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

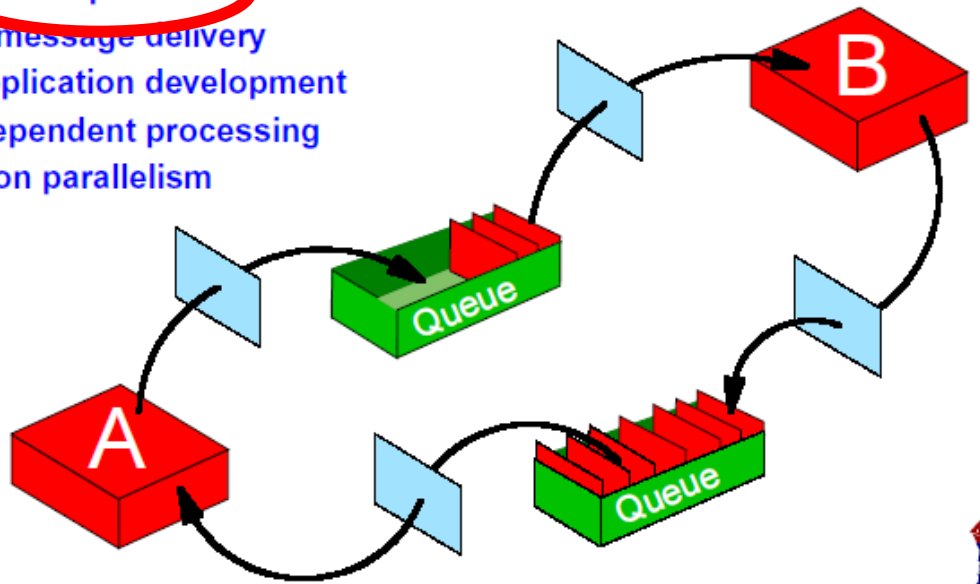
May 2020 – includes MQ V9.1.5

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

IBM MQ is *the* solution for business-critical messaging

The world depends on reliable, secure messaging and **85% of the fortune 100 depend on IBM MQ***

Your bank transfers complete without losing your money, with **all of the worlds top 50 banks using IBM MQ***

IBM Messaging

1 + 1 = 2

Simple

Scalable

Precise

Connected

Reliable

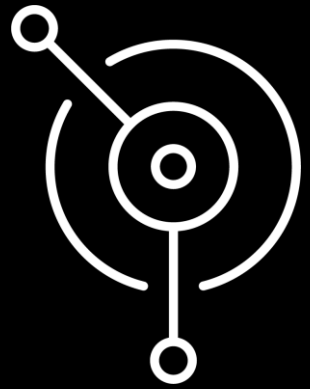
Secure

*Correct as of end 2016 against http://www.relbanks.com/worlds-top-banks/assets and http://beta.fortune.com/fortune500/list/

IBM **MQ**

Messaging, how you need it, where you need it

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



On-premise, software and the MQ Appliance

IBM Z
Linux
AIX
Windows
Solaris
IBMi
HPE NonStop
zLinux
Appliance

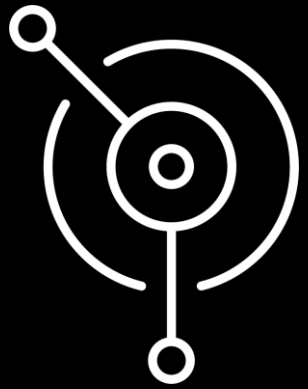
Run MQ yourself in public or private clouds

AWS
Azure
IBM Cloud
Kubernetes
RED HAT OPENSHIFT

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

IBM Cloud
powered by aws

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

MQ on Cloud service

Managed for You

Up and Running in Minutes

Hourly billing

Enabled for Hybrid Cloud Connectivity

IBM **Cloud**

powered by aws

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 free **www.ibm.com/cloud/mq**

MQ in Containers, continually evolving

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

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

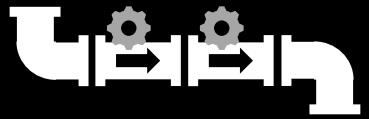
MQ is a core component of IBM's Cloud Pak for Integration, providing enterprise messaging for the Integration Platform solution

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

MQ in Containers, recent updates

Pipeline Support

Clients can develop DevOps pipelines that build customized images, either by layering configuration or rebuilding using the openly available container source and Helm chart



OpenShift Support

In V9.1.3 support for OpenShift was provided, with both Docker and Cri-o container technologies supported.

Increasing OpenShift security

In V9.1.2, IBM MQ runs as non-root. In V9.1.5, IBM MQ Advanced certified container can run under OpenShift's ***restricted*** Security Context Constraints.

Cloud Pak for Integration

The IBM MQ Advanced container is provided with integration for single sign on, common logging and dashboards and the ability to track transactions through multiple Integration capabilities

Introducing the MQ Appliance M2002

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

The M2002

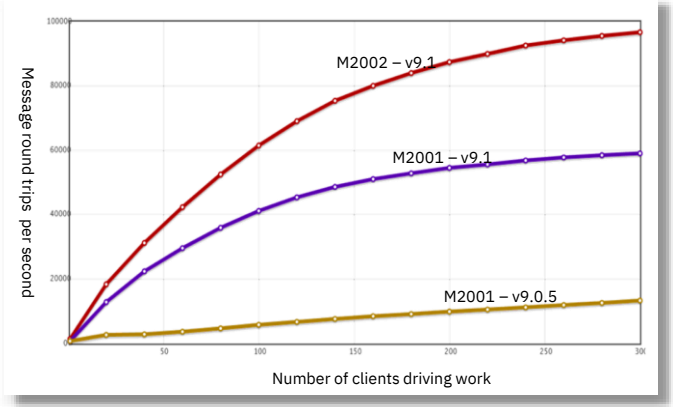
Choice of A/B models to suit different loads

Adds new 40GB network connectivity
- Particularly useful for HA replication

Doubled storage capacity with new RAID10 controller for improved performance

Based on MQ V9.1: Available to run both LTS and CD releases on the MQ Appliance

M2002 headline numbers
- Over **200 thousand persistent**, **HA replicated**, messages produced and consumed per second
- Over **600 thousand non-persistent** messages produced and consumed per second



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

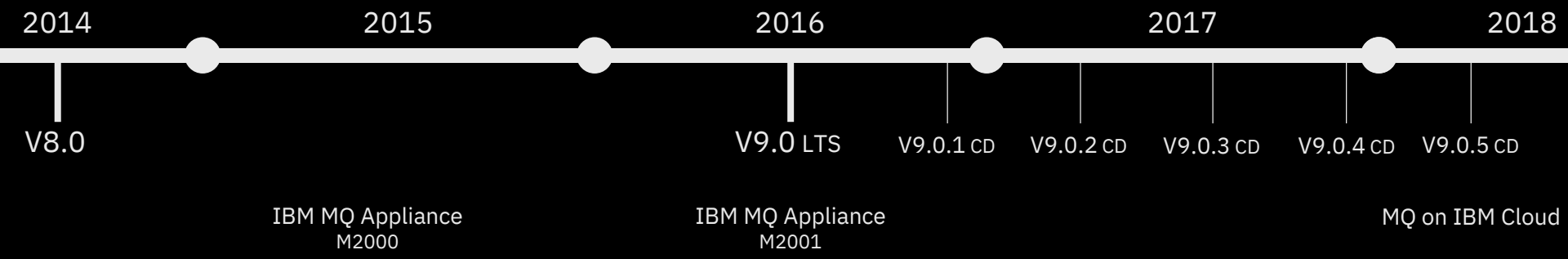
Consistent connectivity with a range of other on-premise and cloud platforms

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

IBM **MQ**

Continuous delivery and innovation

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.

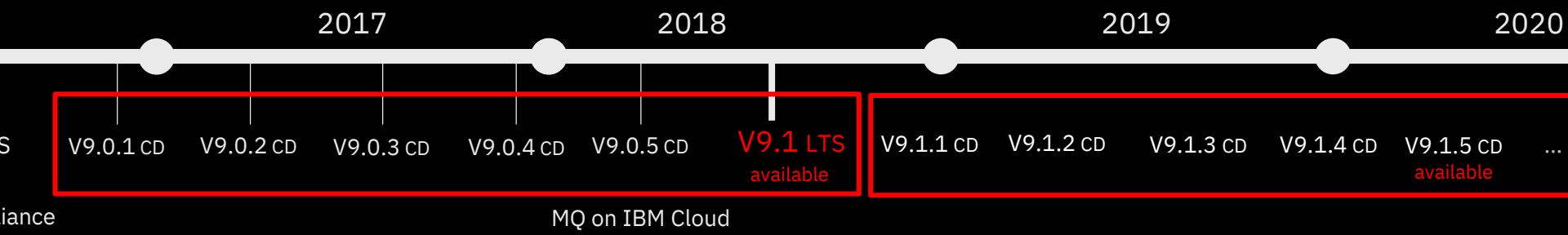
Required by those looking for fixed function.

Mix and Match

Both are available under the same license.

