What's new in MQ 9.0.4?

Plus a few other interesting things!

Carl Farkas
IBM Europe zHybrid Cloud consultant
farkas@fr.ibm.com
(with thanks to Matt Leming and David Ware)

Agenda

- MQ evolution
- Managing MQ
- High availability
- Supporting your applications
- Connectivity
- Securing MQ
- Platform specific goodies

MQ Runs Exactly How and Where You Need It

Cloud

IBM MQ

Message Hub

IBM Bluemix

Azure

IBM MQ

Microsoft Azure

Google

IBM MQ

...

AWS

IBM MQ

amazon web services

docker

openstack™

kubernetes

vmware

...

On-premises

IBM MQ

z/OS

Linux

Windows

HPUX

Solaris

AIX

...

Distributed platforms

IBM MQ

Private cloud

IBM MQ

IBM MQ Appliance

This presentation focuses on CD…

On-premises

Linux

Windows

AIX

…

Distributed platforms

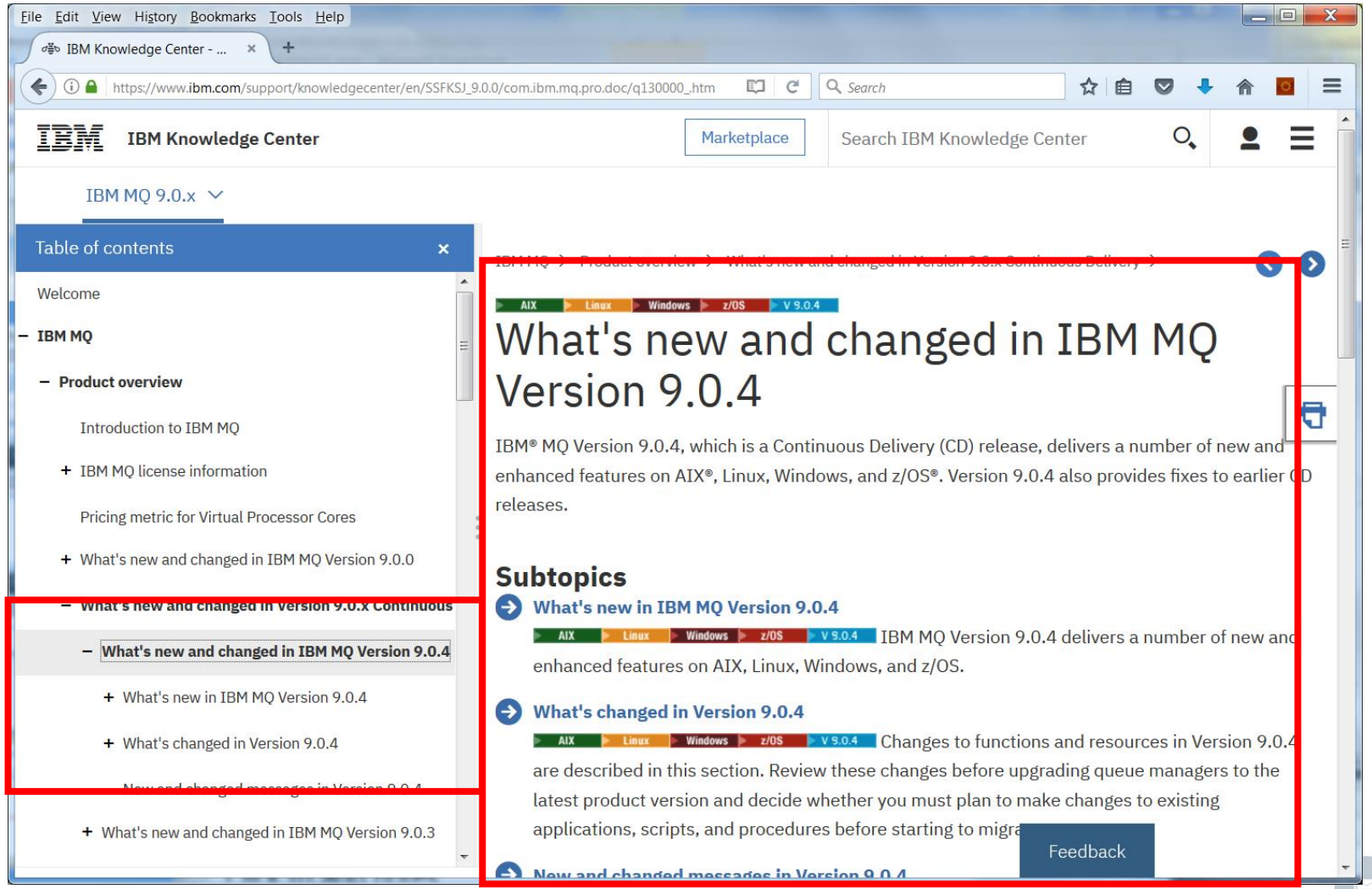
IBM MQ Appliance

IBM MQ 9.0.x CD releases

V9.0.1 CD	**November 2016**
V9.0.2 CD	**March 2017**
V9.0.3 CD	**July 2017**
V9.0.4 CD	**November 2017 – Announced 23rd October**

MQ release-to-release changes

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



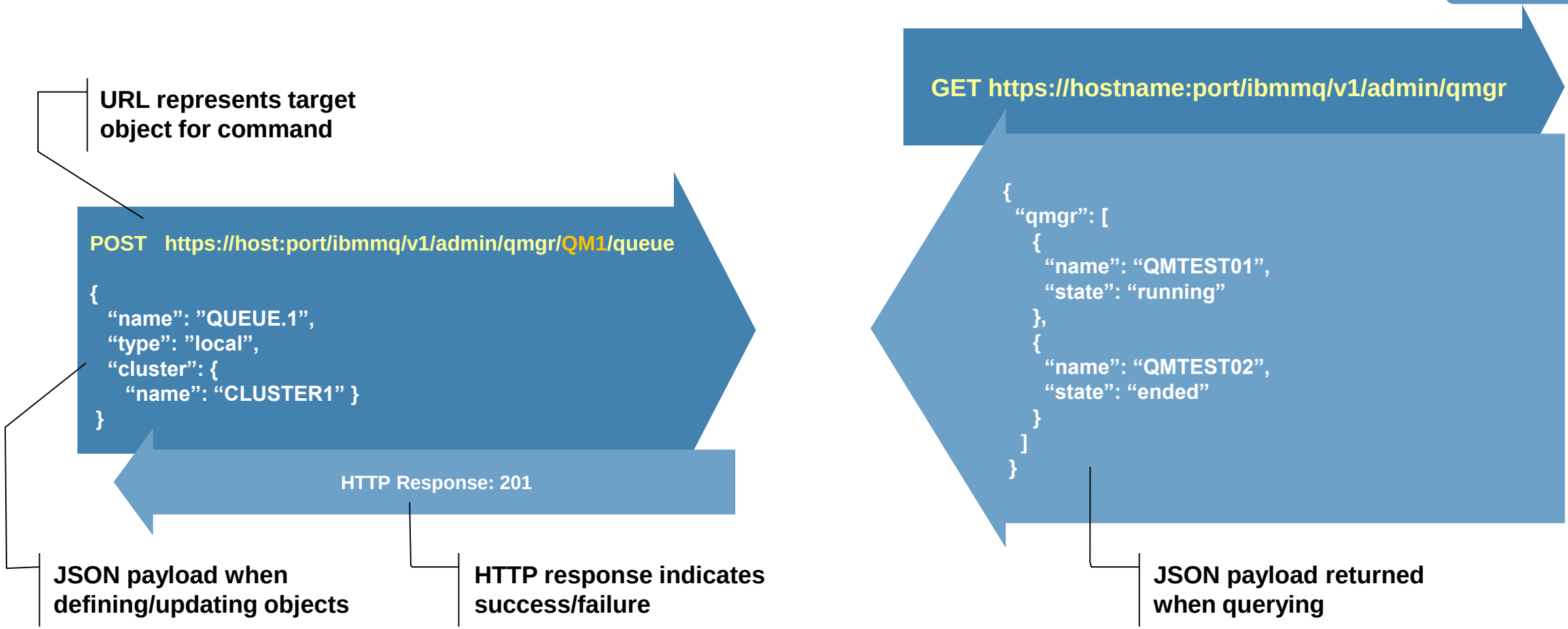
Managing MQ

Using REST

Based off underlying MQ capabilities such as PCF and control commands, but adjusted to adhere to RESTful practices

dspmq, dspmqver, queues, qmgr status, subscriptions, channels

V9.0.1 CD
V9.0.2 CD
V9.0.3 CD
V9.0.4 CD
…



Display channel REST API

Support for displaying TCP/IP based qmgr-qmgr channels

```
C:\>curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/qm904/channel/SYSTEM.DEF.SENDER?attributes=connectionManagement -u mqadmin:mqadmin
{"channel": [{
  "connectionManagement": {
    "disconnectInterval": 6000,
    "heartbeatInterval": 300,
    "keepAliveInterval": -1,
    "localAddress": [],
    "longRetry": {
      "count": 999999999,
      "interval": 1200
    },
    "shortRetry": {
      "count": 10,
      "interval": 60
    }
  },
  "name": "SYSTEM.DEF.SENDER",
  "sender": {
    "connection": [],
    "transmissionQueueName": ""
  },
  "type": "sender"
}]}
```

- This implementation essentially combines DISPLAY CHANNEL and DISPLAY CHSTATUS
- See KC q130490 for details

