# *Connecting MQ to the World: New Bridges*

**Mark Taylor**
*marke_taylor@uk.ibm.com*
**IBM Hursley**

## Introduction

- MQ has always included links to other application technologies
  - Not just a set of APIs
  - Not always devolved to MessageBroker/IIB

- For example
  - CICS, IMS Bridges
  - SAP R/3 Link (obsolete)
  - Resource Adapter for JEE
  - AMQP

- Now has bridges to two new environments:
  - **Salesforce**
  - **Blockchain**

## Objectives

- Provide simple way for MQ apps to integrate with these new systems

- Make use of existing MQ infrastructure

- Build on MQ skills

- Complement capabilities in other products such as IIB

# IBM MQ
# BRIDGE TO SALESFORCE

## IBM and Salesforce partnership

IBM and Salesforce
announce a new landmark
partnership

Read the press release

IBM | salesforce

Accelerate decision making and
drive greater customer success

IBM and Salesforce will deliver joint solutions designed to leverage artificial
intelligence and enable companies to make smarter decisions, faster than ever
before. The partnership will bring insights from Watson directly into the Salesforce
Intelligent Customer Success Platform, combining deep customer insights from
Salesforce Einstein with Watson's structured and unstructured data across many
industries including healthcare, financial services, retail and weather.

Sign up for our webinar to learn how you can bring IBM and Salesforce to your
organization.

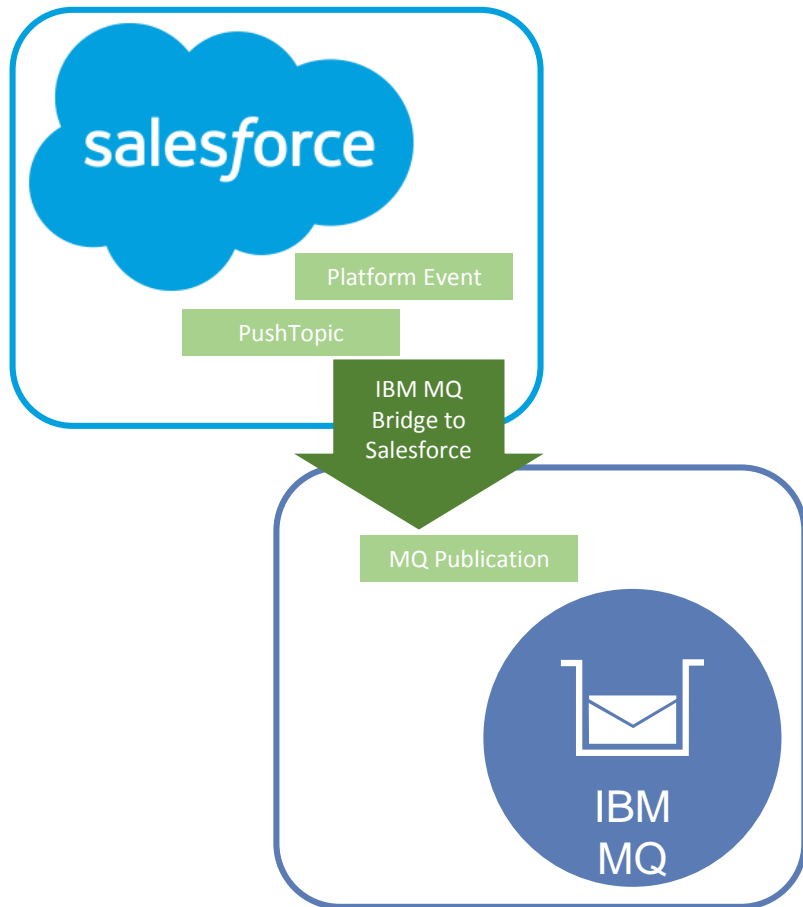Register for webinar

# MQ Bridge to Salesforce

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

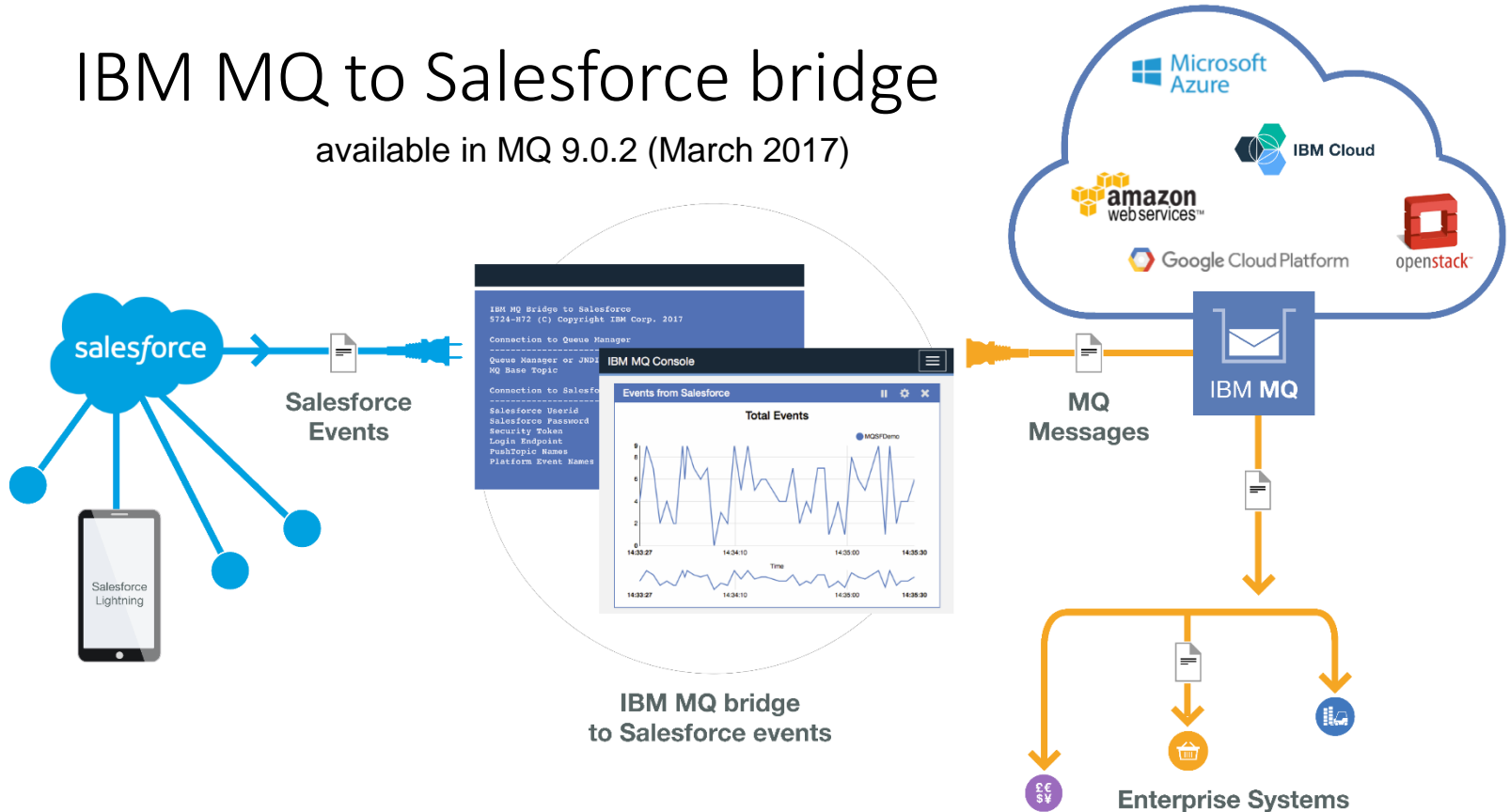Supports Salesforce Platform Events and PushTopics

