
AWS
re:Invent

ENT308-S

Build your next microservices application with modern AWS services

Nik Jain
Solution Architect
New Relic

Javier Miguez
Director of Technology Operations
Fleet Complete

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

Safe Harbor

This presentation and the information herein (including any information that may be incorporated by reference) is provided for informational purposes only and should not be construed as an offer, commitment, promise or obligation on behalf of New Relic, Inc. ("New Relic") to sell securities or deliver any product, material, code, functionality, or other feature. Any information provided hereby is proprietary to New Relic and may not be replicated or disclosed without New Relic's express written permission.

Such information may contain forward-looking statements within the meaning of federal securities laws. Any statement that is not a historical fact or refers to expectations, projections, future plans, objectives, estimates, goals, or other characterizations of future events is a forward-looking statement. These forward-looking statements can often be identified as such because the context of the statement will include words such as "believes," "anticipates," "expects" or words of similar import.

Actual results may differ materially from those expressed in these forward-looking statements, which speak only as of the date hereof, and are subject to change at any time without notice. Existing and prospective investors, customers and other third parties transacting business with New Relic are cautioned not to place undue reliance on this forward-looking information. The achievement or success of the matters covered by such forward-looking statements are based on New Relic's current assumptions, expectations, and beliefs and are subject to substantial risks, uncertainties, assumptions, and changes in circumstances that may cause the actual results, performance, or achievements to differ materially from those expressed or implied in any forward-looking statement. Further information on factors that could affect such forward-looking statements is included in the filings New Relic makes with the SEC from time to time. Copies of these documents may be obtained by visiting New Relic's Investor Relations website at ir.newrelic.com or the SEC's website at www.sec.gov.

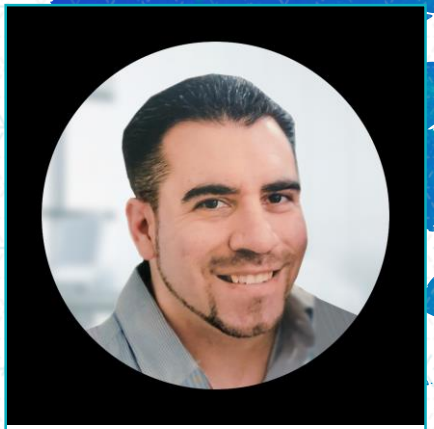
New Relic assumes no obligation and does not intend to update these forward-looking statements, except as required by law. New Relic makes no warranties, expressed or implied, in this presentation or otherwise, with respect to the information provided.

©2008–19 New Relic, Inc. All rights reserved

Nik Jain

Solutions Architect
Living in Melbourne

©2008–19 New Relic, Inc. All rights reserved



Javier Miguez

Director, TechOps
Living in Toronto

©2008–19 New Relic, Inc. All rights reserved

Agenda

1. Definition of a modern microservice
2. How Fleet Complete achieves its double bottom line
3. Human migration to space app architecture
4. Live demonstration and app interaction
5. Observability strategy to build and manage modern services

©2008–19 New Relic, Inc. All rights reserved

Start with why?!

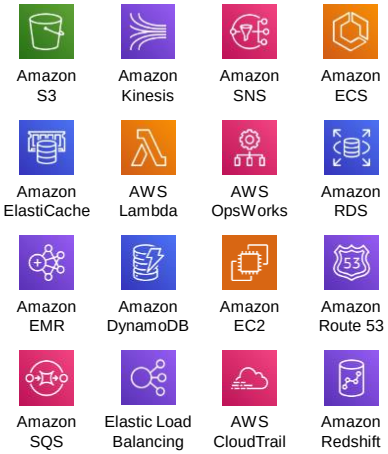
Containers/Virtualization

Enterprise Relational Databases

Queues

Hosts

SaaS Services



Infrastructure Applications

Modern Databases & Storage

©2008–19 New Relic, Inc. All rights reserved

FleetComplete® Transformation

Monolithic

vmware

Microsoft SQL Server

Microservices

kubernetes

Linux

Apache Kafka
A high-throughput distributed messaging system.

