IBM Messaging in the Cloud

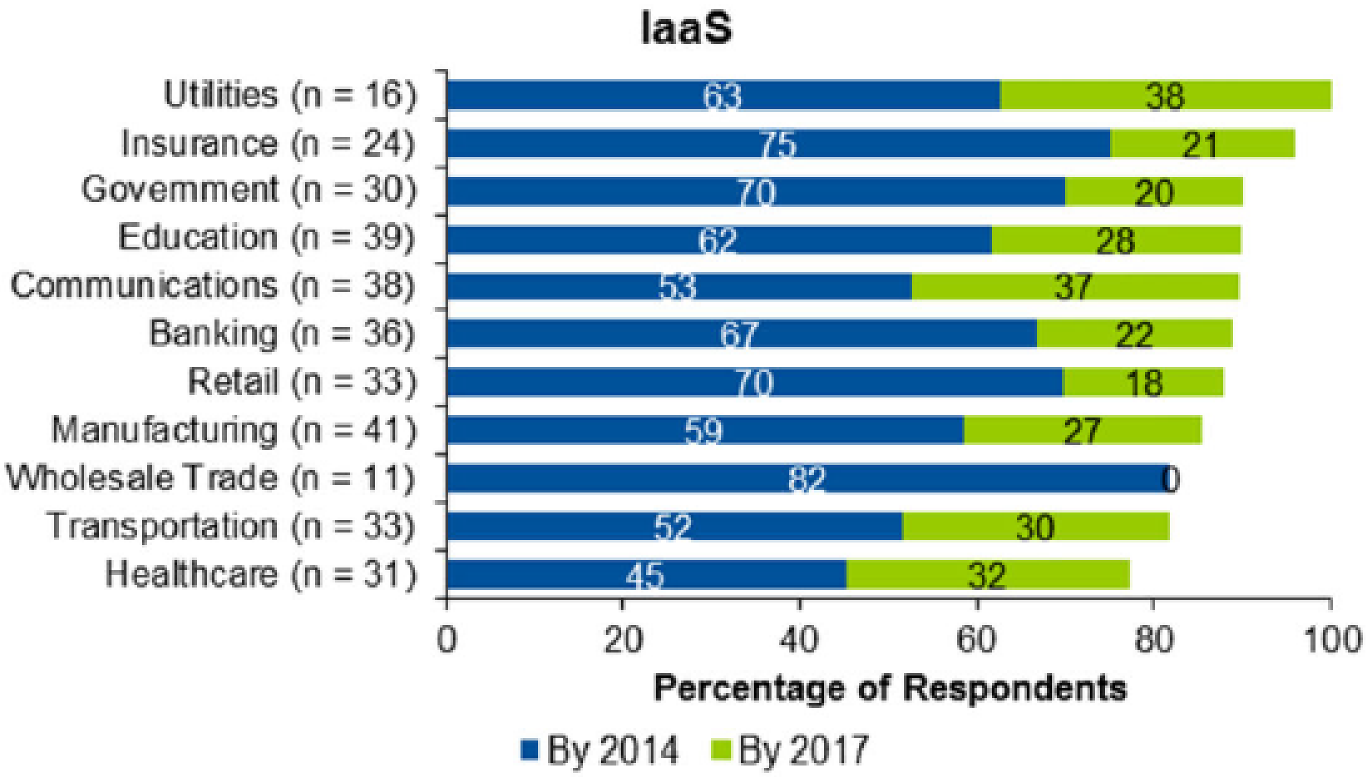
Matthew Whitehead
IBM MQ Development
mwhitehead@uk.ibm.com

Agenda

- Recap of cloud options
- Overview of IBM messaging solutions
- Running MQ in different cloud architectures
  - Deploying and Installation
  - Persisting data
  - Availability
  - Monitoring and metrics
  - Security
  - Service discovery
  - Cloud ingress/egress

Why Cloud?

Percentage of organizations currently using or planning to use cloud services by industry



Why Cloud?

- Doing more with less
- Being more ready to change
- Making the development process less heavyweight
- Paying for what you use
- Integrating with other cloud services
- Rapidly scaling up and down with demand

Which type of cloud?

IaaS? PaaS? CaaS? FaaS? SaaS?

Not made up

Which Type of Cloud?

- **IaaS** (Infrastructure-as-a-Service - VMs)
  - Are good for large services/apps, but generally not ideal
  - May be used more like physical machines, but with added flexibility

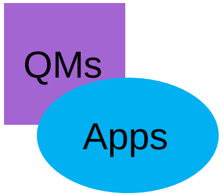
- **CaaS** (Containers-as-a-Service - e.g. Kubernetes)
  - Are good for micro-services/apps
  - Potentially quite short-lived

- **PaaS** (Platform-as-a-Service - e.g. Bluemix, Cloud Foundry)
  - Are great for application code in general
  - Handing off infrastructure worries to someone else

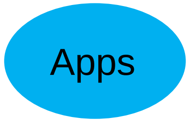
- **FaaS** (Functions-as-a-Service - e.g. OpenWhisk, AWS Lambda)
  - Could be used for occasional compute loads
  - Will likely drive lots of short-lived connections, so may not perform well for some messaging workloads
  - Most support JavaScript (could use the MQ Light API), but some can support Java, C# and more

Which Type of Cloud?

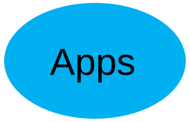
- **IaaS** (Infrastructure-as-a-Service - VMs)
  - Are good for large services/apps, but generally not ideal
  - May be used more like physical machines, but with added flexibility



- **CaaS** (Containers-as-a-Service - e.g. Kubernetes)
  - Are good for micro-services/apps
  - Potentially quite short-lived



- **PaaS** (Platform-as-a-Service - e.g. Bluemix, Cloud Foundry)
  - Are great for application code in general
  - Handing off infrastructure worries to someone else



- **FaaS** (Functions-as-a-Service - e.g. OpenWhisk, AWS Lambda)
  - Could be used for occasional compute loads
  - Will likely drive lots of short-lived connections, so may not perform well for some messaging workloads
  - Most support JavaScript (could use the MQ Light API), but some can support Java, C# and more

Which Cloud?

amazon web services

- EC2
- EBS/S3/EFS
- Lambdas
- Cloudwatch
- …

Microsoft Azure

- VMs
- App Service
- Active Directory
- IoT Hub
- Visual Studio Services
- …

IBM Bluemix™

SOFTLAYER® an IBM Company

- VMs
- OpenWhisk
- BluemixContainer Service
- CloudFoundry
- Logmet
- …

Google Cloud Platform

- Compute Engine
- App Engine
- Container Engine
- Cloud Functions
- BigQuery
- …

Poll Time!

- Show of hands for:
  - Virtual Machines (EC2, Bluemix VMs, Azure VMs, On-prem etc.)
  - Containers (Docker, AWS ECS, Bluemix Containers etc.)
  - Hybrid Messaging (Linking on-prem messaging to cloud-apps)
  - Open Stack (On premise, or in the cloud?)

- Production, development, test?

Containers

docker

IBM Containers

Google Container Engine

Microsoft Azure

Azure Container Service

MESOS

- Containers provide a similar environment to a VM but lighter in weight
  - A virtual machine provides an abstraction of the physical hardware
  - A container abstracts the OS level, typically at the user level

- Linux containers
  - Containers all share the same OS kernel
  - Images are constructed from layered filesystems
  - Containers isolate applications from each other and the underlying infrastructure

Container Orchestration

kubernetes

MARATHON

docker SWARM

IBM Containers

amazon web services

Google Container Engine

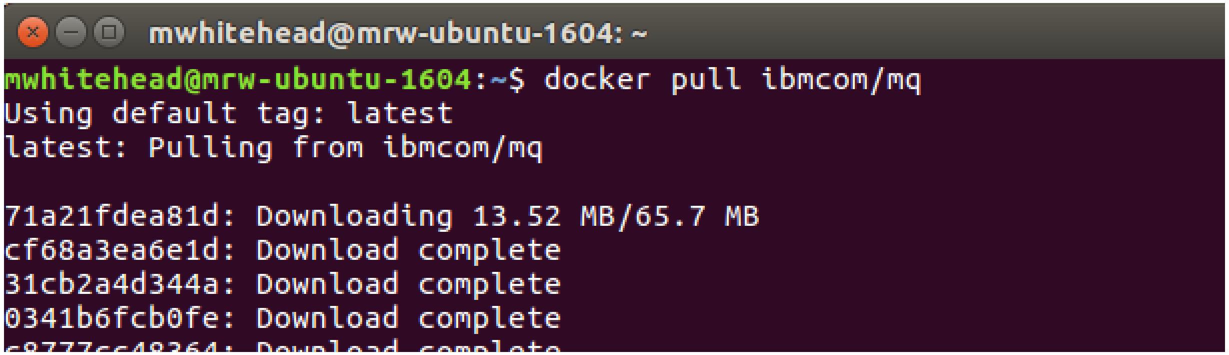
Microsoft Azure

Azure Container Service

- Running a single container is one thing, but…
- … deploying a more sophisticated architecture is more complex
    - Inter-Container Communication
    - Shared storage
    - Monitoring processes
    - Logging processes
    - Scalability
- Various technologies exist to manage larger groups of containers
- The different cloud container providers offer some of these features