Events are published into the MQ network

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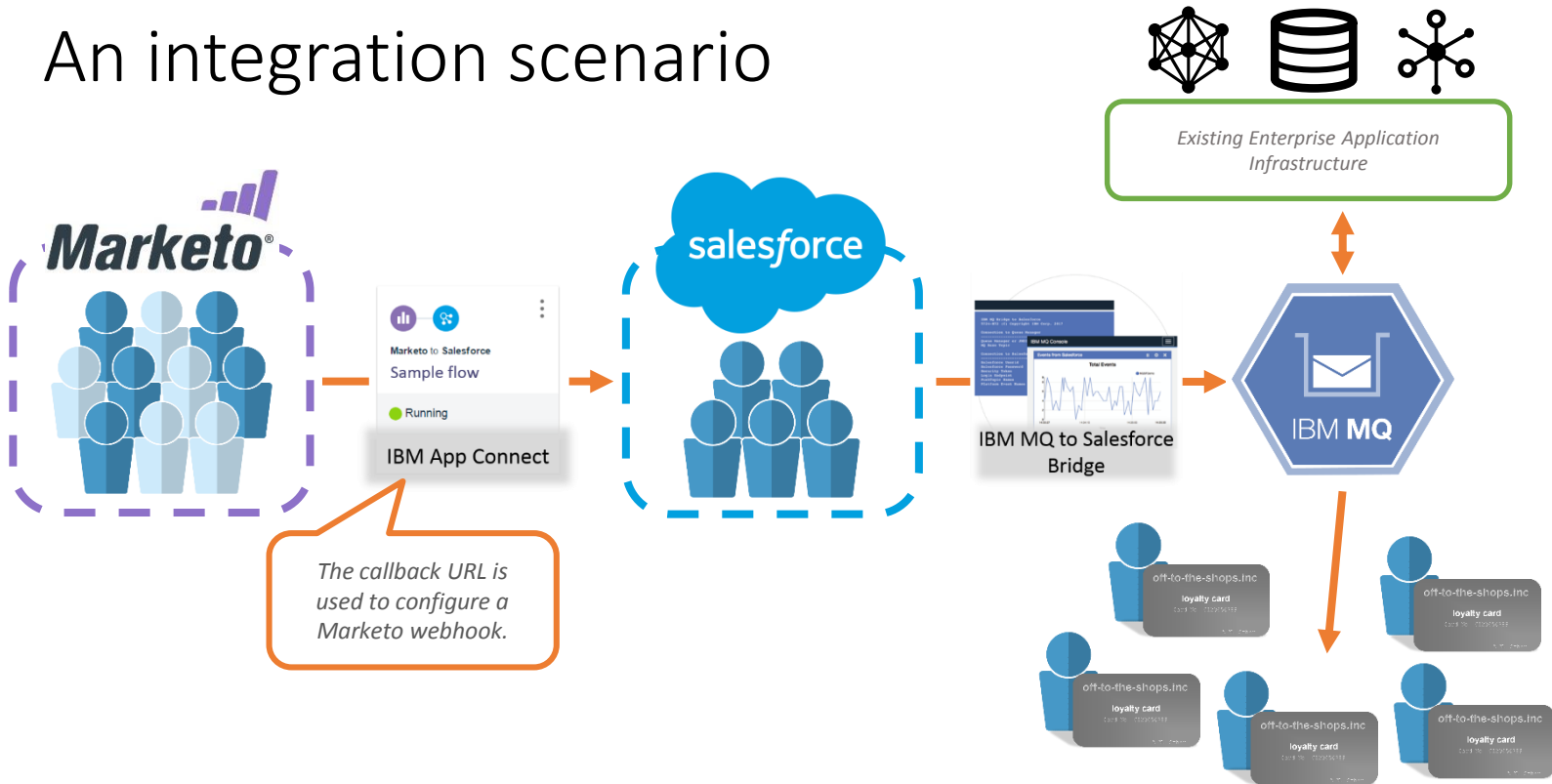
salesforce

Platform Event

PushTopic

IBM MQ Bridge to Salesforce

MQ Publication

IBM MQ

# IBM MQ to Salesforce bridge

available in MQ 9.0.2 (March 2017)



**Salesforce Events**

**IBM MQ bridge to Salesforce events**

**MQ Messages**

**Enterprise Systems**

IBM MQ Bridge to Salesforce
5724-H72 (C) Copyright IBM Corp. 2017

Connection to Queue Manager

Queue Manager or JNDI
MQ Base Topic

Connection to Salesforce

Salesforce Userid
Salesforce Password
Security Token
Login Endpoint
PushTopic Names
Platform Event Names

IBM MQ Console

Events from Salesforce

Total Events

MQSFDemo

Time

Salesforce Lightning

Microsoft Azure

IBM Cloud

amazon web services™

Google Cloud Platform

openstack™

IBM MQ

# An integration scenario



Existing Enterprise Application Infrastructure

**Marketo**

IBM App Connect
Marketo to Salesforce
Sample flow
● Running

*The callback URL is used to configure a Marketo webhook.*

salesforce

IBM MQ to Salesforce Bridge

IBM MQ

off-to-the-shops.inc
loyalty card

# DETAILS

## What are PushTopics and Platform Events

- PushTopics are "implicit" events generated by changes to Salesforce database tables
- Caused by a SOQL (SQL-ish) statement
- Whenever a table is changed, the SOQL statement is executed to see if it is satisified
  - If so, PushTopic is created containing requested fields

- Platform Events are "explicit" events, created by user-written application code
- New capability in Salesforce since March 2017
- Administrator defines the name of the event, and the fields it should contain

- Both are made available via a pub/sub API and protocol to any application

# TABLE: "CASE"

| ID | NAME | STATUS | SUBJECT |
|---|---|---|---|
| AAAAAA | Customer1 | Normal | Paper jam |

# TABLE: "CASE"

| ID | NAME | STATUS | SUBJECT |
|---|---|---|---|
| AAAAAA | Customer1 | Normal | Paper jam |
| AAABBB | Customer2 | Normal | Phone won't stop ringing |

# TABLE: "CASE"

| ID | NAME | STATUS | SUBJECT |
|---|---|---|---|
| AAAAAA | Customer1 | Normal | Paper jam |
| AAABBB | Customer2 | Normal | Phone won't stop ringing |
| BBBCCC | Customer3 | Escalated | Printer on fire |

