



Welcome to the

2018

Global **Azure**
BOOTCAMP

Rick van den Bosch
@rickvdbosch
@BetabitNL

© Betabit Nederland B.V.

betabit {...}

Day's Schedule

Time	Activity / Session Title
9:30 – 10:15	Guest arrival, Registration, Introduction
10:15 – 12:15	What is Serverless? Microsoft Flow & Azure Logic Apps LAB
12:15 – 13:00	Lunch
13:00 – 14:30	Azure Functions LAB
14:30 – 14:45	Break
14:45 – 16:00	Azure EventGrid LAB Other neat stuff
16:00 – 17:00	Closing Remarks, Questions, Raffle & Guest Departure

Rick van den Bosch
@rickvdbosch
@BetabitNL



What is Serverless?

GAB 2018

Rick van den Bosch
@rickvdbosch
@BetabitNL

© Betabit Nederland B.V.

betabit {...}

Agenda

What is Serverless?

Serverless vs. Alternatives

NoOps?

Benefits & Drawbacks

Tools

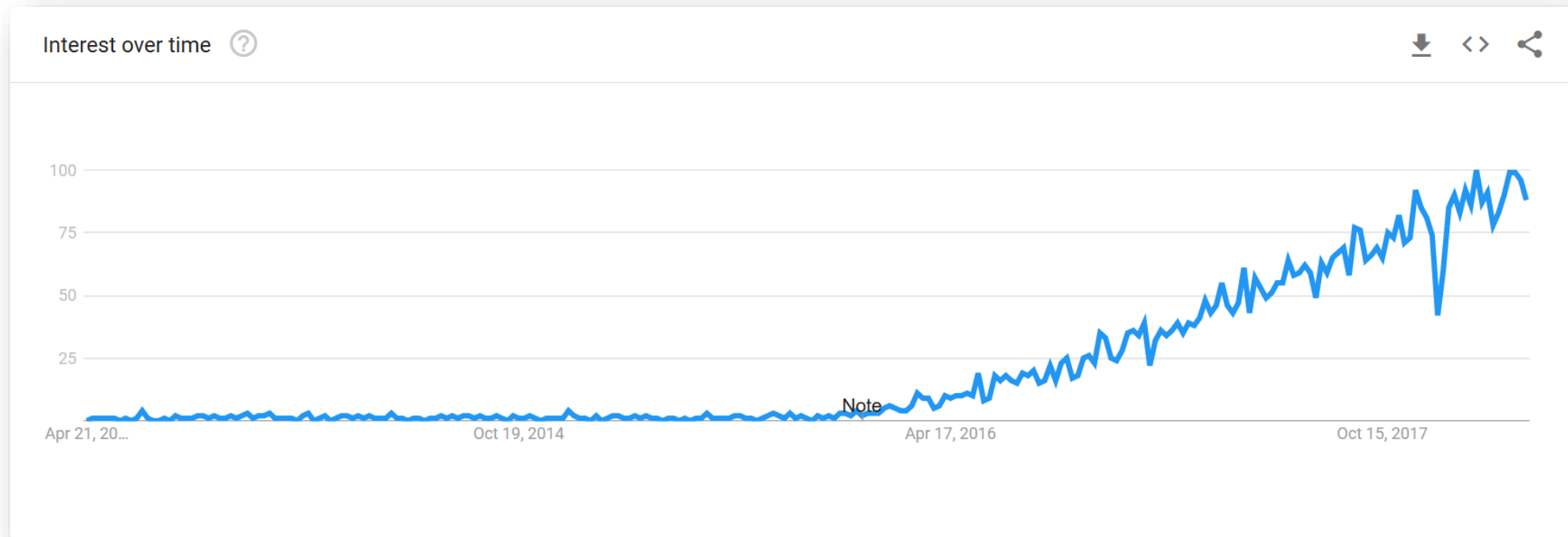
What *is* Serverless?

Does Serverless mean that there are **no** servers?

Does Serverless mean that there are **less** servers?

Maybe Serverless is not a great name...

Interest over time



Source: **Google Trends**

About Serverless...

*“Serverless is **not** just a **fad** or **buzzword**.*

{...}

With serverless, you need only worry about your code and how it is triggered. The platform takes care of the rest.”

Serverless computing defined

Abstraction of servers

Each invocation could run on a different host

Event-driven scale

Scale is defined by triggers such as timers, http calls, incoming queue messages etc.

Micro-billing

Only pay for execution and used resource



Focus on apps
not servers

Serverless and Serverless

- Backend as a Service
- Functions as a Service

Backend as a Service

- 3rd party applications / services in the cloud to manage server-side logic and state
- Applications using them were the first to be called Serverless
- Rich client applications (SPA or mobile apps)

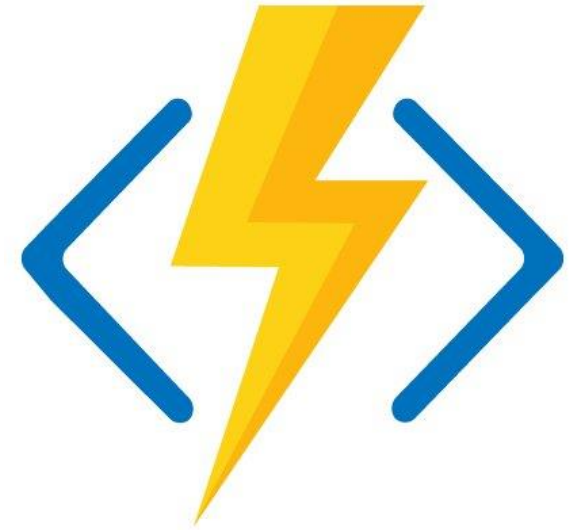
(Mobile) Backend as a Service: Example



Function as a Service

- Server-side logic written by developer
- Run in stateless compute containers
- Event-triggered
- Short-lived
- Fully managed by 3rd party

