Integration Technical Conference 2019

M13: What makes MQ on z/OS so special?

Matt Leming
lemingma@uk.ibm.com

IBM **Cloud**

IBM

IBM MQ for z/OS V9.1.1 announcement

Statement of general direction

IBM intends to deliver the following new capabilities within future Continuous Delivery releases:

• **The ability to apply and remove Advanced Message Security policies transparently between Advanced Message Security (AMS) and non-AMS enabled queue managers**

• **An enhanced z/OS Connect Enterprise Edition service provider for MQ to add support for the Build and API Toolkits, and also automated service deployment**

• zHyperwrite support for MQ log files

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

https://www-01.ibm.com/common/ssi/ShowDoc.wss?docURL=/common/ssi/rep_ca/0/897/ENUS218-510/index.html&lang=en&request_locale=en#sodx

IBM MQ simplifies the challenges of connecting systems

Connects Virtually Everything

Reduces the time and cost of connecting applications

Secure

Protects sensitive messages from all forms of intrusion or attack

Reliable

Once-and-once-only transaction processing - no loss and no duplication

Flexible

Loosely coupled connectivity to avoid application changes

Scalable

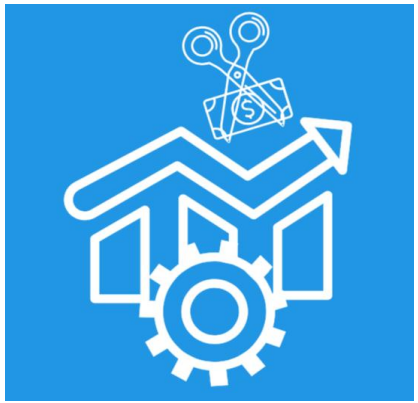
Seamlessly manages changes and spikes in transaction volumes

Robust

Hardened over many years to support the most demanding environments

Why MQ for z/OS?

Connect



- Strong integration with z/OS platform and subsystems
- Hybrid connectivity to public and private clouds
- Bridge to new technologies such as Kafka or Blockchain

Protect

- Connectivity that protects data end-to-end
- Policy-based control for encryption and signing
- Establish a “firewall” with external environments

Optimize

- The gold standard for resilient and highly available connectivity
- Exploit the z/OS platform for maximum performance

A perfect partnership

Best of breed messaging

Underpinning global industry for 25 years

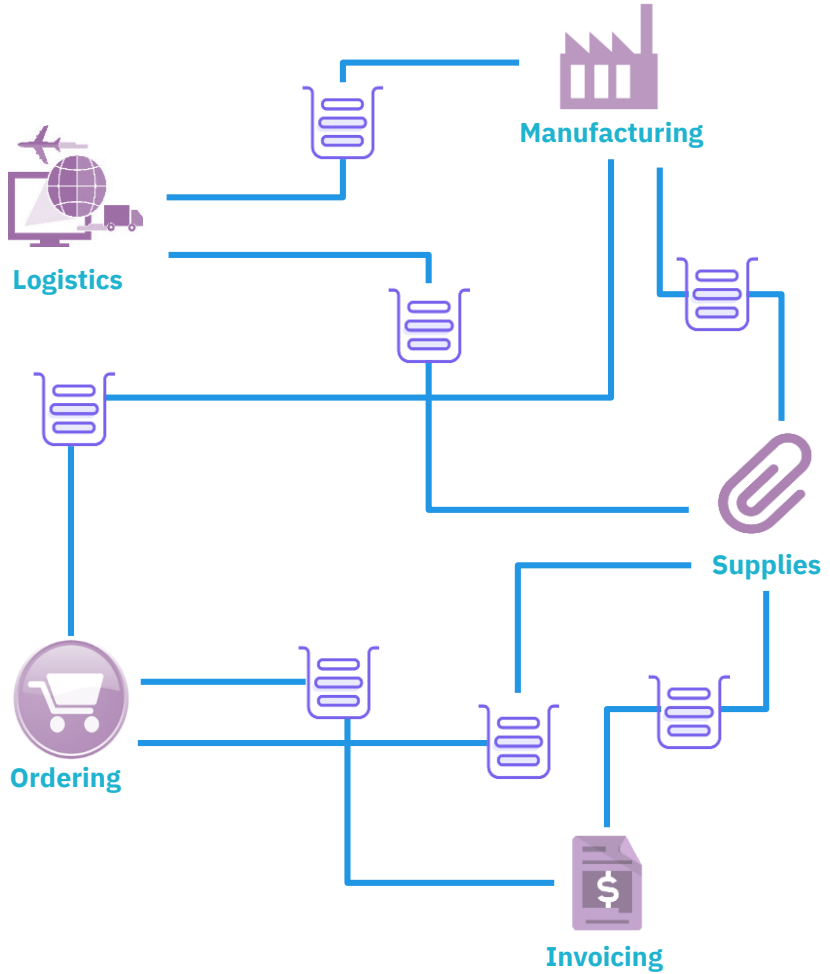
Best of breed hardware

The most trusted hardware platform

IBM Cloud

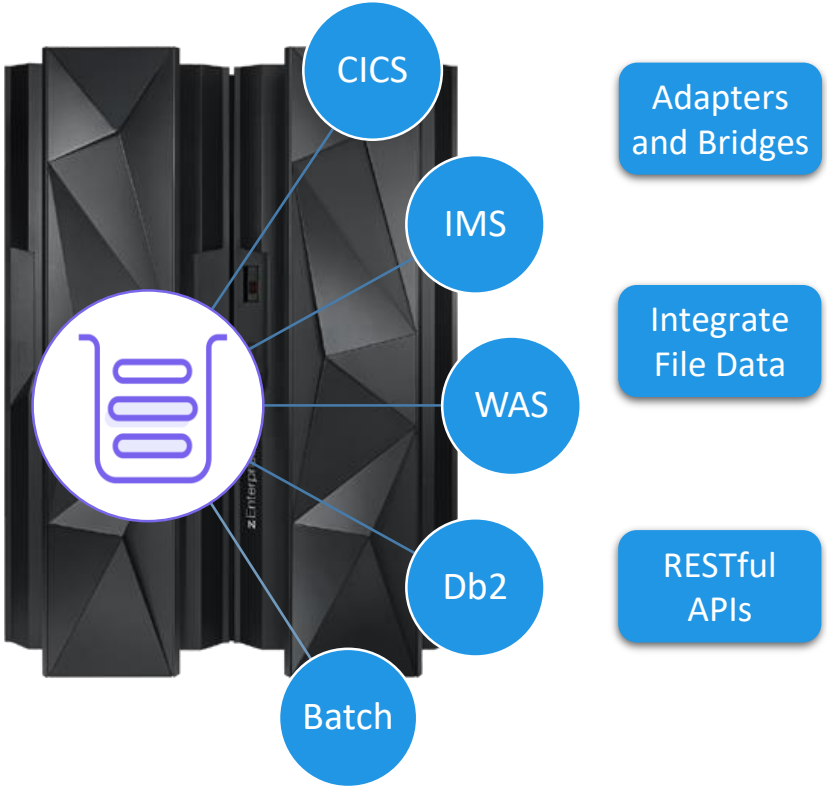
Resilient Secure Strong

Connect



*"The most valuable feature is the ability to **connect our different systems** seamlessly"*

