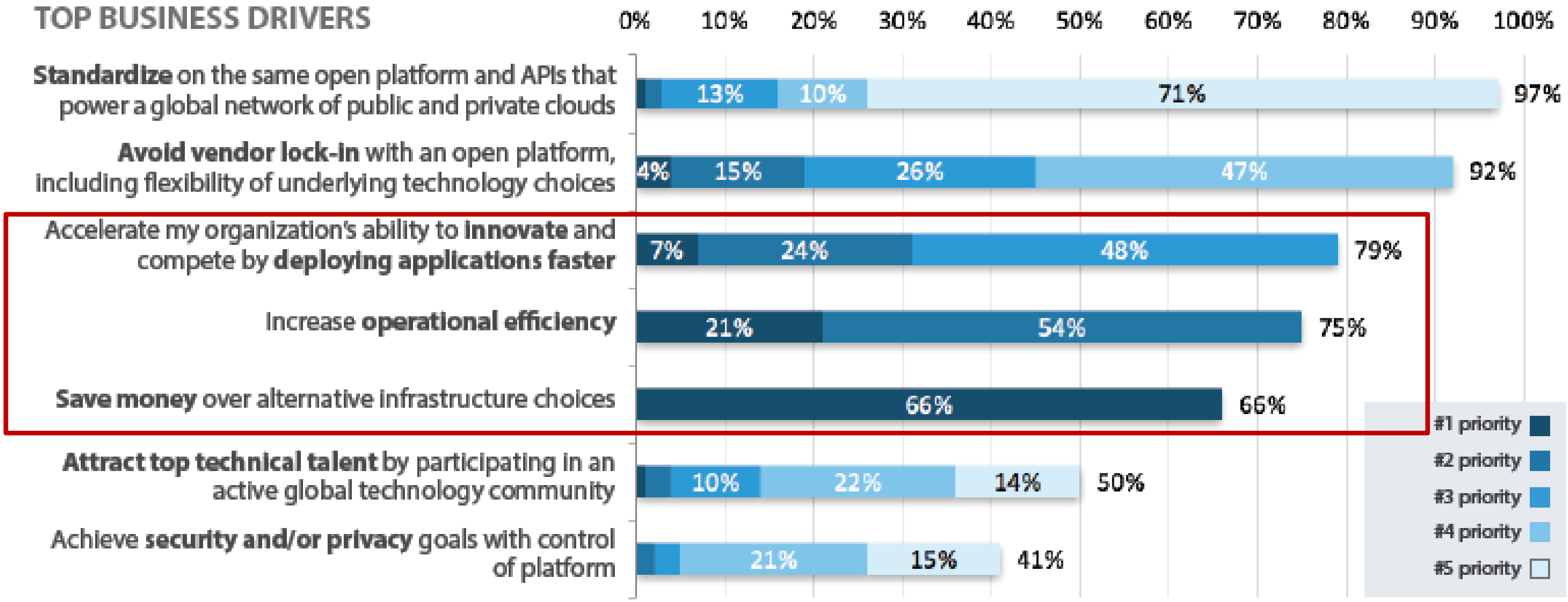
Designing MQ deployments for the cloud generation

WebSphere User Group, London

Arthur Barr, Senior Software Engineer, IBM MQ

30th March 2017

Top business drivers for cloud



Source: OpenStack user survey, April 2016

In other words…

“Deliver more with less”
...more often
...with less money
...with fewer people

Characteristics of a cloud environment

- **Self-service** – empowers users to provision resources without requiring human intervention, most likely using a web-based portal or an API.
- **Elastic scaling** – enables scaling up and down on demand, driving the need for high levels of automation.
- **Shared resources** – offers economies of scale through the use of shared infrastructure and software, securely separating the resources at a logical level.
- **Metered usage** – allows pay-as-you-go billing through monitoring, measurement and reporting of usage.

