Microsoft

Making Sense Out of Serverless Computing

Focus is not on the servers, nor on the lack of them.

Bruno Borges – @brunoborges

Microsoft Azure

Developer Relations Group

Microsoft

Bruno Borges

Principal [Cloud Developer] Advocate

Since 2001 – Developer, Speaker, Blogger

Previously: Product Management & Developer Relations at Oracle

medium.com/@brunoborges

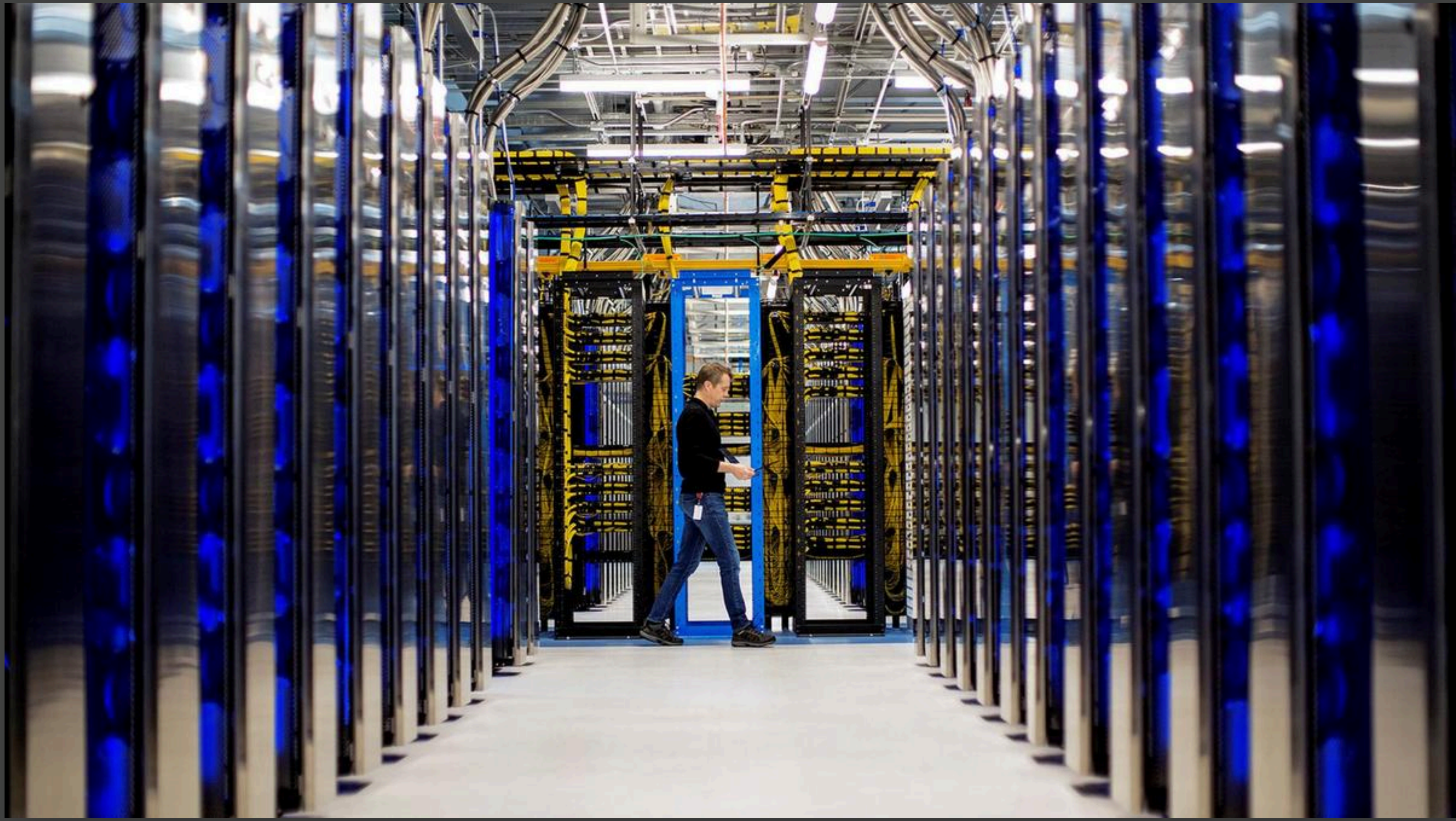
@brunoborges

bruno.borges@microsoft.com

“Serverless”

Idiotic term

but nevertheless cool :-)



“Serverless is a consequence of a focus on business value.”

