Building cross-platform hybrid applications using AMQP 1.0 with Apache Qpid

David Ingham (Microsoft)

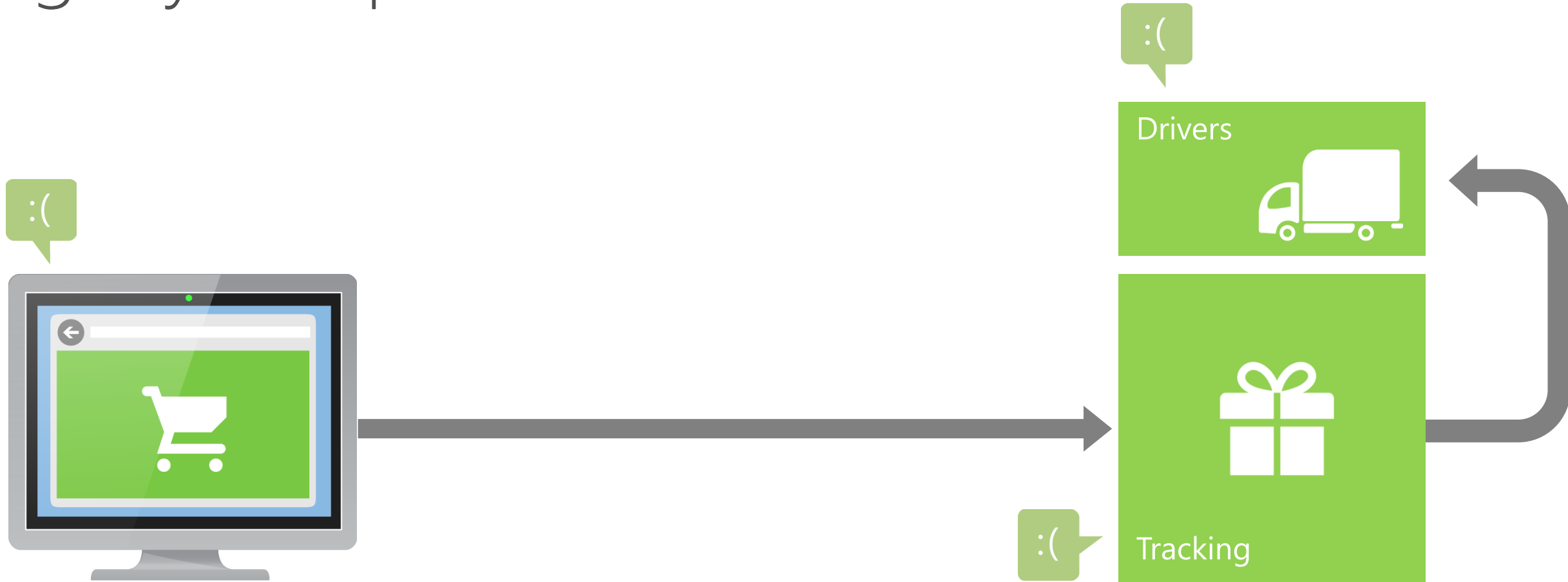Robert Godfrey (JP Morgan Chase)

William Henry (Red Hat)

Agenda

Introduction to messaging

The case for a standard protocol

What is AMQP 1.0?

