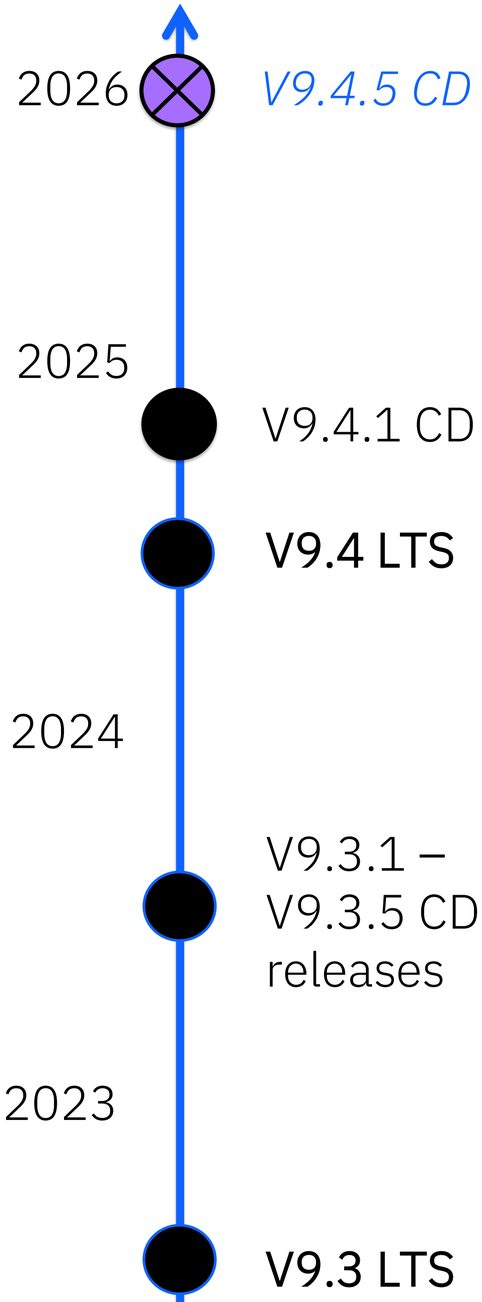
28529: What's new in IBM MQ for z/OS
Up to MQ 9.4.5

Toby Keegan
Software Engineer, IBM MQ for z/OS
toby.keegan@ibm.com

Where are we now?

Support Lifecycle



CD

New Continuous Delivery (CD) versions of MQ are released approximately every 4 months, incrementally introducing new capabilities and fixes.

LTS

Approximately every 2 years, a new LTS version of MQ is released, rolling up many of the CD capabilities into a release with 5+1+3 support.

5+1+3

LTS releases of IBM MQ come with 5 years of support, a 1-year critical fix extension, and usage and existing fixes for 3 years.

What’s new in IBM MQ 9.4? (up to 9.4.5)

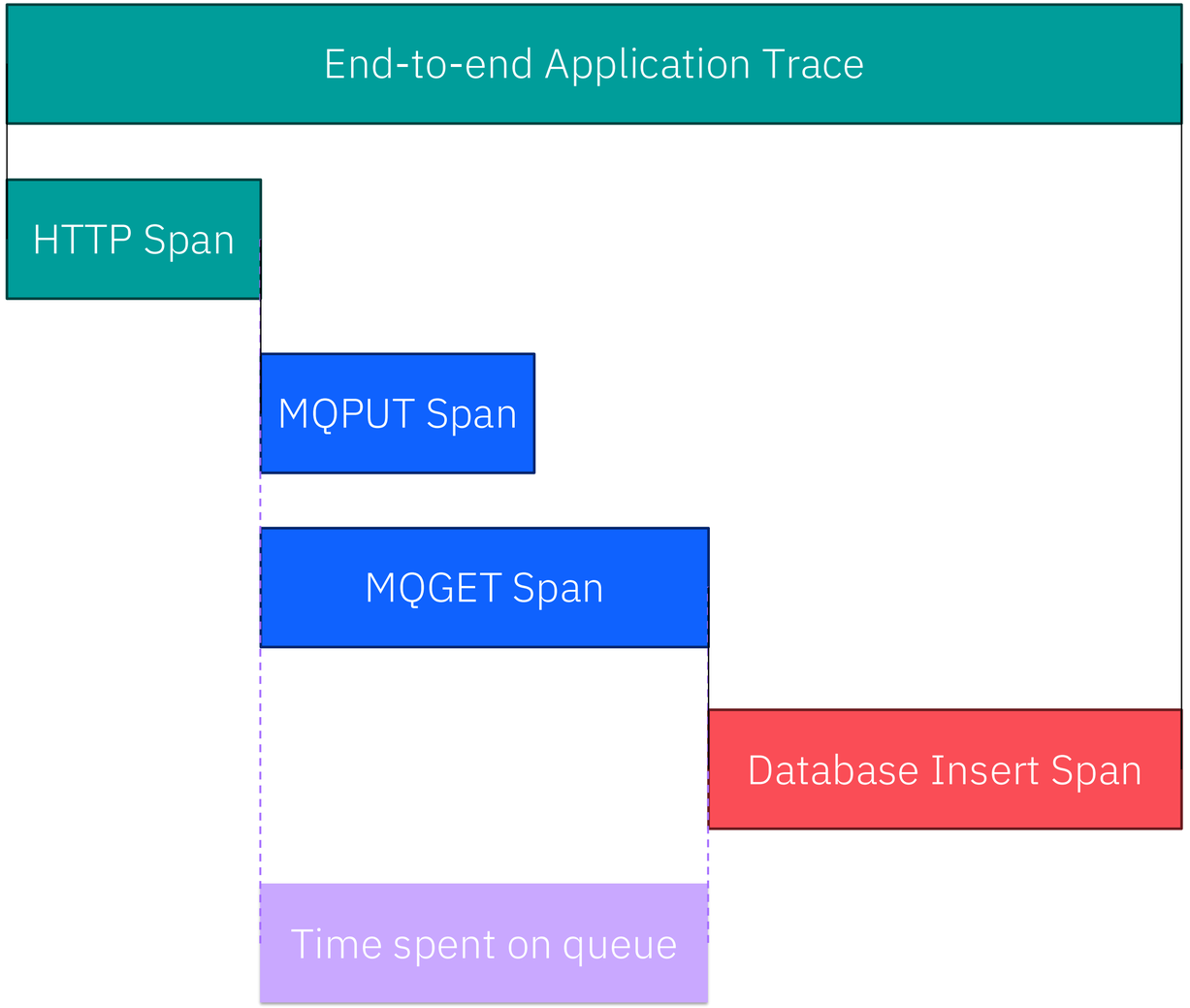
.NET 6 Libraries	First-class CAPEXPRY	RDQM on RHEL 9	MQIPT MQCSP password protection	64-bit Channel Initiator	SMF queue statistics enhancement	Streaming queues for shared queues	Console observability	Instana transaction tracing with CP4I in OCP	ANY* generic CipherSpecs for AMQP
Enhanced DIS QMSTATUS	TLS 1.3 for managed .NET clients	MQ Console Application View	Java modular application support	Simplified REST Messaging API security	Native HA entitlement with MQ Advanced	SMF116 STREAMEDN for streaming queues	Appliance DR between two HA pairs	Developer Essentials badge for AWS	Restricted MQ Operator permissions
MQ Console, object, and application association linkage	AMQP performance enhancement	Remote REST messaging	JSON web tokens initial support	Native HA auto recovery of damaged objects	Comprehensive per-queue SMF stats	Appliance CLI SSH certificate authentication	Appliance disk space monitoring	ARM Developer container for MacOS	Configurable LogFilePages with MQ Operator
Smarter recovery log media image scheduling	MQ Console Overview Landing Page	Tuning options for log I/O warnings	Supported Kafka Connectors with MQ Advanced	Enhanced Appliance Support for SSH signatures	Smarter JMS Uniform Cluster auto balancing	Exactly-once Kafka Connectors	Simplified Java and JMS topologies for JMS	Automated installs with Ansible Galaxy	Message property support for CSQ1LOGP
Stand-alone IBM MQ Web Server Install	Open Telemetry tracing	Expanded JWT support for JMS applications	MQ Console application and network observability tabs	Updated compiler support for AIX C applications	Simpler zHyperWrite enablement on z/OS	Appliance secure backup and restore	AMQP samples	Developer tutorials for Uniform Clusters	MacOS client toolkit in homebrew
JWKS for token authentication	Updated cipher support for MFT	Persistent Volume flexibility for containers	LZ4 compression	TLS certificate bypass option	.NET 8 support	Automate MQ fix pack version upgrades	Non-production tags for IBMi	RedHat OCP odd version support	zHyperLink Support for log writes