Types of cloud environments

Infrastructure as a Service (IaaS)

- IBM Bluemix Infrastructure
- Amazon Web Services
- Microsoft Azure
- OpenStack

Containers as a Service (CaaS)

- IBM Containers on Bluemix
- Amazon Elastic Container Service
- Microsoft Azure Container Service
- Kubernetes
- Docker Swarm
- Apache Mesos

Platform as a Service (PaaS)

- IBM Bluemix Cloud Foundry Runtimes
- Amazon Elastic Beanstalk
- Microsoft Azure App Service
- Cloud Foundry
- OpenShift

Functions as a Service (FaaS)

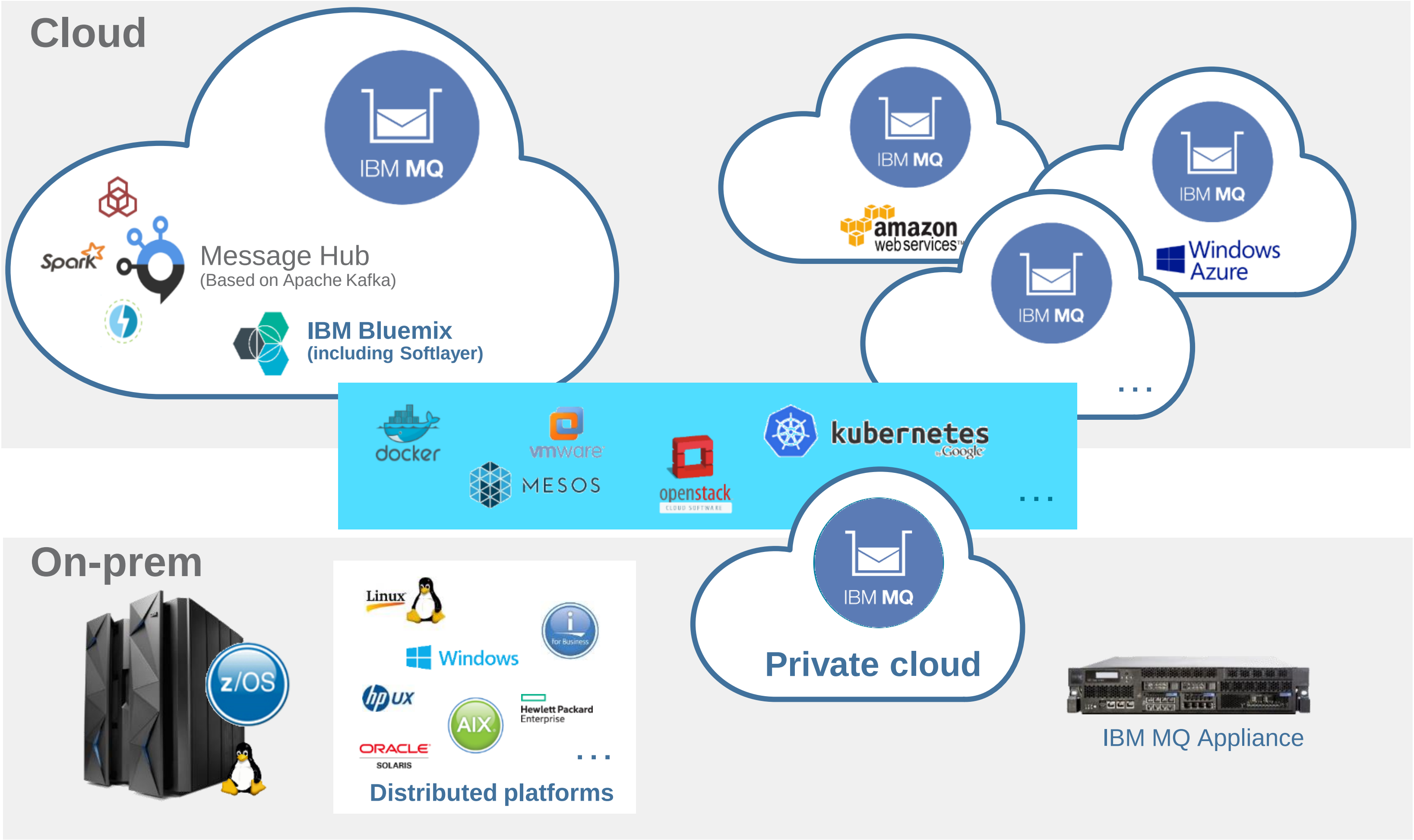
- IBM Bluemix OpenWhisk
- Amazon Lambda
- Microsoft Azure Functions

