AWS re:Invent

DAT402-R

Going deep on Amazon Aurora Serverless

Tony Hooper
Senior Development Manager
Aurora Development
Amazon Web Services

Rudi Leibbrandt
Principal Product Manager
Product Management
Amazon Web Services

AWS re:Invent

aws

Agenda

- Amazon Aurora fundamentals
- What is Amazon Aurora Serverless?
- How Aurora Serverless works
- Q&A

Amazon Aurora

Enterprise database at open-source price

Delivered as a managed service

Speed and availability of high-end commercial databases

Simplicity and cost effectiveness of open-source databases

Drop-in compatibility with MySQL and PostgreSQL

Simple pay-as-you-go pricing

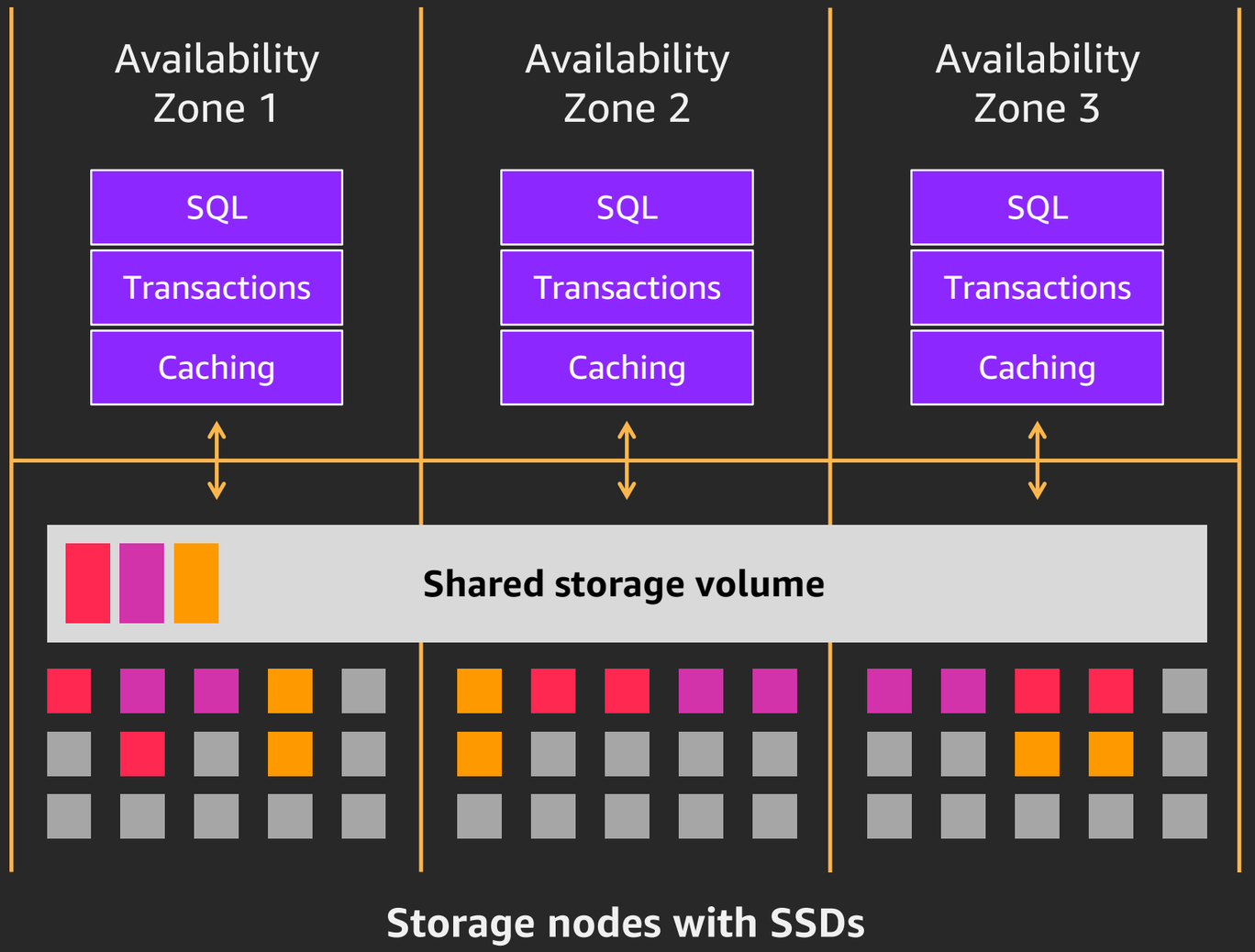
Aurora scale-out, distributed architecture

Purpose-built, log-structured, distributed storage system designed for databases

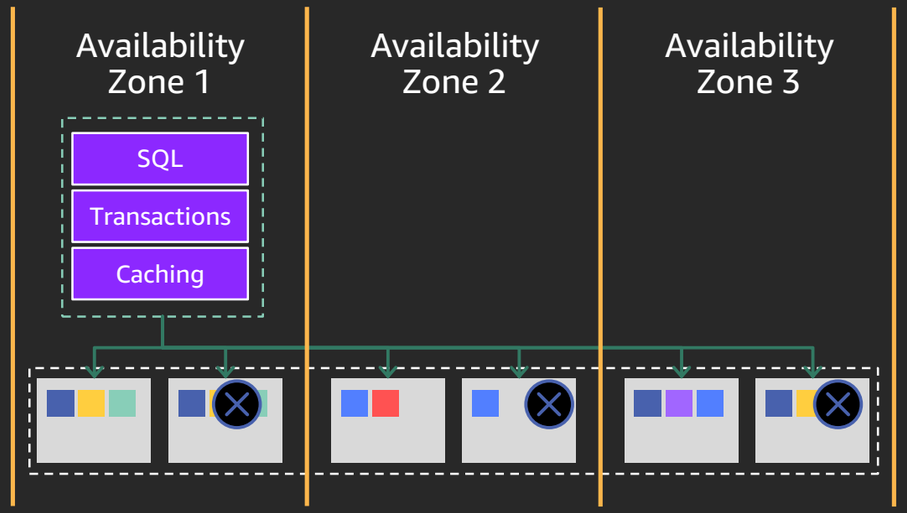
Storage volume is striped across hundreds of storage nodes distributed over 3 different Availability Zones

Six copies of data, two copies in each Availability Zone to protect against AZ+1 failures

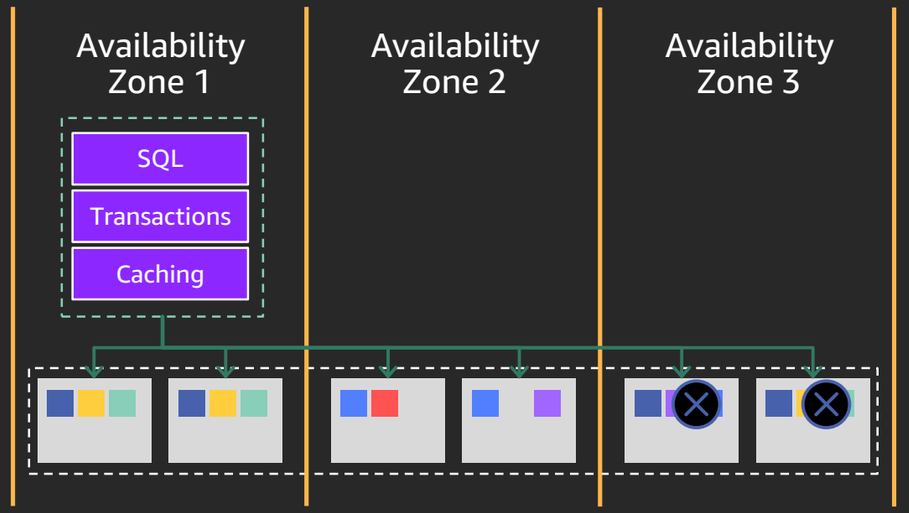
Data is written in 10GB “protection groups,” growing automatically up to 64TB



Six-way replicated storage



Read availability



Read and write availability

Data is written to all six nodes asynchronously, in parallel

Writes require a quorum of 4/6 nodes, and reads require 3/6 nodes

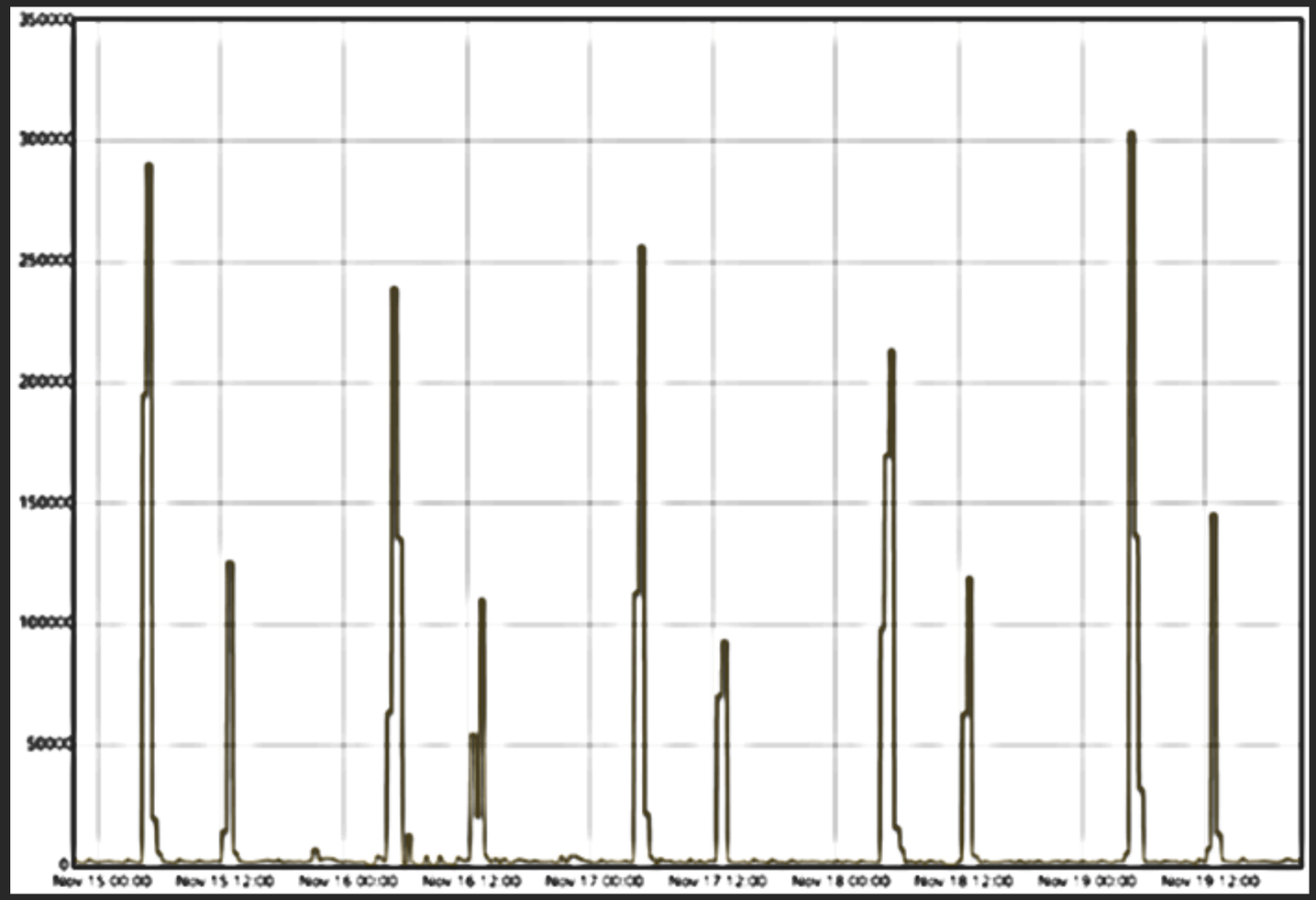
Peer-to-peer “gossip protocol” is used for repairs

What is Amazon Aurora Serverless?

