

AZURE DEVELOPMENT FUNDAMENTALS (1)

Mai Hoang Giang

Agenda

I. Azure Service Bus

1. What's Azure Service Bus
2. What's queue
3. What's topic
4. Examples

II. Azure function

1. Some basic terms
2. What is azure function
3. Some concepts: Trigger and binding, Dependency injection
4. Examples and Tom scenarios

III. Durable Function

IV. Discussion

Mai Hoang Giang



Azure Service Bus

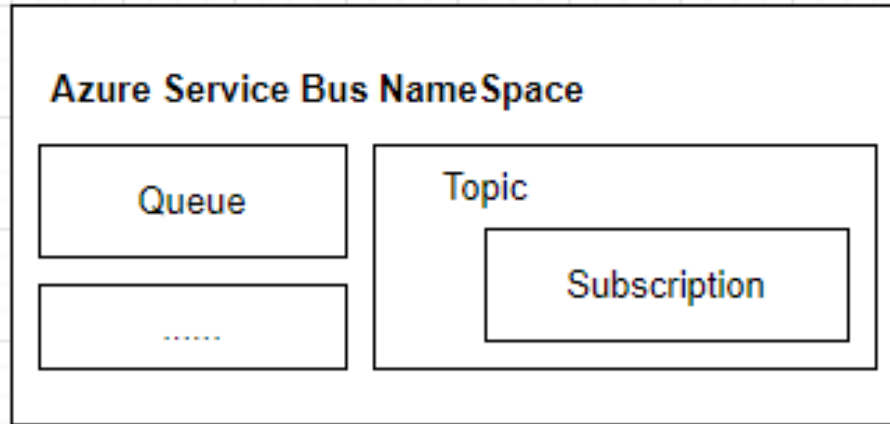
Mai Hoang Giang

I. Azure Service Bus – What's it?

- **Message broker**
- **Can decouple applications and services**
- **Some common scenarios:**
 - Messaging: Transfer business data
 - Decouple application: client and server don't have to be online at the same time.
 - Topic and subscriptions: enable 1:n relationships between publisher and subscriber.
 - Message session: implement workflows that require message ordering or message deferral.

I. Azure Service Bus – What's namespace?

- Is a container for all messaging component.
- Multiple queues and topics can be in a single namespaces

