Microsoft Azure Service Bus In-depth

UNDERSTANDING THE AZURE SERVICE BUS

Alan Smith
ACTIVE SOLUTION

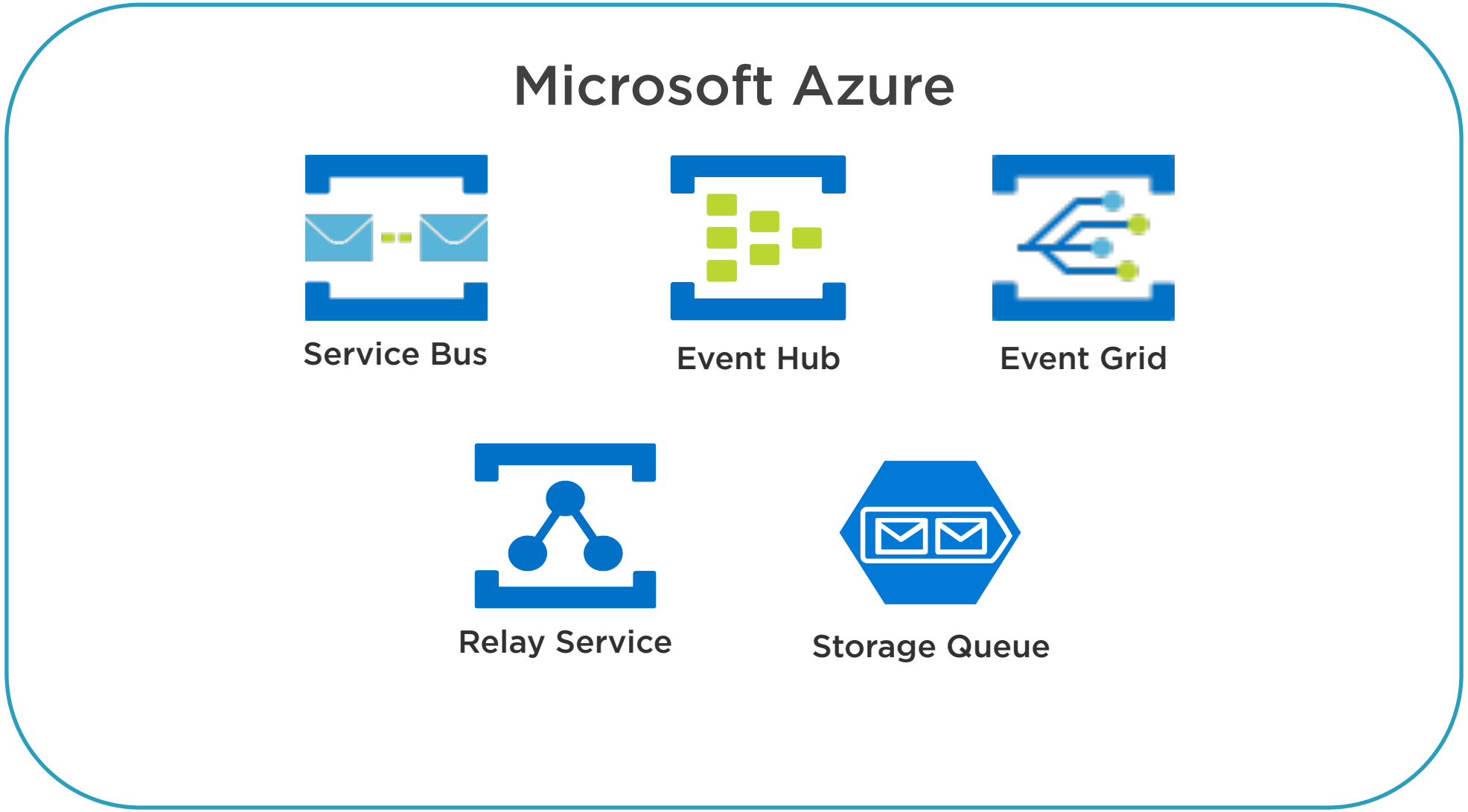
@alansmith www.cloudcasts.net

Overview

- Microsoft Azure Messaging Services
- Brokered Messaging Scenarios
- Microsoft Azure Service Bus
- The Microsoft Azure Service Bus SDK
- Demo: Simple Brokered Messaging
- Demo: Creating a Chat Client with Publish-subscribe Messaging

Microsoft Azure Messaging Services

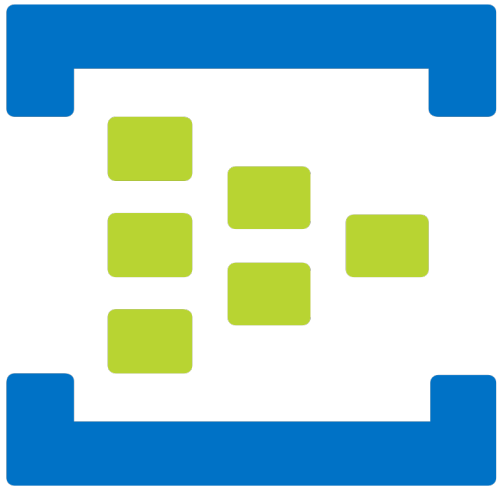
Microsoft Azure Messaging Services



Service Bus

- Durable brokered messaging
- Point-to-point messaging
- Publish subscribe messaging
- Enterprise messaging functionality
- Cost-efficient

Event Hub

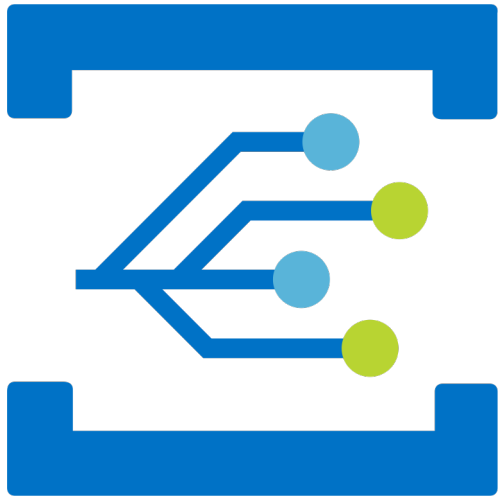


Large-scale telemetry ingestion

Buffered storage

Massively scalable

Event Grid

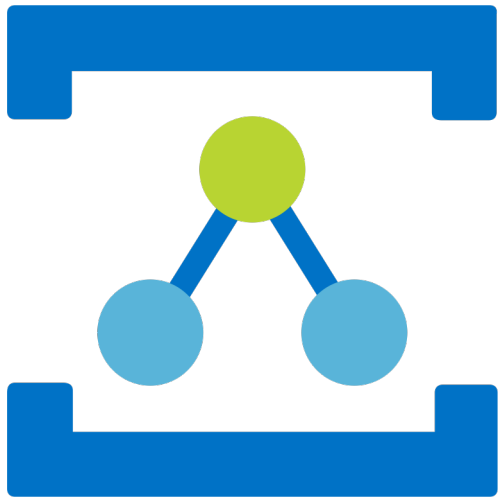


HTTP event routing and delivery

Near real-time notifications

Supported by many Azure services

Relay Service

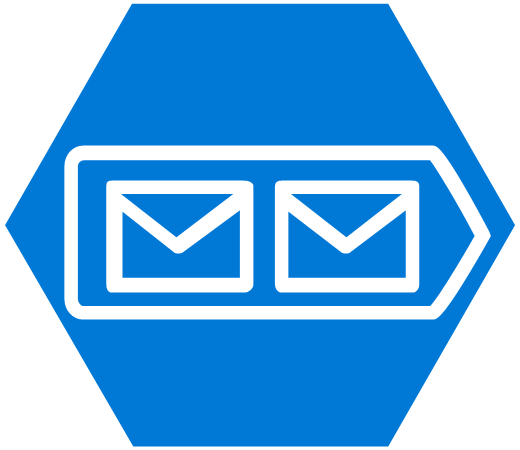


Securely exposes on-premises services

Endpoint “in the cloud”