Ben Kehoe

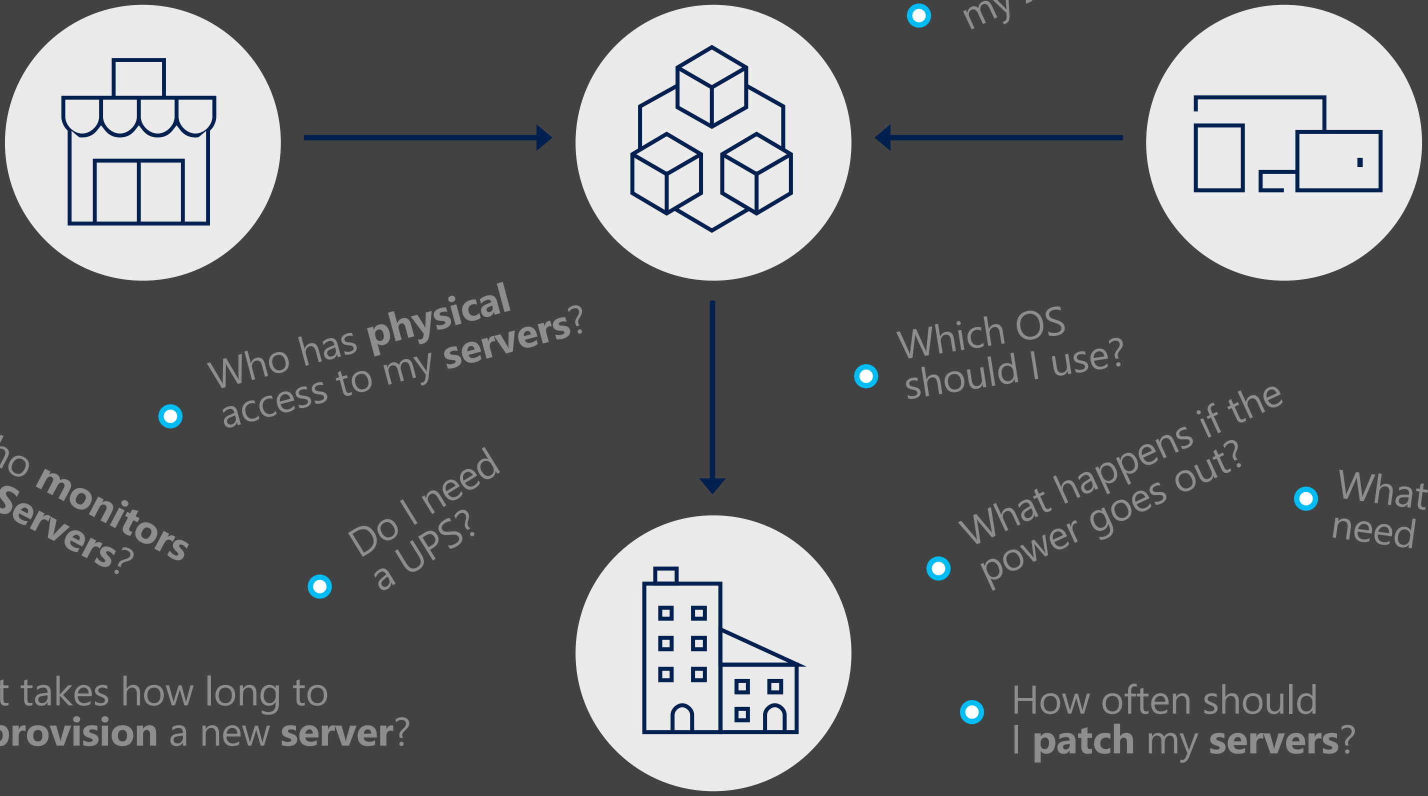
Serverless is a State of Mind

aka.ms/serverless-state-of-mind

Before cloud

Copyright Computer Laboratory, University of Cambridge. Reproduced by permission.

- What media should I use to keep backup?
- What is the right **size** of **servers** for my business needs?
- How do I **deploy** new **code** to my **server**?
- What happens in case of **server hardware** failure?
- Which packages should be on my **server**?
- What size of **servers** should I **buy**?
- Who **monitors** my **App**?
- How often should I backup my **server**?
- How can I **scale** my app?



- How can I increase **server** utilization?
- Do I need secondary network connection?
- Who has **physical** access to my **servers**?
- Which OS should I use?
- Are my **server** in a secure location?
- Who **monitors** my **Servers**?
- Do I need a UPS?
- What happens if the power goes out?
- What storage I need to use?
- How many **servers** do I need?
- It takes how long to **provision** a new **server**?
- How often should I **patch** my **servers**?
- How can I dynamically configure my app?

On-Premises

The “evolution” of application platforms

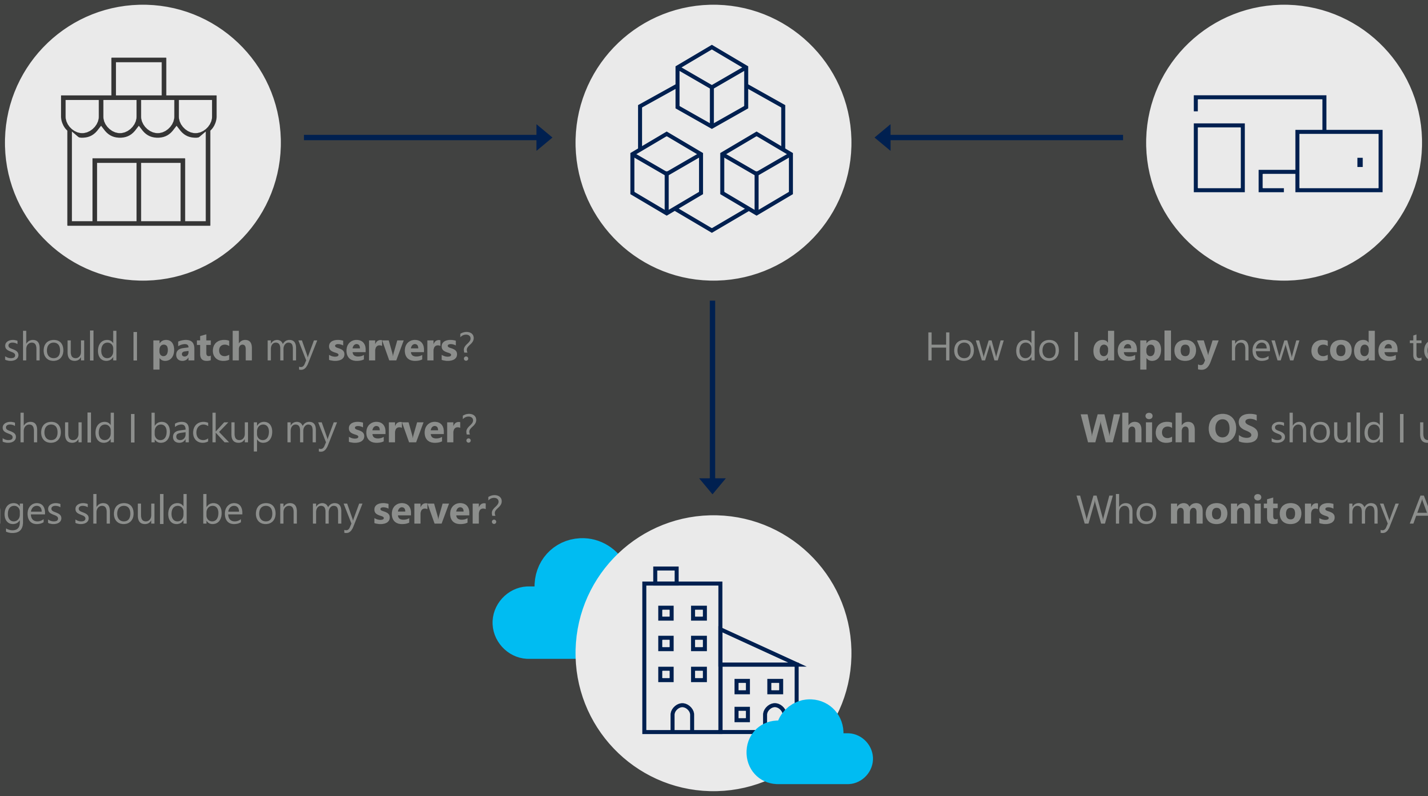
@brunoborges Microsoft

What is the right **size** of **servers** for my business needs?

How can I increase **server** utilization?

How many **servers** do I need?

How can I **scale** my app?



How often should I **patch** my **servers**?

How often should I backup my **server**?

Which packages should be on my **server**?

How do I **deploy** new **code** to my **server**?

Which OS should I use?

Who **monitors** my App?

On-Premises
IaaS

The “evolution” of application platforms

What is the right **size** of **“servers”** for my business needs?

How can I increase **“server”** utilization?

How many **“servers”** do I need?

How can I **scale** my app?