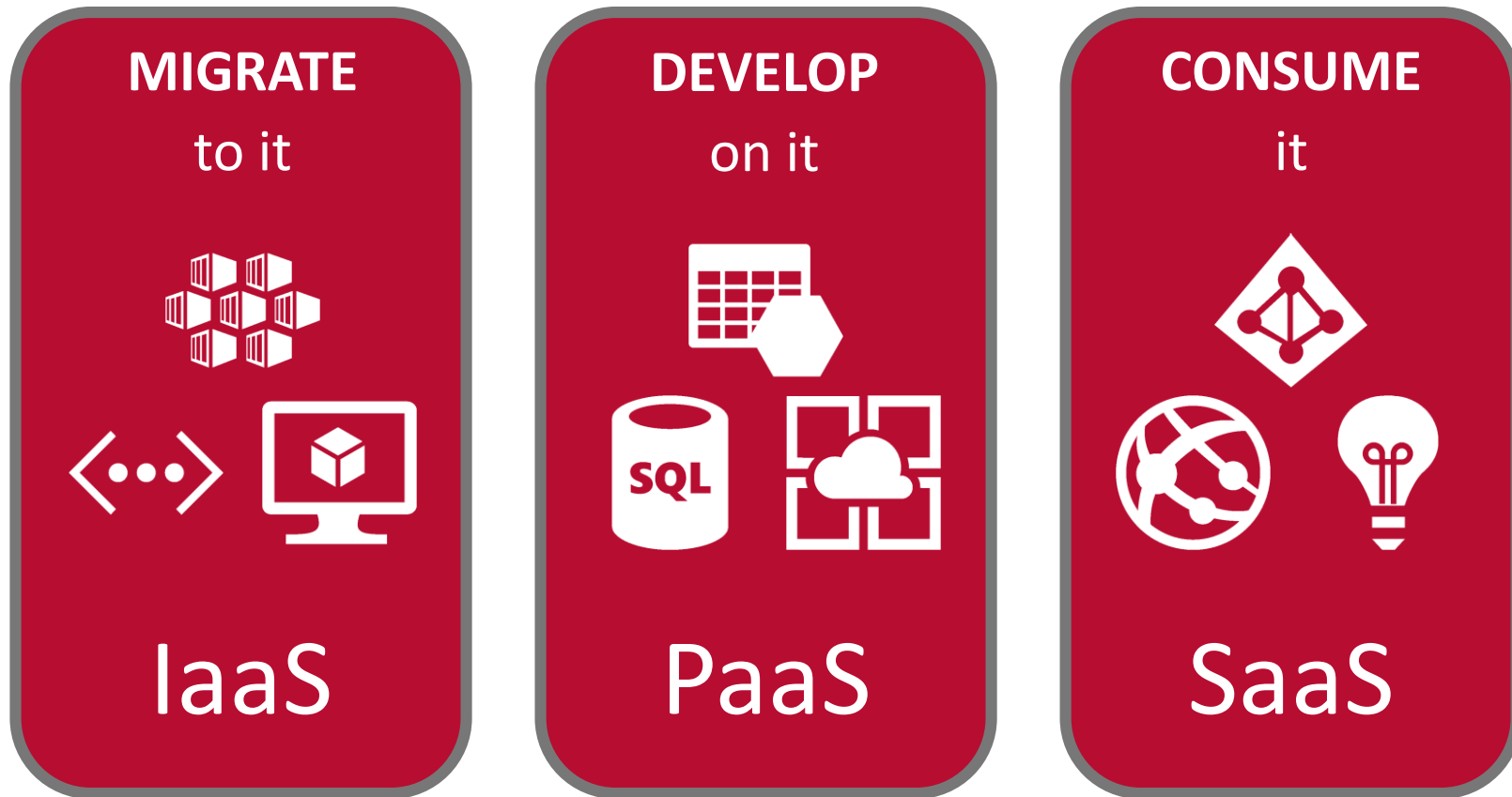
Functions as a Service: Example



Serverless vs. Alternatives

IaaS, PaaS & SaaS

Microsoft Azure



As a service?



On Premises
HaaS



IaaS



CaaS



PaaS



FaaS



SaaS

“Most PaaS applications are not geared towards bringing entire applications up and down for every request, whereas Serverless platforms do exactly this. ...”

As a service?

Datacenter

Hardware as a unit of scale

IaaS

OS as a unit of scale

PaaS

Application as a unit of scale

FaaS

Function as a unit of scale

NoOps?

NoOps?

- Serverless *might* mean 'No internal SysAdmin'
- But Ops still needed for...
 - Monitoring
 - Deployments
 - Security
 - And the rest (they normally don't have time for)

Benefits & Drawbacks

Benefits: Costs

- Economy of Scale
- BaaS:
 - Less development
 - Less infrastructure
 - Less ops
- FaaS:
 - Only pay for the compute you need

Benefits: Ops

- Simpler scaling
- Reduced packaging and deployment complexity
- Time to market / experimentation

Drawbacks: Inherent

- Vendor control
- Vendor lock-in
- Multitenancy
- Security concerns
- FaaS == stateless

Drawbacks: Implementation

- Execution duration
- Testing
- Monitoring / Debugging
- Lack of tools & experience

Tools

First wave

- Many barriers to developer productivity
 - No debugging support
 - No local development experience
 - Monitoring was hard
- Serverless became popular *despite* these downsides

Now

- Tooling and local development experience unique among cloud vendors
- Azure Functions runtime for local development
- Triggering off Azure events (hybrid debugging experience)
- Application Insights integration
 - Live event stream
 - Runtime metrics
 - Log custom metrics
 - Alerts



Microsoft Flow & Azure Logic Apps

GAB 2018

Rick van den Bosch
@rickvdbosch
@BetabitNL

© Betabit Nederland B.V.

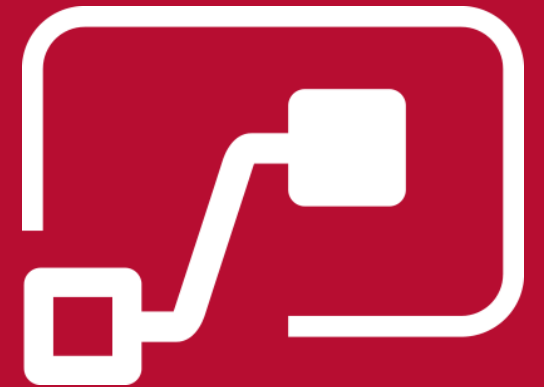
betabit {...}

Agenda

Microsoft Flow
demo

Azure Logic Apps
demo

Microsoft Flow



Flow?