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

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 9.0.x CD releases is available in the long term support release **V9.1 LTS**

MQ 9.0.x CD content, now available with V9.1 LTS

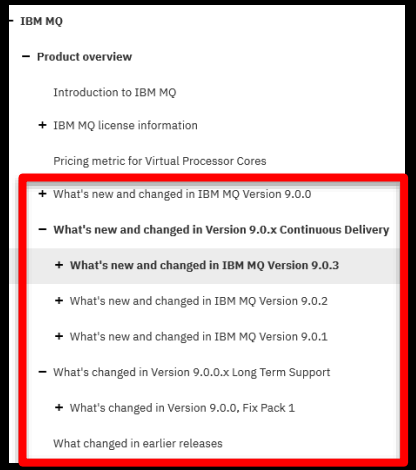
Replicated Data Queue Manager for MQ Advanced	Linear logging automation and performance	RESTful administration	Error log formatting	Web Console	RESTful messaging
MQ Appliance performance improvements	MQ JMS in CICS Liberty Profile	Salesforce bridge	AMS confidentiality performance on z/OS Advanced	Blockchain 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

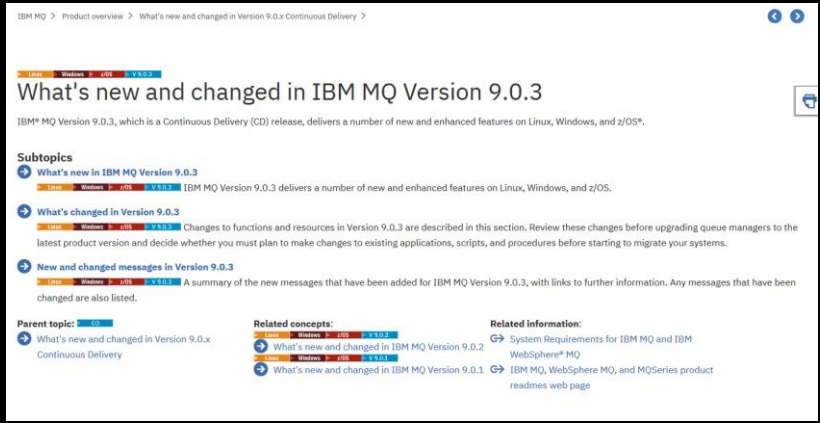
MQ 9.1.x CD content...

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
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
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
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
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	...so far

MQ release-to-release changes

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





Fault tolerance

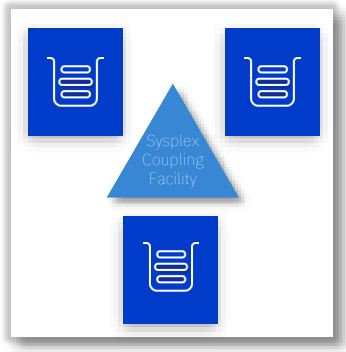
Protecting your critical data

Fault tolerance

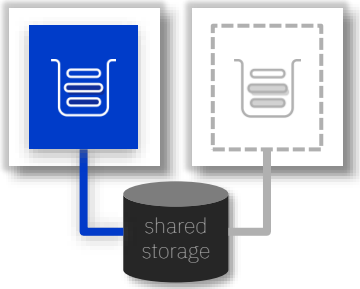
MQ delivers HA through the ability to build horizontally scaled, active-active systems and typically **active-passive HA** of the data itself*, the messages.

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.

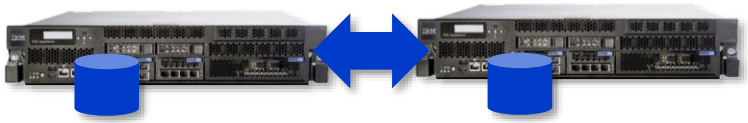
The **MQ Appliance** changed this with a fully integrated HA solution, providing built in machine to machine data replication and failover.



z/OS Queue Sharing Groups



Multi-instance queue managers and HA Cluster



MQ Appliance

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

Replicated Data Queue Managers

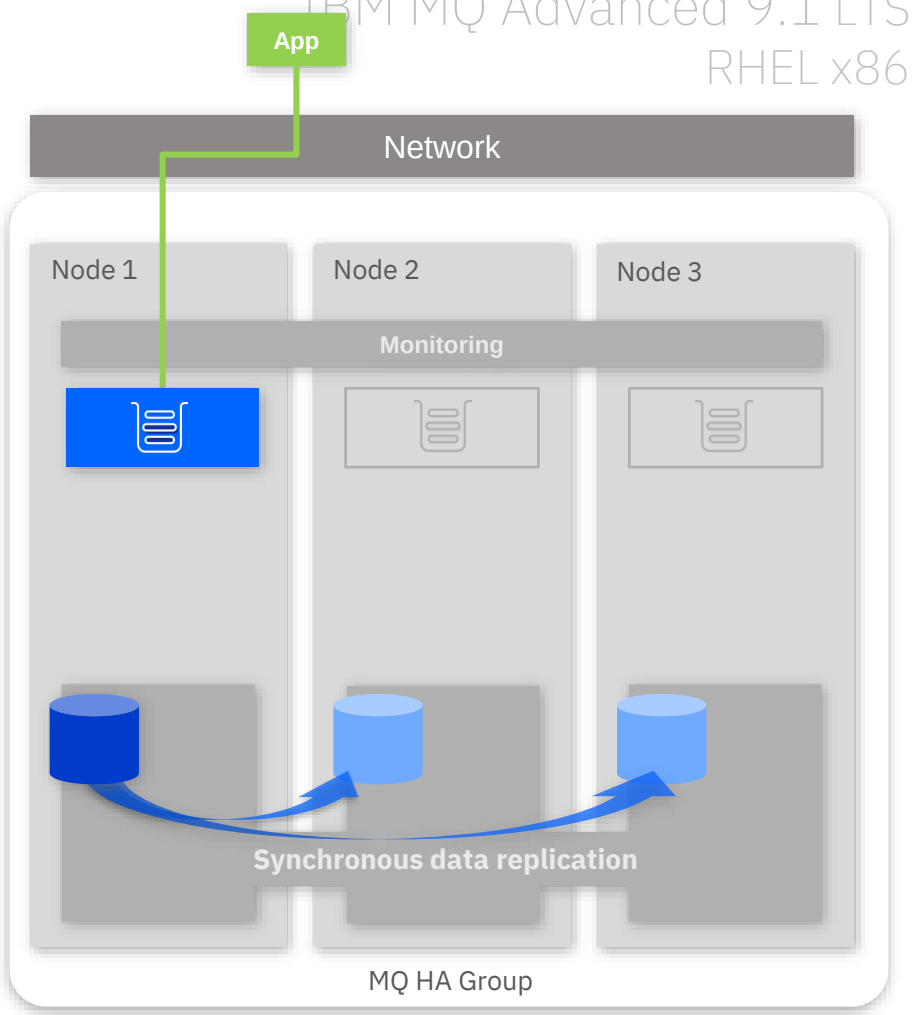
Linux only, MQ Advanced HA solution with no need for a shared file system or HA cluster

Three-way replication and monitoring for quorum support

Synchronous data replication for once and once only transactional delivery of messages

Active/passive queue managers with **automatic takeover**

MQ Advanced for RHEL x86-64



Replicated Data Queue Managers

Linux only, MQ Advanced HA solution with no need for a shared file system or HA cluster

Three-way replication and monitoring for quorum support

Synchronous data replication for once and once only transactional delivery of messages

Active/passive queue managers with **automatic takeover**

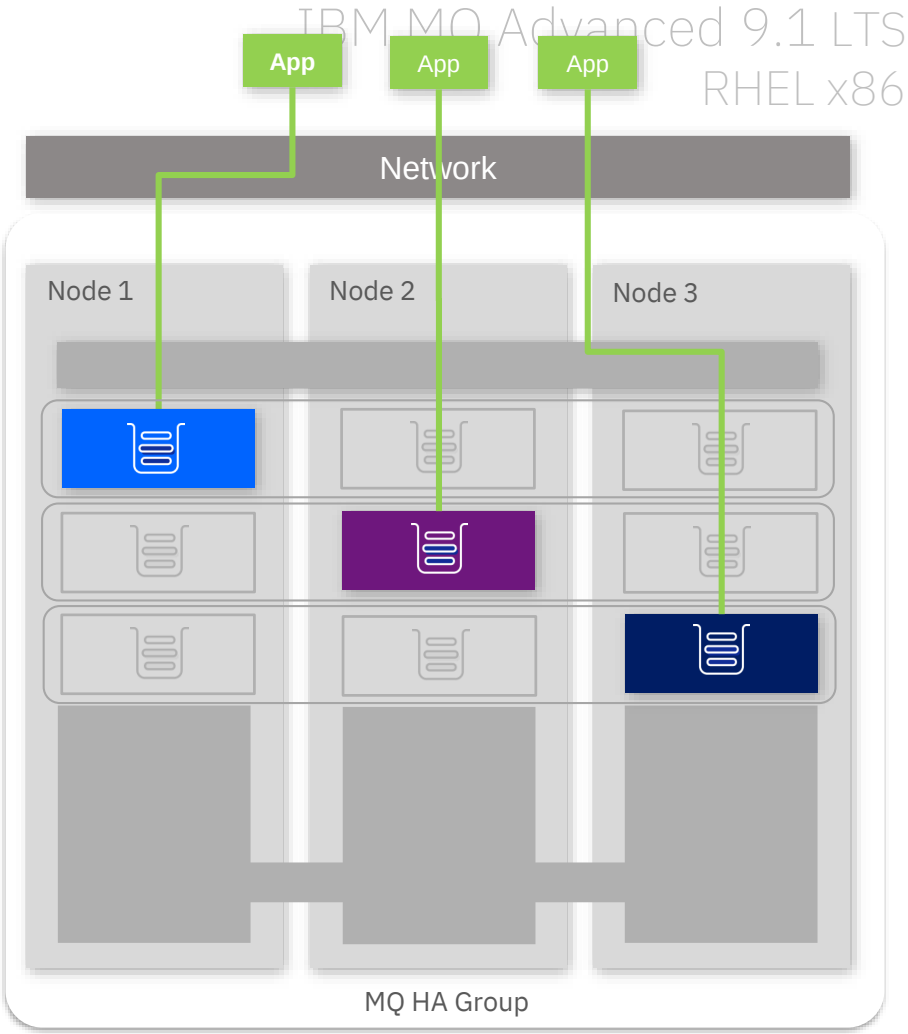
Per queue manager control to support active/active utilisation of nodes

