AWS re:Invent

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

ADM304

Applying Amazon SageMaker to real-time advertising workloads

Gerry Louw
Head of WW Solutions Architecture, Advertising & Marketing Technology
AWS

Nick Reffitt (he/him)
VP, Data Science & Engineering
Tapjoy

Ram Vegiraju (he/him)
Machine Learning Solutions Architect
AWS

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

Agenda

Real-time advertising at Tapjoy

Amazon SageMaker

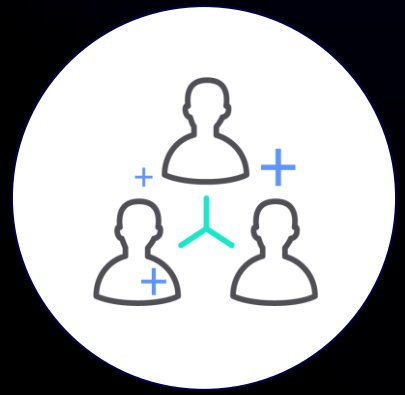
Best practices and recommendations

Q&A

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

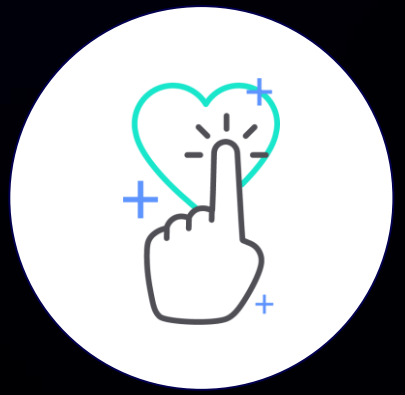
Mobile app advertising and monetization platform