Connect on z/OS



Natural and non-invasive connectivity

Inbound: MQ drives work in a subsystem

Outbound: subsystem sends MQ messages

Unlock data held in files

File conversion to transport as messages

Utilize existing MQ teams, infrastructure, security

Simplify access to MQ resources and operations

Built-in APIs for applications using Messaging

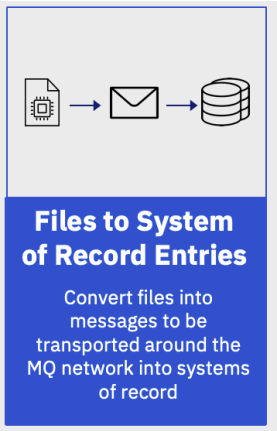
MQ service provider for z/OS Connect Enterprise Edition for generic access

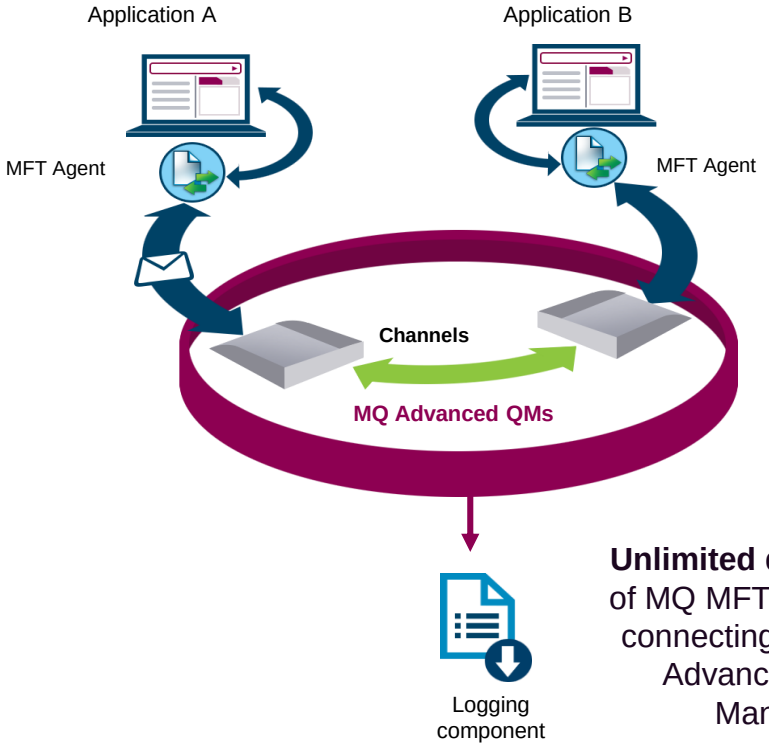
Unlock data held in files

A consistent approach to transporting application data and file data as messages

Utilize existing MQ infrastructure to extend benefit of investment

- Integrate file workloads using existing teams, skills & tools
- Same security, reliability, scalability..





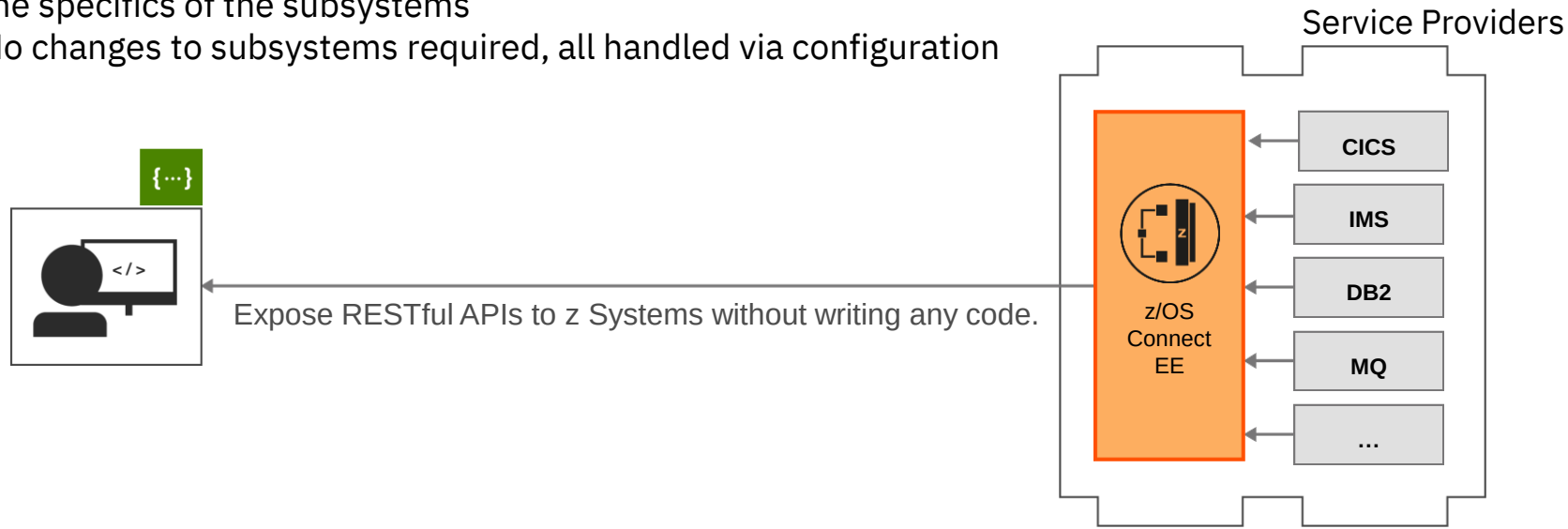
Unlimited deployments of MQ MFT Agents when connecting them to MQ Advanced Queue Managers

Only available in **IBM MQ Advanced for z/OS VUE**

Simplify access to MQ resources and operations

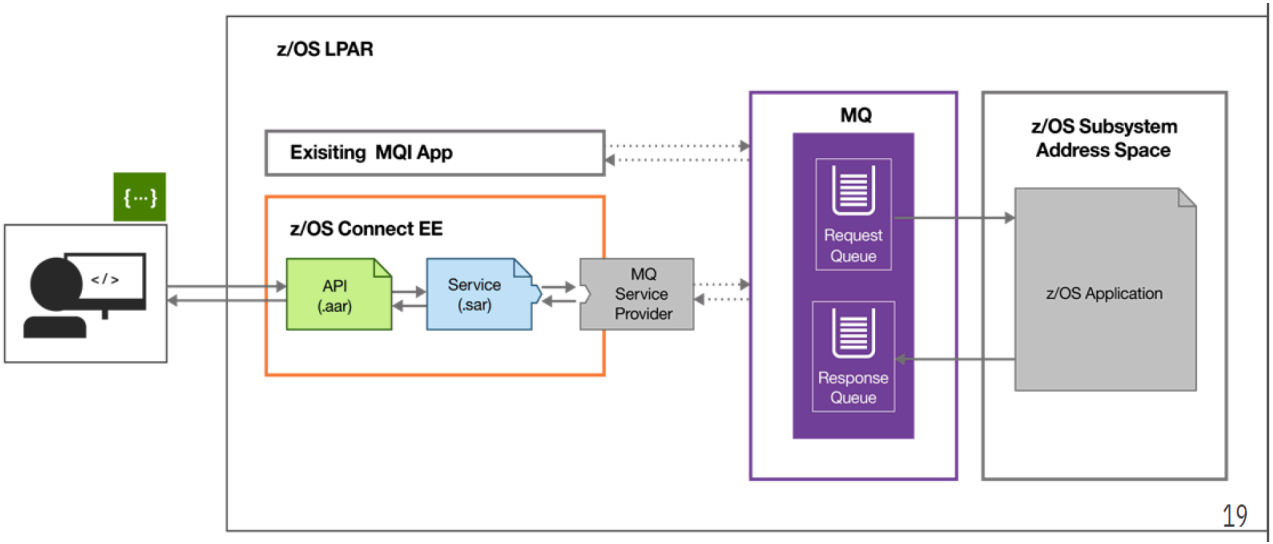
z/OS Connect EE

- IBM z/OS Connect EE provides a single, RESTful entry point to your z systems assets and data
- Enables reuse of existing assets, exposing them to environments where it is natural to use REST
- Those new consumers do not need to understand or be aware of the specifics of the subsystems
- No changes to subsystems required, all handled via configuration