MQ **licensing** is aligned to maximise benefits

Improvements in queue manager restart times

9.1.1 9.1.2

MQ Advanced for RHEL x86-64



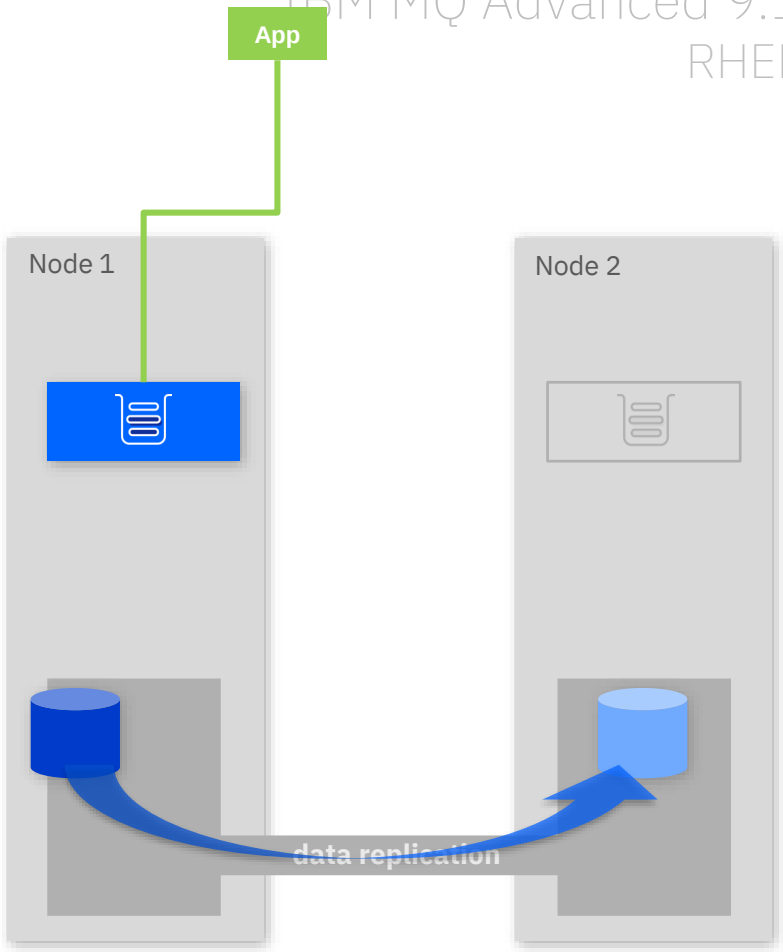
Replicated Data Queue Managers

Manual failover

RDQM also supports a looser coupled pair of nodes for data replication but with no automatic failover, often for **Disaster Recovery**

Data replication can be

- **Asynchronous** for systems separated by a high latency network
- **Synchronous** for systems on a low latency network



MQ Advanced for RHEL x86-64

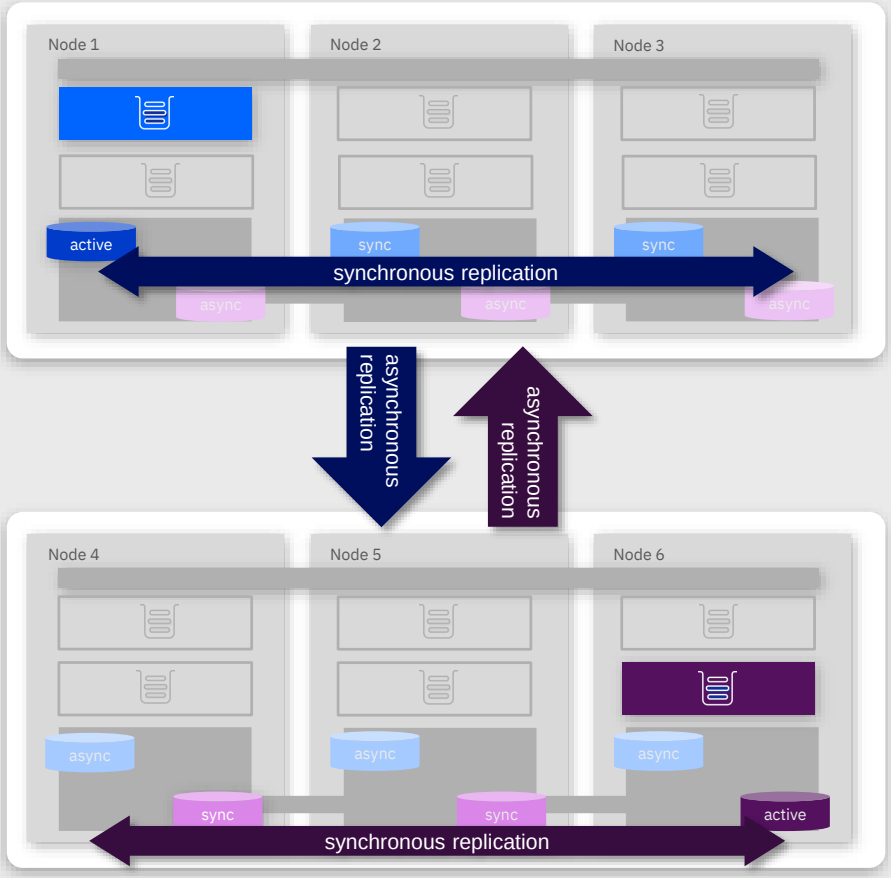
Combining HA with DR

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

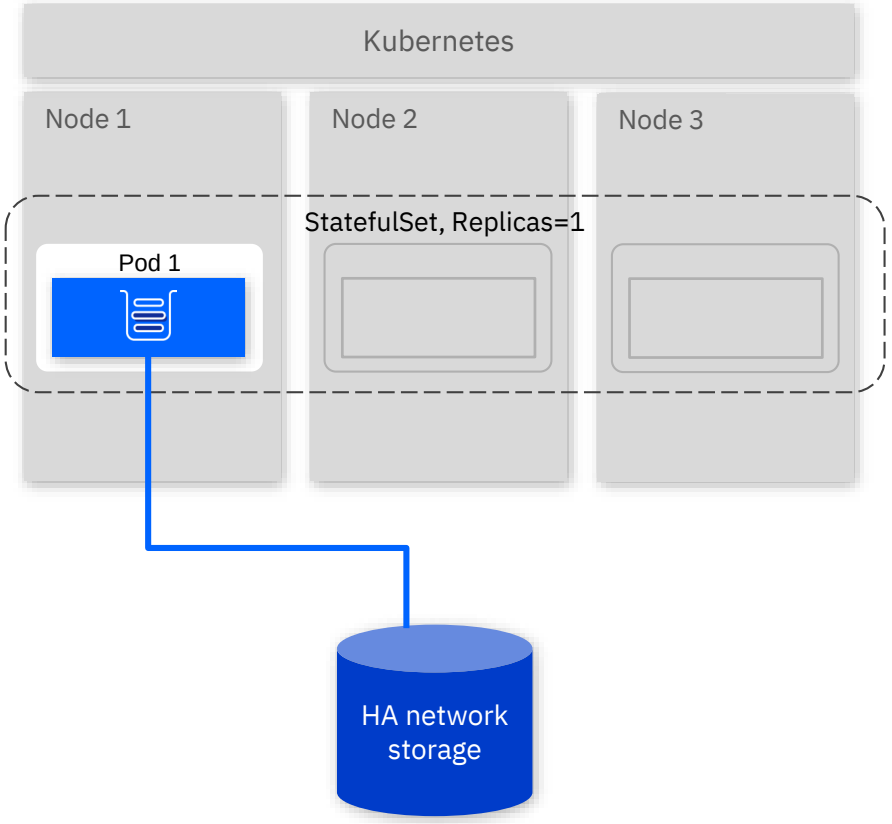


High Availability with Kubernetes

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...