IBM MQ | Observability

OpenTelemetry Tracing

*Available for MQ on **all** platforms*

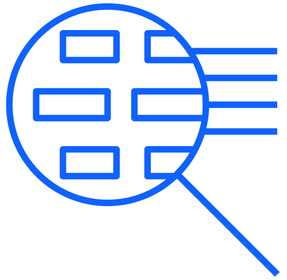
- OpenTelemetry is an observability framework and toolkit designed to create and manage trace, metrics, and log data.
- It is vendor and tool agnostic – allowing it to be used with a broad variety of observability solutions.
  - Commercial offerings like IBM Instana
  - Open-source tools like Jaeger and Prometheus



OpenTelemetry Tracing

*Available for MQ on **all** platforms*

Diagnose problems across multiple systems fast, using data to pinpoint where issues are.



Gain insights to optimise performance end-to-end, ultimately improving customer experience.

Futureproof – open and likely to become a cross-industry de facto standard for observability.

CLOUD NATIVE COMPUTING FOUNDATION

What you need to know:

- Instana Exit shipped with IBM MQ as a supporting program
- Support on MQ for z/OS is native and does not require any additional installation

9.4.3

- IBM MQ for z/OS can participate in OpenTelemetry traces, emitting spans and propagating context for point-to-point messaging operations
- Support for automatic context propagation with CICS TS 6.3

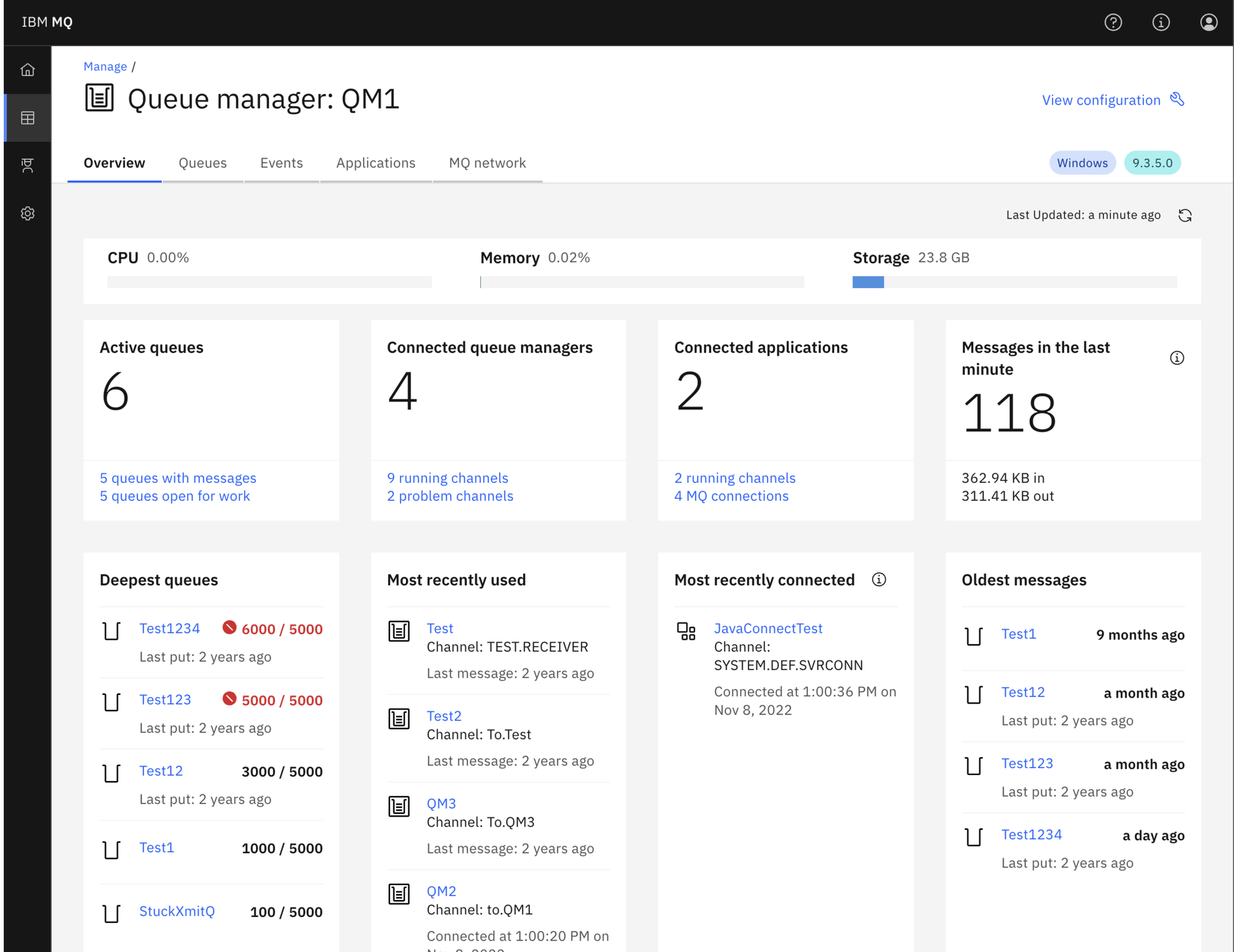
9.4.5

- Additional support for traces in publish/subscribe messaging operations
- Support for automatic context propagation with IMS 15.5 (with APAR PH62728) via the IMS Bridge

MQ Console Dashboards

The IBM MQ Console allows common administrative tasks to be performed through a graphical user interface.

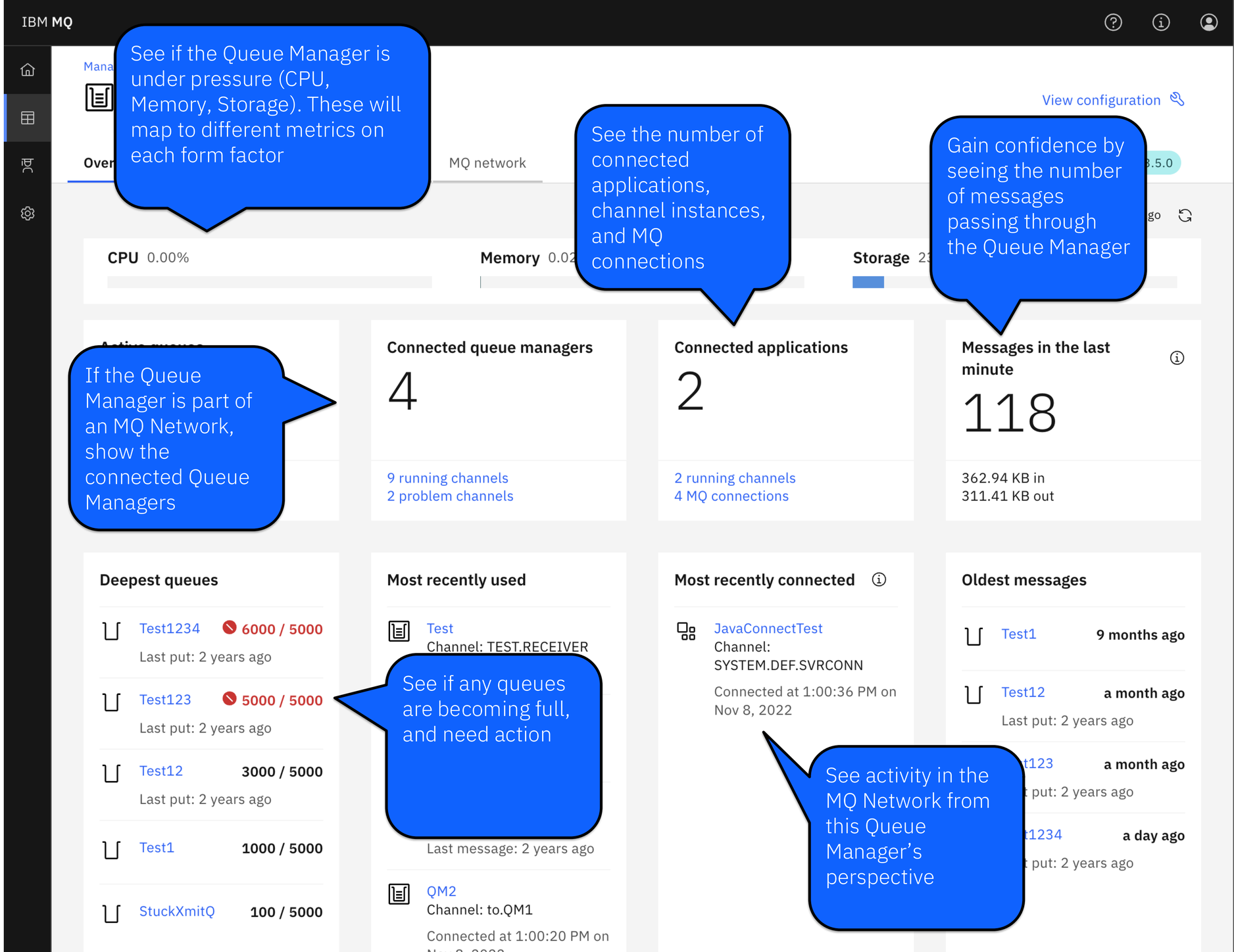
The new Overview tab eliminates the manual look-up of key information so you can quickly understand the state of your queue manager, and act on emerging issues before they become problems.