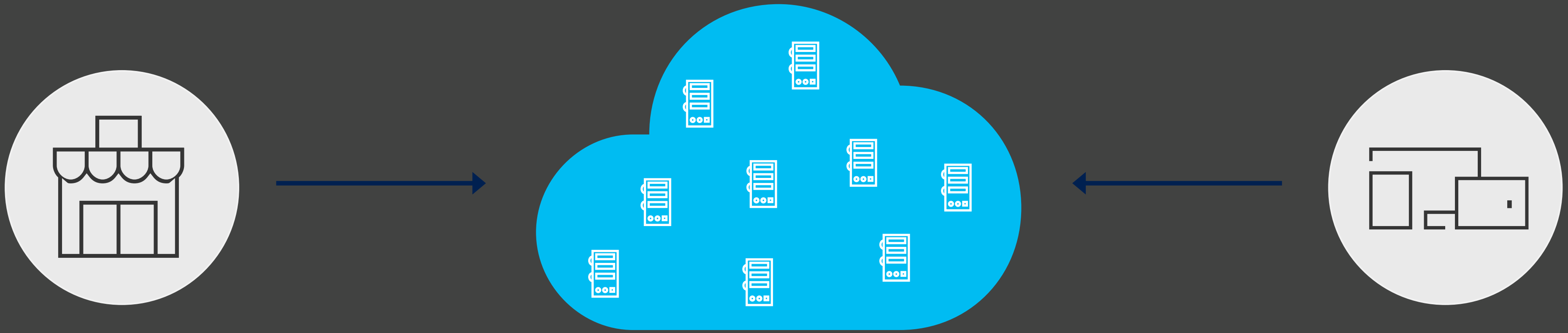
On-Premises

IaaS

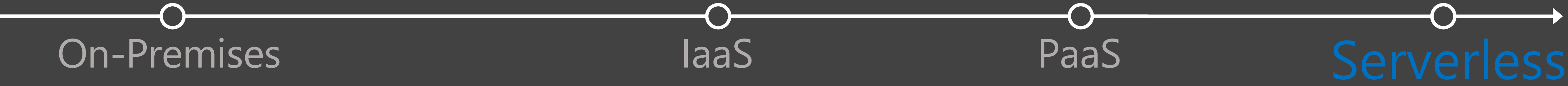
PaaS

The “evolution” of application platforms

How do I **architect** my app?



Serverless, the platform for next gen apps



The “evolution” of application platforms

@brunoborges Microsoft

“Serverless”

Event-driven

Micro-billing per usage

No server setup or maintenance

Scalability and high availability

Cloud Computing

The basics.

- Applications
  - Monoliths, Microservices
- Runtimes
  - Application Servers, Stacks, Engines, JVMs
- Containers
  - Docker, Kubernetes, OpenShift, Cloud Foundry
- Virtual Machines
  - Windows, Linux, Puppet, Chef, Ansible

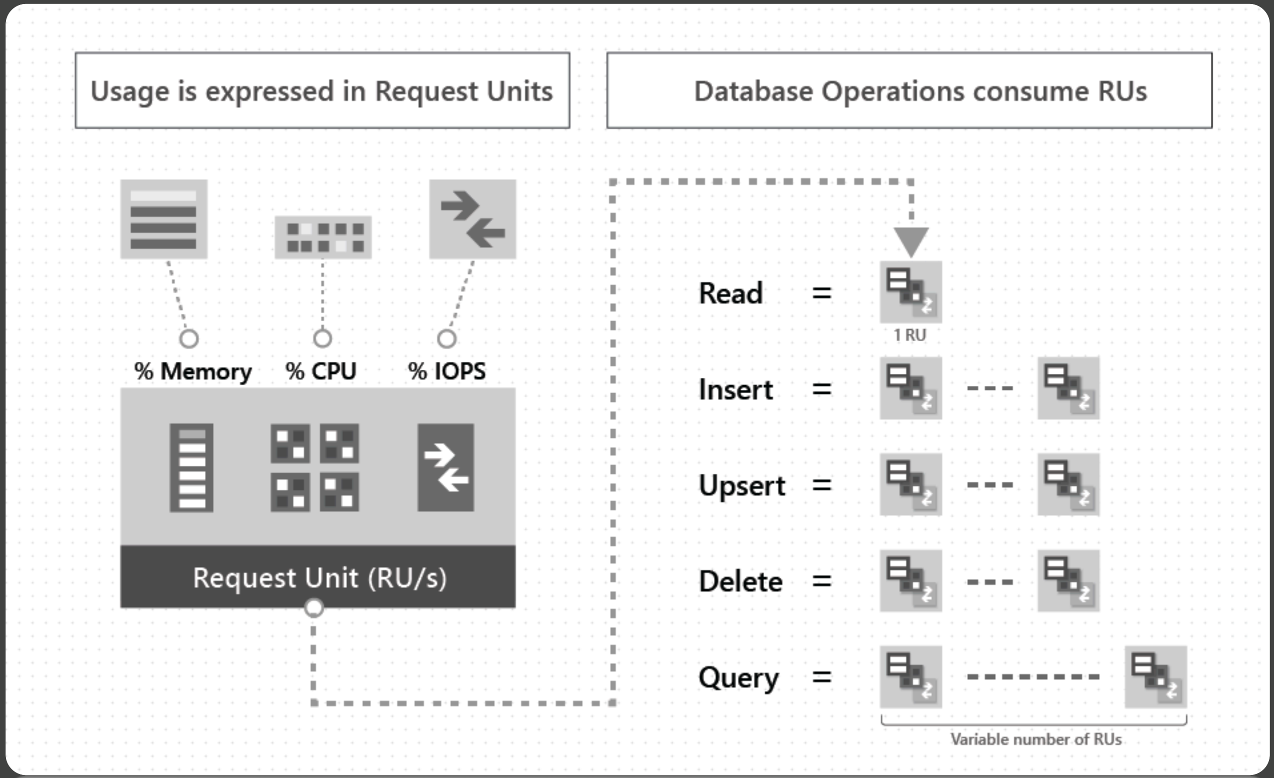
Serverless Cloud Computing

The basics. But it can go beyond these.

- Functions
  - Cloud services: Azure Functions, Google Cloud Functions, AWS Lambda
  - Projects: Azure Functions, OpenFaaS, Apache OpenWhisk, Fn Project, Knative
- Events
  - Azure Event Hubs / Event Grids, Amazon Kinesis, Google Cloud Pub/Sub
- Flows
  - IFTTT-style: Azure Logic Apps, Amazon SWF
- Databases
  - Azure Cosmos DB, AWS Aurora Serverless

Serverless Databases

Azure Cosmos DB
NoSQL
Auto-scale
Highly Available
Pay per usage



Usage is expressed in Request Units

Database Operations consume RUs

% Memory

% CPU

% IOPS

Request Unit (RU/s)

