IBM MQ certified containers

Rob Parker
Security focal, Software Engineer
IBM MQ
parrobe@uk.ibm.com

Important disclaimers

IBM’s statements regarding its plans, directions and intent are subject to change or withdrawal without notice at IBM’s sole discretion. Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision. The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract. The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.

Important disclaimers

- **IBM Confidential**. Unless specifically advised otherwise, you should assume that all the information in this presentation (whether given in writing or orally) is IBM Confidential and restrict access to this information in accordance with the confidentiality terms in place between your organization and IBM.
- **Content Authority**. The workshops, sessions and materials have been prepared by IBM or the session speakers and reflect their own views. They are provided for informational purposes only, and are neither intended to, nor shall have the effect of being, legal or other guidance or advice to any participant. While efforts were made to verify the completeness and accuracy of the information contained in this presentation, it is provided AS-IS without warranty of any kind, express or implied. IBM shall not be responsible for any damages arising out of the use of, or otherwise related to, this presentation or any other materials. Nothing contained in this presentation is intended to, nor shall have the effect of, creating any warranties or representations from IBM or its suppliers or licensors, or altering the terms and conditions of the applicable license agreement governing the use of IBM software.
- **Performance**. Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.
- **Customer Examples**. Any customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics may vary by customer. Nothing contained in these materials is intended to, nor shall have the effect of, stating or implying that any activities undertaken by you will result in any specific sales, revenue growth or other results.
- **Availability**. References in this presentation to IBM products, programs, or services do not imply that they will be available in all countries in which IBM operates.

Trademarks

- IBM, IBM Cloud etc are trademarks of International Business Machines Corporation, registered in many jurisdictions

- Amazon Web Services, AWS and related names may be trademarks owns by AWS as described at https://aws.amazon.com/trademark-guidelines/

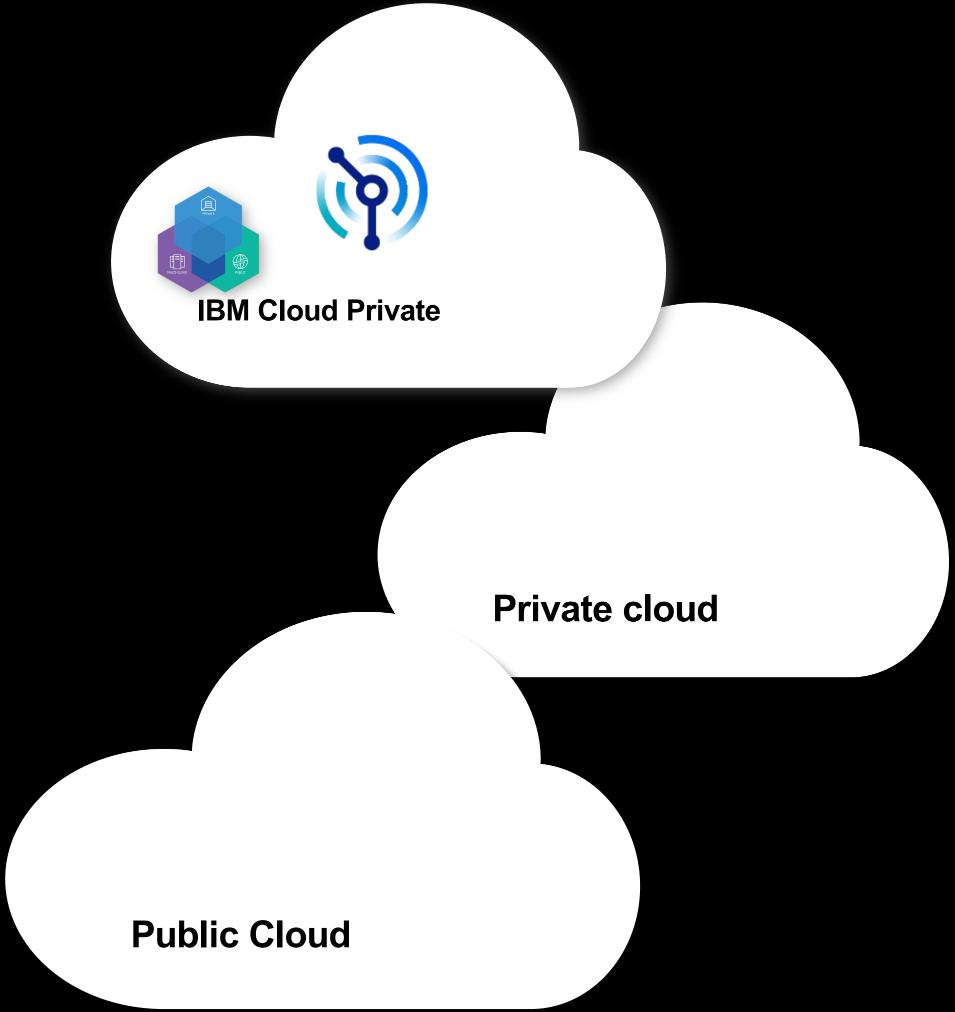
- Other company, product and service names may be trademarks, registered marks or service marks of their respective owners

- A current list of IBM trademarks is available on the web at "Copyright and trademark information" ibm.com/legal/copytrade.shtml

Introduction of Containers/cloud

Characteristics of a cloud environment

- **Self-service**
  - Empowers users to provision resources without requiring human intervention, most likely using a web-based portal or an API.
- **Elastic scaling**
  - Enables scaling up and down on demand, driving the need for high levels of automation.
- **Shared resources**
  - Offers economies of scale through the use of shared infrastructure and software, securely separating the resources at a logical level.
- **Metered usage**
  - Allows pay-as-you-go billing through monitoring, measurement and reporting of usage.

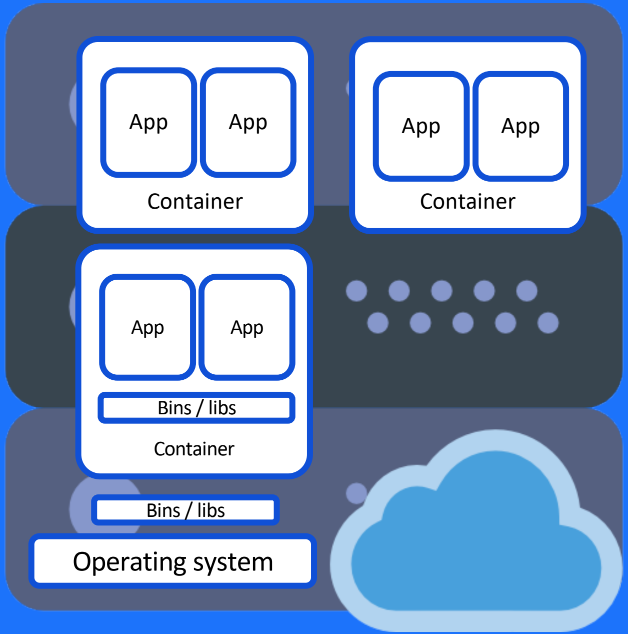


Containers Introduction

Containers

- Containers provide a similar environment to a VM but lighter in weight
  - ➢ A **virtual machine** provides an abstraction of the physical hardware
  - ➢ A **container** abstracts the OS level, typically at the user level