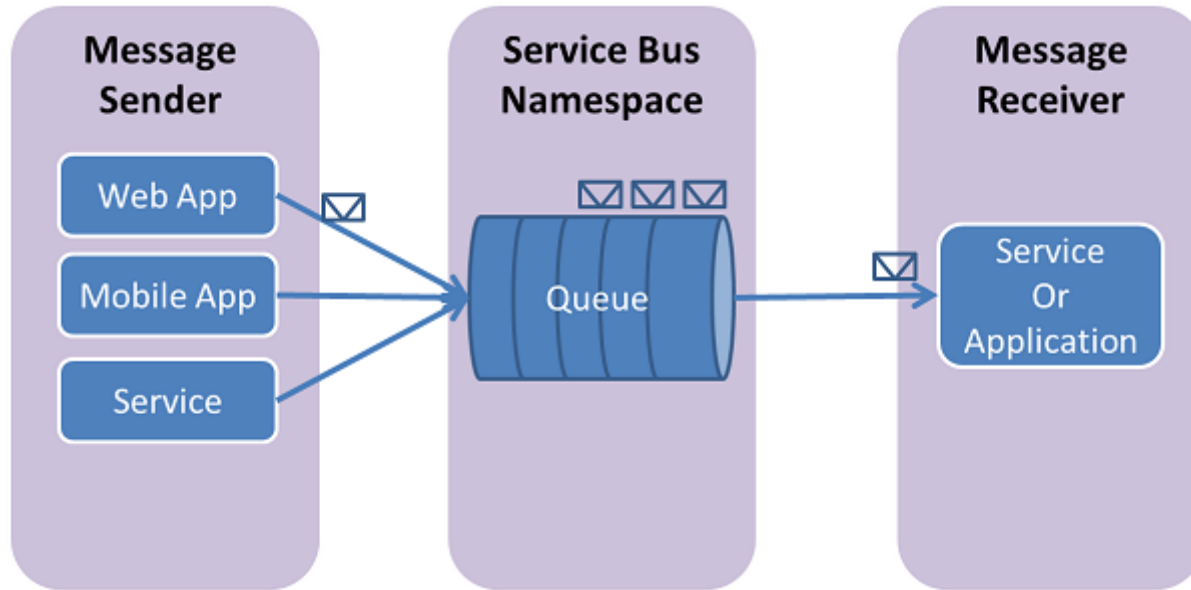


I. Azure Service Bus – What's Queues?

- FIFO
- Often used for point-to-point communication

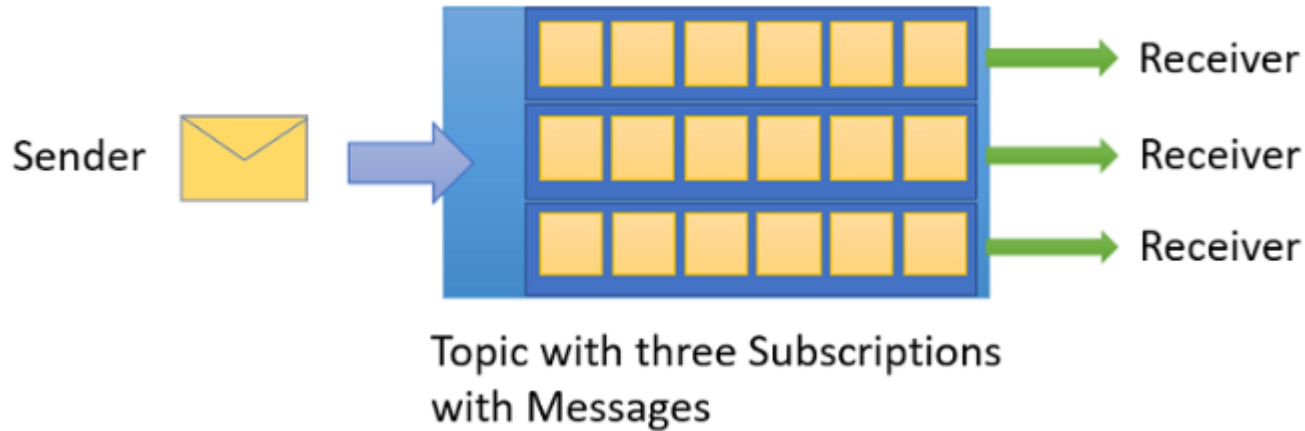


I. Azure Service Bus – What's Queues?

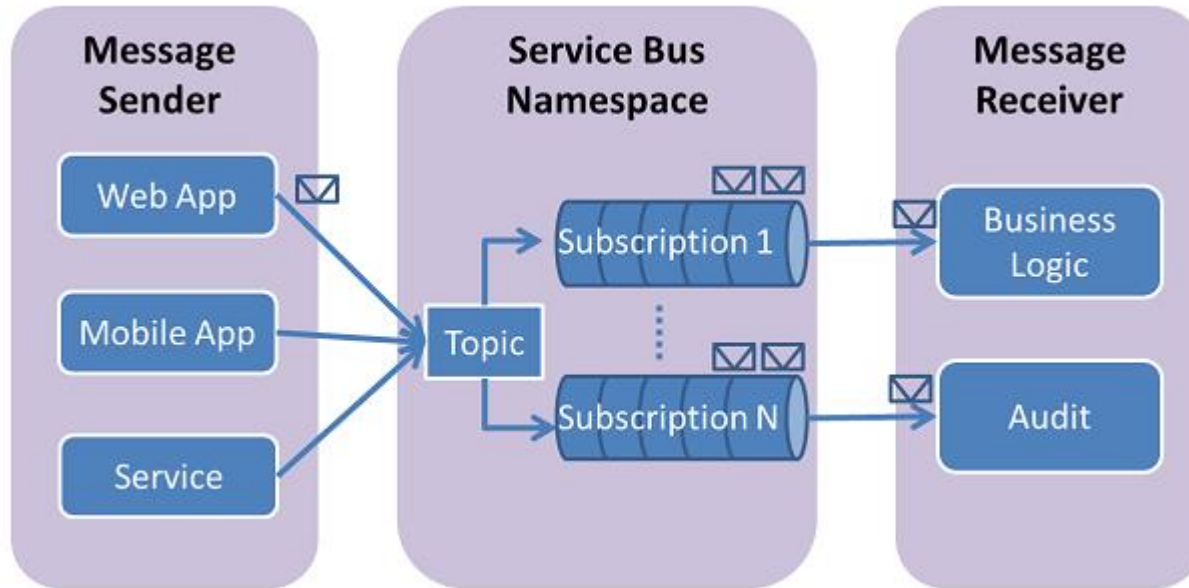


I. Azure Service Bus – What's Topic?

- Topics and subscriptions
- Useful in publish/subscribe scenario
- Topics can have multiple, independent subscriptions