Tapjoy



1.65B

active users per month



40M

daily ad engagements



Top 200

grossing app publishers trust our platform



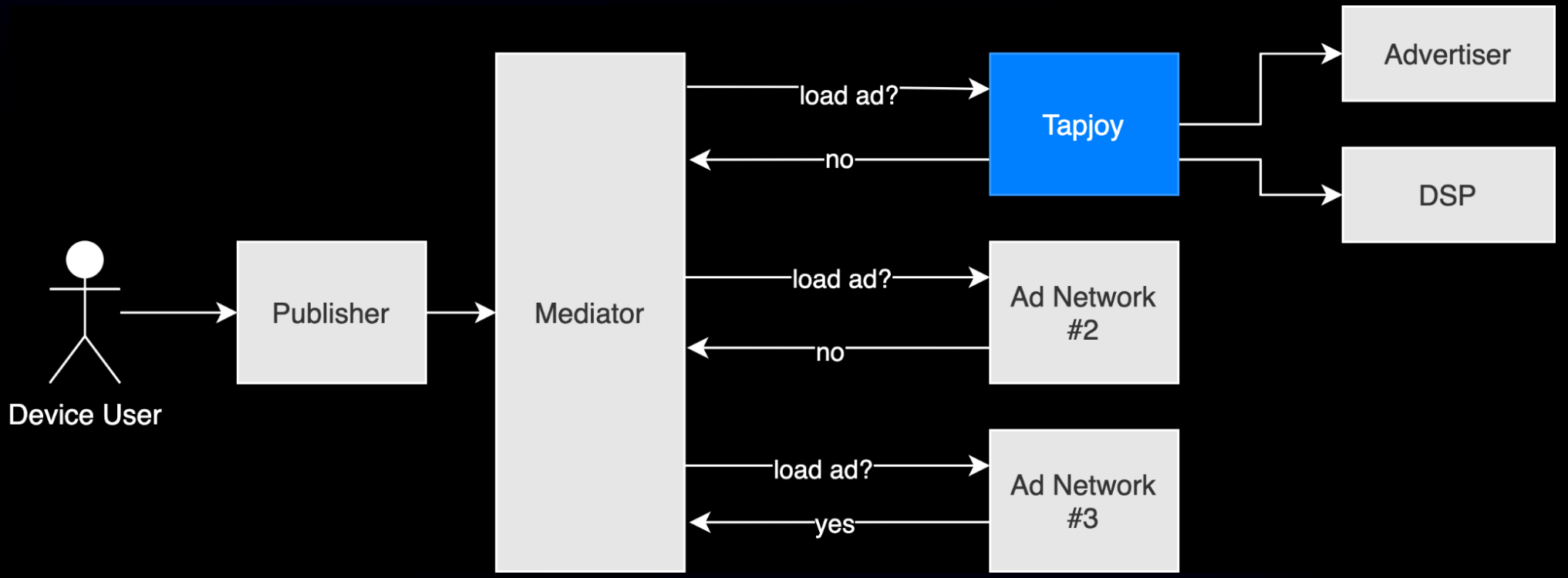
61K+

Tapjoy SDKs embedded in mobile apps

aws

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

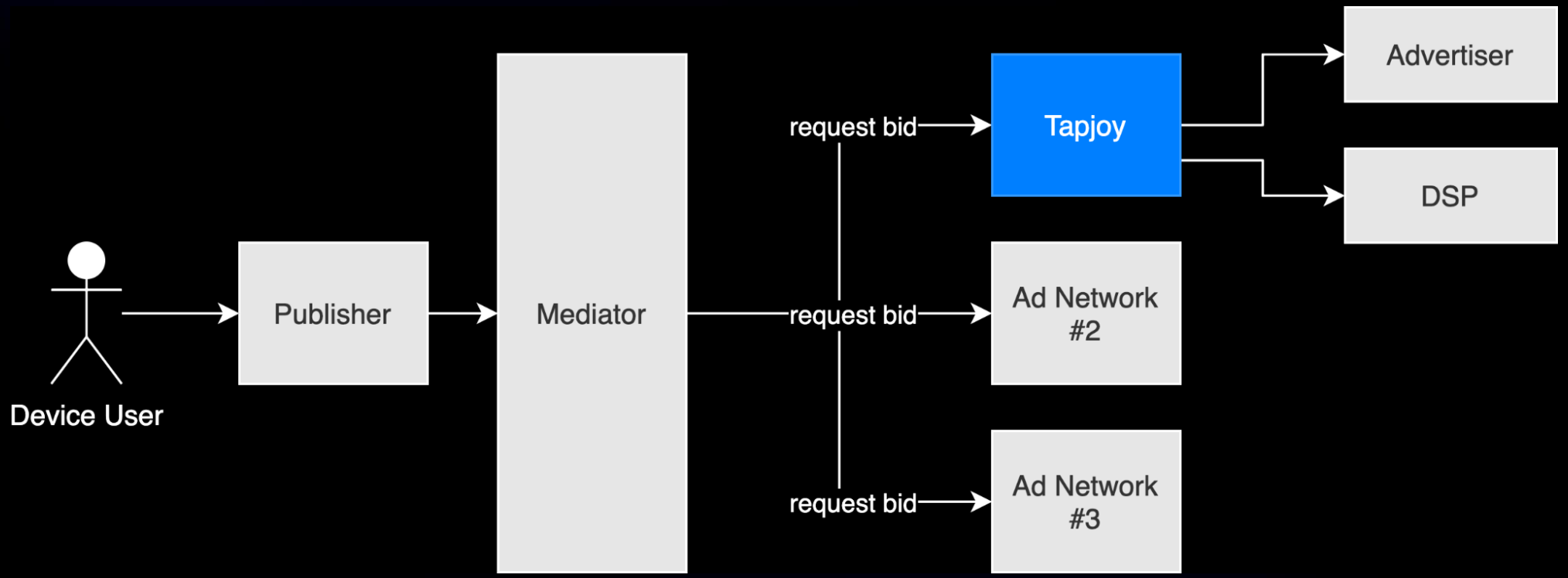
Pre-2017 mobile ecosystem: Waterfall



aws

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

Today’s mobile ecosystem: Programmatic



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

Data science requirements

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

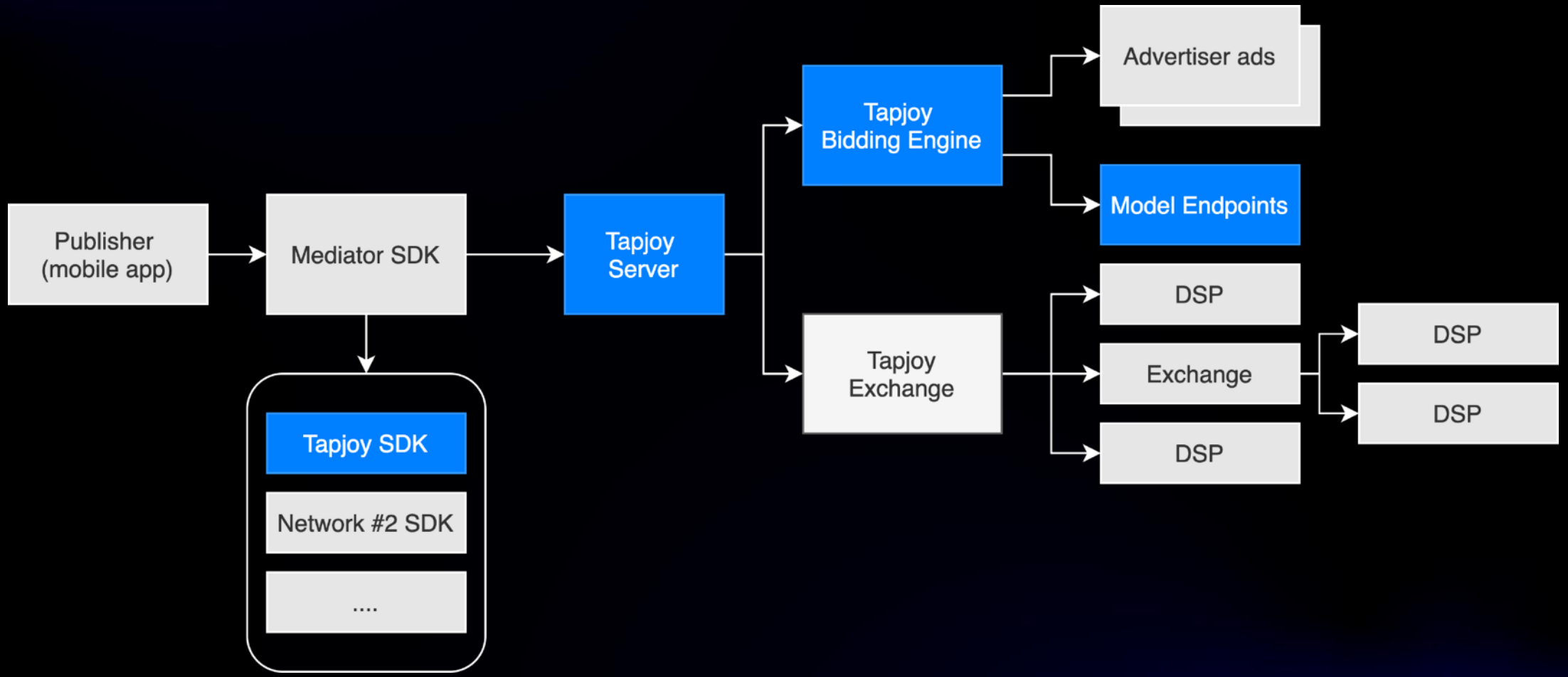
Data science requirements

- Real-time predictions of user behavior at scale (>50K RPS)
- Low latency (~5 ms)
- Flexibility to test several endpoints in production
- Machine learning (ML) framework agnostic
- Minimal overhead for infrastructure team
- Need to ensure within infrastructure budgets

aws

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

Ad request flow



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

ML platform journey

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

Products explored

Product	Key points
Amazon SageMaker	+ Feature-rich end-to-end ML platform + Managed service + Runs on AWS; majority of Tapjoy systems run here
Product 2	+ Managed service - Focus on TensorFlow; limited support for other frameworks - Increases dependency on multi-cloud providers
Product 3	+ Strong OSS community support - Significant ramp-up time on k8s for DS team - Not a managed service
Product 4	- Small OSS community - Not a managed service

