Cloud-Native Architecture

For .NET Developers

Jonathan "J." Tower

Overview

Assumptions

Background

Patterns

Cloud-Native Technologies

Hi, I'm J.

Jonathan "J." Tower

Partner & Principal Consultant

Trailhead Technology Partners

- Microsoft MVP in ASP.NET
- Business Owner
- Organizer of Beer City Code

- jtower@trailheadtechnology.com
- trailheadtechnology.com/blog
- jtowermi

github.com/jonathantower/cloud-native-dotnet

If You Give $200, So Will I!

bit.ly/chicao-cloud

"charity:water is a non-profit organization that provides clean and safe drinking water to people in developing nations. The organization was founded in 2006 and has helped fund 22,936 projects in 24 countries, benefiting over 4.6 million people." - Wikipedia

"4/4 Stars"
- CharityNavigator.org

charity: water

Assumptions

Experience or preference for Microsoft stack—C#, .NET, Azure, etc.

You want a high-level and conceptual overview

Cloud-Native Terms

To Make Sure We’re All On The Same Page

What is “Cloud Native”?

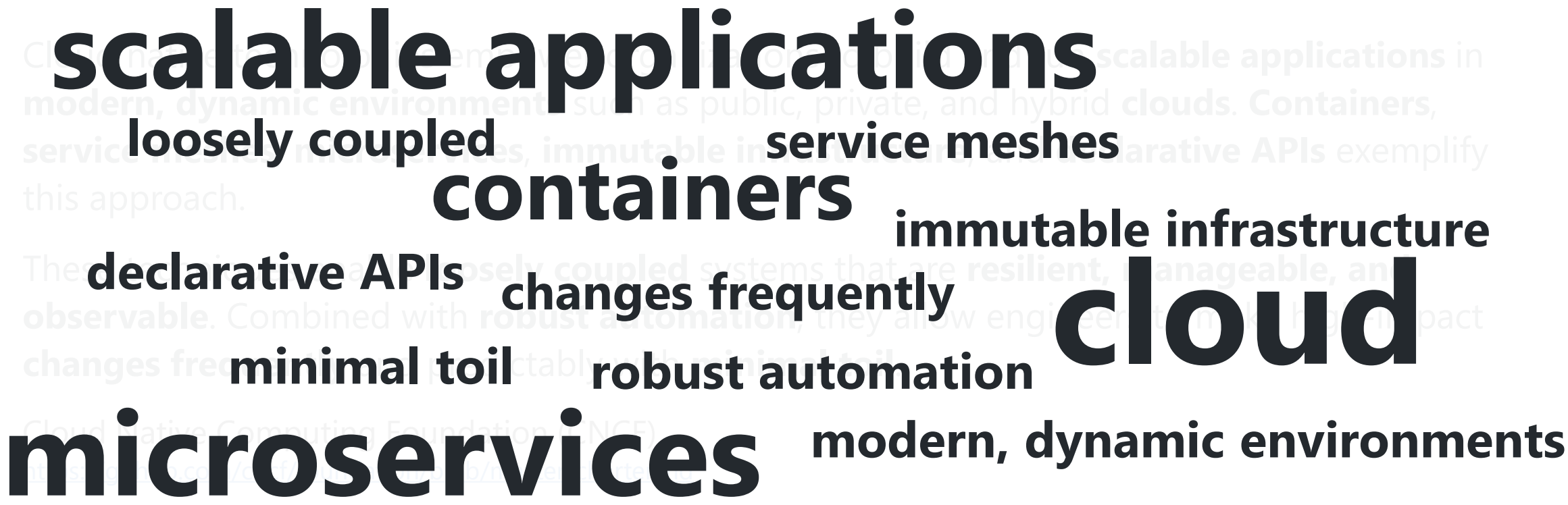
Cloud native technologies empower organizations to build and run **scalable applications** in **modern, dynamic environments** such as public, private, and hybrid **clouds**. **Containers**, **service meshes**, **microservices**, **immutable infrastructure**, and **declarative APIs** exemplify this approach.

These techniques enable **loosely coupled** systems that are **resilient, manageable, and observable**. Combined with **robust automation**, they allow engineers to make high-impact **changes frequently** and predictably with **minimal toil**.

Cloud Native Computing Foundation (CNCF)
https://github.com/cncf/foundation/blob/master/charter.md

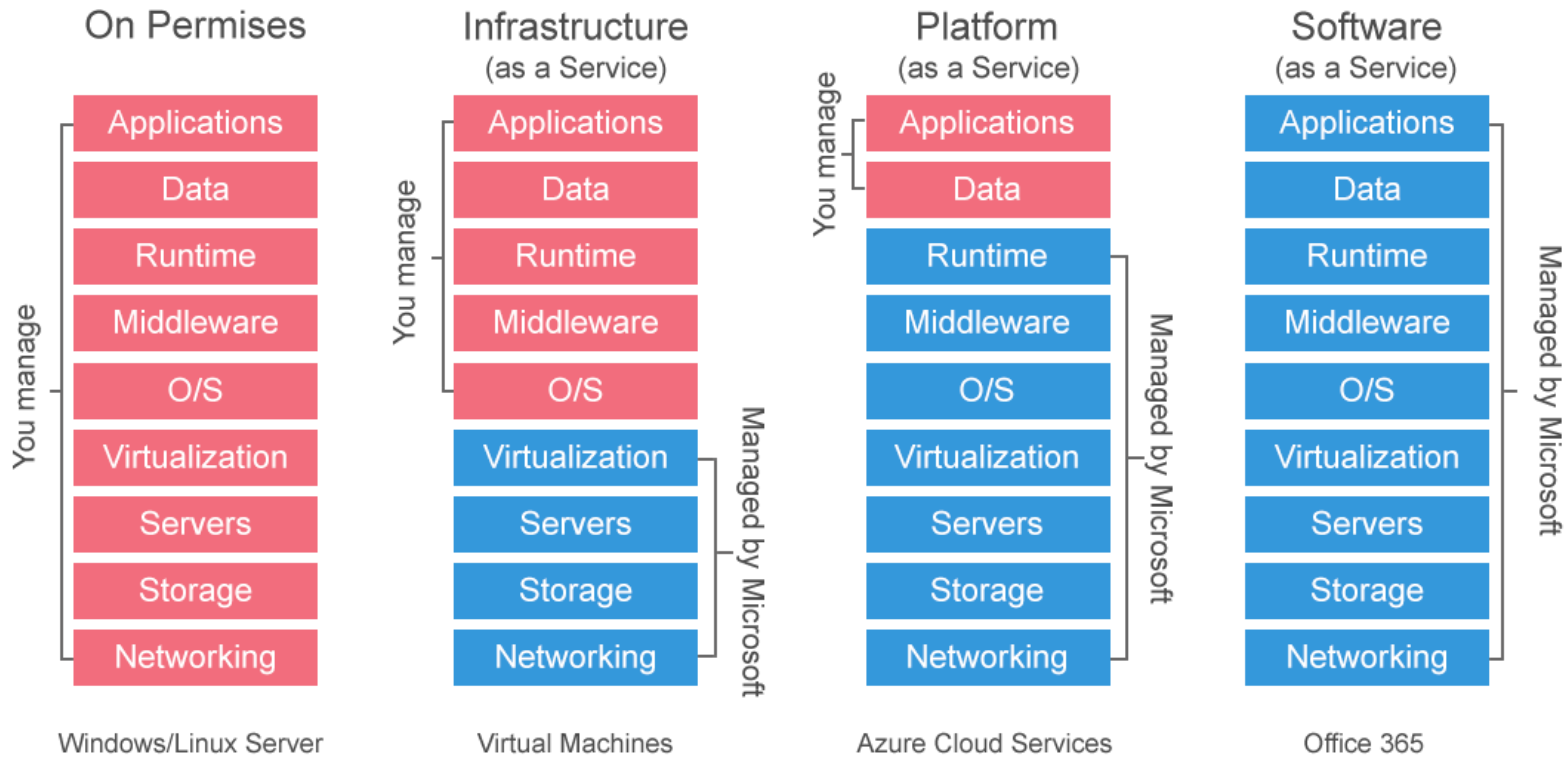
TRAILHEAD
TECHNOLOGY PARTNERS

What is “Cloud Native”?