I. Azure Service Bus – What's Topic?



I. Azure Service Bus – Example

- **Create Service Bus Namespace**
- **Create Queue**
- **Create Topic**

Mai Hoang Giang

I. Azure Service Bus – Example

- Send and receive message with Queue
- Send and receive message with Topic
- Reference Links:
- <https://docs.microsoft.com/en-us/azure/service-bus-messaging/service-bus-messaging-overview#queues>

Mai Hoang Giang



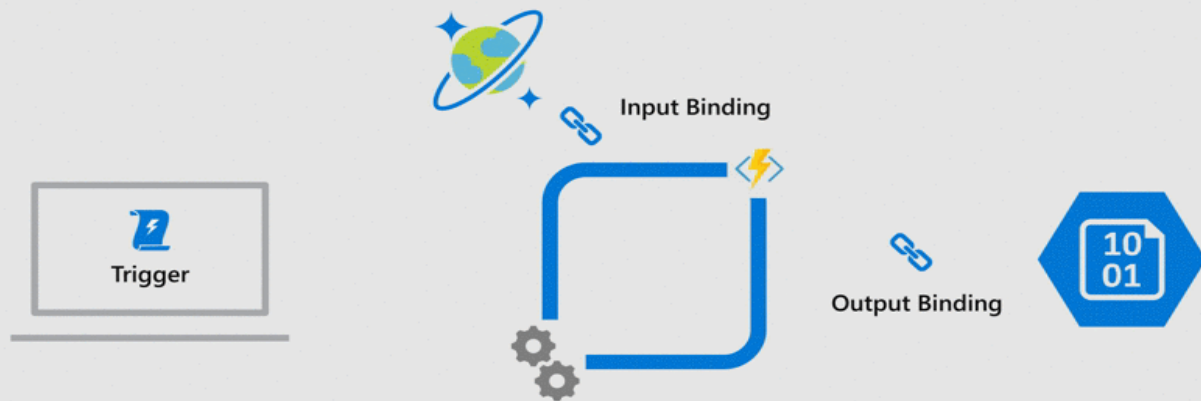
Azure Function

Mai Hoang Giang

II. Azure function – what's it?

- Allow us run a piece of code/ don't worry about infrastructure (Cloud infrastructure provide all, application scale)
- Serverless application (PaaS).
- Pay per use pricing model. (Consumption plan, Premium plan, and Dedicated (App Service) plan)
 - <https://docs.microsoft.com/en-us/azure/azure-functions/functions-scale>
- A function is triggered by specific event type

II. Azure function – Triggers and Bindings



Mai Hoang Giang

II. Azure function – Triggers

- Are what cause a function run.
- A function must have exactly one trigger
- Have associated data, as a payload of a function
- Some type of trigger: Http Trigger, Service Bus Trigger, Event Grid Trigger...

II. Azure function – Bindings

- Is a way of declaratively connecting another resource.
- Can be input bindings or output bindings.
- Data from binding is considered as parameters of a function.
- Bindings are optional.

II. Azure function – Trigger and binding definitions

Platform	Triggers and bindings are configured by...
C# class library	decorating methods and parameters with C# attributes
All others (including Azure portal)	updating <code>function.json</code> (schema)

C#

```
public static class QueueTriggerTableOutput
{
    [FunctionName("QueueTriggerTableOutput")]
    [return: Table("outTable", Connection = "MY_TABLE_STORAGE_ACCT_APP_SE")]
    public static Person Run(
        [QueueTrigger("myqueue-items", Connection = "MY_STORAGE_ACCT_APP_")]
        ILogger log)
    {
        return new Person() {
            PartitionKey = "Orders",
            RowKey = Guid.NewGuid().ToString(),
            Name = order["Name"].ToString(),
            MobileNumber = order["MobileNumber"].ToString() };
    }
}
```

Mai Hoang Giang

II. Azure function – Supported bindings

This table shows the bindings that are supported in the major versions of the Azure Functions runtime:

Type	1.x	2.x and higher ¹	Trigger	Input	Output
Blob storage	✓	✓	✓	✓	✓
Azure Cosmos DB	✓	✓	✓	✓	✓
Dapr ³		✓	✓	✓	✓
Event Grid	✓	✓	✓		✓
Event Hubs	✓	✓	✓		✓
HTTP & webhooks	✓	✓	✓		✓
IoT Hub	✓	✓	✓		✓