AMQP 1.0 implementations

Demo with code drill-down
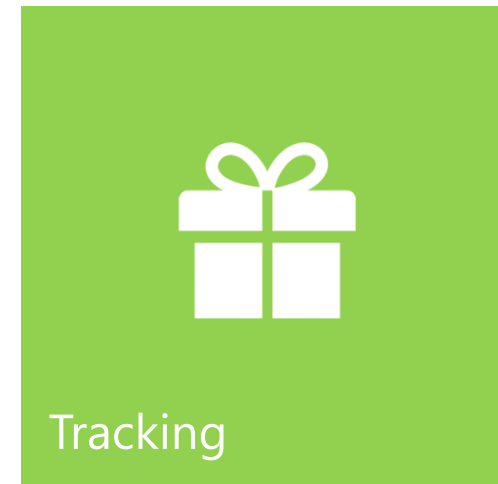
Summary

Tightly Coupled

Drivers

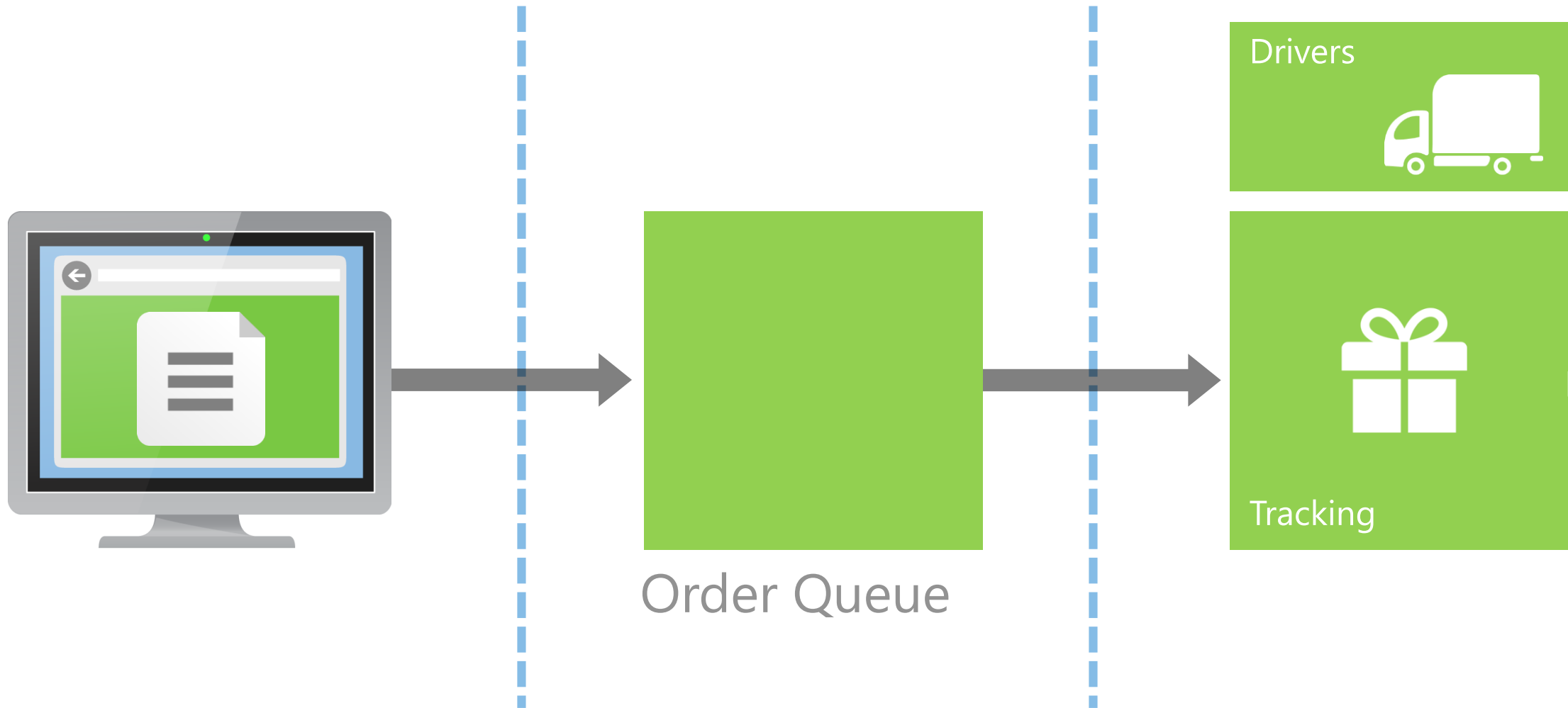Tracking

Tightly Coupled

Loosely Coupled

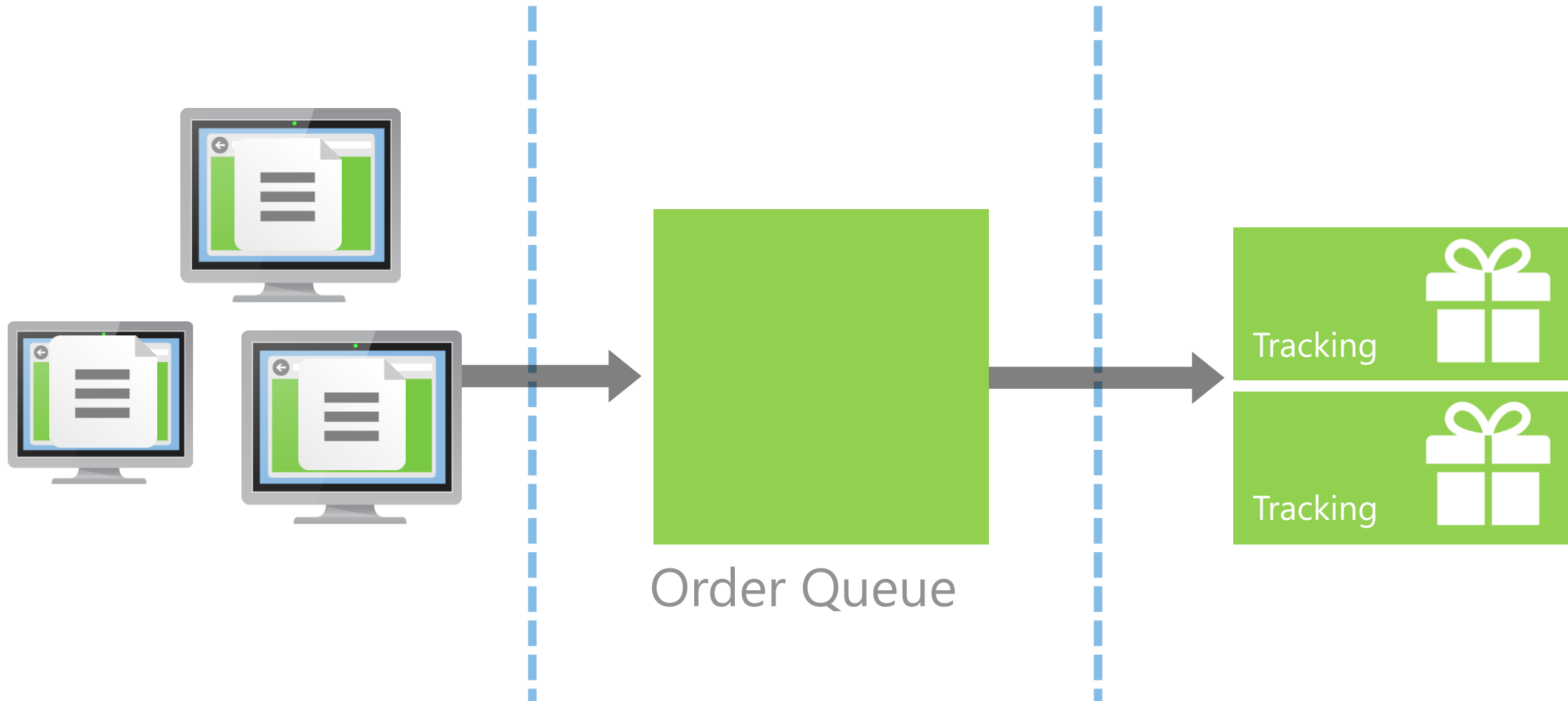Order Queue

Drivers

Tracking

Loosely Coupled

Order Queue

Drivers

Tracking

Loosely Coupled

Drivers

Order Queue

:(

Tracking

Loosely Coupled

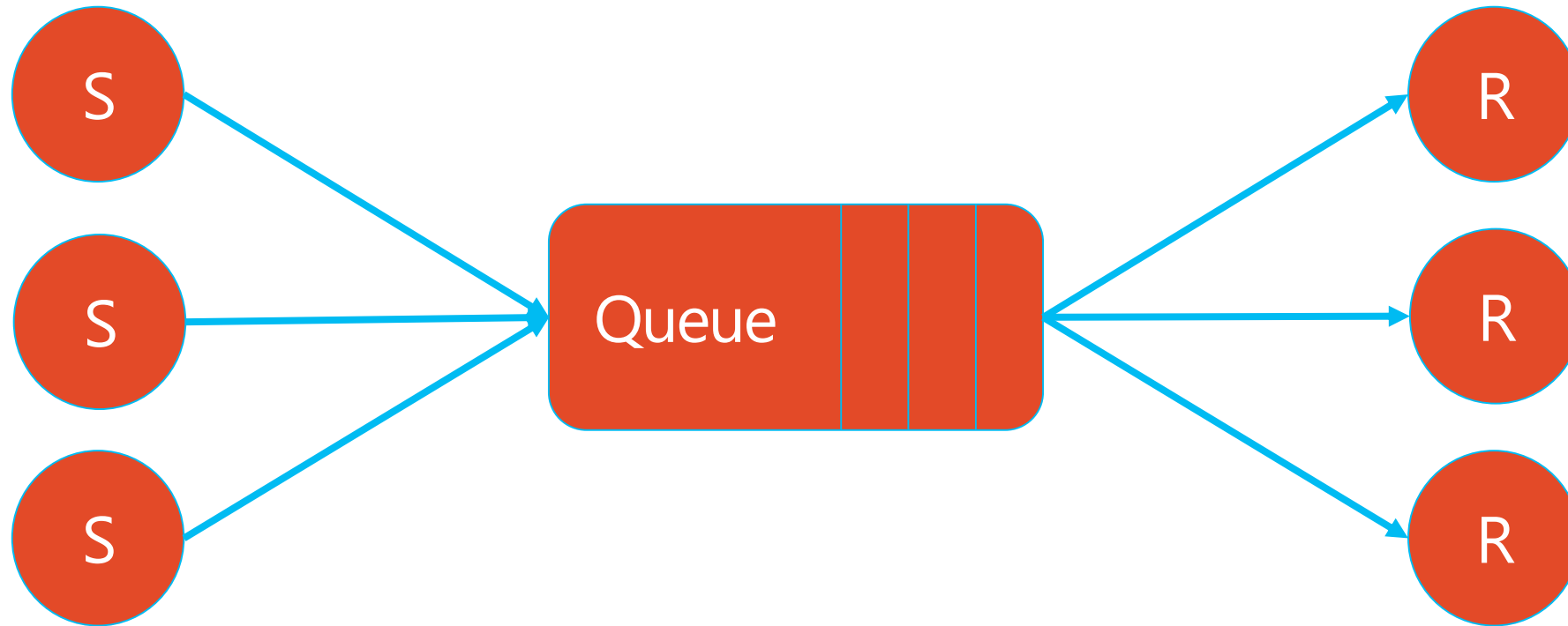Order Queue

Tracking

Tracking
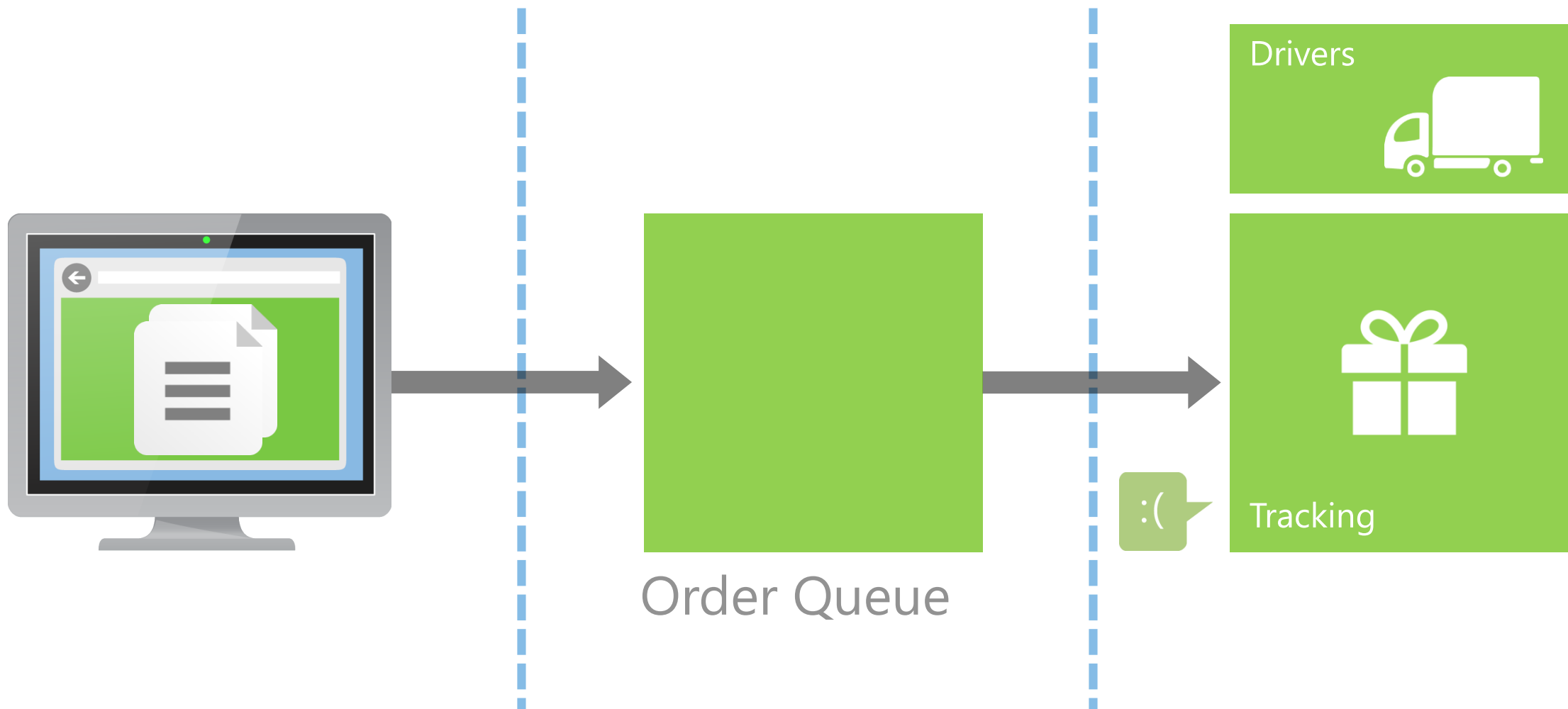
Messaging concepts: queuing