Display subscriptions REST API

Support for displaying subscriptions

```
C:\>curl -k https://localhost:9443/ibmmq/rest/v1/admin/qmgr/qm904/subscription?attributes=destination -u mqadmin:mqadmin
{"subscription": [
  {
    "destination": {
      "correlationId": "000000000000000000000000000000000000000000000000",
      "isManaged": false,
      "name": "",
      "qmgrName": ""
    },
    "id": "414D5120716D393034202020202020207FEDF95914B3EB06",
    "name": "SYSTEM.DEFAULT.SUB",
    "resolvedTopicString": ""
  },
  {
    "destination": {
      "correlationId": "414D515901010000000000000000000000000000000000",
      "isManaged": false,
      "name": "SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS",
      "qmgrName": "qm904"
    },
    "id": "414D5120716D393034202020202020288EDF95923D22807",
    "name": "qm904 SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS 414D515901010000000000000000000000000000000000 SYSTEM.BROKER.ADMIN.STREAM MQ/qm904
    "resolvedTopicString": "SYSTEM.BROKER.ADMIN.STREAM/MQ/qm904                                        /StreamSupport"
  }
]}
```

- See KC q129410 for details

MQSC for REST

Tailored RESTful support for individual MQ objects and actions are in the works…

However, to speed up full MQ admin support over REST we will be adding the ability to submit arbitrary MQSC commands over REST

- ✓ Gives complete MQSC coverage quickly
- ✓ Simple to convert existing scripts
- ✗ Does not benefit from improved usability
- ✗ Note that you cannot use this MQSC resource today with the REST Admin API gateway (see below)

See q129385 for details

```
POST https://host:port/ibmmq/v1/admin/action/qmgr/QMGR1/mqsc

{
   "type": "runCommand",
   "parameters": {
      "command": "STOP CHANNEL(CHANNEL.TEST)"
   }
}
```

```
{
  "commandResponse": [{
      "completionCode": 0,
      "reasonCode": 0,
      "text": ["AMQ8019: Stop IBM MQ channel accepted."]
   }],
  "overallCompletionCode" : 0,
  "overallReasonCode" : 0
}
```

Stopping a channel

Enabling your whole estate for REST administration

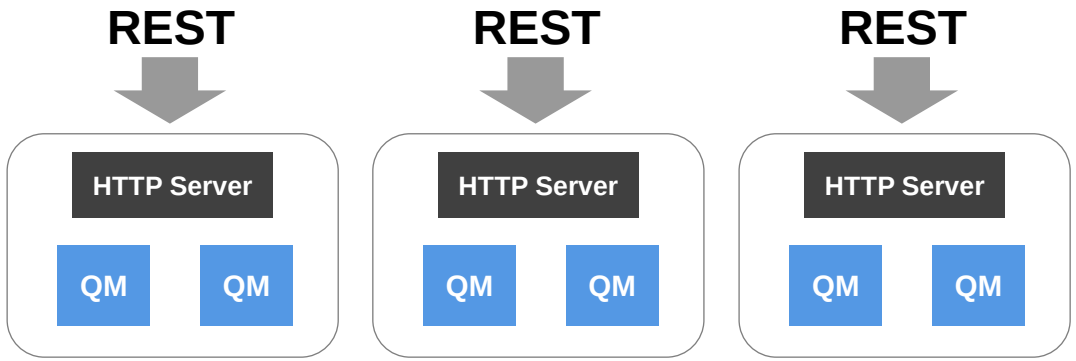
Requested by you! RFE 104343 ✅

V9.0.4 CD

...

Option 1

Administer each MQ installation separately, they must all be on a MQ 9.0.x CD release

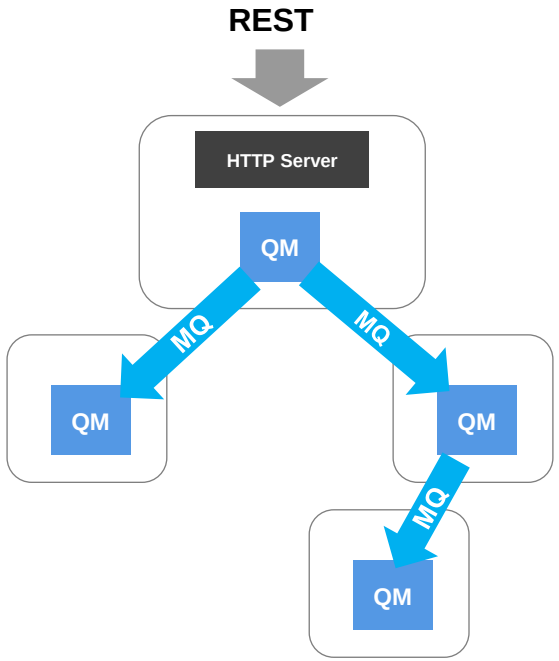


Option 2

Manage a network of systems through gateway entry points

Not every queue manager will need to expose HTTPS endpoints

Queue managers >= V8 are able to be administered through those 9.0.x gateways



REST Admin Gateway

Requested by you!
RFE 104343 ✅

V9.0.4 CD

…

Allows a REST request to be sent via a gateway qmgr

Target qmgr specified in URL

GET /ibmmq/rest/v1/admin/qmgr/TARGET/queue

Gateway qmgr specified in one of two ways

Per request in a header:

ibm-mq-rest-gateway-qmgr: GATEWAY

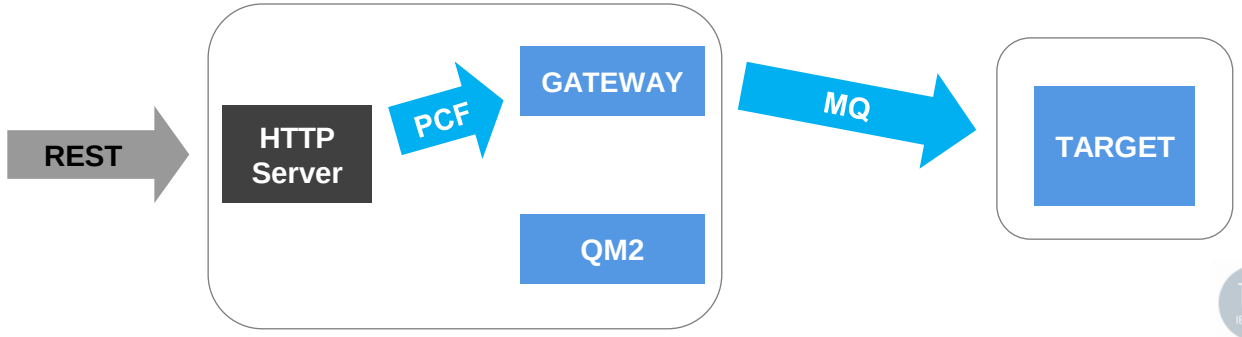
Or

Default in configuration:

<variable name="mqRestGatewayQmgr" value="DEFAULT.GATEWAY"/>

Relies on appropriate transmission queues, channels, security being set up for remote administration

Currently supported with: qmgr, queue and subscription REST API. Channels and MQSC coming



Simplified administration of the mqweb server (recap)

The mqweb server hosts the MQ Console and REST API

Configuration is xml based so can be done with any text editor

Samples provided to get you started quickly

A number of variables are provided to simplify set up of common things like ports, etc

```
C:\Program Files\IBM\MQ904GA\web\mq\samp\configuration>ls
basic_registry.xml  ldap_registry.xml  local_os_registry.xml  no_security.xml
```

```
<!--
Roles for the MQ REST API
-->
<enterpriseApplication id="com.ibm.mq.rest">
    <application-bnd>
        <security-role name="MQWebAdmin">
            <group name="mqm"/>
            <!--
            Add users and/or groups here for full admin access
            -->
        </security-role>
        <security-role name="MQWebAdminRO">
            <!--
            Add users and/or groups here for read-only access
            -->
        </security-role>
        <security-role name="MQWebUser">
            <special-subject type="ALL_AUTHENTICATED_USERS"/>
        </security-role>
    </application-bnd>
</enterpriseApplication>
```

```
<!--
Uncomment the following two variables, and adjust them, to change
the default CORS settings.
-->
<variable name="mqRestCorsAllowedOrigins" value="https://localhost:9883"/>
<variable name="mqRestCorsMaxAgeInSeconds" value="120"/>
```

V9.0.4 CD

...

Simplified administration of the mqweb server

Aim over time is to reduce the amount of xml editing

One and a half new control commands added

setmqweb: for altering configuration. Currently only variables are supported

dspmqweb: for displaying configuration. Currently only variables are supported

Existing ability to display status preserved

Available on z/OS too!

```
C:\>setmqweb properties -k httpsPort -v 12345
MQWB1100I: The 'setmqweb' command completed successfully.
```

```
C:\>dspmqweb properties -a
name="httpHost" value="localhost"
name="httpPort" value="-1"
name="httpsPort" value="12345"
name="ltpaExpiration" value="120"
name="maxTraceFileSize" value="20"
name="maxTraceFiles" value="2"
name="mqConsoleAutostart" value="true"
name="mqRestAutostart" value="true"
name="mqRestCorsAllowedOrigins" value=""
name="mqRestCorsMaxAgeInSeconds" value="0"
name="mqRestCsrfExpirationInMinutes" value="30"
name="mqRestCsrfValidation" value="true"
name="mqRestGatewayEnabled" value="true"
name="mqRestGatewayQmgr" value=""
name="mqRestMessagingEnabled" value="true"
name="mqRestRequestTimeout" value="30"
name="traceSpec" value="*=info"
MQWB1100I: The 'dspmqweb' command completed successfully.
```

Managing your distributed error logs

V9.0.3 CD

V9.0.4 CD

MQ 9.0.3 delivered improvements to the error log content:

- Universal timestamps
- Message severity (by request)

MQ 9.0.4 built on top of this:

