AWS State, Local, and Education Learning Days

Philadelphia

aws Learning Days
State, Local, and Education

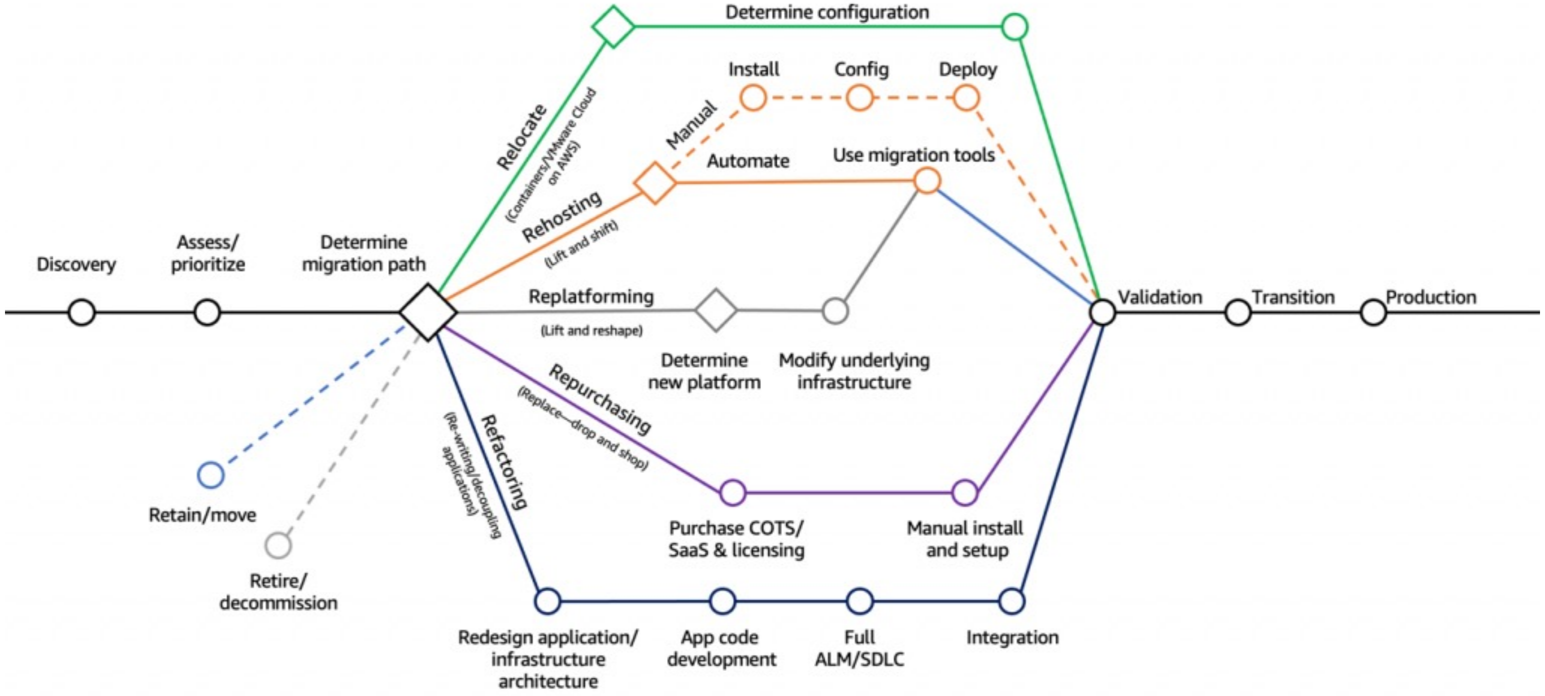
Transform and containerize .NET apps with AWS modernization tools

Bob Palermo

Sr. Solutions Architect
Amazon Web Services
bpalermo@amazon.com

Learning Days
State, Local, and Education

Application migration strategies: “The 7 R’s”



Refactoring/Re-architecting: Reimagining how the application is architected and developed, typically using cloud-native features

Refactor with Microservices

Use Cases

Launch a new feature, fast

- Enhance or extend the application with new features in microservices without touching the old application
- Quickly launch a new capability and expand your business
- Experiment with a new tech stack without risk