Messaging concepts: pub/sub
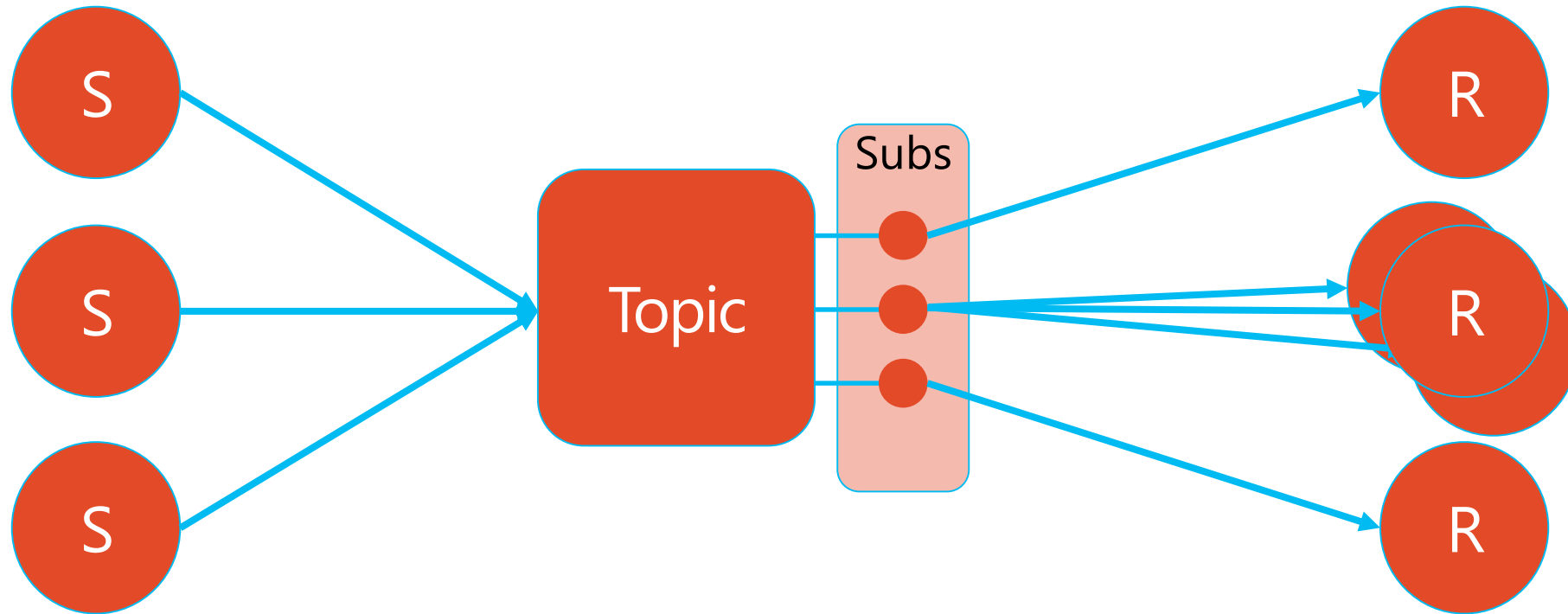
Messaging concepts: message

Message

Key	Value
Key	Value
Key	Value
Key	Value

Body

Properties

Key/value pairs exposed to the broker

Subscription rules can filter based on properties

Body

Opaque payload not exposed to the broker

Can be used for encrypted data

Message-oriented middleware allows application modules to be distributed over **heterogeneous platforms** and reduces the complexity of developing applications that **span multiple operating systems** ...

