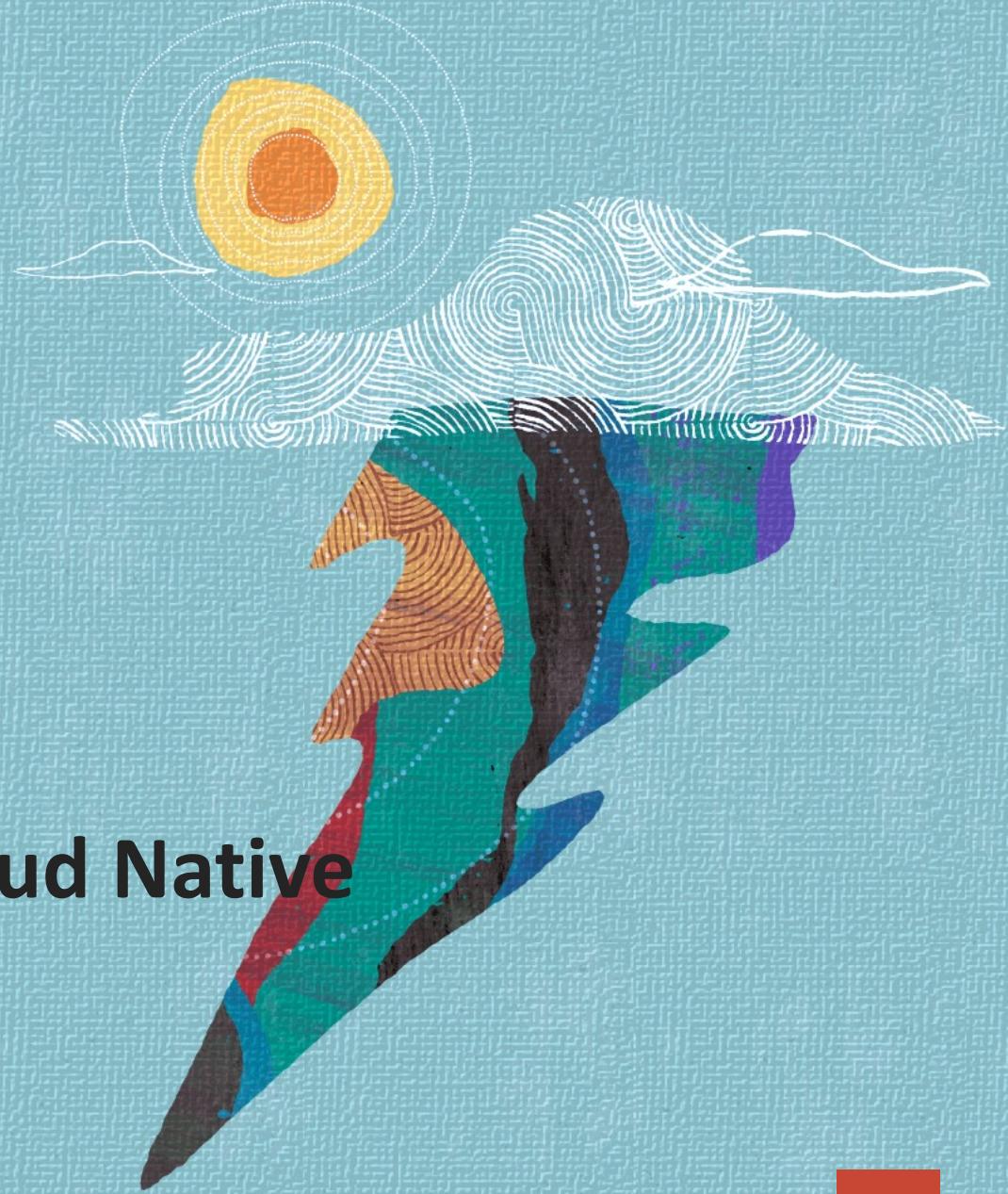


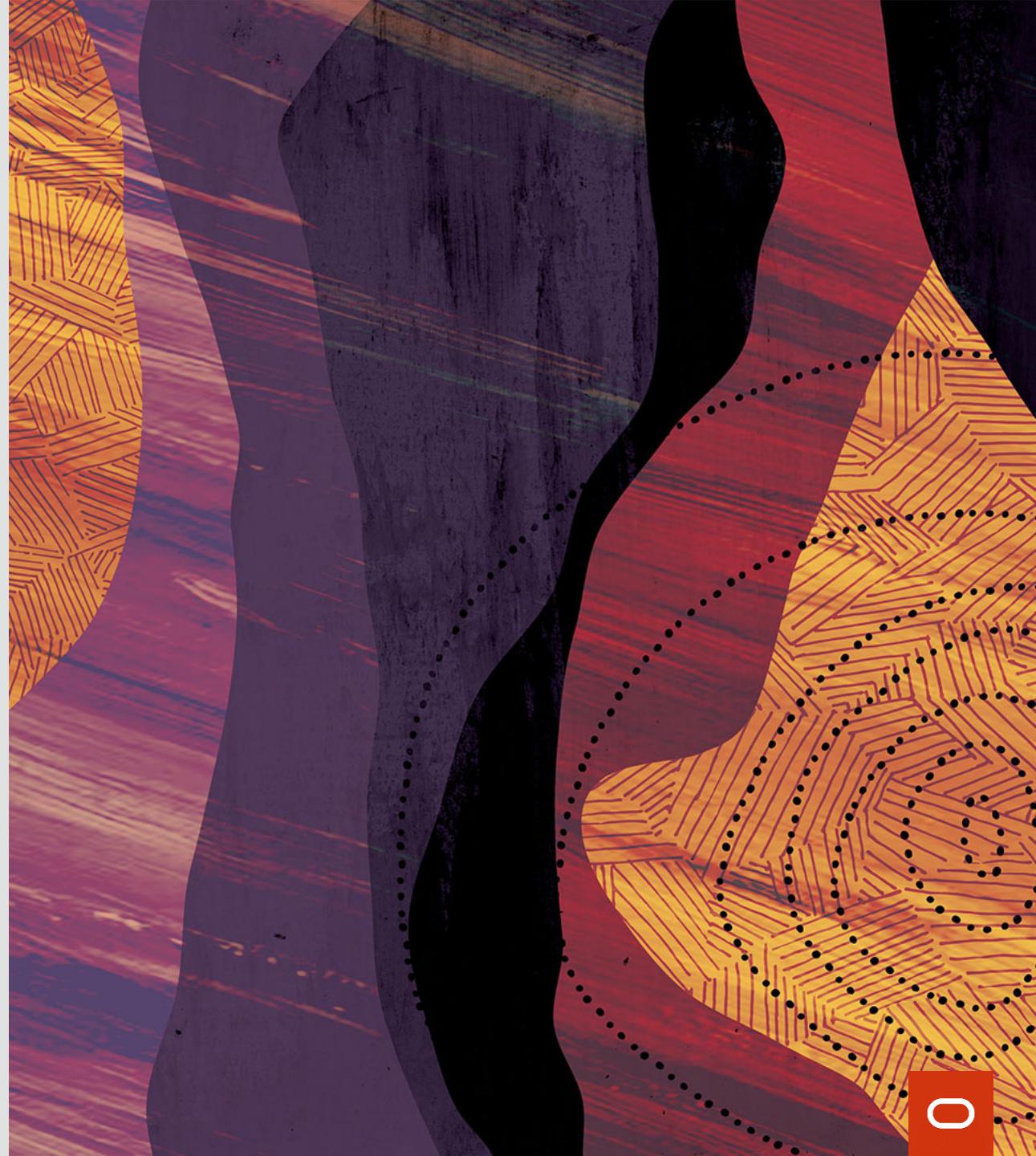
Be British and start queueing – or how I leaned to Queue with Cloud Native Oracle Cloud Infrastructure Queue