- Message severity (by default)
- Easily parsable JSON error log formats in separate file (currently enabled through environment variable AMQ_ADDITIONAL_JSON_LOG=true)
- Message inserts listed separately
- Default log sizes increased (2Mb➔32Mb)
- mqrc utility extended to support language translation

```
05/10/2017 12:22:01 - Process(69708.1) User(another) Program(runmqchi.exe)
                      Host(DUMMY) Installation(MQNI09000400)
                      VRMF(9.0.4.0) QMgr(QM1)
                      Time(2017-10-05T11:22:01.376Z)
                      ArithInsert1(0) ArithInsert2(0)

AMQ9542W: Queue manager is ending.

EXPLANATION:
The program will end because the queue manager is quiescing.
ACTION:
None.
```

```
{
  "ibm_messageId":"AMQ9542W",
  "arith_insert_1":0,
  "arith_insert_2":0,
  "ibm_datetime":"2017-10-05T11:22:01.376Z",
  "ibm_serverName":"QM1",
  "type":"mq_log",
  "host":“DUMMY",
  "loglevel":"WARNING",
  "module":"amqrimna.c:1071",
  "ibm_sequence":"1507202521376344_4991908185865",
  "ibm_processId":69708,
  "ibm_threadId":1,
  "ibm_version":"9.0.4.0",
  "ibm_processName":"runmqchi.exe",
  "ibm_userName":“another",
  "ibm_installationName":"MQNI09000400",
  "ibm_installationDir":"C:\\Development\\errl2\\p900_D\\obj\\amd64_nt_4",
  "message":"AMQ9542W: Queue manager is ending."
}
```

High availability

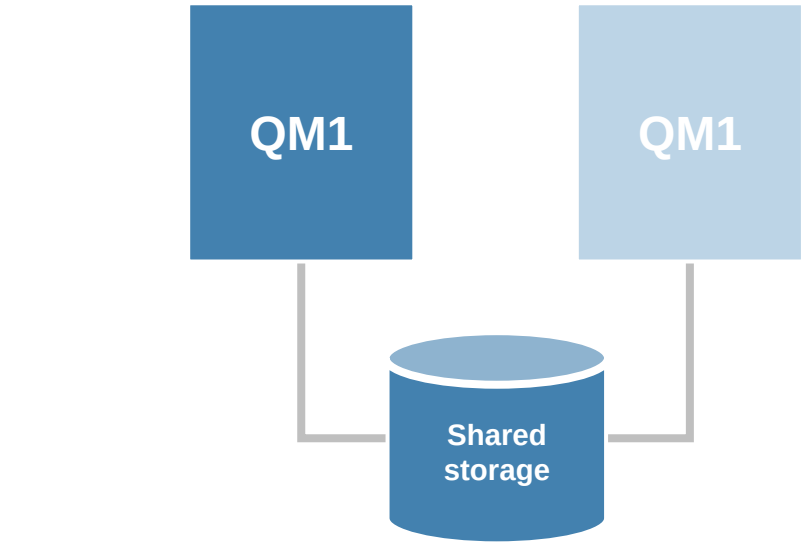
High availability, it's no longer a luxury

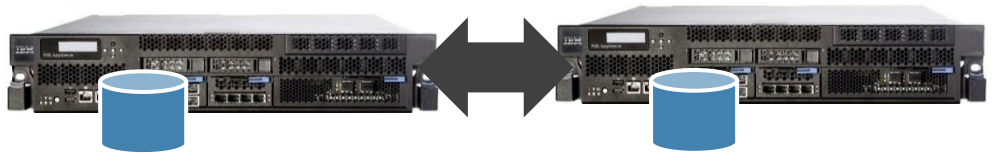
High availability is more important than ever. Users expect their messaging systems to always be available for work and for messages to reach their destination no matter what happens.

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.





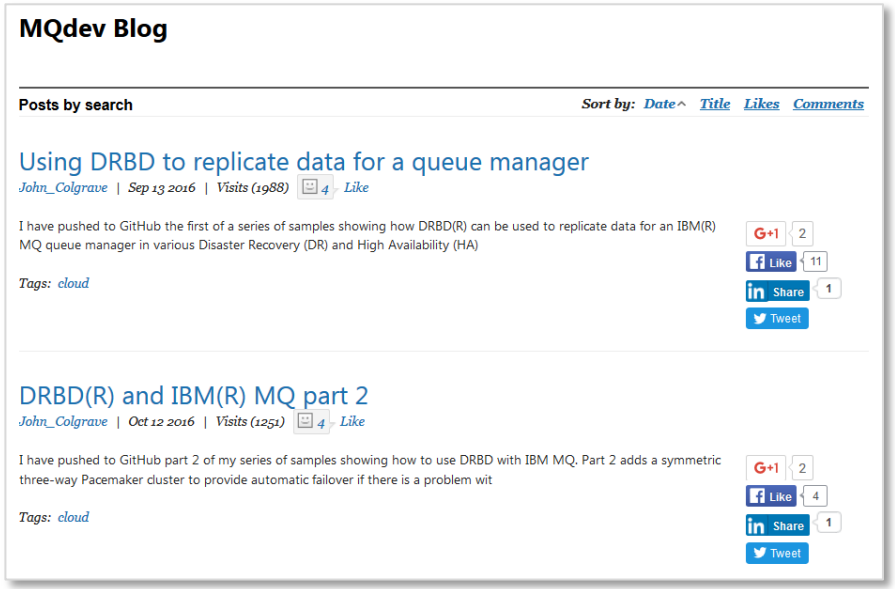
*** z/OS shared queue provides active-active HA of the message data**

Data replicated HA for Software MQ on Linux

The IBM MQ Appliance introduced shared nothing HA through direct data replication

One way to build your own equivalent is to look to a DRBD based solution (guidance given ➔)

But we have been working on an HA solution to provide this capability built directly into **IBM MQ Advanced**…



https://www.ibm.com/developerworks/community/blogs/messaging/

Replicated Data Queue Managers

IBM MQ Advanced | V9.0.4 CD | …

A **Linux only** HA solution with no external dependency

- No requirement for a network filesystem or for an HA cluster

Leverages the proven data replication and monitoring capabilities of **DRBD** and **Pacemaker**

MQ configures the underlying resources to make setup and operations natural to an MQ user

Three-way replication for **quorum** support

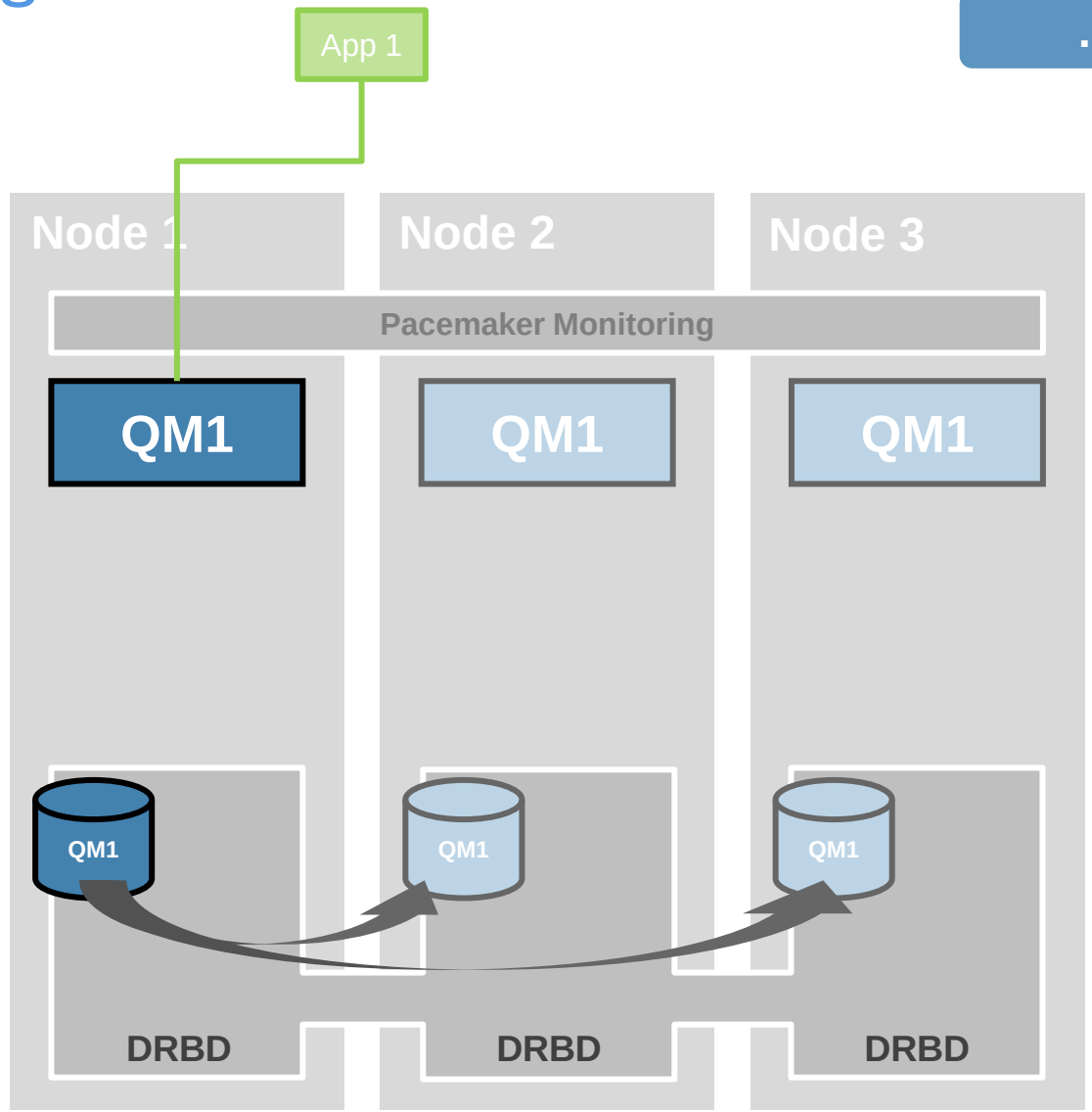
Synchronous data replication for once and once only delivery of messages

