2018

Global Azure

BOOTCAMP

VERONA

Gli sponsor Mondiali

Microsoft

Gli sponsor del nostro evento

Platinum Sponsor

Gli sponsor del nostro evento

Gold Sponsor

Gli sponsor del nostro evento

digital joy

dgroove
Tecnologia emozionale.

Basic Sponsor

Tweet della giornata

#GlobalAzure

@cloudgen_verona

2018 Global Azure BOOTCAMP

ARGOMENTO

Scalare le proprie applicazioni con Azure Functions

ANDREA TOSATO

@ATosato86

andreatosato

andreatosato

2018 Global Azure BOOTCAMP

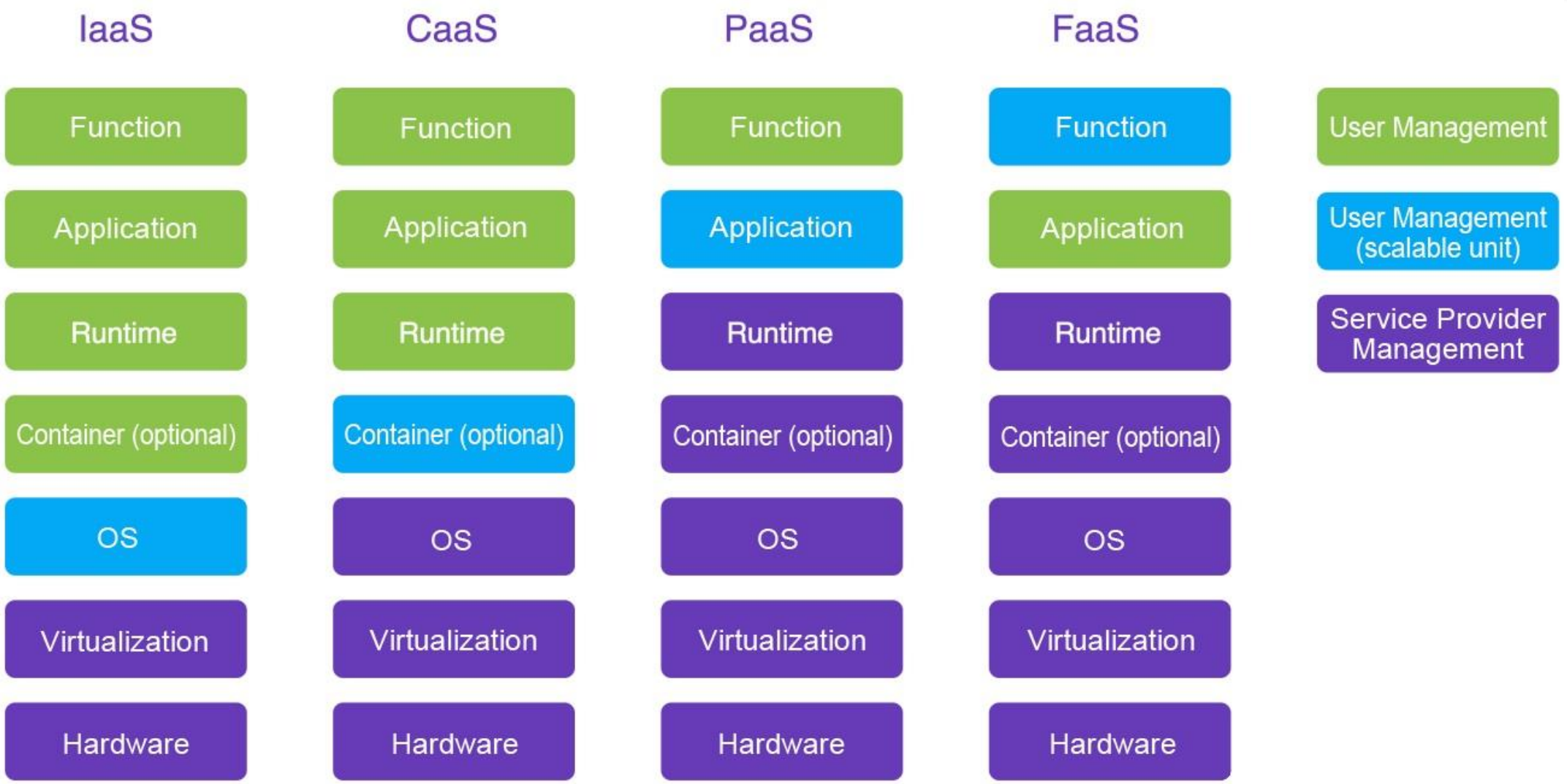
ARGOMENTO

Introduzione

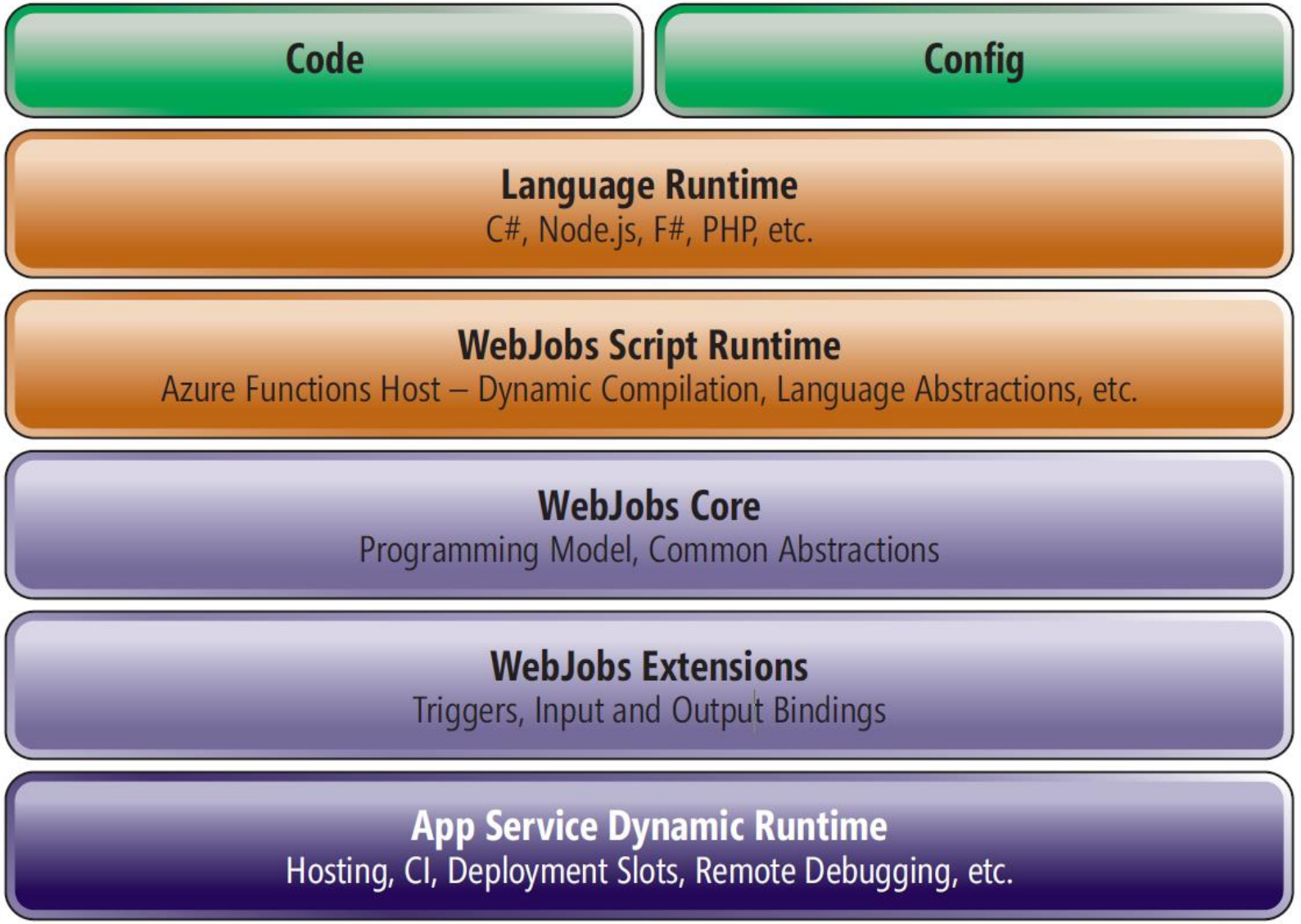
Primi passi con Azure Functions

Il contesto

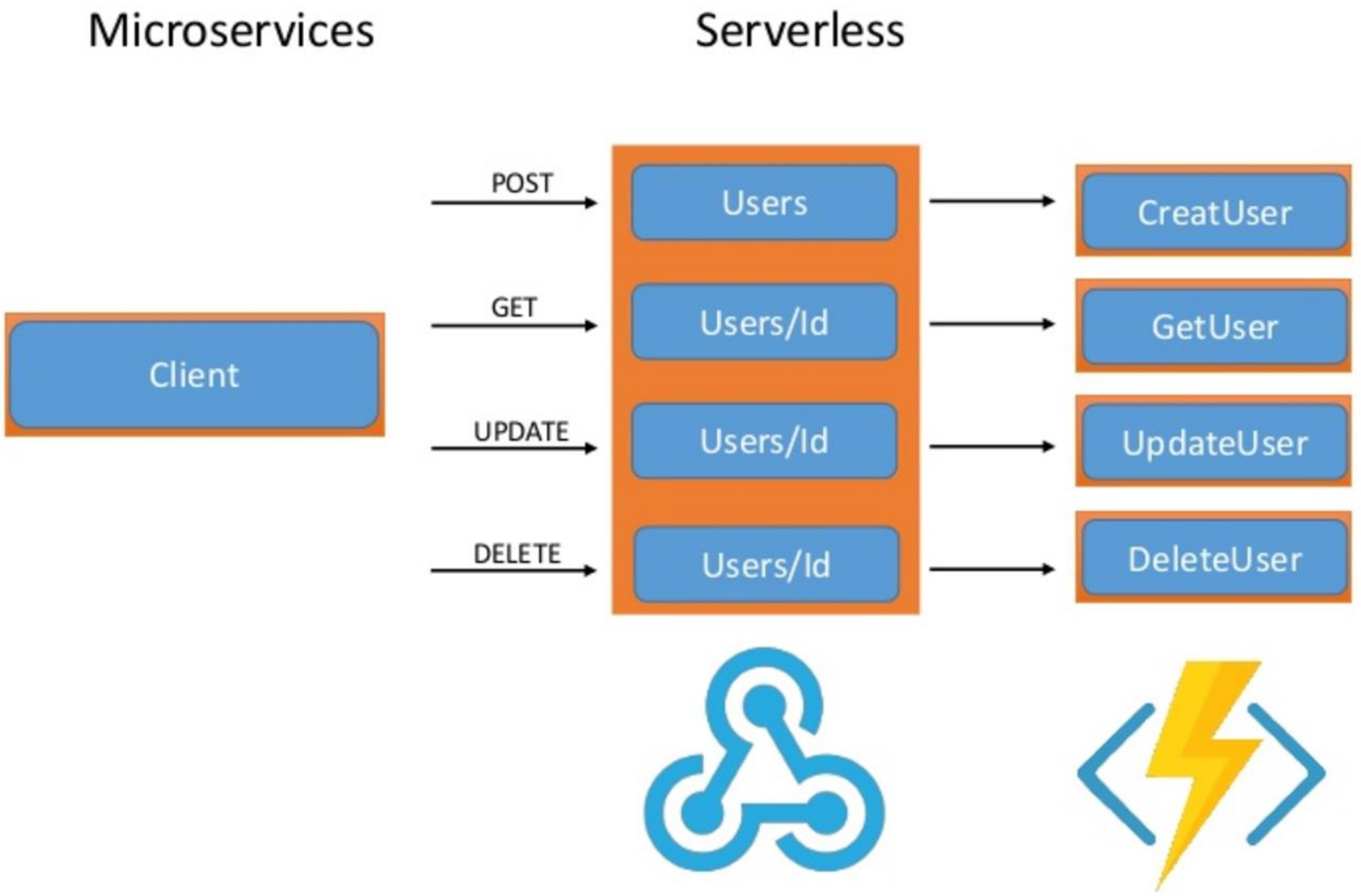

2018
Global Azure
BOOTCAMP
CloudGen


IaaS
CaaS
PaaS
FaaS
Function
Application
Runtime
Container (optional)
OS
Virtualization
Hardware
Function
Application
Runtime
Container (optional)
OS
Virtualization
Hardware
Function
Application
Runtime
Container (optional)
OS
Virtualization
Hardware
Function
Application
Runtime
Container (optional)
OS
Virtualization
Hardware
User Management
User Management (scalable unit)
Service Provider Management

Il contesto


Code
Config
Language Runtime
C#, Node.js, F#, PHP, etc.
WebJobs Script Runtime
Azure Functions Host – Dynamic Compilation, Language Abstractions, etc.
WebJobs Core
Programming Model, Common Abstractions
WebJobs Extensions
Triggers, Input and Output Bindings
App Service Dynamic Runtime
Hosting, CI, Deployment Slots, Remote Debugging, etc.