Phil Wilkins
Developer Evangelist



OCI Queue

Overview

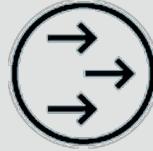
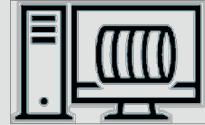
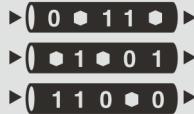


OCI Queue – What?

OCI Queue is a fully managed message service that helps decouple and scale distributed applications and micro-services.

Queue is designed to support high volume of transactional data where throughput scales up and down automatically and transparently for the clients (producers/consumers).

Spectrum of Choices



Distributed Streaming

- Point-to-(Multi)Point
- Brokered communication
- Real-time, high throughput messaging
- Message Re-playability
- Ordering guarantees
- Ex: OCI Streams (Kafka)

Message Brokers

- Point-to-(Multi)Point
- Brokered communication
- Push based async semantics
- Relaxed delivery guarantees
- Ex: Oracle TEQ (AQ)

Message Queues

- Point-to-Point
- Brokered communication
- Reliable async communication
- Once-only Consumption
- Processing guarantees
- Multi-protocol
- Auto scaling
- Ex: OCI Queue

(Streaming)Web Svc

- Point-to-Point
- Direct communication
- Synchronous communication
- Traditional Messaging semantics
- HTTP Web Services, gRPC

Decision Matrix to help select the right tech...

- The architecture center includes a decision matrix that provides the information to help you choose the correct technology
- Helping the less techie - [APIs vs messaging](#) – authority page
- Oracle Learning Library videos

◀ Back to Architecture Center

Get Started

Plan

Identify the Most Appropriate Messaging Service

Considerations for Selecting a Messaging Service

Review the Decision Matrix

About Technology-Based Message Transmission

Explore

Acknowledgments

To help remediate issues with distributed applications, an audit trail enables tracking where a message was transmitted to and received from. Having an audit trail can be useful when you must comply with legislative requirements. Therefore, it may be necessary to learn how the service can support this requirement.

Review the Decision Matrix

Use the following decision matrix to learn about the available choices of OCI services and the considerations or evaluation factors for choosing a particular service.

Service/Factor	Queue	Streaming	Notifications	Oracle Integration (Gen2, Gen3)	Autonomous Database (TEQ and AQ)
Choreography versus Orchestration	Choreography	Choreography	Choreography	Choreography and Orchestration (Depends on integration configuration)	Choreography and Orchestration (Depends on integration configuration)
Pay per message or event versus bulk purchase	Per API call + multiplier for messages > 64 KB	Per GB data transfer + Storage	Depends upon the communication channel. Priced by quantity sent.	Price in blocks of 5000 integration calls per hour	No, Database pricing
Single or multiple consumers for a message	Single consumer. (Race conditions allowed, but only one consumer per message).	Single and Multiple	Single and Multiple	Single and Multiple (Depends on integration definition)	Single and Multiple
Message retention after reading	No	Yes	No	Yes (Depends on integration definition)	Yes
Guaranteed order	Best effort	Yes, within a partition	No	Yes (Depends on integration definition)	Partial (Rules about priorities can be set, which can dictate consumption order)
Asynchronous operation	Yes	Yes	Yes	Yes (Depends on integration definition)	Yes

Challenges with Traditional Message Brokers

Setting up and managing the service

"We currently run multiple ActiveMQ and RabbitMQ instances however managing those take away valuable time and resource from us"

"We don't have the resources to manage the JMS queues on WebLogic and provide the SLA our business wants from us"

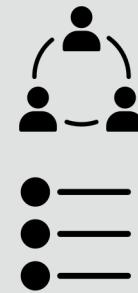
Cost of service

"We use message brokers to smooth temporary volume spikes in our system, however we have to allocate the size of cluster based on the peak load and pay for it"

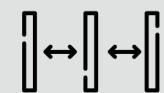
Vendor lock-in

"Our customers don't want to get locked in to any particular vendor (e.g. IBM MQ) and want to use open-source APIs"

OCI Queue – What makes it so Powerful?



Parallel Processing
Multiple Producers
Multiple Consumers



Horizontally Scalable



Highly Performant

OCI Queue – What makes it so Powerful?