Refactor incrementally

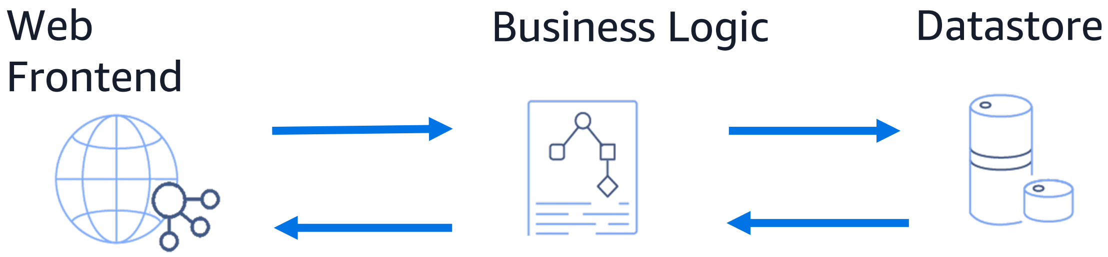
- Define the interface to the first piece to be refactored
- Design, implement, and test the new capability
- Redirect old to new
- Turn off old code or service

Refactor an application’s account structure first

- Safely and incrementally move your app to multiple AWS accounts
- Simplify achieving deployment independence
- Drive team independence and agility

Software Architecture

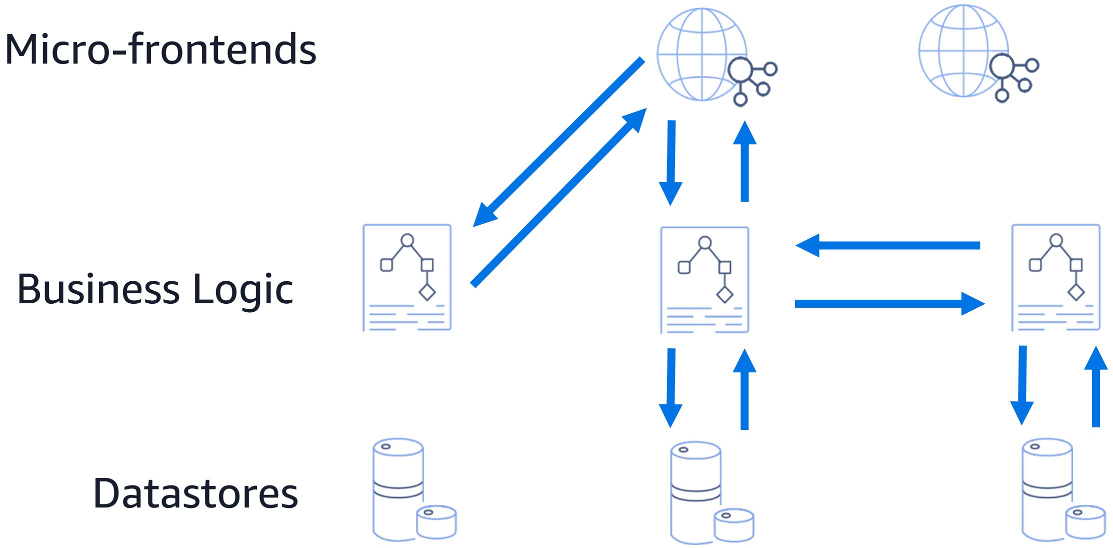
MONOLITHIC ARCHITECTURE



DRAWBACKS

- Hard to Iterate Fast
- Hard to Scale Efficiently
- Reliability Challenges
- CI/CD Time Consuming