- Linux containers
  - ➢ Containers all share the same OS kernel
  - ➢ Images are constructed from layered filesystems
  - ➢ Containers isolate applications from each other and the underlying infrastructure



Benefits of a container strategy

Scalability and Infrastructure Optimization

Improved utilization of system resources when compared to virtualized isolation runtimes

Build Agility

Container build process reduces the effort for maintaining multiple runtime environments

Operational Consistency

Homogeneous administration of heterogeneous components, reducing the range of skillsets required to operate the environments

Fine-grained Resilience

Containers are disposable. Failing containers are removed and replaced with new instances, remove the need to nurture

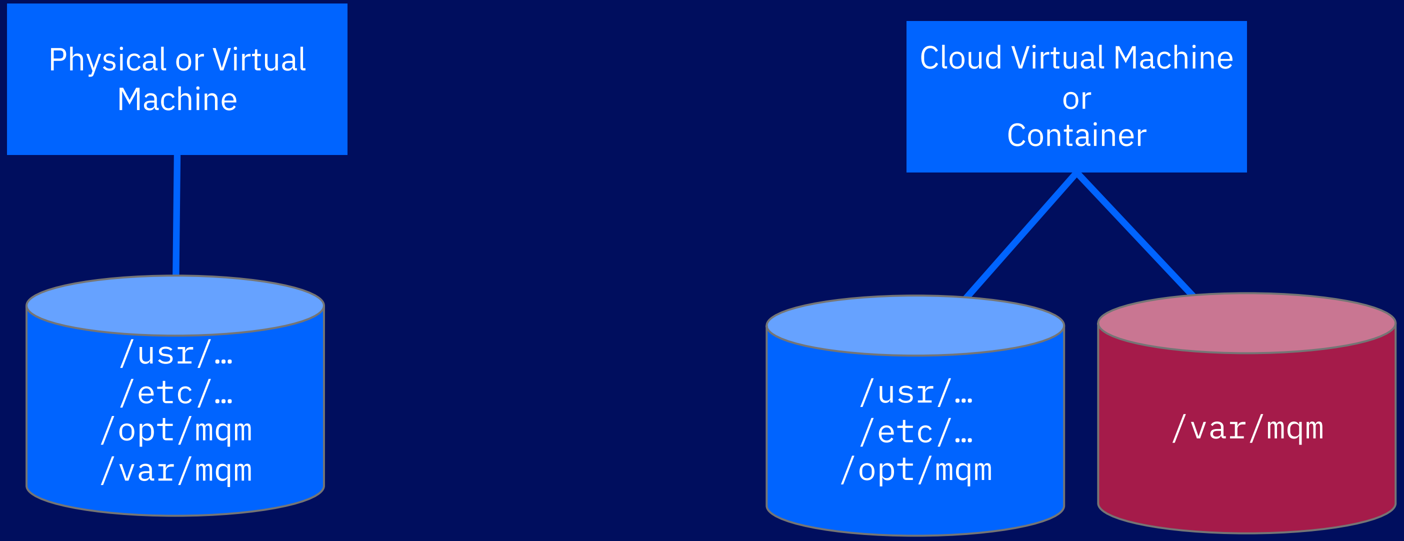
Component Portability

Containers reduce the barrier to moving components between environments (on-premise, public & private clouds)

Team Productivity

Accelerated development, improved consistency across environments, empowering autonomous teams improving productivity and quality

Separate storage from compute



Contents of `/var/mqm` is populated at MQ installation time

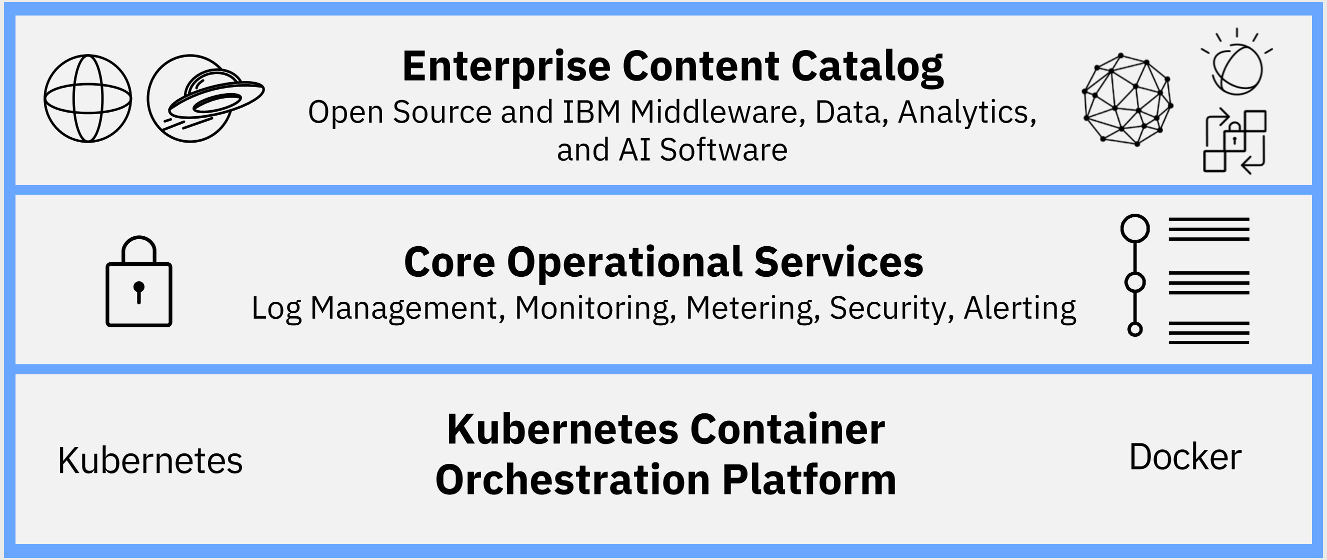
Contents of `/var/mqm` is populated by running `crtmqdir` at runtime.

What is Kubernetes ?

Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation. It has a large, rapidly growing ecosystem. Kubernetes services, support, and tools are widely available.