MQ Console Dashboards

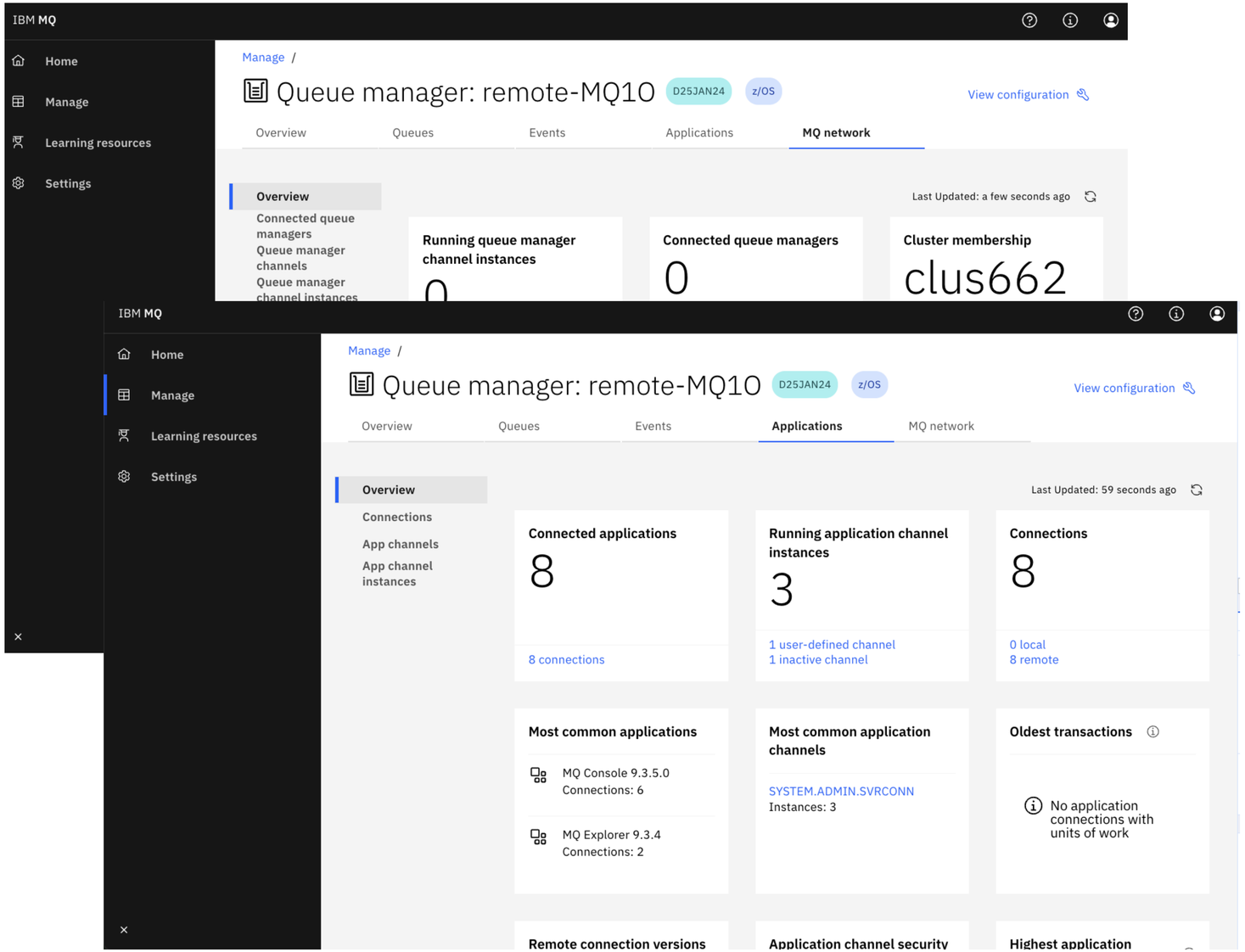
The IBM MQ Console allows common administrative tasks to be performed through a graphical user interface.

The new Overview tab eliminates the manual look-up of key information so you can quickly understand the state of your queue manager, and act on emerging issues before they become problems.



MQ Console Dashboards

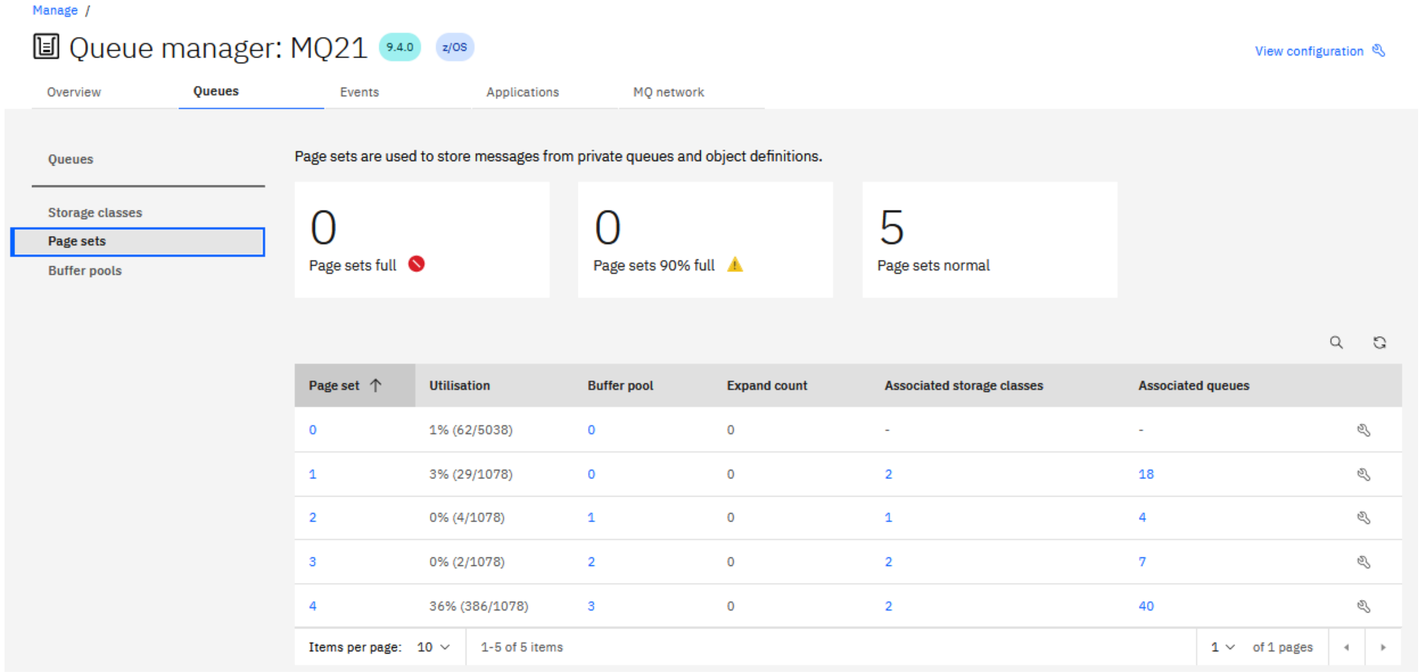
- The new Applications and MQ network tabs give you a quick view of the connected applications, and other connections in your MQ network.
- Easier for admins and developers to perform real-time analysis of:
  - Application connectivity
  - Queue manager to queue manager networking
- Reduce the time to resolution of issues
- Allows less skilled users to carry out analysis