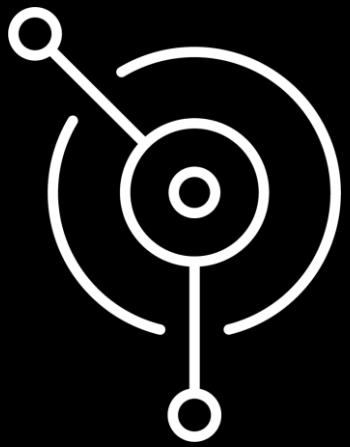
MQ service provider for IBM z/OS Connect EE

- Free of charge z/OS Connect service provider allows existing services fronted by MQ to be accessed via a RESTful front end
- Users need have no knowledge of MQ
- Advanced users can specify some MQ attributes using HTTP headers
- **Coming soon:** MQ SARs can be built as part of an automated build pipeline (using zCEE build toolkit)



- Expose **bespoke** REST APIs to new consumers, who don't understand COBOL copybooks or PL/I
- Backend is hidden and invoked using JSON / HTTP

Extend reach of mainframe to public and private clouds



On-premise, software and the MQ Appliance

Run it yourself in any cloud, public or private

Let IBM host it for you with its managed SaaS MQ service in public clouds

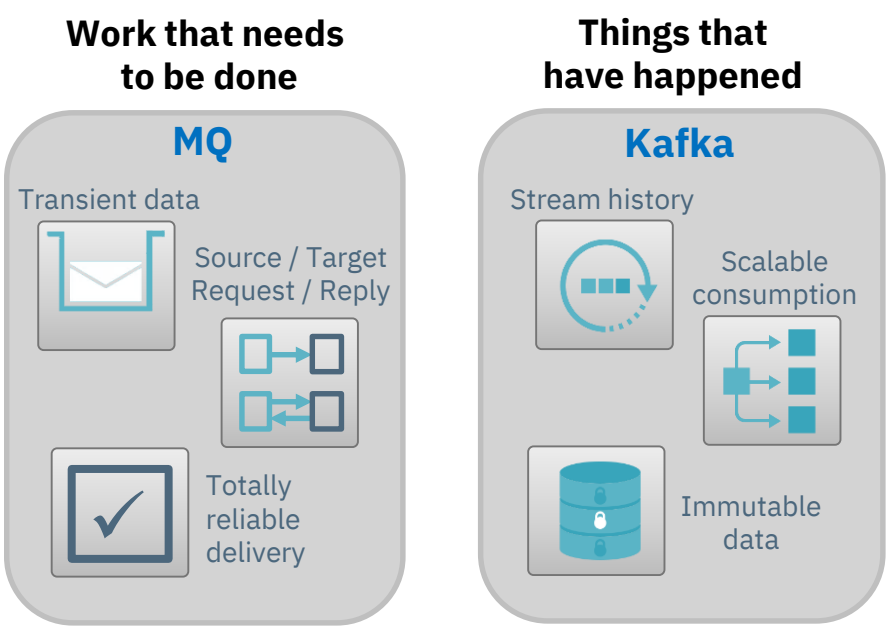
IBM Z
Linux
AIX
Windows
Solaris
zLinux
IBMi
Appliance
...

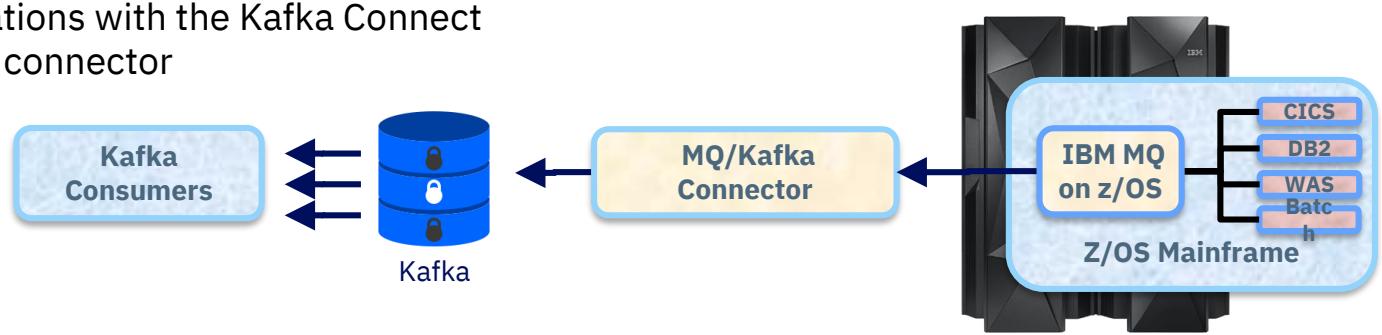
AWS
Azure
IBM Cloud
IBM Cloud Private
Private cloud
MQ on Cloud

Latest MQ features

Connect and leverage
Kafka connectivity

- MQ often handles messages that represent changes to the state of the applications, systems and services it is connecting together
- **Gain valuable insights** by sending notifications of these operations to Event streaming platforms such as Apache Kafka
- **Send messages simply** from MQ to Kafka applications with the Kafka Connect source connector





https://github.com/ibm-messaging/kafka-connect-mq-source

Protect

Average cost of data breach:
$3.94 million

Average time to identify breach: 191 days
Average time to contain breach: 66 days

Average cost per incident:
over $1 million

Average cost per minute for partial outages: $5,600.00
Average cost per hour for partial outages: $300,000.00

GDPR cost per incident**:**
4% of annual global turnover, or €20 Million
– whichever is greater

Source: Cost of Data Breach Study: Impact of Business Continuity Management Ponemon Institute, June 2017

Protect: a paradigm shift

From selective encryption to pervasive encryption

The practice of pervasive encryption can:

- **Reduce risk** associated with undiscovered or misclassified sensitive data
- Make it **more difficult for attackers** to identify sensitive data
- Help **protect** *all* of an organization's digital assets
- Significantly **reduce the cost** of compliance