Da microservizi a functions

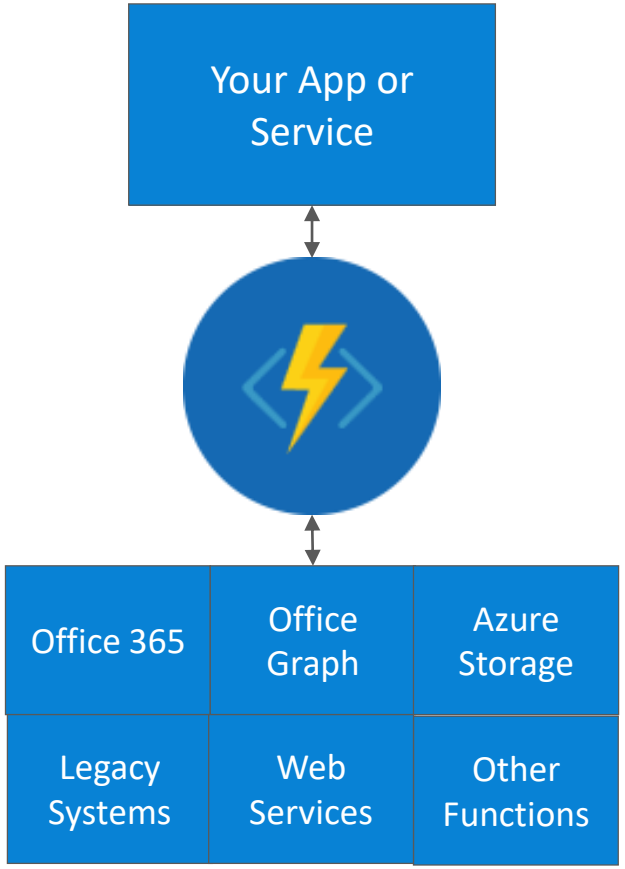


Linguaggi supportati

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

Scenari di utilizzo

- Timer-based processing
- Azure service event processing
- SaaS event processing
- Serverless web application architectures
- Serverless mobile backends
- Real-time stream processing
- Real-time bot messaging



Timer Function Apps

- Run at explicitly specified intervals, like every day at 2:00 am using **CRON** expressions, like “0 */5 * * * *” (every 5 minutes)
- Can send information to other systems, but typically don’t “return” information, only write to logs
- Great for redundant cleanup and data management
- Great for checking state of services
- Can be combined with other functions

Webhook & API Function Apps

Triggered by **events** in other services, like *GitHub, Team Foundation Services, Office 365, OneDrive, Microsoft PowerApps*

Takes in a request and sends back a response
Often mimic **Web API** and legacy web services flows

Typically need **CORS** settings managed

Great for building **Logic Apps**

Demo

Creazione di una functions, template disponibili.
Progetti attribute e settings

Binding supportati

type	1.x	2.x	Trigger	Input	Output
Archiviazione BLOB	✔	✔[1]	✔	✔	✔
Cosmos DB	✔	✔	✔	✔	✔
Griglia di eventi	✔	✔	✔		
Hub eventi	✔	✔	✔		✔
File esterno[2]	✔			✔	✔
Tabella esterna[2]	✔			✔	✔
HTTP	✔	✔[1]	✔		✔
App per dispositivi mobili	✔	✔		✔	✔
Hub di notifica di Azure	✔				✔
Archiviazione code	✔	✔[1]	✔		✔
Bus di servizio	✔	✔	✔		✔
Archiviazione tabelle	✔	✔[1]		✔	✔
Timer	✔	✔	✔		
Webhook	✔		✔		✔

type	1.x	2.x	Trigger	Input	Output
Microsoft Graph Tabelle di Excel		✔		✔	✔
Microsoft Graph File di OneDrive		✔		✔	✔
Microsoft Graph Indirizzo e-mail Outlook		✔			✔
Microsoft Graph Eventi		✔	✔	✔	✔
Microsoft Graph Token di autenticazione		✔		✔	

type	1.x	2.x	Trigger	Input	Output
SendGrid	✔	✔			✔
Twilio	✔	✔			✔

https://docs.microsoft.com/it-it/azure/azure-functions/functions-triggers-bindings

File esterno

Preview

Connettore	Trigger	Input	Output
Box	x	x	x
Dropbox	x	x	x
FTP	x	x	x
OneDrive	x	x	x
OneDrive for Business	x	x	x
SFTP	x	x	x
Google Drive		x	x

Tabella esterna

Preview

Connettore	Trigger	Input	Output
DB2		x	x
Dynamics 365 for Operations		x	x
Dynamics 365		x	x
Dynamics NAV		x	x
Fogli Google		x	x
Informix		x	x
Dynamics 365 for Financials		x	x
MySQL		x	x
Oracle Database		x	x
Common Data Service		x	x

Connettore	Trigger	Input	Output
Salesforce		x	x
SharePoint		x	x
SQL Server		x	x
Teradata		x	x
UserVoice		x	x
Zendesk		x	x

External Table binding


```
{
  "bindings": [
    {
      "type": "manualTrigger",
      "direction": "in",
      "name": "input"
    },
    {
      "type": "apiHubTable",
      "direction": "in",
      "name": "table",
      "connection": "ConnectionAppSettingsKey",
      "dataSetName": "default",
      "tableName": "Contact",
      "entityId": "",
    }
  ],
  "disabled": false
}
```

```
public class Contact
{
    public string Id { get; set; }
    public string LastName { get; set; }
    public string FirstName { get; set; }
}
```

```
CREATE TABLE Contact
(
    Id int NOT NULL,
    LastName varchar(20) NOT NULL,
    FirstName varchar(20) NOT NULL,
    CONSTRAINT PK_Contact_Id PRIMARY KEY (Id)
)
GO
```

External Table binding

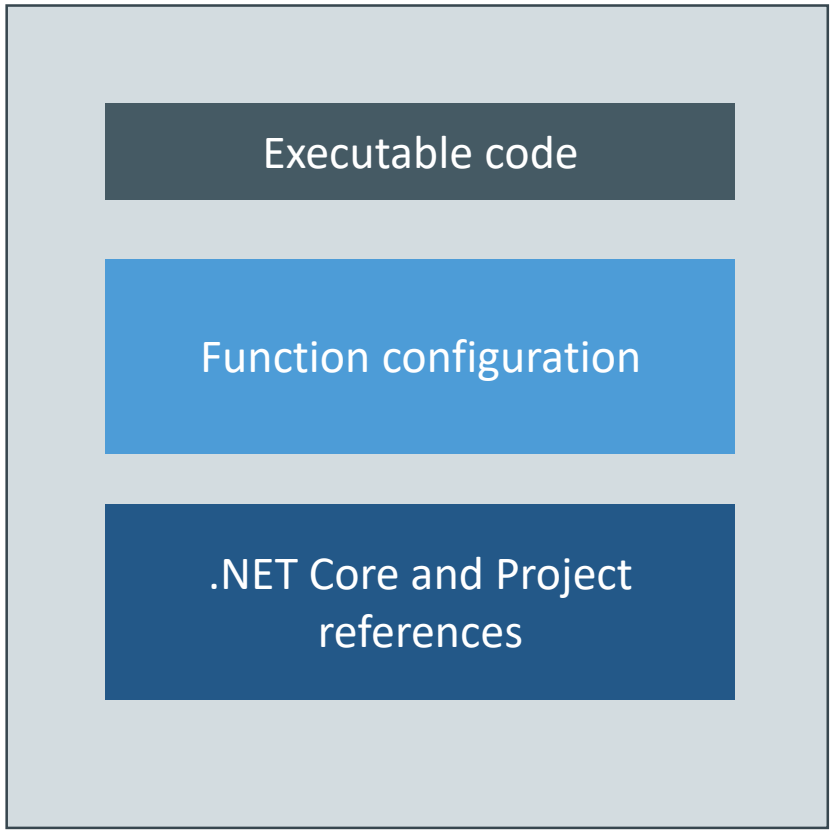

```
public static async Task Run(string input, ITable<Contact> table, TraceWriter log)
{
    //Iterate over every value in the source table
    ContinuationToken continuationToken = null;
    do
    {
        //retrieve table values
        var contactsSegment = await table.ListEntitiesAsync(continuationToken: continuationToken);

        foreach (var contact in contactsSegment.Items)
        {
            log.Info(string.Format("{0} {1}", contact.FirstName, contact.LastName));
        }
        continuationToken = contactsSegment.ContinuationToken;
    }
    while (continuationToken != null);
}
```

```
#r "Microsoft.Azure.ApiHub.Sdk"
#r "Newtonsoft.Json"
using System;
using Microsoft.Azure.ApiHub;
```

Anatomia di una Function

- A “Run” file that containing the function code (**static class**)
- A “Function” file containing all service and trigger bindings and parameters (**function.json**)
- A “Project” file containing project assembly and NuGet package references
- App Service settings, such as connection strings and API keys