MQ Docker Container

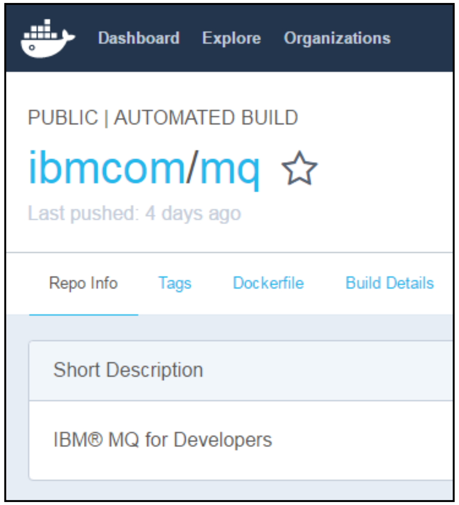
- MQ 8.0.0.4+ supported to run inside a Docker image
  - Details: https://ibm.biz/mqdocker
- Brings the benefits of Docker to MQ
  - Lightweight containers for running MQ
  - Predictable and standardized units for deploying MQ
  - Process, resource and dependency isolation
- IBM samples for customizing and building your own Docker images
  - Runs an MQ queue manager inside a container, isolated from the rest of your system



```
mwhitehead@mrw-ubuntu-1604:~$ docker pull ibmcom/mq
Using default tag: latest
latest: Pulling from ibmcom/mq

71a21fdea81d: Downloading 13.52 MB/65.7 MB
cf68a3ea6e1d: Download complete
31cb2a4d344a: Download complete
0341b6fcb0fe: Download complete
c8777cc48364: Download complete
```

Binary image in Docker Hub



PUBLIC | AUTOMATED BUILD

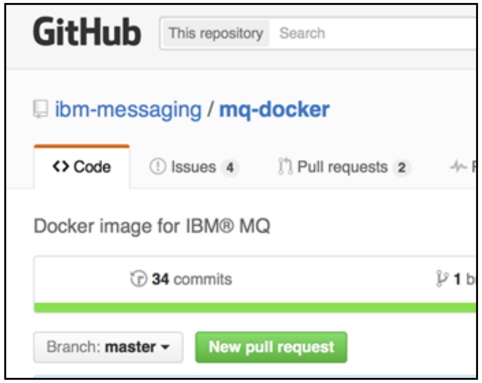
ibmcom/mq ☆

Last pushed: 4 days ago

Short Description

IBM® MQ for Developers

Source in GitHub



ibm-messaging / mq-docker

Docker image for IBM® MQ

MQ Docker Container

IBM MQ
+
docker

- Consideration needs to be given to:
  - Where /var/mqm data goes when the container stops
  - How to name queue managers
  - Changing channel definitions with updated IP address
    - In many container environments a re-provisioned container is given a new IP address
  - How you approach scaling down
  - The difference between more long lived containers (perhaps running full repositories) and short lived containers
- May be useful simply for basic, on-prem scenarios to reduce complexity

Agenda

- Recap of cloud options
- Recap of IBM messaging solutions
- Running MQ in different cloud architectures
  - Deploying and Installation
  - Persisting data
  - Availability
  - Monitoring and metrics
  - Security
  - Service discovery
  - Cloud ingress/egress

IBM Messaging Solutions

MQ	MessageHub	MQ Light	MessageSight and Bluemix IoT
Enterprise features	Message streams	Developer focussed API	IoT scenarios
24x7, 365	Based on Kafka	Dev tooling	MQTT protocol
Multi-platform (z/OS, IBM I, Unixes, Windowses)	Very high throughput, highly scalable	Free SDK server download	Good for very high client numbers (10,000s)
Rich feature set (lots of dials and options)	Ideal for Bluemix micro-services	Deploy against MQ and MessageHub	
Some static components (e.g. full repositories)	Currently Bluemix only		

IBM Messaging Solutions – Cloud Use-Cases

- MQ
  - Hybrid messaging – cloud to/from on-prem
  - Inter-cloud communication
  - Deployable anywhere that can host VMs or containers

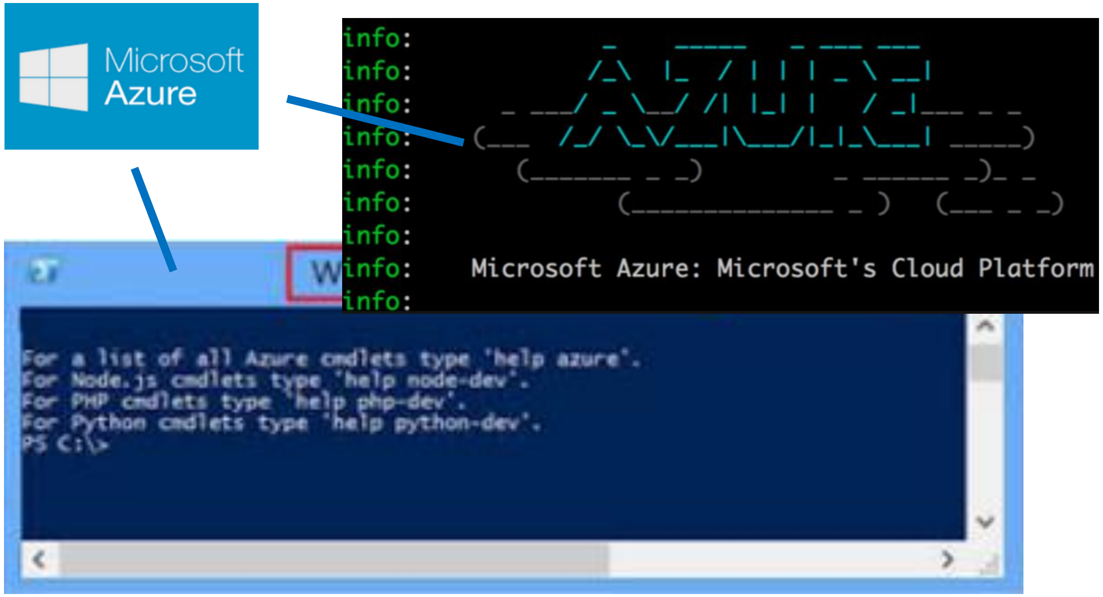
IBM MQ

- MessageHub
  - Ideal for micro-service architectures
  - Streaming/Real-time analytics
  - Bluemix/Softlayer only
- MQ Light
  - Rapid development of cloud applications
  - Deploy against MessageHub and MQ
- MessageSight and IoT
  - Ideal for connecting huge numbers of devices
  - Low-bandwidth network scenarios (e.g. mobile)

Agenda

- Recap of cloud options
- Overview of IBM messaging solutions
- Running MQ in different cloud architectures
  - Deploying and Installation ←
  - Persisting data
  - Availability
  - Monitoring and metrics
  - Security
  - Service discovery
  - Cloud ingress/egress

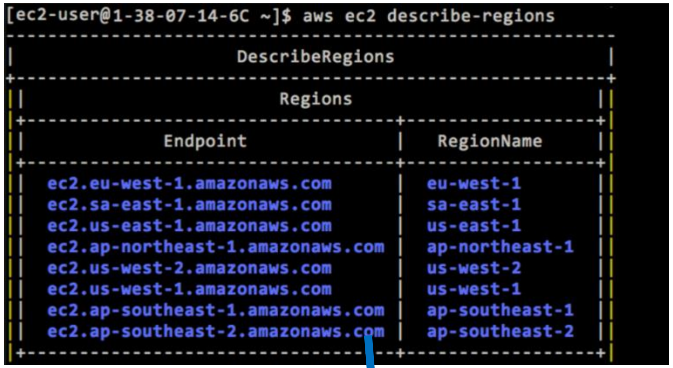
Deploying



Microsoft Azure

```
info:             _    _____   _ ___ ___
info:            /_\  |_  / | | | _ \ __|
info:      _ ___/ _ \__/ /| |_| |   / _|___ _ _
info:    (___  /_/ \_\/___|\___/|_|_\___| _____)
info:       (_______ _ _)         _ ______ _)_ _
info:              (______________ _ )   (___ _ _)
info:
info:     Microsoft Azure: Microsoft's Cloud Platform
info:
```