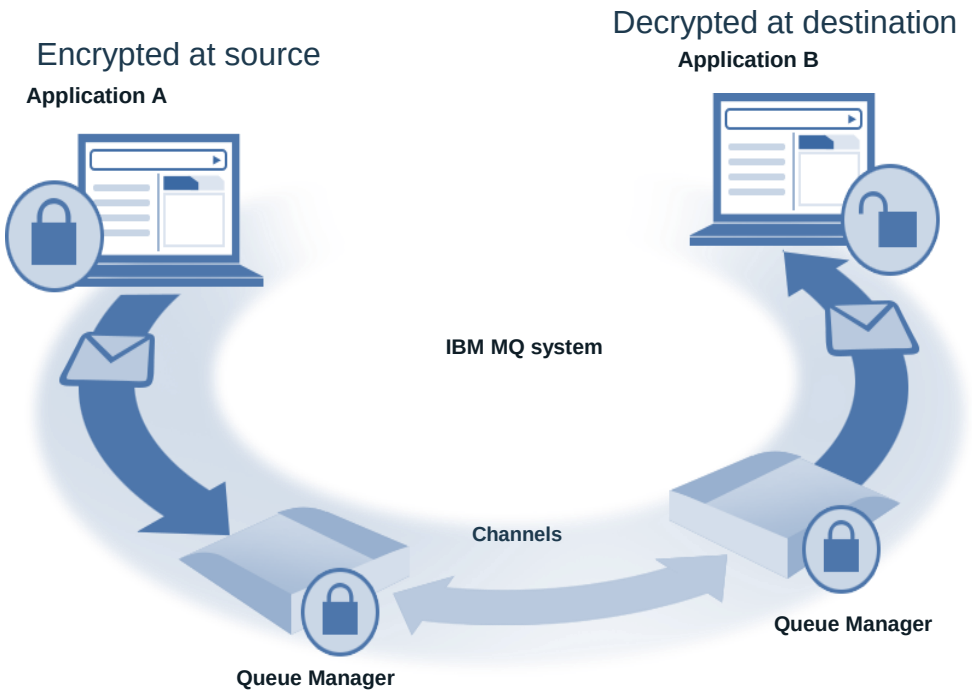
Pervasive encryption is the new **standard**

Encrypting only the data required to achieve compliance should be viewed as a ***minimum threshold***, not a best practice

Protect end-to-end

Secure data at rest, in-flight and in-memory to guarantee privacy of message contents

- Apply **end-to-end encryption** to existing messaging infrastructure easily and with no application changes
- **Authenticate** and **protect** messages across the enterprise making **audit simple**
- **Reduce time and skills** needed to comply with aspects of common security standards
- Confidentiality option for encryption has **minimal performance impact**
- **Coming soon:** AMS policies can be applied and removed as messages move between queue managers, allowing easier interactions with business partners who don't have AMS

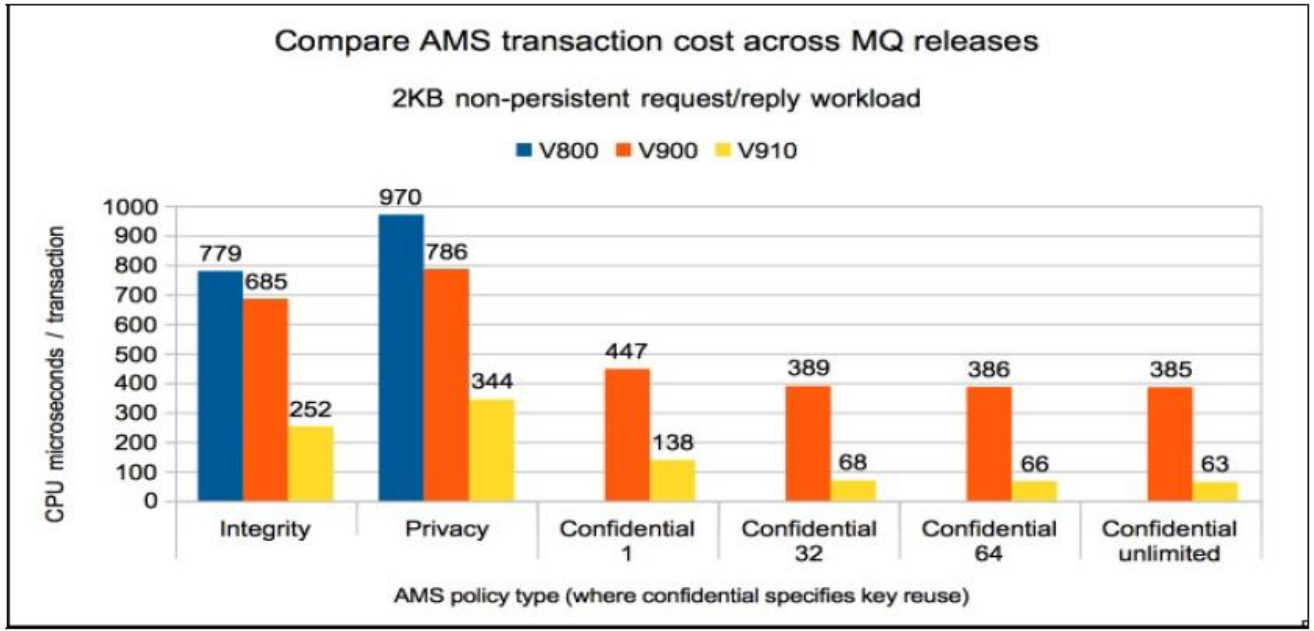


Protecting your **business**, your **reputation**, and your **customers**

Only available in **IBM MQ Advanced for z/OS VUE**

Reducing the costs of protection

AMS policies provide flexibility to configure encryption and digital signing as needed



A cost comparison between v9.1.0 and v9 shows:

- **Integrity: 45%** of the equivalent v9 measurement
- **Privacy: 40%** of the equivalent v9 measurement
- **Confidentiality: 15-25%** of the equivalent v9 measurements

What's Confidentiality?

Encryption only with configurable amount of symmetric key reuse between same producer and consumer

Only available in **IBM MQ Advanced for z/OS VUE**

Optimize

ITIC 2017-18 Server OS Reliability Report

"IBM's Z Systems Enterprise is in **a class of its own**"

"IBM mainframe continues to exhibit **peerless reliability besting all competitors**"