**GENERATE PUSHTOPIC**
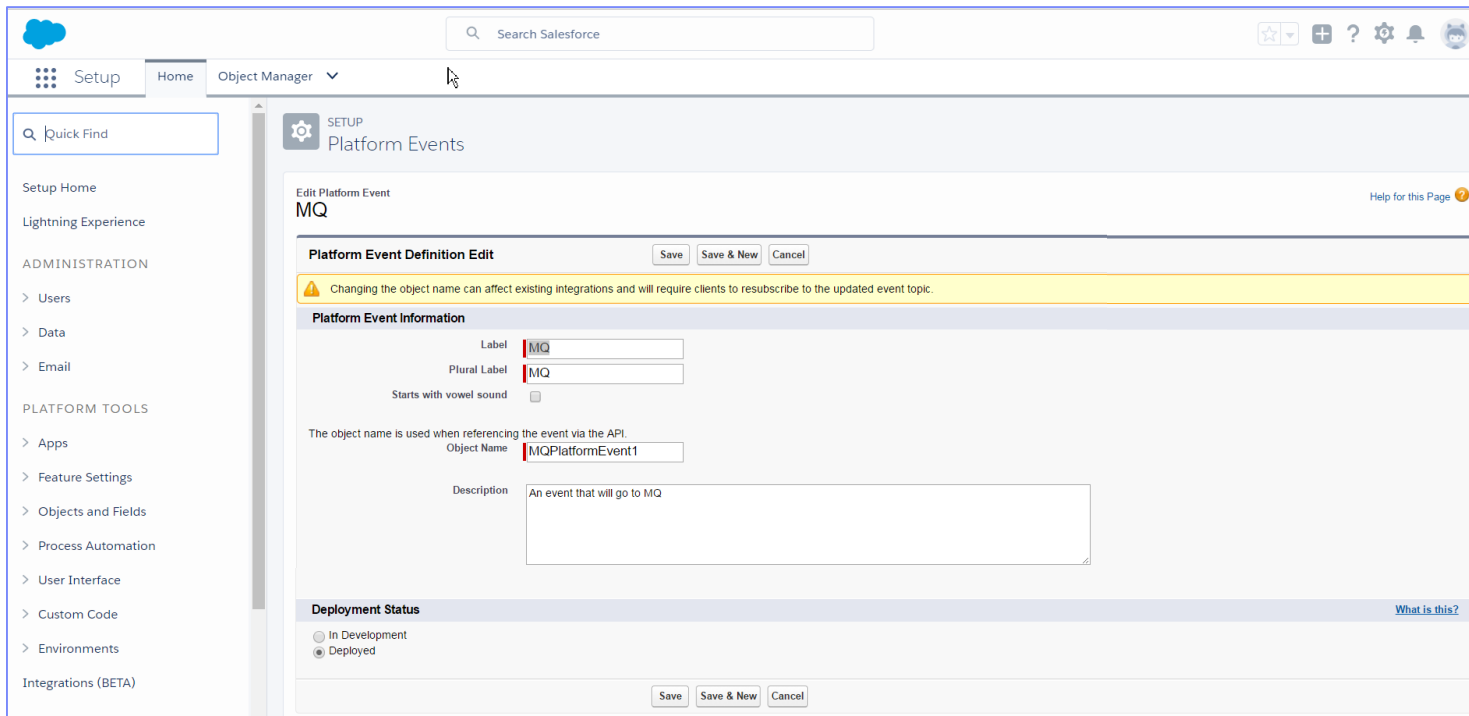
## Defining a PushTopic

- Can be done by executing APEX code
  - Others APIs exist

```
PushTopic pushTopic = new PushTopic();
pushTopic.Name = 'EscalatedCaseUpdates';
pushTopic.Query = 'SELECT Id, AccountId, Subject FROM CASE
                   WHERE Status = \'Escalated\' AND AccountId != NULL';
pushTopic.ApiVersion = 39.0;
pushTopic.NotifyForOperationCreate =true;
pushTopic.NotifyForOperationUpdate =true ;
pushTopic.NotifyForOperationUndelete =true ;
pushTopic.NotifyForOperationDelete = true ;
pushTopic.NotifyForFields = 'Referenced';
insert pushTopic;
```

- PushTopic will now be generated automatically whenever the Query is satisfied

# IBM MQ – Bridges

## Defining a Platform Event

## Platform Event definition in Salesforce



SETUP
Platform Events

**Platform Event**
# MQ

Standard Fields [3]  |  Custom Fields & Relationships [2]

### Platform Event Definition Detail     [Edit] [Delete]

|  |  |  |
|---|---|---|
| Singular Label | MQ | |
| Plural Label | MQ | Deploy |
| Object Name | MQPlatformEvent1 | |
| API Name | MQPlatformEvent1__e | |
| Created By | Mark Taylor, 03/01/2017 11:19 | |

### Standard Fields

| Action | Field Label | Field Name | Data Type |
|---|---|---|---|
| | Created By | CreatedBy | Lookup(User) |
| | Created Date | CreatedDate | Date/Time |
| | Replay ID | ReplayId | External Lookup |

### Custom Fields & Relationships     [New]

| Action | Field Label | API Name | Data Type | Indexed | Controlling Field |
|---|---|---|---|---|---|
| Edit \| Del | MyText | MyText__c | Text(64) | | |
| Edit \| Del | Name | Name__c | Text(32) | | |

## Creating a Platform Event

- Can be done by executing APEX code
  - Others APIs exist

```
MQPlatformEvent1__e pe;

pe = new MQPlatformevent1__e(Name__c='Mark');

Database.SaveResult result = EventBus.publish(pe);

if (result.isSuccess()) {
    System.debug('Published OK');
} else {
    for (Database.Error err : result.getErrors()) {
      System.debug('Failed with ' +  err.getStatusCode() + err.getMessage());
    }
}
```

## Getting started with the bridge

- Configure the bridge
  - Connection details for MQ (queue manager name, channel, userid etc)
  - Connection details for Salesforce (id, password, OAuth secret, URL etc)
  - TLS keystores (uses same stores for both sides of bridge)
  - Topic information (which topics to listen to from Salesforce, publication root for MQ, QoS)

- Configure Salesforce
  - Define pushtopic rules, define platform event names, add any event generation code

- Configure MQ
  - Define queue for synchronisation values
  - Define topic root, set security rules

- And then start the bridge so that any MQ application can receive the publications
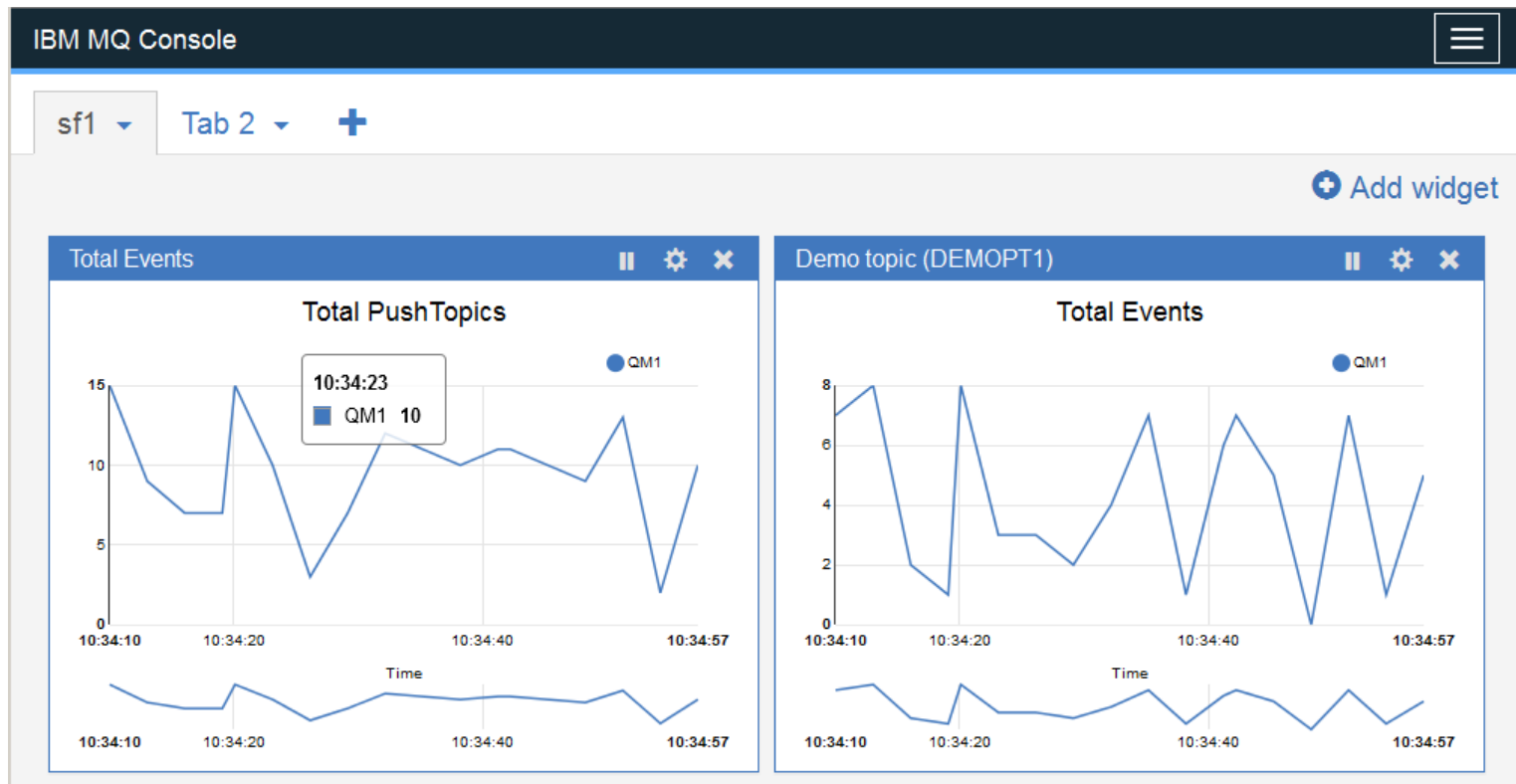
## MQ Applications

- Any MQ application can subscribe to the topics
  - Standard MQ APIs for accessing these
- "Application" could be IIB (AIS) message flow that exploits Salesforce node for further processing