*“Microsoft Flow is a product to help you set up
automated workflows between your
favorite apps and services to
synchronize files, get notifications,
collect data, and more.”*

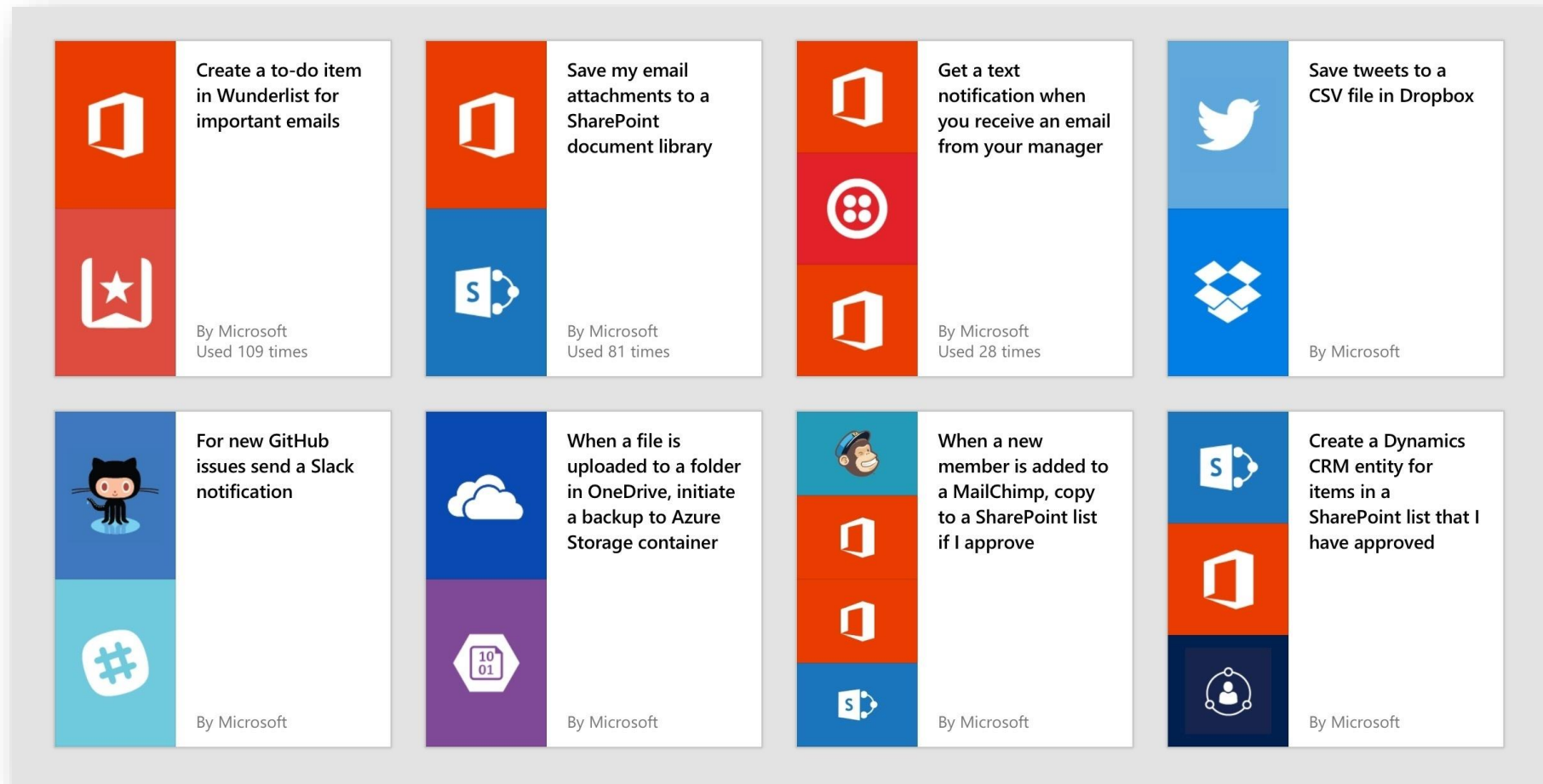
Connectors

180+ and counting

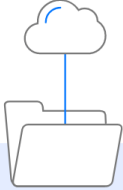
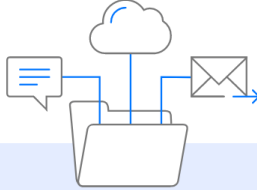

Not only Microsoft apps & services

<http://bit.ly/dwms-flowconn>

Flow templates



Microsoft Flow Pricing

Microsoft Flow for Office 365	Flow Plan 1	Flow Plan 2
		
Free	USD \$5.00 per user/mo	USD \$15.00 per user/mo
2,000 runs per month Unlimited Flow creation 5-min checks Connect to embedded Office 365 or Dynamics services	4,500 runs per month Unlimited flow creation 3-minute checks Premium Connectors	15,000 runs per month Unlimited flow creation 1-minute checks Premium Connectors Org policy settings
Current plan	Free trial Purchase now	Free trial Purchase now

Microsoft Flow Plan Features

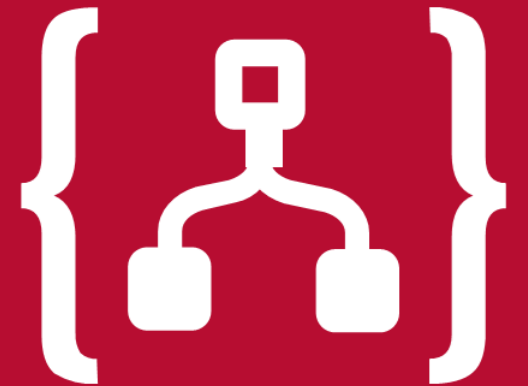
	Flow Free	Flow for Office 365 and Flow for Dynamics 365	Flow Plan 1 USD \$5.00 / user / mo	Flow Plan 2 USD \$15.00 / user / mo
	Free			
	Choose plan		Start free trial	Start free trial
General				
Create unlimited automated workflows and powerful multi-step processes	✓	✓	✓	✓
Maximum number of runs per month (per user)	750	2,000 ¹	4,500 ¹	15,000 ¹
Maximum flow frequency	15 minutes	5 minutes	3 minutes	1 minute
Access your flows on native apps for iOS and Android	✓	✓	✓	✓
Create flows from thousands of templates available in the public gallery	✓	✓	✓	✓
Publish flows to the public gallery	✓	✓	✓	✓
SLA	Not available	99.9%	99.9%	99.9%
Connectivity				
Connect to Office 365, Dynamics 365, Azure SQL, and other Microsoft services	✓	✓	✓	✓
Connect to common cloud-based services like Twitter and Wordpress using standard connectors	✓	✓	✓	✓
Connect to line of business services like Salesforce and Oracle using premium connectors			✓	✓
Access on-premises data using on-premises data gateway		✓ ²	✓	✓
Create custom connectors to connect to your own systems (per user)	One ³	One	Unlimited	Unlimited
Management				
Create environments to deploy your flows (per user)				Two
Invite others to share ownership and run flows		✓	✓	✓
View flow usage across your company				✓
Establish environment policies regarding the usage of different connections and flows ⁴				✓

Microsoft Flow

Let's code!

Actually, not really...

Azure Logic Apps



Logic Apps?

*“Automate the access and use of data
across clouds
without writing code”*

Logic Apps!