Managed

Automates all infrastructure and platform maintenance

Patches all software **online**, tunes settings, diagnoses errors



Secure

Protects data from external and internal threats

Custom encryption key, encrypts all data at rest and in motion, **Private Endpoints**



Scale

Scales online for highest performance and lowest cost

Auto Scales Up or Down as needed; instant, online

OCI Queue – What makes it so Powerful?



Pay as you use

Best price-performance
guarantee

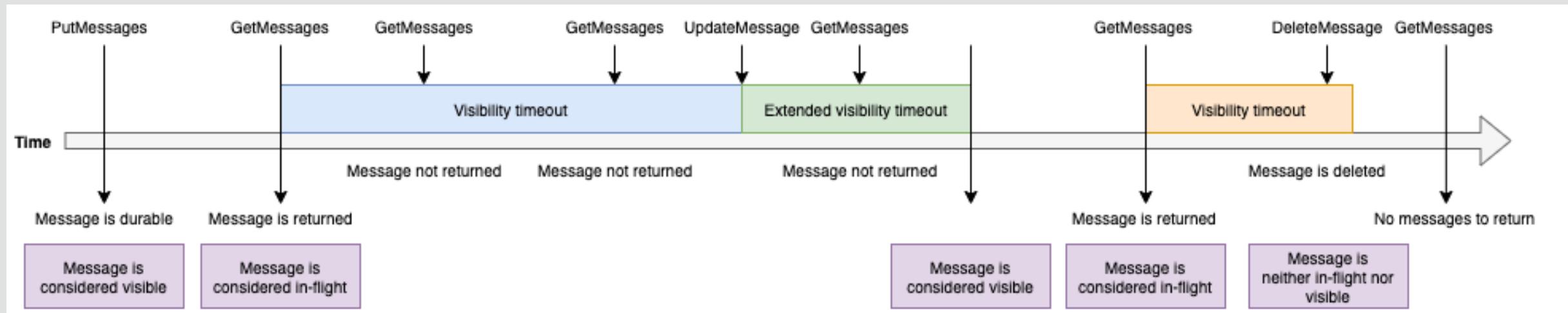
**Pay-as-you use pricing, simple
and minimal pricing
dimension⁺, no additional cost
for data transfer⁺**



**Multi-protocol,
open-API
STOMP support**

Support for **multi-protocol**
including REST and STOMP⁺

Message Lifecycle



- Message is hidden when being consumed for a default period
 - Message gets released if the consumer fails
 - Message can be hidden for longer if consumption takes longer than expected
- Message is removed with read receipt sent back to queue



OCI Queue

**Benefits &
Features**

OCI Queue - Benefits

- Application decoupling
- Reliable message processing
- Buffer and batch operations
- Auto scaling
- At-least-once message delivery
- Dead-letter-queue (DLQ)
- Technology and vendor agnostic
- Control of message visibility



OCI Queue - Benefits

➤ Spend Less

- **Reduce Admin Cost:** Eliminates expensive, tedious, manual monitoring and management
- **Reduce Runtime Cost:** Cloud pay-per-use, No more paying for provisioned capacity

➤ Reduce Risk

- Multi-protocol and open API: no tech restrictions with JMS API support and multi-protocol support for efficient data
- **Always Available:** 99.9% uptime guaranteed and fully fault tolerant
- **Enterprise workloads :** Runs all your enterprise workloads and reliably deliver messages

➤ Innovate more

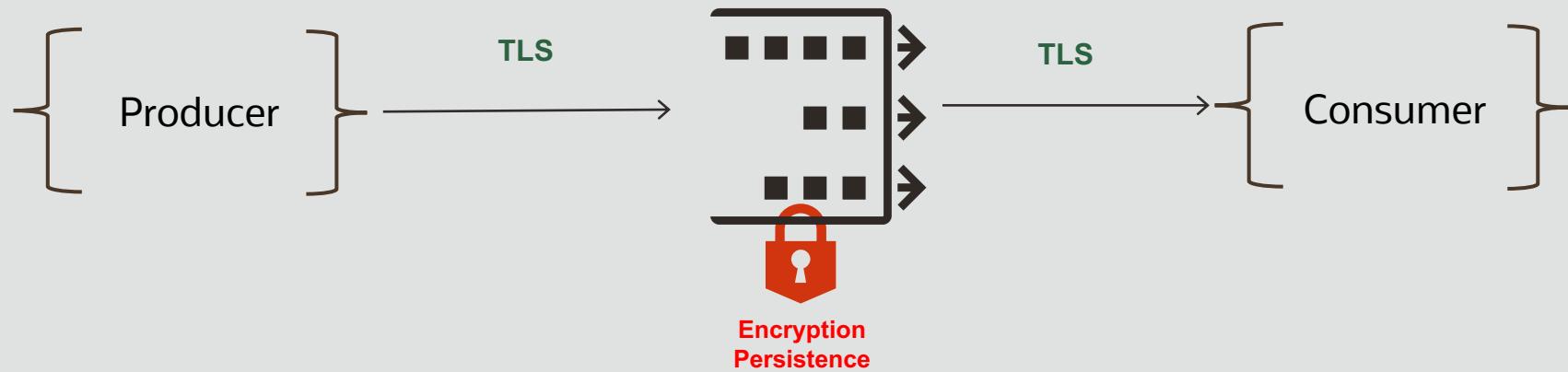
- **Develop faster:** Using numerous OCI SDKs
- **Refocus talent:** Focus on new projects, business logic, user experience, analytics

OCI Queue - Capabilities

- Auto scaling
- High Throughput
- At-Least-Once Delivery
- Best-Effort Ordering
- Message Locking
- Long Polling
- Batching
- Dead Letter Queue
- 7 days retention
- Queue Statistics
- Encryption
- STOMP support

