Integration patterns with Azure Functions

Auckland Connected Systems User Group

GIBC
GLOBAL INTEGRATION BOOTCAMP

adaptiv DATACOM Microsoft theta

Presenter

Daniel Larsen
FastTrack Engineer, Azure Engineering

dalars@microsoft.com
@DanielLarsenNZ

*opinions are my own

Auckland Connected Systems User Group

Azure Functions

Build apps faster with serverless technologies

Daniel Larsen
Microsoft

What are the benefits?

Focus

Solve business problems—not technology problems related to undifferentiated heavy lifting

Efficiency

Shorter time to market

Fixed costs converted to variable costs

Better service stability

Better development and testing management

Less waste

Flexibility

Simplified starting experience

Easier pivoting means more flexibility

Easier experimentation

Scale at your pace—don’t bet the farm on Day 1

Natural fit for microservices

Azure Functions

Functions-as-a-Service programming model use functions to achieve true serverless compute

Single responsibility

Functions are single-purposed, reusable pieces of code that process an input and return a result

Short lived

Functions don't stick around when finished executing, freeing up resources for further executions

Stateless

Functions don't hold any persistent state and don't rely on the state of any other processes

Event driven & scalable

Functions respond to predefined events, and are instantly replicated as many times as needed

Gain flexibility and develop your way

Multiple languages

Write code in C#, JavaScript, F#, Python, PowerShell and Java

Continuous investment in new, experimental languages

Durable Functions

Write stateful functions in a serverless environment

Simplify complex, stateful coordination problems

Add the extension to enable advanced scenarios

Hosting options

Choose from six consumption plans to run Functions

Run your first million function executions for free

Dev options

Simplify coding for new users with native Azure portal

Select from popular editors, like VS, VS Code, CLI, Maven*

*VS and VS Code only support C#; Maven only supports Java

Gain flexibility and develop your way

Hosting options

Consumption	AS Plan	AS Environment	Azure Stack	Runtime	IoT Edge*
Serverless	*Free, Basic, Standard, Premium*	*Network isolation*	*On-premises*	*Functions on your server*	*On devices*
Only pay for what you use; charges apply per execution and per GB second	Gain all the advantages of Functions along with Microsoft's financially-backed SLA and the always-on features of an App Service Plan	Use a dedicated App Service cloud environment (ASE) that comes with network isolation for apps, greater scale, and secure connectivity to local vNets	Bring the power of the entire Azure stack to your own data centers	Run Functions on your local server; does not include the entire Azure stack	Deploy custom Azure modules on IoT devices

*Currently in preview

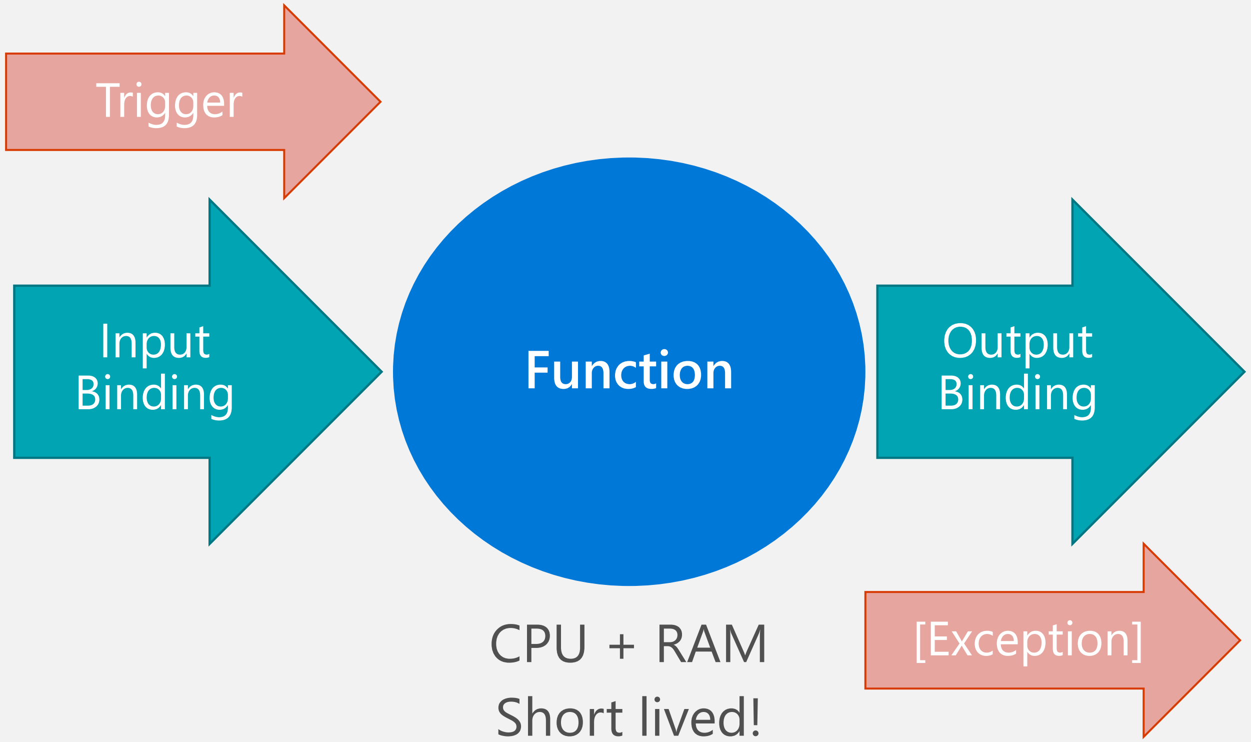
Azure Functions is an **open-source** project

Functions runtime and all extensions are fully open source

https://github.com/Azure/Azure-Functions

Function

Timer
Logic App
Event Hub
Event Grid
Service Bus
Storage
SQL
HTTP
and many more…



Trigger

Input Binding

Function

Output Binding

[Exception]

CPU + RAM

Short lived!
(milliseconds or seconds)

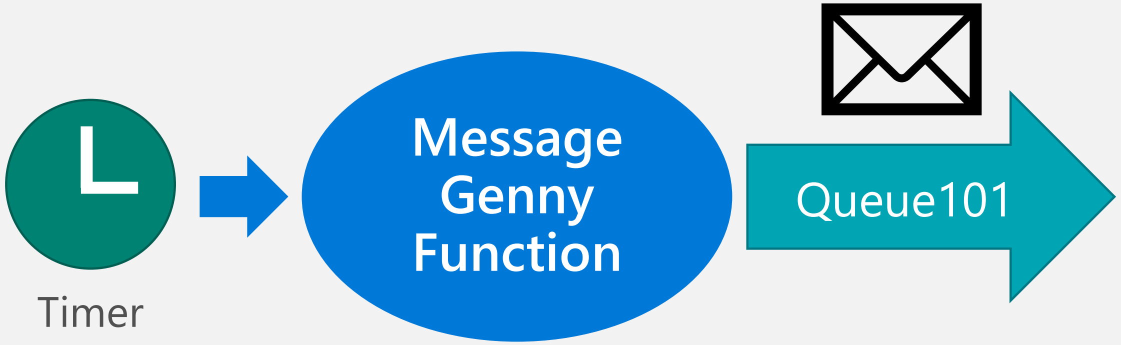
Exceptions are meaningful!

