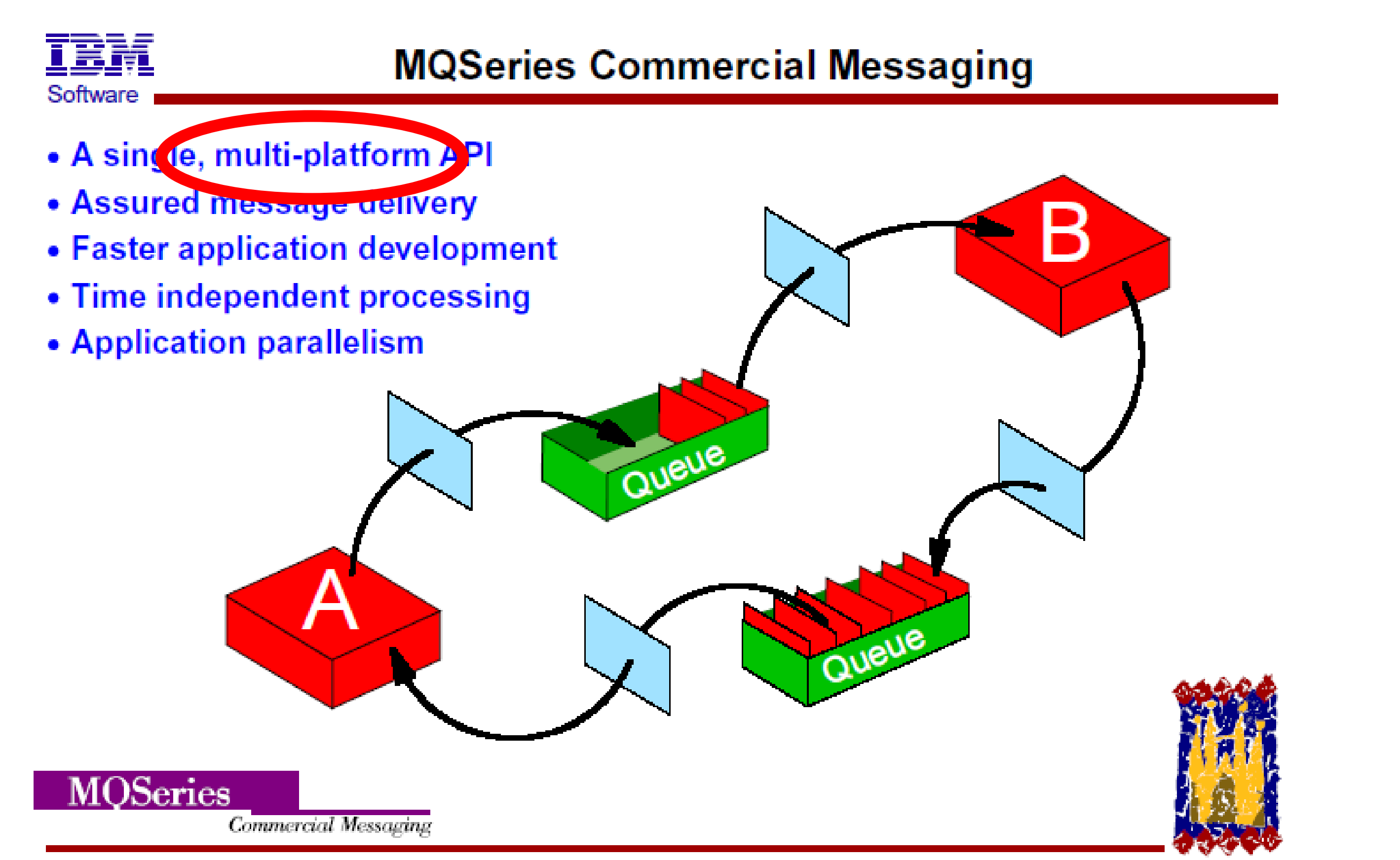
# What's new with IBM MQ: Messaging for the Modern Era

## August 2020 – includes MQ 9.2

Mark Taylor
MQ Development
IBM Hursley

# What is MQ – the 1995 version



IBM Software

## MQSeries Commercial Messaging

- A single, multi-platform API
- Assured message delivery
- Faster application development
- Time independent processing
- Application parallelism

MQSeries

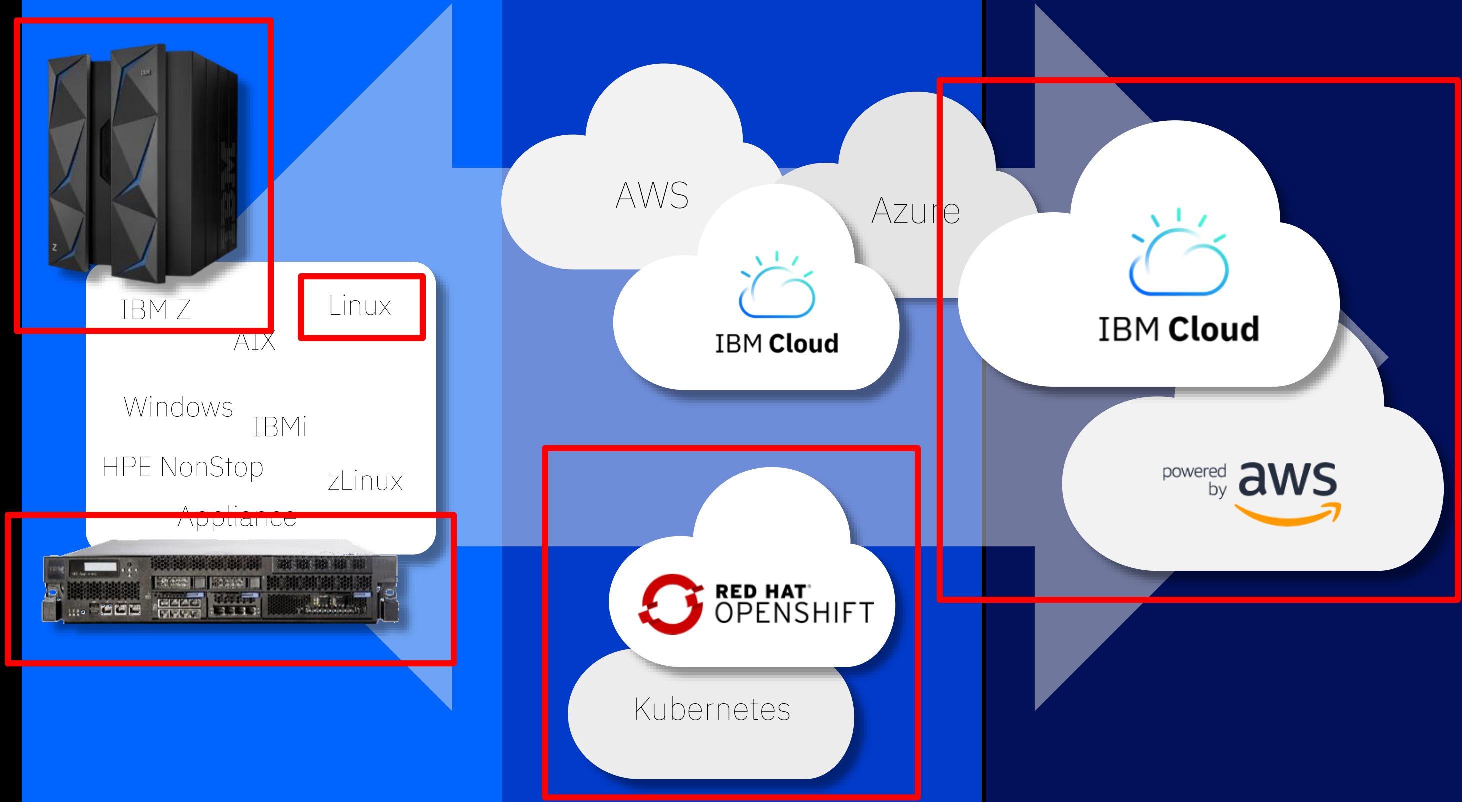
*Commercial Messaging*

# Run IBM MQ in any location or cloud, exactly as you need it

## On-premise, software and the MQ Appliance

## Run MQ yourself in public or private clouds

## Let IBM host MQ for you with its managed SaaS MQ service in public clouds, IBM Cloud and AWS



# IBM MQ Transformation

## Developer Agility

Delivery teams are being empowered within the organization. They need to be enabled to complete their day to day operations independently.

## Adopt Multi-Cloud

Delivery teams are empowered to select their cloud of choice, and expect connectivity to be provided across these.

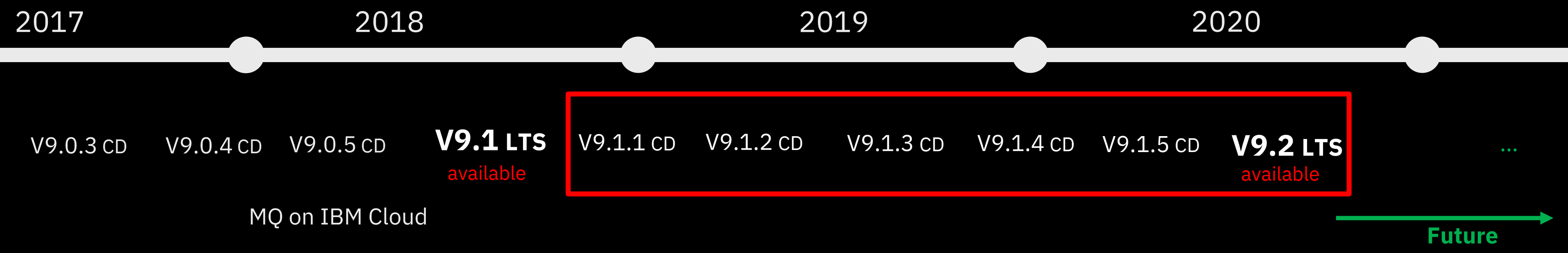
## Operational Agility

IBM MQ operational teams are being challenged to simplify the management of their infrastructure to drive cost savings.

## Organic Growth

Organizations that originally chosen IBM MQ for a project that has matured, and need improved scalability, availability and security.

# IBM MQ: long term support and continuous delivery



In 2016 MQ introduced a dual Long Term Support and a Continuous Delivery model.

## Continuous Delivery

New CD versions of MQ are released approximately every four months, incrementally introducing new product capabilities.

Intended for those that can continually integrate.

## Long Term Support

Approximately every two years a new LTS version is released, rolling up many of the CD capabilities into a release with 5+3 support attached.

Required by those looking for fixed function.

## Mix and Match

Both are available under the same license.

Both can interoperate, just like any previous version of MQ.

All the function delivered in the 9.1.x CD releases will be available in the long term support release **V9.2 LTS**

# MQ 9.0.x CD content, included with V9.1 LTS

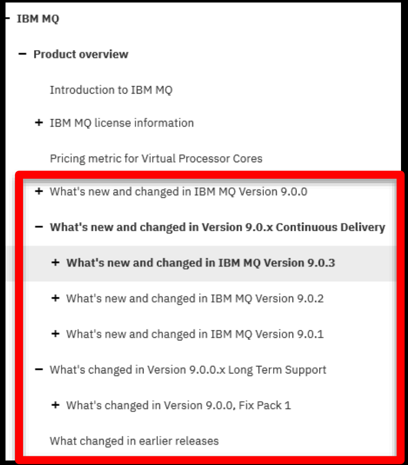
- Replicated Data Queue Manager for MQ Advanced
- Linear logging automation and performance
- RESTful administration
- Error log formatting
- Web Console
- RESTful messaging
- MQ Appliance performance improvements
- MQ JMS in CICS Liberty Profile
- Salesforce bridge
- AMS confidentiality performance on z/OS Advanced
- Blockchain bridge for MQ Advanced
- Floating IP support for MQ Appliance
- Code repository integration
- Backup and Restore on MQ Appliance
- Redistributable MFT agent for MQ Advanced
- Enhanced MFT diagnostics
- Cross LPAR MFT agents for z/OS Advanced
- SNMP and REST support for MQ Appliance