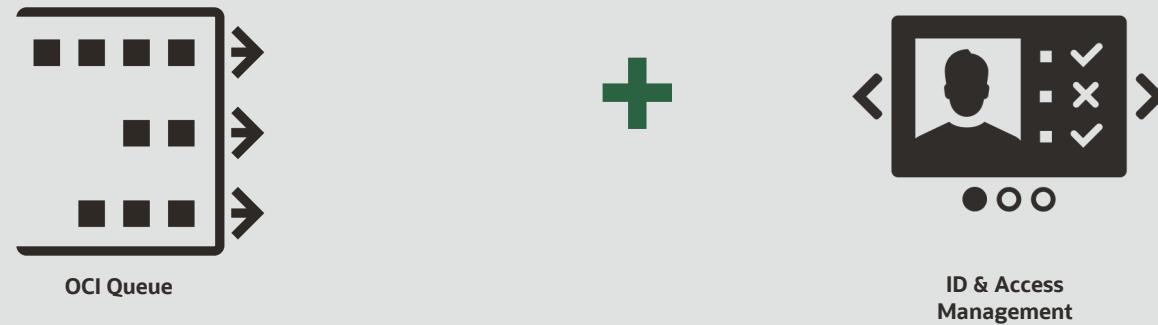


Security – Provides end-to-end Encryption



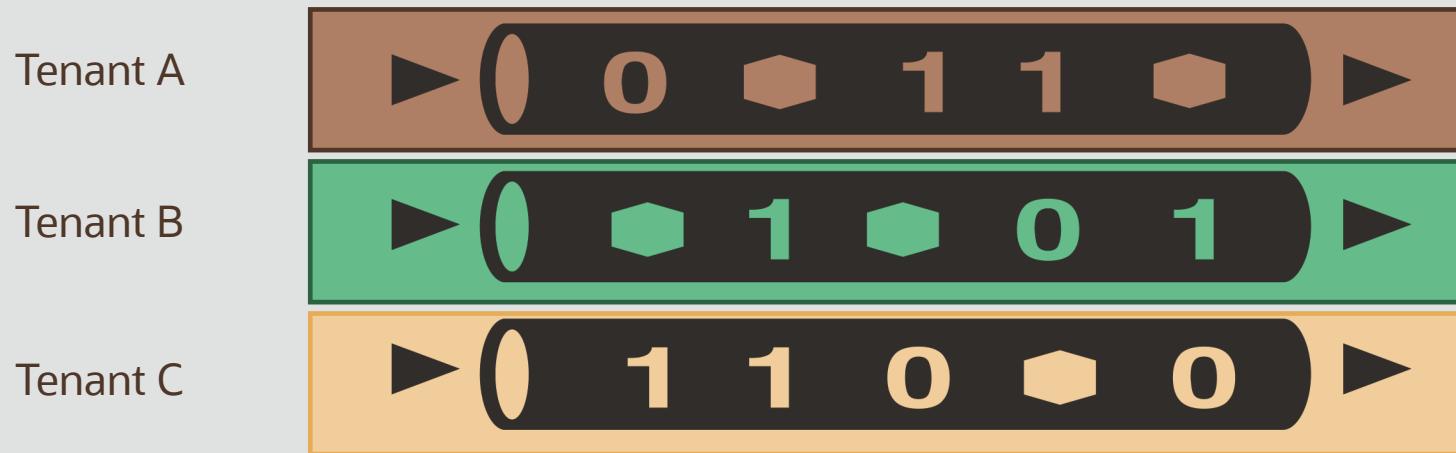
- Queue data is always encrypted – in transit and at rest
- Oracle automatically applies security updates - Monthly, or off-cycle for high-impact security vulnerability

Security – Fully Integrated with OCI IAM



- Queue is fully integrated with OCI AuthN and AuthZ
- Customers get fine grained access control with OCI IAM policies – from queue creation to producing and consuming queue data.