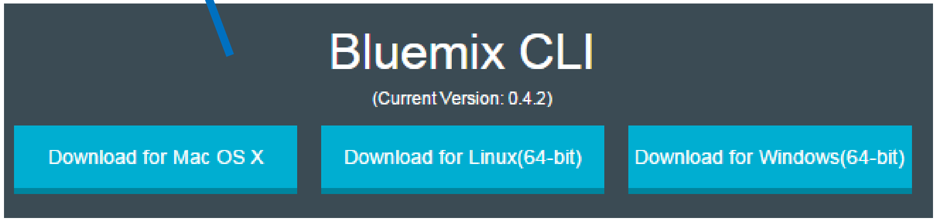
```
For a list of all Azure cmdlets type 'help azure'.
For Node.js cmdlets type 'help node-dev'.
For PHP cmdlets type 'help php-dev'.
For Python cmdlets type 'help python-dev'.
PS C:\>
```



```
[ec2-user@1-38-07-14-6C ~]$ aws ec2 describe-regions
--------------------------------------------------------------
|                      DescribeRegions                       |
+------------------------------------------------------------+
||                          Regions                         ||
|+-------------------------------------+--------------------+|
||              Endpoint               |    RegionName      ||
|+-------------------------------------+--------------------+|
||  ec2.eu-west-1.amazonaws.com        |  eu-west-1         ||
||  ec2.sa-east-1.amazonaws.com        |  sa-east-1         ||
||  ec2.us-east-1.amazonaws.com        |  us-east-1         ||
||  ec2.ap-northeast-1.amazonaws.com   |  ap-northeast-1    ||
||  ec2.us-west-2.amazonaws.com        |  us-west-2         ||
||  ec2.us-west-1.amazonaws.com        |  us-west-1         ||
||  ec2.ap-southeast-1.amazonaws.com   |  ap-southeast-1    ||
||  ec2.ap-southeast-2.amazonaws.com   |  ap-southeast-2    ||
|+-------------------------------------+--------------------+|
```

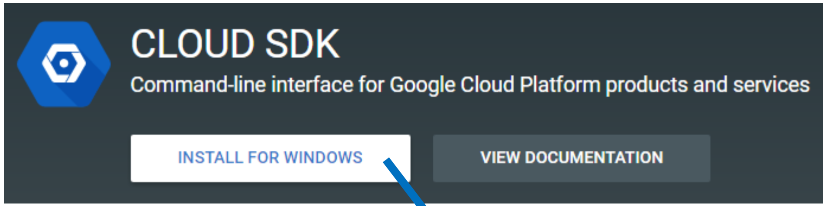
amazon web services

IBM Bluemix™



Bluemix CLI

(Current Version: 0.4.2)



CLOUD SDK

Command-line interface for Google Cloud Platform products and services

Google Cloud Platform

Deploying

TERRAFORM

- Cross-cloud deployment and configuration
- Different ‘provider’ options (Azure, AWS, OpenStack, Google Cloud, SoftLayer)

- AWS orchestration framework
- Configure networks, VMs, storage etc.

CloudFormation

heat
orchestrating multiple cloud
applications

- Define resources, security, storage etc.
- Used to orchestrate OpenStack deployments

Deploying

CloudFormation

heat
orchestrating multiple cloud applications

- All are intended to let you define templates – complete sets of resources that implement your solution
- Designed to give you one-click deployment
- As importantly, they give you one-click tear-down
  - If you want to use clouds to reduce IT expense, this is as/more important than one-click deployment
  - Spin up/tear down short-live deployments (ideal for dev and test)
- Use cloud-specific APIs under the covers

Agenda

- Recap of cloud options
- Overview of IBM messaging solutions
- Running MQ in different cloud architectures
  - Deploying and Installation
  - Persisting data ←
  - Availability
  - Monitoring and metrics
  - Security
  - Service discovery
  - Cloud ingress/egress

Persistent Storage

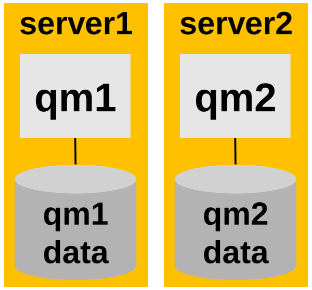
- **Reliability of storage**
  - Replicated across failure domains / availability zones?
  - Are disk writes cached?
  - What's the failure rate of disks?

- **Connecting to the right persistent storage**
  - When a queue manager's compute resource is moved (e.g. run a container in a different VM), then something needs to connect the queue manager to the correct storage.
  - e.g. the correct block storage volume, or directory on networked file storage.

- **Identifying the right persistent storage**
  - Some cloud orchestrators will run identical instances of your image. This could lead to lots of copies of "qm1".

Local Storage

- **Local storage typically has the same life span as compute resource (e.g. VM or bare metal server)**



- **Often very fast to access local storage**
  - SSDs

- **Containers typically have a short life span, usually making local storage an unsuitable option for MQ**

- **This may be an option for long-lived bare metal servers**

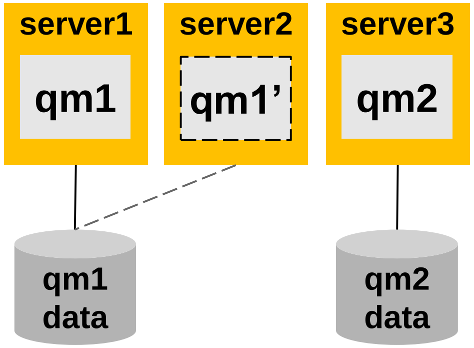
Networked Block Storage

- For example: OpenStack Cinder, Amazon Elastic Block Storage, Ceph, DRBD

- You can use well-tested filesystems

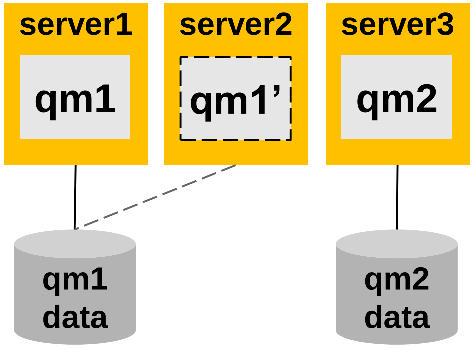
- Performance needs to be considered

- Key challenge: something needs to re-attach the block storage to a different VM if the queue manager is moved (e.g. because of failure, or VM image update)



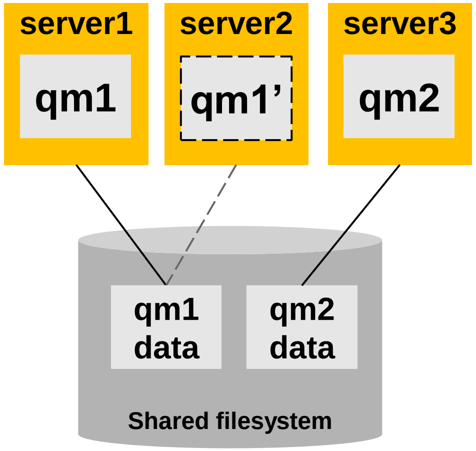
Networked Block Storage

- Can be used to implement MQ HA and/or DR
- Block storage under /var/mqm synchronised to an alternative availability zone
- Something must monitor the state of the primary queue manager…
  - E.g. Pacemaker
- …and then mount the block storage at the standby availability zone before starting the backup queue manager
- DR achieved by synchronising to an alternative region
  - Amount of MQ data eligible to be lost controlled by DRBD

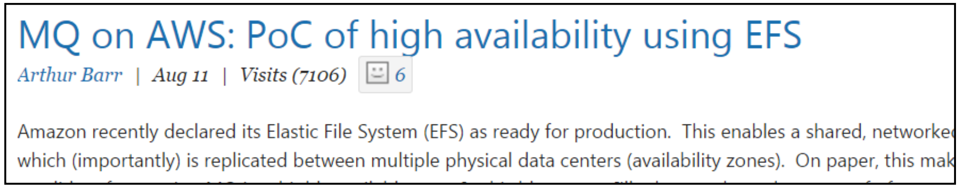


Networked Filesystem

- **Examples: GPFS, NFS, Amazon EFS**
- **Key challenge: MQ is sensitive to filesystem characteristics such as locking**
  - For example, NFS V3 is known not to work
  - IBM publishes a list of filesystems tested with MQ: http://www-01.ibm.com/support/docview.wss?uid=swg21433474
- **Not just for multi-instance queue managers – can easily handle the case where a queue manager is moved.**
- **Performance needs to be considered**



MQdev article discusses MQ on EFS



MQ on AWS: PoC of high availability using EFS

Arthur Barr | *Aug 11* | *Visits (7106)* 6

Amazon recently declared its Elastic File System (EFS) as ready for production. This enables a shared, networke which (importantly) is replicated between multiple physical data centers (availability zones). On paper, this mak

Agenda

- Recap of cloud options
- Overview of IBM messaging solutions
- Running MQ in different cloud architectures
  - Deploying and Installation
  - Persisting data
  - Availability ←
  - Monitoring and metrics
  - Security
  - Load balancing and service discovery
  - Cloud ingress/egress

Availability

What level of availability do you need?