Relays request and response calls

Storage Queues



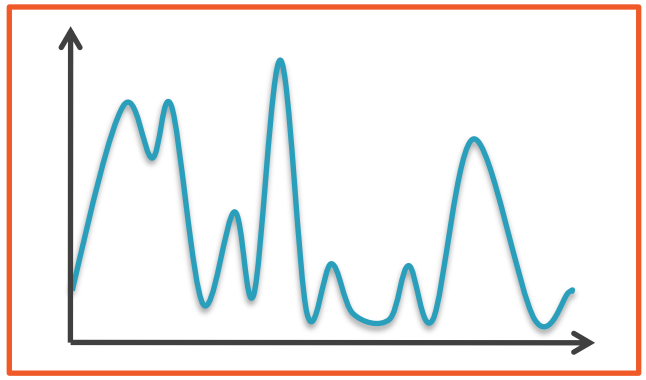
Simple point-to-point messaging

Very cost-effective

Limited functionality

Asynchronous Messaging Scenarios

Connectivity Challenges – Asynchronous Processing



Connectivity Challenges – Hybrid Systems

Public Cloud

Website

Web Job

VM

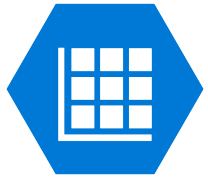


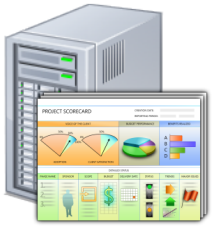
Table Storage

Blob Storage

On-Premises

Active Directory

SQL Server



LOB System

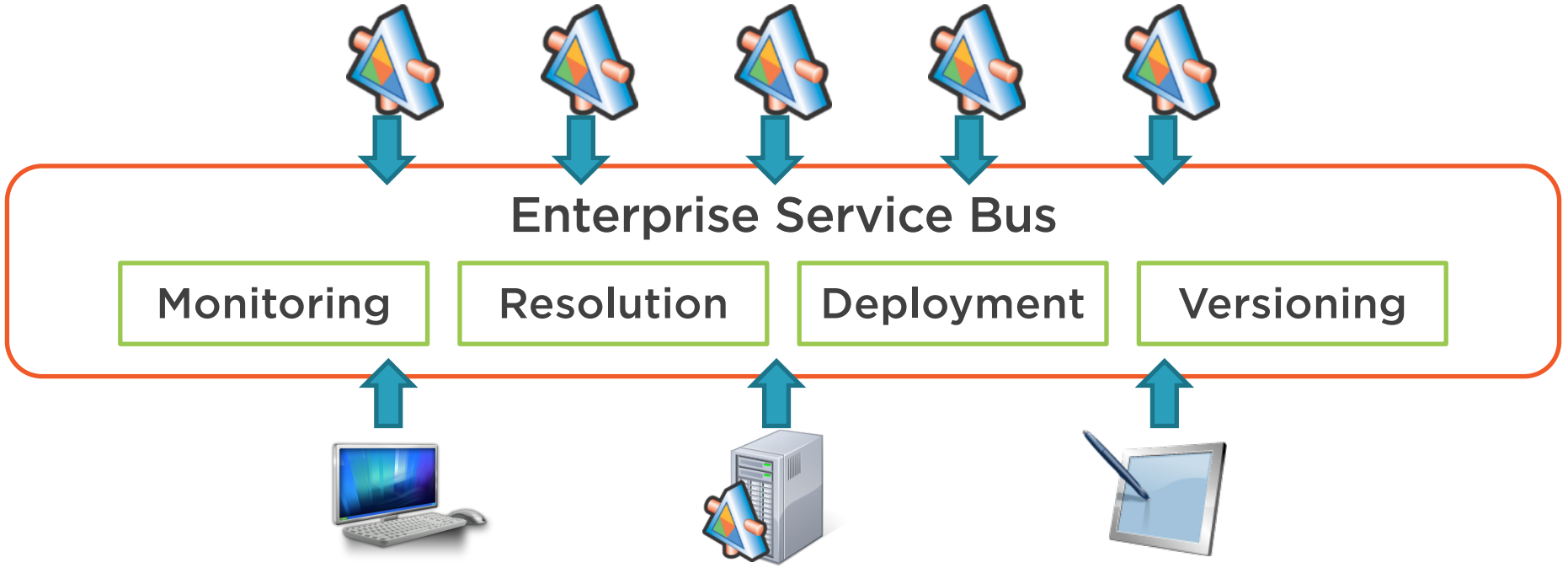
LOB System

BizTalk Server

SharePoint

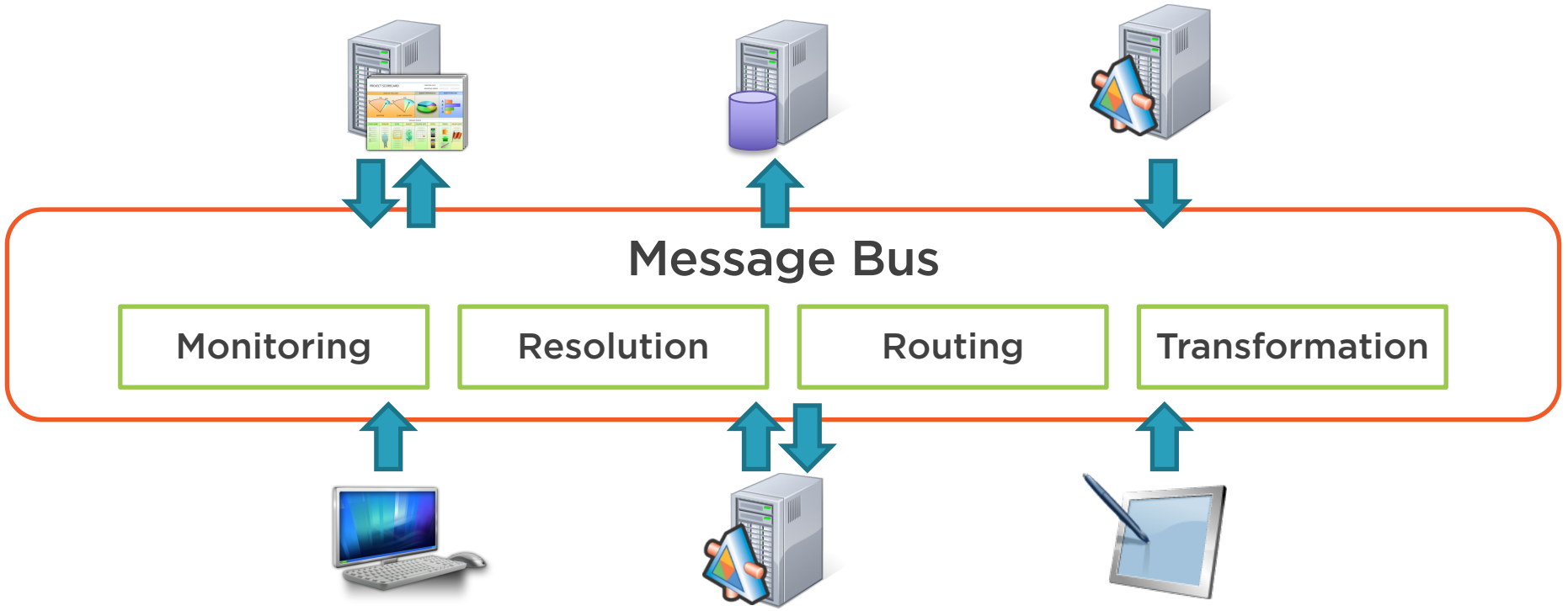
Enterprise Service Bus

“An enterprise service bus (ESB) is a software architecture model used for designing and implementing the interaction and communication between mutually interacting software applications in Service Oriented Architecture.”
- Wikipedia



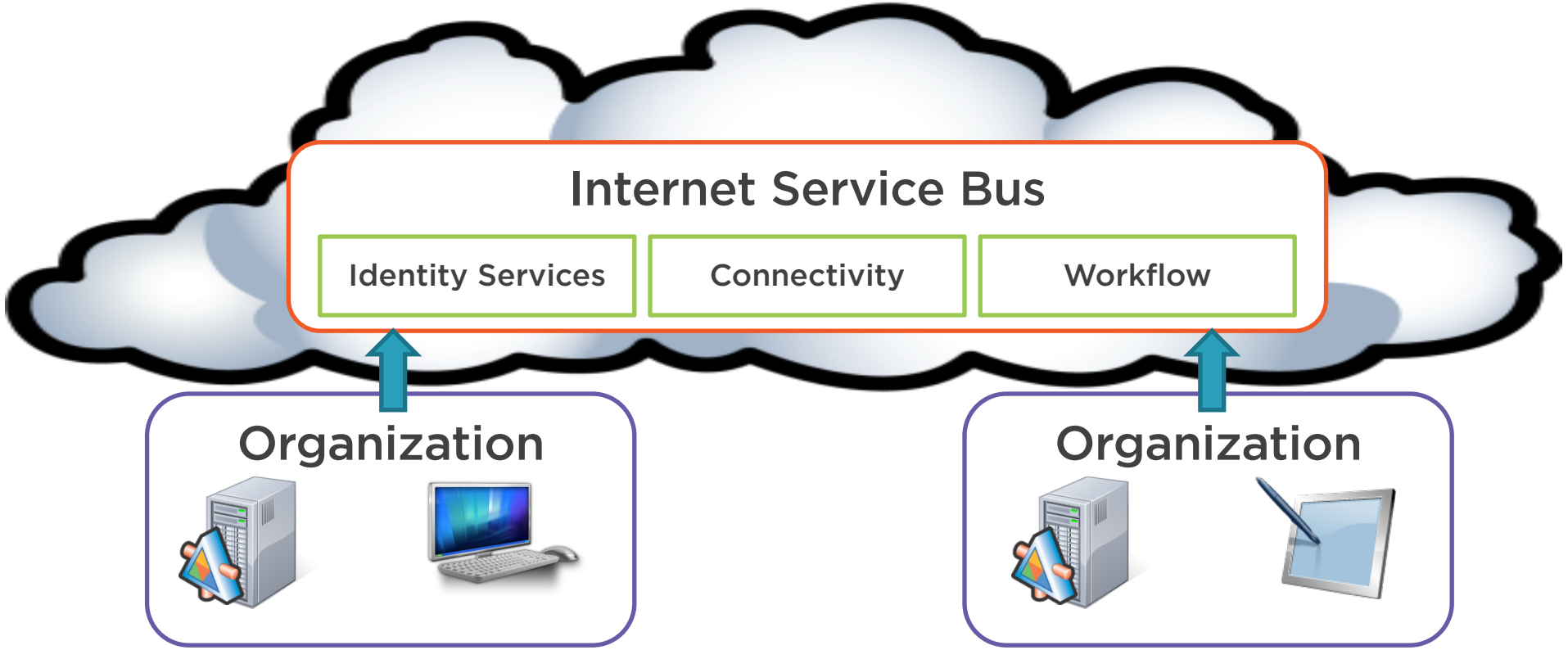
Message Bus

"A Message Bus is a combination of a common data model, a common command set, and a messaging infrastructure to allow different systems to communicate through a shared set of interfaces."
- Enterprise Integration Patterns



Internet Service Bus

"The ISB links devices to each others, devices to local servers, Web sites to Web sites, and ESBs to ESBs, and is itself an ESB. The ISB is a platform for "do-it-yourself" composite applications and business processes. The ISB is also an example of Software as a Service (SaaS)."
- The Architecture Journal, October 2007