-- Wikipedia entry for message-oriented middleware

Proprietary messaging protocols

Difficult to port applications

Requires re-coding all applications

Difficult to integrate

Application level bridges to move messages and translate message formats

Restricted platform support

Limited to whatever vendor provides

AMQP 1.0
Advanced Message Queuing Protocol

AMQP

Open, standard messaging protocol

Enables cross-platform apps to be built using brokers, libraries and frameworks from different vendors

Features

Efficient – binary connection-oriented protocol

Reliable – fire-and-forget to reliable, exactly-once delivery

Portable data representation – cross-platform, full-fidelity exchange

Flexible **–** peer-peer, client-broker, and broker-broker topologies

Broker-model independent **–** no requirements on broker internals

OASIS AMQP 1.0 Standard released

Last week OASIS announced the ratification of the AMQP 1.0 Standard

Software vendors and end-users can bet on AMQP 1.0 knowing it's a stable, well-supported protocol standard

The culmination of several years effort by more than 20 companies

Technology vendors: Axway Software, Huawei Technologies, IIT Software, INETCO Systems, Kaazing, Microsoft, Mitre Corporation, Primeton Technologies, Progress Software, Red Hat, SITA, Software AG, Solace Systems, VMware, WSO2, Zenika. **User firms**: Bank of America, Credit Suisse, Deutsche Boerse, Goldman Sachs, JPMorgan Chase.

VMware

"AMQP 1.0 is a novel addition to the growing toolkit of open protocols for transporting data between systems and virtualized application delivery. Standard transports enable lower cost business integration and messaging. AMQP 1.0 admits many use cases by defining safe message transfer between peers, without the constraint of a message broker model. With its open license, we anticipate both AMQP's wide adoption by messaging servers, and its use as a new API for database and integration products."

--*Alexis Richardson, Senior Director*

Red Hat

"Red Hat is pleased to see the hard work of the Technical Committee come to fruition. We are a founding member of the AMQP Technical Committee and have been active on the specification since the early days. AMQP 1.0 represents a significant improvement in the messaging arena and we expect to continue to support it in our products to best customer needs."

-- Mark Little, Vice President, Middleware Engineering

SoftwareAG

"A platform independent and vendor neutral protocol like AMQP removes hurdles in advancing interoperability of message-oriented middleware technologies. As a founding sponsor member of the AMQP TC and the related AMQP Steering Committee, Software AG is very pleased to see AMQP 1.0 transition to an OASIS Standard. Software AG supports numerous standards in its product suite, and AMQP has been an important addition to webMethods Nirvana, increasing interoperability and providing advanced messaging capabilities to our customers."

-- Prasad Yendluri, VP & Deputy CTO

Kaazing

"As the enablers of the Living Web and HTML5 WebSocket technology, everyone at Kaazing is excited to support OASIS' ongoing efforts to proliferate open standards and create a superior user web experience. Standardizing AMQP and combining it with WebSocket technology is an excellent strategy when building an event driven architecture. Working alongside OASIS, Kaazing has developed a Living Web in order to create the best possible web experience for users, reduce complexity, and increase interoperability."

-- John Fallows, CTO and Co-Founder

Microsoft

"Microsoft congratulates the AMQP community on approval of AMQP version 1.0 as an OASIS Standard. As an open and interoperable messaging protocol that can scale from mobile clients to the cloud, AMQP has benefitted from the participation of technical experts from around the world, and the achievement of this important milestone will lead to continued growth in the AMQP ecosystem. We look forward to working with the community to promote AMQP-based interoperability and innovation."

-- Scott Guthrie, Corporate VP, Microsoft's Server and Tools Business Division

AMQP user perspective

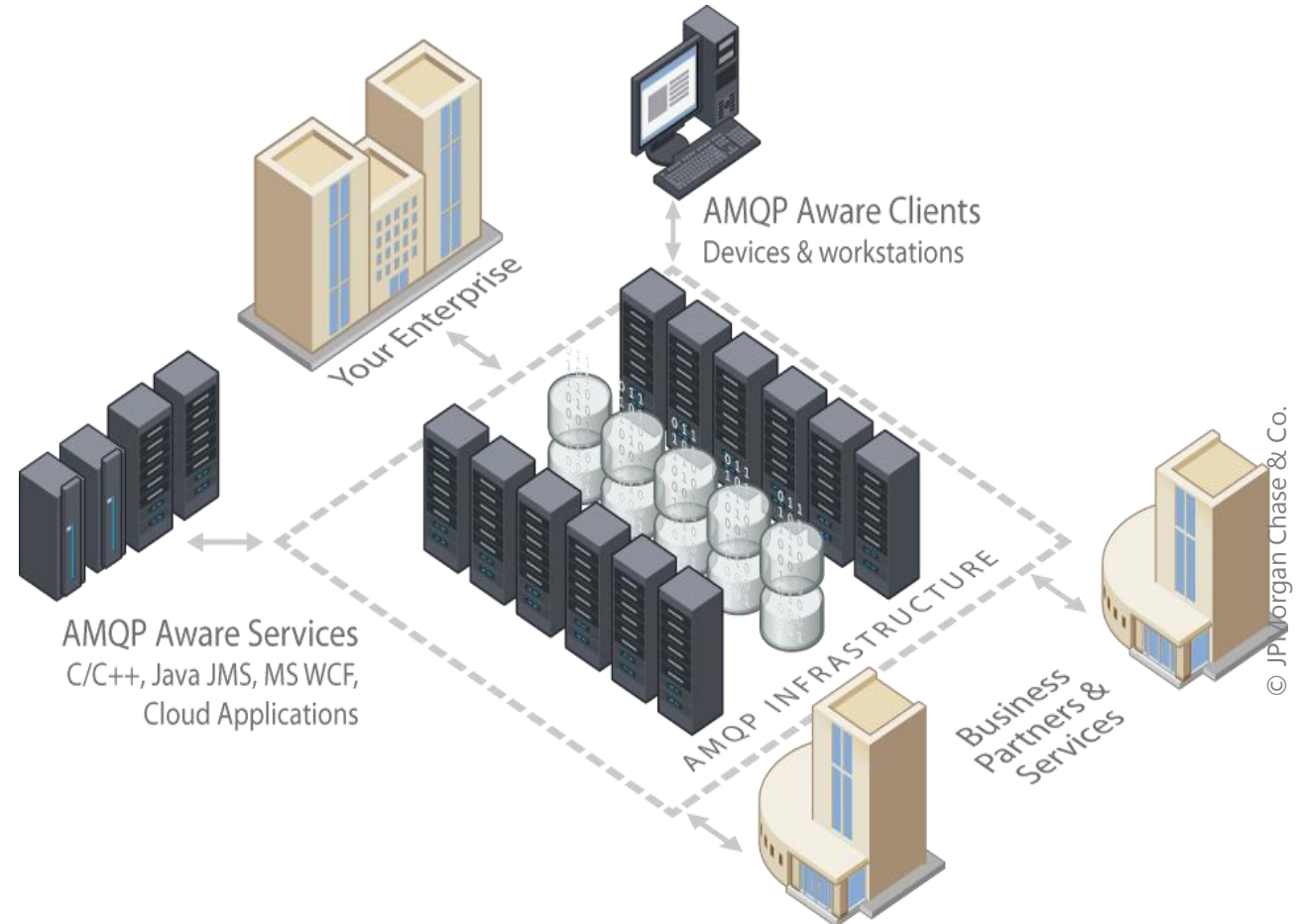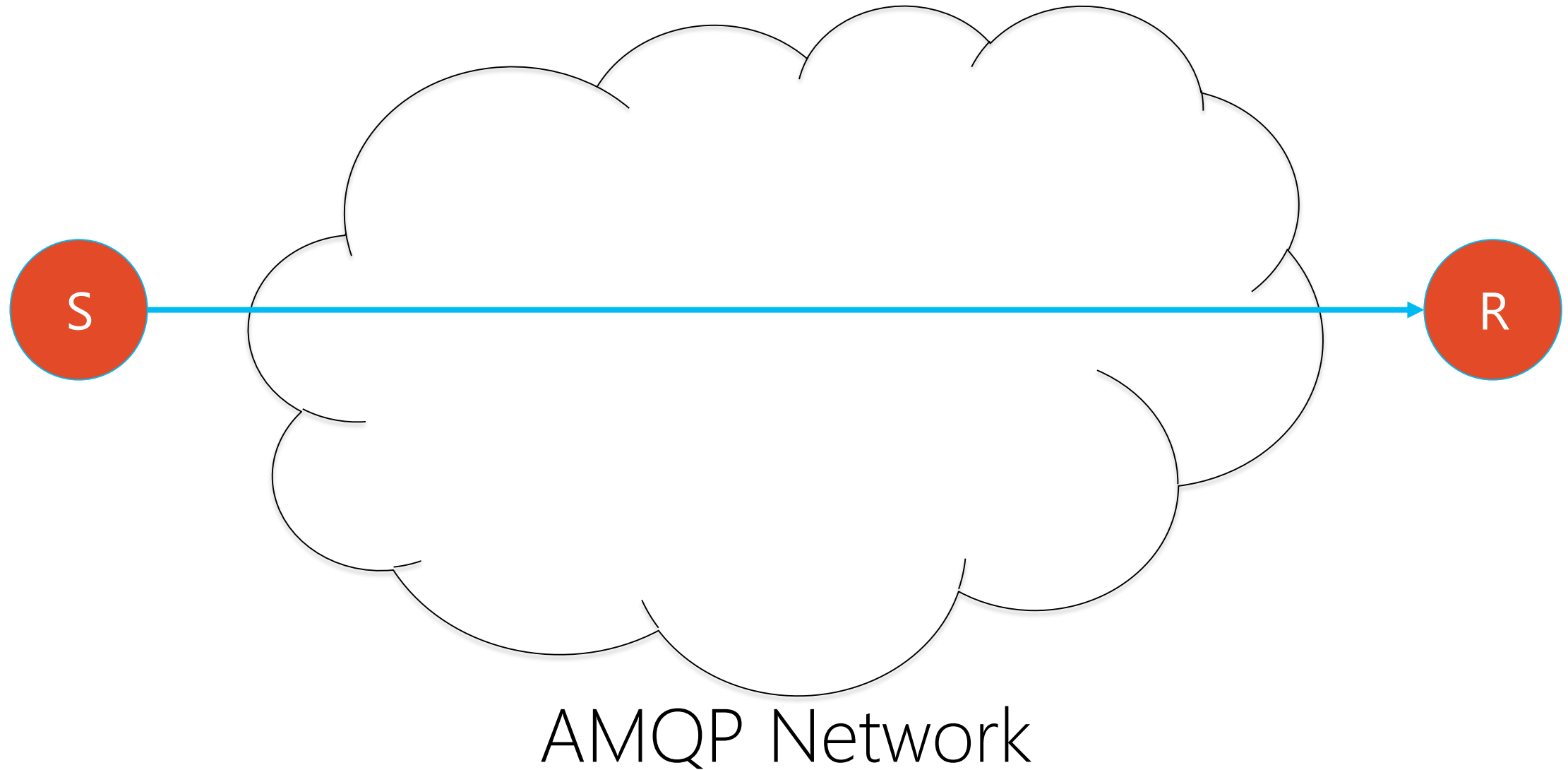Rob Godfrey, VP Integration Services, JP Morgan Chase