Optimize high availability

Queue sharing groups are the **gold standard** for high availability

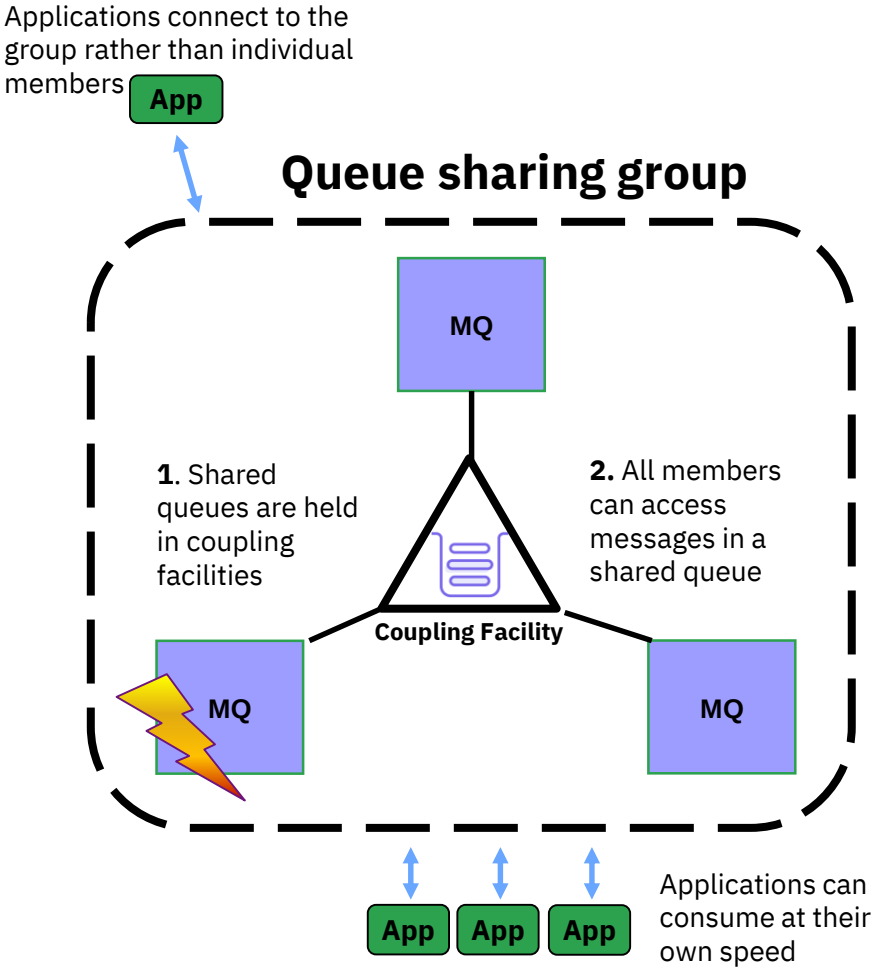
Main advantages of queue sharing groups:

Scalable: a single profile and security definition

Highly available: multiple queues accessing the same messages

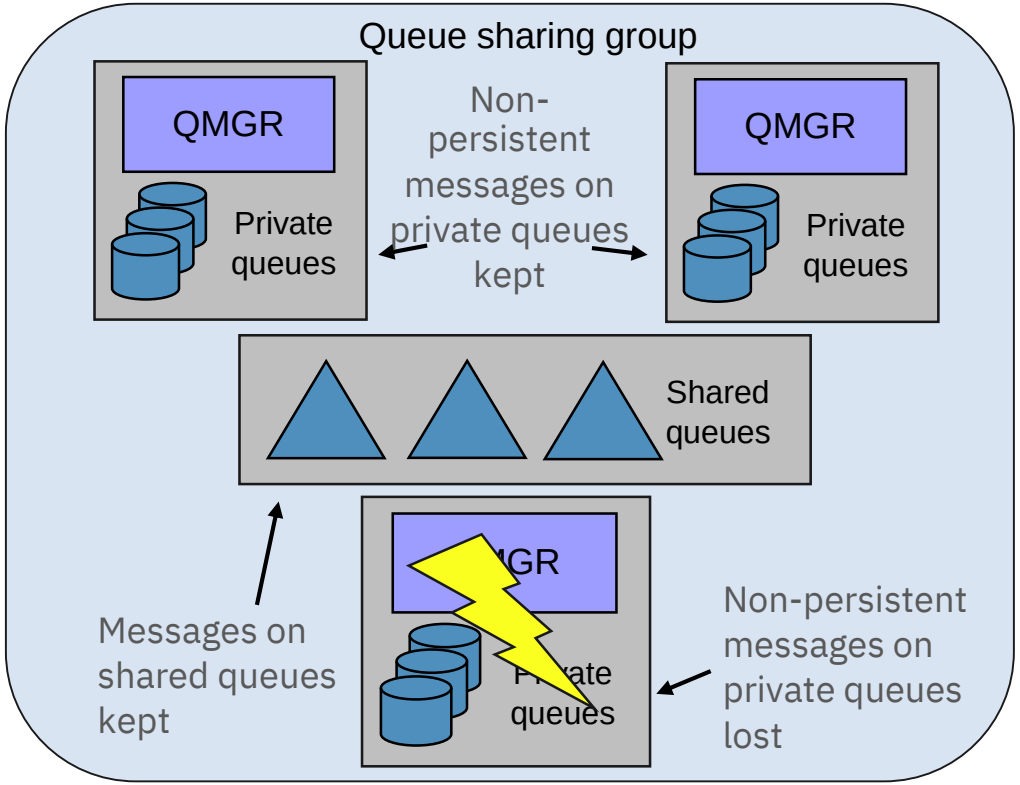
Workload balancing: distributes workload between queues in group

The original uniform cluster!