Il file host

http://json.schemastore.org/host

Demo

Il progetto e i suoi file

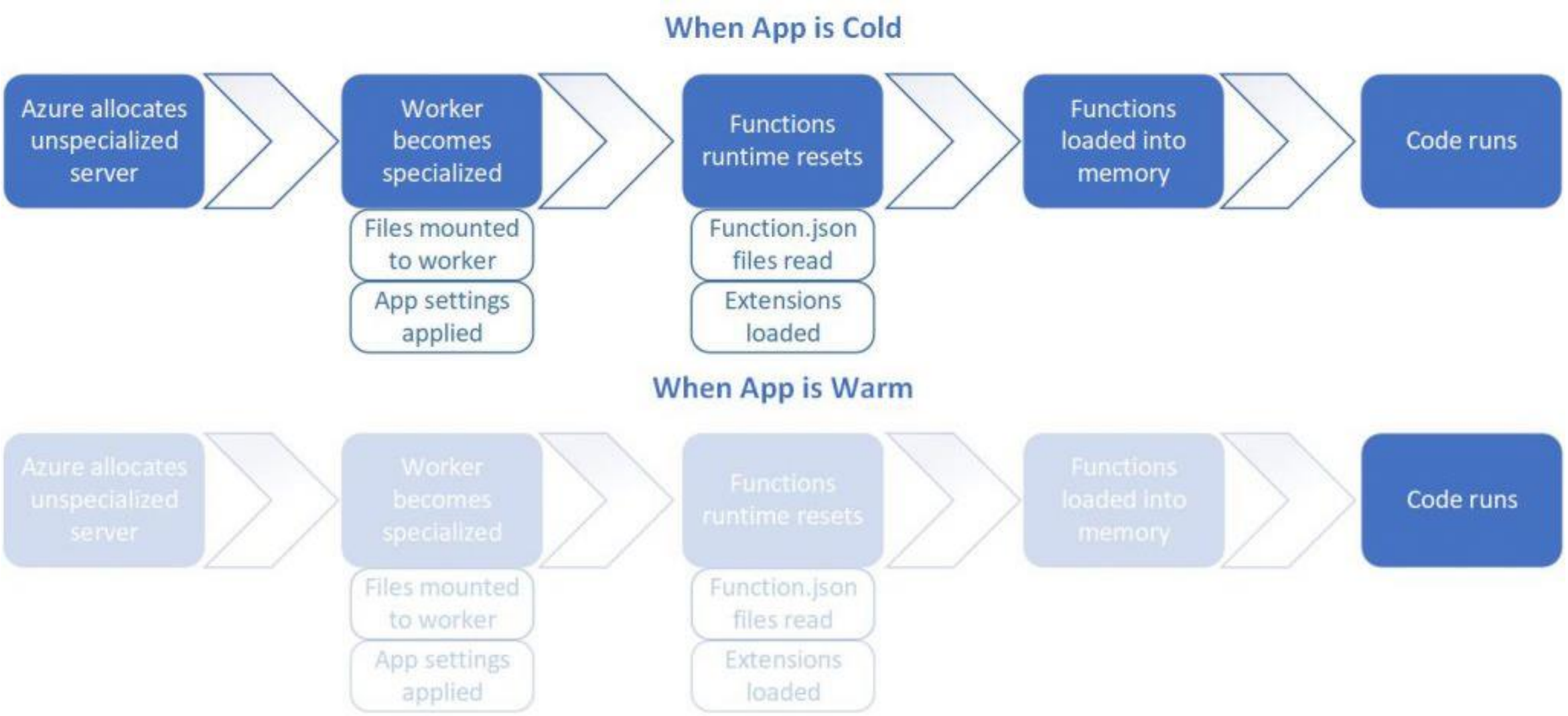
2018 Global Azure BOOTCAMP

ARGOMENTO

Consumption Plan

Come ottimizzare una applicazione per sfruttare al meglio il piano di servizi a consumo

Cold vs Warm


When App is Cold
Azure allocates unspecialized server
Worker becomes specialized
Files mounted to worker
App settings applied
Functions runtime resets
Function.json files read
Extensions loaded
Functions loaded into memory
Code runs
When App is Warm
Azure allocates unspecialized server
Worker becomes specialized
Files mounted to worker
App settings applied
Functions runtime resets
Function.json files read
Extensions loaded
Functions loaded into memory
Code runs

Cold vs Warm

Il **consumption plan** è il vero modello “serverless”
Reagisce agli eventi e scala all’occorenza. Viene pagato per singolo utilizzo.
Tutto avviene senza preoccuparsi del gestore del servizio.

Il **dedicated plan**, risiede sulla virtual machine dedicata.
Consente un controllo maggiore poichè risiede su una propria macchina.
E’ sempre disponibile ed è preferibile utilizzaro:

- Macchina VMs sotto utilizzata
- Esecuzione continua della Functions e un maggior controllo dei costi. (polling continuo)
- Più CPU e memoria
- Esecuzione più lunga di 5 minuti (configurazione standard) o 10 (massimo contentiti)
- Richiede funzioni disponibili solo sull’App Service Plan (VNET/VPN)

https://blogs.msdn.microsoft.com/appserviceteam/2018/02/07/understanding-serverless-cold-start/

Pricing

Azure Functions pricing

Azure Functions consumption plan is billed based on per-second 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.

METER	PRICE	FREE GRANT (PER MONTH)
Execution Time*	$0.000016/GB-s	400,000 GB-s
Total Executions*	$0.20 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.

Demo

Scalabilità tramite Load Test

https://github.com/andreatosato/GlobalAzureBootcampApp

Load Testing 2 minuti

Impostazioni del test

Run duration (minutes)	2
Load pattern	Constant
Max v-users	100
Warmup duration (seconds)	0
Browser mix	IE - 40%, Chrome - 60%

Load Testing 2 minuti

Summary

AVG. RESPONSE TIME	USER LOAD	REQUESTS PER SEC	FAILED REQUESTS	ERRORS →	USAGE
4.6 sec	100 users	17.9 RPS	0 %	0 errors	250 VUMs
			0 failed requests 2145 total requests	0 thresholds violated	

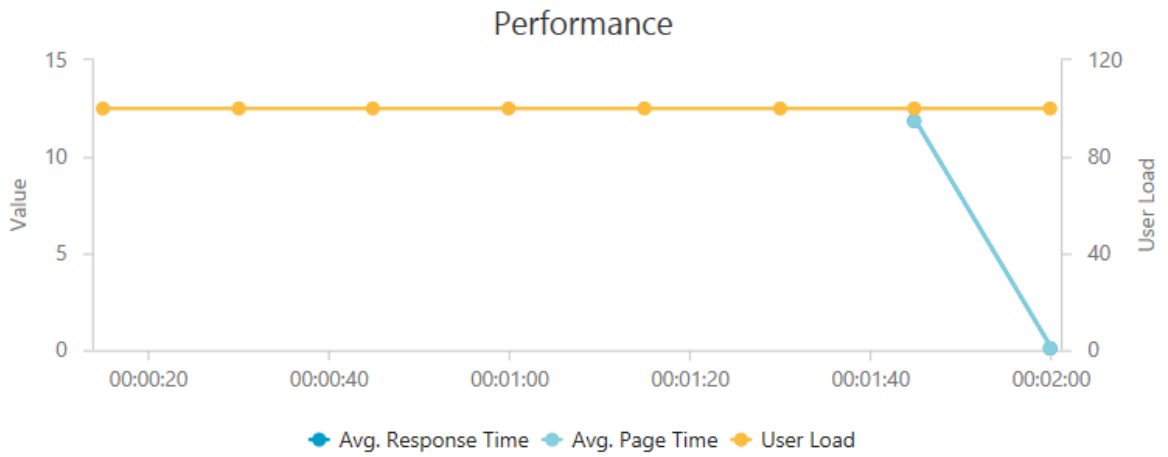
Learn more about metrics and criteria

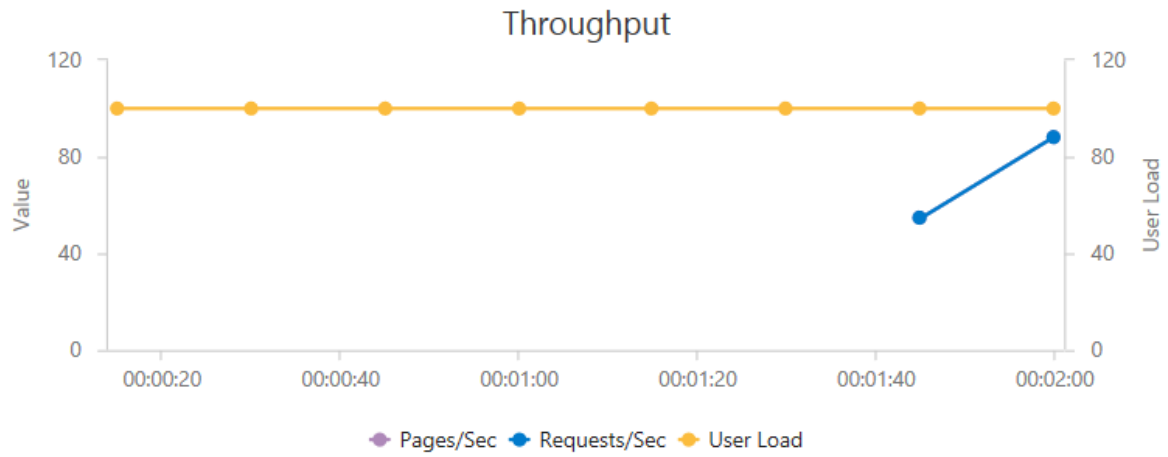
Test settings

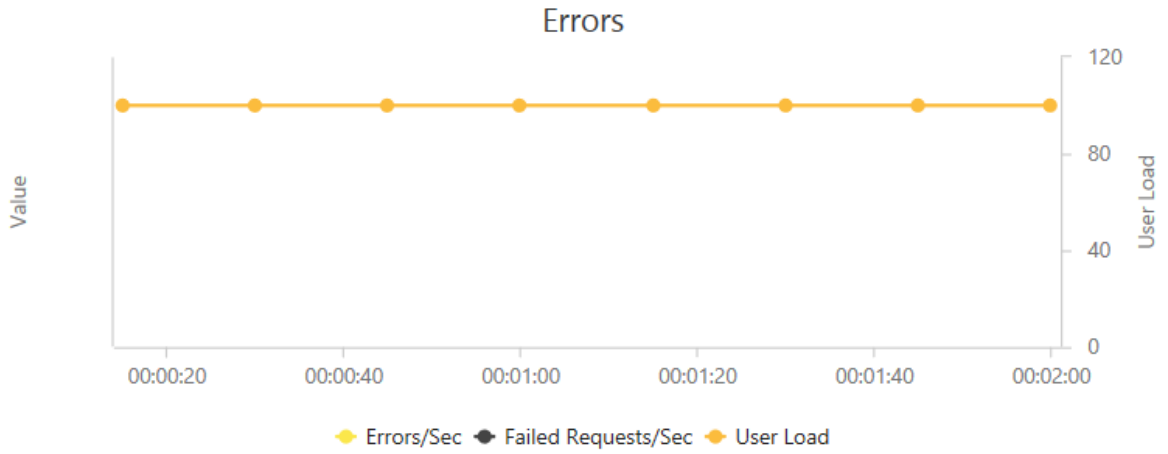
Load duration:	2 min	**Requested by:**	Andrea Tosato	**Run source:**	Team Services portal
Start time:	4/2/2018 5:32:18 PM	**Test:**	GAB2018VR_2Minutes	**Warmup duration:**	-
End time:	4/2/2018 5:34:20 PM	**Location:**	West Europe	**Agent cores:**	1