- Can be used to deploy containers and their resources.
  - e.g. Persistent volumes, load balancers, secrets.
- Containers are deployed as “pods” which can contain multiple containers
  - Pods can be placed on any “node” and be moved between nodes
- Multiple pods can be deployed either singularly or via “Sets”
  - E.g. StatefulSet or ReplicaSet
- Network access is provided by “Services”

IBM Cloud Private



Enterprise Content Catalog

Open Source and IBM Middleware, Data, Analytics, and AI Software

Core Operational Services

Log Management, Monitoring, Metering, Security, Alerting

Kubernetes

Kubernetes Container Orchestration Platform

Docker

Choose your infrastructure:

Power Systems

OpenStack

IBM Spectrum

IBM Z

VMWare

Intel

Microsoft Azure

Amazon Web Services

Google Cloud

Strategic Value:

Reduced infrastructure, license, and maintenance costs

Self-service catalog

Agility, scalability, and elasticity

Self-healing

Enterprise security

No vendor lock-in

IBM Cloud Pak for Integration

Taking IBM’s market leading integration capabilities and adding value to become one simple, fast, and secure integration experience

- **Most powerful integration platform on the market**

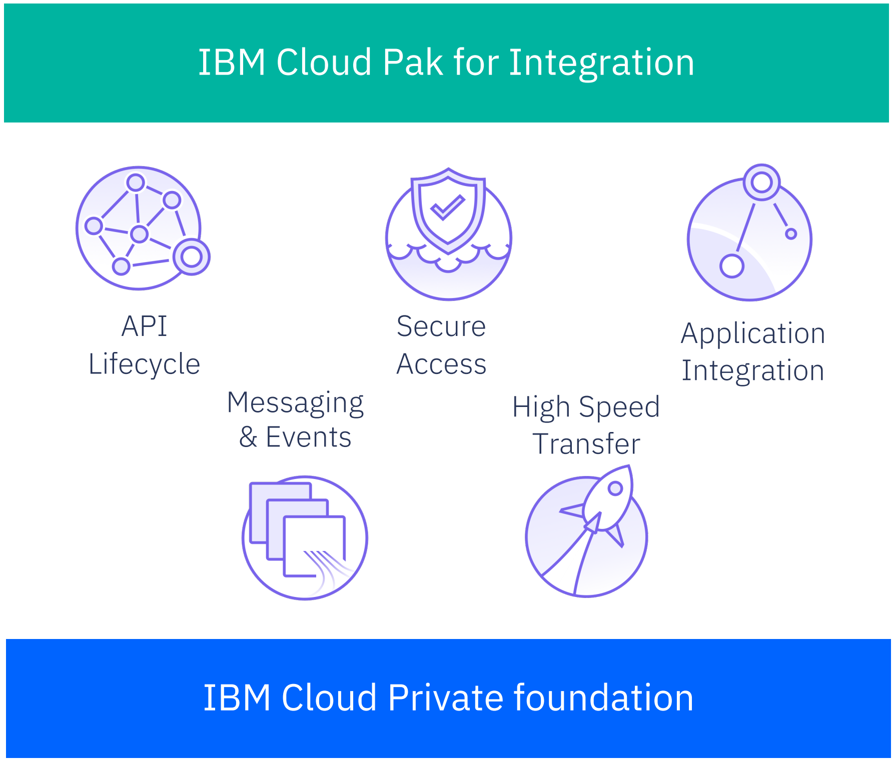
  NEW offering incorporating traditional and modern integration including APIs, App Integration, Message queuing, Event streams and Fast file transfer

- **Deploy wherever needed**

  Supports deployment on-premises or in any cloud

- **Enterprise grade**

  Secure, scalable modern architecture



MQ Modernization

Containerization facilitates the modernization of MQ deployments.

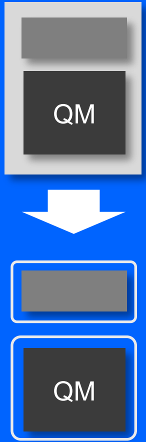
(These pattern also apply outside of containers)

IBM Cloud Transformation Advisor

Analyses your queue managers and JEE applications for suitability for moving to IBM containers

Replatform

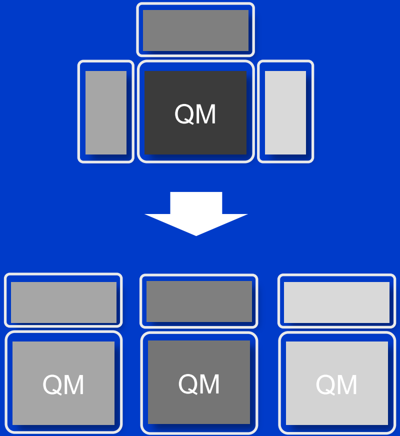
establishing the container orchestration platform, services and capabilities to succeed, and move to a runtime topology that is native to the platform



Containerize MQ queue managers, with applications connected as clients

Repackage

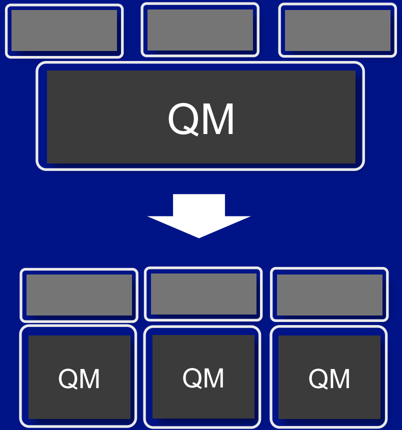
break down the existing artefacts so that they are bounded along line of business and development teams to improve the agility of the organization



Queue managers are dedicated to an application

Refactor

re-work the artefacts that are hard to maintain or prevent the organization from realising the full benefits of their modernization journey



Deploy MQ patterns that provide horizontal scaling and continuous availability

IBM MQ in a container

MQ in Containers

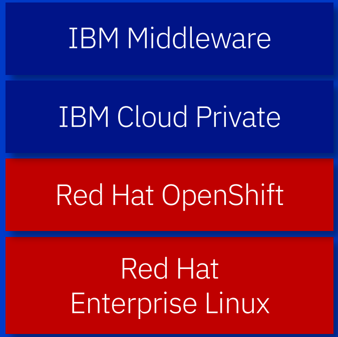
MQ has been supporting Docker containers since 2015 with images on Docker Hub and Docker Store and sample setups on Github

github.com/ibm-messaging/mq-container

MQ Advanced is available as fully supported IBM certified containers with **IBM Cloud Private** and the **IBM Kubernetes Service** on **IBM Cloud**

PRIVATE
MULTI-CLOUD
PUBLIC

Deploy fully supported IBM certified software containers into an IBM provided **Kubernetes** platform or an existing **Red Hat OpenShift** environment

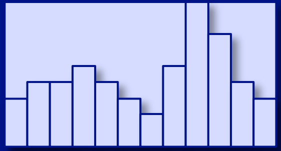

IBM Middleware

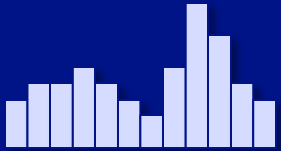
IBM Cloud Private

Red Hat OpenShift

Red Hat Enterprise Linux

IBM has introduced the ability to purchase an entitlement based on the container size in Virtual Processor Cores and the number of hours that MQ was deployed in each container


Traditional licensing


Hourly licensing

MQ is supported in containers

- MQ V8.0.0.4 onwards is supported in Docker V1.6+
- MQ V9.1.0.0 onwards is supported in Docker V1.12+
- IBM recommends using either MQ V9.1 LTS, or MQ V9 Continuous Delivery releases
  - Adds web console
  - Adds REST APIs
  - Easier storage management (crtmqdir)
  - Quicker to receive new features

- IBM will support MQ issues, agnostic to the orchestration environment
- The orchestration vendor will need to support and provide assistance for orchestration issues

MQ container orchestration support

★ **IBM MQ certified container**
Supported image and Helm chart available

● **MQ supported with sample**
Supported, and you need to build your own image (samples/blog available)

▲ **MQ supported with no sample**
Supported, and you need to build your own image.