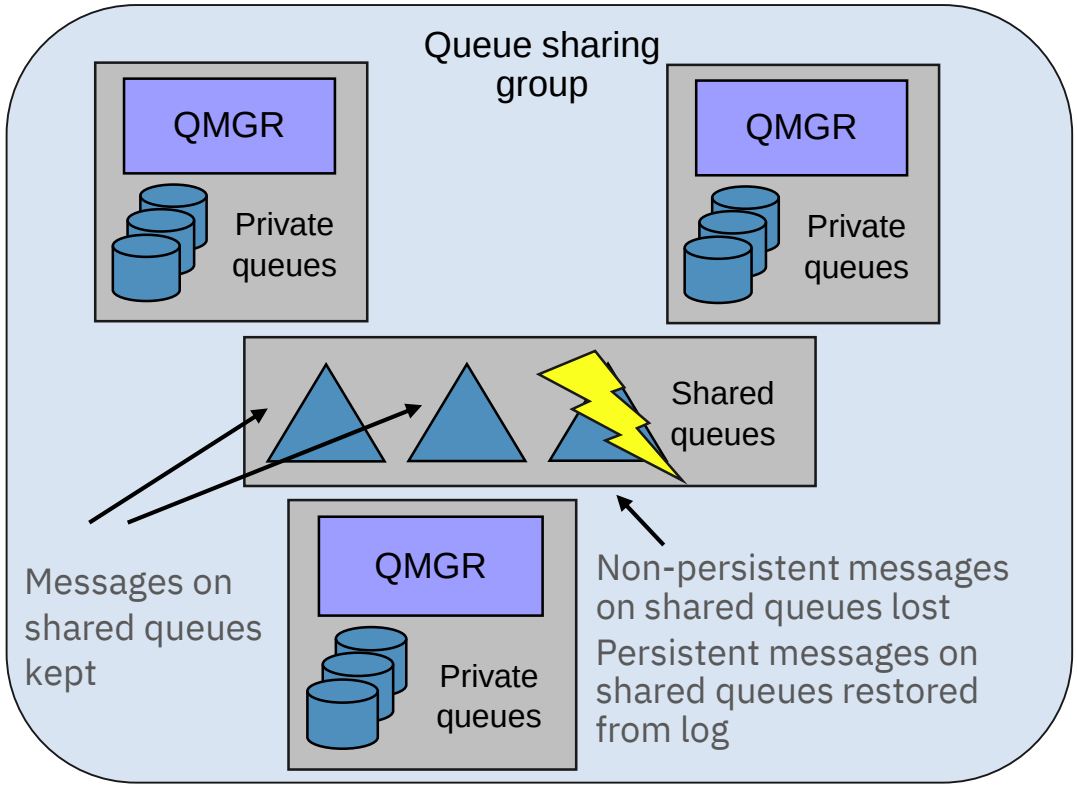


"High availability options are outstanding." ***Financial services company***

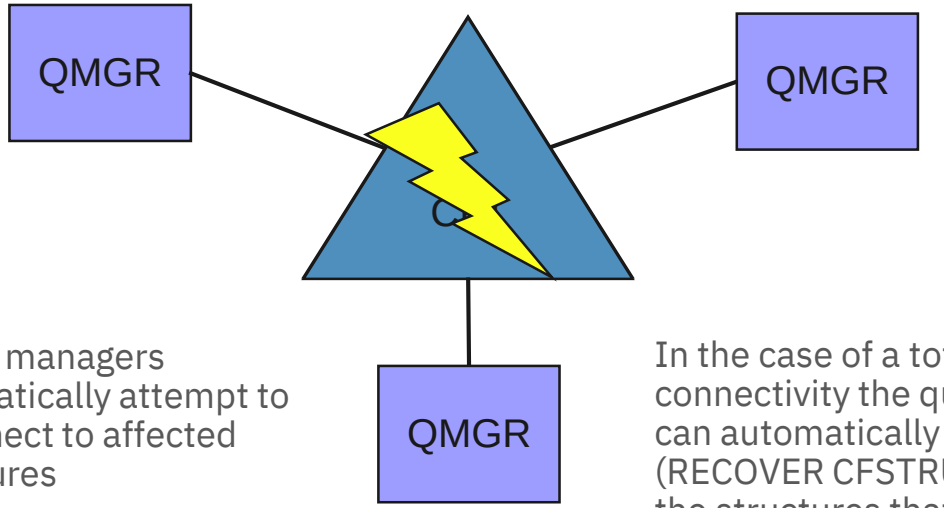
Messages remain available if individual queue managers or LPARs fail



Resilient to structure failure



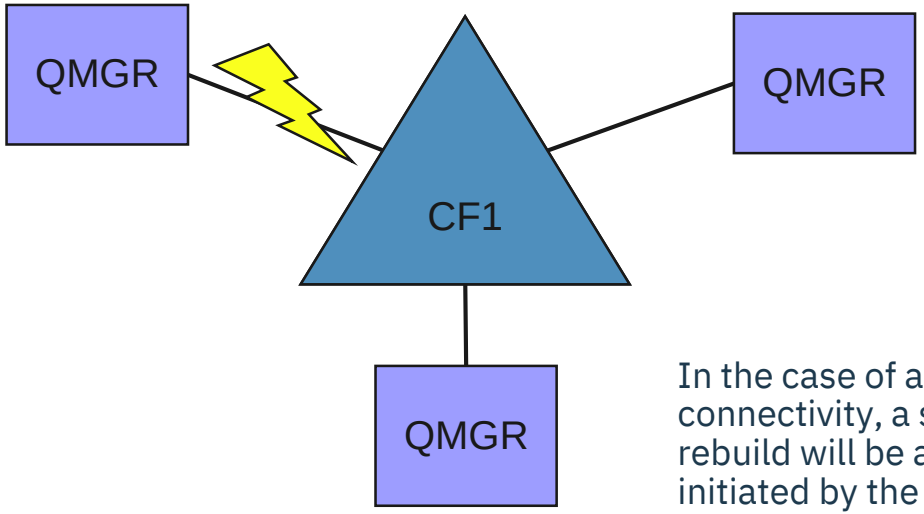
Resilient to coupling facility connectivity failure: total



Queue managers automatically attempt to reconnect to affected structures

In the case of a total loss of connectivity the queue managers can automatically recover (RECOVER CFSTRUCT) the structures that were on the failed coupling facility into an alternative (if available)

Resilient to coupling facility connectivity failure: partial



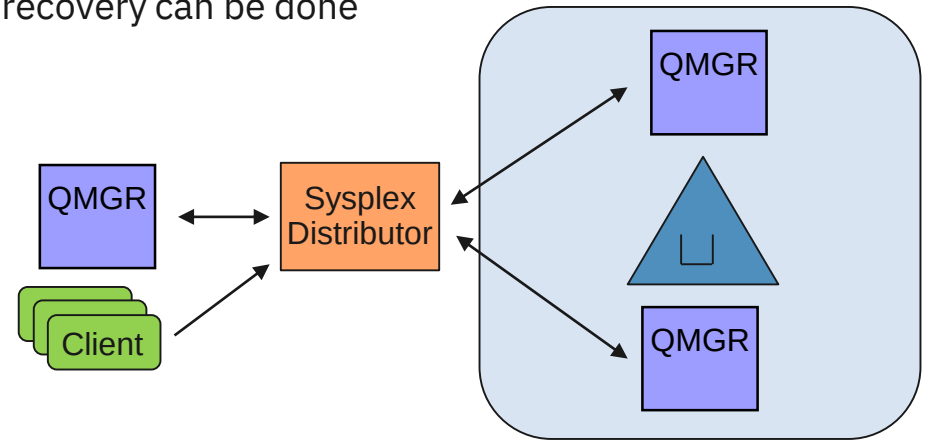
In the case of a partial loss of connectivity, a system managed rebuild will be automatically initiated by the queue managers to rebuild the structures into a more available coupling facility. This will mean that both persistent and non-persistent messages will be retained

Peer recovery for inflight work

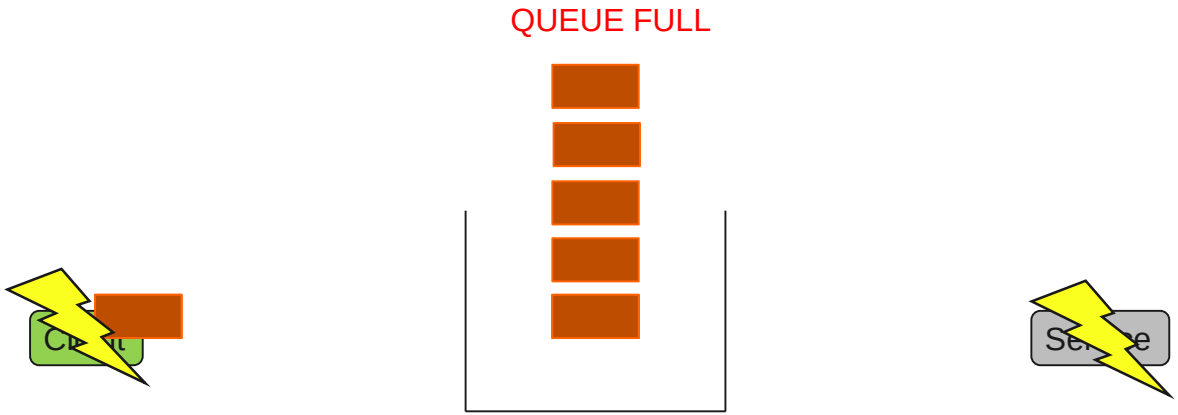
- Individual queue managers perform operations (PUT/GET) on messages on shared queues
- If these operations are performed in a unit of work, then state information about the operations is also stored in the structure
- If the queue manager fails, other queue managers in the queue sharing group will detect the failure and use the state information to resolve in-flight units of work
- This process is known as peer recovery
- Benefits:
  - **Units of work resolved even if failing queue manager can’t be restarted**
  - **Message availability not affected by queue manager failure**
  - **XA transactions can be resolved by any queue manager in queue sharing group**