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

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

Floating IP support per queue manager to provide simple application setup

Delivered through CD, initial focus on **RHEL v7 on x86-64 only**



Replicated Data Queue Managers

IBM MQ Advanced | V9.0.4 CD | …

A **Linux only** HA solution with no external dependency

- No requirement for a network filesystem or for an HA cluster

Leverages the proven data replication and monitoring capabilities of **DRBD** and **Pacemaker**

MQ configures the underlying resources to make setup and operations natural to an MQ user

Three-way replication for **quorum** support

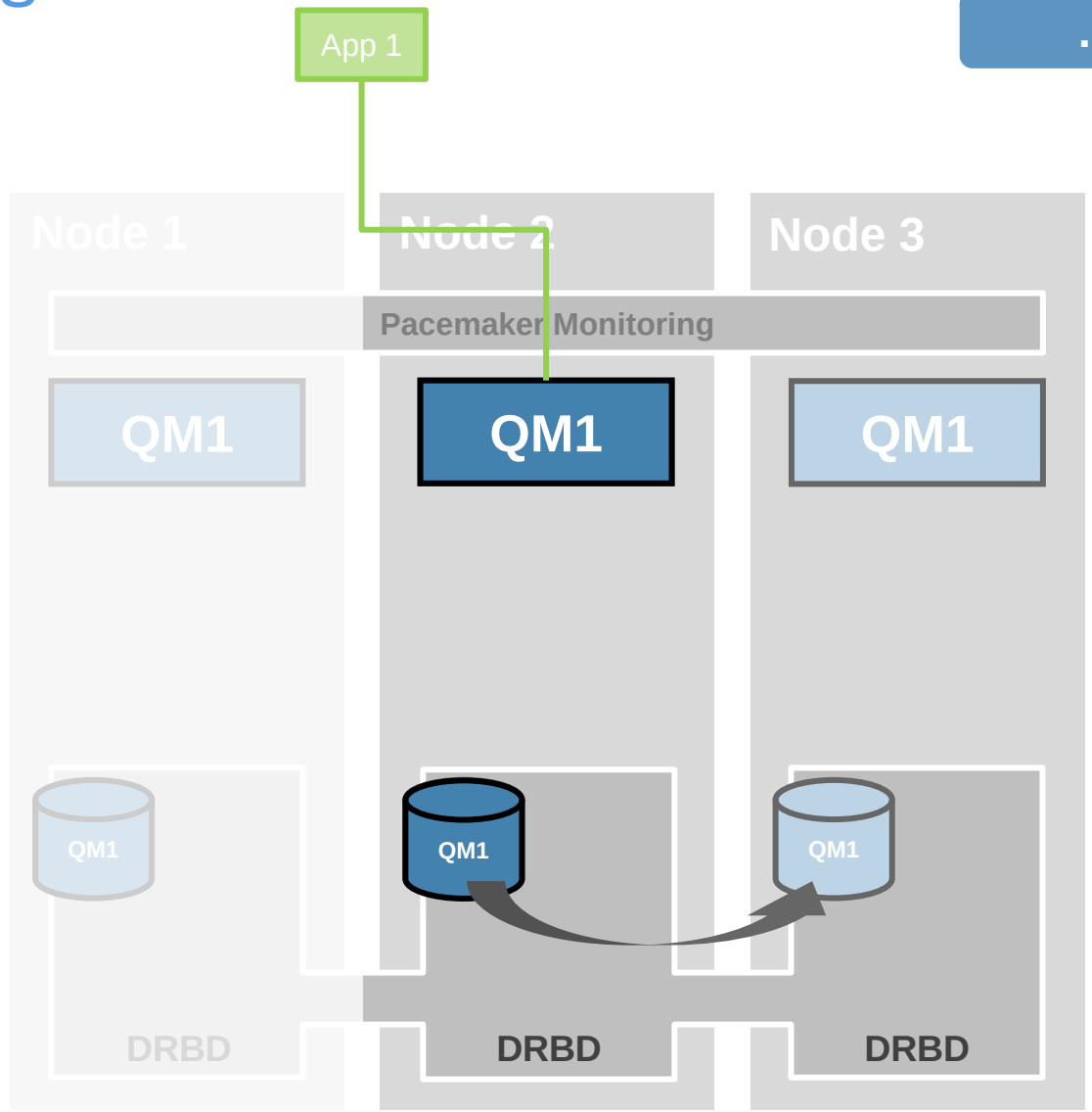
Synchronous data replication for once and once only delivery of messages

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

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

Floating IP support per queue manager to provide simple application setup

Delivered through CD, initial focus on **RHEL v7 on x86-64 only**



Replicated Data Queue Managers

IBM MQ Advanced | V9.0.4 CD | …

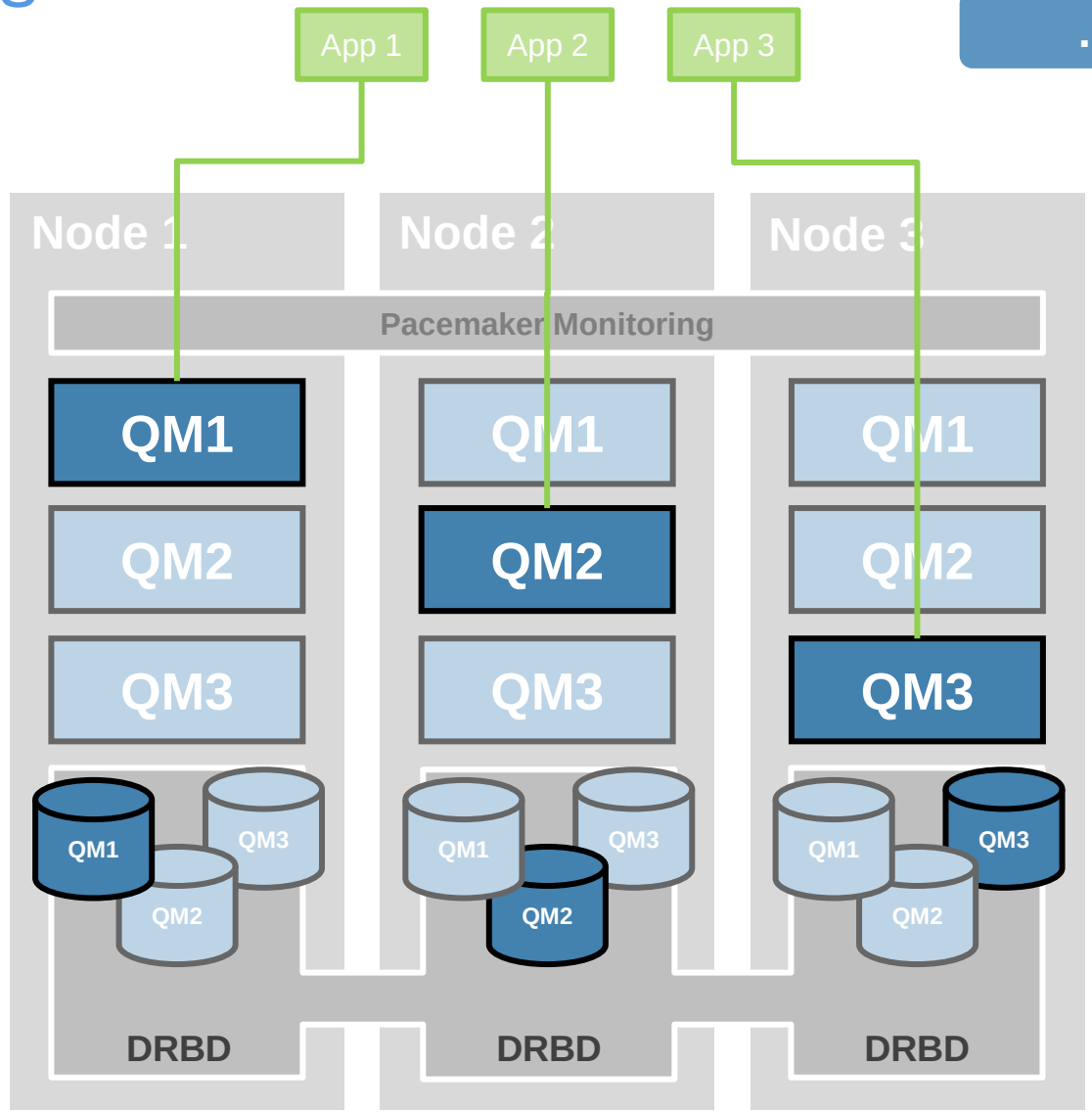
Recommended deployment pattern:

Spread the workload across multiple queue managers and distribute them across all three nodes

No one node runs at total capacity, allowing for failed over queue managers when required

MQ **licensing** is aligned to minimise costs

- One full **IBM MQ Advanced** license and two **High Availability Replica** licenses (previously named Idle Standby)



Supporting your applications

Multiple APIs and Protocols

IBM MQ supports multiple APIs and multiple client protocols. Both proprietary and open.

APIs: **MQI**, **JMS**, **MQ Light**, **REST** …

Protocols: **MQ**, **AMQP**, **MQTT**, **HTTP**

These support a wide range of application styles, from the simplest of messaging needs through to the richest

MQ is the transport, messages produced from any API or protocol can be received by any other API or protocol.