(As an example, AWS commit to 99.95% uptime in an availability zone)

- Single instance with failure detection and automatic restart
  - Data safe but only available to new instance in availability zone
- Active-Passive, warm instance waiting in another availability zone to take over in event of failure
  - MQ multi-instance (using networked file systems or synchronized block storage)
- Active-Active, workload balancing between instances in multiple availability zones
  - MQ clustering
  - Single instances, use cloud-technologies to workload balance

Availability – Pros and Cons

Single instance, automatic restart

- Higher performance – no or local-only data synchronisation ✓
- Persistent storage re-mounted to new instance in same availability zone ✓
- Simple architecture ✓
- Outage time while instance restarted ✘

Active-Passive (MQ Multi Instance)

- Network file share or replicated block storage – performance cost ✘
- Data already synchronized to an alternative availability zone ✓
- Shorter outage while standby QM restarts ✓

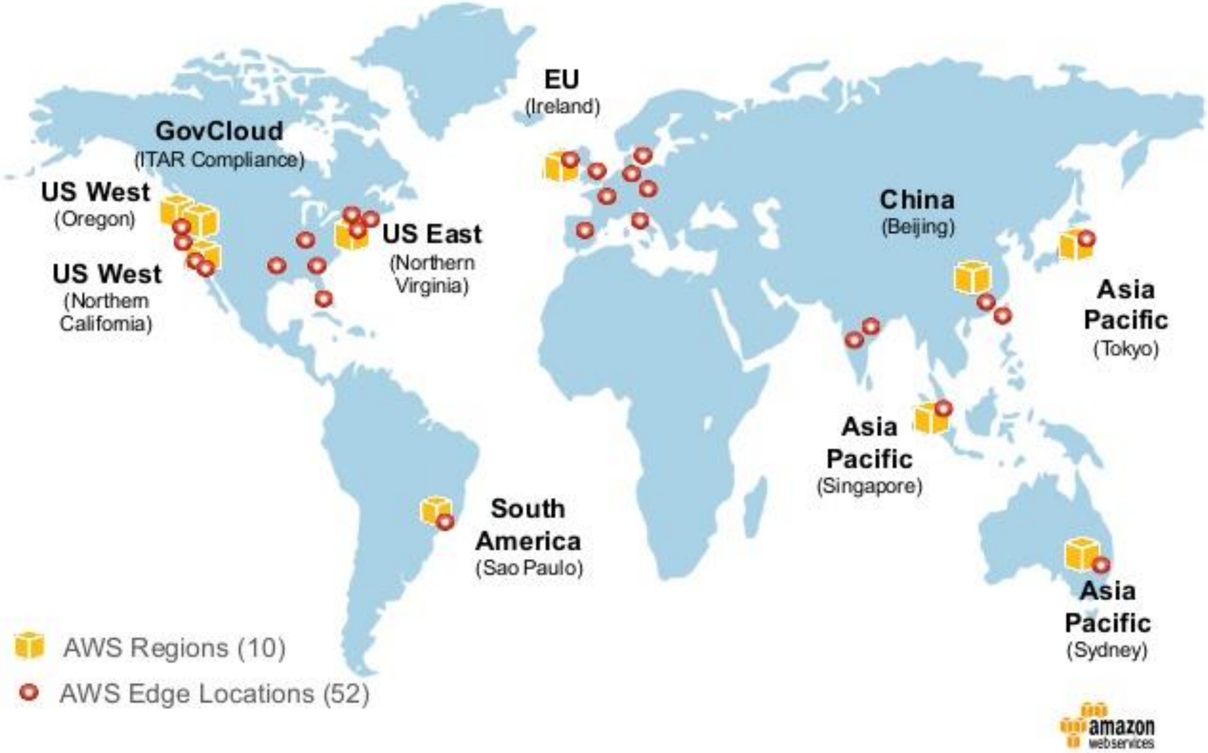
Active-Active (MQ Clustering)

- Fine grain control of workload balancing across queue managers ✓
- No downtime except for messages stuck in transit ✓
- Still require a strategy for restarting each separate instance ✘

Availability

Getting the data there is one thing – is it allowed to be there?

Are there legal requirements on where your commercial and private data is stored or travels through?



Agenda

- Recap of cloud options
- Overview of IBM messaging solutions
- Running MQ in different cloud architectures
  - Persisting data
  - Availability
  - Log management ←
  - Monitoring and metrics
  - Security
  - Service discovery
  - Cloud ingress/egress

Log Management

- **To manage large numbers of servers, you don’t want to SSH into them very often (if ever).**

- **You will still need to diagnose problems**

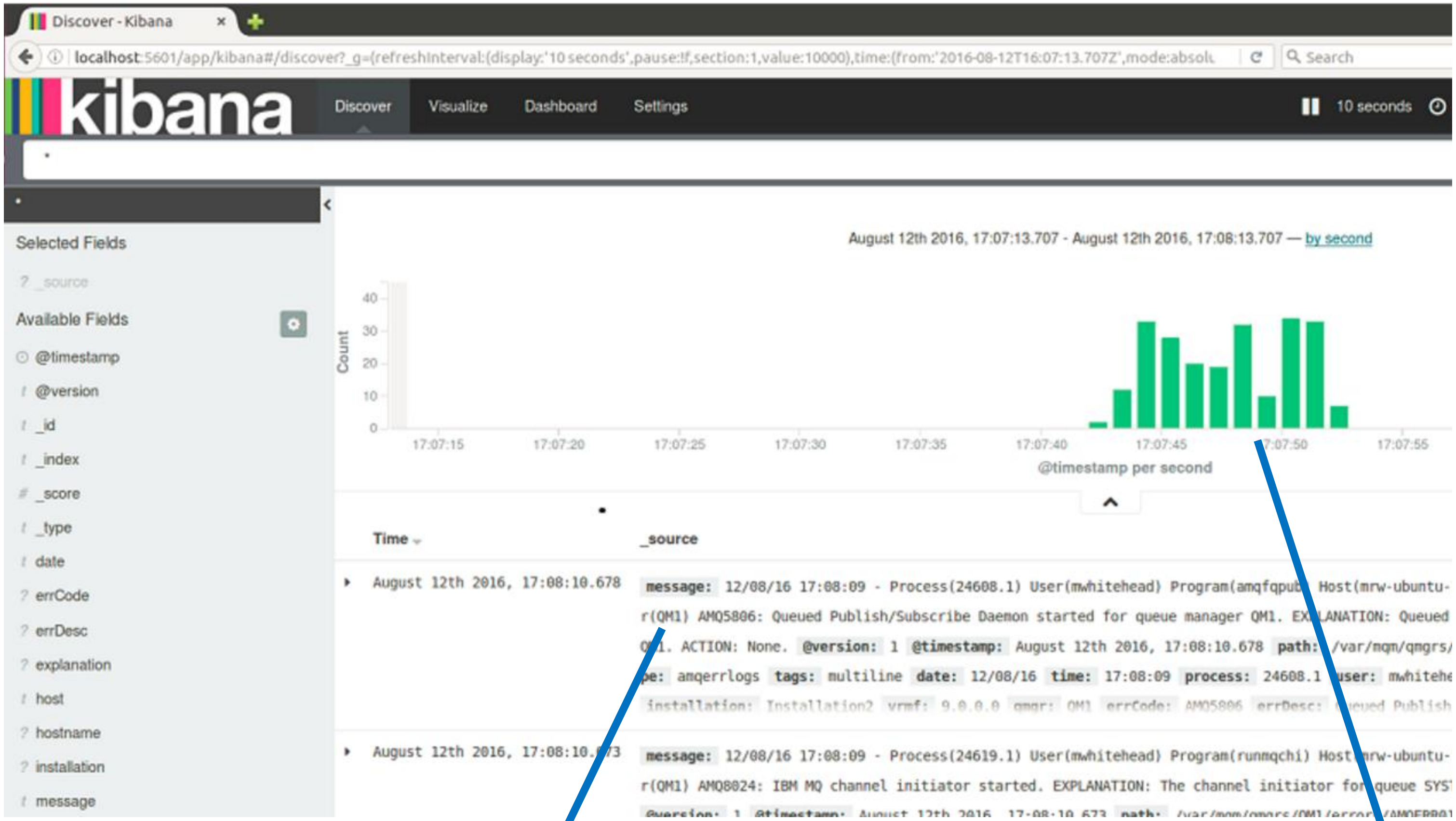
- **Centralized logging is commonly used, where an agent sends MQ and system logs to a centralized location**
  - Store
  - Index to make searchable
  - Analyze

- **For example:**
  - IBM Logmet
  - AWS Cloudwatch
  - ElasticSearch

Capturing error logs

- **In the event of a failure it can be important to gather additional diagnostics**
  - FFDCs
  - Trace
- **If you use local storage and the container or VM unexpectedly disappears you may not have access to diagnostic material**
- **If you have used networked storage do you have a way of spinning up a VM just to gather logs files?**
- **Pushing logs to a remote system (ElasticSearch, Logstash, Logmet, Cloudwatch etc.) might help separate error logs from QM runtimes, but…**
  - You may find not everything made it off-box before the VM terminated (Logstash sends every 30 seconds by default)
  - Requires scraping the file system by reading and parsing AMQERR0x.LOG