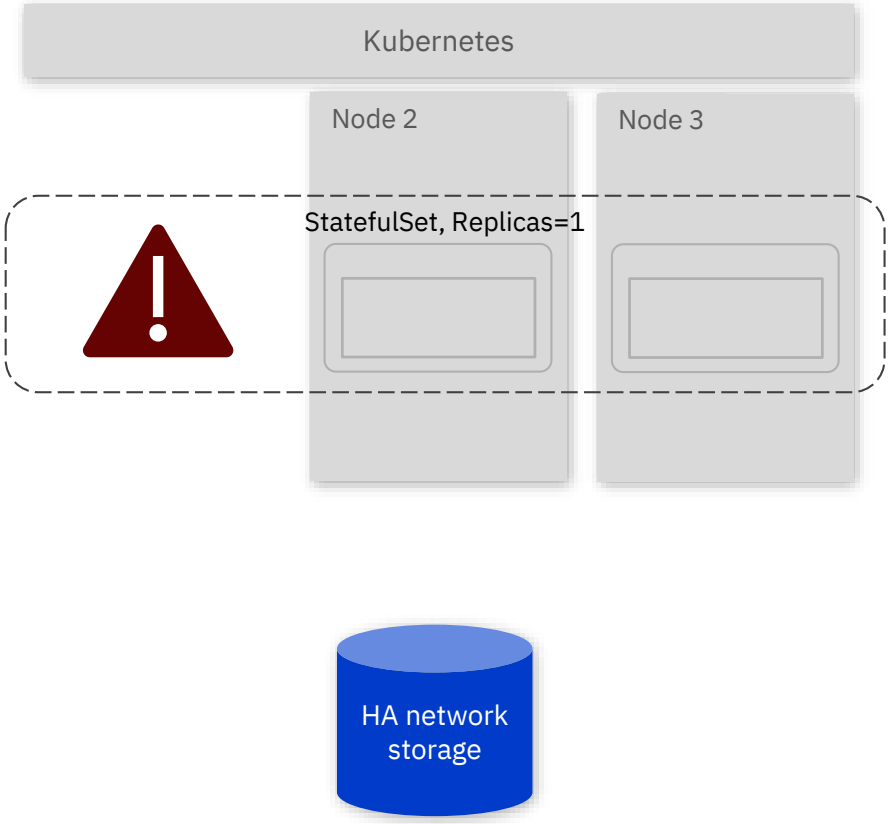
High Availability with Kubernetes

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

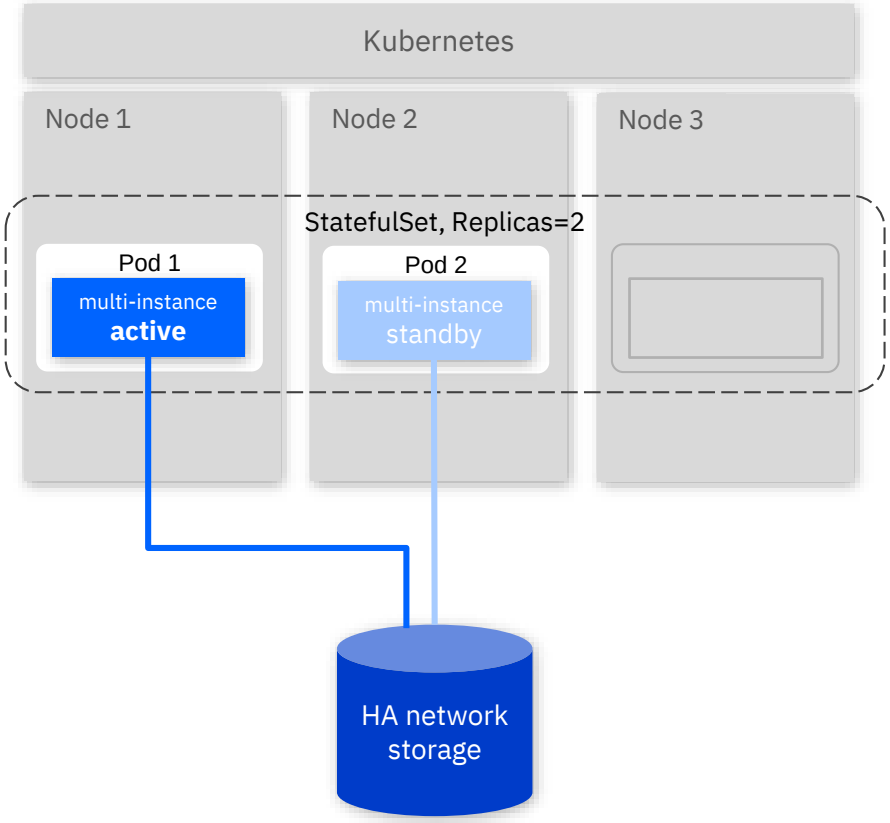
The RDQM solution does not apply to container environments

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



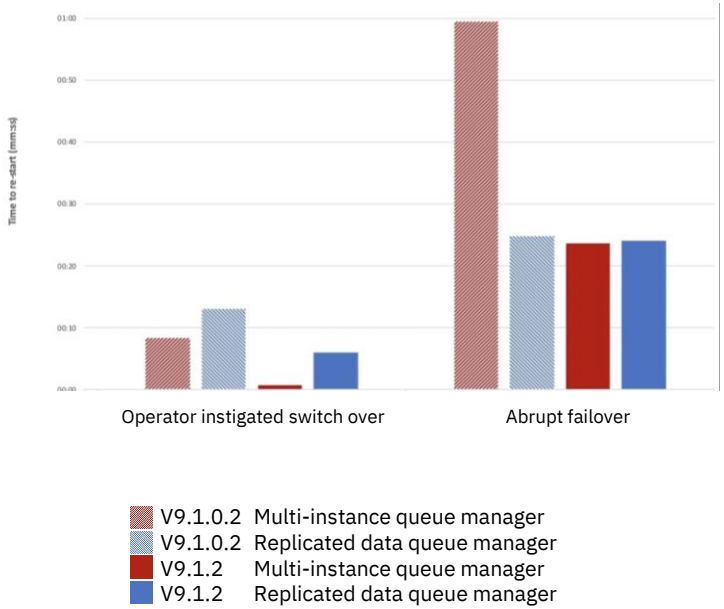
https://www.ibm.com/support/knowledgecenter/SSFKSJ_9.1.0/com.ibm.mq.ctr.doc/ha_for_ctr.htm

Increasing your availability further

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

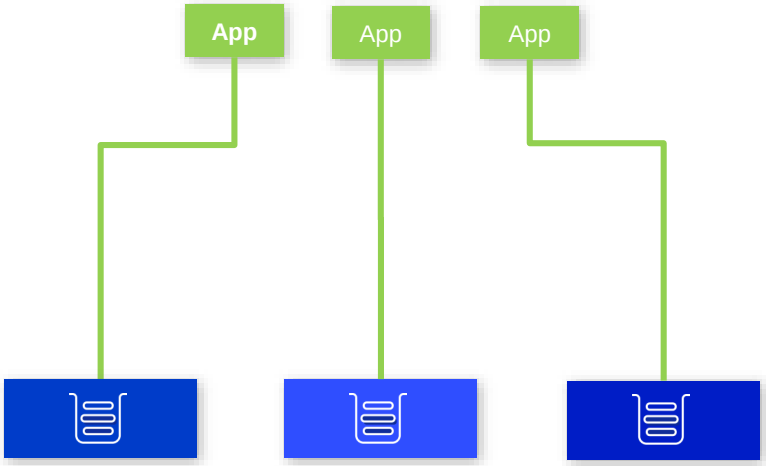
Active/active messaging

Building scalable, fault tolerant, solutions

Building scalable, fault tolerant, solutions

Many of you have built your own continuously available and horizontally scalable solutions over the years

Let's call this the *"**uniform cluster**"* pattern



Building scalable, fault tolerant, solutions

Many of you have built your own continuously available and horizontally scalable solutions over the years

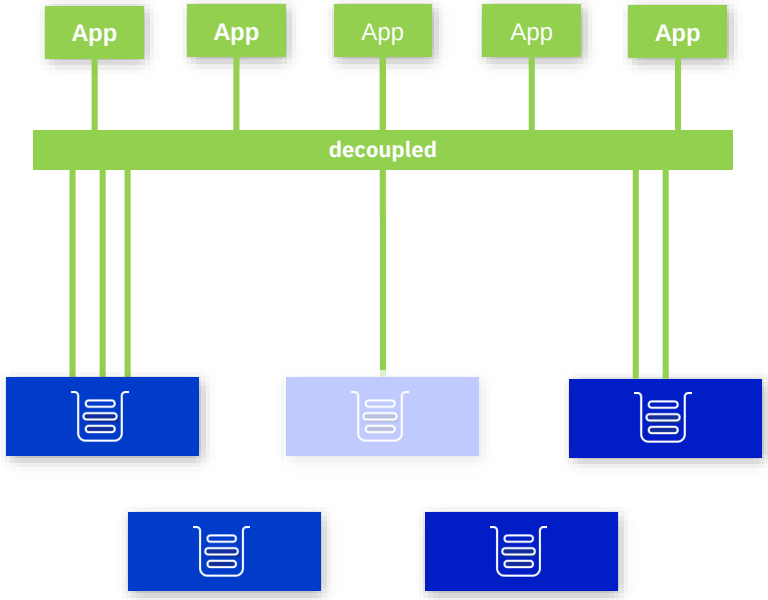
The key is to decouple the applications from the individual queue managers as much as possible.