APIs	Protocols
MQI Exposes the full set of MQ capabilities Uses the MQ protocol	**MQ** MQ's highly reliable and performant messaging protocol
JMS 2.0 Supports the full JMS API for use in many JSE or JEE environments Uses the MQ protocol	**MQTT** Supports the open standard MQTT API and protocol Open source Eclipse Paho clients
MQ Light A simple pub/sub messaging API Uses the AMQP 1.0 protocol	**AMQP** Support for AMQP 1.0 enables support for open source clients such as Qpid Proton
REST A very simple but secure messaging API over HTTP…	**HTTP** Support for MQ

REST Messaging

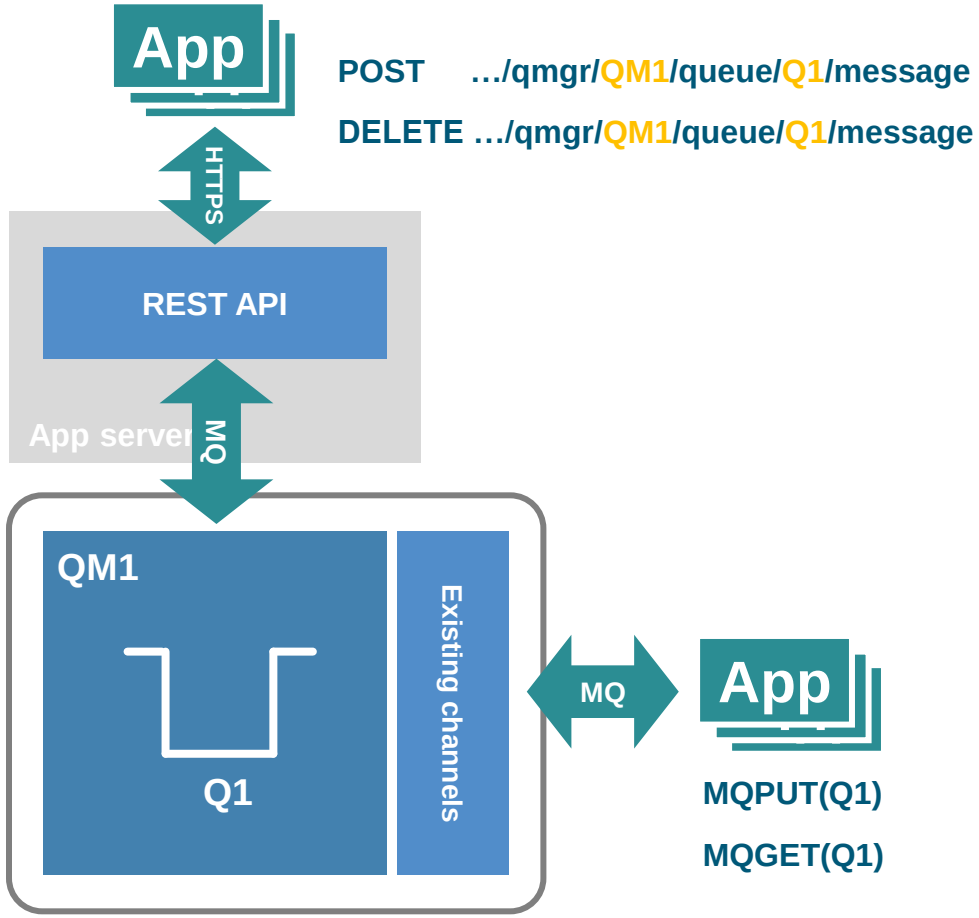
Many users just want the **simplest** way to get messages in and out of an MQ system

A RESTful API gives you just that. Easily enabling messaging from just about **any environment** with no need for an MQ client or platform/language limitations

MQ originally exposed its messaging capabilities using the ***IBM MQ bridge for HTTP***. This requires your own HTTP server and setup. This feature has been **deprecated** since MQ V8.

MQ support for z/OS Connect EE now provides a similar mechanism on z/OS

Neither provide all the components, ready configured, inside MQ



V9.0.4 CD

...

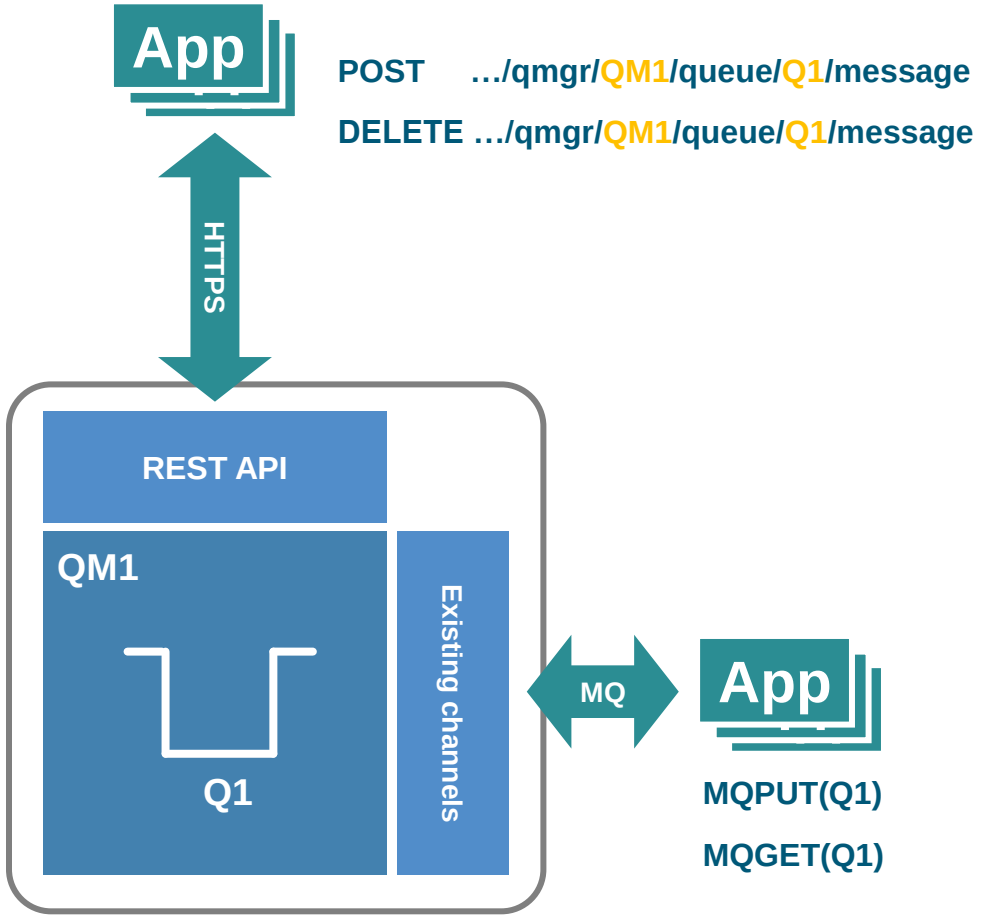
REST Messaging

The new HTTP server support in MQ 9.0.x provides the platform for a properly integrated REST API solution

CD releases start to see a messaging REST API evolve, first with simple synchronous point-to-point support for text based messages

Caller must be in MQWebUser role

Messaging API can be disabled independently of administration API



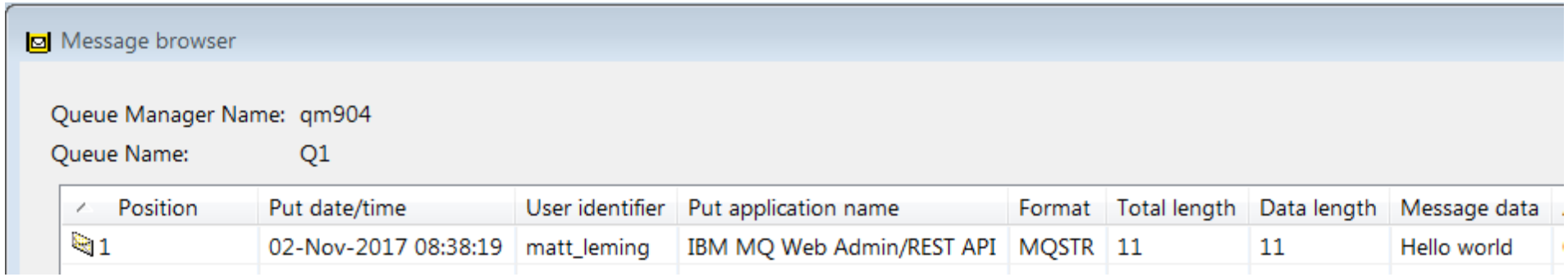
V9.0.4 CD

...

REST Messaging – sending a message

POST /ibmmq/rest/v1/messaging/qmgr/qm904/queue/Q1/message
-d "Hello world"
-H "Content-Type: text/plain“

-u matt_leming:password


Message browser

Queue Manager Name: qm904

Queue Name: Q1

Position	Put date/time	User identifier	Put application name	Format	Total length	Data length	Message data
1	02-Nov-2017 08:38:19	matt_leming	IBM MQ Web Admin/REST API	MQSTR	11	11	Hello world

Caller can specify message expiry, correlation ID, persistence, etc using HTTP headers

V9.0.4 CD

...