✖ **Not supported in containers**

	Standalone container	IBM Cloud Kubernetes Service	IBM Cloud Private			IBM Cloud Private on Red Hat OpenShift	Red Hat OpenShift	Microsoft Azure Kubernetes Service	Amazon Elastic Kubernetes Service
Component/arch	*	x86_64	x86_64	ppc64le (POWER)	s390x (z/Linux)	x86_64	x86_64	x86_64	x86_64
MQ Advanced Server	●	★	★	★	★	★	★ (9.1.4)	▲	▲
MQ Server	▲	▲	▲	▲	▲	▲	▲	▲	▲
MQ Clients	▲	▲	▲	▲	▲	▲	▲	▲	▲
MFT Agent	▲	▲	▲	▲	▲	▲	▲	▲	▲
Salesforce Bridge	●	▲	▲	▲	▲	▲	▲	▲	▲
Blockchain Bridge	✖	✖	✖	✖	✖	✖	✖	✖	✖
RDQM	✖	✖	✖	✖	✖	✖	✖	✖	✖
Internet Pass-Thru	●	▲	▲	▲	▲	▲	▲	▲	▲

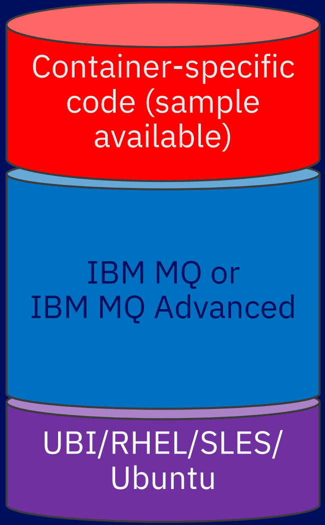
Self-build

MQ Advanced for Developers

MQ Advanced certified container

MQ Advanced certified container for Cloud Pak for Integration

Self-build

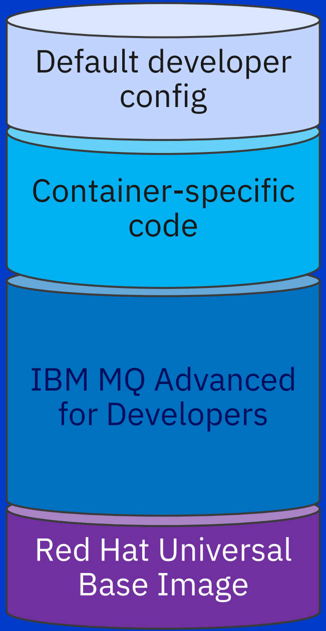


Container-specific code (sample available)

IBM MQ or IBM MQ Advanced

UBI/RHEL/SLES/ Ubuntu

MQ Advanced for Developers



Default developer config

Container-specific code

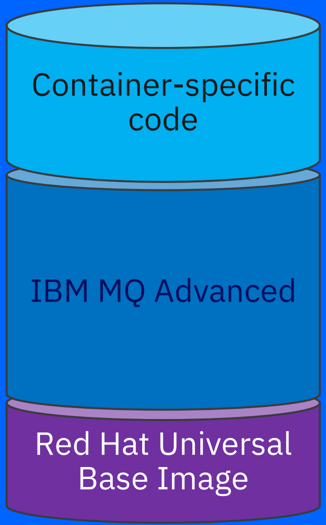
IBM MQ Advanced for Developers

Red Hat Universal Base Image

+

ibm-mqadvanced-server-dev Helm chart

MQ Advanced certified container



Container-specific code

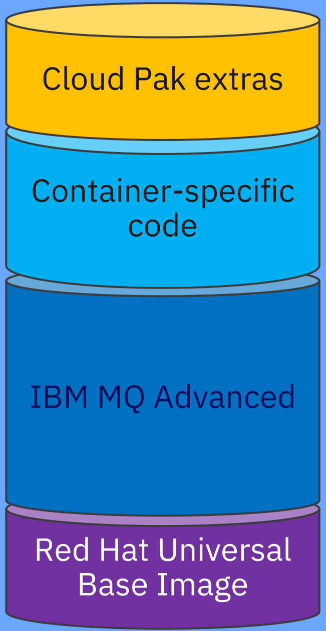
IBM MQ Advanced

Red Hat Universal Base Image

+

ibm-mqadvanced-server-prod Helm chart

MQ Advanced certified container for Cloud Pak for Integration



Cloud Pak extras

Container-specific code

IBM MQ Advanced

Red Hat Universal Base Image

+

ibm-mqadvanced-server-integration-prod Helm chart

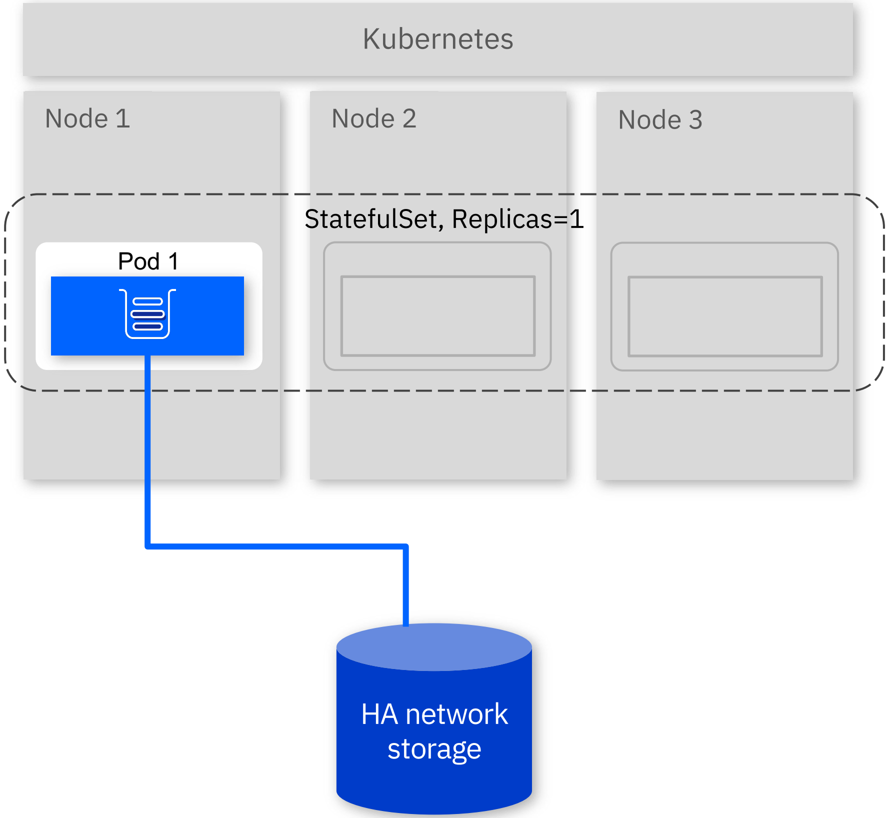
HA considerations

High availability with Kubernetes

The RDQM solution does not apply to container environments

High availability of the MQ data requires highly available replicated storage

Container orchestrators such as Kubernetes handle much of the monitoring and restart responsibilities…