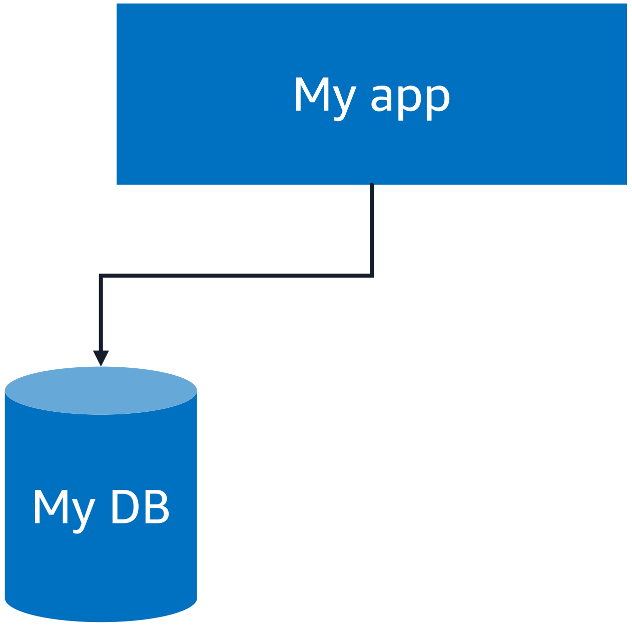
MICROSERVICES ARCHITECTURE

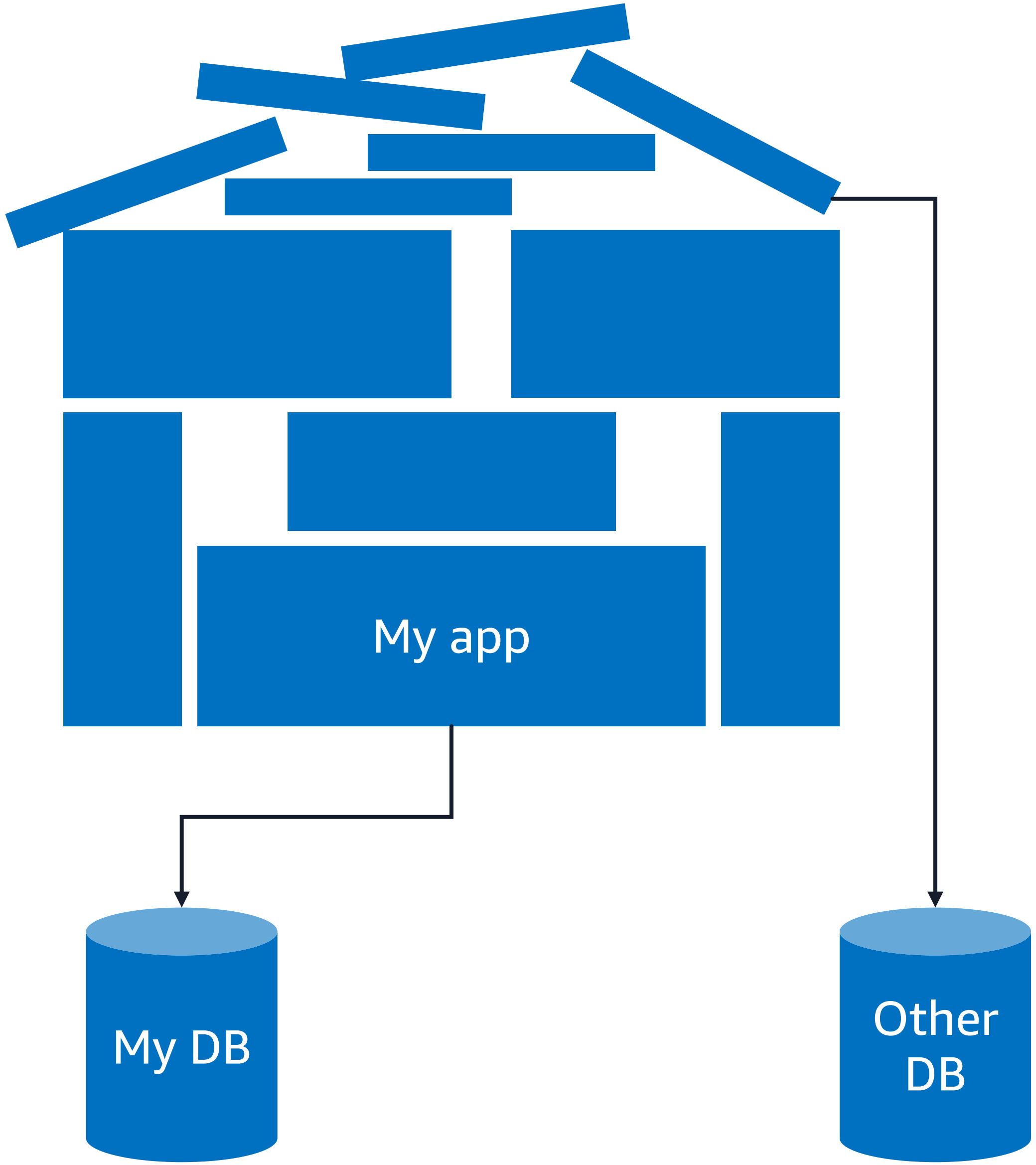


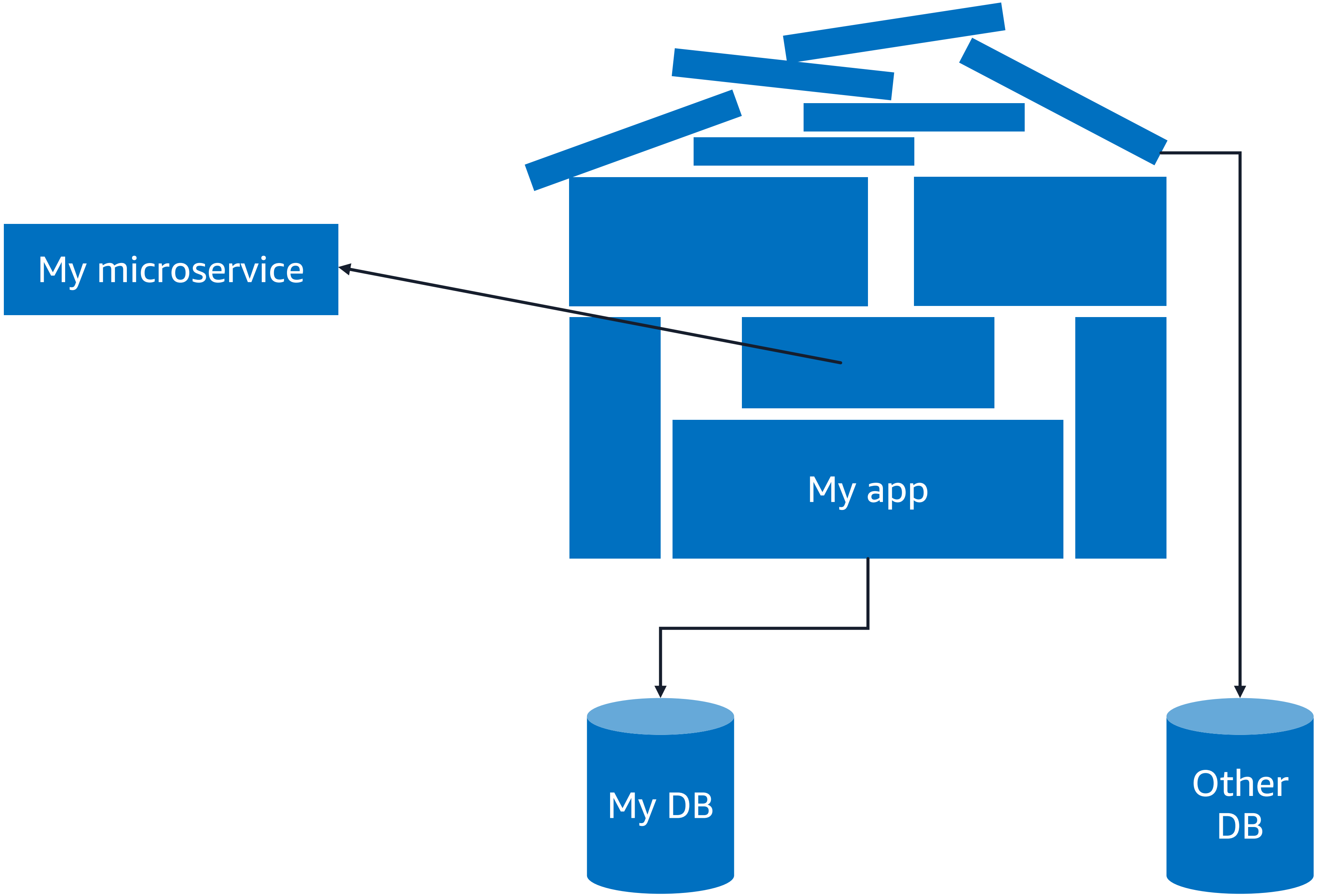
BENEFITS

- Increased Agility
- Easy to Scale
- Improved Innovation
- Reduced Human Errors

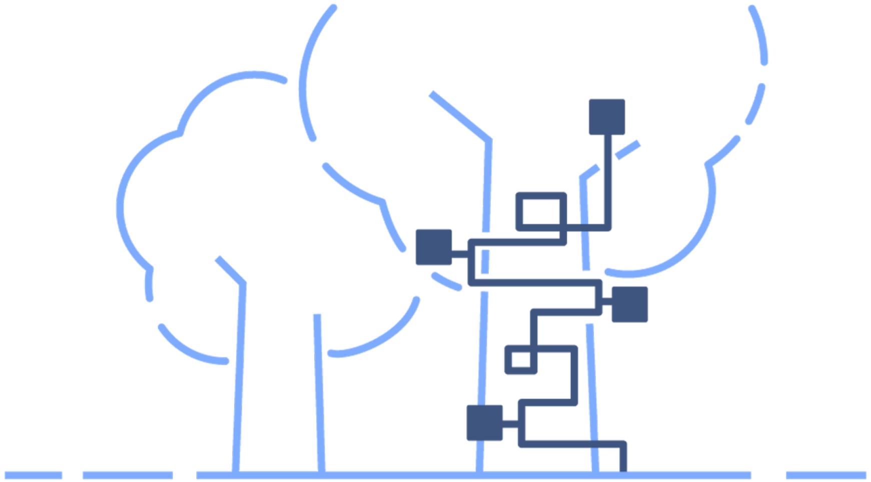
No one sets out to build a monolith




My app
My DB
Other DB


My microservice
My app
My DB
Other DB

Modernize with the strangler fig pattern