- Data is JSON format (option to republish full data or just sobject/payload)

```
{   "clientId":"w211mr7nytfdadqzu1pq4sbovyh",
    "data": {
      "event": {
        "createdDate":"2016-11-12T19:09:31.384Z", replayId":10, "type":"created"
      },
      "sobject": {
        "AccountId":"Y000009Yxb", "Id":"Y000001QGp", "Subject":"Acct updated"
      }
    },
    "channel":"/topic/EscalatedCaseUpdates"
}
```
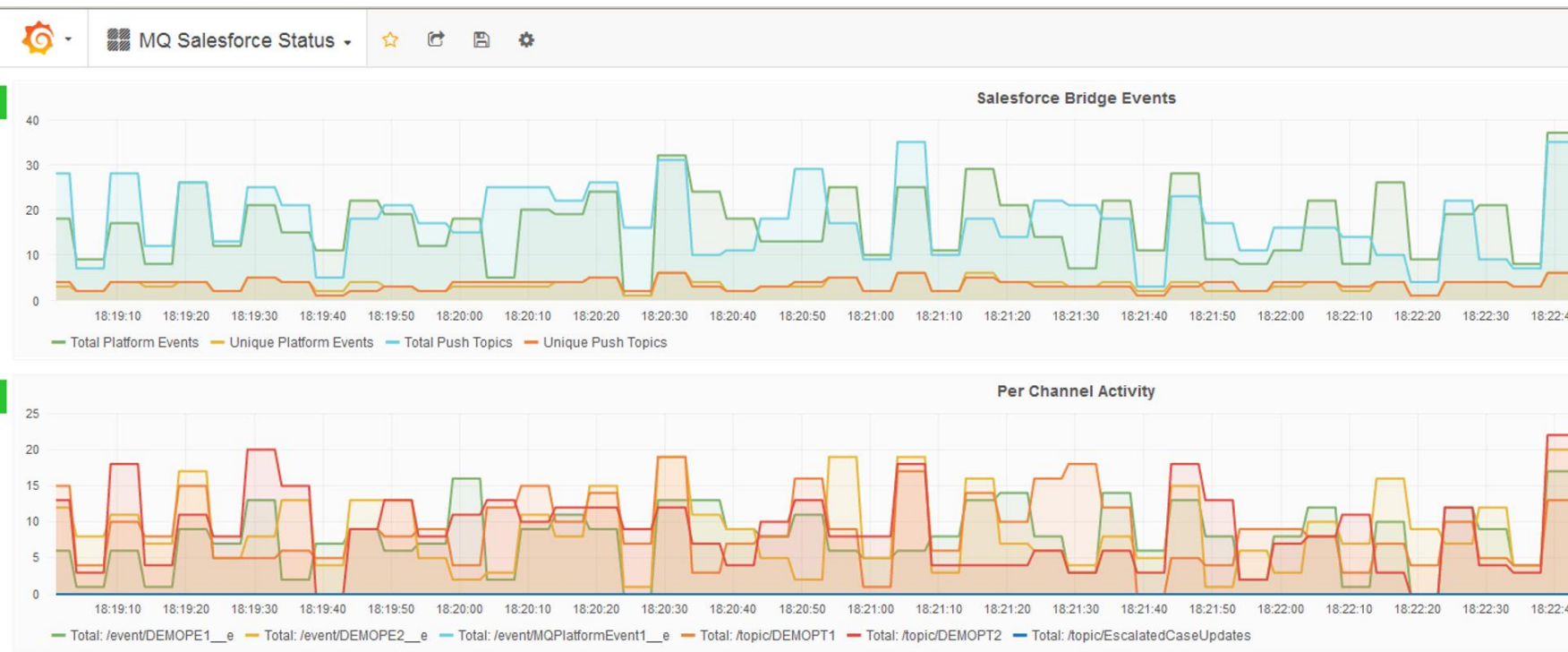
## Qualities of Service

- The bridge supports two QoS
  - At **most** once delivery
  - At **least** once delivery

- All events from Salesforce have a sequence number (replayId) unique to the event name
  - Events are held inside Salesforce for potential replay for up to 24 hours
  - The sequence number of last event processed is persisted in MQ
  - On bridge restart, can use that sequence number to request all events since the last one processed
    - At **least** once QoS
  - Or can just ask for only new events
    - At **most** once QoS

- If the bridge out of contact with Salesforce for more than 24 hours, old events are not available
  - Even with at least once QoS
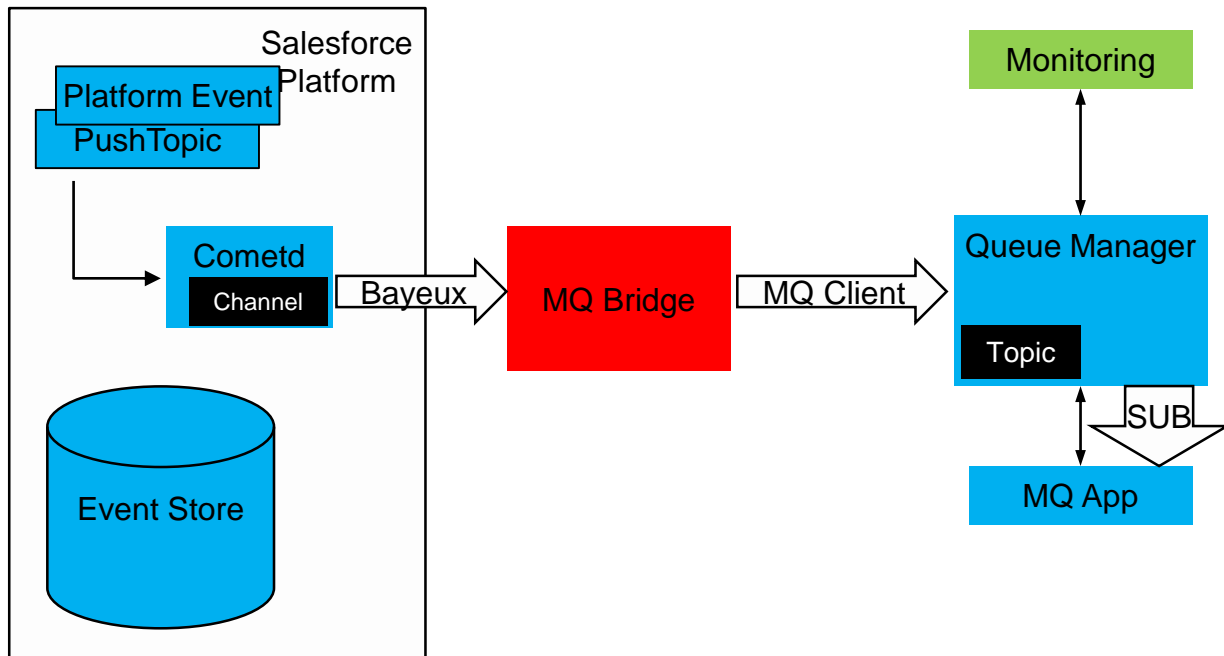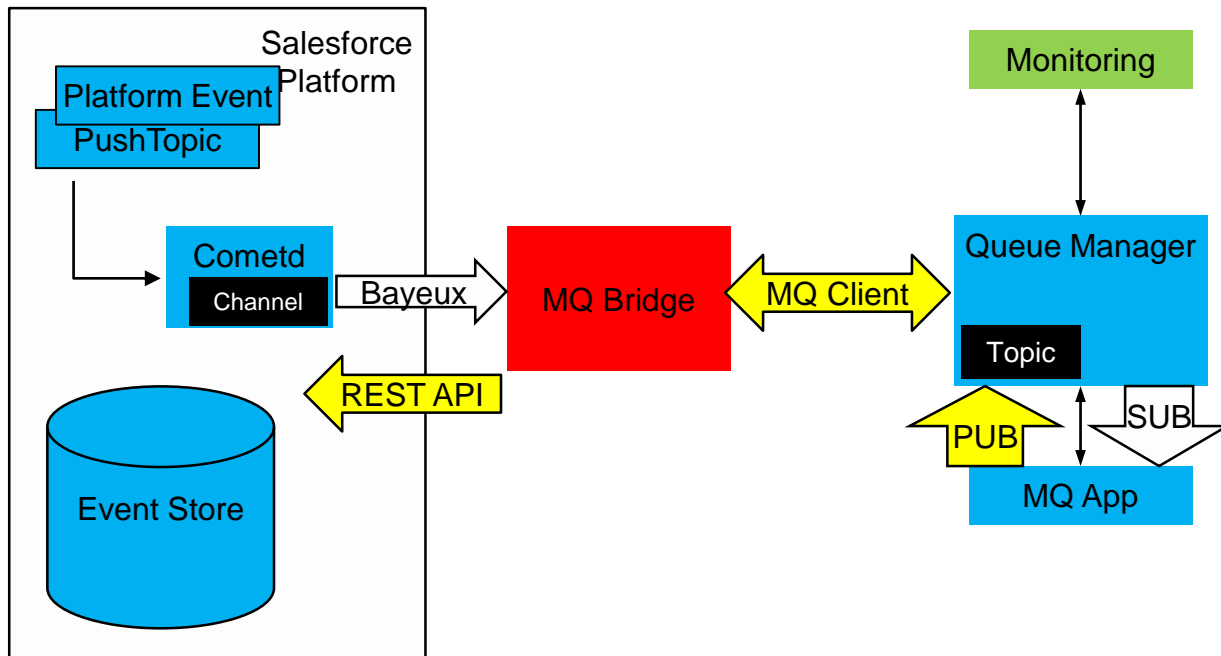
## Monitoring using the MQ Console