Shared channels

- Channels provide network connectivity to MQ
  - Allow applications (clients) to send and receive messages
  - Allow messages to be sent between queue managers
- Normally channels are owned by a particular queue manager
- In a queue sharing group channels can be shared
- Channel can run in any queue manager in queue sharing group
  - State information in held in structures, and Db2, so recovery can be done by any queue manager
- Benefits:
  - **Connectivity provided at queue sharing group level, not at individual queue manager**
  - **Failure of individual queue manager requires application reconnect, not application outage**
  - Load balanced over queue sharing group



Resilient to application failure

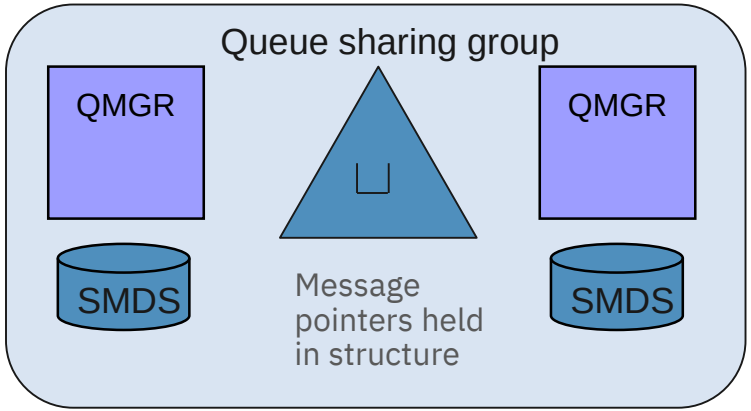


Resilient to application failure

- Messages on private queues are stored in pageset (DASD): **64GB** limit

- Messages on shared queues stored in coupling facility structure: **1TB** max
  - Stored in real memory, which is a relatively scarce resource, so customers never go anywhere near the limit – low 10s of GB is a common upper limit

- SMDS – Shared Message Datasets
  - Message data held on DASD, pointer to message held in structure
  - Each SMDS can be up to **16TB**
  - Used for:
    - messages that are too large for storing in structure (> 63 KB)
    - messages affected by offload rules



Summary of high availability characteristics of shared queues

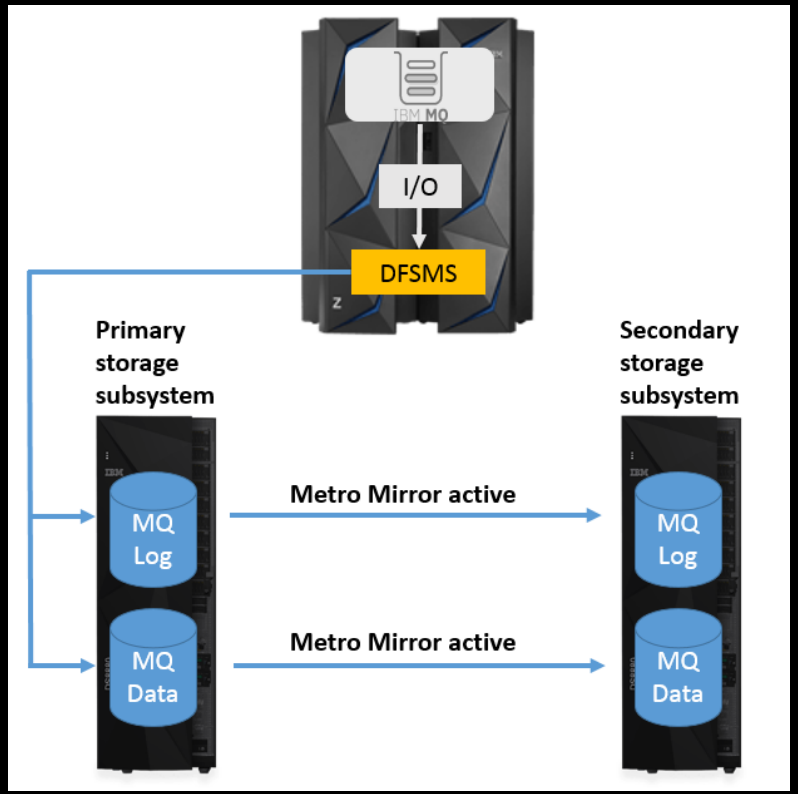
- Messages remain available if individual queue managers or LPARs fail
- Applications and queue managers can continue to connect into queue sharing group if individual queue managers or LPARs fail (shared channels)
- Resilient to structure failure
- Resilient to coupling facility connectivity failure
- Peer recovery for inflight work
- Resilient to application failure

Optimize by exploiting zHyperWrite

Many customers use Metro Mirror (Synchronous PPRC) with MQ to mirror their datasets

This protects against storage subsystem failure, and can be part of an HA/DR strategy

Mirroring has a performance impact, even at zero distance because the write from MQ doesn't complete until the writes to both primary and secondary complete, and these happen in series



Optimize by exploiting zHyperWrite

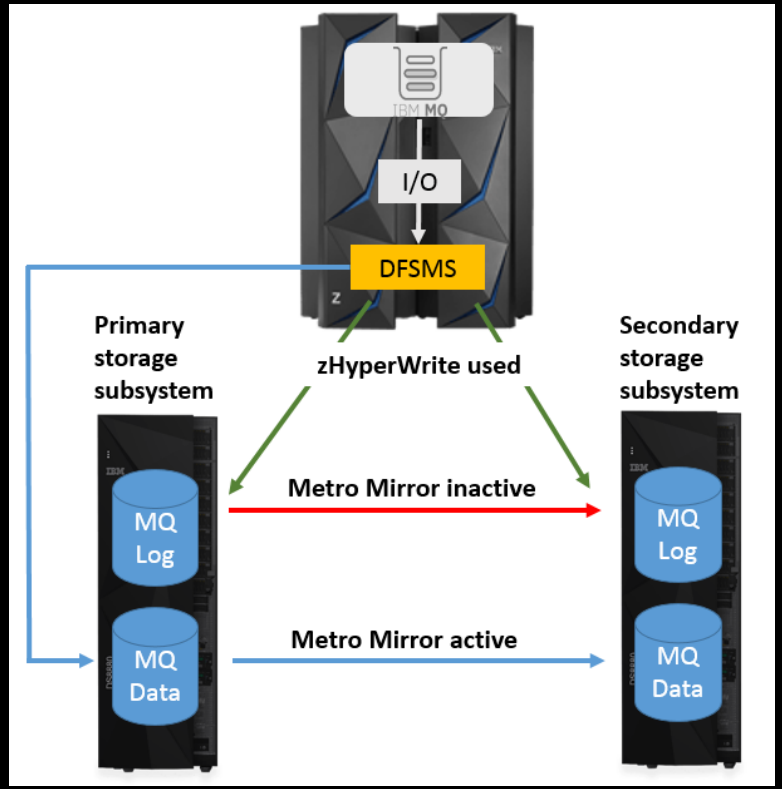
zHyperWrite was introduced to minimize the performance impact of Metro Mirror