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

Journey so far

- ~5 single-model and ~4 multi-model Amazon SageMaker endpoints handling ~70% of our production ad requests
- Using ~40 c5.2xlarge instances
- Model training and hyperparameter tuning achieved on Amazon SageMaker
- Inference via Client SDK – didn’t use AWS Lambda; no need to expose inference calls publicly
- Model deploys and retraining is performed manually
- Couchbase used to serve user-level features; contextual features stored in memory of our bidding engine

aws

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

Tapjoy: Best practices and recommendations

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

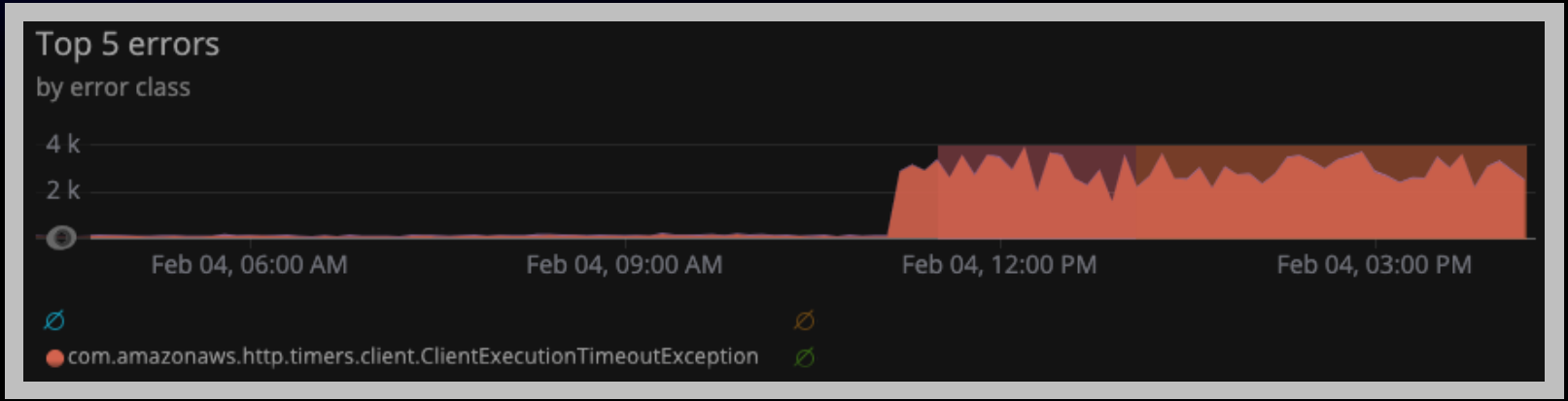
1. Keep track of transactions-per-second limits

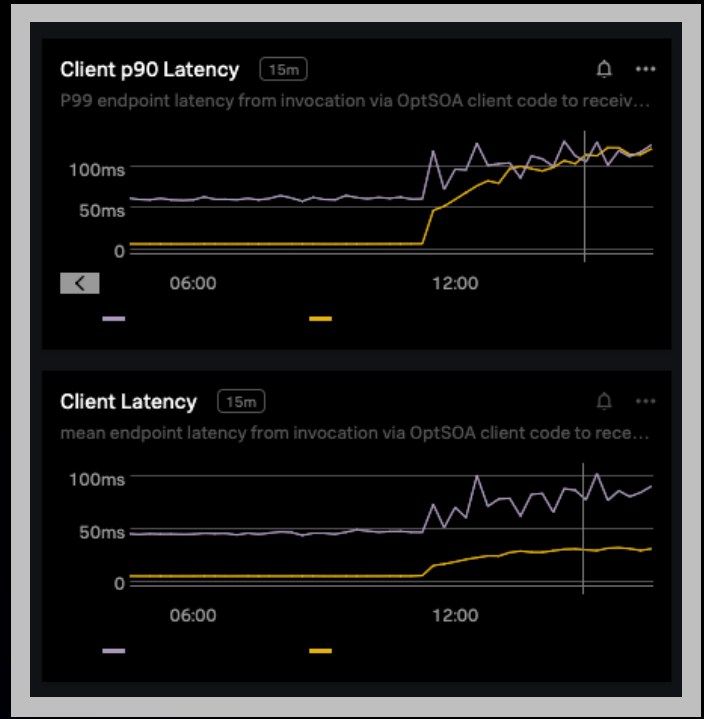
There is an account-level quota (not endpoint-level quota); SageMaker will respond with 5xx once quota is reached

Speak with your account manager and share a plan for phasing traffic into SageMaker (10K TPS is the limit initially)

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

Exceeding TPS quota





Client SDK will throw `ClientExecutionTimeoutException` when quota is reached

All SageMaker endpoints were overwhelmed – not just the endpoint with increased traffic (yellow line); this resulted in an increase in throughput