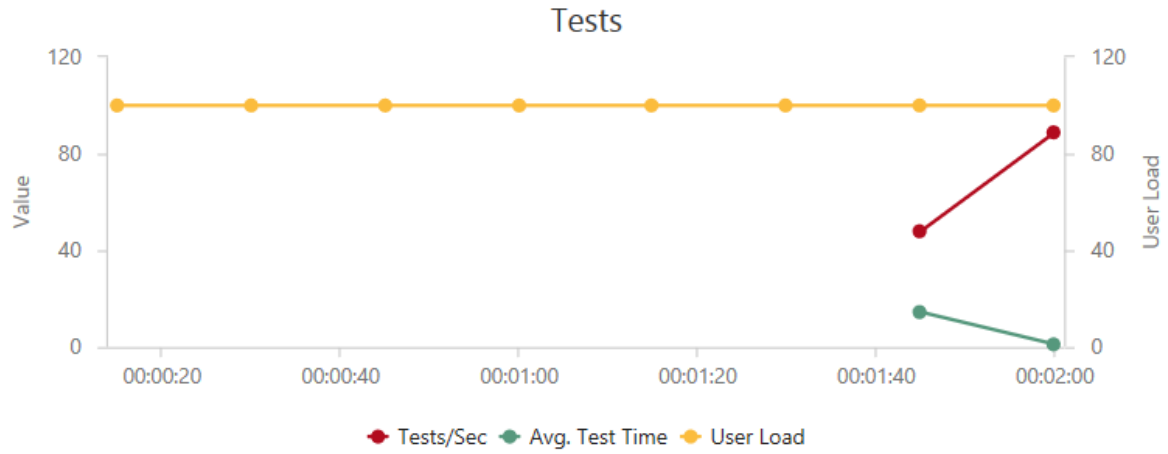
Load Testing 2 minuti

Chart



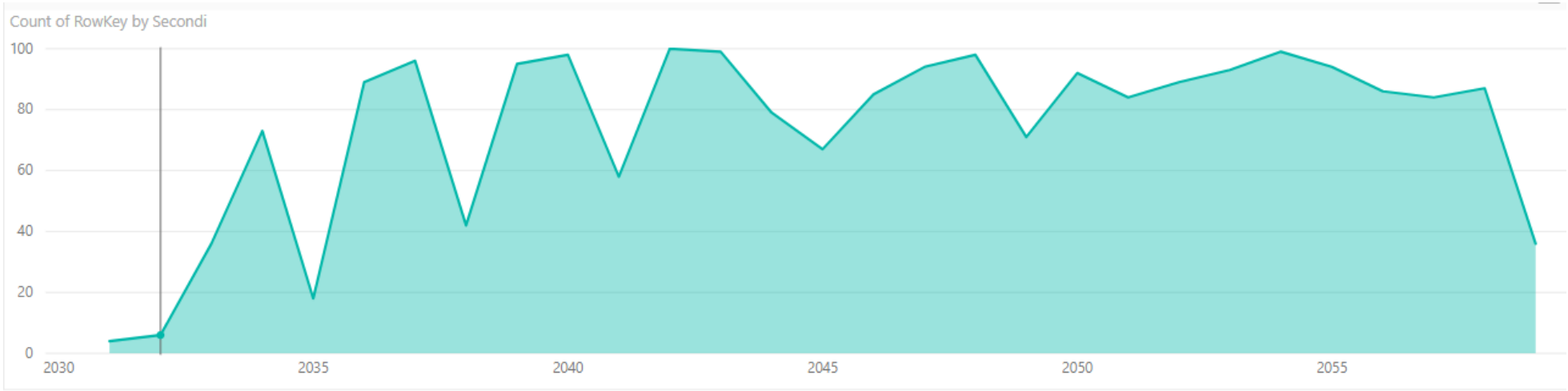


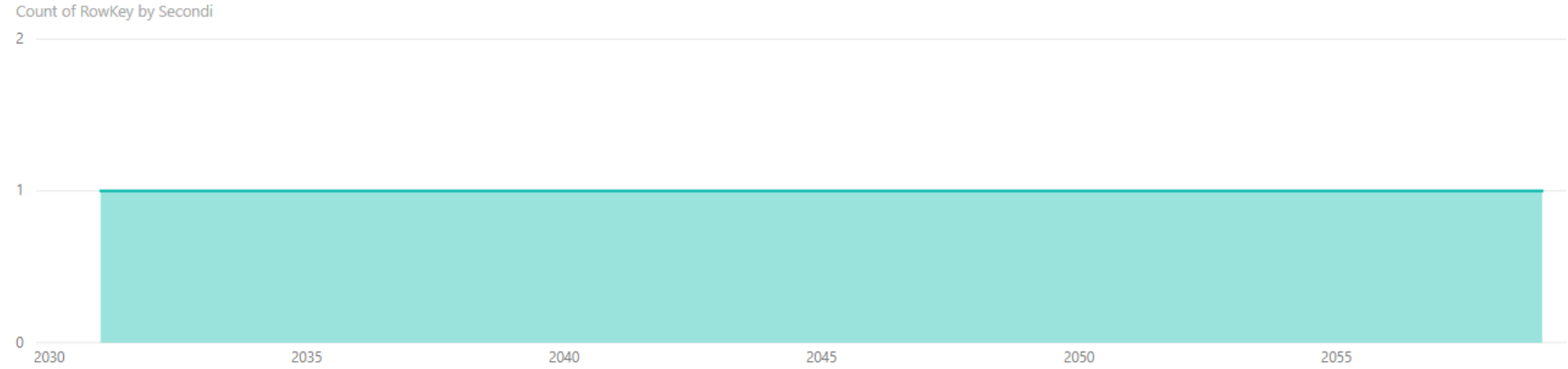




Load Testing 2 minuti

Richieste - Istanze





Load Testing 5 minuti

Impostazioni del test

Run duration (minutes)	5
Load pattern	Step
Max v-users	500
Start user count	10
Step duration (seconds)	10
Step user count (users/step)	10
Warmup duration (seconds)	0
Browser mix	IE - 40%, Chrome - 60%

Load Testing 5 minuti

Summary

AVG. RESPONSE TIME	USER LOAD	REQUESTS PER SEC	FAILED REQUESTS	ERRORS →	USAGE
167.7 ms	300 users	131.9 RPS	0 %	0 errors	1.5K VUMs
			0 failed requests 39581 total requests	0 thresholds violated	

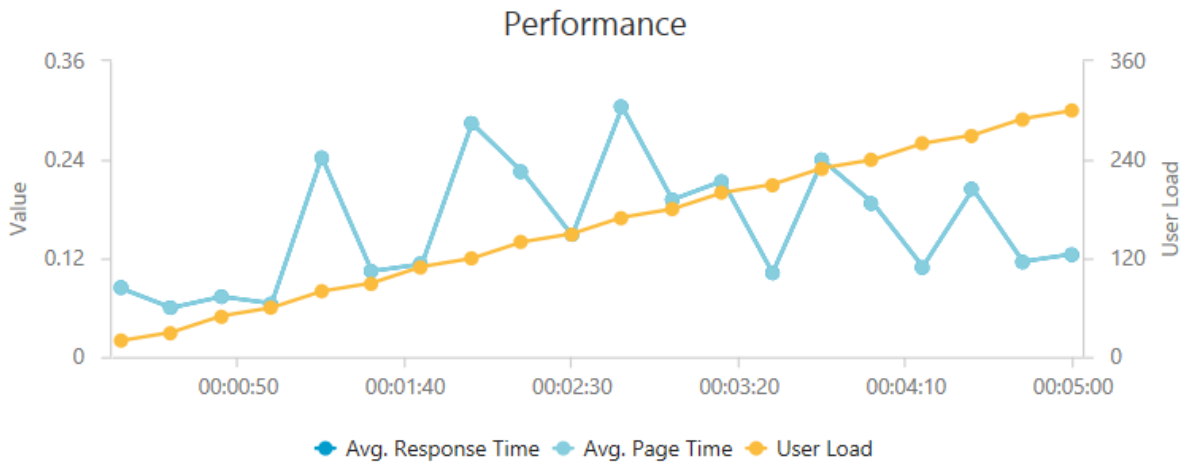
Learn more about metrics and criteria

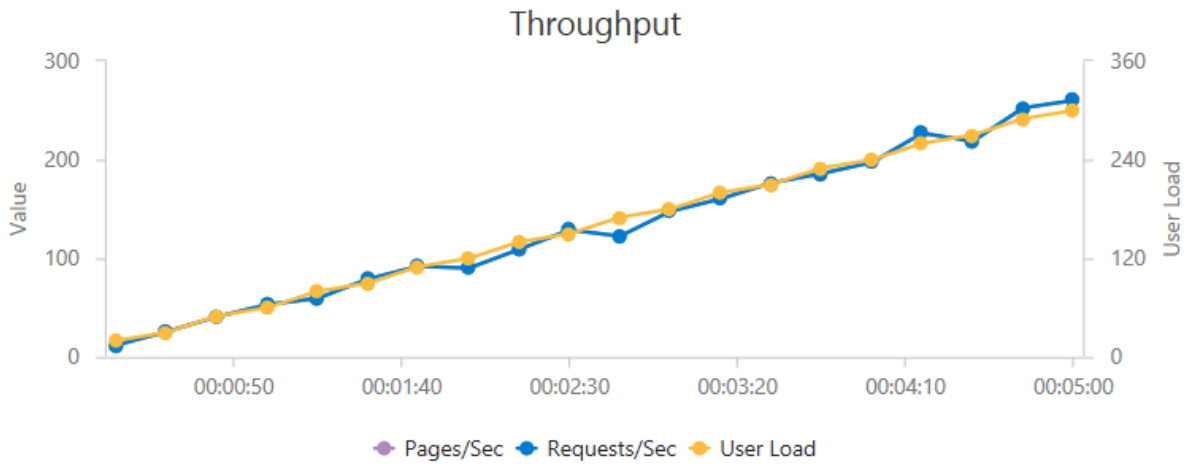
Test settings

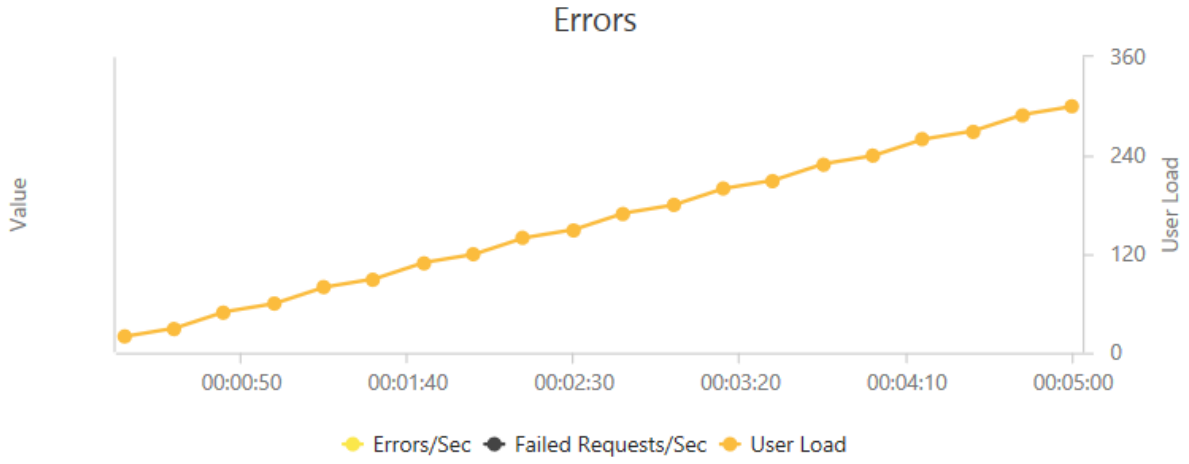
Load duration:	5 min	**Requested by:**	Andrea Tosato	**Run source:**	Team Services portal
Start time:	4/2/2018 5:39:38 PM	**Test:**	GABVR_5Minutes	**Warmup duration:**	-
End time:	4/2/2018 5:44:40 PM	**Location:**	West Europe	**Agent cores:**	1