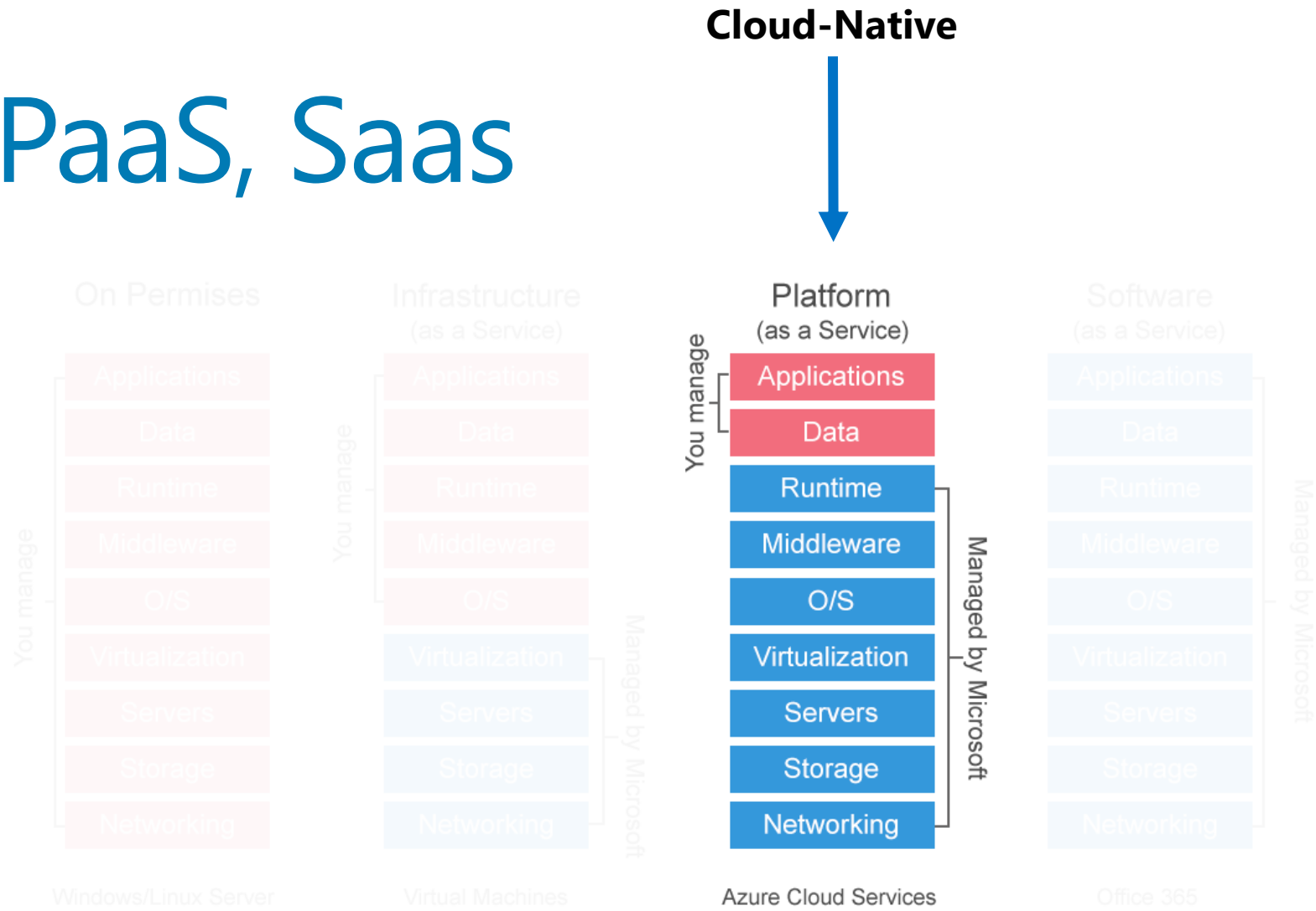
TRAILHEAD
TECHNOLOGY PARTNERS

IaaS, PaaS, Saas



TRAILHEAD
TECHNOLOGY PARTNERS

IaaS, PaaS, Saas



TRAILHEAD
TECHNOLOGY PARTNERS

Alt Definition: 12-Factor Applications

I. **Codebase**
One codebase tracked in revision control, many deploys

II. **Dependencies**
Explicitly declare and isolate dependencies

III. **Config**
Store config in the environment

IV. **Backing services**
Treat backing services as attached resources

V. **Build, release, run**
Strictly separate build and run stages

VI. **Processes**
Execute the app as one or more stateless processes

VII. **Port binding**
Export services via port binding

VIII. **Concurrency**
Scale out via the process model

IX. **Disposability**
Maximize robustness with fast startup and graceful shutdown

X. **Dev/prod parity**
Keep development, staging, and production as similar as possible

XI. **Logs**
Treat logs as event streams

XII. **Admin processes**
Run admin/management tasks as one-off processes

Source: https://12factor.net/

TRAILHEAD TECHNOLOGY PARTNERS

Beyond 12-Factor

XIII. API First
Make everything a service. Assume your code will be consumed by a front-end client, gateway, or another service.

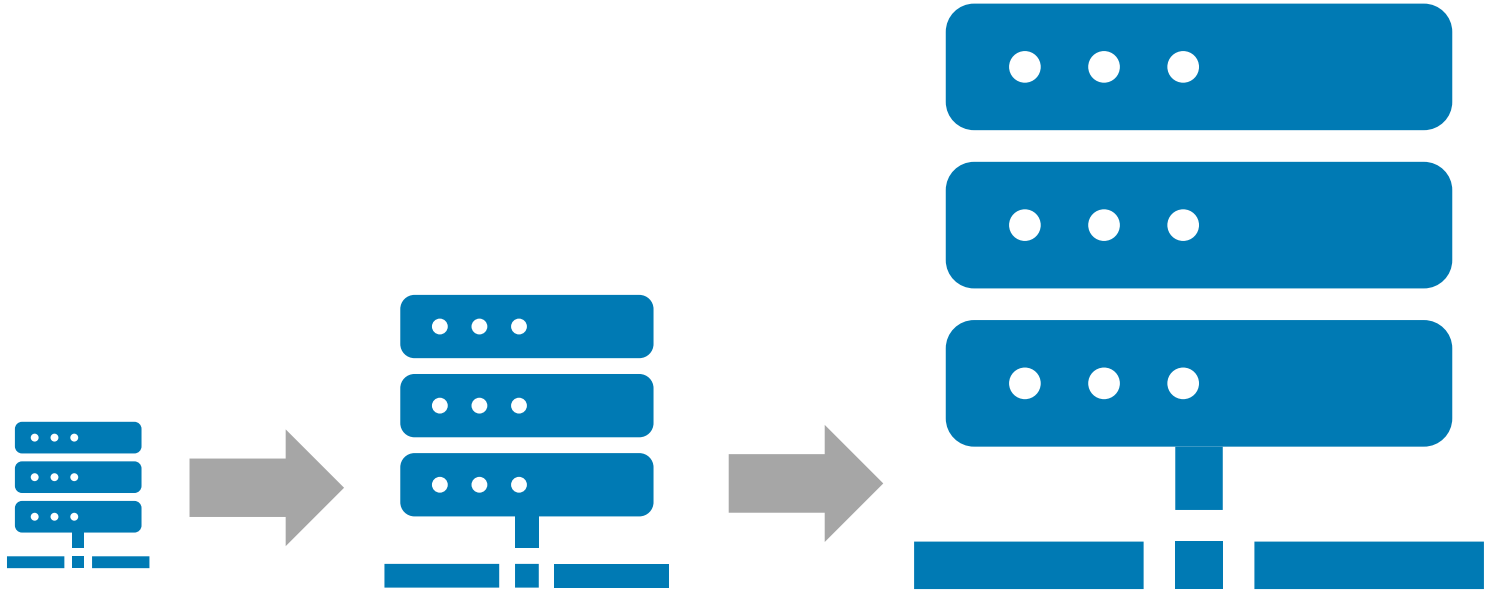
XIV. Telemetry
On a workstation, you have deep visibility into your application and its behavior. In the cloud, you don't. Make sure your design includes the collection of monitoring, domain-specific, and health/system data.

XV. Authentication/Authorization
Implement identity from the start. Consider RBAC (role-based access control) features available in public clouds.

Source: https://tanzu.vmware.com/content/blog/beyond-the-twelve-factor-app

Scaling

Scale Up (Vertical)



Scale Out (Horizontal)