“Gradually create a new system around the edges of the old, letting it grow slowly over several years until the old system is strangled.”

Martin Fowler, Thoughtworks

Benefits

- Agility
- Frequent releases
- Lowers risk
- Immediate benefits
- Easier rollback

https://martinfowler.com/bliki/StranglerFigApplication.html

AWS Microservice Extractor

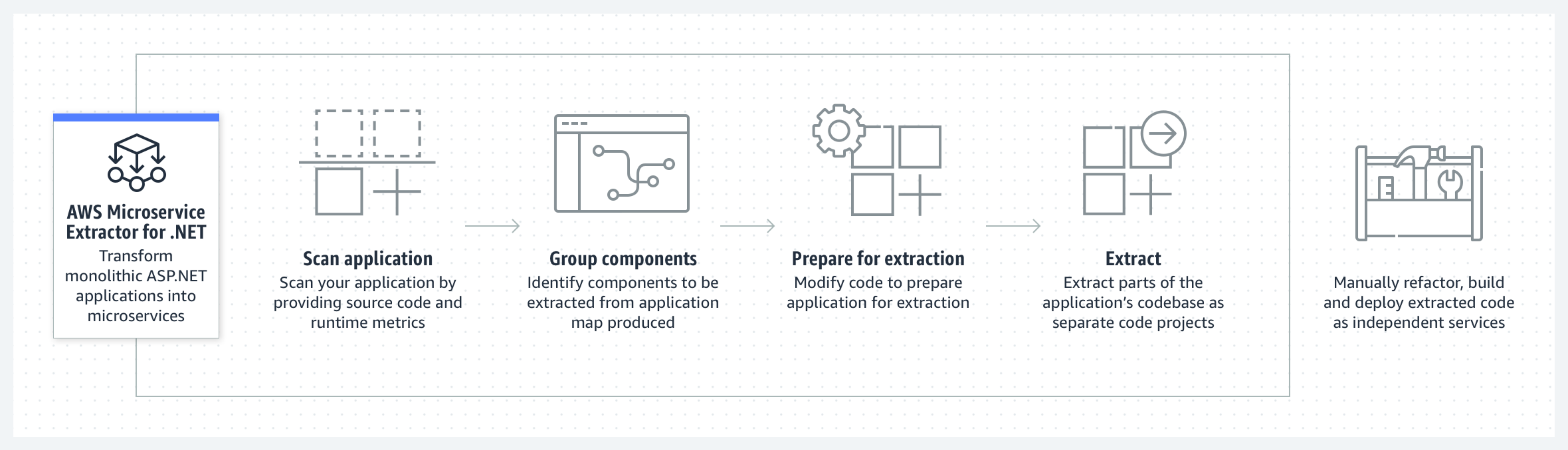
Microservice Extractor for .NET

An assistive tool that becomes both an ***insightful advisor*** and a ***robotic builder*** for enterprise developers to help renovate their applications for the cloud.

Assisted identification of parts of the application to carve out as separate services

Assisted extraction of code into separate projects and deployment alongside the residual monolith

How it works



AWS Microservice Extractor for .NET
Transform monolithic ASP.NET applications into microservices

Scan application
Scan your application by providing source code and runtime metrics

Group components
Identify components to be extracted from application map produced