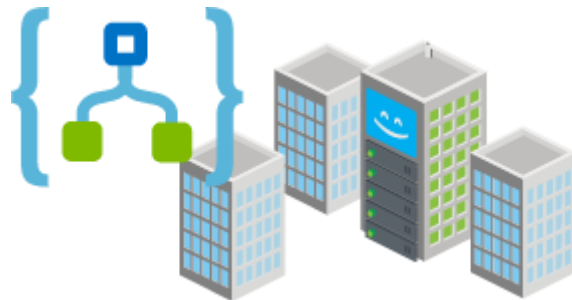
Out-of-the-box
Connectors



Connect and integrate data
from the cloud to on-prem



B2B and enterprise
messaging in the cloud

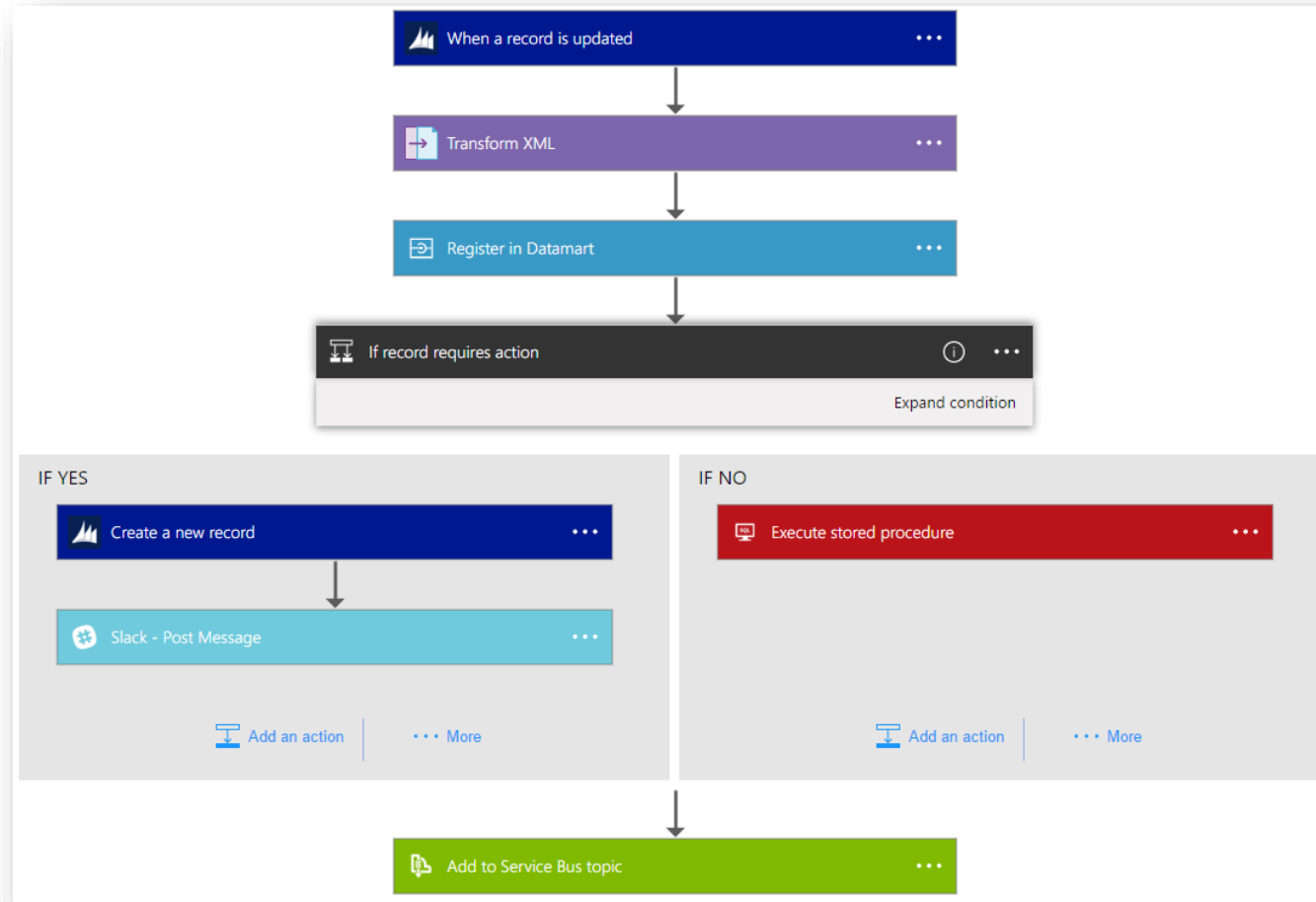


Take advantage of
BizTalk Server investments



Tap into Azure Services

Logic Apps!



Isn't this just Flow?

- Both are *configuration-first* integration services
- Flow is built on top of Logic Apps
- Flow is Business oriented, Logic Apps are developer oriented
- Logic Apps are a part of Azure

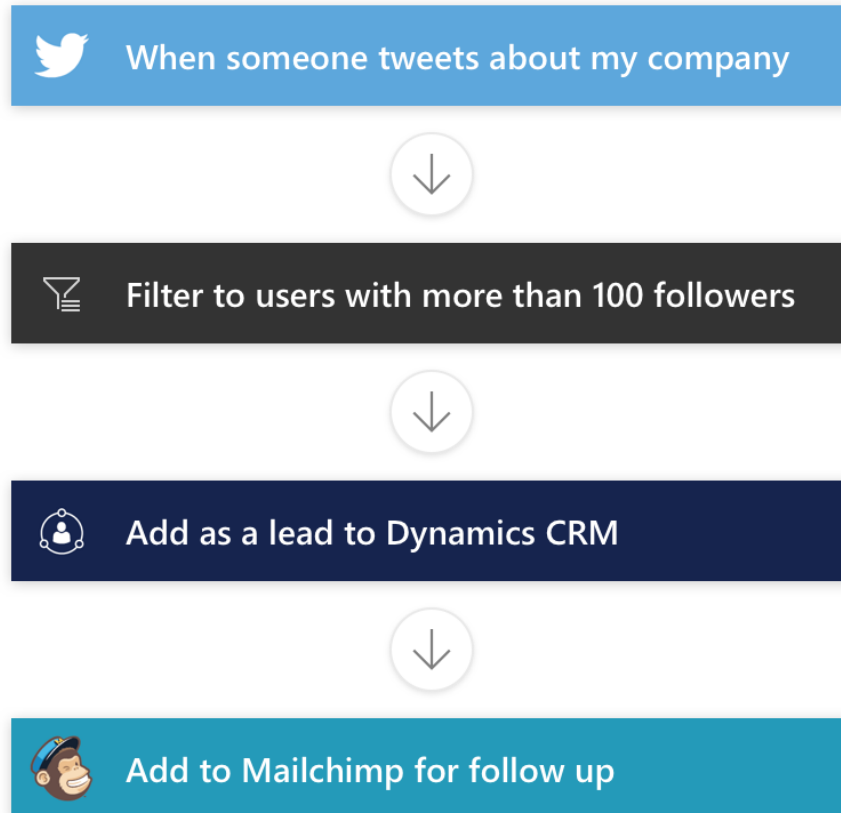
Flow empowers any office worker to perform simple integrations without going through developers or IT

Logic Apps can enable advanced or mission-critical integrations where enterprise-level DevOps and security practices are required

Flow or Logic Apps?