Event Hub
Event Grid
Service Bus
Logic App
Storage
SQL
HTTP
Cosmos
And many more…

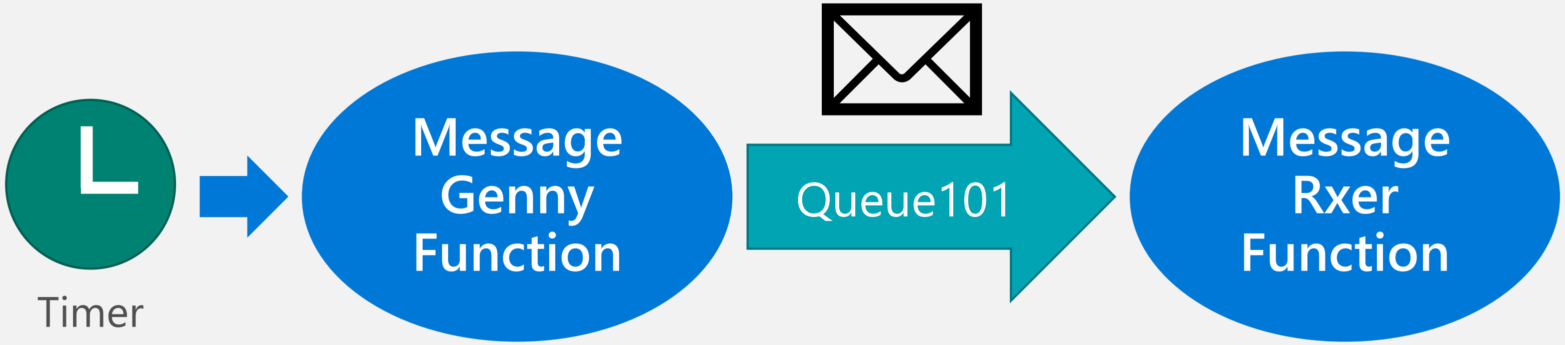
Auckland Connected Systems User Group

Integration patterns

Queue

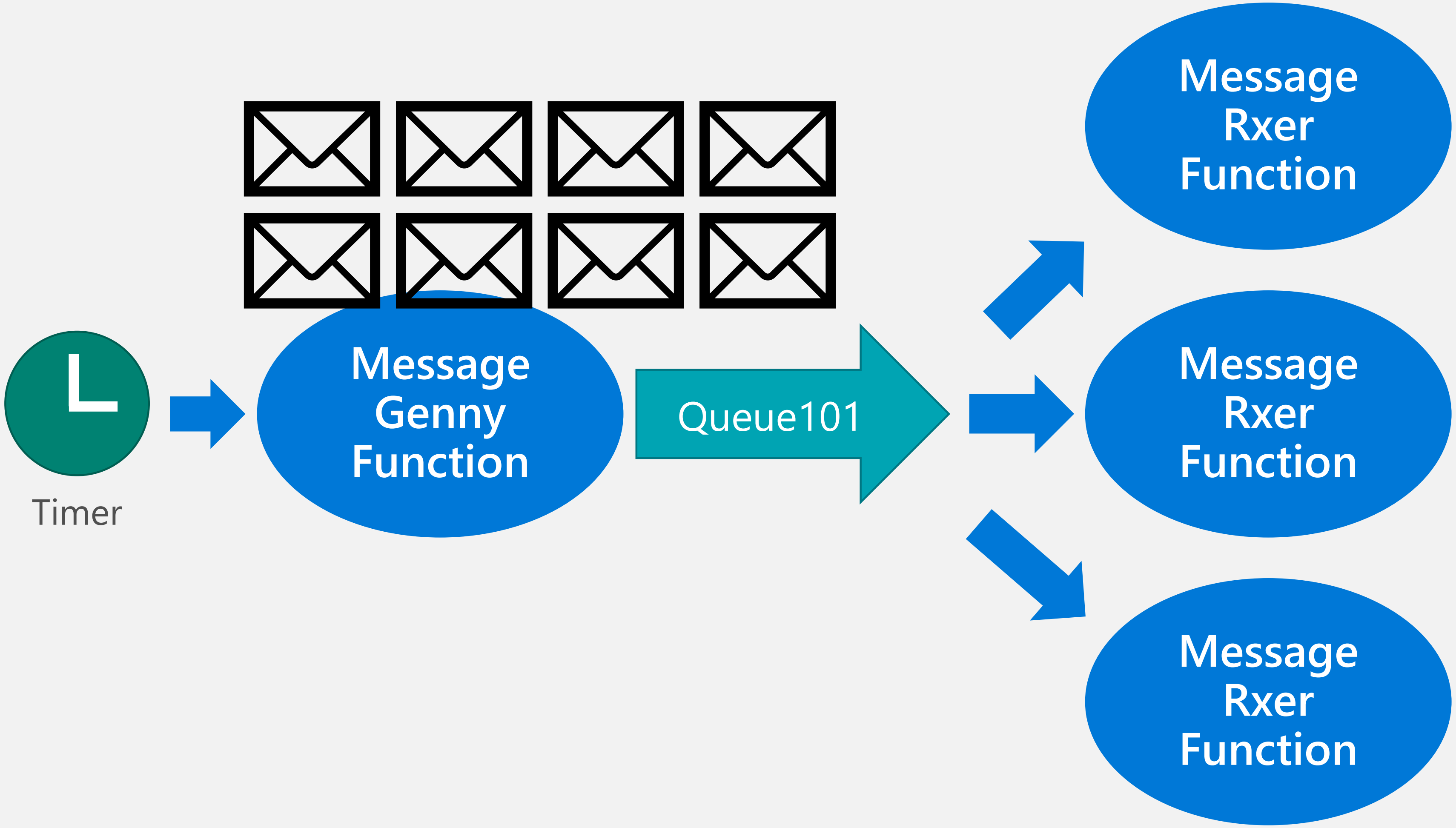


Queue



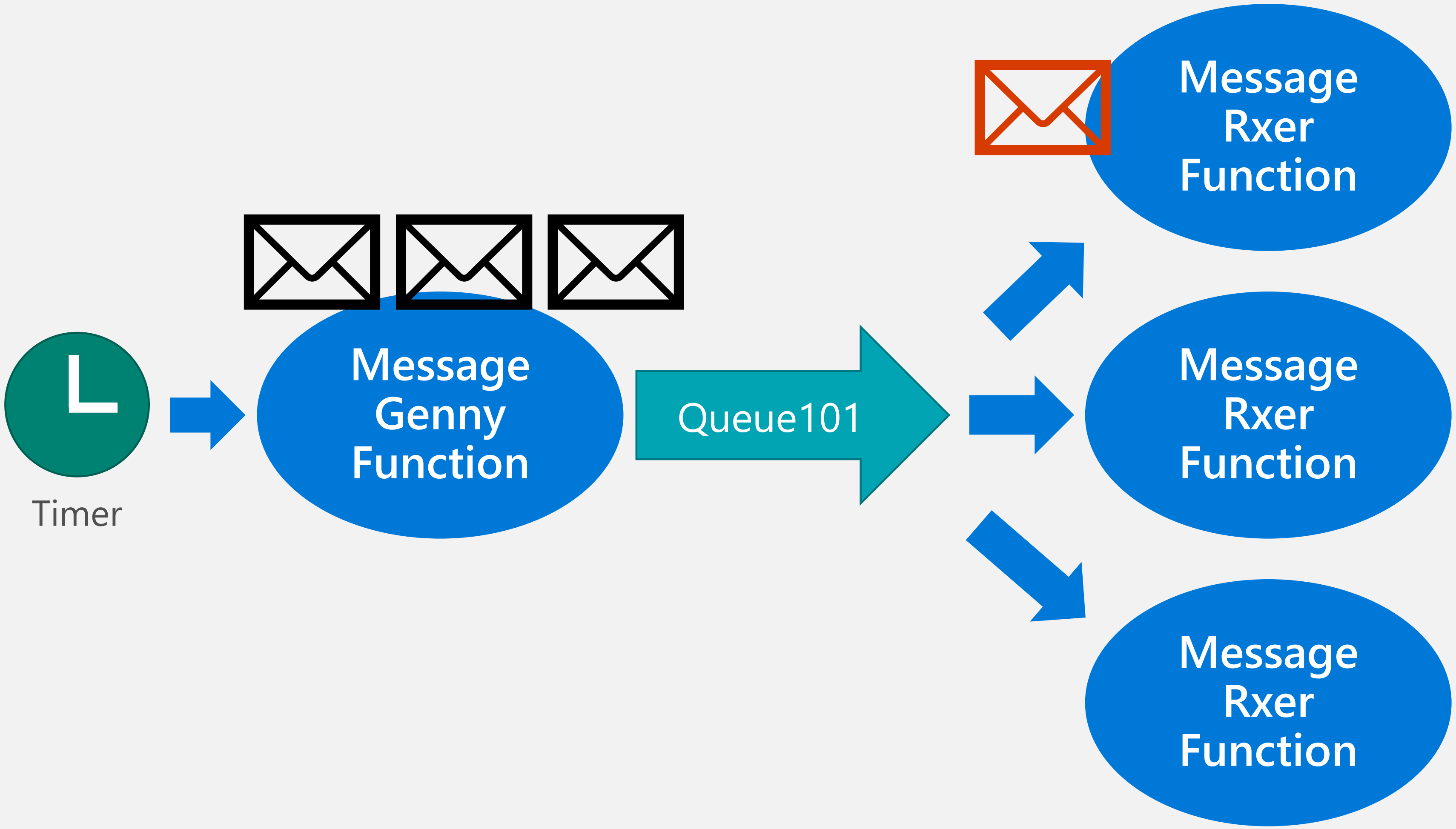
https://docs.microsoft.com/en-us/previous-versions/msp-n-p/dn589781(v=pandp.10)?redirectedfrom=MSDN#sending-and-receiving-messages-by-using-a-message-queue