Read =

1 RU

Insert = ---

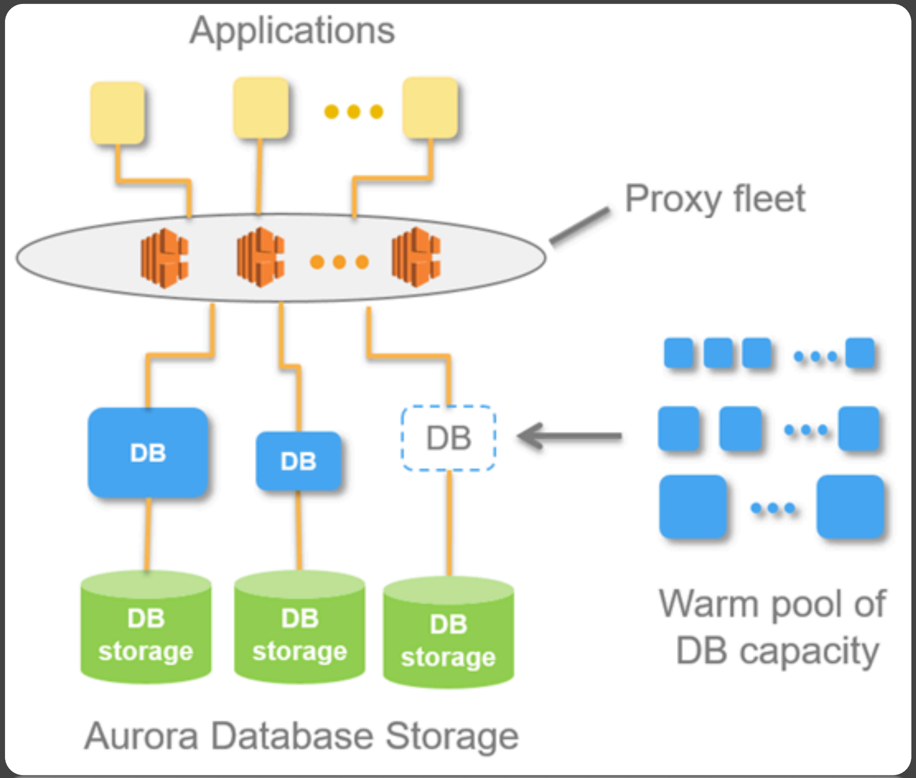
Upsert = ---

Delete = ---

Query = ---------

Variable number of RUs

AWS Aurora Serverless
SQL
Auto-scale
Highly Available
Pay per usage



Applications

Proxy fleet

DB

DB

DB

DB storage

DB storage

DB storage

Warm pool of DB capacity

Aurora Database Storage

Serverless Flows

Workflow in the cloud

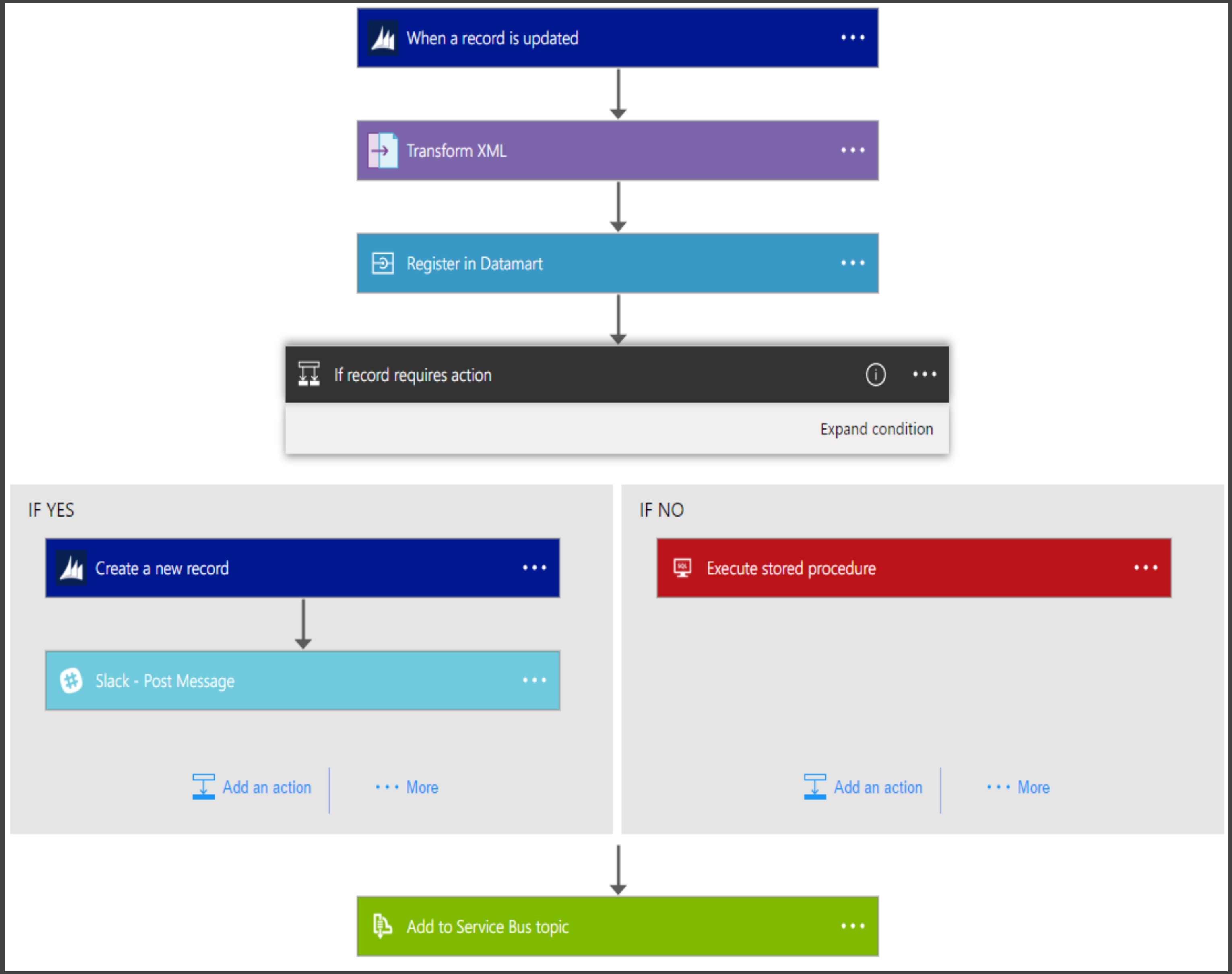
Powerful control flow

Connect disparate applications

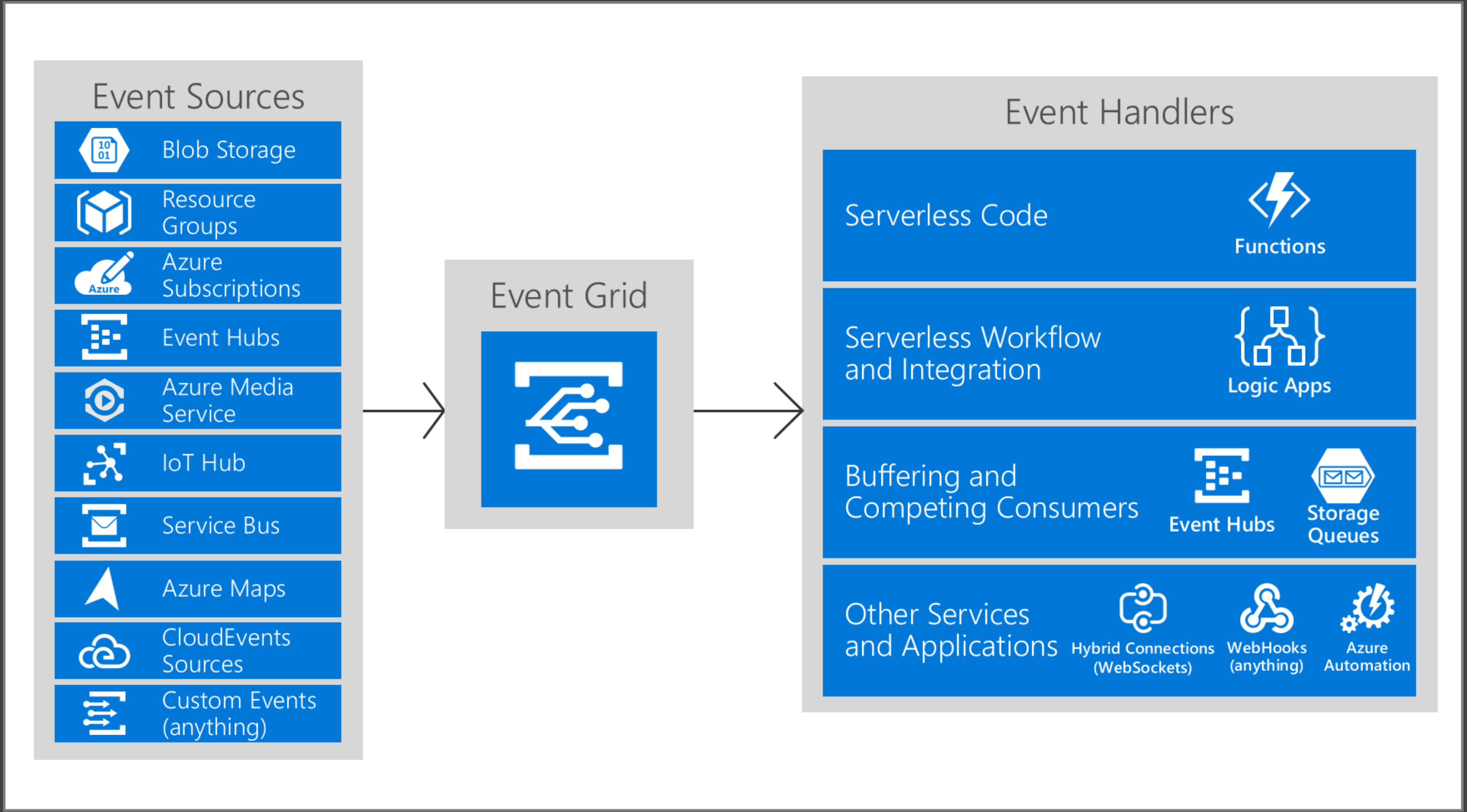
No code designer for rapid creation

Charges per actions and connectors

Extra charge per hour for environment isolation



Serverless Events



100,000 operations **free**

$0,60 per **million** operations

* *Azure Event Grid pricing*

Serverless Functions

Open Source Projects (Runtimes)

KNative

Kubeless

Fn Project

OpenWhisk

OpenFaaS

Azure Functions

Cloud Providers

AWS Lambda

- Proprietary
- AWS only