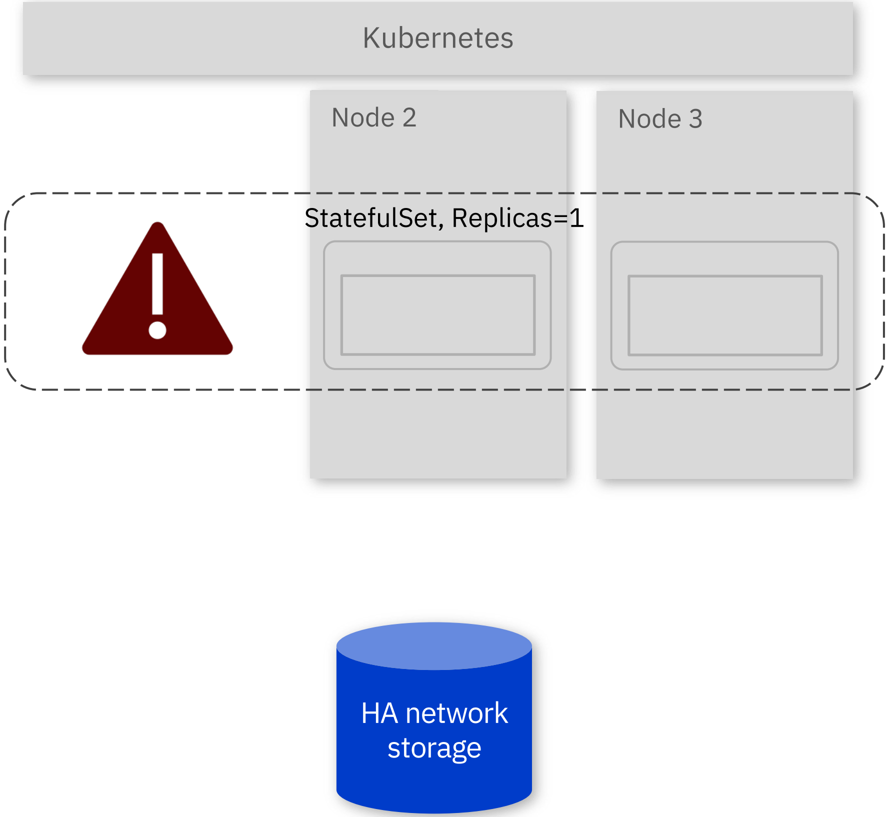
High availability with Kubernetes

The RDQM solution does not apply to container environments

High availability of the MQ data requires highly available replicated storage

Container orchestrators such as Kubernetes handle much of the monitoring and restart responsibilities…

…but not all. StatefulSets such as MQ are not automatically restarted following a Kubernetes node failure



High availability with Kubernetes

IBM MQ 9.1.3 CD

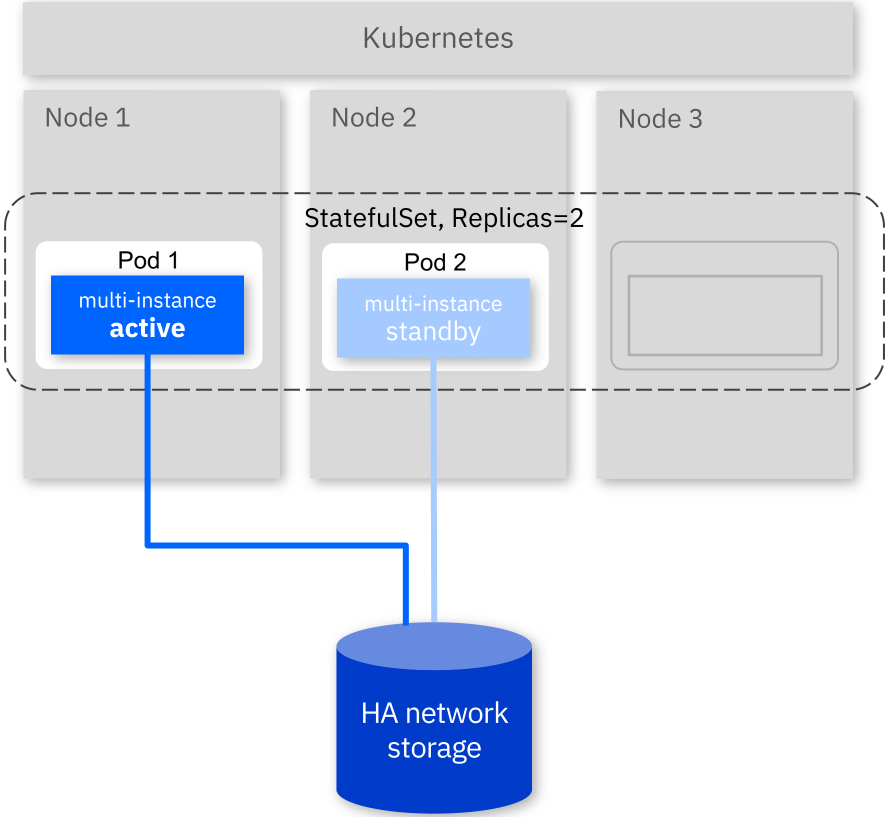
The RDQM solution does not apply to container environments

High availability of the MQ data requires highly available replicated storage

Container orchestrators such as Kubernetes handle much of the monitoring and restart responsibilities…

…but not all. StatefulSets such as MQ are not automatically restarted following a Kubernetes node failure

The MQ container image and Certified Container now supports a two-replica multi-instance queue manager deployment pattern to handle Kubernetes node failures



Storage providers

IBM Cloud File Storage

- Backed by NFS V4
- Not replicated between zones

IBM Spectrum Scale

- Older versions (GPFS V4.0) have been successfully tested with MQ

AWS Elastic File System (EFS) ⚠

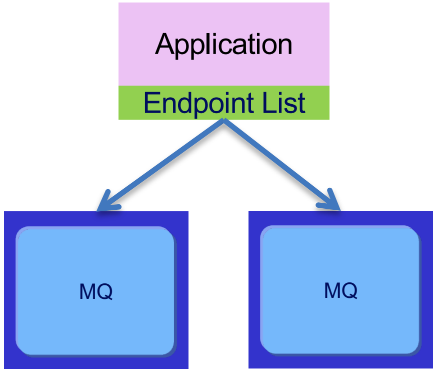
- Backed by NFS V4
- Replicated between zones
- Limited to 256 locks — ensure this meets your scalability needs

GlusterFS ⚠

- V3, V4 and V5 do not meet POSIX standards around locking (e.g. canceling a thread does not release locks)