redis

PostgreSQL

Spark

mongoDB

elastic

RabbitMQ

HAPROXY

Modern and Managed Technologies

New Relic

Amazon EC2

Amazon IoT Core

Amazon Kinesis

Amazon SNS

Amazon Fargate

Amazon ElastiCache

AWS Lambda

AWS OpsWorks

Amazon RDS

Amazon EMR

Amazon DynamoDB

Amazon Elasticsearch

Amazon Route 53

Amazon SQS

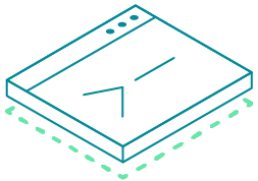
Elastic Load Balancing

AWS CloudTrail

Amazon Redshift

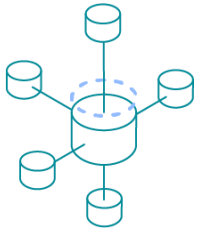
©2008–19 New Relic, Inc. All rights reserved

Core ingredients of a modern microservice



DevOps

Leverages a
code pipeline



Autoscaled

Doesn’t require
manual scaling

Cloud-Native

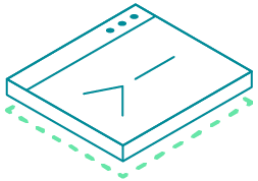
Based on
cloud compute

Orchestrated

Low cost of
operational burden

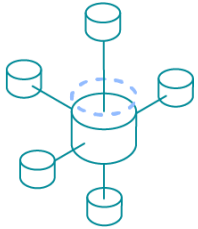
©2008–19 New Relic, Inc. All rights reserved

Core ingredients of a modern microservice



DevOps

Leverages a
code pipeline



Autoscaled

Doesn't require
manual scaling

Cloud-Native

Based on
cloud compute

Orchestrated

Low cost of
operational burden

Amazon
EC2 Auto Scaling

AWS
Lambda

AWS
CloudFront

Amazon
EKS

©2008–19 New Relic, Inc. All rights reserved

Why Amazon EKS + AWS Lambda + AWS Fargate?

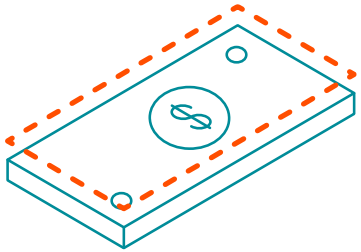
Operations

Don’t worry about underlying servers

Amazon EKS/Fargate helps you
move faster

Cost-efficient

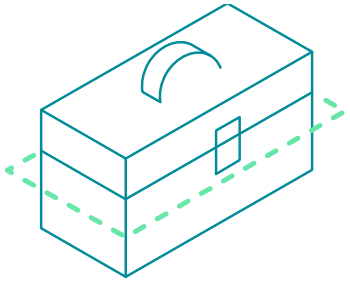
No traffic,
no cost



Amazon EKS/Fargate is
cost efficient

Productivity

Higher abstraction



Amazon EKS/Fargate is
portable

©2008–19 New Relic, Inc. All rights reserved

Why Amazon EKS + AWS Lambda + AWS Fargate?