- Open Source Emulator

Azure Functions

- Open Source
- Runs anywhere

Google Cloud Functions

- Proprietary
- GCP only
- Open Source emulator

Java, JVM, and Serverless

- Cold startup and footprint are an issue
  - **Solutions**
    - Class Data Sharing (JEPs 310, 341, 350)
    - Ahead-of-Time (AOT) Compilation
- Garbage Collection ~~is~~ *may be* unnecessary
  - **Solution**
    - Remove the GC!
    - JEP 318 – Epsilon: A No-Op Garbage Collector (Java 11)
    - Once the available Java heap is exhausted, the JVM will shut down.

GraalVM™

OpenJDK™

OpenJ9

Your First Serverless Function

GIF Loop via NikeTalk.com

Pay as you grow?

Economic Advantage

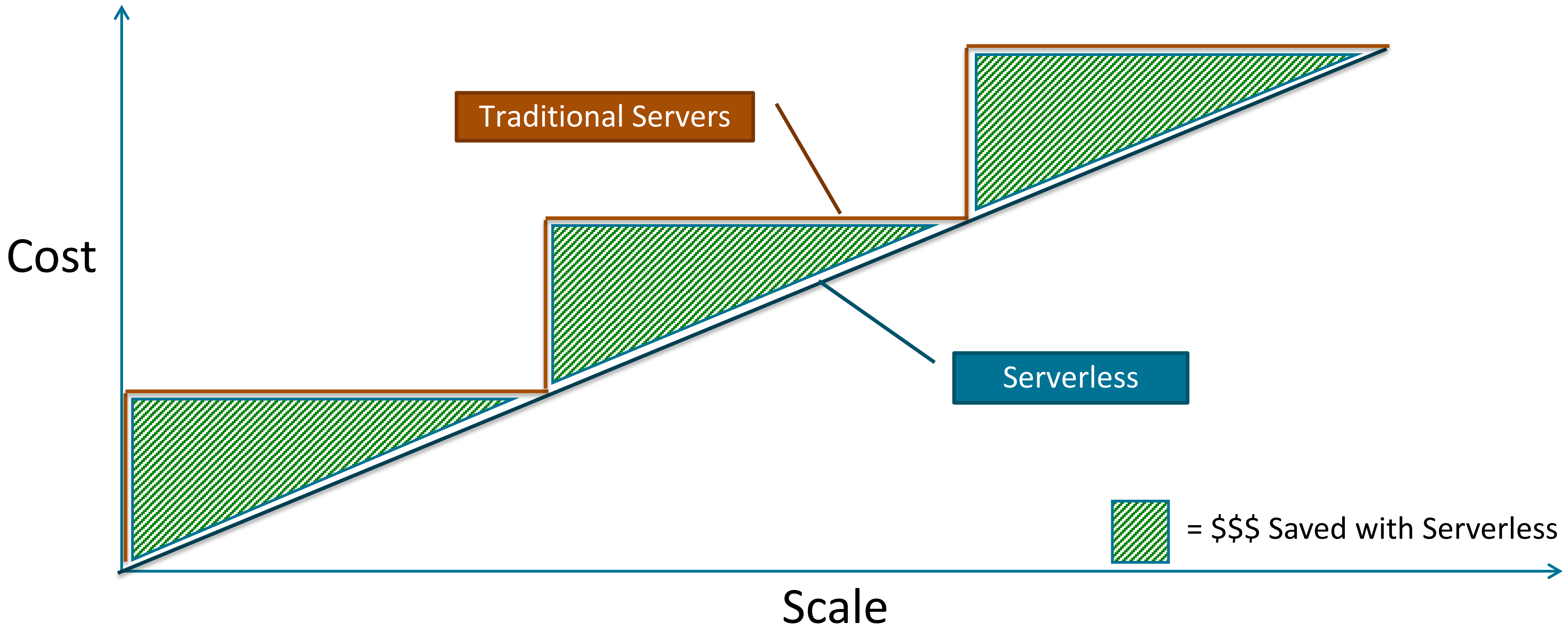
Containers/VMs versus Serverless Functions

- On-premises is upfront cost.
- Cloud Containers/VMs are charged per allocation.
- Containers/VMs with 10% CPU consumption still cost the same when at 100% CPU consumption.