# MQ 9.2 LTS content…

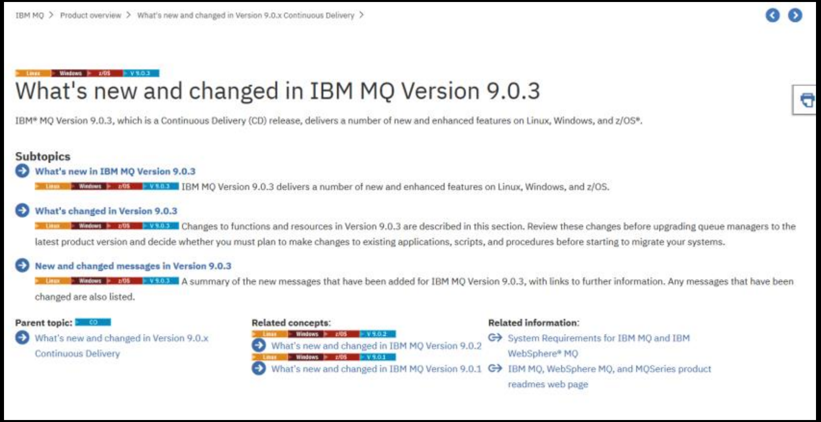
- Uniform Cluster automatic application rebalancing
- Microsoft .NET Core support
- Client connectivity with zCEE
- Developer toolkit for MacOS
- Automatic TLS CipherSpec negotiation
- Enhanced Salesforce Bridge
- Build toolkit for zCEE
- Idempotent MQSC commands
- Browse messages using REST
- MQ Appliance certificate expiry notifications
- Channel enabled AMS policies for z/OS
- JSON format CCDT
- Permitted TLS CipherSpec control
- REST messaging performance enhancements
- Full JSON-syntax REST administration
- MQ Appliance HA event notifications
- Improved distributed queue manager restart times
- Stream MQ Appliance error logs
- Rapid Uniform Cluster rebalancing
- Improved MQIPT management
- New application status checking
- ini file and MQSC injection at startup
- Escalating end queue manager
- MQFT REST list resource monitors
- Enhanced Blockchain Bridge
- WebSphere Liberty MDB pause
- New consistent MQ samples
- MFT REST create file transfer
- FTP server support on IBM I for MFT
- AMS HSM with Oracle JRE
- MQ Appliance admin activity audit logging
- XA support in Liberty for decoupled JMS connections
- Automatic Uniform Cluster configuration
- Packaged MQ Internet Passthru (IPT)
- Highly available MFT Agent deployments
- z/OS data set encryption support
- User controlled application naming
- TLS 1.3 support …
- High speed transfer over long distances with Fasp.io
- Qpid JMS shared subscriptions
- Publish messages over REST
- .NET project templates
- Increased queue size support for Distributed
- New improved Web Console
- Full HA-DR-HA replicated data queue manager deployments
- Uniform Cluster application monitoring
- Java 11 application support
- Distributed queue size control

# MQ release-to-release changes

Always read the **What’s new and changed** sections of the Knowledge Centre to see what each release adds



- IBM MQ
  - − Product overview
    - Introduction to IBM MQ
    - + IBM MQ license information
    - Pricing metric for Virtual Processor Cores
    - + What's new and changed in IBM MQ Version 9.0.0
    - − **What's new and changed in Version 9.0.x Continuous Delivery**
      - + **What's new and changed in IBM MQ Version 9.0.3**
      - + What's new and changed in IBM MQ Version 9.0.2
      - + What's new and changed in IBM MQ Version 9.0.1
    - − What's changed in Version 9.0.0.x Long Term Support
      - + What's changed in Version 9.0.0, Fix Pack 1
    - What changed in earlier releases



IBM MQ > Product overview > What's new and changed in Version 9.0.x Continuous Delivery >

Linux Windows z/OS V 9.0.3

## What's new and changed in IBM MQ Version 9.0.3

IBM® MQ Version 9.0.3, which is a Continuous Delivery (CD) release, delivers a number of new and enhanced features on Linux, Windows, and z/OS®.

**Subtopics**

**What's new in IBM MQ Version 9.0.3**
Linux Windows z/OS V 9.0.3 IBM MQ Version 9.0.3 delivers a number of new and enhanced features on Linux, Windows, and z/OS.

**What's changed in Version 9.0.3**
Linux Windows z/OS V 9.0.3 Changes to functions and resources in Version 9.0.3 are described in this section. Review these changes before upgrading queue managers to the latest product version and decide whether you must plan to make changes to existing applications, scripts, and procedures before starting to migrate your systems.

**New and changed messages in Version 9.0.3**
Linux Windows z/OS V 9.0.3 A summary of the new messages that have been added for IBM MQ Version 9.0.3, with links to further information. Any messages that have been changed are also listed.

**Parent topic:** CD
What's new and changed in Version 9.0.x Continuous Delivery

**Related concepts:**
Linux Windows z/OS V 9.0.2
What's new and changed in IBM MQ Version 9.0.2
Linux Windows z/OS V 9.0.1
What's new and changed in IBM MQ Version 9.0.1

**Related information:**
System Requirements for IBM MQ and IBM WebSphere® MQ
IBM MQ, WebSphere MQ, and MQSeries product readmes web page

# MQ in Containers, continually evolving

MQ first supported Docker containers in 2015, showing how a stateful solution can run in an often stateless world.

docker

MQ was one of the first certified containers available on IBM’s Kubernetes platform, IBM Cloud Private. Showing how to run MQ in a managed container environment.

MQ added support for running on Red Hat OpenShift

Red Hat OpenShift

MQ is a core component of IBM’s Cloud Pak for Integration, providing enterprise messaging for the Integration Platform solution

2015 **hub.docker.com/r/ibmcom/mq** **github.com/ibm-messaging/mq-container** 2020

# MQ Operator

Operators codify operational knowledge and workflows to automate life-cycle management of containerized applications with Kubernetes

```
apiVersion: mq.ibm.com/v1beta1
kind: QueueManager
metadata:
    name: quickstart-cp4i
spec:
    version: 9.1.5.0-r2
    license:
        accept: false
        license: L-RJON-BN7PN3
        use: NonProduction
web:
    enabled: true
queueManager:
    name: "QUICKSTART"
    storage:
        queueManager:
            type: ephemeral
....
```

# MQ on Cloud service

IBM Cloud

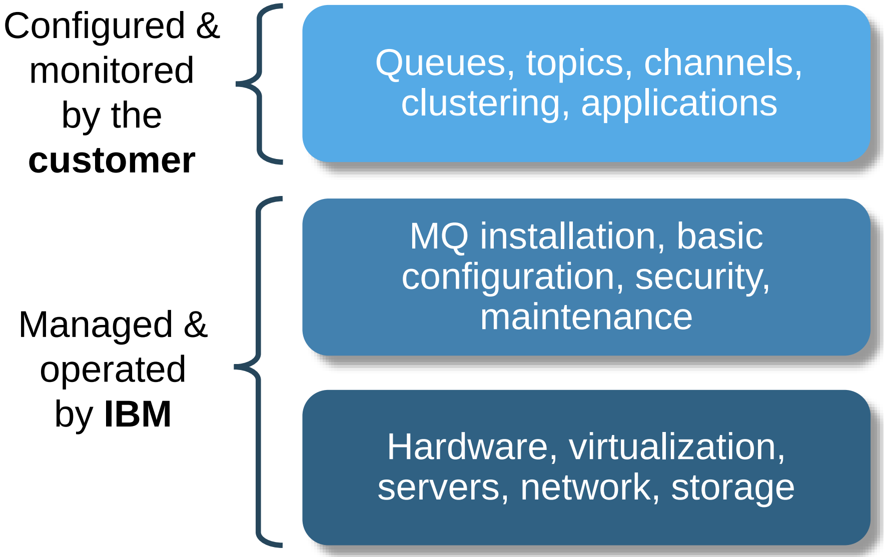
powered by aws

**Managed for You**

**Up and Running in Minutes**

**Hourly billing**

**Enabled for Hybrid Cloud Connectivity**



Configured & monitored by the **customer**

Queues, topics, channels, clustering, applications

Managed & operated by **IBM**

MQ installation, basic configuration, security, maintenance

Hardware, virtualization, servers, network, storage

Try the service for <u>free</u> **www.ibm.com/cloud/mq**

# The evolution on the MQ on Cloud service

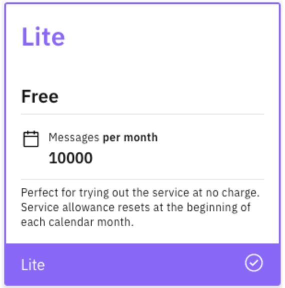
Continually broadened coverage to multiple geographical regions and across multiple public clouds – IBM Cloud and AWS

Added the ability to use indefinitely with no financial commitment through the Lite Plan

Simplified administration with SSO and improvements for diagnostics

Continually enhancing operational efficiencies which goes on to guide MQ’s evolution

IBM Cloud
powered by aws


Lite

Free

Messages **per month**
10000

Perfect for trying out the service at no charge. Service allowance resets at the beginning of each calendar month.

Lite

# MQ Appliance

The scalability and security of IBM MQ

The same familiar administration model for administrators with MQ skills

Supports the same MQ applications

But, with the convenience, fast time-to-value and low total cost of ownership of an appliance

## Easy Integration

Integrates seamlessly into MQ networks and and clusters

## Improved Availability

Built-in support for High Availability and Disaster Recovery

## Simplified ownership

Repeatable and fast, with less configuration or tuning required

Minimises dependencies on other resources and teams

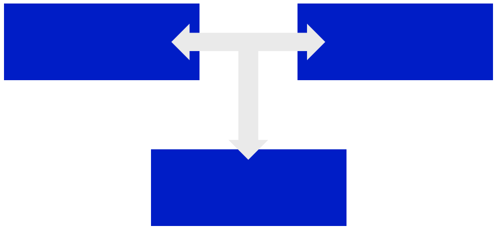
Simpler licensing and easier to assess for security compliance and audit

# Since 9.1

The MQ Appliance has been maturing, taking in real life experiences to drive improvements for all users

Improved diagnostics and setup
- Streaming error logs
- Network config tooling
- Recoverable error reports

Improved HA and DR
- Added resiliency to failures
- Status notifications for monitoring



# MQ for z/OS

## Maximum resilience, performance, and secure connectivity

**zHyperWrite**
Improves the I/O performance of synchronous replication solutions for disaster recovery

**Direct connectivity with IBM Event Streams**
Kafka connectors for MQ can be deployed into z/OS UNIX System Services, reducing latency and simplifying configuration

**Advanced Message Security**
Users are able to apply and remove Advanced Message Security (AMS) policies transparently between AMS and non-AMS enabled queue managers

**Full data encryption**
MQ 9.1.5 completed support for full DataSet encryption, integrating with the CryptoExpress coprocessor for encryption at the storage level

### Resilience

Queue sharing groups exploit the z/OS Parallel Sysplex for unparalleled high availability

### Performance

Create high performance environments able to process millions of messages every second

### Secure connectivity

Adapters and bridges provide tight integration with your business critical Systems of Record

Consistent connectivity with a range of other on-premise and cloud platforms

MQ exploits System SSL on z/OS to utilize CPACF and CryptoExpress cards for pervasive encryption

# Linux High Availability

Protecting your critical data

# Fault Tolerance

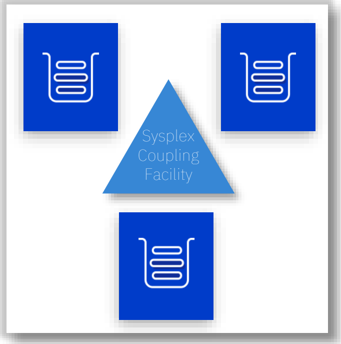
MQ delivers HA through the ability to build horizontally scaled, active-active systems and typically **active-passive HA** of the data itself*, the messages.

Traditionally active-passive HA has been achieved through **HA clusters** or **multi instance** queue managers. Both rely on highly available infrastructure to be setup and relied on.

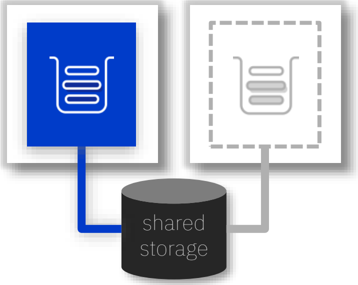
The **MQ Appliance** changed this with a fully integrated HA solution, providing built in machine to machine data replication and failover.

And even more recently, **Replicated Data Queue Managers** on RHEL x86 have provided even more options.

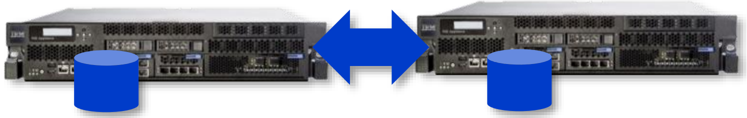
* z/OS shared queue provides active-active HA of the message data!



