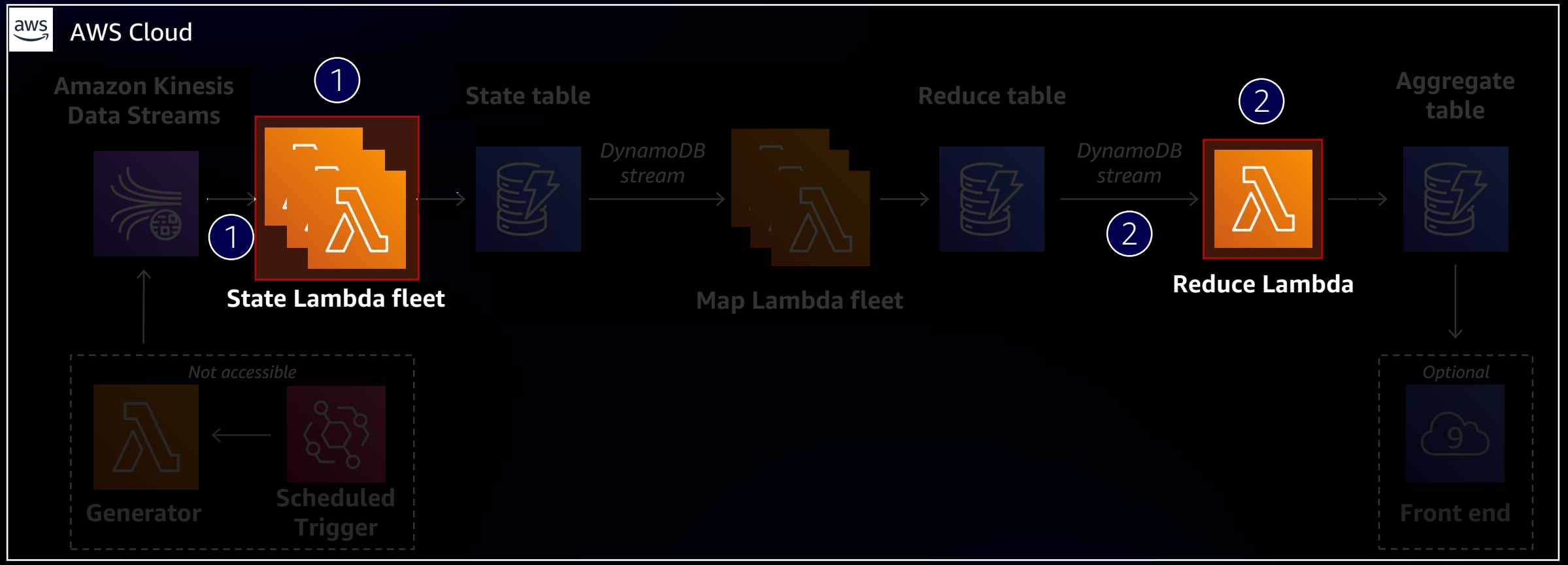
}

Risks	Total
Delta	1564334.25
AMER	1479918.76
APAC	1551.38
EMEA	82864.11
PV	690762.46
AMER	491582.25
APAC	1239.76
EMEA	197940.45
Grand Total	2255096.71