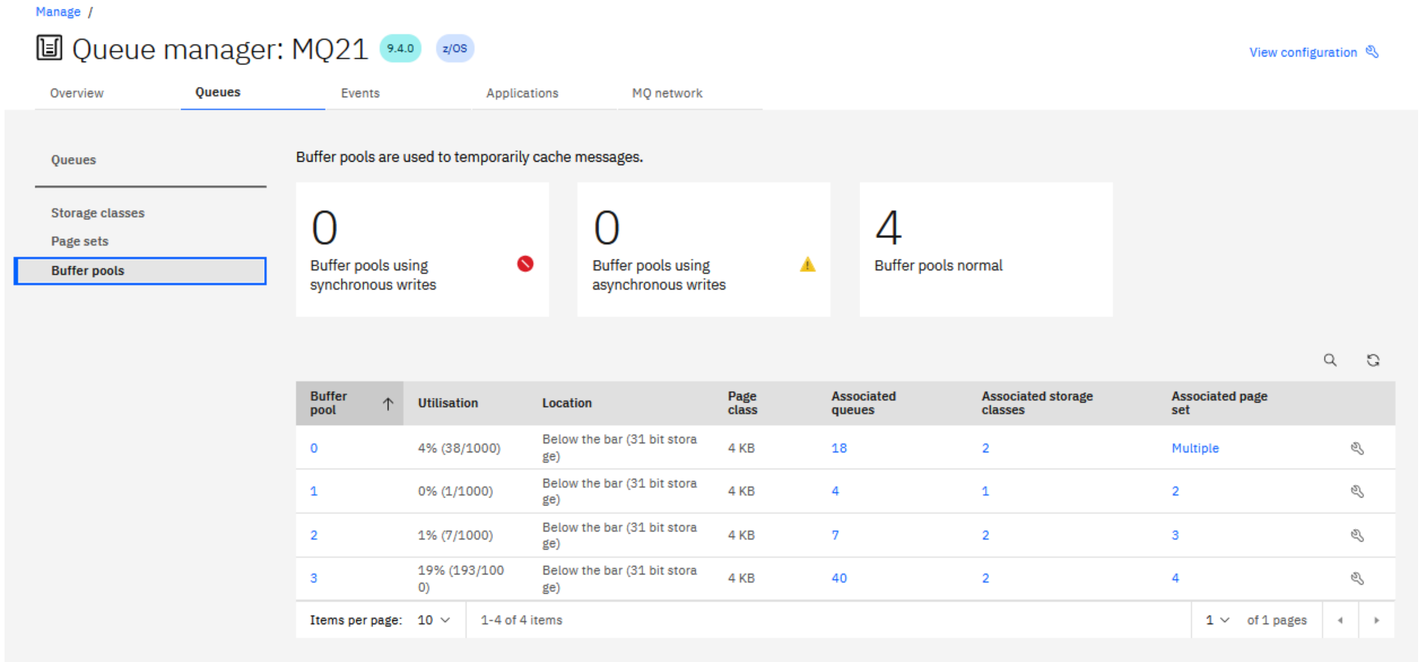


IBM MQ | Administration & Application Development

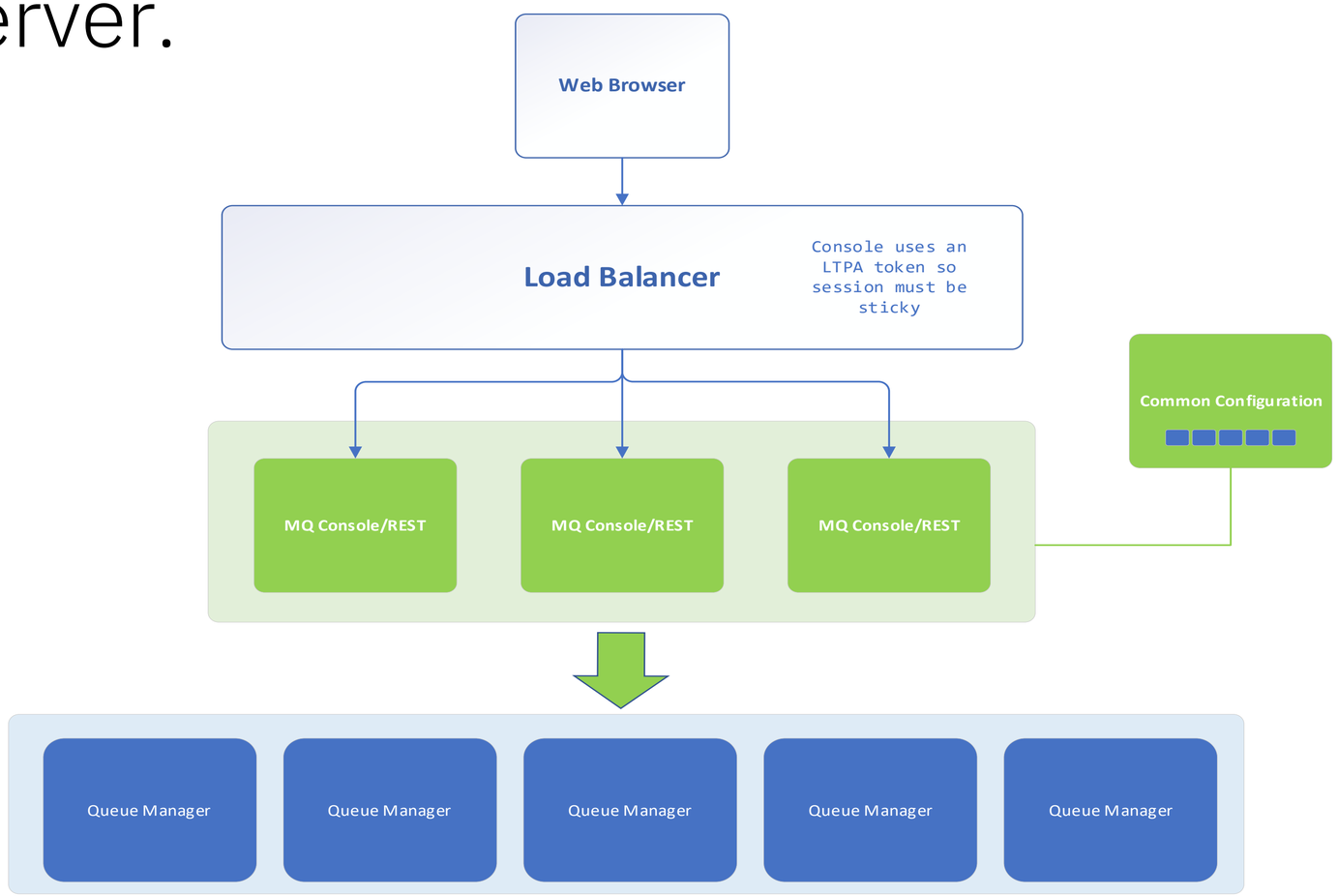
MQ Console for z/OS

- The MQ Console has been enhanced with some z/OS specific pages:
  - Storage classes
  - Page sets
  - Buffer pools
- Allows you to use a single tool to configure and monitor your queue manager.
- The queue manager dashboard also shows a summary of buffer pool and page set usage.





The MQ Console can now be deployed as a standalone web server.



Decoupled

Rapidly configure remote queue managers without the need for a full MQ installation.