Workloads in the wild

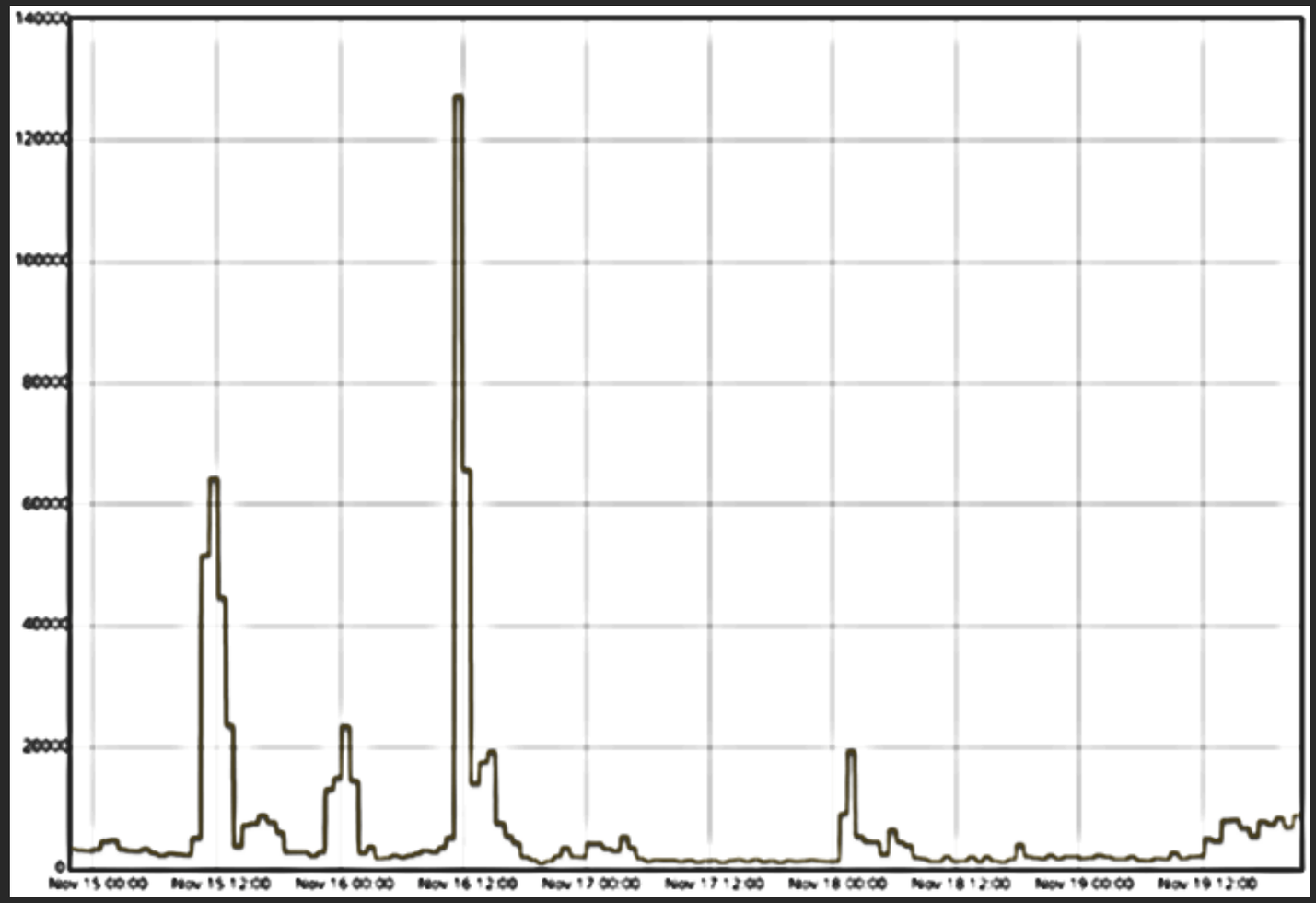
Example #1: Episodic dev-test workload

VolumeWriteIOPs, 120-hour window



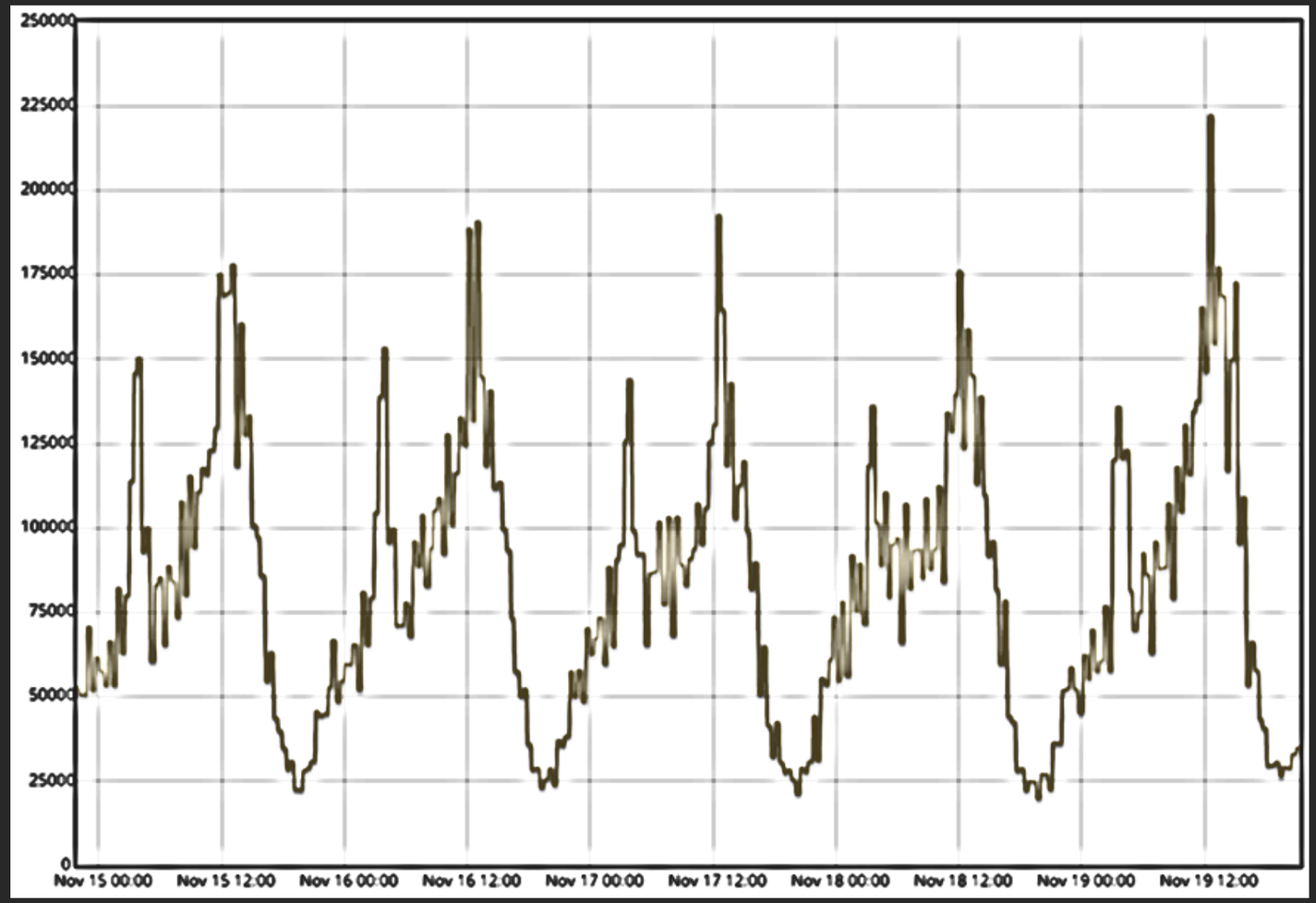
Example #2: Mostly idle dev-test workload

VolumeWriteIOPs, 120-hour window



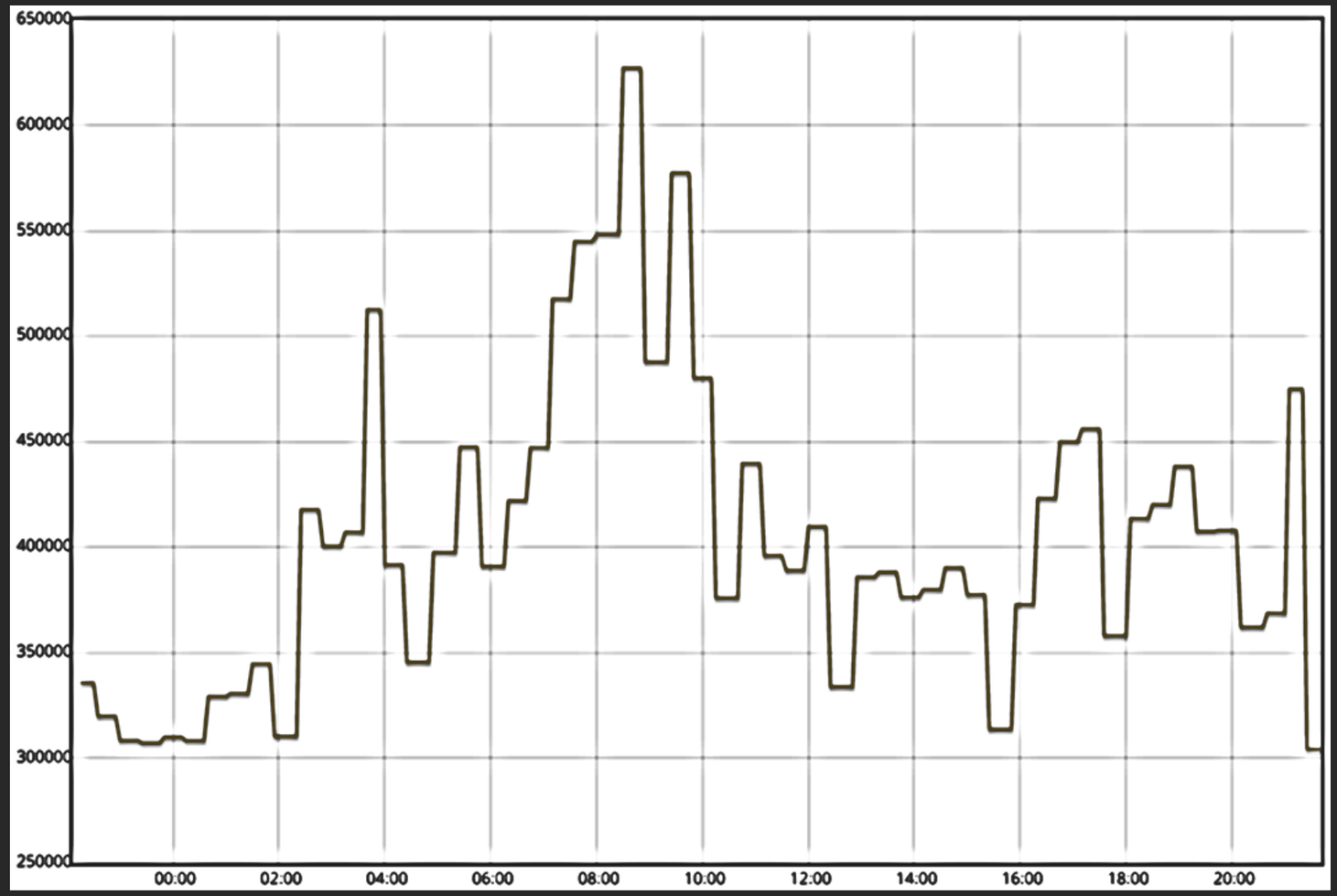
Example #3: Spiky gaming workload

VolumeWriteIOPs, 120-hour window

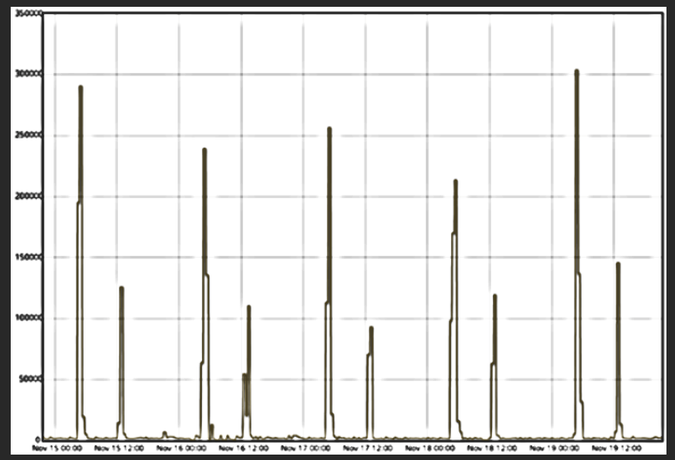


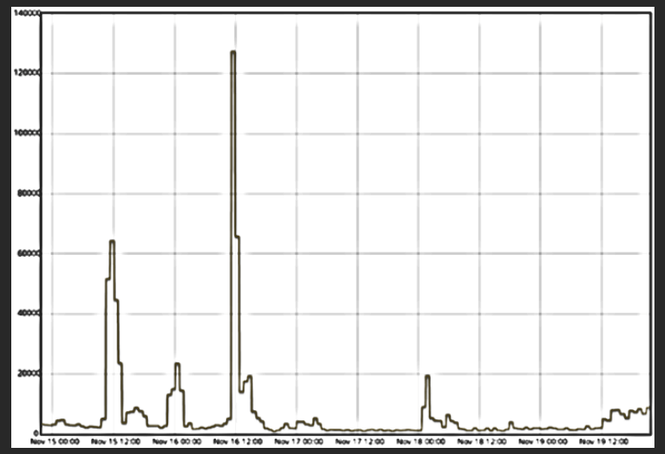
Example #4: E-commerce production workload