Collaboration between DS8K and DFSMS, originally done for Db2

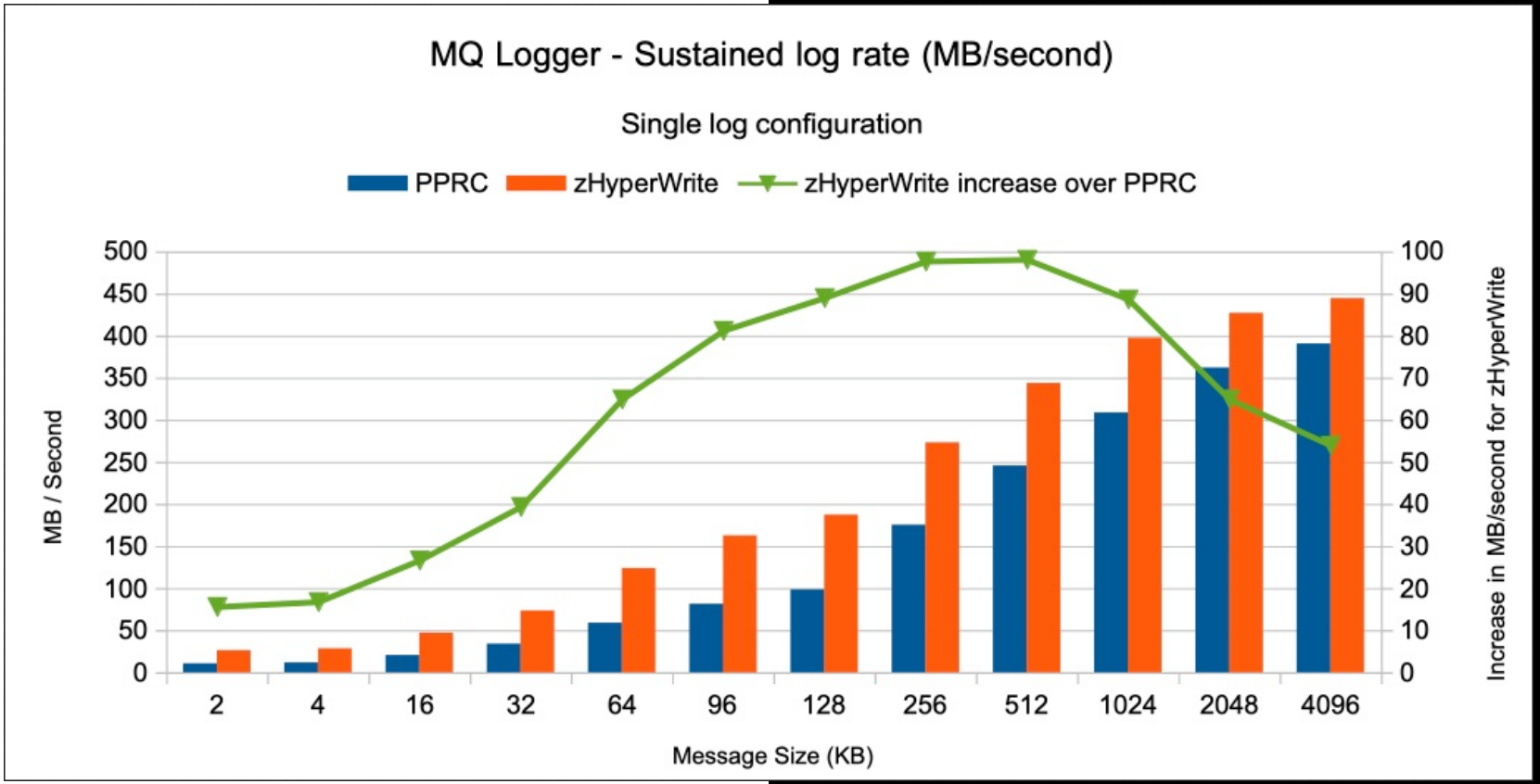
With zHyperWrite the writes to primary and secondary are issued in parallel at the DFSMS level, meaning the write can complete earlier

If a zHyperWrite write fails then it falls-back transparently to Metro Mirror

In 9.1.2 MQ has added support for zHyperWrite for active log datasets



Optimize by exploiting zHyperWrite



Get the support, strength, and longevity that you need

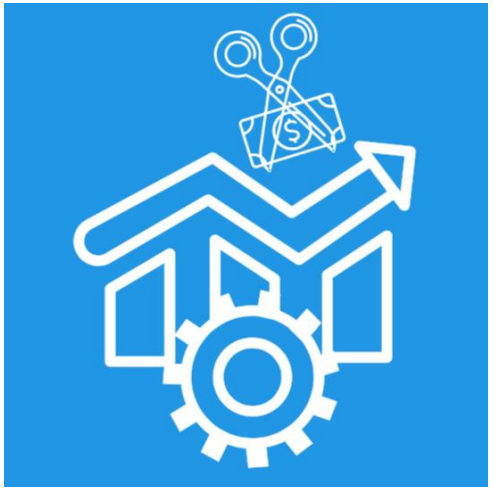
IBM offers **world-class support** 24/7
IBM MQ has a **thriving online community** of responsive experts

IBM MQ is known for its **robustness**
Stability and **reliability** are cited time and again as its strengths

IBM MQ is 25 years young, yet **forever evolving** to support changes
in use cases, platforms and development practices

“I know a lot of people are talking about going out and buying open-source things or trying open-source things. I say, “Stick to products that have been around, that have been proven, and that you have the support of a vendor behind you who's willing to look at these things and develop around you.”

Connect

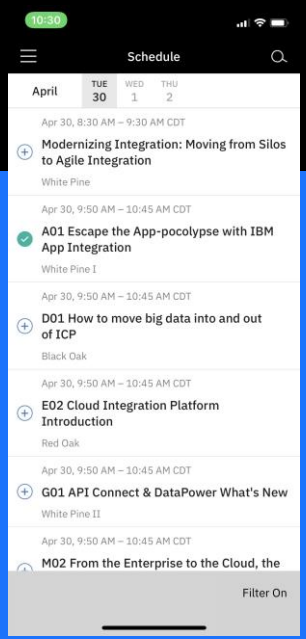


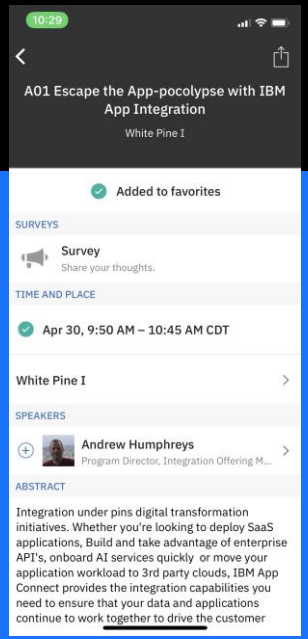
Protect

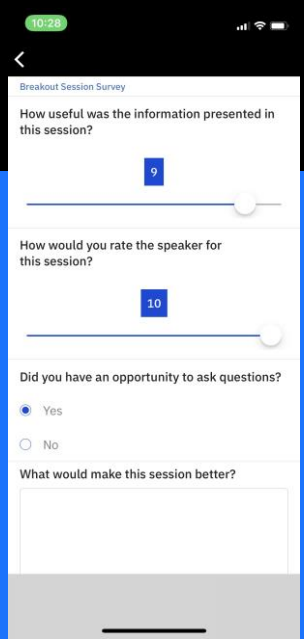
Optimize

With IBM MQ for z/OS

https://www.ibm.com/products/mq/zos







Don’t forget to fill out the survey!

Select your session, select survey, rate the session and submit!

Thank You

Matt Leming
lemingma@uk.ibm.com