MQ has provided you many of the building blocks -

- Client auto-reconnect
- CCDT queue manager groups
- MQ Clustering

But you're left to solve some of the problems, particularly with long running applications -

- Efficiently distributing your applications
- Ensuring all messages are processed
- Maintaining availability during maintenance
- Handling growth and contraction of scale



MQ 9.1.2 started to make that easier

For the distributed platforms, declare a set of matching queue managers to be following the ***uniform cluster pattern***

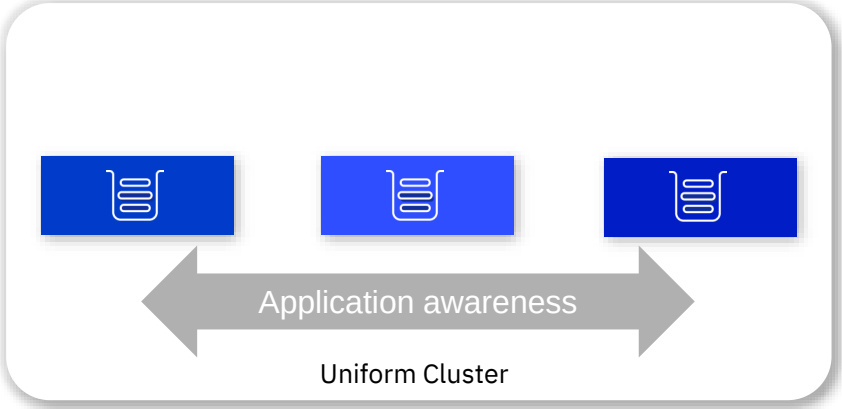
- All members of an MQ Cluster
- Matching queues are defined on every queue manager
- Applications can connect as clients to every queue manager

MQ will automatically share application connectivity knowledge between queue managers

The group will use this knowledge to automatically keep matching application instances balanced across the queue managers

- Matching applications are based on *application name* (new abilities to programmatically define this)

MQ has been incrementally rolling out increased support for this through the CD releases



Automatic Application balancing

IBM MQ 9.1.2+ CD Distributed

Application instances can initially connect to any member of the group

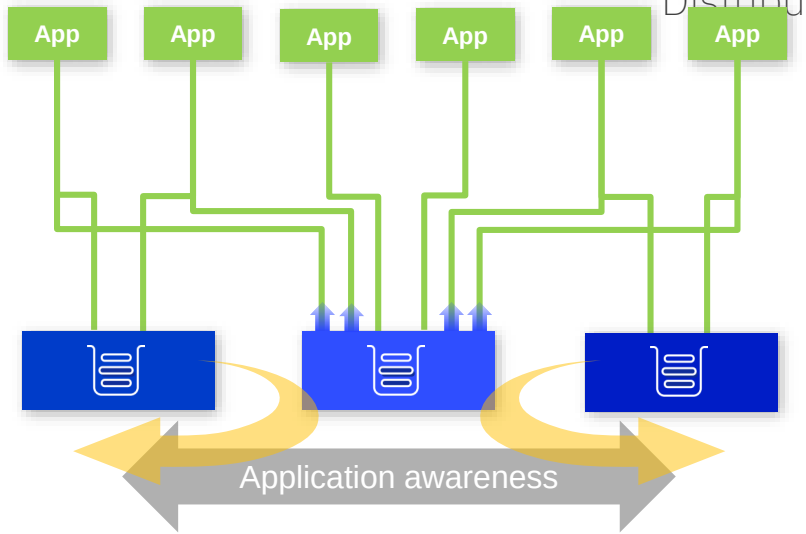
- We recommend you use a queue manager group and CCDT to remove any SPoF

Every member of the uniform cluster will detect an imbalance and request other queue managers to donate their applications

Hosting queue managers will instigate a client *auto-reconnect* with instructions of where to reconnect to

Applications that have enabled *auto-reconnect* will automatically move their connection to the indicated queue manager

- Each MQ release has extended application support, with 9.1.2 CD adding support for **C-based** applications, 9.1.3 CD added **JMS SE** support and 9.1.4 added **.NET** support

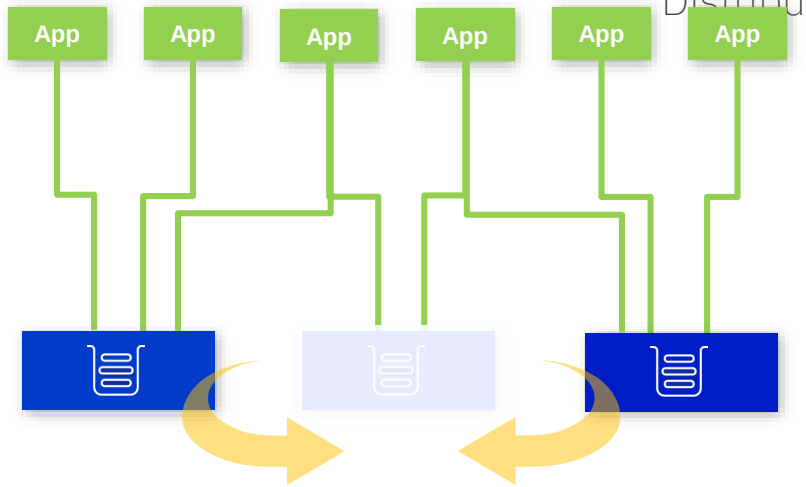


Automatic Application balancing

IBM MQ 9.1.2+ CD Distributed

Automatically handle rebalancing following planned and unplanned queue manager outages

- Existing client auto-reconnect and CCDT queue manager groups will enable initial re-connection on failure
- Uniform Cluster rebalancing will enable automatic rebalancing on recovery



Join the **MQ Beta** program to see *and influence* how the Uniform Cluster pattern and all other MQ features continue to evolve