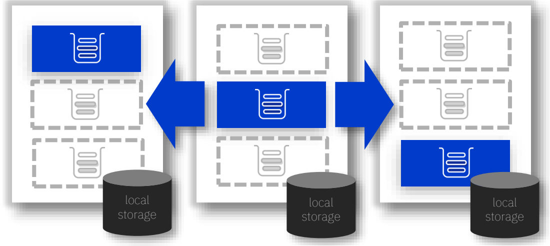
z/OS Queue Sharing Groups



Multi-instance queue managers and HA Cluster



MQ Appliance



Replicated Data Queue Manager

# Combining HA with DR

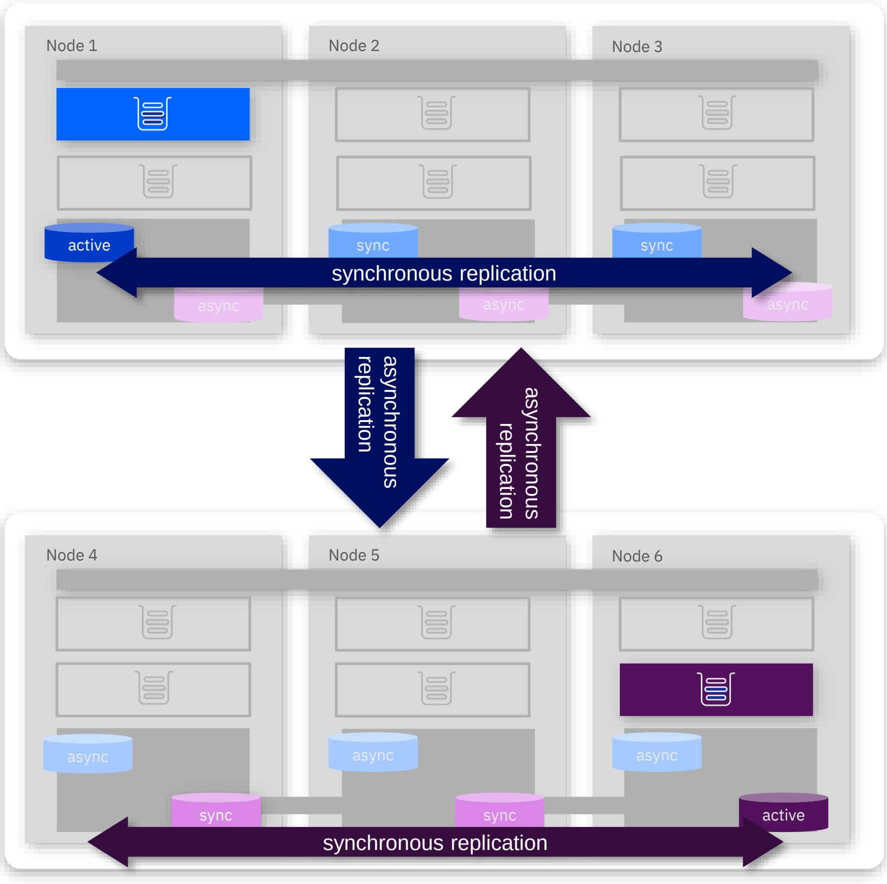
IBM MQ 9.1.5 CD combines two existing RDQM topologies into one

Previously it was a choice between either automatic HA failover with a three node HA group or a manual two node failover configuration supporting asynchronous replication for higher latency deployments (e.g. DR). Not both together.

You can now build a three node HA quorum system, asynchronously replicating queue manager state to a matching three node HA quorum system for simpler DR switch over setups

Both HA quorum systems can be running different active queue managers, with bidirectional asynchronous replication, supporting active/active DR topologies

IBM MQ Advanced 9.1.5 CD
RHEL x86

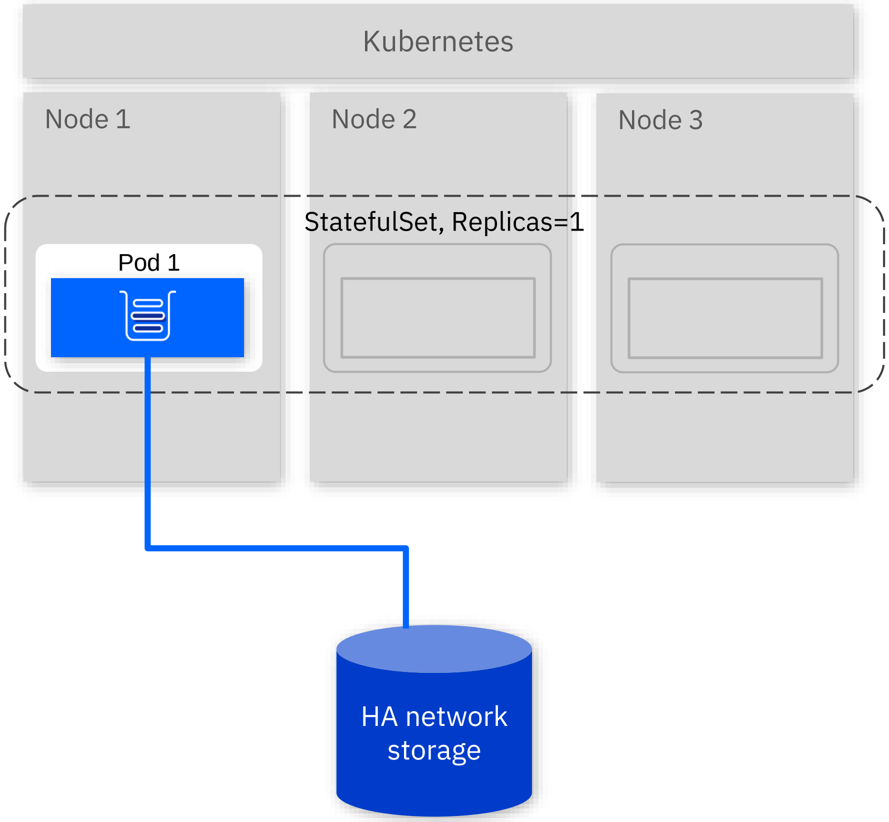


# High Availability with Kubernetes

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…



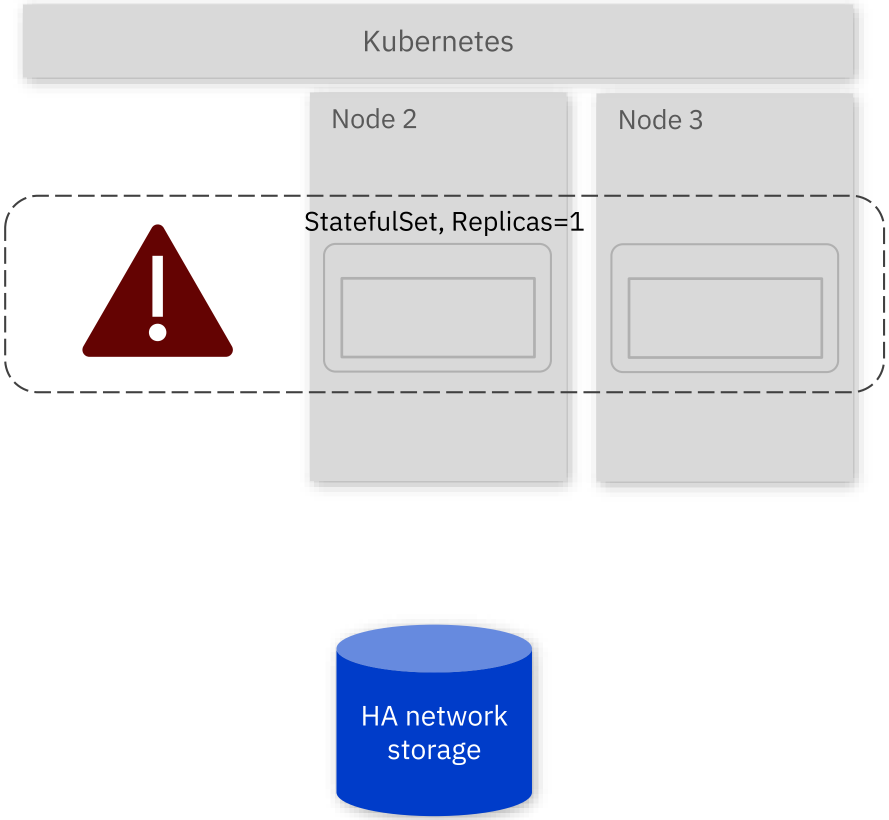
# High Availability with Kubernetes

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. StatefullSets such as MQ are not automatically restarted following a Kubernetes node failure



# High Availability with Kubernetes

IBM MQ 9.1.3 CD

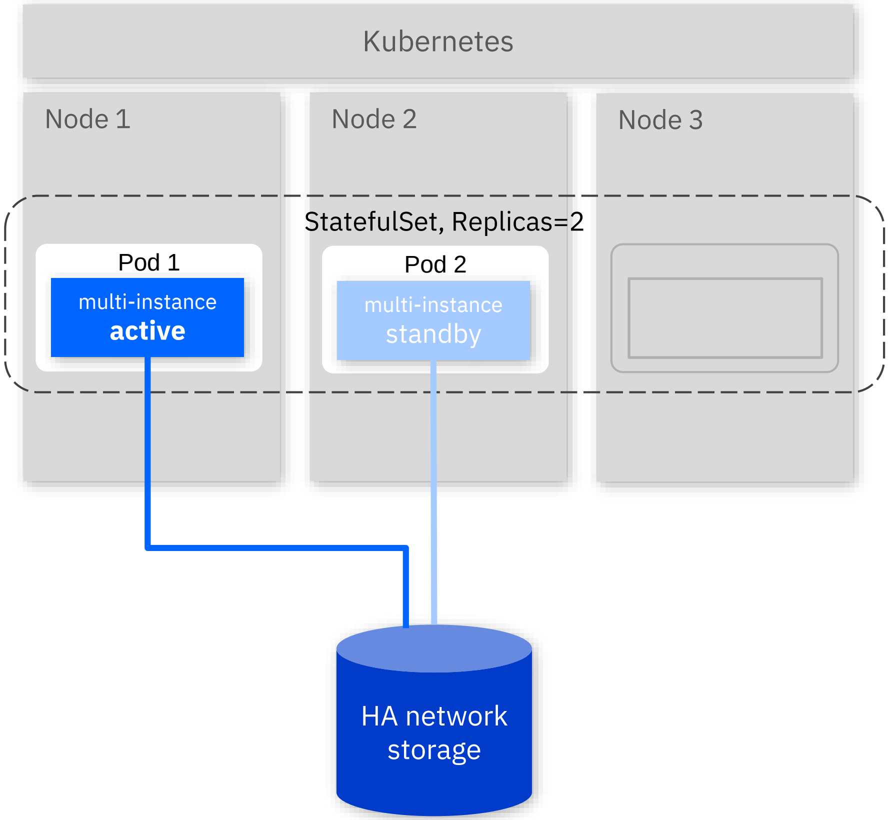
The RDQM solution does not apply to container environments

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. StatefullSets such as MQ are not automatically restarted following a Kubernetes node failure

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



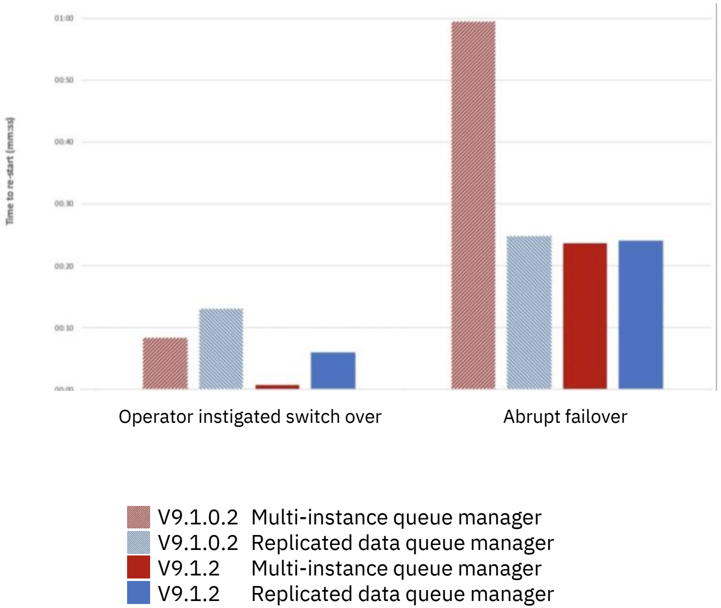
https://www.ibm.com/support/knowledgecenter/SSFKSJ_9.1.0/com.ibm.mq.ctr.doc/ha_for_ctr.htm

# Increasing your availability further

IBM MQ 9.1.1+ CD

With automatic queue manager failover, queue manager restart times have an increasing part to play in achieving the highest levels of availability