REST Messaging – getting a message

```
C:\>curl -k -X DELETE https://localhost:9443/ibmmq/rest/v1/messaging/qmgr/qm904/queue/Q1/message -u matt_leming:password
Hello world
C:\>curl -k -X DELETE https://localhost:9443/ibmmq/rest/v1/messaging/qmgr/qm904/queue/Q1/message -u matt_leming:password

C:\>_
```

Also supports:

- **Get by message/correlation ID**
- **Get with wait**

Certain MQMD fields returned as headers

- **Message/correlation ID etc**

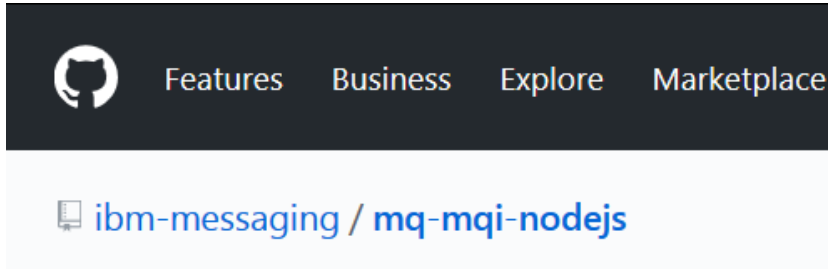
Breaking news: Node.js wrapper for the MQI

Enables Node.js developers to send and receive messages via MQ, and interact with other MQ-enabled applications in the organisation

See https://developer.ibm.com/messaging/2017/11/10/ibm-mq-node-js/ for more info

The most important verbs and structures implemented

Would appreciate feedback so we can judge if further there is value in providing more




```
mq.Conn(qMgr, function(err,hConn) {
   if (err) {
      console.log(formatErr(err));
   } else {
     console.log("MQCONN to %s successful ", qMgr);

     // Define what we want to open, and how we want to open it.
     var od = new mq.MQOD();
     od.ObjectName = qName;
     od.ObjectType = MQC.MQOT_Q;
     var openOptions = MQC.MQOO_INPUT_AS_Q_DEF;
     mq.Open(hConn,od,openOptions,function(err,hObj) {
       if (err) {
         console.log(formatErr(err));
       } else {
         console.log("MQOPEN of %s successful",qName);
         // And loop getting messages until done.
         getMessages(hObj);
       }
       cleanup(hConn,hObj);
     });
   }
});
```

Connectivity

Bridging MQ with external systems

As well as connecting a wide array of applications directly to an MQ system, there are a growing set of bridges between MQ and external systems, many of them cloud services

Message Hub

Connect MQ to the IBM Cloud (ex-Bluemix) messaging service

Salesforce

Integrate MQ's publish/subscribe with Salesforce events

Blockchain

Use MQ messages to query and update the blockchain

Kafka

Use open source Kafka connectors to join MQ with Kafka

IBM Integration Bus

IBM's integration solution for everything else! Also note that IIB can also be used as a fully supported gateway today between MQ and Blockchain, Kafka, etc.

MQ Bridge for Salesforce

V9.0.2 CD

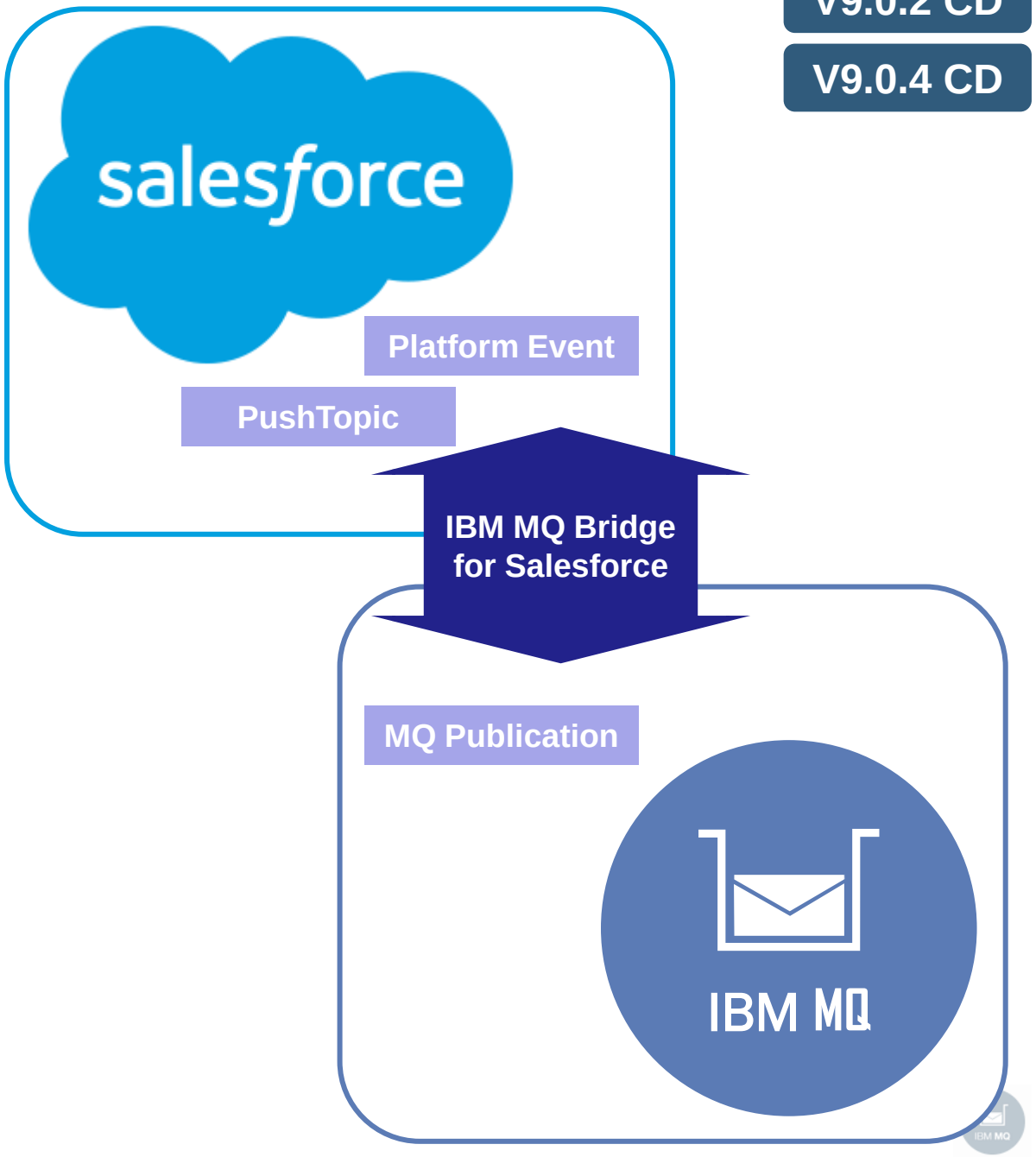
V9.0.4 CD

Safesforce's cloud-based CRM platform enables events to be emitted when changes are made to data, or when applications run.

You can inject these Salesforce events into your own systems using the new MQ Bridge for Salesforce with no need for your backend applications to connect to Salesforce.

With V9.0.4, similarly, events originating in MQ can be injected into Salesforce.

The bridge runs on Linux, but connects to any queue manager and is enabled for monitoring with system topic metrics.



IBM MQ Advanced

MQ Adv z/OS VUE V9.0.3 CD

V9.0.4 CD

Blockchain Bridge

MQ is a natural fit to connect existing business transaction systems to remote intra-business ledger services

Asynchronous request reply MQ message flow for applications to **request and update information from Blockchain** over MQ messages

Originally provided with MQ Advanced for z/OS VUE V9.0.3, and now with V9.0.4, with MQ **Linux** server and client images

Connects to Hyperledger Fabric networks in IBM Cloud (ex-Bluemix), both public and (with 9.0.4) local

With 9.0.4, supported for use with (distributed) 9.0.x MQ Advanced queue managers also

With 9.0.4, Blockchain ledger updates supported (was previously only for queries)

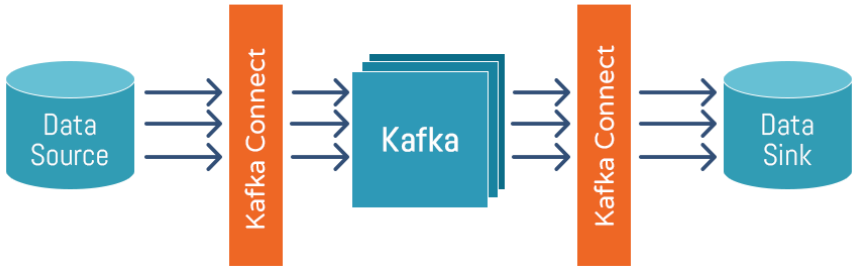
Kafka Connectors for MQ

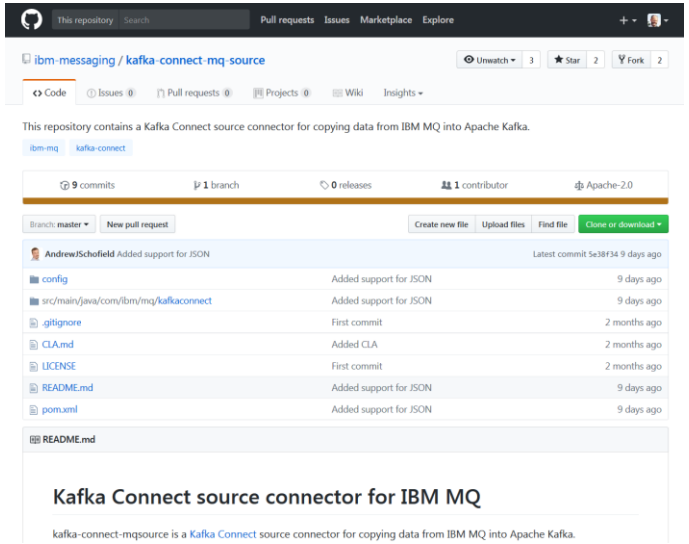
There is an increasing need to bridge between your MQ messaging platform and your Kafka stream data platform, unlocking your enterprise and your stream data

Kafka Connect is a framework included in Apache Kafka that enables this

IBM MQ sink and source connectors are currently being openly developed by IBM

https://www.confluent.io/product/connectors/





Managed File Transfer

IBM MQ Advanced

V9.0.1 CD

V9.0.2 CD

V9.0.3 CD

V9.0.4 CD

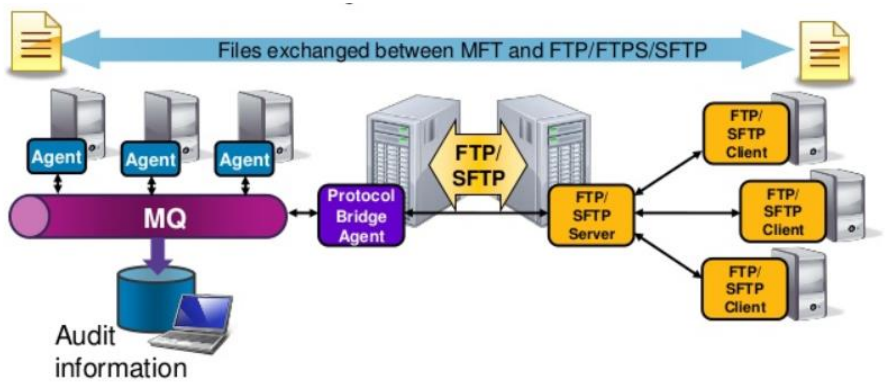
…

Simplified licensing

MFT Agents are now no longer separately and individually licensed but are free to deploy and use when connected to **MQ Advanced entitled queue managers**

Redistributable MFT Agent

The MFT agent now available from FixCentral, users simply **download and unzip.** See KC "agent_zip_configure" for info.