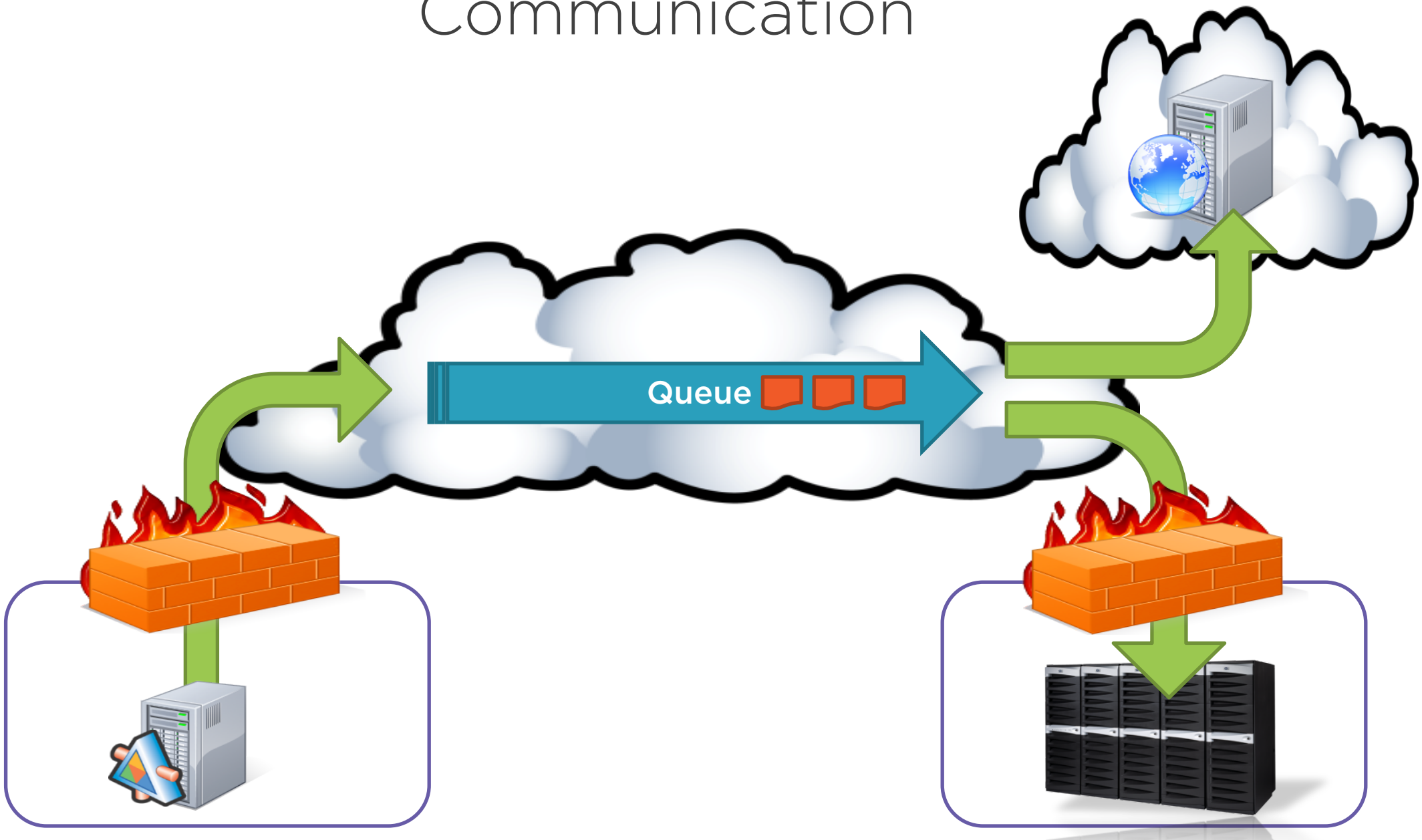


Extending to the Cloud

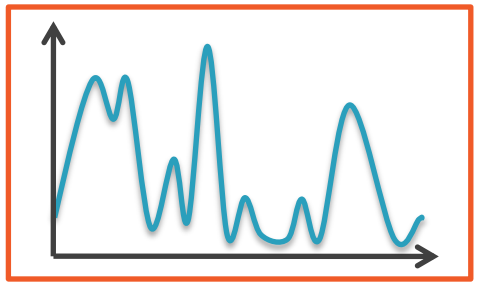
“Leveraging the capabilities provided by cloud-based platforms to enhance or augment on-premise systems.” – Alan Smith

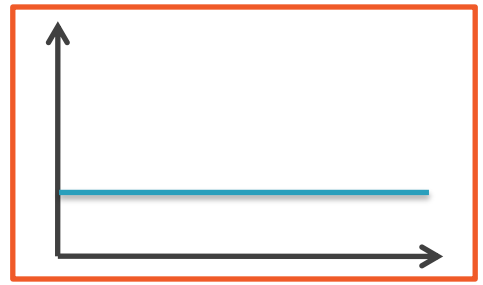
- Low entry cost
- Low risk
- Easy roll-back
- Enhance reach of existing systems