## Monitoring using Grafana

# POSSIBLE ENHANCEMENT: TWO-WAY BRIDGE

## Design



Salesforce Platform

Platform Event

PushTopic

Cometd

Channel

Bayeux

MQ Bridge

MQ Client

Event Store

Monitoring

Queue Manager

Topic

SUB

MQ App

## Design (two-way extension)

## Two-way Extension

- As well as subscribing to Salesforce topics, bridge subscribes to MQ topics
- MQ applications publish to that tree
- Bridge sends them to Salesforce
  - Then available to any Salesforce application
  - Including any MQ apps listening via the bridge

- Publications contain JSON corresponding to Platform Event fields
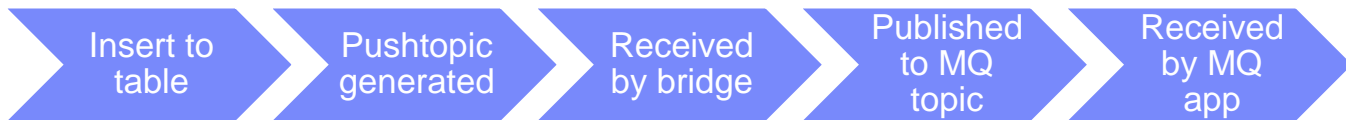  - MQ topic name indicates Salesforce event name

```
Topic:
  /<root>/mqtosfb/event/MQPlatformEvent1__e

Message Body:
  {"MyText__c" : "Some text here", "Name__c" : "Bob Smith" }
```

## Demo Outline

- A change is made to a Salesforce table
- Resulting in an pushtopic being generated
- Picked up by the bridge
- And republished to an MQ application that is subscribed to the topic

| Insert to table | Pushtopic generated | Received by bridge | Published to MQ topic | Received by MQ app |
| --- | --- | --- | --- | --- |

# DEMO

## For more info on Salesforce event processing

- MQ Knowledge Centre
  - https://www.ibm.com/support/knowledgecenter/en/SSFKSJ_9.0.0/com.ibm.mq.con.doc/q129310_.htm

- MQ blog entry (and link to video) at
  - https://www.ibm.com/developerworks/community/blogs/messaging/entry/DRAFT_IBM_MQ_and_Salesforce_messaging?lang=en

- PushTopics described at
  - https://developer.salesforce.com/docs/atlas.en-us.api_streaming.meta/api_streaming/intro_stream.htm

- Platform Events at
  - https://developer.salesforce.com/docs/atlas.en-us.platform_events.meta/platform_events/platform_events_intro.htm

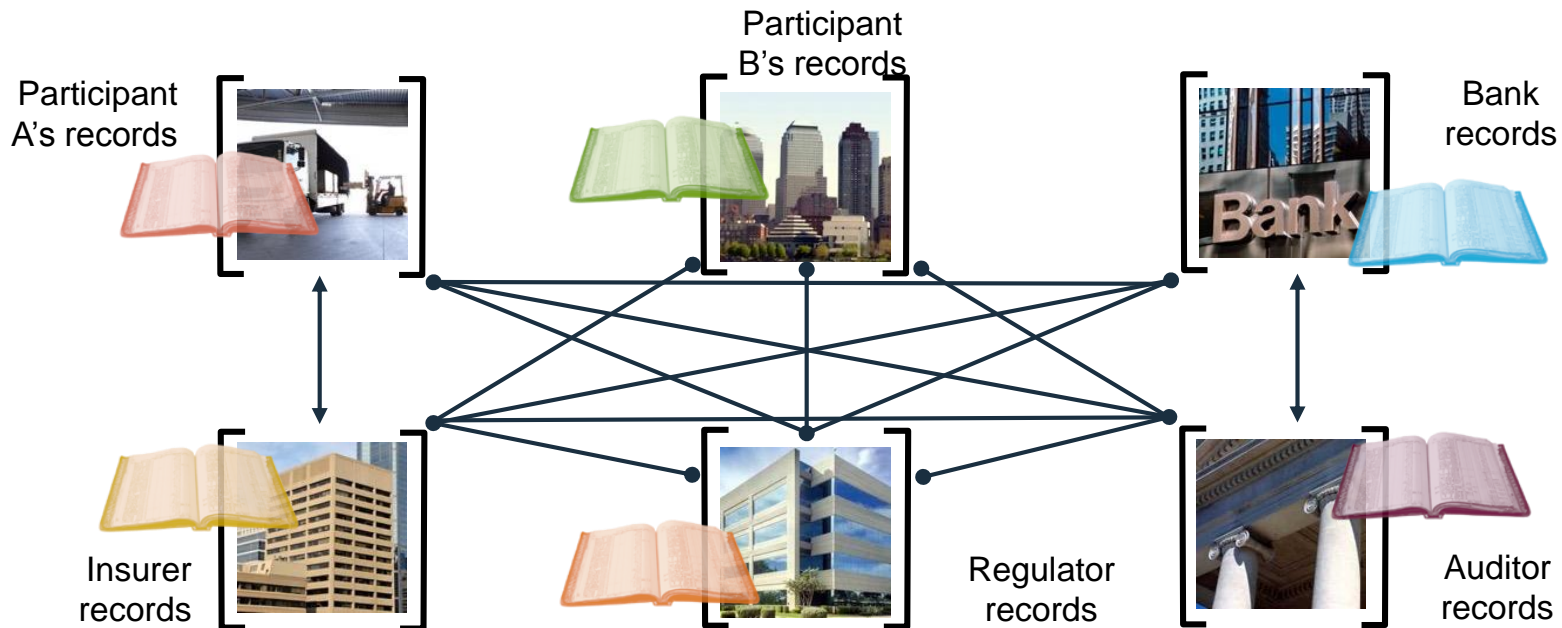- Testing with Salesforce – Can try these out using **FREE** developer accounts

# MQ BRIDGE FOR BLOCKCHAIN IN THE MQ ADVANCED FOR Z/OS VUE PRODUCT

# WHAT IS BLOCKCHAIN

## Problem …



Participant A's records

Participant B's records

Bank records

Insurer records

Regulator records

Auditor records

… inefficient, expensive, vulnerable

## A shared, replicated, permissioned ledger …

Participant
A's records

Participant
B's records

Bank
records

Blockchain

Insurer
records

Regulator
records

Auditor
records
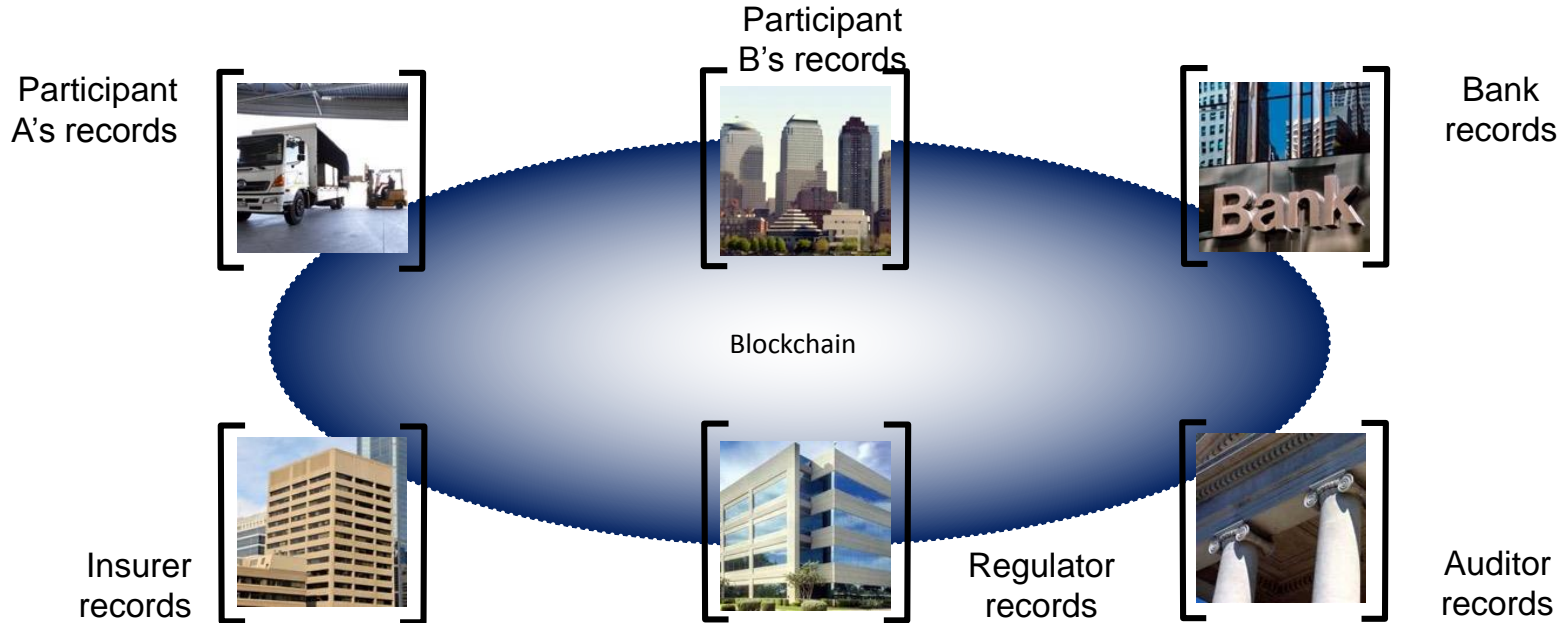
… with consensus, provenance, immutability and finality