Remote Logging – Elasticsearch, Kibana, Logmet



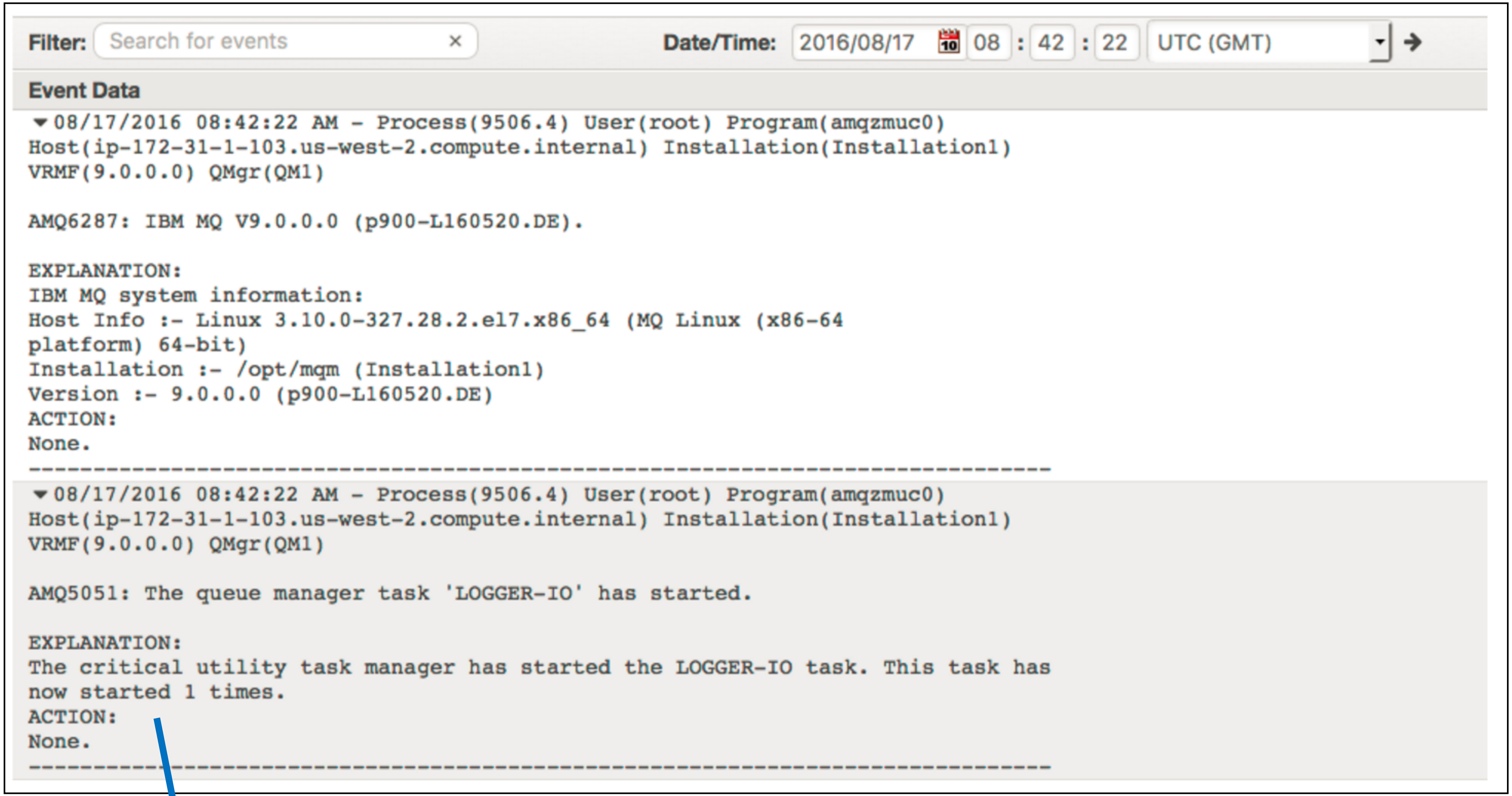
```
August 12th 2016, 17:08:10.678   message: 12/08/16 17:08:09 - Process(24608.1) User(mwhitehead) Program(amqfqpub) Host(mrw-ubuntu-
                                 r(QM1) AMQ5806: Queued Publish/Subscribe Daemon started for queue manager QM1. EXPLANATION: Queued
                                 QM1. ACTION: None. @version: 1 @timestamp: August 12th 2016, 17:08:10.678 path: /var/mqm/qmgrs/
                                 pe: amqerrlogs tags: multiline date: 12/08/16 time: 17:08:09 process: 24608.1 user: mwhitehe
                                 installation: Installation2 vrmf: 9.0.0.0 qmgr: QM1 errCode: AMQ5806 errDesc: Queued Publish

August 12th 2016, 17:08:10.673   message: 12/08/16 17:08:09 - Process(24619.1) User(mwhitehead) Program(runmqchi) Host(mrw-ubuntu-
                                 r(QM1) AMQ8024: IBM MQ channel initiator started. EXPLANATION: The channel initiator for queue SYS
                                 @version: 1 @timestamp: August 12th 2016, 17:08:10.673 path: /var/mqm/qmgrs/QM1/errors/AMQERR01
```

Graphs and charts of log stats

View and drill down into each individual AMQERR0x.LOG entry

Remote Logging – AWS Cloudwatch


Filter: Search for events ×
Date/Time: 2016/08/17 08 : 42 : 22 UTC (GMT)
Event Data

```
▾08/17/2016 08:42:22 AM - Process(9506.4) User(root) Program(amqzmuc0)
Host(ip-172-31-1-103.us-west-2.compute.internal) Installation(Installation1)
VRMF(9.0.0.0) QMgr(QM1)

AMQ6287: IBM MQ V9.0.0.0 (p900-L160520.DE).

EXPLANATION:
IBM MQ system information:
Host Info :- Linux 3.10.0-327.28.2.el7.x86_64 (MQ Linux (x86-64
platform) 64-bit)
Installation :- /opt/mqm (Installation1)
Version :- 9.0.0.0 (p900-L160520.DE)
ACTION:
None.
-------------------------------------------------------------------------------
▾08/17/2016 08:42:22 AM - Process(9506.4) User(root) Program(amqzmuc0)
Host(ip-172-31-1-103.us-west-2.compute.internal) Installation(Installation1)
VRMF(9.0.0.0) QMgr(QM1)

AMQ5051: The queue manager task 'LOGGER-IO' has started.

EXPLANATION:
The critical utility task manager has started the LOGGER-IO task. This task has
now started 1 times.
ACTION:
None.
-------------------------------------------------------------------------------
```

Similarly to Kibana and Elasticsearch you can apply filters and drill down into individual AMQERR0x.LOG entries

Agenda

- Recap of cloud options
- Overview of IBM messaging solutions
- Running MQ in different cloud architectures
  - Persisting data
  - Availability
  - Log management
  - Metrics and monitoring ⟵
  - Security
  - Load balancing and service discovery
  - Cloud ingress/egress