MQ 9.1.x CD has focused on driving down the time it takes to stop and start distributed queue managers under load



ibm-messaging.github.io/mqperf/Queue%20Manager%20Restart%20Times.pdf

500 connected applications, driving 50k-85k msgs/sec

# Active/active messaging

Building scalable, fault tolerant, solutions

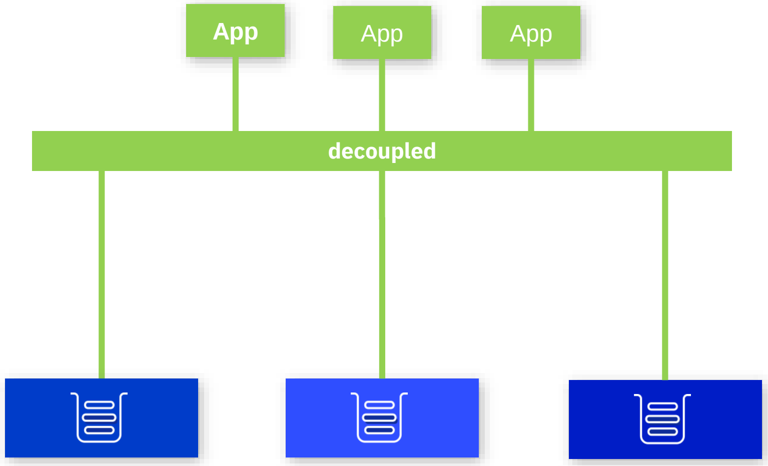
# Building scalable, fault tolerant, solutions

To provide an active/active, solution you need to consider multiple active queue managers acting as a *single service*

Applications also run as multiple instances for availability and scale

Applications treat the queue managers as interchangeable and want to connect to the group in the most efficient and available distribution

MQ introduced the **Uniform Cluster** capability in 9.1.2 CD to enable such deployments much more easily



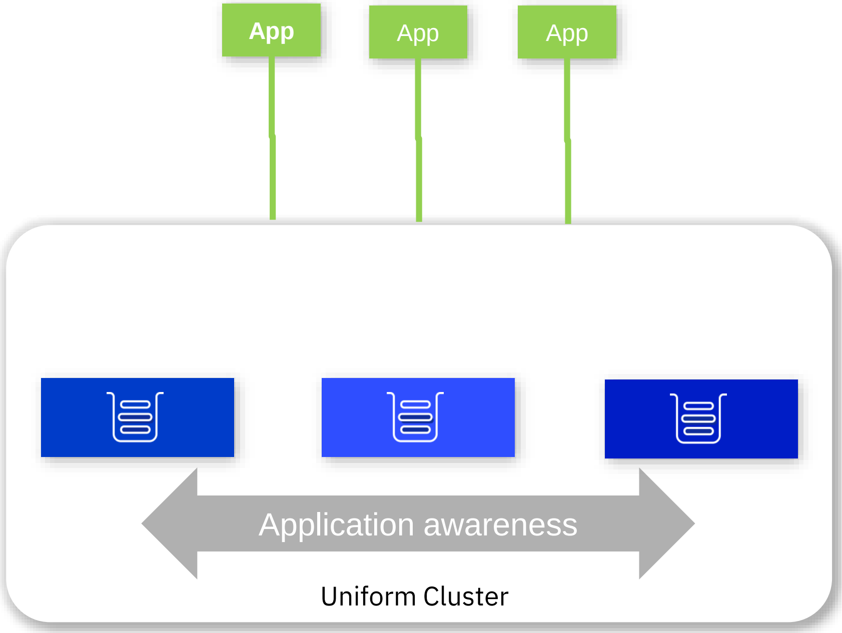
# Building scalable, fault tolerant, solutions

Uniform Clusters are a special type of MQ Cluster. One where all the queue managers provide the same service, such as queues.

Application instances are dynamically distributed across the available queue managers, adjusting as queue managers and application instances stop and start.

A uniform cluster builds on top of existing MQ building blocks -

Client auto-reconnect
CCDT queue manager groups
MQ Clustering



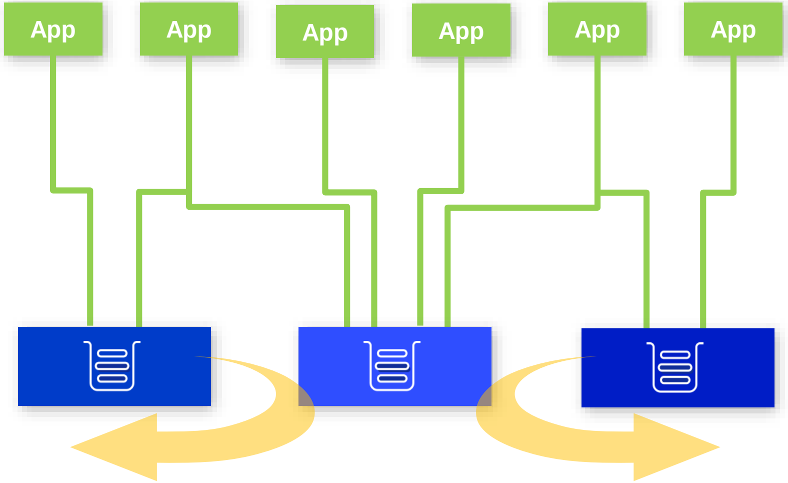
# Automatic Application balancing

Uniform Clusters are a special type of MQ Cluster. One where all the queue managers provide the same service, such as queues.

Application instances are dynamically distributed across the available queue managers, adjusting as queue managers and application instances stop and start.

A uniform cluster builds on top of existing MQ building blocks -

Client auto-reconnect
CCDT queue manager groups
MQ Clustering



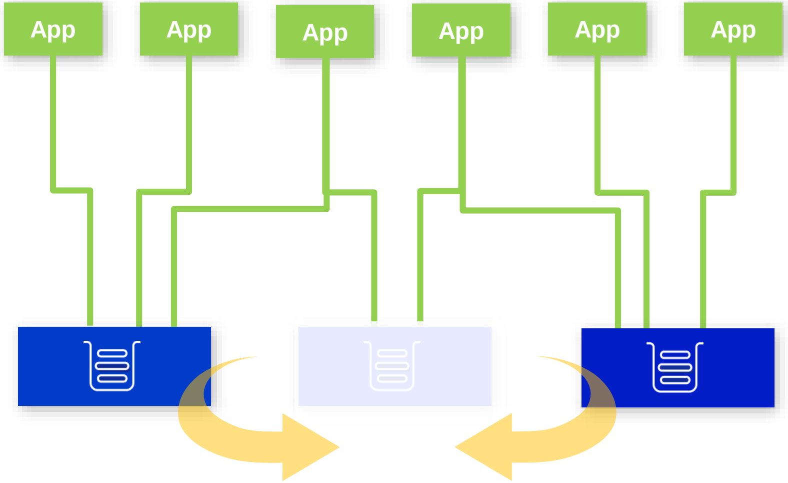
# Automatic Application balancing

Uniform Clusters are a special type of MQ Cluster. One where all the queue managers provide the same service, such as queues.

Application instances are dynamically distributed across the available queue managers, adjusting as queue managers and application instances stop and start.

A uniform cluster builds on top of existing MQ building blocks -

Client auto-reconnect
CCDT queue manager groups
MQ Clustering



# View application status

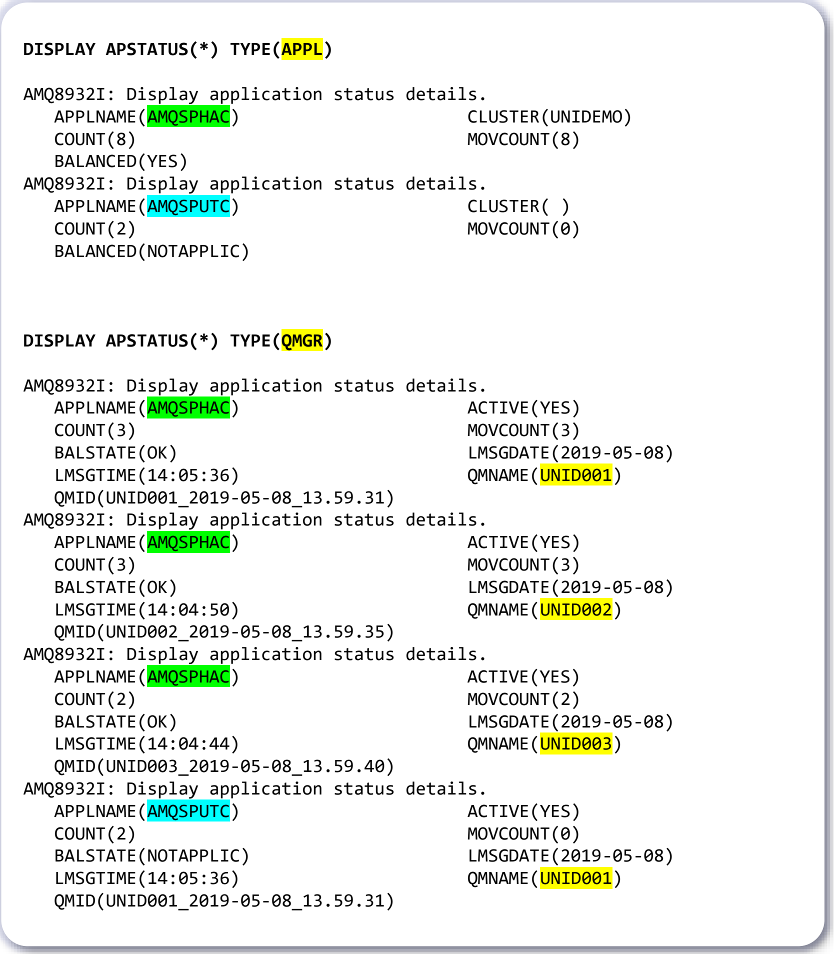
Now that MQ is taking a more application centric view, a new command has been added to Distributed runmqsc to aid the understanding of how applications are balanced across a Uniform Cluster

From any member of the Uniform Cluster, displays applications by name and highlights instances that are not evenly balanced

community.ibm.com/community/user/imwuc/viewdocument/display-application-status-on-a-uni

MQ 9.1.5 CD adds to this by regularly publishing metrics to the system topics on how each application is being rebalanced, enabling live monitoring

community.ibm.com/community/user/imwuc/blogs/louis-horsley1/2020/04/06/uniform-cluster-monitor-application-resource-usage



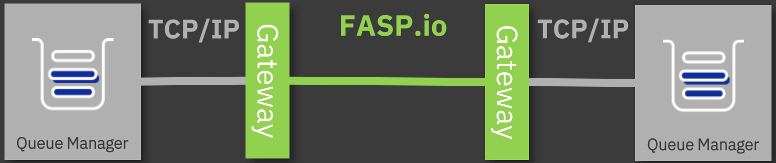
```
DISPLAY APSTATUS(*) TYPE(APPL)

AMQ8932I: Display application status details.
   APPLNAME(AMQSPHAC)                      CLUSTER(UNIDEMO)
   COUNT(8)                                MOVCOUNT(8)
   BALANCED(YES)
AMQ8932I: Display application status details.
   APPLNAME(AMQSPUTC)                      CLUSTER( )
   COUNT(2)                                MOVCOUNT(0)
   BALANCED(NOTAPPLIC)


DISPLAY APSTATUS(*) TYPE(QMGR)

AMQ8932I: Display application status details.
   APPLNAME(AMQSPHAC)                      ACTIVE(YES)
   COUNT(3)                                MOVCOUNT(3)
   BALSTATE(OK)                            LMSGDATE(2019-05-08)
   LMSGTIME(14:05:36)                      QMNAME(UNID001)
   QMID(UNID001_2019-05-08_13.59.31)
AMQ8932I: Display application status details.
   APPLNAME(AMQSPHAC)                      ACTIVE(YES)
   COUNT(3)                                MOVCOUNT(3)
   BALSTATE(OK)                            LMSGDATE(2019-05-08)
   LMSGTIME(14:04:50)                      QMNAME(UNID002)
   QMID(UNID002_2019-05-08_13.59.35)
AMQ8932I: Display application status details.
   APPLNAME(AMQSPHAC)                      ACTIVE(YES)
   COUNT(2)                                MOVCOUNT(2)
   BALSTATE(OK)                            LMSGDATE(2019-05-08)
   LMSGTIME(14:04:44)                      QMNAME(UNID003)
   QMID(UNID003_2019-05-08_13.59.40)
AMQ8932I: Display application status details.
   APPLNAME(AMQSPUTC)                      ACTIVE(YES)
   COUNT(2)                                MOVCOUNT(0)
   BALSTATE(NOTAPPLIC)                     LMSGDATE(2019-05-08)
   LMSGTIME(14:05:36)                      QMNAME(UNID001)
   QMID(UNID001_2019-05-08_13.59.31)
```