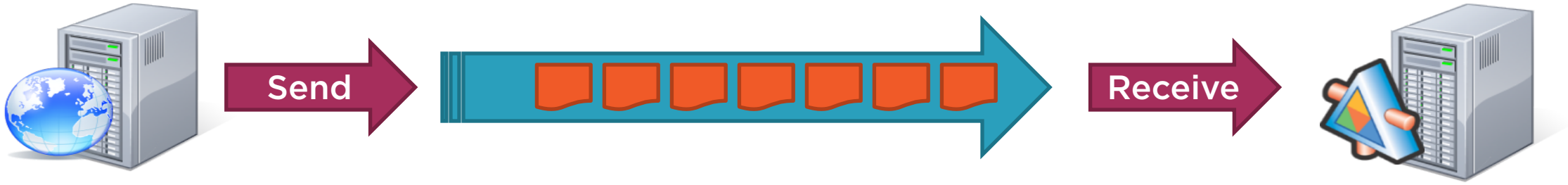
Communication



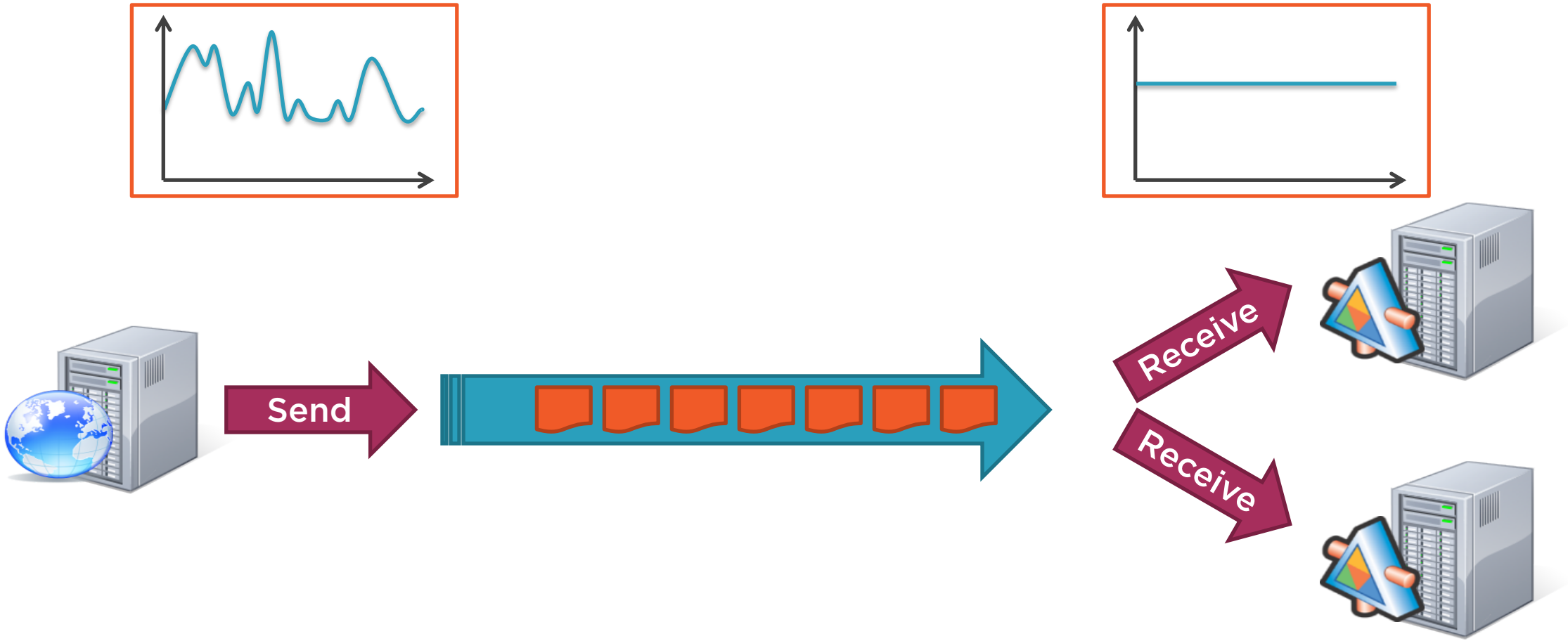
Load Leveling



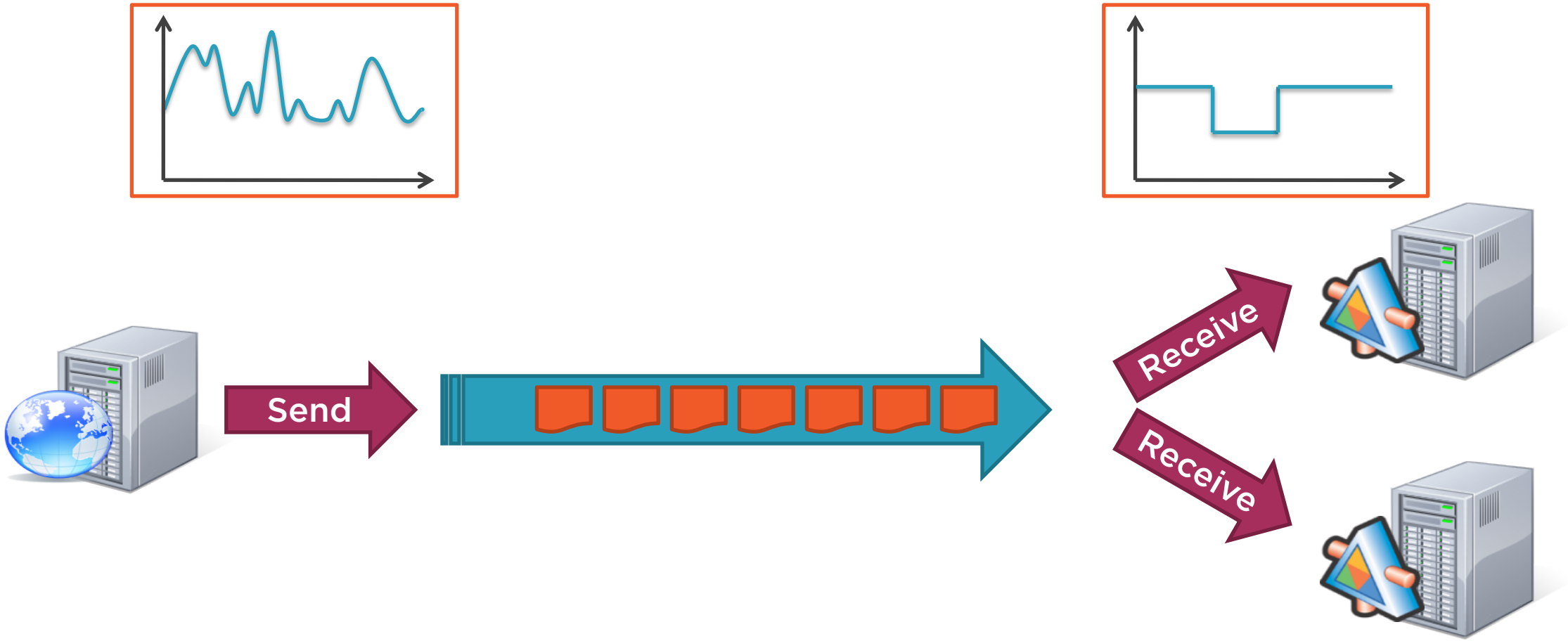




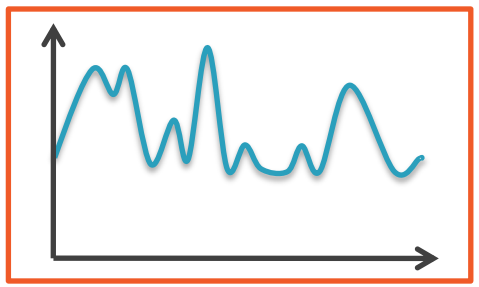
Load Balancing

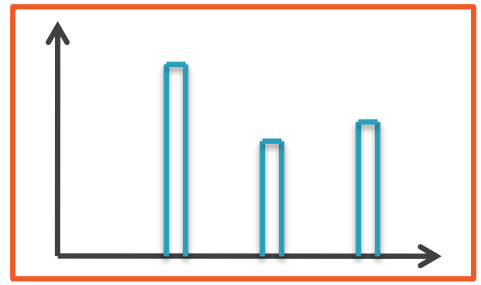


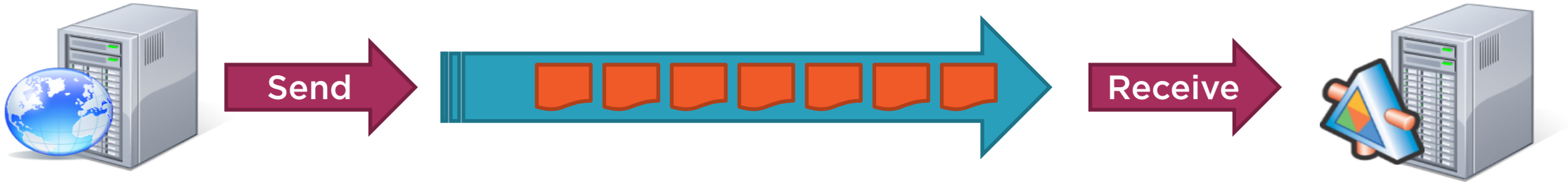
High Availability



Temporal Decoupling

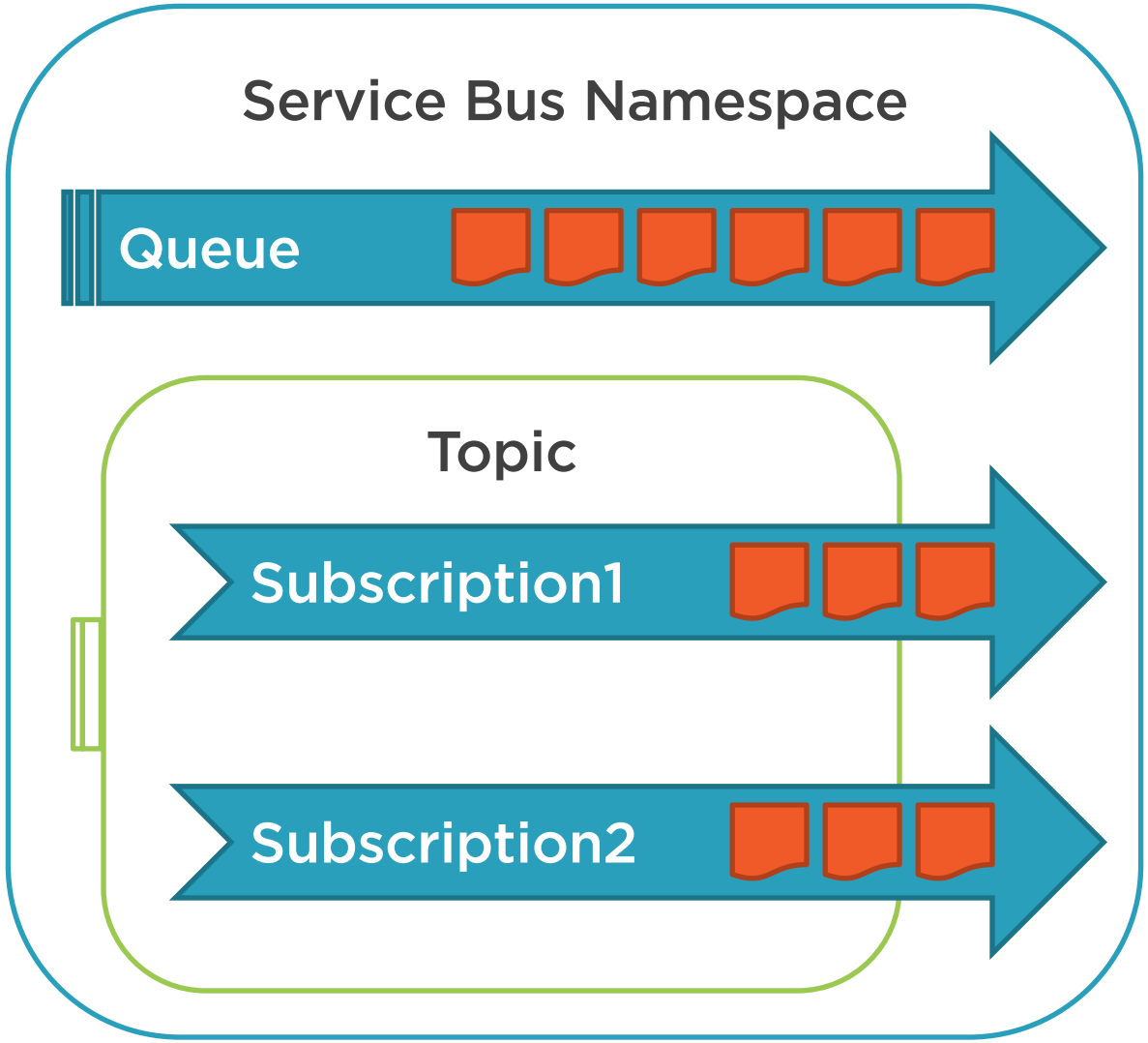






Microsoft Azure Service Bus

Messaging Entities



Namespace

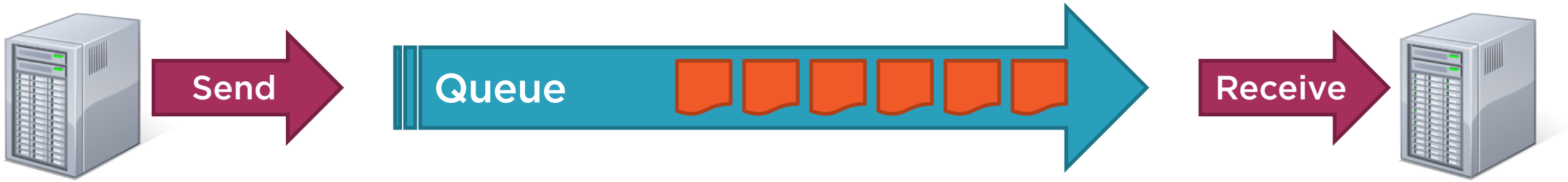
Queue

Topic

Subscription

Message

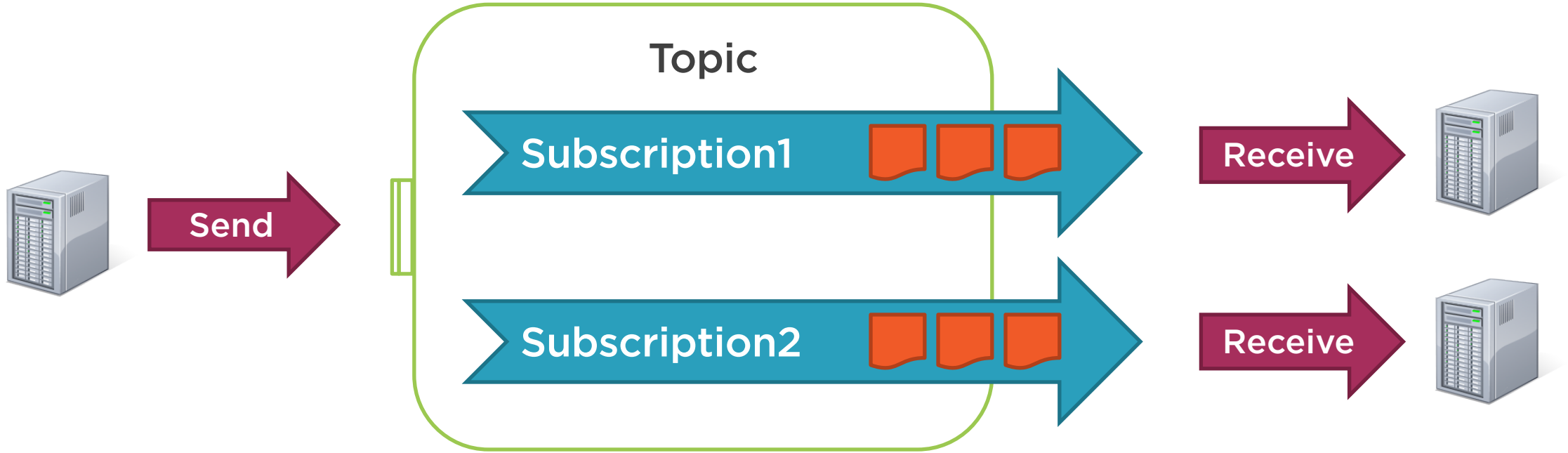
Brokered Messaging – Queues



Point-to-point messaging

First-in first-out (FIFO) processing

Brokered Messaging – Topics and Subscriptions



Publish-subscribe messaging

Messages are sent to topics

Messages are received from subscriptions