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

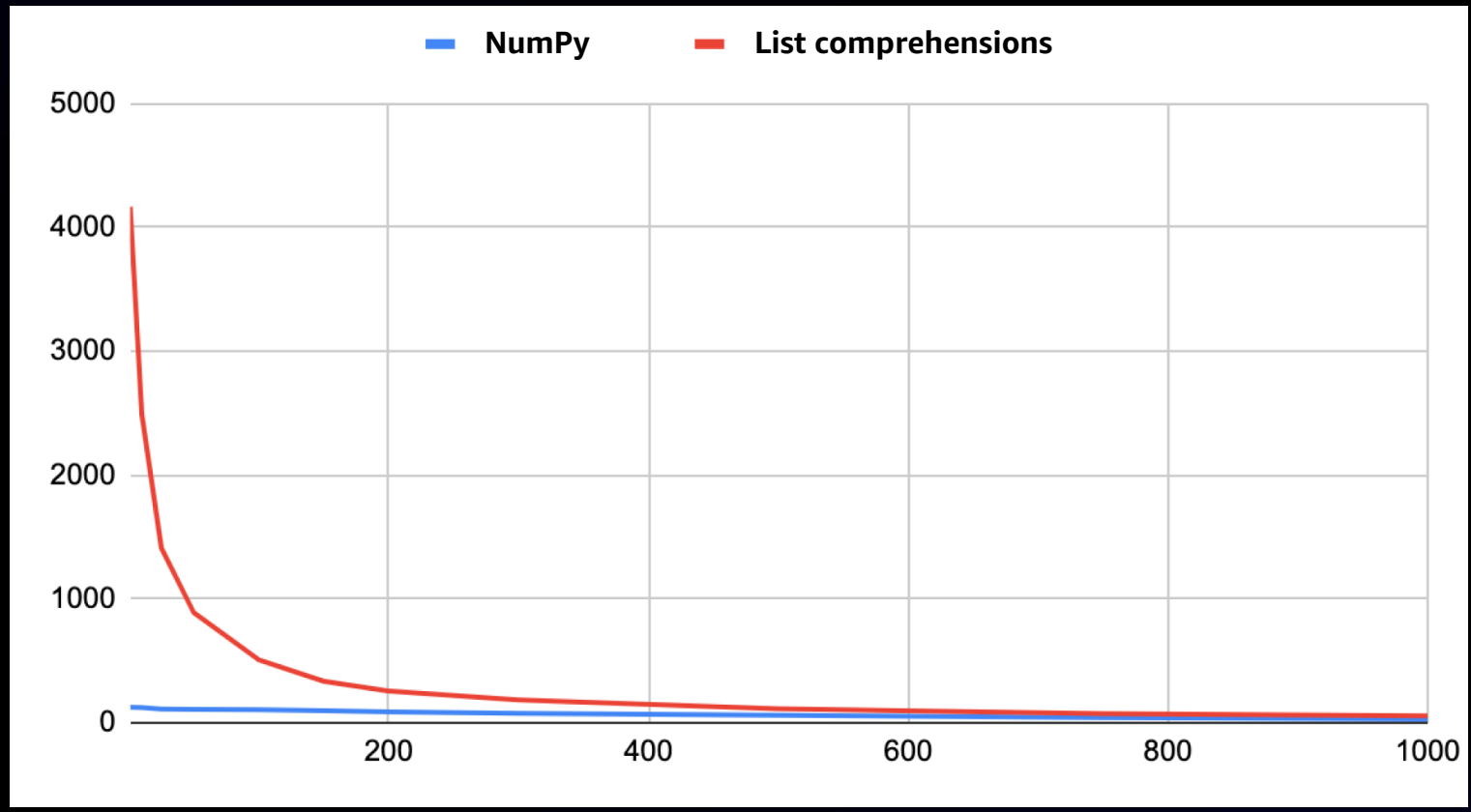
2. Model and overhead latency is tracked separately

Keep an eye on both; model latency is the latency that tracks both the model plus the inference code latency

You might have a fast model, but if your model inference code is slow, this will increase model latency

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

Throughput: NumPy versus list comprehensions



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

3. The objective function in hyperparameter tuning might not stop your jobs early

Some algorithms are supported, but not all, e.g., TensorFlow

Amazon SageMaker suggests to emit the value of the objective metric for Amazon SageMaker consumption

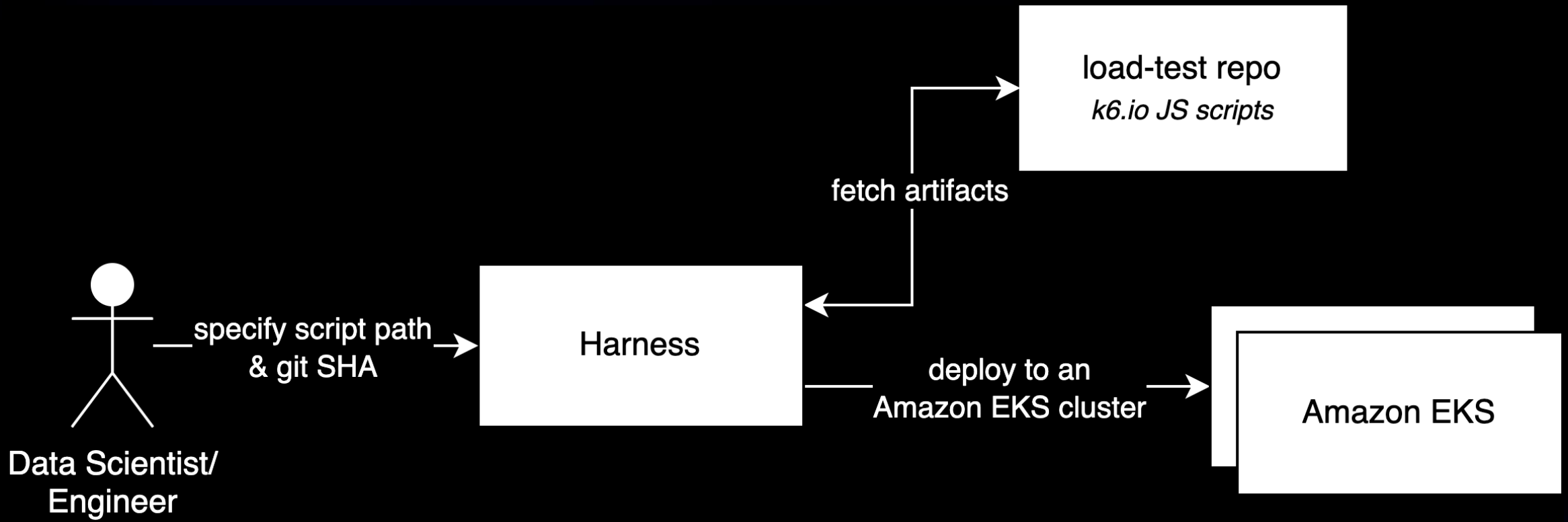
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

4. Load test your endpoints before going to production

This will help you identify bottlenecks in your model latency, as well as help with instance sizing