- Serverless is cost per computing usage.

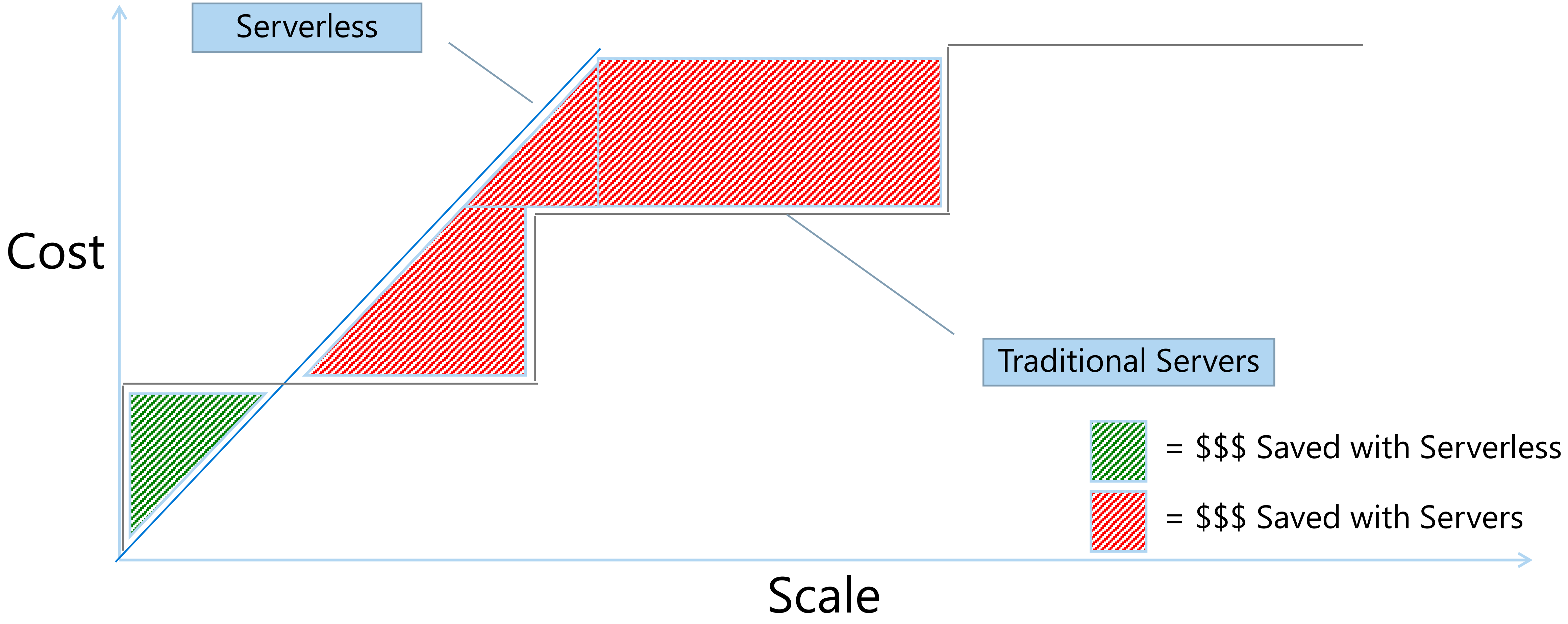
What to expect?

@steveonjava



The reality…

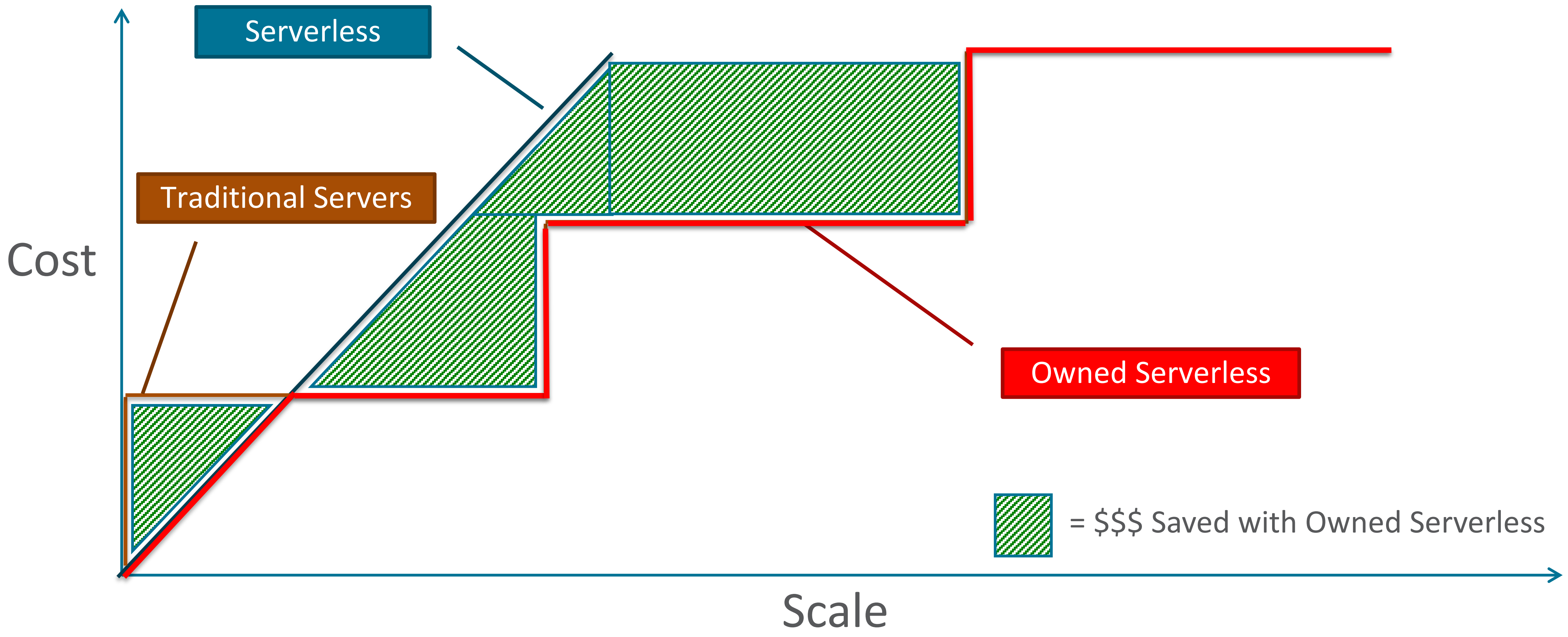
@steveonjava



Predictability

Your own serverless platform: deliver the “developer experience”

@steveonjava



Pay as you grow.
Until you own?