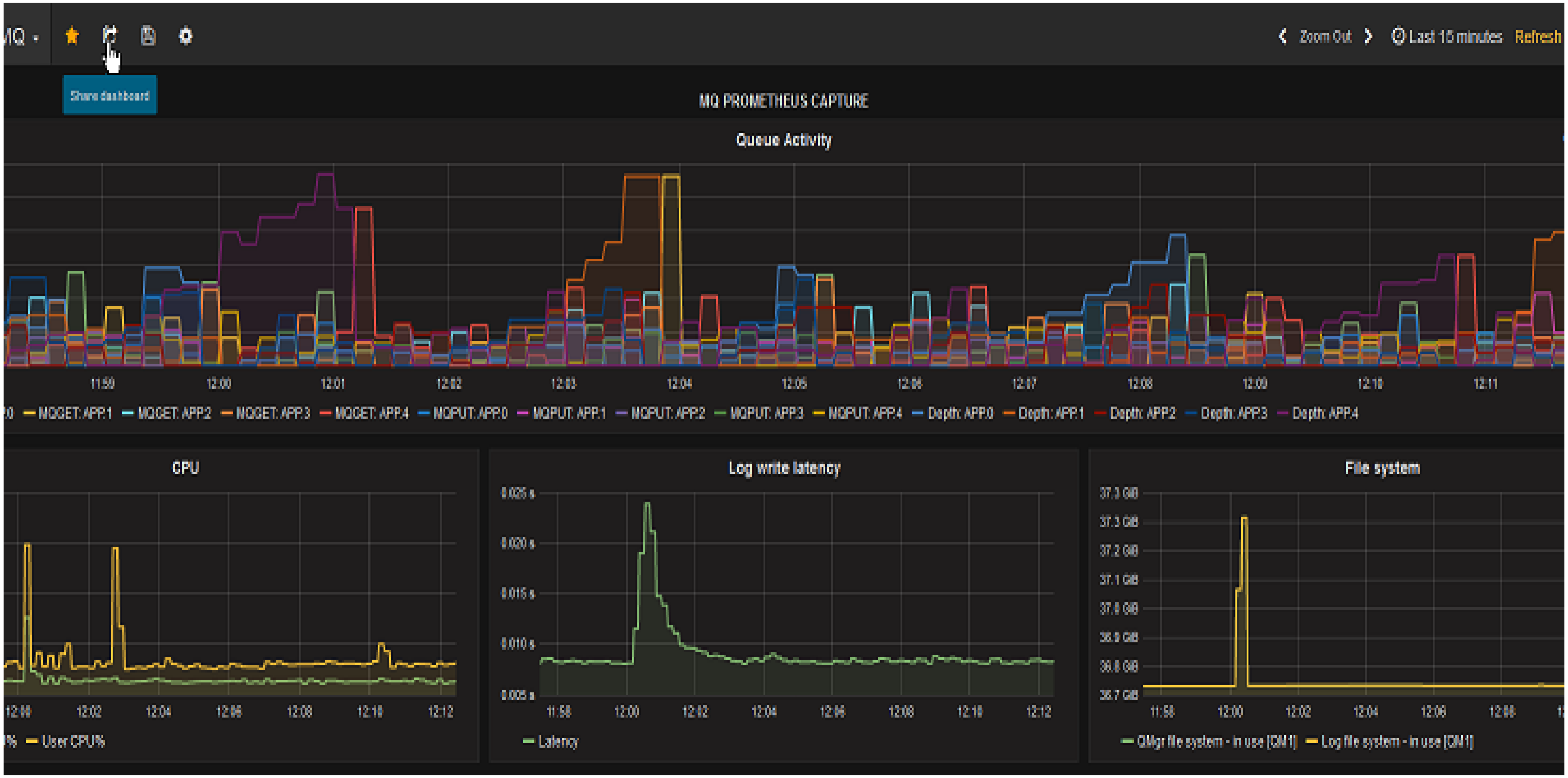
Metrics and Monitoring

- **MQ V9 makes many statistics available through a pub/sub interface**

- **Option to remotely subscribe to topics under $SYS/MQ for information on:**
  - CPU usage
  - Disk usage
  - Connections and disconnections
  - Opening and closing of queues
  - Pub/sub and put/get
  - Syncpoint calls
  - Changes to MQ objects (MQSET and MQINQ)

- **Publish to remote metrics servers e.g. Graphite, Prometheus**
  - Visualise using Grafana

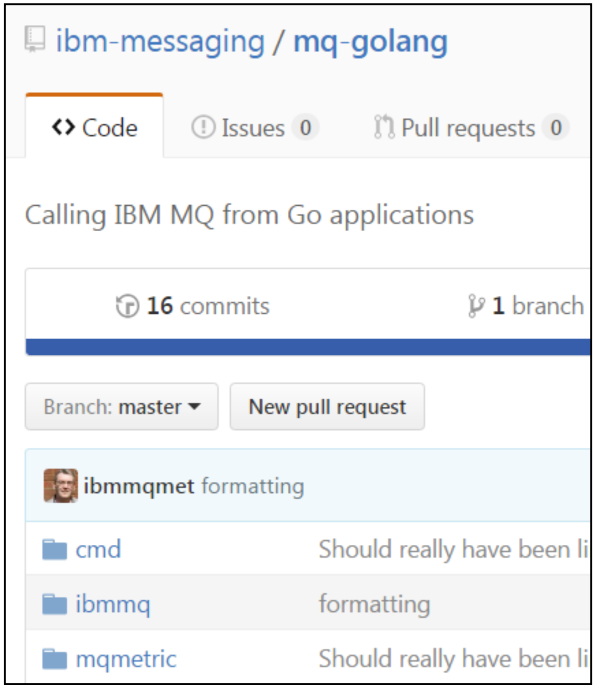
Centrally monitoring metrics



Grafana can by used as the dashboard, connecting to a back-end time series database

Mark Taylor has an MQ sample available on GitHub

- **Written in Go**
- **Subscribes to MQ9 $SYS topics and extracts various MQ and system metrics**
- **Can be used to push data to various logging servers or databases. MQdev blog entries demonstrate:**
  - Prometheus
  - Graphic
  - Logmet
  - AWS Cloudwatch
  - InfluxDB
- **Could also output to something more generic such as collectd**



ibm-messaging / mq-golang

Calling IBM MQ from Go applications

16 commits
1 branch

ibmmqmet formatting

cmd	Should really have been li
ibmmq	formatting
mqmetric	Should really have been li

Agenda

- Recap of cloud options
- Overview of IBM messaging solutions
- Running MQ in different cloud architectures
  - Persisting data
  - Availability
  - Log management
  - Metrics and monitoring
  - Security ⟵
  - Service discovery
  - Cloud ingress/egress

Security

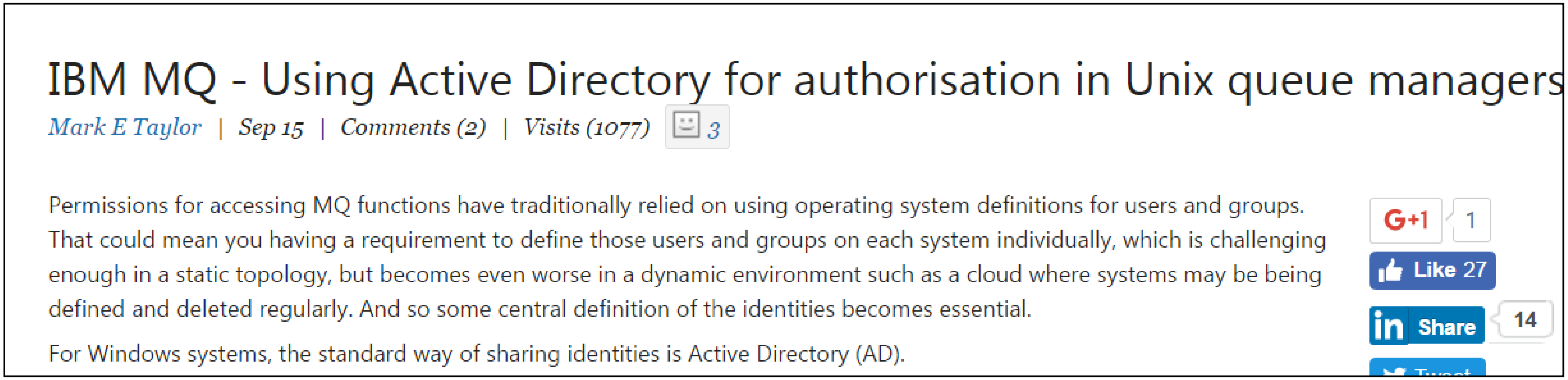
- **For the most part, once you've made the choice to go with a cloud provider, the security considerations are similar to on-premise security**

- **Major differences to traditional on-premises security management:**
  - Key distribution can be more challenging
    - Need to automate everything
  - Identity used for authorization needs to be dynamically generated
  - Increasingly unwise to only secure at the edge of the network
    - Which was a bad practice before

- **Twenty of the greatest myths of cloud security: http://www.cio.com/article/2922374/cloud-security/20-of-the-greatest-myths-of-cloud-security.html**

Security

- Connecting with cloud user directories may be necessary, especially where are identities are auto-generated
- MQ V8 support for LDAP (V9 for Windows) makes it possible to connect to Active Directory servers
- Blog on MQdev describes a setup using the AWS Directory Service
- Unix queue managers connect to AWS Directory Service for authentication/authorization



IBM MQ - Using Active Directory for authorisation in Unix queue managers

Mark E Taylor | *Sep 15* | *Comments (2)* | *Visits (1077)* 3

Permissions for accessing MQ functions have traditionally relied on using operating system definitions for users and groups. That could mean you having a requirement to define those users and groups on each system individually, which is challenging enough in a static topology, but becomes even worse in a dynamic environment such as a cloud where systems may be being defined and deleted regularly. And so some central definition of the identities becomes essential.

For Windows systems, the standard way of sharing identities is Active Directory (AD).