TRAILHEAD
TECHNOLOGY PARTNERS

Cloud-Native Patterns

And How They Help

Challenges in Cloud Software

AVAILABILITY

DATA MANAGEMENT

DESIGN AND IMPLEMENTATION

MESSAGING

MANAGEMENT AND MONITORING

PERFORMANCE AND SCALABILITY

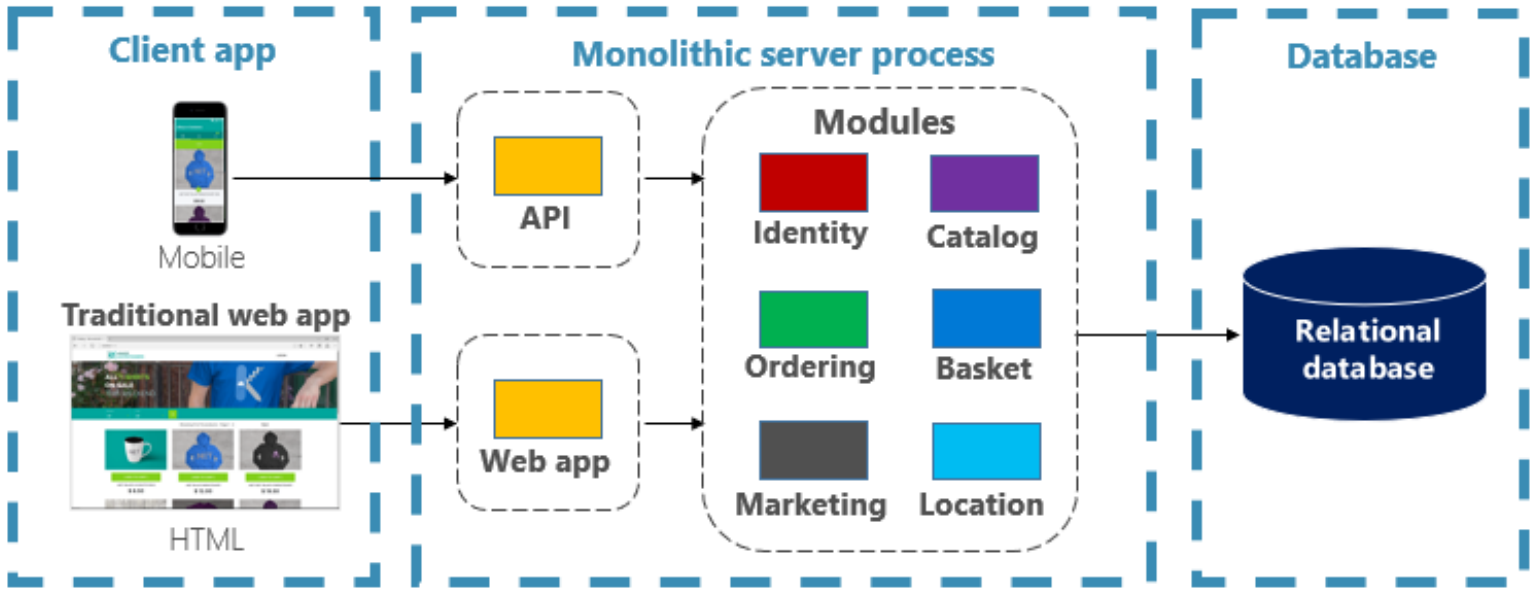
RESILIENCY

SECURITY

TRAILHEAD TECHNOLOGY PARTNERS

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

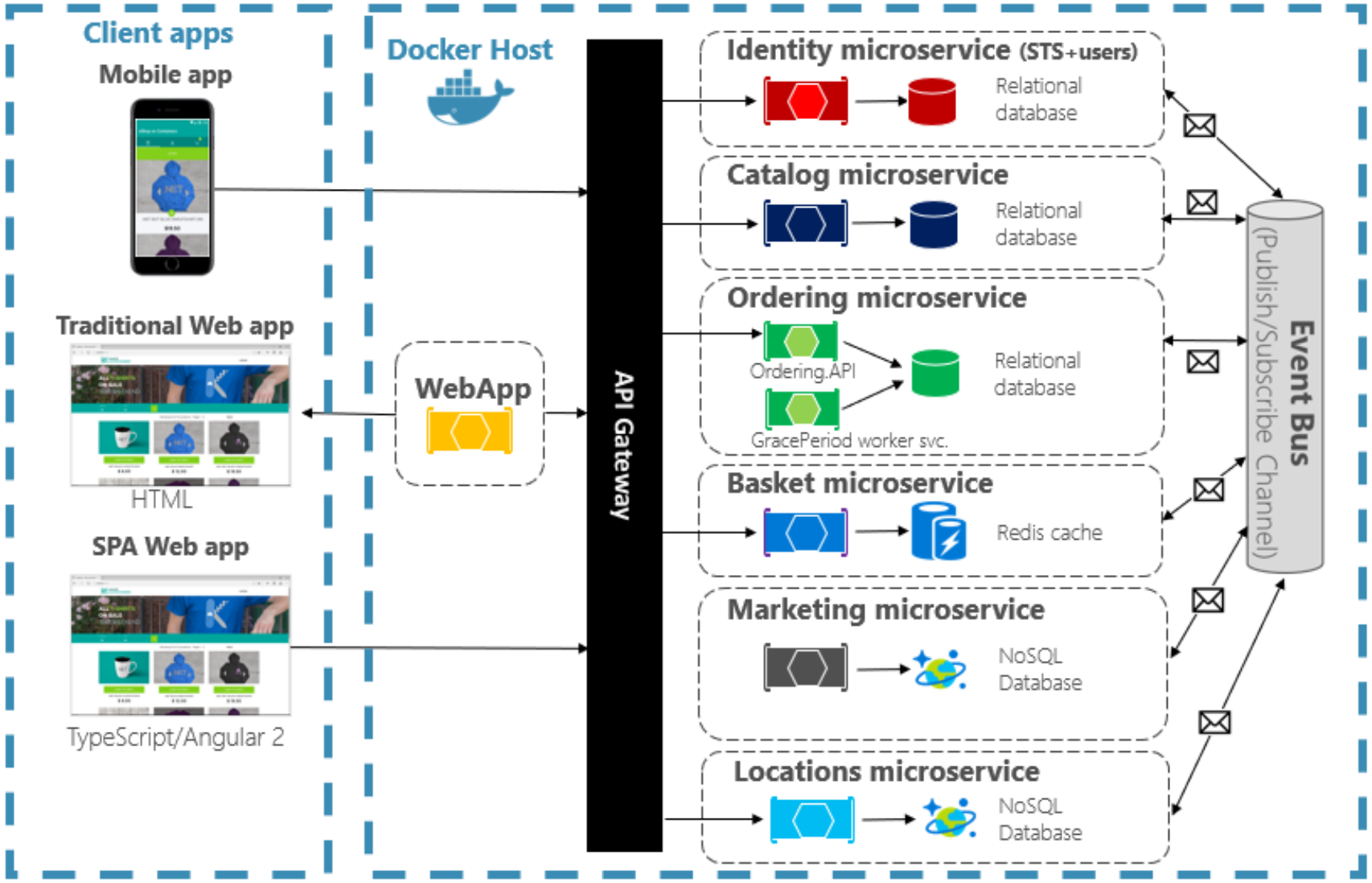
Microservices



Source: https://docs.microsoft.com/en-us/dotnet/architecture/cloud-native/introduction

TRAILHEAD TECHNOLOGY PARTNERS

Microservices



Source: https://docs.microsoft.com/en-us/dotnet/architecture/cloud-native/introduction