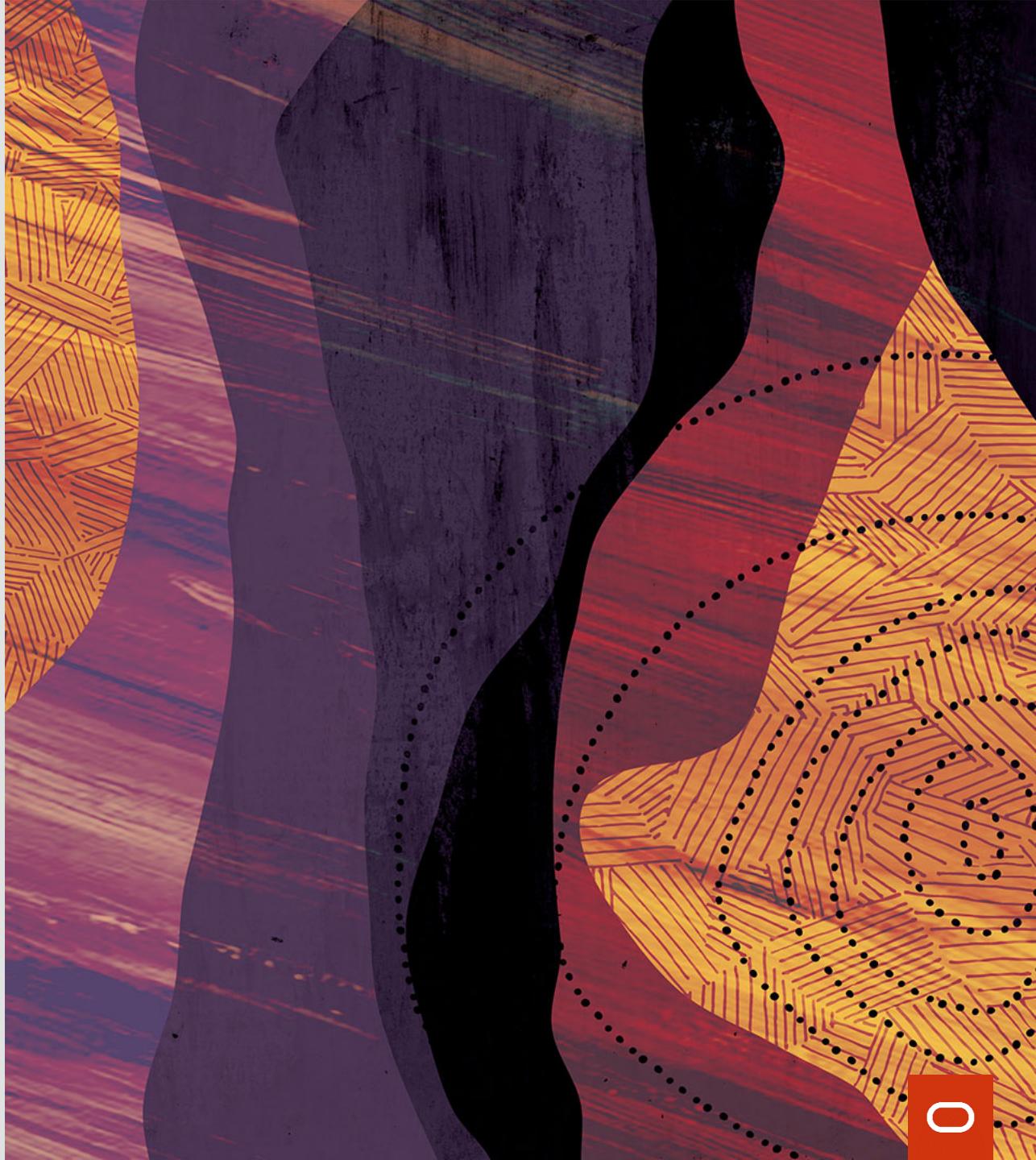
Security – Complete Isolation



- Queue service internal architecture guarantees complete tenancy level isolation of data.
- The service guarantees no noisy neighbor problems irrespective of usage

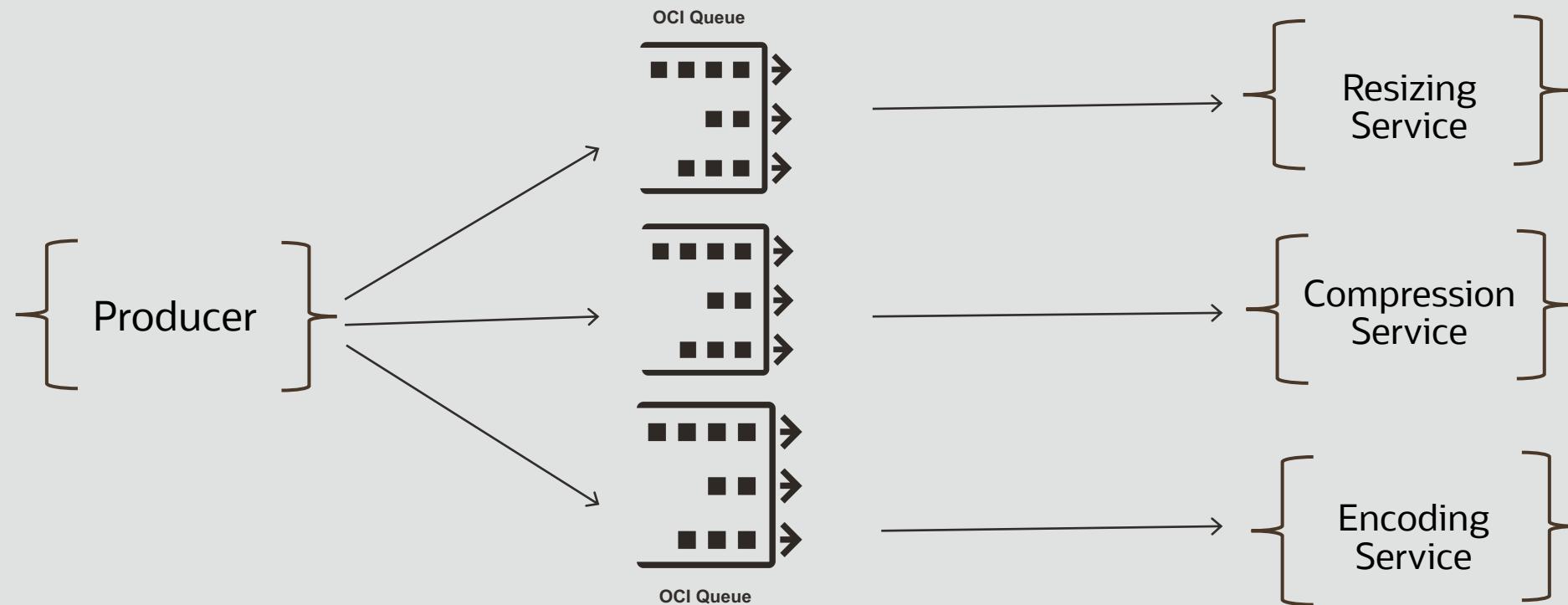
OCI Queue

Use Cases



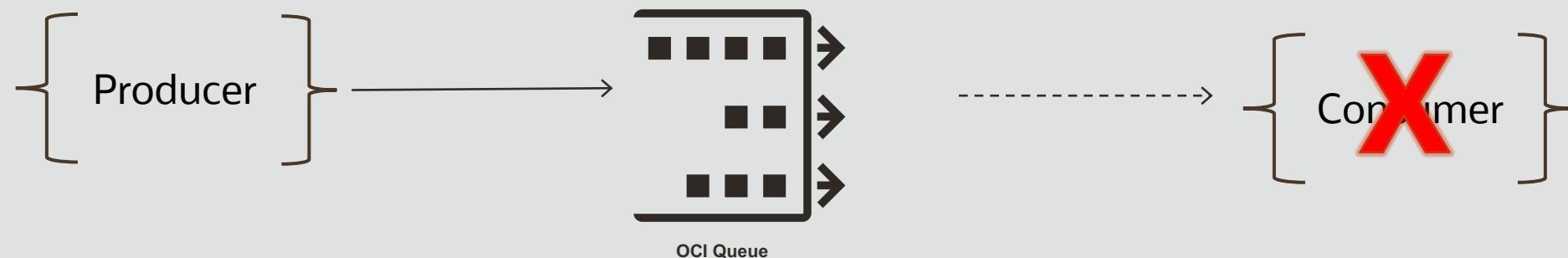
OCI Queue – Use Cases

Application decoupling: Decouple video processing workflow by separating video upload from backend complex processes e.g. encoding, resizing and compression