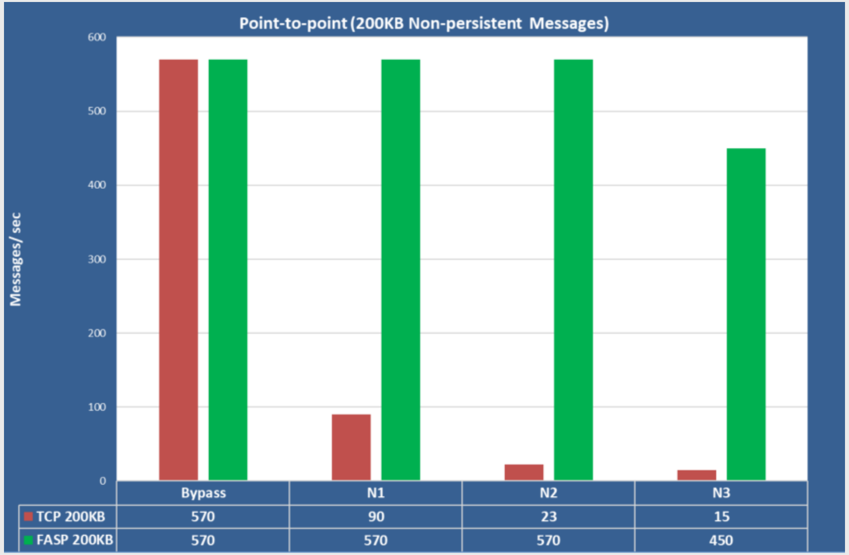
# Extend your MQ network

# A Global Messaging Network
## *IBM MQ Advanced and Aspera*

At the heart of Aspera is the FASP protocol, accelerating the speed of data transport across long distances and poor networks

MQ Advanced brings you the benefits of this when communicating between distant queue managers





| | Bypass | N1 | N2 | N3 |
|---|---|---|---|---|
| 🟥 TCP 200KB | 570 | 90 | 23 | 15 |
| 🟩 FASP 200KB | 570 | 570 | 570 | 450 |

**Bypass**: 0ms network latency (no packet loss)
**N1**: 25ms network latency (no packet loss)
**N2**: 40ms network latency (0.1% packet loss)
**N3**: 50ms network latency (0.5% packet loss)

ibm-messaging.github.io/mqperf/MQ914_fasp_gw.pdf

# A Global Messaging Network
*Internet Pass Through*

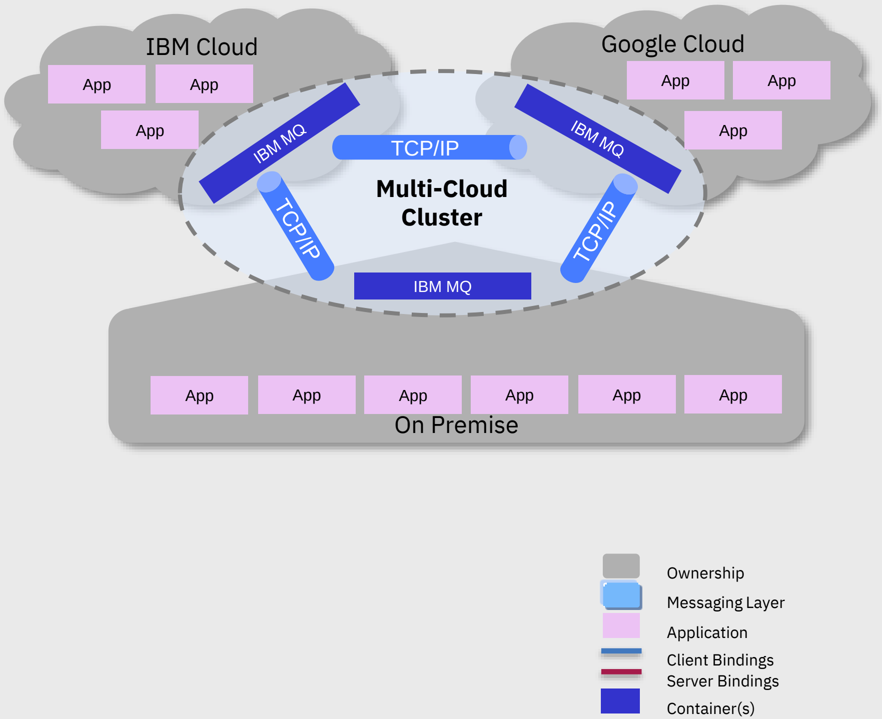
With a Hybrid Multi-Cloud Architecture connecting to external MQ networks is becoming increasingly important.

Internet Pass Through (IPT) has been an IBM MQ support Pac (MS81) for many years. It provides a proxy layer within your architecture which can be useful when exposing MQ outside of the organization data center.

MQ 9.1.4 aligned IPT with the MQ product delivery

MQ Advanced now provides an enhanced IPT entitlement where a Hardware Security Module (HSM) can be used with IPT.

Guidance on how to expose MQ is also provided here: ibm.biz/MQSecureConn

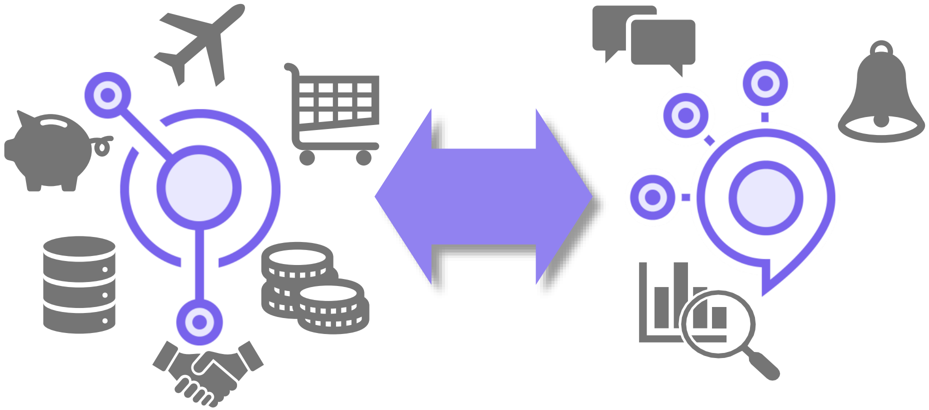


# IBM MQ with IBM Event Streams

**IBM MQ** connects mission-critical systems, requiring **transactional**, **once-only delivery**

**Event Streams** distributes and processes streams of events in real-time to intelligently engage with customers

Connecting the two together, flowing messages and events between then, with the **supported connectors** enables you to unlock the potential of your data



# Managing MQ

Living with your enterprise messaging system

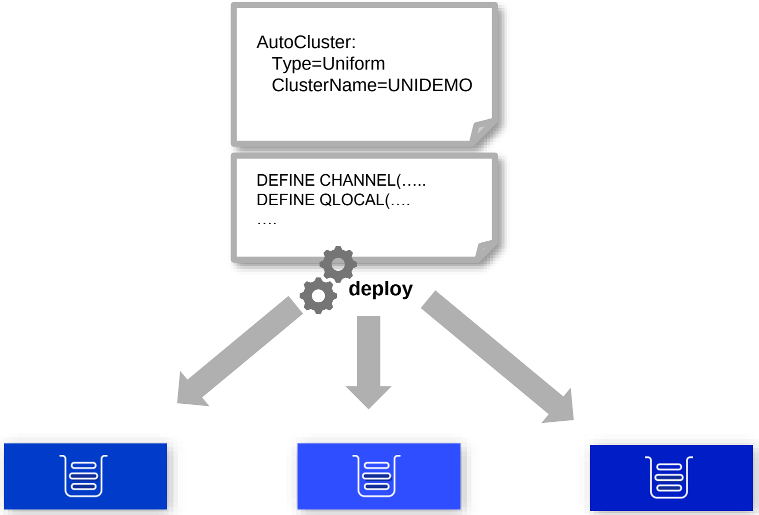
# Automation with MQ

IBM MQ 9.1.4 CD

Scripting is key to automation. MQ has supported scripting through MQSC scripts.

MQ has been evolving to make this even easier

- **Remote runmqsc** enables scripts to be deployed from a system remote to the queue manager
- MQSC commands are now more idempotent
- Queue managers can now automatically pull in updated **MQSC scripts** and **ini file** settings at start time **(MQ 9.1.4)**
- New **REST API** support opens up administration over HTTP using JSON **(MQ 9.1.3)**



```
AutoCluster:
  Type=Uniform
  ClusterName=UNIDEMO
```

```
DEFINE CHANNEL(.....
DEFINE QLOCAL(....
....
```

# REST administration

IBM MQ 9.1.3 CD
All Platforms

JSON format, MQSC style, REST commands

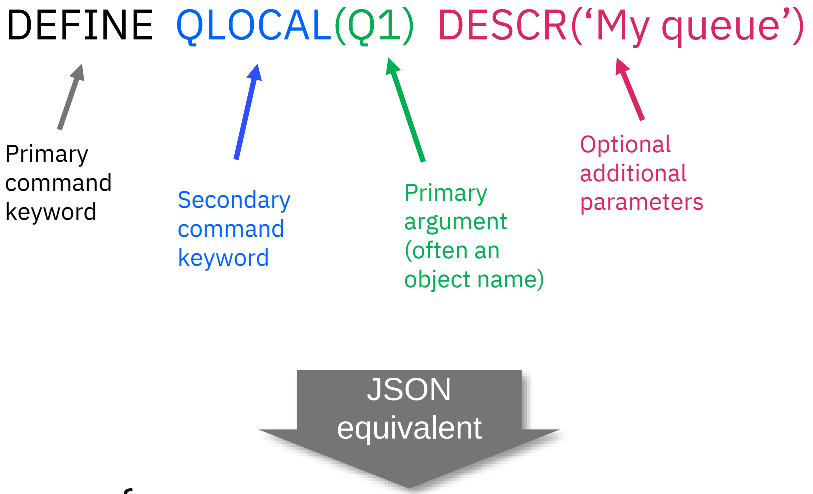
Send request body in **HTTP POST** to **admin/action/qmgr/{qmgrName}/mqsc** resource

New command type of “**runCommandJson**”

Existing command type of “**runCommand**” can still be used to run a plain text MQSC command

MQ 9.1.5 CD carries these APIs over into a new V2 of the REST API. Earlier APIs for per-object manipulation have been stabilised at V1

www.ibm.com/support/knowledgecenter/en/SSFKSJ_9.1.0/com.ibm.mq.pro.doc/q133690_.htm#q133690___restapiv2



DEFINE QLOCAL(Q1) DESCR(‘My queue’)

Primary command keyword
Secondary command keyword
Primary argument (often an object name)
Optional additional parameters
JSON equivalent

```
{
  "type": "runCommandJSON",
  "command": "define",
  "qualifier": "qlocal",
  "name": "q1",
  "parameters": {
    "descr": "My queue"
  }
}
```

# Code as Config for the applications too

IBM MQ 9.1.2+ CD
Clients

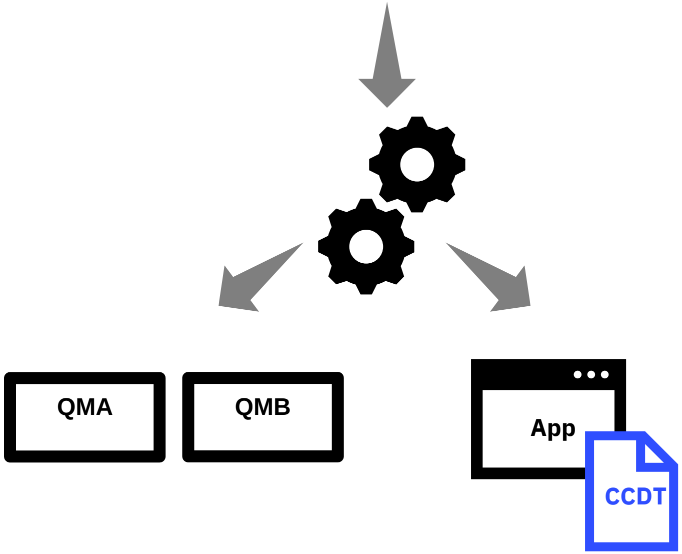
Applications should never encode the MQ connection details, not even the queue manager

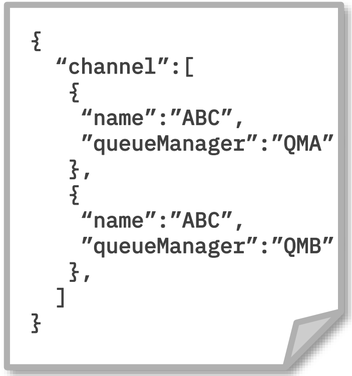
MQ CCDTs encapsulate the connection details

You can now build your **own JSON format CCDTs**

These can be deployed as part of the same pipeline that deploys your queue managers and applications

Supports multiple channels of the same name on different queue managers to simplify the building of uniform clusters





```
{
  “channel”:[
   {
    “name”:”ABC”,
    ”queueManager”:”QMA”
   },
   {
    “name”:”ABC”,
    ”queueManager”:”QMB”
   },
  ]
}
```

# Queue size control

IBM MQ 9.1.5 CD
Distributed

Distributed platforms and the Appliance have introduced per-queue disk space control with MQ 9.1.5 CD.