Competing consumers



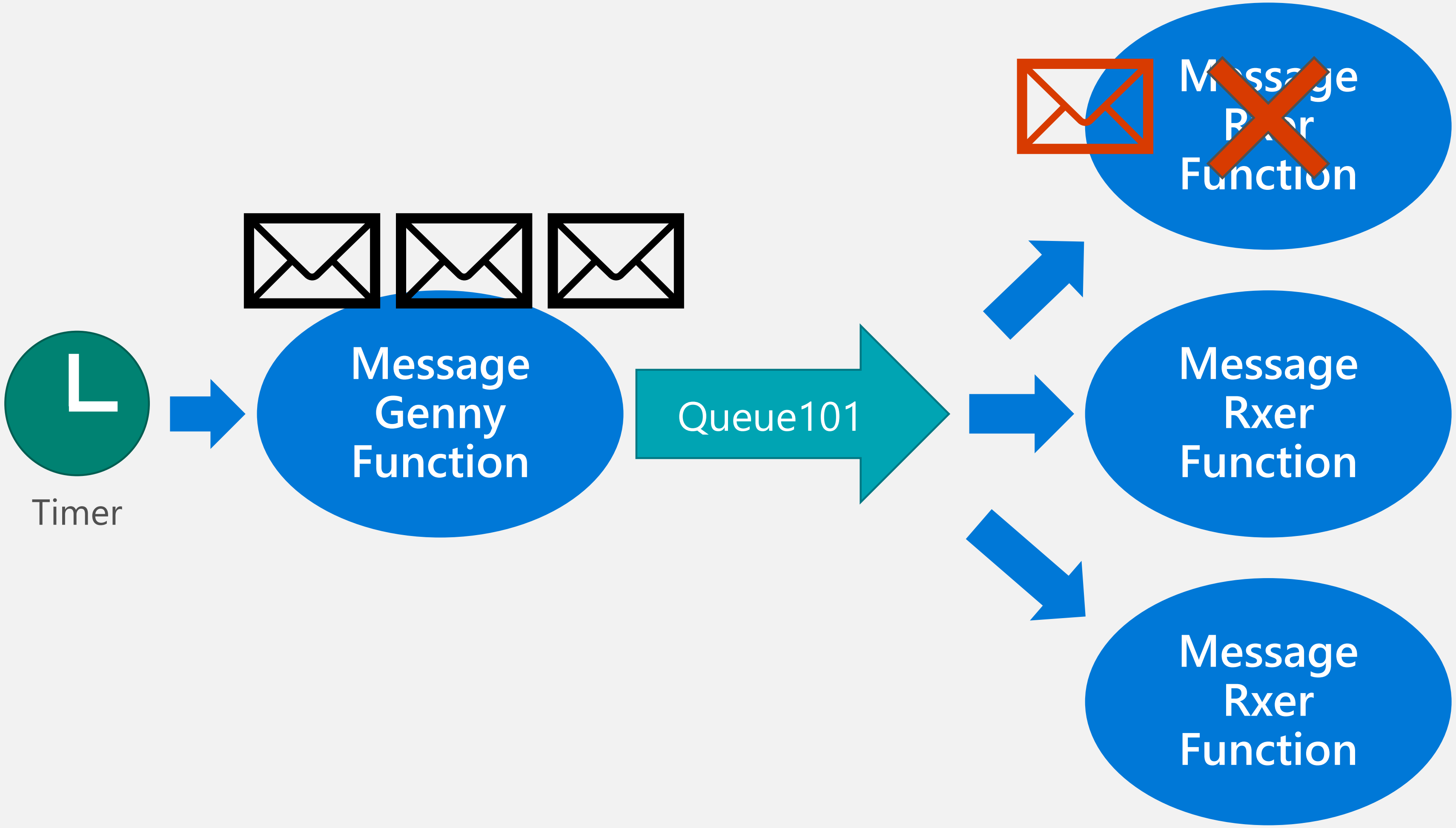
https://docs.microsoft.com/en-us/azure/architecture/patterns/competing-consumers