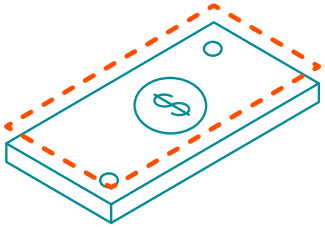
Operations

Don’t worry about underlying servers

Amazon EKS/Fargate helps you
move faster

Cost-efficient

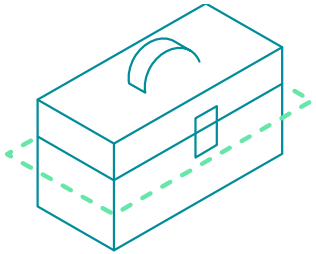
No traffic,
no cost



Amazon EKS/Fargate is
cost efficient

Productivity

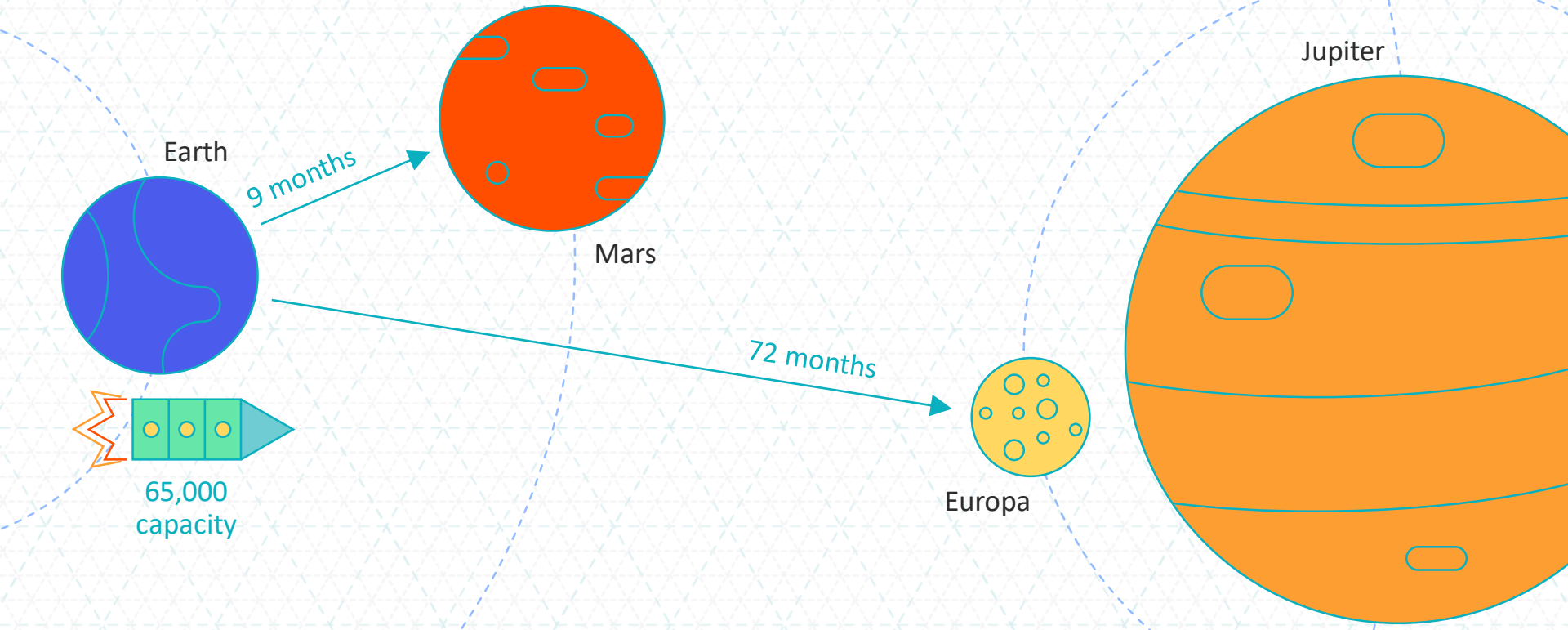
Higher abstraction



Amazon EKS/Fargate is
portable

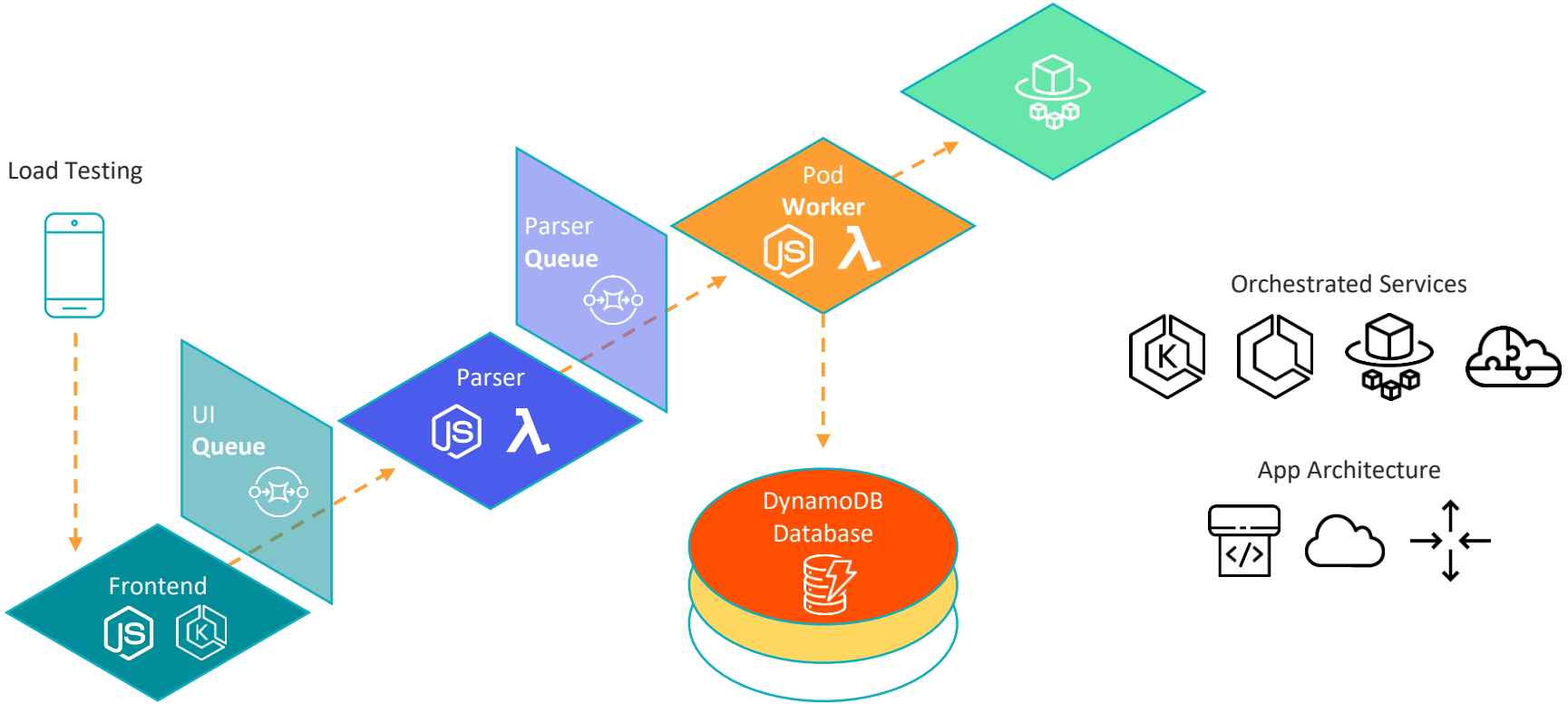
©2008–19 New Relic, Inc. All rights reserved