Prepare for extraction
Modify code to prepare application for extraction

Extract
Extract parts of the application’s codebase as separate code projects

Manually refactor, build and deploy extracted code as independent services

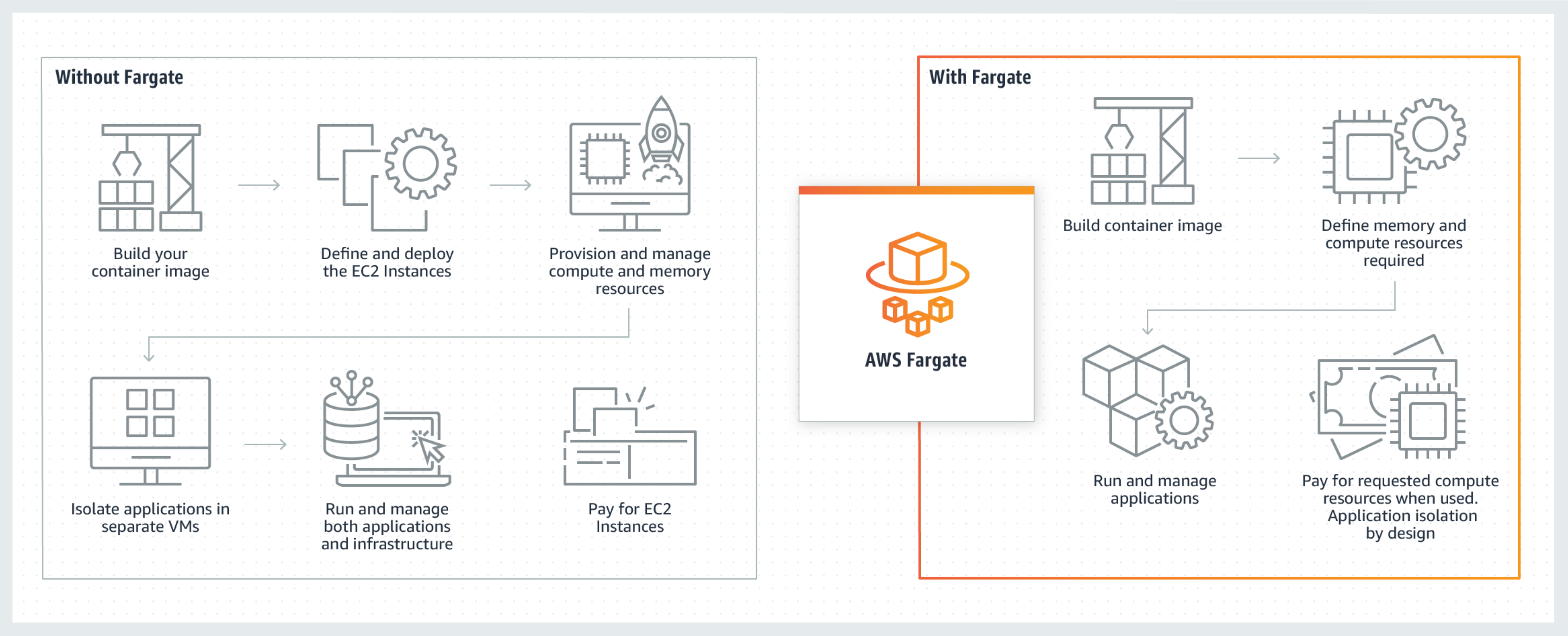
Containerization

Amazon Elastic Container Service and AWS Fargate

What is Amazon ECS?

- Container Management Service
- Makes it easy to run, stop, and manage Docker containers
- Manages container deployments across EC2 and serverless (Fargate) fleets
- Integrates with other AWS services including:
    - Amazon EC2 Load Balancers (Classic, NLB, ALB)
    - AWS Identity and Access Management (IAM)
    - Amazon CloudWatch metrics, Logs and Events
    - AWS Cloud Map and Amazon Route 53
    - Amazon Elastic Container Registry (ECR)

AWS Fargate
Serverless compute for containers


Without Fargate
Build your container image
Define and deploy the EC2 Instances
Provision and manage compute and memory resources
Isolate applications in separate VMs
Run and manage both applications and infrastructure
Pay for EC2 Instances
AWS Fargate
With Fargate
Build container image
Define memory and compute resources required
Run and manage applications
Pay for requested compute resources when used. Application isolation by design

Workshop Labs

Accessing the Workshop Labs

Transform and containerize .NET apps with AWS Modernization Tools

http://bit.ly/4hw5yy5

Learning Days
State, Local, and Education

Thank you!

Bob Palermo

Sr. Solutions Architect
Amazon Web Services
bpalermo@amazon.com

Please complete the survey for this session

Application Modernization, Security, and Governance

Transform and Containerize .NET apps with AWS modernization tools