Microservices Advantages

Isolation

Scalability

Productivity

Flexibility

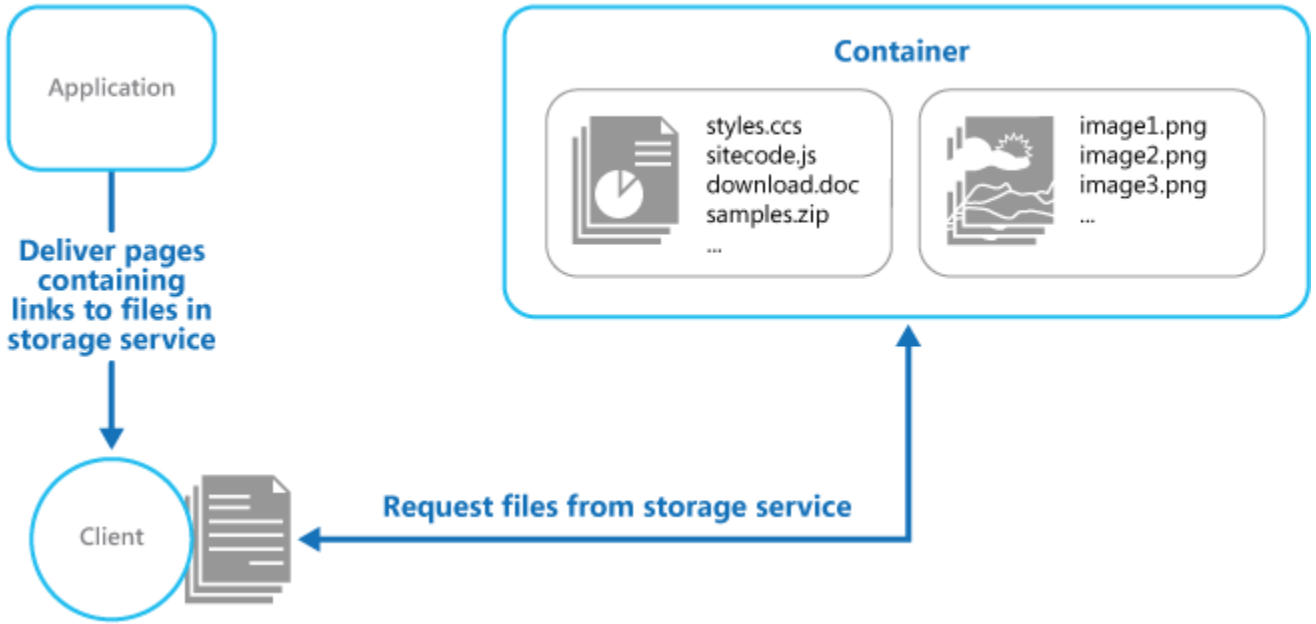
Faster development

Evolutionary

TRAILHEAD TECHNOLOGY PARTNERS

Static Content Hosting Pattern

Design and Implementation, Data Management, Performance and Scalability



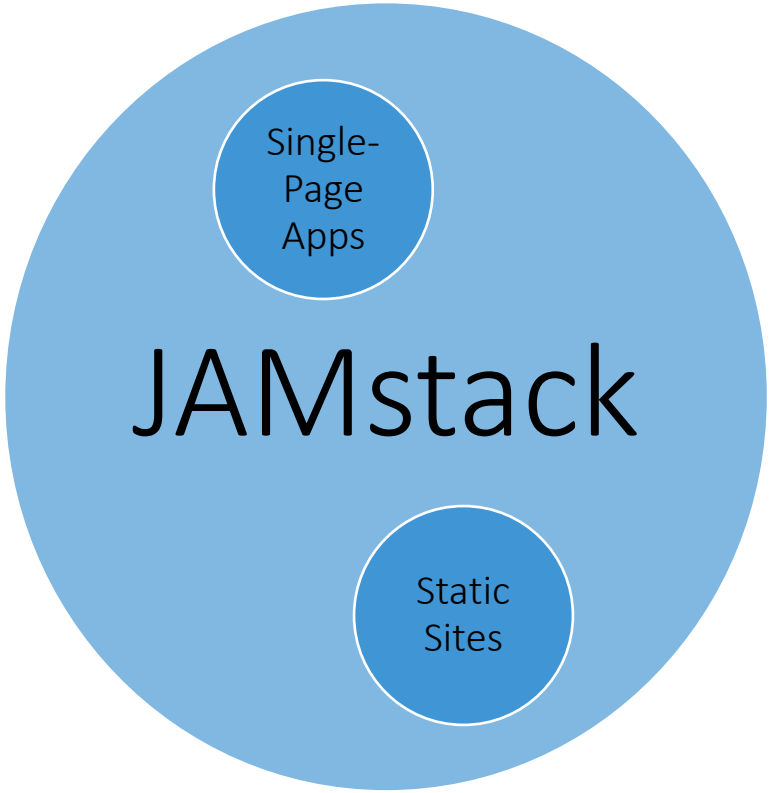
TRAILHEAD TECHNOLOGY PARTNERS

JAMstack

TRAILHEAD
TECHNOLOGY PARTNERS

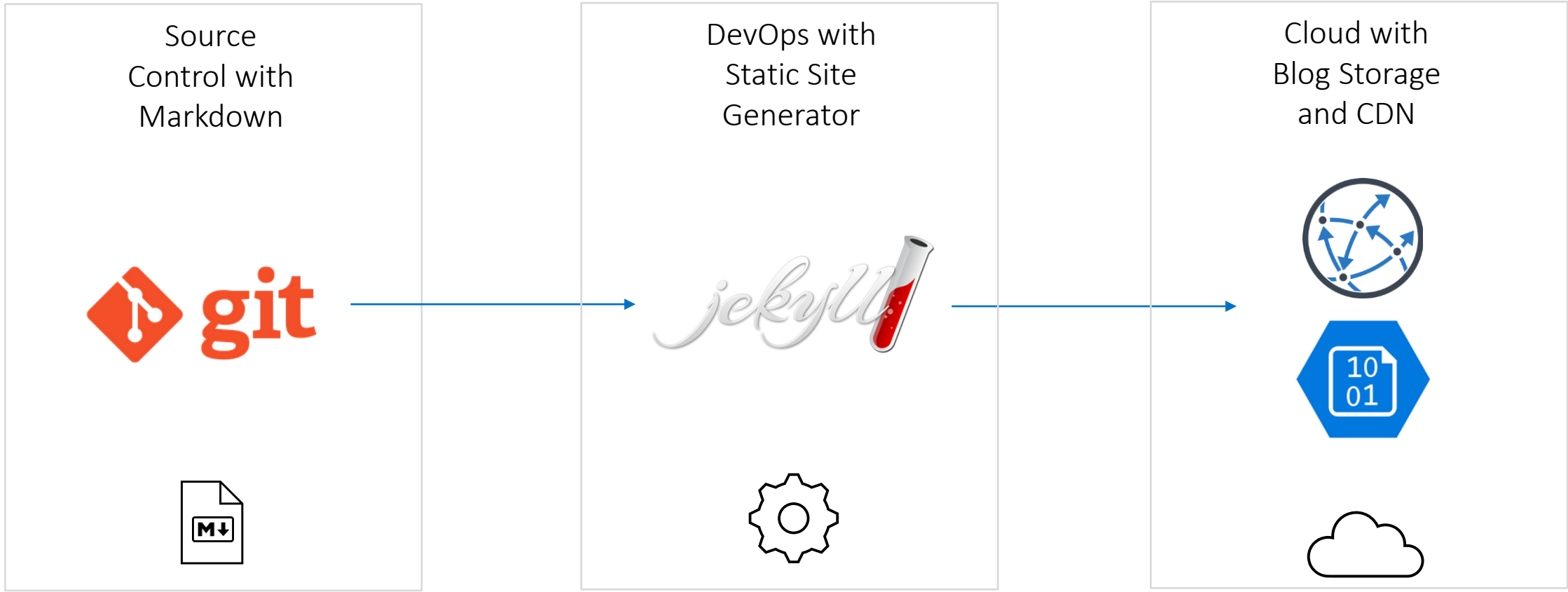
Source: https://www.freecodecamp.org/news/what-is-the-jamstack-and-how-do-i-host-my-website-on-it/

JAMstack



TRAILHEAD
TECHNOLOGY PARTNERS

JAMstack Scenario: Static Site



TRAILHEAD TECHNOLOGY PARTNERS

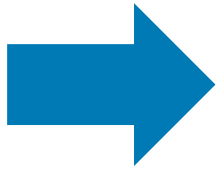
Static Site Case Study

Popular CMS

$50/mo

400ms avg response

No Scalability



Static Generated Files

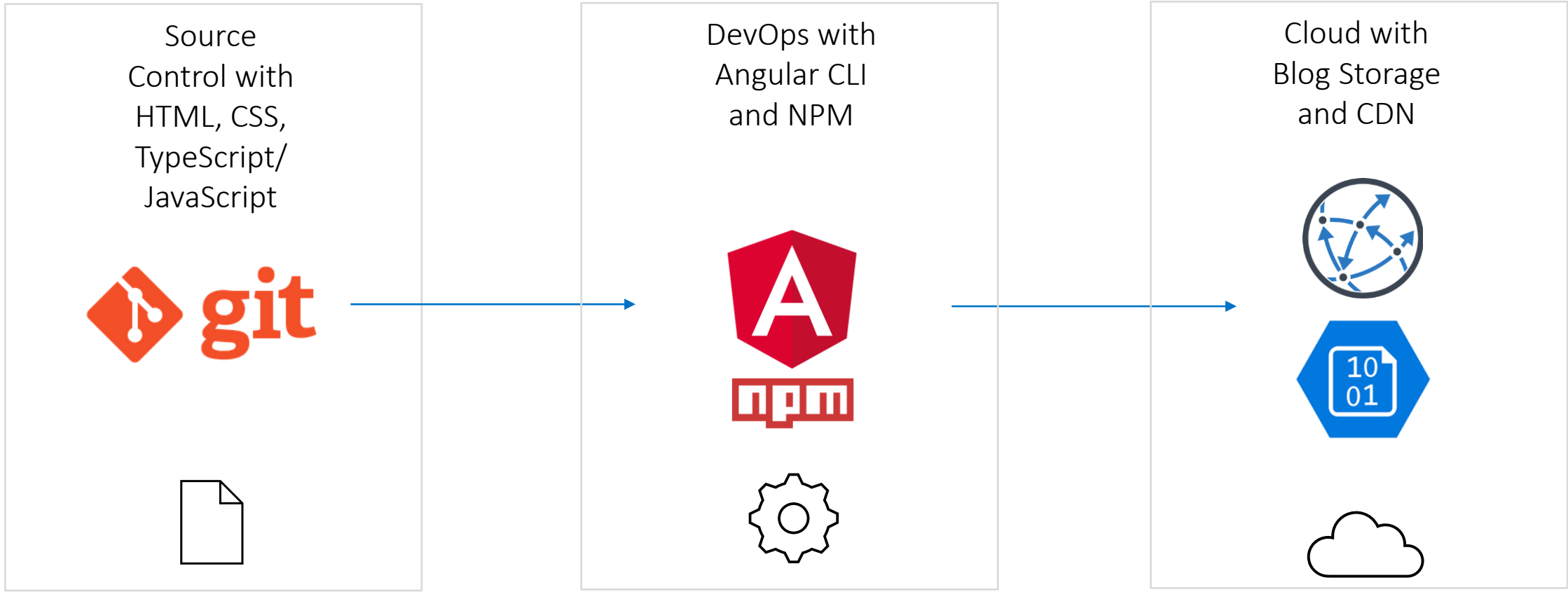
$1-3/mo

<100ms avg response

Easy Scalability

TRAILHEAD
TECHNOLOGY PARTNERS

JAMstack Scenario: SPA

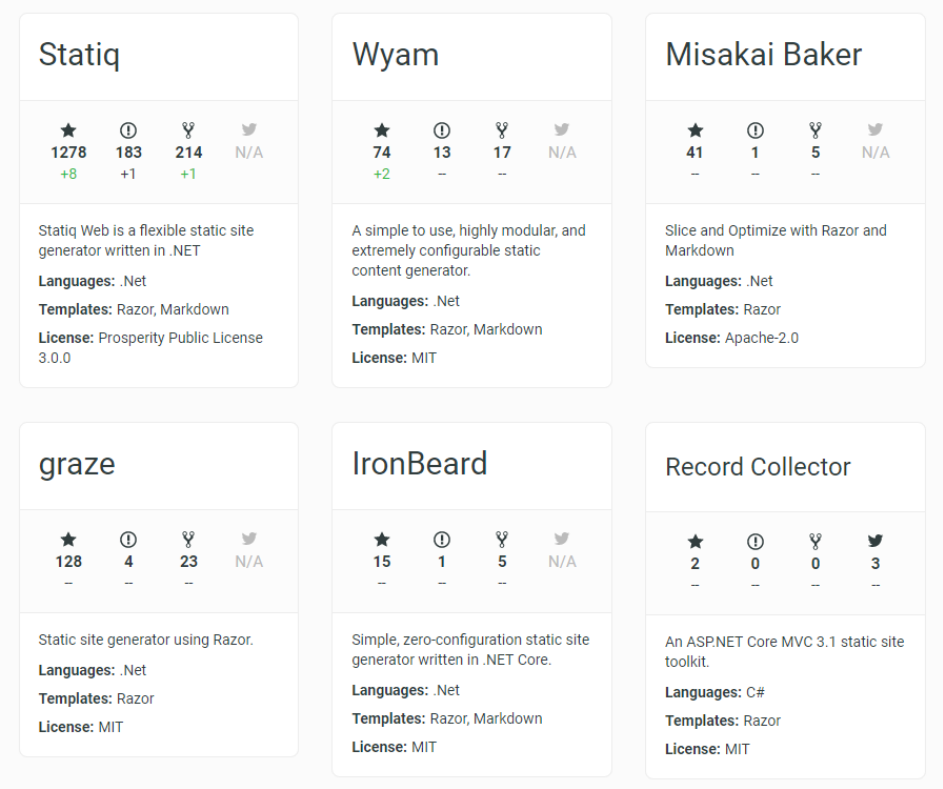


TRAILHEAD TECHNOLOGY PARTNERS

Popular Static Site Generators

TRAILHEAD
TECHNOLOGY PARTNERS

.NET/Razor Static Site Generators



Statiq

★ 1278 +8 | ⓘ 183 +1 | 🍴 214 +1 | 🐦 N/A

Statiq Web is a flexible static site generator written in .NET

Languages: .Net

Templates: Razor, Markdown

License: Prosperity Public License 3.0.0

Wyam

★ 74 +2 | ⓘ 13 – | 🍴 17 – | 🐦 N/A

A simple to use, highly modular, and extremely configurable static content generator.