Mass space migration application



©2008–19 New Relic, Inc. All rights reserved

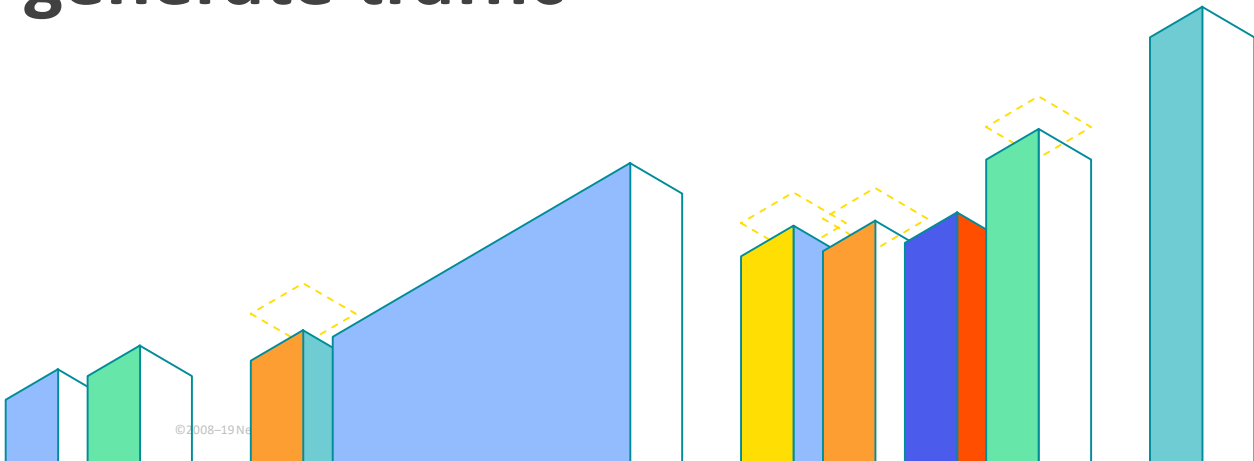
Mass space migration app on AWS



©2008–19 New Relic, Inc. All rights reserved

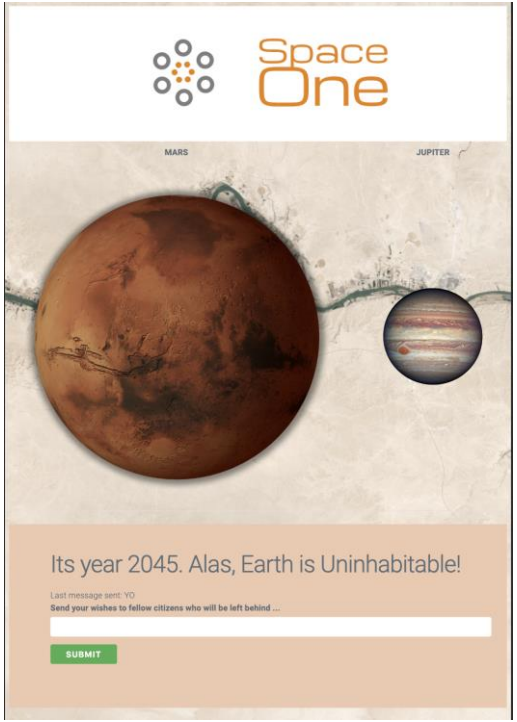
v1.0 Demo

Goal: Interact with the app and generate traffic



©2008–19 N

Pull out your phones …



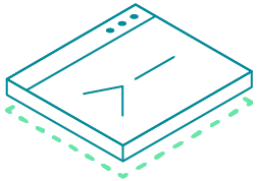
©2008–19 New Relic, Inc. All rights reserved

Key takeaway

- Modern AWS services allow for the least possible operational burden
- AWS CloudFormation templates are easy (automate the deployment of resources)
- Understand cloud cost to namespace level (Amazon EKS)

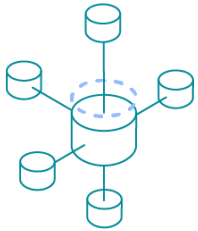
©2008–19 New Relic, Inc. All rights reserved