Software as a Service (SaaS)

- IBM Cloudant (NoSQL database)
- Salesforce (CRM platform)

MQ queue managers

MQ client applications



Cloud

IBM MQ

Message Hub
(Based on Apache Kafka)

IBM Bluemix
(including Softlayer)

IBM MQ

IBM MQ

IBM MQ

. . .

. . .

On-prem

IBM MQ

Private cloud

Distributed platforms

. . .

IBM MQ Appliance

Running MQ in the cloud

- Availability
  - Persistent storage
  - Fail-over
- Security
- Scalability
  - Service discovery
  - Load balancing
- Error log management
- Metrics and monitoring
- Client applications

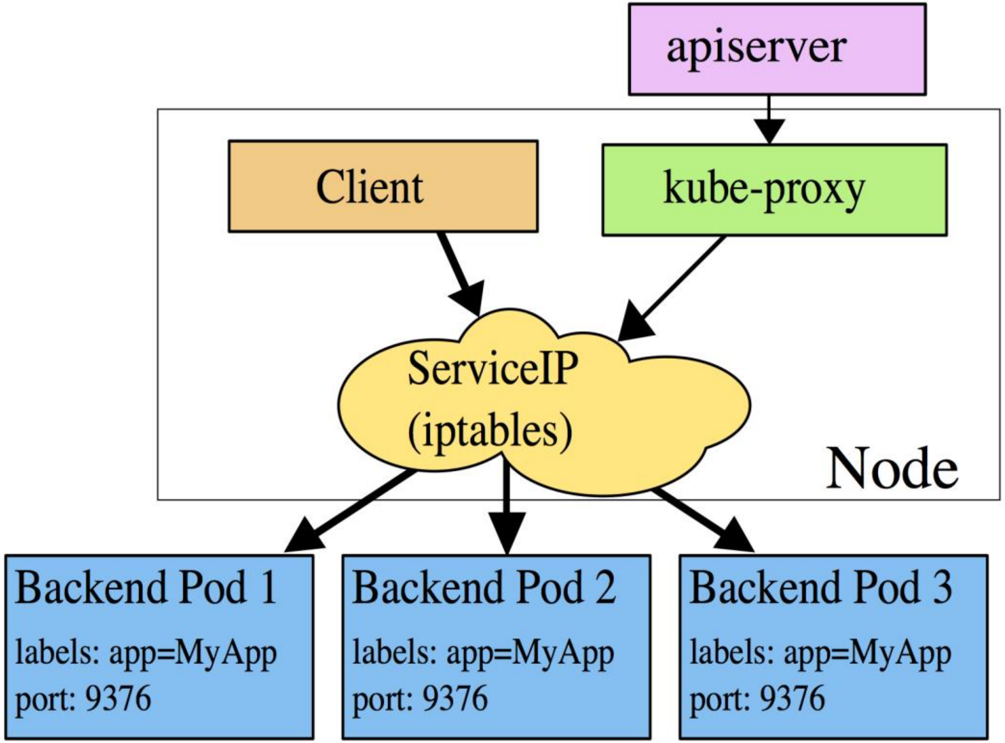
Re-think how you use MQ

Demo time

Kubernetes

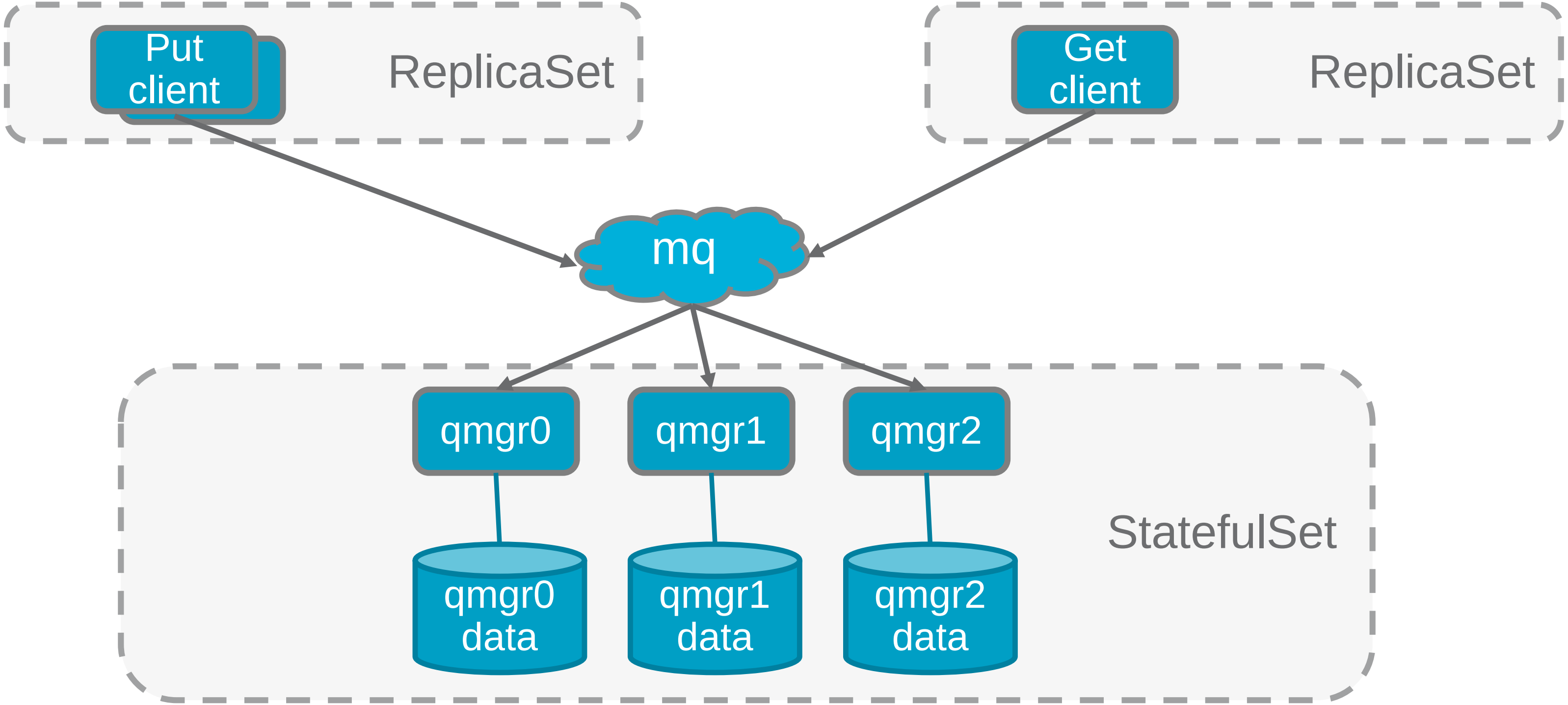
- **Kubernetes is an “open-source system for automating deployment, scaling, and management of containerized applications.”**
- Provides a set of cloud primitives at a higher level than an IaaS
  - Pod
  - ReplicaSet
  - PersistentVolume
  - StatefulSet
  - DaemonSet
  - Service
  - …