	Flow	Logic Apps
Users	Office workers, business users, SharePoint administrators	Pro integrators and developers, IT pros
Scenarios	Self-service	Advanced integrations
Design Tool	In-browser and mobile app, UI only	In-browser and Visual Studio , Code view available
Application Lifecycle Management (ALM)	Design and test in non-production environments, promote to production when ready.	DevOps: source control, testing, support, automation, and manageability in Azure Resource Management
Admin Experience	Manage Flow Environments and Data Loss Prevention (DLP) policies, track licensing https://admin.flow.microsoft.com	Manage Resource Groups, Connections, Access Management, and Logging https://portal.azure.com
Security	Office 365 Security and Compliance audit logs, Data Loss Prevention (DLP), encryption at rest for sensitive data, etc.	Security assurance of Azure: Azure Security , Security Center , audit logs , and more.

Example Logic app



Logic Apps Pricing

Every time a Logic App definition runs the triggers, action and connector executions are metered.

	PRICE PER EXECUTION
Actions	€0.000022
Standard Connector	€0.000106
Enterprise Connector	€0.000844

Data retention

The Integration Account is Microsoft's cloud-based solution for seamlessly integrating business functions and data sources. The Integration account enables customers to take advantage of Logic Apps B2B / EDI and XML processing capabilities.

	BASIC	STANDARD
XMLMaps	50	500
XMLSchemas	50	500
EDI Trading Partners	2	500
EDI Agreements	1	500
Price/Hour	€0.35	€1.16

Azure Logic Apps

Let's code!

Well, almost...

Hands on!

LAB: Microsoft Flow

Create a flow from a template

Use the Microsoft Flow mobile app

bit.ly/bbgab18_lab01

bit.ly/bbgab18_lab02

BONUS

Build flows

Work with approvals

Administer flows

bit.ly/bbgab18_lab03

LAB: Microsoft Logic Apps

Create logic apps – Azure Portal

Quickstart: Build your first logic app workflow

bit.ly/bbgab18_lab04

Create logic apps – Visual Studio

Quickstart: Automate tasks and processes with Azure Logic Apps

bit.ly/bbgab18_lab05

BONUS

Azure 101 Logic App lab

bit.ly/bbgab18_lab06



Azure Functions

GAB 2018

Rick van den Bosch
@rickvdbosch
@BetabitNL

© Betabit Nederland B.V.

betabit {...}

Agenda

Azure Functions
demo

Flow, Logic Apps & Functions

Function Extensions

Best Practices

Azure Functions



Azure Functions



Manage apps,
not infrastructure



Develop your way



Bind into services

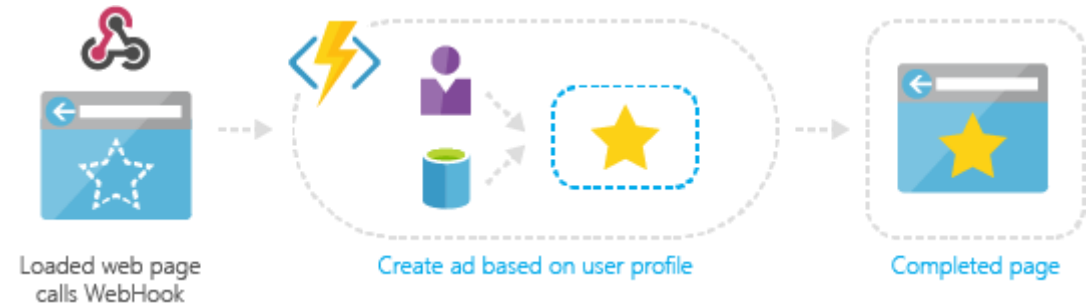
Features

- Choice of language
- Pay-per-use pricing model
- Bring your own dependencies
- Integrated security
- Simplified integration
- Flexible development
- Open-source

Language	1.x	2.x
C#	GA	Preview
JavaScript	GA	Preview
F#	GA	Preview
Java		Preview
Python	Experimental	
PHP	Experimental	
TypeScript	Experimental	
Batch (.cmd, .bat)	Experimental	
Bash	Experimental	
PowerShell	Experimental	

Available triggers

- HTTPTrigger
- TimerTrigger
- GitHub webhook
- Generic webhook
- CosmosDBTrigger
- BlobTrigger
- QueueTrigger
- EventHubTrigger
- ServiceBusQueueTrigger
- ServiceBusTopicTrigger



Available integrations

- Azure Cosmos DB
- Azure Event Hubs
- Azure Event Grid
- Azure Mobile Apps (tables)
- Azure Notification Hubs
- Azure Service Bus (queues and topics)
- Azure Storage (blob, queues, and tables)
- GitHub (webhooks)
- On-premises (using Service Bus)
- Twilio (SMS messages)

Supported bindings

The following table shows the bindings that are supported in the two major versions of the Azure Functions runtime.

Type	1.x	2.x	Trigger	Input	Output
Blob Storage	✓	✓ ¹	✓	✓	✓
Cosmos DB	✓	✓	✓	✓	✓
Event Grid	✓	✓	✓		
Event Hubs	✓	✓	✓		✓
External File ²	✓			✓	✓
External Table ²	✓			✓	✓
HTTP	✓	✓ ¹	✓		✓
Microsoft Graph Excel tables		✓		✓	✓
Microsoft Graph OneDrive files		✓		✓	✓
Microsoft Graph Outlook email		✓			✓

Supported bindings

Microsoft Graph Events		✓		✓		✓		✓
Microsoft Graph Auth tokens			✓			✓		
Mobile Apps		✓		✓			✓	✓
Notification Hubs		✓						✓
Queue storage		✓		✓ ¹		✓		✓
SendGrid		✓		✓				✓
Service Bus		✓		✓		✓		✓
Table storage		✓		✓ ¹			✓	✓
Timer		✓		✓		✓		
Twilio		✓		✓				✓
Webhooks		✓				✓		✓

Running Azure Functions

Consumption plan

When your function runs, Azure provides all of the necessary computational resources. You don't have to worry about resource management, and you only pay for the time that your code runs.