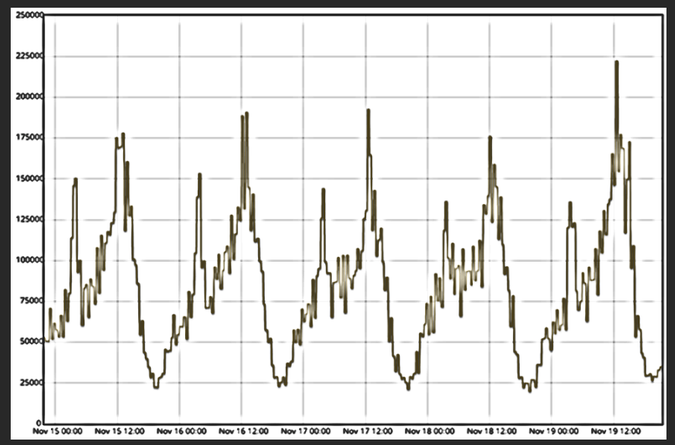
VolumeWriteIOPs, 24-hour window

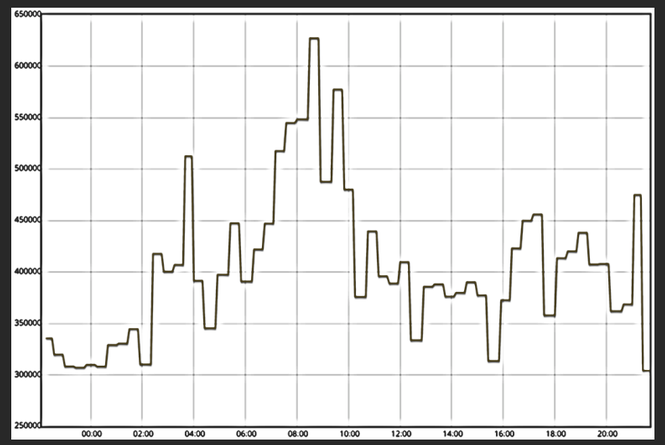


Decisions to make









- Episodic dev-test workload
- Mostly idle dev-test workload
- Spiky gaming workload
- E-commerce production workload

Provision for peak
versus
area under the curve

You have some choices to make

Provision for peak → Expensive

-or-

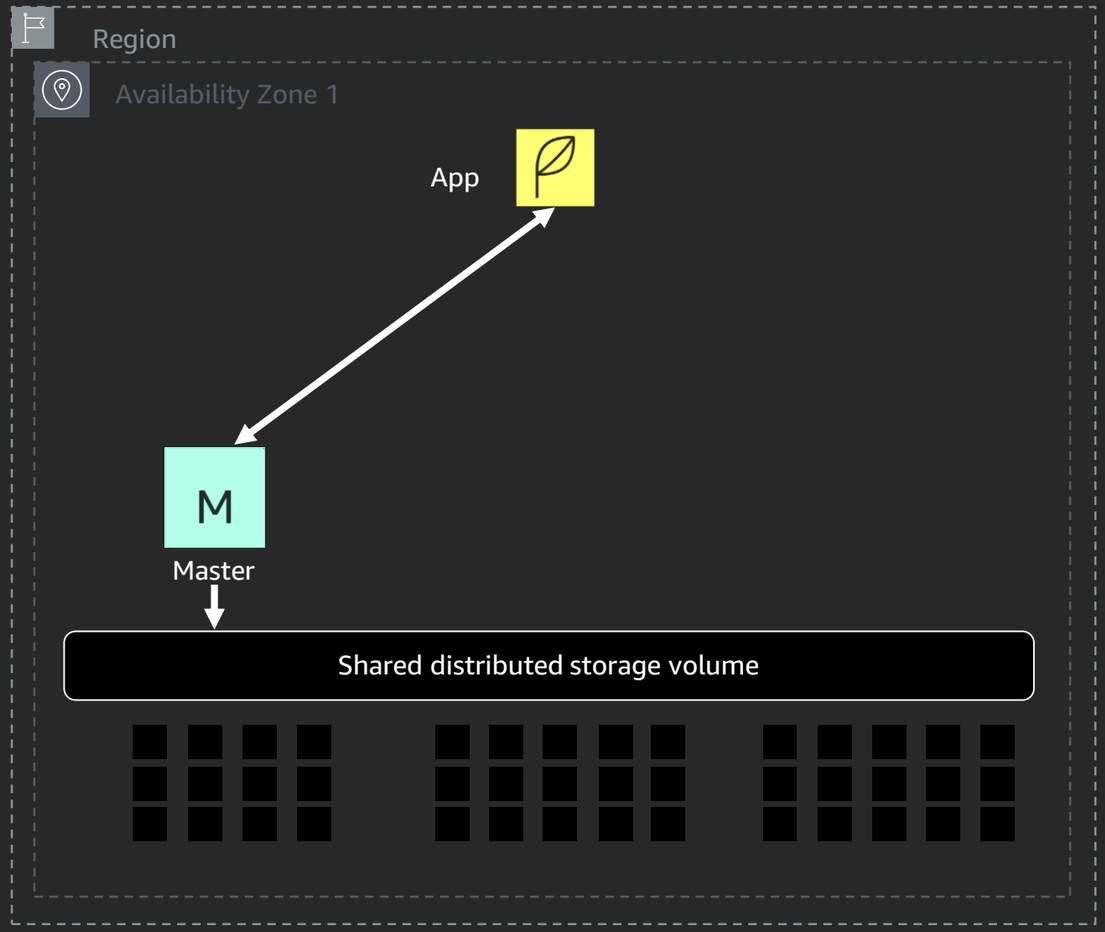
Provision less than peak → End-user (business) impact

-or-

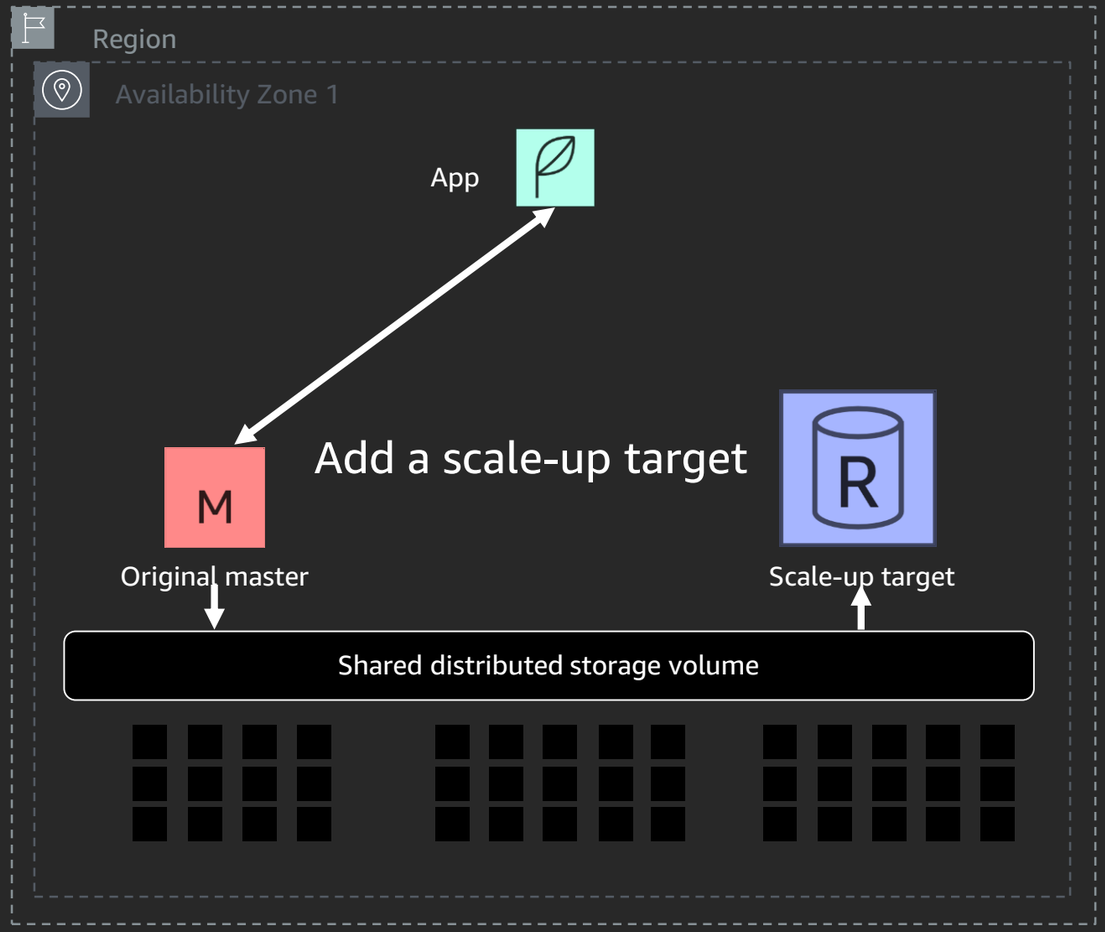
Continuously monitor and manually scale up/down → Hard