Connection Routing

Static Routing



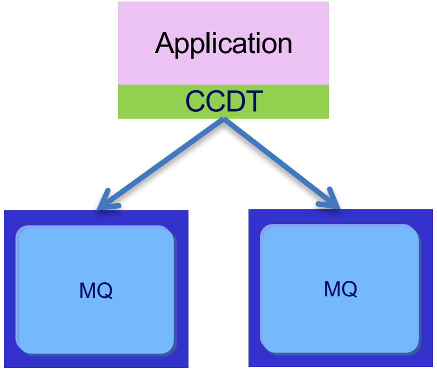
Client embeds endpoints

Performance impact when primary unavailable

Brittle configuration

No load balancing

Client Connection Definition Table

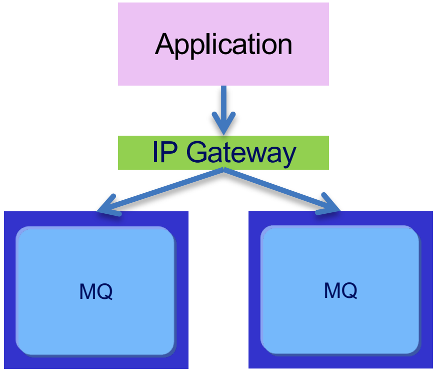


Client references endpoints

Enhanced Workload Management Strategies

Central configuration management

Load Balancer



Client references endpoints

Enhanced Workload Management Strategies

Central configuration management

Not recommended for JMS

Recommended Routing

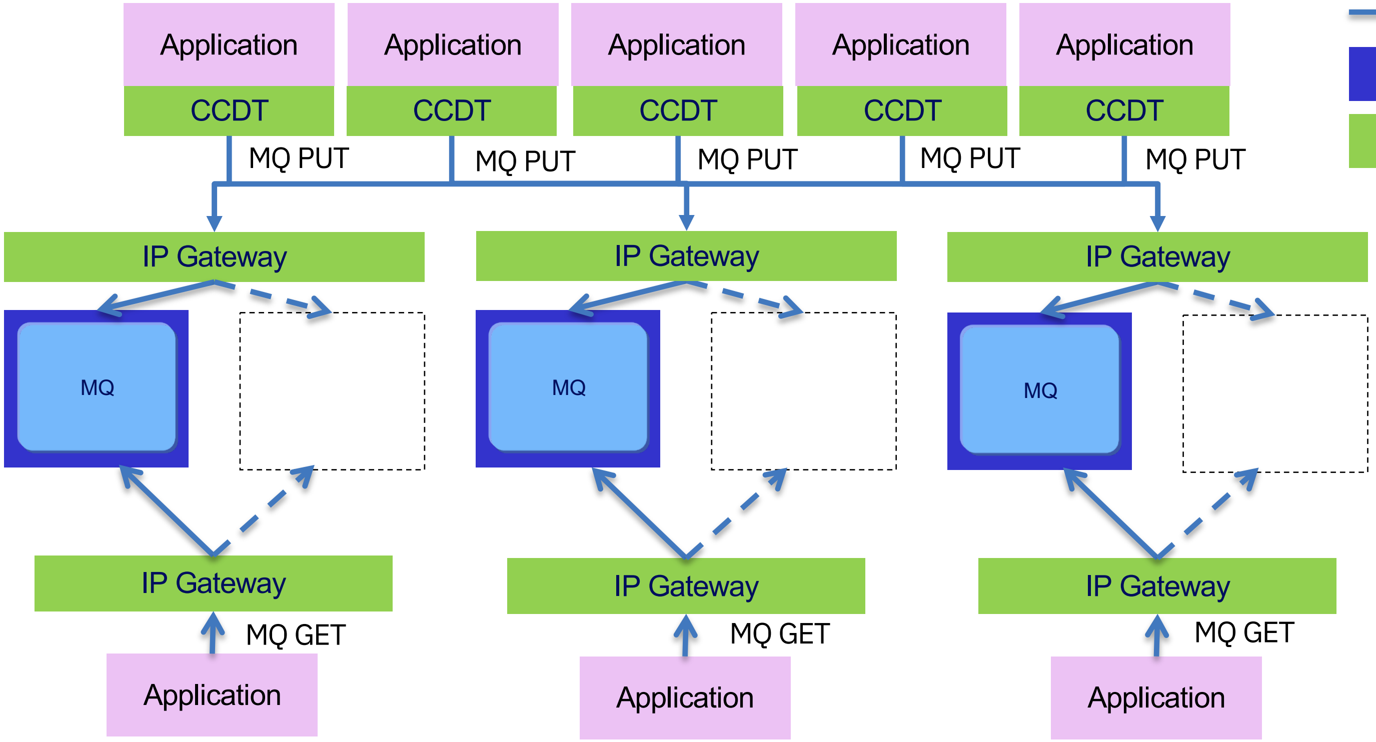
🟦 Messaging Layer

🟪 Application

— Network Connection

🟦 Container(s)

🟩 Connection Routing



Features of IBM MQ Certified Containers

Integration with ICP4I features

Catalog

- Provides quick deployment
- Hundreds of IBM and non-IBM products available
- Can import custom products for own use

Logging

- Central location for all log output
- Searchable
- Uses Kibana technology
- Queue Manager logs mirrored to this service.