MQ.Early.Program@uk.ibm.com

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

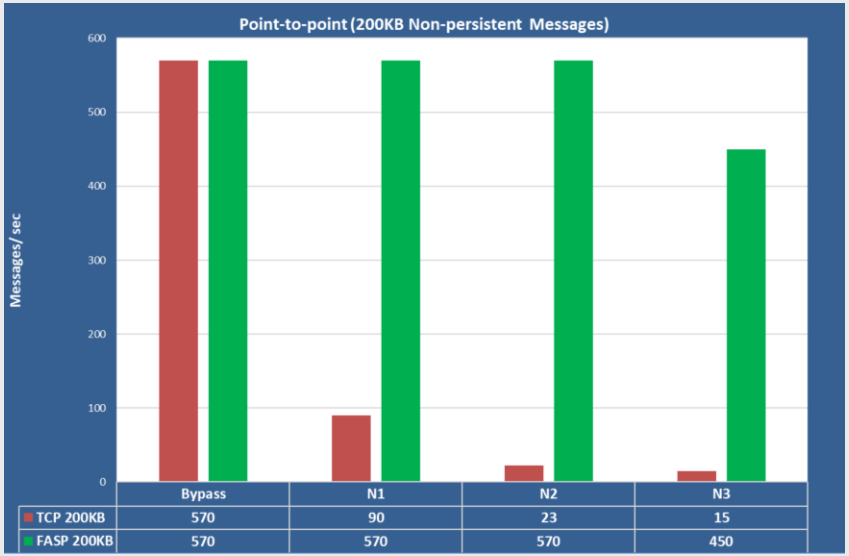
```
DISPLAY APSTATUS(*) TYPE(APPL)

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: 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

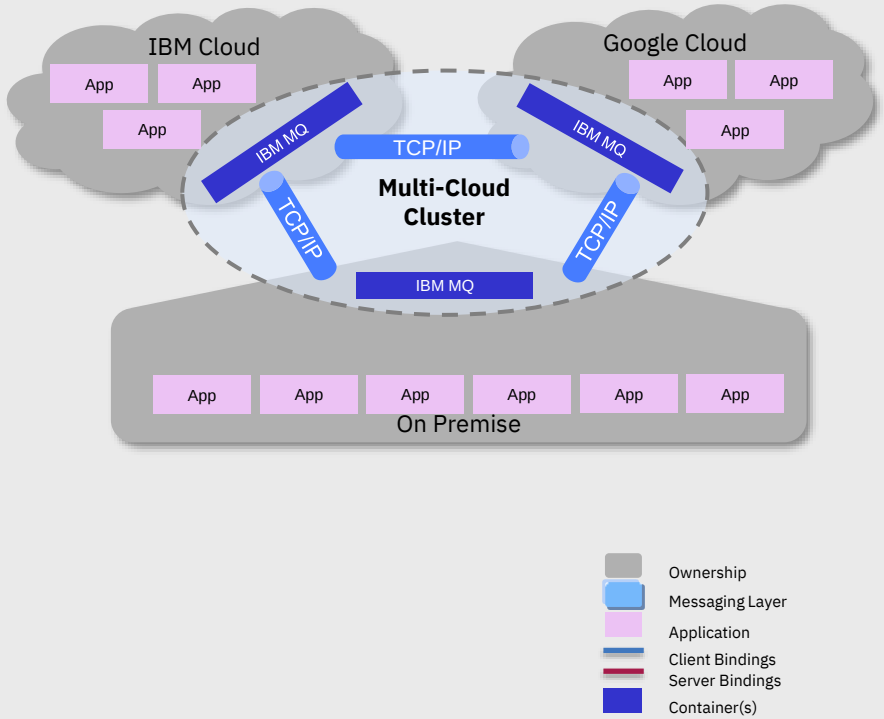
With a Hybrid Multi-Cloud Architecture connecting to external MQ networks is becoming increasingly important.

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



Managing MQ

Living with your enterprise messaging system

Building scalable and available solutions

JSON CCDT

Build your **own JSON format CCDTs**

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

Available with all 9.1.2+ clients

C, JMS, .NET, Node.js, Golang clients

Excellent for use with **Uniform Clusters**

Queue size control

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.

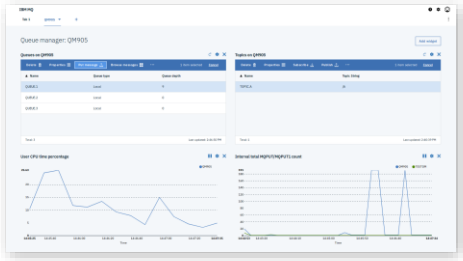
The maximum supported size is now 255TB

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

Making management simpler

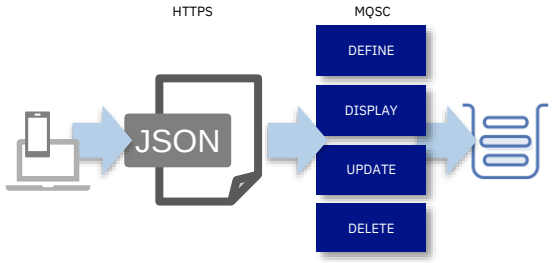
Web console

Simple to use, web based administration for Distributed, z/OS, Cloud and the Appliance



RESTful administration

Administer and manage your queue managers over HTTPS



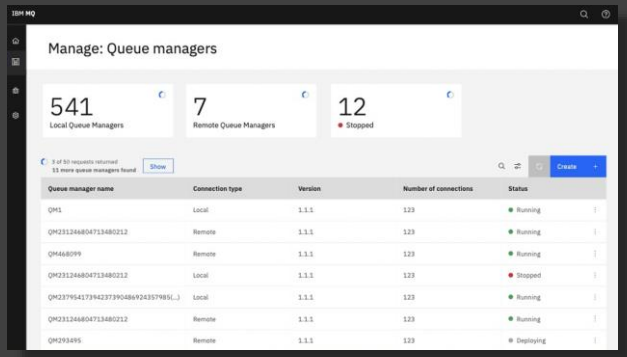
Logging and monitoring

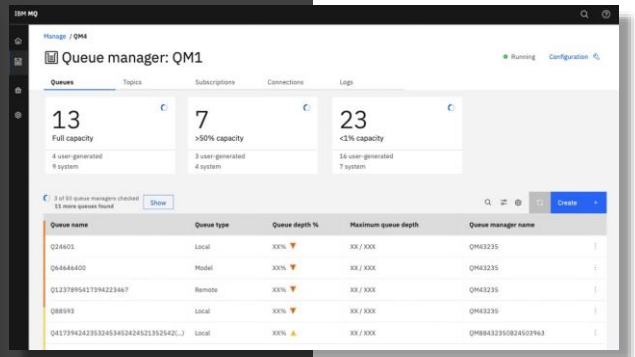
Simplify the streaming of logs and metrics for centralized storage and analyzes

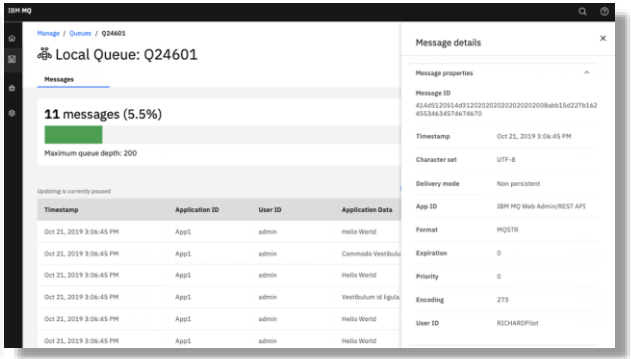
New Web Console

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







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

MQSC command, in JSON

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"
  }
}
```

Managing channel CipherSpecs

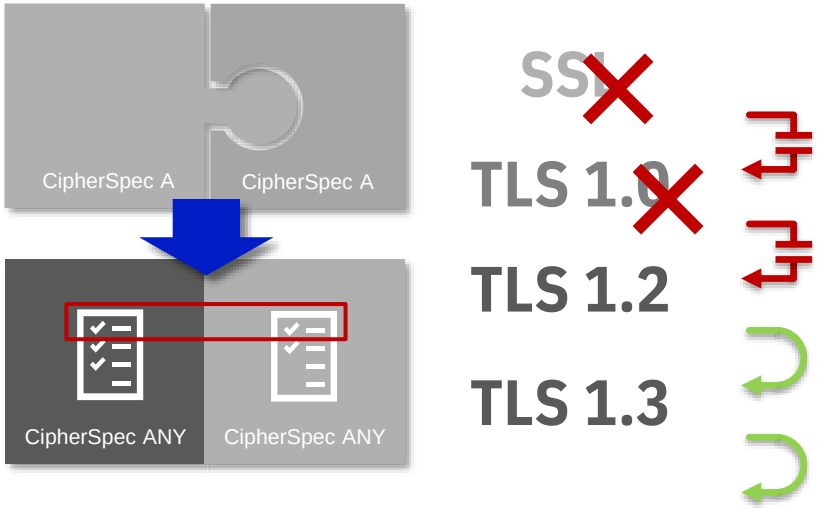
Making it easier to keep up-to-date with ever changing ciphers, simplifying migration

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.

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 strongest CipherSpec available to both ends

(MQ 9.1.1 added ANY_TLS12 on all platforms. MQ 9.1.4 added the 1.3 variants on Distributed platforms)

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



Advanced Message Security

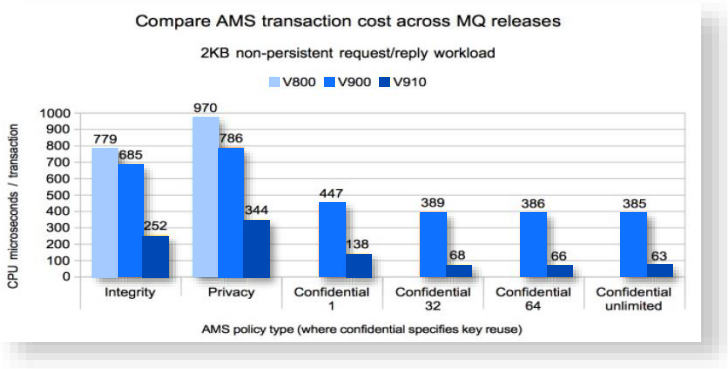
IBM MQ Advanced 9.0-9.1 LTS
All platforms

End-to-end message encryption is now even easier to adopt, thanks to ever improving performance

MQ 9.0 LTS started to drive down the overhead of AMS with the introduction of the *confidentiality* policy

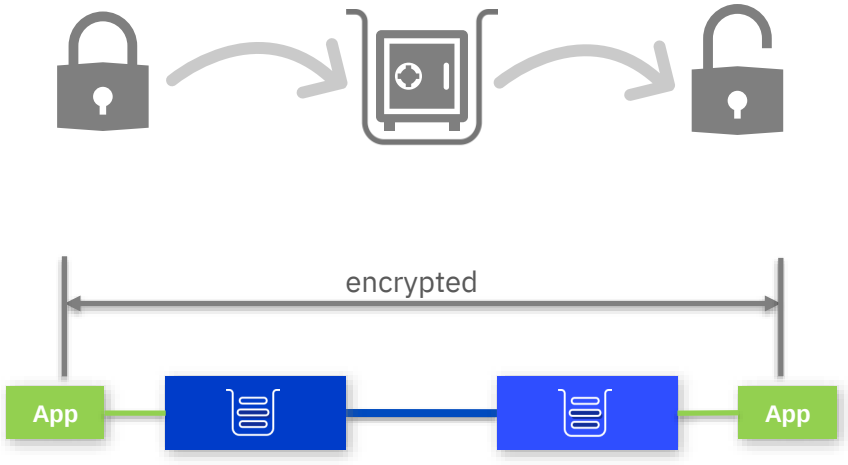
MQ 9.0.1 CD was the point that all platforms benefitted fully from those improvements

