Building Microservices

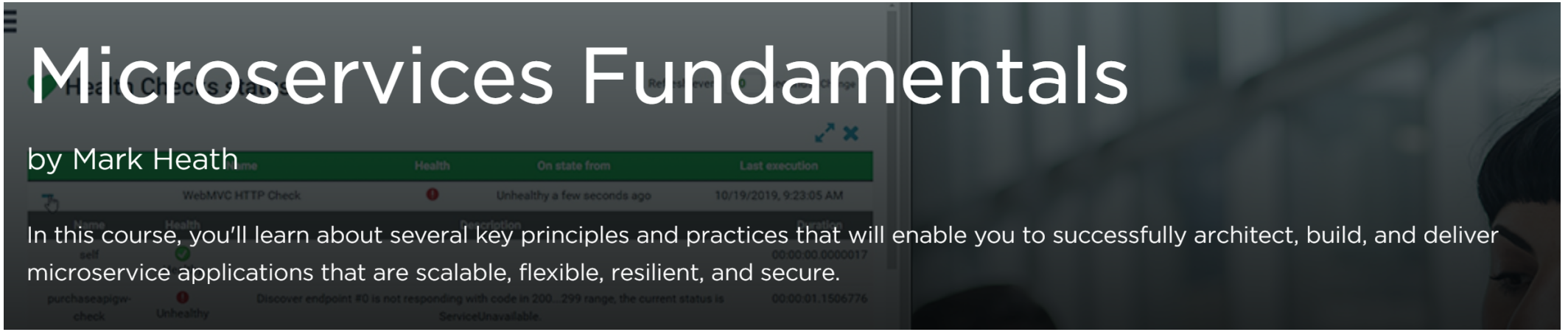
GETTING STARTED WITH BUILDING MICROSERVICES

Mark Heath
CLOUD ARCHITECT

@mark_heath www.markheath.net

Microservices on Pluralsight

This course is part of the microservices learning path on Pluralsight



https://app.pluralsight.com/library/courses/microservices-fundamentals

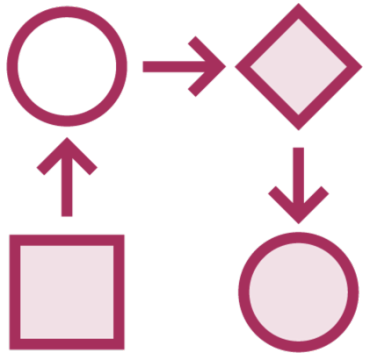
Architecting microservices

Building microservices

Communicating between microservices

Securing and delivering microservices

In This Course



Domain Logic

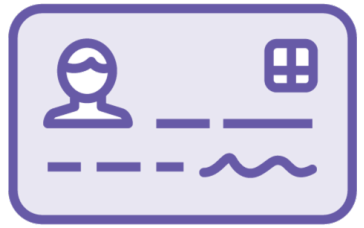
How to structure the code?

Data mapping

Testing

Types of tests

Combined strategy



Authorization

Inter-service communication

Security

Overview

Microservices give us flexibility and freedom

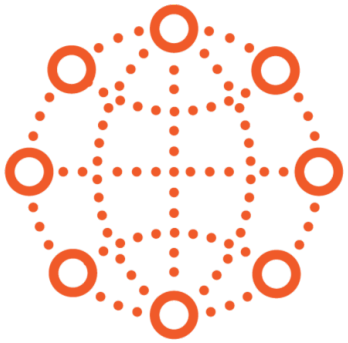
Value of standardization

eShopOnContainers sample application

- Architectural overview
- How to run it locally with Docker

What Are Microservices?

autonomous, independently deployable services...

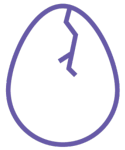


...collaborate to form an application

They are small enough to be rewritten if necessary

They own their own data

Other microservices can only access that data through the public API



Avoid breaking changes to the API

Microservices Give You Options

Programming Language

In this course: C#

Java, Node.js, Python, Go, etc.

