

Implementing EDA

Memi Lavi
www.memilavi.com



Implementing Event Driven Architecture

- We covered a lot of theory
- How do we implement EDA?

Implementing Event Driven Architecture

- Mainly 4 things to consider:

Events Approach

Implementing the Channel

Implementing the Producer

Implementing the Consumer

Events Approach

- Two main approaches for implementing events:

Events are retained

Events are not retained

Retaining Events

- The channel retains the event for future handling
- A retention period is defined which after it expires – the event is removed
- Great for streaming events and when the channel is the source of truth

Not Retaining Events

- The channel publishes the events and does not store them
- If a consumer missed an event – it can't be replayed
- Used mainly for in-system events

Implementing the Channel

- Depends on the events approach

Events are retained

- Use a messaging / queue engine
- Common engines:
 - RabbitMQ
 - Kafka

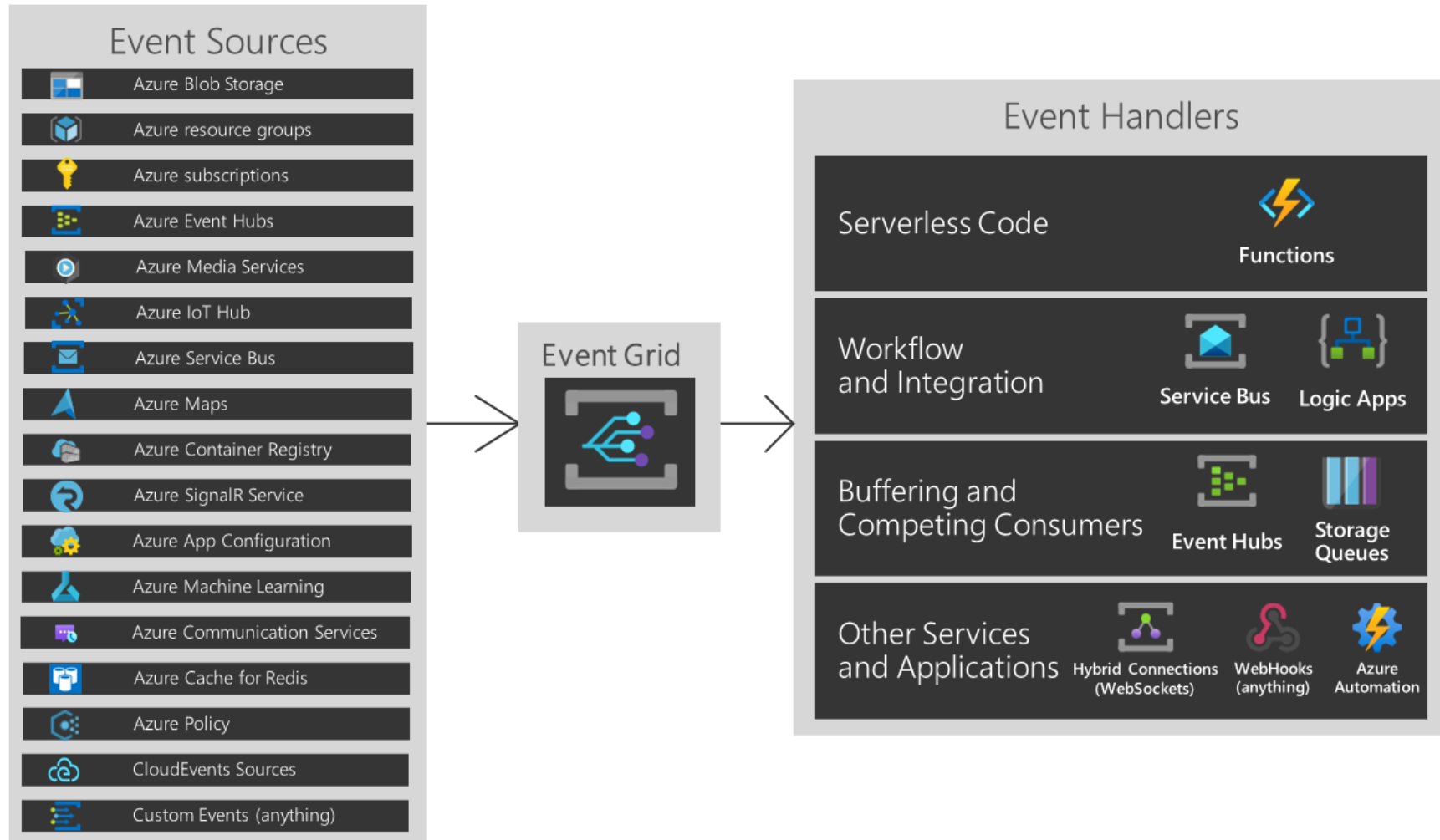
Events are not retained

- Use an event publisher
- Specific engine depends on platform used, types of interfaces and more
- Let's see some examples...

Azure Event Grid

- Events publisher in the cloud
- Fully hosted in the Azure cloud, no installation required
- Great integration with a lot of event sources and handlers
- Can deal with thousands of events / sec

