Azure Service Bus - Message Queues

By
Narasimha Rao T
Microsoft.Net FSD Trainer
tnrao.trainer@gmail.com

What is Azure Service Bus?

- Azure Service Bus is a **fully managed enterprise message broker** offered by Microsoft Azure.
- It enables **asynchronous messaging** between applications and services.
- It is allowing them to communicate **even if they are not running at the same time**.

Key Features of Azure Service Bus

Feature	Description
Queues and Topics	Supports both point-to-point (Queues) and publish-subscribe (Topics) models
Message Sessions	Support for FIFO and session-based messaging
Dead-lettering	Captures messages that cannot be delivered or processed
Duplicate Detection	Avoids sending the same message more than once
Scheduled Delivery	Send messages to be delivered at a later time
Auto-forwarding	Forward messages from one entity to another automatically

Advantages of Using Azure Service Bus

- **Reliable communication** between apps and services
- **Decouples** the sender and receiver (no need to be online simultaneously)
- **Scales easily** for enterprise-grade workloads
- **Built-in security** with Role-Based Access Control (RBAC) and Shared Access Signatures (SAS)
- **Integration-friendly** with Azure Functions, Logic Apps, Event Grid, etc.

Understanding Azure Service Bus Queues

- **Point-to-point communication**
- One sender, one receiver
- Receiver processes messages **in order** (FIFO)
- Messages are stored in the queue until received and processed
- Ideal for **task scheduling**, **load leveling**, or **message buffering**

Example Use Case:

An e-commerce app sends order data to a queue; a backend service picks up and processes each order.

Understanding Topics and Subscriptions

- **Publish-Subscribe model**
- **Topics** act like a single access point for publishing messages
- **Subscriptions** receive messages selectively based on filters
- Allows **one-to-many** message delivery

Example Use Case:

An event (like "new user registered") is published once and consumed by multiple systems: welcome email service, analytics system, and CRM integration.

Service Bus topics and queues

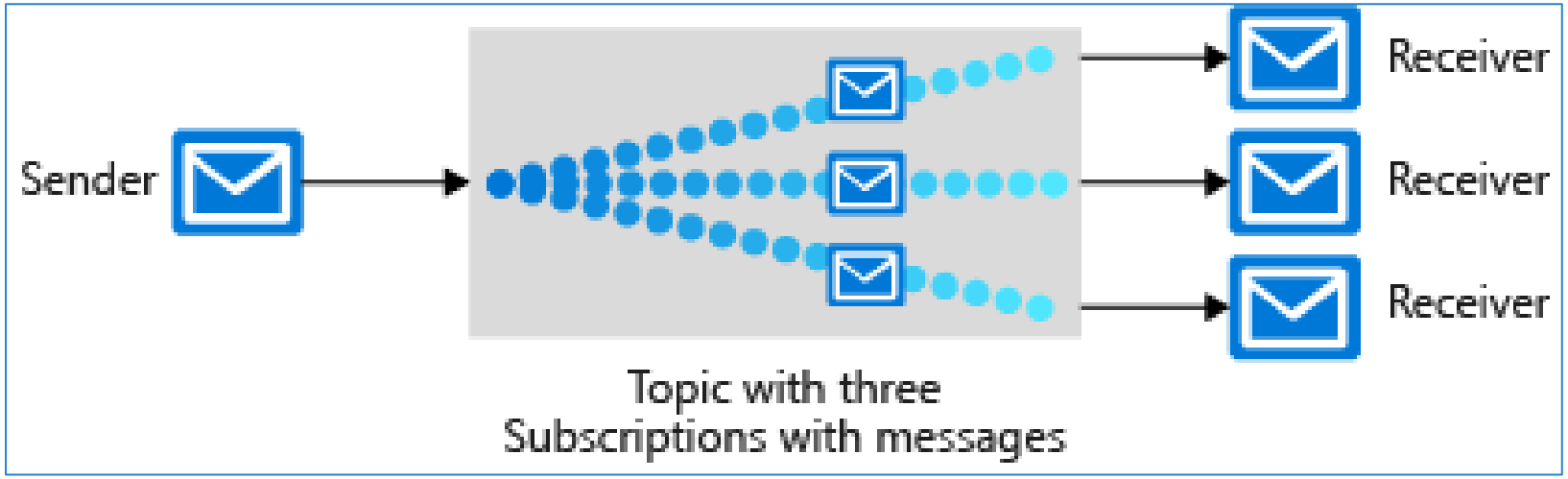
Queue:

- A Service Bus *queue* is a simple temporary storage location for messages.
- A sending component adds a message to the queue.
- A destination component picks up the message at the front of the queue. Under ordinary circumstances, each message is received by only one receiver.

Service Bus topics and queues

Topic:

- A Service Bus *topic* is similar to a queue, but a topic can have multiple subscriptions, which means that multiple destination components can subscribe to a specific topic, so each message is delivered to multiple receivers.



Common Tasks with Azure Service Bus

Task	Description
Create a Namespace	Logical container for queues/topics
Create Queues or Topics	Define the messaging infrastructure
Send Messages	Apps or services push messages to queues/topics
Receive Messages	Workers consume and process messages

Basic Messaging Flow

1. **Sender Application** sends a message to a queue or topic
2. **Service Bus** stores the message securely
3. **Receiver Application** retrieves the message asynchronously
4. Message is **processed and removed** from the queue (or dead-lettered if failed)

Summary

- Azure Service Bus is a powerful, reliable, and flexible messaging platform.
- It supports various enterprise communication patterns using **Queues and Topics**.
- It enables asynchronous workflows, decouples services, and integrates seamlessly with Azure's serverless and automation tools.

Hands-On

Part-1: Creatig Azure Service Bus, Queue and Topics

Hands-On

Part-2 : Sending and Receiving messages using C# through Azure Service Bus

Q & A