Backwards Compatible

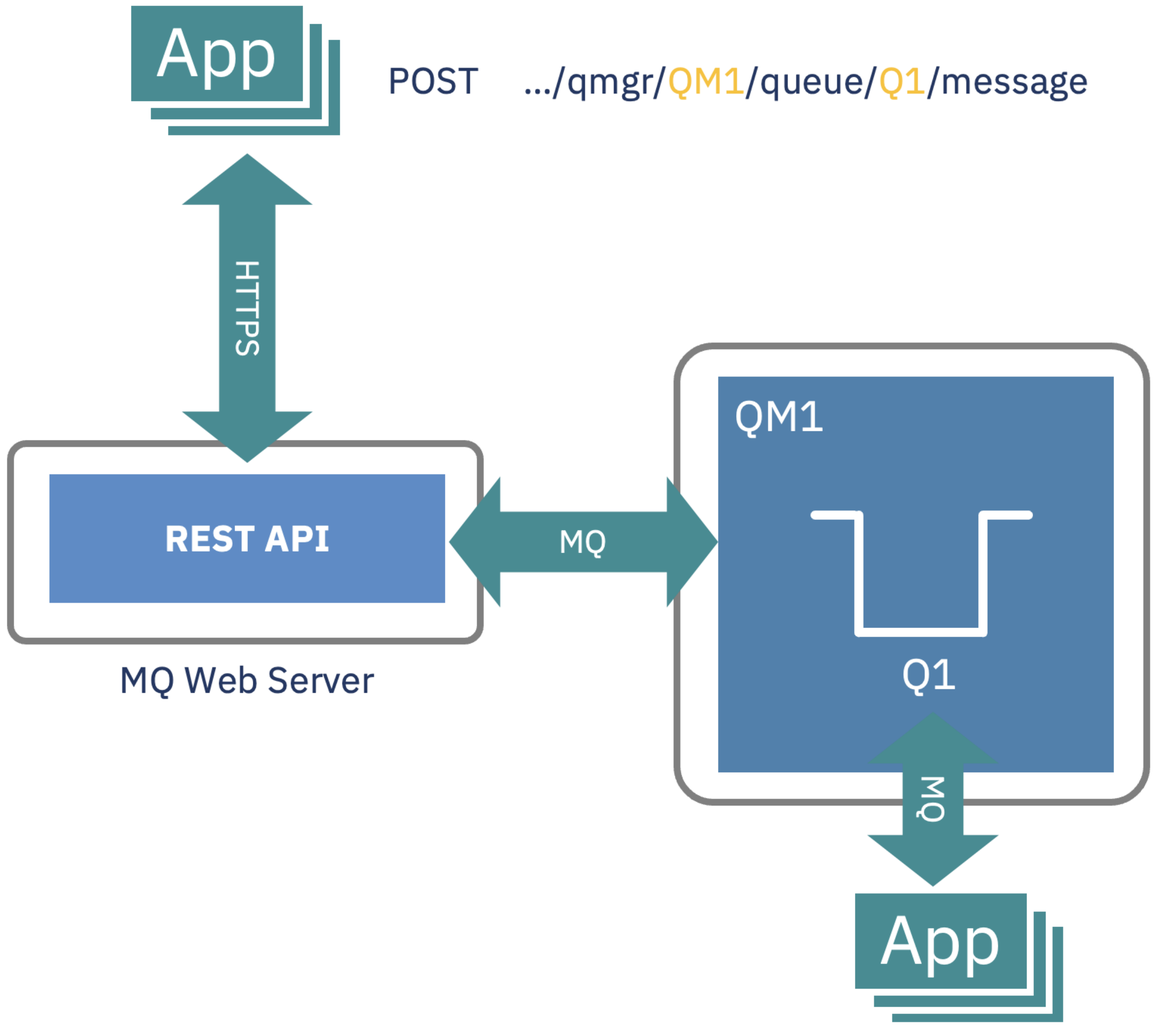
Update the MQ console independently to your queue managers., enabling your production MQ systems to remain on an LTS release, whilst your console can be updated with features from continuous delivery releases.

Scalable and Highly Available

Deploy one or more web servers behind a load balancer, all sharing a common configuration to create a scalable, fault-tolerant administrative experience.

REST Messaging

Remote queue manager support



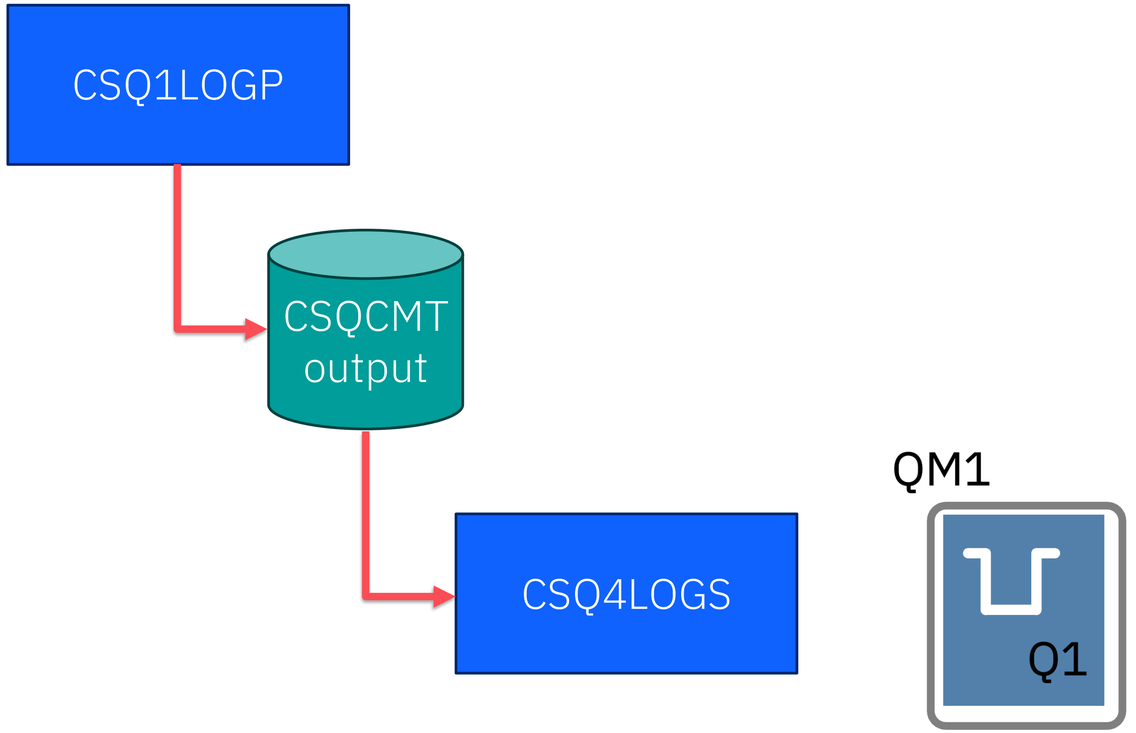
- The REST Messaging API can now be used with remote queue managers, enabling you to interact with any queue manager in your estate, at any version
- Previously, it could only be used with queue managers local to the MQ Web Server. This meant:
  - Same installation on distributed platforms
  - Same LPAR and MQ Version as the MQ web server on z/OS

CSQ1LOGP EXTRACT now supports message properties.