Filters can determine message subscription

Enterprise Messaging Capabilities

Capability	Description
Communication	How can applications in different environments communicate effectively and reliably with one another?
Security	How can the confidentiality and integrity of messages be maintained?
Reliable Delivery	How can the sending application be sure that the receiving application will receive all transmitted messages?
Low-Latency	How can the transmission and processing time of messages be kept as low as possible?
Availability	What level of uptime will a messaging system provide?
Scalability	How easily can a messaging system be upgraded to handle an increase in the message processing load?

Service Bus Brokered Messaging Features

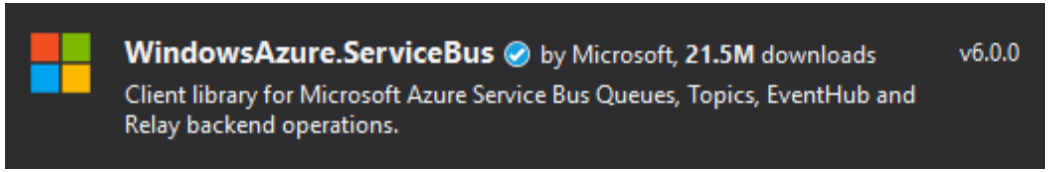
Feature	Description
Publish-subscribe	Messages can be broadcast to multiple receivers based on routing rules in the messaging entities.
Dead-lettering	Invalid or poison messages can be moved to a dead-letter queue.
Message Sessions	Related messages can be grouped together in sessions and processed together.
Request-response Correlation	Response messages can be correlated with the appropriate request messages to allow for asynchronous two-way communication.
Message Deferral	Messages can be preserved on a messaging entity and retrieved later for processing.
Scheduled Enqueue Time	Messages can be sent to a messaging entity and then enqueued at a specified time.
Duplicate Detection	Duplicate messages can be ignored by a messaging entity.
Message Expiration	Messages can be configured to expire after a specified duration.