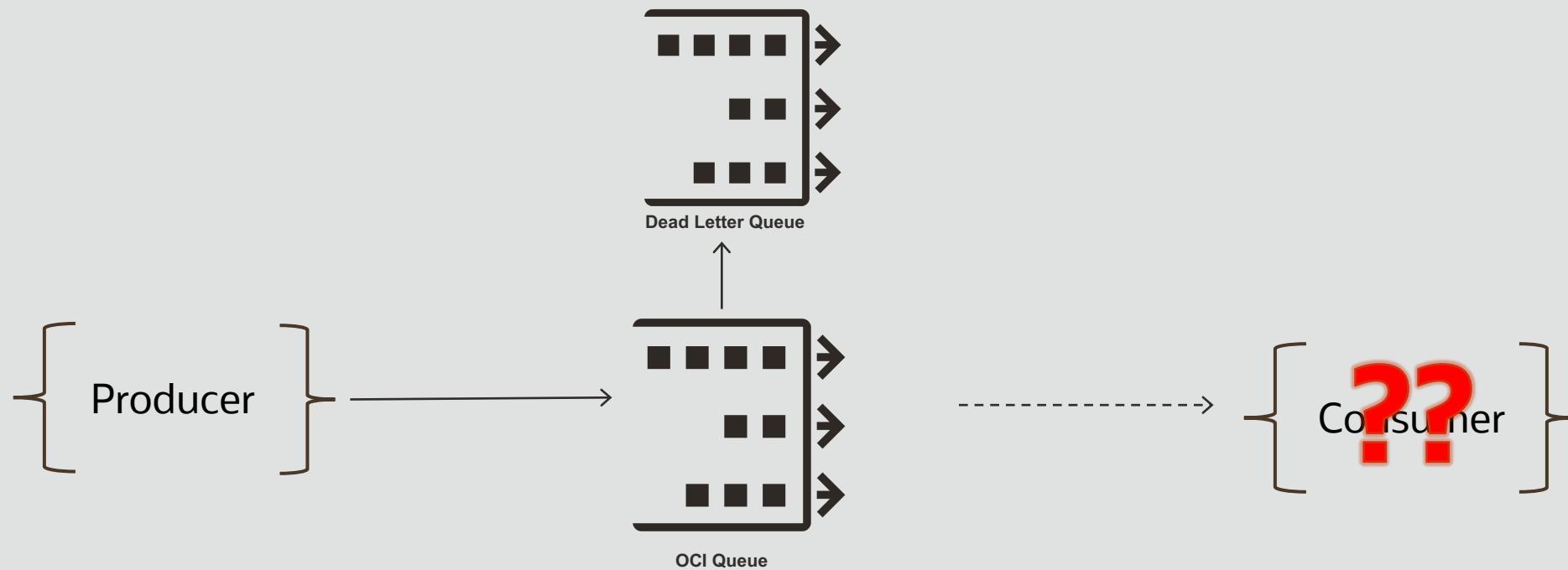
OCI Queue – Use Cases

Reliable message processing: Add resiliency to architecture e.g. an online-order processing system accepts an order even if the supply-chain component may be down



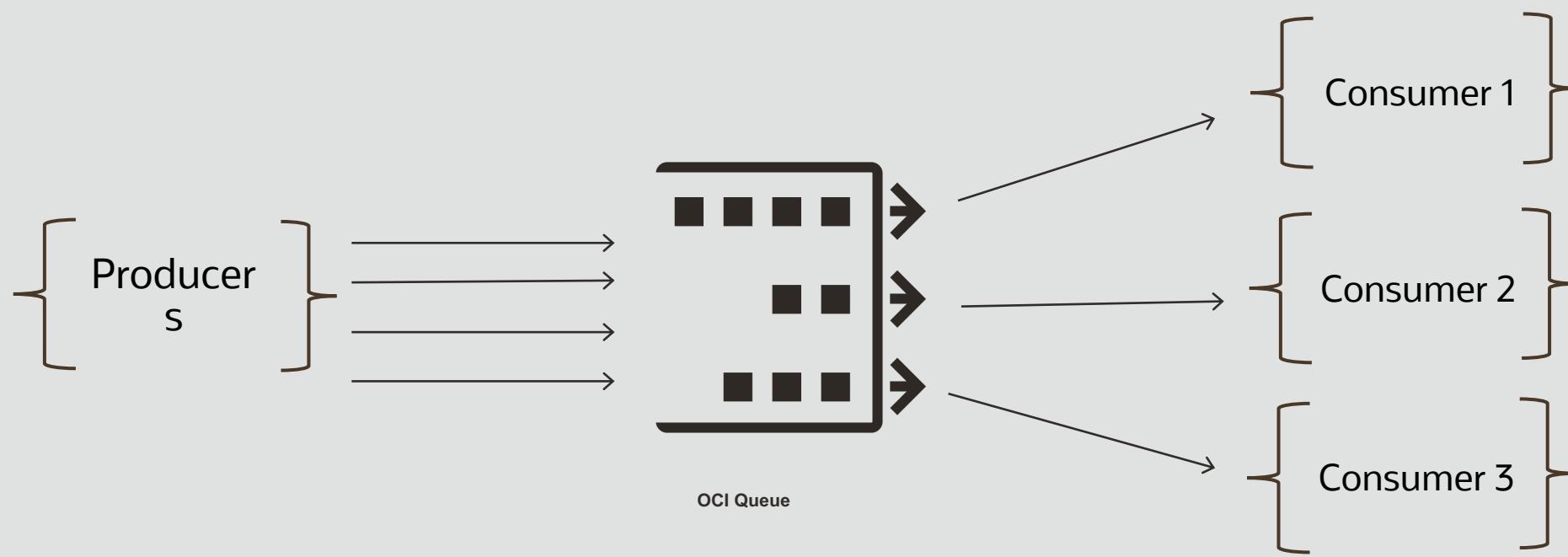
OCI Queue – Use Cases

Avoiding message loss: If a consumer keeps failing to consume a message we can ‘park it’ with a DLQ automatically.