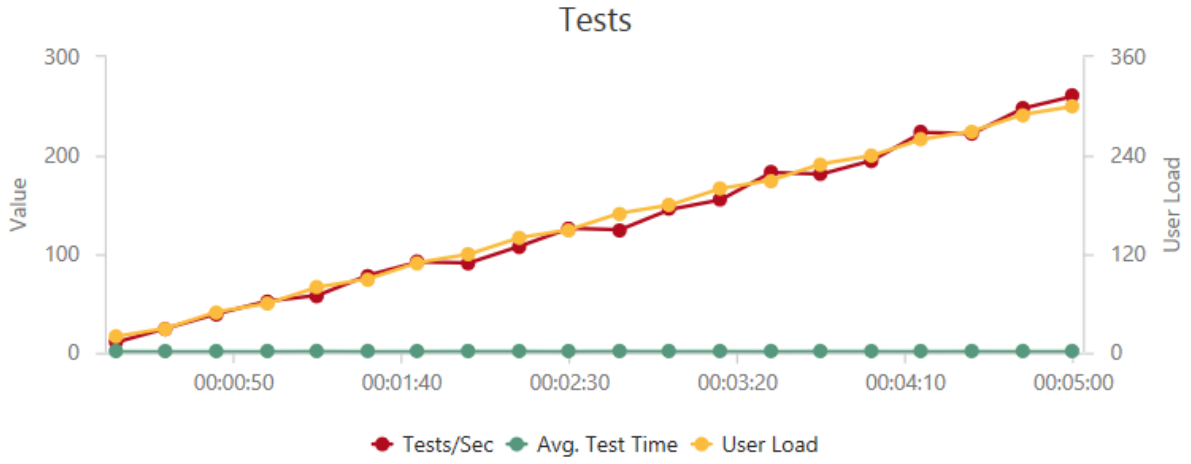
Load Testing 5 minuti

Chart



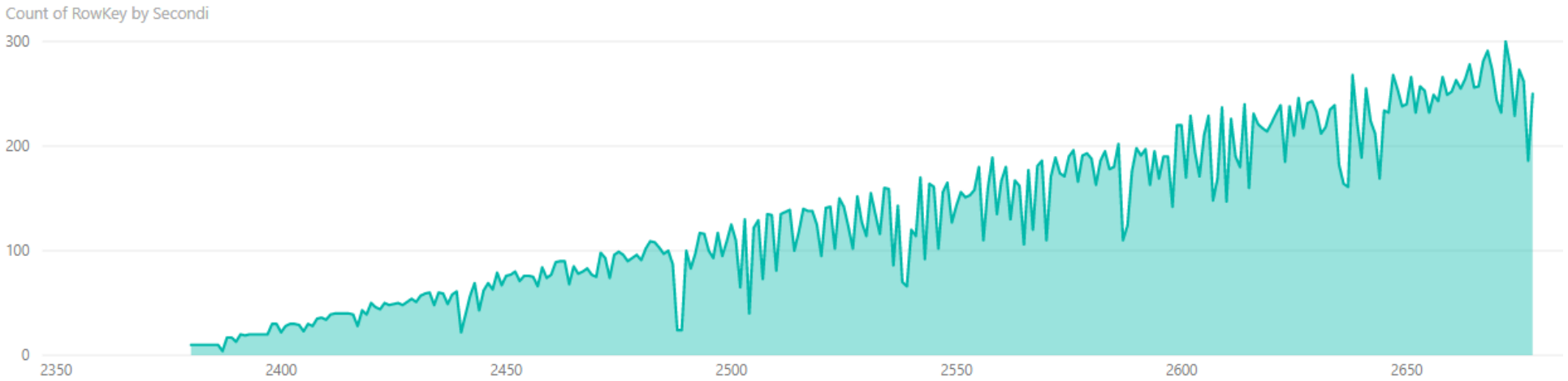


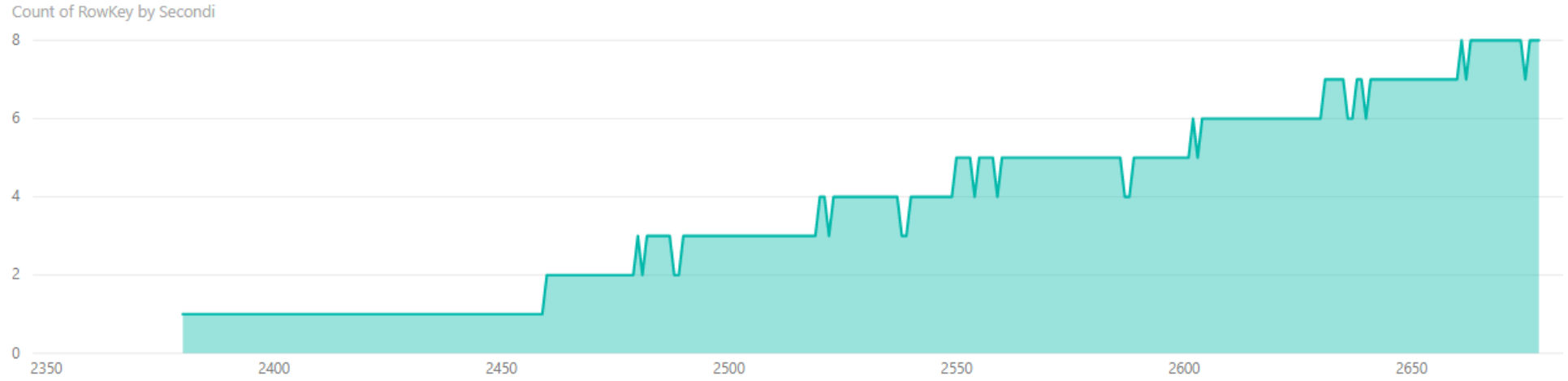




Load Testing 5 minuti

Richieste - Istanze





Load Testing 10 minuti

Impostazioni del test

Run duration (minutes)	10
Load pattern	Step
Max v-users	400
Start user count	10
Step duration (seconds)	20
Step user count (users/step)	10
Warmup duration (seconds)	0
Browser mix	IE - 40%, Chrome - 60%

Load Testing 10 minuti

Summary

AVG. RESPONSE TIME	USER LOAD	REQUESTS PER SEC	FAILED REQUESTS	ERRORS →	USAGE
96.4 ms	300 users	139.7 RPS	0 %	0 errors	3K VUMs
			0 failed requests 83825 total requests	0 thresholds violated	

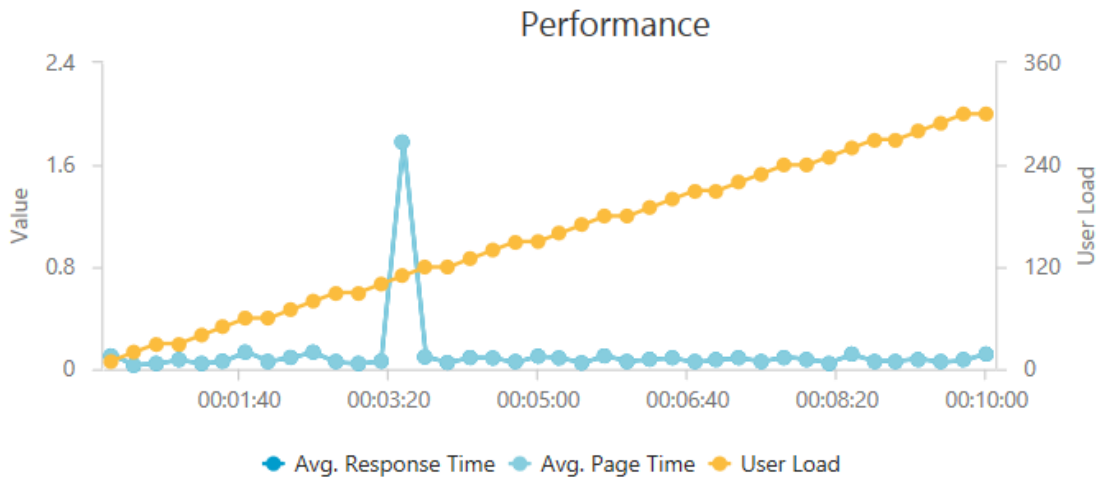
Learn more about metrics and criteria

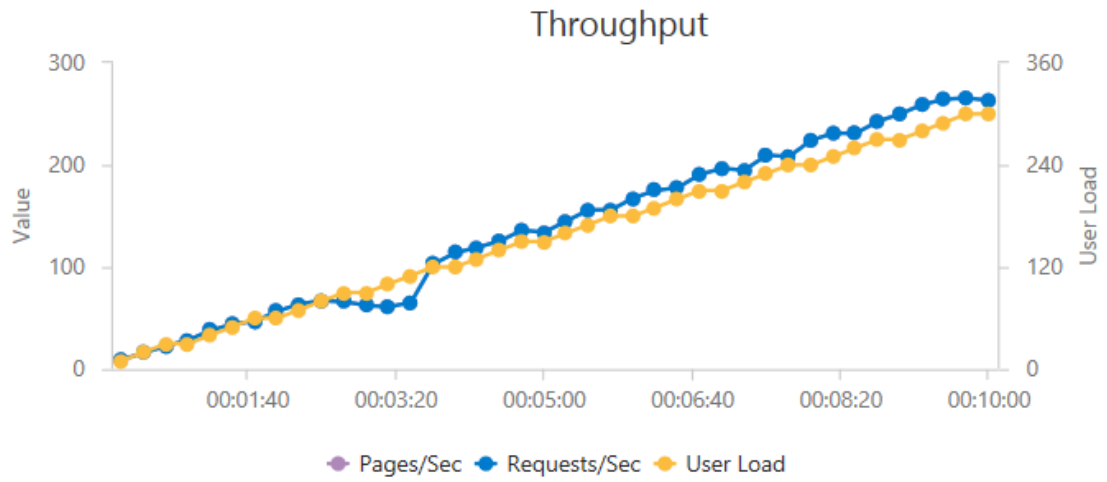
Test settings

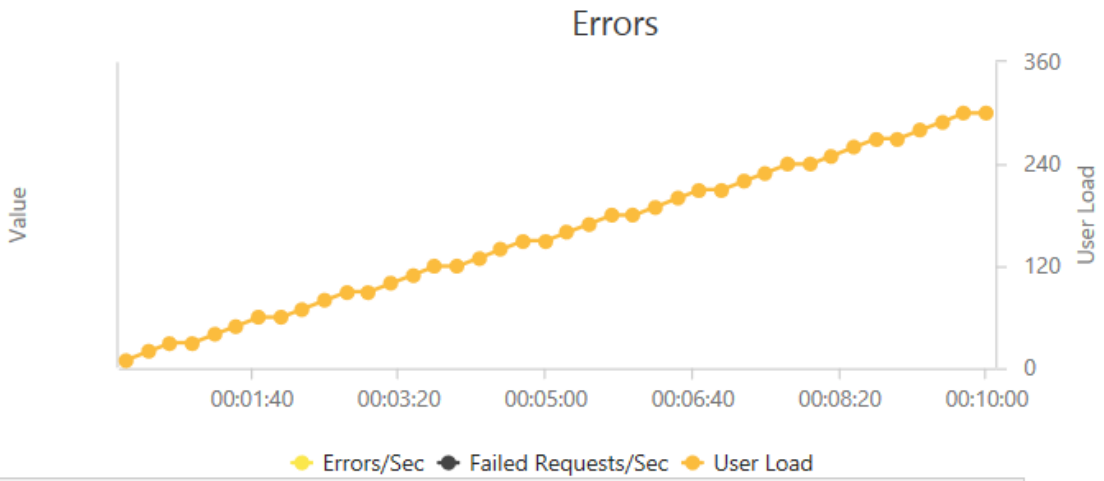
Load duration:	10 min	**Requested by:**	Andrea Tosato	**Run source:**	Team Services portal
Start time:	3/25/2018 9:42:57 PM	**Test:**	GAB2018VR_10Minutes	**Warmup duration:**	-
End time:	3/25/2018 9:52:58 PM	**Location:**	West Europe	**Agent cores:**	1