AMQP 1.0

Applicability
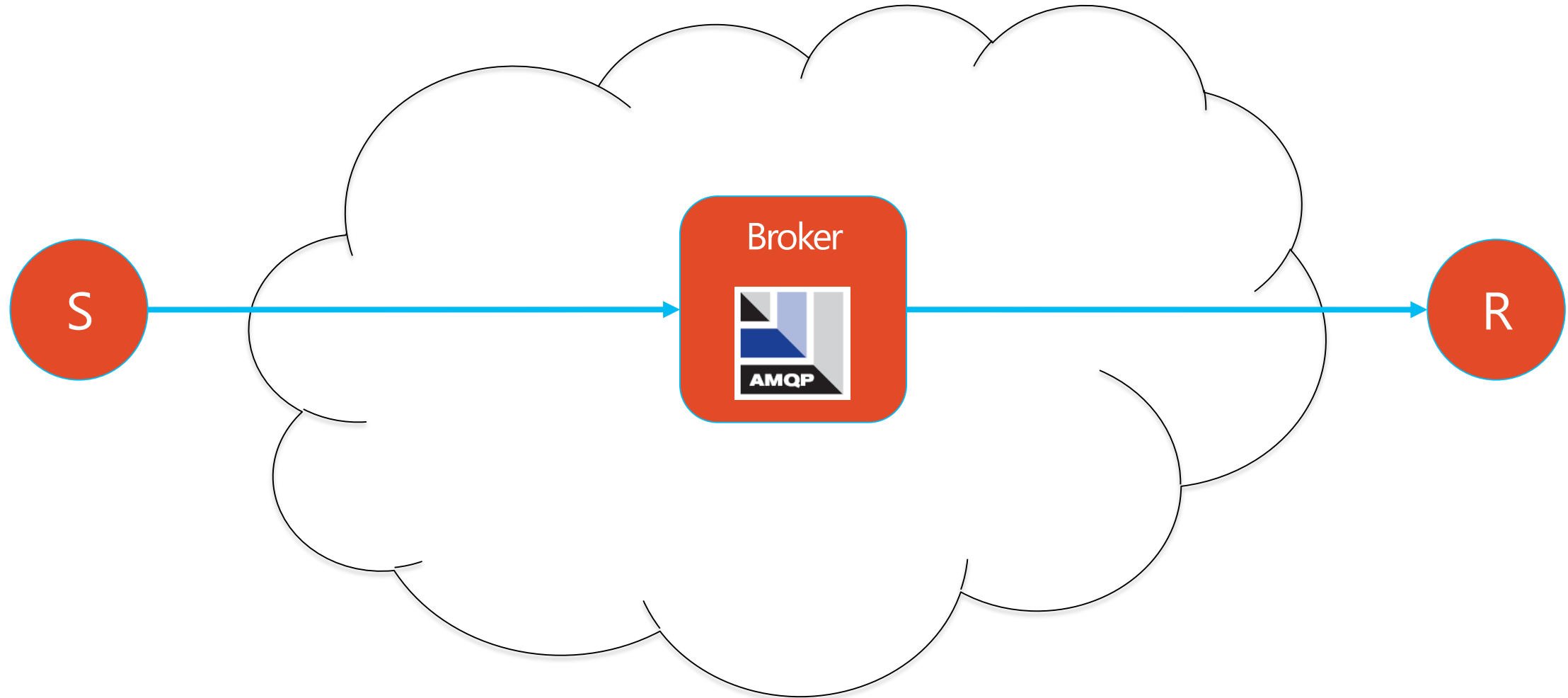Reliability
Interoperability
Ubiquity