Kubernetes Services



Source: https://kubernetes.io/docs/user-guide/services/#proxy-mode-iptables

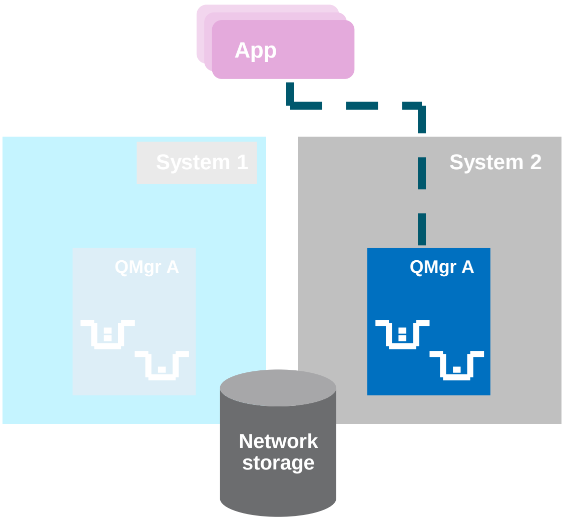
Kubernetes demo

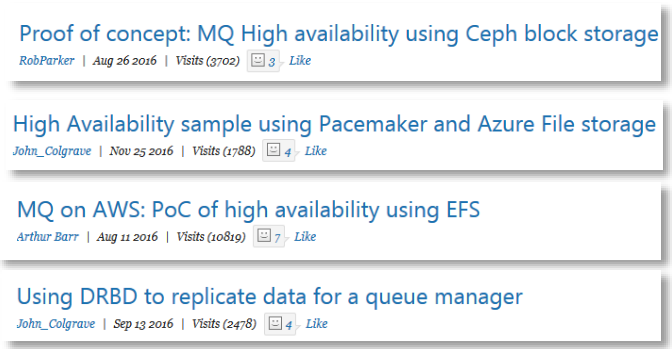


Things to consider

Availability

- Each message only exists in a single queue manager. Queue managers typically rely on **reliable persistent storage** to maintain message integrity.
- As long as the storage is reachable the queue manager runtime can run anywhere
- Storage should be replicated and queue managers must **automatically fail over** to remove any single point of failure




Proof of concept: MQ High availability using Ceph block storage
RobParker | Aug 26 2016 | Visits (3702) | 3 Like

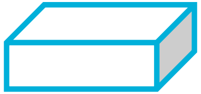
High Availability sample using Pacemaker and Azure File storage
John_Colgrave | Nov 25 2016 | Visits (1788) | 4 Like

MQ on AWS: PoC of high availability using EFS
Arthur Barr | Aug 11 2016 | Visits (10819) | 7 Like

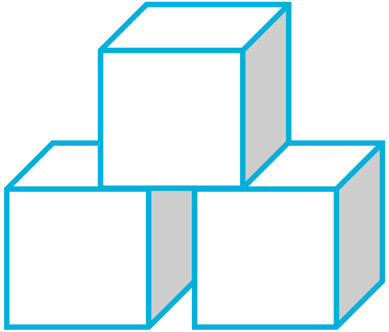
Using DRBD to replicate data for a queue manager
John_Colgrave | Sep 13 2016 | Visits (2478) | 4 Like

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

Persistent storage

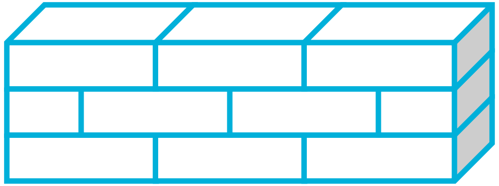


Local storage



Block storage

Bluemix Block Storage
Amazon EBS
OpenStack Cinder
Ceph RBD
DRBD



File storage

NFS V4
Bluemix File Storage
Amazon EFS
OpenStack Manila
CephFS
GlusterFS

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 is moved (e.g. run a container in a different VM), then something needs to re-connect the queue manager to the correct storage.
  - Some cloud orchestrators will run identical instances of your image. This could lead to lots of copies of “qm1”. Other orchestrators, like Kubernetes, allow you to manage separate identities (“qm1”, “qm2”, etc.).

Security

- No “armadillo” security

- All data in motion and at rest needs to be secured
  - Configure MQ channels with TLS
  - Use channel authentication for access control
  - Configure disk encryption either at the OS or cloud provider level.

  - IBM MQ Advanced Message Security is also available to provide additional security protections, such as per-message encryption.