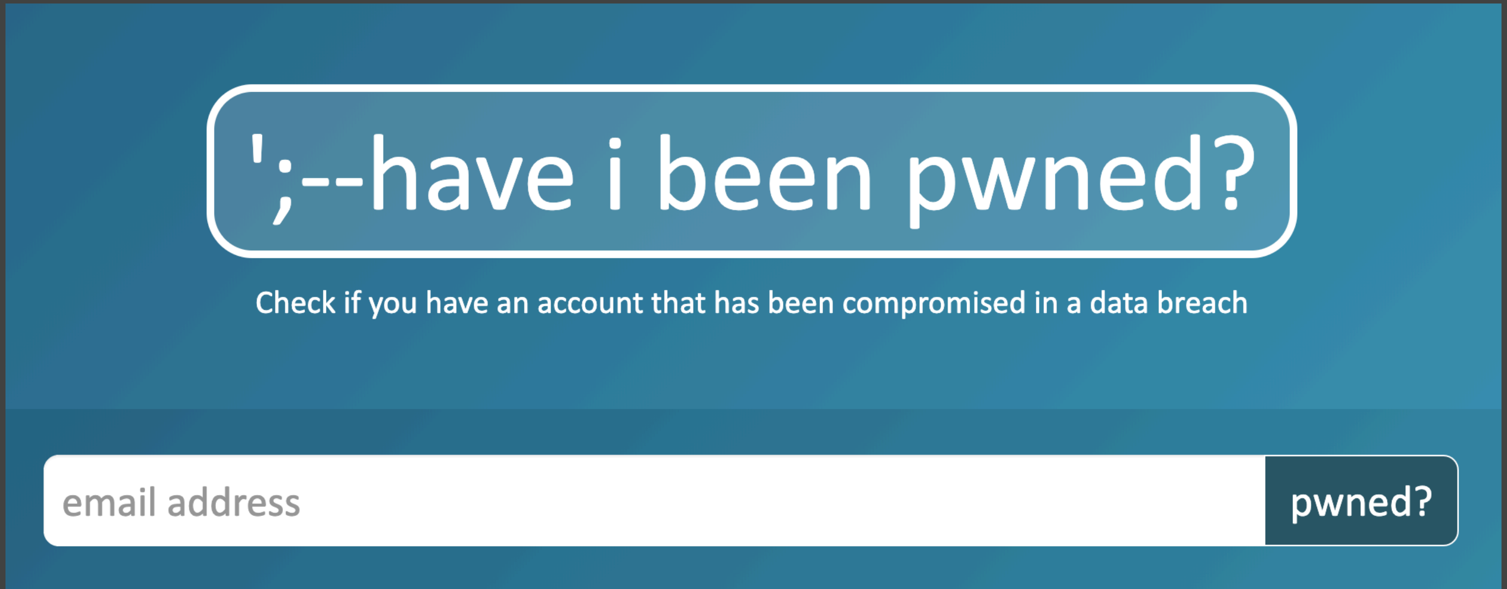
© AKM IMAGES / Splash News

$$$ you'd save by owning, goes into staffing.

What site is this?

haveibeenpwned.com

- 32 million API requests in a week
- 5,5 million API requests in a day
- 477 GB bandwidth in a week



';--have i been pwned?

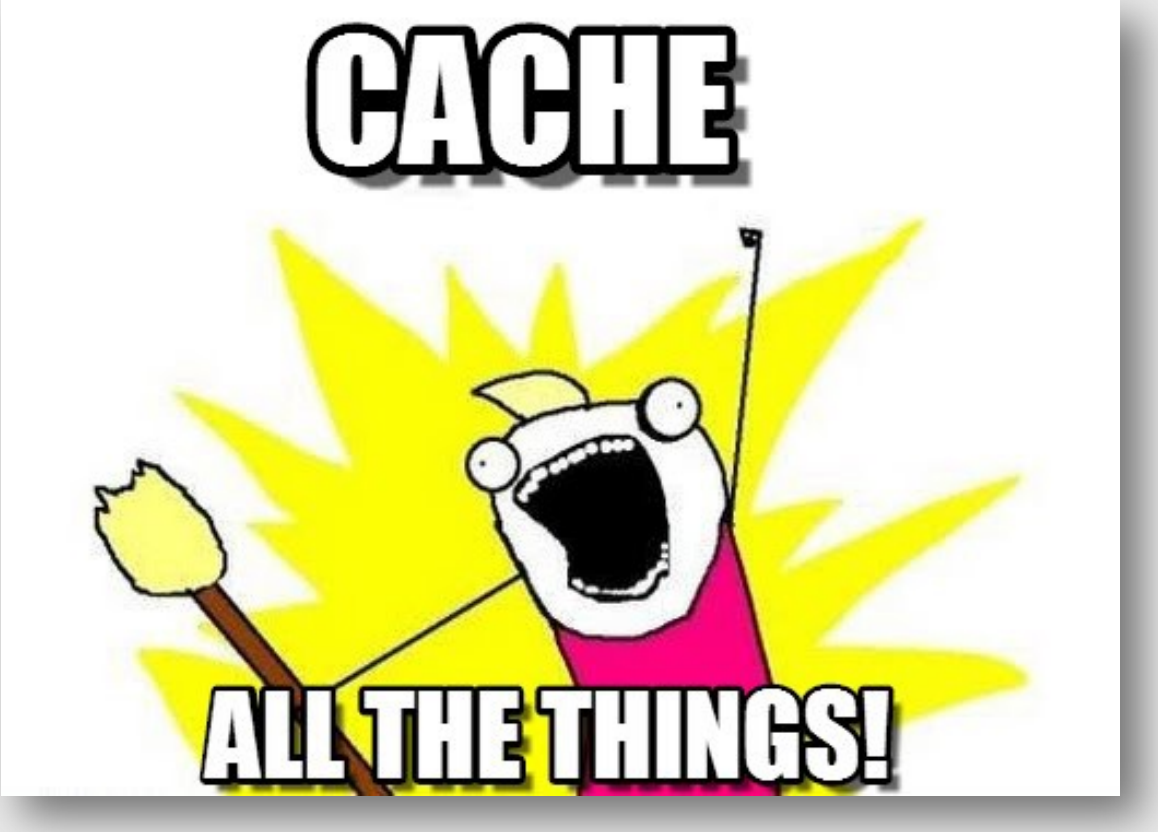
Check if you have an account that has been compromised in a data breach

';--have i been pwned?