AMS on z/OS can now be as efficient as TLS just on the channels



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

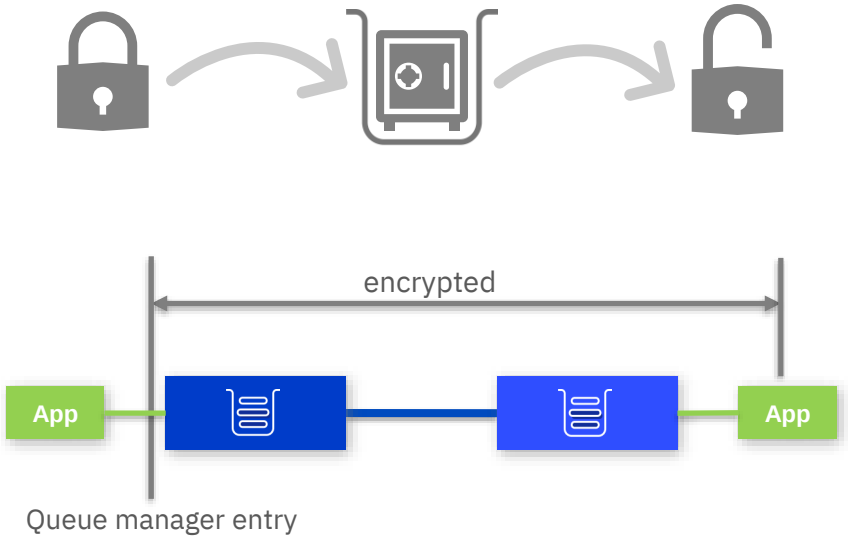


Application to application

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 has always had client level interception to apply AMS policies once messages reach or leave their first queue manager

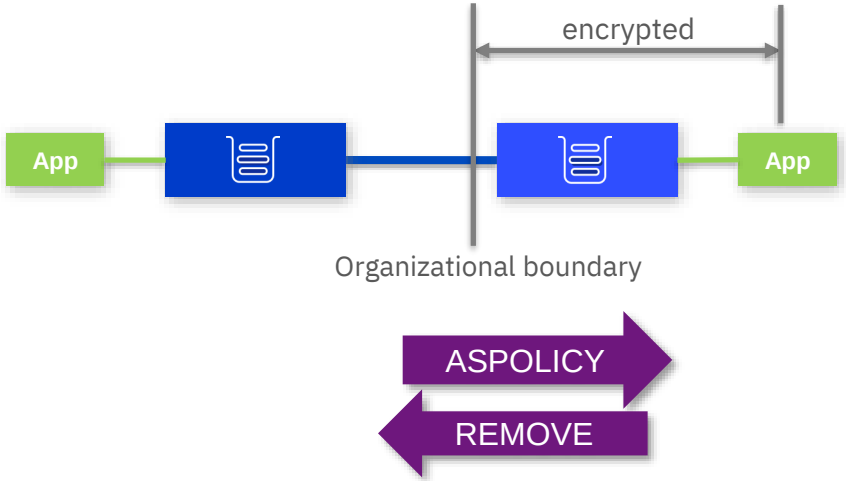


Advanced Message Security

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 has always had 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



New sender and receiver channel property, SPLPROT controls the use of AMS policies for inbound and outbound messages

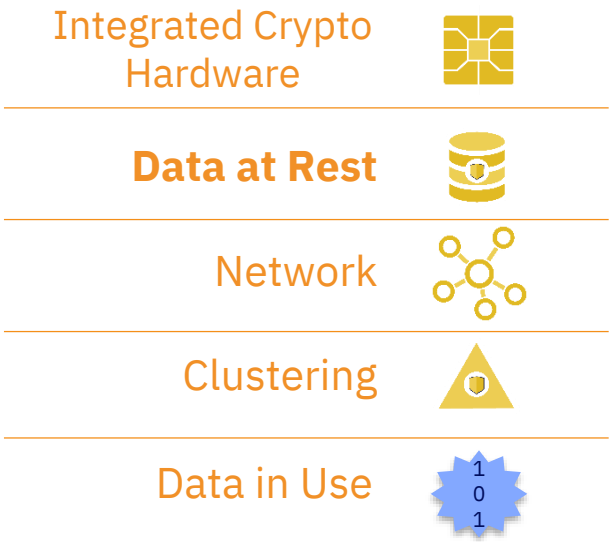
Dataset encryption with MQ on z/OS

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



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

Managed File Transfer

MFT manages your file transfers, and now it's even easier for you to manage MFT, including...

IBM MQ Advanced 9.1 LTS +
All platforms

Simplified MFT Agent licensing
No need to track individual agents with MQ Advanced queue managers

Redistributable MFT agent
Simply download and unpack

Failed transfer timeout
Automatically stop transfers after repeated failures

Resource monitor backups
Simple, single command to backup and restore resource monitors

MQ 9.1 LTS

MFT agent, transfer and resource monitor monitoring through REST

File transfer initiation and resource monitor management through simple REST calls

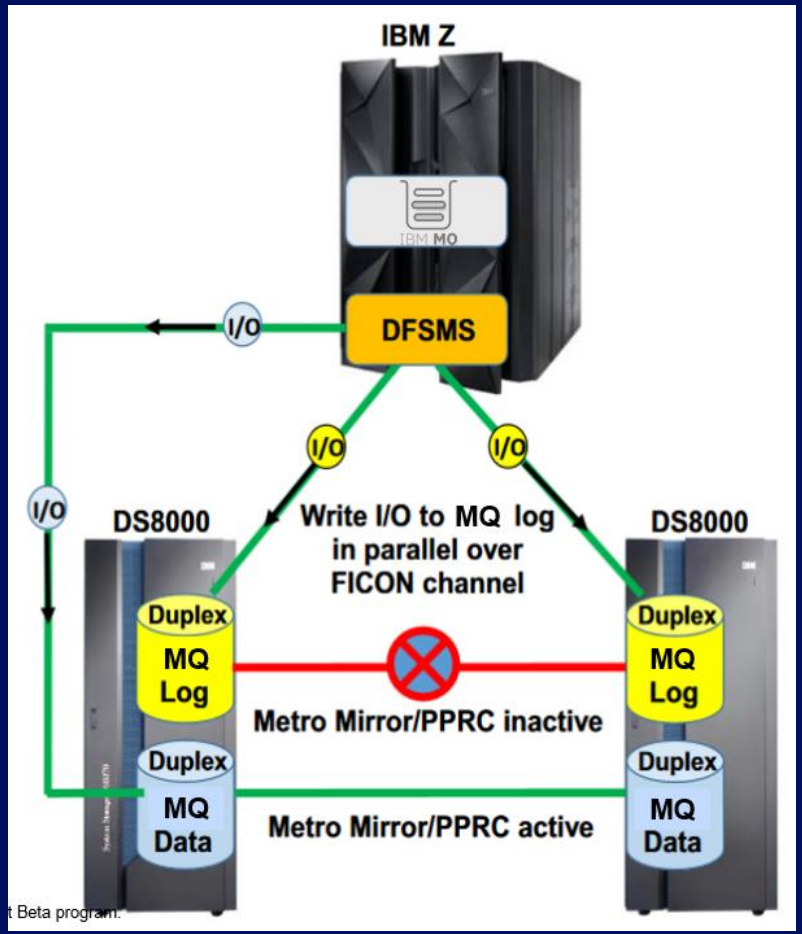
Highly available active/standby MFT agents

MQ 9.1.x CD

z/OS zHyperwrite

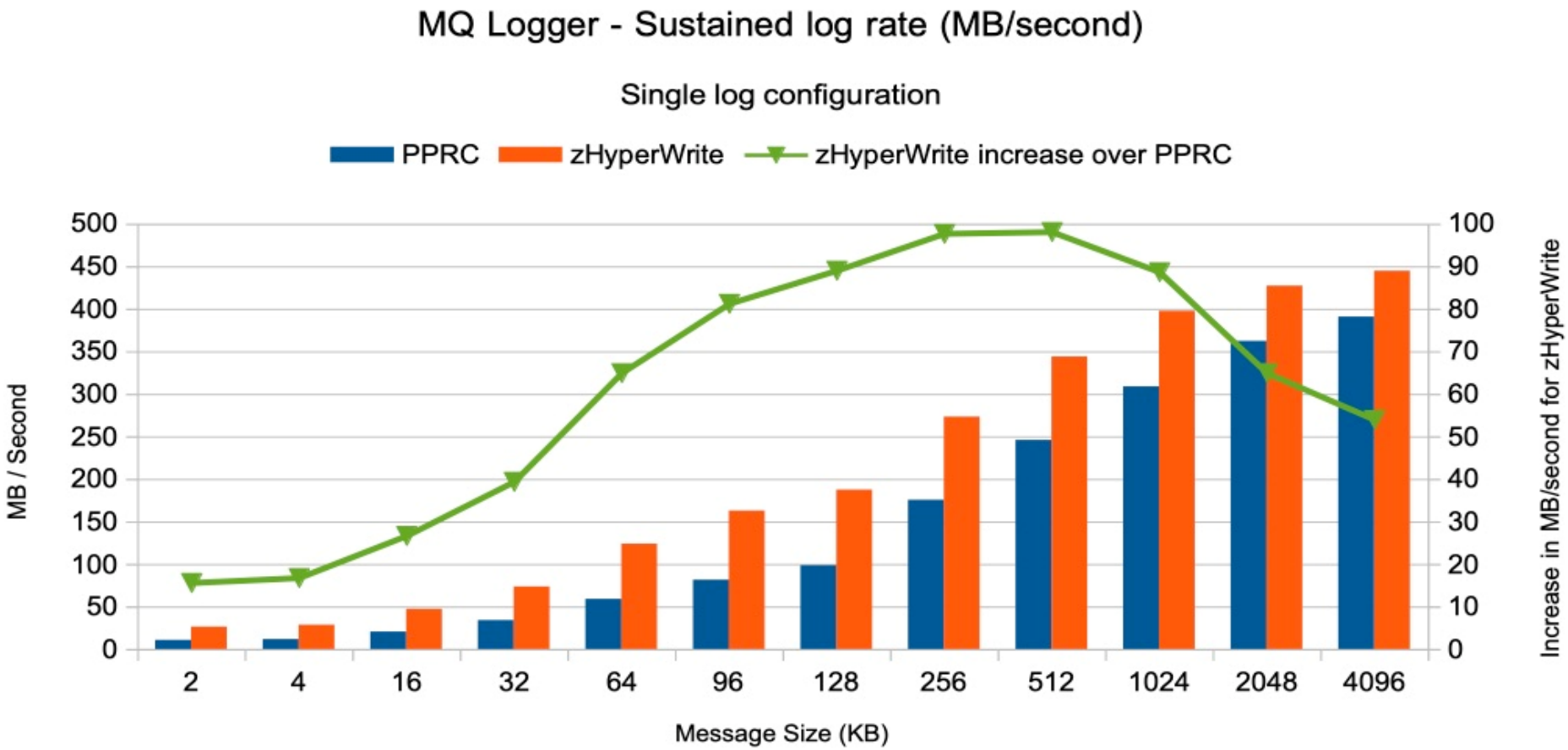
Reduces the cost of using PPRC (Metro-Mirror) to synchronously replicate log data by issuing the write to the primary and secondary copies of the data at the DFSMS (Media Manager) level.
This allows the writes to occur in parallel instead of in series.