- Requires experts
- Risks outages

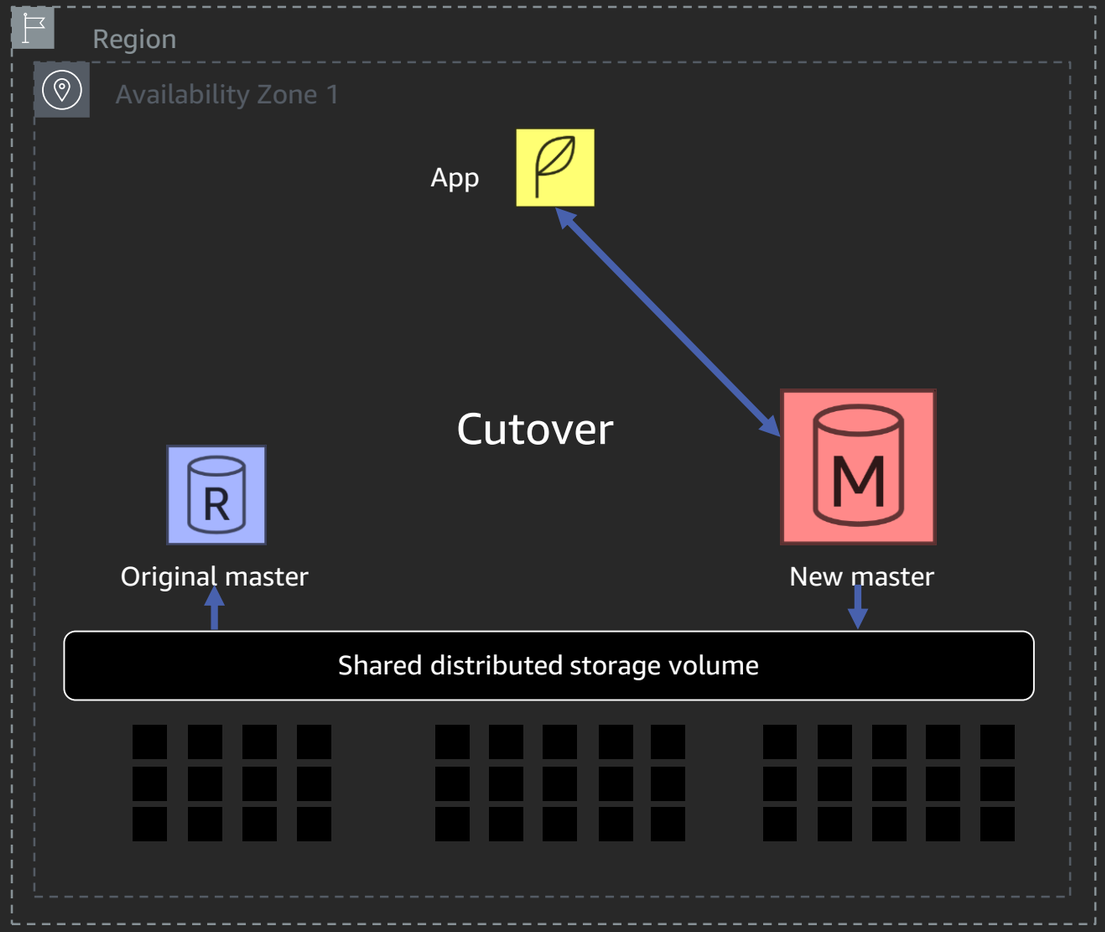
Database capacity management



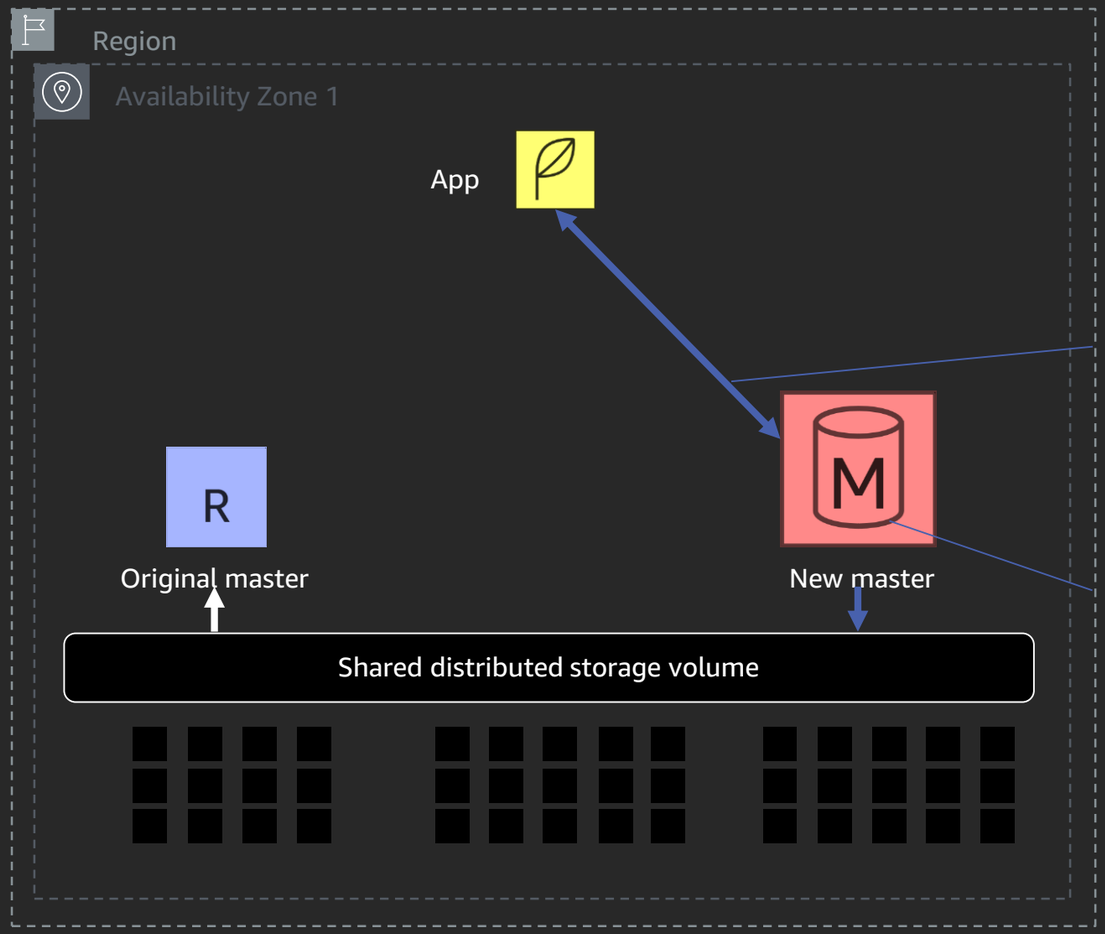
Database capacity management



Database capacity management



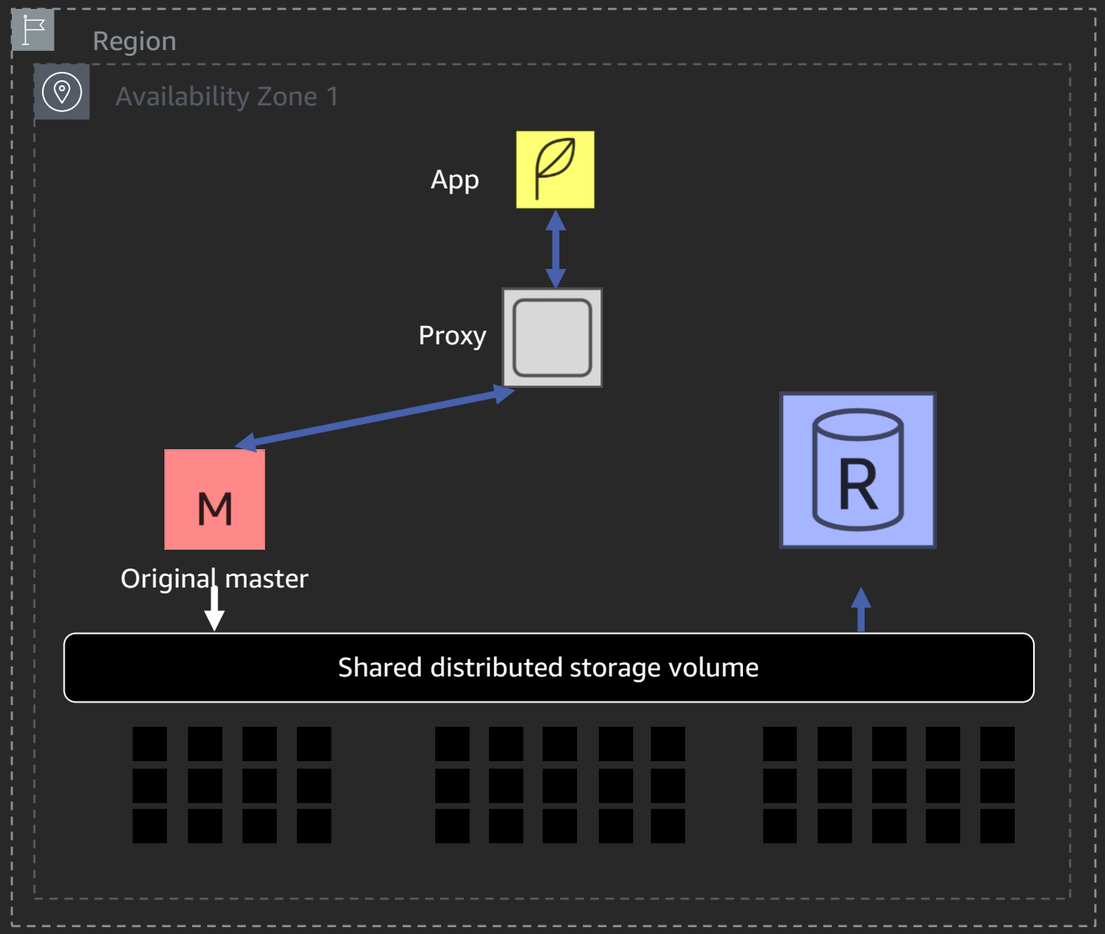
Database capacity management



Cutover can involve downtime

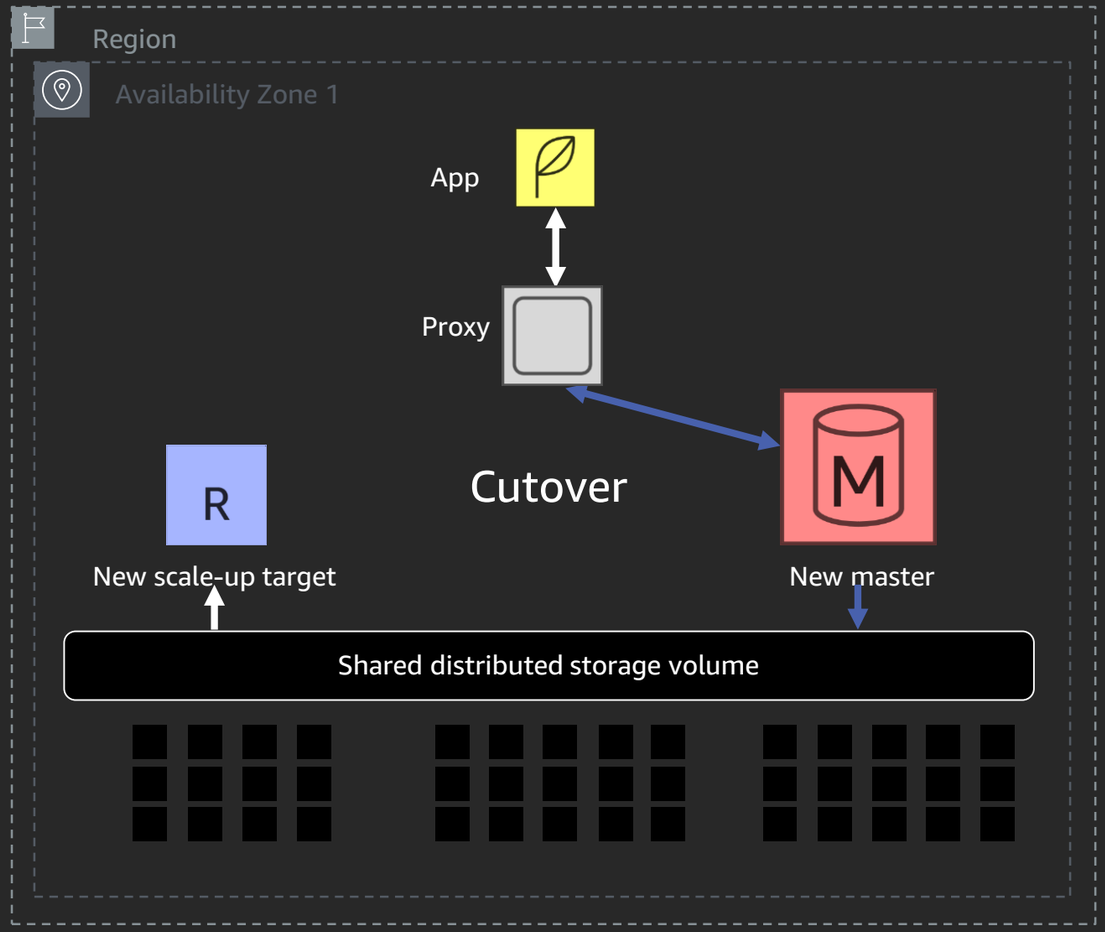
Cold buffer pool after cutover

Database capacity management

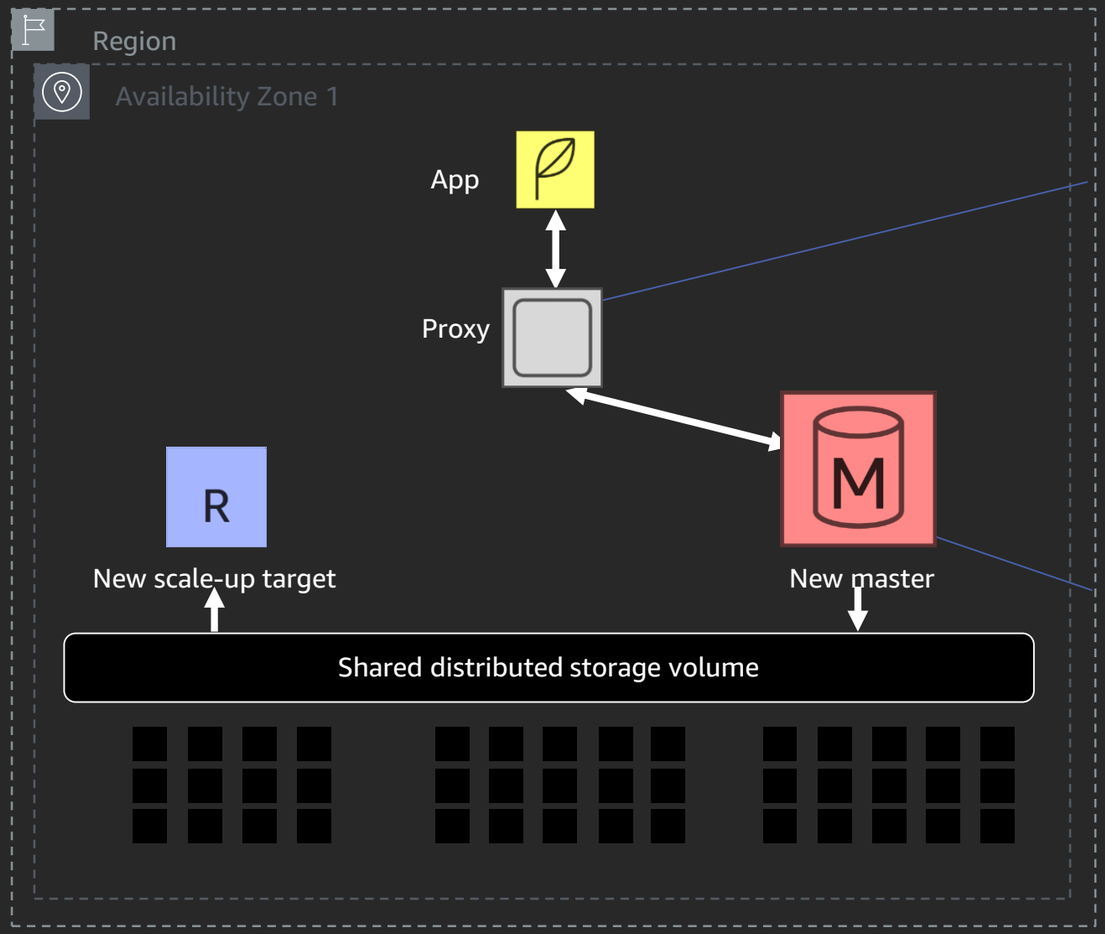


Add a proxy; shield the application from connection changes

Database capacity management



Database capacity management

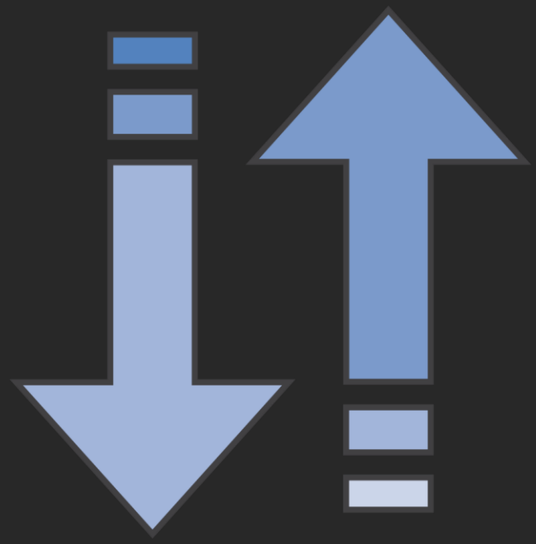


Single point of failure

Cold buffer pool after cutover

Aurora Serverless

- Responds to your application load automatically
- Designed for scale capacity with no downtime
- Multi-tenant proxy is highly available
- Scale target has warm buffer pool
- Shuts down when not in use