Message lock



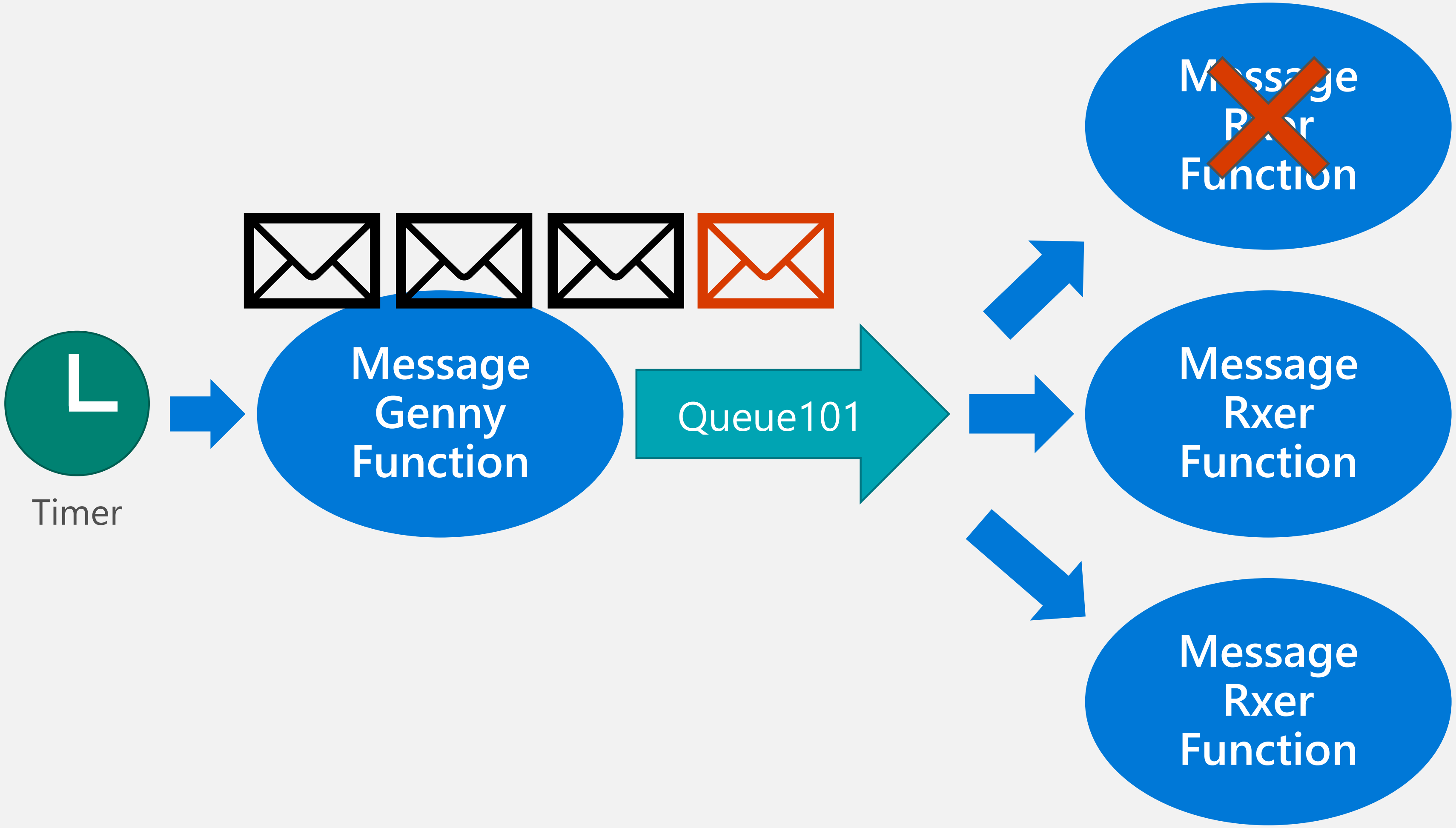
https://docs.microsoft.com/en-us/azure/service-bus-messaging/message-transfers-locks-settlement#settling-receive-operations

Message lock



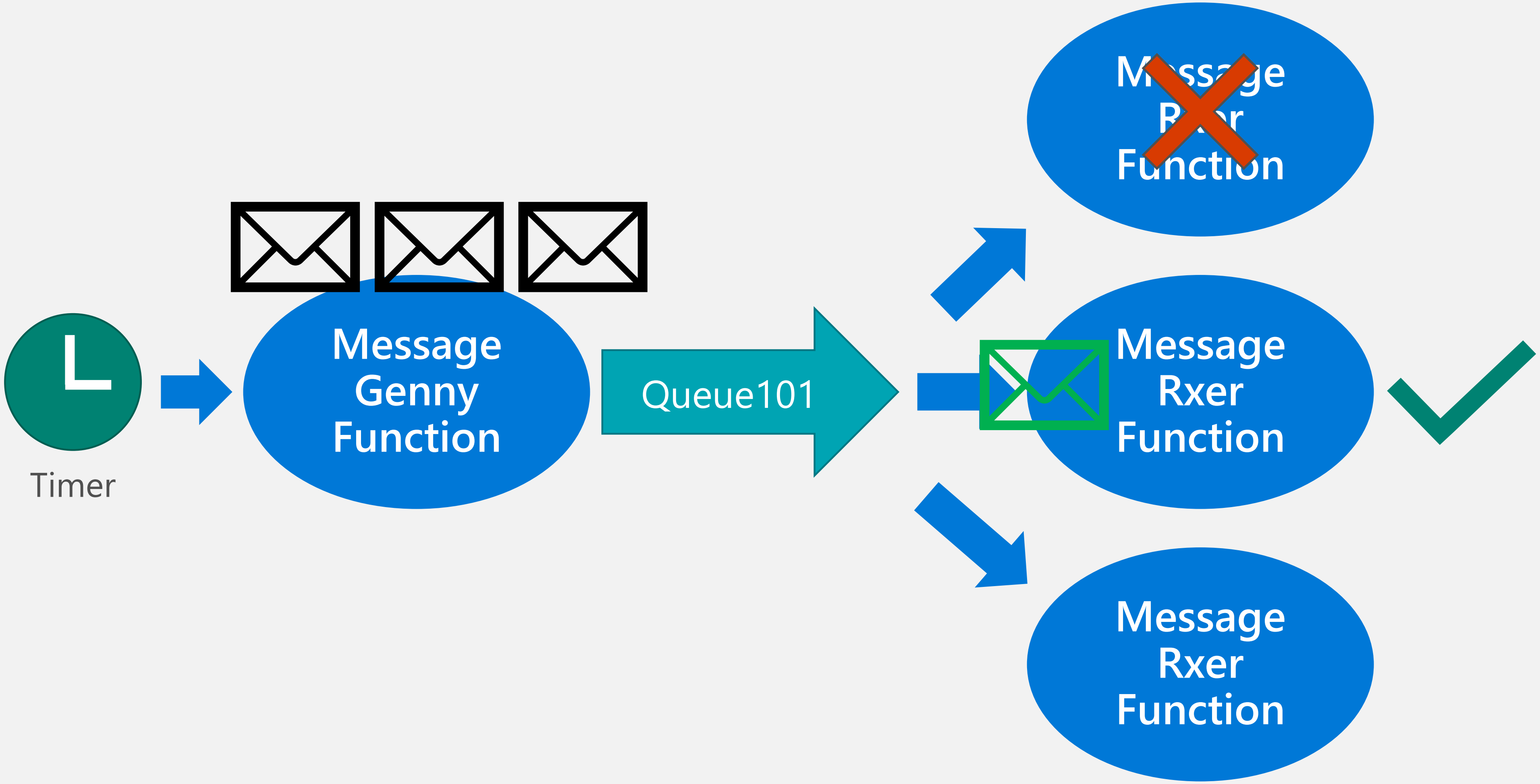
https://docs.microsoft.com/en-us/azure/service-bus-messaging/message-transfers-locks-settlement#settling-receive-operations