Core ingredients of a modern microservice



DevOps

Leverages a
code pipeline



Autoscaled

Doesn't require
manual scaling

Cloud-Native

Based on
cloud compute

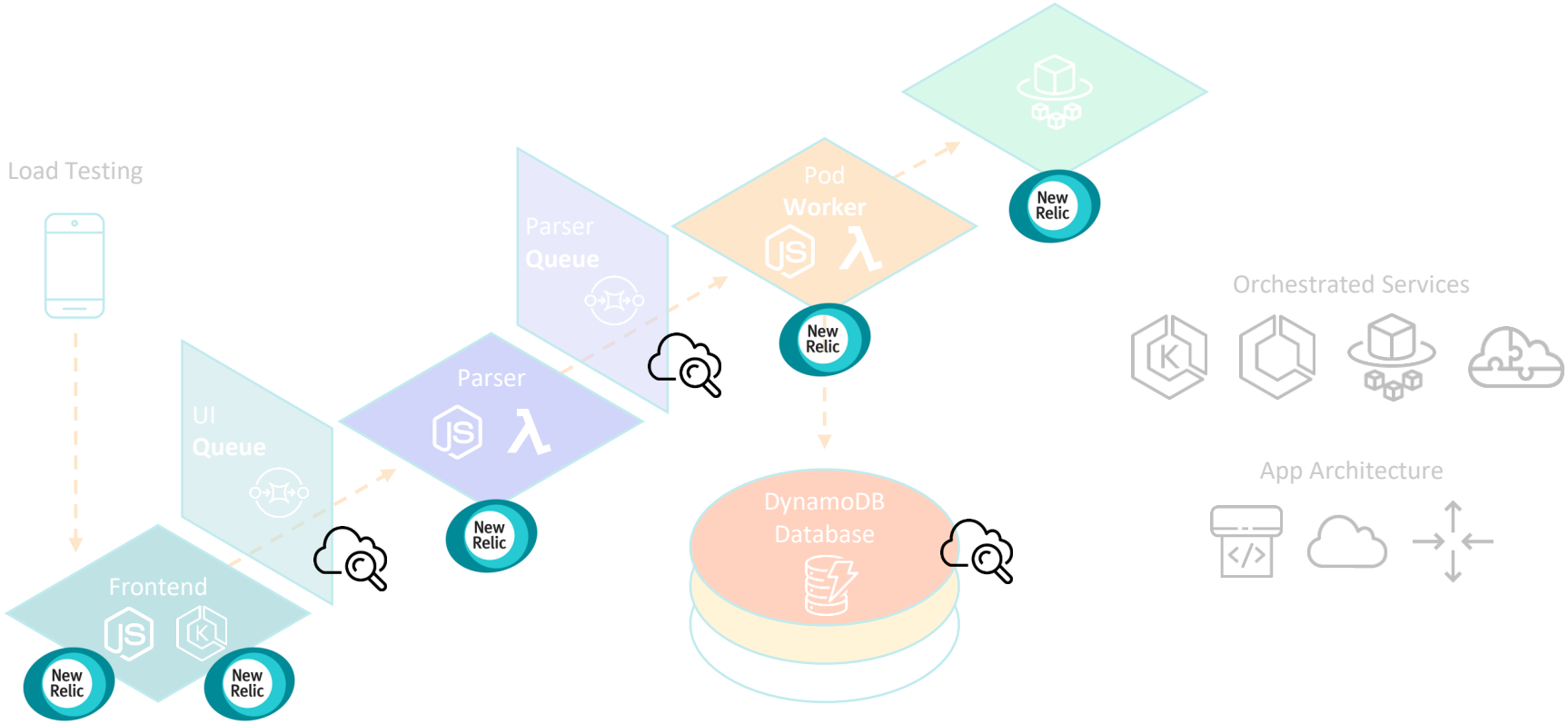
Orchestrated

Low cost of
operational burden

Bedrock of true observability

©2008–19 New Relic, Inc. All rights reserved

Mass space migration app on AWS



©2008–19 New Relic, Inc. All rights reserved

v2.0 Demo

Goal: Use telescopic and microscopic observability patterns for quick resolution

©2008–19 N

Key takeaway

- Easy-to-use wizard and tools for full observability
- Codify your monitoring, and monitor your code pipeline
- Keep the compute lean, keep the planet green (Cloud Optimize)

©2008–19 New Relic, Inc. All rights reserved

An effective observability reliability model

Failure modes	SLIs	SLOs	SLAs
Slow page views (RRN: 900)	95th percentile page view time	99% of page views <7 seconds	99.99% operational (functional uptime)
Business errors (RRN: 1000)	Value streams ($$) error rate	Error rate by client journey <0.05% (error budgets)	99.95% error free experience
Microservices latency (RRN: 800)	95th percentile response time	Min 3 release a day Zero 5xx errors <1 second response time	...
...	...	...	...

©2008–19 New Relic, Inc. All rights reserved

The best way to speed up a website is almost always to add cache and more cache!

(+ AWS + New Relic)

- highscalability.com

©2008–19 New Relic, Inc. All rights reserved

Let’s talk

Join us:

Venetian Suites 3221 and 3222

Level 2 – Grand Canal Shoppe Level

Wednesday, December 4

4:30pm to 6:00pm

©2008–19 New Relic, Inc. All rights reserved

Thank you!

Nik Jain
njain@newrelic.com
https://www.linkedin.com/in/nikmjain

Javier Miguez
javier.miguez@fleetcomplete.com
https://www.linkedin.com/in/javier-miguez

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

Please complete the session survey in the mobile app.

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