How Aurora Serverless works

Aurora Serverless

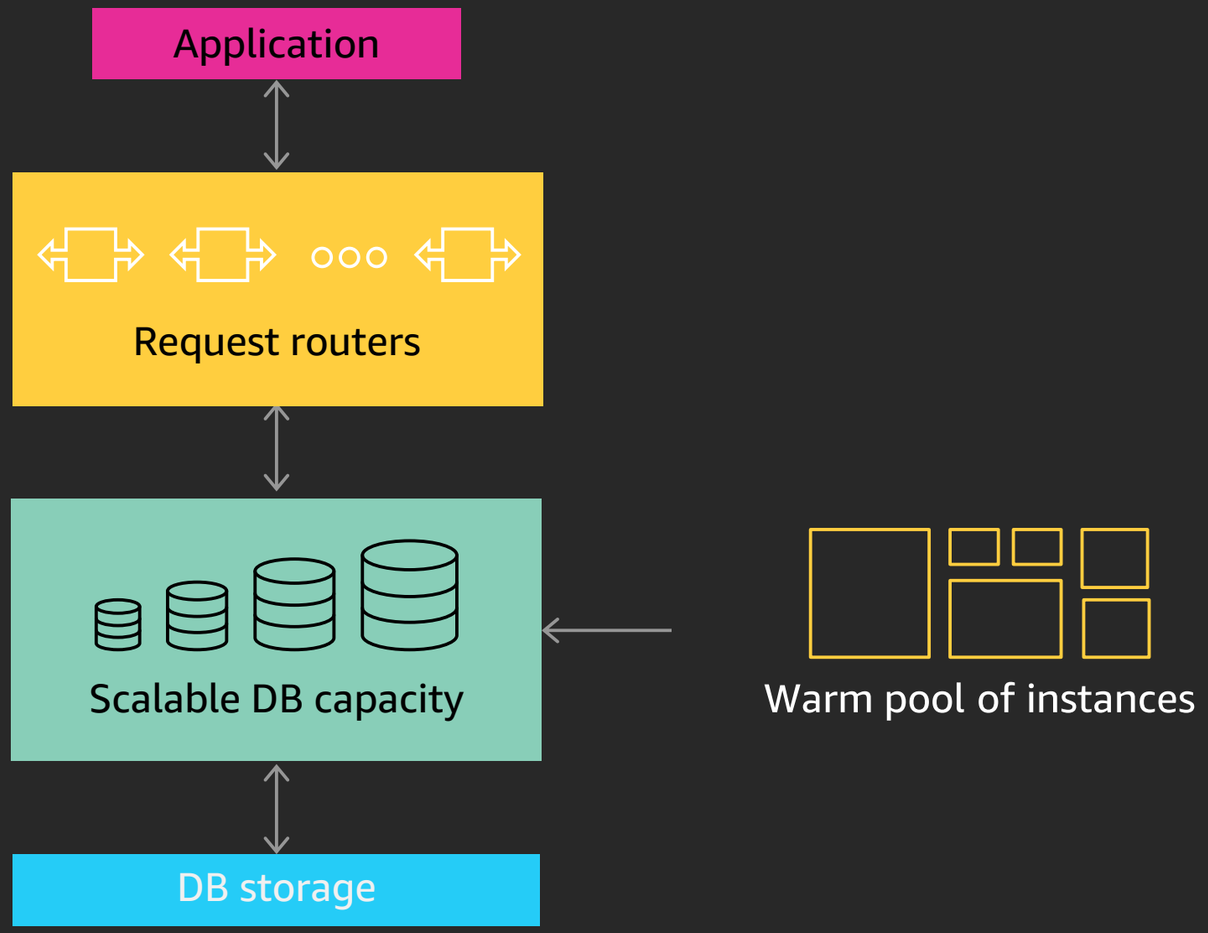
Starts up on demand;
shuts down when not in use

Scales up/down automatically

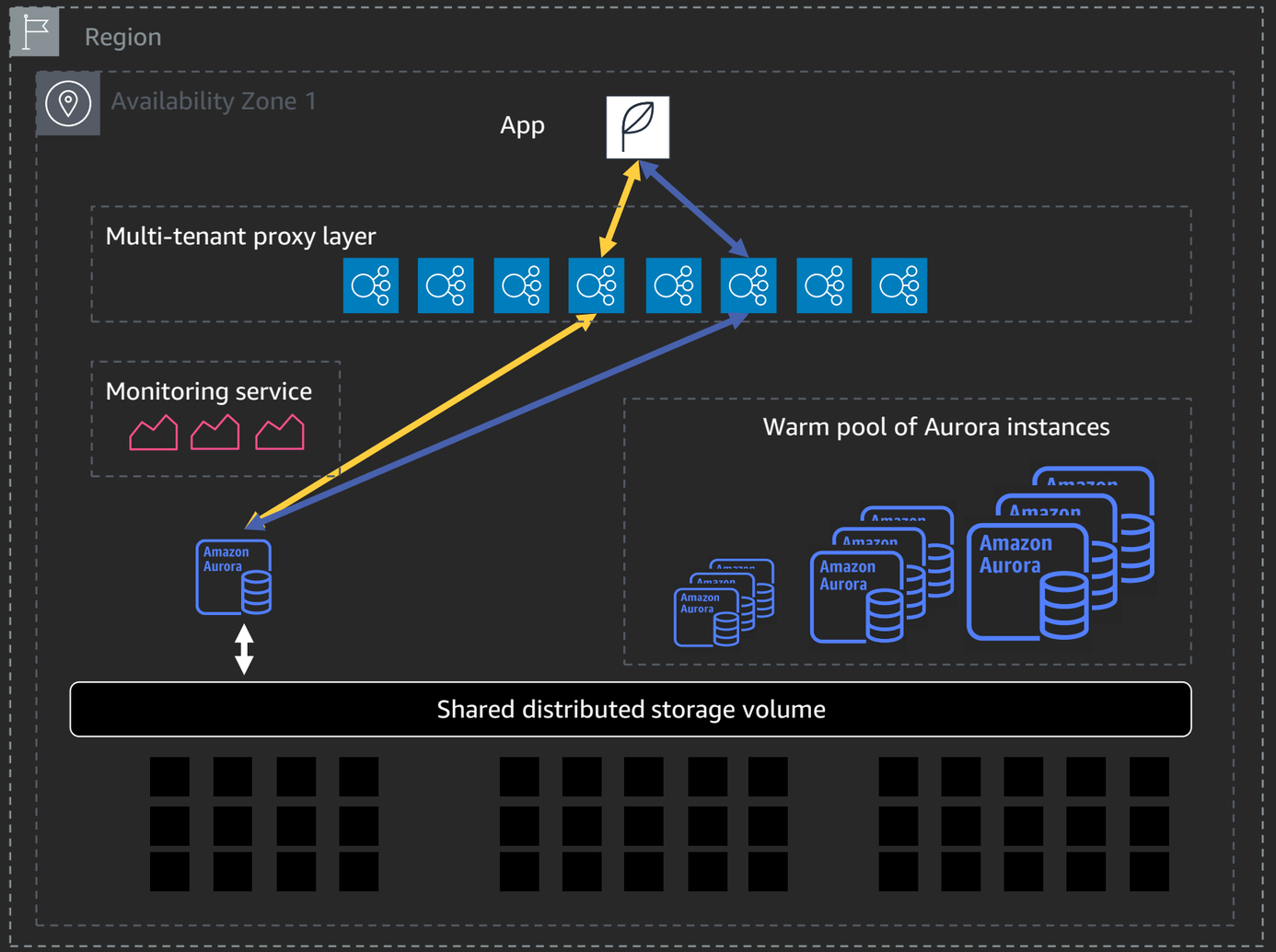
No application impact when scaling

Pay per second, one-minute minimum

Great for infrequently used,
unpredictable, or cyclical workloads

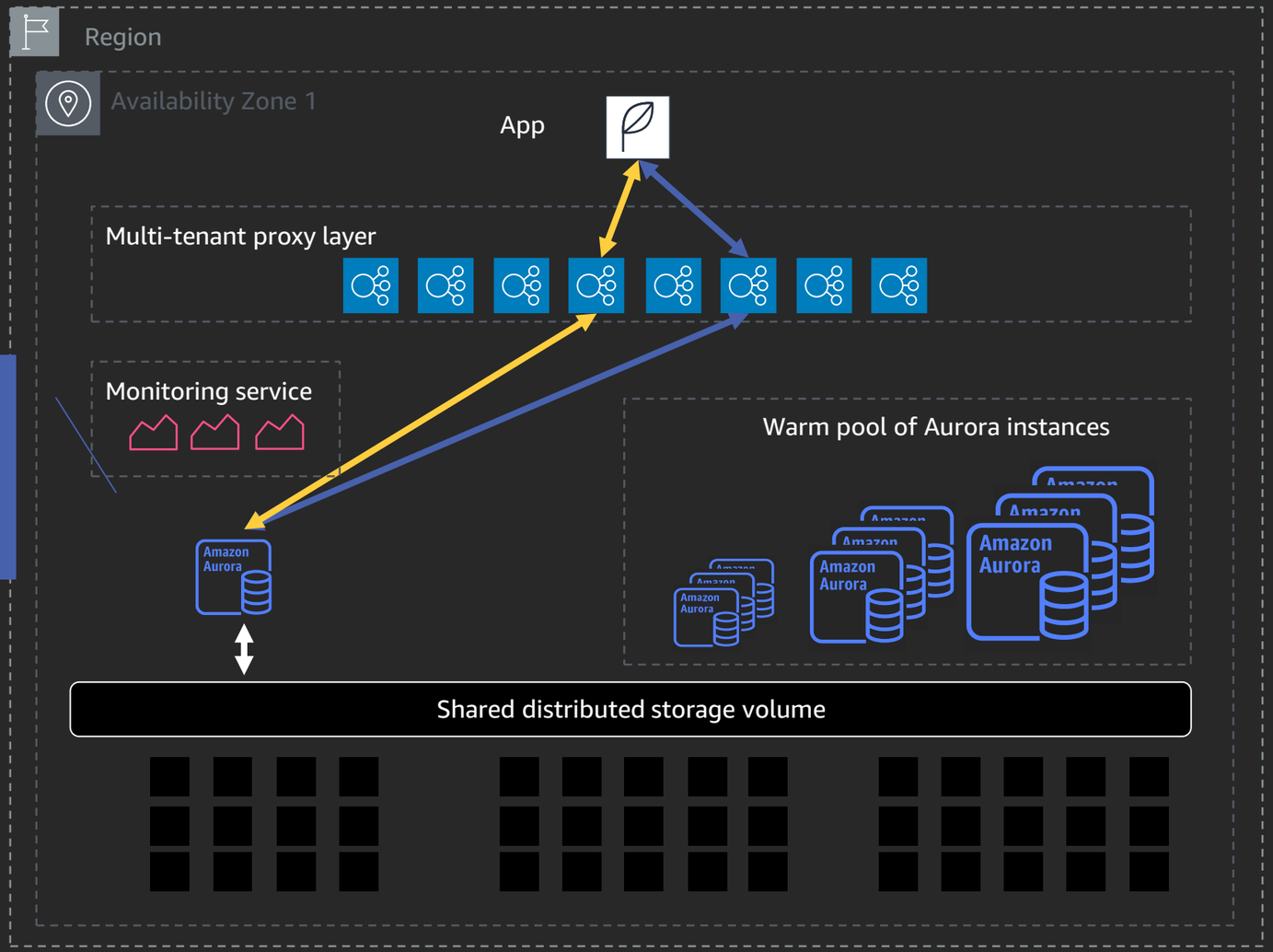


How does it work?



How does it work?