Improved diagnostics

Comprehensive fine grain coverage of FTP errors

Enhanced logging of FTP communications for error diagnosis

File transfer recovery timeout control, new option to automatically cancel failing transfers

MFT agent status reporting, aids problem diagnosis by reporting last contact time

Client support for file logger

Only for Files logging; Database mode still requires a local queue manager

Securing MQ

V9.0.4 CD

CHLAUTH/CONNAUTH enhancements

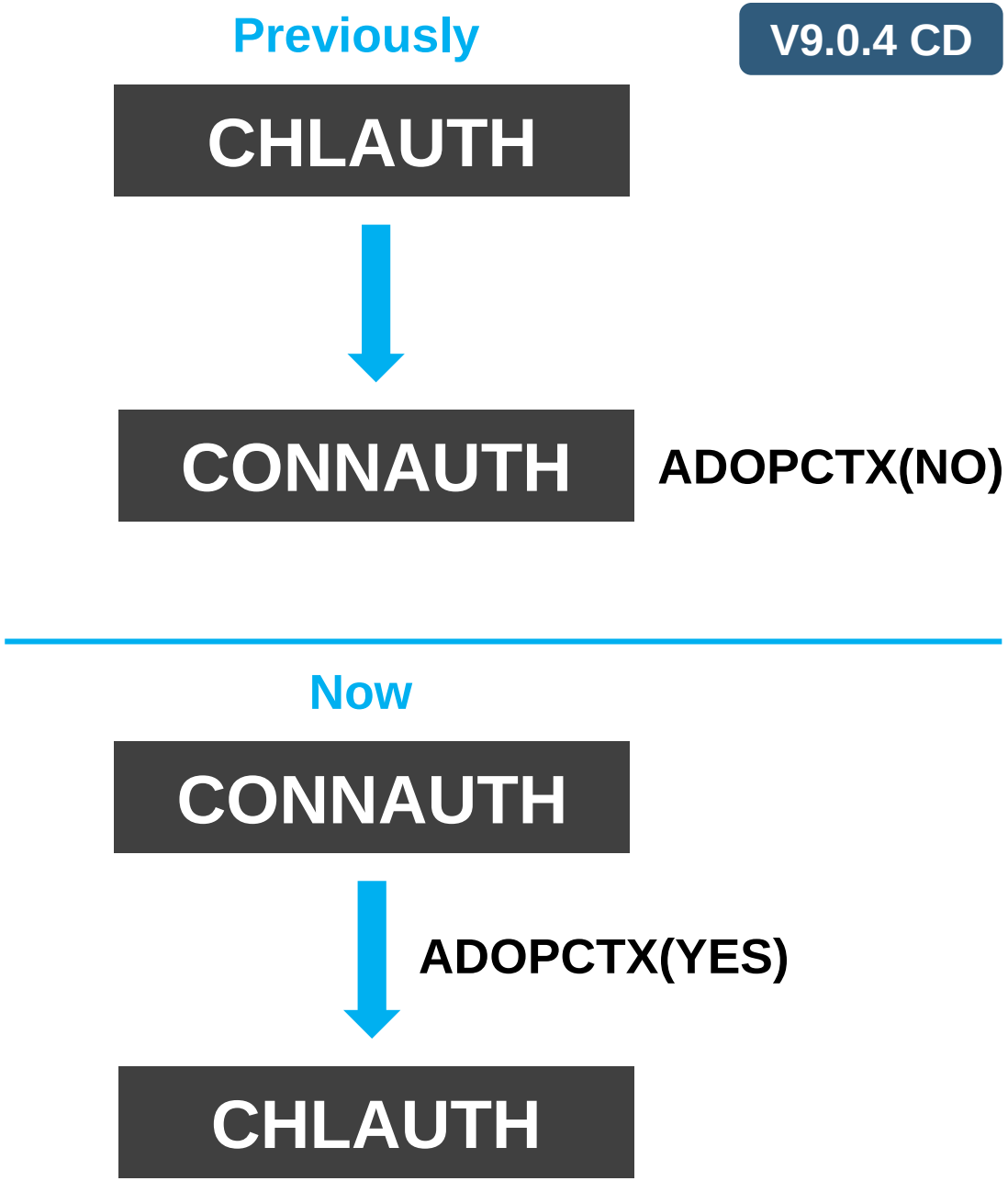
Queue manager defaults changed so that new queue managers will perform connection authentication, and adopt the authenticated user's context, before CHLAUTH rules are applied

A more natural approach to authentication with channels, following current best practices

Supported on all platforms including z/OS

Follows the recommendations laid out here
https://www.ibm.com/developerworks/community/blogs/messaging/entry/Connection_Authentication_Channel_Authentication_interactions

Behaviour of migrated queue managers doesn't change on Distributed



New authentication option with mqweb server (recap)

mqweb server authentication is provided by a registry

Previously the following were available, and provided by WLP:

basic: user ids and passwords provided in xml

LDAP: on both z/OS and Distributed platforms

SAF: on z/OS only. No equivalent for Distributed platforms

```
<!--
Sample Basic Registry
-->
<basicRegistry id="basic" realm="defaultRealm">
    <!--
    This sample defines two users with unencoded passwords
    and a group, these are used by the role mappings above.
    -->
    <user name="mqadmin" password="mqadmin"/>
    <user name="mqreader" password="mqreader"/>
    <group name="MQWebAdminGroup">
       <member name="mqadmin"/>
    </group>
</basicRegistry>
```

New authentication option with mqweb server

New “Local OS Authentication” registry provided for Distributed

Delegates down to OS, so users/groups need to be defined there

No need to worry about passwords

PAM support provided too

```
<!--
Local OS User Registry settings
usePam controls whether user authentication uses Pluggable
Authentication Modules (PAM). On platforms that don't support
PAM the value of this attribute is ignored.
-->
<localOSAuthenticationMQ usePam="true"/>
```

```
<featureManager>
    <feature>appSecurity-2.0</feature>
    <feature>localOSAuthenticationMQ-1.0</feature>
    <feature>basicAuthenticationMQ-1.0</feature>
</featureManager>
```

```
<!--
Roles for the MQ REST API
-->
<enterpriseApplication id="com.ibm.mq.rest">
    <application-bnd>
        <security-role name="MQWebAdmin">
            <group name="mqm"/>
            <!--
            Add users and/or groups here for full admin access
            -->
        </security-role>
        <security-role name="MQWebAdminRO">
            <!--
            Add users and/or groups here for read-only access
            -->
        </security-role>
        <security-role name="MQWebUser">
            <special-subject type="ALL_AUTHENTICATED_USERS"/>
        </security-role>
    </application-bnd>
</enterpriseApplication>
```

Platforms


z/OS

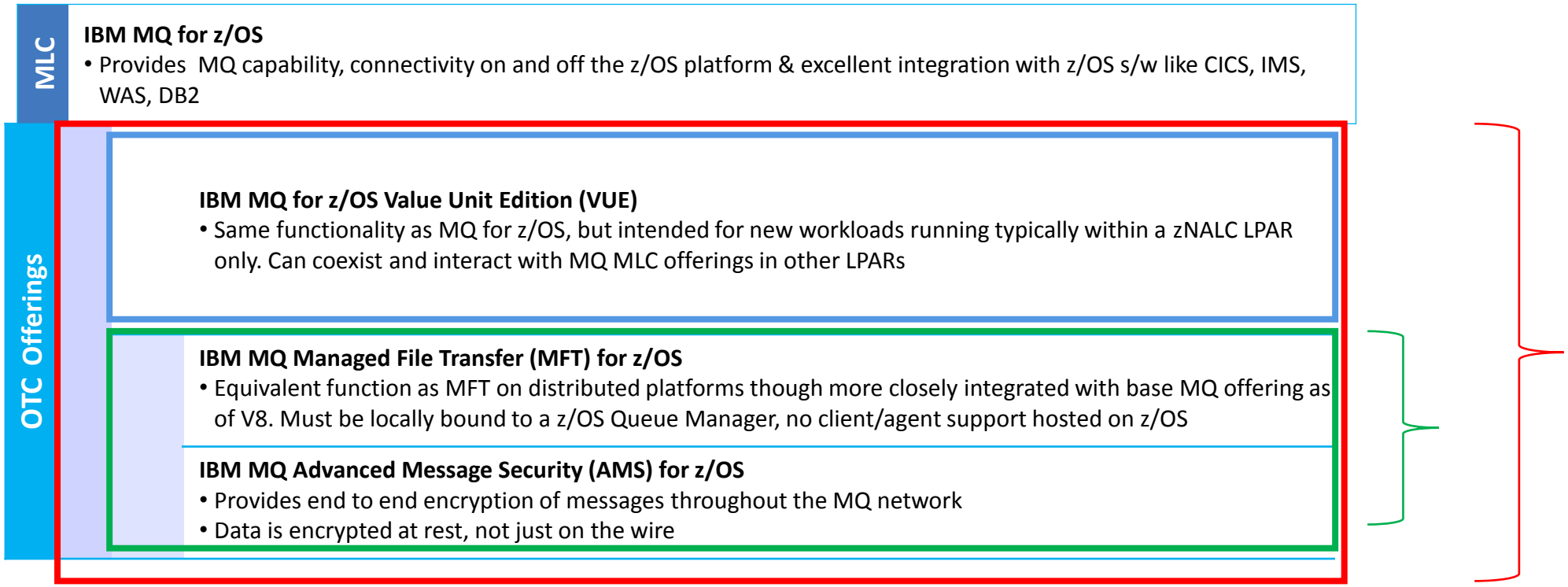
Quick Reminder: MQ for z/OS Portfolio

MLC

IBM MQ for z/OS

- Provides MQ capability, connectivity on and off the z/OS platform & excellent integration with z/OS s/w like CICS, IMS, WAS, DB2