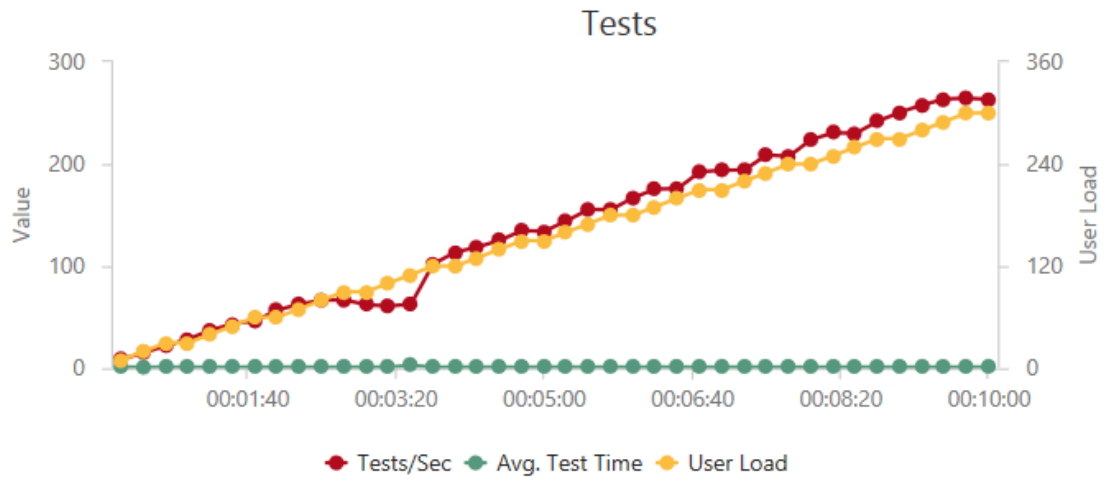
Load Testing 10 minuti

Chart



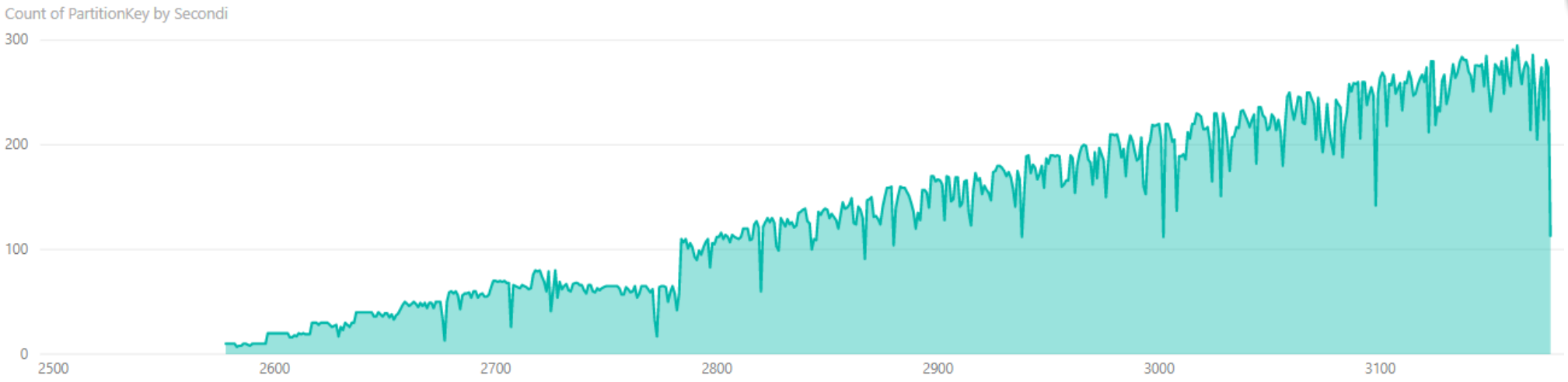


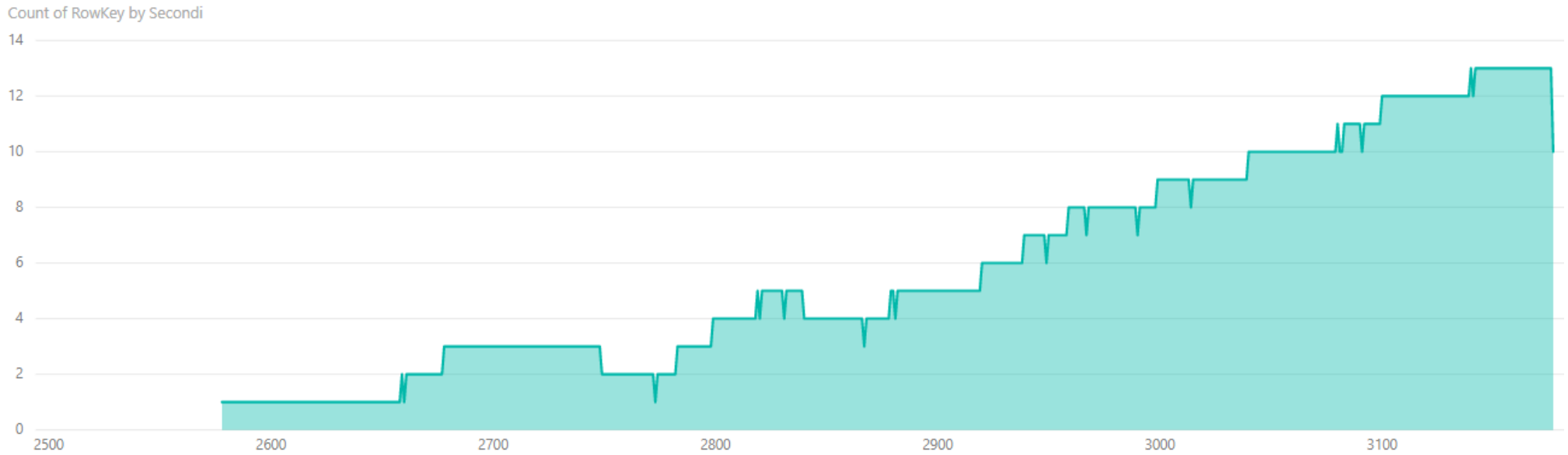




Load Testing 10 minuti

Richieste - Istanze





2018 Global Azure BOOTCAMP

ARGOMENTO

Comparazione con AWS e Google

Comparazione

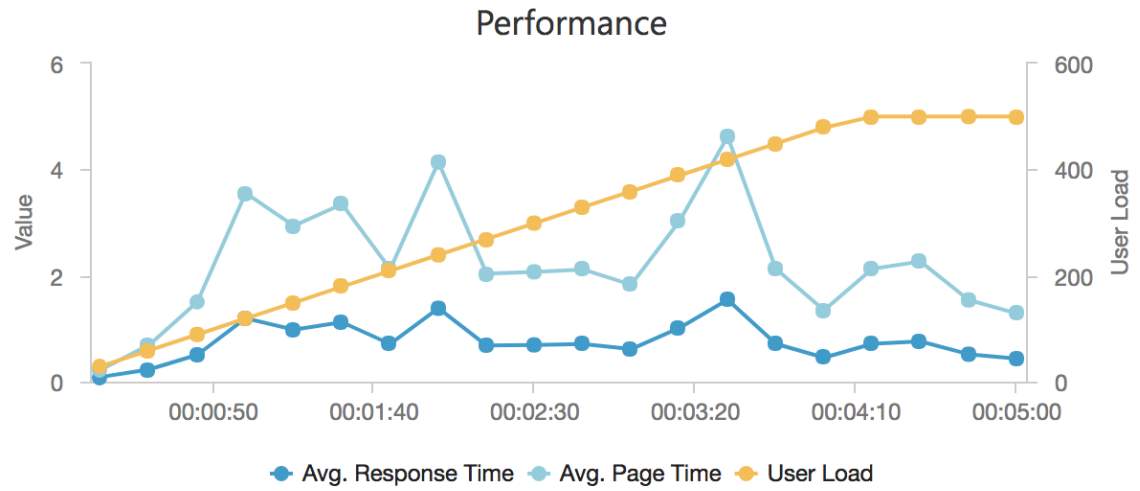
Gennaio 2018

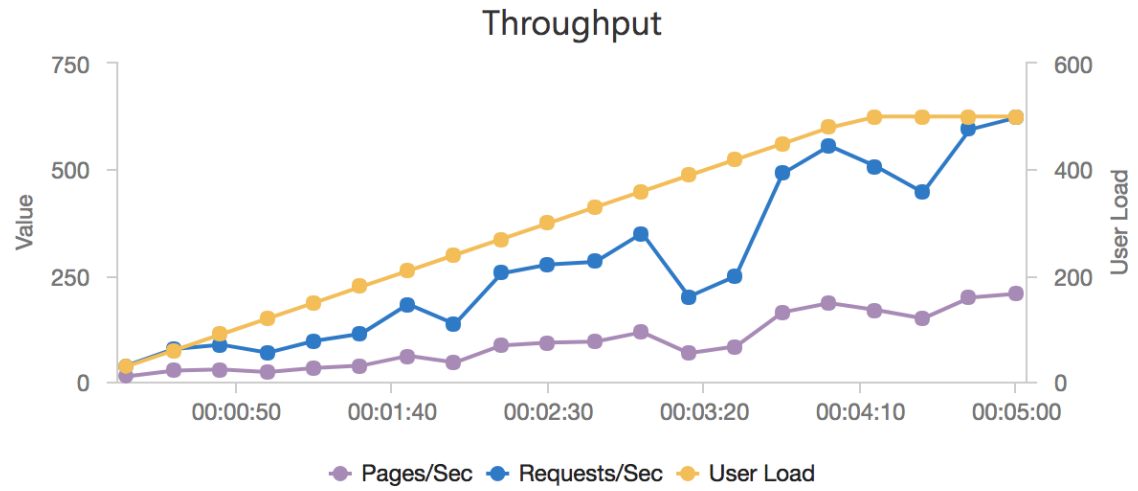
https://www.azurefromthetrenches.com/azure-functions-vs-aws-lambda-scaling-face-off/

	AVG. RESPONSE TIME	USER LOAD	REQUESTS PER SEC	FAILED REQUESTS	ERRORS →	USAGE
Azure	698.7 ms	500 users	281.4 RPS	0 % 0 failed requests 84419 total requests	0 errors 0 thresholds violated	2.5K VUMs
AWS	80.1 ms	500 users	717.6 RPS	0 % 0 failed requests 215271 total requests	2 errors 2 thresholds violated	2.5K VUMs