Message abandon



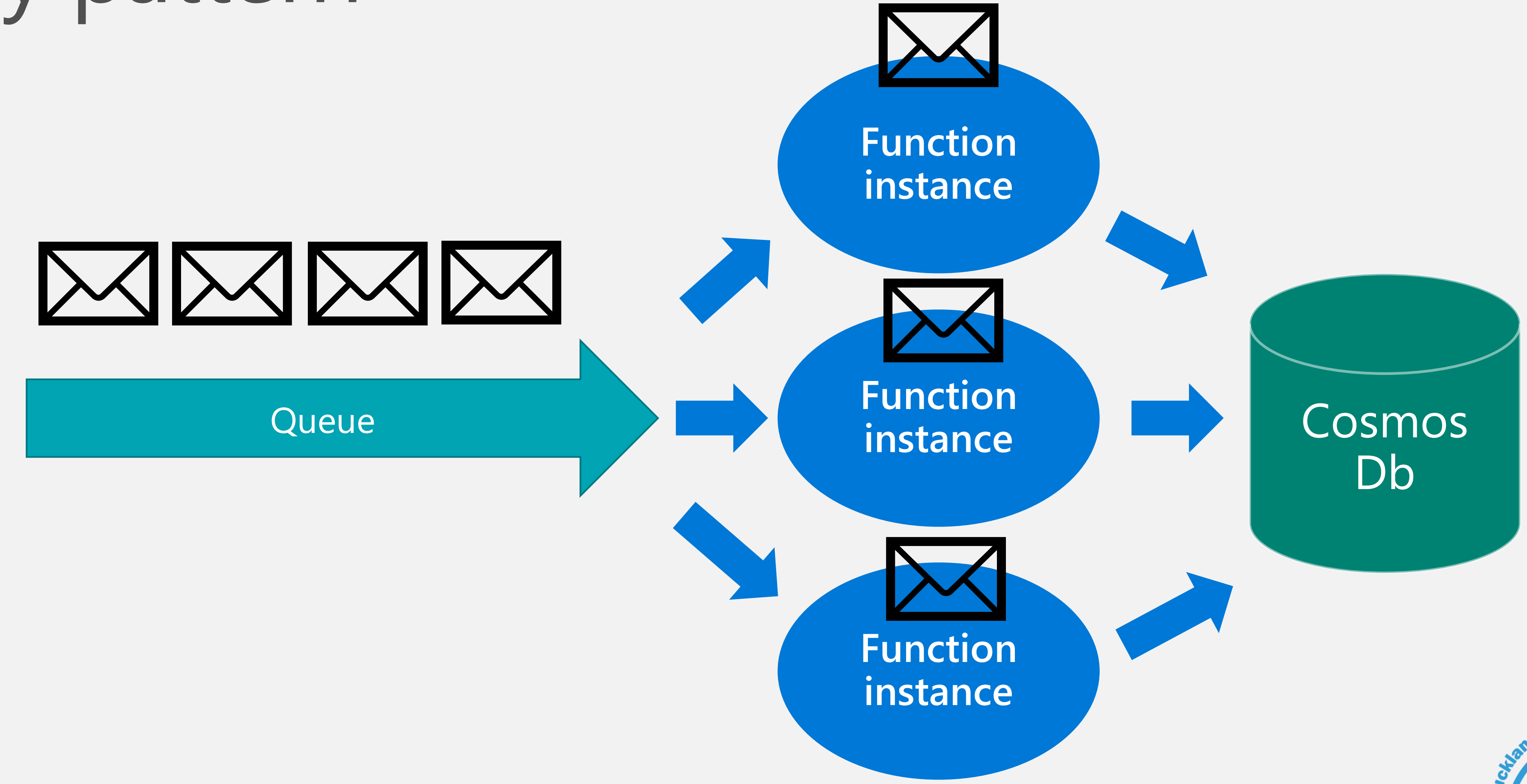
https://docs.microsoft.com/en-us/azure/service-bus-messaging/message-transfers-locks-settlement#settling-receive-operations

Message complete



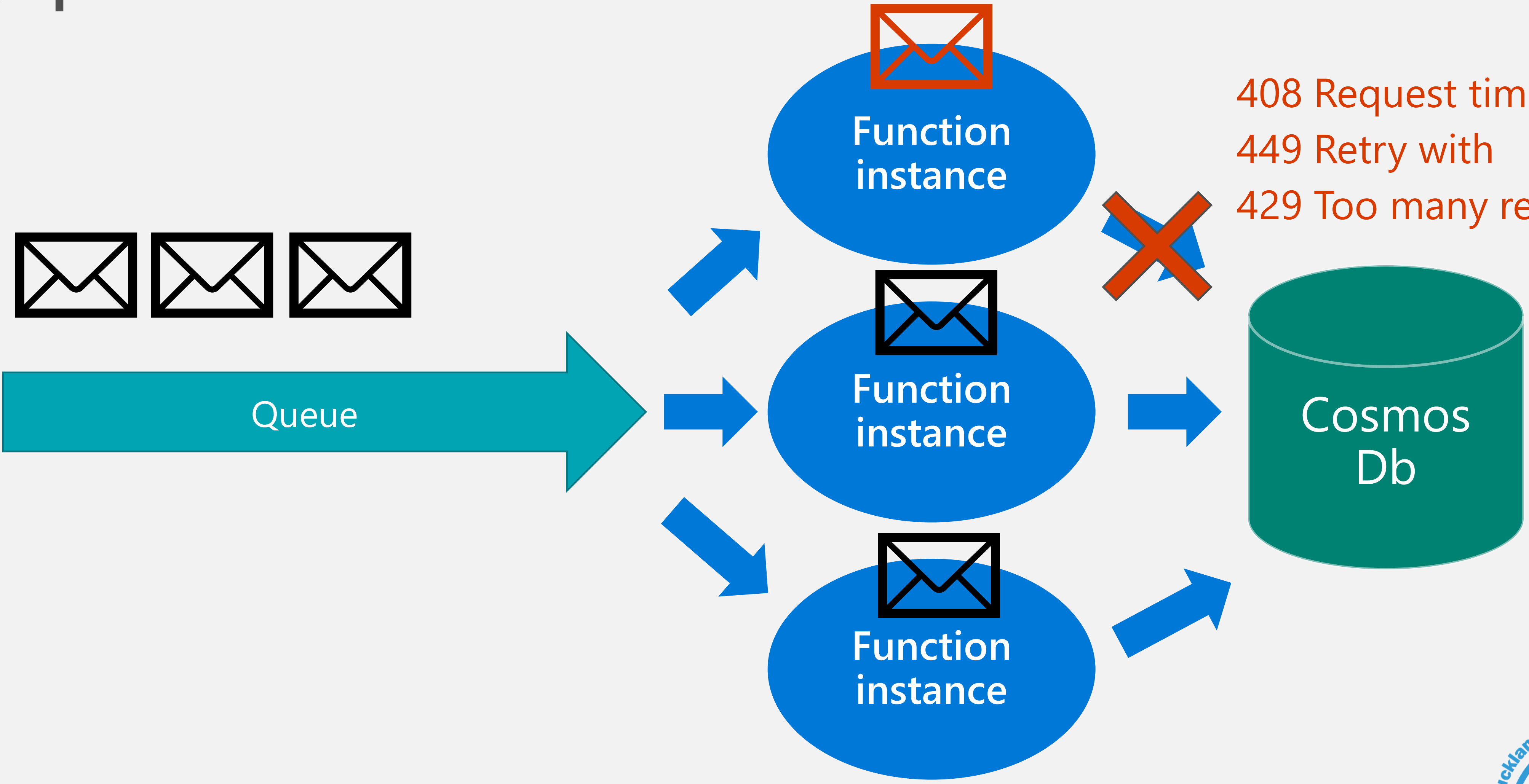
https://docs.microsoft.com/en-us/azure/service-bus-messaging/message-transfers-locks-settlement#settling-receive-operations