Languages: .Net

Templates: Razor, Markdown

License: MIT

Misakai Baker

★ 41 – | ⓘ 1 – | 🍴 5 – | 🐦 N/A

Slice and Optimize with Razor and Markdown

Languages: .Net

Templates: Razor

License: Apache-2.0

graze

★ 128 – | ⓘ 4 – | 🍴 23 – | 🐦 N/A

Static site generator using Razor.

Languages: .Net

Templates: Razor

License: MIT

IronBeard

★ 15 – | ⓘ 1 – | 🍴 5 – | 🐦 N/A

Simple, zero-configuration static site generator written in .NET Core.

Languages: .Net

Templates: Razor, Markdown

License: MIT

Record Collector

★ 2 – | ⓘ 0 – | 🍴 0 – | 🐦 3 –

An ASP.NET Core MVC 3.1 static site toolkit.

Languages: C#

Templates: Razor

License: MIT

TRAILHEAD TECHNOLOGY PARTNERS

Source: https://www.staticgen.com/

If You Give $200, So Will I!

bit.ly/chicago-cloud

"charity:water is a non-profit organization that provides clean and safe drinking water to people in developing nations. The organization was founded in 2006 and has helped fund 22,936 projects in 24 countries, benefiting over 4.6 million people." - Wikipedia

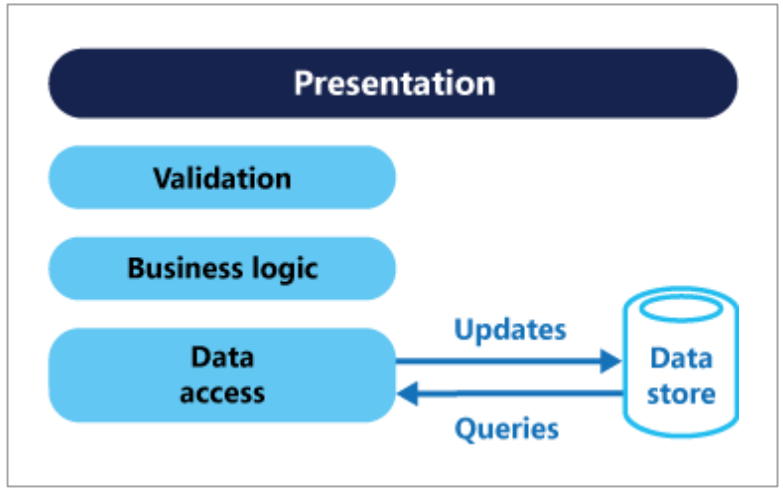
"4/4 Stars"
- CharityNavigator.org

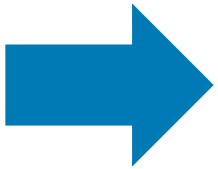
charity: water

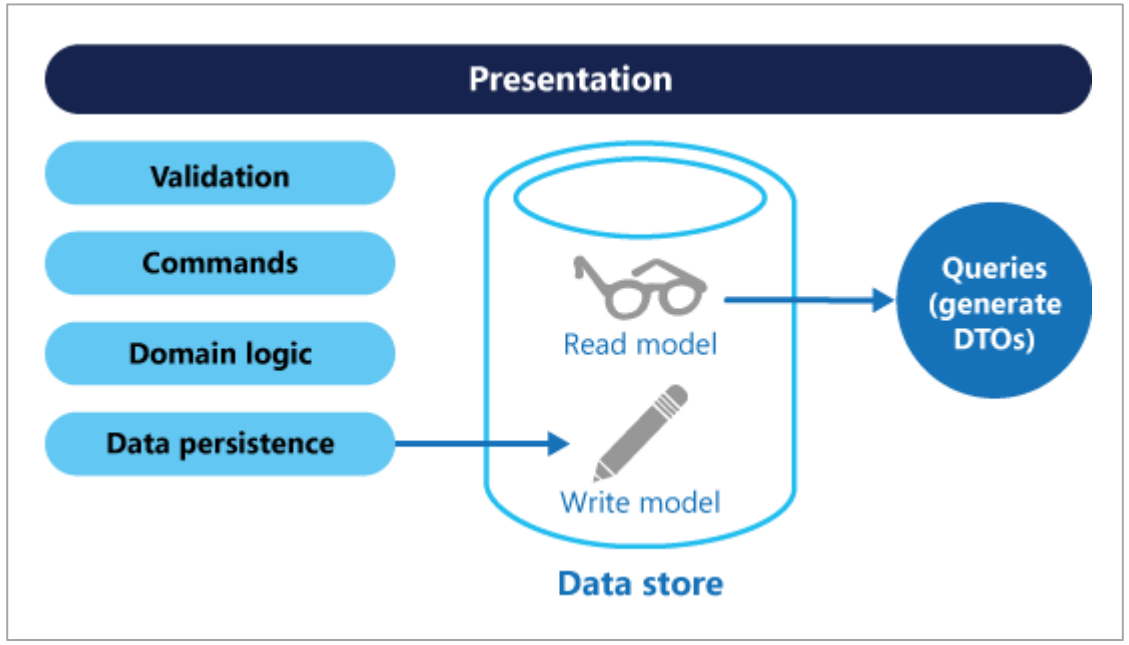
CQRS Pattern

"Command and Query Responsibility Segregation"

Data Management, Design and Implementation, Performance and Scalability







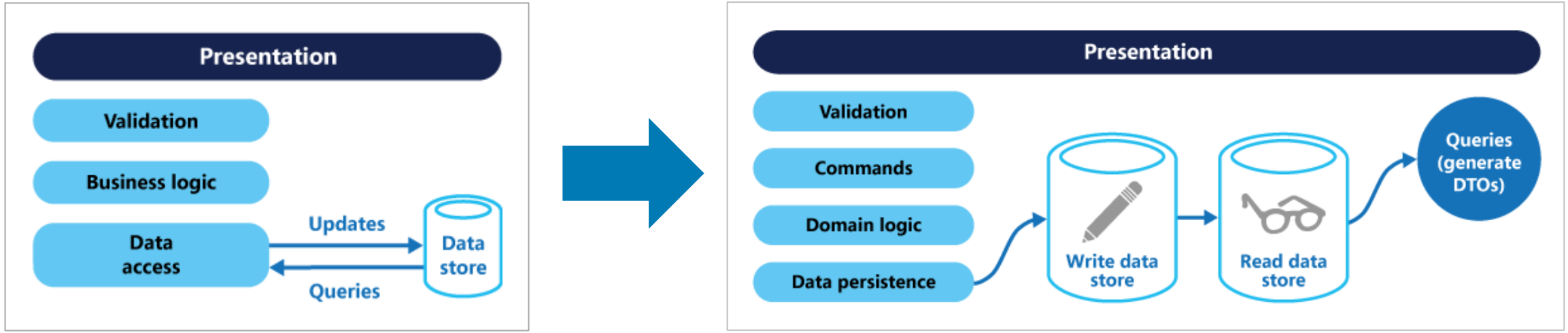
TRAILHEAD TECHNOLOGY PARTNERS

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

CQRS Pattern

"Command and Query Responsibility Segregation"