This enables much greater control over resource usage by individual applications.

Queue size control has also introduced the ability for queues to be much larger than the previously fixed 2 terabytes. This improves MQ's ability to temporarily buffer significant messaging traffic during an extended outage.

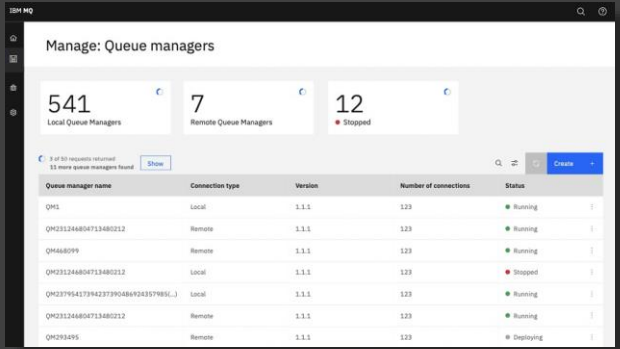
The maximum supported size is now 255TB

community.ibm.com/community/user/imwuc/blogs/louis-horsley1
/2020/04/08/easily-controllable-queue-file-sizes

# New Web Console

MQ 9.1.5 CD replaces the existing web console with a new web console on the Distributed platforms

Focus is on user experience and consistency across IBM products



IBM MQ

## Manage: Queue managers

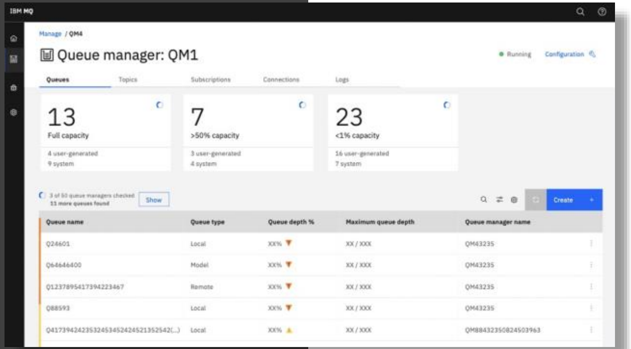
541 Local Queue Managers

7 Remote Queue Managers

12 ● Stopped

3 of 50 requests returned
11 more queue managers found

| Queue manager name | Connection type | Version | Number of connections | Status |
|---|---|---|---|---|
| QM1 | Local | 1.1.1 | 123 | ● Running |
| QM231246804713480212 | Remote | 1.1.1 | 123 | ● Running |
| QM468099 | Remote | 1.1.1 | 123 | ● Running |
| QM231246804713480212 | Local | 1.1.1 | 123 | ● Stopped |
| QM23795417394237390486924357985(...) | Local | 1.1.1 | 123 | ● Running |
| QM231246804713480212 | Remote | 1.1.1 | 123 | ● Running |
| QM293495 | Remote | 1.1.1 | 123 | ● Deploying |



IBM MQ

Manage / QM4

## Queue manager: QM1

● Running

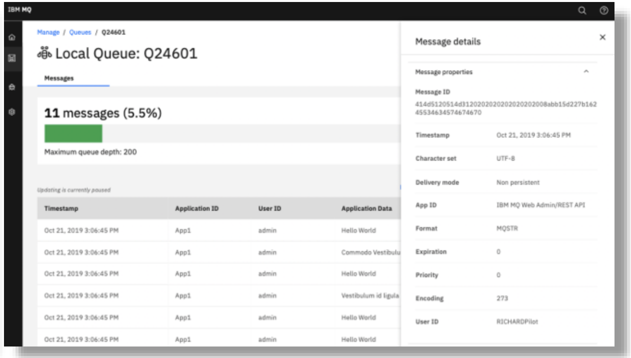
13 Full capacity
4 user-generated
9 system

7 >50% capacity
3 user-generated
4 system

23 <1% capacity
16 user-generated
7 system

3 of 50 queue managers checked
11 more queues found

| Queue name | Queue type | Queue depth % | Maximum queue depth | Queue manager name |
|---|---|---|---|---|
| Q24601 | Local | XX% ▼ | XX / XXX | QM43235 |
| Q64646400 | Model | XX% ▼ | XX / XXX | QM43235 |
| Q1237895417394223467 | Remote | XX% ▼ | XX / XXX | QM43235 |
| Q88593 | Local | XX% ▼ | XX / XXX | QM43235 |
| Q41739424235324534524245213525 42(...) | Local | XX% ▲ | XX / XXX | QM88432350824503963 |



IBM MQ

Manage / Queues / Q24601

## Local Queue: Q24601

11 messages (5.5%)

Maximum queue depth: 200

Updating is currently paused

| Timestamp | Application ID | User ID | Application Data |
|---|---|---|---|
| Oct 21, 2019 3:06:45 PM | App1 | admin | Hello World |
| Oct 21, 2019 3:06:45 PM | App1 | admin | Commodo Vestibulu |
| Oct 21, 2019 3:06:45 PM | App1 | admin | Hello World |
| Oct 21, 2019 3:06:45 PM | App1 | admin | Vestibulum id ligula |
| Oct 21, 2019 3:06:45 PM | App1 | admin | Hello World |
| Oct 21, 2019 3:06:45 PM | App1 | admin | Hello World |

Message details

Message properties

Message ID
414d5120514d312020202020202020202008abb15d227b162455346345746746 70

| | |
|---|---|
| Timestamp | Oct 21, 2019 3:06:45 PM |
| Character set | UTF-8 |
| Delivery mode | Non persistent |
| App ID | IBM MQ Web Admin/REST API |
| Format | MQSTR |
| Expiration | 0 |
| Priority | 0 |
| Encoding | 273 |
| User ID | RICHARDPilot |

https://community.ibm.com/community/user/imwuc/blogs/callum-jackson1/2020/04/09/enhanced-web-console-in-ibm-mq-915

# Managing channel CipherSpecs

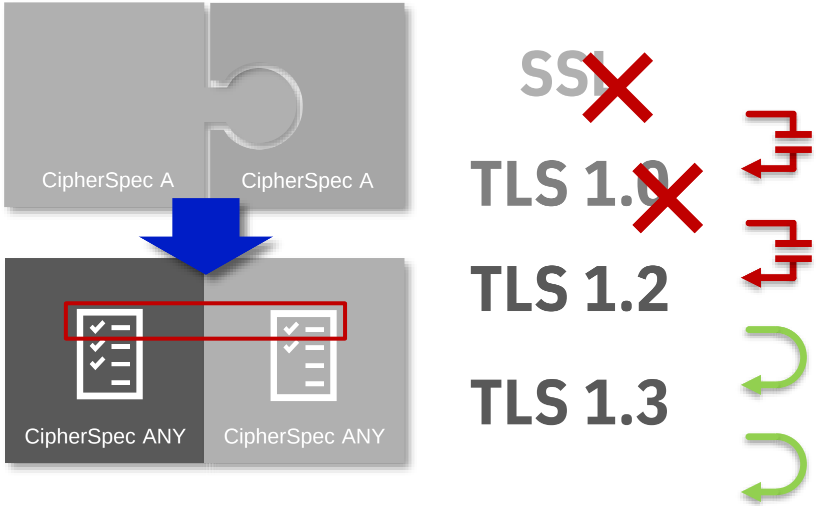
IBM MQ 9.1.1, 9.1.4, 9.1.5 CD

Making it easier to keep up-to-date with ever changing ciphers, simplifying migration

MQ 9.1.4 CD adds TLS 1.3 support for Distributed queue  manager channels and C-based clients. 9.1.5 has added support for Java 11 applications which is needed for TLS 1.3 support there.

Rather than needing to match the CipherSpec on both ends of a channel, MQ has introduced ***ANY_TLSxx*** and ***ANY_TLSxx_OR_HIGHER*** CipherSpecs and MQ will negotiate the CipherSpec available to both ends

For 9.1.1, the distributed platforms also added the ability to configure *exactly* which CipherSpecs a queue manager will accept



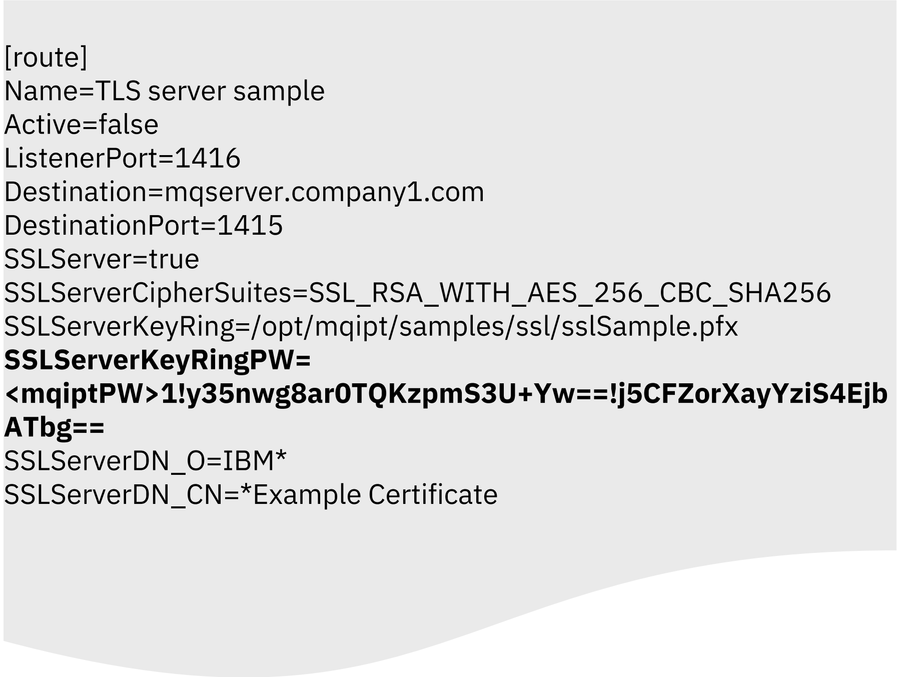
MQ 9.2 has a reordering of the CipherSpecs to match common views on levels of security and the ability for you to reorder that list

# Passwords and Credentials

Where passwords or other credentials need to be stored in a configuration file, many can now be protected with a user-supplied key in a consistent fashion

Components include
- MQ IPT
- AMS Java applications
- Salesforce Bridge
- Blockchain Bridge
- MFT

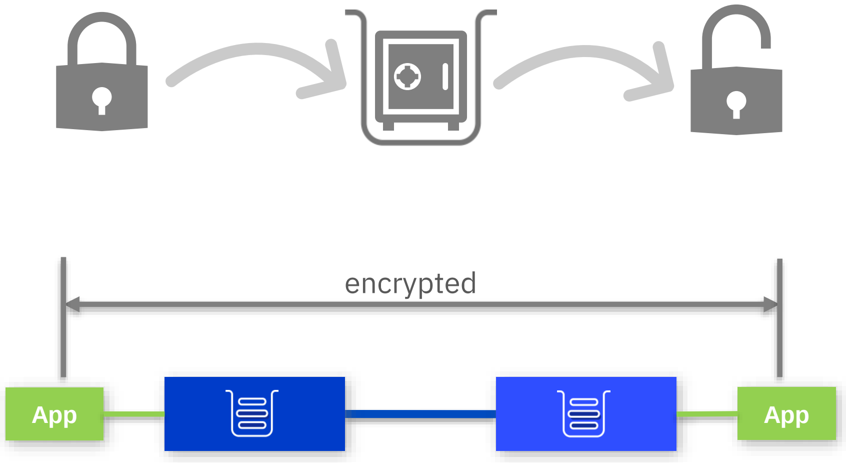
The new feature consists of 2 parts:
- A stronger algorithm
- Ability to use a custom encryption key



```
[route]
Name=TLS server sample
Active=false
ListenerPort=1416
Destination=mqserver.company1.com
DestinationPort=1415
SSLServer=true
SSLServerCipherSuites=SSL_RSA_WITH_AES_256_CBC_SHA256
SSLServerKeyRing=/opt/mqipt/samples/ssl/sslSample.pfx
SSLServerKeyRingPW=
<mqiptPW>1!y35nwg8ar0TQKzpmS3U+Yw==!j5CFZorXayYziS4Ejb
ATbg==
SSLServerDN_O=IBM*
SSLServerDN_CN=*Example Certificate
```