App Service plan

Run your functions just like your web, mobile, and API apps. When you are already using App Service for your other applications, you can run your functions on the same plan at no additional cost.

Routing

Default: `http://<yourapp>.azurewebsites.net/api/<funcname>`

```
JSON Copy
{
  "bindings": [
    {
      "type": "httpTrigger",
      "name": "req",
      "direction": "in",
      "methods": [ "get" ],
      "route": "products/{category:alpha}/{id:int?}"
    },
    {
      "type": "http",
      "name": "res",
      "direction": "out"
    }
  ]
}
```

Routing

Url: `http://<yourapp>.azurewebsites.net/api/products/electronics/123`

C#

Copy

```
public static Task<HttpStatusCode> Run(HttpRequestMessage req, string category, int? id,
                                     TraceWriter log)
{
    if (id == null)
        return req.CreateResponse(HttpStatusCode.OK, $"All {category} items were requested.");
    else
        return req.CreateResponse(HttpStatusCode.OK, $"{category} item with id = {id} has been requested.");
}
```

Routing

All functions routes are prefixed with **api** by default

Can be changed:

```
JSON Copy  
  
{  
  "http": {  
    "routePrefix": ""  
  }  
}
```

Throttling

- `maxOutstandingRequests`

Too many: **429 Too busy**

- `maxConcurrentRequests`

Default: **unbound**

- `dynamicThrottlesEnabled`

Checks system performance counters

Default: **false**

> 80%: **429 Too busy**

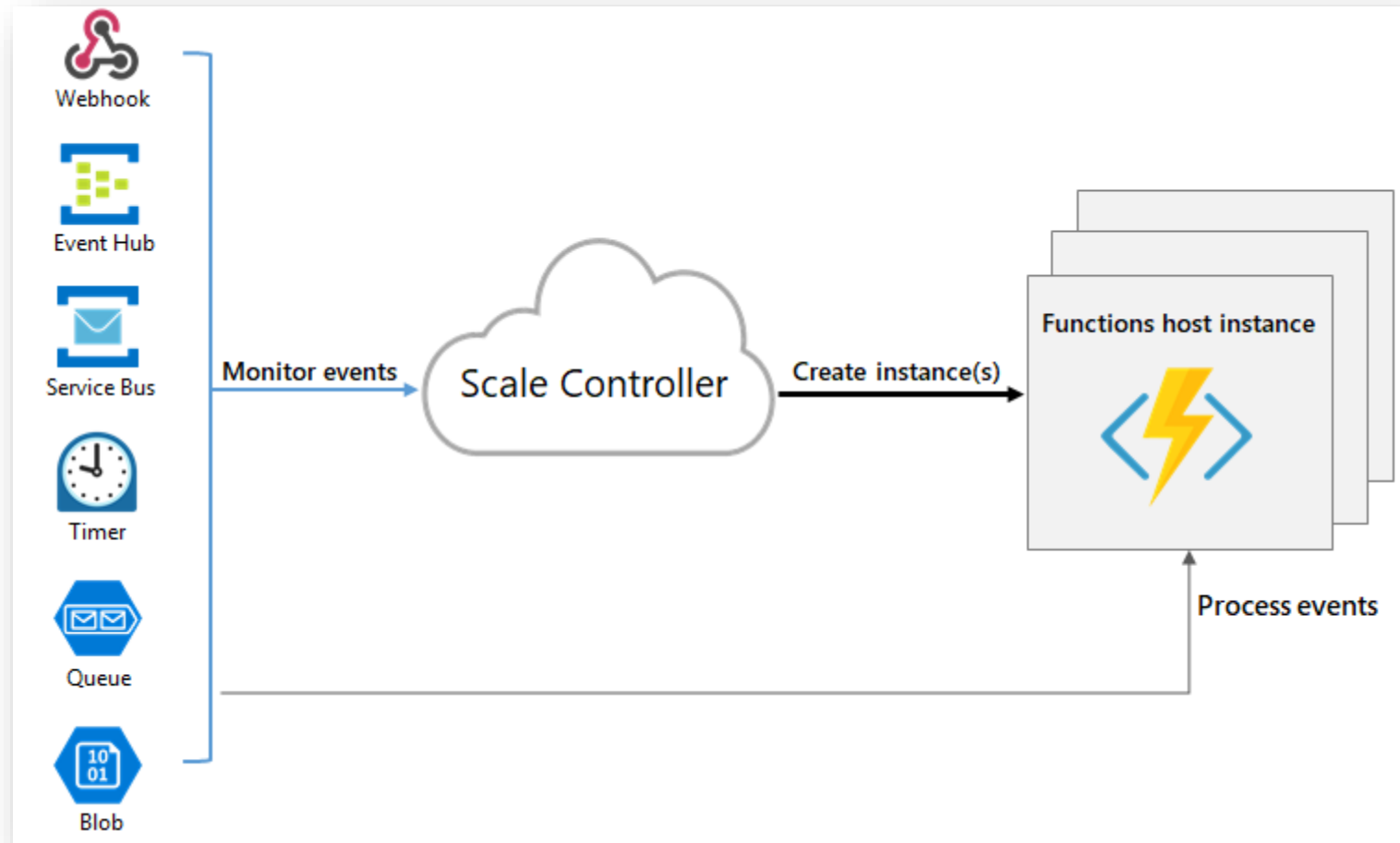
Limitations

The default timeout for Functions on a Consumption Plan is **5 minutes**

Can be increased to 10 minutes

```
1  {  
2    "functionTimeout": "00:10:00",  
3    "http": {  
4      "routePrefix": "dwms",  
5      "dynamicThrottlesEnabled": true  
6    }  
7  }
```


Scaling



Scaling behavior

Can vary on a number of factors

Can scale differently based on trigger and selected language

Constraints:

- Single function app will only scale to a maximum of 200 instances
- Single instance may process more than one message or request at a time (so there's no set limit on number of concurrent executions)
- New instances will only be allocated at most once every 10 seconds

Dual abstraction

Serverless abstracts away the compute

Bindings abstract away the services you interact with

Functions Pricing

Azure Functions pricing

Azure Functions consumption plan is billed based on **resource consumption** and **executions**. Consumption plan pricing includes a monthly free grant of 1 million requests and 400,000 GB-s of resource consumption per month. Customers can also run Functions within their App Service plan at regular App Service plan [rates](#).