Data Management, Design and Implementation, Performance and Scalability

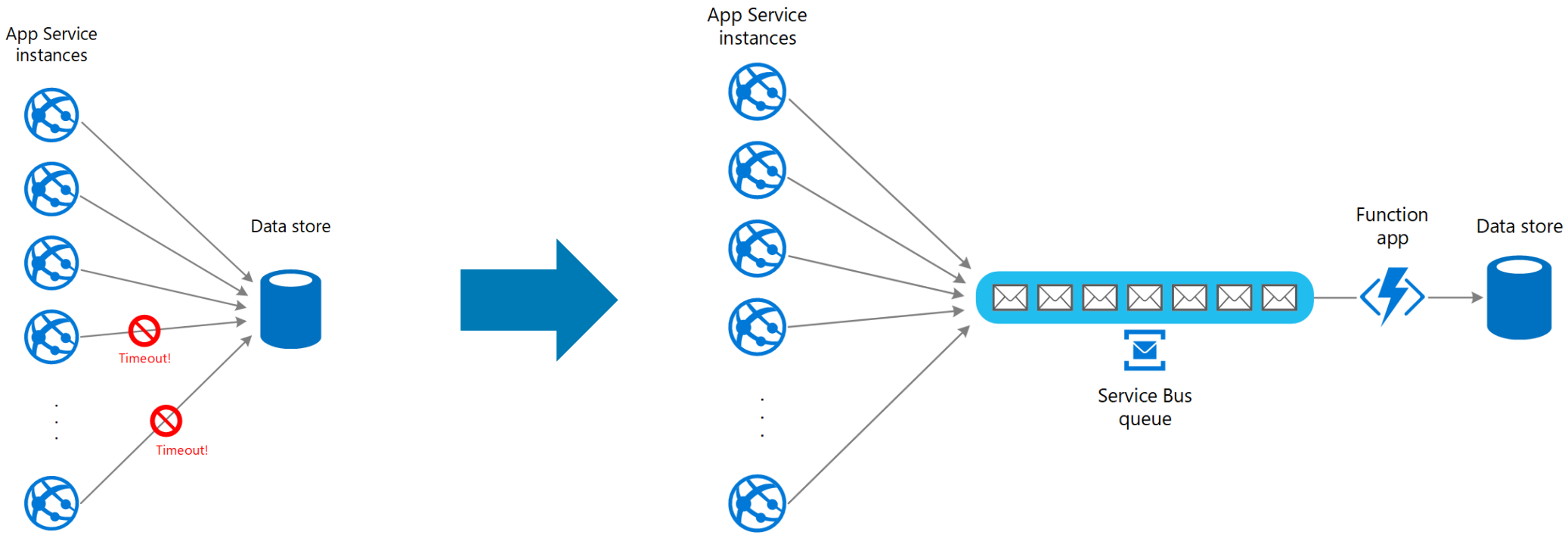


TRAILHEAD TECHNOLOGY PARTNERS

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

Queue-Based Load Leveling Pattern

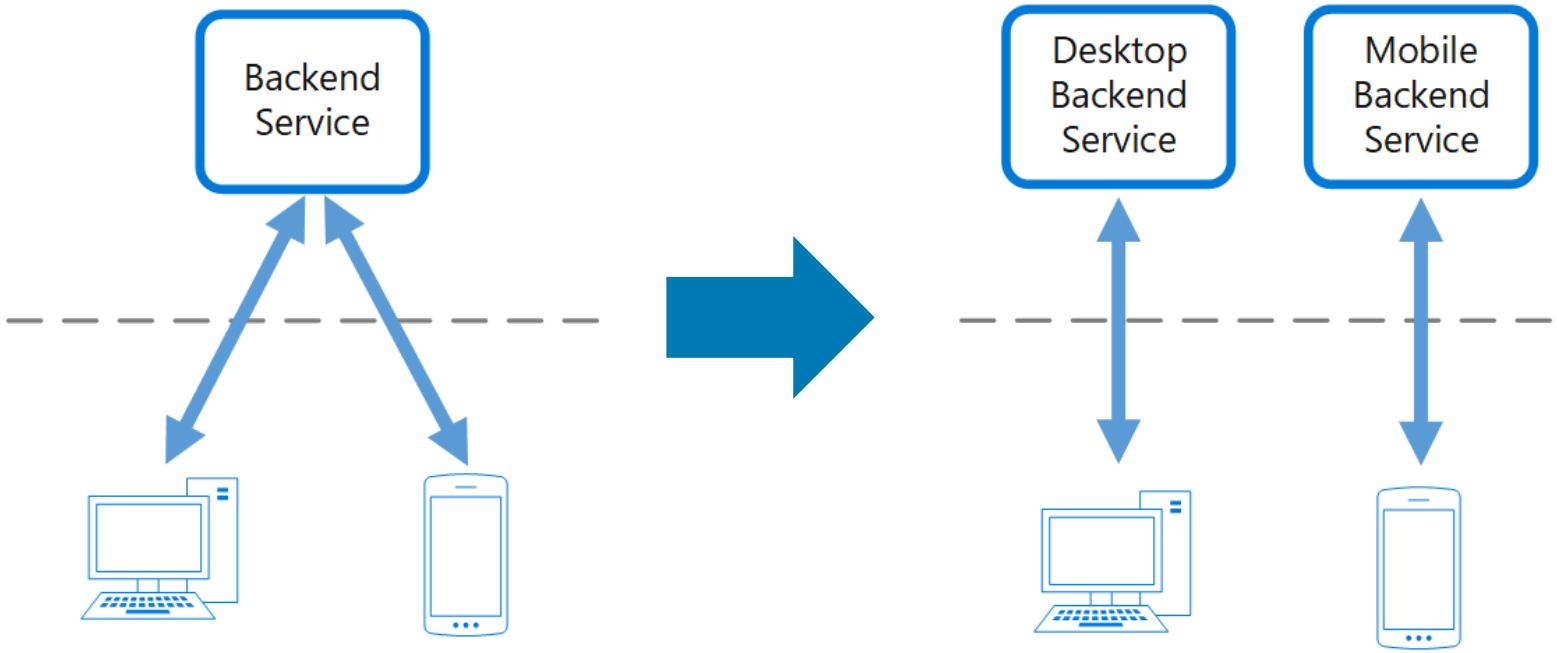
Availability, Messaging, Resiliency, Performance and Scalability



TRAILHEAD TECHNOLOGY PARTNERS

Backends For Frontends Pattern

Design and Implementation



Source: https://docs.microsoft.com/en-us/azure/architecture/patterns/backends-for-frontends

TRAILHEAD TECHNOLOGY PARTNERS

Cloud-Native Technologies

And When to Use Them

.NET Core

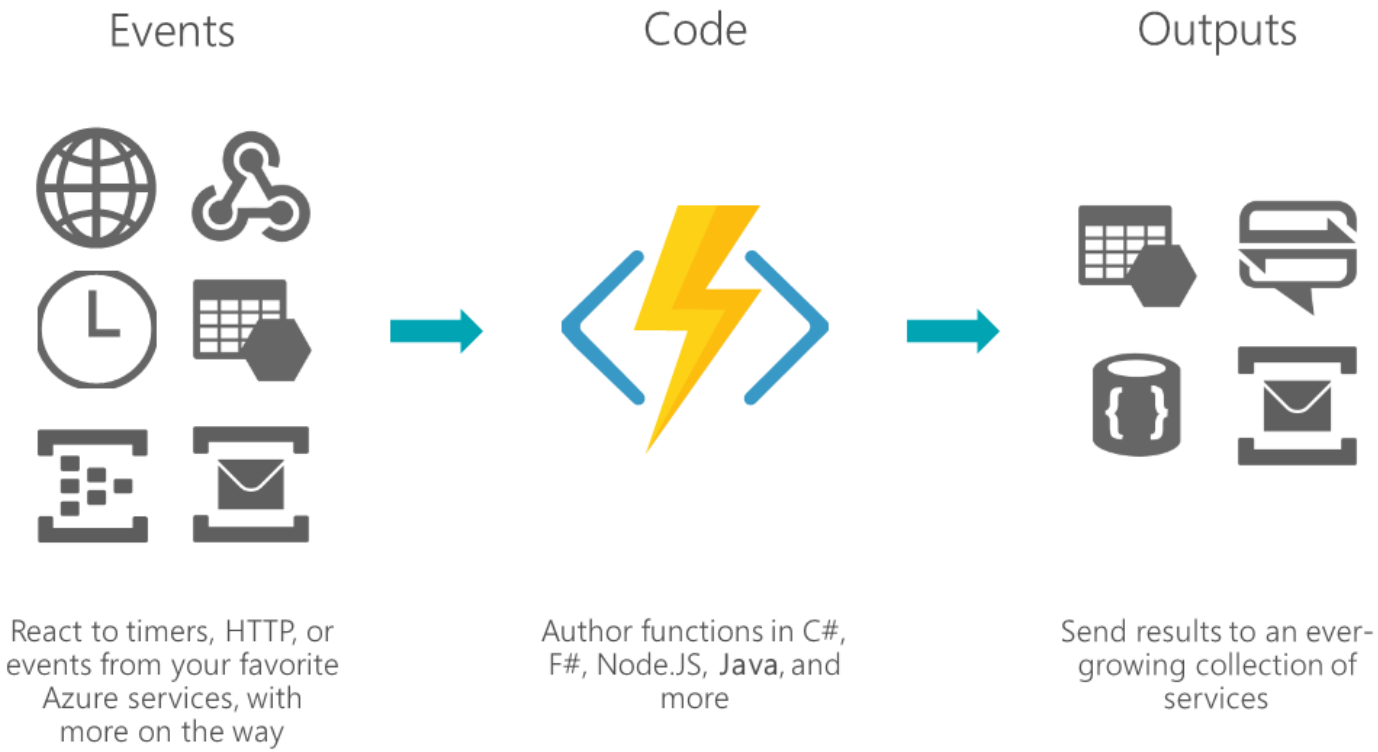
It’s (past?)
time to switch

.NET Unified
by .NET 5

The .NET
Standard

TRAILHEAD
TECHNOLOGY PARTNERS

“Serverless” with Azure Functions

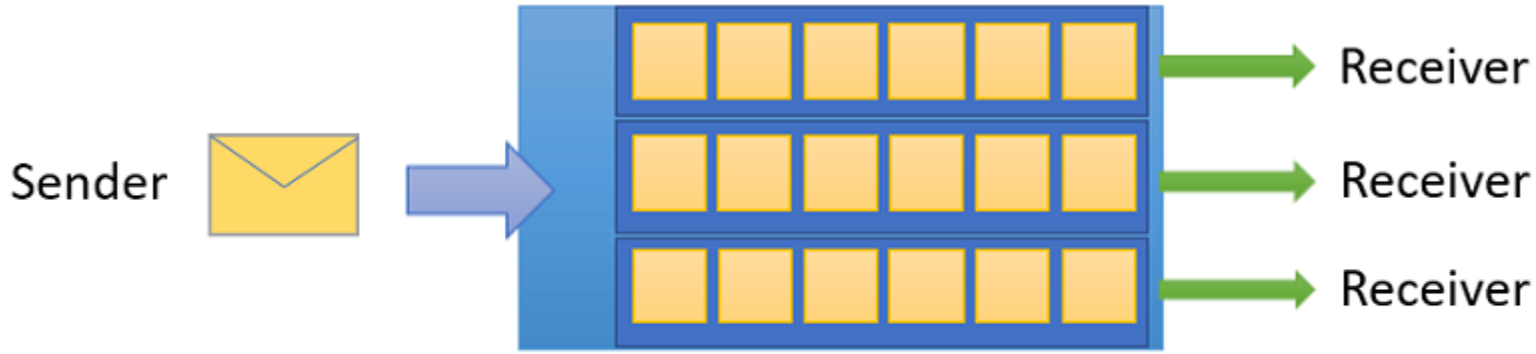


TRAILHEAD TECHNOLOGY PARTNERS

Source: https://medium.com/@lucasmonastirsky/build-a-basic-rest-api-quickly-with-azure-functions-and-docker-fcaf987bb033