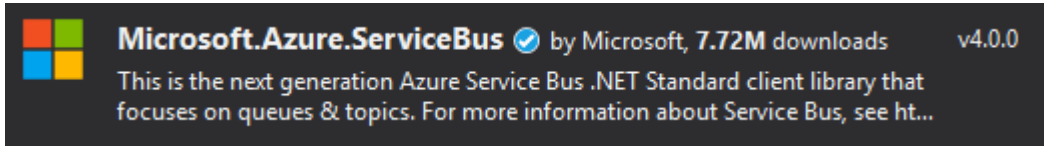
Service Bus Protocols

Protocol	Usage
AMQP	Open messaging protocol. Default protocol used by Service Bus SDK. Supported by many applications and libraries.
HTTP	Provides communication where firewalls may limit connectivity on other protocols. Cross platform compatibility with many clients.

The Microsoft Azure Service Bus SDK

Azure Service Bus NuGet Packages





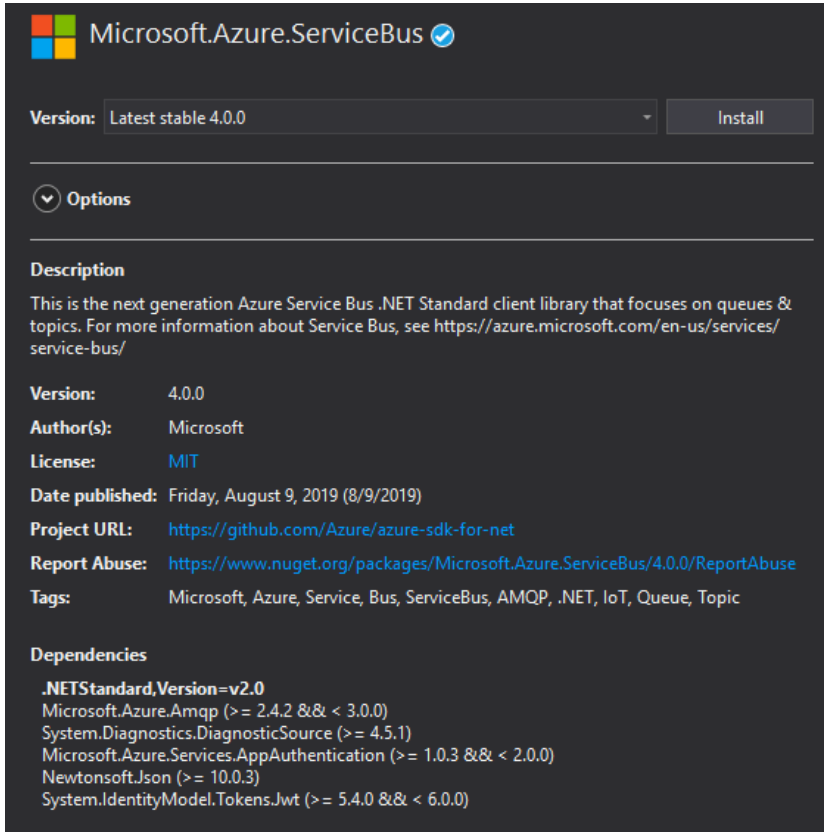
WindowsAzure.ServiceBus

- Old Service Bus SDK
- Supports Queues, Topics and Subscriptions, Event Hubs and Relay
- .NET message serialization by default

Microsoft.Azure.ServiceBus

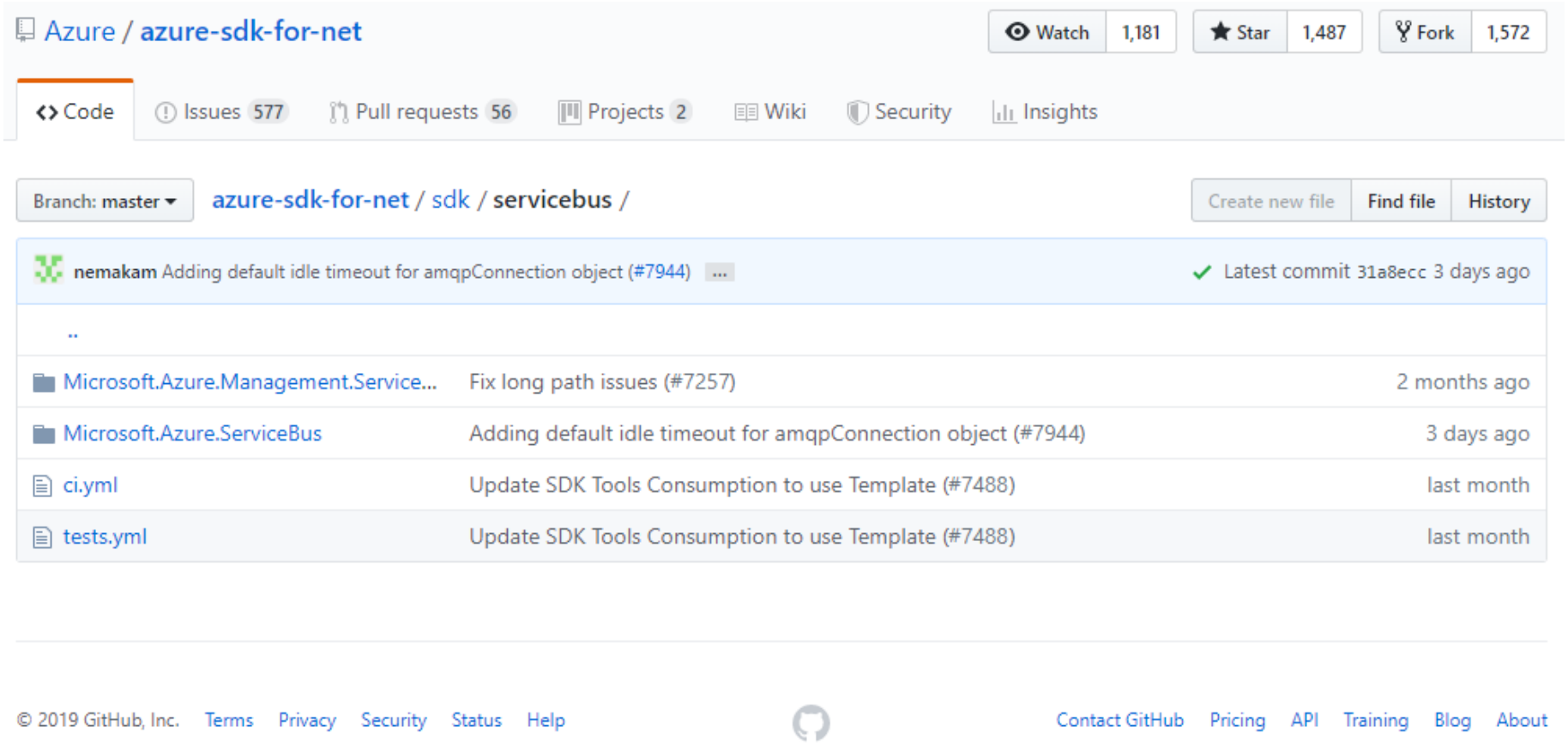
- Latest Service Bus SDK
- Supports Queues, Topics and Subscriptions
- Use other SDKs for Event Hubs and Relay
- Binary message serialization by default