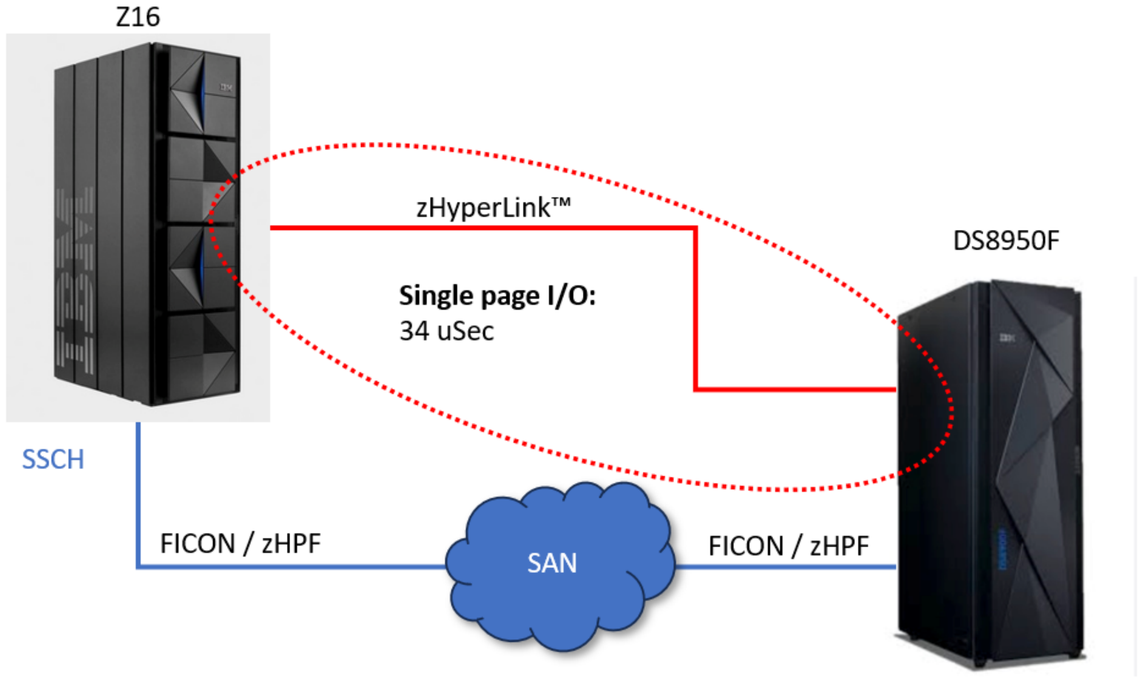
```
//STEP1 EXEC PGM=CSQ1LOGP,REGION=0M
//STEPLIB DD DISP=SHR,DSN=thlqual.SCSQANLE
//        DD DISP=SHR,DSN=thlqual.SCSQAUTH
//        DD DISP=SHR,DSN=thlqual.SCSQLOAD
//ARCHIVE DD DISP=SHR,DSN=xxx.yyy.A0030620
//        DD DISP=SHR,DSN=xxx.yyy.A0030621
//SYSPRINT DD SYSOUT=*
//SYSSUMRY DD SYSOUT=*
//CSQCMT DD DSN=xxx.MSGS.COMMIT,
// DISP=(NEW,CATLG),SPACE=(CYL,(1,10),RLSE),UNIT=SYSDA
//SYSIN DD *
EXTRACT(YES) SUMMARY(NO)
URID(xxxxxxxxxxx)
/*
//STEP2 EXEC PGM=CSQ4LOGS,PARM=('QM1 REPLAY'),REGION=0M
//STEPLIB DD DSN=thlqual.SCSQANLE,DISP=SHR
//        DD DSN=thlqual.SCSQAUTH,DISP=SHR
//        DD DSN=thlqual.SCSQLOAD,DISP=SHR
//FILEIN DD DSN=xxx.MSGS.COMMIT,DISP=SHR
//SYSDBOUT DD SYSOUT=*
//SYSABOUT DD SYSOUT=*
//SYSPRINT DD SYSOUT=*
//SYSOUT DD SYSOUT=*
/*
```

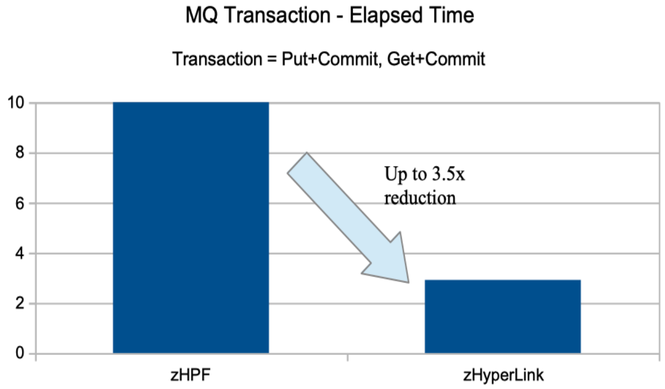
CSQ1LOGP EXTRACT has been enhanced to fully extract messages with message properties going into an RFH2 header.

This means that, if needed, those messages can be replayed, including their properties, using CSQ4LOGS, just like any other message.



zHyperLink allows for synchronous MQ log writes to enable your queue manager to do more.

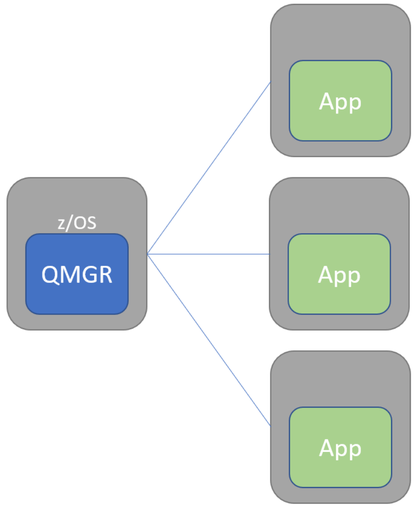




- Traditional I/O is much slower than the CPU, so log writes are performed asynchronously
- With this model, the log writer sleeps until the write completes. This performs well, but can be delayed by re-dispatching of the log task and CPU cache eviction.
- zHyperLink is a direct CEC to DASD connection which is fast enough for log writes to be performed synchronously, spinning the CPU until the write completes.
- This improves persistent message performance, and reduces transaction commit times.
- NB: This can result in higher CPU usage by the logger task.

64-bit Channel Initiator

- Previously, server-connection channel messages were staged by the channel initiator in a set of buffers in 31-bit storage.
- Large messages would limit the maximum number of active channels.
- Now, with 64-bit storage in the channel initiator, a larger number of channels can connect at the same time.
- Samples have been changed to have MEMLIMIT=2G
  - 2GB of 64-bit storage by default



For 104 concurrent clients sending 10MB messages

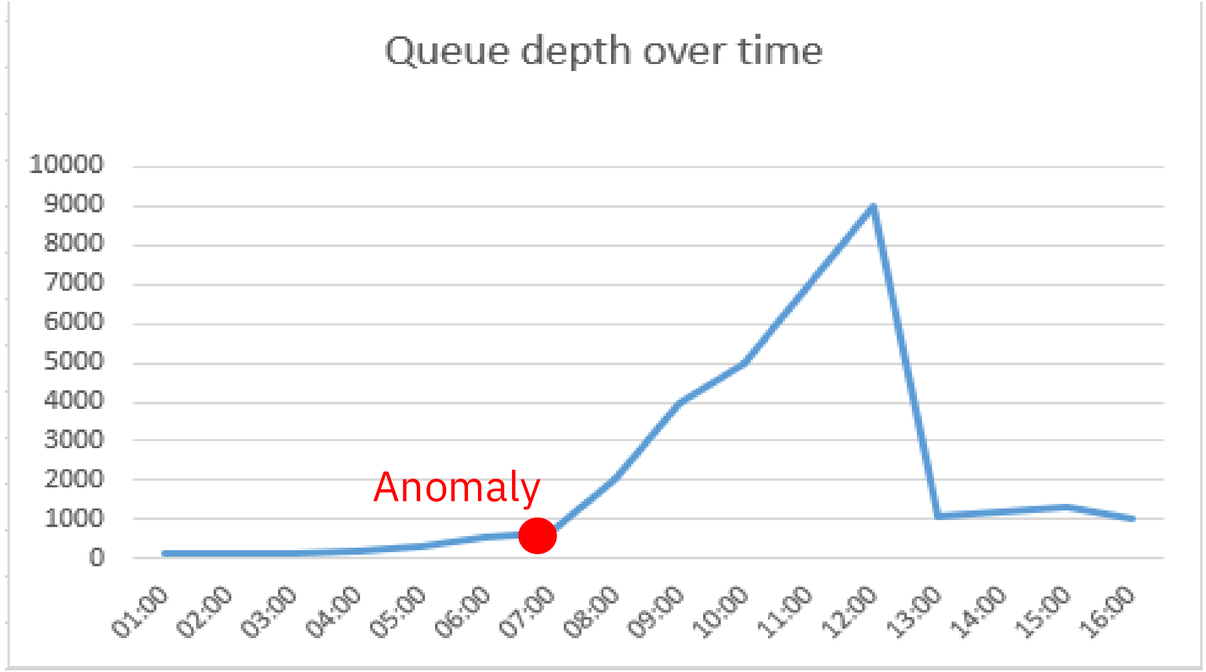
Memory footprint per client:

	9.3.0	9.4.0
31 bit	10.4MB	112KB
64 bit	0	11.6MB

Assuming 1.3GB of free space below the bar, this implies a maximum of 128 clients at IBM MQ 9.3. At 9.3.1 CD (and now 9.4 LTS), we can get to 9999 clients (the maximum) with a suitable MEMLIMIT.