Azure Service Bus

Message Queue with Messages



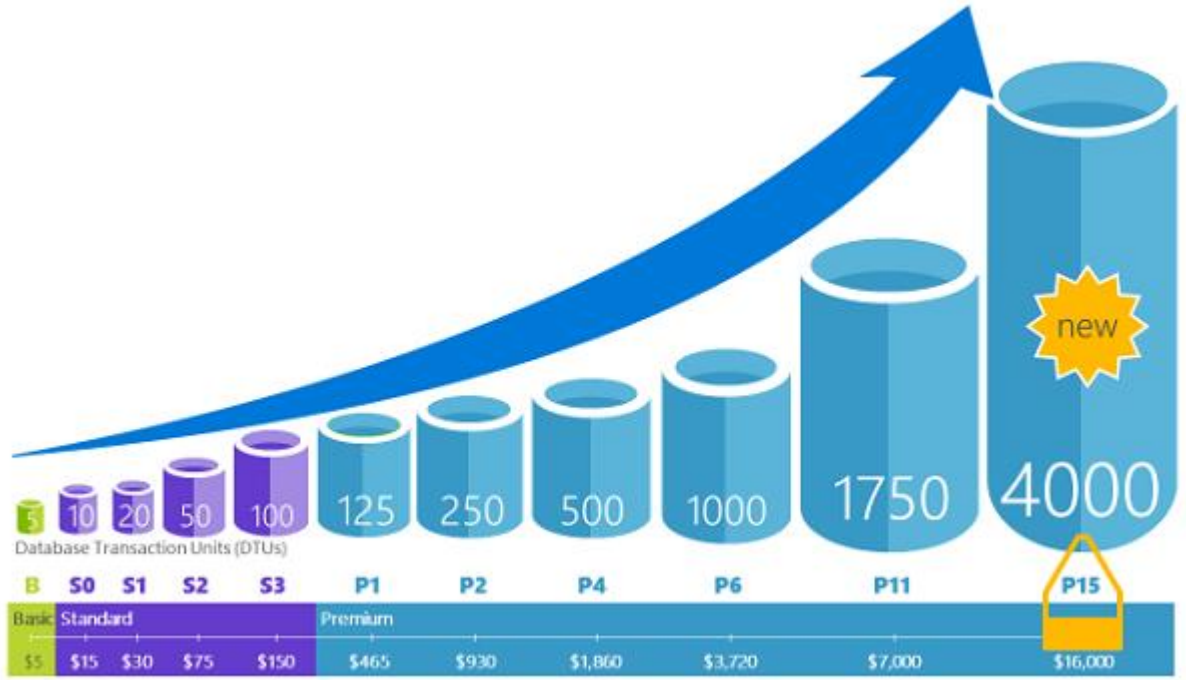
Topic with three Subscriptions
with Messages

TRAILHEAD
TECHNOLOGY PARTNERS

Scalable RDBMS on SQL Azure

Automated scaling on-demand

NOTE: scales VERTICAL only



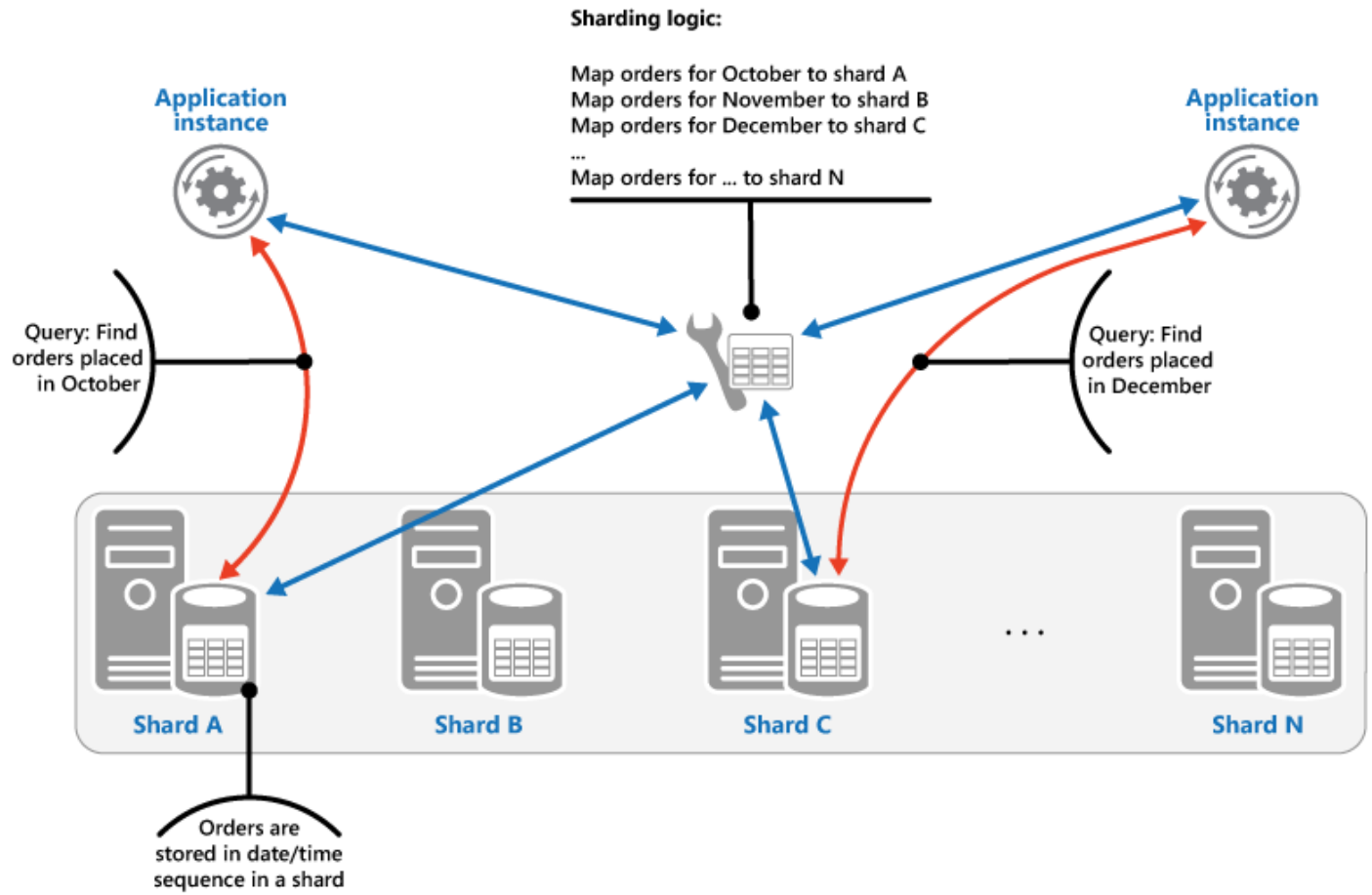
TRAILHEAD
TECHNOLOGY PARTNERS

Scalable RDBMS with Sharding

Data Management, Performance and Scalability

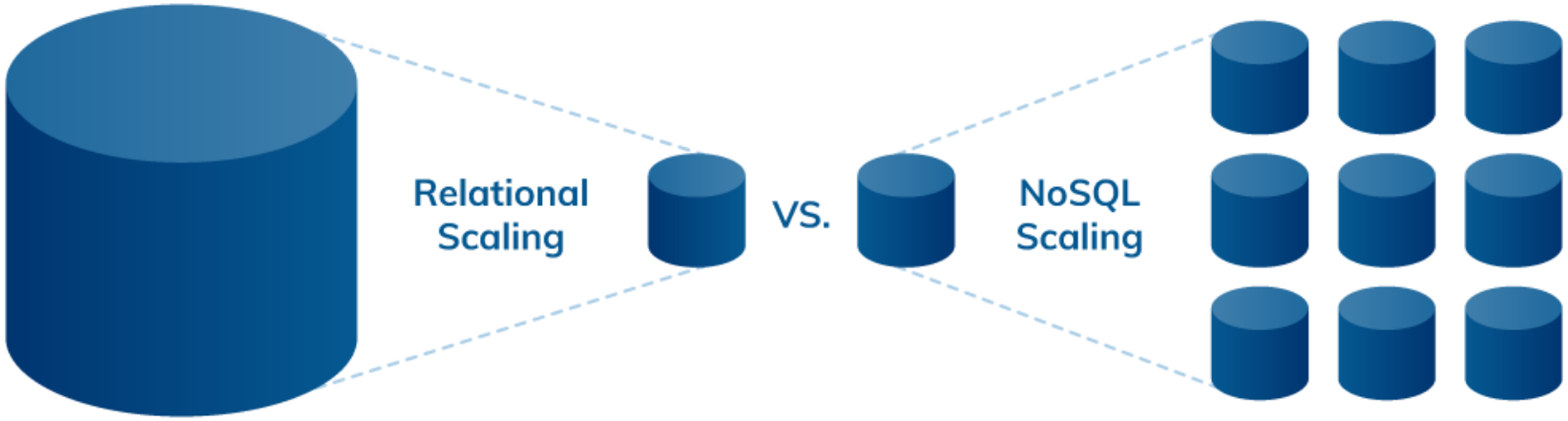
Sharding divides data into set of horizontal partitions or “shards”

Ex: Year, Tenant (user/account)