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

Load test deploy process



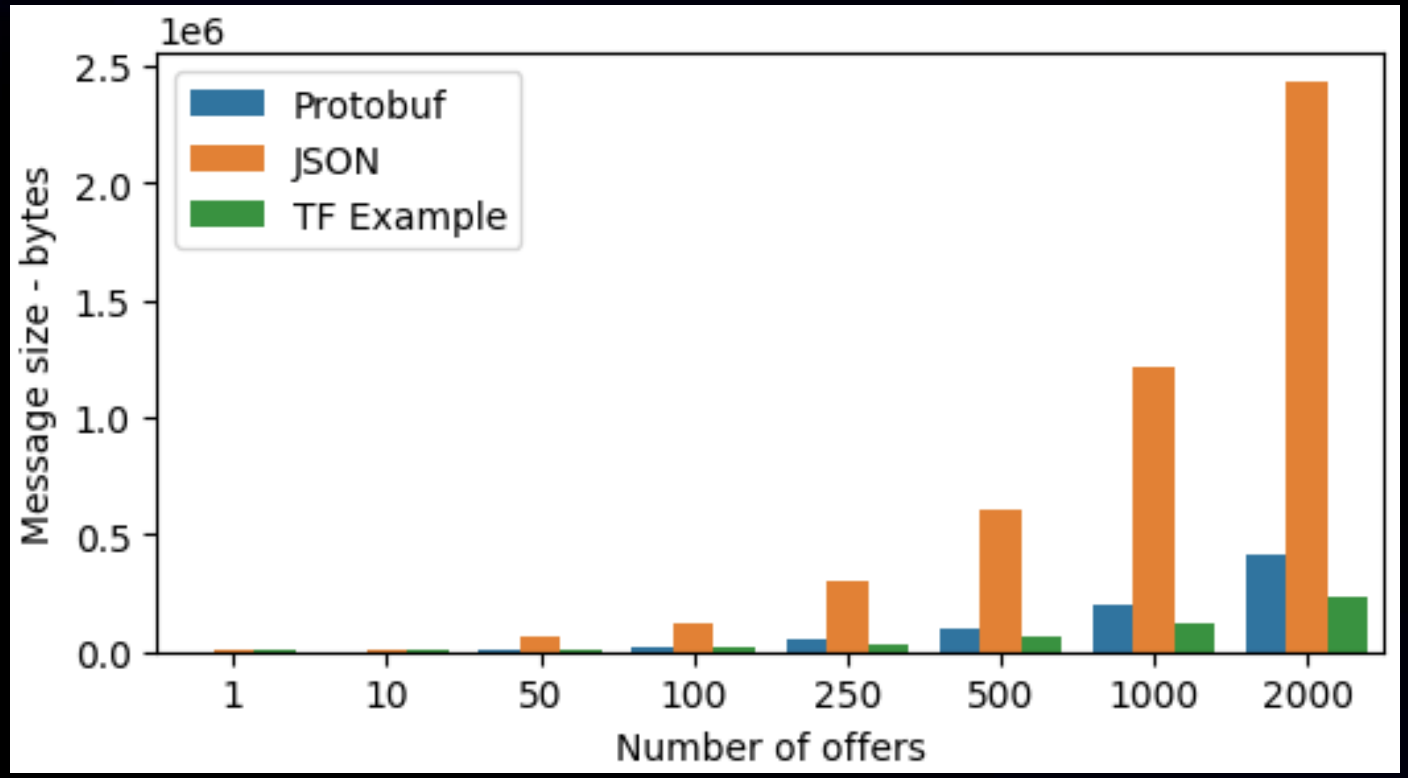
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

5. Test different serialization approaches

Test different serialization approaches (we started with JSON); SageMaker endpoints do not support content encoding, so we tested several frameworks for our recent models, and we landed on tf.train.Example

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

Tapjoy notebook test: JSON vs. TF Example



36% decrease in model latency

40% decrease on Amazon SageMaker costs

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

Future work

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

Future work

- Automate retraining using Amazon SageMaker data capture as input for retraining models that are stable, but are experiencing drift over time
- Automate model deployments
- Amazon SageMaker Model Monitor thresholds for model biases, tests to invoke retraining
- Adopt Amazon SageMaker model registry affords tracking and audit of trained models, and which are appropriate to promote to production