Quick Reminder: MQ for z/OS Portfolio



IBM MQ Advanced for z/OS

- Soft bundling of MQ MFT for z/OS and MQ AMS for z/OS only (i.e. *no MQ z/OS included*)
- Can be deployed with MQ MLC or VUE offerings

IBM MQ Advanced for z/OS Value Unit Edition (VUE)

- Soft bundling of IBM MQ for z/OS VUE + IBM MQ MFT for z/OS + IBM MQ AMS for z/OS

MQ Advanced for z/OS VUE unique function

IBM MQ Advanced

V9.0.3 CD

The Richest Set of z/OS Messaging Capabilities in a Single, Simple to Deploy Offering

IBM Cloud Product Insights support

Provides registration and usage information to the Cloud Product Insights Service to offer insight into the usage of the entire MQ estate across z/OS and distributed systems

MQ Blockchain connector

Enables application integration with the IBM Blockchain service running in IBM Cloud (ex-Bluemix), mediated via MQ.

Managed File Transfer Agent Connectivity

Allows a z/OS Managed File Transfer Agent to remotely connect to a z/OS Queue Manager to simplify the deployment of MFT on z/OS

The same MFT workload may require fewer z/OS queue managers

MQ Advanced for z/OS VUE unique function

IBM MQ Advanced

V9.0.4 CD

The Richest Set of z/OS Messaging Capabilities in a Single, Simple to Deploy Offering

Support for client connections with JMS/Java

Allows batch Java applications to connect as a client to Advanced VUE queue managers. Previously client connections were only available in WebSphere Application Server

IBM Cloud Product Insights support via proxy

No need for direct access to internet

(Also available on Distributed in 9.0.4)

Distributed

V9.0.4 CD

Ability to switch between circular and linear logging

MQ V9.0.2 introduced new abilities to manage recovery logs.

V9.0.4 adds new migmqlog control command introduced

```
>>-migmqlog-- -m -QMgrName--+--------------------+--+------+---><
                            '- -ld -New log path-'  +- -ll +
                                                    '- -lc '
```

Two capabilities:

- Can move logs to Advanced Format disk on Windows (4096 bytes per sector)
- Can switch between logging methods on Windows and UNIX platforms

Queue manager must be stopped!

Logs can be migrated “in place” or to new location

MQ Appliance

V9.0.4 CD

...

SAN support on the appliance

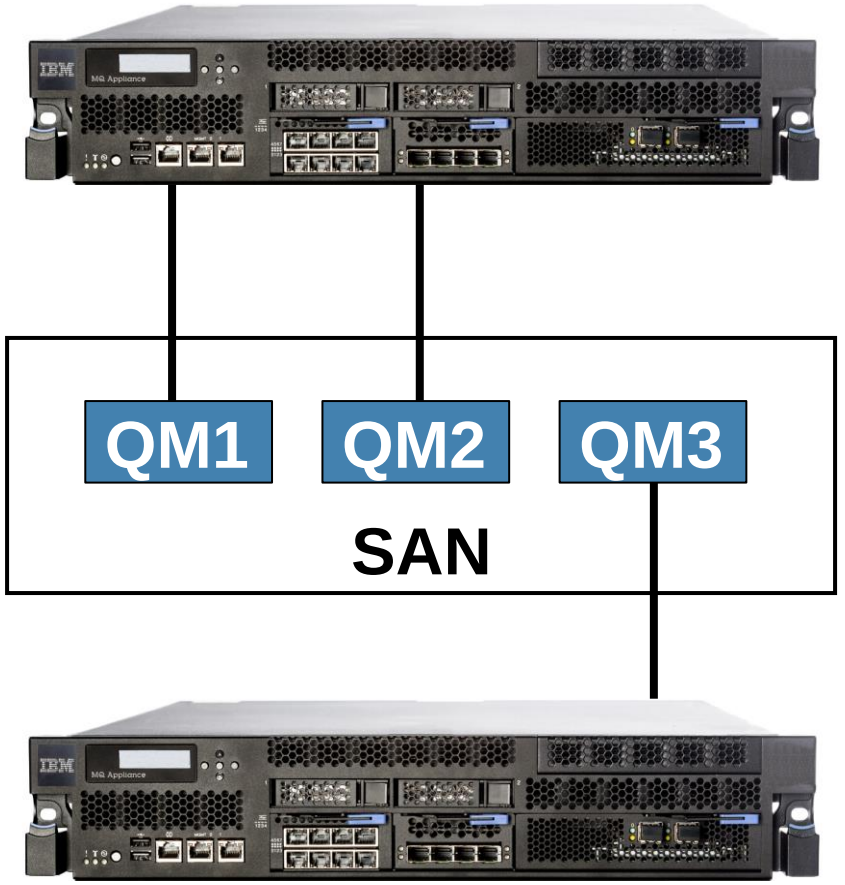
Configure an appliance to store queue manager logs and data on SAN storage

Requested by customers with high storage or I/O performance requirements, or who employ a SAN solution for disaster recovery

One SAN volume per queue manager

Reconnect the same, or a different appliance, to the SAN storage in response to a failure

HA queue managers not currently supported



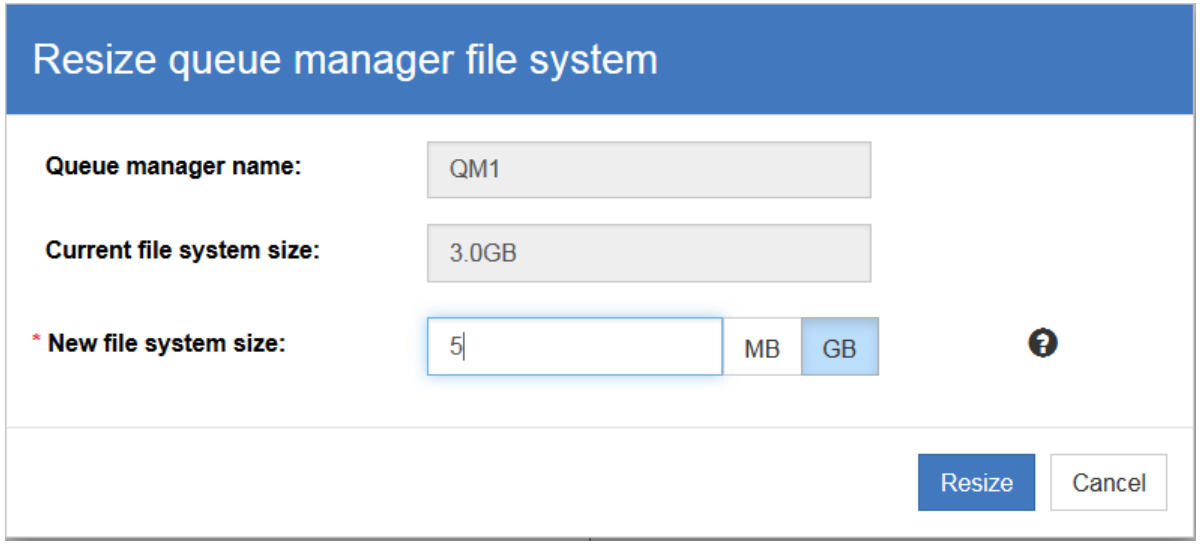
Resize queue managers on the appliance

You can now expand the size of the file system used by a queue manager after you have created the queue manager

Provides constraint relief in response to growing workloads

Resizing queue managers configured for HA, DR or SAN is not currently supported

- **Can temporarily remove HA/DR configuration to resize**



MQ for HPE NonStop

IBM MQ for HPE NonStop

MQ V8 functionality is now offered on the HPE Integrity NonStop platform for both the **NonStop X** servers and **NonStop i** servers

Released in June 2017

Provides many of the MQ V8 capabilities plus some of the unique HP NonStop capabilities from the previous version, MQ V5.3

Will follow the **Continuous Delivery** model of incrementally extending the capabilities to bring functional parity with the HP NonStop V5.3 features

AIX

AIX

Requested by you! ✅

V9.0.4 CD

To streamline continuous delivery the MQ CD 9.0.x releases are focused on the most active platforms. This was initially restricted to the following platforms

Linux, z/OS, Windows, MQ Appliance

High customer demand has meant we have added **AIX** to the list of CD platforms from 9.0.4

Help shape the future of MQ…

ibm.biz/MQ-Customer-Survey