# Advanced Message Security

End-to-end application-to-application encryption may give you the highest level of security, but it’s not always possible to use. For example, where the applications are not AMS enabled or where the originators or recipients of the messages are outside of your domain

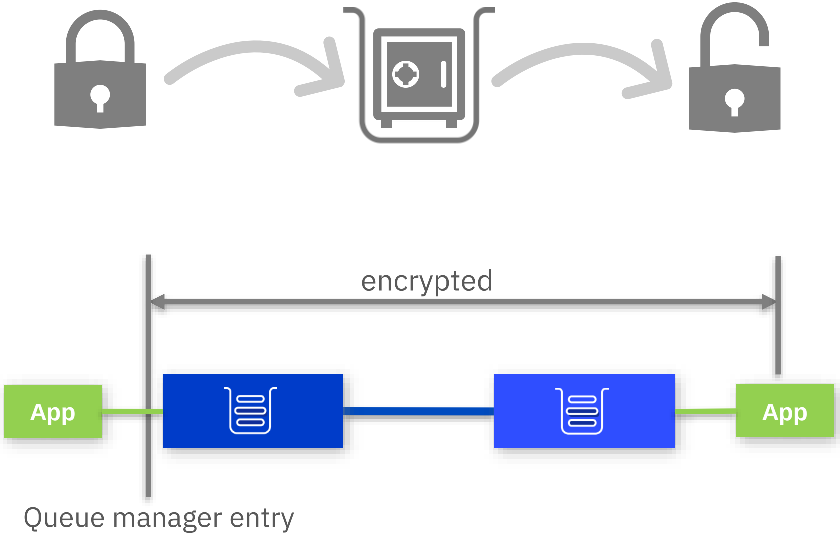


Application to application

# Advanced Message Security

End-to-end application-to-application encryption may give you the highest level of security, but it’s not always possible to use. For example, where the applications are not AMS enabled or where the originators or recipients of the messages are outside of your domain

MQ on Distributed long-ago implemented client level interception to apply AMS policies once messages reach or leave their first queue manager



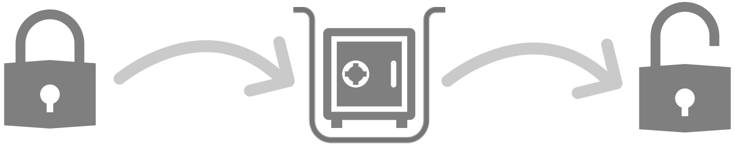
# Advanced Message Security

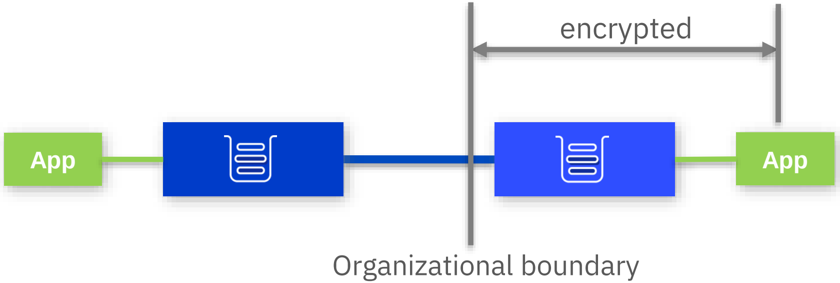
IBM MQ Advanced 9.1.3 CD z/OS

End-to-end application-to-application encryption may give you the highest level of security, but it's not always possible to use. For example, where the applications are not AMS enabled or where the originators or recipients of the messages are outside of your domain

MQ on Distributed long-ago implemented client level interception to apply AMS policies once messages reach or leave their first queue manager

MQ 9.1.3 on z/OS adds the ability to apply those policies at a queue manager-to-queue manager boundary. This enables the use of AMS within one domain without affecting another





# Dataset encryption with MQ on z/OS

z/OS added support for policy based dataset encryption in z/OS 2.2 and later, utilising a CryptoExpress coprocessor

With MQ 9.1.5 CD, Dataset encryption can be used with all of MQ’s datasets

This provides encryption at rest for MQ data, although MQ’s Advanced Message Security capability goes further by providing true end-to-end encryption

## Pervasive encryption with IBM z Systems

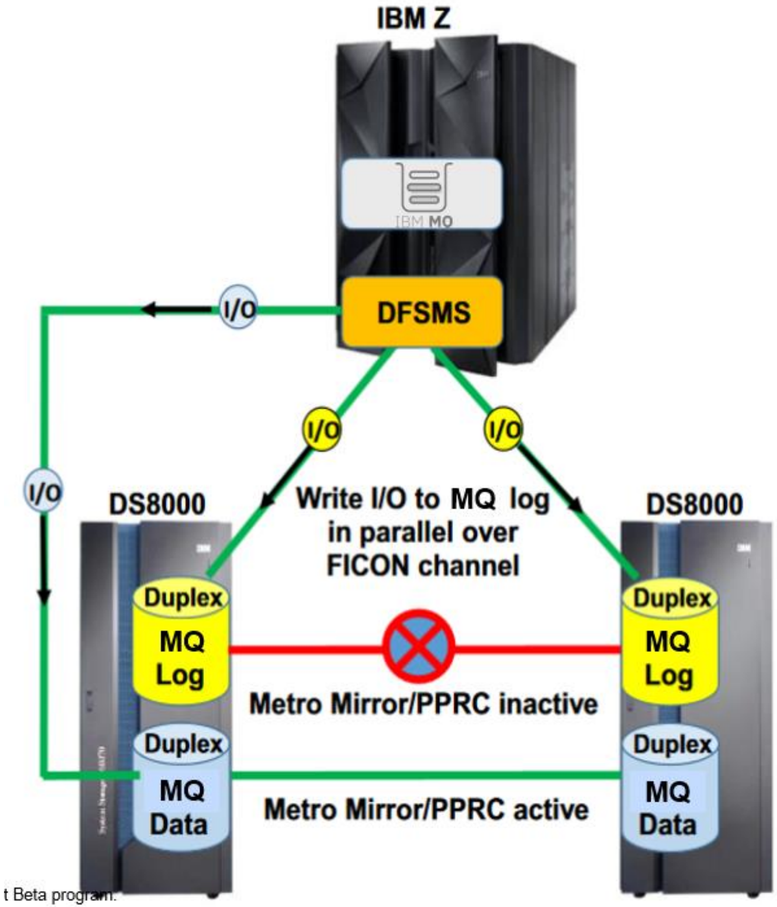
- Integrated Crypto Hardware
- **Data at Rest**
- Network
- Clustering
- Data in Use

https://community.ibm.com/community/user/imwuc/viewdocument/mq-and-the-use-of-data-set-encrypti?CommunityKey=b382f2ab-42f1-4932-aa8b-8786ca722d55

# z/OS zHyperwrite

Reduces the cost of using PPRC (Metro-Mirror) to synchronously replicate log data by issuing the write to the primary and secondary copies of the data at the DFSMS (Media Manager) level.
This allows the writes to occur in parallel instead of in series.

- Reduced I/O times by **up to 60%**.
- Reduced elapsed time for commit by **up to 60%**, which can reduce contention.
- Improved the sustained log rate, allowing each queue manager to process **up to 2.4 times** the volume of workload.



t Beta program.

# Managed File Transfer

IBM MQ Advanced 9.1 LTS +
All platforms

MFT manages your file transfers, with file-to-file and file-to-message.

MQ 9.1.x continued to focus on resiliency and ease of administration.

Active/standby MFT agent support adds highly available topologies

Expanded the breadth of REST APIs, both for monitoring and configuring MFT resources and for initiating file transfers.

# Building applications

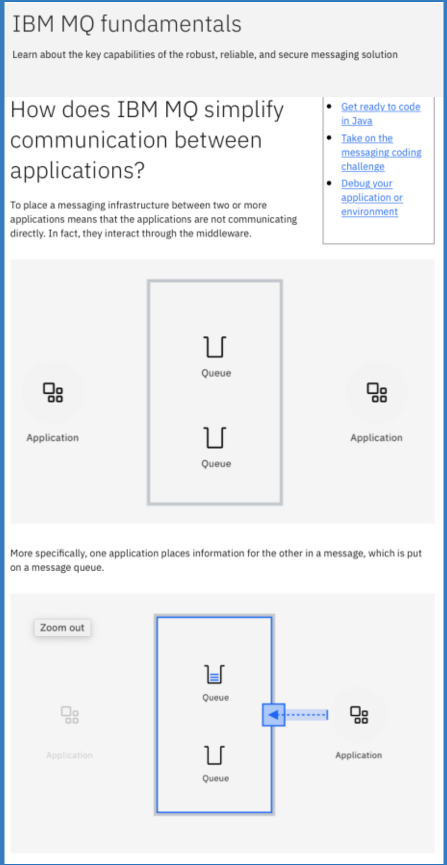
# learn-mq

Finding it hard to get developers started with MQ?

Point them to:

ibm.biz/learn-mq

## Totally new to MQ? Learn the basics



IBM MQ fundamentals

Learn about the key capabilities of the robust, reliable, and secure messaging solution

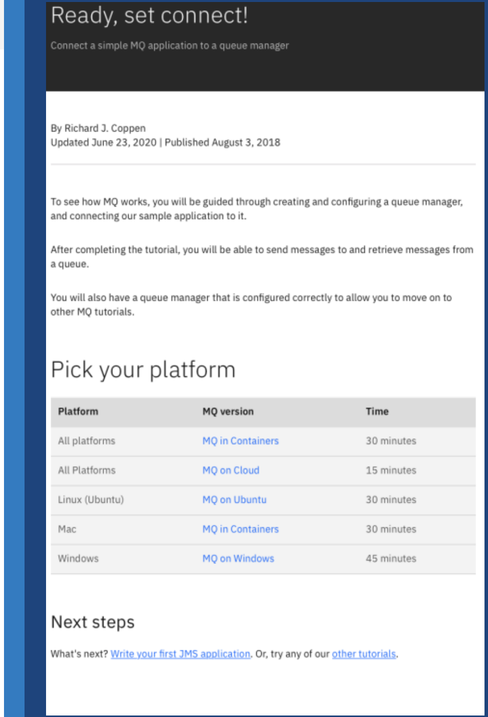
How does IBM MQ simplify communication between applications?

To place a messaging infrastructure between two or more applications means that the applications are not communicating directly. In fact, they interact through the middleware.

- Get ready to code in Java
- Take on the messaging coding challenge
- Debug your application or environment

More specifically, one application places information for the other in a message, which is put on a message queue.

## Step-by-step guide to getting up and running with MQ



Ready, set connect!

Connect a simple MQ application to a queue manager

By Richard J. Coppen
Updated June 23, 2020 | Published August 3, 2018

To see how MQ works, you will be guided through creating and configuring a queue manager, and connecting our sample application to it.

After completing the tutorial, you will be able to send messages to and retrieve messages from a queue.

You will also have a queue manager that is configured correctly to allow you to move on to other MQ tutorials.

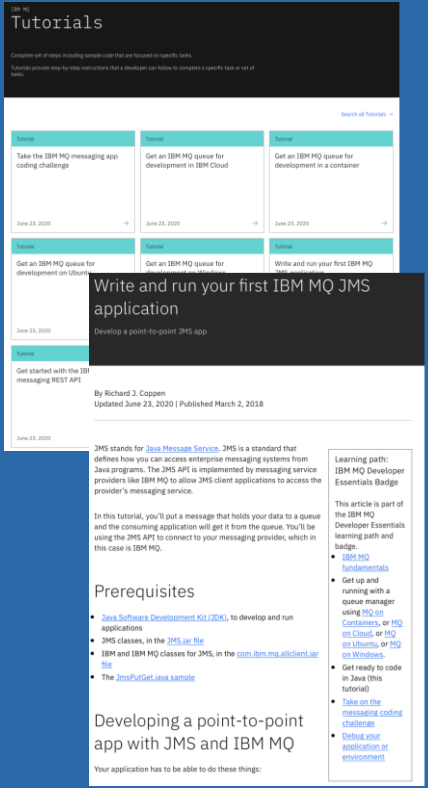
Pick your platform

| Platform | MQ version | Time |
|---|---|---|
| All platforms | MQ in Containers | 30 minutes |
| All Platforms | MQ on Cloud | 15 minutes |
| Linux (Ubuntu) | MQ on Ubuntu | 30 minutes |
| Mac | MQ in Containers | 30 minutes |
| Windows | MQ on Windows | 45 minutes |

Next steps

What's next? Write your first JMS application. Or, try any of our other tutorials.