Monitors compute infrastructure for thresholds (CPU, memory, storage)

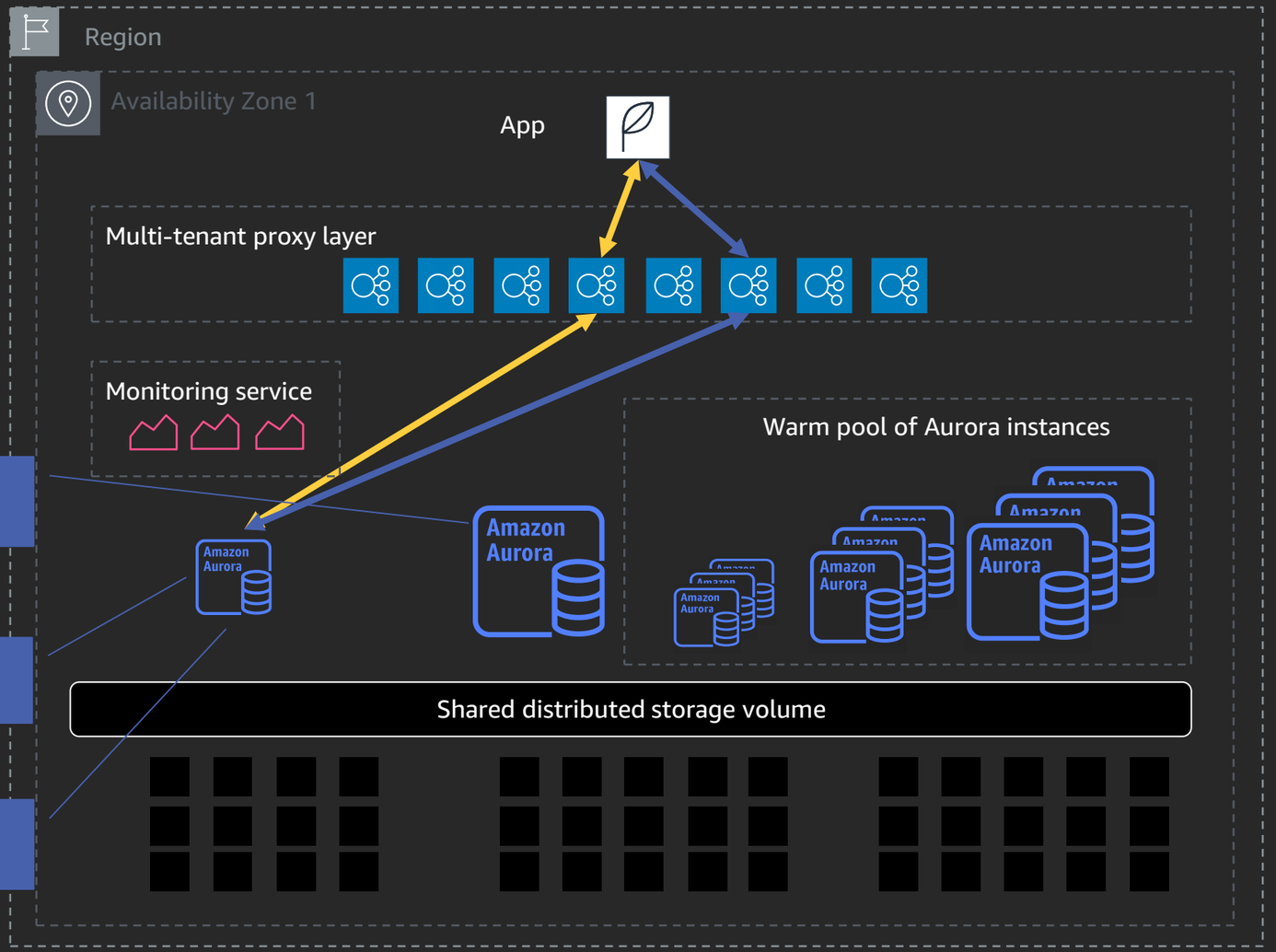


How does it work?

Get server from warm pool

Transfer buffer pool

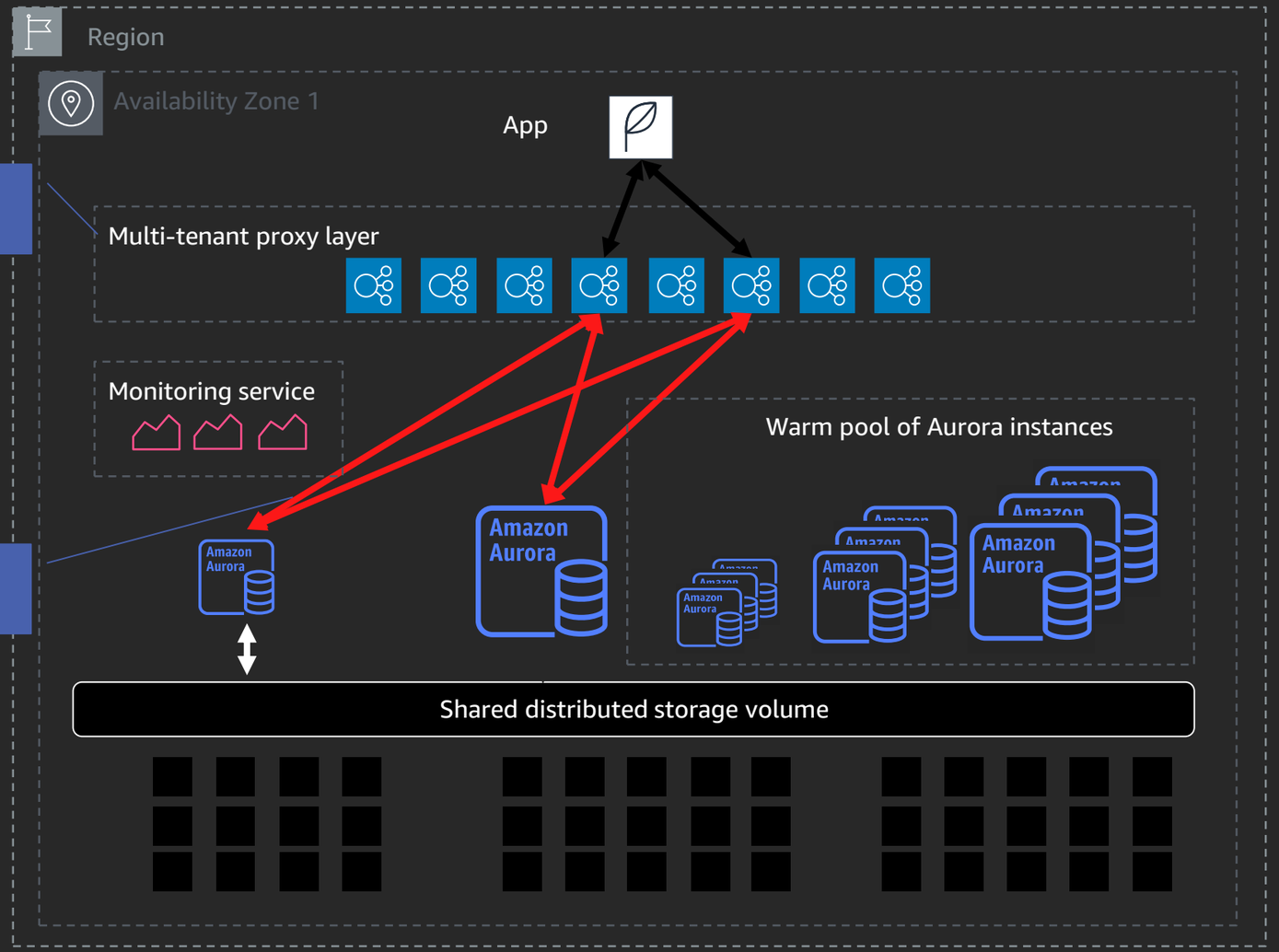
Look for safe scale point



How does it work?

Freeze workload

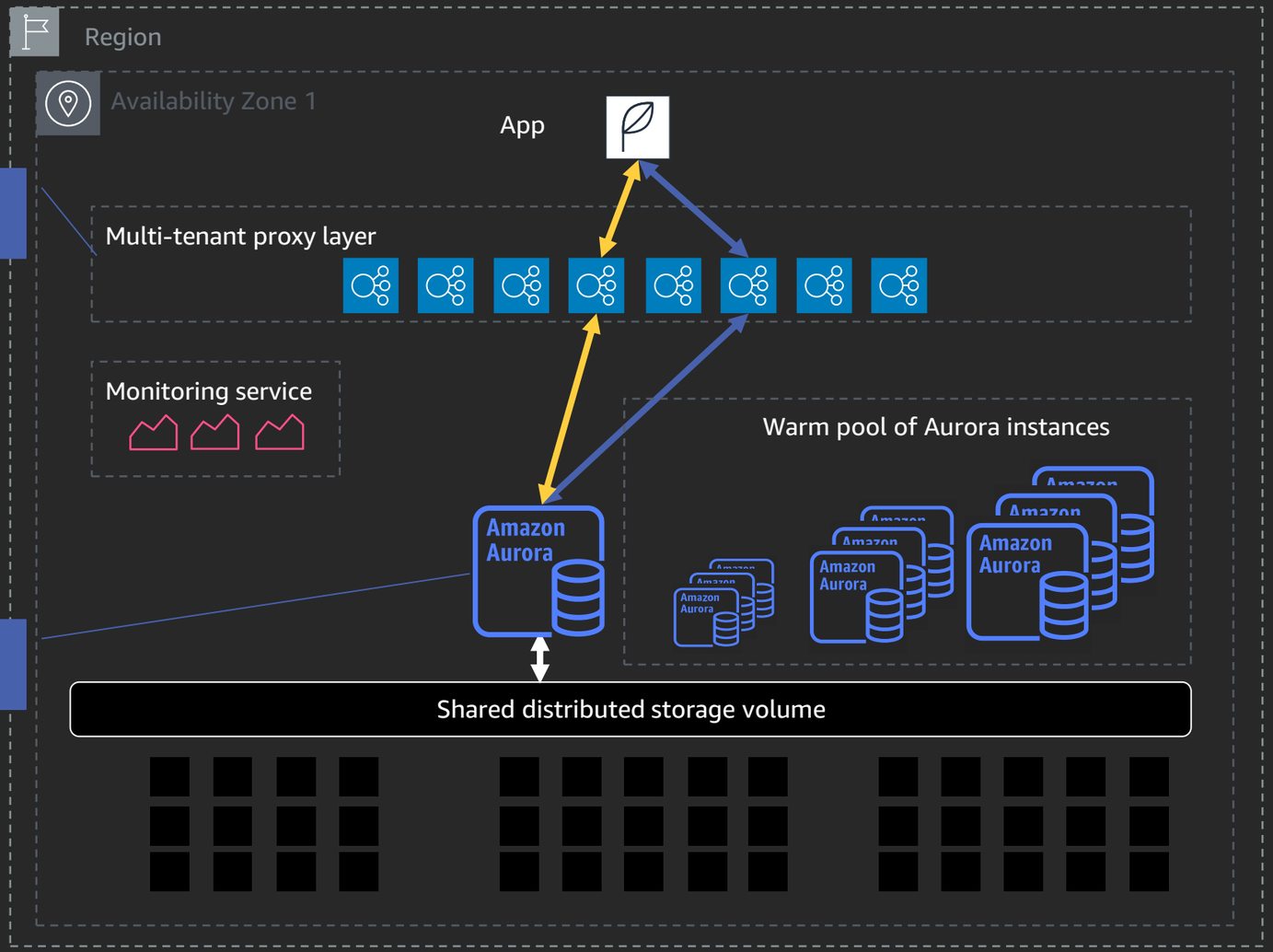
Transfer session state



How does it work?

Resume workload

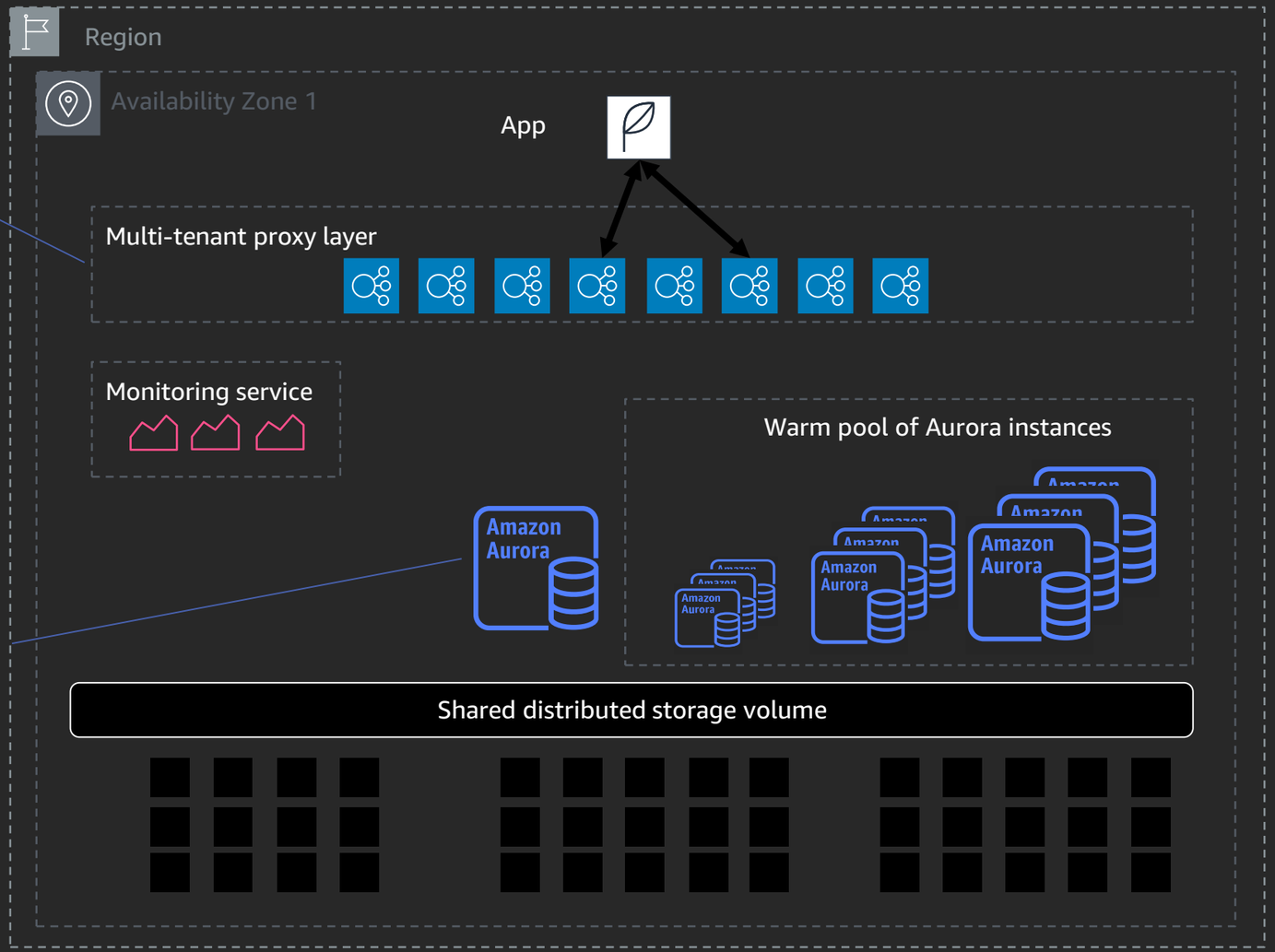
Warm buffer pool



How does it work?