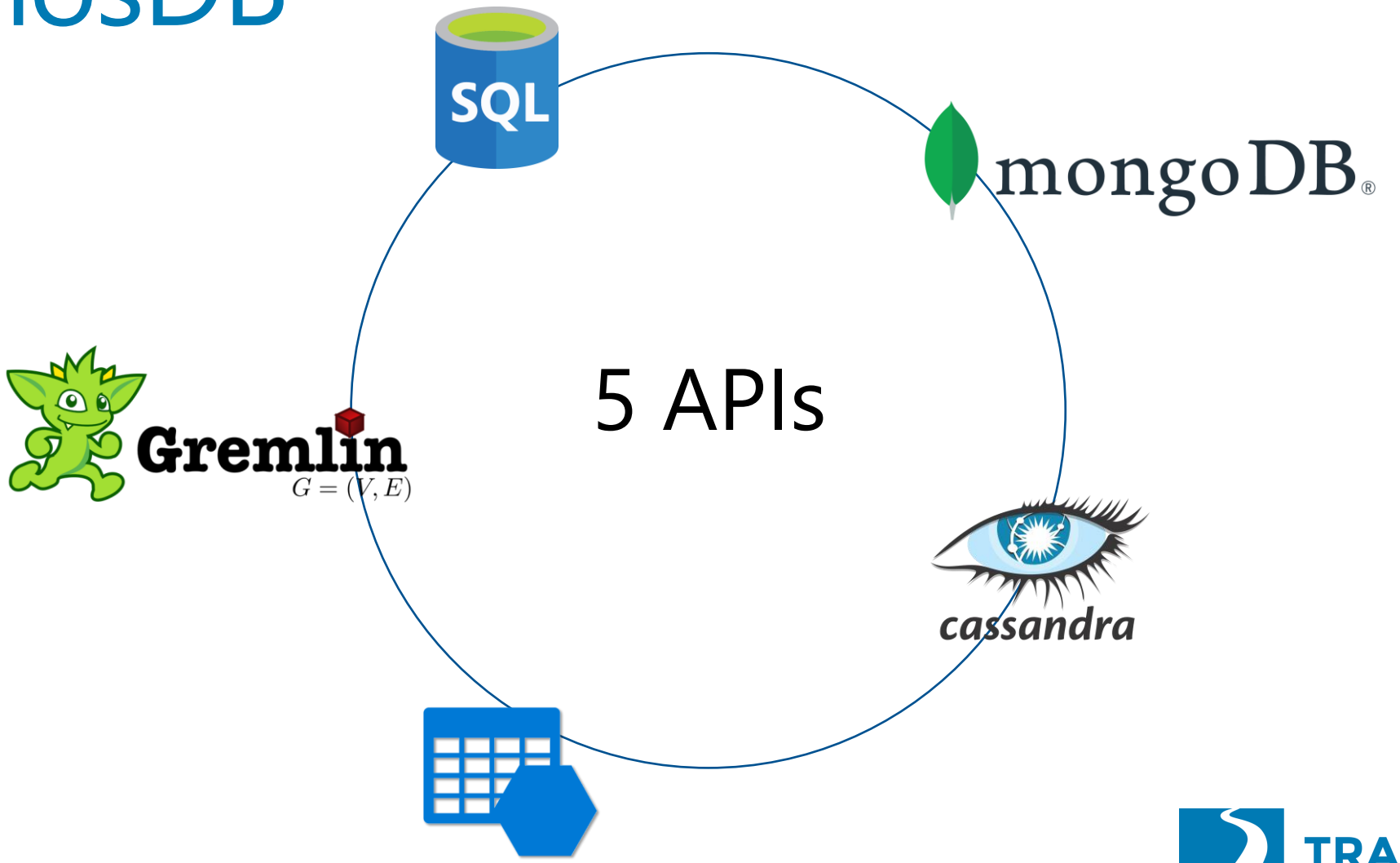
TRAILHEAD
TECHNOLOGY PARTNERS

CosmosDB



Source: https://www.datastax.com/nosql

CosmosDB



TRAILHEAD
TECHNOLOGY PARTNERS

DevOps in the Cloud

Azure Boards

Plan, track, and discuss work across teams, deliver value to your users faster.

Azure Repos

Unlimited cloud-hosted private Git repos. Collaborative pull requests, advanced file management, and more.

Azure Pipelines

CI/CD that works with any language, platform, and cloud. Connect to GitHub or any Git provider and deploy continuously to any cloud.

Azure Test Plans

The test management and exploratory testing toolkit that lets you ship with confidence.

Azure Artifacts

Create, host, and share packages. Easily add artifacts to CI/CD pipelines.

TRAILHEAD
TECHNOLOGY PARTNERS

Containers and Orchestration

Azure Container Instances (ACI)

Azure Container Service (ACS)

Kubernetes, DC/OS, or Docker Swarm clusters

Azure Kubernetes Service (AKS)

"Serverless" Kubernetes

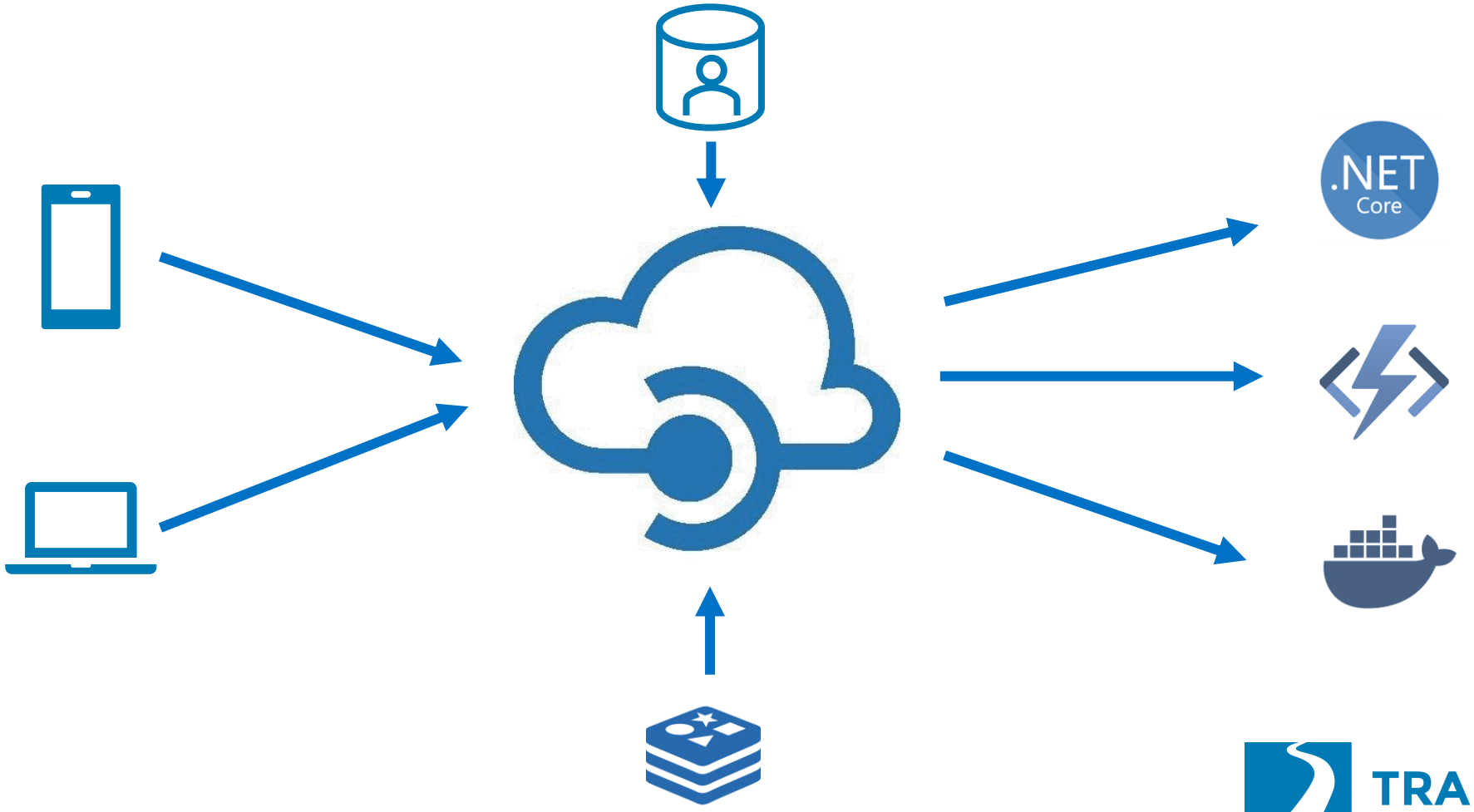
Web App for Containers

App Service for Docker Containers

Azure Service Fabric

TRAILHEAD
TECHNOLOGY PARTNERS

Azure API Management



TRAILHEAD
TECHNOLOGY PARTNERS

Azure Service Fabric & Service Fabric Mesh

Azure Service Fabric

- ≈ Like Kubernetes+ (adds SDK, updates, CI/CD, rollback, etc)
- ≈ Dog food
- ≈ On-prem, multi-cloud, and OS-neutral
- ≈ Cluster/node/VM-based (CPaaS)

Azure Service Fabric Mesh

- ≈ Like AKS+
- ≈ True PaaS, managed clusters
- ≈ "Serverless" Service Fabric

TRAILHEAD
TECHNOLOGY PARTNERS

If You Give $200, So Will I!

bit.ly/chicago-cloud

"charity:water is a non-profit organization that provides clean and safe drinking water to people in developing nations. The organization was founded in 2006 and has helped fund 22,936 projects in 24 countries, benefiting over 4.6 million people." - Wikipedia

"4/4 Stars"
- CharityNavigator.org

charity: water

Recap

Assumptions

Background

Patterns

Cloud-Native Technologies

TRAILHEAD
TECHNOLOGY PARTNERS

Future Reading

Architecting Cloud Native .NET Applications for Azure
https://docs.microsoft.com/en-us/dotnet/architecture/cloud-native/

eShopOnContainers reference app
https://github.com/dotnet-architecture/eShopOnContainers

Serverless apps: Architecture, patterns, and Azure implementation
https://docs.microsoft.com/en-us/dotnet/architecture/serverless/

.NET Microservices: Architecture for Containerized .NET Applications
https://docs.microsoft.com/en-us/dotnet/architecture/microservices/

TRAILHEAD
TECHNOLOGY PARTNERS

Thanks! Questions?

Jonathan "J." Tower
Partner & Principal Consultant
Trailhead Technology Partners

TRAILHEAD
TECHNOLOGY PARTNERS

trailheadtechnology.com

- Microsoft MVP in ASP.NET
- Business Owner
- Organizer of Beer City Code

- jtower@trailheadtechnology.com
- trailheadtechnology.com/blog
- jtowermi

github.com/jonathantower/cloud-native-dotnet