Retry pattern



https://docs.microsoft.com/en-us/azure/architecture/patterns/retry

Retry pattern



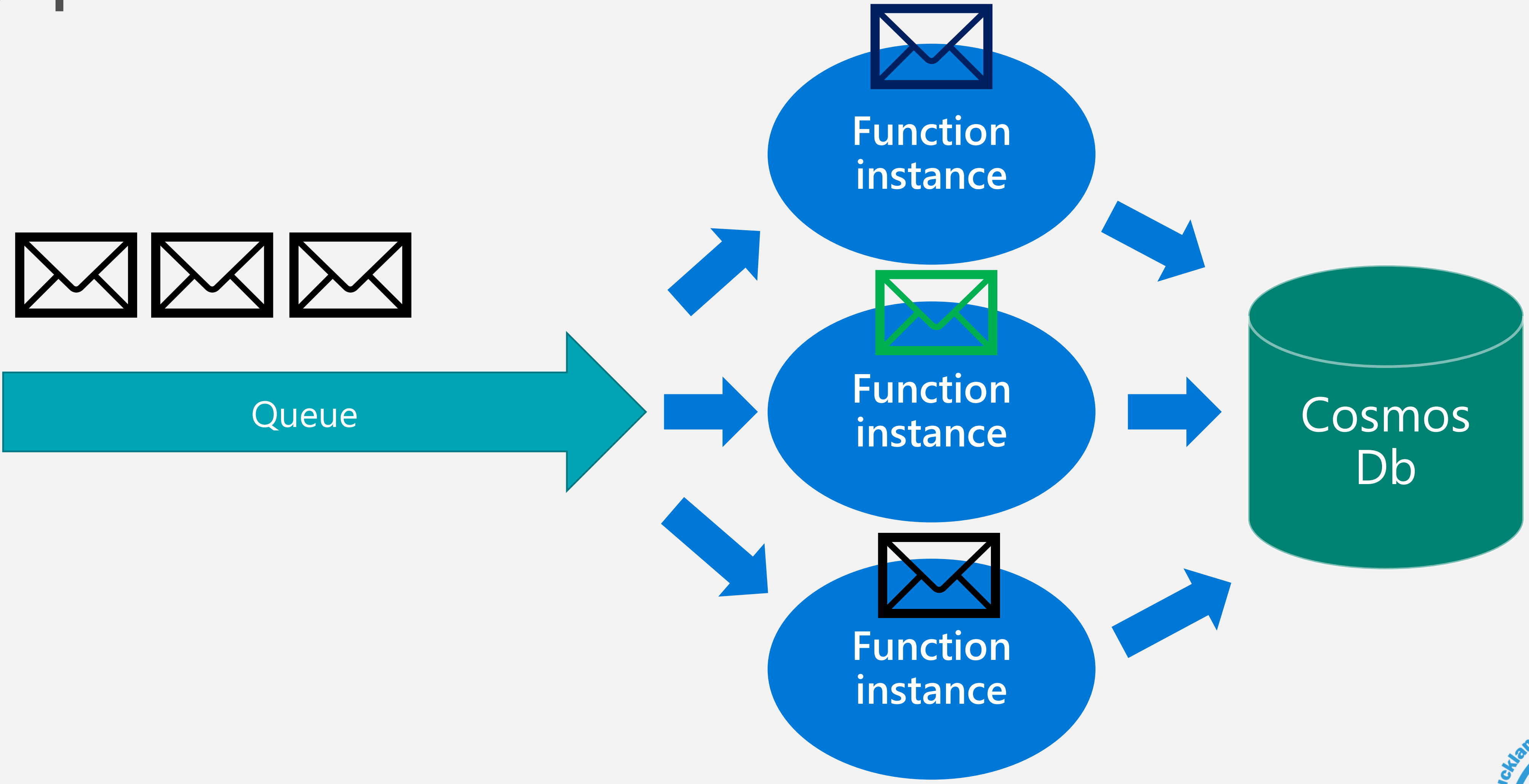
408 Request timeout

449 Retry with

429 Too many requests

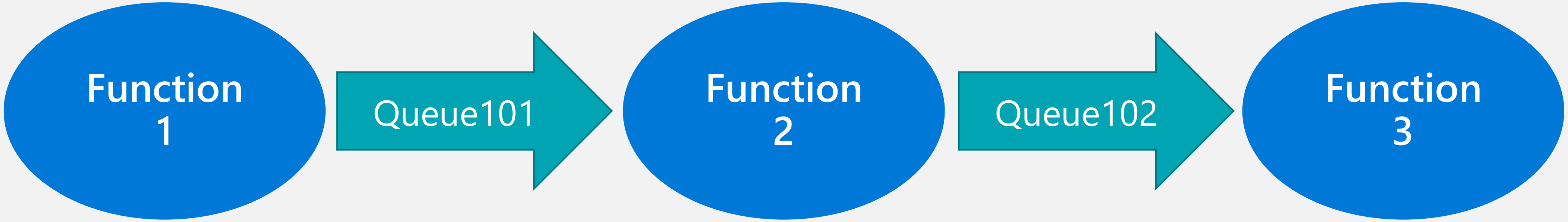
https://docs.microsoft.com/en-us/azure/architecture/patterns/retry

Retry pattern



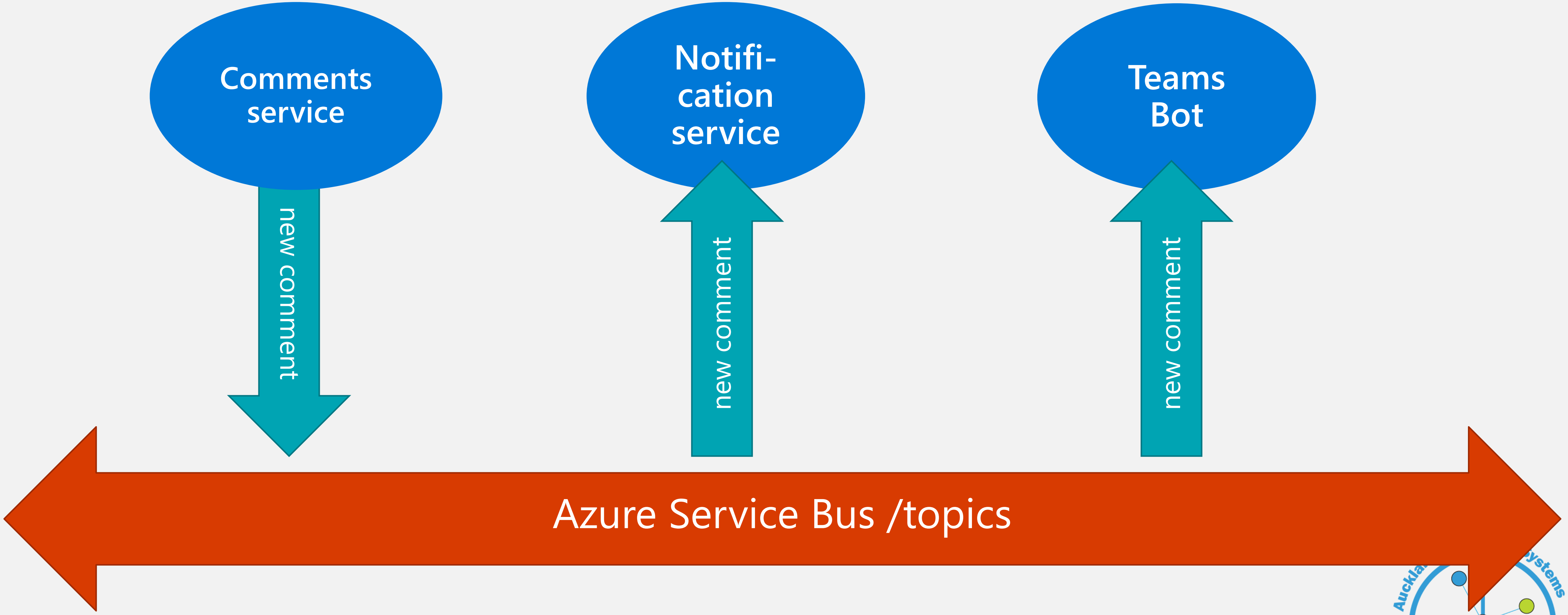
https://docs.microsoft.com/en-us/azure/architecture/patterns/retry

Pipes & filters



https://docs.microsoft.com/en-us/azure/architecture/patterns/pipes-and-filters

Publisher / subscriber (pub/sub)