AMQP Aware Clients
Devices & workstations

Your Enterprise

AMQP Aware Services
C/C++, Java JMS, MS WCF,
Cloud Applications

AMQP INFRASTRUCTURE

Business
Partners &
Services

© JPMorgan Chase & Co.
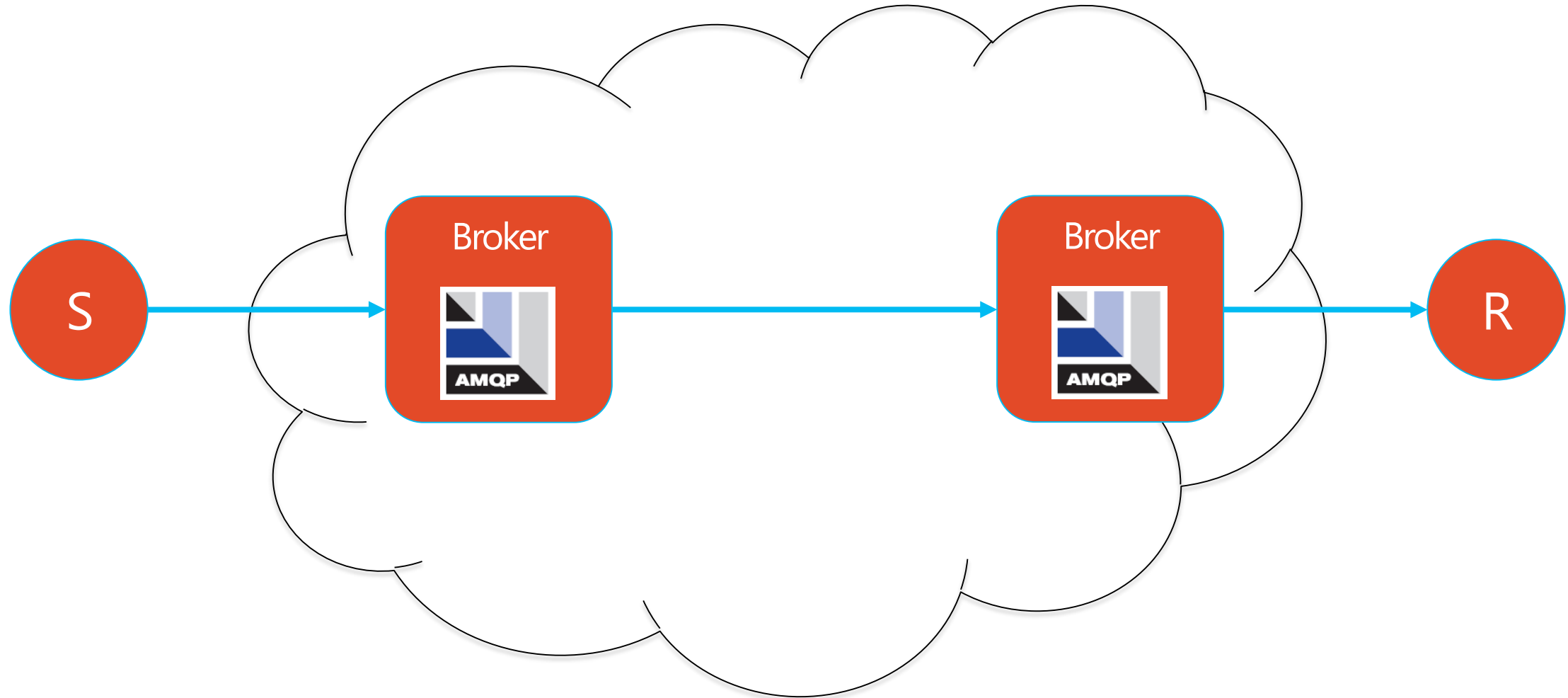
Flexible topologies

AMQP Network
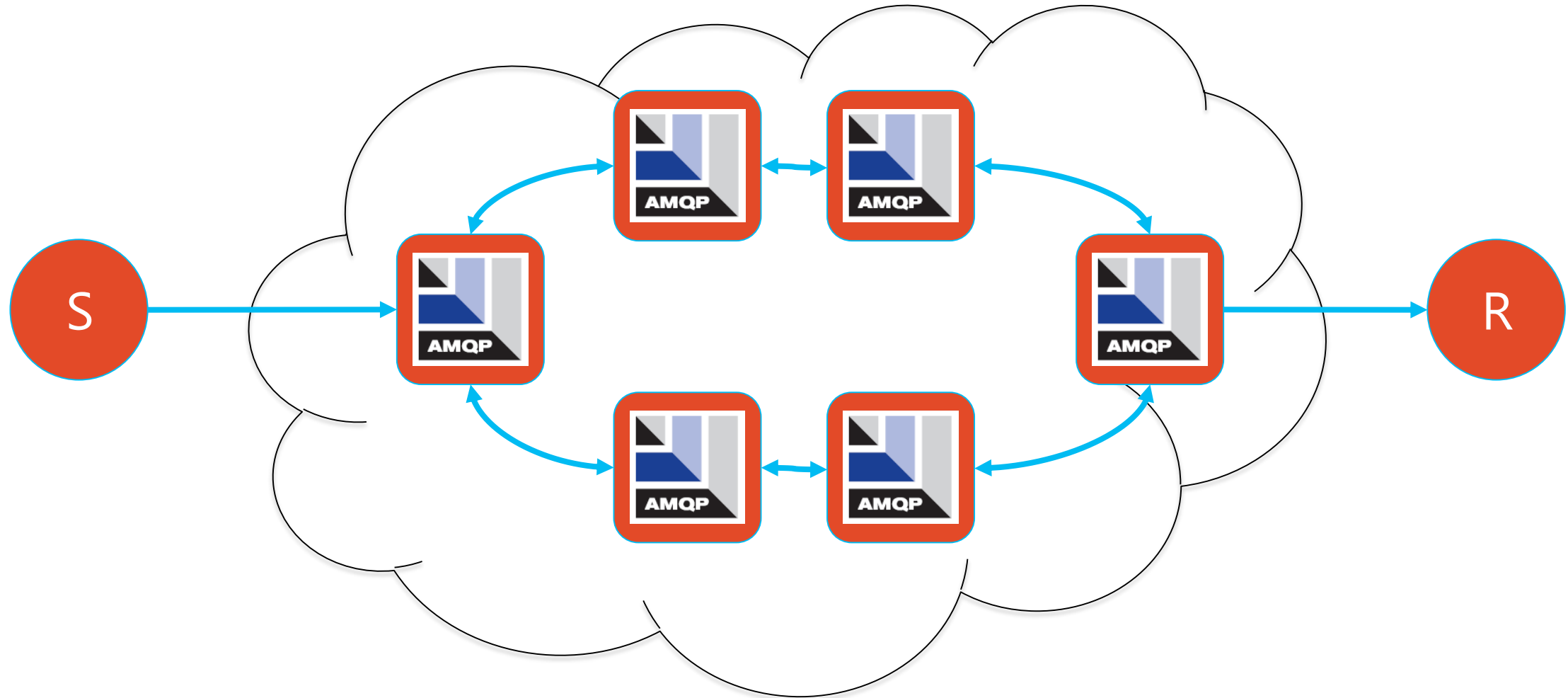
Flexible topologies

Flexible topologies

Flexible topologies

AMQP 1.0 implementations

Apache Qpid

Open Source AMQP Messaging

Brokers and Client libraries supporting AMQP

Java Broker

Supports all versions of AMQP, including AMQP 1.0 since 0.18

JMS Client

Java JMS client which works with any AMQP 1.0 compliant service

C++ Broker

AMQP 1.0 support in next release (scheduled for later this month)