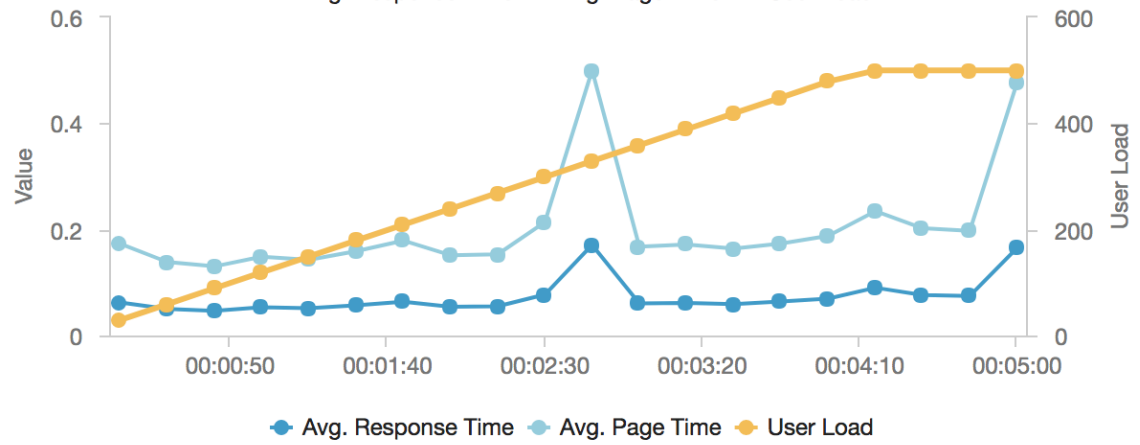
Comparazione

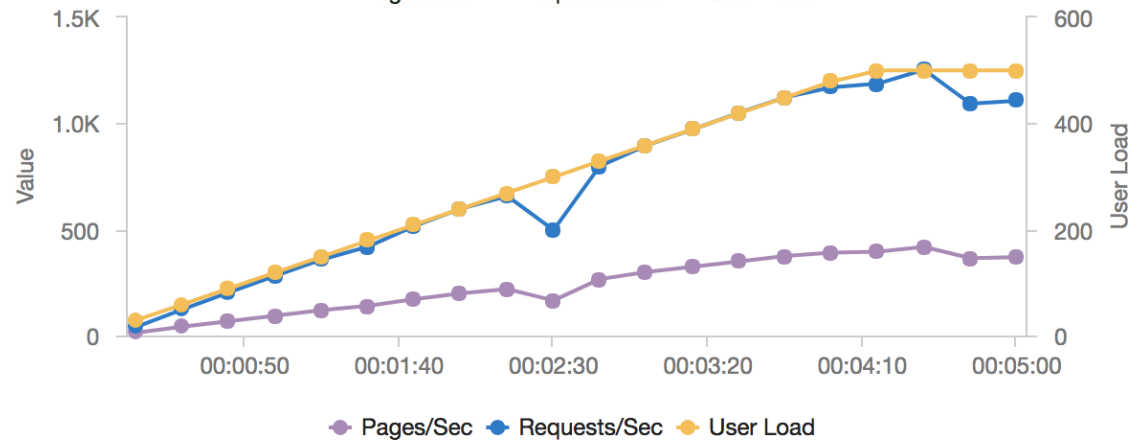
Azure





AWS





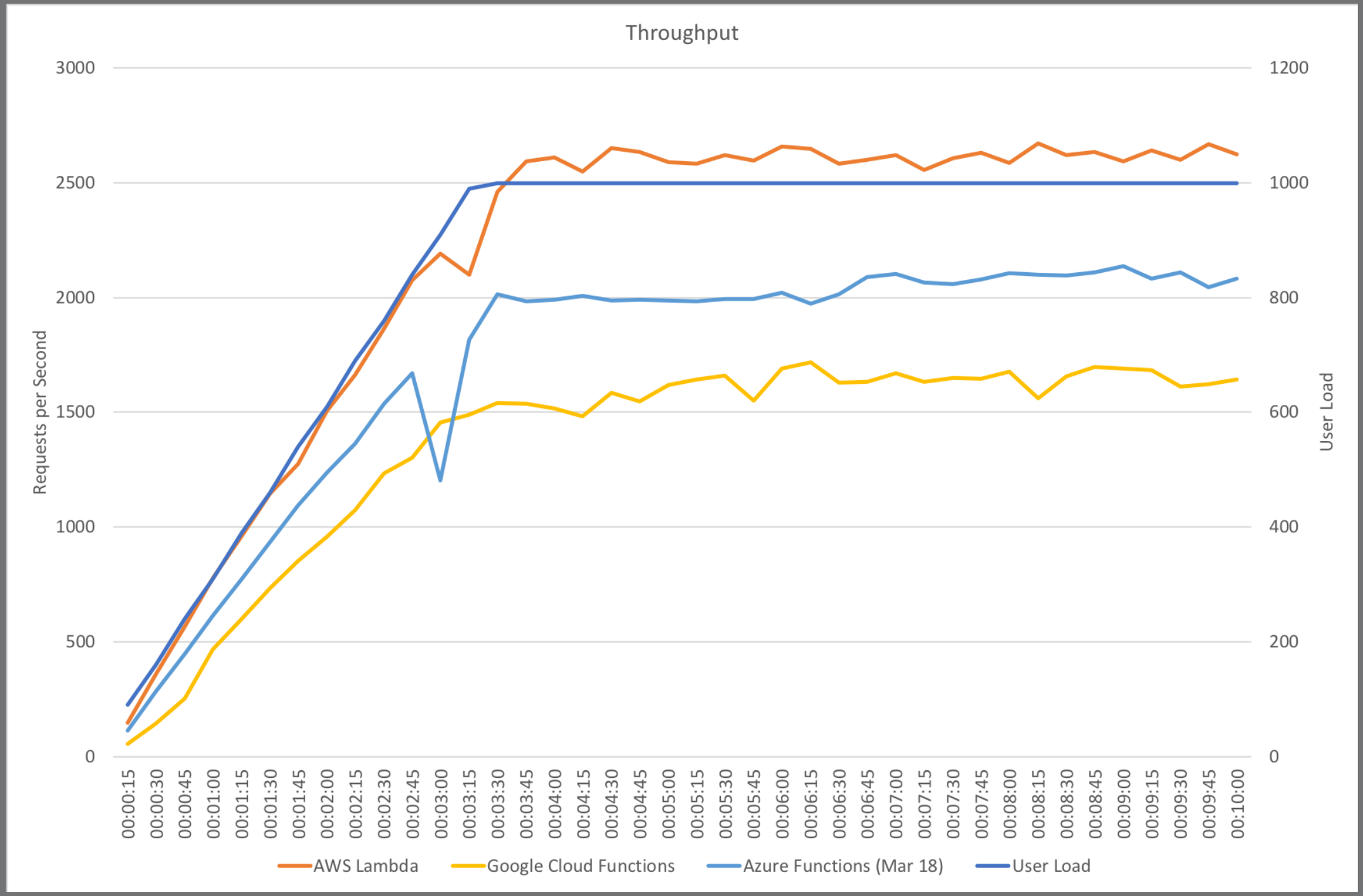
https://www.azurefromthetrenches.com/azure-functions-significant-improvements-in-http-trigger-scaling/

Comparazione

Marzo 2018

https://www.azurefromthetrenches.com/azure-functions-significant-improvements-in-http-trigger-scaling/

	AVG. RESPONSE TIME	USER LOAD	REQUESTS PER SEC	FAILED REQUESTS	ERRORS →	USAGE
AWS	66.4 ms	1K users	2.2K RPS	0 % 102 failed requests 1306051 total requests	4 errors 2 thresholds violated	10K VUMs
Azure	201.3 ms	1K users	1.7K RPS	0 % 17 failed requests 1024595 total requests	3 errors 2 thresholds violated	10K VUMs
Google	141.8 ms	1K users	1.4K RPS	0 % 46 failed requests 816235 total requests	2 errors 0 thresholds violated	10K VUMs


Throughput
Requests per Second
User Load
3000
2500
2000
1500
1000
500
0
1200
1000
800
600
400
200
0
00:00:15
00:00:30
00:00:45
00:01:00
00:01:15
00:01:30
00:01:45
00:02:00
00:02:15
00:02:30
00:02:45
00:03:00
00:03:15
00:03:30
00:03:45
00:04:00
00:04:15
00:04:30
00:04:45
00:05:00
00:05:15
00:05:30
00:05:45
00:06:00
00:06:15
00:06:30
00:06:45
00:07:00
00:07:15
00:07:30
00:07:45
00:08:00
00:08:15
00:08:30
00:08:45
00:09:00
00:09:15
00:09:30
00:09:45
00:10:00
AWS Lambda
Google Cloud Functions
Azure Functions (Mar 18)
User Load

2018 Global Azure BOOTCAMP

http://functionlibrary.azurewebsites.net/

ARGOMENTO

Progetti esempio

2018 Global Azure BOOTCAMP

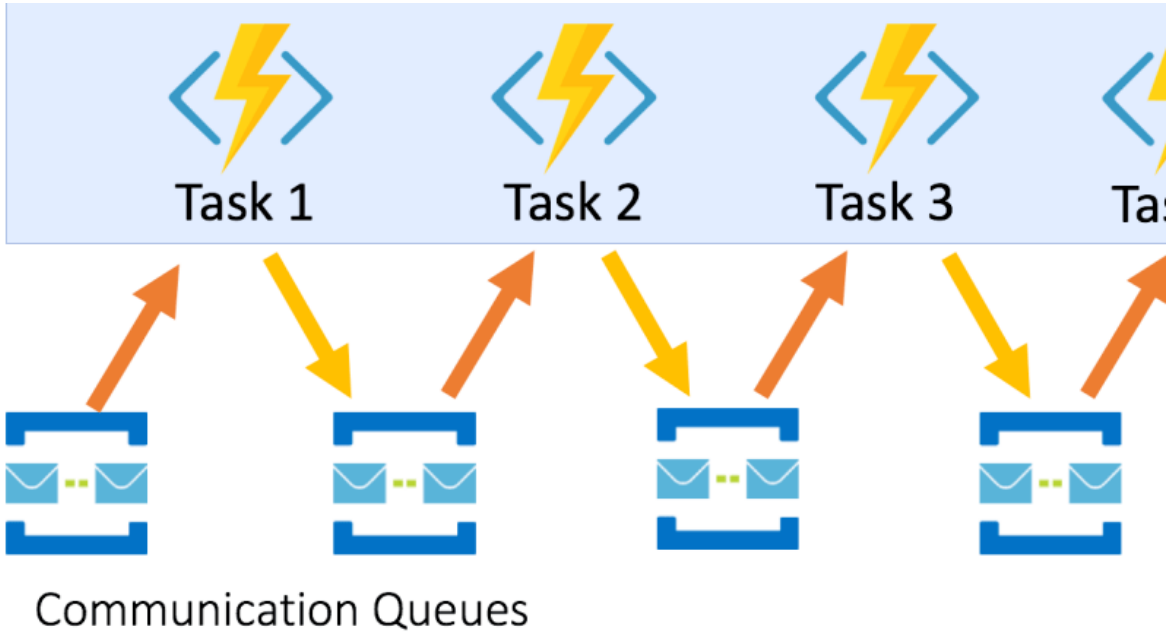
ARGOMENTO

Durable Functions

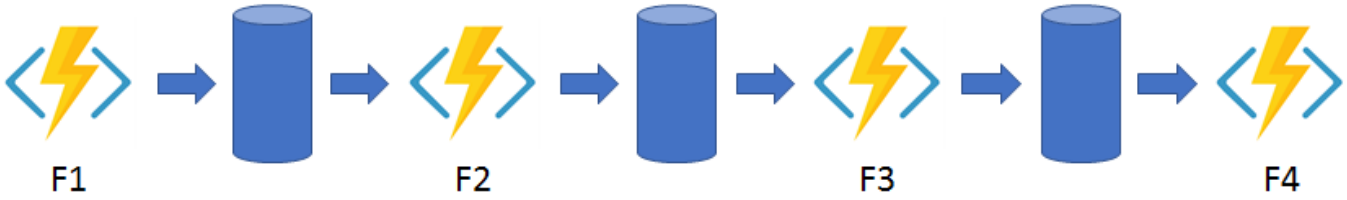
Come ottimizzare una applicazione per sfruttare al meglio il piano di servizi a consumo

Come cambia la gestione di un flusso

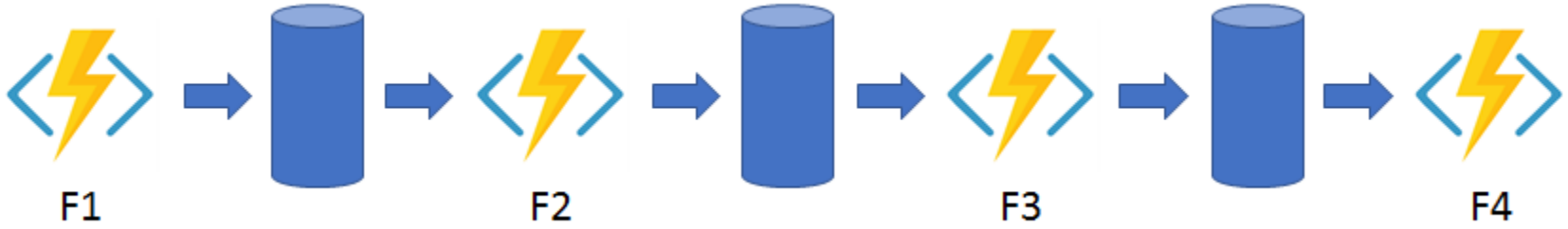
Con durable functions possiamo gestire la rientranza sulla funzione chiamante.



