

Fundamentals of Docker and Kubernetes for .NET Developers

JUSTIFYING CONTAINERS AND PREPARING TO USE THEM



Erik Dahl

PRINCIPAL ARCHITECT

@dahlsailrunner knowyourtoolset.com



Overview



Clarifications

.NET is cross-platform

Containerization is valuable

- To developers
- For deployments

Start with .NET solution

Prepare workstation to use containers

- Docker Desktop
- Windows Terminal
- WSL2



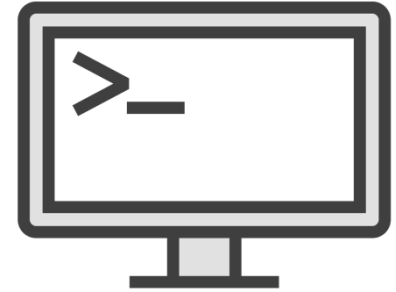
This Course IS...



Address “why?”



Tools you know



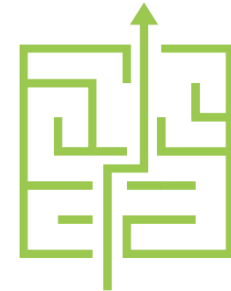
CLI as needed



Docker



Kubernetes



Navigation starter



This Course Is NOT...



ASP.NET guide



Deep reference



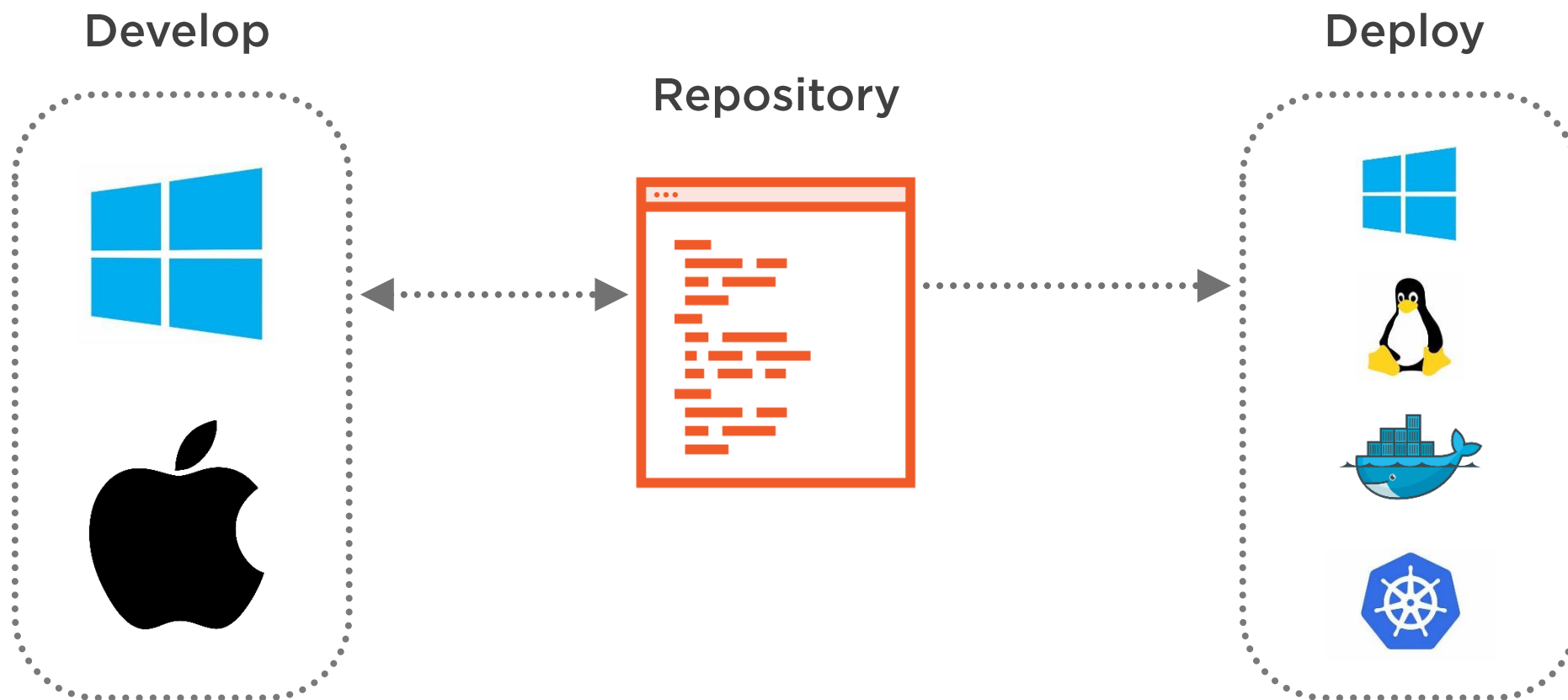
Deployment
recipes



Persistent
storage



.NET Is Cross-platform



Containers Benefit Development



Services

Pull, run, and remove public images easily

Seq, ELK, smtp4dev, redis, nginx, rabbitmq, kafka, ...