Database

Relational (e.g. SQL Server)

Document (e.g. Mongo)

In-memory (e.g. Redis)

Communication Style

HTTP-based web API

Messaging via an event bus

RESTful APIs, gRPC

Hosting Platform

Docker and Kubernetes

Serverless Functions-as-a-Service

Directly on Virtual Machines

Flexibility and Freedom

Choose the best tool for the job

Adopt new technologies

Avoid getting stuck on legacy frameworks

Benefits of Standardization

- Developer productivity
- Consistent approach to deployment and monitoring
- Logging
- Health checks
- Configuration
- Authentication

Microservices give you freedom to choose the best tool for the job

Microservice applications benefit from standardization to boost developer productivity

Introducing our demo application: eShopOnContainers

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

dotnet-architecture / **eShopOnContainers**

Watch 1,377 | Unstar 11,384 | Fork 4,638

Code | Issues 28 | Pull requests 10 | Projects 1 | Wiki | Security | Insights

Easy to get started sample reference microservice and container based application. Cross-platform on Linux and Windows Docker Containers, powered by .NET Core 2.2, Docker engine and optionally Azure, Kubernetes or Service Fabric. Supports Visual Studio, VS for Mac and CLI based environments with Docker CLI, dotnet CLI, VS Code or any other code ...

docker | netcore | windowscontainers | xamarin | spa | microservices | ddd | ddd-patterns

3,276 commits | 90 branches | 14 releases | 96 contributors | MIT

Branch: dev | New pull request | Create new file | Upload files | Find File | Clone or download

mvelosop Merge branch 'clean-up/move-ebooks' into dev — Latest commit 86f7d53 8 days ago

Components	Add GeolocatorPlugin-1.0.3 to IOS and Windows project	2 years ago
ServiceFabric	Added SF sln for gruping all sf projects	2 years ago
build	updated build def for webspa, mvc & status	3 months ago
cli-linux	Improve build-bits-linux.sh (removed dotnet restore (doing with publi...	2 years ago
cli-mac	Update project list in mac build script	2 years ago
cli-windows	Updateded to Show Message to run as administrator	5 months ago
deploy	typo	8 months ago
docs/Decks	e-books moved to https://github.com/dotnet-architecture/eBooks	8 days ago