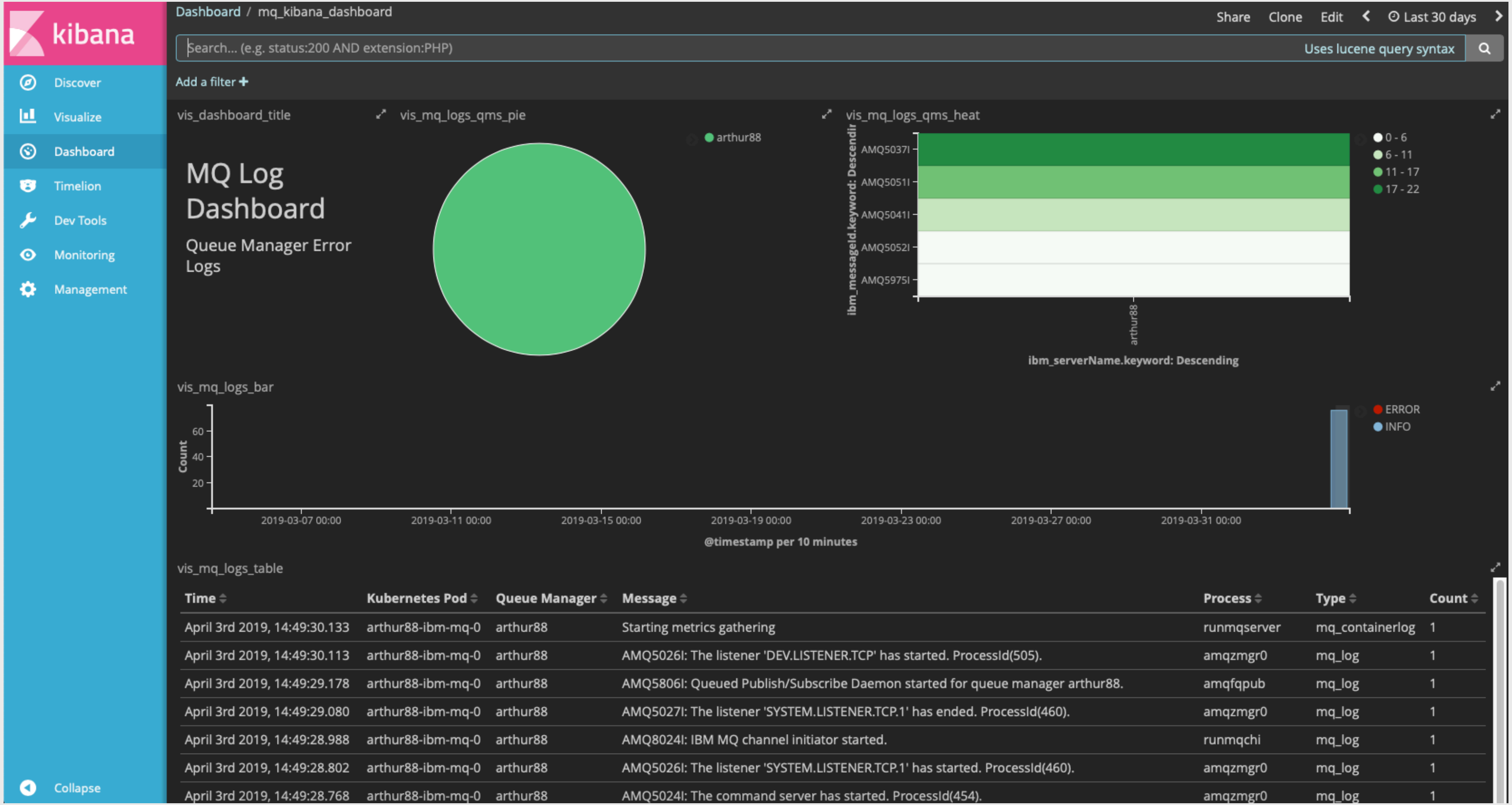
Monitoring

- Central location for monitoring metrics
- Searchable
- Custom graphs
- Uses Grafana technology
- IBM MQ Monitoring metrics sent to service.

Metering

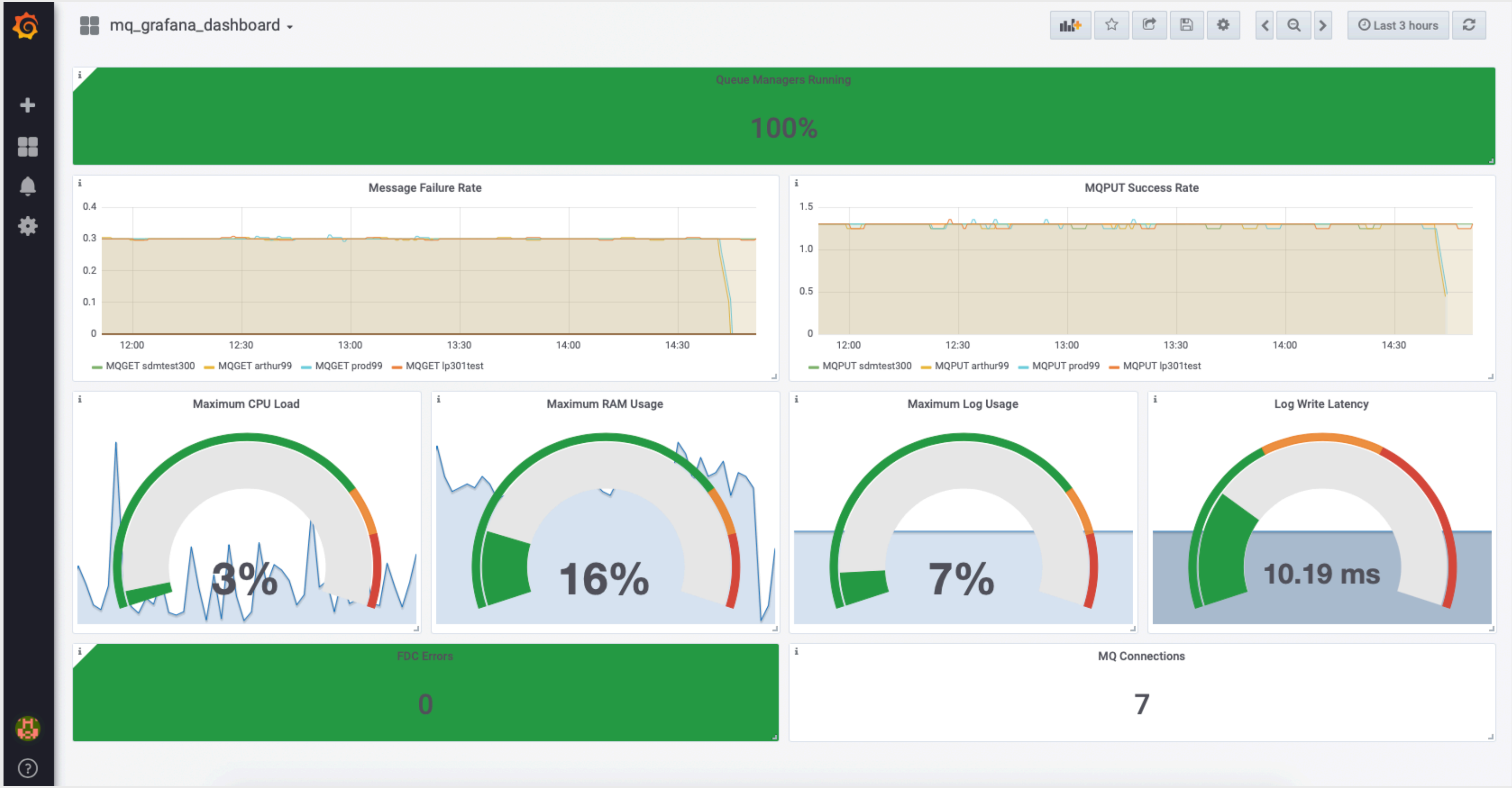
- Monitors up time & CPU cost of deployments
- Can be used for licensing (hourly license model)
- Searchable on deployment and version.

Sample Kibana dashboard



Time	Kubernetes Pod	Queue Manager	Message	Process	Type	Count
April 3rd 2019, 14:49:30.133	arthur88-ibm-mq-0	arthur88	Starting metrics gathering	runmqserver	mq_containerlog	1
April 3rd 2019, 14:49:30.113	arthur88-ibm-mq-0	arthur88	AMQ5026I: The listener 'DEV.LISTENER.TCP' has started. ProcessId(505).	amqzmgr0	mq_log	1
April 3rd 2019, 14:49:29.178	arthur88-ibm-mq-0	arthur88	AMQ5806I: Queued Publish/Subscribe Daemon started for queue manager arthur88.	amqfqpub	mq_log	1
April 3rd 2019, 14:49:29.080	arthur88-ibm-mq-0	arthur88	AMQ5027I: The listener 'SYSTEM.LISTENER.TCP.1' has ended. ProcessId(460).	amqzmgr0	mq_log	1
April 3rd 2019, 14:49:28.988	arthur88-ibm-mq-0	arthur88	AMQ8024I: IBM MQ channel initiator started.	runmqchi	mq_log	1
April 3rd 2019, 14:49:28.802	arthur88-ibm-mq-0	arthur88	AMQ5026I: The listener 'SYSTEM.LISTENER.TCP.1' has started. ProcessId(460).	amqzmgr0	mq_log	1
April 3rd 2019, 14:49:28.768	arthur88-ibm-mq-0	arthur88	AMQ5024I: The command server has started. ProcessId(454).	amqzmgr0	mq_log	1