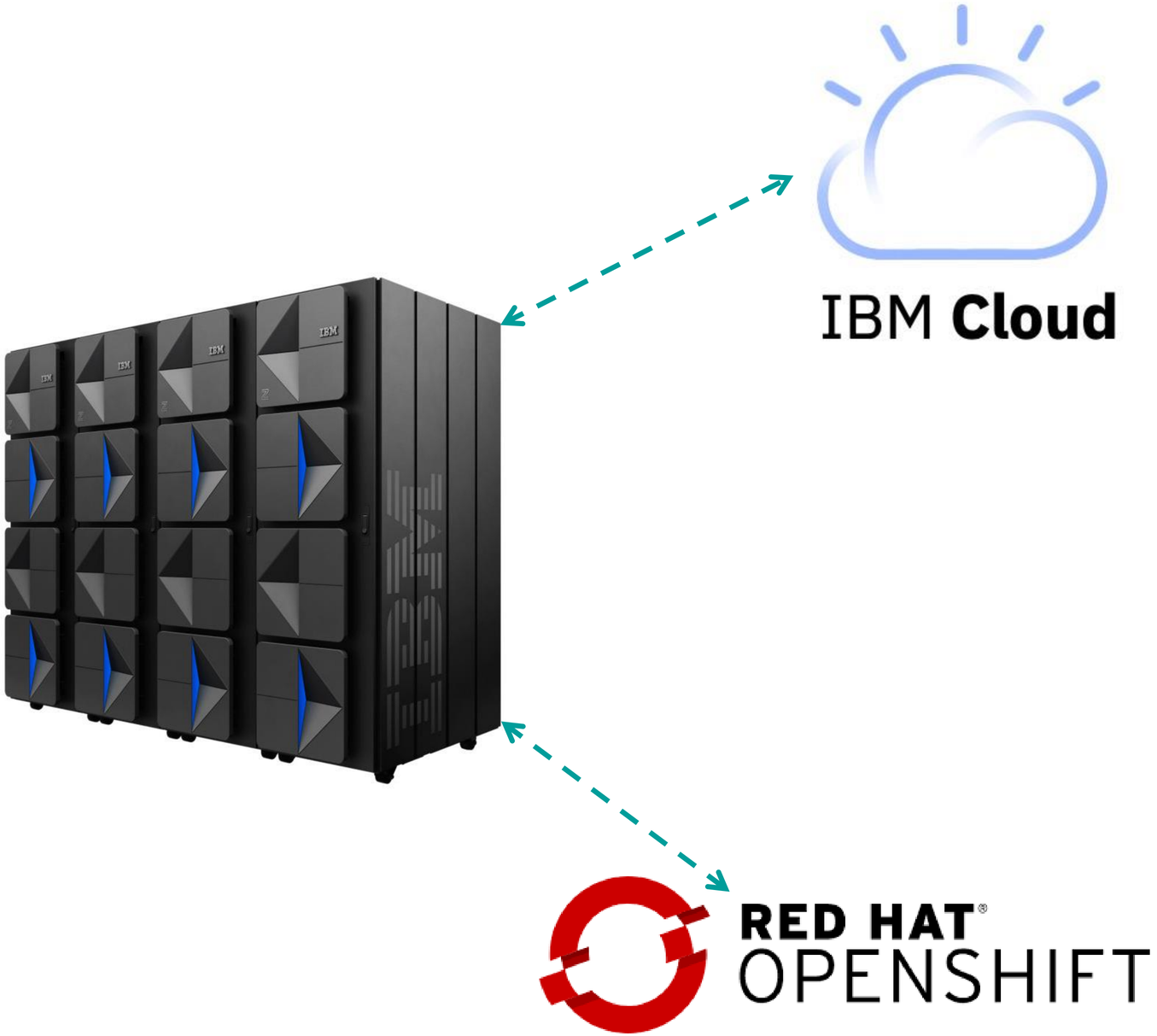
Flexible monitoring on z/OS

- IBM MQ 9.4 includes a wide range of per-queue SMF statistics.
- These enable cheap and simple monitoring of all aspects of your most important queues.
- Use these statistics to power your own ML enhanced anomaly detection
- Alternatively, use the new dashboards in IBM Z Operational Log and Data Analytics to better understand your system.



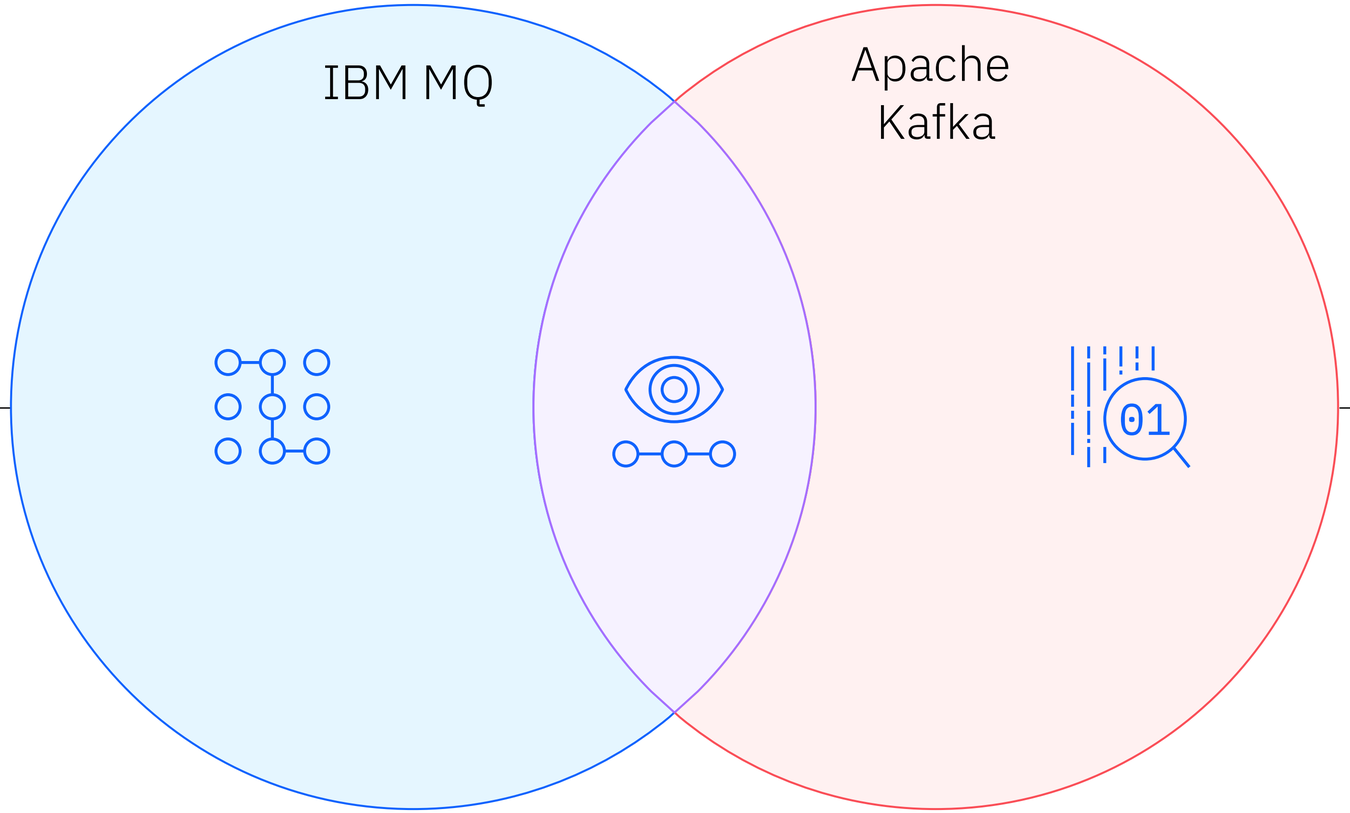
Simplified Cloud Connectivity

- From MQ 9.4.3, z/OS channels now support up to 264 characters in the CONNAME field, replacing the 48 character limit
- This simplifies connecting into cloud or containerized environments, which often have long hostnames.