## Tutorials on building your applications



IBM MQ

Tutorials

Complete set of steps including sample code that are focused on specific tasks.

Tutorials provide step-by-step instructions that a developer can follow to complete a specific task or set of tasks.

Search all Tutorials →

Tutorial
Take the IBM MQ messaging app coding challenge
June 23, 2020 →

Tutorial
Get an IBM MQ queue for development in IBM Cloud
June 23, 2020 →

Tutorial
Get an IBM MQ queue for development in a container
June 23, 2020 →

Tutorial
Get an IBM MQ queue for development on Ubuntu
June 23, 2020

Tutorial
Get an IBM MQ queue for

Tutorial
Write and run your first IBM MQ

Tutorial
Get started with the IBM
messaging REST API
June 23, 2020

Write and run your first IBM MQ JMS application

Develop a point-to-point JMS app

By Richard J. Coppen
Updated June 23, 2020 | Published March 2, 2018

JMS stands for Java Message Service. JMS is a standard that defines how you can access enterprise messaging systems from Java programs. The JMS API is implemented by messaging service providers like IBM MQ to allow JMS client applications to access the provider's messaging service.

In this tutorial, you'll put a message that holds your data to a queue and the consuming application will get it from the queue. You'll be using the JMS API to connect to your messaging provider, which in this case is IBM MQ.

Prerequisites

- Java Software Development Kit (JDK), to develop and run applications
- JMS classes, in the JMS.jar file
- IBM and IBM MQ classes for JMS, in the com.ibm.mq.allclient.jar file
- The JmsPutGet.java sample

Developing a point-to-point app with JMS and IBM MQ

Your application has to be able to do these things:

Learning path: IBM MQ Developer Essentials Badge

This article is part of the IBM MQ Developer Essentials learning path and badge.

- IBM MQ fundamentals
- Get up and running with a queue manager using MQ on Containers, or MQ on Cloud, or MQ on Ubuntu, or MQ on Windows.
- Get ready to code in Java (this tutorial)
- Take on the messaging coding challenge
- Debug your application or environment

# Demonstrating the simplicity of MQ

There's nothing like flashing lights and wires to grab people's attention. We want everyone to know how easy it is to write messaging applications and how powerful MQ is in supporting them

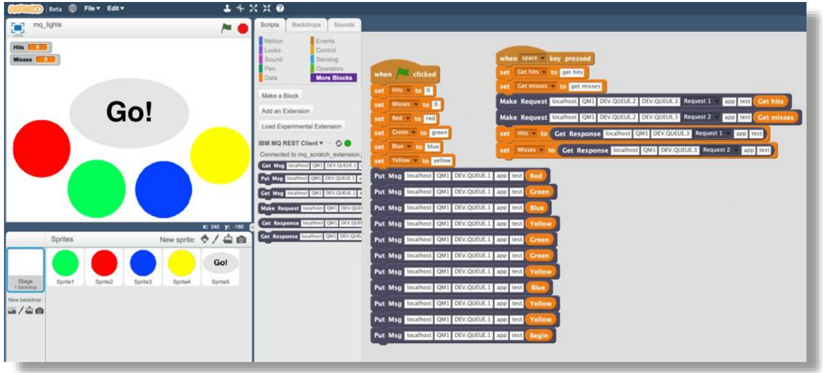
Ever tried **Scratch,** a graphical way to code, aimed at kids but ideal to show how easily asynchronous messaging can improve your applications with an MQ plugin

**ibm.biz/ibmmq-scratch**

Heard of the **Raspberry PI**? You think MQ is a heavyweight solution? We run an HA pair of queue managers on two $5 Raspberry PI Zeros!

**ibm.biz/ibmmq-pi**

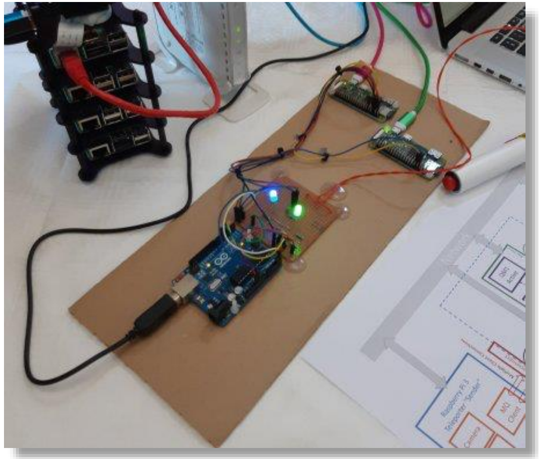
The PI version of MQ is now available under a developer (unsupported) license



```
Get Msg localhost QM1 DEV.QUEUE.1
Put Msg localhost QM1 DEV.QUEUE.1
Get Msg localhost QM1 DEV.QUEUE.1
Make Request localhost QM1 DEV.QUE
Get Response localhost QM1 DEV.QUE
Get Response localhost QM1 DEV.QUE
```

```
when ⚑ clicked
set Hits ▾ to 0
set Misses ▾ to 0
set Red ▾ to red
set Green ▾ to green
set Blue ▾ to blue
set Yellow ▾ to yellow
Put Msg localhost QM1 DEV.QUEUE.1 app test Red
Put Msg localhost QM1 DEV.QUEUE.1 app test Green
Put Msg localhost QM1 DEV.QUEUE.1 app test Blue
Put Msg localhost QM1 DEV.QUEUE.1 app test Yellow
Put Msg localhost QM1 DEV.QUEUE.1 app test Green
Put Msg localhost QM1 DEV.QUEUE.1 app test Green
Put Msg localhost QM1 DEV.QUEUE.1 app test Yellow
Put Msg localhost QM1 DEV.QUEUE.1 app test Blue
Put Msg localhost QM1 DEV.QUEUE.1 app test Yellow
Put Msg localhost QM1 DEV.QUEUE.1 app test Yellow
Put Msg localhost QM1 DEV.QUEUE.1 app test Begin
```

```
when space ▾ key pressed
set Get hits ▾ to get hits
set Get misses ▾ to get misses
Make Request localhost QM1 DEV.QUEUE.2 DEV.QUEUE.3 Request 1 ▾ app test Get hits
Make Request localhost QM1 DEV.QUEUE.2 DEV.QUEUE.3 Request 2 ▾ app test Get misses
set Hits ▾ to Get Response localhost QM1 DEV.QUEUE.3 Request 1 ▾ app test
set Misses ▾ to Get Response localhost QM1 DEV.QUEUE.3 Request 2 ▾ app test
```

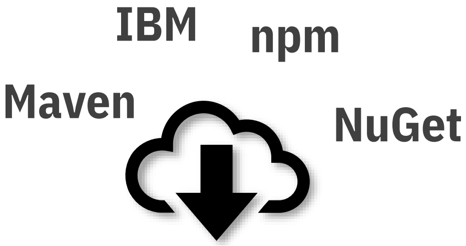


# Developing applications

Build your applications simply, with no need for an MQ installation

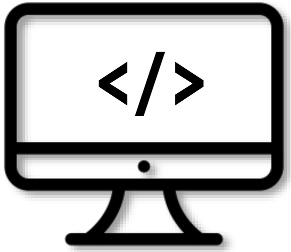
- Pull Java directly from the **Maven** repository and .NET from NuGet
- The **SDK** has been added to the MQ redistributable client, removing the need to install it to build from

**ibm.biz/MQdownloads**



Develop your applications on the platform of your choice for free

The full MQ Advanced for Developers is available on Windows and Linux with the addition of a MacOS MQ client and SDK for Developers



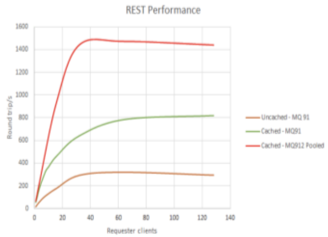
# Writing new applications

## REST Messaging

Providing a very simple way to get messages in and out of your MQ system

9.1.2 CD boosted the performance capability, 9.1.3 CD added message browse and 9.1.5 CD added publish

REST



put, get, browse, publish

## .NET Core

9.1.1 CD brought support for .NET Core on Windows

9.1.2 CD added Linux support

.NET Core

Windows

Linux

## Open Source language bindings

Write MQI applications in Node.js and Golang

New simpler JMS style API for Golang

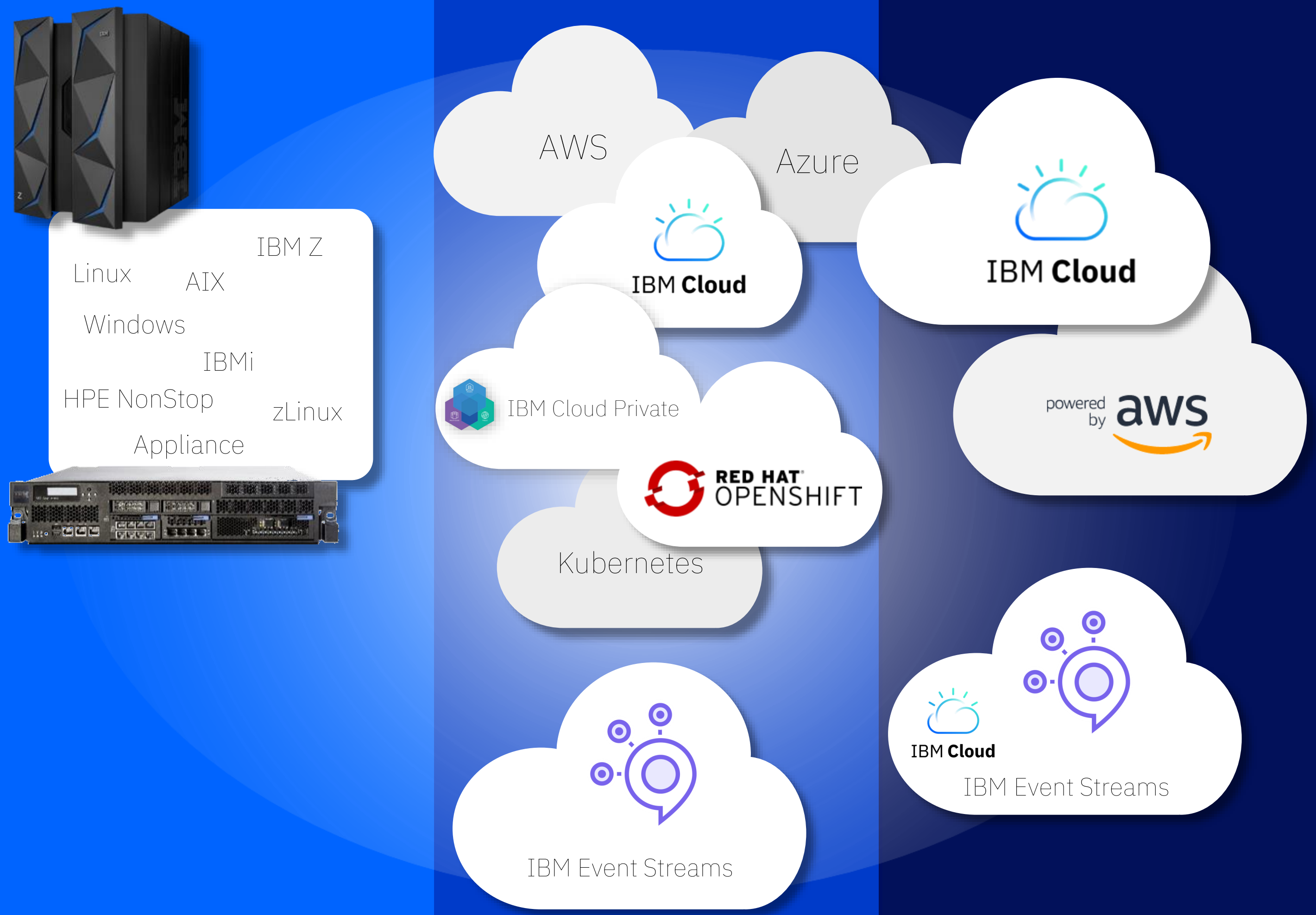
**github.com/ibm-messaging**

Node.js MQI

Golang MQI

Golang JMS

# Run IBM MQ in any location or cloud, exactly as you need it



IBM Z

Linux

AIX

Windows

IBMi

HPE NonStop

zLinux

Appliance

AWS

Azure

IBM Cloud

IBM Cloud

IBM Cloud Private

RED HAT OPENSHIFT

powered by aws

Kubernetes

IBM Event Streams

IBM Cloud

IBM Event Streams

# Thank you