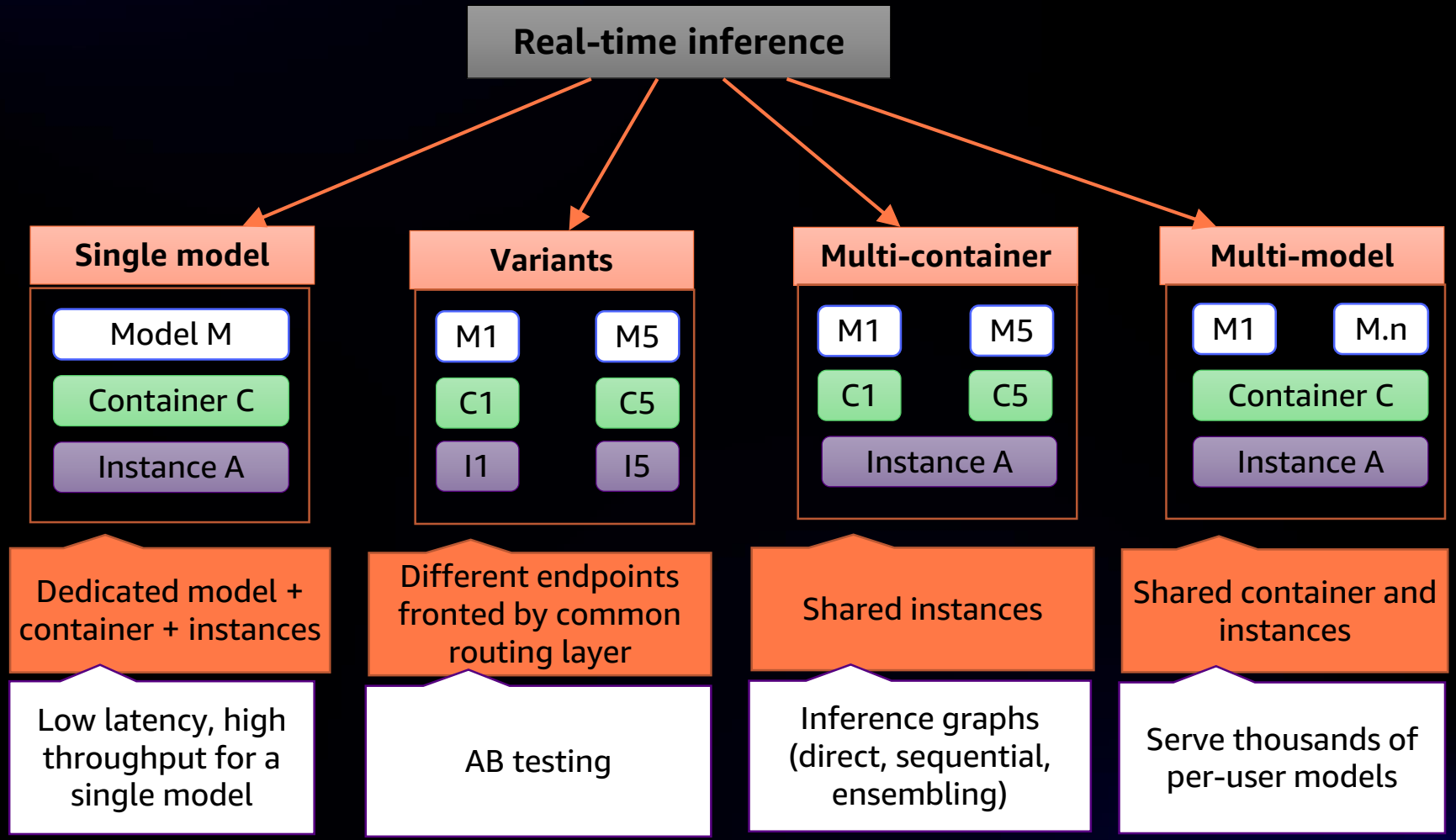
aws

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

Real-time inference with Amazon SageMaker

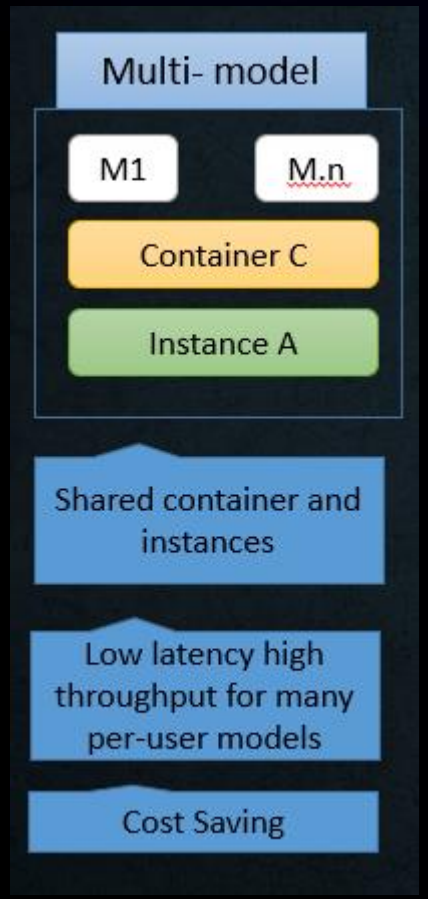
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon SageMaker real-time inference