Azure Event Grid

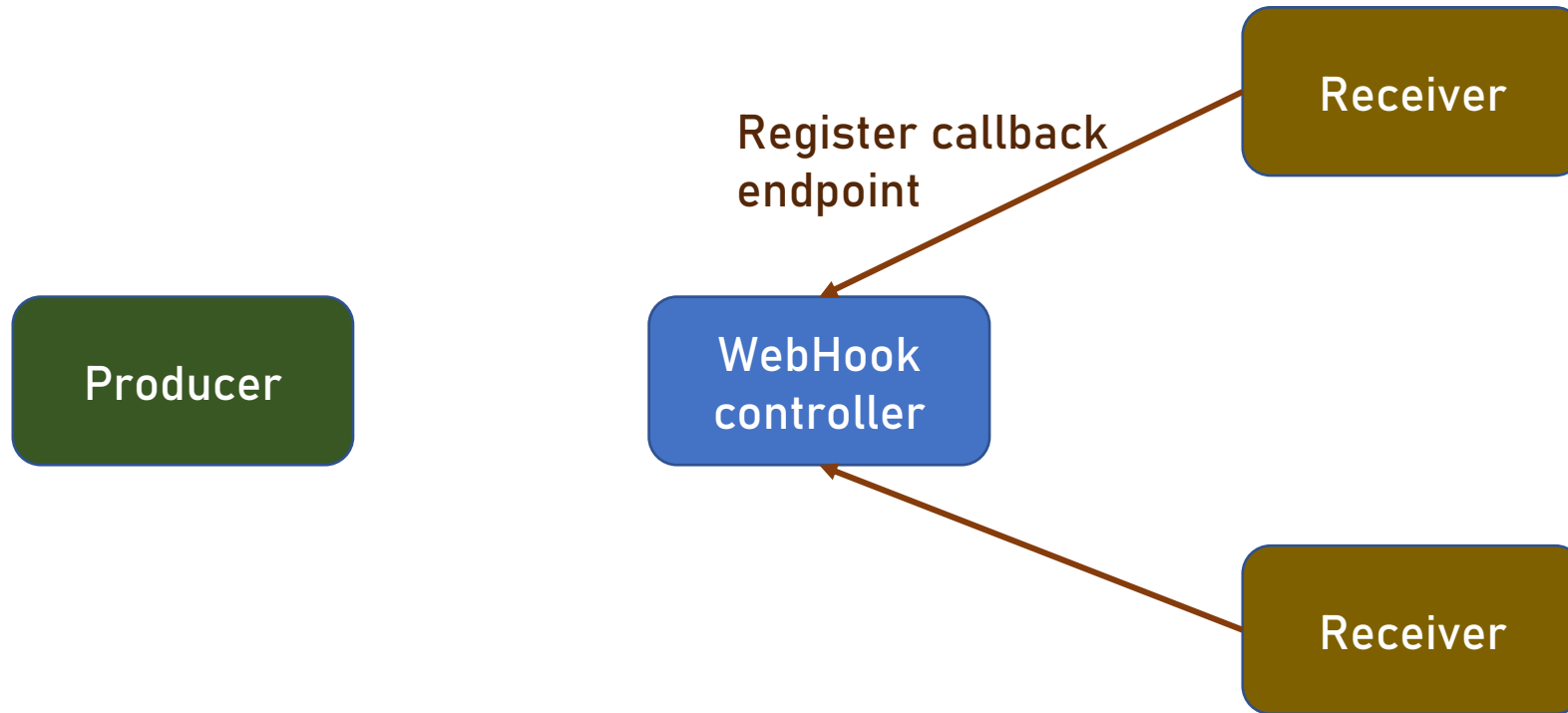


Source: <https://docs.microsoft.com/en-us/azure/event-grid/overview>

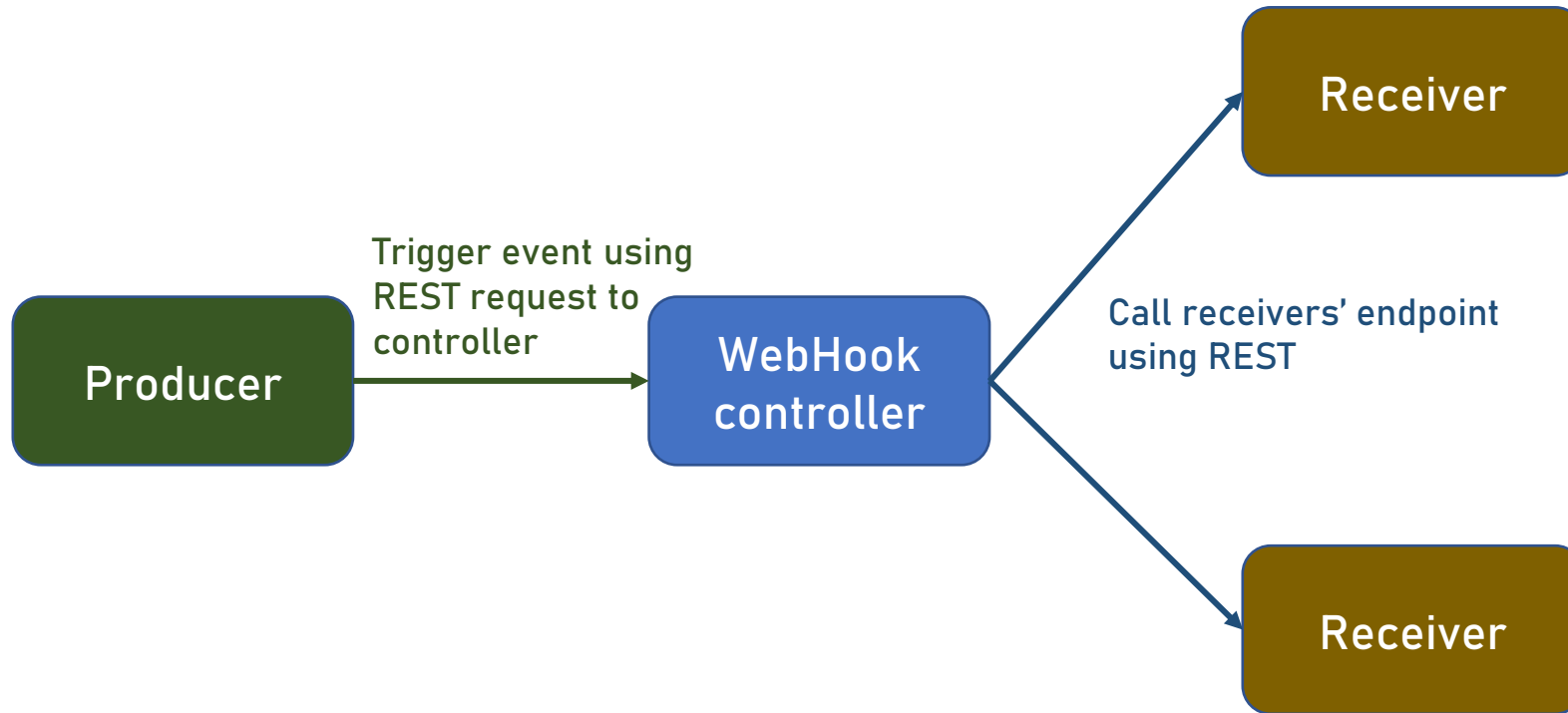
WebHooks

- A standard for publishing events using REST API
- Consumers subscribe to the WebHook engine and register a REST API endpoint that will be called when an event occurs
- The WebHook will call the endpoints when event is triggered
- Easy to implement
- Supported by GitHub, DropBox, PayPal, Stripe and more

WebHooks Flow



WebHooks Flow



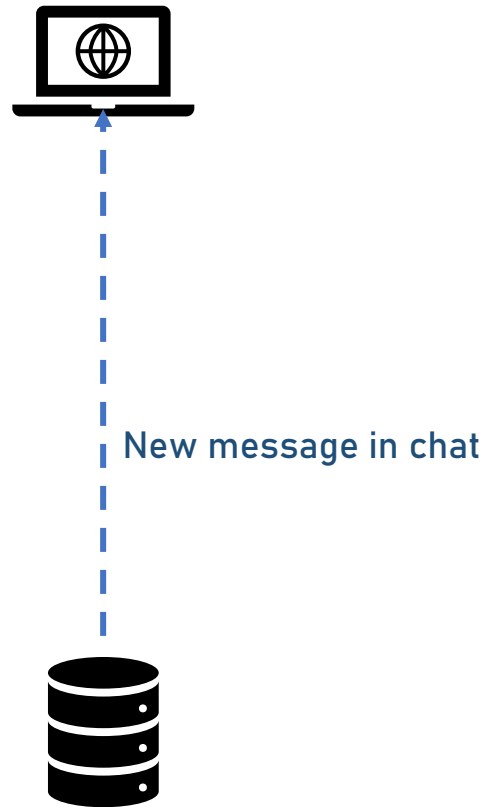
WebHooks Implementation

- Libraries for implementing WebHooks in various platforms
 - E.g. ASP.NET WebHooks
- Websites offering WebHooks:
 - Zapier
 - Ifttt
 - HostedHooks
 - And more...

HTTP Push Notification

- Send events from the server to client(s)
- Great for chats, message notification and more