Sample Grafana dashboard

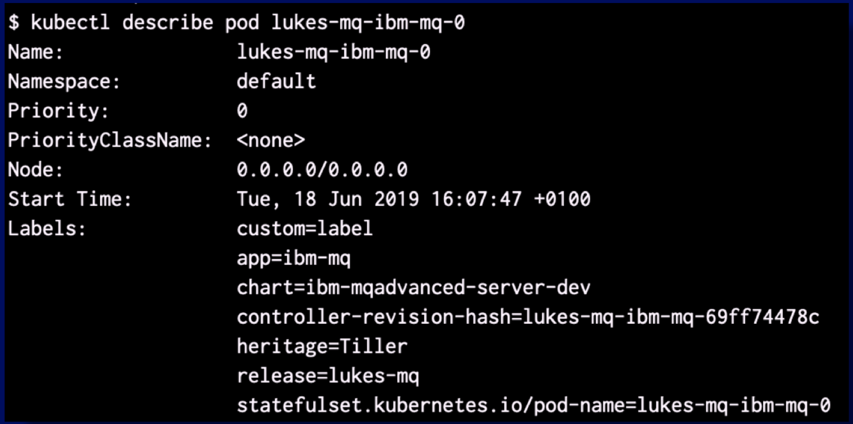


Non-Root

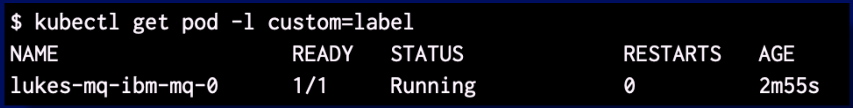
- 9.1.3: Container now runs as the “mqm” user rather than root, with a fixed UID and GID
- Allows for running under a tighter security policy
- Currently able to run under an ‘anyuid’ style Pod Security Policy… More on this later!

Custom Kubernetes Labels

- Kubernetes uses key/value labels on resources to specify attributes that are meaningful/relevant to users
- This release brings functionality to add custom labels to Queue Manager resources (pod, stateful set, PVC, etc)
- Necessary for connection routing as part of scaling
- Useful for creating custom resource queries



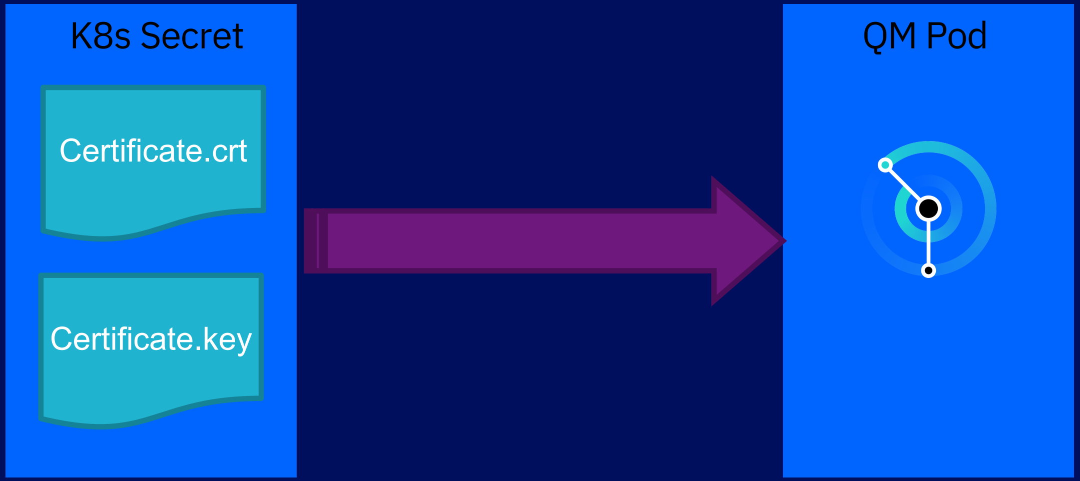
```
$ kubectl describe pod lukes-mq-ibm-mq-0
Name:                lukes-mq-ibm-mq-0
Namespace:           default
Priority:            0
PriorityClassName:   <none>
Node:                0.0.0.0/0.0.0.0
Start Time:          Tue, 18 Jun 2019 16:07:47 +0100
Labels:              custom=label
                     app=ibm-mq
                     chart=ibm-mqadvanced-server-dev
                     controller-revision-hash=lukes-mq-ibm-mq-69ff74478c
                     heritage=Tiller
                     release=lukes-mq
                     statefulset.kubernetes.io/pod-name=lukes-mq-ibm-mq-0
```



```
$ kubectl get pod -l custom=label
NAME                 READY    STATUS          RESTARTS   AGE
lukes-mq-ibm-mq-0    1/1      Running         0          2m55s
```

TLS

- You can now supply PKCS#8 & PKCS#1 unencrypted PEM files to add certificates to your Console & queue manager.
- Multiple certificates and keys can be provided with labels to use.
- First alphabetically will be set as default certificate.
- Uses Kubernetes Secrets to stored Certificate and keys.



IBM MQ in containers with App Connect Enterprise

Why does ACE use MQ?

1. As an asynchronous messaging provider
2. As a co-coordinator for global (two phase commit) transactions

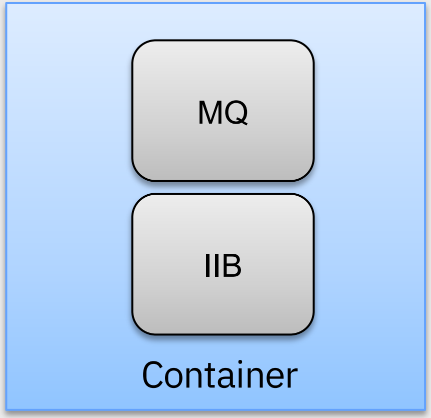
But it needs to connect to a local MQ Server

Running MQ and ACE in containers

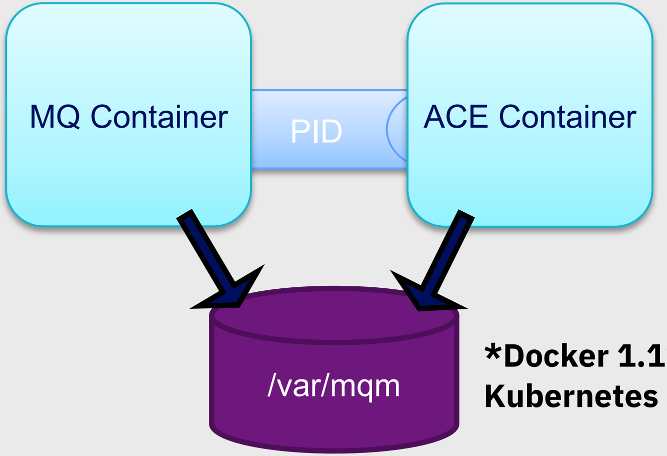
For cases where ACE can use Client bindings connections then you can run MQ and ACE in separate containers.

For cases where ACE requires local (server) bindings connections then you either:

Embedded in same container



Separate containers with shared PID Namespaces*



***Docker 1.12**
Kubernetes 1.10 Alpha

https://www.ibm.com/support/knowledgecenter/SSFKSJ_9.0.0/com.ibm.mq.con.doc/q114488_.html

Thank you

IBM®