Select columns ▾

METER	PRICE	FREE GRANT (PER MONTH)
Execution Time*	€0.000014/GB-s	400,000 GB-s
Total Executions*	€0.169 per million executions	1 million executions

*Free grants apply to paid, consumption subscriptions only.

Note—A storage account is created by default with each Functions app. The storage account is not included in the free grant. Standard [storage rates](#) and [networking rates](#) charged separately as applicable.

View details on [regional availability](#)

Billing example - Resources

Resource consumption (seconds)	
Executions	2 million executions
Execution duration (seconds)	× 1 second
Resource consumption Total	2 million seconds
Resource consumption (GB-s)	
Resource consumption converted to GBs	1,536 MB / 1,024 MB
Execution time (seconds)	× 2 million seconds
Total GB-s	3 million GB-s
Billable resource consumption	
Resource consumption	3 million GB-s
Monthly free grant	– 400,000 GB-s
Total billable consumption	2,600,000 GB-s
Monthly resource consumption cost	
Billable resource consumption	2,600,000 GB-s
Resource consumption price	× €0.000014/GB-s
Total cost	€35.09

Billing example - Executions

Billable executions

Total monthly executions	2 million executions
Monthly free executions	– 1 million executions
Monthly billable executions	1 million executions

Monthly executions cost

Monthly billable executions	1 million executions
Price per million executions	× €0.169
Monthly execution cost	€0.169

Total monthly cost

Monthly resource consumption cost	€35.09
Monthly executions cost	+ €0.169
Total monthly cost	€35.25

Demo

Let's code!

Finally some real code!

Flow, Logic Apps & Functions

When do I choose what?

- For simple business optimization, use Flow
- More advanced integration, DevOps need or security compliance, use Logic Apps
- Highly custom transformation or specialized code, use Functions.

Putting it all together

You can...

- ... call a logic app in a flow
- ... call a function in a logic app
- ... call a logic app in a function

*“Integration continues to improve.
Any investment you make in
these three technologies is worthwhile.”*

Function Extensions

Durable Functions

*enables writing long-running, stateful function orchestrations
in code in a serverless environment*

- Function Chaining
- Fan-out/Fan-in
- Monitors
- Human Interaction & Timeouts

<http://bit.ly/dwms-durable>

Binding Extensions

*The SDK exposes an extensibility model
that allows 3rd party extensions to be written*

Two main types of binding extensions:

Trigger Bindings

- monitor external event sources
- cause a job function to be executed when they occur

Non-Trigger Bindings

- bindings to an external system

[Binding extensions](#)

Binding Extension: SlackOutputBinding

```
[FunctionName("HttpTriggerSlack")]  
public static string Run(  
    [HttpTrigger] SlackMessage message,  
    [Slack(WebHookUrl = "SlackWebHook")] out SlackMessage slackMessage,  
    TraceWriter log)
```

Functions Best Practices

Best Practices

Long running

- Keep the runtime short (default < 5m; configure to 10m)

Stateless

- Don't use state in the host

- Idempotent

Cold start

- Fast start up times

- Keep them small

Control

- 'They' control scaling

- 'They' control when your host is alive

- You control the code!

LAB: Azure Functions

Create a Function in the Azure Portal
Functions with Visual Studio

bit.ly/bbgab18_lab07

bit.ly/bbgab18_lab08

TEST (the result of lab 08)

Mark Harrison – Colors

bit.ly/bbgab18_tst01

BONUS

Build a Serverless app in 30 minutes with
Azure Functions and Logic Apps

bit.ly/bbgab18_lab09

Azure Functions Hands-on-Lab (C#)

bit.ly/bbgab18_lab10



Azure Event Grid

GAB 2018

Rick van den Bosch
@rickvdbosch
@BetabitNL

© Betabit Nederland B.V.

betabit {...}

What is Azure Event Grid?



Azure Event Grid



Simplify event
consumption



Build reliable
cloud apps



Focus on product
innovation

Azure Event Grid

Uniform event consumption

With pub-sub

React to events

In near-real time

Azure & non-Azure

Easy to build event-based architectures

Azure Event Grid

Events as first-class objects with intelligent filtering

Filtering on event type, prefix or suffix

Built to scale

Highly available, handles massive scale automatically

Opens new Serverless possibilities

Enables event-based scenarios to span new services with ease

Lowers barriers to ops automation

Enables simpler operational and security automation

Easier policy enforcement (built-in support for Azure Automation)

Concepts

Events

Event sources

Topics

Event subscriptions

Event handlers

What?

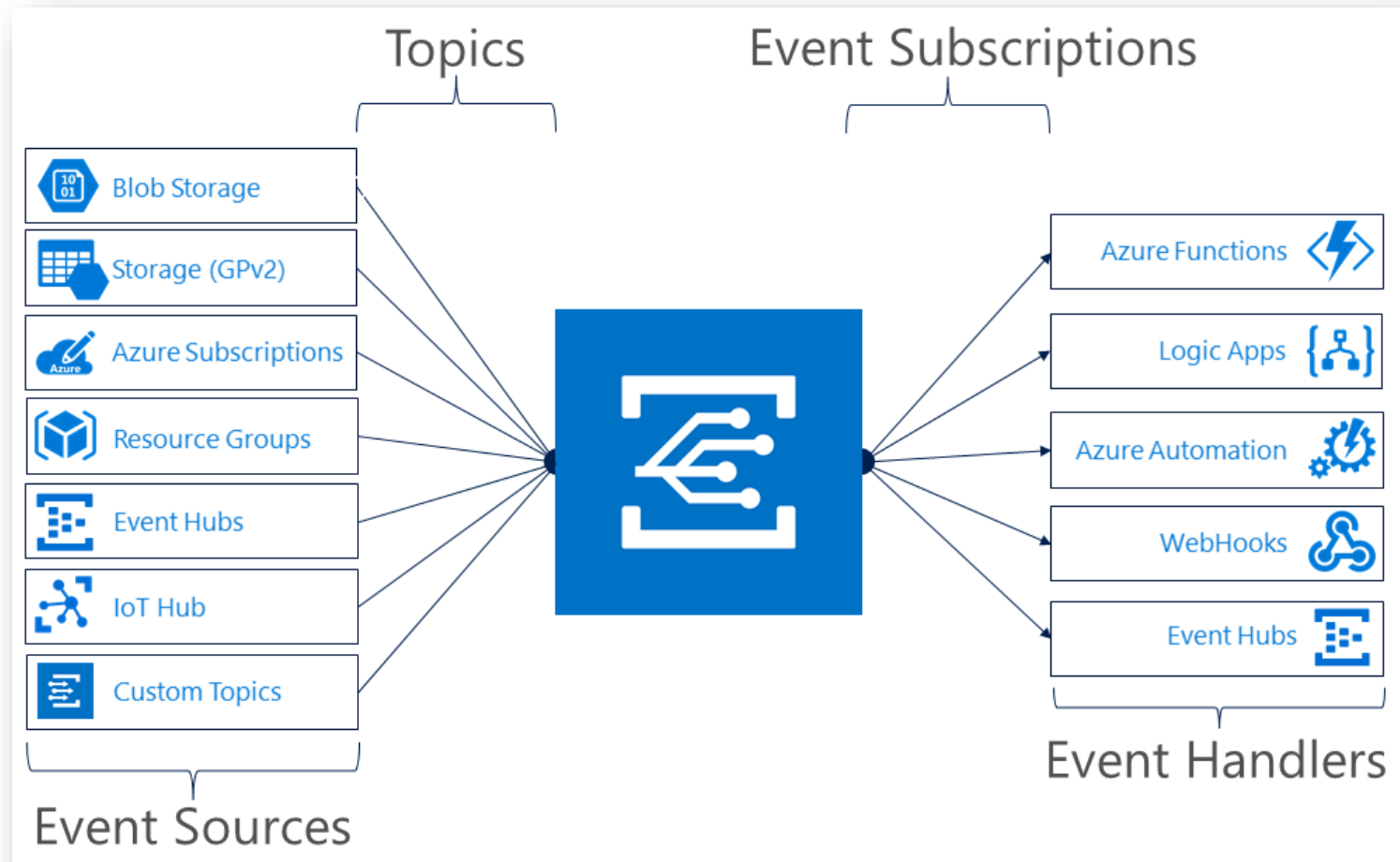
Where?