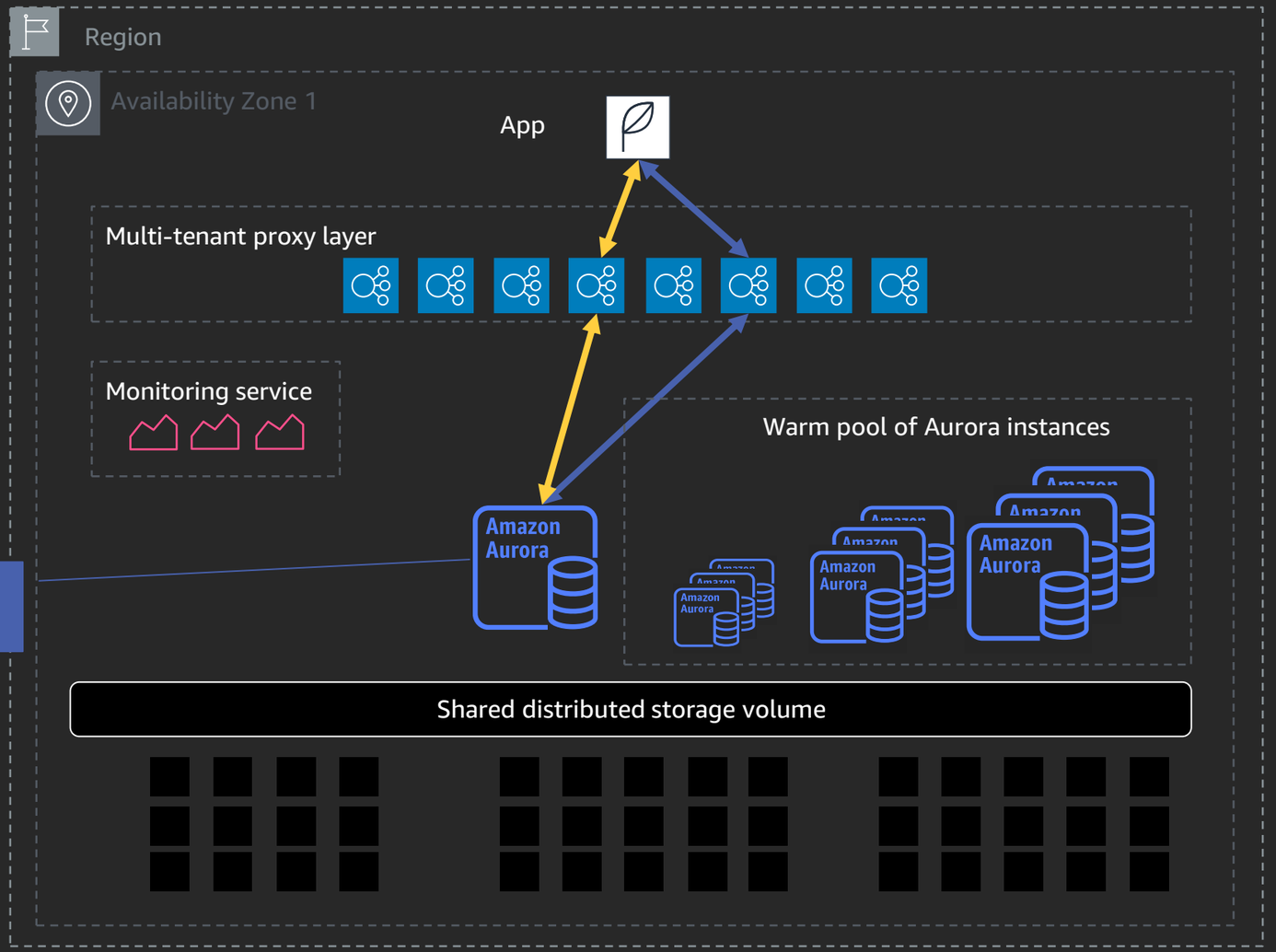
When workload is idle...

Scale down to minimum (or zero)

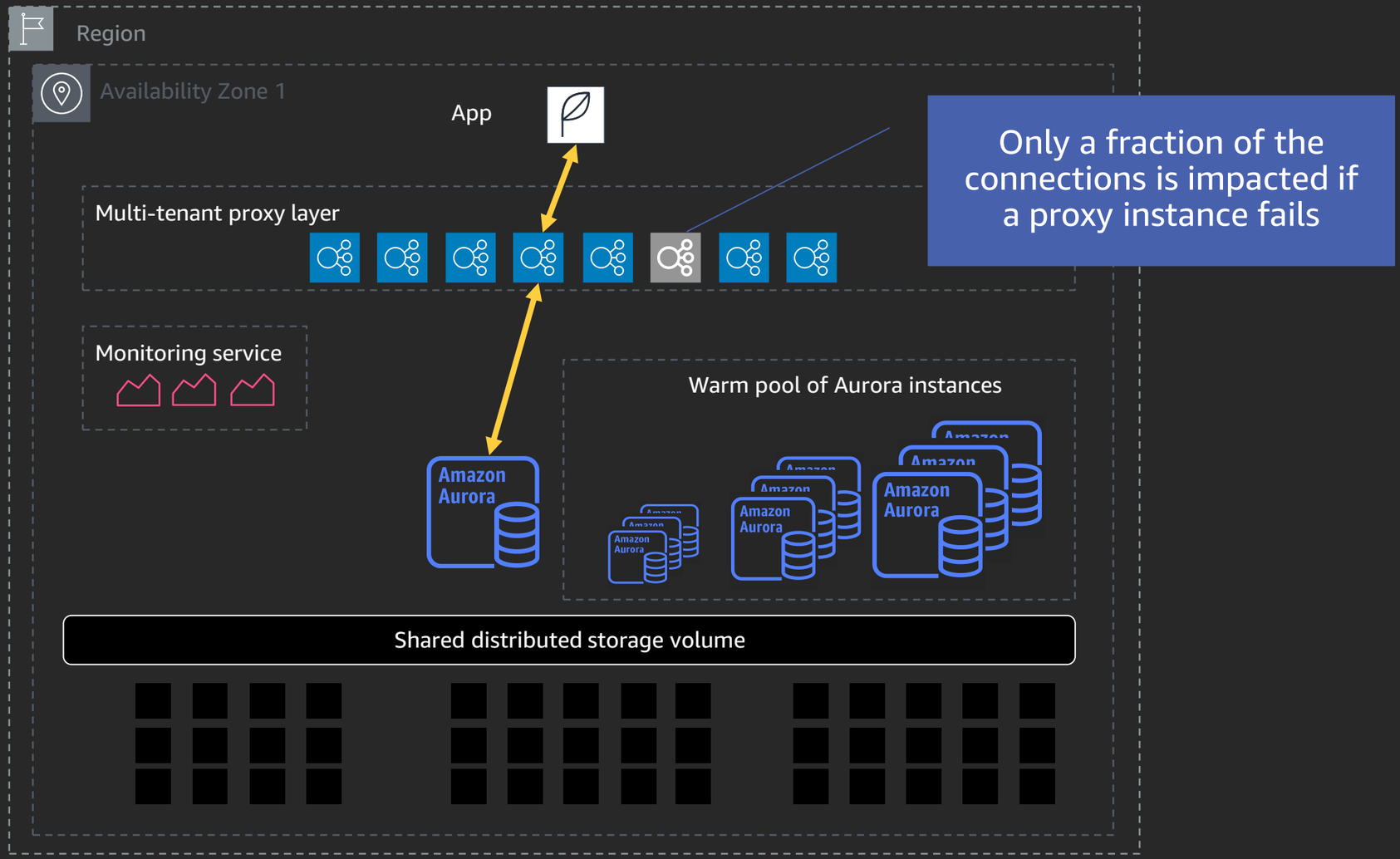


How does it work?

Spin-up on resume



How does it work?

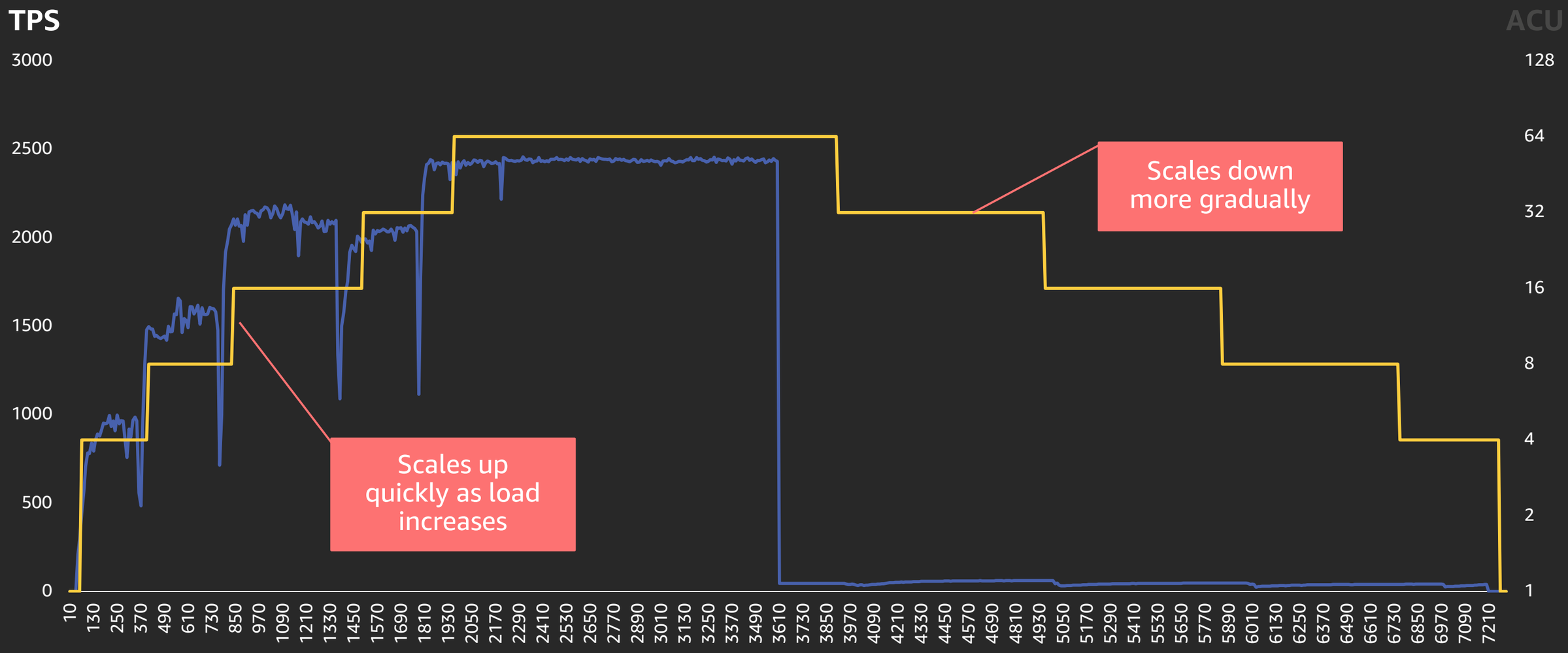


Only a fraction of the connections is impacted if a proxy instance fails

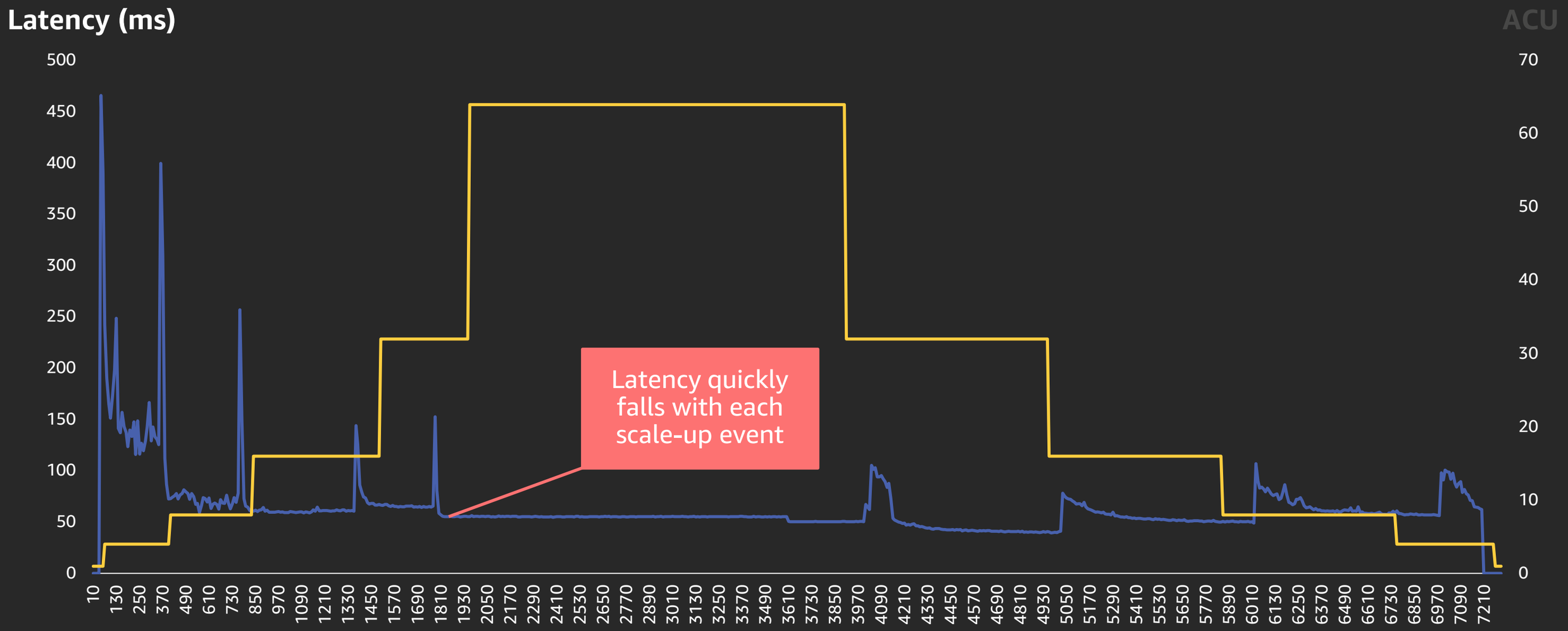
Simpler experience — less to worry about