Home of Proton toolkit library

Apache Qpid Proton

Proton is toolkit for speaking AMQP, including:

The AMQP Messenger API, a simple but powerful interface to send and receive messages over AMQP
The AMQP Protocol Engine, a succinct encapsulation of the full AMQP protocol machinery

Proton is designed for maximum embeddability:

Minimal dependencies
Minimal assumptions about threading model

Proton is designed to scale up and down:

Transparently supports both simple peer to peer messaging and complex globally federated topologies
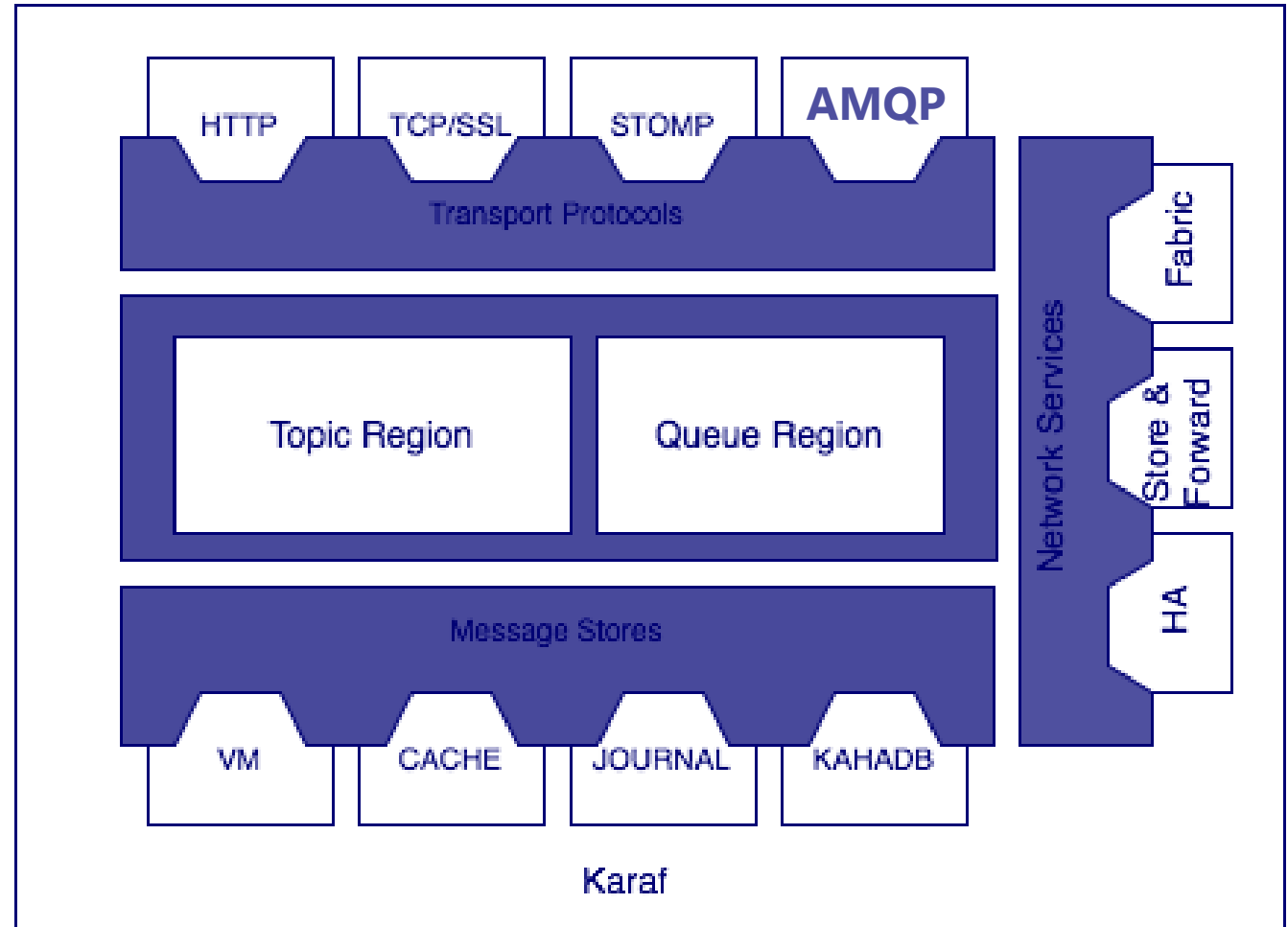
Proton is multi-lingual:

Designed for easy language bindings
Includes full fidelity data exchange: maps, lists, data structures, ...

Apache ActiveMQ

Reliable Java broker

Multi-protocol

Multi-OS

Small footprint

Pluggable architecture allows protocols and features to be added or customized

Coming soon AMQP 1.0!

SwiftMQ

Commercial JMS messaging since 2000

Based on a federated router network

Provides High Availability

First AMQP 1.0 implementation

Available since January 2012

Full-featured inc. transactions & link recovery

Fully integrated with JMS

Includes an AMQP bridge

Bridge between AMQP 1.0, AMQP 0.9.1 & JMS

Azure Service Bus

Unified set of messaging capabilities
Consistent management and observation capabilities

Service Bus Relay
Rich options for interconnecting apps across network boundaries

Service Bus Brokered Messaging
Queuing, publish/subscribe

Easily build hybrid apps

Available as PaaS & on-premise server

Supports AMQP 1.0

AMQP 1.0 client libraries

Language	Library
C#	Service Bus .NET Client Library
Java	Apache Qpid Java Message Service (JMS) client IIT SwiftMQ Java client
C	Apache Qpid Proton-C
PHP	Apache Qpid Proton-PHP
Python	Apache Qpid Proton-Python
Ruby	Apache Qpid Proton-Ruby (coming soon)
Perl	Apache Qpid Proton-Perl (coming soon)
JavaScript	Apache Qpid Proton-JavaScript (coming soon)