OCI Queue – Use Cases

Parallel Processing: Allocate tasks to multiple worker nodes by e.g. a Search ad engine accepts high volume of ad requests



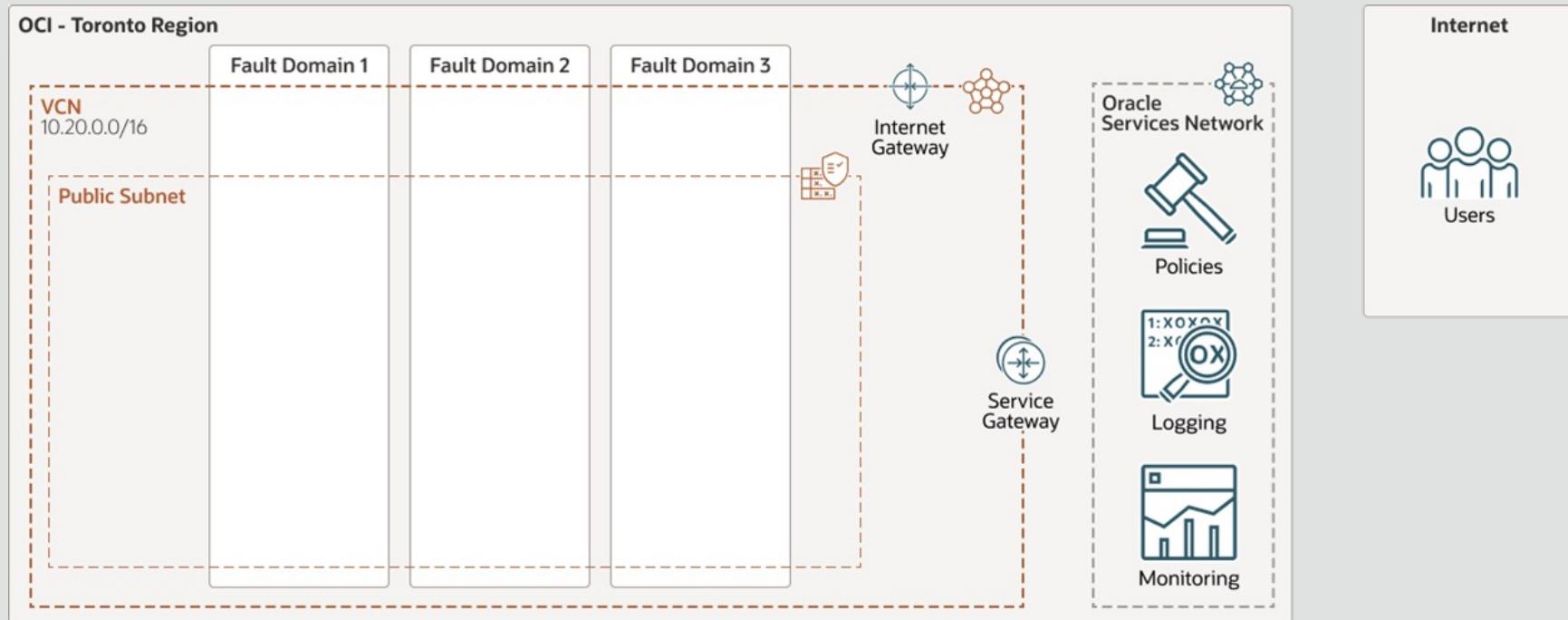
OCI Queue

Demo



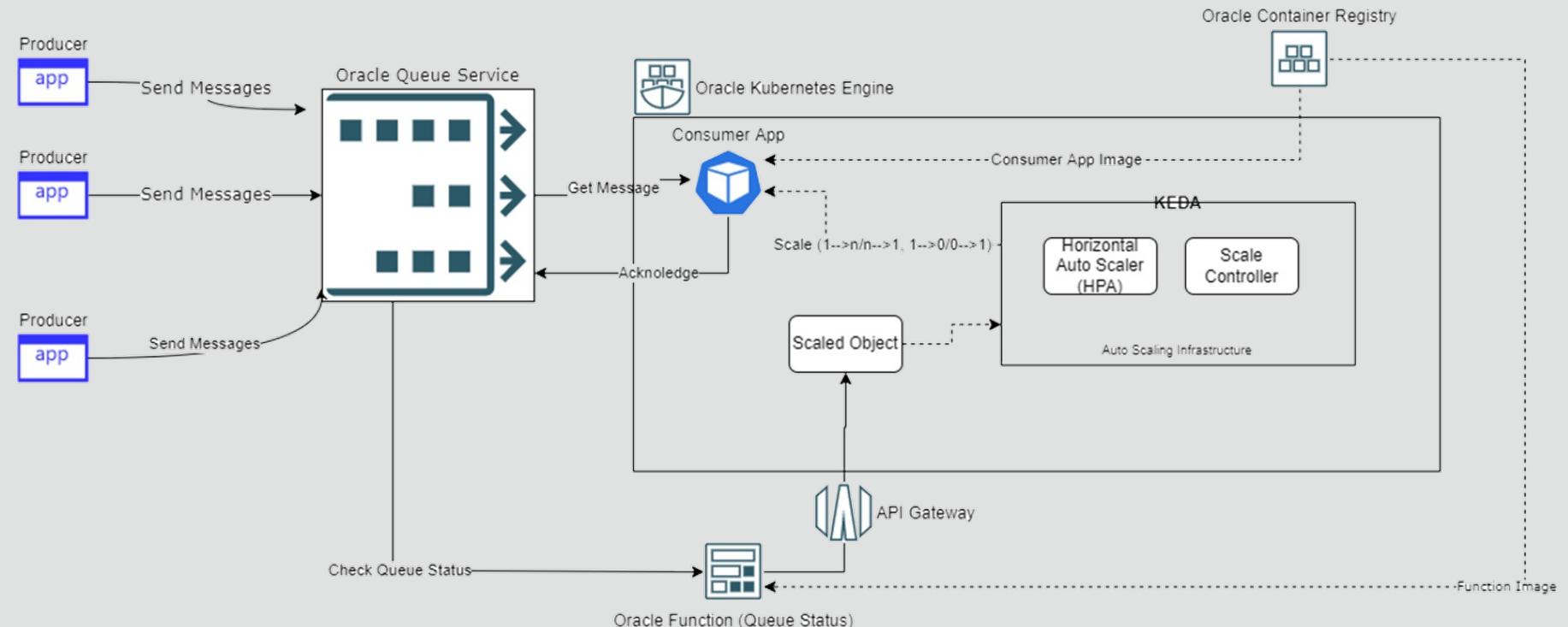
Simple Demo

- Everything can be done via a 1 file Java app
 - Just add authentication
 - (optionally OCID of Queue)
- No OCI deployments needed!
- Can be retrieved from <https://github.com/oracle-devrel/oci-arch-queue-demo>



Advanced Demo

- OCI Queue decouples Kubernetes backend from clients.
- OCI Function + KEDA used to manage Kubernetes scaling to match throughput
- Can be pulled from <https://github.com/oracle-devrel/oci-arch-queue-oke-demo>



Useful Resources

- Oracle Architecture Center (oracle.com/goto/ref-archs)
 - Playbooks
 - Decision Matrix on message technologies
(<https://docs.oracle.com/en/solutions/select-a-messaging-solution/index.html>)
- DevRel Github (oracle.com/goto/gh-devrel)
 - Simple Demo – Java/Groovy code for (covers queue creation, message creation & consumption). No build or deployment requirements.
 - Dynamic Scaling Demo – Using Kubernetes and Functions to consume messages and dynamically scale consumption.
 - Python demo with provider and consumer
- Oracle Learning Library / YouTube overview videos – search OCI Queue