- 32 million API requests in a week. 477 GB bandwidth in a week
- Azure Pricing
  - Azure Functions: first 1 million requests are free. $0,20 per million requests afterwards.
  - Azure Bandwidth: first 5GB / month are free. $ 0.087 per GB within 5GB-10TB.
  - **Total estimate:** $2.697,60 / year

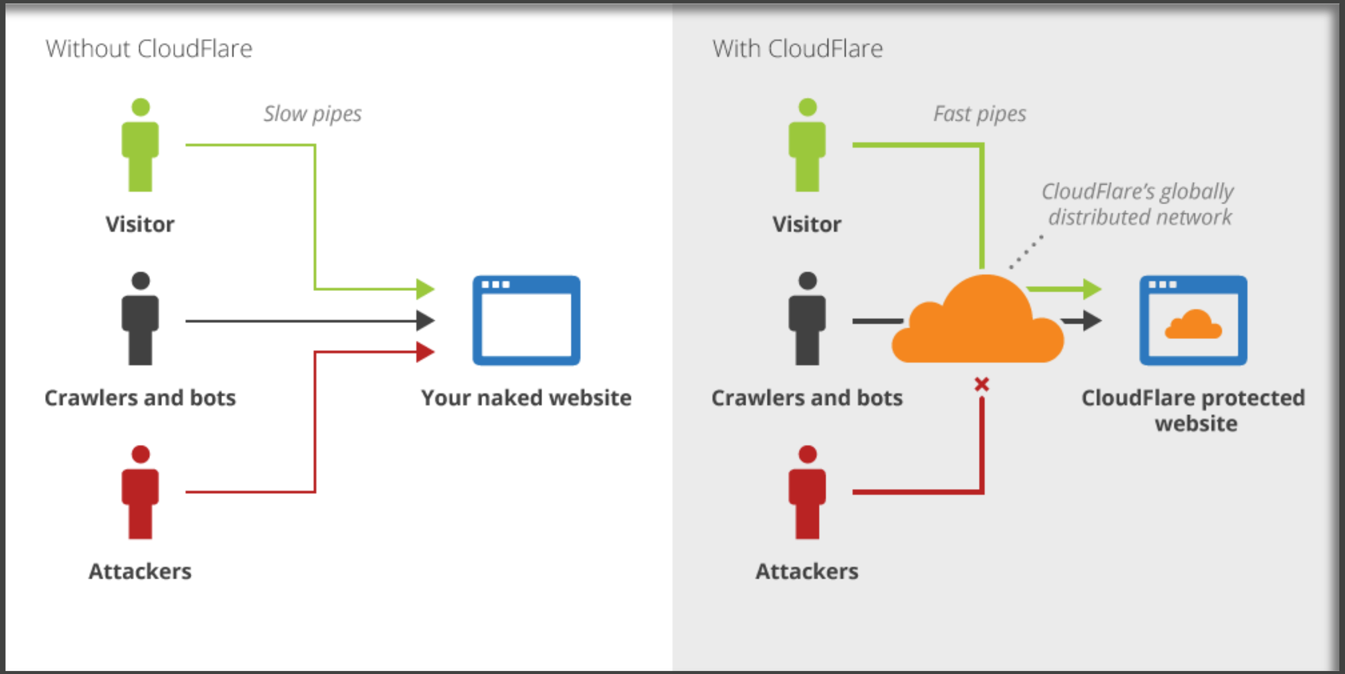
How to Reduce Cost in Serverless Web Apps

Be smart. Stay focused on business value.



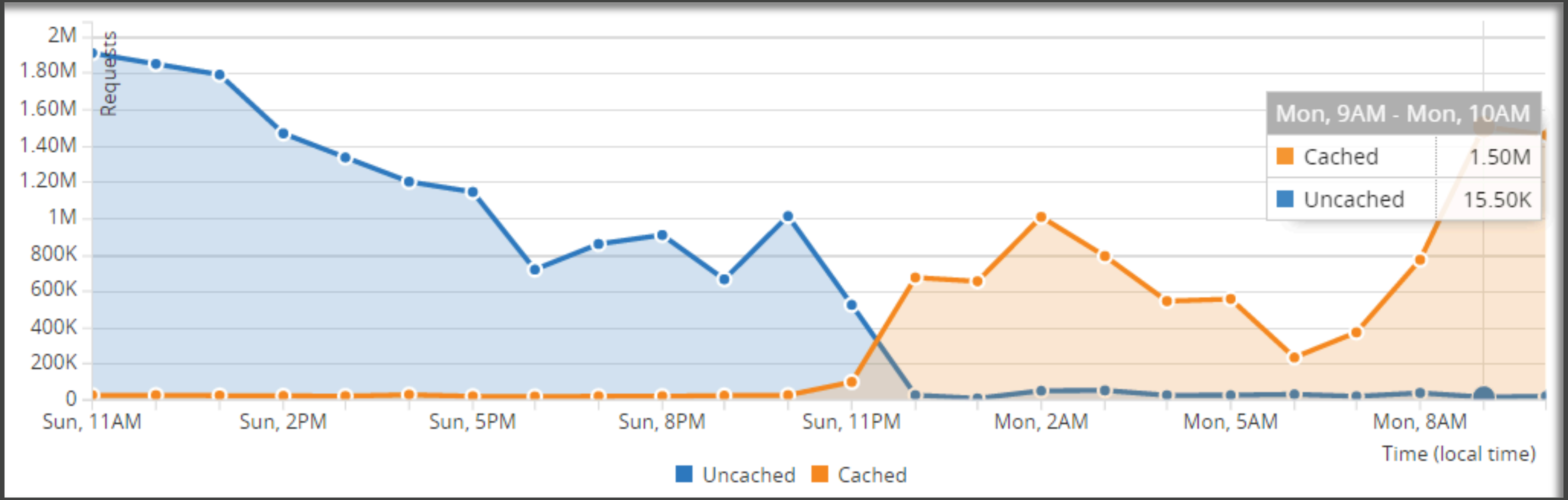
Cloudflare to the rescue

- Prevent requests from hitting your serverless functions
- Block attackers IPs
- Distribute cached data globally (CDNs)



Cloudflare to the rescue

- Only 1% of requests a day ended up hitting Azure Functions
- In the end: ~$ 0,80 /month
  - compared to original estimate of ~ $ 7,50 /day



“The fastest, most cost-effective way of running code on Azure is to avoid hitting Azure!.”

Troy Hunt

How many people work at haveibeenpwned.com?

One

Learn more

aka.ms/troy-hunt-case-cloudflare

aka.ms/troy-hunt-case-azure-functions

@troyhunt

“Craftsman can be 5x even more efficient by knowing what not to do.”

Anton Keks

Focus on business value. Write code for that goal. Save money.

Serverless Cloud Computing’s TLDR

Thank you

Microsoft