IBM MQ | Insight into Business Events

IBM MQ and Apache Kafka – Messaging Patterns



Critical exchange of information from one system to another

Messages must get through: no questions asked. The system must be secure and distributed.

Real-time event notification for microservice scalability

Must be frictionless, scalable and lightweight. The system must be simple to exploit, invisible to the application.

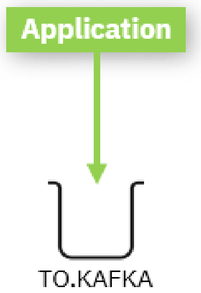
Event streaming for data caching and processing

Must maintain a projection of the data for efficient repeatable processing. The system must scale for large data volumes.

Connecting to Apache Kafka

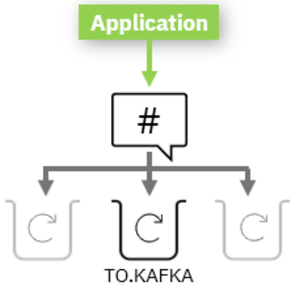
Data from core systems can flow uninterrupted without loss or duplication to Apache Kafka with new assured-delivery sink and source connectors, enabling businesses to drive seamless customer experiences and support strategic business processes.

Direct to queue



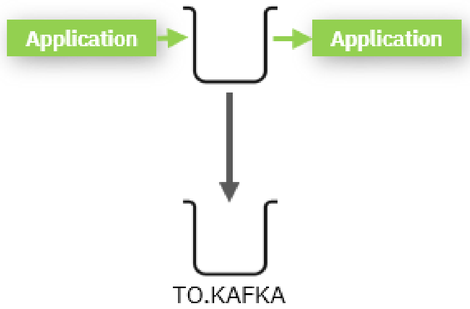
New applications, or applications that are being changed, can put messages directly to a queue used by the source connector.

Subscribe to topic



If messages are already being published to a topic, it's simple to generate another copy for a queue used by the source connector.

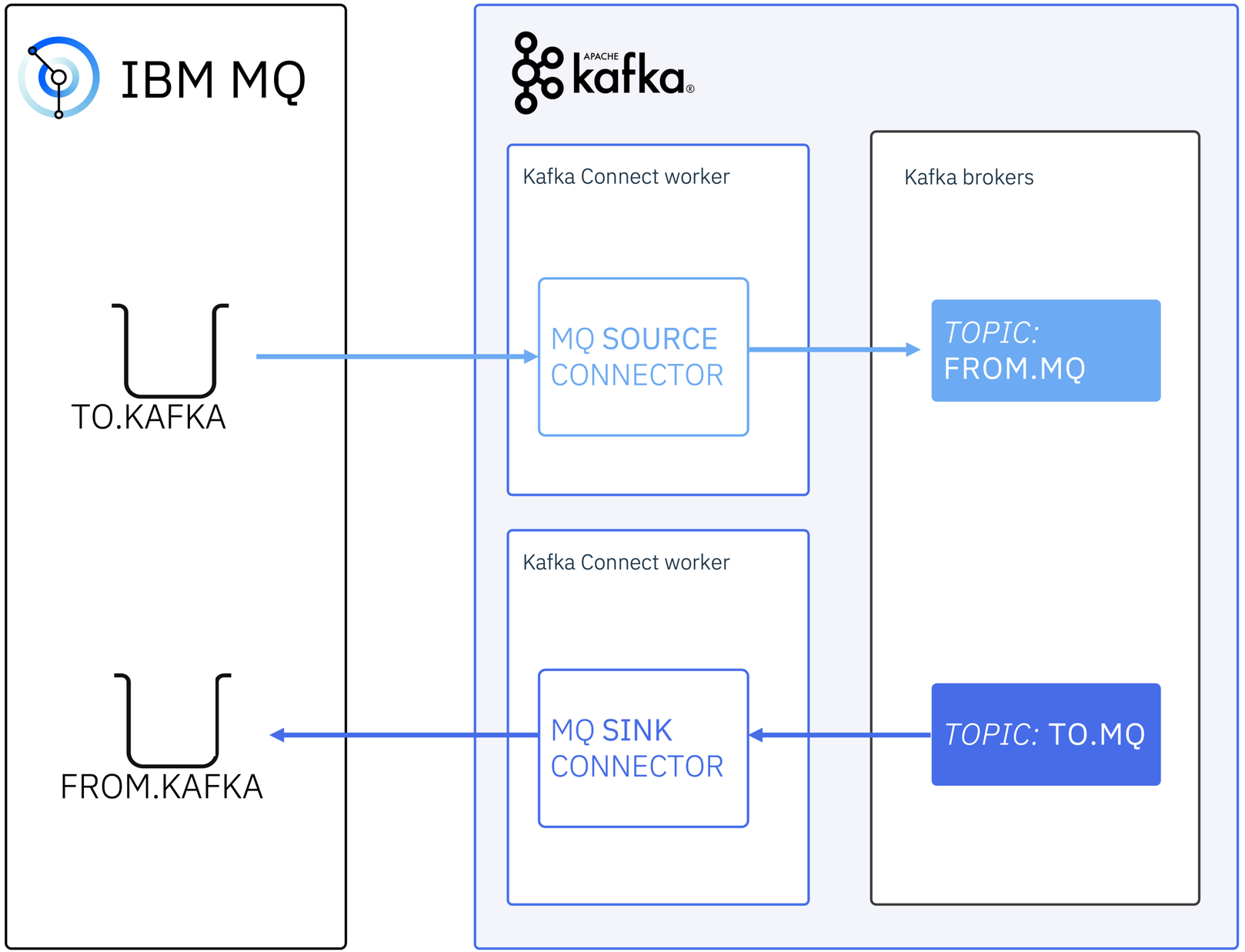
Streaming queue copy



Tap into your mission-critical data as it flows around the enterprise with Streaming Queues, without impacting the existing flow of data. Create a copy of every message onto a second queue that is used by the source connector.

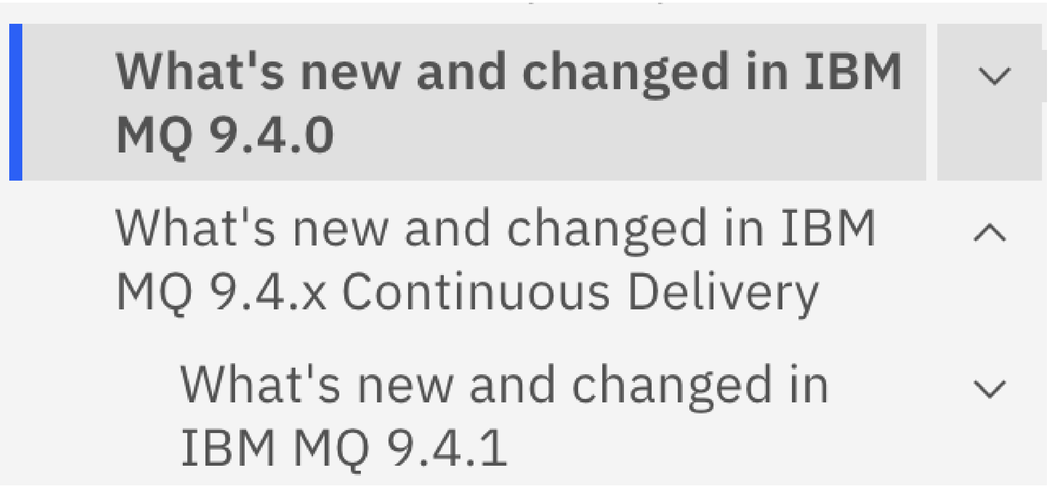
Supported Kafka Connectors

- Sink and source connectors exist for connecting MQ queues to Apache Kafka topics
- The source connector is the most used, enabling MQ to be used as a source for events into Kafka
- IBM provides and supports MQ/Kafka connectors with entitlement to:
  - IBM MQ Advanced
  - Cloud Pak for Integration (CP4I)
- The most recent version of the connector supports “exactly-once” message delivery, preventing message duplication in the event of a Kafka Connect failure.



Questions?

https://www.ibm.com/docs/



Your feedback is important!

SHARE
EDUCATE ▸ NETWORK ▸ INFLUENCE

Submit a session evaluation for each session you attend:

www.share.org/evaluation