Anyone?

Where to?

Handle it!

Concepts



Capabilities

Simplicity

Point & click

Advanced filtering

Event type or event publish

Fan-out

Multiple endpoints

Reliability

24-hour retry, exponential backoff

Pay-per-event

€ 0,506 / 1 million operations (first 100K free)

High throughput

Support for millions of events / second

Built-in events

Resource-defined built-in events

Custom events

Leverage Event Grid features

Pricing example

- You publish 5 million events to Event Grid in a month.
- All events are published to 1 https endpoint.

NUMBER OF OPERATIONS	
Published events	5 million operations
Delivery attempts	5 million operations
Monthly free grant	- 100,000 operations
Total operations	9.9 million
Price per million operations	x €0.506
Total monthly cost	€5.01

Pricing example

- You publish 5 million events to Event Grid in a month.
- All events are published to 2 https endpoints.
- 1 million of the events require advanced matching.

NUMBER OF OPERATIONS	
Published events	5 million operations
Delivery attempts	10 million operations
Advanced match	1 million operations
Monthly free grant	- 100,000 operations
Total operations	15.9 million
Price per million operations	x €0.506
Total monthly cost	€8.046

Pricing example

- You publish 5 million events to Event Grid in a month.
- All events are published to 2 https endpoints.
- 1 million of the events require advanced matching.
- 1 million of the events required 2 delivery attempts.

NUMBER OF OPERATIONS

Published events	5 million operations
Delivery attempts	11 million operations (1 million for second delivery attempt)
Advanced match	1 million operations
Monthly free grant	- 100,000 operations
Total operations	16.9 million
Price per million operations	x €0.506
Total monthly cost	€8.552

Event Sources

- Azure Subscriptions (management operations)
- Custom Topics
- Event Hubs
- IoT Hub
- Resource Groups (management operations)
- Service Bus
- Storage Blob
- Storage General-purpose v2 (GPv2)

Event Handlers

- Azure Automation
- Azure Functions
- Event Hubs
- Logic Apps
- Microsoft Flow
- WebHooks

Example usage scenarios

Serverless Architectures

- Trigger a Logic App when a new blob is uploaded

Operations

- Listen & react on what happens in your subscription by subscribing to Azure Subscription changes

Integration

- Extend existing workflows by triggering a Logic App once there is a new record in your database

Custom

- Create your own by using application topics (aka custom topics)

Reacting to Blob Storage events

Azure CLI

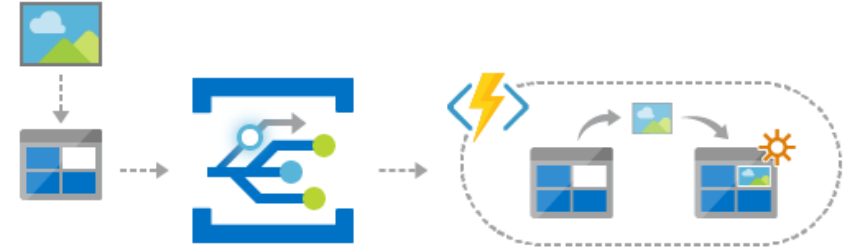
```
az provider register --namespace Microsoft.EventGrid  
az feature register --name storageEventSubscriptions --namespace  
Microsoft.EventGrid
```

Subscriptions are added to the Preview Program as capacity is available.

Example uses

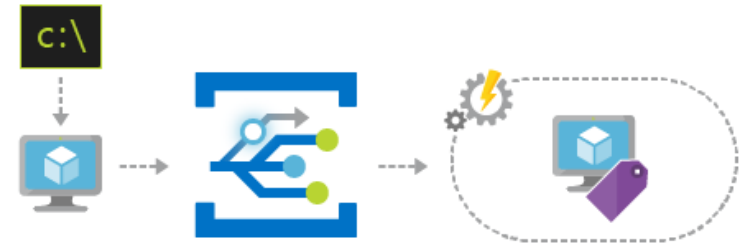
Serverless application architectures

Event Grid connects data sources and event handlers



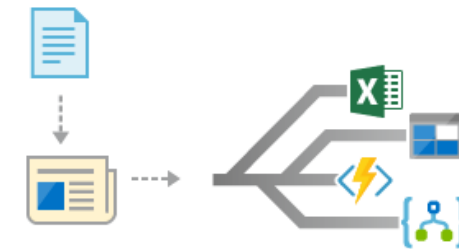
Ops automation

Event Grid allows you to speed automation and simplify policy enforcement



Application integration

Event Grid connects your app with other services



LAB: Azure EventGrid

Automate resizing uploaded images using Event Grid

bit.ly/bbgab18_lab11

Thanks for attending the

2018

Global **Azure**
BOOTCAMP

@Betabit Utrecht

Resources

[Azure Logic Apps Tools for Visual Studio 2017](#)
[Azure Architecture Center](#)

<https://mva.microsoft.com>
<https://azurecitadel.github.io/>