Comments service

Notifi-cation service

Teams Bot

new comment

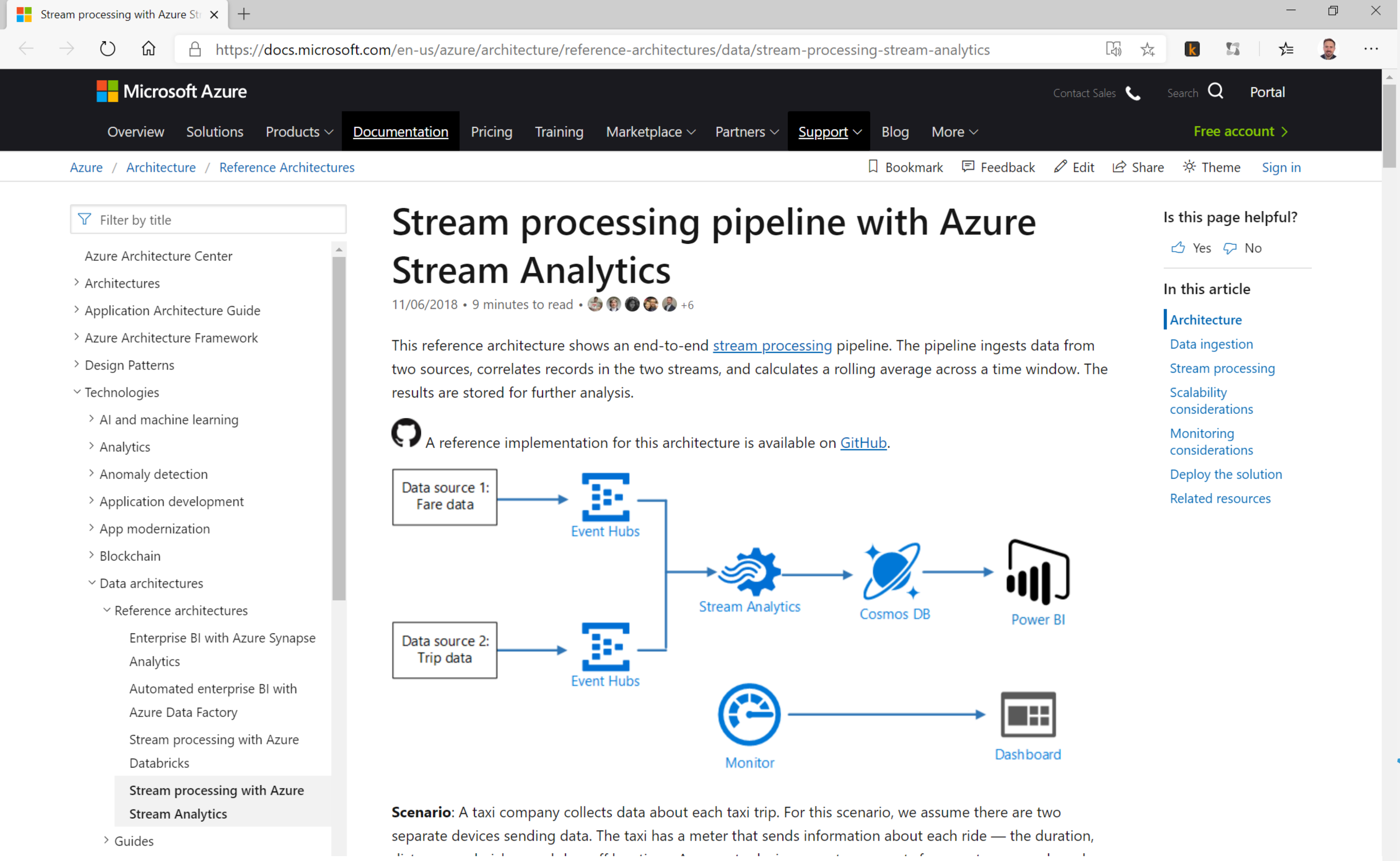
new comment

new comment

Azure Service Bus /topics

https://docs.microsoft.com/en-us/azure/architecture/patterns/publisher-subscriber

Event streaming


Stream processing with Azure St
https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/data/stream-processing-stream-analytics
Microsoft Azure
Contact Sales
Search
Portal
Overview
Solutions
Products
Documentation
Pricing
Training
Marketplace
Partners
Support
Blog
More
Free account
Azure / Architecture / Reference Architectures
Bookmark
Feedback
Edit
Share
Theme
Sign in
Filter by title
Azure Architecture Center
Architectures
Application Architecture Guide
Azure Architecture Framework
Design Patterns
Technologies
AI and machine learning
Analytics
Anomaly detection
Application development
App modernization
Blockchain
Data architectures
Reference architectures
Enterprise BI with Azure Synapse Analytics
Automated enterprise BI with Azure Data Factory
Stream processing with Azure Databricks
Stream processing with Azure Stream Analytics
Guides
Is this page helpful?
Yes
No
In this article
Architecture
Data ingestion
Stream processing
Scalability considerations
Monitoring considerations
Deploy the solution
Related resources

Stream processing pipeline with Azure Stream Analytics

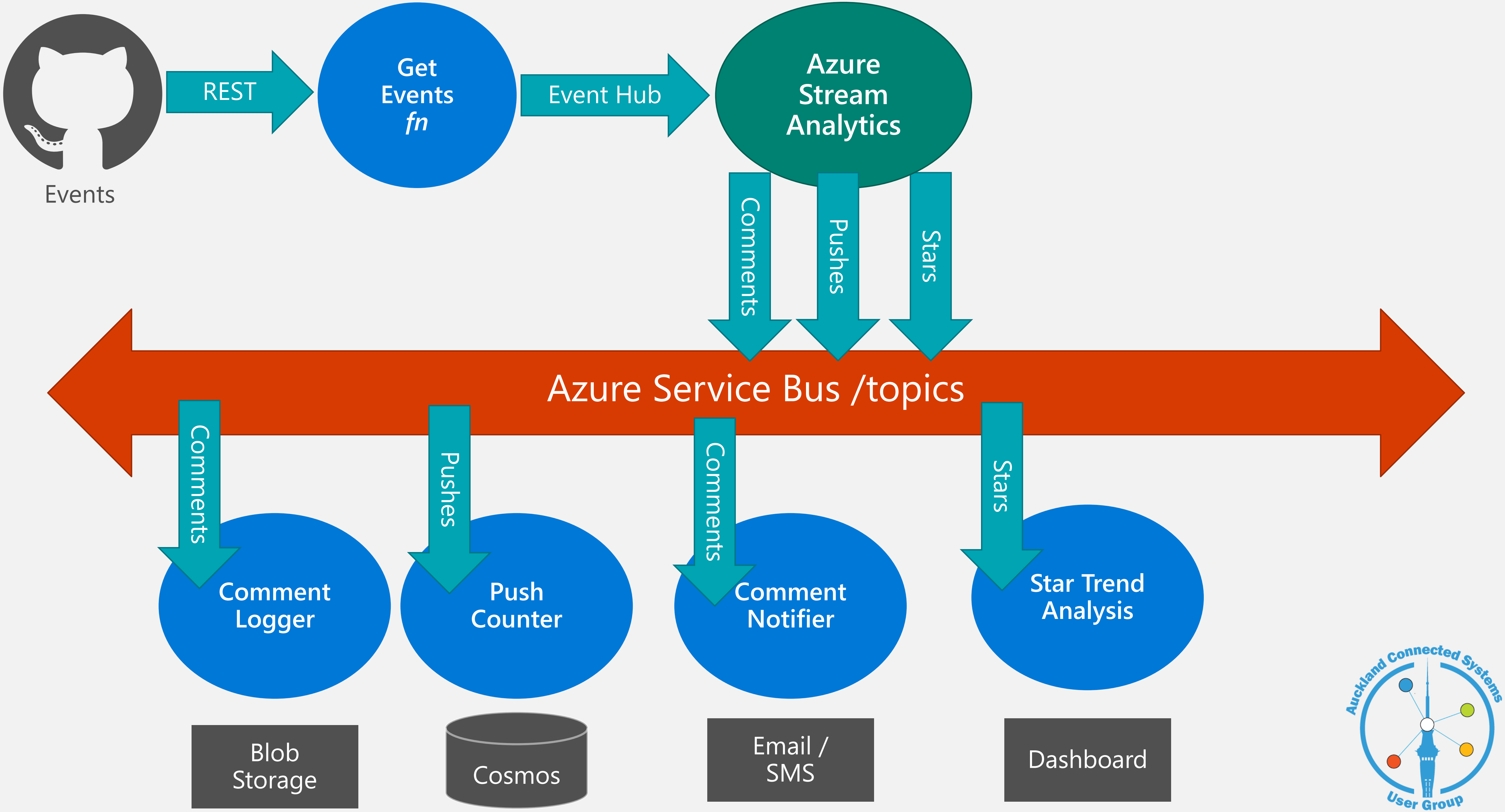
11/06/2018 • 9 minutes to read • +6

This reference architecture shows an end-to-end stream processing pipeline. The pipeline ingests data from two sources, correlates records in the two streams, and calculates a rolling average across a time window. The results are stored for further analysis.