- 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.

z/OS zHyperwrite



Helping developers

Making it easy to build MQ into your applications

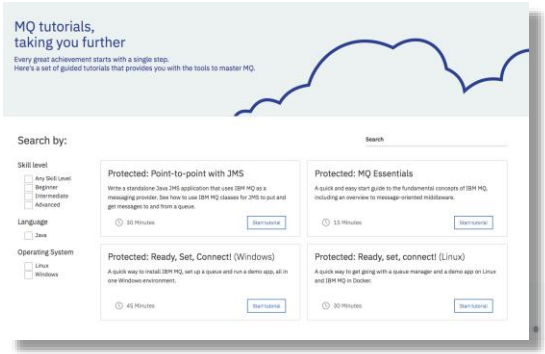
Getting Started

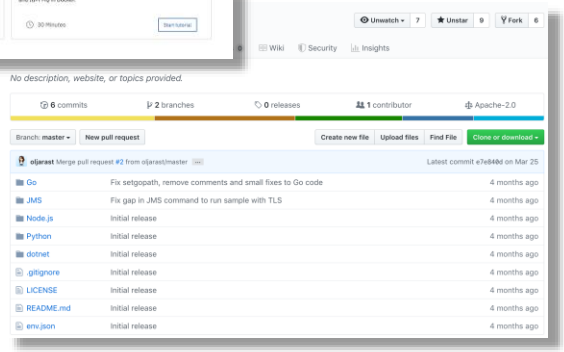
Teach yourself the basics of MQ

ibm.biz/learn-mq

Build on top of simple samples

ibm.biz/mq-dev-patterns





...and prove your skills

Demonstrating the simplicity of MQ

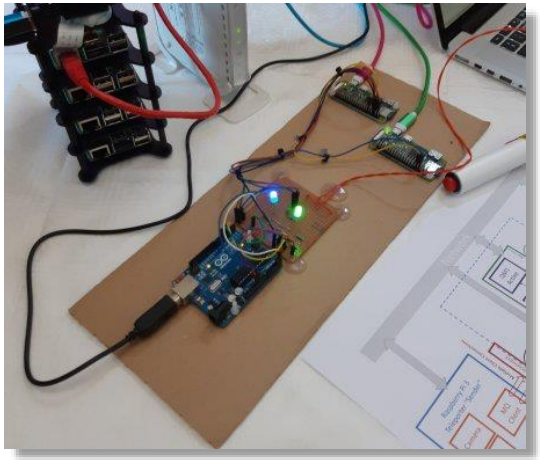
There’s nothing like flashing lights and wires to grab people’s attention. We want everyone to know how easy it is to write messaging applications and how powerful MQ is in supporting them

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

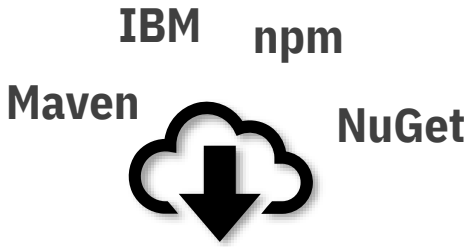


Developing applications

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

- Pull Java directly from the Maven repository since MQ 9.0.4 CD
- MQ 9.1.1 CD added the **SDK** to the MQ redistributable client
- The redistributable client is now available directly, no need to log into IBM
- MQ 9.1.4 CD added .NET clients to NuGet

ibm.biz/MQdownloads



Develop your applications on the platform of your choice with the addition of the MacOS version of the 9.1.1 MQ client and SDK for Developers

ibm.biz/mqmacos

(The MQ for MacOS toolkit includes runmqsc)

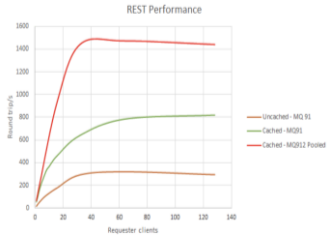
Writing new applications

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

Messaging and Events

Messaging

There are essential capabilities needed of every messaging solution

Essential capabilities:

High Availability

Scalability

Security

Management

Critical data exchange: work that needs to be done

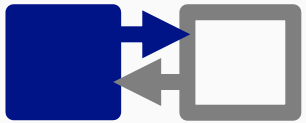
Critical applications demand assured asynchronous interactions

Messages typically represent commands, queries and operations

The message is a way to pass control from the originator of the message to the consumer

Essential capabilities:

Once and once only Delivery



Fine grain messaging

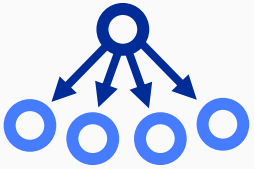
Event Driven: building scalable microservices

Microservices increases the need for communication. API-based interactions can build fragile and unscalable tight bonds between components

Publishing and subscribing to events relaxes the coupling of microservices

Events are messages that communicate that something has occurred

Essential capabilities:



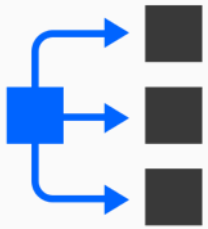
Topics and Subscriptions

Event Streaming: the expanding need for messaging

Event Streaming brings data together from disparate sources, enabling even more responsive and engaging experiences for a wider set of users

Efficient cloud and analytics applications utilize local decoupled buffers of event data

Essential capabilities:



Scalable Subscription

Stream History

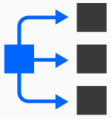
The right tool for the job

Critical data exchange ← → Event driven ← → Event streaming

Once and once only Delivery

Fine grain messaging

Topics and Subscriptions



Scalable Subscription

Stream History

IBM MQ

Focused on message exchange and transactions

Supported connectivity

Apache Kafka with IBM Event Streams

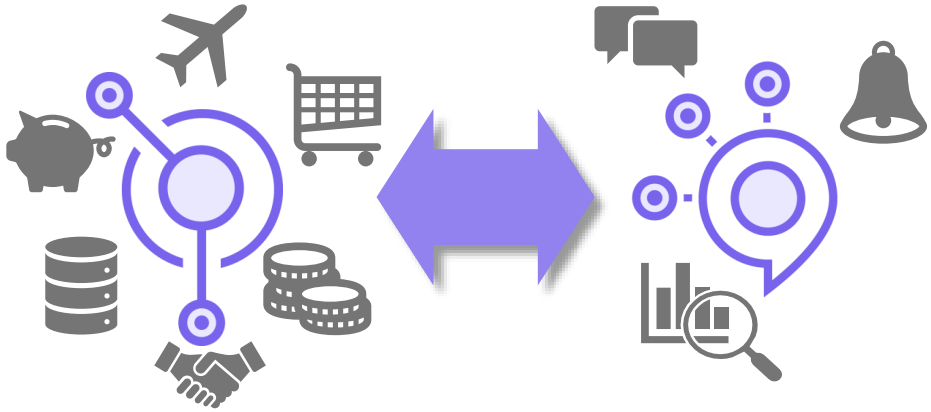
Focused on streaming of events

IBM MQ with IBM Event Streams

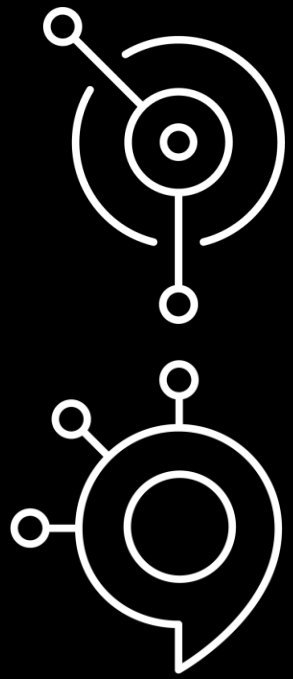
IBM MQ connects mission-critical systems, requiring **transactional**, **once-only delivery**

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



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



Thank you