Demo
PictureMagic

Showcasing support for cross-platform hybrid apps using AMQP 1.0
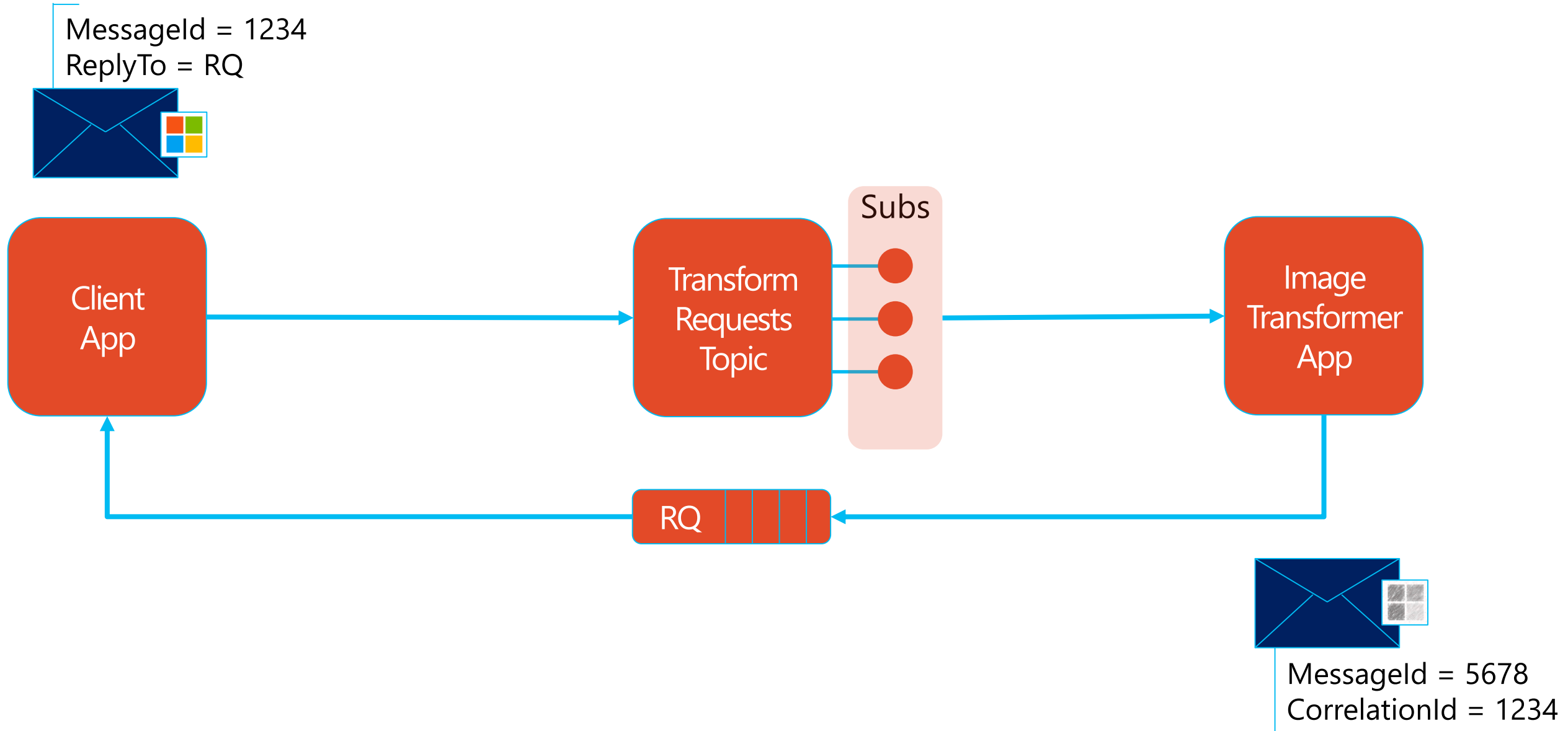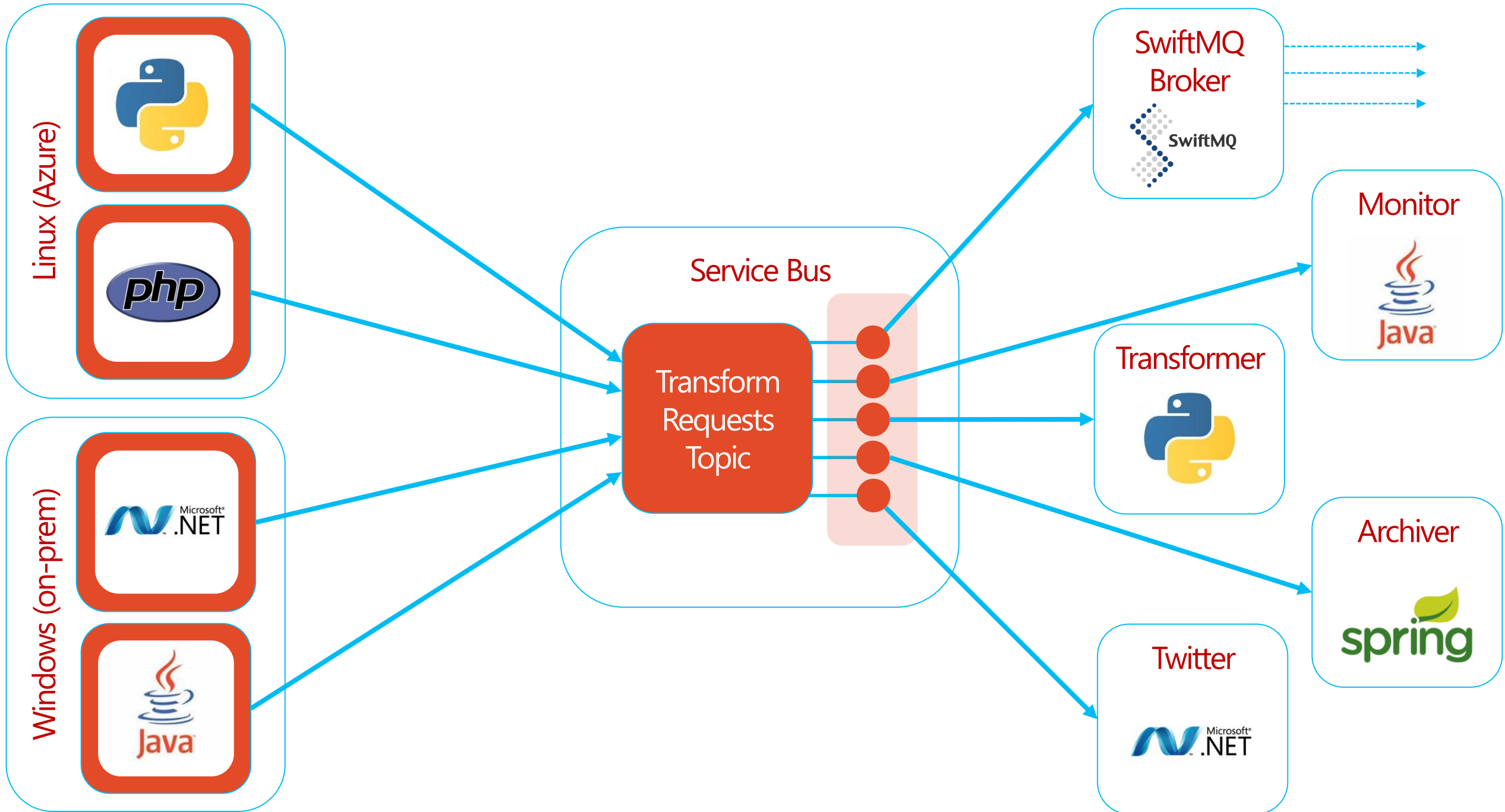
Summary

AMQP 1.0 is the OASIS Standard for messaging

Open, standard, efficient, flexible, reliable

Enables cross-platform messaging applications

Mix languages, operating systems and vendors

Business messages exchanged at full-fidelity

Brokers and clients available now:

Apache Qpid Proton, SwiftMQ, Azure Service Bus

Many more coming soon, inc. ActiveMQ

More information

AMQP Member Section at OASIS

http://www.amqp.org

Apache Qpid Java 0.18

http://qpid.apache.org/

Apache Qpid Proton

http://qpid.apache.org/proton/

Apache ActiveMQ

http://activemq.apache.org

SwiftMQ

http://www.swiftmq.com

Microsoft Windows Azure Service Bus

https://www.windowsazure.com/en-us/develop/net/how-to-guides/service-bus-amqp-overview/

Thank You. Any Questions?