Security for Developers

- **Developing securely might require VPNs or secure tunnels**
- **All cloud providers off some level of VPN connectivity**
  - Typically assume some level or enterprise/appliance VPN support in the DMZ
  - Does your enterprise support the same standards?
  - Extra hoops to jump through?
- **Some offer software VPN support**
  - Well suited to developers who want to quickly connect to their cloud network
  - Vital if you don’t want to expose development servers on publicly facing IP address
  - Some have limitations (Bluemix VPN currently only offers ‘dial-you’ mode)
  - Some have more obscure requirements
    - Microsoft Azure requires you to be using PowerShell, not the Azure CLI

Agenda

- Recap of cloud options
- Overview of IBM messaging solutions
- Running MQ in different cloud architectures
  - Persisting data
  - Availability
  - Log management
  - Metrics and monitoring
  - Security
  - Service discovery ←
  - Cloud ingress/egress

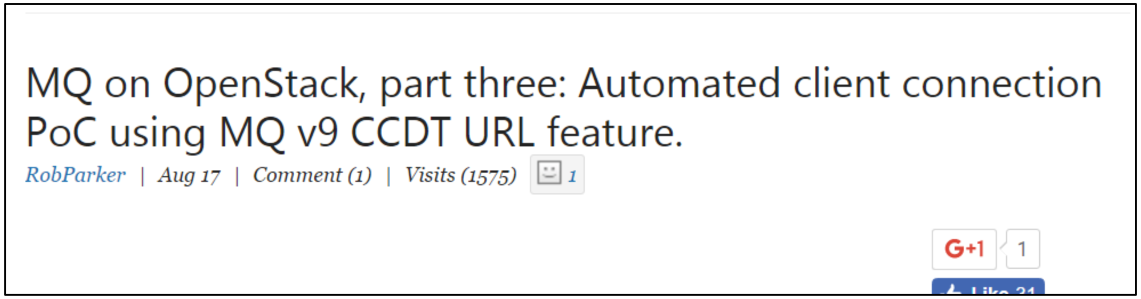
Service Discovery

- **Queue manager IP addresses more likely to change in a dynamic/cloud environment**
  - e.g. new container using same queue manager data will have a different IP address
- **Most cloud providers offer some static management of IP addresses**
  - Amazon Elastic IP
  - Azure Reserved IP
  - Google Cloud Engine Reserved IP
- **DNS available in most clouds**
  - Amazon Route 53
  - Azure DNS
  - Google Cloud DNS
- **but – everything comes at a cost**

Service Discovery

- **MQ channels need to know where they're connecting to**
- **MQ clusters instances tell each other about their own IP addresses**
  - Changing the clussdr conname isn't enough
  - The clusrcvr needs changing as well, and that needs to be distributed around the cluster
- **Client configuration – how do clients find the right queue manager**
  - MQ V9 CCDT URLs help centralize configuration
  - Requires an HTTP server to host the CCDTs
  - And a way of updating CCDTs when configuration changes

Blogs available on MQdev



MQ on OpenStack, part three: Automated client connection PoC using MQ v9 CCDT URL feature.

RobParker | Aug 17 | Comment (1) | Visits (1575) 1

Agenda

- Recap of cloud options
- Overview of IBM messaging solutions
- Running MQ in different cloud architectures
  - Persisting data
  - Availability
  - Log management
  - Metrics and monitoring
  - Security
  - Service discovery
  - Cloud ingress/egress ←

Cloud Ingress/Egress

- **What about data entering/leaving the cloud?**
- **What level of availability do you need between cloud and on-premise?**
- **How is connectivity secured?**
  - VPN, Secure Tunnel, TLS/TCP
- **Performance**
  - What data rates are required?
  - Which cloud region gives the best latency?
  - Is failover to an alternative region acceptable?
- **How is data workload balanced between cloud and on-prem?**

Cloud Ingress/Egress - DMZ

- **Traditional DMZ provides a separation of physical networks**
  - Plus decision making about what comes in and out

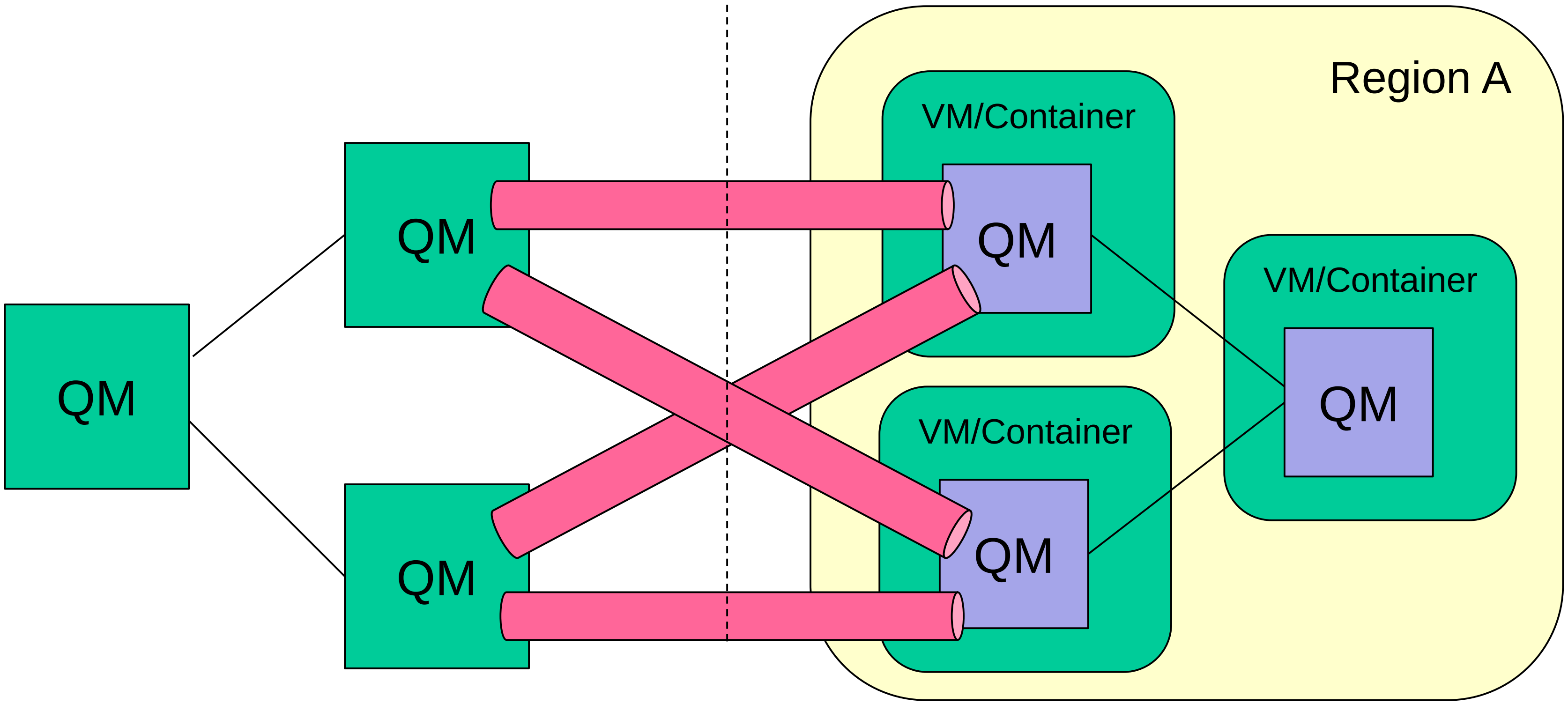
- **Cloud virtual network definitions provide similar segregation of networks**
  - Still necessary to police inbound and outbound traffic
    - Amazon Security Groups
    - Azure Network Security Group
    - Google Cloud Firewall