## Shared ledger

- Shared between participants

- Participants have own copy through replication

- Permissioned, so participants see only appropriate transactions

- THE shared system of record
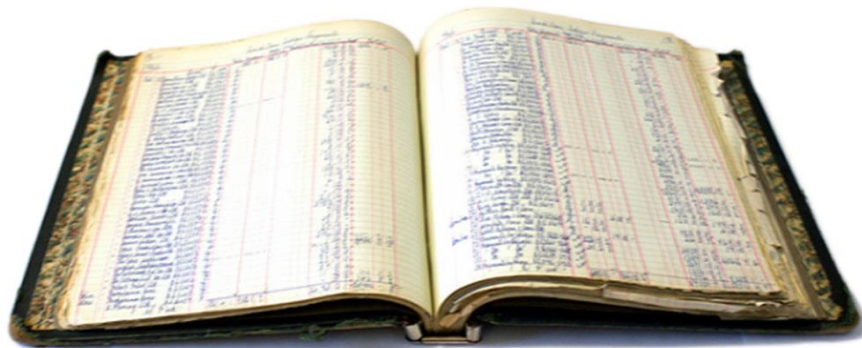
## Hyperledger: A Linux Foundation project

- A collaborative effort created to advance cross-industry blockchain technologies for business

- Announced December 2015, now around 150 members

- Open source, open standards, open governance

- Five frameworks and three tools projects

- IBM is a premier member of Hyperledger

www.hyperledger.org
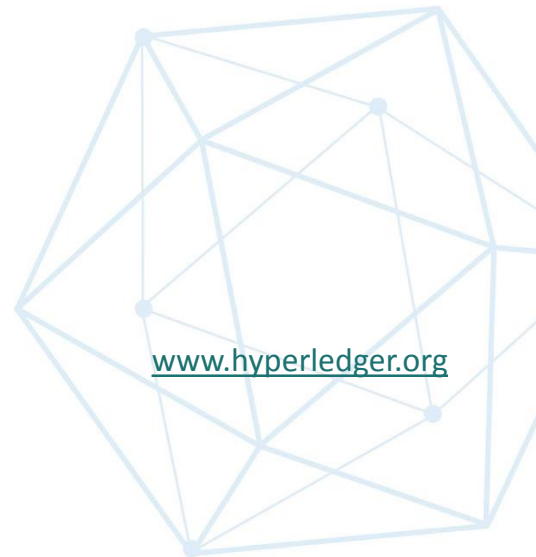
## Hyperledger Fabric: Distributed ledger platform
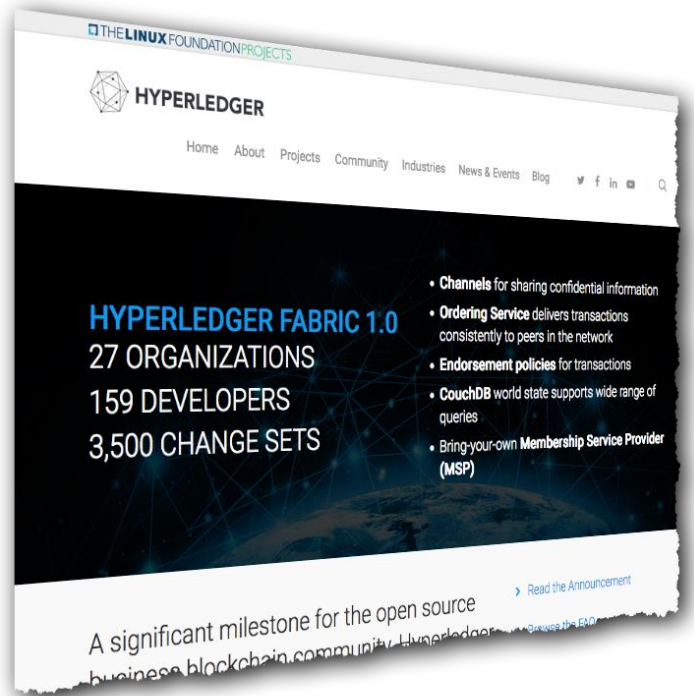


- An implementation of blockchain technology that is a foundation for developing blockchain applications

- Emphasis on ledger, smart contracts, consensus, confidentiality, resiliency and scalability.

- V1.0 released July 2017

  - 159 developers from 27 organizations

  - IBM is one contributor of code, IP and development effort to Hyperledger Fabric

http://hyperledger-fabric.readthedocs.io/

## Introducing the IBM Blockchain Platform

- **The only fully integrated enterprise-ready blockchain platform designed to accelerate the development, governance, and operation of a multi-institution business network**

- Based on Hyperledger Fabric V1 runtime optimized for enterprise requirements

- Specialized compute for security, performance and resilience

- Delivered via the IBM Cloud on a global footprint with 24x7 Integrated Support

- Full lifecycle tooling to speed activation and management of your network

### Develop

Explore and accelerate development time with tools that ensure close alignment between business leaders and developers

### Govern

Speed activation, customization and management of your business network with democratic, multi-party governance tooling

### Operate

Deploy and operate always-on networks with production-ready enterprise performance and security for most demanding use cases

[http://ibm.biz/Platform_Demo]