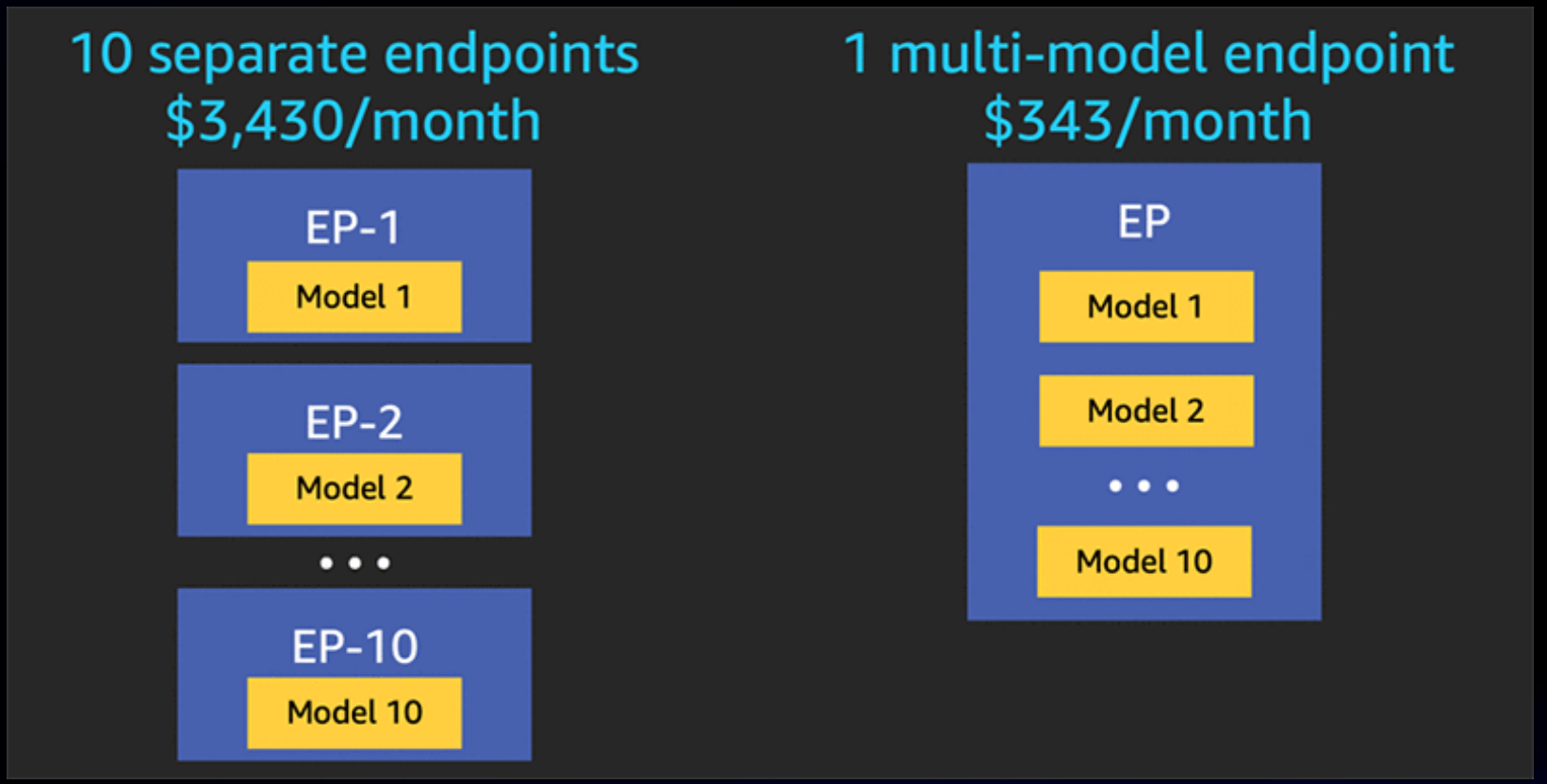


© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon SageMaker multi-model endpoints

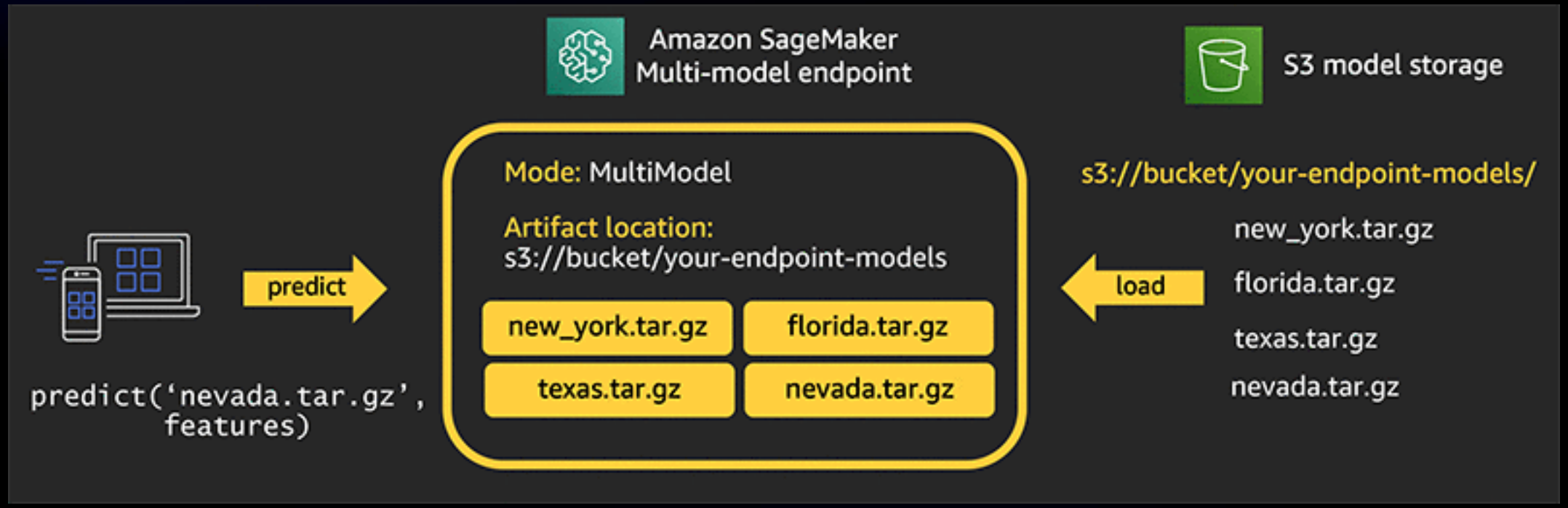


Example of multi-model endpoint savings



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

Amazon SageMaker multi-model endpoints



- Upload trained models to an Amazon S3 bucket; specify bucket/prefix when creating the endpoint
- Invoke a model with relative path; Amazon SageMaker resolves to full path and processes inferences with requested model
- Amazon SageMaker manages model loading/unloading and cache to keep frequently used models in memory
- Options to disable cache or load fixed set of models (cohort mode)