HTTP Push Notifications



Implementing Push Notifications

- Quite a lot of libraries and frameworks:
 - SignalR
 - Socket.IO
 - gRPC
 - And more...

How to Choose Event Publisher

Use...	When...
Azure Event Grid or similar	<ul style="list-style-type: none">- Hosted in the cloud- Need strong integration between backend services
WebHooks	<ul style="list-style-type: none">- Receivers expose REST API- Need something simple and quick
HTTP Push Notification	<ul style="list-style-type: none">- Need to notify the end user

Remember...

NEVER develop your
own channel

Implementing the Producer

- Can be based on any platform
- Needs to be able to communicate with the channel
- Depends on the channel implementation
- Let's see some examples...

Implementing the Producer

- RabbitMQ
 - Use the RabbitMQ client library for your platform
 - There's one for almost every platform
 - Choose from:
 - <https://www.rabbitmq.com/devtools.html>

Implementing the Producer

- SignalR
 - Install the SignalR library
 - Configure the Hub
 - Allow connections from clients
 - Define functions that will send messages to clients
 - Optional – create groups to filter messaging

Implementing the Consumer

- Can be based on any platform
- Needs to be able to communicate with the channel
- Depends on the channel implementation
- Let's see some examples...

Implementing the Consumer

- RabbitMQ
 - Use the RabbitMQ client library for your platform
 - There's one for almost every platform
 - Choose from:
 - <https://www.rabbitmq.com/devtools.html>

Implementing the Consumer

- WebHooks
 - Register the consumer using the WebHook REST API
 - Expose REST API that will be called by the WebHooks