Scalability

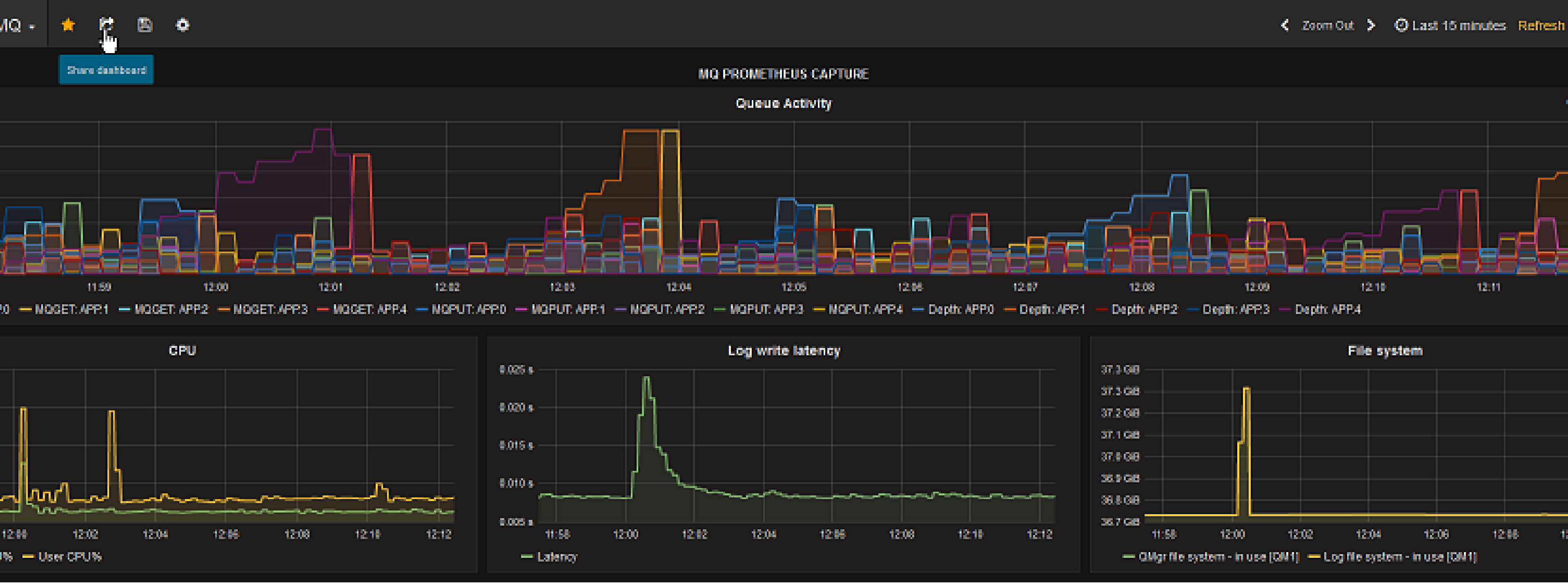
- In general, scale clients separately from servers
- Scaling up is easy
  - MQ cluster
  - Load-balanced set of identical queue managers
- Scaling down depends more on your applications
  - Need to remove a queue manager in a controlled manner making sure that all of the messages are safely processed.
- Message ordering
  - Scaling out rather than scaling up, brings concurrent processing
  - Messages could be received out of sequence.
  - MQ provides features to allow groups of messages to be handled in small ordered batches, or you can manage this yourself in your application.

Error 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 error logs and system logs to a centralized location
  - Store
  - Index to make searchable
  - Analyze
- For example:
  - IBM Monitoring & Analytics
  - AWS Cloudwatch
  - ElasticSearch

Metrics and monitoring

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



MQ metrics in Grafana

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

Client applications

- Cloud typically emphasizes *resilience* over *robustness*
- De-couple applications from your queue manager
- Reminder: use your MQ client’s auto-reconnect feature
  - Cloud servers are likely to get restarted or moved

```
factory.setClientReconnectOptions(WMQConstants.WMQ_CLIENT_RECONNECT);
```

- Service discovery will probably be affected
  - Dynamic IP addresses
  - Scalability may mean multiple equivalent queue managers are available

https://developer.ibm.com/messaging/mq-on-cloud/