Microsoft Azure Service Bus SDK




```
PM> Install-Package Microsoft.Azure.ServiceBus
```

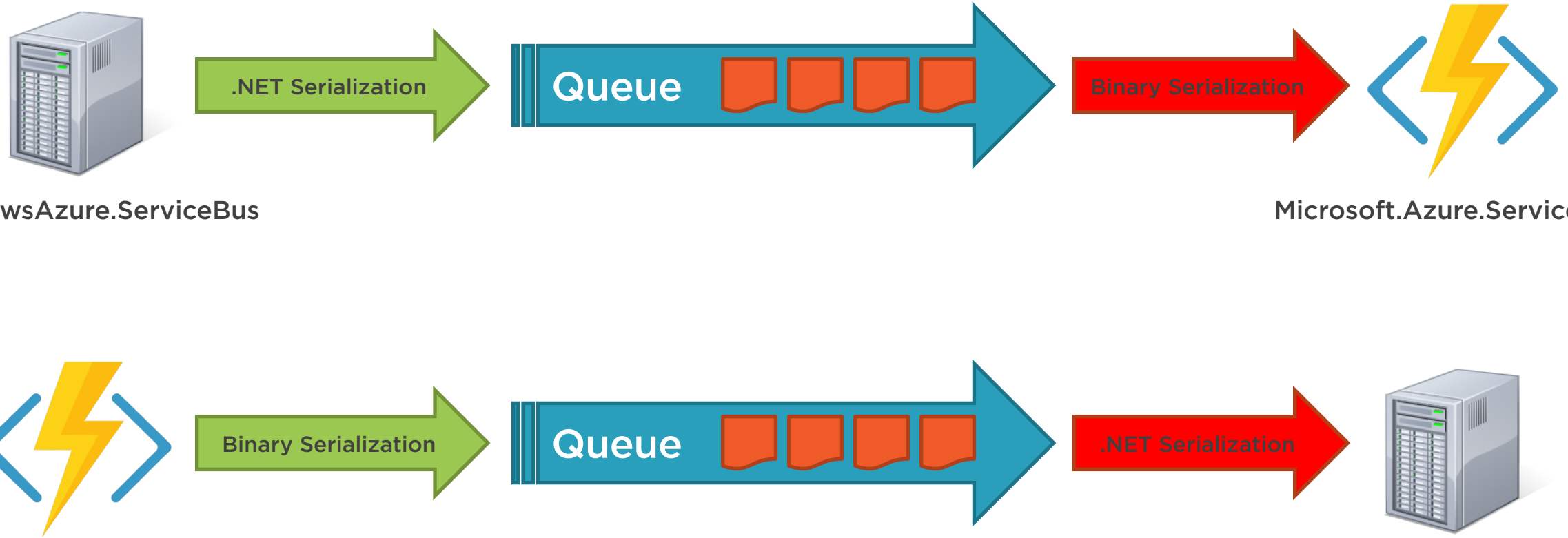
Service Bus SDK on GitHub



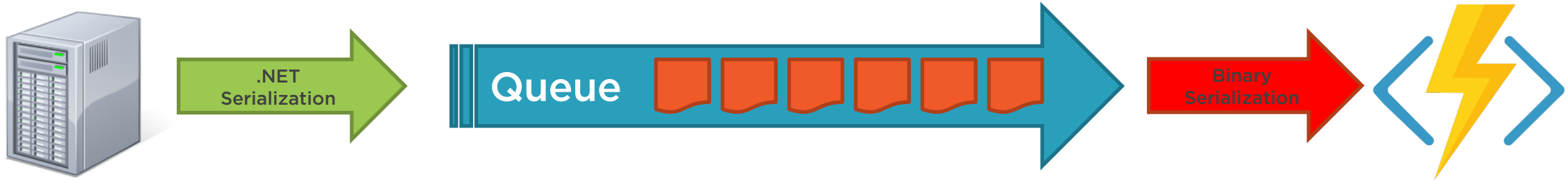
https://github.com/Azure/azure-sdk-for-net/tree/master/sdk/servicebus

Message Serialization Compatibility

Message Serialization Compatibility

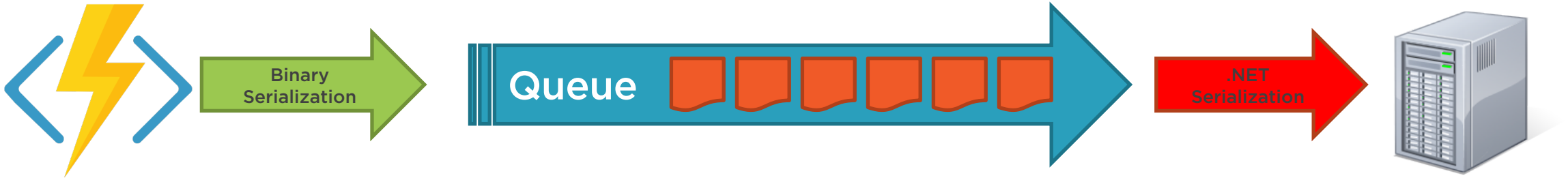


Message Serialization Compatibility



WindowsAzure.ServiceBus

Microsoft.Azure.ServiceBus



Microsoft.Azure.ServiceBus

WindowsAzure.ServiceBus

Commonly Used Classes

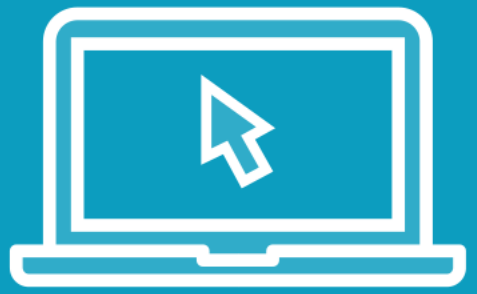
Class	Description
ManagementClient	Used to manage messaging entities within a service bus namespace.
QueueClient	Used by client applications to send and receive messages from a service bus queue.
TopicClient	Used by a client to send messages to a service bus topic.
SubscriptionClient	Used by a client to receive messages from a service bus subscription and to manage filter subscription rules.
Message	Used to represent a message transmitted through the service bus.

Demo

Simple Brokered Messaging

- Creating a Service Bus Namespace
- Adding the Service Bus NuGet package
- Creating and sending messages
- Receiving and processing messages

Demo



Simple Chat Application

- Publish and subscribe messaging with topics and subscriptions

Summary

- Azure Service Bus is one of a number of Azure messaging services
- Azure Service Bus provides enterprise class messaging capabilities
- Service Bus namespaces can contain Queues and Topics
- Topics can contain Subscriptions
- Queues provide point-to-point messaging
- Topics and Subscriptions provide publish-subscribe messaging
- Microsoft.Azure.ServiceBus .NET SDK is available on NuGet and GitHub