Simplify Setup

Clone repo, run Docker Compose project



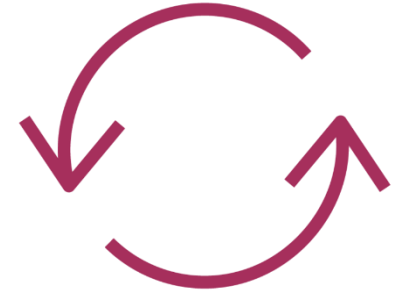
Containers Benefit Deployment



Scaling up and down



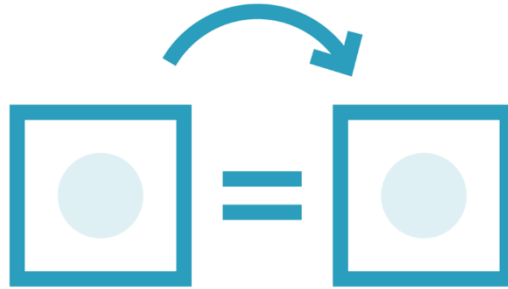
Deployments



Rollbacks



Monitoring



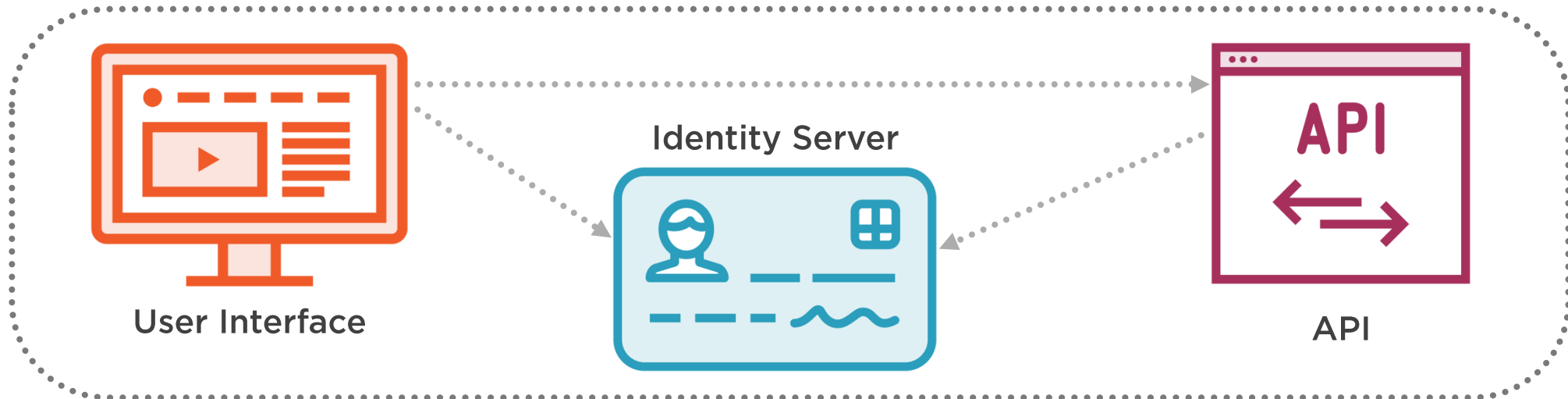
Environments



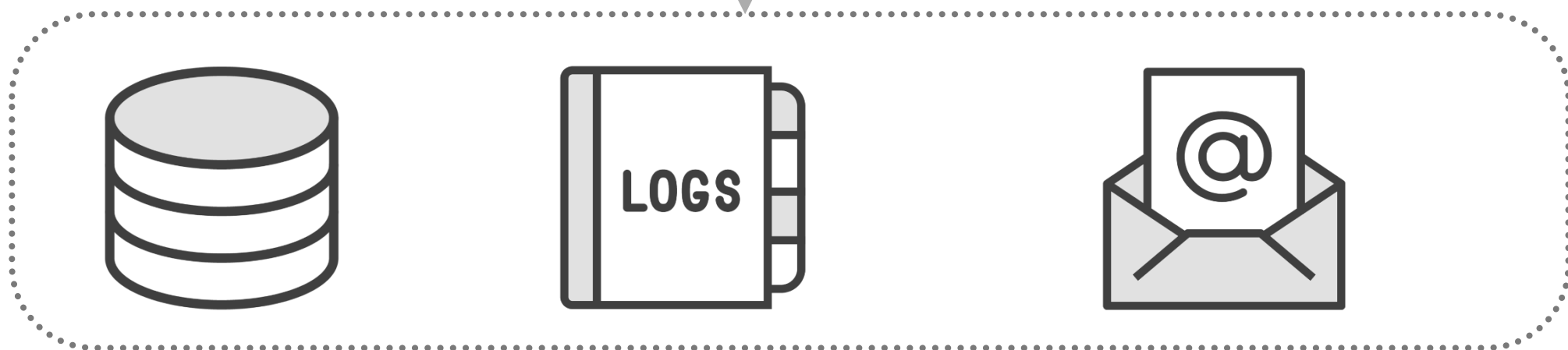
Cost

Solution Overview

Applications



Services



Demo



Set up existing .NET solution

- Focus more on steps than result
- Using .NET 5; Core 3.1 branch available

Install database

- Run SQL scripts

Install Papercut

Logging?

SSL via IIS Express / Visual Studio

- Cert creation / authority / install
- HOSTS file entry(ies)
- IIS setup – sites / apps / app pools



Welcome Aboard!



Let's get you set up!

Here's a list of 1,520 steps.....

Docker Compose can help

- Clone repo
- Run docker-compose project
- SSL may require trusting a cert and HOSTS entries



Demo



Get ready to start using containers

Install Docker Desktop

- Set up account

Install Windows Terminal

- Make it yours

Install Windows Subsystem for Linux (WSL) 2

- Install a Linux distribution (Ubuntu)



What About .NET Framework and Windows Containers?!?!



It's possible

- Newer
- Fewer examples / docs

You'll focus on Linux / .NET 5 / .NET Core

You'll learn concepts that you can apply

Don't "containerize all the things"

Summary



Cross-platform nature of .NET opens new doors

Containerization is valuable

- To developers
- For deployments

Ready to rock with containers!

- Docker Desktop
- Windows Terminal
- WSL2





Each module is a building block

- Use public images
- Containerize your own application(s)
- Run with a more complex setup with docker-compose
- Special handling for databases
- Kubernetes for deployments

Simplify *development* with all but k8s