State table → Reduce table → Aggregate table

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Workshop architecture



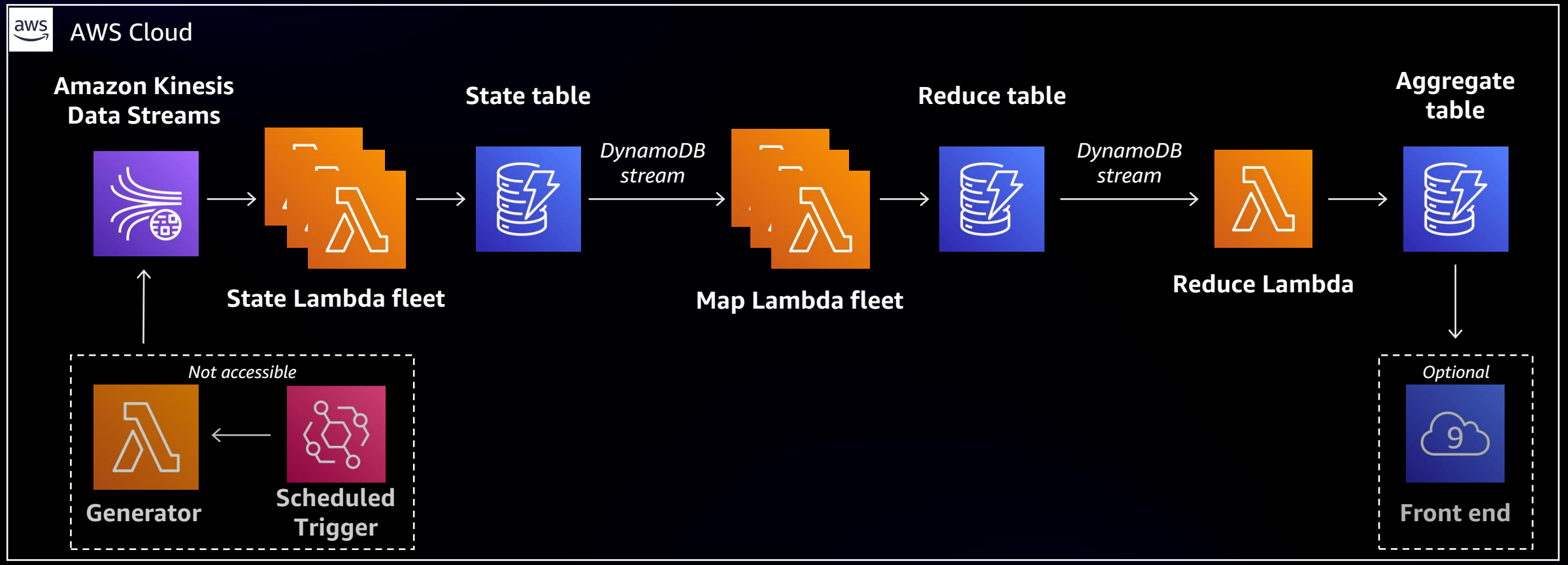
Lab 1: Connect the pipeline

Collect 300 points

Lab 2: Fault-proof the pipeline against manually injected chaos

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Let‘s build it!



Lab 1: Connect the pipeline

Collect 300 points

Lab 2: Fault-proof the pipeline against manually injected chaos

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Key takeaways

- Customers across many industries face the problem of large-scale, real-time data processing
- In many cases, no single server is sufficient to handle the data volumes
- Building scalable and reliable distributed systems is difficult!
- Serverless AWS services such as **Amazon DynamoDB**, **AWS Lambda**, and **Amazon Kinesis** provide building blocks to make it easy, efficient, and reliable

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

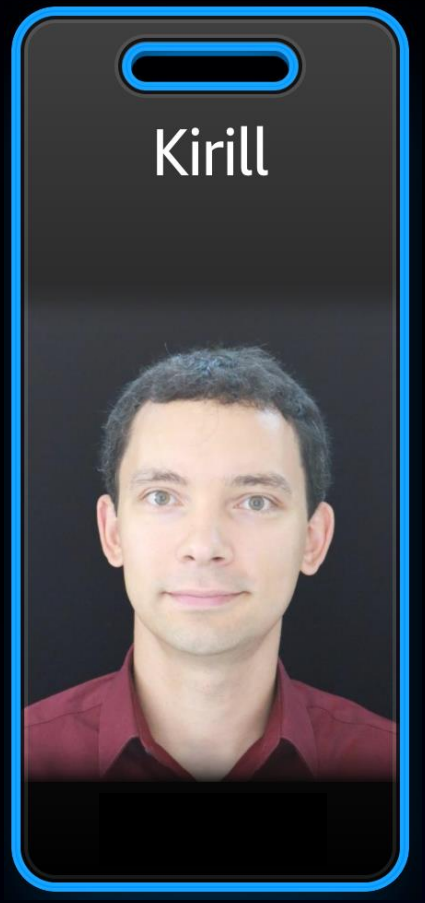
Get the code on GitHub!



https://git.io/JiTKg

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Special thanks to...



© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Thank you!

Lucas Rettenmeier
@lucasrettenmeier
gfs-acceleration-interest@amazon.com

Kirill Bogdanov
@kirillsc

Sean Shriver
@sean_shriver

© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.