Mai Hoang Giang

II. Azure function – Supported bindings (2)

Kafka ²		✓	✓		✓
Mobile Apps	✓			✓	✓
Notification Hubs	✓				✓
Queue storage	✓	✓	✓		✓
RabbitMQ ²		✓	✓		✓
SendGrid	✓	✓			✓
Service Bus	✓	✓	✓		✓
SignalR		✓		✓	✓
Table storage	✓	✓		✓	✓
Timer	✓	✓	✓		
Twilio	✓	✓			✓

II. Azure function – HttpTrigger

- Reference link: <https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-http-webhook>
- Let you invoke a function via http request
- Attributes and annotations
 - “HttpTrigger”: to set authorization level, http methods, route template
- **Payload**
 - HttpRequest: full access to the request object.
 - A custom type: the runtime tries to parse json request body to set the object properties.

```
[FunctionName("HttpTriggerCSharp")]  
public static Task<IActionResult> Run(  
    [HttpTrigger(AuthorizationLevel.Anonymous)] HttpRequest req)  
{  
    ...  
}
```

II. Azure function – HttpTrigger Example

1/ Sample Httptrigger

Mai Hoang Giang

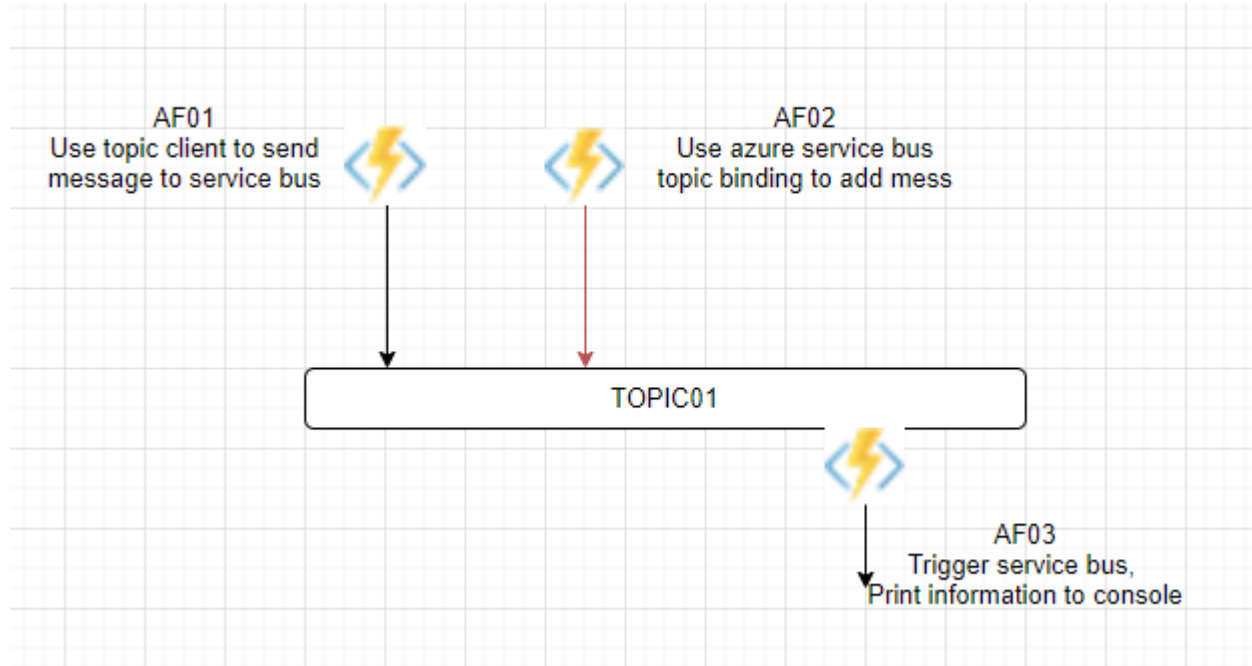
II. Azure function – Service Bus Integration (Trigger and Binding)

- Reference link: <https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-service-bus>
- Let you integrate with Azure Service Bus to react to and send queue or topic message
 - Trigger: run function when queue message or topic message is created.
 - Output binding: Send a message to Azure Service Bus(queue/topic).
- **Attributes and annotations**
 - “ServiceBusTrigger”: set name of queue/ topic and subscription and connection
- **Parameter type for the queue or topic message**
 - String
 - Byte[]
 - Message
 -

Mai Hoang Giang

II. Azure function – Service Bus Trigger – Example

- Service bus trigger
- Service bus as an output binding



Mai Hoang Giang

II. Azure function – SignalR integration

- Reference link: <https://docs.microsoft.com/en-us/azure/azure-functions/functions-bindings-signalr-service>
- Let you integrate with Azure SignalR Service
 - Trigger
 - Input binding
 - Output binding: Broadcast message to clients that subscribe the SignalR service.