Cloud Ingress/Egress Slide – MQ Clusters

On-Premise

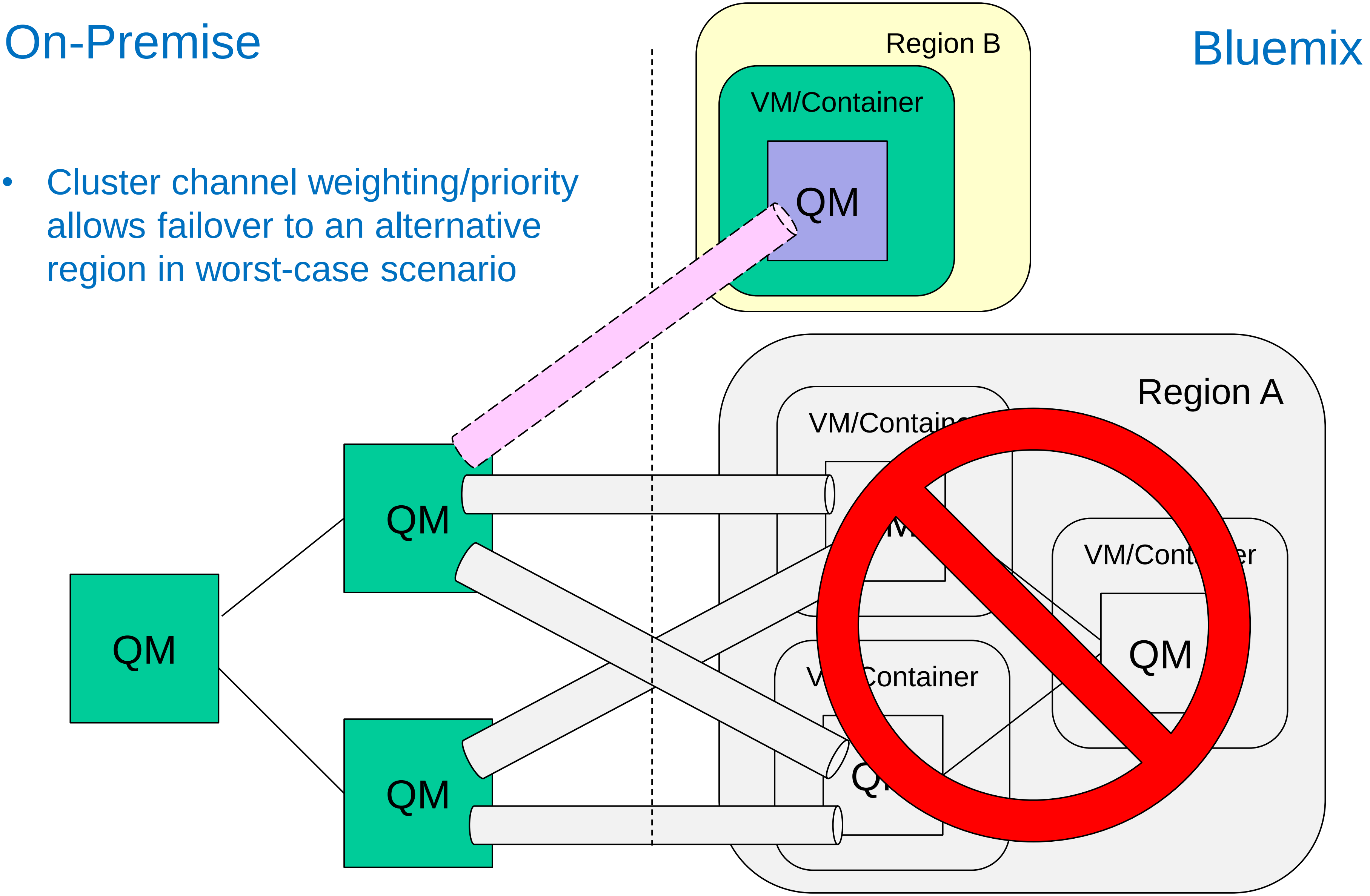
Bluemix

- MQ clustering well suited to active-active round-robin of data to/from the cloud

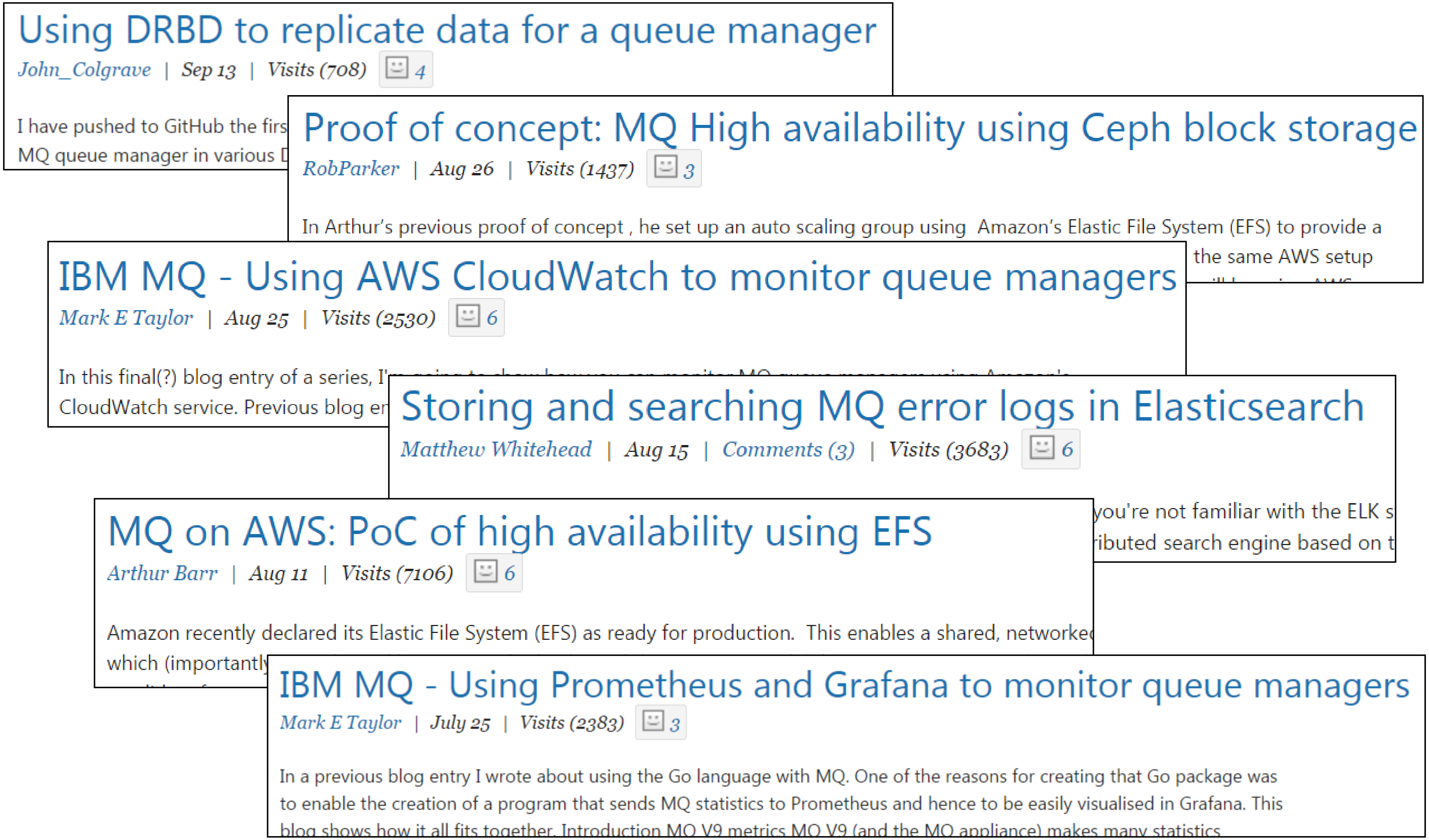


Cloud Ingress/Egress Slide – MQ Clusters

- Cluster channel weighting/priority allows failover to an alternative region in worst-case scenario



MQ Ecosystem Blogs – MQdev



Using DRBD to replicate data for a queue manager

John_Colgrave | Sep 13 | Visits (708) 4

I have pushed to GitHub the firs
MQ queue manager in various D

Proof of concept: MQ High availability using Ceph block storage

RobParker | Aug 26 | Visits (1437) 3

In Arthur's previous proof of concept , he set up an auto scaling group using Amazon's Elastic File System (EFS) to provide a
the same AWS setup

IBM MQ - Using AWS CloudWatch to monitor queue managers

Mark E Taylor | Aug 25 | Visits (2530) 6

In this final(?) blog entry of a series, I'
CloudWatch service. Previous blog en

Storing and searching MQ error logs in Elasticsearch

Matthew Whitehead | Aug 15 | Comments (3) | Visits (3683) 6

you're not familiar with the ELK s
ibuted search engine based on t

MQ on AWS: PoC of high availability using EFS

Arthur Barr | Aug 11 | Visits (7106) 6

Amazon recently declared its Elastic File System (EFS) as ready for production. This enables a shared, networked
which (importantly

IBM MQ - Using Prometheus and Grafana to monitor queue managers

Mark E Taylor | July 25 | Visits (2383) 3

In a previous blog entry I wrote about using the Go language with MQ. One of the reasons for creating that Go package was to enable the creation of a program that sends MQ statistics to Prometheus and hence to be easily visualised in Grafana. This blog shows how it all fits together. Introduction MQ V9 metrics MQ V9 (and the MQ appliance) makes many statistics

Thank You - Questions?

Related session:

Wednesday 8.30am – 9.40am
Deploying MQ in a Self-Service Way – David Ware

Please Note

IBM’s statements regarding its plans, directions, and intent are subject to change or withdrawal without notice at IBM’s sole discretion.

Information regarding potential future products is intended to outline our general product direction and it 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 our products remains at our sole discretion.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user’s job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Trademark Statement

- IBM and the IBM logo are trademarks of International Business Machines Corporation, registered in many jurisdictions. Other marks may be trademarks or registered trademarks of their respective owners.
- Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.
- Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.
- Red Hat Enterprise Linux is a registered trademark of Red Hat, Inc. in the United States and other countries.
- Ubuntu and Canonical are registered trademarks of Canonical Ltd.
- SUSE and SLES are registered trademarks of SUSE LLC in the United States and other countries
- Mac and OS X are trademarks of Apple Inc., registered in the U.S. and other countries
- Other company, product and service names may be trademarks, registered marks or service marks of their respective owners.
- References in this publication to IBM products and services do not imply that IBM intends to make them available in all countries in which IBM operates.