eSHOP onCONTAINERS

E-commerce

Non-trivial

Actively Maintained

Containerized

Cross-platform

Great documentation

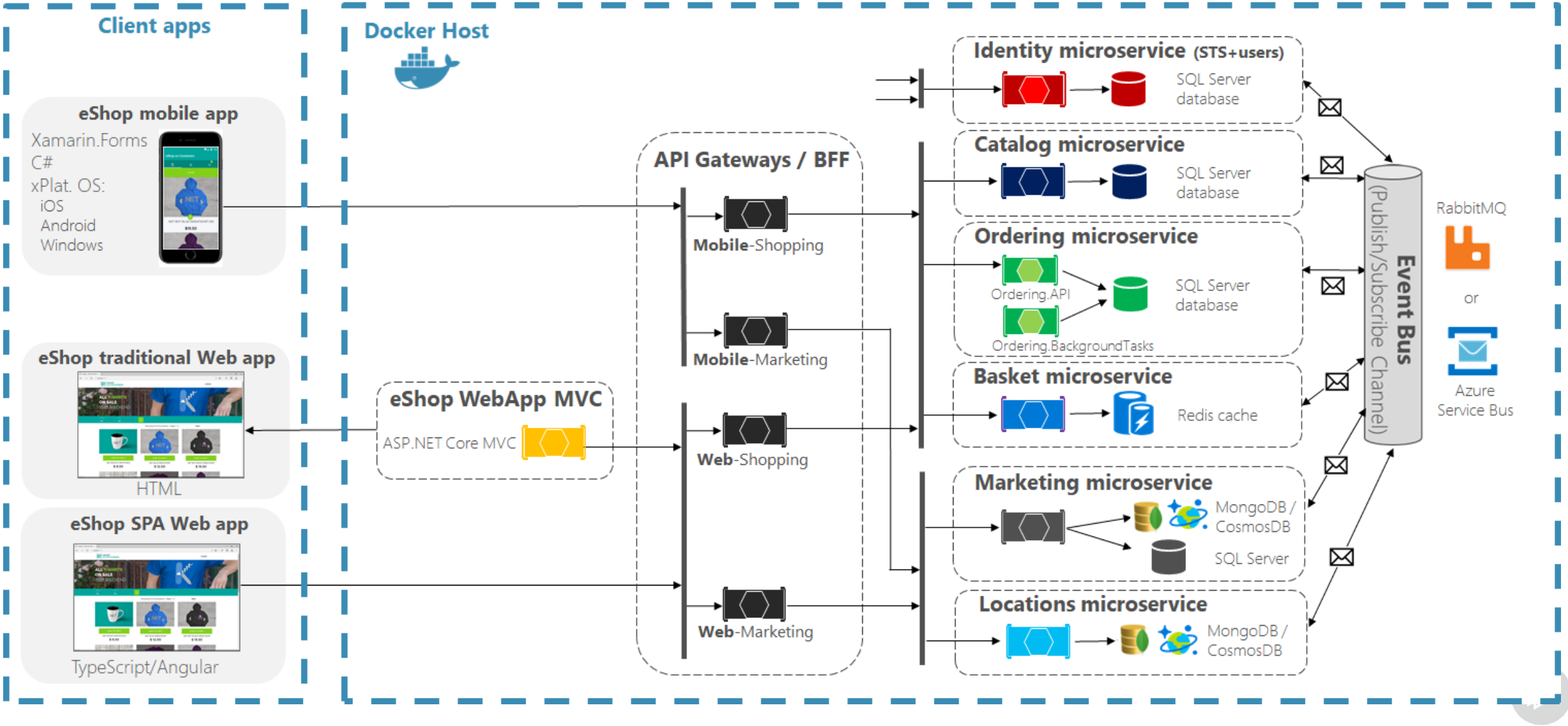
eShopOnContainers Tech Stack

ASP.NET Core

C#

Microservices principles can be applied in any programming language

eShopOnContainers Architecture

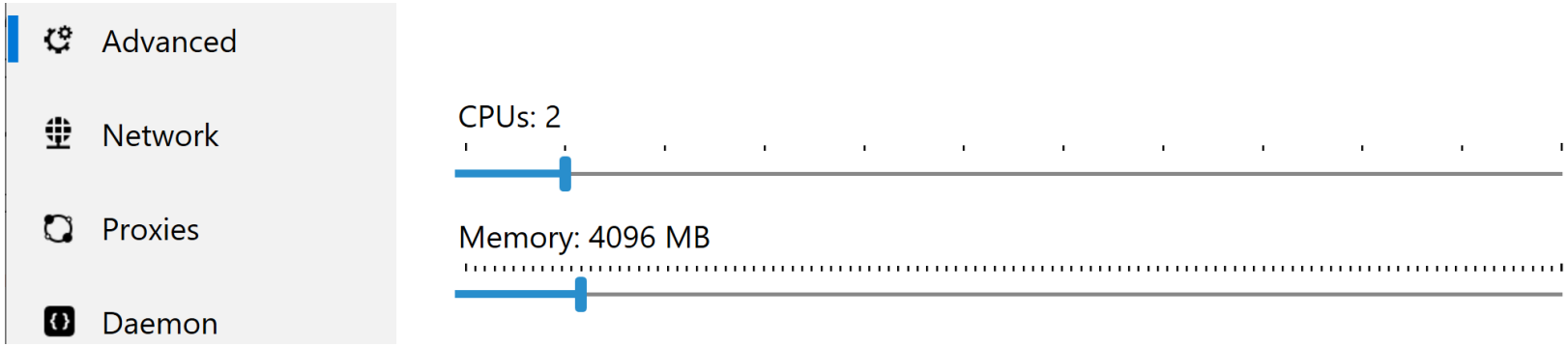


Development Environment Setup

1. **Install Docker Desktop (Windows or Mac)**
   https://www.docker.com/products/docker-desktop

2. **Clone eShopOnContainers source code**
   git clone https://github.com/dotnet-architecture/eShopOnContainers.git

3. **Increase Docker Desktop available memory to 4GB (in Advanced Settings)**



4. **Configure Windows firewall rules (using supplied PowerShell script)**
   .\cli-windows\add-firewall-rules-for-sts-auth-thru-docker.ps1

Demo

Running eShopOnContainers locally

- docker-compose up
- docker-compose.yml

Summary

Microservices give us freedom to choose the best tools for the job

Standardization can simplify development, deployment and monitoring

eShopOnContainers sample application

- ASP.NET Core and C#
- Standardized logging and health checks
- Different technologies (e.g. databases)

Up next...

Implementing domain logic