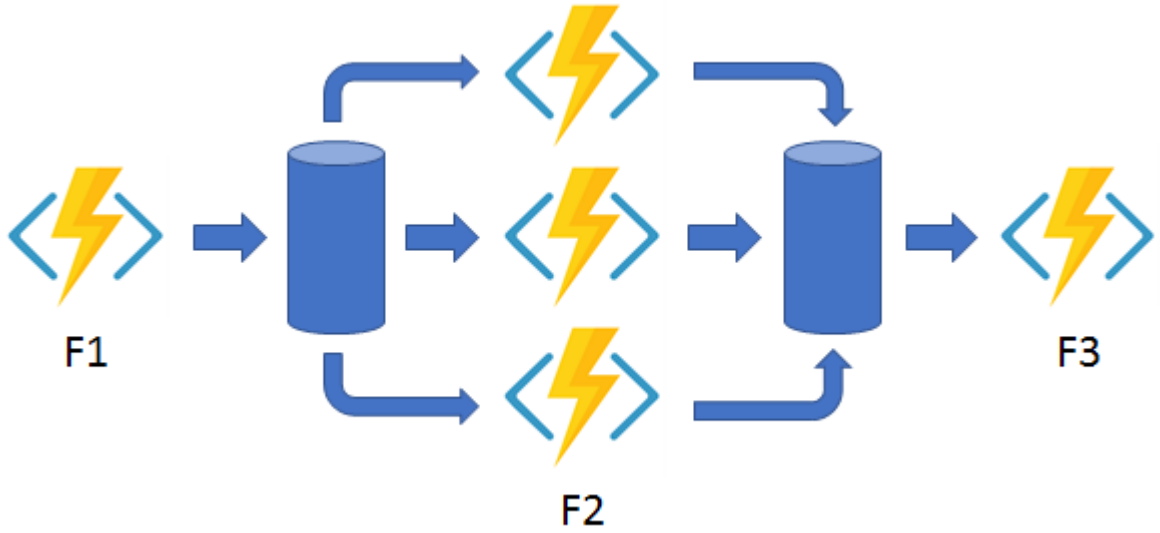
Passiamo da una gestione manuale a una gestione automatizzata



Function chaining

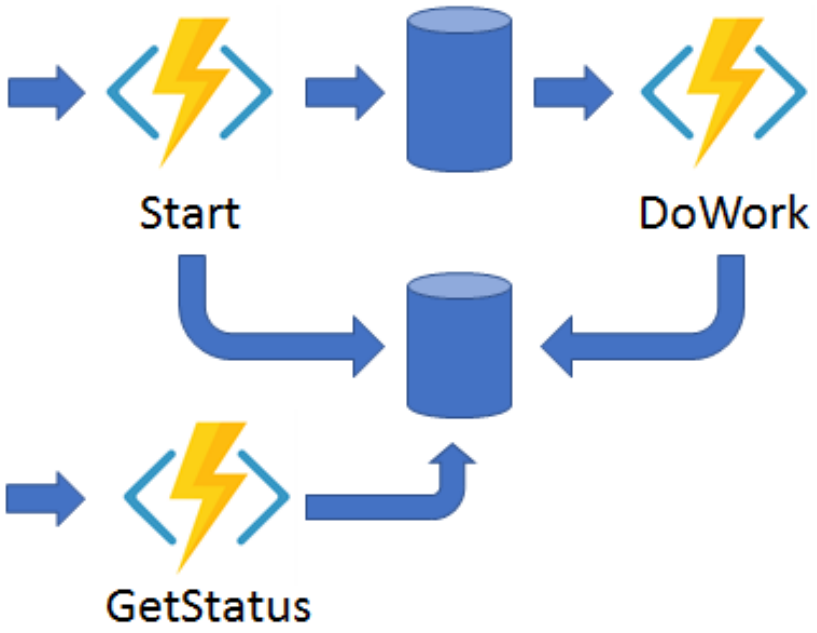


Fan-out/fan-in

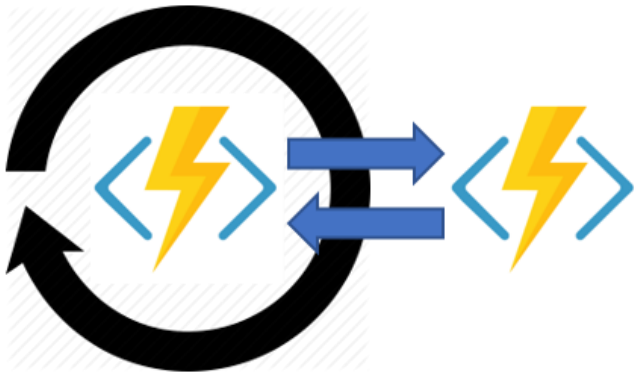


Durable Functions

Async HTTP APIs



Fan-out/fan-in

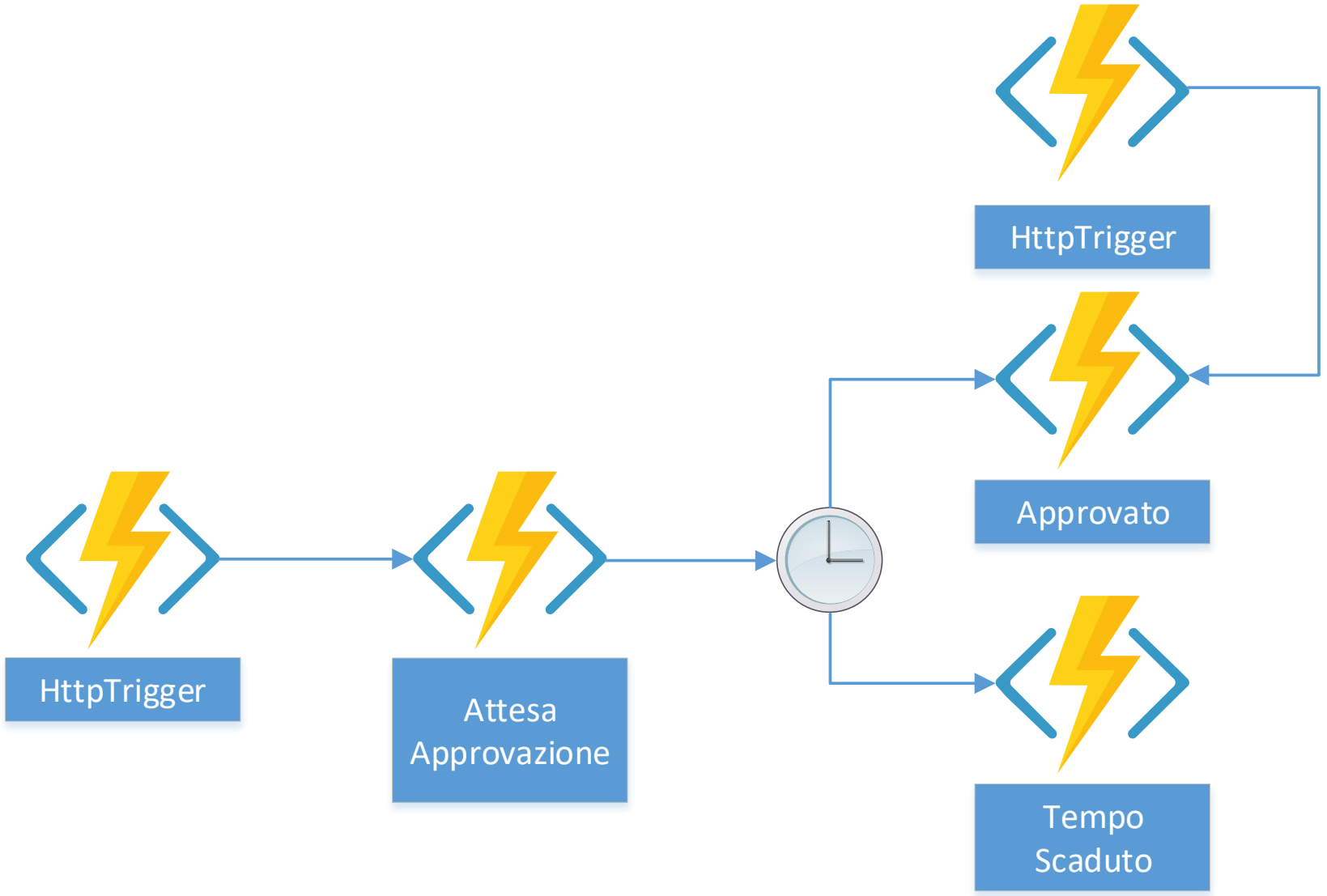


Human interaction

Demo

Human Interaction

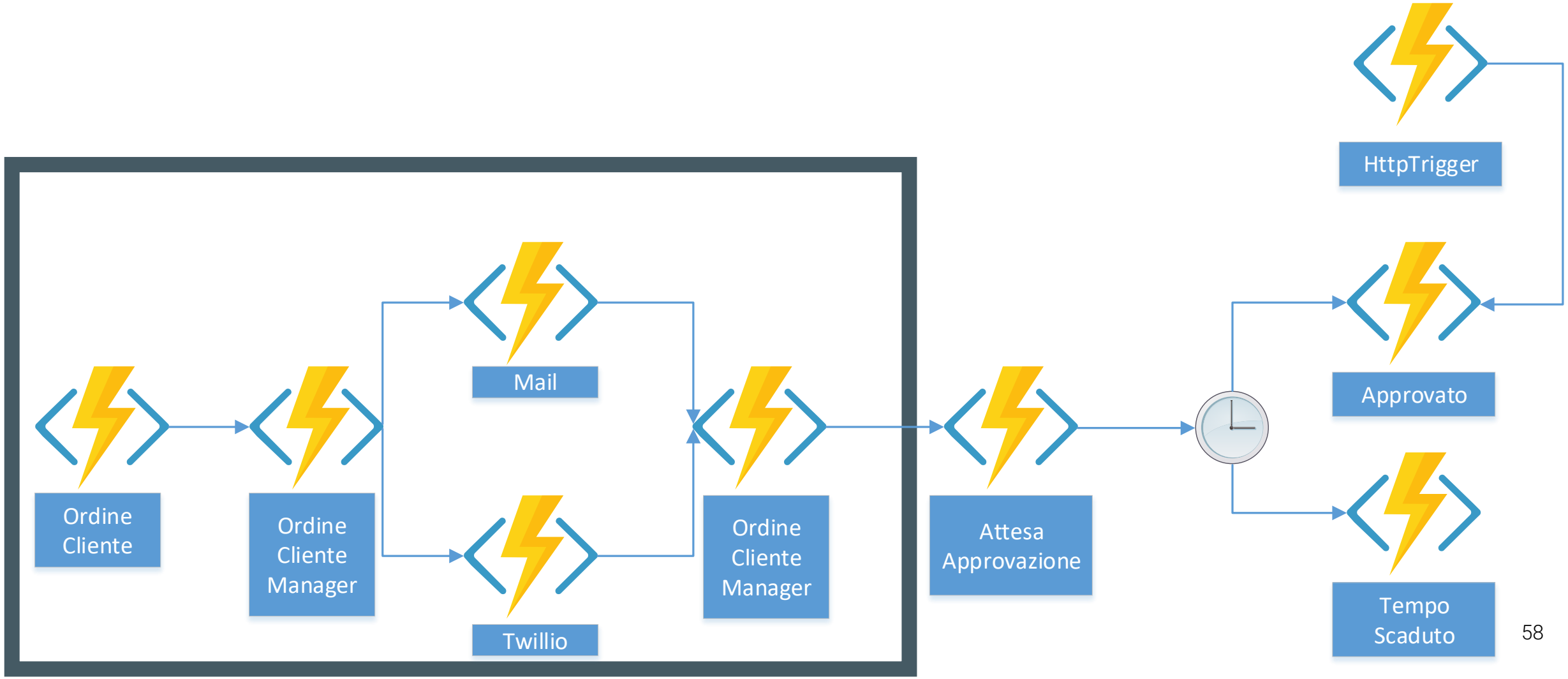
Durable Functions



Demo

Complete Durable Functions

Durable Functions



2018 Global Azure BOOTCAMP

ARGOMENTO

Azure Functions Runtime

Demo

Azure Functions Runtime

Azure functions runtime - Docker

https://hub.docker.com/r/microsoft/azure-functions-runtime/

Grazie

Domande?

 Andreatosato

 @ATosato86

 Andreatosato