II. Azure function – How to develop a azure function

<https://docs.microsoft.com/en-us/azure/azure-functions/functions-create-your-first-function-visual-studio>

Mai Hoang Giang

II. Azure function – Dependency injection

- Reference link: <https://docs.microsoft.com/en-us/azure/azure-functions/functions-dotnet-dependency-injection>
- Familiarity with .Net core dependency injection
- Support Azure Function 2.x
- Service lifetime:
 - Transient
 - Scoped
 - Singleton

Mai Hoang Giang

II. Azure function – Dependency injection – Register services

- Create Startup class inheritance FunctionsStartup
- Use IFunctionsHostBuilder to add(register) component/services

```
using Microsoft.Azure.Functions.Extensions.DependencyInjection;
using Microsoft.Extensions.DependencyInjection;

[assembly: FunctionsStartup(typeof(MyNamespace.Startup))]

namespace MyNamespace
{
    public class Startup : FunctionsStartup
    {
        public override void Configure(IFunctionsHostBuilder builder)
        {
            builder.Services.AddHttpClient();

            builder.Services.AddSingleton<IMyService>((s) => {
                return new MyService();
            });

            builder.Services.AddSingleton<ILoggerProvider, MyLoggerProvider>();
        }
    }
}
```

Mai Hoang Giang

II. Azure function – DI – Use injected dependencies

```
namespace MyNamespace
{
    public class MyHttpTrigger
    {
        private readonly HttpClient _client;
        private readonly IMyService _service;

        public MyHttpTrigger(HttpClient httpClient, IMyService service)
        {
            this._client = httpClient;
            this._service = service;
        }

        [FunctionName("MyHttpTrigger")]
        public async Task<ActionResult> Run(
            [HttpTrigger(AuthorizationLevel.Function, "get", "post", Route = null)]
            ILogger log)
        {
            var response = await _client.GetAsync("https://microsoft.com");
        }
    }
}
```

Mai Hoang Giang



Durable Function

Mai Hoang Giang

II. Durable function

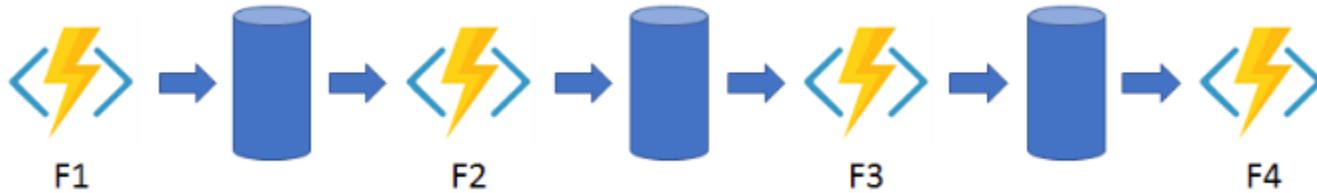
- Reference link: <https://docs.microsoft.com/en-us/azure/azure-functions/durable/durable-functions-overview?tabs=csharp>
- <https://docs.microsoft.com/en-us/dotnet/architecture/serverless/durable-azure-functions>
- <https://docs.microsoft.com/en-us/archive/msdn-magazine/2019/august/azure-affairs-of-state-serverless-and-stateless-code-execution-with-azure-functions>
- **Is an extension of Azure function**
 - Let you write stateful functions.
 - Let you define stateful workflow by writing orchestrator functions

II. Durable function – application patterns

- Function chaining
- Fan-out/fan-in
- Async HTTP APIs
- Monitoring
- Human interaction
- Aggregator (stateful entities)

II. Durable function – Function chaining

- Execute a sequence of function in a specific order.
- The output of a function may be the input of another function



II. Durable function – Function chaining - Example

```
[FunctionName("Chaining")]
public static async Task<object> Run(
    [OrchestrationTrigger] IDurableOrchestrationContext context)
{
    try
    {
        var x = await context.CallActivityAsync<object>("F1", null);
        var y = await context.CallActivityAsync<object>("F2", x);
        var z = await context.CallActivityAsync<object>("F3", y);
        return await context.CallActivityAsync<object>("F4", z);
    }
    catch (Exception)
    {
        // Error handling or compensation goes here.
    }
}
```

II. Durable function – Function chaining - Demo

- Demo function chaining
- Demo Dependency Injection

IV

Q&A

Mai Hoang Giang

Thank you !!!

Mai Hoang Giang