A reference implementation for this architecture is available on GitHub.

Scenario: A taxi company collects data about each taxi trip. For this scenario, we assume there are two separate devices sending data. The taxi has a meter that sends information about each ride — the duration,

https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/data/stream-processing-stream-analytics



Events

REST

Get Events *fn*

Event Hub

Azure Stream Analytics

Comments

Pushes

Stars

Azure Service Bus /topics

Comments

Pushes

Comments

Stars

Comment Logger

Push Counter

Comment Notifier

Star Trend Analysis

Blob Storage

Cosmos

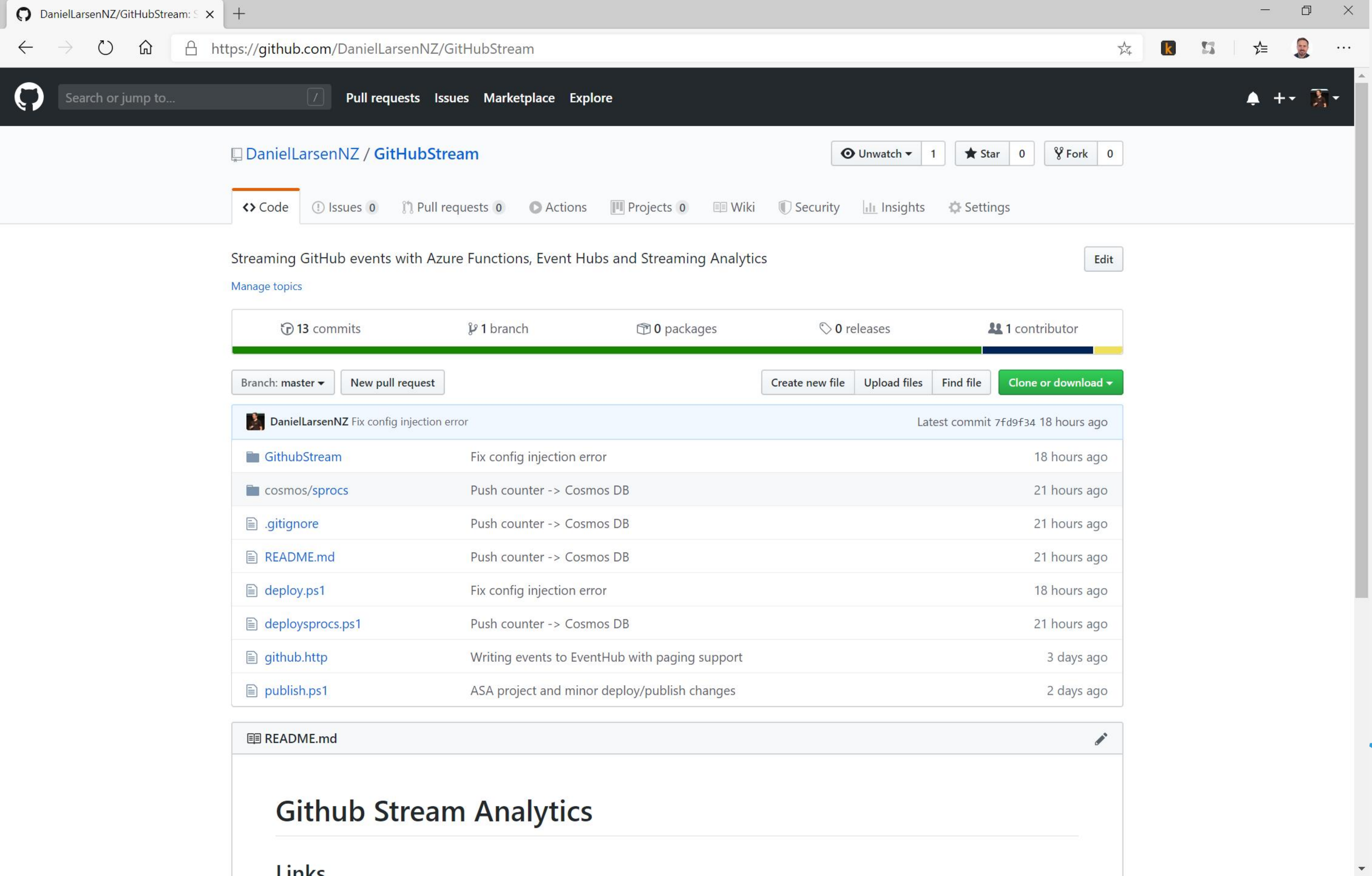
Email / SMS

Dashboard

Auckland Connected Systems User Group

Key points

- Functions allows teams to focus on **Business problems**
- Simple **input-process-output** programming model
- **Short-lived**! Stateless
- Use Service Bus for **retry**, circuit-breaker patterns
- **Compose** Functions with Service Bus, Logic Apps, Stream Analytics and other integration services. Don’t try and do everything in Functions
- Excellent for **Event-driven** architectures



DanielLarsenNZ / GitHubStream

Streaming GitHub events with Azure Functions, Event Hubs and Streaming Analytics

DanielLarsenNZ Fix config injection error — Latest commit `7fd9f34` 18 hours ago

GithubStream	Fix config injection error	18 hours ago
cosmos/sprocs	Push counter -> Cosmos DB	21 hours ago
.gitignore	Push counter -> Cosmos DB	21 hours ago
README.md	Push counter -> Cosmos DB	21 hours ago
deploy.ps1	Fix config injection error	18 hours ago
deploysprocs.ps1	Push counter -> Cosmos DB	21 hours ago
github.http	Writing events to EventHub with paging support	3 days ago
publish.ps1	ASA project and minor deploy/publish changes	2 days ago

README.md

Github Stream Analytics

Links

Reference material

https://docs.microsoft.com/en-us/previous-versions/msp-n-p/dn589781(v=pandp.10)?redirectedfrom=MSDN#sending-and-receiving-messages-by-using-a-message-queue

https://docs.microsoft.com/en-us/azure/architecture/patterns/competing-consumers

https://docs.microsoft.com/en-us/azure/architecture/patterns/pipes-and-filters

https://docs.microsoft.com/en-us/azure/service-bus-messaging/message-transfers-locks-settlement#settling-receive-operations

https://docs.microsoft.com/en-us/azure/architecture/patterns/retry

https://docs.microsoft.com/en-us/azure/architecture/patterns/circuit-breaker

https://docs.microsoft.com/en-us/azure/architecture/patterns/publisher-subscriber

https://docs.microsoft.com/en-us/azure/architecture/reference-architectures/data/stream-processing-stream-analytics

https://github.com/DanielLarsenNZ/GitHubStream

adaptiv DATACOM Microsoft theta