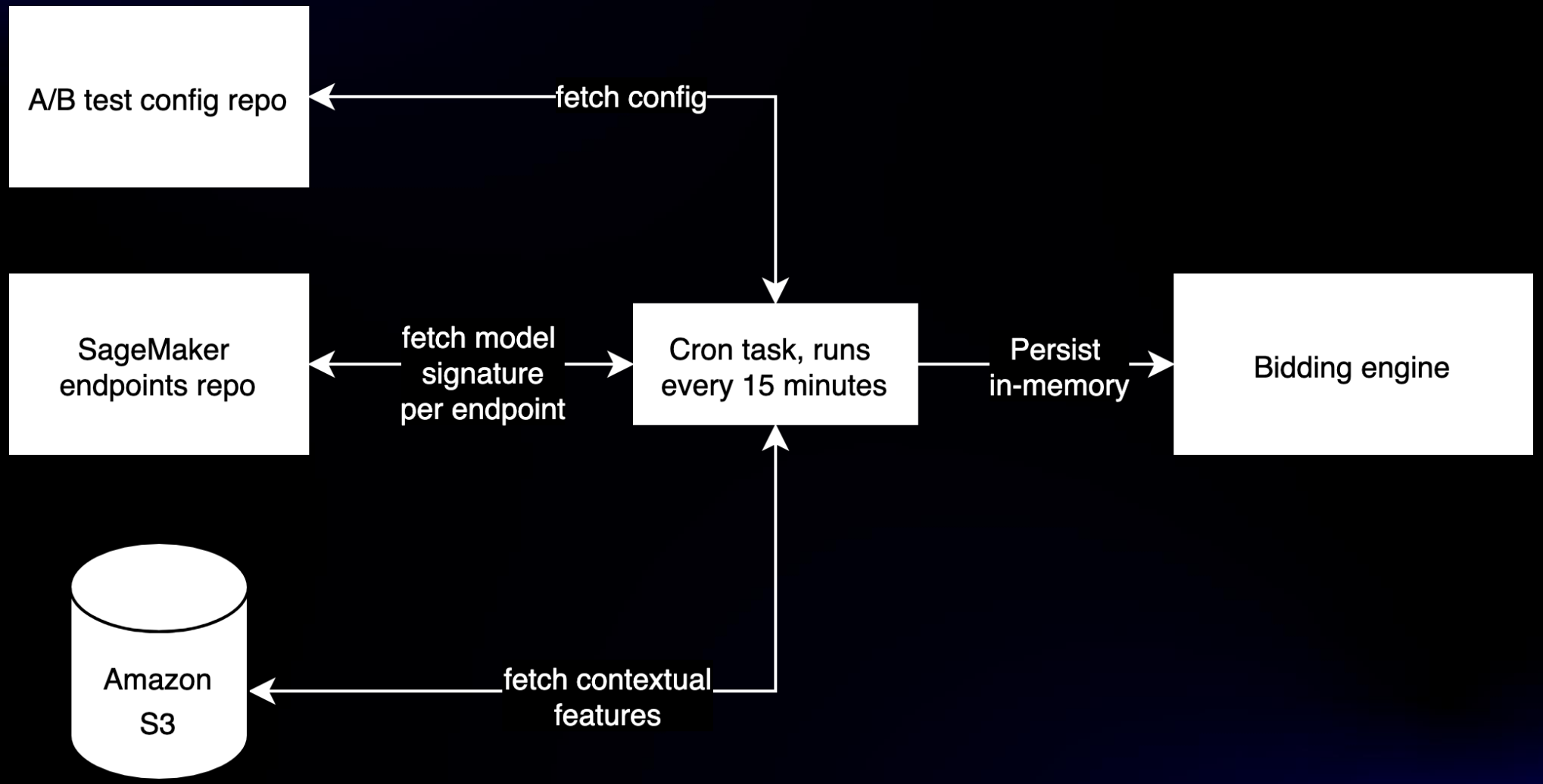
aws

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

Amazon SageMaker endpoint request flow

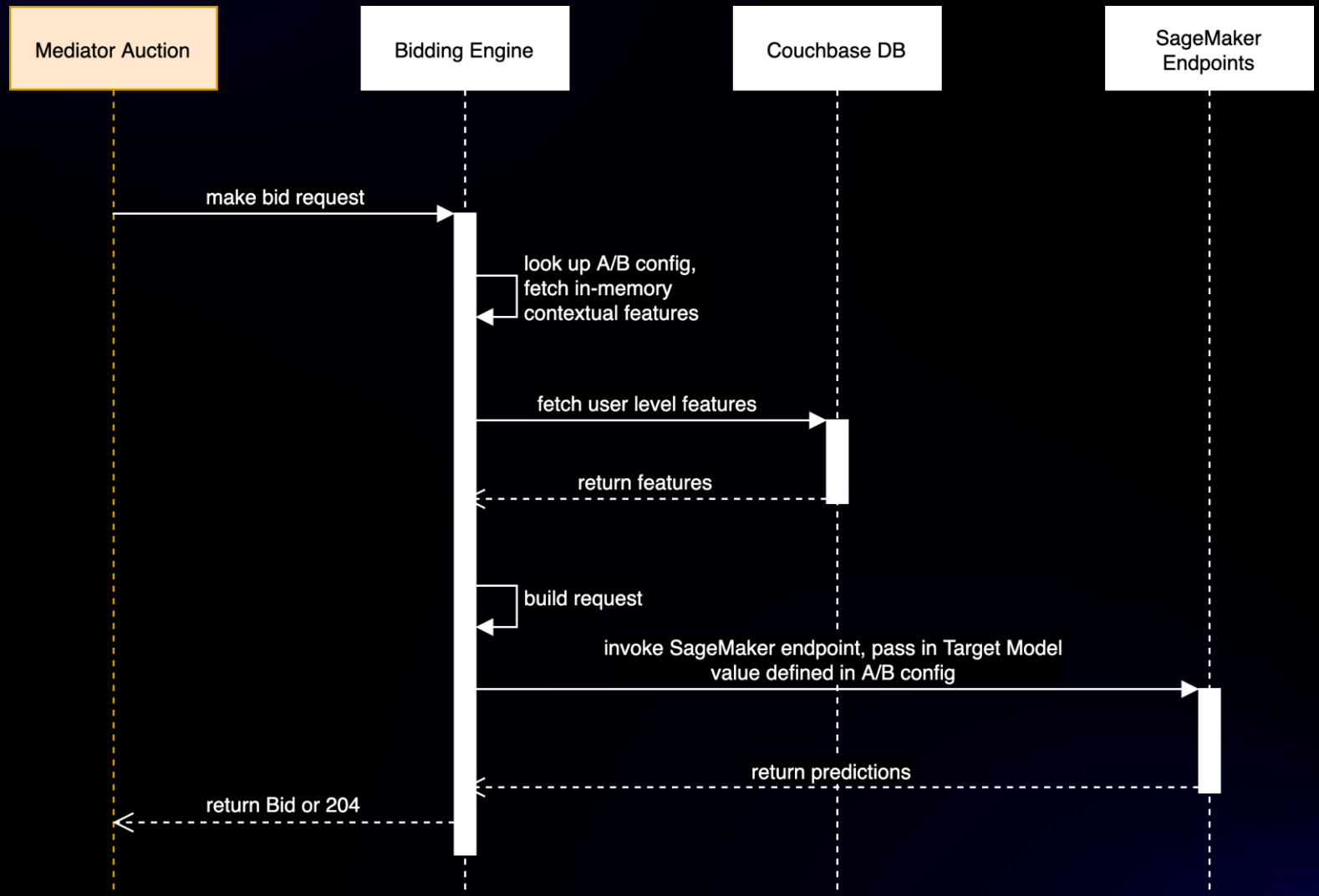
© 2021, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Amazon SageMaker endpoint request flow



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

Amazon SageMaker endpoint request flow



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

Thank you!

Gerry Louw
louwgerr@amazon.com

Nick Reffitt (he/him)
nick.reffitt@tapjoy.com

Ram Vegiraju (he/him)
rvegira@amazon.com

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