# THE MQ BRIDGE TO BLOCKCHAIN

## Blockchain Bridge

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

      MQ ➔ Blockchain ADV VUE V903

- Asynchronous request/reply MQ message flow for applications to request information from Blockchain ("what is the value of the balance on this account") over MQ queues
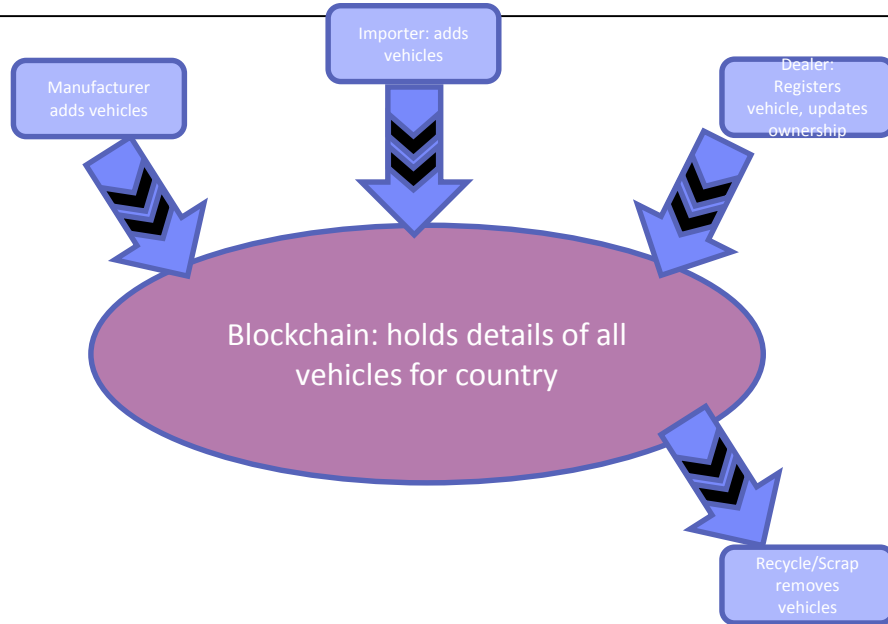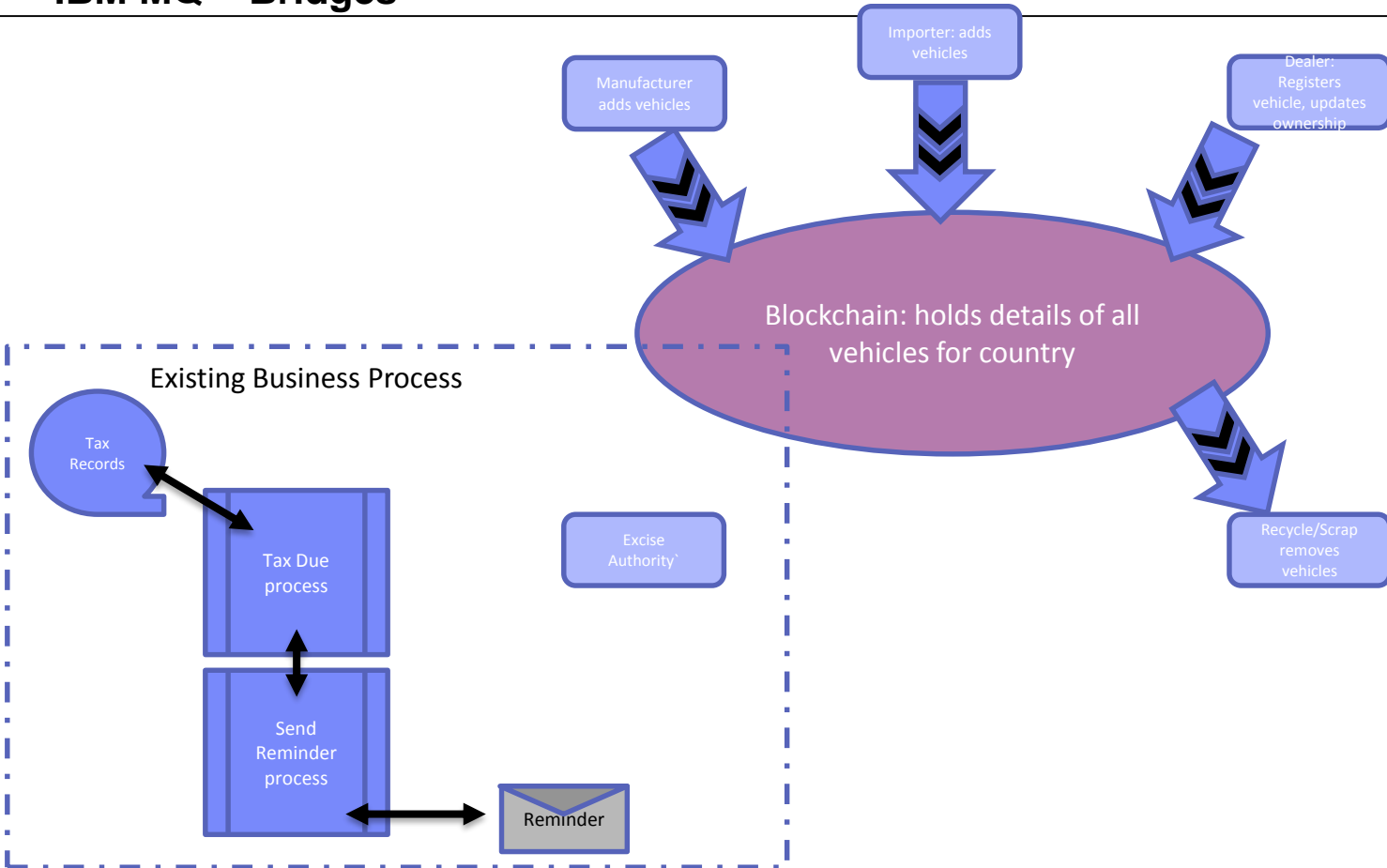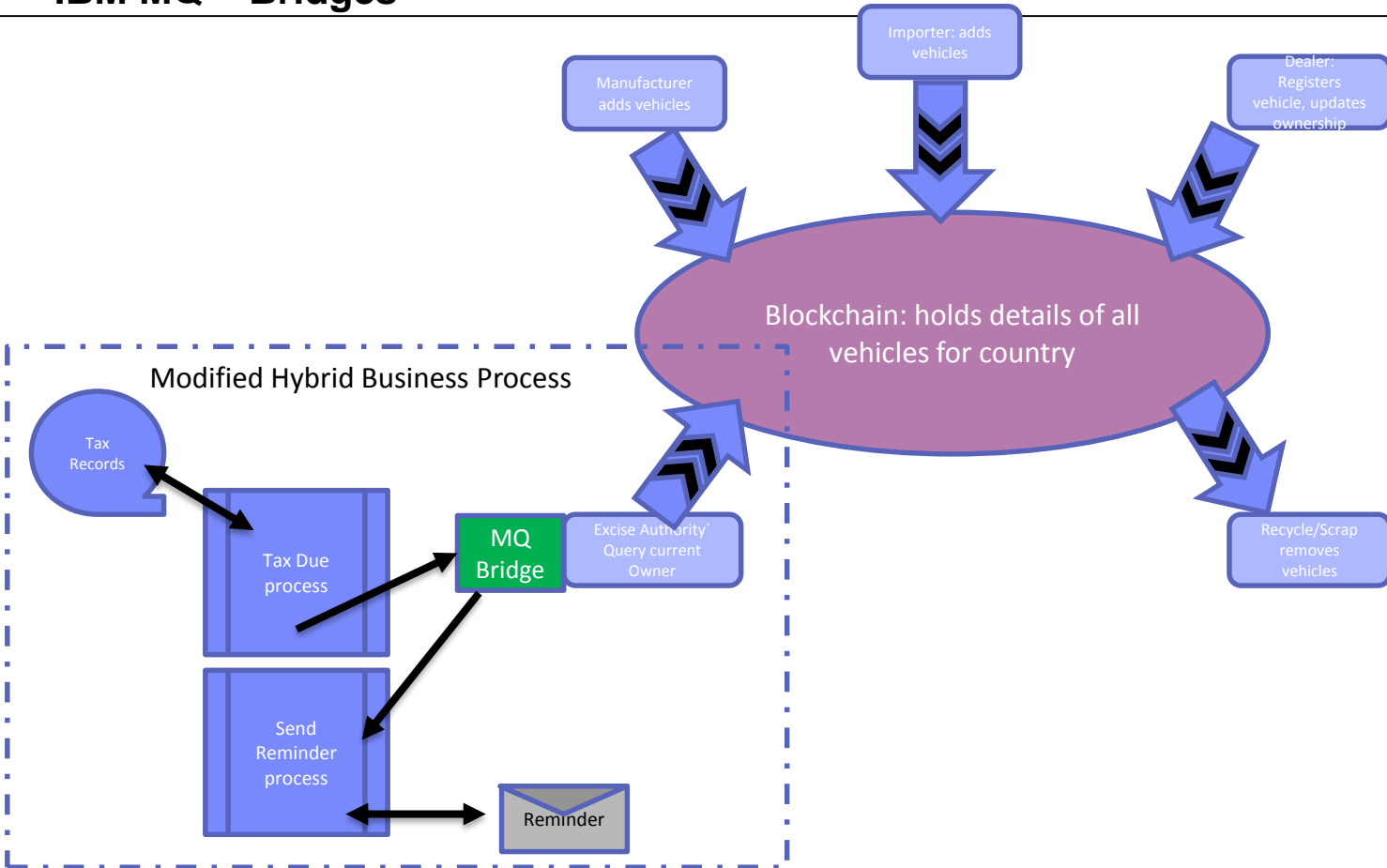
IBM MQ
Bridge for
Blockchain