- No CPU credits to monitor
- No commitment to particular availability zone
- No migrations between instance-type generations
- No DB instance reservations to manage
- No instances to manage
- Encryption at REST is always enabled
- No need to manually suspend and resume database
- No DNS propagation delays
- No maintenance window
- No old database versions to upgrade

Aurora Serverless: Scale up and down with load

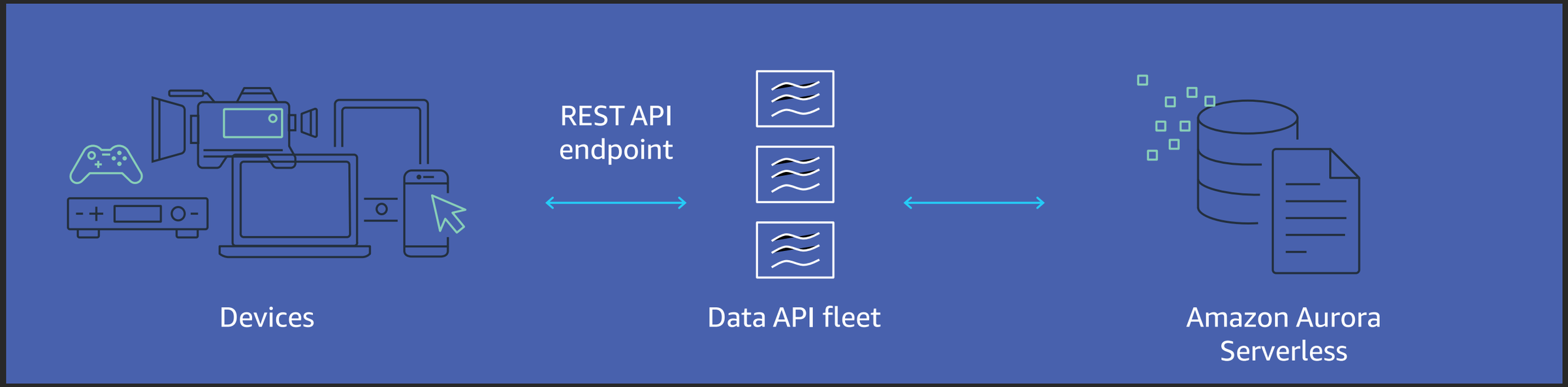


Aurora Serverless: Stabilizing latency quickly



Amazon RDS Data API

Amazon RDS Data API for serverless apps



Serverless apps often have restrictions

Limited network connectivity to the DB

No persistent connection to the DB

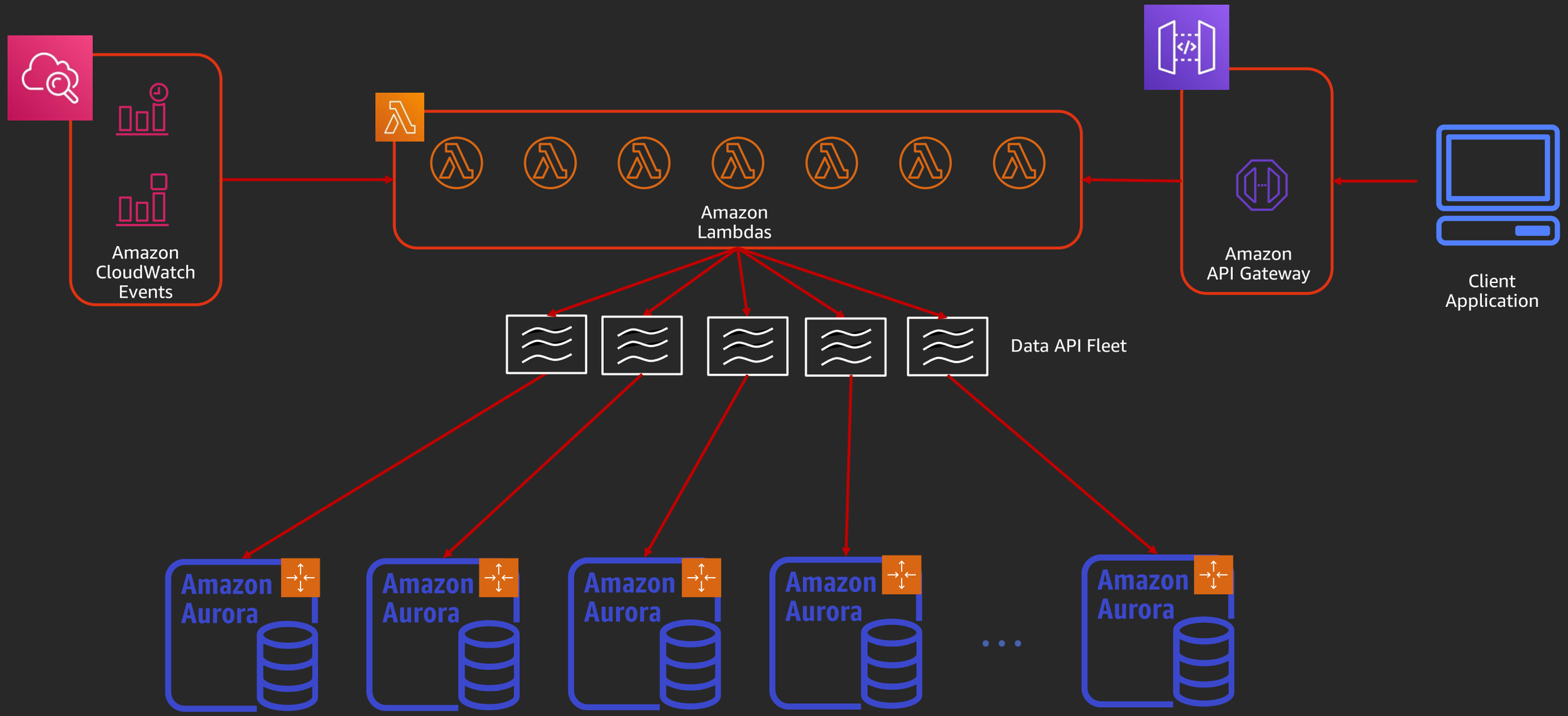
Small client (e.g., IoT) with limited resources

Amazon RDS Data API provides:

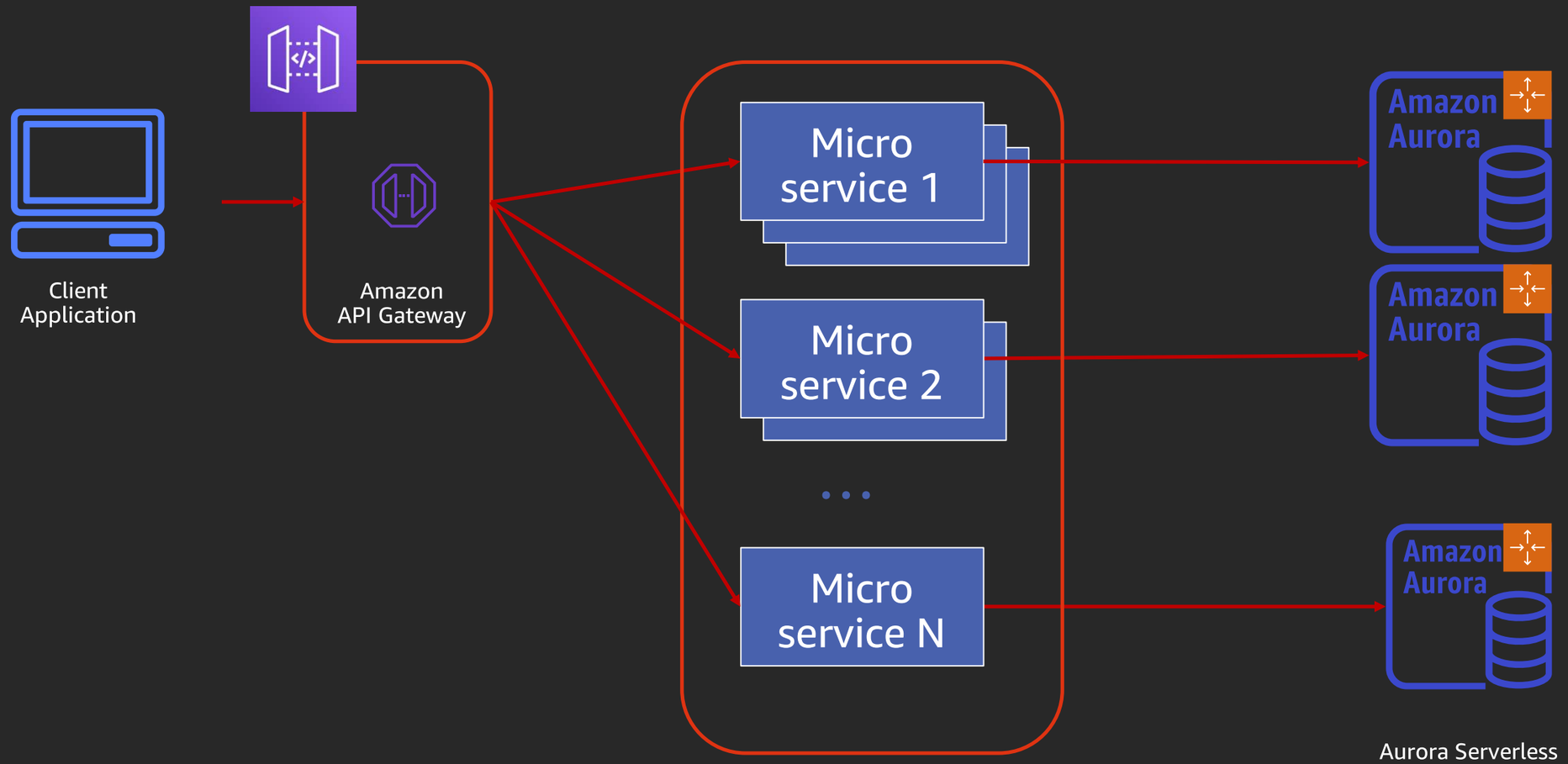
Public endpoint accessible via HTTP

Access without any client configuration

Modern serverless application architecture

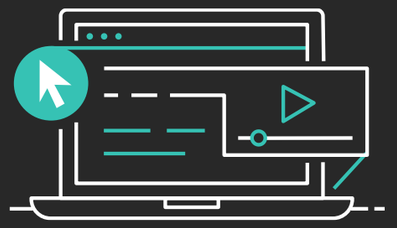


Modern microservice architecture



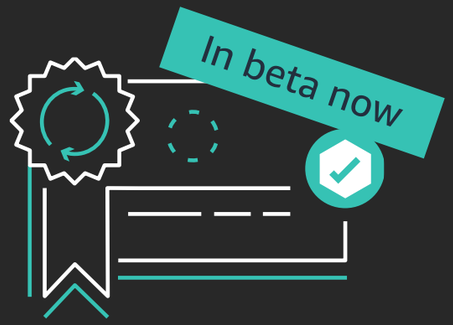
Learn databases with AWS Training and Certification

Resources created by the experts at AWS to help you build and validate database skills



25+ free digital training courses cover topics and services related to databases, including:

- Amazon Aurora
- Amazon Neptune
- Amazon DocumentDB
- Amazon DynamoDB
- Amazon ElastiCache
- Amazon Redshift
- Amazon RDS



Validate expertise with the new **AWS Certified Database - Specialty** beta exam

Visit aws.training

aws training and certification

Thank you!

AWS re:Invent

aws

Please complete the session survey in the mobile app.