Oracle Cloud Free Tier – Special Promo

Try Always Free. No Time Limits.

Always Free

Services you can use for unlimited time



30-Day Free Trial

Free credits you can use for additional OCI services

300\$ 500\$ in Oracle Cloud Credits

To be activated for this special promo:

- Join our Public Slack Workspace and contact me

Join our public Oracle DevRel Workspace

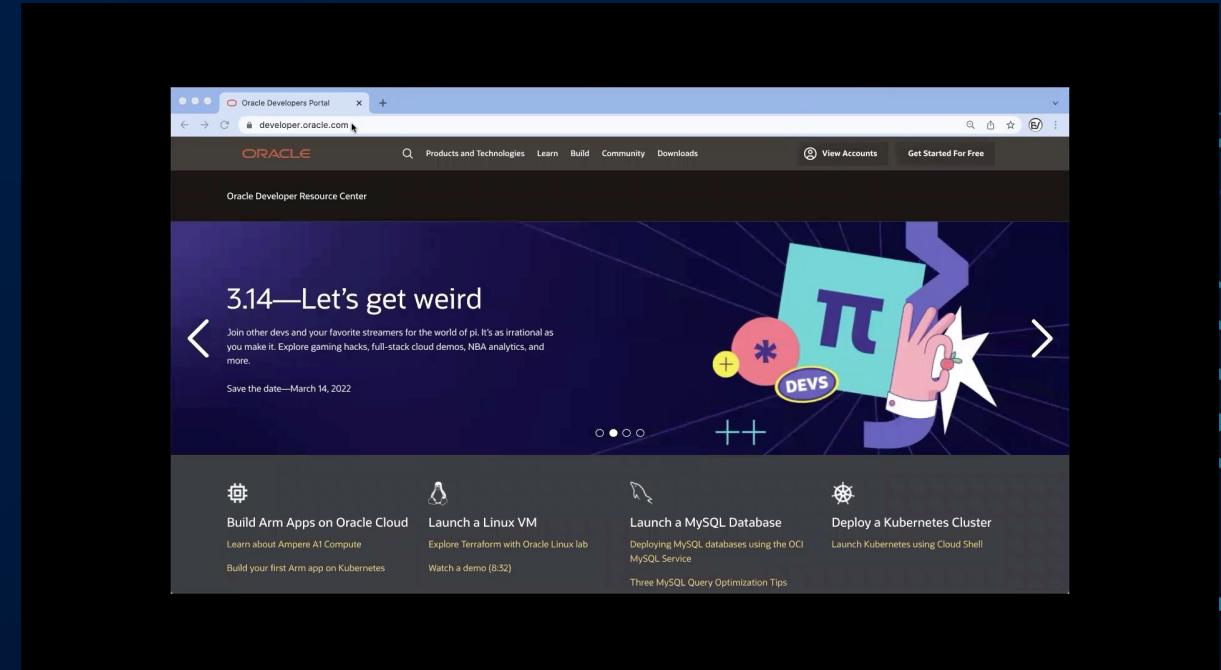


slack

oracledevrel.slack.com

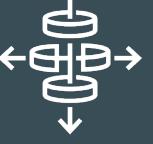
Join the dedicated Slack channel to be part of the conversation and raise your questions to our Experts:

Access the Slack OracleDevRel Workspace following this link:
<https://bit.ly/devrel-slack-emea>



OCI Architecture Center -- Free Content & More

URLS are [https://oracle.com/goto/...](https://oracle.com/goto/)

Reference Architectures  /ref-archs	Playbooks  /playbooks	Built & Deployed  /deployed	Live Labs  /labs	Tutorials  /tutorial	GitHub - DevRel  /gh-devrel
GitHub - Samples  /gh-samples	Learning Videos  /youtube	Developer  /dev	Open Source  /open	PaaS Community  /paas	GitHub - Oracle  /gh-oracle
Oracle Community  /community	Blogs  /blog	Apex  /apex	Cloud Customer Connect  /connect		

URLS are [https://oracle.com/goto/...](https://oracle.com/goto/)



Thank you