IBM **MQ**

Manufacturer adds vehicles

Importer: adds vehicles

Dealer: Registers vehicle, updates ownership

Blockchain: holds details of all vehicles for country

Recycle/Scrap removes vehicles

Manufacturer adds vehicles

Importer: adds vehicles

Dealer: Registers vehicle, updates ownership

Blockchain: holds details of all vehicles for country

Recycle/Scrap removes vehicles

Existing Business Process

Tax Records

Tax Due process

Send Reminder process

Excise Authority`

Reminder

# IBM MQ – Bridges

# DETAILS

## Initial V9.0.3 Release

**MQ for z/OS (VUE)**

**x64 Linux**

**IBM Bluemix**

Queue Manager

MQ Bridge to Blockchain

IBM Blockchain

**Input Queue**

**Reply Queue**

**Reply (JSON)**

**parse request, submit query**

**parse response, build, return reply messsage**

MQ JMS Client

Hyperledger JAVA SDK

**HSBN Plan Service**

**Blockchain**

**Request (JSON)**

**User Application**
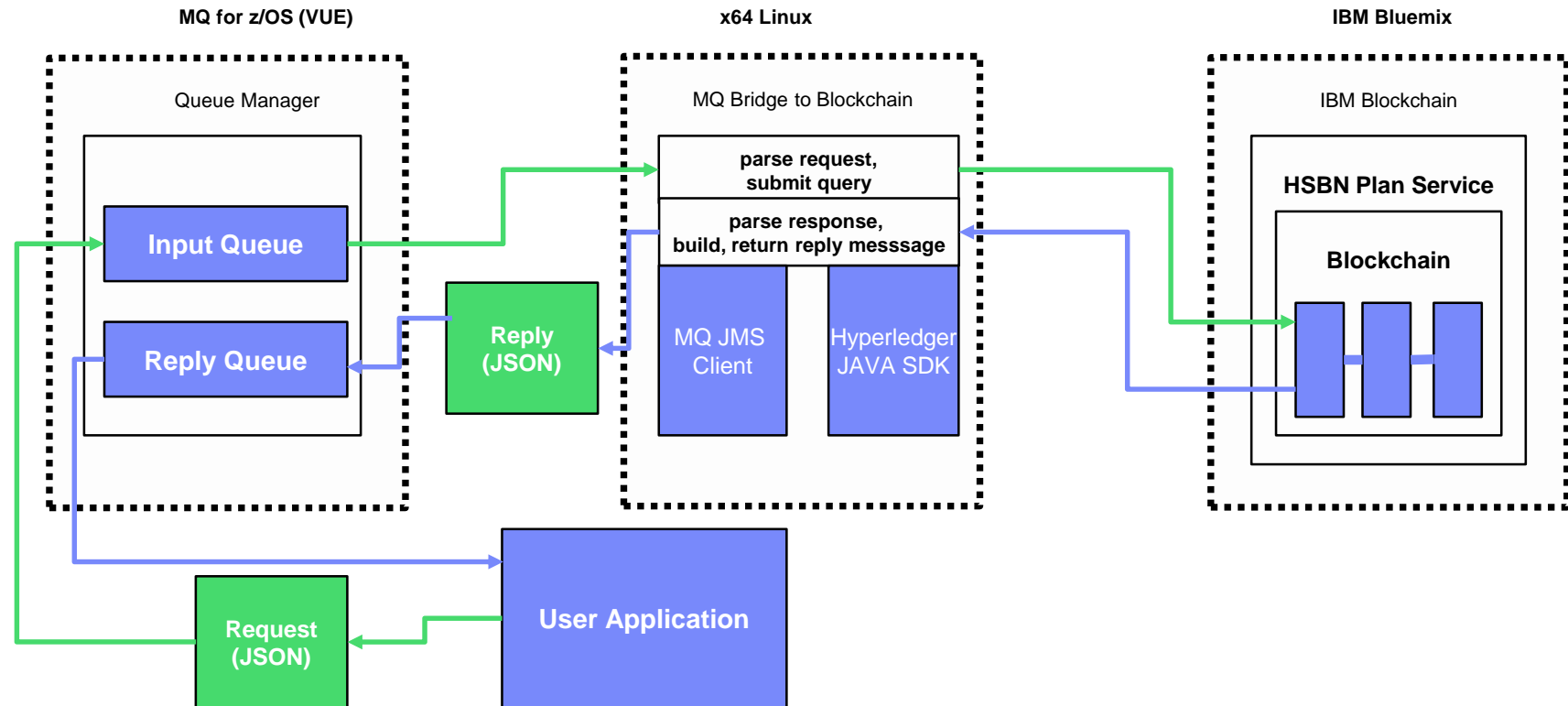
## Installation/configuration

- Bridge package is shipped as single zip file on MQ z/OS VUE
  - Includes bridge code, sample code and definitions, and prereq 3<sup>rd</sup> party jars

- Copy file to xLinux system where MQ JMS already installed, unpack it, and then run configuration

- Config asks for connection information to MQ and to Blockchain/Bluemix
  - For blockchain, need names/ports for various servers, a userid and password, PEM file. All available in the detailed configuration file as part of the Bluemix service
  - For MQ, needs client connection information, optionally userid and password

- Configure qmgr with necessary queues and security

- Can then start in normal runtime mode
  - Connections made to both sides at startup, so configuration errors are rapidly discovered

## Configuration (partial)

```
Connection to Queue Manager
---------------------------
Queue Manager                    : [V9000_A]
Bridge Input Queue               : [SYSTEM.BLOCKCHAIN.INPUT.QUEUE]
Bridge User Identity Queue       : [SYSTEM.BLOCKCHAIN.IDENTITY.QUEUE]
MQ Channel                       : [SYSTEM.DEF.SVRCONN]
MQ Conname                       : [host1.example.com(3714)]
Blockchain - User Identification
--------------------------------
Blockchain Userid                : [WebAppAdmin]
Enrollment Secret                : []******
Blockchain - Organisation Identification
----------------------------------------
Organisation Name                : [PeerOrg1]
Organisation MSPId               : [PeerOrg1]
Certificate Authority servers    : [ldn1-zbc5a.2.secure.blockchain.ibm.com:14511]
Blockchain server locations
---------------------------
Peer servers (usually at least 2) : [
fabric-peer-1a ldn1-zbc5a.2.secure.blockchain.ibm.com:14521,
fabric-peer-1c ldn1-zbc5c.2.secure.blockchain.ibm.com:14521]
…
```

## Application interface

- Applications interact with the bridge by putting a JSON-formatted message on the queue
  - Sample app provided to show construction of fields

```
{ "channel": "myFirstChannel",
  "chaincodeName": "example1",
  "chaincodeVersion": "1",
  "function" : "invoke",
  "args": ["query", "b", "c", "d"]
}
```

- Response contains requested data indication of success or failure
  - JSON fields to distinguish (statusCode=20x is success)
  - Errors may come directly from the bridge (eg badly formatted input message), from the hyperledger fabric (eg unknown chaincode), or from the chaincode itself (eg bad function parameters)

```
{ "data": ["50","10","10"],
  "message": "OK",
  "statusCode": 200,
  "statusType": "SUCCESS"
}
```

```
{ "message": "Sending proposal to peer-1a failed
because of gRPC failure=Status{code=UNKNOWN,
description={\"Error\":\"Nil amount for c\"},
cause=null}",
    "statusCode": 500, "statusType": "FAILURE" }
```
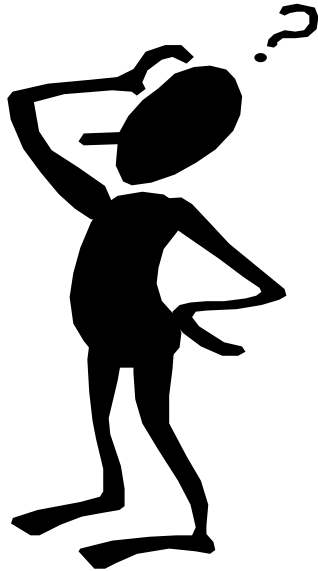
# POSSIBLE ENHANCEMENTS

## Possible enhancements

- Extend platform coverage
  - Make installable in usual way on a distributed platform
  - Support more queue manager platforms
  - Support more than Bluemix for Hyperledger Fabric instances

- Functional enhancements
  - Be able to issue "update" as well as "query" operations

## Summary

- Simple extension allows MQ apps to exploit new services
  - Event-driven Salesforce applications
  - Integration with Hyperledger Fabric Blockchain

- Follows existing MQ application models

- Easy configuration

- Easy to extend existing processes

**Any questions?**